~~~
---
title: "Project Proposal"
author: ""
date: "`r Sys.Date()`"
output: openintro::lab_report
---

```{r load-packages, message=FALSE, include=FALSE}
#packages
library(tidyverse)
library(openintro)
library(infer)
library(openintro)
#load data
stem_salaries<-read_csv("/home/_______________/Levels_Fyi_Salary_Data.csv")

```
~~~

What factors contribute to Stem Salaries?

###Introduction

How do you make the most money as a software engineer? In our data analysis, we hope to answer this question through a variety of statistical methods. The data set we will be using comes from the data science website Kaggle. The data is from 60,000 participants from various STEM fields including their background information (race, education level, etc.), yearly earnings, where they work, the company they work for, their job title, and experience levels. This data was scraped from the levels.fyi website where users submit this information to compare earnings and compensation packages. Financial information does get validated by the site through reviews of proof documents (W2, Offer letters, etc.), detection of outliers and forgeries, and identity verification. A majority of users are software engineers so we will be focusing on them and will determine what variables indicate a strong correlation with higher wages.
###Data Analysis Plan

#Variables

For our data analysis, we will be using the total yearly compensation as our dependent variable. This will get faceted among several other predictor variables in our data set. For predictor variables, we will primarily use the employee’s years of experience (numerical), their education level (multilevel categorical predictor), gender (two-level categorical predictor), and the region where they work (multilevel categorical predictor). Using these variables we will try to understand which factors have the greatest impact on the compensation these software engineers earn yearly. It should also be noted that within the title variable we will only be working with a subset including software engineers since they made up close to 66% of the entire data set.

#Possible Issues

There are a couple of potential issues we may run into while trying to make inferences based on the data set. These include both potential issues with how to best represent the data and issues with making our data usable and cleaning it. Issues with how to best represent our data mostly come from the differences between yearly compensation and base salary. For example, when trying to compare yearly compensation and experience, those in their first year might get a large signing bonus that gives them a high yearly compensation but a low base salary. However, base salary may not be representative of those who have earned stocks with the company. Accounting for these discrepancies may prove to be difficult but would help better answer our research question.

Another issue may be problems with making usable data and cleaning it. Some variables such as the `city code` and `dmaid` are not defined within the data set making it unusable. There are also some variables that have somewhat high nonresponse rates such as race and education. However, we do have a large dataset so this may not be such a large issue for us since we should still end up with a reasonable number of respondents.

Preliminary data exploration

##Stat methods

In order to make conclusions on our data, we will use a variety of statistical methods. We will use some basic statistics such as mean, median, and interquartile range. These will give us simple yet descriptive information about some of our main variables to better understand and infer their effects on other variables. We will also create a multi-variable linear regression that will predict the total yearly compensation of software engineers based on several variables. This linear regression will help us draw conclusions on how strong of indicators specific variables are. We will also create a bootstrap that specifically looks at the difference in means

##Exploratory Data Analysis (Visualizations and Summary Statistics)

~~~~
```{r code-chunk-1}
stem_salaries <- stem_salaries %>%
  mutate(log_total_yearly_compensation = log(totalyearlycompensation))

ggplot(data=stem_salaries, mapping=aes(x=log_total_yearly_compensation)) +
  geom_histogram(binwidth=0.2) +
  labs(title="Distribution of Log Transformed Total Yearly Compensation", subtitle="Salaries of Workers in STEM", x="Log Transformed Total Yearly Compensation")
~~~~

~~~~
stem_salaries %>% summarize(mean_log_tyc=mean(log_total_yearly_compensation),
sd_log_tyc=sd(log_total_yearly_compensation))
```

```{r code-chunk-2}
stem_salaries <- stem_salaries %>%
  mutate(basesalary = na_if(basesalary, 0))

stem_salaries <- stem_salaries %>%
  mutate(log_base_salary = log10(basesalary))

ggplot(data=stem_salaries %>% filter(!is.na(log_base_salary)),
mapping=aes(x=log_base_salary)) +
  geom_histogram(binwidth=0.1) +
  labs(title="Distribution of Log Transformed Base Salary", subtitle="Salaries of Workers in
STEM", x="Log Transformed Base Salary")

stem_salaries %>%
  filter(!is.na(log_base_salary)) %>%
  summarize(median_log_bs=median(log_base_salary),
            iqr_log_bs=IQR(log_base_salary))
```

```{r code-chunk-3}
ggplot(data=stem_salaries, mapping=aes(x=yearsofexperience,
y=log_total_yearly_compensation)) +
  geom_point() +
  xlim(0, 50) +
  geom_smooth(method='lm', size=2) +
  labs(title="Log Transformed Total Yearly Compensation vs. Years of Experience",
subtitle="Salaries of Workers in STEM", x="Years of Experience (in years)", y="Log Transformed
Total Yearly Compensation")
```

```{r code-chunk-4}
ggplot(data=stem_salaries %>% filter(!is.na(log_base_salary)),
mapping=aes(x=yearsofexperience, y=log_base_salary)) +
  geom_point() +
  xlim(0, 50) +
  geom_smooth(method='lm', ) +
  labs(title="Log Transformed Base Salary vs. Years of Experience", subtitle="Salaries of
Workers in STEM", x="Years of Experience (in years)", y="Log Transformed Base Salary")
```
~~~~

```{r code-chunk-5}
new_df <- stem_salaries %>%
  filter(!is.na(totalyearlycompensation), !is.na(gender)) %>%
           filter(gender %in% c("Male", "Female")) %>%
           mutate(log_yearly_comp = log(totalyearlycompensation))

ggplot( data = new_df  %>% filter(!is.na(log_yearly_comp), !is.na(gender)),
      mapping = aes(y = log_yearly_comp, x = gender,
                    fill = gender) ) +
  geom_boxplot() +
  labs(
    title = "Total yearly compensation by gender",
    y = "Log of Yearly Salary (in dollars)",
    x = "Gender"
  )
```

What we can learn from this boxplot is that gender appears to make little difference in the outcome of total yearly compensation. In the graph, the median salary for both males and females is about $120,000. Additionally, the spread of the data also seems to be similar as the interquartile range appears to be the same size with the exception that the make range is slightly larger. One difference that was observed is the spread of the outlet data. Males tend to have more outliers that have a higher yearly compensation than women. So far, this helps us learn that gender may not be a strong indicator of yearly compensation. However, the difference in outliers could be further explored to understand why this occurs.

