Statistical implications of (some) computational approximations

Daniel J. McDonald
Indiana University, Bloomington
mypage.iu.edu/~dajmcdon

24 January 2018

OBLIGATORY "DATA IS BIG" SLIDE

Modern statistical applications — genomics, neural image analysis, text analysis, weather prediction — have large numbers of covariates p

Also frequently have lots of observations n.

Need algorithms which can handle these kinds of data sets. With good statistical properties

LESSON OF THE TALK

Computational choices impact statistical performance.

These choices can take many forms:

- choosing tuning parameters
- different optimization algorthms return different solutions
- how long do we run our MCMC (and which kind do we use)

Theory often neglects these choices:

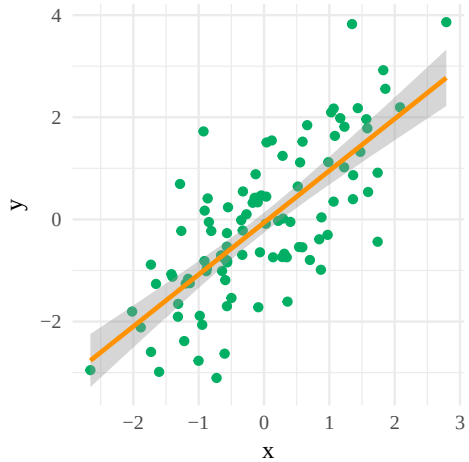
- Lasso works with oracle tuning parameter
- We have the posterior if our MCMC runs forever
- EM gives us a global solution
- Your solver finds the optimum (looking at you `optim`)

In this talk

Many statistical methods use (perhaps implicitly) a singular value decomposition (SVD) to solve an optimization problem:

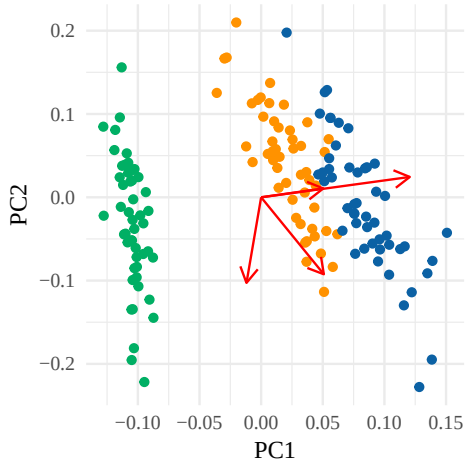
- Penalized Least Squares:

$$\min_{\beta} ||Y - X\beta||_2^2 + \lambda \text{Pen}(\beta)$$



- PCA:

$$\max_{V^\top V = I_d} V^\top X^\top X V$$



SVD == SLOW

The SVD is computationally expensive.

For a generic $n \times p$ matrix, the SVD requires $O\left(\min\left\{np^2, n^2p\right\}\right)$ and storage of the entire matrix in fast memory.

I want to understand the statistical properties of some approximations which speed up computation and save storage.

OUTLINE

1. Introduction
2. (Detour) An illustrative/motivating application
3. Generic problem statement and related work
4. New methods
5. Experimental results
6. Tuning parameters
7. Theory
8. Ongoing/related/future work

(Detour) Estimating the trend in cloud-top temperature volatility

CLIMATE CHANGE

The scientific consensus is that

1. world-wide climate is changing and
2. this change is mostly driven by human behavior.

"Global warming" is disfavored relative to "climate change" because it's the distribution of temperature (and precipitation) that is changing

There are many reasons that increasing mean temperature may understate the costs:

1. More frequent extremes have severe effects
2. Local discrepancies lead to more storms
3. Temporal dependencies mean persistence

USING WEATHER SATELLITES

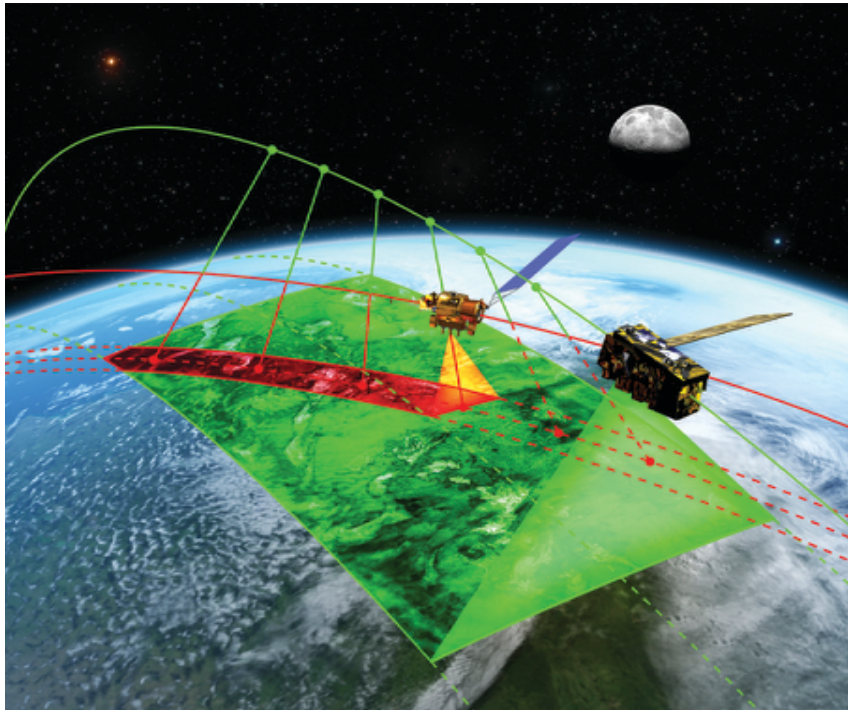
Drivers of weather variation:

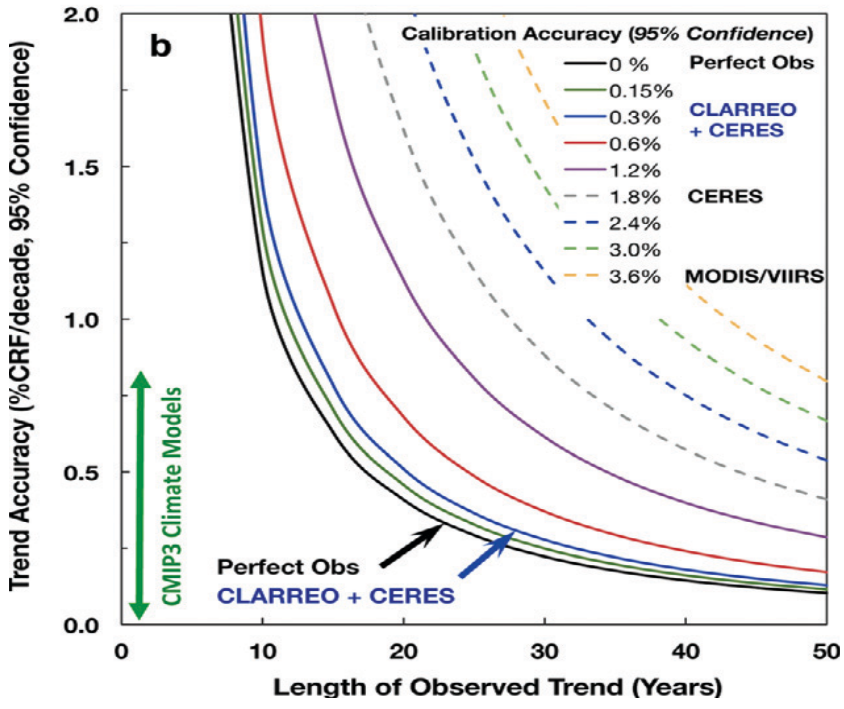
1. Ocean currents
2. Jet stream
3. Annular modes + El Niño/La Niña
4. Cloudiness

CLARREO satellite set to monitor cloud top temperature as it relates to climate.

