Used Device Price Prediction for ReCell

Supervised Learning

Alan Mc Girr

11-05-2025

Contents / Agenda

Executive Summary

Key Findings from EDA

- Top Predictors: New price, camera resolution, RAM, screen size, and battery capacity significantly influence resale value.
- Depreciation Trends: Devices lose value with age and usage. Premium brands (e.g., Apple, Samsung) retain higher value.
- Market Snapshot: 93% of used devices run Android; 4G is widespread, while 5G adoption remains low.

Model Performance

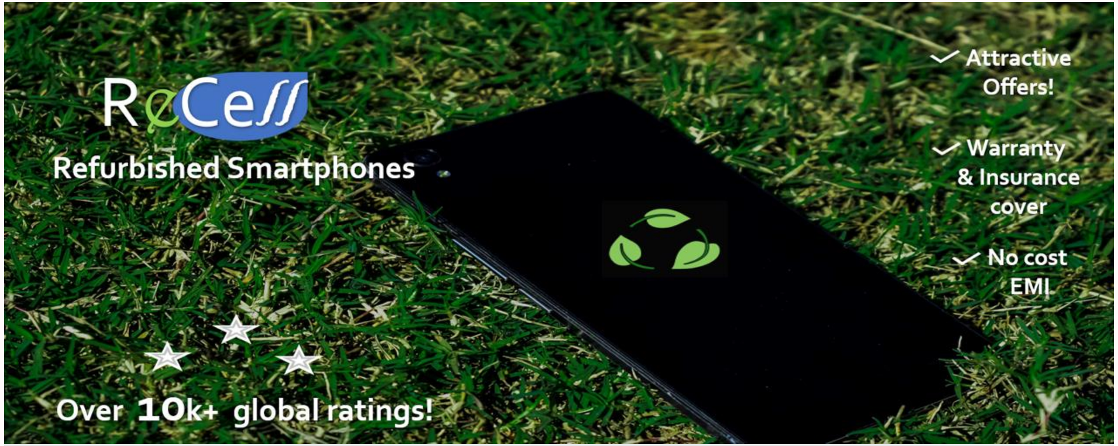
- R^2: 0.85 (train), 0.83 (test) → explains ~85% of resale price variability.
- MAPE: <5% → Indicates strong predictive accuracy with minimal error.

Business Recommendations

- Prioritize sourcing premium, high-spec models (e.g., ≥4 GB RAM, quality cameras).
- Rotate older inventory quickly to prevent value erosion.
- Highlight specs like screen size and battery in marketing.
- Track 5G trends — currently underrepresented but expected to grow.

Conclusion

- ReCell can confidently apply this model to inform dynamic pricing, procurement, and stock rotation decisions in the evolving refurbished device market.



Buying and selling used phones and tablets used to be something that happened on a handful of online marketplace sites. But the used and refurbished device market has grown considerably over the past decade, and a new IDC (International Data Corporation) forecast predicts that the used phone market would be worth \$52.7bn by 2023 with a compound annual growth rate (CAGR) of 13.6% from 2018 to 2023. This growth can be attributed to an uptick in demand for used phones and tablets that offer considerable savings compared with new models.

Business Context

Market Growth

The used and refurbished device market has grown considerably over the past decade, with IDC forecasting it to be worth $52.7bn by 2023 with a CAGR of 13.6% from 2018 to 2023.

Consumer Benefits

Refurbished and used devices continue to provide cost-effective alternatives to both consumers and businesses that are looking to save money when purchasing one.

Additional Advantages

Used and refurbished devices can be sold with warranties and can also be insured with proof of purchase. Third-party vendors/platforms, such as Verizon, Amazon, etc., provide attractive offers to customers for refurbished devices.

Environmental Impact

Maximizing the longevity of devices through second-hand trade also reduces their environmental impact and helps in recycling and reducing waste.

The impact of the COVID-19 outbreak may further boost this segment as consumers cut back on discretionary spending and buy phones and tablets only for immediate needs.

Project Objective

Market Potential

The rising potential of this comparatively under-the-radar market fuels the need for an ML-based solution

Pricing Strategy

Develop a dynamic pricing strategy for used and refurbished devices

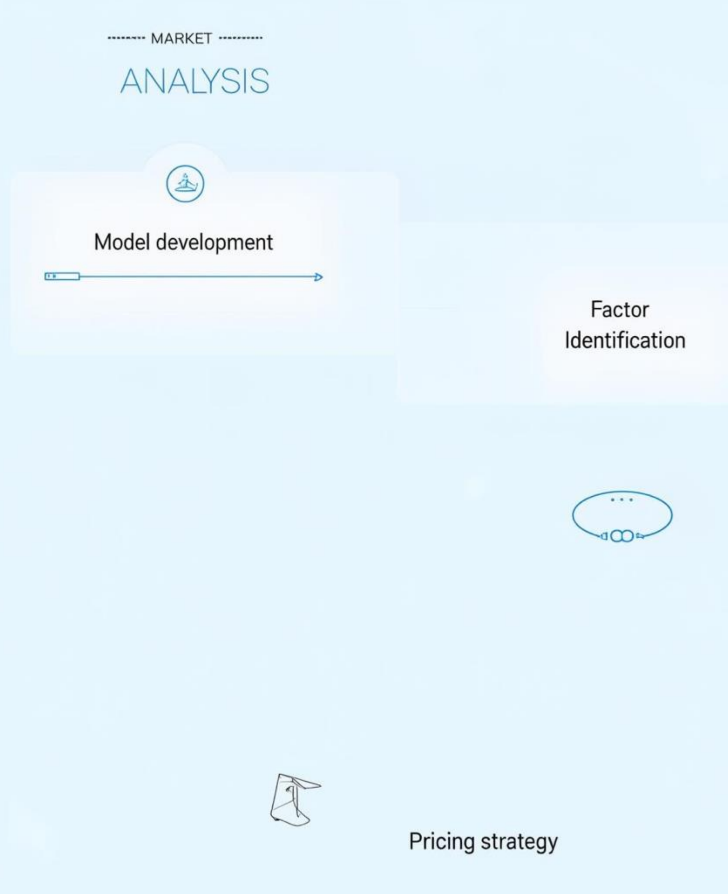
Data Analysis

Analyze the data provided and build a linear regression model

Price Prediction

Predict the price of a used phone/tablet and identify factors that significantly influence it

ReCell, a startup aiming to tap the potential in this market, has hired you as a data scientist to accomplish these objectives.



