April 13, 2023

# C964: Computer Science Capstone Template

## Task 2 parts A, B, C, and D

# Part A: Project Proposal for Business Executives

## Letter of Transmittal

Brad Johnson

CEO

ABC Streaming

123 Main St, Las Vegas NV

Dear Mr. Johnson,

I am writing to present a proposal for the implementation of a program that I have designed. As a data scientist at ABC Streaming, I have experience developing computer systems that are able to analyze data and make accurate predictions based on that data. In order to provide more accurate and personalized recommendations to our users, I propose the implementation of a new system that will leverage machine learning algorithms to analyze user data and provide relevant recommendations.

I am confident that this data product will benefit our customers by improving the accuracy of our recommendations and supporting their decision-making process when selecting anime to watch. Additionally, this system will have a positive impact on our stakeholders by increasing user engagement and loyalty, resulting in increased revenue for the company. I estimate that this project will cost approximately $40,000 to fully develop and implement.

The following document is a detailed proposal outlining the proposed recommendation system, the data used to create it, the objectives of the project, the project methodology, funding requirements, and ethical and legal considerations that we will follow.  I believe that this proposed system will provide significant value to our company.

Sincerely,

Nathan Perkins

Data Scientist

# Project Recommendation

## Problem Summary

- The goal of this project is to develop and implement a system that will accurately recommend anime shows to users of our platform based on their viewing preferences.
- As a provider of streaming media, it should be our goal to be able to provide personalized entertainment for our users. In recent years anime shows have shown a huge increase in popularity among young viewers in the United States. In order for our service to stay ahead of the competition we must be able to provide accurate viewing recommendations for anime shows to our users.
- The current system used to recommend shows on our platform is outdated and only uses basic filtering methods. This has led to a decrease in user engagement.
- This project will deliver a program that will accurately predict a user's preferences based on their past viewing decisions. This program can later be implemented into the streaming service's user interface.

## Application Benefits

- This project will meet the company's needs by providing an updated recommendation system for anime shows. The new system will use historical user data in order to make recommendations based on what other users have watched and enjoyed.
- This improved recommendation system will improve user engagement by providing more accurate recommendations. By keeping users engaged and satisfied with our platform we will see an increase in subscriber retainment and as a result, an increase in revenue.
- This system can also use user viewing data to make better-informed decisions with content acquisition and marketing. By understanding our viewers' preferences we will be able to predict what other media they will enjoy.

## Application Description

- This project will use historical data on users' viewing preferences and the ratings that they have given for each show. Our program will analyze this data and based on a given show, will be able to identify other shows that users also watched and enjoyed. This method of recommendation is called collaborative filtering.
- Our users are complex and have a wide variety of tastes and preferences that cannot be simplified into a single genre. Using this method of recommendations goes beyond what our current system is capable of by basing recommendations on actual user activity instead of just the content of the media.

## Data Description

- The data used in this project will be pulled from myanimelist.com
- MyAnimeList is a popular anime discussion and rating website. Users there are able to list the shows that they have watched and give a rating from 0-10.
- The size of the data set will be approximately 2 million lines of data.
- The data type that will be used is quantitative. Including the user scores for every show in the database.
- Users who have given very few ratings or anime shows that have very few user ratings will be removed from the data set as outliers.

## Objectives and Hypothesis

- The desired outcome for this project is to develop a recommendation system for anime shows that will use collaborative filtering.

## Methodology

- The methodology that will be used for this project is Agile.
- Agile is the most appropriate for this project because it is flexible and allows for changes and adjustments in the project to be made quickly. This is important because after the system is completed we can use new user data to adjust the system based on that new data.
- Planning Phase: During this phase, we will define the scope of the project, identify stakeholders and their requirements, and create a project plan. We will also identify any risks and constraints associated with the project.
- Design Phase: In this phase, we will design the user interface and data processing pipeline. We will also create a test plan and define the success criteria for each test.
- Development Phase: During this phase, we will implement the user interface and data processing pipeline. We will also perform unit testing and integration testing to ensure that the components of the data product work together as intended.
- Testing Phase: In this phase, we will test the data product to ensure that it meets the success criteria defined in the design phase. We will also perform user acceptance testing to gather feedback from users and identify areas for improvement.
- Deployment Phase: During this phase, we will deploy the data product to the production environment. We will also provide training and support to users as needed.
- Maintenance Phase: In this phase, we will monitor the data product and perform ongoing maintenance to ensure that it continues to meet the needs of users.

## Funding Requirements

- The funding required for this project is $40,000.
- This includes the cost of workstations and salaries for two software developers for two months.
- No server costs are included since this program will be implemented into our current system and hardware.

## Data Precautions

- No sensitive information will be used in this project.
- All information that will be used is publicly accessible and anonymous.

