Text Classification Contd + Document Representations

Prof. Sameer Singh

CS 295: STATISTICAL NLP

WINTER 2017

January 17, 2017

Based on slides from Nathan Schneider, Noah Smith, Dan Klein and everyone else they copied from.

Outline

Logistic Regression

Brief Intro to Neural Networks

Document Representations

Outline

Logistic Regression

Brief Intro to Neural Networks

Document Representations

Text Classification

Paper Title

Human machine interface for ABC computer applications

CS Area

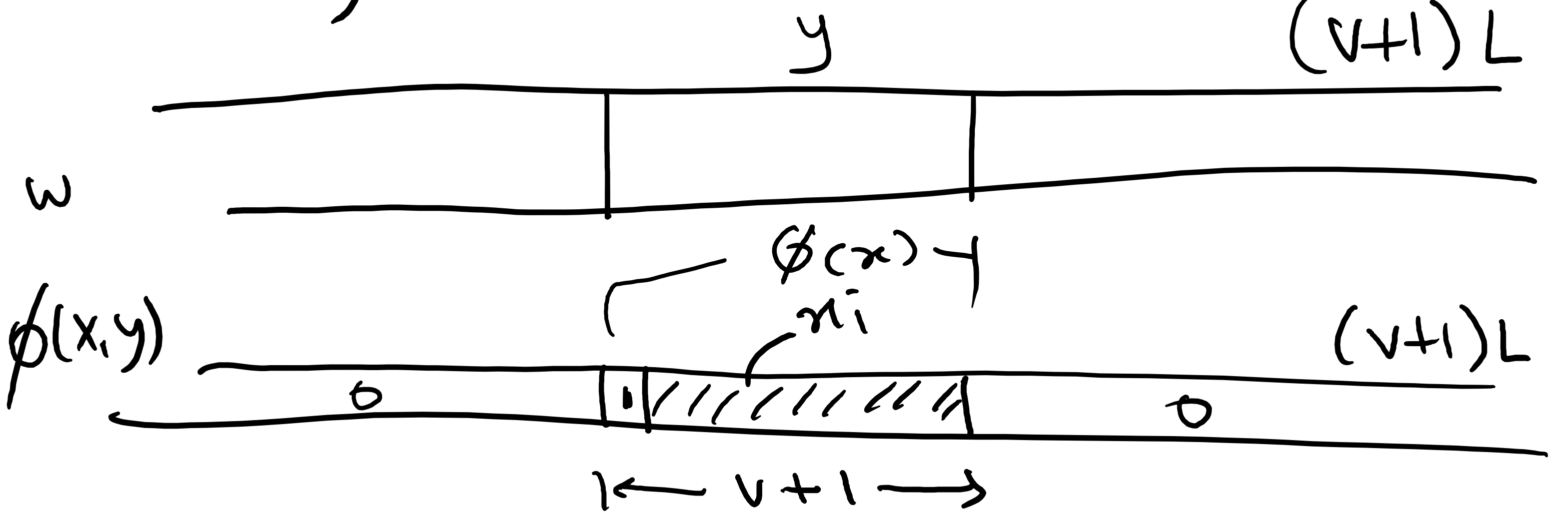
- Human Computer Interaction
- Theory
- Artificial Intelligence
- Systems

Linear Models

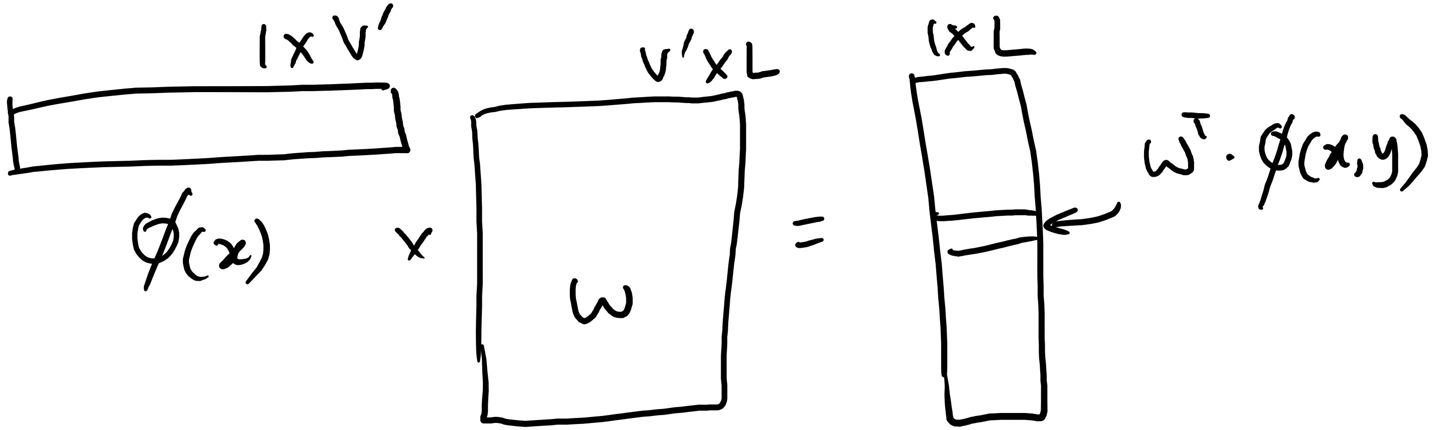
Human machine interface for
ABC computer applications

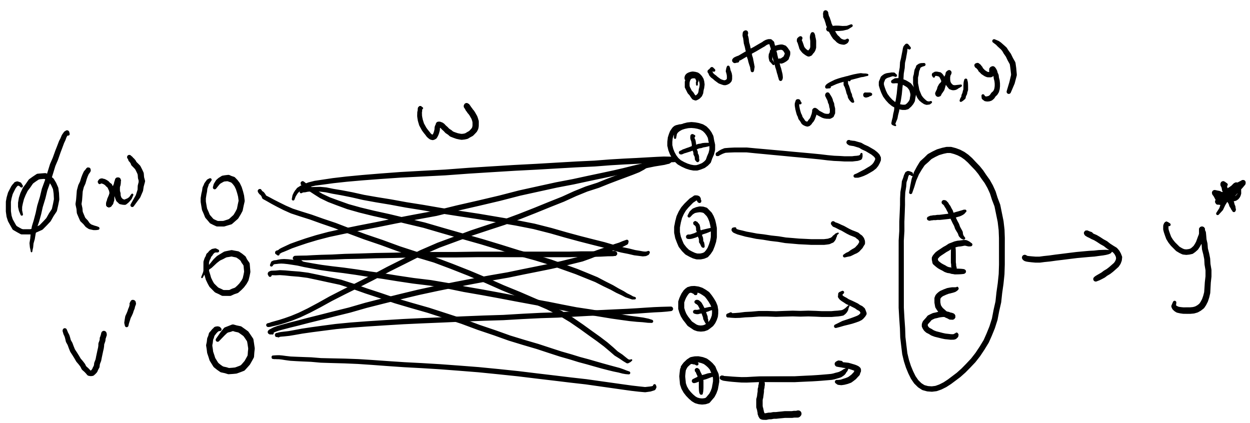
X $\longrightarrow$ $y \in \{\text{HCI}, \text{AI}, \text{Th}, \text{Sys}\}$ $\quad L$

