Advanced Topics in Big Data Analytics

Project in Data Science by –Harshita Bhavesh Patil

Predict Sale Value of Car Using Regression Analysis

ABSTRACT:

Data science is in immense vogue to business nowadays. Hal Varian (chief economist, Google) stated, "The ability to take data—to be able to understand it, to process it, to extract value from it, to visualize it, to communicate it—that's going to be a hugely important skill in the next decades… (Varian, 2009). And his statement has completely come to true. Data science has become so important that everyone in business is talking about it.

Understanding the power of data science, this project proposes the implementation of data science for car sales business to predict value of used car for sale. Predicting the value of product in business helps business to gain the sales which are likely to be missed and allows to plan the business growth by making intelligent business decisions. The predictions are based on historical data collected from advertisements on cars. Different techniques like multiple linear regression analysis, k-nearest neighbors, support vector machine and neural network have been used to make the predictions. The predictions are then evaluated and compared in order to find those which provide the best performances. All the four methods provided comparable performance.

This paper details about the complete process of project including data sourcing, data cleaning, data visualization, data storage technique in the big data environment, modeling using different techniques, evaluating metrics of models, comparing results, choosing the right technique and conclusion.

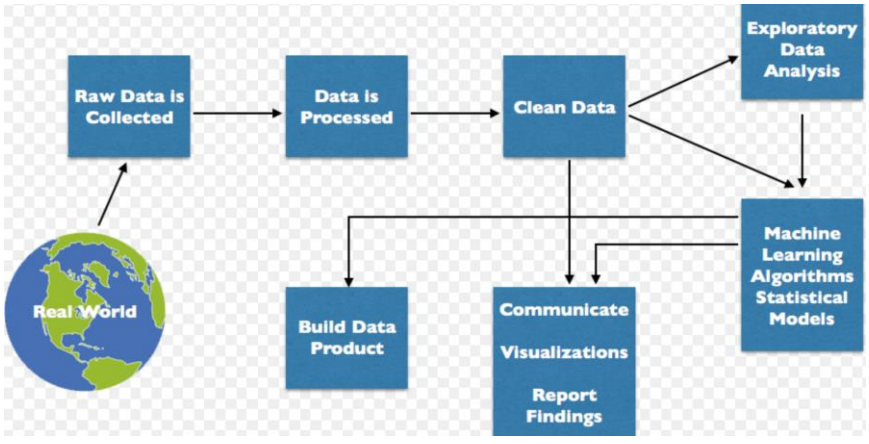


Figure 1: The complete process

DATA REQUIREMENT:

The data is necessary as inputs to the analysis. It is that data which is going to derive us our model. Hence it must be as perfect as possible so as to get the accurate model.

In order to derive the model that can predict the value of car for sales, multiple machine learning technique can be used. These techniques attempt to model the relationship between two or more explanatory variables and a response variable by fitting equation to observed data. These are powerful techniques used for predicting the unknown value of a variable from the known value of two or more variables- also called the predictors. Thus, to apply it to our project we are in search of variables/predictors data that can help us to derive sale value of car. The predictors here can be car manufacture brand, car body type, model, drive type and many more.

DATA SOURCING:

Data sourcing in any data science project is an integral part. Data source must be highly reliable because it is going to get us the data based on which we are going to derive the model. Data can be collected from various sources but the sources needs to be reliable as said. There are two sources of data collection techniques – primary and secondary data collection technique. Primary data collection uses surveys, experiments or direct observations whereas secondary data collection is the collected information from diverse source of documents or electronically stored information. Primary data is more reliable, authentic and objective because it has been collected specifically for the purpose in mind. As its validity is greater, we decided to gather the primary data from advertisements with a purpose of using it for deriving a car value prediction model. As mentioned in Data requirements, we needed data of variables that can help us predict car value, advertisements on car sales seems to be the best source as it not only contains all variables but also its value. The dataset was collected from car sale advertisements for study.

ABOUT DATA:

The dataset contains data for more than 9500 cars. Most of them are used cars so it opens the possibility to analyze features related to car operation.

Dataset contains 9576 rows and 10 variables with essential meanings:

1. **car**: The manufacturer brand of car
2. **price**: seller's price in advertisement (in USD), ranges from $259 to $547,800
3. **body**: car body type e.g., Sedan, Crossover
4. **mileage**: mileage of car as mentioned in advertisement ('000 Km)

5. **engV**: rounded engine volume (‘000 cubic cm) of car
6. **engType**: type of fuel (“Other” in this case should be treated as NA) the car consumes e.g., Gas, Petrol and Diesel
7. **registration**: whether car is registered or not
8. **year**: year of production of car ranges from 1953 to 2016
9. **model**: specific model name of car
10. **drive**: drive type of car e.g., rear, full and front.

