Machine Learning and Computational Statistics (DS-GA-1003 and CSCI-GA.2567)

David Sontag
New York University

Slides adapted from Luke Zettlemoyer, Vibhav Gogate, Pedro Domingos, and Carlos Guestrin

Logistics

- **Class webpage:**
  - http://cs.nyu.edu/~dsontag/courses/ml14/
  - Sign up for mailing list!
- **Required lab (instructor: Yoni Halpern)**
  - Thursdays, 8:10-9pm in WWH 109
  - Optional Q&A session from 9:10-9:40pm
- **My office hours:**
  - Tuesdays 7:15-8:15pm
  - 715 Broadway, 12^{th} floor, Room 1204

Evaluation

- About 7 homeworks (45%)
  - Both theory and programming
  - See collaboration policy on class webpage
- Midterm exam (25%)
- Project (25%)
- Course participation (5%)

Problem sets

- First assignment out tonight! Due 2/6.
- See problem set policy on course website
  - First try to solve the problems on your own
  - Then, can discuss with other classmates
  - Write-up solutions on your own
  - List names of anyone you talked to

- Graders:
  Akshay Kumar, Mick Jermsurawong

Projects

- Be creative – think of new problems that you can tackle using machine learning
  - Scope: ~40 hours/person

- Logistics:
  - 2 students per group
  - Begins in March. Project proposal due week after midterm exam
  - Will still be problem sets during this period!

- Project advisers:
  - David Rosenberg, Kurt Miller, Alex Simma

Prerequisites

MS in Data Science students:

- Intro to Data Science (DS-GA-1001)
- Statistical and Mathematical Methods (DS-GA-1002)

MS in Computer Science students:

- Fundamental Algorithms (CSCI-GA.1170)
- Mathematical Techniques for Computer Science Applications (CSCI-GA.1180)

Background needed

- **Programming**
  - Python or Matlab recommended
- **Linear algebra**
  - Matrices, vectors, systems of linear equations
  - Eigenvectors, matrix rank
  - Singular value decomposition
- **Multivariable calculus**
  - Derivatives, integration, tangent planes
  - Optimization, Lagrange multipliers
- **Probability**
  - Random variables, independence, Bayes' rule, marginalization
  - Gaussian distribution

Source Materials

No textbook required. Readings will come from freely available online material.

If you really want a book for an additional reference, this is a good option:

- K. Murphy, ***Machine Learning: a Probabilistic Perspective***, MIT Press, 2012

What is Machine Learning ?

(by examples)

Classification

from data to discrete classes

Spam filtering

data

Osman Khan to Carlos — show details Jan 7 (6 days ago) Reply

sounds good
+ok

Carlos Guestrin wrote:
Let's try to chat on Friday a little to coordinate and more on Sunday in person?

Carlos

Welcome to New Media Installation: Art that Learns

Carlos Guestrin to 10615-announce, Osman, Michel — show details 3:15 PM (8 hours ago) Reply

Hi everyone,

Welcome to New Media Installation:Art that Learns

The class will start tomorrow.
Make sure you attend the first class, even if you are on the Wait List.
The classes are held in Doherty Hall C316, and will be Tue, Thu 01:30-4:20 PM.

By now, you should be subscribed to our course mailing list: 10615-announce@cs.cmu.edu.
You can contact the instructors by emailing: 10615-instructors@cs.cmu.edu

Natural _LoseWeight SuperFood Endorsed by Oprah Winfrey, Free Trial 1 bottle, pay only $5.95 for shipping mfw rlk Spam | X

Jaquelyn Halley to nherrlein, bcc: thehorney, bcc: ang — show details 9:52 PM (1 hour ago) Reply

=== Natural WeightL0SS Solution ===

Vital Acai is a natural WeightL0SS product that Enables people to lose wieght and cleansing their bodies faster than most other products on the market.

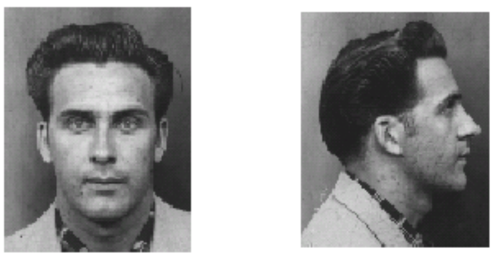
Here are some of the benefits of Vital Acai that You might not be aware of. These benefits have helped people who have been using Vital Acai daily to Achieve goals and reach new heights in there dieting that they never thought they could.

* Rapid WeightL0SS
* Increased metabolism - BurnFat & calories easily!
* Better Mood and Attitude
* More Self Confidence
* Cleanse and Detoxify Your Body
* Much More Energy
* BetterSexLife
* A Natural Colon Cleanse

