Collaborative Filtering: Models and Algorithms

Andrea Montanari

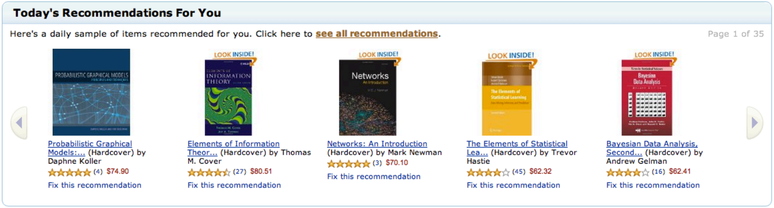
Jose Bento, Yash Deshpande, Adel Javanmard, Raghunandan Keshavan, Sewoong Oh,
Stratis Ioannidis, Nadia Fawaz, Amy Zhang
Stanford University, Technicolor

September 15, 2012

Problem statement

Given data on the activity of a set of users, provide personalized recommendations to users X, Y, Z,. . .

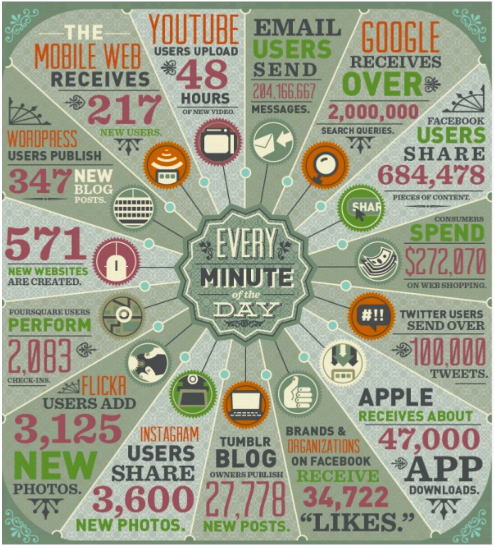
Example



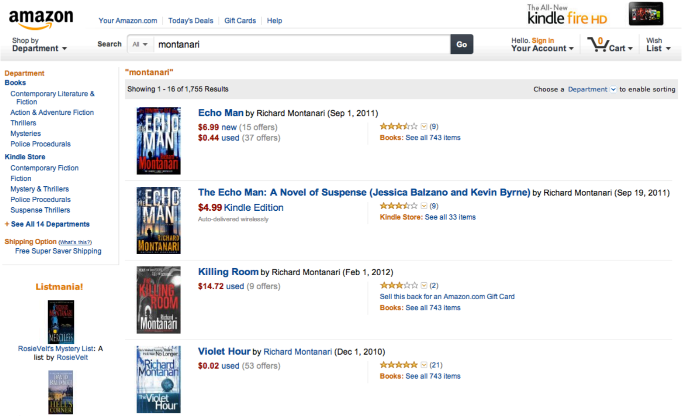
Today's Recommendations For You
Here's a daily sample of items recommended for you. Click here to see all recommendations.
Page 1 of 35
Probabilistic Graphical Models:... (Hardcover) by Daphne Koller
(4) $74.90
Fix this recommendation
Elements of Information Theor... (Hardcover) by Thomas M. Cover
(27) $80.51
Fix this recommendation
Networks: An Introduction (Hardcover) by Mark Newman
(3) $70.10
Fix this recommendation
The Elements of Statistical Lea... (Hardcover) by Trevor Hastie
(45) $62.32
Fix this recommendation
Bayesian Data Analysis, Second... (Hardcover) by Andrew Gelman
(16) $62.41
Fix this recommendation

Andrea, Welcome to Your Amazon.com (If you're not Andrea Montanari, click here.)
Today's Recommendations For You
Here's a daily sample of items recommended for you. Click here to see all recommendations.
Page 5 of 35 (Start over)
Large-Scale Inference: Empirica... (Hardcover) by Bradley Efron
(2) $59.31
Fix this recommendation
Sesame Street - Fiesta! DVD ~ Celia Cruz
(58) $8.49
Fix this recommendation
Introducing Monte Carlo M... (Paperback) by Christian P. Robert
$52.25
Fix this recommendation
Maple Teethers
(9) $13.45
Fix this recommendation
Data Manipulation with R (Use R) (Paperback) by Phil Spector
(15) $46.83
Fix this recommendation

An obviously useful technology



An obviously useful technology



Outline

A model

Setting

$$\text{Users}: \quad i \in \{1, 2, \dots, m\}$$

$$\text{Movies}: \quad j \in \{1, 2, \dots, n\}$$

When user i watches movie j, she enters her rating R_{ij}.

Want to predict ratings for missing pairs.

Setting

$$\text{Users}: \quad i \in \{1, 2, \dots, m\}$$

$$\text{Movies}: \quad j \in \{1, 2, \dots, n\}$$

When user i watches movie j, she enters her rating R_{ij}.

Want to predict ratings for missing pairs.

Linear regression model

Movie j

$$v_j = (\text{genre}; \;\; \text{main actor}; \;\; \text{supporting actor}; \;\; \text{year}; \ldots) \in \mathbb{R}^r$$

$$\mathsf{R}_{ij} \sim c_i + \langle u_i, v_j \rangle + \varepsilon_{ij}$$

Want: User parameters vector u_i

Linear regression model

Movie j

$$v_j = (\text{genre};\ \ \text{main actor};\ \text{supporting actor};\ \ \text{year}; \dots) \in \mathbb{R}^r$$

$$\mathsf{R}_{ij} \sim c_i + \langle u_i, v_j \rangle + \varepsilon_{ij}$$

Want: User parameters vector u_i

Linear regression model

Movie j

$$v_j = (\text{genre}; \;\; \text{main actor}; \;\; \text{supporting actor}; \;\; \text{year}; \ldots) \in \mathbb{R}^r$$

$$\mathsf{R}_{ij} \sim c_i + \langle u_i, v_j \rangle + \varepsilon_{ij}$$

Want: User parameters vector u_i

Linear regression model

Movie j

$$v_j = (\text{genre};\ \ \text{main actor};\ \ \text{supporting actor};\ \ \text{year}; \ldots) \in \mathbb{R}^r$$

$$\mathsf{R}_{ij} \sim \cancel{c_i} + \langle u_i, v_j \rangle + \varepsilon_{ij}$$

Want: User parameters vector u_i

Least squares

$$u_i = \arg\min_{x_i \in \mathbb{R}^r} \left\{ \sum_{j \in \texttt{WatchedBy}(i)} \Big(\mathsf{R}_{ij} - \langle x_i, v_j \rangle \Big)^2 \right\}$$

Ridge regression

$$u_i = \arg\min_{x_i \in \mathbb{R}^r} \left\{ \sum_{j \in \text{WatchedBy}(i)} \Big(\mathsf{R}_{ij} - \langle x_i, v_j \rangle \Big)^2 + \lambda \, \|x_i\|^2 \right\}$$

- How do we construct the v_j's?
- Ad hoc definitions are not suited to recommendation!

