CS838 Topics In Optimization

Convex Geometry in High-Dimensional Data Analysis

Ben Recht
Spring 2010

Logistics

- Class Tu-Th 1-2:15PM
- Office Hours CS4387, Tuesday 2:15-3:30

- Course webpage: http://pages.cs.wisc.edu/~brecht/cs838.html
- Readings will be posted here.

- **Scribing.** All students are required to scribe notes for at least one lecture. LaTeX template will be provided.

- **Project.** All students are required to prepare a 20-30 minute presentation on the themes of this course. This can be a literature review or an application of the course's techniques to your research.

Recommender Systems

amazon.com

NETFLIX

match.com

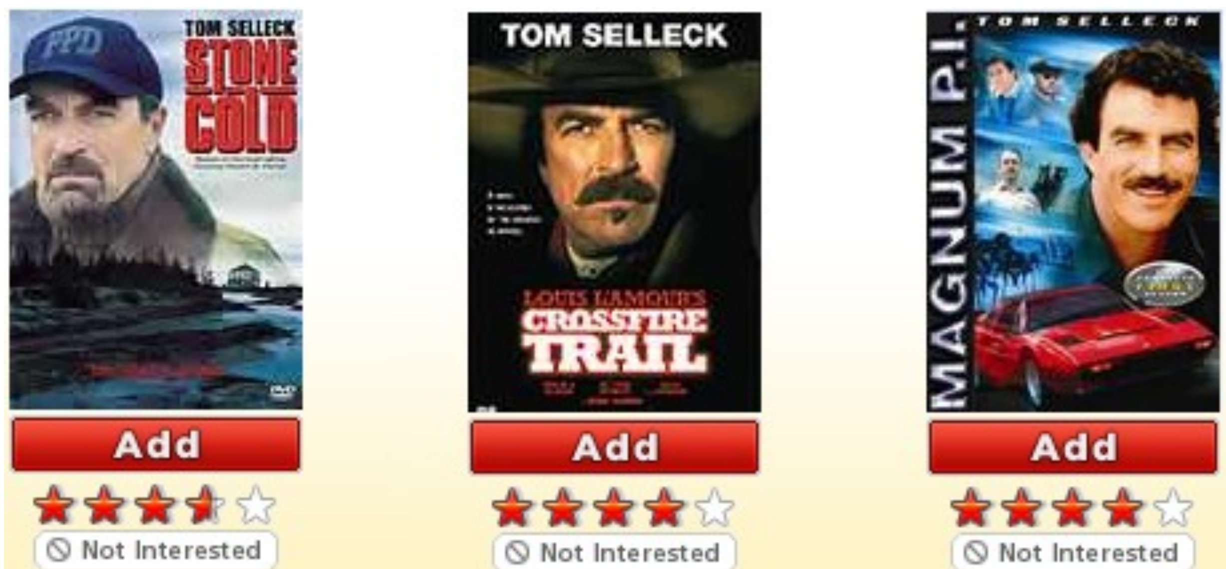
chemistry

eHarmony

Netflix

- Rate some movies…

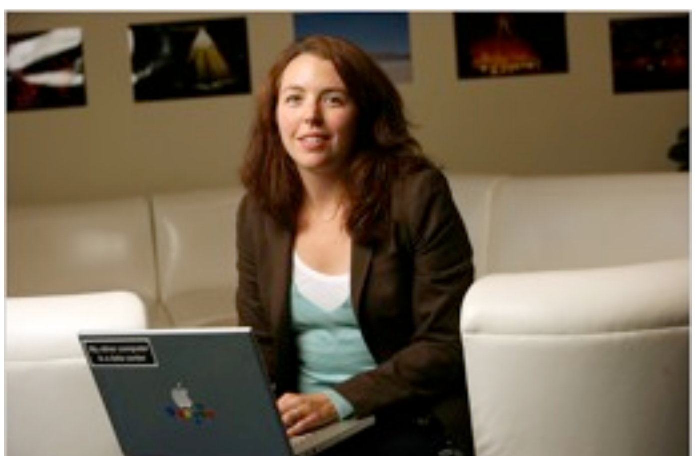
- Get some recommendations:



Netflix Prize

- One million big ones!

Data Sleuths in an Internet Age



HOME PAGE | TODAY'S PAPER | VIDEO | MOST POPULAR | TIMES TOPICS

The New York Times

Technology

WORLD | U.S. | N.Y. / REGION | BUSINESS | TECHNOLOGY | SCIENCE | HEALTH | SPORTS | OPINION

Search Technology

Inside Technology: Internet | Start-Ups | Business Computing

For Today's Graduate, Just One Word: Statistics

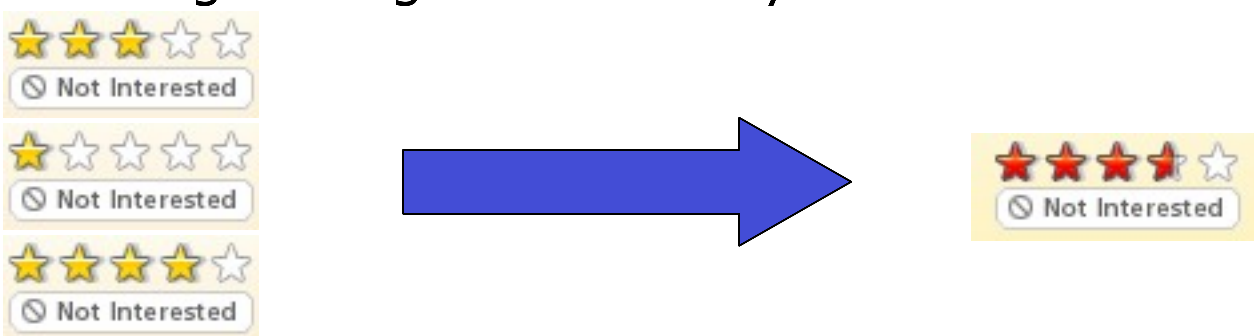
By STEVE LOHR
Published: August 5, 2009

MOUNTAIN VIEW, Calif. — At Harvard, Carrie Grimes majored in anthropology and archaeology and ventured to places like Honduras, where she studied Mayan settlement patterns by mapping where artifacts were found. But she was drawn to what she calls "all the computer and math stuff" that was part of the job.

SIGN IN TO RECOMMEND
TWITTER
COMMENTS (58)
E-MAIL
SEND TO PHONE

Netflix Prize

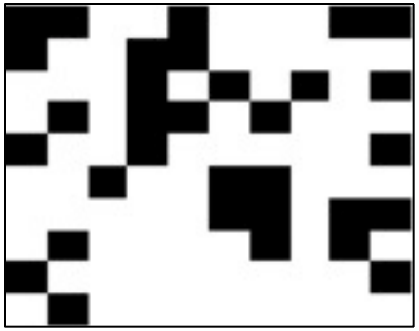
- One million big ones!
- Given 100 million ratings on a scale of 1 to 5, predict 3 million ratings to highest accuracy



- 17770 total movies x 480189 total users
- Over 8 billion total ratings
- How to fill in the blanks?

Abstract Setup: Matrix Completion

$\mathbf{X}$ =



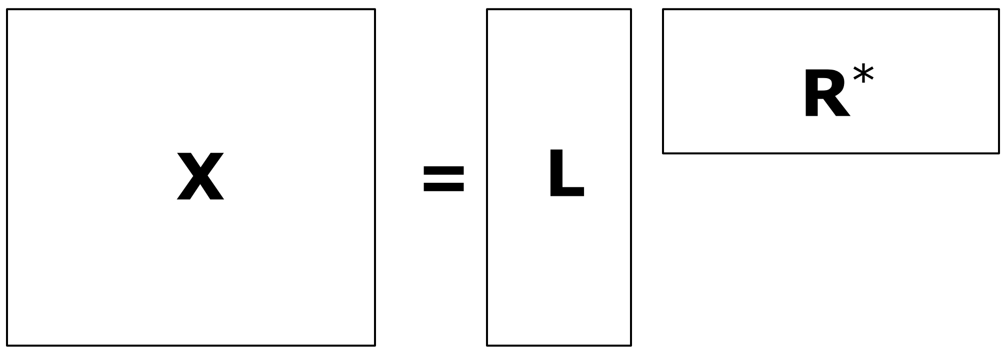
X_{ij} known for black cells

X_{ij} unknown for white cells

Rows index movies

Columns index users

- How do you fill in the missing data?