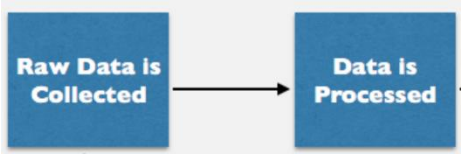
	car	price	body	mileage	engV	engType	registration	year	model	drive
1	Ford	15500.000	crossover	68	2.50	Gas	yes	2010	Kuga	full
2	Mercedes-Benz	20500.000	sedan	173	1.80	Gas	yes	2011	E-Class	rear
3	Mercedes-Benz	35000.000	*NA*	135	5.50	Petrol	yes	2008	CL 550	rear
4	Mercedes-Benz	17800.000	van	162	1.80	Diesel	yes	2012	B 180	front
5	Mercedes-Benz	33000.000	vagon	91	*NA*	*NA*	yes	2013	E-Class	*NA*

Table 1: The overview of data

We can now build a model to predict the unknown value of a variable i.e. predicted price of car from the known value of above listed variables. But dataset has gaps i.e. not all variables in all rows necessarily has value. Few does not have value and gap do exist. Also with small data sets, noise and outliers are especially troublesome. If we build model on such datasets then it won’t get us sensible model. Hence cleaning of data is very crucial.

DATA CLEANING:

Data cleaning or data cleansing refers to detection and correction of corrupt or inaccurate records from datasets to make it consistent. This can include replacing, modifying, or deleting the coarse data.



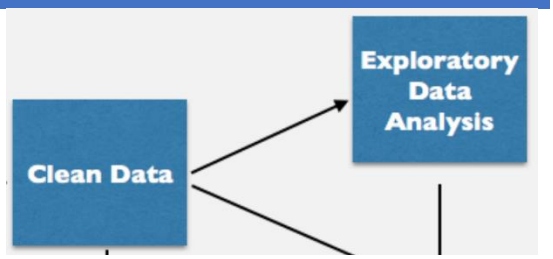
For our dataset, we need to treat the missing data. On analysis it was found that below variables has mentioned number of missing values in records.

Variable	Number of missing value	Variable	Number of missing value
Car	0	engType	462
price	0	registration	0
body	838	year	0
mileage	0	model	0
engV	462	drive	511

Table 2: Detail on missing value

Imputation is the process of replacing missing data with substituted values. Using the function KNN (k-Nearest Neighbor Imputation) we imputated the missing values. K-Nearest Neighbor Imputation is based on a variation of the Gower Distance for numerical, categorical, ordered and semi-continuous variables. Also, it was identified for 267 records that the price of car is $0, which is impossible. Hence all such rows were removed. The dataset now has 9309 unique records with 87 makes and 888 different models of cars.

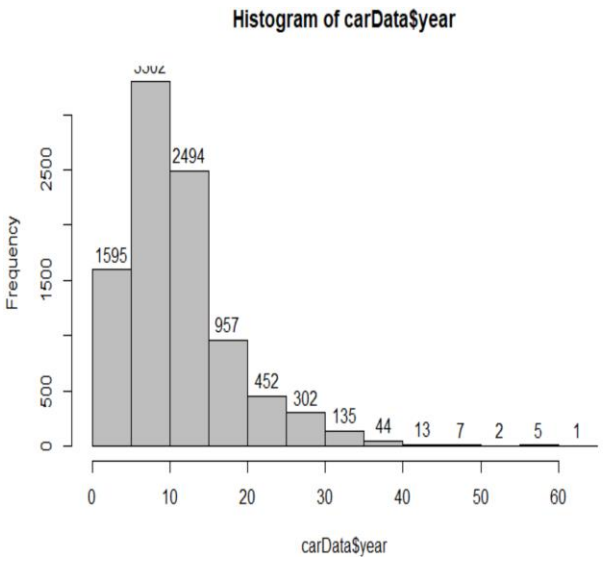
DATA TRANSFORMATION:

