Regularization, optimization, and approximation

The benefits of a convex combination

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

6 February 2019

Obligatory “data are 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)

Statistical 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
- Algorithmic guarantees hold uniformly over ALL datasets

Today's emphasis

Many statistical methods or optimization algorithms use a singular value decomposition (SVD):

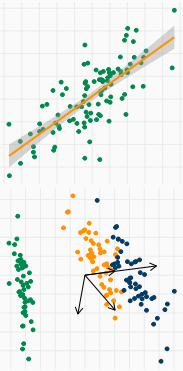
$$X = UDV^\top : U^\top U = I,\ V^\top V = I,\ D \text{ diagonal.}$$

- Penalized Least Squares:

$$\min_\beta \frac{1}{2n}\|Y - X\beta\|_2^2 + \lambda \text{Pen}(\beta)$$

- PCA:

$$\max_{V^\top V = I_d} \text{tr}\left(V^\top X^\top X V\right)$$



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 that speed up computation and save storage.

1. Better algorithmic choices allow us to solve a statistical problem.
2. Algorithmic advances suggest new statistical techniques.
3. Statistical models suggest better algorithms.

Outline

1. Introduction
2. Climate science and choosing better algorithms
3. Approximations for many observations and robust statistics
4. Many measurements and better approximate algorithms
5. Ongoing/related/future work

Estimating the trend in cloud-top temperature volatility

Climate change

The scientific consensus is that

1. World-wide climate is changing.
2. This change is mostly driven by human behavior.

~~Global warming~~ $\longrightarrow$ climate change: the distribution of temperature (and precipitation) is changing

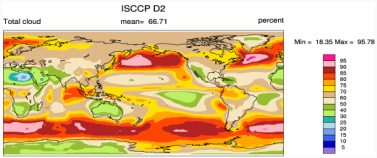
Increasing mean temperature understates the costs:

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

Using weather satellites

Drivers of climate variation:

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

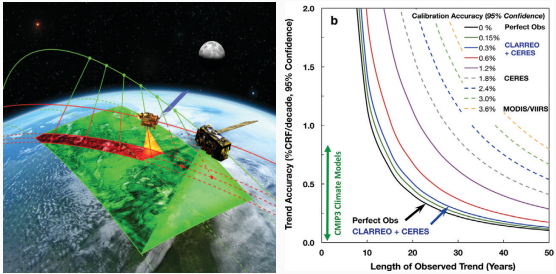


CLARREO satellite: monitor cloud top temperature as it relates to climate.

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

Source: NCAR CCSM3 Diagnostic Plots.

CLARREO vs MetOp/Modis



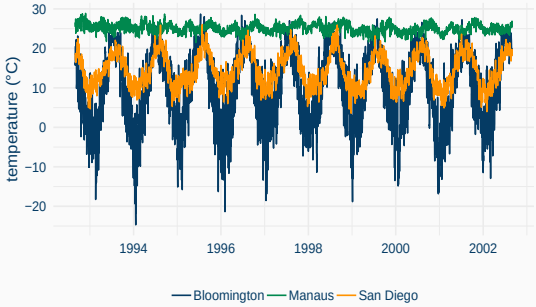
- Weather satellites aren't made for this.
- More information in higher moments than in average?

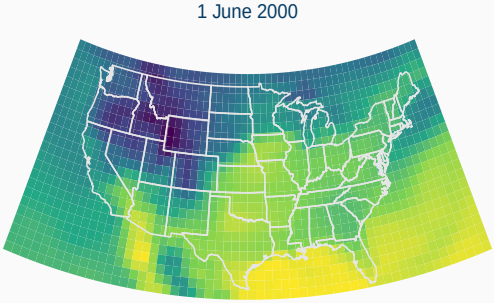
Source: Wielicki, et al. (2013).

Satellite data

Once collaborators do lots of processing…

- 52,000 time series
- daily records over ∼ 40 years
- “trends” are local, nonlinear, not sinusoidal





Trends in variance

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

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

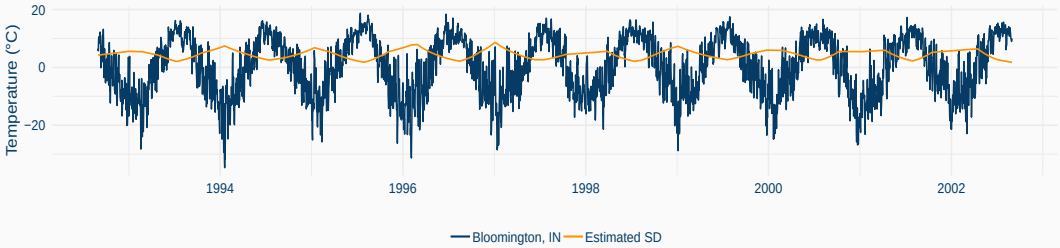
Estimate h, but it should be “smooth” relative to space and time.

Use a matrix $D +$ penalty to encode this smoothness.

Optimization problem

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

Standard optimizer: Primal Dual Interior Point method.



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, dense, and roughly $10^9 \times 10^9$.

This isn't going to work.

