Experiment No	Experiment Name
1	Data Wrangling, I Perform the following operations using Python on any open source dataset (e.g., data.csv) 1. Import all the required Python Libraries. 2. Locate an open source data from the web (e.g. https://www.kaggle.com). Provide a clear description of the data and its source (i.e., URL of the web site). 3. Load the Dataset into pandas data frame. 4. Data Preprocessing: check for missing values in the data using pandas insult(), describe() function to get some initial statistics. Provide variable descriptions. Types of variables etc. Check the dimensions of the data frame. 5. Data Formatting and Data Normalization: Summarize the types of variables by checking the data types (i.e., character, numeric, integer, factor, and logical) of the variables in the data set. If variables are not in the correct data type, apply proper type conversions. 6. Turn categorical variables into quantitative variables in Python. In addition to the codes and outputs, explain every operation that you do in the above steps and explain everything that you do to import/read/scrape the data set
2	Data Wrangling II Create an “Academic performance” dataset of students and perform the following operations using Python. 1. Scan all variables for missing values and inconsistencies. If there are missing values and/or inconsistencies, use any of the suitable techniques to deal with them. 2. Scan all numeric variables for outliers. If there are outliers, use any of the suitable techniques to deal with them. 3. Apply data transformations on at least one of the variables. The purpose of this transformation should be one of the following reasons: to change the scale for better understanding of the variable, to convert a non-linear relation into a linear one, or to decrease the skewness and convert the distribution into a normal distribution. Reason and document your approach properly.
3	Descriptive Statistics - Measures of Central Tendency and variability Perform the following operations on any open source dataset (e.g., data.csv) 1. Provide summary statistics (mean, median, minimum,

	maximum, standard deviation) for a dataset (age, income etc.) with numeric variables grouped by one of the qualitative (categorical) variable. For example, if your categorical variable is age groups and quantitative variable is income, then provide summary statistics of income grouped by the age groups. Create a list that contains a numeric value for each response to the categorical variable. 2. Write a Python program to display some basic statistical details like percentile, mean, standard deviation etc. of the species of „Iris-setosa‟, „Iris-versicolor‟ and „Irisversicolor‟ of iris.csv dataset. Provide the codes with outputs and explain everything that you do in this step.
4	Data Analytics I Create a Linear Regression Model using Python/R to predict home prices using Boston Housing Dataset (https://www.kaggle.com/c/boston-housing). The Boston Housing dataset contains information about various houses in Boston through different parameters. There are 506 samples and 14 feature variables in this dataset. The objective is to predict the value of prices of the house using the given features.
5	Data Analytics II 1. Implement logistic regression using Python/R to perform classification on Social_Network_Ads.csv dataset. 2. Compute Confusion matrix to find TP, FP, TN, FN, Accuracy, Error rate, Precision, Recall on the given dataset.
6	Data Analytics III 1. Implement Simple Naïve Bayes classification algorithm using Python/R on iris.csv dataset. 2. Compute Confusion matrix to find TP, FP, TN, FN, Accuracy, Error rate, Precision, Recall on the given dataset
7	Text Analytics 1. Extract Sample document and apply following document preprocessing methods: Tokenization, POS Tagging, stop words removal, Stemming and Lemmatization. 2. Create representation of document by calculating Term Frequency and Inverse Document Frequency
8	Data Visualization I 1. Use the inbuilt dataset 'titanic'. The dataset contains 891 rows and contains information about the passengers who boarded the

