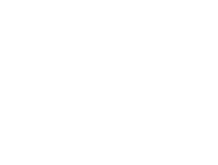


**Fall**



18

ITMD 513 – Open Source Programming

**PREDICTING SUCCESS PERFORMANCE ON IMDB DATA**

**Sadaa Sree Ravichandar (A20395954)**

**I l l i n o i s I n s t i t u t e O f T e c h n o l o g y**

**Table of Contents**

[**I. Introduction 3**](#_Toc520624634)

[**II. Scraping the data from the IMDB website using Python 3**](#_Toc520624635)

[**III. Data Pre-Processing and Classification 4**](#_Toc520624636)

[**IV. Login Credentials 4**](#_Toc520624637)

[**V. Video Link of the Application 4**](#_Toc520624638)

[**VI. Application Functionality & Snapshots 5**](#_Toc520624639)

[**VII. Summary Stats 28**](#_Toc520624640)

[**VIII. Extra Credits 29**](#_Toc520624643)

[**IX. Future Scope & Limitations 34**](#_Toc520624644)

[**X. Conclusions & Observations 35**](#_Toc520624649)

# Introduction

IMDb, an abbreviation of Internet Movie Database, is an [online database](https://en.wikipedia.org/wiki/Online_database) of information related to world films, television programs, home videos and video games.

We refer the IMDB rating every time we want to watch a movie or series online. Without the IMDB rating viewers would find it difficult to select a movie to watch, not knowing whether the movie is good movie or a bad movie.

In this project we build the best classifier that predicts the IMDB Rating based on the meta-score and the number votes for a given movie. Since the meta-score and the IMDB ratings have a positive linear relationship it would be appropriate to predict an IMDB rating based on the meta-score. Also, when comparing votes with IMDB ratings a movie or a video with maximum number of votes would either have a higher rating or vice versa. Hence, we can say that votes and IMDB rating also have a steady relationship which in turn is again appropriate to predict an IMDB rating based on the number of votes received for the movie or video or a game.

# Scraping the data from the IMDB website using Python

We want to analyze the distributions of [IMDB](http://www.imdb.com/) and [Metacritic](http://www.metacritic.com/) movie ratings

to build the classifier. To do this the first 1000 movies of the IMDB website has been scrapped. Since each page contained 50 movies the number on how many pages must be scraped was worked out to get a list of 1000 -1500 movies. After identifying the URL structure and the HTML structure of a single page and using BeautifulSoup to parse the HTML content the data for a single movie was extracted.

To scrap multiple pages on the IMDB website all the requests were sent within the loop and the loop’s rate was controlled to avoid bombarding the server with requests. Also, to avoid hammering the server with more number of requests per second and also to pave way for other user’s request we control the loop’s rate using sleep() function. The following parameters were monitored to avoid all the discrepancies.

The **frequency (speed) of requests**, to make sure the program is not overloading the server.

The **number of requests**, to halt the loop in case the number of expected requests is exceeded.

The [**status code**](https://en.wikipedia.org/wiki/List_of_HTTP_status_codes) of the requests, to make sure the server is sending back the proper responses.

# Data Pre-Processing and Classification

The extracted data from the IMDB website for the given years or given months must be pre-processed before we perform classification process. The meta-score and votes are the X variables and the IMDB rating is the Y variable for which prediction must be made.

To perform classification process on the IMDB data with meta-scores and votes the values of these two variables should fall in the same scale. Hence, we use min max transformation to transform the values within the same range so that the prediction of the IMDB rating would give best results.

After pre-processing, the data is split into 80% percent train data and 20% test data. Based on the accuracy metric we decide the best classifier and predict the IMDB rating using the best classifier. At the end of this project a classifier would be built that can predict an IMDB rating given the meta-score and number of votes for a given movie.

# Login Credentials

Username -itmd513, python

Password – 1017,1009

# Video Link of the Application

* 1. Extracting the IMDB data and building the best classifier based on the given years

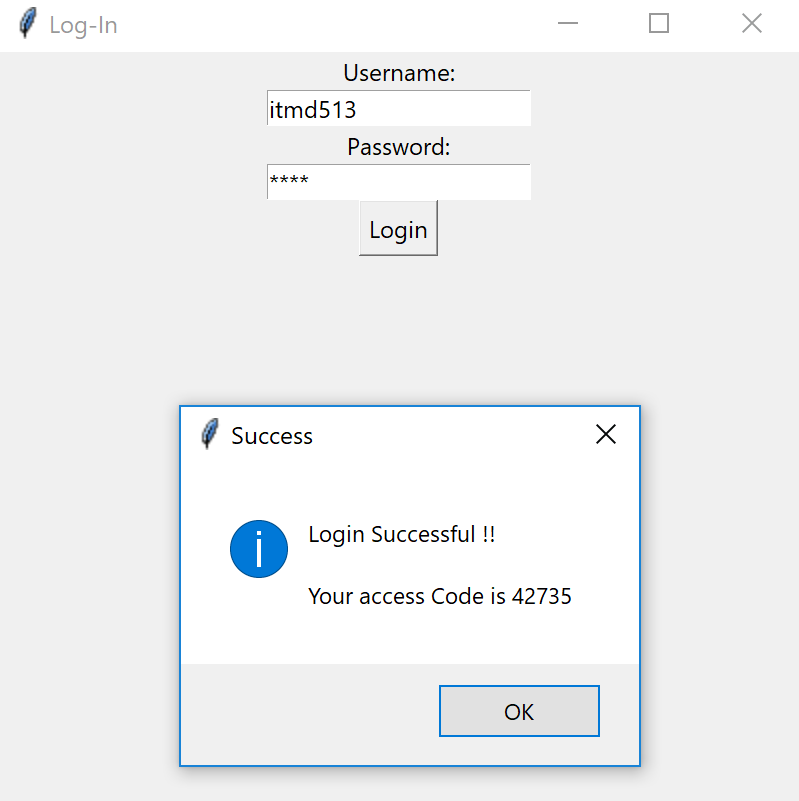
[<https://drive.google.com/file/d/17POLZhjfGRXhRGaDR_GQbasX9YayhlpJ/view?usp=sharing>](https://drive.google.com/file/d/1AWs5IP5L5VLqCDgX1Qt4hr7zFYo0fdLm/view?usp=sharing)

* 1. Extracting the IMDB data and building the best classifier based on a particular range of dates

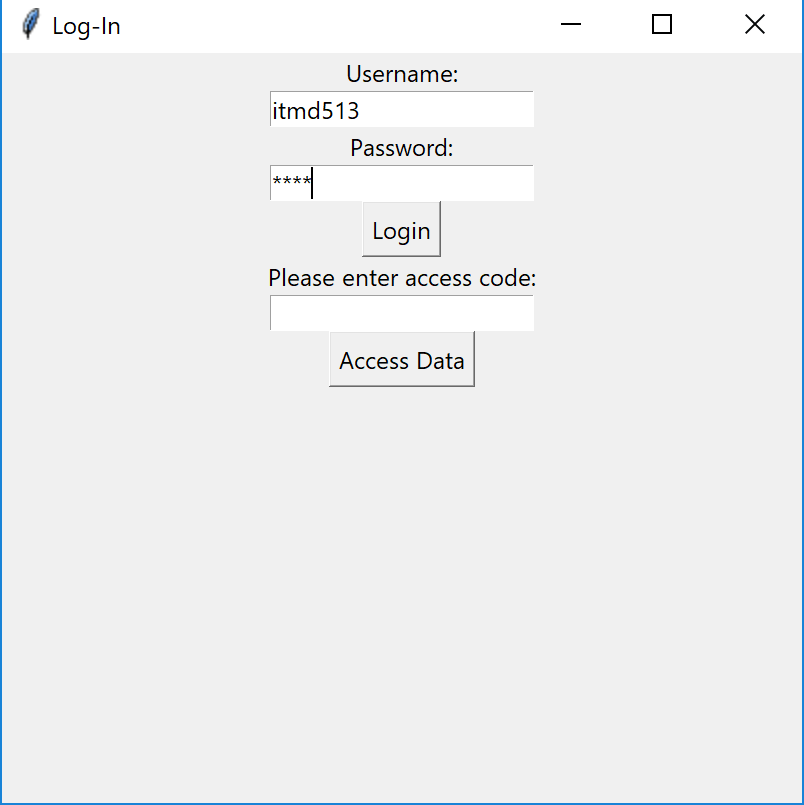
<https://drive.google.com/file/d/1AWs5IP5L5VLqCDgX1Qt4hr7zFYo0fdLm/view?usp=sharing>

# Application Functionality & Snapshots

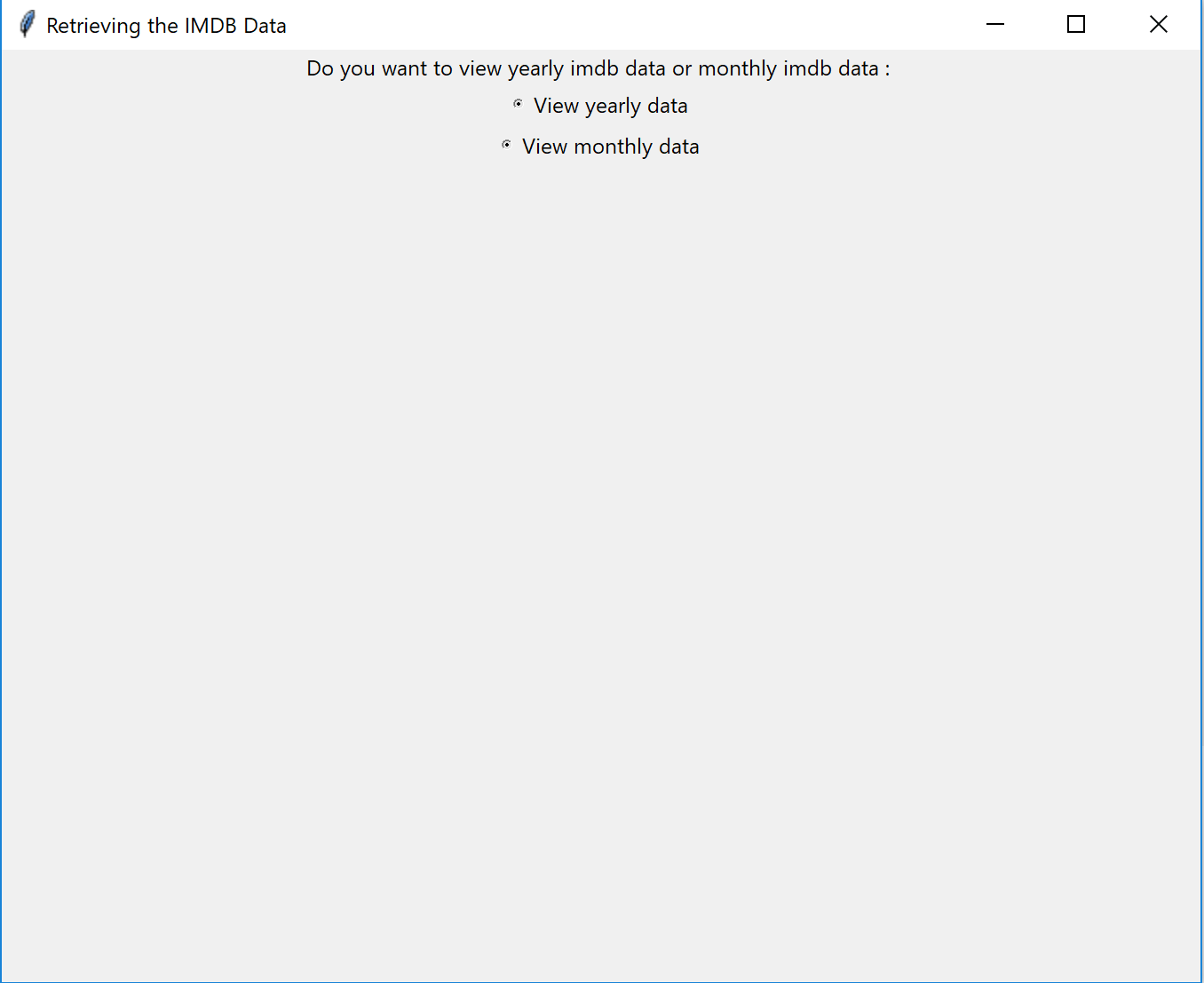
* 1. **LOGIN**
     + 1. User entering the correct username and password for which the user is given with an access code.



* + - 1. The user is then prompted to enter the access code that was displayed earlier for which the user would get access to the data

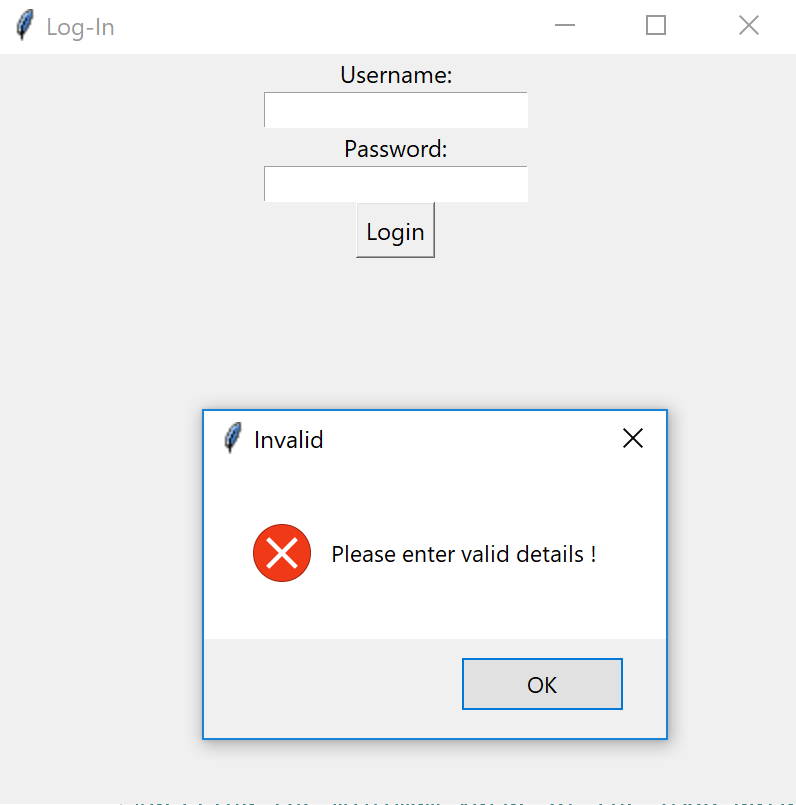


* + - 1. After entering the access code, the user is given an option whether he/she wants to retrieve the IMDB data for a range of years or for a range of dates

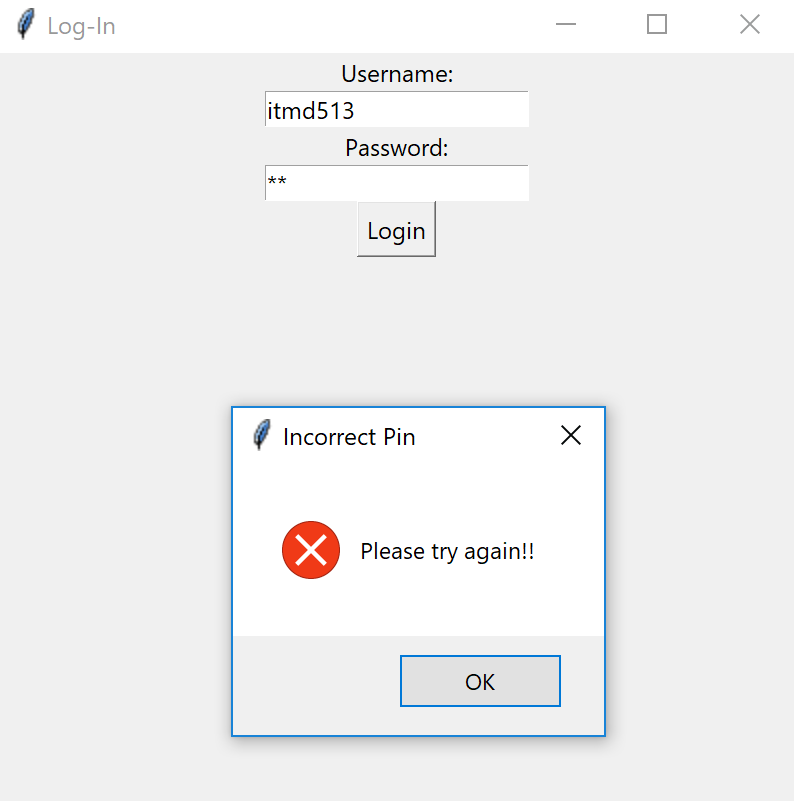


* 1. **ERROR VALIDATION IN ‘LOGIN’ PAGE**

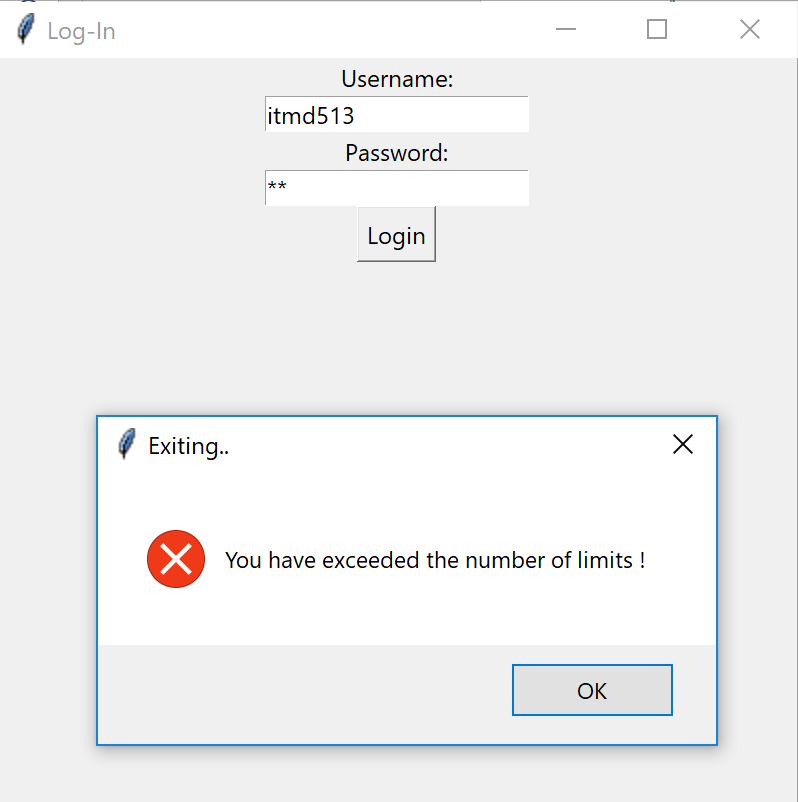
1. When the username and password are left blank an error message is thrown



