A

Project Report On

Predictive Models Based On Supervised Machine Learning

Submitted in partial fulfilment of the requirements for the award of the degree of

Bachelor Of Technology In
Computer Science And Engineering

By

Abhishek Ray	(1783910901)
Dhiraj Gupta	(1783910903)
Umeshchnadra Yadav	(1783910909)

Under The Supervision of

Dr. Vivek Srivastava

Department of Computer Science & Engineering
Rajkiya Engineering College

Kannauj, Uttar Pradesh

Affiliated to

Dr. A.P.J. Abdul Kalam Technical University
Lucknow, Uttar Pradesh

Rajkiya Engineering College,Kannauj
Kannauj-209732(UP)
Affiliated to
Dr. A.P.J. AKTU, Lucknow
AKTU Code 839
Website:- http://reck.ac.in/

CERTIFICATE

This is to certify that the Project report entitled
"Predictive Models Based On Supervised Machine Learning"
submitted by **Abhishek Ray, Dhiraj Gupta & Umeshchandra Yadav**
To the, Department of Computer Science & Engineering of
Rajkiya Engineering College Kannauj, Uttar Pradesh
Affiliated to Dr. A.P.J. Abdul Kalam Technical
University, Lucknow, Uttar Pradesh in partial fulfillment for the award of Degree of Bachelor of Technology in Computer science & Engineering is a bonafide record of the project work carried out by them under my supervision during the year 2019-2020.

Dr. Vivek Srivastava	**Dr . BDK Patro**
Assistant Professor	**Associate Professor & Head**
DEPT. OF CSE	**DEPT. OF CSE**

ACKNOWLEDGEMENT

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. We would like to extend my sincere thanks to all of them.

We are highly indebted to **Dr.Vivek Srivastava** for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.

We are extremely indebted to **Dr. BDK Patro**, the HOD of Department of Computer Science and Engineering at RECK for his valuable suggestions and constant support throughout my project tenure. We would like to express our thanks to all faculty and Staff members of Department of Computer Science and Engineering, RECK for their support in completing this project on time.

We also express gratitude towards our parents for their kind co-operation and encouragement which help me in completion of this project. Our thanks and appreciations also go to our friends in developing the project and people who have willingly helped me out with their abilities.

Abhishek Ray
Dhiraj Gupta
Umeshchandra Yadav

ABSTRACT

Predictive analytics includes many of statistical techniques from machine learning, predictive modeling, and data mining, to analyze current and historical data facts to make predictions about future or unknown events. The use of predictive analytics in marketing, financial services, insurance, travel, mobility, telecommunications, healthcare, Social Networking and other fields.

Predictive modeling is the general concept of building a model that is capable of making predictions.

In this project we attempt to implement a Machine Learning approach to predict Diamond Prices, Gold Prices, Stock Prices, and Covid-19 India dataset. We collected dataset from the various Source like Kaggle.com, yahoo finance, etc. and Some library like Scikit-learn for ML algorithms, pandas for analysis, matplotlib & Seaborn for visualization. Then we Analyse data and predict data in the result.

In Diamond Prices Prediction, the aim is to predict the price of diamond using its cut, price, color, clarity, size. We making prediction using MLR, KNN, LASSO algorithms then also calculate best model for predict price.

In Gold Price we use MLR, KNN Regression for predicting gold price. We fatch the dataset using yfinance api of yahoo finance.

In Stock Market Prediction, the aim is to predict the future value of the financial stocks of a company. We use of RNN and LSTM based Machine learning to predict stock values. Factors considered are open, close, low, high and volume.

In covid-19 prediction we use SVM for prediction of number of cases in Uttar Pradesh India. In SVM we use SVR for prediction of cases how increases.

In all prediction model we try to analyse and understand how the historical data varies and predicting an output.

Keywords – MRL (Multiple Linear Regression)*, Knn (k-nearest neighbors), SVM(Support vector Machine),SVR(Support Vector Regressor),RNN(Recurrent Neural Network).*

Table of Content

CHAPTER: 1

Introduction

We all face the results of predictive models every day. Like we are offered the right incentive to buy a product. We get a text from the bank when something strange is happening to our accounts, And we rarely get to see spam in our inbox. This is thanks to Predictive analytics for the predictions on the data we generate daily.

In this project we attempt to implement Supervised Machine Learning approach to predict Diamond Prices, Gold Prices, Stock Prices, and Covid-19 India dataset. We collected dataset from the various Source like Kaggle.com, yahoo finance, etc. and Some library like Scikit-learn for ML algorithms, pandas for analysis, matplotlib & Seaborn for visualization. Then we Analyse data and predict data in the result.

1.1 Predictive Analytics

Predictive Analytics are a group of statistical techniques from machine learning, predictive modeling, and data mining, to analyze current and historical data facts to make predictions about future or unknown events.The use of predictive analytics in marketing, financial services, insurance, travel, mobility, telecommunications, healthcare, Social Networking and other fields.[1]

The goal Predictive Analytics is to produce a good assessment of what may happen with unknown events.

Predictive analysis is use in various fields like this Online Retail, Customer Relationship Management (CRM), cyber security, Healthcare, Reduces Risks, Education, Recommendation and search engines, Improvised market campaigning, Fraud Detection, weather forecasting, Social Media Analysis, Government Sector etc.

All predictive analytics applications include three fundamental components:

- **Data:** The performance of all predictive model strongly depends on the quality of the historical (previous) data it processes.
- **Statistical modeling:** Various statistical techniques ranging from basic to complex functions are included for derivation of meaning, insight, and conclusion. Regression is the most commonly used statistical technique.
- **Assumptions:** The conclusions drawn from the data collected and analyzed usually assume that the future will follow a pattern related to the past.

1.1.1 How to Do Predictive Analytics

We have three technique for Predictive Analytics

i. Mathematical Models

A mathematical model is a descrip of a system using mathematical concepts and language. These models are heavily used in the physical sciences and other domains. They are usually logically derived from existing theories.

Although sometime we can create mathematical models from data and this is why we are counting mathematical models as a way of doing predictive analytics.

We have most famous example is when kepler to write the elliptical orbits of the planets using data from astronomical observations.

ii. Statistical Models

A statistical model is a class of mathematical model that tries to model systems that have an element of randomness. Statistical models usually contain parameters that are calculated from data. So data is an important element of almost any statistical models.

- Some examples of statisticalmodels includes :
  - Linear Regression Analysis
  - Time-Series models
  - Spatial models
  - Bayesian models

  many other

iii. Machine Learning

Machine Learning model is a computational or algorithmic approach to extract information from data.

Machine Learning is a sub-field of Computer Science and more specifically of Artificial Intelligence. That develop methods to “give computer the ability to learn how to perform tasks like predictions without being explicitly programmed” .

The Huge amount of data that we have this is the approach that has become really popular for doing predictive analytics and has been really successful.

1.2 Machine Learning

Machine learning is a sub-field of computer science. In general the field can be simply describe as “given the computer the ability to learn without being explicitly programmed”.

Machine learning is one of those branches of computer science in which algorithms (running inside computers) learn from the data available to them. With this learning mechanism, various predictive models can come up.

1.2.1 Types of Machine Learning

Machine learning approaches are divided into three broad categories, depend on the nature of the "signal" available to the learning system:

i. **Supervised Learning:** Supervised learning occurs when the machine is taught or trained using well-labeled data (meaning that some data has already been tagged with the correct answer or classification). Once this is done, the machine is provided with a new set of data or examples, so that this algorithm analyses the training data (set of training examples or data sets) and generate the desired results from the labeled data.

In supervised learning, we have sample about something: for each sample we have a set of feature (attribute, variable) and a target variable or quantity that we would like to predict.
The supervised learning model has a specified target output that is either a classification (label) or a continuous variable. The objective of the supervised learning model is to predict a specified outcome.

Examples of Supervised Learning

We have data about	And we would like to predict
E-mails	Spam or Non-Spam
Financial Data (Stock,Gold .etc)	Stock Price Gold Price
News Article	How many views will get

Table 1.1

We will need data about both the features of the thing that we want to predict and the target which in this case will be the column here.

ii. Unsupervised learning: In unsupervised learning, the training data is unlabelled. It is similar to a system trying to learn without a teacher. In unsupervised learning, the training of the machine or system is done using the information that is neither classified nor labelled. Because of this, the algorithm itself will act on information without any guidance. Here, the machine with this unsorted information tries to recognise similarities, patterns and differences, without any training.[3]

No labels are given to the learning algorithm, leaving it on its own to find structure in its input. It's have doing by **Clustering, Dimensionality Reduction, Anomaly Detection, others.**

"unsupervised learning is a set of tasks where you have data about the thing that you liked analyses but you don't have any target values so you have all features."

Examples of Unsupervised Learning

We have data about	And we would like to predict
Costumers	Find Segments
Credit card transaction	Find consumption patterns
Genomic Data	Find group of genes according to biological function

Table 1.2

iii. Reinforcement learning: A form of Machine Learning in which the algorithm interacts with a dynamic environment in which it must perform a certain goal. The program is provided feedback in terms of reward and punishments as it navigates the problem space.

So in this type of learning we build a software agent or machines that can ultimately determine an ideal, behavior within a specific context.

Some of the most successful applications in this type of learning

- Manufacturing Robot
- Self-Driving Car
- Automatic Trading Software

1.3 Software Used

There are different types of software through which we can work on Machine Learning. However, in this project, we have used Anaconda distribution and Jupyter notebook.

i. Anaconda Distribution

According to the creators of the software anaconda or the Anaconda distribution is a free easy to install package manager, environment manager python distribution and collection of over seven hundred and twenty open source packages with free community support.

So in brief Anaconda is basically a toolbox a ready to use collection of related libraries and tools for doing analytics with Python.

Anaconda is mainly used with Python & R as a Data Science tool for scientific computing.

ii. Jupyter notebook

The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.[2]

It is a great tool for doing analytics because you can't explain what you're doing. You can share your ideas explanations visualizations and most importantly your code.

It comes with the Anaconda's distribution.

1.4 Libraries Used

i. **NumPy** is the fundamental package for doing scientific computing with Python.
   It contains N-dimensional exhibit object (nd array), critic (broadcasting) capacities, and routines for fast operations on array, included mathematical & logical operation, linear algebra, statistics, random simulations & much more.

   It is very important because it gives us a nice and fast way to make operations on arrays in a vectorized fashion.

   all the other libraries that we will use in this project are based on NumPy.

ii. **Pandas** are for information control and investigation. Pandas provided fast, flexible, and expressive data structures designed to work with relational or table-like data (SQL table or Excel spreadsheet / Tabular data).
   It is a fundamental high-level building block for doing practical, real world data analysis in Python & it is designed to integrate very well with other tools in python's Data Science stack like NumPy.

   pandas is well suited for:

- Tabular data with heterogeneously-typed columns, as in an SQL table or Excel spreadsheet
- Ordered and unordered (not necessarily fixed-frequency) time series data.
- Arbitrary matrix data (homogeneously typed or heterogeneous) with row and column labels
- Any other form of observational / statistical data sets. The data actually need not be labeled at all to be placed into a pandas data structure

The two primary data structures of pandas, **Series** (1-D) & **DataFrame** (2-D), handle the vast majority of typical use cases in finance, statistics, social science, and many areas of engineering.

Pandas is built on top of NumPy.

iii. **Matplotlib** is a Python 2D plotting library which produces distribution quality figures in an assortment of printed copy designs and intelligent conditions all through frameworks. Matplotlib can be utilized in Python contents, the Python interpreter and IPython shell, the jupyter notebook, web application , and graphical user interface toolkits.

Matplotlib is primarily written in pure Python, it heavy use of NumPy and other extension code to provide good performance even for large arrays.

iv. **Seaborn** provides a high level interface for drawing attractive Statistical Graphics & it is built on top of matplotlib and tightly integrated with the python data science stack. It includes support for NumPy &Pandas data structures & you can use it along with the statistical routines from SciPy & stats models.

So this is a high level Statistical Graphics library that produces very complexs Visualization with very little code.

v. **Scikit-learn** is one of the most popular Python libraries for doing machine learning. It provides a simple and efficient set of tools for doing data modeling and analysis.

It is built on top of NumPy, SciPy & Matplotlib and it works really nicely with pandas and the data structure of Pandas which are Series & DataFrame .
In scikit-learn we usually follow a fixed set of steps for building a machine learning model:

0. Data preparation

1. Import the estimator object (model)

2. Create an instance of the estimator

3. Use the trainning data to train the estimator

4. Evaluate the model

5. Make predictions

These steps recipe is just a general roadmap that may include several sub-steps, going back and forth between steps, and there are a lot of details that need to be considered in every step. However this is a nice mental model to have.

We should add step 0, which is very important and most time-consuming: “Data preparation”.

1.5 Approach

Predictive analysis is used to predict unknown events or unobserved events by analyzing the existing data set with the help of data mining, statistical modeling, and Machine Learning techniques. For prediction analysis, first the objective is defined, and then the data set is prepared. Based upon the prepared data, a model is laid down for deployment and monitoring.[3] Predictive analysis identifies the cause-effect relationship across the variables from the given data set and discovers hidden patterns with the help of data mining techniques. It may apply observed patterns to unknowns in the past, present or the future.

Predictive models are supervised learning models, which try to predict certain values using the values in the input data set. The learning model establishes a relationship between the target figure, that is, the feature being predicted and the predictor features. The predictive models have a clear focus on what they want to learn and how they want to learn.[3]

The models which are used for the prediction of target features of categorical values are known as classification models. The target feature is known as a class and the categories in which the classes are divided into are known as levels. Some of the popular classification models include KNN and Decision Trees. Predictive models may also be used to predict numerical values of the target feature based on the predictor features. The models which are used for the prediction of the numerical values of the target feature of a data instance are known as regression models. They include, Linear Regression, Support Vector Regression, Lasso, Ridge regression and other forms of regression models.

Predictive analytics is said to be driven by data in the sense that the steps, codes and algorithms used, generate a model from the patterns and characteristics of the data alone. We would be applying data preprocessing techniques to a data set and reduce that data set into training data set and test data set. Later, we will generate algorithms using training data set and apply those to the test data set.