Ridge regression

$$u_i = \arg\min_{x_i \in \mathbb{R}^r} \left\{ \sum_{j \in \text{WatchedBy}(i)} \Big(\mathsf{R}_{ij} - \langle x_i, v_j \rangle \Big)^2 + \lambda \, \|x_i\|^2 \right\}$$

- How do we construct the v_j's?
- Ad hoc definitions are not suited to recommendation!

If I knew the users' feature vectors

$$\mathsf{R}_{ij} \sim \langle u_i, v_j \rangle + \varepsilon_{ij}$$

$$v_j = \arg\min_{y_j \in \mathbb{R}^r} \left\{ \sum_{i \in \text{Watched}(j)} \Big(\mathsf{R}_{ij} - \langle u_i, y_j \rangle \Big)^2 + \lambda \, \|y_j\|^2 \right\}$$

If I knew the users' feature vectors

$$\mathsf{R}_{ij} \sim \langle u_i, v_j \rangle + \varepsilon_{ij}$$

$$v_j = \arg\min_{y_j \in \mathbb{R}^r} \left\{ \sum_{i \in \text{Watched}(j)} \Big(\mathsf{R}_{ij} - \langle u_i, y_j \rangle \Big)^2 + \lambda \, \|y_j\|^2 \right\}$$

Everything together

$$
\begin{aligned}
u_i &= \arg\min_{x_i \in \mathbb{R}^r} \left\{ \sum_{j \in \text{WatchedBy}(i)} \Big(\mathsf{R}_{ij} - \langle x_i, v_j \rangle \Big)^2 + \lambda \|x_i\|^2 \right\} \\
v_j &= \arg\min_{y_j \in \mathbb{R}^r} \left\{ \sum_{i \in \text{Watched}(j)} \Big(\mathsf{R}_{ij} - \langle u_i, y_j \rangle \Big)^2 + \lambda \|y_j\|^2 \right\}
\end{aligned}
$$

Minimize (E = Watched)

$$
F(X,Y) = \sum_{(i,j)\in E} \Big(\mathsf{R}_{ij} - \langle x_i, y_j \rangle \Big)^2 + \lambda \sum_{i=1}^{m} \|x_i\|_2^2 + \lambda \sum_{j=1}^{n} \|y_j\|_2^2
$$

Everything together

$$
\begin{aligned}
u_i &= \arg\min_{x_i \in \mathbb{R}^r} \left\{ \sum_{j \in \mathrm{WatchedBy}(i)} \Big(\mathsf{R}_{ij} - \langle x_i, v_j \rangle \Big)^2 + \lambda \, \|x_i\|^2 \right\} \\
v_j &= \arg\min_{y_j \in \mathbb{R}^r} \left\{ \sum_{i \in \mathrm{Watched}(j)} \Big(\mathsf{R}_{ij} - \langle u_i, y_j \rangle \Big)^2 + \lambda \, \|y_j\|^2 \right\}
\end{aligned}
$$

Minimize ($E = \mathrm{Watched}$)

$$
F(X, Y) = \sum_{(i,j) \in E} \Big(\mathsf{R}_{ij} - \langle x_i, y_j \rangle \Big)^2 + \lambda \sum_{i=1}^{m} \|x_i\|_2^2 + \lambda \sum_{j=1}^{n} \|y_j\|_2^2
$$

Objective function

Minimize (E = Watched)

$$F(X,Y) = \sum_{(i,j)\in E} \Big(\mathsf{R}_{ij} - \langle x_i, y_j\rangle\Big)^2 + \lambda \sum_{i=1}^{m} \|x_i\|_2^2 + \lambda \sum_{j=1}^{n} \|y_j\|_2^2$$

$$\equiv \|\mathcal{P}_E(\mathsf{R} - XY^{\mathsf{T}})\|_F^2 + \lambda\|X\|_F^2 + \lambda\|Y\|_F^2$$

$$\mathcal{P}_E(A) = \begin{cases} A_{ij} & \text{if } (i,j)\in E, \\ 0 & \text{otherwise} \end{cases}$$

$$X^{\mathsf{T}} = \Big[x_1\Big|x_2\Big|\cdots\Big|x_m\Big]$$

$$Y^{\mathsf{T}} = \Big[y_1\Big|y_2\Big|\cdots\Big|y_n\Big]$$

Objective function

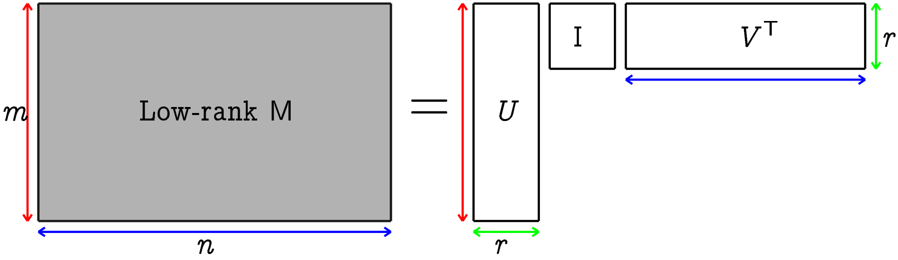
Minimize (E = Watched)

$$
\begin{aligned}
F(X, Y) &= \sum_{(i,j)\in E} \Big(\mathsf{R}_{ij} - \langle x_i, y_j\rangle\Big)^2 + \lambda \sum_{i=1}^{m} \|x_i\|_2^2 + \lambda \sum_{j=1}^{n} \|y_j\|_2^2 \\
&\equiv \|\mathcal{P}_E(\mathsf{R} - XY^{\mathsf{T}})\|_F^2 + \lambda \|X\|_F^2 + \lambda \|Y\|_F^2
\end{aligned}
$$

$$
\mathcal{P}_E(A) = \begin{cases} A_{ij} & \text{if } (i,j) \in E, \\ 0 & \text{otherwise} \end{cases}
$$

$$
\begin{aligned}
X^{\mathsf{T}} &= \Big[x_1 \Big| x_2 \Big| \cdots \Big| x_m \Big] \\
Y^{\mathsf{T}} &= \Big[y_1 \Big| y_2 \Big| \cdots \Big| y_n \Big]
\end{aligned}
$$

