Stat 250 Final Project

Comparing Salary Outlooks: Software Engineer Vs. Product Manager

Eway Cai, Nina Sun, Jeanny Zhang

March 5, 2022

Introduction

As students who will soon graduate and are interested in the information technology industry, we would like to be more informed about the different job opportunities within the field as well as their career prospects. Learning about financial compensation of different jobs provides an important factor in our decision making about choosing one career path over another, as well as developing related skill sets to turn ourselves into more qualified and competitive applicants in the job market. Therefore, a successful direct comparison of the average yearly compensation of the two positions that we are all interested in, software engineer and product manager, will yield interesting and much-needed observations which can be applied to many students during their junior and senior years.

As the technology and information sciences industry continues its expansion, there has undoubtedly been significant increases in the need of software engineers. Meanwhile, while product managers are not directly involved in technical development, they nonetheless play crucial roles by identifying customer needs and delivering products that best suit their needs to maximize company profits. At Carleton, many students consider software engineer positions to be very high paying jobs, but there have been reports suggesting that product managers may be even better compensated (Coren, 2016). As one of the leading corporations in information technology, Google has over 130,000 employees worldwide according to Wikipedia, so we decided to use the data from Google employees to investigate the long term career outlooks of software engineers and product managers by studying whether there is a difference in mean total yearly compensation between senior-level software engineers and product managers.

Methodology

The original data set comes from a Kaggle user named Jack Ogozally, who scraped the data from levels.fyi and conducted thorough tidying efforts. Levels.fyi is a website that allows users to voluntarily contribute their employment status and compare career levels and compensation across various companies in the IT industry. We believe that this platform provides accurate and abundant data for our analysis of interest. We also acknowledge that there are potential limitations in this data collection method, and we will discuss more about this later.

The original data set contains 62,000 salary records from multiple top IT companies. Columns include job locations, years at the company, years of experience, base salary and more. We decided to extract salaries based on job levels because one's job level is more tightly related to one's salary compared to education level, years of experience, or other factors.

For our analyses, further tidying of the data is performed to include only columns and rows of interest. According to Google's ranking system, senior-level employees have job levels L5 and above. We used the tidyverse package in R to filter out all the data from Google employees with a job level L5 and above. The tidy data set contains the following variables:

Title: This variable includes two types of job title, which are either product manager or software engineer.
Level: Job titles that are higher than level 5 within a company.
Total Yearly Compensation: Total yearly income including base salary, stock and bonus.

Results

Exploratory data analysis is first conducted to observe distributions and general trends of the two data sets. We first examined the summary statistics for two types of job titles. The mean total yearly compensation for product managers is 412331.7 with a standard deviation of 187581.9, and the mean for software engineers is 398549.6 with a standard deviation of 142570.7. There are 857 data points for software engineers and only 199 data points for product managers.

Histograms and a boxplot of total yearly compensation for both product managers and software engineers were plotted. Both of the histograms display unimodal but relatively high right-skewed distributions as shown in Figure 1 and Figure 2. Since the two distributions are similar, it is valid to conduct t-test on the data sets. Based on the boxplot, potential outliers are defined as total yearly compensation that are larger than 880,000 dollars.

Figure 1. Histogram of Total Yearly Compensation (Software Engineer)

