CLEMSON UNIVERSITY

PROJECT NAME: NETFLIX PREDICTOR

CPSC 6300: APPLIED DATA SCIENCE

SEMESTER: SPRING 2020

INDIVIDUAL PROJECT: FINAL REPORT

MAY 1st, 2020

COURSE INSTRUCTOR

DR. NINA HUBIG

PROJECT BY:

SHIVAM PANWAR

INTRODUCTION

1. What is the main question your project seeks to answer?

Answer: My Project also known as the "Netflix predictor/Classifier" has the main motive to predict the movies that will be liked by the customers based on the most trending movies running in the market and considering the customer's taste. My project seeks to answer the question of what trending movies and shows will be liked by the customers based on their history. Considering all the movies or shows in a user's wish-list it will have to suggest him movies which will be best suited for him. Since it's just a suggestion a user may like or dislike the suggestion. Our project tries to improve the extender so that majority of the customers like the suggested trend.

2. Provide a brief motivation for your project question. Why is this question important? What can we learn from your project?

Answer: Netflix only suggests the shows or movies based on the movie or show you watched last and it suggests it with the same genre. What if a customer is craving for romantic or dramatic movie after watching a horror movie. I got the motivation based on the drawbacks of the current Netflix predictor. My project aims at improving those limitations and build a model which will gain customer's support and trust.

3. Briefly describe the data source(s) you have used in your project. Where is the data from? How big is the data in terms of data points and/or file size? If the data was not already available, how did you collect the data?

Answer: The dataset I used for my project is a .csv file of 2.4 MB size. I searched for the data sources in many websites and I could finally get my hands on this dataset from KAGGLE. It is a huge dataset enough to train my model which consist of roughly data from the year 1960-2020. I could not get it directly from the website Netflix because they do not release data openly.

SUMMARY OF EDA

1. What is the unit of analysis?

Answer:

```
In [3]: NetflixData.head(5)
```

Out[3]:

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	81145628	Movie	Norm of the North: King Sized Adventure	Richard Finn, Tim Maltby	Alan Marriott, Andrew Toth, Brian Dobson, Cole...	United States, India, South Korea, China	September 9, 2019	2019	TV-PG	90 min	Children & Family Movies, Comedies	Before planning an awesome wedding for his gra...
1	80117401	Movie	Jandino: Whatever it Takes	NaN	Jandino Asporaat	United Kingdom	September 9, 2016	2016	TV-MA	94 min	Stand-Up Comedy	Jandino Asporaat riffs on the challenges of ra...

My dataset consists of 12 parameters just shown as the above image. The main aim of our project is to predict whether a certain song will be liked by the customer judging by the genre of the show as well its rating.

So, the unit of analysis of our data set will be the column of popularity primarily.

We can measure the accuracy as

Accuracy = 1-(number of popular movies or shows missed to predict/ number of actual popular movies or shows liked by a person) * 100.

After studying and analyzing the dataset I found many parameters of not much use to my project and model designing. Therefore, I decided to trim the dataset and only keep the parameters which will be used in future.

2. How many observations in total are in the data set?

Answer:

The original data set before trimming

Number of rows: 3774

Number of columns: 12

```
In [12]: NetflixData.count()

Out[12]: show_id         3774
         type            3774
         title           3774
         director        3774
         cast            3774
         country         3774
         date_added      3774
         release_year    3774
         rating          3774
         duration        3774
         listed_in       3774
         description     3774
         dtype: int64
```

3. How many unique observations are in the data set?

Answer: In my dataset every movie and TV show has a unique key called the show_id. It is easy to locate the movie using the id as it is different for every single movie or show out there.

4. What time period is covered?

Answer: Movies and Shows from the year 1960 to 2020 are included in this dataset.

5. Briefly summarize any data cleaning steps you have performed

There weren't many null values in the column which were found irrelevant for us to predict a song liked by the customer or not. Therefore, we trimmed the dataset from having 12 columns to 6 columns.

I found 6 parameters to be very significant for the data set. I decided to trim the rest of the data as they had no relevance to what operations we are going to perform in designing the model.

