Laptop Price Prediction Report

Objective

The objective of this project is to develop a machine learning model that accurately predicts the **price of laptops** based on their specifications. The model aims to help users understand which features significantly influence laptop pricing.

Step-by-Step Methodology

1. **Data Loading**:

- Loaded the dataset into a pandas DataFrame.

2. **Data Cleaning**:

- Removed units (e.g., "GB" in RAM, "kg" in Weight).
- Converted binary features (Touchscreen, IPSpanel, RetinaDisplay) to numerical format.
- Handled missing values.
- Standardized categorical and numerical data types.

3. **Feature Engineering**:

- Extracted screen pixel count from ScreenW and ScreenH.
- Simplified text-heavy features like Product, CPU_model, GPU_model.

4. **Encoding Categorical Data**:

- Applied One-Hot Encoding to categorical features like Company, TypeName, OS, etc.

5. **Splitting Data**:

- Used train-test split (80-20) to evaluate model performance on unseen data.

6. **Model Training**:

- Trained a **Linear Regression** model using scikit-learn.

7. **Evaluation**:

- Assessed the model using **Mean Squared Error (MSE)** and **R-squared (R^2)**.

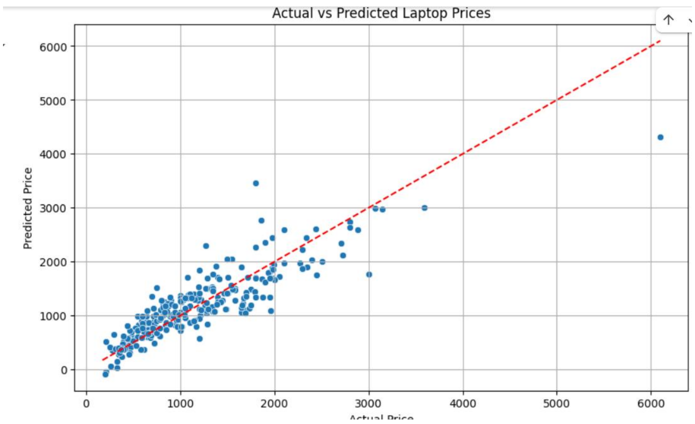
8. **Visualization**:

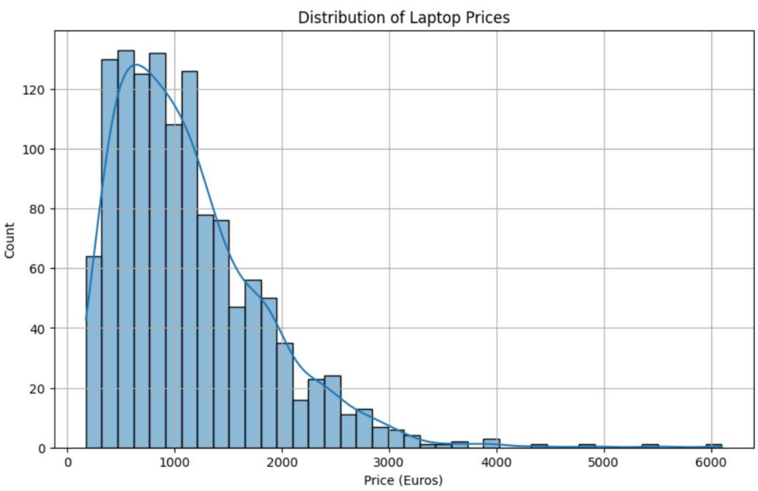
- Plotted price distribution, relationships between price and features like RAM, screen resolution, company, etc.