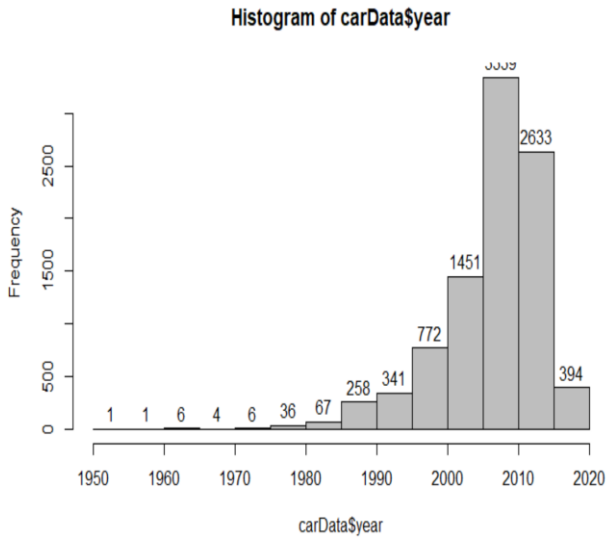


In our dataset, we have variables who has character value. But in order to facilitate algorithm training i.e. to generate model, we would need numerical value. Hence, we need to transform the character value of variables into numerical value. To do that, we reordered the levels of the variable by the mean of its price and then transformed them to numeric value. This way the value for each variable that we get helps us in quantifying the impact of each of its input. For example, consider the variable car which contains value like Volkswagen, Fisker, Bentley, Toyota and so on. When we reorder it by the mean of its price and transform to numeric value, we get the order in which GAZ is first having lowest numeric value and Bentley is last having highest numeric value. Thus, the numeric value of the car manufacturer that we derived helps us to quantify the impact while generating model. Similarly, we did same for all the other variables with character value and derived their numerical value. The transformed data now look like as –

	car	price	body	mileage	engV	engType	registration	year	model	drive
1	28	15500.000	1	68	2.50	2	1	8	504	2
2	53	20500.000	3	173	1.80	2	1	7	337	3
3	53	35000.000	3	135	5.50	3	1	10	273	3
4	53	17800.000	5	162	1.80	1	1	6	213	1
5	53	33000.000	4	91	2.20	1	1	5	337	3

Table 3: The overview of transformed data





DATA STORAGE:

Big data storage demands very large capacity and high processing performance with real-time or near real-time responses. The technique selected to store data should reflect the application and its usage pattern. Traditional data storage techniques supported inflexible storage infrastructure and their operations mined the homogeneous datasets. However, a web analytics work culture of big data analysis demands low latency access to very large number of small files, where scale-out storage consists of a number of compute-and-storage elements, where capacity and performance can be added in relatively small increments.

To understand the big data storage methods, one first needs to understand the amount and type of data they have along with the motivation behind storing the information. Look for a solution that fits data, not the other way around. The storage options vary from flash memory thumb drives to network-attached storage, depending upon the size of data.

However, the dataset of our project which we will be using to derive car price prediction model is of size 508 KB, which is very small and can't be considered as big data. Hence, we don't need the big data storage techniques to store our data at this stage as the hard disk storage in system is enough for its storage and meets the performance needed with respect to response. But that doesn't mean, the system will always suffice to store our data. As the project scope increases, we will need more data for analysis. The data would be collected from various other sources besides advertisements. When the model would actually be implemented by organization, the data for analysis would be huge which definitely will need the big data environment storage system. Considering the real-life scenario and the fact that analysis would involve big data, we need to look for the big data storage solution.

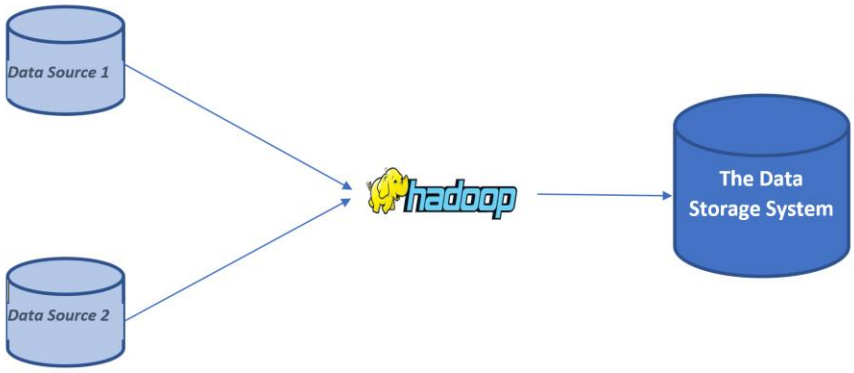
APACHE HADOOP:

Apache Hadoop offers a scalable, flexible and reliable distributed computing big data framework for a cluster of systems with storage capacity and local computing power by leveraging commodity hardware. Hadoop follows a Master Slave architecture for the transformation and analysis of large datasets using Hadoop MapReduce paradigm.

