Big Data is Low Rank

Madeleine Udell

Operations Research and Information Engineering
Cornell University

UC Davis, 11/5/2018

Data table

age	gender	state	income	education	$\cdots$
29	F	CT	$53,000	college	$\cdots$
57	?	NY	$19,000	high school	$\cdots$
?	M	CA	$102,000	masters	$\cdots$
41	F	NV	$23,000	?	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$		

- detect demographic groups?
- find typical responses?
- identify related features?
- impute missing entries?

Data table

m examples (patients, respondents, households, assets)
n features (tests, questions, sensors, times)

$$\begin{bmatrix} & & \\ & A & \\ & & \end{bmatrix} = \begin{bmatrix} A_{11} & \cdots & A_{1n} \\ \vdots & \ddots & \vdots \\ A_{m1} & \cdots & A_{mn} \end{bmatrix}$$

- ith row of A is feature vector for ith example
- jth column of A gives values for jth feature across all examples

Low rank model

given: $m \times n$ data table A, $k \ll m, n$
find: $X \in \mathbf{R}^{m \times k}$, $Y \in \mathbf{R}^{k \times n}$ for which

$$\begin{bmatrix} \\ X \\ \\ \end{bmatrix} \begin{bmatrix} & Y & \end{bmatrix} \approx \begin{bmatrix} & & \\ & A & \\ & & \end{bmatrix}$$

i.e., $x_i y_j \approx A_{ij}$, where

$$\begin{bmatrix} \\ X \\ \\ \end{bmatrix} = \begin{bmatrix} \text{—} x_1 \text{—} \\ \vdots \\ \text{—} x_m \text{—} \end{bmatrix} \qquad \begin{bmatrix} & Y & \end{bmatrix} = \begin{bmatrix} | & & | \\ y_1 & \cdots & y_n \\ | & & | \end{bmatrix}$$

interpretation:

- X and Y are (compressed) representation of A
- $x_i^T \in \mathbf{R}^k$ is a point associated with example i
- $y_j \in \mathbf{R}^k$ is a point associated with feature j
- inner product $x_i y_j$ approximates A_{ij}

Why?

- reduce storage; speed transmission
- understand (visualize, cluster)
- remove noise
- infer missing data
- simplify data processing

Outline

PCA

Generalized low rank models

Applications

- Impute missing data
- Dimensionality reduction
- Causal inference
- Automatic machine learning

Why low rank?

Principal components analysis

PCA: for $A \in \mathbf{R}^{m \times n}$,

$$\text{minimize} \quad \|A - XY\|_F^2 = \sum_{i=1}^m \sum_{j=1}^n (A_{ij} - x_i y_j)^2$$

with variables $X \in \mathbf{R}^{m \times k}$, $Y \in \mathbf{R}^{k \times n}$

- old roots [Pearson 1901, Hotelling 1933]
- least squares low rank fitting
- (analytical) solution via SVD of $A = U \Sigma V^T$
- (numerical) solution via alternating minimization

Outline

PCA

Generalized low rank models

Applications

- Impute missing data
- Dimensionality reduction
- Causal inference
- Automatic machine learning

Why low rank?

Generalized low rank model

$$\text{minimize} \quad \sum_{(i,j)\in\Omega} L_j(x_i y_j, A_{ij}) + \sum_{i=1}^{m} r_i(x_i) + \sum_{j=1}^{n} \tilde{r}_j(y_j)$$

- loss functions L_j for each column
  - *e.g.*, different losses for reals, booleans, categoricals, ordinals, ...
- regularizers $r : \mathbf{R}^{1\times k} \to \mathbf{R}$, $\tilde{r} : \mathbf{R}^k \to \mathbf{R}$
- observe only $(i,j) \in \Omega$ (other entries are missing)

Note: can be NP-hard to optimize exactly...