k x n   k x r   r x n

kn entries   r(k+n) entries

Netflix Prize - Dimensions

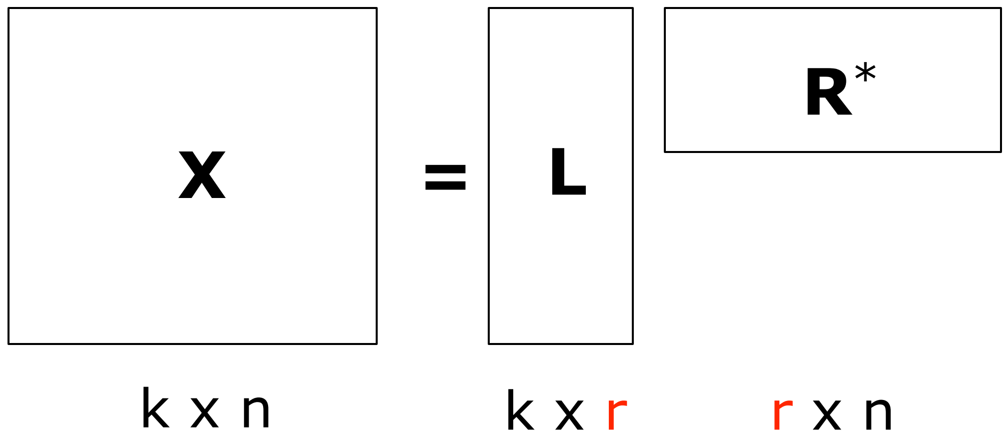
$$X = L \; R^*$$

$k \times n$ $\quad$ $k \times r$ $\quad$ $r \times n$

- k = Number of movies = 2×10^4
- n = Number of users = 5×10^5
- m = Number of Given Ratings = 10^8
- $kn \approx 10^{10}$

- For $r < 200$, $r(k+n) < 10^8$

Matrix Rank



- The rank of **X** is…

  the dimension of the span of the rows

  the dimension of the span of the columns

  the smallest number r such that there exists an k x r matrix **L** and an n x r matrix **R** with **X**=**LR***

Recommender Systems

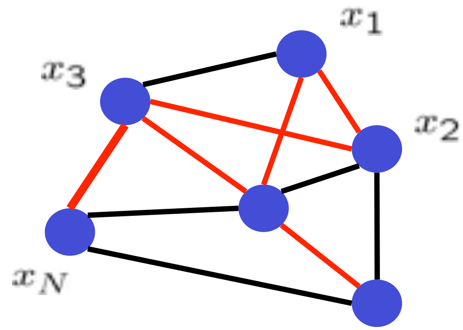
amazon.com

NETFLIX

match.com

chemistry

Euclidean Embedding

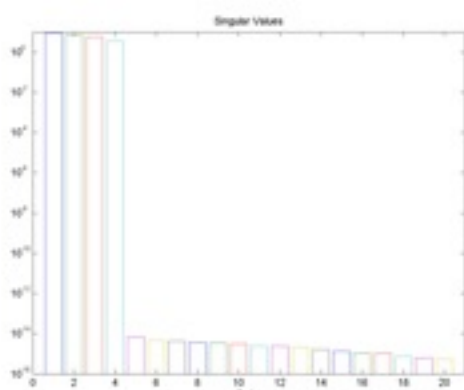


Multitask Learning

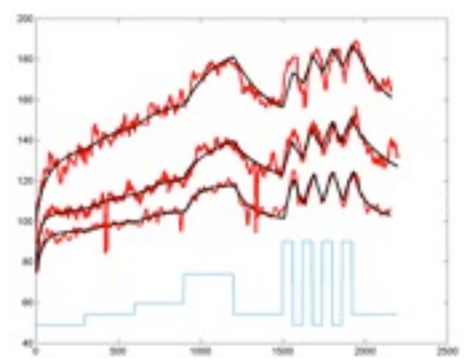
Rank of: Data Matrix

Gram Matrix

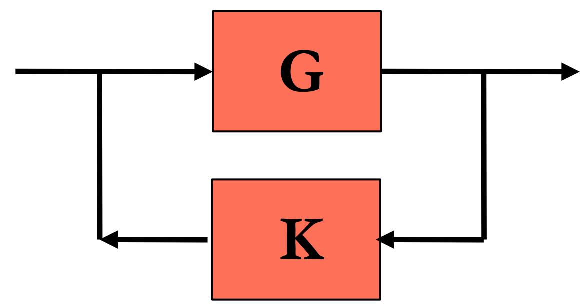
Matrix of Classifiers



Model Reduction



System Identification



Controller Design

Constraints involving the rank of the Hankel Operator, Matrix, or Singular Values

Affine Rank Minimization

- **PROBLEM:** Find the matrix of lowest rank that satisfies/approximates the underdetermined linear system

$$\mathcal{A}(\mathbf{X}) = \mathbf{b} \qquad \mathcal{A} : \mathbb{R}^{k \times n} \to \mathbb{R}^m$$

$$\begin{array}{ll} \text{minimize} & \operatorname{rank}(\mathbf{X}) \\ \text{subject to} & \mathcal{A}(\mathbf{X}) = \mathbf{b} \end{array}$$

- **NP-HARD:**
  - Reduce to finding solutions to polynomial systems
  - Hard to approximate
  - Exact algorithms are awful

Heuristic: Gradient Descent

$$\mathcal{F}(\mathbf{L},\mathbf{R}) = \sum_{i=1}^{k}\sum_{k=1}^{r} L_{ik}^2 + \sum_{j=1}^{n}\sum_{k=1}^{r} R_{jk}^2 + \lambda \sum_{i,j}\left(\sum_{k} L_{ik}R_{jk} - X_{ij}\right)^2$$

- Just run gradient descent to minimize $\mathcal{F}$
- λ determines tradeoff between satisfying constraints and the size of the factors

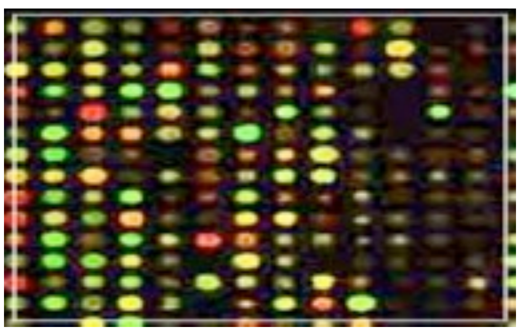
Netflix Prize

Leaderboard

Mixture of hundreds of models, including gradient descent

Gradient descent on low-rank parameterization

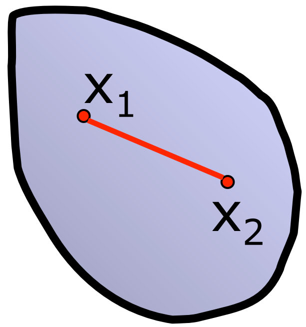
Rank	Team Name	Best Score	% Improvement	Last Submit Time
--	No Grand Prize candidates yet	--	--	--
Grand Prize - RMSE <= 0.8563				
--	No Progress Prize candidates yet	--	--	--
Progress Prize - RMSE <= 0.8625				
1	When Gravity and Dinosaurs Unite	0.8675	8.82	2008-03-01 07:03:35
2	BellKor	0.8682	8.75	2008-02-28 23:40:45
3	[illegible]	0.8708	8.47	2008-02-06 14:12:44
[illegible] Prize 2007 - RMSE = 0.8712 - Winning Team: KorBell				
4	KorBell	0.8712	8.43	2007-10-01 23:25:23
5	acmehill	0.8720	8.35	2008-03-02 05:08:12
6	Dan Tillberg	0.8727	8.27	2008-03-02 08:42:29
7	basho	0.8729	8.25	2007-11-24 14:27:00
8	Just a guy in a garage	0.8740	8.14	2008-02-06 12:16:40
9	BigChaos	0.8748	8.05	2008-03-01 17:26:06
10	Dinosaur Planet	0.8753	8.00	2007-10-04 04:56:45
⋮		⋮	⋮	⋮
50	amgl	0.8897	6.49	2007-12-23 18:44:03
51	Remco	0.8899	6.46	2007-04-04 06:16:56
52	mxlg	0.8900	6.45	2007-12-23 18:54:46
53	JustWithSVD	0.8900	6.45	2008-02-14 16:17:54
54	[illegible]	0.8900	6.45	2008-02-28 09:56:20
55	[illegible]	0.8901	6.44	2008-02-29 05:53:11
[illegible]	Bozo_The_Clown	0.8902	6.43	2007-09-06 17:24:48

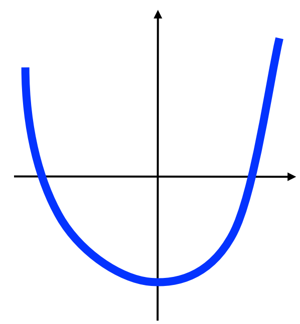
Complex Systems	Predictions	Structure
	 	**Rank**
		**Smoothness**
		**Sparsity**

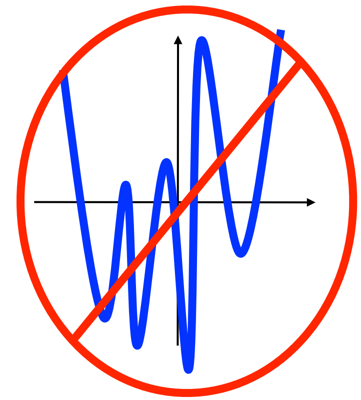
Modeling Simplicity: Strategy

- Find a “natural” ***convex*** heuristic

$$t\mathrm{x}_1 + (1-t)\mathrm{x}_2$$







- Use probabilistic analysis to prove the heuristic succeeds

- Provide efficient algorithms for solving the heuristic

Topics

- Sparsity
- Rank
- Smoothness

Themes

- Random Projections Preserve Geometry (*encoding*)
- Atomic Norms Recover Geometry (*decoding*)

