Contents

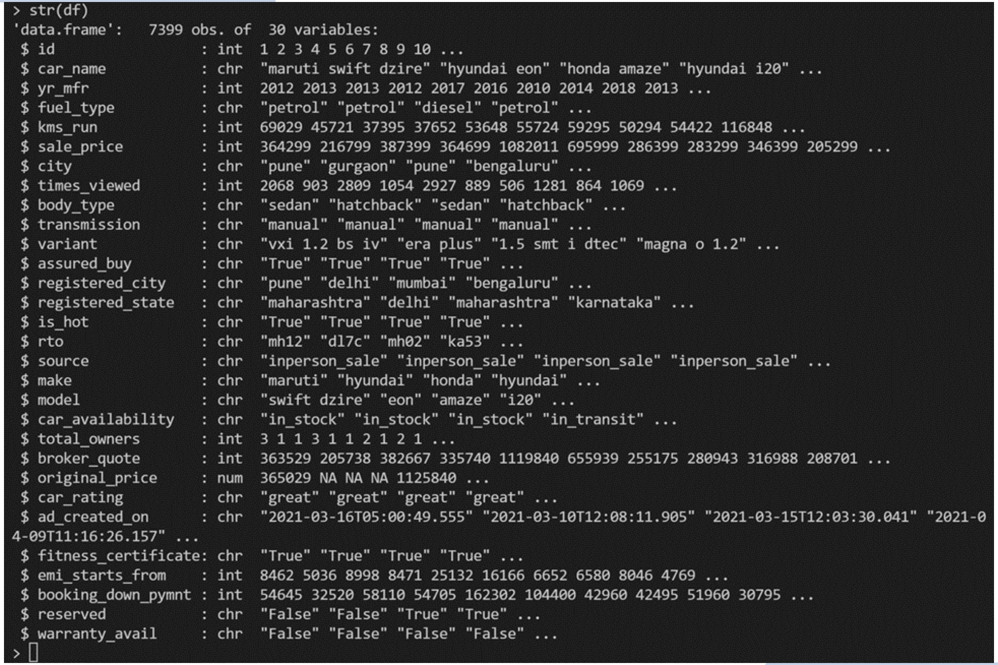
- Problem Statement
- Dataset
- Data Inspection
- Data Cleansing
- Relationships
- Linear Regression
- k-NN Classification
- Conclusion

Problem Statement

- The basic proposition is how do individuals, and resellers identify the good and bad deals to maximize their purchases or profits when dealing with the used car market.
- This is an ever evolving and changing dynamic based upon personal preferences, status, social trends, country-based policies, and the global marketplace.
- A dynamic recently is the introduction of all electric cars.
- Our goal is analyzing and predicting that for a given make, model, and variant, what are the major factors (attributes) that determine resale value.

