Chloe Hicks & Abigail Thomas

STAT 2984 Final Paper

Dr. Tegge

May 3, 2018

For our final project, we wanted to see to see if the population of a state had any relation to how many people from that state are ranked in the top 160 of the MLB. We decided to do this because we are both interested in sports. We have both been involved in sports for our whole lives, but have never directly been involved in baseball. However, even though we both haven't been directly involved in baseball, we both have family members involved in the sport. My grandpa and dad were always very involved in baseball (Chloe) and my brother played on an allstar team for years (Abby). Also, when we were deciding on our question, we knew that baseball data would be easy to access. We have used it multiple times in other statistics classes and if we wanted more data, we could go to MLB.com to get more information.

When we looked at the data sets from MLB.com, we decided to look at the pitching data. Even though we both have close family members that love baseball, we do not know much about the sport ourselves. We had the thought that the hitting stats would be the best indicator on how good you are in terms of how many runs you could score. However, we do understand that everyone has different strengths and that just because you are not as good of a hitter, does not necessarily mean that you are not as good at baseball. Once we chose out MLB data set, we were looking through and deciding to figure out what would be a good thing to compare this data set with. We noticed that a lot of the players came from teams that were in states with large

populations. For example, we were constantly having players that were from California or Texas. So, I (Abby) thought that the state population could have an influence on whether you were a good baseball player. We also knew that population data would be easily accessible. We found data on the US Census on a US Census website called census.gov. This data was one of the more lengthy data sets we found. There was a population for every city and every county in each state. Because of this, we decided that it would be best if we cleaned up the data by deleting all of the variables that we did not need. This made the data easier to read and therefore, easier to code in python.

Before moving on about how we merged our data, I think it is important to know one of the issues we found with both of the data sets. First, the Toronto Blue Jays are included in the MLB, but not included in the US Census. Because of this, we had to disregard the Blue Jays. Second, we needed to have a common variable in both data sets. We decided that the 'State' would be the best option, however, the MLB data set did not have a variable that was clearly the state in which the team was from. But, since we know where each team is from, we were able to create a new column in the MLB data set called 'State' and look up exactly where each team was from and add that to the new column. And lastly, we ran into the minor problem that the US Census data set had a column for the states, but it was called 'STNAME'. This was a problem because python did not recognize that our data had a common variable. We were able to easily fix this problem by changing the name of the column to 'State' so that it matched the MLB data set.

```
# read in player file
f1 = open(player_file, 'r').read().split("\r\n")
player_header = f1[0].strip().split(',')
player_data = []

# skip header column
for line in f1[1:]:
    player_data.append(line.split(','))
```

```
# read in population file
f2 = open(population_file, 'r').read().split("\r\n")
population_header = f2[0].strip().split(',')
population_data = {}

# skip header column
for line in f2[1:]:
    population_array = []
    line_array = line.split(',')

    # ensure variable doesn't show up twice
    if line_array[0] in population_data:
        continue
    else:
        population_array.append(line_array[1])

    # build dictionary of population data
    population_data[line_array[0]] = population_array
```

We started our code by reading both of our data sets into python, which is shown above. We were able to read in the data like this from the help of the class notes we have taken throughout the semester. For the first code block, we had to go into our MLB data and take out the double spaces between the rows and the commas in the player name because it was messing up the columns. Because of this, the data also was not running as smoothly as we would've liked. The first part of the second code block is similar to the first, as it is code for reading in the US Census data. The second part of the second block of code was code that we found from the notes we took in class about merging data sets and we chose to use it to make sure the variables did not show up twice.

```
# initialize data to empty list
merged_data = []

# go through every line in data
for line in player_data:
    if line[18] not in population_data:
        # Key not in population data
        continue
    else:
        merged_data.append(line + population_data[line[18]])

for l in merged_data:
    print l
```

