A REPORT

ON

DEMAND ESTIMATION MODEL FOR REAL-ESTATE RESIDENTIAL MARKET

By

Kushal Shah	**2018A7PS0254G**
Sarvesh Khetan	**2018A4PS0947H**
Sudarshan Mehta	**2018A3PS0579H**
Saksham Kumar	**2018A4PS0676H**
Suyash Tandon	**2018A7PS0271H**

At

A Practice School-1 station of

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

(June 2020)

A Report

ON

DEMAND ESTIMATION MODEL FOR REAL-ESTATE RESIDENTIAL MARKET

By

Kushal Shah	**2018A7PS0254G**	**Computer Science**
Sarvesh Khetan	**2018A4PS0947H**	**Mechanical**
Sudarshan Mehta	**2018A3PS0579H**	**EEE**
Saksham Kumar	**2018A4PS0676H**	**Mechanical**
Suyash Tandon	**2018A7PS0271H**	**Computer Science**

Prepared in partial fulfillment of the
Practice School-I Course Nos.
BITS C221/BITS C231/BITS C241

AT

A Practice School-I station of

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

(June 2020)

ACKNOWLEDGEMENT

We would like to extend our heartfelt gratitude to BITS administration, for providing us with the opportunity of a lifetime to work in one of the prominent and rapidly growing real-estate companies, TVS Emerald. We would like to take this opportunity to thank the several entities and personalities responsible for providing us with such a platform.

We would like to thank **Mr Sridhar VT**, Data Scientist, for taking time out of his busy schedule to guide us and enlighten us with our project and to solve our queries whenever we were stuck at some point.

We would like to extend our gratitude to **Prof. Swarna Chaudhary**, our PS instructors for her extremely insightful role in guiding us and for smoothly facilitating the collaboration. We would like to thank the **PS Division of BITS Pilani University** for taking such a brilliant and necessary initiative. We would like to thank everyone else who was directly or indirectly involved to provide us with this opportunity.

ABSTRACT SHEET

Practice School Division

Station: TVS Emerald **Centre:** Chennai

Duration: 6 weeks **Date of Start:** May 18, 2020

Date of Submission: June 26, 2020

Title of the Project: Demand estimation model for the residential real estate market

Name(s)	ID No.	Discipline(s) of the student(s):
Kushal Shah	2018A7PS0254G	B.E.(Hons.) Computer Science
Sarvesh Khetan	2018A4PS0947H	B.E.(Hons.) Mechanical
Sudarshan Mehta	2018A3PS0579H	B.E.(Hons.) EEE
Saksham Kumar	2018A4PS0676H	B.E.(Hons.) Mechanical
Suyash Tandon	2018A7PS0271H	B.E.(Hons.) Computer Science

Name(s) and designation(s) of the expert(s): Mr Sridhar VT, Data Scientist

Name(s) of the PS Faculty: Prof. Swarna Chaudhary

Keywords: Machine Learning, Linear Regression, Google collab, Python, Jupyter Notebook

Project Area: Data Science

Abstract: Data Science is the key to make use and get insights from data being collected from the start of the internet era. We can analyze various trends in the past data and through its help, we can predict outcomes. This report focuses on our attempt to incorporate data science tools to predict real-estate demand.

Signature(s) of Student(s) Signature of PS Faculty

Date: June 26, 2020

Table of Contents

1. Introduction

1.1 NEED FOR DATA SCIENCE

From the past few years, data is being generated at an enormously fast pace. According to recent statistics, social media usage in 2018 per minute can be summarized as follows

- Twitter users sent 473,400 tweets
- Snapchat users shared 2 million photos
- Instagram users posted 49,380 pictures
- LinkedIn gained 120 new users

Due to this enormous data, storage units like petabytes(1PB = 1024TB) and Zettabytes(1ZB=1024PB) were also introduced lately.

One might question how is this huge data being generated and what it comprises of? With the boom of the IT industry, data is being generated all the time via all individuals irrespective of the user being active or inactive. Let us take the example of Instagram, you are generating data by posting and commenting on posts but you are also producing data when you are inactive. Instagram analyzes your usage patterns and content consumed to keep you engaged by recommending posts that you are interested in.
Hence, data science is used to analyze this huge amount of data and get some useful information so that you can develop a business model on the same and make some profits.