$$f(x) = \arg\max_{y} \vec{w} \cdot \phi(x,y)$$



Matrix/Neural View

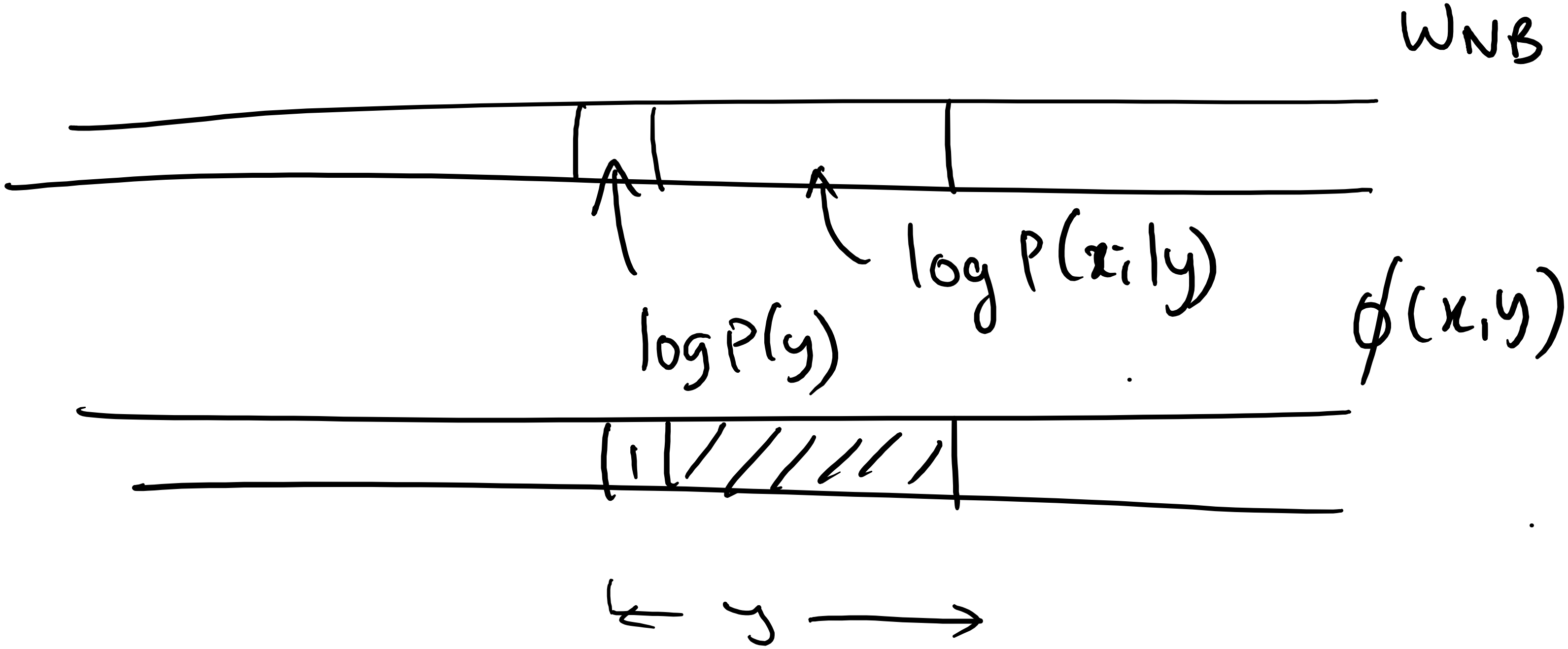




Naïve Bayes as a Linear Model

$$f_{NB}(x) = \arg\max_y P(x,y) = \arg\max_y P(y) \prod_i P(x_i|y)$$

$$= \arg\max_y \log P(y) + \sum_i \log P(x_i|y)$$



Joint vs Conditional Likelihood

$$f_{NB}(x) = \arg\max_{y} P(x,y) \leftarrow P(x)$$

Conditional

$$f_{cond}(x) = \arg\max_{y} P(y|x) \leftarrow P(x)$$

$$X$$

$$\theta^{*}_{cond} = \arg\max_{\theta} \prod_{a \in D} P_{\theta}(y_a|x_a)$$

Logistic Regression Model

$$f(x) = \operatorname{argmax} \; w \cdot \phi(x,y)$$

$$P(y|x) = \frac{e^{w \cdot \phi(x,y)}}{\sum_{y} e^{w \cdot \phi(x,y)}}$$

(softmax)

$$f(x) = \operatorname*{argmax}_{y} \; p(y|x)$$

$$= \operatorname*{argmax}_{y} \; \frac{e^{w \cdot \phi(x,y)}}{\sum_{y'} e^{w \cdot \phi(x,y)}}$$

$$= \operatorname*{argmax}_{y} \; w \cdot \phi(x,y)$$

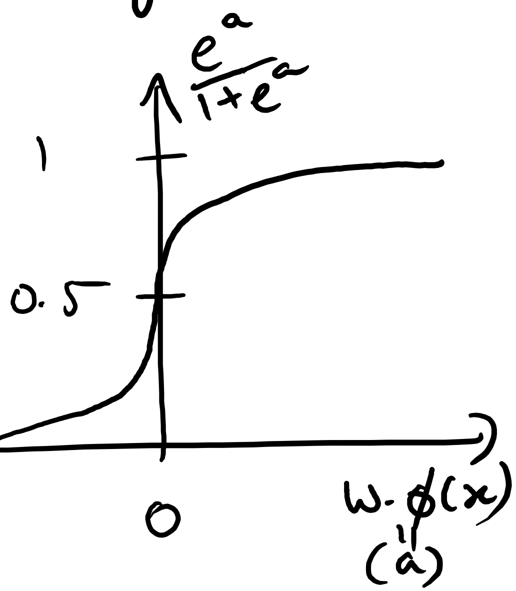
Logistic Regression: 2 classes

$$p(y=1 \mid x) = \frac{e^{w\cdot\phi(x)}}{1+e^{w\cdot\phi(x)}}$$ (sigmoid)

$e^{0\cdot\phi(\cdot)}$

$\phi_k(x) = \#\text{word } k$

"interface" 0



$w_{interface} > 0$

"tree"