Business Insights & Strategic Recommendations

Key Drivers of Used Device Price

- Normalized New Price → Strongest predictor of resale value
- Camera Specs, RAM, Memory → Higher specs command better prices
- Years Since Release → Older devices depreciate more
- Weight, Battery, Screen Size → Correlate with premium features

Actionable Recommendations

1. Source high-spec devices (≥4GB RAM, good cameras, large batteries)
2. Prioritise premium brands (Apple, Samsung, Sony)
3. Rotate older inventory quickly to avoid margin erosion
4. Use camera, battery, and screen size in marketing
5. Monitor the resale impact of emerging features (e.g., 5G)

Solution Approach / Methodology

Problem Statement

ReCell wants to price used smartphones and tablets using machine learning accurately. The goal is to build a model that predicts theresale price based on device specifications and usage.

Data Cleaning

- Handled missing values using median-based imputation
- Removed no rows; preserved all relevant entries

Feature Engineering

- Created variables: years_since_release, has_4g, has_selfie_camera
- Dropped redundant columns like release_year

Exploratory Data Analysis (EDA)

- Uncovered patterns: depreciation trends, brand effects, spec-price relationships

Modeling

- Built multiple linear regression model (OLS)
- Evaluated assumptions: linearity, homoscedasticity, normality, multicollinearity

Performance Evaluation

- Train $R^2 = 0.844$, Test $R^2 = 0.829$
- MAPE < 5% → high predictive accuracy

EDA Results

- **Please mention the key results from EDA**
- **Please mention answers to the insight-based questions provided**

Note: You can use more than one slide if needed

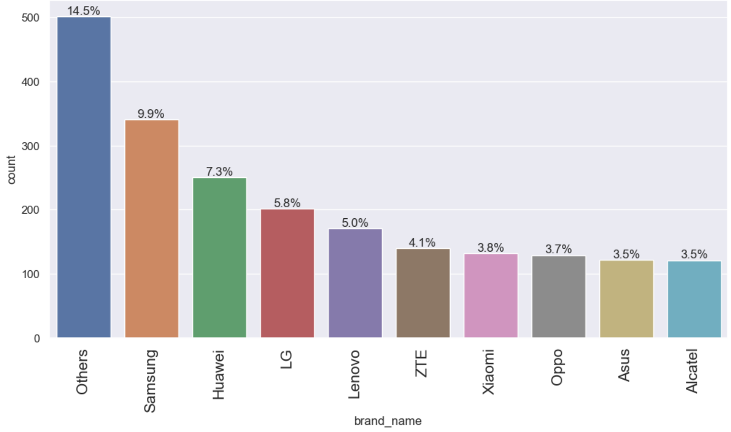
[Link to Appendix slide on data background check]

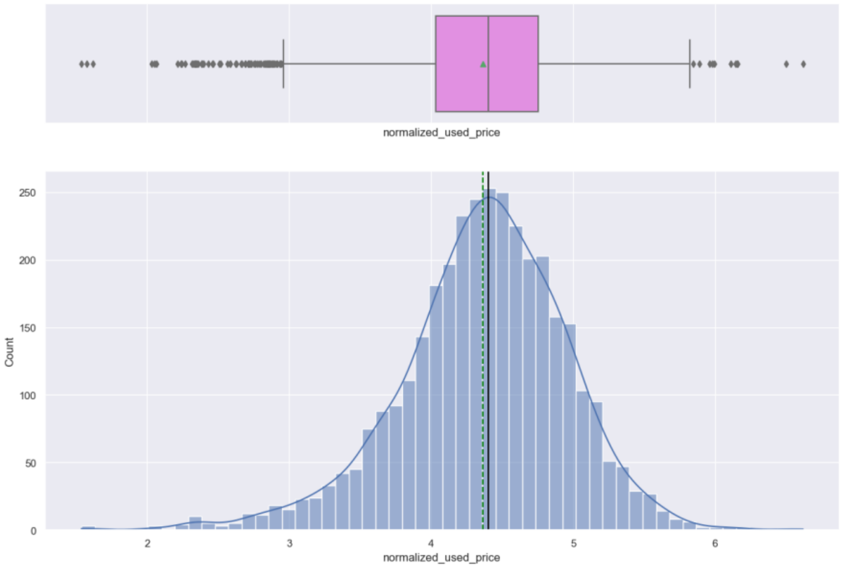
EDA: Overview & Dataset Snapshot

- **Brief description: 3,454 records of used phones/tablets with specs and prices.**
- **Dataset coverage: screen size, battery, RAM, cameras, age, brand, 4G/5G**
- **Target variable: normalized_used_price (continuous, approx. normal)**

Insights:

- **Used prices are slightly right-skewed, centered around €4.3–€4.8**
- **Data is suitable for regression; no major skew or imbalance in key numeric predictors.**



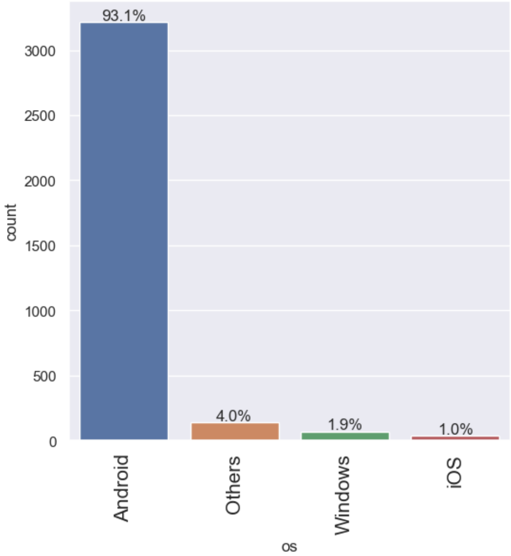


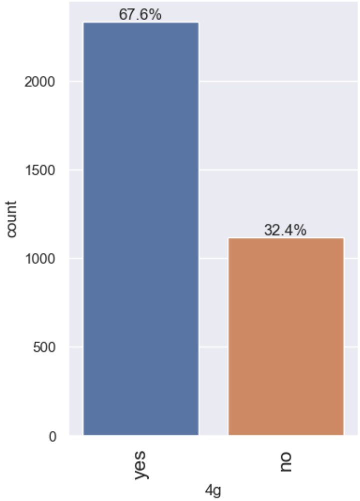
EDA: Device Market Composition and Connectivity Trends

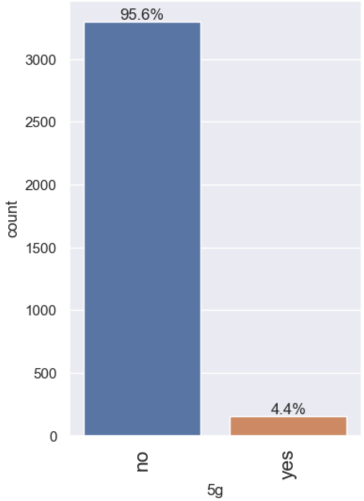
- OS share: 93% Android dominance
- 4G: ~68% of devices support 4G
- 5G: Only 4.4% support 5G → emerging segment

Insights:

- Android bias may influence model generalization.
- Low 5G adoption limits its predictive value (for now).