	unfortunate Titanic ship. Use the Seaborn library to see if we can find any patterns in the data. 2. Write a code to check how the price of the ticket (column name: 'fare') for each passenger is distributed by plotting a histogram
9	Data Visualization II 1. Use the inbuilt dataset 'titanic' as used in the above problem. Plot a box plot for distribution of age with respect to each gender along with the information about whether they survived or not. (Column names : 'sex' and 'age') 2. Write observations on the inference from the above statistics
10	Data Visualization III Download the Iris flower dataset or any other dataset into a DataFrame. (e.g., https://archive.ics.uci.edu/ml/datasets/Iris). Scan the dataset and give the inference as: 1. List down the features and their types (e.g., numeric, nominal) available in the dataset. 2. Create a histogram for each feature in the dataset to illustrate the feature distributions. 3. Create a box plot for each feature in the dataset. 4. Compare distributions and identify outliers.
11	Write a code in JAVA for a simple Word Count application that counts the number of occurrences of each word in a given input set using the Hadoop Map-Reduce framework on local-standalone set-up.
12	Design a distributed application using Map-Reduce which processes a log file of a system. 3. Locate dataset (e.g., sample_weather.txt) for working on weather data which reads the text input files and finds average for temperature, dew point and wind speed.
13	Write a simple program in SCALA using Apache Spark framework
14	Write a case study on Global Innovation Network and Analysis (GINA). Components of analytic plan are 1. Discovery business problem framed, 2. Data, 3. Model planning analytic technique and 4. Results and Key findings
15	Write a case study to process data driven for Digital Marketing OR Health care systems with Hadoop Ecosystem components as shown. (Mandatory) ● HDFS: Hadoop Distributed File System ● YARN: Yet Another Resource Negotiator ● MapReduce: Programming based Data

	Processing ● Spark: In-Memory data processing ● PIG, HIVE: Query based processing of data services ● HBase: NoSQL Database (Provides real-time reads and writes) ● Mahout, Spark MLLib: (Provides analytical tools) Machine Learning algorithm libraries ● Solar, Lucene: Searching and Indexing

Experiment No. 1

Aim:

Data Wrangling, I

Perform the following operations using Python on any open source dataset (e.g., data.csv)

1) Import all the required Python Libraries.

2) Locate an open source data from the web (e.g., https://www.kaggle.com). Provide a clear description of the data and its source (i.e., URL of the web site).

3) Load the Dataset into pandas dataframe.

4) Data Preprocessing: check for missing values in the data using pandas isnull(), describe() function to get some initial statistics. Provide variable descriptions. Types of variables etc. Check the dimensions of the data frame.

5) Data Formatting and Data Normalization: Summarize the types of variables by checking the data types (i.e., character, numeric, integer, factor, and logical) of the variables in the data set. If variables are not in the correct data type, apply proper type conversions.

6) Turn categorical variables into quantitative variables in Python.

Requirement:

• Anaconda Installer

• Windows 10 OS

• Jupyter Notebook

Theory:

Data wrangling is the process of cleaning and unifying messy and complex data sets for easy access and analysis.

With the amount of data and data sources rapidly growing and expanding, it is getting increasingly essential for large amounts of available data to be organized for analysis.

This process typically includes manually converting and mapping data from one raw form

into another format to allow for more convenient consumption and organization of the data.

The Goals of Data Wrangling:

• Reveal a "deeper intelligence" by gathering data from multiple sources

• Provide accurate, actionable data in the hands of business analysts in a timely matter

Reduce the time spent collecting and organizing unruly data before it can be utilized

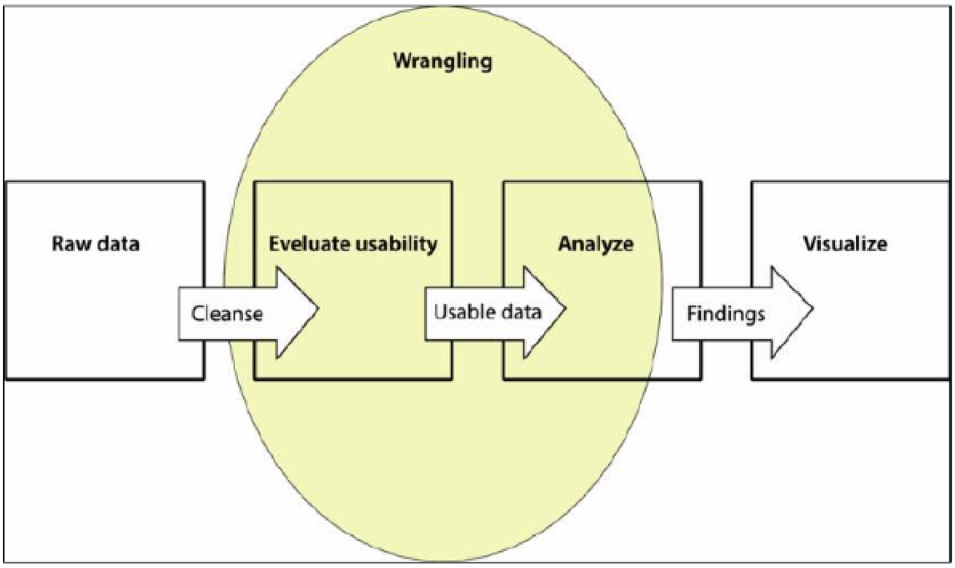
• Enable data scientists and analysts to focus on the analysis of data, rather than the wrangling

