Replace outliers with mean

Wissaurt kanasub 6510545721

Impact of Outliers

Outliers affects

- **Mean**

Susceptible to outliers; can be distorted by extreme values, pulling it towards them and giving a false impression of central tendency.

- **Median**

Less affected by outliers; represents the middle value in a dataset and is not sensitive to extreme values, making it a better measure of the typical value, especially in datasets with outliers.

Outliers can distort relationships and trends between variables

- **Strength of Relationship**

Outliers with extreme values in both variables can exaggerate the perceived strength of their relationship.

- **Direction of Relationship**

Outliers can skew the perceived direction of the relationship, making it appear more positive or negative than it actually is.

Outliers can distort relationships and trends between variables

- **Regression Models**

  Outliers can significantly impact the slope and fit of regression lines, potentially leading to misleading interpretations.

- **Misleading Conclusions**

  Outliers, by their nature, do not represent the majority of the data, and conclusions drawn from data with outliers may be misleading as they might not reflect typical behavior or patterns.

Outliers effects on Models

Normal DistributionAssumption

Outliers can violate the assumption of normal distribution in statistical tests, leading to incorrect results.

Model Performance

Outliers can negatively impact the performance of statistical models like linear regression by causing them to focus excessively on extreme values.

Outliers effects on Models

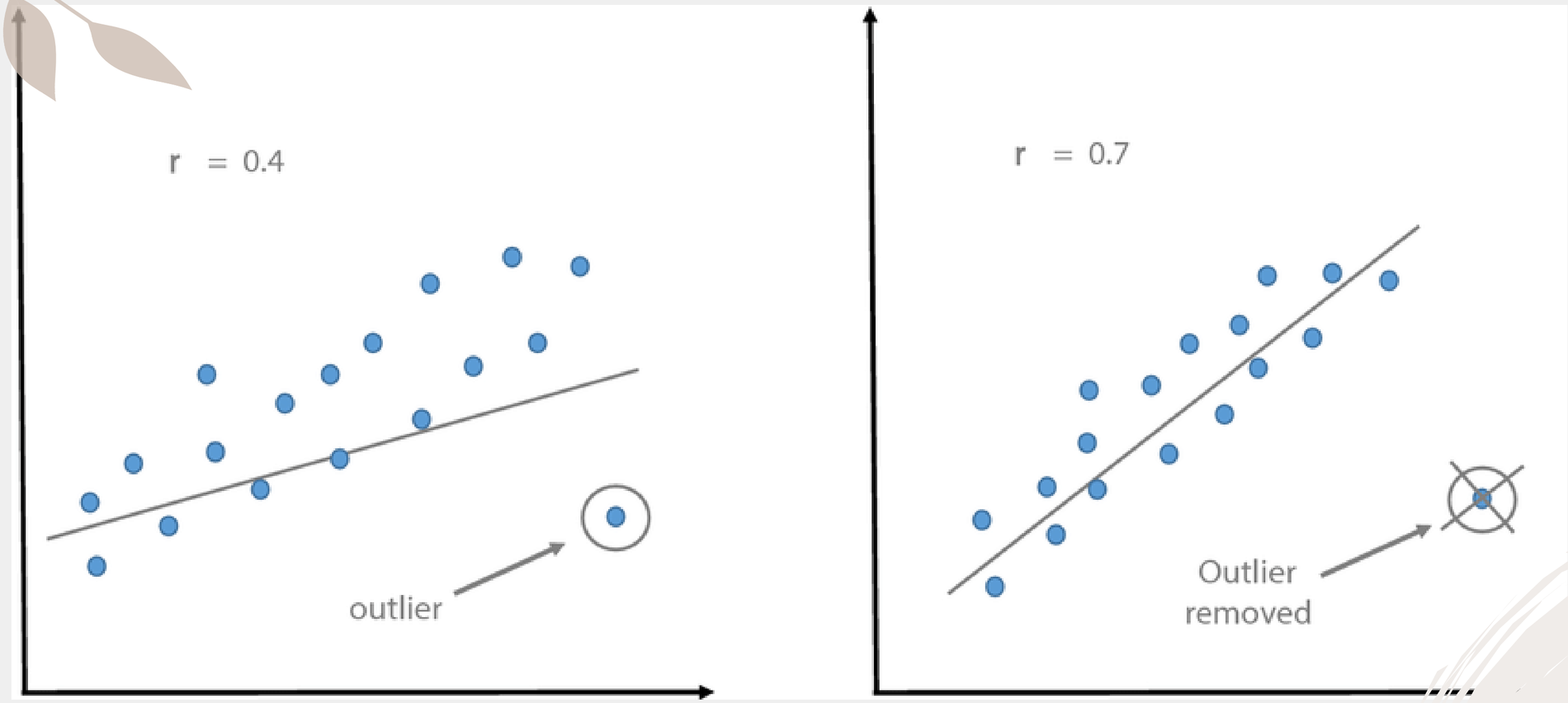
Overfitting

Outliers can contribute to overfitting in machine learning models, where the model becomes too complex and fits the training data too closely.

Bias

Outliers can introduce bias into machine learning algorithms, particularly those relying on distance metrics like k-means clustering, leading to inaccurate results.

Example



ref: reserchgate.net

Data soruce

Laptop Prices Dataset

Laptop prices for Regression practice

kaggle.com

Data exloration and preprocessing

Determine shape and data type of dataset(data source)

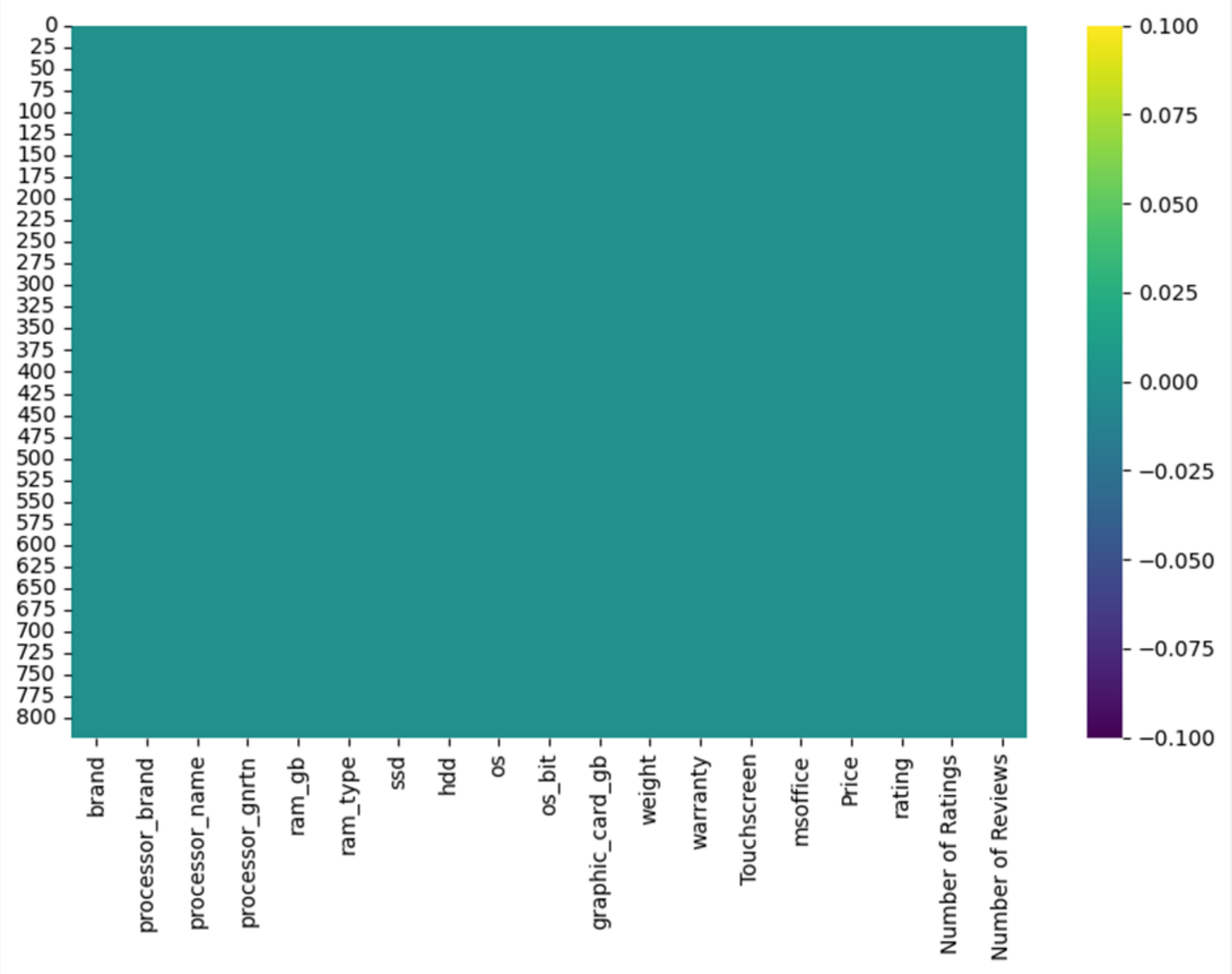
```
Data shape: (823, 19)
------------------------------
brand                object
processor_brand      object
processor_name       object
processor_gnrtn      object
ram_gb               object
ram_type             object
ssd                  object
hdd                  object
os                   object
os_bit               object
graphic_card_gb      object
weight               object
warranty             object
Touchscreen          object
msoffice             object
Price                 int64
rating               object
Number of Ratings     int64
Number of Reviews     int64
dtype: object
------------------------------
```

Data have 834 row
and 19 columns
fosllowing the picture.

Determine descriptive statistics

	Price	Number of Ratings	Number of Reviews
count	823.000000	823.000000	823.000000
mean	76745.177400	315.301337	37.609964
std	45101.790525	1047.382654	121.728017
min	16990.000000	0.000000	0.000000
25%	46095.000000	0.000000	0.000000
50%	64990.000000	17.000000	2.000000
75%	89636.000000	139.500000	18.000000
max	441990.000000	15279.000000	1947.000000

Determine null or missing values by using Heatmap.



Drop needless attributes

processor_name

Redundant if you have another column specifying the processor model.

processor_gnrtn

Generation hard to compare with other generations in other brands.

