Game Application Rating Prediction

Artificial Intelligence

CS-617-C

GILLELLA AVINASH REDDY

Sacred Heart University
School of Computer Science & Engineering
The Jack Welch College of Business & Technology

Submitted To:
Dr. Reza Sadeghi

Spring 2022

Final Project Report of Game Application rating prediction

Name : Gillella Avinash Reddy

SHU Email : gillellaa2@mail.sacredheart.edu

Short Bio. :

I Avinash Reddy Gillella from India. As I pursued my undergrad in computer science I have good knowledge on C, java & python. Both my mini & major project during my undergrad are related to Machine Learning ie.,Movie recommendation system & A Multimodal Attentive for multi-level Sentiment Analysis.

For this project, I am going to predict the rating of the game application by using the KNN (k- nearest neighbours), Support vector machine(SVM) & Logistic Regression algorithms which are well known classification algorithm.

Table of Contents

1. INTRODUCTION

1.1 General Description :

This projects aims predict the rating of game applications in an application store that can be achieved in course of time. Here we consider main attributes like customer rating, customer count, in app purchases so on which supports our prediction. This helps in us to find out the best game genre to be released in a certain area to get a best customer rating.

1.2 Research Question :

Applications to be released in order to gain market attention ?

Predicting rank of game application in today's life ?

1.3 GitHub Repository Address

GitHub Repository Address for Game application rating prediction is the following :

https://github.com/avinashgillella/avinashgillella-Game-Application-rating-prediction.git

2. DATA DESCRIPTION

This project is related to Application rating prediction ; It contains a dataset with 17007 data samples and 16 features:

- Icon URL : It represents the url of the icon which displays while we search in the app store. It is a string datatype.
- Average user rating : It represents the average of all the ratings given by the user.It is a float datatype.
- User rating count : It represents the the count of the users who gave the feedbacks. It is a float datatype.
- Price : It represents the cost of the application for the user to download the application. It is a float datatype.
- In app purchases : It represents the cost of other purchases in the application. It is a float datatype.
- Description : It shows the information about the application. It is a string datatype.
- Developer : It represents the name of the developer who developer the application. It is a string datatype.
- Age rating : It represents the rating of the age. It is a string datatype.
- Languages : It represents all the languages in which the application is displayed. It is a string datatype.
- Size : It represents the memory consumed by the application. It is float datatype.
- Primary genres : It represents the category of game to which it belongs to. It is a string datatype.
- Genres : It represents the sub category of game to which it belong to. It is a string datatype.

- Original Release Date : It represents the release date of the applications. It is a date time datatype.
- Current version Release Date : It represents the release date of the latest version of the application. It is a date time datatype.

Source of Dataset :

[https://verzeo.com/]

- This data is provided by the company called verzeo during my machine learning program internship.

PROJECT PLAN

Project plan consists of two steps

1. Data pre-processing
2. Model construction
3. Optimisation
4. Model Evaluation

Firstly, we are going to apply Pre-Processing techniques that are Data Cleaning, Data Normalisation, Data Reduction, Data Transformation[1].

- Data Cleaning : Data cleaning in data mining is the process of detecting and removing corrupt or inaccurate records from a record set, table or database. Either you can ignore the tuple or fill the missing values manually.

- Data Normalisation : Database normalisation is the process of structuring a database, usually a relational database, in accordance with a series of so-called normal forms in order to reduce data redundancy and improve data integrity.

- Data Reduction : Data reduction is the transformation of numerical or alphabetical digital information derived empirically or experimentally into a corrected, ordered, and simplified form.

- Data Transformation : Data transformation is the process of converting data from one format or structure into another format or structure. It is a fundamental aspect of most data integration and data management tasks such as data wrangling, data warehousing, data integration and application integration.

In this step we build a model by K nearest neighbours, SVM, Logistic Regression it is supervised machine learning algorithm that can be used to solve both classification and regression problems.