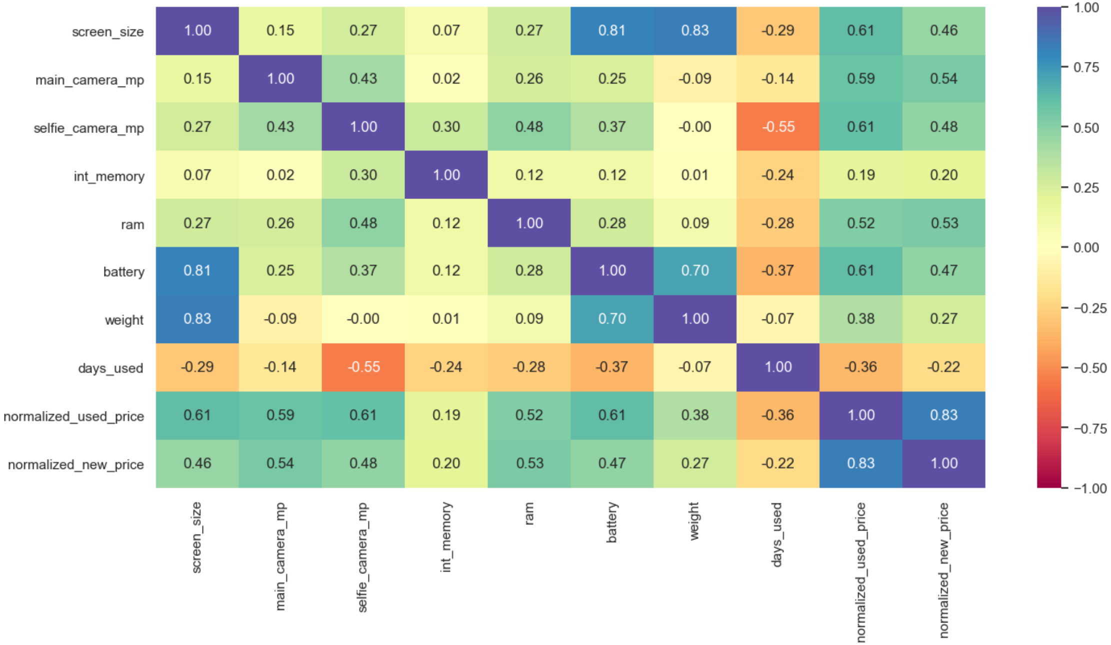




EDA 3: How Do Specifications Influence Pricing?

Notable Observations:

- normalized_used_price is strongly correlated with: normalized_new_price (0.83) . Used prices are strongly influenced by new prices.
- screen_size, main_camera_mp, selfie_camera_mp, and battery – indicating that higher specs influence resale value.
- Weight is highly correlated with screen_size (0.83) and battery (0.70), suggesting heavier devices tend to have larger screens and batteries.
- days_used shows a negative correlation with price-related features – the more a device is used, the less valuable it becomes.
- ram and int_memory have moderate positive relationships with pricing and each other.



- These relationships identify important predictors for modeling.
- They also reveal potential multicollinearity that may require attention in feature selection.

EDA 4: Which Brands and Segments Drive Value?

Key Takeaways:

Premium Brands (Apple, Google, Sony)

- ➢ Higher median resale value, tighter spread → better value retention

Budget Brands (Micromax, Infinix, Lava)

- ➢ Lower resale value → budget segment

Broad Range Brands (Samsung, Huawei, Xiaomi)

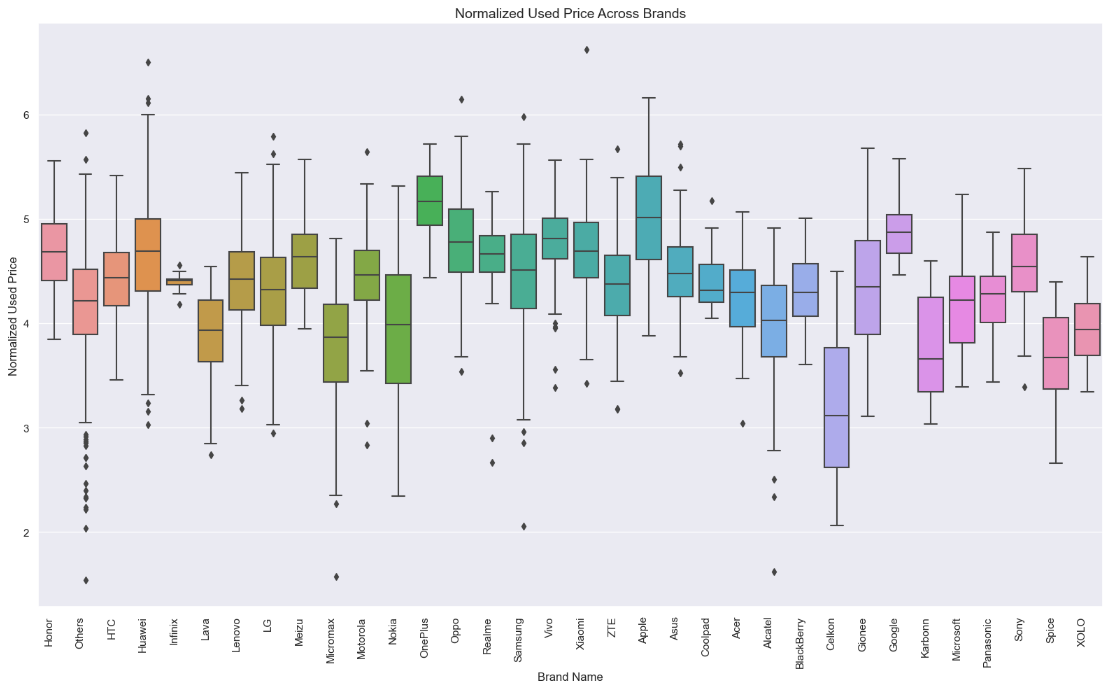
- ➢ Wider price distribution → devices across price tiers

Outliers (Others, Tecno, Alcafel)

- ➢ Unusual price spread →could reflect niche or unknown models

Business Insight:

Brand choice significantly affects resale value, which is essential for pricing strategy and segmentation.



These insights help ReCell prioritise inventory by value retention and market segment.