Algorithms derived and driven from data are used in analytics that may include identifying variables to be included in the model, parameters that define the model, load, or model complexity.

Our aim is to predict the Diamond Prices, Gold Price, Stock Market, covid-19 cases by the previous recorded datasets.

For this we will use predictive analysis using Supervised machine learning.

1.6 Motivation

We were motivated to work in the area of prediction analysis field because data science is an inter-disciplinary field where processes and systems are analyzed to extract information and thereby, knowledge from data in any form, structured as well as unstructured. The availability of advanced machines and special tools has led to the analysis of big data, which may solve many social problems, including poverty and unemployment.

Through this project, we got the opportunity to learn about how the statistics can be used in various ways across various dimensions such as intelligence, defense and artificial intelligence. It also broadened our horizons of looking at an unknown data and trying to find useful features and patterns. We also got an opportunity to learn Python as we programmed our analysis on this language.

CHAPTER: 2

Predictive Model

In recent years and with advances in the computing power of machines, predictive modeling has gone through a revolution. We are now able to run thousands of models at multi-GHz speeds on multiple cores, making predictive modeling more efficient, and more affordable than ever. Virtual machines, such as those provided by Amazon Web Services (AWS), give us access to practically unlimited quantitative power (at the high cost of course!).

Predictive analytics is driven by predictive modeling, Predictive analytics & machine learning go hand-in-hand, as predictive models typically include a machine learning algorithm, Predictive modeling overlaps substantially with the field of machine learning

Predictive modelling is the purpose of building a model that will capable of making predictions values. It’s also process of creating testing and validating a model to predict the probability of an outcome.[4]

The predictive models have a clear focus on what they want to learn and how they want to learn.

2.1 Traditional versus Machine Learning Predictive Model

2.1.1 Traditional Prediction Models

Two Types of Traditional Prediction Models

i. **Statistical Model**

In this traditional prediction model, the input data set with statistical assumptions and calculations determine the prediction algorithm. Input data sets are analyzed using statistical data (or "fitted with data"). Prediction algorithm best suited to describe data determined by statistical analysis.

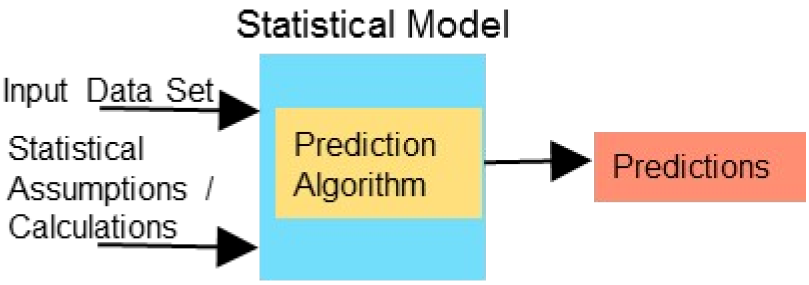


Figure 2.1

ii. Expert Rules Model

In this traditional prediction model uses a clear set of rules (eg, if X then Y) to convert the input into a prediction. Rather than predicting how an "algorithm" can be discovered through statistical calculations, these rules are usually those known by experts in the prediction domain (for example, medical knowledge practitioners have to diagnose / predict a disease).

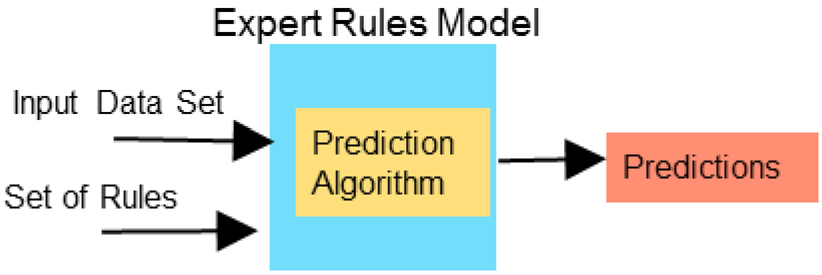


Figure 2.2

2.1.2 Machine Learning Prediction Models

machine learning prediction models are developed in two stages.

i. In the first training stage, a machine learning model is trained. The input data associated with the historical results and a training algorithm are used, arrived at the prediction algorithm.

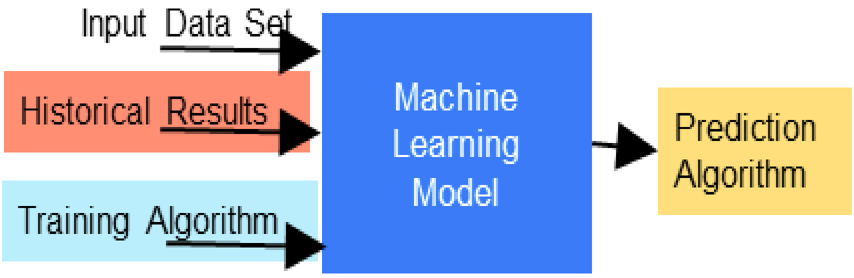


Figure 2.3

End of first stage, the model is “trained” and is now ready to make predictions.

ii. In the second prediction stage, the trained machine learning model uses the prediction algorithm arrived at in the training stage to transform new inputs into predictions.

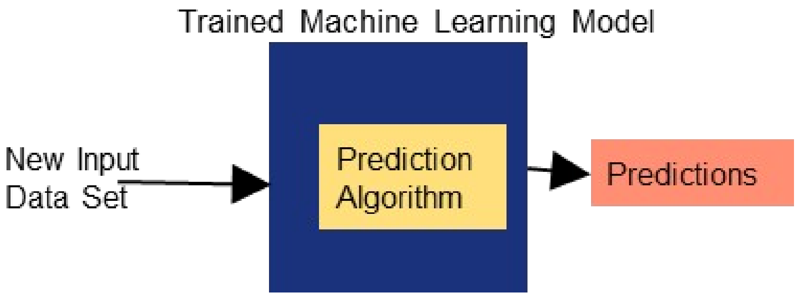


Figure 2.4

The common machine learning predictive models are two types Supervised learning model and Unsupervised learning model , which try to predict certain values using the values in the input data set.

*** **<u>Note</u>:-** This prediction project based on supervised machine learning so I am focuse on only supervised learning models.

2.2 Supervised Learning for Predictive Model

2.2.1 Principle

However, there is a general principle that underlies all supervised machine learning algorithms for predictive modeling.

Machine learning algorithms are described as learning a target function (f) that best maps input variables (X) to an output variable (Y): Y = f(X)

This is a general study task where we would like to make predictions given future (y) new examples of input variables (x). We do not know what the function (f) looks like or its form. If we did, we would use it directly and we would not need to learn it from the data using machine learning algorithms.

The most common type of machine learning is to learn the mapping Y = f(X) to make predictions of Y for new X. This is called predictive modeling or predictive analytics and our aim is to make the most accurate predictions possible.[5]

2.2.2 Supervised Learning Techniques

There are two types of Supervised Learning techniques:

i. **Regression**
Regression have some algorithms like Linear Regression, MLR, SVR, KNN Regressor etc. We apply when the target (output, dependent variable, quantity to be predicted) is numerical variable (continuous). In this many problem solve like House Price Prediction, Stock Price, Gold Price etc.

ii. **Classification**
Classification have some algorithms like Logistic Regression, KNN Classifier, Random Forests, Decision Trees, ANN. We apply this when the target (Output, dependent variable) is a categorical variable. In this many problem solve like Credit default, Spam Classifier etc.

However, one of the fundamental questions every data scientist faces is:
“Which predictive model is more appropriate for the problem at hand?”

Answering this question comes down to a basic, fundamental question in every machine learning problem:

“What does the target you are trying to predict look like?”

If you are trying to predict a continuous target & a numerical variable then you will need a regression model.

But if you are trying to predict a discrete target& a categorical variable then you will need a classification model.

- Both type of task we have many model:

Regression	Classification
Linear Regression	Logistic Regression
KNN Regressor	KNN Classifier
Lasso Regressor	Classification Trees
ANN	ANN

Table 2.1

2.3 Learning Model & Algorithms

The learning model establishes a relationship between the target figure, that is, the feature being predicted and the predictor features.

Learning Model =Model + Learning Algorithm

Model: The general formulation of the relationship between feature and target

Learning Algorithm: The procedure to find the specific form of the Model, Usually by learning some parameter from data.

Model	Learning Algorithm
Linear Regression	Ordinary Least Squares Method
Logistic Regression	Gradient Descent
ANN	Back propagation
SVM	Perceptron Learning Algo

Table 2.2

“when to use supervised learning for doing predictive analytics?”
you could use supervised learning if you have these three elements:

- **i.** There isn't a clearly known mathematical relationship between features and target.
- **ii.** There is a pattern or relationship between features and target and this is precisely what you are trying to learn.
- **iii.** Data :- you must have enough data in both quantity and quality.

2.4 Workflow of Predictive Modelling

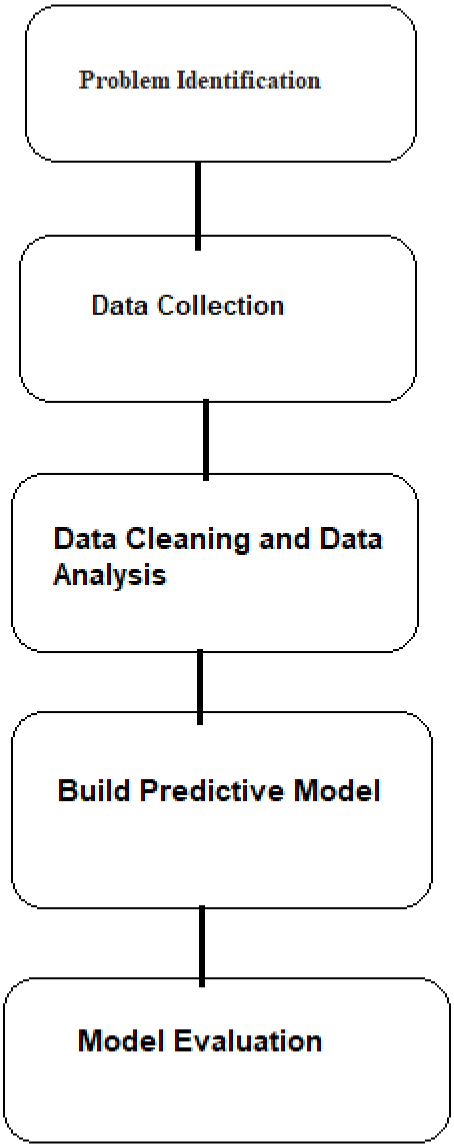


Figure 2.5

i. **Problem Identification**

In this section first identify the problem what we do predict and what are use of problem. What are business objective of problem, outcomes of problem and Scope of problem.

ii. **Data Collection**

Data collection is gathering the necessary details required for the analysis. It involves the historical or past data from an authorized source over which predictive analysis is to be performed.

iii. **Data Cleaning and Data Analysis**

Data Cleaning is the process in which we refine our data sets and we remove un-necessary and erroneous data. It involves removing the redundant data and duplicate data from our data sets.

In Data Analysis involves the exploration of data. We explore the data and analyze it thoroughly in order to identify some patterns or new outcomes from the data set. In this we discover useful information by identifying some patterns or trends.

iv. **Build Predictive Model**

In this of predictive analysis, we use various algorithms to build predictive models. It can be perform by knowledge of python, R, Statistics and MATLAB and so on.

You will require divide your dataset into at least three different subsets for building a model.

- **Training data:** This is the largest subset & on which you will build a model. The model also learns from this data.
- **Validation data:** This is the subset for which the model is evaluated continuously. As you refine the model, you will continue to test against validation data set.
- **Test data:** This is the final data set that you will use to evaluate the fit of the model. This data set should only be used once.

v. **Model Evaluation**

In this part of predictive model we can evaluate model by error mapping using MSE, RMSE, etc.

CHAPTER: 3

Regression

when you are trying to predict a numerical variable then use regression.

In other words when the target or also called out put, depend environmental or the quantitatively predictive is numerical often a continuous viable. We use regression model.

In this many problem solve like House Price Prediction, Stock Price, Gold Price etc.

3.1 Types of Regression

There are a wide range of sorts of regression utilized for forecast analysis model. Example: Multiple Linear Regression, KNN-Regressor, Lasso Regression, SVR and ANN.

3.1.1 Multiple Linear Regression

Multiple linear regression (MLR), also known simply as multiple regression, is a statistical technique that uses several explanatory variables to predict the outcome of a response variable.[6] Multiple linear regression (MLR) aims to model the linear relationship between explanatory (independent) variables and response (dependent) variables.

In essence, multiple regression is the extension of ordinary least-squares (OLS) regression that involves more than one explanatory variable.[6]

The General Formulation of Multiple Linear Regression:

- in this model We try to predict the target using a linear combination of p features.

$$y_{pred} = w_0 + w_1 x_1 + w_2 x_2 + \ldots\ldots.. + w_p x_p + \epsilon$$

Where,

y_{pred}= Dependent Variable

x_i= Expanatory Variables

w_0= y-intercept (constant term)

w_p= slope coefficients for each explanatory variable

ϵ= the model’s error term (also known as the residuals)

- This models often uses a procedure Ordinary least Square to find the w’s such that the following quantity is minimal.

$$\text{RSS} = \sum_{i=1}^{n} (Ypred - Yi)^2$$

3.1.2 KNN-Regression

In this model is the principle behind the KNN model is that in order to make prediction for new observation. First we find a predefine number of taring sample (K) that are closet in distance to the new observation and then use the target values of “K-Neighbors” to predict the target for the new observation. KNN has been used in statistical estimation and pattern recognition already in the beginning of **1970’**s as a non-parametric technique.

Steps in KNN Model

i. Choose the Number of nighbors

ii. Choose a distance metrics (Euclidean Distance)

iii. Compute the distance form the query observation

iv. Find the k closet training observation to query observation these are the K-neighbors

v. Predict the target value of the query observation by calculating a weighted average of the nighbors taget values.

KNN can require a lot of memory or space to store all of the data, but only performs a calculation (or learn) when a prediction is needed, just in time. You can also update and curate your training instances over time to keep predictions accurate.[5]

3.1.3 Lasso Regression

Lasso Regression (Least Absolute Shrinkage and selection Operator) also penalizes absolutely the length of the regression coefficients. further, It's capable to enhancing the accuracy of linear regression models.

it is a clever modification to the multiple regression model that ultimately excludes features are irrelevant to our model.