$w_{tree} < 0$

Estimating the parameters, w

$$w^* = \underset{w}{\operatorname{argmax}} \prod_a P(y_a \mid x_a)$$

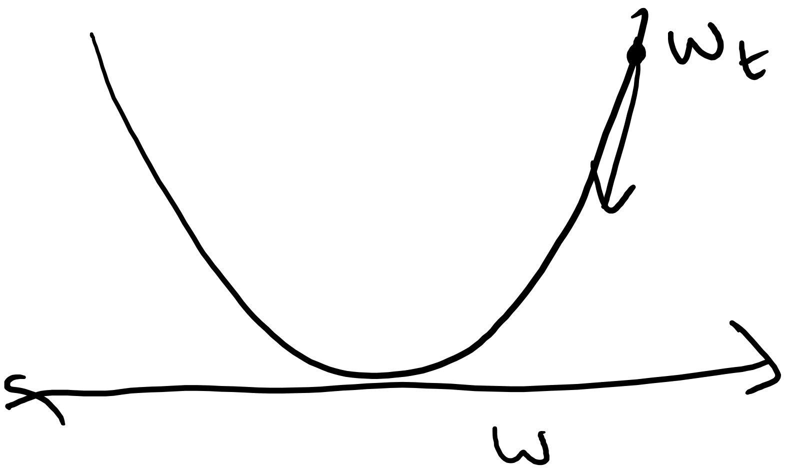
$$= \underset{w}{\operatorname{argmax}} \sum_a \log P(y_a \mid x_a) \rightarrow \frac{e^{w \cdot \phi(x_a, y_a)}}{\sum_y e^{w \cdot \phi(x_a, y)}}$$

$$= \underset{w}{\operatorname{argmax}} \underbrace{\sum_a w \cdot \phi(x_a, y_a) - \log \sum_y e^{w \cdot \phi(x_a, y)}}$$

$$= \underset{w}{\operatorname{argmin}} -$$

Gradient Descent

$$\arg\min \underbrace{- \sum_a w \cdot \phi(x_a, y_a) - \log \sum_y \frac{e^{-}}{\Sigma\ -}}_{L}$$



$$\frac{\partial L}{\partial w_i} = \sum_a \underbrace{\phi_i(x_a, y_a)}_{\text{Data}} - \underbrace{E_{p(y|x)}}_{\text{Model}}\left[\phi_i(x_a, y)\right]$$

Tips and Tricks: TF-IDF

Sparsity of Words

"is"

"interface"

- Remember Zipf's Law? Lots of rare words
- For classification, they can be more informative!

x_{id} = # word$_i$ doc d

(term freq (TF)

D_i = # docs that i appears in

($\sum_d |x_{id}| > 0$

D = # docs

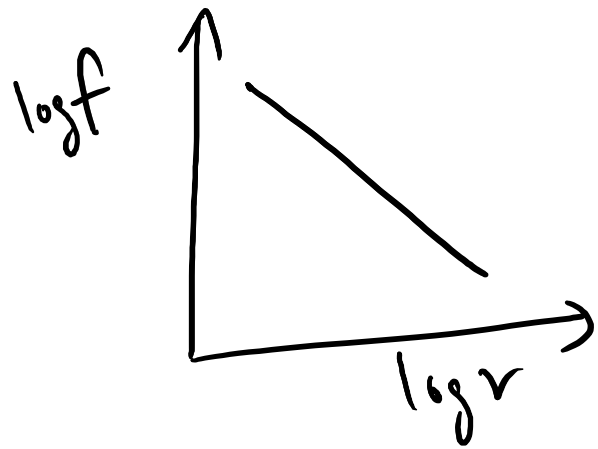
$$\boxed{\phi(x_{id}) = tf(i,d)\, idf(i)} \quad \underline{idf}(i) = -\log P(i) = -\log \frac{D_i}{D} = \log \frac{D}{D_i}$$

inverse doc freq

Tips and Tricks: TF-IDF

Why use log(proportion)

- It works…
- Importance is not a $\underbrace{\text{linear function}}_{\text{"log"}}$
- IDF is an additive function



$$idf(w_1, w_2) = -\log P(w_1, w_2) = -\log P(w_1)P(w_2)$$

$$d_{i2} \qquad = -\log P(w_1) - \log P(w_2)$$

$$= idf(w_1) + idf(w_2)$$

Tips and Tricks: Regularization

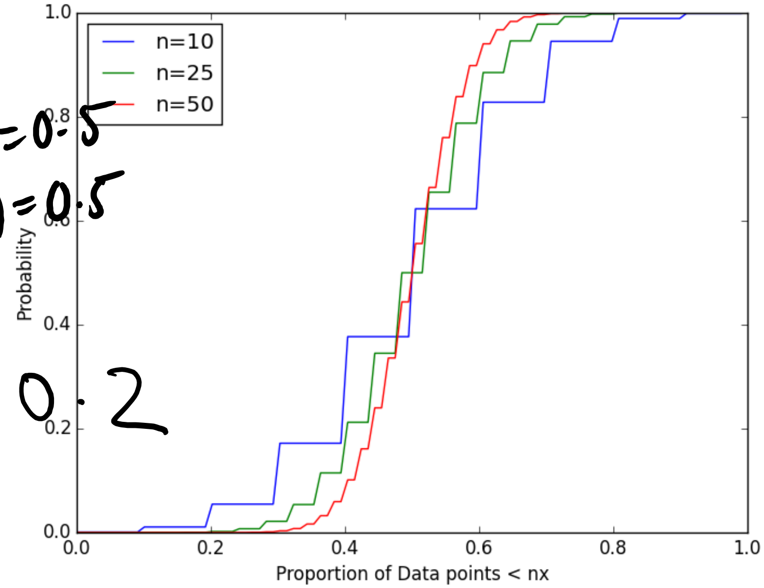
Overfitting

- Training data is finite: thus has spurious correlations
- Rare words that occur with one label!
  - Or don't occur often enough
  - Curse of the Zipf's Law continues

$w \quad p(y_1|w) = 0.5$

For a word that occurs 10 times… $p(y_2|w) = 0.5$

$$P(\#_w = y_1 \geq 7) \sim 0.2$$



There are many that occur ~10 times!