Drop needless attributes

os

Most devices likely have an OS, so this column adds little value unless OS specifics are crucial.

os_bit

Redundant as most modern systems are 64-bit; can be inferred or assumed.

Drop needless attributes

weight

Might not be relevant unless analyzing portability.

warranty

Relevant for reliability analysis but not crucial price analysis.

Drop needless attributes

msoffice

- Redundant if bundled software is covered elsewhere or not a key feature.

Number of Ratings, Number of Reviews,

- From original dataset even number of rating and review are 0 it still has a rating value.

Data after drop needless

	brand	processor_brand	ram_gb	ram_type	ssd	hdd	graphic_card_gb	Touchscreen	Price	rating
0	ASUS	Intel	4 GB	DDR4	0 GB	1024 GB	0 GB	No	34649	2 stars
1	Lenovo	Intel	4 GB	DDR4	0 GB	1024 GB	0 GB	No	38999	3 stars
2	Lenovo	Intel	4 GB	DDR4	0 GB	1024 GB	0 GB	No	39999	3 stars
3	ASUS	Intel	8 GB	DDR4	512 GB	0 GB	2 GB	No	69990	3 stars
4	ASUS	Intel	4 GB	DDR4	0 GB	512 GB	0 GB	No	26990	3 stars
...	...	...	...	...	...	...	...	...	...	...
818	ASUS	AMD	4 GB	DDR4	1024 GB	0 GB	0 GB	No	135990	3 stars
819	ASUS	AMD	4 GB	DDR4	1024 GB	0 GB	0 GB	No	144990	3 stars
820	ASUS	AMD	4 GB	DDR4	1024 GB	0 GB	4 GB	No	149990	3 stars
821	ASUS	AMD	4 GB	DDR4	1024 GB	0 GB	4 GB	No	142990	3 stars
822	Lenovo	AMD	8 GB	DDR4	512 GB	0 GB	0 GB	No	57490	4 stars

823 rows × 10 columns

(Dataset after drop needless columns)

shape: (823,10)

Explore data

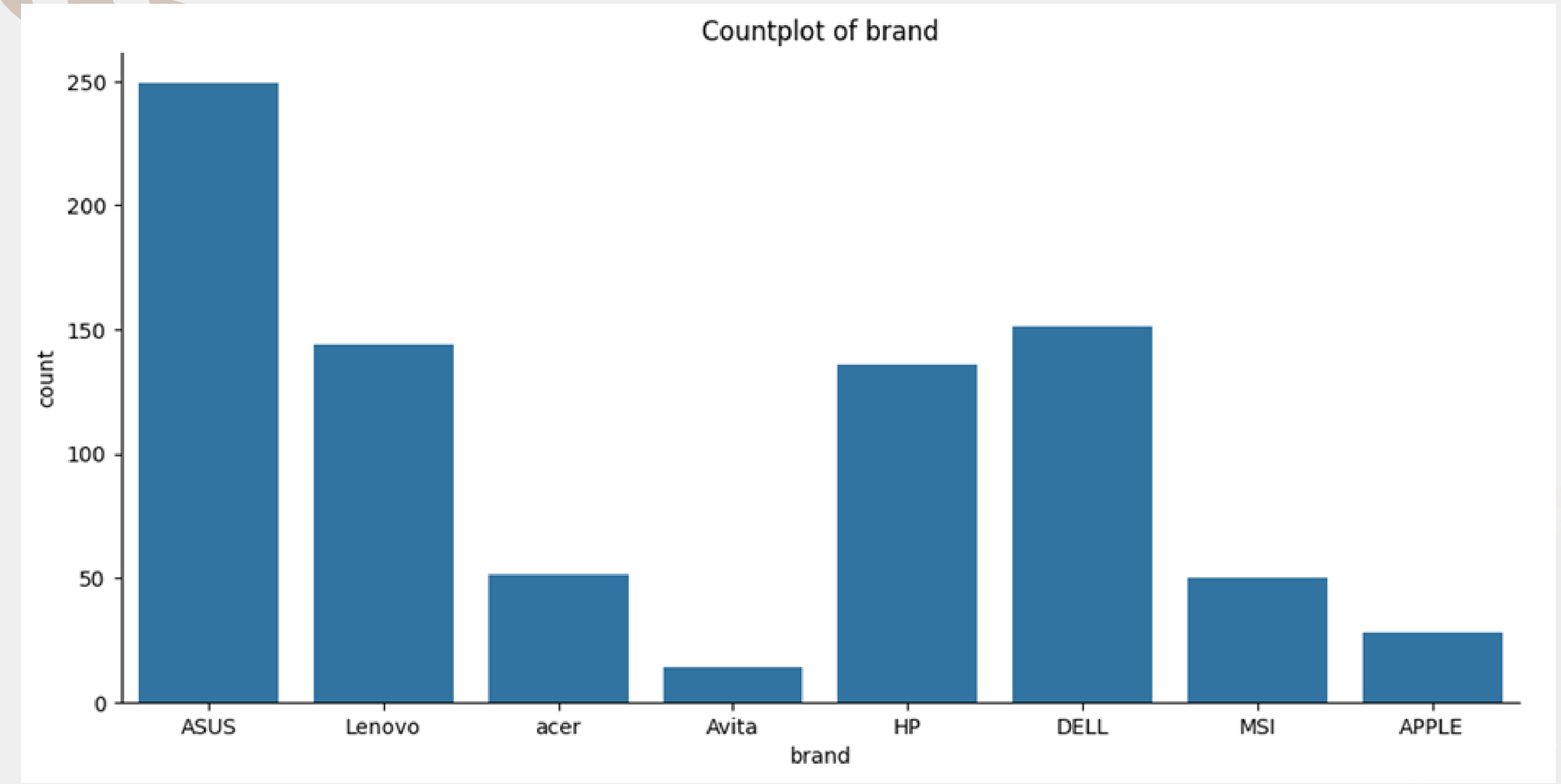
- **Categorical**

  Explore categorical data by using count plot.

