Recell Pricing Strategy Analysis

PGP-DSBA _ Recell Project

October 26, 2023

Contents / Agenda

Executive Summary (1/2)

- The model is able to explain ~84% of the variation in the data and within 4.56% of the normalized price of used devices on the test data; we can, thus, conclude that the model is good for prediction and inference

- All other variables held constant, the increase of the normalized price of a new device by one unit results in an increase of the normalized price of an equivalent old device by 0.4415 units

- The increase of the main camera resolution by one megapixel results in an increase of the normalized price of the corresponding old device by 0.0210 units, all other variables held constant

- The normalized price of an old Xiaomi device, on average, will be 0.0801 units higher than that of an old device of any of the brands not included in the final model, all other variables held constant

- Strangely, on average, the normalized price of an old iOS device would be 0.09 units less than that of on an old Android or Window device, all other variables held constant

- The normalized price of an old 4g-enabled device is 0.0502 units higher than that of an old device that is not 4g-enabled, all other variables held constant

Executive Summary (2/2)

- Another surprising observation: The normalized price of an old 5g-enabled device is 0.0673 units less than that of a non 5g-enable old device, all other variables held constant
- Considering the relatively small numbers of devices with brands such as Xiaomi, iOS devices, and 5g-enabled devices, Recell might want to update the model some months or years later when a larger amount of the devices will have been sold and, thus, determine whether there has been any significant change in the model
- Other parameters such as the network operators and geographical location where the sales were carried out might help improve the model and better predict suitable prices for old and refurbished devices
- In the meantime, the startup can use the present model of price fixing within a 95% confidence interval
- Considering the high correlation between the normalized prices of old and new devices and the significance of the price influence of new devices on the corresponding prices of old devices in the model, the startup might find it profitable to concentrate its sale on refurbished high-end (expensive) devices

Business Problem Overview

- Increased demand for cheap and affordable telecommunications gadgets has boosted the market for used and refurbished devices as underscored by the Compound Annual Growth Rate (CAGR) of 13.6% from 2018 to 2023 projected by the IDC (International Data Corporation)
- The startup, Recell, seeks to make the most of this growing market and my services as Data Scientist have been requested to aid the company model an optimal pricing policy
- My task consist of using machine learning to develop a dynamic pricing strategy for used and refurbished devices, building a linear regression prediction model that clearly identifies and assesses the impact of principal factors on the market prices of these devices and using data collected in 2021 about the sale of devices released between 2013 and 2020

Solution Approach

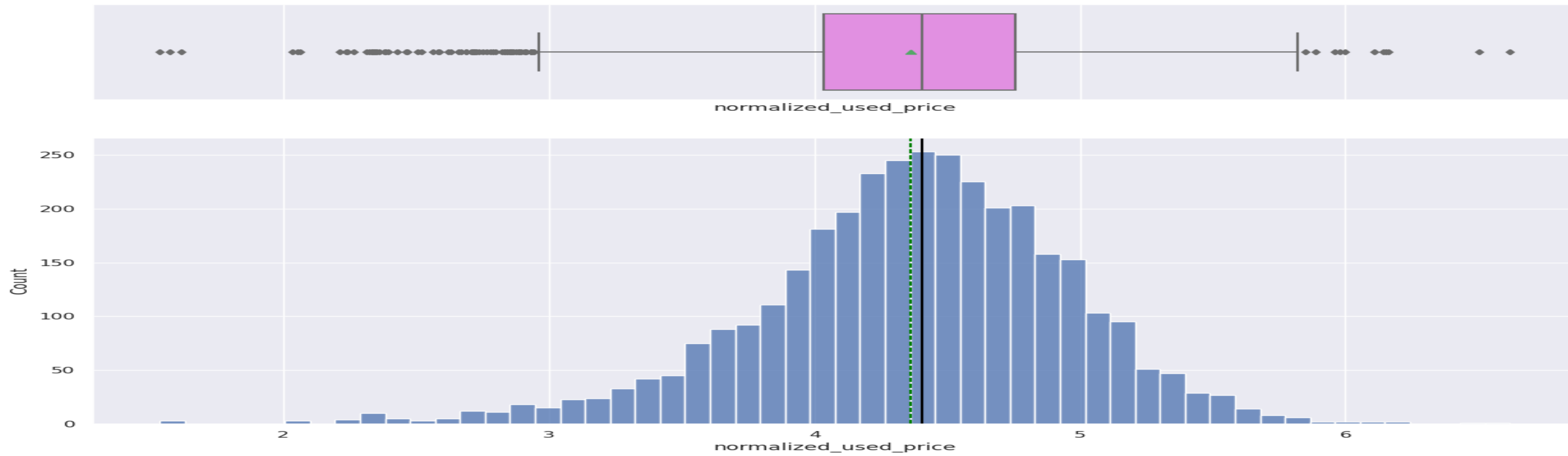
- The method used for this analysis consist of the following steps:

    1. Exploratory Data Analysis of the data collected to identify trends and characteristics of the recorded attribute

    2. Data Preprocessing to identify and treat duplicate entries, missing values, and outliers, and to engineer and prepare identified features necessary for the target model

    3. Design of the model (Ordinary Least Squares - OLS) and Analysis of its Performance

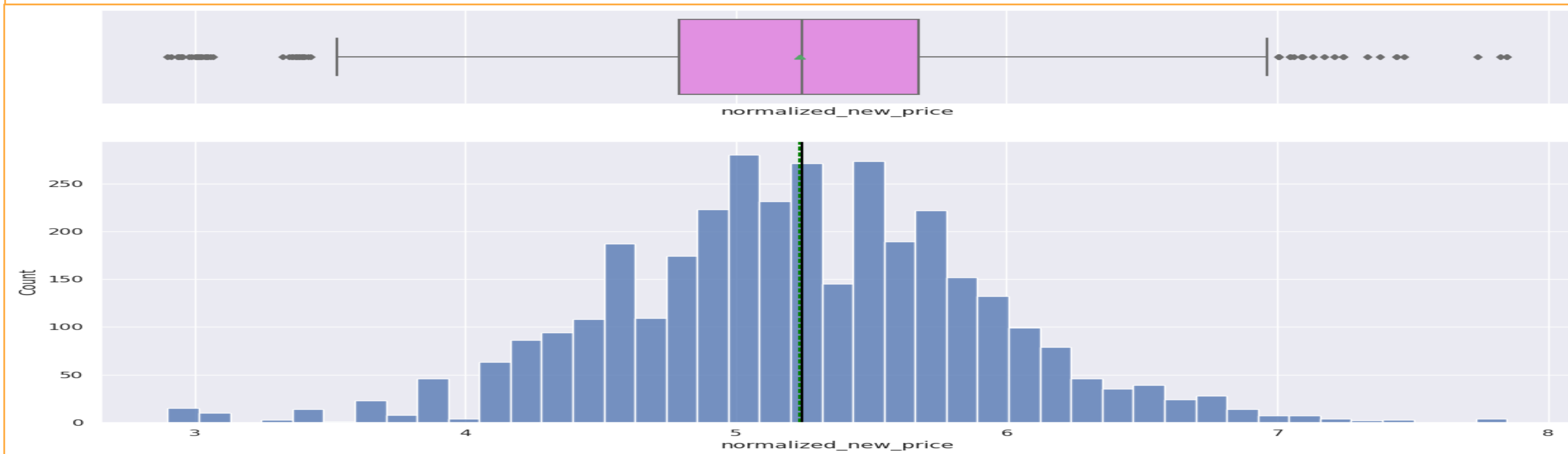
EDA Results: Univariate Analysis: Normalized Price of Used Devices



- The normalized price of used devices is approximately normal with outliers on both sides of the distribution
- The median value is slightly less than 4.5

Link to Appendix slide on data background check

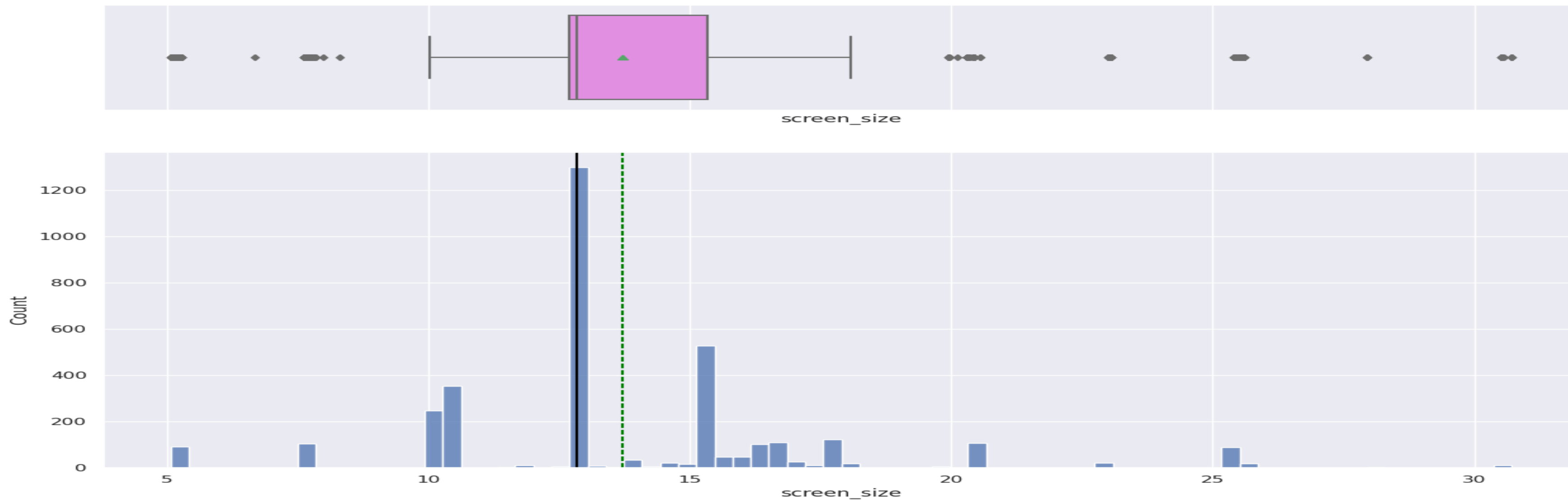
EDA Results: Univariate Analysis: Normalized Price of New Devices



- The normalized price of new devices is approximately normal with outliers on both sides of the distribution
- The median and mean values are approximately 5.2

