Artificial Intelligence

CS 165A

Jan 24, 2018

Instructor: Prof. Yu-Xiang Wang

Today

→ Machine learning Overview

→ Supervised learning

Announcements

- Homework 1 due next week on Jan 29, 2018.
- Machine Problem 1 is posted on Tuesday

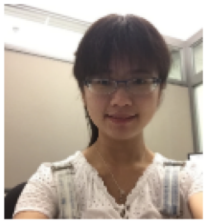
Plan for the next four lectures: Intro to Machine Learning

- Today: ML Overview / Supervised Learning
- Jan 29: Unsupervised Learning
- Jan 31: Continuous optimization
- Feb 5: Statistical Learning Theory (How does machine learning work?)

What's the difference from CS165B?

Week	Date	Topic
1	7-Jan	Introduction and overview
	9-Jan	Decision Trees (updated)
2	14-Jan	Point Estimation
	16-Jan	Point Estimation (updated)
3	21-Jan	*Holiday - no class*
	23-Jan	Linear Regression
4	28-Jan	Naive Bayes
	30-Jan	Logistic Regression
5	4-Feb	Perceptrons, Support Vector Machines
	6-Feb	Kernel Methods
6	11-Feb	Ensemble Learning
	13-Feb	Clustering (KMeans, EM)
7	18-Feb	*Holiday - no class*
	20-Feb	Clustering (KMeans, EM)
8	25-Feb	Features Reduction, PCA
	27-Feb	Non-parametric methods
9	4-Mar	Probablistic Graphical Models
	6-Mar	Collaborative Filtering
10	11-Mar	Neural networks (CNN)
	13-Mar	Neural networks (LSTM)

2019 Winter



Yufei Ding
Assistant Professor
Department of Computer Science
University of California, Santa Barbara

2019 Spring

William Wang
Assistant Professor
Department of Computer Science
University of California, Santa Barbara

This course: ML's place in AI

Focus on understanding:

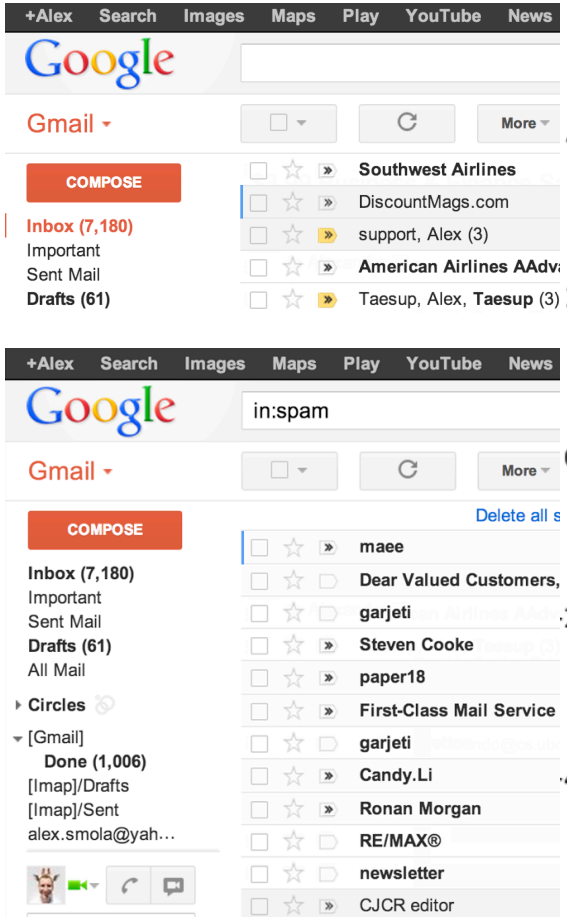
ML's strengths and limitations.