Detailed PDIP

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

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

KKT conditions ($w > 0$) $\implies$

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

- 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

- New algorithms: Khodadadi and M. (2019)
- Don't have to invert that matrix
- Must repeat for many tuning parameters
- Current work studies statistical properties:
- $\longrightarrow$ Algorithm is "exact", no statistical loss
- This is the best case

Solving big regression problems approximately

Core techniques

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

Model:

$$Y = X\beta_* + \epsilon$$

Least squares $\widehat{\beta} = \text{argmin}_\beta \frac{1}{2n}\|X\beta - Y\|_2^2$

Ridge regression $\widehat{\beta}_2(\lambda) = \text{argmin}_\beta \frac{1}{2n}\|X\beta - Y\|_2^2 + \frac{\lambda}{2}\|\beta\|_2^2$

LASSO $\widehat{\beta}_1(\lambda) = \text{argmin}_\beta \frac{1}{2n}\|X\beta - Y\|_2^2 + \lambda\|\beta\|_1$

Dantzig selector

Group LASSO

...

Typical results for approximation

The general philosophy: Find an approximation $\tilde{\beta}$ that is as close as possible to the solution of the original problem UNIFORMLY.

OLS:

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

Ridge:

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

etc.

Statistical analysis

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

$$\begin{aligned}
\mathbb{E}\|\theta - \tilde{\theta}\|_2^2 &= \mathbb{E}\|\tilde{\theta} - \widehat{\theta} + \widehat{\theta} - \theta\|_2^2 \\
&= \mathbb{E}\text{Approx error}^2 + \text{MSE}(\widehat{\theta}) + 2\mathbb{E}\left[(\text{Approx error})\left(\widehat{\theta} - \theta\right)\right] \\
&\leq \sup\left(\text{Approx error}^2\right) + \text{MSE}(\widehat{\theta}) \\
&\quad + 2\sup\left(\text{Approx error}\right) \times \text{RMSE}(\widehat{\theta})
\end{aligned}$$

Statistical analysis

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

$$\begin{aligned}
\mathbb{E}\|\theta - \tilde{\theta}\|_2^2 &= \mathbb{E}\|\tilde{\theta} - \widehat{\theta} + \widehat{\theta} - \theta\|_2^2 \\
&= \mathbb{E}\text{Approx error}^2 + \text{MSE}(\widehat{\theta}) + 2\mathbb{E}\left[(\text{Approx error})\left(\widehat{\theta} - \theta\right)\right] \\
&\leq \sup\left(\text{Approx error}^2\right) + \text{MSE}(\widehat{\theta}) \\
&\quad + 2\sup\left(\text{Approx error}\right) \times \text{RMSE}(\widehat{\theta}) \\
&\leq C\left(\sup\left(\text{Approx error}^2\right) + \text{MSE}(\widehat{\theta})\right)
\end{aligned}$$

- Previous work examines Approx error, assumes $\text{MSE}(\widehat{\theta})$ small

Statistical analysis

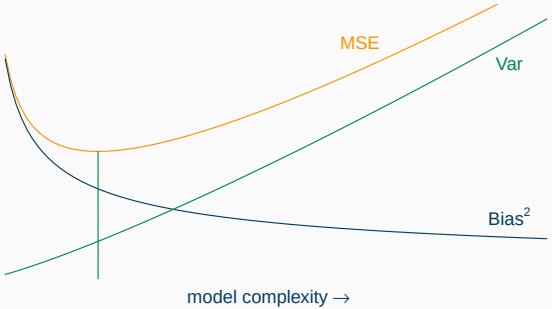
For an approximation $\tilde{\theta}$ of $\widehat{\theta}$,

$$\mathbb{E}\|\theta - \tilde{\theta}\|_2^2 = \text{Var}(\tilde{\theta}) + \text{Bias}(\tilde{\theta})^2$$

$$\begin{aligned}
\mathbb{E}\|\theta - \tilde{\theta}\|_2^2 &= \mathbb{E}\|\tilde{\theta} - \widehat{\theta} + \widehat{\theta} - \theta\|_2^2 \\
&= \mathbb{E}\text{Approx error}^2 + \text{MSE}(\widehat{\theta}) + 2\mathbb{E}\left[(\text{Approx error})\left(\widehat{\theta} - \theta\right)\right] \\
&\leq \sup\left(\text{Approx error}^2\right) + \text{MSE}(\widehat{\theta}) \\
&\quad + 2\sup\left(\text{Approx error}\right) \times \text{RMSE}(\widehat{\theta}) \\
&\leq C\left(\sup\left(\text{Approx error}^2\right) + \text{MSE}(\widehat{\theta})\right)
\end{aligned}$$

- Previous work examines Approx error, assumes $\text{MSE}(\widehat{\theta})$ small
- We examine the MSE of the procedure

Bias-variance tradeoff



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

Core techniques for least squares

If X fits into RAM, there exist excellent algorithms in LAPACK that are

- Double precision
- Very stable
- $O(np^2)$ when $n \gg p$.
- $O(n^2p)$ when $n \ll p$.
- 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 for least squares

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, (2017) "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 ($n \gg p$):

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

Compression

Basic Idea ($n \gg p$):

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

Basic Idea ($n \ll p$):

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

Compression

Basic Idea ($n \gg p$):

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

Basic Idea ($n \ll p$):

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

Finding QX for arbitrary Q and X takes $O(qnp)$ computations.

Solving large least-squares problems

Large n smaller p:

1. Introduce 2 new versions of compression.
2. General theoretical techniques.
3. Show how to choose tuning parameters (without extra computation).
4. Compression a bit worse under “nice” model. But better with outliers.