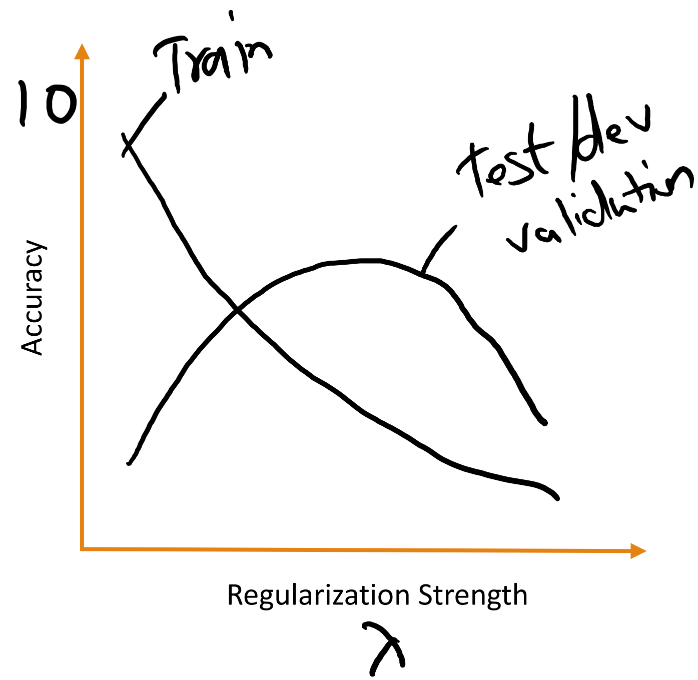
Tips and Tricks: Regularization

Fixing Overfitting

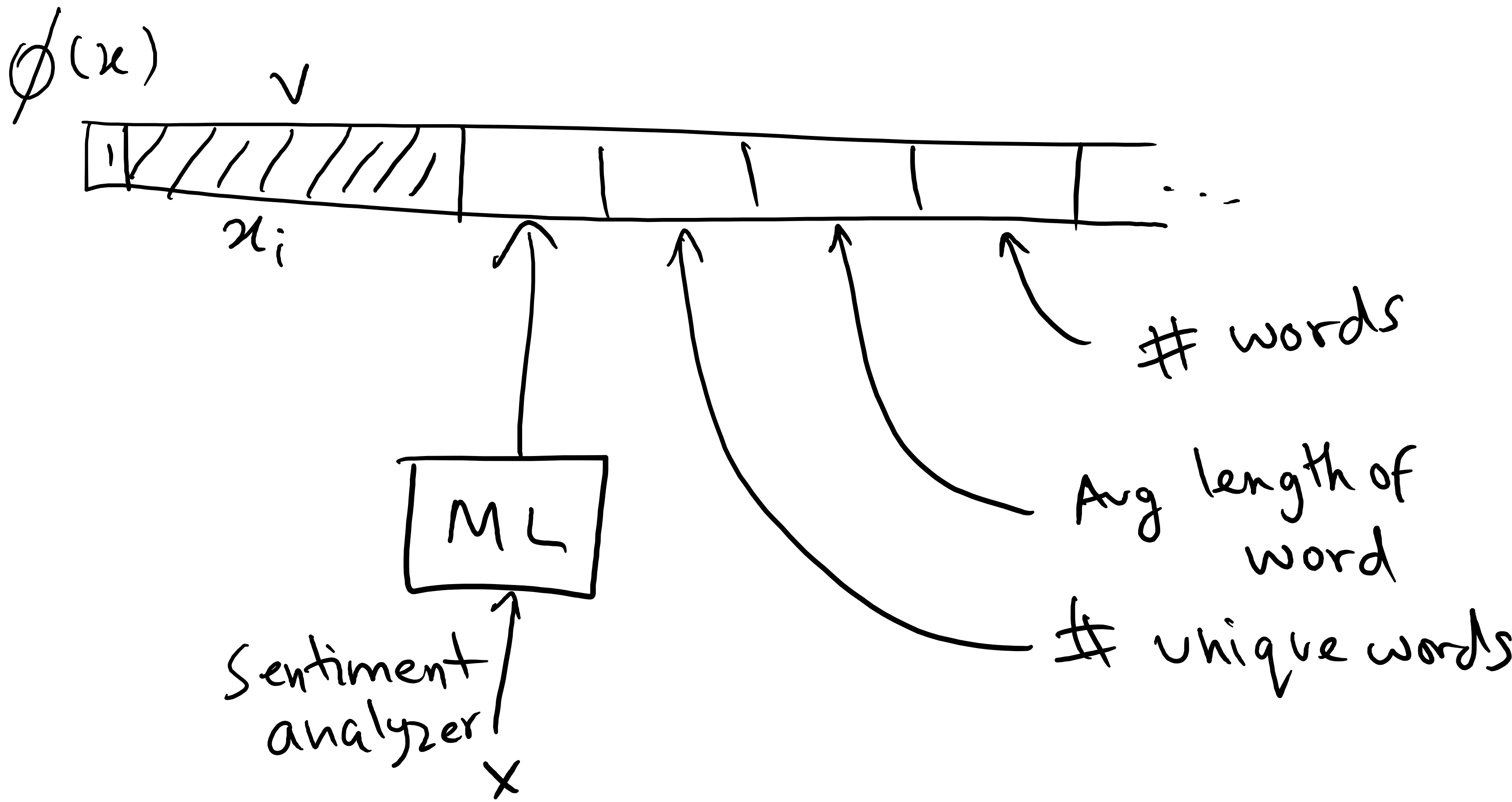
- Ignore rare words (opposite of TF-IDF) $D_i < 10$
- Penalize really high weights…

$$w = \operatorname*{argmin}_{w} \mathcal{L} + \lambda \|w\|_2^2$$

$$\frac{\partial \mathcal{L}}{\partial w_i} + 2\underbrace{\lambda}_{\text{Reg. Strength}} w_i$$