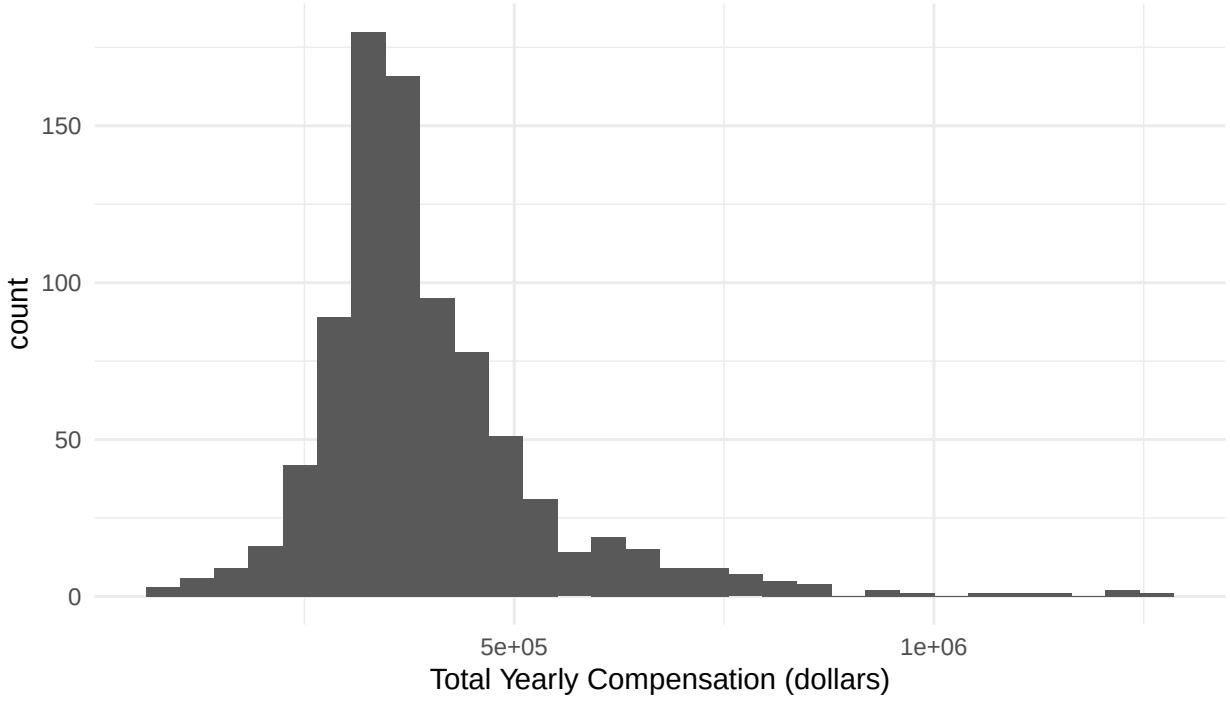
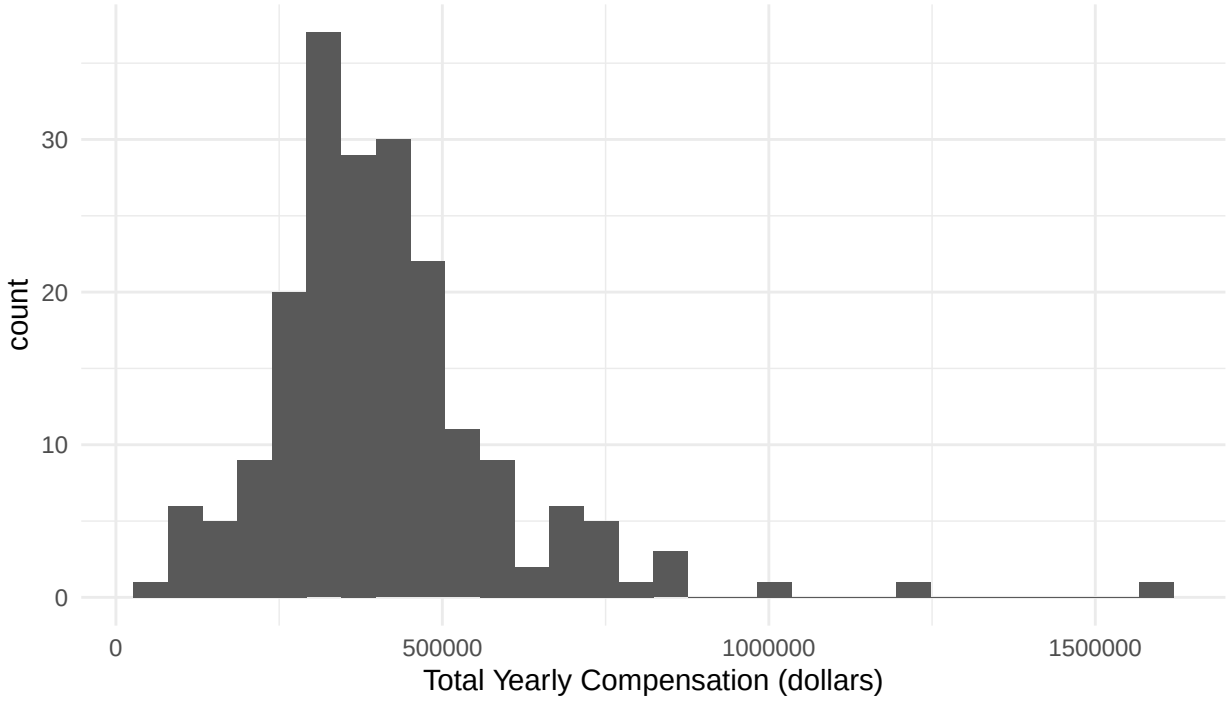
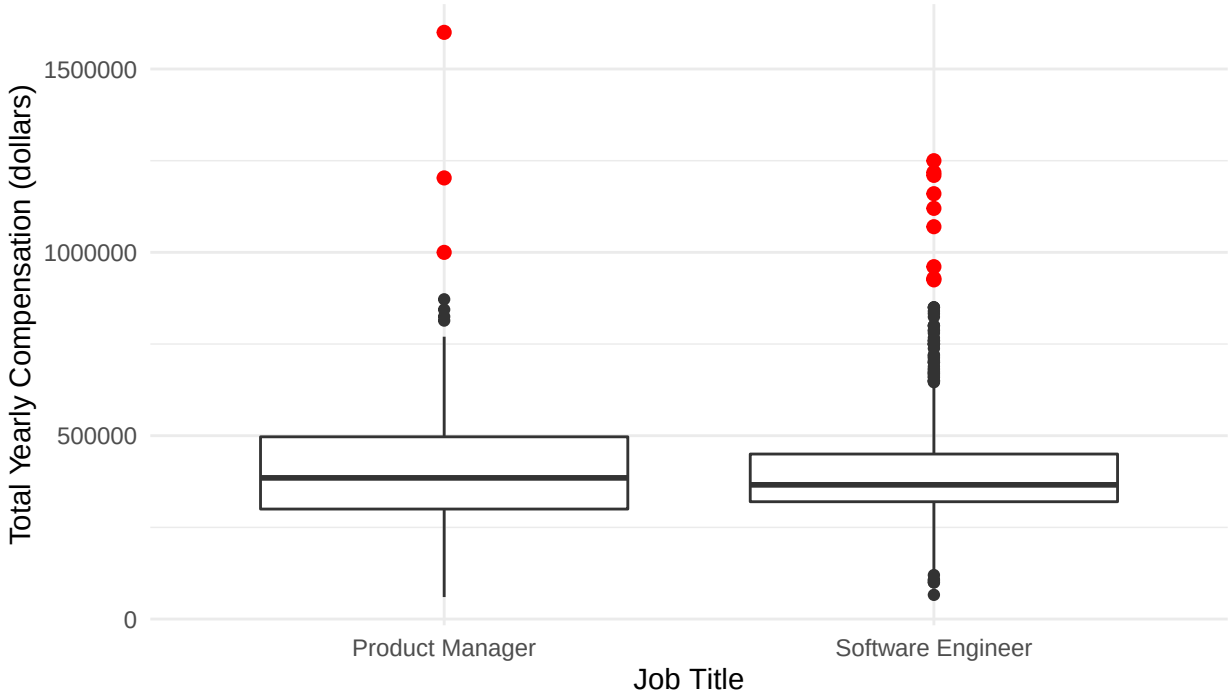