• Drive better decision-making skills by senior leaders in an organization Key steps to Data Wrangling:

• Data Acquisition: Identify and obtain access to the data within your sources.

• Joining Data: Combine the edited data for further use and analysis.

• Data Cleansing: Redesign the data into a usable and functional format and correct/remove any bad data.



Libraries Used:

Pandas: Pandas is a Python library used for working with data sets. It has functions for analyzing, cleaning, exploring and manipulating data.

Numpy: NumPy is a Python library used for working with arrays. It also has functions for working in domain of linear algebra, fourier transform, and matrices.

Conclusion:

Hence, we have implemented data wrangling practical.

Experiment No 2

Aim:

Data Wrangling II

Create an "Academic performance" dataset of students and perform the following operations using Python.

1) Scan all variables for missing values and inconsistencies. If there are missing values and/or inconsistencies, use any of the suitable techniques to deal with them.

2) Scan all numeric variables for outliers. If there are outliers, use any of the suitable techniques to deal with them.

3) Apply data transformations on at least one of the variables. The purpose of this transformation should be one of the following reasons: to change the scale for better understanding of the variable, to convert a non-linear relation into a linear one, or to decrease the skewness and convert the distribution into a normal distribution.

Requirement:

• Anaconda Installer

• Windows 10 OS

• Jupyter Notebook

Theory:

Data Wrangling:

Data wrangling can be defined as the process of cleaning, organizing, and transforming raw data into the desired format for analysts to use for prompt decision making. Also known as data cleaning or data munging, data wrangling enables businesses to tackle more complex data in less time, produce more accurate results, and make better decisions. The exact methods vary from project to project depending upon your data and the goal you are trying to achieve. More and more organizations are increasingly relying on data wrangling tools to make data ready for downstream analytics.

Benefits of Data Wrangling:

- Data wrangling helps to improve data usability as it converts data into a compatible format for the end system.

- It helps to quickly build data flows within an intuitive user interface and easily schedule and automate the data-flow process.

- Integrates various types of information and their sources (like databases, web services, files, etc.)

- Help users to process very large volumes of data easily and easily share data-flow techniques.

Data Wrangling Vs. ETL:

ETL stands for Extract, Transform and Load. ETL is a middleware process that involves mining or extracting data from various sources, joining the data, transforming data as per business rules, and subsequently loading data to the target systems. ETL is generally used for loading processed data to flat files or relational database tables. Though Data Wrangling and ETL look similar, there are key differences between data wrangling and ETL processes that set them apart.

- Users – Analysts, statisticians, business users, executives, and managers use data wrangling. In comparison, DW/ETL developers use ETL as an intermediate process linking source systems and reporting layers.

- Data Structure – Data wrangling involves varied and complex data sets, while ETL involves structured or semi-structured relational data sets.

- Use Case – Data wrangling is normally used for exploratory data analysis, but ETL is used for gathering, transforming, and loading data for reporting.

Data Wrangling Tools:

- Spreadsheets / Excel Power Query - It is the most basic manual data wrangling tool

- Tabula – It is a tool suited for all data types

- Google DataPrep – It is a data service that explores, cleans, and prepares

data

- Data wrangler – It is a data cleaning and transforming tool

Conclusion:

Hence, we have implemented Data Wrangling Practical II.

Experiment No 3

Aim:

1) Provide summery statistics for a dataset with numeric varibles grouped by one of the qualitative variables.

