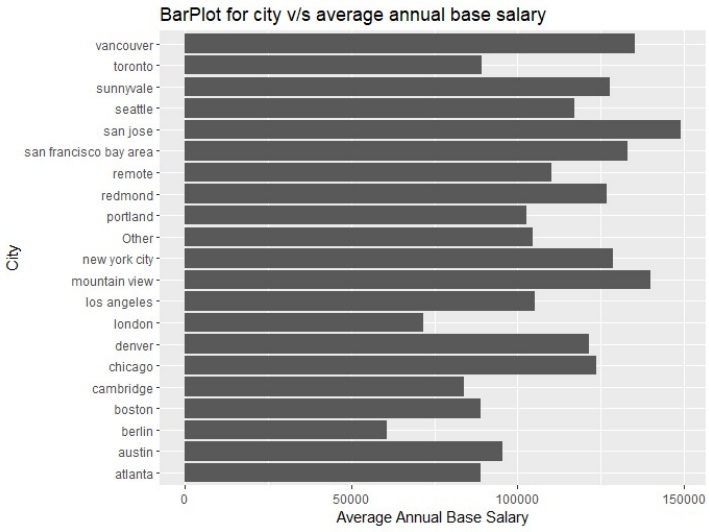
1. Data: This dataset is a dive into the global salary & experience landscape compiled from user survey responses out of 2016 Hacker News. The dataset contains around 1655 responses spread across 19 parameters providing a peek into the pay rates, geographical relevance, job titles and their categories.

Out of the 19 parameters, some notable ones that I've used for my analysis are : -

- location_name: Name of the location which is a character string type variable
- annual_base_pay: Annual base pay of the user which is a numerical type variable.

I've prepared my dataset in following manner: -

1. Firstly, I read the dataset(salaries_clean.csv) into a variable using read_csv function provided along with the tidyverse library in R. While traversing through the dataset, I plotted the data and checked for summary. I noticed there was a lot of improper data in location_name column where-in the city and state were often merged into the location_name column itself.
    a. So, firstly, I used the separate function to split the values in location_name column(on basis of a ",") into 2 columns namely "city" and "state". This way I could be able to use just the city column along with annual_base_pay column for my further analysis.
2. Still, there were quite a huge range of values in annual_base_pay column so I thought of checking the normality of it.
    a. For checking the normality, I used a QQ-plot. Here, I saw a straight horizontal line displayed with a single point far away from all the other data. This is an outlier since its so far away from all the other data points thus we'll have to remove this from our dataset. So, to remove the outlier, I used the 'arrange' command to sort the annual_base_pay column in a descending order and check which is the value causing this behaviour. Thus, the biggest value found was '9999999999'. So, I went through some of the topmost values and used the 'filter' function to only select values below a range of around 2000000, since the general trend of pay was in this range so it could help remove the outlier from the data.
3. Merging cities into 1 category to eliminate duplicate values:
    a. After splitting the location_name column into city, it still had many duplicate or similar looking categories, for example, "san francisco", "sf", "sf bay area", etc. Thus, I merged them into 1 single category per city. I did this same process for 3-4 more cities that I could find from the dataset.
    b. After this, to prepare a plot, I chose some 15-20 city values to show how salary varies amongst them. So to accomplish this, I put all the other categories apart from these 15-20 into a separate category "Other". Plot displayed on the side.



2. Planning and Analysis:

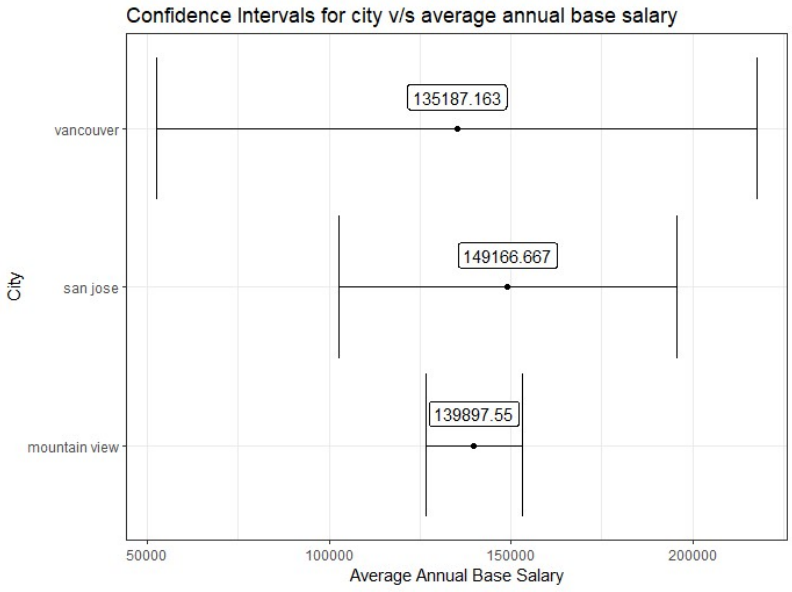
My plan was to develop a plot which could show how much is the average annual base salary of employees varying according to cities. And then select the top 3 cities having the highest salaries and check my hypothesis of whether the sample mean annual base salary would be equal for them or not.

Null Hypothesis (H0): The mean annual base salary for the top 3 cities is equal (i.e. they are drawn from populations with same mean)

Alternate Hypothesis (H1): The mean annual base salary is not equal for the top 3 cities (i.e. its likely that they are drawn from populations with unequal means)

To validate the hypothesis, I found out confidence intervals(C.I.) for the 3 cities and checked if intervals are likely different or not. Then I found out a 95(100-alpha)% confidence interval so the alpha(level of significance)=5% == 0.05. Total observations for top 3 cities= “san jose”=40, “mountain view”=21 & “Vancouver”=12 = 40+21+12=73. Now, I plotted a QQ plot to check for normality, but from the plot I couldn’t confirm if it was Normally distributed or not. Here, we have >=30 data points so as per Central Limit Theorem, we can say that the mean is normally distributed. Thus, I used z-test to calculate our confidence intervals.

For alpha=0.05 -> z-score =1.96 ; C.I. = mean $\pm$ 1.96*(standard error) where standard error = [standard deviation(x)/sqrt(n)]. Then, I used geom_errorbarh() function alongside ggplot() function to plot our confidence intervals on the graph and check our hypothesis validity.



3. Conclusion: As can be seen from the plot, the confidence intervals overlap so we can conclude that we fail to reject the null hypotheses: these sample means are not likely different, and its likely that these sample means are drawn from populations with the same mean.

Dataset Citation: - Telle, B. (2018, June 9). *2016 Hacker News Salary Survey Results*. https://data.world/brandon-telle/2016-hacker-news-salary-survey-results