Link to Appendix slide on data background check

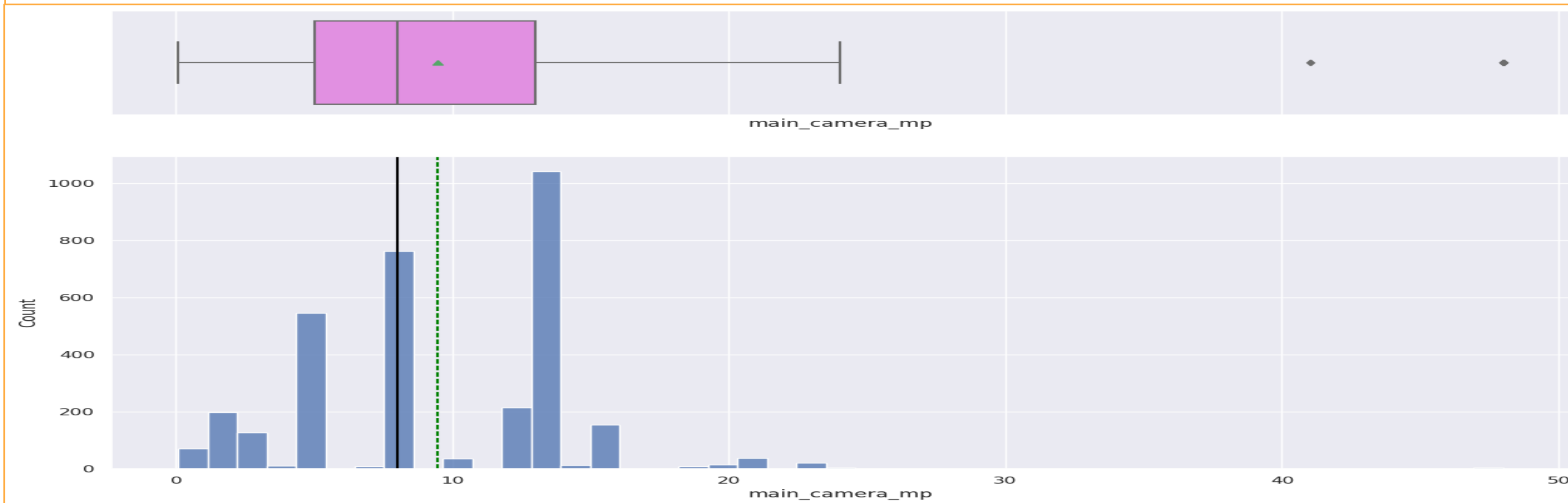
EDA Results: Univariate Analysis: Screen Size



- The distribution of the screen size is sparse and right-skewed, and contains outliers on both sides of the distribution
- The average screen size is about 14 cm while the median value is about 13 cm

[*Link to Appendix slide on data background check*]

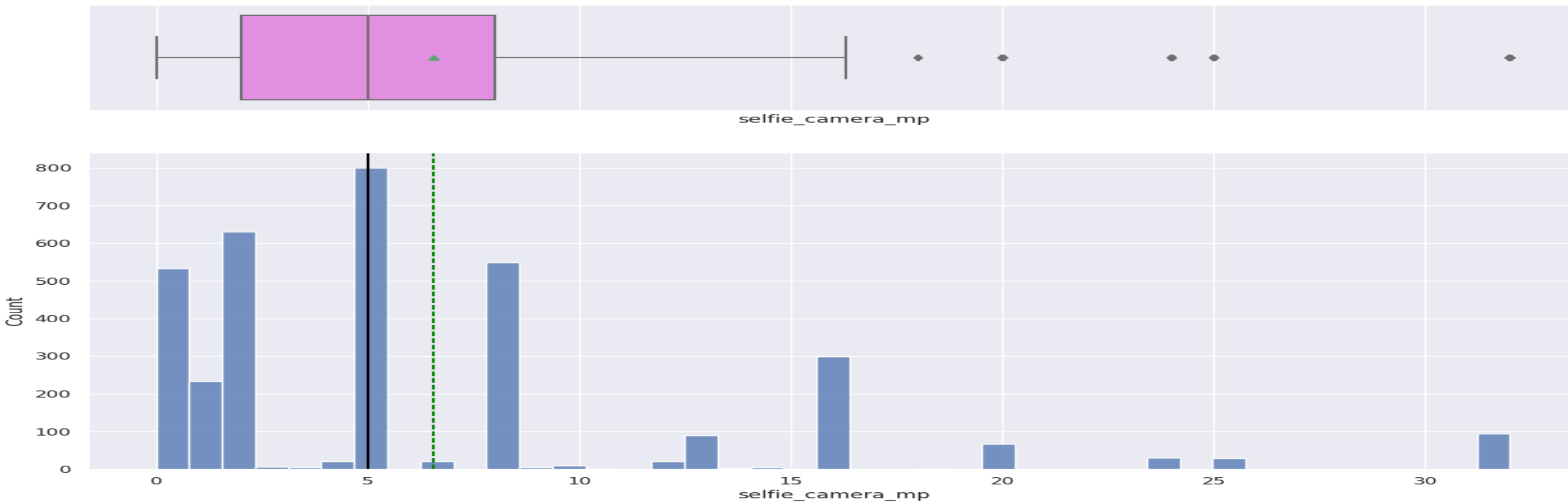
EDA Results: Univariate Analysis: Main Camera Resolution



- The distribution of the main camera resolution is slightly right-skewed and multimodal
- The average value is a little less than 10 megapixels whereas the median is about 8 megapixels

[Link to Appendix slide on data background check]

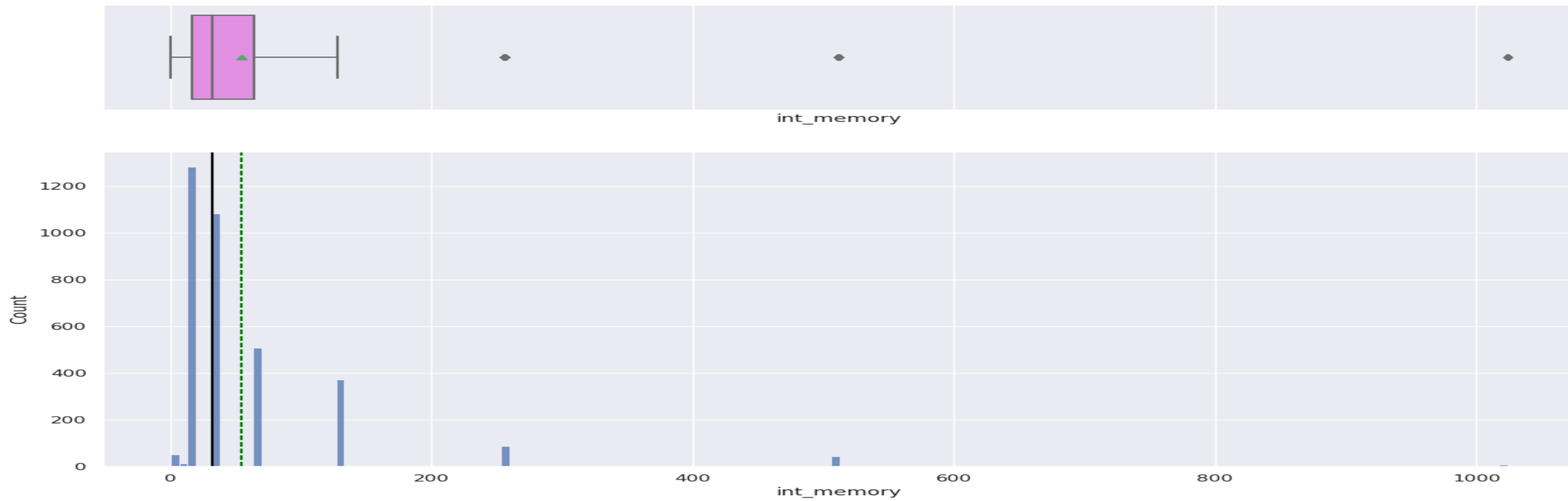
EDA Results: Univariate Analysis: Selfie Camera Resolution



- The distribution of the selfie camera resolution is slightly right-skewed and contains upper outliers
- The median is 5 megapixels meanwhile the average value is about 7 megapixels

[Link to Appendix slide on data background check]()

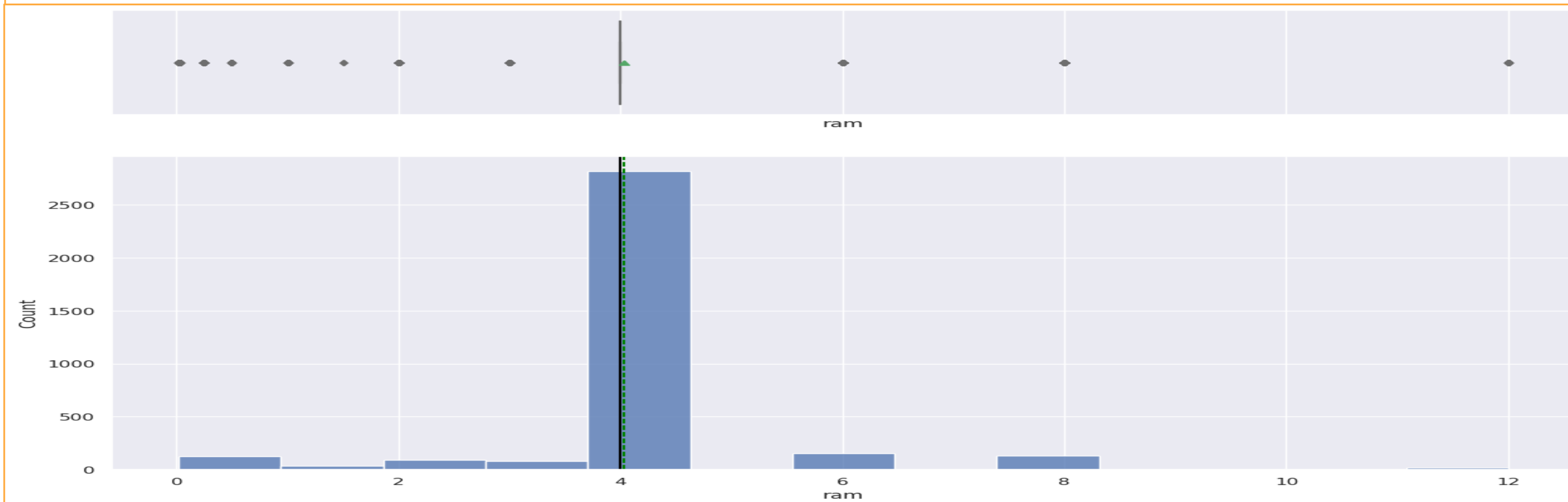
EDA Results: Univariate Analysis: Internal Memory



- The distribution of the internal memory is right-skewed and contains very large upper outliers
- Both the median and mean are below 100 GB

Link to Appendix slide on data background check

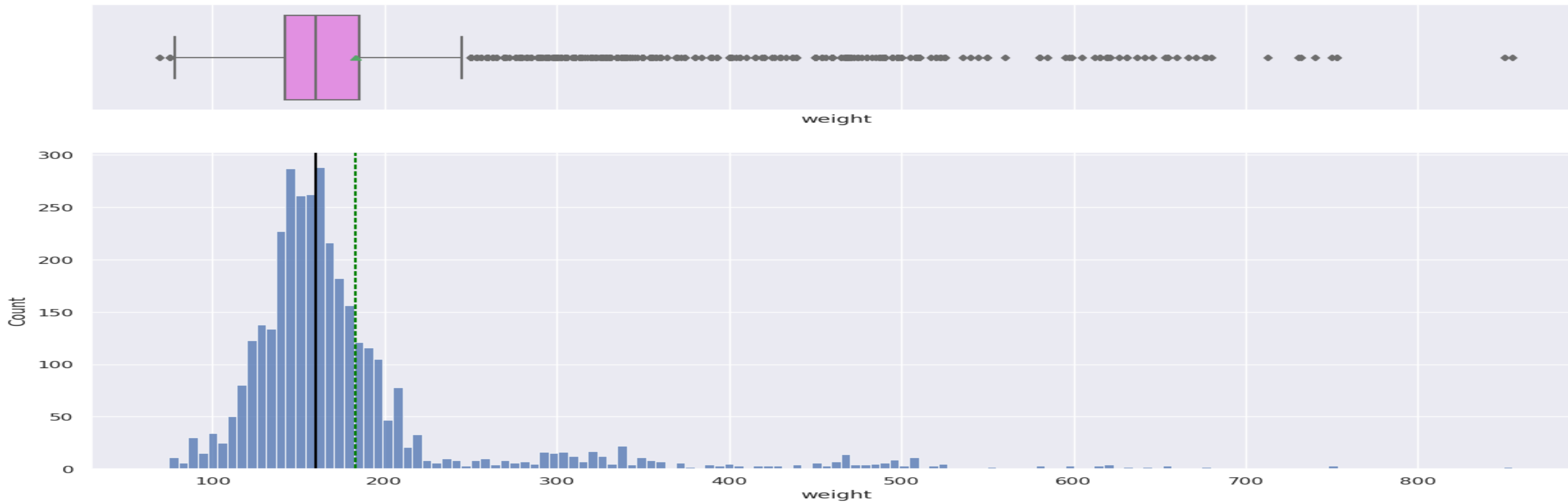
EDA Results: Univariate Analysis: Random Access Memory



