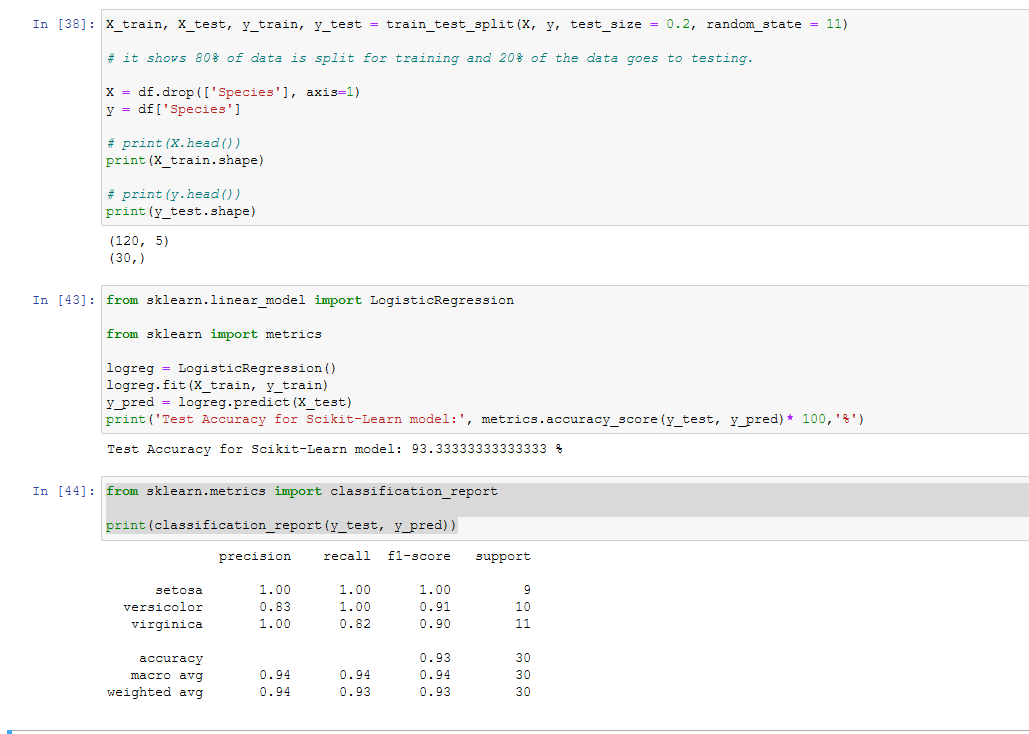
**V.SURESH KUMAR**

**Assignment No: 03**

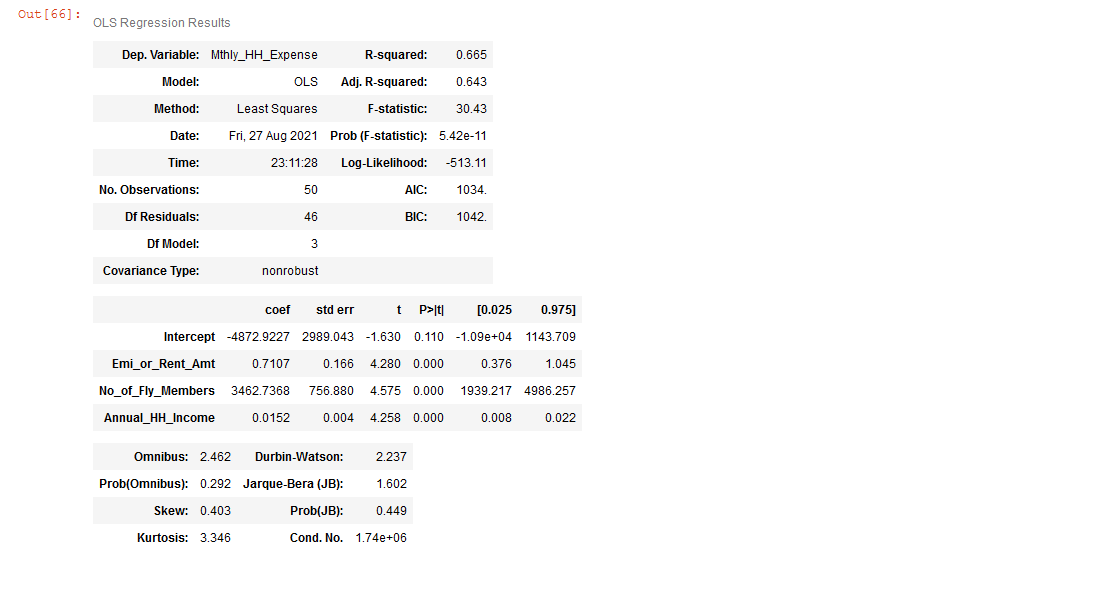
**Assignment (Linear Regression & Logistics Regression)**