Tips and Tricks: Featurizing



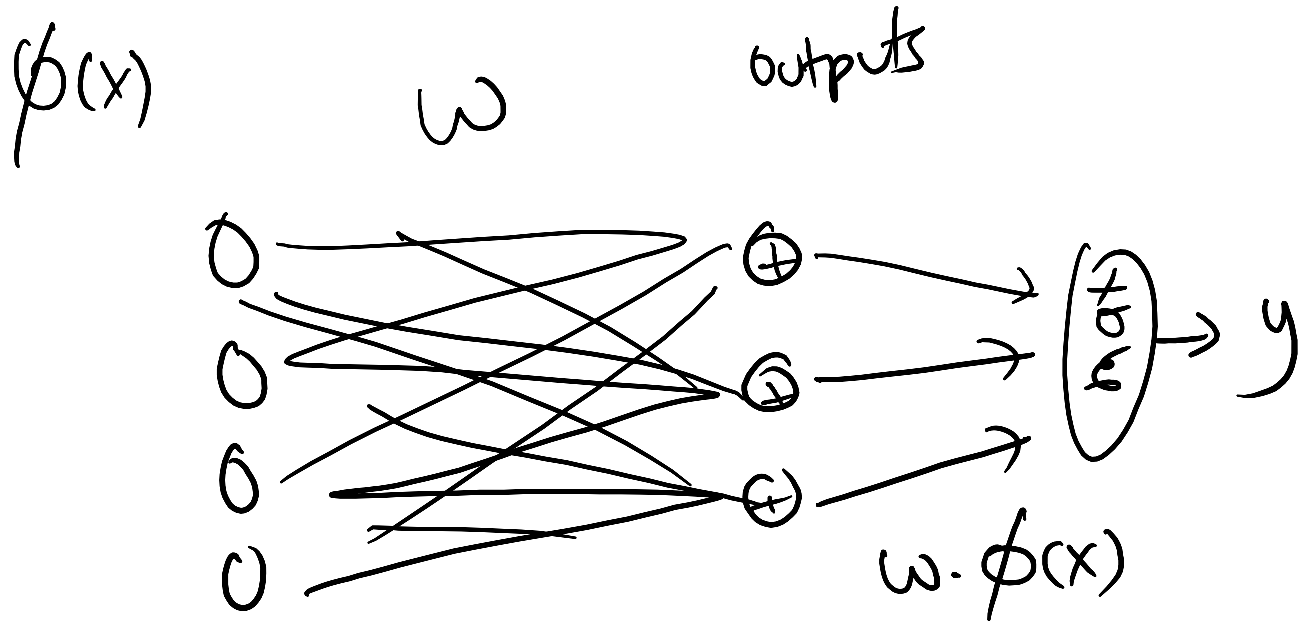
Outline

Logistic Regression

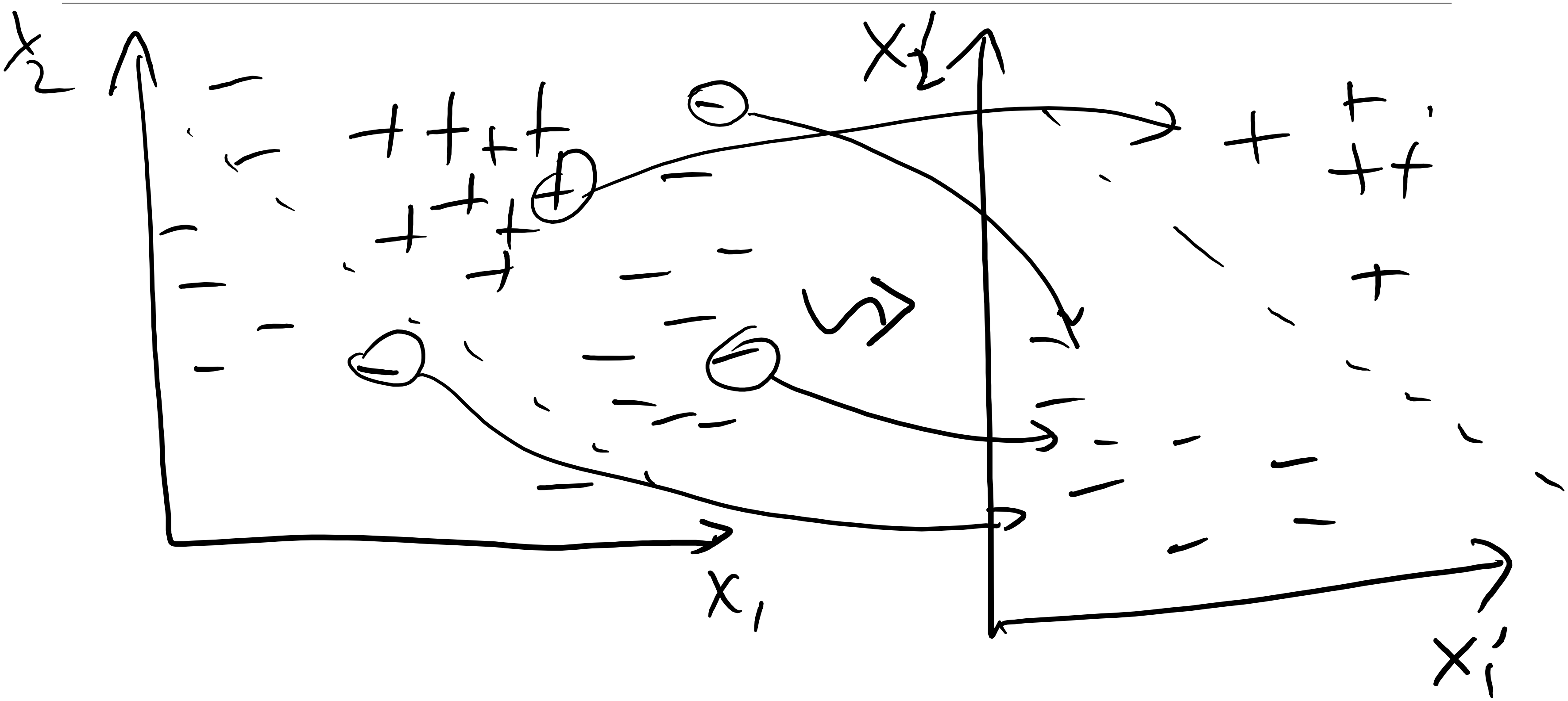
Brief Intro to Neural Networks

Document Representations

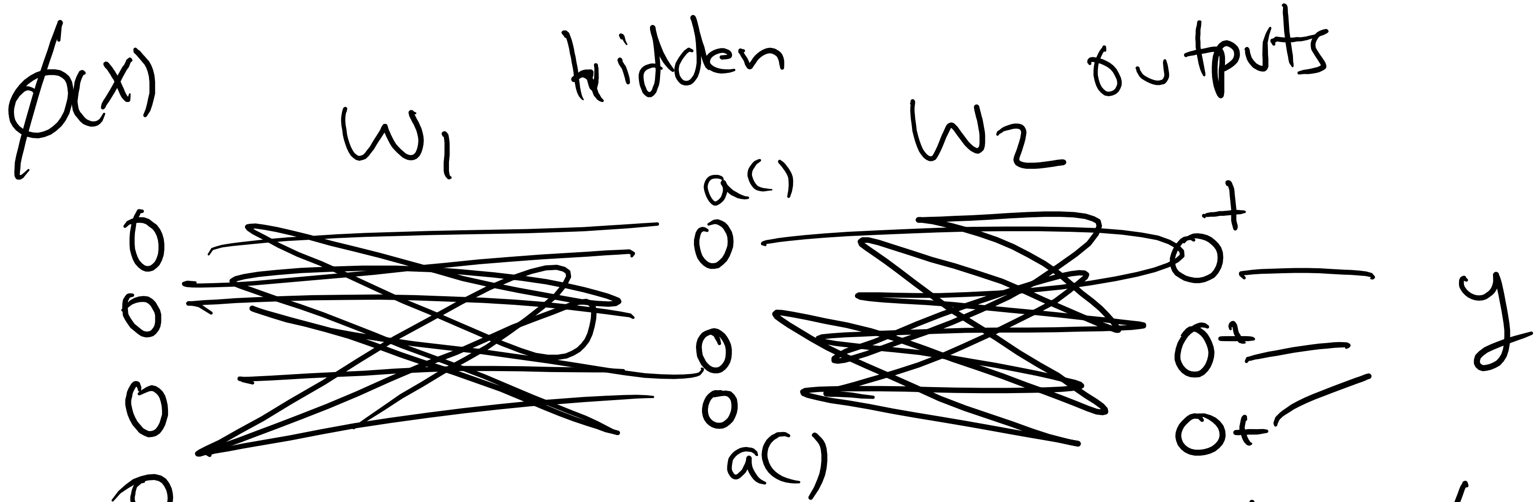
Neural View of Log. Regression



Linear vs Non-linear Model



Introducing a Hidden Layer



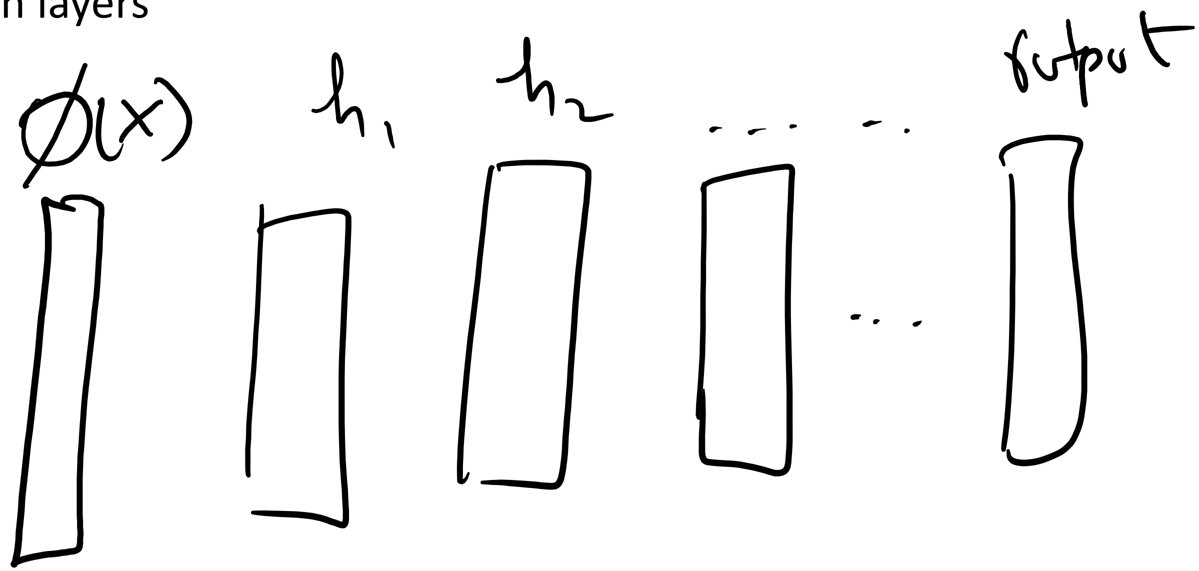
$$h = a(W_1 \phi(x))$$

$$y = W_2 h$$

$$y^* = \arg\max_y$$

What is Deep Learning?

Many hidden layers



In NLP, utilize unlabeled data to learn representations… (next lecture)

Outline

Logistic Regression

Brief Intro to Neural Networks