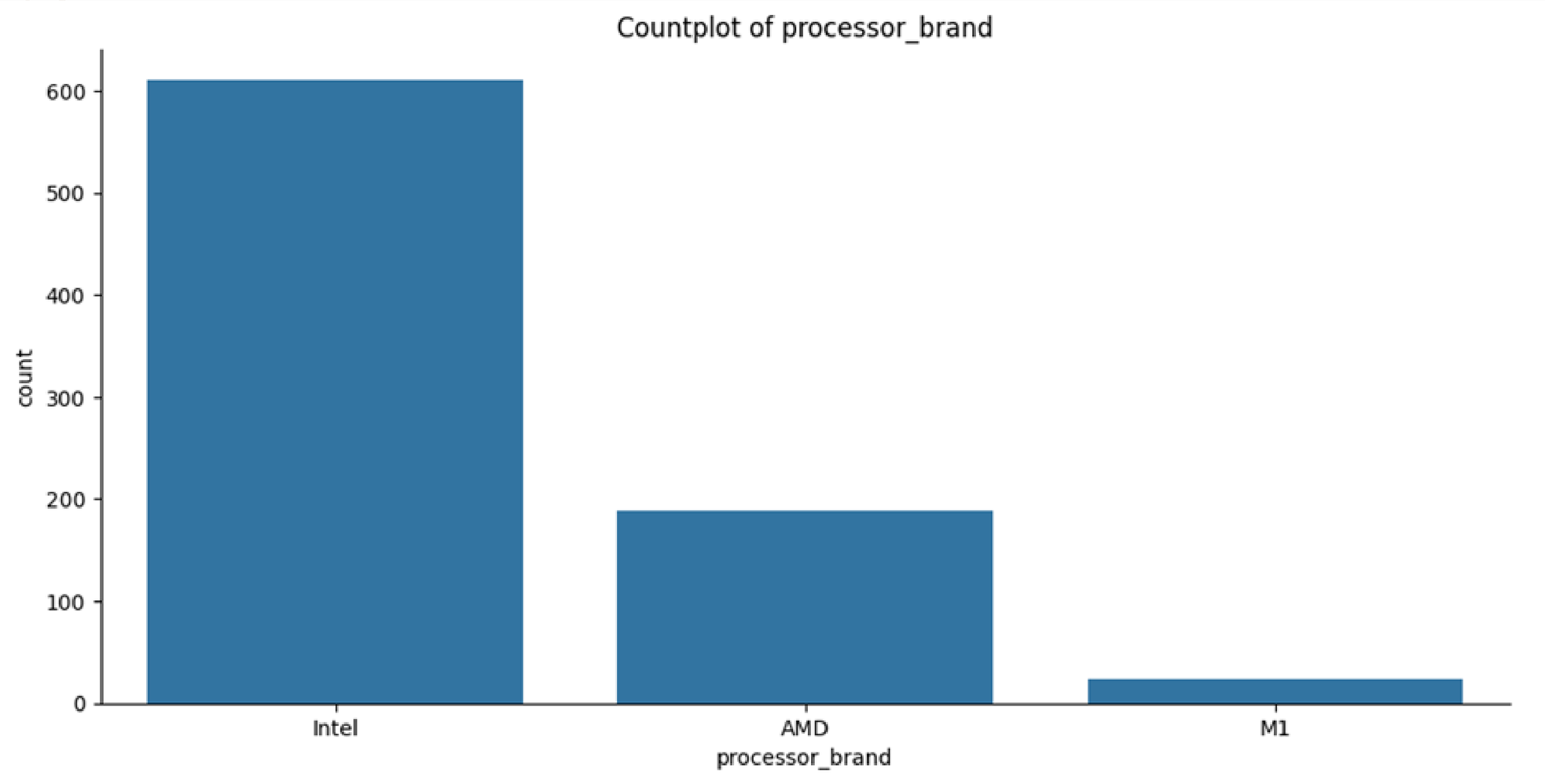
- **Numerical**

  Explore numerical data by using histogram plot to see their distribution

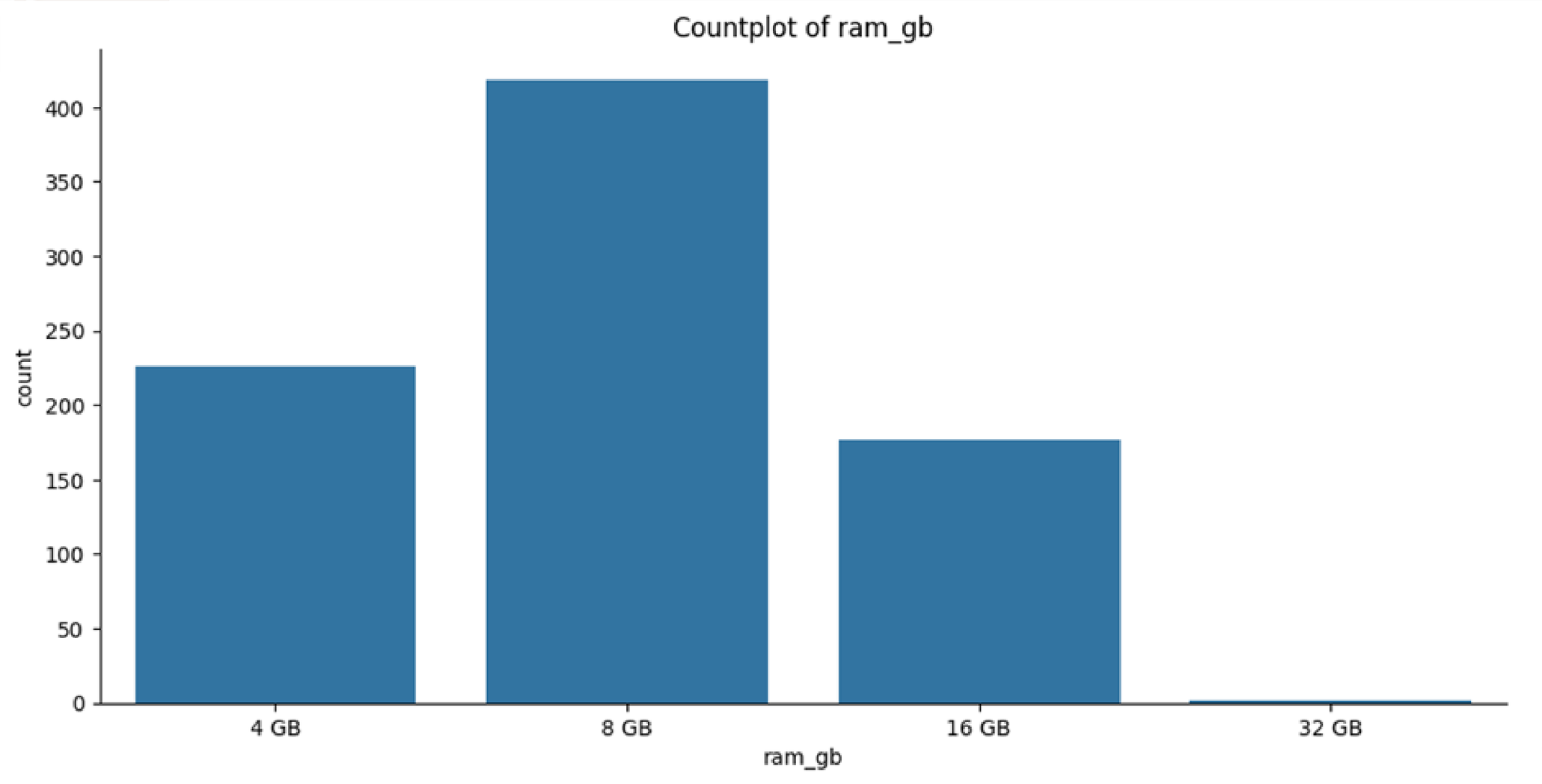
Brand


Countplot of brand
count
250
200
150
100
50
0
ASUS
Lenovo
acer
Avita
HP
DELL
MSI
APPLE
brand

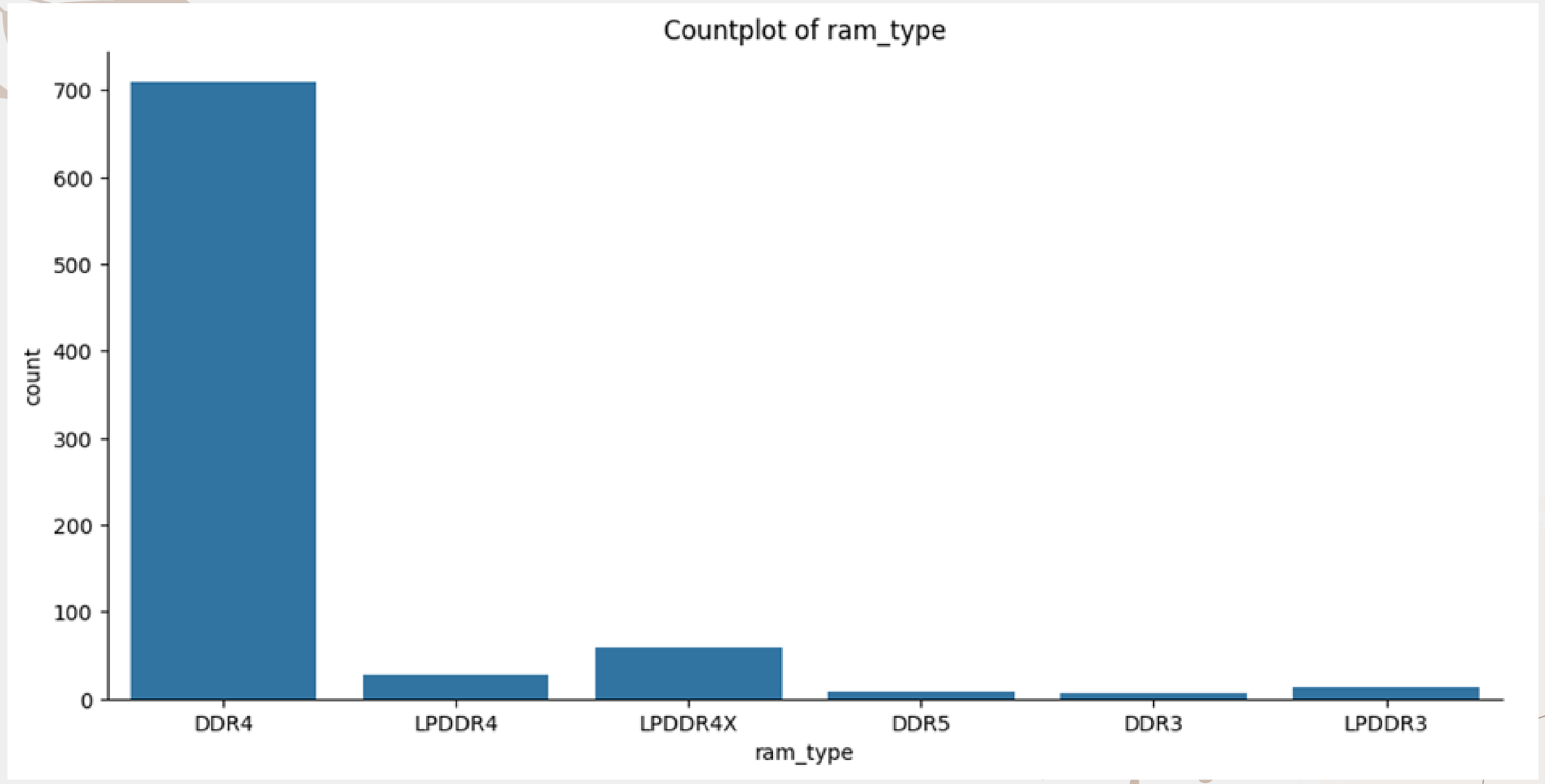
Processor Brand



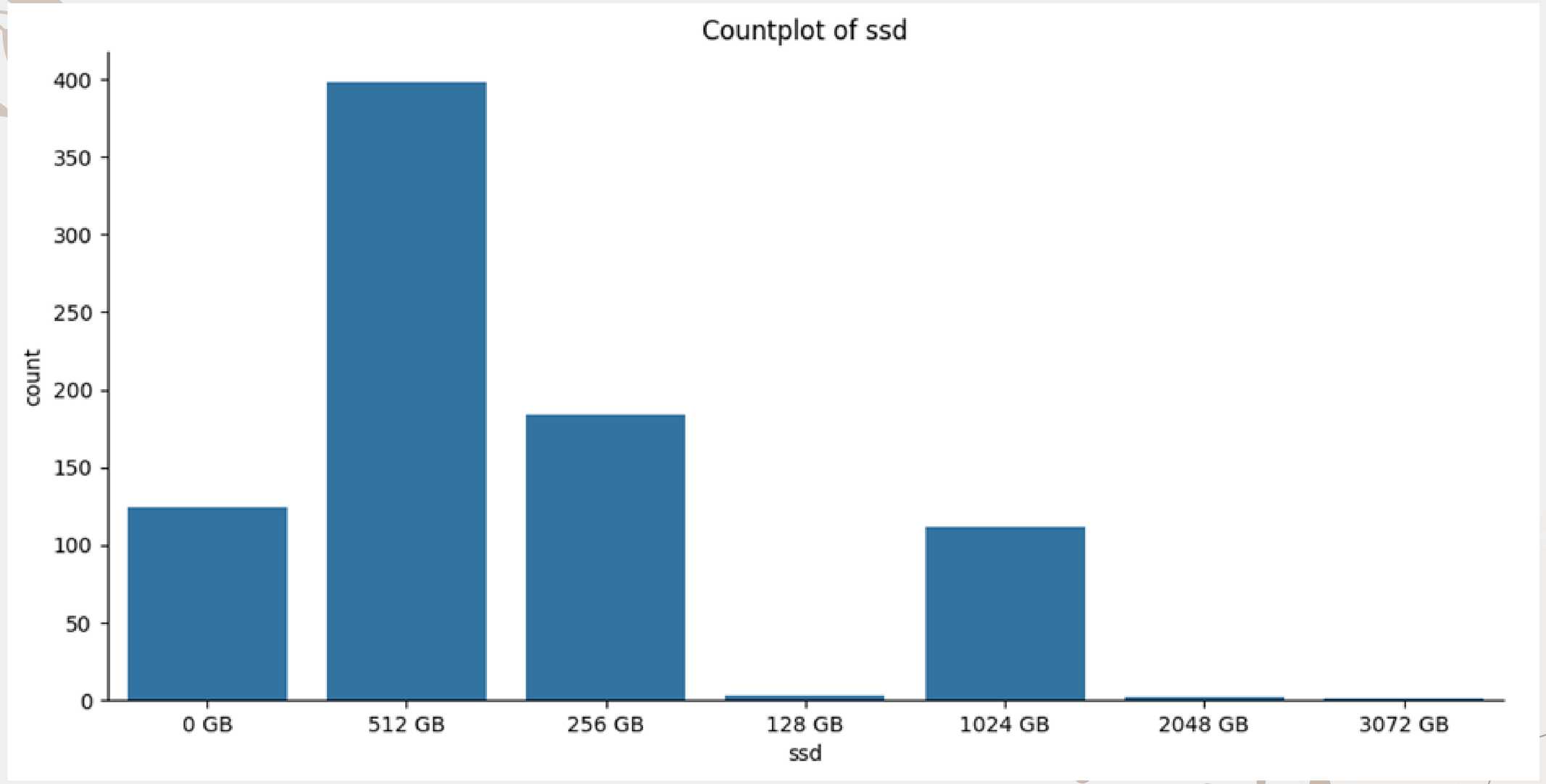
Ram_gb


Countplot of ram_gb
count
0
50
100
150
200
250
300
350
400
4 GB
8 GB
16 GB
32 GB
ram_gb

