Data Wrangling Final Project

Brianna Eskin

4/14/2020

Introduction

The game show Jeopardy!, created by Merv Griffin and hosted by Art Fleming, first premiered on March 30, 1964 on NBC. After a few different syndicated versions, the show as it is known today, hosted by Alex Trebek, premiered on September 10, 1984. Since its premiere, the current version of Jeopardy!, which is in its 36th season, has aired over 8000 episodes, won over 30 Daytime Emmy awards, and is often regarded as one of the greatest television shows in American television history.

Despite a few minor tweaks over time, the gameplay of Jeopardy! has remained largely the same. There are three rounds in a standard game of Jeopardy!, the Jeopardy round, Double Jeopardy, and Final Jeopardy. In the first two rounds, contestants select clues from six predetermined categories valued by difficulty. These clues are presented as "answers" and the responses from the contestants are given in the form of a question. For example, the clue might say "This is the official state university of New Jersey". The first contestant to buzz in would give the response of "What is Rutgers University?". If the contestant responds correctly, he/she is awarded money based on the value of the clue. If the contestant responds incorrectly, he/she loses money based on the value of the clue. Under some clues- one in the Jeopardy round and two in the Double Jeopardy round- there is a Daily Double. When the Daily Double is found, the contestant, who has the clue all to themselves, can wager any or all of the money they have available, and gains or loses that amount depending on their response to the clue. After the first two rounds, there is a Final Jeopardy round. In Final Jeopardy, the contestants are given the category and based on factors including their confidence in the category and the scores of themselves/their opponents, wager any or all of their winnings, similar to a Daily Double. Then the clue is revealed and the contestants write their answers at their podium. Their reponses and wagers are then revealed and after this round, whoever has the most money is declared the victor and comes back the next day as the returning Champion, getting to keep all of the money they earned.

For my project, I will be wrangling data pertaining to the all time highest Jeopardy! winners, as listed on https://www.jeopardy.com/contestant-zone/hall-of-fame. As of May 4, 2020 these contestants are:

1. Brad Rutter
2. Ken Jennings
3. James Holzhauer
4. David Madden
5. Larissa Kelly
6. Matt Jackson
7. Jason Zuffranieri
8. Roger Craig
9. Colby Burnett
10. Julia Collins

For each of the all time highest Jeopardy! contestants, I will be analyzing their overall winnings, as well as their individual regular season and tournament gameplays. I will be looking at their distribution of money earned in regular season play versus the various tournaments Jeopardy! has held over the years. I will also be analyzing individual game scores and statistics.

I was interested in data pertaining to Jeopardy! because of the personal connection Jeopardy! has in my life. Jeopardy! is my favorite television show, and one of the few television programs I watch on a consistent basis. I started wathcing afternoon reruns with my grandmother while she babysat us during school vacations, but given that I was in elementary school, I couldn't keep up most of the time. After she passed away in 2007, I fell off of watching for a bit, but got back into it during high school. My dad would start watching with me, not necessarily because he was also into it, but because I would take up the good television in our living room and he didn't feel like moving. Over time, both of our investments in the game began to grow and for years, we never missed a game, even while he was spending nights in the hospital due to complications from cancer, until he passed away in 2018. Jeopardy! has left such a huge mark on my life, and because there are so many aspects of Jeopardy! to analyze, this made for good data to wrangle.

Data Sources

For my project, I will be scraping data from the J-Archive (https://j-archive.com/), which is a fan run archive of most of the games from the Alex Trebek era of Jeopardy!. The site is broken down to different pages for each individual game. To find the different games each contestant played in, I will scrape from their different player profile pages, 16 pages in total, which were found by manually checking the j-archive, as one player can have multiple player pages. An example page for Jason Zuffranieri is http://www.j-archive.com/showplayer.php?player_id=12824. On this page, I will be pulling the various games each contestant played in and scraping data from that specific game's web page. An example is based on Jason's player page- I know he played in Game #8046, which is also the 36th season's permiere. So I will be scraping data from that page, http://www.j-archive.com/showgame.php?game_id=6410. For that specific game, I will then be scraping data from the score breakdown page, which for the example game #8046 is http://www.j-archive.com/showscores.php?game_id=6410. I will be repeating this process for each game a contestant has played in. Using Jason as an example, he has played in 20 regular season games of Jeopardy!, so I will be scraping 20 game level sites and 20 score breakdown sites.