Parsimonious Models

rank

$$x = \sum_{k=1}^{r} w_k \alpha_k$$

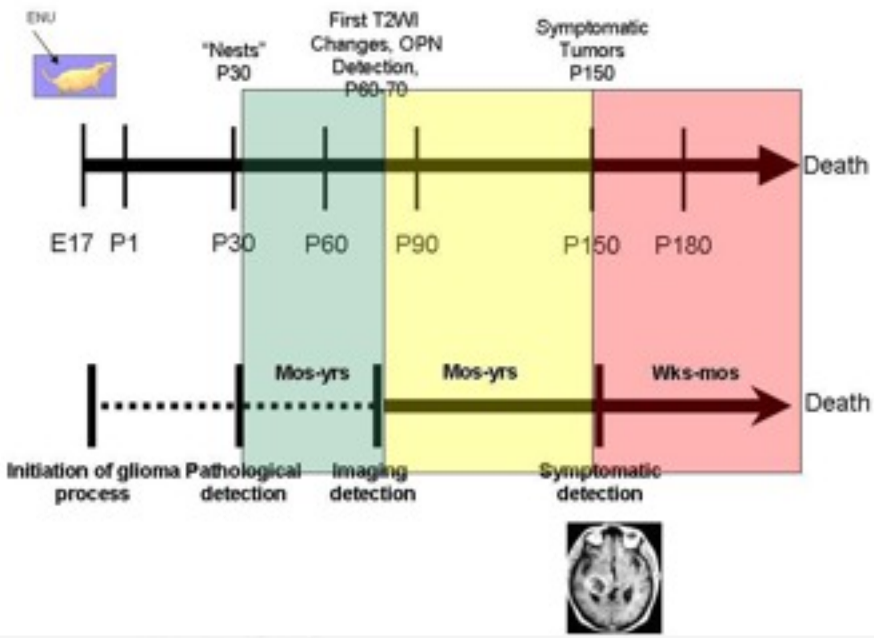
model weights atoms

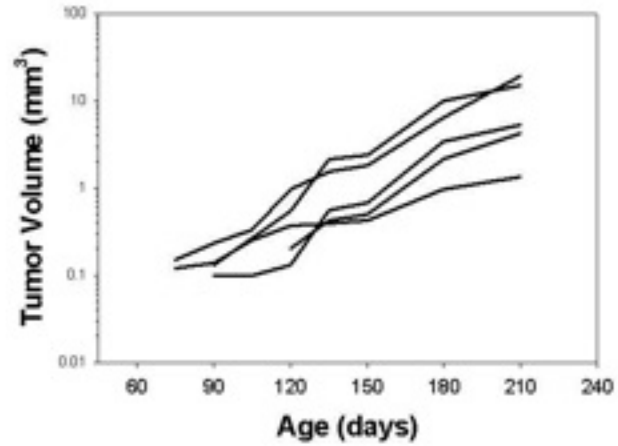
- Search for best linear combination of fewest atoms
- "rank" = fewest atoms needed to describe the model

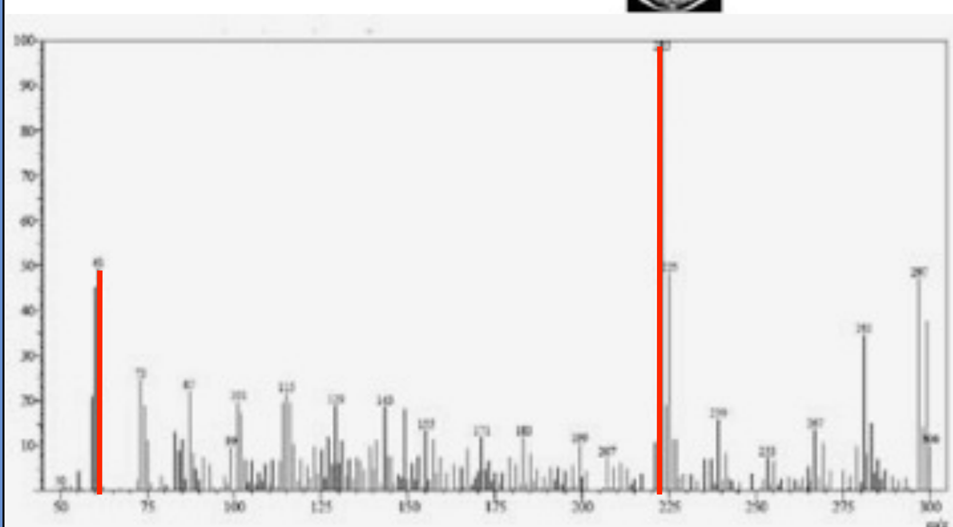
- "natural" heuristic is the *atomic norm*:

$$\|x\|_{\mathcal{A}} \equiv \inf \left\{ \sum_{k=1}^{r} |w_k| \ : \ x = \sum_{k=1}^{r} w_k \alpha_k \right\}$$

Mining for Biomarkers







- $n_{patients} << n_{peaks}$
- If very few are needed for diagnosis, search for a *sparse set of markers*
- l_1, LASSO, etc.

Topic 1: Cardinality/Sparsity

- Vector x has cardinality s if it has at most s nonzeros. (x is *s-sparse)*

$$x = \sum_{k=1}^{s} w_k e_{i_k}$$

- Atoms are a discrete set of orthogonal points
- Typical Atoms:
  - standard basis
  - Fourier basis
  - Wavelet basis

Cardinality Minimization

- **PROBLEM:** Find the vector of lowest cardinality that satisfies/approximates the underdetermined linear system

$$Ax = b \qquad A : \mathbb{R}^n \to \mathbb{R}^m$$

- **NP-HARD:**
  - Reduce to EXACT-COVER [Natarajan 1995]
  - Hard to approximate
  - Known exact algorithms require enumeration

Proposed Heuristic

Cardinality Minimization:

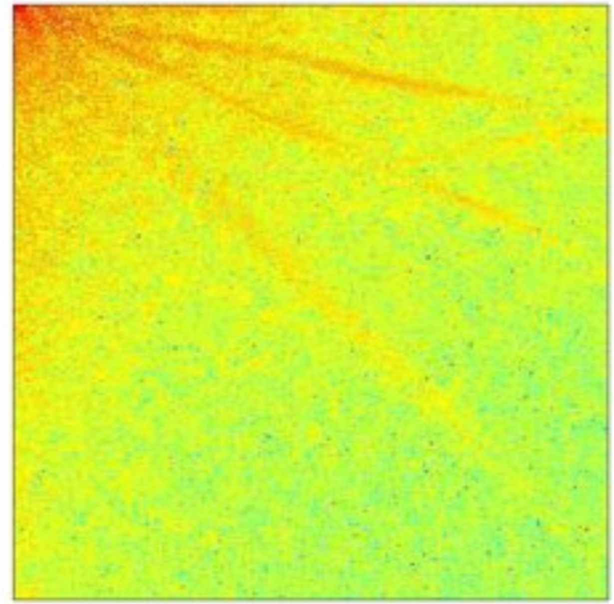
$$\begin{array}{ll} \text{minimize} & \operatorname{card}(x) \\ \text{subject to} & Ax = b \end{array}$$

Convex Relaxation:

$$\begin{array}{ll} \text{minimize} & \|x\|_1 = \sum_{i=1}^{D} |x_i| \\ \text{subject to} & Ax = b \end{array}$$

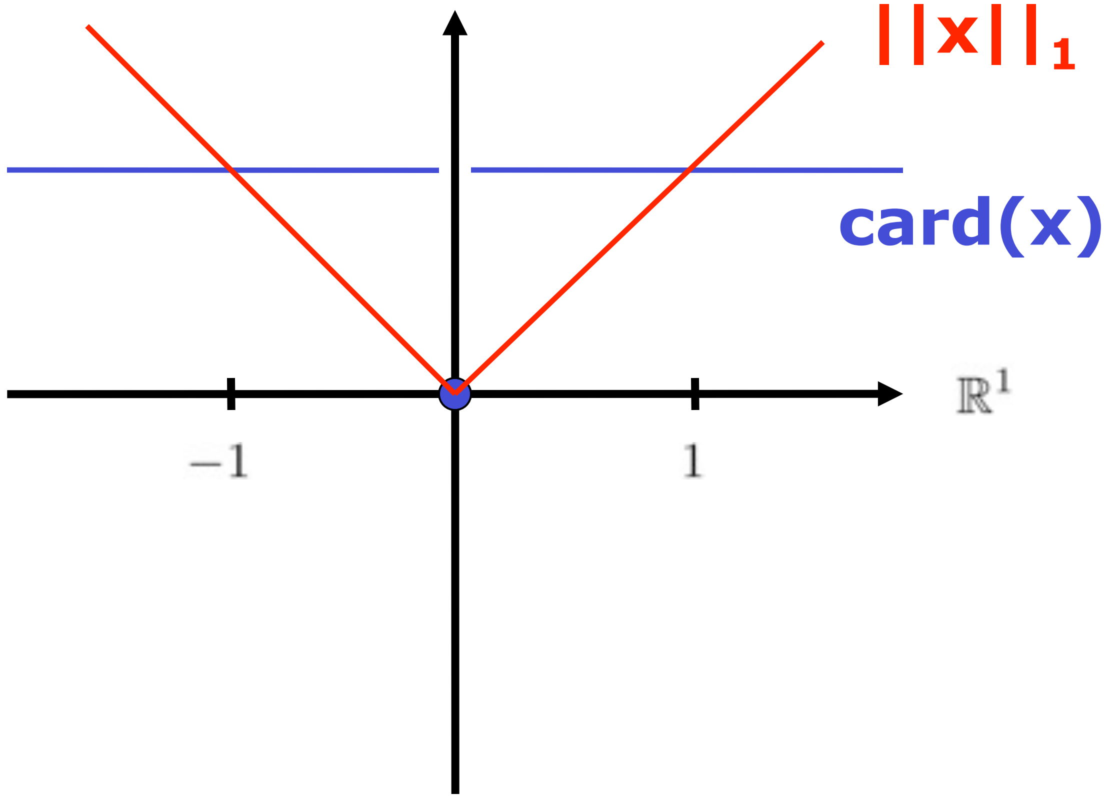
- Long history (back to geophysics in the 70s)
- Flurry of recent work characterizing success of this heuristic: Candès, Donoho, Romberg, Tao, Tropp, etc., etc…
- “Compressed Sensing”