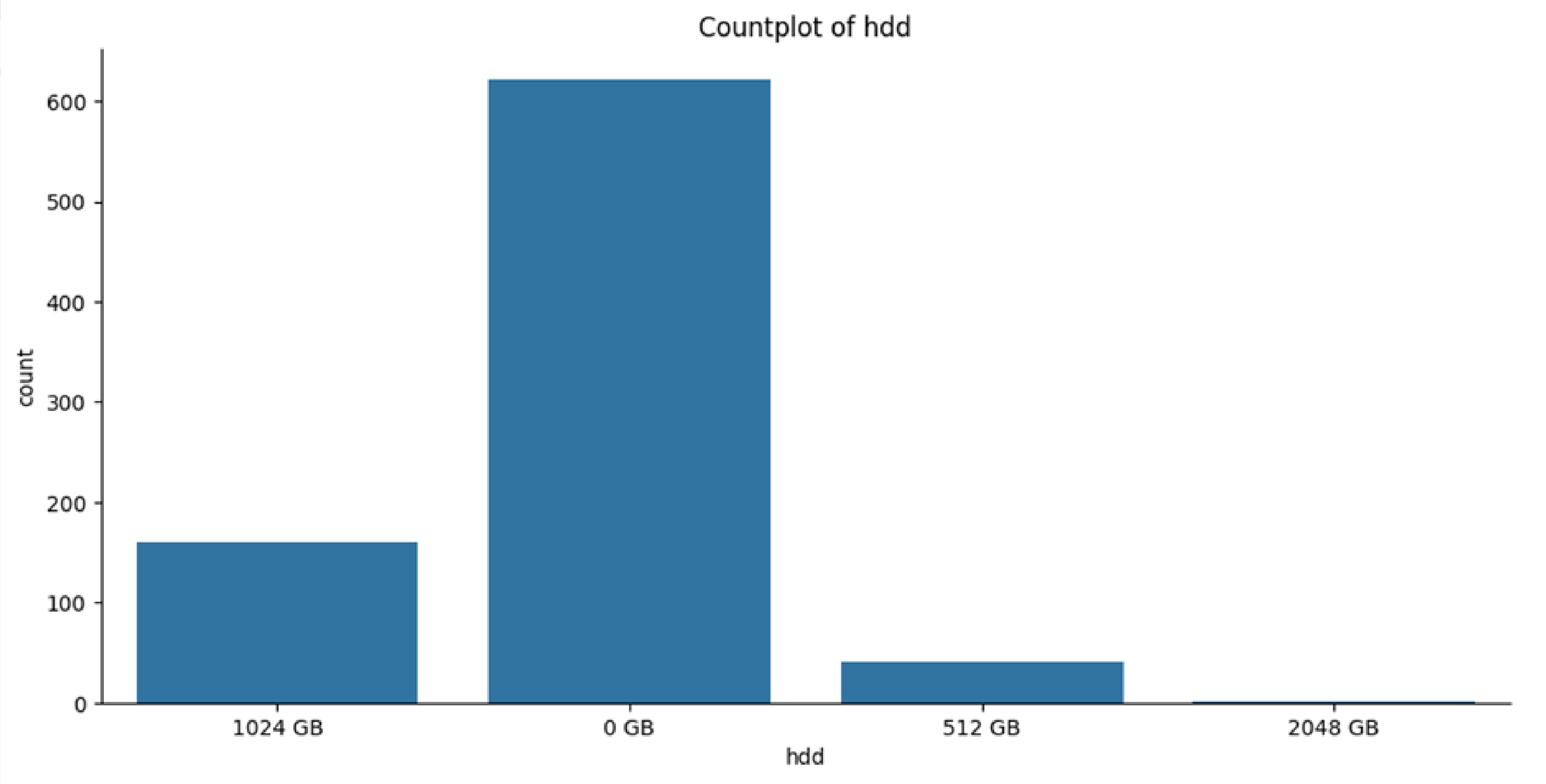
Ram Type


Countplot of ram_type
count
700
600
500
400
300
200
100
0
DDR4
LPDDR4
LPDDR4X
DDR5
DDR3
LPDDR3
ram_type

SSD


Countplot of ssd
count
400
350
300
250
200
150
100
50
0
0 GB
512 GB
256 GB
128 GB
1024 GB
2048 GB
3072 GB
ssd