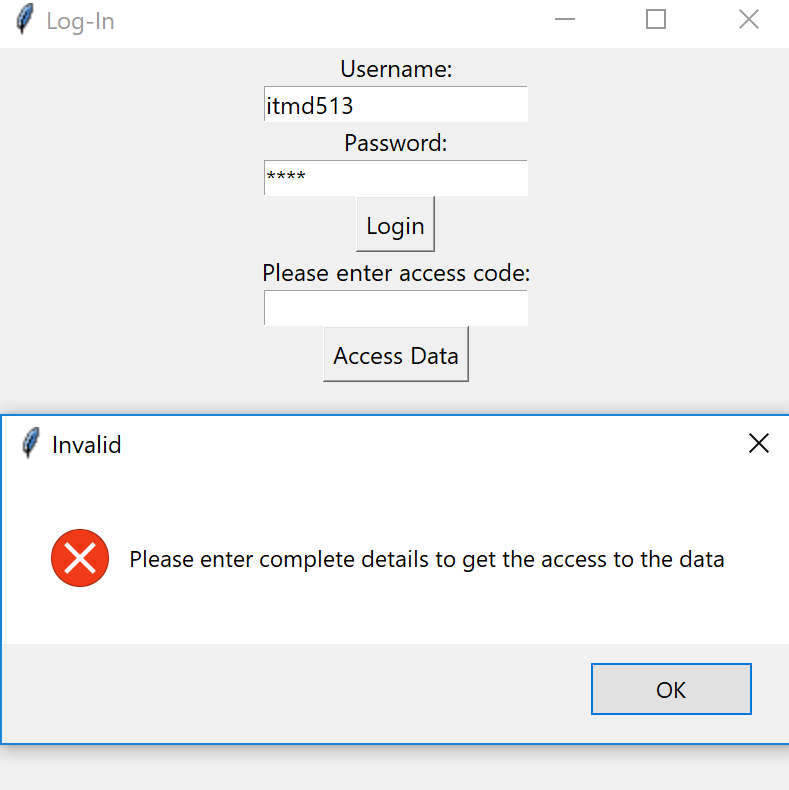
1. When an incorrect pin is entered an error message like shown in the image is thrown



1. When the maximum number of limits for allowing the user to enter the correct pin has exceeded an error message is thrown



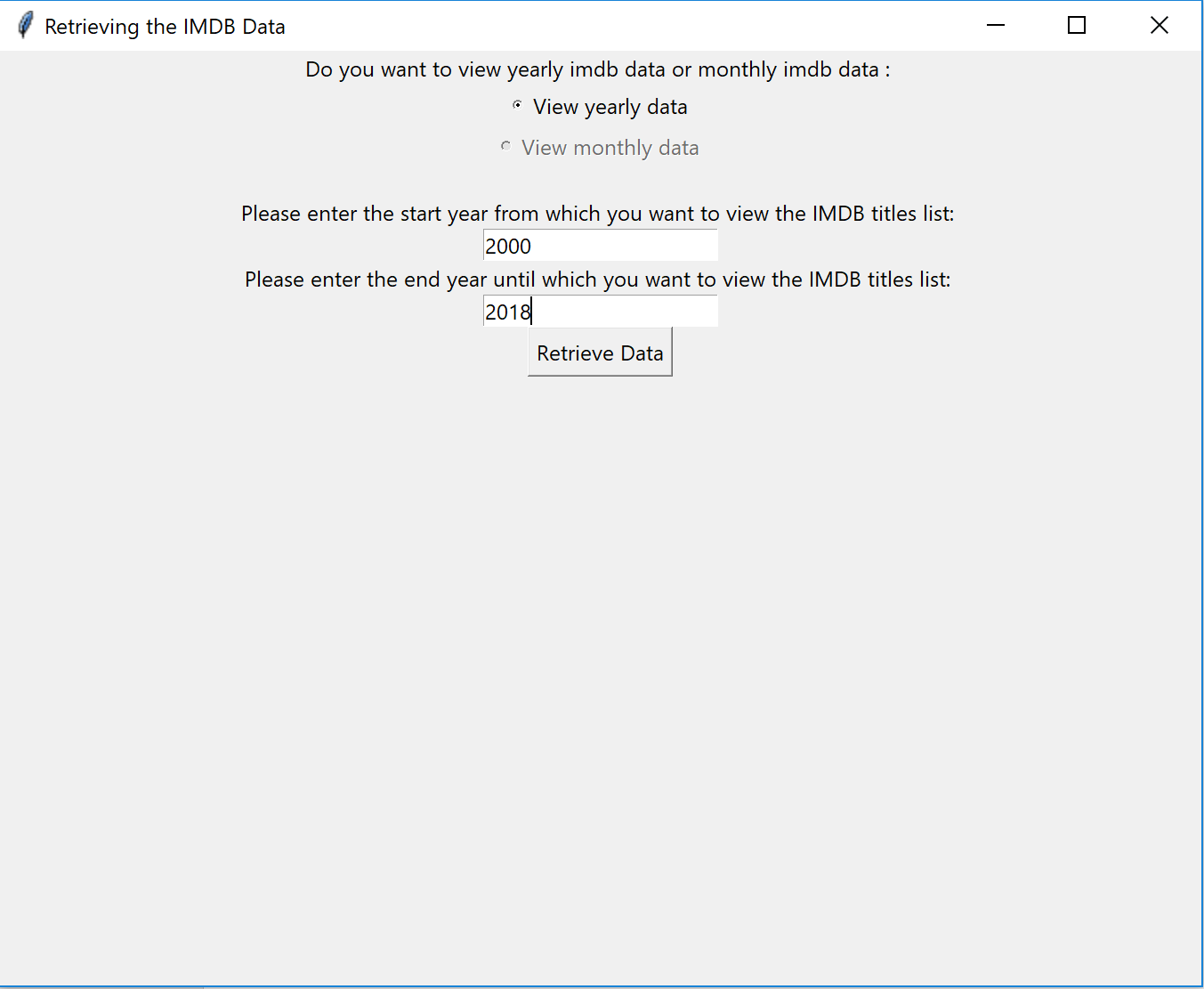
1. Only when all the complete details are given, the user can access the data



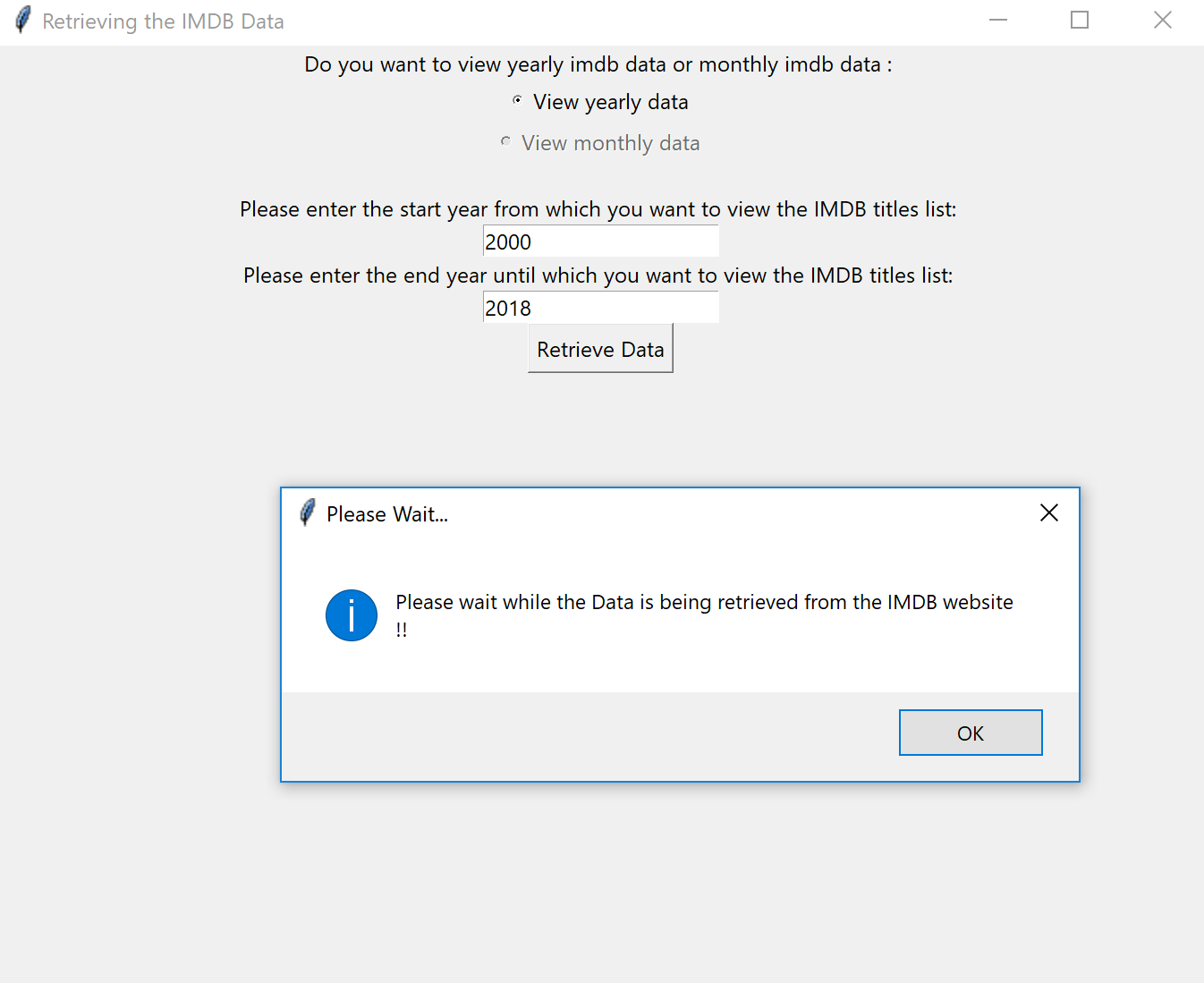
* 1. **RETRIEVING THE IMDB DATA – BASED ON YEARS**

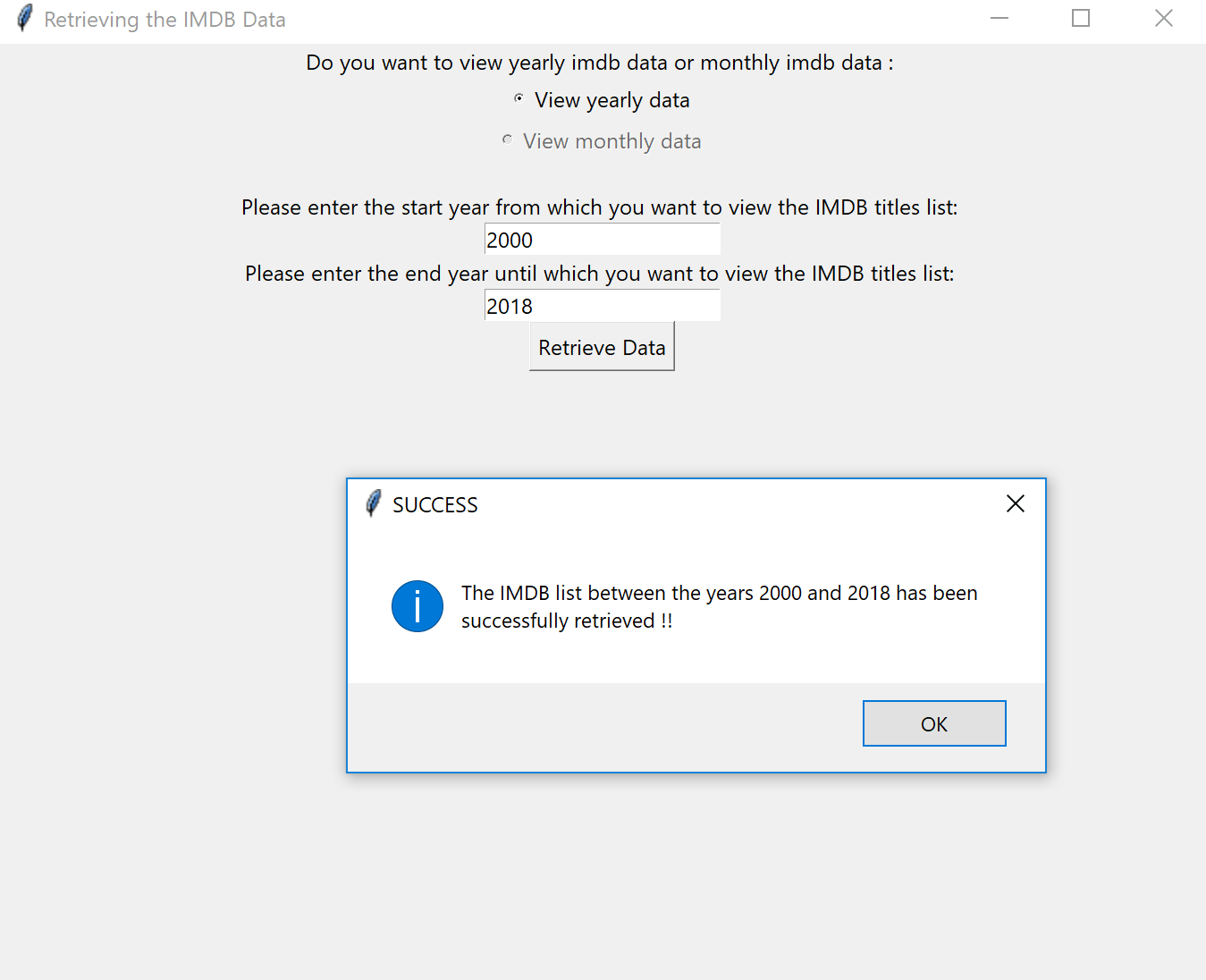
**Getting the IMDB data by entering a range of years and building a classifier that predicts the IMDB rating based on the meta-score and votes retrieved from the IMDB website**

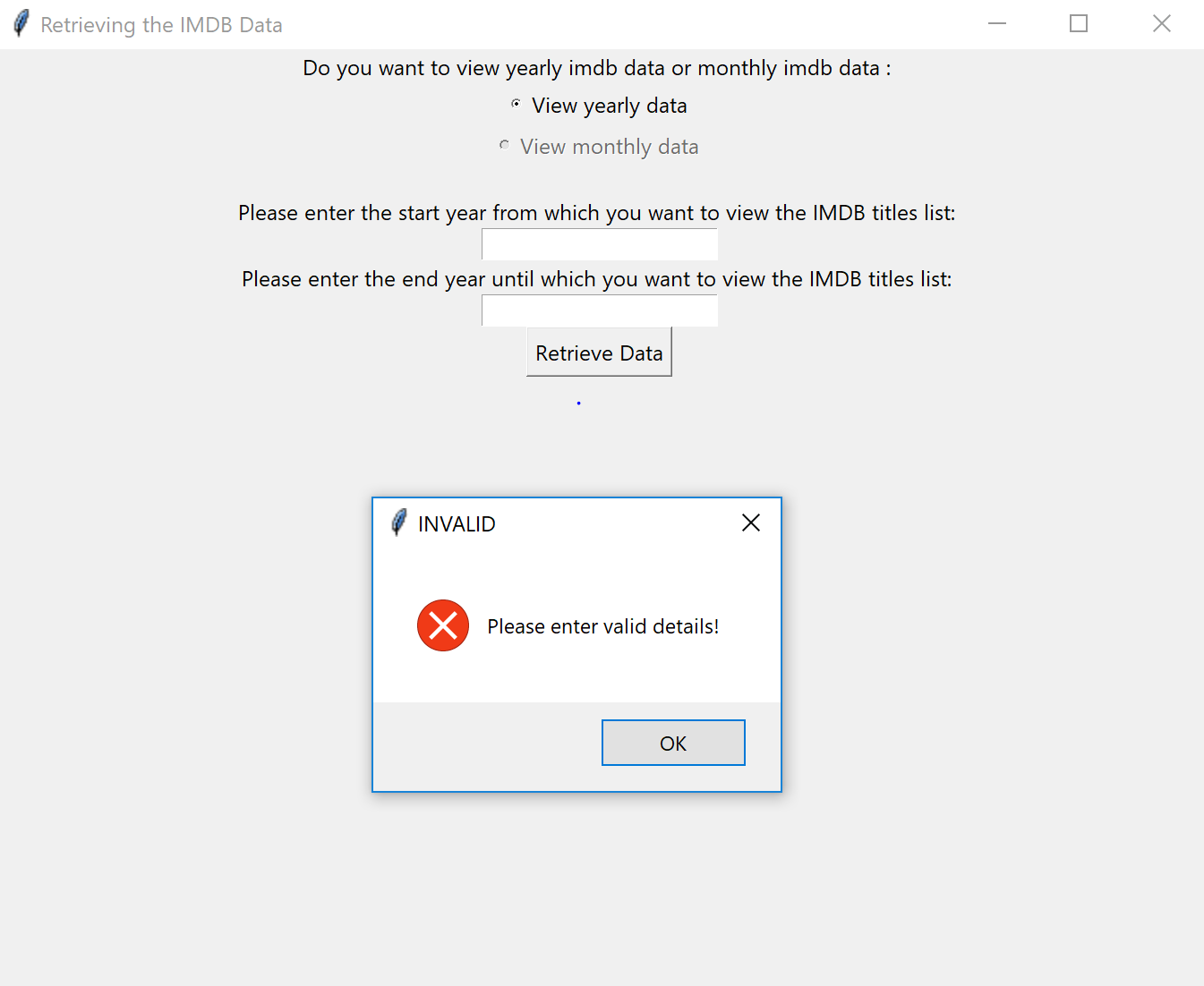
1. To get the yearly data the “View yearly data” is selected which then prompts the user to enter the start and end year for which you wish to retrieve the IMDB data