- The ram distribution has outliers on both sides of the distribution, particularly large on the right
- Most of the devices have ram sizes that are very close to 4 GB and there is little or no variation in the distribution

Link to Appendix slide on data background check

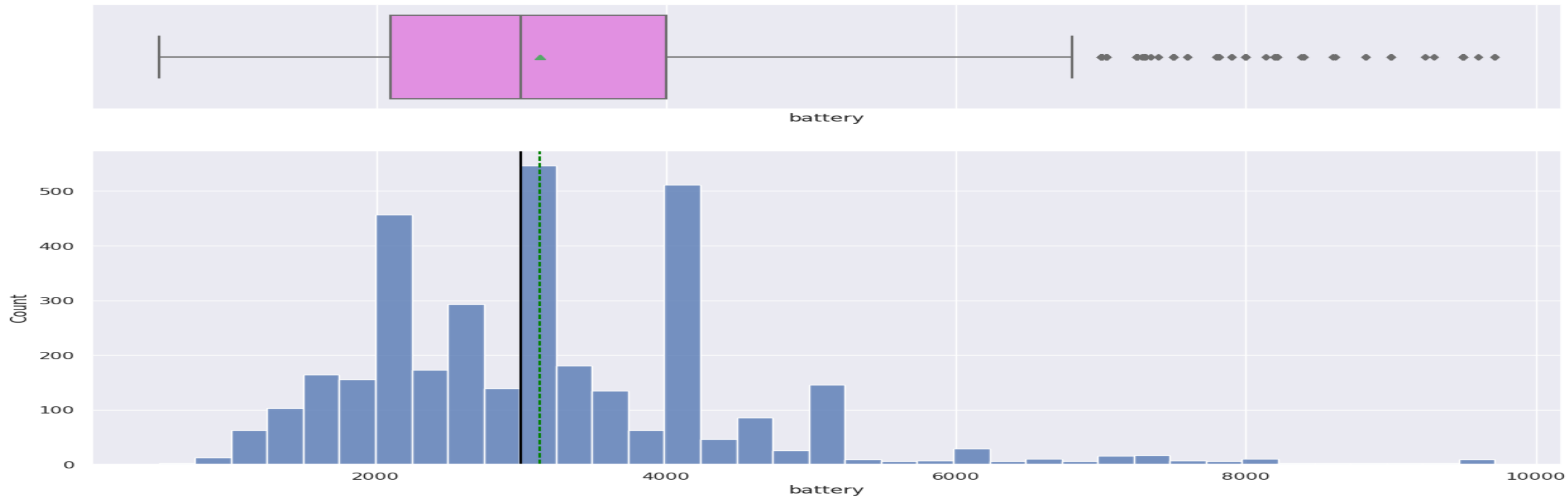
EDA Results: Univariate Analysis: Weight



- The distribution of device weights is approximately symmetrical for most of the data but has several and very large upper outliers
- The median is about 60 grams, approximately 20 grams less than the average

[Link to Appendix slide on data background check]

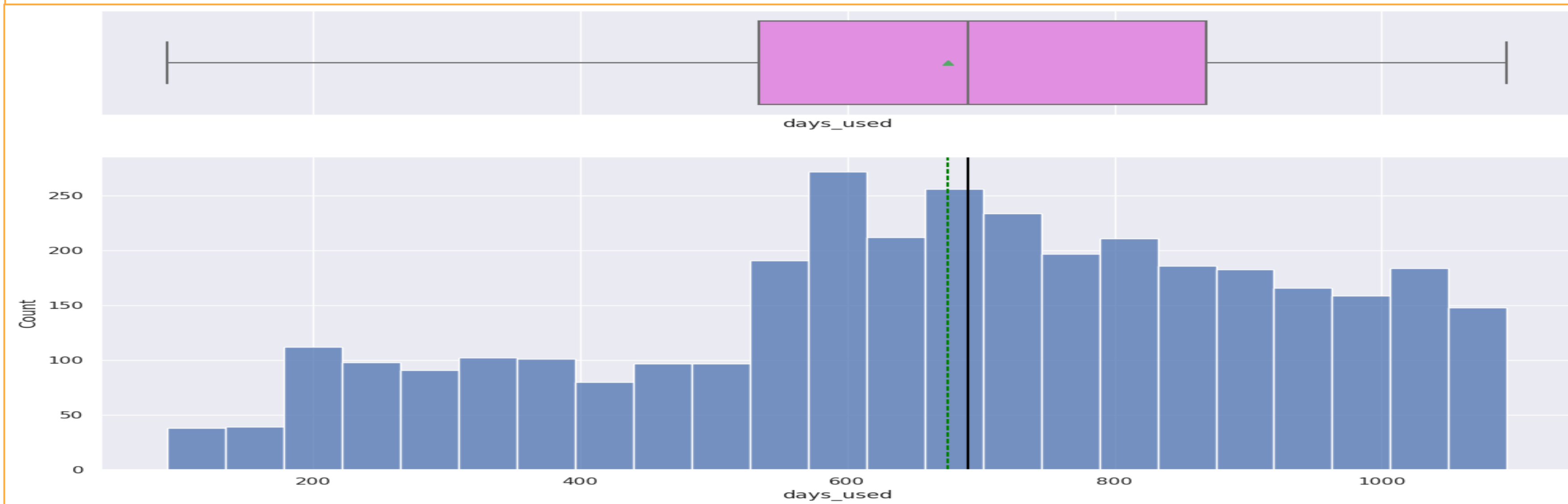
EDA Results: Univariate Analysis: Battery



- The battery distribution is also approximately symmetrical for the most part but contains several upper outliers
- The median, slightly lower than the average, is about 3000 mAh

Link to Appendix slide on data background check

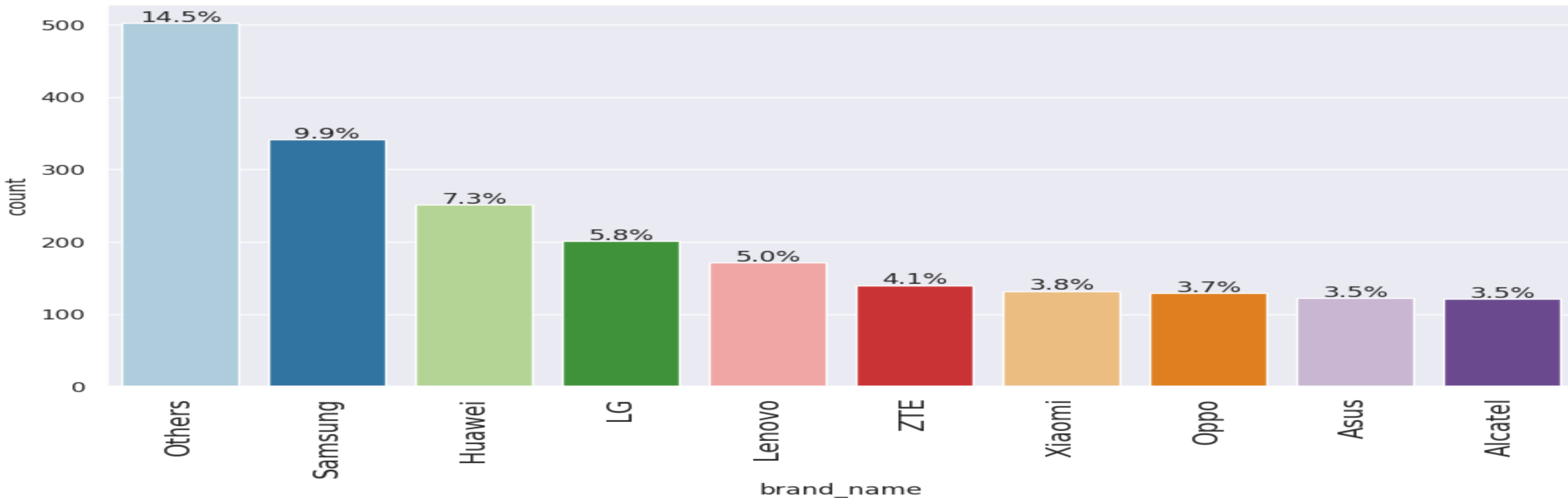
EDA Results: Univariate Analysis: Days Used



- The number of days used is slightly left-skewed
- The median, slightly larger than the mean, is about 700 days

Link to Appendix slide on data background check

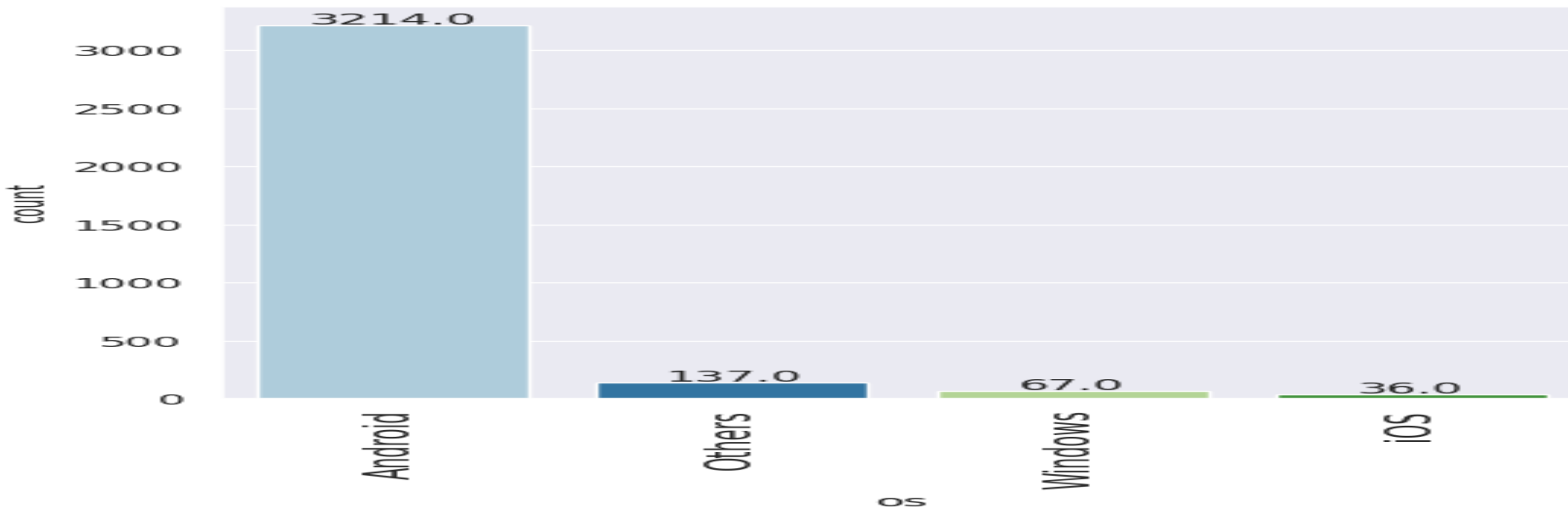
EDA Results: Univariate Analysis: Brand Name