Matrix completion

observe A_{ij} only for $(i,j) \in \Omega \subset \{1,\ldots,m\} \times \{1,\ldots,n\}$

$$\text{minimize} \quad \sum_{(i,j)\in\Omega}(A_{ij} - x_i y_j)^2 + \lambda \sum_{i=1}^m \|x_i\|^2 + \lambda \sum_{j=1}^n \|y_j\|^2$$

two regimes:

- **some entries missing:** don't waste data; "borrow strength" from entries that are *not* missing
- **most entries missing:** matrix completion still works!

Theorem ([Keshavan Montanari 2010])

If A has rank $k' \leq k$ and $|\Omega| = O(nk' \log n)$ (and A is incoherent and Ω is chosen UAR), then matrix completion exactly recovers the matrix A with high probability.

Maximum likelihood low rank estimation

Choose loss function to maximize (log) likelihood of observations:

- gaussian noise: $L(u, a) = (u - a)^2$
- laplacian (heavy-tailed) noise: $L(u, a) = |u - a|$
- gaussian + laplacian noise: $L(u, a) = \mathbf{huber}(u - a)$
- poisson (count) noise: $L(u, a) = \exp(u) - au + a \log a - a$
- bernoulli (coin toss) noise: $L(u, a) = \log(1 + \exp(-au))$

Maximum likelihood low rank estimation works

Theorem (Template)

If a number of samples $|\Omega| = O(n \log(n))$ *drawn UAR from matrix entries is observed according to a probabilistic model with parameter* Z*, the solution to (appropriately) regularized maximum likelihood estimation is close to the true* Z *with high probability.*

examples (not exhaustive!):

- additive gaussian noise [Candes Plan 2009]
- additive subgaussian noise [Keshavan Montanari Oh 2009]
- gaussian + laplacian noise [Xu Caramanis Sanghavi 2012]
- 0-1 (Bernoulli) observations [Davenport et al. 2012]
- entrywise exponential family distribution [Gunasekar Ravikumar Ghosh 2014]
- multinomial logit [Kallus Udell 2016]

Losses

$$\text{minimize} \quad \sum_{(i,j)\in\Omega} L_j(x_i y_j, A_{ij}) + \sum_{i=1}^m r_i(x_i) + \sum_{j=1}^n \tilde{r}_j(y_j)$$

choose loss $L : \mathbf{R} \times \mathcal{F} \to \mathbf{R}$ adapted to data type $\mathcal{F}$:

data type	**loss**	$L(u,a)$
real	quadratic	$(u-a)^2$
real	absolute value	$\lvert u-a \rvert$
real	huber	$\mathbf{huber}(u-a)$
boolean	hinge	$(1-ua)_+$
boolean	logistic	$\log(1+\exp(-au))$
integer	poisson	$\exp(u) - au + a\log a - a$
ordinal	ordinal hinge	$\sum_{a'=1}^{a-1}(1-u+a')_+ +$ $\sum_{a'=a+1}^{d}(1+u-a')_+$
categorical	one-vs-all	$(1-u_a)_+ + \sum_{a'\neq a}(1+u_{a'})_+$
categorical	multinomial logit	$\frac{\exp(u_a)}{\sum_{a'=1}^d \exp(u_{a'})}$

Regularizers

$$\text{minimize} \quad \sum_{(i,j)\in\Omega} L_j(x_i y_j, A_{ij}) + \sum_{i=1}^m r_i(x_i) + \sum_{j=1}^n \tilde{r}_j(y_j)$$

choose regularizers r, $\tilde{r}$ to impose structure:

structure	$r(x)$	$\tilde{r}(y)$
small	$\|x\|_2^2$	$\|y\|_2^2$
sparse	$\|x\|_1$	$\|y\|_1$
nonnegative	$\mathbf{1}(x \geq 0)$	$\mathbf{1}(y \geq 0)$
clustered	$\mathbf{1}(\mathbf{card}(x) = 1)$	0

Outline

PCA

Generalized low rank models

Applications

- Impute missing data
- Dimensionality reduction
- Causal inference
- Automatic machine learning

Why low rank?