###Statistical Methods

In order to make conclusions on our data, we will use a variety of statistical methods. We will use some basic statistics such as mean, median, and interquartile range. These will give us simple yet descriptive information about some of our main variables to help us better understand and infer their effects on other variables. Referring to the histograms we’ve made, the summary statistics give us a better idea of the typical values within the sample and can be a benchmark when we later calculate other values in different categories. In a variable such as the education level of an individual, we expect to find a relationship where those with higher education, such as a master's degree or doctorate, have higher mean and median values for their total yearly compensation than those with a bachelor's degree or no college experience. We would also expect a lower IQR for those with higher education since they might get higher paying offers more consistently, causing not much variation. Completing exploratory data analysis, it sets up a foundation for us to continue with discovering associations (linear regression), making statements about a population (bootstrapping confidence intervals) and then making claims that will answer our research question (hypothesis testing).

We will also create a multi-variable linear regression that will predict the total yearly compensation and a base salary of software engineers based on several variables. This linear regression will help us draw conclusions on how strong the relationship between predictors and response variables are through lines of best fit using the least squares method and calculating the correlation coefficient. The lines of best fit will give us the linear equations that allow us to make specific statements on not only the strength of the associations but also whether it is positive or negative. In our linear regression, we expect to see several variables affect the total yearly compensation and base salary. For the location of the individual, we expect to see individuals that work from large cities have a larger impact on wages than those from smaller cities as represented by a larger y-intercept in the linear regression. We expect tech companies to be more generous with the compensation which can be indicated by the slope of the linear equation than non-tech companies. We expect the years of experience to have some impact on the linear regression which will be shown by a large multiplier on the experience variable in the regression.

We will also use bootstrapping that looks at the slope for two different categories of a variable. After bootstrapping, we can create confidence intervals. This will be a good method to learn more about the plausible values for the population depending on the relationship of a predictor and response variable. We will also use hypothesis testing to test some of our hypotheses we will have once we have completed the previous statistical methods and learned more in depth about how the predictor and response variables affect each other. Hypothesis testing will allow us to make a claim about a relationship between a predictor and response variable that will help us answer the question about the factors that allow an individual to become successful as a software engineer.

Other estimation methods to examine conditional relationships would be facetting. For instance, to learn more about salary and years of experience at a company, we can facet by each company to visualize the variance across companies. To gain insights on our categorical variables, we could also use contingency tables that cross-tabulate frequencies between race and education, or gender and education. This would help us understand which demographics are more or less educated. Other relationships we'd also like to test are between yearly salary and years of experience. If we find a predicted salary based on years of experience, calculating the residuals and plotting this would help us understand how much of our data can be explained by this relationship and understand how far each data point is from the predicted value.

###Data

The data set we choose includes 29 variables with 62,324 observations. This has been subseted to include 31 variables with 14,755 observations.

Header	Description

| `timestamp` | When the data was recorded (**Month**:**Day***:*Year**:**Hour**:**Minute**:**Second**)
| `company` | technology company name
| `level` | What level the observation is at
| `title` |Job Title
| `totalyearlycompensation` | Total yearly compensation (in dollars)
| `location` | job location (City, State)
| `yearsofexperience` | Years of Experience
|`yearsatcompany` | years at a company
| `tag` | Job tag
|`basesalary`| base salary (in dollars)
|`stockgrantvalue`|stock grant value (in dollars)
|`bonus`|bonus (in dollars)
|`gender`| gender (**Male**, **Female**, **NA**)
|`otherdetails`| Additional information which includes clauses for certain benefits
|`cityid`| cityid
|`dmaid`| dmaid
|`rownumber`| row number
|`Masters_Degree`| Masters Degree completed in 1 or 2 years. Denoted as: **1** Yes, **0** for No
|`Bachlors_Degree`| Bachelors Degree (4 years) completed at an accredited University or junior college: **1** Yes, **0** for No
|`Doctorate_Degree`| Doctorate Degree: **1** Yes, **0** for No
|`Highschool`| Completed the High school education Requirements and has earned a high school diploma: **1** Yes, **0** for No
|`Some_College`| Did not earn a bachelors, under 4 years of college or currently in college: **1** Yes, **0** for No
|`Race_Asian`| Asian People **1** Yes, **0** for No
|`Race_White`| White People **1** Yes, **0** for No
|`Race_Two_Or_More`| People with two or more races. **1** Yes, **0** for No
|`Race_Hispanic`| Hispanic People: **1** Yes, **0** for No
|`Race`| Racial Ethnicity (**Asian**, **White**, **Hispanic**, **Black**, **Two or More**)
|`Education`| Education Level (**Bachelors**, **Masters**, **PhD**, **PhD + Masters**, **SomeCollege**, **Highschool**)
|`major_tech_company`| Do they work for a major tech cooperation? (Inc: Apple, Microsoft, Oracle, Amazon, Google, Cisco or Facebook) **yes**, **no**.
|`high_income_state`| Do they work in a high income state? (Inc: WA, CA, NY, MA, MD, and VA) **yes**, **no**.

~~~
```{r data cleaning, echo=FALSE}
#Removes data that's not from the US through str_detect function
stem_salaries<- stem_salaries %>%
        mutate( US =
      ifelse( str_detect(location,