Model optimisation : Optimization is the process where we train the model iteratively that results in a maximum and minimum function evaluation. It is one of the most important phenomena in Machine Learning to get better results. [2]

Model Evaluation : Model evaluation is the process of using different evaluation metrics to understand a machine learning model's performance, as well as its strengths and weaknesses. Model evaluation is important to assess the efficacy of a model during initial research phases, and it also plays a role in model monitoring. [3]

RELATED WORK

- As mobile apps have become so prevalent, and more and more developers make their livelihood off of mobile development alone. It has become important for developers to be able to predict the success of their app. Our goal was to find the overall rating of an app because so much of the users' trust in the app comes from that one statistic alone. Higher rated apps are more likely to be recommended and more likely to be trusted by users that find the app while browsing the app store

https://medium.com/@ertebablu_7511/google-app-store-rating-prediction-ffa7343cf1be

- This research will predict the rating of the application on Google Play using the Random Forest method so that it is hoped that it can help find the weaknesses of the application in a short time from the user's point of view as an ingredient to improve the product.

https://www.researchgate.net/publication/354952581_Predict_App_Rank_on_Google_Play_Using_the_Random_Forest_Method

- 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.

https://www.chegg.com/homework-help/questions-and-answers/app-rating-prediction-project-1-description-objective-make-model-predict-app-rating-inform-q92850531

DATA EXPLORATION

Univariant Analysis :

Univariate analysis is the most basic form of statistical data analysis technique. When the data contains only one variable and doesn't deal with a causes or effect relationships then a Univariate analysis technique is used.[4]

Histogram :

A histogram is a graphical representation that organizes a group of data points into user-specified ranges [5]

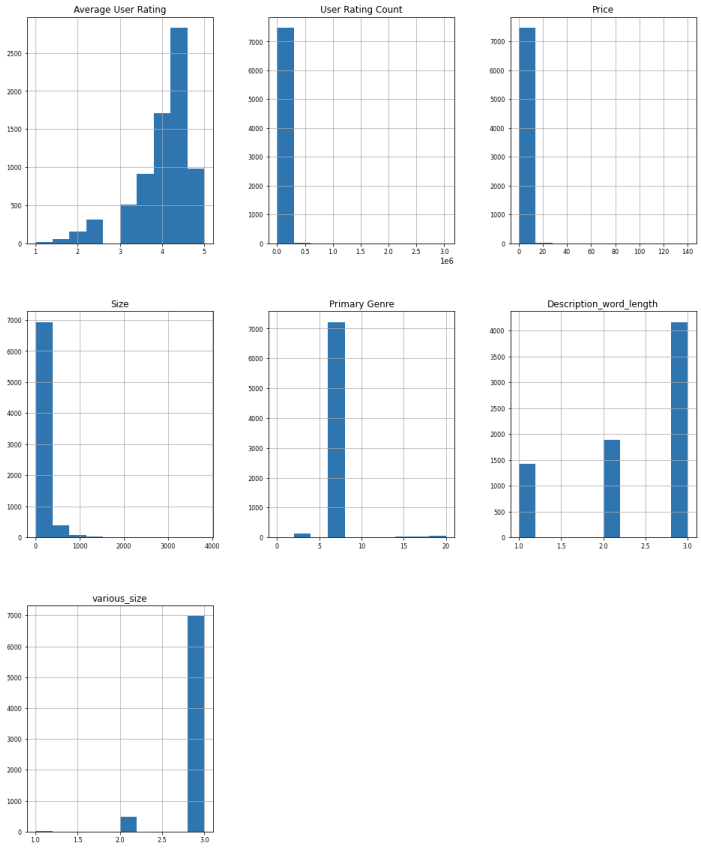

fig 5.1.1 : Histogram of each feature

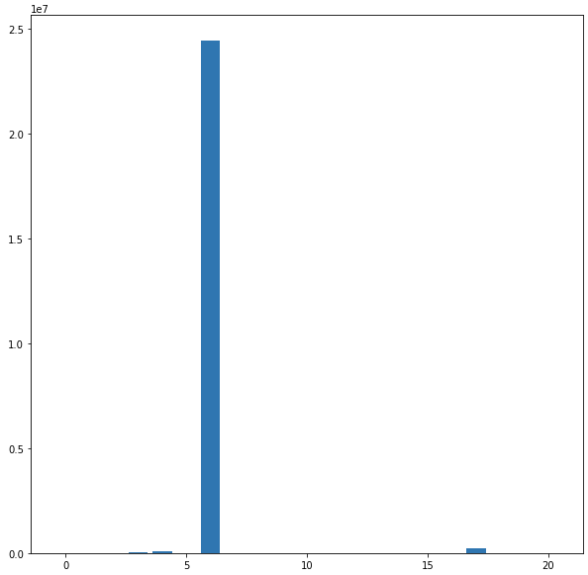

fig 5.1.1 Histogram of user rating count

Descriptive analysis :