Large p smaller n:

1. Procedure for low-rank (sparse) regression.
2. Statistical performance is near-optimal under the right model.
3. Algorithm is approximate, but doesn’t lose anything.

Lots of observations

Family of 4 (current SOTA)

1. Full compression:

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

2. Partial compression:

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

Also called "Hessian Sketching".

Family of 4 (our versions)

Write:

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

$$W = XB$$

3. Linear combination compression:

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

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

4. Convex combination compression:

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

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

Why these?

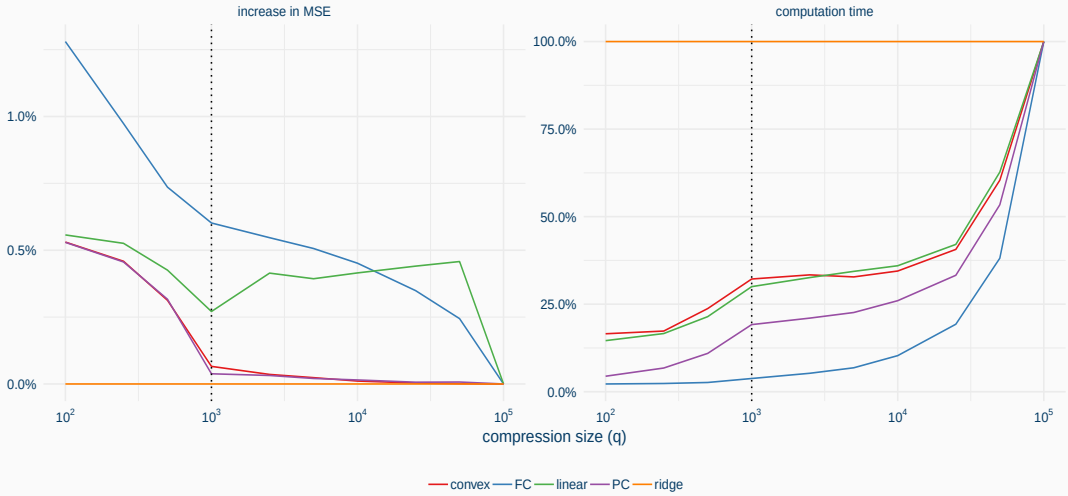
Turns out that FC is unbiased $\implies$ worse than OLS (has high variance)

PC is biased and empirics demonstrate low variance

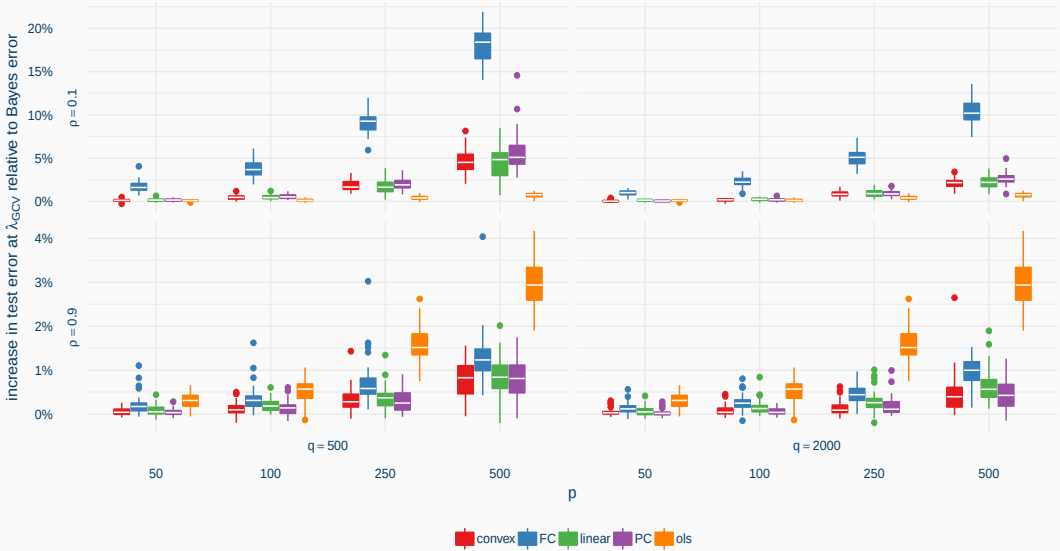
Combination should (and does) give better statistical properties