"AL|AK|AZ|AR|CA|CO|CT|DE|DC|FL|GA|HI|ID|IL|IN|IA|KS|KY|LA|ME|MD|MA|MI|MN|MS|MO|MT
|NE|NV|NH|NJ|NY|NM|NC|ND|OH|OK|OR|PA|RI|SC|SD|TN|TX|UT|VT|VA|VI|WA|WV|WI|WY"),
"yes", "no") )

#Removes some values that got through the initial location filter (e.g. Chennai, TN, India)
stem_salaries<- stem_salaries %>%
        mutate( nonUS =
      ifelse( str_detect(location,
"India|Brazil|Belarus|Colombia|Spain|Latvia|France|Poland|China|Netherlands|United
Kingdom|Australia|Austria|Netherlands|Hong Kong|Denmark"), "yes", "no") )

#Creates Data Subset that only uses data from US workers, Software Engineers/Software
Engineering Manager, Removes N/A values for education and gender.
software_salaries <- stem_salaries %>%
  filter(title == "Software Engineer" | title =="Software Engineering Manager") %>%
filter(US=="yes") %>% filter(nonUS=="no") %>% filter(!is.na(Education)) %>%
filter(!is.na(gender))

#Fixes Education Column so observations with PhD and Masters are denoted and not counted
as masters -- Also helps see which other variables have multiple columns filled.
software_salaries <- software_salaries %>%
  mutate(Education =
Masters_Degree+10*Doctorate_Degree+100*Bachelors_Degree+1000*Some_College+10000*
Highschool) %>%
  mutate(Education = case_when(
    Education ==     1 ~ "Masters",
    Education ==    10 ~ "PhD",
    Education ==    11 ~ "PhD + Masters",
    Education ==   100 ~ "Bachelors",
    Education ==   101 ~ "Masters",  #To get Masters you need Bachelor
    Education ==  1000 ~ "SomeCollege",
    Education == 10000 ~ "Highschool") )
~~~

~~~
```

```{r data prep, echo=FALSE}
#Simplifies location data to either a major tech state or a non-major tech state
software_salaries <- software_salaries %>%
  mutate( high_income_state = ifelse( str_detect(location,"WA|CA|NY|MA|MD|VA" ), "yes", "no")
)
#Adds the log of totalyearlycompensation since data set covers multiple magnitudes of order
software_salaries <- software_salaries %>% mutate(
  log_compensation = log(totalyearlycompensation, base = 10))
#Filters the company data to isolate just large tech companies. Y/N variable with being a large
tech company as the basline.
software_salaries <- software_salaries %>%
          mutate( major_tech_company =
        ifelse( str_detect(company,
"Apple|apple|Microsoft|microsoft|Facebook|facebook|Oracle|oracle|Cisco|cisco|Amazon|amazon
|Google|google"), "yes", "no") )
```
~~~

Where’s the Money at?: The Factors that Influence the Compensation of Software Engineers

Introduction

How do you make the most money as a software engineer? In this report, we hope to answer this question using a variety of statistical tools. One group that may find this data analysis report useful is people seeking to find employment at STEM companies. From our analysis, people may understand from viewing this report that certain factors such as years of experience or education level may have more of an impact on their earnings than others. Other groups that might be interested are the companies themselves since they can learn about competing company compensation offers and make their hiring incentives more competitive to attract qualified applicants.

###Data

This dataset is from a website called Kaggle, which is an open source platform that allows users to upload datasets. This dataset was uploaded from a website called level.fyi which consists of around 60,000 participants from various STEM fields including their background information (race, education level, etc.), yearly earnings, where they work, the company they work for, their job title, and experience levels. The levels.fyi website obtained this data from users who submit this information to compare earnings and compensation packages among various companies. Financial information also gets validated by the site through reviews of proof documents (W2, Offer letters, etc.), detection of outliers and forgeries, and identity verification.

A majority of users are software engineers so we will be focusing on them and will determine what variables indicate a strong correlation with higher wages. Furthermore, not all 60,000 participants had fully complete data in all of the variables that we wanted to use such as education level and gender. We also only included software engineers from the US since it was a large majority of the data and observations from other countries might provide different results.

The data set we choose includes 29 variables with 62,324 observations. This has been subsetted to include 31 variables with 14,755 observations.

Header	Description
`timestamp`	When the data was recorded (**Month**:**Day***:*Year**:**Hour**:**Minute**:**Second**)
`company`	technology company name
`level`	What level the observation is at
`title`	Job Title
`totalyearlycompensation`	Total yearly compensation (in dollars)

| `location` | job location (City, State)
| `yearsofexperience` | Years of Experience
|`yearsatcompany` | years at a company
| `tag` | Job tag
|`basesalary`| base salary (in dollars)
|`stockgrantvalue`|stock grant value (in dollars)
|`bonus`|bonus (in dollars)
|`gender`| gender (**Male**, **Female**, **NA**)
|`otherdetails`| Additional information which includes clauses for certain benefits
|`cityid`| cityid
|`dmaid`| dmaid
|`rownumber`| row number
|`Masters_Degree`| Masters Degree completed in 1 or 2 years. Denoted as: **1** Yes, **0** for No
|`Bachlors_Degree`| Bachelors Degree (4 years) completed at an accredited University or junior college: **1** Yes, **0** for No
|`Doctorate_Degree`| Doctorate Degree: **1** Yes, **0** for No
|`Highschool`| Completed the High school education Requirements and has earned a high school diploma: **1** Yes, **0** for No
|`Some_College`| Did not earn a bachelors, under 4 years of college or currently in college: **1** Yes, **0** for No
|`Race_Asian`| Asian People **1** Yes, **0** for No
|`Race_White`| White People **1** Yes, **0** for No
|`Race_Two_Or_More`| People with two or more races. **1** Yes, **0** for No
|`Race_Hispanic`| Hispanic People: **1** Yes, **0** for No
|`Race`| Racial Ethnicity (**Asian**, **White**, **Hispanic**, **Black**, **Two or More**)
|`Education`| Education Level (**Bachelors**, **Masters**, **PhD**, **PhD + Masters**, **SomeCollege**, **Highschool**)
|`major_tech_company`| Do they work for a major tech cooperation? (Inc: Apple, Microsoft, Oracle, Amazon, Google, Cisco or Facebook) **yes**, **no**.
|`high_income_state`| Do they work in a high income state? (Inc: WA, CA, NY, MA, MD, and VA) **yes**, **no**.

~~~
```{r display, echo=FALSE}
glimpse(software_salaries)

```
~~~