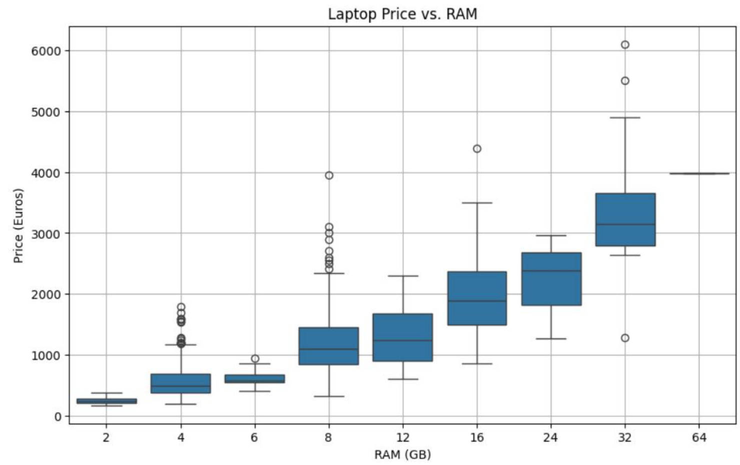
Code Summary and Explanation

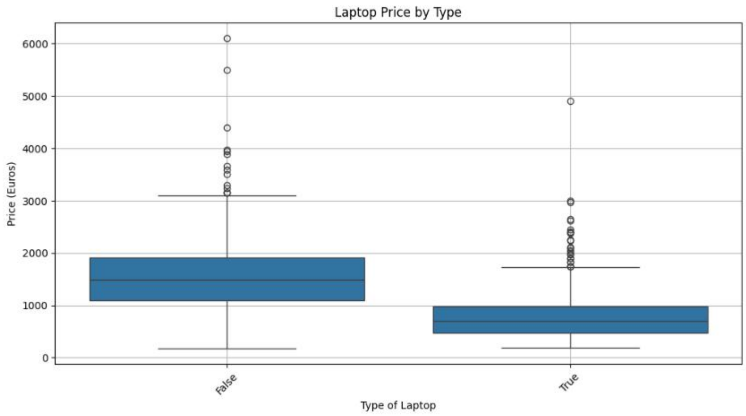
```
# Data Cleaning
laptops['Ram'] = laptops['Ram'].str.replace('GB', '').astype(int)
laptops['Weight'] = laptops['Weight'].str.replace('kg', '').astype(float)
laptops['Touchscreen'] = laptops['Touchscreen'].apply(lambda x: 1 if x == 'Yes' else 0)
# ...continued for other binary and categorical features

# One-Hot Encoding
laptops_encoded = pd.get_dummies(laptops, drop_first=True)

# Splitting Data
X = laptops_encoded.drop('Price_euros', axis=1)
y = laptops_encoded['Price_euros']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Linear Regression
model = LinearRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)

# Evaluation
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
```

