

# LATENT VARIABLES IN STATISTICS AND THEIR APPLICATION TO A PROBLEM OF RATING ESTIMATION

## Experimental study

A.M. Andronov, A. A.Ressin and K. Pilyushonoka

30 мая 2014 г.

## Data volume

Total in Netflix Prize(1) Dataset:

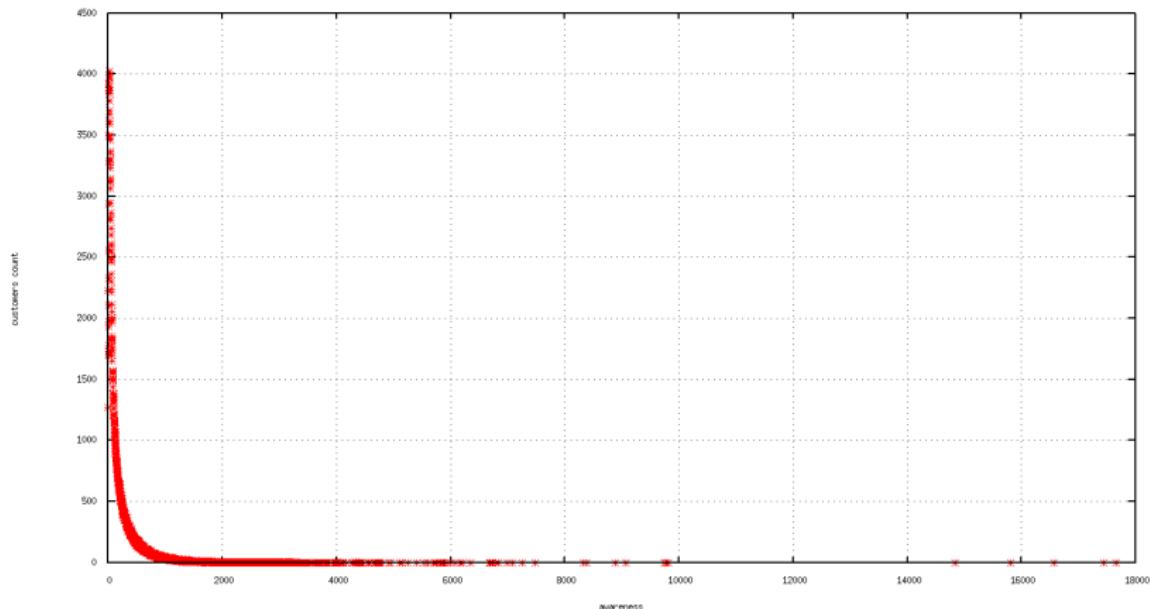
- movies – 17,770;
- customers – 480,189;
- cross-scores – 100,480,507.

# Data volume analysis

- Histogram of customer awareness of movies
  - X – user awareness: count of different movies watched by a single customer;
  - Y – count of users with given awareness.
- Histogram of movies popularities
  - X – movie popularity: count of different customers that watched a single movie;
  - Y – count of movies with given popularity.