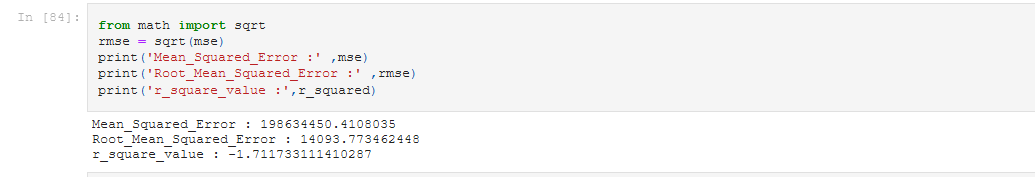
Q1 Write a Python program using Scikit-learn to split the iris dataset into 80% train data and 20% test data. Out of total 150 records, the training set will contain 120 records and the test set contains 30 of those records. Train or fit the data into the model and calculate the accuracy of the model using the Logistics Regression Algorithm.

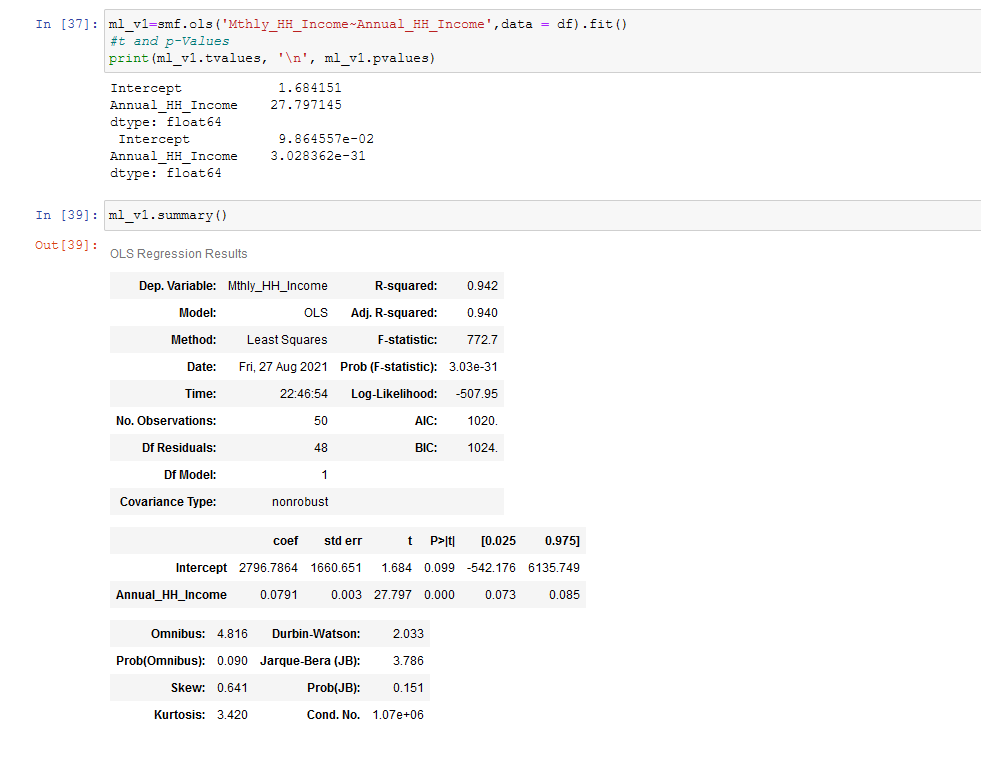


Note : Sir I attached pdf file along with this Assignment part.

