RAJALAKSHMI
ENGINEERING COLLEGE
An AUTONOMOUS Institution
Affiliated to ANNA UNIVERSITY, Chennai

Predicting Car Prices

Submitted by
MR AKASH (211501006)
GAJENDRA RAGAVAN (211501026)

AI19341 Principles of Artificial Intelligence

Department ofArtificial Intelligence and Machine Learning

RajalakshmiEngineering College, Thandalam

BONAFIDE CERTIFICATE

This is to certify that the Mini project work titled "**Predicting Car Prices"** done by **AKASH MR (211501006), R GAJAENDRA RAGAVAN (211501026)** of **AIML** is a record of bonafide work carried out by him/her under my supervision as a part of PROJECT for the subject titled AI19341/Principles of Artificial Intelligence by Department of Artificial Intelligence and Machine Learning.

Dr.S.Baghavathi Priya

HEAD OF THE DEPARTMENT

Department of Artificial Intelligence and Machine Learning,

Rajalakshmi Engineering College,

Thandalam,

Chennai-602 105.

Mrs. Sangeetha K

FACULTY IN CHARGE

Department of Artificial Intelligence and Machine Learning,

Rajalakshmi Engineering College,

Thandalam,

Chennai- 602 105.

This project report is submitted for practical examination for AI19341/Principles of Artificial Intelligence to be held on…….at Rajalakshmi Engineering College, Thandalam.

EXTERNAL EXAMINER **INTERNAL EXAMINER**

TABLE OF CONTENTS

CHAPTER 1

ABSTRACT

Determining whether the listed price of a used car is a challenging task, due to the many factors that drive a used vehicle's price on the market. The focus of this project is developing machine learning models that can accurately predict the price of a used car based on its features, in order to make informed purchases. We implement and evaluate various learning methods on a dataset consisting of the sale prices of different makes and models across cities in the United States. Our results show that Random Forest model and K-Means clustering with linear regression yield the best results, but are compute heavy. Conventional linear regression also yielded satisfactory results, with the advantage of a significantly lower training time in comparison to the aforementioned methods.

CHAPTER 2

INTRODUCTION

The used car market is an ever-rising industry, which has almost doubled its market value in the last few years.The emergence of online portals such as CarDheko, Quikr, Carwale, Cars24, and many others has facilitated the need for both the customer and the seller to be better informed about the trends and patterns that determine the value of the used car in the market. Machine Learning algorithms can be used to predict the retail value of a car, based on a certain set of features. Different websites have different algorithms to generate the retail price of the used cars, and hence there isn't a unified algorithm for determining the price. By training statistical models for predicting the prices, one can easily get a rough estimate of the price without actually entering the details into the desired website. The main objective of this paper is to use three different prediction models to predict the retail price of a used car and compare their levels of accuracy. The data set used for the prediction models was created by Shonda Kuiper[1]. The data was collected from the 2005 Central Edition of the Kelly Blue Book and has 804 records of 2005 GM cars, whose retail prices have been calculated. The data set primarily comprises of categorical attributes along with two quantitative attributes, Every human being has got his/her business in this world .so the term business comes with certain tagged words such as sales, profit and loss. A concerns success is determined by its sales and the performance of its product in the industry is also identified by its sales. Hence in order to improve the standard of the firm /concern the strategies, the techniques the methodologies are built .in this process of development forecasting of certain key terms such as profit,loss,return of interest may pave the way for the successful venture of the concern yet when we look deep in to the details all these key terms are conveniently related to the term called sales. Sales prediction is one of the master trades of business which may open the gateways for obtaining knowledge about the existing market trends and the ways to conquer the market planning is the first step in every activity we perform and hence knowing what lies ahead in terms of sales hugely aids the concern/organization in this planning process .

