Name: Smitesh Patil Class: 1CSD Student Id: 22223696

1. Have a look at the given data, understand the problem based on dependent variable and select a machine learning category that can solve the task/problem. Briefly explain why do you think it is the correct ML category for this problem?

**Answer**: Given data contains four independent variables, namely sepal length, sepal width, petal length, petal width. These variables determine value of our dependent variable “target/label”. Target/label contains two values setosa and virginica. Theses are two types of flowers, and the dependent variables are features that determine the type of the flower. In the given dataset, the target label is given; hence, it is a case of supervised learning problem. Moreover, the target label is a discrete variable as it contains two values setosa and virginica, hence it is a classification task.

1. Explore and report the data and its distribution among training and testing data. Can we call it imbalanced dataset, explain your answer (yes/no) briefly? [2 Mark]

**Answer:** Datasets that contain uneven distribution of target class variable are called imbalanced dataset.

A picture containing graphical user interface

Description automatically generated

In both the training and testing datasets, the target class variables have a even distribution of setosa and virginica observations. Thus, the dataset is not imbalanced.

1. Research and write down about open-source machine learning package that are freely available, and select one that you think will be good and easy for this task. Your report should include a short overview of the main features of the package you have chosen.

**Answer:** There are a plethora of open-source machine learning packages available for python free of cost. For this task we have used packages like NumPy a python package that is useful to transform your data in matrix form that later could be used for performing mathematics operations. Pandas a package that is used for data manipulation purposes. For machine learning tasks there are various libraries like Scikit-learn, TensorFlow, PyTorch and Keras. Scikit-learn is the chosen package for this task. Implementing machine learning models for small datasets is simple in Scikit-learn. TensorFlow, PyTorch and Keras are generally used to develop highly complex deep learning models.

Brief Overview of Sci-Kit-learn package:

Sci-Kit contains various sub packages that are meant to perform specific tasks

1. sklearn.preprocessing : this sub package contains various functions for cleaning and standardizing the dataset for machine learning task these include standardizing functions like StandardScaler, MinMaxScaler.
2. sklearn.model\_selection: It contains functions used to create train, test and validation sets like kfold, train\_train\_split.
3. sklearn.metrics : Metrics that are used to validate the machine learning models like accuracy score, confusion matrix, f1\_score are contained in this sub package.
4. Machine learning models: There are various sub packages that contain machine learning models like sklearn.linear\_model that contains models like Logistic Regression, Linear Regression also for models defined on Naïve Bayes there is a package sklearn.naive\_bayes it contains models like Multinomial, GaussianNB and CategoricalNB. Sklearn.neighbors contains models for Nearest Neighbours algorithms like Nearest Neighbours, KNearestClassifier, KNearestRegressor.
5. In order to use the dataset (Plant-dataset) supplied below, you might need to do some work to prepare it for input into the ML package, depending on the ML category requirements. Document any data preparation (e.g. normalisation) steps in your report. [2 marks max.]

**Answer:** Normalisation for the data is done using MinMaxScaler function from sklearn.preprocessing package. In Normalisation we transform our data between a certain range of values in this case it is between 0 and 1.

For the second model data standardisation is used for preparing data. In Standardisation we transform the data such that mean of the data samples is 0 and standard deviation is zero. In sci-kit learn standardisation can be done by using StandardScaler function from sklearn.preprocessing library.

1. From the ML package, select two different algorithms from the category you selected and apply to the dataset. In your report, include a clear description of both algorithms. Ensure that you acknowledge all of your sources of information. Report the results with and without normalisation of the data. [4 marks max.]

**Answer:** Two machine learning algorithms K-nearest-neighbour and logistic regression are used for this dataset. The task is of classification type and K-nearest-neighbour and logistic regression both are classification algorithms. Moreover, KNN is a lazy learner and logistic regression is an eager learner. During training phase, a lazy learner just stores the data without learning it and classification of data points are done during testing phase. On a contrary, eager learners learn during the training phase. Consequently, eager learners take more time on training compared to testing and lazy learners take more time in testing phase.