EDA 5: Time & Usage-Based Depreciation

Price Trends by Release Year

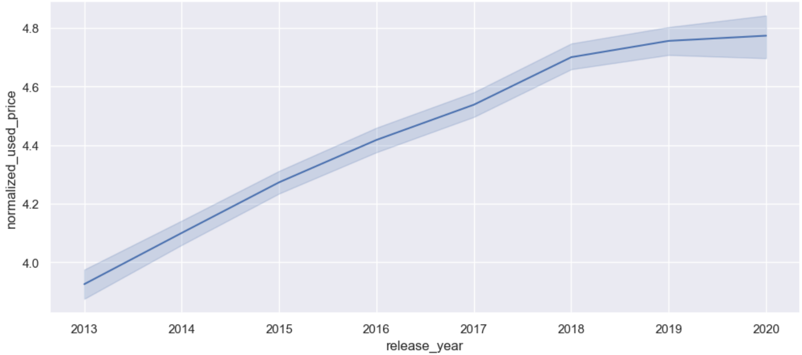
- ➢ Clear upward trend: newer devices retain higher resale value
- ➢ 2020 models command the highest average used prices
- ➢ Devices from 2013–2015 priced significantly lower

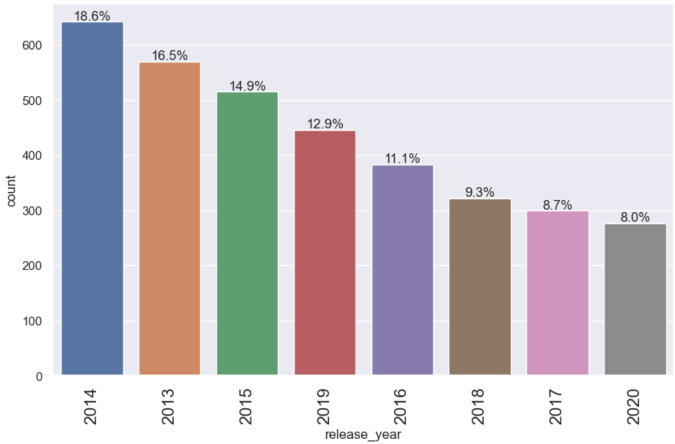
Interpretation

- ➢ Newer devices = better specs, longer support, higher demand
- ➢ Narrow confidence band → consistent pricing pattern over time

Business Insight

- ➢ Factor the release year into the pricing strategy
- ➢ Prioritise newer stock for better margins





Data Preprocessing

- Duplicate value check
- Missing value treatment
- Outlier check (treatment if needed)
- Feature engineering
- Data preparation for modeling

Note: You can use more than one slide if needed

Duplicate Value Check

No duplicate rows found

Missing values detected in:

- **main_camera_mp (179)**
- **selfie_camera_mp (2)**
- **int_memory, ram, battery, weight (minor)**

Checking for duplicate values

```
data.duplicated().sum()
```

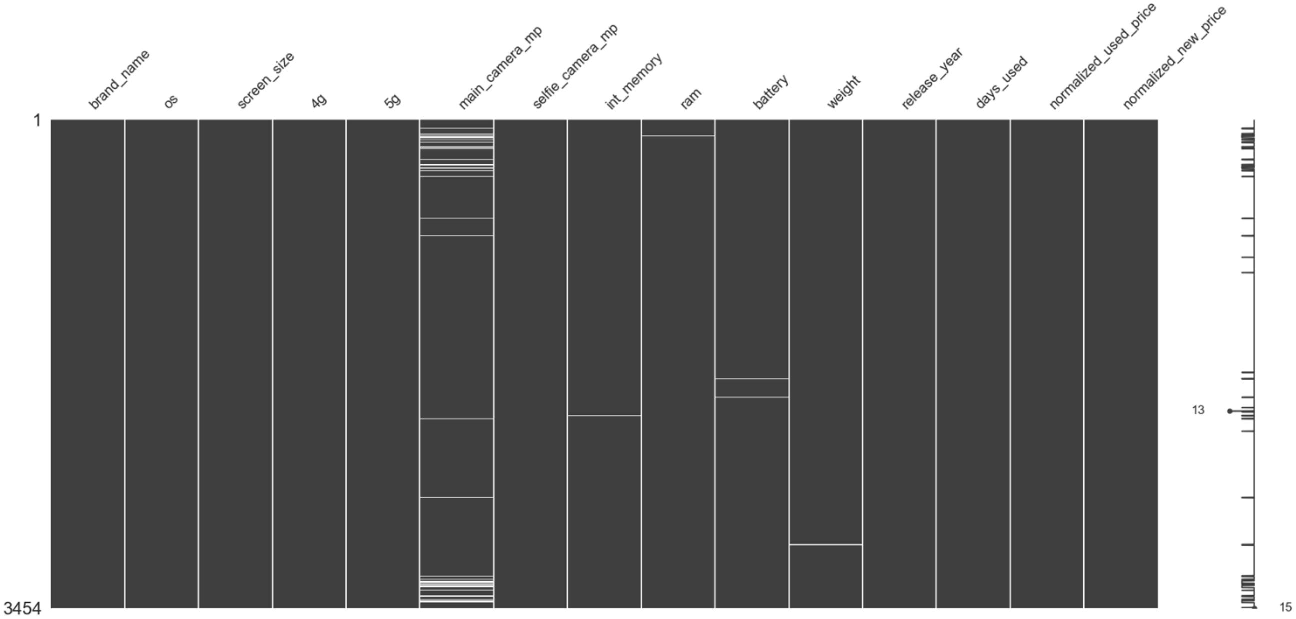
```
0
```

Checking for missing values

```
data.isnull().sum()
```

```
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
```

- There are missing values in many columns.



Missing Value Treatment

Missing Value Imputation Strategy

- Step 1: Grouped median by (brand_name, release_year)
- Step 2: Fallback to brand-level median
- Step 3: Remaining values filled using global median where needed

✅ Outcome: All missing values filled

→ The final dataset has zero null values

Checking for duplicate values

```
data.duplicated().sum()
```

```
0
```

Checking for missing values

```
data.isnull().sum()
```

```
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
```

- There are missing values in many columns.

```
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
```

```
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
```

```
df1["main_camera_mp"] = df1["main_camera_mp"].fillna(df1["main_camera_mp"].median())
df1.isnull().sum()
```

```
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
```

