**PROJECT REPORT**

**ON**

**Smartphone Analysis**

**SUBMITTED BY**

**(Uthej Bhagwanth Reddy)**

**UNDER THE GUIDANCE OF**

**PROF.NILOFAR SHAIKH**

**SUBMITTED IN THE PARTIAL FULFILMENT OF**

**MSc COMPUTER SCIENCE Semester IV**

**FOR THE ACADEMIC YEAR 2019-2020**

**UNIVERSITY OF MUMBAI**

Department of Computer Science



S.I.E.S(Nerul) College of Arts, Science and Commerce,

Navi Mumbai-400706

SIES (NERUL) COLLEGE OF ARTS, SCIENCE AND COMMERCE

NAAC ACCREDITED ‘A’ GRADE COLLEGE (ISO 9001:2015 CERTIFIED INSTITUTION) NERUL, NAVI MUMBAI - 400706



THIS IS TO CERTIFY THAT THE PROJECT TITLED

**SMARTPHONE ANALYSIS**

IS UNDERTAKEN BY

**UTHEJ BHAGWANTH REDDY**

Seat No: 4104636

In partial fulfillment of the MSc - IT / CS Degree (Semester\_\_IV ) Examination in the academic year 2019-2020 and has not been submitted for any other examination and does not form part of any other course undergone by the candidate. It is further certified that he/she has completed all the required phases of the Project.

Project Guide External Examiner

Head of Department Principal

**ACKNOWLEDGEMENT**

The success and final outcome of this project required a lot of guidance and assistance from many people. All that I have done is only due to such supervision and assistance and I would not forget to thanks.

I would like to express my special gratitude to my project guide Asst Professor Nilofar shaikh who took keen interest on our project work and guided us all along. Big Data was a new subject for us and the concepts within it was made simpler and was well explained by our teacher.

I am thankful to and fortunate enough to get constant encouragement, support and guidance from all Teaching staffs of Department of computer science which helped us in successfully gathering the project requirements. Also, I would like to extend our sincere regards to all the non-teaching staff of department of computer science for their timely support.

**TABLE OF CONTENT**

1. **Introduction**
2. **All about Smartphones and History**
3. **Objective**
4. **Flow Of Project**
5. **Project code**
6. **Reference**

**INTRODUCTION:**

The markets for Mobile phones are increasing due to increased Competition and changing consumer brand preferences. The aim is to address the question by analyse which brand is preferred most by mobile phone users while purchasing mobile phones . And also wishes to found out the process of decision making while choosing and purchasing the branded mobile product with respect to their current satisfaction level on the particular brand. The data thus collected were analysed by using simple Big Data algorithms.

The Mobile phones were introduced in the mid-1980s and in the last two decades their ownership and use has increased in many parts of the world. In the Western European markets rank second collectively in mobile phone prevalence, with roughly 80% of all households owning a wireless handset (http://www.ipsosinsight.com). Interestingly, the popularity of mobile phones is a result of the communication and flexibility that they facilitate and the personal safety issues that they overcome.

A small smartphone that is only four inches long enables the sharing of real-time information and knowledge, and it can transform lifestyles. People use smartphones to obtain, share and exchange information whenever they need. The smartphone makes it possible for people to realize a genuinely ubiquitous life in a variety of areas, such as business, education, social interaction, and leisure. In this i investigate the global smartphone market by analyzing its growth and the competitive situation. Based on these analyses, I discuss the possible strategy of each player in the market.

Mobile phones and the Internet have played an increasingly prevalent role in the advancements in communication. Each has separately revolutionized the way that people communicate and access information. The recent advancements in hardware and networking technologies have allowed people to access the Internet on their mobile phones, providing a new platform to develop and communicate. The first part of this thesis discusses existing literature on the background of mobile phones, the Internet, software applications, and existing mobile interface design and usability.

**All about Smartphones and History**

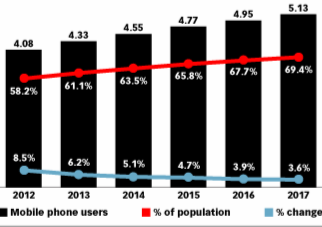
**The concept of the smart phone:**

A smartphone is a new form of mobile Internet device that combines the traditional features of a phone and a PDA.2 Another noteworthy definition of a smartphone is that it is a mobile phone that offers more advanced computing ability and connectivity than a basic current mobile phone does.3 The common aspect of the two definitions is that a smartphone is an integrated device with mobile telephone technology and the ability to access the Internet. The smartphone first achieved both the functionality of a traditional phone and technology of a PC. Unlike traditional phones, which are produced as finished goods, a smartphone enables users to install, add, and delete hundreds of applications. Through various applications, users can also personalize the interface. Hence, because smartphones allow free access to the Internet regardless of time and location, users of smartphones are entering an era of ubiquitous information.

As per literature, the Mobile phones were introduced in the mid-1980s and in the last two decades their ownership and use has increased dramatically in many parts of the world. Also, Mobile phone ownership at the end of 2005 was at near saturation levels in many areas of the world–most notably in East Asia, as over 90% of all households in South Korea, Japan and urban China own at least one mobile phone (http://www.ipsosinsight.com). In the Western European markets rank second collectively in mobile phone prevalence, with roughly 80% of all households owning a wireless handset (http://www.ipsosinsight.com). And in North America, prevalence of mobile ownership is slightly less robust: in the U.S., three in four households own a mobile phone, while just over 60% of Canadian households own a mobile phone today. Interestingly, the popularity of mobile phones is a result of the communication and flexibility that they facilitate and the personal safety issues that they overcome. Interestingly, the design of mobile phones is also evolving, with more functions being added to an increasingly “miniaturized” handset. In themselves, miniaturization and increased function might make the phone more difficult to use while driving as more concentration may be required. The Cellular telephone (commonly "mobile phone" or "cell phone" or "hand phone") is a long-range, portable electronic device used for mobile communication.

**Trends in Mobile Phone Usage Worldwide:**

As per the research findings from the eMarketer website it is found that 4.55 billion people worldwide to use a mobile phone in 2014. However, the Mobile adoption is slowing, but new users in the developing regions of Asia-Pacific and the Middle East and Africa will drive further increases. In the years 2013 and 2017, mobile phone penetration will rise from 61.1% to 69.4% of the global population, according this report.