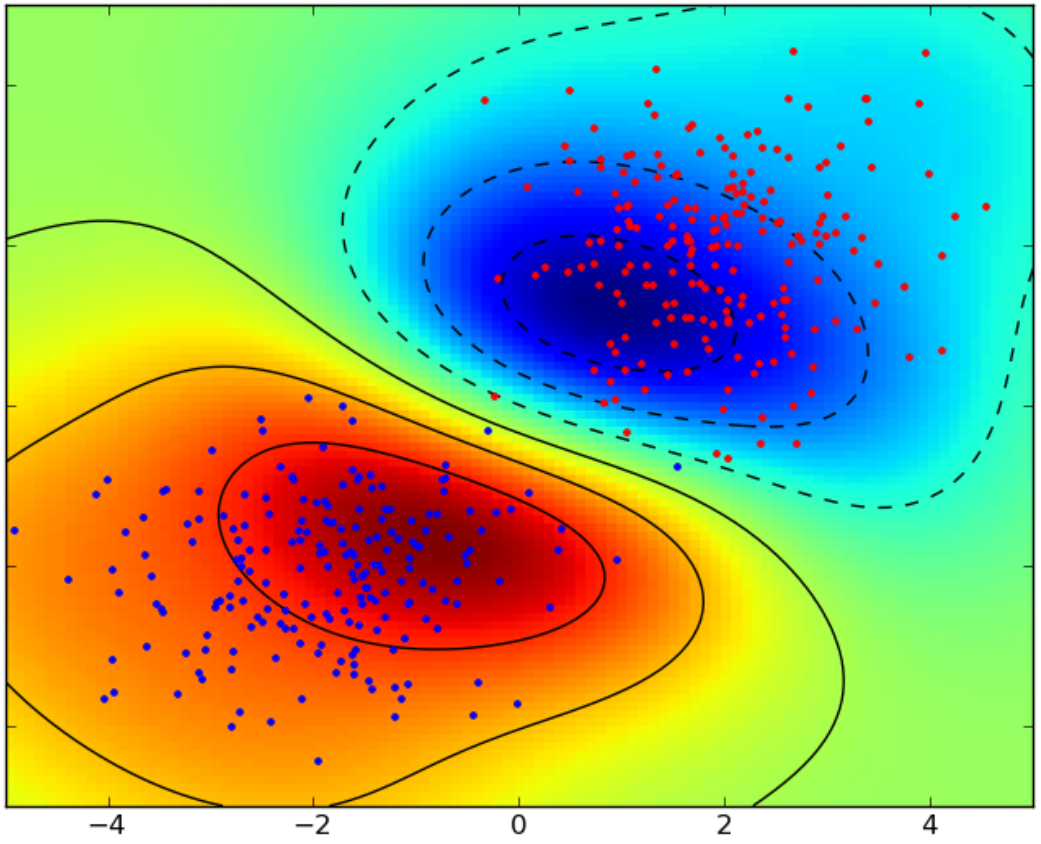
Machine Learning

- Supervised Learning — Spam Filter.
- Unsupervised Learning — Topics of a body of texts
- Reinforcement Learning — Atari Games. Serve Ads.
- Structured Prediction — Machine translation.

Bandits and reinforcement learning after the midterm.

Supervised learning



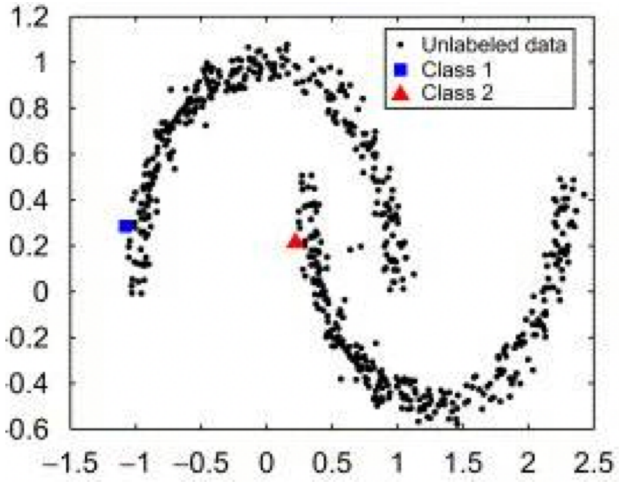


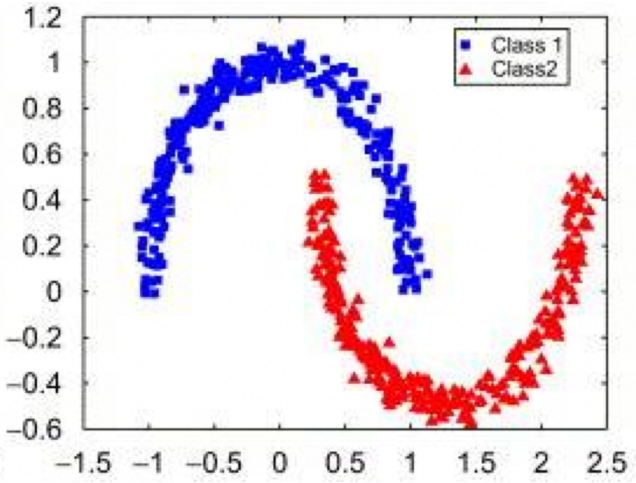
Unsupervised Learning

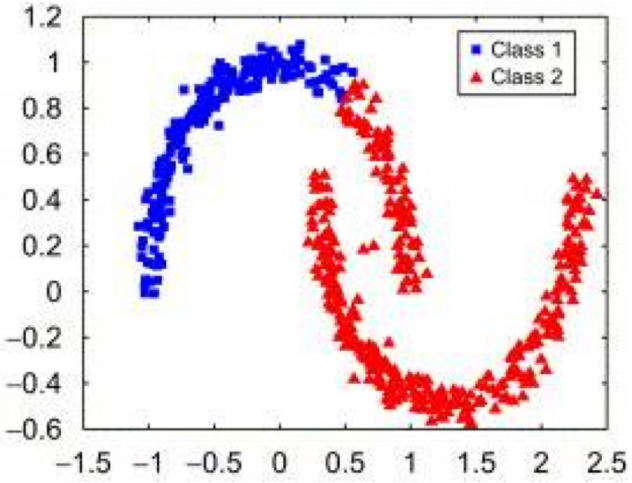
"Arts"	"Budgets"	"Children"	"Education"
NEW	MILLION	CHILDREN	SCHOOL
FILM	TAX	WOMEN	STUDENTS
SHOW	PROGRAM	PEOPLE	SCHOOLS
MUSIC	BUDGET	CHILD	EDUCATION
MOVIE	BILLION	YEARS	TEACHERS
PLAY	FEDERAL	FAMILIES	HIGH
MUSICAL	YEAR	WORK	PUBLIC
BEST	SPENDING	PARENTS	TEACHER
ACTOR	NEW	SAYS	BENNETT
FIRST	STATE	FAMILY	MANIGAT
YORK	PLAN	WELFARE	NAMPHY
OPERA	MONEY	MEN	STATE
THEATER	PROGRAMS	PERCENT	PRESIDENT
ACTRESS	GOVERNMENT	CARE	ELEMENTARY
LOVE	CONGRESS	LIFE	HAITI

The William Randolph Hearst Foundation will give $1.25 million to Lincoln Center, Metropolitan Opera Co., New York Philharmonic and Juilliard School. "Our board felt that we had a real opportunity to make a mark on the future of the performing arts with these grants an act every bit as important as our traditional areas of support in health, medical research, education and the social services," Hearst Foundation President Randolph A. Hearst said Monday in announcing the grants. Lincoln Center's share will be $200,000 for its new building, which will house young artists and provide new public facilities. The Metropolitan Opera Co. and New York Philharmonic will receive $400,000 each. The Juilliard School, where music and the performing arts are taught, will get $250,000. The Hearst Foundation, a leading supporter of the Lincoln Center Consolidated Corporate Fund, will make its usual annual $100,000 donation, too.