All of the code for this project can be found at: https://github.com/briannaeskin/DataWranglingProjectSpring2020

Data Scraping

The first thing I did to start scraping data was I created a static data frame called contestants, whose structure is as below:

Table 1: contestants Table Structure

ColumnName	Definition	Type
Contestants	Full name of a top 10 all-time winning contestant on Jeopardy!	character()
playerIds	Player ID as extracted from the j-archive	integer()
AllStarsTeam	The team a contestant played on during the All-Star tournament in 2019- marked as No Team if not a participant	character()

Next, I created my main data frame, which contains a record of each game a contestant has played of Jeopardy! and some relevant stats for each game. The structure is as follows:

Table 2: Main Table Structure

ColumnName	Definition	Type
Contestant	Full Name of Contestant	character()
GameIdForUrl	The Game ID that will be passed into the URL for scraping	integer()
GameId	The official Jeopardy! Game ID	integer()
Date	The date the episode premiered	character()
FinalScore	The contestants final score (not necessarily winnings)	integer()
Outcome	Contestant's placement in that game. Also notes if game was an accumulated score, such as a Tournament Final	character()
GameFormat	Was the game part of a tournament or the regular season	character()
TournamentName	Tournament name, or regular season if not a tournament	character()
CoryatScore	Score unadjusted for Daily Doubles and Final Jeopardy	integer()
QuestionsRight	Number of questions answered correctly	integer()
DailyDoublesRight	Number of Daily Doubles answered correctly	integer()
QuestionsWrong	Number of questions answered incorrectly	integer()
DailyDoublesWrong	Number of Daily Doubles answered incorrectly	integer()

To fill this table, I had to scrape the game level and score sites for every game played by these Jeopardy! contestants. First, I had to identify every game a contestant played in. To do that, I created a function, getGameIdsForUrl, which inputs a playerId from the contestant table and scrapes the player site for the playerId. The output is then a list of gameIds that will eventually be passed into URLs. In order to do this, I scraped each site using the read_html() function and html_nodes(), using “a” HTML tag since I was interested in the hyperlinks on the page. I then mapped the returned xml nodeset as a dataframe. For the next step, I filtered out unneccessary hyperlinks by filtering links which had “game_id” as part of the link. Once my list was limited to only hyperlinks that pointed to show games, I was able to extract the gameId in the URL with the regular expression “\d+”. The function then outputted this list of Ids.

As a next step, I iterated through this list of gameIds and passed each one into a new function, getRowForGameLevelTable. This function requires the contestant name, the gameId that will be passed into a URL, and the All-Star team the contestant played on during the All-Star Games tournament in 2019. In this function, I scraped two different sites, the game level site and the score site, using the gameId I extracted earlier. Using these two sites, I was able to extract the remiaining information required for my table. I've provided a brief explanation as to how I pulled the remaining attributes required below:

1. GameId and Date: These attributes were pulled from the game level site, reading the “h1” node and extracting based on regular expressions. An example h1 is “Show #8059 - Thursday, September 26, 2019”. For the GameId, I used the expression “(?<=\#)\d+”, to pull all digits after the #. For the Date, I used the expression "(?<=-\s).*(?=\<)" to pull everything after the - , but stopped if I saw a “<”, to avoid pulling the end of the tag as well.
2. FinalScore: To find the final score, I read all of the tables using html_table() from the scores sites, taking the 4th table. After some cleaning to account for cases where the game was part of a multi-day contest in which the winner was decided by the sum of scores over multiple days- such as tournament finals, I ended up with a table that had the contestants, scores, and outcomes. I filtered on the contestant I was interested in and pulled the final score from the second column, using the regular expression “\d+,\d+|\d+” to take all of the digits. I then removed the commas and converted into an integer.
3. Outcome: Using the table I created to extract the final score, I pulled the Outcome using the raw outcome column without any transformations.
4. GameFormat: From the game level site, I extracted nodes called “#game_comments”, which normally includes information such as the contestants playing in that game, any special highlights, and the tournament name, if applicable. To find whether or not a game was part of the regular season or

tournament, I read the first of these “#game_comments” nodes, and set the format to Tournament if the regular expression “Tournament|Masters|Decade|Star” was found. Another possible option for GameFormat is Exhibition, which refers to the IBM Challenge in 2011 in which Brad Rutter and Ken Jennings played Jeopardy against IBM Watson. I set the GameFormat to Exhibition if I matched the expression “IBM”. Otherwise, the format was set to Regular Season.

5. TournamentName: To pull the tournament name, I used a series of ifelse functions checking various regular expressions to match the different tournament names that have occurred over the years. If there were no matches, the TournamentName was set to Regular Season.
6. Coryat Score: I followed the exact same steps as what were used to pull the final score, the only difference is that I pulled the last table from the scores site, instead of always pulling the 4th table.
7. QuestionsRight, DailyDoublesRight, QuestionsWrong, DailyDoublesWrong: I followed slightly similar steps to Outcome, because all of these attributes could be extracted from the third column of the last table from the scores site. To extract all of these values from one cell, I first split the column into two by a comma into a “right” column and a “wrong” column, and then split those columns each into two using the regular expressions “\sR\(including\s|\sDD|\sDDs\)|\sR” and “\sW\(including\s|\sDD|\sDDs\)|\sW”, respectively.

The full data frame has also been exported as a csv: https://github.com/briannaeskin/DataWranglingProjectSpring2020/blob/master/JeopardyTop10PlayerGames.csv.