## Developer's Expertise

- My qualifications for this project are, I will have a Bachelor's degree in computer science from WGU in the near future. I have completed coursework through my studies using a variety of programming languages and am familiar with software engineering practices.

# Part B: Project Proposal

## Problem Statement

- Recently there has been a decrease in user engagement on our streaming platform, specifically with the recommendations being given to the user.

## Customer Summary

- Our customers are people who enjoy watching television and movies on internet streaming platforms. This project aims to target a younger demographic who may be more interested in anime shows.
- The proposed system will solve this problem by being able to serve more accurate recommendations to our users and increase user engagement and retention with our platform. By using our data product, our customers will have access to a personalized recommendation system that will help them discover new shows that align with their preferences.

## Existing System Analysis

- The current system being used is built on simple content-based filtering. This type of recommendation only takes into account the details of a particular show such as genre, length, or time period of release.
- The limitation of this system is that users will only be recommended shows based on the content of their viewing history. A better system would be able to make recommendations based on what similar users have viewed and liked.
- Based on the user data that we are currently seeing, the recommendations that are served to our users are not accurate and do not drive user engagement.

## Data

- The data sets being used in this project will be collected from myanimelist.com using their API.
- The raw data set will be in two parts. First, a data set including information for each show in the database. The important data there is the show's English name, the myanimelist id for the show, overall user score, genre, show ranking, and popularity. Second, a data set of user rating information. This includes the name of the show being ranked, the myanimelist id for the show, the user id, and the rating that the user gave.
- The data will be cleaned and prepared for the machine learning model through a separate program where irrelevant data will be removed and the two data sets will be merged into one CSV
- Outliers and data anomalies will be removed during the data processing step. Outliers will be defined as users who have not given enough ratings and also shows that do not meet a certain threshold of popularity.

## Project Methodology

- The methodology that will be used for this project will be the Agile methodology. Agile works well for projects that may need to be changed or improved as needs change and more information becomes available. With a machine learning project, it is important to test and modify the machine learning model in order to achieve the most accurate model. Agile will work well with this type of work since we will be using historical data as well as new user data after the initial release of the program.
- In the analysis phase, we will gather requirements from stakeholders and decide what is in scope for this project. We will also identify data sources and perform data analysis to determine data quality and identify any limitations or anomalies.
- In the design phase, we will create a detailed design of the application, including the type of machine learning model that we will use to process our data. We will decide on what metrics will be used to measure the accuracy of the model.
- In the development phase, we will implement the application according to the design specifications. We will create unit tests and perform integration testing to ensure the application functions as intended.
- In the testing phase, we will test the accuracy of the machine learning model. We will consider an accuracy score of at least 75% to be a success. Because we are only using historical data for the initial release of the program it will be difficult to measure the accuracy of the recommendations. For this reason, we will measure the accuracy of the model by measuring how well it can predict what shows a user has already seen and rated highly.
- Finally, in the deployment phase, the new program will be integrated into our current system.

## Project Outcomes

- The deliverables for this project are:
  - A working recommendation system that uses a machine learning model.
  - A program used for data cleaning.
  - A final report on the outcome of the project.

## Implementation Plan

- After the approval of the initial design of the program, the development of the project will start.
- First, the data will be collected and analyzed and we will determine how to properly prepare the data to fit the machine learning model. A separate script will be created to do the data cleaning and will output the cleaned data into a new CSV file for our main program to read from. This will help with the processing time of the main program.
- Next, code for the machine learning model will be written as well as methods that will be used in testing the accuracy of the model. Several machine learning models and accuracy metrics will be tested until we achieve the wanted results for the project. At the moment I think that we will most likely use a Nearest Neighbor algorithm for the machine learning model.
- The model will be thoroughly tested and modified until it meets our standards for accuracy and quality.

- Once the machine learning model is complete, we will write code for a prototype user interface that will be used for demonstrating the program to stakeholders before it is fully implemented into the main program for our service.

## Evaluation Plan

- During the development stage of our plan, we will implement code reviews so that our developers can compare and offer advice to each other in order to improve the quality of the code being written.
- Unit tests will be written to ensure that the program is functioning as intended.
- Integration testing will be conducted to ensure that the developed program integrates well with the existing system.
- The program will be tested on its accuracy in regard to how well it is able to predict what shows a user has already watched and rated highly. We will aim for an accuracy score of 75%.
- After the program has been deployed into our production system, we will measure whether there has been a change in user engagement and retention. We will consider the product a success if there is an increase in these metrics by 20% over the course of six months after release.