Semi-supervised Learning







Distribution shift

- Problem (true story, according to Alex Smola)
  - Biotech startup wants to detect prostate cancer
  - Easy to get blood samples from sick patients
  - Hard to get blood samples from healthy ones.

- Solution?
  - Get blood samples from male university students
  - Use them as healthy reference.
  - Classifier gets 100% accuracy.

- What is wrong?

This lecture: Supervised Learning

- A few supervised learning examples.

- Work out a particular supervised learning
  - Naïve Bayes Classifier

Supervised learning: y = f(x)

- Is machine learning just curve fitting?

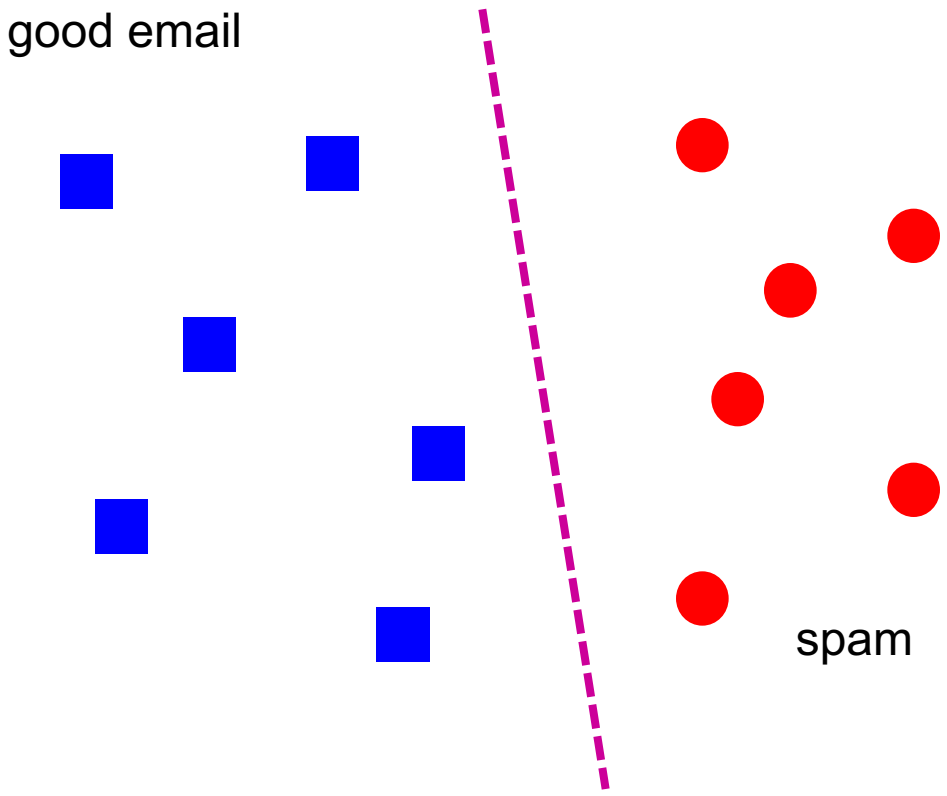
- Binary classification
  Given x find y in {-1, 1}
- Multicategory classification
  Given x find y in {1, … k}
- Regression
  Given x find y in R (or R^d)
- Sequence annotation
  Given sequence x_1 … x_l find y_1 … y_l
- Hierarchical Categorization (Ontology)
  Given x find a point in the hierarchy of y (e.g. a tree)
- Prediction
  Given x_t and y_{t-1} … y_1 find y_t

often with loss

$$l(y, f(x))$$

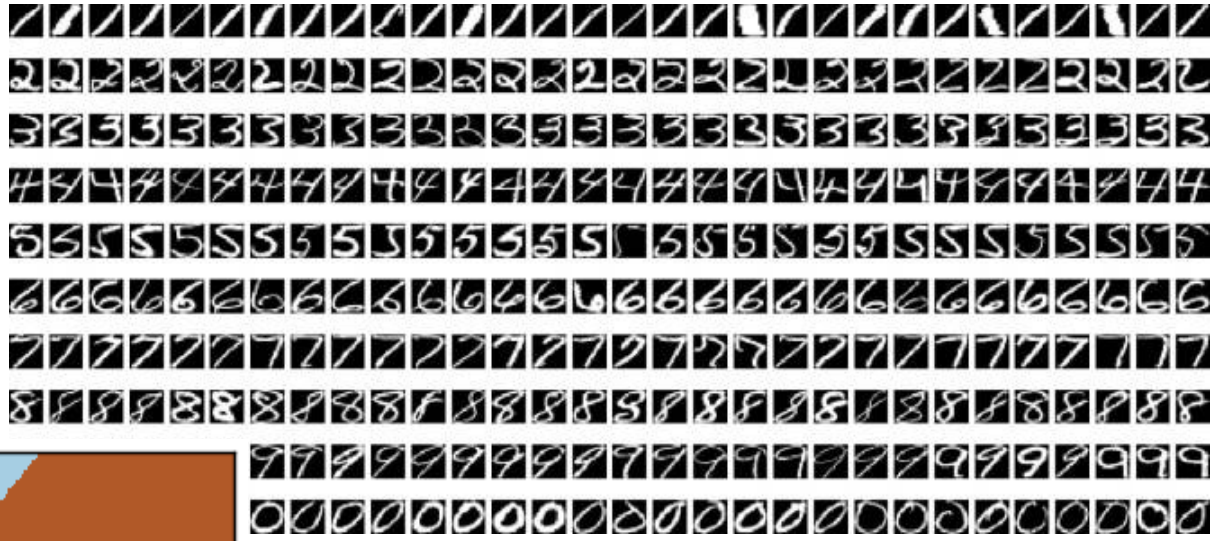
Binary classfication

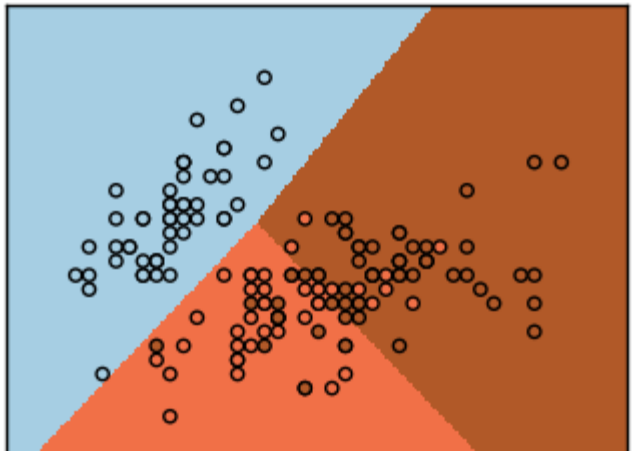
Supervised Learning (Classification)



