Introduction:

Broad Overview:

A major focus of this project was to use Python to make meaning out of a dataset. As a broad overview, our project consisted of parsing a .tsv (tab-separated value) file containing the name of an anime (a Japanese style of film and television) and some of its attributes such as “Best Quote”, “Prettiest Girl”, “Best Guys”. With the data provided, we wanted to devise an algorithm that would systematically rank the animes by assigning a score to each anime. After scoring the animes, we then wanted to display a score breakdown, calculate vital statistics, and graphically represent the data.

Libraries Used:

Python was the main language used for this project and we utilized Numpy, Pandas, and Matplotlib libraries in order to conduct our exploratory data analysis.

Purpose of Project:

It is very common that disagreements over two specific animes and which one is better than the other occur. This problem presents a possible ranking metric where common attributes discussed when describing anime are used, and animes are compared within those attributes. Of course, debate over how the rankings are set is also fair and possible, yet this ranking system aims to diminish the amounts of unjustified reason of how an anime is better than the other.
Ranking algorithms of animes on very popular websites such as myanimelist are based on taking the average of users’ general ratings out of 10. With this possible algorithm, people can now have a quantitative means to explain why an anime is better than the other as a whole by evaluating its placements in the breakdowns.

Division of Responsibilities:

The division of labor required for this project was very equitable. Karthik handled creation/maintenance of Jupyter and Google Colaboratory Notebook to program our algorithm. Karthik also helped to parse the tsv file and design the DataFrame that contained the total score for each anime and the score breakdown for the anime in a DataFrame. Kevin was the main provider of the data used for the algorithm. He designed an Anime class using object oriented principles in order to store information for each of the animes listed. Kevin also conducted an exploratory statistical analysis and designed the graphs using Matplotlib to visually analyze and compare the animes.

Impediments/Challenges:

The use of libraries was an initial challenge to us during construction of the project. Searching up the right methods and algorithms in order to effectively connect the purposes of setting up the Dataframe object, parsing the Dataframe object for vital data, and calculating statistics based on this data. As this was our first time ever utilizing the libraries, a great deal of time was spent on researching the essential functions and coming up with the best code efficiency.

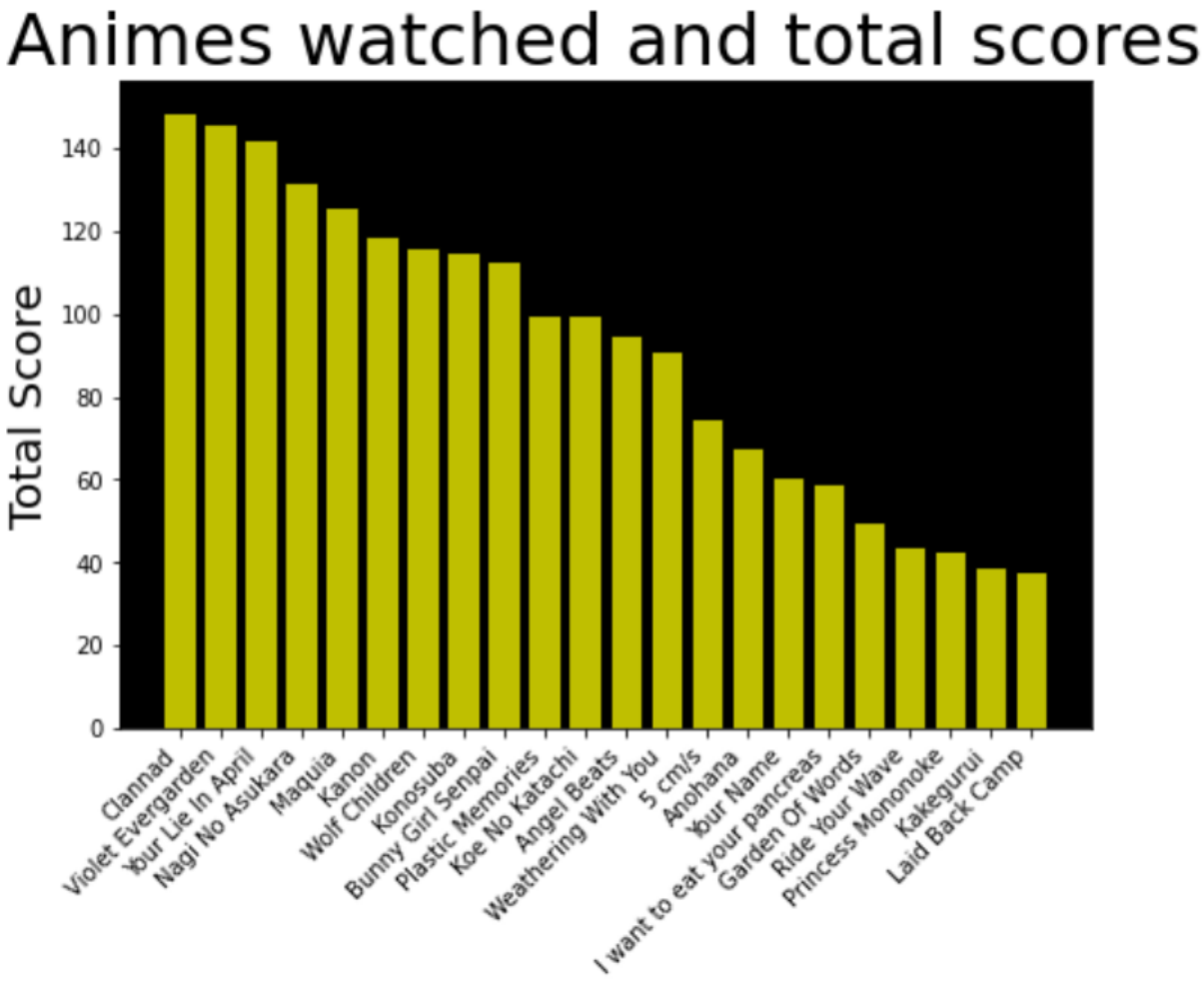
The ranking system required some creativity in order to correctly assign each anime its score based on its placements in the respective categories. For this project, we chose to parse the Dataframe vertically, which in perspective means we analyzed every anime per category instead of which animes were in the same placement of a category. By utilizing a counter and updating the information in the respective anime class object, we were able to come up with a total score for the anime useful in determining final rankings.