## Resources and Costs

- Two developer workstation computers at $1500 each
- Salary for two software developers for two months, estimated at $16,000 each.
- The programming environment and software libraries that will be used are open-source and free.
- The total estimated budget for this project is $40,000

## Timeline and Milestones

| **Milestone** | **Activity** | **Start Date** | **End Date** |
|---|---|---|---|
| 1 | Program designed and approved | 05/01/2023 | 05/06/2023 |
| 2 | Data Collection | 05/07/2023 | 05/10/2023 |
| 3 | Data cleaning | 05/11/2023 | 05/18/2023 |
| 4 | Model development | 05/18/2023 | 06/15/2023 |
| 5 | User interface developed | 06/15/2023 | 06/20/2023 |
| 6 | Final testing | 06/20/2023 | 06/25/2023 |
| 7 | Deployment | 06/25/2023 | 06/30/2023 |
| 8 | Final report written | 07/01/2023 | 07/07/2023 |

# Part C: Application

All application files can be found in this same directory. A guide on how to run the program can be found in part D of this report.

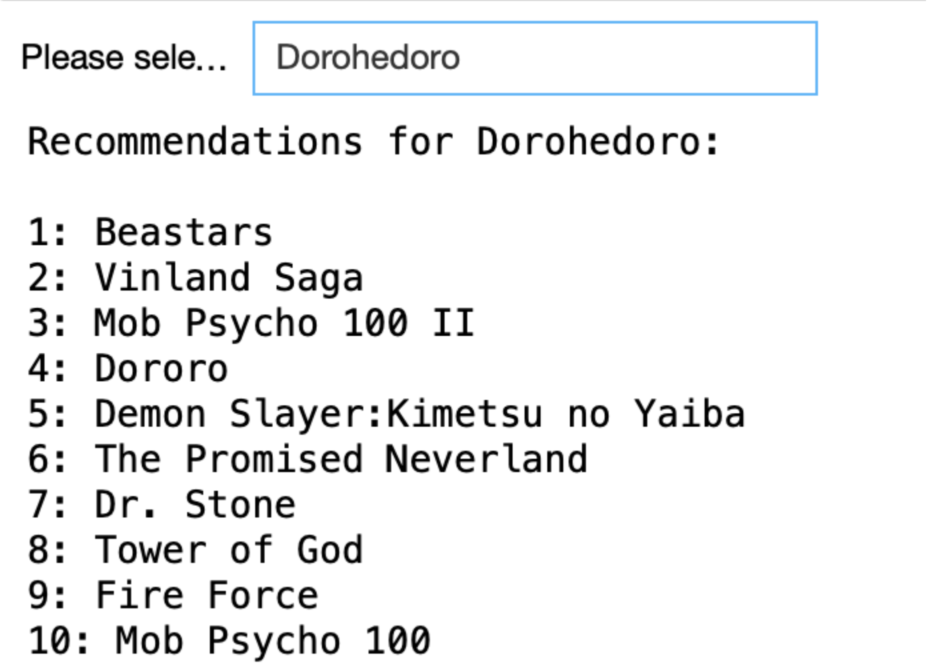
# Part D: Post-implementation Report

## A Business (or Organization) Vision

ABC Streaming has recently seen a decrease in user engagement and user retention on its platform. In order to help with this problem I have developed a system that is able to offer personalized recommendations to users.

The program that I have developed uses machine learning to generate recommendations based on shows that the user has previously watched.

A user is able to enter the name of an anime show and the program will generate 10 recommendations.