Dropping Unused Column

```
In [22]: NetflixData = NetflixData.drop(['date_added','listed_in','description','cast','duration'], axis=1)
         train = NetflixData.drop(NetflixData, axis=1)
         NetflixData.head(5)
```

Out[22]:

	show_id	type	title	director	country	release_year	rating
0	81145628	Movie	Norm of the North: King Sized Adventure	Richard Finn, Tim Maltby	United States, India, South Korea, China	2019	TV-PG
4	80125979	Movie	#realityhigh	Fernando Lebrija	United States	2017	TV-14
6	70304989	Movie	Automata	Gabe Ibáñez	Bulgaria, United States, Spain, Canada	2014	R
7	80164077	Movie	Fabrizio Copano: Solo pienso en mi	Rodrigo Toro, Francisco Schultz	Chile	2017	TV-MA
9	70304990	Movie	Good People	Henrik Ruben Genz	United States, United Kingdom, Denmark, Sweden	2014	R

Data set after trimming
Number of rows: 3774
Number of columns: 6

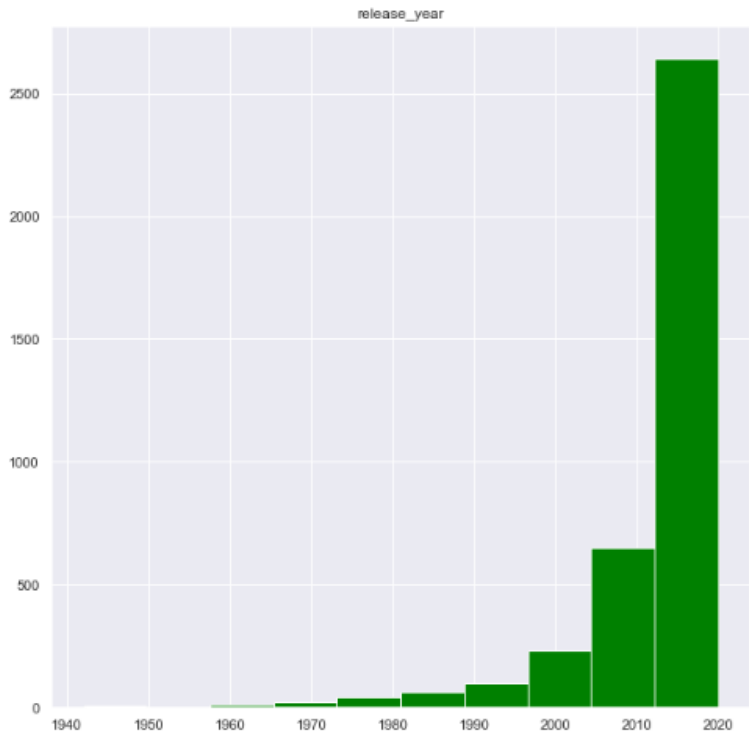
6. Visualization of the response with an appropriate technique.

Answer:

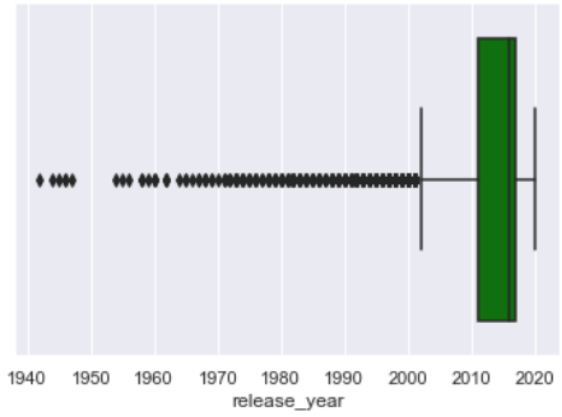
Below is the illustration of a histogram which shows the numbers of shows and movies released in what year. It seems like as the rime moves on there is a steady increase in the movies as people are now preferring to watch on Netflix rather on any other source.

