Dimensionality Reduction
Lecture 24

David Sontag
New York University

Slides adapted from Carlos Guestrin and Luke Zettlemoyer

Dimensionality reduction

- Input data may have thousands or millions of dimensions!
  - e.g., text data has ???, images have ???
- **Dimensionality reduction**: represent data with fewer dimensions
  - easier learning – fewer parameters
  - visualization – show high dimensional data in 2D
  - discover "intrinsic dimensionality" of data
    - high dimensional data that is truly lower dimensional
    - noise reduction

Feature selection

- Want to learn $f{:}\mathbf{X}\rightarrow Y$
  - $\mathbf{X}=\langle X_1,\ldots,X_n\rangle$
  - but some features are more important than others
- **Approach**: select subset of features to be used by learning algorithm
  - **Score** each feature (or sets of features)
  - **Select** set of features with best score

Greedy **forward** feature selection algorithm

- Pick a dictionary of features
  - e.g., polynomials for linear regression
- Greedy: Start from empty (or simple) set of features $F_0 = \varnothing$
  - Run learning algorithm for current set of features F_t
    - Obtain h_t
  - Select **next best feature X_i**
    - e.g., X_j that results in lowest held out error when learning with $F_t \cup \{X_j\}$
  - $F_{t+1} \leftarrow F_t \cup \{X_i\}$
  - Repeat

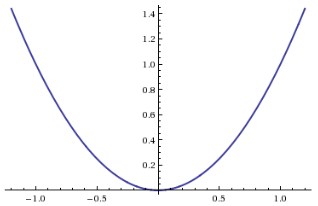
Greedy **backward** feature selection algorithm

- Pick a dictionary of features
  - e.g., polynomials for linear regression
- Greedy: Start with all features $F_0 = F$
  - Run learning algorithm for current set of features F_t
    - Obtain h_t
  - Select **next worst feature X_i**
    - e.g., X_j that results in lowest held out error learner when learning with F_t - $\{X_j\}$
  - $F_{t+1} \leftarrow F_t - \{X_i\}$
  - Repeat

Feature selection through regularization

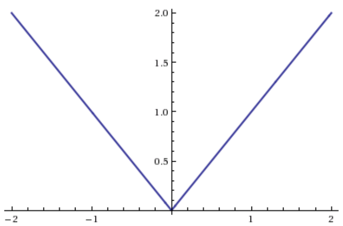
- Previously, we discussed regularization with a squared norm:

$$\hat{\theta} = \arg\min_{\theta} Loss(\theta; \mathcal{D}) + \lambda \sum_i \theta_i^2$$



- We motivated the L2 norm using the idea of **margin**
- What if we have reason to believe that there are only a few relevant features?
- In this case, we should regularize using the L1 norm!

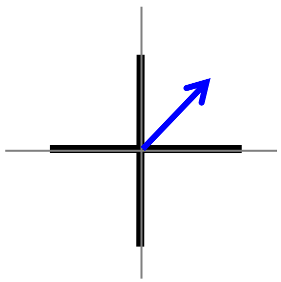
$$\hat{\theta} = \arg\min_{\theta} Loss(\theta; \mathcal{D}) + \lambda \sum_i |\theta_i|$$



- Big area of machine learning called "sparse recovery"

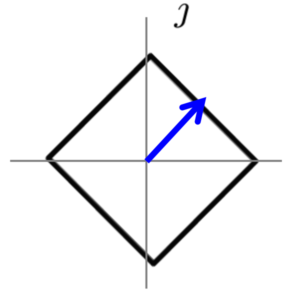
Feature selection through regularization

