

Roger Koenker

R Vinaigrettes

March 16, 2026



Preliminary

Preface

In a 1989 interview for *Econometric Theory* Nick Kiefer asked Art Goldberger, “How do you pick your research questions?” Goldberger replied, “Well nowadays I pick research questions when I get irritated. I have a nit-picking style. If there’s something that irritates me, or if I find something pretentious in somebody’s interpretation of results, I may feel compelled to look into the topic. And I work on something if it’s fun – I think that’s an important consideration.” This may not appear to be the apogee of late style as imagined by Edward Said, but it is honest and scientifically very sound.

When Rossini returned to Paris in 1855 after a relatively fallow period in Bologna, he began composing brief chamber works he called *péchés de vieillesse*, sins of old age, for the entertainment of friends and visitors, but perhaps mainly for his own enjoyment. For the last decade or so, I’ve begun writing a series of short notes about statistical odds and ends motivated in part, like Goldberger, by fun or irritation. These pieces have neither the length or gravitas to be considered by the serious academic journals, but as they have accumulated I began to think that taken together they might offer some encouragement to others to produce more pithy, less ponderous contributions to The Literature.

In this spirit in 2016 I wrote a manifesto for a new genre of statistical papers that I called R Vinaigrettes, which I reproduce here in full.

Knuth’s call for a literate programming style has spawned a new genre of statistical exposition, the R vignette, and thereby raised the dreary task of documenting computer code to the level of a minor art form, like finger painting or tap dancing. These vignettes are intended to reveal something of the authors contribution to the greater glory of data analysis, usually in the form of an R package.

This development has been enormously successful, and yet there is a general unease within the research community, a feeling that many of the almost 10,000 packages currently on CRAN have not received adequate vetting, or vignetting. In this spirit I would like to propose a new genre, the R vinaigrette. These would be brief communications that expose some feature, or bug, in the collective enterprise of statistical software. As the name suggests there should be something piquant about a vinaigrette, some lemon juice to balance the oils, or mustard, or vinegar. I would only insist that, like the vignette, the vinaigrette must be reproducible. Ideally, they

should also satisfy the Kolmogorov dictum that every single discovery should fit in a four-page Doklady note, since "the human brain is not capable of creating anything more complicated at one time." [quoted in Vitányi (2011).]

My friend and more than occasional coauthor, Ivan Mizera, once threatened to arrange to publish a few of these pieces posthumously, so I thought maybe I should save him some trouble.

Contents

1	On Distributional vs. Quantile Regressions	1
1.1	Introduction	1
1.2	Models and Estimators	1
1.3	A Logistic Specification	2
1.4	A Gumbel Specification	2
1.5	Discrete Response	4
1.6	Dicta and Contradicta	4
2	Bayes in the Baño: A Brief History	7
2.1	Conjugal Bayesian Up-Dating	7
3	Promiscuous Bayesian Up-dating	11
3.1	Gibbs Sampling	11
3.2	Metropolis-Hastings	12
4	Some Snarky Remarks on Bayesian Quantile Regression	25
4.1	A Most Perplexing Paradox	25
4.2	“Mommy, Where do likelihoods come from?”	26
5	The Method of Quantiles	29
6	The Targeted MLE: As I (Dimly) Understand It	35
6.1	Introduction	35
6.2	Prelude: Missing Data and Double Robustness	35
6.3	TMLE: A Missing Manual	38
6.4	Quantiles for Missing Data and QTEs	41
6.5	QTET Quintet + 1	47
6.5.1	The Proof of the Pudding	48
6.5.2	Simulation Protocol	50

7	A Gaussian Compound Decision Bakeoff	53
7.1	Four Experiments	53
7.2	Some Oracle Comparisons	56
7.3	Envoi	57
8	Maximum Entropy Deregularized Density Estimation	59
8.1	Introduction	59
8.2	Means of order ρ and a Hierarchy of Concave Densities	59
8.2.1	Estimation of log concave densities	60
8.2.2	Rényi likelihood and weaker forms of concave densities	61
8.3	Rényi likelihood and norm constrained density estimation	64
8.4	Discrete Implementation Details	66
8.5	Conclusion	68
9	Edgeworth on Banking	69
10	Bayesian Deconvolution	73
10.1	Introduction	73
10.2	The Kiefer-Wolfowitz NPMLE	74
10.3	Efron's NPMLE	74
10.4	An Illustration	75
11	Quantile Selection Models	77
11.1	Introduction	77
11.2	A QRious Likelihood	77
11.3	Estimation of the QR Selection Model	79
11.4	A Replication Exercise	81
11.5	Data Management	85
12	The “Tomato Salad Problem” Problem	87
12.1	Introduction	87
12.2	Reparameterization	87
12.3	Two Examples	88
12.4	Ill-behaviour of NPMLE and the tuning of grid size	89
13	Implementation of Rényi Penalized Density Estimation	93
13.1	Introduction	93
13.2	Theory	93
13.3	Praxis	95
13.3.1	Mosek 8	95
13.3.2	Mosek 9	96
13.4	Odds and Ends	97

14	The Group Lasso for Quantile Regression	99
14.1	Introduction	99
14.2	The Model	99
14.3	Computation	100
14.4	Computational Experience	100
14.5	Conclusion	104
14.6	R Coda	104
15	Bivariate Exchangeability	109
15.1	Introduction	109
15.2	Perception of Randomness	110
15.3	Conclusion	111
16	Empirical Bayes Confidence Intervals	113
16.1	Introduction	113
16.2	The Kiefer-Wolfowitz NPMLE	114
16.3	Efron's NPMLE	115
16.4	Some Simulation Evidence	115
16.5	Conclusion	120
17	Conformal Quantile Regression	121
17.1	Introduction	121
17.2	Total Variation Penalization of g Itself	125
17.3	Conclusion	125
18	Extremal Quantile Regression Inference	127
18.1	Introduction	127
18.2	Confidence Intervals for Extreme Regression Quantiles	127
18.3	Estimating Pareto Tail Exponents	128
18.4	Conclusion	131
19	Deracinated Quantile Regression	133
19.1	Introduction	133
19.2	Conclusion	136
20	λ-Selection	137
20.1	Introduction	137
20.2	The AIC Method	138
20.3	The k -fold Cross Validation Method	140
20.4	Conclusion	142
21	Quantile Regression Methods	143
21.1	Introduction	143
21.2	Algorithms for Point Estimation	143
21.2.1	Estimation with Many τ 's	145
21.3	Summary Methods	149

21.4	Bootstrap Methods for Quantile Regression	150
21.5	Smoothing, Gradient Descent and Other Perturbations	153
21.6	Conclusion	156
22	Computing Sample Quantiles	159
22.1	Introduction	159
22.2	Univariate Sample Quantiles	160
22.3	Weighted Quantiles	160
22.4	Updating Sample Quantiles	162
22.5	Conclusion	162
23	Pava and Pavlova	163
23.1	Introduction	163
23.2	Conclusion	167
24	Bradley-Terry and the Ranking Lasso	169
24.1	Introduction	169
24.2	Some Convex Optimization	169
24.3	Implementation	171
24.4	Regularized Median Logistic (Binomial) Regression	173
24.5	An Illustrative Example	174
24.6	Regularized Logistic Binomial Median Regression	176
25	Evaluating Covid Quantile Forecasts	179
25.1	Introduction	179
25.2	Scoring Rules for Quantile Forecasts	179
25.3	Ensemble Forecasting	180
25.3.1	Stacking for ordinary regression	182
25.3.2	Stacking for Quantile Models	182
26	Denoising the Banana	183
26.1	Introduction	183
26.2	This is not a banana	183
26.3	R Code for Bivariate Gaussian Mixtures	184
27	How Quantile Regression Rescued Happiness – for the Rich	189
27.1	Introduction	189
27.2	Replication	190
27.3	Conclusion	192
28	Borges and AI	193
28.1	Introduction	193
28.2	Natural Intelligence	193
28.3	What Now?	194

29	The Dantzig Decoder Ring	195
29.1	Introduction	195
29.2	The Dantzig Selector	196
29.3	Dantzig Decoding	196
29.4	A Simulated Example	196
29.5	Conclusion	198
30	Against Causality	201
30.1	Introduction	201
30.2	Regression Exchangeability	201
30.3	Je ne regrette rien	203
30.4	Structural Equation Models	203
31	Choquet Risk and Portfolio Optimization	205
31.1	Introduction	205
31.2	Pessimistic Risk Measures	206
31.3	Portfolio Optimization	207
31.4	A Simulation Exercise	208
31.5	Conclusion	211
A	Dictionary of Received Ideas of Statistics	213
B	A Simple Protocol for Simulations in R	221
B.1	Introduction	221
B.2	The Four Commandments	221
B.3	Some Implementation Details	221
B.4	The <code>foreach</code> Wrinkle	223
B.5	The Cluster	225
	References	229

On Distributional vs. Quantile Regressions

1.1 Introduction

Recent work by Peracchi (2002), Firpo, Fortin, and Lemieux (2009), Chernozhukov, Fernández-Val, and Melly (2013) and Rothe and Wied (2013) has suggested that estimating families of binary response models using varying “cutoffs” to construct the binary response may provide an attractive alternative to estimating conditional quantile functions. In some very simple iid error settings the two approaches can be directly compared; we undertake such a comparison for two leading forms of the binary response model.

1.2 Models and Estimators

Consider the linear model

$$Y_i = \alpha_0 + x_i^\top \beta_0 + u_i \quad i = 1, \dots, n, \quad (1.1)$$

with iid errors $\{u_i\}$ with distribution function, F , survival function, $S = 1 - F$, and density, f , we can estimate models for the family of binary response models

$$\begin{aligned} S^{-1}(P(Y_i > y | x_i)) &= S^{-1}(P(u_i > y - \alpha_0 - x_i^\top \beta_0)) \\ &= S^{-1}(1 - F(y - \alpha_0 - x_i^\top \beta_0)) \\ &= y - \alpha_0 - x_i^\top \beta_0. \end{aligned}$$

That is, we can define the indicator functions $I(Y_i > y)$ for a particular choice of the cutoff, y and estimate the binary response model with link function, S . The effect of changing the cutoff, y , is simply to shift the intercept of the model. Estimating a family of such models for a variety of cutoffs is termed “distributional regression” in Chernozhukov, Fernández-Val, and Melly (2013).

Under mild conditions, e.g. McCullagh and Nelder (1989), the binary response estimator, $\hat{\theta}_n(y) = (\hat{\alpha}_n(y), \hat{\beta}_n(y))$, is asymptotically Gaussian with covariance matrix $\hat{V}_n = (X^\top W X)^{-1}$ where $W = \text{diag}(w_i)$, and $w_i = f^2(\nu(y)) / (F(\nu(y))(1 - F(\nu(y))))$ where $\nu(y) = y - \alpha_0 - x_i^\top \beta_0$. The matrix X has typical row $(1, x_i^\top)$.

For the corresponding quantile regression estimator,

$$\check{\theta}_n(\tau) = (\check{\alpha}_n(\tau), \check{\beta}_n(\tau)) = \operatorname{argmin}_{a,b} \sum_{i=1}^n \rho_\tau(Y_i - a - x_i^\top b)$$

we have the limiting covariance matrix, $\check{V}_n = (\tau(1-\tau))/f^2(F^{-1}(\tau))(X^\top X)^{-1}$. If we take $y = F^{-1}(\tau)$ in the respective formulas we see that the two covariance matrices look quite similar. Indeed, if $\beta_0 = 0$ then they actually coincide. However, when $\beta_0 \neq 0$ the situation is a bit more complicated and we provide some numerical comparisons along with some simulation evidence on comparative performance in the next section.

1.3 A Logistic Specification

Consider the model (1.1) with $\{u_i\}$ iid from the logistic df , $F(u) = (1 + e^{-u})^{-1}$. This model gives rise to logistic binary response model

$$\begin{aligned} \operatorname{logit}(P(Y_i > y | x_i)) &= \operatorname{logit}(1 - F(y - \alpha - x_i^\top \beta)) \\ &= \operatorname{logit}(F(\alpha - y + x_i^\top \beta)) \\ &= \alpha - y + x_i^\top \beta \end{aligned}$$

At each value of the cutoff, y , we have a logistic regression model with intercept $\alpha - y$. In survival analysis this model is often called the Bennett model. A valuable general discussion of the correspondence between survival models and families of binary response models is given in Doksum and Gasko (1990). In contrast we have the quantile regression model

$$Q_{Y_i | X_i}(\tau | x_i) = \alpha + F_u^{-1}(\tau) + x_i^\top \beta$$

So both models have the same linear slope parameters, but the intercepts are different. However, for any choice of τ 's we can choose corresponding y 's to be $y_k = -F^{-1}(\tau_k)$.

A simulation of this logistic model with $n \in \{100, 500, 1000, 5000\}$, $\{x_i\}$ iid $\mathcal{N}(0, 1)$, $\alpha = 0$, and $\beta = 1$ yields the results reported in Table 1. Reported root mean squared errors are scaled by $\sqrt{n}$ so that they are comparable to the limiting standard errors given by the asymptotic theory described above. The limiting root mean squared errors were computed by numerically evaluating the limiting covariance matrices at the simulation settings with $n = 10,000$. Table 2 reports results for an identical experiment, except that the $\{x_i\}$ are iid log-normal. The results provide some evidence for the superior efficiency of the quantile regression approach.

1.4 A Gumbel Specification

A variety of other models can be accommodated in a similar fashion. As another illustration, consider the linear model (1.1) with $\{u_i\}$ iid from the Gumbel, or Type 1 extreme value distribution, $F(u) = 1 - e^{-e^u}$. Then

Table 1.1: Root Mean Squared Error in 1000 Replications: Bennett Model

	Quantile Regression				Distributional Regression			
	$\tau = 0.2$	$\tau = 0.4$	$\tau = 0.6$	$\tau = 0.8$	$\tau = 0.2$	$\tau = 0.4$	$\tau = 0.6$	$\tau = 0.8$
α								
$n = 100$	3.633	2.866	2.961	3.497	4.186	4.248	4.743	6.750
$n = 500$	3.219	2.619	2.631	3.245	3.916	3.865	4.414	5.973
$n = 1000$	3.253	2.654	2.661	3.243	3.771	3.790	4.316	5.974
$n = 5000$	3.141	2.491	2.516	3.255	3.929	3.880	4.361	6.070
$n = \infty$	2.500	2.042	2.042	2.500	2.796	2.249	2.253	2.808
β								
$n = 100$	1.583	1.291	1.331	1.601	2.925	3.736	5.009	9.048
$n = 500$	1.206	0.997	0.988	1.230	2.753	3.354	4.410	6.948
$n = 1000$	1.389	1.108	1.128	1.337	2.713	3.301	4.172	6.568
$n = 5000$	1.141	0.954	0.967	1.223	2.815	3.410	4.383	6.744
$n = \infty$	2.493	2.035	2.035	2.493	2.975	2.666	2.658	2.946

Table 1.2: Root Mean Squared Error in 1000 Replications: Bennett Model

	Quantile Regression				Distributional Regression			
	$\tau = 0.2$	$\tau = 0.4$	$\tau = 0.6$	$\tau = 0.8$	$\tau = 0.2$	$\tau = 0.4$	$\tau = 0.6$	$\tau = 0.8$
α								
$n = 100$	3.633	2.866	2.961	3.497	4.186	4.248	4.743	6.750
$n = 500$	3.219	2.619	2.631	3.245	3.916	3.865	4.414	5.973
$n = 1000$	3.253	2.654	2.661	3.243	3.771	3.790	4.316	5.974
$n = 5000$	3.141	2.491	2.516	3.255	3.929	3.880	4.361	6.070
$n = \infty$	2.500	2.042	2.042	2.500	2.796	2.249	2.253	2.808
β								
$n = 100$	1.583	1.291	1.331	1.601	2.925	3.736	5.009	9.048
$n = 500$	1.206	0.997	0.988	1.230	2.753	3.354	4.410	6.948
$n = 1000$	1.389	1.108	1.128	1.337	2.713	3.301	4.172	6.568
$n = 5000$	1.141	0.954	0.967	1.223	2.815	3.410	4.383	6.744
$n = \infty$	2.493	2.035	2.035	2.493	2.975	2.666	2.658	2.946

$$\begin{aligned}
P(Y_i > y|x_i) &= P(u_i > y - \alpha - x_i^\top \beta) \\
&= 1 - F(y - \alpha - x_i^\top \beta),
\end{aligned}$$

so

$$\log(-\log(P(Y_i > y|x_i))) = y - \alpha - x_i^\top \beta.$$

This is the complementary log-log binary response model and can be easily estimated using this link function, McCullagh and Nelder (1989). In survival analysis this model corresponds to the Cox proportional hazard model.

Again, we can compare performance of direct quantile regression estimation of the model with estimation of the model via the corresponding binary response estimator. In Table 3

Table 1.3: Root Mean Squared Error in 1000 Replications: Cox Model

	Quantile Regression				Distributional Regression			
	$\tau = 0.2$	$\tau = 0.4$	$\tau = 0.6$	$\tau = 0.8$	$\tau = 0.2$	$\tau = 0.4$	$\tau = 0.6$	$\tau = 0.8$
α								
$n = 100$	1.248	1.315	1.587	2.188	1.662	1.550	1.866	2.796
$n = 500$	1.242	1.371	1.652	2.275	1.674	1.573	1.832	2.565
$n = 1000$	1.237	1.368	1.630	2.264	1.542	1.571	1.855	2.675
$n = 5000$	1.292	1.384	1.603	2.253	1.562	1.589	1.818	2.514
$n = \infty$	1.243	1.337	1.599	2.241	1.552	1.541	1.821	2.544
β								
$n = 100$	1.316	1.419	1.695	2.427	2.645	2.387	2.530	2.951
$n = 500$	1.245	1.309	1.548	2.089	2.264	2.131	2.143	2.452
$n = 1000$	1.233	1.275	1.586	2.190	2.302	2.082	2.078	2.343
$n = 5000$	1.232	1.339	1.659	2.266	2.205	2.046	2.037	2.351
$n = \infty$	1.239	1.333	1.594	2.234	2.250	2.082	2.078	2.339

we report results from another simulation exercise structured exactly as in the first logistic case except that u_i 's are Gumbel and the binary response uses the complementary log-log link function.

1.5 Discrete Response

Another rationale that has been advanced for preferring the $\hat{F}$ approach over the $\hat{Q}$ approach involves the existence of mass points for the response variable, for example points corresponding to the minimum wage in labor economics wage regressions. The details of this argument are a bit unclear, but it may be worth noting that in the simplest setting of iid observations from a discrete random variable, the sample quantiles converge faster than $\sqrt{n}$ except on a subset of measure zero of $(0, 1)$, while $\hat{F}$ converges at its usual $\sqrt{n}$ rate. This observation extends naturally to p -sample data, and to regression settings as well unless there is some form of misspecification. There must be more to this than meets the eye.

1.6 Dicta and Contradicta

Given the parameterization we have considered we have seen that the quantile regression estimator is considerably more accurate than the distributional regression estimator. Since the latter estimator also requires us to make a potentially controversial choice of an appropriate link function, this would seem to be a compelling argument for the quantile regression approach.

On the contrary, it should be remembered that two swallows do not make a summer, and likewise other choices of the iid error distribution and its corresponding link function may well yield different conclusions. Perhaps more crucially, we may prefer to evaluate

performance of the two methods differently: according to how well they do in predicting conditional probabilities, rather than – as implicitly done here – conditional quantiles, and this too may produce contradictory evidence.

Bayes in the Baño: A Brief History

“We will all be Bayesians in 2020, and then we can be a united profession.” D.V. Lindley’s 1995 interview with A.F.M Smith, *Statistical Science*.

“I have lamented that Bayesian statisticians do not stick closely enough to the pattern laid down by Bayes himself: if they would only do as he did and publish posthumously we should all be saved a lot of trouble.” [M. Kendall, On the Future of Statistics, JRSS(A), (1968), 131, 182-204].

2.1 Conjugal Bayesian Up-Dating

The seed from which the great forest of Bayesian statistics grew and was cut down to produce journal articles is the Gaussian mean model with conjugate priors.¹ We briefly review some simple instances of this type before venturing into the “new growth” forest.

Article 18 of the UN Universal Declaration of Human Rights asserts that we are all entitled to our own opinions. However, Bayesians are encouraged to formulate opinions that are conjugal to their models for how observations of the world arise. (Religious dogma usually dictates how we see the world and what we believe about it, so this should hardly be surprising.)

Conjugal prior opinions have the property that they are capable of procreation when suitably coupled with associated probabilistic expressions of the likelihood of observed data. Offspring so generated must take the same form as the parental prior opinion, otherwise they are regarded as sterile, like a mule. We now illustrate this notion of mathematico-sexual compatibility with a series of conjugal examples.

¹ The material surveyed here is available from many sources, Appendix C of Jackman (2009) provides an especially nicely organized collection of detailed proofs.

Example 1. Gaussian mean, known scale. Suppose we have observations $X_1, \dots, X_n$ drawn iid from a $\mathcal{N}(\mu, \sigma^2)$ model with unknown μ and known σ^2 . If our prior opinion about μ happens to be expressed as $\mathcal{N}(\mu_0, \sigma_0^2)$ then a simple computation reveals that our updated opinion about μ should be $\mathcal{N}(\mu_1, \sigma_1^2)$ where $\sigma_1^2 = (n\sigma^{-2} + \sigma_0^{-2})^{-1}$ and $\mu_1 = \sigma_1^2(n\bar{X}\sigma^{-2} + \sigma_0^{-2}\mu_0)$. If, to make things even more convenient, $\sigma_0^2 = \sigma^2$, then the prior acts just like we have one more observation, $X_{n+1} = \mu_0$. But more generally $\bar{X}$ and μ_0 get weighted according to their respective precisions, n/σ^2 for $\bar{X}$ and σ_0^{-2} for μ_0 and combined accordingly. A curious feature of this schema is that if $\bar{X}$ and μ_0 differ considerably relative to their respective precisions, then the updated posterior opinion may express the puzzling view that we now believe that μ lies, with high probability, in a region that neither the observations, nor the prior, suggest is very plausible.

The proof of this result relies on the following elementary algebraic identity:

$$\gamma_0(x - \mu_0)^2 + \gamma_1(x - \mu_1)^2 = (\gamma_0 + \gamma_1) \left(x - \frac{\gamma_0\mu_0 + \gamma_1\mu_1}{\gamma_0 + \gamma_1} \right)^2 + \frac{\gamma_0\gamma_1}{\gamma_0 + \gamma_1}(\mu_1 - \mu_0)^2.$$

Example 2. Gaussian mean, unknown scale I. Suppose that we don't presume to know σ^2 in the previous example, what then? There are many ways to express our prior ignorance about the joint distribution of μ and σ^2 , but of these only a few are conjugal. A curious aspect of all this is that we are about to replace our admitted ignorance about σ^2 , with a very precise expression of its probability distribution. The simplest of these conjugal priors is probably the joint density, $f(\mu, \sigma^2) = g(\mu|\sigma^2)h(\sigma^2)$ with g the conditional Gaussian density

$$g(\mu|\sigma^2) = \sqrt{n_0}\phi(n_0(\mu - \mu_0)^2/\sigma^2)/\sigma$$

and σ^2 as inverse gamma, $\text{inv}\Gamma$, with density,

$$h(\sigma^2) = K(\sigma^2)^{-(\nu_0+2)/2} \exp\{-\nu\sigma_0^2/(2\sigma^2)\}.$$

This formulation of the prior essentially asserts that our prior belief about μ is based on having already seen n_0 observations from exactly the same process generating the X_i with μ_0 taken to be the sample mean from this “precognition.” Cf. Spielberg (2002) Under these circumstances Bayesian mating of prior and likelihood yield the normal/inverse gamma posterior progeny:

$$\begin{aligned} \mu|\sigma^2, X &\sim \mathcal{N}(\mu_1, \sigma^2/n_1) \\ \sigma^2|X &\sim \text{inv}\Gamma(\nu_1/2, \nu_1\sigma_1^2/2), \end{aligned}$$

where $\mu_1 = (n_0\mu_0 + n\bar{X})/(n_0 + n)$, $n_1 = n_0 + n$, $\nu_1 = \nu_0 + n$, $\nu_1\sigma_1^2 = \nu_0\sigma_0^2 + S + n_0n(\mu_0 - \bar{X})^2/n_1$, and $S = \sum (X_i - \bar{X})^2$. The proof of this involves some fairly tedious algebra.

Note that this posterior no longer has a fixed scale; our prior belief revealed that we were uncertain about σ^2 and after seeing the data we are still uncertain. If we are interested in the posterior marginal distribution of μ it is easily approximated by simulation: draw σ^2 's from the specified inverse gamma distribution and plug them into the expressions for μ_1 and σ_1^2 to draw Gaussians. Now mumble the magical Bayesian incantation: *Simulation is Revelation*.

A limiting case of the foregoing takes $\nu_0 = 0$, in which case $\sigma_1 = S/n$ and we are back to something closely resembling classical likelihood based results. However, $\nu_0 = 0$, while yielding a proper inverse gamma posterior, is not itself proper, and therefore does not satisfy the conditions set forth above for conjugality.

Note also that in contrast to the previous example, the scale parameter of the posterior for σ^2 has a term involving $(\mu - \bar{X})^2$, so when the prior and the data disagree the uncertainty about both μ and σ^2 are increased.

Example 3. Gaussian mean, unknown scale II. Suppose instead of prior beliefs about μ depending on σ^2 as in the previous example, we were to assume that μ and σ^2 were stochastically independent with $\mu \sim \mathcal{N}(\mu_0, \sigma_0^2)$ and $\sigma^2 \sim \text{inv}\Gamma(\nu_0/2, \nu_0\sigma_0^2/2)$. With this prior, Bayesian copulation yields the posterior,

$$\begin{aligned}\mu|\sigma^2, X &\sim \mathcal{N}(\mu_1, \sigma_2^2/n_1) \\ \sigma^2|X &\sim \text{inv}\Gamma(\nu_1/2, \nu_1\sigma_1^2/2),\end{aligned}$$

where $\nu_1 = \nu_0 + n$, $\mu_1 = \sigma_1^2(n\sigma^{-2}\bar{X} + \sigma_0^{-2}\mu_0)$, $\sigma_2^2 = (n\sigma^{-2} + \sigma_0^{-2})^{-1}$, and $S = \sum (X_i - \bar{X})^2$. Note that when the prior takes this independent form, the gap between μ_0 and $\bar{X}$ is no longer contributing to the scale of the posterior for σ^2 .

Example 4. Gaussian Regression, unknown scale I. Expanding slightly on the earlier Example 2, suppose that we now have observations $Y_1, \dots, Y_n$ that are multivariate Gaussian, $\mathcal{N}(X\beta, \sigma^2 I_n)$ where X is a matrix of full column rank, p . We now need a multivariate Gaussian prior for the p -vector, β , which we will take to be $\mathcal{N}(\beta_0, \sigma^2 \Omega_0)$ and retain the inverse gamma prior for σ^2 . The posterior becomes,

$$\sigma^2|Y \sim \text{inv}\Gamma(\nu_1/2, \nu_1\sigma_1^2/2),$$

but now,

$$\beta|\sigma^2 Y \sim \mathcal{N}(\beta_1, \sigma^2 \Omega_1)$$

where $\beta_1 = \Omega_1(\Omega_0^{-1}\beta_0 + X^\top X\hat{\beta})$, $\Omega_1 = (\Omega_0^{-1} + X^\top X)^{-1}$, $\nu_1 = \nu_0 + n$, $\nu_1\sigma_1^2 = \nu_0\sigma_0^2 + S + R$, $\hat{\beta} = (X^\top X)^{-1}X^\top Y$, $S = \|Y - X\hat{\beta}\|^2$, and $R = \|\beta_0 - \hat{\beta}\|_{\Omega_1}^2$. This matrix weighted average is a natural generalization of the earlier Example 2.

Example 5. Gaussian Regression, unknown scale II. Similarly we can extend Example 3 to the regression setting. With independent priors, $\beta \sim \mathcal{N}(\beta_0, \Omega_0)$ and $\sigma^2 \sim \text{inv}\Gamma(\nu_0/2, \nu_0\sigma_0^2/2)$, Bayesian updating yields the posterior,

$$\begin{aligned}\sigma^2|Y &\sim \text{inv}\Gamma(\nu_1/2, \nu_1\sigma_1^2/2), \\ \beta|\sigma^2 Y &\sim \mathcal{N}(\beta_1, \sigma^2 \Omega_1)\end{aligned}$$

with $\nu_1 = \nu_0 + n$, $\nu_1\sigma_1^2 = \nu_0\sigma_0^2 + S$, $\beta_1 = \Omega_1(\Omega_0^{-1}\beta_0 + \sigma^{-2}X^\top X\hat{\beta})$, and $\Omega_1 = (\Omega_0^{-1} + \sigma^{-2}X^\top X)^{-1}$.

Such examples can be expanded *ad nauseum* typically continuing with Student t priors for $\hat{\beta}$, but we will resist the temptation to move in this direction.

Promiscuous Bayesian Up-dating

Sexual intercourse began
In nineteen sixty-three
(which was rather late for me) -
Between the end of the Chatterley ban
And the Beatles' first LP.
Phillip Larkin, *Annus Mirabilis*

The gap of more than 30 years between Metropolis et al. (1953) and Gelfand and Smith (1990) can still be partially attributed to the lack of appropriate computing power, . . .

C. Robert and G. Casella, (2008) A History of MCMC.

When priors are not conjugal we can't expect fertile offspring, but some forms of updating may nevertheless still be possible. Indeed, quite a lot of this seems to have been going on since about 1953, but it is only within the last two decades that it has "taken off." The simplest form of this updating is Gibbs sampling, attributed to the American physicist, J. Willard Gibbs (1839-1903), even though he – in all likelihood – had no clue about it.

I will not attempt to recapitulate the rationale for Gibbs sampling, a nice elementary exposition is available by Casella and George (1992). Instead, I will describe an application to estimating a monotone regression function as proposed by Neelon and Dunson (2004).

3.1 Gibbs Sampling

Gibbs sampling relies on the observation that (under conditions we will not go into) successive sampling from a complete set of conditional distributions generates a Markov chain whose stationary distribution will be a valid joint posterior distribution. To illustrate this approach we consider a simple Gaussian nonparametric regression model,

$$y_i = g(x_i) + u_i$$

where the u_i are iid $\mathcal{N}(0, \sigma^2)$ and $g : \mathbb{R} \rightarrow \mathbb{R}$ is a continuous monotone increasing function. Following Neelon and Dunson (2004) we will parameterize $\hat{g}$ by a linear spline with fixed knots, using the basis expansion,

$$g(x) = \beta_0 + \sum_{j=1}^p \varphi_j(x) \beta_j,$$

where $\varphi_j(x) = \max\{\min\{x, \xi_j\} - \xi_{j-1}, 0\}$. Given this parameterization, g is monotone iff all the $\beta_j \geq 0$, and we have the likelihood,

$$\mathcal{L}(y|\beta, \sigma^2) = (2\pi\sigma^2)^{-n/2} \exp\{-\frac{1}{2\sigma^2} \|y - X\beta\|_2^2\}.$$

The prior on β is complicated by the constraint that the last p coordinates should be non-negative. To keep the situation semi-conjugal, the prior is formulated in terms of a latent vector, β^* taken to be Gaussian with $\beta_j = I(\beta_j^* > \delta) \beta_j^*$, for $j = 2, \dots, p$. To impose some smoothness on *hatg*, the prior asserts that β_j^* has mean β_{j-1}^* and common variance $\nu^2 = \lambda^{-2}$ for $j = 2, \dots, p$, i.e.

$$\pi(\beta^*) = \phi(\beta_0^*, \mu_0, \sigma_0) \phi(\beta_1^*, \mu_1, \sigma_1) \prod_{j=2}^p \phi(\beta_j^*, \beta_{j-1}^*, \nu).$$

Some hyperparameters are fixed, so $\mu_0 = \mu_1 = 0$, $\sigma_0 = \sigma_1 = \sqrt{10}$ and $\delta = 0.05$. The parameters σ^2 and ν^2 are taken to be inverse gamma, with rather uninformative parameters.

The conditional posterior distributions described in the Appendix of Neelon and Dunson (2004) are all straightforward conjugal computations except for the transition from their (A.1) to the next expression, say (A.2). The proportionality claimed, on closer examination, follows from the fact that the ratios of the two respective terms have the same omitted factor. Given (A.2) it is easy to compute the integrating constant for the conditional posteriors of the β_j^* 's and show that each of them takes the form of a mixture of two truncated normals. Further details are relegated to the code appearing on the group webpages.

In Figure 1 we illustrate a simple example of the use of this technique. The target function is Beta distribution function depicted in black; there are 100 observations generated with iid Gaussian error. The unconstrained least squares linear spline estimate appears in red, and the mean of the posterior based on 10,000 MCMC iterations appears in blue.

3.2 Metropolis-Hastings

Gibbs sampling may be regarded as a special case of the Metropolis-Hastings MCMC approach that employs a form of rejection sampling.¹ To illustrate this more general technique we will briefly describe an application to estimation and inference about univariate quantiles due to Dunson and Taylor (2005).

¹ Paraphrasing Segal (1970), Gibbs means never having to reject.

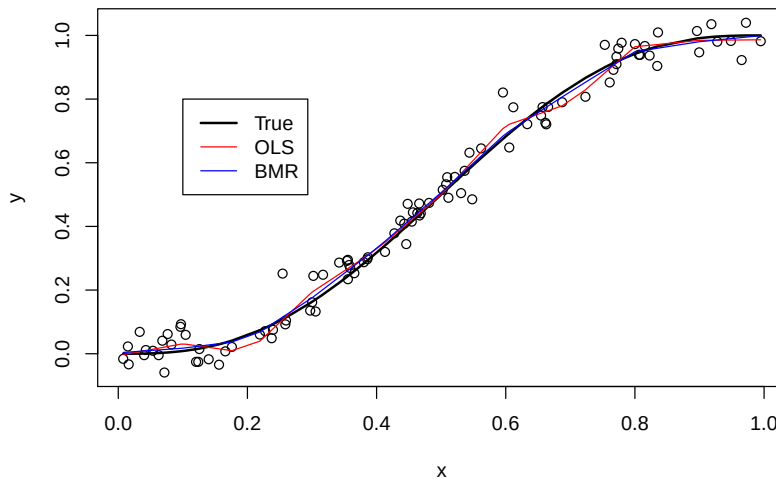


Fig. 3.1: Three Normal Mixture Densities.

Bayesian inference about quantiles is problematic because there is no obvious likelihood for quantiles unless we specify a full global parametric model for the observed data and this is contrary to the think-local mindset of the quantilogue. One of the high priests of Bayesianism offers a way out of this impasse; Jeffreys (1939) §4.4 considers inference about the median and proposes the “substitute likelihood”

$$s(\theta) = \binom{n}{k(\theta)} \left(\frac{1}{2}\right)^2$$

where $k(\theta) = \sum_{i=1}^n I_{(-\infty, \theta]}(Z_i)$. This is easily generalized as in Lavine (1995) to a vector of quantiles, $\theta(\tau_i)$ for $0 < \tau_1 < \dots < \tau_m < 1$ as

$$s(\theta) = \binom{n}{k_1(\theta) \dots k_{m+1}(\theta)} \prod \Delta \tau_i^{k_i(\theta)}$$

with $k_j(\theta) = \sum_{i=1}^n I_{(\theta_{j-1}, \theta_j]}(Z_i)$. Given this “likelihood” we need only formulate a prior and construct a proposal distribution for the MCMC iterations.

Dunson and Taylor (2005) adopt a simple Gaussian prior: $\pi(\theta) \sim \mathcal{N}(\theta_0, \Omega_0)$, but truncated so that $\theta_1 < \theta_2 < \dots < \theta_m$. The proposal distribution is constructed by simply updating in the usual conjugal fashion this prior with an estimate of the usual large sample approximation of the distribution of the sample quantiles. That is, the proposal distribution is $\theta \sim \mathcal{N}(\hat{\theta}, \hat{\Omega})$ truncated to assure ordered θ_i ’s with

$$\hat{\theta} = \hat{\Omega}(\Omega_0^{-1}\theta_0 + \hat{\Omega}_1^{-1}\hat{\theta}_1)$$

$$\hat{\Omega} = (\Omega_0^{-1} + \hat{\Omega}_1^{-1})^{-1}$$

$\hat{\theta}_1$ is a vector of sample quantiles, and $\hat{\Omega}_1$ is an estimate of its asymptotic covariance matrix.

The Metropolis-Hastings iteration proceeds as follows:

0. Set $\theta^{(0)} = \hat{\theta}$.
1. Draw a candidate θ from $\mathcal{N}(\hat{\theta}, \hat{\Omega})$
2. Reject if θ isn't monotone,
3. Else set $\theta^{(t)} = \theta$ with probability

$$p = \min\left\{1, \frac{s(\theta)\pi(\theta)}{s(\theta^{(t-1)})\pi(\theta^{(t-1)})}\right\}$$

or $\theta^{(t)} = \theta^{(t-1)}$ with probability $1 - p$.

A small simulation experiment designed to replicate the simulations of Dunson and Taylor (2005) was carried out. There are three target distributions, all mixtures of normals illustrated in Figure 1. Four sample sizes, $\{25, 50, 100, 200\}$ were studied, and 200 replications per setting were done. Tables 1-3 reports results for the root mean squared error comparison of the ordinary sample quantiles and the sample mean of the MCMC posterior iterations. In each simulation setting 5000 MCMC iterations were done with the last 4000 used to compute the posterior means.

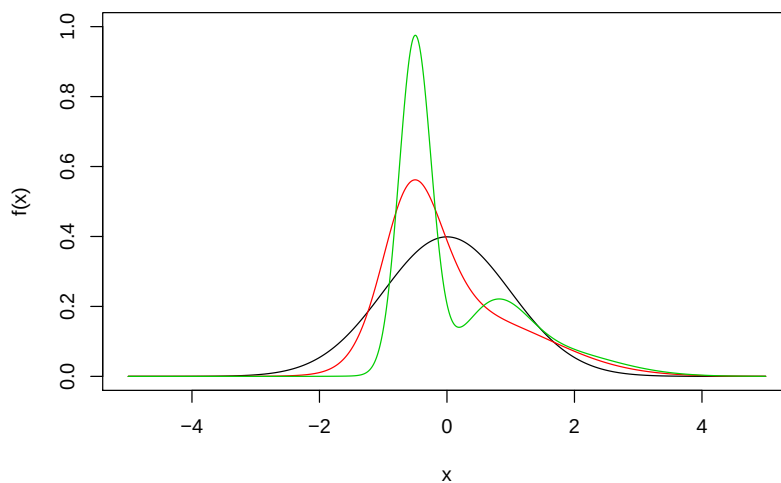


Fig. 3.2: Three Normal Mixture Densities.

Distribution 1	$\tau = 0.1$	$\tau = 0.25$	$\tau = 0.5$	$\tau = 0.75$	$\tau = 0.9$
SampleQ					
$n = 25$	0.353	0.277	0.241	0.265	0.333
$n = 50$	0.246	0.217	0.164	0.194	0.241
$n = 100$	0.177	0.127	0.109	0.124	0.160
$n = 200$	0.125	0.102	0.085	0.100	0.134
BayesQ					
$n = 25$	0.240	0.192	0.183	0.197	0.220
$n = 50$	0.180	0.169	0.157	0.153	0.177
$n = 100$	0.153	0.118	0.110	0.109	0.137
$n = 200$	0.114	0.097	0.086	0.096	0.122

Table 3.1: Root Mean Square Errors: Normal Mixture 1 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distribution 2	$\tau = 0.1$	$\tau = 0.25$	$\tau = 0.5$	$\tau = 0.75$	$\tau = 0.9$
SampleQ					
$n = 25$	0.188	0.173	0.201	0.377	0.472
$n = 50$	0.144	0.117	0.144	0.272	0.385
$n = 100$	0.086	0.082	0.104	0.193	0.251
$n = 200$	0.070	0.059	0.069	0.137	0.179
BayesQ					
$n = 25$	0.239	0.145	0.171	0.235	0.295
$n = 50$	0.133	0.103	0.135	0.210	0.255
$n = 100$	0.087	0.077	0.097	0.157	0.197
$n = 200$	0.066	0.054	0.065	0.123	0.154

Table 3.2: Root Mean Square Errors: Normal Mixture 2 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distribution 3	$\tau = 0.1$	$\tau = 0.25$	$\tau = 0.5$	$\tau = 0.75$	$\tau = 0.9$
SampleQ					
$n = 25$	0.099	0.101	0.228	0.462	0.456
$n = 50$	0.072	0.068	0.158	0.314	0.352
$n = 100$	0.051	0.046	0.095	0.219	0.267
$n = 200$	0.038	0.032	0.060	0.145	0.173
BayesQ					
$n = 25$	0.325	0.104	0.231	0.342	0.293
$n = 50$	0.164	0.062	0.175	0.276	0.263
$n = 100$	0.051	0.045	0.124	0.215	0.211
$n = 200$	0.033	0.031	0.069	0.155	0.154

Table 3.3: Root Mean Square Errors: Normal Mixture 3 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distribution 1	$\tau = 0.1$	$\tau = 0.25$	$\tau = 0.5$	$\tau = 0.75$	$\tau = 0.9$
SampleQ					
$n = 25$	0.312	0.259	0.241	0.259	0.332
$n = 50$	0.223	0.183	0.161	0.201	0.229
$n = 100$	0.156	0.129	0.114	0.135	0.163
$n = 200$	0.115	0.094	0.091	0.107	0.131
BayesQ					
$n = 25$	0.204	0.185	0.194	0.196	0.230
$n = 50$	0.188	0.158	0.160	0.175	0.181
$n = 100$	0.140	0.117	0.107	0.115	0.140
$n = 200$	0.107	0.090	0.093	0.098	0.120

Table 3.4: Root Mean Square Errors: Normal Mixture 1 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distribution 2	$\tau = 0.1$	$\tau = 0.25$	$\tau = 0.5$	$\tau = 0.75$	$\tau = 0.9$
SampleQ					
$n = 25$	0.194	0.181	0.215	0.367	0.447
$n = 50$	0.139	0.112	0.144	0.269	0.388
$n = 100$	0.099	0.087	0.106	0.211	0.260
$n = 200$	0.068	0.060	0.074	0.132	0.181
BayesQ					
$n = 25$	0.265	0.140	0.186	0.233	0.278
$n = 50$	0.148	0.098	0.131	0.212	0.272
$n = 100$	0.092	0.079	0.099	0.176	0.208
$n = 200$	0.063	0.054	0.071	0.118	0.160

Table 3.5: Root Mean Square Errors: Normal Mixture 2 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distribution 3	$\tau = 0.1$	$\tau = 0.25$	$\tau = 0.5$	$\tau = 0.75$	$\tau = 0.9$
SampleQ					
$n = 25$	0.091	0.091	0.230	0.416	0.497
$n = 50$	0.069	0.065	0.115	0.311	0.371
$n = 100$	0.052	0.052	0.095	0.217	0.264
$n = 200$	0.034	0.031	0.060	0.148	0.168
BayesQ					
$n = 25$	0.324	0.092	0.215	0.308	0.310
$n = 50$	0.159	0.066	0.157	0.280	0.263
$n = 100$	0.047	0.050	0.123	0.212	0.207
$n = 200$	0.032	0.031	0.071	0.144	0.147

Table 3.6: Root Mean Square Errors: Normal Mixture 3 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distribution 1	$\tau = 0.1$	$\tau = 0.25$	$\tau = 0.5$	$\tau = 0.75$	$\tau = 0.9$
SampleQ					
$n = 25$	0.327	0.270	0.230	0.261	0.334
$n = 50$	0.250	0.200	0.182	0.202	0.243
$n = 100$	0.174	0.133	0.132	0.147	0.176
$n = 200$	0.113	0.093	0.080	0.096	0.124
BayesQ					
$n = 25$	4.419	4.164	3.852	1.720	1.031
$n = 50$	4.103	3.293	0.835	0.627	0.570
$n = 100$	2.618	0.526	0.351	0.311	0.320
$n = 200$	0.220	0.163	0.148	0.160	0.170

Table 3.7: Root Mean Square Errors: Normal Mixture 1 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distribution 2	$\tau = 0.1$	$\tau = 0.25$	$\tau = 0.5$	$\tau = 0.75$	$\tau = 0.9$
SampleQ					
$n = 25$	0.188	0.163	0.211	0.375	0.494
$n = 50$	0.144	0.125	0.157	0.271	0.404
$n = 100$	0.102	0.082	0.106	0.203	0.269
$n = 200$	0.069	0.062	0.073	0.128	0.182
BayesQ					
$n = 25$	4.298	3.797	3.156	1.385	1.389
$n = 50$	3.959	2.712	0.457	0.610	0.873
$n = 100$	1.676	0.225	0.203	0.363	0.534
$n = 200$	0.112	0.093	0.110	0.224	0.313

Table 3.8: Root Mean Square Errors: Normal Mixture 2 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distribution 3	$\tau = 0.1$	$\tau = 0.25$	$\tau = 0.5$	$\tau = 0.75$	$\tau = 0.9$
SampleQ					
$n = 25$	0.092	0.100	0.279	0.442	0.465
$n = 50$	0.072	0.063	0.136	0.323	0.364
$n = 100$	0.051	0.042	0.080	0.212	0.272
$n = 200$	0.038	0.032	0.054	0.143	0.178
BayesQ					
$n = 25$	3.815	3.160	2.327	1.213	1.388
$n = 50$	2.760	1.269	0.224	0.838	0.846
$n = 100$	0.615	0.089	0.112	0.547	0.454
$n = 200$	0.047	0.043	0.068	0.322	0.292

Table 3.9: Root Mean Square Errors: Normal Mixture 3 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distribution 1	$\tau = 0.1$	$\tau = 0.25$	$\tau = 0.5$	$\tau = 0.75$	$\tau = 0.9$
SampleQ					
$n = 25$	0.331	0.249	0.233	0.271	0.361
$n = 50$	0.208	0.181	0.162	0.199	0.252
$n = 100$	0.160	0.141	0.125	0.138	0.156
$n = 200$	0.127	0.108	0.083	0.089	0.115
BayesQ					
$n = 25$	0.225	0.186	0.184	0.193	0.231
$n = 50$	0.169	0.157	0.162	0.167	0.191
$n = 100$	0.143	0.126	0.125	0.125	0.141
$n = 200$	0.112	0.095	0.083	0.086	0.111

Table 3.10: Root Mean Square Errors: Normal Mixture 1 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distribution 2	$\tau = 0.1$	$\tau = 0.25$	$\tau = 0.5$	$\tau = 0.75$	$\tau = 0.9$
SampleQ					
$n = 25$	0.195	0.150	0.188	0.359	0.483
$n = 50$	0.126	0.118	0.153	0.302	0.364
$n = 100$	0.104	0.086	0.106	0.208	0.258
$n = 200$	0.068	0.055	0.069	0.134	0.174
BayesQ					
$n = 25$	0.254	0.119	0.151	0.215	0.284
$n = 50$	0.131	0.097	0.139	0.234	0.255
$n = 100$	0.094	0.080	0.099	0.181	0.206
$n = 200$	0.067	0.053	0.060	0.121	0.148

Table 3.11: Root Mean Square Errors: Normal Mixture 2 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distribution 3	$\tau = 0.1$	$\tau = 0.25$	$\tau = 0.5$	$\tau = 0.75$	$\tau = 0.9$
SampleQ					
$n = 25$	0.106	0.094	0.216	0.437	0.471
$n = 50$	0.068	0.066	0.161	0.281	0.318
$n = 100$	0.050	0.046	0.071	0.223	0.263
$n = 200$	0.038	0.034	0.055	0.150	0.173
BayesQ					
$n = 25$	0.401	0.098	0.206	0.335	0.296
$n = 50$	0.115	0.063	0.177	0.265	0.240
$n = 100$	0.076	0.047	0.108	0.216	0.198
$n = 200$	0.037	0.032	0.064	0.159	0.153

Table 3.12: Root Mean Square Errors: Normal Mixture 3 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

There are several remarkable features of these tables. Focusing attention first on Table 1 corresponding to the standard Gaussian setting we see that the Bayes estimates do remarkably well, consistently outperforming the sample quantiles. The gap in performance declines with sample size as we would expect from the Bernstein-von-Mises theorem, but is still apparent at $n = 200$. A more puzzling feature of the table is that the sample quantiles themselves seem to be doing quite well, a fact somewhat obscured by the even better performance of the Bayes estimates. Take for example estimation of the median with $n = 100$ in the first table. The RMSE is 0.114. The efficient estimator of the median in this case is the sample mean, which would have RMSE of $1/\sqrt{100} = 0.100$, and the sample median would be expected to have RMSE approximately equal to $\sqrt{\pi/200} \approx 0.125$ based on its asymptotic theory. In the process of investigating this anomaly, it was discovered that computing sample quantiles with `rq(x ~ 1, taus)` when the solution is non-unique, that is when $n\tau$ is an integer, tends to choose the order statistic closest to zero.² Thus, when the target density has median zero, this produces a desirable “shrinkage effect.” Whether this slight shrinkage is responsible to a significant degree for the unexpectedly good performance in Table 1 is unclear.

In the other tables we see that the Bayes estimates are also performing quite well, but in a few cases there seems to be a serious enough bias problem to dominate the variance reduction advantage that they offer. To explore these findings further, three new variants of the experiment were run with results presented in Tables 4-12. Tables 4-6 are simply a reproduction of the first set of experiments with exactly the same settings, but a new random number seed. Tables 7-9 centers the target distribution at $x = 5$, but leaves the prior unchanged. And Tables 10-12 return the centering of the target densities to zero, but increase the prior variance from 1 to 10. Shifting the location of the target densities has a disastrous effect on the Bayes estimator, underlining the need for a prior that doesn’t conflict dramatically with reality. Finally, increasing the dispersion of the prior attenuates the shrinkage advantage exhibited in Tables 1-6 for the Bayes estimators, but not as much as one might have expected. It is worth noting that the performance of the sample median for $n = 100$ in Table 10 is much closer to asymptotic expectations. Since the seeds in Tables 4 and 10 are identical it may seem troubling that the sample medians have different performance, but further reflection reveals that different priors for the Bayes estimators implies different MCMC sequences and therefore different data for the whole simulation. This suggests that the differences in the performance observed across the 4 sets of tables in the sample quantiles reflects the inherent variability of the simulation dictated by the relatively small number (200) of replications. Since each table requires almost 3 hours of cpu time for the MCMC one is reluctant to increase the number of replications in such circumstances. This is one of the most troubling aspects of the MCMC revolution: computing effort is often so substantial that thorough evaluation of the sensitivity of results to the plethora of tuning parameters is quite difficult.

A more stringent test of the success of the Bayesian approach involves examination of the credible intervals produced by the MCMC iterations. To examine this we computed both

² The simplex algorithm used by `rq` must choose one of the two central order statistics when computing the median from an even sample of distinct observations.

Distn 1	Credibility 0.9					Credibility 0.95				
	0.1	0.25	0.5	0.75	0.9	0.1	0.25	0.5	0.75	0.9
$n = 25$	0.355	0.020	0.635	0.030	0.255	0.990	1.000	1.000	1.000	1.000
$n = 50$	0.100	0.050	0.925	0.040	0.050	0.975	1.000	1.000	1.000	0.985
$n = 100$	0.000	0.035	1.000	0.025	0.000	0.980	1.000	1.000	1.000	0.995
$n = 200$	0.000	0.050	1.000	0.070	0.000	0.980	1.000	1.000	1.000	0.975

Table 3.13: Credible Interval Coverage: Normal Mixture 1 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distn 2	Credibility 0.9					Credibility 0.95				
	0.1	0.25	0.5	0.75	0.9	0.1	0.25	0.5	0.75	0.9
$n = 25$	0.495	0.025	0.645	0.020	0.130	0.985	1.000	1.000	1.000	0.970
$n = 50$	0.120	0.040	0.900	0.025	0.030	1.000	1.000	1.000	1.000	1.000
$n = 100$	0.005	0.050	1.000	0.025	0.000	0.985	1.000	1.000	1.000	0.990
$n = 200$	0.000	0.065	1.000	0.030	0.000	0.995	1.000	1.000	1.000	0.980

Table 3.14: Credible Interval Coverage: Normal Mixture 2 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distn 3	Credibility 0.9					Credibility 0.95				
	0.1	0.25	0.5	0.75	0.9	0.1	0.25	0.5	0.75	0.9
$n = 25$	0.690	0.020	0.655	0.010	0.075	0.995	1.000	1.000	1.000	0.985
$n = 50$	0.160	0.055	0.935	0.020	0.045	0.995	1.000	1.000	1.000	0.985
$n = 100$	0.005	0.035	1.000	0.075	0.000	0.980	1.000	1.000	1.000	0.995
$n = 200$	0.000	0.040	1.000	0.035	0.000	0.985	1.000	1.000	1.000	0.985

Table 3.15: Credible Interval Coverage: Normal Mixture 3 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distn 1	Credibility 0.9					Credibility 0.95				
	0.1	0.25	0.5	0.75	0.9	0.1	0.25	0.5	0.75	0.9
$n = 25$	0.290	0.015	0.600	0.015	0.285	0.990	1.000	1.000	1.000	0.990
$n = 50$	0.100	0.025	0.950	0.045	0.070	0.995	1.000	1.000	1.000	0.985
$n = 100$	0.000	0.040	1.000	0.030	0.000	0.995	1.000	1.000	1.000	0.995
$n = 200$	0.000	0.050	1.000	0.055	0.000	0.985	1.000	1.000	1.000	0.985

Table 3.16: Credible Interval Coverage: Normal Mixture 1 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

0.90 and 0.95 intervals for each of the quantiles for all of the experiments described above. Tables 13-24 report coverage frequencies for these experiments.

Distn 2	Credibility 0.9					Credibility 0.95				
	0.1	0.25	0.5	0.75	0.9	0.1	0.25	0.5	0.75	0.9
$n = 25$	0.520	0.035	0.580	0.030	0.125	0.975	1.000	1.000	1.000	0.985
$n = 50$	0.115	0.040	0.945	0.045	0.040	0.995	1.000	1.000	1.000	0.965
$n = 100$	0.000	0.035	0.995	0.025	0.005	0.985	1.000	1.000	1.000	0.975
$n = 200$	0.000	0.030	1.000	0.035	0.000	0.985	1.000	1.000	1.000	0.965

Table 3.17: Credible Interval Coverage: Normal Mixture 2 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distn 3	Credibility 0.9					Credibility 0.95				
	0.1	0.25	0.5	0.75	0.9	0.1	0.25	0.5	0.75	0.9
$n = 25$	0.775	0.055	0.595	0.015	0.090	1.000	1.000	1.000	0.995	0.975
$n = 50$	0.235	0.060	0.945	0.015	0.045	1.000	1.000	1.000	1.000	0.975
$n = 100$	0.005	0.055	1.000	0.055	0.000	0.995	1.000	1.000	1.000	0.985
$n = 200$	0.000	0.030	1.000	0.040	0.000	0.985	1.000	1.000	1.000	0.990

Table 3.18: Credible Interval Coverage: Normal Mixture 3 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distn 1	Credibility 0.9					Credibility 0.95				
	0.1	0.25	0.5	0.75	0.9	0.1	0.25	0.5	0.75	0.9
$n = 25$	0.145	0.000	0.000	0.825	0.970	0.935	0.955	0.070	0.650	0.980
$n = 50$	0.030	0.000	0.000	0.000	0.000	0.905	0.950	0.005	0.120	0.940
$n = 100$	0.000	0.000	0.000	0.000	0.000	0.490	0.580	0.000	0.000	0.000
$n = 200$	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table 3.19: Credible Interval Coverage: Normal Mixture 1 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distn 2	Credibility 0.9					Credibility 0.95				
	0.1	0.25	0.5	0.75	0.9	0.1	0.25	0.5	0.75	0.9
$n = 25$	0.355	0.000	0.000	0.650	0.785	0.805	0.840	0.015	0.295	0.925
$n = 50$	0.060	0.000	0.000	0.000	0.000	0.765	0.800	0.005	0.060	0.735
$n = 100$	0.000	0.000	0.000	0.000	0.000	0.150	0.170	0.000	0.000	0.000
$n = 200$	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table 3.20: Credible Interval Coverage: Normal Mixture 2 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distn 3	Credibility 0.9					Credibility 0.95				
	0.1	0.25	0.5	0.75	0.9	0.1	0.25	0.5	0.75	0.9
$n = 25$	0.270	0.000	0.000	0.380	0.460	0.575	0.585	0.010	0.245	0.735
$n = 50$	0.010	0.000	0.000	0.000	0.000	0.240	0.270	0.000	0.020	0.225
$n = 100$	0.000	0.000	0.000	0.000	0.000	0.010	0.005	0.000	0.000	0.000
$n = 200$	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table 3.21: Credible Interval Coverage: Normal Mixture 3 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distn 1	Credibility 0.9					Credibility 0.95				
	0.1	0.25	0.5	0.75	0.9	0.1	0.25	0.5	0.75	0.9
$n = 25$	0.295	0.030	0.575	0.010	0.305	0.990	1.000	1.000	1.000	0.975
$n = 50$	0.060	0.060	0.935	0.025	0.080	0.990	1.000	1.000	1.000	0.990
$n = 100$	0.005	0.020	1.000	0.055	0.005	0.995	1.000	1.000	1.000	0.985
$n = 200$	0.000	0.040	1.000	0.040	0.000	0.980	1.000	1.000	1.000	0.990

Table 3.22: Credible Interval Coverage: Normal Mixture 1 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distn 2	Credibility 0.9					Credibility 0.95				
	0.1	0.25	0.5	0.75	0.9	0.1	0.25	0.5	0.75	0.9
$n = 25$	0.505	0.030	0.710	0.010	0.095	0.985	1.000	1.000	1.000	0.980
$n = 50$	0.115	0.050	0.955	0.050	0.055	0.995	1.000	1.000	1.000	0.985
$n = 100$	0.005	0.050	1.000	0.040	0.005	0.990	1.000	1.000	1.000	0.980
$n = 200$	0.000	0.050	1.000	0.030	0.000	0.995	1.000	1.000	1.000	0.975

Table 3.23: Credible Interval Coverage: Normal Mixture 2 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

Distn 3	Credibility 0.9					Credibility 0.95				
	0.1	0.25	0.5	0.75	0.9	0.1	0.25	0.5	0.75	0.9
$n = 25$	0.715	0.050	0.620	0.010	0.080	0.995	1.000	1.000	1.000	0.985
$n = 50$	0.240	0.045	0.935	0.050	0.030	0.990	1.000	1.000	1.000	1.000
$n = 100$	0.015	0.070	1.000	0.030	0.000	0.990	1.000	1.000	1.000	0.965
$n = 200$	0.000	0.065	1.000	0.050	0.000	0.995	1.000	1.000	1.000	0.965

Table 3.24: Credible Interval Coverage: Normal Mixture 3 , 200 replications, 5000 Metropolis-Hastings steps with Jeffreys substitute Likelihood

The 0.95 intervals are quite good, but the 0.90 intervals are quite horrible. Recall that the third group of tables corresponds to the shifted location version of the model, and there both sets of intervals are awful. It would be interesting to know whether increasing the length of the MCMC chain would improve the performance of the 0.90 intervals.

Some Snarky Remarks on Bayesian Quantile Regression

To bake a Bayesian π (posterior) I was taught that you needed an $\mathcal{L}$ (likelihood) and a p (prior),¹ so it comes as something of a shock to discover that there are 15,200 web documents² employing the phrase “Bayesian quantile regression.” Quantile regression would seem to be the very antithesis of a likelihood based procedure, committing the investigator to a parametric model for one paltry conditional quantile function, while professing ignorance, even indifference, about the rest of the *Deus ex machina*.³

4.1 A Most Perplexing Paradox

So what is the attraction? What brings Bayesians to quantile regression like bears to honey? Is it that sweet smell of sin, always so powerful for the priesthood? Or is it that jihadist spirit of the Crusades, intent to recapture Jerusalem from the infidels? Maybe. But more likely it is simply that “Anything you can do, I can do better” confidence immortalized by Ethel Merman in the 1946 Irving Berlin musical *Annie Get Your Gun*.⁴ Curiously, this confidence, or should we say credulity, is justified by the familiar Tukeyesque principle that we should, when possible, “borrow strength” from adjacent data. “How much borrowing is prudent?” is always a valid question, however.

If you are interested in the $\tau = 1 - 4/365 \approx 0.99$ quantile of daily maximum ozone concentrations as mandated by the EPA, as in Reich, Fuentes, and Dunson (2009), it is hard to see how borrowing strength from conditional quantiles far below the median can be very prudent. Unless, of course, you are confident – that word again – that covariate “effects” are invariant across quantiles, and that you know quite a lot about the form of the underlying “error” distribution. An appealing compromise reminiscent of Hill’s (1975) early work on inference about Pareto tail exponents, lies in formulating models over relatively

¹ Oh yeah, and probably some data, don’t forget the data.

² As of April 26, 2010, according to Google.

³ Data generating mechanism.

⁴ If you listen to the song carefully, <http://popup.lala.com/popup/937030210442825432> you will hear that the only thing that both parties to this competitive duet agree upon is that neither one can bake a π .

narrow bands of quantiles. This approach is illustrated in the reanalysis by Tokdar and Kadane (2010) of recent data on the intensity of tropical cyclones. A challenge with this approach lies in structuring the likelihood and prior so that they do not assert a global influence that would violate the local intent of the model. This challenge is compounded in problems like the cyclone intensity application by the truncation of the observations below a fixed threshold. Borrowing can be good – when the loan can be repaid with interest, but sometimes it requires taking unjustifiable risks. In such cases the spoils to accrue to the circumspect.

4.2 “Mommy, Where do likelihoods come from?”

Marx famously asserted that he found the Hegelian dialectical method “standing on its head” and proceeded to turn it right side up. Does the Reverend’s philosophy of inverse probability need a similar inversion? For Bayesians the likelihood emerges from the mind fully formed, an exemplar of the synthetic *a priori*. Could we place it on a more materialistic footing?

Having once written the cabalistic equation,

$$Q_{Y_i}(\tau|x_i) = g(x_i, \beta(\tau)), \quad (4.1)$$

intending it to describe a single quantile τ , we may, intoxicated by its beauty, be tempted to believe that it holds for the entire world of $\tau \in (0, 1)$. And, having committed this sin of arrogance, we discover that we have also perpetrated a likelihood! The Bayesian response to this realization seems to be: “Zounds! now all we need is a prior for $\beta(\tau)$ and we’re in business.” The Tukey, exploratory data analysis, reaction is more likely to be: “How do we know g is reasonable, and *especially* how do we know that it is reasonable for all τ ? The answer to these agonizing questions leads us into the excruciating realm of diagnostic testing, as exemplified by Koenker and Xiao (2002). But for those less masochistically inclined, Bayesian introspection about appropriate priors might be a welcome alternative.

Even the most imaginative Bayesian must have days when they need some Virgilian guidance through the circles of such vast *a priori* landscapes. Probability measures on $\{\beta(\tau) \in \mathbb{R}^p | \tau \in (0, 1)\}$, constitute a big space, not quite Hell size, but larger than Texas. Those willing to swing from the empirical branch of the Bayesian family tree might consider the following prescription:

1. Hold nose,
2. Estimate the model (4.1) by frequentist quantile regression⁵ on a grid of τ ’s,
3. Compute the usual sandwich estimator of the asymptotic covariance matrix of these estimates on the grid,
4. At each design point, x_i , compute a (smoothed, monotonized if necessary) estimate the conditional quantile function, and the corresponding estimate of the asymptotic covariance function.
5. Release nose.

⁵ Also known as fuqr, or frequently ugly quantile regression, a Sufi term connoting “a state of contentment and renunciation”, see <http://chishti.org/sufismquran.htm>

At this point you have created, *mirabile dictu*, a Gaussian prior for $\beta(\tau)$. Even more remarkably, however, you have also created a likelihood! We must resist the temptation to call this an empirical likelihood, so we will call it a materialist likelihood. If you insist on a likelihood in the form of a density you will need to differentiate the resulting quantile function with respect to τ , and consider its reciprocal; the covariance function would also need to be modified accordingly via the delta method. But for most “modern” purposes simulation from the prior is what is really needed and this is immediately feasible from the quantile function. Of course there is also the small matter that you have just constructed a prior and a likelihood from the data, you do have some reserved data don’t you? If not, take it up with the Reverend next Sunday. A recent manifestation of the materialist likelihood can be found in Reich, Fuentes, and Dunson (2009) where the normal approximation for the fuqr $\hat{\beta}(\tau)$ is used in conjunction with normal priors for coefficients of a Bernstein polynomial expansion to model spatial ozone concentrations. For those wondering whether there are precedents for this sort of apparent heresy in the sacred texts of Bayesianism, one needs to look no further than Jeffreys (1939). Check the index for “median.”