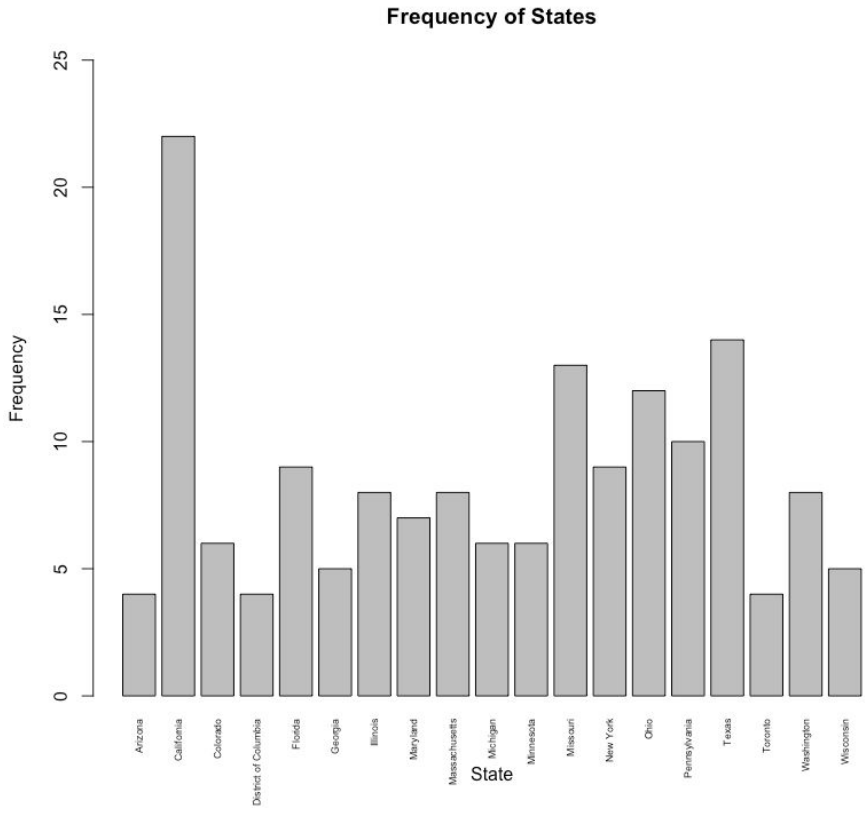
This was the code block for the actual merging of the two data sets. This was one of the more challenging parts of the project being that we are both very new and inexperienced with

coding. Because of this a lot of this code was made by getting help from our notes taken in class, from the internet, and from our friends who were able to help us debug our code.

After successfully reading in and merging our datasets, we then began running the summary statistics for our data. We ran summary statistics for the hitting average and the population of each state. For the state population, we had a maximum of 37,253,956 people in California and a minimum of 601,723 people in Washington D.C.. Next, we wanted to look at the summary statistics for the total number of hits per player. You can see in the table below the summary statistics for the hits. Based off of these you can see that there is a wide range within the top players for how many hits they are making. We thought that a reason for this wide range is because some positions are not required to hit as well as other positions, so they may not be the best hitters but they could be the best at something else.

Minimum	Quartile 1	Median	Mean	Quartile 3	Maximum
65	98.75	108	109.16	121	167

The next thing that we had to do was to create our visuals and graphs. To better look at the data for the states we chose to look at the frequency at which each state appeared in the top 160 players. Based on the chart you can see that California appears drastically more



than any of the other states do. In fact it is included 22 times and Texas is the closest behind it with 14 appearances in the top 160 players. Also from this graph you can see that not all 50 states are included in the data set. This is due to the fact that the states came from where the team is located and not every state has an MLB team. Reflecting back now we would look up where each player was born rather than where the team is located to see if population had an affect.

To further explore if population had any affect on how many hit a player made we decided to make a scatter plot with a line of best fit. When we created the scatter plot it was simple to look at it and see that there was almost no correlation at all. In fact the correlation coefficient was almost exactly zero. From this we were able to conclude that there is not a correlation between the population of a state and the number of hits a player from that state has in a season.

In conclusion, we found reason to believe that the population of a state does affect how many players are in the top 160 players. There are different reasons that this could be. Larger states tend to be able to support an MLB team better than a smaller state thus giving them an opportunity to get a player in the top 160. Also, larger states may also have more than one team. For example California has five MLB teams. This will give them a better chance of having multiple players make the list. Also, the datasets that we used were simple to work with and reliable. If we had to redo the project, we would probably choose to explore something other than if population affects the players or change the state to the state where the players were born rather than where their team is located. For the code next time I think we would attempt to use pandas to merge the data since it is a simple program to use and similar to RStudio.