Figure 2. Histogram of Total Yearly Compensation (Product Manager)



The potential outliers are marked as red in Figure 3. The boxplot shows 7 outliers for software engineers and 3 outliers for product managers.

Figure 3. Boxplot of Total Yearly Compensation



The hypothesis test is described as below:

μ_{swe}: the mean yearly total compensations of senior-level software engineers at Google.
μ_{pm}: the mean yearly total compensations of senior-level product managers at Google.

Null hypothesis: $\mu_{swe} = \mu_{pm}$
Alternative hypothesis: $\mu_{swe} \neq \mu_{pm}$

We are interested in using both a confidence interval and p-value to interpret our question of interest. A two-sample bootstrap t-test is initially conducted with the complete data set, without removing any potential outliers. A 5% significance level is deployed to determine statistical significance. The result is a 95% confidence interval of the difference between the mean yearly total compensations of software engineers and product managers ranging from -44,759.67 to 11,588.65 with a p-value of 0.854. Because of the large p-value and the confidence interval including 0, we cannot reject the null hypothesis that there is no significant difference between the mean total yearly compensation of senior level software engineers and product managers.

Next, we are interested in investigating whether potential outliers have an effect in determining the confidence interval. After removing potential outliers, we conducted the two-sample bootstrap t-test again. The resulting 95% confidence interval ranges from -32,003.35 to 14,425.39 with a p-value of 0.759. After removing the potential outliers, the range of the confidence interval decreases, but our conclusion does not change.

