LAPTOP PRICE PREDICTION

A DATA SCIENCE & DEPLOYMENT PROJECT

By Himanshu Meshram

AGENDA

Project Introduction

Dataset & Feature Overview

Data Preprocessing & EDA

Modeling & Evaluation

Web App Deployment

Key Learnings & Future Scope

PROJECT INTRODUCTION

Why This Project?

Choosing the right laptop can be overwhelming. With so many brands and configurations, prices vary widely—even for similar specs. Consumers often struggle to understand what features truly affect cost.

The Goal

We set out to build a machine learning model that can predict the price of a laptop based on its specifications. The idea is simple: input the specs, get an estimated price

What Makes This Useful?

- ❖ Empowers consumers with transparent price predictions.
- ❖ Helps sellers benchmark product pricing.
- ❖ Turns complex spec sheets into actionable insights.

End Result

A complete ML pipeline—from data preprocessing to model training and evaluation—along with a deployed Streamlit web app that allows real-time laptop price prediction.

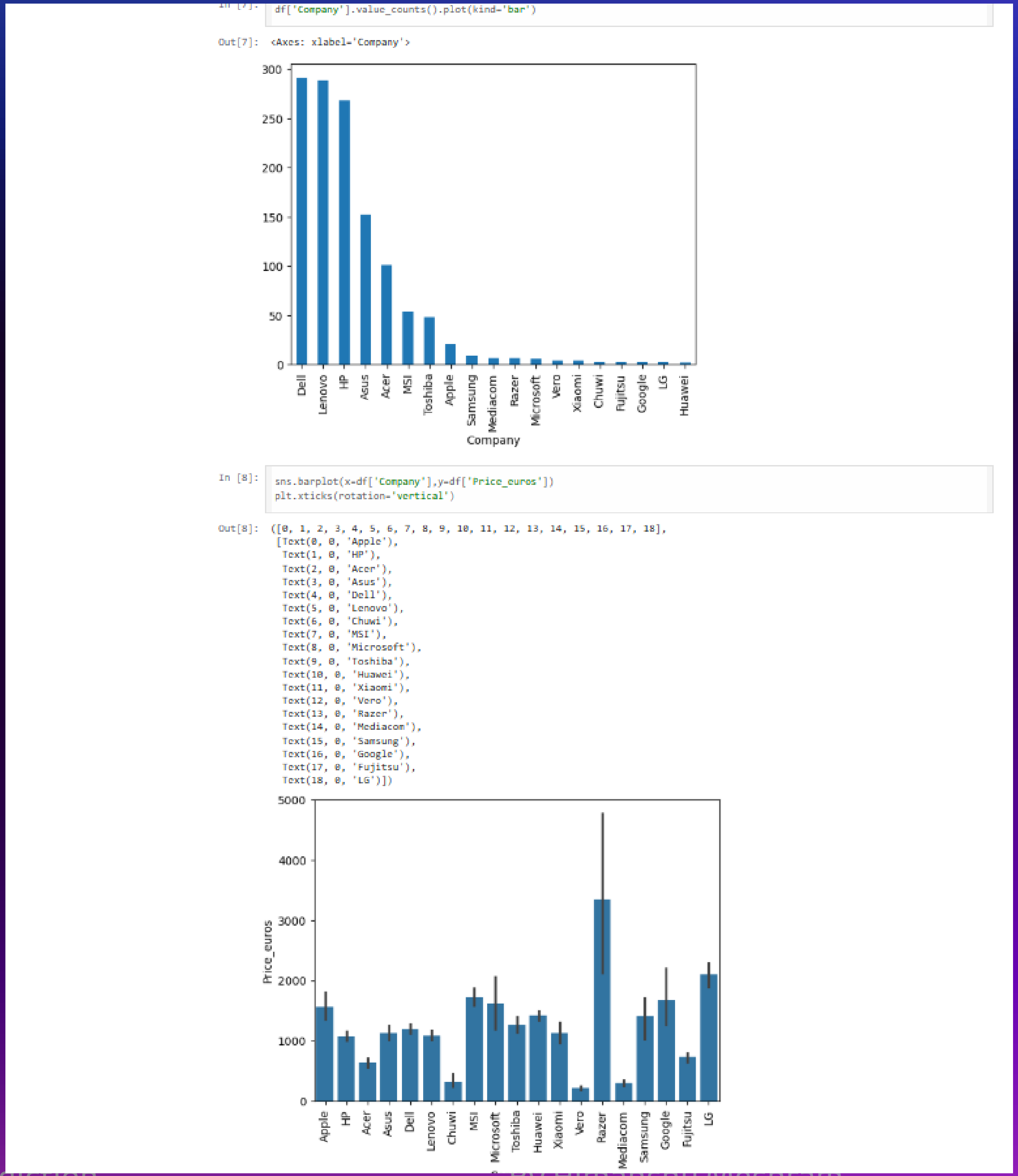
DATASET & FEATURE OVERVIEW

SOURCE & SIZE
PUBLIC DATASET (KAGGLE)
1275 ROWS × 23 COLUMNS

TARGET
PRICE EUROS (LAPTOP PRICE IN EUROS)

KEY FEATURES:
COMPANY, TYPE, SCREEN SIZE, RAM, WEIGHT, CPU, GPU, OS, TOUCHSCREEN, RESOLUTION, PRIMARY & SECONDARY STORAGE

FINAL OUTPUT:
CLEANED & ENCODED DATASET SAVED AS DF.PKL FOR MODELING



DATA PREPROCESSING & EDA

Preprocessing Steps

- Removed duplicates (28 rows)
- Handled missing values (none found)
- Categorical features encoded using One-Hot Encoding
- Created new features:
  - **PPI (Pixels Per Inch)** from resolution & screen size
  - Simplified **CPU** categories