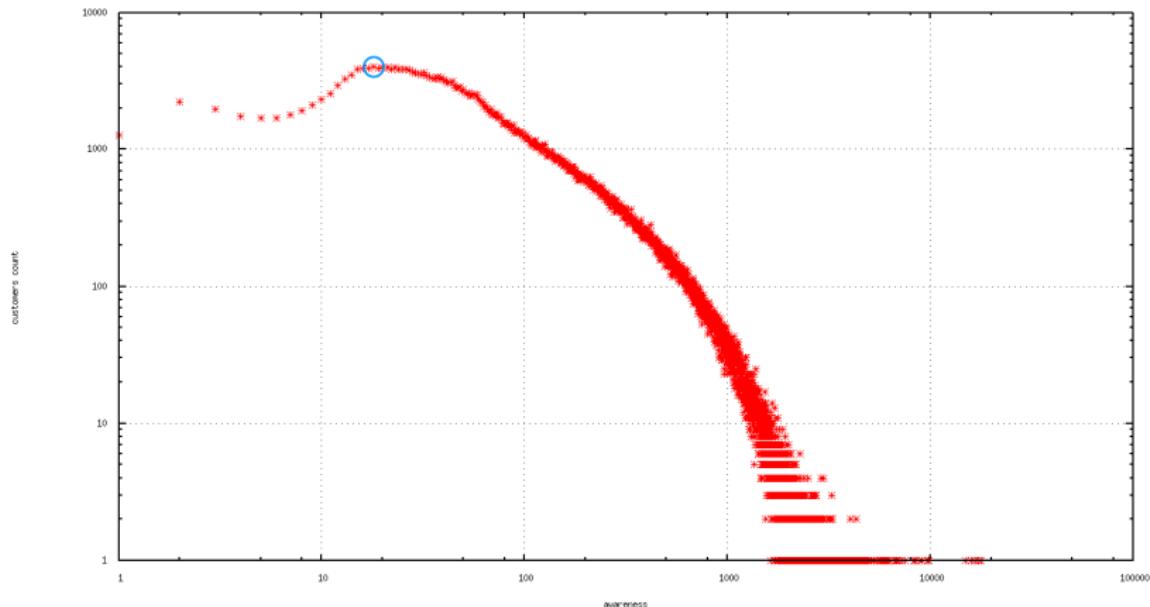
# Data volume analysis

## Users awareness



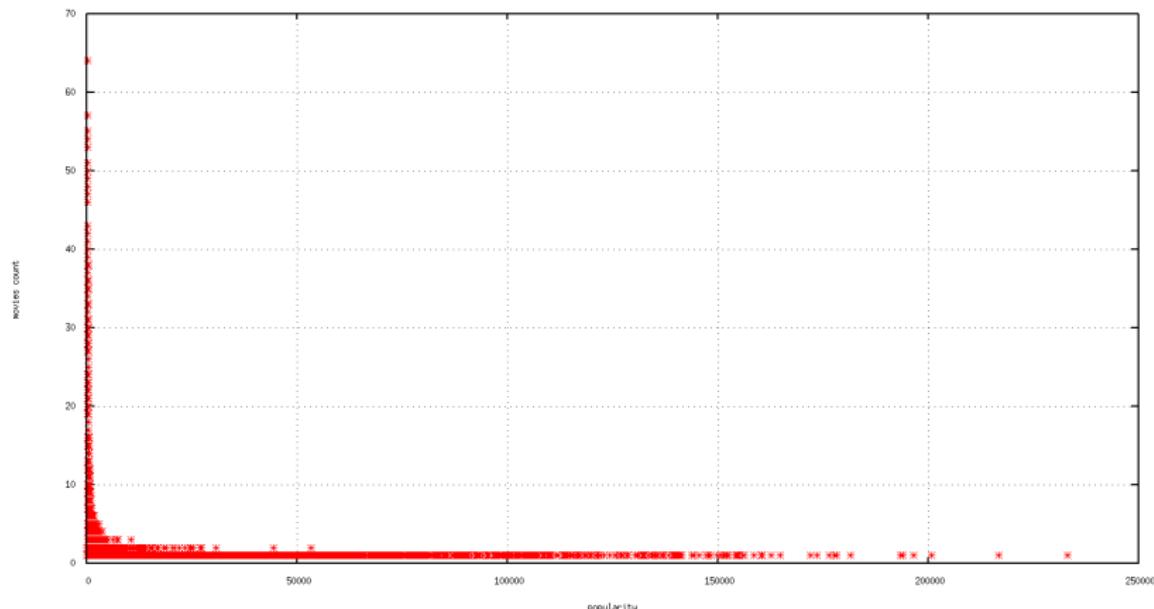
# Data volume analysis

## Users awareness (log-scale)



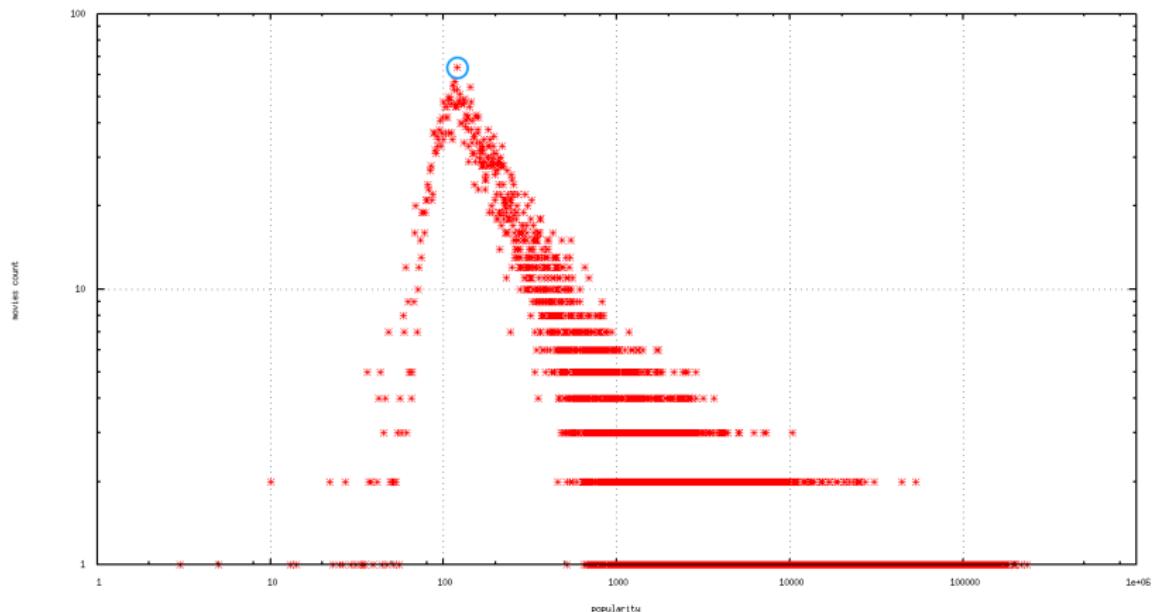
# Data volume analysis

## Movies popularity



# Data volume analysis

## Movies popularity (log-scale)



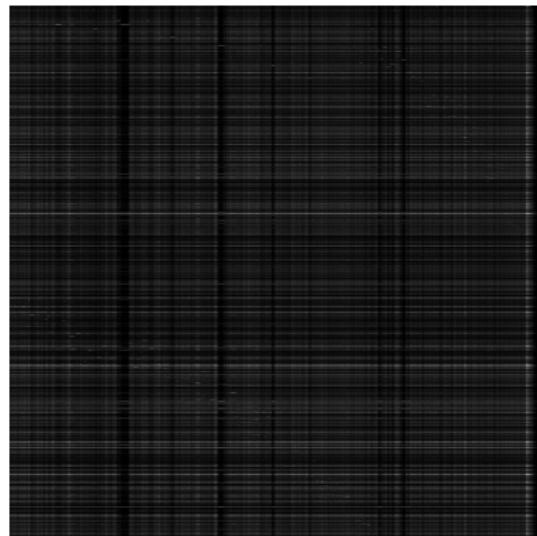
# Data volume analysis

## Summary

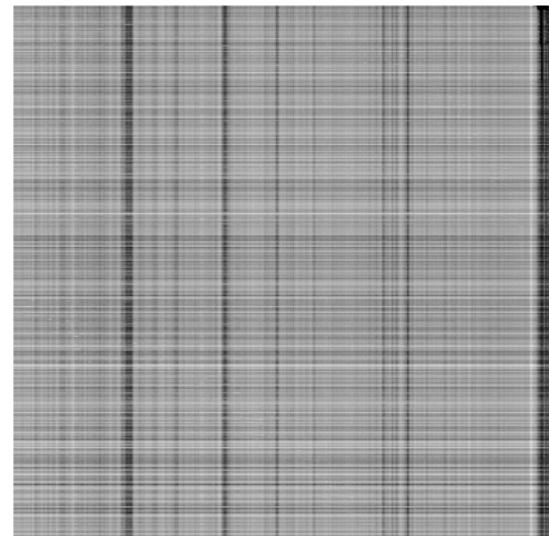
- Awareness:
  - the most frequent (4014 cases) awareness for user is 18;
  - average – 209.25;
  - standard deviation – 302.33.
- Popularity:
  - the most frequent (64 cases) popularity for movie is 120;
  - average – 5654.50;
  - standard deviation – 16909.67.
- Score fill ratio: 1.18%.

# Data availability

Sparse matrix visual representation: grayscale



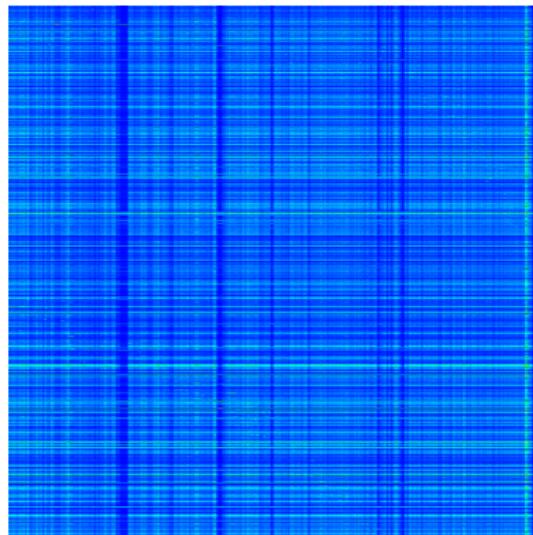