Compressed Sensing



- Model: most of the energy is at low frequencies
- Basis for JPG compression
- Use the fact that the image is sparse in DCT/wavelet basis to reduce number of measurements required for signal acquisition.
- ***decode using l_1 minimization***

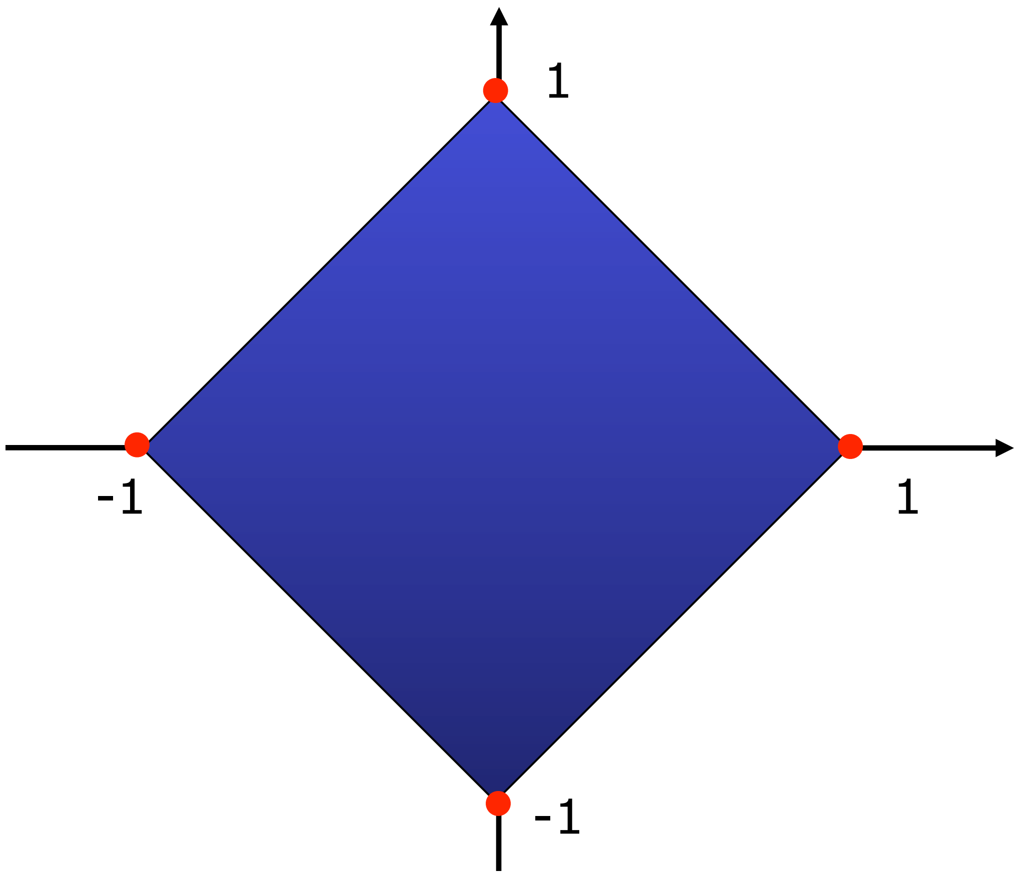
Why l_1 norm?



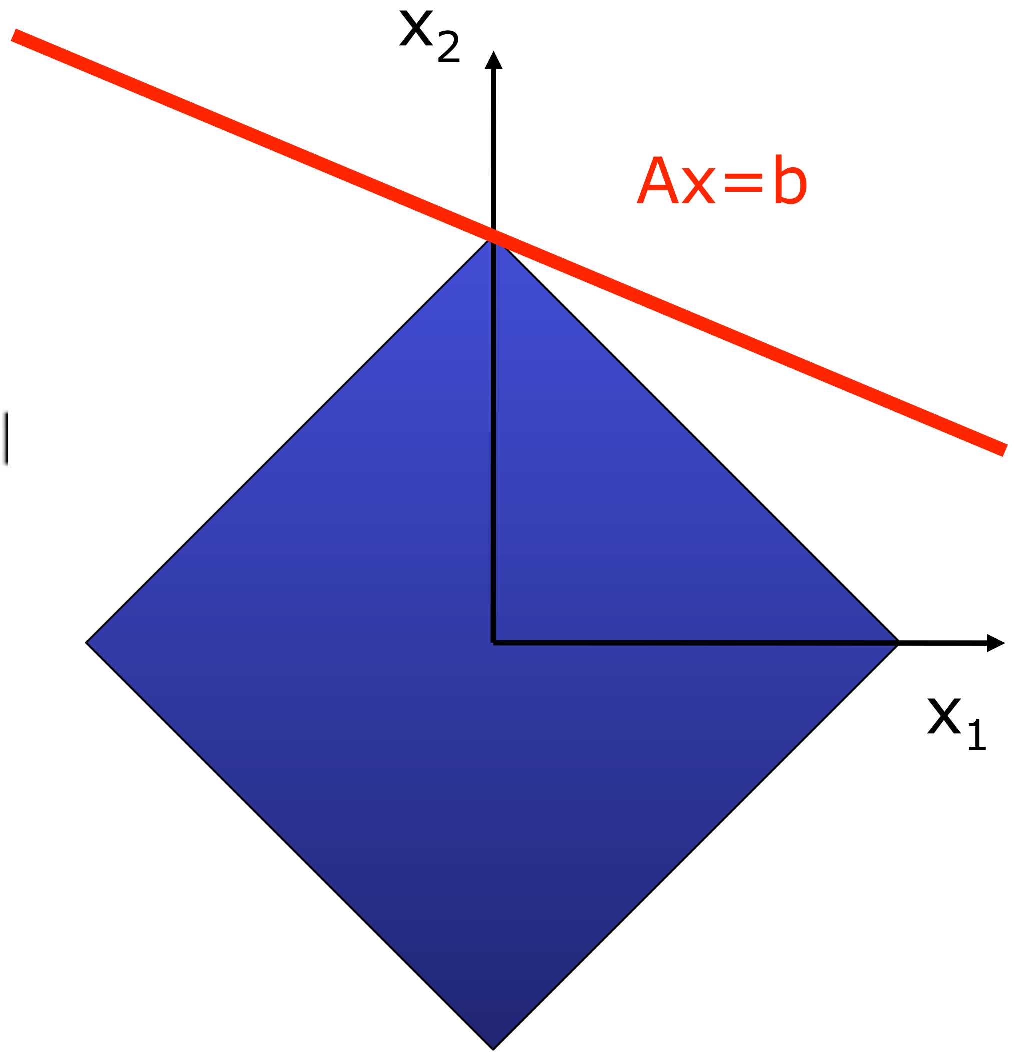
- 1-sparse vectors of Euclidean norm 1
- Convex hull is the unit ball of the l_1 norm

$$\{x \;:\; \|x\|_1 \leq 1\}$$

$$\|x\|_1 = \sum_{i=1}^{n} |x_i|$$



$$\begin{array}{ll} \text{minimize} & \|x\|_1 = \sum_{i=1}^{n} |x_i| \\ \text{subject to} & Ax = b \end{array}$$