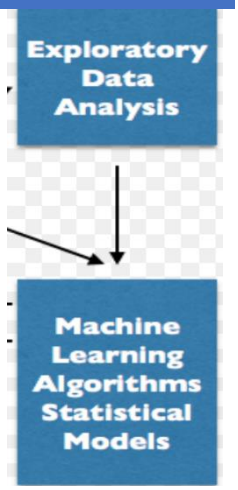
Hadoop has three core components:

- Hadoop Distributed File System (HDFS)
- MapReduce
- Yet Another Resource Negotiator (YARN)

Using the above three components Hadoop facilitates our project by faster data processing. Data processing speed obtained is faster than traditional ETL processing because of its parallel processing capabilities, which can perform jobs ten times faster than those running on a single thread server or on the mainframe. By implementing the Hadoop architecture on data storage platforms would benefit us to ease the process of big data analytics, reduces operational costs, and quickens the time to market.



PREDICTIVE MODELING:



Model that can forecast outcome based on what has happened in the past can be called as Predictive model. Regression predictive models determine the relationship between a dependent or target variable and an independent variable or predictor. For car sales business, to predict the sale value of car, we need one such model that is derived from number of predictors, which are variables that are likely to influence future results. There are various regression algorithms as below which are used for Predictive Modeling and has a wide spectrum of potential applications.

- Multiple linear regression/ Generalized Linear Model – GLM
- K- Nearest Neighbors – KNN
- Support Vector Machine – SVM
- Neural Network – NN

WHY REGRESSION?

Regression analysis is a statistical analysis, where given a set of independent variables, we can predict the outcome of a dependent variable. It is used to predict the result of a quantitative (numerical) variable. The idea here is to fit our data through a regression line so significantly that it can predict the output at any given point. Regression analysis helps us understand the relationship between dependent and independent variables i.e. how does the dependent variable vary when there are some changes made to the independent variable.

We divided dataset into training dataset and validation dataset. Training dataset being the 80% of the total dataset to train the models and validation dataset being the remaining 20% to validate and test the models. We used 10- fold repeated cross validation resampling technique to generate machine learning model.

K FOLD - CROSS VALIDATION TECHNIQUE:

K fold - Cross-validation is a resampling procedure used to evaluate machine learning models on a limited data sample (training data). Cross-validation is used in applied machine learning to estimate the skill of a machine learning model on unseen data i.e., to use a limited sample in order to estimate how the model is expected to perform in general when used to make predictions on data not used during the training of the model.

The general procedure of resampling in cross validation technique is as follows:

1. Shuffle the dataset randomly.
2. Split the dataset into k groups.
3. For each unique group:

i. Take the group as a hold out or test data set.
ii. Take the remaining groups as a training data set.
iii. Fit a model on the training set and evaluate it on the test set.
iv. Retain the evaluation score and discard the model.

4. Summarize the skill of the model using the sample of model evaluation scores

Importantly, each observation in the data sample is assigned to an individual group and stays in that group for the duration of the procedure. This means that each sample is given the opportunity to be used in the hold out set 1 time and used to train the model k-1 times.

As our dataset is very small we applied 10-fold cross validation technique to generate the model.

MULTIPLE LINEAR REGRESSION/ GENERALIZED LINEAR MODEL – GLM:

Multiple linear regression attempts to model the relationship between two or more explanatory variables and a response variable by fitting a linear equation to observed data. It is a powerful technique used for predicting the unknown value of a variable from the known value of two or more variables- also called the predictors.

$$Y = b_0 + b_1 X_1 + b_2 X_2 + \ldots\ldots\ldots\ldots\ldots\ldots + b_k X_k$$

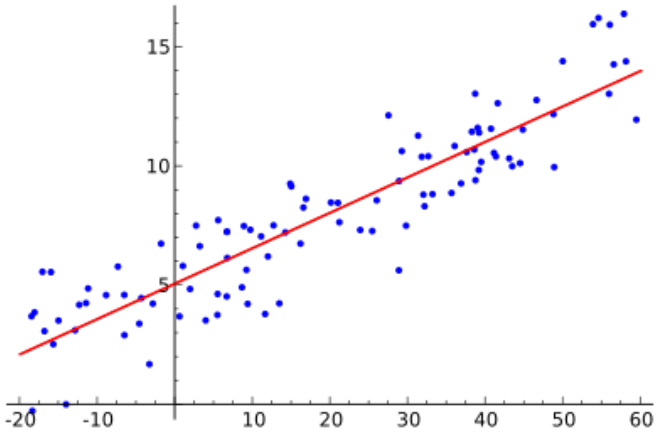


Figure 2: Simple Linear Regression

The dependent value which we want to predict is continuous and does not belong any specific set of values. Also, the independent variables are continuous and not categorical. Hence, we chose Linear Regression with multiple variable and not Logistic Regression.