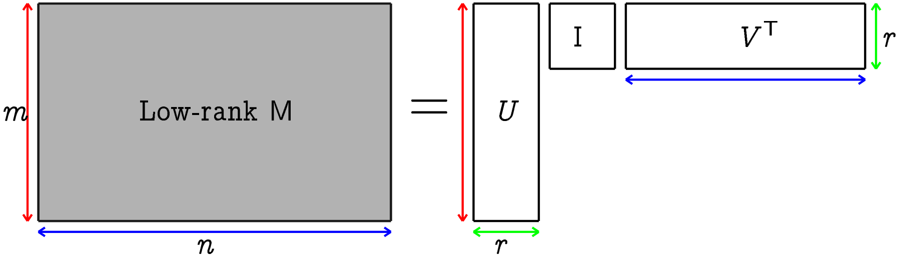
Low rank structure



1. Low-rank matrix M
2. $\mathrm{R} = \mathrm{M} + \mathrm{Z}$
3. Observed subset E

$$\mathcal{P}_E(\mathrm{R})_{ij} = \begin{cases} \mathrm{M}_{ij} + \mathrm{Z}_{ij} & \text{if } (i,j) \in E\,, \\ 0 & \text{otherwise.} \end{cases}$$

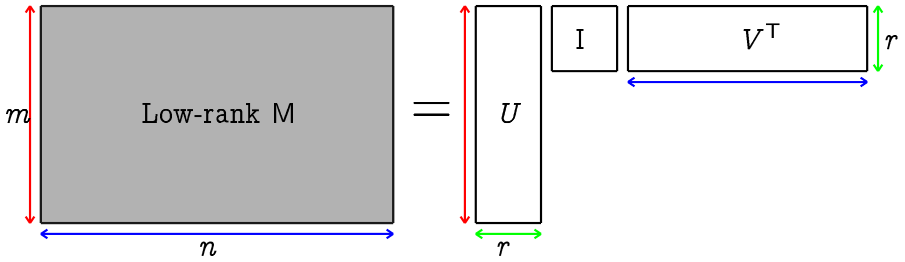
Low rank structure



1. Low-rank matrix M
2. $R = M + Z$
3. Observed subset E

$$\mathcal{P}_E(R)_{ij} = \begin{cases} M_{ij} + Z_{ij} & \text{if } (i,j) \in E\,, \\ 0 & \text{otherwise.} \end{cases}$$

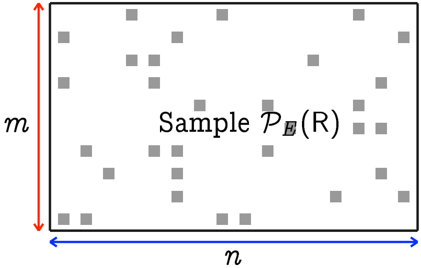
Low rank structure



1. Low-rank matrix M
2. $\mathsf{R} = \mathsf{M} + \mathsf{Z}$
3. Observed subset E

$$\mathcal{P}_E(\mathsf{R})_{ij} = \begin{cases} \mathsf{M}_{ij} + \mathsf{Z}_{ij} & \text{if } (i,j) \in E\,, \\ 0 & \text{otherwise.} \end{cases}$$

Low rank structure



1. Low-rank matrix M
2. $\mathsf{R} = \mathsf{M} + \mathsf{Z}$
3. Observed subset E

$$\mathcal{P}_E(\mathsf{R})_{ij} = \begin{cases} \mathsf{M}_{ij} + \mathsf{Z}_{ij} & \text{if } (i,j) \in E\,, \\ 0 & \text{otherwise.} \end{cases}$$

Algorithms and accuracy

Questions

- How do we minimize $F(X, Y)$?

- What prediction accuracy? $(\mathsf{RMSE} = \|\mathsf{M} - \hat{\mathsf{M}}\|_F/\sqrt{mn})$

How do we minimize $F(X, Y)$?

- Spectral methods.
- Gradient method. [Srebro, Rennie, Jaakkola, 2003]
- Convex relaxations. [Fazel, Hindi, Boyd, 2001]

Spectral method (for simplicity $\lambda = 0$)

Replace

$$\begin{aligned} F(X,Y) &= \|\mathcal{P}_E(\mathsf{R} - XY^{\mathsf{T}})\|_F^2 \\ &= \|\mathcal{P}_E(XY^{\mathsf{T}})\|_F^2 - 2\langle \mathcal{P}_E(R), XY^{\mathsf{T}}\rangle + \text{const.} \end{aligned}$$

with ($p = |E|/mn$ fraction of observed entries)

$$\begin{aligned} \widetilde{F}(X,Y) &= p\,\|XY^{\mathsf{T}}\|_F^2 - 2\langle \mathcal{P}_E(R), XY^{\mathsf{T}}\rangle + \text{const} \\ &= p\,\left\| XY^{\mathsf{T}} - \frac{1}{p}\mathcal{P}_E(R)\right\|_F^2 \end{aligned}$$

Spectral method (for simplicity $\lambda = 0$)

Replace

$$
\begin{aligned}
F(X,Y) &= \|\mathcal{P}_E(\mathsf{R} - XY^{\mathsf{T}})\|_F^2 \\
&= \|\mathcal{P}_E(XY^{\mathsf{T}})\|_F^2 - 2\langle \mathcal{P}_E(R), XY^{\mathsf{T}}\rangle + \text{const.}
\end{aligned}
$$

with ($p = |E|/mn$ fraction of observed entries)

$$
\begin{aligned}
\widetilde{F}(X,Y) &= p\,\|XY^{\mathsf{T}}\|_F^2 - 2\langle \mathcal{P}_E(R), XY^{\mathsf{T}}\rangle + \text{const} \\
&= p\,\left\| XY^{\mathsf{T}} - \frac{1}{p}\mathcal{P}_E(R)\right\|_F^2
\end{aligned}
$$

Spectral method (for simplicity $\lambda = 0$)

Replace

$$\begin{aligned} F(X,Y) &= \|\mathcal{P}_E(\mathsf{R} - XY^{\mathsf{T}})\|_F^2 \\ &= \|\mathcal{P}_E(XY^{\mathsf{T}})\|_F^2 - 2\langle \mathcal{P}_E(R), XY^{\mathsf{T}}\rangle + \text{const.} \end{aligned}$$

with ($p = |E|/mn$ fraction of observed entries)

$$\begin{aligned} \widetilde{F}(X,Y) &= p\,\|XY^{\mathsf{T}}\|_F^2 - 2\langle \mathcal{P}_E(R), XY^{\mathsf{T}}\rangle + \text{const} \\ &= p\,\left\| XY^{\mathsf{T}} - \frac{1}{p}\mathcal{P}_E(R)\right\|_F^2 \end{aligned}$$

