EGR 598 Machine Learning and AI Course Project

USED CAR PRICE PREDICTION

Shun-Yen Wang
Neel Macwan
Shuoh Herng Lee
Srikar Anudeep Remani
Sundeep Yalamanchili

Arizona State University

Problem: Forecasting the price of used cars based on different attributes.

Arizona State University

Image Reference: https://www.google.com/search?q=used+car+prediction+ML&rlz=1C1CHBF_enIN978IN978&sxsrf=ALi

Abstract: This project's primary goal is to forecast used car prices based on each model's unique attributes. It enables you to decide whether a used car is worth the asking price provided by different online used car sites.

Arizona State University

Introduction: The goal of this project is to create a machine learning model that can precisely forecast a used cars' pricing based on its features so that consumers can make an informed decision.

Arizona State University

It can be challenging to determine whether a used car is worth the asking price when perusing online classifieds. The mileage, fuel type, number of owners, year, and other elements can all have an impact on a car's genuine value. A seller's difficulty is how to appropriately price a secondhand car.

Objective: To demonstrate and develop a model that can forecast the accurate price of used cars using a price prediction system that predicts the price on different parameters by using Machine Learning Algorithms and Existing Data.

Arizona State University

Data Description

	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
...	...	...	...	...	...	...	...	...	...
296	city	2016	9.50	11.60	33988	Diesel	Dealer	Manual	0
297	brio	2015	4.00	5.90	60000	Petrol	Dealer	Manual	0
298	city	2009	3.35	11.00	87934	Petrol	Dealer	Manual	0
299	city	2017	11.50	12.50	9000	Diesel	Dealer	Manual	0
300	brio	2016	5.30	5.90	5464	Petrol	Dealer	Manual	0

301 rows × 9 columns

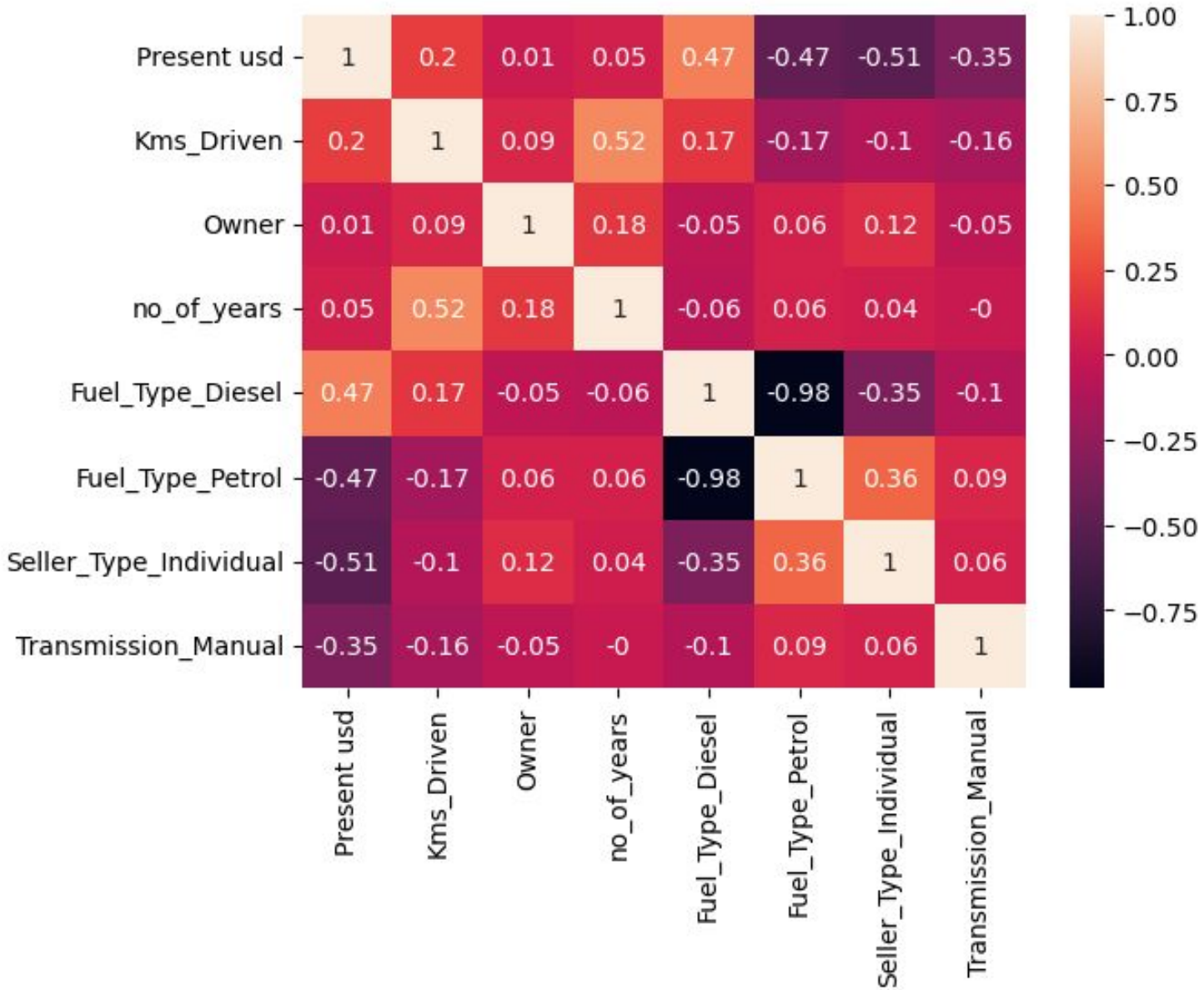
Data Description

	Year	Selling_Price	Present_Price	Kms_Driven	Owner
count	301.000000	301.000000	301.000000	301.000000	301.000000
mean	2013.627907	4.661296	7.628472	36947.205980	0.043189
std	2.891554	5.082812	8.644115	38886.883882	0.247915
min	2003.000000	0.100000	0.320000	500.000000	0.000000
25%	2012.000000	0.900000	1.200000	15000.000000	0.000000
50%	2014.000000	3.600000	6.400000	32000.000000	0.000000
75%	2016.000000	6.000000	9.900000	48767.000000	0.000000
max	2018.000000	35.000000	92.600000	500000.000000	3.000000