Maximum Releasing Years on netflix

```
In [14]: NetflixData.hist(column='release_year', color = "green", figsize=(10,10))
Out[14]: array([[<matplotlib.axes._subplots.AxesSubplot object at 0x1a16b4d090>]],
               dtype=object)
```

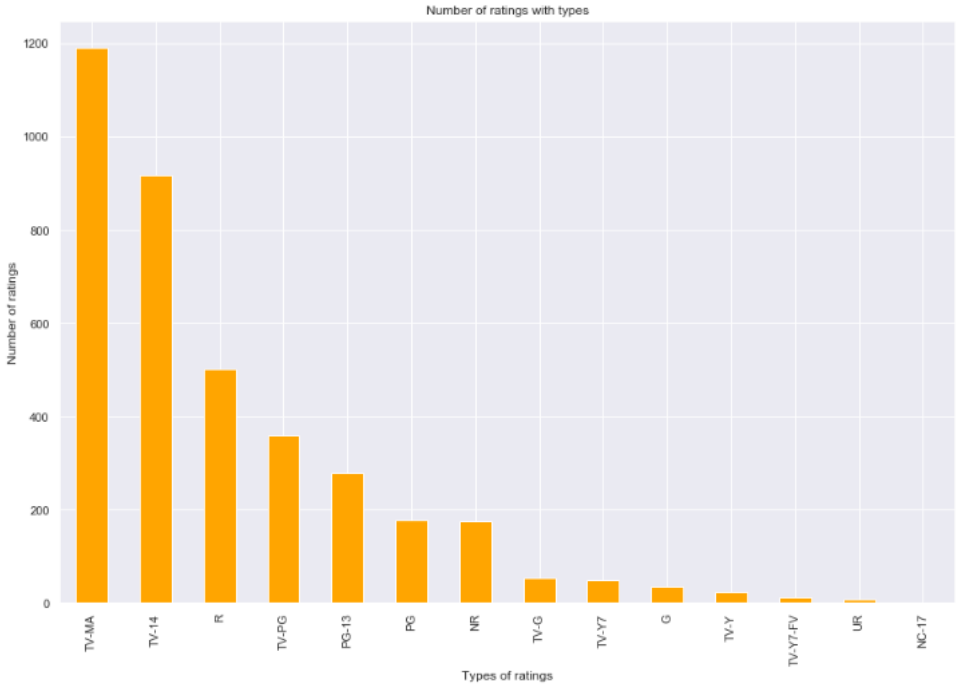



```
In [15]: sns.boxplot(x=NetflixData['release_year'], color = "green")
Out[15]: <matplotlib.axes._subplots.AxesSubplot at 0x1a17d62590>
```



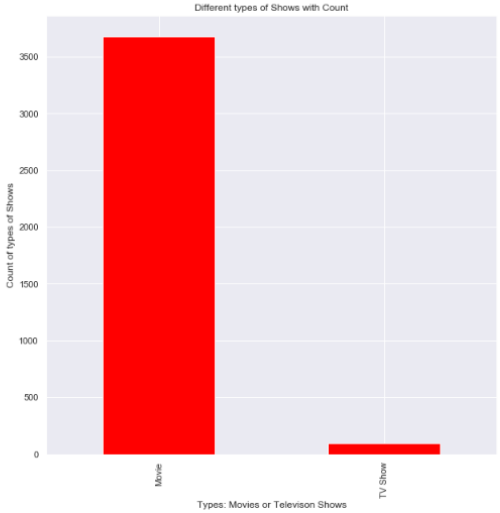
Above is the illustration of the box plot which tells us about the maximum number of movies introduced by Netflix is between the year 2010-2017. This clearly tells us that movies released in those years will clearly have the most trending one and people would love to watch them given them the suggestion at the correct time.

```
In [17]: NetflixData.rating.value_counts().nlargest(40).plot(kind='bar', figsize=(15,10), color = "orange")
         plt.title("Number of ratings with types")
         plt.ylabel('Number of ratings')
         plt.xlabel('Types of ratings');
```