All-Time Winnings

The first piece of data I am planning to analyze is a breakdown of how much money each contestant has earned over different tournaments versus regular season play. I will store the tournament winnings in a new table, tournament_winnings_df, whose structure is:

Table 3: Tournament Winnings Table Structure

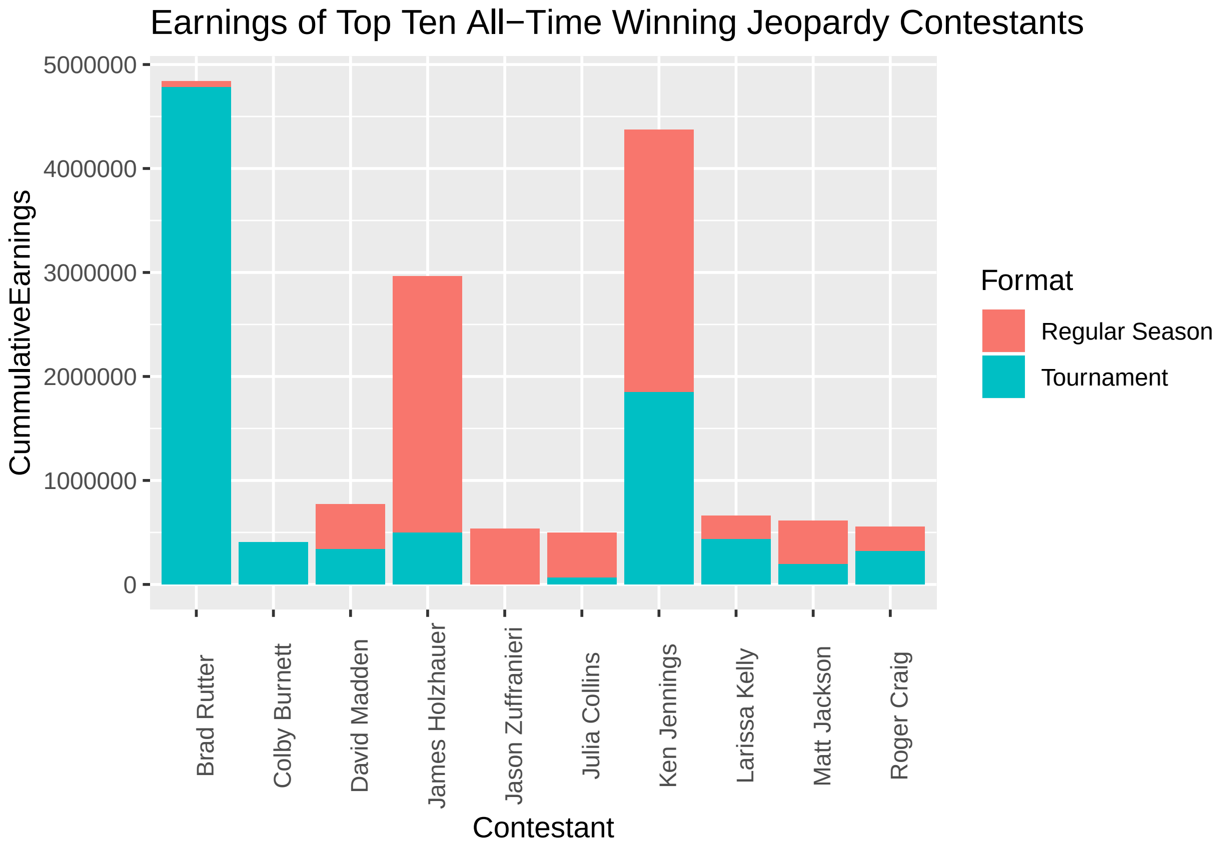
Columns	Definition	Type
Contestant	Contestant Name	character()
TournamentName	Tournament Name	character()
Outcome	Contestant’s performance in final game they competed	character()
Winnings	How much money was won - not necessarily their final score	integer()

To extract the winnings for each tournament a contestant played in, I pulled the unique Contestant/TournamentName pair from my main game level dataframe. Then, filtering through that dataframe, I went back into my main game level dataframe and pulled the last game a contestant played in that specific tournament based on the max gameId. I then checked if the game was an Accumulated Total. If it was, I had to scrape a new table from the scores site for that particular game, which was the table that reflected the accumulated scores over the multiple games in that round of a tournament (usually the finals). From here, I was able to pull the tournament outcome in a similar fashion to how that information was pulled for the main game level dataframe. If the game wasn’t an accumulated total, all I needed to do was pull the outcome from my main game level dataframe. I was then able to pull winnings from the Outcome using the regular expression “(?<=:\s)[\$\d+,]+”. The All-Star Games were special in that the winnings were divided evenly among the three members of that team, so that is accounted for as well by splitting the winnings by 3 when the tournamentName is the All-Star Games.