usual



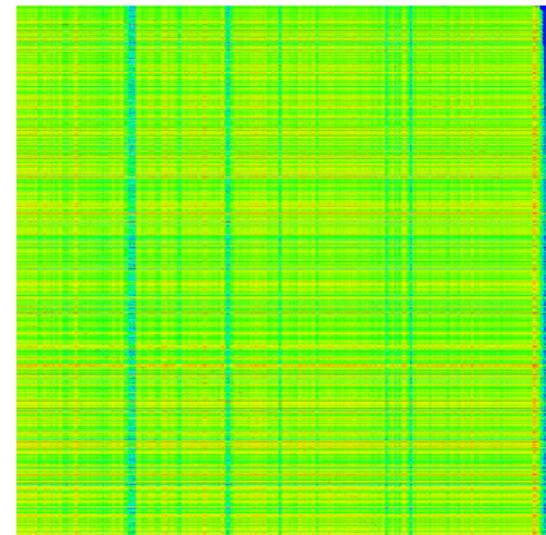
log-scale

# Data presence

Sparse matrix visual representation: rainbow



usual



log-scale

## Solid submatrix of given sparse matrix

- Let  $I$  and  $J$  are set of customers and objects indices respectively. The task is to find such  $I^* \subseteq I$  and  $J^* \subseteq J$ , that  $\forall i \in I^* \forall j \in J^* \exists y_{ij}$ , where  $y_{ij}$  is score of object  $j$  given by user  $i$  and  $|I^*||J^*| \rightarrow \max$ .
- The problem is NP-complete, because it is form of famous Clique Problem.
- Need for problem relaxation: maximize submatrix density for given submatrix size  $|I^*| \times |J^*|$ :

$$|I^*||J^*| - |\{y_{ij} : i \in I^*, j \in J^*, \exists y_{ij}\}| \rightarrow \min.$$

- The problem still is NP-complete, but now it has heuristical solution.