Lasso uses a method that performs variable selection and regularisation in order to enhance the prediction accuracy of the multiple regression model.

$$\sum_{i=1}^{n}(y_i - \sum_{j} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

In this equation shows that Lasso regression uses absolute values within the penalty characteristic, as opposed to squares and that’s the way it differs from ridge regression in a manner that it. This often ends in penalizing values which reasons a number of the parameter estimates to show out precisely zero. The greater the penalty carried out, in addition the estimates get reduced closer to absolute 0. This results to selection of variables out of n variables.

- In this regression normality can't be assumed, rest all the assumptions are same as least square regression.
- It helps in feature selection as it shrinks its coefficients to zero.
- This technique is a regularization strategy and utilizations L2 technique.
- If gathering of indicators are exceptionally associated, rope picks just a solitary one of the indicators if gathering of indicators are exceedingly co-related and contracts the others to zero.

3.1.4 Support Vector regression

Support Vector regression is type of Support vector machine that supports linear and non-linear regression. SVR requires the training data**: { X, Y}** which covers the domain of interest and is accompanied by solutions on that domain.

Support Vector Machine, has every one of the highlights like the primary highlights through which the calculations (maximal edge) are describe .The Support Vector Regression (SVR) incorporates or utilizes indistinguishable standards from the SVM with just a couple of contrasts. Above all else, it turns out to be hard for the forecast of data within reach, since result is a genuine number which has unbounded conceivable outcomes. Be that as it may, the calculation is progressively convoluted consequently to be considered. In any case, the principle thought is: To limit the blunder, boosts the edge by individualizing the hyper-plane, keeping mistakes into record.

"Why SVR over Linear Regression?"

In simple regression we try to minimize the error rate. While in SVR we try to fit the error within a certain threshold. Support Vector Machine, has all the features like SVR, the main features through which the algorithms (maximal margin) are characterize.SVR works on same principle as SVM works with only a few differences. In the same way as with classification approach the main idea is to seek and optimize the generalization bounds mentioned for regression.

3.2 Model Evaluation for Regression

There are many evaluation metrics used to measure the performance of these models, The intuition behind all of them is to measure how close the predicted values are to the observed target values.

some of the most commonly used metrics are the following:

- Mean Squared Error (also its square root , called Root Mean Squared Error)
- Mean absolute error
- Explained variance score
- R-squared

All of them can found in sklearn.metrics module

i. Mean Squared error (MSCE)

these metric measures the average of the square of the errors or deviations, that is the square of the difference between the observed and predicted values. The smaller this metric the better the model

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_{pred_i} - y_i\right)^2$$

i. Root Mean Squared Error

Just the square root of MSE & it is more interpretable because it is in the same units as the target.

RMSE = $\sqrt{MSE}$

However the MSE has nice theoretical properties and this is why is the most popular metric for evaluating regression models.

3.3 Cross-Validation

To estimate how our model will perform with data we have not seen before we use a technique called cross-validation.

Cross-Validation Techniques are many but we are use only “Holdout method” so:

Holdout method

- Hold out a percentage of the observations (between 10 – 40%), 20% or 25% are standard. This is called the “**testing”** data sets (X_test, y_test). This is the data set in which we calculate our evaluation metric.
- The rest of the observations are used to train the model. This is called the “**training**” data set (X_train, y_train)it is used to train the model.

Training
Testing

Figure 3.1

- The observations are randomly assigned to both groups.

3.4 Overfitting

Overfitting is the situation when the model “learns” almost every aspect of the training data including the random noise, as a consequence a model that has been overfit it makes poor predictions for unseen data.

It has poor predicted performance and it is not good for predicting.

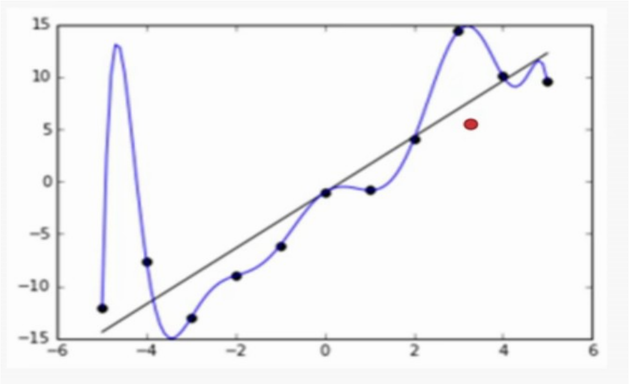


Figure 3.2

We've all seen observations in the picture.

We can see that the blue points is trying really hard to make an accurate prediction for every point in the training data set but because of that it adapts very strange behavior between the points. So this model is not very likely to produce good predictions for unseen data points.

In contrast the straight line we keep a simpler model misses many of the training examples but on average is more likely to produce better predictions for example a new data point. So a chance to read one is better predicted by a straight line model than 40 or more complex.

3.5 Regularization

Regularization are techniques that are used to prevent overfitting.

And some models like Lasso regression & Elastic net regression automatically apply regularisation.

Overfitted model

Figure 3.3

CHAPTER: 4

Experiment and Result

4.1 Diamond Price Prediction

4.1.1 Statement of problem

A company, **Intelligent Diamond Reseller (IDR)**, wants to get into the business of reselling diamonds. They want to innovate in the business, so they will use predictive modelling to estimate how much the market will pay for diamonds. Of course, to sell diamonds in the market, first they have to buy them from the producers, this is where predictive modelling becomes useful.

4.1.2 Objective

In this prediction we use Diamonds data set which containing the prices and other features of almost 54,000 diamonds. When a person or jewelry company want to put a bid to purchase a diamond but is unsure that how much it should bid. In this prediction model to make a recommendation on how much will be bid of a diamond.

The problem is requires Supervised Learning. The instances come with an expected output (i.*e.* the diamond's price).

Predicting the price of a diamond from data set is a Regression Task. More specifically, a Multivariate Regression Task.

We will be using the Batch Learning technique since the data is not live-fed from a source.
We will also be the Mean Square Error (MSE) for our performance measure (typical for Regression tasks).
In this prediction we apply many algorithm MLR, KNN, LASSO etc. Then find best price using this Model “**in Python”.**

With diamond data set I am done this task:
- Presenting & loading the diamonds dataset
- Doing some data preparation
- Building three predictive models
- comparing them using the MSE metric
- Using the best model make prediction about the prices of diamonds

4.1.3 Data Set Information

This project attempts to predict the diamond price with respect to the Cut, Clarity, Color, Price etc. It requires a proper data of diamond as the project also emphasizes. So, it is necessary to have a trusted source having relevant and necessary data required for the prediction.

We will be using diamond dataset (https://www.kaggle.com/) as the primary source of data. A dataset containing the prices and other features of almost 54,000 diamonds.

●**Features description**

●Number of Attributes: 10 (9 predictive features, 1 target)

●Feature Information: A data frame with 53,940 rows and 10 variables:

- price: price in US dollars ($326−−$18,823) **(target)**
- carat: weight of the diamond (0.2–5.01)
- cut: quality of the cut (Fair, Good, Very Good, Premium, Ideal)
- color: diamond colour, from J (worst) to D (best)
- clarity: a measurement of how clear the diamond is (I1 (worst), SI2, SI1, VS2, VS1, VVS2, VVS1, IF (best))
- x: length in mm (0–10.74)
- y: width in mm (0–58.9)
- z: depth in mm (0–31.8)
- depth: total depth percentage = z / mean(x, y) = 2 * z / (x + y) (43–79)
- table: width of top of diamond relative to widest point (43–95)

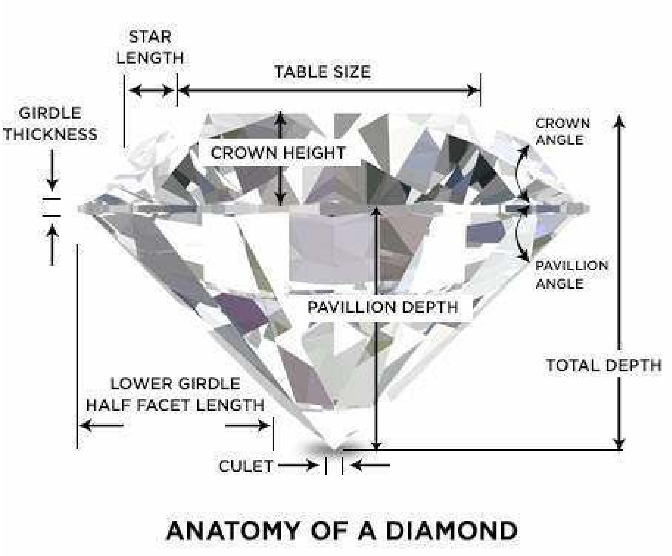
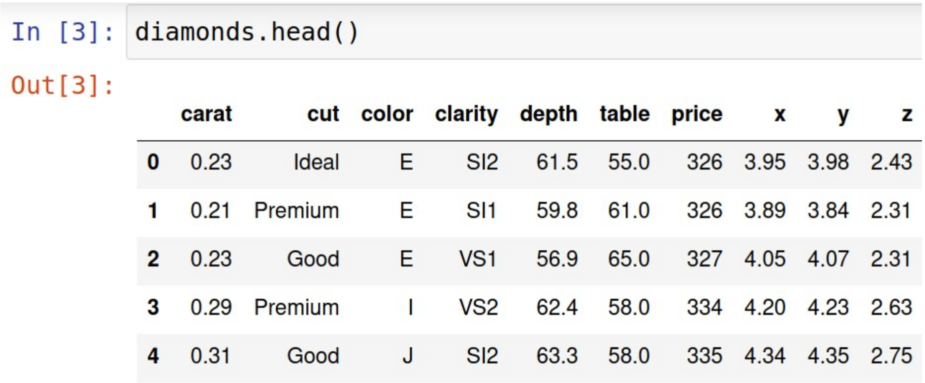


Figure 4.1

4.1.4 Implementation of Predictive Model

i. Importing useful Libraries & Raw Data

First I am import some useful libraries such as NumPy, Pandas, Matplotlib & then import raw data.
you can see how the original dataset looks like here.

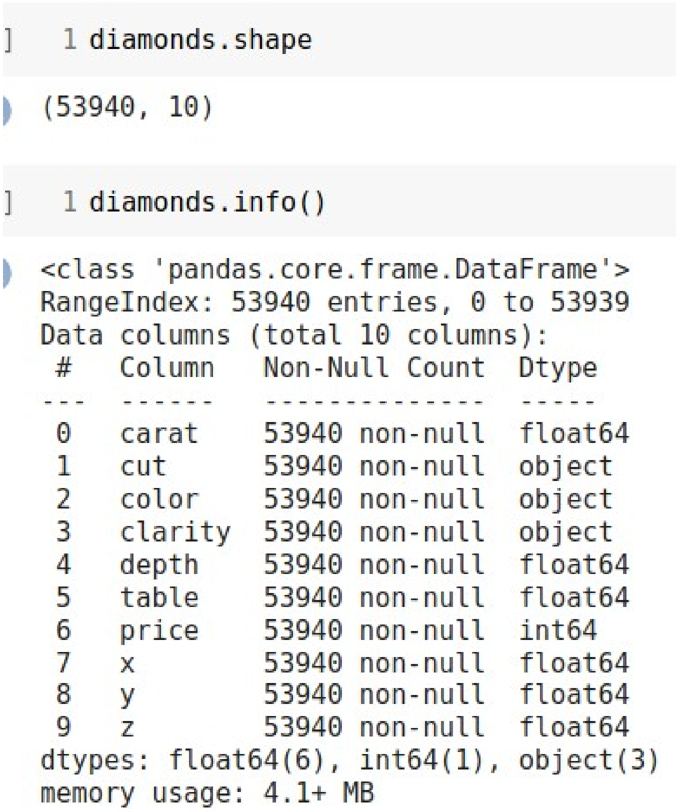


```
In [3]: diamonds.head()
Out[3]:
```

	carat	cut	color	clarity	depth	table	price	x	y	z
0	0.23	Ideal	E	SI2	61.5	55.0	326	3.95	3.98	2.43
1	0.21	Premium	E	SI1	59.8	61.0	326	3.89	3.84	2.31
2	0.23	Good	E	VS1	56.9	65.0	327	4.05	4.07	2.31
3	0.29	Premium	I	VS2	62.4	58.0	334	4.20	4.23	2.63
4	0.31	Good	J	SI2	63.3	58.0	335	4.34	4.35	2.75

Figure 4.2

Here you can see the number of observations that we have and a number of features and some information about this object.



```
diamonds.shape
```

```
(53940, 10)
```

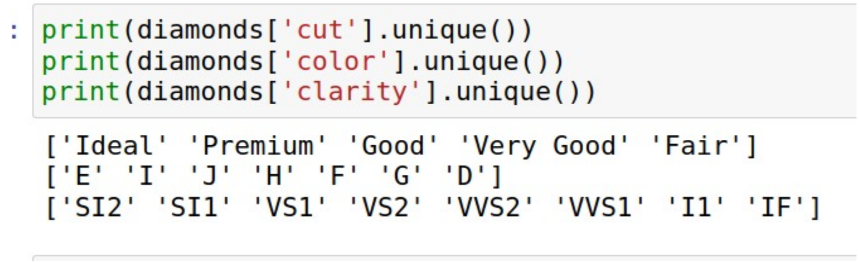
```
diamonds.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 53940 entries, 0 to 53939
Data columns (total 10 columns):
 #   Column    Non-Null Count  Dtype  
---  ------    --------------  -----  
 0   carat     53940 non-null  float64
 1   cut       53940 non-null  object 
 2   color     53940 non-null  object 
 3   clarity   53940 non-null  object 
 4   depth     53940 non-null  float64
 5   table     53940 non-null  float64
 6   price     53940 non-null  int64  
 7   x         53940 non-null  float64
 8   y         53940 non-null  float64
 9   z         53940 non-null  float64
dtypes: float64(6), int64(1), object(3)
memory usage: 4.1+ MB
```

Figure 4.3

ii. Data Cleaning and Data Analysis

We have 3 categorical variables in this data set.