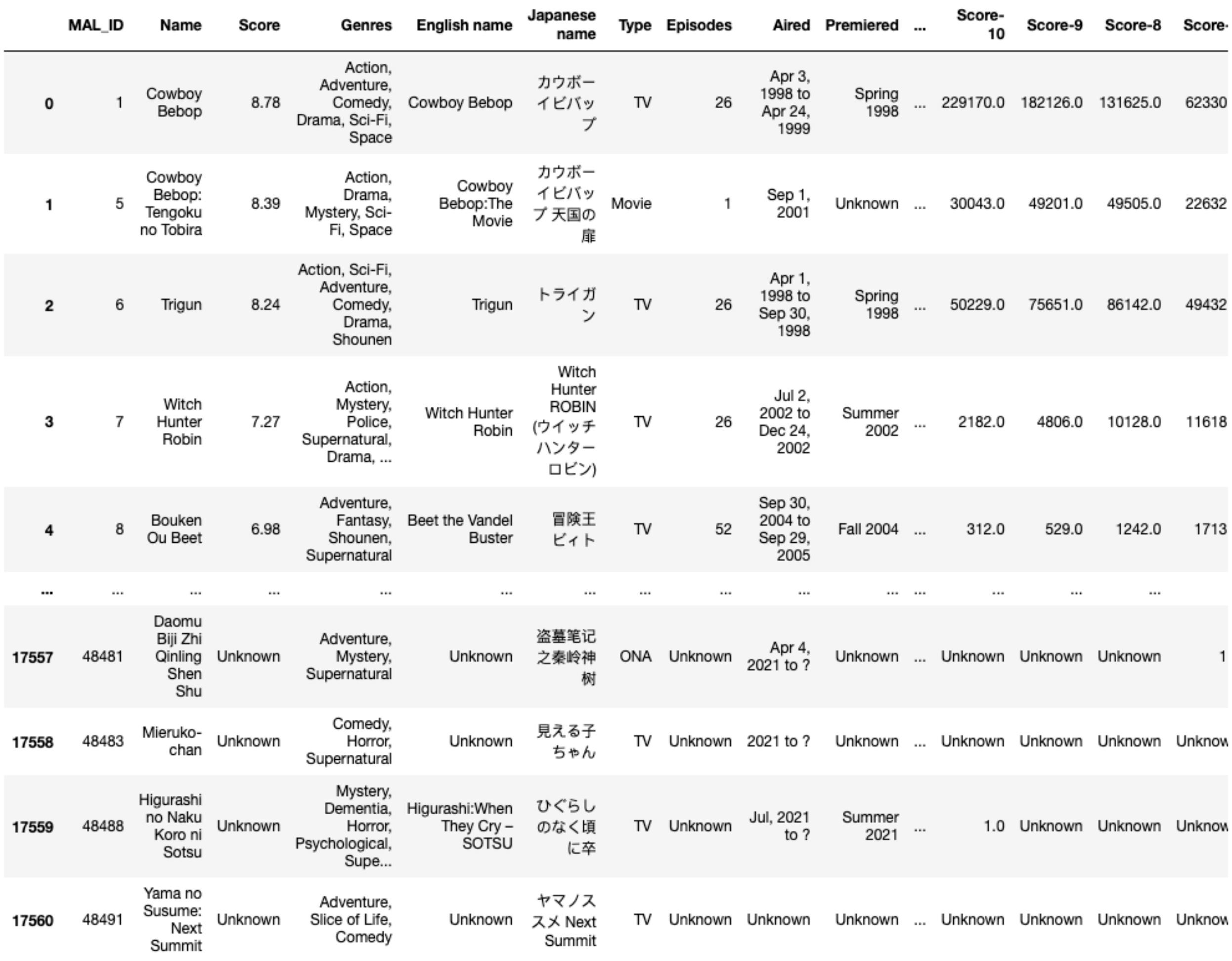


Please sele… Dorohedoro

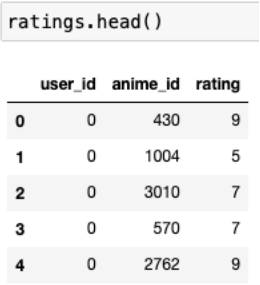
```
Recommendations for Dorohedoro:

1: Beastars
2: Vinland Saga
3: Mob Psycho 100 II
4: Dororo
5: Demon Slayer:Kimetsu no Yaiba
6: The Promised Neverland
7: Dr. Stone
8: Tower of God
9: Fire Force
10: Mob Psycho 100
```

## Datasets

- The raw data sets came in two parts: first, a description of every anime show in the myanimelist.com database, and a list of every user rating given on the myanimelist.com platform.
- Here is an example of the raw data:



| | MAL_ID | Name | Score | Genres | English name | Japanese name | Type | Episodes | Aired | Premiered | ... | Score-10 | Score-9 | Score-8 | Score- |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | Cowboy Bebop | 8.78 | Action, Adventure, Comedy, Drama, Sci-Fi, Space | Cowboy Bebop | カウボーイビバップ | TV | 26 | Apr 3, 1998 to Apr 24, 1999 | Spring 1998 | ... | 229170.0 | 182126.0 | 131625.0 | 62330 |
| **1** | 5 | Cowboy Bebop: Tengoku no Tobira | 8.39 | Action, Drama, Mystery, Sci-Fi, Space | Cowboy Bebop:The Movie | カウボーイビバップ 天国の扉 | Movie | 1 | Sep 1, 2001 | Unknown | ... | 30043.0 | 49201.0 | 49505.0 | 22632 |
| **2** | 6 | Trigun | 8.24 | Action, Sci-Fi, Adventure, Comedy, Drama, Shounen | Trigun | トライガン | TV | 26 | Apr 1, 1998 to Sep 30, 1998 | Spring 1998 | ... | 50229.0 | 75651.0 | 86142.0 | 49432 |
| **3** | 7 | Witch Hunter Robin | 7.27 | Action, Mystery, Police, Supernatural, Drama, ... | Witch Hunter Robin | Witch Hunter ROBIN (ウイッチハンターロビン) | TV | 26 | Jul 2, 2002 to Dec 24, 2002 | Summer 2002 | ... | 2182.0 | 4806.0 | 10128.0 | 11618 |
| **4** | 8 | Bouken Ou Beet | 6.98 | Adventure, Fantasy, Shounen, Supernatural | Beet the Vandel Buster | 冒険王ビィト | TV | 52 | Sep 30, 2004 to Sep 29, 2005 | Fall 2004 | ... | 312.0 | 529.0 | 1242.0 | 1713 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | |
| **17557** | 48481 | Daomu Biji Zhi Qinling Shen Shu | Unknown | Adventure, Mystery, Supernatural | Unknown | 盗墓笔记之秦岭神树 | ONA | Unknown | Apr 4, 2021 to ? | Unknown | ... | Unknown | Unknown | Unknown | 1 |
| **17558** | 48483 | Mieruko-chan | Unknown | Comedy, Horror, Supernatural | Unknown | 見える子ちゃん | TV | Unknown | 2021 to ? | Unknown | ... | Unknown | Unknown | Unknown | Unknow |
| **17559** | 48488 | Higurashi no Naku Koro ni Sotsu | Unknown | Mystery, Dementia, Horror, Psychological, Supe... | Higurashi:When They Cry – SOTSU | ひぐらしのなく頃に卒 | TV | Unknown | Jul, 2021 to ? | Summer 2021 | ... | 1.0 | Unknown | Unknown | Unknow |
| **17560** | 48491 | Yama no Susume: Next Summit | Unknown | Adventure, Slice of Life, Comedy | Unknown | ヤマノススメ Next Summit | TV | Unknown | Unknown | Unknown | ... | Unknown | Unknown | Unknown | Unknow |

