EE 4146 Data Engineering and Learning Systems

Lecture 14: Summary

Semester A, 2021-2022

Schedules

Week	**Date**	**Topics**
1	Sep. 1	Introduction
2	Sep. 8	Data exploration
3	Sep. 15	Feature reduction and selection (HW1 out)
4	Sep. 22	Mid-Autumn Festival
5	Sep. 29	Clustering I: Kmeans based models (HW1 due in this weekend)
6	Oct. 6	Clustering II: Hierarchical/density based/fuzzing clustering
7	Oct. 13	Midterm (no tutorials this week)
8	Oct. 20	Adverse Weather
9	Oct. 27	Linear classifiers
10	Nov. 3	Classification based on decision tree (Tutorial on project) (HW2 out)
11	Nov. 10	Bayes based classifier (Tutorial on codes) (HW2 due in this weekend)
12	Nov. 17	Non-linear Perceptron and Classifier ensemble
13	Nov. 24	Deep learning based models (Quiz)
14		Summary: based on the poll, we will do off-line video recording for the summary and will upload the video before Dec. 1st.

Project

- Reports: (5%)
  - Suggestions: Use standard latex template for a conference
  - **Overleaf** is good way to write a paper with latex

  http://ras.papercept.net/conferences/support/tex.php
  - Word template

  http://ras.papercept.net/conferences/support/word.php
  - Around 4-6 pages
  - Including abstract, introduction, method, results and conclusion
- Submit report before Dec. 8 through canvas

Final exam

- 10 Multiple choices questions (20%)
- 10 true/false (10%)
- 2 Essay questions (10%)
- 6 calculation and understanding related questions (60%)

Outline

- Summary joint with more examples

Schedules

Week	Date	Topics
1	Sep. 1	Introduction
2	Sep. 8	Data exploration
3	Sep. 15	Feature reduction and selection (HW1 out)
4	Sep. 22	Mid-Autumn Festival
5	Sep. 29	Clustering I: Kmeans based models (HW1 due in this weekend)
6	Oct. 6	Clustering II: Hierarchical/density based/fuzzing clustering
7	Oct. 13	Midterm (no tutorials this week)
8	Oct. 20	Adverse Weather
9	Oct. 27	Linear classifiers
10	Nov. 3	Classification based on decision tree (Tutorial on project) (HW2 out)
11	Nov. 10	Bayes based classifier (Tutorial on codes) (HW2 due in this weekend)
12	Nov. 17	Non-linear Perceptron and Classifier ensemble
13	Nov. 24	Deep learning based models (Quiz)
14		Summary: based on the poll, we will do off-line video recording for the summary and will upload the video before Dec. 1st.

Lecture 2: Data exploration

- Attributes and Objects
  - Types of attributes (Nominal, Ordinal, Interval, Ratio)
- Types of Data (record, graph, ordered)
- Data Quality
  - Noise and outliers; wrong data; fake data; missing values; duplicate data
- Similarity and Distance
  - Euclidean, Minkowski Distance
  - Simple Matching and Jaccard Coefficients

SMC = number of matches / number of attributes

$= (f_{11} + f_{00}) / (f_{01} + f_{10} + f_{11} + f_{00})$

J = number of 11 matches / number of non-zero attributes

$= (f_{11}) / (f_{01} + f_{10} + f_{11})$

  - Cosine Similarity, Correlation (property)
  - Entropy: -plogp

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

Lecture 2: Data exploration

$\mathbf{x} = 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0$

$\mathbf{y} = 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1$

$f_{01} = 2$ (the number of attributes where **x** was 0 and **y** was 1)

$f_{10} = 1$ (the number of attributes where **x** was 1 and **y** was 0)

$f_{00} = 7$ (the number of attributes where **x** was 0 and **y** was 0)

$f_{11} = 0$ (the number of attributes where **x** was 1 and **y** was 1)

$$\begin{aligned} \text{SMC} &= (f_{11} + f_{00}) / (f_{01} + f_{10} + f_{11} + f_{00}) \\ &= (0+7) / (2+1+0+7) = 0.7 \end{aligned}$$

$$\text{J} = (f_{11}) / (f_{01} + f_{10} + f_{11}) = 0 / (2 + 1 + 0) = 0$$

Lecture 2: Data exploration

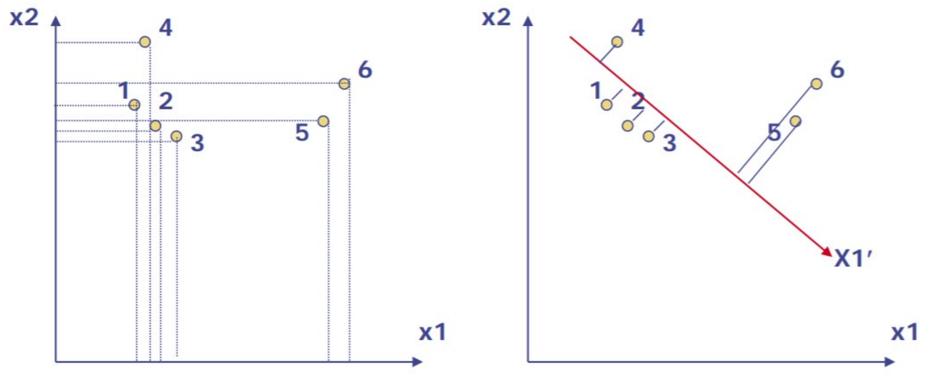
- Data Preprocessing:
  - Aggregation: combine two or more attributes (or objects) into a single attribute (or object) (example: Precipitation)
  - Sampling: simple random sampling; stratified sampling (Split the data into several partitions; then draw random samples from each partition)
  - Discretization: converting a continuous attribute into an ordinal attribute
  - Binarization: map a continuous or categorical attribute into one or more binary variables
  - Attribute transformation: x^k, $\log(x)$, e^x, $|x|$
  - Dimensionality reduction
  - Feature subset selection

Lecture 3: Feature reduction and selection

- PCA: converts a set of observations of possibly correlated variables int o a set of values of linearly uncorrelated variables called principal com ponents
- Computes n-dim subspace such that the projection of the data points onto the subspace has the largest variance among all n-dim subspaces.

$$\alpha_1 = \arg\max_{\alpha} \left(\mathrm{var}\left(\alpha^T \mathbf{X} \right) \right), \alpha \in \mathbb{R}^{p\times 1}$$

$$C = \frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T$$ is the <u>covariance matrix</u> $$C\alpha = \lambda\alpha$$



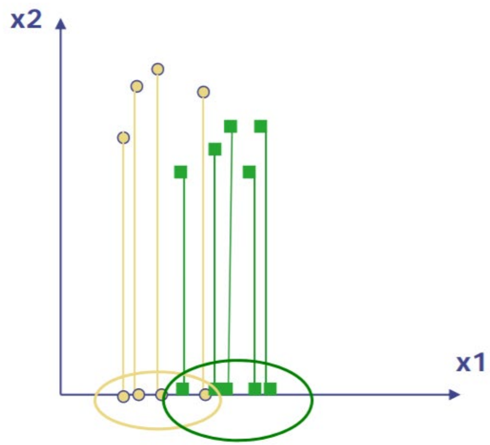
PCA projection

Lecture 3: Feature reduction and selection

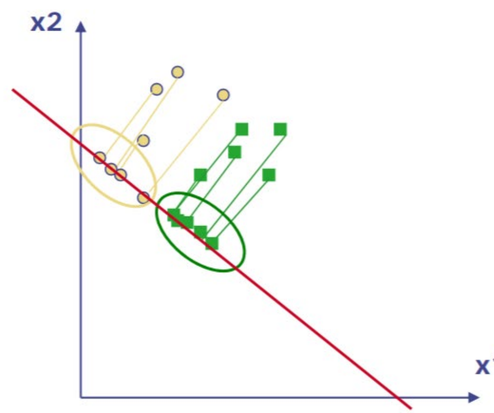
- LDA: attempt to maximize the between class scatter, while minimizing the within class scatter

$$\boxed{W_{opt} = \arg\max_{W} \frac{\left|\tilde{S}_B\right|}{\left|\tilde{S}_W\right|} = \arg\max_{W} \frac{\left|W^T S_B W\right|}{\left|W^T S_W W\right|}}$$

$$S_W^{-1} S_B w = \lambda w$$



Poor Projection



Good

Lecture 3: Feature reduction and selection

- Based on the lecture slides, please do following LDA analysis; For give data: Class 1, {(3,2);(2,3);(4,4);(3,1);(3,5);(3,3)} Class 2, {(9,9);(10,9);(8,7);(8,10);(9,6);(7,9)}

(a) Plot the data in the image;

(b) Calculate the class mean and covariance matrix for these two classes;

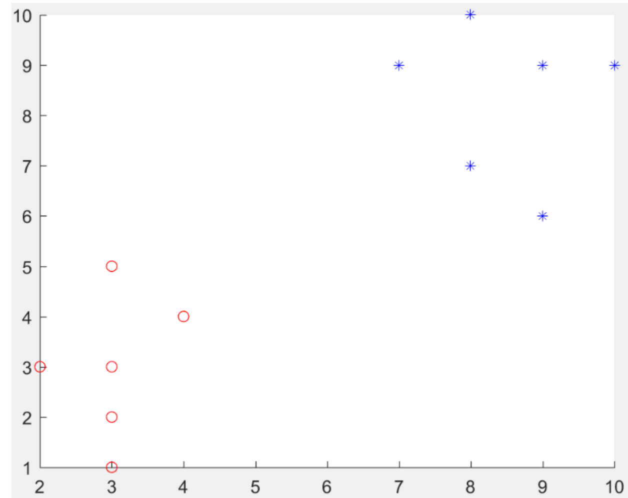
(c) Calculate the Within-class scatter matrix and Between-class scatter matrix;

(d) Write the generalized eigen value problem for the LDA;

(e) Compute the projection vector.