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 301 entries, 0 to 300
Data columns (total 10 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   Car_Name        301 non-null    object
 1   Selling_Price   301 non-null    float64
 2   Present_Price   301 non-null    float64
 3   Kms_Driven      301 non-null    int64
 4   Fuel_Type       301 non-null    object
 5   Seller_Type     301 non-null    object
 6   Transmission    301 non-null    object
 7   Owner           301 non-null    int64
 8   current_year    301 non-null    int64
 9   no_of_years     301 non-null    int64
dtypes: float64(2), int64(4), object(4)
memory usage: 23.6+ KB
```

Arizona State University

Data Description



Arizona State University

Data Preprocessing

```
df['current_year']=2021
df.head()
```

	Car_Name	Year	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission	Owner	current_year
0	ritz	2014	3.35	5.59	27000	Petrol	Dealer	Manual	0	2021
1	sx4	2013	4.75	9.54	43000	Diesel	Dealer	Manual	0	2021
2	ciaz	2017	7.25	9.85	6900	Petrol	Dealer	Manual	0	2021
3	wagon r	2011	2.85	4.15	5200	Petrol	Dealer	Manual	0	2021
4	swift	2014	4.60	6.87	42450	Diesel	Dealer	Manual	0	2021

```
df['no_of_years']=df['current_year']-df['Year']
df.head()
```

	Car_Name	Year	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission	Owner	current_year	no_of_years
0	ritz	2014	3.35	5.59	27000	Petrol	Dealer	Manual	0	2021	7
1	sx4	2013	4.75	9.54	43000	Diesel	Dealer	Manual	0	2021	8
2	ciaz	2017	7.25	9.85	6900	Petrol	Dealer	Manual	0	2021	4
3	wagon r	2011	2.85	4.15	5200	Petrol	Dealer	Manual	0	2021	10
4	swift	2014	4.60	6.87	42450	Diesel	Dealer	Manual	0	2021	7

Data Preprocessing

	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission	Owner	current_year	no_of_years
0	3.35	5.59	27000	Petrol	Dealer	Manual	0	2021	7
1	4.75	9.54	43000	Diesel	Dealer	Manual	0	2021	8
2	7.25	9.85	6900	Petrol	Dealer	Manual	0	2021	4
3	2.85	4.15	5200	Petrol	Dealer	Manual	0	2021	10
4	4.60	6.87	42450	Diesel	Dealer	Manual	0	2021	7