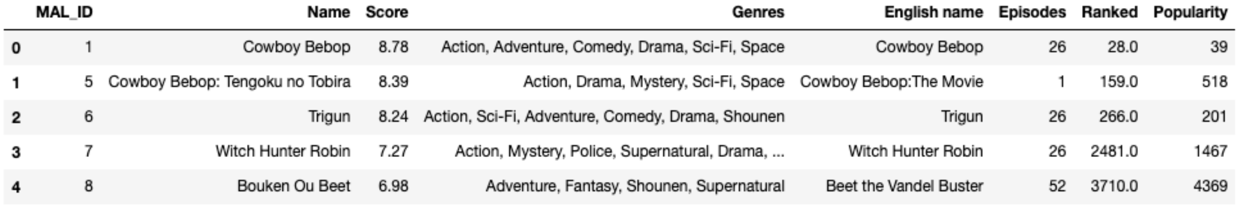


```
ratings.head()
```

| | user_id | anime_id | rating |
|---|---|---|---|
| **0** | 0 | 430 | 9 |
| **1** | 0 | 1004 | 5 |
| **2** | 0 | 3010 | 7 |
| **3** | 0 | 570 | 7 |
| **4** | 0 | 2762 | 9 |

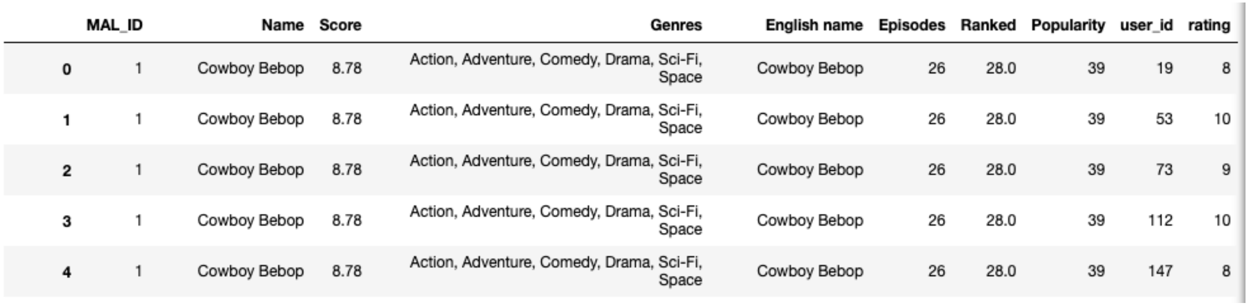
- The data set in a raw form contained a lot of useless information and outliers so a lot of data cleaning was necessary. First, the two data sets were read into the data cleaning program as pandas dataframes. Then the unnecessary columns in the data set were removed. For the data set with the anime show data the columns that were left were: MAL_ID, Name, Average user Score, Genres, English Name, Episodes, Numerical ranking based on score, and Popularity ranking based on the number of user ratings. This data set was then merged with the user rating data set.
- At this point, the data set had over 2 billion entries and I realized that I had to find a way to trim the data down to a more manageable size. So before I merged the data I trimmed the bottom 40% of anime shows based on popularity. Then I also removed all user scores from users who had not given at least 400 reviews.