- Has yet to launch, no sooner than 2022
- Defunded in most recent federal budget

CLARREO vs MetOp/Modis





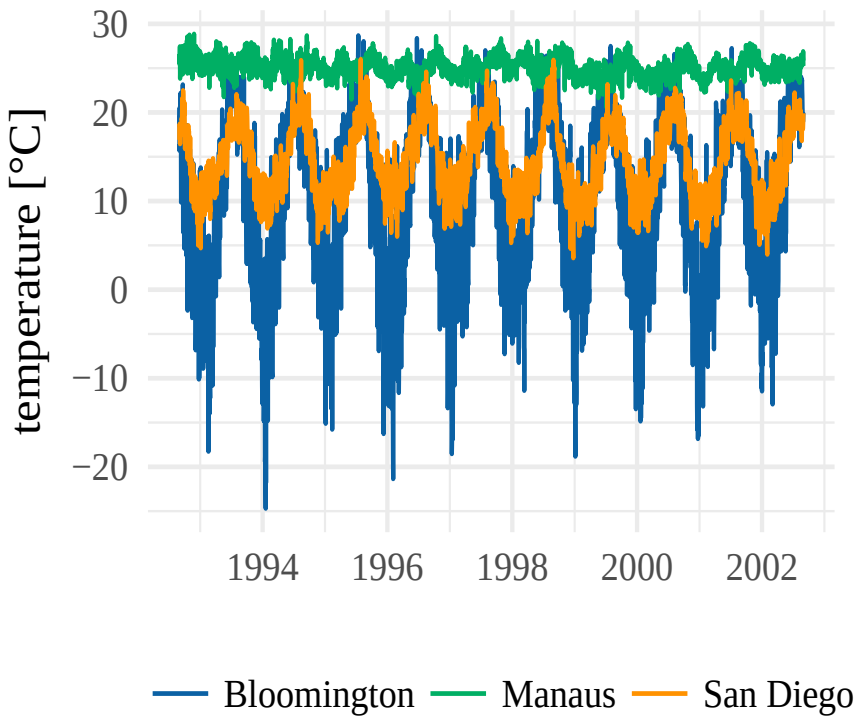
- Weather satellites aren't made for this.
- Compensate: Possibly more information in higher moments than in average.

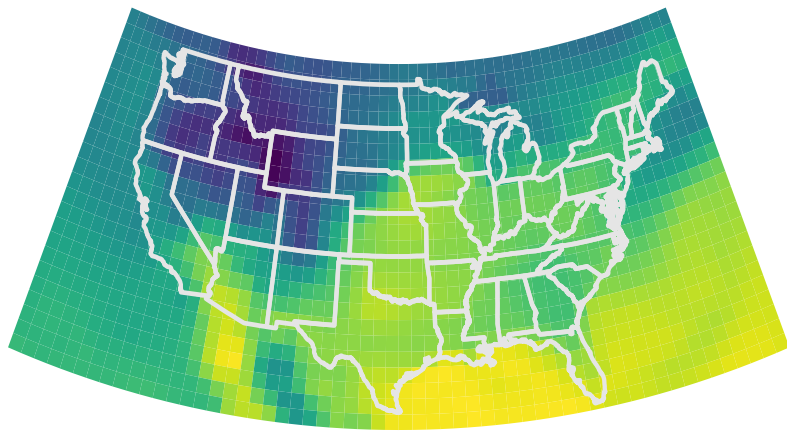
Source: Wielicki, et al. (2013).

SATELLITE DATA

Once collaborators do lots of processing (all of which has statistical implications)...

- 52,000 time series
- daily records over about 40 years
- "trends" are local, nonlinear, not sinusoidal





TRENDS IN VARIANCE

Let y_{ts} be the observed temperature at time t and location s.

"Suppose": $y_{ts} \sim \mathrm{N}(0, \exp(h_{ts}))$

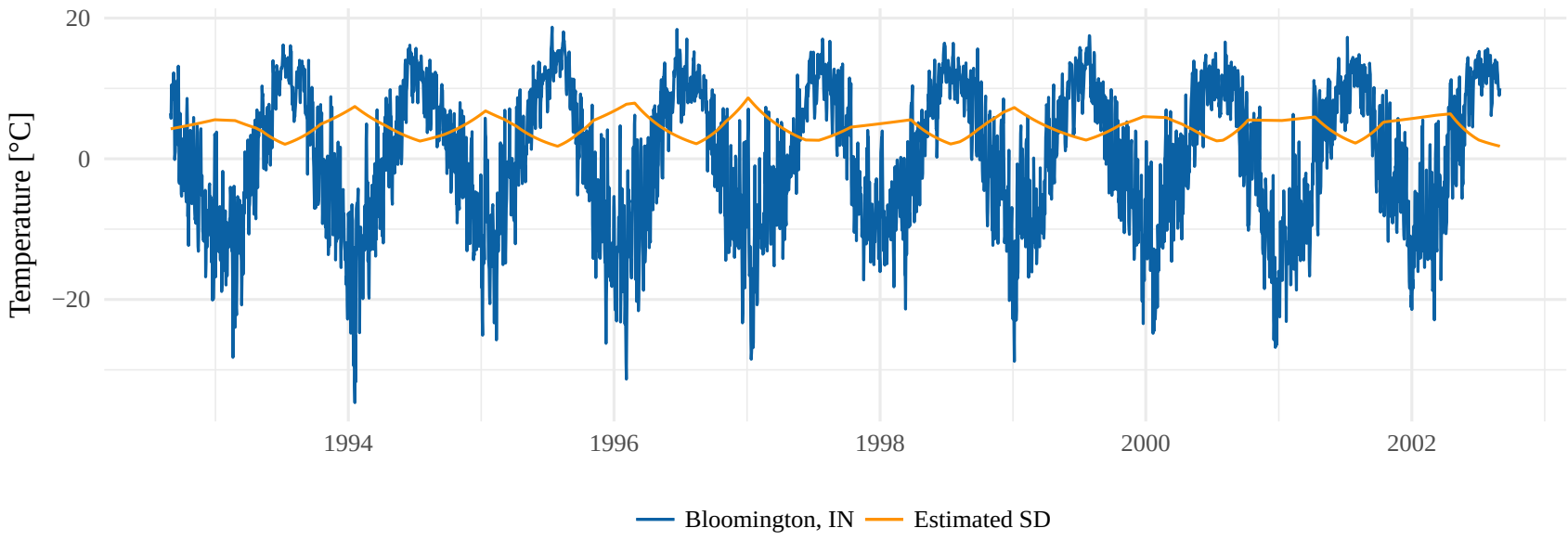
We want to estimate h, but it should be "smooth" relative to space and time.

Use a matrix D to encode this smoothness.

Optimization problem

$$\min_h \sum h_{st} + y_{st}^2 e^{-h_{st}} + \lambda ||Dh||_1$$

Previous work for optimization problems like this suggests using Primal Dual Interior Point method.[1]



[1] see Tibshirani 2014 or K-K-Boyd-G 2009

GENERIC PDIP

1. Start with a guess $h^{(1)}$
2. Solve a linear system $[Au = v]$
3. Calculate a step size
4. Iterate 2 & 3 until convergence

The matrix A changes each iteration, is essentially square and dense, and roughly $10^9 \times 10^9$.

