Principal component-guided sparse regression with pcLasso

Kenneth Tay (PhD student, Stanford University)

with Jerry Friedman & Rob Tibshirani

Jul 29, 2019

Two big ideas in supervised learning

- Supervised learning setting: $\mathbf{X} \in \mathbb{R}^{n \times p}$, $\boldsymbol{y} \in \mathbb{R}^n$ or $\boldsymbol{y} \in \{0, 1\}^n$ (usually $p \gg n$)
- Assume $\boldsymbol{y}$ and columns of $\mathbf{X}$ are centered

Sparsity: The response can be modeled well with just a handful of features.

- The **lasso**:

$$\underset{\beta}{\text{minimize}} \quad J(\beta_0, \beta) = \frac{1}{2}\|\boldsymbol{y} - \mathbf{X}\beta\|_2^2 + \lambda\|\beta\|_1.$$

- **Good:** Fast, sparse solution
- **Not so good:** If signal is *weak* and *spread out over many correlated features* (e.g. genes/proteins along a biological pathway), lasso focuses on individual features, may not predict well

Two big ideas in supervised learning

Dimensionality reduction with principal components: Main sources of variability (and hopefully signal) can be captured by a handful of derived variables.

- Let $\mathbf{X} = (\mathbf{UD})\mathbf{V}^T$ be the singular value decomposition (SVD) of $\mathbf{X}$.
- Columns of $\mathbf{UD}$ are **principal components (PCs)** of $\mathbf{X}$.
- **PC regression**: OLS of $\mathbf{y}$ on first k PCs.
- **Good:** If signal is *weak* and *spread out over many correlated features*, PC regression aggregates signal across features, giving better prediction
- **Not so good:** PC regression not sparse in original variables

Marrying the lasso and PC regression

Goal: Devise a method that...

- Pools together signal from correlated features
- Is sparse in the original features

Sometimes, features come in groups (e.g. one-hot encodings of categorical features, genes in the same pathway)

Sub-Goal: Devise a method that makes use of feature grouping information

Our general idea

- Predictions $\mathbf{X}\beta = (\mathbf{UD})(\mathbf{V}^T\beta)$.
- $\mathbf{V}^T\beta$: Coordinates in principal component space. Think of predictions as a linear combination of PCs with coefficients $\mathbf{V}^T\beta$
- General idea: **Penalize the coefficients in the principal component space!**

One possibility:

$$\begin{aligned}\underset{\beta}{\text{minimize}} \qquad & \frac{1}{2}\|\boldsymbol{y} - \mathbf{X}\beta\|_2^2 + \lambda\|\beta\|_1 + \frac{\theta}{2}\left\|\mathbf{V}^T\beta\right\|_2^2 \\ = & \frac{1}{2}\|\boldsymbol{y} - \mathbf{X}\beta\|_2^2 + \lambda\|\beta\|_1 + \frac{\theta}{2}\beta^T\mathbf{V}\mathbf{V}^T\beta.\end{aligned}$$

Principal components lasso ("pcLasso"): single group case

Principal components lasso ("pcLasso"):

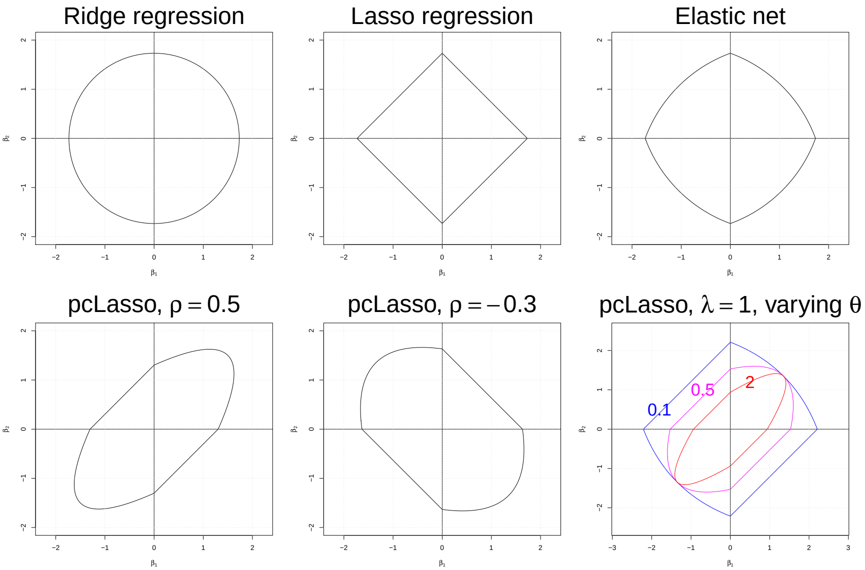
$$\underset{\beta}{\text{minimize}} \quad \frac{1}{2}\|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda\|\beta\|_1 + \frac{\theta}{2}\beta^T \mathbf{V}\mathbf{Z}\mathbf{V}^T\beta, \text{ where}$$