$$\|W\|_0 = \#\{W_j > 0\}$$



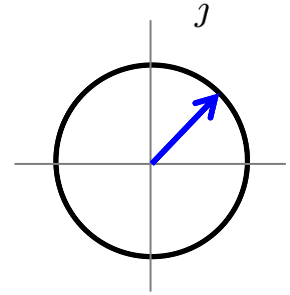
Minimizes # features chosen

$$\|W\|_1 = \sum_j |W_j|$$



Convex compromise

$$\|W\|_2 = \sum_j W_j^2$$

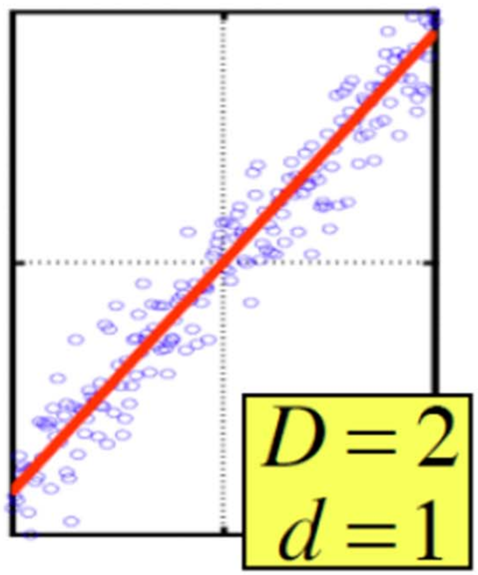


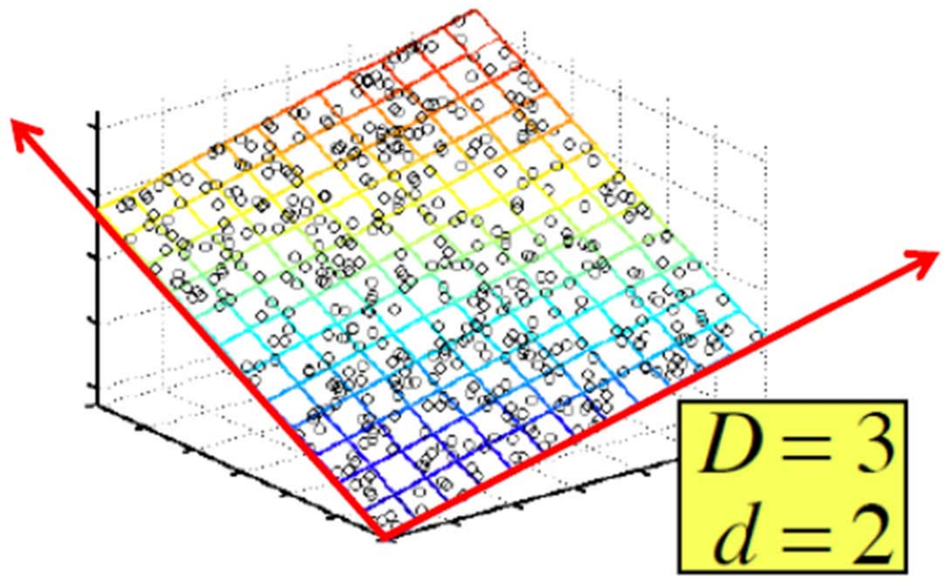
Small weights of features chosen

Slide from Aarti Singh

Dimension reduction

- Assumption: data (approximately) lies on a lower dimensional space
- Examples:





Slide from Yi Zhang

Lower dimensional projections

- Rather than picking a subset of the features, we can obtain new ones by combining existing features $x_1 \ldots x_n$

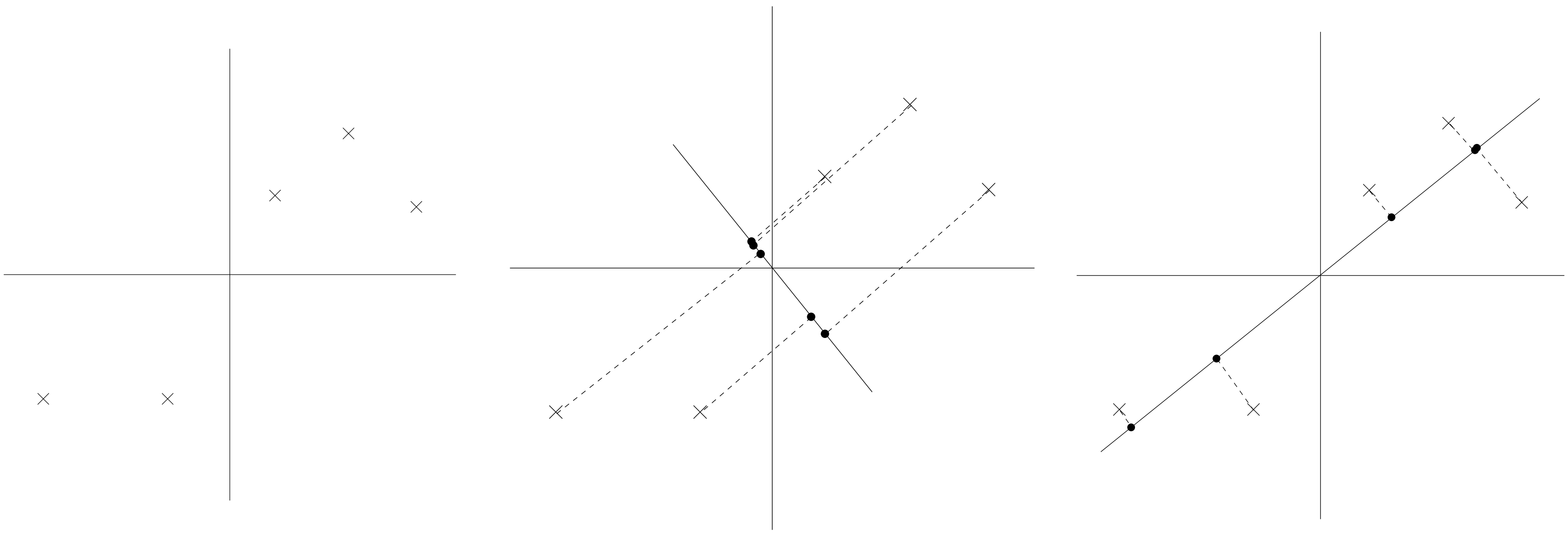
$$z_1 = w_0^{(1)} + \sum_i w_i^{(1)} x_i$$

