Transformation of Variables in Frequency of Helicopters & Two Factor ANOVA of Salary of Technology Workers

Project II - Group 3

Dongmin Wu, Leonard Chandra, Om Chaudhary, Danny Kuei, Harveen Thukral

Instructor: Maxime Guiffo Pouokam

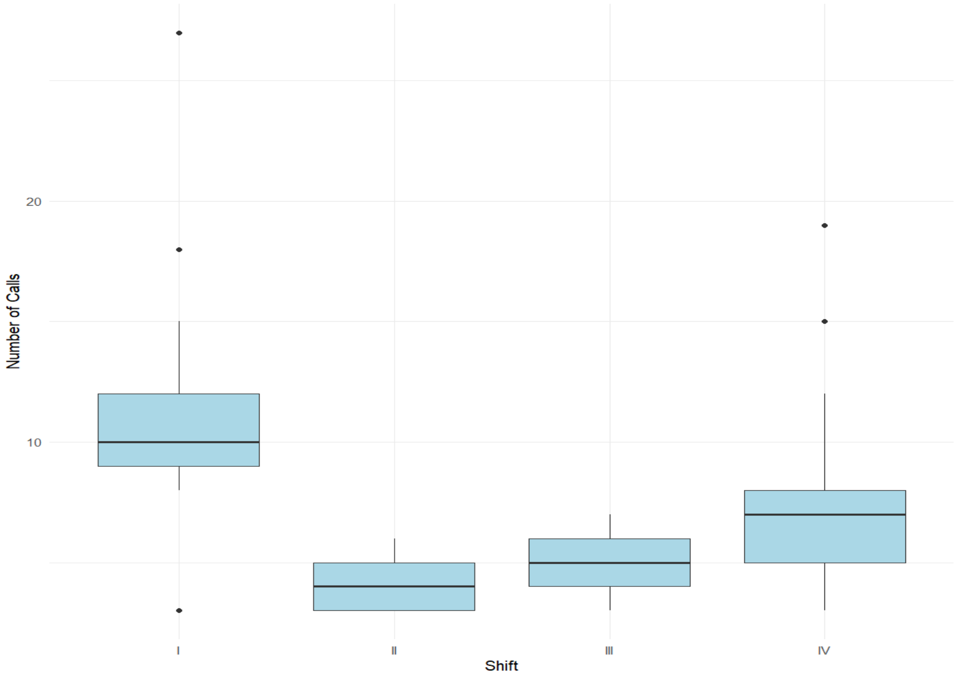
September 2, 2024

Figure 1: Picture of Helicopter in air

Topic I: Transformation of Variables

I. Introduction

This dataset examines the number of times helicopters were called due to emergency situations in different shifts of the Sheriff's Office within a year. The data is divided into two columns: the first column represents the number of helicopter calls, and the second column represents the specific flights that make these calls. The shift is divided into four time periods: Shift I (2:00 AM to 8:00 AM), Shift II (8:00 AM to 2:00 PM), Shift III (2:00 PM to 8:00 PM), and Shift IV (8:00 PM to 2:00 AM). The purpose of this analysis is to determine whether the time of day significantly affects the frequency of helicopters called.