- Samsung is the most used brand (9.9%) followed by Huawei (7.3%)
- Alcatel and Asus are the least used brands (3.5% each)

[Link to Appendix slide on data background check]

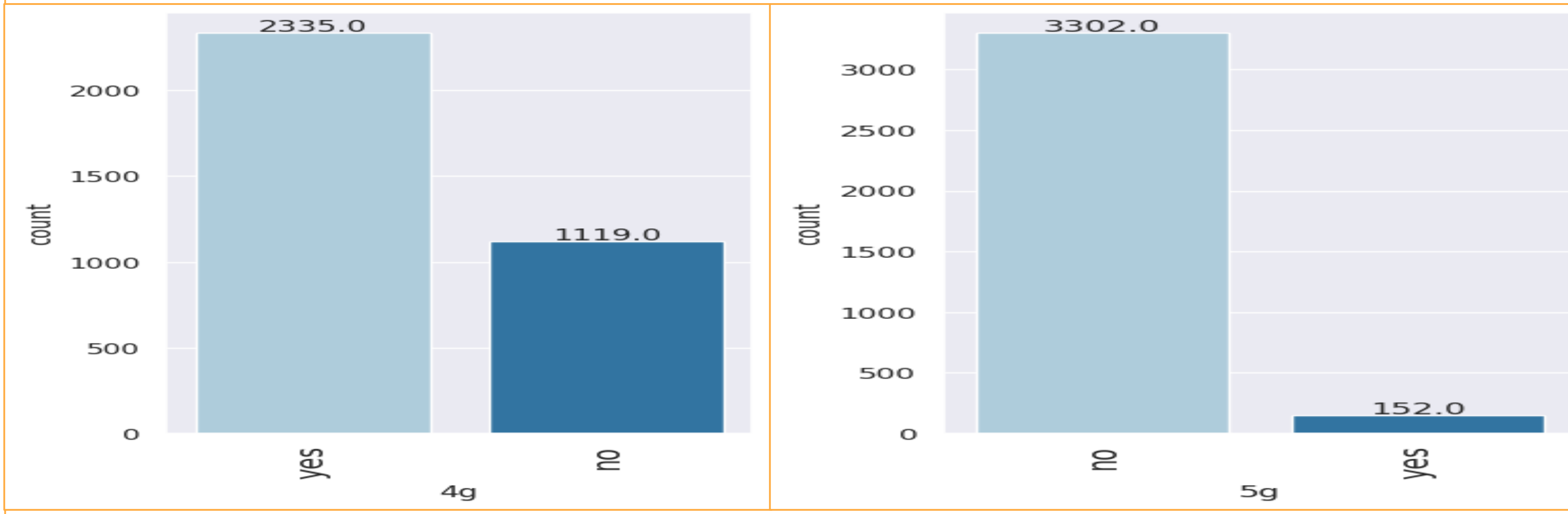
EDA Results: Univariate Analysis: Operating System



- Android is by far the most used operating system (3214)
- The other known operating systems are Windows (67) and iOS (36)

Link to Appendix slide on data background check

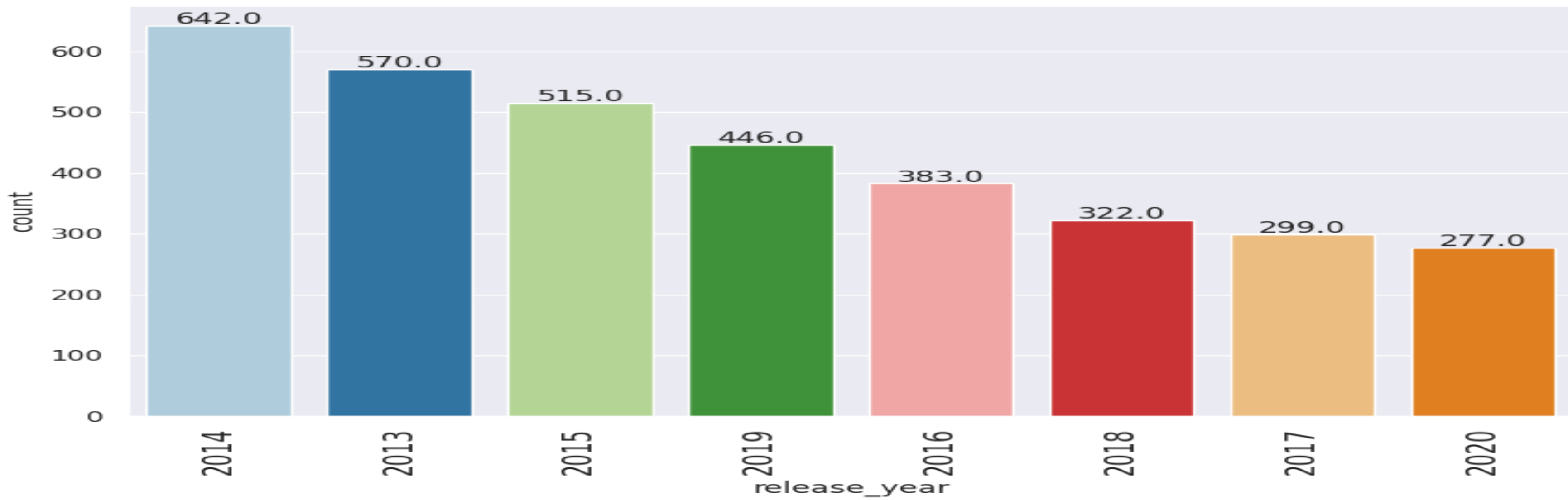
EDA Results: Univariate Analysis: 4g and 5g technologies



- We have visual confirmation that most of the devices are 4g-compatible (2335 against 1119 non-compatible devices)
- Almost all the devices are not 5g compatible (3302 vs 152 5g compatible devices)

Link to Appendix slide on data background check

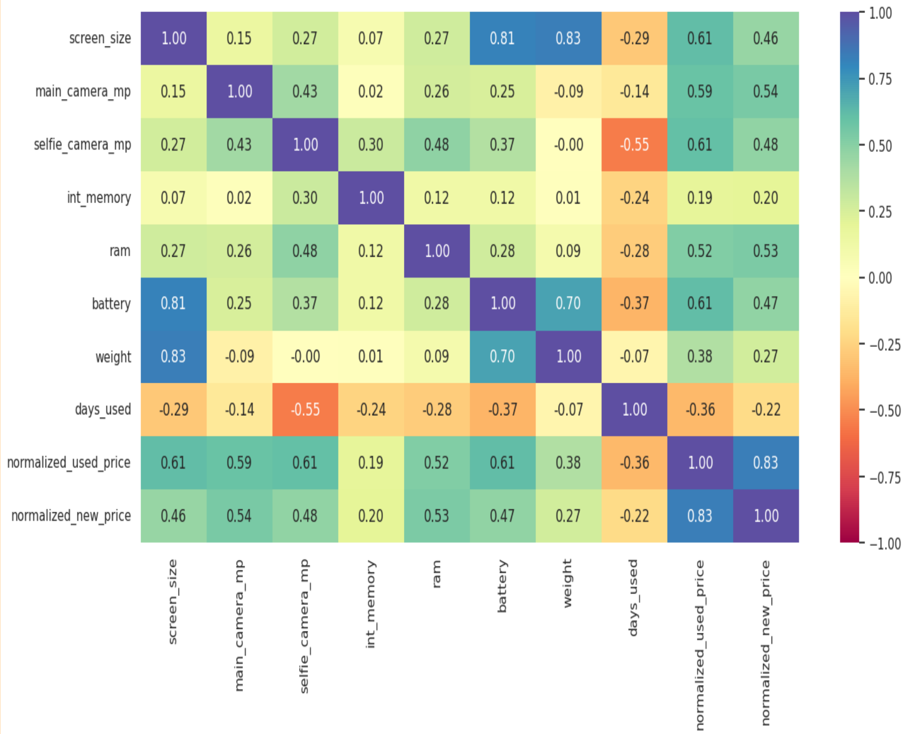
EDA Results: Univariate Analysis: Release Year



- The number of devices per year shows an irregular pattern across the years but has generally decreased from 2013 (570) to 2020 (277)

[Link to Appendix slide on data background check](#)

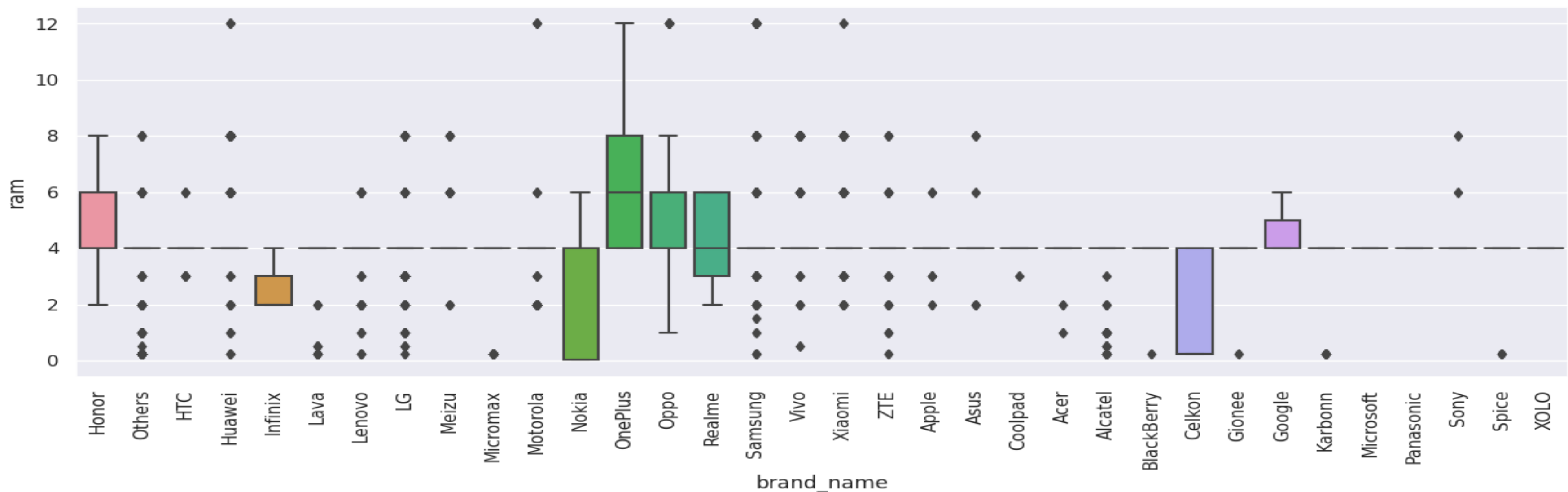
EDA Results: Bivariate Analysis: Correlation