It was noted that Mobile phone users are rapidly switching over to smartphones as devices become more affordable and 3G and 4G networks advance.

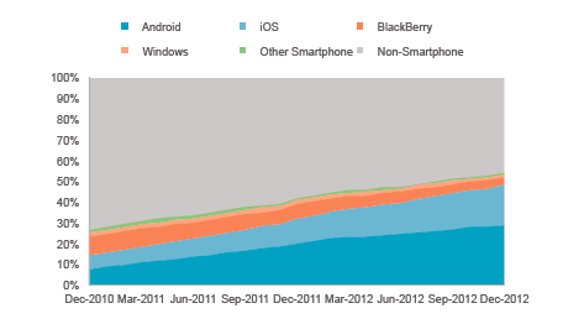
In this research we have made deep study in knowing the brand preferences of student community in the metro city like Hyderabad in India. Initially we found that Nokia is the most favorite brand of thecollege student and 35% student change their mobile phones within 1to2 years and 30% students are using the mobile phones since last 1 to 2 years. 51% students are ready to pay for a mobile phone less than 10,000 and they spend according to their family income. 49% students like the Nokia advertisement most. Mostly students use the mobile phones for talking, SMS and for using the GPRS function. Mostly students have hands free, Bluetooth and memory card. Interestingly, almost all students are aware about the GPRS, Bluetooth and MMS service but least students are aware about the 3G function. The most favorite brand among the college students is Nokia and the least favorite brand is LG. The parameters like appearance, Price, Brand Image and advertisement are the important factors for the students while purchasing mobile phones. Many students prefer slim, medium in weight and large in size handset and influenced by the advertisements on television. There is a slight problem with Nokia as it's hanging service is not so good. The students of Hyderabad are expecting from Nokia that it should provide better service and try to solve the hanging problem. These students wish that cellular companies should increase the awareness about the 3G service. Also they are looking at companies should offer more range of Rs. 10,000 or less than 10,000. It was found that LG and Samsung should try to expand its market share and also should try to increase the awareness through the television advertisement. Finally, students opined that all companies should increase their distribution channels for better reach to the consumers. The researcher of this paper recommends the future researchers to keep on conduct similar research in many other cites of India to get more desirable results, whereby it could be used by marketers of various competing companies in ever growing market of India.

**HISTORY OF THE SMART PHONE :**

Since 1996, when Nokia launched its product, the Nokia 9000, which was the first smartphone, the smartphone market has been grown rapidly. The Nokia 9000 Communicator was a combination of an HP-made PDA and a Nokia-made traditional phone. Next, Nokia released the Communicator 9210, which was equipped with the first color screen and open operating system. The 9500 model was the first camera phone by Nokia and was a genuine mobile Internet phone with access to the Internet via WI-FI4. Since then, Nokia has led the global mobile phone market by combining its competitive cell phone and Symbian, which is an operating system5 developed by Nokia.

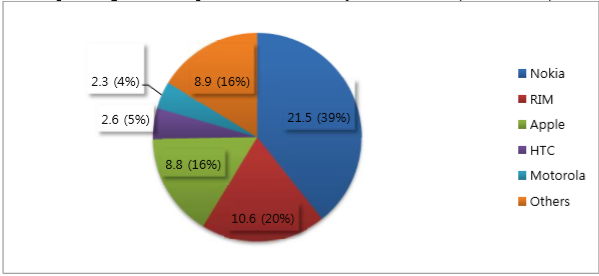
**Market growth:**

While the global traditional phone market faces an ongoing recession due to the global economic downturn, the smartphone market continues its rapid growth. Since 2007, a portion of the smart phone market for the global cell phone industry is expanding in terms of sales volume as well as total sales. According to research conducted by Credit Suisse in December, 2007, while the rate of global cell phone market growth would repeatedly fluctuate up and down during the following four years, the smartphone market would experience tremendous growth. A striking point is that it was predicted that the smartphone market would make up nearly 50% of the entire cell phone market in 2010 in terms of total sales.



**strategy of players in the first competitive market:**

According to the research in 2010, Nokia is the leading company in manufacturing. Nokia had developed its own operating system, and it released the first form of the smartphone before popularized. Nokia holds first place in the smartphone market with a share of 39%.



**The illustration of major players in the current global smartphone market and their strategic options:**

**EXPERIMENTAL SETUP AND RESULTS**

OPERATING SYSTEM: Windows 10

SOFTWARE USED: Anaconda

LANGUAGE: Python

TOOLS: Jupyter Notebook

FILE TYPE: .CSV

To perform analysis all the Tools and softwaare given above needs to be installed and all the datasets needs to be downloaded.

OBJECTIVE

Smartphones are the class of mobile phones and of multiple purpose mobile computing device.

* The aim of this project is to showcase the better brand by comparing them with each other.
* By comparing the brand we can conclude that what are the features which makes that product successful.
* The smartphones of future are expected to be more closely embedded in our day to day lives than ever before.
* In future, if a company launching a new smartphone it can be predicted that which features can be embedded to improve the sales of the product. And also can be predicted that what is the interest of the customers towards the smartphones.

**Flow Of Project**

**System Specification:**

* **Python**

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed. Python has been built with extraordinary Python libraries that are used in Big Data every day for solving problems. which are following

Following is the flow of the project:

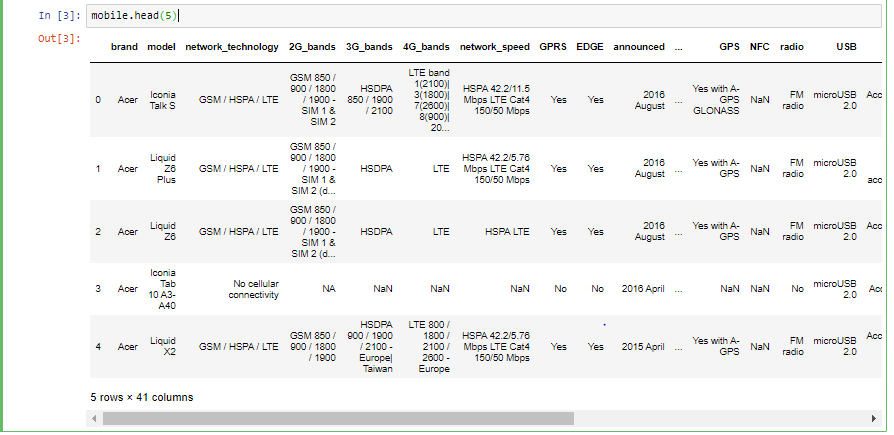
**Data collection**

Data used in this project is a set of smartphone brands and their specification. Each brand has its own 1)product id 2)product name 3) product processor 4) product colour 5) brand.