# Heuristical solution

- Let  $\mathbf{Q} = \{q_{ij}\}_{n \times m}$  – binary matrix representing data availability.
- Let introduce  $\phi_i$  – authority factors of users, and  $\tau_j$  – authority factors of movies, such that

$$\phi_i = \frac{\sum_{j=1}^m \tau_j q_{ij}}{\sum_{j=1}^m q_{ij}}, \text{ and } \tau_j = \frac{\sum_{i=1}^n \phi_i q_{ij}}{\sum_{i=1}^n q_{ij}}.$$

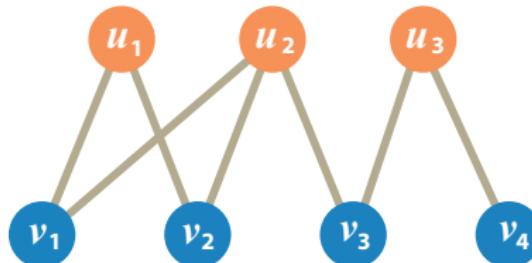
- Take  $I^*$  as first  $\hat{n}$  indices  $i$  from  $I$  with maximal  $\phi_i$ , and  $J^*$  as first  $\hat{m}$  indices  $j$  from  $J$  with maximal  $\tau_j$ , i.e.

$$I^* = \{i_1, \dots, i_{\hat{n}}\} \text{ where } \phi_{i_1} > \dots > \phi_{i_{\hat{n}}} > \dots > \phi_{i_n},$$

$$J^* = \{j_1, \dots, j_{\hat{n}}\} \text{ where } \tau_{j_1} > \dots > \tau_{j_{\hat{n}}} > \dots > \tau_{j_n}.$$

## The idea behind authority factors

We can treat binary matrix  $\mathbf{Q}$  as adjacency matrix of some bipartite graph  $G$ , which vertices are splitted in two disjoined sets  $U$  and  $V$ .

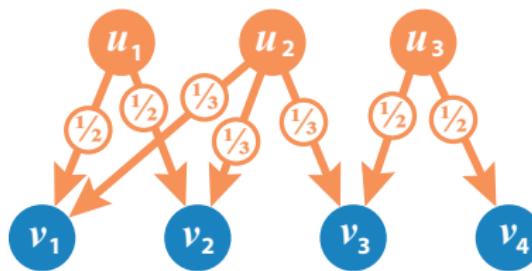


$$\mathbf{Q} = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

Lets consider random walk process on graph  $G$ , with equal probabilities for each outgoing arc. Thus transition from set  $U$  to set  $V$  can be represented in form of stochastic matrix  $\mathbf{Q}_1$  and transition from  $V$  to  $U$  as  $\mathbf{Q}_2^T \dots$

# The idea behind authority factors

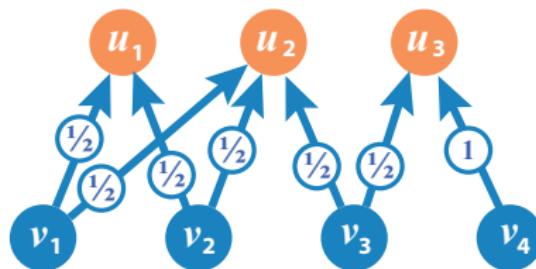
Thus transition from set  $U$  to set  $V$  can be represented in form of stochastic matrix  $\mathbf{Q}_1$  and transition from  $V$  to  $U$  as  $\mathbf{Q}_2^T$



$$\mathbf{Q}_1 = \begin{pmatrix} 1/2 & 1/2 & 0 & 0 \\ 1/3 & 1/3 & 1/3 & 0 \\ 0 & 0 & 1/2 & 1/2 \end{pmatrix}$$

# The idea behind authority factors

Thus transition from set  $U$  to set  $V$  can be represented in form of stochastic matrix  $\mathbf{Q}_1$  and transition from  $V$  to  $U$  as  $\mathbf{Q}_2^T$



$$\mathbf{Q}_2^T = \begin{pmatrix} 1/2 & 1/2 & 0 \\ 1/2 & 1/2 & 0 \\ 0 & 1/2 & 1/2 \\ 0 & 0 & 1 \end{pmatrix}$$