A new email ----------> Spam or not

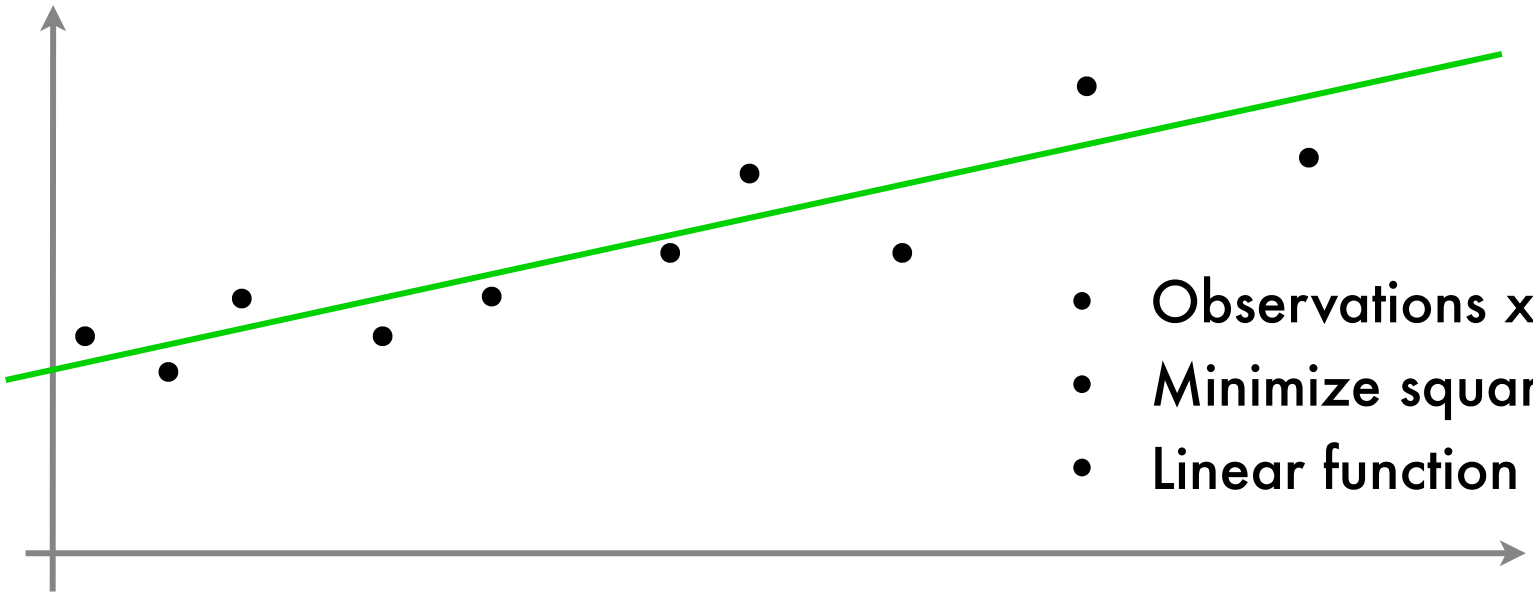
Multiclass classification





map image x to digit y

Linear regression

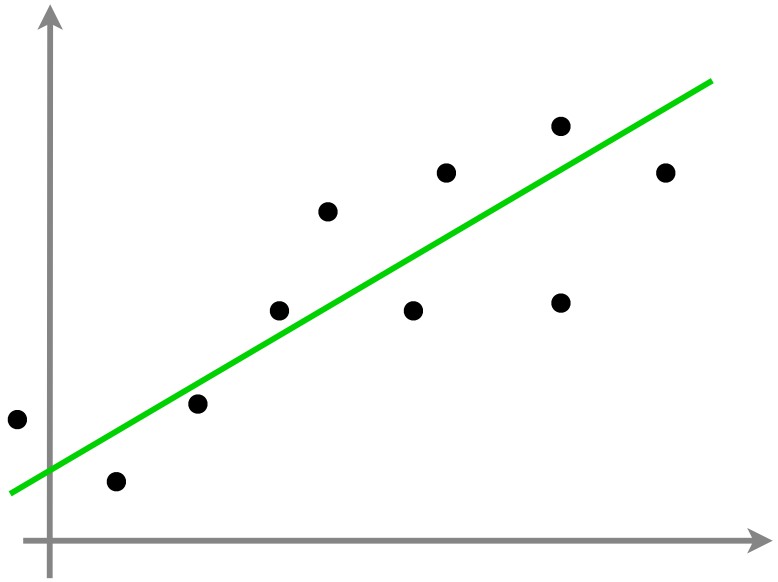


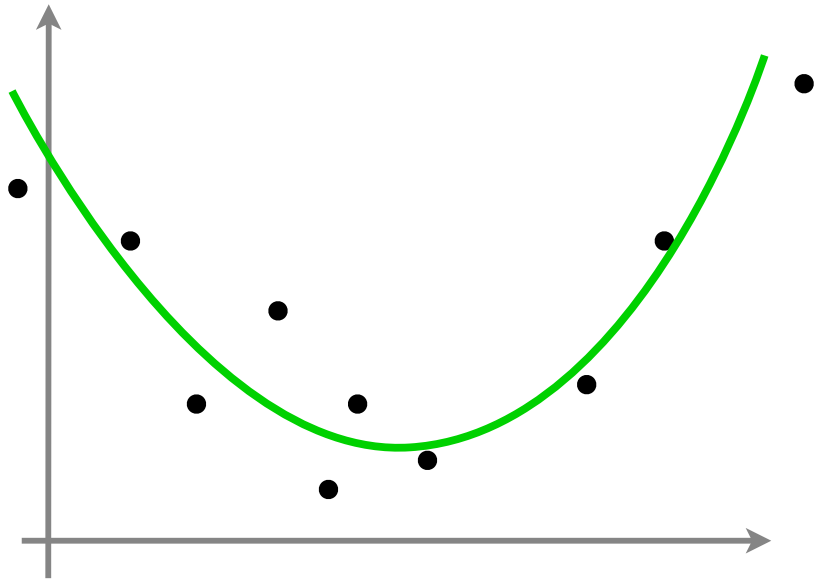
- Observations x, labels y
- Minimize squared distance
- Linear function

$$f(x) = ax + b$$

$$\underset{a,b}{\text{minimize}} \sum_{i=1}^{m} \frac{1}{2}(ax_i + b - y_i)^2$$

Nonlinear regression



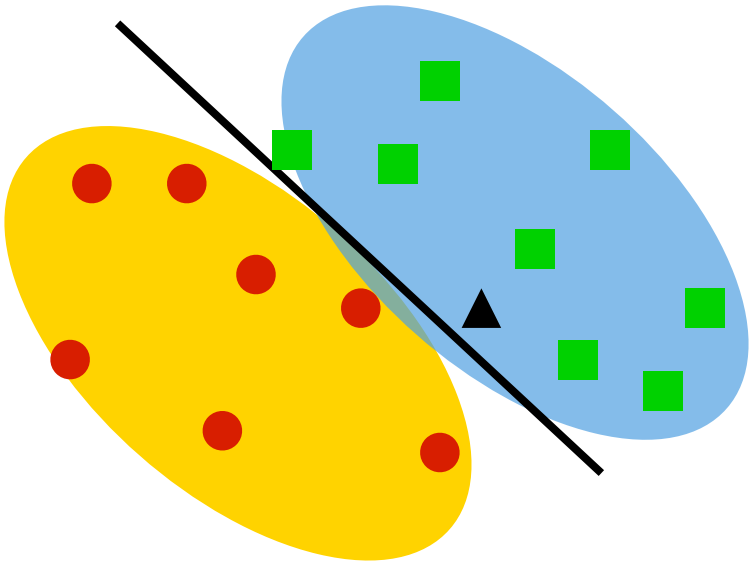


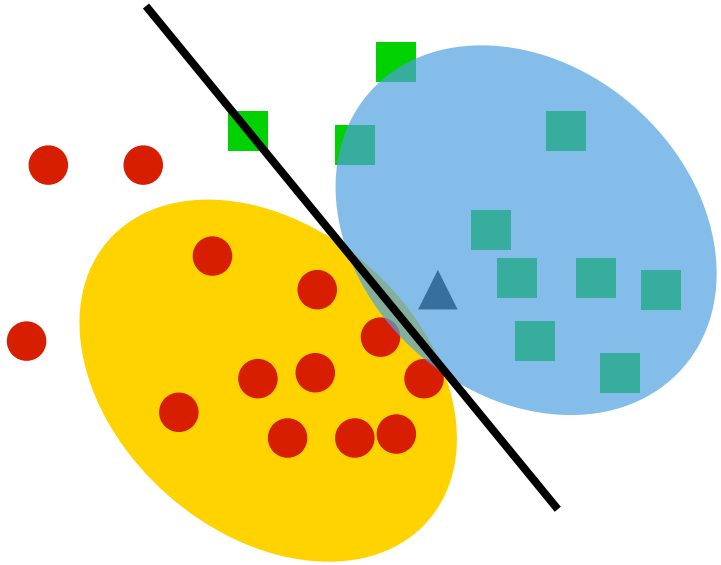
