Supervised Learning and Model Building

Case Study and Model Building for

ReCell - Refurbished Smartphones

Date: 11/11/2022

Mona Desai

Contents / Agenda

Executive Summary

- **Conclusions, actionable insights:**
  - The Model predicts “normaliszed_used_price” with 84% accuracy.
  - From the results of the Model, observations:
    - “Normalized_used_price” increases by 0.021 unit if the “main_camera_mp” and 0.014 unit if “selfie_camera_mp” increases by one unit
    - “Normalized_used_price_ increases by 0.4356 unit for every unit increase in “normalized_new_price”
    - “Normalized_used_price” increases merely 0.0017 unit for every unit increase in “weight”
    - Though “Normalized_used_price” seems to increase 0.0489 for every unit increase in 4G, there might be multicollinearity exits with 5G. So these two variables should be looked together predicting the price.
    - Xiaomi & Karbon brand devices have higher influence on price. Sony and Samsung branded devices have lesser price impact.

- **Recommendations:**
  - ReCell to buy phones with higher RAM. From model - price of used device increases by .0212 for an unit increase in RAM. Prefer Karbon and Xiaomi devices for this reason
  - ReCell to focus on devices with higher MP main cameras as the model suggests it to be important factor in price increase
  - ReCell to focus on newer or less aged phones. The older phones losing value
  - ReCell to buy Sony & Samsung branded devices, which have higher camera specs and would sell more easily
  - Some devices from Motorola, Redmi & Vivo have better battery performance and still lighter. These are good to target travellers who need longer battery & lighter devices.

Business Problem Overview and Solution Approach

- **Problem**
  - The best Price discovery of used and refurbished phones and devices
  - Too many variables influencing the predictions of the "normalized_used_price"
  - If the Model we built predicts the best price using the statistical inference.
  - If the Model is able to explain more than 70% variation in data.
  - If the Model is not suffering from overfitting
  - If all the assumptions of the Linear Regression are satisfied
  - If there is any Multicollinearity exists

- **Solution approach / methodology**
  - Exploratory Data Analysis
  - Model Building - Linear Regression, Removing Multicollinearity
  - Analyzing R-squared, Adj. R-squared and the percentage of the MAPE value
  - Comparing the RMSE and MAE value for test and train dataset

EDA Results

- **Key results from EDA**
    - The Dataset contains different attributes of used cellphones and tablets.
    - The Data in the dataset
        - Collected in year 2021 - For Device period from 2013-2021
        - 3454 Rows
        - 15 Columns
    - 11 Numerical, 4 Categorical Columns and No duplicate Values
    - Statistical Summary suggests a lot of variations in Data for each Numerical columns
    - 6 out of 15 columns have missing values
    - Observation analyzing Boxplot and Histogram for each columns
        - “Normalized_used_price” distributed slightly right skewed, many outliers and average ~4.4 Normalized price in Euro
        - “Normalized_new_price” distributed Normally with many outliers and average ~5.2 Normalized price in Euro
        - “screen_size” distributed left skewed. 1200+ devices have ~13 cm screen size

Link to Appendix slide on data background check

EDA Results continued...

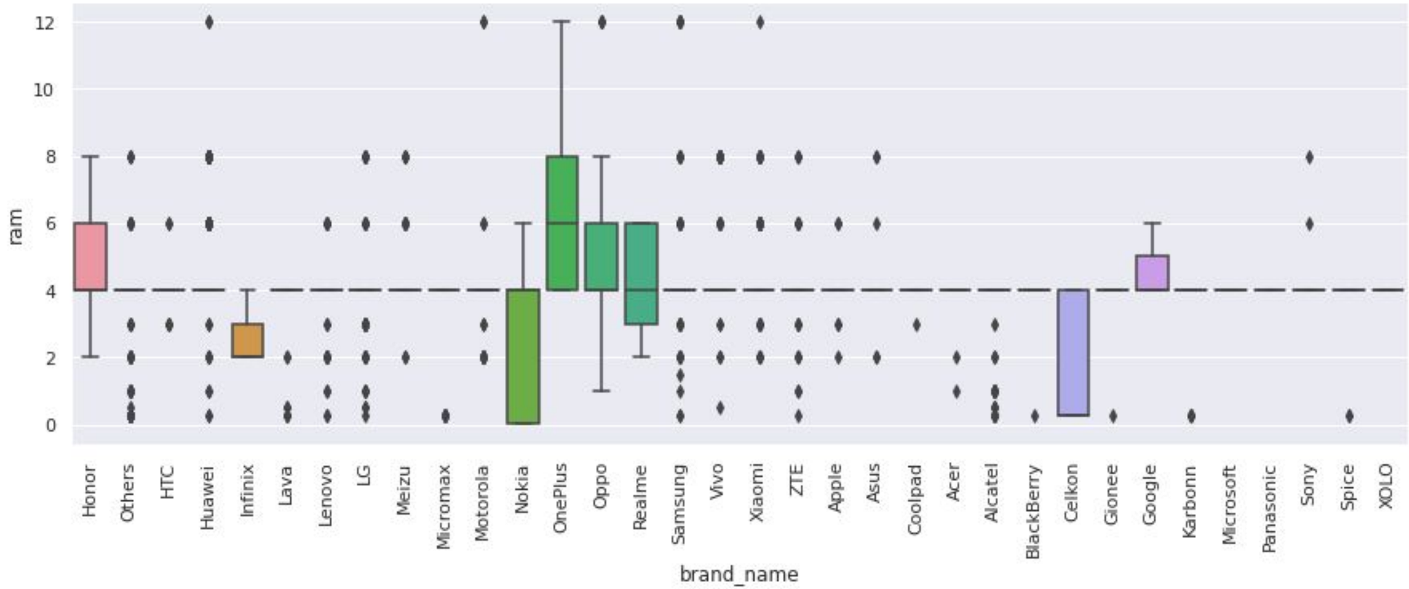
- Observation analyzing Boxplot and Histogram for each columns
  - ~500+ devices have 5 megapixel, 700+ devices have 8 megapixel and 1000+ devices have 13 megapixel resolution in the column “main_camera_mp”
  - “Selfie_camera_mp” distributed left skewed with ~800 devices have 5 megapixel resolution
  - “Int_memory” is highly left skewed. ~2500 devices have 30-40 memory (ROM) in GB
  - ~2700 devices have 4 GB RAM
  - “weight” distributed highly left skewed. Most devices weight between 100-200 gms.
  - ~50% of the devices have 2000, 3000 and 4000 capacity of the batteries in mAh
  - More than 50% devices are used for ~500-1200 days
  - ~10% of the total devices are of brand Samsung, ~7%Huawei and ~15% devices are of mix brands
  - ~90% devices are on “Android” OS.
  - Most devices 4g and very few are 5g
  - 642 devices are of year 2014, oldest devices are of year 2013 and newest are of year 2020