Next, I needed to clean the data pertaining to the regular season winnings. For the most part, the winnings for a particular game could be pulled from the FinalScore column in my main game level dataframe. The exception to this is when a contestant loses. In Jeopardy! regular season play, third place contestants leave with $1000 and second place contestants leave with $2000. In addition, prior to Season 20, which premiered September 8, 2003, there was a 5 game limit for Champions, meaning if they went 5 consecutive games without losing, they were “retired” and three new contestants played the next game. One of the contestants, Brad Rutter, played his original Jeopardy! run under this rule, so when pulling the winnings for Brad, I was always able to use the FinalScore, since his winnings in regular season play always matched what he earned. For everyone else, I had to pull their final regular season game using the max gameId, since their last game would have been the game they lost. I then pulled the winnings from the Outcome in a similar fashion to how I did for the tournament winnings, and adjusted the FinalScore for that game only.

Once I had two separate tables, one for the tournament winnings and one for regular season winnings, I merged the two together using rbind.

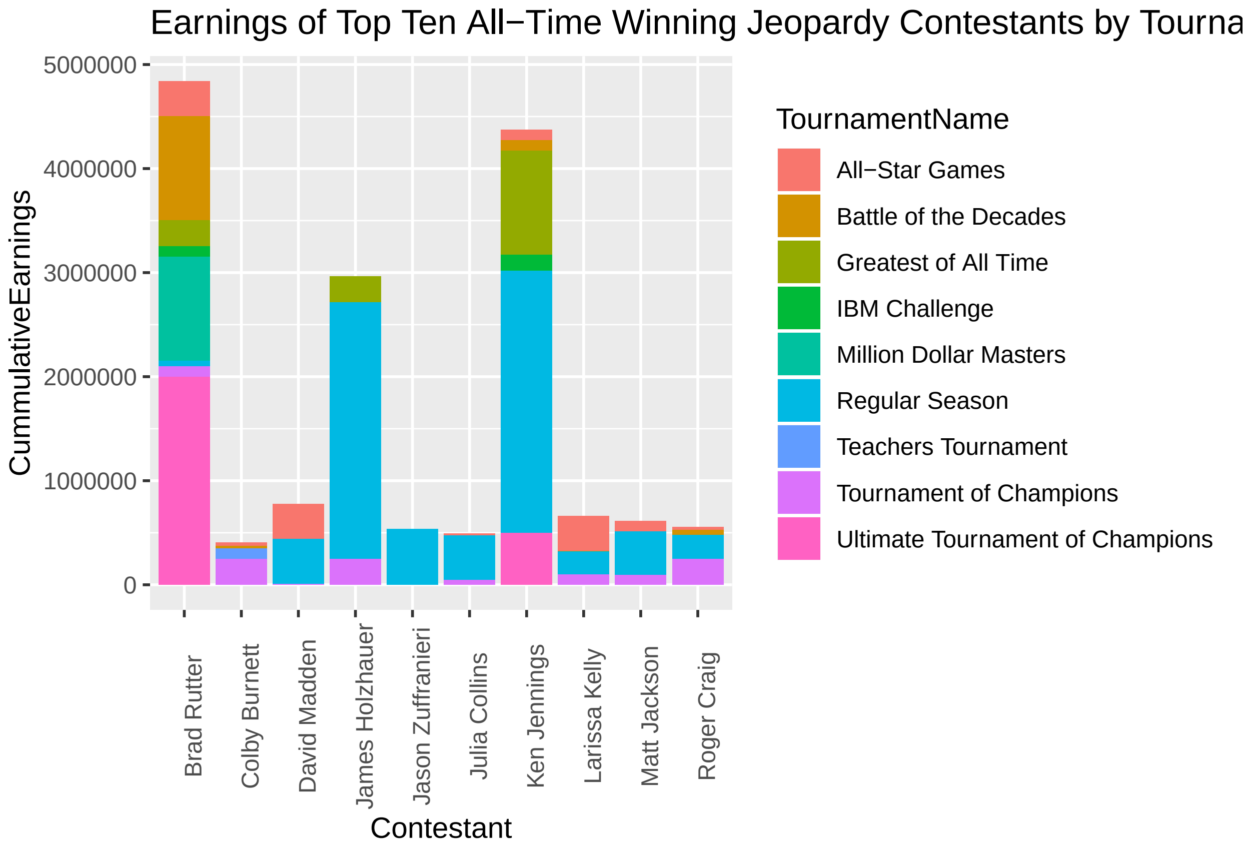
Now that I have one table which stores the winnings for each Jeopardy! contestant, I was able to plot a graph of winnings for each Jeopardy contestant. I chose to use a bar graph where the fill was the Format (Regular Season or Tournament).



