STAT 24620=FINM 34700, STAT 32950
Multivariate Data Analysis
Lecture 11: Sparse PCA and Covariance Shrinkage

Jingshu Wang

The University of Chicago

Outline

From regularized prediction to regularized structure

- Lecture 10 regularized regression coefficients when prediction became unstable in high dimension.
- Today we regularize unsupervised structure: covariance matrices and principal components.
- The same high-dimensional problem appears again: too many variables, too much noise, not enough stable information.

Two goals for today

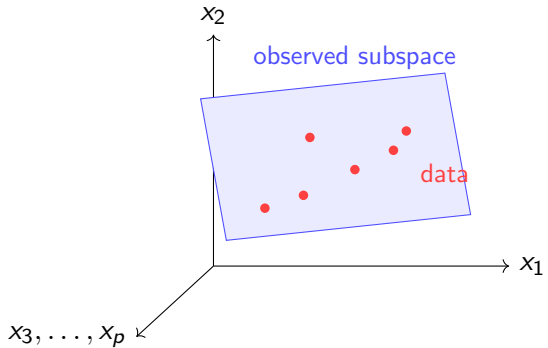
- stabilize covariance estimation by shrinkage;
- make dimension reduction more interpretable through sparsity.

Main theme: in high dimension, unsupervised structure also needs regularization.

Estimating covariance in high dimension

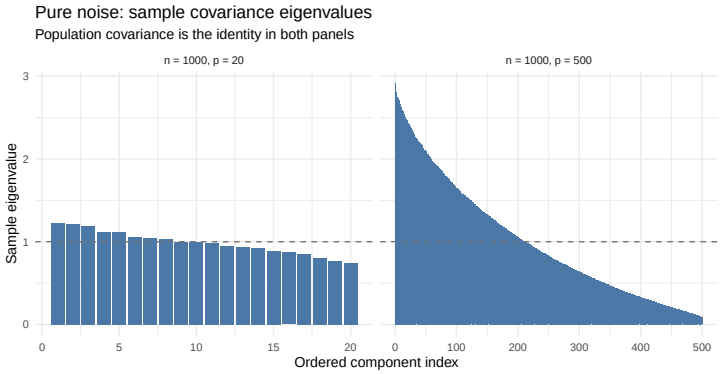
- $\mathbf{S}$ has $O(p^2)$ parameters.
- When p is large relative to n, estimation is noisy.
- If $p \geq n$, the data lie in a low-dimensional subspace.

$$\operatorname{rank}(\mathbf{S}) \leq n - 1,$$



But singularity is not the main issue — noise can distort the structure even when $p < n$, whenever p is large relative to n.

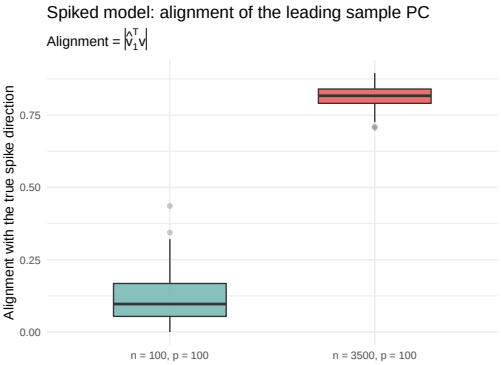
Pure noise already looks different in high dimension



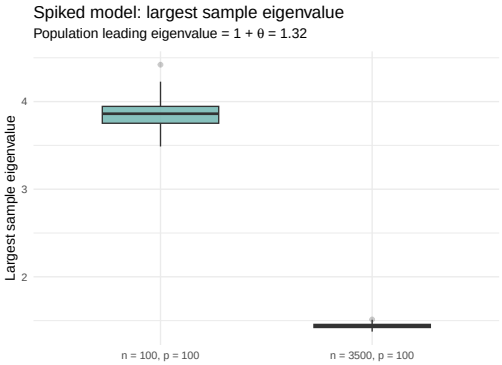
- Here $X_i \sim N(0, I_p)$, so there is no true factor structure.
- Left: $n = 1000$, $p = 20$.
- Right: $n = 1000$, $p = 500$.
- In the high-dimensional regime, the top sample eigenvalues spread out and can look artificially dominant.