We do everything with an ℓ_2 penalty

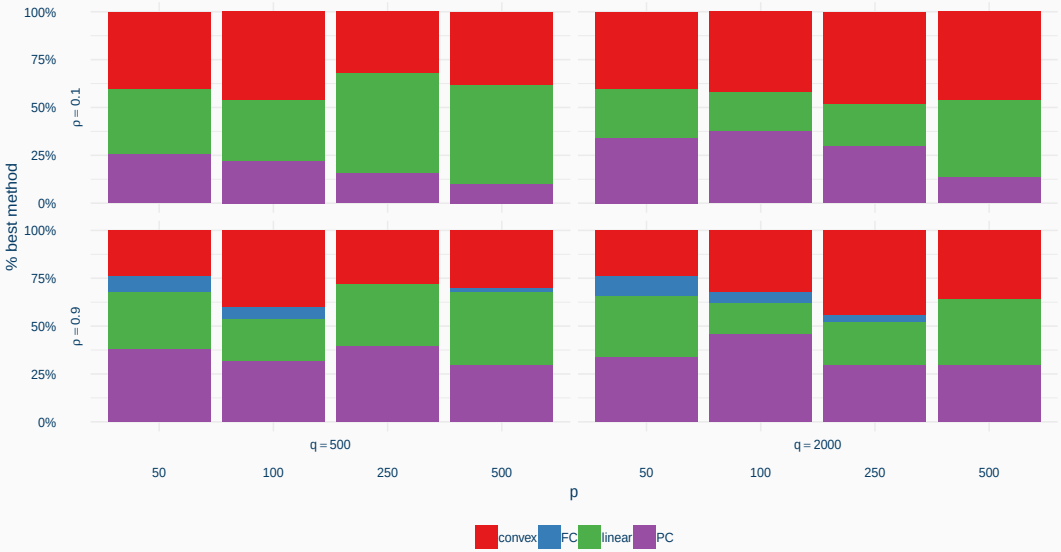
Timing vs. accuracy



Relative prediction error



Which one wins?



Tuning parameters selection

- We use GCV with the degrees of freedom:

$$\mathrm{GCV}(\lambda) = \frac{\frac{1}{n}\|X\tilde{\beta}(\lambda) - Y\|_2^2}{(1 - \mathrm{df}/n)^2}$$

- df is easy for full or partial compression (though ignored in literature)
- 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

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

- For normal linear model (OLS), we have

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

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

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

See e.g. Efron

Applying it

Because linear and convex combination compression have closed form expressions:

e.g. Linear combination:

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

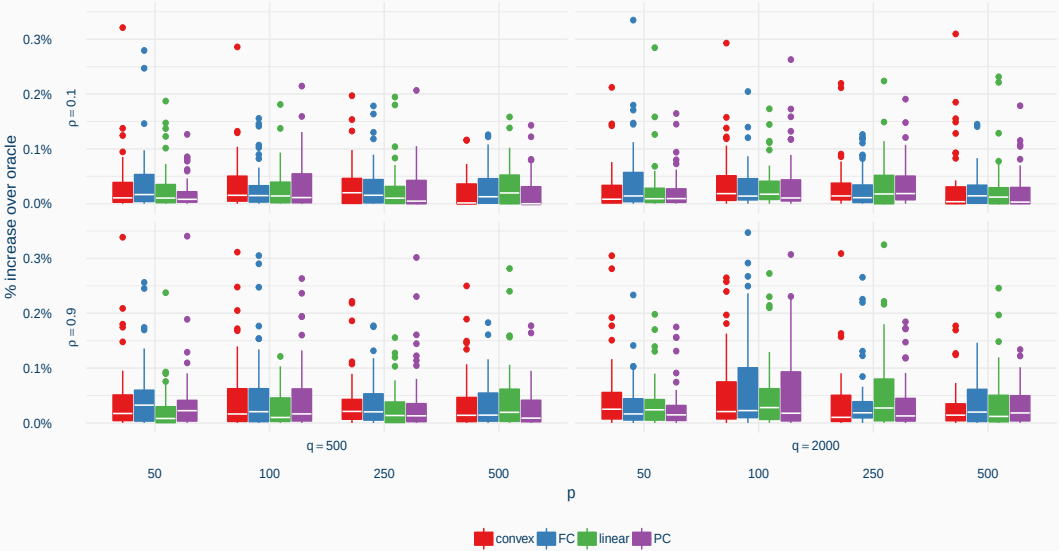
where

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

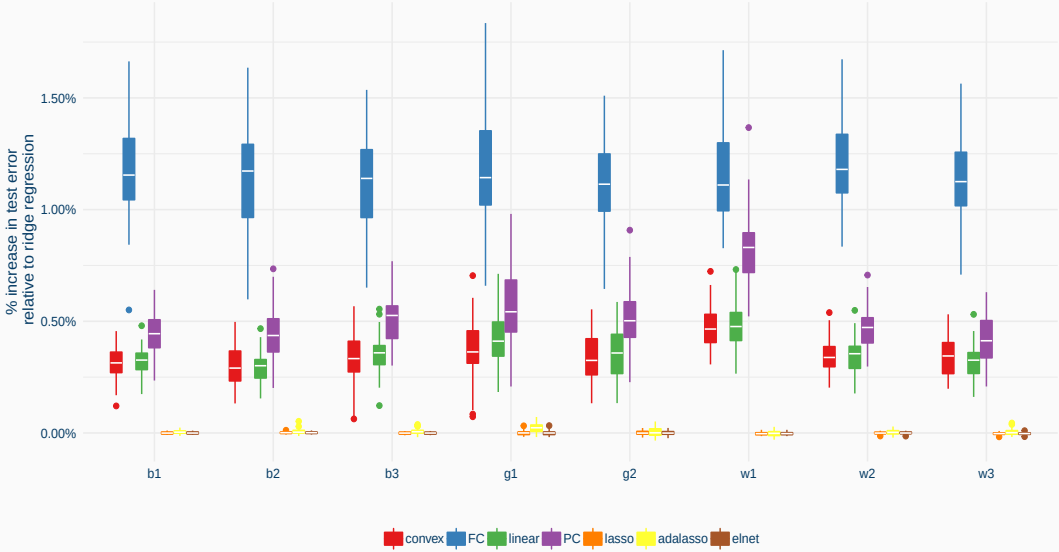
Apply every calculus rule you can find to yield a nasty expression which won't fit on this slide.