CHAPTER 3

LITERATURE SURVEY

Machine learning models and bankruptcy prediction is a paperwork which talks about the improvement that takes place in academics industry with the aid of machine learning algorithms in predicting bankruptcy. The data is derived from integrated resource of Salomon center database which contains the details about the North American firms from the period between 1985 to 2013 . This paper implements the usage of algorithms such as bagging, boosting, random forest and support vector machine for predicting bankruptcy even before the event occurs and a greater span of comparative study takes place with the performance of these results with the results of logistic regression and neural networks [9].

Original Altman'sZ-score variables are used as predictive variables with addition of extra variables such as the operating margin,sales,growth measures related to assets, change in return-on-equity, change in price-to-book, and number of employees based on carton and Hofer(2006). And comparison is made between the models and these variables, the machine learning techniques and the algorithm with most accuracy is determined. Handling class imbalance in customer churn prediction by j.Burez and D. van den poel suggests the customer the various ways to handle class imbalance in churn prediction. AUC and lift are the evaluation metrics with which the sampling methods are interrogated .the modeling techniques such as weighted random forest, gradient boosting are compared with other techniques. The better evaluation metrics and the best modelling techniques are found out with the help of each techniques accuracy and from past studies . Machine learning techniques is the work that determines the worth of a particular customer with respect to his/her general pattern trait of the community that he/she hails from. The customer calling impressions can be told beforehand by making use of a classifier model and cluster analysis for detail selection. The attributes such as accuracy and computational performance are taken in to consideration for comparison of various machine learning techniques [1].

Customer churn prediction using improved balanced random forests is the paper that explains the real time working model that had been used in china. Improved balanced random forest is the hybrid version of balanced random forest and weighted random forest, two interval variables had been introduced such as e and f where e is the middle point and f is the length of interval .Random distribution of these classes are maintained with the help of these variables.Hence it produces more accurate results than its other counterparts [2].

A sampling based sentiment mining approach for e-commerce applications paper puts the limelight on how the customers are being influenced by the Online reviews which is a part of marketing strategy of the e-commerce platforms. Hence this issue is attempted with the help of mining techniques .The two sampling methods are also used for classification of imbalanced data. A modified support vector machine based ensemble algorithm is the methodology used by the researches to identify the performance prediction [10].

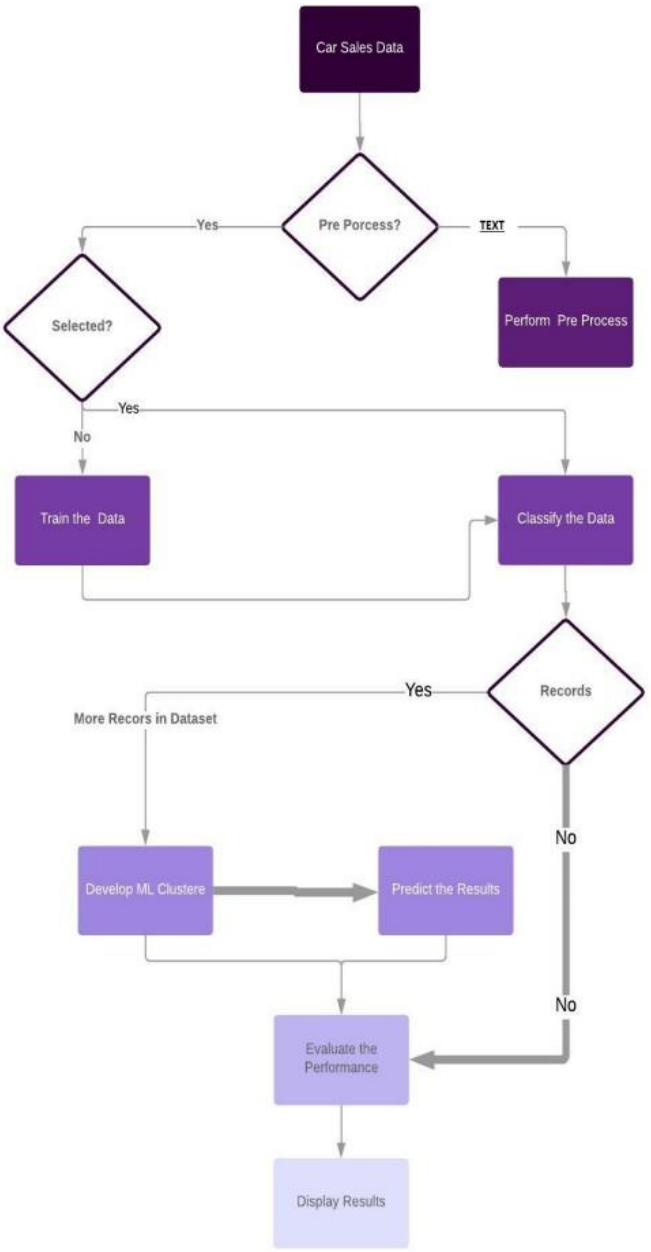
On the differential benchmarking of promotional efficiency with machine learning modelling(II): Practical applications presents two different databases of different categories such as non-seasonal and heavy seasonal and models are analyzed here .The detailed performance of four famous machine learning techniques that has huge complexity is been dissected in this work. Certain features of various machine learning algorithms do not perform well
because of these databases. In order to gain more accurate dissection results and feature extraction there is a need to implement certain correct procedures that may influence the specificity of the behavior of certain categories and product ranges [11].

