Bi-cross-validation of the SVD and the Nonnegative Matrix Factorization

Art Owen and Patrick Perry
presented by Linbo Wang

May 17th, 2013
CSSS 594 Class Talk

Introduction: Statistical modeling [1]

- Statistical model: $\mathbf{Y} = \mathbf{\Theta} + \mathbf{E}$
  - $\mathbf{Y}$: Observed data, potentially a matrix (e.g. subject $\times$ academic fields)
  - $\mathbf{\Theta}$: Mean model: a fixed pattern we want to recover
  - $\mathbf{E}$: Covariance Model: $E[\mathbf{E}] = 0$
- Mean model
  1. Regression model: $\mathbf{\Theta} = \mathbf{\Theta}(\mathbf{B}, \mathbf{X})$, $\mathbf{X}$ observed (given)
     - Supervised learning problem
  2. Rank/factor model: $\mathbf{\Theta} = \mathbf{\Theta}(\mathbf{B}, \mathbf{F})$, $\mathbf{F}$ latent
     - Unsupervised learning problem

[1] From Peter Hoff's notes on CSSS 594

Introduction: Statistical modeling [1]

- Statistical model: $\mathbf{Y} = \mathbf{\Theta} + \mathbf{E}$
  - $\mathbf{Y}$: Observed data, potentially a matrix (e.g. subject $\times$ academic fields)
  - $\mathbf{\Theta}$: Mean model: a fixed pattern we want to recover
  - $\mathbf{E}$: Covariance Model: $E[\mathbf{E}] = 0$
- Mean model
  1. Regression model: $\mathbf{\Theta} = \mathbf{\Theta}(\mathbf{B}, \mathbf{X})$, $\mathbf{X}$ observed (given)
     - Supervised learning problem
  2. Rank/factor model: $\mathbf{\Theta} = \mathbf{\Theta}(\mathbf{B}, \mathbf{F})$, $\mathbf{F}$ latent
     - Unsupervised learning problem

[1] From Peter Hoff's notes on CSSS 594

Example [2]

- Outcome: examination scores from each of 10 different academic fields of 1000 students
- Latent covariates: "verbal intelligence", "mathematical intelligence", "EQ", etc.
- Build a model with the latent covariates

$$\mathbf{Y} = \mathbf{BF} + \mathbf{E}$$

  - $\mathbf{Y} \in \mathbf{R}^{m \times n}, \mathbf{M} \in \mathbf{R}^{m \times n}$
  - $k = rank(\mathbf{M}) = rank(\mathbf{BF})$, $rank(\mathbf{Y}) = min(m, n)$
  - we would want $rank(\mathbf{BF}) < rank(\mathbf{Y})$
  - Mathematically: lower-rank approximation to $\mathbf{Y}$

[2] From wikipedia item on factor analysis

Example [2]

- Outcome: examination scores from each of 10 different academic fields of 1000 students
- Latent covariates: "verbal intelligence", "mathematical intelligence", "EQ", etc.
- Build a model with the latent covariates

$$\mathbf{Y} = \mathbf{BF} + \mathbf{E}$$

- $\mathbf{Y} \in \mathbf{R}^{m \times n}, \mathbf{M} \in \mathbf{R}^{m \times n}$
- $k = rank(\mathbf{M}) = rank(\mathbf{BF})$, $rank(\mathbf{Y}) = min(m, n)$
- we would want $rank(\mathbf{BF}) < rank(\mathbf{Y})$
- Mathematically: lower-rank approximation to $\mathbf{Y}$

[2] From wikipedia item on factor analysis

Estimation method: Truncated SVD

- Singular Value Decomposition (SVD):

$$\mathbf{Y} = \mathbf{U}\mathbf{D}\mathbf{V}^T = \sum_{i=1}^{n} d_i u_i v_i^T$$

  - $d_1 \geq d_2 \geq \cdots \geq d_n \geq 0$

- Eckart and Young(1936): Given $rank(\mathbf{BF}) = k$, the following truncated SVD minimized square error $\|\mathbf{Y} - \mathbf{BF}\|_F^2$

$$\hat{\mathbf{BF}} = \sum_{i=1}^{k} d_i u_i v_i^T = U_k D_k V_k^T$$

Estimation method: Truncated SVD

- $\mathbf{Y}_{m\times n} = \mathbf{U}_{m\times n}\mathbf{D}_{n\times n}\mathbf{V}^T_{n\times n} \approx \mathbf{U}1_{m\times k}\mathbf{D}1_{k\times k}\mathbf{V}1^T_{k\times k}$