```
#one hot encoding
final_df = pd.get_dummies(df,drop_first=True)
```

```
final_df.head()
```

	Selling_Price	Present_Price	Kms_Driven	Owner	current_year	no_of_years	Fuel_Type_Diesel	Fuel_Type_Petrol	Seller_Type_Individual	Transmission_Manual
0	3.35	5.59	27000	0	2021	7	0	1	0	1
1	4.75	9.54	43000	0	2021	8	1	0	0	1
2	7.25	9.85	6900	0	2021	4	0	1	0	1
3	2.85	4.15	5200	0	2021	10	0	1	0	1
4	4.60	6.87	42450	0	2021	7	1	0	0	1

Data Preprocessing

	Present usd	Kms_Driven	Owner	no_of_years	Fuel_Type_Diesel	Fuel_Type_Petrol	Seller_Type_Individual	Transmission_Manual
146	948.192771	15000	0	7	0	1	1	1
260	16385.542169	29223	0	5	0	1	0	1
37	2746.987952	127000	0	18	0	1	1	1
273	9036.144578	61203	0	11	0	1	0	1
164	650.602410	14000	0	5	0	1	1	1

Data split:

Trainning set: 75%

Testing set: 25%

Arizona State University

Model Selection: In this Project we use supervised machine learning methods to forecast used automobile prices. Forecasts are supported by historical information gathered from daily publications. Predictions were made using linear regression analysis, SVR, naive bayes, and decision trees.

Arizona State University

Model Selection:

- Linear Regression
- Decision Tree
- Random Forest
- SVR
- MLP

Arizona State University

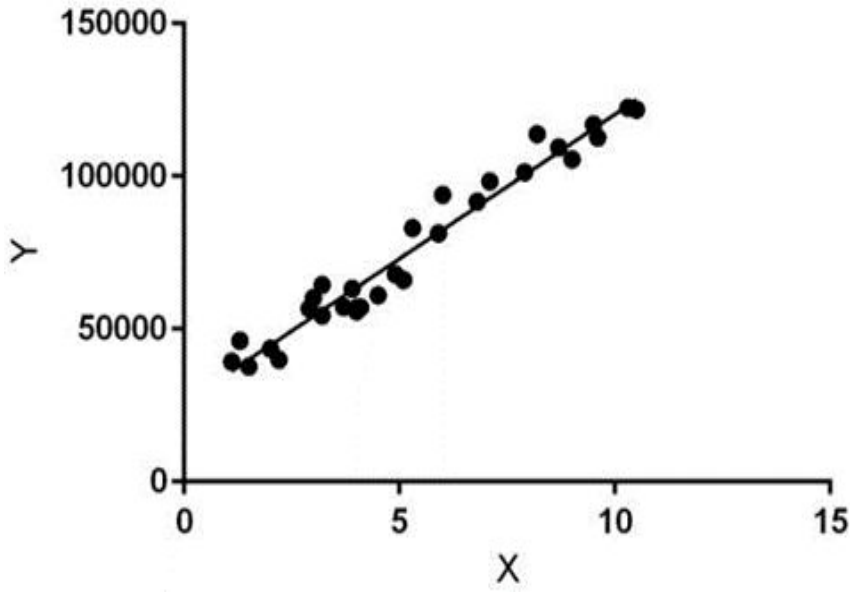
Disadvantages of Existing System:

Time Consuming

More difficult than other Algorithms

Random Forest Algorithm might be able to address the overfitting issue.

Existing System: Linear Regression



Hypothesis function for Linear Regression

$$y = \theta_1 + \theta_2.x$$

Decision Tree:

1. A decision node and a leaf node are the two nodes in a decision tree. While leaf nodes are the outcomes of decisions and have no more branches, decision nodes are used to create decisions and have numerous branches.

2. Based on the features of a specific data collection, a choice or test is made.

3. A visual representation of all potential answers to a choice or problem based on the conditions provided.