Lecture 3: Feature reduction and selection

```
% samples for class 1&2
X1=[3,2;2,3;4,4;3,1;3,5;3,3];
X2=[9,9;10,9;8,7;8,10;9,6;7,9];

% plot the data
scatter (X1(:,1),X1(:,2),'ro');hold on;
scatter (X2(:,1),X2(:,2),'b*');

% class means
Mu1=mean(X1)';
Mu2=mean(X2)';

%covariance matric of the first & second class
S1=cov(X1);
S2=cov(X2);

% within-class scatter matrix
Sw=S1+S2;

% between-class scatter matrix
SB=(Mu1-Mu2)*(Mu1-Mu2)';

%computing the LDA projection
invSw=inv(Sw);
invSw_by_SB=invSw*SB;

% getting the projection vector
[V,D]=eig(invSw_by_SB);

% the projection vector
W=V(:,1);
```



(b) Class mean of class 1 = $\begin{bmatrix}3\\3\end{bmatrix}$

Class mean of class 2 = $\begin{bmatrix}8.5\\8.3333\end{bmatrix}$

Covariance matrix of class 1 = $\begin{bmatrix}0.4 & 0.2\\0.2 & 2\end{bmatrix}$

Covariance matrix of class 2 = $\begin{bmatrix}1.1 & -0.2\\-0.2 & 2.2667\end{bmatrix}$

(c) Within-class scatter matrix = $\begin{bmatrix}1.5 & 0\\0 & 4.2667\end{bmatrix}$

Between-class scatter matrix = $\begin{bmatrix}30.25 & 29.3333\\29.3333 & 28.4444\end{bmatrix}$

(d) The generalized eigen value problem for the LDA is to find the eigenvector of $S_b S_W^{-1}$.

(e) The required projection vector = $\begin{bmatrix}0.9465\\0.3227\end{bmatrix}$

Lecture 5: Clustering

- Finding groups of objects such that the objects in a group will be similar (or related) to one another and different from (or unrelated to) the objects in other groups
- Partitional Clustering
  - divide data objects into non-overlapping subsets (clusters) such that each data object is in exactly one subset
- Hierarchical clustering
  - A set of nested clusters organized as a hierarchical tree
- Density based clustering
  - Discover clusters of arbitrary shape.
  - Cluster dense regions of objects separated by regions of low density

Lecture 5: Clustering

- Kmeans: partition the data points into K clusters randomly. Find the centroids of each cluster.
- For each data point:
  - Calculate the distance from the data point to each cluster.
  - Assign the data point to the closest cluster.
- Recompute the centroid of each cluster.
- Repeat steps 2 and 3 until there is no further change in the assignment of data points (or in the centroids).

$$\sum_{i \in \text{clusters}} \left\{ \sum_{j \in \text{elements of i'th cluster}} \left\| x_j - \mu_i \right\|^2 \right\}$$

Lecture 5: Clustering

- Examples of Kmeans： For given datasets, K=2; randomly choose C1=(2,2), C2=(3,3), illustrate the first iteration of kmeans steps with Euclidean distance.

No	X	Y
1	1	1
2	2	3
3	1	2
4	3	3
5	2	2
6	3	1

Lecture 5: Clustering

1. D1 = {(1 , 1), (2 , 2)}
$= \sqrt{(2-1)^2 + (2-1)^2}$
= 1.41

2. D1 = {(2 , 3), (2 , 2)}
$= \sqrt{(2-2)^2 + (2-3)^2}$
= 1

3. D1 = {(1 , 2), (2 , 2)}
$= \sqrt{(2-1)^2 + (2-2)^2}$

4. D1 = {(3 , 3), (2 , 2)}
$= \sqrt{(2-3)^2 + (2-3)^2}$
= 1.41

5. D1 = {(2 , 2), (2 , 2)}
$= \sqrt{(2-2)^2 + (2-2)^2}$
= 0

6. D1 = {(3 , 1), (2 , 2)}
$= \sqrt{(2-3)^2 + (2-1)^2}$
= 1.41

1. D2 = {(1 , 1), (3 , 3)}
$= \sqrt{(3-1)^2 + (3-1)^2}$
= 2.82

2. D2 = {(2 , 3), (3 , 3)}
$= \sqrt{(3-2)^2 + (3-3)^2}$
= 1

3. D2 = {(1 , 2), (3 , 3)}
$= \sqrt{(3-1)^2 + (3-2)^2}$

4. D2 = {(3 , 3), (3 , 3)}
$= \sqrt{(3-3)^2 + (3-3)^2}$
= 0

5. D2 = {(2 , 2), (3 , 3)}
$= \sqrt{(3-2)^2 + (3-2)^2}$
= 1.41

6. D2 = {(3 , 1), (3 , 3)}
$= \sqrt{(3-3)^2 + (3-1)^2}$
= 2

C1 = {(1 , 1), (1 , 2), (2 , 2), (3 , 1)}
C2 = {(2 , 3), (3 , 3)}

No	X	Y
1	1	1
2	2	3
3	1	2
4	3	3
5	2	2
6	3	1

- C1=(1.75,1.5)
- C2=(2.5,3)

Lecture 5: Clustering

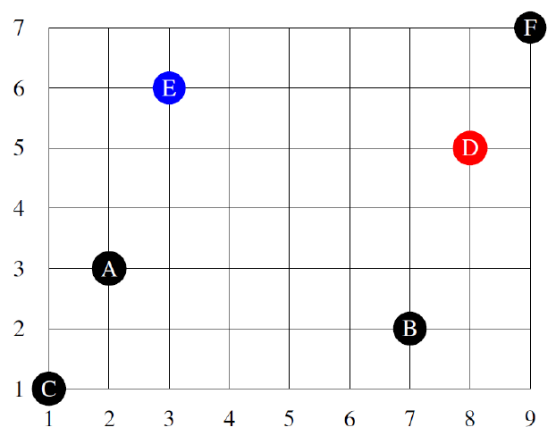
- K-medoids
- *PAM*
  - starts from an initial set of medoids and iteratively replaces one of the medoids by one of the non-medoids if it improves the total distance of the resulting clustering.
  - PAM works effectively for small data sets, but does not scale well for large data sets.
- *CLARA:* draws a sample of the dataset and applies PAM on the sample in order to find the medoids

Lecture 5: Clustering

- Examples of PAM
  - Starts from an initial set of medoids and iteratively replaces one of the medoids by one of the non-medoids if it improves the total distance of the resulting clustering

Consider the following 2-dimensional data set:

	A	B	C	D	E	F
x_1	2	7	1	8	3	9
x_2	3	2	1	5	6	7



Perform the first loop of the PAM algorithm (k = 2) using the Manhattan distance. Select D and E (highlighted in the plot) as initial medoids and compute the resulting medoids and clusters.

Hint: When $C(m)$ denotes the cluster of medoid m, and M denotes the set of medoids, then the total distance TD may be computed as

$$TD = \sum_{m \in M} \sum_{o \in C(m)} d(m, o)$$

Lecture 5: Clustering

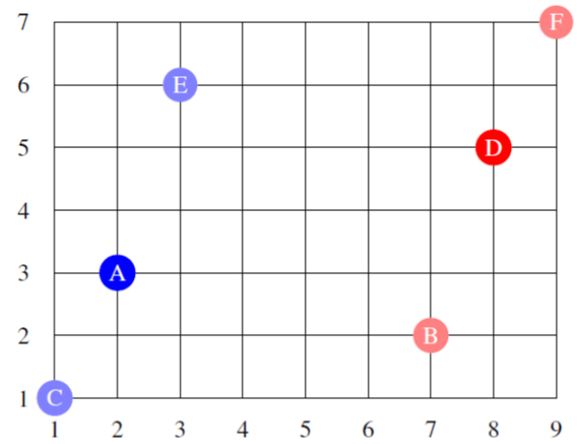
We have the following distance values (values which are clear by symmetry and reflexivity are left out):

	B	C	D	E	F
A	6	3	8	4	11
B		7	4	8	7
C			11	7	14
D				6	3
E					7

The following table shows assignments and $TD_{m \leftrightarrow n}$ value for each pair $(m, n) \in M \times N$ with $M = \{D, E\}$ and $N = \{A, B, C, F\}$.