## Modeling Compensation

If you're trying to make the most money as a software engineer, where should you work? What companies should you work for? What level of education should you pursue? Using a linear regression we can create a model that can determine the strength that predictor variables have on a software engineer's yearly compensation. By splitting our data up we can create our model from one portion of the data and test it on the other portion to see how well it performs and determine the best model for US software engineers.

### What should go in the model?

With many variables to work with, we have to determine which variables might predict the yearly compensation and will therefore be fit for our model. We can accomplish this with some visualizations along with some accompanying statistical tools.

Some of the variables that we might expect to predict yearly compensation include:

`yearsofexperience`, `Education`, `company` ⇒ `major_tech_company`, `location` ⇒ `high_income_state`, `yearsatcompany` and `Gender`

~~~
```{r yearsofexp, echo=FALSE, message=FALSE}
ggplot(data=software_salaries, mapping=aes(x=yearsofexperience , y=log_compensation)) +
  geom_point() +
  geom_smooth(method='lm', size=2) +
  labs(title="Log Transformed Total Yearly Compensation vs. Years of Experience",
subtitle="Salaries of Software Engineers and Software Engineering Managers", x="Years of
Experience (in years)", y="Log Transformed Total Yearly Compensation (in dollars)") +
  theme_bw()
```
~~~

From this graph, we can already conclude that higher `yearsofexperience` is a predictor of higher yearly compensation. However, there is also `yearsatcompany` which is a closely related variable. An increase in a year at a company would also increase a year of experience making these variables very colinear, so we will only select one of these two. To determine which one is a stronger predictor we can create a pairwise scatterplot that looks at how all of these variables relate to one another and compare their correlation values.

~~~
```{r data, echo=FALSE}
software_salaries %>%
  select(log_compensation, yearsatcompany, yearsofexperience) %>%
  pairs()

software_salaries %>%
  select(log_compensation, yearsatcompany, yearsofexperience) %>%
  cor() %>% round(3)
```
~~~

The pairwise function and correlation values demonstrate the collinearity between `yearsofexperience` and `yearsatcompany` and why we shouldn't have both in the regression. It also shows a stronger correlation between experience than years at a company for logarithmic compensation value. For our model, we will use `yearsofexperience` over `yearsatcompany`.

We also want to know how working at different companies effect compensation. We will look at the `major_tech_company` variable that interprets the `company` variable and is denoted as "yes" for software engineers from Amazon, Apple, Cisco, Facebook, Google, Microsoft, and Oracle.

~~~
```{r boxplot4, echo=FALSE}
ggplot(data = software_salaries, mapping = aes(x = log_compensation, y =
major_tech_company)) +
geom_boxplot() +
labs(x = "Log of Total Yearly Compensation",
y = "Major Tech Company")

software_salaries %>% group_by(major_tech_company) %>%
  summarize(Median = median(totalyearlycompensation), IQR = IQR(totalyearlycompensation))

```
~~~

There is a relatively large difference in median values between the two variables making the `major_tech_company` a strong candidate variable for the linear regression.

~~~
```{r code-chunk-6, echo=FALSE}
ggplot(data=software_salaries,
      mapping = aes(y = log_compensation, x = gender, fill = gender)) +
  geom_boxplot() +
  labs(
    title = "Log Transformed Total Yearly Compensation of Software Engineers",
    y = "Log of Total Yearly Compensation (in dollars)",
    x = "Gender",
    subtitle="By Gender (Female and Male) (Including Software Engineer Managers)",
    fill="Gender"
  ) +
  theme_bw()

software_salaries %>% group_by(gender) %>% summarize(Median =
median(totalyearlycompensation))
```
~~~

Looking at the `gender` variable, there is a higher median value for males. This is not a huge difference so we will see how it impacts the linear regression.

Lastly, we want to look at the difference in compensation between those in high income states (WA, CA, NY, MA, MD, and VA) and those who are not.

~~~
```{r boxplot3, echo=FALSE}
ggplot(data=software_salaries, mapping = aes(x=log_compensation, y = high_income_state))+
  geom_boxplot()+
  labs(x = "Log of Total Yearly Compensation",
       y = "High Income State")
software_salaries %>% group_by(high_income_state) %>%
  summarize(Median = median(totalyearlycompensation), IQR = IQR(totalyearlycompensation))
```
~~~

We can see that there is a very large difference in median yearly compensation based on being in a high income state. That makes this variable a good fit for our model.

### Training the Models

~~~
```{r data-split, echo=FALSE}
#Sets random seed for data to get split
set.seed(103101)

ss_split <- initial_split(software_salaries, prop = 0.80)

ss_train <- training(ss_split)
ss_test  <- testing(ss_split)
```
~~~

From the data analysis, we now have a good idea of what some of the stronger linear regressions might look like. To test the models, the data was separated so that we can train our models on 80% of the data and test how well the model works on the 20% left.