4. Because it grows from a root node to additional branches to create a tree-like structure, it is known as a decision tree. The CART algorithm, which stands for Classification and Regression Tree Algorithm, is used to construct the tree.

Decision Tree

- A decision tree is a hierarchical data structure implementing the divide-and-conquer strategy. It is an efficient nonparametric method, which can be used for both classification and regression.

$$f_m(\boldsymbol{x}) : x_j > w_{m0}$$

$$\hat{P}(C_i|\boldsymbol{x}, m) \equiv p_m^i = \frac{N_m^i}{N_m}$$

$$\mathcal{I}_m = -\sum_{i=1}^{K} p_m^i \log_2 p_m^i$$

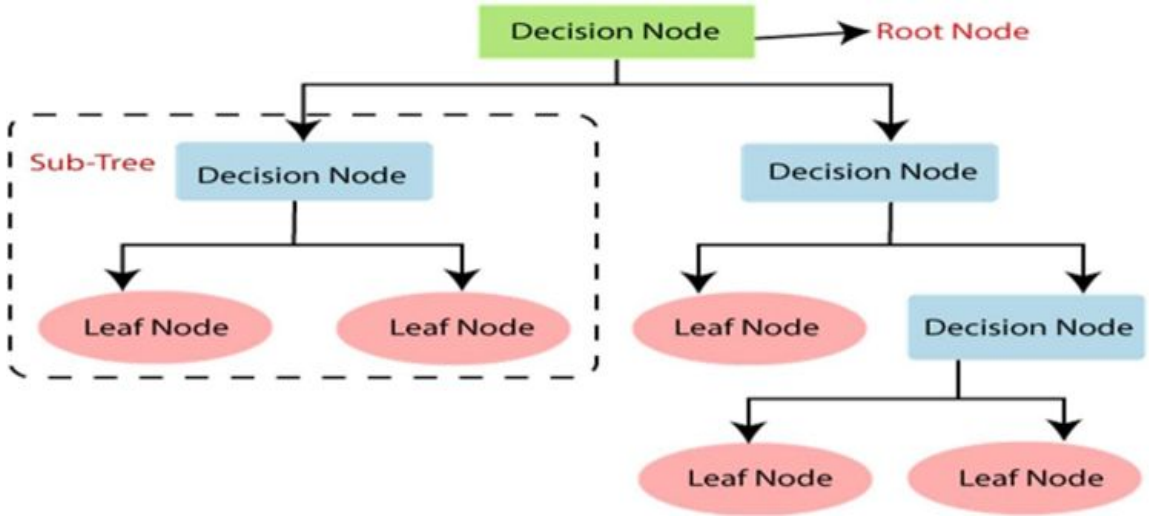
classification

$$E_m = \frac{1}{N_m} \sum_t (r^t - g_m)^2 b_m(\boldsymbol{x}^t)$$

$$b_m(\boldsymbol{x}) = \begin{cases} 1 & \text{if } \boldsymbol{x} \in \mathcal{X}_m: \boldsymbol{x} \text{ reaches node } m \\ 0 & \text{otherwise} \end{cases}$$

$$g_m = \frac{\sum_t b_m(\boldsymbol{x}^t) r^t}{\sum_t b_m(\boldsymbol{x}^t)}$$

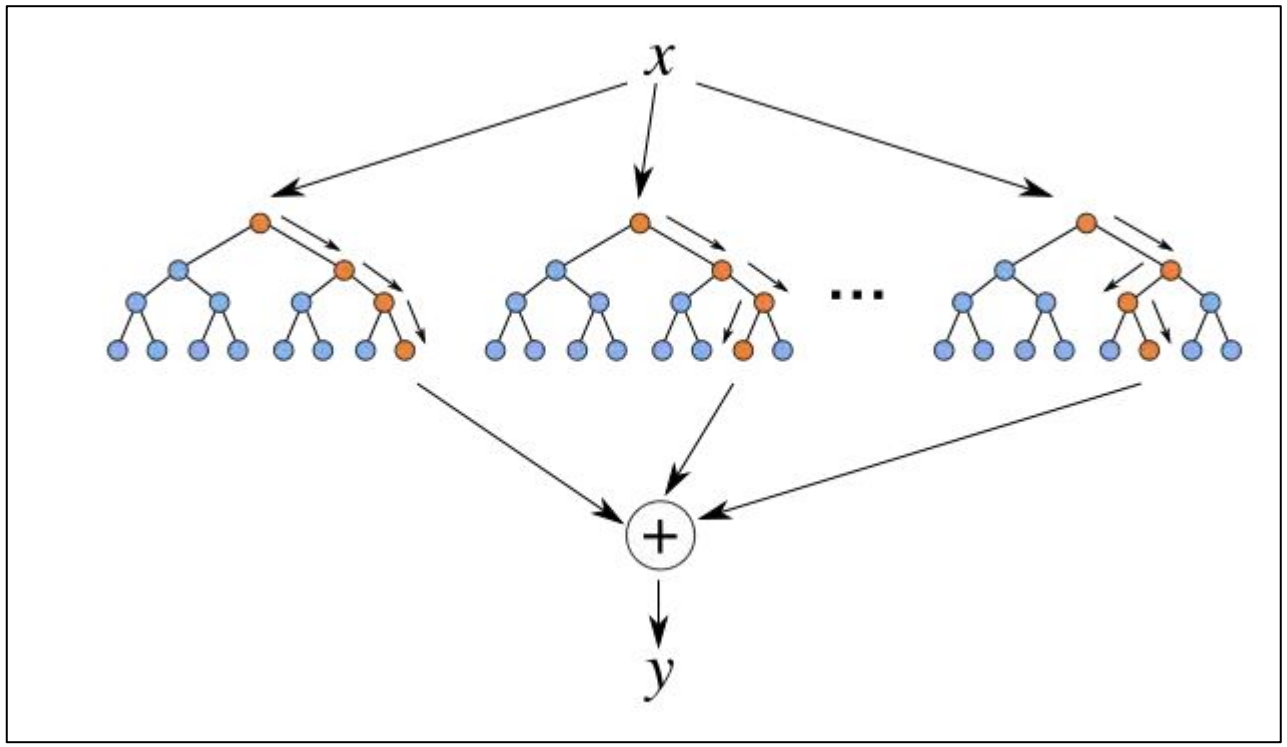
Regression



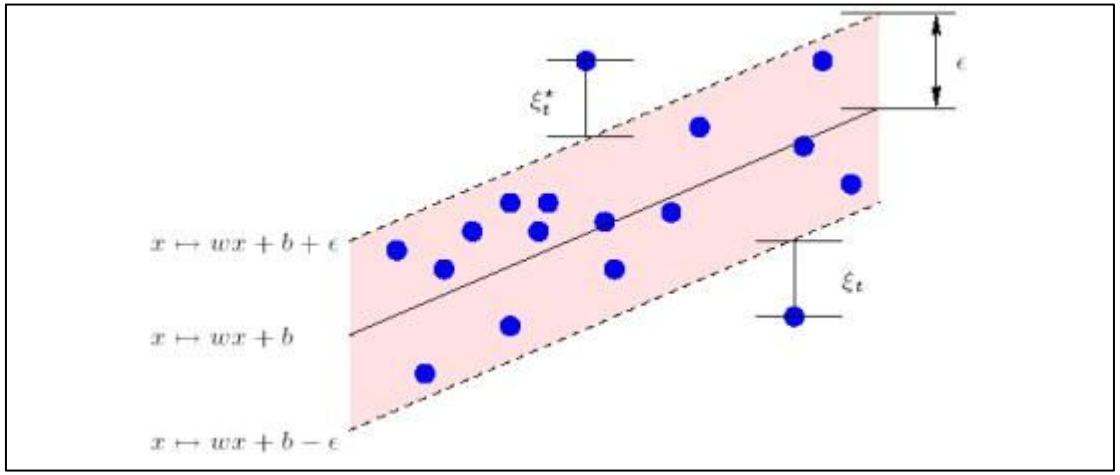
Arizona State University

Random Forest Algorithm:

1. Compared to decision tree algorithms, it is more accurate.
2. Offers a practical means of handling missing data.
3. Able forecast outcomes sensibly without over-adjusting the settings.
4. Fixes the overfitting issue with decision trees.
5. A subset of the elements at each node split point in the random forest tree is chosen at random.



Support Vector Regression



- **Support Vector Regression is a Supervised Learning Algorithm that is used to predict discrete values**
- **The aim of this algorithm is to find the best fit line**
- **SVR tries to fit the best fit line into threshold**

Support Vector Regression

$$e_\epsilon(r^t, f(\boldsymbol{x}^t)) = \begin{cases} 0 & \text{if } |r^t - f(\boldsymbol{x}^t)| < \epsilon \\ |r^t - f(\boldsymbol{x}^t)| - \epsilon & \text{otherwise} \end{cases}$$

- The Data Should be uniform in-order to use SVM Regression Efficiently.
- Model will not work efficiently with Inconsistent data

