Michigan Used Cars Prices Prediction Analysis

Data Science

Coursework 2

Sebastian Torrejon Alonzo

Table of Contents

Introduction

The market for used cars is an important and dynamic part of the global automotive industry. Consumers are increasingly turning to pre-owned vehicles as a cost-effective alternative to buying new cars. However, understanding the factors that influence the pricing of used cars can be challenging for both buyers and sellers.
In this report, we present a comprehensive analysis of the factors that impact the prices of used cars. Using a large dataset from here: ('https://www.kaggle.com/datasets/austinreese/craigslist-carstrucks-data') of used car postings, we examine the impact of variables such as year, odometer, manufacturer, fuel, and condition on the final sale price as shown here. Our findings shed light on the complexities of the used car market and provide valuable insights for both consumers and industry professionals.

Data Type	Features	Description
Numerical Data	Price	This is the label of the model which will be calculated.
Numerical Data	Year	The year of the car is between 1923 and 2021.
Categorical Data	Manufacturer	This is the make of the car, like 'ford', 'tesla', etc.
Categorical Data	Condition	This is the condition of the car, like 'good', 'excellent', etc.
Categorical Data	Fuel	The type of fuel that the car uses, like 'gas', 'electric', etc.
Numerical Data	Odometer	The mileage of the car which is from 0, to hundreds of thousands.

Pre-Processing

Step 1: The dataset was downloaded from Kaggle in CSV format.

Step 2: The dataset was opened in Microsoft Excel. (~500,000 records in the dataset)

Step 3: Deleted all records that are not from Michigan. (~17,000 records in the dataset)

Step 4: Removed irrelevant columns. (url, region, region_url, model, cylinders, transmission, title_status, VIN, drive, size, type, paint_color, image_url, description, county, state, lat, long, posting_date)

Step 5: Removed rows with empty values. (~10,000 records in the dataset)

Step 6: Removed duplicated rows with the same car posted on different websites and regions in the state. (~6,000 records in the dataset)

Step 7: Checked for outliers and extreme values, and removed them. (~5,000 records in the dataset)

vehicles_michigan

id	price	year	manufacturer	condition	fuel	odometer
1	22568	2014	ram	excellent	gas	133275
2	4380	2006	jeep	excellent	gas	160252
3	7800	2012	chevrolet	excellent	gas	130000
4	38990	2020	tesla	good	electric	9665
5	32590	2015	mercedes-benz	good	gas	34811
6	3906	2011	ford	excellent	gas	230739
7	9360	2011	nissan	excellent	gas	125475
8	18990	2020	chevrolet	good	gas	6395
9	30590	2020	ford	good	gas	10740
10	21995	2016	toyota	excellent	hybrid	44438
11	66524	2019	ford	excellent	gas	41468
12	11667	2016	gmc	excellent	gas	104056
13	37990	2019	jeep	good	gas	21020
14	39590	2019	chevrolet	good	gas	21014

Hypothesis

Used Cars Prices: ***Analyzing the data on used cars in Michigan will help the automobile industry to identify the factors affecting the resale value of the cars.***

The three problems that will be addressed in this hypothesis are:

1. What factors affect the resale value of used cars in Michigan?
2. How can automobile manufacturers design cars that retain their resale value?
3. How can used car dealerships price their cars appropriately based on their resale value?

Proposed Solution

The proposed solution for this hypothesis is to use multiple linear regression analysis to identify the factors affecting the resale value of used cars in Michigan. Linear regression is a supervised learning technique that can be used to predict the value of a dependent variable based on one or more independent variables. It is a powerful tool that can help us understand the complex relationships between variables.

The target variable in a dataset of used car prices is continuous, so a regression model is appropriate. Linear Regression was chosen as it is simple and can provide good results. Pre-processing of input features is necessary, including removing missing or invalid values and normalizing continuous values, to ensure effective use by the model. Using a complex model like a neural network may not be necessary and can lead to overfitting. The proposed solution of using Linear Regression with pre-processed input features is appropriate and can provide good results while avoiding overfitting, and can lead to deeper insights into factors affecting the price of used cars in Michigan.