So a "big first PC" is not, by itself, evidence of real signal.

A real signal does not guarantee an accurate leading PC



Alignment of $\hat{v}_1$ with the true spike v



Largest sample eigenvalue $\hat{\lambda}_1$

- Spiked model: $\Sigma = I_p + \theta vv^\top$ with fixed $\theta = 0.32$ and $p = 100$.
- Compare small n ($n = 100$) vs. large n ($n = 3500$).
- With smaller n, the leading PC is poorly aligned with v, even when $\hat{\lambda}_1$ looks large.

Why does PCA behave this way in high dimension?

- Sample covariance $\mathbf{S}$ is noisy when p/n is not small.
- PCA looks for directions of maximum variance, but noise also creates variance.
- With many directions available, some noise directions appear strong, and nearby directions are hard to distinguish.
- This behavior is formalized by the *Marchenko–Pastur law*: even pure noise produces a spread of sample eigenvalues.

Eigenvalues can be inflated, and eigenvectors can be unstable.

A basic shrinkage idea

Instead of using $\mathbf{S}$ directly, estimate covariance by

$$\hat{\Sigma}_\alpha = (1-\alpha)\mathbf{S} + \alpha T, \qquad 0 \le \alpha \le 1,$$

where T is a simpler target matrix.

Common targets

- $T = \tau I$: shrink toward spherical covariance;
- $T = \mathrm{diag}(\mathbf{S})$: keep marginal variances but shrink correlations;
- structured targets motivated by the scientific context.

- Each entry of $\mathbf{S}$ is noisy; shrinkage replaces it with a weighted average of the data-driven estimate and a structured target T.
- This stabilizes the estimate by reducing variability across samples, at the cost of introducing bias.
- This is directly analogous to ridge regression, which shrinks noisy coefficients toward zero.

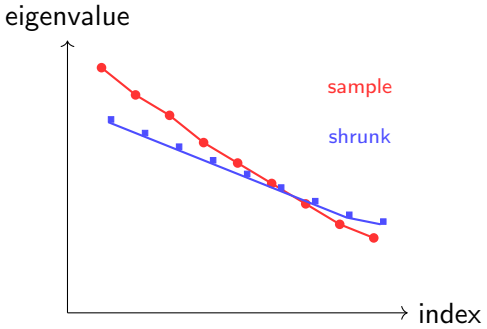
Shrinkage trades a small amount of bias for a substantial reduction in variance.

What does shrinkage do? (eigenvalue view)

- Shrinkage pulls the covariance toward a simple target.
- When $T = \tau I$,

$$\hat{\lambda}_j^{\text{shrink}} = (1 - \alpha)\hat{\lambda}_j + \alpha\tau.$$

- A common choice is $\tau = \frac{1}{p}\,\text{tr}(\mathbf{S})$, the average sample variance.
- Large eigenvalues move down; small ones move up.
- The spectrum becomes less dispersed and better conditioned.



What does shrinkage do?

If $T = \tau I$ and

$$\mathbf{S} = \sum_{j=1}^{p} \hat{\lambda}_j \hat{v}_j \hat{v}_j^\top,$$

then

$$\hat{\Sigma}_\alpha = (1-\alpha)\mathbf{S} + \alpha\tau I = \sum_{j=1}^{p} \left\{(1-\alpha)\hat{\lambda}_j + \alpha\tau\right\} \hat{v}_j \hat{v}_j^\top.$$

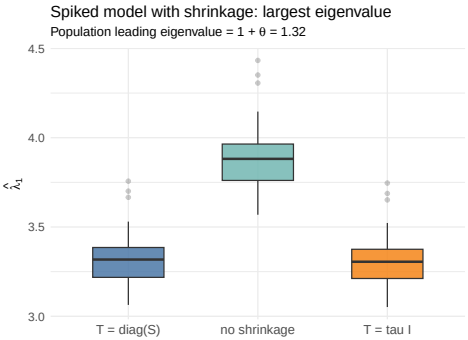
So, for this target:

- eigenvalues are shrunk toward τ;
- eigenvectors are unchanged.