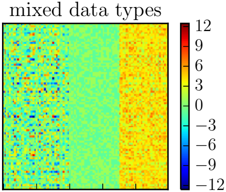
Impute missing data

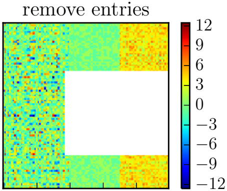
impute most likely true data $\hat{A}_{ij}$

$$\hat{A}_{ij} = \underset{a}{\operatorname{argmin}}\, L_j(x_i y_j, a)$$

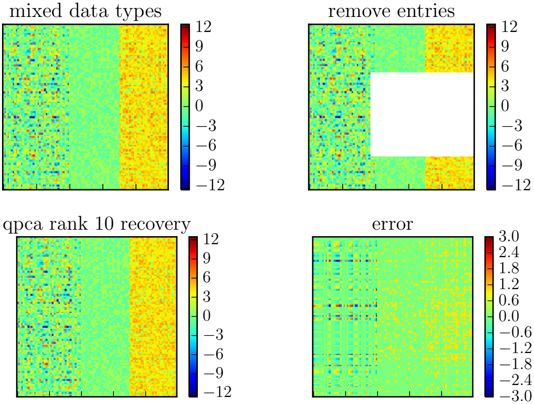
- implicit constraint: $\hat{A}_{ij} \in \mathcal{F}_j$
- when L_j is quadratic, ℓ_1, or Huber loss, then $\hat{A}_{ij} = x_i y_j$
- if $\mathcal{F} \neq \mathbf{R}$, $\operatorname{argmin}_a L_j(x_i y_j, a) \neq x_i y_j$
  - *e.g.*, for hinge loss $L(u, a) = (1 - ua)_+$, $\hat{A}_{ij} = \mathbf{sign}(x_i y_j)$

Impute heterogeneous data

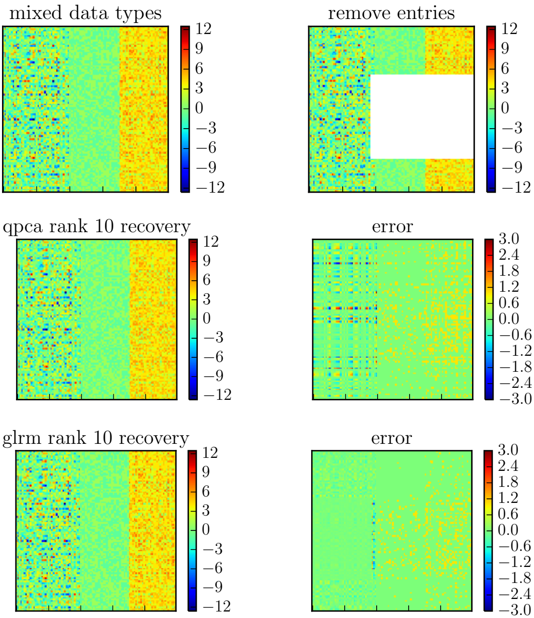




Impute heterogeneous data

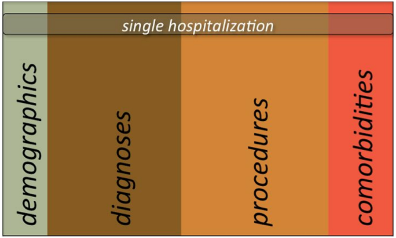


Impute heterogeneous data

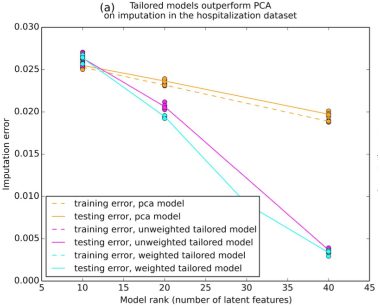


Hospitalizations are low rank

hospitalization data set



GLRM outperforms PCA



[Schuler Udell et al., 2016]

American community survey

2013 ACS:

- 3M respondents, 87 economic/demographic survey questions
  - income
  - cost of utilities (water, gas, electric)
  - weeks worked per year
  - hours worked per week
  - home ownership
  - looking for work
  - use foodstamps
  - education level
  - state of residence
  - ...
- 1/3 of responses missing