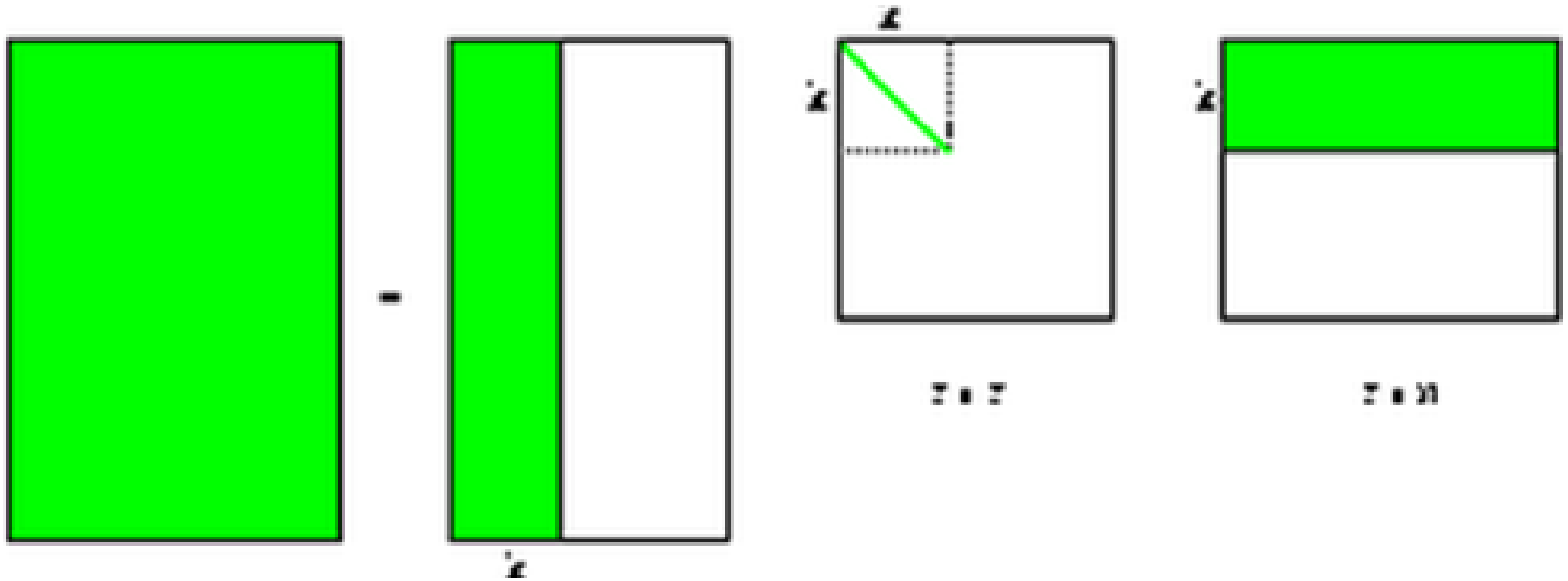

Figure: Comparison of SVD and truncated SVD [3]

- Question: How to choose k?