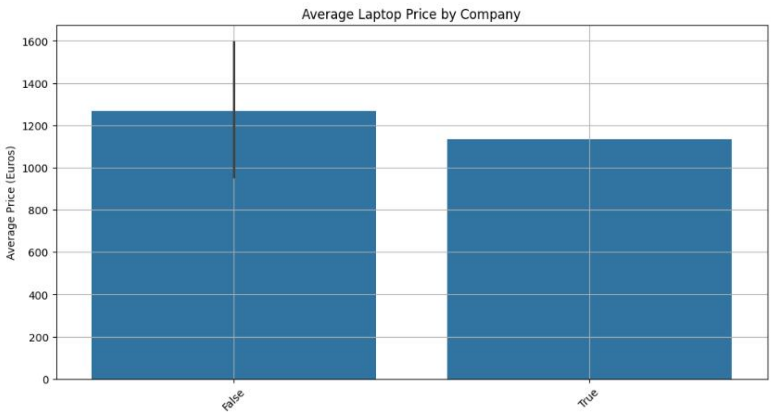
Result

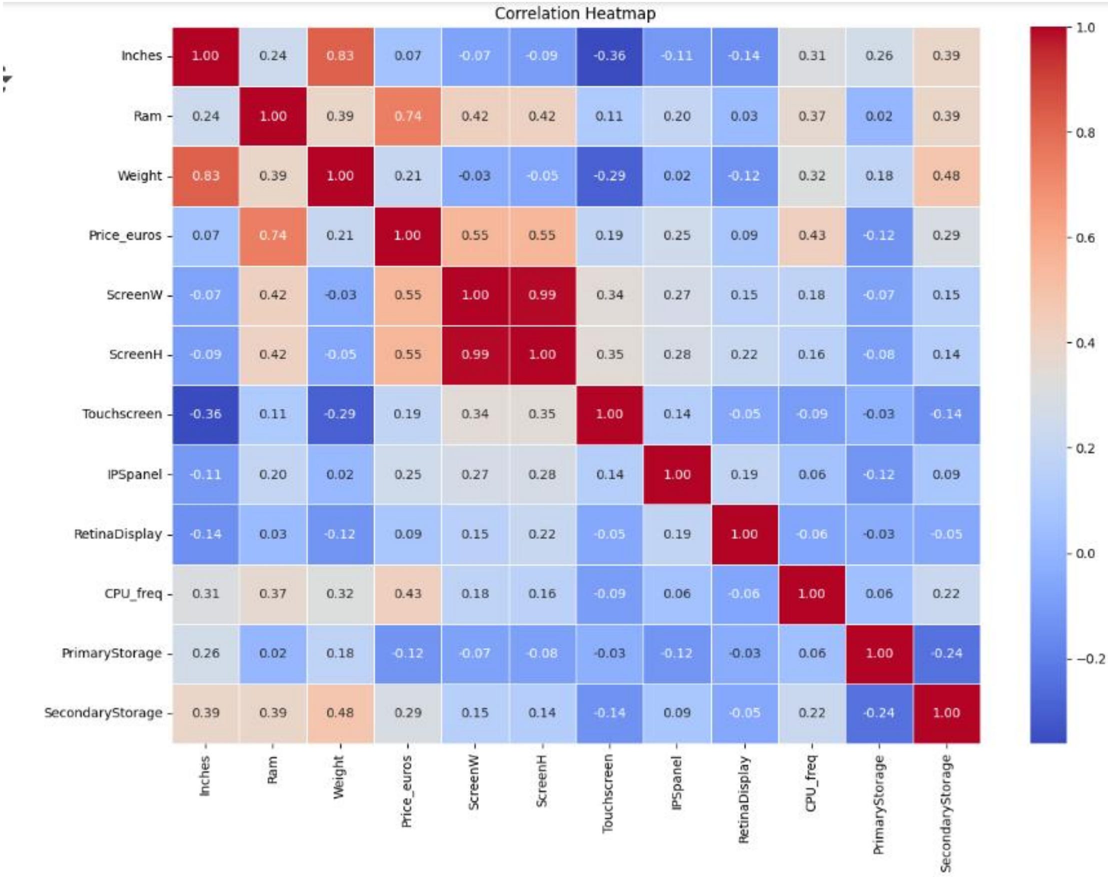


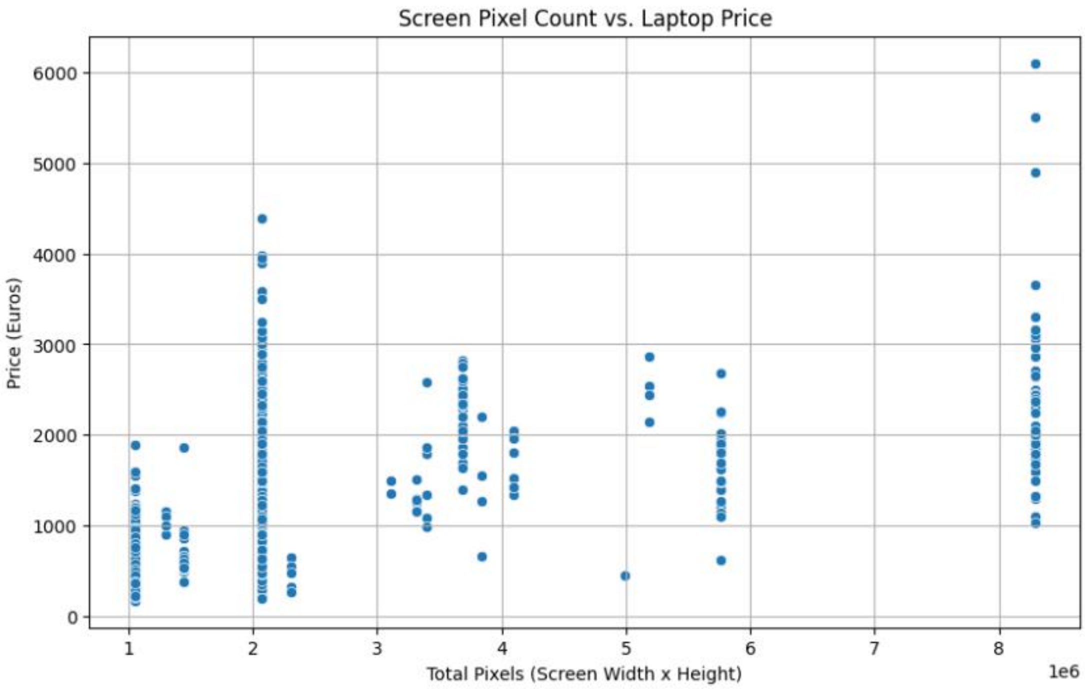



Laptop Price vs. RAM
Price (Euros)
6000
5000
4000
3000
2000
1000
0
2
4
6
8
12
16
24
32
64
RAM (GB)


Laptop Price by Type
Price (Euros)
6000
5000
4000
3000
2000
1000
0
False
True
Type of Laptop


Average Laptop Price by Company
Average Price (Euros)
1600
1400
1200
1000
800
600
400
200
0
False
True





Insights and Conclusions

- **Higher RAM and screen resolution** correlate strongly with increased laptop prices.

- **Company brand** plays a key role: Apple, MSI, and Dell laptops tend to be more expensive.
- **Touchscreen and RetinaDisplay** are also associated with higher prices, but the correlation is weaker.
- The model's **R^2 score indicates a reasonable fit**, but it may not capture non-linear trends effectively.

Future Scope

- Implement **Random Forest** or **XGBoost** to capture non-linear patterns.
- Use **grid search** or **cross-validation** for better hyperparameter tuning.
- Deploy the model via a **web interface** using Flask or Streamlit.
- Enhance model features using natural language processing for product descriptions.

Conclusion

This project successfully demonstrates how **machine learning** can be used to predict laptop prices using structured data. While **Linear Regression** provides a solid baseline, further experimentation with advanced models and richer features could enhance accuracy and practical utility for end users.