EDA Results continued...

- **Insight**
    - “normalised_used_price” and “normalized_new_price” have the strongest correlation.
    - “normalised_used_price” has ~60% correlation with “screen_size”,”main_camera_mp”,”selfie_camera_mp”,”ram” and “battery”
    - “screen_size” has a strong correlation with “weight” and “battery”
    - “battery” and “weight” shows strong correlation

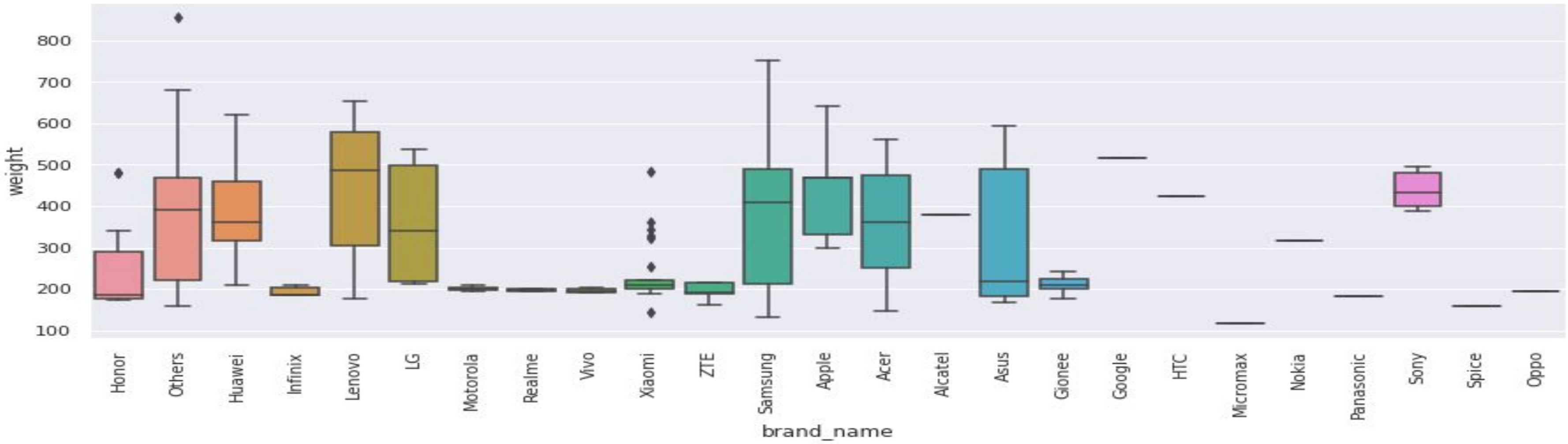
EDA Results continued...

- **Brand_name vs. RAM**
  - Most “Honor”, “OnePlus”, “Oppo”, “Realme”, “Google” devices have more than 4 RAM.



