Costa Rica Big Data School: Introduction to Machine Learning

Weijia Xu
Research Scientist, Group Manager
Data Mining & Statistics
Texas Advanced Computing Center
University of Texas at Austin
Dec. 2018

TACC

What is Machine Learning?

- Definition from T. Mitchell (1997). *Machine Learning book*:

> "A computer program is said to learn from experience $\mathcal{E}$ with respect to some class of tasks $\mathcal{T}$ and performance measure $\mathcal{P}$, if its performance at the tasks improves with the experiences."
>
> --- (Mitchell 1997)

  - **Learn from past experiences**
  - **Improves with experiences**

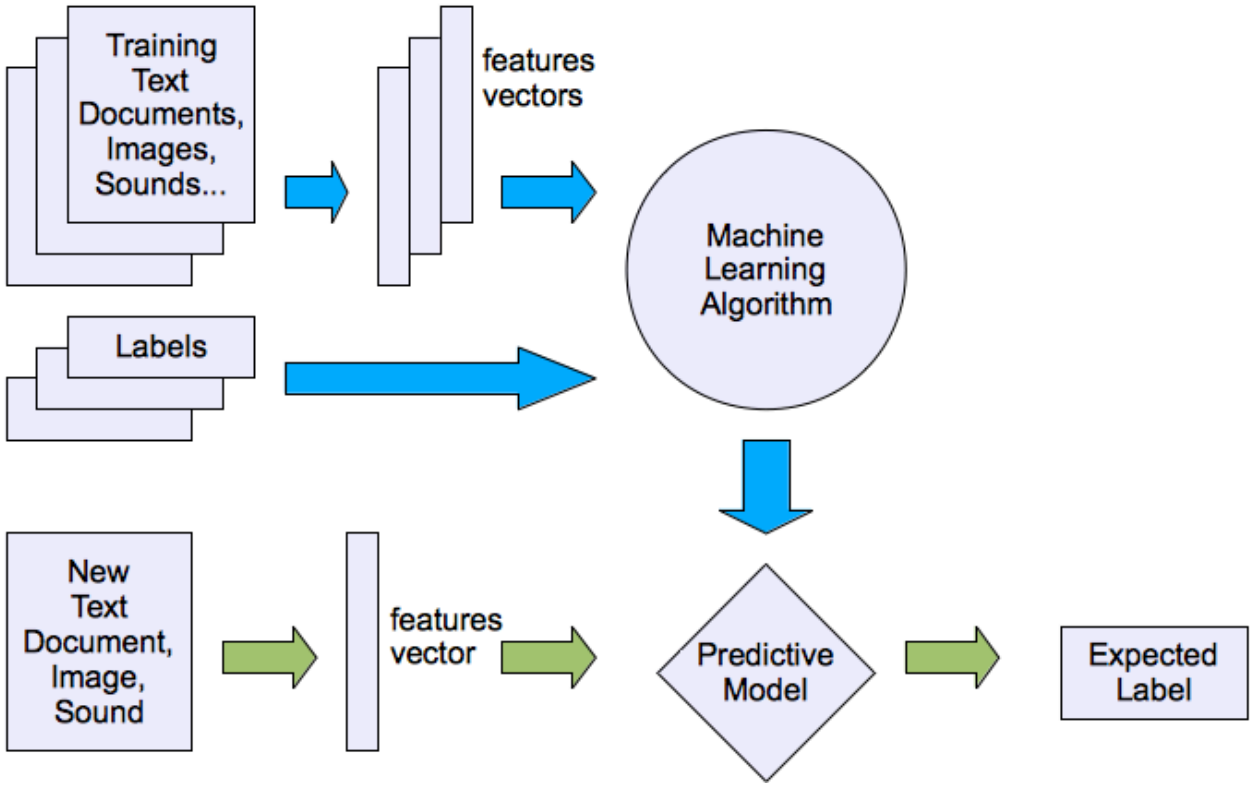
Goal of Machine Learning

- Learning general models from existing data
  - Data is cheap and abundant
  - Noises, hidden variables.
- Model is a good and useful approximation to the data.
- Model is “acquired” in the process different from existing theories/mathematical models.
  - Different from descriptive analysis, such as generating statistical summary.
  - Existing theories and mathematical models are basis and important to drive the machine learning.
  - Using available data to improve the model.

Training and Testing

- Training
  - The process of making the system to learn a model.
  - Data to be “observed” by the learning system
- Testing
  - Process to evaluate the performance of the model
  - Data are not observed by the learning system
- 80/20 split
  - 80% available data are randomly selected for training
  - The rest 20% are used for testing.
- Prediction
  - Apply model to data not in either training or testing data set.
  - Assume the input data and its prediction are from the same process producing the training data.

Supervised Learning



What are we learning?

Supervised Learning

- A model “explains” observed data in training data set and their labels/values.
  - Classification
    - Discrete labels
  - Regression
    - Real continuous values.
- Common learning techniques
  - Linear classifier.
  - Instance based functions (Non-parametric).
  - Probabilistic functions (Parametric),
  - Symbolic functions (Non-metric)
  - Aggregation/Ensemble methods.

Common Methods

- K Nearest Neighbor
- Regression Methods
- Support Vector machine
- Naïve Bayes Classifier
- Decision Tree
- Random Forest

K Nearest Neighbors

Nearest Neighbor Classification

- The basic idea:

  If two data objects are similar (close), they are likely are from the same class.

- General workflow

  Start with a set of data object with class labels

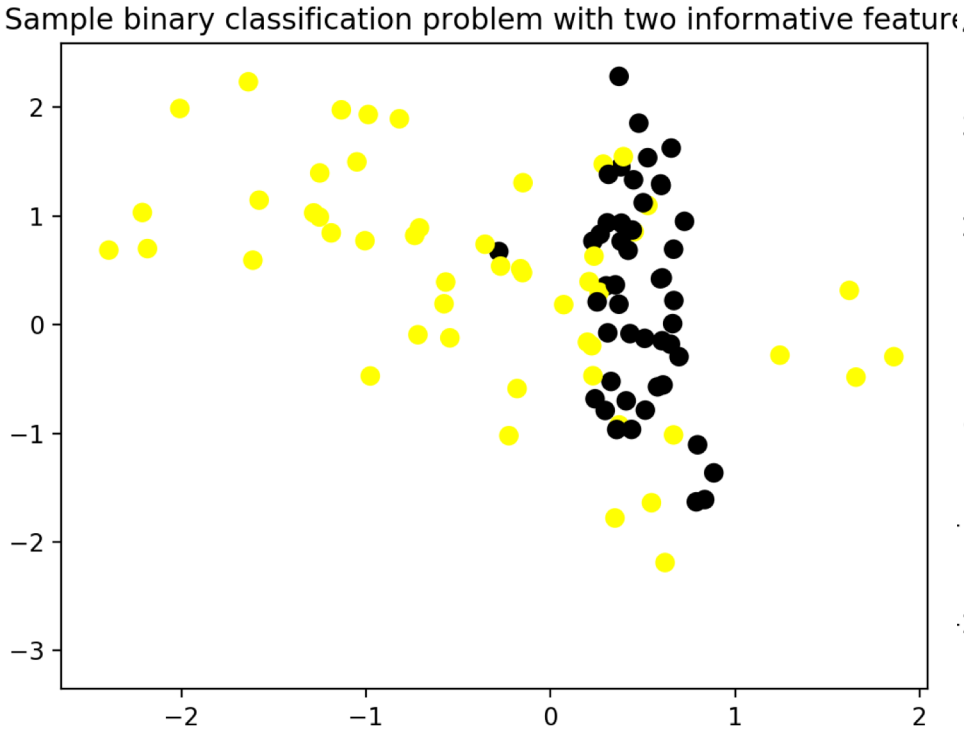
  Given a distance (similarity)measure of comparing data

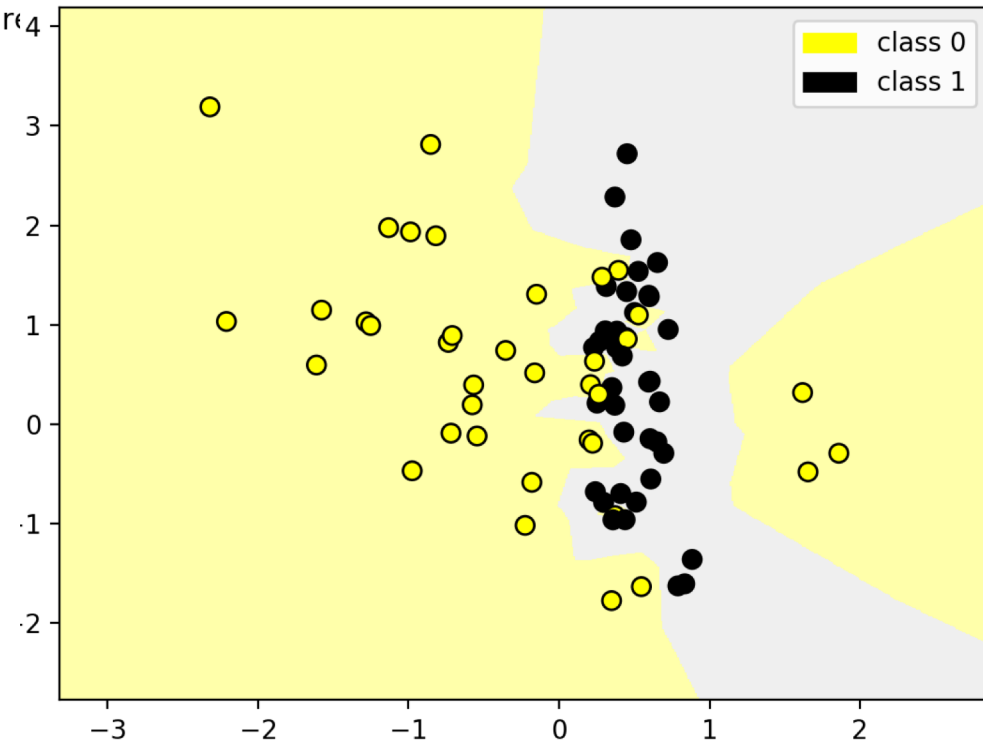
  Retrieve (k) nearest neighbor for the unknown data

  Assign class label based on the retrieved record.