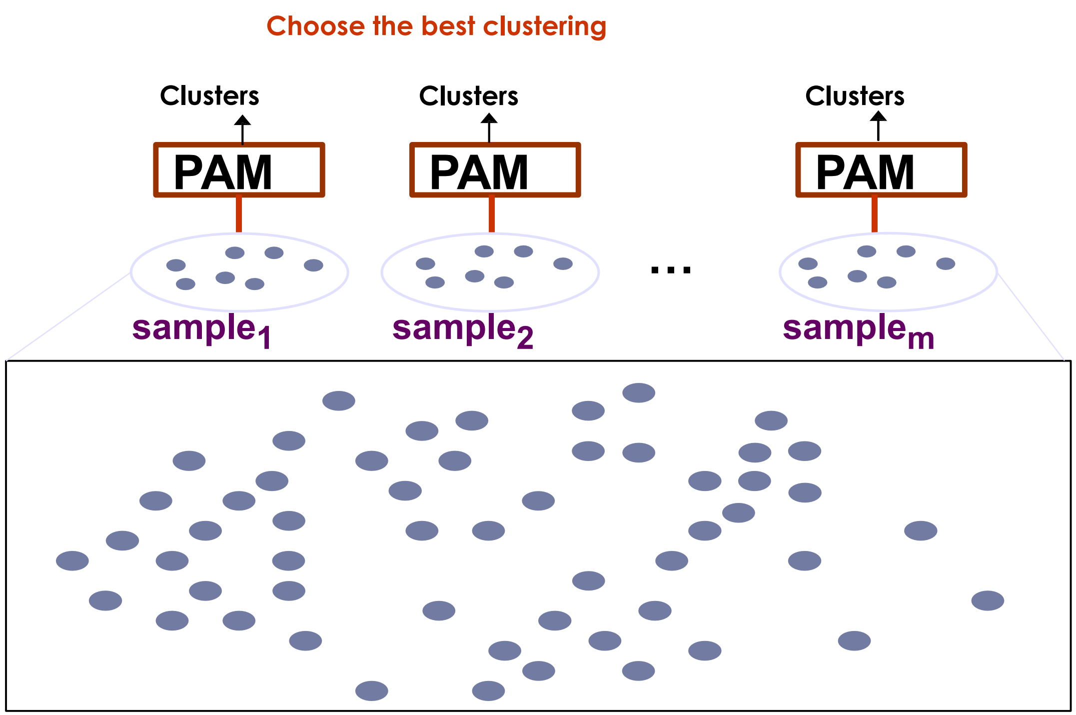
Medoids		Assignment						TD
m_1	m_2	A	B	C	D	E	F	
D	E	1	0	1	0	1	0	**18**
D	A	1	0	1	0	1	0	**14**
D	B	1	1	1	0	0	0	22
D	C	1	0	1	0	0	0	16
D	F	0	0	0	0	0	1	29
E	A	1	1	1	0	0	0	22
E	B	0	1	0	1	0	0	22
E	C	1	1	1	0	0	0	23
E	F	0	1	0	1	0	1	21

The table shows that swapping E and A yields the largest improvement in terms of TD. The updated clustering after the first iteration is shown in the following figure.



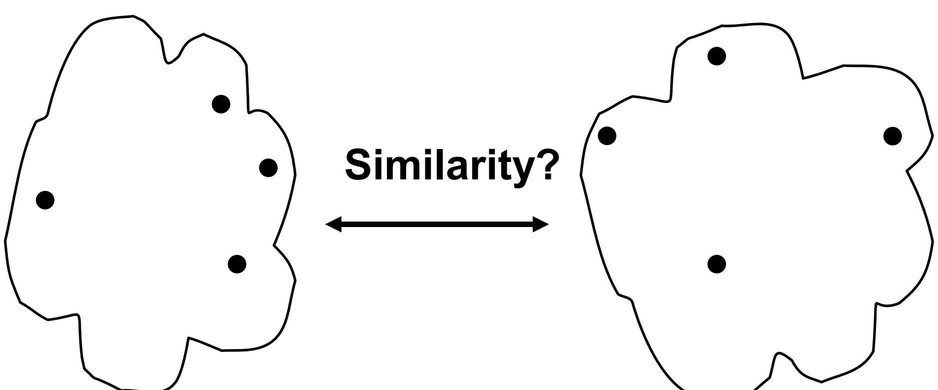
Lecture 5: Clustering

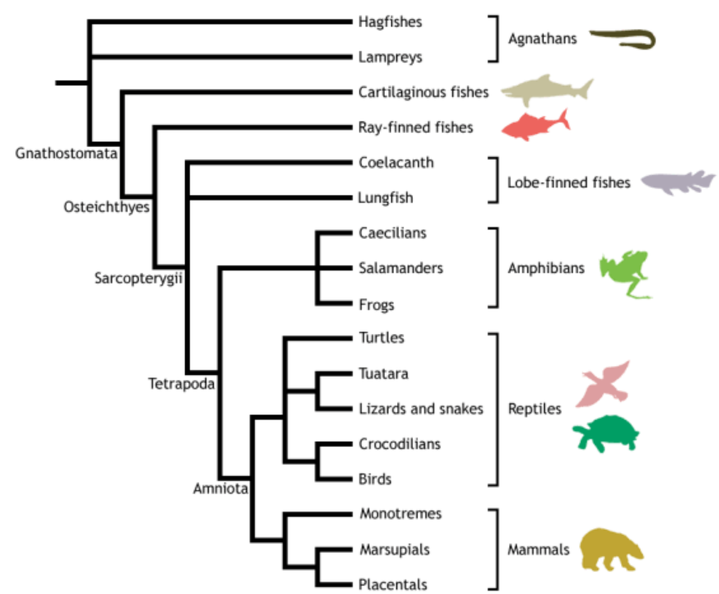
- CLARA (Kaufmann and Rousseeuw in 1990)
  - It draws multiple samples of the data set, applies PAM on each sample, and gives the best clustering as the output.



Lecture 6: Clustering II

- Hierarchical clustering is an alternative approach that does not require a pre-specified choice of K, and which provides a deterministic answer (no randomness)
- Start with clusters of individual points and a proximity matrix
- Merge two clusters based on different similarity
  - MIN: based on two closest points
  - MAX : based on two most distant points
  - Group Average: based on the average distances in the different clusters
  - Distance Between Centroids





Lecture 6: Clustering II

- Examples of Hierarchical clustering For a given matrix, show the hierarchical clustering steps with complete linkage and simple linkage.

	1	2	3	4	5
1	0				
2	9	0			
3	3	7	0		
4	6	5	9	0	
5	11	10	2	8	0

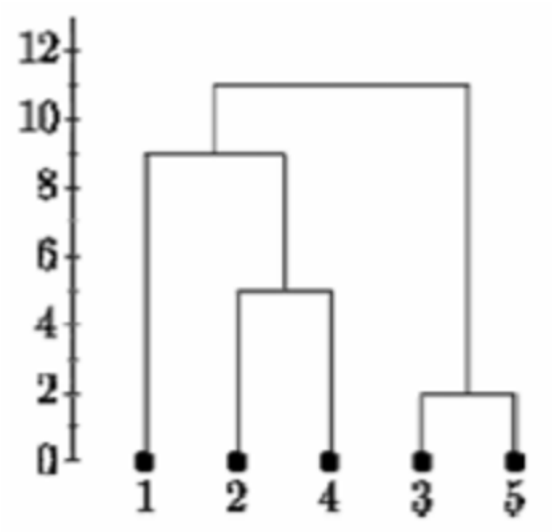
Lecture 6: Clustering II

- Examples of Hierarchical clustering For a given matrix, show the hierarchical clustering steps with complete linkage

	1	2	3	4	5
1	0				
2	9	0			
3	3	7	0		
4	6	5	9	0	
5	11	10	2	8	0

	1	2	3	4	5
1	0				
2	9	0			
3	3	7	0		
4	6	5	9	0	
5	11	10	$\boxed{2}$	8	0

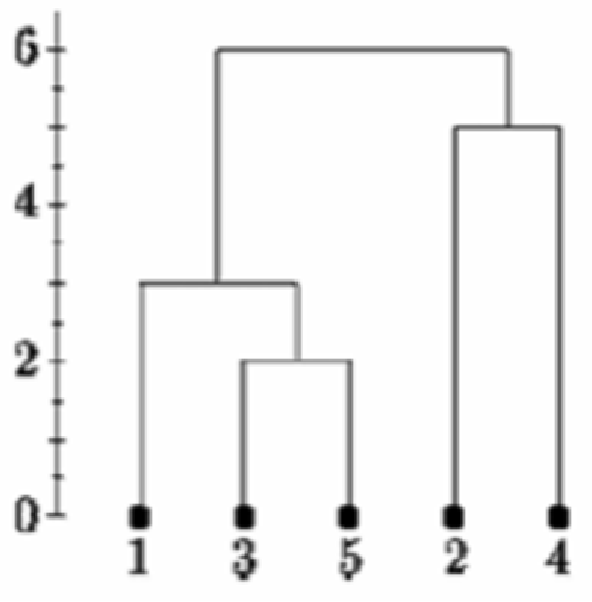
	35	1	2	4
35	0			
1	11	0		
2	10	9	0	
4	9	6	5	0



Lecture 6: Clustering II

- Examples of Hierarchical clustering For a given matrix, show the hierarchical clustering steps with simple linkage.