prediction

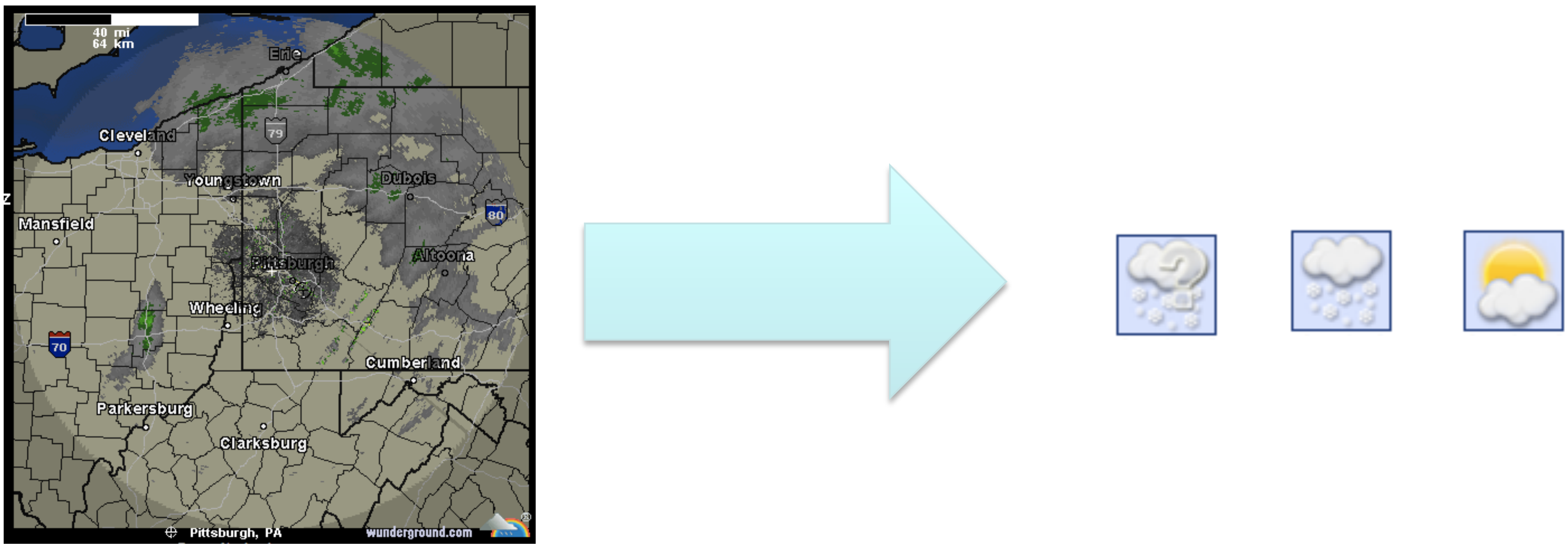
Spam
vs.
Not Spam

Face recognition



Example training images
for each orientation

Weather prediction



Regression

predicting a numeric value

Stock market

Weather prediction revisited

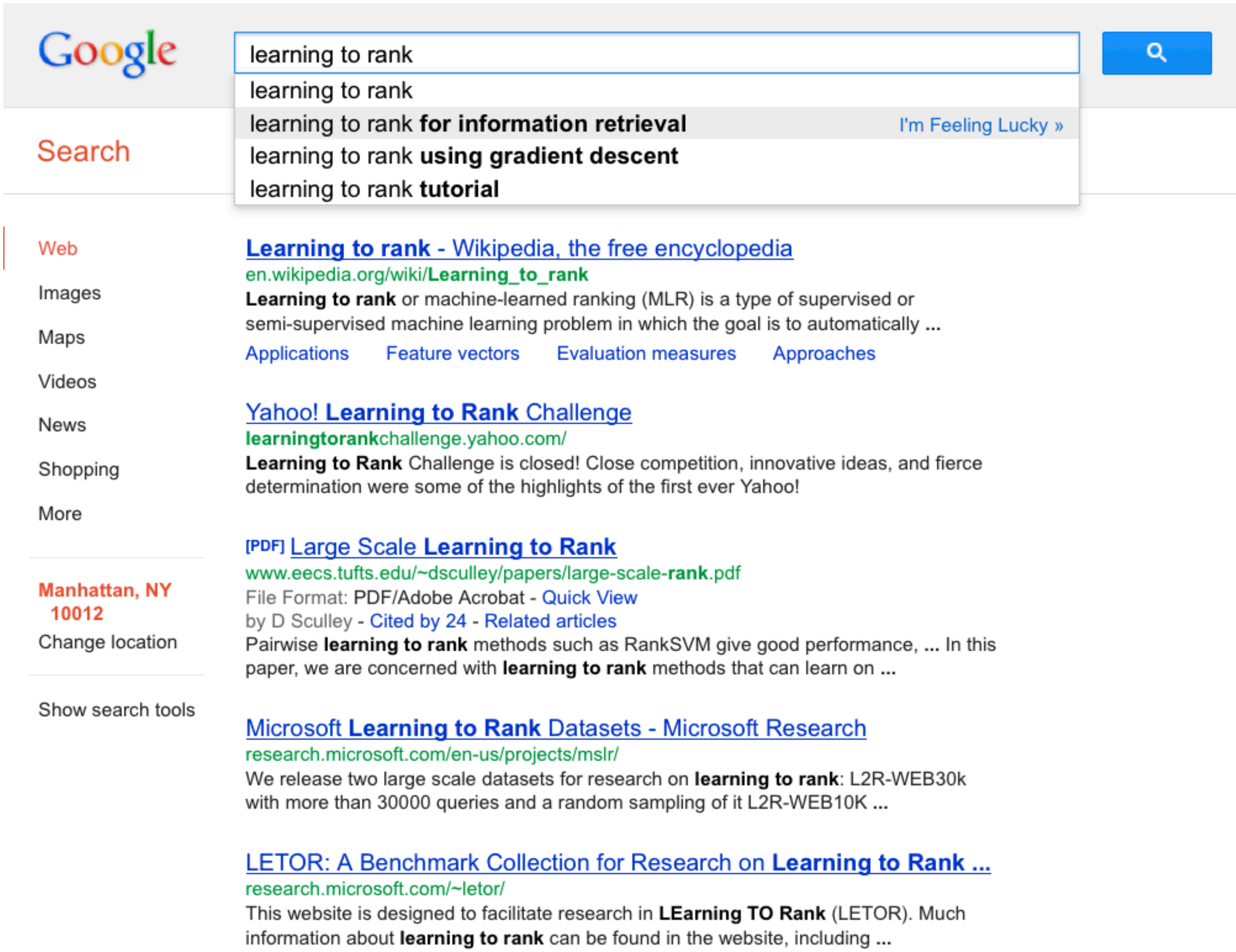
Temperature

72° F

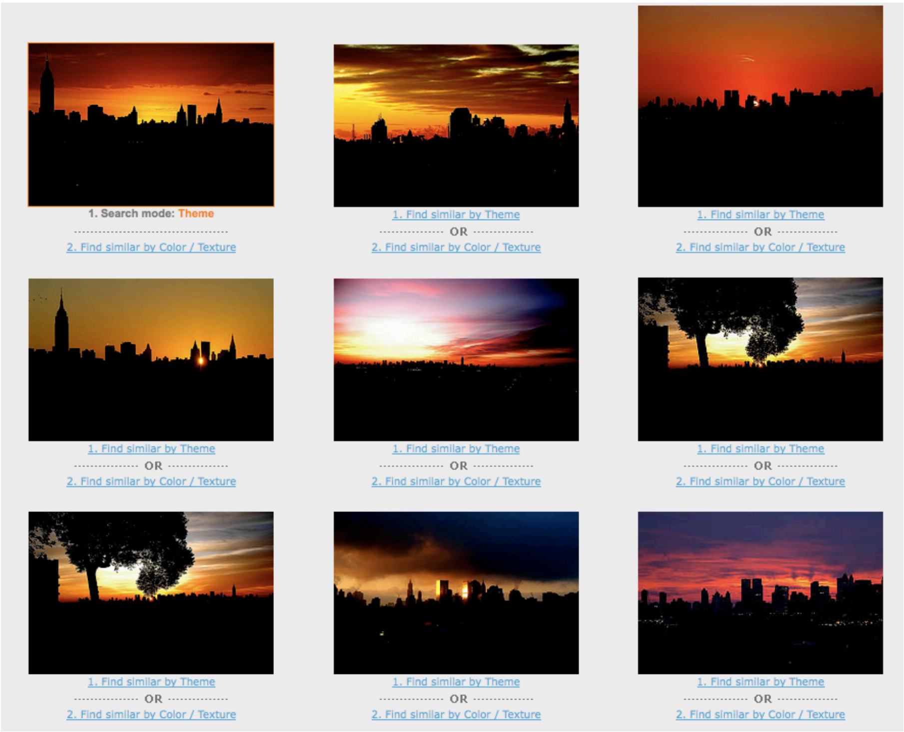
Ranking

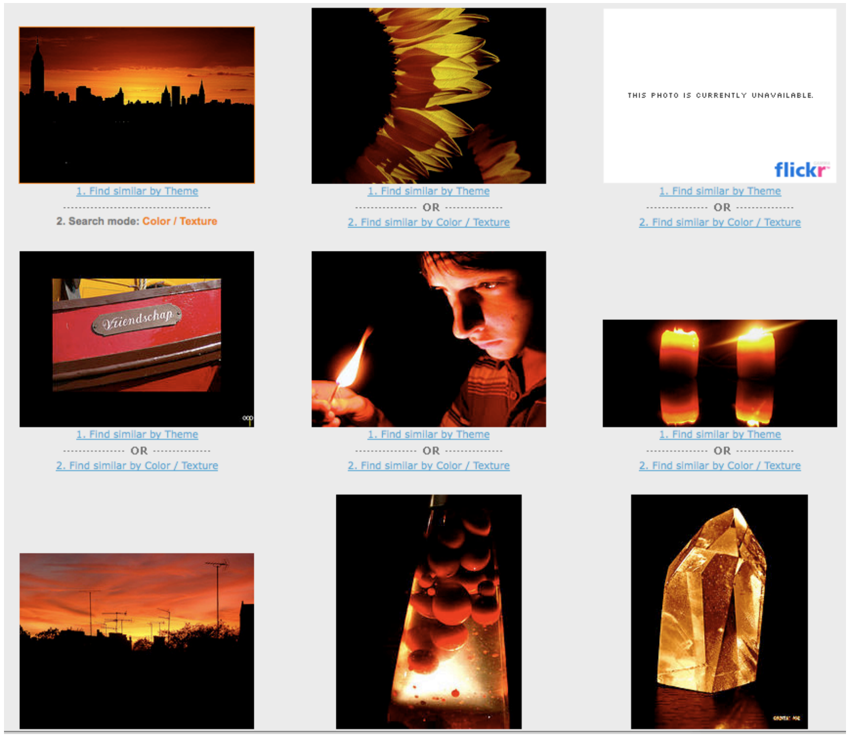
comparing items

Web search


Google
learning to rank
learning to rank
learning to rank for information retrieval
I'm Feeling Lucky »
learning to rank using gradient descent
learning to rank tutorial
Search
Web
Images
Maps
Videos
News
Shopping
More
Manhattan, NY 10012
Change location
Show search tools
Learning to rank - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/Learning_to_rank
Learning to rank or machine-learned ranking (MLR) is a type of supervised or semi-supervised machine learning problem in which the goal is to automatically ...
Applications Feature vectors Evaluation measures Approaches
Yahoo! Learning to Rank Challenge
learningtorankchallenge.yahoo.com/
Learning to Rank Challenge is closed! Close competition, innovative ideas, and fierce determination were some of the highlights of the first ever Yahoo!
[PDF] Large Scale Learning to Rank
www.eecs.tufts.edu/~dsculley/papers/large-scale-rank.pdf
File Format: PDF/Adobe Acrobat - Quick View
by D Sculley - Cited by 24 - Related articles
Pairwise learning to rank methods such as RankSVM give good performance, ... In this paper, we are concerned with learning to rank methods that can learn on ...
Microsoft Learning to Rank Datasets - Microsoft Research
research.microsoft.com/en-us/projects/mslr/
We release two large scale datasets for research on learning to rank: L2R-WEB30k with more than 30000 queries and a random sampling of it L2R-WEB10K ...
LETOR: A Benchmark Collection for Research on Learning to Rank ...
research.microsoft.com/~letor/
This website is designed to facilitate research in LEarning TO Rank (LETOR). Much information about learning to rank can be found in the website, including ...

Given image, find similar images

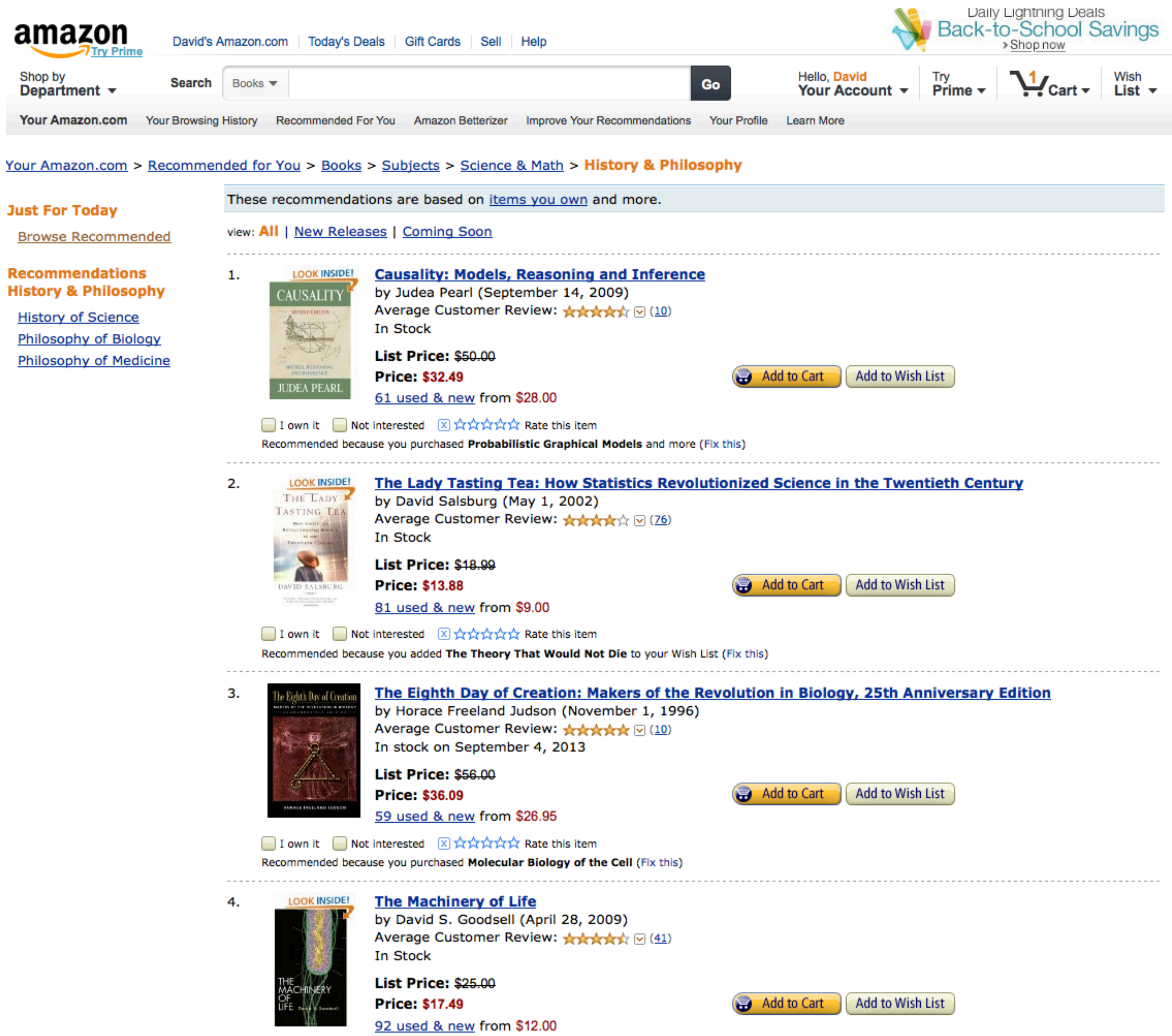




http://www.tiltomo.com/

Collaborative Filtering

Recommendation systems



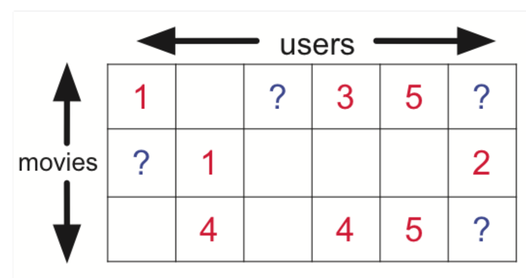
Recommendation systems

Machine learning competition with a $1 million prize

Leaderboard

Display top 20 leaders.