[3] Figure adapted from http://web.eecs.utk.edu/ berry/lsi++/node8.html

Estimation method: Truncated SVD

- $\mathbf{Y}_{m\times n} = \mathbf{U}_{m\times n}\mathbf{D}_{n\times n}\mathbf{V}^T_{n\times n} \approx \mathbf{U}1_{m\times k}\mathbf{D}1_{k\times k}\mathbf{V}1^T_{k\times k}$

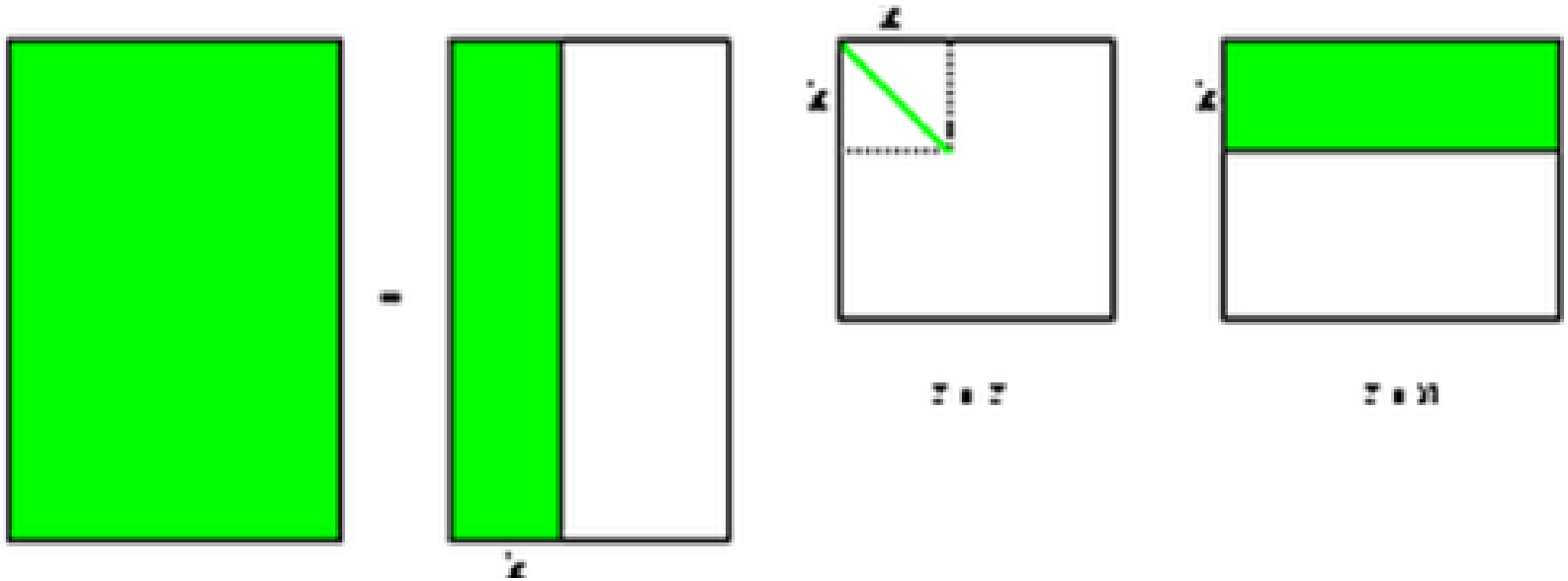

Figure: Comparison of SVD and truncated SVD [3]

- Question: How to choose k?

[3] Figure adapted from http://web.eecs.utk.edu/ berry/lsi++/node8.html

How to choose k?

- Square error $\|\mathbf{Y} - \mathbf{BF}\|_F^2$
  - Prone to overfitting
- Usual practice (Hoff(2007)): look for where the last large gap or elbow appears in a plot of singular values
  - Lack of numerical standards
- F-test(Dias and Krzanowski(2003)): not reliable here.
- Wold(1978) : add terms until the residual standard error matches the noise level - requires knowledge of noise level
- Cross validation: usual practice for supervised learning, as well as providing numerical standards
  - non-trivial under unsupervised learning settings

Cross validation under unsupervised learning

- "The" way to choose k if we know how to do it, especially for prediction purpose
- We don't know covariates as in regression/supervised learning setting, then what can we do?
  - $$\mathbf{Y}_{m\times n} = \mathbf{X}_{m\times p}\mathbf{B}_{p\times n} + \mathbf{E}_{m\times n}$$
  - 10-fold cross-validation: divide the rows of $\mathbf{Y}$ and $\mathbf{X}$ in to 10 parts, use 9 of them as training sample, and the other 1 as test sample.
  - $$(\mathbf{Y}, \mathbf{X}) = \begin{pmatrix} \mathbf{Y}_{1:r,1:n} & \mathbf{X}_{1:r,1:p} \\ \mathbf{Y}_{(r+1):m,1:n} & \mathbf{X}_{(r+1):m,1:p} \end{pmatrix}$$
  - Repeat the process for other choices of test sample.
- What can we do without knowing X?