Exploratory Data Analysis

- Bar plots: Brand & laptop type distribution
- Scatter plots: Weight vs Price, Inches vs Price
- Heatmap: Feature correlation
- Insights:
  - Gaming & workstation laptops cost more
  - Price increases with better screen, CPU, GPU

MODELING & EVALUATION

Models Trained

- Linear Regression (baseline)
- Decision Tree Regressor
- Support Vector Regressor (SVR)
- Random Forest Regressor
- Voting & Stacking Regressors (ensembles)

Evaluation

- Metrics

R^2 Score – Measures model accuracy (closer to 1 is better)

MAE (Mean Absolute Error) – Avg. prediction error in Euros

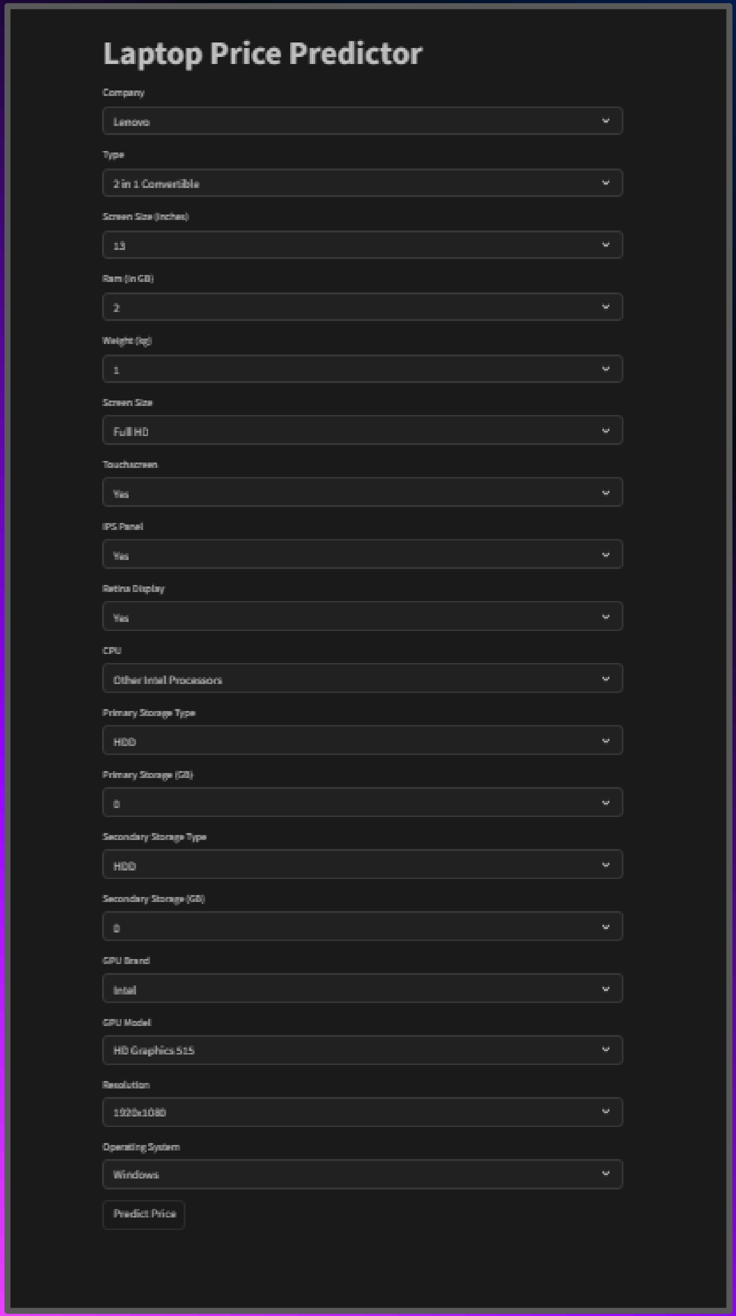
- Best Model

Random Forest Regressor$R^2 \approx 0.88$

MAE ≈ €138

- Outcome

Final model exported as pipe.pkl for deployment



WEB APP DEPLOYMENT

Built With :

- Streamlit – Fast, interactive web interface
- pickle – For loading trained model (pipe.pkl) and data (df.pkl)
- Runs with a single command: streamlit run app.py

User Inputs :

- Brand, Type, Screen Size, RAM, Storage
- Touchscreen, IPS, Retina, CPU, GPU, OS

Behind the Scenes :

- App calculates PPI based on screen resolution & size
- Inputs passed to the model for real-time price prediction

Output :

- Displays predicted price instantly
- Simple, responsive, and easy to use

FUTURE SCOPE

- Try advanced models like XGBoost or LightGBM.
- Use hyperparameter tuning for even better accuracy.
- Add new features: battery life, refresh rate, build quality.
- Continuously update data from e-commerce platforms.
- Host on cloud (AWS, Render, Streamlit Cloud).
- Add user login & prediction history.
- Turn app into a full-stack product with API support.

FINAL TIPS & TAKEAWAYS

Project Summary

- Predicted laptop prices using specs with ~88% accuracy
- Trained and compared multiple regression models
- Built a working Streamlit web app for real-time predictions

Lessons Learned

- Data cleaning & feature engineering greatly impact accuracy.
- Ensemble models (like Random Forest) outperform basic ones
- Streamlit makes deploying ML models fast and intuitive

Real-World Value

- Helps consumers make informed purchase decisions
- Can assist e-commerce platforms in dynamic pricing

THANK YOU

Himanshu Meshram

[LinkedIn]

[Mail]

[Github]