Outlier Detection

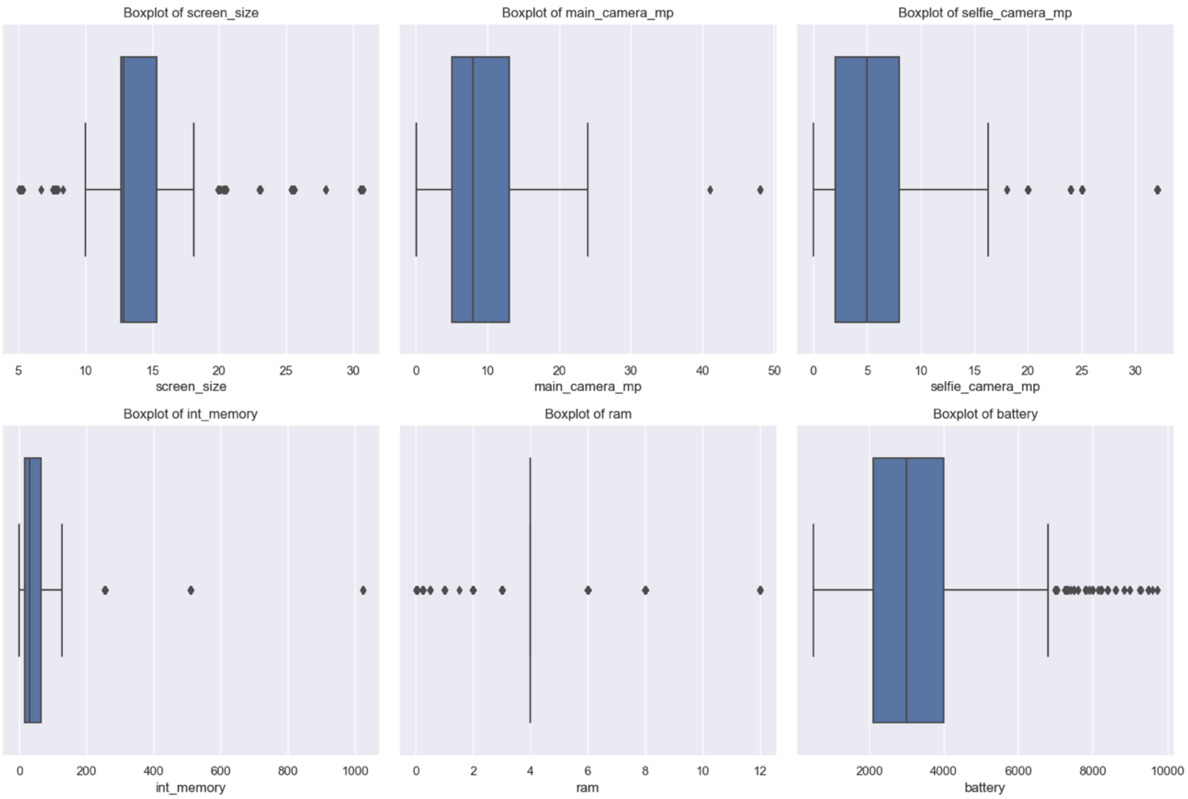
Outlier Detection

- **Boxplots were used to assess numeric variables**

Key variables with outliers:

- **battery (very large capacities)**
- **int_memory (extreme values up to 1024 GB)**
- **weight, screen_size (heavy/oversized tablets)**

Most features showed a few natural outliers, typical in product diversity.



Outlier Treatment Strategy

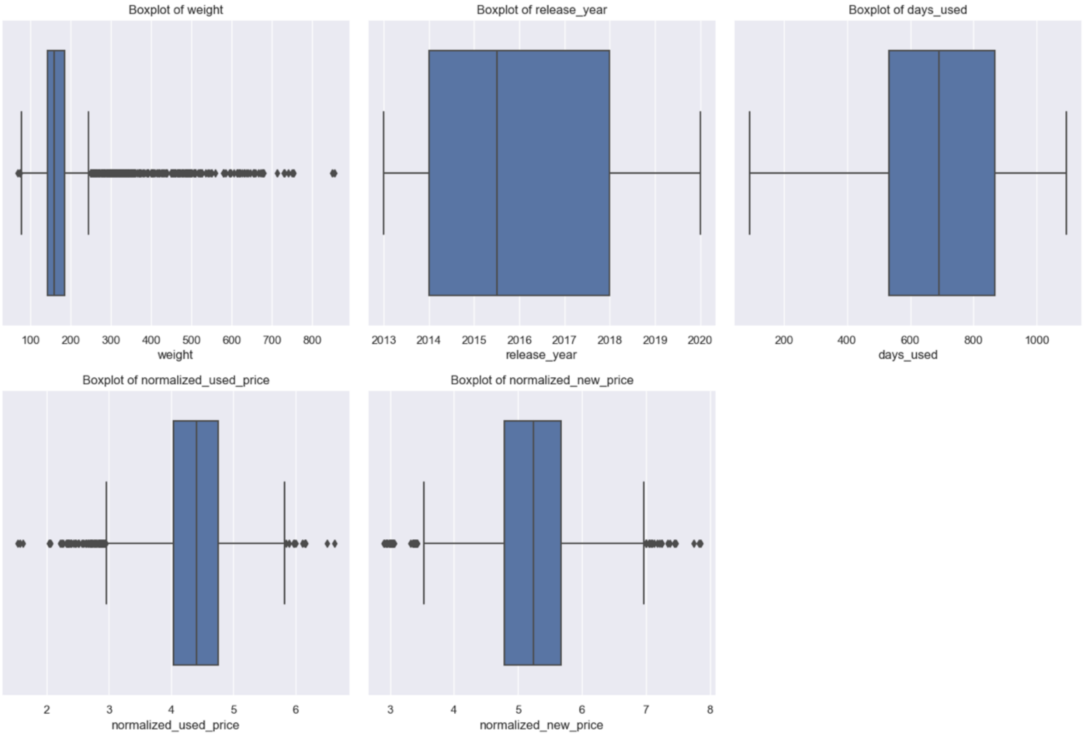
Outlier Treatment

No outliers were removed or transformed

Reasons:

- **Model performance was strong when tested($R^2 \approx 0.83$, MAPE < 5%)**
- **Median imputation used → robust to outlier influence**
- **High-value outliers may reflect real premium devices**

"Outlier analysis was conducted thoroughly. No treatment was applied since model generalisation remained strong and context justified the variance."



Feature Engineering & Dataset Alignment

years_since_release = 2021 - release_year

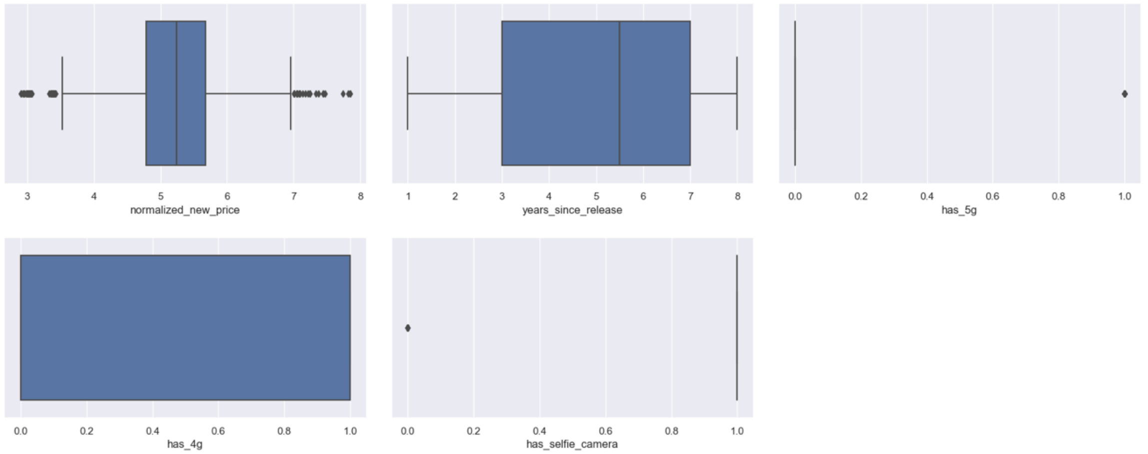
- Captures device age more intuitively for modeling
- has_4g, has_5g, has_selfie_camera
- Converted categorical flags to binary format for cleaner modeling