Descriptive Analysis is the type of analysis of data that helps describe, show or summarize data points in a constructive way such that patterns might emerge that fulfill every condition of the data. It is one of the most important steps for conducting statistical data analysis. [6]

```
In [30]: df.describe()
```

Out[30]:

	Average User Rating	User Rating Count	Price	Size	Primary Genre	Description_word_length	various_size
count	7488.000000	7.488000e+03	7488.000000	7488.000000	7488.000000	7488.000000	7488.000000
mean	4.062099	3.306245e+03	0.569686	144.545651	6.091079	2.366854	2.929754
std	0.750506	4.251578e+04	2.422359	244.092470	1.455905	0.782749	0.265321
min	1.000000	5.000000e+00	0.000000	0.205841	0.000000	1.000000	1.000000
25%	3.500000	1.200000e+01	0.000000	29.066406	6.000000	2.000000	3.000000
50%	4.500000	4.600000e+01	0.000000	75.625000	6.000000	3.000000	3.000000
75%	4.500000	3.072500e+02	0.000000	169.050049	6.000000	3.000000	3.000000
max	5.000000	3.032734e+06	139.990000	3820.029297	20.000000	3.000000	3.000000

fig 5.1.2 Analysis of data

Distplot :

distplot() function is used to plot the distplot. The distplot represents the univariate distribution of data i.e. data distribution of a variable against the density distribution. The seaborn. distplot() function accepts the data variable as an argument and returns the plot with the density distribution. [8]

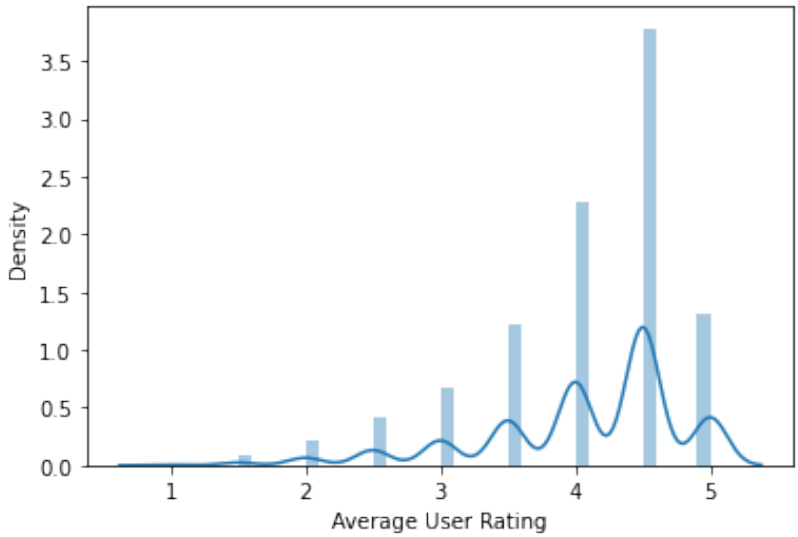

fig 5.1.3 distance graph between Average user rating and Density

The average user rating of the games falls in between 4 and 5, which indicates a good engaging application have been proposed and released also the price range of these games whose rating is high have low application price.

Bivariant Analysis :

Bivariate analysis is slightly more analytical than Univariate analysis. When the data set contains two variables and researchers aim to undertake comparisons between the two data set then Bivariate analysis is the right type of analysis technique.[7]

Countplot :

A countplot is kind of like a histogram or a bar graph for some categorical area. It simply shows the number of occurrences of an item based on a certain type of category.[7]

fig 5.2.1 Count of features in the dataset

By this graph we can see the count of each feature in the data.

Pair plot :

A pairplot plot a pairwise
elationships in a dataset. The pairplot function creates a grid of Axes such that each variable in data will by shared in the y-axis across a single row and in the x-axis across a single column.