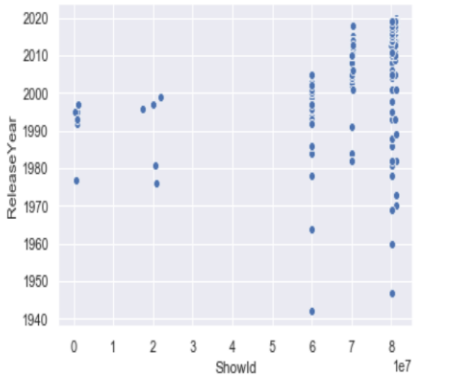


A histogram which shows a plot of Types of ratings vs number of ratings. This gives us a brief an idea about what genre movies are introduced by Netflix and in what number. We can plot a brief summary about which movie is the most liked according to the ratings. So, I consider ratings as one of the most important aspect of my project model.

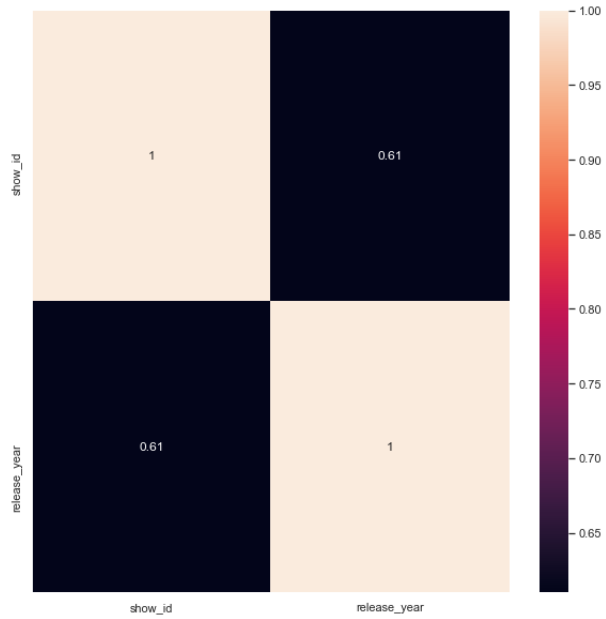
```
In [19]: NetflixData.type.value_counts().nlargest(10).plot(kind='bar', figsize=(10,10), color = "red")
         plt.title("Different types of Shows with Count")
         plt.ylabel('Count of types of Shows')
         plt.xlabel('Types: Movies or Televison Shows');
```




```
Out[29]: <matplotlib.axes._subplots.AxesSubplot at 0x1a19273790>
```



Above graph shows a bar plot between number of movies vs shows Netflix has. It can be clearly seen that there is a huge gap between movies which is leading by a great margin. People have a variety of choice of movies than TV shows.



A simple heat map plotted to find co-relation between different parameters.

7. Visualization of key predictors against the response (e.g., scatterplot, boxplot, etc.). Pick one or two predictors that you think are going to be most important in explaining the response. Your selection of predictors can either be guided by your domain knowledge or be the result of your EDA on all predictors.

Answer:

K-neighbors Classifier model:

In the EDA I observed that the parameter genre and rating is the most important aspect of the model. The main reason I decided to go with th K-neighbor classifier is because it considers the objects which have similar attributes and properties. n *k-NN classification*, the output is a class membership. An object is classified by a plurality vote of its neighbors, with the object being assigned to the class most common among its *k* nearest neighbors (*k* is a positive integer, typically small). If *k* = 1, then the object is simply assigned to the class of that single nearest neighbor. In *k-NN regression*, the output is the property value for the object. This value is the average of the values of *k* nearest neighbors.

Decision Tree Classifier:

Decision tree learning is one of the predictive modeling approaches used in statistics and data mining and machine learning. It uses a decision tree (as a predictive model) to go from observations about an item (represented in the branches) to conclusions about the item's target value (represented in the leaves). Tree models where the target variable can take a discrete set of values are called classification trees; in these tree structures, leaves represent class labels and branches represent explicit features that lead to those class labels.

This model will fit good for the data because it will be making decisions based on the parameters, I give it. A good decision made might help the customer get what he/she wants at the right time.