```
from sklearn.svm import SVR
regr = SVR()
regr.fit(X_train,y_train)
y_pred = regr.predict(X_test)
#print errors
from sklearn import metrics
print('MAE:', metrics.mean_absolute_error(y_test, y_pred))
MAE.append(metrics.mean_absolute_error(y_test, y_pred))
print('MSE:', metrics.mean_squared_error(y_test, y_pred))
MSE.append(metrics.mean_squared_error(y_test, y_pred))
print('RMSE:', np.sqrt(metrics.mean_squared_error(y_test, y_pred)))
RMSE.append(np.sqrt(metrics.mean_squared_error(y_test, y_pred)))
R2 = metrics.r2_score(y_test,y_pred)
print('R2:',R2)
r2.append(R2)
```

```
MAE: 3.4733767817967647
MSE: 37.359024978589375
RMSE: 6.112202956266208
R2: -0.16872087545442205
```

Inconsistent Data

```
from sklearn.svm import SVR
regr = SVR()
regr.fit(X_train,y_train)
y_pred = regr.predict(X_test)
#print errors
from sklearn import metrics
print('MAE:', metrics.mean_absolute_error(y_test, y_pred))
print('MSE:', metrics.mean_squared_error(y_test, y_pred))
print('RMSE:', np.sqrt(metrics.mean_squared_error(y_test, y_pred)))
R2 = metrics.r2_score(y_test,y_pred)
print('R2:',R2)
```

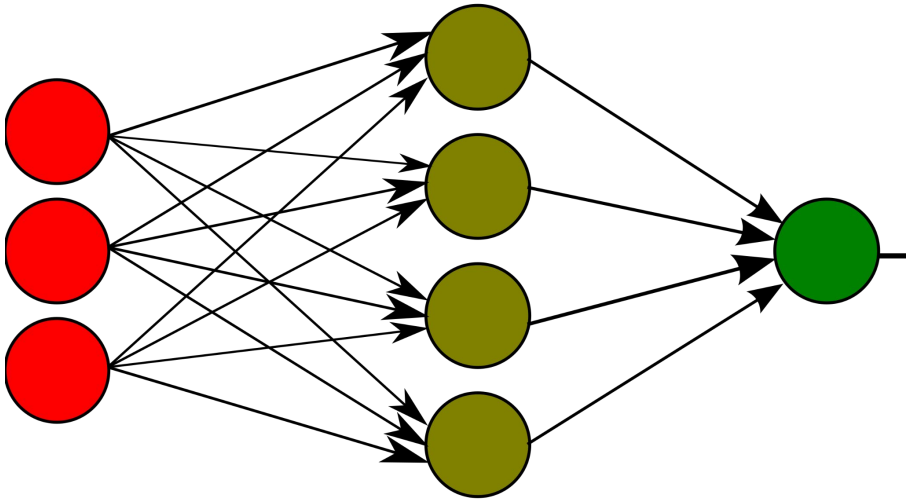
```
MAE: 1.516915708790308
MSE: 13.976464519523276
RMSE: 3.7385110029961495
R2: 0.56276734046522
```

Uniform Data (MinMaxScaler)

Arizona State University

Image Reference: https://stats.stackexchange.com/questions/259018/meaning-of-epsilon-in-svm-regression

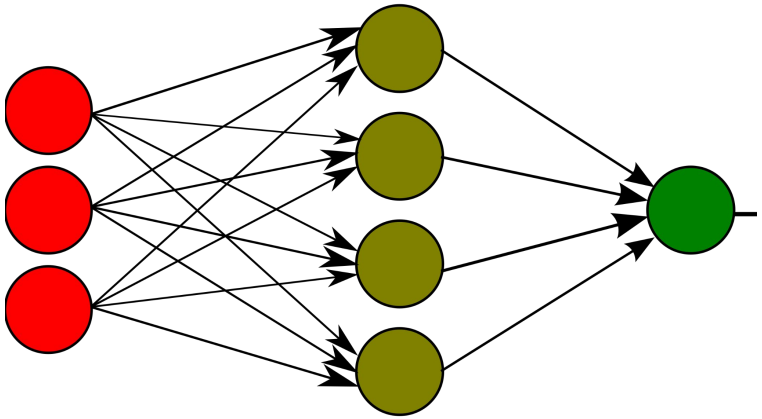
Multi-layer Perceptron