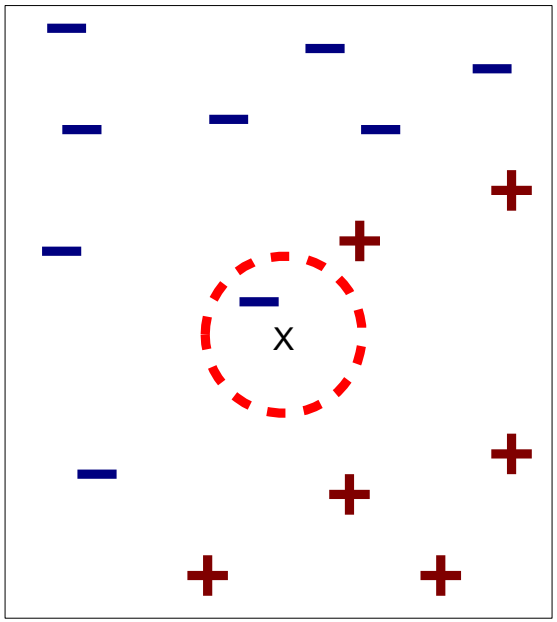
Nearest Neighbor Classification

- A simple binary classification case

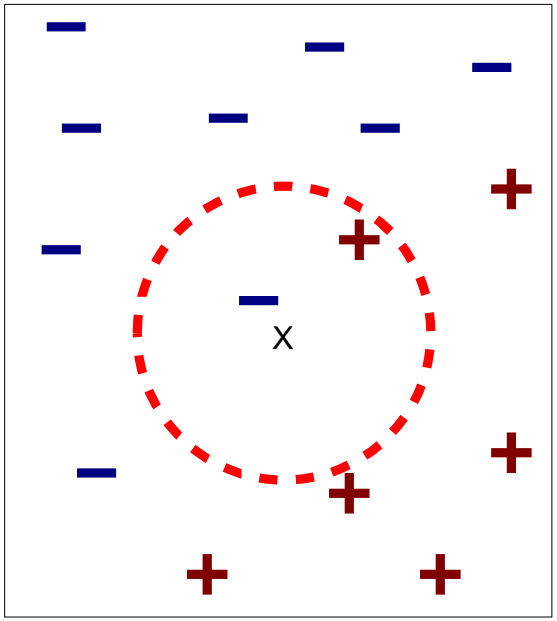




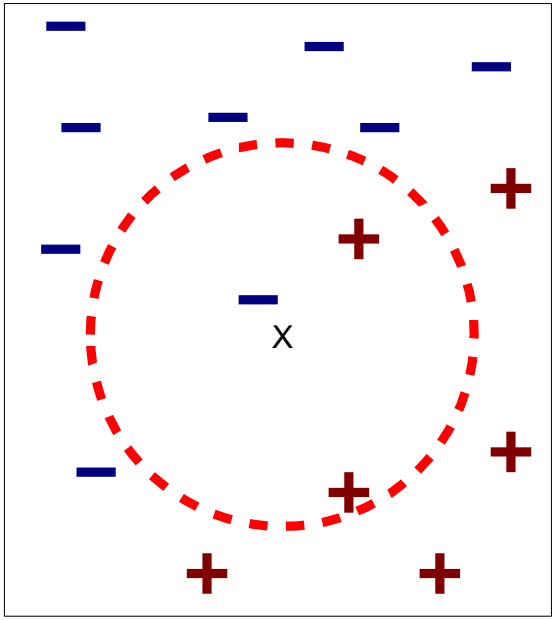
K-NN Classification



(a) 1-nearest neighbor



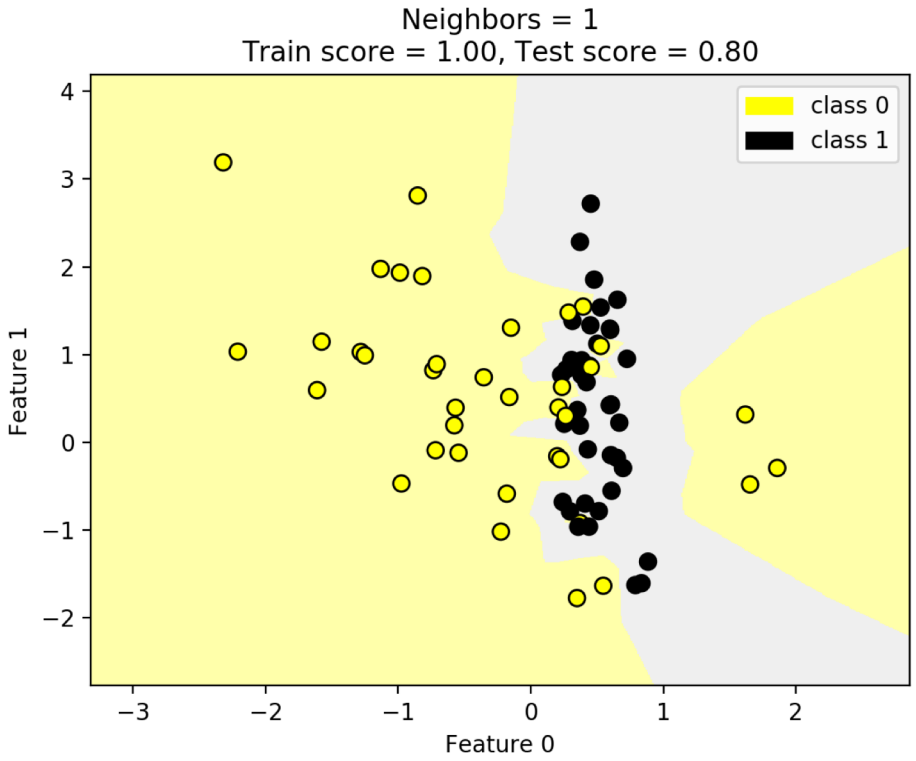
(b) 2-nearest neighbor

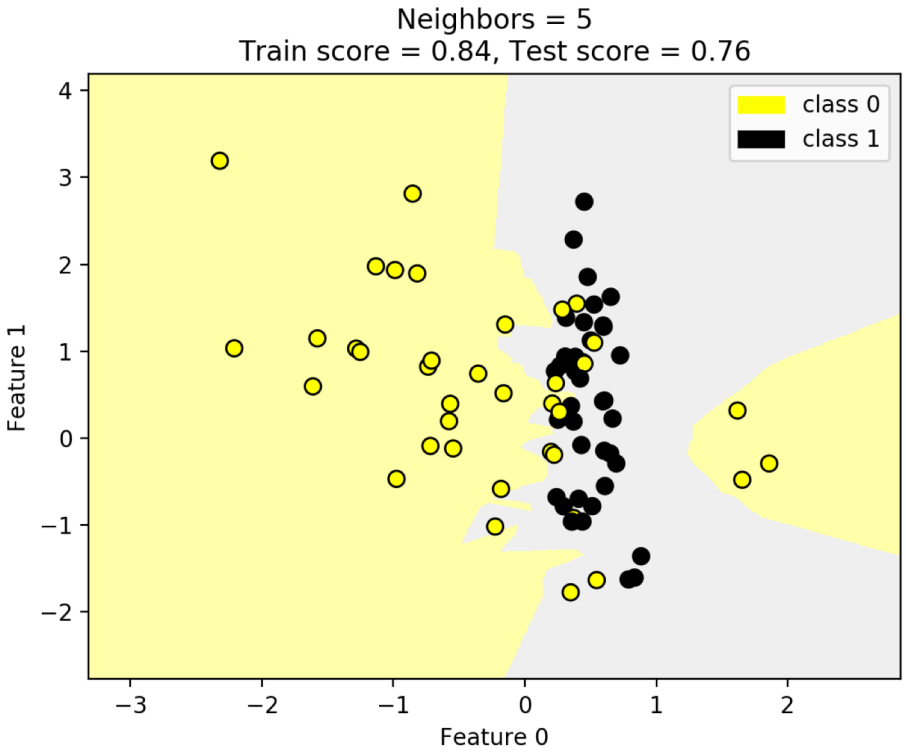


(c) 3-nearest neighbor

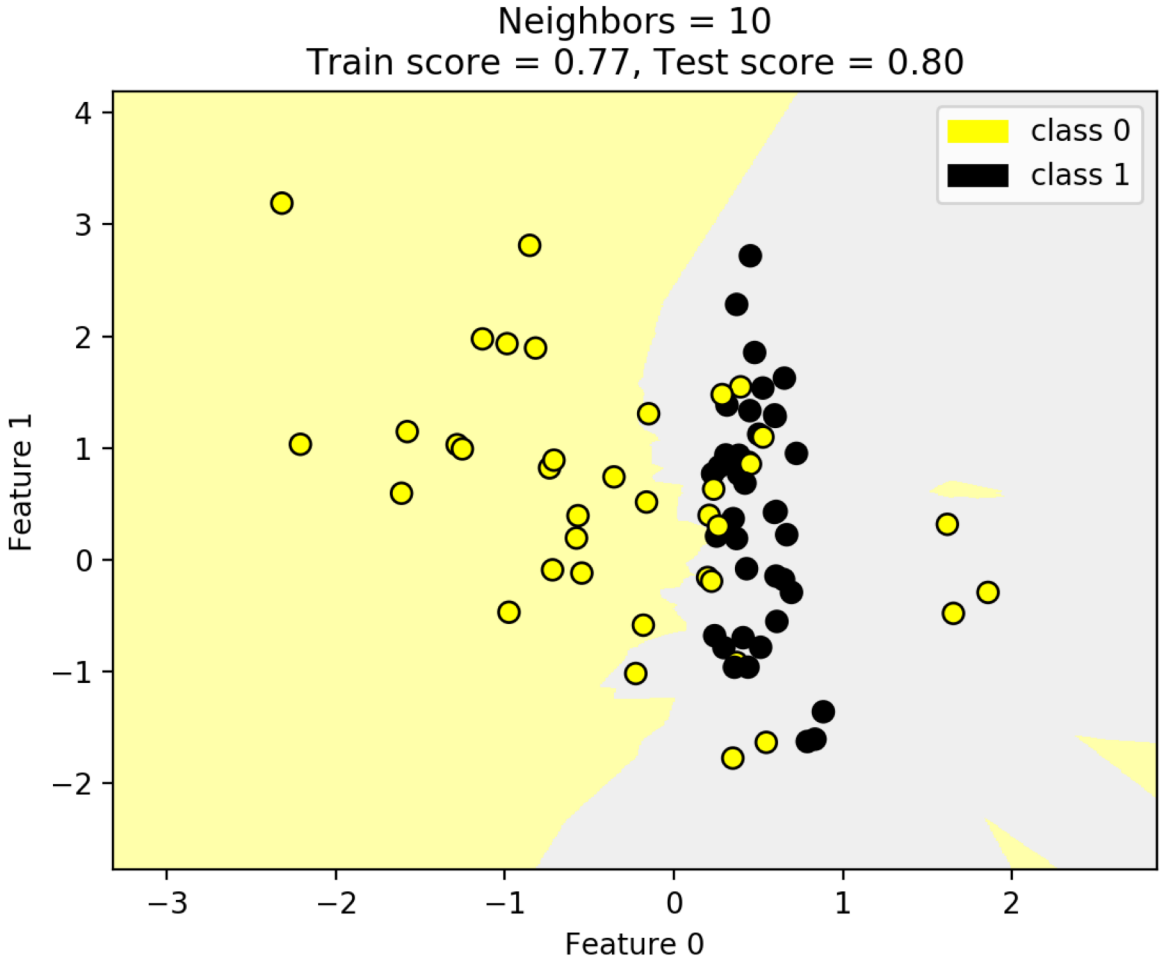
Choice of K

- What changes?





Choice of K


Neighbors = 10
Train score = 0.77, Test score = 0.80
class 0
class 1
Feature 1
Feature 0
4
3
2
1
0
−1
−2
−3
−2
−1
0
1
2

Concerns and Limitations

- Distance/similarity measures might be misleading

1 1 1 1 1 1 1 1 1 1 1 0		1 0 0 0 0 0 0 0 0 0 0 0	
0 1 1 1 1 1 1 1 1 1 1 1	vs	0 0 0 0 0 0 0 0 0 0 0 1	
d = 1.4142		d = 1.4142	

- Less effective for high dimensional data

- Values of distance measures might be dominated by one dimension

Regression

Regression

- A statistical measure to determine (the strength) of the relationship between one dependent variable and one or a series of changing variables
- Help us understand the relationship between variables
- Prediction values for unknown target

A Typical Regression Model

- The independent variables X
- The dependent variables Y
  outcome, target, or criterion variable
- The unknown parameters $\boldsymbol{\theta}$
- Predict/estimate Y with F(X, $\boldsymbol{\theta}$)

Common Goal of the Regression

- Learn the parameters to minimize the cost/prediction errors

- Ordinary least squares (OLS): to minimize

$$\sum (ax^{(i)} - y^{(i)})^2$$

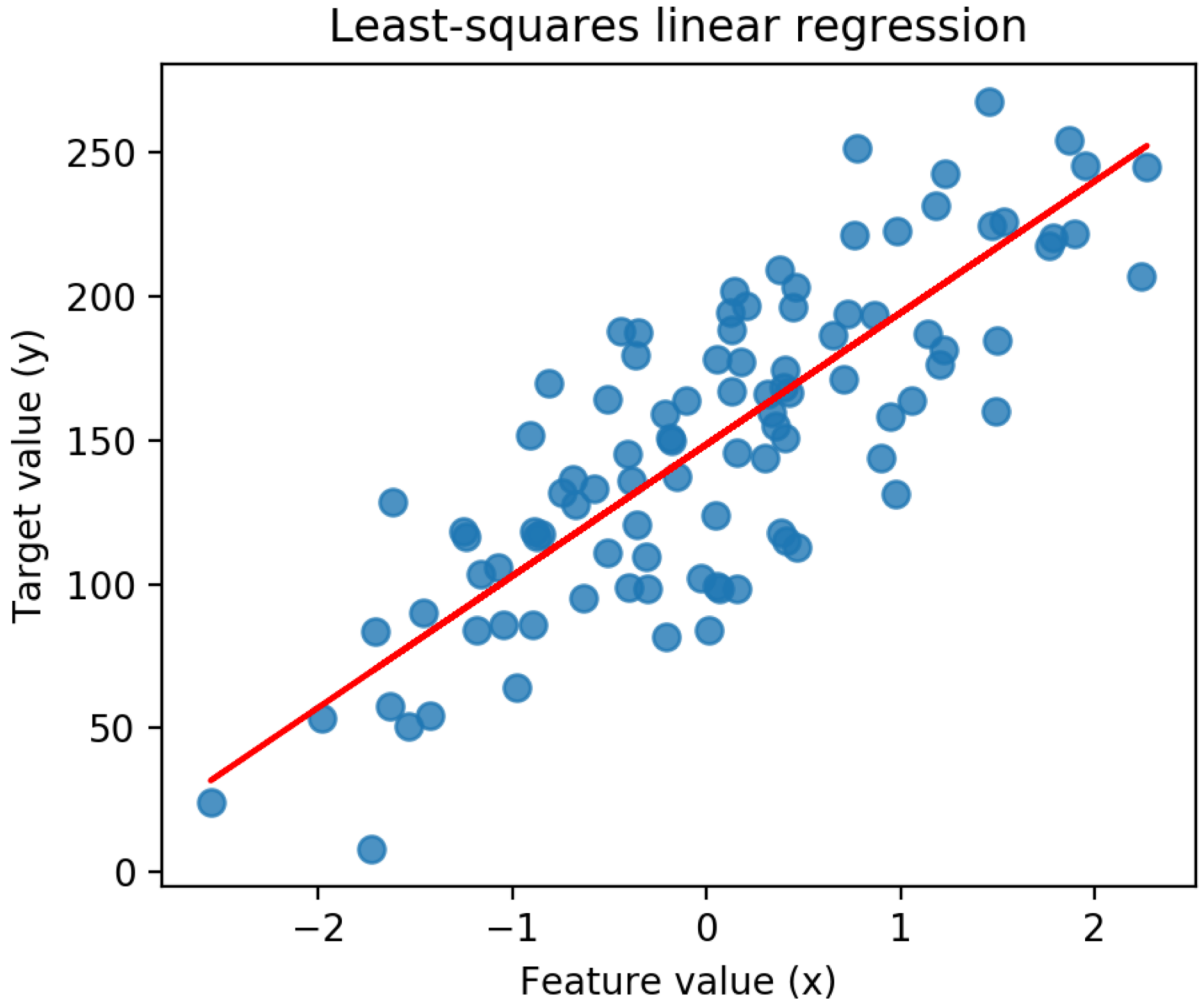
- Least absolute deviations: to minimize

$$\sum |ax^{(i)} - y^{(i)}|$$

Simple Linear Regression

- Only one independent variable (y) and one dependent variable (x)
- The outcome variable is related to a single predictor
- Assuming label and feature are connected through a function e.g. $y = a x + b$

Least Square Solution

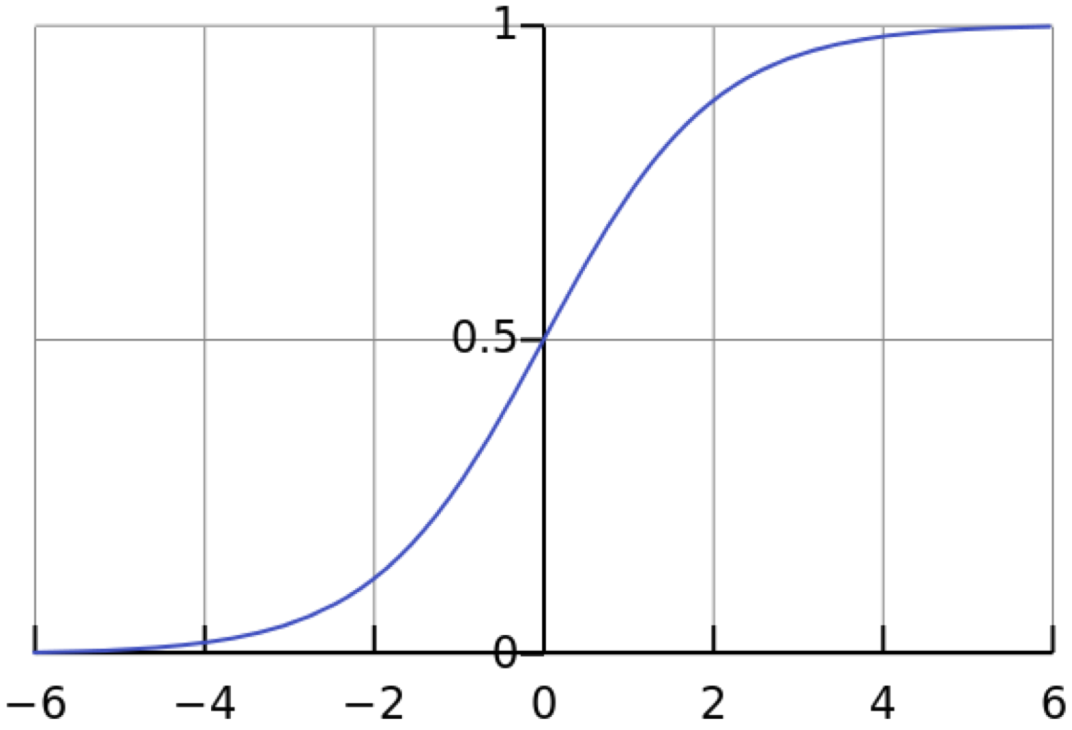


(Binary) Logistic Regression

- (Binary) dependent variable
- ”y” can only take two values (usually 0 and 1)
  Positive/Negative:
  Pass/fail
  Heathy/sick
- One or more independent variables x
- Predict the probability of a binary response