- The normalized price of used devices shows moderate positive correlation with the screen size, the main and selfie camera resolutions, the ram, and the battery; and strong positive correlation with the equivalent price of new devices
- The screen size shows strong positive correlation with the weight and battery and moderate positive correlation with the normalized price of used devices
- The normalized prices of new devices also show moderate positive correlation with the main camera resolution and the ram
- The weight and battery show strong positive correlation
- The number of days used has a moderate negative correlation with the resolution of the selfie camera

Link to Appendix slide on data background check

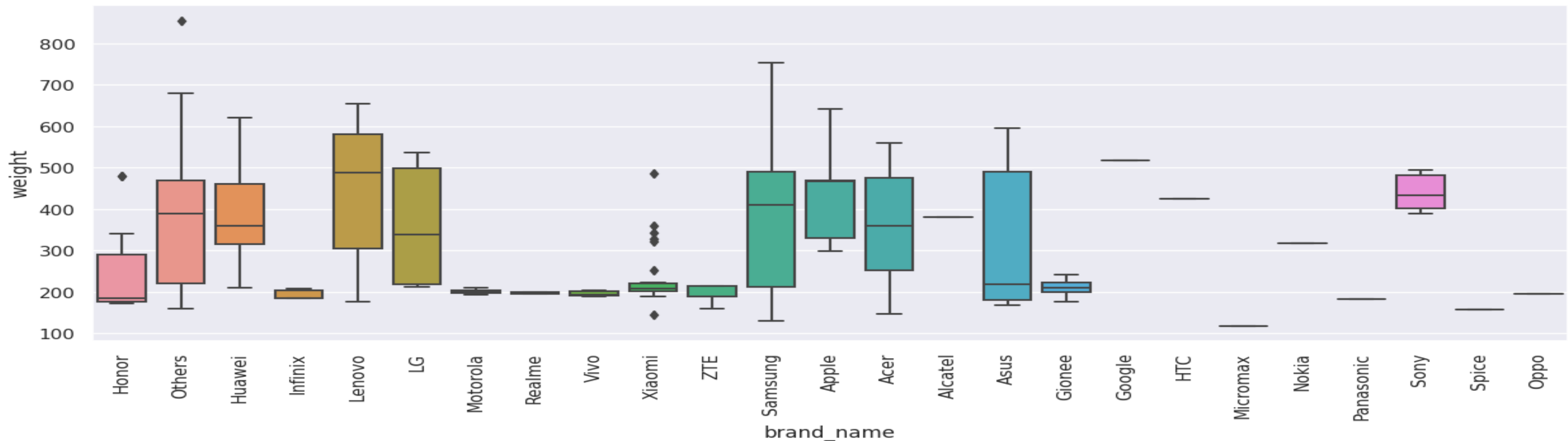
EDA Results: Bivariate Analysis : RAM vs Brand Name



- OnePlus tends to have the largest ram sizes while the least ram sizes are generally registered for Nokia and Celkon
- Most other brands essentially have ram sizes of 4 GB and little or no variation in their ram sizes

Link to Appendix slide on data background check

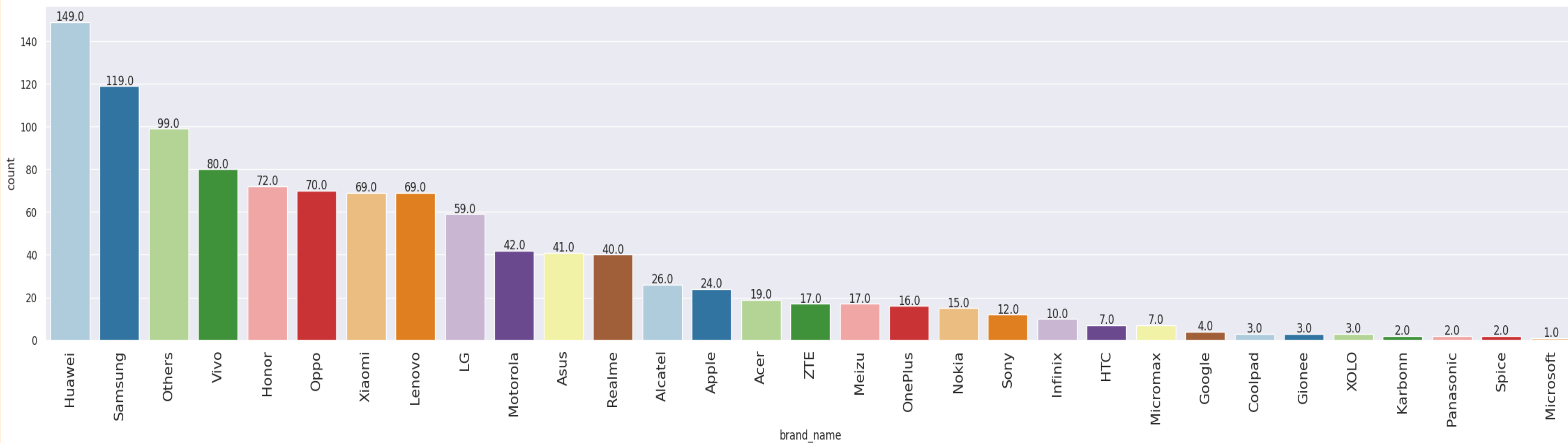
EDA Results: Bivariate Analysis : Weight vs Brand Name for Devices with large batteries (Greater than 4500 mAh)



- 341 devices have battery sizes greater than 4500 mAh
- Among these devices, Google devices tend to be the heaviest while Micromax tend to be the lightest and both brands have little or no variation in weight
- Among these large-battery devices, Lenovo, LG, Samsung, and Asus, show relatively large variation in weight

Link to Appendix slide on data background check

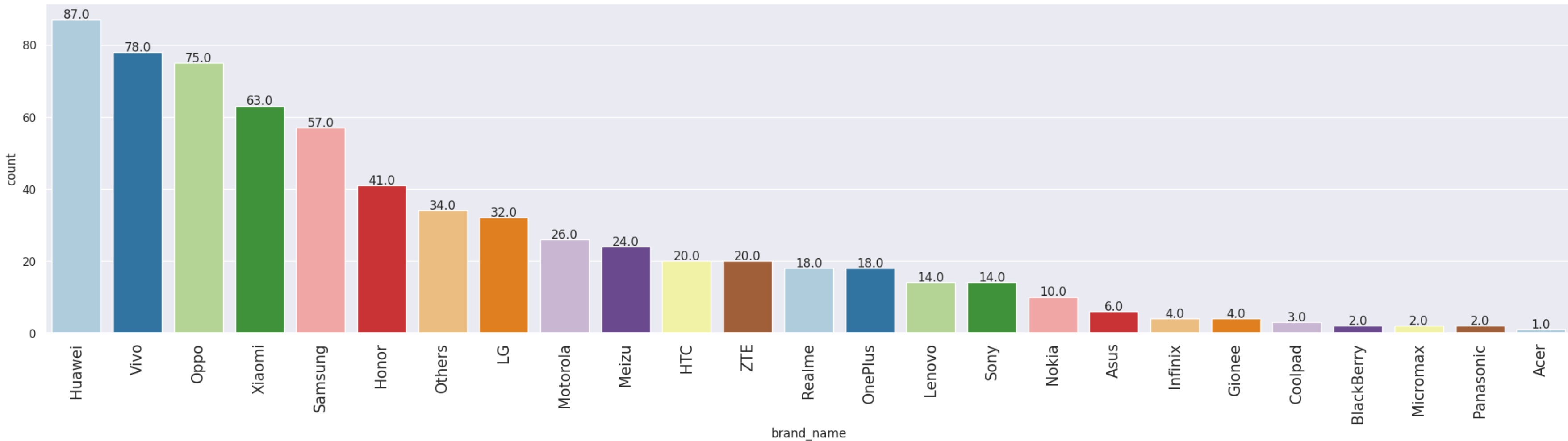
EDA Results: Bivariate Analysis : Brand Name for large-screen (greater than 15.24 cm) Devices



- The screen sizes of 1099 devices are larger than 15.24 cm
- Among these devices, most are Huawei devices (149) followed by Samsung (119)
- Only one is a Microsoft device

Link to Appendix slide on data background check

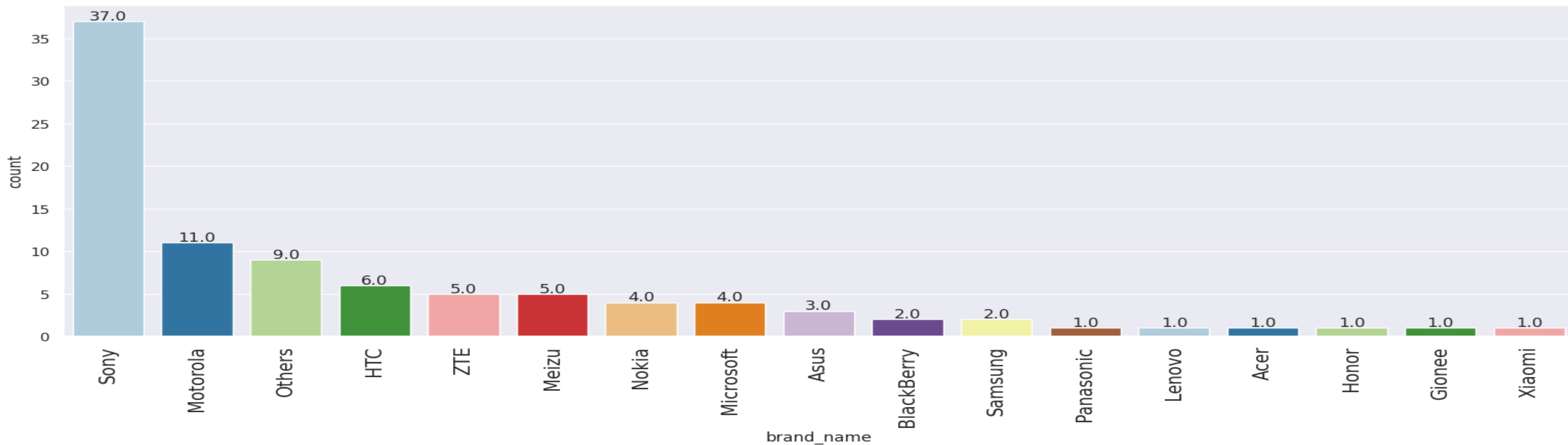
EDA Results: Bivariate Analysis : Brand Name for high-resolution (greater than 8 megapixels) front camera Devices



- 655 devices have selfie camera resolution greater than 8 megapixels
- Huawei once again tops the list of brands, in this case, of number of devices having a selfie camera resolution greater than 8 megapixels (87), followed by Vivo (78)
- Only one of these devices is an Acer