Spectral method (for simplicity $\lambda = 0$)

Minimize

$$\widetilde{F}(X, Y) = \left\| XY^{\mathsf{T}} - \frac{1}{p}\mathcal{P}_E(R) \right\|_F^2$$

Solved by SVD

$$\mathcal{P}_E(\mathsf{R}) = XSY^{\mathsf{T}} \Rightarrow \hat{\mathsf{M}} = \frac{1}{p} X_{m\times r} (S)_{r\times r} Y^{\mathsf{T}}_{n\times r}$$

Spectral method (for simplicity $\lambda = 0$)

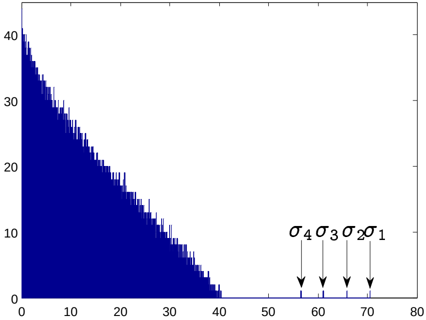
Minimize

$$\widetilde{F}(X, Y) \;=\; \left\| XY^{\mathsf{T}} - \frac{1}{p}\mathcal{P}_E(R) \right\|_F^2$$

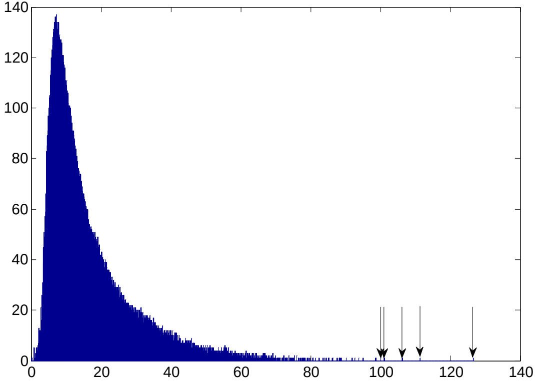
Solved by SVD

$$\mathcal{P}_E(\mathsf{R}) = XSY^{\mathsf{T}} \;\Rightarrow \hat{\mathsf{M}} = \frac{1}{p} X_{m\times r}(S)_{r\times r}\, Y^{\mathsf{T}}_{n\times r}$$

Random matrix $r = 4$, $m = n = 10000$, $p = 0.0012$



Netflix data (trimmed)



RMSE ≈ 0.99

Accuracy guarantees

Theorem (Keshavan, M, Oh, 2009)

Qssume $|\mathsf{M}_{ij}| \le \mathsf{M}_{\max}$. *Then, w.h.p., rank-r projection achieves*

$$\mathsf{RMSE} \quad \le \quad C\,\mathsf{M}_{\max}\sqrt{nr/|E|} + C'\,\|\mathsf{Z}^E\|_2\, n\sqrt{r}/|E|\;.$$

E.g. Gaussian noise: $C''(1+\sigma_z)\,\sqrt{r\,n/|E|}$
[Improves over Achlioptas-McSherry 2003]

Gradient descent

$$F(X,Y) = \sum_{(i,j)\in E} \Big(\mathsf{R}_{ij} - \langle x_i, y_j\rangle\Big)^2 + \lambda \sum_{i=1}^{m} \|x_i\|_2^2 + \lambda \sum_{j=1}^{n} \|y_j\|_2^2$$

Update rule: (γ = 'learning rate')

$$x_i \leftarrow (1-\lambda\gamma)x_i + \gamma \sum_{j\in \text{WatchedBy}(i)} \Big(\mathsf{R}_{ij} - \langle x_i, y_j\rangle\Big) y_j$$

$$y_j \leftarrow (1-\lambda\gamma)y_j + \gamma \sum_{i\in \text{Watched}(j)} \Big(\mathsf{R}_{ij} - \langle x_i, y_j\rangle\Big) x_i$$

Gradient descent

$$F(X,Y) \;=\; \sum_{(i,j)\in E} \Big(\mathsf{R}_{ij} - \langle x_i, y_j\rangle\Big)^2 + \lambda \sum_{i=1}^{m} \|x_i\|_2^2 + \lambda \sum_{j=1}^{n} \|y_j\|_2^2$$

Update rule: (γ = 'learning rate')

$$x_i \;\leftarrow\; (1-\lambda\gamma)x_i + \gamma \sum_{j\in \mathrm{WatchedBy}(i)} \Big(\mathsf{R}_{ij} - \langle x_i, y_j\rangle\Big) y_j$$

$$y_j \;\leftarrow\; (1-\lambda\gamma)y_j + \gamma \sum_{i\in \mathrm{Watched}(j)} \Big(\mathsf{R}_{ij} - \langle x_i, y_j\rangle\Big) x_i$$

Interpretation

$$x_i \leftarrow (1 - \lambda\gamma)x_i + \gamma \sum_{j \in \text{WatchedBy}(i)} w_{ij}\ y_j$$

$$y_j \leftarrow (1 - \lambda\gamma)y_j + \gamma \sum_{i \in \text{Watched}(j)} w_{ij}\ x_i$$

user $\leftarrow$ avg of movies she liked

movie $\leftarrow$ avg of users that liked it

Interpretation

$$
\begin{aligned}
x_i &\leftarrow (1-\lambda\gamma)x_i + \gamma \sum_{j \in \text{WatchedBy}(i)} w_{ij}\; y_j \\
y_j &\leftarrow (1-\lambda\gamma)y_j + \gamma \sum_{i \in \text{Watched}(j)} w_{ij}\; x_i
\end{aligned}
$$

user	←	avg of movies she liked
movie	←	avg of users that liked it

A variant (stochastic gradient)

Pick $(i, j) \in E$:

$$\begin{aligned} x_i &\leftarrow (1 - \lambda\gamma) x_i + \gamma \, w_{ij} \, y_j \\ y_j &\leftarrow (1 - \lambda\gamma) y_j + \gamma \, w_{ij} \, x_i \end{aligned}$$

[Srebro, Rennie, Jaakkola, 2003]
[Srebro, Jaakkola, 2005]

A variant (stochastic gradient)

Pick $(i, j) \in E$:

$$
\begin{aligned}
x_i &\leftarrow (1-\lambda\gamma)x_i + \gamma\ w_{ij}\ y_j \\
y_j &\leftarrow (1-\lambda\gamma)y_j + \gamma\ w_{ij}\ x_i
\end{aligned}
$$

[Srebro, Rennie, Jaakkola, 2003]

[Srebro, Jaakkola, 2005]

In the words of SimonFunk

Only problem is, we don't have 8.5B entries, we have 100M entries and 8.4B empty cells. Ok, there's another problem too, which is that computing the SVD of ginormous matrices is... well, no fun. Unless you're into that sort of thing.