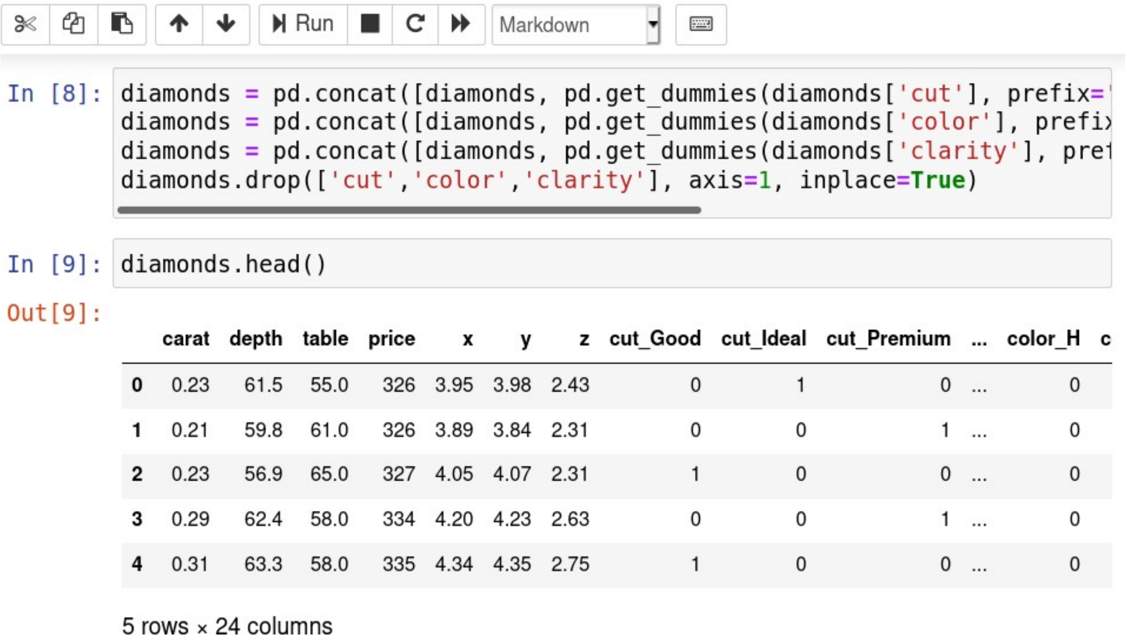


```
: print(diamonds['cut'].unique())
  print(diamonds['color'].unique())
  print(diamonds['clarity'].unique())

  ['Ideal' 'Premium' 'Good' 'Very Good' 'Fair']
  ['E' 'I' 'J' 'H' 'F' 'G' 'D']
  ['SI2' 'SI1' 'VS1' 'VS2' 'VVS2' 'VVS1' 'I1' 'IF']
```

Figure 4.4

One of the transformations we must perform is to tranform the categorical features to the one-hot-encoding format. we will do here in this cell is to transform these categorical variables to a set of binary variables or dummy variables.



```
In [8]: diamonds = pd.concat([diamonds, pd.get_dummies(diamonds['cut'], prefix='
        diamonds = pd.concat([diamonds, pd.get_dummies(diamonds['color'], prefix
        diamonds = pd.concat([diamonds, pd.get_dummies(diamonds['clarity'], pref
        diamonds.drop(['cut','color','clarity'], axis=1, inplace=True)

In [9]: diamonds.head()

Out[9]:
```

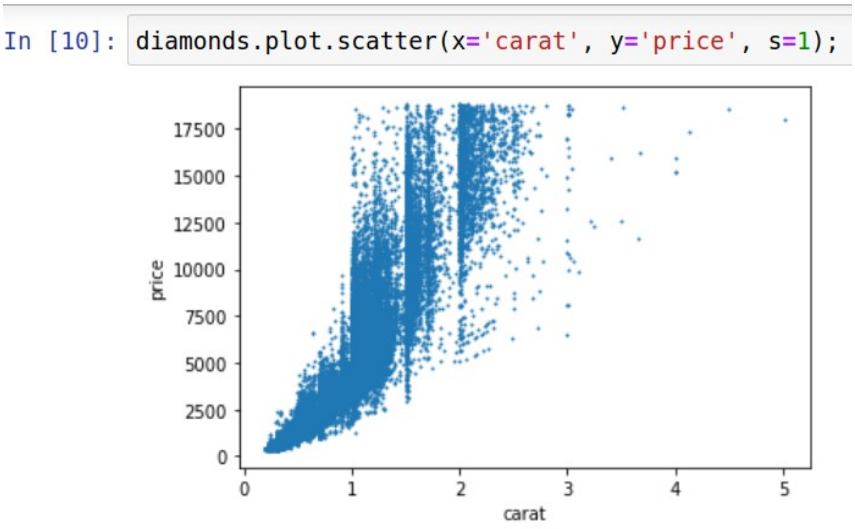
	carat	depth	table	price	x	y	z	cut_Good	cut_Ideal	cut_Premium	...	color_H	c
0	0.23	61.5	55.0	326	3.95	3.98	2.43	0	1	0	...	0	
1	0.21	59.8	61.0	326	3.89	3.84	2.31	0	0	1	...	0	
2	0.23	56.9	65.0	327	4.05	4.07	2.31	1	0	0	...	0	
3	0.29	62.4	58.0	334	4.20	4.23	2.63	0	0	1	...	0	
4	0.31	63.3	58.0	335	4.34	4.35	2.75	1	0	0	...	0	

5 rows × 24 columns

Figure 4.5

This is the one Hutten coding and we will we have to do this transformation because Scikit-learn accepts only numerical features.

- Now we do some feature engineering.In feature engineering means coming up with new features.



```
In [10]: diamonds.plot.scatter(x='carat', y='price', s=1);
```

Figure 4.6

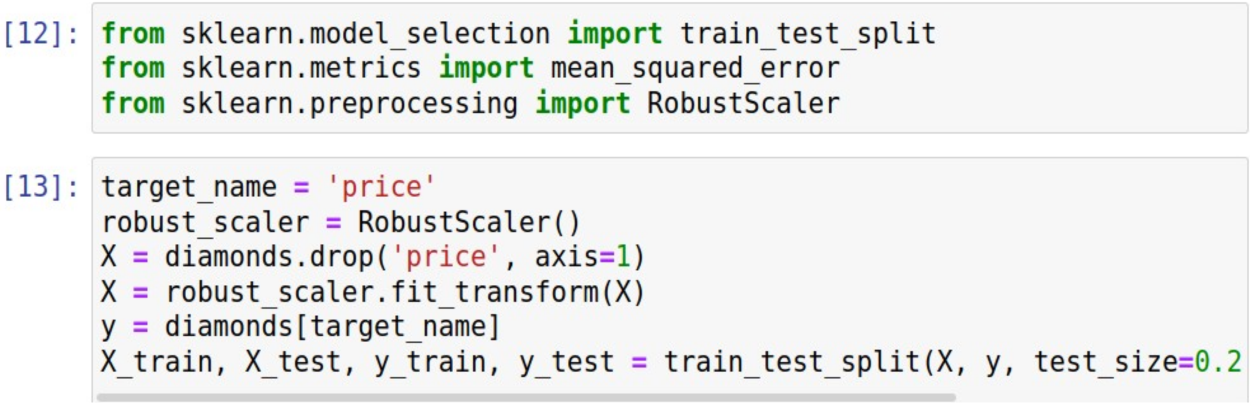
So just I can have to account for the only the oddity of the relationship between carat and price. I can add another feature called carat squared and this feature is just the squared of the current carat feature.
this is just a little example of feature engineering

●**Scaling** is an important operation Before training your models.

Since the features have different scales (carat goes from 0 to 5, y goes from 0 to 58) we need to be careful. Making sure that each feature has approximately the same scale and this is a crucial preprocessing step.

In this case we will use the Robust Scaler object which scales the features to a common scale and it is robust to outliers.

- Getting the train and test sets



```
[12]: from sklearn.model_selection import train_test_split
      from sklearn.metrics import mean_squared_error
      from sklearn.preprocessing import RobustScaler
```

```
[13]: target_name = 'price'
      robust_scaler = RobustScaler()
      X = diamonds.drop('price', axis=1)
      X = robust_scaler.fit_transform(X)
      y = diamonds[target_name]
      X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2
```

Figure 4.7

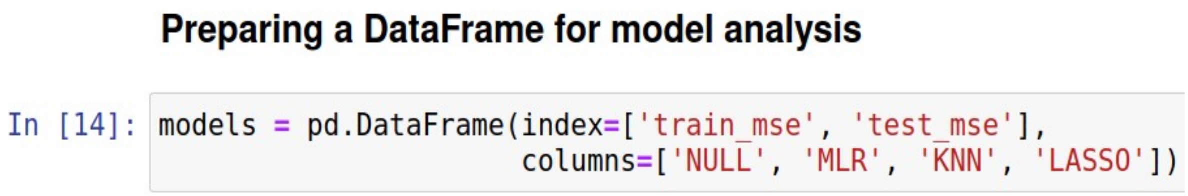
In this cell we prepare all the objects that we will use for training.
Here we import the Robust Scaler and instantiate one instance of this object.
Then we get the features by dropping the price column from our data frame.
Then we use the feet transform method from Robust Scaler to scale all the features to a common scale.

And finally we get the target by extracting the price column from the diamond states data frame.

then we use the train test splite to produce the division between training data and testing data.

we passed the 0.2 meaning that we want 80 percent of the data for training and 20 percent of the data for testing.

- We creat a evaluation matric for every model . That we will calculate



Preparing a DataFrame for model analysis

In [14]:
```
models = pd.DataFrame(index=['train_mse', 'test_mse'],
                      columns=['NULL', 'MLR', 'KNN', 'LASSO'])
```