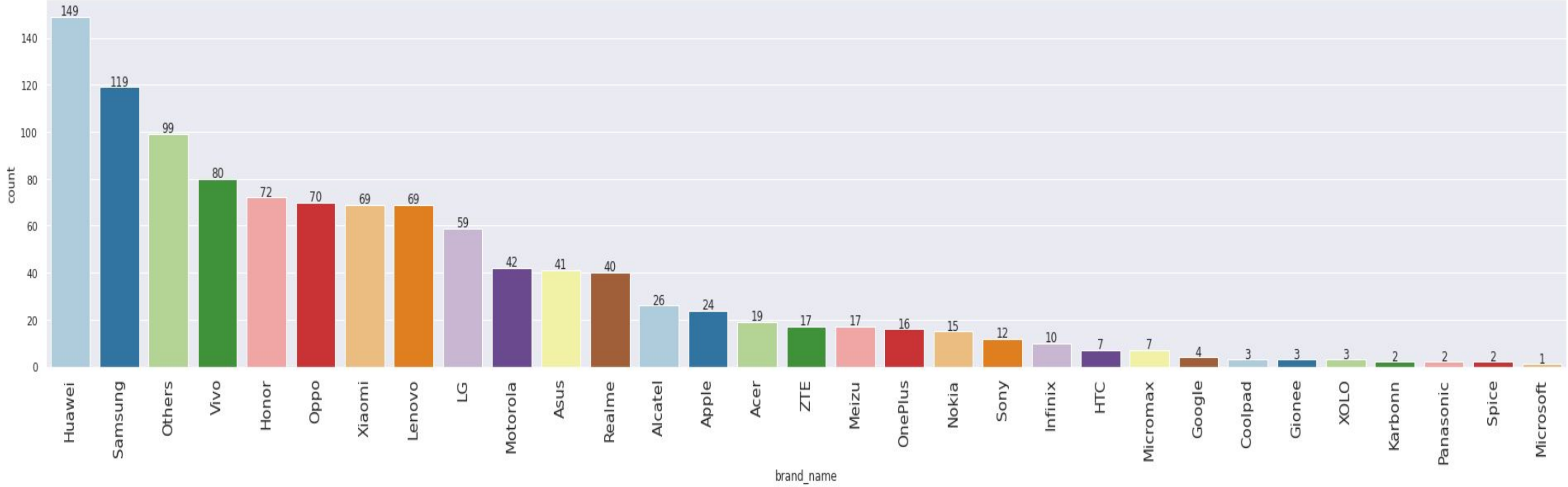
EDA Results continued...

- **Brand_name and weight with battery capacity >4500 in mAh**
  - Among the large battery capacity, “infinix”, “Motorola”,”Realme”, “Vivo”, “ZTE”, “Gionee”, “Micromax”, “Panasonic”, “spice” and “Oppo” brand_name are the best lightweight devices



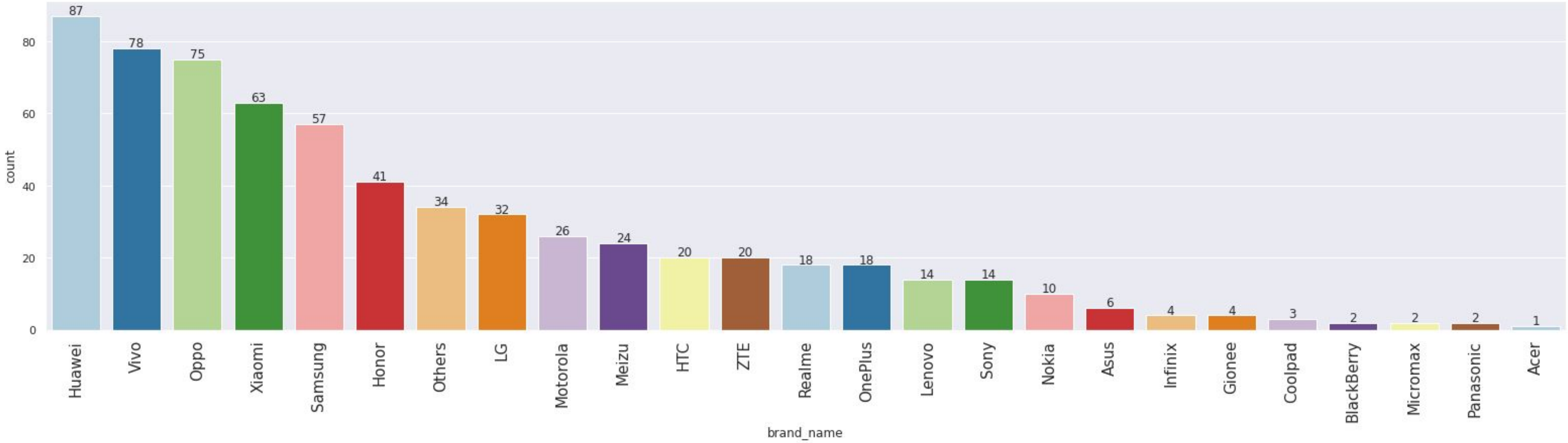
EDA Results continued...