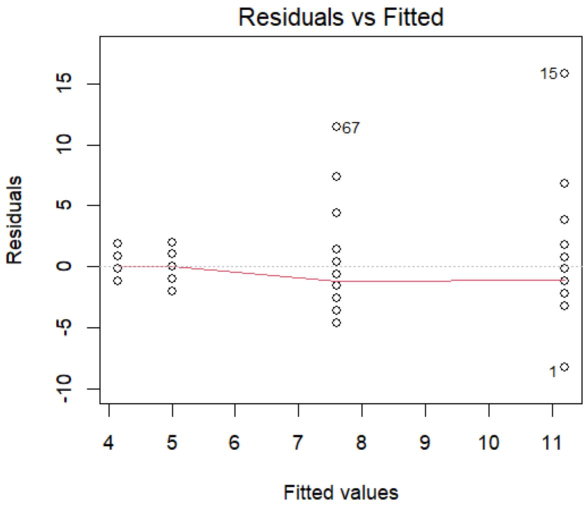


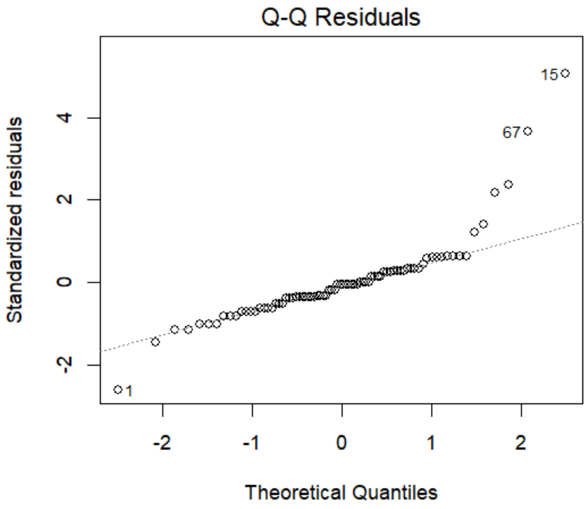
Shift I: The helicopter call has the largest variability and a wider distribution, with several outliers indicating that some flights have significantly more calls than others.

Shift II and III: The median number of calls is lower, and the Interquartile range (IQR) is smaller, indicating that the number of calls during this period is more consistent.

Shift IV: The median of calls is higher compared to shift II and III, and some outliers indicate that emergency calls occasionally reach their peak.

II. Diagnostic Plots/Tests





Residuals vs Fitted: The plot has no obvious pattern, the residuals appear to be scattered randomly around the zero line. However, residuals like those labeled “15” and “67” deviate from the bulk of the data which could mean the presence of outliers which may impact the model’s fit and its assumptions. Outliers seem to make it diverge into a higher spread which might indicate a violation of the constant variance assumption.

Normal Q-Q Plot: The plot shows residuals are mostly following a straight line, however, it has noticeable deviations at both ends of the plot, which might suggest the residuals are not normally distributed. The deviations from the line such as points labeled “1”, “15”, and “67” at both ends of the distribution could indicate the residuals are not perfectly normally distributed.

Shapiro-Wilk normality test: W = 0.79712, p-value = 3.945e-09

	Df	**F value**	**Pr(>F)**
group	3	3.0942	0.03186
	76		

Levene's Test for Homogeneity of Variance (center = median)

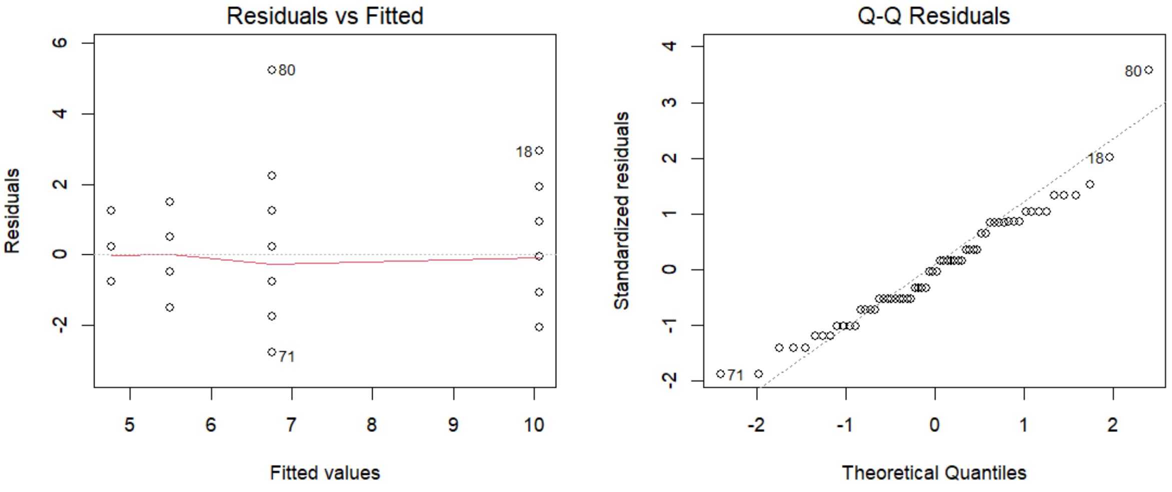
The p-value for Shapiro-Wilk test is extremely small, much less than 0.05 indicating a significant deviation from normality. Which means the residuals do not follow a normal distribution.