- Here is an example of the cleaned data sets:
  Anime data:



| | MAL_ID | Name | Score | Genres | English name | Episodes | Ranked | Popularity |
|---|---|---|---|---|---|---|---|---|
| 0 | 1 | Cowboy Bebop | 8.78 | Action, Adventure, Comedy, Drama, Sci-Fi, Space | Cowboy Bebop | 26 | 28.0 | 39 |
| 1 | 5 | Cowboy Bebop: Tengoku no Tobira | 8.39 | Action, Drama, Mystery, Sci-Fi, Space | Cowboy Bebop:The Movie | 1 | 159.0 | 518 |
| 2 | 6 | Trigun | 8.24 | Action, Sci-Fi, Adventure, Comedy, Drama, Shounen | Trigun | 26 | 266.0 | 201 |
| 3 | 7 | Witch Hunter Robin | 7.27 | Action, Mystery, Police, Supernatural, Drama, ... | Witch Hunter Robin | 26 | 2481.0 | 1467 |
| 4 | 8 | Bouken Ou Beet | 6.98 | Adventure, Fantasy, Shounen, Supernatural | Beet the Vandel Buster | 52 | 3710.0 | 4369 |

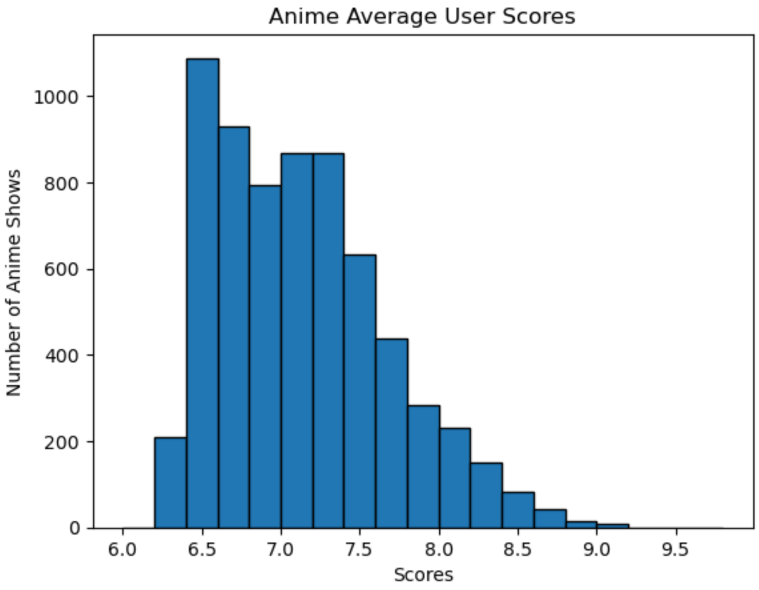
Merged data:



| | MAL_ID | Name | Score | Genres | English name | Episodes | Ranked | Popularity | user_id | rating |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | Cowboy Bebop | 8.78 | Action, Adventure, Comedy, Drama, Sci-Fi, Space | Cowboy Bebop | 26 | 28.0 | 39 | 19 | 8 |
| 1 | 1 | Cowboy Bebop | 8.78 | Action, Adventure, Comedy, Drama, Sci-Fi, Space | Cowboy Bebop | 26 | 28.0 | 39 | 53 | 10 |
| 2 | 1 | Cowboy Bebop | 8.78 | Action, Adventure, Comedy, Drama, Sci-Fi, Space | Cowboy Bebop | 26 | 28.0 | 39 | 73 | 9 |
| 3 | 1 | Cowboy Bebop | 8.78 | Action, Adventure, Comedy, Drama, Sci-Fi, Space | Cowboy Bebop | 26 | 28.0 | 39 | 112 | 10 |
| 4 | 1 | Cowboy Bebop | 8.78 | Action, Adventure, Comedy, Drama, Sci-Fi, Space | Cowboy Bebop | 26 | 28.0 | 39 | 147 | 8 |

- At this stage I was able to analyze the data using matplotlib. Those visualizations will be further explored later in this report.

  Histogram of distribution of average user score:



- The merged user rating data was then turned into a pivot table with the indices being the names of the anime show and the columns being the user id. Then this data was converted to a sparse matrix and passed to the machine learning model.

# Data Product Code

The functionality of this program is split into two parts, first, a data cleaning script, and second, the machine learning model.

As mentioned in the previous section, the data cleaning program makes use of the Pandas and Numpy libraries to clean the raw data into a format that is more manageable and acceptable for data analysis use.

For the creation of the machine learning model, I used the following Python libraries: Pandas, Numpy, and Scikit-learn.

There were many iterations of the model where I tried out different algorithms to see which worked best. All of the algorithms that I tried were versions of a Nearest Neighbor algorithm. The algorithm that worked the best was Scikit-learn's knearestneighbor algorithm.

