Electronic Device Rating Prediction

Department: CS

Team members:

Team Member Name	Team Member ID
اسراء علي رياض عبدالحافظ	2021170068
بسمله نعيم عبد الحكيم عبد الوهاب	2021170116
الاء علاء عاشور محمد	2021170078
احمد هيثم احمد السرسي	2021170052
احمد اشرف رفاعي عبد السلام	2021170012

Preprocessing Phase:

- **Brand:** it was of object type then converted into Category type as it contained set of unique names. **One hot encoding** was then applied.
- **processor_brand**: it was of object type then converted into Category type as it contained set of unique names. **One hot encoding** was then applied as it had no natural order.
- **processor_gnrtn**: it was of object type, preprocessing applied was removing "th" characters from every value, replacing not available with zero, then converting it into category type. **One hot encoding** was then applied as it had no natural order.
- **ram_gb**: it was of object type, preprocessing applied was removing "GB" characters, then converting it into int type. Outliers were detected using boxplot then removed using **IQR** technique.
- **ram_type**: it was of object type, then converted into category type. **One hot encoding** was then applied as it had no natural order.
- **ssd**: it was of object type, preprocessing applied was removing "GB" characters, then converting it into int type. Outliers were detected using boxplot then removed using **IQR** technique.
- **hdd**: it was of object type, preprocessing applied was removing "GB" characters, then converting it into int type. Outliers were detected using boxplot then removed using **IQR** technique.

- **os**: it was of object type, then converted into category type. **One hot encoding** was then applied as it had no natural order.
- **graphic_card_gb**: it was of object type, preprocessing applied was removing “GB” characters then converting it into int type. Outliers were detected using boxplot then removed using **IQR** technique.
- **weight**: it was of object type, then converted it into category type, then encoded using **label encoder** as it had natural order to make it easier for models.
- **warranty**: it was of object type, preprocessing applied was removing “year” or “years”, replacing “No warranty” with “0”, then converting it into category type. **One hot encoding** was then applied as it had no natural order.
- **Touchscreen**: it was of object type, then converted into category type. **One hot encoding** was then applied as it had no natural order.
- **msoffice**: it was of object type, then converted into category type. **One hot encoding** was then applied as it had no natural order.
- **Price**: it was of int type so preprocessing applied was detecting outliers using boxplot then removing them using **IQR** technique.
- **rating**: the target feature, was of object type. Preprocessing applied was removing “stars” substring then converting it into category type as it already contained unique set of values. It then was encoded using **label encoder** as it had natural order.

- **Number of Ratings**: it was of int type, so preprocessing applied was detecting outliers using boxplot then removing them using **IQR** technique.
- **Number of Reviews**: it was of int type, so preprocessing applied was detecting outliers using boxplot then removing them using **IQR** technique.

After applying these preprocessing techniques, duplicates were checked and apparently 30 duplicated were detected, then they were removed. Moreover, we checked for missing data. In addition, we checked if there was any device with no processor generation, no HDD and no SDD and if there were any device found, it meant it had no processor but apparently there was not any. After that, we normalized the data using **MinMax** scaler.

Analysis Phase:

Features were selected based on their correlation with the target column **“rating”**. Selected features had correlation with absolute value greater than **‘0.15’**, any other feature was discarded.

Selected Features were : **“ram_gb”**, **“Number of Ratings”**, **“Number of Reviews”**, **“msoffice_No”**, **“msoffice_Yes”**.

“msoffice_No” was dropped as it had negative perfect correlation with **“msoffice_Yes”**.

So final features were: **“ram_gb”**, **“Number of Ratings”**, **“Number of Reviews”**, **“msoffice_Yes”.**

Model Phase:

The dataset was divided into 75% train and 25% test.

- **Logistic Regression:** it works well with a categorized target. Accuracy of **'95.8%'** was acquired with 300 max iterations and cross validation technique applied.
- **Linear Regression:** Overfitting occurred and accuracy of **'100%'** was acquired even with cross validation.
- **Lasso Regression:** It was a perfect model to control overfitting problems with its normalization hyperparameter and accuracy of **'90.3%'** was acquired with alpha hyperparameter = '0.0062' and with cross validation applied.
- **Ridge Regression:** Another perfect model to control overfitting problems with its normalization hyperparameter. Accuracy of **'90.3%'** was acquired with alpha hyperparameter = '4' and with cross validation applied.
- **Polynomial Regression:** Overfitting occurred as Linear regression with accuracy of **'100%'** acquired and cross validation applied.

Conclusion:

For this phase, we had an intuition that many features like **“processor_gnrtn”**, **“price”**, **“ssd”**, **“processor_brand”** and **“graphic_card_gb”** will play key roles in **“rating”** prediction, and features like **“Number of Ratings”** or **“Number of Reviews”** will be discarded.

But on the contrary, **“Number of Reviews”** and **“Number of Ratings”** were the features that had the highest correlation with **“rating”** in addition to **“ram_gb”** and **“msoffice_Yes”** while the features that were mentioned beforehand were discarded.

The preprocessing almost involved converting the features to their right types to make it easier for prediction. Few duplicates were found, there were no null values. Many outliers were detected in almost all numeric values like **“ram_gb”**, **“hdd”**, **“sdd”**, **“price”**, **“graphic_card_gb”**, **“Number of Ratings”** and **“Number of Reviews”** but were removed using **IQR** technique. Then Normalization using **MinMax** scaler was applied.

We have tried many models, and many were overfiiting like **linear** and **polynomial** regression as they had accuracy of **‘100%’**, While **Lasso** Regression and **ridge** Regression were the perfect models as they had high accuracy of **’90.3%’** and no overfitting. **Logistic** Regression had high accuracy too with accuracy of ’**95.8%**’.