The p-value for Levene’s test below 0.05 indicates that the assumption of equal variances is violated.

III. Transformation

In this section, we will be transforming our data. We will be using the Box-Cox method. Using the grid search method in R, we found lambda to be -0.2626263 with outliers and 0.1414141 without outliers. We will now fit both the transformed data with and without outliers and test the diagnostics for both models.

Data with Outliers Removed

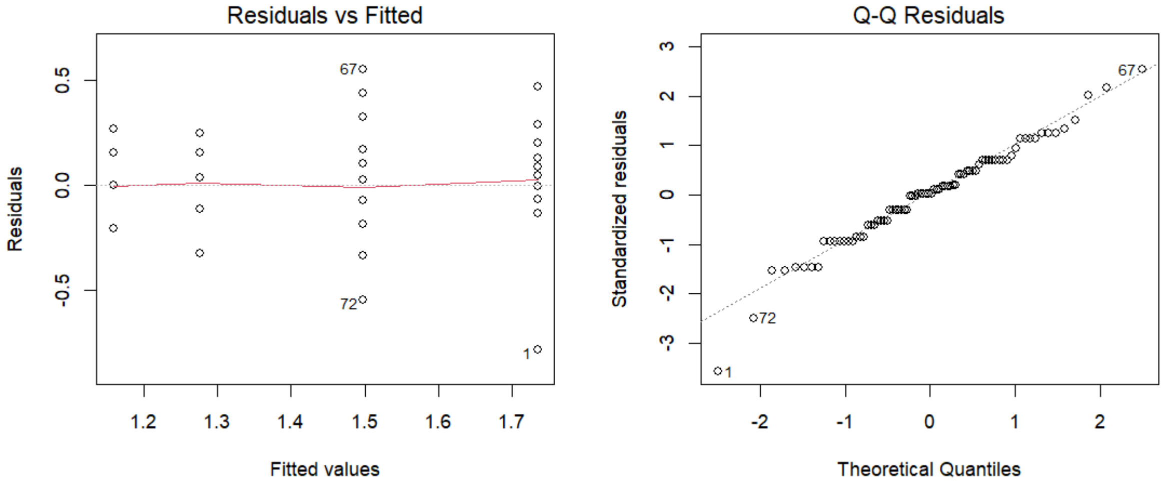


Shapiro-Wilk normality test: W = 0.96125, p-value = 0.04789

	Df	F value	Pr(>F)
group	3	2.5862	0.06172
	58		

Levene's Test for Homogeneity of Variance (center = median)