	1	2	3	4	5
1	0				
2	9	0			
3	3	7	0		
4	6	5	9	0	
5	11	10	2	8	0

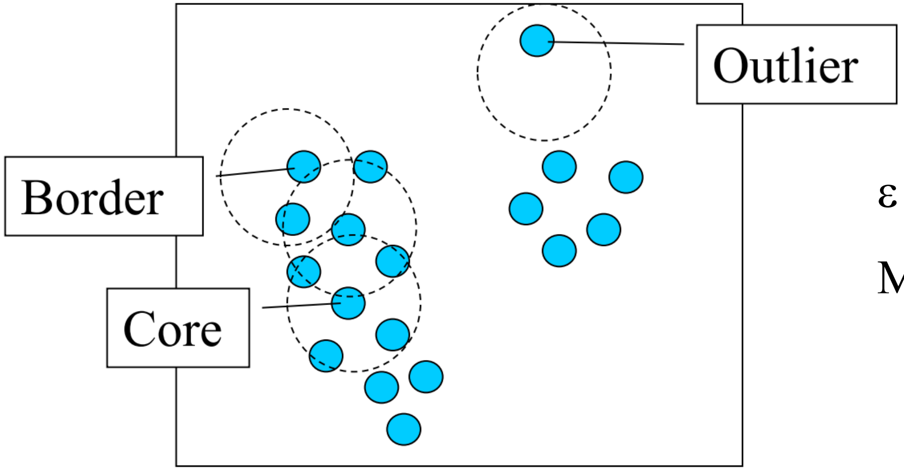


Lecture 6: Clustering II

- Density based clustering
  - A cluster is defined as a maximal set of density-connected points
  - Discovers clusters of arbitrary shape
- DBSCAN: Density-Based Spatial Clustering of Applications with Noise
  - ε -Neighborhood – Objects within a radius of ε from an object.
  - “High density” -ε-Neighborhood of an object contains at least MinPts of objects.

Lecture 6: Clustering II

- According to ε-neighborhood of point p and MinPts, we classify all points into three types
  - **Core points**: Given a point p and a non-negative integer MinPts, if the size of N(p) is at least MinPts, then p is said to be a **core point**.
  - **Border points:** Given a point p, p is said to be a **border point** if it is not a core point but N(p) contains at least one core point.
  - **Noise points**: Given a point p, p is said to be a **noise point** if it is neither a core point nor a border point.

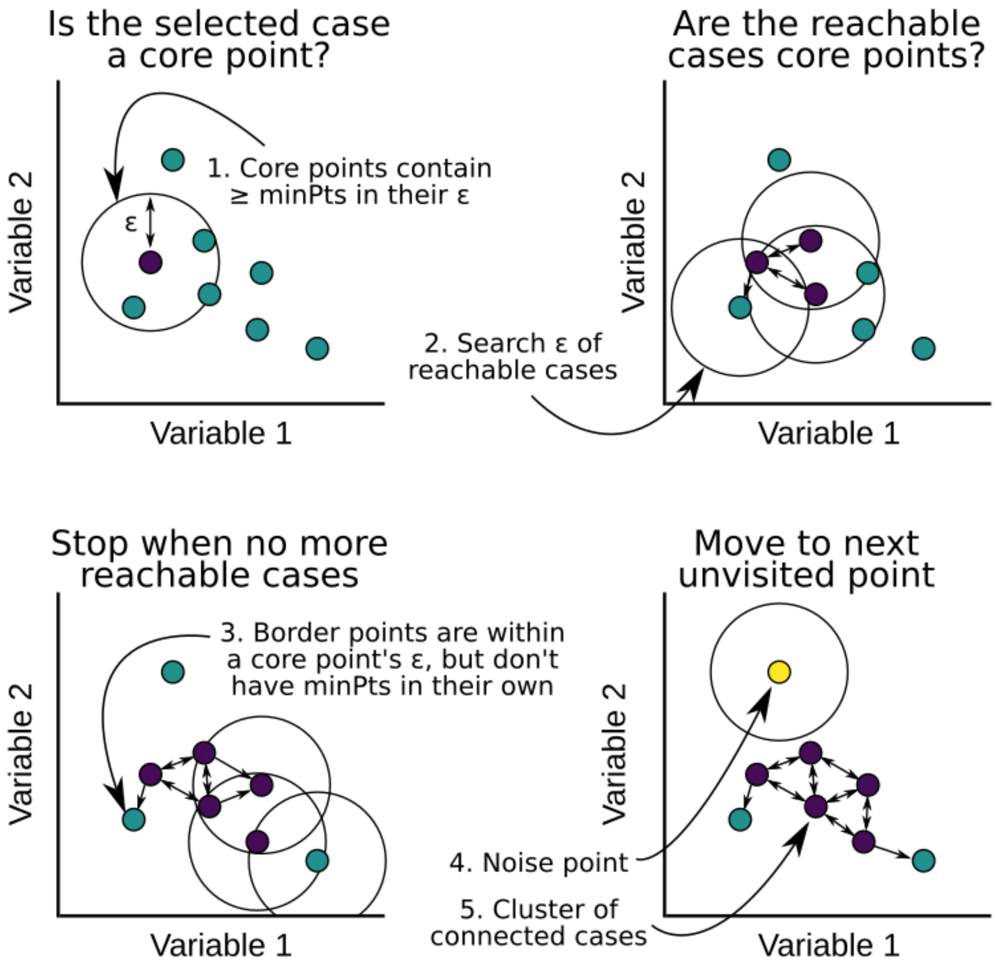


$\varepsilon = 1$

MinPts = 5

Lecture 6: Clustering II

- Examples of DBSCAN



Lecture 6: Clustering II

- If Epsilon is 2 and MinPts is 2, what are the clusters that DBSCAN would discover with the following 8 examples: A1=(2,10), A2=(2,5), A3=(8,4), A4=(5,8), A5=(7,5), A6=(6,4), A7=(1,2), A8=(4,9)? Please illustrate the steps for the DBSCAN clustering. (8points)
- Draw the 10 by 10 space and illustrate the discovered clusters. (3points)
- What if Epsilon is increased to sqrt(10)? (4points)

Lecture 6: Clustering II

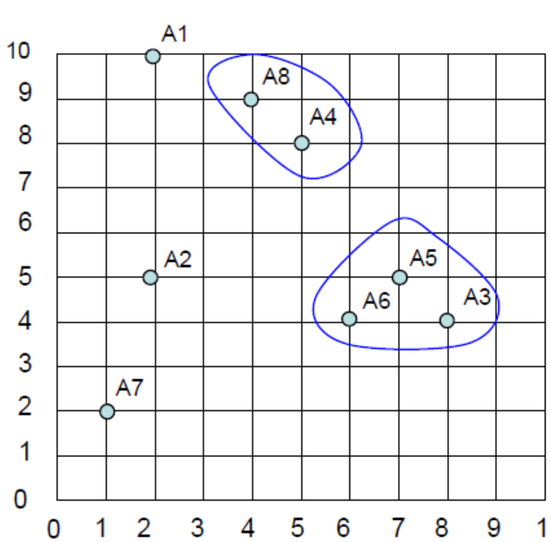
Solution:

What is the Epsilon neighborhood of each point?

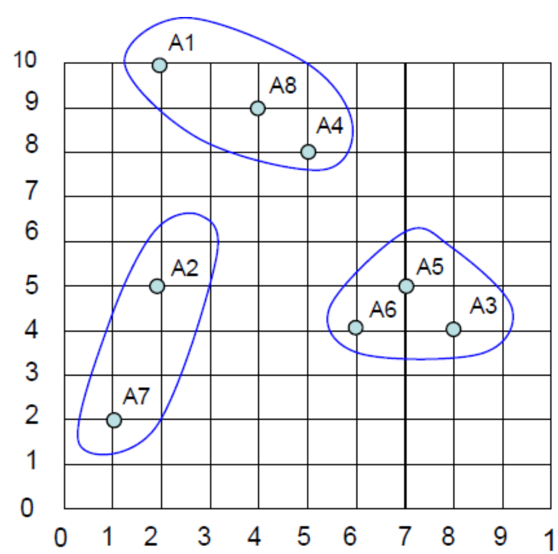
$N_2(A1)=\{\}$; $N_2(A2)=\{\}$; $N_2(A3)=\{A5, A6\}$; $N_2(A4)=\{A8\}$; $N_2(A5)=\{A3, A6\}$;
$N_2(A6)=\{A3, A5\}$; $N_2(A7)=\{\}$; $N_2(A8)=\{A4\}$

So A1, A2, and A7 are outliers, while we have two clusters C1={A4, A8} and C2={A3, A5, A6}

If Epsilon is $\sqrt{10}$ then the neighborhood of some points will increase:
A1 would join the cluster C1 and A2 would joint with A7 to form cluster C3={A2, A7}.