- Linear model $f(x) = \langle w, (1, x) \rangle$
- Quadratic model $f(x) = \langle w, (1, x, x^2) \rangle$
- Cubic model $f(x) = \langle w, (1, x, x^2, x^3) \rangle$
- Nonlinear model $f(x) = \langle w, \phi(x) \rangle$

Other dimensions to categorize supervised learning

- Generative vs. discriminative
- Batch vs. Online

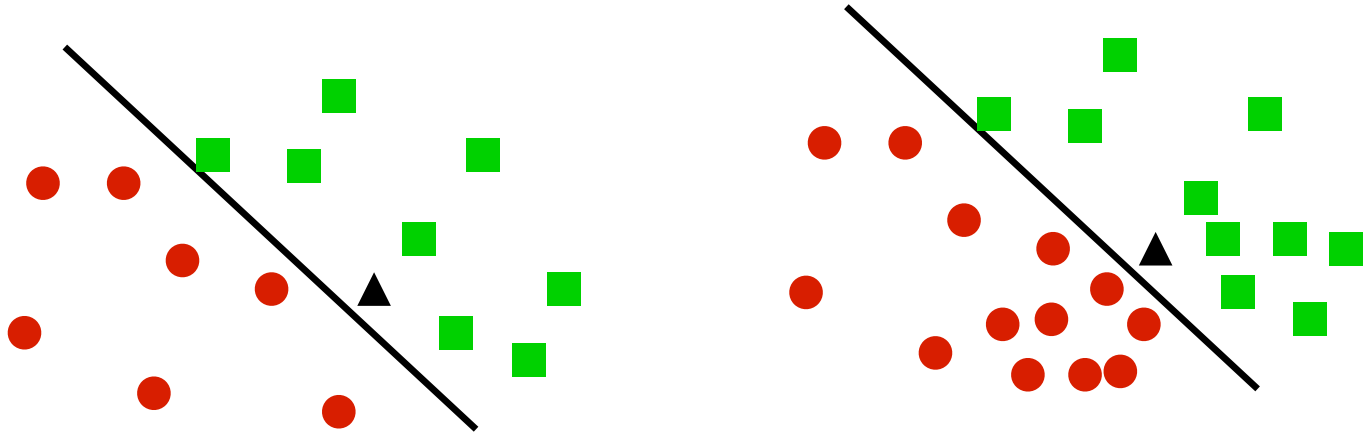
Generative learning





- Model (x,y)
- Infer p(y | x)
- Good for missing variables, diagnostics
- Easy to incorporate prior knowlege

Discriminative learning



- Focus on p(y|x) directly.