This is an unbiased estimate of df.

Risk estimation accuracy



Predicting read counts



Standard ridge results

Theorem

$$\text{bias}^2\left(\widehat{\beta}_2(\lambda) \mid X\right) = \lambda^2 \beta_*^\top V (D^2 + \lambda I_p)^{-2} V^\top \beta_*.$$

$$\text{tr}\left(\text{Var}\left[\widehat{\beta}_2(\lambda) \mid X\right]\right) = \sigma^2 \sum_{i=1}^{p} \frac{d_i^2}{(d_i^2 + \lambda)^2}.$$

Theorem

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

Note: $M = (X^\top X + \lambda I_p)^{-1} X^\top$ and $H = XM$

Corollary

If $\frac{1}{n}X^\top X = I_p$,

$$\begin{aligned}
\text{MSE}\left(\widehat{\beta}_2\right) &= b^2\left(\frac{\theta}{1+\theta}\right)^2 + \frac{p\sigma^2}{n(1+\theta)^2} \\
\text{MSE}\left(\tilde{\beta}_{FC}\right) &= b^2\left(\frac{\theta}{1+\theta}\right)^2 + \frac{p\sigma^2}{n(1+\theta)^2} + \frac{b^2 p\theta^2(s-2)_+}{q(1+\theta)^4} + \frac{p^2\theta^2 b^2}{q(1+\theta)^4} \\
\text{MSE}\left(\tilde{\beta}_{PC}\right) &= b^2\left(\frac{\theta}{1+\theta}\right)^2 + \frac{p\sigma^2}{n(1+\theta)^2} + \frac{p(s-2)_+ b^2}{q(1+\theta)^2} + \frac{pb^2}{q(1+\theta)^4}
\end{aligned}$$

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

What's the trick?

- All the estimators depend (at least) on

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

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

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

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

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

The takeaway message

Compression works in a fraction of the time.

Combining standard (FC and PC) is better.

Their genesis is an examination of the statistical performance.

Are there models under which compressed estimators are better?

The takeaway message

Compression works in a fraction of the time.

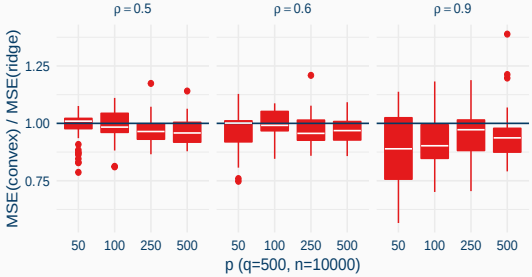
Combining standard (FC and PC) is better.

Their genesis is an examination of the statistical performance.

Are there models under which compressed estimators are better?

Answer: YES! If there are outliers that are also high leverage.

Proof of concept



$$\widehat{\beta}_{n+1} - \widehat{\beta}_n = \left(\frac{y_{n+1} - x_{n+1}^\top \widehat{\beta}_n}{1 + x_{n+1}^\top (X^\top X)^{-1} x_{n+1}} \right) (X^\top X)^{-1} x_{n+1}$$

Lots of measurements

Algorithmic methods (SOTA)

"Random projection"

OLS:

$$\widehat{\gamma} = \underset{\gamma}{\operatorname{argmin}} \frac{1}{2n} \|XQ\gamma - Y\|_2^2$$

$$\widehat{\beta}_{RP} \leftarrow Q\widehat{\gamma}$$

Before, sketched the covariance $X^\top X \longrightarrow X^\top Q^\top QX$.

RP sketches the Gram matrix $XX^\top \longrightarrow XQ^\top QX^\top$.

So existing analyses are equivalent: as long as the sketching grabs most of the action.

Grabbing the action

Before, we needed to get the high-leverage rows.

Now, we need “the high-leverage columns”.

Means most of the predictive information comes from a few variables.

Essentially, a low-rank predictor, driven by a few variables.

This is just sparse PCA regression.

Grabbing the action

Before, we needed to get the high-leverage rows.

Now, we need “the high-leverage columns”.

Means most of the predictive information comes from a few variables.

Essentially, a low-rank predictor, driven by a few variables.

This is just sparse PCA regression.

We do that instead.

High-dimensional PCR

Two stages:

1. $$\widehat{V} = \underset{V^\top V = I_d}{\operatorname{argmax}} \operatorname{tr}\left(V^\top X^\top X V\right)$$

2. $$\widehat{\gamma}_{pcr} = \underset{\gamma \in \mathbb{R}^d}{\operatorname{argmin}} \frac{1}{2n} \|Y - X\widehat{V}\gamma\|_2^2$$

$$\widehat{\beta}_{pcr} \leftarrow \widehat{V}\widehat{\gamma}_{pcr}$$

Here, $\widehat{V}$ is analogous to Q but it uses the structure of X.

But 1. is hard.

Algorithmically: Need the SVD of a $n \times p$ matrix $\longrightarrow O(np^2)$.

Statistically: $p \gg n \Longrightarrow \widehat{V}$ is inconsistent for the population analogue.

Fixing the difficulties

"Supervised" PCA First screen away most of the variables using Y.