Bi-cross-validation to select k: basic idea

- Cross-validation withhold some rows of response.

$$(\mathbf{Y}, \mathbf{X}) = \begin{pmatrix} \mathbf{Y}_{1:r,1:n} & \mathbf{X}_{1:r,1:p} \\ \mathbf{Y}_{(r+1):m,1:n} & \mathbf{X}_{(r+1):n,1:p} \end{pmatrix}$$

- Bi-cross-validation (BCV): leaves out rows and columns simultaneously.
  - $\mathbf{Y}_{m\times n} = \begin{pmatrix} Y_1 & Y_4 & Y_7 \\ Y_2 & Y_5 & Y_8 \\ Y_3 & Y_6 & Y_9 \end{pmatrix} = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$
  - Try to predict A (withheld part) with B, C, D (observed part)
    - For each left-out portion $Y_i, i = 1, \cdots, 9$, for each k, define $BCV_i(k)$
  - Repeat this process for $Y_1, Y_2, \cdots, Y_9$, and take the average: $BCV(k) = \frac{1}{9}\sum_{i=1}^{9} BCV_i(k)$.
    - Select k that minimize $BCV(k)$

Sit back, relax, and think...

- $\mathbf{Y}_{m\times n} = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$
  - Try to predict A (withheld part) with B, C, D (observed part)
- Go back to regression setting (Principal component regression)

$$(\mathbf{Y}, \mathbf{X}) = \begin{pmatrix} \mathbf{Y}_{1:r,1:n} & \mathbf{X}_{1:r,1:p} \\ \mathbf{Y}_{(r+1):m,1:n} & \mathbf{X}_{(r+1):n,1:p} \end{pmatrix}$$

  - $\hat{\mathbf{Y}}_{1:r,1:n} = \mathbf{X}_{1:r,1:p}(\mathbf{X}_{(r+1):n,1:p}{}^{(k)})^{-}\mathbf{Y}_{(r+1):m,1:n}$
    - Math fact: $X^{-}Y = (X^{T}X)^{-1}X^{T}Y$
  - Get the prediction error in Frobenius norm: $\|\hat{\mathbf{Y}}_{1:r,1:n} - \mathbf{Y}_{1:r,1:n}\|_F^2$

Method: It is straightforward!

- $\mathbf{Y}_{m\times n} = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$
  - $\hat{\mathbf{A}} = \mathbf{C}(\mathbf{D}^{(k)})^{-}\mathbf{B}$
  - Get the prediction error in Frobenius norm: $\|\hat{\mathbf{A}} - \mathbf{A}\|_F^2$
- Turns out that it gives reasonable U-shape error curve in practice.
- There are also some theoretical properties for this estimator proved in this paper.
- Historical note: first proposed by Gabriel (2002) in 1×1 case.

Wait a minute..

- My concerns
  - We are using best rank approximation to D as best rank approximation to Y?
    - underestimating?
    - will never recover the truth if the best rank is larger than $\frac{2}{3} * min(m, n)$
  - We call rows of Y "subjects", columns of Y "response" (in both the motivating example and the prediction procedure)
    - assume rows are independent, while columns are correlated?
    - Y could have been transposable! (Example: trade data from one country to another)

Other methods exists..