1.2 DATA SCIENCE PROJECT LIFECYCLE

Building end to end data science is not an easy task and requires expertise in multiple domains. A Data Science project goes through many stages which can be broadly classified into 5 phases.

1.2.1 Data Extraction -

To predict something we need data on which we can train our model. This can be previously collected data or real-time data. The process of collecting data from different sources is called data extraction. There are various agencies both private and government which provide us with data.

1.2.2 Data Integration -

In this stage, we merge data collected from multiple sources in the data extraction stage and put them into one single file on which we can perform our next stage of the life cycle.

1.2.3 Data Analysis -

Once the data is cleaned and becomes ready to use we need to inspect the various features/attributes for discovering useful information, conclusions.

1.2.4 Data Modelling -

To predict something insightful from our data we need to implement machine learning algorithms. As per our requirements, we may train our model with data.

1.2.5 Data interpretation -

The potential of a model lies in its ability to generalise and find unseen patterns in the data. Adding to that we need to visualize and present our findings in such a way that it is clear to an audience with no technical knowledge.

1.3 TOOLS AVAILABLE

As said above, different data science fields have different tools to work with but python language is the epicentre cause all useful libraries are built around it.

Python libraries like **scrapy** and **beautiful soup** are used for data extraction. One can also extract data from **web scraping** and using tools like **Bloomberg** and **APIs**.

For data cleaning, python libraries such as **PrettyPandas, OpenRefine, tabulate, datacleaner,** etc could be used.

Data Analysis involves libraries like **Pandas**, **NumPy.** Big data analysis involves tools like **Apache Hadoop**

Data Modelling is used to implement machine learning and deep learning algorithms. For machine learning, we use libraries like **sklearn** and for deep learning, we use libraries like **TensorFlow**, **Keras** or **Pytorch**

For data interpretation, the tools used are **Matplotlib, Seaborn, ggplot, Tableau,** etc which aid in visualizing data.

1.4 IDE(S) AVAILABLE

There are numerous ways via which one can code python and run same.These include web based ide(s) and local machine based ide(s) some of which are listed below

- > **Anaconda**
- > **Jupyter notebook**
- > **Spyder**
- > **Pycharm**
- > **Sublime text**
- > **Atom**
- > **VS Code**
- > **Google Collab**

2. Indian Real-Estate Market

2.1 CURRENT SCENARIO

Real-estate is one of the most prominent sectors globally. The real estate sector comprises four sub-sectors - housing, retail, hospitality, and commercial.

It is expected that this sector will incur more non-resident Indian (NRI) investments in both the short term and the long term. Bengaluru is expected to be the most favoured property investment destination for NRIs, followed by Ahmedabad, Pune, Chennai, Goa, Delhi, and Dehradun.

By 2040, the real estate market will grow to Rs 65,000 crore (US$ 9.30 billion) from Rs 12,000 crore (US$ 1.72 billion) in 2019. In India, real estate is the second-largest employer after agriculture and will contribute about 13 percent of the country's GDP by 2025.

2.2 ROLE OF DATA SCIENCE

Now as seen above the real-estate market has a tremendous scope of reaching great heights and one needs to select the correct land and invest in the same to be a part of this. Selecting correct land to invest is not an easy take because one needs to take into account multiple factors which determine the demand of the land.

Using data science one can automate this process of selecting the correct land and thus increase its profit. Currently, TVS Emerald doesn't have a proper framework for buying lands based on different factors and in this project we are trying to build this system wherein we need to predict the demand at the micro-market level for residential real estate using external data sources which helps us identify lands at hotspot locations with a good product mix & cost-effectiveness.

