Exercise 6

6.1 Problem Statement:

Implement k-NN algorithm to solve classification problem.

6.2 Description of Machine Learning Algorithm:

K-nearest neighbor or K-NN algorithm basically creates an imaginary boundary to classify the data. When new data points come in, the algorithm will try to predict that to the nearest of the boundary line.

Therefore, larger k value means smother curves of separation resulting in less complex models. Whereas, smaller k value tends to overfit the data and resulting in complex models.

It’s very important to have the right k-value when analyzing the dataset to avoid overfitting and underfitting of the dataset.

Using the k-nearest neighbor algorithm we fit the historical data (or train the model) and predict the future.

The k-nearest neighbor algorithm is from the scikit-learn package.

- Create feature and target variables.
- Split data into training and test data.
- Generate a k-NN model using neighbors value.
- Train or fit the data into the model.
- Predict the future.

6.3 Description of Data Set:

Title of the data set: Wine Dataset

The wine dataset is a classic and very easy multi-class classification dataset. The data in this dataset the result of a chemical analysis of wines grown in the same region in Italy by three different cultivators. There are thirteen different measurements taken for different constituents found in the three types of wine.

6.4 Data Preprocessing and Exploratory Data Analysis:

Data preprocessing is the process of transforming raw data into an understandable format. It is also an important step in data mining as we cannot work with raw data. The quality of the data should be checked before applying machine learning or data mining algorithms.

Major Tasks in Data Preprocessing:

1. Data cleaning
2. Data integration
3. Data reduction
4. Data transformation

Exploratory data analysis (EDA) is a technique that data professionals can use to understand a dataset before they start to model it. Some people refer to EDA as data exploration. The goal of conducting EDA is to determine the characteristics of the dataset. Conducting EDA can help data analysts make predictions and assumptions about data. Often, EDA involves data visualization, including creating graphs like histograms, scatter plots and box plots.

Major Tasks in EDA:

1. Observe your dataset

2. Find any missing values
3. Categorize your values
4. Find the shape of your dataset
5. Identify relationships in your dataset
6. Locate any outliers in your dataset

6.5 Machine Learning Package Used for Model building:

For classification of model we use scikit-learn

Scikit-learn is an open source Machine Learning Python package that offers functionality supporting supervised and unsupervised learning. Additionally, it provides tools for model development, selection and evaluation as well as many other utilities including data pre-processing functionality.

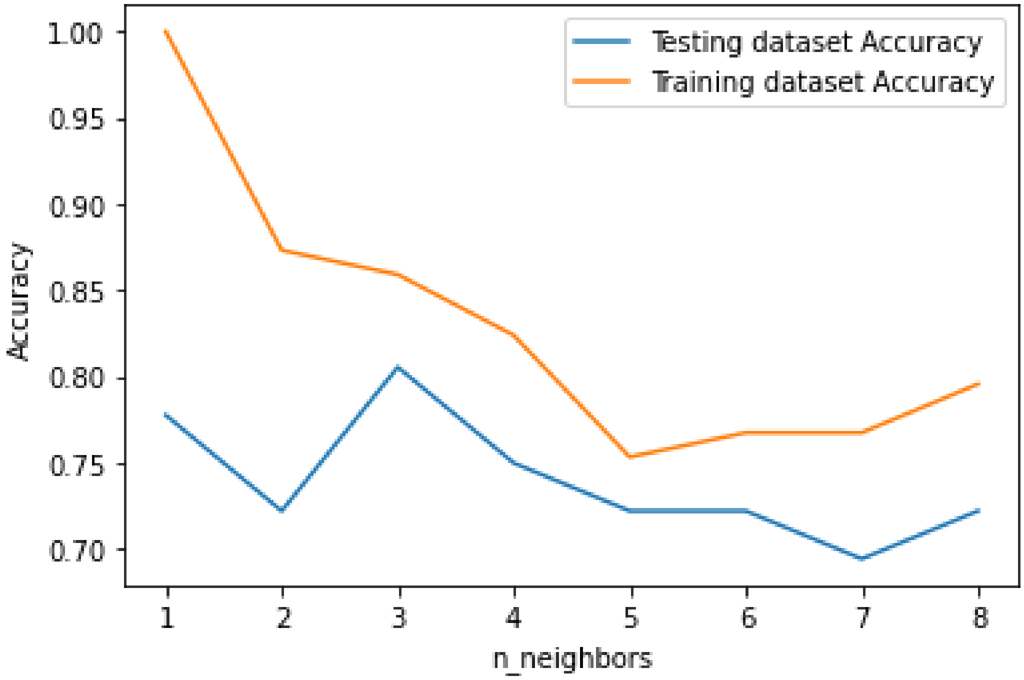
More specifically, scikit-learn’s main functionality includes classification, regression, clustering, dimensionality reduction, model selection and pre-processing. sThe library is very simple to use and most importantly efficient as it is built on **NumPy**, **SciPy** and **matplotlib.**

Neural networks are a machine learning method inspired by how the human brain works. They are particularly good at doing pattern recognition and classification tasks, often using images as inputs. They’re a well established machine learning technique that has been around since the 1950s but have gone through several iterations since that have overcome fundamental limitations of the previous one. The current state of the art neural networks are often referred to as deep learning.

6.6 Implementation:

```
 # Import necessary modules

from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import train_test_split
from sklearn.datasets import load_wine
import numpy as np
import matplotlib.pyplot as plt

wineData = load_wine()

# Create feature and target arrays
X = wineData.data
y = wineData.target

# Split into training and test set
X_train, X_test, y_train, y_test = train_test_split(
             X, y, test_size = 0.2, random_state=42)

neighbors = np.arange(1, 9)
train_accuracy = np.empty(len(neighbors))
test_accuracy = np.empty(len(neighbors))

# Loop over K values
```

```
for i, k in enumerate(neighbors):
    knn = KNeighborsClassifier(n_neighbors=k)
    knn.fit(X_train, y_train)

    # Compute training and test data accuracy
    train_accuracy[i] = knn.score(X_train, y_train)
    test_accuracy[i] = knn.score(X_test, y_test)

# Generate plot
plt.plot(neighbors, test_accuracy, label = 'Testing dataset Accuracy')
plt.plot(neighbors, train_accuracy, label = 'Training dataset Accuracy')

plt.legend()
plt.xlabel('n_neighbors')
plt.ylabel('Accuracy')
plt.show()
```



6.7 Results and Discussion:

The implementation of k-NN algorithm to solve classification problem using wine dataset has been done successfully.