- **Brand_name vs. “screen_size”**
    - A large range of different size screen
    - “Huawei”, “Samsung” brand has large screen devices compare to many others
    - ~500 devices have a screen size between 70- 100 in cms
    - Screen size 60 cm and above seems to be the most popular



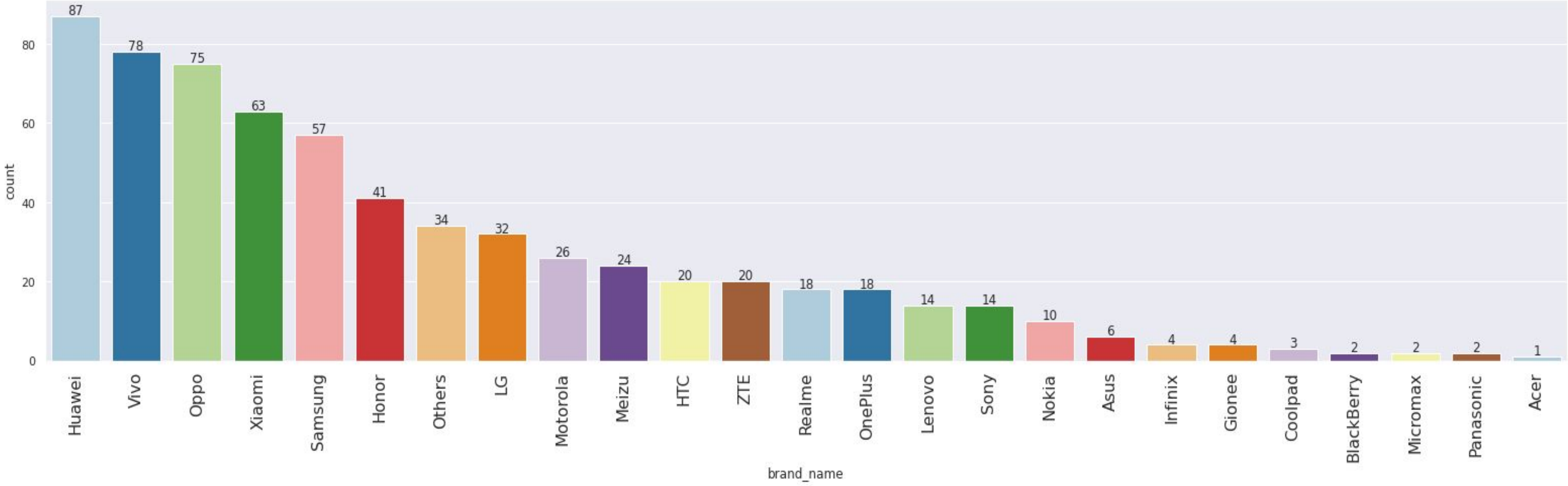
EDA Results continued...

- **Brand_name vs. selfie_camera_mp**
  - “Huawei”, “Vivo”, “Oppo”, “Xiaomi, “ Samsung” are the top 5 brand for the best selfie_camera having >8 megapixel



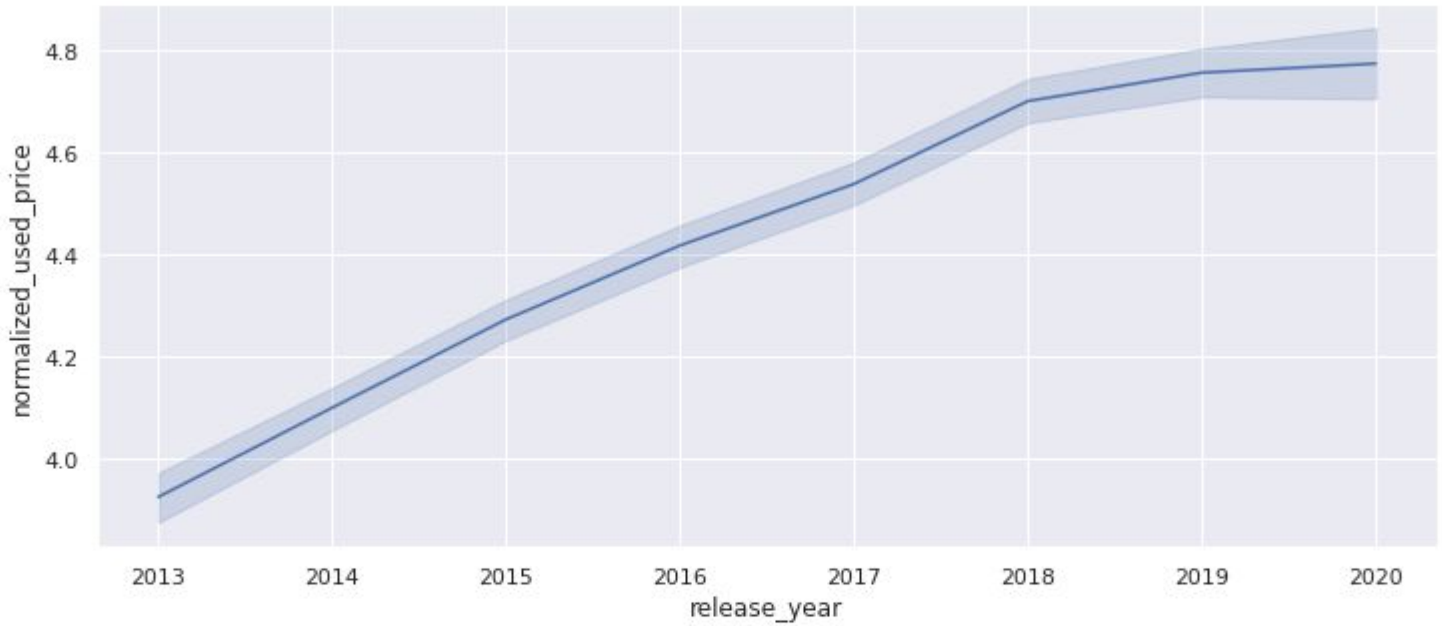
EDA Results continued...