- Eastment and Krzanowski (1982)
- Best known method up to date (cited by 237 up to date)
- Critiques
  - cross-validation errors decrease monotonically with k
    - some awkward adjustments based on estimated degree of freedom are used in practice.
  - sign for singular vectors $\mathbf{u}_i, \mathbf{v}_i^T$ is not determined
  - Linbo: lack of theoretical justification

Properties of our BCV estimator

- Theoretical properties ("Model Selection Consistency"):
  - Self-consistency property
  - Pure Gaussian noise (true k=0)
    - Asymptotically: $E[BCV(1)] > E[BCV(0)]$
  - Rank 1 plus Gaussian noise (true k=1)
    - Asymptotically: $E[BCV(1)] < E[BCV(0)]$ under some conditions
- Empirical properties
  - Cross-validation error is U-shape with respect to k

Model Selection Consistency

- Self-consistency property
  - $\mathbf{Y} = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$
  - Conditions: rank(Y)=rank(D)=k
  - Conclusion: $A - B(\hat{D}^k)^- C = A - BD^- C = 0$
- Eastment and Krzanowski (1982) generally doesn't have such property

Model Selection Consistency

- True rank k=0: $Y_{m\times n} = 0_{m\times n} + Z_{m\times n}$
  - $Z_{ij} \overset{\text{iid}}{\sim} N(0,1)$
  - $c \approx m/n$: size of matrix
  - hold out proportion is constant: $r/m = s/n = \theta$
- True rank k=1: $Y_{m\times n} = \kappa u_{m\times 1} {v_{n\times 1}}^T + Z_{m\times n}$
  - $u_{m\times 1}$ and $v_{n\times 1}$ are unit vectors
  - root mean square of noise: $(E[Z_{ij}^2])^{1/2} = 1$
  - root mean square of signal: $(E(\kappa u v^T)^2/mn)^{1/2} = \kappa\sqrt{mn}$
  - Assume $\kappa^2 = \delta\sqrt{mn}, \delta > 1$ represents the strength of signal.

Choice of hold out portion

- Smaller aspect ratio ($c \approx m/n$ closer to 1) is advantageous.
- Larger hold out portion θ will favor selection of lower rank.
  - Small holdouts is more prone to overfitting.
  - Large holdouts is more prone to underfitting.
- In practice, the authors recommend a $(2 \times 2) - \textit{fold}$ or $(3 \times 3) - \textit{fold}$ BCV.

Simulation!

- Data Generation
  - $\mathbf{Y} = \mathbf{M} + \mathbf{E}$
  - Generate $\mathbf{M}$ to have pre-specified singular values $\tau_1 \geq \tau_2 \geq \cdots \geq \tau_{min(m,n)}$
  - Two patterns for singular values
    1. Binary pattern: $\boldsymbol{\tau} \propto (1, 1, \cdots, 1, 0, \cdots, 0)$
    2. Geometric pattern: $\boldsymbol{\tau} \propto (1, 1/2, 1/4, \cdots, 1/2^{min(m,n)})$
- $k^{opt} = argmin_k \| \hat{Y}^{(k)} - M \|^2$
- Small/large simulation set-up
  1. small: m=50 and n=40
  2. large: m=n=1000

Simulation - methods

1. Our BCV / Gabriel method
2. Eastment-Krzanowski
3. Bai and Ng (2002)'s BIC method

In these, the estimate $\hat{k}$ is the minimizer of

$$\text{(7.1)} \qquad \mathrm{BIC}_1(k) = \log(\|\widehat{X}^{(k)} - X\|^2) + k\frac{m+n}{mn}\log\frac{mn}{m+n},$$

$$\text{(7.2)} \qquad \mathrm{BIC}_2(k) = \log(\|\widehat{X}^{(k)} - X\|^2) + k\frac{m+n}{mn}\log C^2 \quad \text{or}$$

$$\text{(7.3)} \qquad \mathrm{BIC}_3(k) = \log(\|\widehat{X}^{(k)} - X\|^2) + k\frac{\log C^2}{C^2},$$

over k, where $c = c(m,n) = \min(\sqrt{m}, \sqrt{n})$.