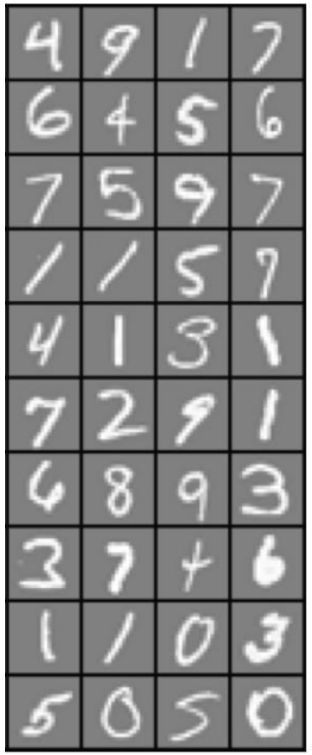
- We never know what the underlying distribution of the data (x,y).

Batch learning

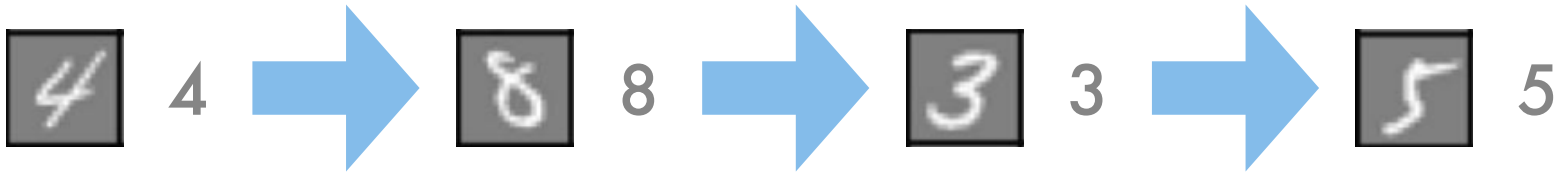
training data

build model

test

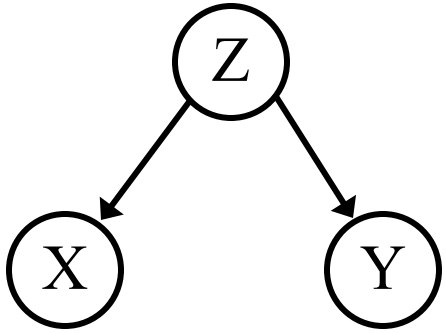


Online learning



Constructing a BN

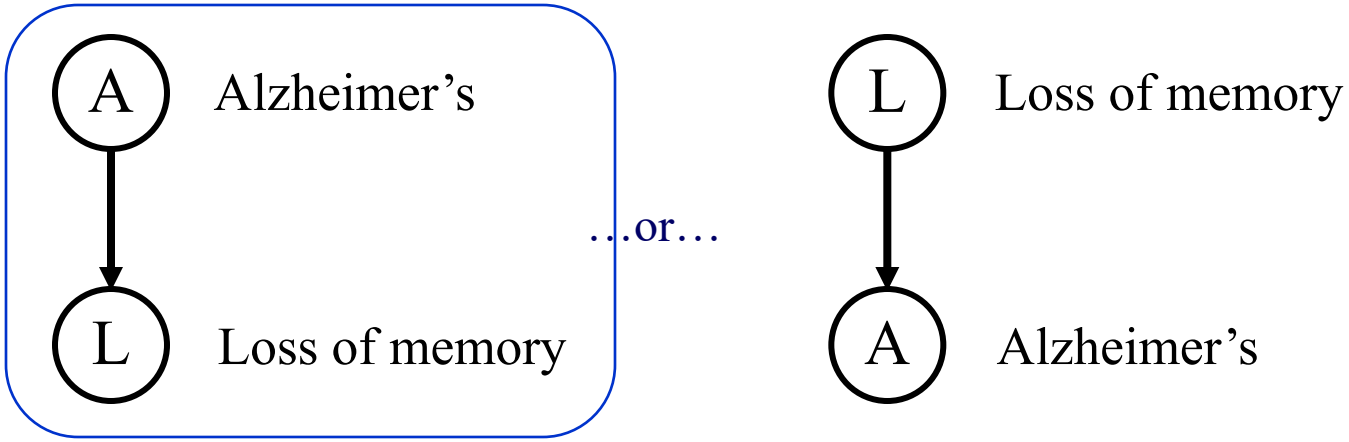
- Constructing a belief network for a given problem is somewhat of an art – there are several choices left to the designer
  - Variables (nodes)
  - Influences (arcs)
  - CPTs



- What is the effect of choosing "wrong" arcs?
  - Inaccurate conditional independences
    - Therefore wrong answers
  - Difficult to construct CPTs
  - May be computationally expensive

Example

- Consider the relationship between Alzheimer's (A) and loss of memory (L)
- We would generally want to know P(A | L)
  - What useful information is typically known? P(L | A) and P(A)
- Which influence diagram makes more sense?