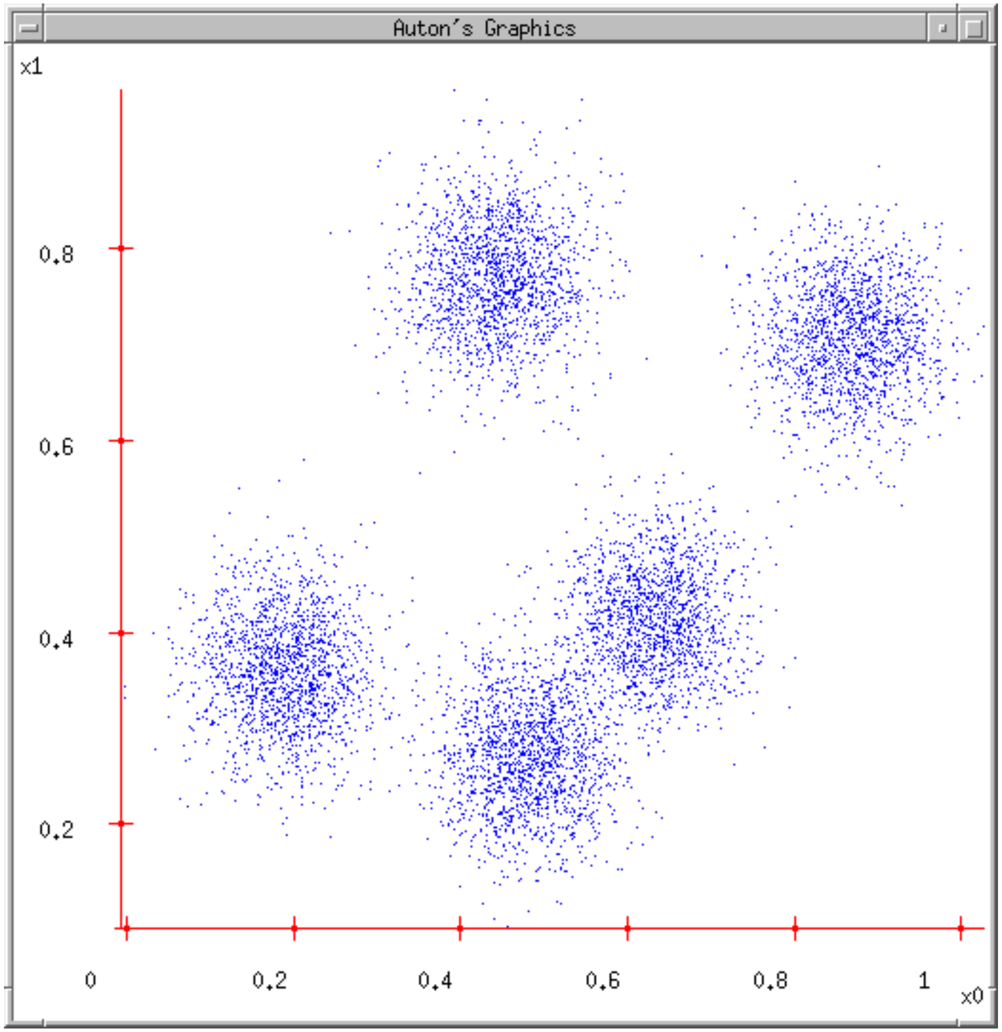
Rank	Team Name	Best Score	% Improvement	Last Submit Time
1	The Ensemble	0.8553	10.10	2009-07-26 18:38:22
2	BellKor's Pragmatic Chaos	0.8554	10.09	2009-07-26 18:18:28
Grand Prize - RMSE <= 0.8563				
3	Grand Prize Team	0.8571	9.91	2009-07-24 13:07:49
4	Opera Solutions and Vandelay United	0.8573	9.89	2009-07-25 20:05:52
5	Vandelay Industries !	0.8579	9.83	2009-07-26 02:49:53
6	PragmaticTheory	0.8582	9.80	2009-07-12 15:09:53
7	BellKor in BigChaos	0.8590	9.71	2009-07-26 12:57:25
8	Dace_	0.8603	9.58	2009-07-24 17:18:43
9	Opera Solutions	0.8611	9.49	2009-07-26 18:02:08
10	BellKor	0.8612	9.48	2009-07-26 17:19:11
11	BigChaos	0.8613	9.47	2009-06-23 23:06:52
12	Feeds2	0.8613	9.47	2009-07-24 20:06:46
Progress Prize 2008 - RMSE = 0.8616 - Winning Team: BellKor in BigChaos				
13	xiangliang	0.8633	9.26	2009-07-21 02:04:40
14	Gravity	0.8634	9.25	2009-07-26 15:58:34
15	Ces	0.8642	9.17	2009-07-25 17:42:38
16	Invisible Ideas	0.8644	9.14	2009-07-20 03:26:12
17	Just a guy in a garage	0.8650	9.08	2009-07-22 14:10:42
18	Craig Carmichael	0.8656	9.02	2009-07-25 16:00:54
19	J Dennis Su	0.8658	9.00	2009-03-11 09:41:54
20	acmehill	0.8659	8.99	2009-04-16 06:29:35
Progress Prize 2007 - RMSE = 0.8712 - Winning Team: KorBell				
Cinematch score on quiz subset - RMSE = 0.9514				



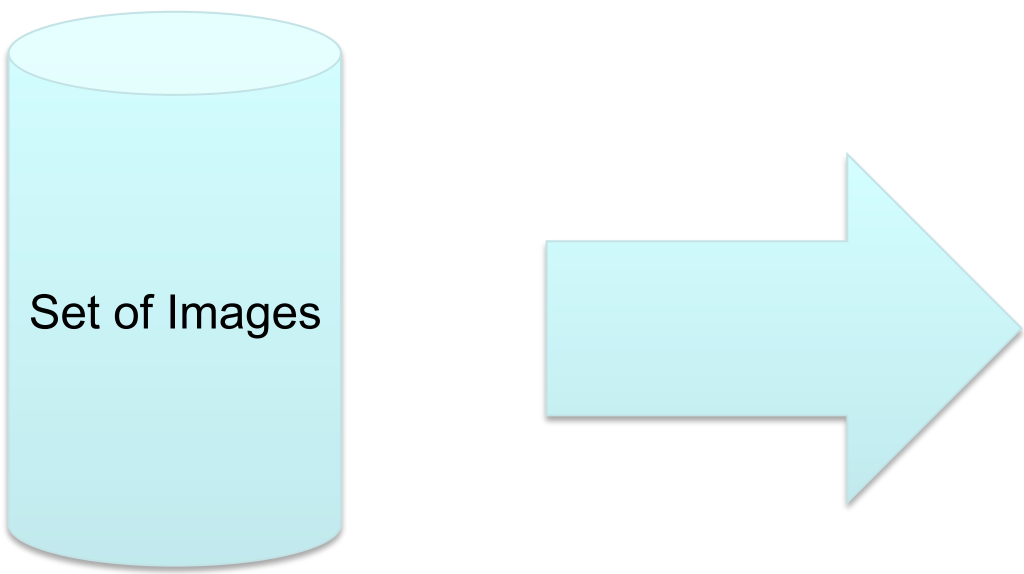
Clustering

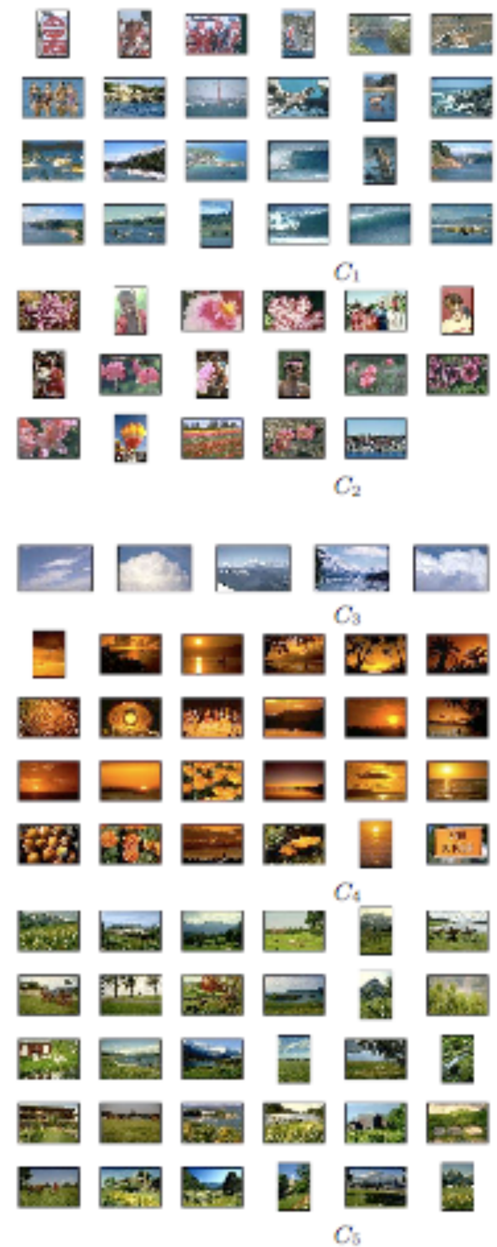
discovering structure in data

Clustering Data: Group similar things



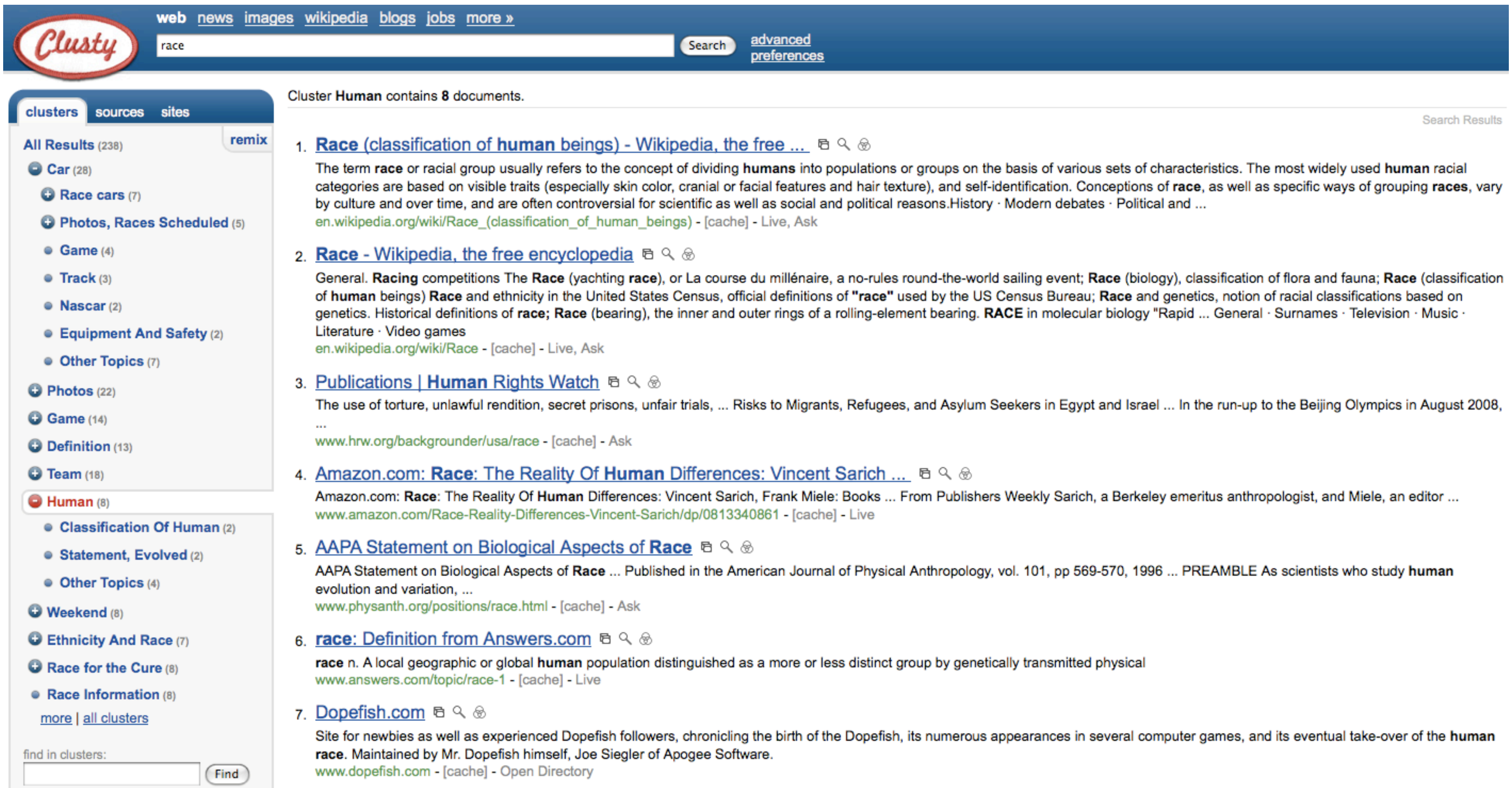
Clustering images





[Goldberger et al.]