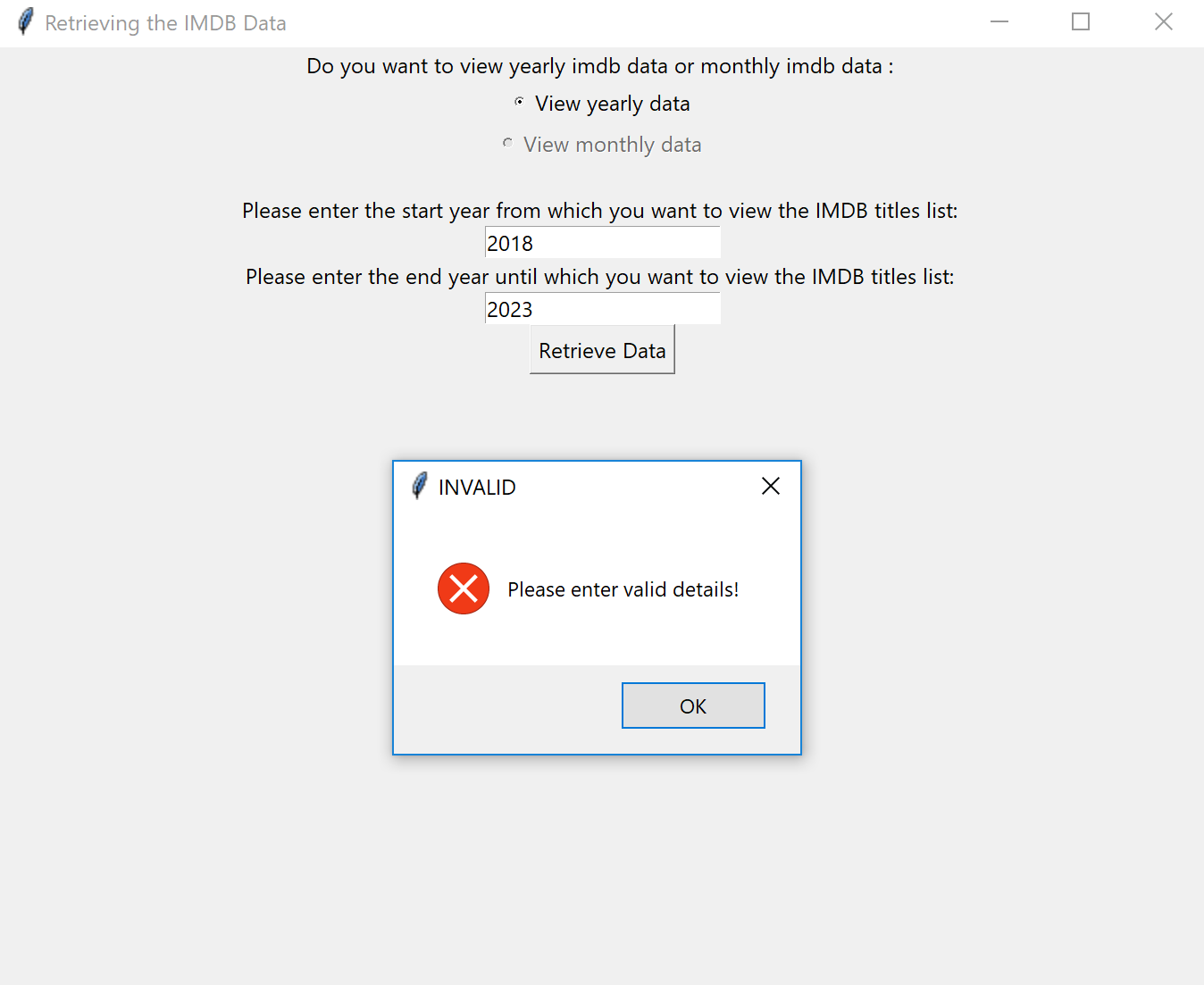
1. On clicking retrieve data the IMDB list of movies are being retrieved from the IMDB website



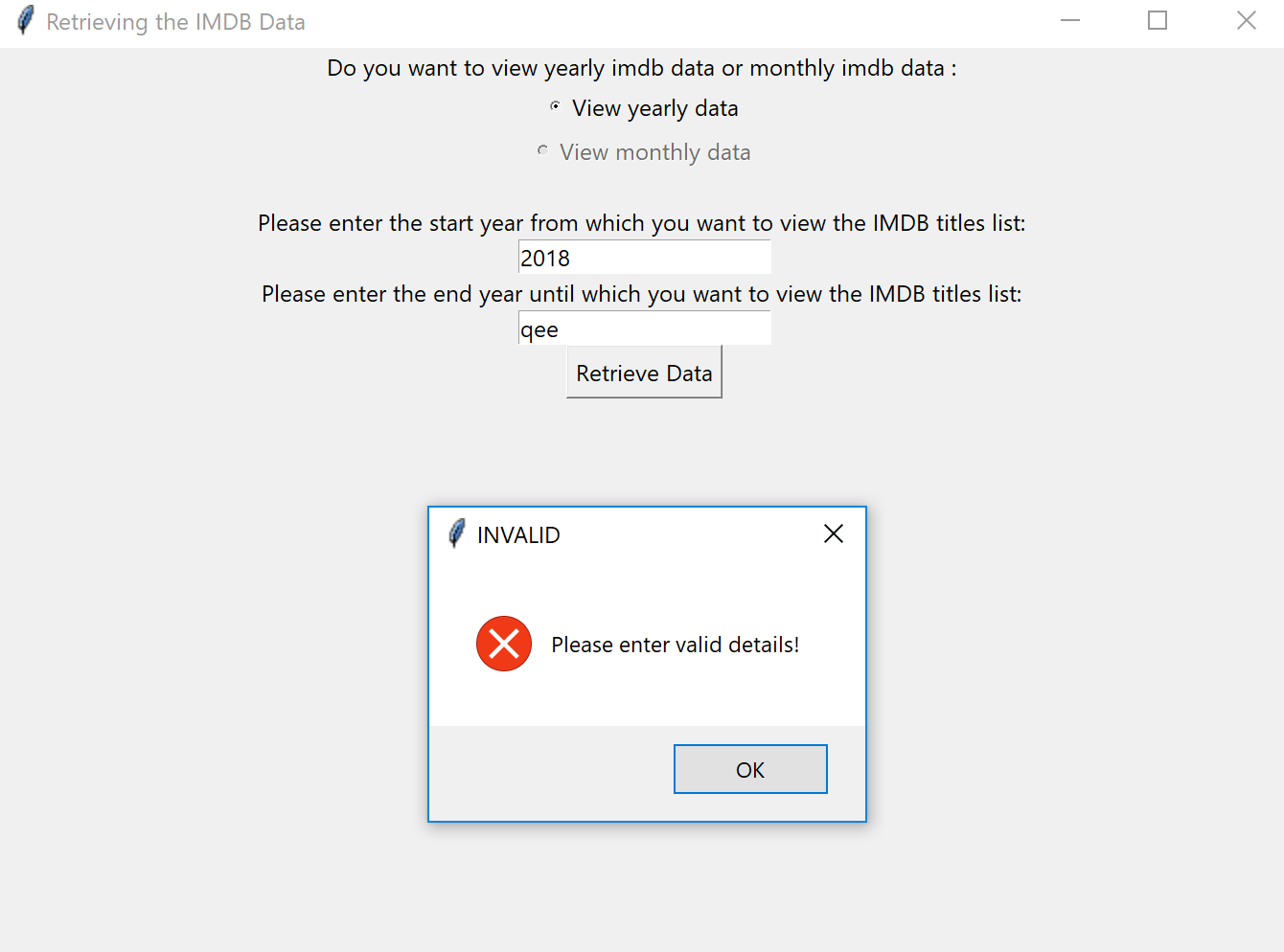
1. After the data is successfully retrieved from the website a message is displayed as shown below
   1. **ERROR VALIDATION IN ‘RETRIEVING THE IMDB DATA’ PAGE**
2. Error message is displayed when no details are entered for the range of years.



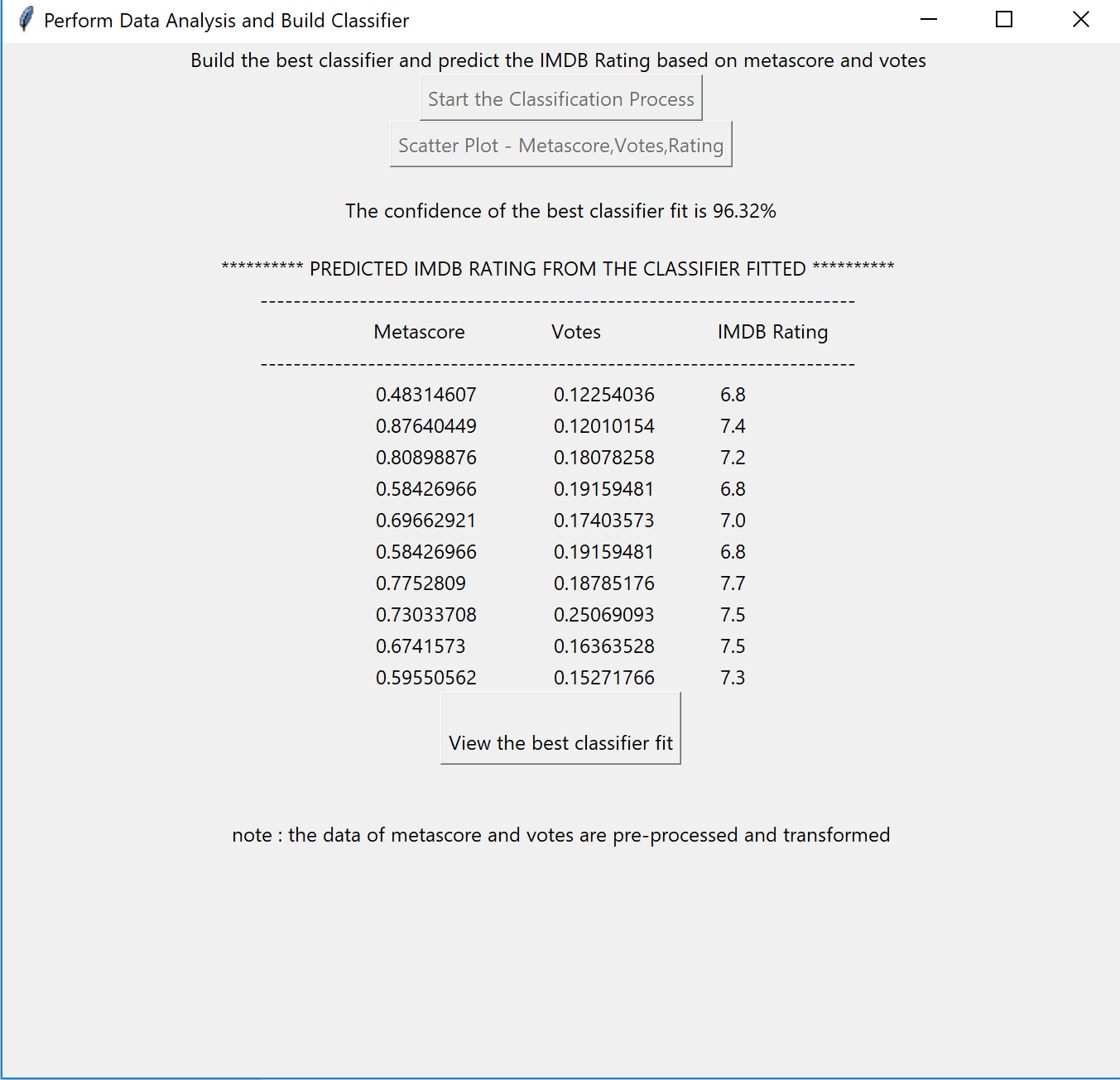
1. Error message is displayed when invalid details are entered for the range of years.(end year cannot be more than current year)



1. Error message is displayed when invalid details are entered for the range of years. (Entering alphabets instead of a year)

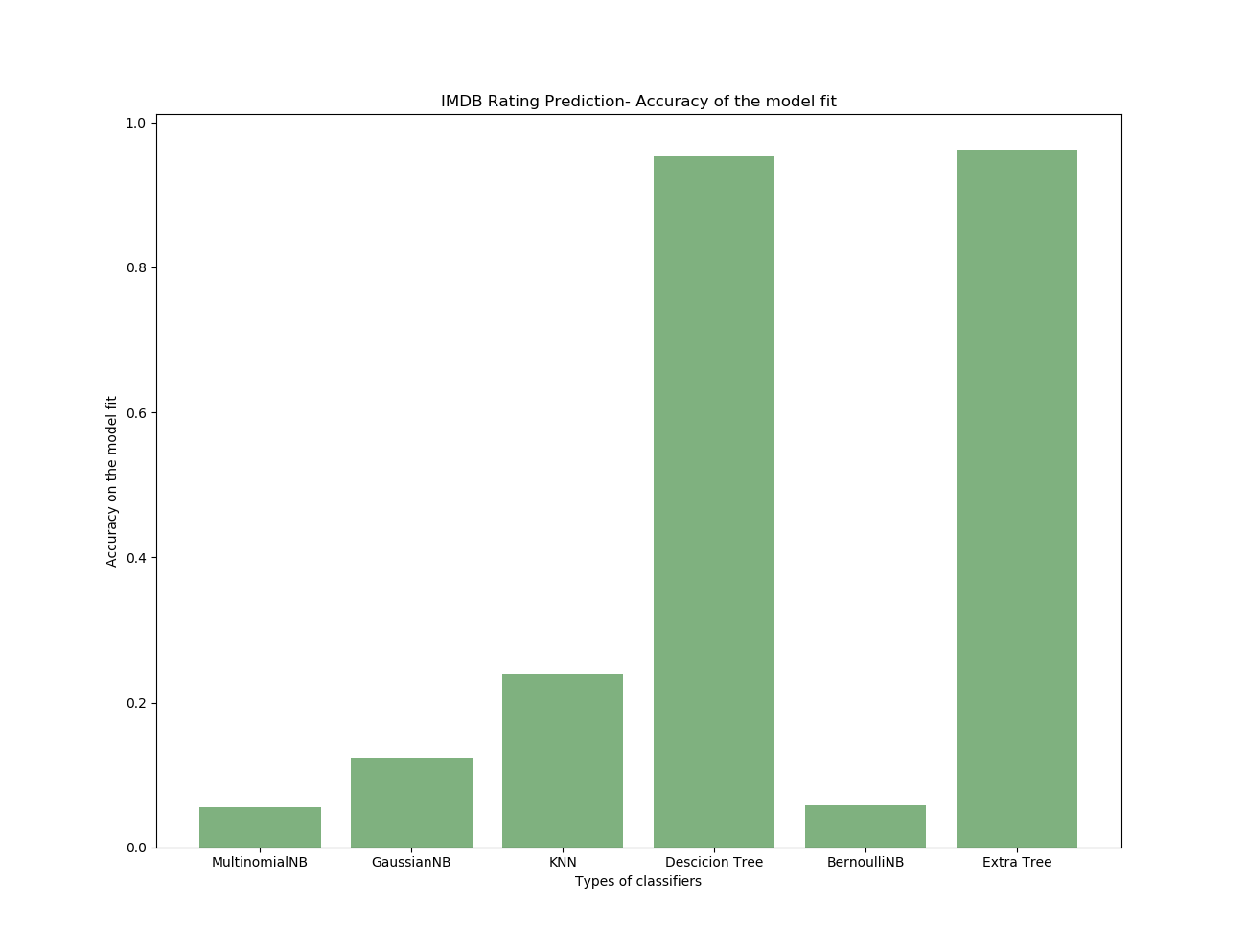


1. **PERFORM DATA ANALYSIS AND BUILD CLASSIFIER**
2. We can find the best classifier fit has an accuracy of 96.32% and we can also see the predicted IMDB ratings for the unknown data. Here the meta-score and votes have been transformed using max min scalar transformation since they had different range of values