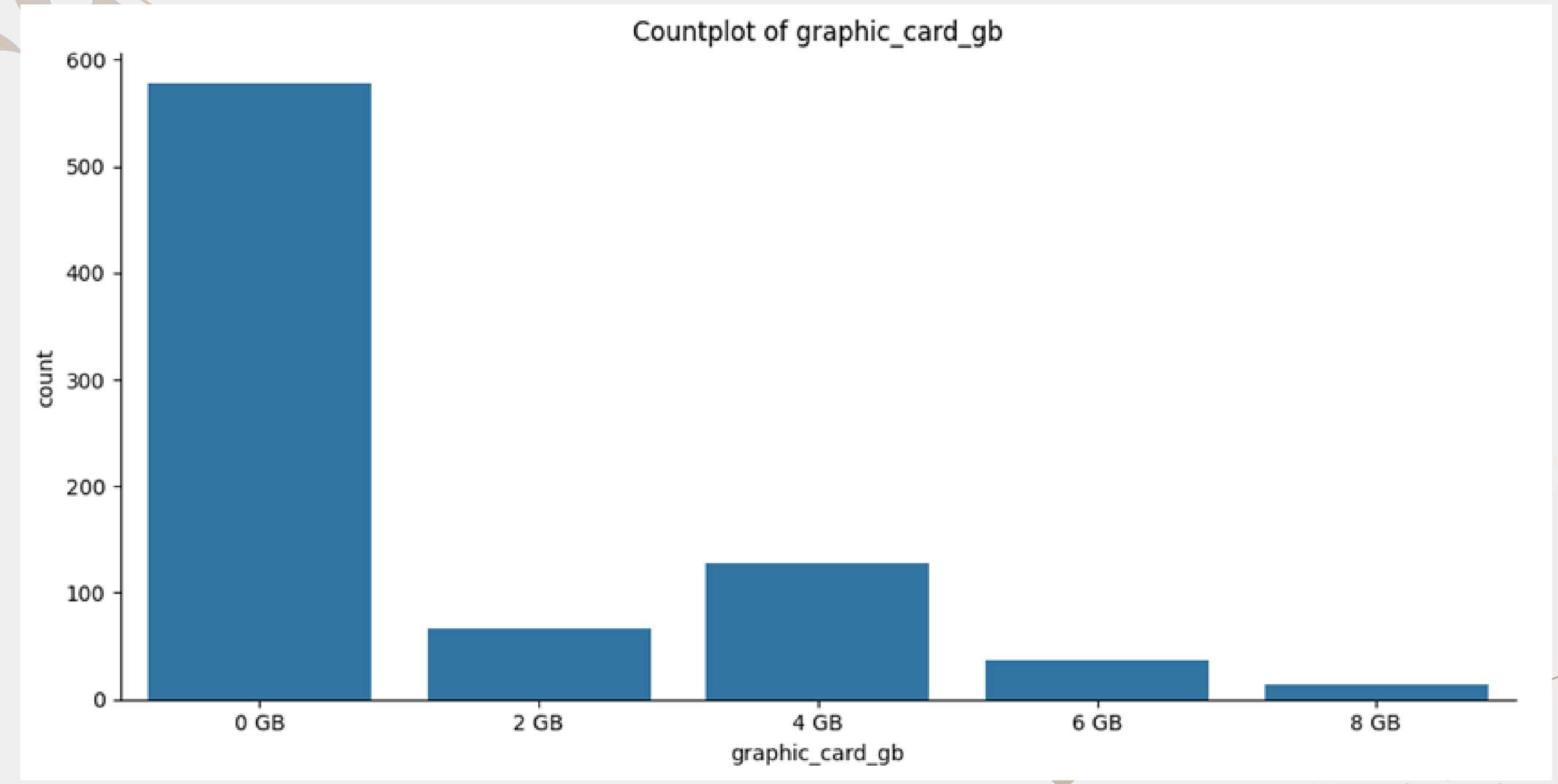
HDD


Countplot of hdd
count
600
500
400
300
200
100
0
1024 GB
0 GB
512 GB
2048 GB
hdd

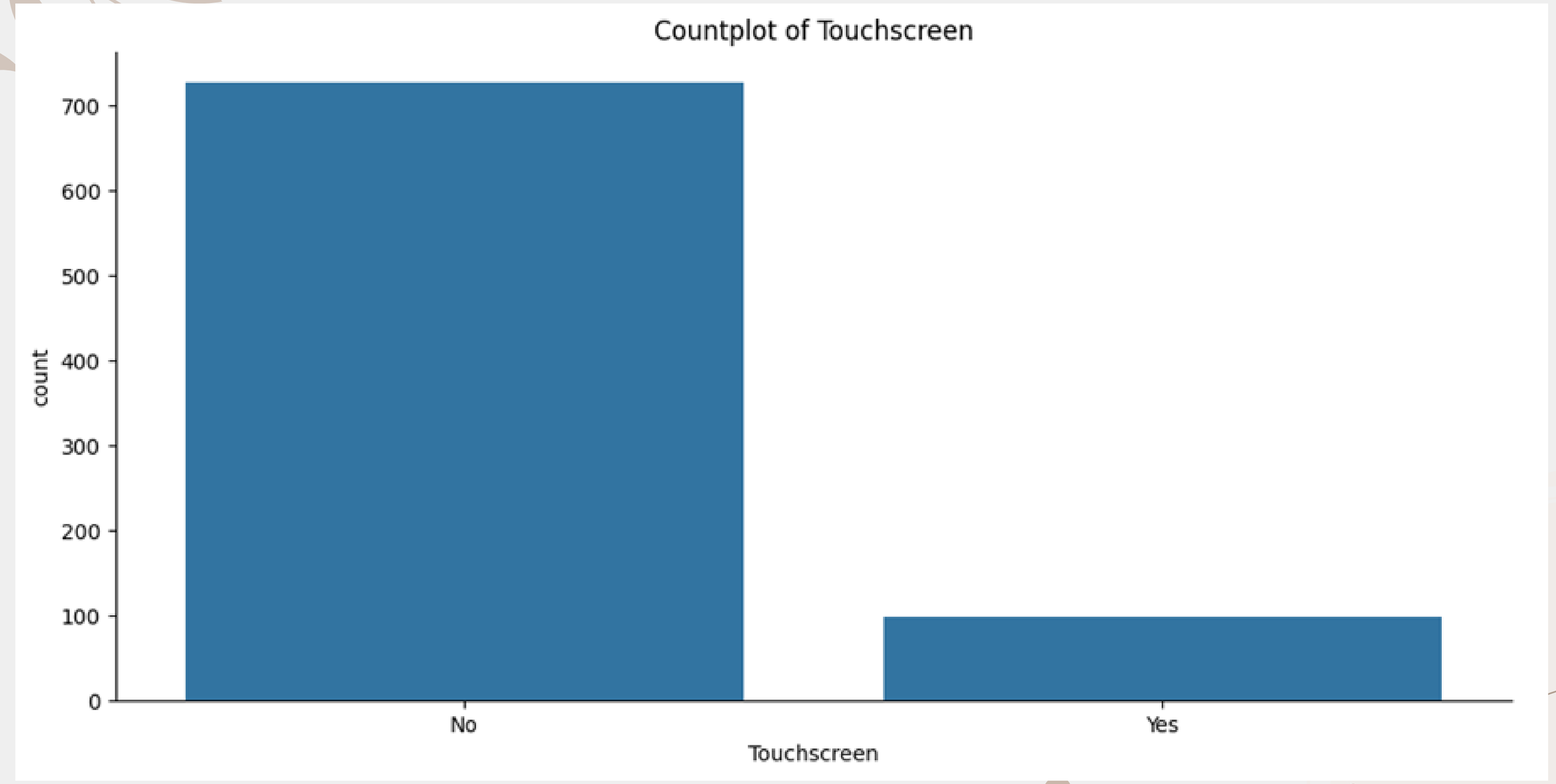
SSD and HDD noticed

From both the SSD and HDD count plots, we will notice that the highest value for HDD is often 0 GB. This suggests that many laptops in the dataset are using SSDs instead of HDDs.

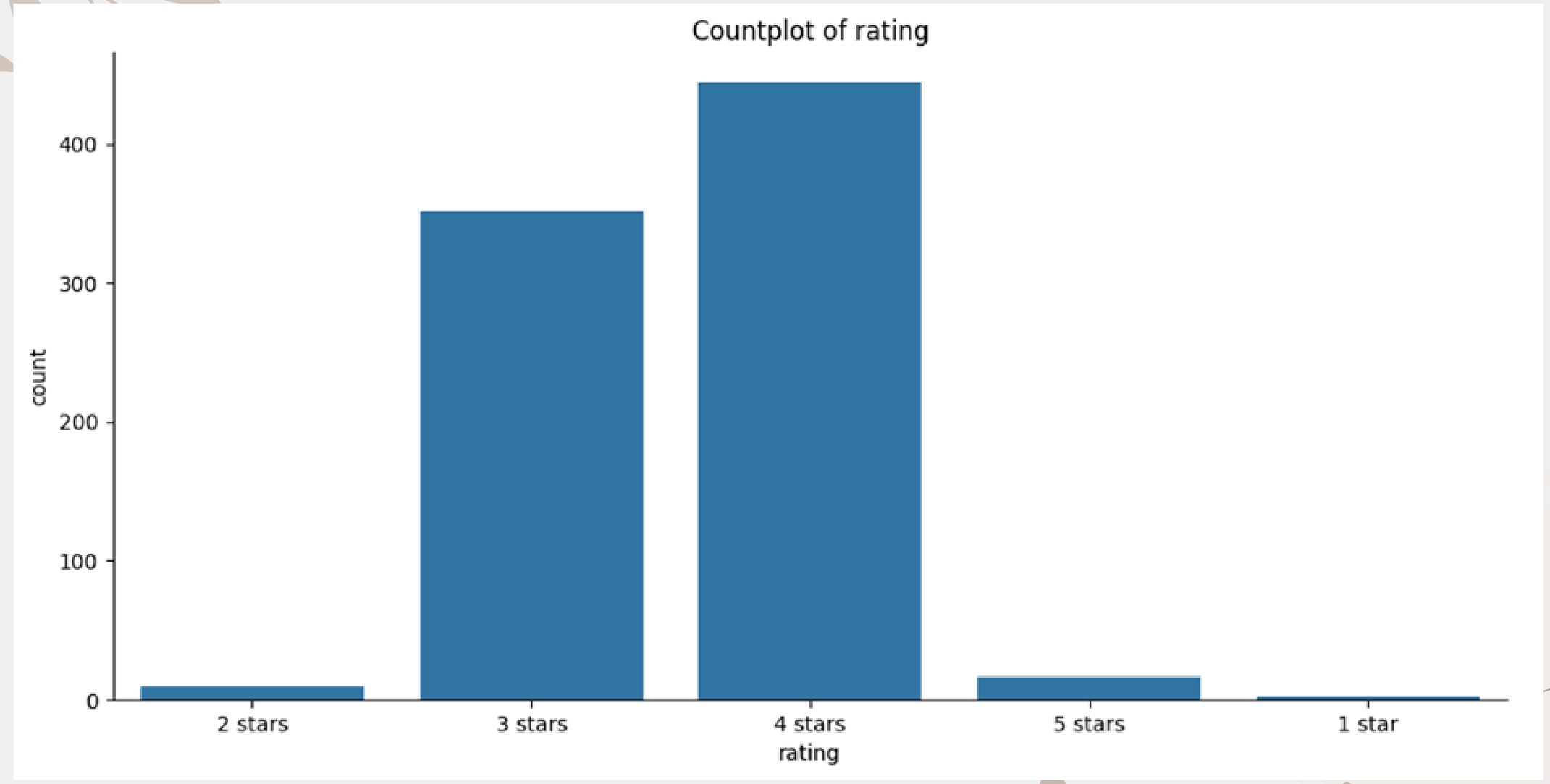
Graphic_card_gb (Vram)



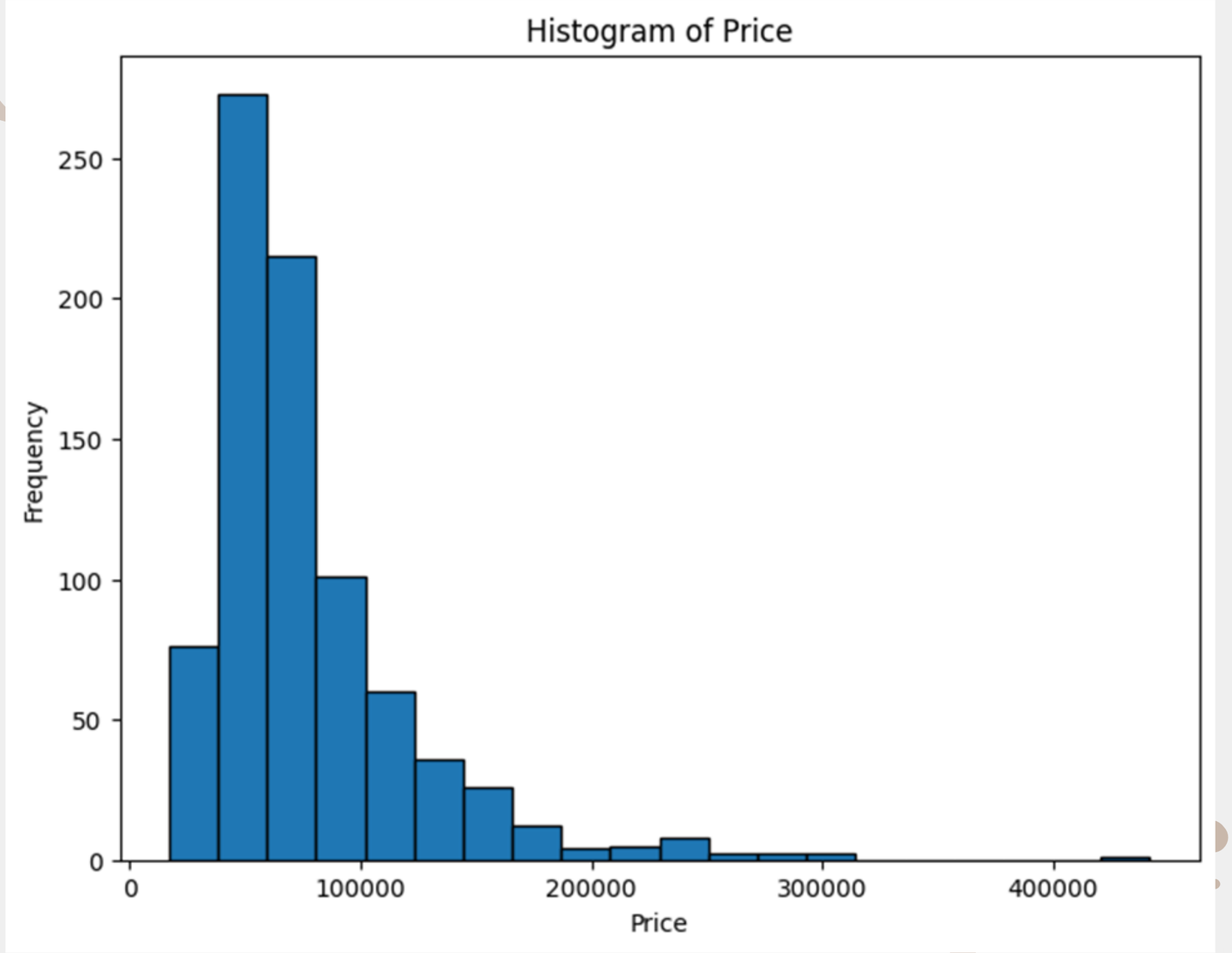
Touchscreen



Rating


Countplot of rating
count
400
300
200
100
0
2 stars
3 stars
4 stars
5 stars
1 star
rating

Numerical attributes (Price)



Price distribution

From histogram we noticed that it has right-skewed distributions.