Figure 4.8

iii. Build Predictive Model

1) Multiple Linear Regressio

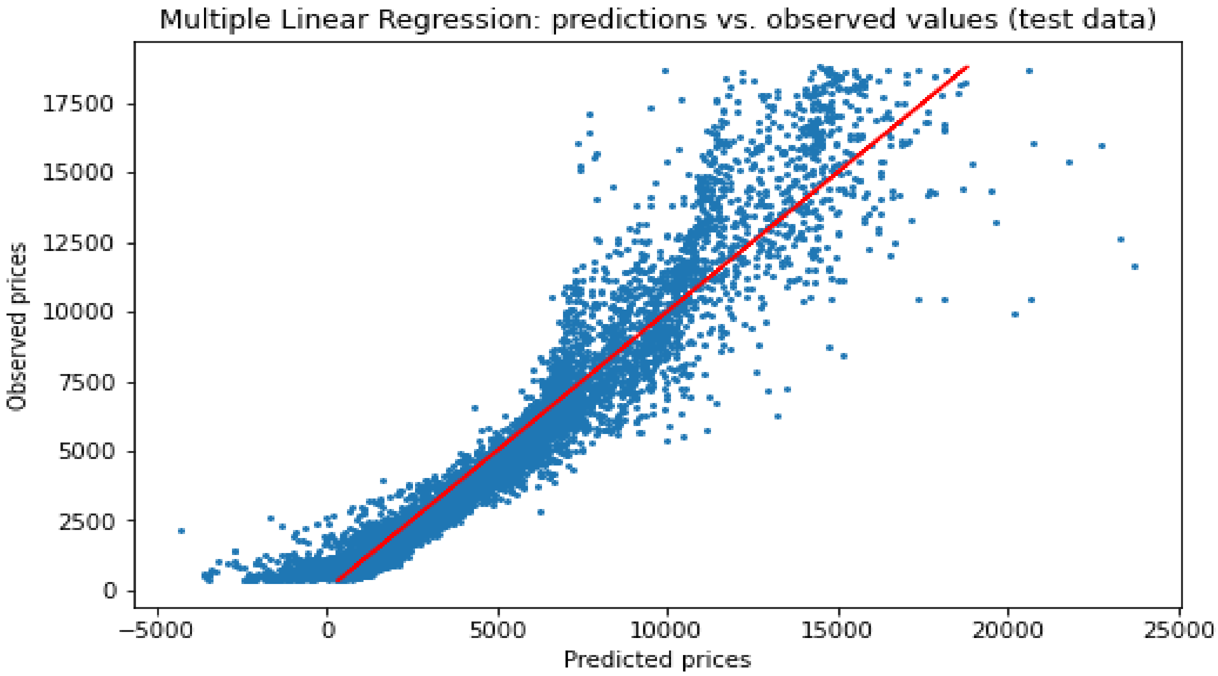


Figure 4.9

2) K-Nearest Neighbors Model

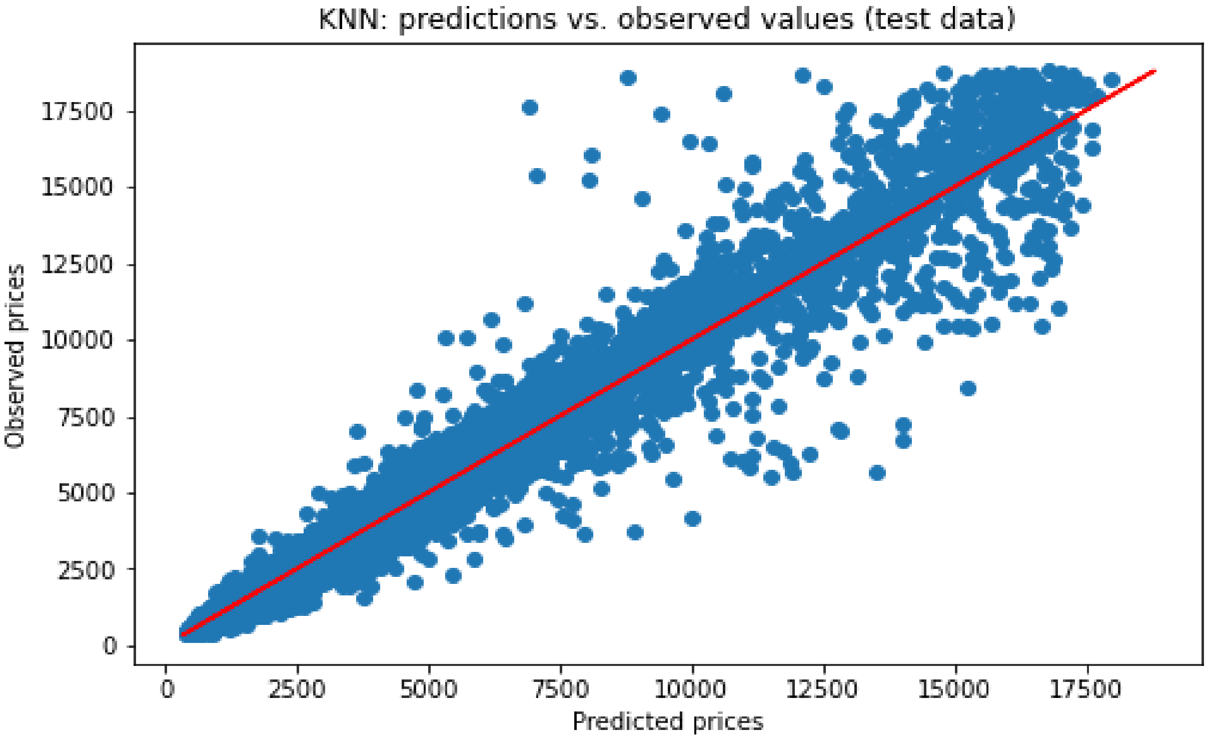


Figure 4.10

3) Lasso

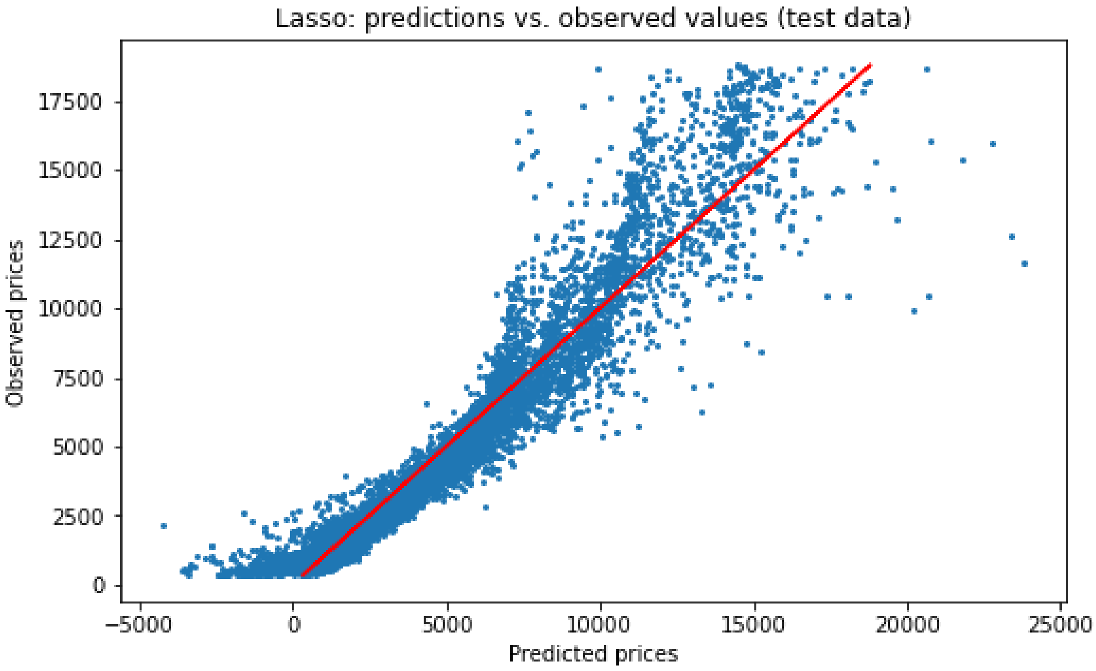


Figure 4.11

iv. Model Evaluation for Regression

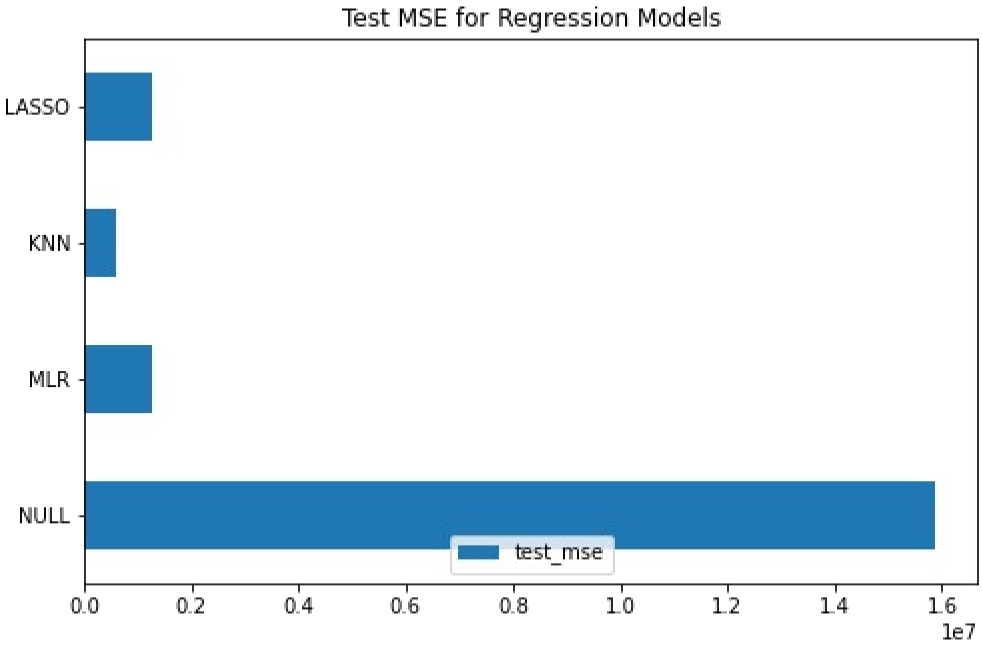


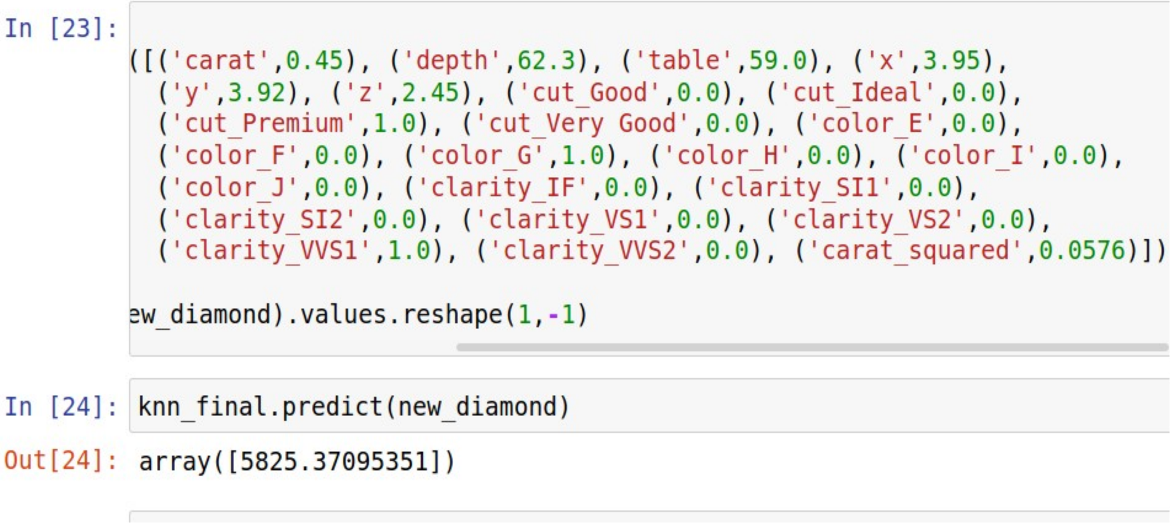
Figure 4.12

Result:- KNN Prediction model are best suitable for this data.

v. using the best model to predict the price of a diamond

I am happy with my model we can re-train it using all observations, and then use it to make predictions.

Give new unseen data for prediction:



```
In [23]:
([('carat',0.45), ('depth',62.3), ('table',59.0), ('x',3.95),
  ('y',3.92), ('z',2.45), ('cut_Good',0.0), ('cut_Ideal',0.0),
  ('cut_Premium',1.0), ('cut_Very Good',0.0), ('color_E',0.0),
  ('color_F',0.0), ('color_G',1.0), ('color_H',0.0), ('color_I',0.0),
  ('color_J',0.0), ('clarity_IF',0.0), ('clarity_SI1',0.0),
  ('clarity_SI2',0.0), ('clarity_VS1',0.0), ('clarity_VS2',0.0),
  ('clarity_VVS1',1.0), ('clarity_VVS2',0.0), ('carat_squared',0.0576)])

ew_diamond).values.reshape(1,-1)

In [24]: knn_final.predict(new_diamond)

Out[24]: array([5825.37095351])
```

Figure 4.13

Result: - with thise Features diamond price predict : $5825 US Dollars

4.2 Gold Price Prediction

4.2.1 Statement of problem

Prediction is function in management to predict outcome. It also describe as the process of estimation of unknown future situation. Gold is a precious yellow good it use as money. It use as money in the world. The demand of gold is risen. Gold plays an important role as a stabilizing influence for investment portfolios.

The gold prices are time series data of gold prices fixed. Factors influencing gold prices are many and we have to be selective in this study to ensure that the model developed is significant.

It is a common practice in gold trade to use Yahoo Financial as the factor for pricing of gold and these become the published benchmark price used by the producers, consumers, investors and central banks.

In this study, we proposed the development for predicting future **gold price using Multiple Linear Regression (MLR).** The data used in this part are the Gold Prices (GP) from the Yahoo Financial. GP will be the single dependent variable in this model. We began by identifying the factors that influence the price of gold. These factors were used as independent variables in this MLR model. We also use **Knn-Regressor** for predicting Price of Gold. Here gold price are in US dollar.

4.2.2 Objective

Objective of this to develop a model for predicting gold prices based on economic factors such as inflation, currency price movements and others. Due to the increase in demand for gold in India and worldwide, it is necessary to develop a model that reflects the structure and pattern of gold market and forecast movement of gold price. The most appropriate approach to the understanding of gold prices is the Multiple Linear Regression (MLR) model. MLR is a study on the relationship between a single dependent variable and one or more independent variables, as this case with gold price as the single dependent variable. The fitted model of MLR will be used to predict the future gold prices.

4.2.3 Methodology for Gold Price Prediction

i. Multiple Linear Regression

We creating a Multiple linear regression model for predicting price so first we need all know about linear regression and Multiple Linear Regressor.

Multiple linear regression (MLR), also tell as multiple regression, is a statistical technique that uses several explanatory variables to predict the outcome of a response variable. The goal of multiple linear regression (MLR) is to model the linear relationship between the independent variables and dependent variable.

This Model we try to predict the target using a Linear Combination of p features

$$\text{Ypred} = \text{W0} + \text{W1X1} + \text{W2X2} + \ldots\ldots\ldots \text{WpXp}$$

This models often uses a procedure Ordinary least Square to find the w's such that the following quantity is minimal.

$$\text{RSS} = \sum_{i=1}^{n} (Ypred - Yi)^2$$

ii. KNN-Regression

In this model is the principle behind the KNN model is that in order to make prediction for new observation. First we find a predefine number of taring sample (K) that are closet in distance to the new observation and then use the target values of "K-Neighbors" to predict the target for the new observation. KNN has been used in statistical estimation and pattern recognition already in the beginning of **1970'**s as a non-parametric technique.

Steps in KNN Model

i. Choose the Number of nighbors

ii. Choose a distance metrics (Euclidean Distance)
iii. Compute the distance form the query observation
iv. Find the k closet training observation to query observation these are the K-neighbors
v. Predict the target value of the query observation by calculating a weighted average of the nighbors taget values.

KNN in Scikit Learn

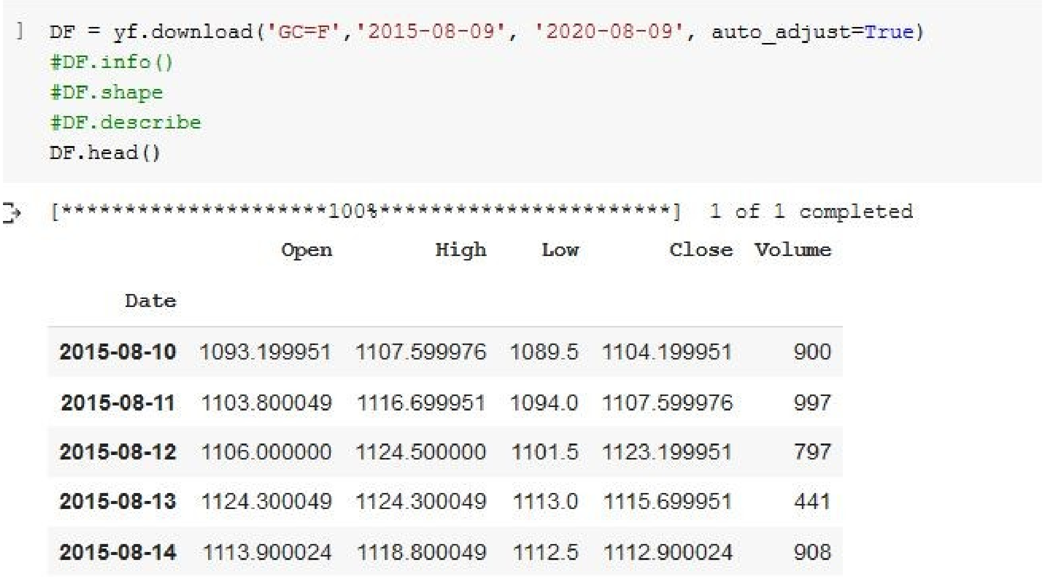
- Use the KNeighborsRegressor estimator (From sklearn.neghbors)
- Important parameters:
- n_neighbors: K the number of neighbors
- weights: Weight function use on prediction. Possible values: Uniform distance or user-define
- metric: The distance metric to use minkowski or Euclidian.