Document Representations

Document Similarity

A survey of user opinion of computer system response time

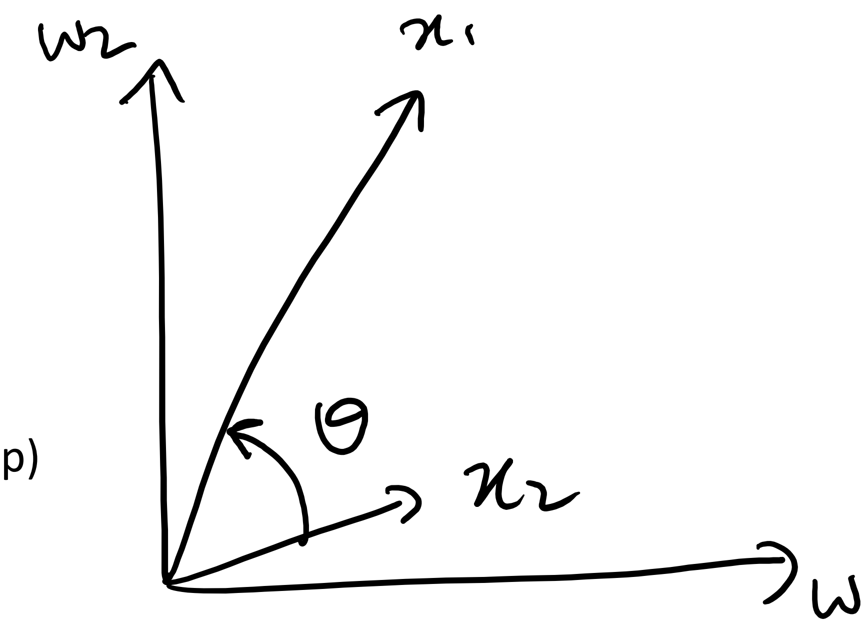
Relation of user perceived response time to error measurement

The generation of random, binary, ordered trees

Cosine Distance

d_1 d_2

$\vec{x}_1$ $\vec{x}_2$

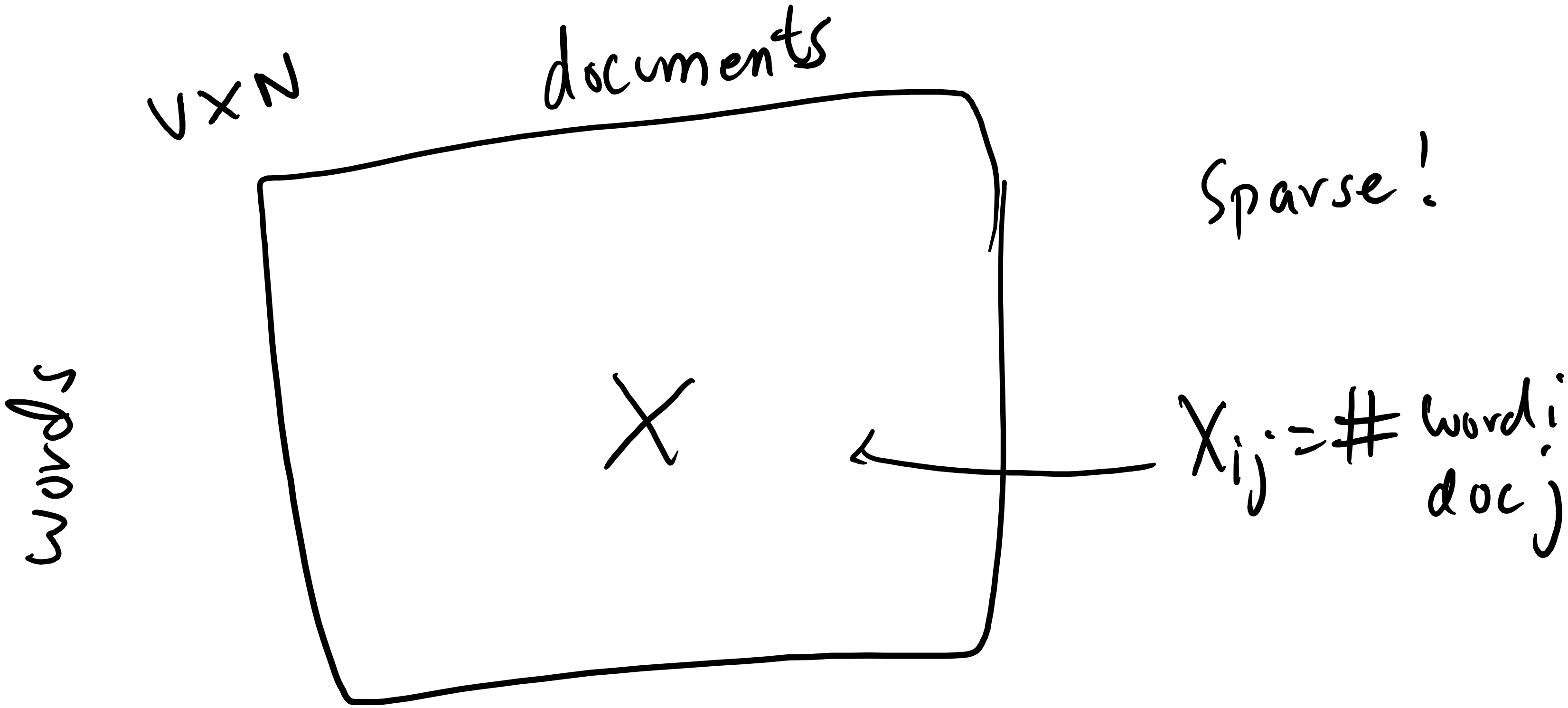


Advantages

- Between -1 and 1 (0 means no overlap)
- If all >0, it is between 0 and 1
- Size of vectors don't matter

$$dist(x,y) = \cos\theta = \frac{x_1 \cdot x_2}{||x_1|| \, ||x_2||} = \frac{\sum_k x_{1k} \cdot x_{2k}}{\sqrt{\sum_k x_{1k}^2}\sqrt{\sum_k x_{2k}^2}}$$