The Dataset

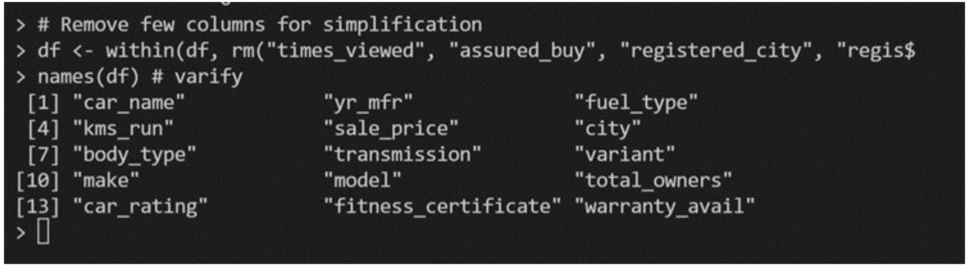
- The original dataset contained 7399 instances of true value cars data across all major tier 1 and tier 2 cities in India.
- There were 30 different features in the dataset.



```
> str(df)
'data.frame':   7399 obs. of  30 variables:
 $ id                  : int  1 2 3 4 5 6 7 8 9 10 ...
 $ car_name            : chr  "maruti swift dzire" "hyundai eon" "honda amaze" "hyundai i20" ...
 $ yr_mfr              : int  2012 2013 2013 2012 2017 2016 2010 2014 2018 2013 ...
 $ fuel_type           : chr  "petrol" "petrol" "diesel" "petrol" ...
 $ kms_run             : int  69029 45721 37395 37652 53648 55724 59295 50294 54422 116848 ...
 $ sale_price          : int  364299 216799 387399 364699 1082011 695999 286399 283299 346399 205299 ...
 $ city                : chr  "pune" "gurgaon" "pune" "bengaluru" ...
 $ times_viewed        : int  2068 903 2809 1054 2927 889 506 1281 864 1069 ...
 $ body_type           : chr  "sedan" "hatchback" "sedan" "hatchback" ...
 $ transmission        : chr  "manual" "manual" "manual" "manual" ...
 $ variant             : chr  "vxi 1.2 bs iv" "era plus" "1.5 smt i dtec" "magna o 1.2" ...
 $ assured_buy         : chr  "True" "True" "True" "True" ...
 $ registered_city     : chr  "pune" "delhi" "mumbai" "bengaluru" ...
 $ registered_state    : chr  "maharashtra" "delhi" "maharashtra" "karnataka" ...
 $ is_hot              : chr  "True" "True" "True" "True" ...
 $ rto                 : chr  "mh12" "dl7c" "mh02" "ka53" ...
 $ source              : chr  "inperson_sale" "inperson_sale" "inperson_sale" "inperson_sale" ...
 $ make                : chr  "maruti" "hyundai" "honda" "hyundai" ...
 $ model               : chr  "swift dzire" "eon" "amaze" "i20" ...
 $ car_availability    : chr  "in_stock" "in_stock" "in_stock" "in_transit" ...
 $ total_owners        : int  3 1 1 3 1 1 2 1 2 1 ...
 $ broker_quote        : int  363529 205738 382667 335740 1119840 655939 255175 280943 316988 208701 ...
 $ original_price      : num  365029 NA NA NA 1125840 ...
 $ car_rating          : chr  "great" "great" "great" "great" ...
 $ ad_created_on       : chr  "2021-03-16T05:00:49.555" "2021-03-10T12:08:11.905" "2021-03-15T12:03:30.041" "2021-0
4-09T11:16:26.157" ...
 $ fitness_certificate: chr  "True" "True" "True" "True" ...
 $ emi_starts_from     : int  8462 5036 8998 8471 25132 16166 6652 6580 8046 4769 ...
 $ booking_down_pymnt : int  54645 32520 58110 54705 162302 104400 42960 42495 51960 30795 ...
 $ reserved            : chr  "False" "False" "True" "True" ...
 $ warranty_avail      : chr  "False" "False" "False" "False" ...
> 
```

Dataset Inspection

- During inspection of the data, it was noticed that the dataset contains an “id” column which was not unique thus it was removed.
- Further inspect was done to find missing values that could be represented by NAs, NULL or empty cells (also by “”).
- Dataset contained 29 features but not all played an important role in influencing used car sale price and were thus removed.
- The updated dataset has 15 features.