There are some observations we can make from the above table. Brad Rutter is the highest all-time winning Jeopardy player, earning almost $5million. As stated previously, Brad’s regular season appearance on Jeopardy! was during the years when contestants were limited to only 5 games. In addition, prior to November 26, 2001, clues were only valued at half of their value today. This also resulted in Brad earning less money in his regular season appearances, only around $50,000. Another interesting observation is how much money Ken Jennings and James Holzhauer won during their regular season appearances compared to other contestats, winning over $2million each. Ken Jennings and James Holzhauer are both known for their long winnings streaks, with Ken winning 74 games and James winning 32. During their streaks, they were also very aggressive players, which allowed them to earn so much money, whereas tournament payouts are fixed sums. Other contestant’s winnings are much more evently distributed when compared to their overall winnings. Two notable expections are Colby Burnett and Jason Zuffranieri. Colby Burnett has only ever played in tournaments, making his debut in the 2012 Teacher’s Tournament, and going on to play in the subseuqent Tournament of Champions, Battle of the Decades, and the All-Star Games. As for Jason Zuffranieri, he is the most recent contestant in this list, having made his first regular season appearance in

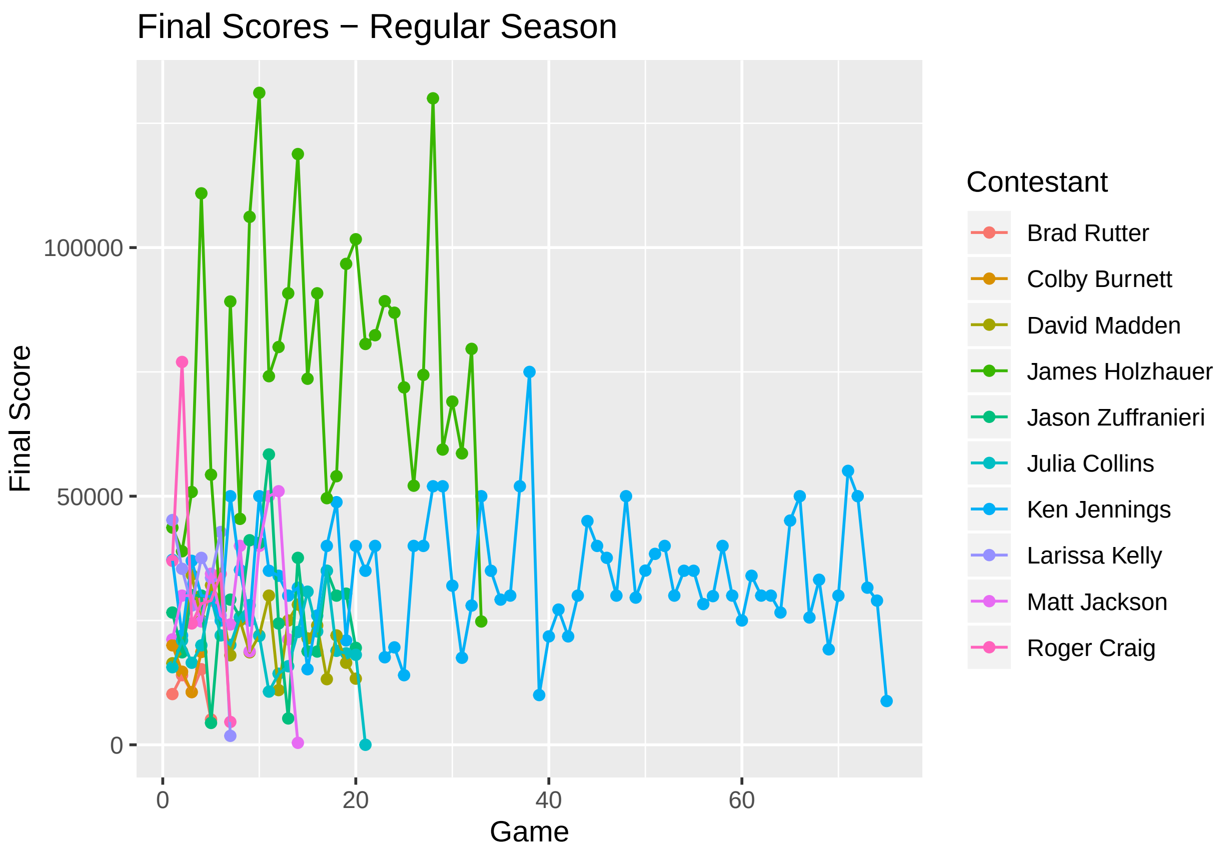
July 2019. Because the cutoff for the msot recent Tournament of Champions was during his regular season run, he was ineligible for that Tournament of Champions and has yet to compete in a tournament.