Epsilon = 2



Epsilon = $\sqrt{10}$

Lecture 6: Clustering II

- Clustering methods discussed so far
  - Every data object is assigned to exactly one cluster
- Some applications may need for fuzzy or soft cluster assignment
  - Ex. An e-game could belong to both entertainment and software
- Fuzzy cluster: A fuzzy set *S*: $F_S : X \rightarrow [0, 1]$ (value between 0-1)
- **The (EM) algorithm:** A framework to approach maximum likelihood or maximum a posteriori estimates of parameters in statistical models.
  - **E-step** assigns objects to clusters according to the current fuzzy clustering or parameters of probabilistic clusters
  - **M-step** finds the new clustering or parameters that maximize the sum of squared error (SSE) or the expected likelihood

Lecture 6: Clustering II

- Fuzzy C-means Objective function:

$$SSE = \sum_{j=1}^{k}\sum_{i=1}^{m} w_{ij}^{p}\, dist(\boldsymbol{x}_i, \boldsymbol{c}_j)^2 \qquad\qquad \sum_{j=1}^{k} w_{ij} = 1$$

- Initialization: choose the weights w_{ij} randomly

- Repeat:
  - Update centroids: $\boldsymbol{c}_j = \sum_{i=1}^{m} w_{ij}\boldsymbol{x}_i \,/ \sum_{i=1}^{m} w_{ij}$
  - Update weights:

$$w_{ij} = (1/dist(\boldsymbol{x}_i, \boldsymbol{c}_j)^2)^{\frac{1}{p-1}} / \sum_{j=1}^{k} (1/dist(\boldsymbol{x}_i, \boldsymbol{c}_j)^2)^{\frac{1}{p-1}}$$

Lecture 6: Clustering II

- Let x=[2,3,4,5,6,7,8,9,10,11], We choose the initial cluster center, c1=3, c2=11, please illustrate the steps to calculate cluster centers for the fuzzing clustering.

$$w_{i1}=\frac{\frac{1}{dist(o_i,c_1)^2}}{\frac{1}{dist(o_i\ ,c_1\ )^2}+\frac{1}{dist(o_i\ ,c_2\ )^2}}=\frac{dist(o_i,c_2)^2}{dist(o_i,c_2)^2+dist(o_i,c_1)^2}$$

$$w_{i2}=\frac{\frac{1}{dist(o_i,c_2)^2}}{\frac{1}{dist(o_i\ ,c_1\ )^2}+\frac{1}{dist(o_i\ ,c_2\ )^2}}=\frac{dist(o_i,c_1)^2}{dist(o_i,c_2)^2+dist(o_i,c_1)^2}$$

- Step 1: assign objects to cluster: c1 and c2;
- Then we can draw the partition matrix

$For\ node\ 1$

$$w_{11} = \frac{(2-11)^2}{(2-11)^2 + (2-3)^2} = \frac{81}{82} = 0.9878$$

$$w_{12} = \frac{(2-3)^2}{(2-3)^2 + (2-11)^2} = \frac{1}{82} = 0.0122$$

$$or \quad w_{12} = 1 - w_{11}$$

$For\ node\ 2$

$$w_{21} = \frac{(3-11)^2}{(3-11)^2 + (3-3)^2} = 1$$

$$w_{22} = \frac{(3-3)^2}{(3-11)^2 + (3-3)^2} = 0$$

$For\ node\ 3$

$$w_{31} = \frac{(4-11)^2}{(4-11)^2 + (4-3)^2} = \frac{49}{50} = 0.98$$

$$w_{32} = \frac{(4-3)^2}{(4-11)^2 + (4-3)^2} = \frac{1}{50} = 0.02$$

$$M = \begin{bmatrix} 0.9878 & 0.0122 \\ 1 & 0 \\ 0.98 & 0.02 \\ 0.9 & 0.1 \\ 0.7353 & 0.2647 \\ 0.5 & 0.5 \\ 0.2647 & 0.7353 \\ 0.1 & 0.9 \\ 0.02 & 0.98 \\ 0 & 1 \end{bmatrix}$$

Lecture 6: Clustering II

- Then recalculate the centroids according to the partition matrix

$$c1 = \frac{(0.9878)^2 \times 2 + 1^2 \times 3 + 0.98^2 \times 4 + 0.9^2 \times 5 + 0.7353^2 \times 6 + 0.5^2 \times 7 + 0.2647^2 \times 8 + 0.1^2 \times 9 + 0.02^2 \times 10 + 0 \times 11}{(0.9878)^2 + 1^2 + 0.98^2 + 0.9^2 + 0.7353^2 + 0.5^2 + 0.2647^2 + 0.1^2 + 0.02^2 + 0}$$
$$= 4.0049$$
$$c2 = \frac{(0.0122)^2 \times 2 + 0^2 \times 3 + 0.02^2 \times 4 + 0.1^2 \times 5 + 0.2647^2 \times 6 + 0.5^2 \times 7 + 0.7353^2 \times 8 + 0.9^2 \times 9 + 0.98^2 \times 10 + 1^2 \times 11}{(0.0122)^2 + 0^2 + 0.02^2 + 0.1^2 + 0.2647^2 + 0.5^2 + 0.7353^2 + 0.9^2 + 0.98^2 + 1^2}$$
$$= 9.4576$$

- Step 2: assign objects to cluster: c1 and c2;

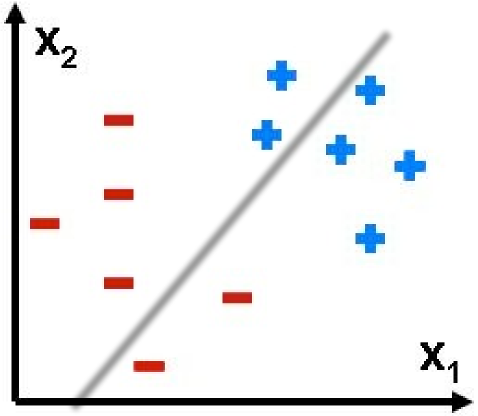
$$M = \begin{bmatrix} 0.9326 & 0.0674 \\ 0.9764 & 0.0236 \\ 1 & 8.0610e-07 \\ 0.9525 & 0.0475 \\ 0.7502 & 0.2698 \\ 0.4024 & 0.5976 \\ 0.1175 & 0.8825 \\ 0.0083 & 0.9917 \\ 0.0081 & 0.9919 \\ 0.0464 & 0.9536 \end{bmatrix}$$

- Then recalculate the centroids according to the partition matrix

$$c1 = 3.9769$$
$$c2 = 9.2641$$

Lecture 9: Linear Classifier

- For starters, let's assume that the training data is in fact perfectly linearly separable.
- Perceptron: iterative.
  - The strategy is to start with a random guess at the weights **w**, and to then iteratively change the weights to move the hyperplane in a direction that lowers the classification error.



Lecture 9: Linear Classifier

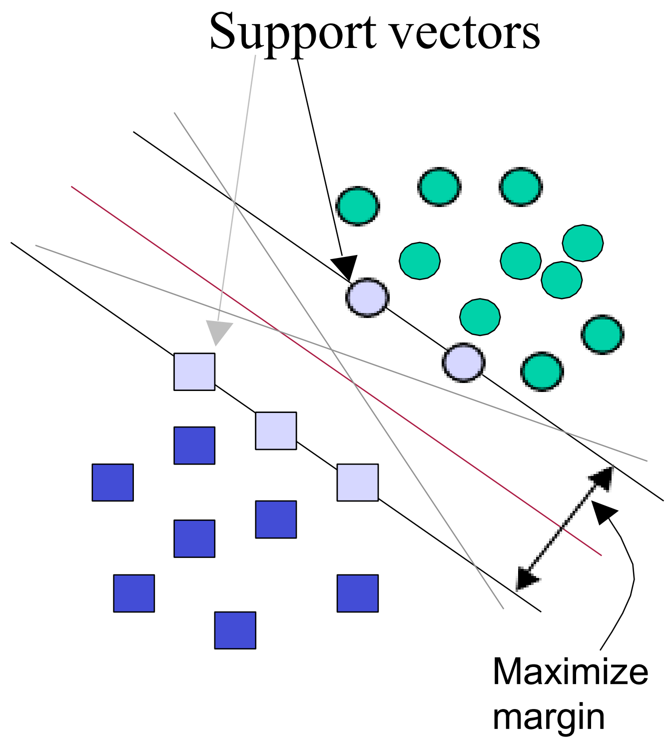
- Logistic regression:
  - introduces an extra non-linearity over a linear classifier, $f(x) = w^T x + b$, by using a logistic (or sigmoid) function, $\sigma()$.
  - The LR classifier is defined as

$$\sigma\left(f(\mathbf{x}_i)\right) \begin{cases} \geq 0.5 & y_i = +1 \\ < 0.5 & y_i = -1 \end{cases}$$

where $\sigma(f(\mathbf{x})) = \frac{1}{1+e^{-f(\mathbf{x})}}$

Lecture 9: Linear Classifier

- Linear SVM:
  - Unlike the Perceptron Algorithm, Support Vector Machines solve a problem that has a unique solution: they return the linear classifier with the maximum margin, that is, the hyperplane that separates the data and is farthest from any of the training vectors.



Lecture 9: Linear Classifier

Consider the following training data:

$$x_1 = (2, 3), x_2 = (3, 2), x_3 = (4, 4), x_4 = (4, 2)$$

$$x_5 = (6, 4), x_6 = (6, 3), x_7 = (7, 2), x_8 = (8, 3)$$

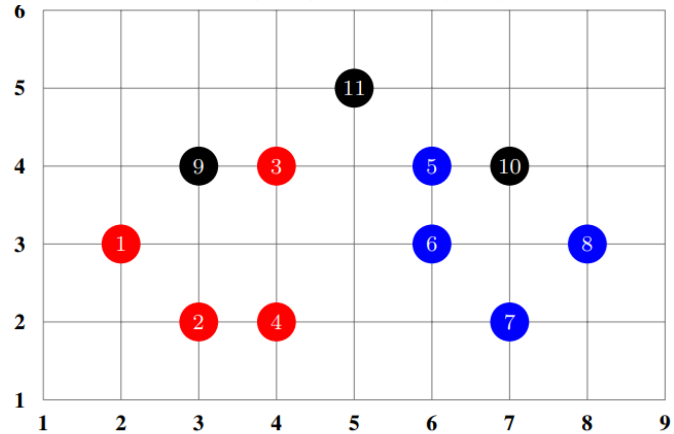
Let $y_A = -1, y_B = +1$ be the class indicators for both classes

$$A = \{x_1, x_2, x_3, x_4\}, B = \{x_5, x_6, x_7, x_8\}.$$

(a) Just using the above-standing plot, specify which of the points should be identified as support vectors.