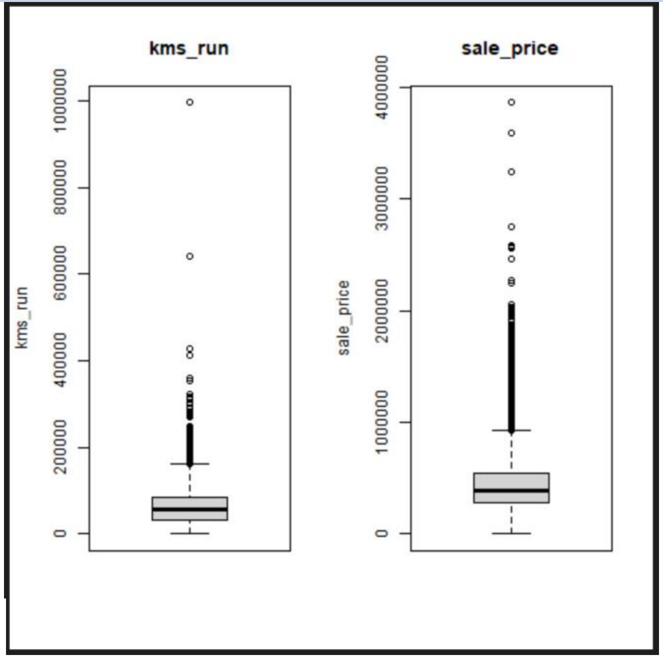


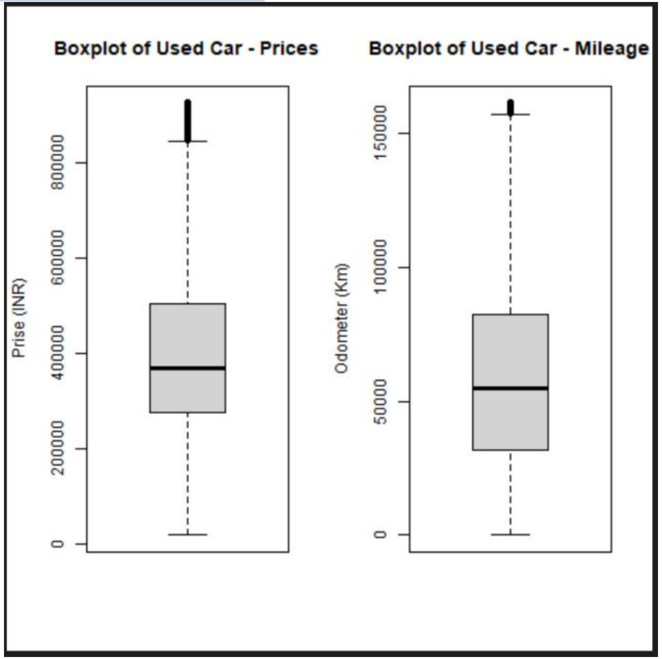
```
> # Remove few columns for simplification
> df <- within(df, rm("times_viewed", "assured_buy", "registered_city", "regis$
> names(df) # varify
 [1] "car_name"          "yr_mfr"             "fuel_type"
 [4] "kms_run"           "sale_price"         "city"
 [7] "body_type"         "transmission"       "variant"
[10] "make"              "model"              "total_owners"
[13] "car_rating"        "fitness_certificate" "warranty_avail"
> 
```

Data Cleansing

- Looking at the “sale_price” detail, found 3 rows with 0 sale price value which mostly likely is an error. To fix that, rows with similar attributes were searched and a Median value was computed.
- There were 414 data points in “sale_price” that were above 928149 sale price value. These were dropped as outliers.
- There were 165 data points in “kms_run” that were over 161605 kms run value. These were dropped as outliers.
- “body_type” , “car_rating”, and “fitness_certificate” had empty cells, these were calculated and inserted into the dataset
- “car_name” contained same information as “make” and “model” and was removed.