<https://stackoverflow.com/questions/43159754/datasets-load-iris-in-python/43159815>

Q2 Write a Python program using Scikit-learn to split the attached dataset (Income & expenses data) into 80% train and 20% test data. Train or fit the data into the model and calculate the accuracy of the model using the Multiple Linear Regression Model? 





Note : Sir I attached pdf file along with this Assignment part.

Q3 What is the difference in the conventional programming and Machine learning Programming?

Major difference between machine learning and conventional programming language is the accuracy of predictions. Conventional programming language depends on algorithms within a variety of input parameters

**Conventional programming languages** involve manually creating programs by providing input data. The computer then generates an output based on programming logic

[**Machine Learning**](https://www.charterglobal.com/technology-solutions/machine-learning-development-services/) is a field of computer science that gives computers the ability to learn without being explicitly programmed. Machine learning teaches computers the ability to solve problems and perform complex tasks on their own. In most situations, problems solved using machine learning are based on the computer’s learning experience for which they wouldn’t have been solved by conventional programming languages. Such problems can be face recognition, driving, and diseases diagnosis. With conventional programming language, on the other hand, the behavior of the computer is coded by first creating a suitable algorithm that follows predesigned sets of rules.

Q4 Explain various steps involved in Machine Learning Process?

These 7 steps of machine learning can be applied to solve the problems :

* Gathering data
* Preparing that data
* Choosing a model
* Training
* Evaluation
* Hyper parameter tuning
* Prediction

Q5 Describe the evaluation techniques used for Linear Regression?

A linear regression is a regression technique where the independent variable has a linear relationship with the dependent variable. It is used in cases where we want to predict some continuous quantity. For example, a linear regression metrics can be used to quantify the relative impacts of gender, age, and diet on height. The linear regression evaluation metric is based on the least square estimation, which states that the regression co-efficient should be chosen in such a way that it minimizes the sum of the squared distance of each observed response and aims at finding the best fitting straight line, known as the regression line.

Evaluation techniques:

1) Mean Absolute Error(MAE)

MAE is a very simple metric which calculates the absolute difference between actual and predicted values.

2) Mean Squared Error(MSE)

MSE is a most used and very simple metric with a little bit of change in mean absolute error. Mean squared error states that finding the squared difference between actual and predicted value.

3) Root Mean Squared Error(RMSE)

As RMSE is clear by the name itself, that it is a simple square root of mean squared error.

Q6 Describe the evaluation techniques for Logistics Regression?

1.Accuracy Score

2.Recall: Recall can be thought of as a measure of a classifiers completeness. A low recall indicates many False Negatives.

3.Precision: Precision can be thought of as a measure of a classifiers exactness. A low precision can also indicate a large number of False Positives.

4. F1 Score: if you are dealing with a imbalanced dataset this is a must. If you are looking to select a model that is balanced between precision and recall F measure should be used.

Q7 What do you understand by imputation of missing values. Write a note on the methods of Imputation techniques?

Missing values are usually classified into three different types :

1. Missing Completely at Random (MCAR):  
Definition: The probability of an instance being missing does not depend on known values or the missing value itself.   
Example: A data table was printed with no missing values and someone accidentally dropped some ink on it so that some cells are no longer readable [2]. Here, we could assume that the missing values follow the same distribution as the known values.

2. Missing at Random (MAR) :  
Definition: The probability of an instance being missing may depend on known values but not on the missing value itself.  
Sensor Example: In the case of a temperature sensor, the fact that a value is missing doesn’t depend on the temperature, but might be dependent on some other factor, for example on the battery charge of the thermometer.   
Survey example: Whether or not someone answers a question - e.g. about age- in a survey **doesn’t** depend on the answer itself, but may depend on the answer to another question, i.e. gender female.