The place where the most changes were made was in how I split the data into testing and training data sets. In my research, before starting this project I saw that many data science projects will use Scikit-learn's test-train-split method in order to split the data set. In the first iterations of the program, I attempted to use this method as well but I struggled to achieve the desired effect with this method. Since the data set being used in the model was a matrix, there was no dependent variable for the model to predict and for that reason, the test-train-split method would not work for my program.

The model was trained on the entire data set that I had. I decided that this was acceptable because the model is not trying to predict any dependent variable in the data set.

Next the model was tested for its accuracy. How the testing was done is explained in a later section.

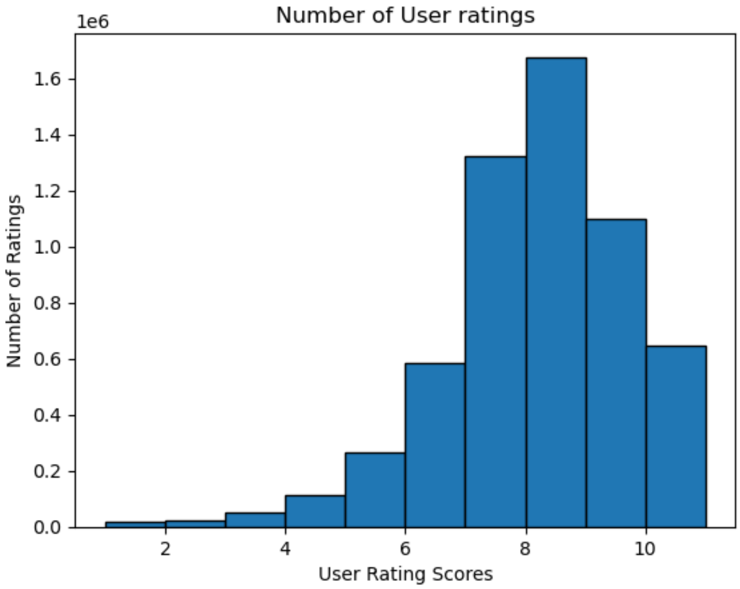
Finally, the code was written for the user to be able to interact with the program. This used Jupyter Notebook's ipywidgets library. I used a combo box that contained a list of every anime show in the data set. I felt that this was important because the people testing the program may not be familiar with any anime shows and also it would limit runtime errors due to misspelled names in a text box.

# Objective (or Hypothesis) Verification

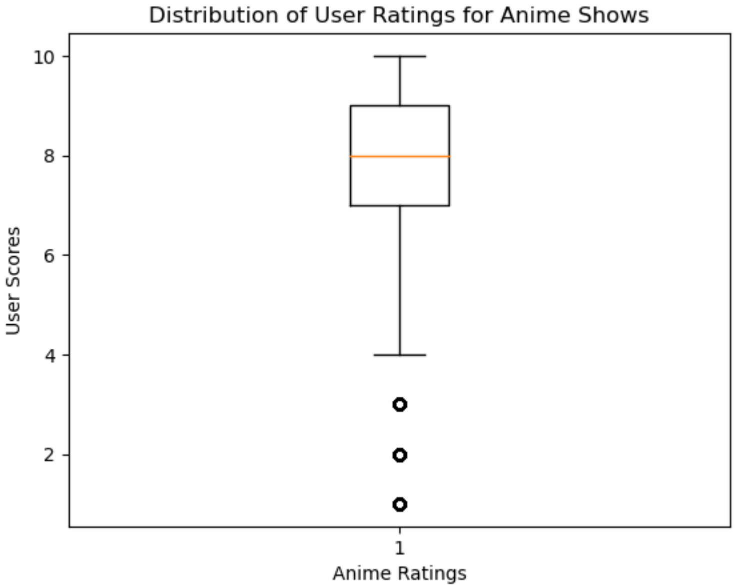
The objective of this project was to use a machine learning algorithm to develop a recommendation system for anime shows that uses collaborative filtering. The difficulty with recommendation systems is that there is no good way to determine how effective the system is without monitoring user activity within that system. The program works as intended in that it can generate recommendations using collaborative filtering. Further analysis of user feedback would be required to fully determine if the system is effective.