For our project the variables/predictors are car manufacture brand, car body type, model, drive type and many more, as discussed in first paper. The unknown value which we are willing to predict is the predicted price of car. The formula for it will look like as –

$$Predicted\ price\ of\ car = b_0 + b_1 * car + b_2 * model + b_3 * mileage + ...$$

Where,

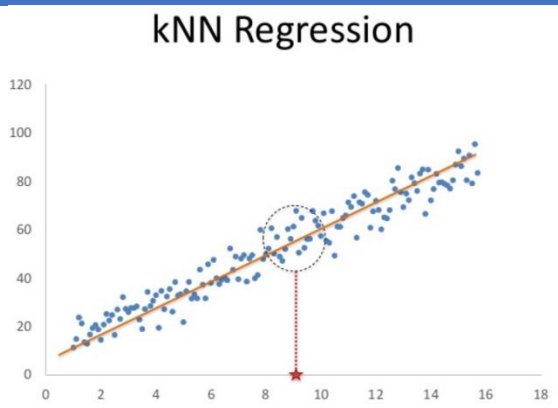
b_0 = The base value of car with no features.

b_1, b_2, b_3 = Coefficient of independent variables i.e., one-unit increase in the variable value (car, model, mileage) would increase the predicted price of car by b_1, b_2, b_3 units respectively.

The betas generated from regression analysis, help in quantifying the impact of each of the inputs. In this way we would be able to predict the sale value of car. On comparing it with actual price of car provided in the data, we can figure out if the price of that car is undervalued or not and if undervalued - we would be able to gain the business which was likely to be missed by setting the sale value of car as predicted price.

KNN

K nearest neighbors is a simple algorithm that stores all available cases and predict the numerical target based on a similarity measure (e.g., distance functions). It is non-parametric method so provide more flexible approach.

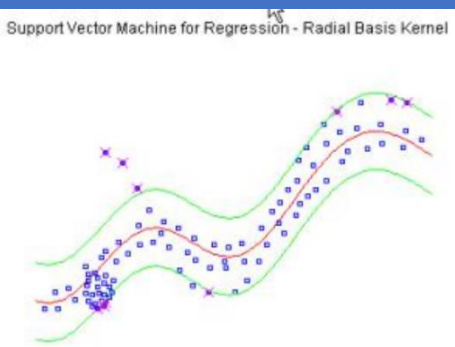


Predictions are made for a new instance by searching through the entire training set for the K most similar instances (the neighbors) and summarizing the output variable for those K instances. For regression this is the mean of output variables.

Assume a value for the number of nearest neighbors K and a prediction point *x*. KNN identifies the training observations *N* closest to the prediction point. KNN estimates *f(x)* using the average of all the responses in *N*, i.e.

$$f(x) = \frac{1}{K} \sum_{xi \in N} yi$$

SVM:

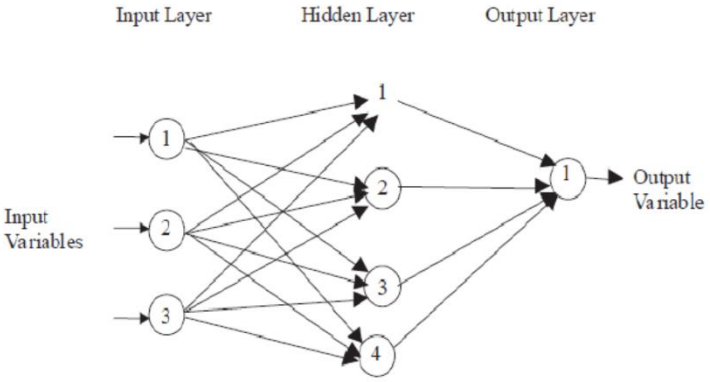


Like other regressors it also tries to fit a line. In SVR, you can deploy a non-linear kernel (here – radial basis function) and end up making non-linear regression, i.e. fitting a curve rather than a line.

The kernel functions transform the data into a higher dimensional feature space to make it possible to perform the linear separation. The capacity of the system is controlled by parameters that do not depend on the dimensionality of feature space.

In the same way as with classification approach there is motivation to seek and optimize the generalization bounds given for regression. They relied on defining the loss function that ignores errors, which are situated within the certain distance of the true value. This type of function is often called – epsilon intensive – loss function. The figure below shows an example of regression function with – epsilon intensive – band. The epsilon boundaries are given with the green lines. Blue points represent data instances. The variables measure the cost of the errors on the training points. These are zero for all points that are inside the band.