## The idea behind authority factors

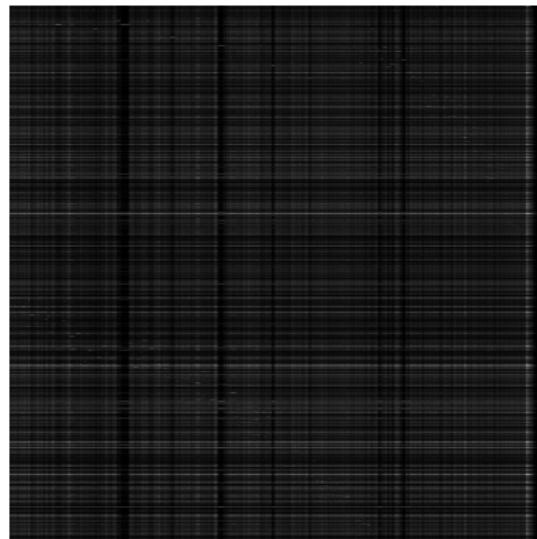
- In this setting authority factors  $\phi_i$  and  $\tau_j$  are essentially stationary probabilities for random walk process defined by stochastic matrices  $\mathbf{Q}_1 \mathbf{Q}_2^T$  and  $\mathbf{Q}_1^T \mathbf{Q}_2$  respectively and can be described by following equations:

$$\phi = \mathbf{Q}_1 \tau, \text{ and } \tau = \mathbf{Q}_2^T \phi.$$

- The algorithm is very similar to one used by Google which assigns PageRank authority factor to each web-page in the Internet as logarithm of stationary probability for random walk through the Internet graph defined by hyperlinks.