3.Not Missing at Random (NMAR):  
Definition: the probability of an instance being missing could depend on the value of the variable itself.  
Sensor example: In the case of a temperature sensor, the sensor doesn’t work properly when it is colder than 5°C.   
Survey example: Whether or not someone answers a question - e.g. number of sick days - in a survey **does** depend on the answer itself - as it could be for some overweight people.

### Different Methods to Handle Missing Values:

### 1.Deletion Methods

 There are three common deletion approaches: listwise deletion, pairwise deletion, and dropping features.

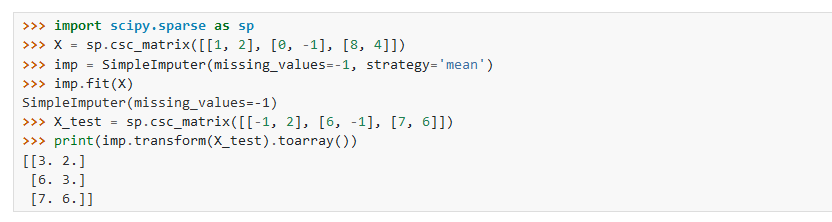
* **Listwise Deletion:** Delete all rows where one or more values are missing.
* **Pairwise Deletion:** Delete only the rows that have missing values in the columns used for the analysis. It is only recommended to use this method if the missing data are MCAR.
* **Dropping Features:** Drop entire columns with more missing values than a given threshold, e.g. 60%.

### Imputation Methods:

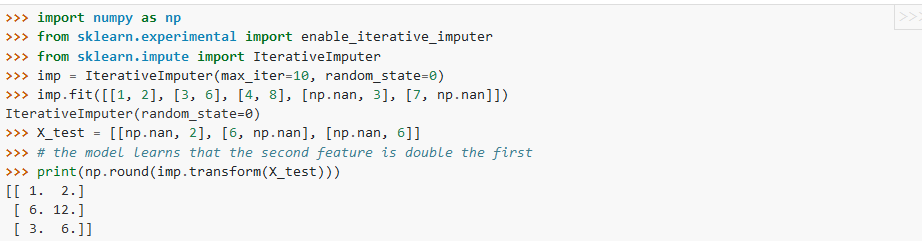
 The idea behind the imputation approach is to replace missing values with other sensible values. As you always lose information with the deletion approach when dropping either samples (rows) or entire features (columns), imputation is often the preferred approach.

The many imputation techniques can be divided into two subgroups: single imputation or multiple imputation.

In **single imputation**, a single / one imputation value for each of the missing observations is generated.  The imputed value is treated as the true value, ignoring the fact that no imputation method can provide the exact value. Therefore, single imputation does not reflect the uncertainty of the missing values.



In **multiple imputation**, many imputed values for each of the missing observations are generated. This means many complete datasets with different imputed values are created. The analysis (e.g. training a linear regression to predict a target column) is performed on each of these datasets and the results are polled. Creating multiple imputations, as opposed to single imputations, accounts for the statistical uncertainty in the imputations



### Single Imputation

 Most imputation methods are single imputation methods, following three main strategies: replacement by existing values, replacement by statistical values, and replacement by predicted values. Depending on the values used for each one of these strategies, we end up with methods that work on numerical values only and methods that work on both numerical and nominal columns. These methods are summarized in Table 1 and explained below.

|  |  |  |
| --- | --- | --- |
| **Replacement by:** | **Numerical Features Only** | **Numerical and Nominal Features** |
| **Existing values** | Minimum / Maximum | Previous / Next / Fixed |
| **Statistical values** | (Rounded) Mean / Median / Moving Average, Linear / Average Interpolation | Most Frequent |
| **Predicted values** | Regression Algorithms | Regression & Classification Algorithms, k-Nearest Neighbours |

*Table 1: Single imputation methods for numerical features only and for numerical and nominal features, based on existing values, statistical measures, and predicted values.*

 ***Fixed Value***

Fixed value imputation is a general method that works for all data types and consists of substituting the missing value with a fixed value. The aggregated customer example we mentioned at the beginning of this article uses fixed value imputation for numerical values. As an example of using fixed value imputation on nominal features, you can impute the missing values in a survey with “not answered”.