The Standard Logistic Function

$$\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}}$$



Logistic Regression

If z is a linear function of a single independent variable x

$$z = \theta_0 + \theta_1 x_1 + \theta_2 x_2$$

$$\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}}$$

$$F(x) = \sigma(z) = \frac{1}{1 + e^{-\theta_0 - \theta_1 x_1 - \theta_2 x_2}}$$

Support Vector Machine

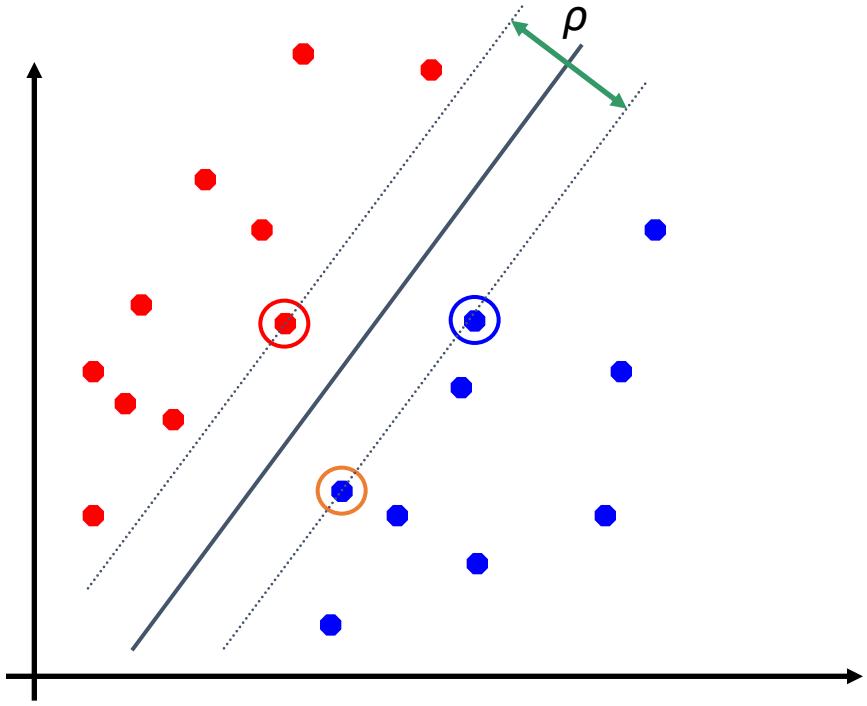
- A discriminative classifier
- Given a few sets of labeled training data
  supervised learning
- Generate an optimal hyperplane
-
- Categorizes new examples.

Binary Classification with Linear Separator

Red and blue dots are representations of objects from two classes in the training data

The line is a linear separator for the two classes

The closets objects to the hyperplane is the support vectors



Classification with SVMs

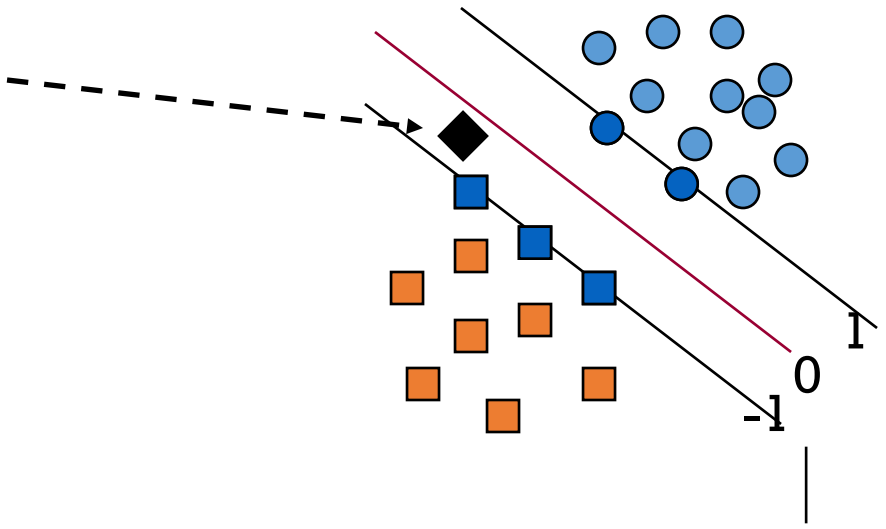
- Given a new point **x**, we can score its projection onto the hyperplane normal:
  i.e., compute score: $\mathbf{w}^T\mathbf{x} + b$
  Decide class based on whether < or > 0

- Can set confidence threshold *t*.

Score > *t*: yes

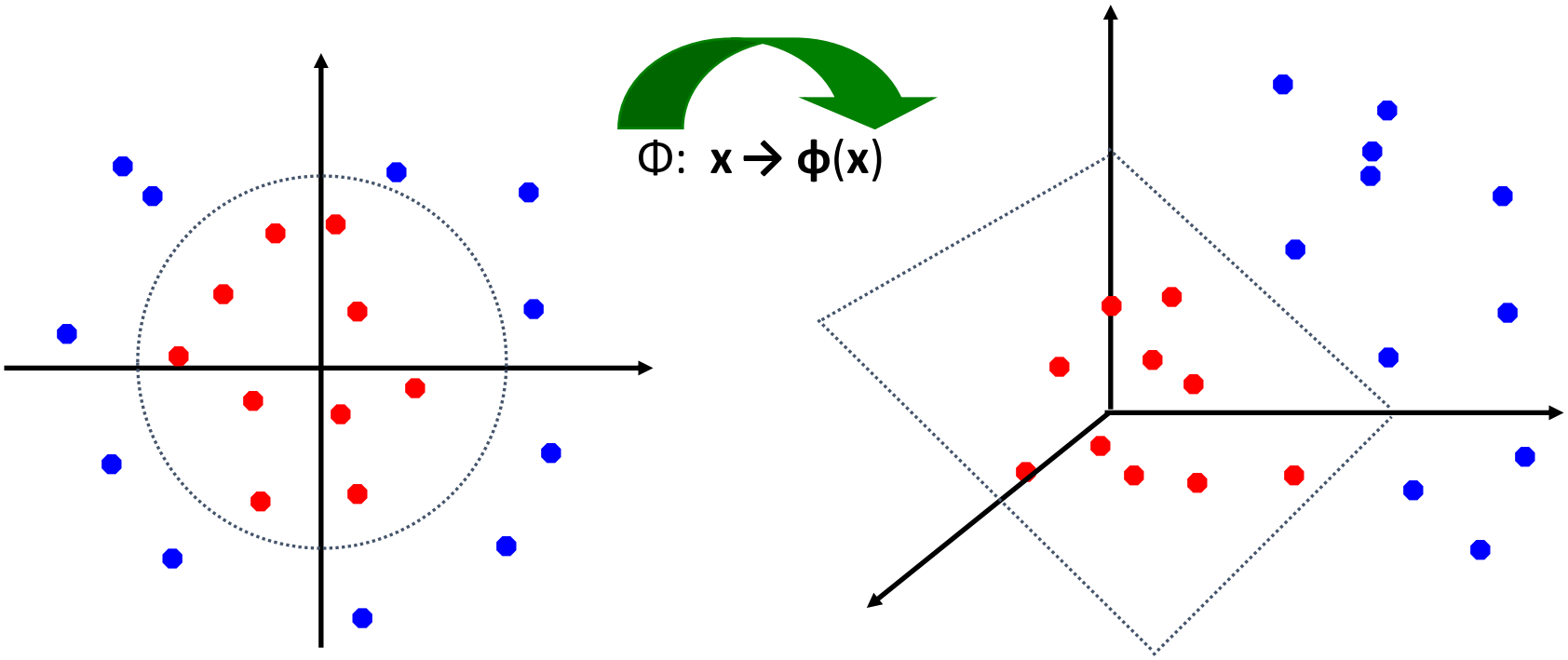
Score < -*t*: no

Else: don't know



Non-linear SVM

Kernels functions: Mapping the original feature space to some higher-dimensional feature space where the training set is separable



Naïve Bayes Classifier

Naïve Bayes Classifier

- The Basic idea
  - Treat the data and its label as some statistic process
  - From historical data
    - → estimate the parameters
  - From test data
    - → compute the probability to be in certain class.

- General Statistic Inference

From joint probability

$$P(AB|I) = P(A|BI)P(B|I)$$

$$P(BA|I) = P(B|AI)P(A|I)$$

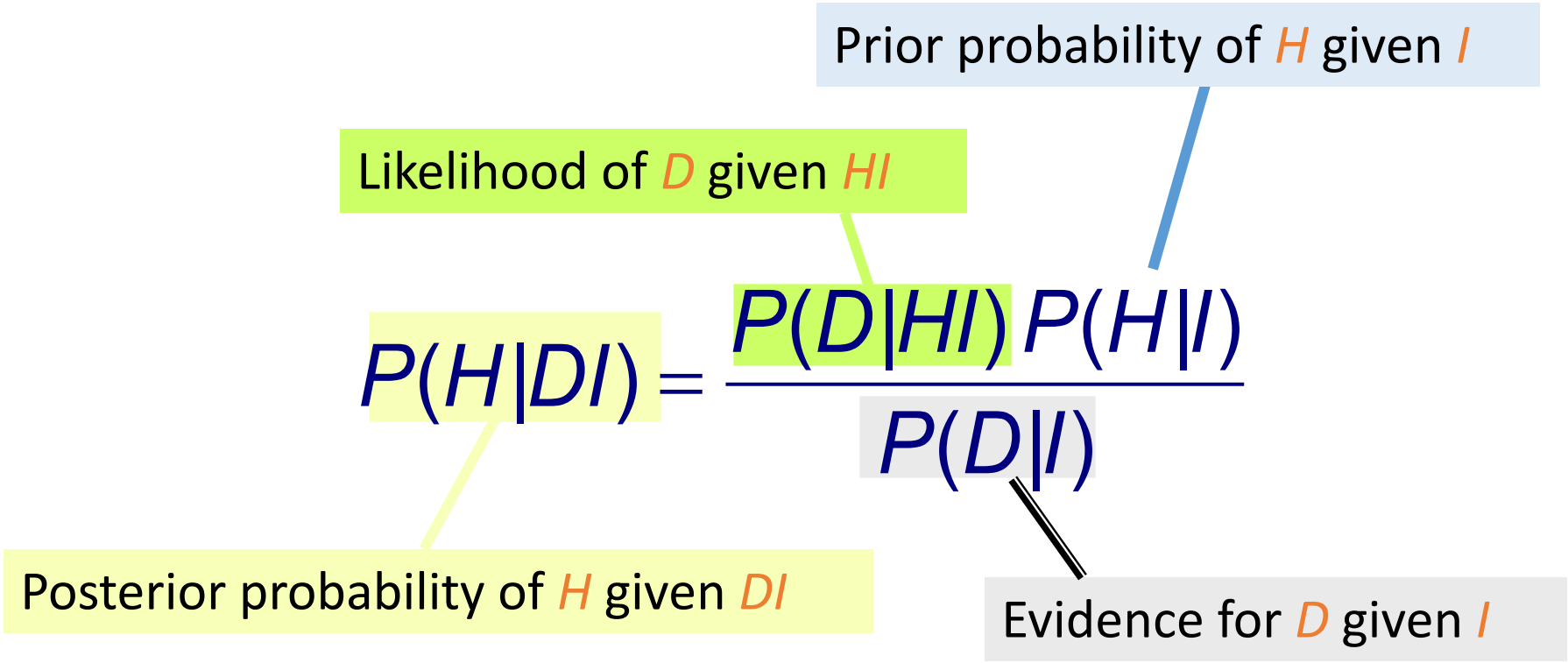
and $AB = BA$ we get Bayes Theorem

$$P(B|AI) = \frac{P(A|BI)\,P(B|I)}{P(A|I)}$$

- Bayes Theorem derives from the axioms of probability calculus and therefore exists in both Frequentist and Bayesian statistics
- In Bayesian statistics it plays a central role

Bayes Theorem for Inference

Let's use H (Hypothesis) and D (Data) instead of A, B



Bayes Theorem: update hypothesis based on the Data

Bayesian Classifiers

Approach:

compute the posterior probability $P(C \mid A_1, A_2, \ldots, A_n)$ for all values of C using the Bayes theorem

$$P(C \mid A_1 A_2 \ldots A_n) = \frac{P(A_1 A_2 \ldots A_n \mid C) P(C)}{P(A_1 A_2 \ldots A_n)}$$

Choose value of C that maximizes

$P(C \mid A_1, A_2, \ldots, A_n)$

Equivalent to choosing value of C that maximizes

$P(A_1, A_2, \ldots, A_n \mid C)\, P(C)$

How to estimate $P(A_1, A_2, \ldots, A_n \mid C)$?

Naïve Bayes Classifier

Assume independence among attributes A_i when class is given:

$$P(A_1, A_2, \ldots, A_n \mid C) = P(A_1 \mid C_j)\, P(A_2 \mid C_j) \ldots P(A_n \mid C_j)$$

Can estimate $P(A_i \mid C_j)$ for all A_i and C_j.

New point is classified to C_j if $P(C_j) \prod P(A_i \mid C_j)$ is maximal.

How to Estimate Probabilities from Data?