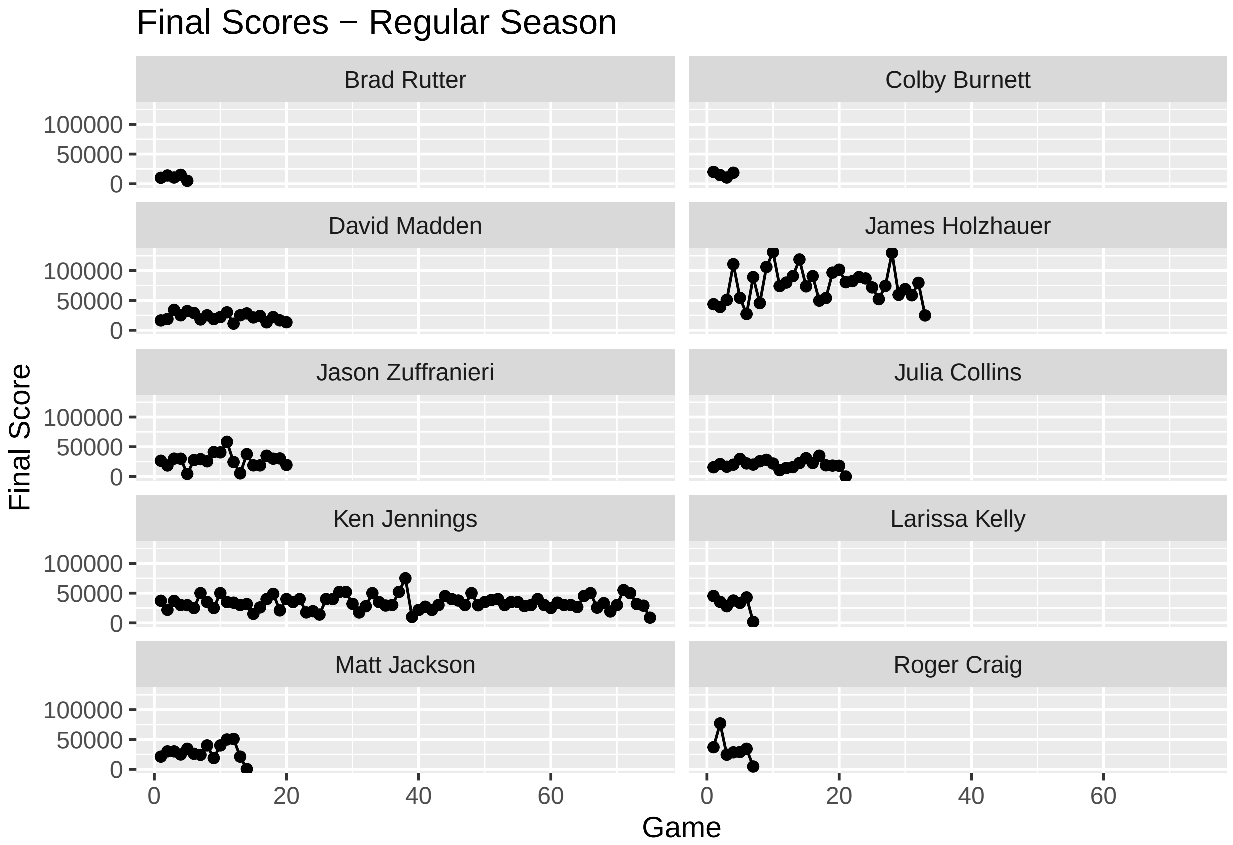
The below table is the same cut of data, except I split out the Tournament winnings by indiviudal tournament as opposed to one sum:



Scores Per Game

The next statistic I will be looking at is each contestant's final score (not necessarily winnings) for each game during their initial appaearance on Jeopardy!. For Colby Burnett, that is the Teacher's Tournament, and for every other contestant, it's their regular season appearance. I will be plotting two sets of graphs. The first graph is a tracing of each contestant's final score over the course of their run, all in one plot. The second plot is split into facets for each contestant.


Final Scores − Regular Season
Final Score
100000
50000
0
0
20
40
60
Game
Contestant
Brad Rutter
Colby Burnett
David Madden
James Holzhauer
Jason Zuffranieri
Julia Collins
Ken Jennings
Larissa Kelly
Matt Jackson
Roger Craig