Fitting a GLRM to the ACS

- construct a rank 10 GLRM with loss functions respecting data types
  - huber for real values
  - hinge loss for booleans
  - ordinal hinge loss for ordinals
  - one-vs-all hinge loss for categoricals
- scale losses and regularizers by $1/\sigma_j^2$
- fit the GLRM

in 2 lines of code:

```
glrm, labels = GLRM(A, 10, scale = true)
X,Y = fit!(glrm)
```

American community survey

most similar features (in *demography space*):

- Alaska: Montana, North Dakota
- California: Illinois, cost of water
- Colorado: Oregon, Idaho
- Ohio: Indiana, Michigan
- Pennsylvania: Massachusetts, New Jersey
- Virginia: Maryland, Connecticut
- Hours worked: weeks worked, education

Low rank models for dimensionality reduction[1]

U.S. Wage & Hour Division (WHD) compliance actions:

company	zip	violations	$\cdots$
Holiday Inn	14850	109	$\cdots$
Moosewood Restaurant	14850	0	$\cdots$
Cornell Orchards	14850	0	$\cdots$
Lakeside Nursing Home	14850	53	$\cdots$
$\vdots$	$\vdots$	$\vdots$	

- 208,806 rows (cases) $\times$ 252 columns (violation info)
- 32,989 zip codes...

[1] labor law violation demo: `https://github.com/h2oai/h2o-3/blob/master/h2o-r/demos/rdemo.census.labor.violations.large.R`

Low rank models for dimensionality reduction

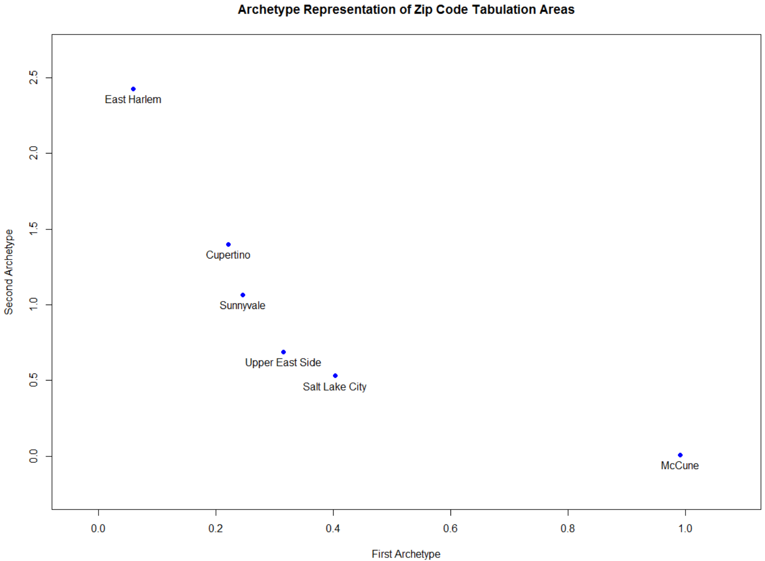
ACS demographic data:

zip	unemployment	mean income	$\cdots$
94305	12%	\$47,000	$\cdots$
06511	19%	\$32,000	$\cdots$
60647	23%	\$23,000	$\cdots$
94121	4%	\$178,000	$\cdots$
$\vdots$	$\vdots$	$\vdots$	

- 32,989 rows (zip codes) $\times$ 150 columns (demographic info)
- GLRM embeds zip codes into (low dimensional) *demography space*

Low rank models for dimensionality reduction

Zip code features:



Low rank models for dimensionality reduction

build 3 sets of features to predict violations:

- categorical: expand zip code to categorical variable
- concatenate: join tables on zip
- GLRM: replace zip code by low dimensional zip code features

fit a supervised (deep learning) model:

method	train error	test error	runtime
categorical	0.2091690	0.2173612	23.7600000
concatenate	0.2258872	0.2515906	4.4700000
GLRM	0.1790884	0.1933637	4.3600000