Krishnamurthy analyses what makes an Online review with the help of a predictive model .This model follows the methodology of extracting linguistic category features such as adjective feature, state verb feature and action verb features it also takes in to account the readability related features for prediction. Hence the hybrid set of features that are obtained after the analysis on two real-life review datasets gives the researches the best accuracy rate of all time [8]. Using machine learning algorithms for housing price prediction: The case of Fairfax County, Virginia housing data paper puts the focus on how this work can contribute to the real estate industry that may cause an adverse effect on the US housing market.an eight step methodologies are used for the dataset of 5359 townhouses in Fairfax County, Virginia. The dataset has been segregated in to training, validation and testing set then training parameters such as C4.5, RIPPER, Naïve Bayesian, and Adaboost are set and the model is trained and evaluated using training set and validation set respectively and this process is iterated until it gains an optimal error in training, validation and testing.Finaly these results are compared to gain the optimal accuracy results [7]. Explaining machine learning models in sales prediction is a generic manuscript that discusses about the recent trends of predictive models, real time scenarios to gain a deep insight about buyers and seller's interaction and the forecasting of sales [5].

Early churn prediction with personalized targeting in mobile social games is a manuscript that explains Customer churn .churn is defined by the act of a customer leaving a product for good. This churn are reduced to a greater extent by following the procedure of mapping the feature with the interest of the customer and pushing the notifications in order to drag back the customer in to the game .this manuscript implements the methodologies such as logistic regression for the simple object linear model ,decision trees for extracting redundancy from features random forest to be used in various situations .Naive Bayes for generating the models and gradient boosting for its popularity [4].

CHAPTER 4

MODEL ARCHITECTURE



CHAPTER 5

IMPLEMENTATION

Gradient Boosting

Gradient boosting is one of the most powerful techniques for building predictive models. Gradient Boosting Algorithm is generally used when we want to decrease the Bias error. Gradient Boosting Algorithm can be used in regression as well as classification problems. Gradient boosting involves three elements: A loss function to be optimized, A weak learner to make predictions, an additive model to add weak learners to minimize the loss function. Before applying this model, we have used 70% of the data for training the model and 30% to test the model and evaluate its accuracy. Initially, we got a 77% accuracy score by applying GradientBoostingRegressor to the data. After optimizing the parameters of the model by the GrindSearchCV method, the model has produced an accuracy score of 92%.

Methodology

Data is collected from Kaggle. The following attributes were captured for each car: Name, Location, Year, Fuel Type, Transmission, Owner Type, Mileage, Engine, Power, Seats, and Price expressed in Indian rupees. After the data was collected and stored, the data preprocessing step was applied. The attributes with unexpected values are processed accordingly i.e. In our case we have replaced them with the most repeated value of the attribute. The cars without a price are discarded in prior. To avoid conflict in mileage among different cars, all the mileages of cars are been scaled to a kmpl because most of the records are in km. To convert categorical data values into numeric attributes like (Company, Name, Location, Fuel, Transmission, Owner) we have used a one- hot encoding approach.

