How to bootstrap the Netflix data
and cross-validate the
non-negative matrix factorization

Art B. Owen

Department of Statistics
Stanford University
### Statistics as usual

<table>
<thead>
<tr>
<th>Variable 1</th>
<th>⋮</th>
<th>Variable C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>⋮</td>
<td></td>
</tr>
<tr>
<td>Case R</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1) Variables are named entities:
   - E.g. pressure, volume, income ⋮
   - They persist

2) Cases are anonymous replicates
   - Sampled IID from some $F$
   - Of no inherent interest

**Under statistic as usual ⋮ ⋮**

... we only care about cases because they show relationships among variables.
### Variables by variables

<table>
<thead>
<tr>
<th>Rating</th>
<th>Viewer 1</th>
<th>Viewer 2</th>
<th>Viewer 3</th>
<th>⋮</th>
<th>Viewer C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Movie 1</td>
<td>4</td>
<td>4</td>
<td>1</td>
<td>⋮</td>
<td>4</td>
</tr>
<tr>
<td>Movie 2</td>
<td>5</td>
<td>5</td>
<td>NA</td>
<td>⋮</td>
<td>NA</td>
</tr>
<tr>
<td>Movie 3</td>
<td>3</td>
<td>3</td>
<td>NA</td>
<td>⋮</td>
<td>2</td>
</tr>
<tr>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td></td>
<td>⋮</td>
</tr>
<tr>
<td>Movie R</td>
<td>NA</td>
<td>5</td>
<td>3</td>
<td>⋮</td>
<td>4</td>
</tr>
</tbody>
</table>

Sometimes specific rows and columns are both of persistent interest:

- IPs $\times$ books $\rightarrow$ purchases
- terms $\times$ documents $\rightarrow$ counts
- candidate $\times$ interviewer $\rightarrow$ rating
- nodes $\times$ more nodes $\rightarrow$ labeled edges
### Triples

<table>
<thead>
<tr>
<th></th>
<th>Movie</th>
<th>Viewer</th>
<th>Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case 1</td>
<td>1</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>Case 2</td>
<td>1</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>Case 3</td>
<td>2</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>Case N</td>
<td>R</td>
<td>C</td>
<td>4</td>
</tr>
</tbody>
</table>

- Now cases are anonymous
- We don’t store the NAs
- 2 categorical variables with lots of levels
- Not independent:
  - Cases 1 & 2 share a movie
  - Cases 1 & 3 share a viewer

How should we bootstrap and cross-validate data like this?

Should we resample cases? leave out cases?
Sample reuse as usual

**Cross validation**

1) Pairs \((X_i, Y_i)\) are IID from \(F\)
2) Leave out some pairs
3) Fit \(\hat{Y} = f(X)\) from retained pairs
4) Predict held out \(Y\)'s

**Bootstrap**

1) Data \(X_i\) are IID from \(F\) (unknown)
2) Estimate \(F\) by \(\hat{F}\) (known)
3) Sample \(X_i^*\) from \(\hat{F}\)
4) \(X_i\) are to \(F\) as \(X_i^*\) are to \(\hat{F}\)

... in a nutshell
What we’d like

**Bootstrap**

1) For complicated methods (e.g. spectral bi-clustering) we would like to
   a) resample the data a few times,
   b) refit the model,
   c) see what is stable

2) For simple things (e.g. do Harry Potter readers like Mozart?) we want to know if small
effects are real

**Cross-validation**

1) We want to leave out some known data and predict them to simulate filling gaps in the matrix

2) We want to avoid leaving out one whole row
# The answer

<table>
<thead>
<tr>
<th></th>
<th>Bootstrap</th>
<th>Cross-validation</th>
</tr>
</thead>
<tbody>
<tr>
<td>By Elements</td>
<td>Wrong</td>
<td>OK (awkward)</td>
</tr>
<tr>
<td>By Rows and Cols</td>
<td>OK (approx)</td>
<td>Good</td>
</tr>
</tbody>
</table>
Random effects model

\[ X_{ij} = \mu + a_i + b_j + \varepsilon_{ij} \quad i = 1, \ldots, R \quad j = 1, \ldots, C \]

\[ a_i \sim N(0, \sigma_A^2) \quad \text{e.g. plants} \]

\[ b_j \sim N(0, \sigma_B^2) \quad \text{e.g. environments} \]

\[ \varepsilon_{ij} \sim N(0, \sigma_E^2) \]