The Method of Quantiles

The underlying idea of Reich, Fuentes, and Dunson (2009) to use the asymptotic normal approximation of the quantile regression estimator as a “substitute” likelihood can be regarded as a convenient dumbing-down of the Jeffrey’s idea elaborated by Lavine (1995) and Dunson and Taylor (2005). The obvious disadvantage of the original Jeffrey’s suggestion is that it is difficult to compute/update the binomial proposal, whereas the normal approximation is made for Bayes rule. One way to enter the dungeons and dragons of Reich, Fuentes, and Dunson (2009) is to peel away all the Bayesian prior layers and focus on the basic model in its simplest setting. This puts us in the realm of MoQ, or method of quantiles.¹

Essentially, MoQ is the much beloved MoM, with moments replaced by quantiles. Quantiles are moments too, of a sort. Suppose we have a parametric model: $Y_i \sim f(y, \theta)$, having quantiles $q(\tau, \theta)$ for $\tau \in (0, 1)$. A vector of sample quantiles $\hat{q}_n = (\hat{q}_n(\tau_i))$ based on a random sample of size n , is asymptotically Gaussian with mean $q(\theta) = (q(\tau_i, \theta))$ and covariance matrix

$$V = \frac{\tau_i \wedge \tau_j - \tau_i \tau_j}{f(q(\tau_i, \theta))f(q(\tau_j, \theta))}$$

Thus, a natural estimator of θ is

$$\hat{\theta} = \operatorname{argmin}_{\theta} \{(\hat{q} - q(\theta))^{\top} V^{-1}(\hat{q} - q(\theta))\}.$$

Usually, V needs to be estimated, but several approaches are already available.

Example 5.1. A particularly simple example of MoQ assumes that we can express,

$$q(\tau, \theta) = \sum_{j=1}^m \varphi_j(\tau) \theta_j.$$

For such linear basis expansion models we can immediately write the asymptotically optimal estimator of θ as,

¹ For some general background see Koenker (2005) Section 4.8. The acronym MoQ should not be confused with the metaphysics of quality, an offshoot of Robert Pirsig’s motorcycle zen zeitgeist.

$$\hat{\theta} = (X^\top V^{-1} X)^{-1} X^\top V^{-1} \hat{q}$$

where $X = (\varphi_j(\tau_i))$ is an p by m design matrix describing the basis expansion. For the sake of definiteness we can consider the Bernstein basis illustrated in Figure 1. Obviously, any

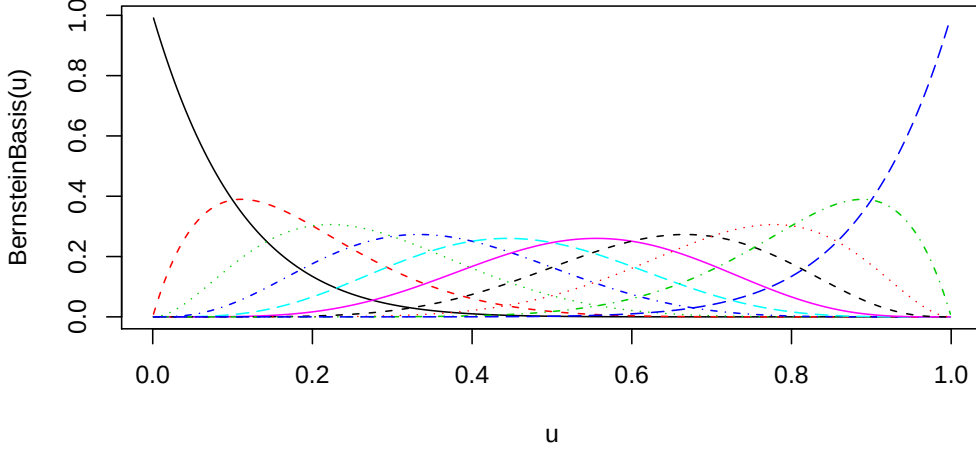


Fig. 5.1: Bernstein basis functions for $m = 10$.

choice of a basis determines a family of distributions especially well suited to that basis in the sense that this family is well approximated by a small number of the basis functions. In general there is likely to be some approximation error, or bias, due to truncation of the basis expansion. To illustrate this, consider the example taken from Reich, Fuentes, and Dunson (2009) with quantile function,

$$q(\tau) = 1 + (\tau + 1)\Phi^{-1}(\tau) + \frac{5}{4}\tau^2$$

To get a visual impression of the skewed form of this distribution we plot its quantile and density function in contrast to the standard Gaussian in Figure 2.

To investigate the adequacy of the Bernstein basis model for this test case we have conducted a small simulation experiment. In Figure 3 we illustrate a typical realization of the experiment. The solid (red) line represents the true quantile curve, the sample quantiles, $\{0.05, 0.10, \dots, 0.95\}$ appear as open circles with associated (pointwise) confidence intervals as vertical error bars. The dashed (blue) curve is the MoQ curve as described above, the dotted (green) curve is a constrained form of the MoQ curve estimated as in equation (6) of Reich, Fuentes, and Dunson (2009). The constraint imposed has two aspects: one requires that the first differences of the Bernstein coefficients be non-negative, this is a sufficient, but

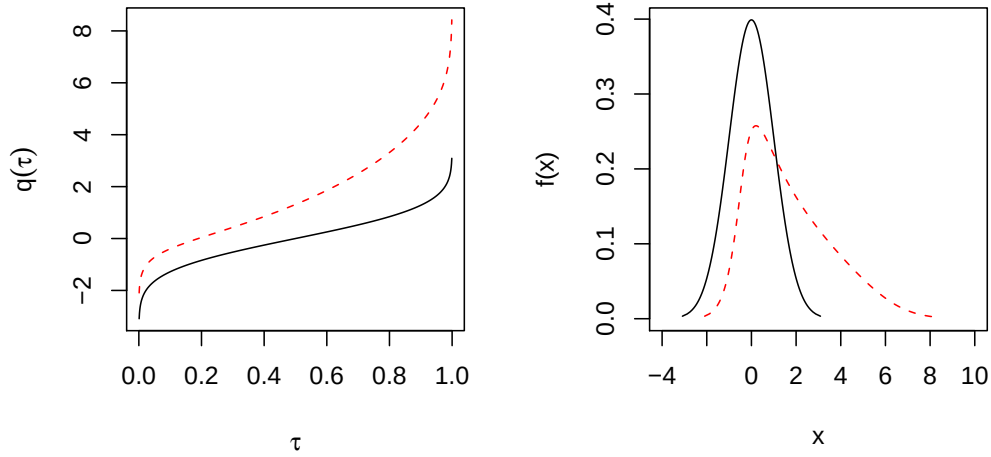


Fig. 5.2: Test quantile and density function contrasted with standard Gaussian.

not necessary condition for monotonicity. The second shrinks these first differences toward zero. The amount of shrinkage imposed by the latter constraint is controlled by a parameter λ that we have taken to be one in this figure. However, $\lambda = 1$ appears to be too stringent, as is evident in the figure, so for the simulation we have reduced this value and selected $\lambda = 0.1$, with this value the constrained and unconstrained MoQ estimates are typically quite similar.

In Figure 4 we illustrate root mean square error performance of the three estimators based on 10000 simulation replications. The lower curve represents the root mean squared error at the 19 selected quantiles of the ordinary sample quantiles. The upper curves representing RMSE performance of the MoQ and constrained MoQ estimators aren't too much worse, but this is about the best that can be said about them. Not surprisingly the main component of the differences in RMSE is the bias component that is depicted in Figure 5. Clearly, the constrained MoQ estimator has problems in both tails of the distribution. The simpler MoQ estimator does somewhat better, but still the restrictions imposed by the Bernstein basis appear to create some difficulties.

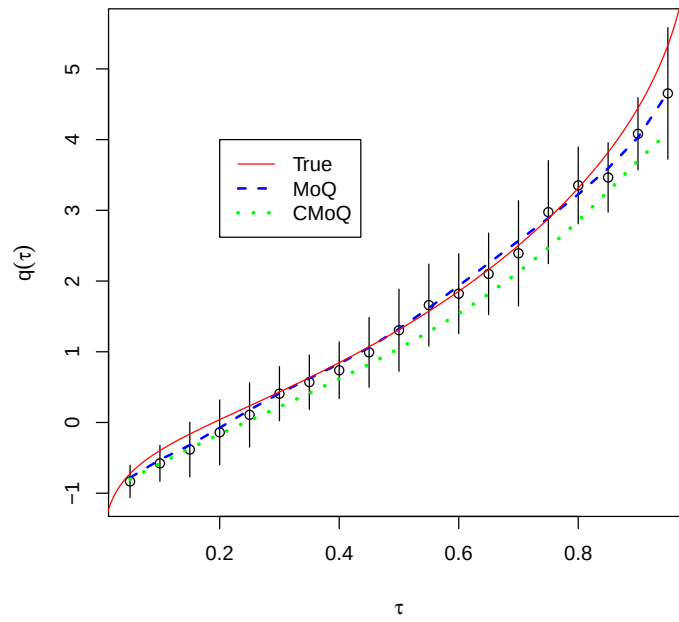


Fig. 5.3: Method of quantiles estimates of the test function.

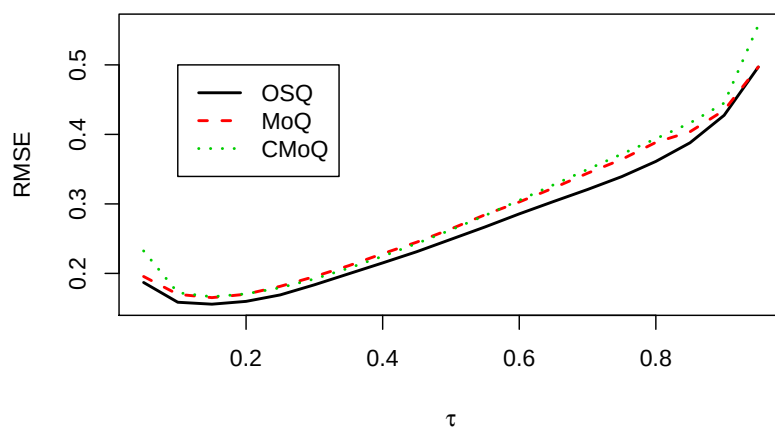


Fig. 5.4: Root mean squared error of three competing estimators of the sample quantiles based on 10000 replications.

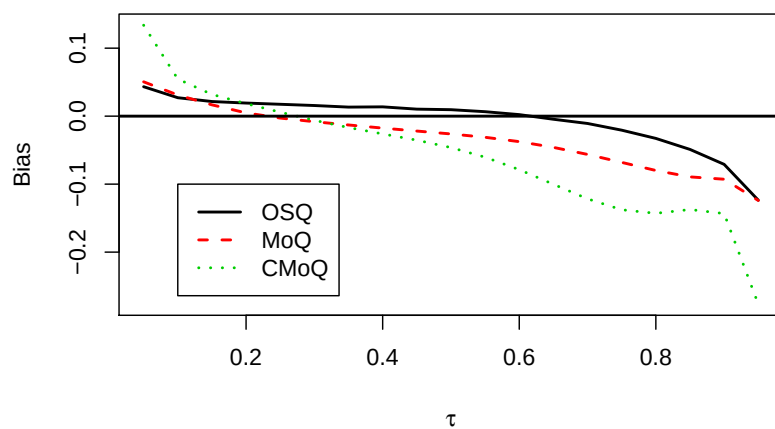


Fig. 5.5: Bias of three competing estimators of the sample quantiles based on 10000 replications.

The Targeted MLE: As I (Dimly) Understand It

6.1 Introduction

Targeted maximum likelihood estimation is a general class of procedures for estimating semiparametric statistical models, that is models with a nonparametric nuisance parameter.

¹ Judea Pearl, in his forward to the comprehensive exposition of van der Laan and Rose (2011), invokes Marschak (1953) Jacob Marshak's (1952) observations that good policy decisions rarely require a fully specified stochastic model. Instead, decisions can be based on only a few restrictive features of the model, e.g. average treatment effects, estimable without a full stochastic specification.

In these notes I would like to illustrate the application of this principle beginning with some simple missing data and binary treatment settings and concluding with a discussion of a recent paper of Díaz (2015) on quantile treatment effects. This expositional strategy is what might be called "inductive learning:" we start with very simple cases and try to leverage patterns we see into a general strategy. Those who prefer a more hard nosed approach should proceed directly to Appendix A of van der Laan and Rose (2011).

6.2 Prelude: Missing Data and Double Robustness

It seems appropriate to begin with a brief discussion of Bang and Robins (2005), which can be viewed as a precursor of the TMLE approach. Consider the standard missing data problem. We observe $Z = (Y, M, X)$. The response Y is observed when $M = 1$ and is missing otherwise. The vector of covariates, X , is available and satisfy:

- i. $P(M = 1|Y, X) = P(M = 1|X) \equiv \pi(X)$
- ii. $\pi(X) > 0 \quad a.e.P.$

¹ I would like to express my appreciation to Iván Díaz for providing R code for the simulations reported in Díaz and Rosenblum (2015) and Díaz (2015) and which I have adapted somewhat for the present notes, which were prepared for my Spring 2016 topics course on semiparametric methods in econometrics. Read at your own risk.

The former condition is usually referred to as "missing at random," but perhaps more descriptively as "no unmeasured confounders." The latter condition is usually referred to as the "common support" condition.

Given a random sample of the triples $\{Z_i\}$ $i = 1, \dots, n$ we may adopt a parametric model for π , say the logistic, and estimate the unconditional mean of Y using the Horvitz and Thompson (1952) estimator,

$$\hat{\mu} = n^{-1} \sum_{i=1}^n M_i Y_i / \hat{\pi}(X_i).$$

In the original survey sampling context, $\pi(X)$ denoted a priori sampling weights and didn't need to be estimated. In the missing data setting $\pi(X)$ is usually referred to as a propensity score and we simply reweight the observed Y_i 's by the inverse of their probability of inclusion to the sample.

Alternatively, we may represent μ as,

$$\mu = \mathbb{E}(\mathbb{E}(Y|M=1, X))$$

and this suggests estimating a model of the form,

$$\mathbb{E}(Y|M=1, X) = \Psi(s(X, \beta))$$

for some known link function, Ψ and then estimating,

$$\hat{\mu} = n^{-1} \sum_{i=1}^n \Psi(s(X_i, \hat{\beta})).$$

Thus, for example, if Ψ is the identity link and $s(X, \beta) = X^\top \beta$, we would simply estimate β on the observed sample and then estimate μ as the mean of the $\hat{Y}_i$'s for the full sample. If Y_i is itself binary we might again use the logistic link, i.e. $\log(u/(1-u))$ and $\hat{\beta}$ would solve the estimating equation,

$$\sum_{i=1}^n M_i \nabla_{\beta} s(X_i, \beta) (Y_i - \Psi(s(X_i, \beta))) = 0,$$

so again $\hat{\beta}$ would be the restricted MLE based on only the $M_i = 1$ observations. When $s(X, \beta) = X^\top \beta$ then $\nabla_{\beta} s(X_i, \beta) = X$.

Inquiring Minds Would Like to Know

1. What happened to the reweighting that is usually applied in the logistic estimating equation (IRLS) approach, see e.g. McCullagh and Nelder (1989) or my 508 notes.
2. Why logit? Why not Cauchit? See e.g. Koenker and Yoon (2009) or my 508 notes.

Rather than choose between these two options, it seems preferable to combine them in some way. How? Ideally, we would like an estimator that consistently estimates our

parameter of interest, μ , provided that *either* the propensity score model is correct, or the outcome regression model is correct. Bang and Robins (2005) refer to this property as "double robustness," I would have called it the belt and suspenders principle.

The implementation of this principle is quite straightforward, we simply augment the outcome regression by writing,

$$\mathbb{E}(Y|M=1, X) = \Psi(s(X, \beta)) + \phi\pi^{-1}(X, \alpha) \equiv \Phi(X, \beta, \phi)$$

Estimation is carried out in two stages: first, α is estimated by logistic regression, then (β, ϕ) are estimated with $\pi^{-1}(X, \hat{\alpha})$ acting as an additional covariate. The latter step solves the estimating equation,

$$0 = \sum_{i=1}^n M_i \nabla \Phi(Y_i - \Phi(X_i, \beta, \phi))$$

which can be rewritten as,

$$0 = U(\mu) = \sum_{i=1}^n \pi_i^{-1} M_i \nabla \Phi(Y_i - \mu) - (\pi_i^{-1} M_i - 1)(\Phi(X_i, \beta, \phi) - \mu)$$

or

$$0 = U(\mu) = \sum_{i=1}^n \pi_i^{-1} M_i \nabla \Phi(Y_i - \Phi_i) - (\Phi_i - \mu).$$

Note that the first term is zero since it is just the ϕ component of the original estimating equation, so,

$$\hat{\mu} = n^{-1} \sum_{i=1}^n \Phi(X_i, \hat{\beta}, \hat{\phi}),$$

can be used as a plugin estimator.

The foregoing missing data formulation can be extended to the standard average treatment effect problem. The data structure is almost the same, we have $Z = (Y, T, X)$, with the assumption that $Y \perp\!\!\!\perp T|X$. Now $T = 1$ indicates treated observations and $T = 0$, controls. We are interested in estimating the average treatment effect, which can be represented either as

$$\mu = \mathbb{E}(TY/\pi(X)) - \mathbb{E}((1-T)Y/(1-\pi(X))),$$

or

$$\mu = \mathbb{E}(\mathbb{E}(Y|T=1, X)) - \mathbb{E}(\mathbb{E}(Y|T=0, X)).$$

Thus, we can estimate μ either by

$$\hat{\mu} = \sum_{i=1}^n (T_i Y_i / \hat{\pi}_i) - ((1-T_i) Y_i / (1-\hat{\pi}_i)),$$

or

$$\hat{\mu} = \sum_{i=1}^n [\Psi(s(1, X_i, \hat{\beta})) - \Psi(s(0, X_i, \hat{\beta}))],$$

where $\hat{\pi}_i = \pi(X_i, \hat{\alpha})$ is the estimated propensity score for receiving treatment. To combine these approaches we define the augmented outcome model,

$$\Phi(T, X, \beta, \phi) = \Psi(s(T, X, \beta)) + \phi_1 T \pi^{-1} + \phi_0 (1 - T)(1 - \pi)^{-1}.$$

We then estimate (β, ϕ) as before with $\hat{\alpha}$ fixed and use,

$$\hat{\mu} = \sum_{i=1}^n [\Phi(s(1, X_i, \hat{\beta}, \hat{\phi})) - \Phi(s(0, X_i, \hat{\beta}, \hat{\phi}))].$$

6.3 TMLE: A Missing Manual

Thus far we have not mentioned efficiency. Double robustness was originally focused on achieving consistency under weaker conditions than those employed in earlier propensity score literature. For a more extensive treatment of this approach see Kang and Schafer (2007) and the discussion thereof. Rotnitzky and Vansteelandt (2014) and van der Laan and Rubin (2006) address the efficiency objective relying to some degree on the prior work of Robins, van der Vaart and others. Their strategy is to devise one-step and multi-step iterative procedures based on the efficient semiparametric score function. I will try to briefly describe how this is supposed to work for the standard model for missing data, and for the conventional binary treatment effect model. While the exposition in this section focuses entirely on mean effects, have faith we are (always) headed toward quantiles.

As in the previous section, we observe a random sample, $Z_1, \dots, Z_n$ with $Z = (Y, M, X)$ satisfying the conditions appearing there. The likelihood of a Z can be factored as,

$$\mathcal{L}(Z) = P_Y(Y|M, X)P_M(M|X)P_X(X).$$

We wish to make no modeling assumption about the marginal distribution of X , so we will assume throughout that P_X will be replaced by its MLE, the empirical distribution function. In contrast, following convention, we will adopt some convenient parametric binary response model for the propensity score, and denote

$$\pi(X) = P_M(M|X).$$

What distinguishes the TMLE approach is the treatment of P_Y . An initial estimate can be produced based on the $M = 1$ subsample; this may involve some fancy machine learning methods, but I won't delve into that. The crucial aspect of the initial estimate, however it is produced, is that it is inherently suspect. It needs to be modified in some way to account for the fact that it is based entirely on the observed Y data. We have already seen a couple of variants of this idea, but they seem rather ad hoc. It would be nice to have a more principled way to design a modification. Ideally, we would like our estimator, $\hat{\theta}$ of the population mean to be efficient in the sense that it satisfied the estimating equation representing the efficient influence function. Of course, this begs the question of how the EIF can be found, but in the following examples the EIFs will appear like rabbits out of the proverbial hat.

For estimating the population mean in the missing at random model the EIF is given by Bang and Robins (2005) as,

$$D(Z) = \frac{M}{\pi(X)}(Y - \mu(X)) + \mu(X) - \theta(Z).$$

Given this EIF we can consider various one parameter submodels, along the lines of van der Vaart (1998), two are suggested in his Example 25.16:

$$p_t(z) = c(t) \exp(tD(z))p_0(z),$$

and

$$p_t(z) = (1 + tD(z))p_0(z).$$

A third option is proposed by Díaz and Rosenblum (2015),

$$p_t(z) = c(t)(1 + \exp(-2tD(z)))^{-1}p_0(z),$$

The latter authors propose an iterative scheme for updating our provisional nonparametric model p_0 , at each step estimating the auxiliary parameter t , and stopping when the MLE of t occurs sufficiently close to zero. The first option would seem to be the most obvious exponential family choice, the others are intended to deal with the (familiar) difficulty of the unbounded IF. A useful, but rather trivial, exercise would be to verify that the MLE for t yields the EIF when t is evaluated at zero.

I'll illustrate the implementation of this approach in R using the first class of submodels. I presume that it will be obvious how to adapt this code to the other submodels. To get started we need to have data; here is a function to generate data according to the three schemes used in Díaz and Rosenblum (2015) for their simulation experiments. The response Y is binary, and observed only when $M = 1$

```
DGP <- function(n, pMf = 1){
  pMf <- switch(pMf,
    function(X) plogis(1 + 2*X),
    function(X) plogis(-1 + 2*X),
    function(X) plogis(-6 + 2*X + X^2)
  )
  pYf <- function(X) plogis(X - X^2)
  Z <- rnorm(n, sd = 1/sqrt(2))
  X <- Z + rnorm(n, sd = 1/sqrt(2))
  M <- rbinom(n, 1, pMf(X))
  Y <- rbinom(n, 1, pYf(X))
  Y[M==0] <- NA
  data.frame(Y, M, X)
}
```

Next we need code for the log likelihood function and its gradient.

```
logLik <- function(t, py, pm, px, Y, M){
  Dp <- function(py, pm, px)
    function(Y, M) (M/pm) * (Y - py) + py - sum(px * py)
```

```

D <- Dp(py, pm, px)
A <- cbind(py * pm, (1 - py) * pm, (1 - pm))
B <- cbind(D(1,1), D(0,1), D(2,0))
a <- (A * exp(t * B)) %*% rep(1,3)
- mean(t * D(Y, M)) + log(mean(px * a))
}
gradLik <- function(t, py, pm, px, Y, M){
  Dp <- function(py, pm, px)
    function(Y, M) (M/pm) * (Y - py) + py - sum(px * py)
  D <- Dp(py, pm, px)
  A <- cbind(py * pm, (1 - py) * pm, (1 - pm))
  B <- cbind(D(1,1), D(0,1), D(2,0))
  a <- (A * exp(t * B)) %*% rep(1,3)
  b <- (A * exp(t * B) * B) %*% rep(1,3)
  - mean(D(Y, M)) + mean(px * b)/mean(px * a)
}

```

Finally we have the main TMLE function. The only tricky bit is the way that the function `pz` gets updated. This is an example of the flexibility of R to "compute on the language," that is for functions to produce functions.

```

tmlex <- function(Z, tol = 1e-4, maxit = 10){
  # Exponential Family TMLE for (binary) missing data model
  # Thanks to Ivan Diaz for original implementation R code
  # Thanks to Mikko Korpela for R-help with the Dcall bit
  pZ <- function(pX, pM, pY)
    function(y,m) pX * pM^m * (1-pM)^(1-m) * pY^(m*y) *
      (1 - pY)^(m * (1 - y)) * (y == 2)^(1-m)
  Dp <- function(py, pm, px)
    function(y, m) (m/pm) * (y - py) + py - sum(px * py)
  Deps <- function(eps, py, pm, px)
    function(y, m) exp(eps * ((m/pm) * (y - py) + py - sum(px * py)))
  Y <- Z$Y; M <- Z$M; X <- Z$X
  Y[is.na(Y)] <- 2 # Aaargh.
  # Initialize
  fitM <- glm(M ~ X + I(X^2), family = binomial, data = Z)
  fitY <- glm(Y ~ X + I(X^2), family = binomial, subset = (M==1), data = Z)
  EMO <- predict(fitM, newdata = Z, type = "response") # g1X in DR
  EY0 <- predict(fitY, newdata = Z, type = "response") # Q1X in DR
  pz <- pZ(1/nrow(Z), EMO, EY0)
  it <- 0
  eps <- Inf
  while(eps > tol & it < maxit){
    it <- it + 1
    p11 <- pZ(1,1); p01 <- pZ(0,1); p20 <- pZ(2,0)

```

```

px <- p11 + p01 + p20 # P(X)
pm <- (p11 + p01)/px # P(M = 1 | X)
py <- p11/(p11 + p01) # P(Y = 1 | M = 1, X)
eps <- optim(0, fn = logLik, gr = gradLik, method = "BFGS",
  py = py, pm = pm, px = px, Y = Y, M = M)$par
Dcall <- do.call("Deps", list(eps = eps, py = py, pm = pm, px = px))
body(pz) <- call("*", body(pz), as.call(list(Dcall, quote(y), quote(m))))
}
p11 <- pz(1,1); p01 <- pz(0,1); p20 <- pz(2,0)
px <- p11 + p01 + p20 # P(X)
pm <- (p11 + p01)/px # P(M = 1 | X)
py <- p11/(p11 + p01) # P(Y = 1 | M = 1, X)
EIF <- Dp(py, pm, px)
theta <- sum(px * py)
list(theta = theta, var = var(EIF(Y,M))/nrow(Z), nit = it)
}

```

Now to illustrate usage we can do:

```

set.seed(1492)
Z <- DGP(1000)
a <- tmlex(Z)

```

Of course the standard MAR model is not that exciting, so it would be nice to demonstrate that there are better motivations for all of this within the domain of econometrics. Fortunately, such a motivation is close at hand, as we have already seen: estimating average treatment effects is essentially the same problem and we can proceed exactly as described at the end of Section 2. Rather than dwelling on this again, however, I'd like to proceed immediately to something even more compelling – estimation of quantiles with missing data and quantile treatment effects.

6.4 Quantiles for Missing Data and QTEs

In this section I'll rely heavily on recent work of Díaz (2015), but let's begin with a brief review of EIFs for quantile models. In the simplest case we have a random sample $\{Y_1, \dots, Y_n\}$ from a distribution F , and we will assume as usual that F has a density f with respect to Lebesgue measure, and that f is strictly positive on the support of F . Then the ordinary sample quantile,

$$\hat{Q}(\tau) = \inf\{y : F_n(y) \geq \tau\}$$

where F_n denotes the usual empirical distribution function $F_n(y) = n^{-1} \sum I(Y_i \leq y)$, satisfies the condition,

$$0 \approx \sum_{i=1}^n \psi_\tau(Y_i - \hat{Q}(\tau)),$$

with $\psi_\tau(u) = I(u < 0) - \tau$. The fishy $\approx$ symbol reflects the possible ambiguity when $n\tau$ is an integer and there are multiple solutions, but this isn't very consequential since it is a $1/n$ phenomenon in a $1/\sqrt{n}$ world. A little more effort, see e.g. my 574 Lecture 15, reveals that the efficient influence function for estimating $Q(\tau)$ is given by,

$$\text{EIF}_{\hat{Q}(\tau)}(y) = \frac{1}{f(Q(\tau))} \sum_{i=1}^n (I(y < \hat{Q}(\tau)) - \tau).$$

This isn't terribly interesting since the density factor doesn't really play an active role. Things get more interesting in regression settings. We can consider the estimation of the linear quantile regression model,

$$Q_{Y|X}(\tau|x) = x^\top \beta(\tau)$$

as semiparametric with finite dimensional parameter, $\beta(\tau)$ and nuisance parameter, $F_{Y|X}$. Now the EIF for β for fixed τ is

$$\text{EIF}_{\hat{\beta}(\tau)}(y, x) = \sum_{i=1}^n \frac{1}{f_i(x_i^\top \beta(\tau))} (I(y < \hat{x}_i^\top \beta(\tau)) - \tau).$$

and each observation is weighted inside the EIF sum according to the local, conditional density terms f_i . This version of the EIF corresponds to the optimally weighted quantile regression problem,

$$\hat{\beta}(\tau) = \text{argmin}_b \sum_{i=1}^n f_i(x_i^\top \beta(\tau)) \rho_\tau(y_i - x_i^\top b).$$

The unweighted quantile regression estimator achieves an asymptotic covariance matrix of the sandwich form, $V = H^{-1} J H^{-1}$ where $H = \lim X^\top \Phi X / n$, $\Phi = \text{diag}(f_i)$ and $J = \lim \tau(1-\tau) X^\top X / n$. In the weighted form, the sandwich collapses to $V = \lim \tau(1-\tau) X^\top X / n$. This conforms to applying the f_i weights in the efficient estimating equation and thereby canceling them out in the EIF. A variety of proposals have been made to construct one-step estimators that estimate weights based on a preliminary estimator of $\beta(\tau)$ and reestimate. A possibly interesting alternative to this approach would be to apply the TMLE strategy and iterate toward the solution of the EIF estimating equation. We will leave this to future consideration and now return to the missing data setting, albeit with the objective of estimating quantiles rather than means. Before doing so, I would like to stress that a virtue of the quantile regression paradigm is that even this efficient score formulation depends only on local information about $F_{Y|X}$ near the τ th conditional quantile.

The missing data setting for quantile estimation is quite similar to that of the previous sections except that the response, Y , when it is observed is no longer binary. Quantiles for binary data are somewhat silly, although conditional quantiles for binary data have their own charms. We maintain our prior conditions

- i. $P(M = 1|Y, X) = P(M = 1|X) \equiv \pi(X)$
- ii. $\pi(X) > 0$ a.e.P.

iii. $Y \perp\!\!\!\perp M|X$

Again, we can factor $P_Z = P_Y(y|M, X)P_M(M|X)P_X$ and as before we will denote the propensity score, $\pi(x) = P_M(M = 1|X = x)$. The unconditional distribution of Y will be denoted by F and we will assume that it has strictly positive density in a neighborhood of $\theta = F^{-1}(\tau)$, which will be our target parameter. We will begin by focusing on the pure missing data model, and then turn our attention to the QTE problem. Throughout we will consider the unconditional quantile θ with the understanding that extensions to conditional quantile estimation would be an attractive subject for future research.

The EIF for estimators of θ at P is given by,

$$D(P, Z) = \frac{1}{f(\theta)} \left[\frac{M}{\pi(X)} (I(Y < \theta) - P_Y(\theta|1, X)) + P_Y(\theta|1, X) - \tau \right].$$

This result, Lemma 1 of Díaz (2015), combines what we have already seen regarding the MAR model for means with the EIF result for unconditional quantiles.

Unlike the binary response setting, or for that matter estimating averages for continuous response, for quantiles we require an initial estimator of the entire conditional distribution, $P_Y(y|M = 1, X)$. In principle this could be accomplished by some form of nonparametric density estimation, but when the dimension of X is large, or even moderate, this is not very practical; so we resort to – what else? – quantile regression. On a grid of $\tau \in (0, 1)$, we compute,

$$\hat{Q}_{Y|M=1, X}(\tau|x) = x^\top \hat{\beta}(\tau),$$

and to ensure monotonicity at all x_i of $\hat{Q}$ we “rearrange” $\hat{Q}(\tau|x)$ as in Chernozhukov, Fernandez-Val, and Galichon (2009). The gory details of this process are illustrated in the R code that is provided below for the QTET problem. Our initial estimate of the conditional distribution of Y is easily obtained by inversion, and the marginal by integrating out over the empirical distribution of X . Note that the latter step integrates over the entire sample, not just the subsample with observed response, that is,

$$\hat{F}_n(y) = n^{-1} \sum_{i=1}^n \hat{Q}^{-1}(y|M = 1, X_i).$$

Our initial estimator of θ becomes $\hat{\theta} = \hat{F}_n^{-1}(\tau)$.

To update $P_Y(y|M = 1, X)$ Díaz (2015) suggests the same exponential family sub-model that we have described above. Let $p_Y(y|M = 1, x)$ be the density corresponding to $P_Y(y|M = 1, x)$, and denote,

$$p_t(y|M = 1, x) = c(t, p_Y) \exp(tD(p_Y, z))p_Y(y|M = 1, x)$$

where $c(t, p_Y)$ is the normalizing constant, and

$$D(p_Y, z) = \frac{1}{\pi(x)} [I(y \leq \theta) - P_Y(y|M = 1, x)].$$

At each iteration we can solve for the MLE of t , and then update to get p_t , continuing until $\hat{t} \approx 0$, at which point we have a fixed point solution for the least favorable model.

Rather than dwell further on the missing data case I'd rather proceed, as does Díaz (2015) to consider the closely related case of estimating QTEs.

In the familiar Neyman-Rubin-Pearl causal model we have a binary treatment, T and observe a response Y_1 if $T = 1$, and a response Y_0 if $T = 0$, but never both. We would like to estimate the quantile treatment effect on the treated, by which we will mean the difference between the τ th quantile of the response distribution of the treated subjects minus the τ th quantile of what would have been the response distribution for the treated subjects if they had not received the treatment. The former quantity is obviously estimable as,

$$\hat{\theta}_1 = \operatorname{argmin} \sum_{i:T_i=1} \rho_\tau(Y_i - \theta),$$

so we will concentrate on the more challenging task of estimating θ_0 , the counter-factual τ th quantile for the treatment subjects had they been controls. As usual we need to make identifying assumptions. On the face of it, the task seems formidable since we don't observe *any* Y_0 's for the treatment subjects. But covariates come to the rescue. The covariates are the pixie dust, and

- i. $P(T = 1|Y, X) = P(T = 1|X) \equiv \pi(X)$
- ii. $\pi(X) > 0 \quad a.e.P.$

are the "happy thoughts" that keep you airborne in the night sky of causal inference.

The distribution of the control responses for the treatment group can then be identified from the expression,

$$F^0(y) = \sum_x P_Y(y|T = 0, x) p_X(x|T = 1).$$

Clearly, $P_Y(y|T = 0, x)$ is estimable from the control observations in exactly the same way we were able to estimate $P_Y(y|T = 1, x)$ from the treatment group. So to obtain an estimate of F^0 we simply reweight according the mass associated with the covariates in the treatment group. For example, if there are a lot of $x = a$ types in the treatment group relative to their abundance in the control group, then those guys will get more weight in this sum. This brings us back to something similar to the missing data problem we have already considered.

The EIF for estimating θ_0 is given by,

$$D_0(p_Y, z) = \frac{1}{f(\theta_0)} \left[\frac{1-T}{\mathbb{E}T} \frac{\pi(x)}{1-\pi(x)} (I(y \leq \theta_0) - P_Y(\theta_0|M=0, x)) + \frac{T}{\mathbb{E}T} (P_Y(\theta_0|M=0, x) - \tau) \right].$$

This looks rather complicated, but it also closely resembles the pure missing data problem we have already considered, except that we are now focused entirely on the missing control observations for the treatment group.

For those, like me, who believe in mathematics only insofar as it can be translated into R code to compute something, and be experimented with, I offer my slightly modified version of the Díaz (2015) TMLE code for the QTET estimator. It will be noted that the function QTET actually returns several competing estimators of the QTET. In addition to the TMLE there is a naive estimator based on only the initial estimators of the quantiles, and two one-step methods, one attributable to Firpo (2007) and the other discussed briefly by Díaz (2015). I will come back to these after a brief discussion of the TMLE implementation.

```

QTET <- function(Z, tau = 0.5, taus = 1:99/100, maxit = 10) {
  # Computes QTET estimators as in Diaz (2015)
  #
  # Args:
  # Z ~ data.frame consisting of:
  # y ~ a response vector
  # t ~ a treatment indicator
  # x ~ some other conditioning covariates
  # tau ~ quantile of interest
  # taus ~ grid of quantiles for CQF estimation
  # maxit ~ upper bound on iteration count
  #
  # Returns:
  # tmle ~ the TMLE estimator of Diaz (2015)
  # se ~ standard error of the tmle
  # firpo ~ Firpo (2007) propensity score estimator
  # onestep ~ one-step estimator based on tmle EIF
  # Some required functions:
  F <- function(x, Q, w) mean(rowSums((Q <= x) * w))
  Finv <- function(u, Q, w, range)
    uniroot(function(x) F(x,Q,w) - u, range, extendInt = "yes")$root
  f <- function(y, Q, w, u = 1:99/100){
    z <- sapply(u, function(u) Finv(u, Q, w, range(Q)))
    approx(density(z), xout = y)$y
  }
  D <- function(x, y, Q, w, g)
    (g/(1-g)) * ((y <= x) - rowSums((Q <= x) * w))
  logLik <- function(eps, t, w, Do, Dq)
    -sum((1-t) * (eps * Do - log(rowSums(exp(eps * Dq) * w))))
  gradLik <- function(eps, t, w, Do, Dq)
    -sum((1-t) * (Do - rowSums(Dq * exp(eps * Dq) * w)) /
      rowSums(exp(eps * Dq) * w))
  # Compute CQF of the control observations
  rqf <- rq(y ~ . - t, tau = taus, data = Z, subset = (t == 0))
  cqf <- predict(rqf, type = "Qhat", newdata = Z, stepfun = TRUE)
  cqf <- lapply(cqf, rearrange)
  Q <- t(sapply(cqf, function(f)f(unique(knots(f)))))
  y <- Z$y
  t <- Z$t
  # Compute propensity score
  fitT <- glm(t ~ . - y, data = Z, family = binomial)
  g <- predict(fitT, type = "response")
  # Initialization:
  n <- nrow(Q)

```

```

m <- ncol(Q)
w <- matrix(1/m, n, m)
it <- 1
crit <- TRUE
q0 <- Finv(tau, Q[t == 1,], w[t == 1,], range(Q))
q1 <- quantile(y[t == 1], tau)
f0 <- f(q0, Q[t == 1,], w[t == 1])
f1 <- approx(density(y[t == 1]), xout = q1)$y
eif0 <- ((1-t) * D(q0, y, Q, w, g) +
  t * (rowSums((Q <= q0) * w) - tau))/(f0 * mean(t)) # (7)
eif1 <- t * ((y <= q1) - tau)/(f1 * mean(t))
eif <- eif1 - eif0
naive <- q1 - q0
onestep <- q1 - q0 - mean(eif)
while(crit && it <= maxit){
  Do <- D(q0, y, Q, w, g)
  Dq <- apply(Q, 2, function(y) D(q0, y, Q, w, g))
  eps <- optim(par = 0, logLik, gradLik, method = "BFGS",
    t = t, w = w, Do = Do, Dq = Dq)$par
  w <- exp(eps * Dq) * w/rowSums(exp(eps * Dq) * w)
  q0 <- Finv(tau, Q[t == 1,], w[t == 1,], range(y))
  it <- it + 1
  crit <- (abs(eps) > 1e-3/n^(3/5))
}
f0 <- f(q0, Q[t == 1,], w[t == 1])
eif0 <- ((1-t) * D(q0, y, Q, w, g) +
  t * (rowSums((Q <= q0) * w) - tau))/(f0 * mean(t)) # (7)
eif <- eif1 - eif0
firpo <- q1 - coef(rq(y ~ 1, tau = tau, data = Z, weights = (1-t) * g/(1-g)))
list(tmle = q1 - q0, naive = naive, firpo = firpo, onestep = onestep,
  se = sqrt(var(eif)/n), nit = it)
}

```

In some respects the code for the QTET is simpler than the `tmlex` code we saw earlier. There is no fancy updating of the submodel function with `do.call`, etc., everything is dealt with explicitly in the way that the array `Dq` is updated. But essentially we have the same algorithmic structure as in earlier TMLE procedures:

```

Initialize  $p_Y$ 
While( $\epsilon$  too big)
  Construct new parametric submodel
  Find MLE for  $\epsilon$  in new model
  Update  $p_Y$ 
Return

```

I wouldn't claim that the foregoing description will enable you to leap out of upper story windows with confidence, and even so you should always remember that Captain Hook and his pirates are lurking down below. But perhaps with some safety netting and further experimentation it will help to guide further exploration of the TMLE paradigm. I believe that the QTE extension of Díaz (2015) is a particularly promising development.

6.5 QTET Quintet + 1

To explore performance of the TMLE estimator of Díaz's "quantile treatment effect on the treated" (QTET) consider a simple simulation experiment. A binary treatment, T , is administered, there is a single, scalar, normally distributed covariate, X , and for each treated subject we observe a continuous response, Y_1 , while for control subjects we see response, Y_0 , where

$$Y_1 = \beta_0(U_1) + \beta_1(U_1)X + \gamma_0(U_1)T + \gamma_1(U_1)T \cdot X$$

and

$$Y_0 = \beta_0(U_0) + \beta_1(U_0)X$$

and (U_0, U_1) are marginally uniform random variables drawn from a one parameter (Frank) copula with dependence parameter, θ . Varying θ we can transit from independence, $\theta = 0$, to comonotonicity, $\theta \rightarrow \infty$. The conditional QTE is clearly

$$\text{QTE}(\tau|X) = \gamma_0(\tau) + \gamma_1(\tau)X$$

and integrating out X over the treated subjects we have,

$$\text{QTET}(\tau) = \gamma_0(\tau) + \gamma_1(\tau)\bar{X}_T$$

where $\bar{X}_T = \frac{1}{|\mathcal{T}|} \sum_{i \in \mathcal{T}} X_i$, and $\mathcal{T} = \{i : T_i = 1\}$.

In order to have any hope of estimating these quantities we must be willing to vouch for the fact that $T \perp\!\!\!\perp (Y_0, Y_1)|X$. In our experiment we consider two logistic specifications:

$$T = I(\alpha_0 + \alpha_1 \exp(X) \geq V),$$

and

$$T = I(\alpha_0 + \alpha_1 X \geq V),$$

with V logistically distributed. The conditional independence condition is satisfied in both cases, since T is assigned solely on the basis of the observable covariate, so no "compliance" issues are allowed to raise their ugly heads. However, the former case as implemented in `msim1.R` implies that the propensity score model used for the TMLE is misspecified.

When the pair (U_0, U_1) is not comonotonic the (dare I say?) classical² QR formulation of the treatment effect model

$$Q_Y(\tau|T, X) = \beta_0(\tau) + \beta_1(\tau)X + \gamma_0(\tau)T + \gamma_1(\tau)T \cdot X$$

is no longer valid, i.e., is no longer a proper quantile function, and we would like to understand better the consequences of this misspecification. Presumably, this could be investigated in a very partial way by varying the copula dependence parameter.

² Here I employ "classical" in the familiar sense of "almost always wrong."

6.5.1 The Proof of the Pudding

Before launching a full comparison of our alternative estimators we first illustrate a graphical comparison of only the first two candidates. In Figure 1 we illustrate the true marginal QTET for our simulation DGP, along with two estimates: one based on the classical (aka ordinary) QR fitting of the model with treatment main and interaction effects, and the other based on fitting separate models for the treated and control subjects and then evaluating their difference at the mean covariate setting. Each *seems* to perform well, but as Hamlet says, "Seems," madam? Nay, it is; I know not "seems." Maybe, we should reserve judgement until there is more evidence.

```
# A Simple Binary Treatment model
require(copula)

## Loading required package: copula
require(quantreg)
DGP <- function(n, copula = frankCopula(1, dim = 2), IA = TRUE) {
  uv <- rCopula(n, copula)
  x <- exp(rnorm(n))
  X <- cbind(1,x)
  T <- (rlogis(n) < (X %*% c(0.5, -1)))
  y0 <- qnorm(uv[,1]) + x * (1 + qnorm(uv[,1])/3)
  if(IA)
    y1 <- qnorm(uv[,2]) + x * (1 + qnorm(uv[,2])/3) +
      T * (1 + log(x)/2) * qchisq(uv[,2],5)
  else
    y1 <- qnorm(uv[,2]) + x * (1 + qnorm(uv[,2])/3) +
      T * qchisq(uv[,2],5)
  y <- T * y1 + (1 - T) * y0
  data.frame(y = y, y0 = y0, y1 = y1, Treat = T, x = x)
}
tau <- 1:9/10
D <- DGP(1000)
f <- rq(y ~ log(x) * Treat, tau = 1:9/10, data = D)
f0 <- rq(y0 ~ log(x), tau = tau, data = D, subset = (Treat == 0))
f1 <- rq(y1 ~ log(x), tau = tau, data = D, subset = (Treat == 1))
qte1 <- coef(f)[3,] + coef(f)[4,] * mean(log(D$x[D$Treat == 1]))
plot(tau, qte1, type = "b", xlab = expression(tau),
     ylab = expression(QTET(tau)))
qte2 <- t(coef(f1) - coef(f0)) %*% c(1, mean(log(D$x[D$Treat == 1])))
lines(tau, qte2, col = 2)
points(tau, qte2, col = 2)
lines(tau, qchisq(tau,5) * (1 + mean(log(D$x[D$Treat == 1])/2)), col = 3)
legend("topleft", c("OQR", "DQR", "Truth"), col = 1:3, lty = 1)
```

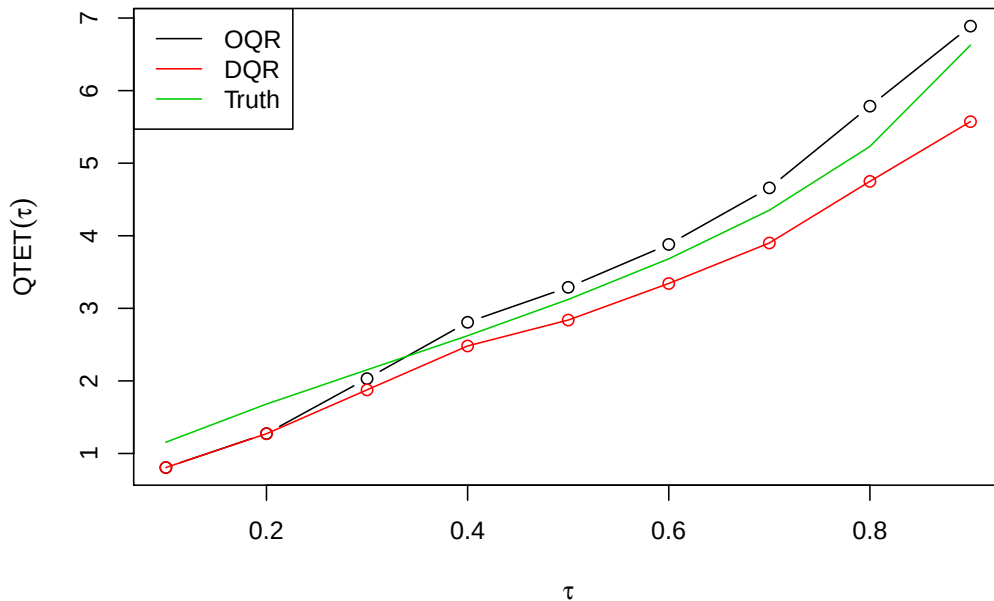


Fig. 6.1: Marginal QTET's for Simulation Model 1

Table 6.1: Performance Comparison of Marginal QTETs for Simulation Model 1

	QR1	QR2	TMLE	Naive	Firpo	OneStep
Bias	0.483	0.0433	-0.547	-0.324	-0.366	-0.686
RMSE	0.597	0.2787	0.618	0.420	0.470	0.754

Rather than keeping track of all the deciles as in Figure 1, we focus our next two experiments on the marginal median QTET. We consider four estimators in addition to the two described in Figure 1: the TMLE a “Naive” estimator, Firpo’s onestep, and a simple onestep all described by (Díaz 2015) and implemented by the `QTET` function. In Table 1 report mean bias and root mean squared error for our six estimators for the misspecified case that the treatment is assigned according to the lognormal model, while the last four estimators all assume that the propensity score model should be linear $\log(X)$. In Table 2 this misspecification is corrected; treatment is assigned as assumed in the propensity score. This should improve performance of all the last four methods, but doesn’t seem to do so as much as might be anticipated.

Table 6.2: Performance Comparison of Marginal QTETs for Simulation Model 2

	QR1	QR2	TMLE	Naive	Firpo	OneStep
Bias	0.572	0.310	-0.468	-0.170	-0.468	-0.657
RMSE	0.644	0.408	0.536	0.296	0.538	0.725

6.5.2 Simulation Protocol

The simulations underlying the results in Tables 1 and 2 were carried out in accordance with the protocol described in (Koenker 2014) using the shell script `Rbatch`. Results for all 1000 replications of the experiment are saved, and tables and figures can then be easily prepared by loading these results. The code for these experiments and for making the tables is reproduced below.

```
# A Simple Treatment model
require(copula)
require(quantreg)
source("QTET.R")
DGP <- function(n, copula = frankCopula(1, dim = 2), IA = TRUE) {
  uv <- rCopula(n, copula)
  x <- exp(rnorm(n))
  X <- cbind(1, x)
  T <- (rlogis(n) < (X %*% c(0.5, -1)))
  y0 <- qnorm(uv[,1]) + x * (1 + qnorm(uv[,1])/3)
  if(IA)
    y1 <- qnorm(uv[,2]) + x * (1 + qnorm(uv[,2])/3) +
      T * (1 + log(x)/2) * qchisq(uv[,2], 5)
  else
    y1 <- qnorm(uv[,2]) + x * (1 + qnorm(uv[,2])/3) +
      T * qchisq(uv[,2], 5)
  y <- T * y1 + (1 - T) * y0
  data.frame(y = y, y0 = y0, y1 = y1, Treat = T, x = x)
}
tau <- 1:9/10
R <- 1000
B <- matrix(0, 7, R)
for(i in 1:R){
  D <- DGP(1000)
  f <- rq(y ~ log(x) * Treat, tau = 1:9/10, data = D)
  f0 <- rq(y0 ~ log(x), tau = tau, data = D, subset = (Treat == 0))
  f1 <- rq(y1 ~ log(x), tau = tau, data = D, subset = (Treat == 1))
  qte1 <- coef(f)[3,] + coef(f)[4] * mean(log(D$x[D$Treat == 1]))
  qte2 <- t(coef(f1) - coef(f0)) %*% c(1, mean(log(D$x[D$Treat == 1])))
  truth <- qchisq(tau[5], 5) * (1 + mean(log(D$x[D$Treat == 1])/2))
}
```

```

Z <- data.frame(y = D$y, t = D$Treat, x = log(D$x))
A <- QTET(Z)
B[,i] <- c(truth, qte1[5], qte2[5], A$tmle, A$naive, A$firpo, A$onestep)
}

```

```

# Another Simple Treatment model
require(copula)
require(quantreg)
source("QTET.R")
DGP <- function(n, copula = frankCopula(1, dim = 2), IA = TRUE) {
  uv <- rCopula(n, copula)
  x <- exp(rnorm(n))
  X <- cbind(1, log(x))
  T <- (rlogis(n) < (X %*% c(0.5, -1)))
  y0 <- qnorm(uv[,1]) + x * (1 + qnorm(uv[,1])/3)
  if(IA)
    y1 <- qnorm(uv[,2]) + x * (1 + qnorm(uv[,2])/3) +
      T * (1 + log(x)/2) * qchisq(uv[,2], 5)
  else
    y1 <- qnorm(uv[,2]) + x * (1 + qnorm(uv[,2])/3) +
      T * qchisq(uv[,2], 5)
  y <- T * y1 + (1 - T) * y0
  data.frame(y = y, y0 = y0, y1 = y1, Treat = T, x = x)
}
tau <- 1:9/10
R <- 1000
B <- matrix(0, 7, R)
for(i in 1:R){
  D <- DGP(1000)
  f <- rq(y ~ log(x) * Treat, tau = 1:9/10, data = D)
  f0 <- rq(y0 ~ log(x), tau = tau, data = D, subset = (Treat == 0))
  f1 <- rq(y1 ~ log(x), tau = tau, data = D, subset = (Treat == 1))
  qte1 <- coef(f)[3,] + coef(f)[4,] * mean(log(D$x[D$Treat == 1]))
  qte2 <- t(coef(f1) - coef(f0)) %*% c(1, mean(log(D$x[D$Treat == 1])))
  truth <- qchisq(tau[5], 5) * (1 + mean(log(D$x[D$Treat == 1])/2))
  Z <- data.frame(y = D$y, t = D$Treat, x = log(D$x))
  A <- QTET(Z)
  B[,i] <- c(truth, qte1[5], qte2[5], A$tmle, A$naive, A$firpo, A$onestep)
}

```

```

load("msim1.Rda")
G <- B[-1,] - B[1,]
H <- rbind(apply(G, 1, mean), sqrt(apply(G^2, 1, mean)))

```

```

dimnames(H) <- list(c("Bias", "RMSE"),
                   c("QR1", "QR2", "TMLE", "Naive", "Firpo", "OneStep"))
cap1 = "Performance Comparison of Marginal QTETs for Simulation Model 1"
latex(H, file = "", rowlabel = "", digits = 3, caption = cap1)

```

```

load("msim2.Rda")
G <- B[-1,] - B[1,]
H <- rbind(apply(G,1,mean), sqrt(apply(G^2, 1, mean)))
dimnames(H) <- list(c("Bias", "RMSE"),
                   c("QR1", "QR2", "TMLE", "Naive", "Firpo", "OneStep"))
cap2 = "Performance Comparison of Marginal QTETs for Simulation Model 2"
latex(H, file = "", rowlabel = "", digits = 3, caption = cap2)

```

A Gaussian Compound Decision Bakeoff

Martin and Walker (2013) have recently proposed a parametric empirical Bayes procedure for the classical Gaussian compound decision problem in which, $Y_i \sim \mathcal{N}(\theta_i, 1)$, $i = 1, 2, \dots, n$ and we wish to estimate $\theta \in \mathbb{R}^n$ subject to squared error loss. I was curious to compare this procedure with the nonparametric empirical Bayes procedure recently introduced by Jiang and Zhang (2009) and further explored in Koenker and Mizera (2013). The latter approach is based on the nonparametric maximum likelihood estimator for mixture models of Kiefer and Wolfowitz (1956). The interior point computational approach suggested in Koenker and Mizera (2013) for the Kiefer-Wolfowitz MLE makes it feasible to study its performance much more easily.

7.1 Four Experiments

To this end, I initially adopted a slightly expanded variant of the simulation design of Martin and Walker: sample size is $n = 200$, with 1000 replications, $\theta = \theta_a \in \{1, 3, 5, 7\}$ for s entries, and 0 otherwise, and $s \in \{10, 20, 40, 80\}$. The default settings for the EBMW procedure including their Gibbs parameters are maintained. Table 1 reports the results of the experiment. For each replication we compute the sum of squared errors for the n observations, these are then averaged over the 1000 replications and rounded to integers, following standard practice as introduced in Johnstone and Silverman (2004).

	s = 10				s = 20				s = 40				s = 80			
	1	3	5	7	1	3	5	7	1	3	5	7	1	3	5	7
EBMW	10	56	22	13	20	98	36	25	40	162	62	48	79	249	106	92
EBKM	13	36	15	7	21	52	19	7	32	74	25	8	44	94	29	8

Table 7.1: MSE based on 1000 replications

As a second experiment I considered an expanded version of the second experiment described in Martin and Walker. This experiment employs a design used by Castillo and

van der Vaart (2012). The setup is very similar to the first experiment, except that $n = 500$, $s \in \{25, 50, 100\}$ and θ_a takes five possible values: $\{1, 2, \dots, 5\}$. The first 8 rows of Table 2 are taken directly from Table 1 of Castillo and van der Vaart (2012), with identifiers as described there; the ninth row is taken from Martin and Walker (2013), each based on 100 replications. In the tenth through twelfth rows we report results for a slightly expanded version of the design including $\theta_a \in \{1, 2\}$ for both the EBMW procedure and the EBKM procedure, both based on 1000 replications. In this table I have also added the performance of the monotone Bayes rule estimator, EBMR, introduced in Koenker and Mizera (2013). It performs slightly worse than EBKM, the Kiefer-Wolfowitz procedure, but considerably better than the other procedures, except in the first column of the table where the Martin and Walker estimator is the (narrow) winner. It might be initially surprising to see that the EBKM and EBMR estimators have $\text{MSE} < s$ for $s = 50$ and $s = 100$ when $\theta_a = 5$. However, when the signal is sufficiently separated from the zeros, the Kiefer-Wolfowitz estimator of the mixing distribution quite accurately estimates the two mass points and thus the Bayes rule provides not only shrinkage toward zero, but also toward this estimate of θ_a .

	s = 25					s = 50					s = 100				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
PM1			111	96	94			176	165	154			267	302	307
PM2			106	92	82			169	165	152			269	280	274
EBM			103	96	93			166	177	174			271	312	319
PMed1			129	83	73			205	149	130			255	279	283
PMed2			125	86	68			187	148	129			273	254	245
EBMed			110	81	72			162	148	142			255	294	300
HT			175	142	70			339	284	135			676	564	252
HTO			136	92	84			206	159	139			306	261	245
EBMW			142	98	53			240	160	93			399	262	151
EBMW	25	94	143	103	54	50	183	244	165	91	99	347	404	258	155
EBMR	30	77	89	65	35	50	123	136	92	48	79	185	193	127	62
EBKM	27	71	80	57	30	46	113	122	81	40	74	171	174	112	53

Given the foregoing results, I was curious to see whether the improvement of the empirical Bayes (Kiefer-Wolfowitz) approach over the methods of Martin and Walker would persist if I replaced the point mass signals with more diffuse signals. So as a third experiment I maintained all the features of the second experiment except that the non-zero elements of the θ vector were now generated as standard Gaussians centered at the θ_a values, rather than as point masses at those values. Since the Kiefer-Wolfowitz MLE produces a small number of point masses my expectation was that this would make things more difficult for EBKM. The results of this experiment are reported in Table 3. As expected, this variant of the problem is more challenging, but the gap in relative performance persists.

Finally, I thought it might be interesting to compare EBKM with the procedures suggested in Bhattacharya, Pati, Pillai, and Dunson (2012). In their simulation setting, $n = 1000$, and θ_a is 10 for the first 10 coordinates, $\theta_a \in \{2, 3, \dots, 7\}$ for the next 90 coordinates, and zero for the remaining 900. They consider four methods: the Bayesian Lasso

	s = 25					s = 50					s = 100				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
EBMW	47	99	117	91	59	108	195	198	169	112	145	281	329	259	173
EBKM	43	69	76	64	45	79	121	123	111	78	104	165	186	165	121

Table 7.2: MSE based on 1000 replications

(BL), two variants of their Dirichlet-Laplace priors with different prior strength ($DL(\alpha)$), and the horseshoe prior of Carvalho, Polson, and Scott (2009). Results in Table 4 are again substantially better than those reported by Bhattacharya et al. The first four rows of the table are taken from Bhattacharya, Pati, Pillai, and Dunson (2012) and are based on 100 replications; the last two rows are based on 1000 replications as in the previous tables. Again the empirical Bayes (Kiefer-Wolfowitz) procedure substantially outperforms the others and this advantage increases with degree of separation of the signal from the zeros.

	2	3	4	5	6	7
BL	299	386	424	450	474	493
DL(1/n)	307	354	271	205	183	169
DL(1/2)	368	679	671	374	214	160
HS	268	316	267	213	193	177
EBMW	324	439	306	175	130	123
EBKM	207	225	151	78	45	38

Table 7.3: MSE based on 100 replications for the first four rows and on 1000 replications for the last two rows

In contrast to all the other procedures considered above, the Bayes rule that emerges from the Kiefer-Wolfowitz estimator of the mixing distribution, i.e. EBKM, places no special significance on the notion of “sparsity,” or at least none on the special status of the number zero. All of the other procedures purport to know that most of the θ_i ’s are zero, and the simulation designs justify the confidence placed on this “prior” knowledge. For Kiefer-Wolfowitz zero is just another number, indeed the Kiefer-Wolfowitz estimator of the mixing distribution is equivariant to shifts of sample location. Thus, if we were to shift our simulations from mixtures $\theta_i \sim \alpha\delta_0 + (1 - \alpha)\delta_{\theta_a}$ to $\alpha\delta_b + (1 - \alpha)\delta_{b+\theta_a}$, we would obtain the same MSE’s for EBKM while the other procedures would suffer increased losses for any $b \neq 0$. Of course, zero, is a natural hypothesis in many applications and it would be entirely reasonable to consider informative priors/penalties that nudged the Kiefer-Wolfowitz MLE back toward zero. A further advantage, however, of the original Kiefer-Wolfowitz approach is that it requires no preliminary tuning parameter elicitation/selection. It may be objected that the discretization of the convex optimization problem underlying the Kiefer-Wolfowitz method requires a choice of a grid for mass points of the mixing distribution, and the optimization itself requires a convergence tolerance. But in my experience these choices are quite benign,

a few hundred equally spaced grid values and a convergence tolerance like $\epsilon = 10^{-9}$ ensure highly accurate solutions to what is, after all, a convex problem.

It is perhaps also worth noting in view of recent discussions of the tradeoff between computational difficulty and statistical efficiency that although the first of the foregoing experiments was conducted with R code made available by Ryan Martin, I found that the Gibbs procedure as provided could be significantly speeded up by a bit of vectorization. Thus, in the remaining experiments results for the EBMW estimator are based on the vectorized version. The EBKM procedure employs the default `GLmix` function from the `REBayes` package, Koenker (2013), for the R language R Core Team (2013). Code to reproduce the experimental results reported above is available from <http://www.econ.uiuc.edu/~roger/research/ebayes/ebayes.html>.

7.2 Some Oracle Comparisons

It would be natural to wonder at this point whether the results reported above are too good to be true. An obvious benchmark would be the performance of the Bayes rule based on the unknown mixing distribution function F . In the simulations settings we have already considered there are a fixed number of non-zero observations, but for the oracle comparisons we consider priors with a multinomial number of non-zeros with expected frequencies equal to the fixed frequencies imposed in the earlier simulations.

In Tables 5, 6 and 8 below the oracle knows exactly the value of θ_a and the *expected* number of $\theta_i \neq 0$; a smarter oracle might be assumed to know the θ_a and the *actual* number of $\theta_i \neq 0$, but this formulation yields essentially similar results. In Table 7 where, like Table 3, the non-null θ_i 's are normally distributed around a θ_a , the oracle is assumed to know only the mean, θ_a .

Using this modified schema for each of our previous simulation designs we report in Tables A5-8 a comparison of performance of the EBKM estimator with the oracle estimator based on the known F distribution corresponding to each of the designs in Tables 1-4. Not surprisingly, the performance of the EBKM estimator is quite similar to its performance with fixed proportions. The oracle estimator is consistently better, but only by a relatively small margin. These results strongly support the convergence result for relative Bayes risk established in Jiang and Zhang (2009).

	s = 10				s = 20				s = 40				s = 80			
	1	3	5	7	1	3	5	7	1	3	5	7	1	3	5	7
Oracle	11	35	10	1	15	45	14	1	24	60	17	1	37	84	24	2
EBKM	15	40	15	6	20	51	20	7	28	66	23	7	42	90	30	9

Table 7.4: MSE based on 1000 replications

	s = 25					s = 50					s = 100				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
Oracle	9	25	29	19	9	16	43	46	30	14	27	66	68	43	19
EBKM	10	28	32	22	11	19	46	49	33	17	30	69	71	46	22

Table 7.5: MSE based on 1000 replications

	s = 25					s = 50					s = 100				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
Oracle	36	60	69	59	41	64	100	112	97	70	109	161	178	154	114
EBKM	41	66	76	66	49	70	107	120	105	78	116	169	187	163	124

Table 7.6: MSE based on 1000 replications

	2	3	4	5	6	7
Oracle	171	189	126	62	33	27
EBKM	180	198	135	71	42	37

Table 7.7: MSE 1000 replications

7.3 Envoi

A reviewer has noted that the theory developed by Jiang and Zhang (2009) for the adaptive minimaxity of the Bayes risk of Kiefer-Wolfowitz procedures excludes settings for ℓ_p classes with $p = 0$. For these “nearly black” classes – to use the terminology of Johnstone and Silverman (2004) – with,

$$\ell_0(\eta) = \{\theta = (\theta_1, \dots, \theta_n) \mid n^{-1} \sum_{i=1}^n I(\theta_i \neq 0) < \eta\}$$

and typifying the experimental settings employed here, little is presently known. I would like to think that the foregoing experimental results might encourage others to explore these questions further.