### Proposed Linear Regression 1

For the first linear regression that will get tested, it will be a main effects model that includes all of the variables that might have or did show a relationship with yearly compensation from our analysis.

~~~
```{r reg1, echo=FALSE}
~~~

```
software_fit1 <- lm(log_compensation ~ yearsofexperience + Education +
major_tech_company + high_income_state + gender, data = ss_train)

tidy(software_fit1) %>% select(term, estimate)
```

We end up with a linear regression represented by the equation:
$$\text{log_compensation} = 5.00 + 0.162 × \text{yearsofexperience} + 0.0024 × \text{EducationHighschool} + \\ 0.0235 × \text{EducationMasters} + 0.144 × \text{EducationPhD} +
0.143 × \text{EducationPhD & Masters} + \\ -0.0074 × \text{EducationSomeCollege} + 0.070 × \text{major_tech_company_yes} + \\0.179 × \text{high_income_stateyes} + 0.0186 × \text{genderMale} + 0.0092 × \text{genderOther}$$

You can plug in values for each of the variables in the equation to create a predicted `log_compensation`. Higher multipliers indicate a greater change in the predicted `log_compensation`. The intercept shows what the value is when all variables are at their baseline (Bachelor's degree, 0 years of experience, female, not in a major tech company, and not in a high-income state). We can interpret several things from this equation. First, many of the variables that we predicted might have an impact on compensation did show a relationship. We can also see some of the variables that might be strong predictors for a higher compensation such as getting a PhD or working in a high-income state.

```{r rsquared1, echo=FALSE}
glance(software_fit1) %>% select(adj.r.squared)
```

But before drawing too many conclusions from this model, we have to analyze how well it actually performs. It has an adjusted R-squared value of 0.45. This is a relatively low value since a value of 1 indicates a line that is perfectly representative of the data. However, we have a highly variable data set so we will get more value from comparing the adjusted R-squared between linear regressions.

```{r testing 1, echo=FALSE, }
suppressMessages(compensation_pred1 <-predict(software_fit1, newdata=ss_test)  %>%
  bind_cols(ss_test %>% select(log_compensation) ) %>%
  rename(pred = ...1))

rmse(compensation_pred1, truth = log_compensation , estimate = pred)  %>% select(.metric,
.estimate)
```

```{r math 1, include=FALSE}
10^0.158
10^5

```

The best way to test the performance of the model is to see how it performs on the 20% of the data set that we put aside to test on. From this testing, we end up with a root-mean-squared error (RMSE). This gives us a standardized error between the value we predict and the actual value, or the residual value. We obtained an RMSE value of 0.158 for this regression. To interpret this we can convert our logarithmic estimates back to their normal values to see this value in terms of money:
$$10^{0.158}=1.44\\10^{5.00}=100,000$$

If we take the example of our baseline value for `log_compensation` which was 5.00, we can convert the logarithmic value back to the dollar amount and get 100,000 dollars. Converting our RMSE value we get 1.44. This value is like a multiplier for the compensation value. So we would end up with 144,000 dollars or a standard error of 44,000 dollars for the baseline value. While this isn't highly accurate, like with the adjusted R-squared, we will get more value from comparing this value with the other models.

### Proposed Linear Regression 2

The second linear regression we will look at will maintain the same variables that were used in the first regression, but instead of using a main effects model, it will use an interaction effects model. Therefore, instead of affecting just the intercept, a variable will also affect the slope. In this case, we chose `high_income_state` to be our interaction effect. This was because where you live might increase wages by a percentage rather than a fixed dollar amount.

```{r reg2, echo=FALSE}
software_fit2 <- lm(log_compensation ~ (yearsofexperience + Education +
major_tech_company + gender) * high_income_state , data = ss_train)

tidy(software_fit2) %>% select(term, estimate)
```

We end up with the equation:

$$\text{log_compensation} = 5.03 + 0.015 × \text{yearsofexperience} + -0.029 ×
\text{EducationHighschool} + \\ 0.020 × \text{EducationMasters} + 0.127 × \text{EducationPhD}
+
0.205 × \text{EducationPhD & Masters} + \\ 0.011 × \text{EducationSomeCollege} + 0.160 ×
\text{major_tech_company_yes} + \\-0.011 × \text{genderMale}+ -0.042 × \text{genderOther} +
0.152 × \text{high_income_stateyes} $$

This looks very similar to the equation from the first linear regression. However at the end of this equation there is this:

$$(0.0015 × \text{yearsofexperience} + 0.0475 × \text{EducationHighschool} + \\0.0043 ×
\text{EducationMasters} + 0.0192 × \text{EducationPhD} + \\
-0.0680 × \text{EducationPhD & Masters} + -0.0260 × \text{EducationSomeCollege} + \\-0.1005
× \text{major_tech_company_yes} + -0.0351 × \text{genderMale} + \\0.0633 ×
\text{genderOther}) × \text{high_income_stateyes}$$

This part of the equation creates a multiplier based on if the observation is in a high-income state. It works by multiplying the sum of the variables, each of which has constants on them to account for their strength of prediction. The product of this is variable slopes based on `high_income_state`.

~~~
```{r rsquared2, echo=FALSE}
glance(software_fit2) %>% select(adj.r.squared)
```
~~~

The performance in adjusted R-squared is the same as the first linear regression at 0.45.

~~~
```{r testing 2, echo=FALSE}
suppressMessages(compensation_pred2 <-predict(software_fit2, newdata=ss_test)  %>%
  bind_cols(ss_test %>% select(log_compensation) ) %>%
  rename(pred = ...1))

rmse(compensation_pred2, truth = log_compensation , estimate = pred)  %>% select(.metric,
.estimate)
```
~~~

When we use this model on the testing set we end up with an RMSE value of 0.157. The value from the first linear regression was 0.158 so the differences in performance between the two regressions are negligable.

### Proposed Linear Regression 3

The final regression we will look at will be similar to the first one where it is a main effects model. However, since we could not conclusively decide if having the `gender` variable would strengthen the model, it will be omitted so we can compare the models.