Implementation

To implement this solution, we will use Python and the Pandas library to prepare the dataset. We load the dataset into the data frame, and convert the predictors to numerical values.

```
# import libraries
import pandas as pd
import numpy as np

# load dataframe
df = pd.read_csv('./vehicles_michigan.csv')

# year column into array
predictor1 = np.array(pd.to_numeric(df['year']), ndmin=2).T
#manufacturer column to dummies values
predictor2 = pd.get_dummies(df['manufacturer'], drop_first=True)
predictor3 = pd.get_dummies(df['condition'], drop_first=True)
predictor4 = pd.get_dummies(df['fuel'], drop_first=True)
predictor5 = np.array(pd.to_numeric(df['odometer']), ndmin=2).T
y = np.array(df['price'], ndmin=2).T
```

Once the dataset is ready, we will use multiple linear regression and libraries to implement the Least Square algorithm which will give us the slope or the line with the least mistakes possible. In simple words, we are training the data and making the price range as small as possible. The following code converts the predictors to a single matrix which will be used to calculate the least square equation.

```
X = np.column_stack([np.ones(predictor1.shape), predictor1,
predictor2, predictor3, predictor4, predictor5]) # predictors into
one matrix
XTX = np.dot(X.T, X) # Step 1
XTX_inv = np.linalg.inv(XTX) # Step 2
XTX_invXT = np.dot(XTX_inv, X.T) # Step 3
w = np.dot(XTX_invXT, y) # parameters of least squares
```

Finally, we will get input values from the user to calculate the predicted price of a car by multiplying the user input by the least square parameters we calculated earlier.

```
year = int(input("Enter the year of the car: ")) # Try '2018'
manufacturer = input("Enter the manufacturer of the car: ") # Try 'tesla'
condition = input("Enter the condition of the car: ") # Try 'good'
fuel = input("Enter the fuel of the car: ") # Try 'electric'
odometer = int(input("Enter the odometer of the car: ")) # Try '22000'

x1 = np.array([year]) # input year
manufacturers = np.sort(df['manufacturer'].unique()) # get manufacturers list
x2_arr = np.zeros(len(manufacturers)) # populate dummie values
x2_arr[np.where(manufacturers == manufacturer)] = 1 # add 1 to the input manufacturer
x2 = x2_arr[1:]
conditions = np.sort(df['condition'].unique())
x3_arr = np.zeros(len(conditions))
x3_arr[np.where(conditions == condition)] = 1
x3 = x3_arr[1:]
fuels = np.sort(df['fuel'].unique())
x4_arr = np.zeros(len(fuels))
x4_arr[np.where(fuels == fuel)] = 1
x4 = x4_arr[1:]
x5 = np.array([odometer])

x = np.concatenate(([1], x1, x2, x3, x4, x5)) # get input values into one matrix
price = np.dot(x, w) # calculate regression

print("The car price is: $", end = '')
print(f"{price[0]:,.2f}") # Should get over ~$40,000
```

Reflection

The proposed solution is a practical and effective approach to identifying the factors that influence the resale value of used cars in Michigan. The use of multiple linear regression analysis provides a powerful tool to understand the complex relationships between variables and helps automobile manufacturers to design cars that retain their resale value and used car dealerships to price their cars appropriately based on their resale value.

Overall, this report provides valuable insights into the complexities of the used car market and presents a practical solution that can benefit both consumers and industry professionals. The report demonstrates the importance of data analysis and pre-processing in identifying the factors that affect the pricing of used cars and highlights the potential of multiple linear regression analysis as a powerful tool for predicting the value of a dependent variable based on one or more independent variables.

```
year = int(input("Enter the year of the car: ")) # Try '2018'
manufacturer = input("Enter the manufacturer of the car: ") # Try 'tesla'
condition = input("Enter the condition of the car: ") # Try 'good'
fuel = input("Enter the fuel of the car: ") # Try 'electric'
odometer = int(input("Enter the odometer of the car: ")) # Try '22000'

x1 = np.array([year]) # input year
manufacturers = np.sort(df['manufacturer'].unique()) # get manufacturers list
x2_arr = np.zeros(len(manufacturers)) # populate dummie values
x2_arr[np.where(manufacturers == manufacturer)] = 1 # add 1 to the input manufacturer
x2 = x2_arr[1:]
conditions = np.sort(df['condition'].unique())
x3_arr = np.zeros(len(conditions))
x3_arr[np.where(conditions == condition)] = 1
x3 = x3_arr[1:]
fuels = np.sort(df['fuel'].unique())
x4_arr = np.zeros(len(fuels))
x4_arr[np.where(fuels == fuel)] = 1
x4 = x4_arr[1:]
x5 = np.array([odometer])

x = np.concatenate(([1], x1, x2, x3, x4, x5)) # get input values into one matrix
price = np.dot(x, w) # calculate regression

print("The car price is: $", end = '')
print(f"{price[0]:,.2f}") # Should get over ~$40,000
```

```
Enter the year of the car: 2018
Enter the manufacturer of the car: tesla
Enter the condition of the car: good
Enter the fuel of the car: electric
Enter the odometer of the car: 22000
The car price is: $42,466.01
```