Maximum Entropy Deregularized Density Estimation

1

8.1 Introduction

Suppose that you would like to estimate a unimodal density. There are several exploratory inference approaches that could be employed, notably Cox (1966), Silverman (1981), Silverman (1983) and Hartigan and Hartigan (1985). More recently interest has focused on maximum likelihood estimation of log concave densities as described by Rufibach (2007), Walther (2009) and Cule, Samworth, and Stewart (2010), who offer a more direct approach to estimation of *strongly* unimodal densities as characterized by Ibragimov (1956). Weaker notions of unimodality have been explored in Koenker and Mizera (2010) and Han and Wellner (2016). In this note I would like to briefly describe some further experience with these weaker forms of unimodality and their relevance for both shape constrained estimation and norm constrained estimation of densities.

Our path leads us away from the well paved highway of maximum likelihood, but it is arguably more scenic. I will briefly review means of order ρ and their connection to classes of concave densities and Rényi entropy, and then illustrate their application to shape constrained and norm constrained density estimation. Further details about the computational methods are provided in Section 8.4.

8.2 Means of order ρ and a Hierarchy of Concave Densities

A natural hierarchy of concave functions can be built on the foundation of the weighted means of order ρ studied by Hardy, Littlewood, and Pólya (1934). For any p in the unit simplex, $\mathcal{S} = \{p \in \mathbb{R}^n | p \geq 0, \sum p_i = 1\}$, let

¹ The author would like to thank Ivan Mizera and Jon Wellner for very helpful conversations, and Fatih Guvenen for providing the kernel density estimate reconsidered in Example 1.

$$M_\rho(a; p) = M_\rho(a_1, \dots, a_n; p) = \left(\sum_{i=1}^n p_i a_i^\rho \right)^{1/\rho}, \quad \rho \neq 0,$$

with $M_0(a; p) = M_\rho(a_1, \dots, a_n; p) = \prod_{i=1}^n a_i^{p_i}$ as a limit as $\rho \rightarrow 0$. The familiar arithmetic, geometric, and harmonic means correspond to ρ equal to 1, 0, and -1 , respectively.

Following Avriel (1972), a non-negative, real function f , defined on a convex set $C \subset \mathbb{R}^d$ is called ρ -concave if for any $x_0, x_1 \in C$, and $p \in \mathcal{S}$,

$$f(p_0 x_0 + p_1 x_1) \geq M_\rho(f(x_0), f(x_1); p).$$

In this terminology log-concave functions are 0-concave, and concave functions are 1-concave. Since $M_\rho(a, p)$ is monotone increasing in ρ for $a \geq 0$ and any $p \in \mathcal{S}$, it follows that if f is ρ -concave, then f is also ρ' -concave for any $\rho' < \rho$. Thus, concave functions are log-concave, but not vice-versa. In the limit $-\infty$ -concave functions satisfy the condition

$$f(p_0 x_0 + p_1 x_1) \geq \min\{f(x_0), f(x_1)\},$$

so they are *quasi-concave*, and consequently so are all ρ -concave functions. Further details and motivation for ρ -concave densities can be found in Prékopa (1973), Borell (1975), and Dharmadhikari and Joag-Dev (1988).

8.2.1 Estimation of log concave densities

A probability density function, f , is called *log-concave* if $-\log f$ is a (proper) convex function on the support of f . Maximum likelihood estimation of log concave densities can be formulated as a convex optimization problem. Let $X = \{X_1, \dots, X_n\}$ be a collection of data points in $\mathbb{R}^d$ such that the convex hull of X , $\mathcal{H}(X)$, has a nonempty interior in $\mathbb{R}^d$; such a configuration occurs with probability 1 if $n \geq d$ and the X_i behave like a random sample from f_0 , a probability density with respect to the Lebesgue measure on $\mathbb{R}^d$. Setting $g = -\log f$, we can formulate the maximum likelihood problem as,

$$\min_g \left\{ \sum_{i=1}^n g(X_i) \mid \int e^{-g} dx = 1, g \in \mathcal{K}(X) \right\}, \quad (\mathbb{P}_0)$$

where $\mathcal{K}(X)$ denotes the class of closed convex functions on $\mathcal{H}(X) \subset \mathbb{R}^d$. As shown in Koenker and Mizera (2010) such problems have solutions that admit a finite dimensional characterization determined by the function values of $\hat{g}$ evaluated at the observed X_i with values of g elsewhere determined by linear interpolation. This primal problem has an equivalent dual formulation as,

$$\max_f \left\{ - \int f \log f dy \mid f = (d(P_n - G))/dy, \quad G \in \mathcal{K}(X)^o \right\}, \quad (\mathcal{D}_0)$$

where $\mathcal{K}(X)^o$ denotes the polar cone corresponding to $\mathcal{K}(X)$, that is,

$$\mathcal{K}(X)^o = \left\{ G \in \mathcal{C}^*(X) \mid \int g dG \leq 0 \text{ for all } g \in \mathcal{K}(X) \right\},$$

and $\mathcal{C}^*(X)$ denotes the space of (signed) Radon measures on $\mathcal{H}(X)$, its distinguished element is P_n , the empirical measure supported by the data points $\{X_i, i = 1, \dots, X_n\}$.

It is a notable feature of the log concave MLE that it is tuning parameter free, well posed without any further need for regularization. This feature carries over to weaker forms of concave, shape constrained estimators we will consider next. It is hardly surprising in view of prior likelihood experience that our dual formulation involves maximizing Shannon entropy, since we are already well aware of the close connection to Kullback-Leibler divergence. One potentially disturbing aspect of foregoing formulation is the finding that solutions, $\hat{g}$ must be piecewise linear, so the estimated density, $\hat{f}$ must be piecewise exponential, which when extrapolated into the tails implies sub-exponential tail behavior. This finding motivated consideration of weaker forms of concavity that permit heavier tail behavior as well as more peaked densities. A second, seemingly anomalous feature of the dual problem is the fact that G must be chosen to annihilate the jumps in P_n in order to produce a density, f , with respect to Lebesgue measure. Further details on this may be found in Koenker and Mizera (2010).

8.2.2 Rényi likelihood and weaker forms of concave densities

Given the appearance of Shannon entropy in the likelihood formulation of log concave density estimation, it is natural, or at least tempting, to consider the family of Rényi (1961) entropies,

$$R_\alpha(f) = (1 - \alpha)^{-1} \log\left(\int f^\alpha(x) dx\right)$$

as a vehicle for estimating ρ -concave densities. Shannon is conveniently nested as $\alpha = 1$ in this family. We will focus attention on $\alpha < 1$, corresponding to $\rho = \alpha - 1 < 0$. Maximizing $R_\alpha(f)$ with respect to f is equivalent to maximizing, for fixed $\alpha < 1$,

$$\alpha^{-1} \int f^\alpha(x) dx.$$

This yields the new pairing of primal and dual problems:²

$$\min_g \left\{ \sum_{i=1}^n g(X_i) + \frac{|1 - \alpha|}{\alpha} \int g^\beta dx \mid g \in \mathcal{K}(X) \right\}, \quad (\mathbb{P}_\alpha)$$

and

$$\max_f \left\{ \frac{1}{\alpha} \int f^\alpha(y) dy \mid f = d(P_n - G)/dy, \quad G \in \mathcal{K}(X)^o \right\}, \quad (\mathcal{D}_\alpha)$$

with α and β conjugates in the usual sense that $\alpha^{-1} + \beta^{-1} = 1$.

This formulation weakens the unimodality constraint admitting a larger class of heavier tailed, more peaked densities; at the same time it modifies the fidelity criterion replacing log likelihood with a criterion based on Rényi entropy. Why not stick with log likelihood and just modify the constraint, as suggested by Seregin and Wellner (2010)? The pragmatic

² The primal version of this pairing corrects an error in formula (3.7) of Koenker and Mizera (2010) in the event that $\alpha < 0$.

reason is that modifying both preserves an extremely convenient form of the convex optimization problem. This motivation is further elaborated in Koenker and Mizera (2010). From a more theoretical perspective weaker concavity requirements pose difficulties for the standard likelihood formulation, Doss and Wellner (2016) elaborate on these difficulties and demonstrate among many other things that when imposing concavity constraints with $\alpha < 0$ the MLE fails to exist.

Example 1 In Koenker and Mizera (2010) we stressed the (Hellinger) case that $\alpha = 1/2$, so $\beta = -1$, and $g = -1/\sqrt{f}$. Since g is constrained to be concave we conclude that the estimated density, f is ρ -concave for $\rho = -1/2$, a class that includes all of the Student t densities with degrees of freedom, $\nu \geq 1$, as well as all the log concaves. To illustrate the applicability of this Hellinger criterion within the Rényi family, we reconsider a density estimation problem arising in econometrics. Guvenen, Karahan, Ozkan, and Song (2016) have estimated models of income dynamics using very large (10 percent) samples of U.S. Social Security records linked to W2 data. Their work reveals quite extreme tail behavior in annual log income increments. In the left panel of Figure 8.1 we reproduce the Guvenen et al plot of a conventional kernel density estimate of the log density of annual increments of log income based on their sample. Clearly, this density is not log-concave, however when we plot instead $-1/\sqrt{f}$ we see that concavity looks extremely plausible. When the Rényi estimate is superimposed in red, it fits almost perfectly.

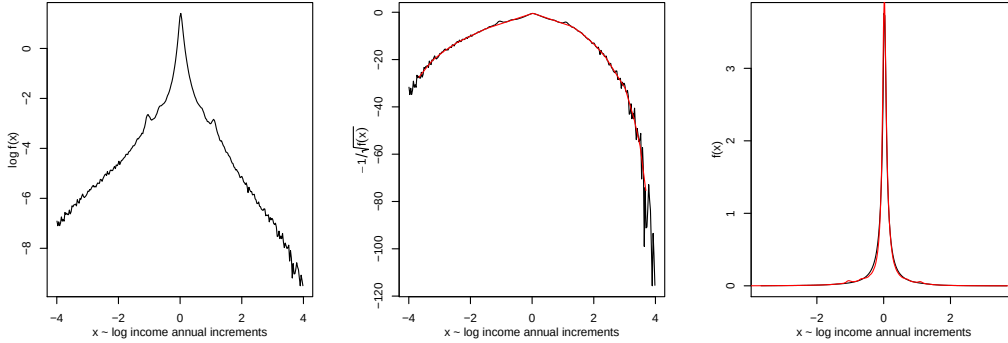


Fig. 8.1: Density estimation of annual increments in log income for U.S. individuals over the period 1994-2013. The left panel of the figure reproduces a plot of the logarithm of a kernel density estimate from Guvenen showing that annual income increments are clearly not log concave. However the middle panel showing $-1/\sqrt{f}$ does appear to be nicely concave and is fit remarkably well by the Rényi procedure with $\alpha = 1/2$.

Permitting Cauchy tail behavior may be regarded as sufficiently indulgent for most statistical purposes, but the next example illustrates that more extreme Rényi fitting criteria with $\alpha < 1/2$ is sometimes needed to accommodate sharp peaks in the target density.

Example 2 We reconsider the rotational velocity of stars data considered previously in Koenker and Mizera (2010). The data was taken originally from Hoffleit and Warren (1991) and is available from the R package **REBayes**. Figure 8.2 illustrates a histogram of the 3806 positive rotational velocities from the original sample of 3933. After dropping the 127 zero velocity observations, the histogram looks plausibly unimodal and we superimpose three distinct Rényi shape constrained estimates. The Hellinger ($\alpha = 1/2$) estimate is clearly incapable of capturing the sharp peak around $x = 18$, and even the fit for $\alpha = 0$ fails to do so. But pressing further, we see that setting $\alpha = -2$ provides an excellent fit by constraining $-1/f^3$ to be concave.

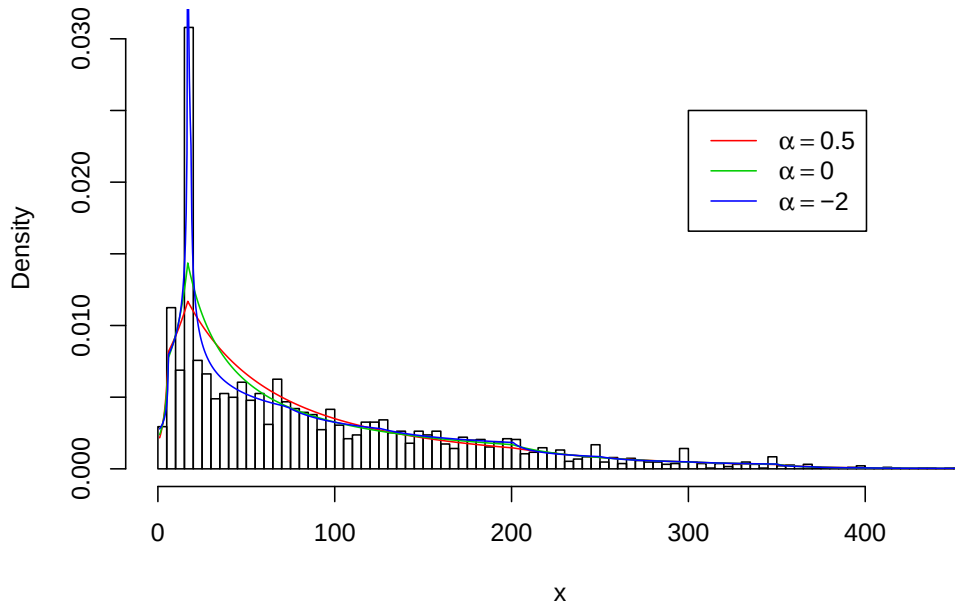


Fig. 8.2: Rotational velocity of stars with three quasi concave shape constrained density estimates using the Rényi likelihood.

8.3 Rényi likelihood and norm constrained density estimation

Although our original intent for the Rényi likelihood was strictly pragmatic – to maintain the convexity of the optimization problem underlying the estimation while maintaining weaker forms of the concavity constraint – I would now like to briefly consider its use in norm constrained settings where the objective of penalization is smoothness of the estimated density rather than shape constraint.

There is a long tradition of norm penalized nonparametric maximum likelihood estimation of densities. Perhaps the earliest example is Good (1971) who proposed the penalty,

$$J(f) = \int (\sqrt{f}')^2 dx,$$

which shrinks the estimated density toward densities with smaller Fisher information for location. The deeper rationale for this form of shrinkage remains obscure, and most of the subsequent literature has instead focused on penalizing derivatives of $\log f$, with the familiar cubic smoothing spline penalty,

$$J(f) = \int (\log f'')^2 dx,$$

receiving most of the attention. Silverman (1982) proposed penalizing the squared L_2 norm of the *third* derivative of $\log f$ as a means of shrinking toward the Gaussian density.

Squared L_2 norm penalties are ideal for smoothly varying densities, but they abhor sharp bends and kinks, so there has been some interest in exploring total variation penalization as a way to expand the scope of penalty methods. The taut-string methods of Davies and Kovac (2001) and Davies and Kovac (2004) penalize total variation of the density itself. Koenker and Mizera (2007) describe some experience with penalties of the form,

$$J(f) = \int |\log f''| dx,$$

that penalize the total variation of the first derivative of $\log f$. In the spirit of Silverman (1982) the next example illustrates penalization of the total variation of the third derivative of $\log f$, again with the intent of shrinking toward the Gaussian, but in a manner somewhat more tolerant of abrupt changes in the derivatives than with Silverman's squared L_2 norm.

Example 3 *In Figure 8.3 we illustrate a histogram based on 500 standard Gaussian observations, and superimpose two fitted densities estimated by penalized maximum likelihood as solutions to*

$$\min_f \left\{ - \sum_{i=1}^n \log f(X_i) + \lambda \int |\log f'''| dx, \right.$$

for two choices of λ . For λ sufficiently large solutions to this problem conform to the parametric Gaussian MLE since the penalty forces the solution to take a Gaussian shape, but does not constrain the location or scale of the estimated density. For smaller λ we obtain a more oscillatory estimate than conforms more closely to the vagaries of the histogram.

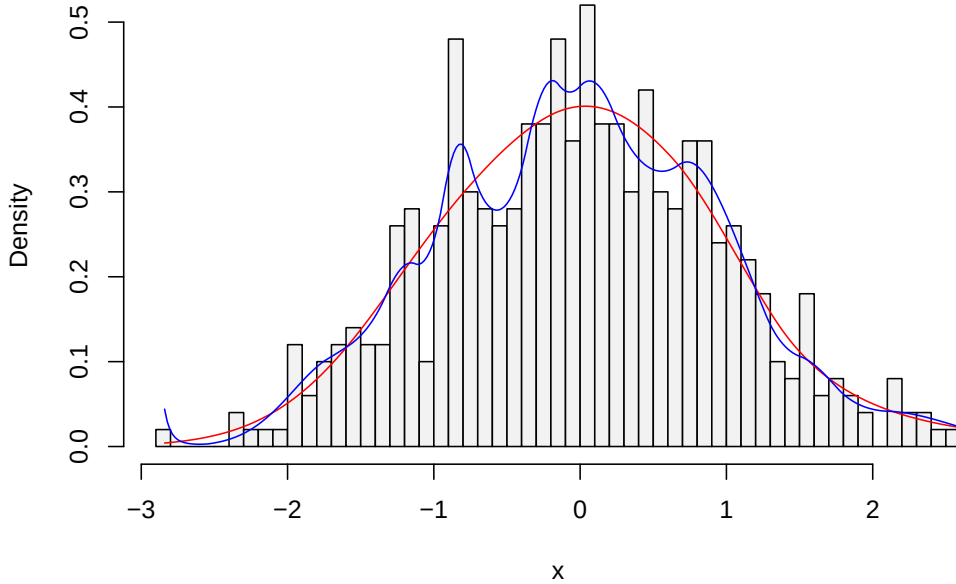


Fig. 8.3: Gaussian histogram based on 500 observations and two penalized maximum likelihood estimates with total variation norm penalty and $\lambda \in \{0.5 \times 10^{-4}, 0.5 \times 10^{-6}\}$.

Penalizing total variation of $\log f''$ as in Figure 8.3 raises the question: What about other Rényi exponents for $\alpha \neq 1$? Penalizing $\log f''$ is implicitly presuming sub-exponential tail behavior that may be better controlled by weaker Rényi penalties. To explore this conjecture we consider a in the next example estimating a mixture of three lognormals.

Example 4 Figure 8.4 illustrates a histogram based on 500 observations from the mixture of lognormals density depicted in red. I have used this density for several years in class to illustrate how difficult it can be to choose an effective bandwidth for conventional kernel density estimation. A bandwidth sufficiently small to distinguish the two left-most modes is almost surely incapable of producing a smooth fit to the upper mode. Logspline methods as proposed by Kooperberg and Stone (1991) perform much better in such cases, but they can be sensitive to knot selection strategies. The methods under consideration here are allied more closely to the smoothing spline literature, and thereby circumvent the knot selection task, but in so doing have introduced new knobs to turn and buttons to push. Not only do we need to choose the familiar λ , there is now a choice of the order of the derivative in the penalty, and the Rényi exponent, α , determining the transformation of the density. I would argue that these choices are more easily adapted to particular applications, but others may disagree.

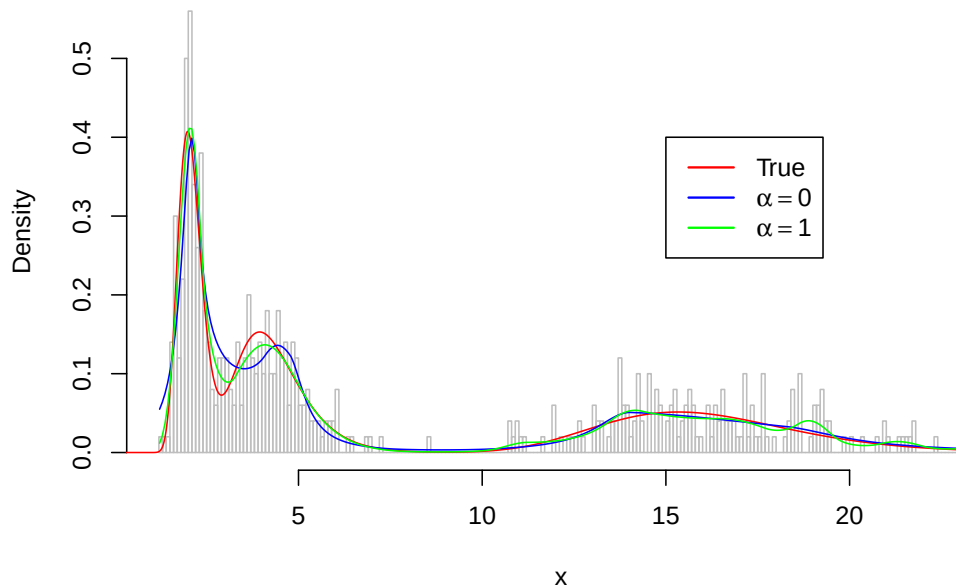


Fig. 8.4: Mixture of three lognormals histogram and two Rényi likelihood estimates with total variation (L_1 norm) penalty with $\alpha \in \{0, 1\}$.

From a Bayesian perspective, however, it seems indisputable that more diversity in the class of tractable prior specifications is desirable.

Examining Figure 8.4 we see that the $\alpha = 1$ maximum likelihood estimate is a bit too smooth, failing to find the second mode, whereas the $\alpha = 0$ solution is too enthusiastic about the fitting the first mode, but at least does distinguish the second mode. Both methods produce an excellent fit to the third mode, almost indistinguishable from the true density.

8.4 Discrete Implementation Details

The discrete formulation of the variational problems described above lead to extremely efficient algorithms that exploit modern interior point methods for convex optimization. All of our computational results were carried out with the function `medde` from the package **RE-Bayes** for the R language and available from the CRAN website, <https://cran.r-project.org>. This package relies in turn on the Andersen (2010) optimization system and its Friberg (2012) interface for the R language. The current implementation in `medde` is restricted to univariate densities as we will do here as well. Koenker and Mizera (2010) describes some

extensions to bivariate settings. Most of the other functionality of the **REBayes** package is devoted to empirical Bayes methods and described in Koenker and Gu (2017).

For univariate densities convexity of piecewise linear functions can be enforced by imposing linear inequality constraints on the set of function values $\gamma_i = g(\xi_i)$ at selected points $\xi_1, \xi_2, \dots, \xi_m$. We typically choose these ξ_i 's on an equally spaced grid of a few hundred points, and the convex cone constraint can then be expressed as $D\gamma \geq 0$ for a tridiagonal matrix D when the penalization is imposed on second derivatives as in the case of our concavity constraints, and quindagonal in the case of third derivative constraints. By default in **medde** we choose $m = 300$, with the ξ_i 's support extending slightly beyond the empirical support of the observations.

As described in Koenker and Mizera (2010) the primal formulation of the shape-constrained problem takes the discrete form,

$$\{w^\top L\gamma + s^\top \Psi(\gamma) \mid D\gamma \geq 0\} = \min! \quad (\text{P})$$

where $\Psi(\gamma)$ denotes an m -vector with typical element $\psi(g(\xi_i)) = \psi(\gamma_i)$, L is an “evaluation operator” which either selects the elements from γ , or performs the appropriate linear interpolation from the neighboring ones, so that $L\gamma$ denotes the n -vector with typical element, $g(X_i)$, and w is an n -vector of observation weights, typically $w_i \equiv 1/n$. The matrix D is the discrete derivative operator that constrains the fitted function to lie in the convex cone $\mathcal{K}(X)$. The vector s denotes weights that impose the integrability constraint on the fitted density. As long as the grid is sufficiently fine in univariate settings elements of s can be averages of the adjacent spacings between the ξ_i 's.

Associated with the primal problem (P) is the dual problem,

$$\{-s^\top \Psi^*(-\phi) \mid S\phi = -w^\top L + D^\top \eta, \phi \geq 0, D^\top \eta \geq 0\} = \max! \quad (\text{D})$$

Here, η is an m -vector of dual variables and ϕ is an m -vector of function values representing the density evaluated at the ξ_i 's, and $S = \text{diag}(s)$. The vector Ψ^* is the convex conjugate of Ψ defined coordinate-wise with typical element $\psi^*(y) = \sup_x \{yx - \psi(x)\}$. Problems (P) and (D) are strongly dual in the sense of the following result, which may be viewed as the discrete counterpart of Theorem 2 of Koenker and Mizera (2010).

For $\Psi(x)$ with typical element $\psi(x) = e^{-x}$ we have Ψ^* with elements $\psi^*(y) = -y \log y + y$, so the dual problem corresponding to maximum likelihood can be interpreted as maximizing the Shannon entropy of the estimated density subject to the constraints appearing in (D). Since g was interpreted in (P) as $\log f$ this result justifies our interpretation of solutions of (D) as densities provided that they satisfy our integrability condition. This is easily verified and thus justifies the implicit Lagrange multiplier of one on the integrability constraint in (P). Then solutions ϕ of (D) satisfy $s^\top \phi = 1$ and $\phi \geq 0$, as shown in Proposition 2 of Koenker and Mizera (2010). The crucial element of the proof of this proposition is that the differencing operator D annihilates the constant vector and therefore the result extends immediately to other norm-type penalties as well as to the other entropy objectives that we have discussed. Indeed, since the second difference operator representing our convexity constraint annihilates any affine function it follows by the same argument that the mean of the estimated density also coincides with the sample mean of the observed X_i 's.

For penalties with Rényi exponents less than one, the dual formulation takes $\psi(y) = y^\alpha$ except of course for $\alpha = 0$ for which $\psi(y) = \log y$. To implement the total variation regularization rather than the concavity constraint, the L_1 constraint on $D\gamma$ in the primal becomes an L_∞ constraint in the dual, so in the dual formulation we simply constrain $\|D^\top \eta\|_\infty \leq \lambda$, and similarly for total variation (L_1 norm) constraints on higher order derivatives.

Code to reproduce each of the figures appearing above is available from **REBayes** package in the file `medde.Rnw` located in the `vignettes` directory. Readers are cautioned that although all of the computational problems described above are strictly convex and therefore possess unique solutions, extreme choices of the parameters can stress even the excellent optimization software provided by Mosek. In Example 2 we have seen that attempts to push the Rényi α parameter much below -1, cause difficulty. In Example 3 a choice of λ somewhat larger than those reported here also causes trouble. Fortunately, it is relatively easy to find values of these parameters that are within an empirically sensible range.

8.5 Conclusion

Density estimation by penalty methods is one of those [IJ] Good ideas of the 1970's that has matured rather slowly. Fortunately, recent developments in convex optimization have greatly expanded the menu of possible penalties, and there are promising opportunities for embedding these methods into more complex semi-parametric analyses.

Many aspects remain to be explored. We have elementary Fisher consistency results from Koenker and Mizera (2010) and some rate and limiting distributional results from Han and Wellner (2016), and others, but there are many interesting theoretical questions. It would be nice to know more about multivariate extensions. Little is known about choice of the Rényi α , can it be estimated in a reasonable way? If only we could divert some energy away from kernel methods, maybe some progress could be made in one or more of these directions.

Edgeworth on Banking

1

These speculations are at an immense height of abstraction above the affairs of earth. Appropriate general conceptions, rather than exact propositions, have been the object of the writer. He does not pretend to base any practical recommendations on the theory. There is some sense as well as absurdity in the sentiment which Thomas Peacock in his amusing *Nightmare Abbey*, puts into the mouth of Mr.Flosky, the caricature of Coleridge. The philosopher having been consulted in a practical emergency, replies: "Madam, I should lose my transcendental reputation, if it could be said of Ferdinand Flosky that he was ever known to give any useful information to any one." The theorist must not pretend to wisdom, if he knows so little what he is about as to mistake his abstract formulae for rules immediately applicable to practice.

F.Y. Edgeworth (1888) Edgeworth's (1888) paper "A Mathematical Theory of Banking" is generally regarded as laying the foundation of modern inventory theory. Although it is often referenced I have been unable to find an explicit link between Edgeworth's formal theory and modern developments. Arrow's (1958) historical survey emphasizes Edgeworth's contribution, but focuses on his observation that reserves should expand like the square-root of deposits without describing the simple model Edgeworth introduces that anticipates the extensive modern literature on what has become known as the "newsvendor problem." The influential paper of Arrow, Harris, and Marschak (1951), which introduced dynamic "sS" rules, also contains a similar static formulation to that of Edgeworth, albeit with an additional complication of a fixed charge for exceeding the chosen inventory level. No mention is made, however, of Edgeworth, nor is any made in Arrow's (2002) more recent memoir. Wagner's (1969) encyclopedic survey of operations research contains a thorough discussion of what he calls the "newsboy problem," but again there is no mention of Edgeworth. My aim in this note is simply to provide a more explicit exegesis of Edgeworth's model in the hope that someone might eventually be inspired to fill this historical gap.

¹ Thanks to Gib Bassett, Steve Portnoy and Minchul Shin for valuable comments on an earlier draft.

Edgeworth's basic model, p. 120-1, takes the form of what he calls a parable; it is really just a simple game. Players are each allocated 100 tokens that they may invest in one of three assets: an illiquid asset that pays 1% per period, a semi-liquid asset that he calls "at call" that pays 0.4% per period, and a cash reserve. Periodically, a positive random variable, Y is generated. Players are then required to pay the amount Y to the game authority. This can be done at no penalty out of reserves if they are sufficient, or out of holdings of the "at call" asset if reserves are insufficient. However, drawing upon the "at call" asset entails a penalty of 10% of the amount required. If Y exceeds the amount invested in the two liquid assets combined, the player must declare bankruptcy and pay a large penalty.

The mechanism for generating Y is amusingly quaint: 22 decimal digits are selected "at random from the pages of some mathematical or statistical tables," and then summed. Presuming these digits are drawn from the least significant in the tables, i.e. ignoring Benford's Law, they will be approximately uniformly distributed and consequently their sum, say X , will be approximately Gaussian with mean 99 and variance, 181.5. Edgeworth computes the modulus of this distribution as 19, which is defined, e.g. Stigler (1990), as the square root of twice the variance. In effect the random variable he then considers is $Y = |X - EX|$.

Edgeworth argues, somewhat cavalierly, that the probability that the half-normal Y exceeds 75 is "one in 100 million" and therefore it is safe to invest 25 of the initial tokens in the illiquid asset. A more pedantic estimate of the risk based on the Gaussian approximation yields one in 259 million, an even more pedantic estimate of the risk based on the discrete uniform convolution is one in 1.561 billion.

How should the remaining 75 be divided? To this question Edgeworth responds very tersely: "the portion held at call should be 56, the reserve 19." Having noted earlier that "it need not be pointed out that the professor of Probability will have a great advantage in this game," it is a matter of natural curiosity to try to determine how he arrived at this split. Although the game as originally described almost begs for a dynamic formulation Edgeworth is at pains to emphasize that it is intended to be a series of one-shot decisions with no reinvested earnings and the original setting restored at the beginning of each play.

We can begin our inquiry by noting that if the portion, say $\hat{Y}$, devoted to the reserve is set too high so $Y < \hat{Y}$, we would have sacrificed $0.004(\hat{Y} - Y)$ in lost interest, while if it is set too low so $Y > \hat{Y}$, we must pay the penalty of $0.1(Y - \hat{Y})$. In such circumstances we know that $\hat{Y}$ should be chosen as the $\tau = 1/1.04$ quantile of the distribution of Y . Now it is easily seen that the quantile function of Y is $Q_Y(\tau) = \sigma\Phi^{-1}((\tau + 1)/2)$, and consequently that $\hat{Y} \approx 27.89$. The split should be 28 in reserve and $75 - 28 = 47$ at call. Something is clearly amiss. Having folded the distribution of Y to obtain a positive random variable, it is tempting to imagine that Edgeworth might have introduced a spurious two. If $Y = 2|X - EX|$, so $Q_Y(\tau) = 2\sigma\Phi^{-1}((\tau + 1)/2)$, then $\hat{Y} \approx 55.77$, which Edgeworth would have rounded to 56. Perfect! But wait! Now Edgeworth seems to have gotten it backwards! Our calculation would imply that 56 should be placed in reserve and 19 at call. We might also want to reconsider allocating 25 to the illiquid asset since the more dispersed Y now has a somewhat higher probability ($\approx 1/827,000$) of exceeding the threshold we had previously set at 75. We are left with something of a quandary. Edgeworth's formulation of the problem is very precise and he is obviously aware of the need for an asymmetric asset allocation in response to the asymmetric losses, but the rationale for his computations remains rather

murky. Our epigraph, drawn from the concluding paragraph of Edgeworth's article, thus seems well vindicated.

Admittedly, this wouldn't seem to be an auspicious beginning for the theory of inventory policy, however it should be remembered that problem formulation is often more important than an authoritative solution. It is fair to say that the extensive literature now referred to as the "newsvendor problem" reviewed in Choi (2012) flows directly from Edgeworth's formulation. And not coincidentally, so does quantile regression.

Bayesian Deconvolution

1

Nonparametric maximum likelihood estimation of general mixture models pioneered by the work of Kiefer and Wolfowitz (1956) has been recently reformulated as an exponential family regression spline problem in Efron (2016). Both approaches yield a low dimensional estimate of the mixing distribution, g -modeling in the terminology of Efron. Some casual empiricism suggests that the Efron approach is preferable when the mixing distribution has a smooth density, while Kiefer-Wolfowitz is preferable for discrete mixing settings. In the classical Gaussian deconvolution problem both maximum likelihood methods appear to be preferable to (Fourier) kernel methods. Kernel smoothing of the Kiefer-Wolfowitz estimator appears to be competitive with the Efron procedure for smooth alternatives.

10.1 Introduction

Efron (2016) has recently introduced the phrase “Bayesian deconvolution” to describe a maximum likelihood procedure for estimating mixture models of the general form,

$$f(y) = \int \varphi(y|\theta) dG(\theta),$$

where φ denotes a known parametric “base” model and G denotes an unknown, nonparametric mixing distribution. Such models are fundamental in empirical Bayes compound decision settings where we have the (iid) hierarchical structure,

$$Y_i \sim \varphi(y|\theta_i); \quad \theta_i \sim G.$$

When θ is a location parameter, so $\varphi(y|\theta_i) = \varphi(y - \theta_i)$ this is a conventional deconvolution problem usually evoking characteristic function methods, however Efron’s maximum likelihood procedure recalls the NPMLE of Kiefer and Wolfowitz (1956) except rather than producing a discrete estimate of G it yields a smooth estimate.

¹ It seemed appropriate that my first venture in this new genre should be samokritika (self-criticism) directed at the R package REBayes, Koenker (2013).

This note contrasts the foregoing methods in a very simple, special case and argues that maximum likelihood offers considerable advantages over prior (Fourier) deconvolution methods, perhaps most significantly by extending the domain of applications beyond the location shift model.

10.2 The Kiefer-Wolfowitz NPMLE

In Koenker and Mizera (2013) we have advocated the Kiefer-Wolfowitz NPMLE approach to estimating G and constructing estimates of the θ_i 's for compound decision problems. In sharp contrast to finite dimensional mixture problems with highly multimodel likelihoods, discrete formulations of the general nonparametric mixture problem are strictly convex and therefore admit unique solutions. Consider a grid $t_0, t_1, \dots, t_m$ with associated masses $\{g \in \mathbb{R}^m | g_i \geq 0, \sum_{i=1}^m g_i \Delta t_i = 1\}$, we can approximate the log likelihood by,

$$\ell(G) = \sum_{i=1}^n \log f_i$$

where the n vector $f = Ag$ and A is the n by m matrix with typical element $\varphi(y_i, t_j)$. As is well known from Laird (1978) or Lindsay (1983) the NPMLE, $\hat{G}$, has $p \leq n$ positive mass points, while in practice this p is usually closer to $\log n$ than n . Interior point methods for solving such problems are considerably more efficient than earlier EM approaches greatly facilitating the study of their performance in simulation experiments. Unfortunately, little is known about their statistical efficiency from a theoretical perspective beyond the basic consistency results of Kiefer and Wolfowitz (1956) and Pfanzagl (1988).

10.3 Efron's NPMLE

Efron (2016) has proposed an alternative approach to estimating G that expresses its log derivative by a regression spline,

$$g(y|\theta) = \exp\left\{\sum_{j=1}^p \theta_j \psi_j(y) - \psi_0(\theta)\right\},$$

as in the pure density estimation methods of Stone (1990) and Barron and Shue (1991). We can maintain the same discretization for the support of G , and set,

$$g = (g_j) = (g(t_j|\theta)),$$

so the log likelihood can be expressed as above, except that now we are estimating a finite dimensional parameter θ of predetermined dimension. Efron suggests natural splines for the ψ_j functions and the penalization,

$$\ell_n(G_\theta) + \lambda \|\theta\|$$

by the Euclidean norm of the vector θ , thereby shrinking θ toward the origin and $\hat{G}$ toward the uniform distribution.

A striking feature of both the Efron and Kiefer-Wolfowitz proposals is that neither depend upon the mixture model being a formal convolution. Of course when θ is a location parameter so $\varphi(y|\theta) = \varphi(y - \theta)$ then classical deconvolution methods are also applicable. Efron compares the performance of his procedure with the kernel deconvolution method of Stefanski and Carroll (1990), and concludes that the latter is “too variable in the tails.”

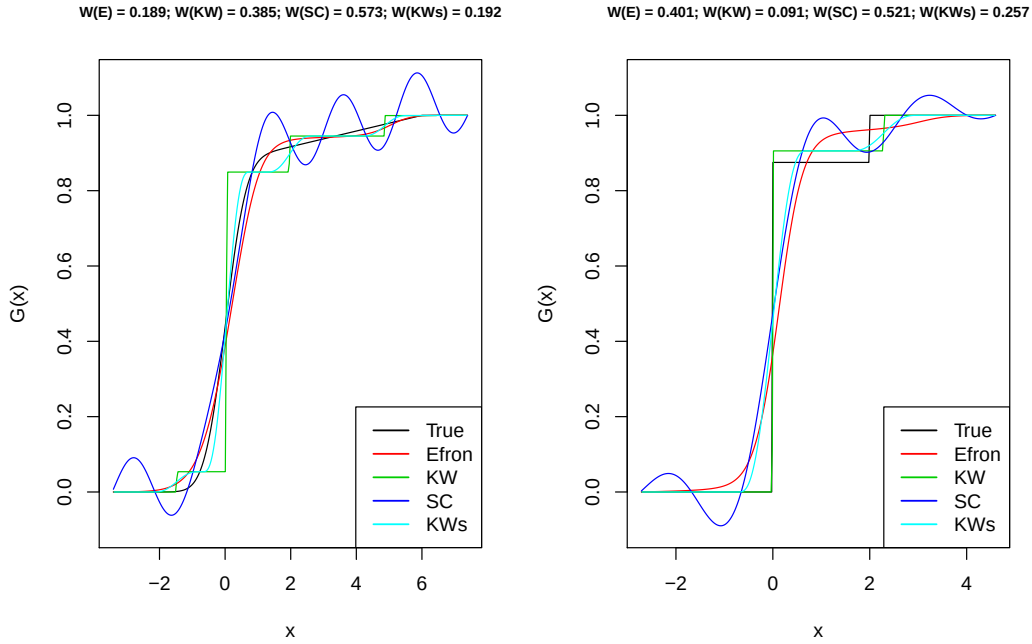


Fig. 10.1: Four estimates of the mixing distributions G : In the left panel the true mixing distribution is smooth, in the right panel it is discrete as described in the text.

10.4 An Illustration

To compare performance of the three estimators of G described above, I have considered a slight variant of the simulation setting of Efron. The observed Y_i are $\mathcal{N}(\theta, 1)$ with θ_i 's drawn iidly either a.) from the mixture of Gaussian and uniform distributions,

$$G(\theta) = (1 - \epsilon)\Phi(\theta/\sigma) + \epsilon\theta I(0 \leq \theta < M)/M$$

with $\epsilon = 1/7$, $\sigma = 1/2$ and $M = 6$, or from b.) the discrete mixing distribution with $\epsilon = 1/7$ and,

$$G(\theta) = (1 - \epsilon)I(0 \leq \theta) + \epsilon I(2 \leq \theta)$$

Figure 1 depicts typical realizations with sample size $n = 1000$. Following Efron we have set the dimension of the natural spline model to $p = 5$, and his penalty parameter to one. The scaling parameter for the Stefanski-Carroll procedure was $1/3$, also following Efron's suggestion. For all three estimators the grid was equally spaced on the support of the observed Y_i with $m = 300$ distinct values. Wasserstein distance, L_1 distance between distribution functions, is reported above the figure for each of three estimates.

The performance of Efron's estimator is very impressive, while the oscillation of the kernel method in the tails confirms Efron's criticism. The Kiefer-Wolfowitz estimator is respectable, at least the estimated $\hat{G}$ stays within the $[0,1]$ bounds, but it is clearly inferior to the smoother Efron procedure. When the target G is discrete with a small number of mass points, the performance advantage, not surprisingly is reversed. One might count the lack of tuning parameters for the Kiefer-Wolfowitz NPMLE as an advantage over its competitors, or not, depending on one's outlook on minimalism.

In the spirit of competition, I couldn't resist trying to smooth the Kiefer Wolfowitz NPMLE to see whether one might be able to approach the performance of the Efron estimator, so the last (cyan) curve is a biweight kernel smooth with bandwidth equal 0.7. This does almost as well as the default Efron procedure for our test case for the smooth alternative, but spoils the auspicious performance for the discrete case. Encouraged by the former improvement I decided to wade a little further out in the water by replicating the experiment. In 1000 trials of the experiment with the smooth G_0 the mean Wasserstein error was 0.186 for the Efron estimator, 0.342 for the unsmoothed KW-NPMLE and 0.18 for the smoothed KW-NPMLE. Of course, this result proves nothing at all, except perhaps that I'm acquainted with an excellent bandwidth oracle.

Quantile Selection Models

11.1 Introduction

Arellano and Bonhomme (2017) have recently proposed a new approach to analysing sample selection effects in the context of a general quantile regression model, thereby extending the classical parametric selection methods of Heckman (1979). A central feature of their approach is the estimation of the parameters of a copula function that determines the dependence between the random components of the (latent) selection model and the observable outcome model. This note contrasts a profile likelihood alternative with the method of moments strategy for estimating the copula parameters proposed by Arellano and Bonhomme (2017). The possibility of likelihood based alternatives was already suggested by Arellano and Bonhomme (2017), so this can be considered a tentative first step in this direction.

11.2 A QRious Likelihood

Quantile regression, at least as it was originally conceived, posits a *local* statistical model focussed on estimating a single conditional quantile function while professing total indifference about the form of adjacent conditional quantile functions. However, when one writes,

$$Q_{Y|X}(\tau|x) = x^\top \beta(\tau) \quad \tau \in (0, 1),$$

a global model for the entire conditional distribution of Y given X has been specified, so it is natural to ask: Can we compute a global likelihood value for fitted models of this form?

To achieve this dubious objective we obviously need estimates of the conditional density of Y at each observed setting of the conditioning covariate vector, X . Given estimates of the conditional quantile function at each $X = x_i$, this is just a matter of smoothing. In problems of moderate size all of the solutions to the problem,

$$\hat{\beta}(\tau) = \operatorname{argmin}_b \sum_{i=1}^n \rho_\tau(y_i - x_i b),$$

can be very efficiently computed by parametric linear programming, and thus for any x we have $\hat{Q}(\tau|x)$ as a piecewise constant (CAGLAD) function on $(0,1)$. In the **quantreg** package for R the invocation,

```
data(stackloss)
fit <- rq(stack.loss ~ stack.x, tau = -1)
fhat <- predict(fit, type = "fhat")
```

loads the infamous stackloss data, fits a model for the entire quantile regression process and computes a list of n conditional densities, one for each of the original observations. Smoothing is done with the function `akj`, which implements Silverman's well-known adaptive kernel density estimator, Silverman (1986) Given these conditional densities it is easy to define a function to compute the log likelihood:

```
logLik.rq.process <- function(fit){
  y <- model.response(model.frame(fit))
  fhat <- predict(fit, type = "fhat")
  fy <- mapply(function(f,y) f(y), fhat, y)
  sum(log(fy))
}
```

To illustrate we can compare the log likelihoods for the unconditional and conditional quantile models as follows,

```
f0 <- rq(stack.loss ~ 1, tau=-1)
f1 <- rq(stack.loss ~ stack.x, tau=-1)
l0 <- logLik(f0)
l1 <- logLik(f1)
```

which yields $l1 - l0 = -34.446 - -70.736 = 36.289$. It remains to be seen if Professor Wilks ghost can be persuaded to divulge a limiting distribution theory for such log likelihood ratio statistics. It is not at all obvious how one might count degrees of freedom for such global quantile regression models. It is also evident that in applications with much larger sample sizes it is impractical, and obviously superfluous to compute *all* the distinct solutions of the QR process. Portnoy (1991) shows that the expected number of distinct solutions is of order $\mathcal{O}(n \log n)$. In what follows we adopt the pragmatic attitude that a grid of a few hundred $\tau \in (0,1)$ is sufficient to produce a reasonable estimate of conditional densities provided a reasonable choice of the initial bandwidth for the pilot estimate of the Silverman procedure is employed.

It may be noted at this point that a fully efficient likelihood procedure would require that we weight the usual QR objective by estimates of the local conditional density. See Koenker (2005) Section 5.3.1. We reserve this (rather utopian) diversion for future exploration.

11.3 Estimation of the QR Selection Model

Leaving aside for the moment (or millennium) the distribution theory for tests based on such LRTs, our motivating application for the QR likelihood is estimation of the quantile regression selection model of Arellano and Bonhomme (2017). Their approach involves GMM estimation of a parametric copula model that captures the dependence between the random components of the latent selection model and the outcome model. The objective function involves evaluating indicator functions along a grid of τ 's evaluated at "rotated" quantile regression estimates. It seemed worth exploring whether the foregoing likelihood approach might provide a more direct way to implement estimation of their model.

Our implementation of the Arellano and Bonhomme (2017) estimator exploits general features of the R protocol for fitting linear models. The model is specified as a generalized formula as introduced in Zeileis and Croissant (2010),

$$y|D \sim X|Z,$$

where y is the observable outcome variable, D is the binary selection indicator, X specifies the conditioning covariates of the outcome model, and the union of X and Z specifies the conditioning covariates of the selection model. The outer wrapper of the fitting functions looks like this:

```
rqsels <- function(formula, data, taus = 1:99/100, rhodom = c(-5,1),
  grid = 0, copula = frankCopula, link = "probit", rhometh = Liksel){
  # Generic Formula is y/D ~ X/Z
  Form <- Formula(formula)
  oform <- formula(Form, lhs = 1, rhs = 1)
  sform <- formula(Form, lhs = 2, rhs = 1:2, collapse = TRUE)
  v <- glm(sform, family = binomial(link = link), data = data)
  D <- model.response(model.frame(v))
  v <- v$fitted[D == 1]
  fit <- rq(oform, tau = taus, data = data, method = "fmb", eps = 1e-4)
  if(grid){
    rhos <- seq(rhodom[1], rhodom[2], length = grid)
    objs <- sapply(rhos,function(x, fit, copula, v)
      rhometh(x, fit, copula, v), fit = fit, copula=copula, v=v)
    return(list(x = rhos, y = objs))
  }
  else
    rhohat <- optimize(rhometh, rhodom, fit = fit, v = v,
      copula = copula)$minimum
  rq.fit.sel(rhohat, fit, copula, v)
}
```

The formula is first parsed into its outcome and selection pieces, and a binary response model is then estimated for the selection variable, D . A propensity score, v is then extracted and evaluated for all the selected observations, i.e., those with $D = 1$. In the next step a

naive QR model is estimated that ignores the sample selection effect; the output of this step serves only as a repository for subsequent information for fitted objects that do account for the selection. Note that the fitting of the QR model automatically drops the $D = 0$ observations since their response variable is coded as “NA”, i.e., missing. QR fitting is done on a grid of $\tau \in (0, 1)$, by default at the percentiles.

At this stage we are almost ready to optimize over the dependence parameter of the copula function that links the selection and outcome equations. We will restrict attention to settings with a scalar copula parameter, by default with the Frank copula, but this could be easily extended to more general parametric settings such as the generalized Frank specification of Arellano and Bonhomme (2017).¹

We will consider two criteria for estimating the copula parameter: the method of moments criterion of Arellano and Bonhomme (2017), and the profile likelihood criterion described in the previous section as implemented in the functions `Momsel` and `Liksel` respectively.

```
Liksel <- function(rho, fit, copula, v) {
  f <- rq.fit.sel(rho, fit, copula, v)
  y <- model.response(model.frame(fit))
  fhats <- predict(f, type = "fhat")
  fy <- mapply(function(f,y) f(y), fhats, y)
  -sum(log(fy))
}
Momsel <- function(rho, fit, copula, v) {
  R <- rq.fit.sel(rho, fit, copula, v)$resid
  sum(apply(R, 2, mean))^2
}
```

Both approaches rely on the novel QR estimation strategy introduced by Arellano and Bonhomme (2017) that modifies the conventional fixed τ weighting of the objective function with a observation specific $\hat{\tau}_i$ depending upon the estimated propensity score and the copula dependence relation,

$$\hat{\tau}_i = C(\tau, v_i; \rho) / v_i,$$

where C denotes the distribution function of the chosen copula function evaluated at the proposed τ , the propensity score, v_i of the i th observation and the trial value of the copula parameter, ρ . This is implemented in the function `rq.fit.sel`.

```
rq.fit.sel <- function(rho, fit, copula, v){
  taus <- fit$tau
  fit$frho <- rho
  x <- model.matrix(terms(fit), model.frame(fit))
  y <- model.response(model.frame(fit))
  cop <- copula(rho)
  for(j in 1:length(taus)) {
```

¹ This would simply entail replacing `optimize` by `optim`, and the interval `rhodom` by a vector of starting values.

```

    u <- pCopula(cbind(taus[j], v), cop)/v
    rhs <- t(x) %*% (1 - u)
    f <- rq.fit.fnb(x, y, taus[j], rhs = rhs, eps = 1e-4)
    fit$coefficients[,j] <- f$coef
    fit$residuals[,j] <- v * ((f$resid <= 0) - u)
  }
  fit
}

```

In the dual formulation of the QR problem that is typically used to construct algorithms, the introduction of this observation specific τ vector is trivially accommodated by changing the right hand side of the dual equality constraints. Instead of a scalar τ with dual constraint $X^\top a = (1 - \tau)X^\top 1$, we can write, $X^\top a = X^\top (1 - u)$, where u denotes the new τ specific vector. Such problems are efficiently solved with the so-called Frisch-Newton linear programming algorithm invoked by `rq.fit.fnb`, and described in detail in Portnoy and Koenker (1997). This is a fortran implementation of the same algorithm used by Arellano and Bonhomme (2017), in Matlab.

Estimated coefficients in our repository `fit` object are replaced by the coefficients of this new solution for each τ on our grid, and residuals are replaced by a vector of scaled and centered residual signs needed to evaluate the moment criterion. This new fitted object is passed either to `Momsel` or `Liksel` which evaluate the copula fitting criteria. In the former case, estimated conditional densities are evaluated at the observed response, logged and summed. In the latter case we simply sum mean discrepancies from the hypothesized moment condition and square the sum.

11.4 A Replication Exercise

To illustrate the foregoing approach we reconsider the model posited in Arellano and Bonhomme (2017).² The `grid` argument in `rqsel` can be used to evaluate the criterion function on a grid of ρ 's equally spaced on the domain `rhodom`. The number of evaluation points in this interval is set by specifying `grid` as a positive integer. This option is primarily intended as an exploratory device for determining an appropriate domain for the optimization for the copula parameter. We will illustrate with a plot of both criteria for all four subsamples, together with an point estimate of ρ based on optimization.

```

Mfit <- as.list(1:4)
Lfit <- as.list(1:4)
MForm <- lw | work ~ ed + age + region + trend + I(trend^2) + I(trend^3) +
  kids_d1 + kids_d2 + kids_d3 + kids_d4 + kids_d5 + kids_d6 |
  I(married * ben)

```

² The Appendix describes some code that transforms the data files distributed from <https://sites.google.com/site/stephanebonhomme/research/> into R data frames that are more efficiently stored and loaded for R.

```

SForm <- lw | work ~ ed + age + region + trend + I(trend^2) + I(trend^3) +
  kids_d1 + kids_d2 + kids_d3 + kids_d4 + kids_d5 + kids_d6 |
  I((1-married) * ben)
load("M.Rda")
G <- G[G[, "married"] == 1,] # Married men
Mfit[[1]] <- rqsel(MForm, data = G, grid = 15, rhometh = Momsel,
  rhodom = c(-3, -0.5))
Lfit[[1]] <- rqsel(MForm, data = G, grid = 15, rhometh = Liksel,
  rhodom = c(-1.5, 0.5))
load("M.Rda")
G <- G[G[, "married"] == 0,] # Single men
Mfit[[2]] <- rqsel(SForm, data = G, grid = 15, rhometh = Momsel,
  rhodom = c(-12, -5))
Lfit[[2]] <- rqsel(SForm, data = G, grid = 15, rhometh = Liksel,
  rhodom = c(-2.5, 1))

load("F.Rda")
G <- G[G[, "married"] == 1,] # Married women
Mfit[[3]] <- rqsel(MForm, data = G, grid = 15, rhometh = Momsel,
  rhodom = c(-3, -0.5))
Lfit[[3]] <- rqsel(MForm, data = G, grid = 15, rhometh = Liksel,
  rhodom = c(-1.5, 0.5))
load("F.Rda")
G <- G[G[, "married"] == 0,] # Single women
Mfit[[4]] <- rqsel(SForm, data = G, grid = 15, rhometh = Momsel,
  rhodom = c(-2.5, 1))
Lfit[[4]] <- rqsel(SForm, data = G, grid = 15, rhometh = Liksel,
  rhodom = c(-2.5, 1))

```

```

Mf <- as.list(1:4)
Lf <- as.list(1:4)
MForm <- lw | work ~ ed + age + region + trend + I(trend^2) + I(trend^3) +
  kids_d1 + kids_d2 + kids_d3 + kids_d4 + kids_d5 + kids_d6 |
  I(married * ben)
SForm <- lw | work ~ ed + age + region + trend + I(trend^2) + I(trend^3) +
  kids_d1 + kids_d2 + kids_d3 + kids_d4 + kids_d5 + kids_d6 |
  I((1-married) * ben)
load("M.Rda")
G <- G[G[, "married"] == 1,] # Married men
Mf[[1]] <- rqsel(MForm, data = G, rhometh = Momsel,
  rhodom = c(-3, -0.5))
Lf[[1]] <- rqsel(MForm, data = G, rhometh = Liksel,
  rhodom = c(-1.5, 0.5))

```

```

load("M.Rda")
G <- G[G[, "married"] == 0,] # Single men
Mf[[2]] <- rqsel(SForm, data = G, rhometh = Momsel,
                rhodom = c(-12, -5))
Lf[[2]] <- rqsel(SForm, data = G, rhometh = Liksel,
                rhodom = c(-2.5, 1))

load("F.Rda")
G <- G[G[, "married"] == 1,] # Married women
Mf[[3]] <- rqsel(MForm, data = G, rhometh = Momsel,
                rhodom = c(-3, -0.5))
Lf[[3]] <- rqsel(MForm, data = G, rhometh = Liksel,
                rhodom = c(-1.5, 0.5))

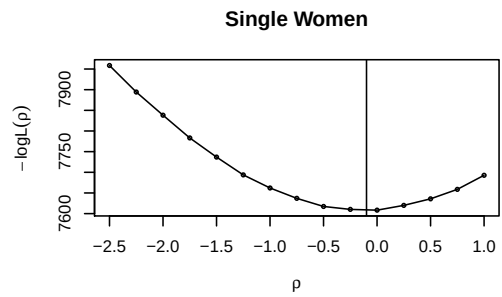
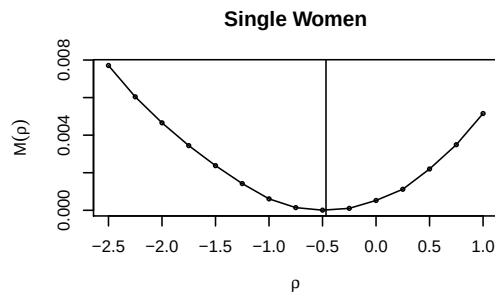
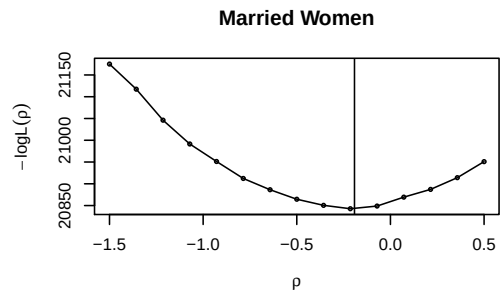
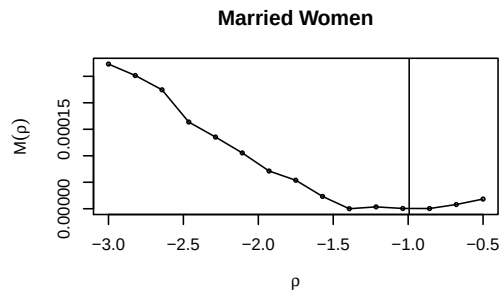
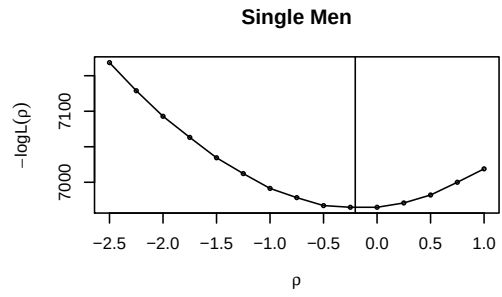
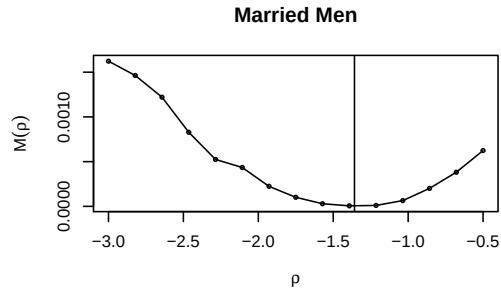
load("F.Rda")
G <- G[G[, "married"] == 0,] # Single women
Mf[[4]] <- rqsel(SForm, data = G, rhometh = Momsel,
                rhodom = c(-2.5, 1))
Lf[[4]] <- rqsel(SForm, data = G, rhometh = Liksel,
                rhodom = c(-2.5, 1))

```

```

par(mfrow = c(4,2))
status <- c("Single", "Married")
gender <- c("Men", "Women")
for(i in 1:4){
  main <- paste(status[1 + (i %% 2)], gender[1 + (i > 2)])
  plot(Mfit[[i]], xlab = expression(rho), ylab = expression(M(rho)),
       main = main, cex = 0.5)
  lines(Mfit[[i]])
  abline(v = Mf[[i]]$frho)
  plot(Lfit[[i]], xlab = expression(rho), ylab = expression(-logL(rho)),
       main = main, cex = 0.5)
  lines(Lfit[[i]])
  abline(v = Lf[[i]]$frho)
}

```



```

SForm <- lw | work ~ ed + age + region + trend + I(trend^2) + I(trend^3) +
  kids_d1 + kids_d2 + kids_d3 + kids_d4 + kids_d5 + kids_d6 |
  I((1-married) * ben)
load("M.Rda")
G <- G[G[, "married"] == 0,] # Single men
Mc <- rqsel(SForm, data = G, rhometh = Momsel,
  link = "cauchit", rhodom = c(-8, -1))

```

The method of moments estimates of ρ are quite close to those obtained by Arellano and Bonhomme (2017), however, there are several puzzles that have yet to be resolved. Foremost among these is the fact that the estimated $\hat{\rho}$'s are consistently more negative for the method of moments criterion than for the log likelihood criterion. For single men this difference is quite substantial. Sensitivity to choices of copula and propensity score models is also of considerable interest. As a very small step in this direction, I tried replacing the probit link with the cauchit for the sample of single men. This had little impact on the likelihood estimate of ρ , but increased the method of moments estimate from -8.482 to -4.327 .

11.5 Data Management

In this appendix we document the procedure employed to simplify the data sources required for the replication of the Arellano and Bonhomme (2017) results. Data from <https://sites.google.com/site/stephanebonhommeresearch/> takes the form of four comma separated value (csv) files. Only two of these are used as detailed in the code below. These files are distinguished by gender, but have substantial overlap in the variables they contain. These files collapsed into two files stored in R save format which is considerably compressed and more efficiently loaded. In the process we have also collapsed groups of binary indicator variables into single R factor variables. It may be noted that the omitted category of these factor variables is given a null name.

```

undummy <- function(A) { # Convert dummy variables to factor
  if(!all(A %in% c(0,1))) stop("not dummies")
  A <- as.matrix(A)
  a <- A %*% rep(1,ncol(A))
  if(sum(a) < length(a)) A <- cbind(1-a, A)
  as.factor(colnames(A)[A %*% 1:ncol(A)])
}
MakeRda <-function(){
  Ddir <- "Bonhomme/Codes_for_replication/Data_files/"
  files <- c("M", "F")
  for(i in 1:2){
    file <- paste(Ddir, "datasub_", i, ".out", sep = "")
    G <- as.matrix(read.csv(file, header = TRUE))
    ed <- undummy(G[,6:7])
    trend <- G[,8]
  }
}

```

```
age <- undummy(G[,11:14])
region <- undummy(G[,15:25])
kids <- G[,26:31]
ben <- G[,4] + G[,5]
ofile <- paste(files[i], ".Rda", sep = "")
G <- data.frame(G[,1:5], ben, ed, age, region, trend, G[,26:31])
save(G, file = ofile)
}
```

The “Tomato Salad Problem” Problem

12.1 Introduction

A fundamental problem in stereology initially considered by Wicksell (1925) involves estimating the distribution of 3d spherical radii from a sample of 2d cross-sectional radii. The problem may be viewed as an idealization of a microscopy setting in which opaque spherical objects embedded in a translucent medium are observable only from 2d slices, or more mundanely as inferring the distribution of the radii of some idealized, spherical tomatoes from a sample of slices. To add an element of verisimilitude to the problem it is convenient to assume that radii, y , are bounded above by $\bar{y}$ and below by $\underline{y}$. Wicksell showed that the relationship between the density of the radii of the slices, f , and the density of the radii of the tomatoes, g , is given by,

$$f(x) = C \int_{\underline{y}}^{\bar{y}} I_{[x, \bar{y}]}(y) x(y^2 - x^2)^{-1/2} dG(y).$$

In an effort to make this look as much like a conventional mixture problem as possible, one can write this as,

$$f(x) = \int_{\underline{y}}^{\bar{y}} \varphi(x, y) dG(y).$$

where $\varphi(x, y) = (x/y)(y^2 - x^2)^{-1/2}$ can be interpreted as a conditional density for x given y , supported on the interval $[0, y]$. To accomodate the truncation we can renormalize the conditional density as done for the missing species Poisson mixture problem.

12.2 Reparameterization

For a moment let’s simplify by setting $\underline{y} = 0$ and $\bar{y} = +\infty$ and use the reparameterization in Groeneboom and Jongbloed (2014) to consider squared radii of both the balls and the circles. Abusing notations slightly we have the following density function of the observed squared radii of the circles

$$f(z) = C \int_z^{+\infty} (y - z)^{-1/2} g(y) dy$$

with $C = 2m_F := 2 \int_0^{+\infty} \sqrt{y} g(y) dy$. Absorbing the constant into $g(y)$ we can reformulate $f(z)$ as

$$f(z) = \int_z^{+\infty} \varphi(z, y) h(y) dy$$

with $\varphi(z, y) = \frac{1}{2}(y - z)^{-1/2}$ and $h(y) := g(y) / \int_0^{+\infty} \sqrt{y} g(y) dy$. This leads to a slight modification of the nonparametric maximum likelihood problem in Koenker and Gu (2017):

$$\min_{h \in \mathcal{H}} \left\{ - \sum_{i=1}^n \log f(z_i) \mid f(z_i) = \int \varphi(z_i, y) h(y) dy, i = 1, \dots, n \right\}$$

where $\mathcal{H}$ denotes the set of functions satisfying

$$\mathcal{H} := \{h(y) : \int_0^{+\infty} \sqrt{y} h(y) dy = 1, h(y) \geq 0 \quad \forall y \in [0, \infty)\}$$

This is again a convex optimization problem and the final estimator for the density g can be obtained as

$$g(y) = h(y) / \int_0^{+\infty} h(y) dy$$

12.3 Two Examples

We consider two simple examples here. The first assumes the true distribution G is standard uniform. In this case, as discussed in Chapter 4.1 of Groeneboom and Jongbloed (2014), we have a closed form for the density $f(z)$ as

$$f(z) = \frac{3}{2} \sqrt{1 - z} 1\{0 \leq z \leq 1\}$$

The second example assumes the true distribution G is standard exponential. Again we have a nice closed form for the density of z that $C = 2m_F = \sqrt{\pi}$ and

$$f(z) = \frac{1}{\sqrt{\pi}} \int_z^{+\infty} \frac{1}{\sqrt{y - z}} e^{-y} dy = e^{-z} 1\{z \geq 0\}$$

In each case we generate 200 realizations. Figure 1 illustrates the NPMLE estimates, the left panel plots the estimates for $g(y)$ and the right panel plots the estimated cumulative distribution $G(y)$ against its true distribution curve in blue.