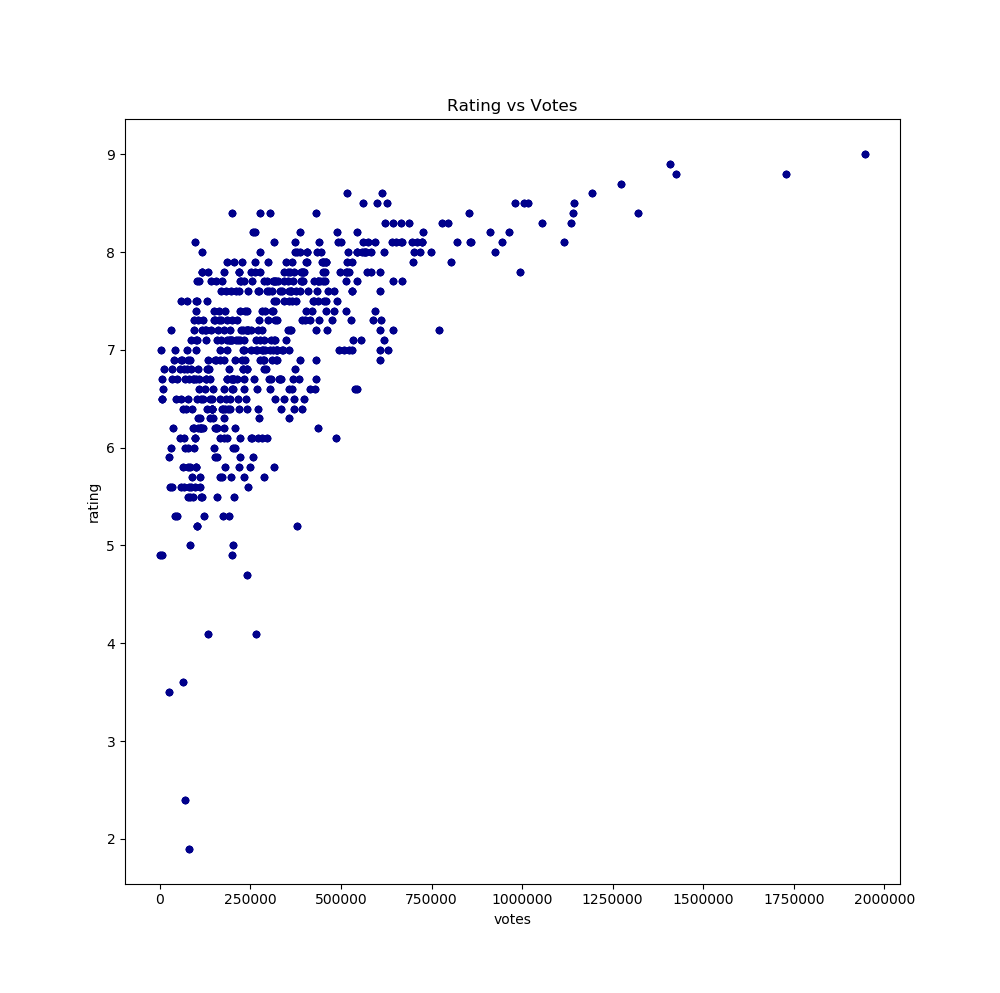
1. **ACCURACY CHART**
2. Classifiers built are Multinomial Naïve bayes, Gaussian Naïve bayes, KNN, Decision Tree, Bernoulli Naïve bayes and Extra Tree classifier

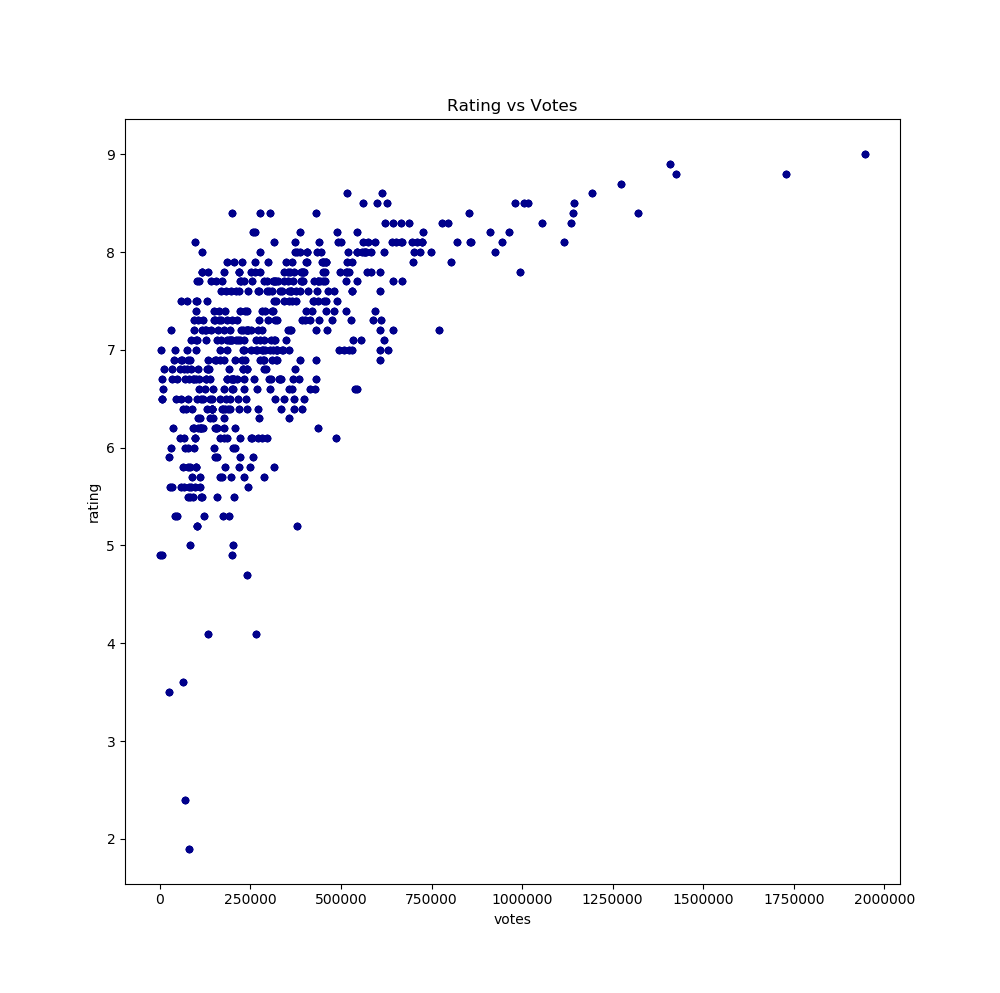
**We can see here that the best classifier built is Extra Tree Classifier**



1. **SCATTER PLOT**
2. The scatter plot between rating and votes

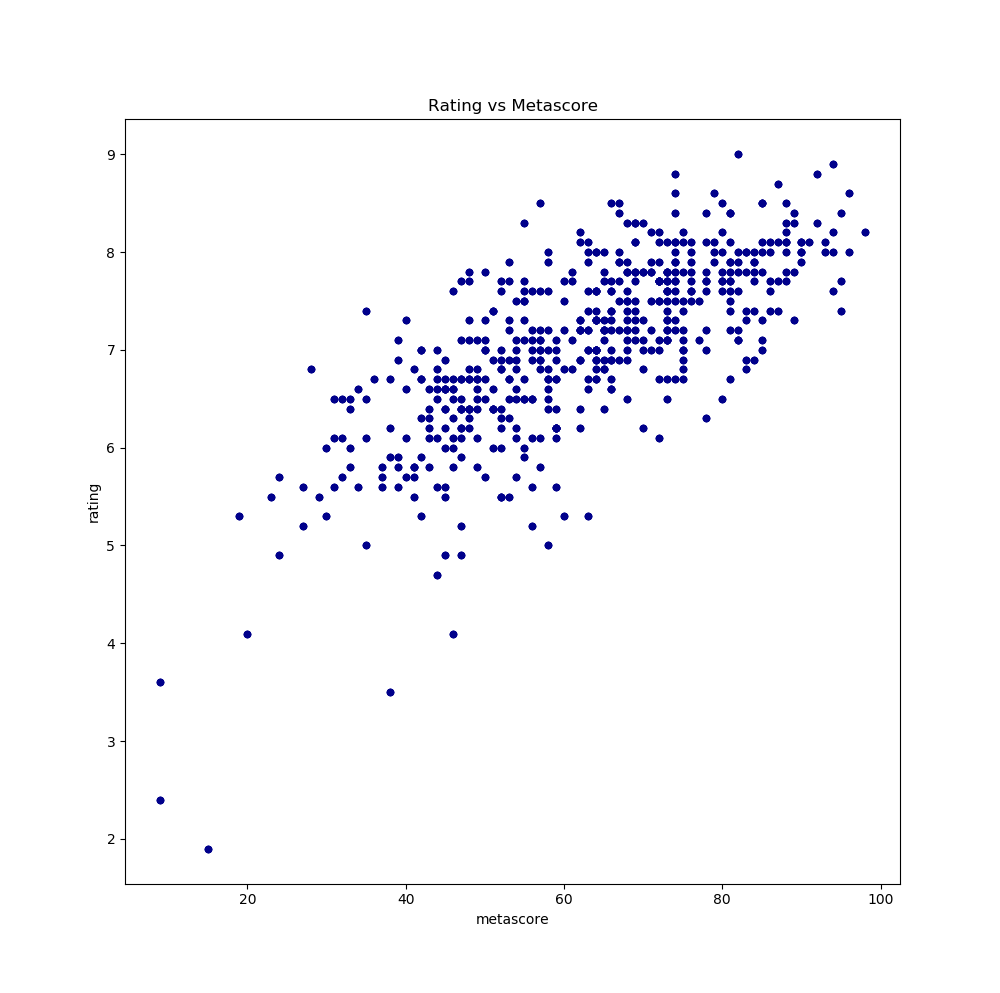
**We can observe that as the number of votes increase the IMDB rating increases indicating strong positive linear relationship**



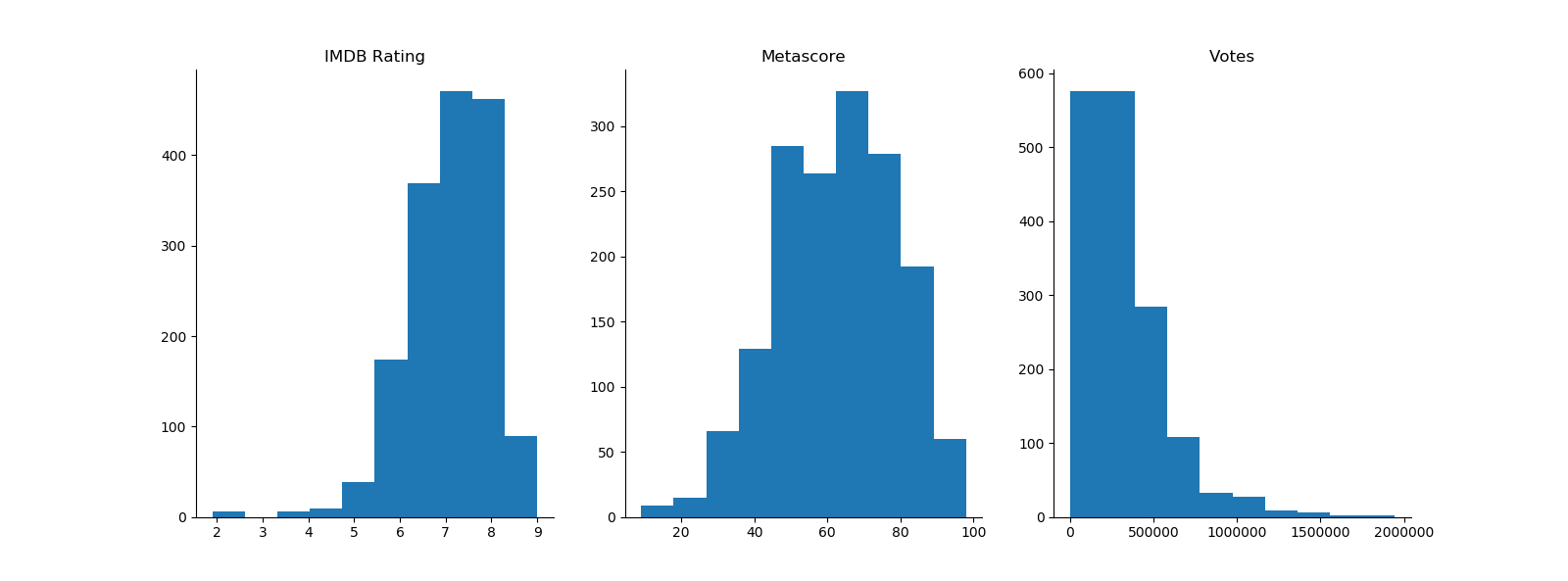


1. The scatter plot between rating and meta-scores

**As meta-score increases the IMDB rating increases indication strong positive linear relationship**



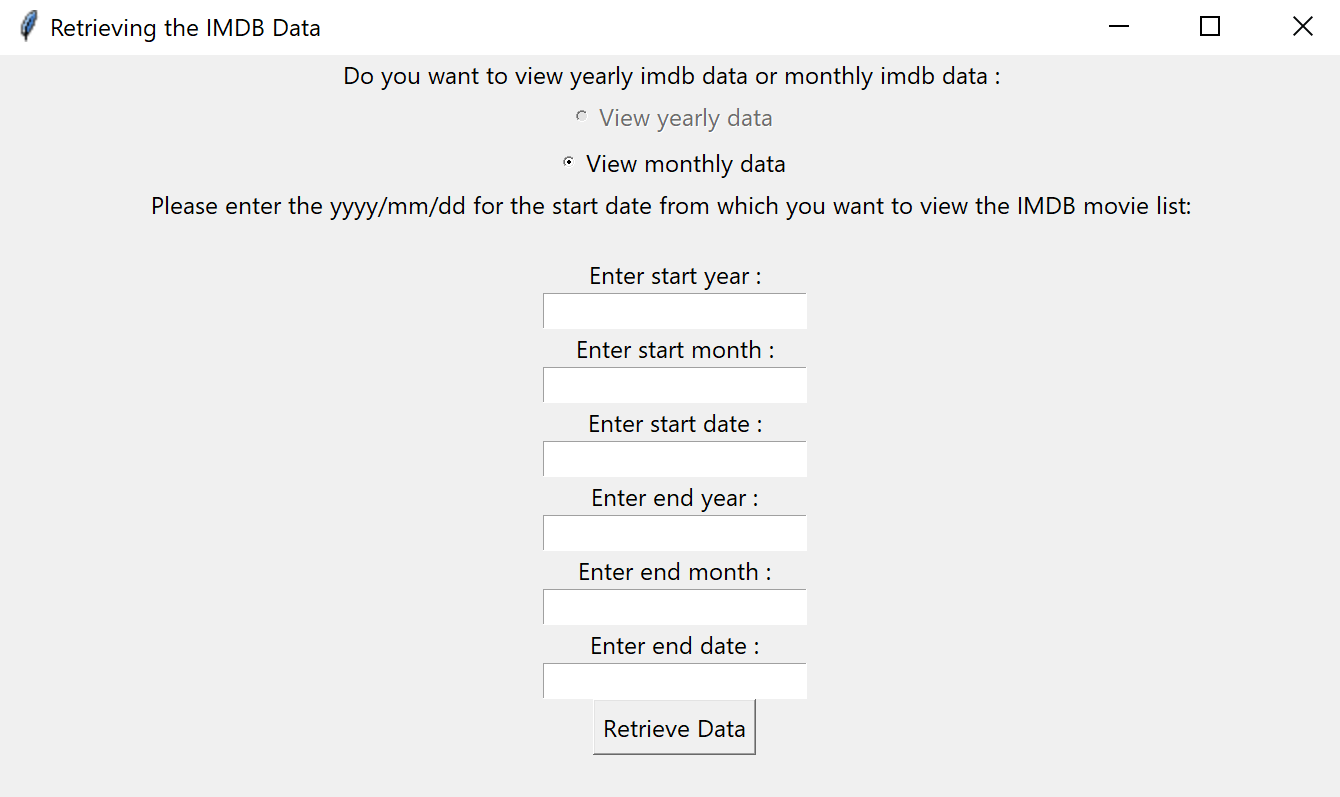
1. The histogram of Meta-score ,IMDB rating and votes (EXTRA CREDITS)

**We can observe that the IMDB rating values are slightly left skewed while meta-score shows close to normal distribution and the votes are right skewed.**