Causal inference with messy data

Gender	State	Age	IQ
M	CT	29	110
?	NY	57	?
F	?	?	108
F	CA	41	99
?	NV	50	?

fit $\longrightarrow$

×	×
×	×
×	×
×	×
×	×

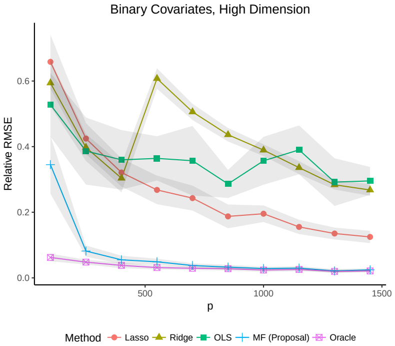
×	×	×	×
×	×	×	×

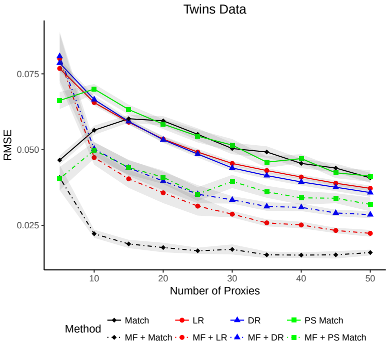
$\longleftarrow$ Estimate treatment effect of college

Income	College
5500	1
3000	0
5800	1
3300	0
4000	0

[Kallus Mao Udell, 2018]

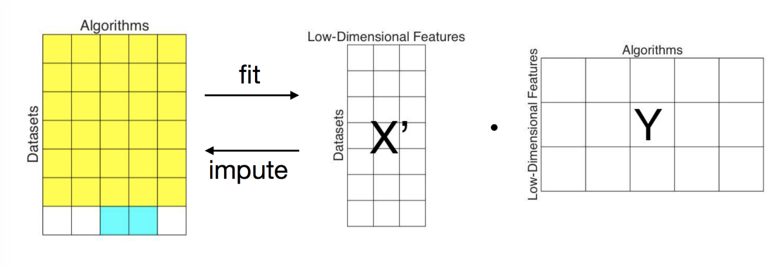
Matrix completion reduces error





[Kallus Mao Udell, 2018]

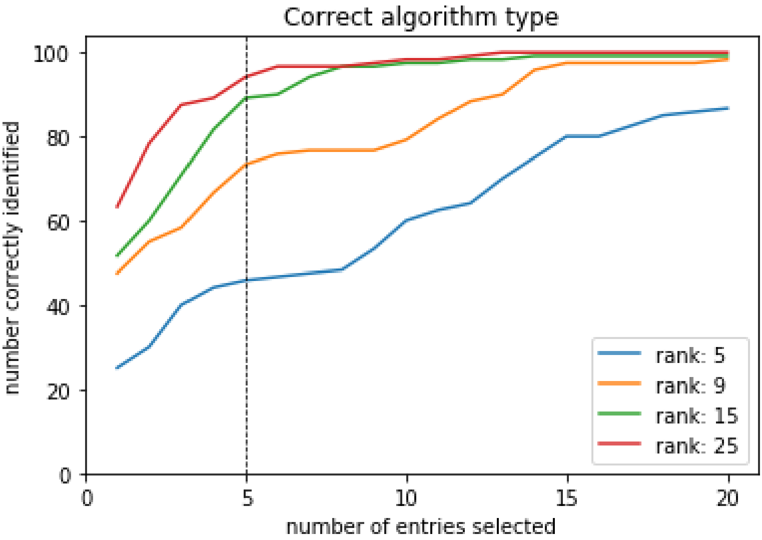
Low rank method for automatic machine learning



- yellow blocks are fully observed: results of each algorithm on each training dataset
- last row is a new dataset
- blue blocks are observed results of algorithms on new dataset
- white blocks are unknown entries
- fit a low rank model to impute white blocks

[Yang Akimoto Udell, 2018]

Low rank fit correctly identifies best algorithm type



[Yang Akimoto Udell, 2018]