This isn't going to work.

Detailed PDIP

Primal	Dual
$\min_h \quad f(h) + \lambda \left\|\left\|Dh\right\|\right\|_1$	$\min_v \quad f^*(-D^\top v)$ s.t. $\quad \left\|\left\|v\right\|\right\|_\infty \leq \lambda$

- $f(h) := \sum h_{st} + y_{st}^2 e^{-h_{st}}$
- $f^*(u) := \sum (u_{st} - 1) \log \frac{y_{st}^2}{1 - u_{st}} + u_{st} - 1$

KKT conditions $(w > 0) \Longrightarrow$

$$r_w(v, \mu_1, \mu_2) := \begin{bmatrix} \nabla f^*(-D^\top v) + D(v - \lambda \mathbf{1})^\top \mu_1 - D(v + \lambda \mathbf{1})^\top \mu_2 \\ -\mu_1 (v - \lambda \mathbf{1}) + \mu_2 (v + \lambda \mathbf{1}) - w^{-1} \mathbf{1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

- As $w \to \infty$, this converges to the optimum.
- But this is a nonlinear system, can't solve.
- Use Newton steps, which give the $[Au = v]$ thing
- A is the Jacobian of r_w.

HINTS AND CAVEATS

- More carefully exploit the structure of D
- We could use consensus ADMM to parallelize
- We could get creative, farm out computing to Amazon
- Need to repeat for many tuning parameters
- Could have used ℓ_2
- Really big Extended Kalman Filter
- Approximations...

Generic problem statement

Suppose we have a matrix $\mathbb{X} \in \mathbb{R}^{n \times p}$ and vector $Y \in \mathbb{R}^n$

LEAST SQUARES:

$$\widehat{\beta} = \underset{\beta}{\operatorname{argmin}} \, ||\mathbb{X}\beta - Y||_2^2$$

ℓ_2 REGULARIZATION:

$$\widehat{\beta}(\lambda) = \underset{\beta}{\operatorname{argmin}} \, ||\mathbb{X}\beta - Y||_2^2 + \lambda \, ||\beta||_2^2$$

CORE TECHNIQUES

If $\mathbb{X}$ fits into RAM, there exist excellent algorithms in LAPACK that are

- Double precision
- Very stable
- $O(np^2)$ with small constants when $n \gg p$.
- require extensive random access to matrix

There is a lot of interest in finding and analyzing techniques that extend these approaches to large(r) problems

OUT-OF-CORE TECHNIQUES

Many techniques focus on randomized compression

This is sometimes known as sketching or preconditioning

- Rokhlin, Tygert, (2008) “A fast randomized algorithm for overdetermined linear least-squares regression.”
- Drineas, Mahoney, et al., (2011) “Faster least squares approximation.”
- Woodruff, (2014) “Sketching as a Tool for Numerical Linear Algebra.”
- Wang, Lee, Mahdavi, Kolar, Srebro, (2016) "Sketching meets random projection in the dual."
- Ma, Mahoney, and Yu, (2015), "A statistical perspective on algorithmic leveraging."
- Pilanci and Wainwright, (2015-2016). Multiple papers.
- Others.

COMPRESSION

Basic Idea:

- Choose some matrix $Q \in \mathbb{R}^{q \times n}$.
- Under many conditions, sufficient to choose $q = \Omega(p)$.
- Use $Q\mathbb{X}$ (and) QY instead in the optimization.
- $O(np^2) \longrightarrow O(p^3)$.

Finding $Q\mathbb{X}$ for arbitrary Q and $\mathbb{X}$ takes $O(qnp)$ computations.

So we're back to $O(np^2)$.

To get this approach to work, we need some structure on Q

THE Q MATRIX

- Gaussian:
  Well behaved distribution and eas(ier) theory. Dense matrix
- Fast Johnson-Lindenstrauss Methods
- Randomized Hadamard (or Fourier) transformation:
  Allows for $O(np\log(p))$ computations.
- $Q = \pi\tau$ for π a permutation of I and $\tau = [I_q\ 0]$:
  $Q\mathbb{X}$ means "read q (random) rows"
- Sparse Bernoulli:

$$Q_{ij} \overset{i.i.d.}{\sim} \begin{cases} 1 & \text{with probability } 1/(2s) \\ 0 & \text{with probability } 1 - 1/s \\ -1 & \text{with probability } 1/(2s) \end{cases}$$

This means $Q\mathbb{X}$ takes $O\left(\frac{qnp}{s}\right)$ "computations" on average.

TYPICAL RESULTS

The general philosophy: Find an approximation algorithm that is as close as possible to the solution of the original problem.

For OLS, typical results would be to produce an $\tilde{\beta}$ such that

$$\left|\left|\mathbb{X}\tilde{\beta} - Y\right|\right|_2^2 \leq (1+\epsilon)\left|\left|\mathbb{X}\widehat{\beta} - Y\right|\right|_2^2,$$
$$\left|\left|\mathbb{X}(\tilde{\beta} - \widehat{\beta})\right|\right|_2^2 \leq \epsilon\left|\left|\mathbb{X}\widehat{\beta}\right|\right|_2^2,$$
$$\left|\left|\tilde{\beta} - \widehat{\beta}\right|\right|_2^2 \leq \epsilon\left|\left|\widehat{\beta}\right|\right|_2^2,$$

Here, $\tilde{\beta}$ should be 'easier' to compute than

$$\widehat{\beta} = \underset{\beta}{\operatorname{argmin}} \, ||\mathbb{X}\beta - Y||_2^2$$