**Syntex:** to import dataset

mobile=pd.read\_csv('phone\_dataset.csv')

mobile.head(5)



Data Cleaning:

Data cleansing or data cleaning is the process of identifying and removing (or correcting) inaccurate records from a dataset, table, or database and refers to recognising unfinished, unreliable, inaccurate or non-relevant parts of the data and then restoring, remodelling, or removing the dirty or crude data.

Data cleaning techniques may be performed as batch processing through scripting or interactively with data cleansing tools.

After cleaning, a dataset should be uniform with other related datasets in the operation. The discrepancies identified or eliminated may have been basically caused by user entry mistakes, by corruption in storage or transmission, or by various data dictionary descriptions of similar items in various stores.

1. Missing Data:

Dealing with [missing data](https://en.wikipedia.org/wiki/Missing_data)/value is one of the most tricky but common parts of data cleaning. While many models can live with other problems of the data, most models don’t accept missing data.

1. Irregular Data (Outliers):

[Outliers](https://en.wikipedia.org/wiki/Outlier) are data that is distinctively different from other observations. They could be real outliers or mistakes.

**Data Preprocessing**

The data preprocessing unit is responsible for preparing a data for further processing. Classification will be based on sales of a particular product.

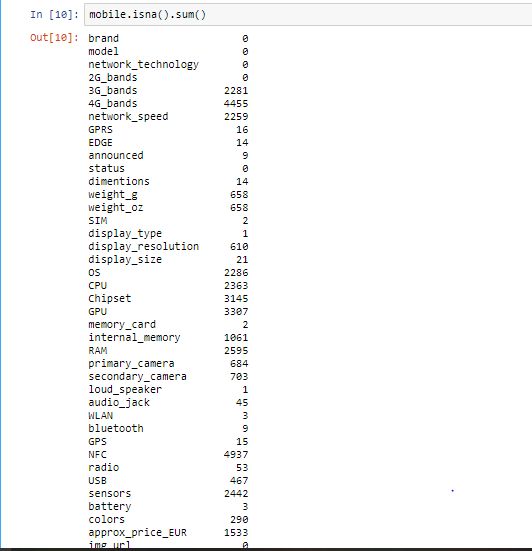
In the this I performed Exploratory Data Analysis (EDA).

## **What is Exploratory Data Analysis?**

Exploratory Data Analysis or (EDA) is understanding the data sets by summarizing their main characteristics often plotting them visually. This step is very important especially when we arrive at modeling the data in order to apply Machine learning. Plotting in EDA consists of Histograms, Box plot, Scatter plot and many more. It often takes much time to explore the data. Through the process of EDA, we can ask to define the problem statement or definition on our data set which is very important.

## **How to perform Exploratory Data Analysis?**

This is one such question that everyone is keen on knowing the answer. Well, the answer is it depends on the data set that you are working. There is no one method or common methods in order to perform EDA, whereas in this tutorial you can understand some common methods and plots that would be used in the EDA process.



HeatMap:



The above HeatMap shows amount of NA values in the Dataset. So I removed those NA values using Drop method.

**Converting String values into Integers:**

I used following commands to convert string values into Integers:

mobile['approx\_price\_EUR']=mobile['approx\_price\_EUR'].astype(np.int64)

mobile['weight\_g']=mobile['weight\_g'].astype(str)

mobile['weight\_oz']=mobile['weight\_oz'].astype(str)

mobile['RAM']=mobile['RAM'].astype(np.int64)

mobile['EDGE']=mobile['EDGE'].astype(np.int64)

mobile['loud\_speaker']=mobile['loud\_speaker'].astype(np.int64)

mobile['audio\_jack']=mobile['audio\_jack'].astype(np.int64)

mobile['OS']=mobile['OS'].astype(np.int64)

mobile['internal\_memory']=mobile['internal\_memory'].astype(np.int64)

mobile['status']=mobile['status'].astype(np.int64)

mobile['radio']=mobile['radio'].astype(np.int64)

**Imputing missing values to the required columns:**

mobile['network\_speed'].fillna(mobile['network\_speed'].value\_counts().index[0],inplace = True)

mobile['OS'].fillna(mobile['OS'].mode()[0],inplace = True)

mobile['RAM'].fillna(mobile['RAM'].mode()[0],inplace = True)

mobile['sensors'].fillna(mobile['sensors'].mode()[0],inplace = True)

mobile['approx\_price\_EUR'].fillna(mobile['approx\_price\_EUR'].mode()[0],inplace = True)

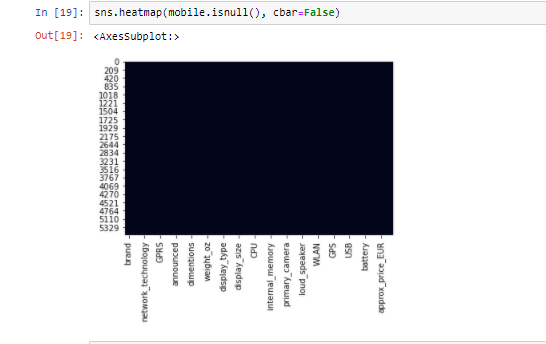
mobile['internal\_memory'].fillna(mobile['internal\_memory'].mode()[0],inplace = True)

mobile['CPU'].fillna(mobile['CPU'].mode()[0],inplace = True)

mobile['EDGE'].fillna(mobile['EDGE'].mode()[0],inplace = True)

mobile['status'].fillna(mobile['status'].mode()[0],inplace = True)

HeatMap After cleaning the data:



Types of Datatypes available in the dataset:



**Correlation**

**Correlation analysis** is a statistical method used to evaluate the strength of relationship between two quantitative variables. A high **correlation** means that two or more variables have a strong relationship with each other, while a weak **correlation** means that the variables are hardly related.