Tid	Refund	Marital Status	Taxable Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Class probability:

$P(C) = N_c/N$

e.g., P(No) = 7/10,

P(Yes) = 3/10

Attribute probability

For discrete attributes:

$P(A_i \mid C_k) = |A_{ik}|/ N_c$

k

where $|A_{ik}|$ is number of instances having attribute A_i and belongs to class C_k

Examples:

P(Status=Married|No) = 4/7

P(Refund=Yes|Yes)=0

Estimation using Probability Density function

Tid	Refund	Marital Status	Taxable Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Normal distribution:

$$P(A_i \mid c_j) = \frac{1}{\sqrt{2\pi\sigma_{ij}^2}} e^{-\frac{(A_i-\mu_{ij})^2}{2\sigma_{ij}^2}}$$

One for each (A_i, c_i) pair

For (Income, Class=No):

If Class=No

sample mean = 110

sample variance = 2975

$$P(Income = 120 \mid No) = \frac{1}{\sqrt{2\pi}(54.54)} e^{-\frac{(120-110)^2}{2(2975)}} = 0.0072$$

Example of Naïve Bayes Classifier

Given a Test Record:

$$X = (\text{Refund} = \text{No}, \text{Married}, \text{Income} = 120\text{K})$$

naive Bayes Classifier:

P(Refund=Yes|No) = 3/7
P(Refund=No|No) = 4/7
P(Refund=Yes|Yes) = 0
P(Refund=No|Yes) = 1
P(Marital Status=Single|No) = 2/7
P(Marital Status=Divorced|No)=1/7
P(Marital Status=Married|No) = 4/7
P(Marital Status=Single|Yes) = 2/7
P(Marital Status=Divorced|Yes)=1/7
P(Marital Status=Married|Yes) = 0

For taxable income:
If class=No: sample mean=110
sample variance=2975
If class=Yes: sample mean=90
sample variance=25

- P(X|Class=No) = P(Refund=No|Class=No)
  × P(Married| Class=No)
  × P(Income=120K| Class=No)
  = 4/7 × 4/7 × 0.0072 = 0.0024

- P(X|Class=Yes) = P(Refund=No| Class=Yes)
  × P(Married| Class=Yes)
  × P(Income=120K| Class=Yes)
  = 1 × 0 × 1.2 × 10^{-9} = 0

Since P(X|No)P(No) > P(X|Yes)P(Yes)

Therefore P(No|X) > P(Yes|X)

=> Class = No

Naïve Bayes (Summary)

Robust to isolated noise points

Handle missing values by ignoring the instance during probability estimate calculations

Robust to irrelevant attributes

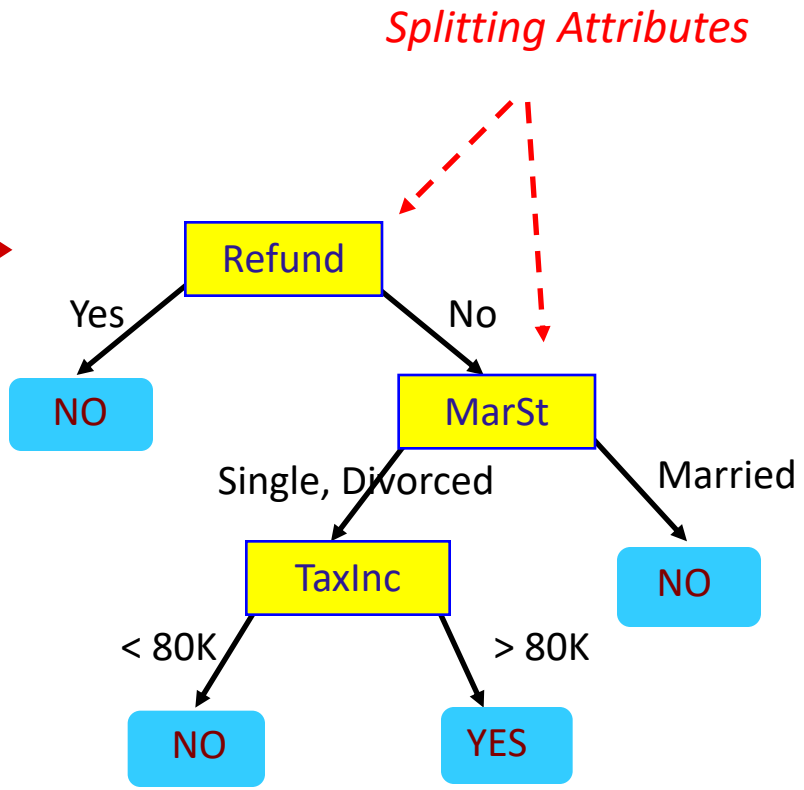
Independence assumption may not hold for some attributes

Decision Tree Classifier

Decision Tree

Tid	Refund	Marital Status	Taxable Income	Cheat
	categorical	categorical	continuous	class
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Training Data

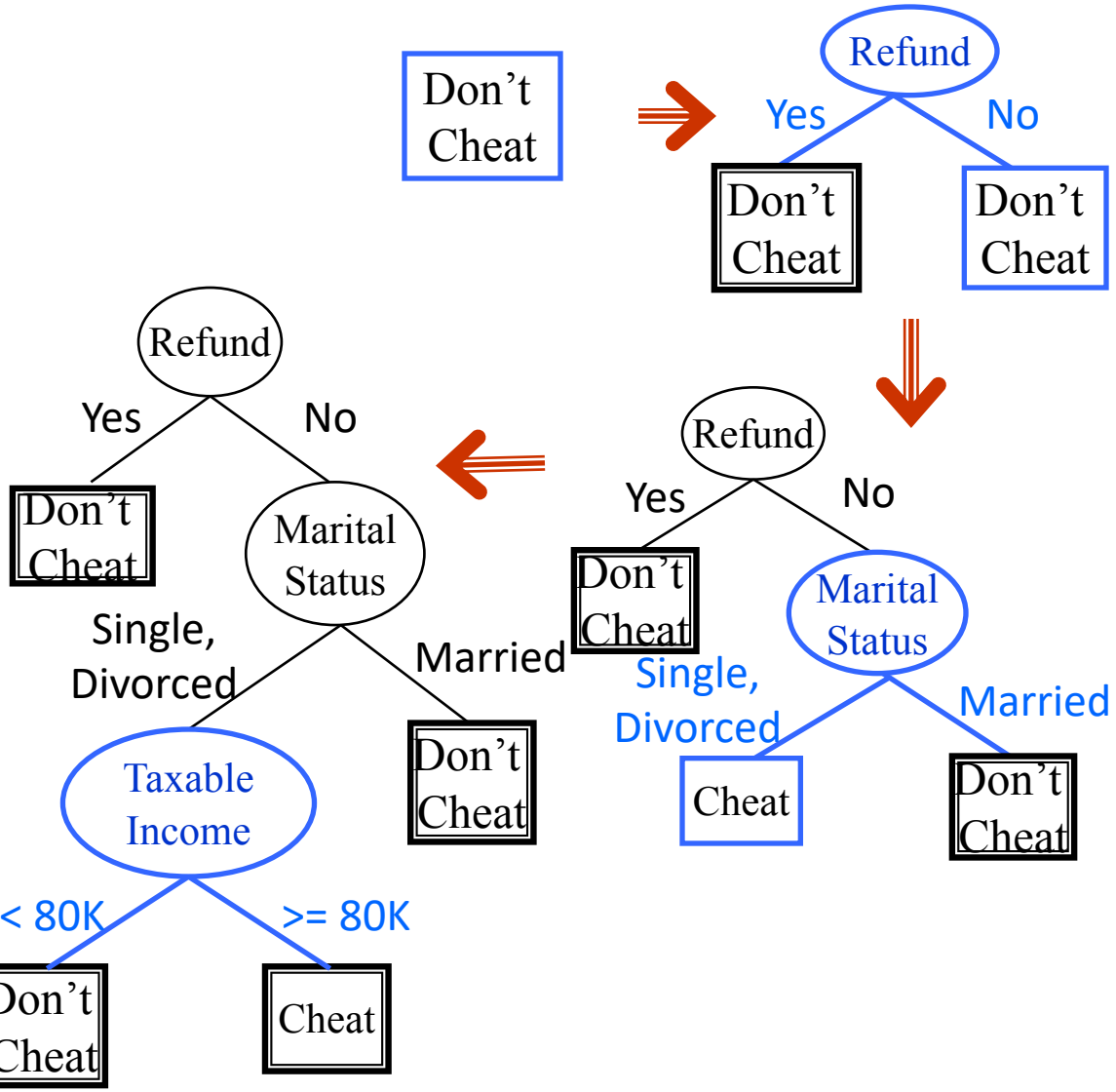


Model: Decision Tree

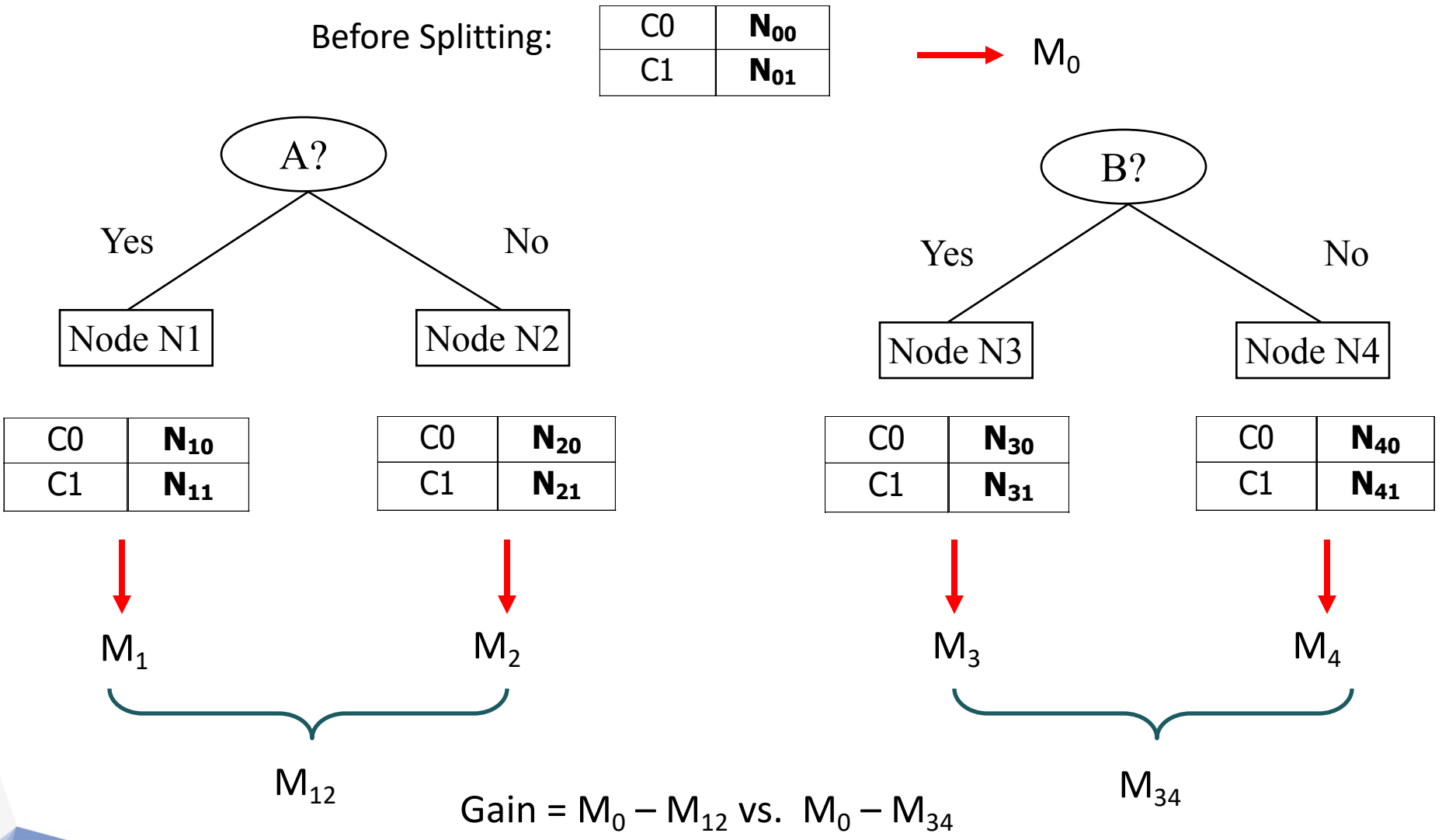
Hunt's Algorithm

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Class 1: Don't cheat (default class)
Class 2: Cheat



How to Find the Best Split



Decision Tree Based Classification

- Advantages:
  - Inexpensive to construct
  - Extremely fast at classifying unknown records
  - Easy to interpret for small-sized trees
  - Accuracy is comparable to other classification techniques for many simple data sets

- Less effective when
  - High background noise.
  - Large scale data
  - Data with high dimension