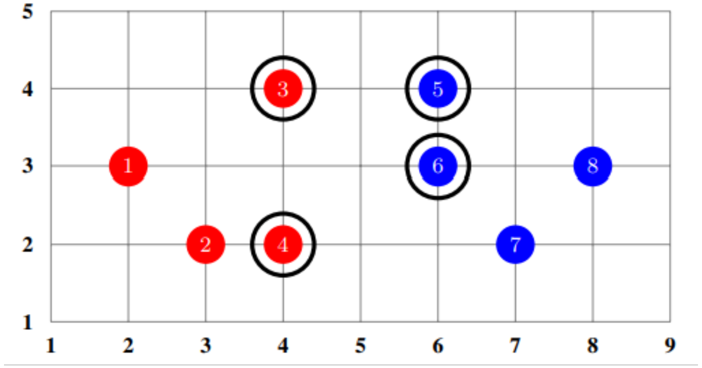
(b) Draw the maximum margin line which separates the classes (you don't have to do any computations here). Write down the normalized normal vector $\mathbf{w} \in \mathbb{R}^2$ of the separating line and the offset parameter $b \in \mathbb{R}$.

(c) Consider the decision rule: $H(x) = \langle \mathbf{w}, x \rangle + b$. Explain how this equation classifies points on either side of a line. Determine the class for the points $x_9 = (3, 4)$, $x_{10} = (7, 4)$ and $x_{11} = (5, 5)$.

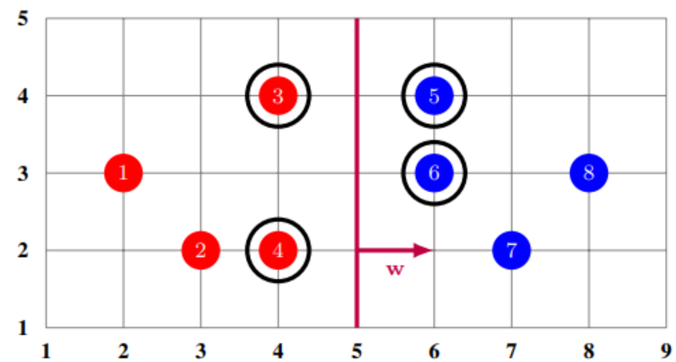


Lecture 9: Linear Classifier

- (a) The points {x3, x4, x5, x6} are chosen as support vectors.



- (b) We obtain w = $(1, 0)^T$, and b = −5.



Lecture 9: Linear Classifier

(c) Consider the decision rule: $H(x) = \langle \mathbf{w}, x \rangle + b$. Explain how this equation classifies points on either side of a line. Determine the class for the points $x_9 = (3, 4)$, $x_{10} = (7, 4)$ and $x_{11} = (5, 5)$.

We have the following decision rule:

$$H(x) = sign\left(\left\langle \begin{pmatrix} 1 \\ 0 \end{pmatrix}, x \right\rangle - 5\right)$$

and hence,

$$H\left(\begin{pmatrix} 3 \\ 4 \end{pmatrix}\right) = sign\left(\left\langle \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 3 \\ 4 \end{pmatrix} \right\rangle - 5\right) = sign(3-5) = sign(-2) = -1,$$

i.e. point x_9 is classified as belonging to class A (red).

$$H\left(\begin{pmatrix} 7 \\ 4 \end{pmatrix}\right) = sign\left(\left\langle \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 7 \\ 4 \end{pmatrix} \right\rangle - 5\right) = sign(7-5) = sign(2) = 1,$$

i.e. point x_{10} is classified as belonging to class B (blue).

$$H\left(\begin{pmatrix} 5 \\ 5 \end{pmatrix}\right) = sign\left(\left\langle \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 5 \\ 5 \end{pmatrix} \right\rangle - 5\right) = sign(5-5) = sign(0) = 0,$$

i.e. point x_{11} lies exactly on the decision boundary.

Lecture 9: Linear Classifier

- Evaluation for Classification/Imbalanced Issues
- Confusion Matrix:

	PREDICTED CLASS	
	Yes	**No**
ACTUAL CLASS **Yes**	TP	FN
ACTUAL CLASS **No**	FP	TN

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN}$$

$$ErrorRate = 1 - accuracy$$

$$Precision = Positive\ Predictive\ Value = \frac{TP}{TP + FP}$$

$$Recall = Sensitivity = TP\ Rate = \frac{TP}{TP + FN}$$

$$Specificity = TN\ Rate = \frac{TN}{TN + FP}$$

$$FP\ Rate = \alpha = \frac{FP}{TN + FP} = 1 - specificity$$

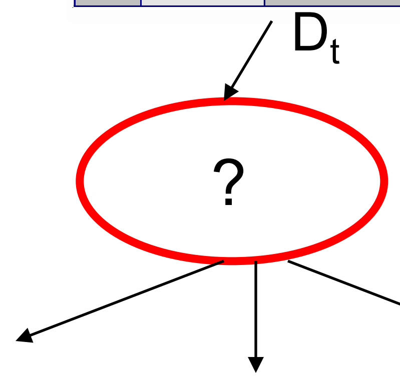
$$FN\ Rate = \beta = \frac{FN}{FN + TP} = 1 - sensitivity$$

$$Power = sensitivity = 1 - \beta$$

Lecture 10: Decision Tree

- Hunt's Algorithm
- Let Dt be the set of training records that reach a node t
- General Procedure:
  - If D_t contains records that belong the same class y_t, then t is a leaf node labeled as y_t
  - If D_t contains records that belong to more than one class, use an attribute to split the data into smaller subsets. Recursively apply the procedure to each subset.

ID	Home Owner	Marital Status	Annual Income	Defaulted Borrower
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



e.g. D_t contains all records "not home owner"

Lecture 10: Decision Tree

- How to determine the Best Split
  - Measures of Node Impurity
  - Gini index
  - Entropy
  - Misclassification error
- For 2-class problem (p, 1 – p)
  - $GINI = 1 – p^2 – (1 – p)^2 = 2p (1-p)$
  - $Entropy = – p\log p – (1 – p) \log(1-p)$
  - $Error=1-\max(p,(1-p))$

Lecture 10: Decision Tree

- Finding the Best Split
- Compute impurity measure (P) before splitting
- Compute impurity measure (M) after splitting
  - Compute impurity measure of each child node
  - M is the weighted impurity of children
- Choose the attribute test condition that produces the highest gain

$$\text{Gain} = P - M$$

or equivalently, lowest impurity measure after splitting (M)

Lecture 10 : Decision Tree

- Binary Attributes
- Categorical Attributes
- Continuous Attributes (using gini as example)
  - for each attribute, sort the attribute on values
  - Linearly scan these values, each time updating the count matrix and computing gini index
  - Choose the split position that has the least gini index

Cheat	No	No	No	Yes	Yes	Yes	No	No	No	No
	Annual Income									
Sorted Values →	60	70	75	85	90	95	100	120	125	220

Lecture 10 : Decision Tree

- Continuous Attributes (using gini as example)
  - for each attribute, sort the attribute on values
  - Linearly scan these values, each time updating the count matrix and computing gini index
  - Choose the split position that has the least gini index

	Cheat	No	No	No	Yes	Yes	Yes	No	No	No	No
		Annual Income									
Sorted Values		60	70	75	85	90	95	100	120	125	220
Split Positions	55	65	72	80	87	92	97	110	122	172	230
	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >

Lecture 10 : Decision Tree

- Continuous Attributes (using gini as example)
  - for each attribute, sort the attribute on values
  - Linearly scan these values, each time updating the count matrix and computing gini index
  - Choose the split position that has the least gini index

Cheat	No	No	No	Yes	Yes	Yes	No	No	No	No
	Annual Income									
Sorted Values →	60	70	75	85	90	95	100	120	125	220

Split Positions →	55		65		72		80		87		92		97		110		122		172		230	
	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>
Yes							0	3														
No							3	4														
Gini							0.343															

Gini(N1) = $1 - (0/3)^2 - (3/3)^2 = 0$
Gini(N2) = $1 - (3/7)^2 - (4/7)^2 = 0.4898$

Gini(Children) = 3/10*0+ 7/10 * 0.4898
= 0.343

Lecture 10 : Decision Tree

- Continuous Attributes (using gini as example)
  - for each attribute, sort the attribute on values
  - Linearly scan these values, each time updating the count matrix and computing gini index
  - Choose the split position that has the least gini index

Sorted Values →

Cheat	No	No	No	Yes	Yes	Yes	No	No	No	No
Annual Income										
	60	70	75	85	90	95	100	120	125	220

Split Positions →

	55		65		72		80		87		92		97		110		122		172		230	
	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>
Yes							0	3	1	2												
No							3	4	3	4												
Gini							0.343		0.417													