Future Enhancements

A car price prediction has been a high-interest research area, as it requires noticeable effort and knowledge of the field expert. A considerable number of distinct attributes are examined for reliable and accurate predictions. The major step in the prediction process is the collection and pre-processing of the data. In this project, data was normalized and cleaned to avoid unnecessary noise for machine learning algorithms. Applying a single machine algorithm to the data set accuracy was less than 70%. Therefore, the ensemble of multiple machine learning algorithms has been proposed and this combination of ML methods gains an accuracy of 93%. This is a significant improvement compared to the single machine learning method approach. However, the drawback of the proposed system is that it consumes much more computational resources than a single machine learning algorithm. Although this system has achieved astonishing performance in the car price prediction problem, it can also be implemented using an advanced machine learning model and with Deep learning techniques to improve its efficiency and accuracy. Moreover, as innovation has been increased in automobiles and we can observe Electric vehicles have gained public attention and are preferred by most than a normalcar.

CHAPTER 6
RESULTS AND DISCUSSIONS

Some people preferred a good corridor, some are high or low. price with all of their demanded features, some are only weak for. notorious brands of the auto only. To elect the perfect auto is still a delicate task though some parameters like color, comfort, seating capacity, etc. areknown [3]. That's why we tried to compare some algorithms for prognosticating auto buying purposes that which one gives better delicacy. Execution of the Naive Bayes Classification method is proposed by Fitrina . Credulous Bayes is known as a simple probabilistic classifier. They connected this strategy for predicting buy. They utilized a dataset on 20 car buying information and got 75% precision. Srivastava connected the powerful learning strategy, Bolster Vector Machines(SVM) to different sorts of information like Diabetes Information, Heart Information, Satellite Data, and Carry Information. Those datasets have multi classes. They have too demonstrated the examination of the comparative consequences of the utilization of jumper's bit capacities on their paper. Osisanwo proposed a Supervised machine learning method. They compared seven different Administered learning calculations and portrayed those. They moreover found the foremost viable classification algorithm.

CHAPTER 7
CONCLUSION

The increased prices of new cars and the financial incapability of the customers to buy them, Used Car sales are on a global increase. Therefore, there is an urgent need for a Used Car Price Prediction system which effectively determines the worthiness of the car using a variety of features. The proposed system will help to determine the accurate price of used car price prediction. This paper compares 3 different algorithms for machine learning : Linear Regression, Lasso Regression and Ridge Regression.

CHAPTER 8
REFERENCES

[1] Sameerchand Pudaruth, “Predicting the Price of Used Cars using MachineLearning Techniques”;(IJICT 2014)

[2] Enis gegic, Becir Isakovic, Dino Keco, Zerina Masetic, Jasmin Kevric, ”CarPrice Prediction Using Machine Learning”; (TEM Journal 2019)

[3] Ning sun, Hongxi Bai, Yuxia Geng, Huizhu Shi, “Price Evaluation ModelIn Second Hand Car System Based On BP Neural Network Theory”; (Hohai University Changzhou, China)

[4] Nitis Monburinon, Prajak Chertchom, Thongchai Kaewkiriya, Suwat Rungpheung, Sabir Buya, Pitchayakit Boonpou, “Prediction of Prices for UsedCar by using Regression Models” (ICBIR 2018)

[5] Doan Van Thai, Luong Ngoc Son, Pham Vu Tien, Nguyen Nhat Anh, Nguyen Thi Ngoc Anh, “Prediction car prices using qualify qualitative data andknowledge-based system” (Hanoi National University)

APPENDIX 1-CODING

Importing the dependencies

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
# loading the data from csv file to pandas dataframe
car_dataset = pd.read_csv('/content/car data.csv')
```