2) write a python program to display some basic statistical details like percentile, mean, standard devivation , etc.

Requirement:

•Anaconda Installer

•Windows 10 OS

•Jupyter Notebook

Theory:

Step 1: Provide summery statistics such as mean, median, mode, standard deviation.

(1) Mean:

"Average" value is termed as mean of the dataset

means= sum of all data values / Total number of Data Values

(2) Median:

The middle values of sorted dataset is known as median.

(3) Mode:

mode refers to most frequently occuring values in the dataset.

e.g. Consider the weight (in kg) of 5children as 36,40,32,42,30.lets compute

mean, median,mode

(1) Mean=(36+40+32+42+30) / 5

= 36 kg

(2) median: arrange the data in ascending order : 30,22,36,40,42

the middle value is 36. so median is 36 kg.

(3) mode: 36 kg occurs most number of times so mode=36 kg

Calculate mean using python:

```
df = pd.Dataframe(dict)
mean=df.[‘score’].mean()
print(mean)
```

Calculate median using python:

```
df = pd.Dataframe(dict)
mode=df.mode()
print(mode)
```

Calculating maximum and minimum python:

```
df=pd.Dataframe[[10,20,30,40],[7,14,21,28],[55,15,8,12],[15,14,1,8],[7,1,1,8],[5,4,9,2]]
coloumns=[‘Apple’, ‘Orange’, ‘Banana’ , ‘Peer’]
index=[‘Basket1’ , ‘Basket2’ , ‘Basket3’ , ‘Basket4’ , ‘ Basket5’ , ‘ Basket6’]
minimum=df.[[‘Apple’ , ‘Orange’ , ‘ Banana’, ‘Peer’]].min();
maximum=df.[[‘Apple’ , ‘Orange’ , ‘ Banana’, ‘Peer’]].max();
print(minimum);
print(maximum);
```

Standard Deviation:

The standard deviation is a statistic that measures the dispersion of a dataset relative to its mean and is calculated as the square root of the variance.

The standard deviation is calculated as the square root of variance by determining each data point's deviation relative to the mean.

The standard deviation is a statistic that measures the dispersion of a dataset relative to its mean and is calculated as the square root of the variance.

The standard deviation is calculated as the square root of variance by determining each data point's deviation relative to the mean.

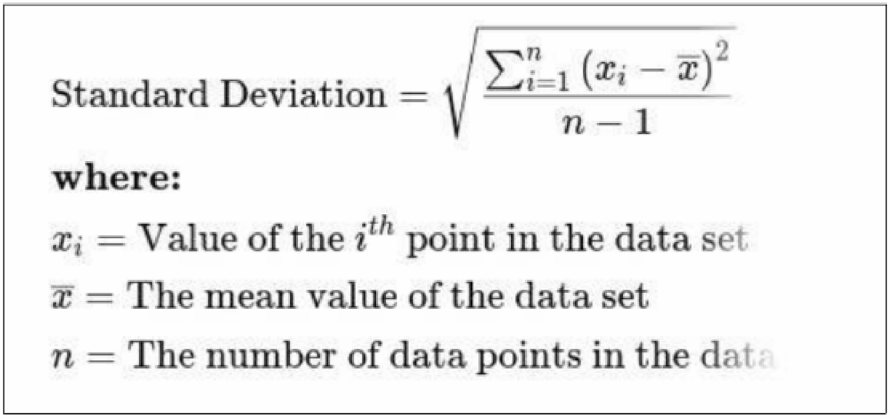
Key Takeaways:

•It is calculated as the square root of the variance.

•Standard deviation, in finance, is often used as a measure of a relative riskiness of an asset.

•A volatile stock has a high standard deviation, while the deviation of a stable blue-chip stock is usually rather low.