Shape of curve

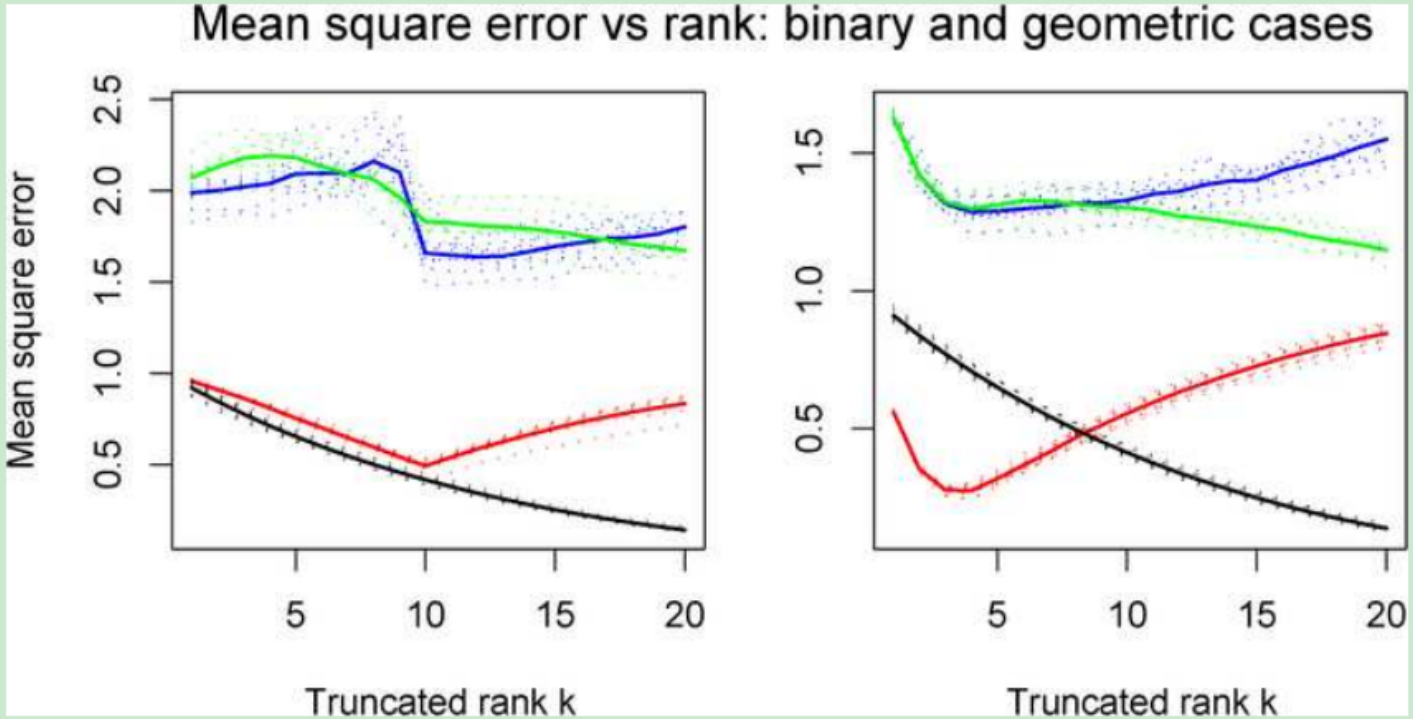

FIG. 2. *This figure shows the results of Gabriel and Eastment–Krzanowski* 1×1 *hold out cross-validation on some 50 by 40 matrix examples described in the text. In the left panel the signal matrix has 10 positive and equal singular values and 30 zero singular values. The dotted red curves show the true mean square error per matrix element* $\|\widehat{X}^{(k)} - \mu\|^2/(mn)$ *from 10 realizations. The solid red curve is their average. Similarly, the black curves show the naive error* $\|\widehat{X}^{(k)} - X\|^2/(mn)$*. The green curves show the results from Eastment–Krzanowski style cross-validation. The blue curves show Gabriel style cross-validation. The right panel shows a similar simulation for singular values that decay geometrically. In both cases the mean square signal was equal to the expected mean square noise.*