Clustering web search results

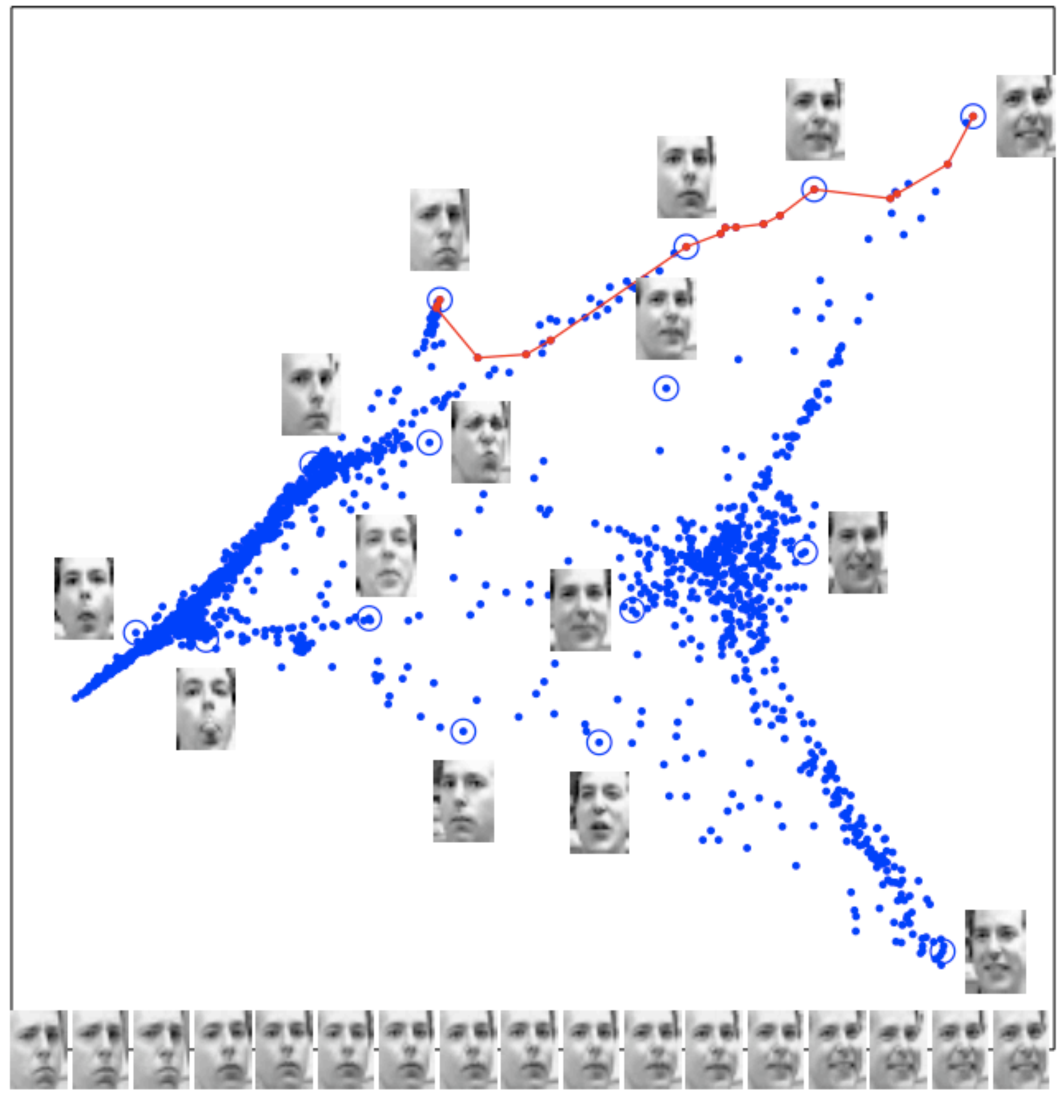


Embedding

visualizing data

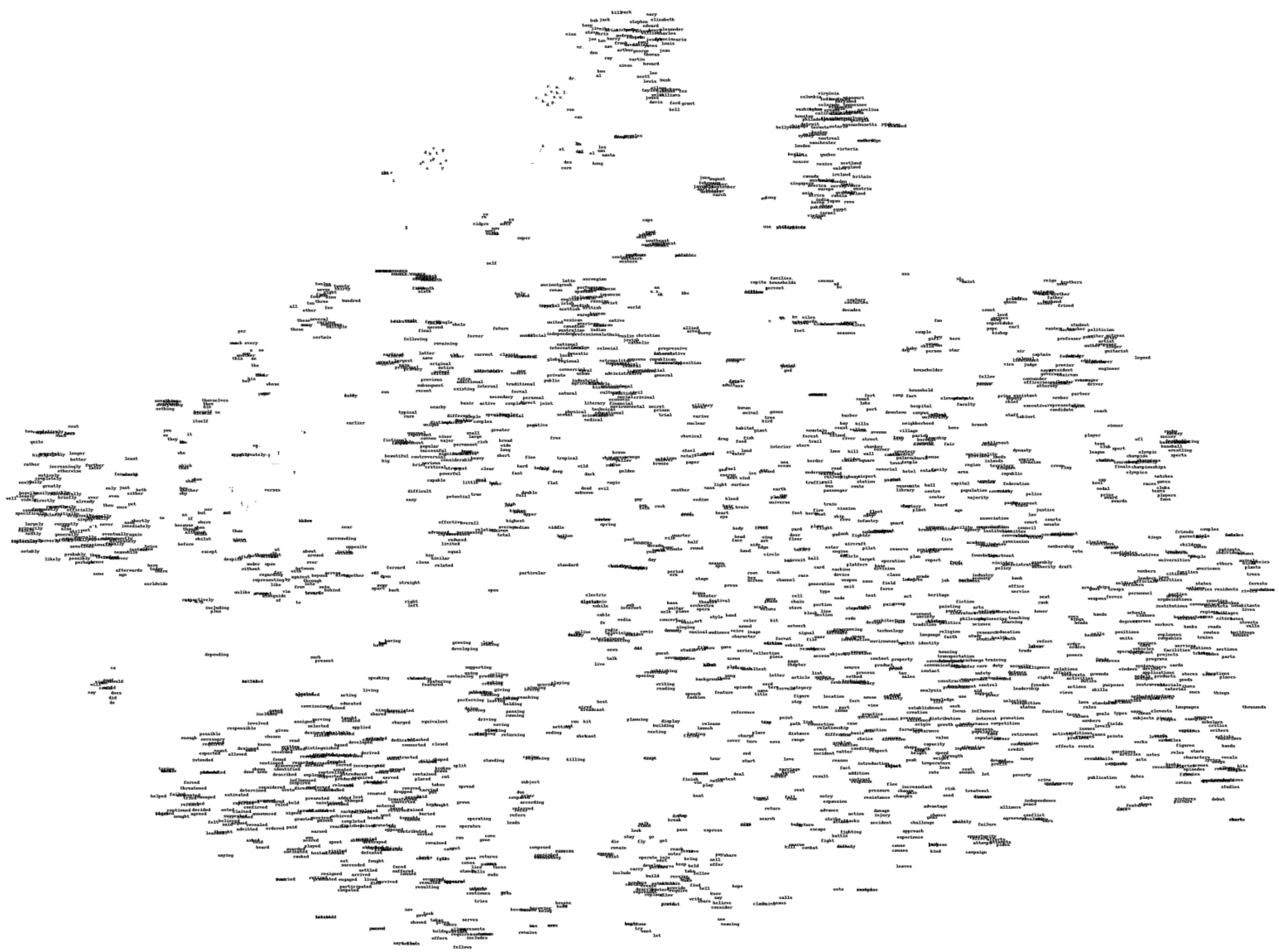
Embedding images

- Images have thousands or millions of pixels.
- Can we give each image a coordinate, such that similar images are near each other?



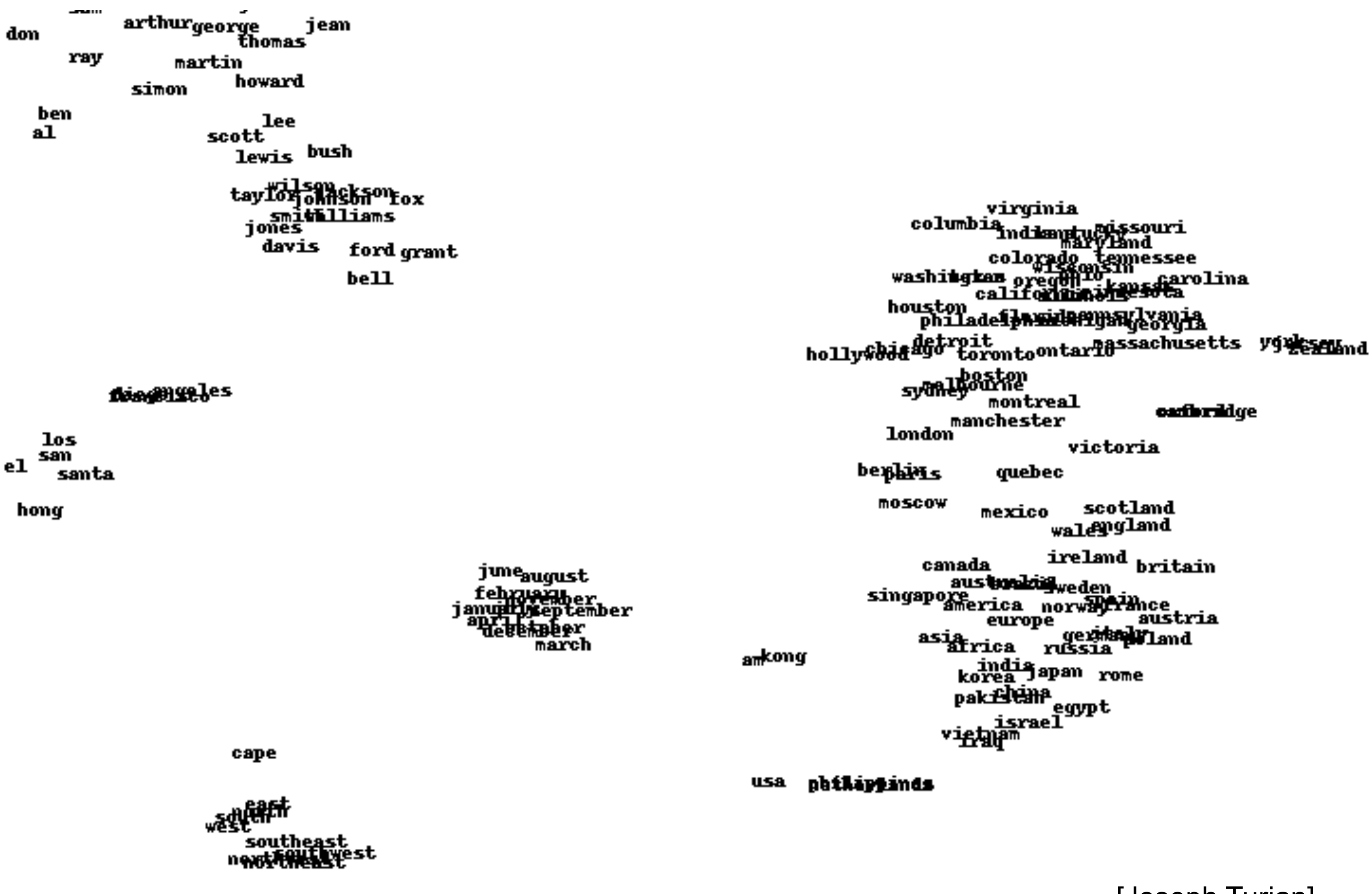
[Saul & Roweis '03]