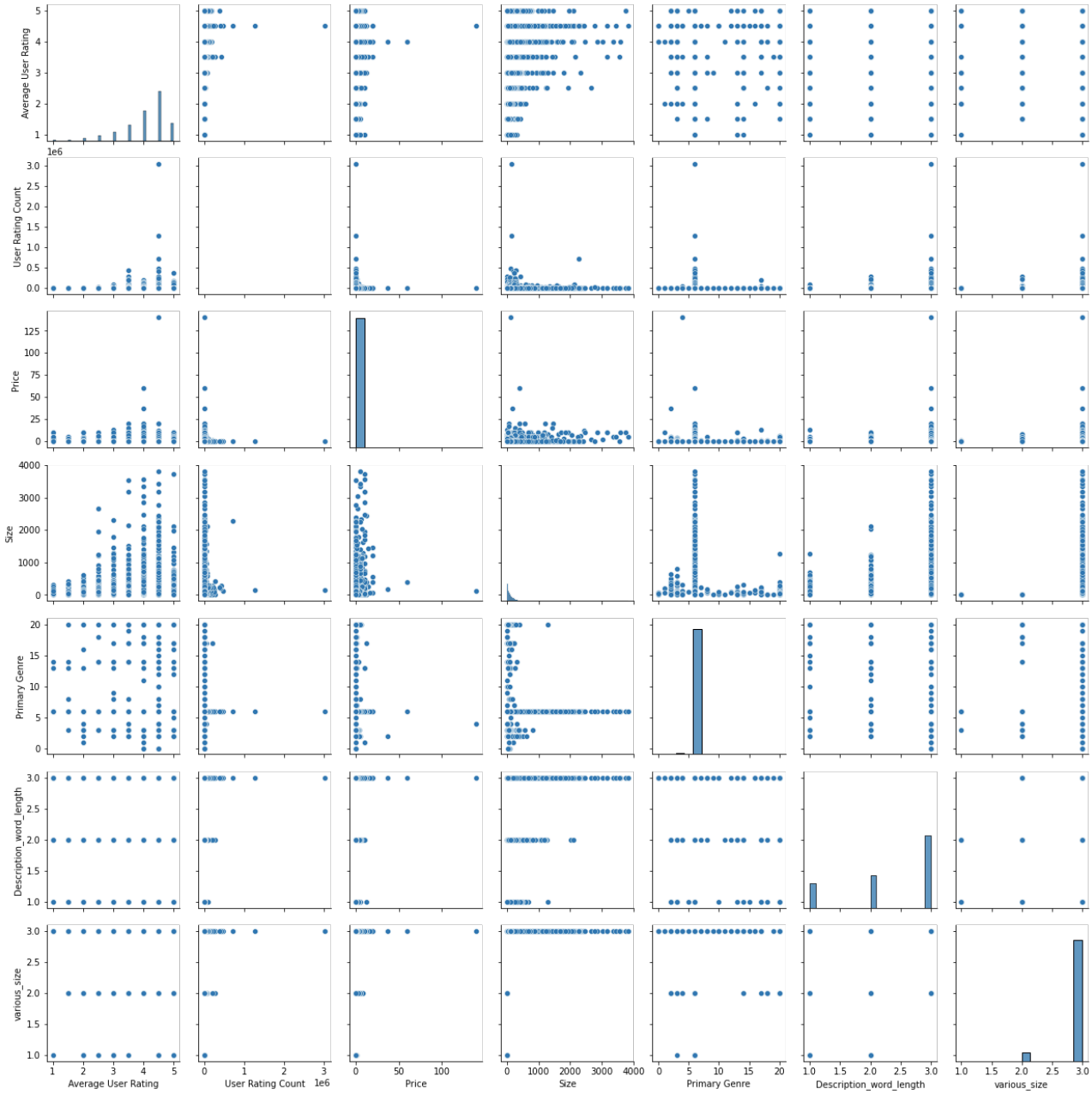

fig 5.2.2 : pair plot of dataset

DATA MODELLING

Data Splitting :

Data splitting is commonly used in machine learning to split data into a train, test, or validation set. This approach allows us to find the model hyper-parameter and also estimate the generalization performance.[9]

```
In [49]: X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.33, random_state=42)

In [50]: len(X_train), len(X_test), len(y_train), len(y_test)
Out[50]: (5016, 2472, 5016, 2472)
```

Here, we have divided the entire dataset into two parts train data and test data.