```
# inspecting the first 5 rows of the dataframe
```

	Car_Name	Year	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission	Owner
0	ritz	2014	3.35	5.59	27000	Petrol	Dealer	Manual	0
1	sx4	2013	4.75	9.54	43000	Diesel	Dealer	Manual	0
2	ciaz	2017	7.25	9.85	6900	Petrol	Dealer	Manual	0
3	wagon r	2011	2.85	4.15	5200	Petrol	Dealer	Manual	0
4	swift	2014	4.60	6.87	42450	Diesel	Dealer	Manual	0

Number of rows

```
# checking the number of rows and columns
car_dataset.shape
```

```
(301, 9)
```

```
# getting some information about the dataset
car_dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 301 entries, 0 to 300
Data columns (total 9 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   Car_Name       301 non-null    object 
 1   Year           301 non-null    int64  
 2   Selling_Price  301 non-null    float64
 3   Present_Price  301 non-null    float64
 4   Kms_Driven     301 non-null    int64  
 5   Fuel_Type      301 non-null    object 
 6   Seller_Type    301 non-null    object 
 7   Transmission   301 non-null    object 
 8   Owner          301 non-null    int64  
dtypes: float64(2), int64(3), object(4)
memory usage: 21.3+ KB
```

Distribution of data

```
# checking the distribution of categorical data
print(car_dataset.Fuel_Type.value_counts())
print(car_dataset.Seller_Type.value_counts())
print(car_dataset.Transmission.value_counts())
```

```
Petrol    239
Diesel     60
CNG         2
Name: Fuel_Type, dtype: int64
```

```
# encoding "Fuel_Type" Column
car_dataset.replace({'Fuel_Type':{'Petrol':0,'Diesel':1,'CNG':2}},inplace=True)

# encoding "Seller_Type" Column
car_dataset.replace({'Seller_Type':{'Dealer':0,'Individual':1}},inplace=True)

# encoding "Transmission" Column
car_dataset.replace({'Transmission':{'Manual':0,'Automatic':1}},inplace=True)
```

```
car_dataset.head()
```

	Car_Name	Year	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission	Owner
0	ritz	2014	3.35	5.59	27000	0	0	0	0
1	sx4	2013	4.75	9.54	43000	1	0	0	0
2	ciaz	2017	7.25	9.85	6900	0	0	0	0
3	wagon r	2011	2.85	4.15	5200	0	0	0	0
4	swift	2014	4.60	6.87	42450	1	0	0	0

Splitting the data and target

```
X = car_dataset.drop(['Car_Name','Selling_Price'],axis=1)
Y = car_dataset['Selling_Price']
```

```
print(X)
```

```
     Year  Present_Price  Kms_Driven  ...  Seller_Type  Transmission  Owner
0    2014           5.59       27000  ...            0             0      0
1    2013           9.54       43000  ...            0             0      0
2    2017           9.85        6900  ...            0             0      0
3    2011           4.15        5200  ...            0             0      0
4    2014           6.87       42450  ...            0             0      0
..    ...            ...         ...  ...          ...           ...    ...
296  2016          11.60       33988  ...            0             0      0
297  2015           5.90       60000  ...            0             0      0
298  2009          11.00       87934  ...            0             0      0
299  2017          12.50        9000  ...            0             0      0
300  2016           5.90        5464  ...            0             0      0

[301 rows x 7 columns]
```

```
print(Y)
```

```
0       3.35
1       4.75
2       7.25
3       2.85
4       4.60
       ...
296     9.50
297     4.00
298     3.35
299    11.50
300     5.30
Name: Selling_Price, Length: 301, dtype: float64
```

Splitting the training and test data

Splitting Training and Test data