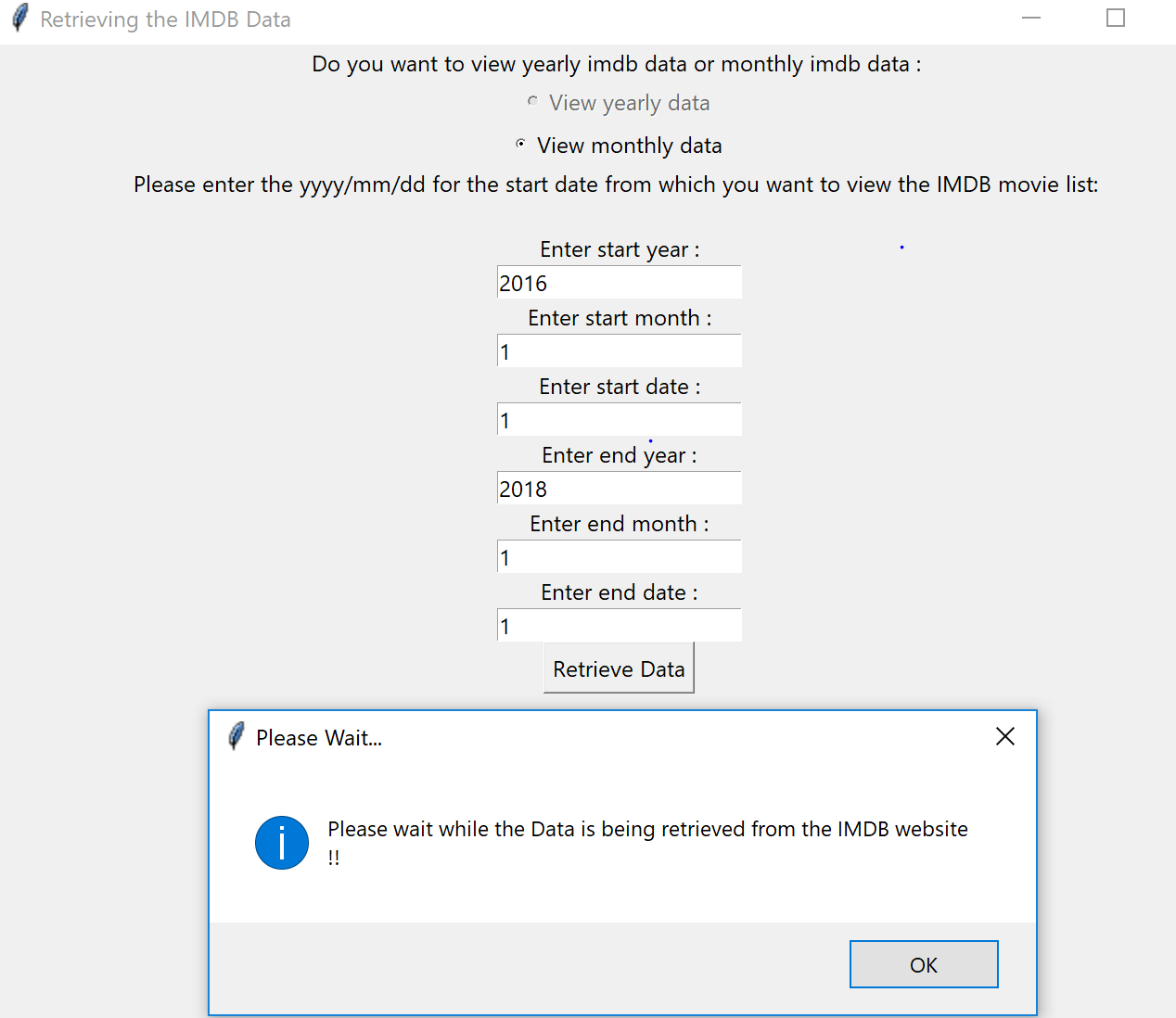
1. **RETRIEVING THE IMDB DATA – BASED ON DATES**

**Getting the IMDB data by entering a range of dates and building a classifier that predicts the IMDB rating based on the meta-score and votes retrieved from the IMDB website**

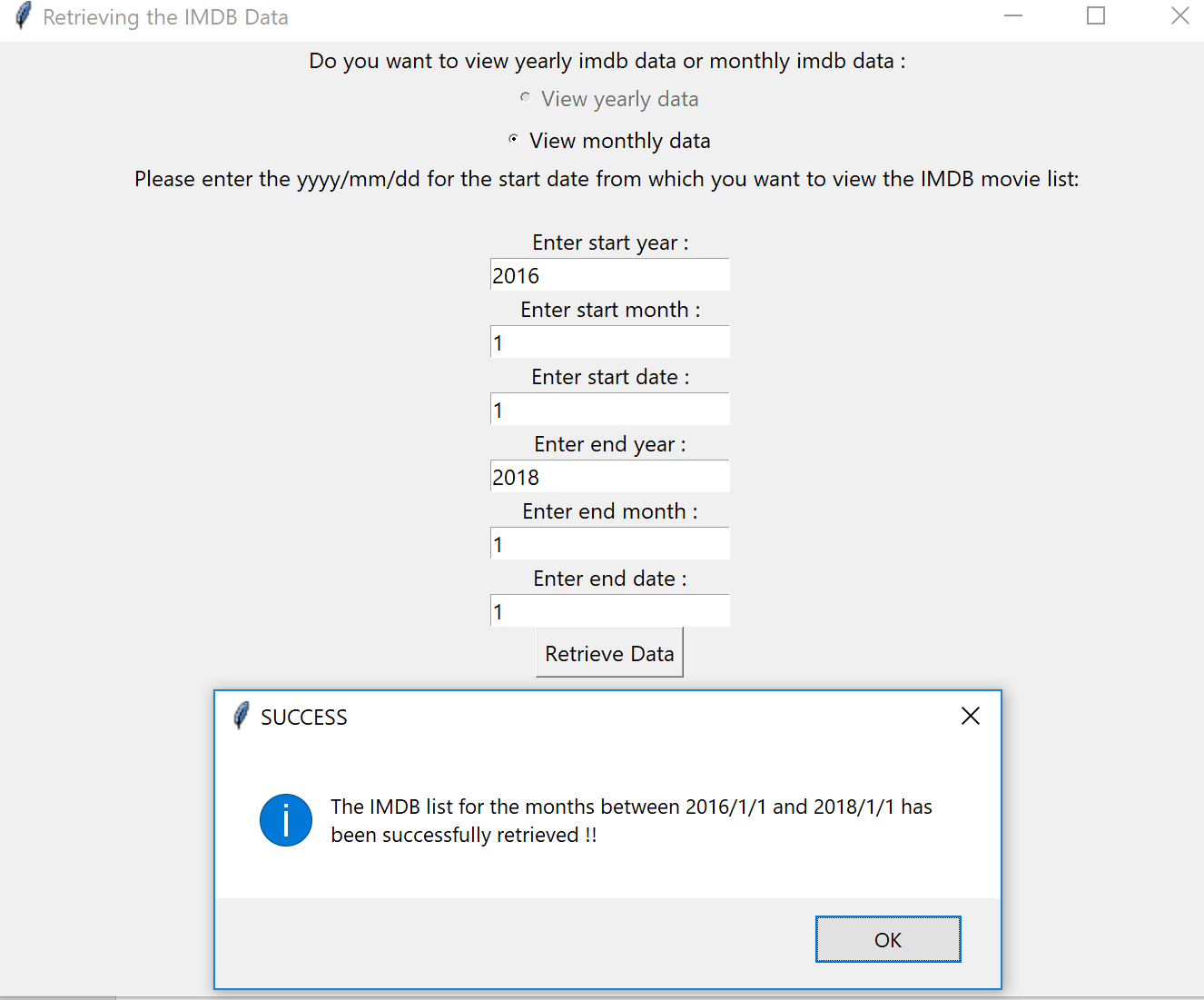
1. To get the yearly data the “View monthly data” is selected which then prompts the user to enter the start and end year for which you wish to retrieve the IMDB data



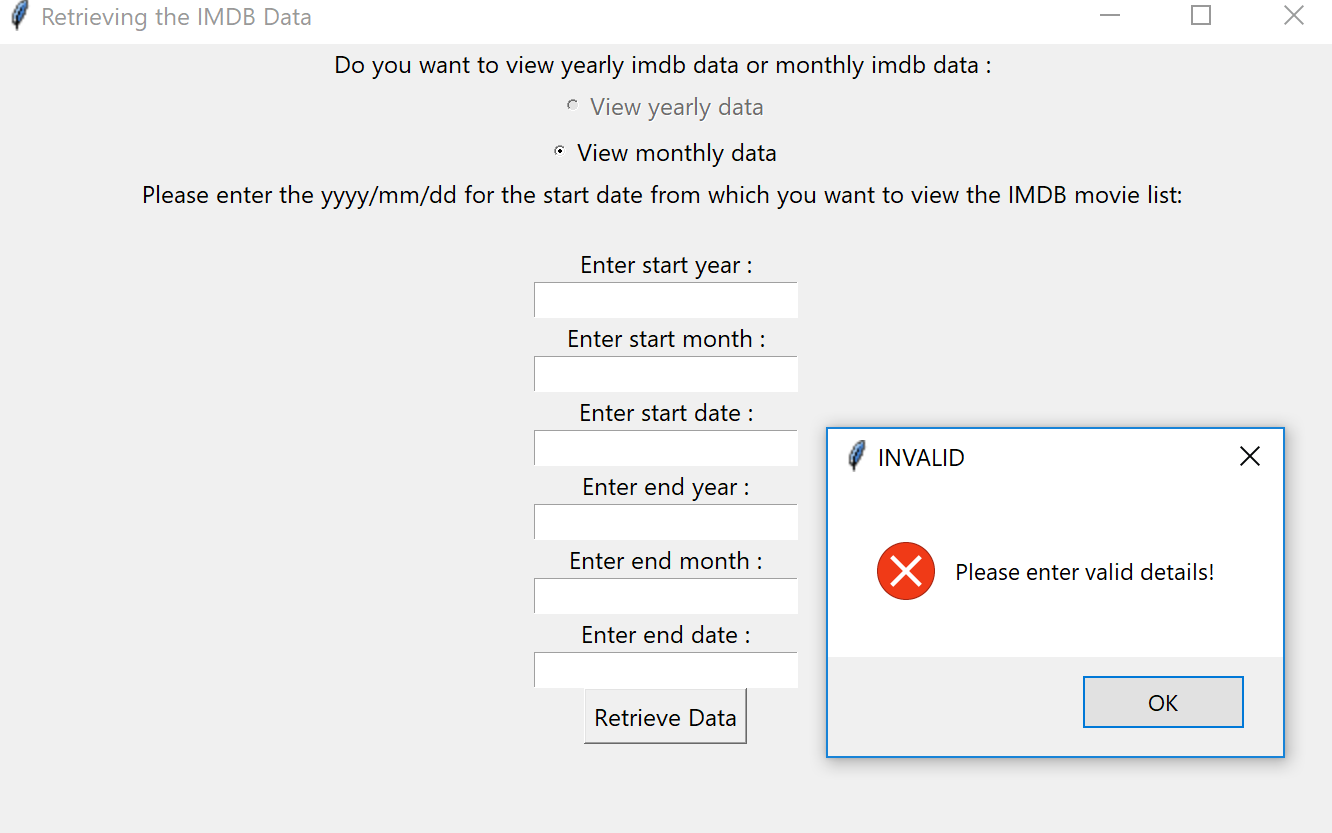
1. On clicking retrieve data the IMDB list of movies are being retrieved from the IMDB website



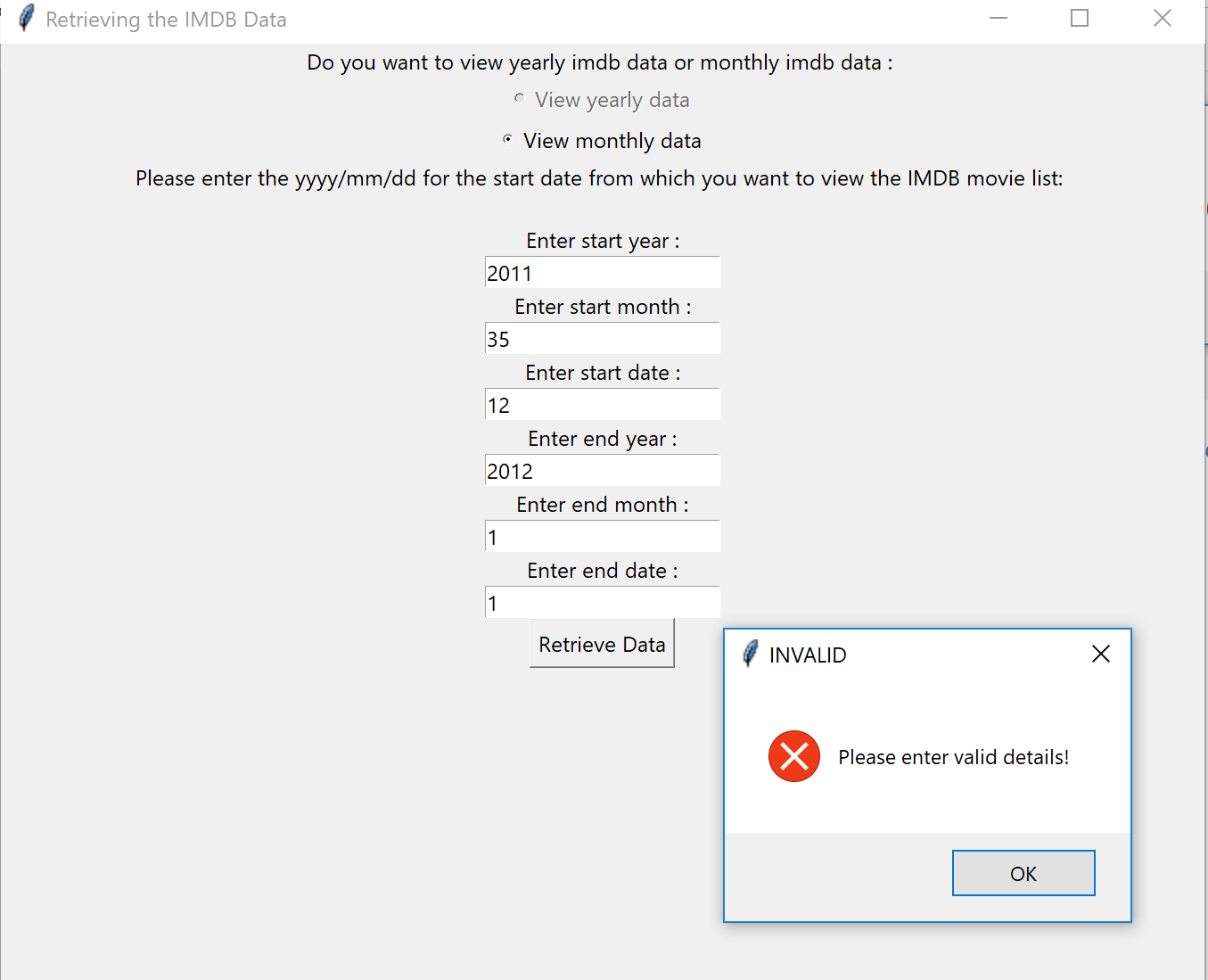
1. After the data is successfully retrieved from the website a message is displayed as shown below



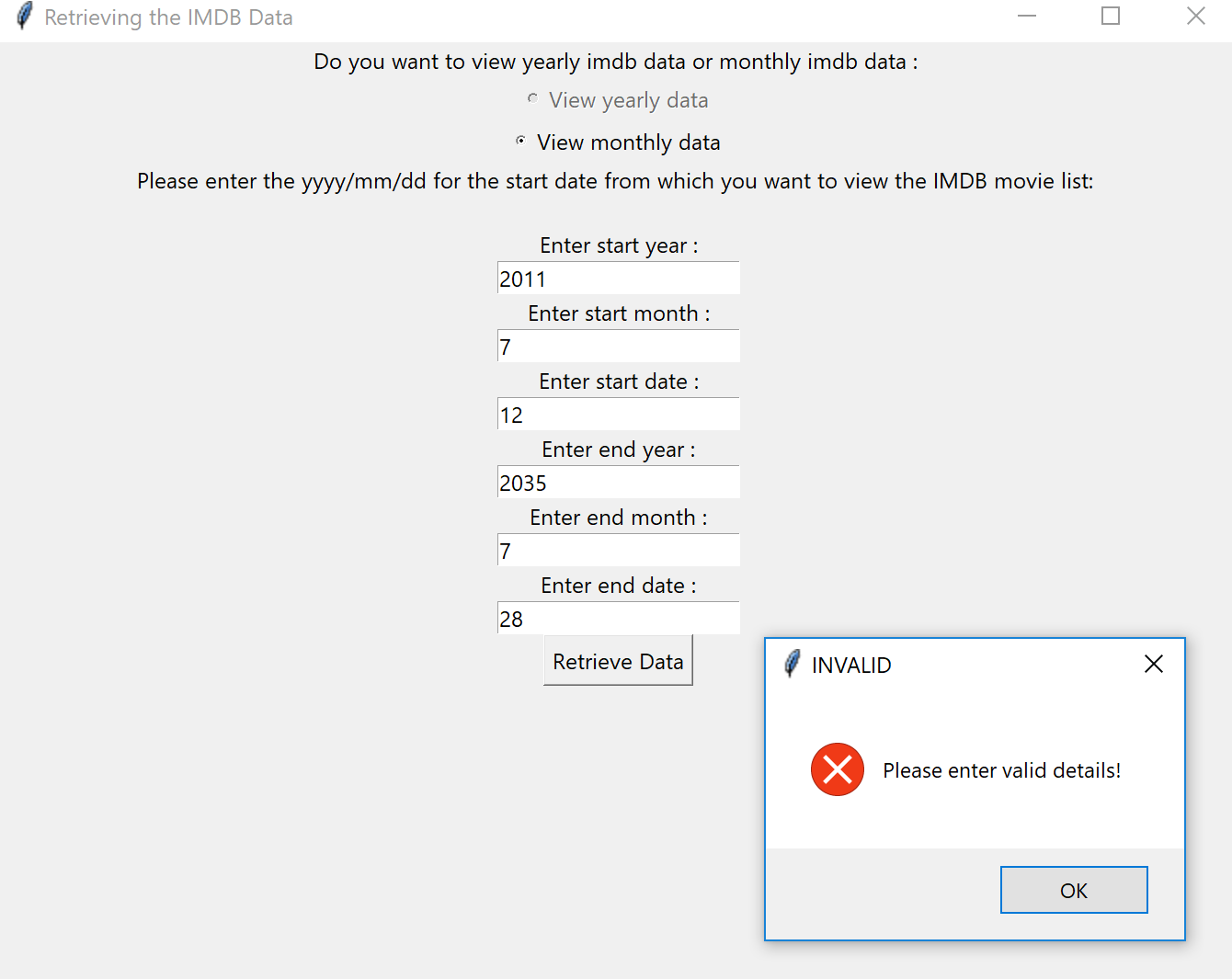
1. **ERROR VALIDATION IN ‘RETRIEVING THE IMDB DATA’ PAGE**
2. An error message is displayed when no valid range of dates are entered