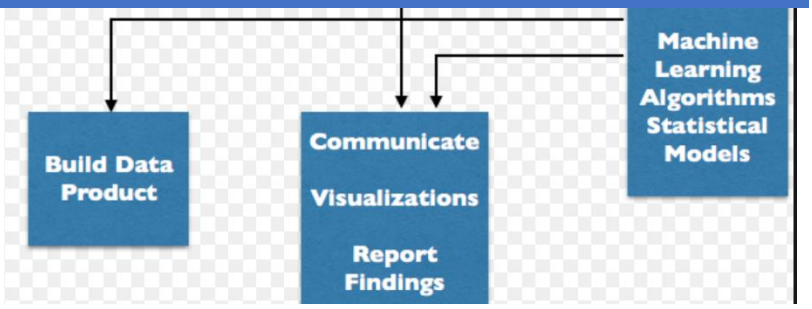
NN:



Neural nets are a means of doing machine learning, in which a computer learns to perform some task by analyzing training examples. A neural net consists of thousands or even millions of simple processing nodes that are densely interconnected. Most of today's neural nets are organized into layers of nodes, and they're "feed-forward," meaning that data moves through them in only one direction. An individual node might be connected to several nodes in the layer beneath it, from which it receives data, and several nodes in the layer above it, to which it sends data. To each of its incoming connections, a node will assign a number known as a "weight." When the network is active, the node receives a different data item — a different number — over each of its connections and multiplies it by the associated weight. It then adds the resulting products together, yielding a single number. If that number is below a threshold value, the node passes no data to the next layer. If the number exceeds the threshold value, the node "fires," which in today's neural nets generally means sending the number — the sum of the weighted inputs — along all its outgoing connections.

When a neural net is being trained, all of its weights and thresholds are initially set to random values. Training data is fed to the bottom layer — the input layer — and it passes through the succeeding layers, getting multiplied and added together in complex ways, until it finally arrives, radically transformed, at the output layer. During training, the weights and thresholds are continually adjusted until training data with the same labels consistently yield similar outputs.

MODEL EVALUATION:



It is very important to evaluate the model derived in order to know its reliability. Evaluation metrics explain the performance of a model. An important aspects of evaluation metrics is their capability to discriminate among model results. Simply, building a predictive model is not our motive. But, creating and selecting a model which gives high accuracy on out of sample data. Hence, it is crucial to check accuracy of the model prior to computing predicted value of cars. There are various metrics that helps in evaluating model accuracy.

MEAN ABSOLUTE ERROR (MAE):

MAE measures the average magnitude of the errors in a set of predictions, without considering their direction. It's the average over the test sample of the absolute differences between prediction and actual observation where all individual differences have equal weight.

ROOT MEAN SQUARED ERROR (RMSE):

RMSE is the most popular evaluation metric used in regression problems. It follows an assumption that errors are unbiased and follow a normal distribution. It is a frequently used measure of the differences between values predicted by a model and the actual values. RMSD is the square root of the average of squared errors. The effect of each error on RMSD is proportional to the size of the squared error. RMSD is always non-negative, and a value of 0 would indicate a perfect fit to the data. In general, a lower RMSD is better than a higher one.

RSQUARED (COEFFICIENT OF DETERMINATION):

R^2 is a statistic that will give some information about the goodness of fit of a model. In regression, the R^2 coefficient of determination is a statistical measure of how well the regression predictions approximate the real data points. An R^2 of 1 indicates that the regression predictions perfectly fit the data.

Above three are the metrics that can evaluate our models. On the basis of these evaluation, we would be able compare the performance of our models and find the best one.

METRICS ON TRAINING DATASET:

Below figure shows the evaluation of our models and comparison among them.