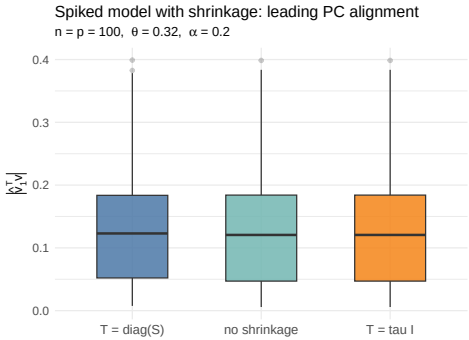
Other targets. If $T = \mathrm{diag}(\mathbf{S})$ or another structured target, then $\mathbf{S}$ and T usually do not share eigenvectors. Therefore shrinkage can also change the directions.

The target determines what structure is preserved and what structure is regularized.

What changes after shrinkage in the spiked example?



Largest eigenvalue across 150 replications



Alignment of the leading eigenvector with v

- We compare no shrinkage, $T = \tau I$, and $T = \mathrm{diag}(\mathbf{S})$, with $\alpha = 0.2$.
- Shrinkage clearly reduces the top eigenvalue and its variability.
- Here $T = \tau I$ leaves eigenvectors unchanged; $T = \mathrm{diag}(\mathbf{S})$ changes them only slightly.

Shrinkage corrects the scale of the leading component, but not its direction.

Beyond PCA: where else does shrinkage matter?

- **In PCA:**

$$S = \sum_{j=1}^{p} \hat{\lambda}_j \hat{v}_j \hat{v}_j^\top$$

  Shrinkage stabilizes the eigenvalues used to rank components.

- **Gaussian mixture models:**

$$x \sim \mathcal{N}(\mu_k, \mathbf{\Sigma}_k)$$

- **Mahalanobis distance:**

$$d(x, y) = (x - y)^\top \mathbf{\Sigma}^{-1} (x - y)$$

- **LDA:**

$$\delta_k(x) = x^\top \mathbf{\Sigma}^{-1} \mu_k - \tfrac{1}{2} \mu_k^\top \mathbf{\Sigma}^{-1} \mu_k$$

All rely on estimating or inverting covariance matrices.

Shrinkage stabilizes these methods in high dimensions.

From shrinkage to structure

- Shrinkage improves stability of covariance estimation.
- But the leading principal components can still be poorly aligned with the true signal.
- In addition, the loadings are often dense and hard to interpret.

To recover meaningful directions, we need to impose additional structure on the eigenvectors.

Idea: sparse principal component analysis (sparse PCA)

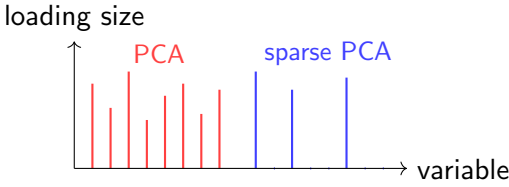
Sparse PCA: adding structure to PCA

PCA solves

$$\max_{\|v\|_2=1} v^\top S v.$$

Sparse PCA imposes sparsity:

$$\max_{v} v^\top S v \quad \text{s.t.} \quad \|v\|_2 = 1,\ \|v\|_0 \leq k.$$



- Same variance criterion as PCA.
- Only k variables can have nonzero loadings.
- Smaller $k \rightarrow$ more interpretable, but less variance explained.

Sparse PCA imposes structure on the eigenvectors.

From ℓ_0 to ℓ_1

Sparse PCA can be formulated as

$$\max_{\|v\|_2=1,\ \|v\|_0\le k} v^\top S v.$$

- k directly controls the number of nonzero loadings.
- But the problem is combinatorial and hard to solve.