Syntex:

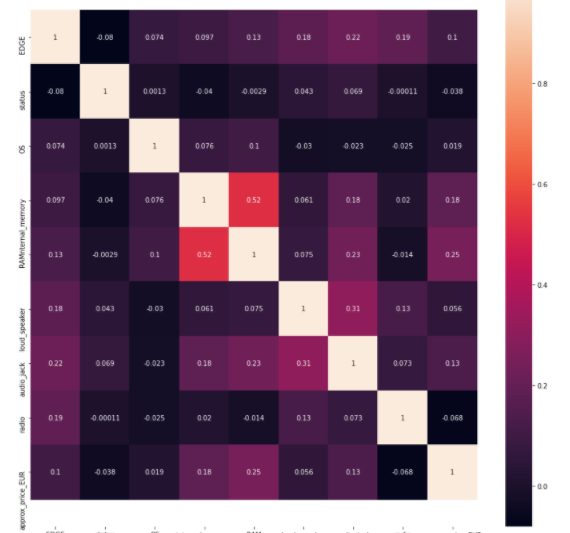
corr=mobile.corr()

fig, ax = plt.subplots(figsize=(15,15))

ax = sns.heatmap(corr , xticklabels=corr.columns , yticklabels=corr.columns , annot=True)

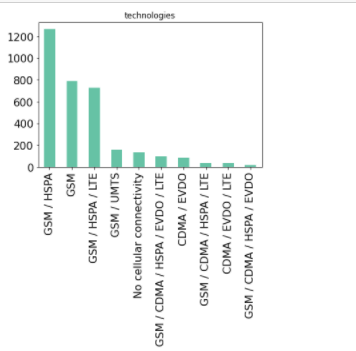
bottom, top = ax.get\_ylim()

ax.set\_ylim(bottom + 0.5, top - 0.5);

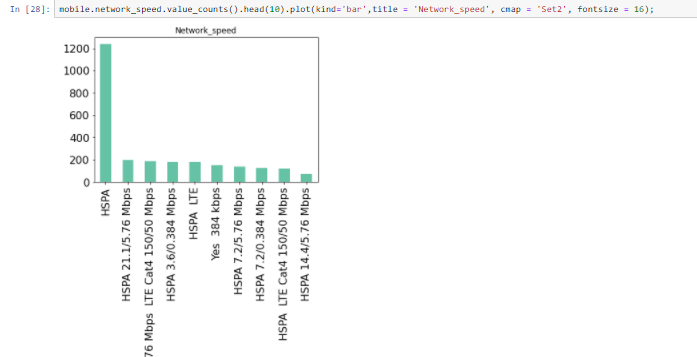


**Data Visualization:**

**Data visualization** refers to the techniques used to communicate **data** or information by encoding it as visual objects (e.g., points, lines or bars) contained in graphics. The goal is to communicate information clearly and efficiently to users. It is one of the steps in **data** analysis or **data science**.

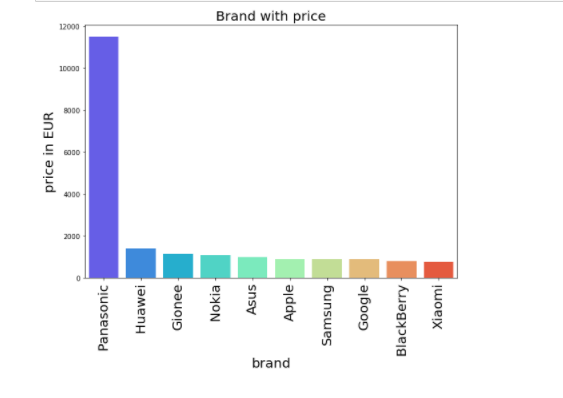


In the above graph we can see that mostly GSM/HSPA technologies are used in smartphones.



In the above graph HSPA technology gives maximum speed.

Comparison between brand and price.



Above graph shows that Panasonic brand has highest price as compare to others.

After processing the dataset now data is ready for modelling.

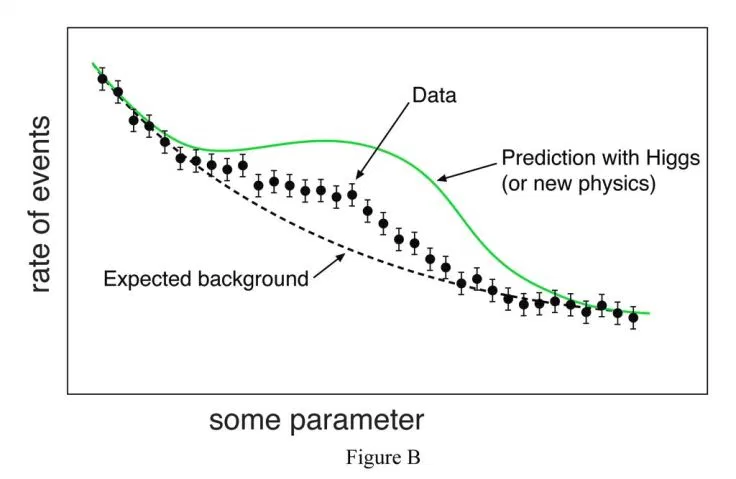
**Removing Outliers:**

Outliers are extreme values that deviate from other observations on data , they may indicate a variability in a measurement, experimental errors or a novelty. In other words, an outlier is an observation that diverges from an overall pattern on a sample.

## Types of outliers

Outliers can be of two kinds: **univariate** and **multivariate**. Univariate outliers can be found when looking at a distribution of values in a single feature space. Multivariate outliers can be found in a n-dimensional space (of n-features). Looking at distributions in n-dimensional spaces can be very difficult for the human brain, that is why we need to train a model to do it for us.

Outliers can also come in different flavours, depending on the environment: **point outliers**, **contextual outliers**, or **collective outliers**. Point outliers are single data points that lay far from the rest of the distribution. Contextual outliers can be noise in data, such as punctuation symbols when realizing text analysis or background noise signal when doing speech recognition. Collective outliers can be subsets of novelties in data such as a signal that may indicate the discovery of new phenomena (As in figure B).