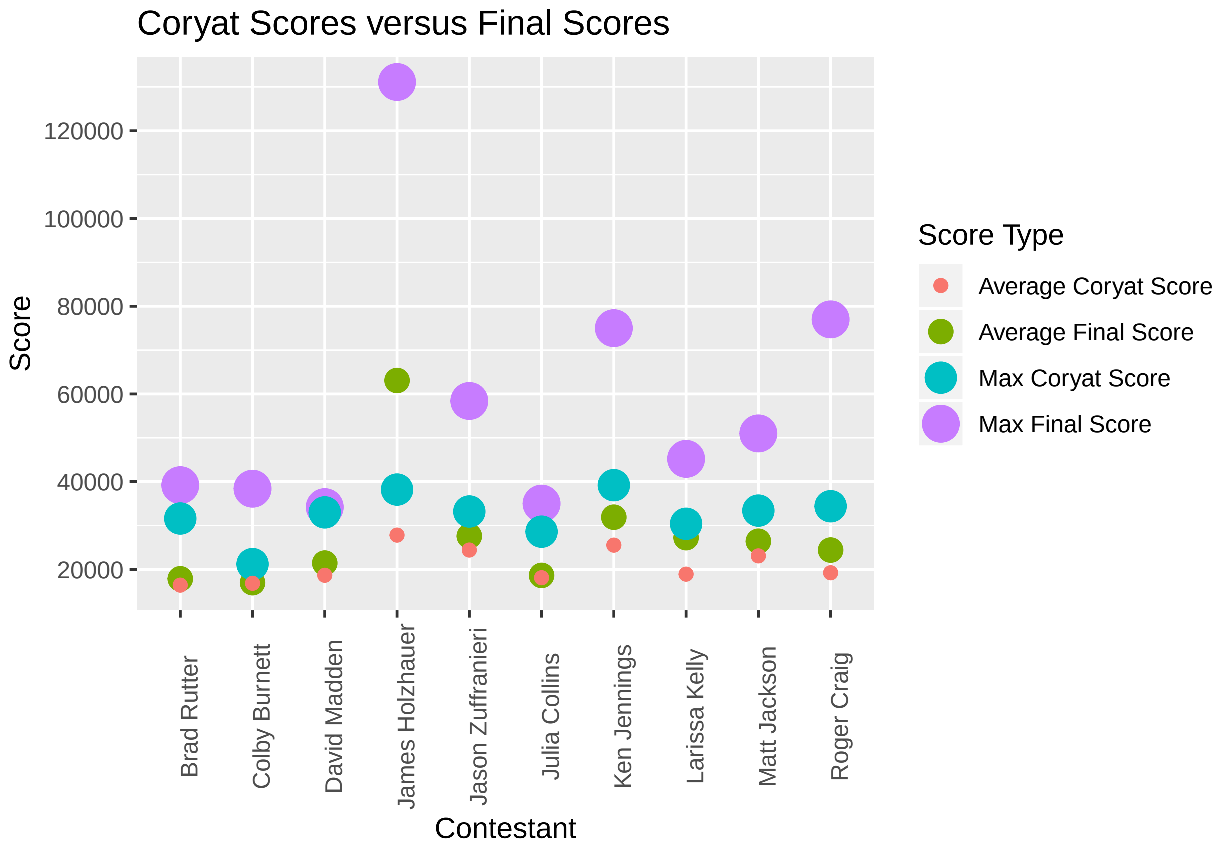
Two of the most interesting observations that can be made by looking at these graphs is how low Brad's scores are in comparison to everyone else, and how high James's scores are compared to everyone else. As previously mentioned, Brad's original Jeopardy! appearance was during an era when clues were worth half of the value they are worth today. Therefore, Brad's scoring potential was only 50% of that of the other contestants. As for James, his strategy was to hunt for and bet large on Daily Doubles. That, combined with his high buzzer speed and accuracy, helped him cross the $100,000 mark in a game on multiple occassions.

Because of a lot of other variables such as luck, risk aversion, and earning potential due to clue values and Daily Double placement, the score at the end of the game is not always indicative of who played the best that game. There are often games where a contestant played better than their opponents, but lost because one of the other contestants hit all of the Daily Doubles or because they themselves bet big on a Daily Double and missed. The next statistic we will discuss, the Coryat Score, takes more of these outside variables into account.

Coryat Score

According to the J-Archive!, the term Coryat Score was coined by Season 12 2-game champion Karl Coryat, and is the contestant's score disregarding all wagering. It was originally created as a metric that could be used by viewers at home to gauge their performance against the contestants (Source: http://www.j-archive.com/help.php#coryatscore). This score has since been adapted by the J-Archive to reflect a "raw" score for each game. Final Jeopardy is not taken into account, and for Daily Doubles, there is no penalty for an incorrect response, since you are forced to give a response on a Daily Double, whereas if it was a regular clue you aren't forced to buzz in if you do not know the answer. However, the trade off is that if answered correctly, the clue is worth its original value.

The below graph plots each contestant's max/average Coryat scores, as well as their max/average final scores, to give an idea of how much Final Jeopardy and Daily Doubles play a role in a contestant's final score. This time, to account for Brad playing during the pre-doubled values era, his regular season final scores and Coryat scores have been doubled here. These plots are for each game a contestant has played in, excluding the All-Star games. I excluded the All-Star games from this analysis because each round was played by a different contestant in a team, and therefore the Coryat scores and final scores cannot be accurately calculated.



Now by looking at this plot, especially when compared to the game level final scores, we see that when taking into account Coryat's and doubled clue values, Brad's scores are more in line with other contestants. Similarly, while James still has one of the highest max and average Coryat scores, the variance is a lot less, being around $10,000 higher than most of the other contestants, and being much more comparable to Ken's scores.