3. Project Description

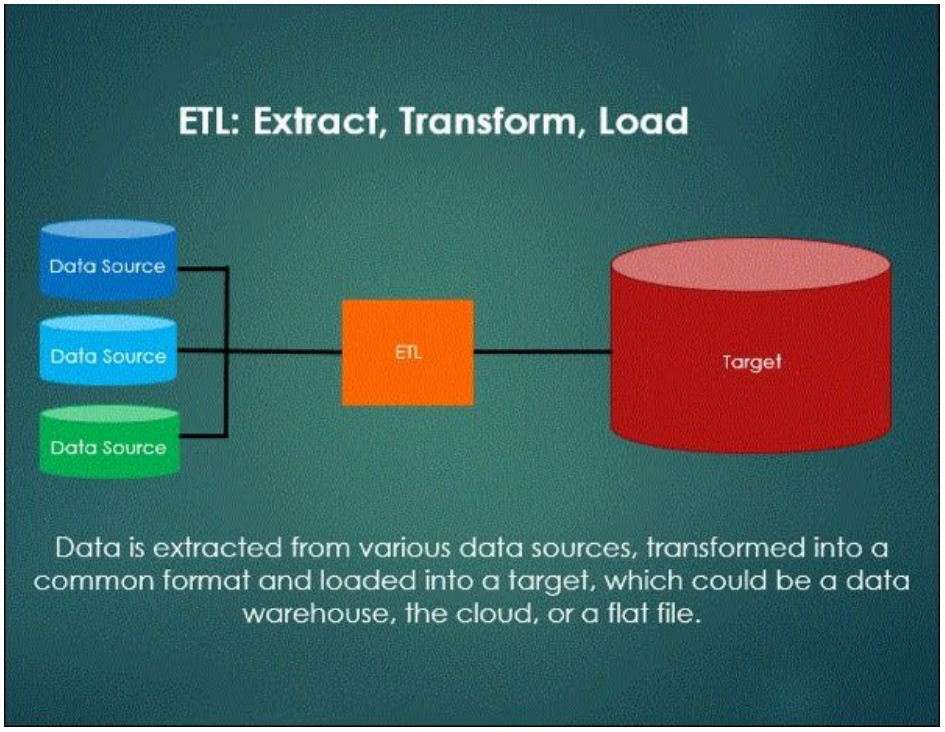
Now based on the data science project life cycle we have also divided our project into the same lifecycle.

DATA EXTRACTION:

Firstly, in the first two weeks of our PS, we determined various factors on which demand of a real estate household depends upon. Based on these collected factors our mentor provided us with multiple datasets collected from sources which TVS Emerald had a subscription to.

DATA INTEGRATION:

This dataset was provided in multiple CSV file format. In the third week, we were assigned the task to merge all these CSV files into one single file on which data analysis could be performed. This was done using the pandas library in python language.



DATA ANALYSIS: Data analysis can be further bifurcated into two major steps: **Feature Engineering** and **Feature Selection**. Feature Engineering is also called the **Data pre-processing** stage.

FEATURE ENGINEERING

In the fourth week of our PS, we performed feature engineering on our dataset. Feature engineering involves tasks such as handling missing values and outliers, converting categorical data into numerical data using feature encoding techniques and many more. Below we will describe all the step that we did in detail.

To perform all the above-stated feature engineering techniques we will use python libraries like **pandas** and **NumPy**

----------Feature encoding----------

Cause of problem:
One cannot train a mathematical model based on categorical(string) type variables because computers can only understand numbers. Since our dataset has multiple categorical variables we will convert them to numerical variables. This is called Feature Encoding.

Handling the problem:
There are numerous ways to do feature encoding but we used **one hot encoding** method to implement feature encoding on our dataset.A small gist of the approach is shown below.

REVIEW	GOOD	V.GOOD	EXCELLENT
good	1	0	0
bad	0	0	0
bad	0	0	0
v.good	0	1	0

good	1	0	0
excellent	0	0	1
good	1	0	0
v.good	0	1	0
bad	0	0	0
excellent	0	0	1

Hence you can see that all the categorical variables available in animals column have been converted into numerical data. The important thing to note is that there is no column for bad, because if all the columns with numerical values are 0 simply means that it is bad. Hence in one-hot encoding, if we have n-type of categories then n-1 new columns are generated.

The disadvantage of one-hot encoding is that if our categorical data consists of hundreds and thousands of categories then this method leads to the introduction of many variables in your dataset thus affecting the later results.

Alternative methods available:

- Ordinal numbering encoding
- Count or frequency encoding
- Target guided ordinal encoding
- Mean Encoding
- Probability ration encoding

----------Handling Missing Values----------

Cause of problem:
Imagine for example that the data comes from a survey, and the data are entered manually into an online form. The person entering the data could easily forget to complete a field in the form, and therefore, that value for that form would be missing. This is the sole cause of having missing data in our dataset. In python, the missing values are stored as

Nan. One cannot pass null values to the model and hence we will have to handle missing values from our dataset.

