Chapter - 7

Data classification -

It is a two step process.

(1) Learning step or training phase - where a classification algorithm builds the classifier by analyzing or learning from a training set made up of database tuples and their associated class labels.

The individual tuples making up the training set are referred to as training tuples.

Because the class label of each training tuple is provided, this step is also known as supervised learning

unsupervised learning → the class label of each training tuple is not known.

~~A test set~~

(2) Testing phase

A test set is used, made up of test tuples and their associated class labels. These tuples are randomly selected from the general data set. They are not used to construct the classifier.

The accuracy of a classifier on a given test set is the percentage of test set tuples that are correctly classified by the classifier.

Prediction

Suppose if we would like to predict how much a given customer will spend during a sale. This data analysis is an example of numeric prediction, where the model

constructed, predicts value. This model is a Predictor.

Regression analysis is a statistical methodology that is often used for numeric prediction.

Issues regarding classification and prediction

(1) Preparing the data for classification and prediction :-

The following preprocessing steps may be applied to the data to help improve the accuracy, efficiency and scalability of the classification or prediction process.

(a) Data cleaning → Remove or reduce noise and the treatment of missing values.

(b) Relevance analysis.

Many of the attributes in the data may be redundant. Correlation analysis can be used to identify whether any two given attributes are statistically related.

A database may also contain irrelevant attributes. Attribute subset selection can be used in these cases to find a reduced set of attributes.

(3) Data transformation and reduction

comparing classification and prediction methods

Classification and prediction methods can be compared and evaluated according to the following criteria:-

Accuracy

The accuracy of a classifier refers to the ability of a given classifier to correctly predict the class label of new or unseen data.

The accuracy of a predictor refers to how well a given predictor can guess the value of the predicted attribute for new or previously unseen data.

Speed → This refers to the computational costs involved in generating and using the given classifier or predictor.

Robustness → This is the ability of the classifier or predictor to make correct predictions given noisy data or data with missing values.

Scalability → This refers to the ability to construct the classifier or predictor efficiently given large amount of data.

Interpretability → This refers to the level of understanding and insight that is provided by the classifier or predictor.

K-NN

suppose each sample in our dataset has n attributes, which we combine to form an n-dimensional vector

$$X = (x_1, x_2, \cdots x_n)$$

These n attributes are considered to be independent variables. Each sample also has another attribute denoted by y (the dependent variable)

Suppose, a set of T such vectors are given together with their corresponding classes:

(i) (i) for $i = 1, 2, \cdots, T$
x, y

This set is referred to as the training set

Suppose we are given a new sample where $x = c$. we want to find the class of the new sample

The idea in K-Nearest Neighbor method is to identify K samples in the training whose independent variables are similar to c and to use these K samples to classify this new sample into a class, c.

calculate the Euclidean distance between the points x and c is.

$$d(x, c) = \sqrt{\sum_{i=1}^{n} (x_i - c_i)^2}$$

Let K=1, where we find the sample in the training set that is closest to c and set $v = y$, where y is the class of this nearest neighboring sample.

For K-NN, find the nearest K neighbors of x_q and then use a majority decision rule to classify the new sample.

For a given unlabeled example x_q find the K closest labeled examples in the training data set and assign x_q the class that appears most frequently within the K-subset.

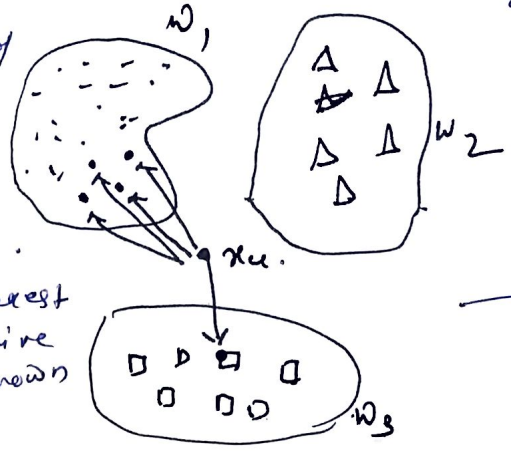
The K-NNR only requires

1. An integer K.
2. A set of labeled examples (training data)
3. A measure - distance measure to find the closeness.

Ex K-odd value.

To classify a new record

1) compute distance to other training records.
2) Identify K nearest neighbours.
3) use class labels of nearest neighbours to determine the class label of unknown record.



Requires

1) Training data.
2) Distance metric to compute distance between records.
3) The value of K.
   → the no. of nearest neighbours to retrieve from which to set majority class

1. We have three classes and the goal is to find a class label of the unknown example x_q.
2. Let us use Euclidean distance and value of $k = 5$ neighbours.
3. Of the 5 closest neighbors, 4 belong to w_1 and 1 belongs to w_3. So x_q is assigned to w_1.

Naive Bayesian classification (Bayesian classifier)

RID	age	income	student	credit rating	class: buys computer
1	youth	high	no	fair	no
2.	youth	high	no	excellent	no
3.	middle-aged	high	no.	fair	yes
4.	senior	medium	no	fair	yes
5.	senior	low	yes	fair	yes
6.	senior	low	yes	EX	no
7.	middle-aged	low	yes	EX	yes.
8.	youth	medium	no	fair	no
9.	youth	low	yes	fair	yes.
10.	senior	medium	yes	fair	yes.
11.	youth	medium	yes	EX	yes.
12.	middle-aged	medium	no	EX	yes.
13.	middle-aged	high	yes	fair	yes.
14.	senior	medium	no	EX	no

class labeled attribute → {yes or no}

Let C_1 = yes C_2 = NO.