- The perceptron is a basic computing element that takes input from an environment or another system And gives an output or multiple outputs.
- Each Perceptron activates based on the input it gets and the activation function

$$y = \sum_{j=1}^{d} w_j x_j + w_0$$

Multi-layer Perceptron



- Activation function fires up when input is greater than threshold and gives 1 in output
- In classification we choose the class depending on activation function

$$y = \sum_{j=1}^{d} w_j x_j + w_0$$

$$s(a) = \begin{cases} 1 & \text{if } a > 0 \\ 0 & \text{otherwise} \end{cases}$$

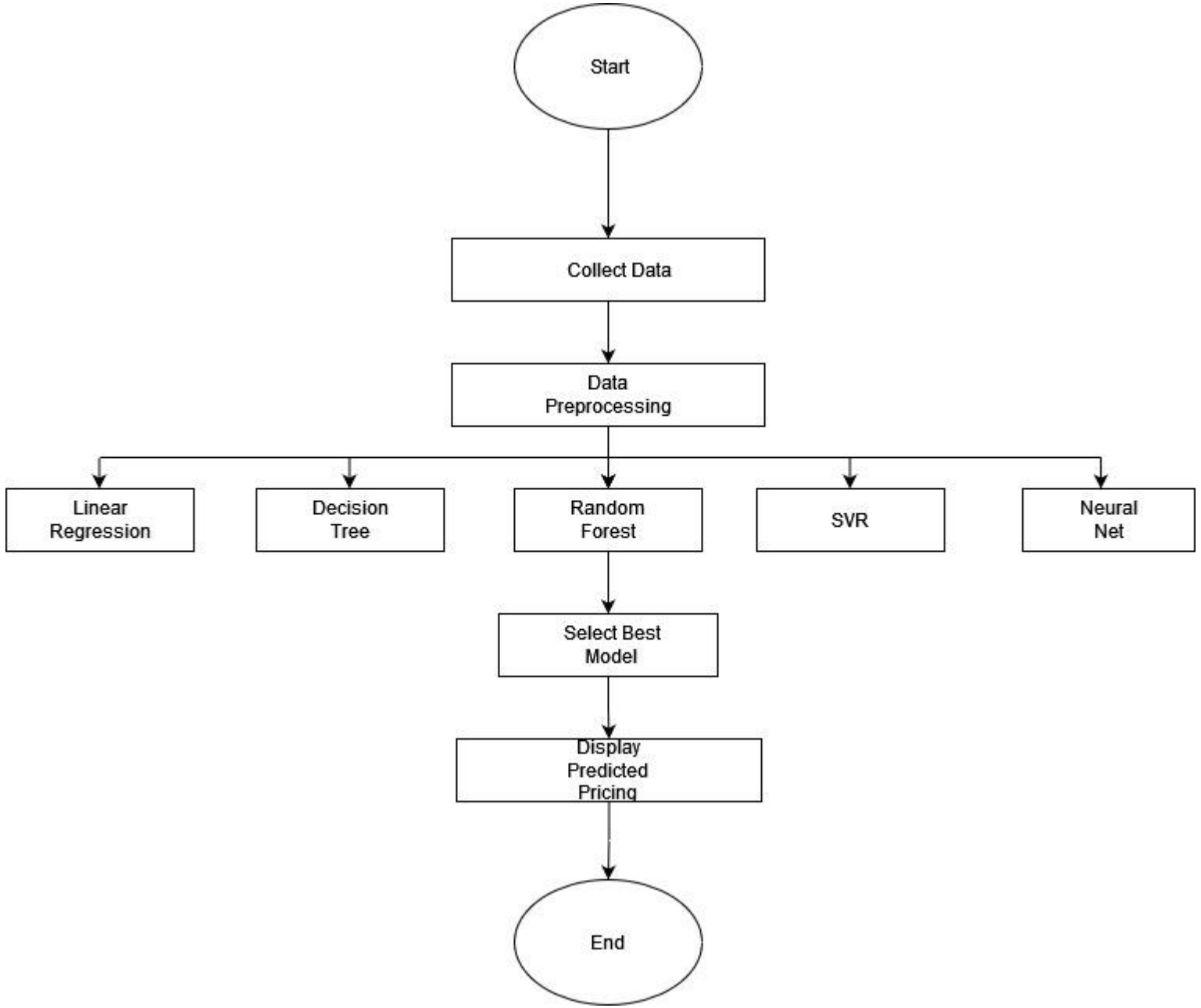
then we can

$$\text{choose} \begin{cases} C_1 & \text{if } s(\mathbf{w}^T\mathbf{x}) > 0 \\ C_2 & \text{otherwise} \end{cases}$$

For $i = 1, \ldots, K$
  For $j = 0, \ldots, d$
    $w_{ij} \leftarrow \text{rand}(-0.01, 0.01)$
Repeat
  For all $(\mathbf{x}^t, r^t) \in \mathcal{X}$ in random order
    For $i = 1, \ldots, K$
      $o_i \leftarrow 0$
      For $j = 0, \ldots, d$
        $o_i \leftarrow o_i + w_{ij} x_j^t$
    For $i = 1, \ldots, K$
      $y_i \leftarrow \exp(o_i) / \sum_k \exp(o_k)$
    For $i = 1, \ldots, K$
      For $j = 0, \ldots, d$
        $w_{ij} \leftarrow w_{ij} + \eta(r_i^t - y_i) x_j^t$
Until convergence

Arizona State University

Data Flow



Arizona State University

System Evaluation:

Mean Absolute Error (MAE): The absolute average difference between the observed and predicted data is measured, but significant prediction errors are not penalized.