Handling the problem:

There are numerous ways to handle missing values but we have used the **mean_median imputation** to handle missing data in our dataset. A small gist of the approach is shown below.

12	15	Nan	19	10	Nan	27	56	24	09

Now suppose our aim is to remove all the Nan in the above column, the first step is to take the average of all the available data.

Average(Mean) = 12+15+19+10+27+56+24+09 / 8
= 21.5

Now we replace all the Nans present in that column with this 21.5 value.

12	15	21.5	19	10	21.5	27	56	24	09

Hence now our dataset does not contain any missing values and hence all the missing values have been handled successfully. Here we have only considered for one column, but in actual data, it is done for each column. There is a function available in pandas to do this, we need not do it manually.

Alternative methods available:

- Complete Case Analysis (CCA)
- Random Sample Imputation
- End of Distribution Imputation
- Frequent Category Imputation

----------Feature scaling----------

Cause of problem:
Suppose a feature in a dataset has a value between 10-100 and another feature has value between 1000-10000, this is a major issue because this disparity in our dataset causes slow learning of data in the data modelling stage and hence we need to bring uniformity. For a given column we have a value and each value has a magnitude and a unit. Say height has a value 170cm then here 170 is magnitude and cm is the unit. Our main aim is to scale down all this value like converting cm to mm by dividing all values by 10.This is one of the methods.

Handling the Problem:
Consider the following two columns of a dataset. We can do feature scaling using a technique called **standardization**, the main idea here is to convert the distribution of the given column into standard normal distribution.

40	90	120	76	15	7	68	46	11	99

Mean of the above distribution is = 49.1
Deviation of the above distribution is = 30.69

Now to convert above distribution into standard normal distribution we should subtract the mean from each value and then divide by standard deviation

-0.29	1.33	2.3	0.88	-1.11	-1.37	0.62	-0.10	-1.24	1.63

Hence we have successfully scaled-down all the values

FEATURE SELECTION

The fifth week of our PS was dedicated to feature selection. Feature selection is an important step to be performed because we cannot train our model on all the available features, this might hamper our end results hence we are supposed to select only those features that truly affect our model accuracy. This is what is feature selection. There are numerous ways via which one can do feature selection, some of them are listed below

1. Using Correlation and Covariance
2. Using Principal Component Analysis (PCA)
3. Using LDA
4. Using Factor Analysis
5. Using Regression Algorithms

----------CORRELATION AND COVARIANCE----------

In this method, our aim is to calculate the correlation between any two variables of the dataset. This can be done both mathematically and graphically. Consider the following dataset

10	20	30	40	50	60	70	80	90	100
7	24	41	45	59	66	79	84	99	55

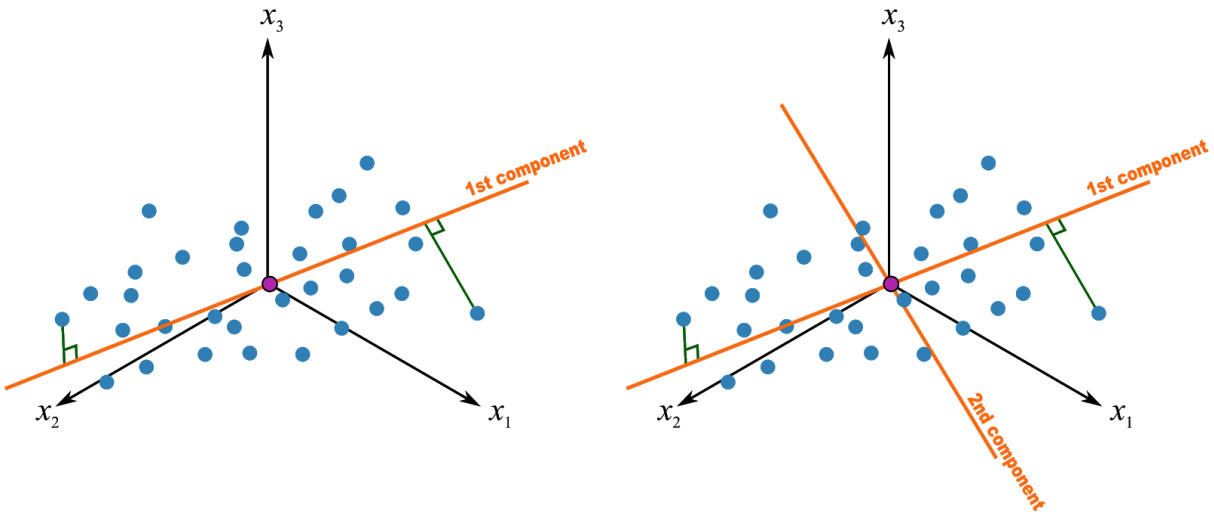
To calculate covariance we use Pearson correlation formula. It is represented by R.When you calculate Pearson correlation for above we find it to be 0.93. Pearson correlation value ranges from -1 to 1.