Term Document Matrix



Local and Global Weighting

$$X_{ij} = l_{ij}\, g_i$$

Local Weighting

- Binary: 1
- Term Freq:
- Log: $\log(1+\#_{ij})$

Global Weighting

- Binary: 1
- Normal: $\frac{1}{D_i}$
- IDF: $idf(i)$

Example: Documents

c1: Human machine interface for ABC computer applications
c2: A survey of user opinion of computer system response time
c3: The EPS user interface management system
c4: System and human system engineering testing of EPS
c5: Relation of user perceived response time to error measurement

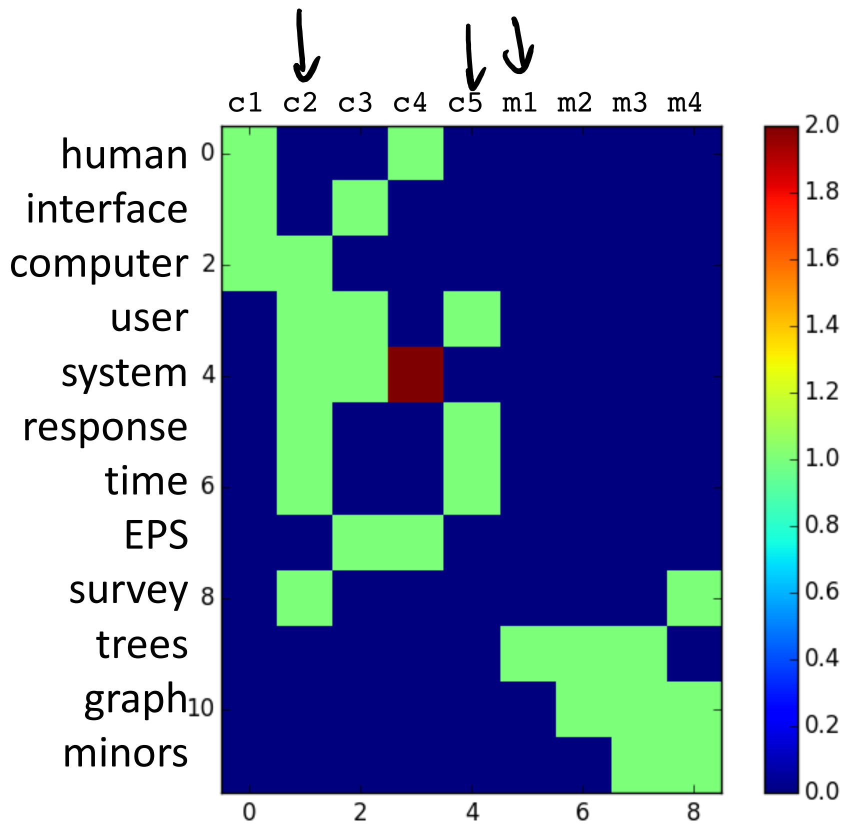
HCI

m1: The generation of random, binary, ordered trees
m2: The intersection graph of paths in trees
m3: Graph minors IV: Widths of trees and well-quasi-ordering
m4: Graph minors: A survey