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n} |x_i - x|$$

Mean Square Error (MSE): The average squared distance between the actual data and the predicted data is measured here.

MSE formula = $(1/n) * \Sigma(\text{actual} - \text{forecast})^2$

Root mean square error : It shows how far predictions fall from measured true values using Euclidean distance**.** It is one of the most used metrics for measuring forecast quality is the root mean square error or root mean square deviation. RMSE requires and utilizes actual measurements at each anticipated data point, it is frequently utilized in supervised educational applications.

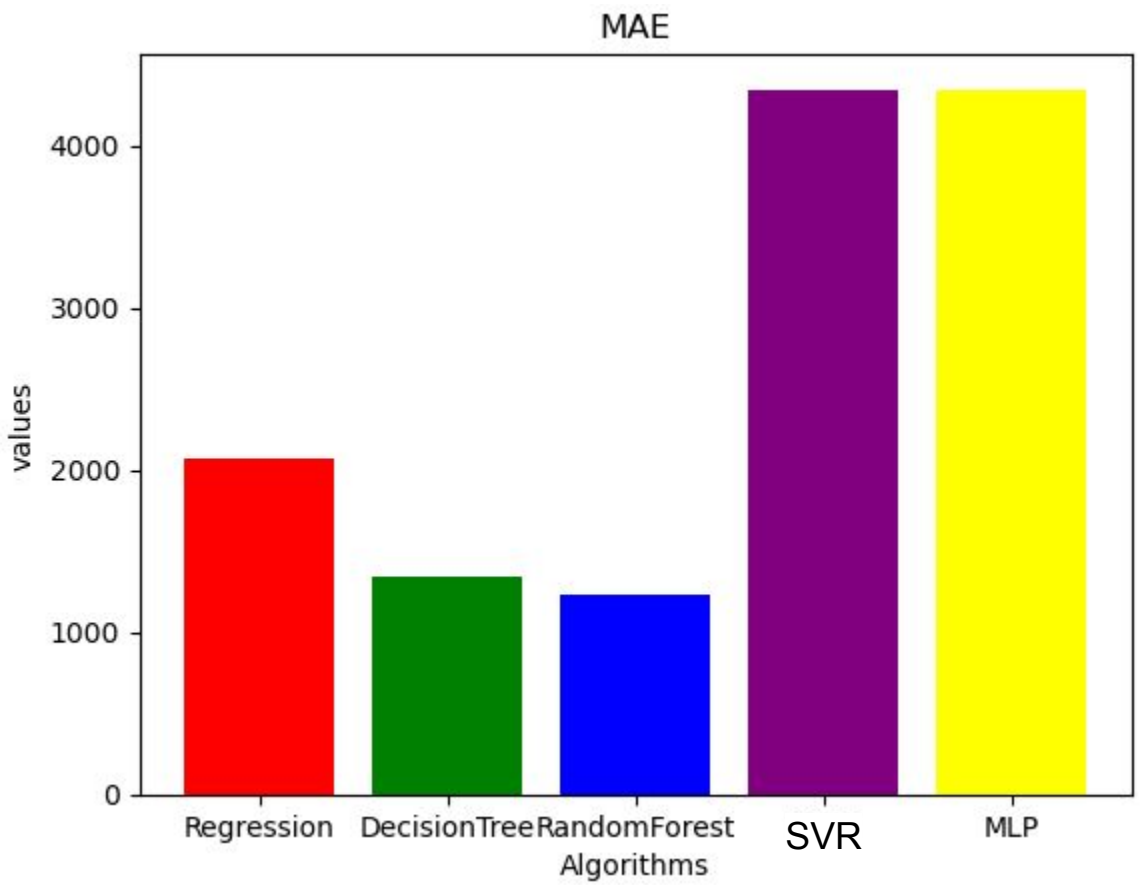
$$RMSE = \sqrt{\overline{(f-o)^2}}$$

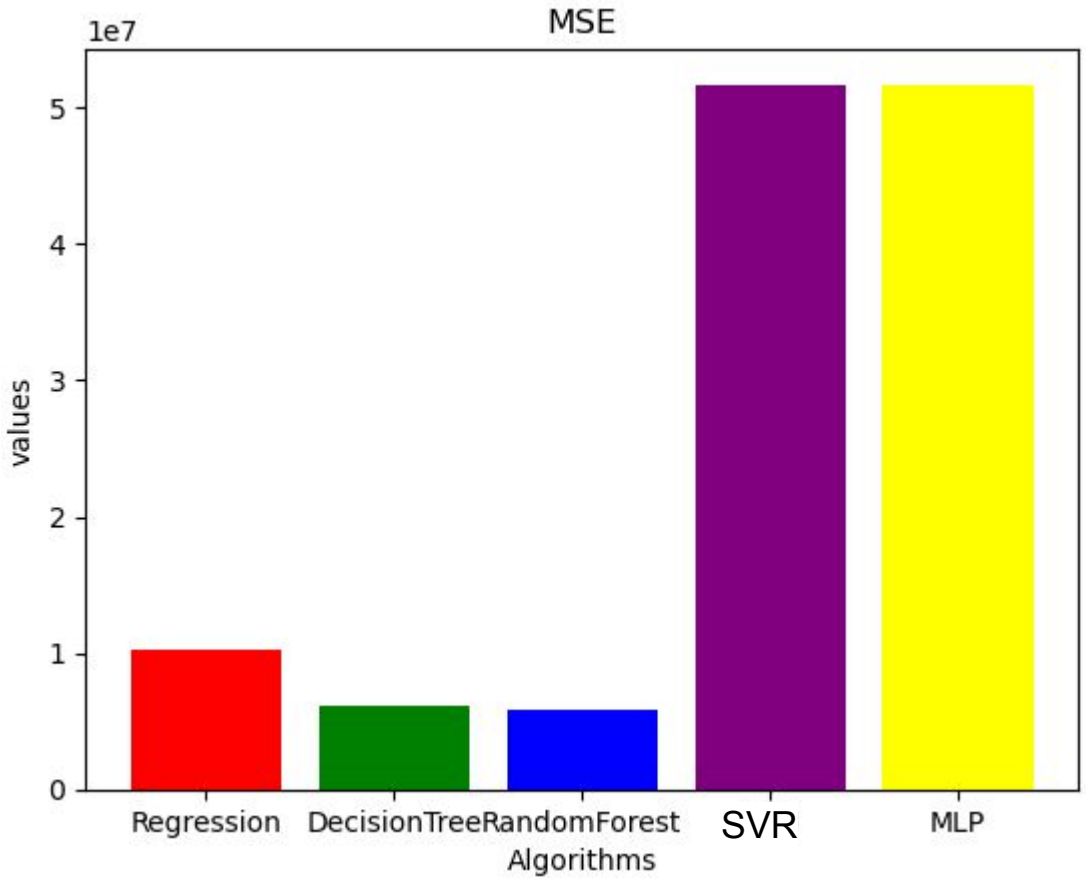
R squared score:

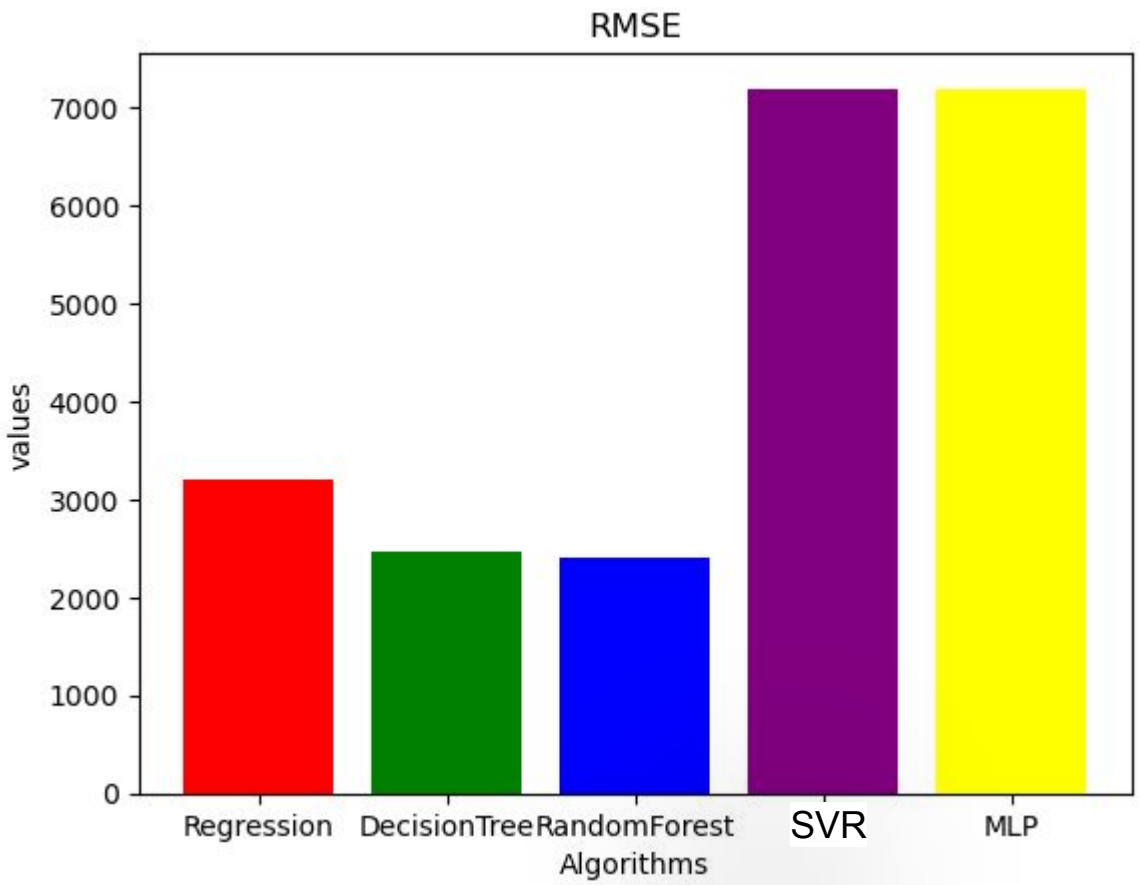
A crucial indicator for assessing the effectiveness of a regression-based machine learning model is the R2 score. The coefficient of determination, commonly known as R squared, is pronounced as R sq. It measures how much variance in the forecasts that the data set can account for.

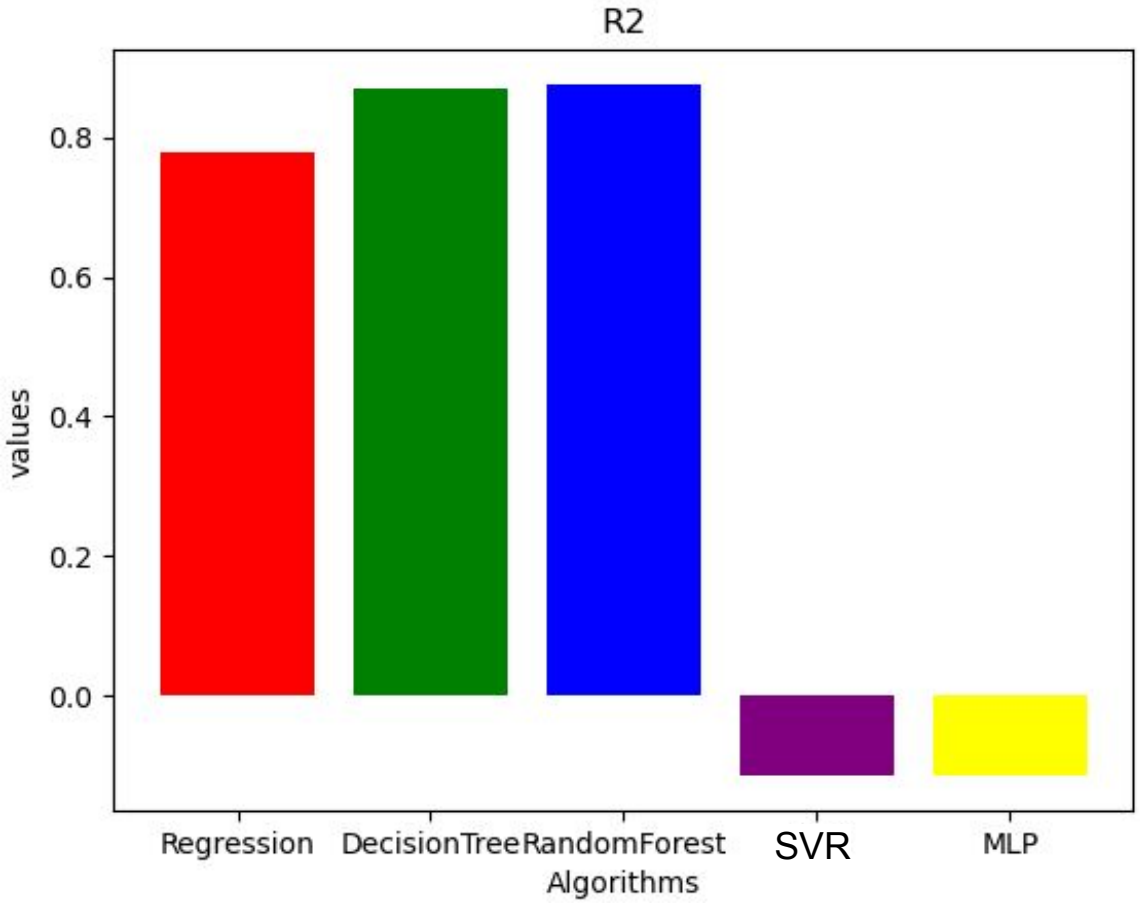
The formula is:

$$R^2_{adj} = 1 - \left[\frac{(1 - R^2)(n-1)}{n-k-1} \right]$$









Arizona State University

Conclusion: As new car prices have soared and consumers can no longer afford them, used car sales are rising globally. Because of this, there is a critical need for a system that can accurately estimate the worth of a used car from various attributes. With the proposed (random forest) approach, it is possible to anticipate used car prices with greater accuracy.

Arizona State University

Futurescope: Future connections between this machine learning model and several websites that offer real-time data for price prediction are possible. Large historical car price data can be added as well, which will assist the machine learning model's accuracy. As a user interface to communicate with the user, we can construct an Android application.

References:

1. https://www.kaggle.com/jpayne/852k-used-car-listings
2. N. Monburinon, P. Chertchom, T. Kaewkiriya, S. Rungpheung, S. Buya and P. Boonpou, "Prediction of prices for used car by using different models," 2018 5th International Conference on Business and Industrial
3. Research (ICBIR), Bangkok, 2018, pp. 115-119.
3.https://scikit-learn.org/stable/modules/classes.html: Scikit-learn: Machine Learning in Python,
Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.
4. https://github.com/topics/car-price-prediction
5. DATASET: https://www.kaggle.com/nehalbirla/vehicle-dataset-from-cardekho?select=car+data.csv
6. Sameerchand Pudaruth, “Predicting the Price of Used Cars using Machine Learning Techniques” ;(IJICT 2014).
7. Ning sun, Hongxi Bai, Yuxia Geng, Huizhu Shi, “Price Evaluation Model in Second Hand Car
System Based on BP Neural Network Theory”; (Hohai University Changzhou, China).

Thank you!

Arizona State University