Link to Appendix slide on data background check

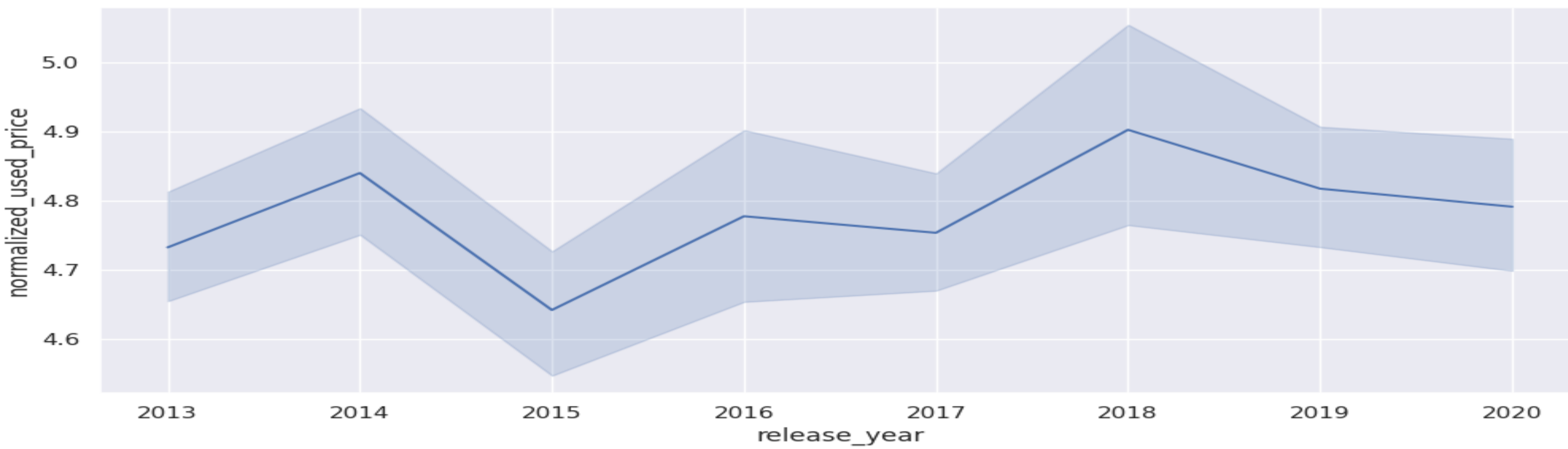
EDA Results: Bivariate Analysis : Brand Name for high-resolution (greater than 16 megapixels) rear camera Devices



- 94 devices have main camera resolutions greater than 16 megapixels
- Now, Sony tops the list of brands for devices having a main camera resolution greater than 16 megapixels (37) and is followed from afar by Motorola (11)
- Among these devices, Panasonic, Lenovo, Acer, Honor, Gionee, and Xiaomi are represented, each, by only one device

Link to Appendix slide on data background check

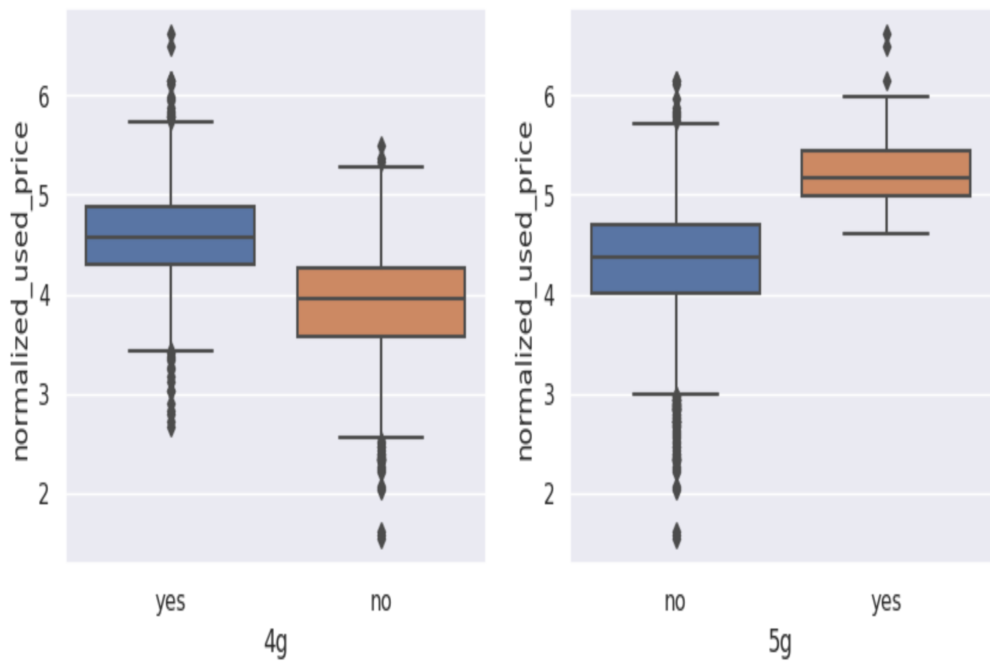
EDA Results: Bivariate Analysis : Normalized Price of Used Devices vs Year of Release



- The normalized price of used devices has an undulating yet generally increasing trend from 2013 to 2020

[Link to Appendix slide on data background check]()

EDA Results: Bivariate Analysis : Normalized Price of Used Devices vs 4g- and 5g-compatibility



- 4g: The normalized prices of used devices is approximately 0.6 normalized price greater for 4g-enabled devices; the variation is slightly larger for non 4g-enabled devices; and the outliers are registered on both sides of each of the two distributions
- 5g: The normalized prices of used devices is generally almost 1 normalized price greater for 5g-enabled devices; the variation is slightly greater for non 5g-enabled devices; outliers are registered on both sides of non 5g-enabled devices and on the right of 5g-enabled devices

[Link to Appendix slide on data background check]

Data Preprocessing: Duplicate value check and Missing value treatment

0: Null Untreated

```
brand_name                 0
os                         0
screen_size                0
4g                         0
5g                         0
main_camera_mp           179
selfie_camera_mp           2
int_memory                 4
ram                        4
battery                    6
weight                     7
release_year               0
days_used                  0
normalized_used_price      0
normalized_new_price       0
dtype: int64
```

1: Imputation of Median Grouped by Release year and Brand Name

```
brand_name                 0
os                         0
screen_size                0
4g                         0
5g                         0
main_camera_mp           179
selfie_camera_mp           2
int_memory                 0
ram                        0
battery                    6
weight                     7
release_year               0
days_used                  0
normalized_used_price      0
normalized_new_price       0
dtype: int64
```

2: Imputation of Median Grouped by Brand Name

```
brand_name                 0
os                         0
screen_size                0
4g                         0
5g                         0
main_camera_mp            10
selfie_camera_mp           0
int_memory                 0
ram                        0
battery                    0
weight                     0
release_year               0
days_used                  0
normalized_used_price      0
normalized_new_price       0
dtype: int64
```

3: Imputation of the Median of the Main Camera Resolution

```
brand_name                 0
os                         0
screen_size                0
4g                         0
5g                         0
main_camera_mp             0
selfie_camera_mp           0
int_memory                 0
ram                        0
battery                    0
weight                     0
release_year               0
days_used                  0
normalized_used_price      0
normalized_new_price       0
dtype: int64
```

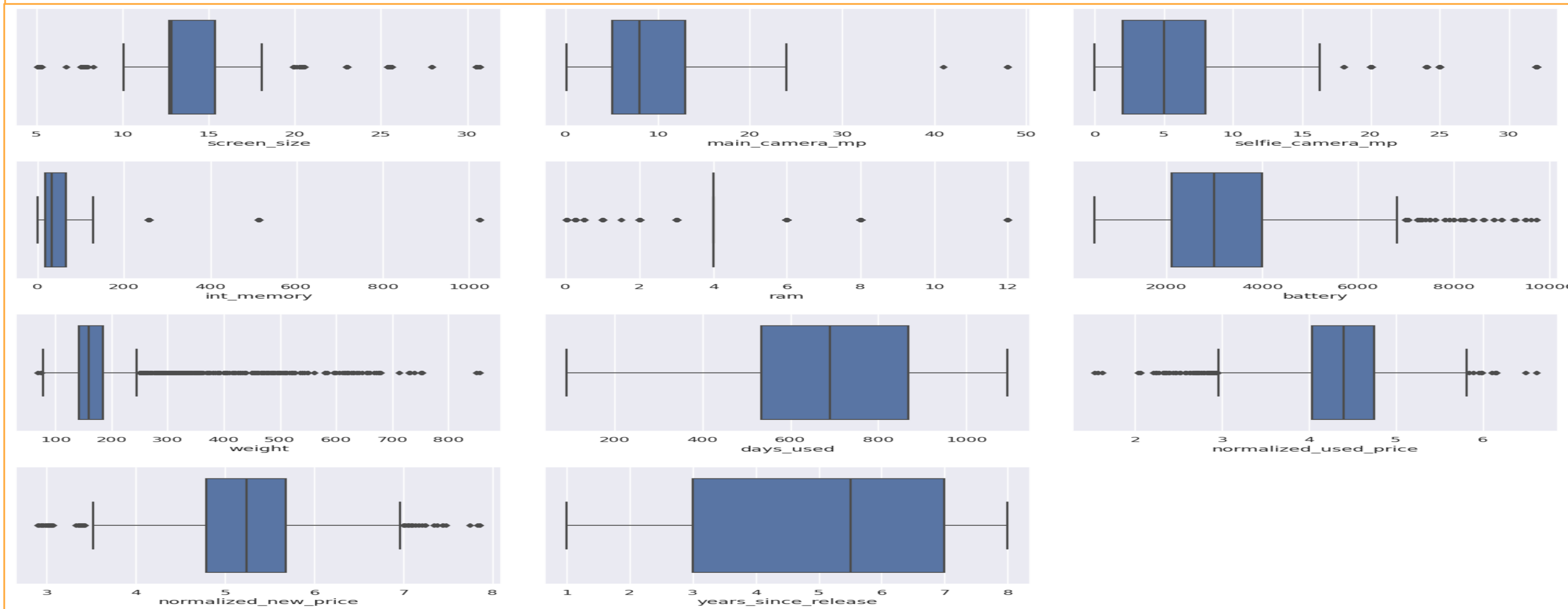
- The data contains no duplicates
- main_camera_mp, selfie_camera_mp, int_memory, ram, battery, and weight (6 columns) all contain missing values

1. After first imputation of medians grouped by year of release and brand name (assuming that the prices for a given brand in a specific year are close to one another), we have 4 columns with missing values: main_camera_mp, selfie_camera_mp, battery, and weight

2. A second imputation of medians grouped by brand name results in one column still having missing values: main_camera_mp

3. A final imputation of the median of main_camera_mp imputed to the missing values of this column eradicates the remaining missing values

Data Preprocessing: Outlier check



- All the numerical variables have outliers except days_used and years_since_release, but we have no reason to consider any of the values extraneous