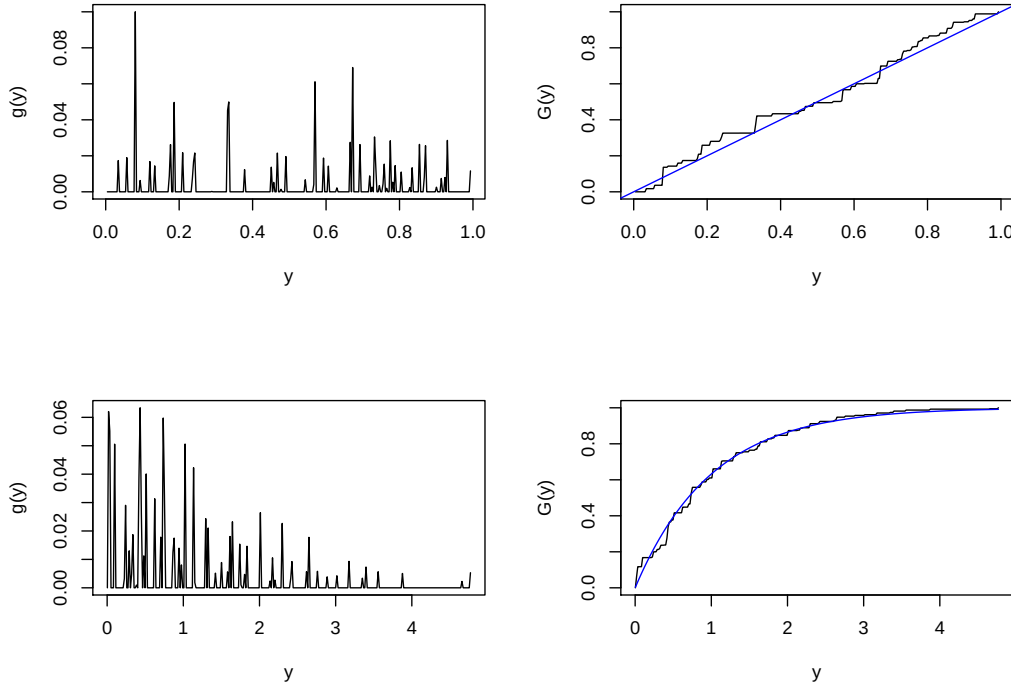


Fig. 12.1: NPMLEs for two Wicksell experiments

12.4 Ill-behaviour of NPMLE and the tuning of grid size

The fit of NPMLE in the above two examples using grid size equals to 300 seem to work like a charm. However a close inspection of the likelihood function suggests that in the deconvolution formulation of the Wicksell problem, the grid size in fact acts like a regularization. This curiously contrasts with mixture models with exponential family kernel (i.e. Gaussian mixture of location with known variance, Poisson mixture and many other examples discussed in Koenker and Gu (2017)) where typically grid size is only a reflection of data resolution: once the grid is refined enough, increasing its fineness does not increase the likelihood further. In contrast, the kernel in the Wicksell problem has a singularity at zero. By choosing $h(y)$ to have a mass point arbitrarily close to any of the data points would lead to an explosion of the likelihood. In light of this, the grid size controls the smallest distance between any grid point and data point, hence acting as a regularization device. In a similar spirit, Jongbloed (2001) suggests¹ restricting the class of function $\mathcal{H}$ to have $h(y)$ taking

¹ Jongbloed used a different parameterization of the problem. Here we stick to the formulation in Groeneboom and Jongbloed (2014 book section 4.1, c.f. ex 4.8 & 4.9).

non-zero values for $y \in \{z_1, \dots, z_n\}$. To be more explicitly, let $z_{(1)} < z_{(2)} < \dots < z_{(n)}$ be the order statistics, and let

$$\bar{\mathcal{H}} := \{h(y), \int_0^{+\infty} \sqrt{y}h(y)dy = 1, h(y) \geq 0, \forall y \in [0, \infty) \text{ and } h(y) > 0 \text{ for } y \in \{z_1, \dots, z_n\}\}$$

For $h \in \bar{\mathcal{H}}$ let $(h_1, \dots, h_n)$ be the vector of weights associated to the grid points which are simply the order statistics of z , then the log-likelihood can be written as $\sum_{i=1}^n \log f(z_{(i)})$ with

$$\begin{aligned} f(z_{(i)}) &= \int_{(z_{(i)}, +\infty)} \frac{1}{2} \frac{1}{\sqrt{x - z_{(i)}}} h(x) dx \\ &= \frac{1}{2} \frac{1}{\sqrt{z_{(i+1)} - z_{(i)}}} h_{i+1} + \frac{1}{2} \frac{1}{\sqrt{z_{(i+2)} - z_{(i)}}} h_{i+2} + \dots + \frac{1}{2} \frac{1}{\sqrt{z_{(n)} - z_{(i)}}} h_n \text{ for } i = 1, \dots, n-1 \\ f(z_{(n)}) &= \int_{(z_{(n)}, +\infty)} \frac{1}{2} \frac{1}{\sqrt{x - z_{(n)}}} h(x) dx = 0 \end{aligned}$$

Clearly the density for the largest sample point would lead to $-\infty$ for the log-likelihood, one simple remedy is to exclude it. Given the fact that $P(z_i = z_j) = 0$ for any $i \neq j$ since Z is a continuous random variable, for all $h \in \bar{\mathcal{H}}$, $\sum_{i=1}^{n-1} \log f(z_{(i)}) < \infty$ hence the NPMLE for $g(y)$ is well-defined when we restrict the class of function to $\bar{\mathcal{H}}$. The Wicksell function can be easily modified to accommodate the discussion above and Figure 2 illustrates the NPMLE when restricted to only allowing $h(y)$ to have jump points on data points.

But one may wonder, why is that an optimal class of function to consider? Our previous experience with NPMLE of mixture models suggests that optimal jump points of NPMLE is not necessarily data points (with the exception of the one-dimensional Cosslett problem), hence restricting mass points at data points as in Jongbloed (2001) may also lead to some sort of efficiency loss?

We know that for a given grid size, there is a unique solution for $h(y)$ and hence $g(y)$ by maximizing the log-likelihood. One consideration might be when the grid size increases, the corresponding NPMLE for $h(y)$ may lead to over-fitting for the marginal density $f(z)$. By constructing some goodness-of-fit criteria for the density of the observed data z may provide some guidance for how to choose grid size. The log product spacing method considered in Roeder (1992) may become useful. See also Roeder (1990).

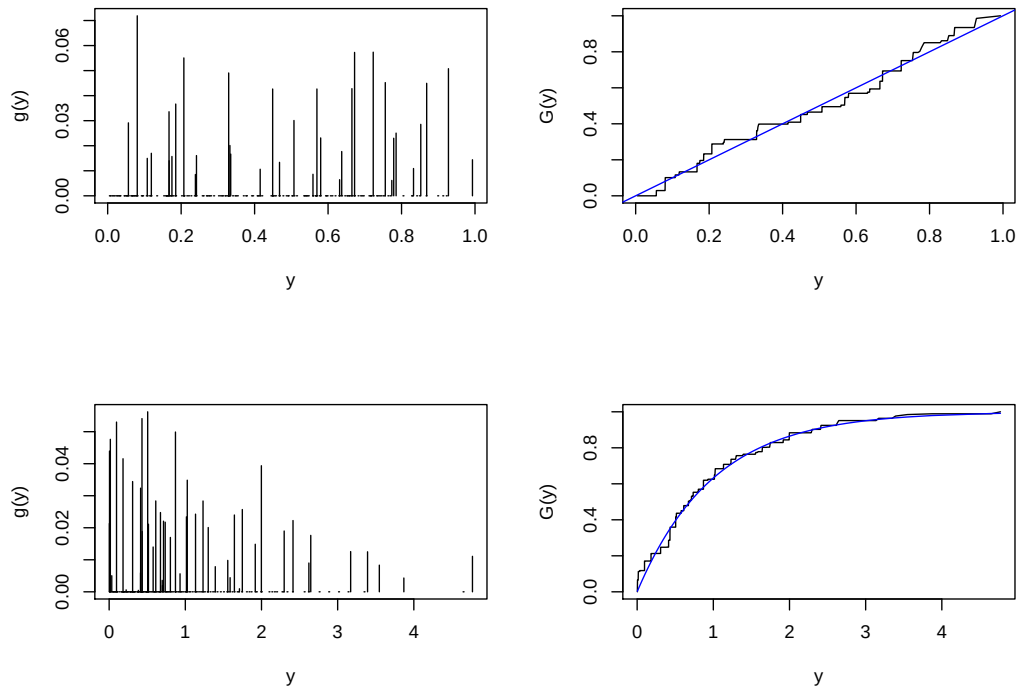


Fig. 12.2: NPMLEs for two Wickseil experiments

Implementation of Rényi Penalized Density Estimation

medder, v. used in Bahamian dialect, mostly on the Family Islands like Eleuthera and Cat Island meaning "mess with" "get involved," "get entangled," "fool around," "bother:" "I don't like to medder up with all kinda people" "Don't medder with people (chirren)" "Why you think she medderin up in their business."
[Urban Dictionary]

13.1 Introduction

Our medderin' about with maximum entropy de-regularized density estimation began with an exploration of total variation penalties for smoothing density estimates in Koenker and Mizera (2007). This led to subsequent work on shape constrained density estimation initially focused on log-concavity and eventually to weaker concavity penalties that required replacing the maximum likelihood objective by an alternative Rényi entropy criteria. From the beginning our implementations of these methods relied on the convex optimization software Mosek, initially within a Matlab interface, and more recently within the Rmosek interface to R. These notes were prepared mainly as an *aide memoire* for the transition of the software implementation of the function `medde` in the R package `REBayes` from Mosek V8 to V9. For additional details one can consult ApS (2018) and ApS (2019).

13.2 Theory

Koenker and Mizera (2010) began with the variational formulation of the log-concave MLE problem for given observations $X = \{X_1, \dots, X_n\}$, with $X_i \in \mathbb{R}^d$:

$$\min \left\{ \frac{1}{n} \sum_{i=1}^n g(X_i) + \int e^{-g(x)} dx \mid g \in \mathcal{K}(X) \right\}, \quad (P_1)$$

with $\mathcal{K}(X)$ denoting the set of closed convex functions on the convex hull, $\mathcal{H}(X)$, of X . A solution $\hat{g} : \mathcal{H}(X) \mapsto \mathbb{R}$ yields a density estimate $\hat{f}(x) = \exp(-\hat{g}(x))$ on $\mathcal{H}(X)$; the fact

that this obviously positive quantity is a probability density estimate, that is, its integral is equal to one, is assured by the presence of the integral term in (P_1) . Outside $\mathcal{H}(X)$, the solution $\hat{g}(x) = -\infty$, implying that $\hat{f}(x) = 0$. Interpreting (P_1) as a “primal” formulation in the context of convex programming, the associated “dual” problem is,

$$\max \left\{ \int -f \log f dx \mid f = \frac{d(\mathcal{Q}(X) - G)}{dx}, G \in \mathcal{K}(X)^o \right\}, \quad (D_1)$$

where $\mathcal{Q}(X) = n^{-1} \sum_{i=1}^n \delta_{X_i}$ is the empirical probability measure,

$$\mathcal{K}(X)^o = \left\{ G \in \mathcal{C}^*(X) \mid \int g dG \leq 0, g \in \mathcal{K}(X) \right\}$$

is the polar cone associated with $\mathcal{K}(X)$, and $\mathcal{C}^*(X)$ denotes the set of (signed) Radon measures on $\mathcal{H}(X)$. The appearance of the Shannon entropy in the dual formulation (D_1) may be interpreted as the intention to find $\hat{f}$ closest in Kullback-Leibler divergence to the uniform distribution on $\mathcal{H}(X)$ subject to the concavity constraint.

For the problem (P_1) , the solutions admit further characterization: $\hat{g}$ are piecewise linear on $\mathcal{H}(X)$, so estimated densities are piecewise exponential. This feature motivated a search for larger classes of quasi-concave densities that would accommodate heavier tails and more sharply peaked densities than the log concaves. Such classes are provided by s -concave functions. Loosely speaking, a function is called s -concave, for $s > 0$, if its s -th power is concave. More precisely, a non-negative, real function f , defined on a convex set $C \subset \mathbb{R}^d$ is s -concave, if there is a convex function g such that

$$f = \begin{cases} (-g)^{1/s} & \text{for } s > 0, \\ e^{-g} & \text{for } s = 0, \\ g^{1/s} & \text{for } s < 0. \end{cases}$$

Note that log-concave functions are 0-concave, and concave functions are 1-concave; also, if f is s -concave, then f is also s' -concave for any $s' < s$. The limiting class of $-\infty$ -concave, the union of all s -concave classes for all $s \in \mathbb{R}$, is the class of *quasi-concave* functions – functions with upper level sets convex. In the one-dimensional case, for $d = 1$, this class is identical with that of unimodal functions. In higher dimensions this equivalence no longer holds.

Once log-concavity is imposed, maximizing log likelihood in (P_1) appears to be especially convenient, as it leads to a convex program with the only nonlinearity arising from the integrability constraint. However, when weaker forms of concavity are considered, it proves more convenient to adapt the fitting criterion – in particular to retain the convexity of the optimization formulation. This was already apparent in an earlier work of Groeneboom, Jongbloed, and Wellner (2001) who employed least squares fitting rather than log-likelihood when imposing the stronger requirement of concavity of the density itself. While it is not really obvious how to adapt (P_1) to obtain a viable fitting formulation, the appearance of the Kullback-Leibler divergence in (D_1) suggests the possibility of replacing it by one of the abundant assortment of alternative divergences. Koenker and Mizera (2008, 2010) pointed out that for s -concave densities, this turns out to produce a lucky match. They

proposed replacing the Shannon entropy in (D_1) by a variationally equivalent form of the Rényi entropy, a move that yielded a family of new dual and primal pairings,

$$\max \left\{ \frac{1}{\alpha} \int f^\alpha(y) dy \mid f = \frac{d(\mathcal{Q}(X) - G)}{dy}, \quad G \in \mathcal{K}(X)^o \right\}, \quad (D_\alpha)$$

and

$$\min \left\{ \sum_{i=1}^n g(X_i) + \frac{|1-\alpha|}{\alpha} \int g^\beta dx \mid g \in \mathcal{K}(X) \right\}. \quad (P_\alpha)$$

The Rényi exponent α here corresponds to Avriel's $s = \alpha - 1$, and β is conjugate to α in the usual sense: $1/\alpha + 1/\beta = 1$.

Among the Rényi entropies, the ones enjoying particular connections to the existing literature are those with α being a multiple of $1/2$. Koenker and Mizera (2010) focused primarily on the log concave, $\alpha = 1$, case and the Hellinger, $\alpha = 1/2$, case; the latter imposes the weaker constraint that $-1/\sqrt{f}$ be concave. The implementation described here also allows us to venture into the netherworld of $\alpha \leq 0$. It should be noted that since $\alpha = 0.5$ already already subsumes tail behavior like Cauchy smaller α may be considered somewhat pathological. Sceptics, however, are encouraged to consider the examples in Koenker and Mizera (2019).

13.3 Praxis

Earlier implementations of these methods were developed in the standalone package **MeddeR**. However, a unification of the methods seemed desirable and has now been realized in a single function **medde** provided by the R package **REBayes** available on CRAN. These notes are intended as further documentation of this implementation for the convex optimization software Mosek. Because Mosek underwent a quite dramatic transition from Version 8 to 9, we treat both implementations here. We will begin by treating the shape constrained case and then turn to norm constraints. Some connections to the NPMLE methods for mixture models will be described briefly in a final section.

13.3.1 Mosek 8

The separable convex optimization **scopt** formalism of Mosek 8 allows additive objective functions with nonlinear components that can take the following types: “ent” $fx \log(x)$; “exp” fe^{gx+h} ; “log” $f \log(gx+h)$ and “pow” $f(x+h)^g$, where f, g, h are specified constants. For our dual formulation these terms appear only in the objective function not in the constraints, so they are represented in the Mosek formulation in the “opro” matrix of dimension 5 by p and type “list” with rows containing respectively: “type”, the index j of the coordinate of x , and the corresponding elements, f , g , and h . Thus, for example for the log-concave case with $\alpha = 1$ we would have the matrix:

$$\begin{bmatrix} \text{"ent"} & \text{"ent"} & \dots & \text{"ent"} \\ 1 & 2 & \dots & p \\ -1 & -1 & \dots & -1 \\ 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \end{bmatrix}.$$

Whereas for $\alpha = 0$ we have

$$\begin{bmatrix} \text{"log"} & \text{"log"} & \cdots & \text{"log"} \\ 1 & 2 & \cdots & p \\ 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ 0 & 0 & \cdots & 0 \end{bmatrix}.$$

and for other α we have

$$\begin{bmatrix} \text{"pow"} & \text{"pow"} & \cdots & \text{"pow"} \\ 1 & 2 & \cdots & p \\ -\text{sgn}(\beta) & -\text{sgn}(\beta) & \cdots & -\text{sgn}(\beta) \\ \alpha & \alpha & \cdots & \alpha \\ 0 & 0 & \cdots & 0 \end{bmatrix}.$$

where, as usual, $1/\alpha + 1/\beta = 1$. In contrast, the primal versions for $\alpha \in \{0, 1\}$ are

$$\begin{bmatrix} \text{"exp"} & \text{"exp"} & \cdots & \text{"exp"} \\ 1 & 2 & \cdots & p \\ 1 & 1 & \cdots & 1 \\ -1 & -1 & \cdots & -1 \\ 0 & 0 & \cdots & 0 \end{bmatrix},$$

and

$$\begin{bmatrix} \text{"log"} & \text{"log"} & \cdots & \text{"log"} \\ 1 & 2 & \cdots & p \\ -1 & -1 & \cdots & -1 \\ 1 & 1 & \cdots & 1 \\ 0 & 0 & \cdots & 0 \end{bmatrix}.$$

For other α the primal form is: [check this!!](#)

$$\begin{bmatrix} \text{"pow"} & \text{"pow"} & \cdots & \text{"pow"} \\ 1 & 2 & \cdots & p \\ \text{sgn}(\beta) & \text{sgn}(\beta) & \cdots & \text{sgn}(\beta) \\ \beta & \beta & \cdots & \beta \\ 0 & 0 & \cdots & 0 \end{bmatrix}.$$

13.3.2 Mosek 9

For Mosek 9 a major revision occurred that replaced the `scopt` formulation with an alternative scheme that allowed users greater flexibility, but required them to express nonlinear components of the objective function in terms of convex cone constraints. Now there are two essential types of cone constraints: exponential cones and power cones.

The canonical exponential cone is,

$$\mathcal{K}_e = \{x \in R^3 | x_1 \geq x_2 \exp(x_3/x_2), x_1, x_2 \geq 0\},$$

or equivalently,

$$\mathcal{K}_e = \{x \in R^3 | x_3 \leq x_2 \log(x_1/x_2), x_1, x_2 \geq 0\}.$$

Thus, if we introduce auxiliary variables, $t_1, \dots, t_p$, and require that,

$$\begin{pmatrix} e_i^\top & 0 \\ 0 & 0 \\ 0 & e_i^\top \end{pmatrix} \begin{pmatrix} x \\ t \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \in \mathcal{K}_e$$

we have effectively imposed that $\log(x_i) \geq t_i$, so if we can now replace our nonlinear objective function with a linear one in the auxiliary t variables. This is precisely what is required for our $\alpha = 0$ case. If instead, we require that

$$\begin{pmatrix} 0 & 0 \\ e_i^\top & 0 \\ 0 & e_i^\top \end{pmatrix} \begin{pmatrix} x \\ t \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \in \mathcal{K}_e$$

we have imposed that $t_i \leq x_i \log(1/x_i) = -x_i \log(x_i)$. This corresponds to our primal problem with $\alpha = 1$. For other settings of α we must rely on the Mosek implementation of power cones.

The canonical three variable power cone is

$$\mathcal{K}_\alpha = \{x \in R^3 | x_1^\alpha x_2^{1-\alpha} \geq |x_3|, x_1, x_2 \geq 0\}.$$

Such cones can be employed to formulate our problem for $\alpha \notin \{0, 1\}$. Three cases are considered separately. For $\alpha \in (0, 1)$ we can simply use:

$$\begin{pmatrix} e_i^\top & 0 \\ 0 & 0 \\ 0 & e_i^\top \end{pmatrix} \begin{pmatrix} x \\ t \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \in \mathcal{K}_\alpha$$

which implies that $x_i^\alpha \geq |t_i|$. For $\alpha > 1$ we simply flip the role of x and t and replace α by its reciprocal. And for $\alpha < 0$ we impose,

$$\begin{pmatrix} e_i^\top & 0 \\ 0 & e_i^\top \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ t \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \in \mathcal{K}_\alpha$$

which implies that $t_i \geq x_i^\alpha$. In all cases we scale the summands of the now linear objective by the factor, $-\text{sgn}(\beta)$

13.4 Odds and Ends

The foregoing discussion was intentionally focused on transition from nonlinear objective functions in Mosek 8 to the reformulation of in terms of cone constraints in which the parameter α plays a crucial role. This has been motivated primarily by the family of concavity shape constraints. It is worth noting the flexibility of the Rényi fitting criteria can also be applied to the norm constrained estimation of densities. The parameter `lambda` in the

`medde` function when it is negative imposes some form concavity restriction determined by the specification of `alpha`, when it is positive it controls the degree of smoothing imposed by the total variation penalty on some transformation of the density. When `alpha` is 1, this transformation is logarithmic.

Finally, it should be noted that `medde` also provides an opportunity to control the degree of the differential operator determining the constraints. Thus far, we have implicitly assumed that `Dorder` took the value 1, implying either a concavity constraint, or a norm constraint on the total variation of the first derivative of the fitted density. However, one can also set `Dorder` = 0, which imposes monotonicity on the estimated density when `lambda` is negative, or penalizes total variation of a transformation of the density when it is positive. When `Dorder` = 2 one can impose TV smoothing on the second derivative of the transformed density as illustrated in the `demo(Silverman)`. This effectively imposes an L_1 penalty on the third derivative of the log density.

The Group Lasso for Quantile Regression

Shrinkage is powerful, but a well-chosen basis expansion is sublime.

14.1 Introduction

In Koenker (2011a) I have tried to describe some ideas for estimation and inference about additive quantile regression models based on total variation penalization. This approach has several advantages, not the least of which is the efficient linear programming formulation for the computation. But a sticking point has always been choice of smoothing parameters. An alternative approach to additive models has been proposed in Kato (2011) based on a group lasso formulation in which the Euclidean norm of the coefficients of the additive components are penalized. An innovative aspect of this approach is the choice of penalty parameters which are selected by simulating the gradient condition of the QR model, yielding a automatic, data-dependent smoothing strategy. There is a close connection to prior work of Belloni and Chernozhukov (2011) for the QR lasso problem, but the explicit link to additive models and regression smoothing of the Kato approach made it particularly appealing for me. In this note I will briefly describe some computational experience with this approach; as usual there are some positive and negative aspects.

14.2 The Model

We will consider a simple QR model of the form,

$$Q_{Y|z}(\tau|z) = g(z),$$

where $\tau \in (0, 1)$ will be fixed, and the target function $g : \mathbb{R} \rightarrow \mathbb{R}$ will be assumed to be approximable, in some vague sense, by a B-spline expansion,

$$g(z) = \beta_0 + \sum_{j=1}^p \psi_j(z) \beta_j.$$

We will write the n by p design matrix, $X = (\psi_j(z_i))$, with rows, x_i , so we have a simple garden variety linear quantile regression model. Rather than assuming that we are clever enough to chose the basis expansion parsimoniously, we will rely instead on penalization to control the variability of the estimated parameters $\hat{\beta}$. It is presumably obvious that this setting can be expanded to incorporate more complicated additive models, but we will restrict attention to this simple one component model.

14.3 Computation

There are potentially many schemes for penalization, but the approach we will consider here is the Euclidean norm penalty of Kato, $P(\beta) = \|S\beta\|_2$, where $S = (n^{-1}X^\top X)^{1/2}$ is intended to normalize the coordinates of β . The intercept parameter β_0 is left unpenalized. The resulting estimator solves,

$$\min \sum_{i=1}^n \rho_\tau(Y_i - \beta_0 + x_i^\top \beta) + \lambda P(\beta),$$

which can be reformulated as the second order cone programming problem,

$$\max\{y^\top a \mid 1^\top a = 0, \|b\| \leq \lambda, Sb = X^\top a, a \in [\tau - 1, \tau]^n\}.$$

This dual formulation can be efficiently solved with modern interior point methods. In Appendix 14.6 we illustrate an implementation for R and the optimization suite Mosek, Andersen (2010).

But how to we select λ ? Kato, extending the proposal of Belloni and Chernozhukov (2011) for the QR lasso, suggested that λ should be chosen so that the probability of the event $\{\lambda \geq c\Lambda\}$ is close to one, where c is slightly larger than 1, and Λ can be approximated by an upper quantile of the random variable,

$$\tilde{\Lambda} = \left\| \sum_{i=1}^n (\tau - I(U_i \leq \tau)) (S^{-1/2} x_i / \sqrt{p}) \right\|_2,$$

where U_i are iid uniform $[0, 1]$. This quantile can be easily simulated and we will explore several choices of the quantile in the examples in the next section. In accordance with the suggestions in Kato (2011) we have chosen $c = 1.1$ for the implementation appearing in Appendix 14.6. The implementation is structured so that one can specify a choice of λ directly, or specify a quantile for the automated λ selection. We will employ both options below.

14.4 Computational Experience

Our test problem is taken from Polson and Scott (2016): z observations are taken as equally spaced on $[0, 1]$ and we contrast B-spline expansions with 10 and 50 basis functions. The target functions are given by,

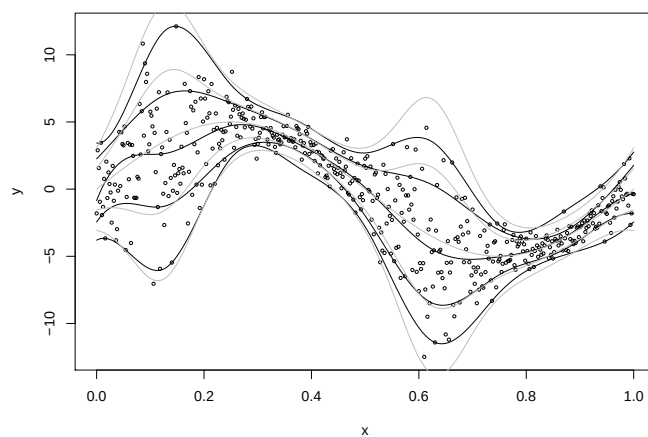


Fig. 14.1: Sample of 500 points from the Polson-Scott model, with true conditional quantile curves for $\tau \in \{.02, .15, .5, .85, .98\}$, in grey, and fitted curves in black based on automated λ selection with 10 B-spline basis functions.

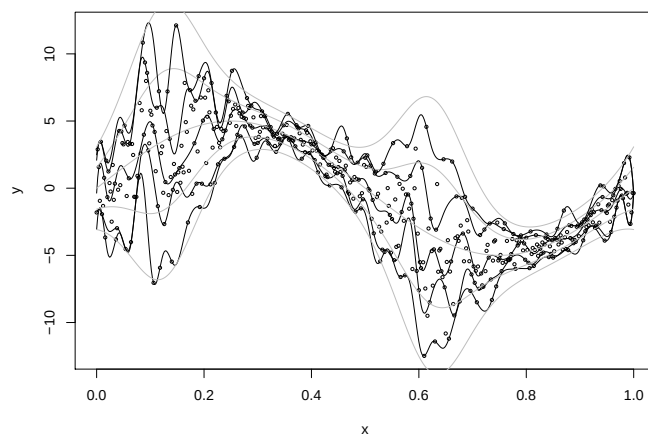


Fig. 14.2: Sample of 500 points from the Polson-Scott model, with true conditional quantile curves for $\tau \in \{.02, .15, .5, .85, .98\}$, in grey, and fitted curves in black based on automated λ selection with 50 B-spline basis functions.

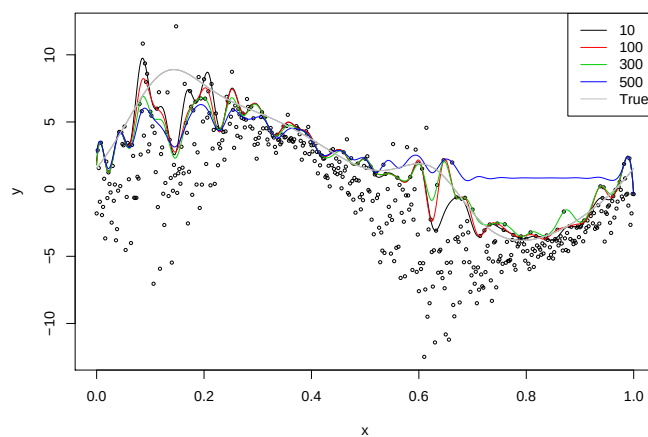


Fig. 14.3: Sample of 500 points from the Polson-Scott model, with true conditional quantile curve for $\tau = .85$, in grey, and fitted curves in various colors for several λ 's with 50 B-spline basis functions.

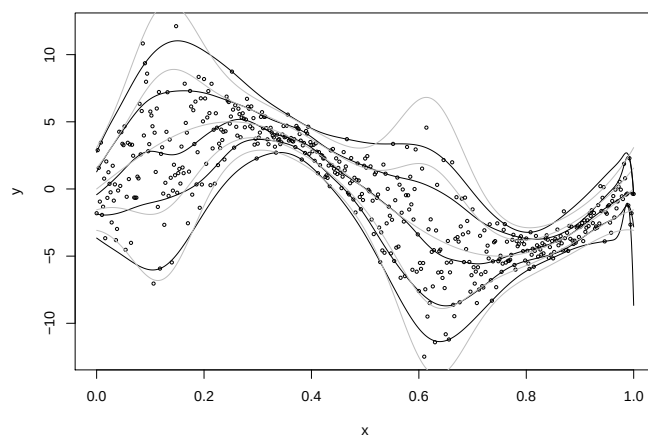


Fig. 14.4: Sample of 500 points from the Polson-Scott model, with true conditional quantile curves for $\tau \in \{.02, .15, .5, .85, .98\}$, in grey, and fitted curves in black based on semi-automated λ selection with 50 B-spline basis functions and second-differenced Euclidean norm penalty.

$$Q_{Y|x}(\tau|x) = 5 * \sin(2 * \pi * x) + (0.5 + \exp(1.5 * \sin(4 * \pi * x)))\Phi^{-1}(\tau)$$

The target functions are all quite smooth, but the model is quite heteroscedastic so the conditional quantile functions are quite distinctive.

We begin by illustrating performance of the estimator when the basis expansion is relatively parsimonious so there are only 10 basis function. Figure 14.1 a sample of 500 observations from this model. We have super-imposed the true conditional quantile curves in grey for $\tau \in \{.02, .15, .5, .85, .98\}$, and estimated conditional quantile curves in black with λ chosen by the automated procedure described above with the default (Kato) choice of $\theta = 0.1$ yielding the 0.9 quantile of the simulated distribution with 500 realizations of the Bernoulli vector. One can always be suspicious about simulations of size one so diligent readers are encouraged to repeat this exercise with altered seeds, but I believe that it is fair to say that this performance is encouraging.

Encouraged, we now repeat the exercise with 50 basis functions rather than only 10. Same setup, same seed, but now the fit in Figure 14.2 is horribly undersmoothed. Is this the fault of our automated λ selection? Let's see, focusing on $\tau = 0.85$ we can try to increase λ "gradually" and see what happens. In Figure 14.3 we have turned off the automated λ selection and tried exploring with the well-known, but much maligned "eyeball method." Our automated selection was $\hat{\lambda} = 9.97$, so we started by turning up the knob to $\lambda = 100$. Surprisingly, this made almost no visible difference, still the estimate was drastically undersmoothed. So what about $\lambda = 300$? Same story. At $\lambda = 500$ we do get something quite different, but even worse. Instead of producing a nice smooth curve throughout the relevant domain, we see the same wiggles for small x , and then there is a range of x from about 6 to 9.5 in which the estimate flattens out and as a result completely misses the dip in the target function. So apparently there is no λ that can successfully capture the slowly undulating structure of our test problem when the B-spline basis expansion is so expansive.

What went wrong? When p was 10 everything looked splendid. If we think back to our first lesson in B-splines it will be recalled that their main virtue is that they are locally supported, so with $p = 50$ have lots of basis functions all with small support, and this seems to inevitably lead to a highly oscillatory fit. When there are fewer basis functions we can expect to get something much smoother, and we do. Unfortunately, the penalization applied here isn't clever enough to find a good linear combination of 50 basis functions that would produce reasonable fit. What we would need is a penalty that understood that adjacent basis functions should have coefficients that didn't differ too much. This is precisely the feature of the familiar L_2 smoothing spline penalty, which boils down to shrinking *second differences* in adjacent spline coefficients. This can be adapted quite easily to the QR setting as already suggested in much earlier work of Bosch, Ye, and Woodworth (1995) and more recently by Koenker and Mizera (2013) and Fasiolo, Goude, Nedellec, and Wood (2017).

Indeed, it is relatively easy to adapt the Euclidean penalty described above so that it penalized second differences simply by modifying the matrix S . In Appendix 14.6 we also include a modified version of the original code that replaces the scaled Euclidean penalty of Kato (2011) with a version that penalizes the Euclidean norm of the second differences of the B-spline coefficients. Unfortunately, automatic λ selection still produces a severely undersmoothed fit, but when λ is dialed up to produce a smoother fit we do get the expect-

edly smoother fit. How to automatically select λ in this formulation is still an open question however.

14.5 Conclusion

Admittedly, the foregoing examples don't really *prove* anything, but they offer *prima facie* evidence that the Euclidean penalty, unless paired with a carefully selected basis expansion, is difficult to tune automatically. In contrast, the total variation smoothing penalty, because it is tied more closely to the data points via the linear programming structure, seems better capable of accomplishing this.

14.6 R Coda

```
# Mosek implementation of (dual) group lasso method for QR a la Kato (2011)
glasso = function(X, y, G, lambda = NULL, tau = 0.5, theta = NULL, rtol = 1e-6, verb = 0, ...){
  # G is a list of group subsets of the columns of X.
  # By convention G[[1]] contains un-penalized columns
  # No sanity check for G (yet!)
  q = length(G)
  p = NCOL(X)
  n = NROW(X)
  Pk = 0
  lams = rep(NA, q)
  P = list(sense = "max")
  P$c = c(y, rep(0, p+2))
  P$bc = matrix(0,2,p)
  P$cones = matrix(list(),2,q)
  P$cones[1,1:q] = "QUAD"
  P$A = Matrix(0, p, p)
  for(k in 1:q){
    Xk = X[,G[[k]], drop = FALSE]
    pk = NCOL(Xk)
    Sk = chol(crossprod(Xk)/n)/sqrt(pk)
    lams[k] = lamhat(Xk, tau = tau, theta = theta)
    P$A[(Pk + 1):(Pk + pk), (Pk + 1):(Pk + pk)] = Sk
    P$cones[2,k] = list(c(n + p + k, (n + Pk + 1):(n + Pk + pk)))
    Pk = Pk + pk
  }
  if(!length(lambda)) lambda = max(lams[-1])
  P$bx = rbind(c(rep(tau-1,n),rep(-Inf,p),0,lambda),c(rep(tau,n),rep(Inf,p), 0, lambda))
  P$A = cbind(t(X), -P$A,0,0)
  P$dparam$intpnt_co_tol_rel_gap <- rtol
  z <- mosek(P, opts = list(verbose = verb))
  status <- z$sol$itr$solsta
  coef = z$sol$itr$slc - z$sol$itr$suc
  resid = y - X %*% coef
```

```

    list(coef = coef, resid = resid, lambda = lambda, status = status)
  }
lamhat = function(X, tau, R = 500, c = 1.1, theta = 0.1){
  n = NROW(X)
  p = NCOL(X)
  S = backsolve(chol(crossprod(X)/n), diag(p)) %*% t(X)/sqrt(p)
  B = tau - (matrix(runif(n * R), n, R) < tau)
  v = apply((S %*% B)^2, 2, sum)
  c * quantile(sqrt(v), 1 - theta)
}

# fig code for the glasso vinaigrette
require(Rmosek)
require(splines)
source("glasso.R")
set.seed(1848)
# Polson-Scott Model
dgp <- function(n) {
  x <- 0:n/n
  y <- rnorm(n+1, 5 * sin(2 * pi * x), 0.5 + exp(1.5 * sin(4 * pi * x)))
  data.frame(x = x, y = y)
}
M = dgp(500)
x = M$x
y = M$y
X = cbind(1,bs(x, df = 10))
taus = c(.02, .15, .5, .85, .98)
pdf(file = "fig1.pdf", height = 6, width = 8)
plot(x,y,cex = .5)
for(i in 1:length(taus)){
  F = glasso(X,y, G = list(1,2:NCOL(X)), tau = taus[i], theta = .1)
  yF = X %*% F$coef
  lines(x, yF)
  lines(x, 5 * sin(2 * pi * x) + (0.5 + exp(1.5 * sin(4 * pi * x))) * qnorm(taus[i]), col = "grey")
}
dev.off()
X = cbind(1,bs(x, df = 50))
taus = c(.02, .15, .5, .85, .98)
pdf(file = "fig2.pdf", height = 6, width = 8)
plot(x,y,cex = .5)
for(i in 1:length(taus)){
  F = glasso(X,y, G = list(1,2:NCOL(X)), tau = taus[i], theta = .1)
  yF = X %*% F$coef
  lines(x, yF)
  lines(x, 5 * sin(2 * pi * x) + (0.5 + exp(1.5 * sin(4 * pi * x))) * qnorm(taus[i]), col = "grey")
}
dev.off()
X = cbind(1,bs(x, df = 50))
lambdas = c(10, 100, 300, 500)

```

```
pdf(file = "fig3.pdf", height = 6, width = 8)
plot(x,y,cex = .5)
for(i in 1:length(lambdas)){
  F = glasso(X,y, G = list(1,2:NCOL(X)), tau = 0.85, lambda = lambdas[i], theta = .1)
  lambdas[i] = F$lambda
  yF = X %*% F$coef
  lines(x, yF, col = i)
  lines(x, 5 * sin(2 * pi * x) + (0.5 + exp(1.5 * sin(4 * pi * x))) * qnorm(0.85), col = "grey")
}
legend("topright", c(expression(lambda = 10), expression(lambda = 100), expression(lambda = 300),
  expression(lambda = 500), expression(True)), col = c(1:4,"grey"), lty = 1)
dev.off()
```

```
# NB. This version replaces Kato's standardization with smoothing penalty.
glasso = function(X, y, G, lambda = 1, tau = 0.5, rtol = 1e-6, verb = 0, ...){
# G is a list of group subsets of the columns of X.
# By convention G[[1]] contains un-penalized columns
# No sanity check for G (yet!)
q = length(G)
p = NCOL(X)
n = NROW(X)
Pk = 0
P = list(sense = "max")
P$c = c(y, rep(0, p+2))
P$bc = matrix(0,2,p)
P$cones = matrix(list(),2,q)
P$cones[1,1:q] = "QUAD"
P$A = Matrix(0, p, p)
for(k in 1:q){
  Xk = X[,G[[k]], drop = FALSE]
  pk = NCOL(Xk)
  if(k == 1){
    Sk = diag(pk)
    P$A[(Pk + 1):(Pk + pk),(Pk + 1):(Pk + pk)] = Sk
  }
  else {
    Sk = diff(diff(diag(pk)))
    P$A[(Pk + 1):(Pk + pk - 2),(Pk + 1):(Pk + pk)] = Sk
  }
  P$cones[2,k] = list(c(n + p + k, (n + Pk + 1):(n + Pk + pk)))
  Pk = Pk + pk
}
P$bx = rbind(c(rep(tau-1,n),rep(-Inf,p),0,lambda),c(rep(tau,n),rep(Inf,p), 0, lambda))
P$A = cbind(t(X), -P$A,0,0)
P$dparam$intpnt_co_tol_rel_gap <- rtol
z <- mosek(P, opts = list(verbose = verb))
status <- z$sol$itr$solsta
```

```

coef = z$sol$itr$slc - z$sol$itr$suc
resid = y - X %*% coef
list(coef = coef, resid = resid, status = status)
}

# New fig2 with smoother penalty
require(Rmosek)
require(splines)
source("glassos.R")
set.seed(1848)
# Polson-Scott Model
dgp <- function(n) {
  x <- 0:n/n
  y <- rnorm(n+1, 5 * sin(2 * pi * x), 0.5 + exp(1.5 * sin(4 * pi * x)))
  data.frame(x = x, y = y)
}
M = dgp(500)
x = M$x
y = M$y
X = cbind(1,bs(x, df = 50))
taus = c(.02, .15, .5, .85, .98)
pdf(file = "fig4.pdf", height = 6, width = 8)
plot(x,y,cex = .5)
for(i in 1:length(taus)){
  F = glasso(X,y, G = list(1,2:NCOL(X)), tau = taus[i], lambda = 6, verb = 5)
  yF = X %*% F$coef
  lines(x, yF)
  lines(x, 5 * sin(2 * pi * x) + (0.5 + exp(1.5 * sin(4 * pi * x))) * qnorm(taus[i]), col = "grey")
}
dev.off()

```


Bivariate Exchangeability

15.1 Introduction

A couple of weeks ago googling around for additional insight into exchangeability I ran across Persi Diaconis's review in the AMS Bulletin of Kallenberg's Diaconis (2009) *Probabilistic symmetries and invariance principles*. The review is exemplary in many respects, but among these it provides a quite readable introduction to bivariate exchangeability and some connections to the literature on visual perception in the work of Bela Julesz at Bell Labs in the 1970s. The latter connections are spelled out in more detail in Diaconis and Freedman (1981).

The formalism of all this is confined to “coin tossing” where in the simplest setting of Bernoulli sequences we have de Finetti's theorem that tells us that any exchangeable probability measure P can be expressed as a mixture,

$$P = \int_0^1 P_\theta \mu(d\theta),$$

where P_θ is Bernoulli with parameter θ and μ is a unique probability measure on $[0, 1]$. For arrays this generalizes nicely, but there are some unexpected subtleties when invariance is restricted to only rows and columns. Diaconis considers matrices with 0-1 entries, x_{ij} such that,

$$P(x_{ij} : 1 \leq i, j \leq n) = P(x_{\sigma(i)\tau(j)} : 1 \leq i, j \leq n)$$

for permutations, σ and τ and all n . In the univariate setting we can imagine constructing sequences in the following way: first, a θ is drawn from the distribution μ , then a coin is flipped which lands heads with probability θ , if heads is observed our entry is 1, otherwise 0. In the bivariate setting we consider an arbitrary Borel function $\varphi : [0, 1]^2 \rightarrow [0, 1]$. We generate a pair of independent uniforms (U_i, V_j) on $[0, 1]$, and now flip a $\varphi(U_i, V_j)$ coin entering a 1 if it lands heads, 0 if tails. It transpires that any row/column exchangeable P is a mixture of such φ -processes,

$$P = \int P_\varphi \mu(d\varphi),$$

so these P_φ are extreme points from a Choquet perspective. Apparently, uniqueness of μ for this variant turned out to be quite difficult.

15.2 Perception of Randomness

Suppose we generate a 0-1 matrix and plot it, can we tell if it has some structure that distinguishes it from merely choosing each entry with a flip of a fair coin? This was the subject of one line of Julez's research. He conjectured that if the process generating entries had the same first and second order probabilities, that is if:

1. the chance of a 1 appearing in any given position was $1/2$, and
2. the chance of two 1's appearing in any two positions was $1/4$,

then the matrix would be indistinguishable from the pure coin tossing realization. Diaconis and Freedman (1981) produced a family of counterexamples that were quite simple, yet easily distinguishable. The simplest of these examples is illustrated above and can be constructed in the following way: generate the first row as a fair Bernoulli sequence, then again flip fair coin if it falls heads repeat the first row as the second row, if tails make the second row $x_{2i} = (x_{1i} + 1) \bmod(2)$ $i = 1, 2, \dots, n$, and continue in this fashion to fill the remaining rows. As can be seen the texture of the resulting figure is quite "unrandom" looking, although it has the same first, second (and third order) probabilities!

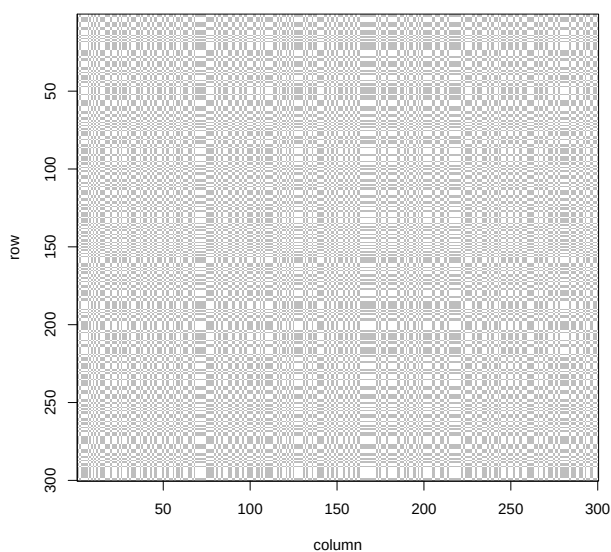


Fig. 15.1: Diaconis Counterexample to the Julez Conjecture

15.3 Conclusion

It seems that φ -processes of the sort employed here are closely related to copula functions. There might be a temptation to think that they are also related to rank 1 matrices, since they depend upon only two Boolean n -vectors, but they are not. Extending their ideas to infinite domains offers some further challenges that bring Voronoi tessellations into the picture.

Empirical Bayes Confidence Intervals

1

16.1 Introduction

Renewed interest in Stein shrinkage and empirical Bayes methods more generally has prompted new work on confidence intervals for empirical Bayes estimates. Bruce Hansen’s cogent discussion of Armstrong, Kolesár, and Plagborg-Møller (2020) in a recent Chamberlain Seminar aroused my curiosity about performance of posterior empirical Bayes intervals based on the nonparametric maximum likelihood methods of Robbins (1950) and Kiefer and Wolfowitz (1956). This note describes some further simulation evidence based on the implementation of this approach in Koenker (2013), and closely related methods of Efron (2019). While the “minimalist” approach based on posterior percentile intervals constructed from the NPMLE of Kiefer-Wolfowitz perform poorly, the smoother G modeling approach of Efron is found to possess good “robustness of validity” and “robustness of efficiency.”

Consider a family of mixture models of the form,

$$f(y) = \int \varphi(y|\theta) dG(\theta),$$

where φ denotes a known parametric “base” model and G denotes an unknown, nonparametric mixing distribution. Such models are fundamental in empirical Bayes compound decision settings where we have the (exchangeable) hierarchical structure,

$$Y_i \sim \varphi(y|\theta_i); \quad \theta_i \sim G.$$

When θ is a location parameter, so $\varphi(y|\theta_i) = \varphi(y - \theta_i)$ this is a conventional deconvolution problem usually evoking characteristic function methods, however recent work has emphasized the applicability of maximum likelihood methods. An authoritative overview of this approach is provided in Efron (2019) and the discussion thereof.

¹ Thanks to Bruce Hansen for inciting my interest in this topic and sharing his simulation code. Thanks too to Jiaying Gu for helpful comments and continuing collaboration on empirical Bayes methods.

When the mixing distribution, G , is Gaussian, familiar Bayesian computations yield the James-Stein shrinkage formulas for (compound) squared error loss. This is the parametric empirical Bayes setting explored in Armstrong, Kolesár, and Plagborg-Møller (2020). In this note I want to very briefly explore two nonparametric approaches for confidence interval construction. We will see that the minimalist NPMLE methods of Kiefer and Wolfowitz while well suited to point estimation are not so well adapted to interval estimation. This conclusion complements the recent simulation findings and discussion of Jiang (2019), who considers several smoothed versions of the NPMLE.

16.2 The Kiefer-Wolfowitz NPMLE

In Koenker and Mizera (2013) we have advocated the Kiefer-Wolfowitz NPMLE approach to estimating G and constructing estimates of the θ_i 's for compound decision problems. In sharp contrast to finite dimensional mixture problems with highly multimodel likelihoods, discrete formulations of the general nonparametric mixture problem are strictly convex and therefore admit unique solutions. Consider a grid $t_0, t_1, \dots, t_m$ with associated masses $\{g \in \mathbb{R}^m | g_i \geq 0, \sum_{i=1}^m g_i \Delta t_i = 1\}$, we can approximate the log likelihood by,

$$\ell(G) = \sum_{i=1}^n \log f_i$$

where the n vector $f = Ag$ and A is the n by m matrix with typical element $\varphi(y_i, t_j)$. As is well known from Laird (1978) or Lindsay (1983) the NPMLE, $\hat{G}$, has $p \leq n$ positive mass points, while in practice this p is usually closer to $\log n$ than n . Interior point methods for solving such problems are considerably more efficient than earlier EM approaches greatly facilitating the study of their performance in simulation experiments. Unfortunately, little is known about their statistical efficiency from a theoretical perspective beyond the basic consistency results of Kiefer and Wolfowitz (1956) and Pfanzagl (1988).

Given an estimator for G , the usual Bayesian machinery can be invoked to produce point estimates with respect to various loss functions. In particular, we have Tweedie's formula, see e.g. Efron (2011), for φ standard Gaussian,

$$\hat{\theta}_i = y_i + f'(y_i)/f(y_i)$$

as a (nonlinear) shrinkage formula that can be implemented by plugging in $\hat{G}$ for G . Had we known G this is just the posterior mean, the Bayes rule with respect to (compound) quadratic loss. Other loss functions produce other point estimates, so quantile loss yields posterior quantiles. This raises the natural question: are these posterior quantile estimates good for anything. In particular, could they be used to construct confidence intervals for the θ_i 's? Before putting this to the test, let me briefly describe an alternative approach due to Efron.

16.3 Efron's NPMLE

Efron (2016) has proposed an alternative approach to estimating G that expresses its log derivative by a regression spline,

$$g(y|\theta) = \exp\left\{\sum_{j=1}^p \theta_j \psi_j(y) - \psi_0(\theta)\right\},$$

as in the pure density estimation methods of Stone (1990) and Barron and Shue (1991). We can maintain the same discretization for the support of G , and set,

$$g = (g_j) = (g(t_j|\theta)),$$

so the log likelihood can be expressed as above, except that now we are estimating a finite dimensional parameter θ of predetermined dimension. Efron suggests natural splines for the ψ_j functions and the penalization,

$$\ell_n(G_\theta) + \lambda \|\theta\|$$

by the Euclidean norm of the vector θ , thereby shrinking θ toward the origin and $\hat{G}$ toward the uniform distribution. A virtue of the Efron approach over the Kiefer-Wolfowitz form of the NPMLE is that it produces smooth $\hat{G}$'s that are better suited to inference. Their downside, as we have stressed in our contributed discussion, Koenker and Geling (2001), is the need to choose tuning parameters for the dimension of the basis expansion and the penalty parameter λ .

A striking feature of both the Efron and Kiefer-Wolfowitz proposals is that neither depend upon the mixture model being a formal convolution. Of course when θ is a location parameter so $\varphi(y|\theta) = \varphi(y - \theta)$ then classical deconvolution methods are also applicable. Efron compares the performance of his procedure with the kernel deconvolution method of Stefanski and Carroll (1990), and concludes that the latter is “too variable in the tails.” This is confirmed in the comparisons reported in our contributed discussion of Efron (2019).

16.4 Some Simulation Evidence

We consider three simulation settings: the two proposed by Hansen, and one bonus setting proposed by an astute reader of the first draft who shall remain anonymous. All settings are special cases of the standard Gaussian sequence model,

$$Y_i = \theta_i + u_i,$$

with $u_i \sim \mathcal{N}(0, 1)$ and $\theta_i \sim G$. In the first setting G is also Gaussian with mean 0, and constant variance, V . This is obviously a favorable setting for the linear shrinkage Stein rule. There is almost no “signal” we are just observing Gaussians with variance $1 + V$, when V is large there will be a few observations that are sufficient “unusual” that they would be unlikely to come from the standard Gaussian, but especially when V is small this setting

is quite challenging for any NPMLE. The second setting is can be viewed as more typical of genomic Gaussian sequence models: $G = 0.90\delta_0 + 0.10\delta_a$, where δ_x is the usual Dirac δ -function with mass 1 at the point x . In the former setting there are four values of V that are considered, $\{0.1, 0.5, 1, 2\}$, while in the latter setting $a = \sqrt{10V}$ for one element of same set of V 's. The procedure proposed by Armstrong, Kolesár, and Plagborg-Møller (2020) is now included in the simulation comparisons as implemented in their R package “ebci”.

In our third bonus setting G is gamma with shape parameter $6\sqrt{V}$ and rate parameter 3. The corresponding densities for these G distributions are depicted in Figure 16.1.

In Table 16.1 I report observed coverage for posterior $[0.025, 0.975]$ intervals based on the Kiefer-Wolfowitz NPMLE on the left and the Efron NPMLE as well as the AKP-M procedure. For the Efron intervals I'm using the default 5 degrees of freedom natural spline basis expansion and $\lambda = 0.1$. It is painfully apparent that the using the Kiefer-Wolfowitz $\hat{G}$ produces severe under coverage in this setting, while the default Efron procedure is quite reliable. To explore this a bit further we report mean length of the intervals in Table 16.2, and root mean squared error of the posterior means in Table 16.3. It is evident from these tables that the under coverage of the KW intervals is due to their length; the KW posterior mean estimates are slightly more accurate than the point estimates delivered by the Efron posterior means. The discrete nature of the KW estimate, $\hat{G}$ inevitably produces intervals that are sensitive to the mass points of the $\hat{G}$, and can result in wildly optimistic intervals. The Armstrong et al (AKP-M) intervals have good coverage except for the $V = 0.1$ case where they are seriously uncovering. In contrast the Efron intervals are quite reliable throughout and at the same time considerably shorter than the AKP-M intervals except in the situations in which the latter uncover.

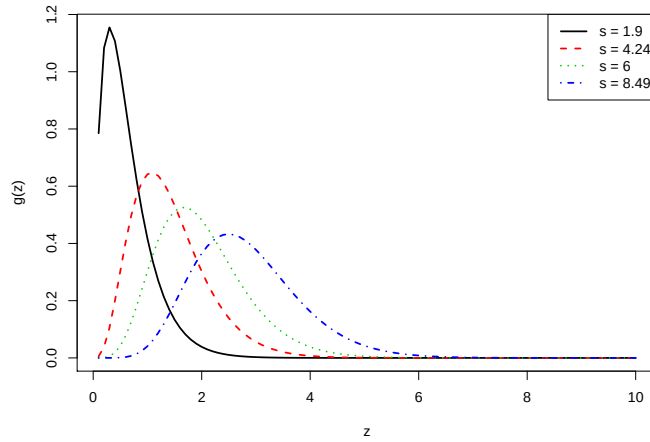


Fig. 16.1: Four gamma densities for the third (bonus) simulation setting.

	Kiefer-Wolfowitz				Efron				AKP-M			
	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000
V = 0.1	0.411	0.433	0.490	0.554	0.954	0.956	0.952	0.951	0.668	0.714	0.800	0.871
V = 0.5	0.611	0.663	0.721	0.750	0.912	0.926	0.912	0.908	0.947	0.965	0.974	0.975
V = 1	0.655	0.710	0.758	0.789	0.924	0.935	0.928	0.904	0.972	0.977	0.978	0.978
V = 2	0.672	0.721	0.774	0.799	0.929	0.938	0.941	0.925	0.975	0.978	0.978	0.978

Table 16.1: Simulation Setting 1: Observed coverage proportion in 1000 trials for intended nominal coverage 0.95

	Kiefer-Wolfowitz				Efron				AKP-M			
	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000
V = 0.1	0.645	0.640	0.680	0.734	1.481	1.387	1.305	1.275	1.095	1.018	1.045	1.097
V = 0.5	1.392	1.490	1.612	1.670	2.186	2.208	2.167	2.155	2.528	2.569	2.618	2.613
V = 1	1.810	1.952	2.086	2.172	2.667	2.715	2.694	2.629	3.208	3.240	3.253	3.257
V = 2	2.156	2.301	2.463	2.542	3.084	3.130	3.158	3.113	3.720	3.730	3.735	3.738

Table 16.2: Simulation Setting 1: Observed interval length in 1000 trials for intended nominal coverage 0.95

In Tables 16.4 to 16.6 we report parallel results for (outlier) Setting 2 of the simulation experiment. The Kiefer-Wolfowitz intervals are again consistently too short under covering the true parameters. The AKP-M intervals coverage is again quite good except when the outliers are relatively close to the null mass point at $\theta = 0$. However, the lengths of the AKP-M intervals are considerably wider than the corresponding Efron intervals despite the fact that the Efron intervals tend to be unnecessarily conservative. KW posterior means are slightly more accurate than the Efron point estimates as in Setting 1 and they are considerably more accurate than the point estimates delivered by linear shrinkage underlying the AKP-M procedure.

In Tables 16.7 to 16.9 we report parallel results for Setting 3 of the simulation experiment. The KW intervals are still poor in terms of coverage but now both the Efron and AKP-M intervals perform well in terms of coverage. But the AKP-M intervals are much too wide. The Efron smoothing of the MLE performs well in this asymmetric setting producing somewhat better RMSE performance than the Kiefer-Wolfowitz estimator.

	Kiefer-Wolfowitz				Efron				AKP-M			
	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000
V = 0.1	0.130	0.112	0.102	0.097	0.346	0.325	0.313	0.308	0.288	0.236	0.159	0.109
V = 0.5	0.385	0.362	0.349	0.343	0.608	0.594	0.592	0.589	0.347	0.339	0.337	0.335
V = 1	0.564	0.539	0.522	0.512	0.734	0.722	0.721	0.727	0.511	0.504	0.504	0.502
V = 2	0.751	0.715	0.691	0.682	0.843	0.828	0.824	0.835	0.682	0.668	0.667	0.668

Table 16.3: Simulation Setting 1: Observed root mean squared error in 1000 trials of posterior mean

	Kiefer-Wolfowitz				Efron				AKP-M			
	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000
V = 0.1	0.418	0.472	0.544	0.598	0.938	0.932	0.924	0.921	0.701	0.758	0.848	0.888
V = 0.5	0.459	0.485	0.539	0.521	0.967	0.973	0.978	0.981	0.930	0.933	0.930	0.930
V = 1	0.392	0.367	0.426	0.452	0.982	0.987	0.990	0.988	0.949	0.950	0.951	0.953
V = 2	0.340	0.338	0.371	0.381	0.992	0.995	0.998	0.997	0.964	0.964	0.967	0.967

Table 16.4: Simulation Setting 2: Observed coverage proportion in 1000 trials for intended nominal coverage 0.95

	Kiefer-Wolfowitz				Efron				AKP-M			
	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000
V = 0.1	0.579	0.607	0.649	0.668	1.444	1.373	1.286	1.256	1.067	1.052	1.051	1.108
V = 0.5	1.036	1.058	1.069	1.032	1.943	1.914	1.849	1.839	2.517	2.581	2.605	2.618
V = 1	0.913	0.849	0.828	0.803	2.022	1.938	1.836	1.795	3.229	3.243	3.248	3.260
V = 2	0.564	0.521	0.480	0.433	1.881	1.749	1.607	1.522	3.734	3.732	3.739	3.740

Table 16.5: Simulation Setting 2: Observed interval length in 1000 trials for intended nominal coverage 0.95

	Kiefer-Wolfowitz				Efron				AKP-M			
	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000
V = 0.1	0.118	0.102	0.091	0.087	0.332	0.315	0.299	0.294	0.288	0.238	0.149	0.110
V = 0.5	0.285	0.261	0.242	0.236	0.534	0.517	0.504	0.498	0.348	0.341	0.336	0.335
V = 1	0.281	0.249	0.235	0.228	0.530	0.511	0.499	0.495	0.511	0.507	0.503	0.501
V = 2	0.159	0.135	0.118	0.115	0.404	0.377	0.360	0.355	0.677	0.672	0.667	0.669

Table 16.6: Simulation Setting 2: Observed root mean squared error in 1000 trials of posterior mean

	KW				KWs				Efron				AKP-M			
	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000
V = 0.1	0.467	0.520	0.591	0.625	0.922	0.952	0.967	0.971	0.933	0.941	0.938	0.938	0.954	0.970	0.974	0.976
V = 0.5	0.581	0.620	0.689	0.716	0.926	0.948	0.962	0.967	0.913	0.927	0.919	0.934	0.977	0.979	0.978	0.979
V = 1	0.616	0.670	0.720	0.749	0.934	0.951	0.961	0.964	0.921	0.932	0.920	0.932	0.976	0.976	0.976	0.976
V = 2	0.642	0.694	0.746	0.772	0.936	0.952	0.960	0.963	0.922	0.936	0.928	0.932	0.973	0.972	0.972	0.972

Table 16.7: Simulation Setting 3: Observed coverage proportion in 1000 trials for intended nominal coverage 0.95

	KW				KWs				Efron				AKP-M			
	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000
V = 0.1	0.858	0.928	1.023	1.077	1.762	1.850	1.915	1.928	1.667	1.633	1.585	1.582	2.725	2.777	2.802	2.810
V = 0.5	1.309	1.385	1.514	1.557	2.256	2.337	2.409	2.439	2.107	2.111	2.078	2.127	3.839	3.842	3.849	3.851
V = 1	1.518	1.640	1.747	1.808	2.515	2.610	2.666	2.679	2.355	2.385	2.345	2.376	4.076	4.078	4.079	4.079
V = 2	1.733	1.858	1.991	2.049	2.763	2.856	2.918	2.925	2.594	2.639	2.622	2.628	4.176	4.177	4.177	4.177

Table 16.8: Simulation Setting 3: Observed interval length in 1000 trials for intended nominal coverage 0.95

	KW				KWs				Efron				AKP-M			
	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000	n=100	n=200	n=500	n=1000
V = 0.1	0.212	0.194	0.182	0.177	0.215	0.200	0.190	0.186	0.202	0.188	0.179	0.176	0.393	0.386	0.382	0.380
V = 0.5	0.360	0.340	0.324	0.320	0.358	0.344	0.333	0.332	0.348	0.332	0.326	0.320	0.721	0.715	0.712	0.714
V = 1	0.438	0.422	0.406	0.397	0.433	0.424	0.417	0.410	0.424	0.412	0.410	0.400	0.827	0.826	0.825	0.823
V = 2	0.532	0.510	0.493	0.483	0.520	0.510	0.505	0.498	0.513	0.497	0.496	0.488	0.901	0.906	0.907	0.900

Table 16.9: Simulation Setting 3: Observed root mean squared error in 1000 trials of posterior mean

16.5 Conclusion

Viewed in retrospect it is hardly surprising that the Kiefer-Wolfowitz posterior intervals perform poorly. Their discrete character of the estimated posterior makes it almost impossible for them to adapt to a desired α level. What is more surprising about the foregoing exercise is the excellent performance of the Efron intervals. It might have been expected that since they do not account for variability in $\hat{G}$, they too might under cover, but they are actually a little too conservative in all three settings of the experiment. It is tempting to suggest that the performance of the KW intervals could be ameliorated by some judicious smoothing of the KW $\hat{G}$. But this would draw us back into the contested territory of tuning parameter selection and our four page limit has already been exceeded. Like his son's predictions about the length of the Iraq war, let's just concede that Jack Wolfowitz's intervals are just too short. The main take away from the experiments is that nonparametric G modeling not only produces good performance for point estimation and prediction, but it is also capable of delivering reliable estimates of the precision of such estimates. In contrast linear shrinkage and associated "least favorable" confidence interval procedures, as least as currently implemented in Armstrong, Kolesár, and Plagborg-Møller (2020) and the associated R package, has room for improvement on both of George Box's (1953) goals of "robustness of validity" and "robustness of efficiency."