- **Brand_name vs. main_camera_mp**
  - Sony brand has the best camera with >16 Megapixel
  - None of the brand has the best both cameras “main” and “selfie”



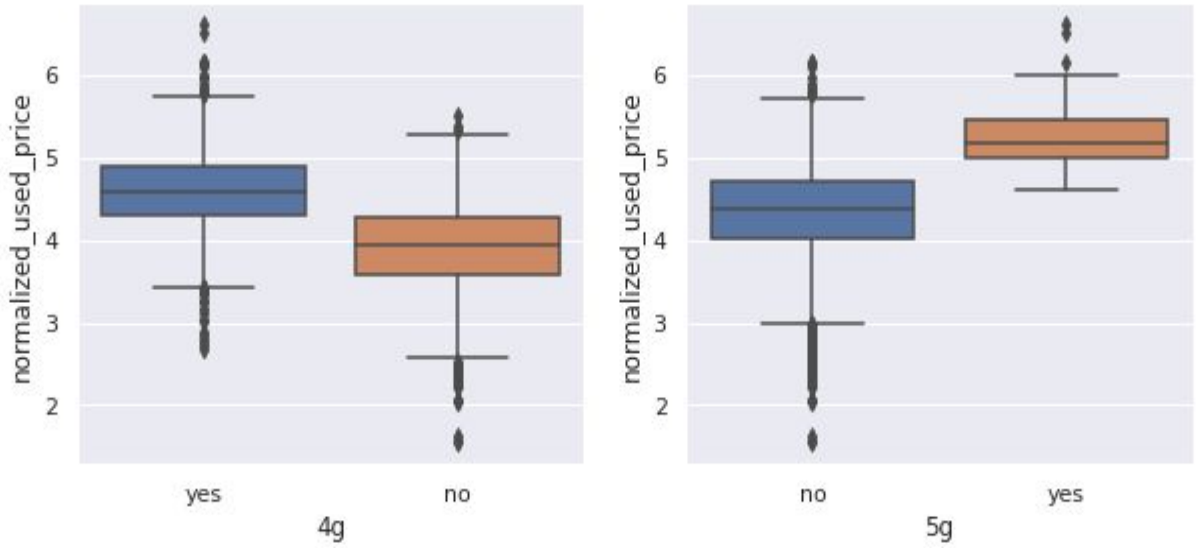
EDA Results continued...

- **Release_year and "normalized_used_price"**
  - The line plot clearly shows older devices have less price.
  - However, there isn't much price difference for the devices from year 2018-2020



EDA Results continued...

- **Normalized_used_price for 4G and 5G network**
  - Avg. price for 4G is 4.6 and 5G is 5.2
  - Price varies more in 4G devices vs. 5G



Data Preprocessing

- **Duplicate value check**
  - There is no Duplicate value in the dataset
- **Missing value treatment**
  - Chose to use median values to impute all the missing values because EDA suggested that the data are highly skewed.
  - First we imputed the missing value in the data columns “main_camera_mp”, “selfie_camera_mp”, “int_memory”, “ram”, “battery”, “weight” by the column medians group by “release_year” and “ brand_name”
  - Then we imputed the left values by their columns medians using group by function for the “brand_name”
  - Last we imputed all the remaining missing values in the “main_camera_mp” columns by its column median.

Data Preprocessing continued...

- **Outlier check (treatment if needed)**
  - Many outliers in the data
  - Chose not to treat them since they all are in number values.
- **Feature engineering**
  - Created a new column “year_since_release” keeping year 2021 baseline and dropped “release_year” from the dataset
- **Data preparation for modeling**
  - “Normalized_used_price” is the dependent variable since we need ML to predict its best value
  - encode “categorical” data
  - Create dummies for the categorical data
  - To build the model on the train set, split the data into train and test sets in 70:30 part.
  - Build the linear regression model using the data from the train set

Data Preprocessing continued...