Data Preprocessing: Feature Engineering and Data Preparation for Modeling

```
count    3454.000000
mean        5.034742
std         2.298455
min         1.000000
25%         3.000000
50%         5.500000
75%         7.000000
max         8.000000
Name: years_since_release, dtype: float64
```

```
  brand_name       os  screen_size   4g   5g  main_camera_mp  \
0      Honor  Android        14.50  yes   no            13.0
1      Honor  Android        17.30  yes  yes            13.0
2      Honor  Android        16.69  yes  yes            13.0
3      Honor  Android        25.50  yes  yes            13.0
4      Honor  Android        15.32  yes   no            13.0

   selfie_camera_mp  int_memory  ram  battery  weight  days_used  \
0               5.0        64.0  3.0   3020.0   146.0        127
1              16.0       128.0  8.0   4300.0   213.0        325
2               8.0       128.0  8.0   4200.0   213.0        162
3               8.0        64.0  6.0   7250.0   480.0        345
4               8.0        64.0  3.0   5000.0   185.0        293

   normalized_new_price  years_since_release
0              4.715100                    1
1              5.519018                    1
2              5.884631                    1
3              5.630961                    1
4              4.947837                    1

0    4.307572
1    5.162097
2    5.111084
3    5.135387
4    4.389995
Name: normalized_used_price, dtype: float64
```

	const	screen_size	main_camera_mp	selfie_camera_mp	int_memory	ram	battery	weight	days_used	normalized_new_price	...	brand_name_Spice	brand_name_Vivo	brand_name_XOLO	brand_name_Xiaomi	brand_name_ZTE	os_Others	os_Windows	os_iOS	4g_yes	5g_yes
0	1.0	14.50	13.0	5.0	64.0	3.0	3020.0	146.0	127	4.715100	...	0	0	0	0	0	0	0	0	1	0
1	1.0	17.30	13.0	16.0	128.0	8.0	4300.0	213.0	325	5.519018	...	0	0	0	0	0	0	0	0	1	1
2	1.0	16.69	13.0	8.0	128.0	8.0	4200.0	213.0	162	5.884631	...	0	0	0	0	0	0	0	0	1	1
3	1.0	25.50	13.0	8.0	64.0	6.0	7250.0	480.0	345	5.630961	...	0	0	0	0	0	0	0	0	1	1
4	1.0	15.32	13.0	8.0	64.0	3.0	5000.0	185.0	293	4.947837	...	0	0	0	0	0	0	0	0	1	0

5 rows x 49 columns

- The devices were released 1 to 8 years before 2021
- On average, the devices were released 5 years before 2021
- The dummy-transformation of categorical columns has moved the number of columns to 49
- The train data has 2417 records meanwhile the test data has 1037 records

Model Building: Primary Model

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:	Wed, 25 Oct 2023	Prob (F-statistic):	0.00
Time:	15:52:37	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

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.78e+05. This might indicate that there are strong multicollinearity or other numerical problems.

- The Adjusted R-Squared is 0.842, which is good (about 84% of the variance is explained by the model)
- The const coefficient(y-intercept) is 1.3156
- The coefficient of normalized_new_price is 0.4311
- days_used and the dummy variables of the brand and os categorical variables all have p-values greater than the level of significance (0.05) and thus will have to be trimmed and the model regenerated and reassessed

Link to Appendix slide on model assumptions

Model Building: Model Performance Check

Training Performance

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

Test Performance

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

- The training R-squared is 0.84, so the model is not underfitting
- The train and test RMSE and MAE are comparable, so the model is not overfitting either
- MAE suggests that the model can predict the normalized price of used devices within a mean error of 0.18 on the test data
- MAPE of 4.5 on the test data means that we are able to predict within 4.5% of the normalized price of used devices

Link to Appendix slide on model assumptions

Model Building: Checking Linear Regression Assumptions: Treatment of multicollinearity

	feature	VIF
0	const	227.744081
1	screen_size	7.677290
2	main_camera_mp	2.285051
3	selfie_camera_mp	2.812473
4	int_memory	1.364152
5	ram	2.282352
6	battery	4.081780
7	weight	6.396749
8	days_used	2.660269
9	normalized_new_price	3.119430
10	years_since_release	4.899007
11	brand_name_Alcatel	3.405693
12	brand_name_Apple	13.057668
13	brand_name_Asus	3.332038
14	brand_name_BlackBerry	1.632378
15	brand_name_Celkon	1.774721
16	brand_name_Coolpad	1.468006
17	brand_name_Gionee	1.951272
18	brand_name_Google	1.321778
19	brand_name_HTC	3.410361
20	brand_name_Honor	3.340687
21	brand_name_Huawei	5.983852
22	brand_name_Infinix	1.283955
23	brand_name_Karbonn	1.573702
24	brand_name_LG	4.849832
25	brand_name_Lava	1.711360
26	brand_name_Lenovo	4.558941
27	brand_name_Meizu	2.179607
28	brand_name_Micromax	3.363521
29	brand_name_Microsoft	1.869751
30	brand_name_Motorola	3.274558
31	brand_name_Nokia	3.479849
32	brand_name_OnePlus	1.437034
33	brand_name_Oppo	3.971194
34	brand_name_Others	9.711034
35	brand_name_Panasonic	2.105703
36	brand_name_Realme	1.946812
37	brand_name_Samsung	7.539866
38	brand_name_Sony	2.943161
39	brand_name_Spice	1.688863
40	brand_name_Vivo	3.651437
41	brand_name_XOLO	2.138070
42	brand_name_Xiaomi	3.719689
43	brand_name_ZTE	3.797581
44	os_Others	1.859863
45	os_Windows	1.596034
46	os_iOS	11.784684
47	4g_yes	2.467681
48	5g_yes	1.813900

VIF after dropping screen_size

	feature	VIF
0	const	202.673906
1	main_camera_mp	2.281835
2	selfie_camera_mp	2.809009
3	int_memory	1.362043
4	ram	2.282350
5	battery	3.842989
6	weight	2.993855
7	days_used	2.648929
8	normalized_new_price	3.077650
9	years_since_release	4.730315
10	brand_name_Alcatel	3.405533
11	brand_name_Apple	13.000338
12	brand_name_Asus	3.326698
13	brand_name_BlackBerry	1.631042
14	brand_name_Celkon	1.774528
15	brand_name_Coolpad	1.467719
16	brand_name_Gionee	1.941437
17	brand_name_Google	1.319334
18	brand_name_HTC	3.399980
19	brand_name_Honor	3.340354
20	brand_name_Huawei	5.981046
21	brand_name_Infinix	1.283526
22	brand_name_Karbonn	1.573494
23	brand_name_LG	4.832548
24	brand_name_Lava	1.711092
25	brand_name_Lenovo	4.553789
26	brand_name_Meizu	2.176424
27	brand_name_Micromax	3.358629
28	brand_name_Microsoft	1.868243
29	brand_name_Motorola	3.262356
30	brand_name_Nokia	3.464643
31	brand_name_OnePlus	1.437004
32	brand_name_Oppo	3.965445
33	brand_name_Others	9.652572
34	brand_name_Panasonic	2.104853
35	brand_name_Realme	1.943845
36	brand_name_Samsung	7.523421
37	brand_name_Sony	2.937375
38	brand_name_Spice	1.683302
39	brand_name_Vivo	3.650625
40	brand_name_XOLO	2.137844
41	brand_name_Xiaomi	3.713988
42	brand_name_ZTE	3.788971
43	os_Others	1.625212
44	os_Windows	1.595936
45	os_iOS	11.678957
46	4g_yes	2.466915
47	5g_yes	1.810289

	col	Adj. R-squared after_dropping col	RMSE after dropping col
0	screen_size	0.838381	0.234703
1	weight	0.838071	0.234928

- screen_size and weight display moderate multicollinearity while
- we will be ignoring the vif of dummy variables
- We will drop screen_size since dropping it has the the least impact on R-squared
- After dropping screen_size, the VIFs of all the non-dummy predictor variables are below 5; thus, we have eliminated multicollinearity from the data used for the model

[Link to Appendix slide on model assumptions]