~~~
```{r reg3, echo=FALSE}
software_fit3 <- lm(log_compensation ~ yearsofexperience + Education +
major_tech_company + high_income_state , data = ss_train)

tidy(software_fit3) %>% select(term, estimate)
```
~~~

The equation from this regression is:

$$\text{log_compensation} = 5.02 + 0.0164 × \text{yearsofexperience} + 0.0042 ×
\text{EducationHighschool} + \\ 0.0233 × \text{EducationMasters} + 0.144 × \text{EducationPhD}
+
0.143 × \text{EducationPhD & Masters} + \\ -0.0067 × \text{EducationSomeCollege} + 0.070 ×
\text{major_tech_company_yes} + \\0.179 × \text{high_income_stateyes}$$

The equation is very similar to the first one with most of the constants changing by only around a thousandth if at all.

```{r rsquared, echo=FALSE}
glance(software_fit3) %>% select(adj.r.squared)
```

Compared to the first regression we also get the same adjusted R-squared value of 0.45.

```{r testing 3, echo=FALSE}
suppressMessages(compensation_pred3 <-predict(software_fit3, newdata=ss_test)  %>%
  bind_cols(ss_test %>% select(log_compensation) ) %>%
  rename(pred = ...1))

rmse(compensation_pred3, truth = log_compensation , estimate = pred)  %>% select(.metric,
.estimate)
```

When we test the model we get an RMSE value of 0.158 which is also the same as the first regression.

### Selecting the Best One

With three linear regressions that all had nearly identical performances, how do you select the best one? There is a principle called Occam's Razor which states that the simplest solution is often the correct solution. Applying this to the regressions we can see that our third regression is the simplest. The first one had the addition of the `gender` variable that did not seem to impact the accuracy of our predictions. The second one used the idea of an interaction model that makes the solution much more complex and unlikely to be the best model not unless it can perform much better. Therefore, the best model we could create to predict `log_compensation` is represented by the model:

$$\text{log_compensation} = 5.02 + 0.0164 × \text{yearsofexperience} + 0.0042 ×
\text{EducationHighschool} + \\ 0.0233 × \text{EducationMasters} + 0.144 × \text{EducationPhD}
+
0.143 × \text{EducationPhD & Masters} + \\ -0.0067 × \text{EducationSomeCollege} + 0.070 ×
\text{major_tech_company_yes} + \\0.179 × \text{high_income_stateyes}$$

Using this equation we can get an idea of how you might make the most money as a software engineer. First, we can see that time is on your side. `yearsofexperience` increases your compensation by quite a bit. For every year of experience, there is a predicted increase in `log_compensation` by 0.0164. Interpreting this, we can look at what 10 years of experience does to `log_compensation`:
```{r math 2, include=FALSE}
10^(10*0.0164)
```
$$10^{10×0.0164}=1.46$$
10 years of experience predicts a 1.46 fold increase in `totalyearlycompensation` versus 0 years of experience.
```{r math 3, include=FALSE}
10^(0.144)
10^(0.023)
stem_salaries %>% count(Education)
```
Secondly, for education, there are two standout levels. Those with both a PhD and master's degree along with those with just a PhD are predicted to have a compensation around 1.4 times larger than those with a bachelor's. A master's degree didn't prove to be as influential only improving compensation over a bachelor's by about 5%. As for those with some college education or only high school education, they make a small portion of the data set. The ability to get a job as a software engineer at that education level is not represented in the model. Both levels had nearly no predicted deviation in compensation from a bachelor's degree represented by their very small constants.
```{r math 4, include=FALSE}
10^(0.070)
10^(0.179)
```

Next, being in a major tech company proved to be somewhat lucrative. Belonging in one, our model predicts an increase in `log_compensation` by 0.070 or a 17% increase over those who are not in a major tech company.

Finally, our model predicts working in a high-income state increases yearly compensation. It was the strongest indicator of the categorical variables with an increase in `log_compensation` of 0.179 or a 51% increase over those living in other states.

## Confidence Interval and Bootstrap Distribution

Given that our main linear regression model was determined to be our best fit, this section tests the confidence interval of the difference in median total yearly compensation among the various explanatory variables as graphed in our linear regression. This includes, education, years of experience, gender, major tech companies, and high income states.

### Confidence Interval 1: median difference in total yearly compensation

When viewing our data set of software engineers and managers, we observed that they had a variety of education levels which includes people who had completed up to high school, some college, a Bachelors degree, a Masters degree, or a Doctorate. Collectively, the proportion of software engineers and managers with Bachelors and Masters made up 91% of our data set. Due to the high proportion of individuals with these degrees, our interests were focused on those education levels. As our overarching research question is to determine what factors influence salaries, we wanted to test whether education level made any significant difference in the outcome of compensation.

Within our findings, we have found the median total yearly compensation for software engineers with a Masters degree and Bachelors is 205,000 and 180,000 respectively. What we learn from this is that software engineers and managers earn significantly more with a masters degree by a difference of $25,0000. If this difference is true, we can conclude that education does in part, determine STEM salaries.

But, to make certain of this difference, we use bootstrapping to estimate the typical distance of sample estimates from this population value to gain confidence in this finding of the median difference. So, to first address this I re-coded the Education column to categorize individuals with Bachelors, Masters, and other levels of education. Next, I filtered out the other education levels such that I could make a binary comparison between individuals with a Masters and a Bachelors when I made the bootstrap distribution. Then, I set the seed and specified my response and explanatory variables and generated samples. This was achieved using bootstrap and calculating the difference in medians. A visualization down below shows how the distribution of these samples is relatively normal with a slight left skew.

Assuming the population is represented by our sample, we are 95% confident that the difference of median total yearly compensation between software engineers and managers with a bachelors and a masters is between 30,000 and 22,000.