If R is equal to 1 means they are highly correlated while if it is -1 means they are highly not correlated. This same thing can also be represented in the form of a graph

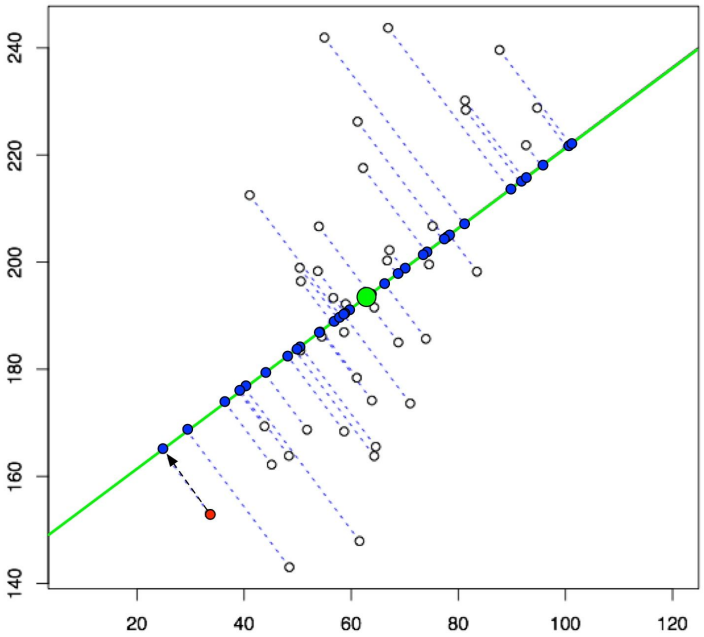
----------PRINCIPAL COMPONENT ANALYSIS (PCA)----------

Using this we can combine K variables into N variables, where N<K. Now we saw in the above approach that if two variables have high covariance then we can drop one of them. Now deciding which one to drop is not an easy task, hence instead of dropping either of the two variables, we can combine them into one. This is done by taking projections on the required axis.

3D to 2D



2D to 1D



Similarly higher order can also be cut down to lower order using this projection technique.

----------REGRESSION ALGORITHMS----------

Feature selection is a very critical step. When presented data with very high dimensionality, models usually choke because:

●**Training time** increases exponentially with the number of features
●Models have an increased risk of **overfitting** with increasing features

Feature Selection methods help with these problems by reducing the dimensions without much loss of the total information.

As our cleaned data file consisted of more than 250 features so we performed 5 different feature selection methods on the data :

- **Linear Regression**
- **Ridge regression**
- **Lasso regression**
- **Random Forest**
- **Gradient Boosting method**

Each of the 5 methods selected some of the features. We stored these selected features as a list & then merged these lists to select top 50 most frequent features for our time series forecasting model.

1. Linear Regression

- Linear Regression is used to model a relationship between two variables by fitting a linear equation to the observed data.
- It establishes a relationship between various attributes of the dataset and the dependent variable.
- Thus it yields an equation of the following form:

$$Y = b_0 + b_1x_1 + b_2x_2 + ... + b_nx_n$$

- Here Y is the independent variable whereas x1,x2,x3,.....,xn are the different attributes in the dataset and their coefficients have different values where higher value implies that the associated attribute has higher significance in comparison to the ones having lower coefficient value.
- Now, the coefficients b1, b2, b3,....., bn are mainly taken into consideration.
- The coefficient having a value greater than the mean of all the coefficients are taken for feature selection.
- The problem with this way of feature selection is that it has an overfitting problem.
- Thus the model is not generalized hence we need to apply regularization techniques in order to deal with the overfitting problem.