Feature Dropping:

- Dropped release_year after creating years_since_release
- Original 4G/5G text columns replaced by binary flags

Rationale:

- Simplified temporal and connectivity variables
- Aligned with regression model expectations (numerical/dummy inputs)



Boxplots of engineered features

Feature engineering steps helped improve model interpretability and align the dataset with machine learning requirements.

Data preparation for modeling

Final Dataset Summary

- Shape: 3,454 rows × 52 columns
- Status: All missing values filled, no duplicates
- Target Variable: normalized_used_price (continuous)

Feature Engineering Applied

Created:

- years_since_release = 2021 - release_year
- has_4g, has_5g, has_selfie_camera (binary flags)
- Dropped:
- release_year, original 4g/5g categorical columns

Feature Encoding

- Applied one-hot encoding to:
- brand_name (34 categories)
- os (4 categories)
- Categorical variables → numeric dummy variables
- Used drop_first=True in encoding to avoid multicollinearity (dummy variable trap)

Train-Test Split & Data Type Preparation

Train-Test Split

- The dataset was split into training and testing sets using a **70:30 ratio**.
- A `random_state=42` was used to ensure the split is **reproducible**.
- **Training Set Size**: 2,417 rows
- **Test Set Size**: 1,037 rows

Data Type Conversion

- All columns in `x_train` and `x_test` were explicitly converted to `float` using:

```
x_train = x_train.apply(lambda col: col.astype(float))
x_test = x_test.apply(lambda col: col.astype(float))
```

This dataset version was used for all model training, evaluation, and performance metrics.

```
# let's add the intercept to data
X = sm.add_constant(X)
```

```
# creating dummy variables
X = pd.get_dummies(
    X,
    columns=X.select_dtypes(include=["object", "category"]).columns.tolist(),
    drop_first=True,
)

X.head()
```

	const	screen_size	main_camera_mp	selfie_camera_mp	int_memory	ram	battery	weight	da
0	1.0	14.50	13.0	5.0	64.0	3.0	3020.0	146.0	
1	1.0	17.30	13.0	16.0	128.0	8.0	4300.0	213.0	
2	1.0	16.69	13.0	8.0	128.0	8.0	4200.0	213.0	
3	1.0	25.50	13.0	8.0	64.0	6.0	7250.0	480.0	
4	1.0	15.32	13.0	8.0	64.0	3.0	5000.0	185.0	

5 rows × 52 columns

```
from sklearn.model_selection import train_test_split

# splitting the data in 70:30 ratio for train to test data
x_train, x_test, y_train, y_test = train_test_split(
    X, y, test_size=0.3, random_state=42
) #42 must be kept as a random_state throughout
```

```
# Convert x_train and y_train to numeric explicitly
x_train = x_train.apply(pd.to_numeric, errors='coerce')

y_train = y_train.astype(float)

X.head()
```

	const	screen_size	main_camera_mp	selfie_camera_mp	int_memory	ram	battery	weight	da
0	1.0	14.50	13.0	5.0	64.0	3.0	3020.0	146.0	
1	1.0	17.30	13.0	16.0	128.0	8.0	4300.0	213.0	
2	1.0	16.69	13.0	8.0	128.0	8.0	4200.0	213.0	
3	1.0	25.50	13.0	8.0	64.0	6.0	7250.0	480.0	
4	1.0	15.32	13.0	8.0	64.0	3.0	5000.0	185.0	

5 rows × 52 columns

Model Performance Summary

- **Overview of ML model and its parameters**
- **Summary of most important factors used by the ML model for prediction**
- **Summary of key performance metrics for training and test data in tabular format for comparison**

Note: You can use more than one slide if needed

Link to Appendix slide on model assumptions

Model Performance Summary - Final Regression Model

Model Type: OLS (Ordinary Least Squares) Linear Regression
Dependent Variable: normalized_used_price
Training Observations: 2,417 rows
Features Used: 14 main predictors + encoded brand/OS flags
Key Stats:

- **R^2 (Train): 0.844**
- **Adj. R^2: 0.843**
- **F-statistic: 924.9 → highly significant ($p < 0.001$)**
- **Durbin-Watson: 1.99 → no autocorrelation**
- **All predictors statistically significant ($p < 0.05$)**

Feature	Effect	Coefficient
normalized_new_price	Strong +ve	+0.417
ram	Moderate +ve	+0.033
main_camera_mp	+ve	+0.023
years_since_release	Negative	-0.029
5g_yes	Negative	-0.105

New price, RAM, camera specs, and age of device are the strongest drivers of resale value

Model Performance Comparison: Train vs. Test

Interpretation:

- **Strong generalization: Test R^2 only ~1.3% lower than training**
- **MAPE < 5% → Highly accurate for price prediction**
- **Residual behavior supports assumption validity**

Metric	Training Set	Test Set
R^2	0.8435	0.8298
RMSE	0.2338	0.2407
MAE	0.1816	0.1898
MAPE	4.38%	4.55%