```{r CI1-chunk-1}

#Summary statistics of both software engineers and managers with a masters and a Bachelors. The mean and median is calculated for each.

software_salaries %>%
```

~~~
  filter(Education == "Masters") %>%
  summarize(Mster_comp_mean = mean(totalyearlycompensation),
            Mster_comp_median = median(totalyearlycompensation))

software_salaries %>%
  filter(Education == "Bachelors") %>%
  summarize(Bach_comp_mean = mean(totalyearlycompensation),
            Bach_comp_median = median(totalyearlycompensation))


```
```{r CI1-chunk-2}

# A new data frame is created for analysis. A new column is added to differentiate between people with Bachelors, Masters, and other types of education.

analysis_1df <- software_salaries %>%
  mutate(
    Education_New = case_when(

      Education == "Bachelors" ~ "BachelorsDegree",
      Education == "Masters" ~ "MastersDegree",
      TRUE ~ "other"
    )
  )

# Filters out other eduction levels as to make a binary comparison between software engineers and managers with a Masters degree and a Bachelors Degree.

analysis_1df <- analysis_1df %>%
  filter(Education_New != "other")

analysis_1df

```



```{r CI1-chunk-3}

# Setting the seed
set.seed(5734)

#Calculated a bootstrap distribution of the difference in median total yearly compensation between software engineers and managers with a bachelors and a masters degree.
~~~

```
boot_diff_median1 <- analysis_1df %>%
  filter(!is.na(Education_New)) %>%
  specify(totalyearlycompensation ~ Education_New ) %>%
  generate(reps=5000, type="bootstrap") %>%
  calculate(stat="diff in medians",
          order=c("BachelorsDegree", "MastersDegree") )

boot_diff_median1 %>% visualize()

boot_diff_median1 %>% summarize(median = median(stat),
                   lower = quantile(stat, 0.025),
                   upper = quantile(stat, 0.975))

```

### Confidence Interval 2: difference of median total yearly compensation for years of experience greater than 5 years (above) and 5 years or less (below).

In this section, we find the confidence interval for the difference of median total yearly compensation for years of experience. To do some minor cleaning, we make years of experience binary by categorizing data that is above the median (5 years) and below. As in the first confidence interval, the same process of setting the seed, specifying the explanatory and responding variables, generating re-sampling with replacement using bootstrap, and calculating the difference in medians was followed. The distribution is relatively normal with a slight right skew. Assuming the population is represented by our sample, we are 95% confident that the difference of median total yearly compensation between software engineers and managers with five or less years of experience and more than five is between 77,000 and 84,000. What we can observe is that the upper and lower limits are fairly close. It is also interesting to see such large numbers for the median difference in total yearly compensation. However, this is not surprising given that these values are aggregated. In reality, the years of experience ranges from 0 to 40 which greatly affect the difference in median total yearly compensation.

```{r CI2-chunk-1}

software_salaries %>%
  summarize(med_years_exp = median(yearsofexperience))

analysis_2df <- software_salaries %>%
  filter(!is.na(yearsofexperience)) %>%
  mutate(years_experience_binary = ifelse(yearsofexperience > 5, "above", "below"))

```

```{r CI2-chunk-2}
```

```
set.seed(5734)

boot_diff_median2 <- analysis_2df %>%
  specify(totalyearlycompensation ~ years_experience_binary) %>%
  generate(reps=5000, type="bootstrap") %>%
  calculate(stat="diff in medians",
            order=c("above", "below") )

boot_diff_median2 %>% visualize()

boot_diff_median2 %>% summarize(median = median(stat),
                        lower = quantile(stat, 0.025),
                        upper = quantile(stat, 0.975))


```

### Confidence Interval 3: difference of median total yearly compensation between males and females.

This confidence interval displays the difference of median total yearly compensation between males and females. The only data cleaning I conducted was to filter out any genders that were considered other. This left only males and females in which I created bootstrap distribution. Based on this distribution, we are 95% confident that the difference of median total yearly compensation between software engineers and managers with five or less years of experience and more than five is between 15,000 and 22,000. The distribution peak of the distribution is around 19,000 but there is also a spike at the lower limit of 15,000. What we've observed is significant given the fact that men have historically been compensated more than women. However, other contributing factors may be from the higher proportion of men to women in this dataset as well as the STEM field being a more male dominated field in general.

```{r CI3-chunk-1}
analysis_3df <- software_salaries %>%
  filter(gender != "Other")

set.seed(5734)

boot_diff_median3 <- analysis_3df %>%
  filter(!is.na(gender)) %>%
  specify(totalyearlycompensation ~ gender ) %>%
  generate(reps=5000, type="bootstrap") %>%
  calculate(stat="diff in medians",
            order=c("Male", "Female") )
```

```
boot_diff_median3 %>% visualize()

boot_diff_median3 %>% summarize(median = median(stat),
                    lower = quantile(stat, 0.025),
                    upper = quantile(stat, 0.975))


```

### Confidence Interval 4: difference of median total yearly compensation between major tech companies and non-major tech companies.

This confidence interval displays the difference of median total yearly compensation between major tech companies and non-major tech companies. Assuming the population is represented by our sample, we are 95% confident that the difference of median total yearly compensation between software engineers and managers with major tech companies and non-major tech companies is between 48,000 and 55,000. This tells us that there is a significant difference between the salaries of software engineers at major tech and non-tech companies. This means that the title of software engineer or manager is not as valued as the type of company you're working for.