- Train data is use to train an algorithm or machine learning model to predict the outcome you design your model to predict.
- Test data is used to measure the performance, such as accuracy or efficiency, of the algorithm you are using to train the machine. (ie., 33% data from entire dataset)

Fitting the model :

Fitting a model means that you're making your algorithm learn the relationship between predictors and outcome so that you can predict the future values of the outcome.[10]

In this project, I have build three models

- K Nearest Neighbor Classifier :

  The k-nearest neighbors (KNN) algorithm is a simple, supervised machine learning algorithm that can be used to solve both classification and regression problems. It's easy to implement and understand, but has a major drawback of becoming significantly slows as the size of that data in use grows.[11]

```
neigh = KNeighborsClassifier(n_neighbors=50)
neigh.fit(X_train, y_train)
```

- Support Vector Machine :

  SVM works by mapping data to a high-dimensional feature space so that data points can be categorized, even when the data are not otherwise linearly separable. A separator between the categories is found, then the data are transformed in such a way that the separator could be drawn as a hyperplane.[12]

```
from sklearn import svm
sv_clf = svm.SVC()
sv_clf.fit(X_train, y_train)
```

- Logistic Regression Classifier :

  Logistic regression is basically a supervised classification algorithm. In a classification problem, the target variable(or output), y, can take only discrete values for a given set of features(or inputs), X. Contrary to popular belief, logistic regression IS a regression model.[13]

```
from sklearn.linear_model import LogisticRegression
log_clf = LogisticRegression(max_iter=1000)
log_clf.fit(X_train, y_train)
```

Measuring Performance :

- K Nearest Neighbor Classifier :

1. Accuracy : It is one of the metric for evaluating the model.

   accuracy_score(y_test, y_predicted)

   0.73220064724919

   Here we got 73.2 % accuracy for the model.

2. Confusion matrix : A Confusion matrix is an N x N matrix used for evaluating the performance of a classification model, where N is the number of target classes. The matrix compares the actual target values with those predicted by the machine learning model. [14]

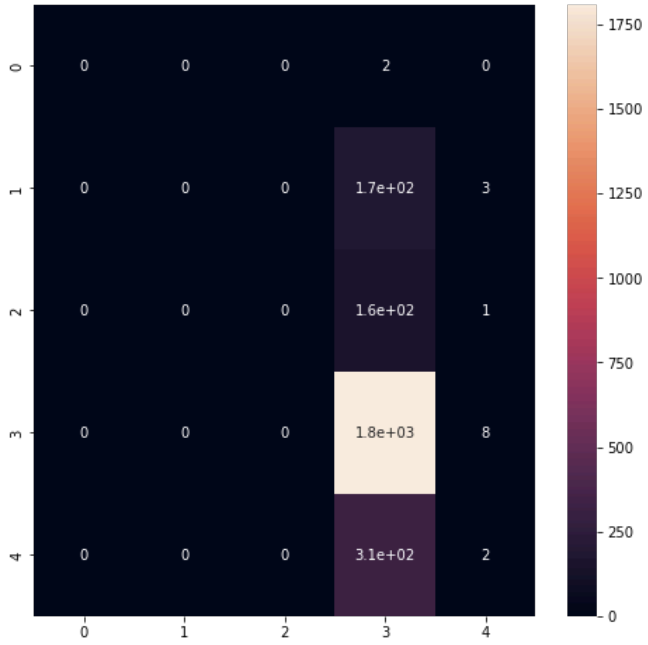

fig 6.4.1 : Confusion matrix using KNN algm

- Support Vector Machine (SVM) :

1. Accuracy : It is one of the metric for evaluating the model.

   accuracy_score(y_test, y_predicted)

   0.7346278317152104

   Here we got 73.5 % accuracy for the model.

2. Confusion matrix : A Confusion matrix is an N x N matrix used for evaluating the performance of a classification model, where N is the number of target classes. The matrix compares the actual target values with those predicted by the machine learning model.[14]