But, just because there are five hundred really complicated ways of computing singular value decompositions in the literature doesn't mean there isn't a really simple way too: Just take the derivative of the approximation error and follow it. This has the added bonus that we can choose to simply ignore the unknown error on the 8.4B empty slots.

So, yeah, you mathy guys are rolling your eyes right now as it dawns on you how short the path was.

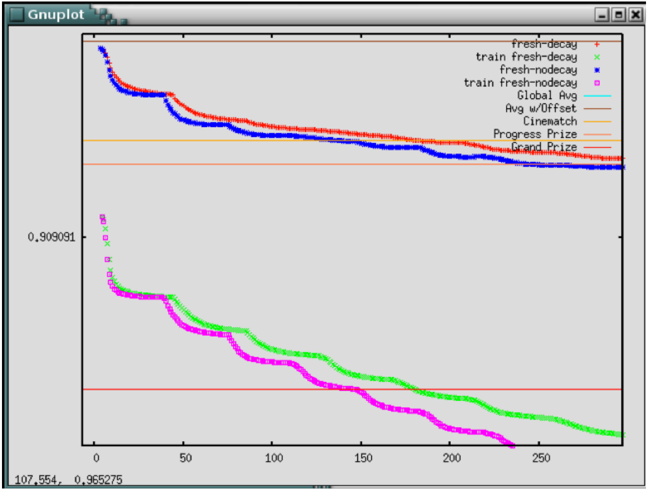
If you write out the equations for the error between the SVD-like model and the original data--just the given values, not the empties--and then take the derivative with respect to the parameters we're trying to infer, you get a rather simple result which I'll give here in C code to save myself the trouble of formatting the math:

```
userValue[user] += lrate * err * movieValue[movie];
movieValue[movie] += lrate * err * userValue[user];
```

This is kind of like the scene in the Wizard of Oz where Toto pulls back the curtain, isn't it. But wait... let me fluff it up some and make it sound more impressive.

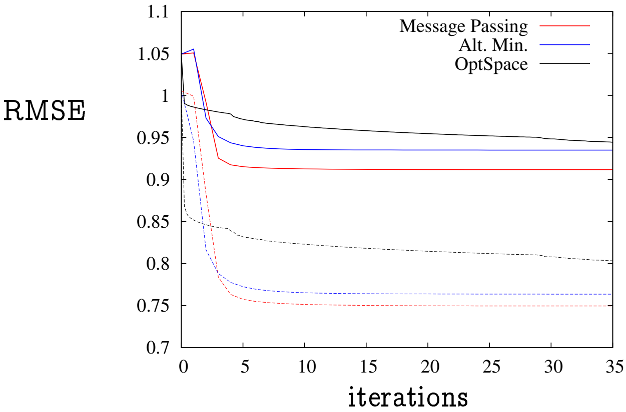
[Neflix challenge, 2006-2009]

And his results



Target RMSE ≤ 0.8564

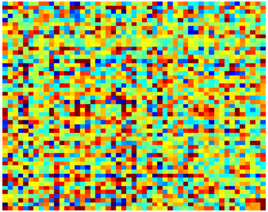
Three variants



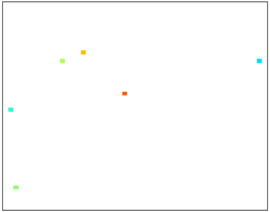
OptSpace ∼ Gradient descent on Grassmannian [Keshavan-M.-Oh 2009]
Alternating Least Squares [Koren-Bell 2008]
Message Passing [Keshavan-M. 2011]

Accuracy guarantees: Vignette ($m = n = 2000$ rank-8)

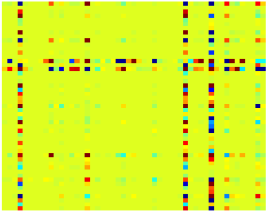
low-rank matrix M



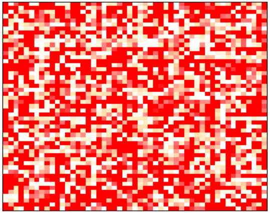
sampled matrix $M^{\mathbb{E}}$



OptSpace output $\hat{M}$



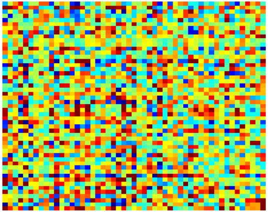
squared error $(M - \hat{M})^2$



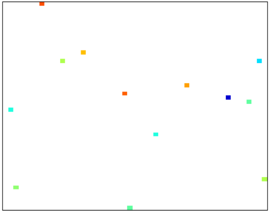
0.25% sampled

Accuracy guarantees: Vignette ($m = n = 2000$ rank-8)

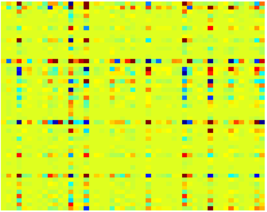
low-rank matrix M



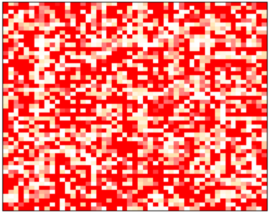
sampled matrix $M^{\mathbb{E}}$



OptSpace output $\hat{M}$



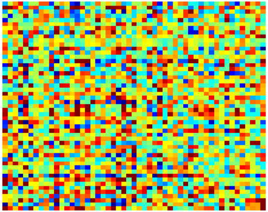
squared error $(M - \hat{M})^2$



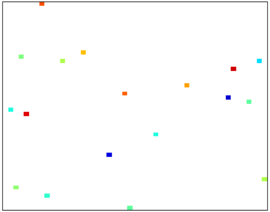
0.50% sampled

Accuracy guarantees: Vignette ($m = n = 2000$ rank-8)

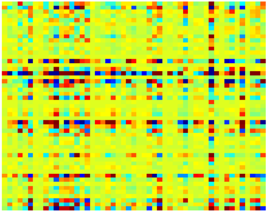
low-rank matrix M



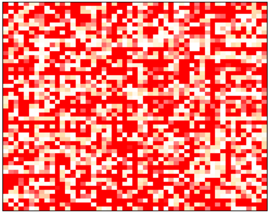
sampled matrix $\mathsf{M}^{\mathbb{E}}$