SUMMARY OF MACHINE LEARNING MODELS

1. Justify your model choices based on how your response is measured and any observations you have made in your EDA.

Answer:

In the EDA I observed that the parameter genre and rating are the most important aspects of my model. The main reason I decided to go with th K-neighbor classifier is because it considers the objects which have similar attributes and properties.

- n *k-NN classification*, the output is a class membership. An object is classified by a plurality vote of its neighbors, with the object being assigned to the class most common among its *k* nearest neighbors (*k* is a positive integer, typically small). If $k = 1$, then the object is simply assigned to the class of that single nearest neighbor.

- In *k-NN regression*, the output is the property value for the object. This value is the average of the values of *k* nearest neighbors.

Decision Tree Classifier Model:

Decision tree plots the choice according to the choices we give to the model. After inducing certain conditions.

Using the decision algorithm, we start at the tree root and split the data on the feature that results in the largest information gain **(**IG**)** (reduction in uncertainty towards the final decision).
In an iterative process, we can then repeat this splitting procedure at each child node until the leaves are pure. This means that the samples at each leaf node all belong to the same class.
In practice, we may set a **l**imit on the depth of the tree to prevent overfitting. We compromise on purity here somewhat as the final leaves may still have some impurity.

Now after considering the observation made from the EDA, a conclusion has been made to consider K nearest neighbor algorithm for model selection. The main reason why they had to choose this model is because this algorithm will consider all the nearby songs which is close to the most popular and trending movie or show to suggest the same to the customers. All popular movies and shows are clubbed together to form a group of most trending set. The other models I have considered is the LIGHTGBM and Neural network. For current instance we have considered these models to fit our data well and have a look how well it predicts other measures.

2. Report the results from at least two different models:
    - For each model, report the model's test error. Justify your choice.
    - For each model, discuss how well the model fits the data.

Answer:

K-neighbor classifier:

In [28]:
```
#KNeighbors Classifier

from sklearn.neighbors import KNeighborsClassifier

knn = KNeighborsClassifier(n_neighbors=100, metric='euclidean')
knn.fit(x_train, y_train)

# knn_pred = c.predict(x_test)
knn_pred = knn.predict(x_test)

score= accuracy_score(y_test, knn_pred) * 100

print("Accuracy using Knn Tree: ", round(score, 1), "%")
```

```
Accuracy using Knn Tree:  47.3 %
```

In [30]:
```
print('Mean Absolute Error:', metrics.mean_absolute_error(y_test, knn_pred))
print('Mean Squared Error:', metrics.mean_squared_error(y_test, knn_pred))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test, knn_pred)))
```

```
Mean Absolute Error: 2.6455026455026456
Mean Squared Error: 64.65255731922399
Root Mean Squared Error: 8.040681396450427
```

In [38]:
```
df=pd.DataFrame({'Actual':y_test, 'Predicted':y_pred})
df
```

Out[38]:

	Actual	Predicted
805	1973	2018
5580	2015	2015
3634	2016	2017
2942	2018	2017
1484	2015	2018
...	...	...
4762	2015	2015
839	2005	2007
2919	2019	2018
2041	2016	2017
3071	1988	2007

The model fits decent with the dataset. It is not perfect, but it also does not overfit. Overfitting would have made my model look bad as I would have got poor results for my

prediction. The results above show that the model has been trained well and the results are pretty good as compared to the other two models. K-neighbor model has the highest accuracy or 48% among the all the models I implemented and tested therefore I decided to go with this model as my final choice.

	ShowId	type	title	director	country	ReleaseYear	rating
0	81145628	Movie	Norm of the North: King Sized Adventure	Richard Finn, Tim Maltby	United States, India, South Korea, China	2019	TV-PG
4	80125979	Movie	#realityhigh	Fernando Lebrija	United States	2017	TV-14
6	70304989	Movie	Automata	Gabe Ibáñez	Bulgaria, United States, Spain, Canada	2014	R
7	80164077	Movie	Fabrizio Copano: Solo pienso en mi	Rodrigo Toro, Francisco Schultz	Chile	2017	TV-MA
9	70304990	Movie	Good People	Henrik Ruben Genz	United States, United Kingdom, Denmark, Sweden	2014	R