"Small" sample performace

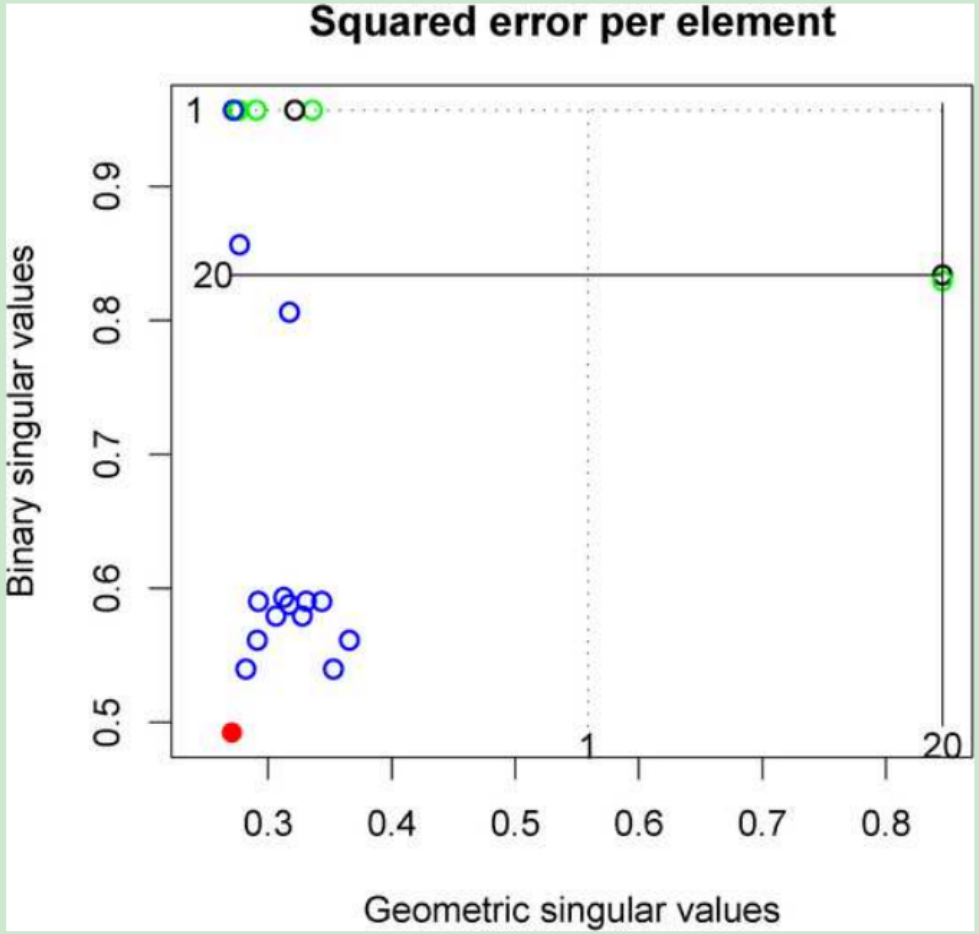

FIG. 3. *This figure shows the mean square error, per element, for all the methods applied to the* 50×40 *example. The case with geometrically decaying singular values is on the horizontal axis, and the binary case with ten equal nonzero singular values is on the vertical axis. Gabriel's method and our generalizations are shown in blue, (generalized) Eastment–Krzanowski is in green, the oracle is red, and Bai and Ng's BIC estimators are in black. There are horizontal and vertical reference lines for methods that always pick* $k = 1$ *or* $k = 20$*. The cluster of blue points in the lower left corner is discussed in the text.*