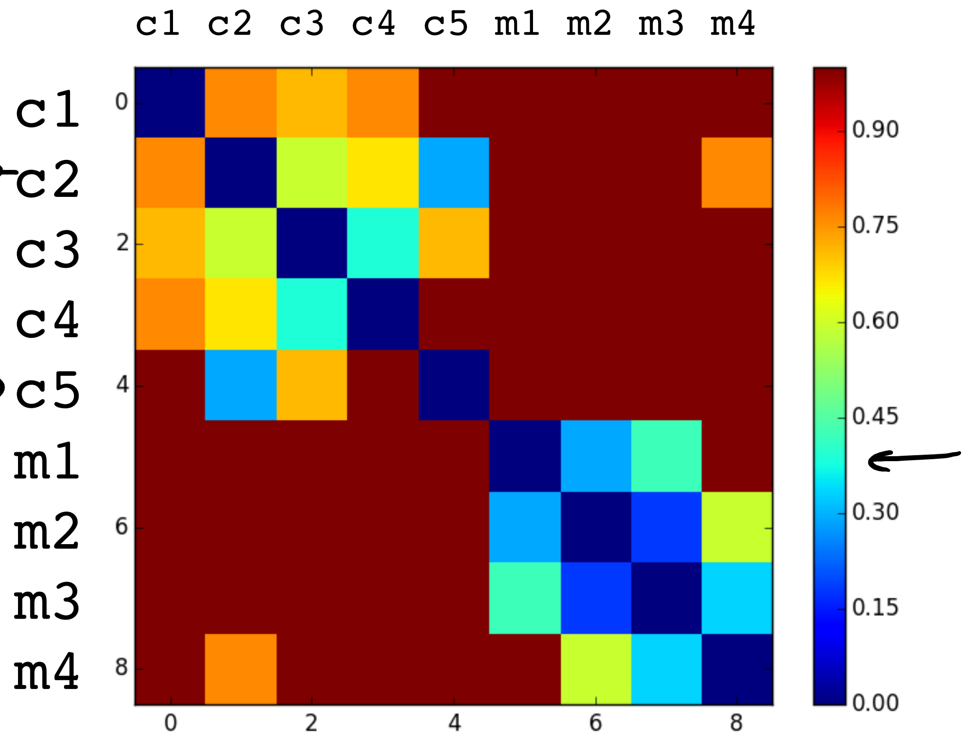
Theory

From http://lsa.colorado.edu/papers/dp1.LSAintro.pdf

Example: Term-Doc Matrix



Example: Distance Matrix



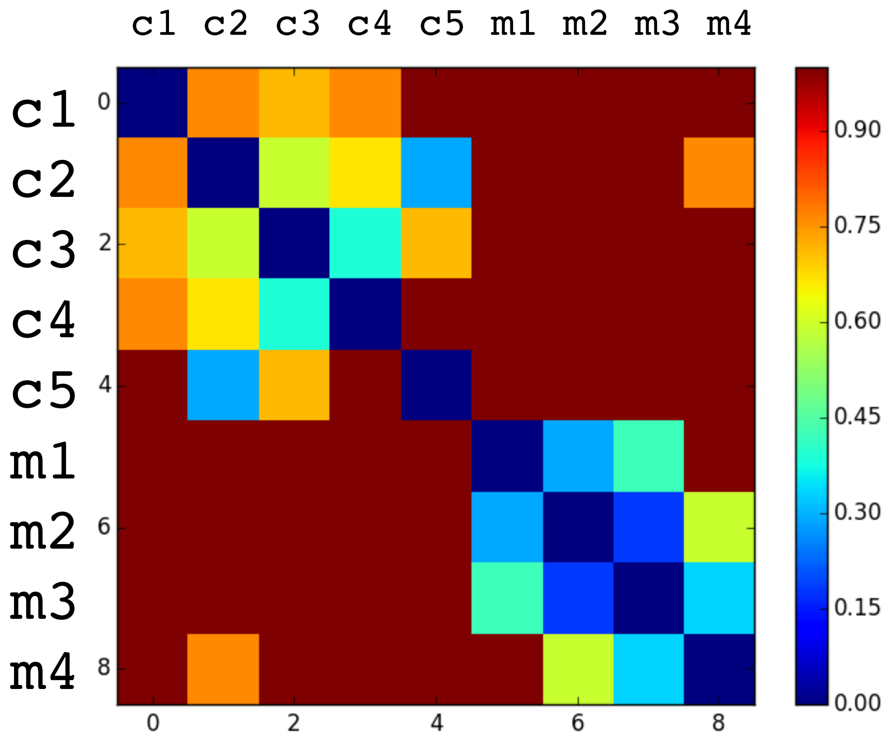
Problems with Sparse Vectors

c2: A survey of user opinion of computer system response time

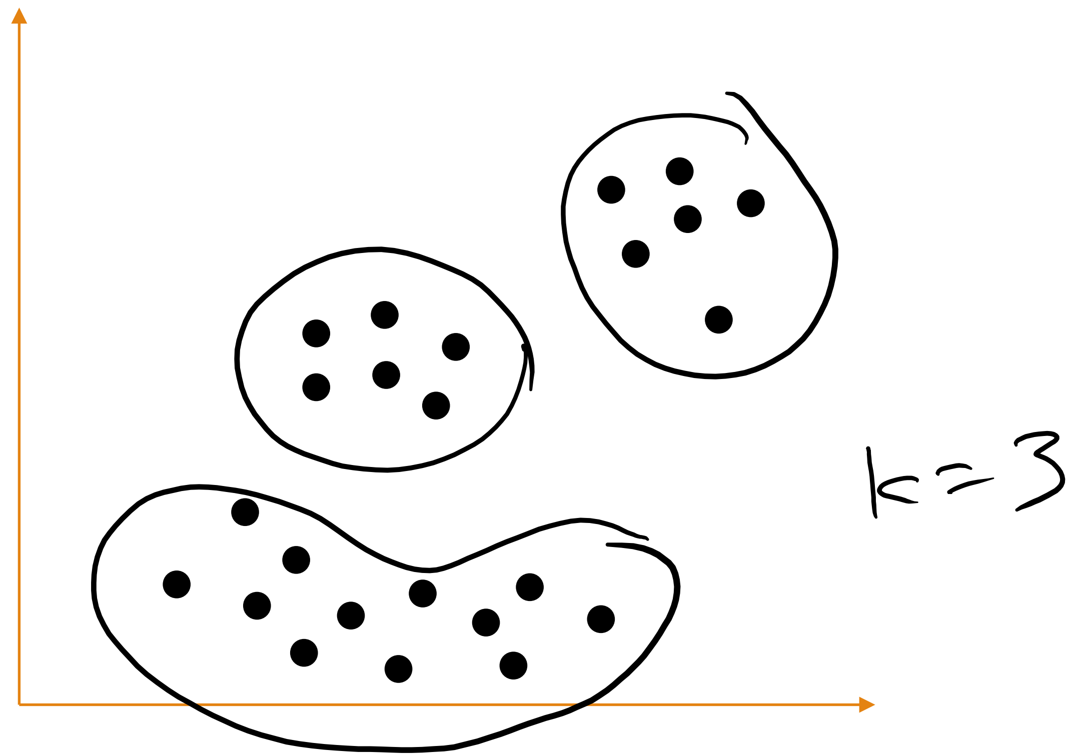
m4: Graph minors: A survey

c1: Human machine interface for ABC computer applications

Example: Distance Matrix



Option 1: Clustering

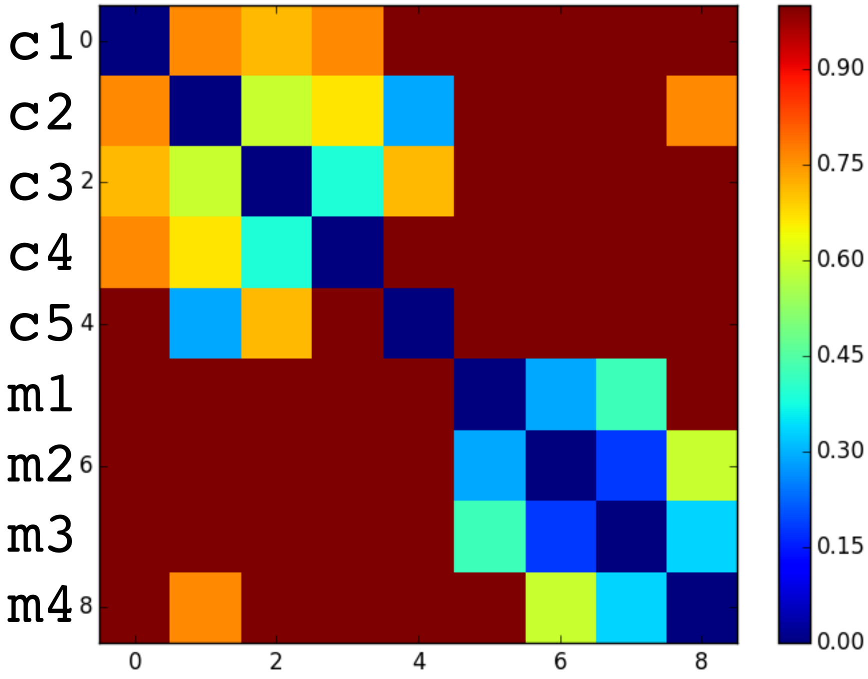


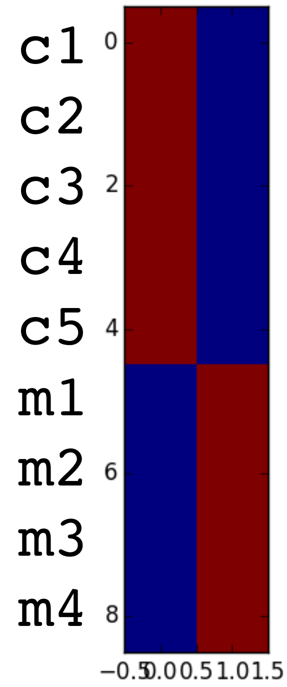
Example: Clustering

$\phi(x)$

1 0

0 1





$K=2$

Upcoming…

Homework

- Homework 1 is up!
- No more material will be covered
- Due: January 26, 2017

Project

- Project pitch is due January 23, 2017!
- Start assembling teams now
- Tons of datasets on the “projects” page on website