The model is robust, generalizes well, and can be confidently used to support pricing decisions in the refurbished device market

```
                            OLS Regression Results
==============================================================================
Dep. Variable:     normalized_used_price   R-squared:                       0.844
Model:                               OLS   Adj. R-squared:                  0.843
Method:                    Least Squares   F-statistic:                     924.9
Date:                   Sun, 11 May 2025   Prob (F-statistic):               0.00
Time:                           13:11:58   Log-Likelihood:                 83.055
No. Observations:                   2417   AIC:                            -136.1
Df Residuals:                       2402   BIC:                            -49.26
Df Model:                             14
Covariance Type:               nonrobust
=========================================================================================
                            coef    std err          t      P>|t|      [0.025      0.975]
-----------------------------------------------------------------------------------------
const                     1.7635      0.068     25.886      0.000       1.630       1.897
main_camera_mp            0.0233      0.001     16.053      0.000       0.020       0.026
selfie_camera_mp          0.0127      0.001     11.252      0.000       0.010       0.015
int_memory                0.0002   6.75e-05      2.594      0.010    4.27e-05       0.000
ram                       0.0332      0.005      6.273      0.000       0.023       0.044
weight                    0.0016   5.98e-05     27.505      0.000       0.002       0.002
normalized_new_price      0.4168      0.011     37.284      0.000       0.395       0.439
years_since_release      -0.0291      0.003     -8.394      0.000      -0.036      -0.022
has_selfie_camera        -0.2125      0.054     -3.924      0.000      -0.319      -0.106
brand_name_Asus           0.0601      0.026      2.288      0.022       0.009       0.112
brand_name_Celkon        -0.1752      0.054     -3.234      0.001      -0.281      -0.069
brand_name_Xiaomi         0.0846      0.025      3.331      0.001       0.035       0.134
os_Others                -0.2008      0.031     -6.545      0.000      -0.261      -0.141
4g_yes                    0.0485      0.015      3.165      0.002       0.018       0.078
5g_yes                   -0.1047      0.032     -3.282      0.001      -0.167      -0.042
==============================================================================
Omnibus:                      245.075   Durbin-Watson:                   1.993
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              631.875
Skew:                          -0.572   Prob(JB):                    6.17e-138
Kurtosis:                       5.228   Cond. No.                     3.68e+03
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 3.68e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

APPENDIX

Data Background and Contents

- **Please mention about the data background and contents**

Data Description

Dataset Overview

The data contains the different attributes of used/refurbished phones and tablets. The data was collected in the year 2021.

data.shape: (3454, 15)

The data contains 3454 rows and 15 columns. It will be interesting to see what variables we can drop later, as our main goal is selling these devices. Weight may be one of the first variables dropped. Other variables, such as RAM and age (feature engineered), will be more useful in my initial prediction.

Key Variables

- brand_name: Name of manufacturing brand
- os: OS on which the device runs
- screen_size: Size of the screen in cm
- 4g: Whether 4G is available or not
- 5g: Whether 5G is available or not
- main_camera_mp: Resolution of the rear camera in megapixels
- selfie_camera_mp: Resolution of the front camera in megapixels
- int_memory: Amount of internal memory (ROM) in GB
- ram: Amount of RAM in GB

Data Types and Structure