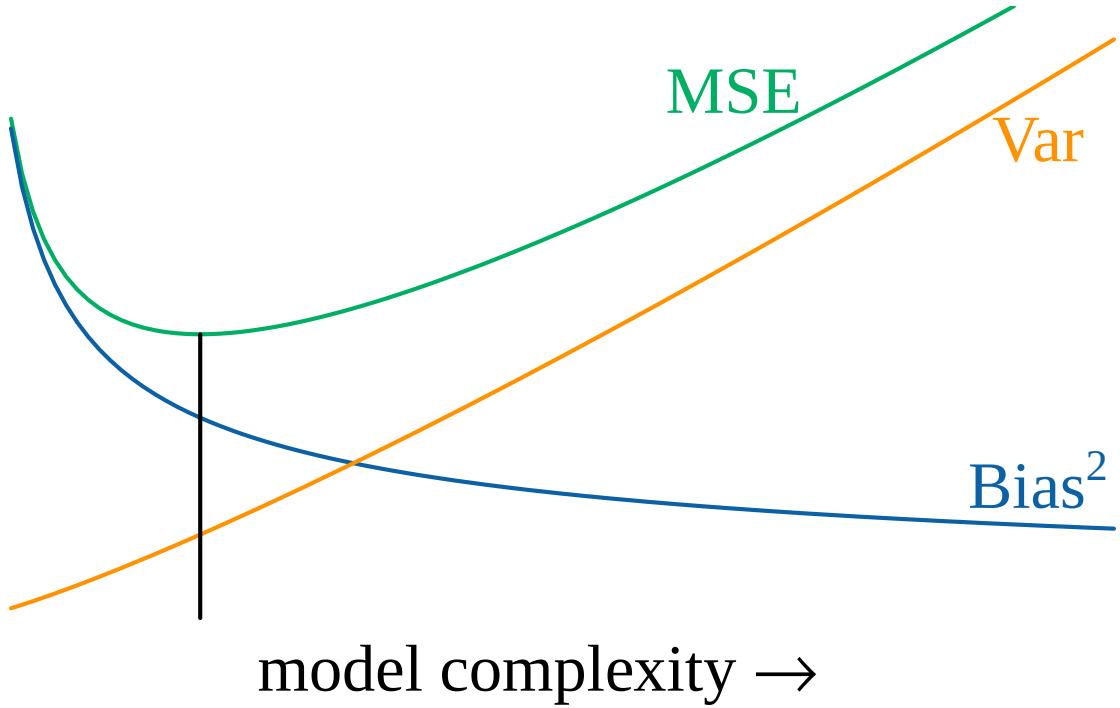
STATISTICAL ANALYSIS

For an approximation $\tilde{\theta}$ of $\widehat{\theta}$,

$$\begin{aligned}\mathbb{E}\left\|\theta-\tilde{\theta}\right\|_2^2 &= \mathbb{E}\left\|\tilde{\theta}-\widehat{\theta}+\widehat{\theta}-\mathbb{E}\widehat{\theta}+\mathbb{E}\widehat{\theta}-\theta\right\|_2^2 \\ &\leq \mathbb{E}\text{Approx. error}^2+\text{Variance}+\text{Bias}^2\end{aligned}$$

- Previous analyses focus only on the approximation error (with expectation over the algorithm, not the data).
- Implications:
  1. Don't care about bias
  2. Variance $\nearrow$

BIAS-VARIANCE TRADEOFF



- We examine $\mathbb{E}\left|\left|\theta - \tilde{\theta}\right|\right|_2^2$ where the expectation is over everything random.
- Only other similar is Ma, Mahoney, and Yu (JMLR, 2015).

CONTRIBUTIONS (TODAY)

Solving penalized least-squares:

1. Introduce 2 new versions of compression.
2. Examine their statistical performance (empirics and theory).
3. Show how to choose tuning parameters (without extra computation).

Methodology

Family of 4

1. Full compression:

$$\begin{aligned}\widehat{\beta}_{FC} &= \underset{\beta}{\operatorname{argmin}} \, ||Q(\mathbb{X}\beta - Y)||_2^2 \\ &= \underset{\beta}{\operatorname{argmin}} \, ||Q\mathbb{X}\beta||_2^2 - 2Y^\top Q^\top Q\mathbb{X}\beta \\ &= (\mathbb{X}^\top Q^\top Q\mathbb{X})^{-1}\mathbb{X}^\top Q^\top QY\end{aligned}$$

2. Partial compression:[1]

$$\begin{aligned}\widehat{\beta}_{PC} &= \underset{\beta}{\operatorname{argmin}} \, ||Q\mathbb{X}\beta||_2^2 - 2Y^\top \mathbb{X}\beta \\ &= (\mathbb{X}^\top Q^\top Q\mathbb{X})^{-1}\mathbb{X}^\top Y\end{aligned}$$

[1] Also called "Hessian Sketching".

We also combine these

Write:

$$B = [\, \widehat{\beta}_{FC}\ \widehat{\beta}_{PC}\,]$$

$$W = \mathbb{X}B$$

3. Linear combination compression:

$$\widehat{\alpha}_{lin} = \underset{\alpha}{\operatorname{argmin}} \, ||W\alpha - Y||_2^2$$

$$\widehat{\beta}_{lin} = B\widehat{\alpha}_{lin}$$

4. Convex combination compression:

$$\widehat{\alpha}_{con} = \underset{\substack{0 \leq \alpha \\ \sum \alpha = 1}}{\operatorname{argmin}} \, ||W\alpha - Y||_2^2$$

$$\widehat{\beta}_{con} = B\widehat{\alpha}_{con}$$

- These are simple to calculate given FC and PC.

WHY THESE?

- Turns out that FC is (approximately) unbiased, and therefore worse than OLS (has high variance)
- On the other hand, PC is biased and empirics demonstrate low variance
- Combination should give better statistical properties
- We do everything with an ℓ_2 penalty

Evidence from simulations

Simulation setup

- Draw $\mathbb{X}_i \sim \text{MVN}(0, (1-\rho)I_p + \rho \mathbf{1}\mathbf{1}^\top)$
  - We use $\rho = \{0.2,\ 0.8\}$.
- Draw $\beta \sim \text{N}(0, \tau^2 I_p)$
- Draw $Y_i = \mathbb{X}_i^\top \beta + \epsilon_i$ with $\epsilon_i \sim \text{N}(0, \sigma^2)$.

BAYES ESTIMATOR

- For this model, the optimal estimator (in MSE) is

$$\widehat{\beta}_B = (\mathbb{X}^\top \mathbb{X} + \lambda_* I_p)^{-1} \mathbb{X}^\top Y$$

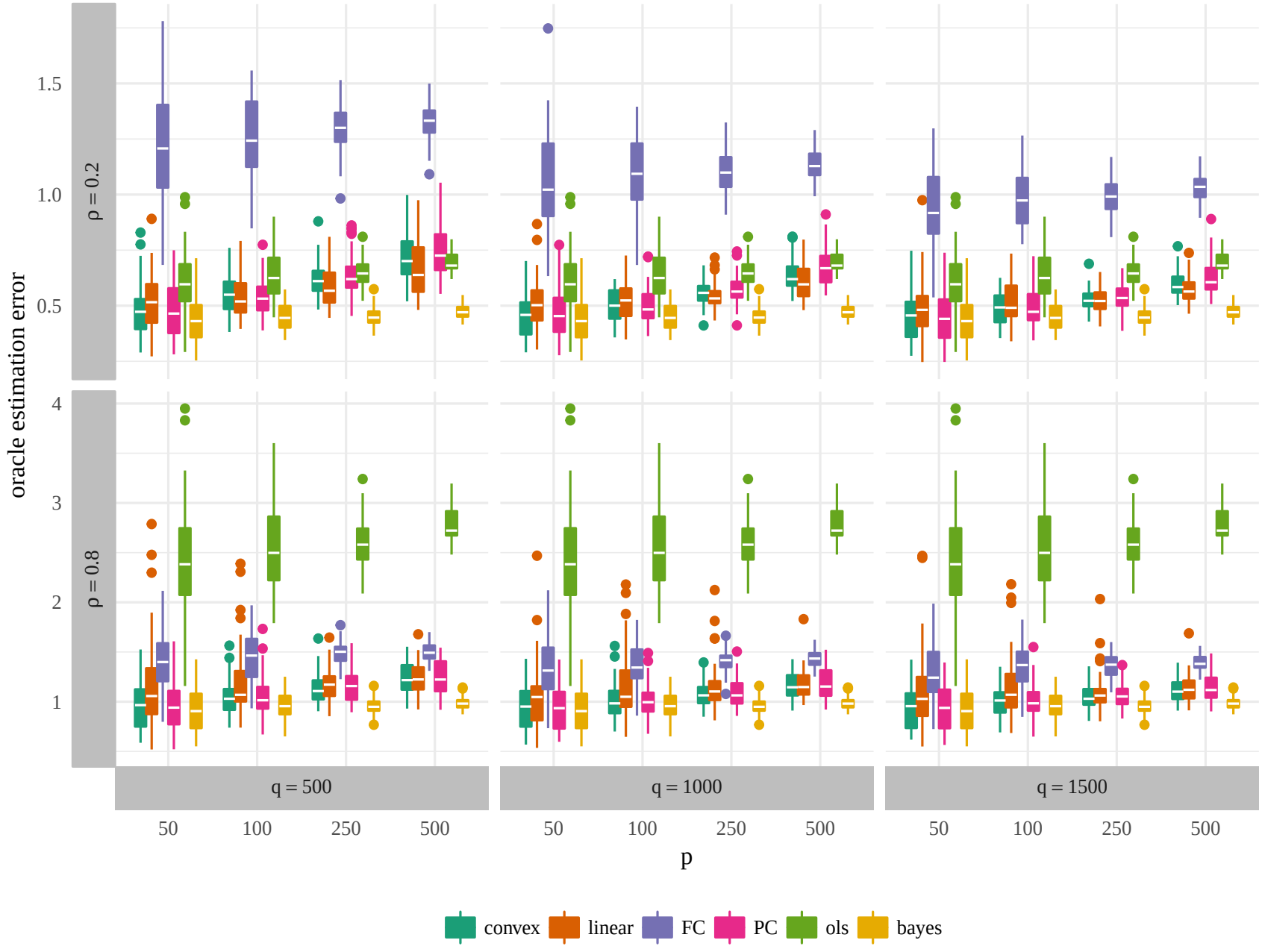
- In particular, with $\lambda_* = \frac{\sigma^2}{n\tau^2}$
- This is the posterior mode of the Bayes estimator under conjugate normal prior
- It is also the ridge regression estimator for a particular λ