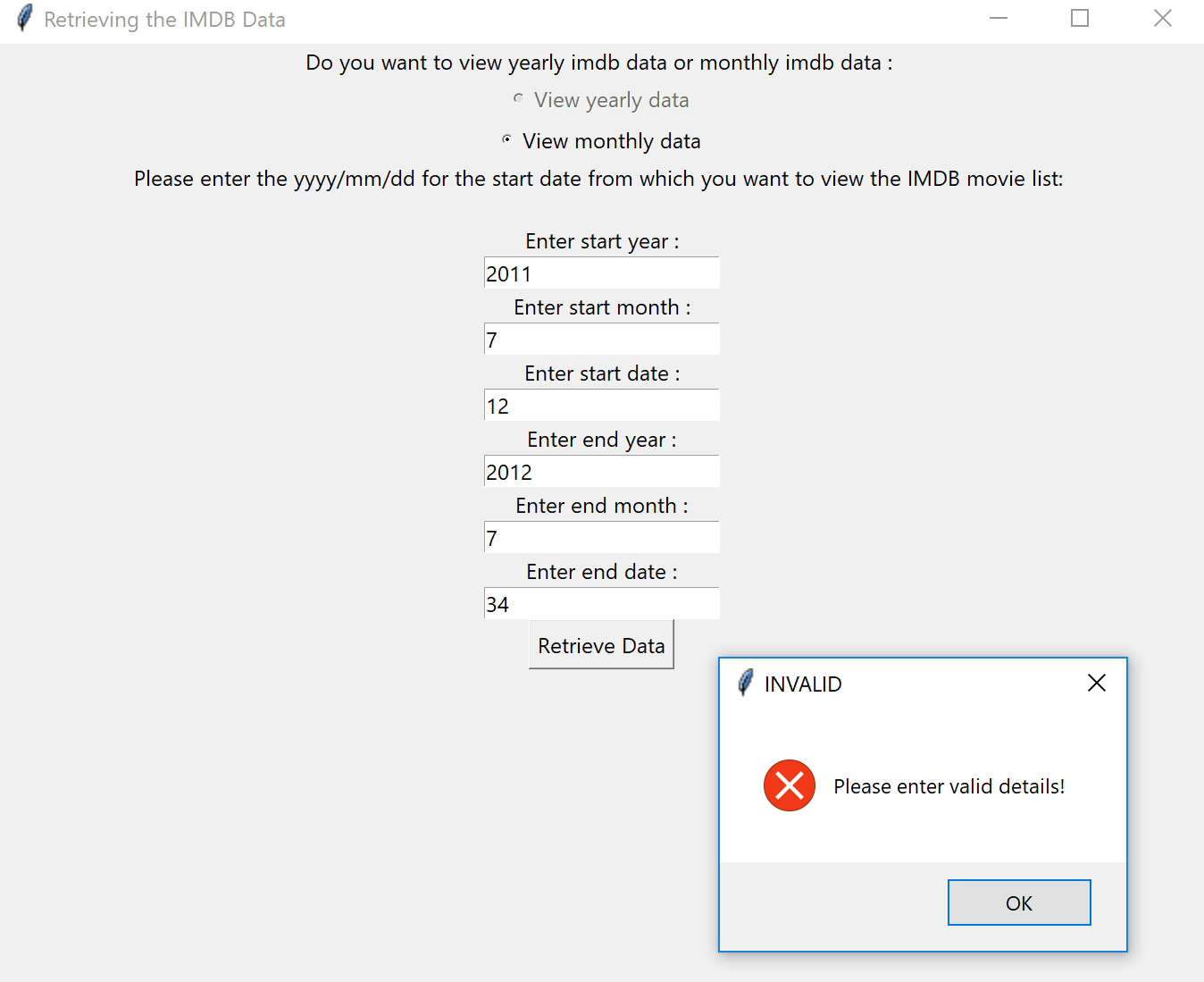
1. An error message is displayed when an invalid month is entered



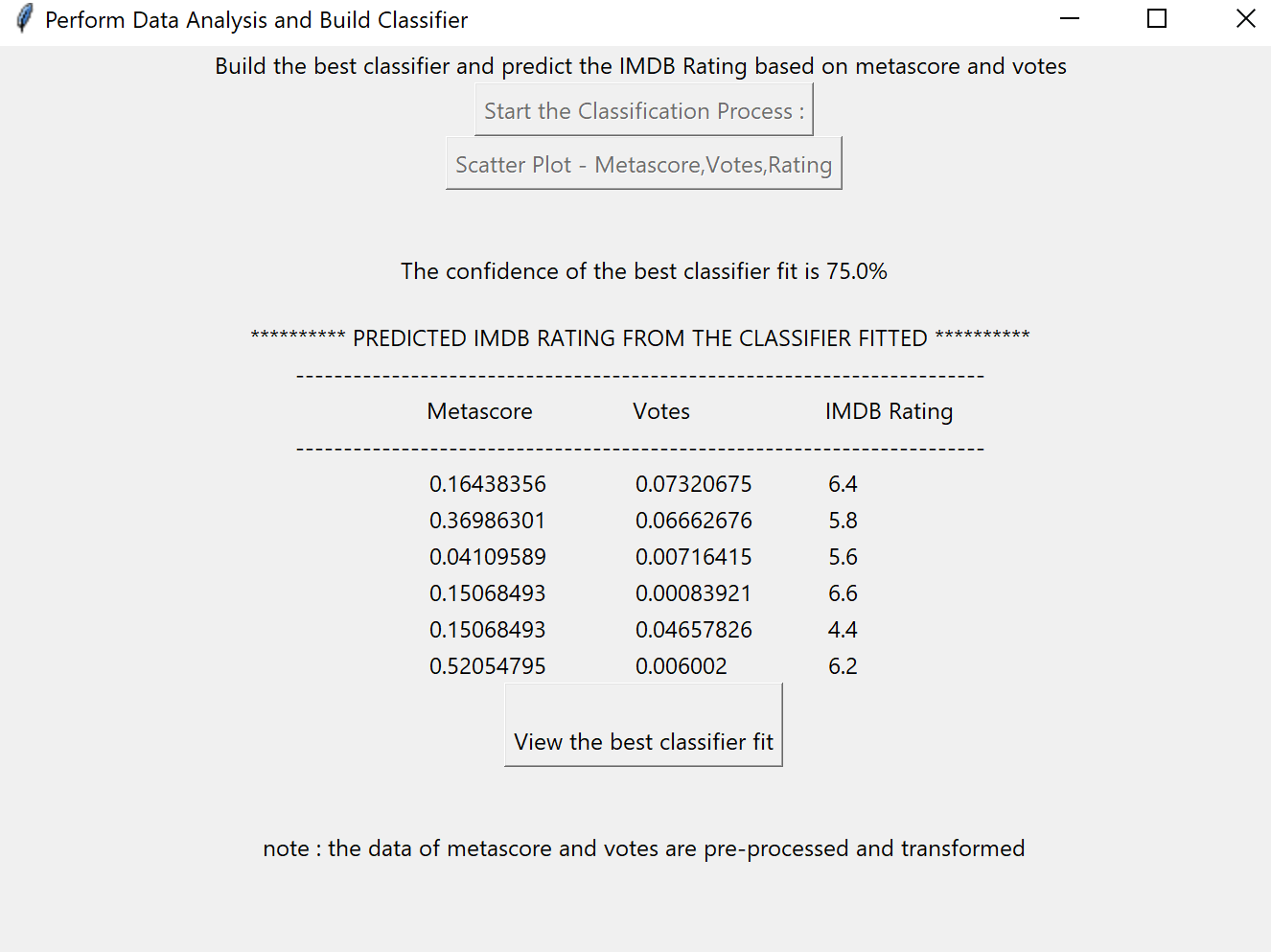
1. An error message is displayed when an invalid year is entered



1. An error message is displayed when an invalid date is entered

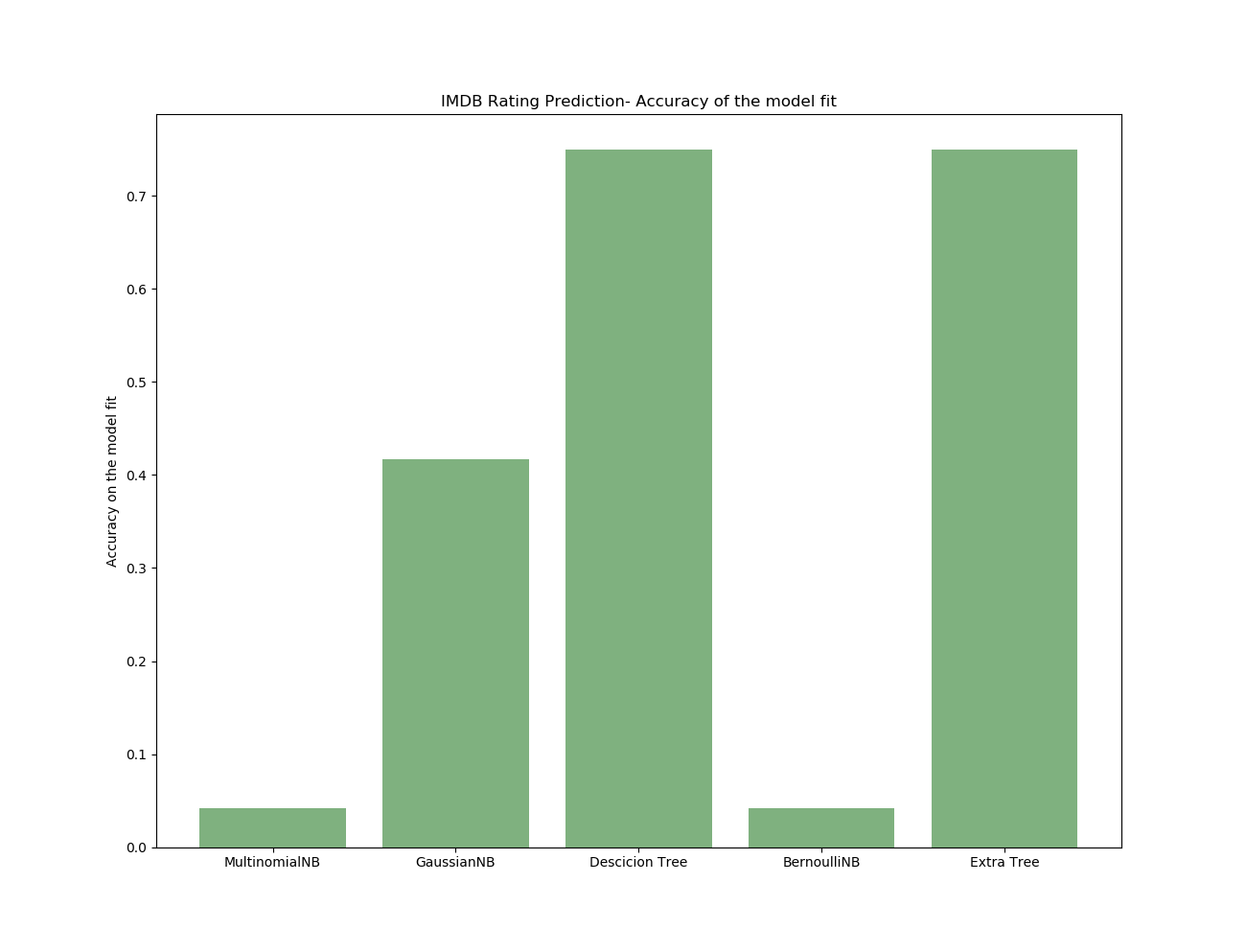


1. **PERFORM DATA ANALYSIS AND BUILD CLASSIFIER**
2. We can find the best classifier fit has an accuracy of 75% and we can also see the predicted IMDB ratings for the unknown data. Here the meta-score and votes have been transformed using max min scalar transformation since they had different range of values



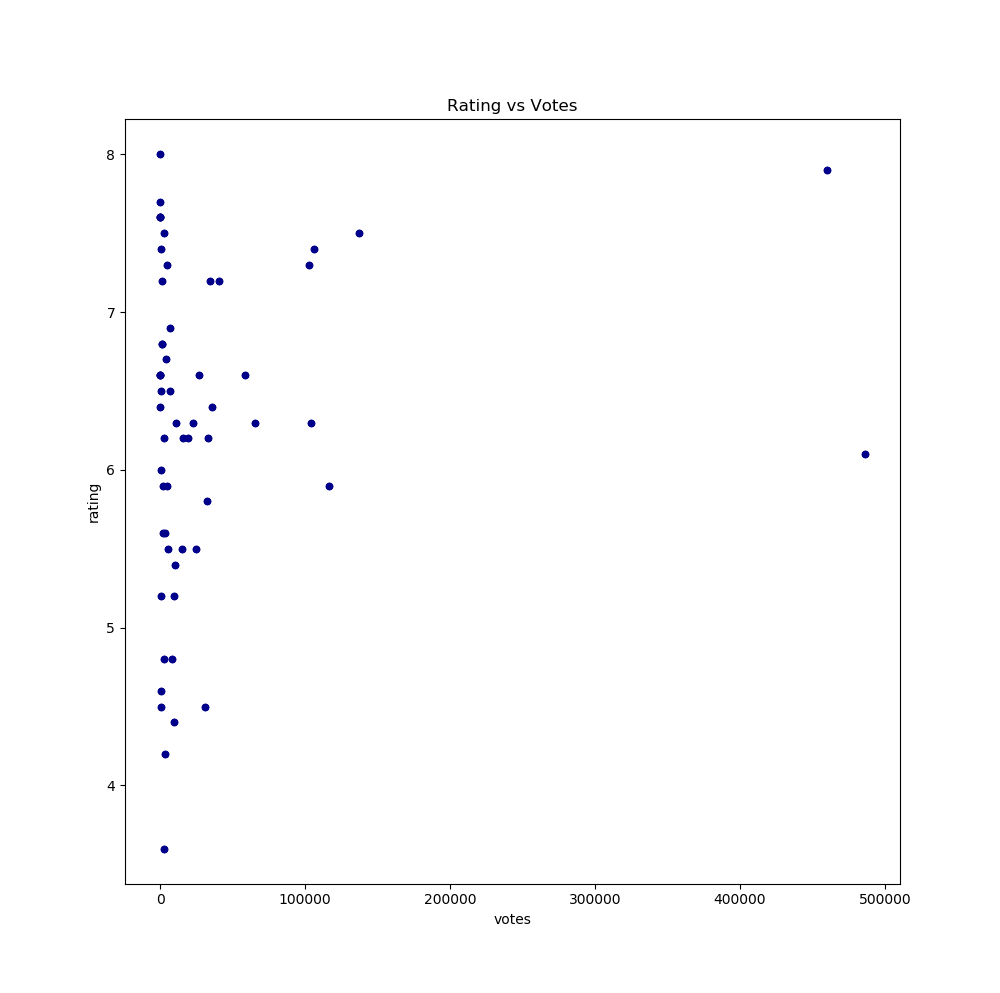
1. **ACCURACY CHART**
2. Classifiers built are Multinomial Naïve bayes, Gaussian Naïve bayes,KNN,Decision Tree,Bernoulli Naïve bayes and Extra Tree classifier