Linear Regression built from train and test set

```
                            OLS Regression Results
==============================================================================
Dep. Variable:     normalized_used_price   R-squared:                       0.845
Model:                               OLS   Adj. R-squared:                  0.842
Method:                    Least Squares   F-statistic:                     268.7
Date:                   Fri, 11 Nov 2022   Prob (F-statistic):               0.00
Time:                           23:04:11   Log-Likelihood:                 123.85
No. Observations:                   2417   AIC:                            -149.7
Df Residuals:                       2368   BIC:                             134.0
Df Model:                             48
Covariance Type:               nonrobust
==========================================================================================
                            coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------
const                     1.3156      0.071     18.454      0.000       1.176       1.455
screen_size               0.0244      0.003      7.163      0.000       0.018       0.031
main_camera_mp            0.0208      0.002     13.848      0.000       0.018       0.024
selfie_camera_mp          0.0135      0.001     11.997      0.000       0.011       0.016
int_memory                0.0001   6.97e-05      1.651      0.099   -2.16e-05       0.000
ram                       0.0230      0.005      4.451      0.000       0.013       0.033
battery               -1.689e-05   7.27e-06     -2.321      0.020   -3.12e-05   -2.62e-06
weight                    0.0010      0.000      7.480      0.000       0.001       0.001
days_used              4.216e-05   3.09e-05      1.366      0.172   -1.84e-05       0.000
normalized_new_price      0.4311      0.012     35.147      0.000       0.407       0.455
years_since_release      -0.0237      0.005     -5.193      0.000      -0.033      -0.015
brand_name_Alcatel        0.0154      0.048      0.323      0.747      -0.078       0.109
brand_name_Apple         -0.0038      0.147     -0.026      0.980      -0.292       0.285
brand_name_Asus           0.0151      0.048      0.314      0.753      -0.079       0.109
brand_name_BlackBerry    -0.0300      0.070     -0.427      0.669      -0.168       0.108
brand_name_Celkon        -0.0468      0.066     -0.707      0.480      -0.177       0.083
brand_name_Coolpad        0.0209      0.073      0.287      0.774      -0.122       0.164
brand_name_Gionee         0.0448      0.058      0.775      0.438      -0.068       0.158
brand_name_Google        -0.0326      0.085     -0.385      0.700      -0.199       0.133
brand_name_HTC           -0.0130      0.048     -0.270      0.787      -0.108       0.081
brand_name_Honor          0.0317      0.049      0.644      0.520      -0.065       0.128
brand_name_Huawei        -0.0020      0.044     -0.046      0.964      -0.089       0.085
brand_name_Infinix        0.1633      0.093      1.752      0.080      -0.019       0.346
brand_name_Karbonn        0.0943      0.067      1.405      0.160      -0.037       0.226
brand_name_LG            -0.0132      0.045     -0.291      0.771      -0.102       0.076
brand_name_Lava           0.0332      0.062      0.533      0.594      -0.089       0.155
brand_name_Lenovo         0.0454      0.045      1.004      0.316      -0.043       0.134
brand_name_Meizu         -0.0129      0.056     -0.230      0.818      -0.123       0.097
brand_name_Micromax      -0.0337      0.048     -0.704      0.481      -0.128       0.060
brand_name_Microsoft      0.0952      0.088      1.078      0.281      -0.078       0.268
brand_name_Motorola      -0.0112      0.050     -0.226      0.821      -0.109       0.086
brand_name_Nokia          0.0719      0.052      1.387      0.166      -0.030       0.174
brand_name_OnePlus        0.0709      0.077      0.916      0.360      -0.081       0.223
brand_name_Oppo           0.0124      0.048      0.261      0.794      -0.081       0.106
brand_name_Others        -0.0080      0.042     -0.190      0.849      -0.091       0.075
brand_name_Panasonic      0.0563      0.056      1.008      0.314      -0.053       0.166
brand_name_Realme         0.0319      0.062      0.518      0.605      -0.089       0.153
brand_name_Samsung       -0.0313      0.043     -0.725      0.469      -0.116       0.053
brand_name_Sony          -0.0616      0.050     -1.220      0.223      -0.161       0.037
brand_name_Spice         -0.0147      0.063     -0.233      0.816      -0.139       0.109
brand_name_Vivo          -0.0154      0.048     -0.318      0.750      -0.110       0.080
brand_name_XOLO           0.0152      0.055      0.277      0.782      -0.092       0.123
brand_name_Xiaomi         0.0869      0.048      1.806      0.071      -0.007       0.181
brand_name_ZTE           -0.0057      0.047     -0.121      0.904      -0.099       0.087
os_Others                -0.0510      0.033     -1.555      0.120      -0.115       0.013
os_Windows               -0.0207      0.045     -0.459      0.646      -0.109       0.068
os_iOS                   -0.0663      0.146     -0.453      0.651      -0.354       0.221
4g_yes                    0.0528      0.016      3.326      0.001       0.022       0.084
5g_yes                   -0.0714      0.031     -2.268      0.023      -0.133      -0.010
==============================================================================
Omnibus:                      223.612   Durbin-Watson:                   1.910
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              422.275
Skew:                          -0.620   Prob(JB):                     2.01e-92
Kurtosis:                       4.630   Cond. No.                     1.78e+05
==============================================================================
```

Results

- R-squared: 0.845
- Adj. R- squared: 0.842
- Const. Coefficient: 1.315
- Const.coefficient of the “screen_size”:0.024
- Const. Coefficient of the “main_camera_mp”: 0.020

Data Preprocessing continued...