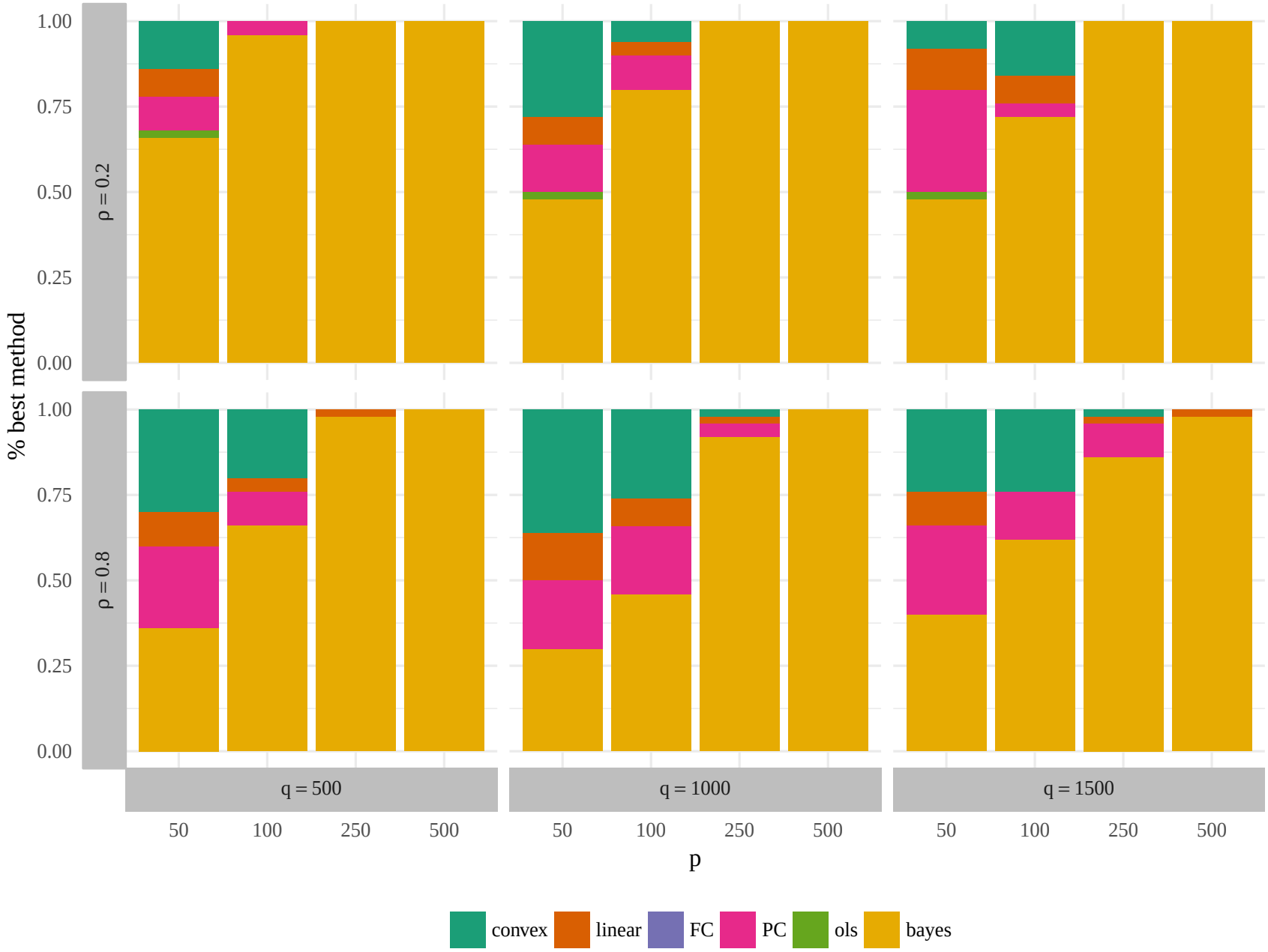
GOLDILOCKS

With $p < n$

1. If λ_* is too big, we will tend to shrink all coefficients to 0.
   - This problem is too hard.
2. If λ_* is too small, OLS will be very close to the optimal estimator.
   - This problem is too easy.
3. Need τ^2, σ^2 "just right".

- Take $\tau^2 = \pi/2$. This implies $\mathbb{E}[|\beta_i|] = 1$ (convenient)
- Take $n = 5000$. Big but computable.
- Take $\sigma = 50 \Rightarrow \log(\lambda_*) \approx -1.14$ (reasonable)
- Take $p \in \{50,\ 100,\ 250,\ 500\}$