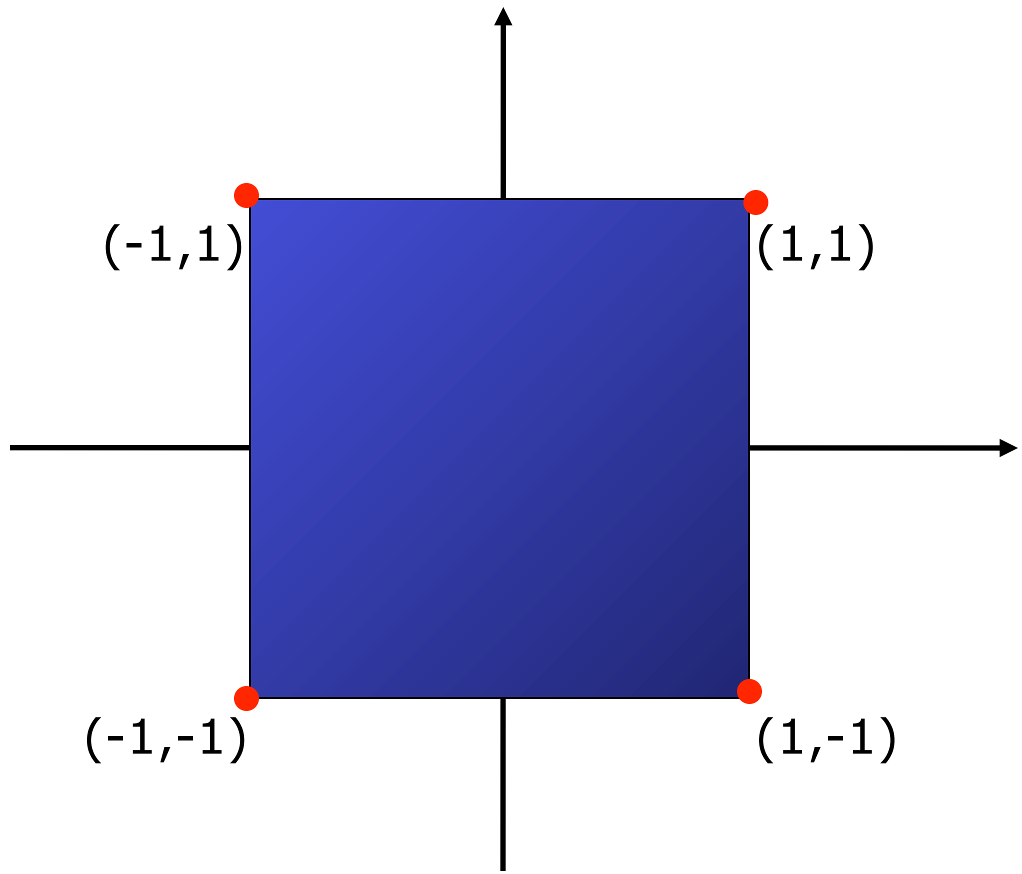
Integer Programming

- Integer solutions: all components of x are ±1

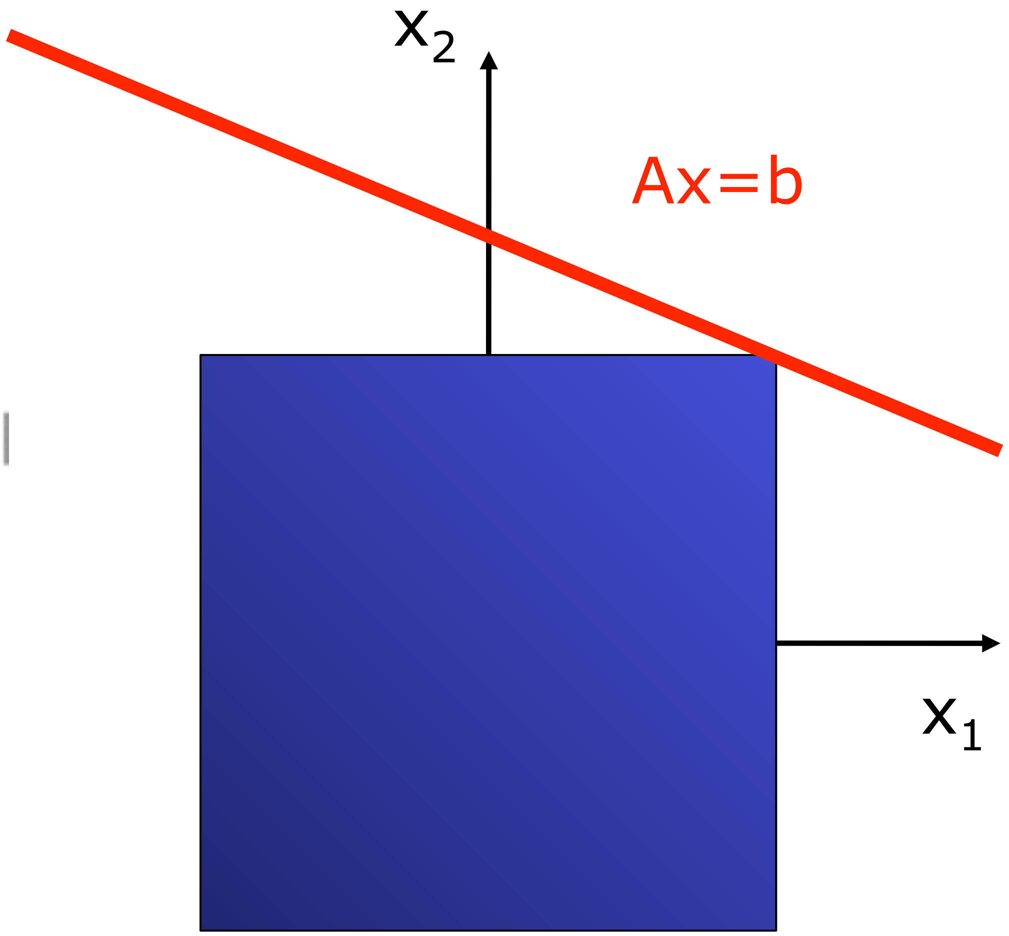
- Convex hull is the unit ball of the l_∞ norm

$$\{x \ : \ \|x\|_\infty \leq 1\}$$

$$\|x\|_\infty = \max_i |x_i|$$



$$\begin{array}{ll} \text{minimize} & \|x\|_\infty = \max_i |x_i| \\ \text{subject to} & Ax = b \end{array}$$



Cardinality/Sparsity

- How many samples are required to reconstruct sparse vectors?
- Relationship to coding theory
- When can we guarantee the l1 heuristic works?
- What are efficient ways to compute minimum l1 norm solutions?

Topic 2: (Matrix) Rank

- Matrix X has rank r if it has at most r nonzero singular values.

$$X = \sum_{j=1}^{r} \sigma_j u_j v_j^* = \sum_{j=1}^{r} \sigma_j A_j$$

- Atoms are the set of all rank one matrices
- Not a discrete set

Singular Value Decomposition (SVD)

- If **X** is a matrix of size k x n (k≤m) then there matrices **U** (k x k) and **V** (n x k) such that

$$\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^*$$

$$\mathbf{U}^*\mathbf{U} = I_m \qquad \mathbf{V}^*\mathbf{V} = I_m$$

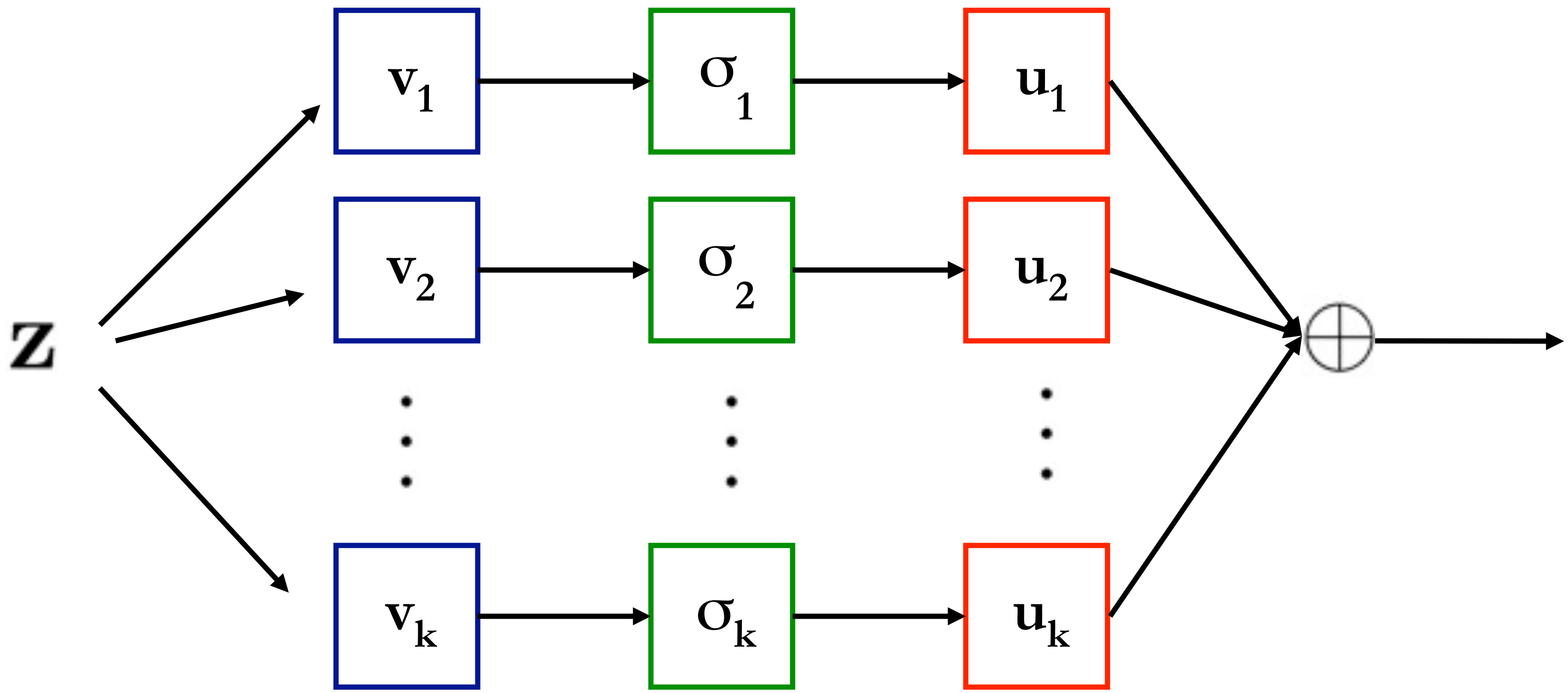
- $\sum$ a diagonal matrix, $\sigma_1 \geq \ldots \geq \sigma_k \geq 0$

- σ_i^2 is an eigenvalue of $\mathbf{XX}^*$. **U** are eigenvectors of $\mathbf{XX}^*$.