OptSpace output $\hat{\mathsf{M}}$

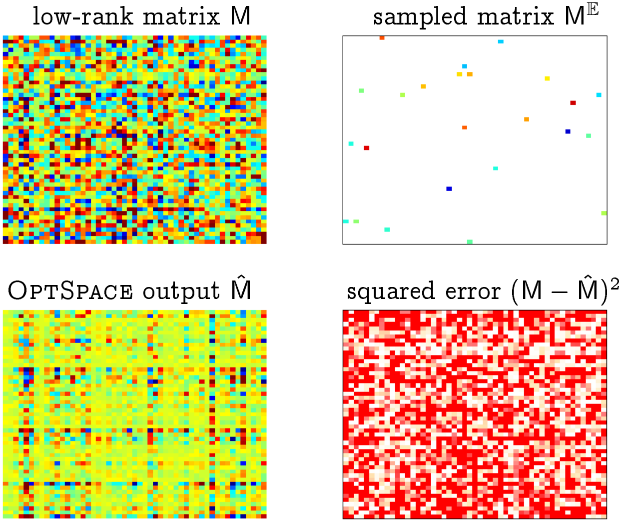


squared error $(\mathsf{M} - \hat{\mathsf{M}})^2$



0.75% sampled

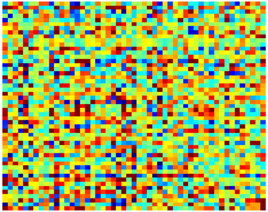
Accuracy guarantees: Vignette ($m = n = 2000$ rank-8)



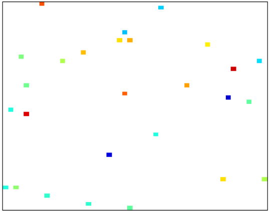
1.00% sampled

Accuracy guarantees: Vignette ($m = n = 2000$ rank-8)

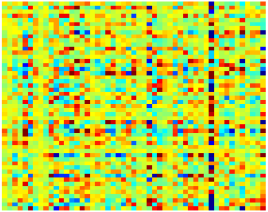
low-rank matrix M



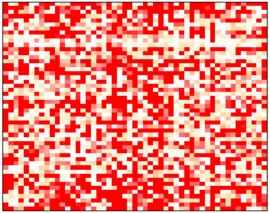
sampled matrix $M^{\mathbb{E}}$



OPTSPACE output $\hat{M}$



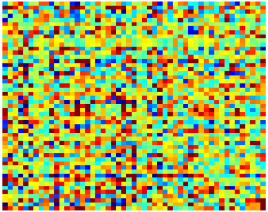
squared error $(M - \hat{M})^2$



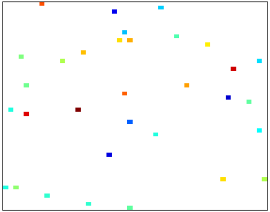
1.25% sampled

Accuracy guarantees: Vignette ($m = n = 2000$ rank-8)

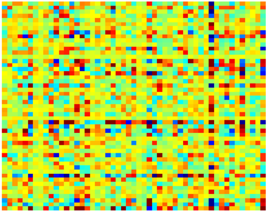
low-rank matrix M



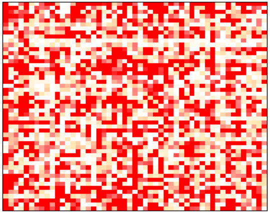
sampled matrix $M^{\mathbb{E}}$



OptSpace output $\hat{M}$



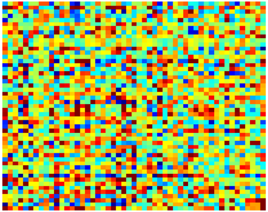
squared error $(M - \hat{M})^2$



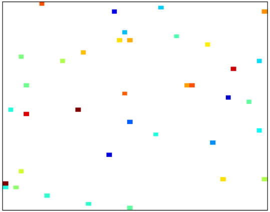
1.50% sampled

Accuracy guarantees: Vignette ($m = n = 2000$ rank-8)

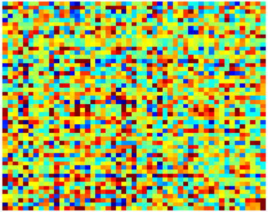
low-rank matrix M



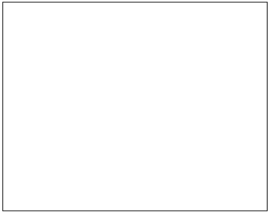
sampled matrix $M^{\mathbb{E}}$



OptSpace output $\hat{M}$



squared error $(M - \hat{M})^2$



1.75% sampled

Accuracy guarantees: Unstructured factors

Incoherence

$$\|u_i\|^2 \le \mu \,\langle \|u_\cdot\|^2 \rangle_{\rm av}\,, \qquad \|v_j\|_2^2 \le \mu \,\langle \|v_\cdot\|^2 \rangle_{\rm av}\,.$$

[Candés, Recht 2008]

Accuracy guarantees

Theorem (Keshavan, M, Oh, 2009)

Assume $|\mathsf{M}_{ij}| \le \mathsf{M}_{\max}$. *Then, w.h.p., rank-r projection achieves*

$$\mathsf{RMSE} \;\le\; C\,\mathsf{M}_{\max}\sqrt{nr/|E|} + C'\,\|\mathsf{Z}^E\|_2\, n\sqrt{r}/|E|\;.$$

Theorem (Keshavan, M, Oh, 2009)

Let M *be incoherent with* $\sigma_1(M)/\sigma_r(M) = O(1)$. *If* $|E| \ge Cn\,\min\{r(\log n)^2, r^2\log n\}$ *then, w.h.p.,* OptSpace *achieves*

$$\mathsf{RMSE} \le C''\frac{n\sqrt{r}}{|E|}\|\mathsf{Z}^E\|_2\,,$$

with complexity $O(nr^3(\log n)^2)$.

E.g. Gaussian noise: $C''\sigma_z\sqrt{r\,n/|E|}$

Accuracy guarantees

Theorem (Keshavan, M, Oh, 2009)

Assume $|\mathsf{M}_{ij}| \le \mathsf{M}_{\max}$. *Then, w.h.p., rank-$r$ projection achieves*

$$\mathsf{RMSE} \;\; \le \;\; C\,\mathsf{M}_{\max}\sqrt{nr/|E|} + C'\,\|\mathsf{Z}^E\|_2\, n\sqrt{r}/|E|\;.$$

Theorem (Keshavan, M, Oh, 2009)

Let M *be incoherent with* $\sigma_1(M)/\sigma_r(M) = O(1)$. *If* $|E| \ge Cn\, \min\{r(\log n)^2, r^2 \log n\}$ *then, w.h.p.,* OptSpace *achieves*

$$\mathsf{RMSE} \le C''\frac{n\sqrt{r}}{|E|}\|\mathsf{Z}^E\|_2\;,$$

with complexity $O(nr^3(\log n)^2)$.

