App Rating Prediction

roject 1 Gradable (

DESCRIPTION

Objective: Make a model to predict the app rating, with other information about the app provided.

Problem Statement:

Google Play Store team is about to launch a new feature wherein, certain apps that are promising, are boosted in visibility. The boost will manifest in multiple ways including higher priority in recommendations sections ("Similar apps", "You might also like", "New and updated games"). These will also get a boost in search results visibility. This feature will help bring more attention to newer apps that have the potential.

Domain: General

Analysis to be done: The problem is to identify the apps that are going to be good for Google to promote. App ratings, which are provided by the customers, is always a great indicator of the goodness of the app. The problem reduces to: predict which apps will have high ratings.

Content: Dataset: Google Play Store data ("googleplaystore.csv")

Fields in the data App: Application name Category: Category to which the app belongs

- Rating: Overall user rating of the app
- Reviews: Number of user reviews for the app
- Size: Size of the app
 - Installs: Number of user downloads/installs for the app
 - Type: Paid or Free
 - Price: Price of the app
 - Content Rating: Age group the app is targeted at Children / Mature 21+ / Adult
- Genres: An app can belong to multiple genres (apart from its main category). For example, a musical family game will belong to Music, Game, Family genres.
- Last Updated: Date when the app was last updated on Play Store
- Current Var Current version of the ann available on Dlay Story
- Current Ver: Current version of the app available on Play Store
- Android Ver: Minimum required Android version

1. Load the data file using pandas.

Steps to perform:

5. Sanity checks:

- 2. Check for null values in the data. Get the number of null values for each column.
- 3. Drop records with nulls in any of the columns.
- 4. Variables seem to have incorrect type and inconsistent formatting. You need to fix them:
- 1. Size column has sizes in Kb as well as Mb. To analyze, you'll need to convert these to numeric.

 - 2. Multiply the value by 1,000, if size is mentioned in Mb
- 2. Reviews is a numeric field that is loaded as a string field. Convert it to numeric (int/float).

1. Extract the numeric value from the column

- 3. Installs field is currently stored as string and has values like 1,000,000+.
 - 1. Treat 1,000,000+ as 1,000,000
- 2. remove '+', '.' from the field, convert it to integer
- 4. Price field is a string and has \$ symbol. Remove '\$' sign, and convert it to numeric.
- 1. Average rating should be between 1 and 5 as only these values are allowed on the play store. Drop the rows that have a value outside this range.
- 2. Reviews should not be more than installs as only those who installed can review the app. If there are any such records, drop them.
 - 3. For free apps (type = "Free"), the price should not be >0. Drop any such rows.

- 5. Performing univariate analysis:
- Boxplot for Price
- Are there any outliers? Think about the price of usual apps on Play Store.
- Boxplot for Reviews
- Are there any apps with very high number of reviews? Do the values seem right?
- Histogram for Rating
 - How are the ratings distributed? Is it more toward higher ratings?
- Histogram for Size

Note down your observations for the plots made above. Which of these seem to have outliers?

- 6. Outlier treatment:
- 1. Price: From the box plot, it seems like there are some apps with very high price. A price of \$200 for an application on the Play Store is very high and suspicious!
- 1. Check out the records with very high price

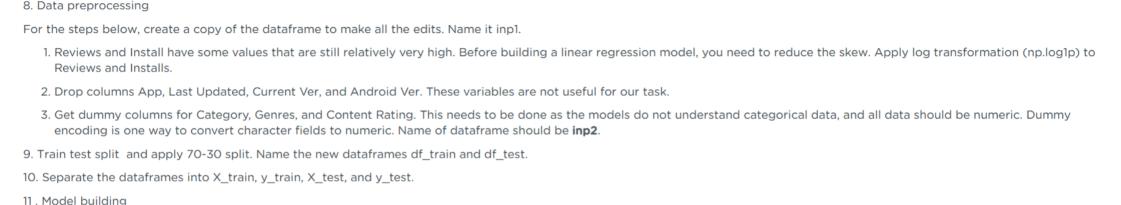
1. Is 200 indeed a high price?

- 2. Drop these as most seem to be junk apps
- 2. Reviews: Very few apps have very high number of reviews. These are all star apps that don't help with the analysis and, in fact, will skew it. Drop records having more than 2 million reviews.
- 3. Installs: There seems to be some outliers in this field too. Apps having very high number of installs should be dropped from the analysis.
 - 1. Find out the different percentiles 10, 25, 50, 70, 90, 95, 99
 - 2. Decide a threshold as cutoff for outlier and drop records having values more than that
 - y values more than that

- 7. Bivariate analysis: Let's look at how the available predictors relate to the variable of interest, i.e., our target variable rating. Make scatter plots (for numeric features) and box plots (for character features) to assess the relations between rating and the other features.
- 1. What pattern do you observe? Does rating increase with price?
- Make scatter plot/joinplot for Rating vs. Size
 Are heavier apps rated better?

1. Make scatter plot/joinplot for Rating vs. Price

- 3. Make scatter plot/joinplot for Rating vs. Reviews
- 1. Does more review mean a better rating always?
- 4. Make boxplot for Rating vs. Content Rating
- 1. Is there any difference in the ratings? Are some types liked better?
- 5. Make boxplot for Ratings vs. Category
 - 1. Which genre has the best ratings?



Use linear regression as the technique

12. Make predictions on test set and report R2.

Report the R2 on the train set