Conformal Quantile Regression

17.1 Introduction

Romano, Patterson, and Candès (2019) have recently introduced a variant of conformal inference based on quantile regression. Their approach and implementation is based on machine learning methods, notably QR random forest and deep learning algorithms typically expressed in python. I thought it might be interesting to make an R implementation. I will focus on low dimensional settings that can be adequately modeled with the `rqss` function from the **quantreg** package. The function `rqss` is a general purpose additive modeling function with univariate and bivariate functional components. Component functions are penalized by total variation of their derivatives.

The essential feature of the conformal inference approach is a sample splitting device that allows one to adjust a confidence band constructed with training data based on its performance on a validation sample. The R implementation using `rqss` is very simple and looks like this:

```
conformal <- function(formula, taus = c(0.05, 0.95), newdata, split = 0.5,
                      data = parent.frame(), method = "fn", ...){

  D <- data[,all.vars(formula)]
  s <- c(1,2,sample(1:nrow(D), floor(split * nrow(D))))
  qlo <- rqss(formula, tau = taus[1], data = D[s,], method = method, ...)
  qhi <- rqss(formula, tau = taus[2], data = D[s,], method = method, ...)
  x2 <- D[-s,]
  y2 <- D[-s,1]
  ylo2 <- predict(qlo, newdata = x2)
  yhi2 <- predict(qhi, newdata = x2)
  E <- pmax(ylo2 - y2, y2 - yhi2)
  QE <- quantile(E, diff(taus))
  pred <- NULL
  if(!missing(newdata))
    pred <- cbind(predict(qlo, newdata = newdata) - QE,
```

```

        predict(qhi, newdata = newdata) + QE)
  list(qlo = qlo, qhi = qhi, QE = QE, pred = pred)
}

```

Estimates of lower and upper conditional quantile functions are made, predictions are then made for the validation sample observations, and finally a modified prediction interval is made for a potential newdata set. The main impediment to this seemed to be the fact that the current version of **quantreg** didn't have a **subset** argument for the **rqss** function. Fixing this involved relearning some tricks of the formula processing trade, but had the added benefit that it should have been there all along. In the end I realized that there was a simpler strategy that avoided using the **subset** argument, but c'est la vie.

Several other issues require some further investigation:

- Another issue is that the **predict** method for **rqss** doesn't know how to extrapolate. I would be curious to know how other implementations deal with this, but I've not looked into it, yet. This issue is kludged in the example to follow by introducing two extreme elements of the training data.
- The function **rqss** relies on total variation penalized quantile regression with univariate terms in the additive model penalizing total variation of the first derivative of the estimated function. In the next example, taken from Romano, Patterson, and Candès (2019) it will be evident that it might have been better to penalize the function itself rather than its derivative.
- The function **rqss** is intended to be a full service additive modeling function, however the current code hasn't been tested with multiple components, or for that matter for bivariate components.
- As usual it is difficult to select smoothing parameters for the TV penalization, whether machine learning methods can assist in this respect remains to be seen.
- The performance guarantee offered by the conformal predictions is based on average coverage over a new hypothetical sample like that of the training and validation samples, so it may provide little comfort to anyone looking for a narrower (or wider) focus on the design space. Some further comments on these issues is available in Barbor, Candès, Ramdas, and Tibshirani (2020).

```

# Code for Figure 1
# See: https://github.com/yromano/cqr/blob/master/cqr_synthetic_data_example_1.ipynb
require(quantreg)
set.seed(1729)
n = 7000
x = c(0,5, runif(n, 0, 5)) # Docs sez U[1,5], code sez U[0,5]
y = rpois(n+2, sin(x)^2 + 0.1) + 0.03*x*rnorm(n+2) + 25*(runif(n+2) < 0.01) * rnorm(n+2)
D = data.frame(y = y, x = x)
plot(x,y, cex = .25, ylim = c(-2,6), col = 'grey')
newx = seq(0,5,by = 0.05)
f = conformal(y ~ qss(x,lambda = .5), split = 2/7, data = D, newdata = list(x = newx))
plot(f$qlo, add = TRUE, col = 2, lwd = 3)

```

```
plot(f$qhi, add = TRUE, col = 2, lwd = 3)
lines(newx, f$pred[,1], col = 4)
lines(newx, f$pred[,2], col = 4)
```

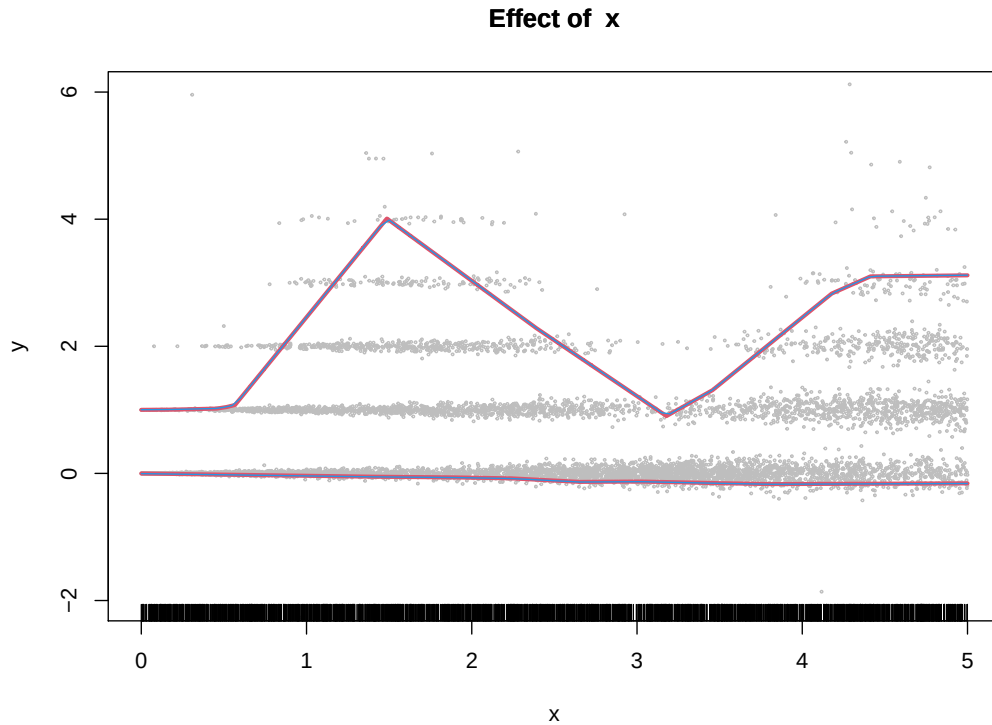


Fig. 17.1: Example 1 of Romano, Patterson and Candès: As shown in the code below, the response is concentrated in bands determined by a Poisson component with some quite extreme outliers that are (mostly) invisible in this plot. The Poisson rate is periodic accounting for the obvious heteroscedasticity. The red curves depict the predicted 0.05 and 0.95 conditional quantile estimates based on the training data, while the blue curves depict the conformally modified estimates. In this example the conformity score E is quite small and the conformal modification is negligible.

It is interesting, perhaps, to compare the conformal band with more traditional confidence bands for the two conditional quantile functions illustrated in Figure 1. As described in Koenker (2011a) it is relatively easy to construct pointwise bands based on conditioning on the smoothing parameter λ , and it is only slightly more difficult to construct uniform bands using the Hotelling tube technique. It is striking to compare these bands, shown in

Figure 2, with the conformal band and contemplate the differences in the promises they make about the arrival of new data.

Code for Figure 2

```
plot(x,y, cex = .25, ylim = c(-2,6), col = 'grey')
plot(f$qlo, add = TRUE, col = 2, lwd = 3, bands = "both")
plot(f$qhi, add = TRUE, col = 2, lwd = 3, bands = "both")
```

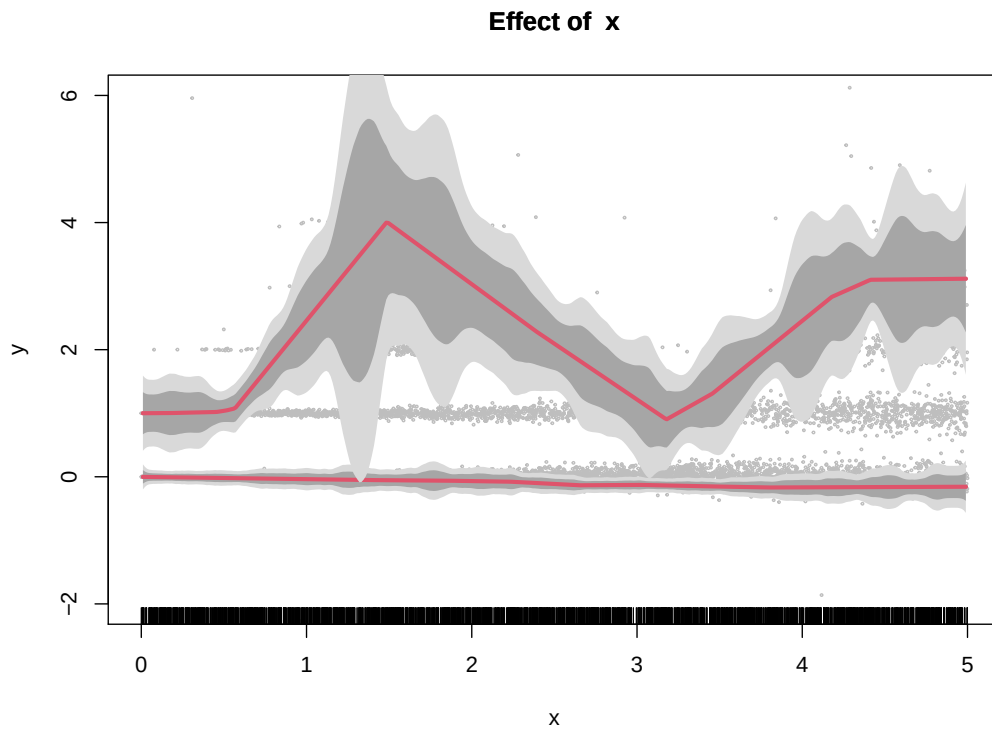


Fig. 17.2: Pointwise and Uniform Confidence Bands for RPC Example: In contrast to the conformal prediction band, pointwise and uniform bands for the 0.05 and 0.95 conditional quantile functions are considerably wider. The uniform band is based on the Hotelling tube construction described in Koenker (2011) and is depicted as the light grey shaded band enclosing the darker grey pointwise band.

17.2 Total Variation Penalization of g Itself

As noted above the RPC example begs for penalization of $TV(g)$ rather than $TV(g')$. This turns out to be remarkably easy to implement, so I've added an optional argument `Dorder` to the `qss` function. When `Dorder = 1` as in the default we penalize $TV(g')$ as illustrated above, while if `Dorder = 0`, $TV(g)$ is penalized instead. The resulting fit for the RPC example is illustrated in Figure 3, lending some support to the original conjecture.

```
# Code for Figure 3
plot(x,y, cex = .25, ylim = c(-2,6), col = 'grey')
newx = seq(0,5,by = 0.05)
f = conformal(y ~ qss(x,lambda = .05, Dorder = 0), split = 2/7, data = D, newdata = list(x = newx))
plot(f$qlo, add = TRUE, col = 2, lwd = 3)
plot(f$qhi, add = TRUE, col = 2, lwd = 3)
lines(newx, f$pred[,1], col = 4)
lines(newx, f$pred[,2], col = 4)
```

17.3 Conclusion

Conformal inference for quantile regression is relatively easy to implement for the additive modeling formulation of the `rqss` function in R. However, there are several remaining issues, not the least of which is whether the performance guarantees offered by the method are relevant to practical decision making.

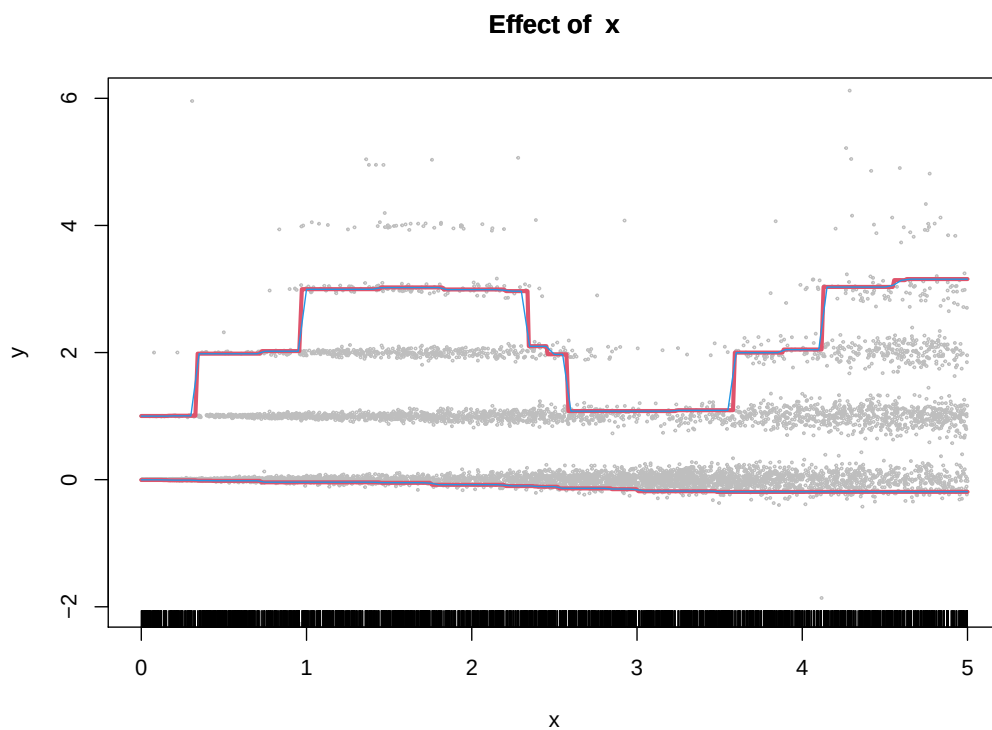


Fig. 17.3: Example 1 of Romano, Patterson and Candès: As a contrast to the earlier piecewise linear fit obtained by total variation penalization of the first derivative of g . Here we plot the fit for a total variation penalized estimate of g itself. Clearly this penalty is better suited to the example and mimics quite well the fit depicted in the RPC paper.

Extremal Quantile Regression Inference

1

18.1 Introduction

Confidence intervals for extreme quantiles require a distinct theory from that describing the behavior of central quantiles. This theory has been pioneered for quantile regression by Victor Chernozhukov in his Phd thesis and a considerable body of subsequent work, as reviewed in (Chernozhukov, Fernandez-Val, and Kaji 2018). In this note I would like to describe very briefly the implementation of some of these methods in the R package **quantreg**.

18.2 Confidence Intervals for Extreme Regression Quantiles

In accordance with general R principles inference about various linear model coefficient estimates is available by subjecting a fitted object containing point estimates to a **summary** function. In the case of fitted objects produced by the function **rq** in the **quantreg** there are already quite a few options to choose among; the latest addition to these methods is the option **se = "extreme"**. It uses a relatively new bootstrapping option described in (Chernozhukov, Fernandez-Val, and Melly 2019) to do subsampling and produces percentile intervals for each of the estimated coefficients. There are several tuning parameter choices involved in this procedure: the size of the subsamples, denoted by **mofn**; a bandwidth, denoted **kex**; and the number of bootstrap replications, denoted by **R**. These parameters are described in more detail in the documentation of the function **summary.rq** and of course in the code itself.

To illustrate the use of this option we will consider the financial application in (Chernozhukov, Fernandez-Val, and Kaji 2018), investigating value-at-risk estimates for Citigroup return

¹ Thanks to Victor Chernozhukov and Ivan Fernandez-Val for making their code available and suggesting an implementation in **quantreg**. Thanks too to Shuowen Chen for helpful comments on an earlier version of the **quantreg** implementation.

from 2009 to 2015. The data is constructed as described in the supplemental material for the Handbook chapter and stored as the compressed image `D.Rda`. Then the code to reproduce a version of Figure 18.2 of their chapter is quite simple.

```
library(quantreg)
load("D.Rda")
p <- ncol(D)
mofn <- floor(50 + sqrt(nrow(D)))
taus <- (1:99)/100
fit <- rq(Y ~ ., tau = taus, data = D)
extreme <- summary(fit, se = "extreme", mofn = mofn, kex = 5+p)
pfit <- plot(extreme)
```

```
central <- summary(fit, se = "ker")
plot(central, ylim = pfit$ylim) # Use "extreme" plotting limits
```

One shouldn't draw any firm conclusions from a single realization like this, but I hope that the new syntax might encourage someone to make a more systematic comparison. It is slightly odd that the extremal method produces a few very short intervals for central quantiles, but of course it is not intended to be used for central quantiles. (Chernozhukov, Fernandez-Val, and Kaji 2018) suggest the rule of thumb that $\tau n/p < 30$ as a threshold for the use of the extremal intervals.

18.3 Estimating Pareto Tail Exponents

The other new feature introduced for handling extremal quantile regression problems is a pair of functions for estimating the tail index of the response observations. These are implemented in a umbrella function `ParetoTest` with options for the (Hill 1975) method and the (Pickands 1975) method. To explore the performance of these two methods a very simple simulation experiment was undertaken. The setup is a very simple bivariate linear model with iid Student t_2 errors. The Pareto tail exponent is thus $1/2$, and we would hope and expect that both methods would produce estimates and confidence intervals that reflected this target value. As can be seen in the summary table, the Hill estimator is quite well centered at the desired value, but the Pickands estimator is substantially downward biased, as is its bias corrected version and confidence interval. This is confirmed in an examination of histograms of the 500 replications of the two estimators illustrated in Figure 3.

```
# Toy Simulation of Hill vs Pickands for t2 model
require(quantreg)
set.seed(1729)
n = 2000
R = 500
A = array(0, c(R, 6, 2))
x = rnorm(n)
```

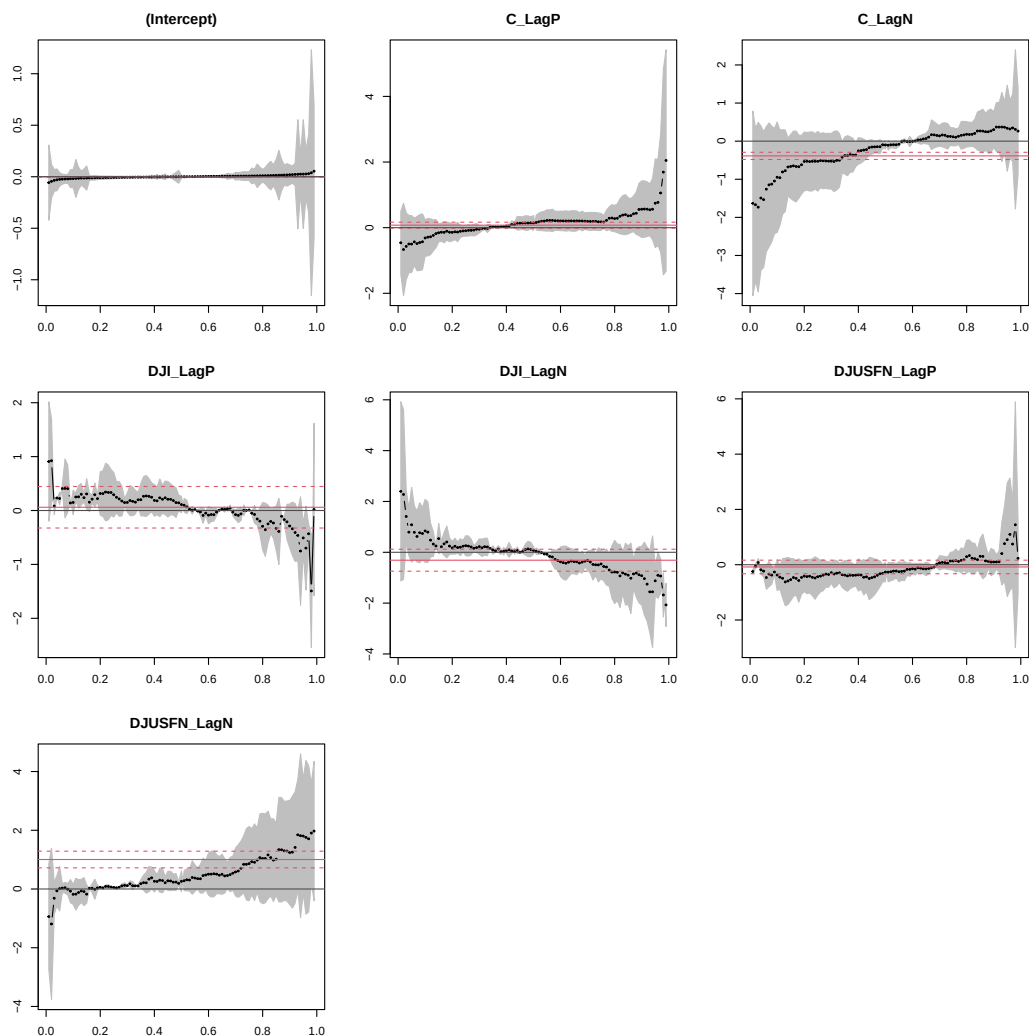


Fig. 18.1: Extreme Value Quantile Inference for Citigroup Data: The pointwise bands illustrated here use the bootstrap method of Chernozhukov et al

```
for(i in 1:R){
  y = x + rt(n,2)
  A[i,,1] = ParetoTest(y ~ x, 0.05, flavor = "Hill")$z
  A[i,,2] = ParetoTest(y ~ x, 0.05, flavor = "Pickands")$z
}
B = apply(A,2:3,mean)[1:4,]
dimnames(B) = list(c("Estimate", "BC-Estimate", "Lower", "Upper"),
```

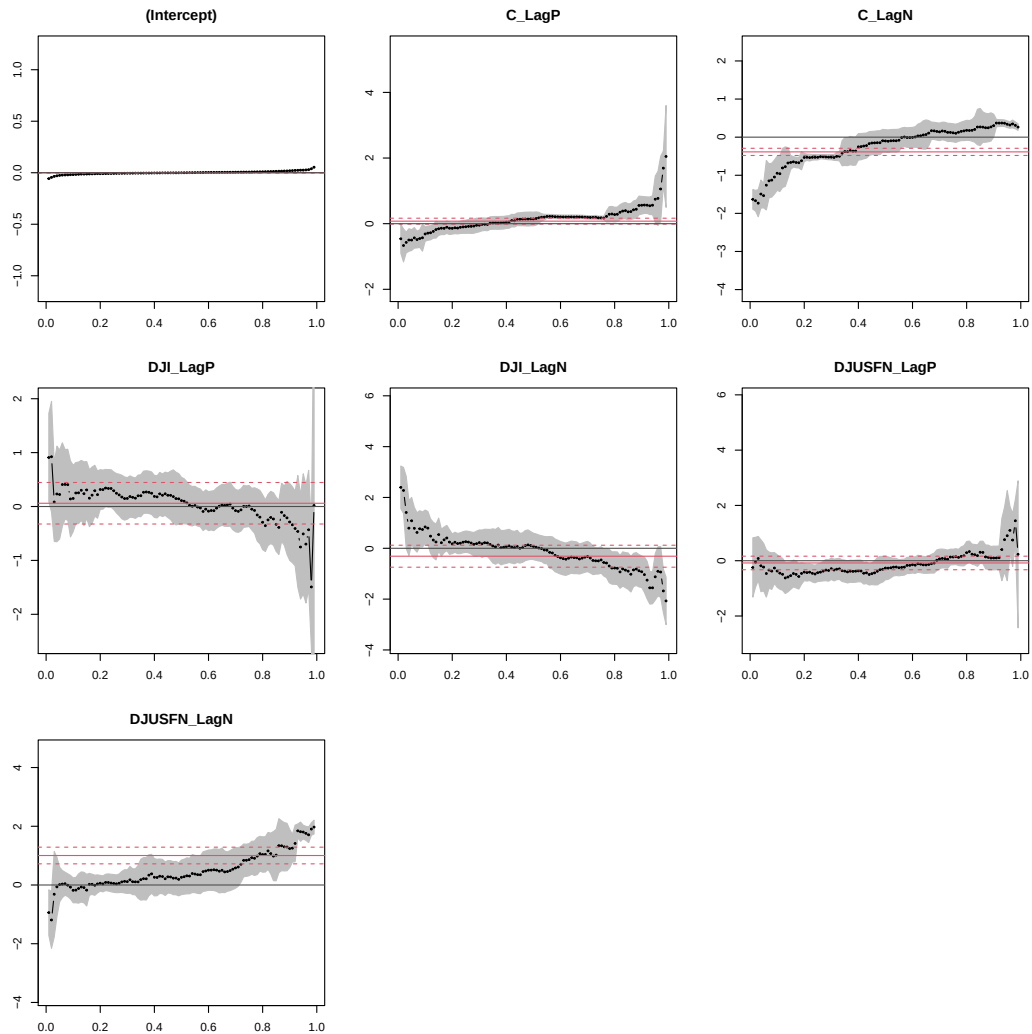


Fig. 18.2: Conventional Central Quantile Inference for Citigroup Data: The pointwise bands illustrated here use the kernel method of Powell.

```
c("Hill", "Pickands"))
```

```
par(mfrow = c(1,2))
load("sim1.Rda")
hist(A[,1,1],40, main = 'Hill Estimates', xlab = 'Estimate')
hist(A[,1,2],40, main = 'Pickands Estimates', xlab = 'Estimate')
```

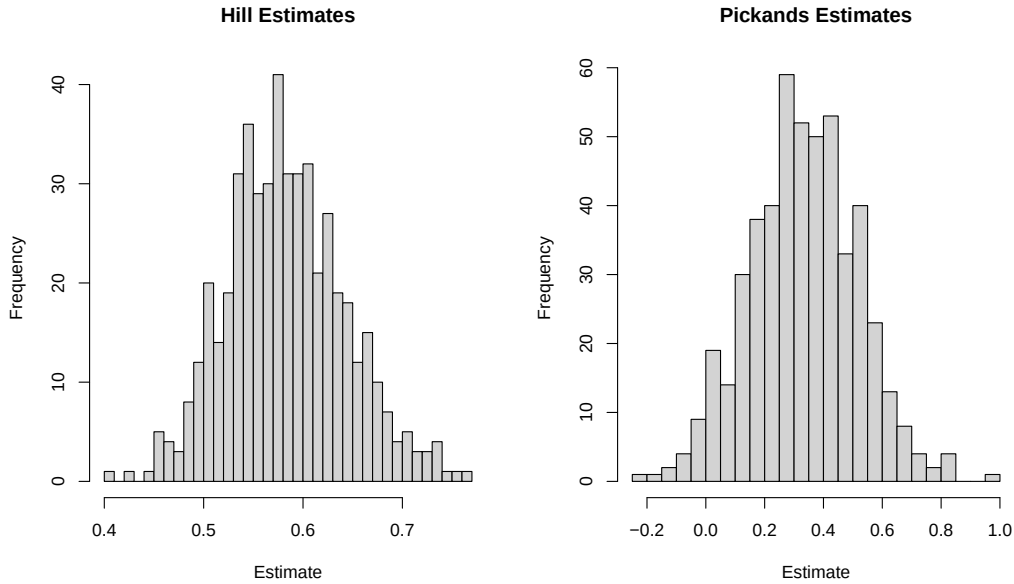


Fig. 18.3: Histograms of 500 Hill and Pickands estimates of a Pareto Tail Exponent: The simulation setting has target tail exponent of $1/2$, so the Hill estimator is quite well centered, however the Pickands estimator is somewhat downwardly biased.

	Hill	Pickands
Estimate	0.583	0.333
BC-Estimate	0.507	0.375
Lower	0.399	0.068
Upper	0.604	0.685

Table 18.1: Simulation Results for the ParetoTest Function: A simple bivariate linear model with iid Student t errors with two degrees of freedom is considered. Mean estimates based on 500 replications for Hill and Pickands estimates are reported as well as bias-corrected estimates and percentile confidence bounds based on 200 bootstrap replications.

18.4 Conclusion

Inference for extremal quantile regression is an important practical objective that has heretofore been neglected in the **quantreg** package. It is to be hoped that making the implementations described above available will encourage others to explore and improve upon these features.

Deracinated Quantile Regression

19.1 Introduction

Suppose you have names for triangles and squares and now you need a name for a pentagon: should you call it a generalized triangle, or a generalized square? Having chosen, what would you now call an hexagon to distinguish from the pentagon? You see where this is going? Generalized isn't really very descriptive.¹

What is “deracinated quantile regression” you may ask. It is my alternative term for “generalized quantile regression” as introduced in Powell (2020). Deracinated means uprooted, that sense of being torn away from your homeland by some involuntary force of man or nature. DQR isn't regression and it isn't even really about quantiles. What is regression? There are quite a few flavors, but like ice cream these flavors should have some common feature, and for regression that feature is that they should be methods for estimating *conditional* somethings – mostly, I would happily concede – conditional mean functions.² However, in the case of quantile regression conditional quantile functions are the objective, the parameter, if you will. Even instrumental variable quantile regression as introduced by Chernozhukov and Hansen (2004) is focused on estimating conditional quantile functions, as is the triangular system estimators of Chesher (2003) and Ma and Koenker (2006). What about “unconditional quantile regression” I hear you ask? This too is conditional, the model is

$$\mathbb{P}(Y < \xi | X = x) = \Lambda(x^\top \beta(\xi))$$

so conditional probabilities of the response falling below a fixed threshold ξ are modeled by a linear predictor transformed by a monotone link function. By varying ξ we trace out the complete family of the conditional distribution functions.³

¹ I confess that for most of 1975 Gib Bassett and I called regression quantiles “generalized sample quantiles for the linear model.” But eventually we realized that this was foolish and we began using the term “regression quantiles,” the term then stuck until Moshe Buchinsky and Gary Chamberlain began using the inverted term “quantile regression.” At that point it became more than a statistical curiosity, and for me, for a time, became (almost) a way of life.

² For an elegant exposition of this viewpoint, see Goldberger (1962).

³ For a very elementary comparison of quantile regression and distributional regression, see Koenker (2011b).

Given either a family of conditional quantile functions or a family of conditional distribution functions, one can easily build other objects of interest. They are the bricks and if made with with the appropriate mixture of hypothetical straw one can estimate various counterfactuals: for example marginal distributions of the response by integrating out various marginal distributions of the covariates, as in Mata and Machado (2005) and Chernozhukov, Fernandez-Val, and Melly (2019).

DQR as imagined by Powell (2020) falls somewhere in a crevasse between the objectives of estimating conditionals and estimating marginals of this type. He posits two types of covariates, a policy manipulable sort, called D , and some others, called X . To illustrate this consider the first of Powell's simulation examples which looks like this:

$$Y_i = (1 - D_i) * (U_i + X_i), \quad X_i \sim U[0, 1], \quad D_i \sim U[0, 1], \quad U_i \sim [0, 0.1].$$

The conditional quantile functions of the response look like this:

$$Q_{Y|D,X}(\tau|d, x) = x - xd + \frac{1}{10}(\tau - \tau d),$$

so there is an interaction term in a location shift and simple form of heteroscedasticity that implies that the distribution of the response is degenerate at $D = 1$. The quantile treatment effect is,

$$\frac{\partial Q_{Y|D,X}(\tau|d, x)}{\partial d} = -x - \frac{1}{10}\tau,$$

This is a rather strange model, but it can be estimated quite accurately.

Powell Simulation Setting 1.

```
require(quantreg)
set.seed(1729)
n = 1000
D = runif(n)
X = runif(n)
U = runif(n,0,0.1)
Y = (X + U)*(1 - D)
f = rq(Y ~ X + D + I(X*D), tau = 1:9/10)
round(f$coef,3)
```

	tau= 0.1	tau= 0.2	tau= 0.3	tau= 0.4	tau= 0.5	tau= 0.6	tau= 0.7
## (Intercept)	0.008	0.015	0.027	0.040	0.049	0.060	0.072
## X	1.002	1.007	1.003	1.002	1.002	1.003	0.999
## D	-0.007	-0.014	-0.027	-0.040	-0.050	-0.061	-0.072
## I(X * D)	-1.002	-1.008	-1.003	-1.002	-1.001	-1.002	-0.999

	tau= 0.8	tau= 0.9
## (Intercept)	0.083	0.092
## X	0.996	0.996
## D	-0.083	-0.092
## I(X * D)	-0.996	-0.996

So as expected the main effect of X and the interaction term have coefficients of 1 and -1, respectively, while the main effect of the treatment variable, D , fluctuates around its correct value, $\tau/10$. We can contrast this with what happens when we fail to condition on the covariate X . Now,

$$Q_{Y|D}(\tau|d) = Q_V(\tau)(1 - d)$$

where $V = X + U$. The quantile function of the convolution, V , because U is supported on the interval $[0, 0.1]$ is almost the same as the quantile function of the standard uniform, that is to say, the identity function. See Figure 1. Thus, $Q_{Y|D}(\tau|d) \approx \tau(1 - d)$, and the QTE from this perspective is simply, $-\tau$. Powell comments that, “quantile regression with covariates [i.e. conditioning on X] is not estimating the quantile function of interest since controlling for additional covariates alters the quantile function.” It is not at all clear in any general sense what he means by the “quantile function of interest,” but when D is randomly assigned, i.e. independent of X , then apparently $Q_{Y|D}$ is his intended target. But why? Wouldn’t we like to know that the treatment acts quite differently on patients having different values of X ? Indeed, most of the variability of the response in this, admittedly somewhat bizarre, circumstance is due to X and not to U , so ignoring X seems quite imprudent.

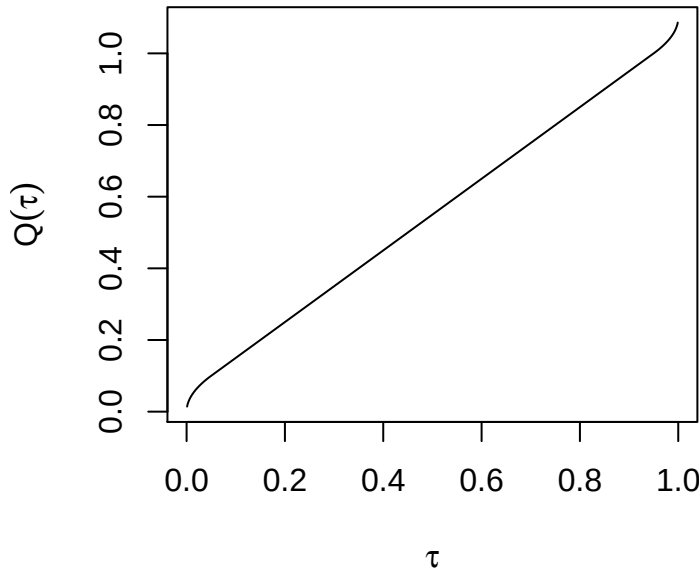


Fig. 19.1: Quantile Function of V

19.2 Conclusion

There are a variety of other troubling aspects of the Powell paper. Aside from the ambiguity of what might be meant by the “quantile function of interest,” there are several places where assumptions are casually introduced that drastically alter the nature of the model under consideration. In particular there seems to be an implicit assumption that the instrumental variables, denoted by Z are valid throughout the full range of the QTE. Such assumptions strike me as dangerous since in practice it is often plausible that they are only effective in some narrow range of τ . This is a point very effectively made in the fundamental paper of Chesher (2003). Computational implementation of the proposed methods by the generalized method of moments⁴ is also a travesty given the non-smooth nature of the underlying problem.

⁴ What is needed in such circumstances is a generalized method of mollifiers; neither of these methods should be confused with the generalized method of moments that seems to be the dominant feature of modern political economy.

λ -Selection

1

20.1 Introduction

Before I became obsessed with models for conditional quantile functions I was enamored by the work of Grace Wahba on smoothing splines. Eventually, there was an opportunity to combine these obsessions in Koenker, Ng, and Portnoy (1994) and Koenker and Mizera (2013) which replaced the classical L_2 smoothing penalty with an L_1 penalty that had the interpretation of penalizing the total variation of the first derivative of the fitted function or its gradient, respectively. These methods were implemented in my **quantreg** package and have been gradually developed over the years, the most important step in this development, described in Koenker (2011a), was the introduction of methods for estimating and plotting confidence bands for the fitted functions. A glaring lacuna in this development, the fly in the ointment, so to speak, is that I never managed to provide a decent method for automatically selecting the parameters that control the degree of smoothness imposed by the methods.

It is true that in the papers referred to already, there was some mumbling about minimizing AIC/BIC like criteria, and hidden away in **quantreg** there is a **demo** that shows how to implement an AIC procedure for so-called λ selection for univariate components. This failure of personal courage was partially due to some early, traumatic experiences with GCV methods for classical smoothing splines. There are no shortage of proposals for λ selection in related problems, but it always seemed to be one of those problems that was best left to others.

¹ After circulating an initial draft of this note I received a very thoughtful email from Likun Zhamg the lead author of the Netflix paper who noted that in the very large samples that they were considering the MCV approach did tend to produce much smoother objective functions than those reported with much smaller samples in the present note. At the same time, the AIC procedures difficulties with identifying the “correct” dimensionality of the fitted model became more severe for larger sample sizes. Both of these points seem quite persuasive to me and I hope to pursue the MCV selection approach in future work.

Then, a few months ago, I received a note from someone at Netflix who mentioned that his colleagues had had some success using our TV smoothing methods in A/B testing for models of network latency. A bit of googling revealed a recent paper Zhang, Castillo, Berglund, and Tingley (2020) that described this and some other successful applications of k -fold cross validation combined with the Kleiner, Talwalkar, Sarkar, and Jordan (2014) bag of little bootstraps idea for both univariate and bivariate smoothing. This provided an incentive to revisit the problem of λ selection in the hope that I could finally provide a viable implementation. I will briefly describe here some new experience with both the AIC approach and with a variant of k -fold cross validation. Unfortunately neither seems to offer the panacea that I was looking for.

20.2 The AIC Method

Nonparametric smoothing for conditional quantile functions is implemented in **quantreg** by the **rqss** function. This function is modeled after the **gam** function in the **mgcv** package of Wood (2017). The model is specified by a formula that describes a series of additive terms: nonparametric terms are represented by contributions that look like **qss(x, lambda = lam)**, where **x** is a conditioning covariate, and **lambda** is the smoothing parameter for the penalty contribution,

$$\lambda P(g) = \lambda TV(g') = \lambda \int |g''(x)| dx,$$

and linear (parametric) terms appear naked separated by “+” signs. (Inside the **qss** terms one can also restrict the fitted g functions to be monotone, or convex, or concave, but I will not dwell further on this, here.) The idea behind the AIC method of selecting λ ’s is to define an effective dimensionality of the fitted model and use that to construct an objective function to be minimized, as λ increases the fitted g is forced to become more linear. Effective dimensionality of any $\hat{g}$ can be defined in several ways. Penalizing total variation of g' implies that the $\hat{g}$ must be piecewise linear, so we can simply count the number of distinct pieces for each additive, nonparametric term. Alternatively, one can simply count the number of interpolated observations to compute the effective dimensionality of the model as a whole. The adverb “simply” is used ironically here in view of the fact that both of the aforementioned procedures rely on dubious features of finite precision arithmetic. Distinguishing what is zero from what is small is notoriously difficult. It should be noted however that this is no different from the situation often recommended for L_2 penalized smoothers where the trace of the quasi-projection matrix plays this role. An elegant unification of such estimates of dimensionality is provided by the proposal of Martin and Walker (2013) to use

$$\dim(\hat{g}) = \text{div}(\hat{g}) = \sum_{i=1}^n \partial \hat{g}(x_i) / \partial y_i.$$

In the case of quantile regression this formula reduces to “simply” counting the number of interpolated points, since $\hat{g}$ is insensitive, locally, to movements of the uninterpolated points.

Given an estimate, p_λ , of the dimensionality of the fitted model, the AIC criterion may be expressed as,

$$AIC(\hat{g}_\lambda, k) = 2n \log(n^{-1} \sum_{i=1}^n \rho_\tau(y_i - \hat{g}_\lambda(x_i))) + kp_\lambda.$$

The original AIC criterion would use $k = 2$, while the Schwarz criterion would use $k = \log n$.² When $k < 0$ the Schwarz option is employed. In the example illustrated in Figure 1, I've followed my past inclination and used the Schwarz variant of the AIC criterion that tends to produce somewhat smoother estimates.

In Figure 1 I've illustrated an example of this sort of AIC fitting. As can be seen in the R code, the data illustrated in the right panel has been generated by a simple location-scale model with logarithmic mean and Gaussian noise. The optimization of AIC is carried out with the R function `optimize`, but as can be seen in the left panel where $AIC(\lambda)$ is plotted on an equally spaced grid the objective is quite rough. This makes the problem of optimization quite challenging: an optimistic view would be that the fit isn't very sensitive to the choice of λ in the region from about 0.5 to 1.5, so the utility of a highly accurate optimization can easily be exaggerated. On the other hand if one imagines solving for several λ 's in a more complicated additive model the problem seems more daunting.

```
arqss <- function(x, y, tau, interval){
  g <- function(lam, y, x, tau)
    AIC(rqss(y ~ qss(x, lambda = lam), tau = tau), k = -1)
  lamstar <- optimize(g, interval, x = x, y = y, tau = tau)
  rqss(y ~ qss(x, lambda = lamstar$min))
}
n <- 200
x <- sort(runif(n, 0, 20))
g0 <- function(x, tau)
  log(x) + 0.2*(log(x))^3 + log(x) * qnorm(tau)/4
y <- g0(x, runif(n))
par(mfrow = c(1,2))
lams <- aics <- seq(0.025, 10, by = 0.2)
for(i in 1:length(lams))
  aics[i] <- AIC(rqss(y ~ qss(x, lambda = lams[i])), k = -1)
plot(lams, aics, cex = .5, lwd = 2, type = 'l',
     xlab = expression(lambda), ylab = expression(AIC( lambda )))
plot(x, y, cex = .5, col = "grey")
f <- arqss(x, y, 0.5, c(0.01, 10))
plot(f, add = TRUE, lwd = 2)
lines(x, g0(x, 0.5), col = "red", lwd = 2)
lam <- round(f$lambda, 3)
```

² As usual we are playing fast and loose with additive and multiplicative constants that are independent of λ here, see Koenker (2016b) for gory details. Since we are just looking for where the criterion is minimized such constants are irrelevant.

```
text(10, 1, bquote(lambda == ~ .(lam)))
```

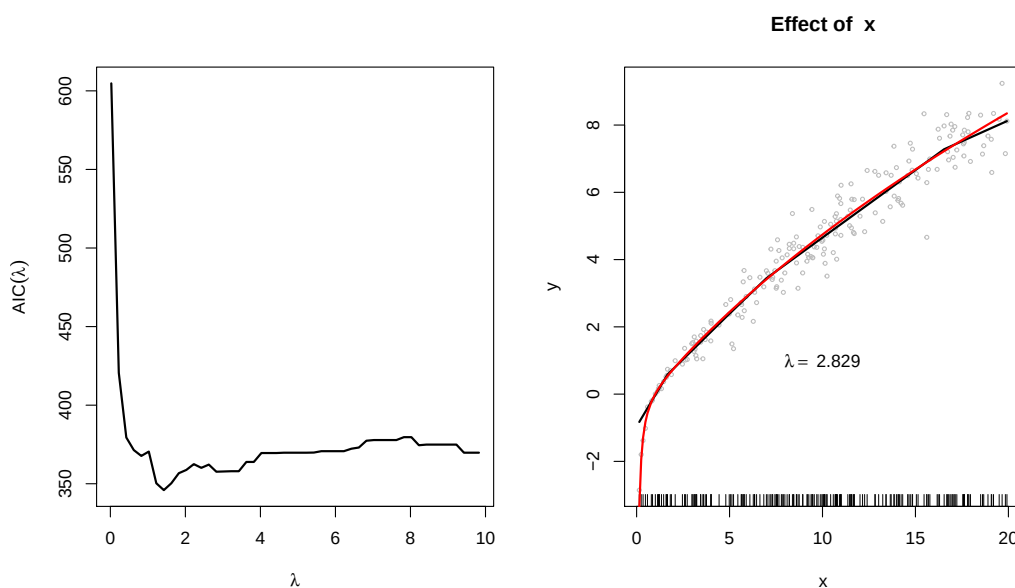


Fig. 20.1: Automatic λ Selection by AIC Criterion: Left panel plots the AIC criterion as a function of λ , while the right panel plots the estimated $\hat{g}_\lambda$ for the selected λ^* . Optimize clearly doesn't find a global optimum here.

20.3 The k -fold Cross Validation Method

The idea of multi-fold cross validation (MCV) was introduced by Geisser (1975) as an alternative to the classical jackknife, and has been widely adopted for many nonparametric smoothing parameter selection problems. Zhang, Castillo, Berglund, and Tingley (2020) argue convincingly that it can be usefully adapted to large-scale nonparametric quantile regression settings as well. The motivation is quite compelling: rather than the traditional delete-1 jackknife, which has well-known efficiency problems in situations where the Sherman, Morrison Woodbury formula isn't available, deleting larger groups can be implemented quite efficiently in a much larger class of problems, especially so since it is quite easily parallelizable. I was quite curious to explore this further.

A basic implementation is quite easy. The parameter k is used to divide the observations into k groups, $V_1, \dots, V_k$ each of approximate size n/k . The model is fit by omitting each of these groups, in turn, and computing a measure of prediction error for the omitted group.

The predict method for `rqss` doesn't understand how to extrapolate so any observations that require extrapolation have to be removed before the prediction error is computed. The prediction errors are then aggregated into a final prediction error for each choice of λ and then minimized over λ . This procedure is illustrated in Figure 2 using the same test problem used for the AIC criterion. As we saw for AIC the MCV criterion also produces a very rough objective function, however since we are minimizing over the gridded values of λ in this case, we happen to select a much smaller λ than with AIC. It would be hard to argue that it is a better λ , however. It yields a somewhat rougher fitted function.

```
MCV <- function(lambdas, formula, data, tau = 0.5, k = 10){
  F <- Munge(formula, lambdas = lambdas)
  f <- rqss(F, data, tau = tau)
  n <- f$n
  m <- length(f$qss)
  y <- f$y[1:n]
  folds = sample(rep(1:k, length = n))
  U = NULL
  for(i in 1:k){
    s = which(folds != i)
    M = rqss(F, data = data[s,], tau = tau)
    nd = data[-s,]
    G = matrix(0, nrow(nd), m)
    for(j in 1:m){ #remove extrapolates, if any
      g = f$qss[[j]]$xyz[,1]
      G[,j] = (min(g[s]) < g[-s]) & (g[-s] < max(g[s]))
    }
    h = as.logical(apply(G, 1, prod))
    u = predict(M, newdata = nd[h,]) - (y[-s])[h]
    U = c(U, (u * (tau - (u < 0))))
  }
  mean(U)
}

n <- 200
x <- sort(runif(n, 0, 20))
g0 <- function(x, tau)
  log(x) + 0.2*(log(x))^3 + log(x) * qnorm(tau)/4
y <- g0(x, runif(n))
D <- data.frame(y = y, x = x)
lams <- mcvs <- seq(.02, 10, by = 0.2)
for(i in 1:length(mcv))
  mcvs[i] <- MCV(lams[i], y ~ qss(x, lambda = lambdas[1]), D)
par(mfrow = c(1,2))
plot(lams, mcvs, cex = .5, lwd = 2, type = 'l',
     xlab = expression(lambda), ylab = expression(MCV( lambda )))
lambdastar <- lams[which.min(mcv)]
```

```

plot(x, y, cex = .5, col = "grey")
f <- rqss(y ~ qss(x, lambda = lambdastar), data = D)
plot(f, add = TRUE, lwd = 2)
lines(x, g0(x, 0.5), col = "red", lwd = 2)
text(10, 1, bquote(lambda == ~ .(lambdastar)))

```

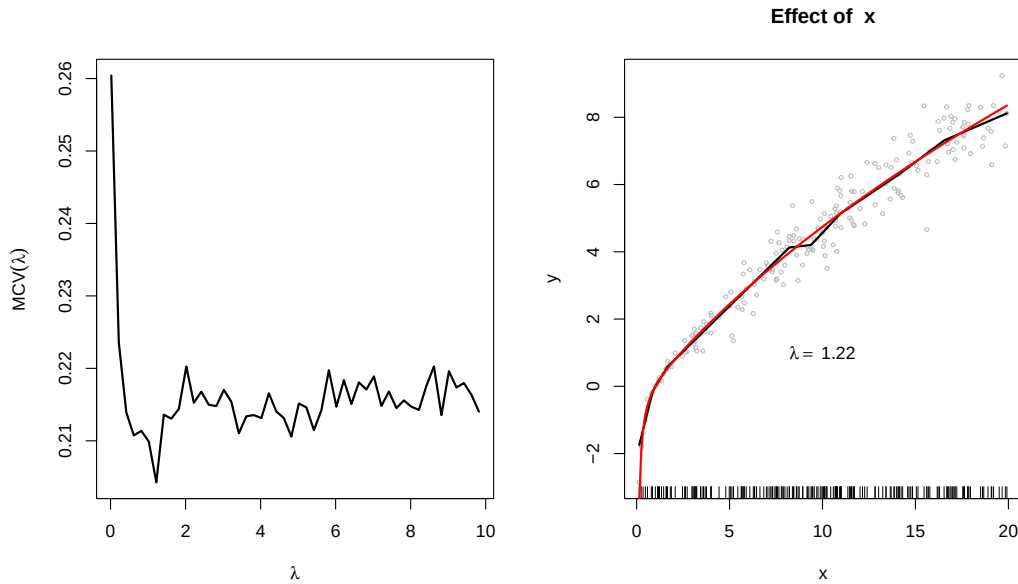


Fig. 20.2: Automatic λ Selection by MCV Criterion: Left panel plots the MCV criterion as a function of λ , while the right panel plots the estimated $\hat{g}_\lambda$ for the selected λ^*

20.4 Conclusion

A theme of these Vinaigrettes is that numerical results like those we have illustrated here prove nothing, by themselves. We could repeat the exercise with larger sample sizes, different test functions or different distributional assumptions, but the unfortunate fact remains that it is difficult to choose λ 's. The more there are, the more difficult it will be. Perhaps we can console ourselves with the countervailing fact that it doesn't matter much, even half-hearted optimization of a rather nasty objective function can produce a decent answer.

Quantile Regression Methods

21.1 Introduction

In the beginning data was scarce and life was hard. Imagine estimating quantile regression models in 1975 for the infamous "stackloss" data, 21 observations with 4 covariates, on a CDC 7600 the size of a small New York kitchen with SAS linear programming algorithms and punch cards. Gradually, data became more plentiful and interacting with computers became easier, in the process methods of estimation and inference for quantile regression became more sophisticated. This note is intended to provide some guidance for selecting among the various methods offered by the author's R package **quantreg** depending on problem size and other characteristics. This is a prelude, I hope, to some constructive new development for the package. In the immortal words of Marx's last thesis on Feuerbach: "Philosophers have only tried to understand the world, our task is to change it." But it usually helps to understand it first.

Figure 1 offers an overview of the scope of the problem. Algorithms for point estimation of quantile regression parameters appear in blue, summary methods for assessing the precision of these estimates appear in pink, among these there are several distinct bootstrap methods appearing in brown, and for each bootstrap method there are one or more bootstrap algorithms available. Quantile regression models can be roughly characterized by four features: the sample size, n ; the parametric dimension of the model, p ; the degree of sparsity of the design matrix; and the number of quantiles to be estimated. Our objective is to offer some advice on how to select methods depending upon these features.

21.2 Algorithms for Point Estimation

Edgeworth described an algorithm for median regression in 1888, but it wasn't until the early 1950's that general median regression algorithms emerged with the development of linear programming. The Barrodale and Roberts (1974) dual bounded variables algorithm for median regression was ideally suited to the elaboration to quantile regression models that were linear in parameters, as described in Koenker and d'Orey (1987). This simplex approach was implemented in S in the late 1970's and remains an efficient method for problems of

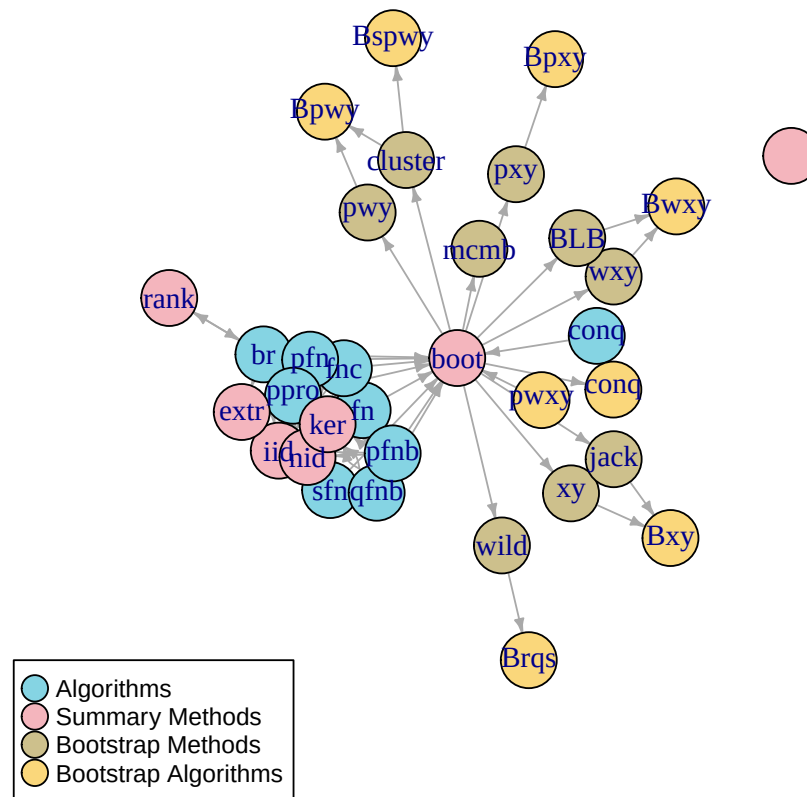


Fig. 21.1: Network of Quantile Regression Methods

modest size. Interior point methods for linear programming that offered advantages for larger problems arrived in the 1990's and have been gradually incorporated into the **quantreg** package, first for unconstrained problems, then for linear inequality constrained problems and finally for problems with sparse designs. There are currently ten distinct algorithms for estimating parametric, linear quantile regression models:

”br” The original simplex algorithm based on Barrodale and Roberts (1974). An unfortunate aspect of this algorithm is that the underlying fortran is highly unstructured and difficult to follow. At some point in 1998 I realized that the basic idea of the algorithm was quite

close the essential idea of Edgeworth's 1888 paper, and I wrote a version in R that is only about 25 lines. This prototype version is available at <http://www.econ.uiuc.edu/~roger/research/rq/rqx.R>.

- "fn" A basic interior point algorithm based on ideas of Mehrotra (1992) that I refer to as the Frisch-Newton method since it can be seen as a Newton-type method with log-barrier constraints as originally suggested by Ragnar Frisch.
- "fnc" A variant of the "fn" interior point algorithm that introduces linear inequality constraints.
- "sfn" A variant of the "fn" method that exploits sparse algebra for problems in which the design matrix is sparse.
- "sfnc" A variant of the "fnc" method that exploits sparse algebra for problems in which the design matrix is sparse.
- "pfn" A variant of the "fn" method that exploits a preprocessing strategy described in Portnoy and Koenker (1997) to reduce the effective sample size of the problem.
- "ppro" A variant of the "pfn" method that exploits preprocessing when multiple quantiles are being estimated.
- "qfnc" A variant of the "fnc" method designed for multiple quantiles all estimated within a fortran loop.
- "pfnc" A variant of the "ppro" method but implemented in fortran.
- "lasso" A variant of the "fn" method that incorporates a lasso, or scad penalty.

Further details about these methods is available from the package documentation for the function `rq` and the references provided therein.

Figure 2 compares cpu time in seconds for four of these algorithms: "br", "fn", "sfn" and "pfn." The model has p standard Gaussian covariates and Gaussian response. When n is less than 25,000 the simplex algorithm of Barrodale and Roberts is quite competitive, but for larger sample sizes the Frisch-Newton algorithm is quicker. Preprocessing is clearly advantageous beyond $n = 100,000$, or so, but its advantage dissipates somewhat for larger parametric dimension, p , of the model. For dense design matrices like these sparse algebra is not helpful. The four methods all achieve the same precision to seven decimal digits.

Figure 3 compares performance in a similar setting except that a factor variable with q distinct levels has been added to 5 dense covariates of the model. Now Frisch-Newton is clearly superior to simplex, but the sparse algebra version of Frisch-Newton is much faster especially when q is large.

21.2.1 Estimation with Many τ 's

Thus far we have only considered estimating a single conditional quantile model, but many applications involve estimating several quantiles as an ensemble. This is relatively easy to do by specifying, for example for the deciles, `tau = 1:9/10` in the call to `rq`. A little reflection will suggest that adjacent quantiles should have somewhat similar solutions, so some form of preprocessing might be helpful rather solving for each value of τ , *de novo*. This idea is closely related to the preprocessing proposed in Portnoy and Koenker (1997); given a solution at some τ_0 if we want a solution at some nearby τ_1 we can presume that the signs of the residuals for τ_0 should be predictive of the sign of the residuals for τ_1 .

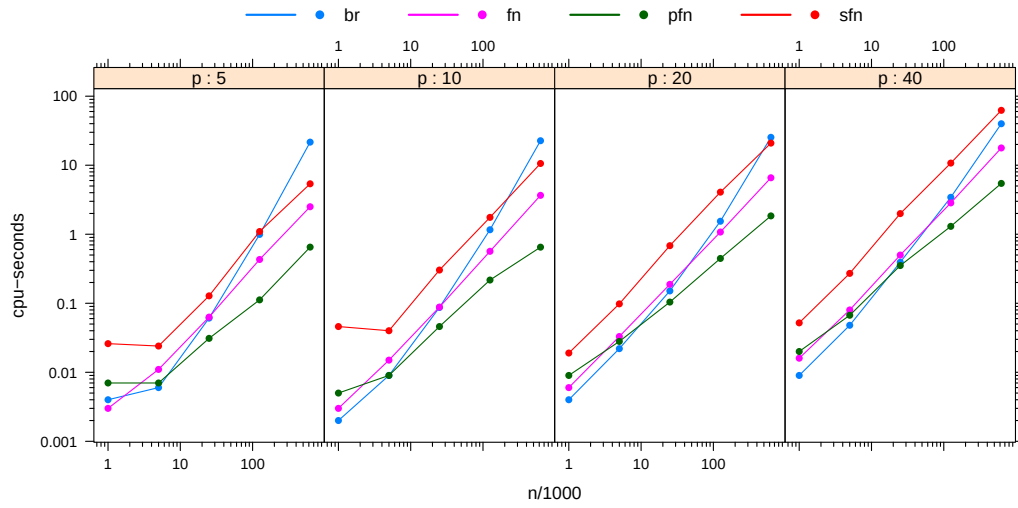


Fig. 21.2: Comparison of Algorithm CPU Time for Dense Problems

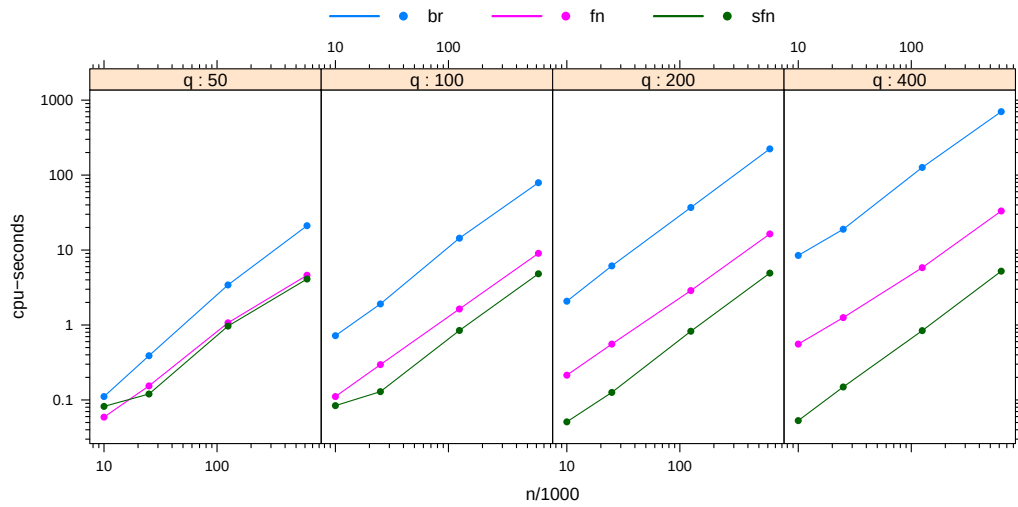


Fig. 21.3: Comparison of Algorithm CPU Time for Sparse Problems

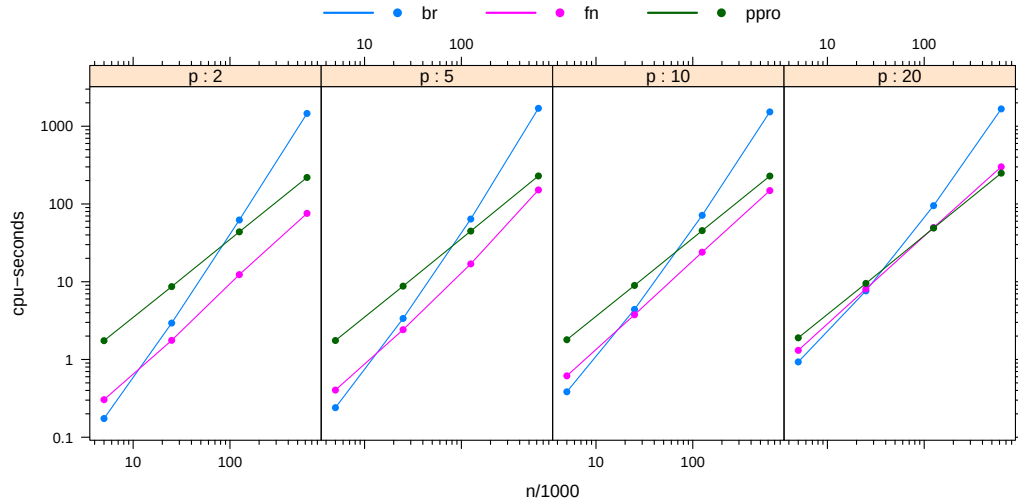


Fig. 21.4: Comparison of Algorithm CPU Time for Multiple taus

This being the case we can combine observations with large negative residuals into one new pseudo-observation with a very negative response, and likewise combine observations with large positive residuals into another pseudo-observation with a very positive response and thereby effectively reduce the effective sample size of the estimation problem. In Portnoy and Koenker (1997) this process was based on a preliminary fit for a model with a fixed τ , but the idea for which we coined the highly sophisticated technical term “globbing” is very similar. Globbing for adjacent quantiles was implemented by Blaise Melly and described in further detail in Chernozhukov, Fernandez-Val, and Melly (2019), and is now implemented in **quantreg** with the `method = "ppro"` option for `rq` function.

Figure 4 compares CPU effort for “br”, “fn” and “ppro” for our dense model (without the factor variable) with $\tau \in \{0.02, 0.04, \dots, 0.98\}$, for 49 distinct, equally spaced τ ’s. The results are quite surprising, so much so that I suspected that I’d blundered somewhere. There is no performance gain from preprocessing, indeed except for the largest of the models fitted, with $n = 625,000$ and $p = 20$, “fn” is always quicker than “ppro”. Profiling the preprocessing code I discovered that most of cpu time was being spent computing sums rather than actually doing the “real work” of optimization. This finding led to a extended detour into the wonderful world of ratfor and fortran.¹

¹ Ratfor is a dialect of fortran (rational fortran) developed at Bell Labs in the late 1970’s providing better control structures and more flexible formatting. Like many ancient languages it is somewhat endangered, but after some extensive googology I found an updated C version maintained by Brian Gaeke that generated fully functional fortran 77. This had the valuable side benefit that it allowed me to clean up some incompatibilities in my earlier ratfor code as well as allowing me to write new code. I’ve added links on my Reproducibility webpage to a tar image of the source files for Gaeke’s version of ratfor and to Brian Kernighan’s brilliant 10 page exposition of the ratfor language.

Figure 5 compares performance of the “ppro” R implementation with two new fortran implementations for the same problems considered in Figure 4. The “qfnb” method was an intermediate prototype to see whether I could still remember how to write ratfor; it simply loops over a vector of τ ’s solving the full n by p problem at each step. The “pfnb” method implements the preprocessing approach of “ppro” but all of the computation stays inside the fortran call. The preprocessing strategy when “fortranized” shows a clear performance advantage. There are several possible refinements, or tuning parameters to choose for the preprocessing; the most crucial of these is the choice of the initial sample size that I have provisionally set to be $n^{2/3}p^{1/2}$. Further experimentation could well lead to even better performance.