Used in agriculture

Studied for decades

\( \hat{\mu} \) is \( \bar{X}_{..} \)

No bootstrap exists for \( V(\hat{\mu}) \)

None can exist \( \cdots \)

\( \cdots \) McCullagh (2000)

We can’t even bootstrap a balanced \( \bar{X} \)!
Bi-bootstrap and bi-cross-validation

McCullagh (2000)

For \( \hat{\mu} = \bar{X}_{..} = \frac{1}{RC} \sum_{i=1}^{R} \sum_{j=1}^{C} X_{ij} \)

Boot-I  Resample from \( N = RC \) values

Boot-II  Resample \( R \) rows indep of \( C \) columns

\[ V(\hat{\mu}) = \frac{\sigma_A^2}{R} + \frac{\sigma_B^2}{C} + \frac{\sigma_E^2}{RC} \]

true var

\[ E(\hat{V}_I(\hat{\mu})) = \left( \sigma_A^2 + \sigma_B^2 + \sigma_E^2 \right) \frac{1}{RC} \]

way too small

\[ E(\hat{V}_{II}(\hat{\mu})) = \frac{\sigma_A^2}{R} + \frac{\sigma_B^2}{C} + \frac{3\sigma_E^2}{RC} \]

close

Boot-I is seriously flawed, Boot-II is close
Generalization (O 2007)

Recall

\[ i = 1, \ldots R \quad \text{rows} \]
\[ j = 1, \ldots C \quad \text{columns} \]
\[ X_{ij} = \mu + a_i + b_j + \varepsilon_{ij} \]

Now allow missing values

\[ Z_{ij} = \begin{cases} 1 & \text{if } ij \text{ observed} \\ 0 & \text{else} \end{cases} \]

Nonnormal data

\[ E(a_i) = E(b_j) = E(\varepsilon_{ij}) = 0 \]
\[ V(a_i) = \sigma_A^2, \quad V(b_j) = \sigma_B^2, \quad V(\varepsilon_{ij}) = \sigma_E^2 \]
Sample size quantities

\[ n_{i.} = \sum_{j=1}^{C} Z_{i,j} \quad n_{.j} = \sum_{i=1}^{R} Z_{i,j} \]

\[ \nu_A = \frac{1}{N} \sum_{i=1}^{R} n_{i.}^2 \quad \nu_B = \frac{1}{N} \sum_{j=1}^{C} n_{.j}^2 \]

\[
V(\hat{\mu}_x) = \sigma_A^2 \frac{\nu_A}{N} + \sigma_B^2 \frac{\nu_B}{N} + \sigma_E^2 \frac{1}{N}
\]

\[
E(\hat{V}_I(\hat{\mu})) = \left( \sigma_A^2 + \sigma_B^2 + \sigma_E^2 \right) \frac{1}{N}
\]

\[
E(\hat{V}_{II}(\hat{\mu})) = \sigma_A^2 \frac{\nu_A}{N} + \sigma_B^2 \frac{\nu_B}{N} + \sigma_E^2 \frac{3}{N}
\]

Typically \( 1 \ll \nu \ll N \)

Netflix: \( \nu_{\text{Movies}} \approx 56,200 \quad \nu_{\text{Cust}} \approx 646 \quad N \approx 100,000,000 \)
Generalizations

Non-constant variance: let

\[ V(a_i) = \sigma^2_{A(i)} \quad V(b_j) = \sigma^2_{B(j)} \quad V(\varepsilon_{ij}) = \sigma^2_{E(i,j)} \]

uniformly bounded away from 0 and \( \infty \)

Then as \( N \to \infty \)

\[ \frac{E(\hat{V}_{II}(\hat{\mu}_x)) - V(\hat{\mu}_x)}{V(\hat{\mu}_x)} = O(\epsilon_N) \]

Where

\[ \epsilon_N = \max \left( \frac{1}{R}, \frac{1}{C}, \frac{1}{\nu_A}, \frac{1}{\nu_B}, \max_{i} \frac{n_{i \cdot}}{N}, \max_{j} \frac{n_{\cdot j}}{N} \right) \]