- **Fact:** If **X** has rank r, then **X** has only r non-zero singular values.

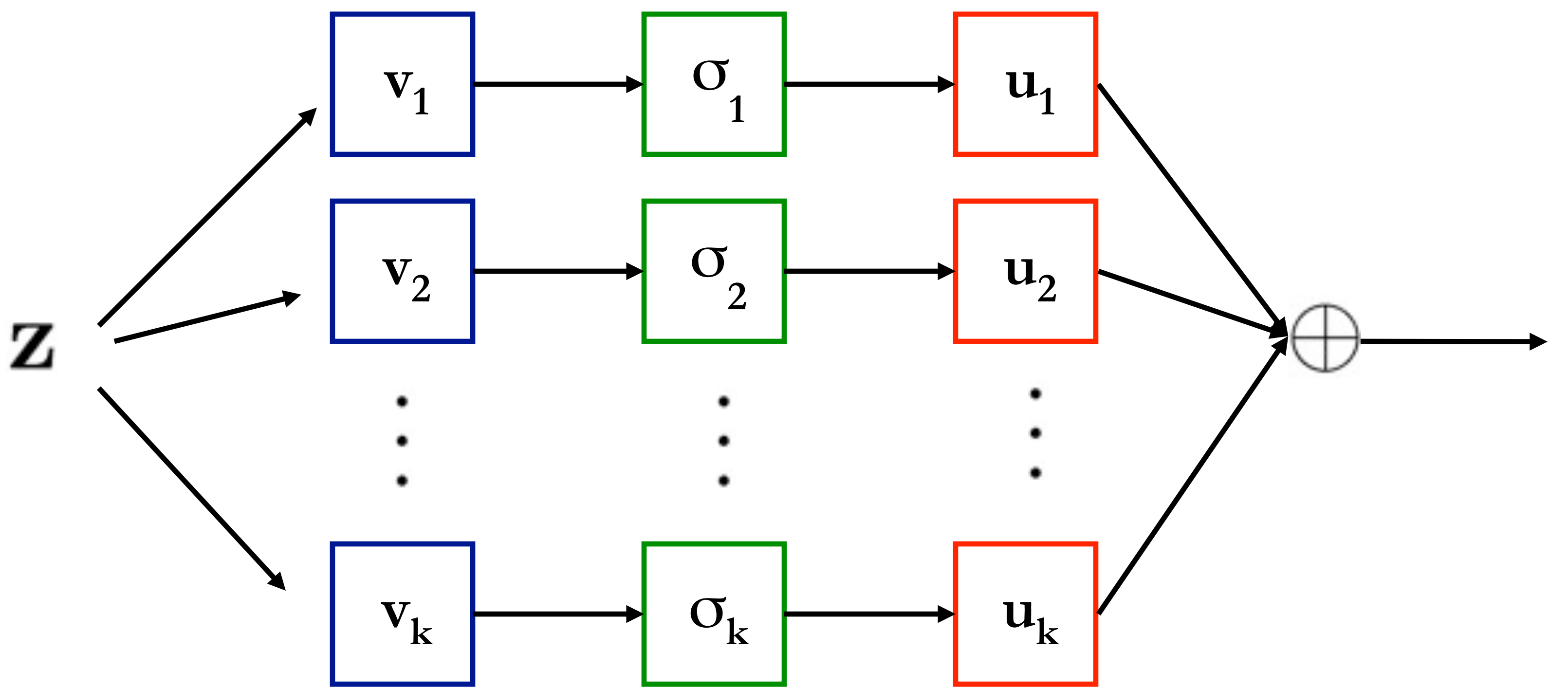
SVD = Filter Bank

- Multiply a vector $\mathbf{z}$ by $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^*$



Collaborative Filterings

- $\mathbf{Z}$ is a linear combination of eigenusers, $\mathbf{v}_1, \ldots, \mathbf{v}_k$.
- $\mathbf{u}_1, \ldots, \mathbf{u}_k$ are the eigenratings



Which Algorithm?

Affine Rank Minimization:

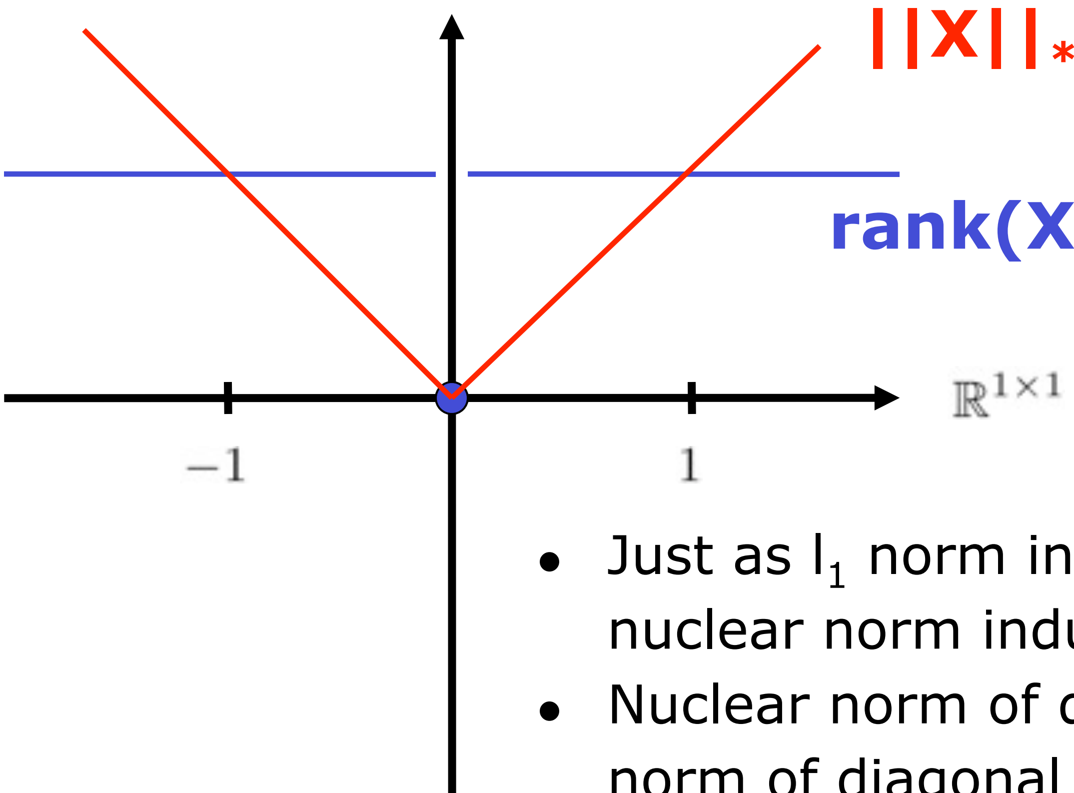
$$\begin{array}{ll} \text{minimize} & \operatorname{rank}(\mathbf{X}) \\ \text{subject to} & \mathcal{A}(\mathbf{X}) = \mathbf{b} \end{array}$$

Convex Relaxation:

$$\begin{array}{ll} \text{minimize} & \|\mathbf{X}\|_* = \sum_{i=1}^k \sigma_i(\mathbf{X}) \\ \text{subject to} & \mathcal{A}(\mathbf{X}) = \mathbf{b} \end{array}$$

- Proposed by Fazel (2002).
- Nuclear norm is the "numerical rank" in numerical analysis
- The "trace heuristic" from controls if **X** is p.s.d.

Why nuclear norm?