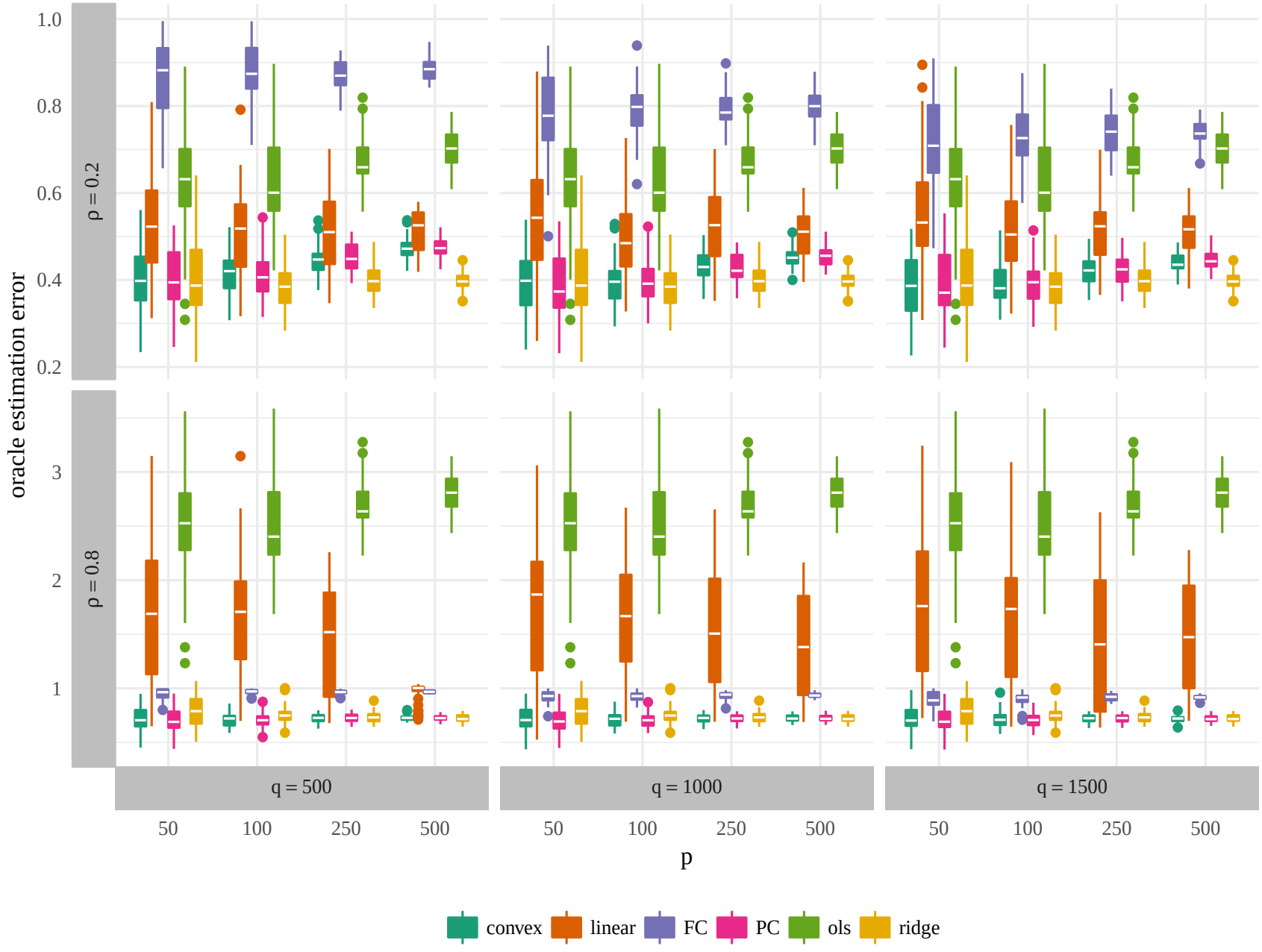

oracle estimation error
ρ = 0.2
ρ = 0.8
q = 500
q = 1000
q = 1500
1.5
1.0
0.5
4
3
2
1
50
100
250
500
p
convex
linear
FC
PC
ols
bayes

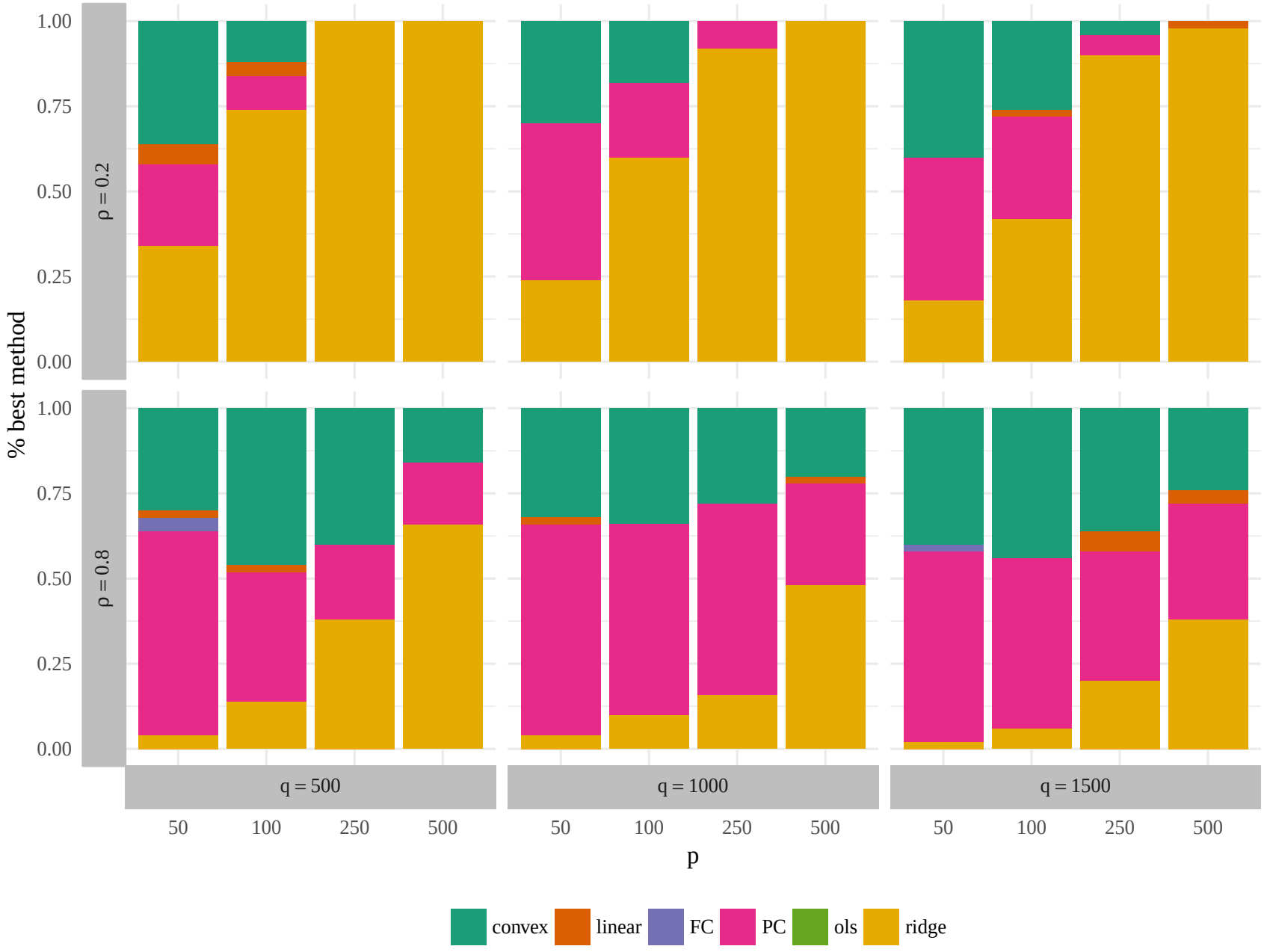

% best method
1.00
0.75
0.50
0.25
0.00
ρ = 0.2
ρ = 0.8
q = 500
q = 1000
q = 1500
50
100
250
500
p
convex
linear
FC
PC
ols
bayes

SECOND VERSE. . .

In that case, ridge was optimal.

We did it again with $\beta_i \in \{-1, 1\}$.


oracle estimation error
ρ = 0.2
ρ = 0.8
q = 500
q = 1000
q = 1500
1.0
0.8
0.6
0.4
0.2
3
2
1
50
100
250
500
p
convex
linear
FC
PC
ols
ridge


% best method
1.00
0.75
0.50
0.25
0.00
ρ = 0.2
ρ = 0.8
q = 500
q = 1000
q = 1500
50
100
250
500
p
convex
linear
FC
PC
ols
ridge

Tuning parameter selection

MINIMAL EXTRA COMPUTATIONS

- We use GCV with the degrees of freedom:

$$\text{GCV}(\lambda) = \frac{\left|\left|\mathbb{X}\widehat{\beta}(\lambda) - Y\right|\right|_2^2}{(1 - \text{df}/n)^2}$$

- df is easy for full or partial compression.
- For the other cases, an ad hoc approximation works, but has no justification.
- We derive an estimate via Stein's method.
- Easy to calculate both for a range of λ without extra computations.