Random Forest

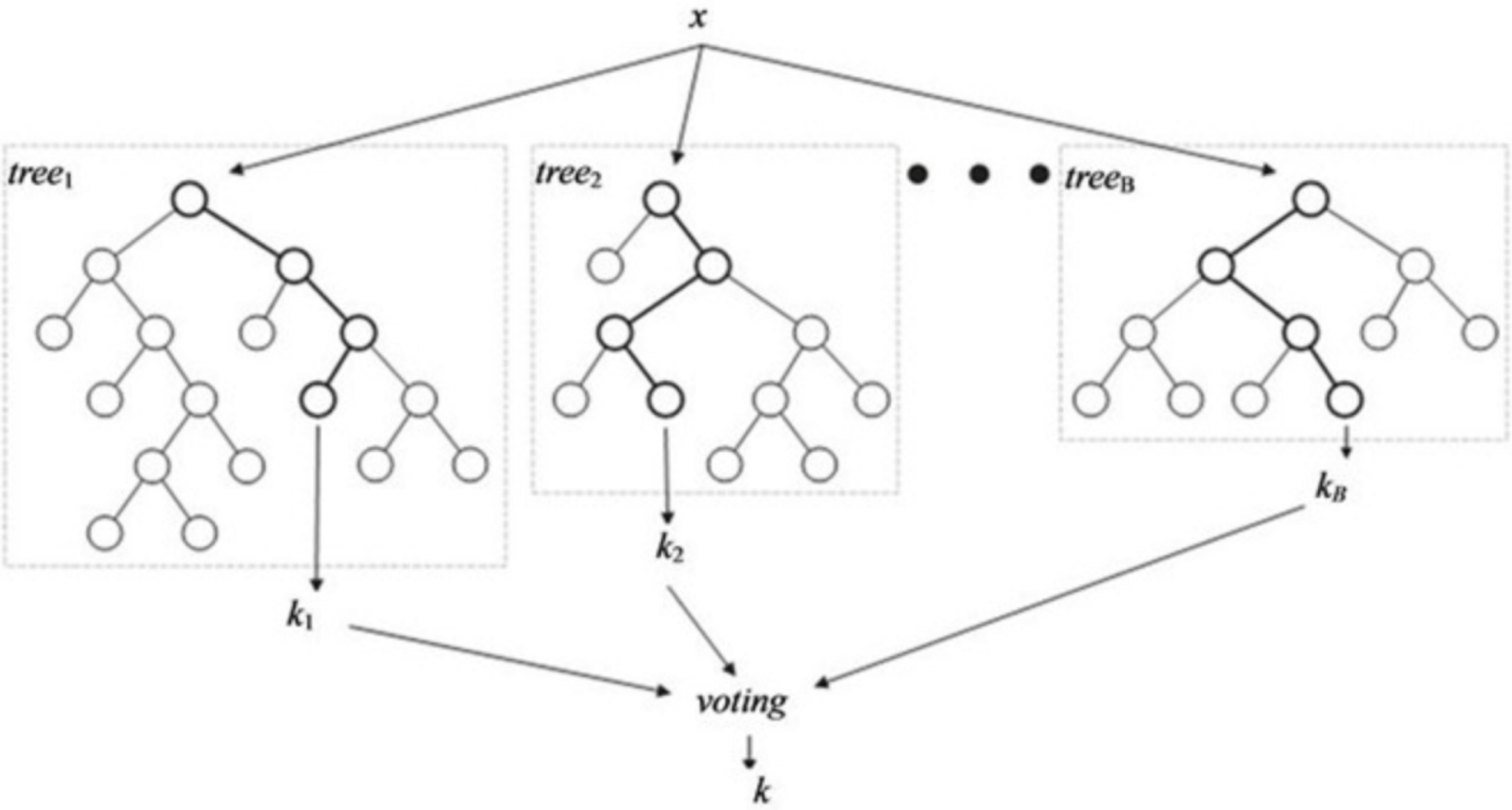
A collection of decision trees

Building stage

- Only a subset of attributes are chosen to make decision at each node of the tree
- Select best split from selected subset of attributes.

Classifying

- All trees are used to make a decision.
- The final class label is determined statistically e.g. majority rule.


x
tree1
tree2
treeB
k1
k2
kB
voting
k

Random Forest

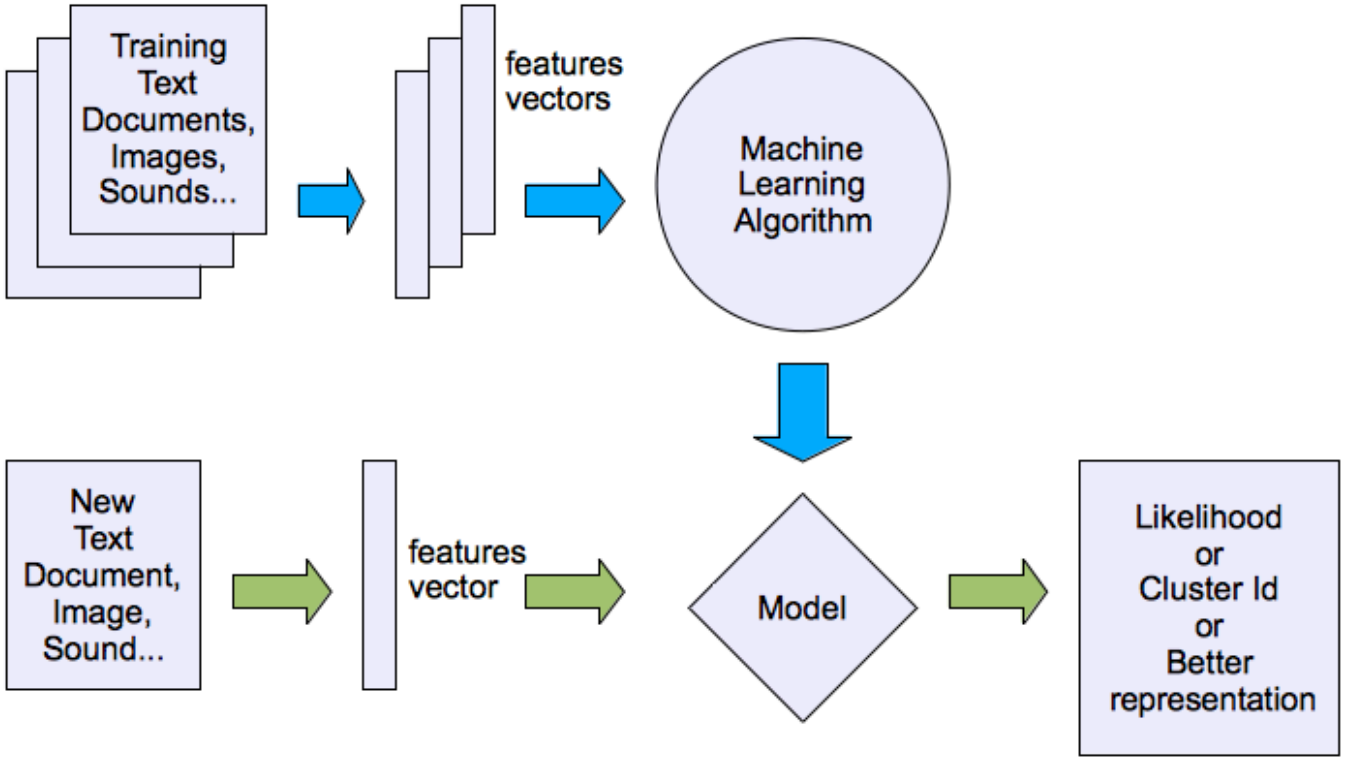
Advantage

- Accurate
- Good for high dimensional data
- Reduce the need of feature detection and selection.
- Give estimate on important features.

Disadvantage

- More computational requirement
- Less effective with noisy training data.
- The result is an statistical ensemble and might be hard to interpret.

Unsupervised Learning



What are we learning?

Unsupervised Learning

- Learning “structure embedded within the data”
  - What normally happens.
  - Better representations of the data
- No output models
- Common techniques/examples :
  - Clustering: membership based on instance similarity
  - Association: connections/inferences among instances
  - Density estimation: Gaussian mixture model
  - Reduction/Summarization: alternative representations of the data

Learning Data Representation

Principle Component Analysis

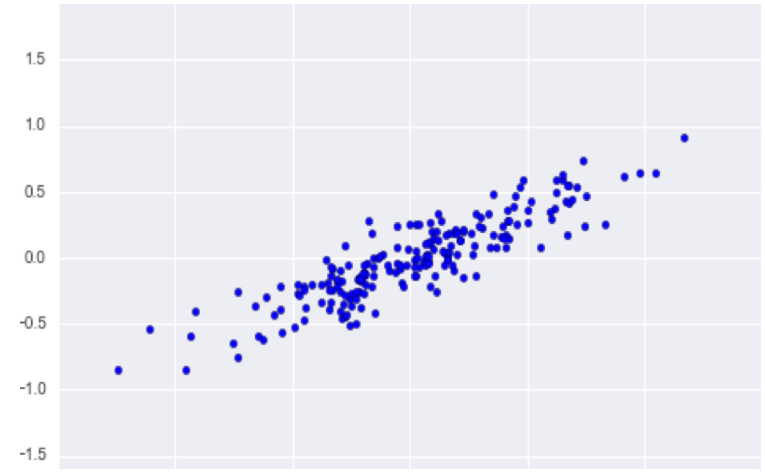
- A data transformation method:

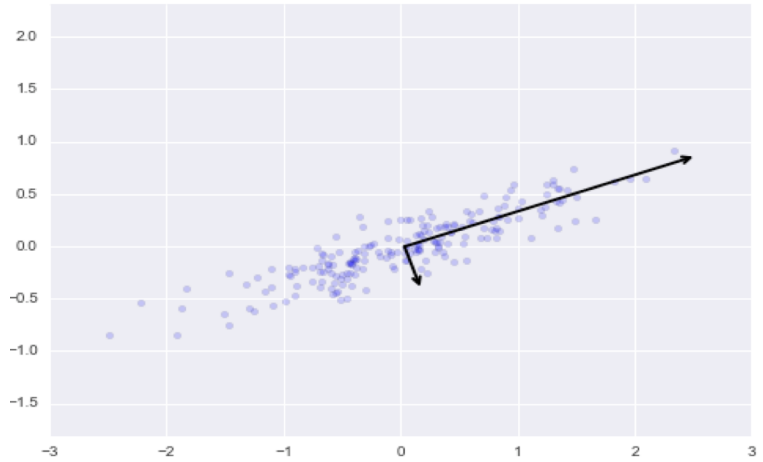
$$x = a_1 v_1 + a_2 v_2 + \cdots + a_N v_N$$

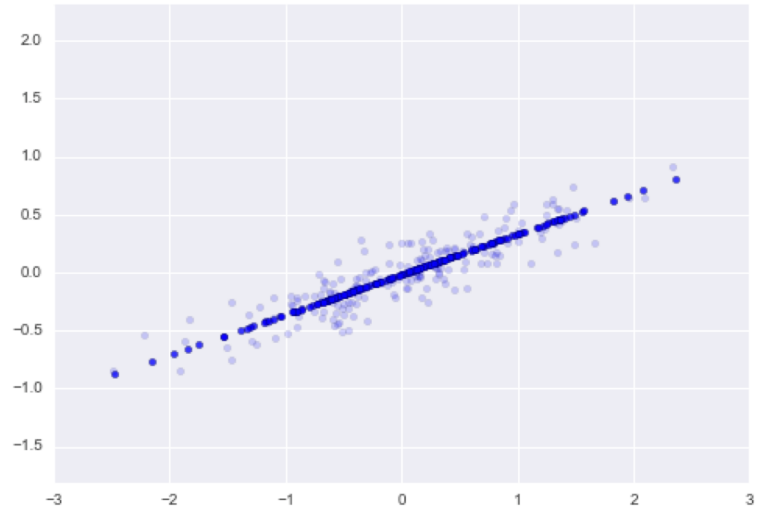
$$\hat{x} = b_1 u_1 + b_2 u_2 + \cdots + b_K u_K$$

- Potential usages
  - Alternative data representation
  - Data visualization
  - Noise filtering
  - Feature extraction

Change Data Representation



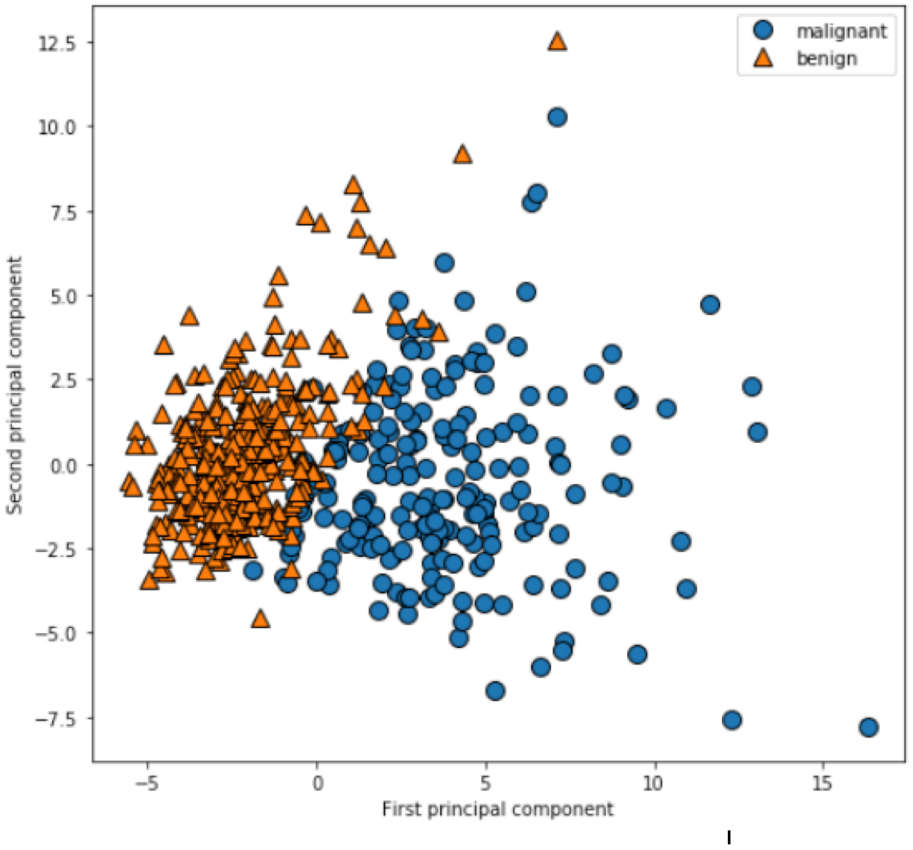




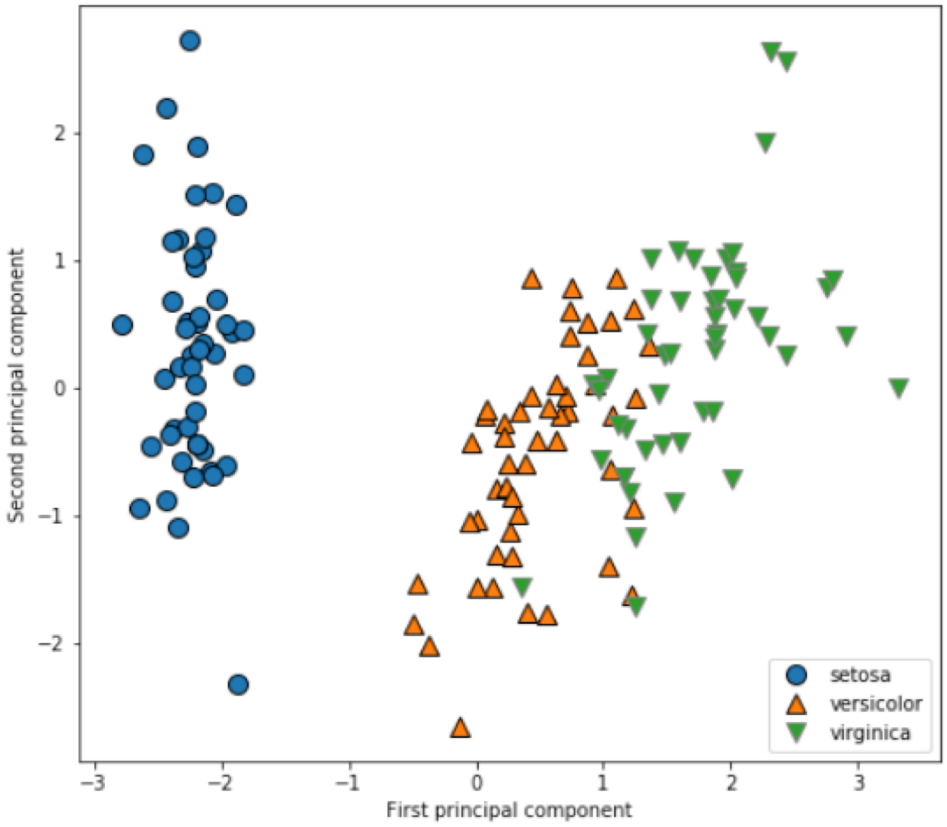
Visualizing High Dimensional Data

- A common usage of PCA is to visualize high dimensional data using the first two components.