•As a downside, the standard deviation calculates all uncertainty as risk, even when it's in the investor's favor—such as above-average returns



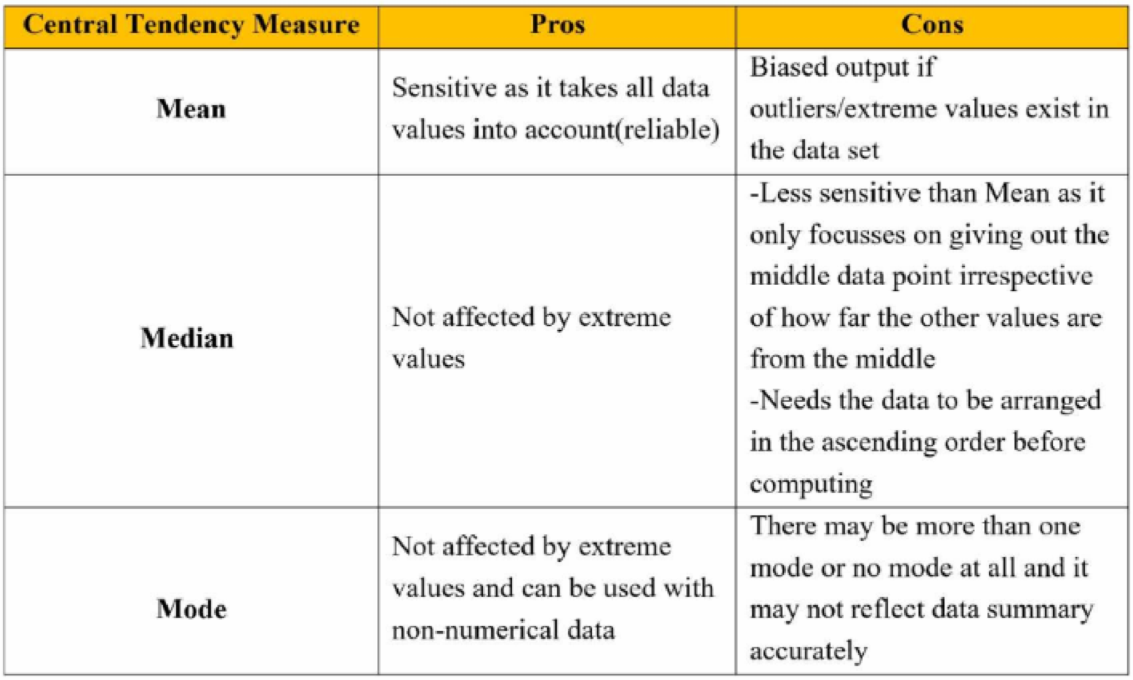
$$\text{Standard Deviation} = \sqrt{\frac{\sum_{i=1}^{n} (x_i - \overline{x})^2}{n-1}}$$

where:

x_i = Value of the i^{th} point in the data set

$\overline{x}$ = The mean value of the data set

n = The number of data points in the data



Central Tendency Measure	Pros	Cons
Mean	Sensitive as it takes all data values into account(reliable)	Biased output if outliers/extreme values exist in the data set
Median	Not affected by extreme values	-Less sensitive than Mean as it only focusses on giving out the middle data point irrespective of how far the other values are from the middle -Needs the data to be arranged in the ascending order before computing
Mode	Not affected by extreme values and can be used with non-numerical data	There may be more than one mode or no mode at all and it may not reflect data summary accurately

Libraries Used:

1. Pandas: Pandas is a Python library used for working with data sets. It has functions for analyzing, cleaning, exploring and manipulating data.

2.Numpy: is a library consisting of multidimensional array objects and a collection of routines for processing those arrays. Using NumPy, mathematical and logical operations on arrays can be performed.

3. Seaborn: Seaborn is a data visualization library built on top of matplotlib and closely integrated with pandas data structures in Python.

Conclusion:

From this experiment we learnt how to calculate mean, median and mode.

Experiment No: 4

Aim: Create a Linear Regression model using python to predict home prise using

Boston Housing dataset.