Lecture 10 : Decision Tree

- Continuous Attributes (using gini as example)
  - for each attribute, sort the attribute on values
  - Linearly scan these values, each time updating the count matrix and computing gini index
  - Choose the split position that has the least gini index

Sorted Values →

Cheat	No	No	No	Yes	Yes	Yes	No	No	No	No
Annual Income	60	70	75	85	90	95	100	120	125	220

Split Positions →

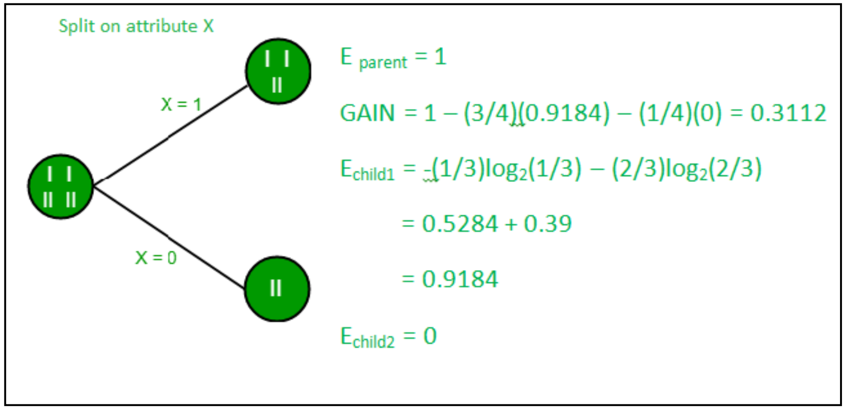
	55		65		72		80		87		92		97		110		122		172		230	
	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>
Yes	0	3	0	3	0	3	0	3	1	2	2	1	3	0	3	0	3	0	3	0	3	0
No	0	7	1	6	2	5	3	4	3	4	3	4	3	4	4	3	5	2	6	1	7	0
Gini	0.420		0.400		0.375		0.343		0.417		0.400		***0.300***		0.343		0.375		0.400		0.420	

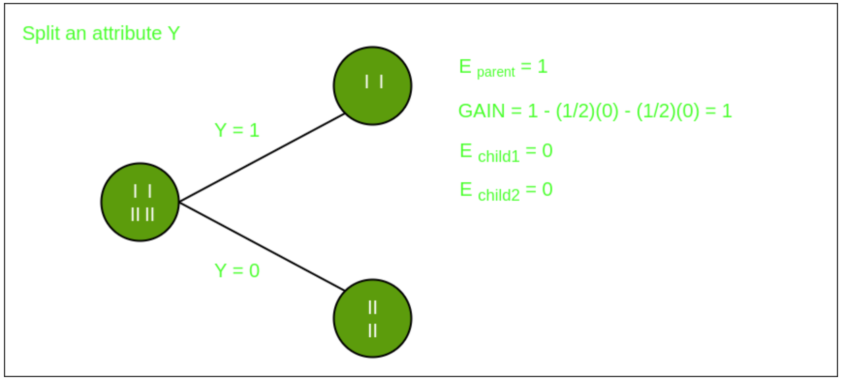
Lecture 10: Decision Tree

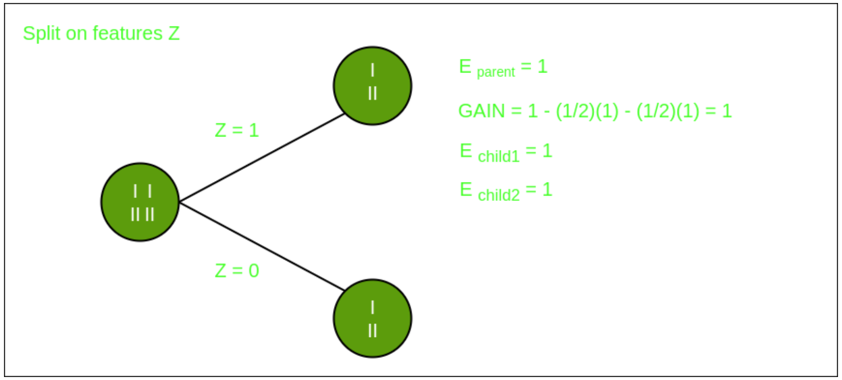
- Example: lets draw a Decision Tree for the following data using Information gain. Training set: 3 features and 2 classes, please identify the root points

X	Y	Z	C
1	1	1	I
1	1	0	I
0	0	1	II
1	0	0	II

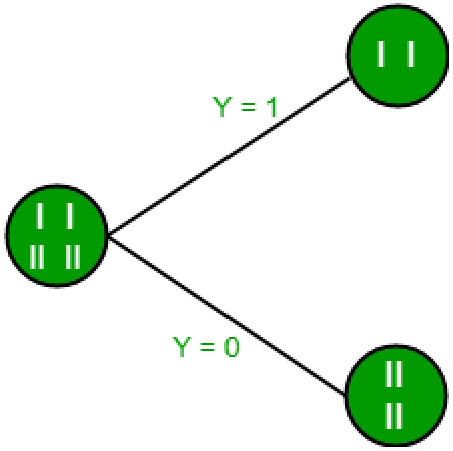
Lecture 10: Decision Tree





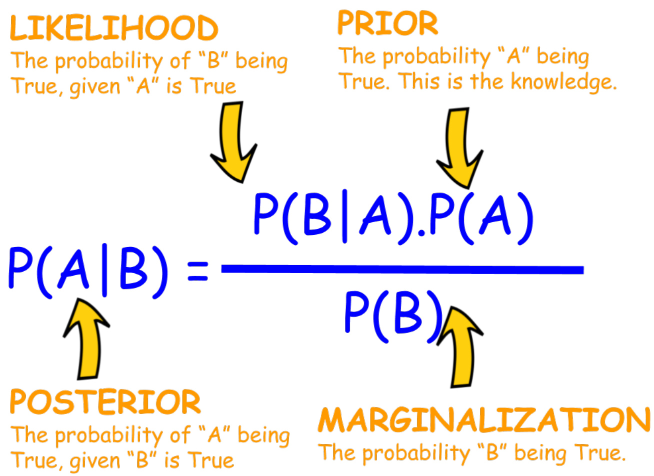


- From the above images we can see that the information gain is maximum when we make a split on feature Y. So, for the root node best suited feature is feature Y.
- Now we can see that while splitting the dataset by feature Y, the child contains pure subset of the target variable. So we don't need to further split the dataset.



Lecture 11: Bayes Classifier

- A probabilistic framework for solving classification problems



- Conditional Probability:

$$P(Y \mid X) = \frac{P(X,Y)}{P(X)}$$

$$P(X \mid Y) = \frac{P(X,Y)}{P(Y)}$$

- Bayes theorem:

$$P(Y \mid X) = \frac{P(X \mid Y)P(Y)}{P(X)}$$

Lecture 11: Bayes Classifier

- Example
- Consider the data set shown in following Table, Predict the class label for a test sample (A = 0, B = 1, C = 0) using the naive Bayes approach.

Record	A	B	C	Class
1	0	0	0	+
2	1	0	0	-
3	1	1	0	-
4	1	1	0	-
5	1	0	0	+
6	1	0	1	+
7	1	0	1	-
8	1	0	1	-
9	1	1	1	+
10	1	0	1	+

$P(+ \mid A=0, B=1, C=0)$

$$= \frac{P(A=0, B=1, C=0 \mid +) \cdot P(+)}{\underbrace{P(A=0, B=1, C=0)}_{X}} = \frac{P(A=0 \mid +)\, P(B=1 \mid +)\, P(C=0 \mid +)\, \boxed{P(+)}}{P(A=0, B=1, C=0)}$$

$$= \frac{\frac{1}{5} \cdot \frac{1}{5} \cdot \frac{2}{5} \cdot \frac{1}{2}}{P(A=0, B=1, C=0)}$$

$P(- \mid A=0, B=1, C=0)$

$$= \frac{P(A=0, B=1, C=0 \mid -)\, P(-)}{P(X)} = \frac{\overset{=0}{P(A=0 \mid -)}\, \overset{\frac{2}{5}}{P(B=1 \mid -)}\, \overset{\frac{3}{5}}{P(C=0 \mid -)}\, \boxed{P(-)}}{P(X)}$$

$$= 0$$

So classification = "+"

Lecture 11: Bayes Classifier

- Issues with Naïve Bayes Classifier
  - If one of the conditional probabilities is zero, then the entire expression becomes zero
  - Need to use other estimates of conditional probabilities than simple fractions
  - Probability estimation:

$$\text{Original}: P(A_i \mid C) = \frac{N_{ic}}{N_c}$$

$$\text{Laplace}: P(A_i \mid C) = \frac{N_{ic} + 1}{N_c + c}$$

$$\text{m - estimate}: P(A_i \mid C) = \frac{N_{ic} + mp}{N_c + m}$$

c: number of classes

p: prior probability of the class

m: parameter

N_c: number of instances in the class

N_{ic}: number of instances having attribute value A_i in class c

Lecture 11: Bayes Classifier

Consider the table with Tid = 7 deleted

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**
8	No	Single	85K	**Yes**
9	No	Married	75K	**No**
10	No	Single	90K	**Yes**

For class Yes, let m=3, p=1/3.

mp=1

For class No, let m=3, p=2/3.