Solves both problems if $\mathbb{E}\left[X_j Y\right] = 0 \Rightarrow \beta_{*,j} = 0$.

See work of Bair+Paul+Hastie (2004, 2006, 2008) or Tay, Friedman, Tibshirani (2018)

"Sparse" PCA Solve a constrained version of 1.

Good statistical properties for 1. but ignores Y

See d'Aspremont et al. (2007) Johnstone and Lu (2009) Zhang and Ghaoui (2011), among others

Our version — Fantope regression

Two stages:

1. $$\widehat{V} = \underset{\substack{0 \preceq VV^\top \preceq I \\ \operatorname{tr}(V)=d}}{\operatorname{argmax}} \operatorname{tr}\left(V^\top X^\top X V\right) - \lambda \|VV^\top\|_{1,1}$$

2. $$\widehat{\gamma}_{FR} = \underset{\gamma \in \mathbb{R}^d}{\operatorname{argmin}} \frac{1}{2n} \|Y - X\widehat{V}\gamma\|_2^2$$
$$\widehat{\beta}_{FR} \leftarrow \widehat{V}\widehat{\gamma}_{FR}$$

$\|VV^\top\|_{1,1}$ forces rows of V to be 0

$V_j = 0 \Longrightarrow V_j\gamma = 0 \Longrightarrow X_j$ doesn't predict Y.

Vu et al. (2013) call 1. "Fantope projection", nearly minimax optimal

Bad algorithmic properties: $O(\#\text{ iterations} \times p^3)$

Alternating direction method of multipliers

Restate your optimization

Original		Equivalent	
$\min_x$	$f(x) + g(x)$	$\min_{x,z}$	$f(x) + g(z)$
		s.t.	$x - z = 0$

Then, iterate the following with $\rho > 0$

$$x \leftarrow \operatorname*{argmin}_x f(x) + \frac{\rho}{2}\|x - z + u\|_2^2$$

$$z \leftarrow \operatorname*{argmin}_z g(z) + \frac{\rho}{2}\|x - z + u\|_2^2$$

$$u \leftarrow u + x - z$$

Why would you do this?

- It decouples f and g
- If f and g have the right structure $\Longrightarrow$ parallelizable
- There are often many ways to decouple a problem
- The individual minimizations don't have to be solved in closed form

ADMM for Fantope projection:

$$A \leftarrow \Pi_{\mathcal{F}^d}\left(V - U + \frac{1}{n\rho}X^\top X\right)$$

$$V \leftarrow \mathcal{S}_{\lambda/\rho}(A + U)$$

$$U \leftarrow U + A - V$$

$[\mathcal{S}_a(b)]_k = \text{sgn}(b_k)(|b_k| - a)_+$

Projecting onto the Fantope

Given an eigen decomposition of $Z = \sum_i \gamma_i z_i z_i^\top$.

$$\Pi_{\mathcal{F}^d}(Z) = \sum_i \gamma_i^+(\theta) z_i z_i^\top$$

$$\gamma_i^+(\theta) = \min(\max(\gamma_i - \theta,\ 0),\ 1), \qquad \theta \ \text{ s.t. } \ \sum_i \gamma_i^+(\theta) = d$$

- The γ-θ stuff solves a monotone, piecewise linear equation.
- But we have to do the decomposition at every iteration.

Conditions for convergence of ADMM

- When the updates are exact, all you need for convergence is

  1. f, g are convex, extended real valued.
  2. $f(x) + g(z) + u^\top(x - z)$ has a saddle point.

- The convergence rate is not well understood (seems linear).

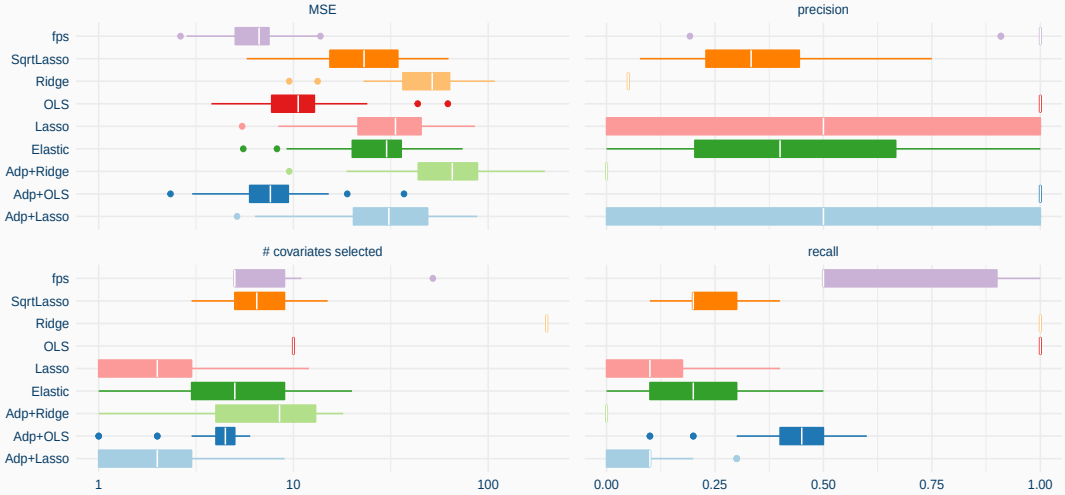
- It turns out, you can solve the minimizations approximately.