A common relaxation replaces ℓ_0 by ℓ_1:

$$\max_{\|v\|_2=1} \Big\{ v^\top S v - \lambda \|v\|_1 \Big\}.$$

- λ controls sparsity more smoothly: larger λ gives sparser solutions.
- The problem is still nonconvex, but the ℓ_1 penalty gives useful thresholding algorithms.

ℓ_1 provides a tractable way to induce sparsity.

Soft-thresholding and algorithmic intuition

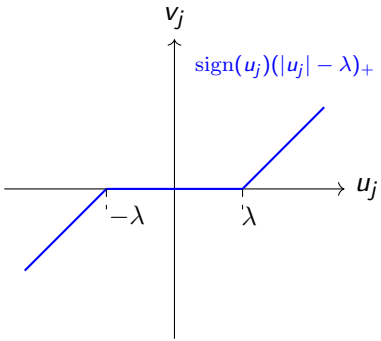
Algorithm (first sparse PC)
Iterate until convergence:

- Given v, compute $u = Sv$
- Apply soft-thresholding:

$$v_j \leftarrow \text{sign}(u_j)\,(|u_j| - \lambda)_+$$

- Normalize $\|v\|_2 = 1$

PCA step + thresholding step



Multiple sparse PCs

After computing v_1, how do we get $v_2, \ldots, v_m$?

In PCA: orthogonality comes automatically.
In sparse PCA: it does not.

Two strategies

- **Sequential (deflation):**

$$S \leftarrow S - \hat{\lambda}_1 v_1 v_1^\top$$

  then repeat.

- **Joint:**

$$\max_{V^\top V = I} \operatorname{tr}(V^\top S V) \quad \text{with sparse columns}$$

Modeling choice: enforce orthogonality ($V^\top V = I$) or relax it (allow correlated components).

Sparsity and orthogonality are competing goals.

Sparse PCA and the lasso analogy

- In supervised learning, lasso regularizes regression coefficients and can set some exactly to zero.
- In sparse PCA, regularization acts on the loading vector and can set some loadings exactly to zero.
- So the selected variables are not chosen for predicting a response Y.
- They are chosen for defining a low-dimensional latent direction.

Lasso selects predictors for a supervised target; sparse PCA selects variables for an unsupervised component.

What changes in sparse PCA?

- Ordinary PCA maximizes sample variance with no structural constraint.
- Sparse PCA restricts the loading vector to be sparse.
- As a result, it typically achieves a smaller value of $v^\top S v$ on the sample.
- However, the sparse solution can be more stable and better aligned with the true signal.

Sparse PCA trades sample variance for structure, which can improve population performance.

Sparse PCA versus covariance shrinkage

Covariance shrinkage

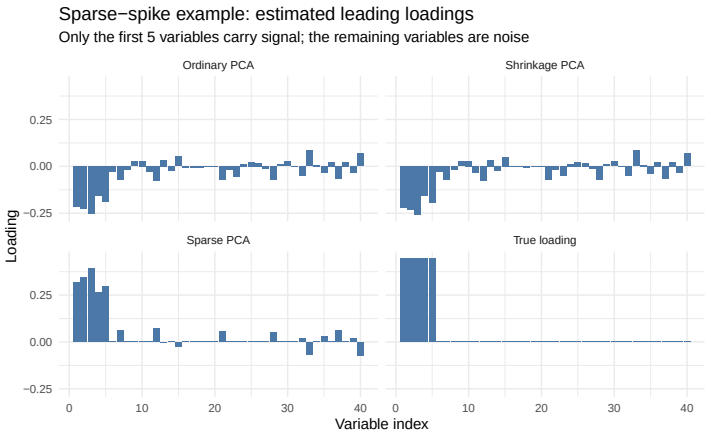
- stabilizes eigenvalues and improves conditioning;
- does not directly improve eigenvector estimation;
- mainly useful for covariance estimation and downstream tasks.

Sparse PCA

- imposes sparsity on eigenvectors;
- can accurately recover the leading direction if it is truly sparse;
- can perform poorly if the true direction is dense.

Shrinkage stabilizes scale; sparse PCA targets structure.