Embedding words



[Joseph Turian]

Embedding words (zoom in)

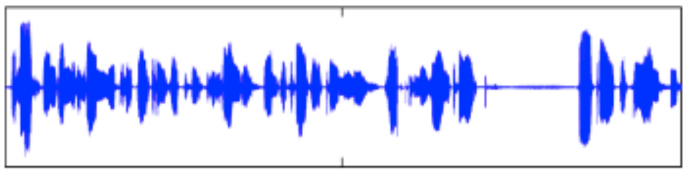


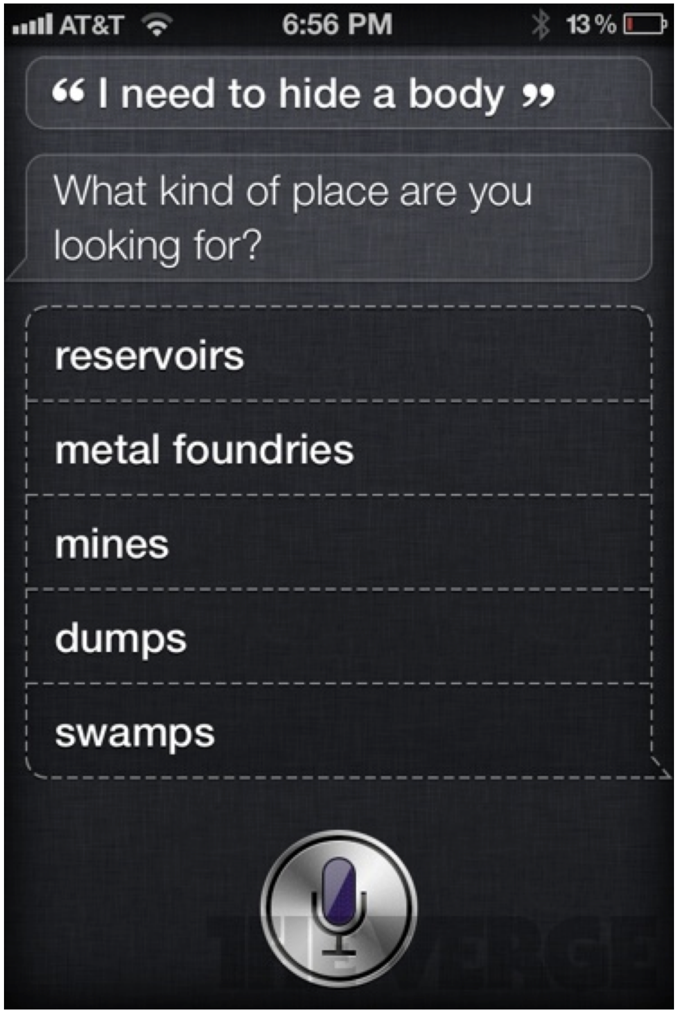
[Joseph Turian]

Structured prediction

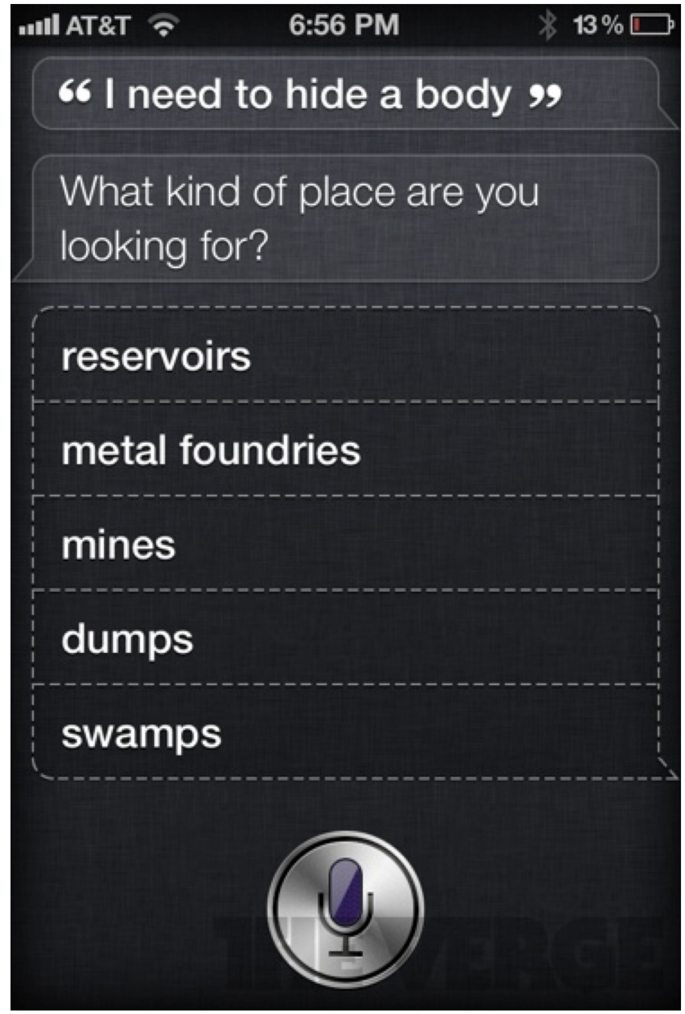
from data to discrete classes

Speech recognition





Natural language processing



I need to hide a body

noun, verb, preposition, …

Growth of Machine Learning

- Machine learning is preferred approach to
  - Speech recognition, Natural language processing
  - Computer vision
  - Medical outcomes analysis
  - Robot control
  - Computational biology
  - Sensor networks
  - …
- This trend is accelerating
  - Big data
  - Improved machine learning algorithms
  - Faster computers
  - Good open-source software

Course roadmap

- **First half of course: discriminative methods**
  - SVMs, kernel methods
  - Learning theory
  - Decision trees, boosting, deep learning
- **Second half of course: generative methods**
  - Graphical models, Gibbs sampling
  - Unsupervised learning, EM algorithm
  - Dimensionality reduction
  - LDA, topic models

Supervised Learning: find f

- Given: Training set $\{(x_i, y_i) \mid i = 1 \ldots N\}$
- Find: A good approximation to $f : X \rightarrow Y$

Examples: what are X and Y ?

- Spam Detection
  - Map email to {Spam, Not Spam}
- Digit recognition
  - Map pixels to {0,1,2,3,4,5,6,7,8,9}
- Stock Prediction
  - Map new, historic prices, etc. to $\Re$ (the real numbers)

A Supervised Learning Problem

Dataset:

Example	x_1	x_2	x_3	x_4	y
1	0	0	1	0	0
2	0	1	0	0	0
3	0	0	1	1	1
4	1	0	0	1	1
5	0	1	1	0	0
6	1	1	0	0	0
7	0	1	0	1	0

- Our goal is to find a function $f : X \rightarrow Y$
  - $X = \{0,1\}^4$
  - $Y = \{0,1\}$
- Question 1: How should we pick the *hypothesis space*, the set of possible functions f?
- Question 2: How do we find the best f in the hypothesis space?

Most General Hypothesis Space

Consider all possible boolean functions over four input features!

- 2^{16} possible hypotheses
- 2^{9} are consistent with our dataset
- How do we choose the best one?

x_1	x_2	x_3	x_4	y
0	0	0	0	?
0	0	0	1	?
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	?
1	0	0	0	?
1	0	0	1	1
1	0	1	0	?
1	0	1	1	?
1	1	0	0	0
1	1	0	1	?
1	1	1	0	?
1	1	1	1	?

Dataset:

Example	x_1	x_2	x_3	x_4	y
1	0	0	1	0	0
2	0	1	0	0	0
3	0	0	1	1	1
4	1	0	0	1	1
5	0	1	1	0	0
6	1	1	0	0	0
7	0	1	0	1	0

A Restricted Hypothesis Space

Consider all conjunctive boolean functions.

- 16 possible hypotheses
- None are consistent with our dataset
- How do we choose the best one?

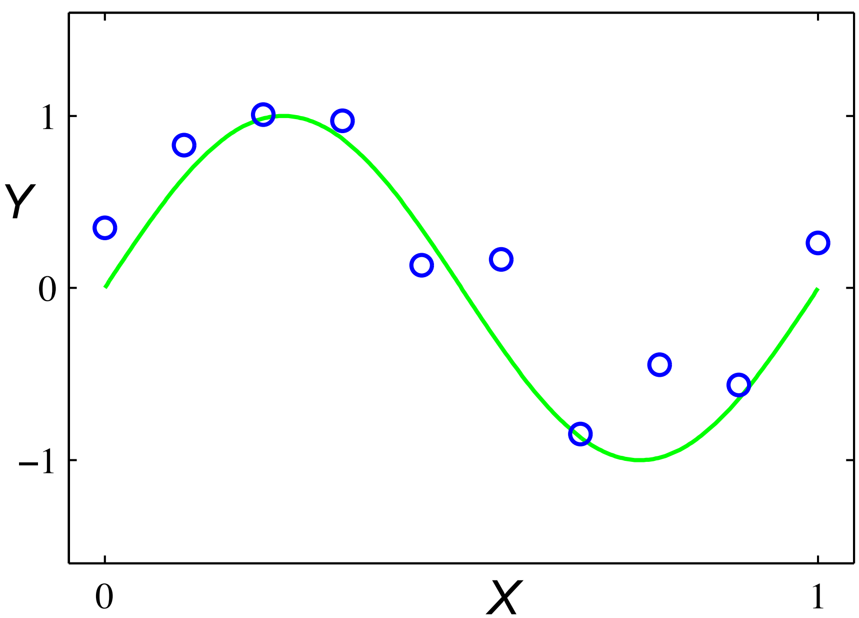
Rule	Counterexample
$\Rightarrow y$	1
$x_1 \Rightarrow y$	3
$x_2 \Rightarrow y$	2
$x_3 \Rightarrow y$	1
$x_4 \Rightarrow y$	7
$x_1 \wedge x_2 \Rightarrow y$	3
$x_1 \wedge x_3 \Rightarrow y$	3
$x_1 \wedge x_4 \Rightarrow y$	3
$x_2 \wedge x_3 \Rightarrow y$	3
$x_2 \wedge x_4 \Rightarrow y$	3
$x_3 \wedge x_4 \Rightarrow y$	4
$x_1 \wedge x_2 \wedge x_3 \Rightarrow y$	3
$x_1 \wedge x_2 \wedge x_4 \Rightarrow y$	3
$x_1 \wedge x_3 \wedge x_4 \Rightarrow y$	3
$x_2 \wedge x_3 \wedge x_4 \Rightarrow y$	3
$x_1 \wedge x_2 \wedge x_3 \wedge x_4 \Rightarrow y$	3

Dataset:

Example	x_1	x_2	x_3	x_4	y
1	0	0	1	0	0
2	0	1	0	0	0
3	0	0	1	1	1
4	1	0	0	1	1
5	0	1	1	0	0
6	1	1	0	0	0
7	0	1	0	1	0