The tuple we wish to classify is

$X = \{$ age = Y, income = medium, student = Y, credit-rating = fair$\}$

we need to maximize $P(X|C_i)P(C_i)$ for $i = 1, 2$.

$P(C_i)$ of each class can be computed based on the training tuples.

$P($buys-computer = yes$) = 9/14 = 0.643$

$P($buys-computer = NO$) = 5/14 = 0.357$

To compute $P(X|C_i)$, for $i = 1, 2$. we compu
the following conditional probabilities

$P(age = youth \mid buys\text{-}computer = yes) = \frac{2}{9} = 0.222$

$P(age = youth \mid buys\text{-}computer = No) = 3/5 = 0.600$

$P(income = medium \mid buys\text{-}computer = yes) = \frac{4}{9} = 0.444$

$P(income = medium \mid buys\text{-}computer = No) = \frac{2}{5} = 0.4$

$P(student = Y \mid buys\text{-}computer = yes) = \frac{6}{9} = 0.667$

$P(student = Y \mid buys\text{-}computer = No) = \frac{1}{5} = 0.2$

$P(credit\text{-}rating = fair \mid buys\text{-}computer = Y) = \frac{6}{} = .667$

$P(credit\text{-}rating = fair \mid buys\text{-}computer = N) = \frac{2}{5} = 0.4$

$P(X \mid buys\text{-}computer = yes)$

$= P(age = youth \mid buys\text{-}computer = yes) \times$
$P(income = Medium \mid buys\text{-}computer = yes) \times$
$P(student = yes \mid buys\text{-}computer = yes) \times$
$P(credit\_rating = fair \mid buys\text{-}computer = yes)$
$= 0.222 \times 0.444 \times 0.667 \times 0.667 = 0.044$

Similarly

$P(X \mid buys\text{-}computer = no) = 0.6 \times 0.4 \times 0.2 \times 0.4$
$= 0.019$

To find the class C_i, that maximize $P(X|C_i)P(C_i)$

$P(X \mid buy\text{-}computer = yes) P(buys\text{-}computer = yes)$
$= 0.044 \times 0.643 = 0.028$

$P(X \mid buys\text{-}computer = No) P(buys\text{-}computer = no)$
$= 0.019 \times 0.357 = 0.007$

Hence, the tuple X belongs to (yes) class.

Algorithm

1) Let D' be a training set of tuples and their associated class labels.
Each tuple is represented by an n-dimensional attribute vector, $X = \{x_1, x_2, \ldots x_n\}$ having n measurements made from n attributes respectively $A_1, A_2, \ldots A_n$.

2) Suppose that there are m classes, $C_1, C_2, \ldots C_m$. Given a tuple X, the classifier will predict that X belongs to the class having highest probability, ~~conditioned~~ conditioned on X.
The naive Bayesian classifier predicts that tuple X belongs to the class, C_i if and only if

$$P(C_i|X) > P(C_j|X) \text{ for } 1 \le j \le m, j \ne i$$

Thus we maximize $P(C_i|X)$. The class C_i for which $P(C_i|X)$ is maximized is called the maximum posteriori hypothesis.

By Bayes's theorem.

$$P(C_i|X) = \frac{P(X|C_i)\,P(C_i)}{P(X)}$$

3) As $P(X)$ is constant for all classes, only $P(X|C_i)\,P(C_i)$ need to be maximized. The class prior probability may be estimated by $P(C_i) = \frac{|C_{i,D}|}{|D|}$, where $|C_{i,D}|$ = the no. of training tuples of class C_i in D.

4) For evaluating $P(X|C_i)$, the assumption of class conditional independence is made.

Thus $$P(X|C_i) = \prod_{K=1}^{n} P(x_K|C_i)$$

$$= P(x_1|C_i) * P(x_2|C_i) * \ldots * P(x_n|C_i)$$

x_K = value of attribute A_K for tuple X.

(a) If A_K is categorical, then $P(x_K|C_i)$ is the no. of tuples of class C_i in D having the value x_K for A_K divided by $|C_i, D|$, the no. of tuples of class C_i in D.

(b) If A_K is continuous-valued, then

$$P(x_K|C_i) = g(x_K, \mu_{C_i}, \sigma_{C_i}) \quad \text{---(1)}$$

$$g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \quad \text{---(2)}$$

μ_{C_i} and σ_{C_i} → are the mean and standard deviation of the values of attribute A_K for training tuples of class C_i.

A continuous valued attribute is assumed to have a Gaussian distribution with a mean, μ and standard deviation, σ.

(5) In order to predict the class label of X $P(X|C_i)P(C_i)$ is evaluated for each class C_i. The classifier predicts that the class label of tuple X is the class C_i, if

$$P(X|C_i)P(C_i) > P(X|C_j)P(C_j) \text{ for } 1 \le j \le m, j \neq i$$