Test Performance

	RMSE	MAE	R-squared	Adj. R-squared	MAPE
0	0.238358	0.184749	0.842479	0.834659	4.501651

Training Performance

	RMSE	MAE	R-squared	Adj. R-squared	MAPE
0	0.229884	0.180326	0.844886	0.841675	4.326841

- The linear regression model is not overfitting because the RMSE>MSE in both train and test sets and are comparable
- MAPE value is 4.5% <5% shows the model forecast for the used_price prediction is very accurate
- R-squared: 0.84, shows the model is not underfitting

Model Assumptions

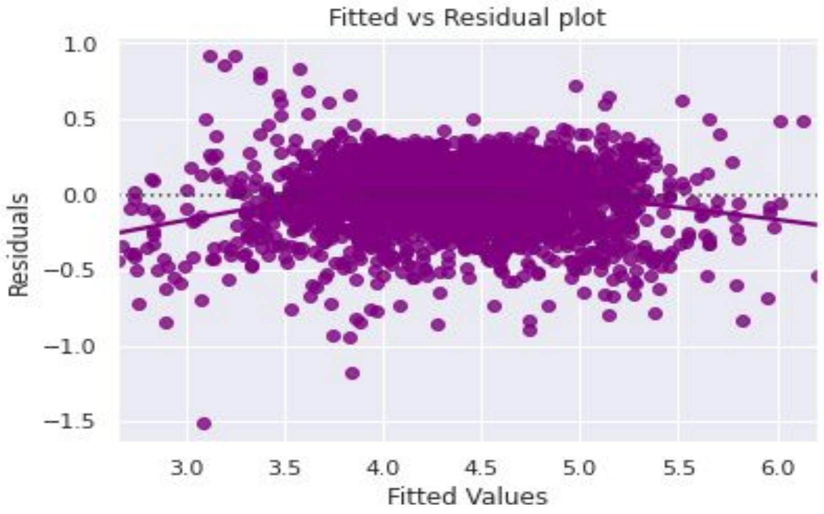
- **Multicollinearity**
  - VIF test to check the variance inflation
    - “Screen_size”, “”weight”, “os_IOS”, “brand_name_samsung”, “brand_name_others”, “brand_name_Apple”, “brand_name_Huawei” variable have more than 5 VIF
    - Dropping these variables one by one, create a model and checking “adj. R-squared” and RMSE everytime
    - Check the VIF values for the remaining variables and repeat the above step
    - After dropping “os_IOS” and “screen_size” for the multicollinearity, VIF for rest all variable showed ~<=5.
    - We ignored the VIF values for dummy variables and constant
    - Adj.R-squared: 0.83, after dropping high VIF variable: . Which has reduced a little.
    - We rebuilt the linear model using the rest of the variables

Model Assumptions continued...

- Dropping high p-value variables
  - Almost all variables that p-value>0.05.
  - Those variable with the p-value >0.05 do not impact the dependent variable - “normalized_used_price”
  - Created a loop, in which
    - we built the model -> check the p-value of variables -> drop the column with the high p-value -> then created a new model without the dropped feature -> check the p-value -> again drop the column with the high p-values till there are no column left with p-value >0.05.
  - The model we create, end of the process of eliminating high p-value variables is our final model.
  - Adj.R-squared of the final model is 0.83, which shows the variables we dropped were not affecting the depend variable and the model.

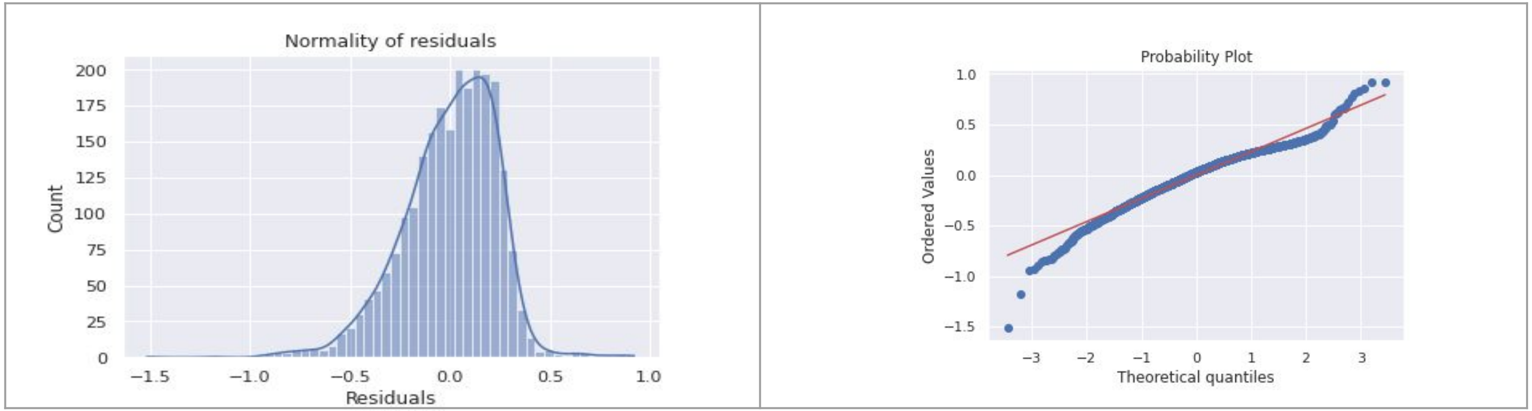
Model Assumptions continued...