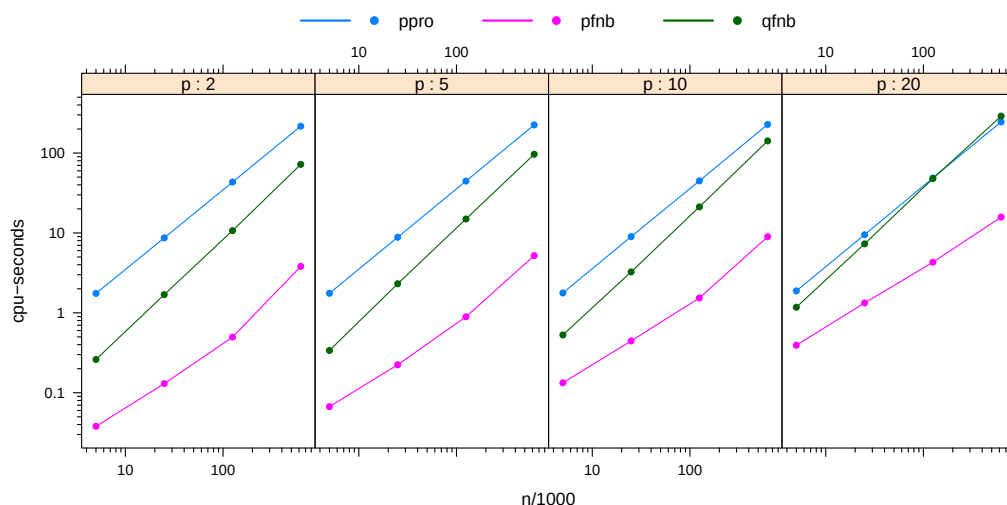


Fig. 21.5: Comparison of New Fortran Algorithms for Multiple taus

There are hints scattered around the R ecosystem, including the authoritative R Core Team (2020) suggesting that the standard R interface with Fortran, i.e. `.Fortran`, is inefficient compared to the more recently introduced `.Call` interface, although I couldn’t find any quantitative evidence for the magnitude of this “overhead” penalty. This prompted me to write an inquiry to the newsgroup R-devel that provoked several responses including the suggestion that the package `dotCall64` might offer a relatively easy way to explore the magnitude of such effects. So in the spirit of scientific exploration I wrote a new version of `rq.fit.pfnb` that simply replaced the call to `.Fortran` with a call to `.C64` in this package and then running a moderately large problem with $n = 500,000$ and $p = 5$ for 49 equally spaced quantiles.

In this example there is a slight performance advantage for the new `.C64` code, which requires 4.37, versus 4.505 seconds for the `.Fortran` version; for somewhat smaller problems

the `.Fortran` was actually quicker. My provisional conclusion from this exercise was that this performance gain didn't justify the additional layer of coding.²

21.3 Summary Methods

In R parlance a “summary” method can be anything that elaborates or coagulates the salient features of an R object. For fitted objects from linear models this typically means that they provide some form of inference about the reliability of the parameters of the fitted model. Like quantile regression fitting methods there are quite a few summary method options. The earliest of these offered direct estimates of the asymptotic covariance (sandwich) matrix, which takes the form,

$$V = \tau(1 - \tau)H_n^{-1}J_nH_n^{-1},$$

where $J_n = n^{-1} \sum_i x_i x_i^\top$, $H_n = n^{-1} \sum_i x_i x_i^\top f_{Y|x}(\xi_i(\tau))$ and $f_i(\xi_i(\tau)) = f_i(x_i^\top \beta(\tau))$ is the conditional density of the response given covariates.

These original methods are:

- iid estimates a scalar conditional density value from fitted residuals assuming that H_n and J_n are the same up to a factor of proportionality,
- nid estimates a vector of conditional density values for H_n by differencing fitted models at $\tau \pm h$, for some bandwidth h ,
- ker estimates H_n by the kernel method of Powell (2020).

Two other methods emerged slightly later:

- rank estimates a confidence interval for each parameter by inverting a rank test derived from the general theory of Groeneboom and Jongbloed (2014) using parametric linear programming methods. This procedure is quite reliable, however it is crucially dependent upon simplex steps in the implementation of “br” fitting routine and consequently is only practical in sample sizes up to a few thousand observations.
- boot as indicated in the earlier network diagram there are now a plethora of bootstrapping methods designed for various circumstances,

The main objective of the present note is to provide some guidance about which of these methods to use under various circumstances, and possibly to suggest directions for future development.

Given a fitted object constructed by one of the foregoing estimation methods any of the summary methods can, in principle, be used by simply invoking the R command `summary(fit, se = meth)` where `meth` is the name of one of these summary methods. As a general rule, for small problems, that is problems with less than 5000 observations, it is advisable to use the `se = "rank"` approach, which produces asymmetric “percentile intervals based on the rank inversion procedure. However, for larger problems this can be rather slow since many simplex pivoting operations are required. There are various other options

² Kaspar Daniel Hansen suggested that perhaps it would also be worthwhile to compare the memory requirements of the two methods and indeed this does reveal a significant gain: 146Mb versus 262Mb for the `.Fortran` version.

to this approach documented in the help file for `rq.fit.br`. These options can be passed along via the `...` argument of the `summary` function. The basic theory of these confidence intervals is sketched in Section 3.5.5 of Koenker (2005) with further references cited around there. To illustrate we can consider a log transformed version of the classical Engel food expenditure model.

```
data(engel)
f <- rq(log(foodexp) ~ log(income), tau = 0.5, data = engel)
summary(f)

##
## Call: rq(formula = log(foodexp) ~ log(income), tau = 0.5, data = engel)
##
## tau: [1] 0.5
##
## Coefficients:
##              coefficients lower bd upper bd
## (Intercept) 0.41833      0.06966  0.90053
## log(income) 0.87659      0.80514  0.93016
```

Another limitation of the rank method is that it operates coefficient by coefficient so it doesn't know how to cope with covariance matrix estimation for the vector of coefficients as would be required, for example, by the `anova` test procedures. By default if no `se` option is specified, the rank method is used if the sample size is less than 1001 and the covariance option for `summary` is not requested, otherwise by default the `se = "nid"` option is invoked. For historical reasons, the “nid” option is treated as a default in such situations even though in retrospect it might have been preferable to use the `se = "ker"` option. The `se = "iid"` option should be avoided except in rare cases where one is very confident about the iid assumption.

When multiple τ 's are specified in the fitting function then a list of fitted objects are returned of class `rqs`, and `summary` then acts on this list sequentially returning a list of tables of coefficients and their confidence intervals in the case that the `se = "rank"` option is used, or standard errors, t-statistics and p-values in the case that one of the other options is used.

21.4 Bootstrap Methods for Quantile Regression

As is evident in the network diagram there are many flavors of the bootstrap for quantile regression and the task in this section is to try to make some sense out of which of them are appropriate in various settings. Let's begin with a brief description of the candidates, in approximate order of introduction into the **quantreg** package:

- xy The classical progenitor of all these methods is the “xy” bootstrap studied by Bickel and Freedman (1981) for the mean regression model. Pairs (x_i, y_i) are drawn at random with replacement, the model is reestimated for each of the R bootstrap samples and a covariance matrix is estimated from the R by p estimates of the coefficients.

- pwy

the Parzen, Wei, and Ying (1994) bootstrap exploits the fact that the subgradient condition for optimality of the quantile regression estimator is a pivotal statistic, so one can resample Bernoulli random variables and again construct R new estimates of the coefficients to produce a covariance matrix.
- spwy

a variant of the “pwy” method that uses the sparse Frisch-Newton method, intended specially for the “cluster” bootstrap method and used automatically when the design matrix of the fitted object is stored in sparse form.
- wxy

instead of the multinomial sampling of the “xy” method one can resample exponential weights and construct weighted quantile regression estimates; this has the advantage that all the observations appear in each replication, albeit with different weights, so it avoids singularities of the resampled design matrices that can sometimes cause problems.
- jack

a variant of the “xy” method that operates on subsamples proposed by Portnoy (2014).
- wild

a variant of the wild bootstrap that has proven valuable in other settings adapted to the quantile regression and proposed by Feng, He, and Hu (2011).
- MCMB

a Markov chain resampling scheme proposed by Kocherginsky, He, and Mu (2004) that updates one coefficient at a time to accelerate inference for large sample sizes,
- BLB

the “bag of little bootstraps” proposed by Kleiner, Talwalkar, Sarkar, and Jordan (2014) intended for large samples.
- cluster

a variant of the wild bootstrap adapted to clustered observations proposed by Hagemann (2017)
- conquer

a variant of the “wxy” method implemented in the **conquer** package of He, Pan, Tan, and Zhou (2020) and using the smoothing methods developed there, intended for large problems.
- extr

a variant of the “pwy” bootstrap adapted for inference for extreme quantiles and proposed by Chernozhukov, Fernandez-Val, and Kaji (2018)
- pwx

A variant of the “wxy” bootstrap implemented in fortran and using preprocessing to accelerate computation, suitable for large applications.

To compound the choice problem each of these options has several sub-options or tuning parameters and potentially more than one implementation in terms of lower level languages. The typical implementation draws random variables required to construct the bootstrap samples, and passes this information to a lower level, either Fortran or in the case of MCMB a C routine, where the matrix of bootstrap coefficients is computed inside a loop. This strategy keeps all the random number generation in R where it is subject to the usual reproducibility mechanisms and avoids potential slow looping constructs in R. In large samples, however, it is subject to memory constraints and overlooks potential opportunities for parallelization.

The selection of a bootstrap method is controlled by the `bsmethod` argument to the `boot.rq` function and can be passed as a argument to `summary`. One of the earliest innovations in quantile regression bootstrapping was introduced in Buchinsky (1994) who proposed computing bootstrap replications on subsamples of the observations rather than on samples of the original size n . This was later justified more formally in Bickel and Sakov (2008) and became the subject of considerable subsequent research. I enjoy the irony that for Moshe it was a case of “necessity as the mother of invention,” a way to speed up progress on his thesis research. This “ m out of n ” version of the bootstrap is implemented with the `mofn` argument of the `boot.rq` function. The number of bootstrap replications is also an impor-

tant influence on both the speed and reliability of the method; the default of $R = 200$ is probably somewhat smaller than would be ideally specified.

An important aspect of the bootstrapping ecosystem in **quantreg** is the relation of bootstrap methods and fitting methods. Early on it was recognized that doing bootstrap replications in an R loop was painfully slow as soon as samples became moderately large, so most of the early methods were implemented by passing sampling weights to a wrapper function that called an implementation of a simplified “br” method inside a fortran loop. This is undesirable since if users have – in their wisdom – estimated their model with the “fn” option or the “sfn” method they should be entitled to expect that the bootstrapping that is done in with the summary method would use the same fitting procedure. An exception to this state of affairs is the “cluster” bootstrap method, which uses either “fn” or “sfn” depending upon whether the design matrix associated with the fitted object is an ordinary R matrix, or special sparse matrix. There is also a prototype “pxy” bootstrap method that implements a variant of the preprocessing strategy used for the “ppro” and “pfmb” fitting methods. It would seem to be desirable to develop a fortran version of this intended for large problems.

To this end, I’ve coded a new `boot.rq.pwxy` option that implements the preprocessing method within a fortran loop. Unlike earlier fortranizations of bootstrap methods that passed a n by R array of data to construct the bootstrap samples, this implementation draws bootstrap weights inside the fortran loop one replication at a time. The former strategy is fine for small sample sizes, but is obviously impractical in terms of memory requirements when n is large. One may worry that generating random variates from within fortran would disrupt the usual reproducibility mechanisms of `set.seed`, etc. Fortunately, R provides functionality to resolve this quite easily. In contrast to the prototype `boot.rq.ppro` which uses multinomial sampling, this version uses the weighted bootstrap with standard exponential weights, which should be more robust especially in settings with factor covariates.

Figure 6 compares timings for three bootstrap methods: “wxy” the fortran simplex version using exponential weighting, “pxy” the Frisch-Newton R version using preprocessing and multinomial sampling, and “pwxy” the fortran Frisch-Newton version using preprocessing and exponential weighting. Timings are reported in seconds for $R = 200$ replications of the bootstrap procedure. The simplex version is clearly not competitive except perhaps at lowest sample size, $n = 500$. In the largest sample size of the experiment, $n = 625,000$ the simplex procedure is between 100 and 500 times slower than the preprocessed procedures. Somewhat surprisingly, at least to me, the fortran preprocessing version is only slightly faster than the R preprocessing version, and then only when the model dimension is moderate. Whether this can be improved by better choice of tuning parameters remains an open question. There is an obvious advantage in multinomial sampling in that it yields a further reduction in sample size, but this is offset – in my view – by the concomitant increase in risk of design matrix singularities. There is a rather delicate balance in setting the width of the initial confidence band in any implementation of the preprocessing approach; in our bootstrap experiment I’ve used $\tau \pm \sqrt{(np)/2n}$. This choice trades off the advantage of reduced sample size against the likelihood of violations of the predicted signs condition that require repeatedly solving the globbed problem. Further investigation of this trade-off is clearly warranted. Some further experimentation with the form of the confidence band has also been suggested by Portnoy (1997). Clearly as p becomes larger the problem becomes more challenging; in many large p applications one might hope that the design matrices would be sufficiently sparse to justify

using the sparse Frisch-Newton algorithm with preprocessing. This suggests yet another option to add to the todo list.

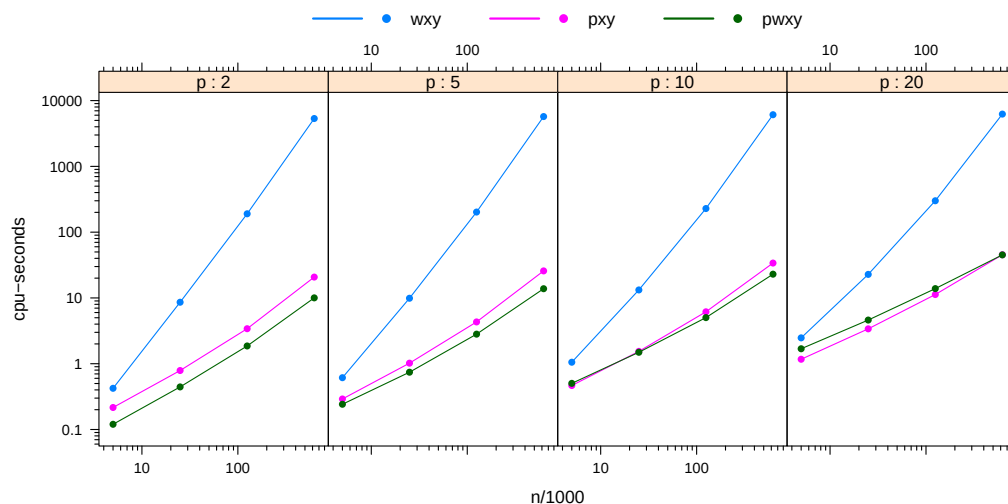


Fig. 21.6: Comparison of Three Bootstrapping Methods

21.5 Smoothing, Gradient Descent and Other Perturbations

Steve Portnoy organized a session at the 2010 JSM in Vancouver with Laurie Davies, Emmanuel Candès and I. At lunch after the session Candès made a very persuasive case for the obsolescence of interior point methods for “big data” applications, arguing that in high dimensional problems it wasn’t practical to do repeated Cholesky factorization of large Hessian matrices. Instead, one needed to rely on first order methods like gradient descent. In my usual lackadaisical fashion I began reading and experimenting with such methods originally motivated by Parikh and Boyd (2013) and the elegant R implementation of Fougner (2014). I implemented a proximal gradient method for fitting quantile regression models. Some experiments with this approach are reported in Koenker (2018). This was my introduction to what we might call “spurious precision in statistical computing.” At the time I was very conscious of precision in convex optimization because I was beginning to compare interior point methods for computing the NPMLE for various mixture problems with solutions from the EM algorithm. EM was very slow to converge, while interior point methods produced much more accurate solutions very quickly. In the mixture setting since one was estimating a distribution function on a relatively fine grid of a few hundred points, the differences in precision were very apparent visually: EM even after a few thousand iterations still produced a rather smooth approximation of the mixing distribution, while the interior point method rapidly converged to solutions with just a few point masses, as expected from the theory.

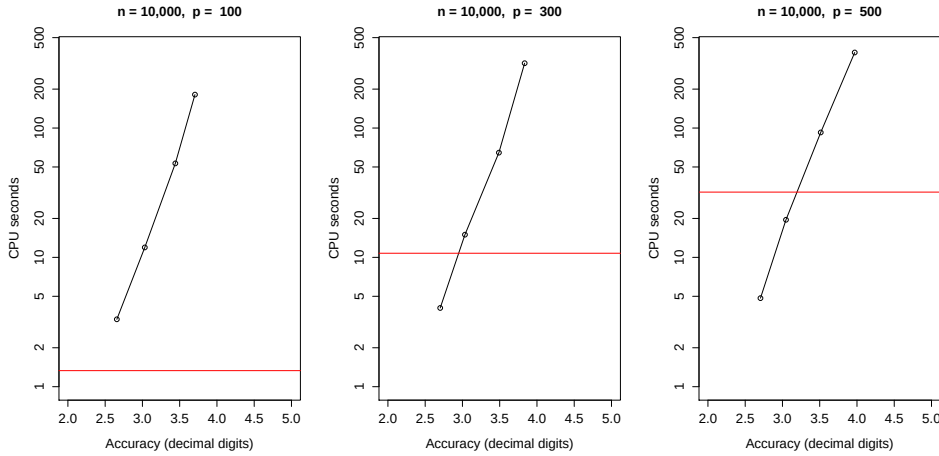


Fig. 21.7: Accuracy vs. Computational Effort: CPU effort (in seconds) is plotted against accuracy in the number of correct decimal digits, averaged over the p coefficients for the POGS GPU solutions to the primal quantile regression problem. Baseline accuracy is determined by the interior point solution depicted by the horizontal (red) line, which is accurate to at six decimal digits. Although the POGS procedure is quite quick to produce a solution with two to three digit accuracy, the effort required to produce better accuracy increases rapidly. In contrast, there is little advantage observed in the interior point timings when the convergence tolerance is relaxed.

Something similar happens with the QR computing in the sense that simplex and interior point methods converge to essentially identical solutions, say up to 7 or 8 decimal digits, whereas the proximal method was quite quick to achieve 2 or 3 digits of precision, but very slow to make further progress. This is depicted in Figure ?? taken from Koenker (2018). However, in the case of the mixture problems accuracy seems important because it reveals a qualitative difference in the nature of the solutions, in the QR setting it is less clear. If we are plotting results, or even if we are reporting tables of results, it is unclear that we need more than a few “significant” digits. After all the original data is rarely more accurate than that. Maybe accuracy is just a fetish? It should be noted, especially so since it wasn’t noted in the original source, that the problem dimensions of Figure ?? are modest by modern standards, and much greater computational gains could be made in larger problems with better GPU hardware and thus more parallelism. I recall Candès mentioning daily CCTV footage from Beijing.

There have been several proposals for implementation of the ADMM (alternating direction method of multipliers) and related first order approaches for quantile regression, but none really attracted my attention until Xuming He sent a preprint of He, Pan, Tan, and Zhou (2020). Drawing on prior work of Fernandes, Guerre, and Horta (2021) and Barzilai and Borwein (1988), Xuming and his coauthors develop a scalable gradient descent based algorithm that performs very efficiently on large problems. Their approach has been imple-

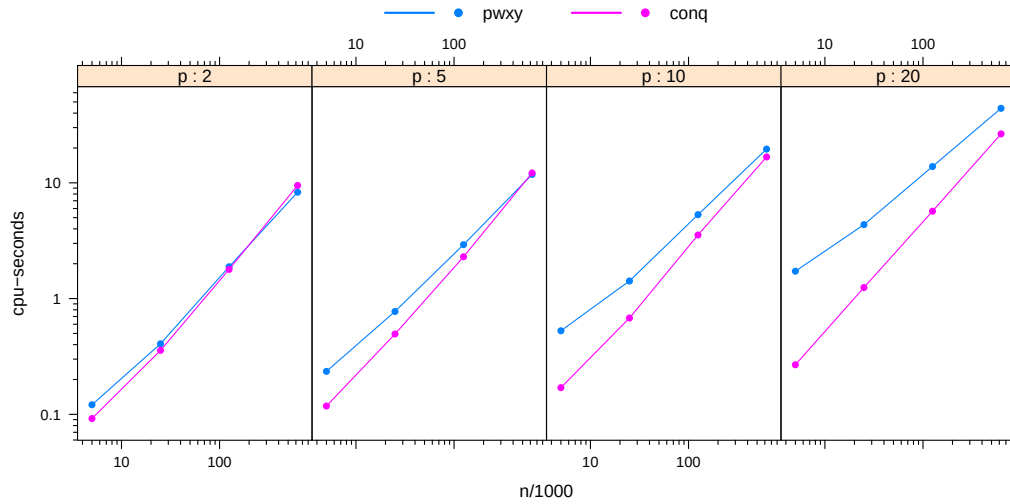


Fig. 21.8: Comparison of Two More Bootstrapping Methods

mented in the R package, **conquer** and I have created a hook in **quantreg** so that both fitting and summary methods can easily access it. In contrast to the interior point methods implemented in **quantreg** that can be seen as taking full Newton steps for a log-barrier formulation of the underlying linear programming problem, their approach adopts a quasi-Newton, gradient descent approach that avoids the full Cholesky factorization at each iteration. Step sizes are chosen according to a proposal of Barzilai and Borwein (1988). A crucial aspect of the new method is the smoothing of the usual quantile objective function as recently proposed by Fernandes, Guerre, and Horta (2021). In contrast to other proposed strategies for smoothing their convolution method preserves the convexity of the objective function and leads to improved performance. Of course a consequence of this smoothing, is that the original problem is somewhat perturbed, but this too may have a silver lining. Provided that the conditional density of the response with respect to the covariates satisfies weak smoothness conditions, smoothing the objective function results in a modest efficiency gain. This is analogous to other methods of smoothing as discussed in Koenker (2005) Section 5.2. The experiments in Fernandes, Guerre, and Horta (2021) and He, Pan, Tan, and Zhou (2020) bear out this theoretical finding.

The improvement in computational speed is particularly important in accelerating bootstrap computations, so I was obviously interested to see how this new gradient descent approach compared to the preprocessing approach described above as “pwx.” Figure 8 reveals that, indeed, gradient descent as implemented in the **conquer** package is substantially quicker than “pwx,” although it appears that for very large sample sizes perhaps it is catching up.

A final comparison for this section involves the “bag of little bootstraps” of Kleiner, Talwalkar, Sarkar, and Jordan (2014) designed for “big data” applications. The implementation of this approach in **quantreg** is somewhat cheesy in the sense that there is no real parallelism

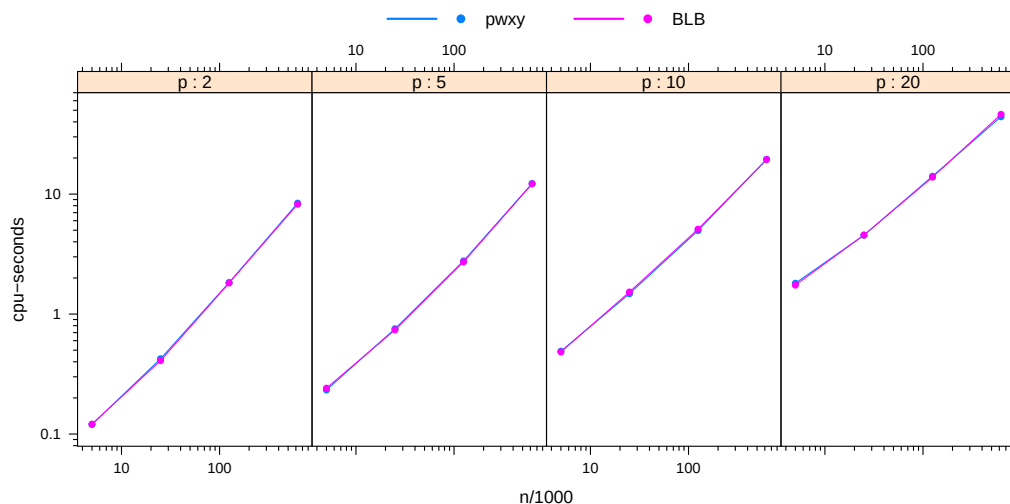


Fig. 21.9: Comparison of Two More Bootstrapping Methods

as one might expect given the original description of the method. On the other hand, I was curious to see how it might compare with the preprocessing approach of `boot.rq.pwxy`. Figure 9 illustrates comparison. The “BLB” method adopts the suggested default using groups of size $\lfloor n^{0.7} \rfloor$ and for each group generating a bootstrap sample using `boot.rq.pwxy`, so effectively we are comparing doing many smaller sample size preprocessed bootstraps and then aggregating them, to doing one larger preprocessed bootstrap. Curiously, at least over our problem domain, the timings are almost identical! Of course, this suggests that in a world where one has access to many cores the “BLB” option seems quite promising. If/when I decide to upgrade to a M1 Apple Mini, I may try to revisit this comparison.

21.6 Conclusion

What have we learned? When the sample size is less than about 10,000, the ancient simplex machinery is quite serviceable even for conventional bootstrapping. However, for larger sample sizes it is advantageous to turn to interior point methods especially when one wants solutions for a large number of quantiles, or when bootstrapping is desired. In both cases preprocessing can help to significant speed up the computations. Further exploration of the tuning parameter selection for preprocessing is certainly warranted. When the parametric dimension of the model is large, but the design matrix is sparse as in our experiment for Figure 3, there is a large gain to using the sparse version of the Frisch-Newton method. Unfortunately, there is (as yet) no preprocessed sparse version for multiple taus, or for bootstrapping so this is something that should probably be on the todo list. Finally, there is significant scope for further exploration of first order methods along the lines of the **conquer** package.

Some progress has been made with the introduction of the fortran versions called by `rq.fit.pfnb` and `boot.rq.pwxy`, although this has the obvious downside that it adds to the clutter of options for both fitting and inference. It is difficult with only a handful of experimental settings to be precise about problem dimension boundaries and choice of methods. As always, I would appreciate hearing about user experience, both good and bad. This is main mechanism for improvement.

Computing Sample Quantiles

22.1 Introduction

A classical problem in computational statistics is: how to compute the τ th sample quantile? It seems ridiculously simple: don't you just sort the observations and find the $k = \tau n$ smallest one. Maybe, but not if you are interviewing for a job at Google. Sorting is quite efficient, and can be done in $O(n \log n)$ comparisons, but why would we want to fully sort the observations when we only need to care about a small range of them?

Linear methods began appearing in the 1960s, notably Hoare (1961), an adaptation of the better known “quicksort,” Hoare (1962). An elegant analysis of Eppstein (2007) shows that “quickselect” requires an expected number of comparisons,

$$[2n(1 + \log(n/(n - k)) + 2k \log((n - k)/k))(1 + o(n)),$$

or, in the median, $k = n/2$, worst case, $2n(1 + \log(2) + o(1)) \leq 3.3863n + o(n)$ comparisons.

Floyd and Rivest (1975) proposed an algorithm¹ that improved upon Hoare's “quickselect,” and provided an implementation in Algol 68. Subsequent work by Kiwiel (2005) established an expected performance bound of $n + \min(k, n - k) + O(\sqrt{n})$ for this algorithm. In 1996 I made a very literal translation of the Floyd-Rivest “select” algorithm from Algol to Ratfor, a preprocessor for Fortran, and incorporated it into my Splus package for quantile regression. This implementation involved Fortran that made a recursive call (to itself). This was formally a no-no in Fortran 77, but for a time certain compilers tolerated it. After the transition to R about 2001, at some point compilers on some CRAN test machines began to balk at this recursive feature and I had to remove it from the package. Coincidentally, around this time I had an email exchange with Krzysztof Kiwiel who had been working on “select” and had run across my Fortran version. His paper, Kiwiel (2005) clarified several aspects of the theoretical performance of the “select” algorithm and also provided a Fortran version that circumvented the recursive call. He kindly gave me permission in 2006 to include his

¹ There seems to be some confusion in the literature about the terminology of these early contributions. Hoare's “quickselect” is sometimes referred to as “quickfind,” and an early paper of Tibshirani (2008) misattributes “quickselect” to Floyd and Rivest. Knuth (1998) offers an authoritative account.

code as part of my **quantreg** package. In a somewhat quixotic effort to convince R-core to replace the extant **quantile** function, I wrote interface code to produce all nine varieties of quantiles as described in the R man page for **quantile**. It was eventually judged to be insufficiently faster to justify the switch.²

In Koenker (2020) I recently tried to review the ecosystem of methods in the **quantreg** package for computing quantile regression estimates and evaluating their precision. This rekindled my interest in computation of univariate quantiles, and in particular the weighted quantiles that correspond to solving univariate quantile regression problems that force the fitted quantile regression line to be a ray through the origin. In the remainder of this note I'd like to first review the the classical univariate sample quantile problem, and then briefly turn to the weighted quantile problem.

22.2 Univariate Sample Quantiles

My interface to the Kiwiel code, because in part it accounts for all nine varieties of quantiles, seemed a bit bulky, so I wondered whether something closer to the original implementation of Floyd and Rivest might perform somewhat better. In the interim I had learned that Fortran 90 had reinstated the possibility of recursive calls. So I thought I would experiment with my old literal translation as a warm-up exercise before embarking on the weighted quantile coding that was my ultimate aim. The modification required for this was essentially just to add the word “recursive” before subroutine, The directory **src/ratfor** in the package **quantreg** contains the new version of the code in Ratfor.

In Figure 1 I compare the cpu time required to compute a single quantile for samples of various sizes with the functions, **quantile** from base-R, **kquantile** from the **quantreg** package, and **q489** from the **quantreg** package. The shaded regions represent the band between the first and third quantiles for each method based upon 50 replications and measured by the **microbenchmark** function. Both **quantreg** procedures exhibit a clear advantage over the base-R method with **q489** having a slight advantage due perhaps to the overhead entailed by the more elaborate R interface of the **kquantile** method.

22.3 Weighted Quantiles

In their simplest form weighted quantiles arise when we have repeated values of some observations and would like to use case weights to account for this repetition. More generally, we may wish to consider more general weights for each observation. These weights should be non-negative to preserve the convexity of the underlying optimization problem. One interpretation of the general weighted quantile estimate is to consider the problem,

$$\min_{a \in \mathbb{R}} \sum_{i=1}^n \rho_{\tau}(y_i - x_i a)$$

² It always amazed me that there were nine such varieties to resolve what was, after all, an ambiguity of measure zero. Empson (1930) could only come up with seven.

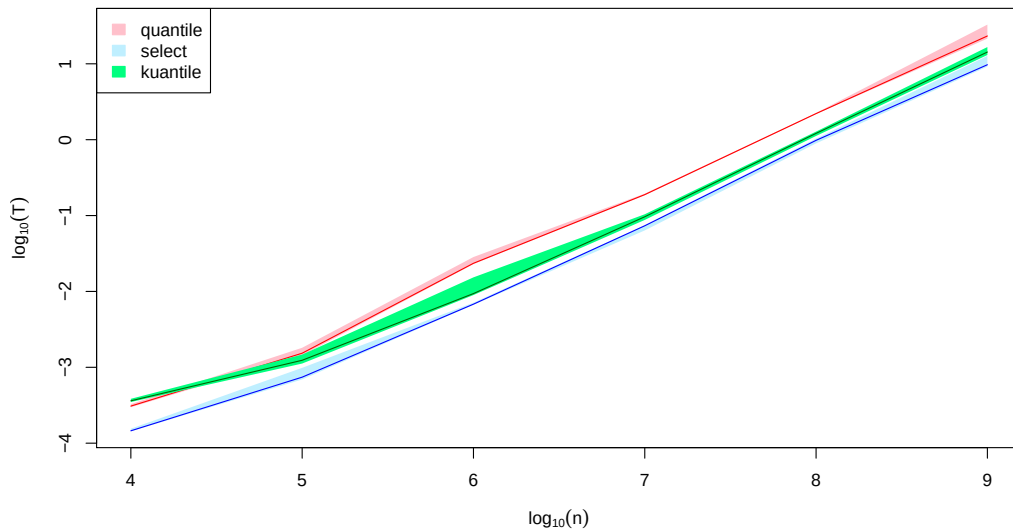


Fig. 22.1: CPU Time (in seconds) for Computing a Single Quantile as a Function of Sample Size

this is sometimes referred to as “regression through the origin” since we seek a scalar slope coefficient to minimize the usual quantile regression objective function. In effect this is a weighted quantile problem with observations $z_i = y_i/x_i$ and weights $w_i = |x_i|$. To solve such problems we need to order the z_i ’s and find the element such that the cumulative sum of the ordered weights exceeds a threshold, a concise implementation of this strategy is given in the following R code.

```
wquantile <- function(x, y, t = 0.5) {
  ord <- order(y/x)
  z <- (y/x)[ord]
  wabs <- abs(x[ord])
  k <- sum(cumsum(wabs) < ((t - 0.5) * sum(x) + 0.5 * sum(wabs)))
  z[k + 1]
}
```

Of course, this requires a full sort of the z_i ’s so there should be a more efficient algorithm that operates like the Floyd-Rivest “select” method. A UK astrophysics project called “Starlink” produced such an open source implementation called KPG1_QNTLx, unfortunately my attempts to link it via R to do comparisons floundered; for sample sizes up to about 1,000,000 it performs quite well, but for larger sample sizes it segfaulted with a “memory not mapped” error. I was not able to track down the source of this error despite attempts

to explore with my limited knowledge of fortran debugging with gdb and valgrind. As an alternative I wrote a version of “select” employing weights that was almost successful; it was relatively quick, didn’t segfault at large sample sizes but unfortunately didn’t quite get the right answer. It was typically only wrong by one or two order statistics, but again I lost patience trying to determine what was going wrong. Intrepid readers, if any, interested in pursuing this are encouraged to take a closer look at the code which will be made available with the pdf of this document.

22.4 Updating Sample Quantiles

Given an sample quantile based on n observations, we might also ask whether there is an efficient way to update our estimate when presented with new observations. This was the first example in the celebrated Robbins and Monro (1951) paper on stochastic approximation. There are several variants of this problem. One approach is closely tied to the Robbins-Monro algorithm and has been investigated by Holst (1985), Toulis, Horel, and Airoidi (2020) and others. Space efficient estimation methods are proposed by Tierney (1983), and Rousseeuw and Bassett (1990). Estimation of quantiles in distributed networks has been explored by Chambers, James, Lambert, and Vander Wiel (2006), and Hammer, Yazidi, and Rue (2019), among others.

22.5 Conclusion

Quantiles play a crucial role in many statistical applications and larger sample sizes inevitably pose new challenges. My own interest in efficient computation of weighted sample quantiles stemmed from an old realization that the bounded variable simplex method of Barrodale and Roberts (1974) for solving ℓ_1 regression problems, and *a fortiori* other quantile regression problems, is essentially gradient descent combined with a step length optimization solving a weighted quantile problem. See Koenker (1996) for further details.

Pava and Pavlova



“Whisk in 1 tsp white wine vinegar.”

23.1 Introduction

Isotonic regression has a long and illustrious history beginning with work in the 1950s by van Eeden and the four B’s as elegantly sketched in de Leeuw, Hornik, and Mair (2009). It was from this paper and the associated R package **isotone** that I first learned about the generalization of the classical “pool adjacent violators algorithm” (PAVA) method of estimation for isotonic mean regression to isotonic median, and quantile regression. In Figure

1 I've illustrated an example where because the conditional distribution of the response, $y|x$ is right skewed, the fitted mean function lies above the fitted median function over most of the domain.

```
require(isotone)
set.seed(1729)
n <- 100
x <- sort(rchisq(n,4)) + 2
y <- log(x) + .1*(log(x))^2 + log(x)*rchisq(n,4)/4
plot(x, y, cex = 0.5, col = "blue")
plotpava <- function(x, col = 1){
  o <- order(x$z)
  xval <- x$o
  yval <- x$y[o]
  lines(xval, yval, type = "S", col = col, lwd = 2)
}
g1 <- gpava(x,y,solver = weighted.fractile, p = 0.5)
g2 <- gpava(x,y)
plotpava(g1, col = 2)
plotpava(g2)
legend("bottomright", c("mean","median"), lwd = 2, col = 1:2)
```

The piecewise constant form of these estimates raised the question: Is there an equivalent formulation of the underlying optimization problem based on total variation penalization? In my experience TV penalization is usually imposed on the derivative, or gradient, of the fitted function, but there is a substantial literature on penalization of the fitted function itself, this leads naturally to fitted functions that are piecewise constant, rather than piecewise linear. Thus, to take the simplest example we might consider the penalized quantile regression problems,

$$\hat{g} = \operatorname{argmin}_{g \in \mathcal{G}} \sum_{i=1}^n \rho_{\tau}(y_i - g(x_i)) + \lambda TV(g).$$

This is closely related to Davies and Kovac (2001) “taut-string” methods except that there the emphasis is on estimation of mean functions, and they have a much more sophisticated approach to a local rather than global penalty parameter, λ . The penalty enforces the piecewise constant form of the solutions since any departure from this class of solutions would inflate the penalty without improving the fidelity. From here, it is tempting to consider imposing the further constraint that $\hat{g}$ be monotone. Since we only need worry about the fitted function values at the observed design points, this entails only the requirement that the (ordered) $\hat{y}_i$'s be increasing, or decreasing.

Fortuitously, this formalism is easily implemented in the **rqss** framework of my **quantreg** package in R, so it is simple to compare solutions with those from the **isotone** package. The only fly in the ointment seems to be what to do with λ ? We don't really want to do any smoothing, so we can just set $\lambda = 0$. I was doubtful that this would work, and assumed at first that I'd have to use some bogus “small” value, but to my surprise, zero seems to be fine.

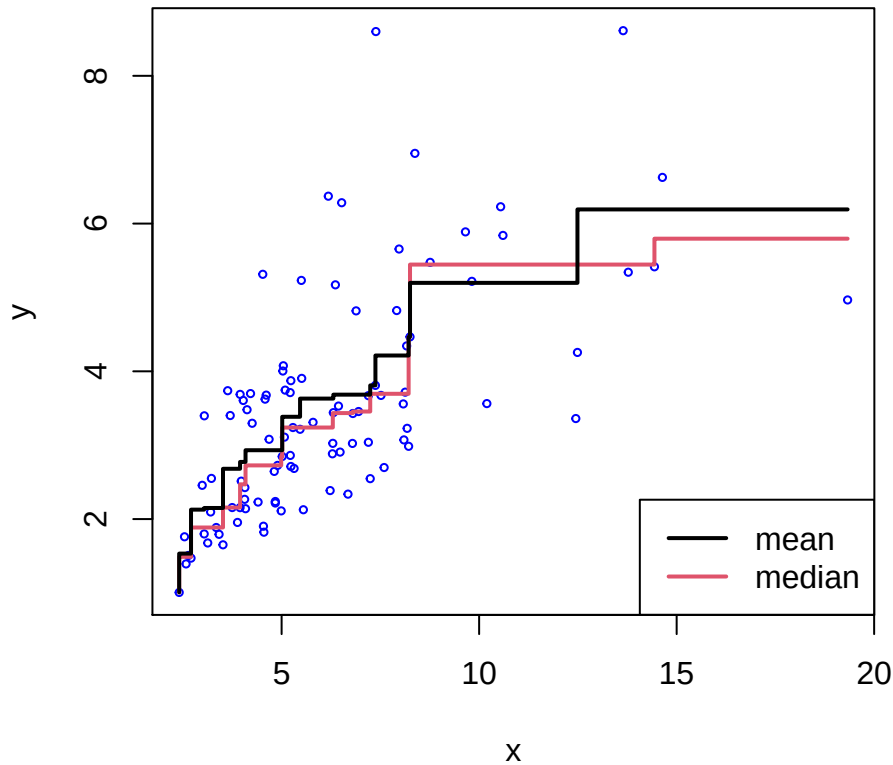


Fig. 23.1: Comparison of mean and median isotonic regression: The response is conditionally rescaled χ^2 and consequently the isotonic mean fit is somewhat above the conditional median fit for this data.

```
set.seed(1729)
n <- 100
x <- sort(rchisq(n,4)) + 2
y <- log(x) + .1*(log(x))^2 + log(x)*rchisq(n,4)/4
plot(x, y, cex = 0.5, col = "blue")
f <- rqss(y ~ qss(x, constraint= "I", lambda = 0, Dorder = 0), tau = .501)
g1 <- gpava(x,y,solver = weighted.fractile, p = .501)
plotpava(g1)
plot(f, add = TRUE, col = 2)
```

```

Rho <- function(u, tau = 0.5) sum(u * (tau - (u < 0)))
R0 <- f$fidelity
R1 <- Rho(g1$y - g1$x, tau = .501)
legend("bottomright", c("isotone", "TV Penalty"), lwd = 2:1, col = 1:2)

```

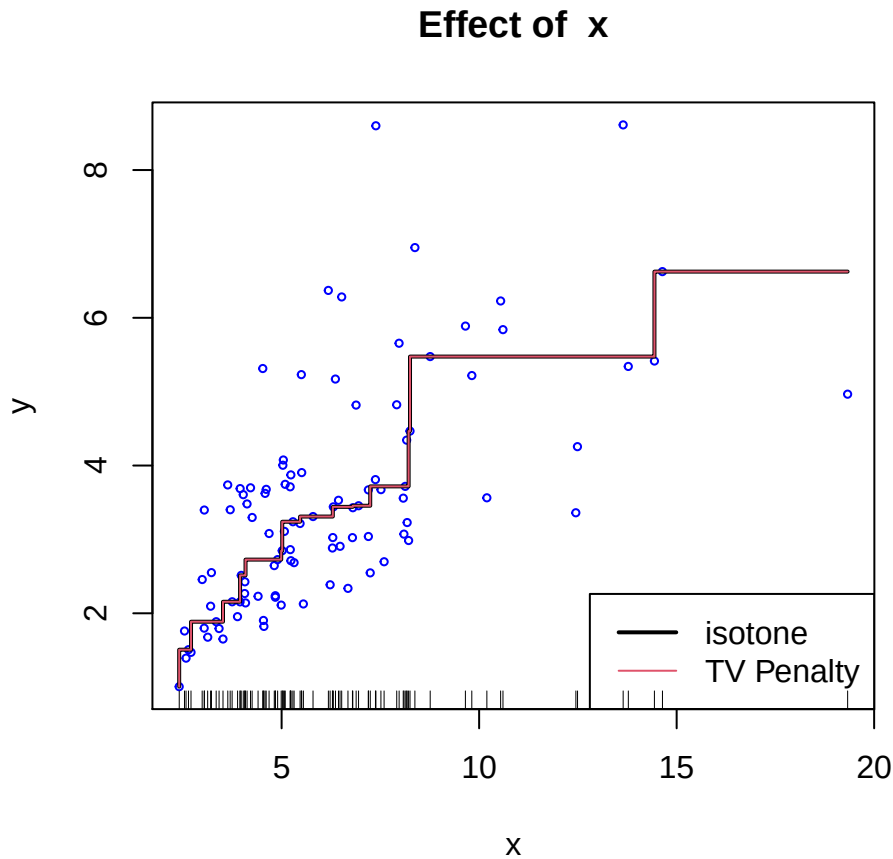


Fig. 23.2: Comparison of median isotonic regression with penalized median regression: The response is the same conditionally rescaled χ^2 . The TV penalized solution overplots the isotonic solution and the realized values of the objective function agree to 11 decimal digits.

Note that in the `qss` term of the `rqss` fitting function I've specified `Dorder = 0` so that the TV penalty acts on g rather than g' . I've also specified `tau = 0.501` which may seem a bit odd, but serves the useful purpose that it avoids the multiple solutions that

arise when `tau = 0.5` is specified. It should be noted that the same solutions can also be obtained by specifying `Dorder = 1` since the monotonicity enforces the same piecewise constant solutions.

In retrospect it is hardly surprising that the two problems produce the same solution, although I still find it remarkable that the two rather different algorithms, the active-set algorithm from the **isotone** package and the sparse Frisch-Newton interior point algorithm from **quantreg** produce solutions whose objective functions agree to eleven decimal digits.

23.2 Conclusion

Arguably, an advantage of the penalized version is that it fits nicely into the additive modeling framework of the **rqss** function, so one could easily incorporate additional covariate effects. As always with shape constrained estimation, inference is a challenging subject. One might hope that having yet another perspective on the formulation of such problems might eventually shed some new light on these questions as well.

Bradley-Terry and the Ranking Lasso

1

24.1 Introduction

In accordance with Stigler’s law of eponymy, the Bradley and Terry (1952) model for ranking competitors based on paired comparisons was first proposed by Zermelo (1929) for rating chess players.² In the simplest setting we have n players of ability, $\alpha_1, \alpha_2, \dots, \alpha_n$ who meet in pairs; the probability that a player i defeats a player j is given by.

$$\pi_{ij} = \alpha_i / (\alpha_i + \alpha_j).$$

With a sufficient accumulated history of play, the α ’s can be estimated by maximum likelihood. Since n can be rather large there is an inducement to regularize and many proposals have been made. Among these the proposal of Masarotto and Varin (2012) seems particularly intriguing since it purports to group the estimated abilities into a few equivalence classes using a variant of ℓ_1 penalization. The ranking penalty has already been used earlier for clustering by Hocking, Joulin, Bach, and Vert (2011), and in the context of quantile regression panel models by Gu and Volgushev (2019).

24.2 Some Convex Optimization

It is convenient to reparameterize abilities so $\theta_i = \log \alpha_i$ and π_{ij} , becomes,

$$\pi_{ij} = \frac{1}{1 + \exp(-(\theta_i - \theta_j))}$$

and to write the (logistic) log likelihood for m binary outcomes, $y_1, y_2, \dots, y_m$, as,

¹ Thanks to Jiaying Gu, Ryan Tibshirani, Addison Hu and especially Michal Adamaszek for helpful suggestions. They bear no responsibility for errors, egregious or otherwise.

² Useful background on all this is provided by David (1988), Cattelan (2012) and Glickman (2013).

$$\ell(\theta|y) = \sum_{k=1}^m y_k \log(h_\theta(x_k)) + (1 - y_k) \log(1 - h_\theta(x_k))$$

where $h_\theta(x) = 1/(1 + \exp(-\theta^\top x_k))$, x_k is an n vector with i th element 1, and j th element -1, and other elements 0. The “ranking lasso” of Masarotto and Varin (2012) proposes to estimate the vector, $\theta = (\theta_1, \dots, \theta_n)$ by minimizing,

$$R(\theta) = -\ell(\theta|y) + \lambda \|D\theta\|_1,$$

where $\|D\theta\|_1 = \sum_{i < j} |\theta_i - \theta_j|$. There is an extensive literature on ℓ_1 penalized logistic regression, notably Koh, Kim, and Boyd (2007), but typically such methods are designed to shrink some or all of the coefficients toward zero. In contrast the ranking lasso tries to reduce *differences* in the coefficients and thereby tries to identify groups of coefficients that take the same value. Masarotto and Varin (2012) propose an EM approach to minimizing $R(\theta)$; I would like to explore an alternative interior point approach that relies on Andersen (2010).

Since R is a convex function the problem can be reformulated to minimize a linear function subject to convex cone constraints. This can be written as,

$$\min \left\{ \sum_{k=1}^m t_k + \lambda r \mid t_k \in \mathcal{T}_k, k = 1, \dots, m, r \geq \|D\theta\|_1 \right\},$$

where $\mathcal{T}_k = \{t \mid t \geq y_k \log(h_\theta(x_k)) + (1 - y_k) \log(1 - h_\theta(x_k))\}$. In Mosek terminology the $\mathcal{T}_k$ are exponential cones. An exponential cone in $\mathbb{R}^3$ is the closure of the set of points satisfying,

$$K = \{x \in \mathbb{R}^3 \mid x_1 \geq x_2 \exp(x_3/x_2), x_1 > 0, x_2 > 0\}$$

or equivalently,

$$K = \{x \in \mathbb{R}^3 \mid x_3 \leq x_2 \log(x_1/x_2), x_1 > 0, x_2 > 0\}.$$

For the logistic function $f(x) = \log(1 + e^x)$ its epigraph constitutes the points (t, x) such that $t \geq \log(1 + e^x)$, or equivalently, $e^{x-t} + e^{-t} \leq 1$, so we have the cone,

$$C = \{u + v \leq 1 \mid (u, 1, x - t) \in K, (v, 1, -t) \in K\}$$

from which we can construct all the t_k constraints. The penalty term involving only linear inequality constraints poses no further problems. In Mosek the ℓ_1 norm can be treated as a special case of quadratic cone constraints since $t \geq \sum |x_i|$ can be formulated as $\{(z_i, x_i) \in Q^2, \sum z_i = t\}$ with $Q^2 = \{x \in \mathbb{R}^2 \mid x_1 \geq \sqrt{x_2^2}\}$. So this adds n quadratic cone constraints with the role of x_1 played by r and the x_2 's by the elements of the vector $D\theta$. In practice it seems simpler and may be faster to formulate the ℓ_1 constraint as linear inequality constraints as illustrated in the next section.

An advantage of Mosek's insistence that problems be formulated with linear objective functions and cone constraints is that their dual formulation is straightforward. Given a primal problem,

$$\min \{c^\top x \mid Ax = b, x \in K\} \tag{P}$$

then we have dual problem,

$$\min \{b^\top y \mid c - A^\top y \in K^*\} \tag{D}$$

where K^* is the dual cone of K .

24.3 Implementation

After embarking on this mini-project I discovered that the documentation for the Mosek APIs for C, python and java all contained code for an ℓ_2 , aka ridge, penalized version of logistic regression. So the remaining task seemed to be to translate one of these implementations into R and replace the penalty with the ranking lasso penalty. Unfortunately, this proved to be more difficult than it first appeared, and closer reading of Koh, Kim, and Boyd (2007) suggested that perhaps Mosek wasn't an ideal vehicle for future development anyway. At that point I toyed with the idea of making an implementation based on the C code from Koh, Kim, and Boyd (2007). If I were more conversant with C this probably would have been a reasonable strategy, but in the end I decided to take the coward's way out and construct something based on code that I already had considerable experience with – to wit quantile regression. Further details about this adventure are provided in the Appendix.

Fortunately, just as my initial experimentation with this median regression implementation was drawing to a close, I received a personal communication Adamaszek (2021) with illustrative Rmosek code for the ℓ_2 penalized version of the logistic regression problem. From there it was easy to implement an alternative version with an ℓ_1 penalty on $D\theta$. The implementation in R using the Rmosek package, Friberg (2012), is quite concise, further bells and whistles, notably weighting effects can be easily added.

```
# Regularized logistic regression
# L2 version by Michal Adamaszek Aug 16, 2021
# See: https://groups.google.com/g/mosek/c/T2MYKTc8uD4
# L1 version slightly modified by Roger Koenker
# Problem: min{ -l(theta | X, y) + lambda || D theta ||_1 }
RLR <- function(X, y, D, lambda)
{
  prob <- list(sense="min")
  n <- nrow(X)
  p <- ncol(X)
  m <- nrow(D)

  # Variables: r, theta(d), u(m), t(n), z1(n), z2(n)
  prob$c <- c(lambda, rep(0,p+m), rep(1, n), rep(0,n), rep(0,n))
  prob$bx <- rbind(rep(-Inf, 1+p+m+3*n), rep(Inf, 1+p+m+3*n))

  # l1 constraints
  A1 <- rbind(cbind(0, D, diag(m), Matrix(0, m, 3*n)),
             cbind(0, -D, diag(m), Matrix(0, m, 3*n)),
             c(1,rep(0,p),rep(-1,m),rep(0, 3*n)))
  # z1 + z2 <= 1
  A2 <- sparseMatrix( rep(1:n, 2),
                     c((1:n)+1+p+m+n, (1:n)+1+p+m+2*n), x = rep(1, 2*n))
  prob$A <- rbind(A1,A2)
  prob$bc <- rbind(c(rep(0,1+2*m), rep(-Inf, n)),
```

```

                                c(rep(Inf,1+2*m),rep(1, n)))

# (z1(i), 1, -t(i)) \in \EXP,
# (z2(i), 1, (1-2y(i))*X(i,) - t(i)) \in \EXP
FE <- Matrix(nrow=0, ncol = 1+p+m+3*n)
for(i in 1:n) {
  FE <- rbind(FE, sparseMatrix( c(1, 3, 4, rep(6, p), 6),
                                c(1+p+m+n+i, 1+p+m+i, 1+p+m+2*n+i, 2:(p+1), 1+p+m+i),
                                x = c(1, -1, 1, (1-2*y[i])*X[i,], -1),
                                dims = c(6, 1+p+m+3*n) ) )
}
gE <- rep(c(0, 1, 0, 0, 1, 0), n)

prob$F <- FE
prob$g <- gE
prob$cones <- matrix(list("PEXP", 3, NULL), nrow=3, ncol=2*n)
rownames(prob$cones) <- c("type","dim","conepar")

# Solve, no error handling!
r <- mosek(prob, list(verbose = 0))

# Return theta
r$sol$itr$xx[2:(p+1)]
}

```

A simple test problem using the classical $\|\theta\|_1$ penalty produces a typical lasso shrinkage plot.

```

require(Matrix)

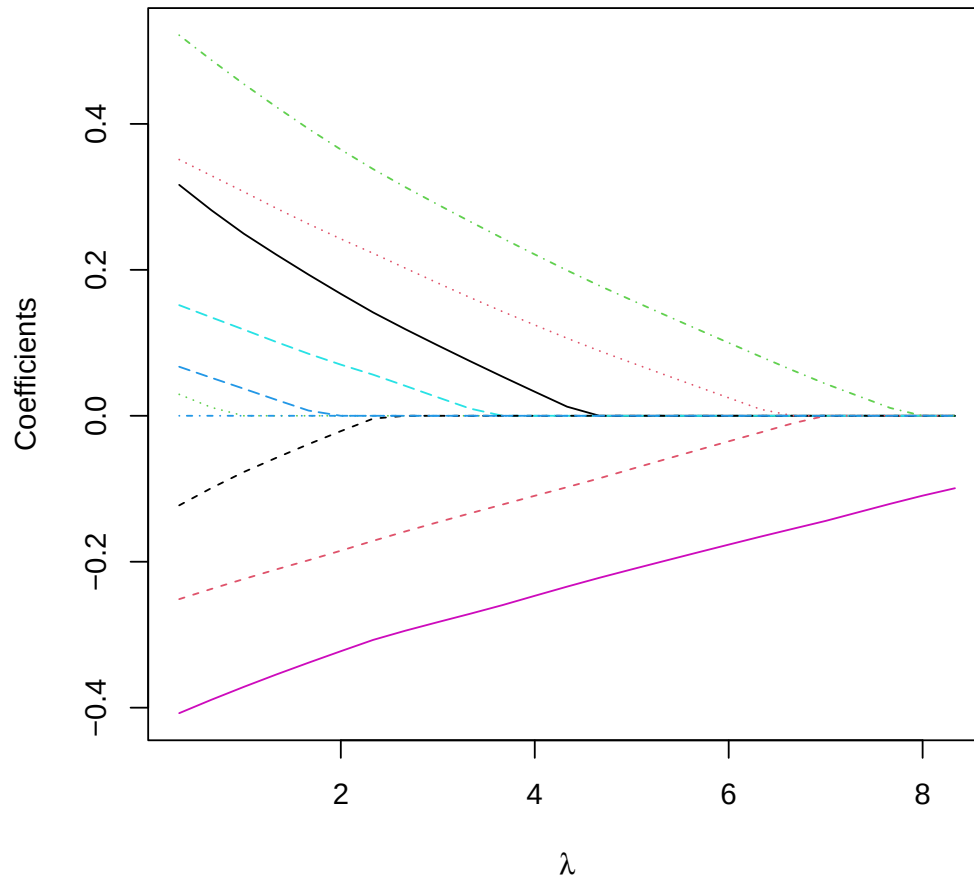
## Loading required package: Matrix
require(Rmosek)

## Loading required package: Rmosek

set.seed(1729)
n = 100
p = 10
X <- matrix(rnorm(n*p),n,p)
y <- sample(0:1, n, replace = TRUE)
lambdas <- 1:25/3
B <- matrix(0, length(lambdas),10)
for(i in 1:length(lambdas))
  B[i,] <- RLR(X,y,diag(p),lambdas[i])
matplot(lambdas, B, type = "l", xlab = expression(lambda), ylab = "Coefficients")
title("A Logistic Regression Lasso Plot")

```

A Logistic Regression Lasso Plot



In our intended applications the matrix, D , would be slightly more complicated as illustrated in the median regression implementation in the Appendix.

24.4 Regularized Median Logistic (Binomial) Regression

What you might ask does ℓ_1 regularized logistic regression have to do with quantile regression? The obvious answer is: When you have a hammer everything looks like a nail. But two factors conspired to make the RLR problem look more like a nail than one might think. The first factor was that I had no intention of dealing with applications that involved the usual binary response model in its pure Boolean form, rather my applications were all focused on models for which I could aggregate responses into binomial form with relatively large cell

counts. Thus, the idea of exploiting the Gaussian approximation to the binomial offered an escape route from Mosek's exponential cones. After all, even in my introductory applied econometrics course hadn't I taught that with frequency data one could get by estimating models with weighted least squares? See e.g. Koenker (2016a). But why least squares, when it would be much easier to attach the ℓ_1 penalty to median regression? All of a sudden, we have a simple data augmentation formulation with a very sparse design structure that is ideally suited to the algorithms already available in the R package **quantreg**.

The canonical implementation I want to consider looks like this,

$$\min_{\theta} \left\{ \sum_{i=1}^n w_i \rho_{\tau}(y_i - x_i^{\top} \theta) + \lambda \|D\theta\|_1 \right\},$$

where w_i 's denote weights that would be typically the square root of the weights one would use in the associated least squares problem. The matrix D might also incorporate weights of the form used in Masarotto and Varin (2012) designed to shrink coordinates more when their unconstrained estimates was already close to zero. The intention is to restrict applications to $\tau = 1/2$, but you never know when the temptation to stray from this intention might arise.

24.5 An Illustrative Example

To illustrate the approach consider the simple baseball example of Turner and Firth (2012) based on results from the 1987 season involving the 7 teams in the eastern division of the American League. Each pair of teams play 13, 6 at home and 7 away, or vice versa. The unconstrained Bradley Terry model can be estimated easily in R with the following code:

```
library(BradleyTerry2)
MBTm <- BTm(cbind(home.wins, away.wins), home.team, away.team,
             data = baseball, id = "team")
MBTm$coef

##      teamBoston teamCleveland  teamDetroit teamMilwaukee  teamNew York
##      1.1076977    0.6838528    1.4364084    1.5813559    1.2476178
##      teamToronto
##      1.2944851
```

The fitting is accomplished with the aid of base R's `glm.fit` function and uses the default `family = binomial` option with the logistic link function. By default the first team is assigned rating zero. The **BTm** package is somewhat opaque when revealing how the model is actually constructed and fed into the `glm.fit` function so in an effort to demystify things a bit, one can also do this:

```
X <- model.matrix(~home.team-1, data = baseball) -
      model.matrix(~away.team-1, data = baseball)
Y <- with(baseball, cbind(home.wins, away.wins))
```

```
Mglm <- glm(Y ~ X[,-1] - 1, family = binomial)
Mglm$coef

##      X[, -1]home.teamBoston X[, -1]home.teamCleveland  X[, -1]home.teamDetroit
##                1.1076977                0.6838528                1.4364084
## X[, -1]home.teamMilwaukee  X[, -1]home.teamNew York  X[, -1]home.teamToronto
##                1.5813559                1.2476178                1.2944851
```

Note that we have dropped the first column of the X matrix and removed the intercept in the formula of the `glm` call. Except for different labeling of the coefficients the fits seem to agree. An unweighted, unregularized median regression version looks like this:

```
logit <- function(p, eps = 0.01) {
  p <- pmax(eps, pmin(1-eps,p))
  log(p/(1-p))
}
y <- logit(Y[,1]/apply(Y,1,sum))
Mrq <- rq(y ~ X[,-1] - 1)
Mrq$coef

##      X[, -1]home.teamBoston X[, -1]home.teamCleveland  X[, -1]home.teamDetroit
##                1.3217558                0.9162907                1.6094379
## X[, -1]home.teamMilwaukee  X[, -1]home.teamNew York  X[, -1]home.teamToronto
##                1.8971200                1.6094379                1.3217558
```

The usual estimate of the variance of $\logit$ of $\hat{p}_i$ is $1/(n_i p_i (1 - p_i))$ so we should weight the median regression objective function by $\sqrt{n_i p_i (1 - p_i)}$.

```
w <- Mrq$fitted
w <- sqrt(apply(Y,1,sum) * exp(w)/(1 + exp(w))^2)
Mwrq <- rq(y ~ X[,-1] - 1, weights = w)
Mwrq$coef

##      X[, -1]home.teamBoston X[, -1]home.teamCleveland  X[, -1]home.teamDetroit
##                1.0340738                0.6286087                1.6094379
## X[, -1]home.teamMilwaukee  X[, -1]home.teamNew York  X[, -1]home.teamToronto
##                1.6094379                1.3217558                1.0340738
```

The ranking, or grouping, penalty is easily implemented as a data augmentation device, we simply introduce pseudo observations corresponding to the penalty terms: Since the first team is assigned rating zero, the penalty matrix includes an identity term that shrinks all the remaining coefficients towards this value, as well as rows that shrink the remaining coefficients toward one another. It is conventional to weight the coordinates of the penalty contribution by the reciprocals of their unconstrained estimates, but these weights can be introduced in the call to `rq` as indicated above. The λ that controls the global shrinkage will be incorporated into the D matrix as shown below.

```

teams <- factor(levels(baseball[,2])[-1])
pairs <- t(combn(teams,2))
D <- model.matrix(~ pairs[,1] + 0) - model.matrix(~ pairs[,2] + 0)
D <- rbind(diag(length(teams)),D)
lambda = 0.1
penw <- abs(D %*% Mwrq$coef) + 1
yp <- c(y, rep(0, length(penw)))
Xp <- rbind(X[,-1], lambda * D)
wp <- c(w, 1/penw)
Mprq <- rq(yp ~ Xp - 1, weights = wp)
Mprq

## Call:
## rq(formula = yp ~ Xp - 1, weights = wp)
##
## Coefficients:
##      Xphome.teamBoston Xphome.teamCleveland Xphome.teamDetroit
##           1.0340738           0.6931472           1.6094379
## Xphome.teamMilwaukee Xphome.teamNew York Xphome.teamToronto
##           1.6094379           1.3217558           1.0340738
##
## Degrees of freedom: 63 total; 57 residual

```

In this example, two of the proposed penalty weights were zero, so I've introduced a arbitrary additive factor of one to all the weights. With a $\lambda = 0.1$ we get a solution that ranks Detroit and Milwaukee as equivalent and Boston and Toronto as equivalent. Of course, as usual we do not have a principled way to choose λ . Further experimentation with this is clearly needed.

24.6 Regularized Logistic Binomial Median Regression

Assembling the foregoing code we can construct a unified function for future reference.

```

RLBMR <- function(D, lambda = 1){
  # D is a data.frame with columns W,L,T1,T2
  X <- model.matrix(~ T1 - 1, data = D) - model.matrix(~ T2 - 1, data = D)
  Y <- with(D,cbind(W, L))
  n <- nrow(X)
  M <- glm(Y ~ X[,-1] - 1, family = binomial)
  logit <- function(p, eps = 0.01) {
    p <- pmax(eps, pmin(1-eps,p)) # Somewhat Kludgy
    log(p/(1-p))
  }
  y <- logit(Y[,1]/apply(Y,1,sum))
}

```

```

M <- rq(y ~ X[, -1] - 1)
w <- M$fitted
w <- sqrt(apply(Y, 1, sum) * exp(w) / (1 + exp(w))^2)
M <- rq(y ~ X[, -1] - 1, weights = w)
teams <- factor(levels(D$T2)[-1])
pairs <- t(combn(teams, 2))
P <- model.matrix(~ pairs[, 1] - 1) - model.matrix(~ pairs[, 2] - 1)
P <- rbind(diag(length(teams)), P)
penw <- abs(P %*% M$coef) + 1 # Very kludgy
yp <- c(y, rep(0, length(penw)))
Xp <- rbind(X[, -1], lambda * P)
wp <- c(w, 1/penw)
f <- rq(yp ~ Xp - 1, weights = wp)
coef <- c(0, f$coef)
names(coef) <- levels(D$T1)
list(coef = coef, resid = f$resid[1:n], lambda = lambda)
}

```


Evaluating Covid Quantile Forecasts

25.1 Introduction

Motivated by some email conversations with Ryan Tibshirani last summer, I've been curious about how Nick Reich's Forecast Hub <https://covid19forecasthub.org> creates their ensemble forecasts and how they are evaluated. I'll attempt to describe evaluation first and then try to deal with ensembles.