E.g. Gaussian noise: $C''\sigma_z\,\sqrt{r\;n/|E|}$

Two surprises

Can do much better than SVD !

Error = Noise / Sampling factor

Analogous guarantees for convex relaxations

Candés, Recht, 2008 *

Candés, Plan, 2009

Candés, Tao, 2009

Gross, 2010

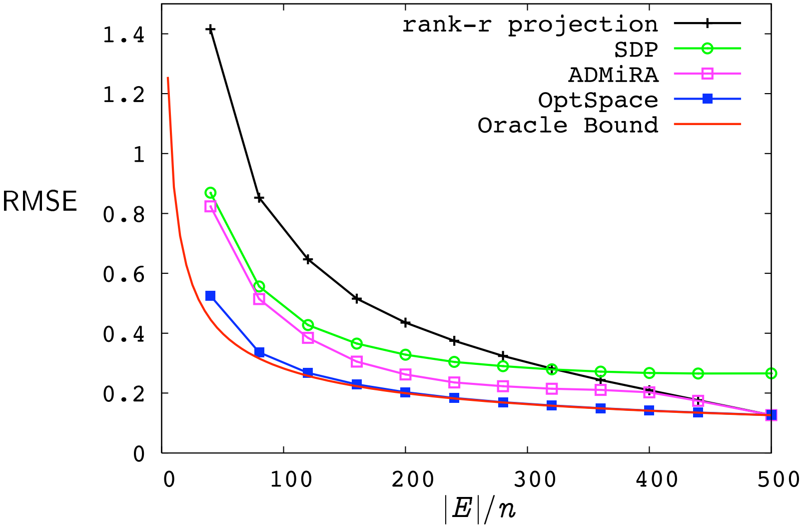
Negahbahn, Wainwright, 2010

Koltchinskii, Lounici, Tsybakov, 2011

. . .

A noisy example

- $n = 500, r = 4, \sigma_z = 1$, example from [Candés, Plan, 2009]



Challenge #1: Privacy

Research question

User ratings → User feature vector

User ratings → User private attributes ???

Let us try to do it!

Research question

User ratings $\rightarrow$ User feature vector

User ratings $\rightarrow$ User private attributes ???

Let us try to do it!

Research question

User ratings $\rightarrow$ User feature vector

User ratings $\rightarrow$ User private attributes ???

Let us try to do it!

Which attribute?

Number of persons in the household

- Non-obvious
- Recommender has incentive
- 2011 CAMRa Challenge [no prize :-(we won it :-)]

Which attribute?

Number of persons in the household

- Non-obvious
- Recommender has incentive
- 2011 CAMRa Challenge [no prize :-(we won it :-)]

CAMRa dataset

- $m \approx 2 \cdot 10^5$ users, $n \approx 2 \cdot 10^4$ movies
- $|E| \approx 4.5 \cdot 10^6$ ratings ($p \approx 0.001$)
- 272 households of size 2
- 14 households of size 3
- 4 households of size 4

CAMRa dataset

- $m \approx 2 \cdot 10^5$ users, $n \approx 2 \cdot 10^4$ movies
- $|E| \approx 4.5 \cdot 10^6$ ratings ($p \approx 0.001$)
- 272 households of size 2
- ~~14 households of size 3~~
- ~~4 households of size 4~~

CAMRa dataset

- $m \approx 2 \cdot 10^5$ users, $n \approx 2 \cdot 10^4$ movies
- $|E| \approx 4.5 \cdot 10^6$ ratings ($p \approx 0.001$)
- 272 households of size 2

- Can you identify whether two users shared an account?
- Can you identify which user watched a movie?

Short answer

Yes: For a significant fraction of the accounts.

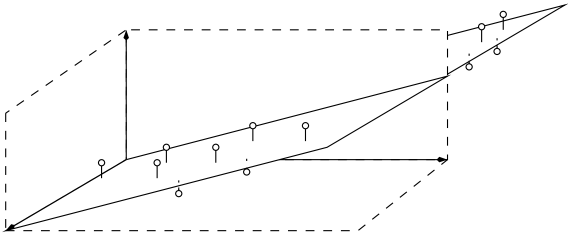
Two examples (Netflix dataset, no ground truth)

User 1	User 2
TLOTR: The Fellowship of the Ring†(5), TLOTR: The Return of the King†(5), TLOTR: The Two Towers†(5), The Whole Nine Yards(4), Immortal†(1), The Deep End(2), Toys†(4), The Addams Family(5)	H.R. Pufnstuf(5), Sex and the City: Season 5$^{\heartsuit}$(1), Me Myself & Irene(1), All the Real Girls$^{\heartsuit\triangle}$(5), Titanic$^{\heartsuit}$(5), George Washington$^{\triangle}$(5), The Siege(1), In the Bedroom$^{\triangle}$(5)

User 1	User 2
Monsters Inc.$^{\diamondsuit}$(5), Finding Nemo$^{\diamondsuit}$(5), Whale Rider(5), Con Air(4), Lilo and Stitch$^{\diamondsuit}$(4), Ice Age$^{\diamondsuit}$(5), Ring of Fire(4), Star Trek: Nemesis(3),	In America$^{\spadesuit}$(2), Super Size Me(2), A Very Long Engagement$^{\spadesuit}$(1), Bend It Like Beckham(2), 21 Grams$^{\spadesuit}$(1), Airplane II: The Sequel(4), Spun$^{\spadesuit}$(1), Fahrenheit 9/11(1)

No movie info used!

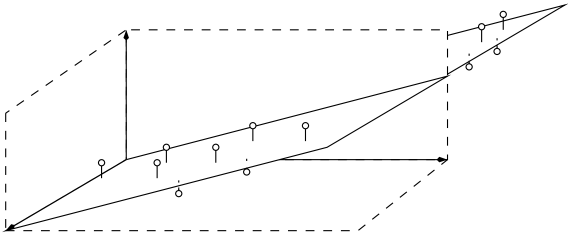
How did you do it?



$$\mathsf{R}_{ij} \sim \langle u_i, v_j \rangle + \varepsilon_{ij}$$

$$\Big\{(\mathrm{R}_{ij}, -v_j) \in \mathbb{R}^{r+1}, \qquad j \in \mathrm{WatchedBy}(i)\Big\} \subseteq \mathrm{Hyperplane}$$

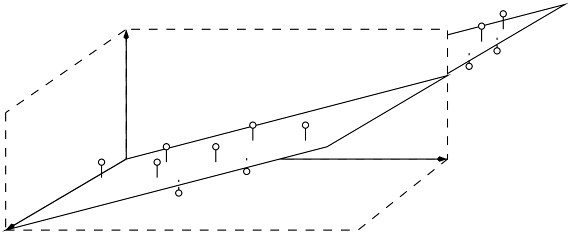
How did you do it?