2. Ridge Regression

- Ridge regression is a regularization technique that is helpful in predicting the attributes which have high importance and has a high impact on the dependent variable.
- It helps to reduce the overfitting problem, complexity problem, and the problem of the interpretability of the model.
- It optimizes the model by adding a penalty to the cost function of ridge regression.
- The Cost function in case of ridge regression looks like the following:

$$Cost\ (W) = RSS(W) + \lambda * (sum\ of\ squares\ of\ weights)$$

$$= \sum_{i=1}^{N} \left\{ y_i - \sum_{j=0}^{M} w_j\, x_{ij} \right\}^2 + \lambda \sum_{j=0}^{M} w_j^2$$

- Here, RSS is the residual squared sum, and lambda is used to adjust the number of attributes to be used.
- Lambda is used to adjust the cost function which in turn is used to adjust the number of attributes that strongly affect the dependent variable.
- Here the penalty function is lambda multiplied by the sum of the square of the coefficients.
- What happens in Ridge regression is that it reduces each and every coefficient thereby it makes the model computationally viable.
- So in total ridge regression helps us to select features, solve the overfitting problem, improves the accuracy, reduces the complexity, and improves the interpretability of the model.
- It also helps in generalizing the model.

3. Lasso Regression

- The acronym “LASSO” stands for **L**east **A**bsolute **S**hrinkage and **S**election **O**perator.
- Lasso is a regression analysis method that performs both variable selection and regularization in order to enhance the prediction accuracy and interpretability of the statistical model it produces.
- Lasso regression performs **L1 regularization**, which adds a penalty equal to the absolute value of the magnitude of coefficients. This type of regularization can result in sparse models with few coefficients; Some coefficients can become zero and eliminate from the model. Larger penalties result in coefficient values closer to zero, which is ideal for producing simpler models.

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

adds penalty equivalent to the **absolute value of the magnitude** of coefficients

Minimization objective = LS Obj + α * (sum of absolute value of coefficients)

here ‘LS Obj’ refers to ‘least-squares objective’, i.e. the linear regression objective without regularization.

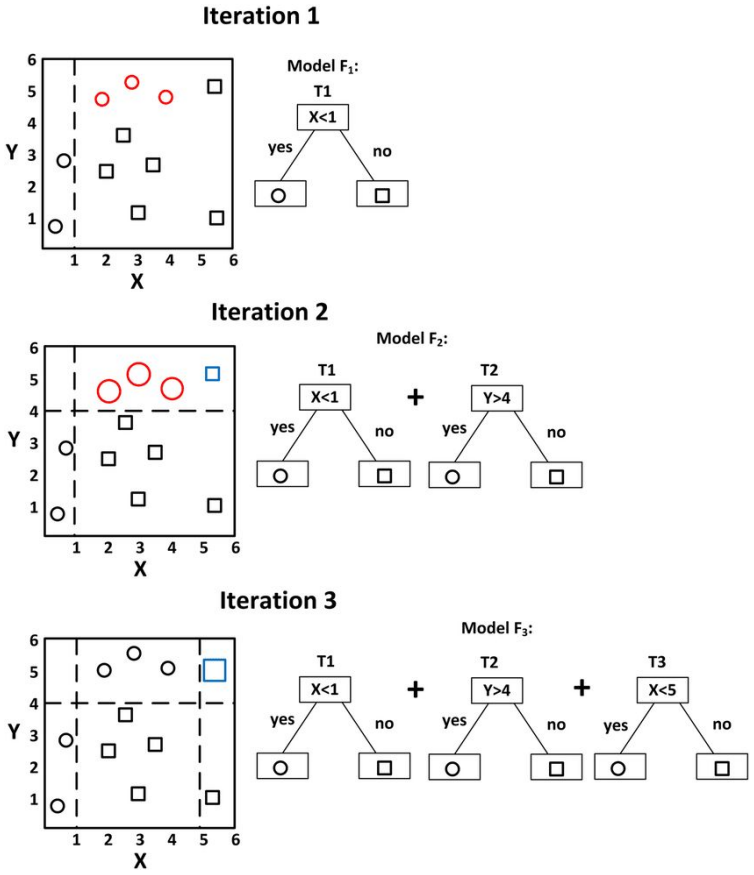
- A **tuning parameter**, λ controls the strength of the L1 penalty. λ is basically the amount of shrinkage:
    1. When λ = 0, no parameters are eliminated. The estimate is equal to the one found with linear regression.
    2. As λ increases, more and more coefficients are set to zero and eliminated (theoretically, when λ = ∞, *all* coefficients are eliminated).
    3. As λ increases, bias increases.
    4. As λ decreases, variance increases.