25.2 Scoring Rules for Quantile Forecasts

There is quite an extensive emerging literature on forecast evaluation heavily influenced by Gneiting and Raftery (2007), which tries to come to grips with something more than point forecasts and the dreaded mean squared error. Density forecasts are now commonplace in econometrics, epidemiology and other fields, so one might well ask: Given a forecast density, f , and a single observation, y , how should I evaluate the forecast? In Gneiting-Raftery terminology a strictly proper scoring rule for a density forecast is one that would induce the forecaster to report the true density presuming that it were known. Thus, for example, the logarithmic score,

$$\log S(f, y) = \log f(y)$$

is strictly proper since it is minimized when the true density is g , by choosing $f = g$. Unless one would like to restrict attention to parametric models, such scoring rules impose a rather heavy burden on the forecaster since they require that f be specified in its entirety. A much simpler strictly proper scoring rule is the quantile rule,

$$S_\tau(f, y) = \rho_\tau(y - Q_f(\tau))$$

where $Q_f(\tau)$ is τ th quantile of the f distribution and ρ_τ is the usual check function. This is obviously minimized by choosing f so it has “right” τ th quantile. Of course one quantile doesn't provide a very full description of the whole distribution, so it is natural to consider combining several such scoring rules. In particular, we can construct an interval scoring rule,

$$IS_\alpha(f, y) = \rho_\alpha(y - Q_f(\alpha)) + \rho_{1-\alpha}(y - Q_f(1 - \alpha))$$

as depicted in Figure 1 for $\alpha = 0.1$ and interval limits $[-1, 1]$. For $y \in [-1, 1]$ the score is 0.1, and then increases linearly for y outside this interval. Similar scoring functions can be built by concatenating several interval scoring rules.

The Forecast Hub protocol is that those submitting forecasts for consideration should submit a point estimate of the median, μ , and $K = 11$ prediction intervals for each region and forecast horizon with nominal coverages, $1 - \alpha_k$, and $\alpha_1 = 0.02, \alpha_2 = 0.05, \alpha_3 = 0.10, \dots, \alpha_K = 0.90$. An aggregated score is then constructed as described Bracher, Ray, Gneiting, and Reich (2021),

$$WIS_\alpha(f, y) = \left[w_0 |y - \mu| + \sum_{k=1}^K w_k IS_{\alpha_k}(f, y) \right] / (K + 1/2)$$

with suggested weights, $w_0 = 1/2$ and $w_k = \alpha_k/2$ for $k = 1, \dots, K$. This weighted interval scoring rule can be viewed as a measure of distance between the predictive distribution and distribution of the (random variable) y . The red curve superimposed in the figure illustrated one such rule based on five distinct intervals. It approximates the “continuous ranked probability score,”

$$CRPS(F, y) = \int (F(x) - \mathbb{1}(x \geq y))^2 dx,$$

and can be conveniently decomposed to investigate aspects of the forecast distribution that are unsatisfactory. Bracher, Ray, Gneiting, and Reich (2021) suggest several graphical devices for this purpose.

25.3 Ensemble Forecasting

Over the course of the pandemic the Covid-19 Forecast Hub team has received forecasts in the format specified above from groups of researchers around the globe. In the last week of November, 2021, for example, 50 forecasts were submitted. These can be individually evaluated, but a more challenging and potentially rewarding task is to combine these forecasts into an ensemble forecast that reflects some sort of consensus forecast that can then be promulgated by the CDC. Since forecasts are reported for cases, hospitalizations and deaths at the county, state and national level at one and two week horizons. This represents an enormous data management problem even before any decisions are taken on designing an aggregation mechanism.

Various forms of model selection, model averaging and related ensemble forecasting methods have proliferated, but it seems fair to say that there is no agreed upon general strategy. Over the course of the pandemic several methods have been employed:

- From April 13 to July 21 2020, the ensemble simply computed sample means of the predicted quantiles for each quantile for all eligible models at each location,
- From July 28, sample means were replaced by sample medians.

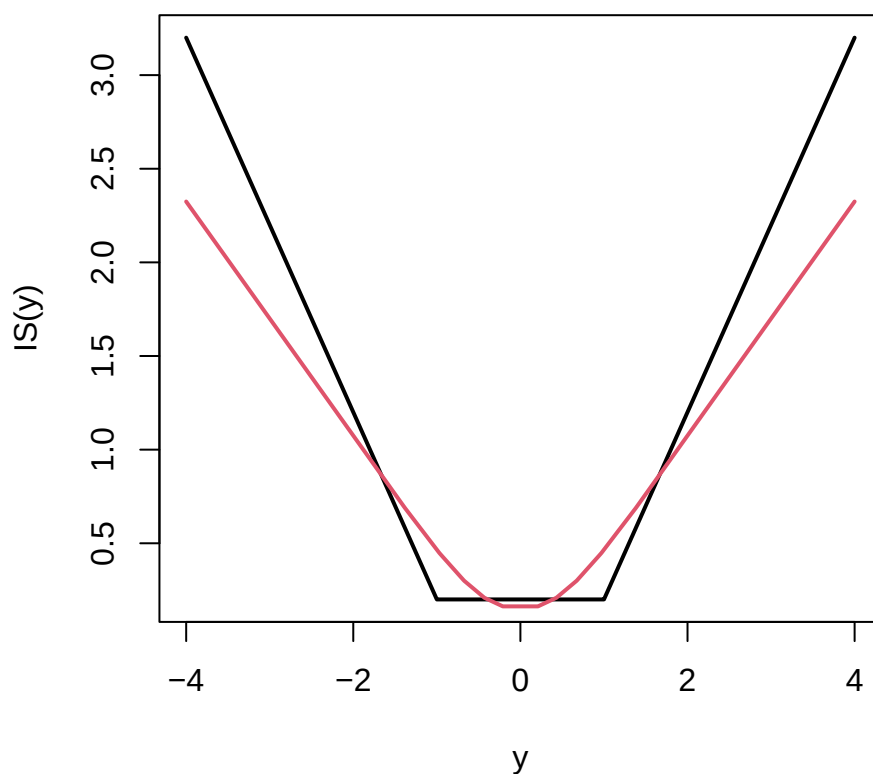


Fig. 25.1: Interval Score Function

- From November 15 2021 the ensemble was based on the ten best model submissions in the prior 12 weeks evaluated by their weighted interval score (WIS), with models with better WIS over this period receiving more substantial weight.

See <https://covid19forecasthub.org/doc/ensemble/> for further details.

One can easily imagine other ensemble aggregation methods, but at the scale of the covid forecasting effort it is obviously necessary to exercise considerable restraint before leaping into new procedures. One can view the weighted WIS procedure currently in use as a compromise between the earlier sample mean and sample median procedures. I was motivated to look into this again by the paper of Yao, Vehtari, Simpson, and Gelman (2018) that reviews the Wolpert (1992) and Breiman (1996) notions of “stacking” from a Bayesian perspective. Before delving into modeling entire distributions, I will try to briefly describe

my understanding of how stacking works in conventional regression, point-forecasting applications.

25.3.1 Stacking for ordinary regression

Consider the standard regression setting in which we see data consisting of n pairs, (y_i, x_i) . We have K candidate models $f_k : k = 1, \dots, K$. For each model we can compute estimates $\hat{f}_k^{(-i)}(x_i)$ as the leave-one-out prediction of the i th observation from the k th model when we omit the i th observation. Now we solve, for weights $\omega \in \Omega \subset \mathbb{R}^K$,

$$\hat{\omega} = \operatorname{argmin} \sum_{i=1}^n \left(y_i - \sum_{k=1}^K \omega_k \hat{f}_k^{(-i)}(x_i) \right)^2,$$

In the Breiman formulation Ω is K simplex, and the positivity constraint helps to ensure that there is a unique solution, and that poor models receive weight zero. Positivity constraints like this also encourage a form of sparsity by this elimination of poor models in the spirit of Breiman's non-negative garotte. Stacking typically performs better from a predictive vantage point than other forms of model averaging. However, as Yao et al remark in its original point estimation form it is not well suited to Bayesian settings where the entire posterior is targeted. To rectify this oversight, Yao et al consider several scoring rules sanctioned by Gneiting and Raftery (2007), but curiously not the WIS rule. I attribute this to the fact that being Bayesian they are addicted to parametric models. Instead they focus on the logarithmic scoring rule that keeps things in the KL playground. This is probably convenient as long as the models under consideration are all parametric, but my intention is to play in the quantile regression sandbox, for which the WIS rules seem opportune.

25.3.2 Stacking for Quantile Models

Leave-one-out fitting for quantile regression isn't enough of a perturbation to accomplish anything useful. Leave- m -out methods might serve better, but are computationally costly. Fortunately, there is no need to evaluate within sample fit for each of the constituent models at the Forecast Hub. This might be prudent for each submitted model *before* submission, but at the Hub there it is only necessary to aggregate based on prior out-of-sample performance of the models, for example by their *WIS*. Of course there are components being forecast, several time horizons so how this should be done is still problematic. This is all left for another day.

Denoising the Banana

26.1 Introduction

Sometime in the early 1990's I made my last trip to Bell Labs in Murray Hill to give a talk in the Statistics Department. While there I spoke with Don X. Sun about a project that he had been working on to identify fruits and vegetables as they came down the conveyor belt in a supermarket checkout station. One way to look at this problem was to try to estimate the outer envelope of the object and this seemed to be something that quantile regression might be deployed to do. I continued to think about this problem periodically. It was tentatively called "How to recognize a banana." Whenever I saw a new idea about multivariate quantiles I would ask, "Can you estimate a banana?" This was a non-trivial test case since it involved a non-convex object, and required rather strong equivariance properties. I think that it is fair to say that none of the quantile regression approaches were a resounding success.

26.2 This is not a banana

Then a few days ago I discovered a wonderful new paper, Soloff, Guntuboyina, and Sen (2025), that I should have seen much earlier since it appeared on arXiv last September. It provides a very thorough analysis of the NPMLE for multivariate heteroscedastic Gaussian mixtures. An example illustrating their approach appears early on in the paper: Gaussian observations with means on a circle of radius 2 are generated, and the bivariate NPMLE estimates the mixing distribution, G , and is used as a "plug-in" prior to denoise the observations delivering a very impressive posterior mean fit closely clustered around the original circle. More to the point the fit is nearly as good as an oracle fit that knows the empirical distribution of the true means. For amusement I thought I should try this on my ancient banana problem, so I coded a simplistic version of our usual REBayes style function to fit bivariate Gaussians and cooked up a test problem.

Figure 26.1 shows a somewhat artificial banana surrounded by some black dots. The latter are used as centers for 1000 observations appearing in Figure 26.2. Now, given the noisy red points we can estimate the mixing distribution and use it to compute posterior

means for each of the observed red points. These shrunken (posterior mean) points are depicted in Figure 26.3 in blue.

```
> library(png)
> require(mvtnorm)
> R <- readPNG("banana.png")
> R <- as.raster(R)
> plot(1:10, 1:10, col = 0, xlab = "", ylab = "")
> rasterImage(R, 1, 1, 10, 10)
> load("banana.Rda") # centers via locator(30)
> points(banana, pch = 19)

> rbanana <- function(n, b = banana, sd = 0.25){
+   k <- sample(1:length(b$x), n, replace = TRUE)
+   t <- matrix(rnorm(2 * n, c(b$x[k], b$y[k]), sd = sd), n, 2)
+   dimnames(t)[[2]] <- c("x", "y")
+   class(t) <- "banana"
+   t
+ }
> B <- rbanana(1000)
> plot(1:10, 1:10, col = 0, xlab = "", ylab = "")
> points(banana, pch = 19)
> points(B, cex = .3, col = 2)

> plot(1:10, 1:10, col = 0, xlab = "", ylab = "")
> points(banana, pch = 19)
> points(B, cex = .3, col = 2)
> points(z$du, z$dv, cex = 0.3, col = 4)
```

This is obviously just a simplistic imitation of the toy example in Soloff, Guntuboyina, and Sen (2025). As usual with bivariate gridding it is rather awkward to code and at least in the present implementation in R, quite slow. This is primarily due to the apply step to evaluate the bivariate Gaussian density, the optimization is actually very quick. The code should also be generalized to accommodate heteroscedastic noise, which has not yet been implemented in the R version. Finally, this is still not addressing the original objective of estimating the outer envelope, it only targets the conditional mean of the banana. La lutte continué.

26.3 R Code for Bivariate Gaussian Mixtures

```
> GL2mix

function(X, S, u = 30, v = 30, eps = 1e-04, ...){
  # Should be extended to heteroscedastic S
  n <- nrow(X)
```

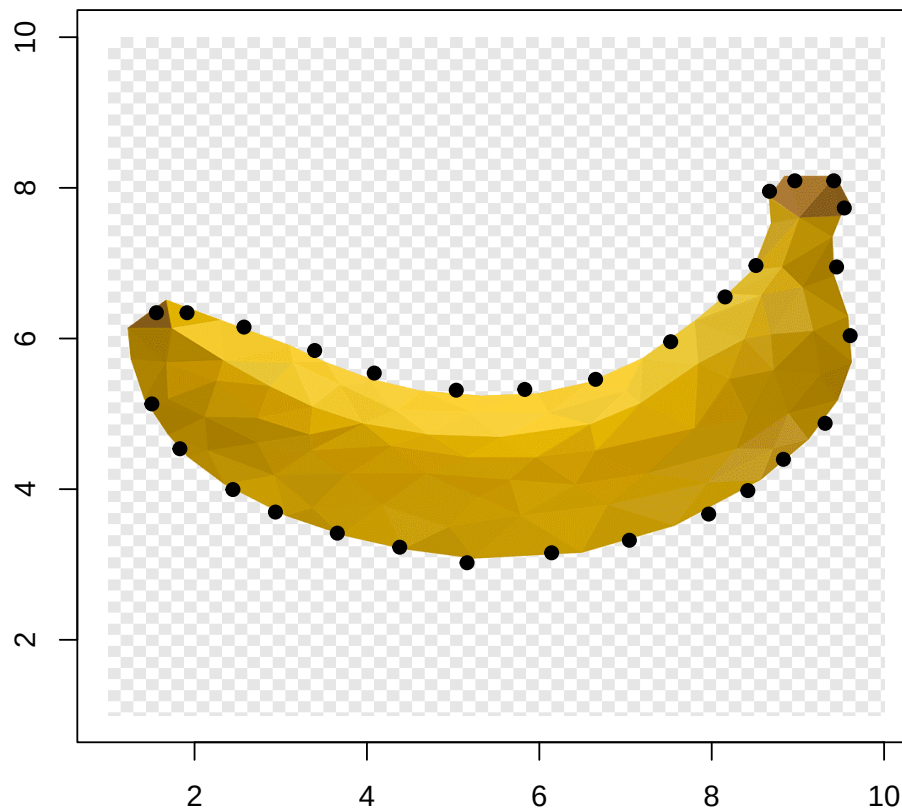


Fig. 26.1: This is not a banana.

```

w <- rep(1/n,n)
if (length(u) == 1)
  u <- seq(min(X[,1]) - eps, max(X[,1]) + eps, length = u)
if (length(v) == 1)
  v <- seq(min(X[,2]) - eps, max(X[,2]) + eps, length = v)
pu <- length(u)
du <- rep(1, pu)
pv <- length(v)

```

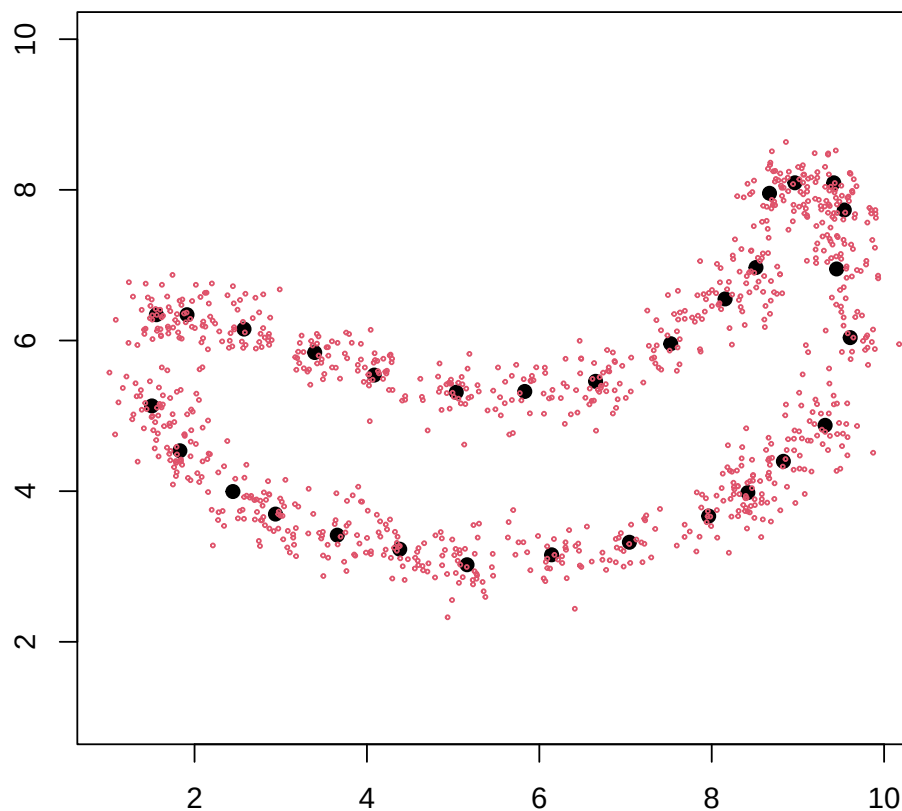


Fig. 26.2: Noisy banana with centers

```

dv <- rep(1, pv)
UV <- expand.grid(u,v)
A <- array(0,c(n,p,2))
A[,1] <- outer(X[,1], UV[,1], "-")
A[,2] <- outer(X[,2], UV[,2], "-")
A <- apply(A, 1:2, function(x) dmvnorm(x, sigma = diag(2)))
duv <- as.vector(kronecker(du, dv))
f <- KWDual(A,duv, w, ...)

```

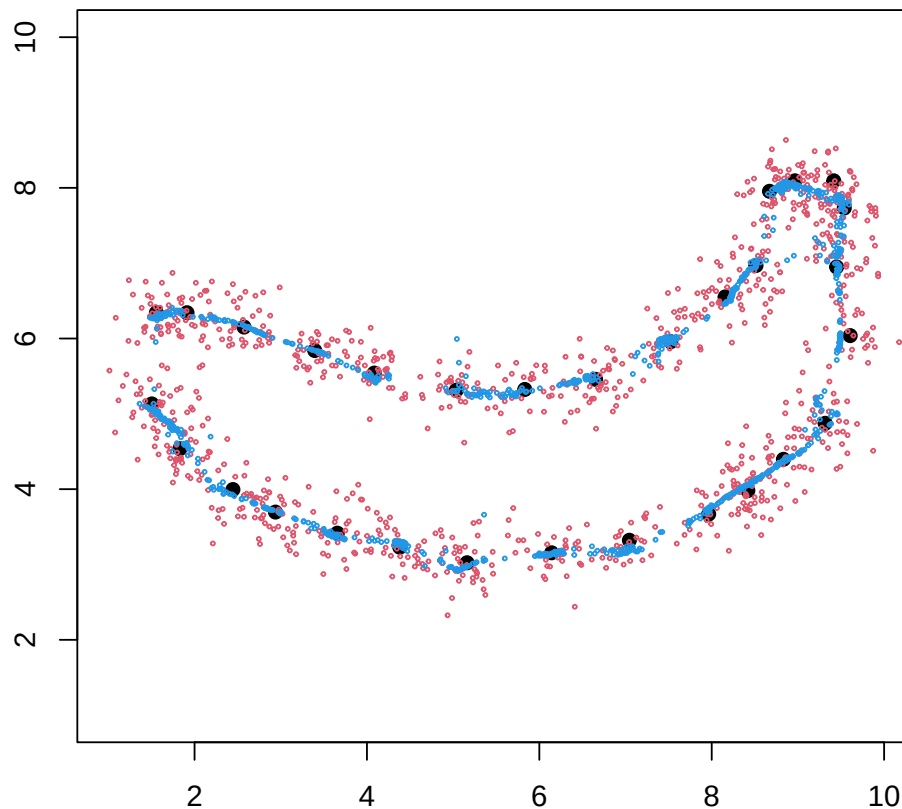


Fig. 26.3: Posterior mean banana

```
fuv <- f$f
uv <- expand.grid(alpha = u, theta = v)
g <- as.vector(A %*% (duv * fuv))
du <- A %*% (uv[, 1] * duv * fuv)/g
dv <- A %*% (uv[, 2] * duv * fuv)/g
z <- list(u = u, v = v, fuv = fuv, logLik = logLik, du = du,
         dv = dv, A = A, status = f$status)
class(z) <- "GLVmix"
```

}^z

How Quantile Regression Rescued Happiness – for the Rich

1

27.1 Introduction

Killingsworth, Kahneman, and Mellers (2023) have recently argued that earlier work of Kahneman and Deaton (2010) that found that the relationship between log income and self-reported happiness “flattened” at higher incomes was flawed due to overlooked heterogeneity. Quantile regression methods revealed that this effect was attributable to the least happy few, while those at upper quantiles of the conditional distribution of happiness found incremental income beyond \$100,000 just as joyful as below this threshold. Assertions in Killingsworth, Kahneman, and Mellers (2023) that the conclusions of Kahneman and Deaton (2010) might have been different if the relationship between income and unhappiness were studied make no sense to me, nor do claims that there is a “correct” solution to questions raised by this line of research.

I will resist the temptation to indulge in speculation about the measurement of happiness, or for that matter income. The data employed by Killingsworth, Kahneman, and Mellers (2023) is available from <https://osf.io/qye4a/>. and consists of 33,391 individual observations with reported income and responses to the question, “How do you feel right now” at randomly selected times over the survey period. How these temporal responses were aggregated is not clear from the PNAS paper. Other covariates were also collected, but do not appear in the results reported or the data source provided.

¹ Thanks to Angus Deaton for clarifying differences in the two data sources used in the earlier and later papers. The former paper used responses from the Gallup survey question: “did you experience a lot of happiness yesterday?” The Killingsworth data came from responses to randomly timed “beeper” inquiries.

27.2 Replication

Figure 1 contains a scatter plot of the raw data and estimated decile estimates. If you squint carefully you will see that the the first and second conditional deciles have slightly flatter slopes after the (red) \$100,000 income threshold than before the threshold. The blue curve represents the ordinary least squares fit of the simple broken-line regression model. Note that the OLS fit *does not flatten* noticeably as reported in the original Kahneman and Deaton (2010) paper, and as claimed in new Killingsworth, Kahneman, and Mellers (2023) paper. Since the two papers employ quite different data, and the former paper incorporates additional covariate effects this is hardly surprising, but it calls into question the claim of the latter authors that they have found a conclusive, “correct” explanation for the findings in the prior paper.

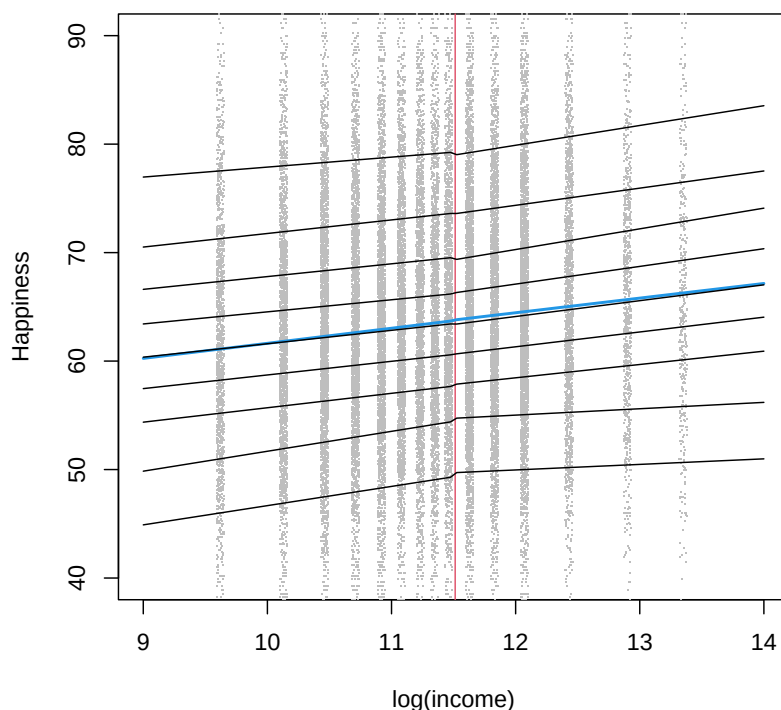


Fig. 27.1: Raw Data and Regression Decile Estimates. The red vertical line is the \$100,000 threshold. The heavy blue line is the mean regression fit. Income responses have been dithered in an effort to convey a sense of their observed density.

Figure 2 depicts the quantile regression point estimates with pointwise confidence bands. The red horizontal lines represent the least squares estimate and its associated confidence interval. The black lines and the associated shaded regions represent the corresponding quantile regression estimates. The indicator variable d takes the value 1 if income exceeds \$100,000, and 0 otherwise, so the model permits two distinct linear segments: one for income below this threshold and another for income above it. Of primary interest in the present case is the difference in the slopes of these segments at the various quantiles. From the plots in Figure 2 it is apparent that at the lower deciles the difference in these slopes given by the panel labeled $x : d$ is considerably lower for the segment above the threshold than below. Although the OLS point estimate is essentially zero for this interaction term, for the 0.10 and 0.20 quantile regression estimate it is significantly below 0, as revealed by the “flattening” observed in Figure 1 for these quantiles. Note too that in the upper tail the situation is reversed; the slope of the income/happiness relationship is higher above the threshold than below it. This is also visible in Figure 1 where the 0.90 conditional quantile bends upward slightly at the threshold.

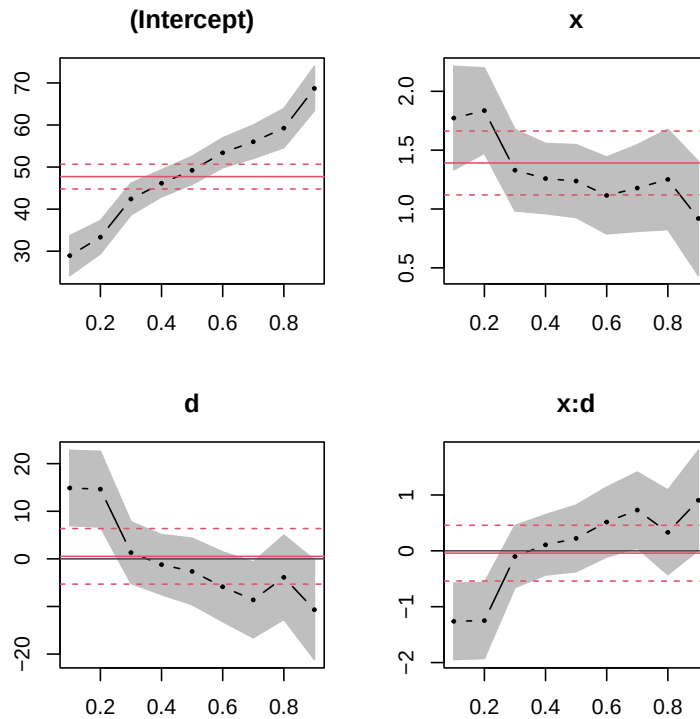


Fig. 27.2: Pointwise Confidence Bands for QR Estimates

27.3 Conclusion

What does it all mean? I suppose if one were to take self-reported happiness scales seriously – as it happens I do not – the results would suggest that most respondents get approximately linear satisfaction over the entire log income range surveyed, as anticipated by Daniel Bernoulli. Those who are already quite happy at low to moderate incomes get even more joy from increments to log income at higher income levels. On the contrary, those that are relatively unhappy at lower income levels don't get much of a boost from log income increments beyond the \$100,000 threshold. For the naive utilitarian this might suggest redistribution schemes from the sad to the glad, but you would have to be mad to subscribe to that.

Borges and AI

28.1 Introduction

In the novel *Galatea 2.0* Rick Powers imagined a proto-LLM trained by the narrator and a CS colleague to pass a qualifying exam in English literature. We seem to be much closer to the reality of this scenario today than we were in 1995 when Powers's novel appeared, but we still have a long ways to go.

Of course, Borges was there well before Powers and judging from a little googling, there has been quite an outpouring of Borgesian commentary about the current state of AI lately. I was aware that Walter Sosa-Escudero, an outstanding econometrics student at UIUC, now at Universidad de San Andres, wrote *Sosa-Escudero (2020)* several years ago. Unfortunately, it is still not available in English.

Recently, Bottou and Schölkopf (2023) have explored some connections between contemporary LLMs and the work of Borges, principally through *Garden of the Forking Paths*, and *Library of Babel*. The latter story seems particularly apt since the Library includes not only the known corpus of world libraries, but all the false catalogues comprising permutations of the extant corpus. The task of the librarians is to find vindications, “books of apologia and prophecies that would vindicate for all time the actions of every person in the universe.” This sifting might be thought to be the job of science, or even philosophy, not something that could be outsourced to mere machines, but such sifting is also the aim of the Chatbot as it appends token after token to produce a response.¹

28.2 Natural Intelligence

Vindication is hard, there are many ways to continue a story, but only a few good ways. This was brought home to me while reading about the recent Gino/Ariely controversy in

¹ My experience with Geppetto has been limited to a few queries about sed/awk constructions, for which it seems remarkably well prepared. For a less successful experience, I recommend Elif Batuman's hilarious account of asking Geppetto to find a passage from Proust. <https://eliflife.substack.com/p/my-conversation-with-chat-gpt-about>

behavioral economics. If we were to judge Harvard Business Review articles by the standards of the fiction contributions to the New Yorker or the Sewanee Review, surely these controversial papers would never have appeared in print. They begin with a small psychological conceit to be explored, some characters – usually undergrads – who never appear as more than one dimensional just data points in a misaligned excel sheet. There is no plot, or perceptible philosophical or political point. Gross abuse of data seems the least of their sins. On the other hand, when there is so little other merit in the enterprise maybe respect for the data is the least we should expect.

28.3 What Now?

On a tangentially related topic Donoho (2024) has written a meditation on the cataclysmic consequences of AI that may lurk just around the corner. He identifies three factors that have contributed to the rapid recent progress in LLMs: Data availability and sharing, ease of code sharing, and commonly agreed upon challenges with accepted measures of performance. The first two are familiar – they are just the reproducible research agenda that many of us have pushed for a long time. The third is new, although Donoho’s Tukey lecture already emphasised it in his discussion of machine translation. I found the whole argument very compelling, but the message seems a bit discouraging for economics. Economics lacks a common objective. Unlike computer vision, or machine translation there is repository of tasks whose performance can be judged by a common standard.² Maybe this is what will save us, while auditors, financial managers and lawyers are eaten alive by future generations of Gepetto.

The more speculative part of the Donoho paper involves whether AI will evolve into a few massive hegemonies, or fragment into many smaller units driven by innovations in software and hardware. For the moment his “brutal scaling” scenario seems to holding sway as start-ups are consumed by Amazon, Google and Meta, but maybe there is hope for a turn-around. There is a curious tension in Donoho’s three factors all of which thrive in the chaos of the open source arena, and the tendency to produce hegemonies of enormous power due to sheer computational capacity.

² Like performance on the suite of Atari games for reinforcement learning in Bellemare, Naddaf, Veness, and Bowling (2013).

The Dantzig Decoder Ring

29.1 Introduction

¹

As a secret agent of the deep state, your job is to transmit messages back to your superiors who are in exile somewhere in wilderness of Idaho. Messages consist of a vector of real numbers², $x \in \mathbb{R}^p$. Transmission is over a noisy channel so the following protocol has been developed. You and your superiors share knowledge of a n by p matrix A with $n \gg p$. You send the enlarged message $y = Ax$ through the noisy channel. On good days they receive the message $\hat{y} = y + u$ where $u \in \mathbb{R}^n$ is a n -vector with iid (standard) Gaussian components. On such good days decoding the messages is relatively simple: the receiver can just compute the least squares projection of $\hat{y}$ on the column space of A ,

$$\hat{x} = (A^\top A)^{-1} A^\top \hat{y},$$

and the message will be denoised relatively accurately, depending to some degree on the choice of n and A .

However, on bad days the noise in transmission can be much worse. In particular, it may contain a limited number of very large errors that can severely disrupt the quality of the $\hat{x}$ solution. More explicitly, suppose that on bad days, instead of $\hat{y}$ the vector $\tilde{y} = y + u + v$ is received with v being arbitrary, but relatively sparse in the sense that most of its coordinates are zero. Decoding under these conditions is considerably more difficult, but you have developed the following alternative protocol. Let,

$$\tilde{x} = (A^\top A)^{-1} A^\top (\tilde{y} - \tilde{v}),$$

where

$$\tilde{v} = \operatorname{argmin}\{\|v\|_1 \text{ such that } \|Q(\tilde{y} - v)\|_\infty \leq K\}$$

and $Q = I - A(A^\top A)^{-1} A^\top$. This form of denoising is closely related to the so-called Dantzig selector of Candes and Tao (2007), which we will now describe in more detail.

¹ An earlier version this note was written in 2007 for an exercise in my Econometric Theory course.

An extensive collection of references and code for related methods can be found at Candes (2007).

² As a practical matter, given the noise, only a few binary digits of the numbers are relevant.

29.2 The Dantzig Selector

The basic idea of the Dantzig selector is to solve regression problems that look like this:

$$\min\{\|b\|_1 \mid \|X^\top r\|_\infty < K\}.$$

where $r = y - Xb$. This looks a lot like the familiar lasso penalized least squares regression problem, and that's because it is very similar. Rather than adding an ℓ_1 penalty to the usual least squares regression objective we minimize the ℓ_1 norm of the coefficients subject to a ℓ_∞ norm constraint on the gradient of the least squares criterion. Rather than solving a quadratic programming problem we have a somewhat simpler linear programming problem instead.

We consider a slightly more general version:

$$\min\{\|b\|_1 \mid K \leq X^\top r \leq K\}.$$

for vector valued K . We can implement this version with the inequality constrained quantile regression function, `rq.fit.fnc(X, y, R, r)` from the R package **quantreg**. This function solves problems of the form:

$$\min\{\|y - Xb\|_1 \mid Rb \geq r\}$$

If we set, $X = I_p$ the p dimensional identity matrix and $y = 0 \in \mathbb{R}^p$ with $R = [X^\top X; -X^\top X]$ and $r = [(X^\top y - K)^\top, (-X^\top y - K)^\top]^\top$ we have the Dantzig selector as a special case. Obviously, the parameter K controls the amount of shrinkage of the coefficients b .

29.3 Dantzig Decoding

Returning to our denoising problem, let V denote the subspace spanned by the columns of A and $V^\perp$ the orthocomplement of V . We first construct a matrix, Q , whose columns form an orthobasis for $V^\perp$, so $Q^\top A = 0$. Now, we can solve for $\tilde{v}$ with the Dantzig selector, setting $X = Q^\top$ and $y = Q^\top \tilde{y}$. It only remains to choose the tuning parameter K . This can be somewhat delicate, since it depends upon the scale of the Gaussian noise vector, u , and other features of the problem. We adopt a simple expedient in the illustrative code below. Actually, this isn't quite the end of it. Like the well-known feature/bug of the lasso, the Dantzig selector also tends to act too aggressively on some coordinates, so Candes and Randall (2008) recommend a "reprojection" step that adjusts for this. Code for both the Dantzig selector and the Dantzig Decoder in the R language appears in the next section.

29.4 A Simulated Example

In this section we illustrate the use of the Dantzig decoder with a simple example. The original message consists of a $p = 256$ iid Gaussian vector. The shared matrix A is $n = 2p$ by p with iid Gaussian entries. A fraction $\rho = 0.1$ of the coordinates of the vector, $y = Ax$,

to be transmitted are contaminated by replacing their intended values by the negative of those values. Finally, Gaussian noise is added to the transmitted vector.

In Figure 1 we show three scatter plots each plotting a decoded version of the message against its true values, x . In the left panel we plot $\hat{x}$ using the naive least squares decoding. In the middle panel we plot the Dantzig decoding, $\tilde{x}$. And in the right panel we plot an oracle decoding achievable if the contamination component v were known. In each panel we report root mean squared error of the respective decodings. It can be seen that the Dantzig decoding is almost as good as the oracle decoding, while the Gaussian decoding is roughly 10 times noisier than the Dantzig method.

```
DantzigSelector <- function(X,y,K){
  require(quantreg)
  n <- nrow(X)
  p <- ncol(X)
  A <- crossprod(X)
  R <- rbind(A, -A)
  a <- crossprod(X,y)
  r <- c(a-K, -a-K)
  zp <- rep(0,p)
  Ip <- diag(p)
  f <- rq.fit.fnc(Ip, zp, R=R, r=r)
  return(f)
}

DantzigDecoder <- function(A,y,sigma,Kval = 3){
  NullSpace <- function(A){
    S <- svd(A,nrow(A))
    r <- length(S$d)
    S$u[(r+1):nrow(A),drop=FALSE]
  }
  Q <- t(NullSpace(A))
  Qy <- Q %*% y
  K <- sqrt(diag(crossprod(Q))) * sigma * Kval
  f <- DantzigSelector(Q, Qy, K)
  # Reprojection Step
  s <- abs(f$coef) > sigma
  e <- rep(0,nrow(A))
  e[s] <- lm(Qy ~ Q[,s] - 1)$coef
  xhat <- lm((y-e) ~ A - 1)$coef
  return(xhat)
}
```

```
p <- 256
n <- 2 * p
set.seed(1729)
A <- matrix(rnorm(p*n),n,p)
```

```

rho <- 0.10
k <- round(rho * n)
x <- rnorm(p)
y <- A %*% x
yt <- y
sigma <- median(abs(y))/16
s <- sample(1:n,k)
yt[s] <- -yt[s] # Original Candes example
u <- rnorm(n)*sigma
yt <- yt + u
xhat <- DantzigDecoder(A, yt, sigma)

par(mfrow=c(1,3))
g <- coef(lm(yt ~ A -1))
rmseG <- round(sqrt(var(x-g)),3)
plot(x,g,ylab=expression(hat(x)),main="Gauss")
text(1,-2,paste("RMSE = ",rmseG))
plot(x,xhat,ylab=expression(hat(x)),main="Dantzig")
rmseD <- round(sqrt(var(x-xhat)),3)
text(1,-2,paste("RMSE = ",rmseD))
h <- coef(lm((y + u) ~ A - 1))
plot(x,h,ylab=expression(hat(x)),main="Oracle")
rmseH <- round(sqrt(var(x-h)),3)
text(1,-2,paste("RMSE = ",rmseH))

```

29.5 Conclusion

“OK, so it seems to work, but why?” To answer this question it is necessary to delve a bit deeper into the inner workings of median regression estimator. A hint about this is provided by the sparseness of the contamination vector, v . The Dantzig decoder acts like the conventional ℓ_1 regression estimator. As long as there are only a few non-zero entries in v , it is capable of easily identifying them, and once they are removed decoding is essentially as good as the oracle operating on purely Gaussian data.

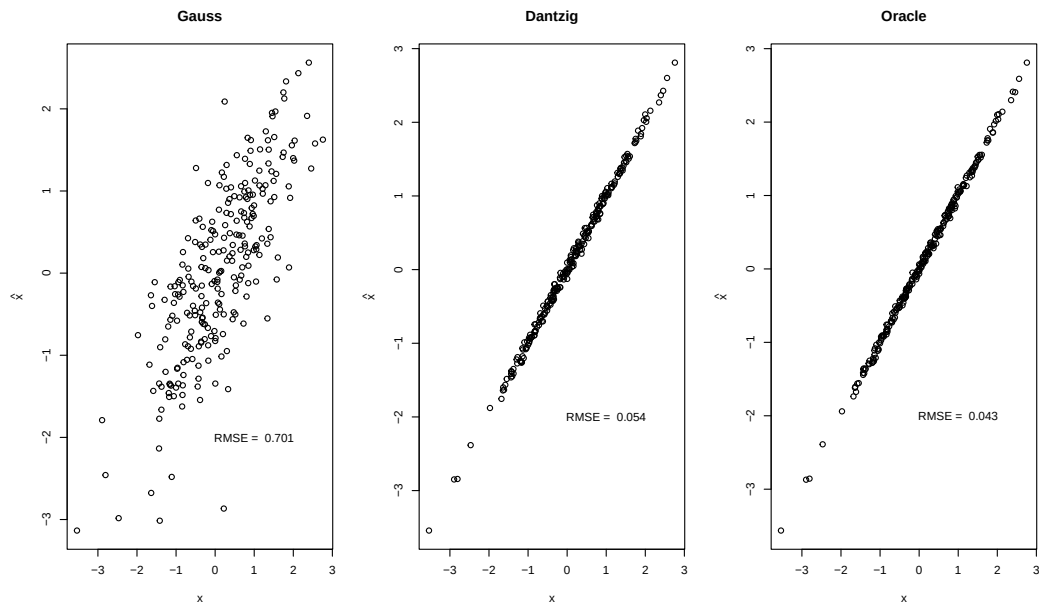


Fig. 29.1: RMSE Effectiveness of Gaussian and Dantzig Decoding

Against Causality

30.1 Introduction

Causal inference has become a central pillar of social and biomedical statistical science, and yet the foundations for this pillar remain mired in controversy. The mantra “correlation is not causation” is taught everywhere. Far be it from me to endorse correlation in any of its myriad forms, but if causation is not correlation then what is it? Do we really need it?

Bertrand Russell thought not. Russell (1912) writes “‘cause’ is so inextricably bound up with misleading associations as to make its complete extrusion from the philosophical vocabulary desirable. . . . The law of causality, I believe, like much that passes muster among philosophers, is a relic of a bygone age, surviving, like the monarchy, only because it is erroneously supposed to do no harm.” Russell’s argument rests partly on fact that physics seems to do quite nicely without any reliance on causality: physical laws are time reversible. Without some asymmetry it is difficult to imagine how a valid notion of causality can be developed. But if physics can do without causality, why does social science find it so compelling? A naive psychological answer to this question might be that we wish to assign some blame or credit for events as they occur around us. But there are usually a plethora of possible candidates for causal factors, and even more problematically such factors rarely induce a unique outcome, but rather a probabilistic distribution of outcomes. Thus, medical treatments are almost never deterministic “cures” – they are at best coin flips. I will argue that these considerations mandate a more circumspect view of statistical methods, one that puts conditional distributions front and center and abandons claims for causality. Of course, evidence accumulated by estimating conditional distributions can eventually yield reliable guidance about smoking Doll, Peto, Boreham, and Sutherland (2004) and other issues of everyday life and science.

30.2 Regression Exchangeability

McCullagh (2005) offers a useful framework he calls regression exchangeability for considering what can be extracted from statistical models. Exchangeability in its original de Finetti form requires permutation invariance, but it can be extended to regression settings by the

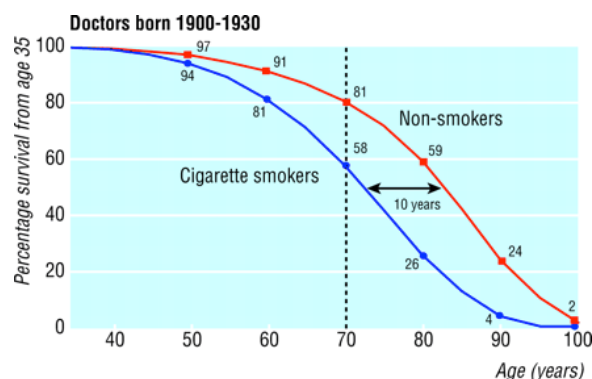


Fig. 30.1: Source: Doll, Peto, Boreham and Sutherland

requirement that observations with the same covariate values are exchangeable and therefore have the same finite dimensional marginal distributions. McCullagh emphasizes Gaussian regression models with and without random effects, but his viewpoint is completely in accord with nonparametric quantile regression principles; penalty methods can also be easily incorporated. Treatment effects are easily distinguished from classification factors, Cox (1984) although one should always entertain the possibility that there are important interactions that need to be accounted for.

What do potential outcomes add to this effort? In my view they have only muddled the waters. Too much attention has been lavished on binary treatment binary response settings in which because treatment is deterministic paradoxes quickly emerge as soon as one admits that treatment effects might be stochastic. Gelman and Mikhael (2025) illustrates this in some simple settings where treatments with random outcomes combined with a potential outcome outlook lead to violations of the likelihood principle.

Rather than fixating on binary response it seems better to adopt a survival analysis perspective at least for most medical applications. Treatments have associated survival functions conditioned perhaps on covariates. To illustrate, consider Figure 30.1 that depicts the results of a large observational study, Doll, Peto, Boreham, and Sutherland (2004) of 34,439 UK doctors born between 1900 and 1930 followed up from 1951 to 2001. If we imagine a random doctor assigned a number $U \sim U[0, 1]$ we can read off his survival time in the figure. For example, the median doctor assigned $U = 1/2$ lives to about 72 if he was a smoker, but 82 if not. Of course the chorus of causal cacophonists will say: How can we know that a median non-smoker will also be a median smoker? Of course we can't. It may well be that smokers engage in all sorts of other dangerous behavior that makes them die earlier, so all of the 10 years difference in median survival shouldn't be attributed to smoking.¹ This ambiguity itself is reason to question the usefulness of causality as a category of statistical analysis. As a matter of interpretation, sure, but by moving us from the empirical to the synthetic a

¹ This study also provides valuable evidence on how the timing of cessation of smoking influences survival. Stopping at ages 25-34 is almost indistinguishable from never smoking, while stopping between 55-64 reduces the median survival time from 10 years to 7 years.

priori it takes us out of the realm of science. This claim is vindicated by the fact that two of the most prominent statisticians of the 20th century vigorously disputed the causal link between smoking and cancer: R.A. Fisher and Joseph Berkson, the head of biostatistics at the Mayo Clinic. Eventually, the *association* between smoking and both cancer and heart disease became so apparent that with or without causality the medical establishment came around to the view that smoking was harmful.

30.3 Je ne regrette rien

In contrast to the analysis of treatments for medical conditions finance offers a distinct environment for decision theory. Consider two different portfolio strategies: we can evaluate the performance of both strategies in any future period since they depend on the same realization of future returns. This is quite different than the medical setting where the consequences of only one “treatment” is observed. This opens the way to the consideration of regret. There can be no regret in the medical setting since we never see the consequences of the alternative not taken, whereas in roulette or portfolio management it is staring us in the face every moment. We are immediately aware of what we should have done, or might have done had we been more farsighted. Is there a role for causality in such applications? I think not. Certainly, those of the “no causation without manipulability” school would have to agree, unless of course there were ways to tilt the roulette wheel or corner the market.

30.4 Structural Equation Models

Structural equation modeling has a long and illustrious history in econometrics beginning with the path breaking work Sewall and Phillip Wright on demand analysis for agricultural commodities. This strand of the literature eventually led to the so-called cobweb model of supply and demand and was extended to multivariate settings by Wold and Strotz (1960). Leamer (1985) offers a characteristically insightful overview of the econometric side of this relationship. Although his title suggests that he intends to confine his remarks to vector autoregressions the scope of the paper is much broader. Indeed, it renders much of what I intended to say superfluous, so I can be brief.

More recently structural equation modeling has been resurrected as a framework for causal inference. Bollen and Pearl (2013) discuss several “myths” that they claim inhibit appreciation of SEMs for causal inference. My gripe about their utility for this purpose is orthogonal to their defense of them. Traditionally, econometrics has restricted attention to linear structural models with the attendant restriction of probabilistic effects to conditional means. This is perhaps inevitable in applications like macroeconometrics where data is in such short supply. But it is now generally acknowledged that treatments have stochastic effects and models should ideally be structured to reveal distributional effects. A leading example of this more modern approach is Chesher (2003) who formulates nonlinear structural models for conditional quantile functions. When such models are recursive the Rosenblatt (1952) transform implies that they are causal as discussed in Ma and Koenker (2006).

However, estimation and prediction in such models even in low dimensions can be very challenging.

To see why consider a simple two equation cobweb type model in which supply responses lag one period and demand adjusts to price changes to clear the market. In the traditional linear conditional mean version of this model this is all quite straightforward. However, when we would like to construct a conditional quantile version we must be aware that the distribution of demand adjusts to the distribution of supply so we have an operator that takes supply into demand, and vice-versa. This is perhaps made more clear in considering models of peer effects in studying student performance in schools. Ideally, we would like to design experiments in which the distribution of student abilities in various classes was randomized so we could infer something about whether the presence of talented students helped only their talented compatriots or had beneficial effects for the less talented students as well. Such experiments would be difficult to design and illustrate why I am skeptical about their success when it is crucial to consider the distributional effects of treatments. Inherently, once we admit that treatments have stochastic consequences we are left without a clear causal mechanism and claims to the contrary not withstanding we should proceed with extreme caution.

Choquet Risk and Portfolio Optimization

31.1 Introduction

In Bassett, Koenker, and Kordas (2004) we described a new approach to portfolio optimization that replaced the Markowitz notion of risk as the variance of returns by a class of pessimistic risk measures based on lower tail expectations. Instead of relying entirely on curvature of the utility function to account for attitudes toward risk, the new approach allowed decision makers to focus increased attention on riskiest outcomes in the lower tail. Suppose that we have a scalar assets, X , and Y with return distributions F and G and associated quantile functions, F^{-1} and G^{-1} . Conventional expected utility theory a la von Neumann Morgenstern says that we should prefer X to Y iff

$$\mathbb{E}_F u(X) = \int_{-\infty}^{\infty} u(x) dF(x) > \int_{-\infty}^{\infty} u(y) dG(y) = \mathbb{E}_G u(Y).$$

Changing variables we have the equivalent decision rule that X is preferred iff

$$\mathbb{E}_F u(X) = \int_0^1 u(F^{-1}(t)) dt > \int_0^1 u(G^{-1}(t)) dt = \mathbb{E}_G u(Y).$$

In effect, we have transformed the integrand into “equally likely” slices of the possible outcomes *ordered by their desirability*.

In contrast, we considered Choquet expectations that reweighted the outcome probabilities by a probability measure ν , so X is now preferred to Y iff,

$$\mathbb{E}_{\nu, F} u(X) = \int_0^1 u(F^{-1}(t)) d\nu(t) > \int_0^1 u(G^{-1}(t)) d\nu(t) = \mathbb{E}_G u(Y).$$

We called ν a distortion function and required that it be concave from $[0, 1]$ to $[0, 1]$. Concavity implies that the probability of adverse events are weighted more than the more favorable events which is consistent with the usual notion of pessimism. We can think of this Choquet expectation as a risk measure, indeed the class of pessimistic Choquet expectations defined in this way coincides with what Artzner, Delbaen, Eber, and Heath (1999) call coherent risk measures.

A simple one parameter class of pessimistic distortion functions is,

$$\nu_\theta = 1 - (1 - t)^\theta, \quad \theta \geq 1.$$

Members of this family range from the perfectly equanimous with $\theta = 1$ to the profoundly pessimistic as $\theta \rightarrow \infty$. Changing variables, we can write,

$$\mathbb{E}_{\nu_\theta, F} X = \int_0^1 F^{-1}(t) d\nu(t) = \int_{-\infty}^{\infty} u dG_\theta(u),$$

where $G_\theta(u) = (1 - (1 - F(u))^\theta)$. When θ is a positive integer, say m , the pessimist imagines that instead of getting a single draw of X from the distribution F , he gets m draws of X among which he ultimately receives the worst. Such preferences are clearly pessimistic in the sense of the ancient principle: Expect the worst, and you won't be disappointed.

Perhaps the simplest example of the pessimistic Choquet expectation involves the one parameter family, $\nu_\alpha(t) = \min\{t/\alpha, 1\}$ for $\alpha \in [0, 1]$. In this case we have,

$$\mathbb{E}_{\nu_\alpha, F} u(X) = \alpha^{-1} \int_0^\alpha u(F^{-1}(t)) dt,$$

and we see that – compared to the usual expected utility computation – the probabilities of the α least favorable outcomes are inflated and those of the $1 - \alpha$ most favorable outcomes are discounted entirely.

31.2 Pessimistic Risk Measures

The general class of pessimistic risk measures can be constructed from the building blocks of the previous example. At this point we will drop the explicit utility function and focus on measuring risk directly as linear in outcomes. Our basic building block, which we call α -risk, is defined as,

$$\varrho_{\nu_\alpha}(X) = -\alpha^{-1} \int_0^\alpha F^{-1}(t) dt.$$

We will define our general class of risk measures by,

$$\varrho(X) = \int_0^1 \varrho_{\nu_\alpha}(X) d\varphi(\alpha),$$

for some probability measure φ . To see why such measures should be viewed as pessimistic, note that by Fubini,

$$\varrho(X) = - \int_0^1 \alpha^{-1} \int_0^\alpha F^{-1}(t) dt d\varphi(\alpha) = - \int_0^1 F^{-1}(t) \int_t^1 \alpha^{-1} d\varphi(\alpha) dt.$$

Now suppose we take φ as a finite sum of (Dirac) point masses, say, $d\varphi = \sum_{i=0}^m \varphi_i \delta_{\tau_i}$ with $\varphi_i \geq 0$, $\sum \varphi_i = 1$, and $0 = \tau_0 < \tau_1 < \dots < \tau_m \leq 1$. Noting that

$$\int_t^1 \alpha^{-1} \delta_\tau(\alpha) d\alpha = \tau^{-1} I(t < \tau),$$

we can write

$$\varrho(X) = \int_0^1 \varrho_{\nu_\alpha}(X) d\varphi(\alpha) = -\varphi_0 F^{-1}(0) - \int_0^1 F^{-1}(t) \gamma(t) dt \quad (31.1)$$

where $\gamma(t) = \sum_{i=1}^m \varphi_i \tau_i^{-1} I(t < \tau_i)$. Positivity of the point masses, φ_i , assures that the resulting density weights are decreasing, so the resulting distortion in probabilities acts to accentuate the implicit likelihood of the least favorable outcomes, and depress the likelihood of the most favorable ones. As noted in Bassett, Koenker, and Kordas (2004) this class of risk measures coincides with the class of coherent risk measures as defined by Artzner, Delbaen, Eber, and Heath (1999).

31.3 Portfolio Optimization

We now come to the important question: What good is all this? A provisional answer is provided by demonstrating how to construct portfolios using this new measure of risk in place of variance. This brings us to quantile regression. Suppose we have a scalar random variable X , with $\mathbb{E}X = \mu < \infty$, say the return on a portfolio with fixed portfolio weights, we can compute the α th quantile, ξ_α , of the returns on the portfolio, X , by solving,

$$\min_{\xi \in \mathbb{R}} \mathbb{E} \rho_\alpha(X - \xi),$$

where $\rho_\alpha(u) = u(\alpha - I(u < 0))$. It is easy to see that,

$$\begin{aligned} \mathbb{E} \rho_\alpha(X - \xi_\alpha) &= \alpha(\mu - \xi_\alpha) - \int_{-\infty}^{\xi_\alpha} (x - \xi_\alpha) dF_X(x) \\ &= \alpha\mu - \int_0^\alpha F^{-1}(t) dt \\ &= \alpha\mu + \alpha \varrho_{\nu_\alpha}(X). \end{aligned}$$

This observation is the key to constructing portfolio weights using the new risk measure ϱ .

Given a random sample $\{x_i : i = 1, \dots, n\}$ on X an empirical analogue of α risk can be expressed as,

$$\hat{\varrho}_{\nu_\alpha}(x) = (n\alpha)^{-1} \min_{\xi \in \mathbb{R}} \sum_{i=1}^n \rho_\alpha(x_i - \xi) - \hat{\mu}_n$$

where $\hat{\mu}_n$ denotes an estimator of $\mathbb{E}X = \mu$, presumably just $\bar{x}_n$. Now suppose that X is no longer a return on a single asset, but instead a vector of returns, $X = (X_1, \dots, X_p)^\top$ and we are interested in forming portfolios, $Y = X^\top \pi$. In particular we would like to choose the portfolio weights, π so as to minimize our pessimistic measure of risk subject to a constraint to expected returns. Portfolio weights should sum to one, so we can write the Lagrangian of the constrained problem as, sum to one, and reformulate the problem as,

$$\min_{\pi} \varrho_{\nu_{\alpha}}(X^{\top} \pi) \text{ s.t. } \mu(X^{\top} \pi) = \mu_0, \mathbf{1}^{\top} \pi = 1.$$

Taking the first asset (arbitrarily) as numeraire we can rewrite the problem as,

$$\min_{(\beta, \xi) \in \mathbb{R}^p} \sum_{i=1}^n \rho_{\alpha}(x_{i1} - \sum_{j=2}^p (x_{i1} - x_{ij})\beta_j - \xi) \text{ s.t. } \bar{x}^{\top} \pi(\beta) = \mu_0, \quad (31.2)$$

where $\pi(\beta) = (1 - \sum_{j=2}^p \beta_j, \beta^{\top})^{\top}$.

This is almost a standard quantile regression problem; the only minor idiosyncrasy is the mean constraint. This constraint may be imposed by adding a single pseudo-observation with response $\kappa(\bar{x}_1 - \mu_0)$ and design row $\kappa(0, \bar{x}_1 - \bar{x}_2, \dots, \bar{x}_1 - \bar{x}_p)^{\top}$. The zero element corresponds to the intercept column of the design matrix. Choosing κ sufficiently large, we can ensure that the mean constraint is satisfied.

At this point we have assumed that risk is just the simple α -risk, but this is easily generalized to weighted averages of α -risks,

$$\varrho_{\nu}(X) = \sum_{k=1}^m \nu_k \varrho_{\nu_{\alpha_k}}(X),$$

where the weights $\nu_k : 1, \dots, m$ are positive and sum to one, giving the empirical implementation,

$$\min_{(\beta, \xi) \in \mathbb{R}^{p+m}} \sum_{k=1}^m \sum_{i=1}^n \nu_k \rho_{\alpha_k}(x_{i1} - \sum_{j=2}^p (x_{i1} - x_{ij})\beta_j - \xi_k) \text{ s.t. } \bar{x}^{\top} \pi(\beta) = \mu_0. \quad (31.3)$$

We have simply concatenated m quantile regression objective functions allowing each to have a distinct intercept, ξ_k , but constraining the slope coefficients representing the portfolio weights to be the same for each of the k components. Since any pessimistic distortion function can be represented by a piecewise linear concave function like this we have a general solution to the problem of estimating pessimistic portfolio weights. The large sample behavior of such estimators is described in Koenker (1984). A software implementation of this proposed approach is available in **quantreg** package of the R language in the function **qrisk**.

Further refinements may be introduced. For example we may wish to preclude short sales by requiring the portfolio weights to be positive, or bound the magnitudes of portfolio weights. Any such linear inequality constraints are easily accommodated within the linear programming framework of quantile regression.

31.4 A Simulation Exercise

When I presented the original version of this work the question that always seemed to come up was: Would individuals with these Choquet preferences survive in a securities market with traders who had traditional mean variance preferences. My snarky answer was “Sure, why not? Variance is a dumb measure of risk.” But this was never investigated properly. Maybe now is the time to do this.

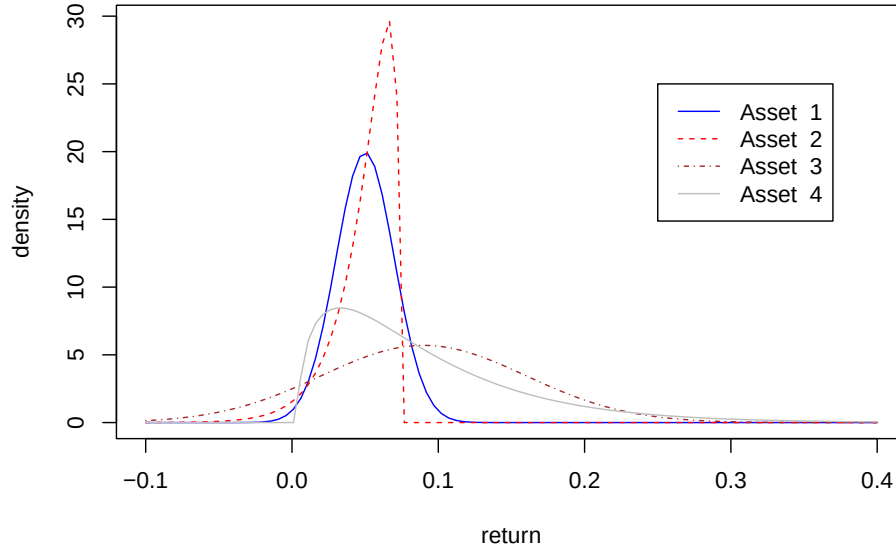


Fig. 31.1: Four Asset Densities: Note that assets 1 and 2 have identical mean and variance of returns, likewise assets 3 and 4 have the same mean and variance. However, due to asymmetry of the returns distribution asset 1 has better performance in both tails than asset 2, and asset 4 has better performance than asset 3 in both tails.

Rather than using real data about which I'm sadly uninformed, I've decided to use artificial data based on the matched pairs of densities used in the earlier paper. These were constructed as follows: Asset 1 is normally distributed with mean 0.05 and standard deviation 0.02. Asset 2 has a reversed χ^2_3 density with location and scale chosen so that its mean and variance are identical to those of asset 1. Similarly, assets 3 and 4 are constructed to have identical means (0.09) and standard deviations (0.07). But for the second pair of densities the χ^2_3 asset is skewed to the right, as usual, not to the left. The four densities are illustrated in Figure 31.1

In an attempt to make the simulation setting somewhat more realistic I have constructed a "returns" matrix of dimension 1000 by 60. The first 30 columns consist of pairs with marginal densities of type 1 and 2, the last 30 columns of pairs of type 3 and 4. The pairwise correlations of these columns are in the range 0.5 to 0.85 for each pair, but independent between the two pairs. Given this data structure we compute portfolio weights by both the Markowitz and Choquet methods. For the Choquet portfolio we use the distortion function, $d\nu = 0.7\delta_{0.3} + 0.3\delta_{0.6}$.

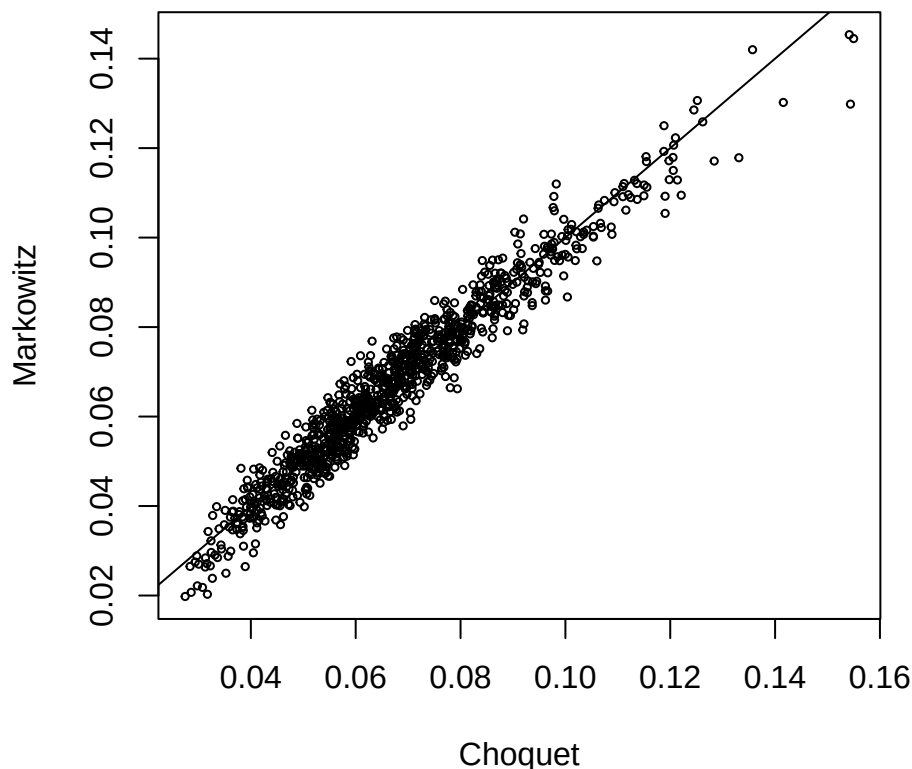


Fig. 31.2: Markowitz vs Choquet Returns for the Evaluation Sample. Note that the Choquet portfolio has better performance both in the lower and upper tail of the returns distribution.