- Breast Cancer Data:
  - 2 classes
  - 569 observation
  - 30 features



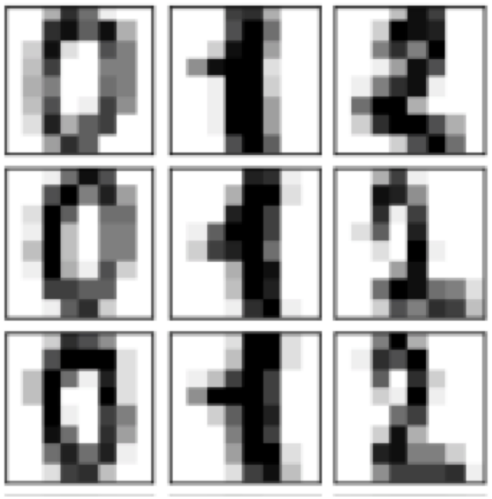
- Iris data set
  - 3 classes
  - 150 X 4



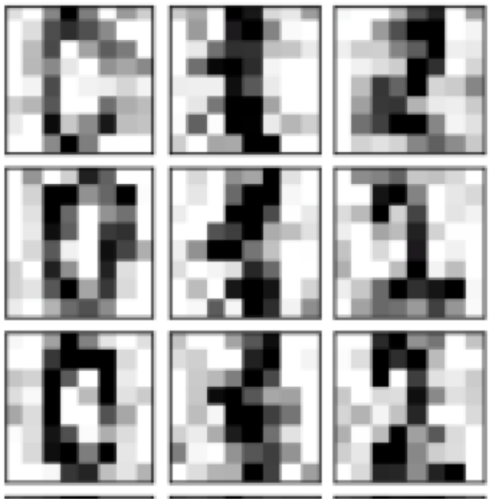
PCA as Noise Filtering

- Components with smaller eigenvalue means less variance
- Components with variance much larger value are less sensitive to the noise.
- Hence, by selecting fewer components to reconstruct the data, we can remove noise points.

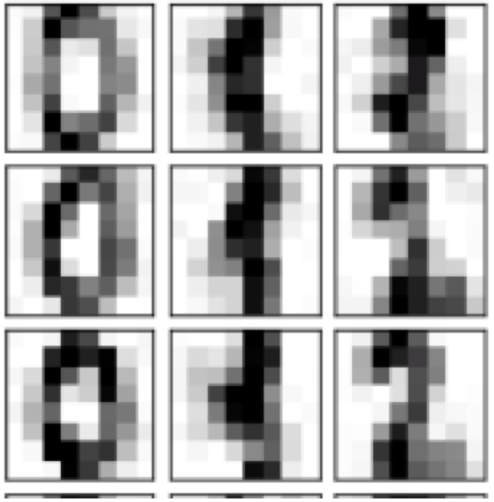
Original Data



Noise Data



Filtered Data



Can it be used in the reverse to detect outliers?

PCA Summary

- PCA is more than just for dimensionality reduction.
- Easy to compute and interpretable
- Sensitive to outliers
  - Randomized PCA
  - SparsePCA
- Works best with linear relationships within the data?

Learning Data Structure

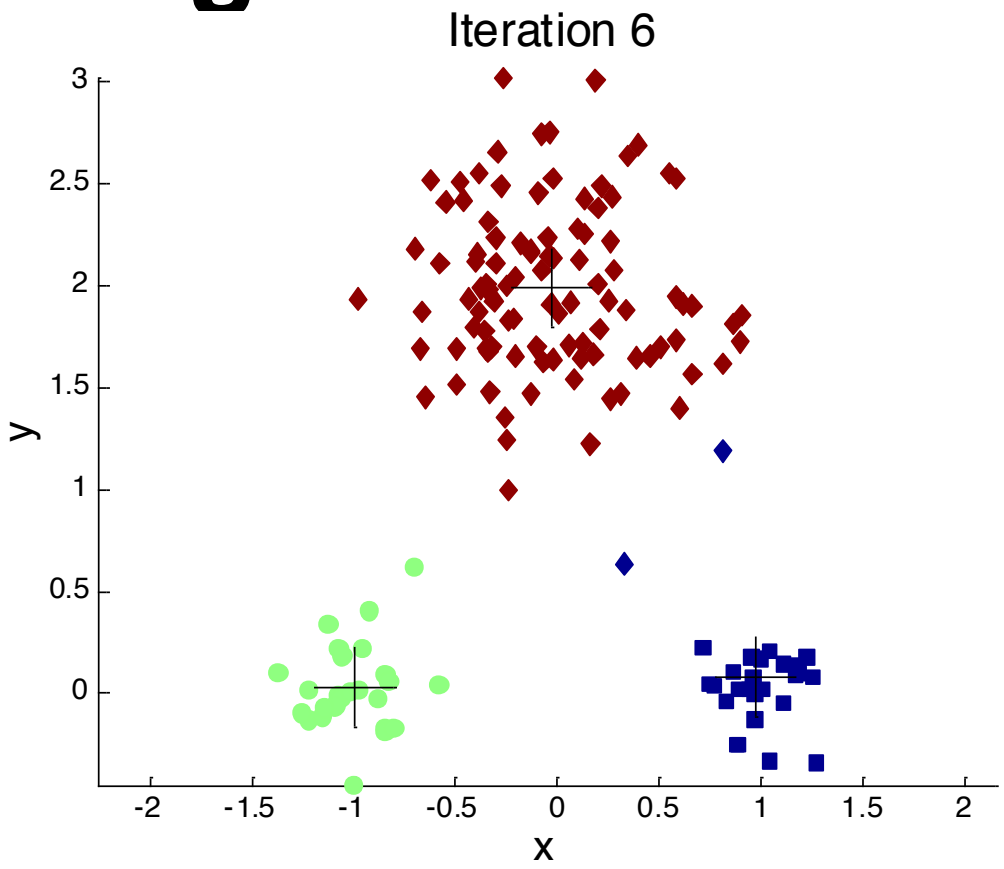
K-means Clustering

- Partitional clustering approach
- Each cluster is associated with a centroid (center point)
- Each point is assigned to the cluster with the closest centroid
- Number of clusters, K, must be specified
- The basic algorithm is very simple

1: Select K points as the initial centroids.
2: **repeat**
3: Form K clusters by assigning all points to the closest centroid.
4: Recompute the centroid of each cluster.
5: **until** The centroids don't change

An Example of k-means Clustering

K=3

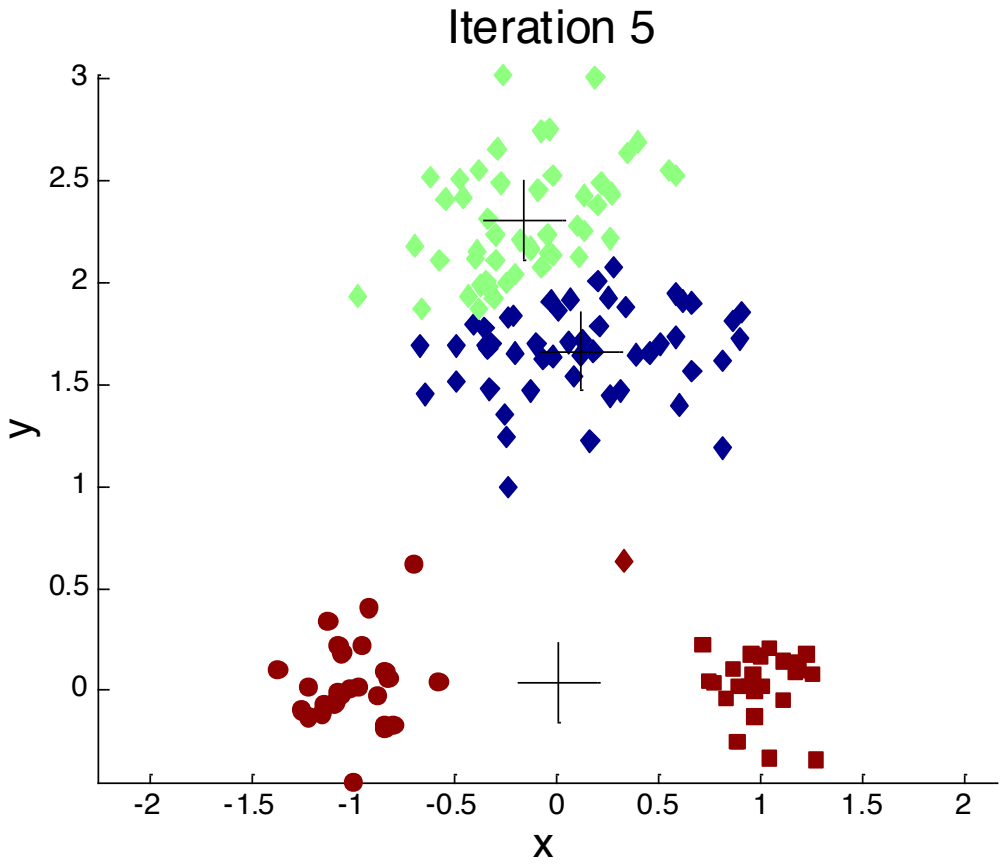


Examples are from Tan, Steinbach, Kumar *Introduction to Data Mining*

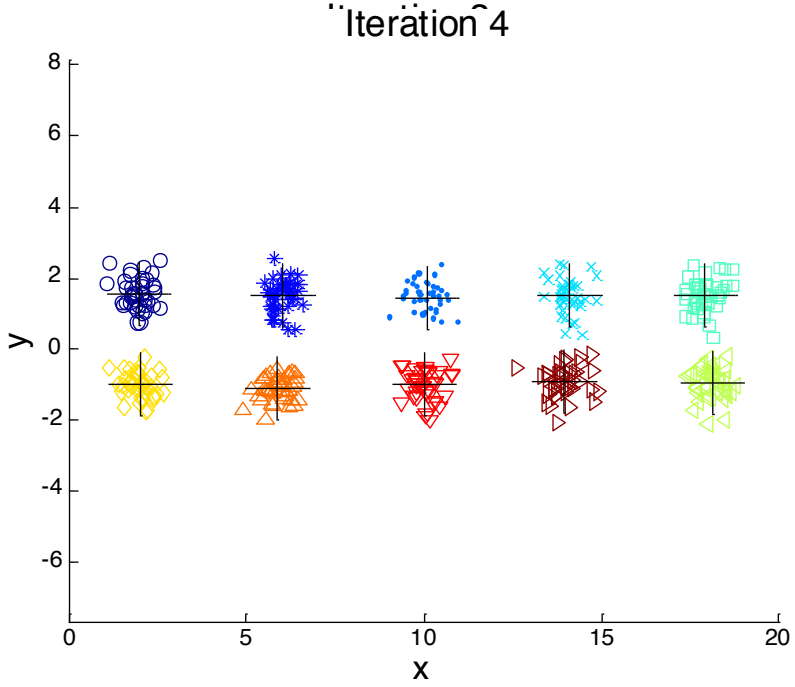
More about on K-means

- Will the algorithm always converge?
  - Convergence, algorithm reaches its goal, e.g. error within a threshold
  - K-means will usually converge fast for common similarity measures.
    - Euclidean distance
    - Mahalanobis distance
    - Bregman divergence
- Will the algorithm guarantee to find the most optimal solutions?
- Will the algorithm always find the same solution.

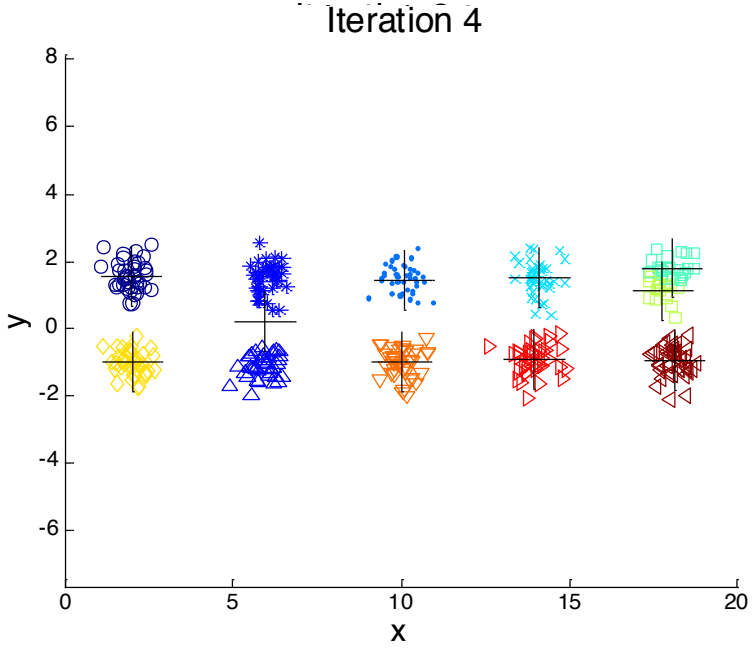
Initial Centroids Selection



10 Clusters Example



Starting with two initial centroids in one cluster of each pair of clusters



Starting with some pairs of clusters having three initial centroids, while other have only one.