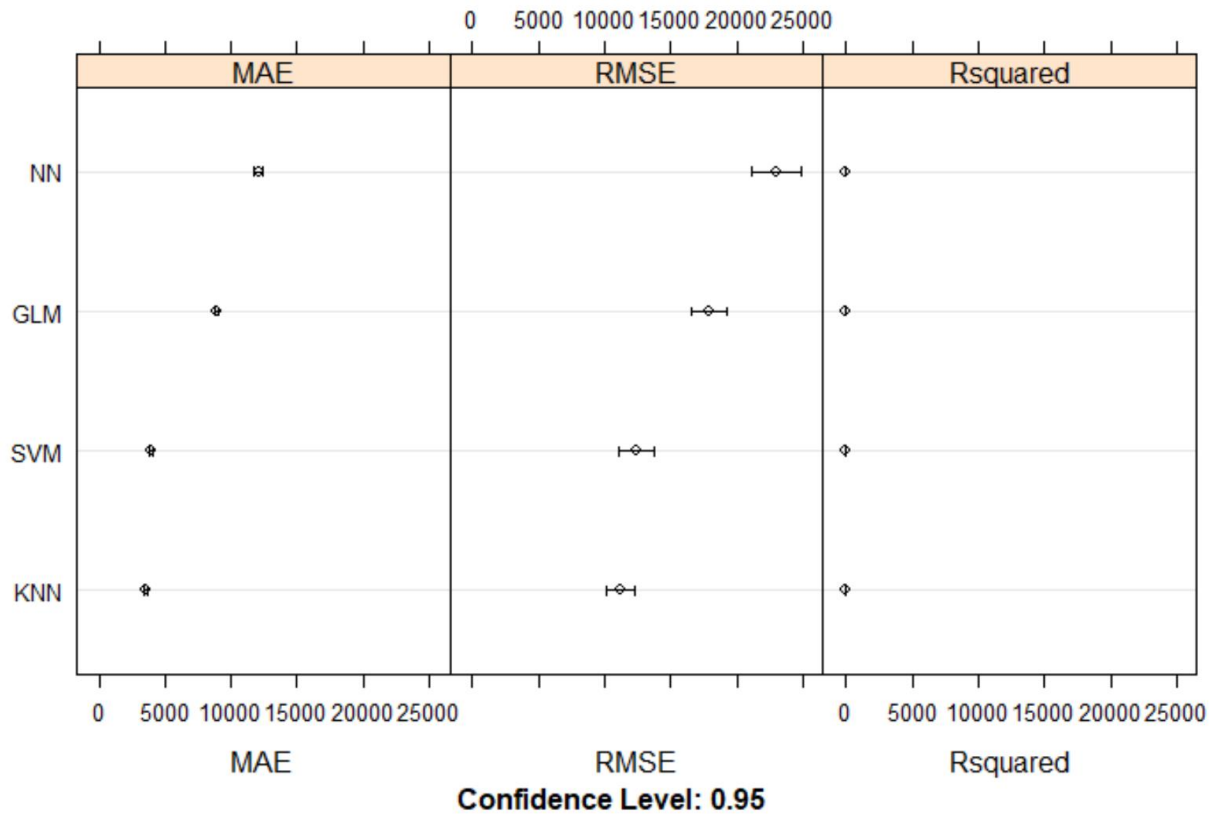


Figure 3: Metrics on training dataset

We could see that MAE and RMSE is lowest for KNN whereas its R^2 value is highest. Hence, we can say that compared to all other models, KNN is giving us better results on training datasets. Now, let's find their performances on validation dataset and check if same applies there.

METRICS ON VALIDATION DATASET:

We plotted the difference of actual price and predicted price on validation dataset for each model and it is –