4. Random Forest

- Data science provides a variety of classification algorithms such as logistic regression, support vector machine, naive Bayes classifier, and decision trees. But at the top of the classifier hierarchy is the random forest classifier.
- Decision trees are the building blocks of the random forest model
- Random forest, like its name implies, consists of a large number of individual decision trees that work as an ensemble. Each tree in the random forest spits out a class prediction and the class with the most votes becomes our model's prediction.
- The low correlation between models is the key.
- Uncorrelated models produce ensemble predictions that are more accurate than any of the individual predictions.
- While some trees can be wrong, many other trees will be right, so as a group the trees are able to move in the correct direction. So the prerequisites for the random forest to perform well are:
  1. There needs to be some signal in our features so that models built using those features do better than random guessing.
  2. The predictions (and therefore the errors) made by the individual trees need to have low correlations with each other.
- It takes into account bagging and features randomness when building each individual tree to try to create an uncorrelated forest of trees whose prediction by committee is more accurate than that of any individual tree.
- Overall, the random forest is generally a fast, simple, and flexible tool, but not without some limitations.

5. Gradient Boosting method

- Gradient boosting is another method that uses an ensemble of decision trees to create an accurate model.
- Initially a weak decision tree is created which predicts but highly underfits the labeled data.
- Then a succession of trees is created which learns from the mistakes of their predecessor and improves the accuracy.
- The learning rate of the model is input by the user, however the predicament is that if a high learning rate is input, a compromise will be made on the features used to distinguish between different data points.
- Overall, it is a highly accurate model and its computation time depends on the learning rate input by the user and max_depth allowed for the ensemble of decision trees.



DATA MODELLING: Time Series Forecasting-

In the final week, we were left with the time series forecasting task. Since our target feature was dependent on time variables, we had to solve a time series forecasting problem, and more specifically it was a **multivariate time series forecasting** problem because our target feature was dependent on multiple independent variables of which one was time. To solve such a problem we have numerous machine learning algorithms like:

1. VAR (Vector AutoRegression)
2. ARIMAX
3. VARMAX

VAR and VARMAX are used in multivariate time series and ARIMA models are used in univariate time series.

Why we used multivariate:

A univariate time series, as the name suggests, is a series with a single time-dependent variable whereas a multivariate time series has more than one time-dependent variable. Each variable depends not only on its past values but also has some dependency on other variables. This dependency is used for forecasting future values.

What are VAR models:

In a VAR model, each variable is a linear function of the past values of itself and the past values of all the other variables. To explain this in a better manner, I'm going to use a simple visual example:

We have two variables, y1 and y2. We need to forecast the value of these two variables at time t, from the given data for past n values. For simplicity, I have considered the lag value to be 1.

Variable y1	Variable y2	Variable y1	Variable y2
$y1_{t-n}$	$y2_{t-n}$	$y1_{t-n}$	$y2_{t-n}$
...	...	...	...
$Y1_{t-2}$	$Y2_{t-2}$	$Y1_{t-2}$	$Y2_{t-2}$
$Y1_{t-1}$	$Y2_{t-1}$	$Y1_{t-1}$	$Y2_{t-1}$
$y1_{t}$	$y2_{t}$	$y1_{t}$	$y2_{t}$

For calculating y1(t), we will use the past value of y1 and y2. Similarly, to calculate y2(t), past values of both y1 and y2 will be used. Below is a simple mathematical way of representing this relation:

$$y_1(t) = a_1 + w_1 11 * y_1(t-1) + w_1 12 * y_2(t-1) + e_1(t-1)$$

$$y_2(t) = a_2 + w_1 21 * y_1(t-1) + w_1 22 * y_2(t-1) + e_2(t-1)$$

Here,

- a1 and a2 are the constant terms,
- w11, w12, w21, and w22 are the coefficients,
- e1 and e2 are the error terms

VAR is able to understand and use the relationship between several variables. This is useful for describing the dynamic behavior of the data and also provides better forecasting results. Additionally, implementing VAR is as simple as using any other univariate technique

To forecast any time series the data should be stationary(A stationary series is one in which the properties – mean, variance and covariance, do not vary with time) so to check the stationarity of a multivariate time series we use the Johansen test(similar to Augmented Dickey-Fuller test for univariate series). If the score of the test for every variable was less than one then the time series is stationary and can be used for forecasting.