Requirement:

•Anaconda Installer

•Windows 10 OS

•Jupyter Notebook

Theory:

Linear Regression in data science:

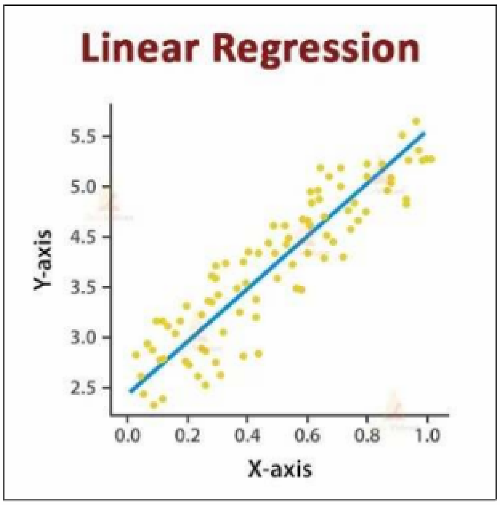


fig. Linear Regression

The term regression is used when you try to find the relationship between variables. In Machine Learning and in statistical modeling, that relationship is used to predict the outcome of events.

Simple Linear Regression:

Simple linear regression is an approach for predicting a response using a single feature. It is assumed that the two variables are linearly related. Hence, we try to find a linear function that predicts the response value(y) as accurately as possible as a

function of the feature or independent variable(x).

Multiple linear regression:

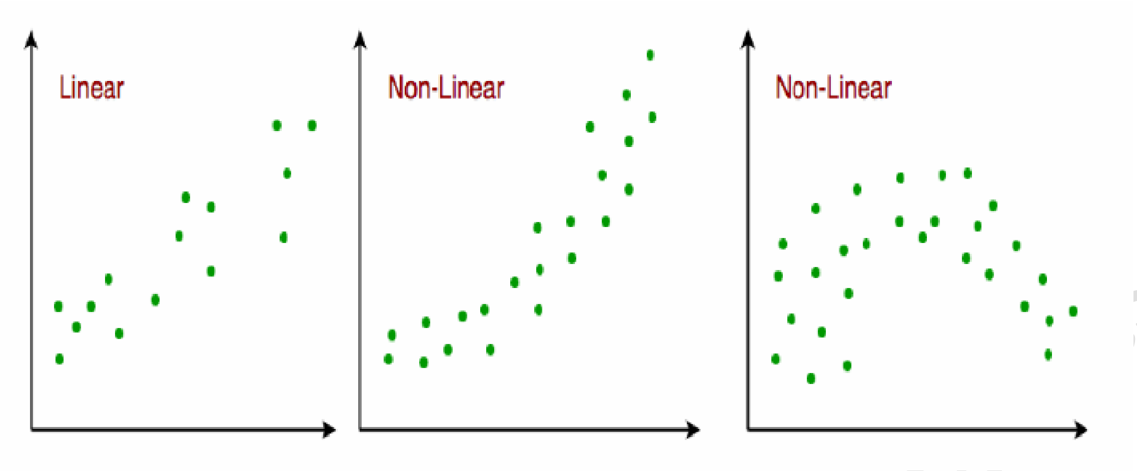
Multiple linear regression attempts to model the relationship between two or more features and a response by fitting a linear equation to the observed data. Clearly, it is nothing but an extension of simple linear regression.

Assumptions:

Given below are the basic assumptions that a linear regression model makes regarding

a dataset on which it is applied:

• Linear relationship: Relationship between response and feature variables should be linear. The linearity assumption can be tested using scatter plots. As shown below, 1st figure represents linearly related variables whereas variables in the 2nd and 3rd figures are most likely non-linear. So, 1st figure will give better predictions using linear regression.



Applications:

1. Trend lines:6A trend line represents the variation in quantitative data with the passage of time (like GDP, oil prices, etc.). These trends usually follow a linear relationship. Hence, linear regression can be applied to predict future values. However, this method suffers from a lack of scientific validity in cases where other potential changes can affect the data.

2. Economics:6Linear regression is the predominant empirical tool in economics. For example, it is used to predict consumer spending, fixed investment spending, inventory investment, purchases of a country's exports, spending on imports, the demand to hold liquid assets, labor demand, and labor supply.

3. Finance:6The capital price asset model uses linear regression to analyze and quantify the systematic risks of an investment.

Biology: Linear regression is used to model causal relationships between parameters in biological systems.