Both are equally valid, since P(A, L) = P(A) P(L|A) = P(L) P(A|L)

Categorization/Classification

- Given:
  - A description of an instance, $x \in X$, where X is the *instance space*.
  - A fixed set of categories:

    $C = \{c_1, c_2, \ldots, c_n\}$
- Determine:
  - The category of x: $c(x) \in C$, where $c(x)$ is a *categorization function* whose domain is X and whose range is C.
    - We want to know how to build categorization functions ("classifiers").

Maximum a posteriori Hypothesis

$$h_{MAP} \equiv \underset{h \in H}{\operatorname{argmax}} P(h \mid D)$$

$$= \underset{h \in H}{\operatorname{argmax}} \frac{P(D \mid h) P(h)}{P(D)}$$

$$= \underset{h \in H}{\operatorname{argmax}} P(D \mid h) P(h)$$

As *P(D)* is constant

Maximum likelihood Hypothesis

If all hypotheses are a priori equally likely, we only need to consider the $P(D|h)$ term:

$$h_{ML} \equiv \underset{h \in H}{\operatorname{argmax}} P(D \mid h)$$

Bayesian Methods

- Learning and classification methods based on probability theory.
- Build a *generative model* that approximates how data is produced
- Uses *prior* probability of each category
- Categorization produces a *posterior* probability distribution over the possible categories given a description of an item

Bayes Classifiers

Task: Classify a new instance D based on a tuple of attribute values $D = \langle x_1, x_2, \cdots x_n \rangle$ into one of the classes $c_j \in C$

$$c_{MAP} = \underset{c_j \in C}{\operatorname{argmax}} P(c_j \mid x_1, x_2 \cdots, x_n)$$

$$= \underset{c_j \in C}{\operatorname{argmax}} \frac{P(x_1, x_2, \cdots, x_n \mid c_j) P(c_j)}{P(x_1, x_2, \cdots, x_n)}$$

$$= \underset{c_j \in C}{\operatorname{argmax}} P(x_1, x_2, \cdots, x_n \mid c_j) P(c_j)$$

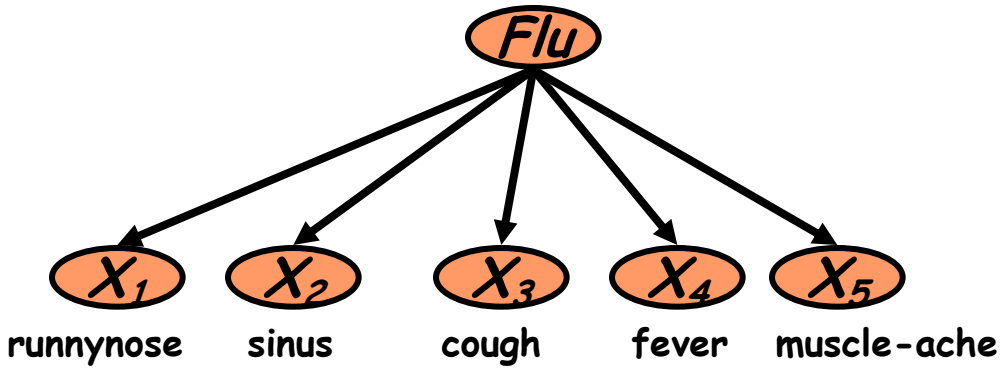
Naïve Bayes Classifier: Assumption

- $P(c_j)$
  - Can be estimated from the frequency of classes in the training examples.
- $P(x_1,x_2,\ldots,x_n|c_j)$
  - O($|X|^n \bullet |C|$) parameters
  - Could only be estimated if a very, very large number of training examples was available.

Naïve Bayes Conditional Independence Assumption:

- Assume that the probability of observing the conjunction of attributes is equal to the product of the individual probabilities $P(x_i|c_j)$.

The Naïve Bayes Classifier

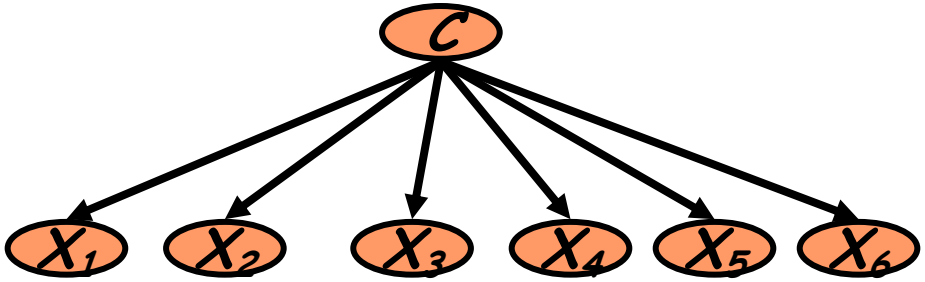


- **Conditional Independence Assumption:** features are independent of each other given the class:

$$P(X_1, \cdots X_5 \mid C) = P(X_1 \mid C) \bullet P(X_2 \mid C) \bullet \cdots \bullet P(X_5 \mid C)$$

- This model is appropriate for binary variables

Learning the Model

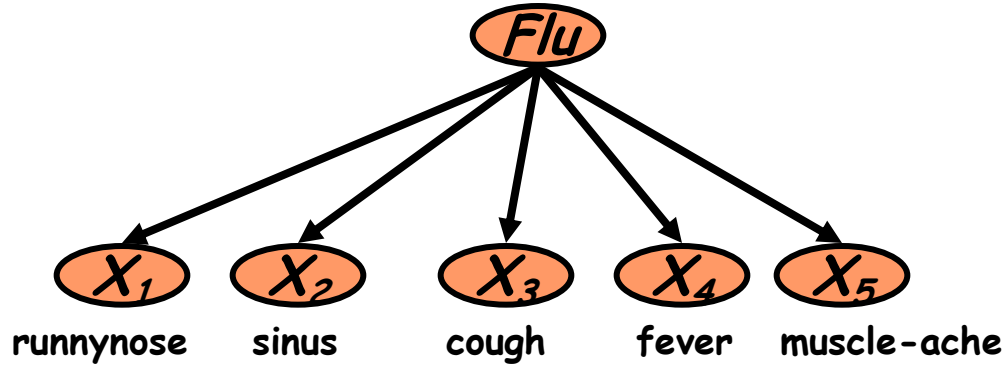


- First attempt: maximum likelihood estimates
  - simply use the frequencies in the data

$$\hat{P}(c_j) = \frac{N(C = c_j)}{N}$$

$$\hat{P}(x_i \mid c_j) = \frac{N(X_i = x_i, C = c_j)}{N(C = c_j)}$$

Problem with Maximum Likelihood



$$P(X_1,\ldots,X_5 \mid C) = P(X_1 \mid C) \bullet P(X_2 \mid C) \bullet \cdots \bullet P(X_5 \mid C)$$