$$\cdots$$

$$z_k = w_0^{(k)} + \sum_i w_i^{(k)} x_i$$

- New features are linear combinations of old ones
- Reduces dimension when k<n
- Let's consider how to do this in the unsupervised setting
  - just **X**, but no Y

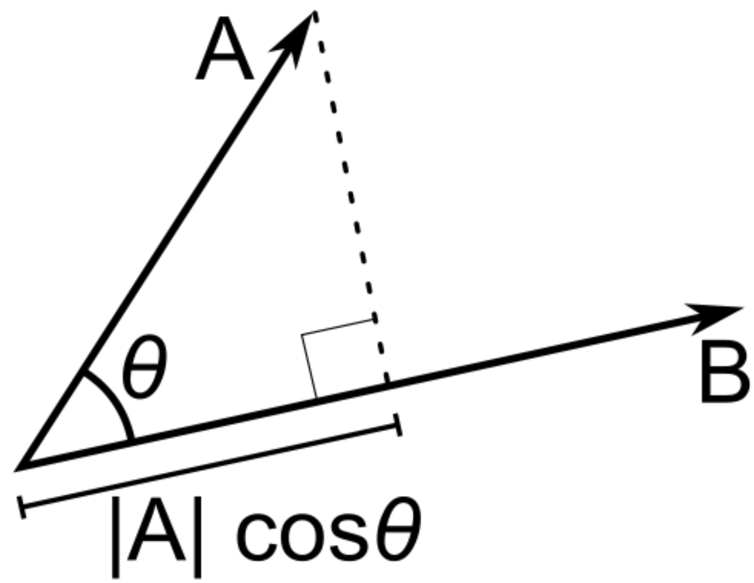
Which projection is better?



From notes by Andrew Ng

Reminder: Vector Projections

- Basic definitions:
  - $A.B = |A||B| \cos \theta$
  - $\cos \theta = |adj|/|hyp|$



- Assume $|B|=1$ (unit vector)
  - $A.B = |A| \cos \theta$
  - So, dot product is length of projection!!!

Maximize variance of projection

Let $x^{(i)}$ be the i^{th} data point minus the mean.

Choose unit-length u to maximize:

$$
\begin{aligned}
\frac{1}{m}\sum_{i=1}^{m}({x^{(i)}}^T u)^2 &= \frac{1}{m}\sum_{i=1}^{m} u^T x^{(i)} {x^{(i)}}^T u \\
&= u^T\left(\frac{1}{m}\sum_{i=1}^{m} x^{(i)} {x^{(i)}}^T\right) u.
\end{aligned}
$$

Let ||u||=1 and maximize. Using the method of Lagrange multipliers, can show that the solution is given by the principal eigenvector of the covariance matrix! **(shown on board)**

Basic PCA algorithm

- Start from m by n data matrix **X**
- **Recenter**: subtract mean from each row of **X**
  - $\mathbf{X_c} \leftarrow \mathbf{X} - \overline{\mathbf{X}}$
- **Compute covariance** matrix:
  - $\Sigma \leftarrow 1/m\ \mathbf{X_c}^T \mathbf{X_c}$
- Find **eigen vectors and values** of Σ
- **Principal components:** k eigen vectors with highest eigen values