Model Building: Checking Linear Regression Assumptions: Dropping high p-value variables

```
                            OLS Regression Results
==============================================================================
Dep. Variable:     normalized_used_price   R-squared:                       0.839
Model:                               OLS   Adj. R-squared:                  0.838
Method:                    Least Squares   F-statistic:                     895.7
Date:                   Mon, 23 Oct 2023   Prob (F-statistic):               0.00
Time:                           18:31:48   Log-Likelihood:                 80.645
No. Observations:                   2417   AIC:                            -131.3
Df Residuals:                       2402   BIC:                            -44.44
Df Model:                             14
Covariance Type:               nonrobust
=========================================================================================
                            coef    std err          t      P>|t|      [0.025      0.975]
-----------------------------------------------------------------------------------------
const                     1.5000      0.048     30.955      0.000       1.405       1.595
main_camera_mp            0.0210      0.001     14.714      0.000       0.018       0.024
selfie_camera_mp          0.0138      0.001     12.858      0.000       0.012       0.016
ram                       0.0207      0.005      4.151      0.000       0.011       0.030
weight                    0.0017      6e-05     27.672      0.000       0.002       0.002
normalized_new_price      0.4415      0.011     39.337      0.000       0.419       0.463
years_since_release      -0.0292      0.003     -8.589      0.000      -0.036      -0.023
brand_name_Karbonn        0.1156      0.055      2.111      0.035       0.008       0.223
brand_name_Samsung       -0.0374      0.016     -2.270      0.023      -0.070      -0.005
brand_name_Sony          -0.0670      0.030     -2.197      0.028      -0.127      -0.007
brand_name_Xiaomi         0.0801      0.026      3.114      0.002       0.030       0.130
os_Others                -0.1276      0.027     -4.667      0.000      -0.181      -0.074
os_iOS                   -0.0900      0.045     -1.994      0.046      -0.179      -0.002
4g_yes                    0.0502      0.015      3.326      0.001       0.021       0.080
5g_yes                   -0.0673      0.031     -2.194      0.028      -0.127      -0.007
==============================================================================
Omnibus:                      246.183   Durbin-Watson:                   1.902
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              483.879
Skew:                          -0.658   Prob(JB):                    8.45e-106
Kurtosis:                       4.753   Cond. No.                     2.39e+03
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 2.39e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Training Performance

	RMSE	MAE	R-squared	Adj. R-squared	MAPE
0	0.23403	0.182751	0.83924	0.838235	4.395407

Test Performance

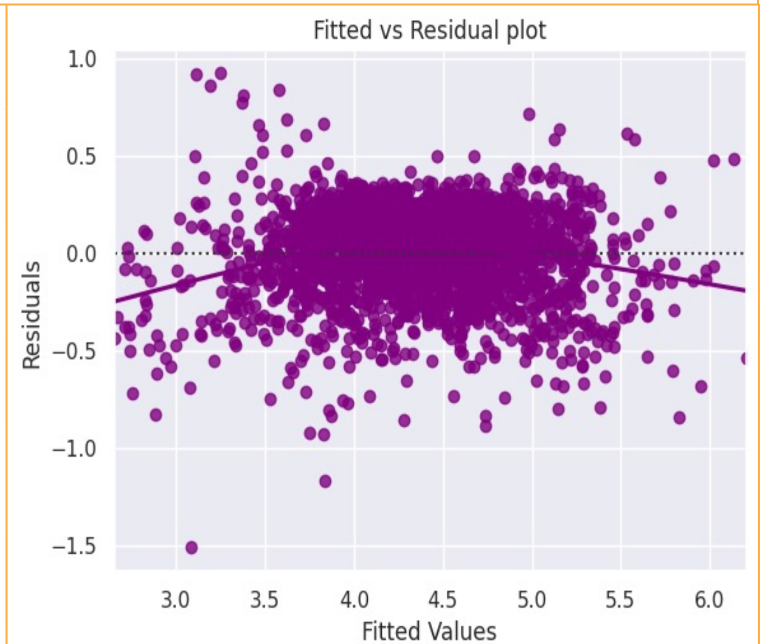
	RMSE	MAE	R-squared	Adj. R-squared	MAPE
0	0.241434	0.186649	0.838387	0.836013	4.556349

- After eliminating the predictor variables with high p-values, the list of variables (excluding the y-intercept) to be used for the model reduces to 14 variables
- Now no feature has p-value greater than 0.05, so we'll consider the features in x_train3 as the final set of predictor variables and olsmodel2 as the final model to move forward with
- The adjusted R-squared is now 0.838, i.e., our model is able to explain ~84% of the variance (not very far from the value, 0.842, of olsmodel1); thus, variables dropped did not significantly affect the model
- RMSE and MAE values are comparable for train and test sets, indicating that the model is not overfitting
- MAE suggests that the model can predict the normalized price of used devices within a mean error of 0.19 on the test data
- MAPE of 4.56 on the test data means that we are able to predict within 4.56% of the normalized price of used devices

Link to Appendix slide on model assumptions

Model Building: Checking Linear Regression Assumptions: Test for Linearity and Independence

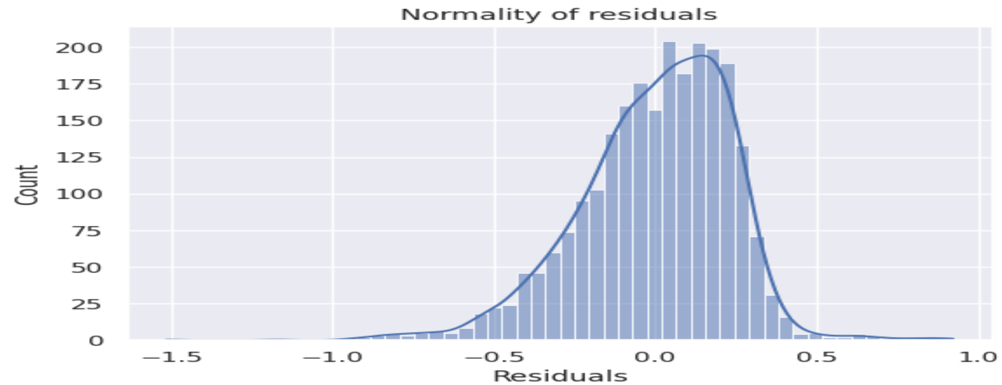
	Actual Values	Fitted Values	Residuals
3026	4.087488	3.867319	0.220169
1525	4.448399	4.602001	-0.153602
1128	4.315353	4.286957	0.028395
3003	4.282068	4.195169	0.086899
2907	4.456438	4.490563	-0.034125

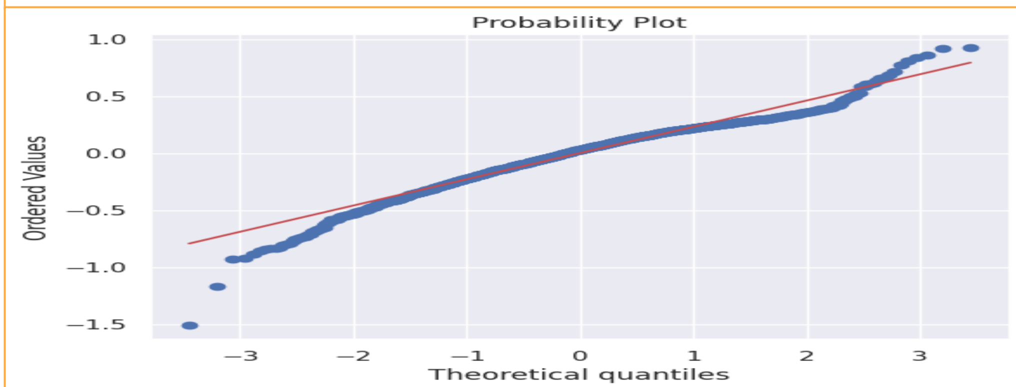


- The assumptions of linearity and independence are satisfied since the plot has no discernible pattern

Link to Appendix slide on model assumptions

Model Building: Checking Linear Regression Assumptions: Test for Normality and Homoscedasticity





- The distribution of residuals is somewhat bell-shaped
- The Q-Q plot of residuals follows a straight line for the most part
- The p-values (0.44) of the goldfeldquandt test is greater than 0.05 and, thus, we fail to reject the null hypothesis; in other words, we can conclude that residuals are homoscedastic

Link to Appendix slide on model assumptions

Model Performance Summary

Training Performance

	RMSE	MAE	R-squared	Adj. R-squared	MAPE
0	0.23403	0.182751	0.83924	0.838235	4.395407

Test Performance

	RMSE	MAE	R-squared	Adj. R-squared	MAPE
0	0.241434	0.186649	0.838387	0.836013	4.556349

- We used the Ordinary Least Squares (OLS) Method to design our model
- After imputation of missing values and engineering of the years since release column, the data was split into a y-variable (normalized_used_price) and a series of X-variables (containing the other columns); and a constant (intercept) was added to the X-variables series
- The categorical columns of X (brand_name, os, 4g, and 5g) were dummy-transformed in preparation for the model building
- As a final step before the model building, the data (y- and X-variables) were split into training and test data in a ratio of 70:30
- After building the model and tuning it to eliminate multicollinearity and high p-values for significance, we arrived at a model with the following principal features:
  - The adjusted R-squared is 0.838, i.e., our model is able to explain ~84% of the variance
  - RMSE and MAE values are comparable for train and test sets, indicating that the model is not overfitting
  - MAE suggests that the model can predict the normalized price of used devices within a mean error of 0.19 on the test data
  - MAPE of 4.56 on the test data means that we are able to predict within 4.56% of the normalized price of used devices

Link to Appendix slide on model assumptions

APPENDIX

Data Background and Contents: Data Overview

	0	1	2	3	4
brand_name	Honor	Honor	Honor	Honor	Honor
os	Android	Android	Android	Android	Android
screen_size	14.5	17.3	16.69	25.5	15.32
4g	yes	yes	yes	yes	yes
5g	no	yes	yes	yes	no
main_camera_mp	13.0	13.0	13.0	13.0	13.0
selfie_camera_mp	5.0	16.0	8.0	8.0	8.0
int_memory	64.0	128.0	128.0	64.0	64.0
ram	3.0	8.0	8.0	6.0	3.0
battery	3020.0	4300.0	4200.0	7250.0	5000.0
weight	146.0	213.0	213.0	480.0	185.0
release_year	2020	2020	2020	2020	2020
days_used	127	325	162	345	293
normalized_used_price	4.307572	5.162097	5.111084	5.135387	4.389995
normalized_new_price	4.7151	5.519018	5.884631	5.630961	4.947837

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3454 entries, 0 to 3453
Data columns (total 15 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   brand_name             3454 non-null   object
 1   os                     3454 non-null   object
 2   screen_size            3454 non-null   float64
 3   4g                     3454 non-null   object
 4   5g                     3454 non-null   object
 5   main_camera_mp         3275 non-null   float64
 6   selfie_camera_mp       3452 non-null   float64
 7   int_memory             3450 non-null   float64
 8   ram                    3450 non-null   float64
 9   battery                3448 non-null   float64
 10  weight                 3447 non-null   float64
 11  release_year           3454 non-null   int64
 12  days_used              3454 non-null   int64
 13  normalized_used_price  3454 non-null   float64
 14  normalized_new_price   3454 non-null   float64
dtypes: float64(9), int64(2), object(4)
memory usage: 404.9+ KB
```

	count	unique	top	freq	mean	std	min	25%	50%	75%	max
brand_name	3454	34	Others	502	NaN	NaN	NaN	NaN	NaN	NaN	NaN
os	3454	4	Android	3214	NaN	NaN	NaN	NaN	NaN	NaN	NaN
screen_size	3454.0	NaN	NaN	NaN	13.713115	3.80528	5.08	12.7	12.83	15.34	30.71
4g	3454	2	yes	2335	NaN	NaN	NaN	NaN	NaN	NaN	NaN
5g	3454	2	no	3302	NaN	NaN	NaN	NaN	NaN	NaN	NaN
main_camera_mp	3275.0	NaN	NaN	NaN	9.460208	4.815461	0.08	5.0	8.0	13.0	48.0
selfie_camera_mp	3452.0	NaN	NaN	NaN	6.554229	6.970372	0.0	2.0	5.0	8.0	32.0
int_memory	3450.0	NaN	NaN	NaN	54.573099	84.972371	0.01	16.0	32.0	64.0	1024.0
ram	3450.0	NaN	NaN	NaN	4.036122	1.365105	0.02	4.0	4.0	4.0	12.0
battery	3448.0	NaN	NaN	NaN	3133.402697	1299.682844	500.0	2100.0	3000.0	4000.0	9720.0
weight	3447.0	NaN	NaN	NaN	182.751871	88.413228	69.0	142.0	160.0	185.0	855.0
release_year	3454.0	NaN	NaN	NaN	2015.965258	2.298455	2013.0	2014.0	2015.5	2018.0	2020.0
days_used	3454.0	NaN	NaN	NaN	674.869716	248.580166	91.0	533.5	690.5	868.75	1094.0
normalized_used_price	3454.0	NaN	NaN	NaN	4.364712	0.588914	1.536867	4.033931	4.405133	4.7557	6.619433
normalized_new_price	3454.0	NaN	NaN	NaN	5.233107	0.683637	2.901422	4.790342	5.245892	5.673718	7.847841

```
brand_name                 0
os                         0
screen_size                0
4g                         0
5g                         0
main_camera_mp           179
selfie_camera_mp           2
int_memory                 4
ram                        4
battery                    6
weight                     7
release_year               0
days_used                  0
normalized_used_price      0
normalized_new_price       0
dtype: int64
```

- The data contains 3454 rows (records) and 15 columns (attributes)
- The data contains 4 categorical columns (object) and 11 numerical columns (9 float, 2 int)
- The normalized price for used devices ranges from 1.54 to 6.62 with an average of 4.36 and median of 4.41 giving the impression that the distribution is almost normal (since the mean and median are close)
- The normalized price of new devices ranges from 2.90 to 7.85 with an average of 5.23 and a median of 5.25; probably also having a normal distribution
- The brand names of several devices (502) are not identified
- Android is the most popular operating system (3214)
- Most devices are 4g-enabled (2335) but many more do not support 5g (3302)

Model Assumptions

Variance Inflation Factor (VIF)

- The VIF was used to test for multicollinearity
- screen_size and weight had VIF values between 5 and 10 and, thus, showed moderate levels of multicollinearity
- The multicollinearity was completely eliminated by excluding screen_size

Fitted versus Residual Plot

- The Fitted (predicted values) vs Residual (error – difference between predicted and actual values) plot was used to test for linearity and independence
- No pattern was identifiable from the plot; thus, the assumptions of linearity and independence were satisfied

Distribution of Residuals , Quantile-Quantile (Q-Q) plot, and Shapiro-Wilks test

- The Q-Q plot (plot of ordered values vs theoretical quantiles) was used to test for normality
- Most of the plotted values followed the theoretical 45 degrees line, thus demonstrating that the distribution of the residuals is approximately normal
- The approximate normal distribution of the residuals was confirmed by the corresponding distribution plot
- However, the Shapiro-Wilks test showed departure from strict normality by producing a p-value less than 0.05

Goldfeldquandt test

- The goldfeldquandt test was used to test for homoscedasticity
- The goldfeldquandt test produced a p-value greater than 0.05, thus preventing rejection of the null hypothesis; so, we concluded that the residuals are homoscedastic

Happy Learning !