$$\mathbf{Z} = \mathbf{D}_{d_1^2 - d_j^2} = \begin{pmatrix} d_1^2 - d_1^2 & & & \\ & d_1^2 - d_2^2 & & \\ & & \ddots & \\ & & & d_1^2 - d_m^2 \end{pmatrix},$$

$d_1, \ldots, d_m$ are the singular values of $\mathbf{X}$.

pcLasso gives no penalty ("**a free ride**") to the part of β that lines up with the first PC; penalty increases for components that line up with the second, third etc. components.

Penalty contours: two predictors

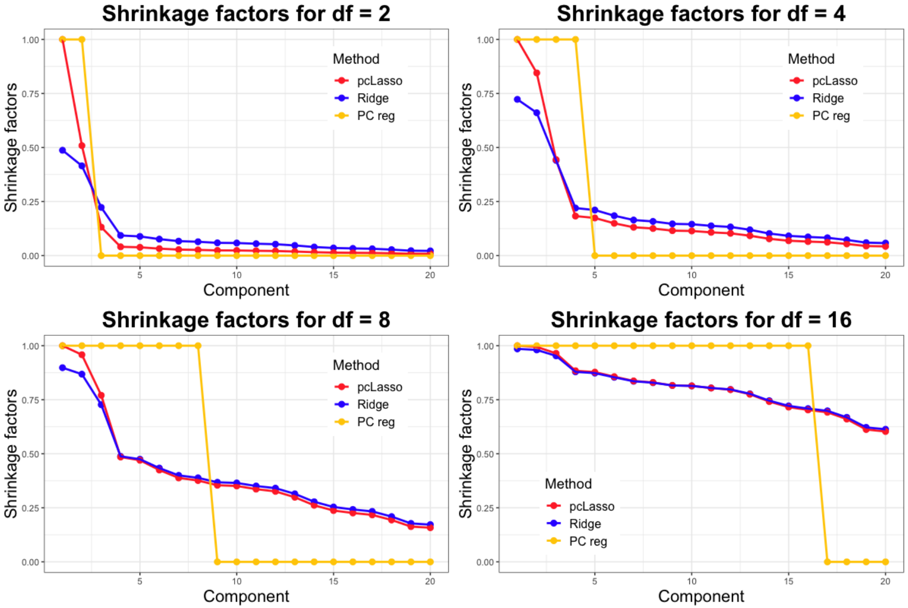


Comparing shrinkage factors for prediction

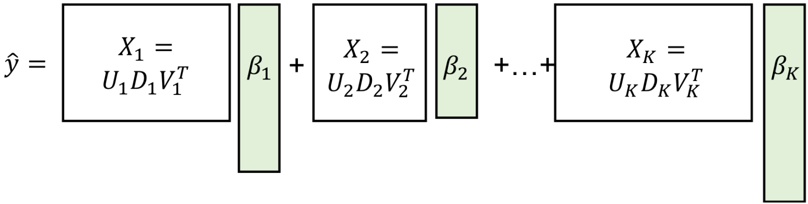
Method	Predictions
Ordinary linear regression	$\sum_{j=1}^{m} \boldsymbol{u}_j \boldsymbol{u}_j^T \boldsymbol{y}$
Principal components regression of rank k	$\sum_{j=1}^{m} 1\{j \leq k\} \boldsymbol{u}_j \boldsymbol{u}_j^T \boldsymbol{y}$
Ridge regression	$\sum_{j=1}^{m} \frac{d_j^2}{d_j^2 + \mu} \boldsymbol{u}_j \boldsymbol{u}_j^T \boldsymbol{y}$
pcLasso without ℓ_1 penalty	$\sum_{j=1}^{m} \frac{d_j^2}{d_j^2 + \theta(d_1^2 - d_j^2)} \boldsymbol{u}_j \boldsymbol{u}_j^T \boldsymbol{y}$