**Model Building** :

The **model building** process involves setting up ways of collecting **data**, understanding and paying attention to what is important in the **data** to answer the questions you are asking, finding a statistical, mathematical or a simulation **model** to gain understanding and make predictions.

Linear Regression:

In statistics, **linear regression** is a **linear** approach to modeling the relationship between a scalar response (or dependent variable) and one or more explanatory variables (or independent variables). The case of one explanatory variable is called simple **linear regression**.

In linear regression, the relationships are modeled using linear predictor functions whose unknown model parameters are estimated from the data. Such models are called linear models.[[3]](https://en.wikipedia.org/wiki/Linear_regression#cite_note-3) Most commonly, the conditional mean of the response given the values of the explanatory variables (or predictors) is assumed to be an affine function of those values; less commonly, the conditional median or some other quantile is used. Like all forms of regression analysis, linear regression focuses on the conditional probability distribution of the response given the values of the predictors, rather than on the joint probability distribution of all of these variables, which is the domain of multivariate analysis.

Linear regression was the first type of regression analysis to be studied rigorously, and to be used extensively in practical applications.[[4]](https://en.wikipedia.org/wiki/Linear_regression#cite_note-4) This is because models which depend linearly on their unknown parameters are easier to fit than models which are non-linearly related to their parameters and because the statistical properties of the resulting estimators are easier to determine.

Syntex:

from sklearn import linear\_model

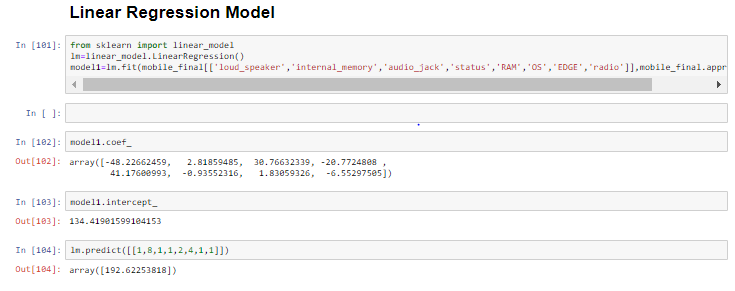
lm=linear\_model.LinearRegression()

model1=lm.fit(mobile\_final[['loud\_speaker','internal\_memory','audio\_jack','status','RAM','OS','EDGE','radio']],mobile\_final.approx\_price\_EUR)

model1.coef\_

model1.intercept\_

lm.predict([[1,8,1,1,2,4,1,1]])



## Logistic Function

Logistic regression is named for the function used at the core of the method, the logistic function.

The [logistic function](https://en.wikipedia.org/wiki/Logistic_function), also called the sigmoid function was developed by statisticians to describe properties of population growth in ecology, rising quickly and maxing out at the carrying capacity of the environment. It’s an S-shaped curve that can take any real-valued number and map it into a value between 0 and 1, but never exactly at those limits.

1 / (1 + e^-value)

Where e is the [base of the natural logarithms](https://en.wikipedia.org/wiki/E_(mathematical_constant)) (Euler’s number or the EXP() function in your spreadsheet) and value is the actual numerical value that you want to transform. Below is a plot of the numbers between -5 and 5 transformed into the range 0 and 1 using the logistic function.



Logistic Function

Now that we know what the logistic function is, let’s see how it is used in logistic regression.

Syntex:

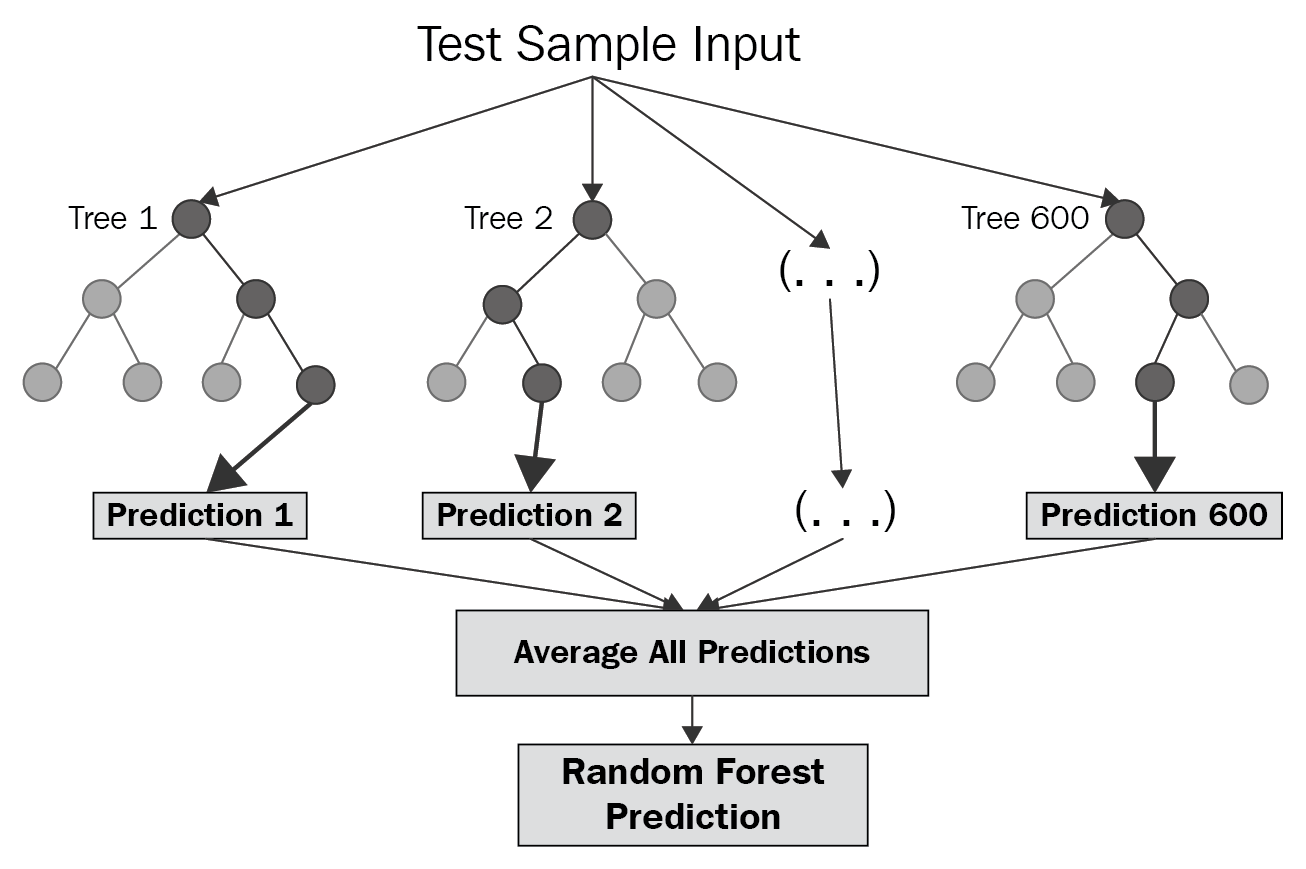
from sklearn.linear\_model import LogisticRegression