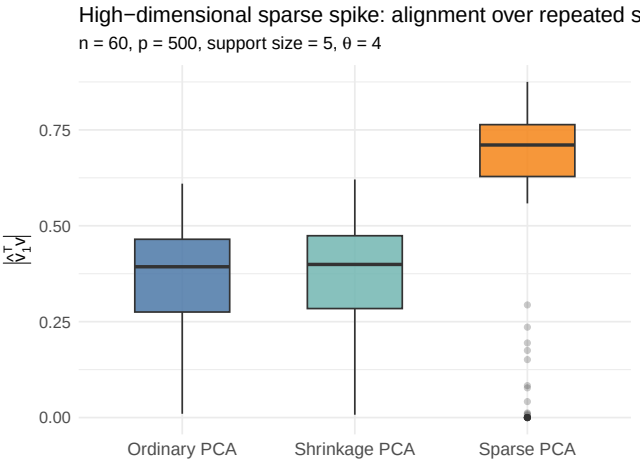
A sparse-spike example



- Simulation: $n = 60$, $p = 500$.
- The true leading eigenvector is sparse: only the first 5 variables carry signal.
- PCA and shrinkage PCA spread weight over many noise variables.
- Sparse PCA concentrates more strongly on the signal block.

When the population eigenvector is sparse, sparsity can improve both interpretation and recovery.

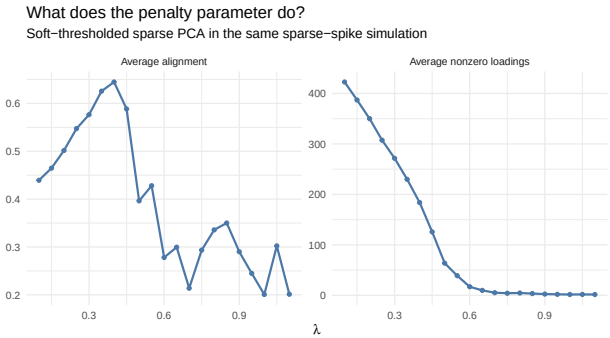
Repeated samples: sparse PCA recovers the direction better



- We repeat the sparse-spike simulation 150 times.
- PCA and shrinkage PCA have only moderate alignment with the true direction.
- Sparse PCA is substantially better in this setting.
- The gain comes from the sparsity assumption, not from eigenvalue shrinkage.

Shrinkage helps covariance estimation; sparse PCA helps direction recovery when sparsity is real.

How should we tune sparse PCA?



$$\text{alignment} = |\hat{v}^\top v|$$

absolute cosine similarity between estimated and true directions.

- Larger λ gives sparser loadings.
- Moderate λ works best here: too small stays dense, too large loses signal.

- On real data, the true direction is unknown:
  - variance-based criteria (even out-of-sample) often favor less sparsity;
  - we look for λ where the selected variables and directions remain stable across nearby values.

A practical workflow

- Standardize variables when scales are not comparable.
- Inspect the spectrum and the correlation structure.
- If covariance estimation looks unstable, consider shrinkage first.
- If PCA loadings are too dense to interpret, consider sparse PCA.
- Check stability: do the leading variables persist across resamples or tuning choices?

In unsupervised learning, interpretability and stability are often more important than maximizing sample fit.

What should you take away?

- In high dimension, leading sample PCs can be unstable and misleading.
- Shrinkage improves stability, but not directions.
- Sparse PCA can recover structure when the signal is sparse.
- But it is sensitive to tuning and can fail if sparsity is misspecified.

Structure helps only when it is correct; stability helps regardless.

Lecture 11 summary

- When p/n is not small, sample PCs can be misleading.
- Shrinkage addresses *stability* (conditioning, eigenvalues).
- Sparse PCA addresses *interpretability* (structured loadings).
- Use shrinkage for unstable covariance; use sparse PCA for diffuse loadings.
- Compare nearby tuning choices and look for stable conclusions.

The goal is not the largest variance, but a stable and meaningful low-dimensional summary.