Discussion

In this report, we tested whether there is a statistically discernible difference between the average yearly compensation of senior-level software engineers and product managers. Based on the analysis described in the results section, we conclude that there is no difference between the mean total yearly compensations of level 5 or higher software engineers and product managers at Google. However, we cannot infer this to a larger population due to two reasons. One, our sample only contains data from Google. Two, the data set is not randomly sampled since the data are voluntarily contributed by users of Levels.fyi.

There are potential confounding variables such as education level, years of working experience, and location. Work location and time the employees filled out their salary may affect the outcome of our analysis. However, although we do have data on these variables, we have not learned a method to include these variables into consideration. We also propose that some variables will have limited effect in impacting the result of our analysis, as we believe education level and working experience have already been reflected in an employee's level. In addition, there are confounding variables which we might never obtain sufficient data on such as one's relationship with their supervisors which could potentially affect one's compensation and our analysis.

We would also like to address a few limitations in our study. Firstly, the number of data points for software engineers and product managers are significantly different. There are 857 data points for senior level software engineers, while there are only 199 data points for senior level product managers. Therefore a direct comparison between the two salaries does not yield a real-life accurate result.

Secondly, all the data from Levels.fyi are contributed by users voluntarily. As mentioned above, the data is not randomly sampled which could lead to more confounding variables such as one's confidence in their salaries. However, given the confidential nature of the subject of salary, it is difficult to obtain data by random sampling.

Further studies in related topics will provide more comprehensive perspectives into the job market. For example, one good comparison may be to cross examine the mean yearly compensation of software engineers and product managers from all five FAANG companies and see whether this non-significant difference in wages holds. Another interesting study might be to compare the mean yearly compensation between senior-level data engineers and software engineers. Any of these studies will provide great guidance for college

students in determining their future plans.

Appendix

Sources:
https://en.wikipedia.org/wiki/Google
https://qz.com/766658/the-highest-paid-workers-in-silicon-valley-are-not-software-engineers/

```
# EDA
favstats(swe$totalyearlycompensation)
```

```
##    min     Q1 median     Q3     max     mean       sd   n missing
##  66000 320000 366000 450000 1250000 398549.6 142570.7 857       0
```

```
favstats(pm$totalyearlycompensation)
```

```
##    min    Q1 median     Q3     max     mean       sd   n missing
##  60000 3e+05 385000 497000 1600000 412331.7 187581.9 199       0
```

```
# t test
t.test(swe$totalyearlycompensation, pm$totalyearlycompensation, alt="two.sided")
```

```
##
##  Welch Two Sample t-test
##
## data:  swe$totalyearlycompensation and pm$totalyearlycompensation
## t = -0.97323, df = 253.63, p-value = 0.3314
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -41670.41  14106.28
## sample estimates:
## mean of x mean of y
##  398549.6  412331.7
```

```
# remove large potential outliers from both datasets
t.test(swe[-c(swe_outlier_index),]$totalyearlycompensation, pm[-c(pm_outlier_index), ]$totalyearlycompen
```

```
##
##  Welch Two Sample t-test
##
## data:  swe[-c(swe_outlier_index), ]$totalyearlycompensation and pm[-c(pm_outlier_index), ]$totalyearl
## t = -0.68919, df = 256.76, p-value = 0.4913
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -31119.67  14984.42
## sample estimates:
## mean of x mean of y
##  391172.2  399239.8
```

```
# bootstrap t test

# remove outliers
set.seed(233)
xbar <- mean(swe$totalyearlycompensation) - mean(pm$totalyearlycompensation)
```