Convert categorial to numerical

From exploring categorical data some of the data is numeric but it represents its string or text form. So we need to convert it to numeric

Data after converted

	brand	processor_brand	ram_gb	ram_type	ssd	hdd	graphic_card_gb	Touchscreen	Price	rating
0	ASUS	Intel	4.0	DDR4	0.0	1024.0	0.0	No	34649	2.0
1	Lenovo	Intel	4.0	DDR4	0.0	1024.0	0.0	No	38999	3.0
2	Lenovo	Intel	4.0	DDR4	0.0	1024.0	0.0	No	39999	3.0
3	ASUS	Intel	8.0	DDR4	512.0	0.0	2.0	No	69990	3.0
4	ASUS	Intel	4.0	DDR4	0.0	512.0	0.0	No	26990	3.0
...	...	...	...	...	...	...	...	...	...	...
818	ASUS	AMD	4.0	DDR4	1024.0	0.0	0.0	No	135990	3.0
819	ASUS	AMD	4.0	DDR4	1024.0	0.0	0.0	No	144990	3.0
820	ASUS	AMD	4.0	DDR4	1024.0	0.0	4.0	No	149990	3.0
821	ASUS	AMD	4.0	DDR4	1024.0	0.0	4.0	No	142990	3.0
822	Lenovo	AMD	8.0	DDR4	512.0	0.0	0.0	No	57490	4.0

823 rows × 10 columns

Using LabelEncoder.

Encode some columns to numeric to improve modeling performance in the future.

```python
# Convert categorical to numberical with Label Encoder. This use only Unique data.
to_convert_column = ['brand', 'processor_brand', 'ram_type', 'Touchscreen']
label_encoder = preprocessing.LabelEncoder() # Create object.

for col in to_convert_column:
    original_data[col] = label_encoder.fit_transform(original_data[col])
    display(f"{col}: {dict(zip(label_encoder.classes_, label_encoder.transform(label_encoder.classes_)))}")

display(original_data)
```

After Using LabelEncoder.

	brand	processor_brand	ram_gb	ram_type	ssd	hdd	graphic_card_gb	Touchscreen	Price	rating
0	1	1	4.0	1	0.0	1024.0	0.0	0	34649	2.0
1	5	1	4.0	1	0.0	1024.0	0.0	0	38999	3.0
2	5	1	4.0	1	0.0	1024.0	0.0	0	39999	3.0
3	1	1	8.0	1	512.0	0.0	2.0	0	69990	3.0
4	1	1	4.0	1	0.0	512.0	0.0	0	26990	3.0
...	...	...	...	...	...	...	...	...	...	...
818	1	0	4.0	1	1024.0	0.0	0.0	0	135990	3.0
819	1	0	4.0	1	1024.0	0.0	0.0	0	144990	3.0
820	1	0	4.0	1	1024.0	0.0	4.0	0	149990	3.0
821	1	0	4.0	1	1024.0	0.0	4.0	0	142990	3.0
822	5	0	8.0	1	512.0	0.0	0.0	0	57490	4.0