***Minimum / Maximum Value***

If you know that the data has to fit a given range [minimum, maximum], and if you know from the data collection process that  the measuring system stops recording and the signal saturates beyond one of such boundaries, you can use the range minimum or maximum as the replacement value for missing values. For example, if in the monetary exchange a minimum price has been reached and the exchange process has been stopped, the missing monetary exchange price can be replaced with the minimum value of the law’s exchange boundary.

***(Rounded) Mean / Median Value / Moving Average***

Other common imputation methods for numerical features are mean, rounded mean, or median imputation. In this case, the method substitutes the missing value with the mean, the rounded mean, or the median value calculated for that feature on the whole dataset. In the case of a high number of outliers in your dataset, it is recommended to use the median instead of the mean.

***Most Frequent Value***

Another common method that works for both numerical and nominal features uses the most frequent value in the column to replace the missing values.

***Previous / Next Value***

There are special imputation methods for time series or ordered data. These methods take into account the sorted nature of the dataset, where close values are probably more similar than distant values. A common approach for imputing missing values in time series substitutes the next or previous value to the missing value in the time series. This approach works for both numerical and nominal values.

***Linear / Average Interpolation***

Similarly to the previous/next value imputation, but only applicable to numerical values, is linear or average interpolation, which is calculated between the previous and next available value, and substitutes the missing value. Of course, as for all operations on ordered data, it is important to sort the data correctly in advance, e.g. according to a timestamp in the case of time series data.

***K Nearest Neighbors***

The idea here is to look for the k closest samples in the dataset where the value in the corresponding feature is not missing and to take the feature value occurring most frequently in the group as a replacement for the missing value.

***Missing Value Prediction***

Another common option for single imputation is to train a machine learning model to predict the imputation values for feature x based on the other features. The rows without missing values in feature x are used as a training set and the model is trained based on the values in the other columns. Here we can use any classification or regression model, depending on the data type of the feature. After training, the model is applied to all samples with the feature missing value to predict its most likely value.

In the case of missing values in more than one feature column, all missing values are first temporarily imputed with a basic imputation method, e.g. the mean value. Then the values for one column are set back to missing. The model is then trained and applied to fill in the missing values. In this way, one model is trained for each feature with missing values, until all missing .

### Multiple Imputation

 Multiple imputation is an imputation approach stemming from statistics. Single imputation methods have the disadvantage that they don’t consider the uncertainty of the imputed values. This means they recognize the imputed values as actual values not taking into account the standard error, which causes bias in the results [3][4].

An approach that solves this problem is multiple imputation where not one, but many imputations are created for each missing value. This means filling in the missing values multiple times, creating multiple “complete” datasets [3][4].

A number of algorithms have been developed for multiple imputation. One well known algorithm is Multiple Imputation by Chained Equation (MICE).

***Multiple Imputation by Chained Equations (MICE)***

Multiple Imputation by Chained Equations (MICE) is a robust, informative method for dealing with missing values in datasets. MICE operates under the assumption that the missing data are Missing At Random (MAR) or Missing Completely At Random (MCAR) [3].

The procedure is an extension of the single imputation procedure by “Missing Value Prediction” (seen above): this is step 1. However, there are two additional steps in the MICE procedure.

Step 1: This is the process as in the imputation procedure by “Missing Value Prediction” on a subset of the original data. One model is trained to predict the missing values in one feature, using the other features in the data row as the independent variables for the model. This step is repeated for all features. This is a cycle or iteration.

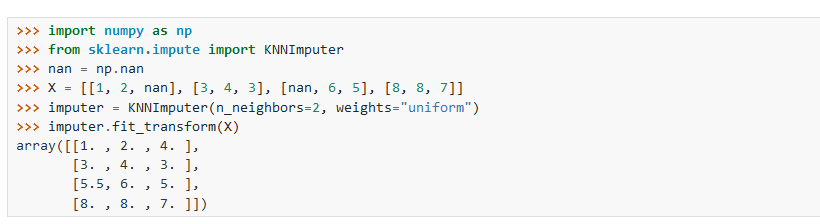
Step 2: Step 1 is repeated k times, each time using the most recent imputations for the independent variables, until convergence is reached. Most often, k=10 cycles are sufficient.