To compare performance we construct portfolio weights for the two methods based on an initial sample of 1000 observations, and then evaluate on the basis of a new sample drawn with the same structure as the initial sample. In Figure 31.2 we plot the Markowitz returns against the Choquet returns for the evaluation sample of 1000 observations. For reference the 45 degree line divides the plot into two regions: points in the lower triangle are those for which the Choquet return exceeds the Markowitz return. Note that such points predominate in both extremes of the returns distribution. This is to be expected since the Choquet risk measure is sensitive to the lower tail performance, while the Markowitz risk measure is sometimes fooled into judging upper tail returns as risky.

31.5 Conclusion

The Markowitz portfolio paradigm is flawed both because variance as a measure of risk fails to focus on the lower tail of returns and because it attributes risk to the upper tail of returns. In contrast, the Choquet paradigm allows the investor to construct a more meaningful measure of risk to trade off with expected return. In doing so, he can achieve better portfolio performance in both tails of the returns distribution without compromising expected return.

A

Dictionary of Received Ideas of Statistics

This appendix is a fragment of a dictionary of received ideas of statistics. Many of the original entries were composed one conference evening after too much wine. More recently it has become a commonplace book to record observations of others, see nostrification. It is obviously modeled after Flaubert's famous compendium but inevitably is only the palest shadow of the work of Le Maître. As a restorative you can find a scanned version of Flaubert's Dictionary here: <http://www.gutenberg.org/etext/14156>.

Aristotle Claimed you can fill space with regular tetrahedrons, but you can't. Don't mention horses teeth, or any Bacon.

Arithmetic has four branches, citing the Mock Turtle, Ambition, Distraction, Ugification and Derision.

Bayesianism Like Marxism, better in theory than in practice. Don't forget to mention Matthew Prior's (!) (1664-1721) meta-satirical poem, The hind and the panther transvers'd to the story of The country-mouse and the city-mouse. <http://quod.lib.umich.edu/cgi/t/text/text-idx?c=eebo;idno=A55907.0001.001>

Bayesians Ask: How should one compute the discounted expected utility of one's afterlife? See Pascal's wager. With some trepidation, quote Maurice Kendall: "I have lamented that Bayesian statisticians do not stick closely enough to the pattern laid down by Bayes himself: if they would only do as he did and publish posthumously we should all be saved a lot of trouble." [On the Future of Statistics – A Second Look, JRSS(A), (1968), 131, 182-204].

Casual Empiricism cite Virgil, "ab uno disce omnes", from one example all is revealed.

Causal Inference see Casual Empiricism.

Causality Quote Strotz and Wold: "No one has monopoly rights in defining 'causality.'" or Bertrand Russell: "The law of causality ... survives, like the monarchy, only because it is erroneously believed to do no harm." See DiD

Chance Ask what does "a 40 percent chance of rain" mean in Iowa? Also mention the fabulous painting "Allegory of Fortune" by the 16th century Ferrarese master Dosso Dossi.

Coin-flipping the real foundation of probability, see the opening scene of Rosencrantz and Guildenstern Are Dead. Mention that Doob and Feller settled the question of whether

to use the term "random variable" and "chance variable" by the flip of a coin. Consider also the grook by Piet Hein:

Psychological Tip

Whenever you're called on to make up your mind,
and you're hampered by not having any,
the best way to solve the dilemma, you'll find,
is simply by spinning a penny.
No – not so that chance shall decide the affair
while you're passively standing there moping;
but the moment the penny is up in the air,
you suddenly know what you're hoping.

Co-median the median of a distribution, F , is usually defined, in accordance with cadlag conventions, as the infimum of the set of values, $S = \{x | F(x) \geq 1/2\}$. This is the smallest value, μ that minimizes $E|X - \mu|$. When there are several minimizers of this expectation we may refer to the other (larger) ones as co-medians. [See Stigler (1977, Fractional Order Statistics), note added March 2013, I thought this was my joke, but it turns out I was just recycling Stigler.]

Conferences Say David Lodge's "Small World" is the compleat guide

Confidence What science wants, and the statistician lacks. Mention that EB Wilson (JASA, 1927) was the first to insist that random (confidence) intervals cover the true parameter with some probability, rather than that a random parameter lies in fixed interval with some probability. No less than Milton Friedman seemed confused about this point, see Neyman's (1938) Lectures and Conferences on Mathematical Statistics, p 145.

Cooking is to Chemistry as Statistics is to Mathematics, cite Speed (1986) Questions Answers and Statistics, but don't forget to mention that "Much depends on dinner." Or, "Cooking is become an art, a noble science; cooks are gentleman" Titus Livius (Livy) quoted by Knuth in the preface to Volume 3 of AoCP.

Conservative quote Oakeshott, "To be conservative ... is to prefer the familiar to the unknown, to prefer the tried to the untried, fact to mystery, the actual to the possible, the limited to the unbounded, the near to the distant, the sufficient to the superabundant, the convenient to the perfect, present laughter to utopian bliss." In other words to disdain progress and discovery.

Correlation isn't causation, it is a Gaussian delusional conspiracy by globalists. Or say, its just a cosine, why all the fuss? Cosiness is what we are all searching for.

cryptography Useless to others is the essence of

DiD The grammar of Pearl's do-calculus (and its past tense did) is tough. Quote Colm Toibin's novel "The Magician" channeling Thomas Mann and his wife Katia on trying to learn English after coming to the US. 'This "did"', Thomas said, will be the end of me. You can actually say "he did do" and then the opposite is "he didn't do". No wonder the English are so warlike. 'What about "does"?' Katia asked. 'Should it not be "do"?' he asked. 'It's both. And then there are phrasal verbs,' Katia said. "I have ordered a book on them."

Education Quote Deng Xiaopeng: "Execution is an indispensable means through which to educate the masses."

Economics Chicago Style – cite Harry Johnson: Science demands the submergence of social conscience in a welter of statistical squabbles. (cited by Griliches JPE, 1979) See intelligence. Mention Marx's complaint that no one had written so much about money and had so little of it, and that "this economic crap" was keeping him from writing his big book on Balzac. (Eagleton, LRB 29 June 2023).

Econometrics Mention Stefan Valavanis's (1959) textbook characterization: "Econometric theory is like an exquisitely balanced French recipe, spelling out precisely with how many turns to mix the sauce, how many carats of spice to add, and for how many milliseconds to bake the mixture at exactly 474 degrees of temperature. But when the statistical cook turns to raw materials, he finds that hearts of cactus fruit are unavailable, so he substitutes chunks of cantaloupe; where the recipe calls for vermicelli he uses shredded wheat; and he substitutes green garment dye for curry, ping-pong balls for turtle's eggs and, for Chalifougnac vintage 1883, a can of turpentine."

Economist Quote Ginger Baker, "Brian [Jones] was much more of a musician than Jagger will ever be — although Jagger's a great economist."

Endogeneity Ask who put the dodgy in it.

Error Cite Pareto: "Give me fruitful error anytime full of seeds, bursting with its own contradictions. You can keep your sterile truth for yourself."

ESP Quote Mosteller "If there is ESP, that is exciting. However, thus far it does not look as if it will replace the telephone." (1991 Stat. Science)

Eugenics produced the tasteless, square tomato; mention male pattern dumbness, or the comment by Karlin about I.J. Schoenberg who married the daughter of the eminent mathematician Edmund Landau and thus became a true example of the "established genetic principle" that mathematical talent is an inheritable trait usually transmitted from father to son-in-law. [J. Approx. Theory, 1973]

Elicitation see tuning parameter, mention Tom Sawyer's method of painting fences – find someone else to do your dirty work.

Evidence Quote the Venerable Sherlock Holmes: "I see no more than you but I have trained myself to notice what I see." Or Richard Pryor: "Who are you going to believe, me, or your lying eyes." Or the Soviet maxim, "He lies like an eye witness."

Finance the alchemy of modern science.

Fun as interruption, cite Gelman's blog entry about a 2007 talk by Sarah Igo about her book *The Averaged American*, who commented that "when she speaks to historians, they just let her speak, but we're more fun because we interrupt her frequently". Gelman assured her that if interruption=fun, then economists are the most fun of all. . .

Gambling mention Aldous Huxley's comment, "...the relation between craps and Reality is a real one", letter of December 1969, cited in OED. Or quote Samuelson: "When I go to a casino, I go not alone for the dollar prizes but also for the pleasures of gaming – for the soft lights and sweet music." (Econometrica, 1952, p 671).

Gauss mention J. L. Bernstein's advice, "it is wise to degauss the heads prior to each recording session." Cited in the OED.

Gödel quote Regis Debray, "Ever since Gödel showed that there does not exist a proof of the consistency of Peano arithmetic that is formalizable within that theory (1931), political

scientists had the means for understanding why it was necessary to mummify Lenin and display him to the ‘accidental’ comrades in a mausoleum.”

History Statistics with small samples. Cite Mark Twain, “History doesn’t often repeat itself, but it rhymes.”

Iidly as in “drawn iidly” –drawn as independent and identically distributed, that is somewhere between idly, and idyllically, although we shouldn’t discount the Freudian connotation “drawn by the Id.” See independence.

Image Processing Suggest the sphinx as an example of image denosing.

Index Finding your name in the index is a shortcut to reading a new book. Quote Jonathan Swift: “... to pretend to understand a book by scouting thro’ the Index, is as if a Traveller should go about to describe a Palace, when he had seen nothing but the Privy.”

Inequality The toolbox of mathematics is filled with them. Quote GH Hardy: “A science is said to be useful if its development tends to accentuate the existing inequalities in the distribution of wealth,” (A Mathematician’s Apology, p 120), but note that he later called this remark – only an excusable rhetorical flourish of 1915.

Independence The great superstition of probability theory. See Kac’s Statistical Independence, and marvel at the apparent sophistication of the Haverford undergrads in the audience. See also iidly.

Intelligence Cite GB Shaw, “it is the mark of a truly intelligent person to be moved by statistics,” then offer an award for an explicit citation, since it seems to more plausibly attributed to Bertrand Russell. See economics.

Kernel Ask, What is the canonical one? In statistics, say kernel density estimation is like making a martini: you have your gin – that is the empirical df, then you add noise – that is the vermouth, finally you stir – that is the convolution. Sip slowly while arguing about the bandwidth – the amount of vermouth. Then, ask what do kernel density estimates optimize?

Kolmogorov Thought that every single discovery should fit in a four-page Doklady note, since “the human brain is not capable of creating anything more complicated at one time.” quoted in <http://homepages.cwi.nl/~paulv/papers/kolmogorov.pdf>. Might also mention Michael Frayn: “if your vita turns out to be reasonably longa, as mine has, then to compensate your ars gets naturally more brevis.” Observer, May 3, 2020.

Liberal see conservative.

Literature Statistics with falsified data.

Love Brownian emotion.

Markov Chains Quote le Carre, I mean childhood, at my age, is no excuse for anything.

MCMC Mention that importance sampling was invented by Herman Kahn, the model for Dr. Strangelove in Kubrick’s movie of the same name. And don’t forget that Edward Teller, another candidate for Dr Strangelove, was a coauthor of the Metropolis paper. Both Kahn and Teller are members of that exclusive club of people without which the world would have been a better place. <https://xianblog.wordpress.com/2022/06/14/when-importance-was-born/>

Martingale Mention the pants of Rabelais’ Panurge, with an opening at the back “a drawbridge on the ass that makes excretion easier.” Francois I wore such clothes: “This brave knight had to have a bowel movement every time he wanted to begin fighting and therefore usually wore martingale pants.” <https://www.jehps.net/juin2009/Mansuy.pdf>

Mathematics What comes after arithmetic. Quote Kolmogorov: At any given moment there is only a fine layer between the trivial and the intractable. Mathematical discovery takes place in this layer. (diary, 14 September 1943) Should be useless according to GH Hardy, see cryptography.

Median is the message.

Mixtures cite Larry Wasserman: "I have decided that mixtures, like tequila, are inherently evil and should be avoided at all costs." ... leaving a gap in the literature filled by Kiefer and Wolfowitz (1956).

Moments mention Tukey's "misty staircase," or say "quantiles are moments too, of a sort." Or more in the spirit of Flaubert citing Voltaire, Moments are like God, if they didn't exist, it would be necessary to invent them. Pearson's method of moments should not be confused with the method of emoluments which has emerged as the dominant mechanism of modern political economy.

Normal Mention Stigler's comment, "a rare one-word oxymoron". see also Gauss.

Nostrification cite Constance Reid's biography of Courant: The Gottingers had a facetious expression for the process of making someone else's ideas one's own. They called it "nostrification." There were many levels of the process: "conscious nostrification" – "unconscious nostrification" – even "self-nostrification." This last occurred when one came up with a marvelous new idea which he later discovered had already appeared in earlier work of his own.

Ontology mention that the Peggy Lee song "Is that all there is?" written by Leiber and Stoller was inspired by the Thomas Mann (1896) short story "Disillusionment."

p-value Mention the meta-fictional treatise: Shame and the p-value. See evidence.

Paper An obsolete technology of academic communication, replaced by vapor, as in "I finished a vapor on X yesterday and submitted it to arXiv." When subjected to increased pressure, may turn into a liquid.

Pascal's Wager quote Diderot, "Any Imam could just as well reason the same way."

Poetry is about incidental parameters, everything else is about distributions.

Poker Quote Robert Pinsky quoting Walter Matthau (!) "Poker exemplifies the worst aspects of capitalism that have made our country so great."

Prices Say Kantorovich called them "expedient values" to avoid trouble with officials at Gosplan. Boldyrev and Duppe (BJHS, 2020)

Probability Flip a coin for de Finetti or Kolmogorov versions, see Chance. Quote Borel late in life: "Je vais pantoufler dans les probabilités," and if the verb isn't sufficiently clear quote Larousse: "Pour un fonctionnaire, en particulier pour un ancien élève des grandes écoles, quitter le service de l'Etat pour celui d'une entreprise privée;" Or quote Morris DeGroot commenting on IJ Good's theory of (hierarchical) "woolly probabilities: "I suppose that Poisson probabilities are woolly because their mean is lambda." [Trabajos de estadística y de investigación operativa, 1980]

Professor Quote Auden: Someone who talks in someone else's sleep. –Tony O'Hagan in LRB 23 Oct 2014.

Proof Some day soon they will all be checked by machine, useful as an aide memoire for what is claimed by the theorem. Mention Larry Shepp, who when confronted with an earlier version of one of his results responded, "yes, but when I proved it, it stayed proved." see Nostrification.

Publish or perish. If possible, follow the advice of Gian-Carlo Rota "Publish the same result several times: ... Riesz would publish the first rough version of the idea in some obscure Hungarian journal. A few years later he would send a series of notes to the French Academy's Comptes Rendus in which the material was further elaborated. A few more years would pass, and he would publish the definitive paper, either in French or in English." [Indiscrete Thoughts], advice only available to the multi-lingual. See nostrification. On what to publish cite Halmos (AMM, 1975): "A striking criterion for how to decide not to publish something was offered by my colleague John Conway. Suppose that you have just finished typing a paper. Suppose now that I come to you, horns, cloven hooves, forked tail and all, and ask: if I gave you \$1000.00, would you tear the paper up and forget it? If you hesitate, your paper is lost - do not publish it. That's part of a more general rule: when in doubt, let the answer be no." Don't forget to update the cash allowance to current dollars.

Quantiles Are moments too, of a sort.

R what comes after Q. Quote Virginia Woolf (To the Lighthouse): "It was a splendid mind. For if thought is like the keyboard of a piano, divided into so many notes, or like the alphabet is ranged in twenty-six letters all in order, then his splendid mind had no sort of difficulty in running over those letters one by one, firmly and accurately, until it had reached the letter Q. He reached Q. Very few people in the whole of England reach the letter Q.... But after Q? What comes next?... Still, if he could reach R it would be something. Here at least was Q. He dug his heels in at Q. Q he was sure of. Q he could demonstrate. If Q then is Q-R-.... "Then R..." He braced himself. He clenched himself. "...In that flash of darkness he heard people saying—he was a failure—that R was beyond him. He would never reach R. On to R, once more. R—.... "...He had not genius; he had no claim to that: but he had, or he might have had, the power to repeat every letter of the alphabet from A to Z accurately in order. Meanwhile, he stuck at Q. On then, on to R."

Random Numbers Mention von Neumann's comment: "Any one who considers arithmetical methods of producing random digits is, of course, in a state of sin." as quoted by Knuth, ACP:V2.

Refereeing Mention that Dante, who missed out on the experience of peer review failed to anticipate the overcrowding of purgatory while St Peter tried to work through his backlog of "refereeing of souls."

Regression is demeaning.

Robustness Burned with such intensity that, like Marxism-Leninism, only the ashes of the most pure remain.

Simulation What makes statistics an experimental science. See Random Numbers. Description of the results of simulation experiments is called dissimulation. Ask: Have you read Francis Bacon's essay "Simulation and Dissimulation?"

Smoking Say R.A. Fisher found it healthy and became wealthy. Add that Joseph Berkson Chief Statistician at Mayor Clinic from 1934-64, also subscribed to this view.

Statistics The (futile) attempt to offer certainty about uncertainty. Cite Kafka, "I have reached the stage that I no longer desire certainty."

Structural estimation see strictural estimation.

Strictural estimation estimation subject to one or more strictures, cite OED – "There is no strictural obstruction to the progress of the faeces." J.M. Duncan (1886).

tenure say Thorstein Veblen is the patron saint of the untenured, having been forced to resign at both University of Chicago and Stanford, and resigning voluntarily from the University of Missouri without having tenure. One could also nominate C.S. Peirce who after finally getting a lecturer position at Hopkins was, like Veblen, forced to resign for traveling with a woman who was not his wife.

Test Quote "Joe" the CIA's chief "expert" on Iraq's aluminum tubes and their potential role for WMD, testifying in Congress on why he didn't consult Energy Department experts on the subject: "Because we funded it. It was our testing. We were trying to prove some things that we wanted to prove with the testing." (NYTimes, Oct 3, 2004). Mention that in sequential analysis this is called testing to a foregone conclusion.

Tukey "never estimate intercepts, always estimate centercepts!" Quote von Neumann's comment: "There is this very bright graduate student, and the remarkable thing is that he does it all on milk." (Brillinger's (2002) obituary in the AMS Notices.) Tukey on Friedman: Milton was well acquainted with the statistical side and very sharp. Probably easier to cut yourself on him than anybody else in the Biometric Society or the Institute [of Advanced Study.]

Tuning parameter What the Bayesians call the "degree of belief in the prior", say neither frequentists or Bayesians know how to select them.

Truth see error. Ask: is it true that Somerset Maugham coined the phrase "Nothing is too rum to be true"? Corde says, "I heard it years ago on a Channel train from an old English gentleman. He was a lifelong atheist, but maybe God did exist, he said." Saul Bellow, *Dean's* December.

University The soft underbelly of capitalism, or cite Veblen's *The Higher Learning in America*, originally subtitled "A Study in Total Depravity," or say, "higher education – the highest form of money laundering."

Writing mention Geoff Dyer: "a way of postponing the day when I won't do it any more, the day when I will sink into a depression so profound it will be indistinguishable from perfect bliss."

B

A Simple Protocol for Simulations in R

1

B.1 Introduction

It is easy to get sloppy with the organization of simulation exercises and this usually results in painful experiences at the latter stages of projects, and ultimately to the irreproducibility of results. See the entry on reproducibility in the Dictionary of Received Ideas in Statistics. If one could adhere to a simple, consistent protocol I believe many of these problems could be avoided.

B.2 The Four Commandments

I propose the following rules:

- 1 Always use `set.seed( )`.
- 2 Never run simulations interactively, always run them from source files.
- 3 Document the initial environment.
- 4 Save the final environment.

B.3 Some Implementation Details

There are undoubtedly many ways to implement these rules; I will briefly sketch one way that seems to be quite convenient. Suppose that we create a source file called `sim.R` that looks like this:

¹ This protocol describes my evolving strategy for doing numerical simulations. It is undoubtedly already badly dated, especially the discussion of interacting with clusters, but I hope it still retains some useful avuncular advice.

```
# toy MC experiment to test the Gossett binary response estimator

source("plink.R")
system("hostname")
date()
sessionInfo()

R <- 500
n <- 500
set.seed(1968)
x <- 5*rnorm(n)
dfs <- c(1,2,6)
A <- array(0,c(3,3,R))
for(i in 1:length(dfs)){
  df <- dfs[i]
  print(paste("i = ",i))
  for(j in 1:R){
    y <- (1.0 * x + rt(n,df) > 0)
    f <- pglm(y ~ x,link="Gossett")
    A[,i,j] <- c(f$nuhat,f$nu0,f$nu1)
  }
}
```

We begin by `source()`ing some functions from the file `plink.R`. These functions and any data residing in `plink.R` will be saved as part of the global environment at the end of the process and will therefore be available for post-mortem analysis. Next we record the name of the machine, the date and some basic information about the version of R and the versions of the packages that we have attached. Then we initialize the seed for the random number generator and get to work. The print statement is used to monitor progress of the job by occasionally peeking in the resulting `sim.Rout` file using, e.g. the shell command `tail`. In more complicated situations it would often be preferable to write a function that encapsulated a single iteration of the simulation.

The file `sim.R` would normally be invoked from the command line with,

```
R CMD BATCH sim.R
```

By so doing we automatically generate two new files: `sim.Rout` which contains a transcript of the executed session, and `.RData` which by default contains a binary version, in standard R format, of the global environment including all functions and data as it existed at the end of the session. It seems prudent to rename this file to something more meaningful. The process can be automated by the following shell script: These four rules produce a pair of new files `sim.Rout` and `sim.Rda`.

```
R CMD BATCH $1.R
mv .RData $1.Rda
chmod -w $1.R
```

This approach enforces a consistent naming convention. Calling the script `Rbatch`, one would simply invoke it with

`Rbatch sim &`

This automatically moves the standard output file `.RData` created by the simulation into a more specifically named file `sim.Rda` that can be preserved for posterity.

Remarks

1. The use of `set.seed( )` is essential if one wants to ensure the reproducibility of results. The R random number generators allow one to restart the sequence by simply resetting the seed to the same value that was used previously.
2. The batch approach provides an explicit record of what was done. It also, conveniently, provides automatic timing information on the run at the end of the `sim.Rout` file.
3. The `sessionInfo( )` call documents the version of R and any included packages, thereby enabling restoration of the environment used to create the results. Note that prior versions of R and its packages are available from CRAN. This resolves a long-standing problem with proprietary software for which it was difficult, or even impossible, to restore prior versions of the software to repeat a prior analysis.
4. Use of `save.image( )` at the end of the session, which occurs automatically (by default with the R `CMD BATCH` command), ensures a complete record of what was produced by the originating batch file, and can be easily restored using `load( )`. This allows one to separate the relatively time consuming simulation phase of the project from the analysis phase. The latter typically requires relatively little computational effort but may involve considerable fine tuning to produce appropriate tables and graphics. It is advantageous to have separate files for each table and figure, each of which can rely on `load` to recreate the output of the simulation. For procedures that produce large `.Rda` files one could also add a compression step. The latter can be accomplished automatically by setting,


```
options(save.image.defaults = list(compress=TRUE))
```

 either in the `sim.R` file or in the `.Rprofile` file that gets sourced automatically at startup.
5. The array `A` that is produced by the simulation can usually be easily manipulated using R's `apply` function to produce tables and or figures as appropriate. For tables intended for `LATEX` documents there are several very useful tools for semi-automatically generating `LATEX` code available from the `Hmisc` package of Frank Harrell. The `quantreg` package also contains a `latex.table` function adapted from an early version of Harrell's function of the same name.

B.4 The `foreach` Wrinkle

The `doMC` package provides a convenient way to use multicore machines to do simply parallelized simulations, but it also offers some opportunities to destroy the reproducibility of what seemed to be a simple simulation exercise. I think that I now understand how to rectify this. The basic issue is: How does one insure that each of the pieces of a `foreach` loop starts with the same seed? If this behavior is not what is desired, then I have no advice.² An important caveat – thanks to Tom Parker for this – is that one needs to be sure that

² It may seem weird to have each instance of the loop starting from the same seed, but the way that I usually do simulations it seems natural: each instance of the `foreach` typically represents a

the number of cores used in the foreach doesn't exceed the number of cores available on the machine being used, if it does then seeds are updated when cores are recycled.

Here is a some test code to illustrate how this works:

```
# Test of set.seed of foreach command
  require(doMC) #loads multicore and foreach automatically.
  registerDoMC(5) # intention to use 5 cores
# Now the setup for the simulation
date()
system("hostname")
sessionInfo()
set.seed(1968)
J <- 5
opt <- list(set.seed = FALSE)
AJ <- foreach(j = 1:J, .options.multicore = opt) %dopar% {
  A <- sum(rnorm(10))
}
```

As we can see from examining the AJ object produced, each component of the list is identical. In this code we have to set the options for multicore in a somewhat unintuitive way in order to get each of the processes to use the same initial seed. But once this is done things seem to go as desired.

Of course if we don't want to use the same seed in each foreach instance then we can simplify this somewhat.

```
# Test of set.seed of foreach command
  require(doMC) #loads multicore and foreach automatically.
  registerDoMC(5) # intention to use 5 cores
# Now the setup for the simulation
date()
system("hostname")
sessionInfo()
J <- 5
seeds <- 1:J
AJ <- foreach(j = 1:J) %dopar% {
  set.seed(seeds[j])
  A <- sum(rnorm(10))
}
```

In this case we know how to assign seeds to each instance and all the seeds and results are available in the final `sim.Rda` object.

different distributional assumption, or sample size, or some other feature of the model, so starting each at the same seed means that any one of the instances can be reproduced by simply doing it individually using the original seed.

B.5 The Cluster

The next stage of complication is expanding the use of `foreach` from a desktop machine like my mac pro to a cluster like UIUC's new LAS cluster. For the moment I have a rather primitive elaboration of the foregoing that seems to work. I would appreciate any suggestions of refinements. Essentially this requires replacing the parallel backend `doMC` by the cluster version `doMPI`.

The easy part of the cluster transition was getting R installed and in my case installing Mosek and some related packages, this was all quite straightforward and could be done without any administrative privileges within my own home directory.

Once this was done I created a sbatch file `foo.s` that looked like this:

```
#!/bin/tcsh

#SBATCH --job-name=foo
#SBATCH --partition=e
#SBATCH -n 21
#SBATCH --time=48:00:00
#SBATCH --mem-per-cpu=2048
#SBATCH --mail-type=FAIL
#SBATCH --mail-type=END
#SBATCH --mail-user=rkoenker@illinois.edu

mpirun -np 21 R CMD BATCH foo.R
```

This provides a jobname, specifies that the job is destined for the economics nodes, a maximum time limit of 48 hours, maximum memory of 2G, etc. The `mpirun` command runs a R batch job specified by the file `foo.R`. A simple example `foo.R` looks like this:

```
# Pseudo simulation exercise for testing script to automate cluster job control
# The specifics are obviously silly since this could be done more efficiently.

require(doMPI)
cl <- startMPIcluster()
registerDoMPI(cl)

date()
system("hostname")
sessionInfo()
opt <- list(set.seed = FALSE)

set.seed(22)
R <- 50
trims <- (1:7)/20
```

```

ns <- c(100,500,1000)
dfs <- 1:5
G <- expand.grid(ns,dfs)
files <- paste("f",1:nrow(G), sep = "")
A <- array(0, c(length(trims), R))
AK <- foreach(k = 1:nrow(G), .options.multicore = opt) %dopar% {
  n <- G[k,1]
  df <- G[k,2]
  file <- files[k]
  for(j in 1:R){
    x <- rt(n, df)
    for(i in 1:length(trims)){
      A[i,j] <- mean(x, trim = trims[i])
    }
    cat(j, "\n", file = file) # Monitor progress
  }
  A
}
save.image()
closeCluster(cl)
mpi.finalize()

```

Now provided that both `foo.R` and `foo.s` are in the same directory on the cluster I can simply invoke:

```
sbatch foo.s
```

to run the job, which will then get distributed to the various nodes of the cluster.

The job runs producing the usual log file called in this case `foo.Rout` and an output file called `.RData` containing the results of the job. Apparently, the R packages are found and promulgated to the nodes. As I understand it, on the LAS cluster we have 12 core machines so I suppose if one wanted 48 cores, one could rescale the “-n” line in `Rbatch`, but I’ve not yet tested this. There are a variety of open questions about how to deal with scheduler options, that I also hope to sort out soon.

The downside of initializing the job on the cluster is that it creates lots of opportunities for file name confusion since there are multiple copies of files on one’s desktop machine and on the cluster. And the file `foo.s` seems a bit redundant; since it would seem to be easily constructed automatically, thereby avoiding clashes of filenames. In this spirit, I’ve tried to construct a shell script that would generate a `foo.s` file send it and the associated `foo.R` file to the cluster, execute the job, and hopefully return the output files to my desktop machine. This is a work in progress, but the first stages seem to do what was intended. I will update this as I learn more about how to do things. (This involves relearning what little I once knew about shell programming from ancient times at Bell Labs.) The following (Bourne) shell script lives in my `bin` directory as `Cbatch`.

```
#!/bin/sh
```

```

pwd=$PWD
while :
do
    case "$1" in
    -n) shift; n="$1";;
    -*) usage "bad argument $1";;
    *) break;;
    esac
    shift
done
job=$1

cat > $job.s <<%
#!/bin/tcsh

#SBATCH --job-name=$job
#SBATCH --partition=e
#SBATCH -n $n
#SBATCH --time=48:00:00
#SBATCH --mem-per-cpu=2048
#SBATCH --mail-type=FAIL
#SBATCH --mail-type=END
#SBATCH --mail-user=rkoenker@illinois.edu

mpirun -np $n R CMD BATCH $job.R
mv .RData $job.Rda
scp $job.Rda $job.Rout roger@yzzy.econ.illinois.edu:$pwd
%
scp $job.R $job.s rkoenker@keeling:SimBin
ssh rkoenker@keeling "cd SimBin; sbatch $job.s"

```

Obviously, this assumes that loginless access to the cluster has been setup for ssh, and the short pathname for keeling is known. Equally obviously, it assumes that access is permitted the other direction so the output files can be returned to the desktop machine. This is what is happening at the very end of the so-called “Here document,” i.e. in the scp line before the trailing percent sign. The `$pwd` variable assures that the output files go back to the directory from whence they were generated. Finally, any files that the job expects to be able to source have to already be in the directory `SimBin`.

This script is invoked on my desktop like this:

```
Cbatch -n 15 foo
```

I suppose that it should go without saying, but since I seem to have a difficult time adhering to this procedure I’ll say it anyway: Always run a prototype simulation with a trivial number of replications on a noncluster machine for debugging purposes. It is very difficult to track down bugs on the cluster since various pieces of the job are running on

different nodes and cores. Running the simulation in a simply nested for loop allows one to use `traceback` and other debugging tools. Another useful strategy is to combine outer loops using `expand.grid()`, as illustrated in the example above. I've also found it useful to include the file progress monitor strategy adopted there, in case there is a crash and one would like to reconstruct the data for which it occurred. At the very least running a small version of the simulation permits one to get a rough estimate of the time to completion for the full version.

References

- ADAMASZEK, M. (2021): “Google Groups Mosek Conversations,” Personal Communication: Available at: <https://groups.google.com/g/mosek/c/T2MYKTc8uD4>.
- ANDERSEN, E. D. (2010): “The Mosek Optimization Tools Manual, Version 6.0,” Available from <http://www.mosek.com>.
- APS, M. (2018): *Rmosek Release 8.1.47*.
- (2019): *Rmosek Release 9.0.89*.
- ARELLANO, M., AND S. BONHOMME (2017): “Quantile Selection Models,” *Econometrica*, forthcoming.
- ARMSTRONG, T. B., M. KOLESÁR, AND M. PLAGBORG-MØLLER (2020): “Robust Empirical Bayes Confidence Intervals,” <http://arxiv.org/abs/2004.03448>.
- ARROW, K. (1958): “Historical Background,” in *Studies in the Mathematical Theory of Inventory and Production*, ed. by K. Arrow, S. Karlin, and H. Scarf. Stanford U. Press.
- (2002): “The Genesis of ”Optimal Inventory Policy”,” *Operations Research*, 50, 1–2.
- ARROW, K., T. HARRIS, AND J. MARSCHAK (1951): “Optimal Inventory Policy,” *Econometrica*, 19, 250–272.
- ARTZNER, P., F. DELBAEN, J.-M. EBER, AND D. HEATH (1999): “Coherent Measures of Risk,” *Math. Finance*, 9, 203–228.
- AVRIEL, M. (1972): “ r -Convex Functions,” *Math. Programming*, 2, 309–323.
- BANG, H., AND J. M. ROBINS (2005): “Doubly robust estimation in missing data and causal inference models,” *Biometrics*, 61, 962–973.
- BARBOR, R. F., E. J. CANDÈS, A. RAMDAS, AND R. J. TIBSHIRANI (2020): “The limits of distribution-free conditional predictive inference,” Available from: ArXiv 1903.04684v2.
- BARRODALE, I., AND F. ROBERTS (1974): “Solution of an overdetermined system of equations in the ℓ_1 norm,” *Communications of the ACM*, 17, 319–320.
- BARRON, A., AND C. SHUE (1991): “Approximation of density functions by sequences of exponential families,” *Annals of Statistics*, 19, 1347–1369.
- BARZILAI, J., AND J. M. BORWEIN (1988): “Two-point step size gradient methods,” *IMA J. Numerical Analysis*, 8, 141–48.
- BASSETT, GILBERT W., J., R. KOENKER, AND G. KORDAS (2004): “Pessimistic Portfolio Allocation and Choquet Expected Utility,” *Journal of Financial Econometrics*, 2, 477–492.
- BELLEMARE, M. G., Y. NADDAF, J. VENESS, AND M. BOWLING (2013): “The arcade learning environment: an evaluation platform for general agents,” *J. Artif. Int. Res.*, 47, 253–279.
- BELLONI, A., AND V. CHERNOZHUKOV (2011): “ ℓ_1 -penalized quantile regression in high-dimensional sparse models,” *The Annals of Statistics*, 39, 82–130.

- BHATTACHARYA, A., D. PATI, N. S. PILLAI, AND D. B. DUNSON (2012): “Bayesian shrinkage,” *arXiv:1212.6088*.
- BICKEL, P. J., AND D. A. FREEDMAN (1981): “Some Asymptotic Theory for the Bootstrap,” *The Annals of Statistics*, 9, 1196–1217.
- BICKEL, P. J., AND A. SAKOV (2008): “On the choice of m in the m out of n bootstrap and confidence bounds for extrema,” *Statistica Sinica*, 18, 967–985.
- BOLLEN, K., AND J. PEARL (2013): “Eight Myths about Causality and Structural Equation Models,” in *Handbook of Causal Analysis for Social Research*, ed. by S. Morgan, pp. 301–328. Springer.
- BORELL, C. (1975): “Convex set functions in d -space,” *Periodica Math. Hungarica*, 6, 111–136.
- BOSCH, R. J., Y. YE, AND G. G. WOODWORTH (1995): “A Convergent Algorithm for Quantile Regression With Smoothing Splines,” *Computational Statistics and Data Analysis*, 19, 613–630.
- BOTTOU, L., AND B. SCHÖLKOPF (2023): “Borges and AI,” Available from <https://arxiv.org/abs/2310.01425>.
- BRACHER, J., E. L. RAY, T. GNEITING, AND N. G. REICH (2021): “Evaluating epidemic forecasts in an interval format,” *PLOS Computational Biology*.
- BRADLEY, R. A., AND M. E. TERRY (1952): “Rank Analysis of Incomplete Block Designs: I. The Method of Paired Comparisons,” *Biometrika*, 39, 324–345.
- BREIMAN, L. (1996): “Stacked regressions,” *Machine Learning*, 24, 49–64.
- BUCHINSKY, M. (1994): “Changes in US Wage Structure 1963–87: An Application of Quantile Regression,” *Econometrica*, 62, 405–458.
- CANDES, E. (2007): “ ℓ_1 Magic,” <https://candes.su.domains/software/l1magic/>.
- CANDES, E., AND T. TAO (2007): “The Dantzig selector: Statistical estimation when p is much larger than n ,” *The Annals of Statistics*, 35, 2313 – 2351.
- CANDES, E. J., AND P. A. RANDALL (2008): “Highly Robust Error Correction by Convex Programming,” *IEEE Trans. Inf. Theor.*, 54, 2829 – 2840.
- CARVALHO, C., N. POLSON, AND J. SCOTT (2009): “Handling sparsity via the horseshoe,” *Journal of Machine Learning Research*, 5, 73–80.
- CASELLA, G., AND E. GEORGE (1992): “Explaining the Gibbs sampler,” *American Statistician*, 46, 167–174.
- CASTILLO, I., AND A. VAN DER VAART (2012): “Needles and Straw in a Haystack: Posterior Concentration for Possibly Sparse Sequences,” *Annals of Statistics*, 40, 2069–2101.
- CATTELAN, M. (2012): “Models for Paired Comparison Data: A Review with Emphasis on Dependent Data,” *Statistical Science*, 27, 412 – 433.
- CHAMBERS, J. M., D. A. JAMES, D. LAMBERT, AND S. VANDER WIEL (2006): “Monitoring Networked Applications With Incremental Quantile Estimation,” *Statistical Science*, 21, 463–475.
- CHERNOZHUKOV, V., I. FERNANDEZ-VAL, AND A. GALICHON (2009): “Improving point and interval estimators of monotone functions by rearrangement,” *Biometrika*, p. asp030.
- CHERNOZHUKOV, V., I. FERNANDEZ-VAL, AND T. KAJI (2018): “Extremal Quantile Regression,” in *Handbook of Quantile Regression*, ed. by R. Koenker, V. Chernozhukov, X. He, and L. Peng. CRC Press.
- CHERNOZHUKOV, V., I. FERNÁNDEZ-VAL, AND B. MELLY (2013): “Inference on counterfactual distributions,” *Econometrica*, forthcoming.
- CHERNOZHUKOV, V., I. FERNANDEZ-VAL, AND B. MELLY (2019): “Fast Algorithms for the Quantile Regression Process,” available from: arXiv, 1909.05782.
- CHERNOZHUKOV, V., AND C. HANSEN (2004): “An IV Model of Quantile Treatment Effects,” *Econometrica*, 73, 245–261.
- CHESHER, A. (2003): “Identification in Nonseparable Models,” *Econometrica*, 71, 1405–1441.
- CHOI, T.-M. (ed.) (2012): *Handbook of Newsvendor Problems*. Springer.
- COX, D. (1966): “Notes on the analysis of mixed frequency distributions,” *The British Journal of Mathematical and Statistical Psychology*, 19, 39–47.

- (1984): “Interaction,” *International Statistical Review*, 52, 1–31.
- CULE, M., R. SAMWORTH, AND M. STEWART (2010): “Computing the Maximum Likelihood Estimator of a Multidimensional Log-Concave Density, with discussion,” *Journal of the Royal Statistical Society, B.*, 72, 545–600.
- DAVID, H. A. (1988): *The method of paired comparisons*. Oxford University Press, 2nd edn.
- DAVIES, P. L., AND A. KOVAC (2001): “Local extremes, runs, strings and multiresolution,” *The Annals of Statistics*, 29, 1–65.
- (2004): “Densities, Spectral Densities and Modality,” *The Annals of Statistics*, 32, 1093–1136.
- DE LEEUW, J., K. HORNIK, AND P. MAIR (2009): “Isotone Optimization in R: Pool-Adjacent-Violators Algorithm (PAVA) and Active Set Methods,” *Journal of Statistical Software, Articles*, 32, 1–24.
- DHARMADHIKARI, S., AND K. JOAG-DEV (1988): *Unimodality, Convexity and Applications*. Academic Press, Boston.
- DIACONIS, P. (2009): “Review of *Probabilistic symmetries and invariance principles*, by Olav Kallenberg,” *AMS Bulletin*, 46, 691–696.
- DIACONIS, P., AND D. FREEDMAN (1981): “On the Statistics of Vision: The Julesz Conjecture,” *J. of Mathematical Psychology*, 24, 112–138.
- DÍAZ, I. (2015): “Efficient Estimation of Quantiles in Missing Data Models,” *arXiv:1512.08110*.
- DÍAZ, I., AND M. ROSENBLUM (2015): “Targeted Maximum Likelihood Estimation using Exponential Families,” *The international journal of biostatistics*, 11, 233–251.
- DOKSUM, K., AND M. GASKO (1990): “On a correspondence between models in binary regression and survival analysis,” *Intl. Stat. Rev.*, 58, 243–252.
- DOLL, R., R. PETO, J. BOREHAM, AND I. SUTHERLAND (2004): “Mortality in relation to smoking: 50 years observations on male British doctors,” *British Medical Journal*, 328.
- DONOHU, D. (2024): “Data Science at the Singularity,” *Harvard Data Science Review*, 6.
- DOSS, C. R., AND J. A. WELLNER (2016): “Global rates of convergence of the MLEs of log-concave and s -concave densities,” *Annals of Statistics*, 44, 954–981.
- DUNSON, D., AND J. TAYLOR (2005): “Approximate Bayesian inference for quantiles,” *Journal of Nonparametric Statistics*, 17(3), 385–400.
- EDGEWORTH, F. (1888): “A Mathematical Theory of Banking,” *J. Royal Stat Soc*, 51, 113–127.
- EFRON, B. (2011): “Tweedie’s Formula and Selection Bias,” *Journal of the American Statistical Association*, 106, 1602–1614.
- (2016): “Empirical Bayes Deconvolution Estimates,” *Biometrika*, 103, 1–20.
- EFRON, B. (2019): “Bayes, Oracle Bayes and Empirical Bayes,” *Statistical Science*, 34, 177–201.
- EMPSON, W. (1930): *Seven Types of Ambiguity*. Chatto and Windus.
- EPPSTEIN, D. (2007): “Blum-style analysis of Quickselect,” Available from <https://11011110.github.io/blog/2007/10/09/blum-style-analysis-of.html>.
- FASIOLO, M., Y. GOUDE, R. NEDELLEC, AND S. N. WOOD (2017): “Fast calibrated additive quantile regression,” Available from <https://arxiv.org/abs/1707.03307>.
- FENG, X., X. HE, AND J. HU (2011): “Wild Bootstrap for Quantile Regression,” *Biometrika*, 98, 995–999.
- FERNANDES, M., E. GUERRE, AND E. HORTA (2021): “Smoothing Quantile Regressions,” *Journal of Business & Economic Statistics*, 39, 338–357.
- FIRPO, S. (2007): “Efficient semiparametric estimation of quantile treatment effects,” *Econometrica*, 75, 259–276.
- FIRPO, S., N. FORTIN, AND T. LEMIEUX (2009): “Unconditional Quantile Regressions,” *Econometrica*, 77, 953–973.
- FLOYD, R. W., AND R. L. RIVEST (1975): “Algorithm 489: The Algorithm SELECT for Finding the i th Smallest of n elements,” *Comm. ACM*, 18, 173.

- FOUGNER, C. (2014): *POGS: Proximal Operator Graph Solver* <https://github.com/foges/pogs>.
- FRIBERG, H. A. (2012): “Users Guide to the R-to-Mosek Interface,” Available from <http://rmosek.r-forge.r-project.org>.
- GEISSER, S. (1975): “The predictive sample reuse method with applications,” *J. of Am. Statistical Association*, 70, 320–328.
- GELMAN, A., AND J. MIKHAEL (2025): “Russian roulette: the need for stochastic potential outcomes when utilities depend on counterfactuals,” *Biometrika*, 112.
- GLICKMAN, M. E. (2013): “Introductory Note to 1928,” Available from <http://www.glicko.net/research/preface-z28.pdf>.
- GNETING, T., AND A. E. RAFTERY (2007): “Strictly proper scoring rules, prediction, and estimation,” *J Am Stat Assoc.*, 102, 359–378.
- GOLDBERGER, A. S. (1962): “Best Linear Unbiased Prediction in the Generalized Linear Model,” *Journal of the American Statistical Society*, 57, 369–375.
- GOOD, I. J. (1971): “A nonparametric roughness penalty for probability densities,” *Nature*, 229, 29–30.
- GROENEBOOM, P., AND G. JONGBLOED (2014): *Nonparametric Estimation under Shape Constraints: Estimators, Algorithms and Asymptotics*. Cambridge University Press.
- GROENEBOOM, P., G. JONGBLOED, AND J. A. WELLNER (2001): “Estimation of a Convex Function: Characterizations and Asymptotic Theory,” *The Annals of Statistics*, 29(6), 1653–1698.
- GU, J., AND S. VOLGUSHEV (2019): “Panel data quantile regression with grouped fixed effects,” *Journal of Econometrics*, 213, 68–91.
- GUVENEN, F., F. KARAHAN, S. OZKAN, AND J. SONG (2016): “What Do Data on Millions of U.S. Workers Reveal about Life-Cycle Earnings Dynamics?,” Federal Reserve Bank of New York Staff Reports.
- HAGEMANN, A. (2017): “Cluster-Robust Bootstrap Inference in Quantile Regression Models,” *Journal of the American Statistical Association*, 112, 446–456.
- HAMMER, H. L., A. YAZIDI, AND H. RUE (2019): “A new quantile tracking algorithm using a generalized exponentially weighted average of observations,” *Applied Intelligence*, 49.
- HAN, Q., AND J. A. WELLNER (2016): “Approximation and estimation of s-concave densities via Rényi divergences,” *Annals of Statistics*, 44, 1332–1359.
- HARDY, G. H., J. E. LITTLEWOOD, AND G. PÓLYA (1934): *Inequalities*. Cambridge U. Press, London.
- HARTIGAN, J., AND P. HARTIGAN (1985): “The dip test of unimodality,” *Annals of Statistics*, 13, 70–84.
- HE, X., X. PAN, K. M. TAN, AND W.-X. ZHOU (2020): *conquer: Convolution-Type Smoothed Quantile Regression* <https://CRAN.R-project.org/package=conquer>.
- HECKMAN, J. J. (1979): “Sample Selection Bias as a Specification Error,” *Econometrica*, 47, 153–161.
- HILL, B. (1975): “A simple general approach to inference about the tail of a distribution,” *The Annals of Statistics*, 3(5), 1163–1174.
- HOARE, C. A. R. (1961): “Algorithm 65: Find,” *Comm. ACM*, 4, 321–322.
- (1962): “Quicksort,” *Commuter Journal*, 5, 10–16.
- HOCKING, T. D., A. JOULIN, F. BACH, AND J.-P. VERT (2011): “Clusterpath: an algorithm for clustering using convex fusion penalties,” *Proceedings of the 28th International Conference on Machine Learning*, pp. 745–752.
- HOFFLEIT, D., AND W. H. WARREN (1991): *The Bright Star Catalog (5th ed.)*. Yale University Observatory, New Haven.
- HOLST, U. (1985): “Recursive Estimation of Quantiles,” in *Contributions to Probability and Statistics in Honour of Gunar Blom*, ed. by G. Lindgren, pp. 179–188. Lund University Press.

- HORVITZ, D. G., AND D. J. THOMPSON (1952): "A generalization of sampling without replacement from a finite universe," *Journal of the American statistical Association*, 47, 663–685.
- IBRAGIMOV, I. A. (1956): "On the composition of unimodal distributions," *Theory of Probability and its Applications*, 1, 255–260.
- JACKMAN, S. (2009): *Bayesian Analysis for the Social Sciences*. Wiley, New York.
- JEFFREYS, S. (1939): *Theory of probability*. Oxford University Press.
- JIANG, W. (2019): "Comment: Empirical Bayes Interval Estimation," *Statistical Science*, 34, 219–223.
- JIANG, W., AND C.-H. ZHANG (2009): "General maximum likelihood empirical Bayes estimation of normal means," *Annals of Statistics*, 37, 1647–1684.
- JOHNSTONE, I., AND B. SILVERMAN (2004): "Needles and straw in haystacks: Empirical Bayes estimates of possibly sparse sequences," *Annals of Statistics*, 32, 1594–1649.
- JONGBLOED, G. (2001): "Sieved Maximum Likelihood Estimation in Wicksell's problem and related deconvolution problems," *Scandinavian Journal of Statistics*, 28, 161–183.
- KAHNEMAN, D., AND A. DEATON (2010): "High income improves evaluation of life but not emotional well-being," *Proc. Natl. Acad. Sci.*, 107, 16489–16493.
- KANG, J. D., AND J. L. SCHAFER (2007): "Demystifying double robustness: A comparison of alternative strategies for estimating a population mean from incomplete data," *Statistical science*, pp. 523–539.
- KATO, K. (2011): "Group Lasso for high dimensional sparse quantile regression models," Available from <https://arxiv.org/abs/1103.1458>.
- KIEFER, J., AND J. WOLFOWITZ (1956): "Consistency of the Maximum Likelihood Estimator in the Presence of Infinitely Many Incidental Parameters," *The Annals of Mathematical Statistics*, 27, 887–906.
- KILLINGSWORTH, M. A., D. KAHNEMAN, AND B. MELLERS (2023): "Income and emotional well-being: A conflict resolved," *Proc. Natl. Acad. Sci.*, 120.
- KIWIEL, K. C. (2005): "On Floyd and Rivest's SELECT algorithm," *Theoretical Computer Science*, 347, 214–238.
- KLEINER, A., A. TALWALKAR, P. SARKAR, AND M. JORDAN (2014): "A Scalable bootstrap for massive data," *J. Royal Statistical Society (B)*, 76, 795–816.
- KNUTH, D. K. (1998): *The Art of Computing Programming: Sorting and Searching*, vol. 3. Addison-Wesley, 2nd edn.
- KOCHERGINSKY, M., X. HE, AND Y. MU (2004): "Practical Confidence Intervals for Regression Quantiles," *J. of Comp. and Graphical Stat.*, forthcoming.
- KOENKER, R. (1984): "A Note on L-estimators for Linear Models," *Stat and Prob Letters*, 2, 323–325.
- KOENKER, R. (1996): "rqx: Barrodale and Roberts lite," Available from: <http://www.econ.uiuc.edu/~roger/research/rq/rqx.R>.
- (2005): *Quantile Regression*. Cambridge U. Press.
- (2011a): "Additive models for quantile regression: Model selection and confidence band-aids," *Brazilian Journal of Probability and Statistics*, 25, 239–262.
- (2011b): "On Distributional vs. Quantile Regression," Available from <http://www.econ.uiuc.edu/~roger/research/vinaigrettes/dreg.pdf>.
- (2013): *REBayes: Empirical Bayes Estimation and Inference in R*, R package version 0.39, <http://www.r-project.org>.
- (2014): "A Protocol for Simulations in R," <http://www.econ.uiuc.edu/~roger/research/repro/protocol.pdf>.
- (2016a): "Lecture 20: Binary Response Models," Available from: <http://www.econ.uiuc.edu/~roger/courses/508/lectures/L20.pdf>.
- (2016b): "Model Selection and Fishing for Significance," Available from <http://www.econ.uiuc.edu/~roger/courses/508/lectures/L4.pdf>.

- KOENKER, R. (2018): "Computational Methods for Quantile Regression," in *Handbook of Quantile Regression*, ed. by R. Koenker, V. Chernozhukov, X. He, and L. Peng. CRC Press.
- (2020): "Quantile Regression Methods: An R Vinaigrette," Available from <http://www.econ.uiuc.edu/~roger/research/vinaigrettes/vinaigrette.html>.
- KOENKER, R., AND V. D'OREY (1987): "Computing Regression Quantiles," *Applied Statistics*, 36, 383–393.
- KOENKER, R., AND O. GELING (2001): "Reappraising Medfly Longevity: A Quantile Regression Survival Analysis," *J. of Am. Stat. Assoc.*, 96, 458–468.
- KOENKER, R., AND J. GU (2017): "REBayes: An R Package for Empirical Bayes Mixture Methods," *Journal of Statistical Software*, 82, 1–25.
- KOENKER, R., AND I. MIZERA (2007): "Density Estimation by Total Variation Regularization," in *Advances in Statistical Modeling and Inference: Essays in Honor of Kjell Doksum*, ed. by V. Nair, pp. 613–633. World Scientific.
- (2008): "Primal and dual formulations relevant for the numerical estimation of a probability density via regularization," in *Tatra Mountains Mathematical Publications*, ed. by A. Pázmán, J. Volaufová, and V. Witkovský, vol. 39, pp. 255–264. Slovak Academy of Sciences, Proceedings of the conference ProbaStat '06 held in Smolenice, Slovakia, June 5–9, 2006.
- (2010): "Quasi-concave density estimation," *Annals of Statistics*, 38, 2998–3027.
- KOENKER, R., AND I. MIZERA (2013): "Convex Optimization, Shape Constraints, Compound Decisions, and Empirical Bayes Rules," *J of American Statistical Association*, forthcoming, available at: <http://www.econ.uiuc.edu/~roger/research/ebayes/ebayes.html>.
- KOENKER, R., AND I. MIZERA (2019): "Shape Constrained Density Estimation Via Penalized Rényi Divergence," *Statistical Science*, 33, 510–526.
- KOENKER, R., P. NG, AND S. PORTNOY (1994): "Quantile Smoothing Splines," *Biometrika*, 81, 673–80.
- KOENKER, R., AND Z. XIAO (2002): "Inference on the quantile regression process," *Econometrica*, 70, 1583–1612.
- KOENKER, R., AND J. YOON (2009): "Parametric links for binary choice models: A Fisherian–Bayesian colloquy," *Journal of Econometrics*, 152, 120–130.
- KOH, K., S.-J. KIM, AND S. BOYD (2007): "An Interior-Point Method for Large-Scale ℓ_1 -Regularized Logistic Regression," *J. of Machine Learning Research*, 8, 1519–1555.
- KOOPERBERG, C., AND C. J. STONE (1991): "A Study of Logspline Density Estimation," *Computational Statistics and Data Analysis*, 12, 327–347.
- LAIRD, N. (1978): "Nonparametric Maximum Likelihood Estimation of a Mixing Distribution," *Journal of the American Statistical Association*, 73, 805–811.
- LAVINE, M. (1995): "On an approximate likelihood for quantiles," *Biometrika*, 82(1), 220–222.
- LEAMER, E. E. (1985): "Vector autoregressions for causal inference?," *Carnegie-Rochester Conference Series on Public Policy*, 22, 255–304.
- LINDSAY, B. (1983): "The Geometry of Mixture Likelihoods: A General Theory," *The Annals of Statistics*, 11, 86–94.
- MA, L., AND R. KOENKER (2006): "Quantile regression methods for recursive structural equation models," *Journal of Econometrics*, 134, 471 – 506.
- MARSCHAK, J. (1953): "Economic measurements for policy and prediction," in *Studies in Econometric Method*, ed. by W. Hood, and T. Koopmans, pp. 1–26. Yale.
- MARTIN, R., AND S. G. WALKER (2013): "Asymptotically Minimax Empirical Bayes Estimation of a Sparse Normal Mean," *arXiv:1304.7366*.
- MASAROTTO, G., AND C. VARIN (2012): "The ranking lasso and its application to sport tournaments," *The Annals of Applied Statistics*, 6, 1949 – 1970.
- MATA, J., AND J. A. F. MACHADO (2005): "Counterfactual decomposition of changes in wage distributions using quantile regression," *Journal of Applied Econometrics*, 20, 445–465.

- MCCULLAGH, P. (2005): "Exchangeability and Regression Models," in *Celebrating Statistics: Papers in Honour of Sir David Cox on his 80th Birthday*, ed. by A. Davison, Y. Dodge, and N. Wermuth. Oxford University Press.
- MCCULLAGH, P., AND J. NELDER (1989): *Generalized Linear Models*. Chapman and Hall, London, second edn.
- MEHROTRA, S. (1992): "On the implementation of a primal-dual interior point method," *SIAM J. of Optimization*, 2, 575–601.
- NEELON, B., AND D. DUNSON (2004): "Bayesian isotonic regression and trend analysis," *Biometrics*, 60, 398–406.
- PARIKH, N., AND S. BOYD (2013): "Proximal Algorithms," *Foundations and Trends in Optimization*, 1, 123–221.
- PARZEN, M. I., L. WEI, AND Z. YING (1994): "A resampling method based on pivotal estimating functions," *Biometrika*, 81, 341–350.
- PERACCHI, F. (2002): "On estimating conditional quantiles and distribution functions," *Computational Statistics & Data Analysis*, 38, 433–447.
- PFANZAGL, J. (1988): "Consistency of maximum likelihood estimators for certain nonparametric families, in particular: mixtures," *Journal of Statistical Planning and Inference*, 19, 137–158.
- PICKANDS, J. (1975): "Statistical inference using extreme order statistics," *The Annals of Statistics*, 3, 119–131.
- POLSON, N. G., AND J. SCOTT (2016): "Mixtures, envelopes and hierarchical duality," *J. of the Royal Statistical Society (B)*, 78, 701–727.
- PORTNOY, S. (1991): "Asymptotic behavior of the number of regression quantile breakpoints," *SIAM J. Scientific and Statistical Computing*, 12, 867–883.
- (1997): "On computation of regression quantiles: Making the Laplacian Tortoise faster," in *L₁-statistical procedures and related topics*, ed. by Y. Dodge, pp. 187–200. Institute of Mathematical Statistics, Hayward, CA.
- (2014): "The jackknife's edge: Inference for censored regression quantiles," *Computational Statistics & Data Analysis*, 72, 273–281.
- PORTNOY, S., AND R. KOENKER (1997): "The Gaussian hare and the Laplacian tortoise: computability of squared-error versus absolute-error estimators," *Statistical Science*, 12, 279–300.
- POWELL, D. (2020): "Quantile Treatment Effects in the Presence of Covariates," *Review of Economic and Statistics*, 102, 994–1005.
- PRÉKOPA, A. (1973): "On Logarithmic Concave Measures and Functions," *Acta Sci. Math. (Szeged)*, 32, 335–343.
- R CORE TEAM (2013): *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, <http://www.R-project.org>.
- (2020): *Writing R Extensions*, R Foundation for Statistical Computing, Vienna, Austria, <https://www.R-project.org/>.
- REICH, B., M. FUENTES, AND D. DUNSON (2009): "Bayesian Spatial Quantile Regression," <http://www4.stat.ncsu.edu/~reich/SpatialQuantile.pdf>, forthcoming in JASA.
- RÉNYI, A. (1961): "On Measures of Information and Entropy," in *Proceedings of the 4th Berkeley Symposium on Mathematics, Statistics and Probability*, pp. 547–561. University of California Press.
- ROBBINS, H. (1950): "A Generalization of the Method of Maximum Likelihood; Estimating a Mixing Distribution (Preliminary Report)," *The Annals of Mathematical Statistics*, 21, 314–315.
- ROBBINS, H., AND S. MONRO (1951): "A Stochastic Approximation Method," *Annals of Mathematical Statistics*, 22, 400–407.
- ROEDER, K. (1990): "Density Estimation With Confidence Sets Exemplified by Superclusters and Voids in the Galaxies," *Journal of the American Statistical Association*, 85, 617–624.

- (1992): “Semiparametric Estimation of Normal Mixture Densities,” *The Annals of Statistics*, 20, 929–943.
- ROMANO, Y., E. PATTERSON, AND E. J. CANDÈS (2019): “Conformalized Quantile Regression,” 33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.
- ROSENBLATT, M. (1952): “Remarks on a Multivariate Transformation,” *Annals of Mathematical Statistics*, 23, 470–472.
- ROTHER, C., AND D. WIED (2013): “Misspecification testing in a class of conditional distribution models,” *J. Am. Stat. Assoc.*, 108, 314–324.
- ROTNITZKY, A., AND S. VANSTEELANDT (2014): “Double-robust methods,” in *Handbook of missing data methodology*, pp. 185–212. CRC Press.
- ROUSSEEUW, P. J., AND G. W. BASSETT (1990): “The Remedial: A Robust Averaging Method for Large Data Sets,” *Journal of the American Statistical Association*, 85, 97–104.
- RUFIBACH, K. (2007): “Computing Maximum Likelihood Estimators of a Log-Concave Density Function,” *J. Statistical Computation and Simulation*, 77, 561–574.
- RUSSELL, B. (1912): “On the Notion of Cause,” *Proceedings of the Aristotelian Society*, 13, 1–26.
- SEGAL, E. (1970): *Love Story*. HarperTorch.
- SEREGIN, A., AND J. A. WELLNER (2010): “Nonparametric estimation of multivariate convex-transformed densities,” *Annals of Statistics*, 38, 3751–3781.
- SILVERMAN, B. (1981): “Using kernel density estimates to investigate multimodality,” *Journal of the Royal Statistical Society (B)*, 43, 97–99.
- (1983): “Some properties of a test for multimodality based on kernel density estimates,” in *Probability, Statistics and Analysis*, ed. by J. Kingman, and G. Reuter, pp. 248–259. Cambridge University Press.
- SILVERMAN, B. W. (1982): “On the estimation of a probability density function by the maximum penalized likelihood method,” *Ann. Statist.*, 10, 795–810.
- (1986): *Density estimation for statistics and data analysis*. Chapman & Hall.
- SOLOFF, J. A., A. GUNTUBOYINA, AND B. SEN (2025): “Multivariate, Heteroscedastic Empirical Bayes via Nonparametric Maximum Likelihood,” .
- SOSA-ESCUDEO, W. (2020): *Borges, big data y yo*. Siglo XXI Editores.
- SPIELBERG, S. (2002): “Minority Report,” 20th Century Fox.
- STEFANSKI, L. A., AND R. J. CARROLL (1990): “Deconvolving kernel density estimators,” *Statistics*, 21, 169–184.
- STIGLER, S. (1990): *History of Statistics: The Measurement of Uncertainty Before 1900*. Harvard U Press.
- STONE, C. J. (1990): “Large-Sample Inference for Log-Spline Models,” *The Annals of Statistics*, 18, 717–741.
- TIBSHIRANI, R. J. (2008): “Fast Computation of the Median by Successive Binning,” available from: [urlhttps://www.stat.cmu.edu/~ryantibs/papers/median.pdf](https://www.stat.cmu.edu/~ryantibs/papers/median.pdf).
- TIERNEY, L. (1983): “A Space-Efficient Recursive Procedure for Estimating a Quantile of an Unknown Distribution,” *SIAM Journal on Scientific and Statistical Computing*, 4, 706–711.
- TOKDAR, S., AND J. KADANE (2010): “Simultaneous Linear Quantile Regression: A Semiparametric Bayesian Approach,” preprint.
- TOULIS, P., T. HOREL, AND E. M. AIROLDI (2020): “The proximal Robbins–Monro method,” *Journal of the Royal Statistical Society: Series B*.
- TURNER, H., AND D. FIRTH (2012): “Bradley-Terry Models in R: The BradleyTerry2 Package,” *Journal of Statistical Software*, 48, 1–21.
- VAN DER LAAN, M. J., AND S. ROSE (2011): *Targeted learning: causal inference for observational and experimental data*. Springer.
- VAN DER LAAN, M. J., AND D. RUBIN (2006): “Targeted maximum likelihood learning,” *The International Journal of Biostatistics*, 2, 1–40.

- VAN DER VAART, A. (1998): *Asymptotic Statistics*. Cambridge U. Press.
- VITÁNYI, P. M. B. (2011): “Remembering Kolmogorov,” *Metascience*, 20, 509–511.
- WAGNER, H. (1969): *Principles of Operations Research*. Prentice-Hall.
- WALTHER, G. (2009): “Inference and modeling with log-concave distributions,” *Statistical Science*, 24, 319–327.
- WICKSELL, S. D. (1925): “The corpuscle problem: A mathematical study of a biometric problem,” *Biometrika*, 17, 84–99.
- WOLD, H. O. A., AND R. H. STROTZ (1960): “Recursive vs. Nonrecursive Systems: An Attempt at Synthesis (Part I of a Triptych on Causal Chain Systems),” *Econometrica*, 28, 417–427.
- WOLPERT, D. H. (1992): “Stacked generalization,” *Neural Networks*, 5, 241–259.
- WOOD, S. (2017): *Generalized Additive Models: An Introduction with R*. Chapman and Hall/CRC, 2 edn.
- YAO, Y., A. VEHTARI, D. SIMPSON, AND A. GELMAN (2018): “Using Stacking to Average Bayesian Predictive Distributions (with Discussion),” *Bayesian Analysis*, 13, 917–1007.
- ZEILEIS, A., AND Y. CROISSANT (2010): “Extended Model Formulas in R: Multiple Parts and Multiple Responses,” *Journal of Statistical Software*, 34, 1–13.
- ZERMELO, E. (1929): “Die Berechnung der Turnier-Ergebnisse als ein Maximum-problem der Wahrscheinlichkeitsrechnung,” *Mathematische Zeitschrift*, 29, 436–460.
- ZHANG, L., E. D. CASTILLO, A. BERGLUND, AND M. TINGLEY (2020): “Computing confidence intervals from massive data via penalized quantile smoothing splines,” *Computational Statistics and Data Analysis*, 144, 106885.