4.2.4 Implementation of Predictive Model

Importing Dataset of Gold

We will create a machine learning linear regression model that takes information from the Yahoo Financial gold prices in US dollar and returns a prediction of the Gold price the next day. We will cover the following topics ito predict gold prices using machine learning in python.

We first install library of yahoo financial to fetch data set of gold then importing its library. Dataset have Date, Open, high, close, low and volume. We use dataset between 2015-2020.



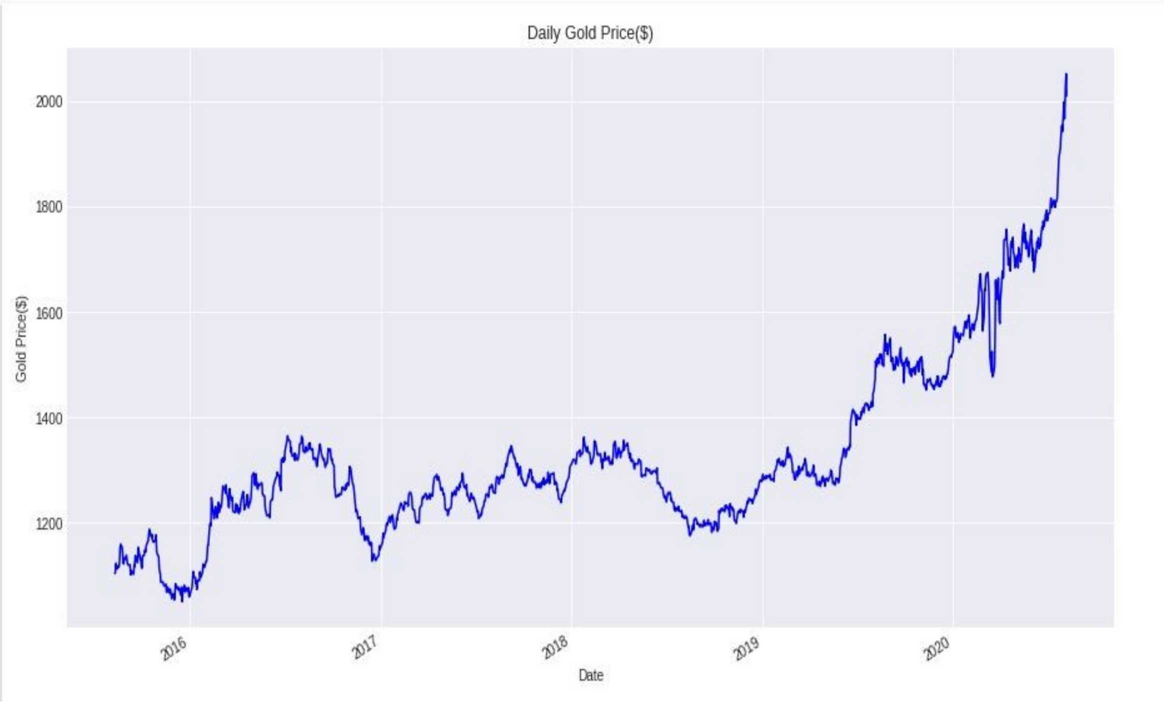
```
DF = yf.download('GC=F','2015-08-09', '2020-08-09', auto_adjust=True)
#DF.info()
#DF.shape
#DF.describe
DF.head()
```

```
[*********************100%***********************]  1 of 1 completed
```

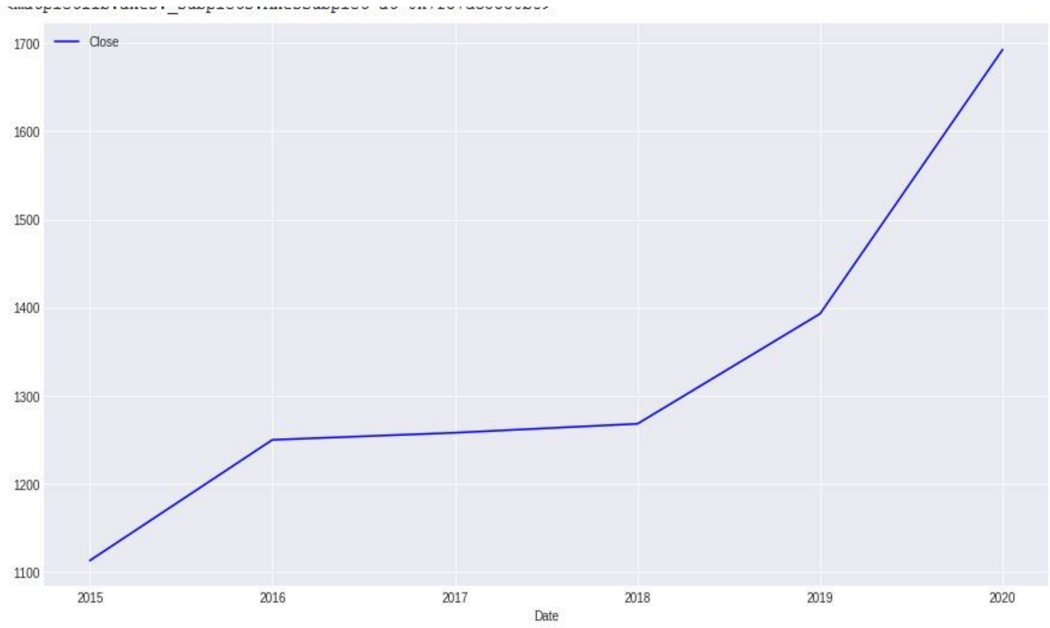
	Open	High	Low	Close	Volume
Date					
2015-08-10	1093.199951	1107.599976	1089.5	1104.199951	900
2015-08-11	1103.800049	1116.699951	1094.0	1107.599976	997
2015-08-12	1106.000000	1124.500000	1101.5	1123.199951	797
2015-08-13	1124.300049	1124.300049	1113.0	1115.699951	441
2015-08-14	1113.900024	1118.800049	1112.5	1112.900024	908

Data Cleaning and Data Analysis

After data collection we do cleaning and Analysis of data set by many different Way. It first fig1 show daily price of gold in graph.



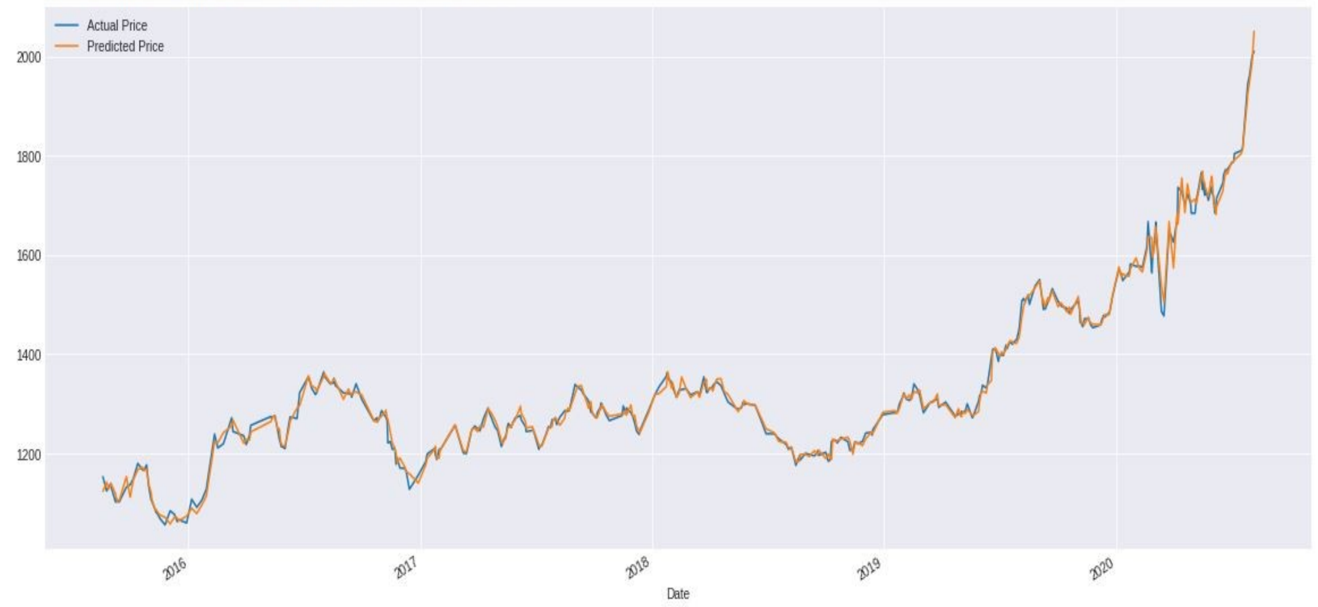
How the gold price year wise varies in 2015-2020.



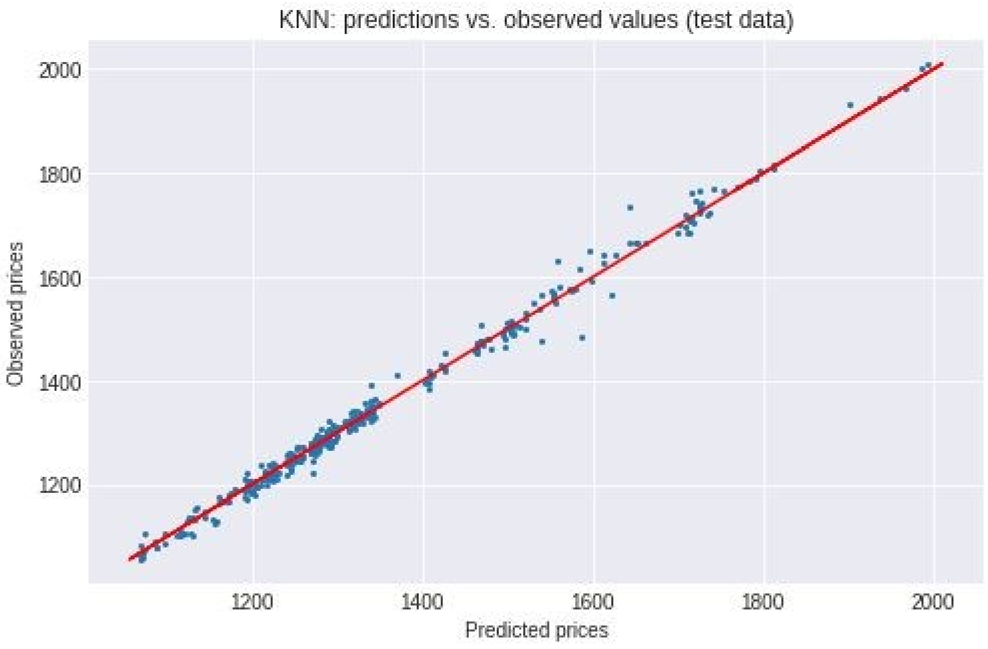
Result of Prediction of Gold Price

iii. Using MLR (Multiple Linear Regressor)