The 3 remains but gets swamped
Netflix data

- $N = 100,480,507$ ratings, $17,770$ movies, $480,189$ customers

- It would be fun to look for small effects linked to customer demographics

- But those are not available for privacy reasons

- So I look at the day of the week effect

- Ratings made on Tuesdays are a tad low (average 3.596)

<table>
<thead>
<tr>
<th>Advantage over Tuesday</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mon</td>
</tr>
<tr>
<td>0.002</td>
</tr>
</tbody>
</table>

A very small effect. Is it real? Maybe. The sample size is large.
10 bootstraps

- Sunday vs Tuesday
- 10 resamplings (open)
- Real data (solid)
- Small but real: $p < 2 \times 10^{-5}$.

Maybe:

a) Worse movies rated on Tuesday
b) or each movie does worse on Tuesday
c) or tougher customers
d) or each customer tougher on Tuesday

Place your bets!
10 bootstraps

- Sunday vs Tuesday
- 10 resamplings (open)
- Real data (solid)
- Small but real: $p < 2 \times 10^{-5}$.

Maybe:

a) Worse movies rated on Tuesday
b) or each movie does worse on Tuesday
c) or tougher customers
d) or each customer tougher on Tuesday

All of the above
Some details

\( Z_{ij} = 1 \iff \text{Cust i rates movie j} \)

\( Y_{ij} = \text{Rating} \in \{1, 2, 3, 4, 5\} \)

\( D^{\text{Tue}}_{ij} = 1 \iff \text{Rating on Tuesday} \)

\[
\hat{\mu}_{\text{Tue}} = \frac{\sum_{ij} Z_{ij} D^{\text{Tue}}_{ij} Y_{ij}}{\sum_{ij} Z_{ij} D^{\text{Tue}}_{ij}}
\]

\[
\mu_{\text{Tue}} = \frac{\sum_{ij} \mathbb{E}(Z_{ij} D^{\text{Tue}}_{ij} Y_{ij})}{\sum_{ij} \mathbb{E}(Z_{ij} D^{\text{Tue}}_{ij})}
\]

\[
\hat{\theta} = \hat{\mu}_{\text{Tue}} - \hat{\mu}_{\text{Sun}}
\]

\[
\theta = \mu_{\text{Tue}} - \mu_{\text{Sun}}
\]

Use Taylor expansions (delta method)
Modelling $Z_{ij}$

- We do not model the missingness
- Analysis is conditional on $Z_{ij}$
- No need to resample unobserved $Y_{ij}$'s

Can/should we do that?

- Missingness is very important
- Less so if you’re predicting ratings that were actually made
- Modelling $Z_{ij}$ requires untestable assumptions (from outside the data)
The data

Movie popularity vs number of ratings

Customer's mean rating vs number of ratings
Sunday vs Tuesday again

- Solid curve:
  - Cust avg score on $x$ axis
  - Cust Sun–Tue on vertical axis
  - Get 100,000s of points
  - Smooth the points

- Dashed curve:
  - Same for movies
Bootstrap conclusion

1) Don’t resample matrix entries

2) Resample row and col entities independently
Outer product models

Data are

\[ X_{ij}, \quad \text{for} \quad i = 1, \ldots, m \quad \text{and} \quad j = 1, \ldots, n \]

Usual ANOVA

\[ X_{ij} = \mu + \alpha_i + \beta_j + \varepsilon_{ij} \]

Fisher & MacKenzie, 1923

\[ X_{ij} = \mu + \alpha_i + \beta_j + du_i v_j + \varepsilon_{ij} \]

Focus on multiplicative part

\[ X_{ij} \doteq du_i v_j \]

Add more factors

\[ X_{ij} \doteq \sum_{\ell=1}^{k} d_{\ell} u_{i\ell} v_{j\ell} \]
Outer product models

Expressions:

\[ X_{ij} = \sum_{\ell=1}^{k} \sigma_{\ell} u_{i\ell} v_{i\ell} \]

\[ X = \sum_{\ell=1}^{k} \sigma_{\ell} u_{\ell} v_{\ell}^{\prime} \quad u_{\ell} \in \mathbb{R}^m, \quad v_{\ell} \in \mathbb{R}^n \]

\[ X = U \Sigma V^{\prime} \]

\[ X = LR \quad L \in \mathbb{R}^{m \times k}, \quad R \in \mathbb{R}^{k \times n} \]