- Just as l_1 norm induces sparsity, nuclear norm induces low rank
- Nuclear norm of diagonal matrix = l_1 norm of diagonal

- 2x2 matrices
- plotted in 3d

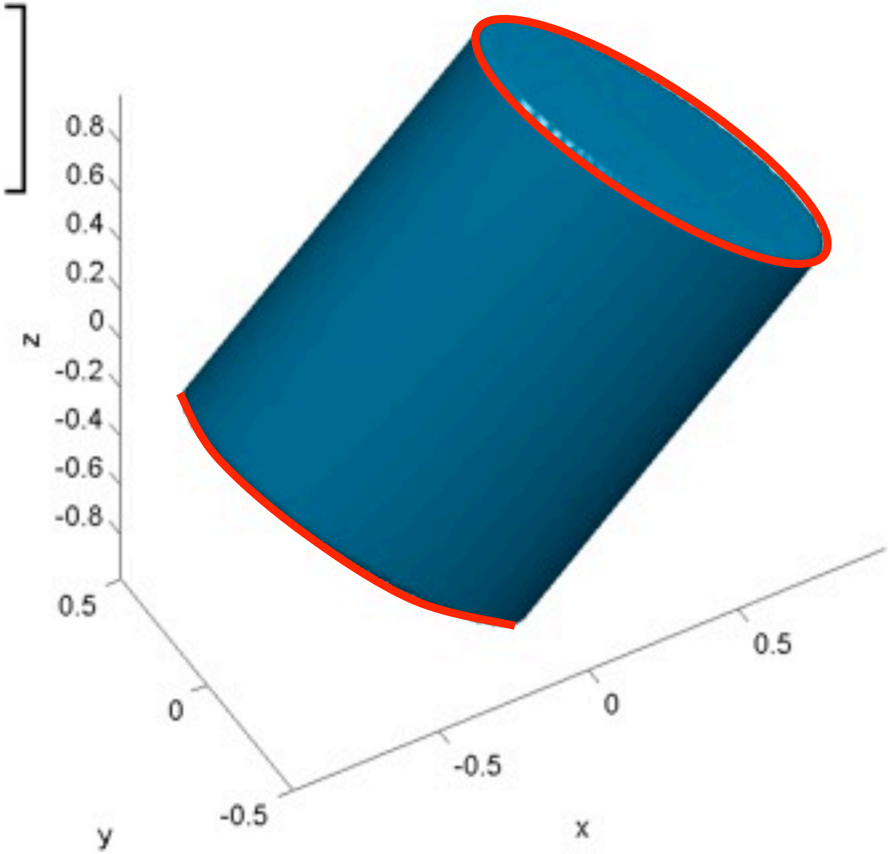
$$\begin{bmatrix} x & y \\ y & z \end{bmatrix}$$

rank 1

$x^2 + z^2 + 2y^2 = 1$

Convex hull:

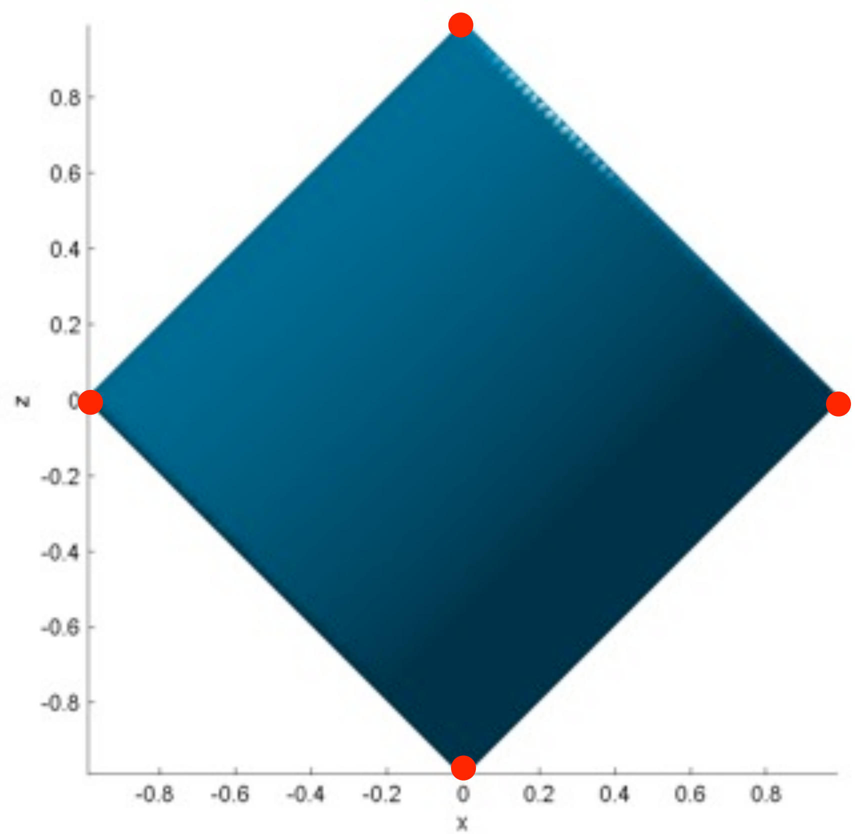
$$\{X \ : \ \|X\|_* \leq 1\}$$



- 2x2 matrices
- plotted in 3d

$$\left\| \begin{bmatrix} x & 0 \\ 0 & z \end{bmatrix} \right\|_* \leq 1$$

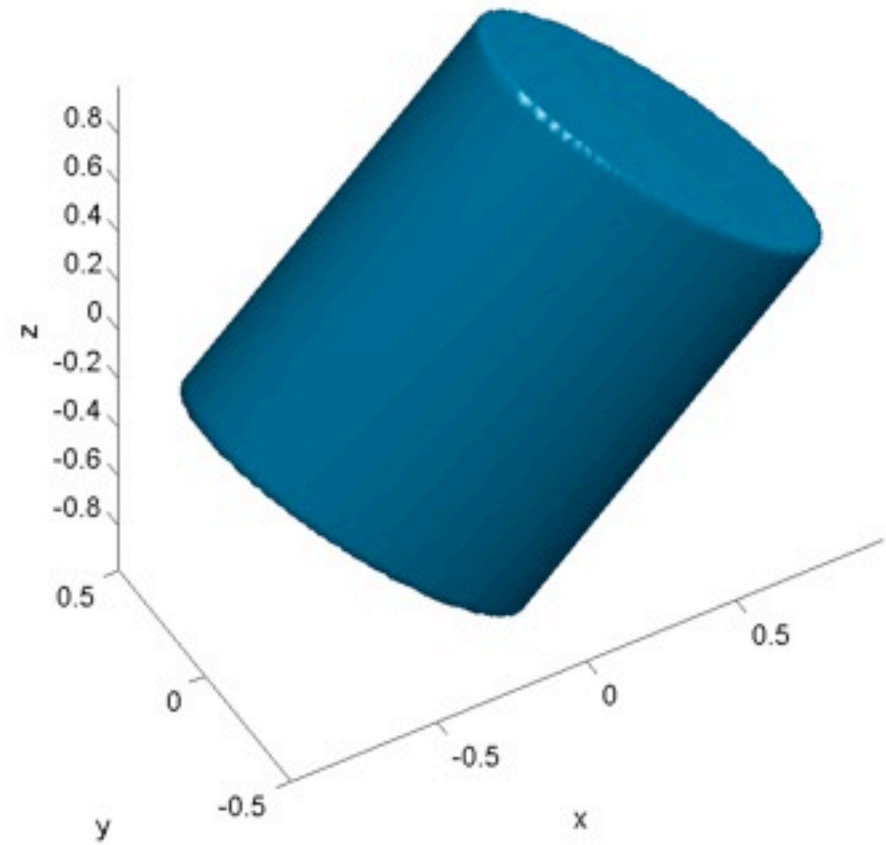
- Projection onto x-z plane is l_1 ball



- 2x2 matrices
- plotted in 3d

$$\left\| \begin{bmatrix} x & y \\ y & z \end{bmatrix} \right\|_* \leq 1$$

- Not polyhedral…



So how do we compute it? And when does it work?

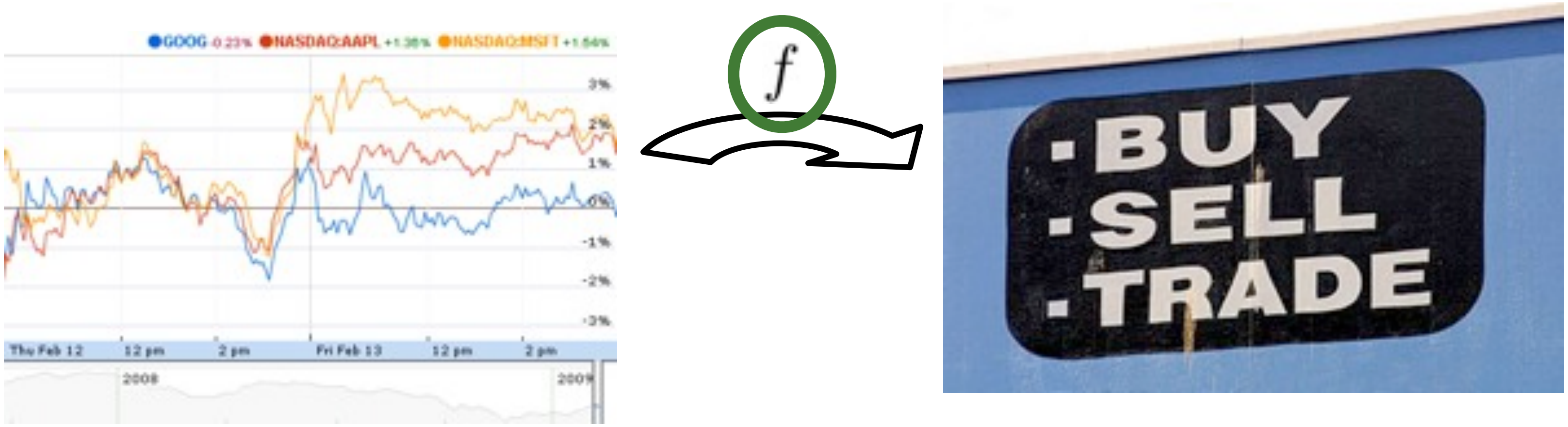
Computationally: Gradient Descent!