823 rows × 10 columns

Brands	
Apple	0
Asus	1
Avita	2
Dell	3
HP	4

Lenovo	5
MSI	6
acer	7

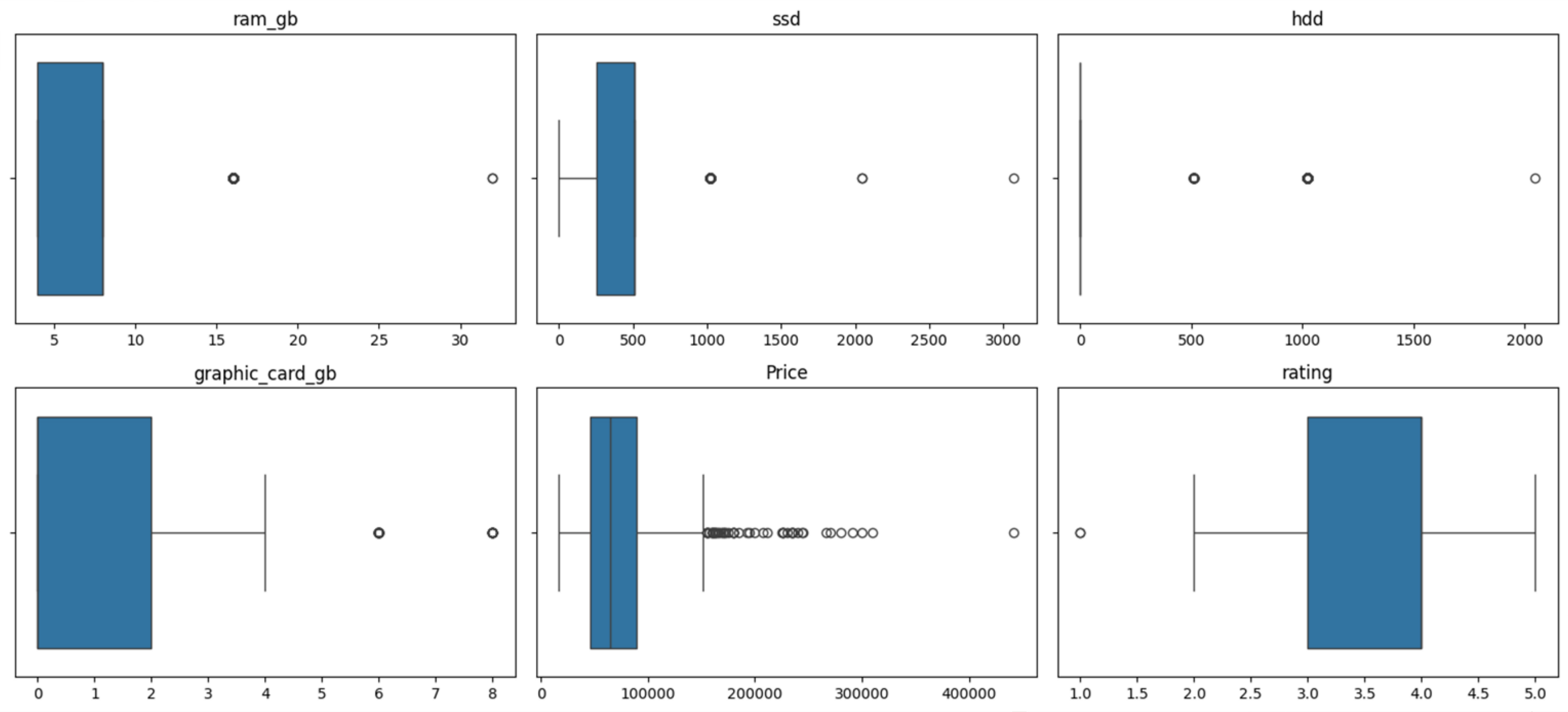
Processor Brands

AMD	0
Intell	1
M1	2

Ram Type	
DDR3	0
DDR4	1
DDR5	2
LPDDR3	3
LPDDR4	4
LPDDR4X	5

Tochscreen	
NO	0
YES	1

Using Boxplot to finding outliers



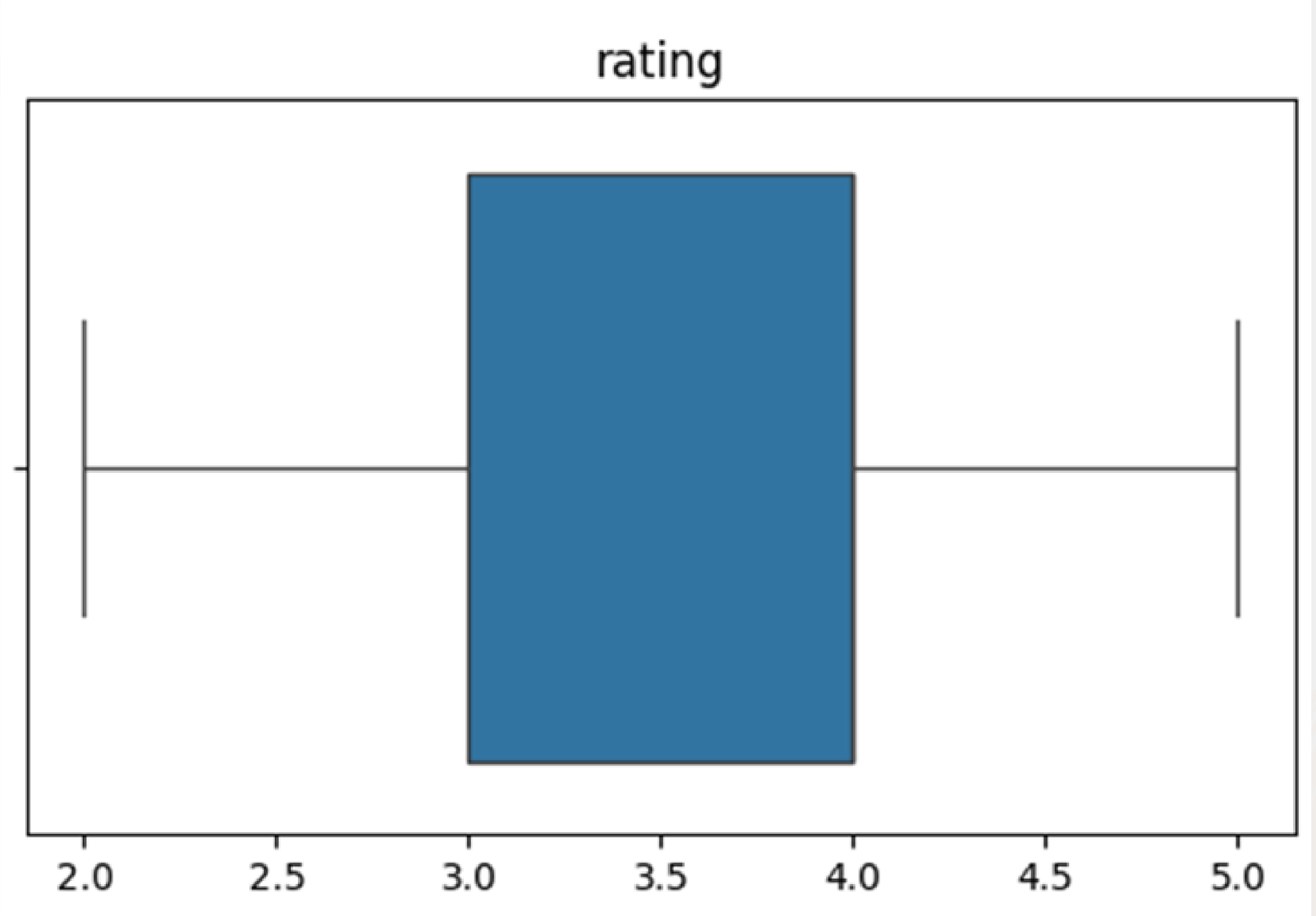
The Outliers

From the plot below we see that there are columns containing outliers.

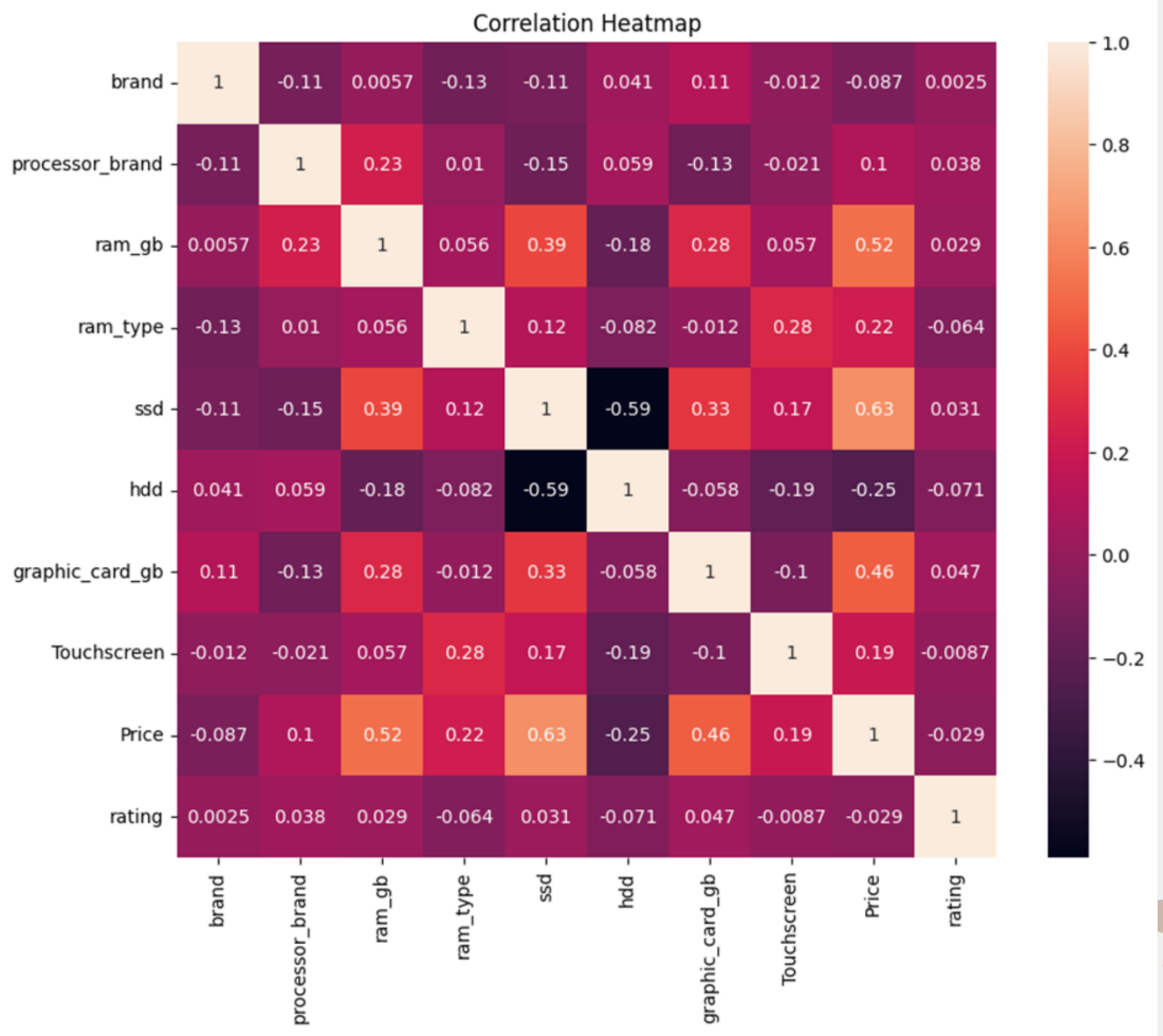
I need to replace outliers with the mode only for the 'Rating' attribute because I believe that other numerical attributes, such as RAM memory, VGA capacity, HDD size, and SSD storage, are crucial factors affecting the price of laptops in real-world scenarios. This decision was made to ensure that the model's accuracy remains high and reflects the significant impact of these hardware specifications on laptop pricing.

I need to replace outliers with mean (Technique that we interested) on column “Price”

After replace rating's outliers with mode.

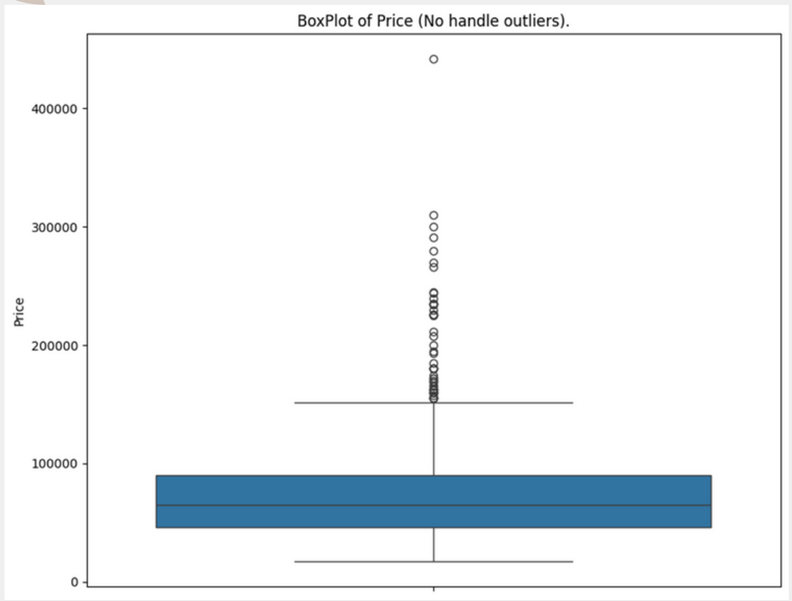


Find correlation by heatmap

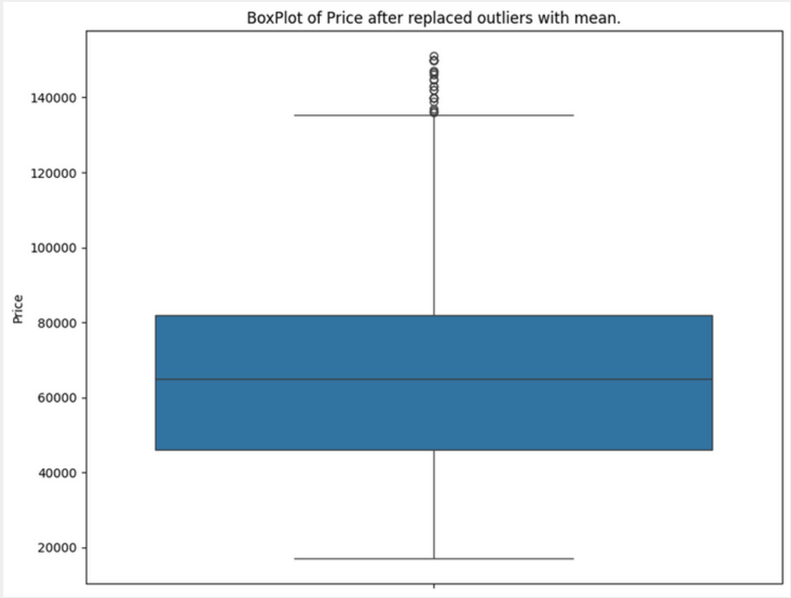


Prepare 3 data for the modeling part.

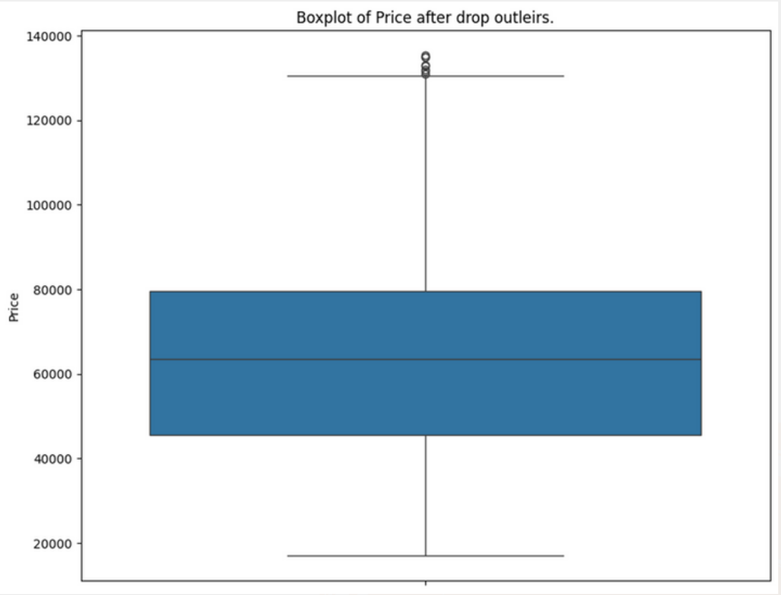
no handle outliers



Replace outliers with mean



drop outliers



Pre modeling

We will use these 3 data sets for training models and evalu
compare the model.