## Effective Visualization and Reporting

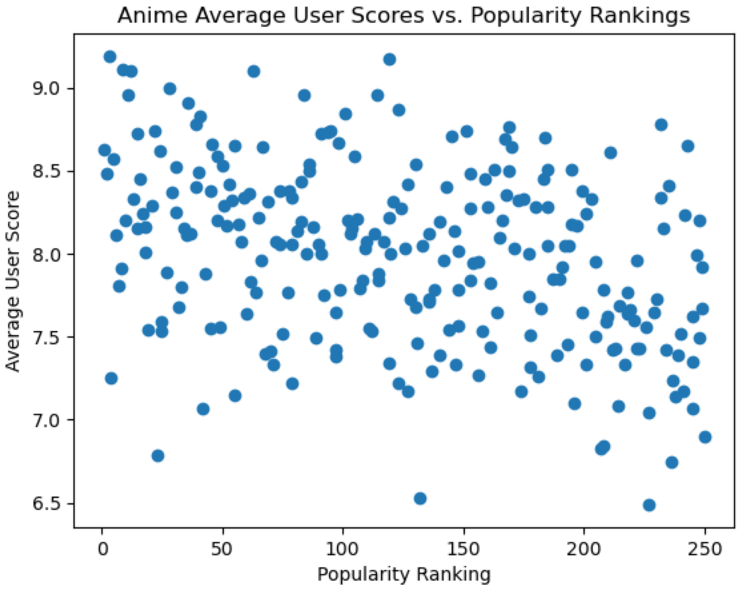
Using the Matplotlib library I was able to visualize and analyze the data using a few different graphs. Using the final data set I created a histogram showing the number of user scores from 1-10. This helped me decide how to define a top-rated show for the users. The largest number of ratings was an 8 out of 10, so I decided that a top-rated show would be anything above an 8 on that scale.



Similar to the histogram, this box plot further visualizes the range, median, and spread of all user ratings in the data set.



This scatter plot visualizes the relationship between the average user score and the popularity ranking of the top 250 most popular anime. This graph does not show a strong correlation between those two variables. If our recommendation system were using content filtering, then popularity would have influenced the recommendations that were generated.



## Accuracy Analysis

To determine the accuracy of the model, I had the model predict 10 shows for every user and then compare to see how many of those shows the user had seen and rated highly. First I decided that if a user rated a show with a score of 9 or 10 that would determine if the show was a top-rated show. I would use the algorithm to try and predict 10 top-rated shows for each user given a random top-rated show for that user. So the test data set was a 2d array of the indices of all top-rated shows for each user, where the outer array was a list of users and the inner array was the indices of top-rated shows. Then, for every user in that array, a random show would be selected and run through the model. Then the program would compare how many of the predicted shows indices were in that user's list of top-rated shows. The accuracy would be calculated as the number of overall correct predictions divided by the total number of predictions made. The model consistently returned an accuracy of around 52%. This fell short of our accuracy goal of 75%

This metric for measuring the accuracy of the model is not perfect. Since it is based on historical data it can only be measured on anime shows that the user has already watched. In order to more accurately determine the accuracy of the model, the system would need to be implemented in a production environment where users could interact with it and also provide feedback.

## Application Testing

The use of Jupyter Notebook was especially useful for the testing of the program. Since I could split code segments into individual cells I was able to test the program as I wrote it. Also, splitting the program into a data cleaning script and the main program helped with testing the program. The data sets that I was working with were very large and the data cleaning portion of the code could take several minutes to run, so having the data cleaning script run separately and create a new CSV file for the main program to read helped reduce testing time.

The program was initially tested using a method that would get a random anime index from the data set and run it through the model. The indices and distances would be calculated and output to the console. This method helped me identify an issue with how I was splitting the data set into training and testing sets. In the first iterations of the program, the model was outputting the same 10 anime shows as recommendations no matter what the input was. This was when I developed a new method to test the accuracy of the model as I described in the previous section.

# Application Files

Here is how the submission files are organized:

/Capstone/

/Anime Recommender Final.ipynb

/anime.csv

/C964 task 2 Documentation.docx

/Data_Cleaner.ipynb

/rating_complete.csv

Other libraries required to run the program are:

- Numpy
- Pandas
- Matplotlib
- Ipywidgets
- Scikit-learn

# User Guide

- This guide assumes that the user has already installed Jupyter Notebook, Python 3, and the previously mentioned Python libraries.
- Using Jupyter Notebook open the file /Capstone/Data_Cleaner.ipynb
- Run all of the data cleaner script. This will create CSV files for the main program to use
- Using Jupyter Notebook open the file /Capstone/Anime Recommender Final.ipynb
- Run the Anime Recommender program.
- In the final cell, enter the name of an anime show and press enter on the keyboard.

## Summation of Learning Experience

During the completion of this assignment, I was able to utilize and expand upon the education that I received while studying at WGU.  The experience that I gained, especially with Python programming and learning about artificial intelligence models, was essential for me to be able to complete this project.

To complete this project, I needed to learn how to use Pandas, NumPy, and Scikit-learn, which were not covered in my coursework. I spent a significant amount of time learning how to use these tools by reading documentation, watching online tutorials, and practicing on sample datasets. I also had to learn how to use the Jupyter Notebook environment, which was new to me.

Overall, this project has helped me develop a better understanding of machine learning concepts and how to apply them to real-world problems. It has also highlighted the importance of continuous learning and the need to constantly update my skills to stay relevant in the field. This project has contributed to my concept of lifelong learning by showing me how new technologies and tools can be learned and applied in a relatively short amount of time with dedication and practice.