- **Test for Linearity and independence of the variable**
  - Making a plot of fitted values vs. residuals
  - There is no pattern in the plot so the assumption of non linearity in data, and the linearity and independence of the residuals are satisfied



Model Assumptions continued...

- **Test for Normality**
  - Q-Q plot to check the distribution of residuals
  - The histplot does show that residuals have a bell shape
  - The residuals almost follow straight line except the tails



  - Shapirio-Wilk test
    - P-value < 0.05,so the assumption that the residuals are not distributed normally
    - But we can assume that residuals are approximately normally distributed

Model Assumptions continued...

- **Test for Homoscedasticity**
  - Goldfeld Quandt test
    - Null hypothesis: Residuals have homoscedasticity
    - Alternate hypothesis : Residuals have heteroscedasticity
    - P-value:0.44>0.05, so we can say residuals are homoscedastic
- **Prediction on the test set**
  - Using the Predict function on the final test set, we took the sample of 10 rows
  - It shows that our model has predicted a very good results comparing actual and predicted values

	Actual	Predicted
1995	4.566741	4.385993
2341	3.696103	4.004162
1913	3.592093	3.645572
688	4.306495	4.101958
650	4.522115	5.193282
2291	4.259294	4.397114
40	4.997685	5.452947
1884	3.875359	4.052736
2538	4.206631	4.036163
45	5.380450	5.225507

Model Performance Summary continued...

Final Model

```
                            OLS Regression Results
==============================================================================
Dep. Variable:     normalized_used_price   R-squared:                       0.839
Model:                               OLS   Adj. R-squared:                  0.838
Method:                    Least Squares   F-statistic:                     963.1
Date:                   Sat, 12 Nov 2022   Prob (F-statistic):               0.00
Time:                           00:52:20   Log-Likelihood:                 78.646
No. Observations:                   2417   AIC:                            -129.3
Df Residuals:                       2403   BIC:                            -48.23
Df Model:                             13
Covariance Type:               nonrobust
=========================================================================================
                            coef    std err          t      P>|t|      [0.025      0.975]
-----------------------------------------------------------------------------------------
const                     1.5185      0.048     31.912      0.000       1.425       1.612
main_camera_mp            0.0212      0.001     14.946      0.000       0.018       0.024
selfie_camera_mp          0.0140      0.001     13.121      0.000       0.012       0.016
ram                       0.0212      0.005      4.259      0.000       0.011       0.031
weight                    0.0017      6e-05     27.586      0.000       0.002       0.002
normalized_new_price      0.4366      0.011     39.843      0.000       0.415       0.458
years_since_release      -0.0288      0.003     -8.496      0.000      -0.035      -0.022
brand_name_Karbonn        0.1142      0.055      2.084      0.037       0.007       0.222
brand_name_Samsung       -0.0342      0.016     -2.082      0.037      -0.066      -0.002
brand_name_Sony          -0.0650      0.030     -2.131      0.033      -0.125      -0.005
brand_name_Xiaomi         0.0808      0.026      3.141      0.002       0.030       0.131
os_Others                -0.1292      0.027     -4.726      0.000      -0.183      -0.076
4g_yes                    0.0489      0.015      3.241      0.001       0.019       0.079
5g_yes                   -0.0645      0.031     -2.104      0.036      -0.125      -0.004
==============================================================================
Omnibus:                      246.471   Durbin-Watson:                   1.907
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              482.249
Skew:                          -0.660   Prob(JB):                    1.91e-105
Kurtosis:                       4.745   Cond. No.                     2.37e+03
==============================================================================
```

opendix slide on model

Model Performance Summary continued..

- Overview of ML model and its parameters
  - Our model performs with 84% accuracy
  - The Model is not suffering over fitting
  - Parameters
    - R-squared: 0.839
    - Adjusted. R squared: 0.838
    - Constant coefficient: 1.51
    - Coefficient of independent variable
      - “Main_camera_mp”: 0.021
      - “Selfie_camera_mp”: 0.014
      - Weight: ”0.0017
      - 4G_yes: 0.048
      - 5G_yes: -0.065
- Summary of most important factors used by the ML model for prediction
  - “Main_camera_mp”, “Selfie_camera_mp”, “Ram”, “Weight”, “Normalized_new_price”, “Years_since_release”, “4g”-”5g” are the most important factors in influencing the predictions of the “normalized_used_price”

Model Performance Summary continued..

- Summary of key performance metrics for training and test data in tabular format for comparison

Test Performance

	RMSE	MAE	R-squared	Adj. R-squared	MAPE
0	0.241654	0.186761	0.838093	0.835875	4.559346

Training Performance

	RMSE	MAE	R-squared	Adj. R-squared	MAPE
0	0.234224	0.1831	0.838974	0.838035	4.404288

- The final model's RMSE and MAE values have not changed since our linear regression even after dropping so many variables
- RMSE>MAE in both the test and train data and are very much comparable
- RMSE: 0.23 (the best range 0.2-0.5)shows that our model predicts highly accurate "nomalized_used_price
- MAPE: 4.55, suggests that the prediction of the price is falling within 4.5%
- Hence, we can say that our model is the best fit for predicting the "normalized_used_price" for all devices data that is feed into it as well as for the inference