Decision Tree Classifier:

In [35]:
```
# Create Decision Tree classifer object
clf = DecisionTreeClassifier(criterion="entropy", max_depth=3)

# Train Decision Tree Classifer
clf = clf.fit(X_train,y_train)

#Predict the response for test dataset
y_pred = clf.predict(X_test)
```

In [36]:
```
# Model Accuracy, how often is the classifier correct?
print("Accuracy using Decision Tree Model:",metrics.accuracy_score(y_test, y_pred)*100)
```

```
Accuracy using Decision Tree Model: 34.68667255075022
```

In [37]:
```
print('Mean Absolute Error:', metrics.mean_absolute_error(y_test, y_pred))
print('Mean Squared Error:', metrics.mean_squared_error(y_test, y_pred))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test, y_pred)))
```

```
Mean Absolute Error: 3.0335392762577227
Mean Squared Error: 64.06266548984996
Root Mean Squared Error: 8.003915634853353
```

The decision tree is the second-best model among the three models I try to implement and deploy.

ADA Boost Classifier model:

In [41]:
```
from sklearn.ensemble import AdaBoostClassifier

ada = AdaBoostClassifier(n_estimators=3)
ada.fit(x_train, y_train)

ada_pred = ada.predict(x_test)

from sklearn.metrics import accuracy_score

score = accuracy_score(y_test, ada_pred) * 100
print("Accuracy using AdaBoost Model: ", round(score, 2), "%")
```

```
Accuracy using AdaBoost Model:  22.75 %
```

In [42]:
```
print('Mean Absolute Error:', metrics.mean_absolute_error(y_test, ada_pred))
print('Mean Squared Error:', metrics.mean_squared_error(y_test, ada_pred))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test, ada_pred)))
```

```
Mean Absolute Error: 3.2980599647266313
Mean Squared Error: 56.2962962962963
Root Mean Squared Error: 7.503085784948503
```

The model does not fit very well with the dataset as compared to the other two models.

SUMMARY AND CONCLUSION

Summary:

In my group project also named the "Netflix predictor/classifier" data set, model must predict the most upcoming trending movies and shows which will be liked by the customers in future or give them suggestions for their playlist. After the Exploratory Data analysis, I saw that genre and rating are the main parameter that needs to be considered along with the history of the client's movies or shows watched. With the help of these three parameters, there is a need to select a model that fits well with the data set and predicts accurately with the prediction of upcoming trending movies.
From my project, I have successfully learnt to plot graphs of a given dataset. I have learnt how to do the exploratory data analysis. I can choose from a plethora of existing classifiers the one which would suit my data the most and give the most optimal results. I learned about the classifiers.

1. Going back to the question that has motivated your project, how would you answer that question given the results of your analysis?

   Answer: The model I implemented has a decent accuracy but not the best. I feel that neural networks with convolution layers and feature maps would have done much better. The NN model would have understood the pattern of trending songs from the dataset and without even much error it would have presented a better result than our current model. I am happy about our model that it did not overfit the training data. If NN could have been implemented it could be considered in real time application as well because deep learning can be very productive in such extender applications. I would even consider LIGHTGBM as an option to improvise my project.

2. Think about domain experts in the field you have analyzed. What can they learn from your project? How could the results of your analysis inform their work?

   Answer: The traditional extender only suggests the movies and shows that are have being listened to recently by the customers or have been watched most the times in a wish-list. My deployed model introduces functions which can capture movies with different genre as well. Since my model does not overfit the training data it produces optimal results with the test data.

3. Identify one way that your project could be improved if you had more time and resources to work on this project. For example, what additional data would you gather? What alternative data cleaning decisions would you make? What additional models would you estimate?

   Answer: I could have deployed the model of neural networks if given additional time. I would have trained the NN model with more than 100 datasets of Netflix movies and shows. NN results could have excelled our model by miles as we would have achieved excellent results of prediction. Keras libraries and tensor flow convolution layers might have uplifted my project with optimal and better results than our current existing model.