Dataset used:

•In this experminet we are going to use the boston housing dataset which contain information about various houses in boston through different parameters.

•There are total 506 samples ad 14 features (columns) in this dataset.

•Our objective is to predict the value of prices of the house using features with the help of linear regression.

Libraries Uesd:

1. Pandas: Pandas is a Python library used for working with data sets. It has functions for analyzing, cleaning, exploring and manipulating data.

2. Sklearn: It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python.

Conclusion:

In this experiment we have studied about linear regression and done house prise prediction using boston housing dataset.

Experiment No 5

Aim: Implement logistic regression using python to perform classifiaction on social network ads , cv dataset.

Requirement:

•Anaconda Installer

•Windows 10 OS

•Jupyter Notebook

Theory:

Logistic Regression?

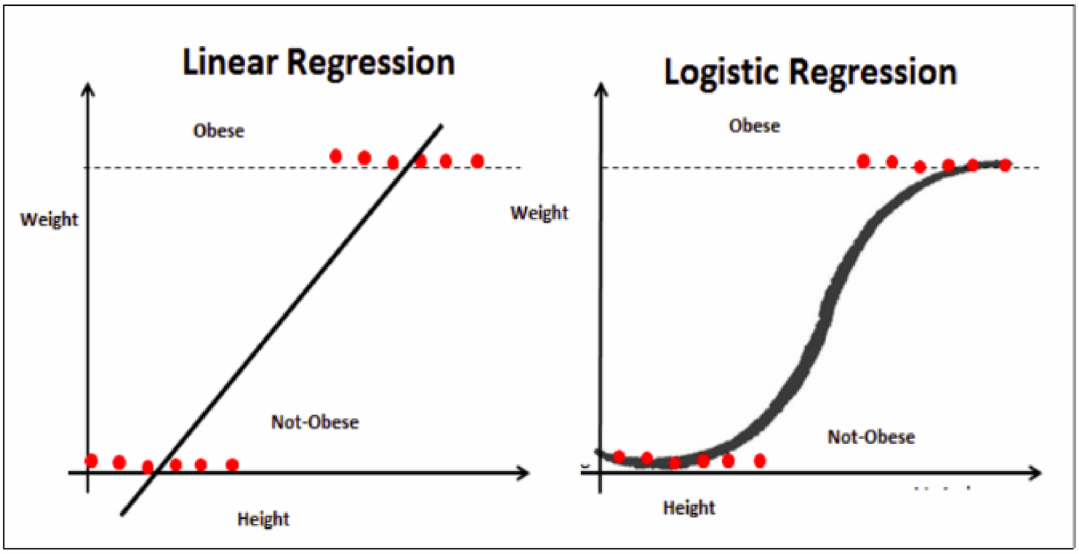


Fig. Linear Regression Vs Logistic Regression

•Logistic regression is a supervised learning algorithm used to predict a dependent categorical target variable. In essence, if you have a large set of data that you want to categorize, logistic regression may be able to help.

•For example, if you were given a dog and an orange and you wanted to find out whether each of these items was an animal or not, the desired result would be for the dog to end up classified as an animal, and for the orange to be categorized as not an animal.

•Animal is your target; it is dependent on your data in order to be able to classify the item correctly. In this example, there are only two possible answers (binary logistic regression), animal or not an animal. However, it is also possible to set up your logistic regression with more than two possible categories (multinomial logistic regression).

•To dive a little deeper into how your model might attempt to classify these two items directly, let's consider what else the model would need to know about the items in order to decide where they belong. Other similar aspects of these items would need to be looked at when considering how to classify each item or data point. Aspects, or features, may include color, size, weight, shape, height, volume or amount of limbs.

•In this way, knowing that an orange's shape was a circle may help the algorithm to conclude that the orange was not an animal. Similarly, knowing that the orange had zero limbs would help as well.

•Logistic regression requires that the dependent variable, in this case whether the item was an animal or not, be categorical. The outcome is either animal or not an animal—there is no range in between.

•A problem that has a continuous outcome, such as predicting the grade of a student or the fuel tank range of a car, is not a good candidate to use logistic regression. Other options like linear regression may be more appropriate.