Data Cleansing

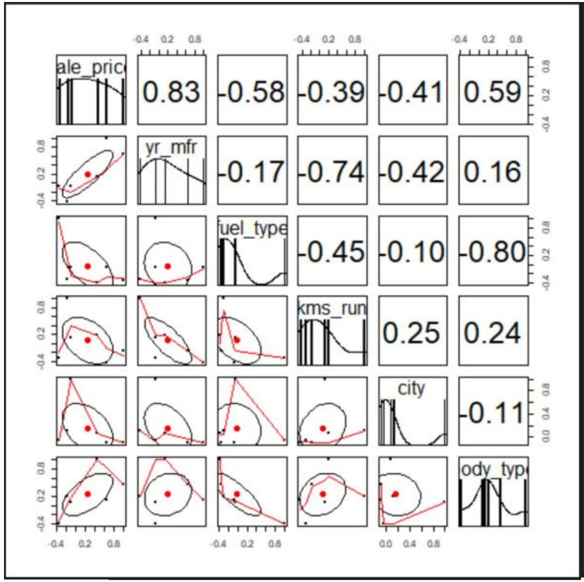


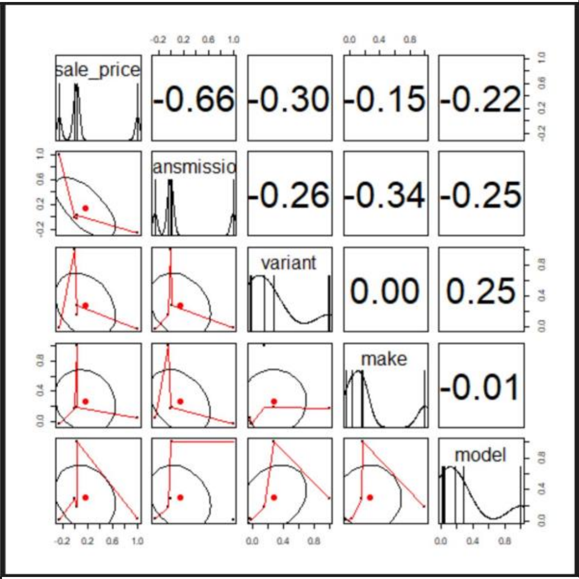


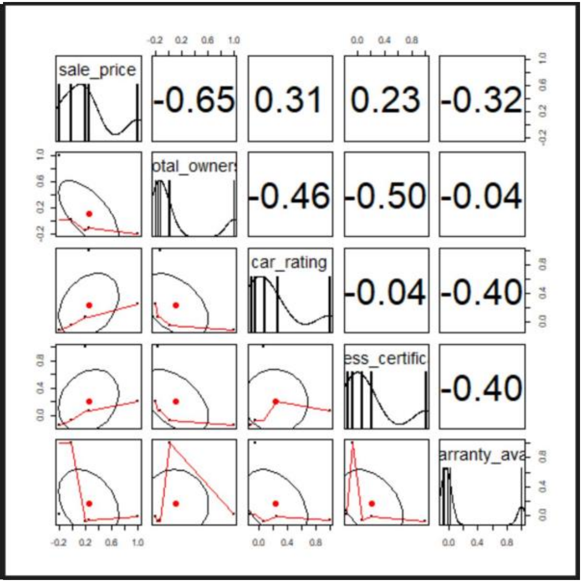
Outliers: Before

After

Visualizing Relationships







Visualizing Relationships

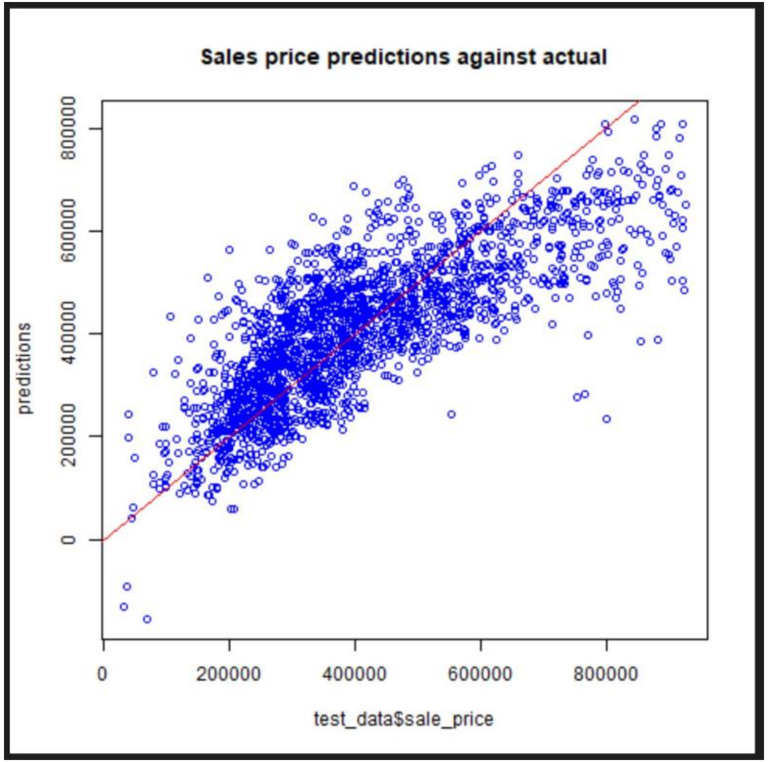
- As expected, a correlation plot shows strong positive relation between “sale_price” and “yr_mfr” (year manufactured) i.e., newer cars are more expensive
- There is a strong negative correlation between “sale_price” and “transmission” i.e., automatic used cars are more expansive than manual cars.
- There is a strong negative correlation between “sale_price” and “total_owners” i.e., more owners reduce the price.
- The correlation between “sale_price” and “make”, “model” and “fitness_certificate” are weak and thus will be excluded.

Linear Regression

- The dataset was randomly split into 70 percent (train_data) and 30 percent (test_data).
- In the lm() function, “sale_price” was used as the dependent variable and “kms_run” + “transmission” + “car_rating” + “total_owners” + “warranty_avail” + “variant” + “yr_mfr” + “fuel_type” + “body_type” + “city” were used as independent variables.
- The beta coefficients from the model indicates the estimated increase in sale price for an increase of one in each of the features provided all other values are held constant.
- For example, for each additional year in “yr_mfr”, we would expect sale price to increase by 30189 on average provided everything else is equal.