**K –** Nearest Neighbour on the dataset:

1. Apply standardisation on both testing and training data. StandardScaler transforms data such that mean of data = 0 and standard deviation = 1.

Text

Description automatically generated

1. Appy KNearestNeighbours on standardised dataset. In KNN first we need to set value of hyperparameters. We take the value of K = 3 and metric as Euclidean distance.
2. KNN thinks of every training case in training set as a datapoint in space. And during the testing phase it projects testing case in this hyperspace.
3. The distance of all the datapoints in the training set are calculated against testing case. As the value of K is 3, three nearest datapoints are considered.
4. The output of the target variable depends on the target variable value of these three nearest datapoint. Average of three labels is considered and assigned as output.
5. For example, if the target variable of three datapoints with the least Euclidean distance to the testing datapoint are [‘setosa’, ‘virginica’, ‘setosa’] then setosa would be assigned as target variable for the testing case.

Logistic Regression:

1. Apply normalisation on both testing and training data. StandardScaler transforms data such that range of variables are between range of 0 and 1.

Text

Description automatically generated

1. Logistic regression is a classification algorithm. In this example we are doing to use logistic regression to predict target label which belong to either of the following values [‘setosa’, ‘virginica’].
2. Just like in linear regression, we need to train a hypothesis in logistic regression. In this case hypothesis of logistic regression would be.

Here y is the target/label

are independent variables

And are slopes for the independent variables that need to be calculated

is the intercept.

1. Now in regression we try to minimize the error rate of the hypothesis i.e the value predicted by hypothesis and actual value y should be minimized denoted by cost function.
2. Minimizing the cost in done by using gradient descent algorithm. Which minimizes the cost while iterating over the training data. The output Y is right now a continuous value it can be converted to binary form by using sigmoid function. Sigmoid/Logistic function basically converts any input value to an output that is between range of 0 and 1.

Graphical user interface

Description automatically generated with medium confidence

After training is completed, the hypothesis is tested on unseen testing data to check the accuracy of the model trained.

1. Train and test your chosen algorithms using the training set provided in **plant-train.csv**. You should then test your trained models using the test set provided in **plant-test.csv**. Report on the results with appropriate performance metric e.g. accuracy that you consider best for each model on the training set and on the test set. Also include details of the classification models constructed – these may include graphics if appropriate. [ 6 marks max.]

**Answer**: Training data and testing data were normalised (for Logistic Regression) and standardised( for K-Nearest Neighbours). Now, the data needs to be trained on machine learning models for KNN and logistic regression from sckit-learn library.

Training on KNN and Logistic Regression

Hyper- Parameters for K-Nearest Neighbour: Hyperparameter: K = 3; distance measure = “Euclidean”.

Graphical user interface, text, application, email

Description automatically generatedGraphical user interface

Description automatically generated with low confidence

Performance Metrics for algorithms

For: K-Nearest Neighbour For: Logistic Regression

Graphical user interface, application

Description automatically generated with medium confidenceGraphical user interface, text, application

Description automatically generated with medium confidence

For both the algorithms we get all the test samples classified correctly, In the confusion matrix we observe that 10 cases were classified as true positive and 10 as true negative for both the algorithms.

1. Discuss in your report whether the two models give very similar or significantly different results, and why. [2 marks max.]

**Answer**:

Text

Description automatically generated with medium confidence

A picture containing graphical user interface

Description automatically generated

As you can see by runtime logistic regression is much faster in testing phase than K-Nearest Neighbour as KNN is a lazy learner it learns during the testing phase whereas in logistic regression hypothesis is formed during the training phase.

Although the time taken by KNN and Logistic Regression for generating output differs, they are performing well on testing data. Testing data is unseen, yet they both predict samples in testing dataset with high accuracy. This maybe because we have a small balanced dataset, that our algorithms are training on, if we try on large dataset our algorithms can learn intricacies of the problem statement better.