Experiment design for timely model selection

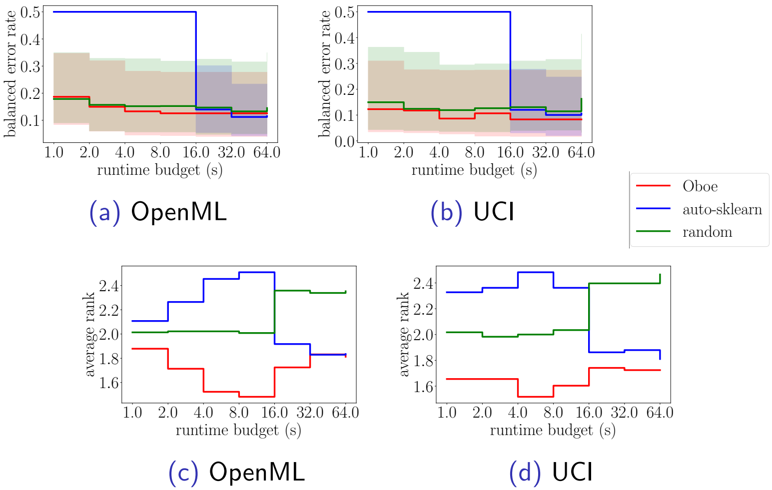
Which algorithms to use to predict performance?

$$\begin{aligned} \underset{v_j}{\text{maximize}} \quad & \log\det\left(\sum_{j=1}^{n} v_j y_j y_j^T\right) \\ \text{subject to} \quad & \sum_{j=1}^{n} v_j \hat{t}_j \leq \tau \\ & v_j \in [0,1] \qquad \forall j \in [n]. \end{aligned}$$

- $\hat{t}_j$: estimated runtime of each machine learning model
- τ: runtime budget

[Yang Akimoto Udell, 2018]

Oboe: Time-constrained AutoML



(a) OpenML (b) UCI

(c) OpenML (d) UCI

Figure: In 1a and 1b, shaded area = 75th–25th percentile.
In 1c and 1d, rank 1 is best and 3 is worst.

Outline

PCA

Generalized low rank models

Applications

- Impute missing data
- Dimensionality reduction
- Causal inference
- Automatic machine learning

Why low rank?

Latent variable models

Suppose $A \in \mathbf{R}^{m \times n}$ generated by a latent variable model (LVM):

- $\alpha_i \sim \mathcal{A}$ iid, $i = 1, \ldots, m$
- $\beta_j \sim \mathcal{B}$ iid, $j = 1, \ldots, n$
- $A_{ij} = g(\alpha_i, \beta_j)$

Latent variable models: examples

inner product:

- $\alpha_i \sim \mathcal{A} \subseteq \mathbf{R}^k$ iid, $i = 1, \ldots, m$
- $\beta_j \sim \mathcal{B} \subseteq \mathbf{R}^k$ iid, $j = 1, \ldots, n$
- $A_{ij} = \alpha_i^T \beta_j$

Latent variable models: examples

inner product:

- $\alpha_i \sim \mathcal{A} \subseteq \mathbf{R}^k$ iid, $i = 1, \ldots, m$
- $\beta_j \sim \mathcal{B} \subseteq \mathbf{R}^k$ iid, $j = 1, \ldots, n$
- $A_{ij} = \alpha_i^T \beta_j$
- rank of A?

Latent variable models: examples

inner product:

- $\alpha_i \sim \mathcal{A} \subseteq \mathbf{R}^k$ iid, $i = 1, \ldots, m$
- $\beta_j \sim \mathcal{B} \subseteq \mathbf{R}^k$ iid, $j = 1, \ldots, n$
- $A_{ij} = \alpha_i^T \beta_j$
- rank of A? k

Latent variable models: examples

inner product:

- $\alpha_i \sim \mathcal{A} \subseteq \mathbf{R}^k$ iid, $i = 1, \ldots, m$
- $\beta_j \sim \mathcal{B} \subseteq \mathbf{R}^k$ iid, $j = 1, \ldots, n$
- $A_{ij} = \alpha_i^T \beta_j$
- rank of A? k

univariate:

- $\alpha_i \sim \mathrm{Unif}(0, 1)$ iid, $i = 1, \ldots, m$
- $\beta_j \sim \mathrm{Unif}(0, 1)$ iid, $j = 1, \ldots, n$
- $A_{ij} = g(\alpha_i, \beta_j)$

Latent variable models: examples

inner product:

- $\alpha_i \sim \mathcal{A} \subseteq \mathbf{R}^k$ iid, $i = 1, \ldots, m$
- $\beta_j \sim \mathcal{B} \subseteq \mathbf{R}^k$ iid, $j = 1, \ldots, n$
- $A_{ij} = \alpha_i^T \beta_j$
- rank of A? k

univariate:

- $\alpha_i \sim \mathrm{Unif}(0, 1)$ iid, $i = 1, \ldots, m$
- $\beta_j \sim \mathrm{Unif}(0, 1)$ iid, $j = 1, \ldots, n$
- $A_{ij} = g(\alpha_i, \beta_j)$
- rank of A?

Latent variable models: examples

inner product:

- $\alpha_i \sim \mathcal{A} \subseteq \mathbf{R}^k$ iid, $i = 1, \ldots, m$
- $\beta_j \sim \mathcal{B} \subseteq \mathbf{R}^k$ iid, $j = 1, \ldots, n$
- $A_{ij} = \alpha_i^T \beta_j$
- rank of A? k

univariate:

- $\alpha_i \sim \mathrm{Unif}(0, 1)$ iid, $i = 1, \ldots, m$
- $\beta_j \sim \mathrm{Unif}(0, 1)$ iid, $j = 1, \ldots, n$
- $A_{ij} = g(\alpha_i, \beta_j)$
- rank of A?
  - can be large!
  - if g is analytic with $\sup_{x \in \mathbf{R}} |g^{(d)}(x)| \leq M$, entrywise ϵ-approximation to A has rank $\mathcal{O}(\log(1/\epsilon))$

Nice latent variable models

We say LVM is *nice* if

- distributions $\mathcal{A}$ and $\mathcal{B}$ have bounded support
- g is piecewise analytic and on each piece: for some $M \in \mathbf{R}$,

$$\|D^\mu g(\alpha, \beta)\| \leq CM^{|\mu|}\|g\|.$$

($\|g\| = \sup_{x \in \mathbf{dom}\, g} g(x)$ is sup norm.)

Examples: $g(\alpha, \beta) = \text{poly}(\alpha, \beta)$ or $g(\alpha, \beta) = \exp(\text{poly}(\alpha, \beta))$

Rank of nice latent variable models?

Question: Suppose $A \in \mathbf{R}^{m \times n}$ is drawn from a nice LVM.
How does rank of ϵ-approximation to A change with m and n?

$$\begin{array}{ll} \text{minimize} & \mathbf{Rank}(X) \\ \text{subject to} & \|X - A\|_\infty \leq \epsilon \end{array}$$

Rank of nice latent variable models?

Question: Suppose $A \in \mathbf{R}^{m \times n}$ is drawn from a nice LVM. How does rank of ϵ-approximation to A change with m and n?

$$\begin{array}{ll} \text{minimize} & \mathbf{Rank}(X) \\ \text{subject to} & \|X - A\|_\infty \leq \epsilon \end{array}$$

Answer: rank grows as $\mathcal{O}(\log(m+n)/\epsilon^2)$

Rank of nice latent variable models?

Question: Suppose $A \in \mathbf{R}^{m \times n}$ is drawn from a nice LVM.
How does rank of ϵ-approximation to A change with m and n?

$$\begin{array}{ll} \text{minimize} & \mathbf{Rank}(X) \\ \text{subject to} & \|X - A\|_\infty \leq \epsilon \end{array}$$

Answer: rank grows as $\mathcal{O}(\log(m+n)/\epsilon^2)$

Theorem (Udell and Townsend, 2017)

Nice latent variable models are of log rank.