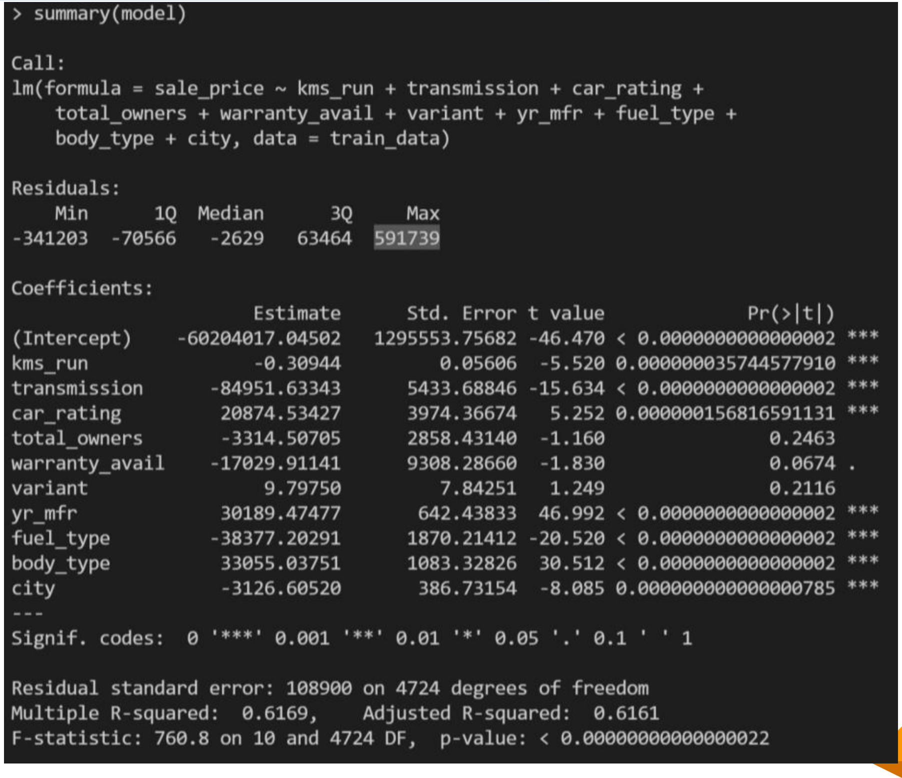
Linear Regression

Plotting model prediction



Linear Regression

Model Evaluation



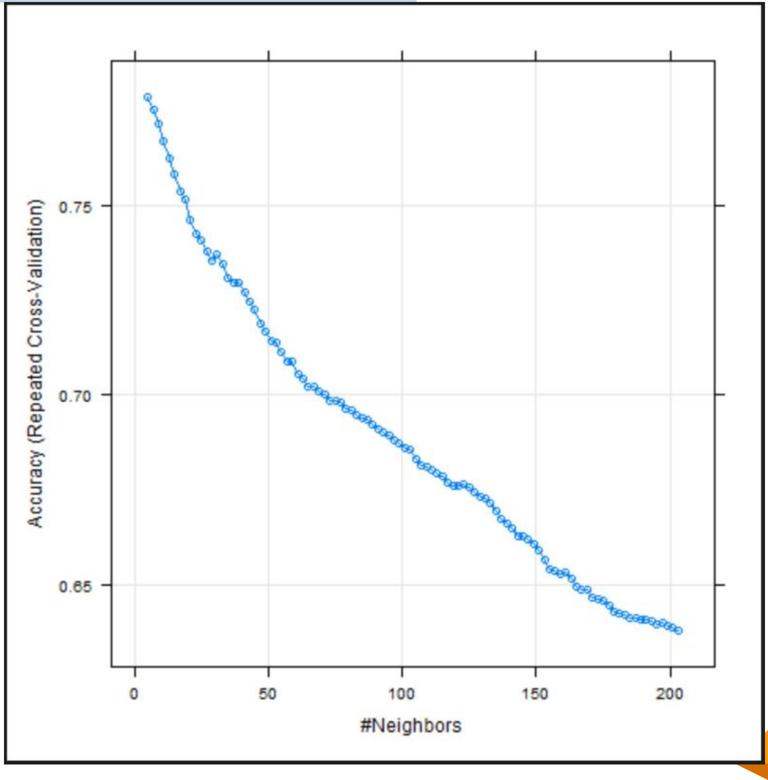
```
> summary(model)

Call:
lm(formula = sale_price ~ kms_run + transmission + car_rating + 
    total_owners + warranty_avail + variant + yr_mfr + fuel_type + 
    body_type + city, data = train_data)

Residuals:
    Min      1Q  Median      3Q     Max 
-341203  -70566   -2629   63464  591739 

Coefficients:
                      Estimate      Std. Error t value             Pr(>|t|)    
(Intercept)    -60204017.04502   1295553.75682 -46.470 < 0.0000000000000002 ***
kms_run               -0.30944         0.05606  -5.520 0.000000035744577910 ***
transmission      -84951.63343      5433.68846 -15.634 < 0.0000000000000002 ***
car_rating         20874.53427      3974.36674   5.252 0.000000156816591131 ***
total_owners       -3314.50705      2858.43140  -1.160               0.2463    
warranty_avail    -17029.91141      9308.28660  -1.830               0.0674 .  
variant                9.79750         7.84251   1.249               0.2116    
yr_mfr             30189.47477       642.43833  46.992 < 0.0000000000000002 ***
fuel_type         -38377.20291      1870.21412 -20.520 < 0.0000000000000002 ***
body_type          33055.03751      1083.32826  30.512 < 0.0000000000000002 ***
city               -3126.60520       386.73154  -8.085 0.000000000000000785 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 108900 on 4724 degrees of freedom
Multiple R-squared:  0.6169,	Adjusted R-squared:  0.6161 
F-statistic: 760.8 on 10 and 4724 DF,  p-value: < 0.00000000000000022
```