```{r CI4-chunk-1}
set.seed(5734)

boot_diff_median4 <- software_salaries %>%
  filter(!is.na(major_tech_company)) %>%
  specify(totalyearlycompensation ~ major_tech_company) %>%
  generate(reps=5000, type="bootstrap") %>%
  calculate(stat="diff in medians",
          order=c("yes", "no") )

boot_diff_median4 %>% visualize()

boot_diff_median4 %>% summarize(median = median(stat),
                    lower = quantile(stat, 0.025),
                    upper = quantile(stat, 0.975))


```

### Confidence Interval 5: difference of median total yearly compensation between high income states and not high income states.

This confidence interval displays the difference of median total yearly compensation between high income states and non-high income states. Assuming the population is represented by our sample, we are 95% confident that the difference of median total yearly compensation between software engineers and managers in high income states and not high-income states is between 76,000 and 83,000. This tells us that there is a significant difference between the salaries of software engineers in high income states and not high income states. This means that location heavily influences salary. Other contributing factors to this result could be the location of major tech companies and non-major tech companies.

~~~~
```{r CI5-chunk-1}
set.seed(5734)

boot_diff_median5 <- software_salaries %>%
  filter(!is.na(high_income_state)) %>%
  specify(totalyearlycompensation ~ high_income_state) %>%
  generate(reps=5000, type="bootstrap") %>%
  calculate(stat="diff in medians",
            order=c("yes", "no") )

boot_diff_median5 %>% visualize()

boot_diff_median5 %>% summarize(median = median(stat),
                       lower = quantile(stat, 0.025),
                       upper = quantile(stat, 0.975))


```
~~~~

## Hypothesis Testing

### Education: Bachelor's, Master's?

~~~~
```{r bach-vs-mast}
compensation_prop_edu <- software_salaries_bach_mast %>%
  count(Education, compensation_level) %>%
  group_by(Education) %>%
  mutate(prop=n/sum(n))

compensation_prop_edu

set.seed(141)
~~~~

~~~
null_dist <- software_salaries_bach_mast %>%
  specify(response=compensation_level, explanatory=Education, success="above median") %>%
  hypothesize(null="independence") %>%
  generate(reps=10000, type="permute") %>%
  calculate(stat = "diff in props", order=c("Masters", "Bachelors"))

obs_diff = 0.541-0.412
null_dist %>% visualize() + shade_p_value(obs_stat=obs_diff, direction="both")
null_dist %>% get_p_value(obs_stat=obs_diff, direction="both")
```


### Years of Experience

Null Hypothesis: Years of experience and total yearly compensation are independent

Alternative: Years of experience and total yearly compensation are dependent

```{r experience-level}
software_salaries <- software_salaries %>%
  mutate(experience_level = case_when(
    yearsofexperience > median(yearsofexperience) ~ "experienced",
    yearsofexperience < median(yearsofexperience) ~ "not experienced"))

compensation_prop_exp <- software_salaries %>%
  filter(!is.na(high_low_compensation), experience_level %in% c("experienced", "not experienced")) %>%
  count(experience_level, high_low_compensation) %>%
  group_by(experience_level) %>%
  mutate(prop=n/sum(n))

compensation_prop_exp

set.seed(141)
null_dist_2 <- compensation_prop_exp %>%
  specify(high_low_compensation ~ experience_level, success="high") %>%
  hypothesize(null="independence") %>%
  generate(reps=1000, type="permute") %>%
  calculate(stat= "diff in props", order=c("experienced", "not experienced"))

obs_diff_2 <- 0.702-0.294
null_dist_2 %>% visualize() + shade_p_value(obs_stat = obs_diff_2, direction="both")
~~~

~~~
null_dist_2 %>% get_p_value(obs_stat = obs_diff_2, direction="both")
```
~~~

### High income state

Null Hypothesis: High income state and total yearly compensation are independent

Alternative: High income state and total yearly compensation are dependent

~~~
```{r state}
compensation_prop_state <- software_salaries %>%
  filter(!is.na(high_low_compensation), high_income_state %in% c("yes", "no")) %>%
  count(high_income_state, high_low_compensation) %>%
  group_by(high_income_state) %>%
  mutate(prop=n/sum(n))

compensation_prop_state

set.seed(141)
null_dist_3 <- compensation_prop_state %>%
  specify(high_low_compensation ~ high_income_state, success="high") %>%
  hypothesize(null="independence") %>%
  generate(reps=1000, type="permute") %>%
  calculate(stat= "diff in props", order=c("yes", "no"))

obs_diff_3 <- 0.588-0.189
null_dist_3 %>% visualize() + shade_p_value(obs_stat = obs_diff_3, direction="both")
null_dist_3 %>% get_p_value(obs_stat = obs_diff_3, direction="both")
```
~~~

## Conclusion

What factors influence the compensation of US software engineers? We were able to model the effects of specific variables on yearly compensation. Also, we were able to give confidence levels for these variables on how much they increase yearly compensation. Lastly, we were able to determine how significant observed differences in variables were through p-value testing.

From this, we were able to find several key variables that determine the yearly compensation a software engineer can earn. Getting an education beyond a bachelor's degree proved to be a strong indication of earning a higher compensation. In terms of what company to work for, tech companies tended to provide a larger compensation. Perhaps the strongest variable was where you lived. Living in a high-income state greatly improved the compensation a software engineer earns, albeit likely at the cost of higher living expenses. Lastly, having a lot of experience was shown to improve the compensation of software engineers.

Overall, having a data set with over 10,000 observations makes us confident that we can be representative of a population. But of what population? We would like for our analysis to apply to all software engineers in the US but there are issues with this assumption. Our data comes from a website where participants submit their financial information to compare with each other. Would a well-off engineer use this site to show off their new raise? Or would a meager young engineer that feels like they aren't getting paid enough be more likely to use this site? We don't know. In reality, our results are really only representative of users from this site and may not be representative of software engineers as a whole. Going out and obtaining data may provide more representative results.