Proof sketch

- For each α, expand g around $\beta = 0$ by its Taylor series

$$\begin{aligned} g(\alpha,\beta) - g(\alpha,0) &= \langle \nabla g(\alpha,0), \beta \rangle + \langle \nabla^2 g(\alpha,0), \beta\beta^\top \rangle + \ldots \\ &= \begin{bmatrix} \nabla g(\alpha,0) \\ \mathrm{vec}(\nabla^2 g(\alpha,0)) \\ \vdots \end{bmatrix}^\top \begin{bmatrix} \beta \\ \mathrm{vec}(\beta\beta^\top) \\ \vdots \end{bmatrix} \end{aligned}$$

  collect terms depending on α and on β
- Notice: very high dimensional inner product!
- apply Johnson Lindenstrauss lemma to reduce dimension

Johnson Lindenstrauss Lemma

Lemma (The Johnson–Lindenstrauss Lemma)

Consider $x_1, \ldots, x_n \in \mathbb{R}^N$. *Pick* $0 < \epsilon < 1$ *and set* $r = \lceil 8(\log n)/\epsilon^2 \rceil$.
There is a linear map $Q : \mathbb{R}^N \to \mathbb{R}^r$ *such that for all* $1 \leq i, j \leq n$,

$$(1-\epsilon)\|x_i - x_j\|^2 \leq \|Q(x_i - x_j)\|^2 \leq (1+\epsilon)\|x_i - x_j\|^2.$$

(Here, $\lceil a \rceil$ *is the smallest integer larger than a.)*

Johnson Lindenstrauss Lemma

Lemma (The Johnson–Lindenstrauss Lemma)

Consider $x_1, \ldots, x_n \in \mathbb{R}^N$. *Pick* $0 < \epsilon < 1$ *and set* $r = \lceil 8(\log n)/\epsilon^2 \rceil$.
There is a linear map $Q : \mathbb{R}^N \to \mathbb{R}^r$ *such that for all* $1 \leq i, j \leq n$,

$$(1-\epsilon)\|x_i - x_j\|^2 \leq \|Q(x_i - x_j)\|^2 \leq (1+\epsilon)\|x_i - x_j\|^2.$$

(Here, $\lceil a \rceil$ *is the smallest integer larger than* a*.)*

Lemma (Variant of the Johnson–Lindenstrauss Lemma)

Under the same assumptions, there is a linear map $Q : \mathbb{R}^N \to \mathbb{R}^r$ *such that for all* $1 \leq i, j \leq n$,

$$\left| x_i^T x_j - x_i^T Q^T Q x_j \right| \leq \epsilon \left(\|x_i\|^2 + \|x_j\|^2 - x_i^T x_j \right).$$

Summary

big data is low rank

- in social science
- in medicine
- in machine learning

we can exploit low rank to

- fill in missing data
- embed data in vector space
- infer causality
- choose machine learning models

References

- Generalized Low Rank Models. M. Udell, C. Horn, R. Zadeh, and S. Boyd. Foundations and Trends in Machine Learning, 2016.
- Revealed Preference at Scale: Learning Personalized Preferences from Assortment Choices. N. Kallus and M. Udell. EC 2016.
- Discovering Patient Phenotypes Using Generalized Low Rank Models. A. Schuler, V. Liu, J. Wan, A. Callahan, M. Udell, D. Stark, and N. Shah. Pacific Symposium on Biocomputing (PSB), 2016.
- Low rank models for high dimensional categorical variables: labor law demo. A. Fu and M. Udell. `https://github.com/h2oai/h2o-3/blob/master/h2o-r/demos/rdemo.census.labor.violations.large.R`
- Causal Inference with Noisy and Missing Covariates via Matrix Factorization. N. Kallus, X. Mao, and M. Udell. NIPS 2018.
- OBOE: Collaborative Filtering for AutoML Initialization. C. Yang, Y. Akimoto, D. Kim, and M. Udell.
- Why are Big Data Matrices Approximately Low Rank? M. Udell and A. Townsend. SIMODS, to appear.