k-NN Classification

- The same used car dataset was used and therefore did not require data cleansing. Independent features were all factorized and converted to numeric with “sale_price” as the outcome.
- The normalized dataset was split into 80 percent for training purpose and 20 percent for testing purposes.
- “sale_price” was not a category type yet. This feature was converted to category type with 3 levels – high, medium, and low.
- The target variable (sale_price) was used to create class labels.
- To implement k-NN algorithm, knn() function was used from class package.
- To find the optimal value for k, train() function was used. The result showed that 5 was the best value for k.

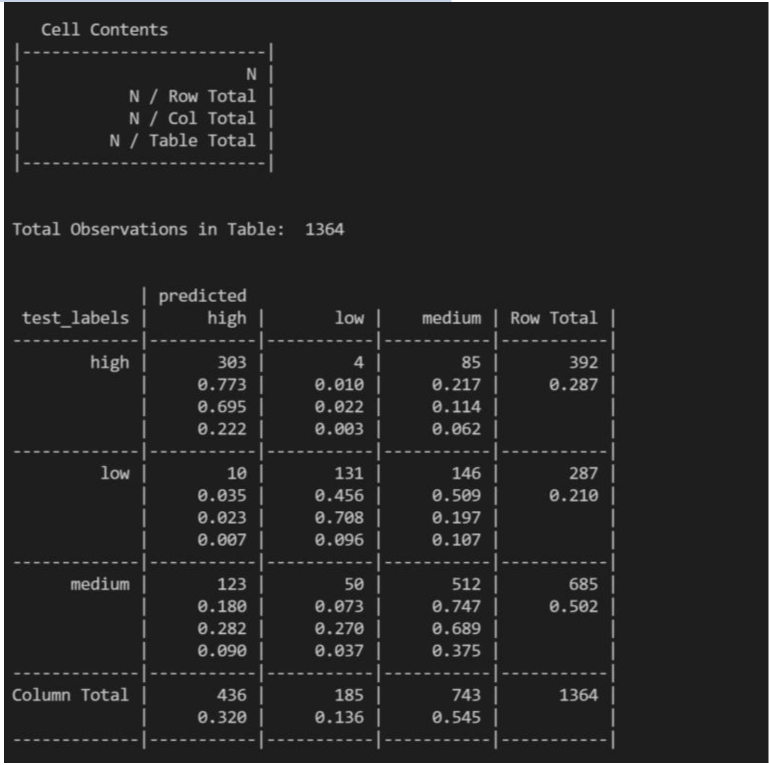
k-NN Classification

Train() function shows that optimal value for k is 5.