Second example: Regression

Dataset: 10 (X,Y) points generated from a sin function, with noise



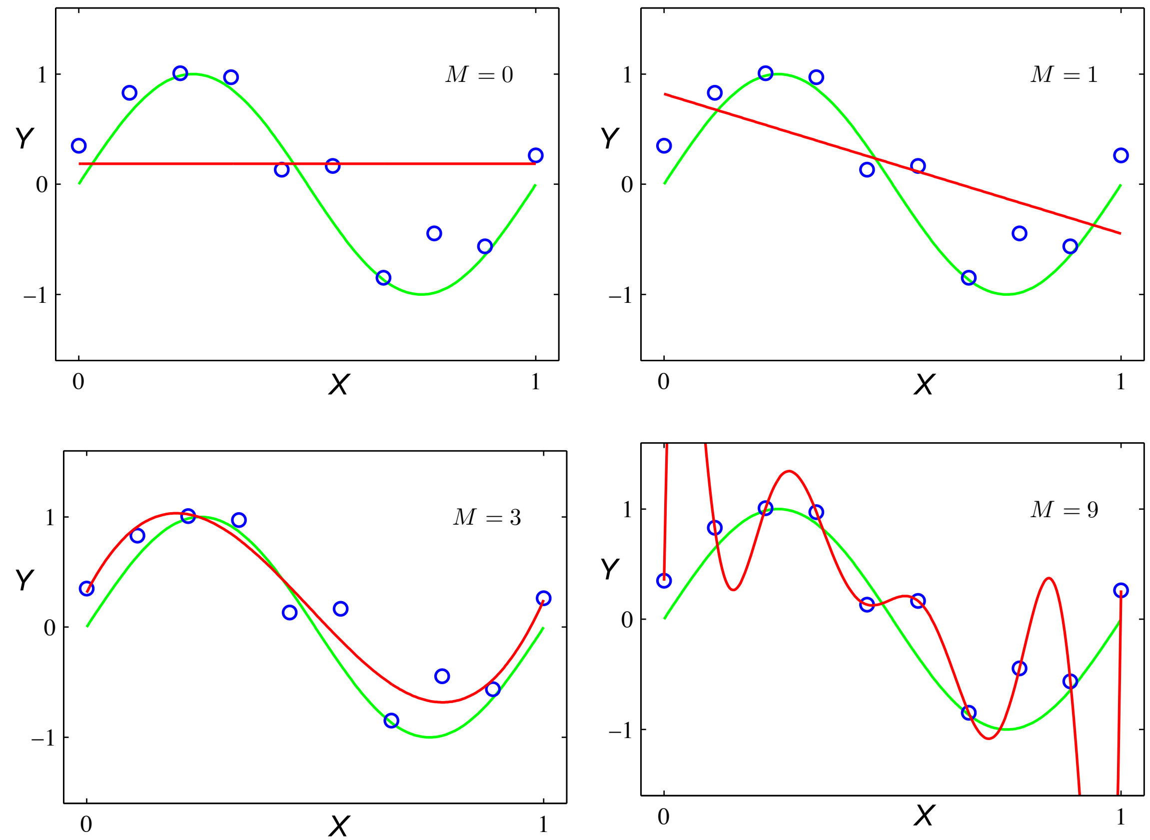
- Regression:
  - $f : X \rightarrow Y$
  - $X = \Re$
  - $Y = \Re$

[Bishop]

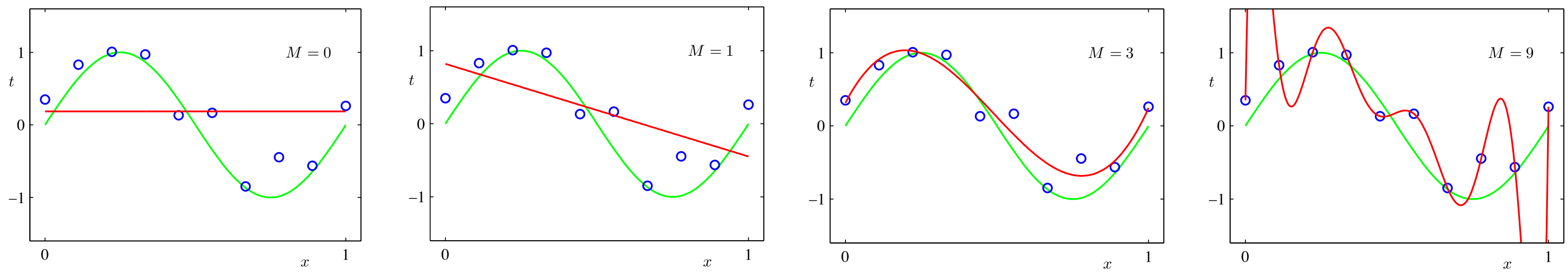
Degree-M Polynomials

How about letting *f* be a degree M polynomial?

- Which one is **best**?



Hypo. Space: Degree-N Polynomials



We measure error using a *loss function* $L(y, \hat{y})$

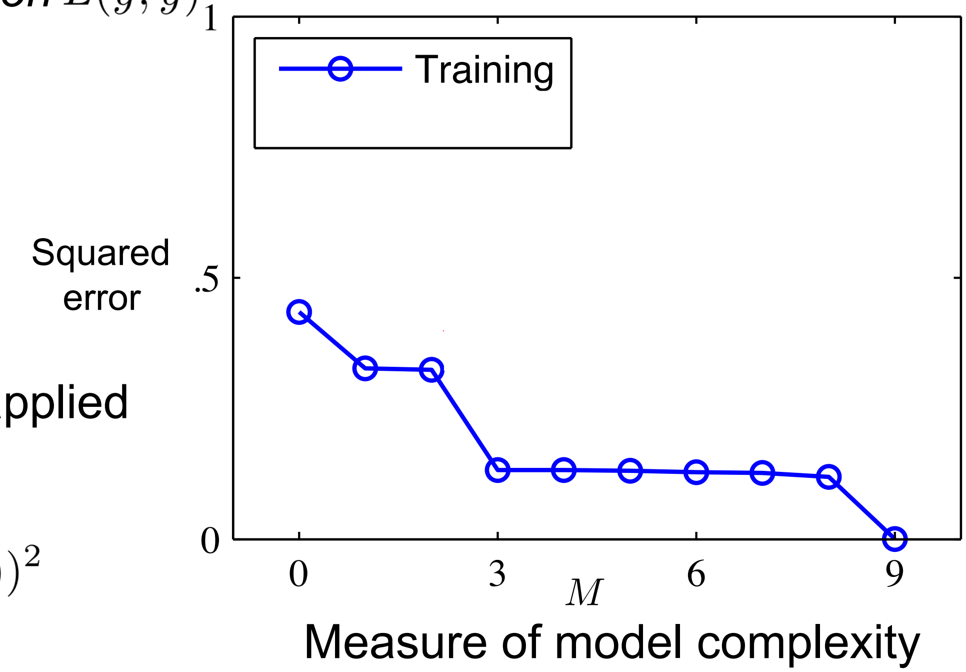
For regression, a common choice is squared loss:

$$L(y_i, f(x_i)) = (y_i - f(x_i))^2$$

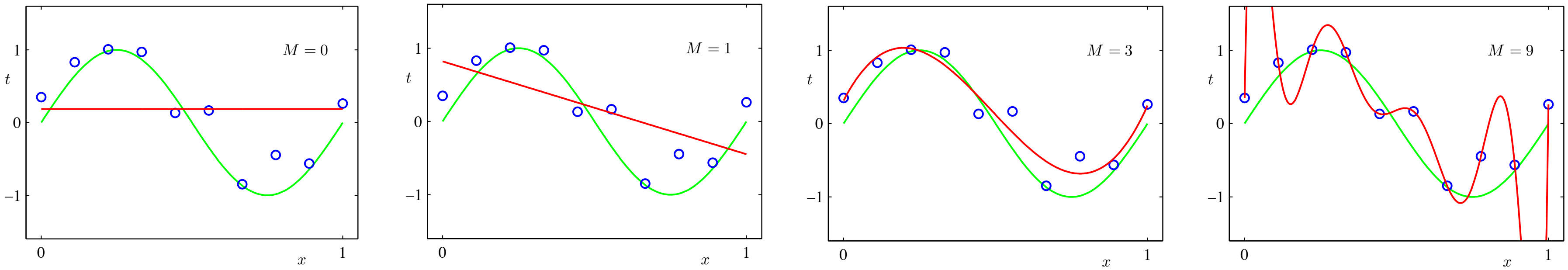
The *empirical loss* of the function *f* applied to the training data is then:

$$\frac{1}{N}\sum_{i=1}^{N} L(y_i, f(x_i)) = \frac{1}{N}\sum_{i=1}^{N}(y_i - f(x_i))^2$$

Learning curve



Hypo. Space: Degree-N Polynomials



We measure error using a *loss function* $L(y, \hat{y})$

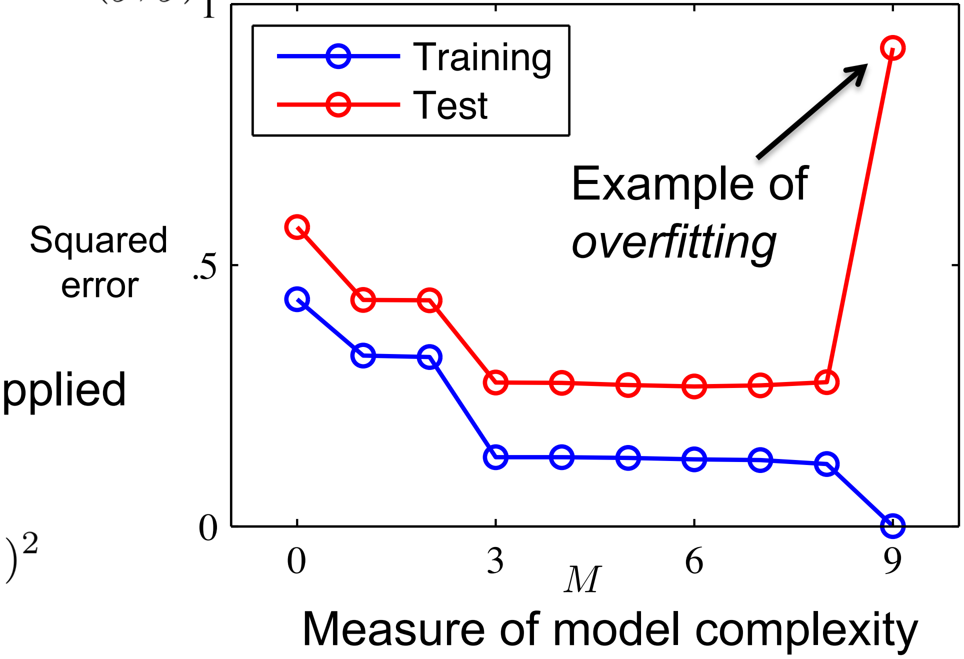
For regression, a common choice is squared loss:

$$L(y_i, f(x_i)) = (y_i - f(x_i))^2$$

The *empirical loss* of the function *f* applied to the training data is then:

$$\frac{1}{N}\sum_{i=1}^{N} L(y_i, f(x_i)) = \frac{1}{N}\sum_{i=1}^{N}(y_i - f(x_i))^2$$

Learning curve



Occam's Razor Principle

- William of Occam: Monk living in the 14th century
- Principle of parsimony:

"One should not increase, beyond what is necessary, the number of entities required to explain anything"

- When many solutions are available for a given problem, we should select the simplest one
- But what do we mean by simple?
- We will use prior knowledge of the problem to solve to define what is a simple solution

Example of a prior: smoothness

[Samy Bengio]

Key Issues in Machine Learning

- How do we choose a hypothesis space?
  - Often we use **prior knowledge** to guide this choice
- How can we gauge the accuracy of a hypothesis on unseen data?
  - **Occam's razor:** use the *simplest* hypothesis consistent with data! This will help us avoid overfitting.
  - ***Learning theory*** will help us quantify our ability to ***generalize*** as a function of the amount of training data and the hypothesis space
- How do we find the best hypothesis?
  - This is an **algorithmic** question, the main topic of computer science
- How to model applications as machine learning problems? (engineering challenge)

Binary classification

- Input: email
- Output: spam/ham
- Setup:
  – Get a large collection of example emails, each labeled "spam" or "ham"
  – Note: someone has to hand label all this data!
  – Want to learn to predict labels of new, future emails
- Features: The attributes used to make the ham / spam decision
  – Words: FREE!
  – Text Patterns: $dd, CAPS
  – Non-text: SenderInContacts
  – …

Dear Sir.

