

EE4708 - Data Analytics Laboratory 2020

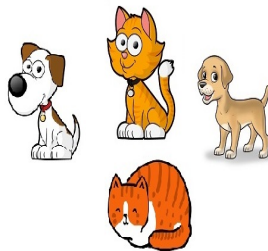
Week 5

Table of Contents

1. Classification
2. Nearest Neighbors Classifier
3. Naive Bayes Classifier

Classification

Features: Weight, Height, Tail Length,
Tounge Out, Face Shape,
Claws Size, Whiskers to face ratio.



Input

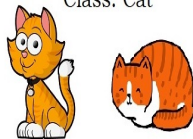
Training Data



Classifier



Class: Cat



Class: Dog



Output

Figure: Classification Example

Classification (cont.)

Definition:

It is the process of learning a predictive model that relates input features to discrete data classes or categories.

Key-points:

- ▶ This model can then be used to classify a new inputs to a particular class or category.
- ▶ If the outcomes are either positive or negative, then it is referred to as binary classification
- ▶ If the outputs can take more than two classes, then it is referred to as multi-class classification

Nearest Neighbours Classifier

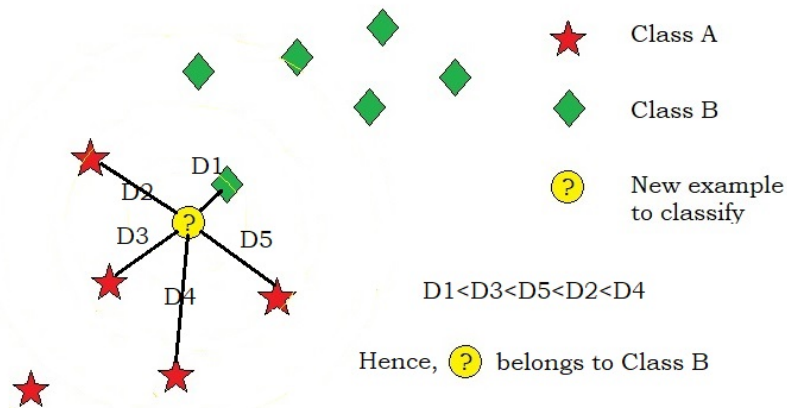


Figure: Example of Nearest Neighbours where D denotes distance

k-Nearest Neighbours Classifier

- ▶ Considering only the nearest neighbour is not effective in presence of noise
- ▶ Therefore, the K nearest neighbours are considered in finding the class of a sample
- ▶ kNN classifier works on the assumption that samples belonging to same class exist in close proximity
- ▶ kNN captures the idea of similarity or proximity using some distance metric, generally euclidean distance
- ▶ As the name suggests, the kNN classifier takes the k closest samples in the feature space and based on their labels, assigns a class to given input

k-Nearest Neighbours

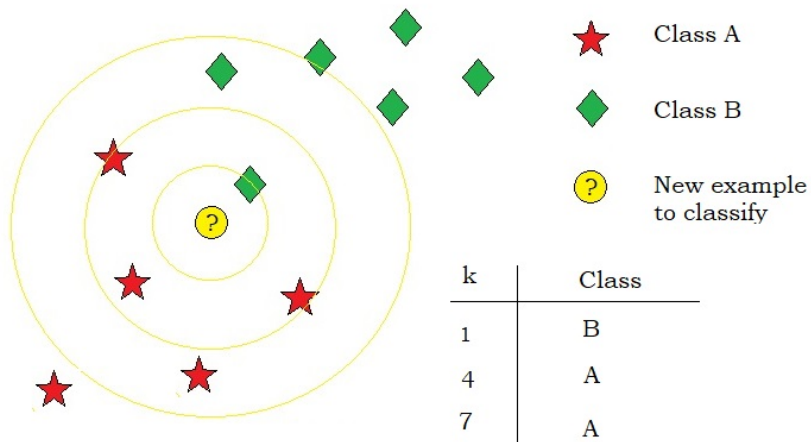


Figure: kNN Example

kNN: Algorithm Pseudo-code

1. Let x be the sample whose class is to be determined and x_1, x_2, \dots, x_n be the samples whose classes are known
2. Calculate $D(x, x_i) \quad \forall i = 1, 2, \dots, n$; where D denotes the Euclidean distance between the points.
3. Arrange the calculated n Euclidean distances in ascending order.
4. Let k be a chosen positive integer. Take the first k distances from the sorted list.
5. Find the k -samples corresponding to these k -distances.
6. Class of x is given by the majority class among these k -points