# Data availability

Densities before and after sorting by authority factors



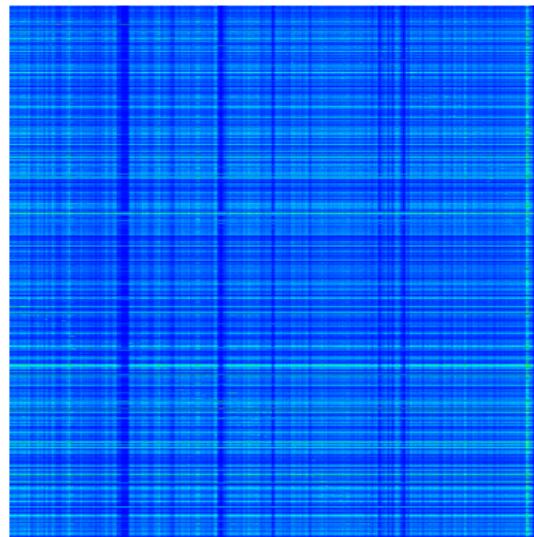
initial

after 5 iterations



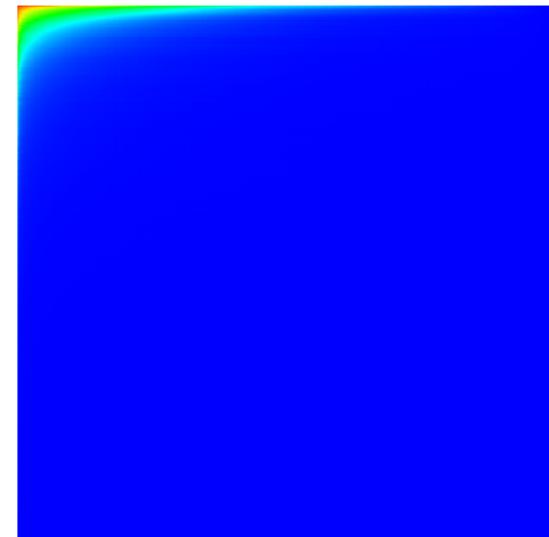
# Data availability

Densities before and after sorting by authority factors



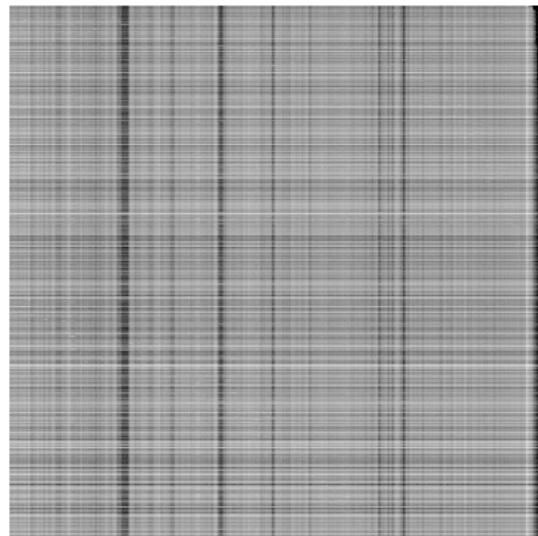
initial

after 5 iterations

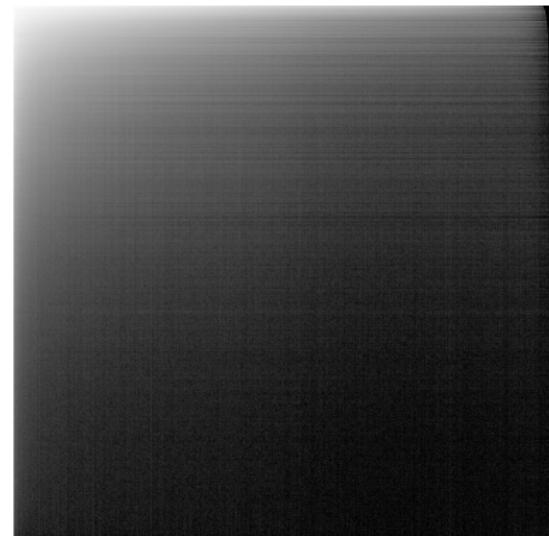


# Data availability

Densities before and after sorting by authority factors: log-scale



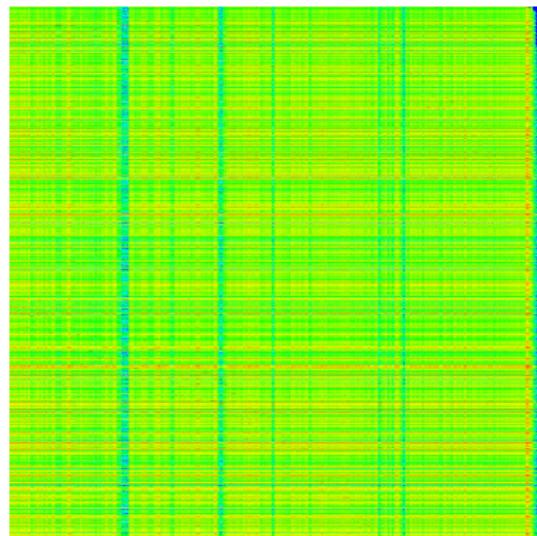
initial



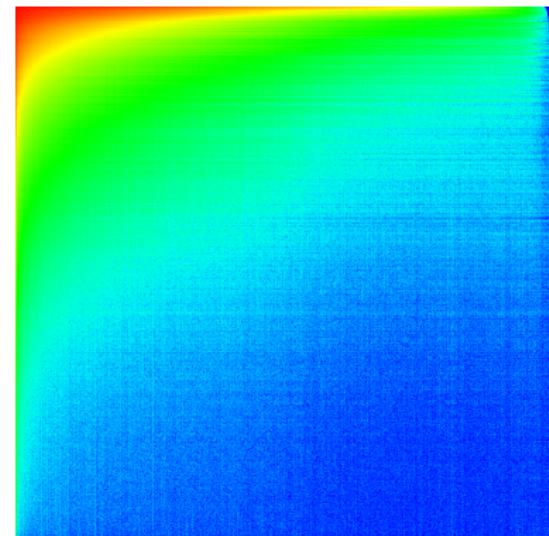
after 5 iterations

# Data availability

Densities before and after sorting by authority factors: log-scale



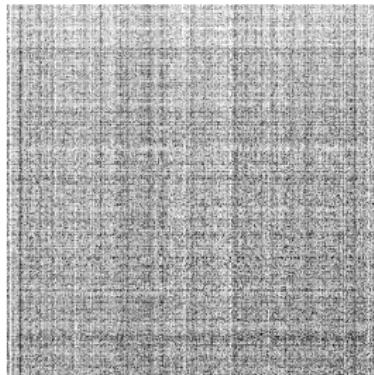
initial



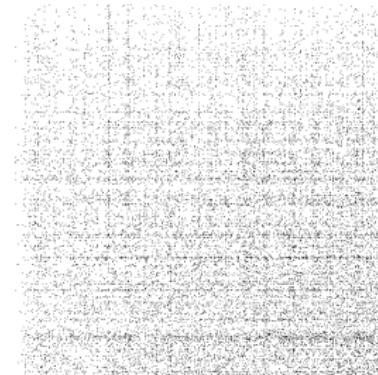
after 5 iterations

## Submatrix selection

- $500 \times 500$  – upper left corner of data sorted according to authority factors;
- missing ratio – 7% (score fill ratio increased from 1.18% to 93%).



scores



missing data (black pixels)

# Interpolating missing data

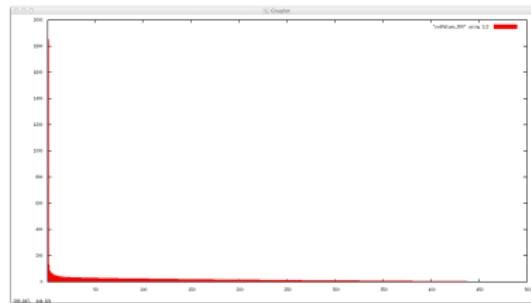
- Interpolation aim:
  - provide solid matrix for singular value decomposition for checking hypothesis about low rank approximation;
  - not to predict missing data(!).
- Lets denote  $I(j) = \{i : \exists y_{ij}\}$  and  $J(i) = \{j : \exists y_{ij}\}$ .
- Interpolation algorithm:

$$\tilde{y}_{ij} = \frac{\sum_{k \in I(j)} y_{kj} + \sum_{k \in J(i)} y_{ik}}{|I(j)| + |J(i)|}.$$