Once we got the forecasted time series we utilized our models created on our original datasets to predict future demand and to compare it with forecasted demand(found again by forecasting), if the accuracy was greater than 80% the series was well developed or if it was less than that we tweaked the time series until we reached the wanted accuracy.

4. Conclusion

Our main aim is to create a predictive model with 80% accuracy based on pre-collected data. To achieve the same we will be using machine learning algorithms like linear regression and python libraries like pandas, NumPy, scikit, Matplotlib, and sklearn.

Data analysis is a crucial step in making a predictive model and about 90% of the entire project time is devoted to the same. It is important to train our model on a good pre-processed data set so that our model can train faster and as efficiently as possible. After the model is created we would test it with some test data and compare the results, after which if there is room for improvement we will try to improve the accuracy of the model before deploying.

This model will predict the demand at the micro-market level for the residential real estate market using external data sources which will help identify lands at hotspot locations with a good product mix & cost-effectiveness.

Housing prices are an important reflection of the economy, and housing price demand is of great interest for real estate firms like TVS Emerald. In this project, house demand will be predicted given explanatory variables that cover many aspects of residential houses.

5. Bibliography

WEBSITES / LINKS / BLOGS

- https://www.ibef.org/industry/real-estate-india.aspx
- https://www.cbre.co.in/en/research-reports/India-Real-Estate-Market-Outlook-2019
- https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/
- https://www.analyticsvidhya.com/blog/2015/09/hypothesis-testing-explained/
- https://towardsdatascience.com/everything-you-need-to-know-about-hypothesis-testing-part-i-4de9abebbc8a
- https://www.researchgate.net/publication/318702207_Hypotheses_and_Hypothesis_Testing
- https://www.researchgate.net/publication/325846748_FORMULATING_AND_TESTING_HYPOTHESIS
- https://towardsdatascience.com/multivariate-time-series-forecasting-653372b3db36
- https://towardsdatascience.com/prediction-task-with-multivariate-timeseries-and-var-model-47003f629f9
- https://www.analyticsvidhya.com/blog/2018/09/multivariate-time-series-guide-forecasting-modeling-python-codes/
- https://blog.edugrad.com/forecasting-and-modeling-with-a-multivariate-time-series-in-python/
- https://analyticsindiamag.com/multivariate-time-series-analysis-for-data-science-rookies/
- https://www.researchgate.net/publication/330431870_Advanced_Multivariate_Time_Series_Forecasting_Models
- https://www.statisticshowto.com/lasso-regression/#:~:text=Lasso%20regression%20is%20a%20type,i.e.%20models%20with%20fewer%20parameters).

6. GLOSSARY

- **Machine Learning**: Machine learning is the study of computer algorithms that improve automatically through experience. It is seen as a subset of artificial intelligence.

- **Regression**: is a linear approach to modelling the relationship between a scalar response and one or more explanatory variables. The case of one explanatory variable is called simple linear regression. For more than one explanatory variable, the process is called multiple linear regression.

- **Real-Estate**: is property consisting of land and the buildings on it, along with its natural resources such as crops, minerals, or water; immovable property of this nature; an interest vested in this an item of real property, buildings or housing in general.

- **Time Series Forecasting**: is the use of a model to predict future values based on previously observed values. Time series are widely used for non-stationary data, like economics, weather, stock price, and retail sales in this post.

- **One Hot Encoding:** It refers to splitting the column which contains numerical *categorical data* to many columns depending on the number of categories present in that column. Each column contains “0” or “1” corresponding to which column it has been placed.

- **Feature selection:** The process where we select those features which contribute most to our prediction variable or output in which we are interested in. Having irrelevant features in our data can decrease the accuracy of the models and make our model learn based on irrelevant features.