model2=LogisticRegression()

model2.fit(mobile\_final[['loud\_speaker','internal\_memory','audio\_jack','status','RAM','OS','EDGE','radio']],mobile\_final.approx\_price\_EUR)

model2.predict([[1,8,1,1,2,4,1,1]])

## ****Random Forest****



Random Forest Structure

Random forest is a **Supervised Learning algorithm** which uses ensemble learning method for **classification and regression**.

**Random forest** is a **bagging** technique and **not a boosting** technique. The trees in **random forests** are run in parallel. There is no interaction between these trees while building the trees.

It operates by constructing a multitude of decision trees at training time and outputting the class that is the **mode** of the **classes (classification)** or **mean prediction (regression)** of the individual trees.

A random forest is a meta-estimator (i.e. it combines the result of multiple predictions) which **aggregates many decision trees**, with some helpful modifications:

Syntex:

from sklearn.ensemble import RandomForestRegressor

rfreg=RandomForestRegressor(n\_estimators=10,random\_state=0)

rfreg.fit(mobile\_final[['loud\_speaker','internal\_memory','audio\_jack','status','RAM','OS','EDGE','radio']],mobile\_final.approx\_price\_EUR)

rf\_pred=rfreg.predict((x\_test))

from sklearn import metrics

r\_square=metrics.r2\_score(y\_test,rf\_pred)

print('Rsquared error associated with random forest regression is:',r\_square)

price\_pred=rfreg.predict([[1,8,1,1,2,4,1,1]])

price\_pred

Program code:

get\_ipython().run\_line\_magic('matplotlib', 'inline')

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

mobile=pd.read\_csv('phone\_dataset.csv')

mobile.head(5)

mobile.shape

mobile.dtypes

mobile.describe()

mobile.info()

mobile.nunique()

#Keeping a copy of the original data

mobile\_original = mobile.copy()

mobile\_original.head()

mobile.isna().sum()

sns.heatmap(mobile.isnull(), cbar=False)

plt.show();

mobile['EDGE']= mobile['EDGE'].map({'Yes':1, 'No':0})

mobile['RAM']= mobile['RAM'].map({'1 GB RAM':1, '2 GB RAM':2,'3 GB RAM':3})

mobile['audio\_jack']= mobile['audio\_jack'].map({'Yes':1, 'No':0})

mobile['radio']= mobile['radio'].map({'FM radio':1,'Stereo FM radio':2,'Stereo FM radio| RDS':3,'To be confirmed':4,'FM radio| RDS':5, 'No':0})

mobile['internal\_memory']= mobile['internal\_memory'].map({'4 GB':4,'8 GB':8,'16 GB':16,'32 GB':32,'512 GB':512})

mobile['status']= mobile['status'].map({'Available':1,'Discontinued':0,'Available. Released 2015 August':1,'Available. Released 2014 May ':1,'Available. Released 2013 October':1})

mobile['OS']= mobile['OS'].map({'Android 4.4.2 (KitKat)':4.4,'Android 2.2 (Froyo)':2.2,'Android 5.1 (Lollipop)':5.1,'Android 4.2 (Jelly Bean)':4.2,'Microsoft Windows Mobile 6.0 Professional':6.0})

mobile['loud\_speaker']= mobile['loud\_speaker'].map({'Yes':1,'Yes with stereo speakers':1, 'No':0})

mobile['network\_speed'].fillna(mobile['network\_speed'].value\_counts().index[0],inplace = True)

mobile['OS'].fillna(mobile['OS'].mode()[0],inplace = True)

mobile['RAM'].fillna(mobile['RAM'].mode()[0],inplace = True)

mobile['sensors'].fillna(mobile['sensors'].mode()[0],inplace = True)

mobile['approx\_price\_EUR'].fillna(mobile['approx\_price\_EUR'].mode()[0],inplace = True)

mobile['internal\_memory'].fillna(mobile['internal\_memory'].mode()[0],inplace = True)

mobile['CPU'].fillna(mobile['CPU'].mode()[0],inplace = True)

mobile['EDGE'].fillna(mobile['EDGE'].mode()[0],inplace = True)

mobile['status'].fillna(mobile['status'].mode()[0],inplace = True)

mobile.head(5)

mobile.drop([ '2G\_bands','3G\_bands','4G\_bands','NFC','GPU','Chipset'], axis = 1, inplace=True)

mobile.isna().sum()

#droping NA values

mobile.drop(mobile.columns[mobile.columns.str.contains('unnamed',case=False)],axis=1,inplace=True)

mobile.dropna(axis=0,inplace=True)

mobile.isna().sum()

sns.heatmap(mobile.isnull(), cbar=False)

mobile.columns

mobile['approx\_price\_EUR']=mobile['approx\_price\_EUR'].astype(np.int64)

mobile['weight\_g']=mobile['weight\_g'].astype(str)

mobile['weight\_oz']=mobile['weight\_oz'].astype(str)

mobile['RAM']=mobile['RAM'].astype(np.int64)

mobile['EDGE']=mobile['EDGE'].astype(np.int64)

mobile['loud\_speaker']=mobile['loud\_speaker'].astype(np.int64)

mobile['audio\_jack']=mobile['audio\_jack'].astype(np.int64)

mobile['OS']=mobile['OS'].astype(np.int64)

mobile['internal\_memory']=mobile['internal\_memory'].astype(np.int64)

mobile['status']=mobile['status'].astype(np.int64)

mobile['radio']=mobile['radio'].astype(np.int64)

mobile.dtypes

corr=mobile.corr()

fig, ax = plt.subplots(figsize=(15,15))

ax = sns.heatmap(corr , xticklabels=corr.columns , yticklabels=corr.columns , annot=True)

bottom, top = ax.get\_ylim()

ax.set\_ylim(bottom + 0.5, top - 0.5);