kNN: Choosing k value

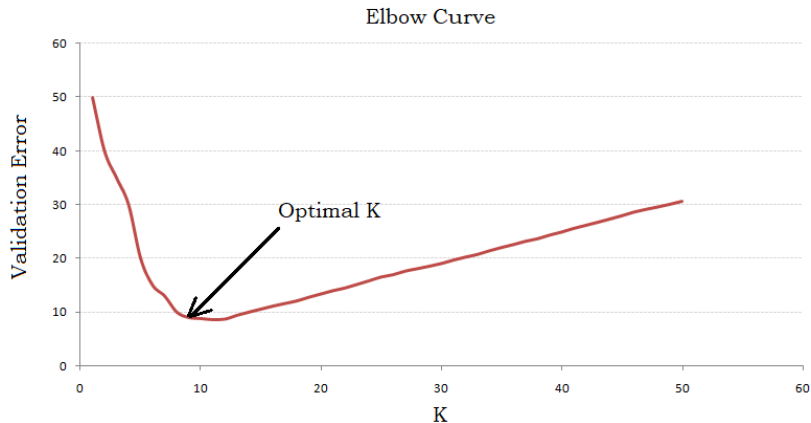




Figure: The elbow method showing the optimal K

kNN (cont.)

- ▶ There are no pre-defined statistical methods to find the most favorable value of K .
- ▶ A heuristic method called the elbow curve method can be used to determine an optimal value for K .
- ▶ The following steps describe the method:
 1. Initialize a random K value and start computing.
 2. Choosing a small value of K leads to unstable decision boundaries.
 3. The substantial K value is better for classification as it leads to smoothening the decision boundaries.
 4. Derive a plot between error rate and K denoting values in a defined range.
 5. When K increases, the centroids are closer to the clusters centroids. The improvements will decline, at some point rapidly, creating the elbow shape. The curve is called **elbow curve** and the point at elbow is the optimal value for K .

Naive Bayes Classifier

Normal (N) Message	Content (Frequency)	Probabilities	Prior Probabilities
	Dear (8) Friend (5) Food (3) Money (1)	$p(\text{Dear} \text{N})=8/17$ $p(\text{Friend} \text{N})=5/17$ $p(\text{Food} \text{N})=3/17$ $p(\text{Money} \text{N})=1/17$	$p(\text{N})=8/(8+4)$ $=8/12$
# N Messages = 8			
Spam (S) Message			
	Dear (2) Friend (1) Food (0) Money (4)	$p(\text{Dear} \text{S})=2/7$ $p(\text{Friend} \text{S})=1/7$ $p(\text{Food} \text{S})=0/7$ $p(\text{Money} \text{S})=4/7$	$p(\text{S})=4/(8+4)$ $=4/12$
# S Messages = 4			

Test Message: Dear Friend

$$p(\text{N}|\text{Test Message}) \propto p(\text{N}) \times p(\text{Dear}|\text{N}) \times p(\text{Friend}|\text{N})=320/3468=0.092$$

$$p(\text{S}|\text{Test Message}) \propto p(\text{S}) \times p(\text{Dear}|\text{S}) \times p(\text{Friend}|\text{S})=8/588=0.013$$

$$p(\text{N}|\text{Test Message}) > p(\text{S}|\text{Test Message})$$

Hence, the test message belongs to Normal Message.

Figure: Spam filtering using Naive Bayes

Naive Bayes Classifier

- ▶ Naive Bayes is a probabilistic classifier which is built using Baye's theorem
- ▶ It works based on the assumption (naive assumption) that all the features are independent of each other
- ▶ Suppose y represents the output labels and \mathbf{x} represents vector of input features, the posterior probability of a class given input features can be calculated using Baye's theorem as follows:

$$P(y|\mathbf{x}) = \frac{P(\mathbf{x}|y)P(y)}{P(\mathbf{x})}$$

- ▶ Based on the probability $P(y|\mathbf{x})$ and a chosen threshold, the class label of an input can be predicted

Naive Bayes Classifier

- ▶ In naive bayes's classifier, this posterior probability is calculated by obtaining the probabilities $P(\mathbf{x}|y)$, $P(y)$, $P(\mathbf{x})$ from data
- ▶ These calculations are done based on the naive assumption of feature independent
- ▶ If $\mathbf{x} = (x_1, x_2, \dots, x_d)$ are the d features, then feature independent implies:

$$P(y|x_1, x_2, \dots, x_d) = \frac{P(x_1, x_2, \dots, x_d|y)P(y)}{P(x_1, x_2, \dots, x_d)}$$
$$P(y|\mathbf{x}_1, x_2, \dots, x_d) = \frac{P(x_1|y)P(x_2|y) \dots P(x_d|y)P(y)}{P(x_1)P(x_2) \dots P(x_d)}$$

- ▶ The probabilities in the above formula can be calculated from the data

Naive Bayes (cont.)

Advantages:

1. They are extremely fast for both training and prediction.
2. They provide straightforward probabilistic prediction.
3. They are often very easily interpretable.
4. They have very few (if any) tunable parameters.

When to use it?

1. Though naive assumptions hold rarely in real datasets, this method works in practice as long as the features have low correlation among them.
2. When it is required to learn models quickly with large datasets and make fast predictions
3. Mostly used in text classification and spam filtering