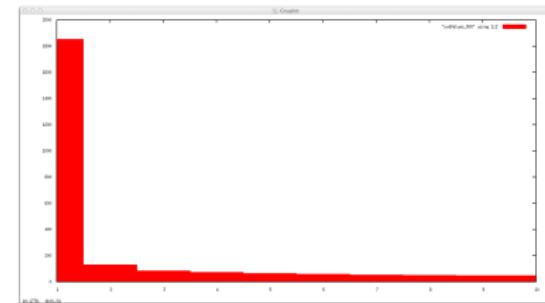
- Patch matrix  $\hat{\mathbf{Y}} = \{\hat{y}_{ij}\}$  where

$$\hat{y}_{ij} = \begin{cases} y_{ij} & \text{if } y_{ij} \text{ is available;} \\ \tilde{y}_{ij} & \text{otherwise;} \end{cases}$$

# Singular Values of matrix $\hat{Y}$

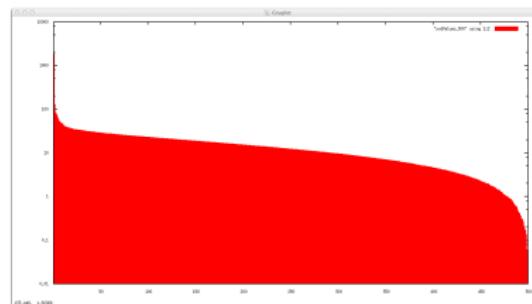


all values

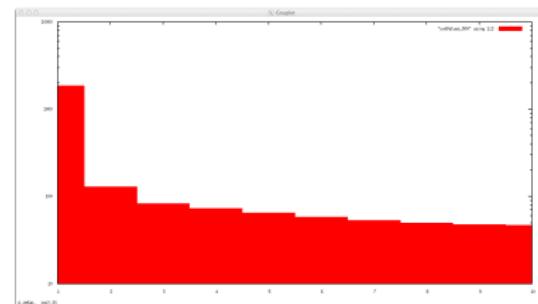


first 10 values

# Singular Values of matrix $\hat{Y}$ : logscale



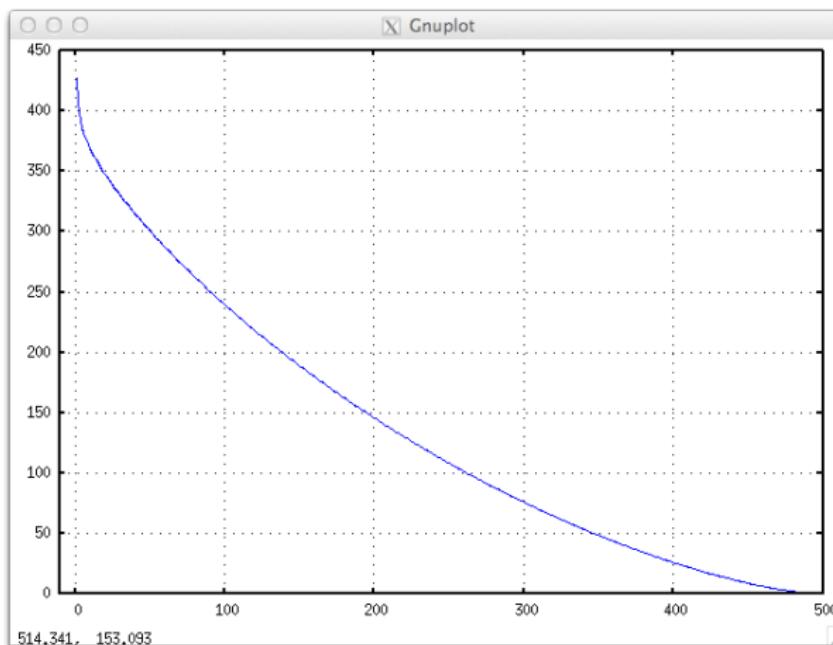
all values



first 10 values

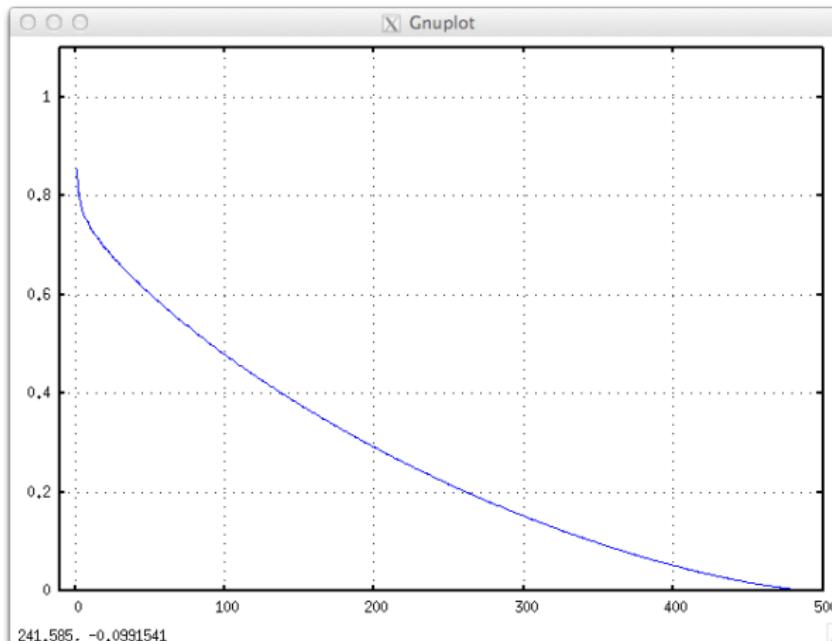
# Dependency $|\hat{\mathbf{Y}} - \tilde{\mathbf{Y}}|_F$ over approximation rank $k$