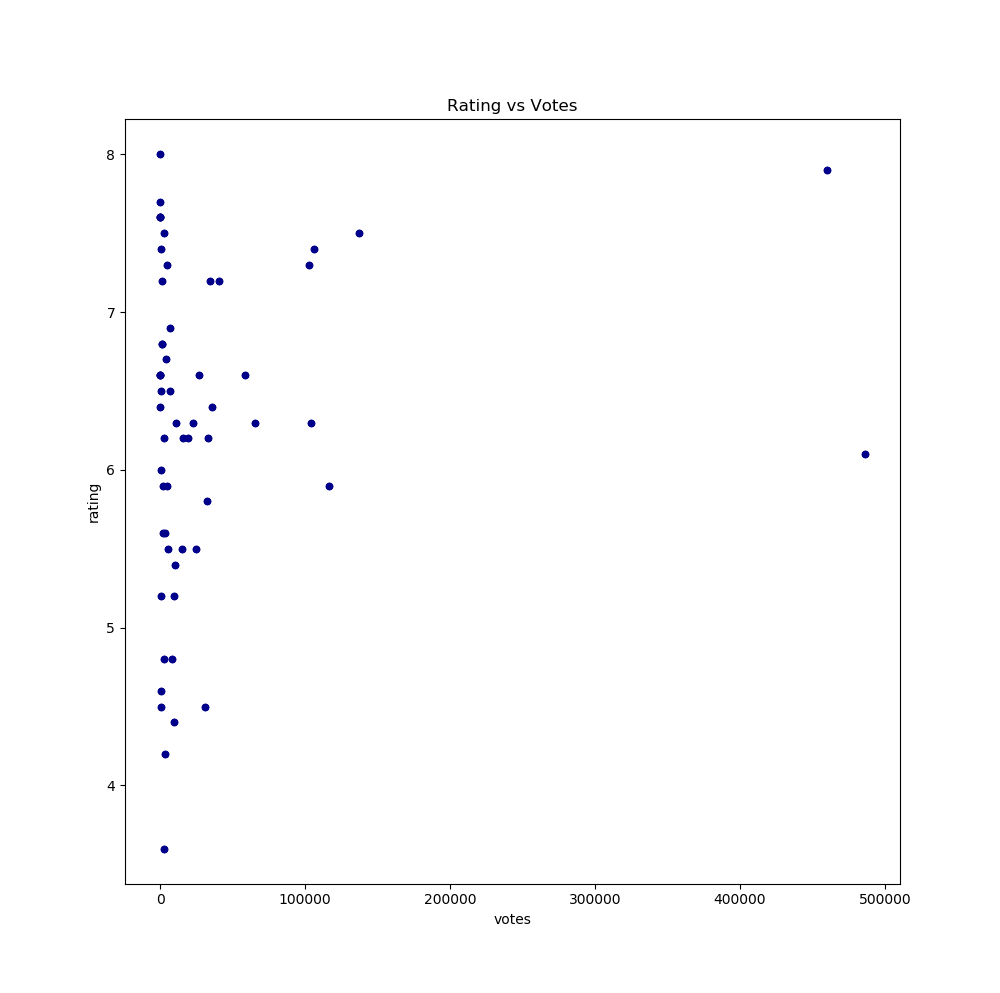
**Here we can see the Extra Tree Classifier has the best accuracy and so the best classifier built is Extra Tree Classifier**



1. **SCATTER PLOT**
2. The scatter plot between rating and votes

**Since the data is retrieved for only 24 months, the data points are very less but we can still find that Rating and votes have a positive linear relationship**

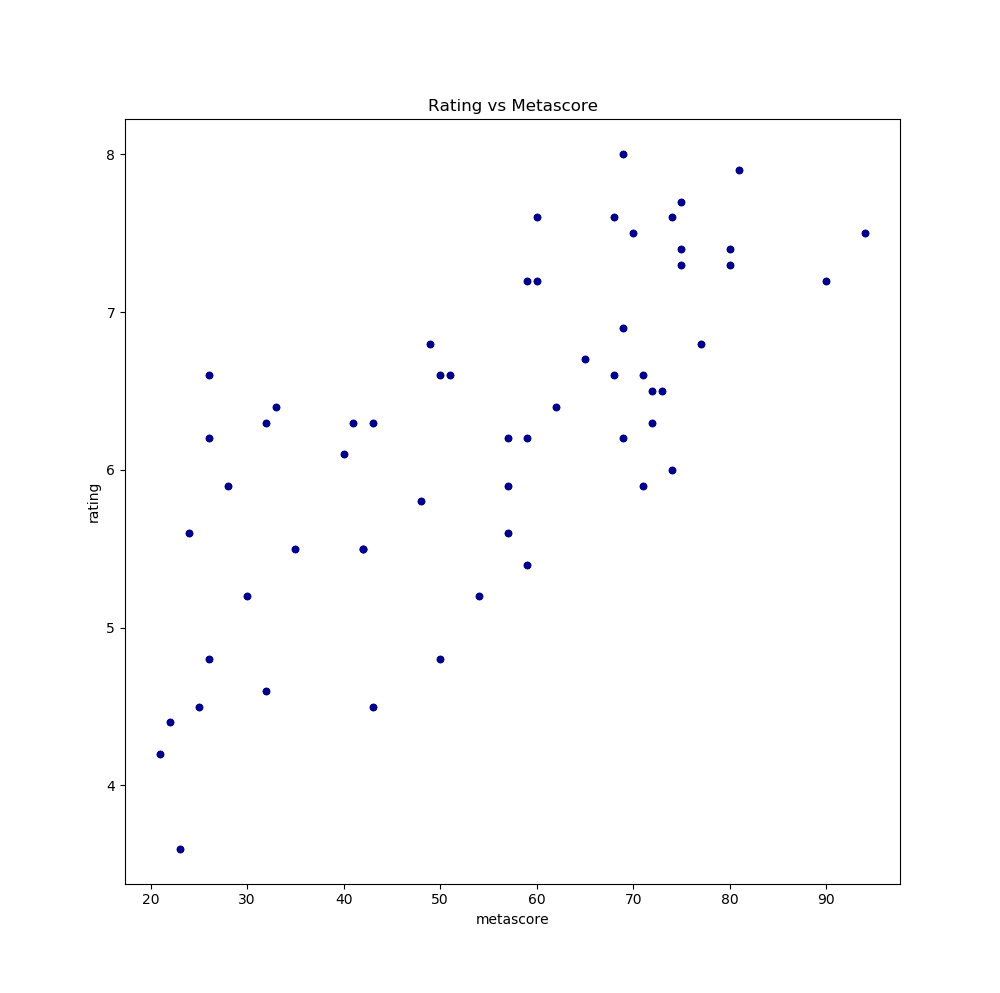


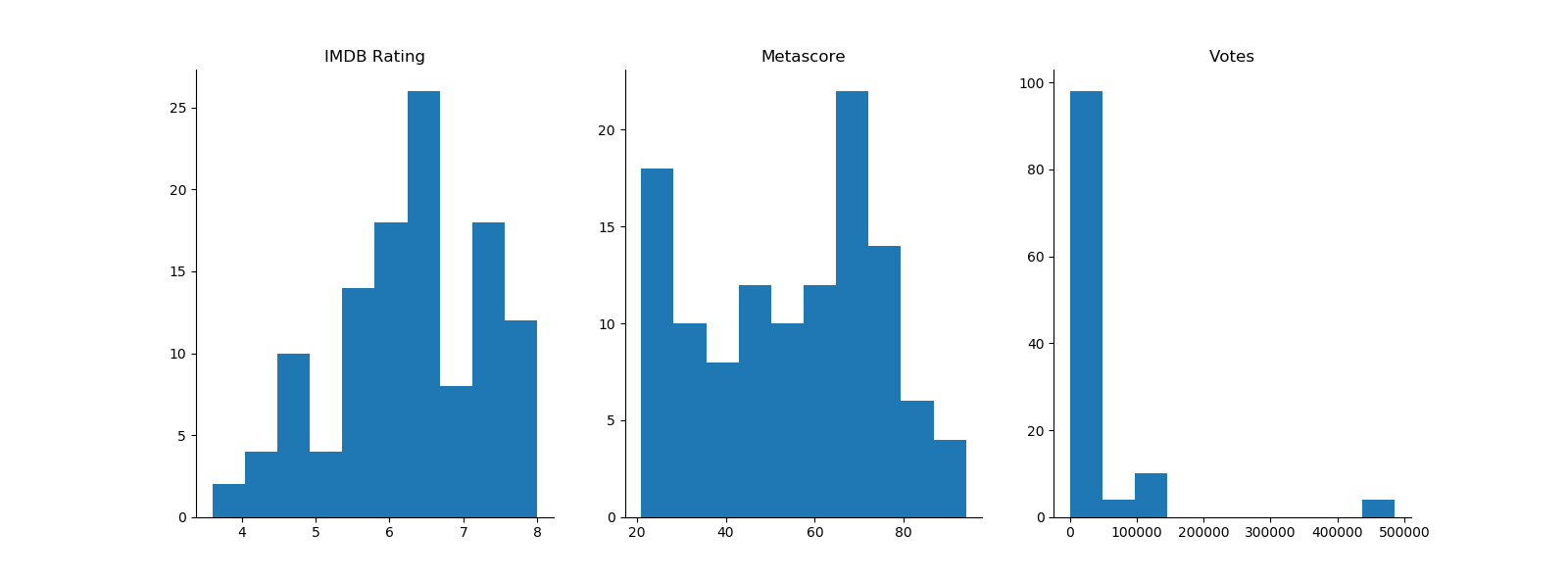


1. The scatter plot between rating and meta-score

**We can observe that the Rating increases as the meta-score increase indicating that**

**they have a positive linear relationship.**



1. The histogram of rating, meta-score and votes 

# Summary Stats

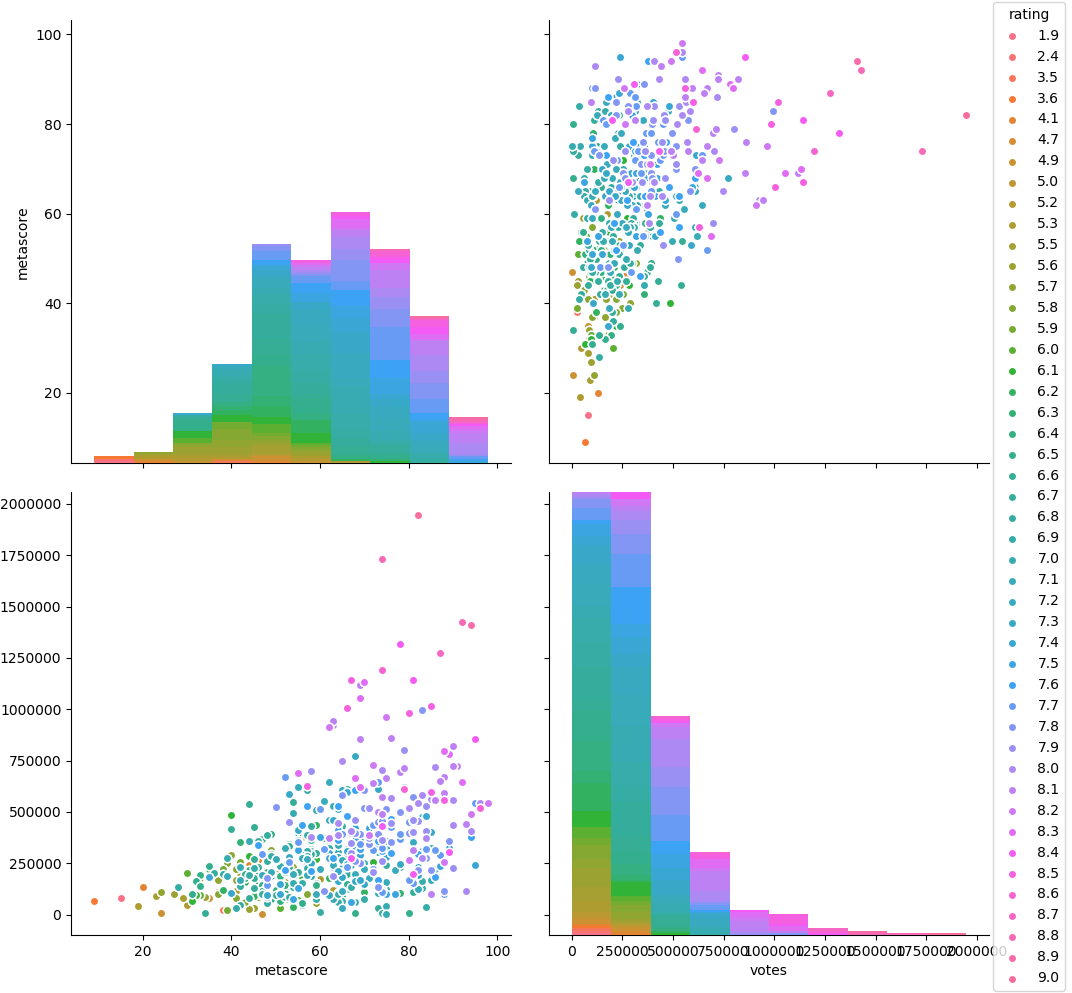
# The summary stats for the IMDB list that was retrieved for year between 2000 and 2018

# The summary stats for the IMDB list that was retrieved for dates between 2016/1/1 and 2018/1/1

# 

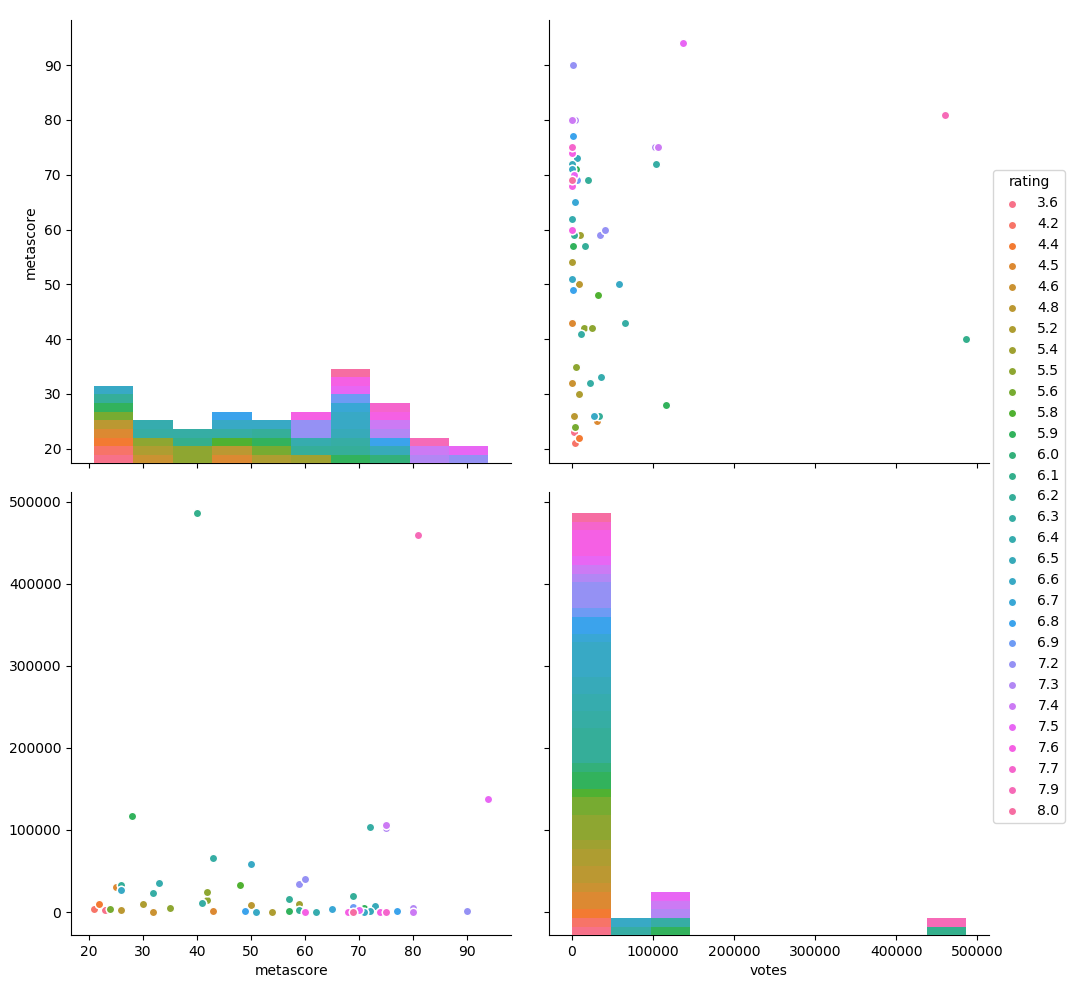
# Extra Credits

* + - 1. **MULTIPLOT GRAPHS – Histogram of meta-score and votes and scatter plot of metascore vs rating and Votes vs Rating – (2000 to 2018 IMDB List)**