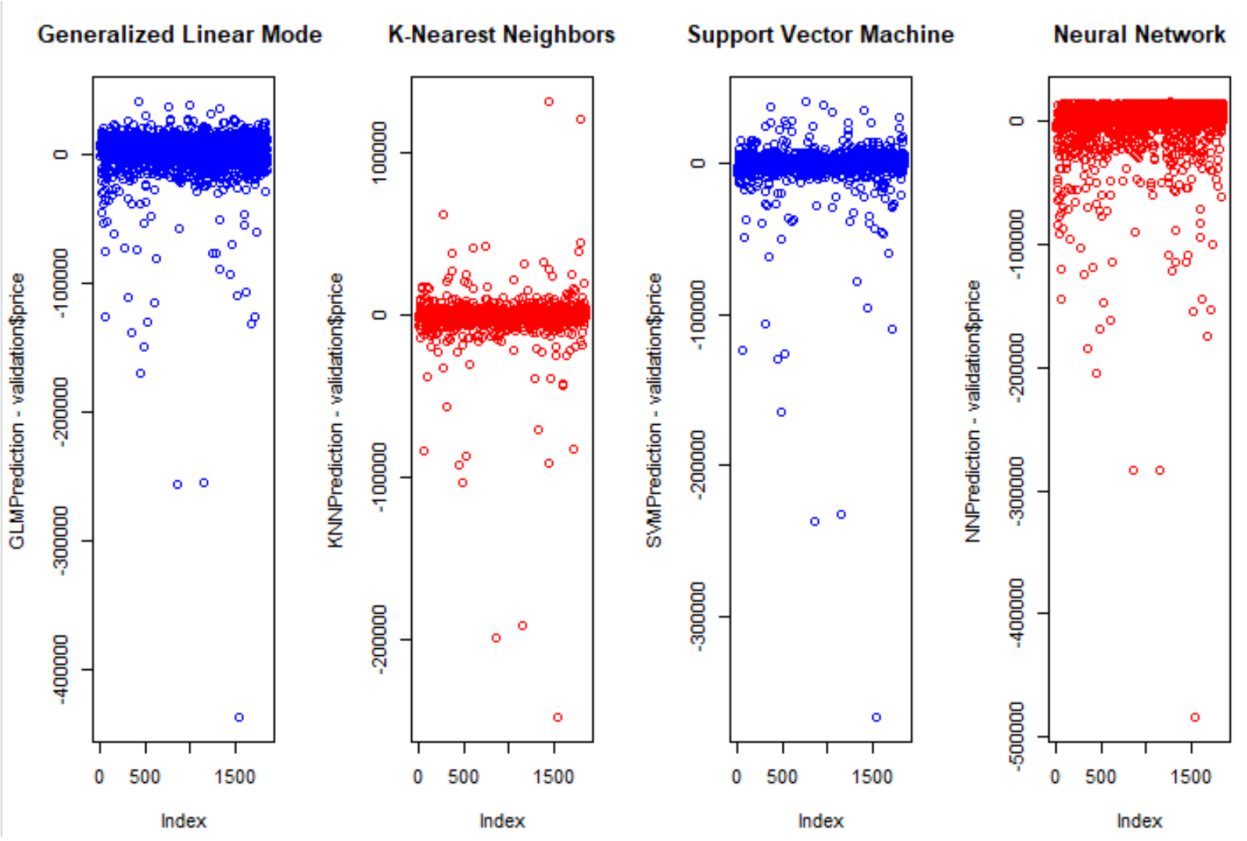


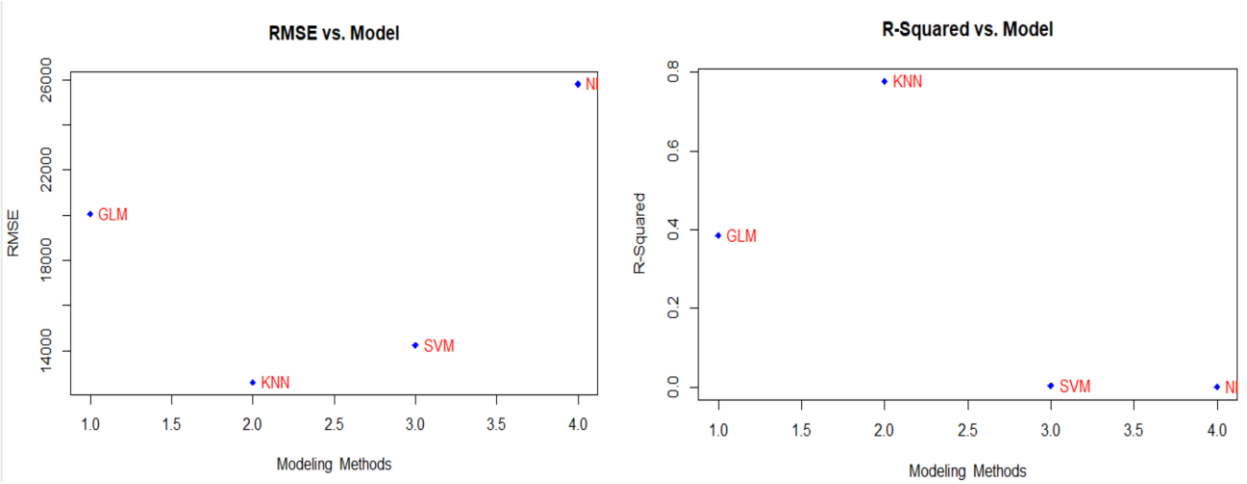
Figure 4: Difference of Actual price and Predicted price

We could notice the most distribution/variance is in model GLM and NN. Thus, SVM and KNN seems to perform better on validation datasets. Now let's compare their RMSE and R^2 values.

The RMSE and R^2 values obtained for models are –

	GLM	KNN	SVM	NN
RMSE	9552.69	5218.25	5264.69	14599.79
R^2	0.572	0.872	0.863	0.03

Table 4: Metrics on validation dataset



Thus, we can see that KNN has least RMSE value and highest R-Squared value. Hence the recommended model is KNN.

CONCLUSION:

In this paper, we discussed and studied a full process of data science life cycles for a project to predict sale value of used cars. Four different machine learning techniques were used to forecast the price of used cars. The root mean square error for k-nearest neighbor(KNN) technique has been least and also it has the highest R^2 value which conveys that among the four techniques used, KNN model gave the best performance and hence it is recommended. KNN are known to perform well with small, clean and not wide datasets.

FUTURE SCOPE:

As future work, we intend to –

- Look more in-depth at pricing strategies between different dealers, regions, seasons, etc.
- Investigate more thoroughly the factors that go into pricing. There are many other factors that can be used -

  The braking system, Exterior and interior color, Safety features etc.
- Build in a way to make the output attractive and interactive