Let  $\tilde{\mathbf{Y}}$  is lower rank  $k$  approximation  $\hat{\mathbf{Y}}$  of matrix  $\mathbf{Y}$



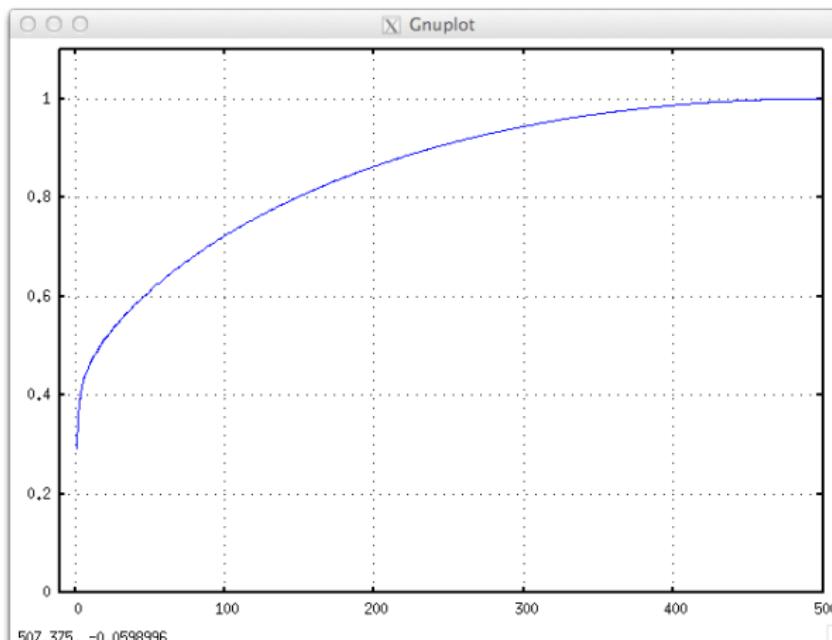
# Dependency of RMSE over approximation rank $k$

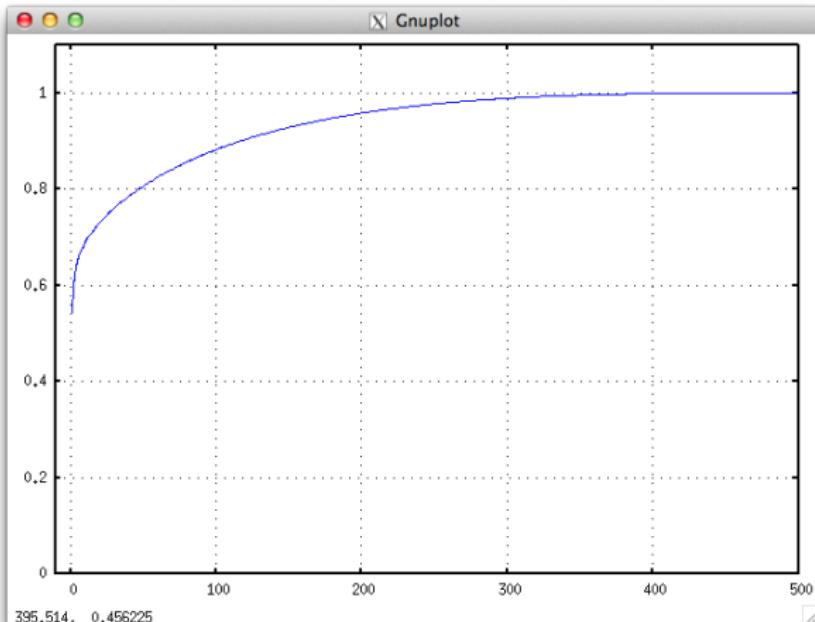
$$RMSE = \sqrt{\frac{|\hat{\mathbf{Y}} - \tilde{\mathbf{Y}}|_F^2}{\hat{n}\hat{m}}}$$



# Dependency of $R_{adj}^2$ over approximation rank $k$

$R_{adj}^2 = 1 - \frac{ESS/(p-q)}{TSS/(p-1)}$  where  $p = \hat{n}\hat{m}$ , and  $q = (\hat{n} + \hat{m})\frac{k}{2}$



Dependency of  $\text{correl}(\mathbf{Y}, \hat{\mathbf{Y}})$  over approximation rank  $k$ 

- [1] J. Bennett, S. Lanning, and N. Netflix, “The netflix prize,” in *In KDD Cup and Workshop in conjunction with KDD*, 2007.