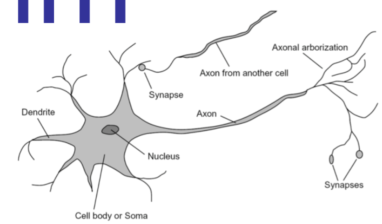
First, I must solicit your confidence in this transaction, this is by virture of its nature as being utterly confidencial and top secret. …

TO BE REMOVED FROM FUTURE MAILINGS, SIMPLY REPLY TO THIS MESSAGE AND PUT "REMOVE" IN THE SUBJECT.

99 MILLION EMAIL ADDRESSES FOR ONLY $99

Ok, Iknow this is blatantly OT but I'm beginning to go insane. Had an old Dell Dimension XPS sitting in the corner and decided to put it to use, I know it was working pre being stuck in the corner, but when I plugged it in, hit the power nothing happened.

The perceptron algorithm



- 1957: Perceptron algorithm invented by Rosenblatt

  Wikipedia: "A handsome bachelor, he drove a classic MGA sports… for several years taught an interdisciplinary undergraduate honors course entitled "Theory of Brain Mechanisms" that drew students equally from Cornell's Engineering and Liberal Arts colleges…this course was a melange of ideas .. experimental brain surgery on epileptic patients while conscious, experiments on .. the visual cortex of cats, ... analog and digital electronic circuits that modeled various details of neuronal behavior (i.e. the perceptron itself, as a machine)."

  – Built on work of Hebbs (1949); also developed by Widrow-Hoff (1960)

- 1960: Perceptron Mark 1 Computer – hardware implementation

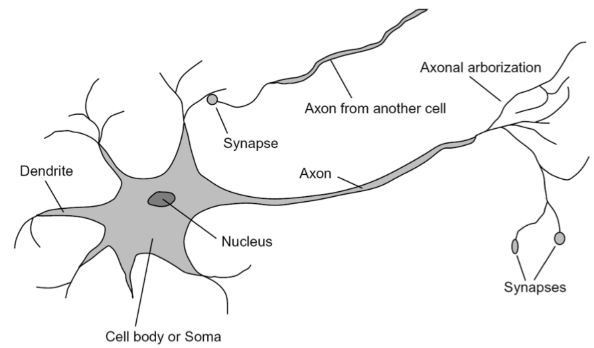
- 1969: Minksky & Papert book shows perceptrons limited to *linearly separable* data, and Rosenblatt dies in boating accident

- 1970's: Learning methods for two-layer neural networks

[William Cohen]

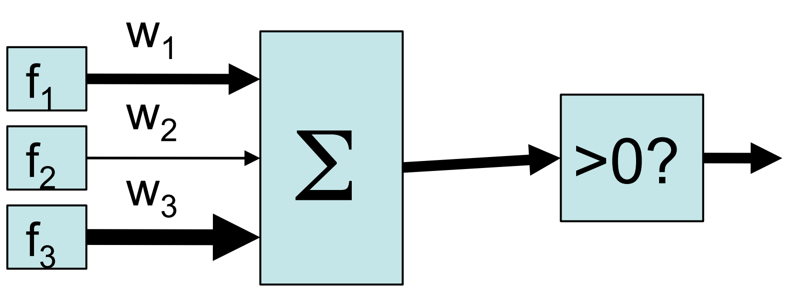
Linear Classifiers

- Inputs are feature values
- Each feature has a weight
- Sum is the activation



$$\text{activation}_w(x) = \sum_i w_i \cdot f_i(x) = w \cdot f(x)$$

- If the activation is:
  - Positive, output *class 1*
  - Negative, output *class 2*



Example: Spam

- Imagine 3 features (spam is "positive" class):
  1. free (number of occurrences of "free")
  2. money (occurrences of "money")
  3. BIAS (intercept, always has value 1)

$$w \cdot f(x)$$

$$\sum_i w_i \cdot f_i(x)$$

x	$f(x)$	w
"free money"	BIAS : 1 free : 1 money : 1 ...	BIAS : -3 free : 4 money : 2 ...

$$\begin{aligned} &(1)(-3) \quad + \\ &(1)(4) \quad + \\ &(1)(2) \quad + \\ &\ldots \\ &= 3 \end{aligned}$$

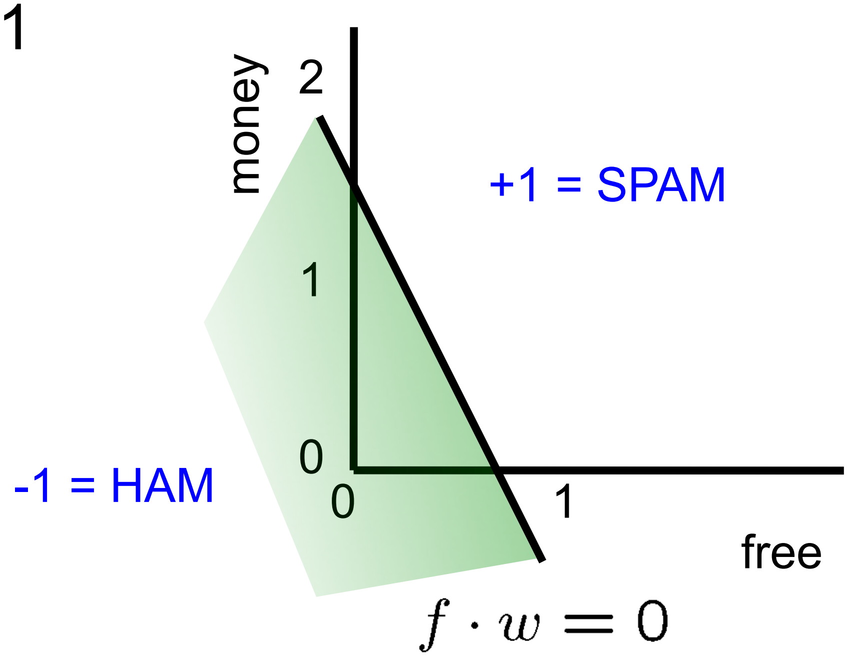
w.f(x) > 0 ➔ SPAM!!!

Binary Decision Rule

- In the space of feature vectors
  - Examples are points
  - Weight vector and bias define a hyperplane
  - One side corresponds to Y=+1
  - Other corresponds to Y=-1

w

```
BIAS  : -3
free  :  4
money :  2
...
```



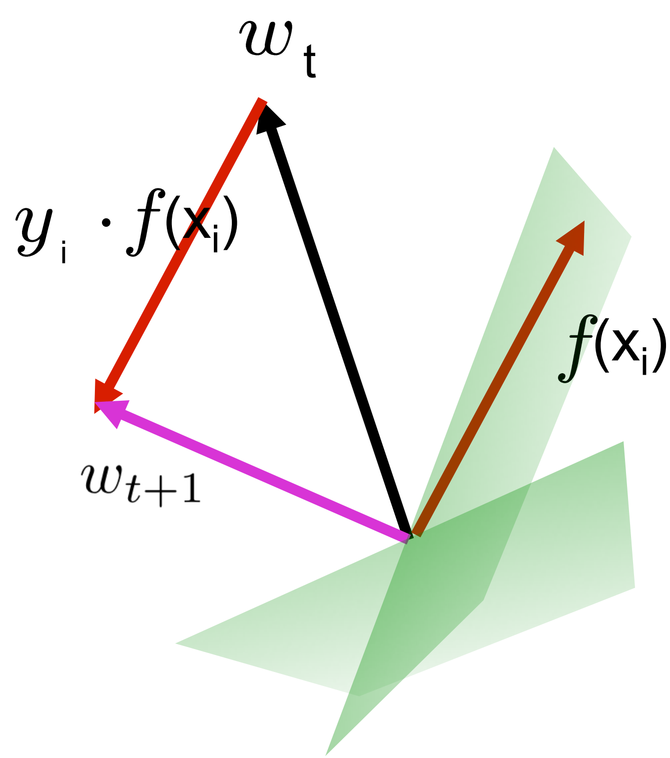
The perceptron algorithm

- Start with weight vector = $\vec{0}$
- For each training instance (x_i, y_i):
  - Classify with current weights

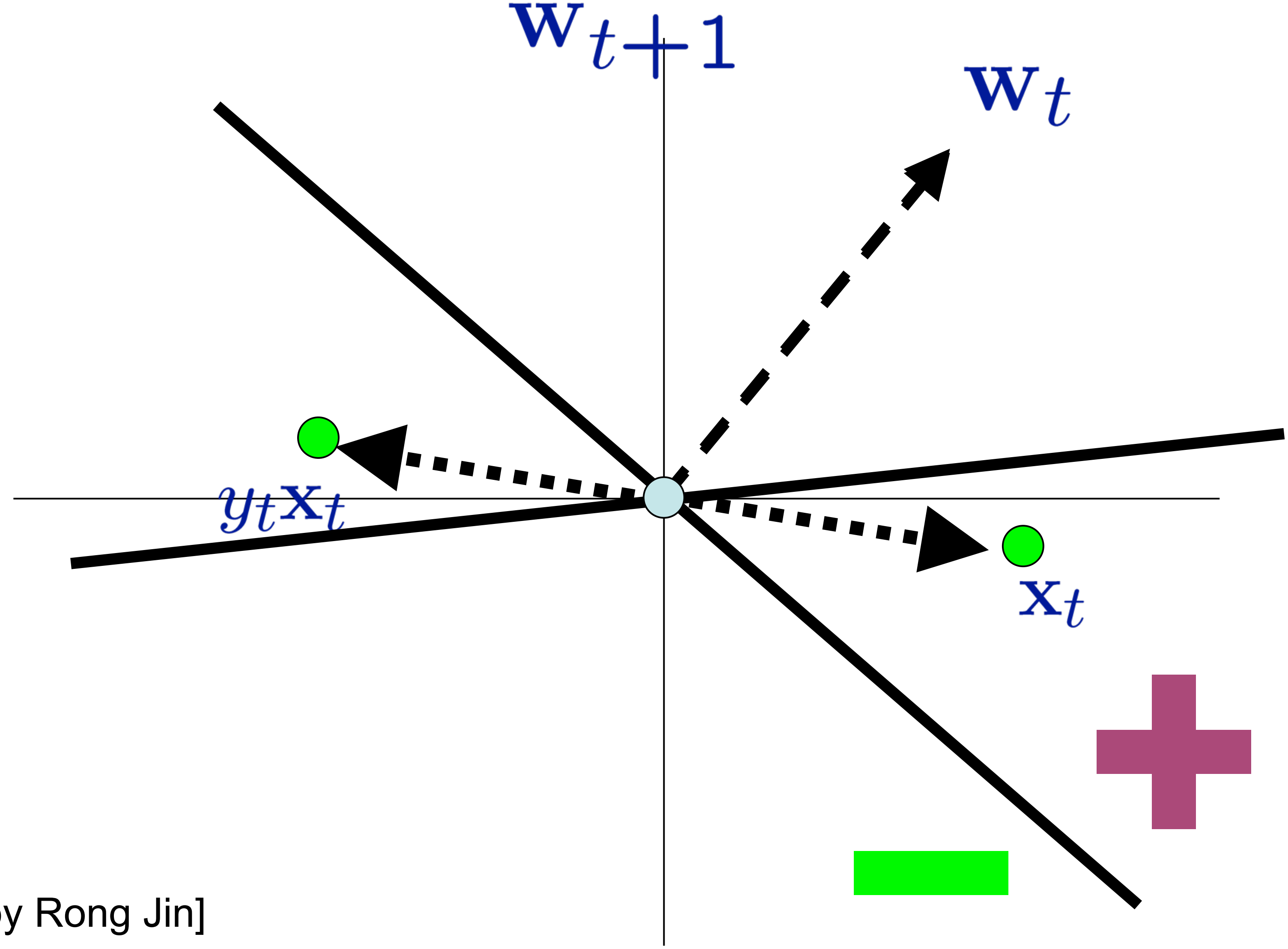
$$y = \begin{cases} +1 & \text{if } \; w \cdot f(x_i) \geq 0 \\ -1 & \text{if } \; w \cdot f(x_i) < 0 \end{cases}$$