mp=2

Given X=(Refund=Yes, Divorced, 120k)

P(X|NO)=(2+2)/(6+3) * 2/(6+3) *...>0

P(X|Yes)=1/(3+3) * (1+1)/(3+3) *...>0

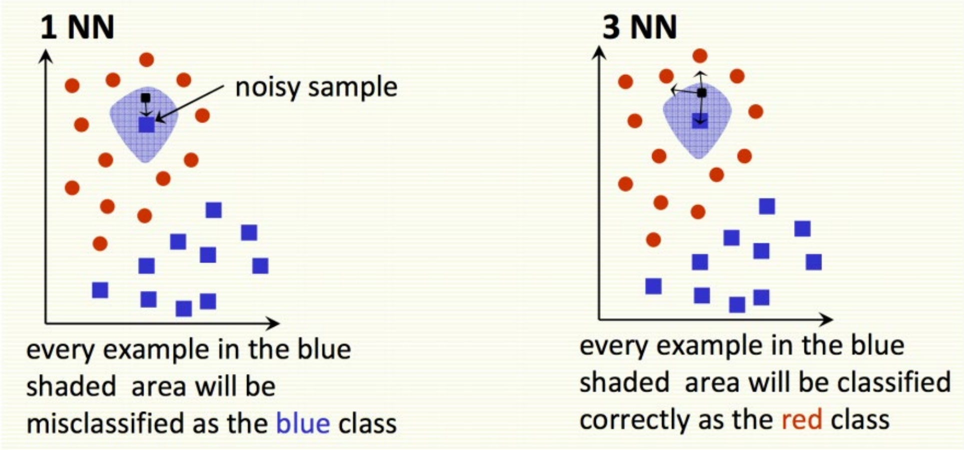
Given X = (Refund = Yes, Divorced, 120K)

P(X | No) = 2/6 X 0 X 0.0083 = 0

P(X | Yes) = 0 X 1/3 X 1.2 X 10^{-9} = 0

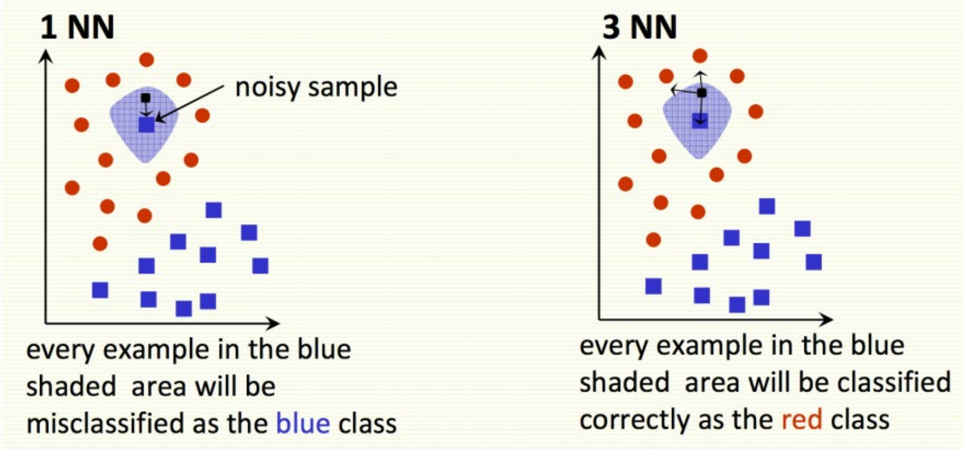
Lecture 11: KNN

- Nearest neighbor classifier are based on learning by analogy, that is, by comparing a giving test tuple with training tuples that are similar to it.
- The training tuples are described by n attributes.
- When K=1, the unknown tuple is assigned the class of the training tuple that is closest to it in pattern space.
- Larger K leads to stable results



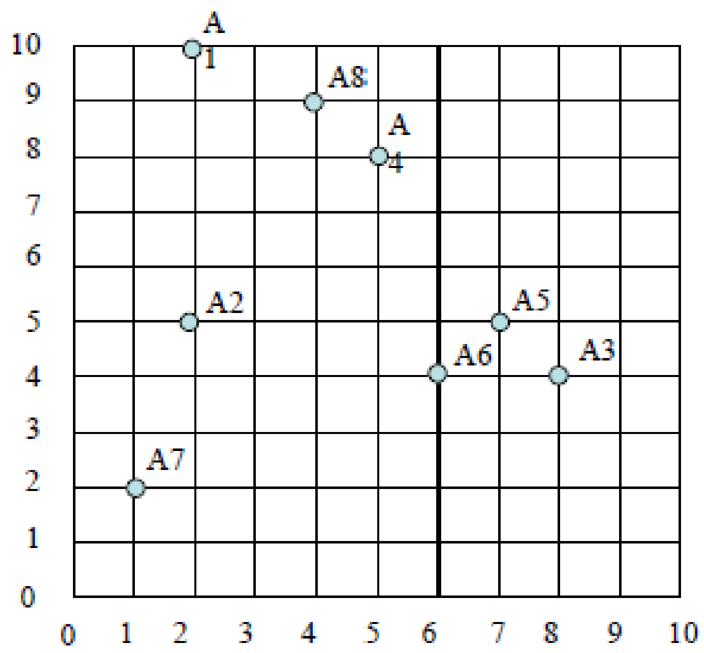
Lecture 11: KNN

- Nearest neighbor classifier are based on learning by analogy, that is, by comparing a giving test tuple with training tuples that are similar to it.
- The training tuples are described by n attributes.
- When K=1, the unknown tuple is assigned the class of the training tuple that is closest to it in pattern space.
- Larger K leads to stable results



Lecture 11: KNN

- Use the Nearest Neighbor clustering algorithm and Euclidean distance to cluster the examples from the previous exercise: A1=(2,10), A2=(2,5), A3=(8,4), A4=(5,8), A5=(7,5), A6=(6,4), A7=(1,2), A8=(4,9). Suppose that the threshold t is 4.



Lecture 11: KNN

- Use the Nearest Neighbor clustering algorithm and Euclidean distance to cluster the examples from the previous exercise: A1=(2,10), A2=(2,5), A3=(8,4), A4=(5,8), A5=(7,5), A6=(6,4), A7=(1,2), A8=(4,9). Suppose that the threshold t is 4.

A1 is placed in a cluster by itself, so we have K1={A1}.

We then look at A2 if it should be added to K1 or be placed in a new cluster.
d(A1,A2)= $\sqrt{25} = 5 > t$ ➔ K2={A2}

A3: we compare the distances from A3 to A1 and A2.
A3 is closer to A2 and d(A3,A2)=$\sqrt{36} > t$ ➔ K3={A3}

A4: We compare the distances from A4 to A1, A2 and A3.
A1 is the closest object and d(A4,A1)= $\sqrt{13} < t$ ➔ K1={A1, A4}

A5: We compare the distances from A5 to A1, A2, A3 and A4.
A3 is the closest object and d(A5,A3)=$\sqrt{2} < t$ ➔ K3={A3, A5}

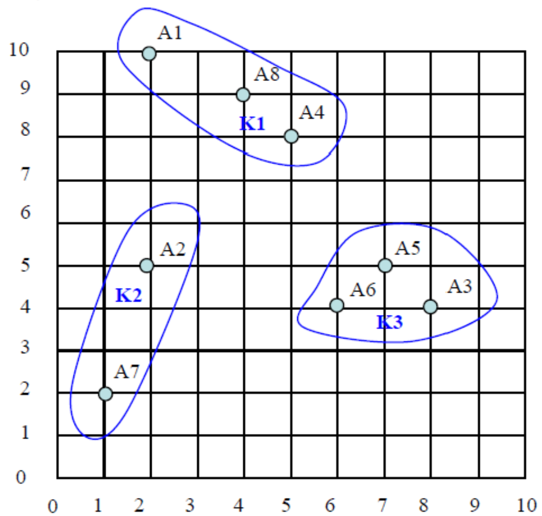
A6: We compare the distances from A6 to A1, A2, A3, A4 and A5.
A3 is the closest object and d(A6,A3)=$\sqrt{2} < t$ ➔ K3={A3, A5, A6}

A7: We compare the distances from A7 to A1, A2, A3, A4, A5, and A6.
A2 is the closest object and d(A7,A2)=$\sqrt{10} < t$ ➔ K2={A2, A7)

A8: We compare the distances from A8 to A1, A2, A3, A4, A5, A6 and A7.
A4 is the closest object and d(A8,A4)= $\sqrt{2} < t$ ➔ K1={A1, A4, A8)

Thus: K1={A1, A4, A8), K2={A2, A7), K3={A3, A5, A6)

Yes, it is the same result as with K-means.



Lecture 12: classification ensemble

- The goal of **ensemble methods** is to combine the predictions of several base estimators built with a given learning algorithm in order to improve generalizability / robustness over a single estimator.
- Bagging:
  - the driving principle is to build several estimators independently and then to average their predictions. On average, the combined estimator is usually better than any of the single base estimator because its variance is reduced.
- Boosting

Lecture 12: classification ensemble

- Boosting
  - An iterative procedure to adaptively change distribution of training data by focusing more on previously misclassified records
  - Initially, all N records are assigned equal weights (for being selected for training), weights may change at the end of each boosting round Records that are wrongly classified will have their weights increased in the next round, Records that are classified correctly will have their weights decreased in the next round