```
[ ] X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size = 0.1, random_state=2)
```

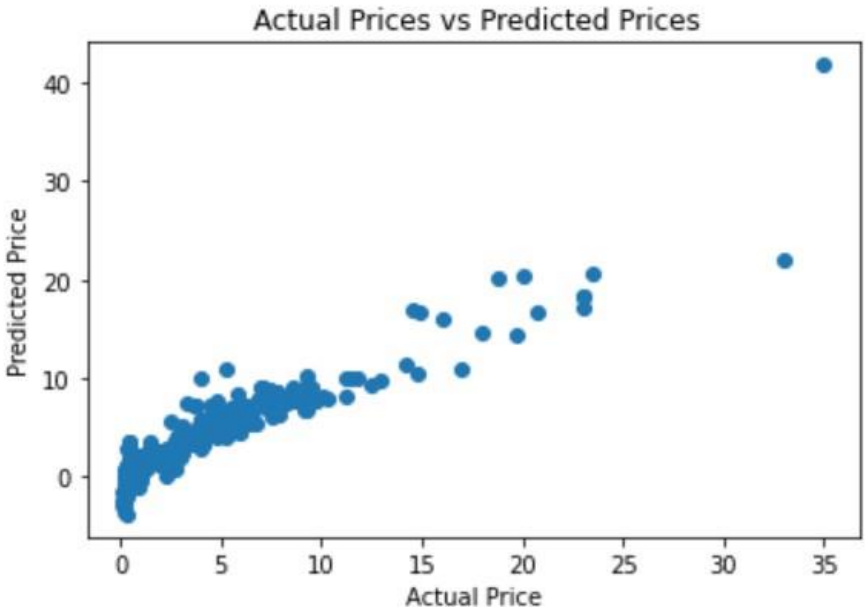
```
# prediction on Training data
training_data_prediction = lin_reg_model.predict(X_train)
```

```
# R squared Error
error_score = metrics.r2_score(Y_train, training_data_prediction)
print("R squared Error : ", error_score)
```

```
R squared Error :  0.8799451660493711
```

Visualize the prices

```
plt.scatter(Y_train, training_data_prediction)
plt.xlabel("Actual Price")
plt.ylabel("Predicted Price")
plt.title(" Actual Prices vs Predicted Prices")
plt.show()
```



```
# prediction on Training data
test_data_prediction = lin_reg_model.predict(X_test)
```

```
# R squared Error
error_score = metrics.r2_score(Y_test, test_data_prediction)
print("R squared Error : ", error_score)
```

```
R squared Error :  0.8365766715027051
```

```
plt.scatter(Y_test, test_data_prediction)
plt.xlabel("Actual Price")
plt.ylabel("Predicted Price")
plt.title(" Actual Prices vs Predicted Prices")
plt.show()
```

Lasso regression

```
[ ]  # loading the linear regression model
     lass_reg_model = Lasso()
```

```
[ ]  lass_reg_model.fit(X_train,Y_train)
```

```
Lasso(alpha=1.0, copy_X=True, fit_intercept=True, max_iter=1000,
      normalize=False, positive=False, precompute=False, random_state=None,
      selection='cyclic', tol=0.0001, warm_start=False)
```

Model Evaluation

```
[ ]  # prediction on Training data
     training_data_prediction = lass_reg_model.predict(X_train)
```

```
[ ]  # R squared Error
     error_score = metrics.r2_score(Y_train, training_data_prediction)
     print("R squared Error : ", error_score)
```

```
R squared Error :  0.8427856123435794
```

```
plt.scatter(Y_train, training_data_prediction)
plt.xlabel("Actual Price")
plt.ylabel("Predicted Price")
plt.title(" Actual Prices vs Predicted Prices")
plt.show()
```

```
# prediction on Training data
test_data_prediction = lass_reg_model.predict(X_test)
```

```
# R squared Error
error_score = metrics.r2_score(Y_test, test_data_prediction)
print("R squared Error : ", error_score)
```

```
R squared Error :  0.8709167941173195
```

```
plt.scatter(Y_test, test_data_prediction)
plt.xlabel("Actual Price")
plt.ylabel("Predicted Price")
plt.title(" Actual Prices vs Predicted Prices")
plt.show()
```

APPENDIX II- OUTPUT SCREENSHOTS