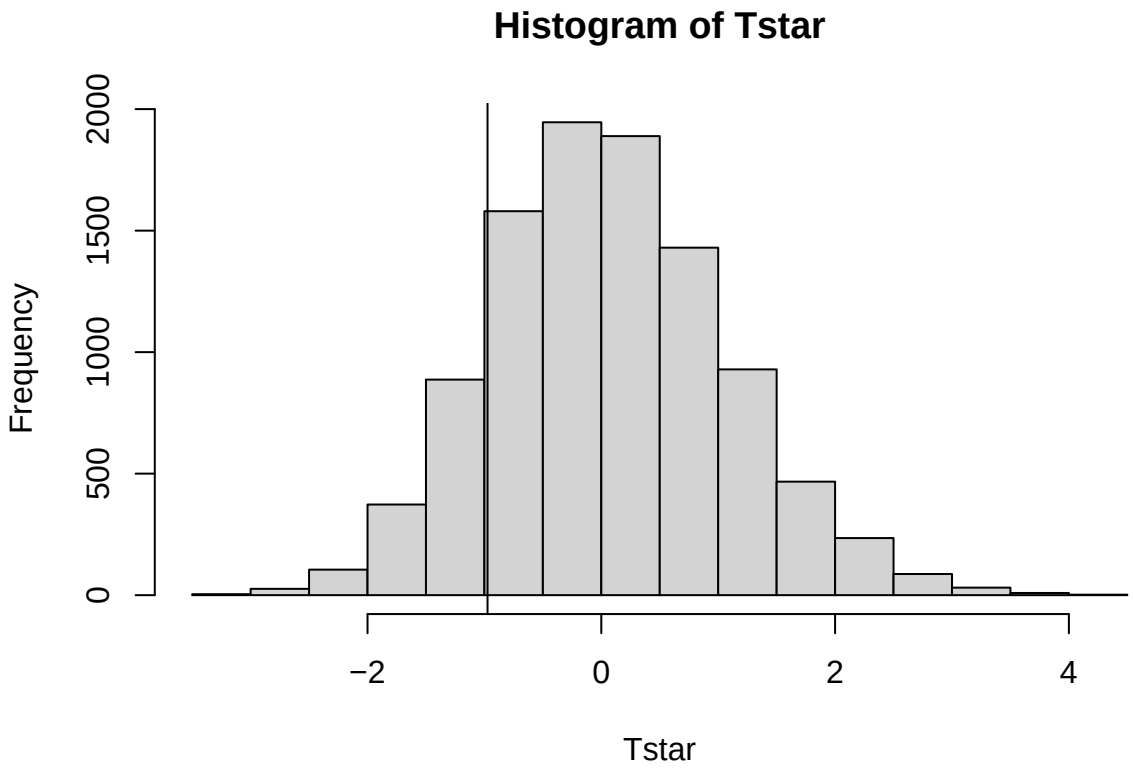
```r
swe_n <- length(swe$totalyearlycompensation)
pm_n <- length(pm$totalyearlycompensation)

B <- 10^4
Tstar <- numeric(B)
# run bootstrap
for (i in 1:B) {
  swe_boot <- sample(swe$totalyearlycompensation, swe_n, replace = TRUE)
  pm_boot <- sample(pm$totalyearlycompensation, pm_n, replace = TRUE)
  Tstar[i] <- (base::mean(swe_boot, na.rm = TRUE) - base::mean(pm_boot, na.rm = TRUE) - xbar) / (sqrt(s
}

# get 2.5th and 97.5th quantiles
q1 <- quantile(Tstar, 0.025)
q2 <- quantile(Tstar, 0.975)

# 95% confidence interval
print(str_c("2.5%: ", xbar - q2*sqrt(stats::sd(swe$totalyearlycompensation, na.rm = TRUE)^2/swe_n + sta
```

```
## [1] "2.5%: -44759.6726928517"
```

```r
print(str_c("97.5%: ", xbar - q1*sqrt(stats::sd(swe$totalyearlycompensation, na.rm = TRUE)^2/swe_n + st
```

```
## [1] "97.5%: 11588.6461407078"
```

```r
# calculate p-value
observedT <- t.test(swe$totalyearlycompensation, pm$totalyearlycompensation, alt="two.sided")$statistic
# plot histogram with observed t
hist(Tstar)
abline(v = observedT)
```



```r
print(str_c("p-value: ", (sum(Tstar >= observedT)+1)/(B+1)))
```

```
## [1] "p-value: 0.854014598540146"
```