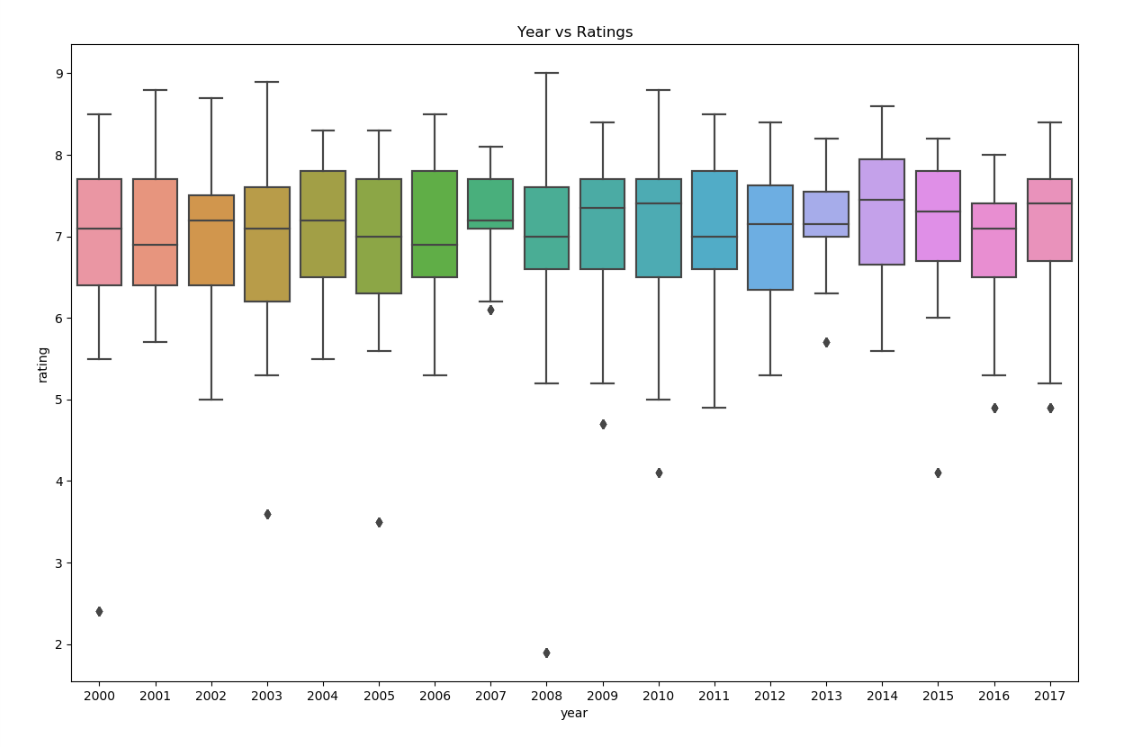


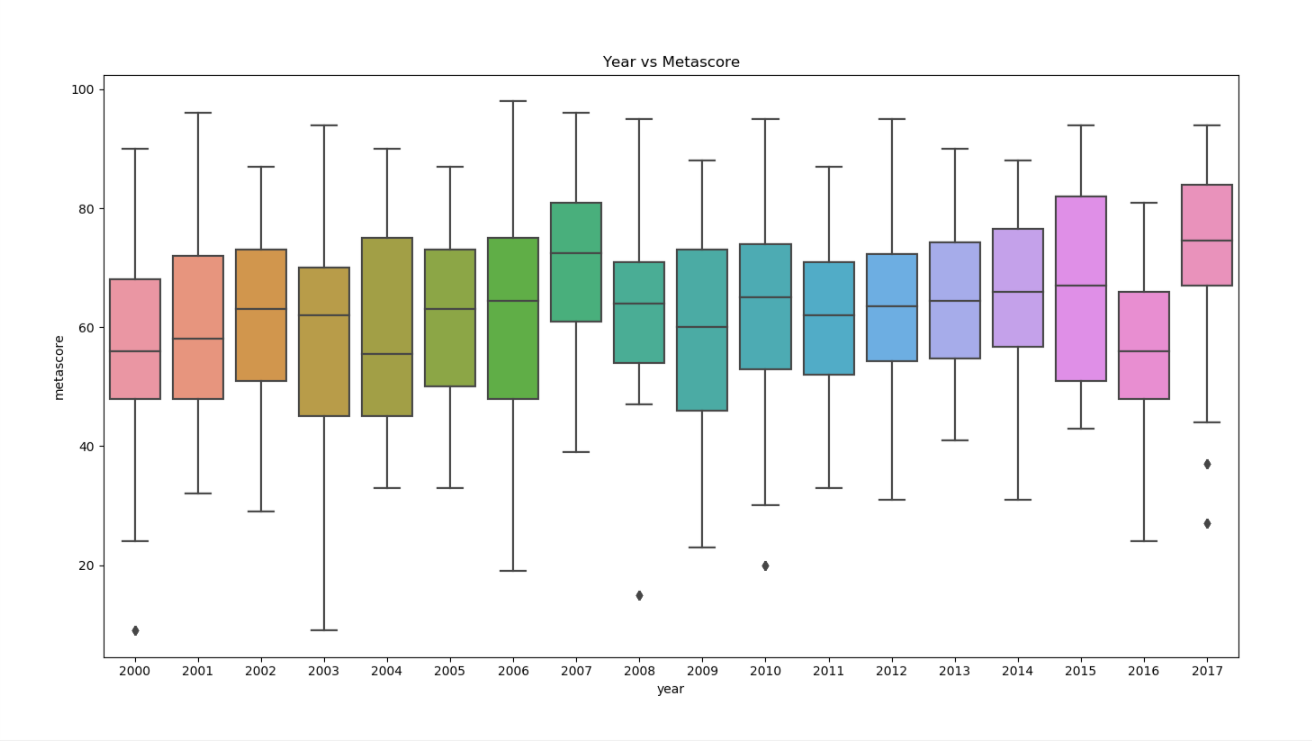
# 

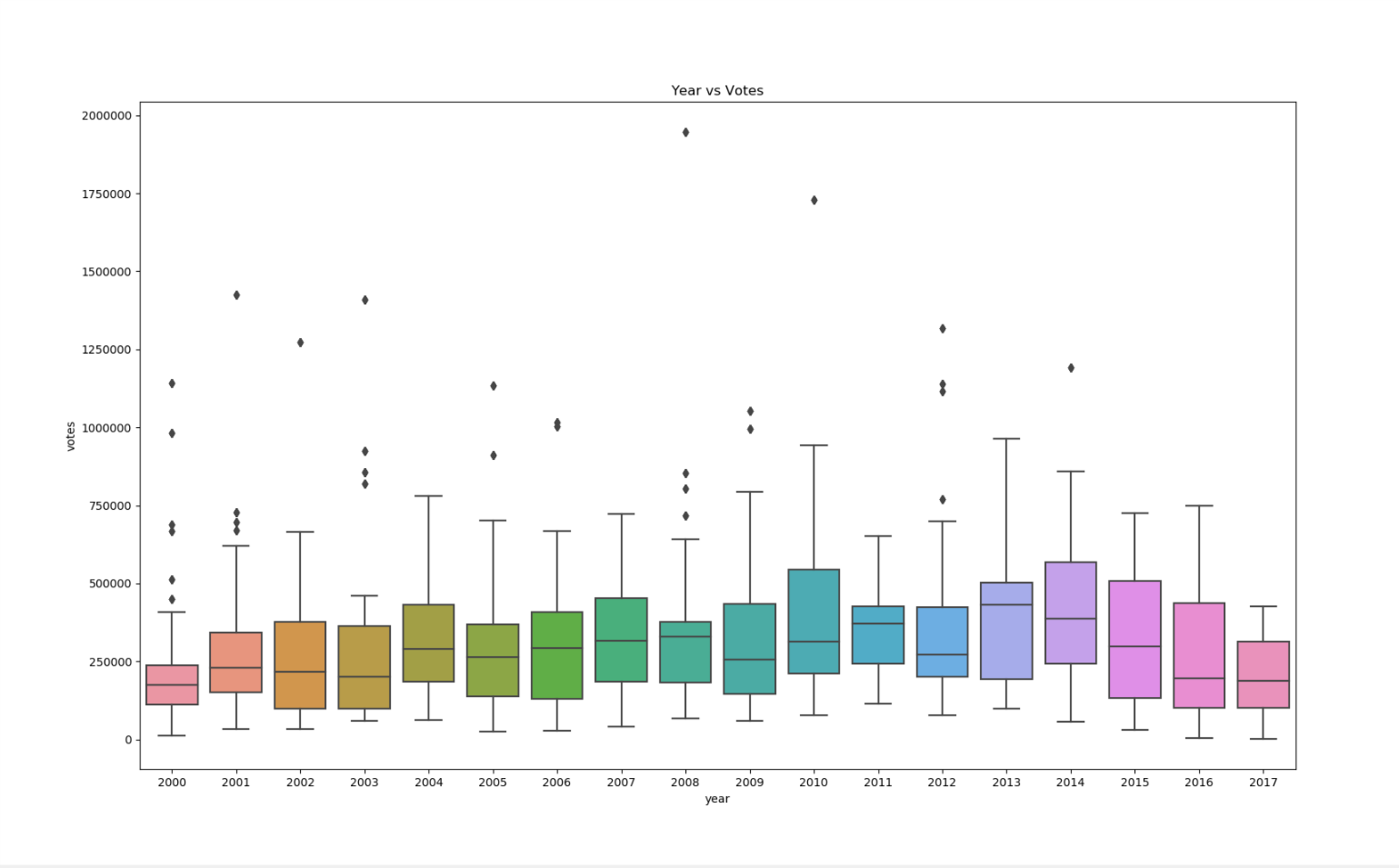
* + - 1. **MULTIPLOT GRAPHS – Histogram of meta-score and votes and scatter plot of metascore vs rating and Votes vs Rating – (2016/1/1 to 2018/1/1 IMDB List)**



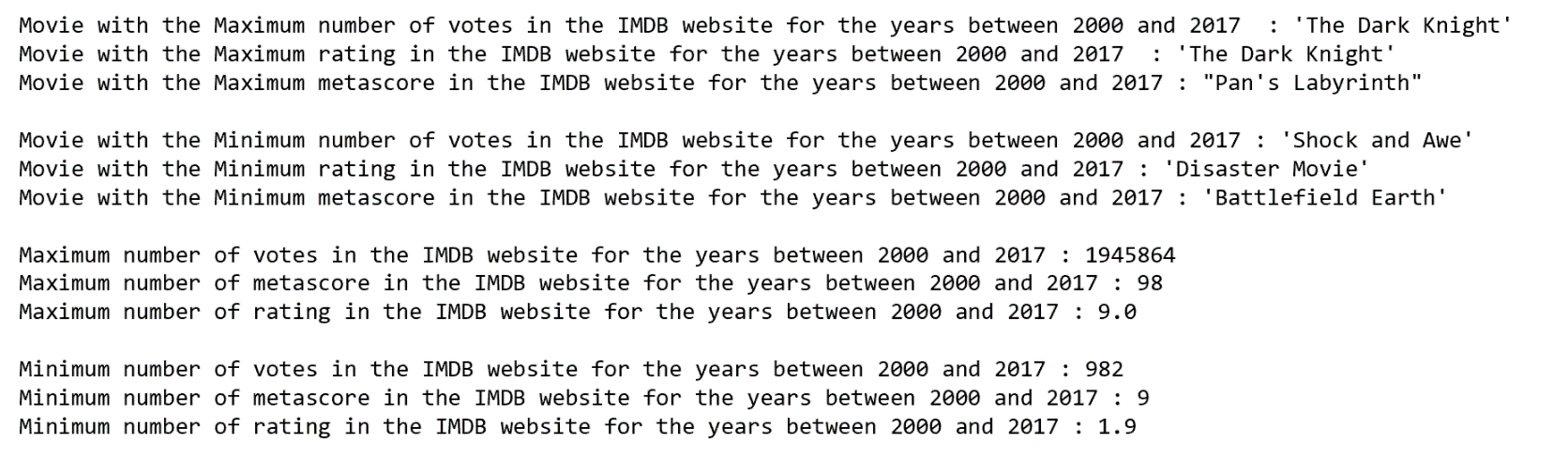
* + - 1. **BOX PLOT – YEAR VS RATINGS**



* + - 1. **BOX PLOT – YEAR VS METASCORE**
      2. **BOX PLOT – YEAR VS VOTES**



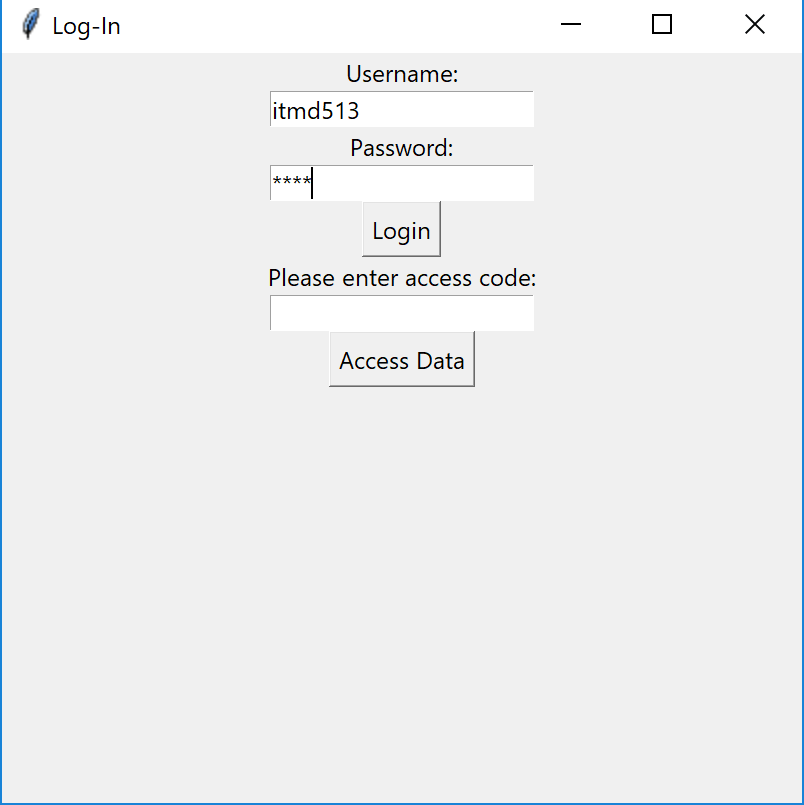
* + - 1. **Q6. QUERIES RUN BY USING SQLITE 3 FOR THE 200 to 2018 DATASET**



* + - 1. **USED JUPYTER NOTEBOOK FOR SCRPITING**

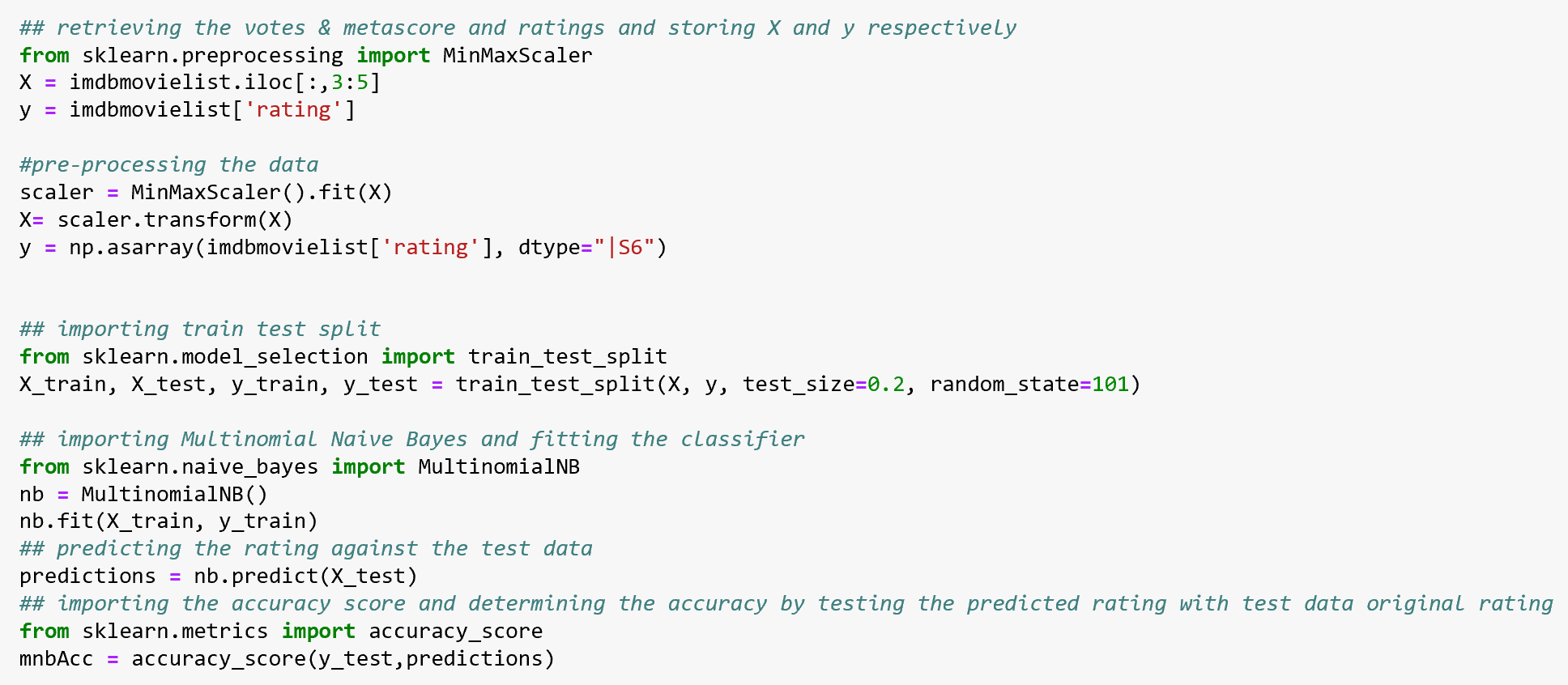
[**http://localhost:8892/notebooks/ITMD513\_A20395954\_OpenSourceProgramming\_Python\_FinalProject.ipynb**](http://localhost:8892/notebooks/ITMD513_A20395954_OpenSourceProgramming_Python_FinalProject.ipynb)

* + - 1. **PASSWORD HASHING**



* + - 1. ***MACHINE LEARNING (TRAIN 80% DATA AND TRAIN 20% DATA)***

*Followed all the machine learning concepts thoroughly like data preprocessing, train and test split, classification, using accuracy metric for deciding the best classifier*



# Future Scope & Limitations

# In future we can apply more classification and machine learning algorithms and compare them to predict the values even more accurate.

# Since only votes and meta-score are considered as the X variables, in future we can take gross collection of the movies to predict the IMDB ratings.

# The gross collection of the movie can be related to the release date of the movie like vacation period and festival time.

# We could see if the cast of the movie affects the gross collection or the ratings of the movies.

# Conclusions & Observations

# *By performing machine learning algorithms on the IMDB dataset, we could see that,*

# The votes and meta-score have a positive linear relationship with the IMDB rating and hence the predictions of the IMDB rating are closer to accurate.

# We have built many classifiers like Multinomial Naïve Bayes, Gaussian Naïve Bayes, Bernoulli Naïve Bayes, Decision Tree Classifier and Extra Tree Classifier and we figured out that Extra Tree Classifier is the best classifier among all the classifiers.

# With this classifier we would be able to predict an IMDB rating of a movie based on the votes and meta-scores of the movie