  - If correct (i.e., $y=y_i$), no change!
  - If wrong: update

$$w = w + y_i \, f(x_i)$$



Geometrical Interpretation



[Slide by Rong Jin]

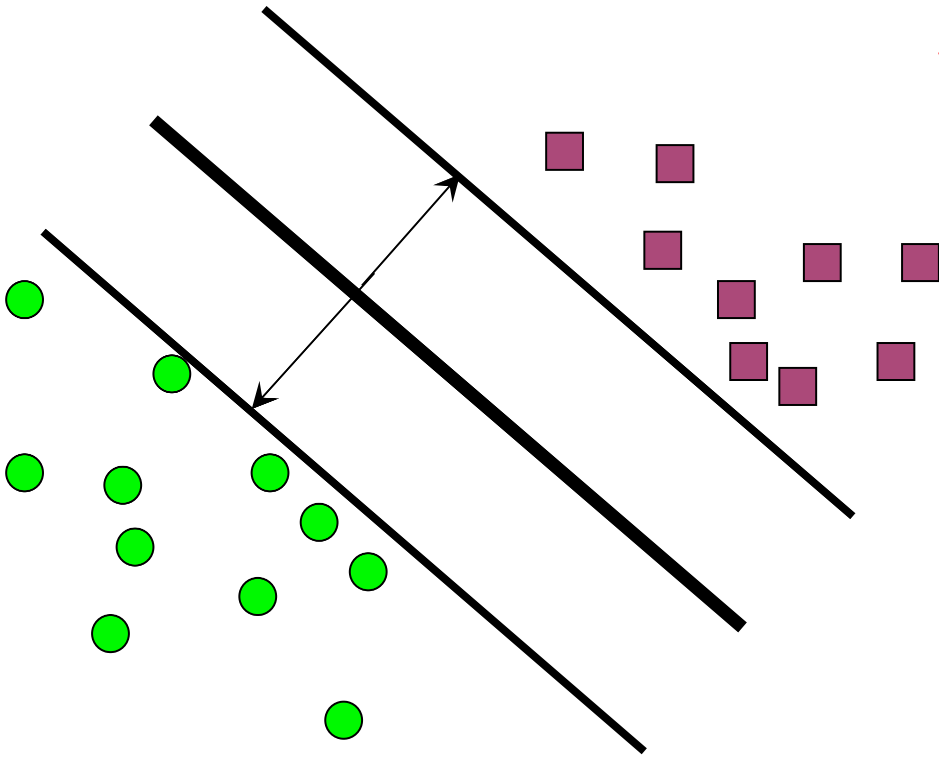
What questions should we ask about a learning algorithm?

- What is the perceptron algorithm's running time?
- If a weight vector with small training error exists, will perceptron find it?
- How well does the resulting classifier generalize to unseen data?

Linearly Separable

$\exists \mathbf{w}$ such that $\forall t$ $\qquad y_t(\mathbf{w} \cdot \mathbf{x}_t) \geq \gamma > 0$

Called the *functional margin* with respect to the training set



Equivalently, for $y_t = +1$,

$$w \cdot x_t \geq \gamma$$

and for $y_t = -1$,

$$w \cdot x_t \leq -\gamma$$

Mistake Bound for Perceptron

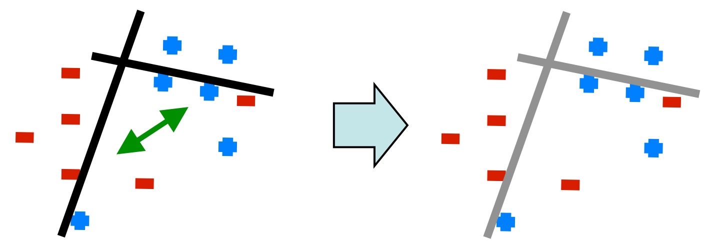
- Assume the data set *D* is linearly separable with *geometric* margin γ, i.e.,

$$\exists w^* \text{ s.t. } \|w^*\|_2 = 1 \text{ and } \forall t,\ y_t(w \cdot x_t) \geq \gamma$$

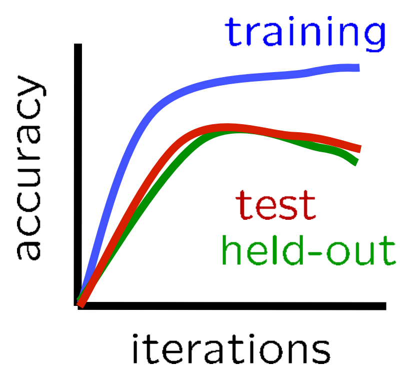
- Assume $\|x_t\|_2 \leq R, \forall t$
- Theorem: The maximum number of mistakes made by the perceptron algorithm is bounded by R^2/γ^2

Problems with the perceptron algorithm

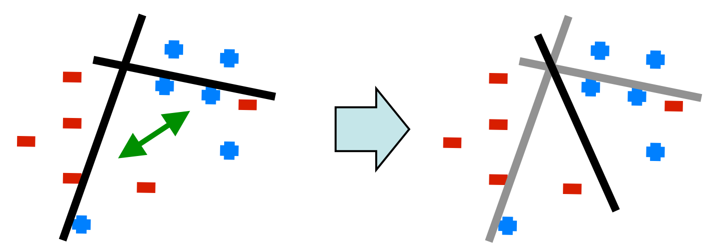
- If the data isn't linearly separable, no guarantees of convergence or training accuracy



- Even if the training data is linearly separable, perceptron can overfit

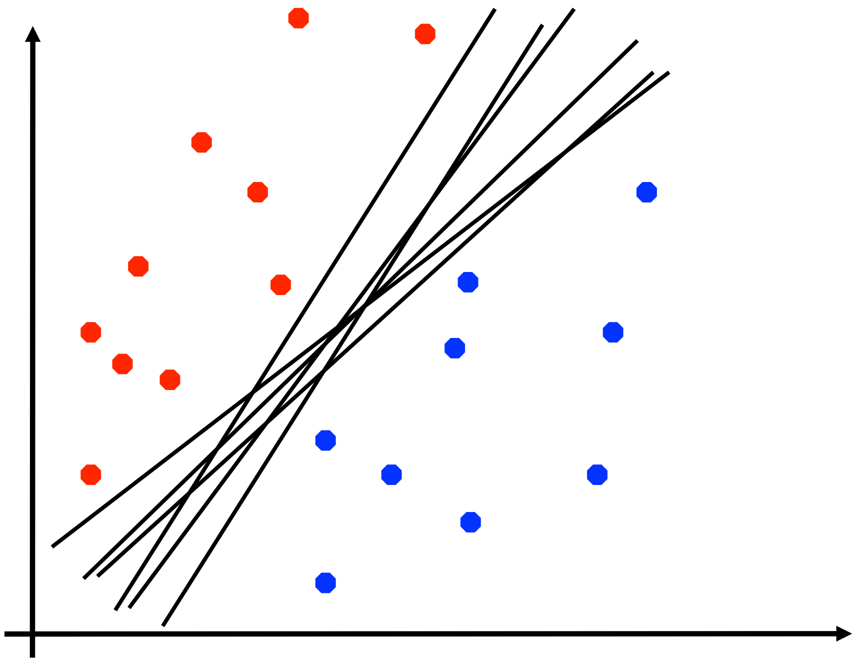


- **Averaged** perceptron is an algorithmic modification that helps with both issues
  - Averages the weight vectors across all iterations



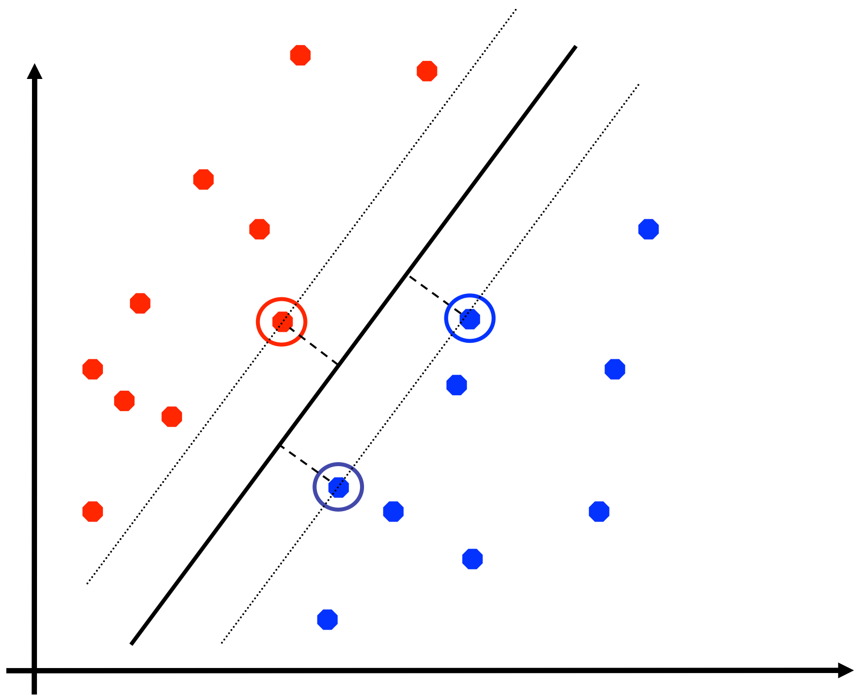
Linear Separators

- Which of these linear separators is optimal?



Next week: Support Vector Machines

- SVMs (Vapnik, 1990's) choose the linear separator with the **largest margin**



- Good according to intuition, theory, practice