**# # Visualization**

**# # 1.Network\_Technologies**

mobile.network\_technology.value\_counts()

mobile.network\_technology.value\_counts().head(10).plot(kind='bar',title = 'technologies', cmap = 'Set2', fontsize = 16);

**# # 2. Network\_speed**

mobile.network\_speed.value\_counts()

mobile.network\_speed.value\_counts().head(10).plot(kind='bar',title = 'Network\_speed', cmap = 'Set2', fontsize = 16);

**# # 3.GPRS**

mobile.GPRS.value\_counts()

mobile.GPRS.value\_counts().head(10).plot(kind='bar',title = 'GPRS', cmap = 'Set2', fontsize = 16);

**# # 4.EDGE**

mobile.EDGE.value\_counts().head(10).plot(kind='bar',title = 'EDGE', cmap = 'Set2', fontsize = 16);

**# # 5.Status**

mobile.status.value\_counts()

mobile.status.value\_counts().head(10).plot(kind='bar',title = 'Availability', cmap = 'Set2', fontsize = 16);

**# # 6.Announcement**

mobile.announced.value\_counts()

mobile.announced.value\_counts().head(10).plot(kind='bar',title = 'date of announcement', cmap = 'Set2', fontsize = 16);

**# # 7.Dimensions**

mobile.dimentions.value\_counts()

mobile.dimentions.value\_counts().head(10).plot(kind='bar',title = 'dimentions of model', cmap = 'Set2', fontsize = 16);

**# # 8.Weight**

mobile.weight\_g.value\_counts()

mobile.weight\_g.value\_counts().head(10).plot(kind='bar',title = 'weight of model', cmap = 'Set2', fontsize = 16);

**# # 9.SIM**

mobile.SIM.value\_counts()

mobile.SIM.value\_counts().head(10).plot(kind='bar',title = 'SIM slot types', cmap = 'Set2', fontsize = 16);

**# # 10.Internal\_memory**

mobile.internal\_memory.value\_counts()

mobile.internal\_memory.value\_counts().head(10).plot(kind='bar',title = 'memory storage', cmap = 'Set2', fontsize = 16);

**# # 11.RAM**

mobile.RAM.value\_counts()

mobile.RAM.value\_counts().head(10).plot(kind='bar',title = 'RAM', cmap = 'Set2', fontsize = 16);

**# # 12.Primary Camera**

mobile.primary\_camera.value\_counts()

mobile.primary\_camera.value\_counts().head(10).plot(kind='bar',title = 'Camera type', cmap = 'Set2', fontsize = 16);

**# # 13.Secondary Camera**

mobile.secondary\_camera.value\_counts()

mobile.secondary\_camera.value\_counts().head(10).plot(kind='bar',title = 'Camera type', cmap = 'Set2', fontsize = 16);

**# # 14.Loud Speaker**

mobile.loud\_speaker.value\_counts()

mobile.loud\_speaker.value\_counts().head(10).plot(kind='bar',title = 'loudspeaker is avialable or not', cmap = 'Set2', fontsize = 16);

**# # 15.Audio Jack**

mobile.audio\_jack.value\_counts()

mobile.audio\_jack.value\_counts().head(10).plot(kind='bar',title = 'audio jack is there or not', cmap = 'Set2', fontsize = 16);

**# # 16.WLAN**

mobile.WLAN.value\_counts()

mobile.WLAN.value\_counts().head(10).plot(kind='bar',title = 'wireless connectivity', cmap = 'Set2', fontsize = 16);

**# # 17.Bluetooth**

mobile.bluetooth.value\_counts()

mobile.bluetooth.value\_counts().head(10).plot(kind='bar',title = 'Bluetooth', cmap = 'Set2', fontsize = 16);

**# # 18.GPS**

mobile.GPS.value\_counts()

mobile.GPS.value\_counts().head(10).plot(kind='bar',title = 'GPS', cmap = 'Set2', fontsize = 16);

**# # 19.Radio**

mobile.radio.value\_counts()

mobile.radio.value\_counts().head(10).plot(kind='bar',title = 'radio type', cmap = 'Set2', fontsize = 16);

**# # 20.USB**

mobile.USB.value\_counts()

mobile.USB.value\_counts().head(10).plot(kind='bar',title = 'USB Connectivity', cmap = 'Set2', fontsize = 16);

**# # 21.Sensors**

mobile.sensors.value\_counts()

mobile.sensors.value\_counts().head(10).plot(kind='bar',title = 'Sensors available', cmap = 'Set2', fontsize = 16);

**# # 22.Battery**

mobile.battery.value\_counts()

mobile.battery.value\_counts().head(10).plot(kind='bar',title = 'Sensors available', cmap = 'Set2', fontsize = 16);

**# # 23.Colors**

mobile.colors.value\_counts()

mobile.colors.value\_counts().head(10).plot(kind='bar',title = 'Sensors available', cmap = 'Set2', fontsize = 16);

**# # 24.Price of product**

mobile.approx\_price\_EUR.value\_counts()

mobile.approx\_price\_EUR.value\_counts().head(10).plot(kind='bar',title = 'price of model in Euroes', cmap = 'Set2', fontsize = 16);

**# # 25.Brands**

mobile.brand.value\_counts()

mobile.brand.value\_counts().head(10).plot(kind='bar',title = 'price of model in Euroes', cmap = 'Set2', fontsize = 16);

mobile.hist(figsize=(15,15), xrot=-45); ## Display the labels rotated by 45 degress

plt.show();

**# # Brand with there price**

plot\_gross = pd.DataFrame(mobile.groupby(['brand'])['approx\_price\_EUR'].max().sort\_values(ascending=False)[:10]).reset\_index()

plot\_gross = plot\_gross[plot\_gross['approx\_price\_EUR'] > 0]

plt.subplots(figsize=(10,7))

sns.barplot(x='brand',y='approx\_price\_EUR',data=plot\_gross,palette='rainbow', )

plt.ylabel('price in EUR', fontsize=20)

plt.xticks(rotation=90,fontsize=20)

plt.xlabel('brand', fontsize=20)

plt.title('Brand with price', fontsize=20)

plt.show()