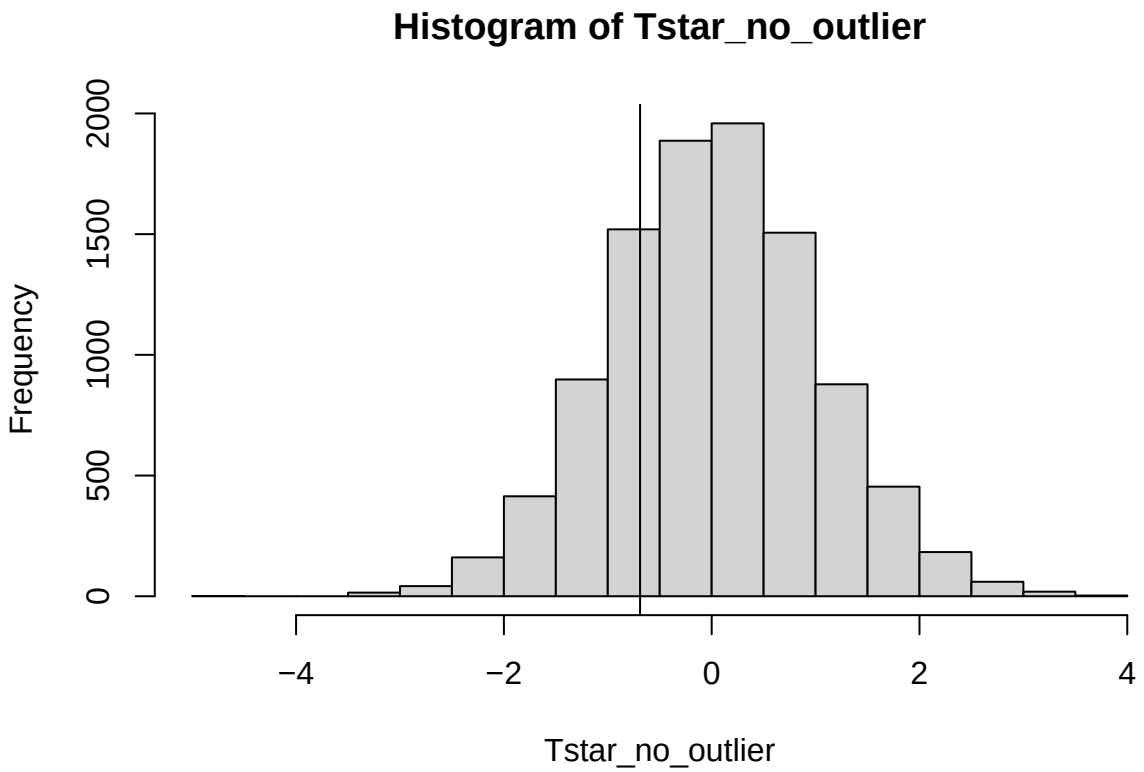
```r
# bootstrap t test without outliers

# remove outliers
swe_no_outliers <- swe[-c(swe_outlier_index),]
pm_no_outliers <- pm[-c(pm_outlier_index),]

set.seed(233)
xbar_no_outlier <- mean(swe_no_outliers$totalyearlycompensation) - mean(pm_no_outliers$totalyearlycompen
swe_n_no_outlier <- length(swe_no_outliers$totalyearlycompensation)
pm_n_no_outlier <- length(pm_no_outliers$totalyearlycompensation)

B <- 10^4
Tstar_no_outlier <- numeric(B)
# run bootstrap
for (i in 1:B) {
  swe_boot <- sample(swe_no_outliers$totalyearlycompensation, swe_n_no_outlier, replace = TRUE)
  pm_boot <- sample(pm_no_outliers$totalyearlycompensation, pm_n_no_outlier, replace = TRUE)
  Tstar_no_outlier[i] <- (base::mean(swe_boot, na.rm = TRUE) - base::mean(pm_boot, na.rm = TRUE) - xbar
}

# get 2.5th and 97.5th quantiles
q1_no_outlier <- quantile(Tstar_no_outlier, 0.025)
q2_no_outlier <- quantile(Tstar_no_outlier, 0.975)

# 95% confidence interval
print(str_c("2.5%: ", xbar_no_outlier - q2_no_outlier*sqrt(stats::sd(swe_no_outliers$totalyearlycompensa
```

```
## [1] "2.5%: -32003.3541209041"
```

```r
print(str_c("97.5%: ", xbar_no_outlier - q1_no_outlier*sqrt(stats::sd(swe_no_outliers$totalyearlycompens
```

```
## [1] "97.5%: 14425.3933176388"
```

```r
# calculate p-value
observedT_no_outlier <- t.test(swe_no_outliers$totalyearlycompensation, pm_no_outliers$totalyearlycompen
# plot histogram with observed t
hist(Tstar_no_outlier)
abline(v = observedT_no_outlier)
```




```
print(str_c("p-value: ", (sum(Tstar_no_outlier >= observedT_no_outlier)+1)/(B+1)))
```

```
## [1] "p-value: 0.759324067593241"
```