- What if we have seen no training cases where patient had no flu and muscle aches?

$$\hat{P}(X_5 = t \mid C = nf) = \frac{N(X_5 = t, C = nf)}{N(C = nf)} = 0$$

Smoothing to Avoid Overfitting

$$\hat{P}(x_i \mid c_j) = \frac{N(X_i = x_i, C = c_j) + 1}{N(C = c_j) + k}$$

# of values of X_i

- Somewhat more subtle version

overall fraction in data where $X_i = x_{i,k}$

$$\hat{P}(x_{i,k} \mid c_j) = \frac{N(X_i = x_{i,k}, C = c_j) + mp_{i,k}}{N(C = c_j) + m}$$

extent of "smoothing"

Using Naive Bayes Classifiers to Classify Text: Basic method

- Attributes are text positions, values are words.

$$c_{NB} = \underset{c_j \in C}{\operatorname{argmax}} P(c_j) \prod_i P(x_i \mid c_j)$$

$$= \underset{c_j \in C}{\operatorname{argmax}} P(c_j) P(x_1 = \text{"our"} \mid c_j) \ldots P(x_n = \text{"text"} \mid c_j)$$

- Still too many possibilities
- Assume that classification is *independent* of the positions of the words
  - Use same parameters for each position
  - Result is bag of words model (over tokens not types)

Naïve Bayes: Learning

- From training corpus, extract *Vocabulary*
- Calculate required $P(c_j)$ and $P(x_k \mid c_j)$ terms
  - For each c_j in C do
    - $docs_j \leftarrow$ subset of documents for which the target class is c_j
    - $$P(c_j) \leftarrow \frac{|docs_j|}{|\text{total \# documents}|}$$
      - $Text_j \leftarrow$ single document containing all $docs_j$
        - for each word x_k in *Vocabulary*
          - $n_k \leftarrow$ number of occurrences of x_k in $Text_j$

$$P(x_k \mid c_j) \leftarrow \frac{n_k + \alpha}{n + \alpha |Vocabulary|}$$

Naïve Bayes: Classifying

- Return c_{NB}, where

$$c_{NB} = \underset{c_j \in C}{\operatorname{argmax}} P(c_j) \prod_{i \in positions} P(x_i \mid c_j)$$

Naive Bayes: Time Complexity

- Training Time:

  $O(|D|L_d + |C||V|))$

  where L_d is the average length of a document in D.
  - Generally just $O(|D|L_d)$ since usually $|C||V| < |D|L_d$
- Test Time:

  $O(|C| L_t)$

  where L_t is the average length of a test document.
- Very efficient overall, linearly proportional to the time needed to just read in all the data.

Underflow Prevention

- Multiplying lots of probabilities, which are between 0 and 1 by definition, can result in floating-point underflow.
- Since $\log(xy) = \log(x) + \log(y)$, it is better to perform all computations by summing logs of probabilities rather than multiplying probabilities.
- Class with highest final un-normalized log probability score is still the most probable.

$$c_{NB} = \underset{c_j \in C}{\operatorname{argmax}} \log P(c_j) + \sum_{i \in positions} \log P(x_i \mid c_j)$$

Violation of NB Assumptions

- Conditional independence
- “Positional independence”

Naïve Bayes Posterior Probabilities

- Classification results of naïve Bayes (the class with maximum a posteriori probability) are usually fairly accurate.
- However, due to the inadequacy of the conditional independence assumption, the actual posterior-probability numerical estimates are not accurate.
  - Output probabilities are generally very close to 0 or 1.

When does Naive Bayes work?

Sometimes NB performs well even if the Conditional Independence assumptions are badly violated.

Classification is about predicting the correct class label and NOT about accurately estimating probabilities.

Assume two classes c_1 and c_2. A new case A arrives.

NB will classify A to c_1 if:

$P(A, c_1) > P(A, c_2)$

	$P(A,c_1)$	$P(A,c_2)$	Class of A
Actual Probability	0.1	0.01	c_1
Estimated Probability by NB	0.08	0.07	c_1

Besides the big error in estimating the probabilities the classification is still correct.

Correct estimation ⇒ accurate prediction

but NOT

~~accurate prediction ⇒ Correct estimation~~

Naive Bayes is Not So Naive

- Naïve Bayes: First and Second place in KDD-CUP 97 competition, among 16 (then) state of the art algorithms

  Goal: Financial services industry direct mail response prediction model: Predict if the recipient of mail will actually respond to the advertisement – 750,000 records.

- Robust to Irrelevant Features

  Irrelevant Features cancel each other without affecting results

  Instead Decision Trees can heavily suffer from this.

- Very good in Domains with many equally important features

  Decision Trees suffer from *fragmentation* in such cases – especially if little data

- A good dependable baseline for text classification (but not the best)!

- Optimal if the Independence Assumptions hold: If assumed independence is correct, then it is the Bayes Optimal Classifier for problem

- Very Fast: Learning with one pass over the data; testing linear in the number of attributes, and document collection size

- Low Storage requirements

- Warning: There are other much advanced classifiers (Machine Learning Course)