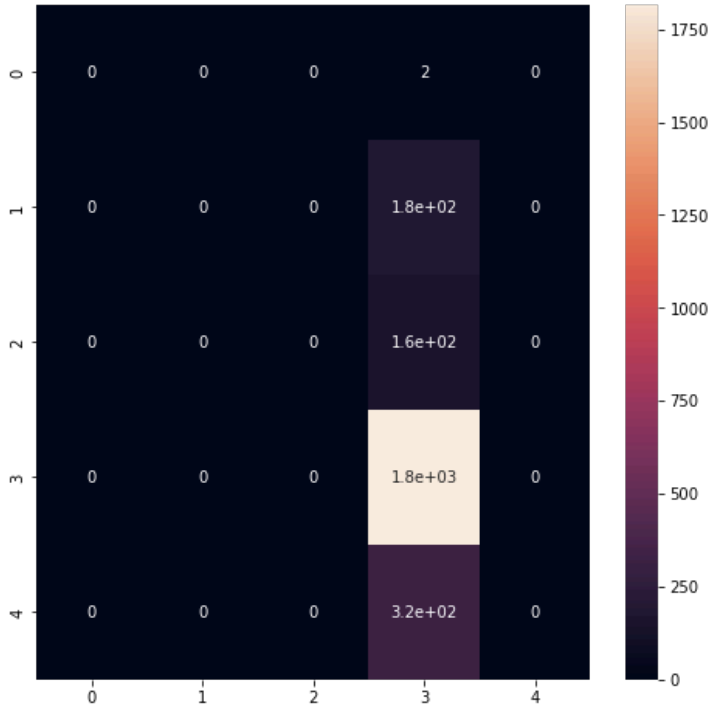

fig 6.4.1 : Confusion matrix using SVM algm

- Logistic Regression Classifier :

1. Accuracy : It is one of the metric for evaluating the model.

   y_predict = log_clf.predict(X_test)

   accuracy_score(y_test, y_predicted)

   0.7338187702265372

   Here we got 73.4 % accuracy for the model.

2. Confusion matrix : A Confusion matrix is an N x N matrix used for evaluating the performance of a classification model, where N is the number of target classes. The matrix compares the actual target values with those predicted by the machine learning model.[14]

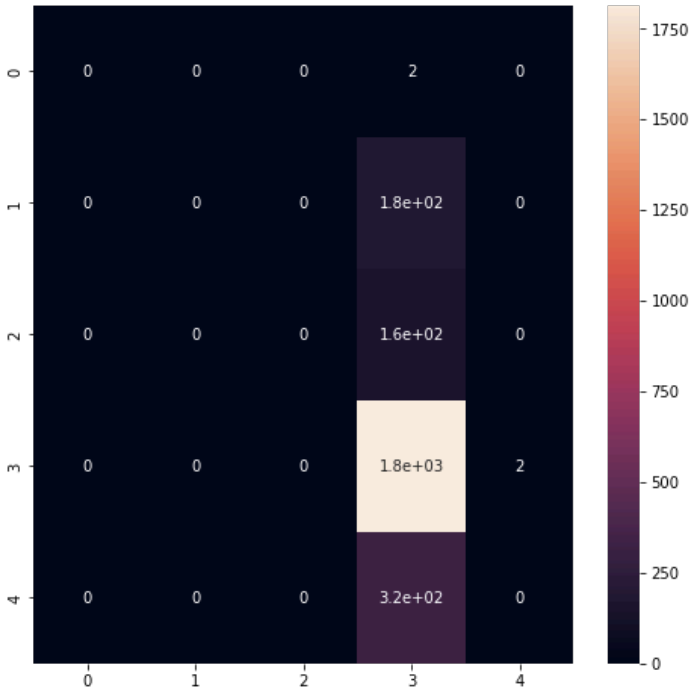

fig 6.4.1 : Confusion matrix using Logistic Regression algm

Optimisation :

Optimization is the process where we train the model iteratively that results in a maximum and minimum function evaluation. It is one of the most important phenomena in Machine Learning to get better results. [2]

K Fold splitting :

K-Folds cross-validator. **Provides train/test indices to split data in train/test sets**. Split dataset into k consecutive folds (without shuffling by default). Each fold is then used once as a validation while the k - 1 remaining folds form the training set. Parameters n_splitsint, default=5.

```
from sklearn.model_selection import KFold
kf = KFold(n_splits=3, random_state=21, shuffle=True)
kf.get_n_splits(x)
```

Here I have split the data into 3 parts.

Creating new Features :

```
df['price_range'] = [1 if i<= 1 else 2 if 1<i<=2 else 3 for i in df[‘Price']]
```

Here I have created new feature using price feature ie., divided price range to 3 categories if price is less than or equal to 1 the the sample represents 1 else if price is greater than 1 and less than or equal to 2 then display 2 else 3.