Types of Logistic Regression:

There are three main types of logistic regression:

1) binary

2) multinomial

3) ordinal.

They differ in execution and theory. Binary regression deals with two possible values, essentially: yes or no. Multinomial logistic regression deals with three or more values. And ordinal logistic regression deals with three or more classes in a predetermined order.

1. Binary logistic regression:

Binary logistic regression was mentioned earlier in the case of classifying an object as an animal or not an animal—it's an either/or solution. There are just two possible outcome answers. This concept is typically represented as a 0 or a 1 in coding.

Examples include:

• Whether or not to lend to a bank customer (outcomes are yes or no).

• Assessing cancer risk (outcomes are high or low).

• Will a team win tomorrow's game (outcomes are yes or no).

2. Multinomial logistic regression:

Multinomial logistic regression is a model where there are multiple classes that an item can be classified as. There is a set of three or more predefined classes set up prior to running the model.

Examples include:

• Classifying texts into what language they come from.

• Predicting whether a student will go to college, trade school or into the workforce.

• Does your cat prefer wet food, dry food or human food?

3. Ordinal logistic regression:

Ordinal logistic regression is also a model where there are multiple classes that an item can be classified as; however, in this case an ordering of classes is required. Classes do not need to be proportionate. The distance between each class can vary.

Examples include:

• Ranking restaurants on a scale of 0 to 5 stars.

• Predicting the podium results of an Olympic event.

• Assessing a choice of candidates, specifically in places that institute ranked-choice voting.

4. Logistic regression assumptions:

• Remove highly correlated inputs.

• Consider removing outliers in your training set because logistic regression will not give significant weight to them during its calculations.

• Does not favor sparse (consisting of a lot of zero values) data.

• Logistic regression is a classification model, unlike linear regression.

Libraries Used:

1. Pandas: Pandas is a Python library used for working with data sets. It has functions for analyzing, cleaning, exploring and manipulating data.

2. Sklearn: It provides a selection of efficient tools for machine learning and statistical modelling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python.

3. Seaborn: Seaborn is a data visualization library built on top of matplotlib and closely integrated with pandas data structures in Python.

4. Matplotlib: Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Matplotlib makes easy things easy and hard things possible.

Conclusion:

In this experiment we have studied about the logistic regression model. We have performed the classification on the social network. Ads dataset using various libraries of python.

Experiment No 6

Aim: Implement simple Navie Bayes Classifications algorithm. Using python on iris.csv dataset.

Requirement:

•Anaconda Installer

•Windows 10 OS

•Jupyter Notebook

Theory:

Naïve Bayes Classifier Algorithm:

•Naïve Bayes algorithm is a supervised learning algorithm, which is based on(Bayes theorem(and used for solving classification problems.

•It is mainly used in(text classification(that includes a high-dimensional training dataset.

•Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.

•It is a probabilistic classifier, which means it predicts on the basis of the probability of an object.

•Some popular examples of Naïve Bayes Algorithm are(spam filtration, Sentimental analysis, and classifying articles. Why is it called Naïve Bayes?

The Naïve Bayes algorithm is comprised of two words Naïve and Bayes, Which can be described as:

•Naïve: It is called Naïve because it assumes that the occurrence of a certain feature is independent of the occurrence of other features. Such as if the fruit is identified on the bases of color, shape, and taste, then red, spherical, and sweet fruit is recognized as an apple. Hence each feature individually contributes to identify that it is an apple without depending on each other.

•Bayes: It is called Bayes because it depends on the principle of(Bayes' Theorem.

Bayes' Theorem:

•Bayes' theorem is also known as(Bayes' Rule(or(Bayes' law, which is used to determine the probability of a hypothesis with prior knowledge. It depends on the conditional probability.

•The Formula for Bayes’ Theorem is given by

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Where,

P(A|B) is Posterior probability: Probability of hypothesis A on the observed event B.

P(B|A) is Likelihood probability: Probability of the evidence given that the probability of a hypothesis is true.

P(A) is Prior Probability: Probability of hypothesis before observing the evidence.

P(B) is Marginal Probability: Probability of Evidence