Statistical Techniques:

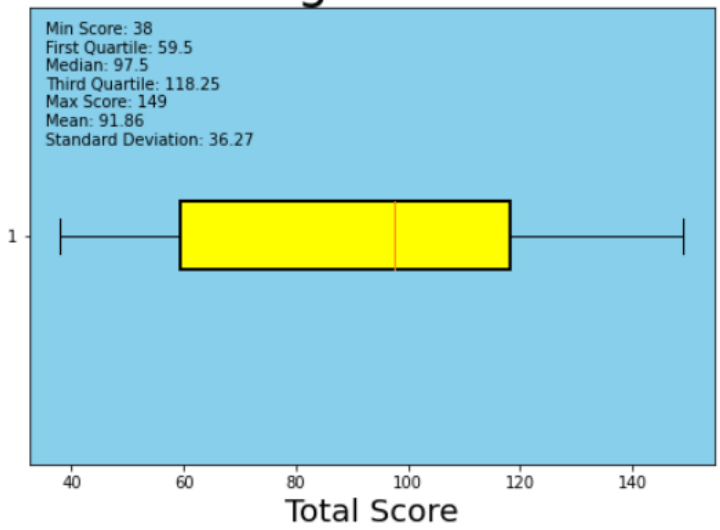
Standard statistics were calculated from the anime's data, including average score along with the respective standard deviation, and the 5 number summary (minimum, first quartile, median, third quartile, and maximum score). With this data, along with the Matplotlib library used, we created a sorted bar graph and boxplot in order to better visually comprehend our project.

Possible improvements to this data involve normalizing the data so different values do not affect the overall distribution more than others, and using a bigger anime dataset in order to have a bigger sample size and, as a result, more statistical reliability.

Picture of the Two Graphs:



Box plot demonstrating distribution of total scores



Possible Improvements and Extensions:

While this project provided a visually satisfying perspective on the numerical judgement of anime, the methods are by no means completely perfect.
The efficiency of certain methods in the code must be restudied to ensure that we are parsing, judging, and analyzing the data in the most efficient way possible. In this way, we provide the best solution for potentially bigger sets of data than the one used for this project.
Another relatively minor coding flaw in this project was how to cut ties. For example, two animes would be sorted in alphabetical order in our project should they achieve a final score equal to each other. Possible solutions include weighing different aspects of an anime differently depending on its importance to the user; unfortunately, this solution is purely subjective and not satisfactory in providing a clear conclusion since different viewers desire different things. Since rankings are shown by a counter, a solution that would be best would be to manipulate the increment of the counter, or use a different method of ranking other than a counter.
In this project, we clearly knew the amount of anime we were judging, and as a result hardcoded the boundaries of the Dataframe. The problem with this is that future datasets using this project only need a different number of anime to be judged before the code turns faulty. The simplest solution for this problem would be to devise a method so that the Dataframe parses the number of animes right no matter the number, whether by a keyword or some other mechanism.

Possible extensions that would be further helpful to the user could be more information about the anime itself. For example, we could gather more data on which studio created the anime, how much money went into the anime's production, and other attributes besides those in the Dataframe. While this does not directly impact the actual ranking system of the animes, this could potentially intrigue the user into doing more research into anime and stimulating possible interest in the subject.
Sentiment analysis on the quotes, and potentially a ranking system based on this, is another hypothetical extension. Using machine learning techniques, we can use the "quotes" attribute in the Dataframe to get a very general idea of how emotional the anime purports to be based on the quote provided using sentiment analysis. Along with the "emotional" ranking by the anime, the user can get a clearer picture and a deeper understanding of animes he or she might be interested in.