Recreating the structure :

```
# recreating the structure
np.random.seed(0)
sample_weight = abs(np.random.randn(len(x)))
sample_weight_constant = np.ones(len(x))
# and bigger weights to some outliers
sample_weight[15:] *= 5
sample_weight[9] *= 15

# for reference, first fit without sample weights
# fit the model
clf_weights = svm.SVC(gamma=1)
clf_weights.fit(x, y, sample_weight=sample_weight)
```

```
clf_no_weights = svm.SVC(gamma=1)
clf_no_weights.fit(x, y)

clf_weights.score(X_test, y_test)
```

Model Evaluation :

Model evaluation is the process of using different evaluation metrics to understand a machine learning model's performance, as well as its strengths and weaknesses. Model evaluation is important to assess the efficacy of a model during initial research phases, and it also plays a role in model monitoring. [3]

Here I have used Random forest classifier to build a new model and evaluate the performance of the model

Random forest. classifier :

It is a ensemble learning method that operates by constructing multitudes of decision trees at training time.

Classification task :

```
y_predict = clf_weights.predict(X_test)
accuracy_score(y_test, y_predict)
con_matrix = confusion_matrix(y_test, y_predict)
plt.figure(figsize=(8,8))
sns.heatmap(con_matrix, annot = True)
```

CONCLUSION

To conclude that most of the users are interested in game applications which are either free or have low price. Rating ranges from 4 to 5 for most of the games with average application size.

Future enhancement :

Data that has been collected was totally related game genre, instead in future we can collect all types of genre applications without bias and bring an inference like which application is most popular, and which type of can should be released in up coming days so that it can sustain the market and with stand in it by considering application size, price, genre,

REFERENCES

[1] https://www.analyticsvidhya.com/blog/2021/08/data-preprocessing-in-data-mining-a-hands-on-guide/

[2] https://towardsdatascience.com/understanding-optimization-algorithms-in-machine-learning-edfdb4df766b#:~:text=Optimization%20is%20the%20process%20where,Learning%20to%20get%20better%20results.

[3] https://www.dominodatalab.com › data-science-dictionary

[4]. https://www.google.com/search?q=univariate+analysis&oq=univaria&aqs=chrome.2.69i57j69i59j0i67i433l3j0i67l4j0i20i263i512.5083j0j7&sourceid=chrome&ie=UTF-8

[5] https://www.google.com/search?q=histogram&oq=histogram&aqs=chrome.0.69i59j0i20i263i433i512l2j0i67i433j0i67j0i433i512j0i67j0i512l3.4124j0j9&sourceid=chrome&ie=UTF-8

[6] https://medium.com/appengine-ai/descriptive-analysis-machine-learning-268507a99e2#:~:text=Descriptive%20Analysis%20in%20Machine%20Learning,types%20of%20Data%20Analysis%20concepts.

[7] https://www.google.com/search?q=Bivariant+Analysis&sxsrf=APq-WBs-st7mMn3R1JrhF6E-2CA0HqYtQA%3A1647667122949&ei=smc1YtnSOcOnptQP8J6OsAk&ved=0ahUKEwjZ88isttH2AhXDk4kEHXCPA5YQ4dUDCA4&uact=5&oq=Bivariant+Analysis&gs_lcp=Cgdnd3Mtd2l6EAMyBwgAELEDEAoyBAgAEAoyBAgAEAoyBAgAEAoyBAgAEAoyBAgAEAoyBAgAEAoyBAgAEAo6BwgAEEcQsAM6BAgAEA1KBAhBGABKBAhGGABQ1gRYkQpg0w5oAXABeACAAViIAaEBkgEBMpgBAKABAcgBCMABAQ&sclient=gws-wiz

[7] https://www.geeksforgeeks.org/countplot-using-seaborn-in-python/

[8]. https://pythonbasics.org/seaborn-pairplot/#:~:text=A%20pairplot%20plot%20a%20pairwise,axis%20across%20a%20single%20column.

[9]. https://www.diva-portal.org/smash/get/diva2:1506870/FULLTEXT01.pdf

[10] https://www.quora.com/What-does-fitting-a-model-mean-in-data-science#:~:text=Fitting%20a%20model%20means%20that,future%20values%20of%20the%20outcome.

[11] https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761

[12] https://www.ibm.com/docs/it/spss-modeler/SaaS?topic=models-how-svm-works

[13] https://www.geeksforgeeks.org/understanding-logistic-regression/

[14] https://www.analyticsvidhya.com/blog/2020/04/confusion-matrix-machine-learning/