STEIN'S METHOD DETAILS

- The degrees of freedom for a generic predictor f is[1]

$$\mathrm{df}(f) := \frac{1}{\sigma^2} \sum_{i=1}^{n} \mathrm{Cov}(Y_i, f_i(Y)).$$

- For normal linear model (OLS), we have

$$\mathrm{df} = \frac{1}{\sigma^2} \mathbb{E}\left[\widehat{Y}^\top (Y - \mu_Y)\right] = \frac{1}{\sigma^2} \mathbb{E}\left[Y^\top H Y\right] = p.$$

- In general, Stein's Lemma gives us the following: if $Y_i - \mathbb{E}\left[Y_i | X_i\right] \sim \mathrm{N}(0, \sigma^2)$

$$\mathrm{Cov}(Y_i, f_i(Y)) = \mathbb{E}\left[(Y_i - \mathbb{E}\left[Y_i | X_i\right]) f_i(Y)\right] = \mathbb{E}\left[\nabla f_i(Y)\right].$$

[1] See e.g. Efron

APPLYING IT

Because linear and convex combination compression have closed form expressions:

e.g. Linear combination:

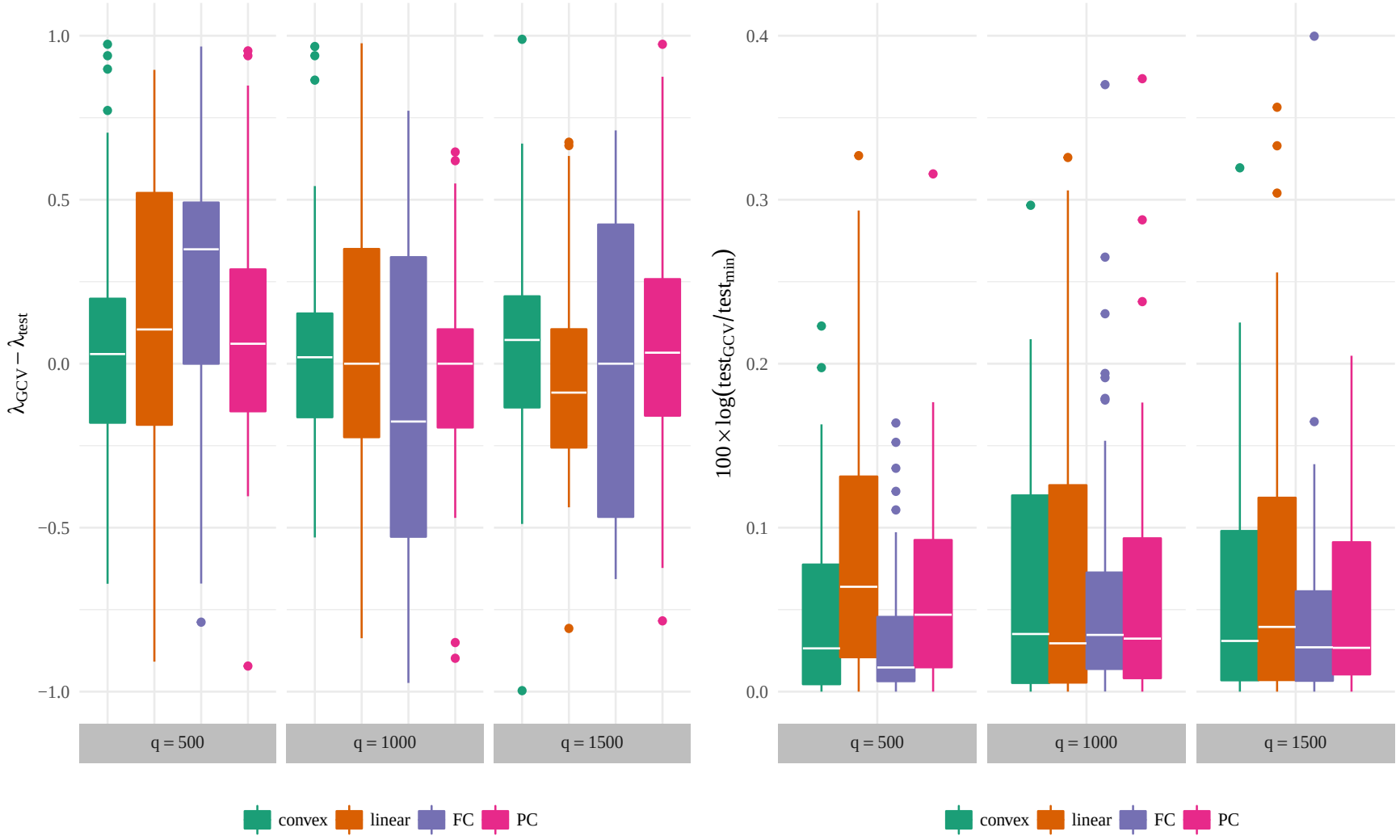
$$f(Y) = \left[\mathbb{X}B\right]\left[(B^\top \mathbb{X}^\top \mathbb{X}B)^{-1} B^\top \mathbb{X}^\top Y\right]$$

where

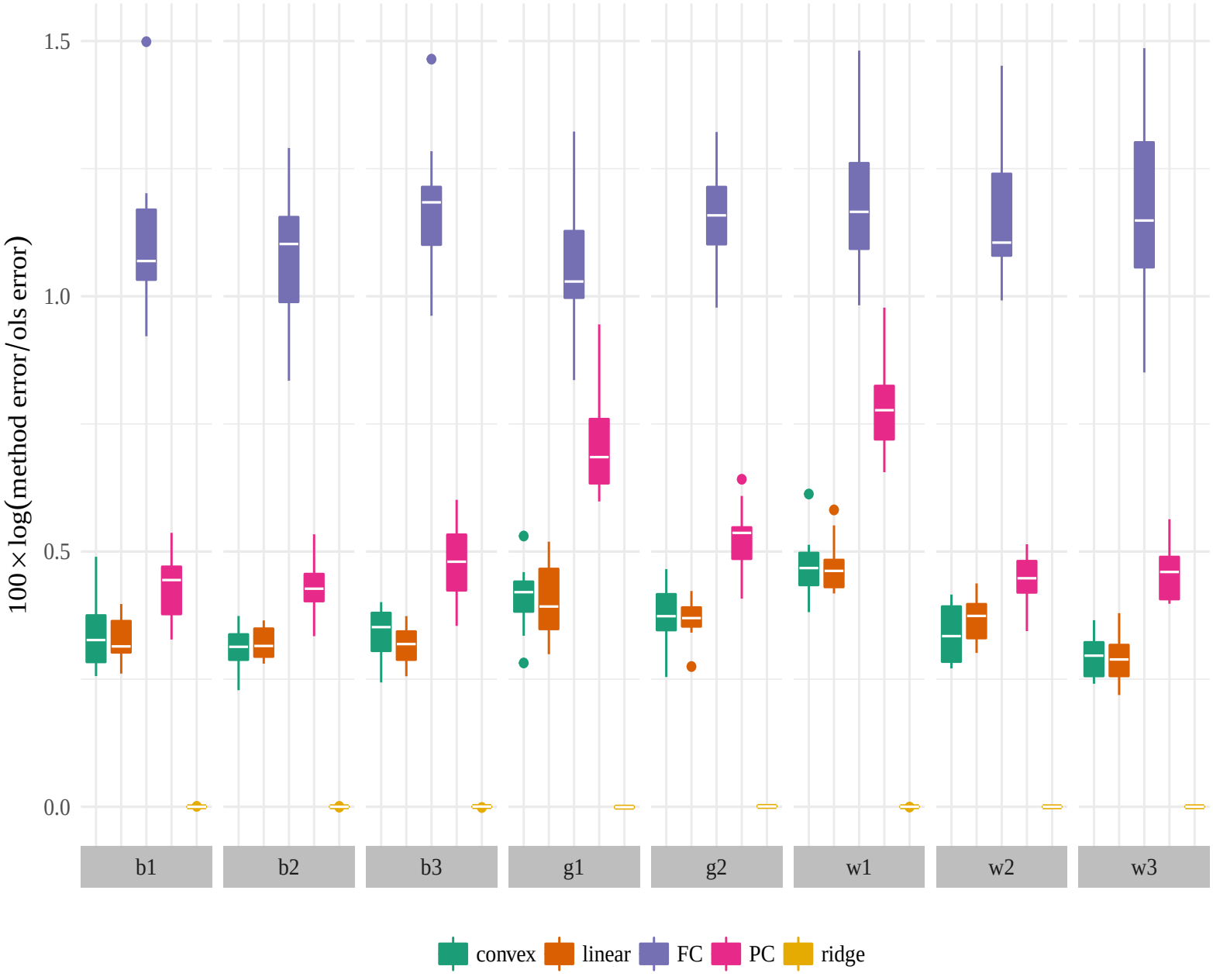
$$B = \left[(\mathbb{X}^\top Q^\top Q\mathbb{X} + \lambda I)^{-1}\mathbb{X}^\top\right]\begin{bmatrix} Q^\top QY \\ Y \end{bmatrix}.$$