Accuracy

As mentioned, luck plays a major factor in one's success on Jeopardy!. There is luck in the categories that are on the board- are they aligned with a contestant's strengths or are they aligned with an opponent's strengths? There is luck in who you are playing against- do they have strong reflexes and can they ring in on the buzzer faster? One of the most luck factors are the three Daily Doubles in the game- can you hit the them, are you confident in your abilities to answer the question in that specific category, and can you provide the correct response? Below, we look at each contestant's Daily Double hit ratio per game (max is 3) and the percentage of Daily Doubles they answered correctly, across every game except for the All-Star tournament:

Table 4: Daily Double Stats

Contestant	DailyDoublesPerGame	DailyDoubleRightPct
Brad Rutter	1.514286	75.47170
Colby Burnett	1.363636	66.66667
David Madden	2.090909	95.65217
James Holzhauer	2.000000	92.22222
Jason Zuffranieri	2.100000	80.95238
Julia Collins	1.600000	72.50000
Ken Jennings	1.924731	82.68156
Larissa Kelly	1.500000	77.77778
Matt Jackson	1.944444	91.42857
Roger Craig	1.750000	89.28571

To create this table, I filtered my main game level table to exclude the All-Star Games, grouped the rows by contestant, then created a few new variables using summarize()- GamesPlayed, which was equal to the number of rows, DailyDoublesTotal, which is the sum of how many Daily Doubles they answered correctly and how many were answered incorrectly, DailyDoublesPerGame, which was the DailyDoublesTotal/GamesPlayed, and DailyDoubleRightPct, which was the percentage of Daily Doubles a contestant answered correctly.

Let's compare this to their overall accuracy:

Table 5: Overall Response Stats

Contestant	QuestionsAnsweredPerGame	QuestionsRightPct
Brad Rutter	23.34286	90.33048
Colby Burnett	21.09091	92.24138
David Madden	25.04545	92.01452
James Holzhauer	33.71111	96.24258
Jason Zuffranieri	29.95000	94.32387
Julia Collins	24.72000	91.90939
Ken Jennings	35.58065	91.56845
Larissa Kelly	25.41667	92.45902
Matt Jackson	27.66667	96.18474
Roger Craig	29.06250	86.45161

To create this table, I filtered my main game level table to exclude the All-Star Games, grouped the rows by contestant, then created a few new variables using summarize()- GamesPlayed, which was equal to the number of rows, QuestionsTotal, which is the sum of how many Daily Doubles they answered correctly and how many were answered incorrectly, QuestionsAnsweredPerGame, which was the DailyDoublesTotal/GamesPlayed, and QuestionsRightPct, which was the percentage of Daily Doubles a contestant answered correctly.

With 60 questions in a standard Jeopardy! match (excluding Final Jeopardy and assuming every clue is revealed), three contestants who are evenly matched would answer 20 questions correctly, with one being a Daily Double. Each contestant is over those numbers, with 4 players, James, Jason, Ken, and Roger, answering 50% of the questions. This usually means strong buzzer speed, and that comfort ringing in on clues is key to being successful on Jeopardy!. In addition, each player has a very high accuracy on questions they attempt to answer, proving that they can also execute once they do ring in.

Conclusion

Every contestant discussed in this report is a Jeopardy! legend for one reason or another. Brad Rutter has won almost every tournament he has appeared in, losing in regulation play for the first time ever in the Greatest of All Time tournament earlier this year (The IBM Challenge is considered an exhibition for record purposes). Ken Jennings and James Holzhauer are known for their long streaks and high regular season winnings, winning over $2million each. Julia Collins, Jason Zuffranieri, Matt Jackson, and David Madden round out the Top 6 of longest winnings streaks and highest regular season earnings, with David also being on the winning team in last year's All-Star Games. Larissa Kelly was the first woman to win over 5 games of Jeopardy! after the win limit was lifted, and is currently the highest winning female player. Roger Craig previously held the record for highest single day earnings and was the player to revolutionize the "Go Big or Go Home" approach to Daily Doubles. Colby Burnett was the first (and only) contestant to win both the Teacher's Tournament and the Tournament of Champions. Over the course of this report, we took closer looks at their winnings and how they were earned, their game level scores, Coryat scores, and their response accuracy. While some contestants excelled more in certain categories than others, the common theme amongst them is that every facet of their gameplay was exceptionally high, and showed that both skill and a little bit of luck can help anyone succeed at Jeopardy!

Over this project, my goal was to apply as many of the techniques we learned in class as possible, while still making sense in the context of the data I was wrangling. I scraped over 600 webpages and cleaned data from all of them, created data frames, wrote data to a csv file, and generated tables using knitr. I utilized many features of the tidyverse, including filtering and creating new variables in a data frame, created different types of plots, such as bar graphs, line graphs, and facet graphs. Given the different variations of tournaments and matches Jeopardy! has done over the years as they continue to innovate their show, that made some aspects of web scraping and cleaning very difficult, and a lot of my project effort was spent just on creating a main data source from which I could create the rest of my plots and charts from. Towards the last few days of my project, I encountered some issues with Latex, and an entire day was wasted reinstalling Latex. Also, as someone whose R experience is limited to undergrad class experience only, this project helped me get more comfortable with the software and RStudio, as up until this class I only ever used the Console, which is more difficult to use as the scope of a problem gets larger and larger. Overall, I learned a lot from this class, and this project.