$$R_{ij} \sim \langle u_i, v_j \rangle + \varepsilon_{ij}$$

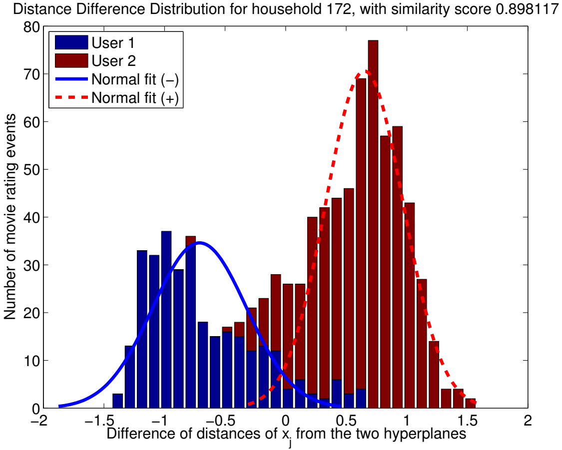
$$\Big\{(R_{ij}, -v_j) \in \mathbb{R}^{r+1}, \qquad j \in \text{WatchedBy}(i)\Big\} \subseteq \text{Hyperplane}$$

How did you do it? Subspace clustering

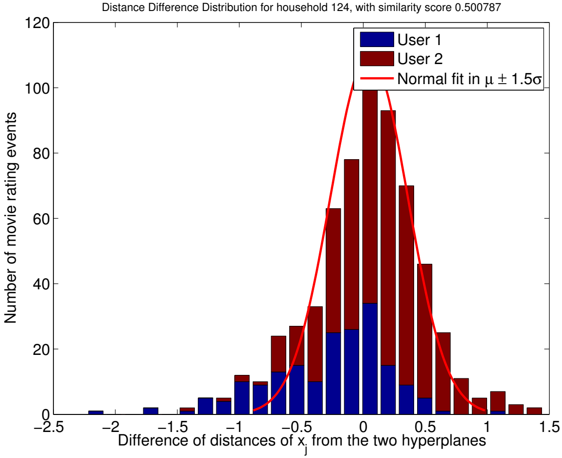


One user	$\rightarrow$	One hyperplane
Two users	$\rightarrow$	Two hyperplanes

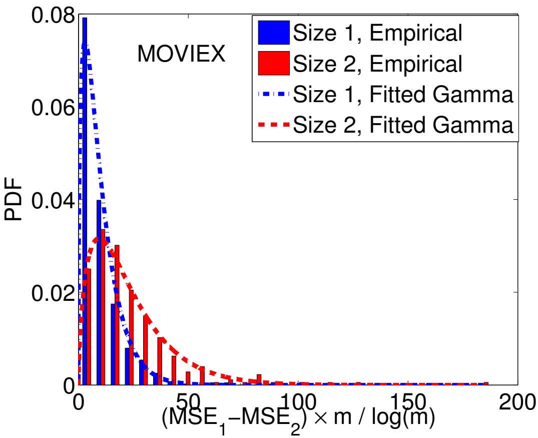
Example: A two-user household (CAMRA)



Example: Another two-user household (CAMRa)



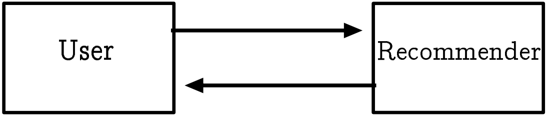
Classifying households



$$
\begin{aligned}
\mathrm{MSE}_1 &\quad\rightarrow\quad \text{MSE using one hyperplane} \\
\mathrm{MSE}_2 &\quad\rightarrow\quad \text{MSE using two hyperplanes}
\end{aligned}
$$

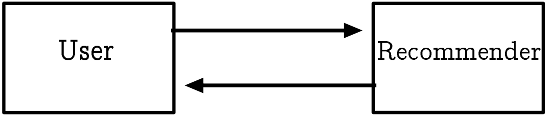
Challenge #2: Interactivity

We want to design the system



We took time out of the picture.

We want to design the system



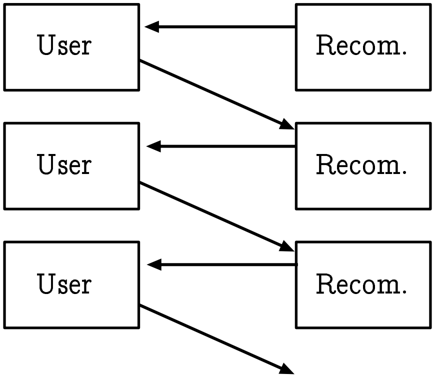
We took time out of the picture.

Interactive system

$t = 1$

$t = 2$

$t = 3$



Abstract from other users

At time t

- Recommender suggests movie v_t;
- User gives feedback R_t

Focus on user u

$$R_t \sim \langle u, v_t \rangle + \varepsilon_t$$

Abstract from other users

At time t

- Recommender suggests movie v_t;
- User gives feedback R_t

Focus on user u

$$\mathsf{R}_t \sim \langle u, v_t \rangle + \varepsilon_t$$

Linear bandits

Decision:

$$v_t$$

Observations:

$$\mathsf{R}_t \sim \langle u, v_t \rangle + \varepsilon_t$$

Reward:

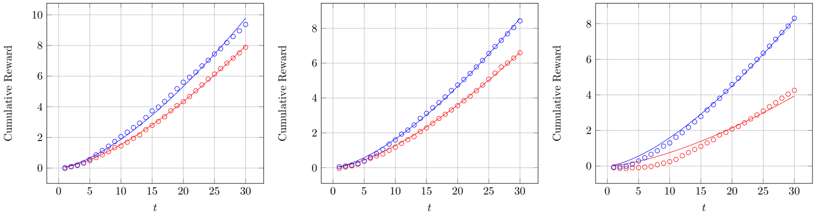
$$\mathcal{O}_t = \sum_{\ell=1}^{t} \langle u, v_\ell \rangle$$

[Rusmevichientong, Tsitsiklis, 2008]

Important differences

- Number of observations $\sim$ Dimensions
- Cannot explore completely at random.

Simulating an interactive system (Netflix data)



- 3 ‘typical’ users
- Constant-optimal policy

Conclusion

Conclusion

- A crucial technology for modern information networks.
- Only scratched the surface.

Thanks!

Conclusion

- A crucial technology for modern information networks.
- Only scratched the surface.

Thanks!