$$\sum_{k=1}^{\infty} \|\Pi(y^k) - \widetilde{\Pi}(y^k)\|_2 < \infty$$

- We do that using earlier work (Homrighausen and McDonald, 2016)

Eckstein and Bertsekas 1992

Results in simulations



Theoretical results

Assume many conditions, $s := |\beta_*|$, $\text{supp}(v) := \{j : v_j \neq 0\}$,

Theorem

$$\|\widehat{\beta}_{FR} - \beta_*\|_2 = O_P\left(\sigma\sqrt{\frac{(s^2 + d)\log p}{n}}\right),$$

and

$$\left|\text{supp}(\widehat{\beta}_{FR}) \triangle \text{supp}(\beta_*)\right| = O_P\left(\sigma\sqrt{\frac{s^2 \log p}{n}}\right).$$

The big picture

Statistics + optimization + approximation

When you have large datasets and complicated estimators

1. Algorithms can avoid hard computations, enable inference.
2. Understanding how algorithms work can motivate statistical advances.
3. Understanding statistics may motivate new algorithms.

Related work on statistics + optimization + approximation

- Approximation for Least Squares ($n \gg p$)
  - Homrighausen and McDonald. (2019+). Under review.
- Approximation for dimension reduction ($p \gg n$)
  - Homrighausen and McDonald. (2016). JCGS.
  - Ding and McDonald. (2017). Bioinformatics.
  - Ding and McDonald. (2019). Under review.
- Algorithms for large data
  - ADMM for large constrained kernel PCA.
  - McDonald and Khodadadi. (2019). AAAI.
  - Trend filtering for Spatio-temporal exponential families.

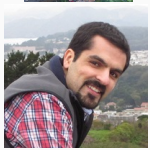
Related work on statistics + optimization + approximation

- Approximation for Least Squares ($n \gg p$)
  - Homrighausen and McDonald. (2019+). Under review.
- Approximation for dimension reduction ($p \gg n$)
  - Homrighausen and McDonald. (2016). JCGS.
  - Ding and McDonald. (2017). Bioinformatics.
  - Ding and McDonald. (2019). Under review.
- Algorithms for large data
  - ADMM for large constrained kernel PCA.
  - McDonald and Khodadadi. (2019). AAAI.
  - Trend filtering for Spatio-temporal exponential families.

Tuning parameter selection and risk estimation

- GCV and SURE for compressed regression
- SURE for exponential families.
- 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). JSCS.
  - McDonald and Shalizi. (2018+). Under review.
  - McDonald, Shalizi and Schervish. (2017). JMLR.

Collaborators and funding



Institute for
New Economic
Thinking

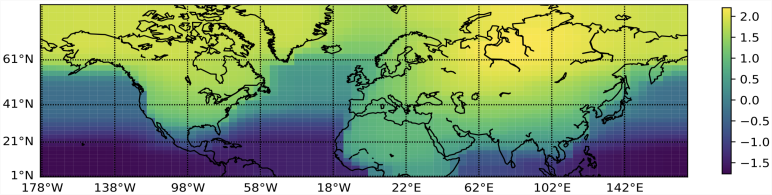
Appendix

Algorithm 1 Linearized ADMM

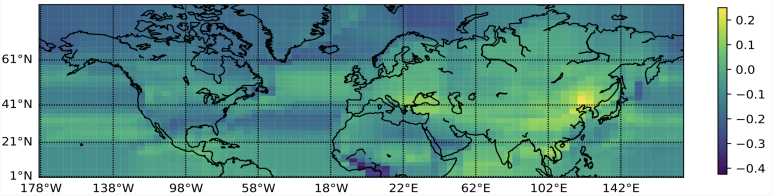
Input data y, penalty matrix D, $\epsilon, \rho, \lambda_t, \lambda_s > 0$.
Set $h \leftarrow 0, z \leftarrow 0, u \leftarrow 0$. ▷ Initialization
repeat
  $h_k \leftarrow \mathcal{W}\left(\frac{y_k^2}{\mu} \exp\left(\frac{1-\mu u_k}{\mu}\right)\right) + \frac{1-\mu u_k}{\mu}$ for all $k = 1, \ldots TS$. ▷ Primal update
  $z \leftarrow S_{\rho\lambda}(u)$. ▷ Elementwise soft thresholding
  $u \leftarrow u - z$. ▷ Dual update
until $\max\{\|Dh - z\|, \ \|z^{m+1} - z^m\|\} < \epsilon$
Return z.

Results

Average variance



Change in average variance from 1961–2011



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.

Subsampling $Q = \pi\tau$ for π a permutation of I and $\tau = [I_q \; 0]$. QX 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 QX takes $O\left(\frac{qnp}{s}\right)$ "computations" on average.

Why combine?