k-NN Classification

Model Evaluation using Cross Table

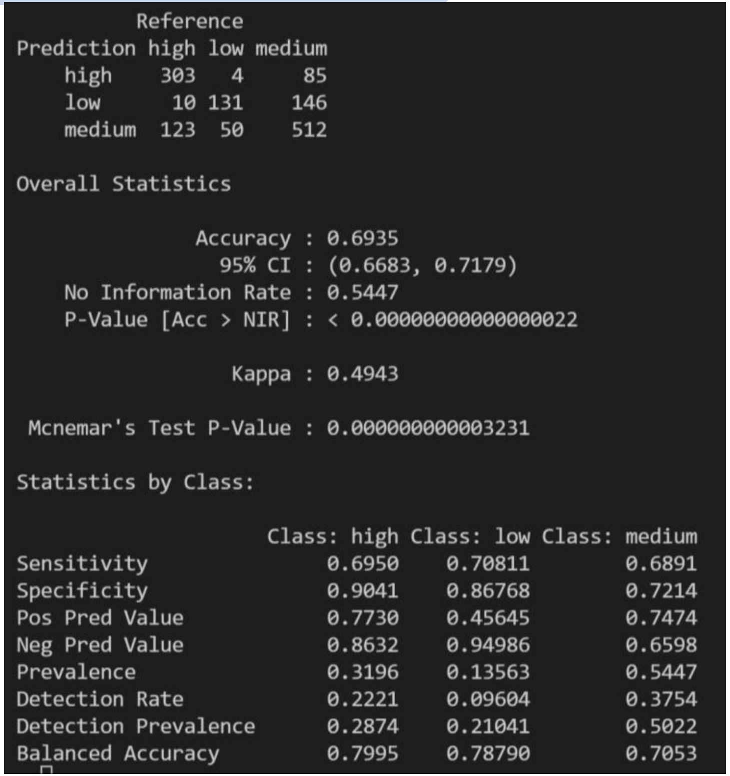


```
  Cell Contents
|-------------------------|
|                       N |
|           N / Row Total |
|           N / Col Total |
|         N / Table Total |
|-------------------------|


Total Observations in Table:  1364


             | predicted
 test_labels |      high |       low |    medium | Row Total |
-------------|-----------|-----------|-----------|-----------|
        high |       303 |         4 |        85 |       392 |
             |     0.773 |     0.010 |     0.217 |     0.287 |
             |     0.695 |     0.022 |     0.114 |           |
             |     0.222 |     0.003 |     0.062 |           |
-------------|-----------|-----------|-----------|-----------|
         low |        10 |       131 |       146 |       287 |
             |     0.035 |     0.456 |     0.509 |     0.210 |
             |     0.023 |     0.708 |     0.197 |           |
             |     0.007 |     0.096 |     0.107 |           |
-------------|-----------|-----------|-----------|-----------|
      medium |       123 |        50 |       512 |       685 |
             |     0.180 |     0.073 |     0.747 |     0.502 |
             |     0.282 |     0.270 |     0.689 |           |
             |     0.090 |     0.037 |     0.375 |           |
-------------|-----------|-----------|-----------|-----------|
Column Total |       436 |       185 |       743 |      1364 |
             |     0.320 |     0.136 |     0.545 |           |
-------------|-----------|-----------|-----------|-----------|
```

k-NN Classification

Model Evaluation using

Confusion Matrix



```
          Reference
Prediction high low medium
    high    303   4     85
    low      10 131    146
    medium  123  50    512

Overall Statistics

               Accuracy : 0.6935
                 95% CI : (0.6683, 0.7179)
    No Information Rate : 0.5447
    P-Value [Acc > NIR] : < 0.0000000000000022

                  Kappa : 0.4943

 Mcnemar's Test P-Value : 0.000000000003231

Statistics by Class:

                     Class: high Class: low Class: medium
Sensitivity               0.6950    0.70811        0.6891
Specificity               0.9041    0.86768        0.7214
Pos Pred Value            0.7730    0.45645        0.7474
Neg Pred Value            0.8632    0.94986        0.6598
Prevalence                0.3196    0.13563        0.5447
Detection Rate            0.2221    0.09604        0.3754
Detection Prevalence      0.2874    0.21041        0.5022
Balanced Accuracy         0.7995    0.78790        0.7053
```

Conclusion

- This was not an ideal dataset, it had missing values, outliers, duplicated fields, and insignificant fields.
- Correlations between Sale Price and Manufacturing Year, Transmission Type, car rating, fuel type, body type, odometer and the Number of Owners was proven significant.
- These are the only factors that should be used as a guideline in buying used cars in India from the supplied dataset.