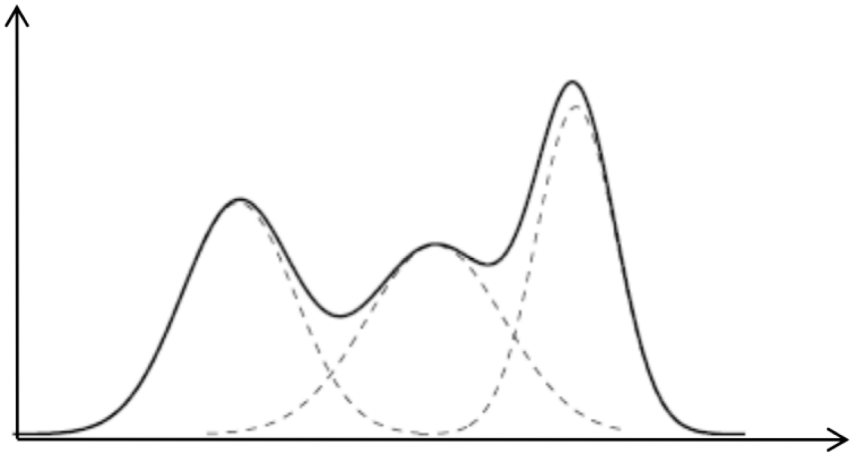
BiSecting K-means

- A top down clustering
  - Starts with one cluster of all data points
  - Split clusters at each level.
- Generates an hierarchical structure of data
- Inherent the same randomness of k-means.

Learning Data Density

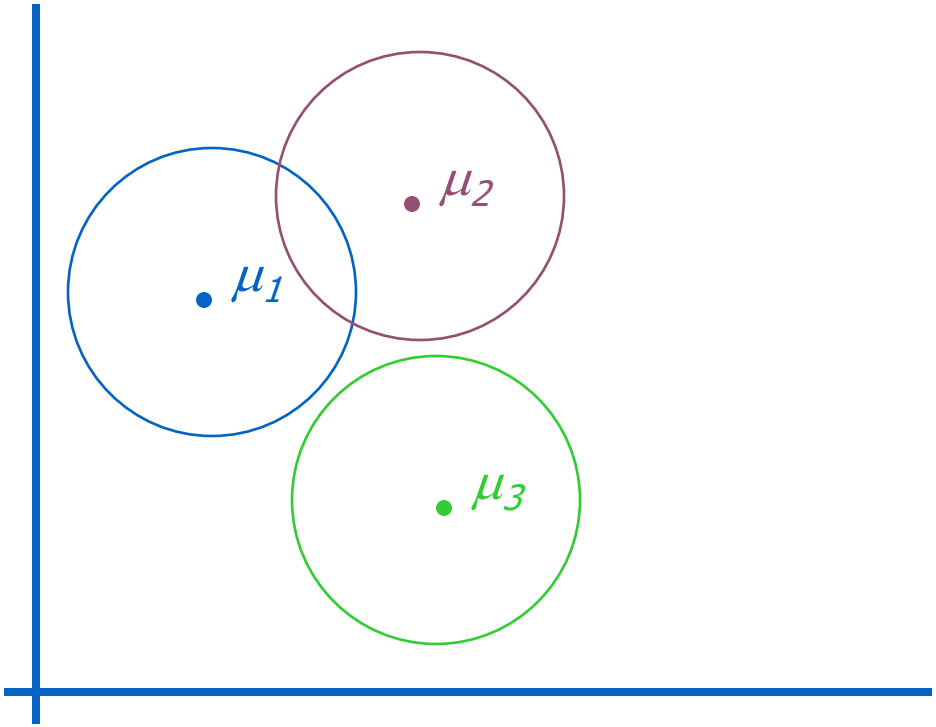
Gaussian Mixture Model

- Consider the observation is a mixture of K Gaussian distributions.
- Given a set of input data, estimate distribution parameters that can best describe date.
- Each cluster consists of observation follows the same Gaussian distribution.



Basic Assumption

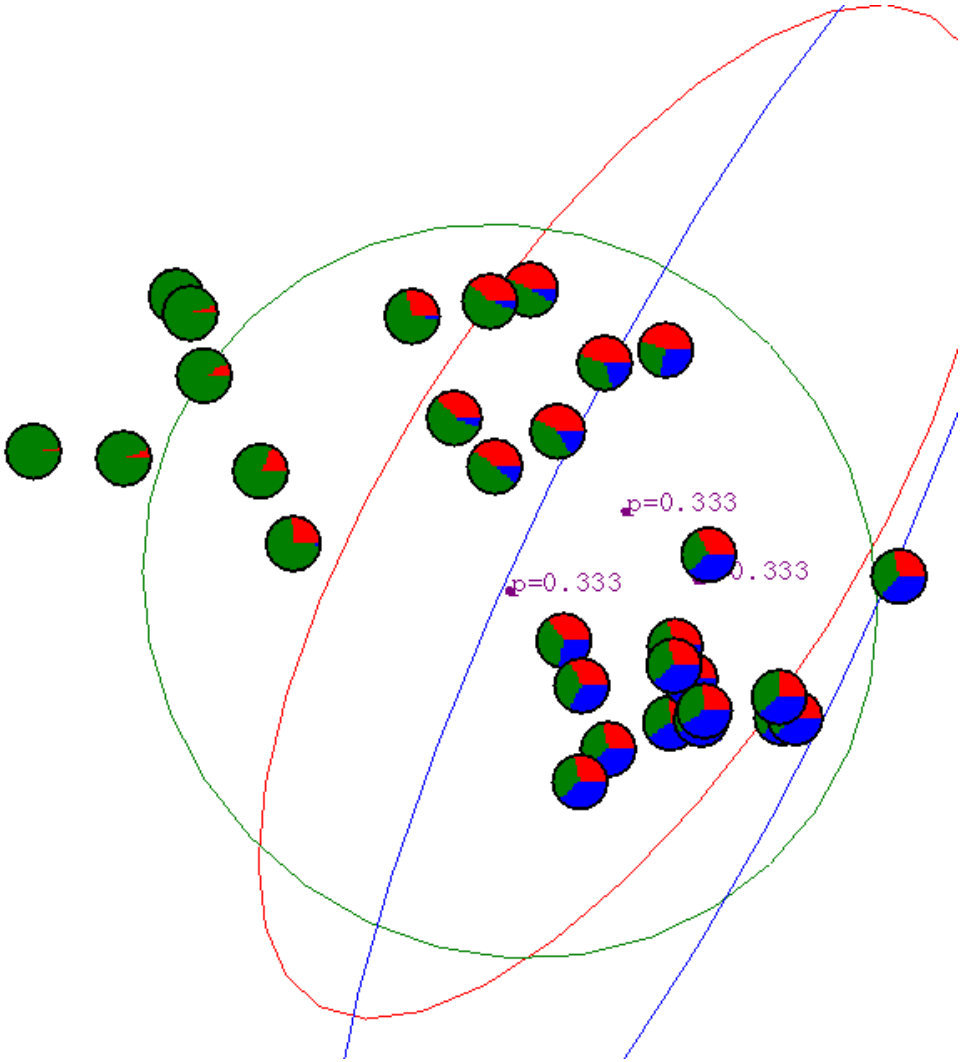
- There are k components. The i' th component is called ω_i
- Component ω_i has an associated mean vector μ_i
- Each component generates data from a Gaussian with mean μ_i and covariance matrix $\sigma^2\boldsymbol{I}$



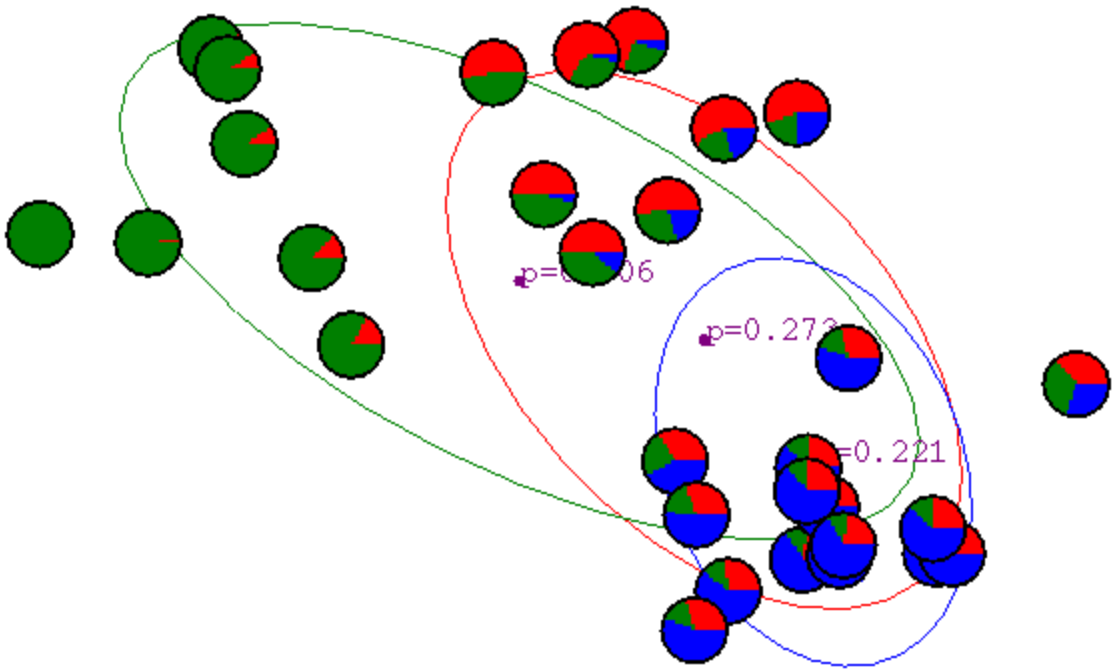
General Steps

- Steps in GMM,
  - choose initial guess of the parameters
  - E-step: Find excepted classes for each data point
  - M-Step: update parameters of each component, e.g. location, normalization and shape.
- Goal: maximize the log likelihood of all data.
- The output cluster is described by a smooth Gaussian model instead of sphere.