$$\mathcal{F}(\mathbf{L}, \mathbf{R}) = \sum_{i=1}^{k} \sum_{j=1}^{r} L_{ij}^2 + \sum_{i=1}^{n} \sum_{j=1}^{r} R_{ij}^2 + \lambda \left\| \mathcal{A}(\mathbf{L}\mathbf{R}^*) - \mathbf{b} \right\|^2$$

- “Method of multipliers”
- Schedule for λ controls the noise in the data
- Same global minimum as nuclear norm

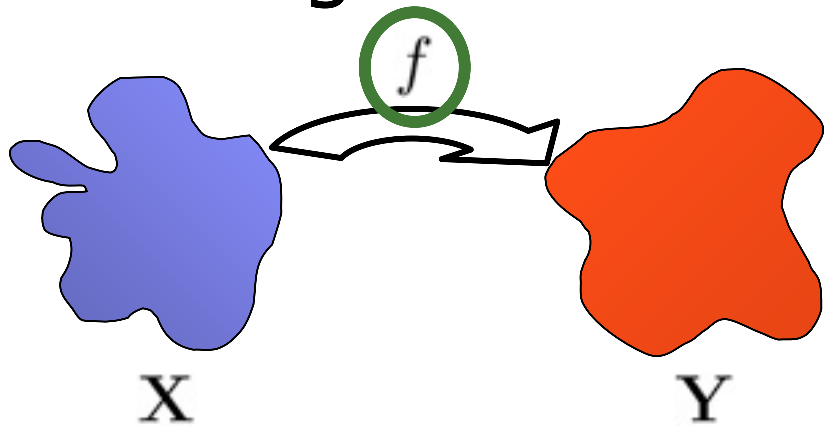
Topic 2: Rank

- How many samples are required to reconstruct low-rank matrices?
- Fast algorithms for SVD as compressed sensing
- When can we guarantee the nuclear norm heuristic works?
- What are efficient ways to compute minimum nuclear norm solutions?


GOOG -0.23% NASDAQ:AAPL +1.38% NASDAQ:MSFT +1.64%
3%
2%
1%
0%
-1%
-2%
-3%
Thu Feb 12
12 pm
2 pm
Fri Feb 13
12 pm
2 pm
2008
2009
f
BUY
SELL
TRADE

Learning Functions

Typically a list of example inputs



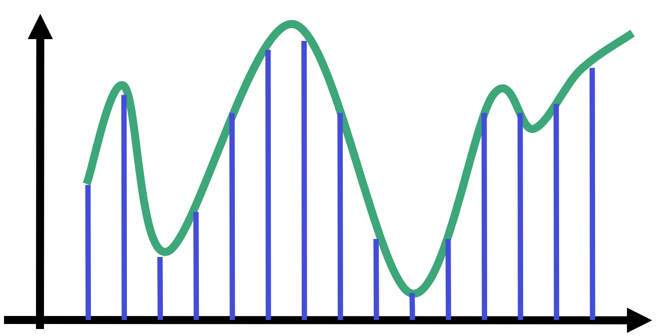
classes covariate state

$$\min_{f\in\mathcal{F}} \ \text{fitness}(f, \text{data})$$

Which space of functions?

dictated by application

- **Goal:** Find a class $\mathcal{F}$ which is easy to search over, but can approximate complex behavior.



Blessing of smoothness

$$\text{error} < O(n^{-\frac{s}{d}})$$

Number of examples

Curse of dimensionality

Topic 3: Approximation

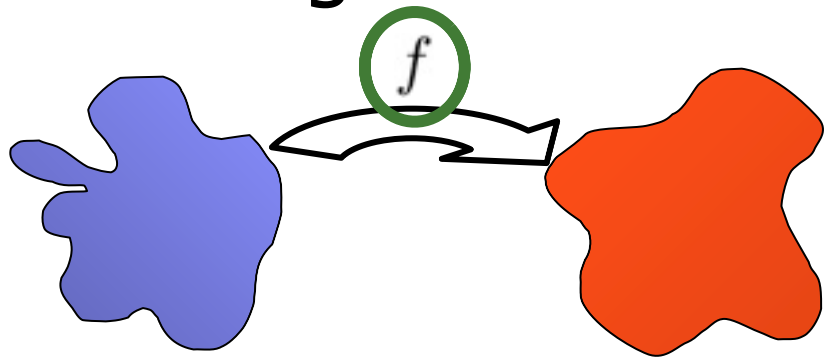
- Try to write a function as a sum of (non-orthogonal) bases:

$$f(x) \approx \sum_{k=1}^{n} c_k \phi_k(\mathbf{x}; \theta_k)$$

- Atoms are sets of basis functions
- Not a discrete set, infinite dimensional space.

Learning Functions

Typically a list of example inputs



classes covariate state

$\mathbf{X}$ $\mathbf{Y}$

$$\min_{f\in\mathcal{F}} \ \mathrm{fitness}(f, \mathrm{data})$$

Which space of functions? dictated by application

- **Goal:** Find a class $\mathcal{F}$ which is easy to search over, but can approximate complex behavior.

- **Solution:** Approximate $f(\mathbf{x})$ by $f_n(\mathbf{x}) = \sum_{k=1}^{n} c_k \phi_k(\mathbf{x};\theta_k)$

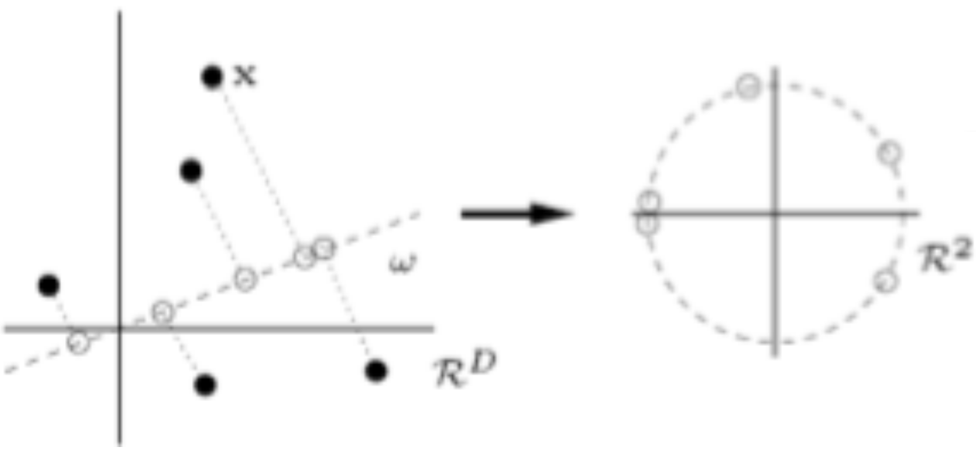
optimize sample

- Approximate $f(\mathbf{x})$ by $f_n(\mathbf{x}) = \sum_{k=1}^{n} c_k \phi_k(\mathbf{x}; \theta_k)$

For large class of *f*, sampling θ_k i.i.d. and optimizing c_k yields

$$\|f - f_n\| = O\left(\frac{1}{\sqrt{n}}\right)$$

Analysis via convex hull norm where the atoms are $\phi(\mathbf{x}; \theta)$



$$\phi(\mathbf{x};\, \omega, b) = \cos(\omega'\mathbf{x} + b)$$
$$\omega \sim \mathcal{N}(0, 1)$$
$$b \sim \text{unif}[-\pi, \pi]$$

$$\boxed{k(\mathbf{x}, \mathbf{y}) = \exp(-\gamma\|\mathbf{x} - \mathbf{y}\|^2)}$$

Radial Basis Functions

```
% Approximates Gaussian Process regression
%   with Gaussian kernel of variance gamma
% lambda: regularization parameter
% dataset: X is dxN, y is 1xN
% test: xtest is dx1
% D: dimensionality of random feature

% training
  w = randn(D, size(X,1));
  b = 2*pi*rand(D,1);
  Z = cos(sqrt(gamma)*w*X + repmat(b,1,size(X,2)));
  alpha = (lambda*eye(size(X,2)+Z*Z')\(Z*y);

% testing
  ztest = alpha(:)'*cos( sqrt(gamma)*w*xtest(:) + …
            + repmat(b,1,size(X,2)) );
```

Topic 3: Approximation

- How many bases are required to approximate complicated behavior?
- What are efficient ways to fit functions in infinite dimensional function spaces?
- What are fast ways to fit functions when we are overwhelmed by data?