- $Y = X\widehat{\beta} + \widehat{e}$ with $P(\widehat{e} \in \mathrm{col}(X)) = 0$, and $\mathbb{E}[\widehat{e}] = 0$.

$$\begin{aligned}\widehat{\beta}_{FC}(0) &= (X^\top Q^\top QX)^\dagger X^\top Q^\top QY = (X^\top Q^\top QX)^\dagger X^\top Q^\top Q(X\widehat{\beta} + \widehat{e}) \\ &= \widehat{\beta} + (X^\top Q^\top QX)^\dagger X^\top Q^\top Q\widehat{e} \\ \Rightarrow \mathbb{E}\left[\widehat{\beta}_{FC}(0)\right] &= \mathbb{E}\left[\widehat{\beta}\right] = \beta_*\end{aligned}$$

$$\begin{aligned}\widehat{\beta}_{PC}(0) &= (X^\top Q^\top QX)^\dagger X^\top Y = (X^\top Q^\top QX)^\dagger X^\top (X\widehat{\beta} + \widehat{e}) \\ &= (X^\top Q^\top QX)^\dagger X^\top X\widehat{\beta} \\ \Rightarrow \mathbb{E}\left[\widehat{\beta}_{PC}(0)\right] &= (X^\top Q^\top QX)^\dagger X^\top X\beta_*\end{aligned}$$

Simulation setup ($n \gg p$)

- Draw $X_i \sim \text{MVN}(0, (1-\rho)I_p + \rho \mathbf{1}\mathbf{1}^\top)$
- Draw $\beta \sim \text{N}(0, \tau^2 I_p)$
- Draw $Y_i = X_i^\top \beta_* + \epsilon_i$ with $\epsilon_i \sim \text{N}(0, \sigma^2)$.
- For this model, the optimal estimator (in MSE) is

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

with $\lambda_* = \frac{\sigma^2}{n\tau^2}$

Divergence of Linear Combination

Define the linear combination prediction as $\widehat{Y}_{LC} = XB(\lambda)\widehat{\alpha}_{LC} = XB(\lambda)\left(B^\top(\lambda)X^\top XB(\lambda)\right)^\dagger B^\top(\lambda)X^\top Y$ and let $Z := XB(\lambda)$ so that $Z(Z^\top Z)^\dagger Z^\top Y =: P_Z Y$.

Theorem:

$$\begin{aligned}\mathrm{div}_{lin}(\lambda) = \widehat{\alpha}_{FC}\mathrm{df}_{FC} + \widehat{\alpha}_{PC}\mathrm{df}_{PC} + \mathrm{tr}(P_Z) - \widehat{\alpha}_{PC}\,\mathrm{tr}(P_Z H_{PC}) - \widehat{\alpha}_{FC}\,\mathrm{tr}(P_Z H_{FC}) \\ - \mathrm{tr}\left(Z^\dagger [H_{PC}\widehat{Y}_{LC}\; H_{FC}\widehat{Y}_{LC}]\right).\end{aligned}$$

Divergence of Convex Combination

Theorem:

$$\begin{aligned}\mathrm{div}_{con}(\lambda) = \widehat{\alpha}_{PC}\mathrm{df}_{PC} + (1-\widehat{\alpha}_{PC})\mathrm{df}_{FC} \\ + \frac{(Y-\widehat{Y}_{FC})^\top H_{PC}(\widehat{Y}_{PC}-\widehat{Y}_{FC})}{\left(\widehat{Y}_{PC}-\widehat{Y}_{FC}\right)^\top\left(\widehat{Y}_{PC}-\widehat{Y}_{FC}\right)} + \frac{\widehat{Y}_{PC}^\top(I_n-H_{FC})(\widehat{Y}_{PC}-\widehat{Y}_{FC})}{\left(\widehat{Y}_{PC}-\widehat{Y}_{FC}\right)^\top\left(\widehat{Y}_{PC}-\widehat{Y}_{FC}\right)} \\ - \frac{2\left(\widehat{Y}_{PC}-\widehat{Y}_{FC}\right)^\top\left(Y-\widehat{Y}_{FC}\right)}{\left(\left(\widehat{Y}_{PC}-\widehat{Y}_{FC}\right)^\top\left(\widehat{Y}_{PC}-\widehat{Y}_{FC}\right)\right)^2}(\widehat{Y}_{PC}-\widehat{Y}_{FC})^\top(H_{PC}-H_{FC})(\widehat{Y}_{PC}-\widehat{Y}_{FC}).\end{aligned}$$

Flop count for different algorithms

Approach	$O(\cdot)$
Linear system	$np^2 + p^3$
Low-rank linear system	$npr + r^3$
Gradient descent	$n^{3/2}p^2 \log \epsilon^{-1}$
Acc. gradient descent	$n^{5/4}p^{3/2} \log \epsilon^{-1}$
Coordinate descent	$n^{3/2}p \log \epsilon^{-1}$
SVRG, SDCA, SAG	$(np + n^{1/2}p^2) \log \epsilon^{-1}$
Catalyst, APPA	$(np + n^{3/4}p^{3/2}) \log \epsilon^{-1}$
DSPDC	$npr + (nr + n^{3/4}p^{3/2}r) \log \epsilon^{-1}$
Iterative Hessian sketch	$np \log p + n^{1/4}p^{3/2}r \log^2 \epsilon^{-1}$
Dual random projection	$np \log n + (nr^2 + r^3) \log \epsilon^{-1}$

Genes and the SPC model

sparsity of Σ_{xx}^{-1}	1.0000	0.9999	0.9998	0.9995	0.9991	0.9984	0.9975	0.9963	0.9946	0.9922
% non-zero β_*'s	0.0162	0.0216	0.0287	0.0418	0.0618	0.0843	0.1193	0.1803	0.2645	0.3699
False Negative Rate	0.0000	0.2500	0.4340	0.6117	0.7374	0.8077	0.8641	0.9100	0.9387	0.9562