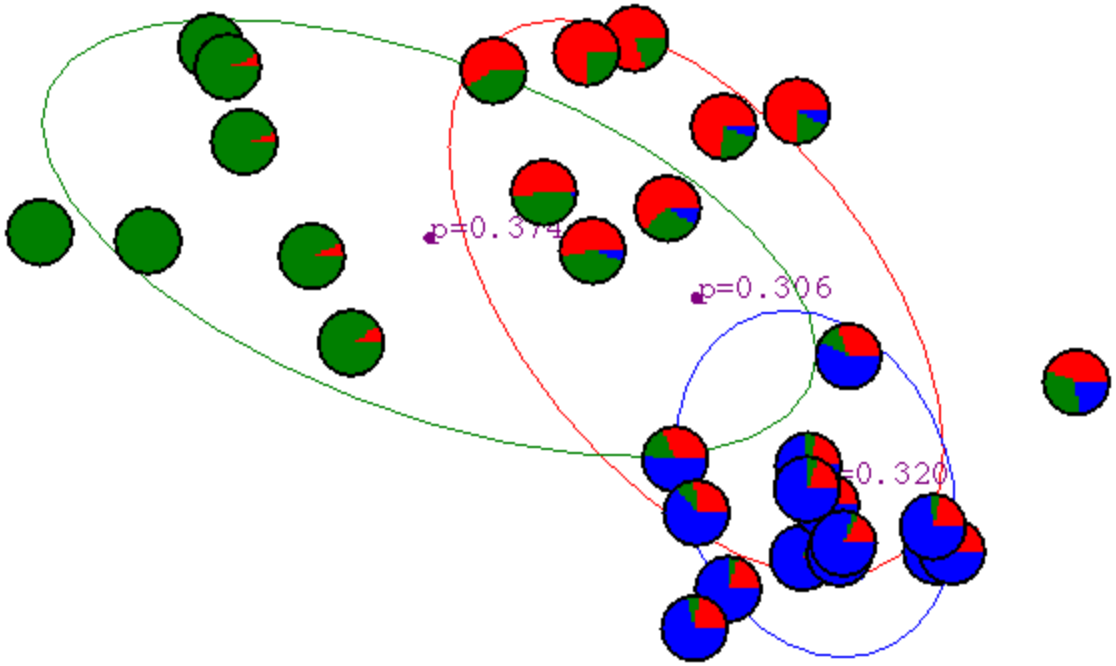
Gaussian Mixture Example: Start



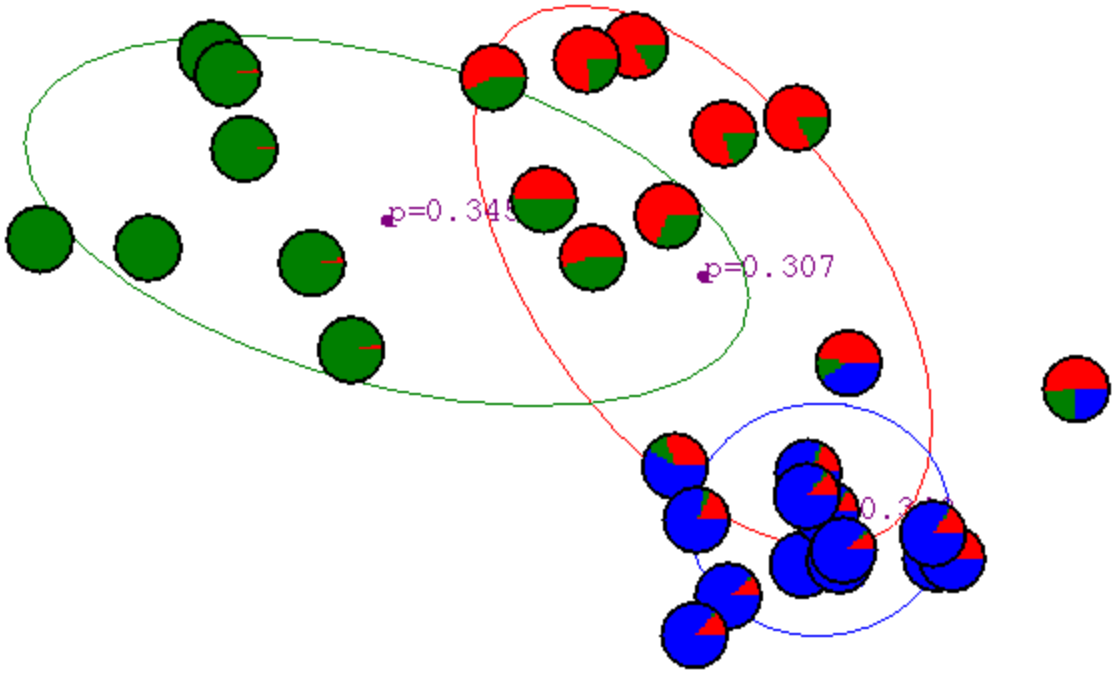
After first iteration



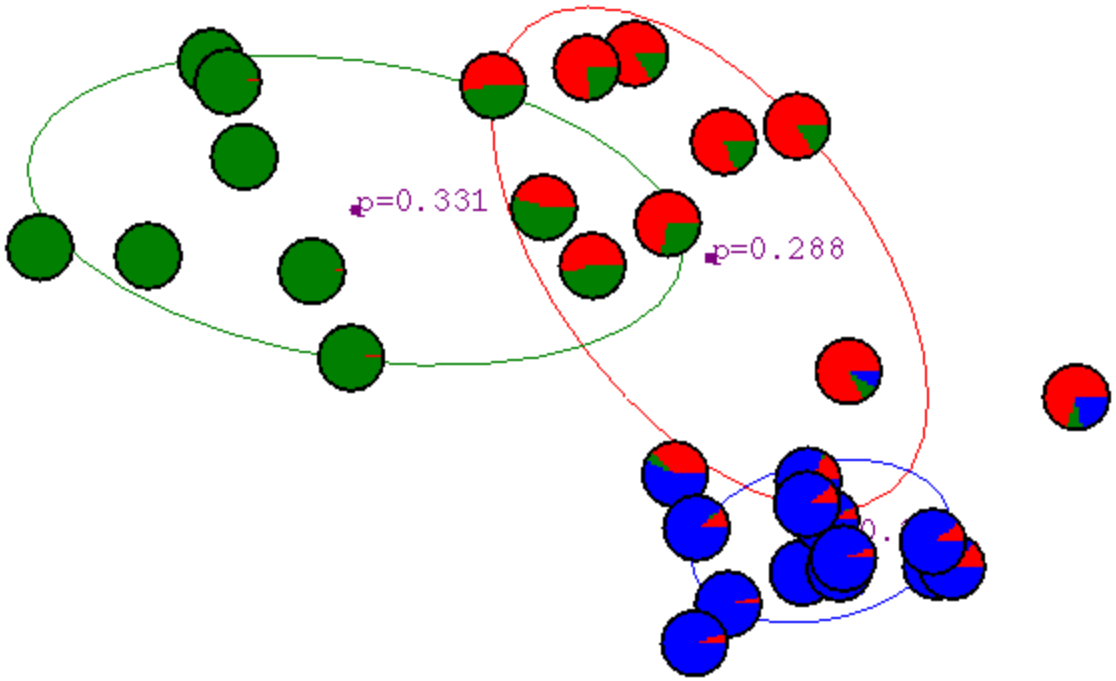
After 2nd iteration



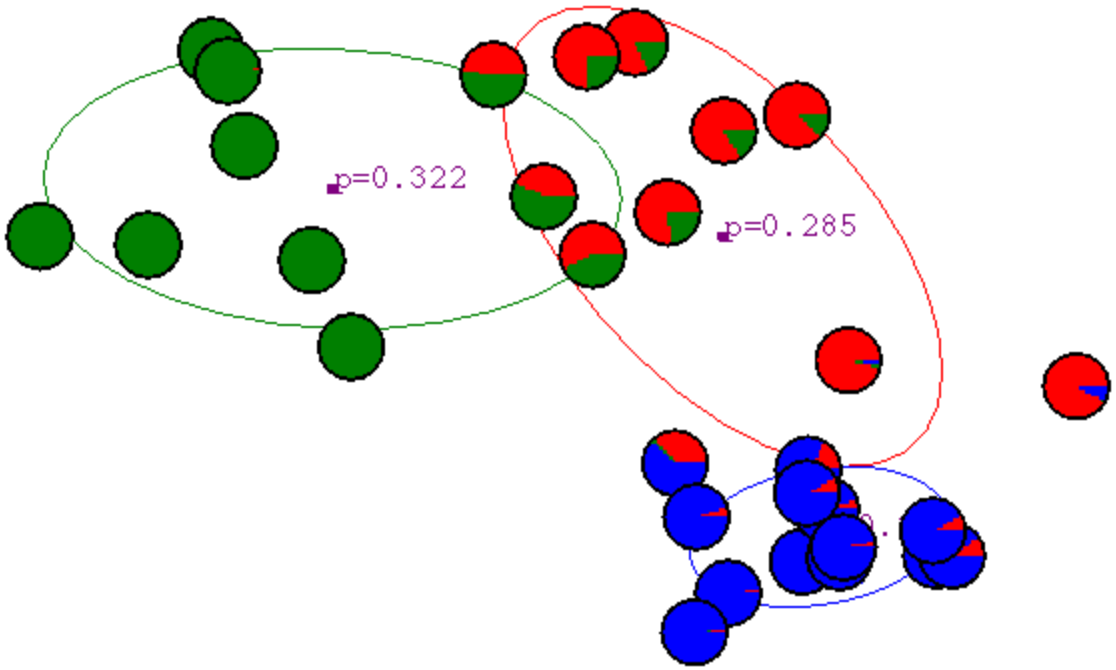
After 3rd iteration



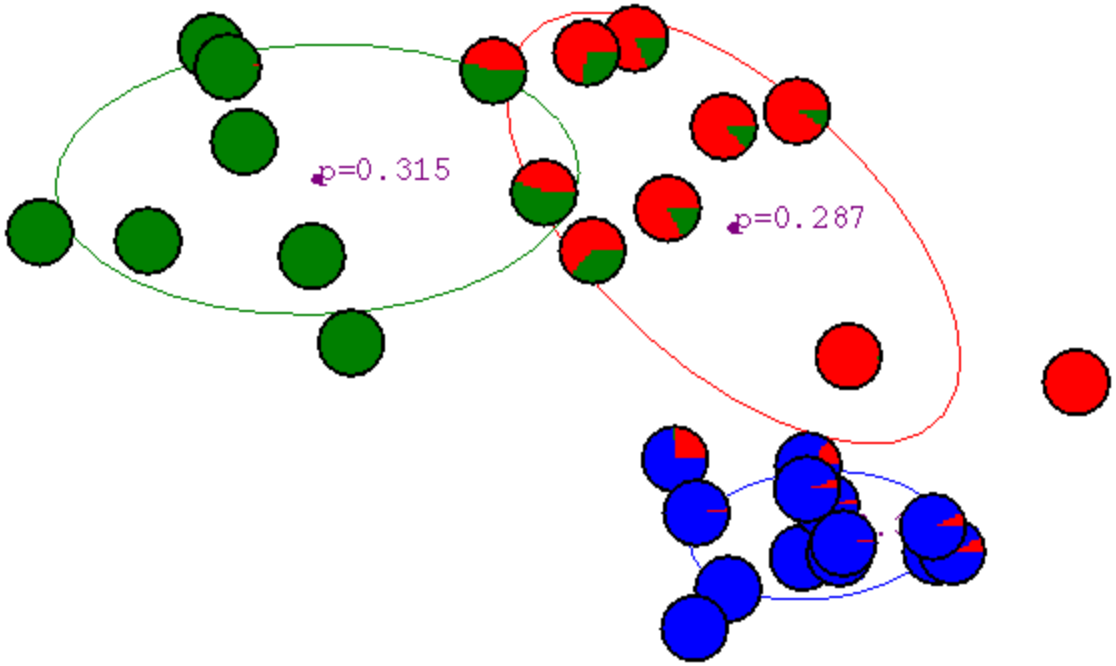
After 4th iteration



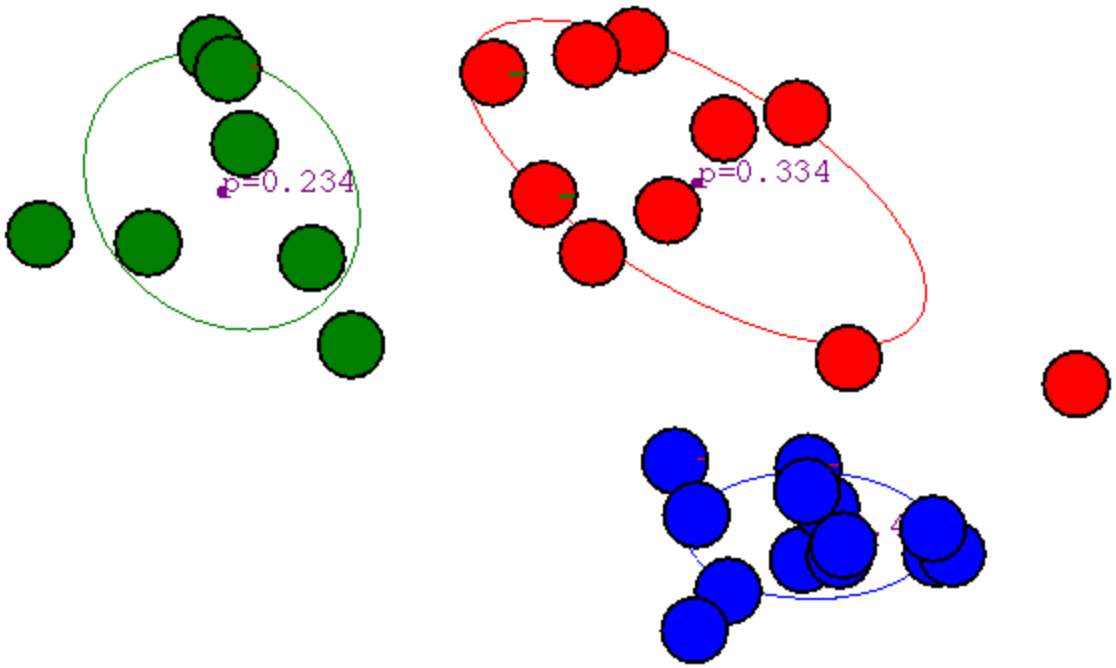
After 5th iteration



After 6th iteration

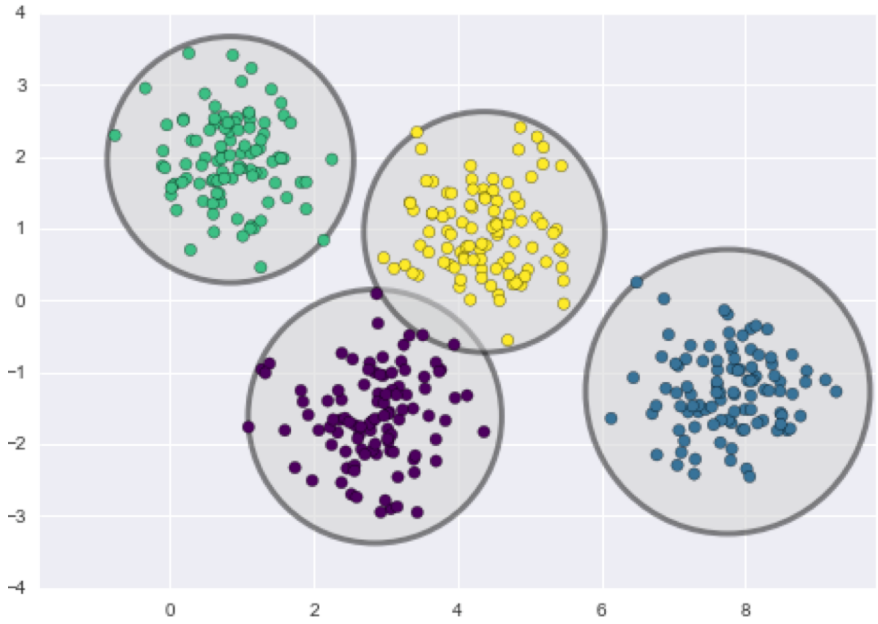


After 20th iteration

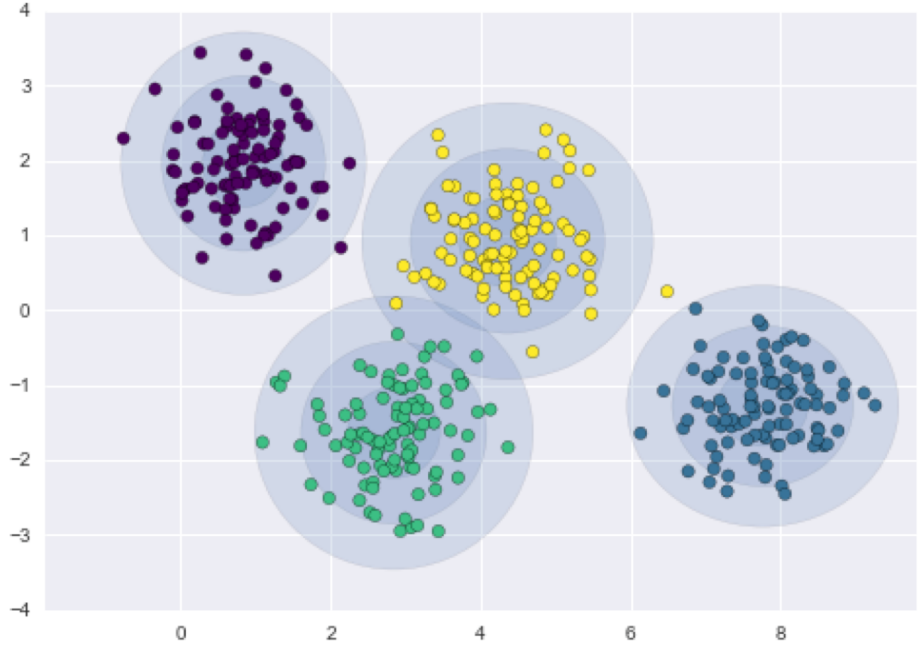


GMM vs. K-Means

Clustering Boundaries in K-Means



Clustering Boundaries with GMM



Summary

- Unsupervised learning learns the “data”.
- Supervised learning learns how the data should be labeled.
- Other topics and tools:
  - SGD, Neuron Network and deep learning
  - Reinforcement learning
  - Anomaly detection
  - Association and recommendation methods
  - Graph analysis
  - …