Both techniques I only handle outliers for one time because Replacing outliers' multiple times can lead to distortion of the data and potentially skew the distribution.

Training a models Using regrssion

Linear regression

Detail 1

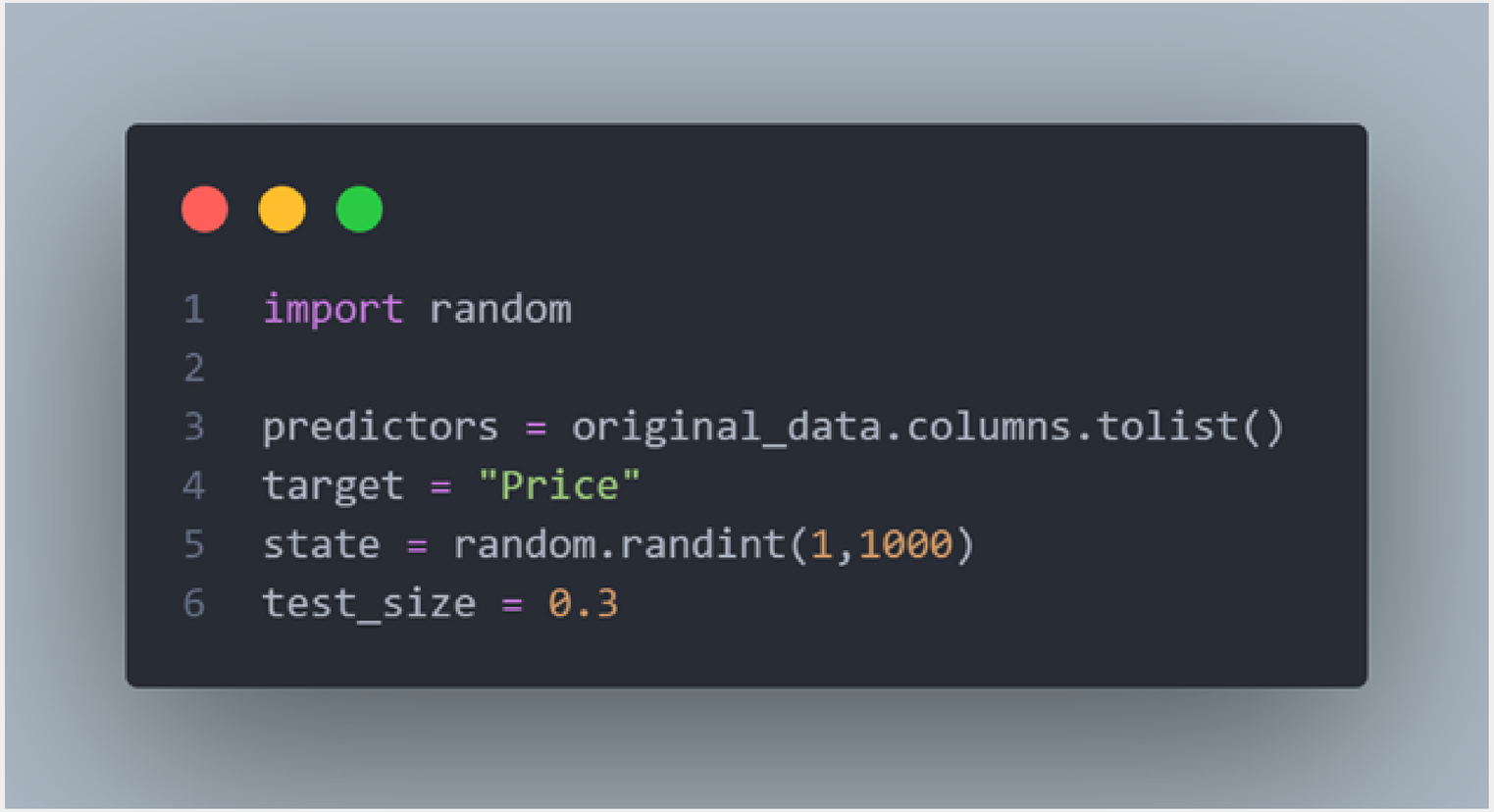
We will use a linear regression model to predict the "Price" of laptops. The predictor attributes used for modeling are consistent across all datasets.

Detail 2

The three datasets, each processed differently with respect to handling outliers, will be utilized to assess the impact of outlier handling techniques on the performance of the linear regression model in predicting laptop prices.

Setup modeling constant

- Predictor is all attributes
- Test size 0.3
- Target is “Price”
- Same Random state


import random
predictors = original_data.columns.tolist()
target = "Price"
state = random.randint(1,1000)
test_size = 0.3

Function to evaluate the the model

```python
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score, mean_absolute_percentage_error

def evaluate_linear_regression(model, X_test, y_test):
    # Make predictions
    predictions = model.predict(X_test)

    # Print the linear regression equation
    print("Linear Regression Equation:")
    print("y =", model.intercept_, end=" ")
    for i, coef in enumerate(model.coef_):
        print("+", coef, "* X" + str(i+1), end=" ")
    print()

    # Evaluate the model
    mae = mean_absolute_error(y_test, predictions)
    r2 = r2_score(y_test, predictions)
    mape = mean_absolute_percentage_error(y_test, predictions)

    # Print evaluation metrics
    print("\nEvaluation Metrics:")
    print(f"Mean Absolute Error (MAE): {mae}")
    print(f"Mean Absolute Percentage Error (MAPE): {mape}%")
    print(f"R-squared (R2) Score: {r2}")
```

Evaluate the model

Evaluate technique

- **MAE**

Average of the absolute differences between predicted and actual values. It measures the average magnitude of errors.

- **MAPE**

Average of the absolute percentage differences between predicted and actual values. It measures the average magnitude of percentage errors.

Evaluate technique

- **R2 Score**

Proportion of variance in the dependent variable explained by the independent variables in the model. It ranges from 0 to 1, with higher values indicating better fit.

No handle outliers data set

Equation:

y = 76285.80 - (1532.06 * X1) + (6602.07 * X2) + (8115.63 * X3) + (6691.74 * X4) + (23009.13 * X5) + (5577.03 * X6) + (14730.81 * X7) + (4409.49 * X8) - (2727.68 * X9)

Mean Absolute Error **(MAE)**	**21007.95**
Mean Absolute Percentage Error **(MAPE)**	**0.276**
R-squared **(R2)** Score	**0.530**

Replace outliers with mean

Equation:

y = 68545.19 - (921.19 * X1) + (393.18 * X2) + (5010.77 * X3) + (5020.03 * X4) + (6424.01 * X5) - (1514.39 * X6) + (8690.12 * X7) + (5438.34 * X8) - (2430.38 * X9)

Mean Absolute Error **(MAE)**	**15668.13**
Mean Absolute Percentage Error **(MAPE)**	**0.250**
R-squared **(R2)** Score	**0.448**

Drop outliers

Equation:

y = 66372.31 - (1462.28 * X1) - (97.06 * X2) + (6769.80 * X3) + (3908.28 * X4) + (5775.34 * X5) - (1434.02 * X6) + (5669.88 * X7) + (5401.64 * X8) - (1344.15 * X9)

Mean Absolute Error **(MAE)**	**14043.48**
Mean Absolute Percentage Error **(MAPE)**	**0.232**
R-squared **(R2)** Score	**0.428**

Evaluate the model

outliers Replace with mean vs No handling

●	Replace outlier with mean can reduce MAE and MAPE
●	Replace outlier with mean tend to increase R-squared (R2)score

Conclusion : Replacing outliers with the mean tends to reduce MAE and MAPE because it brings extreme values closer to the central majority. However, it often decreases the R-squared score because it flattens the data distribution, compromising the model's ability to capture true variability.

outliers Replace with mean vs drop outliers

- Replace outliers with mean tend to have MAPE value larger than MAPE of data which drop rows of outliers.
- But Replace outliers with mean tend to have R2 score larger than R2 score of data which drop rows of outliers.

Conclusion : Replacing outliers with the mean may inflate the MAPE due to distortion of the data distribution, it can improve the R-squared score by preserving more data points and providing a better representation of the overall trend in the data.

Handle outliers vs no Handle outliers

Conclusion : No matter what techniques are used in managing outliers It will result in the model's predictions being more accurate.

Conclusion technique.

Replace outliers with mean

Benefits consider

- **Impact Reduction**

  Replacing outliers with the mean can indeed help mitigate the influence of extreme values on statistical measures such as the mean and standard deviation.

- **Maintaining Data Structure**

  Imputing with the mean does indeed preserve the overall structure of the dataset.

Benefits consider

- **Potential Biases**

However, imputing with the mean can introduce biases, particularly if the outliers represent genuine patterns in the data.

- **Sensitive to Outliers**

This method doesn’t eliminate the impact of outliers; instead, it redistributes their influence across the dataset.

Cons consider

- **Impact Reduction**

Replacing outliers with the mean can distort the original distribution of the data. By assigning extreme values to the mean, the representation of the data's spread and variability may become skewed, leading to potential misinterpretation of the dataset's characteristics.

- **Loss of Information**

Outliers can sometimes carry valuable information or signal about unique patterns or events in the data.

Cons consider

Dataset : Laptop Prices Dataset (kaggle.com)

Definition of Outliers:
Navigating Outliers for Accurate Data Analysis & Decisions (statusneo.com)

Sklearn metrics:
บันทึก training data science EP 4: Scikit-learn & Linear Regression – แนวโน้มของเส้นตรง (bluebirz.net)

Pandas: https://pandas.pydata.org/

Thank you