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f37df230978>
```



iv. Using KNN-Regressor



4.3 Stock Price Prediction

4.3.1 Statements of Problem

Stock market is very vast and difficult to understand. It is considered too uncertain to be predictable due to huge fluctuation of the market.
Financial investors of today are facing this problem of trading as they do not properly understand as to which stocks to buy or which stocks to sell in order to get optimum result. So, the purposed project will reduce the problem with suitable accuracy faced in such real time scenario.

4.3.2 Objective

To identify factors affecting stock market.
To predict an approximate value of share price.
The main feature of this project is to generate an approximate forecasting output and create a general idea of future values based on the previous data by generating a pattern

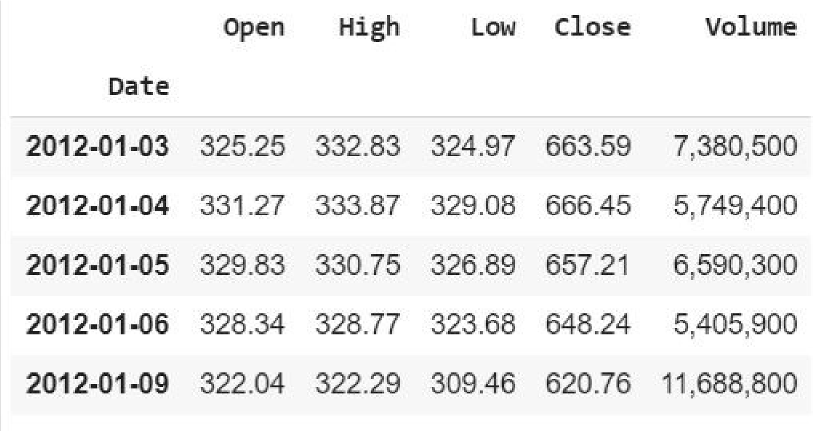
4.3.3 Methodology and Implementation

Stock market prediction seems a complex problem because there are many factors that have yet to be addressed and it doesn't seem statistical at first. But by proper use of machine learning techniques, one can relate previous data to the current data and train the machine to learn from it and make appropriate assumptions.

4.4.4 Implementation of Predictive Model

Importing Raw Data

This project attempts to predict the stock value with respect to the stock's previous value and trends. It requires historic data of stock market as the project also emphasizes on data mining techniques. So, it is necessary to have a trusted source having relevant and necessary data required for the prediction. We will be using google Stock Price website (https://www.kaggle.com/) as the primary source of data. This website contains all the details such as: Opening value, Closing value, Highest value, Lowest value, volume.



	Open	High	Low	Close	Volume
Date					
2012-01-03	325.25	332.83	324.97	663.59	7,380,500
2012-01-04	331.27	333.87	329.08	666.45	5,749,400
2012-01-05	329.83	330.75	326.89	657.21	6,590,300
2012-01-06	328.34	328.77	323.68	648.24	5,405,900
2012-01-09	322.04	322.29	309.46	620.76	11,688,800

Data Preprocessing

The Preprocessing stage involves Data discretization, Data transformation, Data cleaning, Data Integration After the dataset is transformed into a clean data set, the dataset is divided into Training and Testing sets to evaluate.

Data discretization is defined as a process of converting continuous data attribute values into a finite set of intervals and associating with each interval some specific data value.

Data transformation is basically a data normalization the process of converting data from one format or structure into another format or structure

Data cleaning, data cleansing, or data scrubbing is the process of improving the quality of data by correcting inaccurate records from a record set.

Feature scaling is a method used to normalize the range of independent variables or features of data. here we are taking data from day1 to day60 and make prediction on the day61 and so on.

Feature Extraction

When the input data to an algorithm is too large to be processed and it is suspected to be redundant then it can be transformed into a reduced set of features. Determining a subset of the initial features is called feature selection. The selected features are expected to contain the relevant information from the input data, so that the desired task can be performed by using this reduced representation instead of the complete initial data.

In this only feature which are fed to neural network are chosen. This is the first step building the RNN. Here we are importing Keras packages and libraries. Keras is basically TensorFlow high level API for building and training model we are importing four libraries Sequential, Dense, LSTM, Dropouts.

A **Sequential** is appropriate for a plain stack of layers where each layer has exactly one input tensor and one output tensor.

Dense layer is the regular deeply connected neural network layer. It is most common and frequently used layer. Dense layer does the below operation on the input and return the output.

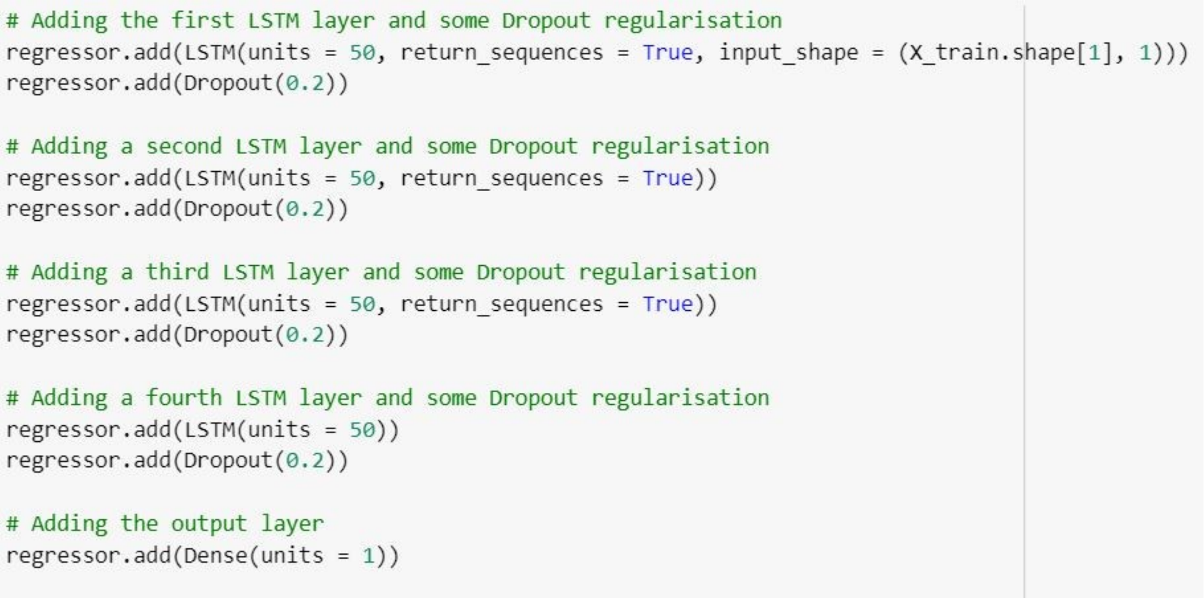
Long Short Term Memory networks usually just called "LSTMs" are a special kind of RNN, capable of learning long-term dependencies. All recurrent neural networks have the form of a chain of repeating modules of neural network.

Dropout is a technique where randomly selected neurons are ignored during training.

Taining neural network

The model here are RNN and LSTM based model for the prediction of stock prices The data is fed to the neural network and trained for prediction assigning random bias and weights.

In this the model composed of sequential Input layer followed by three LSTM layer and a dense layer with activation and finally a dense output layer with linear activation function.



```
# Adding the first LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50, return_sequences = True, input_shape = (X_train.shape[1], 1)))
regressor.add(Dropout(0.2))

# Adding a second LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50, return_sequences = True))
regressor.add(Dropout(0.2))

# Adding a third LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50, return_sequences = True))
regressor.add(Dropout(0.2))

# Adding a fourth LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50))
regressor.add(Dropout(0.2))

# Adding the output layer
regressor.add(Dense(units = 1))
```

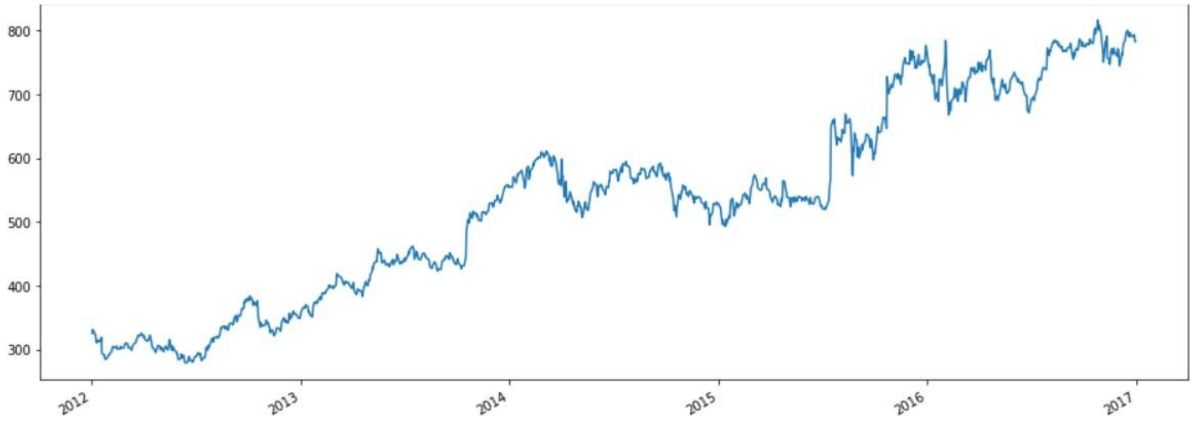
The type of optimizer used can greatly affect how fast the algorithm converges to the minimum value Here we have chosen to use Adam optimizer. The Adam optimizer combines the perks of two other optimizers.

Another aspect of training the model is making sure the weights do not get too large, hence, overfit for this purpose, we have chosen to use **Tikhonov regularization**.

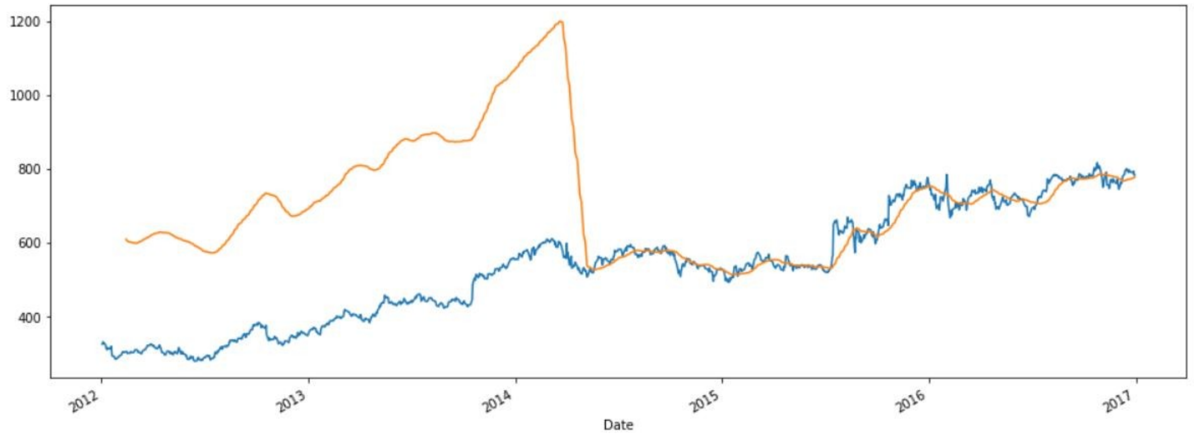
Analysis

The output value generated by the layer of RNN is compared with the target value. The error or the difference between the target and the obtained value is minimized by the back-propagation algorithm.

Growth of Stocks from 2012 to 2017



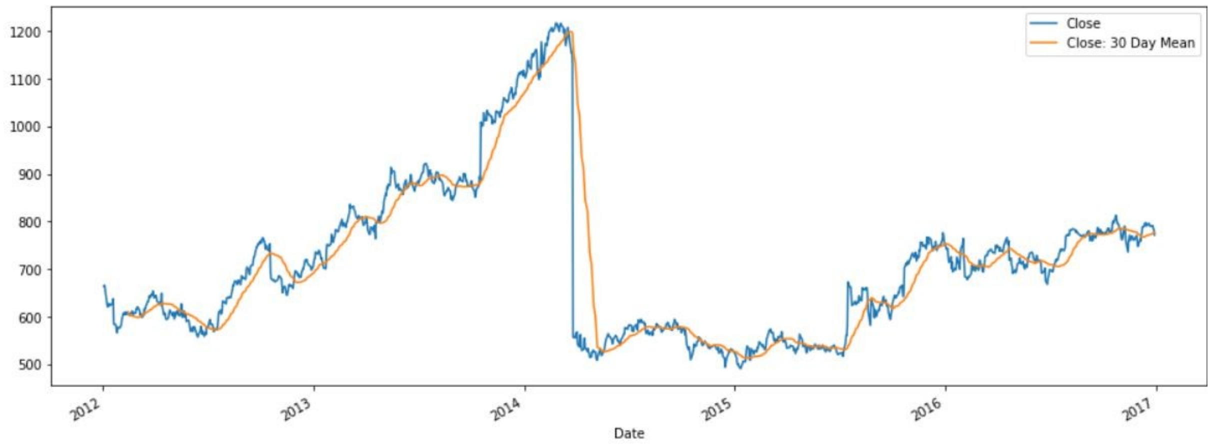
Comapares by the Prevoius Graph to the rolling mean (Moving average of the past 30 days)



The plot of close column verses the Seven day moving average of the closed column

Blue line-the close line column

Orange line-Thirty day rolling mean of the truce column

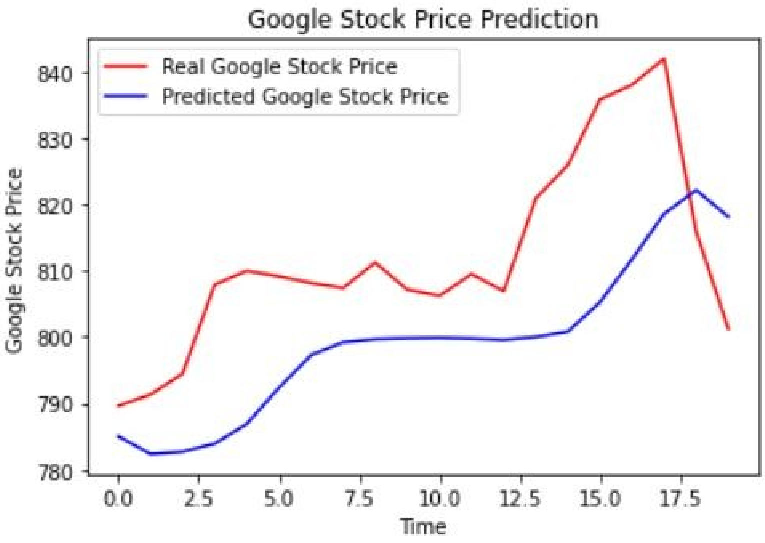


Visualization and Results

A rolling analysis of a time series model is often used to assess the model's stability over time. when analyzing financial time series data using a statistical

Model, a key assumption is that the parameters of the model are constant over time. Here we are using Iloc to select rows and columns by in order they are appear in the data frame.

To get the Predicted value we are going to merge the train dataset and test dataset on zero axis. And reshape the data.



4.4 Covid-19 Cases Prediction

4.4.1 Statement of problem

The outbreak of COVID-19 in different parts of the world is a major concern for all the administrative units of respective countries. India is also facing this very tough task for controlling the virus outbreak and has managed its growth rate through some strict measures.

In this presents the current situation of coronavirus spread in Uttar Pradesh India. With the help of data sources (till 14th Aug 2020) from Kaggle and Ministry of Health and Family Welfare, Government of India, and presents various trends and patterns.

4.4.2 Objective

In this prediction model we use SVR for predict no of cases how it varies. It can show the map of increasing case and help to Government and people can take many step for controlling rising or cases.

In this we also plot many Graph for show data analysis in how Covid-19 cases are increasing UP India.

4.4.3 Methodology for Covid-19 Cases UP India Prediction

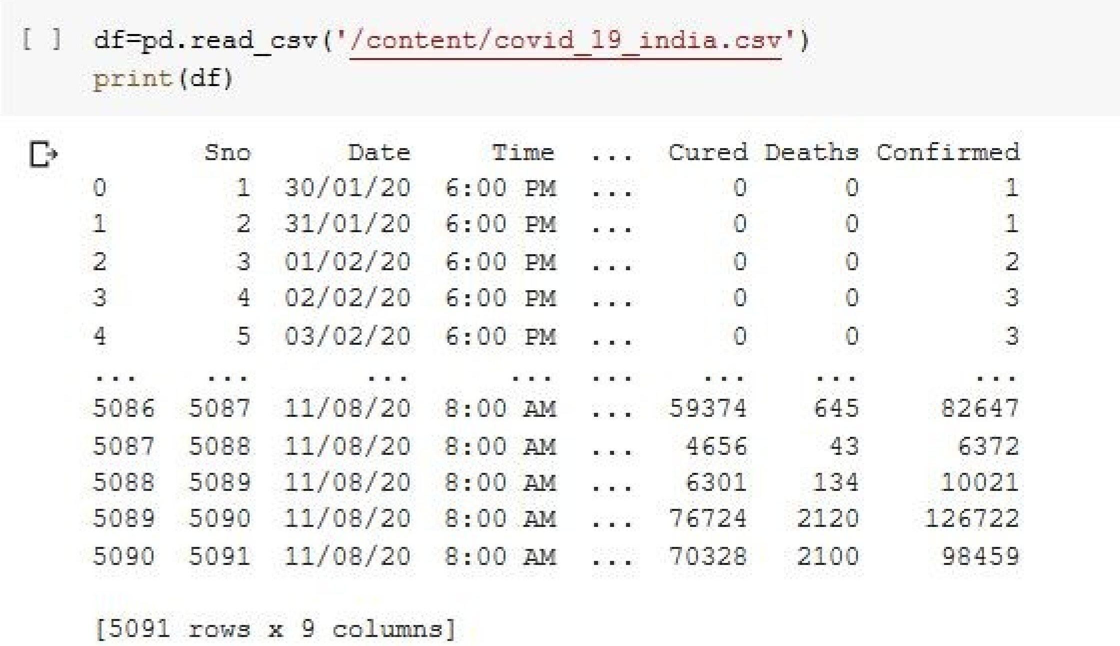
In this we use Support Vector Machine (SVM) for prediction of cases risen in Uttar Pradesh India. We first learn the concept of SVM here we use Support Vector Regressor for predicting number of cases.

Support Vector regression is type of Support vector machine that supports linear and non-linear regression. SVR requires the training data**: { X, Y}** which covers the domain of interest and is accompanied by solutions on that domain.

4.4.4 Implementation of Predictive Model

Importing Dataset of Covid-19 India

In this predictive model we use dataset of covid-19 India via Kaggle.com. Dataset contain Date, Number of Confirm case, Number of Cured, and number of Deaths. It also have data details according to state-wise of India. In this dataset have data till 11/08/2020.



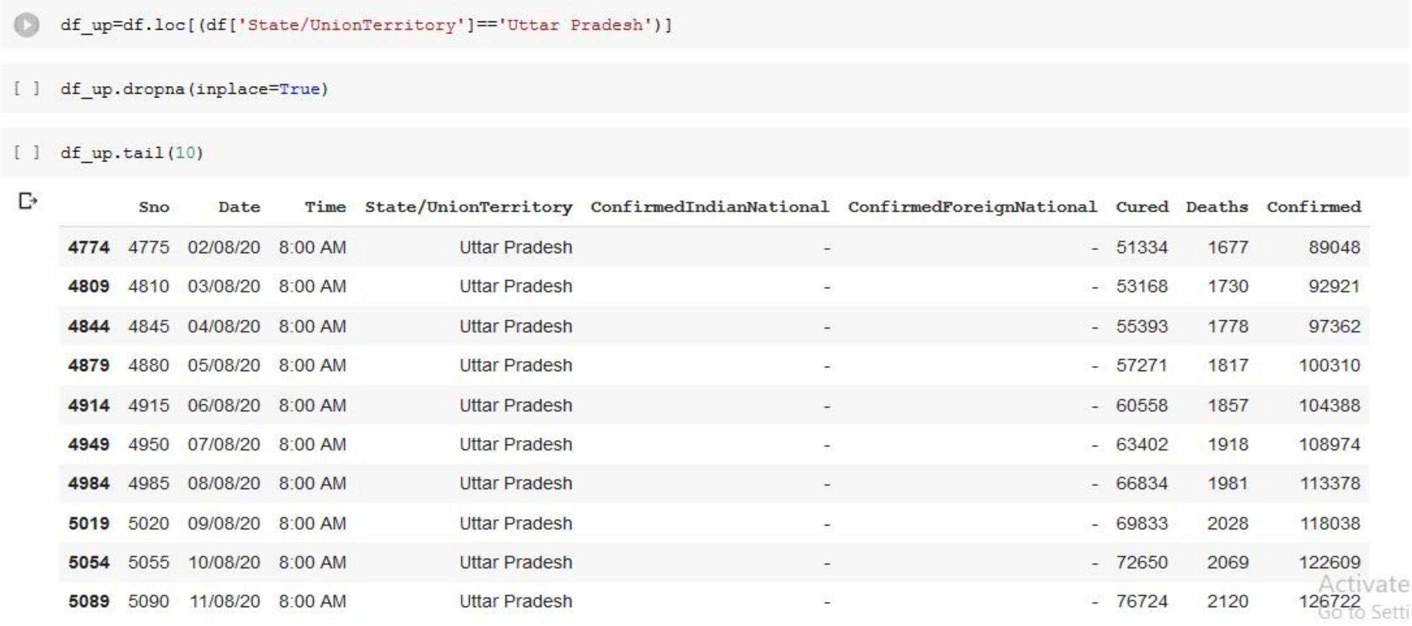
```
[ ]  df=pd.read_csv('/content/covid_19_india.csv')
     print(df)
```