Apply every calculus rule you can find to yield a nasty expression which won't fit on this slide.

This is an unbiased estimate of df.



$$n = 5000, \quad p = 50, \quad \rho = 0.8, \quad \beta \sim \mathbf{N}(0, \pi/2)$$


1.5
1.0
0.5
0.0
100 × log(method error/ols error)
b1
b2
b3
g1
g2
w1
w2
w3
convex
linear
FC
PC
ridge

TAKE-AWAY MESSAGE

- $q = 10000$ results in data reductions between 74% and 93%
- $q = 20000$ gives reductions between 48% and 84%
- ridge and ordinary least squares give equivalent test set performance (differing by less than .001%)
- This is an "easy" problem.
- Full compression is the worst.

Theoretical results (sketch)

STANDARD RIDGE RESULTS

Theorem

$$\text{bias}^2\left(\widehat{\beta}_{ridge}(\lambda) \mid \mathbb{X}\right) = \lambda^2 \beta_*^\top V (D^2 + \lambda I_p)^{-2} V^\top \beta_*.$$

$$\text{tr}\left(\mathbb{V}[\widehat{\beta}_{ridge}(\lambda) \mid \mathbb{X}]\right) = \sigma^2 \sum_{i=1}^{p} \frac{d_i^2}{(d_i^2 + \lambda)^2}.$$

WHAT'S THE TRICK?

- Similar results are hard for compressed regression.
- All the estimators depend (at least) on

$$(\mathbb{X}^\top Q^\top Q \mathbb{X} + \lambda I_p)^{-1}$$

- We derived properties of $Q^\top Q$

$$\mathbb{E}\left[\frac{s}{q} Q^\top Q\right] = I_n$$

$$\mathbb{V}\left[\text{vec}\left(\frac{s}{q} Q^\top Q\right)\right] = \frac{(s-3)_+}{q} \text{diag}(\text{vec}\,(I_n)) + \frac{1}{q} I_{n^2} + \frac{1}{q} K_{nn}$$

- So the technique is to do a Taylor expansion around $\frac{s}{q} Q^\top Q = I_n$.

MSE of Full Compression

Theorem:

$$\text{bias}^2[\widehat{\beta}_{FC} \mid \mathbb{X}] = \lambda^2 \beta_*^\top V (D^2 + \lambda I_p)^{-2} V^\top \beta_* + o_p(1)$$

$$\begin{aligned}\text{tr}(\mathbb{V}[\widehat{\beta}_{FC} \mid \mathbb{X}]) = \sigma^2 &\sum_{i=1}^{p} \frac{d_i^2}{(d_i^2 + \lambda)^2} + o_p(1) \\ &+ \frac{(s-3)_+}{q} \text{tr}\left(\text{diag}(\text{vec}\,(I_n)) M^\top M \otimes (I - H) M \beta_* \beta_*^\top M^\top (I - H)\right) \\ &+ \frac{\beta_*^\top M^\top (I - H)^2 M \beta_*}{q} \text{tr}(M M^\top) \\ &+ \frac{1}{q} \text{tr}\left((I - H) M \beta_* \beta_*^\top M^\top (I - H) M^\top M\right).\end{aligned}$$

Note: $M = (\mathbb{X}^\top \mathbb{X} + \lambda I_p)^{-1} \mathbb{X}^\top$ and $H = \mathbb{X} M$ (hat matrix for ridge regression)

Corollary:

If $\mathbb{X}^\top \mathbb{X} = nI_p$,

$$\text{MSE}(\widehat{\beta}_{ridge}) = b^2 \left(\frac{\theta}{1+\theta}\right)^2 + \frac{p\sigma^2}{n(1+\theta)^2}$$

$$\text{MSE}(\widehat{\beta}_{FC}) = b^2 \left(\frac{\theta}{1+\theta}\right)^2 + \frac{p\sigma^2}{n(1+\theta)^2} + \frac{b^2 p\theta^2 (s-2)_+}{q(1+\theta)^4} + \frac{p^2\theta^2 b^2}{q(1+\theta)^4}$$

$$\text{MSE}(\widehat{\beta}_{PC}) = b^2 \left(\frac{\theta}{1+\theta}\right)^2 + \frac{p\sigma^2}{n(1+\theta)^2} + \frac{p(s-2)_+ b^2}{q(1+\theta)^2} + \frac{pb^2}{q(1+\theta)^4}$$

Where $b^2 := ||\beta_*||_2^2$, and $\theta := \lambda/n$

The big picture

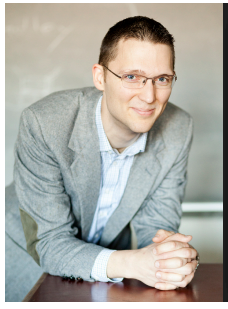
Related work on approximation

- Today: Approximation for Least Squares $(n \gg p)$
  - Homrighausen and McDonald. (2018+). Under review.
- Approximation for dimension reduction $(p \gg n)$
  - Homrighausen and McDonald. (2016). JCGS.
  - Ding and McDonald. (2017). Bioinformatics.
  - Sketching and PCA (ongoing work with Ding).
- Approximation inside of optimization
  - ADMM for large, sparse kernel PCA (with Ding)
  - Sketched PDIP (with Khodadadi)

TUNING PARAMETER SELECTION AND RISK ESTIMATION

- Today: GCV and SURE for compressed regression
- CV and ℓ_1 regularization
  - Homrighausen and McDonald. (2013). ICML.
  - Homrighausen and McDonald. (2014). Machine Learning.
  - Homrighausen and McDonald. (2017). Stat. Sinica.
- Dependence and high dimensions
  - Homrighausen and McDonald. (2018+). Under review.
  - McDonald and Shalizi. (2018+). Under review.
  - McDonald, Shalizi and Schervish. (2017). JMLR.

COLLABORATORS AND FUNDING



Institute for
New Economic Thinking