**Assignment No. 05**

**PRN: 2019BTECS00067**

**Name: Vaibhav Kute**

**Aim:**

Use / extend the data analysis tool (menu driven GUI) developed in Assignment No. 1

to perform the following classification task :

1. Design and implement the following classifiers:

a) Regression classifier.

b) Naïve Bayesian Classifier.

c) k-NN classifier (Take k = 1,3,5,7)

d) Three layer Artificial Neural Network (ANN) classifier (use back

propagation). Plot error graph (iteration vs error).

2. Tabulate the results in confusion matrix and evaluate the performance of above

classifier using following metrics :

a) Recognition rate

b) Misclassification rate

c) Sensitivity

d) Specificity

e) Precision & Recall

3. Use the following data sets from UCI machine learning repository :

a) IRIS

b) Breast Cancer

**Theory:**

**Regression**

Regression is a process of finding the correlations between dependent and independent variables. It helps in predicting the continuous variables such as prediction of Market Trends, prediction of House prices, etc.

The task of the Regression algorithm is to find the mapping function to map the input variable(x) to the continuous output variable(y).

Example: Suppose we want to do weather forecasting, so for this, we will use the Regression algorithm. In weather prediction, the model is trained on the past data, and once the training is completed, it can easily predict the weather for future days.

Types of Regression Algorithm:

* Simple Linear Regression
* Multiple Linear Regression
* Polynomial Regression
* Support Vector Regression
* Decision Tree Regression
* Random Forest Regression

**Bayes’ Theorem**

Bayes’ Theorem finds the probability of an event occurring given the probability of another event that has already occurred. Bayes’ theorem is stated mathematically as the following equation:



where A and B are events and P(B) ? 0.

* Basically, we are trying to find probability of event A, given the event B is true. Event B is also termed as **evidence**.
* P(A) is the **priori** of A (the prior probability, i.e. Probability of event before evidence is seen). The evidence is an attribute value of an unknown instance(here, it is event B).
* P(A|B) is a posteriori probability of B, i.e. probability of event after evidence is seen.

Now, with regards to our dataset, we can apply Bayes’ theorem in following way:



where, y is class variable and X is a dependent feature vector (of size *n*) where:



Just to clear, an example of a feature vector and corresponding class variable can be: (refer 1st row of dataset)

X = (Rainy, Hot, High, False)

y = No

So basically, P(y|X) here means, the probability of “Not playing golf” given that the weather conditions are “Rainy outlook”, “Temperature is hot”, “high humidity” and “no wind”.

**Naive assumption**

Now, its time to put a naive assumption to the Bayes’ theorem, which is, **independence** among the features. So now, we split **evidence** into the independent parts.

Now, if any two events A and B are independent, then,

P(A,B) = P(A)P(B)

Hence, we reach to the result:



which can be expressed as:



Now, as the denominator remains constant for a given input, we can remove that term:



Now, we need to create a classifier model. For this, we find the probability of given set of inputs for all possible values of the class variable *y* and pick up the output with maximum probability. This can be expressed mathematically as:



So, finally, we are left with the task of calculating P(y) and P(xi | y).

Please note that P(y) is also called **class probability** and P(xi | y) is called **conditional probability**.

The different naive Bayes classifiers differ mainly by the assumptions they make regarding the distribution of P(xi | y).

**K-Nearest Neighbor(KNN) Algorithm for Machine Learning**

K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.

K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.

K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.

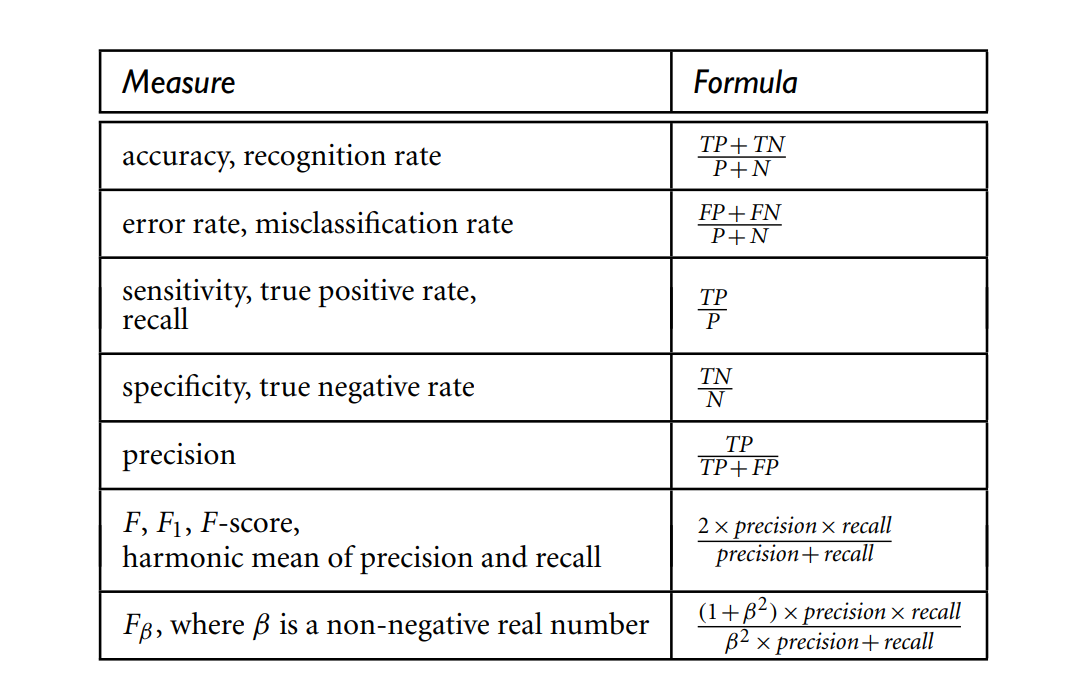
K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.

K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data.

It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.

KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.

Example: Suppose, we have an image of a creature that looks similar to cat and dog, but we want to know either it is a cat or dog. So for this identification, we can use the KNN algorithm, as it works on a similarity measure. Our KNN model will find the similar features of the new data set to the cats and dogs images and based on the most similar features it will put it in either cat or dog category.



Run commands:

Go to folder location of main.py file and run command :

streamlit run main.py

**Procedure:**

1. First divided dataset into x-train(70%) and x-test(30%).

2. Calculated Euclidian distance for x-train dataset.

3. Added column “distance” in dataset.

4. Sorted dataset by distance column.

5. Selected top k tuples for query.

**Results snapshots:**

Result snapshots are included in other pdf files.

**Conclusion:**

Using streamlit GUI and python coding all the tasks are implemented well.