Examples:

1) Factor analysis
2) Principal components
3) Singular value decomp (SVD)
4) Nonnegative matrix decomp
5) Semi-discrete decomp

Problem: how to pick \( k \)?
Picking $k$

Novartis: $n = 101$ tissues $m = 12600$ genes, expression

- Bigger $k$: smaller error (might overfit)
- Not always an obvious gap
Classical solution

1) Fit SVD with $k$ and with $k + 1$
2) Find sum squares $SS(k)$
3) Count parameters $d(k)$
4) Prefer $k + 1$ if $F = \frac{SS(k) - SS(k+1)}{d(k+1) - d(k)}$ is large

Problems

1) No good way to count df
2) ... and it doesn’t work  
   dos Dias & Krzanowski 2003
   i) get 5% vs 66% for $k = 0$ vs $k = 1$
   ii) then too conservative for larger $k$
Cross validating the SVD

Hold out one value, say $X_{11}$

Eastment & Krzanowski 1982
1) Do SVD leaving out row 1 . . . take $v$'s
2) Do SVD leaving out col 1 . . . take $u$'s
3) Pool the $\sigma$'s (geometric mean)
4) Reassemble
   Still get bigger $k \implies$ better fit

Gabriel 2003
1) $X = \begin{pmatrix} X_{11} & X_{12:n} \\ X_{2:m 1} & X_{2:m 2:n} \end{pmatrix}$
2) Fit $X_{2:m 2:n} \doteq U \Sigma_{1:k} V'$
3) $\hat{X}_{11} = X_{12:n} (V \Sigma_{1:k}^+ U') X_{2:m,1}$
4) Seems to work (in crop science)

Why does Gabriel’s expression work?
Generalize to $r \times s$ blocks

$$X = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$$

- Take $X \in \mathbb{R}^{m \times n}$
- Leave out $A \in \mathbb{R}^{r \times s}$
- Fit SVD to $D \in \mathbb{R}^{m-r \times n-s}$
- Cut at $k$ terms $\hat{D}^{(k)}$
- $\hat{A} = B(\hat{D}^{(k)}) + C$ (pseudo $BD^{-1}C$)
- Repeat for $(m/r) \times (n/s)$ blocks
- Sum squared err $\|\hat{A} - A\|^2$

Puzzler: are we leaving out $A$ or $A$, $B$, and $C$?
Suppose that \( X = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \) has rank \( k \) and so does \( D \). Then

\[
A = BD^+ C = B(\hat{D}(k))^+ C
\]

where \( D^+ \) is the Moore-Penrose generalized inverse.

This justifies treating \( A - B(\hat{D}(k))^+ C \) as a residual from rank \( k \).
Idea of proof

\[ X = \begin{pmatrix} A & B \\ C & D \end{pmatrix} = U \Sigma V' = \begin{pmatrix} U_1 \Sigma V_1' & U_1 \Sigma V_2' \\ U_2 \Sigma V_2' & U_2 \Sigma V_2' \end{pmatrix} \]

- It would be immediate if \( D = U_2 \Sigma V_2' \) were an SVD.
- But \( U_2 \) and \( V_2 \) are not orthogonal.
- We still get \( \cdots D^+ = V_2 (V'_2 V_2)^{-1} \Sigma^{-1} (U'_2 U_2)^{-1} U'_2 \)
- Then \( U_1 \Sigma V_2' \left( V_2 (V'_2 V_2)^{-1} \Sigma^{-1} (U'_2 U_2)^{-1} U'_2 \right) U_2 \Sigma V_1' = U_1 \Sigma V_1' = A \)
Exceptions: $\text{rank}(D) < \text{rank}(X)$

**Spike**

$$X = \begin{pmatrix}
1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix}$$

$A = 1$ but $\hat{A} = 0$

**Stripe**

$$X = \begin{pmatrix}
1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix}$$

$A = 1$ but $\hat{A} = 0$

**Arrow**

$$X = \begin{pmatrix}
1 & 1 & 1 & 1 \\
1 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 \\
1 & 0 & 0 & 0
\end{pmatrix}$$

$A = 1$ but $\hat{A} = 0$
Upshot

- Any feature with few rows or columns than we hold out could be lost
- This could bring robustness versus noise
- Or make us miss sparse features
- Almost the same thing.