```
        Sno      Date     Time  ...  Cured Deaths Confirmed
0         1  30/01/20  6:00 PM  ...      0      0         1
1         2  31/01/20  6:00 PM  ...      0      0         1
2         3  01/02/20  6:00 PM  ...      0      0         2
3         4  02/02/20  6:00 PM  ...      0      0         3
4         5  03/02/20  6:00 PM  ...      0      0         3
...     ...       ...      ...  ...    ...    ...       ...
5086   5087  11/08/20  8:00 AM  ...  59374    645     82647
5087   5088  11/08/20  8:00 AM  ...   4656     43      6372
5088   5089  11/08/20  8:00 AM  ...   6301    134     10021
5089   5090  11/08/20  8:00 AM  ...  76724   2120    126722
5090   5091  11/08/20  8:00 AM  ...  70328   2100     98459

[5091 rows x 9 columns]
```

Data Cleaning and Data Analysis

After data collection we do cleaning and Analysis of data set by many different way. First we collect data of Uttar Pradesh using data cleaning And then ally analysis for showing how dataset impact in history.



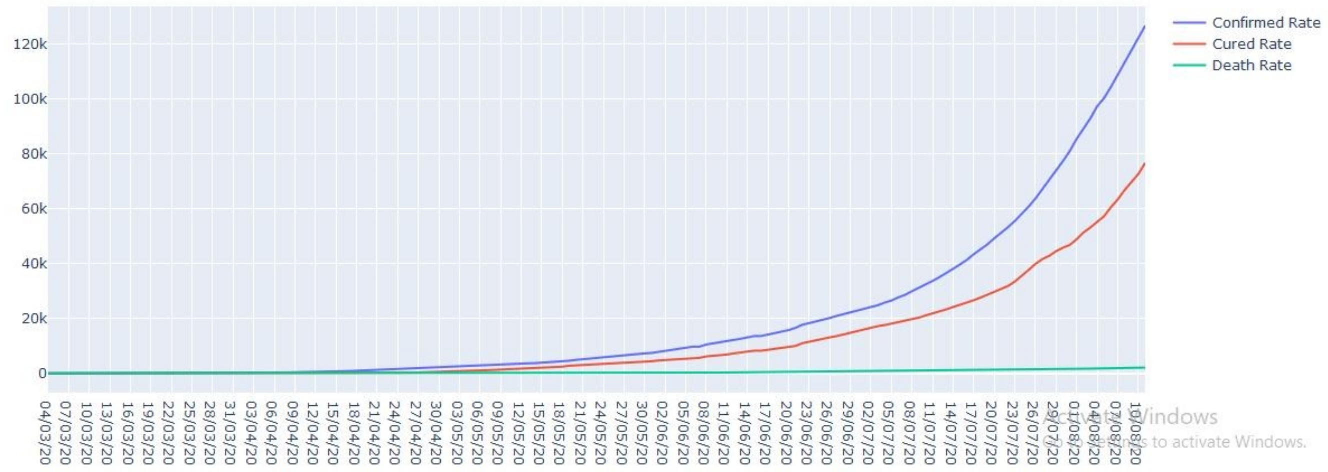
```
df_up=df.loc[(df['State/UnionTerritory']=='Uttar Pradesh')]
```

```
[ ]  df_up.dropna(inplace=True)
```

```
[ ]  df_up.tail(10)
```

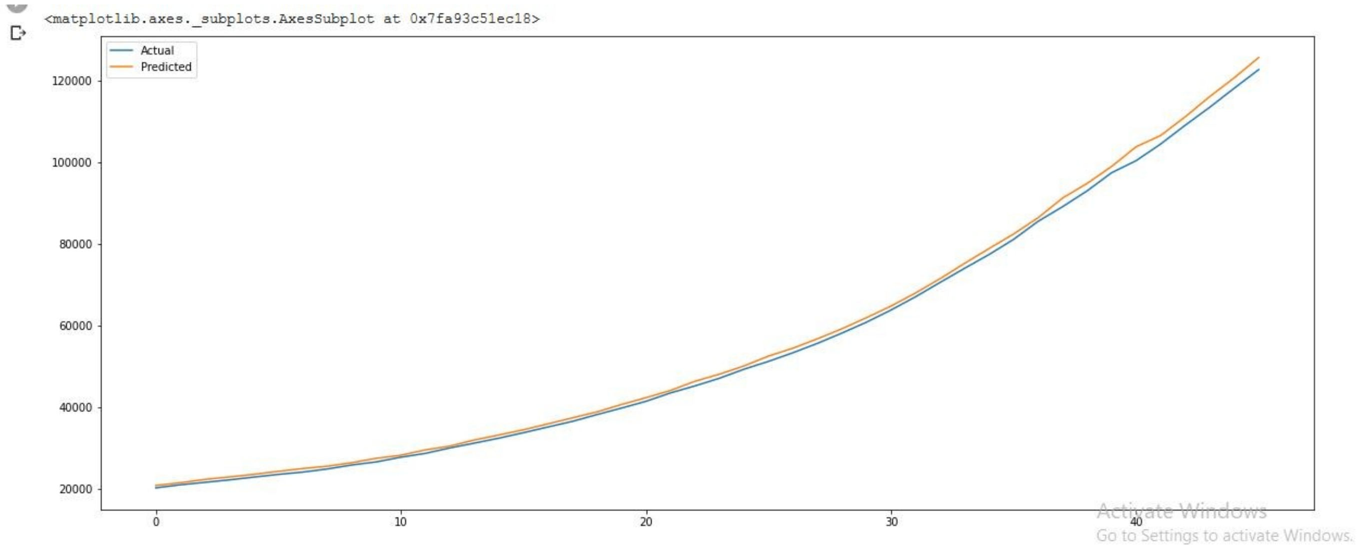
	Sno	Date	Time	State/UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNational	Cured	Deaths	Confirmed
4774	4775	02/08/20	8:00 AM	Uttar Pradesh	-	-	51334	1677	89048
4809	4810	03/08/20	8:00 AM	Uttar Pradesh	-	-	53168	1730	92921
4844	4845	04/08/20	8:00 AM	Uttar Pradesh	-	-	55393	1778	97362
4879	4880	05/08/20	8:00 AM	Uttar Pradesh	-	-	57271	1817	100310
4914	4915	06/08/20	8:00 AM	Uttar Pradesh	-	-	60558	1857	104388
4949	4950	07/08/20	8:00 AM	Uttar Pradesh	-	-	63402	1918	108974
4984	4985	08/08/20	8:00 AM	Uttar Pradesh	-	-	66834	1981	113378
5019	5020	09/08/20	8:00 AM	Uttar Pradesh	-	-	69833	2028	118038
5054	5055	10/08/20	8:00 AM	Uttar Pradesh	-	-	72650	2069	122609
5089	5090	11/08/20	8:00 AM	Uttar Pradesh	-	-	76724	2120	126722

Creating a graph for show Number of cases using plotly library.



Result of Prediction of Covid-19 prediction using SVR

Here the result of prediction of model it show the result how to cases are increases in the state.


```
<matplotlib.axes._subplots.AxesSubplot at 0x7fa93c51ec18>
```

Conclusion

The popularity of prediction model is increasing rapidly which encourage the researchers to find the new method for the Prediction. The forecasting not only help researcher but also help investors or any person dealing with in market.

In order to development of Diamonds prediction model for a regression model, we used kageel dataset for the diamond price prediction i.e., the dependent variable. Here Diamond price in US dollar we do multiple analysis and the build model for predict price using MLR and KNN-Regressor , Lasso.

In order to development of gold prediction model for a regression model, we used Yahoo finance dataset for the gold price prediction i.e., the dependent variable. Here gold price in US dollar we do multiple analysis and the build model for predict price using MLR and KNN-Regressor.

In order to development of Stock Market Price prediction good accuracy is required for this we have Recurrent Neural Network and LSTM based Stock market Prediction. That help investors and any person that dealing with stock market. Our initial analysis show significant correlation between different input parameter. And finally we have predicted the Google stock Price.

In order to development of Covid-19 UP India cases prediction we use a SVR model for predict the number of cases. That show how cases are increasing in state. Then what people can do for stop to spread of covid-19.

REFERENCES

[1] Nyce, Charles (2007), *Predictive Analytics White Paper (PDF)*, American Institute for Chartered Property Casualty Underwriters/Insurance Institute of America
[2] https://jupyter.org/

[3] Z. Sun, F. Chen, M. Chi, and Y. Zhu, “A spark-based big data platform for massive remote sensing data processing,” in Data Science, New York, NY, USA: Springer, pp. 120–126 2015.
[4] https://en.wikipedia.org/wiki/Predictive_modelling

[5] https://builtin.com/data-science/tour-top-10-algorithms-machine-learning-newbies

[6] https://www.investopedia.com/terms/m/mlr.asp

[7] https://www.digitalistmag.com/digital-economy/2018/03/15/differences-between-machine-learning-predictive-analytics-05977121/

[8] https://www.opensourceforu.com/2018/09/machine-learning-building-a-predictive-model-with-scikit-learn/

[9] https://blogs.gartner.com/jitendra-subramanyam/prediction-models-traditional-versus-machine-learning/

[10] Finlay "Predictive Analytics, Data Mining and Big Data: Myths, Misconceptions and Methods", 1st edition, New York: Palgrave Macmillan, 2014.

[11]. Joubert, D., 2003. The dollar and gold. http://www.sagolds.com

[12] Ismail, Z., F. Jamaluddin and F. Jamaludin, 2008. Time series regression model for forecasting Malaysian electricity load demand. Asian J. Math. Stat., 1: 139-149. DOI: 10.3923/ajms.2008.139.149

[13] Graham, S., 2001. The price of gold and stock price indices for the United States. http://www.gold.org/value/stats/research/