Column	Non-Null Count	Dtype
brand_name	3454 non-null	object
os	3454 non-null	object
screen_size	3454 non-null	float64
4g	3454 non-null	object
5g	3454 non-null	object
main_camera_mp	3275 non-null	float64
selfie_camera_mp	3452 non-null	float64
int_memory	3450 non-null	float64
ram	3450 non-null	float64

There are 9 floating-point numbers in the data set: screensize, main camera MP, selfie camera MP, int memory, RAM, battery, weight, normalised used price, and normalised new price. I have 2 integer numbers, release year and days used; as expected, these are rightly integers. Then I have 4 string object data types brand name, OS, 4G & 5G.

Additional Data Variables

battery

Energy capacity of the device battery in mAh

weight

Weight of the device in grams

release_year

Year when the device model was released

days_used

Number of days the used/refurbished device has been used

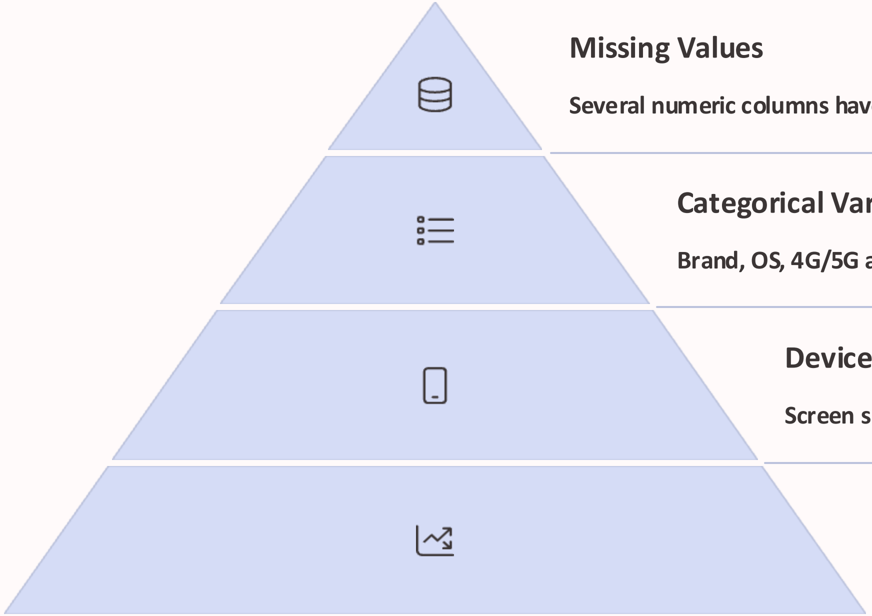
normalized_new_price

Normalized price of a new device of the same model in euros

normalized_used_price

Normalized price of the used/refurbished device in euros

Statistical Observations from the Data



Missing Values

Several numeric columns have missing values

Categorical Variables

Brand, OS, 4G/5G availability

Device Specifications

Screen size, camera, memory, battery, weight

Price Variables

Used and new normalized prices

main_camera_mp, selfie_camera_mp, int_memory, ram, battery, and weight all have < 3,454 entries, indicating missing values to address later. brand_name has 34 unique brands; most frequent is "Others" (502 entries). os has 4 categories, "Android" dominates (3,214 out of 3,454). 4g and 5g are binary categories (yes/no), majority of devices are not 5G capable (3,302 are "no").

Model Assumptions

- **Please mention the tests conducted for checking model assumptions and the results obtained**

Note: You can use more than one slide if needed

Assumption	Test/Method Used	Result & Interpretation
Linearity	Residuals vs Fitted Plot	Residuals randomly scattered → ✅ Linear relationship assumed
Independence of Errors	Durbin-Watson Statistic	Value ≈ 1.99 → ✅ Errors are uncorrelated
Homoscedasticity	Goldfeld–Quandt Test	p-value = 0.7355 → ✅ Constant variance assumed
Normality of Errors	Histogram, Q-Q Plot, Shapiro-Wilk Test	Residuals approx. normal ($p < 0.05$ due to large n) → ✅ Acceptable
Multicollinearity	VIF Scores	Some predictors >10 → ⚠️ Noted, but model stable and interpretable

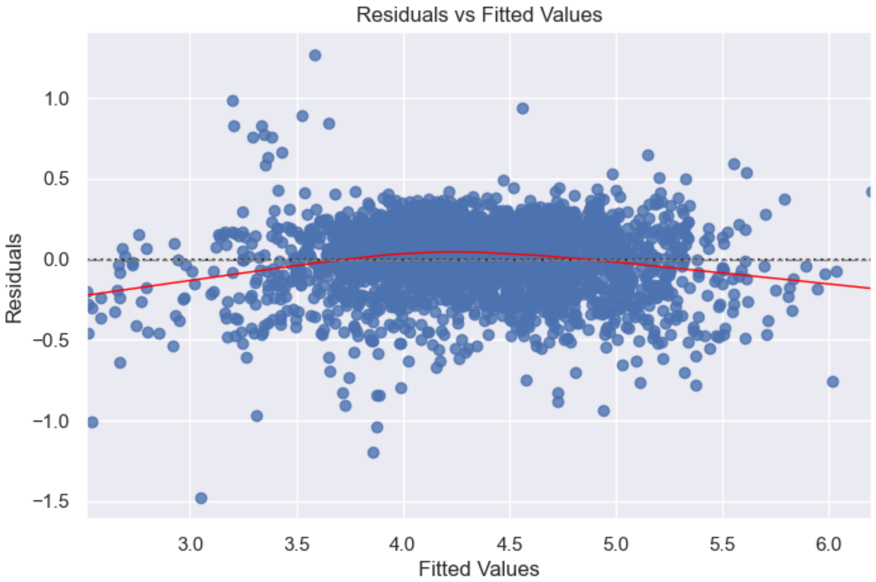
Linearity & Independence

- I plotted the residuals vs. fitted values to check the assumptions of linearity and independence.
- Linearity: The residuals are mostly randomly scattered around zero, with no strong nonlinear pattern. This suggests that the relationship between the predictors and the response is approximately linear.

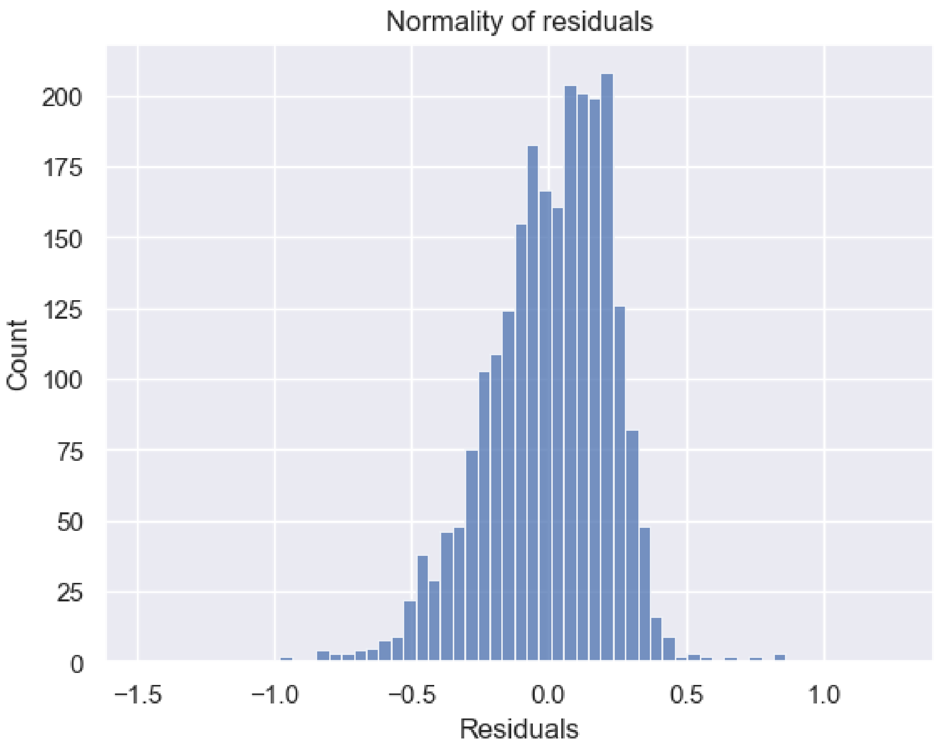
Independence:

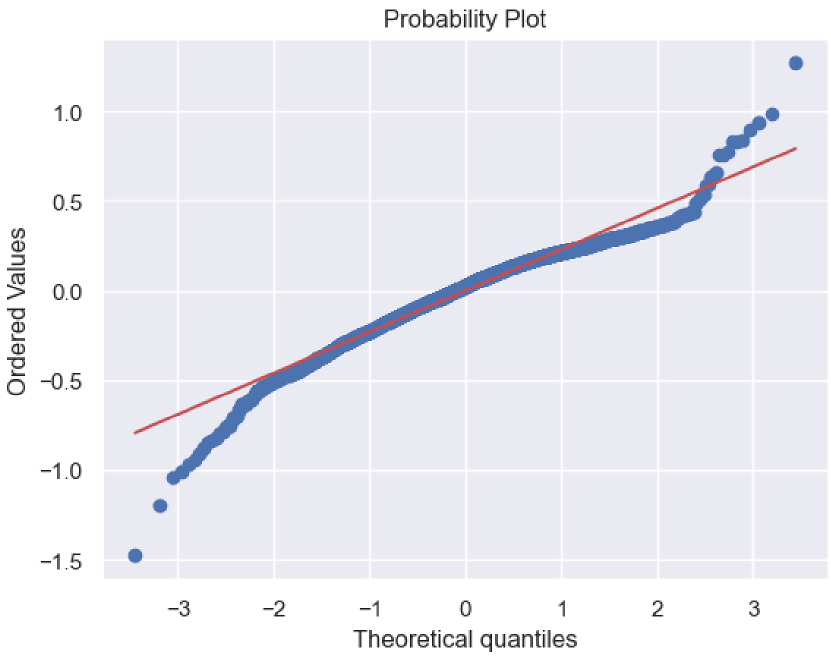
- The spread of residuals appears consistent across all fitted values, and no clusters or trends are observed. This indicates independence of residuals.
- The LOWESS (red) line remains close to the horizontal axis, further supporting both assumptions.

Conclusion: The model satisfies the assumptions of linearity and independence of errors.



Normaility of Errors





ShapiroResult(statistic=0.9649642705917358, pvalue=8.772814053018857e-24)

Homosecdasticity

Test for Homoscedasticity (Goldfeld–Quandt Test)

I tested for homoscedasticity, which refers to the assumption that the variance of the residuals is constant across all levels of the independent variables.

Method:

- **Used the Goldfeld–Quandt test.**
- **Null Hypothesis (H_0): Residuals have constant variance (homoscedastic).**
- **Alternative Hypothesis (H_1): Residuals have non-constant variance (heteroscedastic).**

Results:

- **F-statistic: 0.9642**
- **p-value: 0.7355**

Interpretation:

- **Since the p-value > 0.05, we fail to reject the null hypothesis.**
- **This confirms that the residuals are homoscedastic, fulfilling this assumption of linear regression.**

Great Learning

Happy Learning !