Fix (if you want it)

\[ \tilde{X} = \mathcal{O}_L X \mathcal{O}_R \]

\( \mathcal{O}_L \in \mathbb{R}^{m \times m} \) random orthogonal

\( \mathcal{O}_R \in \mathbb{R}^{n \times n} \) random orthogonal

\( X \) and \( \tilde{X} \) have same singular values
Novartis results

- **Solid** = cross-validated squared error
- **Open** = naive squared error
Cross validating the NNMF

Lee & Seung (1999)

\[ X \doteq W H \]
\[ W \in [0, \infty)^{m \times k} \]
\[ H \in [0, \infty)^{k \times n} \]

- Non-negative entries.
- Often sparse.
- Interpretation as parts.

Let \( D \doteq W_D H_D \)

Then \( \hat{A} = B(W_D H_D)^+ C \)

We don’t always get

\[ (Y Z)^+ = Z^+ Y^+ . \]

But by MacDuffee’s theorem

\[ (W_D H_D)^+ = H_D^+ W_D^+ \]

when both have rank \( k \).
NNMF Example

Binary factor model

\[ X = WH + E \]

- \( W \in \{0, 1\}^{100 \times 3} \) \text{ half 1s} \\
- \( H \in \{0, 1\}^{3 \times 50} \) \text{ half 1s} \\
- \( E \in \{0, 1\}^{m \times n} \) \text{ 1/4th 1s} \\

For 100 data sets used cross-validation to pick \( k \)
Results

Bi cross validated squared errors

<table>
<thead>
<tr>
<th>$k$</th>
<th>$\ell$</th>
<th>$\hat{r} = 3$</th>
<th>$\hat{r} = 4$</th>
<th>$\hat{r} = 5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>10</td>
<td>82</td>
<td>18</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
<td>88</td>
<td>12</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>95</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td>85</td>
<td>14</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>87</td>
<td>12</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>94</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>94</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>97</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>99</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

(2,2) bi-cross-validation
(10,10) bi-cross-validation
Random matrix theory

Let $CV(k)$ be cross validated error using $k$ factors

Let $E_k$ denote expectation assuming $k$ factors plus Gaussian noise

For true rank $= 0$

- $E_0(CV(1)) > E_0(CV(0))$ but just barely . . .
- $\lim E_0(CV(1))/E_0(CV(0)) = 1$
- Helps to have big $r$ and $s$ (even $r \approx m$ and $s \approx n$)

For true rank $= 1$

- $E_1(CV(1)) < E_1(CV(0))$ by large margin
- Hurts to have big $r$ and $s$
Other outer product models

Semidiscrete decomposition

$$X \doteq U\Sigma V' \quad U \in \{-1, 0, 1\}^{m \times k} \quad V \in \{-1, 0, 1\}^{k \times n}$$

$k$: means

$$X \doteq U\Sigma V' \quad U \in \{0, 1\}^{m \times k} \quad V \in \mathbb{R}^{k \times n}$$

Plaid

$$X_{ij} = \sum_\ell \rho_{i\ell} \kappa_{j\ell} (\mu_\ell + \alpha_{i\ell} + \beta_{j\ell}) \quad \rho_{i\ell}, \kappa_{j\ell} \in \{0, 1\}$$

also can be cross-validated
Cross validation conclusion

1) We can leave out blocks $r \times s$

2) Leaving out $1 \times 1$ may be slow but is not incorrect.
   So leaving out scattered point sets should be ok.
   But it requires missing data methods.
Acknowledgments

1) The cross-validation work is joint with Patrick O. Perry, of Stanford’s department of Statistics.
2) This talk was first given in seminars at Google and at Allstate Research in September of 2007. Many thanks to Samy Bengio and Sam Roweis (of Google) and Paul Louisell (Allstate) for hosting.
3) I’d like to thank Netflix for making such a rich data set available. It’s hard to get data from corporations out to researchers, and they’ve done a great job.