"Large" sample performace

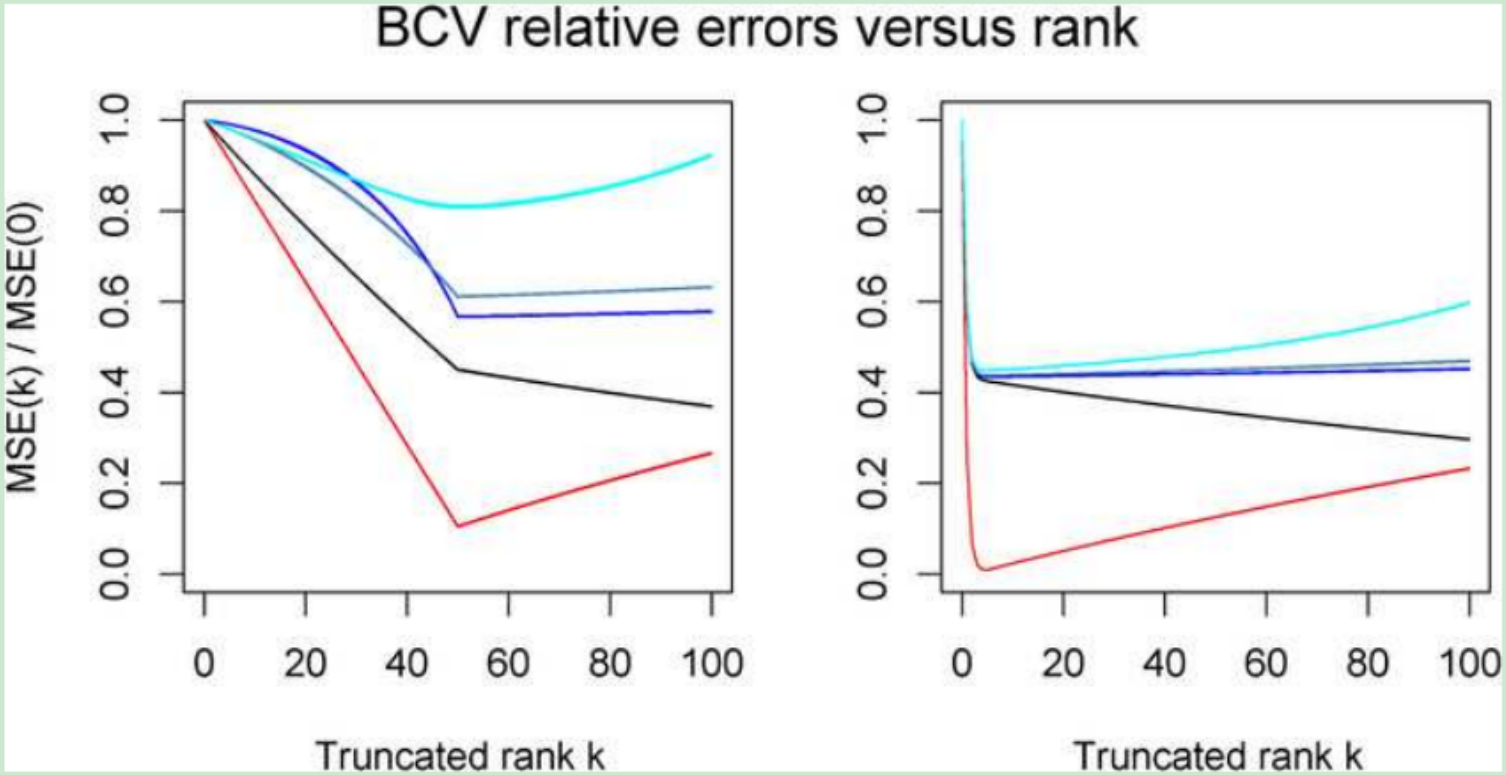

FIG. 4. *This figure shows the BCV errors for the* 1000×1000 *examples with equal signal and noise magnitude, as described in the text. The left panel shows the results when there are 50 positive and equal singular values. The horizontal axis is fitted rank, ranging from 0 to 100. The vertical axis depicts square errors, in each case relative to the squared error for rank* $k = 0$*. There are 10 nearly identical red curves showing the true error in 10 realizations. The black curves show the naive error. Blue curves from dark to light show BCV holdouts of* 200×200*,* 500×500 *and* 800×800*. The right panel shows the results for a geometric pattern in the singular values.*

Summary

- Factor model is a popular alternative to regression model.
- Lower rank approximation to the observed data can be obtained via truncated SVD.
- Current practice of choice of k is arbitrary.
- Problem: cross-validation for k in unsupervised learning.
- Approach: mimic cross-validation for principal component regression, the authors come up with a bi-cross-validation(BCV) method.
- Some theoretical justifications for BCV are presented, and simulation studies show that the estimator is good overall.
- More can be done above this!

Questions?