* $\boldsymbol{u}_j = j$*th column of* $\mathbf{U}$, $m = rank(\mathbf{X})$
* k, μ, θ: *hyperparameters*

Comparing shrinkage factors: $\mathbf{X} \approx$ rank-3 matrix



Principal components lasso ("pcLasso") for groups



Principal components lasso for groups:

$$\underset{\beta}{\text{minimize}} \quad J(\beta) = \frac{1}{2}\|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda\|\beta\|_1 + \frac{\theta}{2}\sum_{k=1}^{K} \beta_k^T \mathbf{V}_k \mathbf{D}_{d_{k1}^2 - d_{kj}^2} \mathbf{V}_k^T \beta_k.$$

The quadratic penalty gives a *free ride* to components of β_k that align with the first PC of group k.

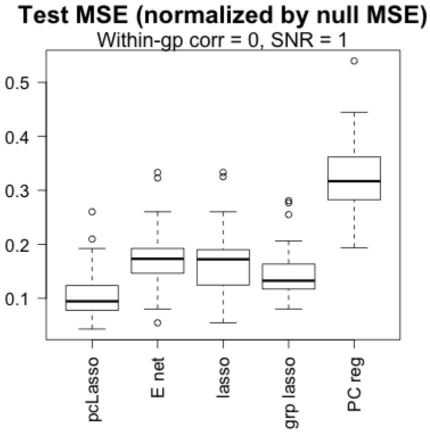
Some notes on computation

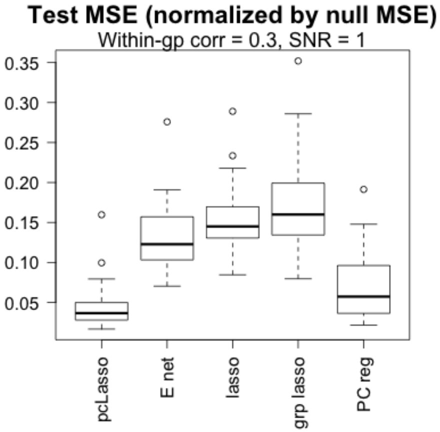
$$\underset{\beta}{\text{minimize}} \quad J(\beta) = \frac{1}{2}\|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda\|\beta\|_1 + \frac{\theta}{2}\sum_{k=1}^{K} \beta_k^T \mathbf{V}_k \mathbf{D}_{d_{k1}^2 - d_{kj}^2} \mathbf{V}_k^T \beta_k.$$

- J convex, non-smooth component separable $\Rightarrow$ coordinate descent works.
- Can be extended easily to logistic and Cox regression models.
- Costly part: initial SVD of each $\mathbf{X}_k$.
  - Possible approximation: Use SVD of lower rank instead.
  - After initial SVDs, pcLasso **is almost as fast as** `glmnet`!

Example: simulated data

- $n = 200$, $p = 50$, 5 groups of 10 predictors each
- Response: a linear combination of top eigenvector in first 2 groups





Summary

- Introduced a new method, *principal components lasso*, which combines lasso sparsity with shrinkage toward leading PCs
- Works when features come in pre-assigned groups (non-overlapping and overlapping)
- Computationally fast
- Other things we did (see paper on arXiv:1810.04651):
  - Derived some theoretical properties
  - Degrees of freedom formula for single group, full-rank case
  - Strong rules for efficient screening of variables
  - Connection to group lasso and sparse group lasso
  - Extensive simulation results
- R package available: `pcLasso`

Thank you!

arXiv:1810.04651
kjytay@stanford.edu
kjytay.github.io