**# # Brand with RAM capacity**

plot\_gross = pd.DataFrame(mobile.groupby(['brand'])['RAM'].max().sort\_values(ascending=False)[:10]).reset\_index()

plot\_gross = plot\_gross[plot\_gross['RAM'] > 0]

plt.subplots(figsize=(10,7))

sns.barplot(x='brand',y='RAM',data=plot\_gross,palette='rainbow', )

plt.ylabel('RAM', fontsize=20)

plt.xticks(rotation=90,fontsize=20)

plt.xlabel('brand', fontsize=20)

plt.title('Brand comparing with RAM', fontsize=20)

plt.show()

**# # price comparing with RAM**

sns.catplot(x="RAM", y="approx\_price\_EUR",hue='EDGE', kind="bar", data=mobile, palette='rainbow')

plt.title('price with RAM', fontsize=20);

**# # comparing internal memory with EDGE**

sns.catplot(x="internal\_memory", y="EDGE",hue='OS', kind="bar", data=mobile, palette='rainbow')

plt.title('internal memory with EDGE', fontsize=20)

mobile.info()

mobile\_final = mobile.drop(["brand","sensors","memory\_card","dimentions","SIM","model","network\_speed","network\_technology","GPRS","announced","weight\_g","weight\_oz","display\_type","display\_resolution","display\_size","CPU","primary\_camera","secondary\_camera","WLAN","bluetooth","GPS","USB","colors","battery","img\_url"], axis=1)

print(mobile\_final.head())

print(mobile\_final.shape)

x = mobile\_final.drop("approx\_price\_EUR", axis=1) #independent columns

y = mobile\_final["approx\_price\_EUR"] #target column i.e PRICE

from sklearn.feature\_selection import SelectKBest

from sklearn.feature\_selection import chi2

from sklearn.feature\_selection import f\_classif

#apply SelectKBest class to extract top 17 best features

bestfeatures = SelectKBest(score\_func=f\_classif , k=8)

fit = bestfeatures.fit(x,y)

dfscores = pd.DataFrame(fit.scores\_)

dfcolumns = pd.DataFrame(x.columns)

**#concat two dataframes for better visualization**

featureScores = pd.concat([dfcolumns,dfscores],axis=1)

featureScores.columns = ['Features','Score'] #naming the dataframe columns

print(featureScores.nlargest(8,"Score"))

from sklearn.ensemble import ExtraTreesClassifier

import matplotlib.pyplot as plt

model = ExtraTreesClassifier()

feature = model.fit(x,y)

print(model.feature\_importances\_) #use inbuilt class feature\_importances of tree based classifiers

#plot graph of feature importances for better visualization

feat\_importances = pd.Series(model.feature\_importances\_, index=x.columns)

feat\_importances.nlargest(9).plot(kind='barh')

plt.show()

**# Lets Divide the Data in Train and Test Set**

from sklearn.model\_selection import train\_test\_split

x\_train, x\_test, y\_train, y\_test = train\_test\_split(x,y, test\_size=0.2,random\_state=0)

mobile\_final.plot(kind='box')

mobile\_final.plot(kind='hist')

**# # Removing Outliers**

lower\_bound=0.1

upper\_bound=0.95

res=mobile\_final.approx\_price\_EUR.quantile([lower\_bound,upper\_bound])

res

true\_index=(res.loc[lower\_bound]<mobile\_final.approx\_price\_EUR.values)& (mobile\_final.approx\_price\_EUR.values<res.loc[upper\_bound])

true\_index

false\_index=~true\_index

false\_index

mobile\_final.approx\_price\_EUR[true\_index]

mid=np.median(mobile\_final.approx\_price\_EUR[true\_index])

mid

mobile\_final[false\_index]=mid

mobile\_final

mobile\_final.plot(kind='box')

**# # Linear Regression Model**

from sklearn import linear\_model

lm=linear\_model.LinearRegression()

model1=lm.fit(mobile\_final[['loud\_speaker','internal\_memory','audio\_jack','status','RAM','OS','EDGE','radio']],mobile\_final.approx\_price\_EUR)

model1.coef\_

model1.intercept\_

lm.predict([[1,8,1,1,2,4,1,1]])

**# # Logistic Regression**

from sklearn.linear\_model import LogisticRegression

model2=LogisticRegression()

model2.fit(mobile\_final[['loud\_speaker','internal\_memory','audio\_jack','status','RAM','OS','EDGE','radio']],mobile\_final.approx\_price\_EUR)

model2.predict([[1,8,1,1,2,4,1,1]])

**# # RandomForest Regressor**

from sklearn.ensemble import RandomForestRegressor

rfreg=RandomForestRegressor(n\_estimators=10,random\_state=0)

rfreg.fit(mobile\_final[['loud\_speaker','internal\_memory','audio\_jack','status','RAM','OS','EDGE','radio']],mobile\_final.approx\_price\_EUR)

rf\_pred=rfreg.predict((x\_test))

from sklearn import metrics

r\_square=metrics.r2\_score(y\_test,rf\_pred)

print('Rsquared error associated with random forest regression is:',r\_square)

price\_pred=rfreg.predict([[1,8,1,1,2,4,1,1]])

price\_pred

import pickle

pickle.dump(model1, open('linreg.pkl','wb'))

REFERENCES

1. A study comparing table-based and list-based smartphone interface usability By Patrick Finley
2. <https://towardsdatascience.com/a-brief-overview-of-outlier-detection-techniques-1e0b2c19e561> Outlier detection.
3. <https://machinelearningmastery.com/logistic-regression-for-machine-learning/> Logistic regression.
4. <http://www.eiu.com/industry/article/782421662/> the smart growth in indias phone market /2014-10-27.
5. <https://machinelearningmastery.com/linear-regression-for-machine-learning/> Linear regression.
6. <https://www.datacamp.com/community/tutorials/exploratory-data-analysis-python> exploratory data Analysis
7. <https://www.geeksforgeeks.org/data-analysis-visualization-python/> Data visualization