Step 3: The whole process is repeated N times on N different random subsets. The resulting N models will be slightly different, and will produce N slightly different predictions for each missing value.

The analysis, e.g. training a linear regression for a target variable, is now performed on each one of the N final datasets. Finally the results are combined, often this is also called pooling.

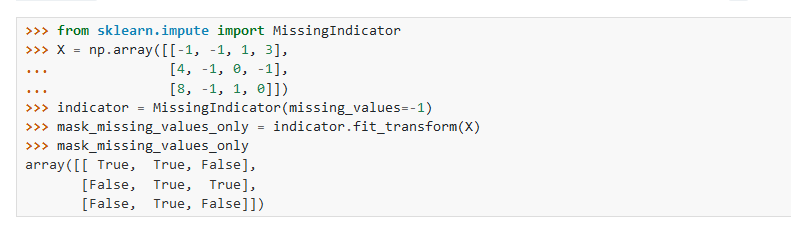
**3. Nearest neighbors imputation**

The [KNNImputer](https://scikit-learn.org/stable/modules/generated/sklearn.impute.KNNImputer.html#sklearn.impute.KNNImputer) class provides imputation for filling in missing values using the k-Nearest Neighbors approach. By default, a Euclidean distance metric that supports missing values, nan\_euclidean\_distances, is used to find the nearest neighbors. Each missing feature is imputed using values from n\_neighbors nearest neighbors that have a value for the feature. The feature of the neighbors are averaged uniformly or weighted by distance to each neighbor. If a sample has more than one feature missing, then the neighbors for that sample can be different depending on the particular feature being imputed. When the number of available neighbors is less than n\_neighbors and there are no defined distances to the training set, the training set average for that feature is used during imputation. If there is at least one neighbor with a defined distance, the weighted or unweighted average of the remaining neighbors will be used during imputation. If a feature is always missing in training, it is removed during transform.



4. Marking imputed values

The [MissingIndicator](https://scikit-learn.org/stable/modules/generated/sklearn.impute.MissingIndicator.html#sklearn.impute.MissingIndicator) transformer is useful to transform a dataset into corresponding binary matrix indicating the presence of missing values in the dataset. This transformation is useful in conjunction with imputation. When using imputation, preserving the information about which values had been missing can be informative.



Q7 Discuss some of the ways we can find the normality of the Numerical data?

### 1. MinMaxScaler()

Normalization is a rescaling of the data from the original range so that all values are within the new range of 0 and 1.

We can demonstrate the usage of this class by converting two variables to a range 0-to-1, the default range for normalization. The first variable has values between about 4 and 100, the second has values between about 0.1 and 0.001.

y = (x – min) / (max – min)

2. Data Standardization

Standardizing a dataset involves rescaling the distribution of values so that the mean of observed values is 0 and the standard deviation is 1.

y = (x – Mean) / SD

Q8 What is meant by balanced and unbalanced data?

**Balanced Dataset:** Let’s take a simple example if in our data set we have positive values which are approximately same as negative values. Then we can say our dataset in balance.

Female :50% , Male:50%

**Imbalanced Dataset: —** If there is the very high different between the positive values and negative values. Then we can say our dataset in Imbalance Dataset.

Split the data set as : 70% Positive and 30% Negative

Q9 Which of the following statement is TRUE.

1. Outliers should be identified and removed always from the dataset.
2. Outliers can never be present in the testing dataset.
3. Outlier is a data point which is significantly close to other data points.
4. The nature of our business problem determines how outliers are used.

Answer :D

Q10 In Regression modelling we develop a mathematical equation that  
describes how, (Predictor-Independent variable, Response-Dependent variable)

 (A) one predictor and one or more response variables are related.

 (B) several predictors and several response variables response is related.

 (C) one response and one or more predictors are related.

 (D) All of these are correct.

Answer C: Explanation: In the regression problem statement, we have several independent variables but only one dependent variable.