Box-Cox Transformation

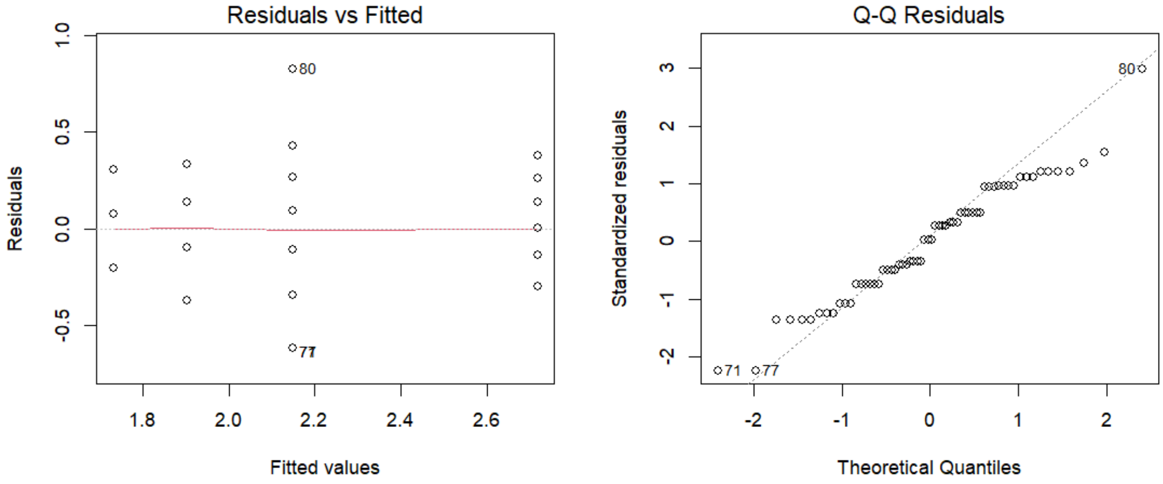


Shapiro-Wilk normality test: W = 0.9743, p-value = 0.1069

	Df	**F value**	**Pr(>F)**
group	3	0.5405	0.656
	76		

Levene's Test for Homogeneity of Variance (center = median)

Box-Cox Transformation (After Removing Outliers)



Shapiro-Wilk normality test: 0.96987, p-value = 0.1309

	Df	**F value**	**Pr(>F)**
group	3	1.8833	0.1424
	58		

Levene's Test for Homogeneity of Variance (center = median)

III. Results

The **Box-Cox Transformation (After Removing Outliers)** appears to be the best model because:

- The **Shapiro-Wilk Test** shows the highest p-value, indicating the residuals are most likely be normally distributed.
- The **Levene's Test** shows p-value is well above 0.05 indicating equal variance.
- The **Q-Q plot** and **Residuals vs. Fitted values plot** shows more consistent behavior compared to the original data.
- The **outliers** can have a significant impact on the results of ANOVA by influencing the mean and variance, which is why we choose Box-Cox without outliers over Box-Cox with outliers even though the Q-Q plot appears to be slightly better for Box-Cox with outliers.
- The p-value for the Shapiro-Wilk test for data without outliers is still below 0.05, indicating the residuals are not normally distributed, which is why we don't choose data without outliers and transformation.

IV. Discussion

Did Transforming the Data Help?

Yes, using the Box-Cox transformation significantly improved the normality of the residuals and the homogeneity of variances, which are important assumptions for ANOVA.

What are the Downsides?

After transformation, interpretation can be more complex and difficult. Since after the Box-Cox transformation, data is not in the original units of measurement anymore and can be hard to interpret. Removing outliers can sometimes result in the loss of valuable information if outliers are some important data points i.e. representing rare but significant events which might cause a less accurate representation of the real-world scenario.

Is the Transformed Data a Better Fit?

Yes, the transformed data provides a better fit for ANOVA. The diagnostics show improved adherence to ANOVA assumptions, which means the results from the transformed data are more reliable.

Recommendations for a Client Using This Data Set for ANOVA

I recommend using the Box-Cox Transformation after removing outliers since it gives the best balance across all diagnostic checks, which also ensures the assumptions of ANOVA are met. If the client is worried about losing important data from removing outliers, they can consider using Box-Cox Transformation with outliers.

Topic II: Two Factor ANOVA

I. Introduction

In part two of this project, we chose to use the Salary dataset. This data is taken from a random sample of “technology workers” from Seattle and San Francisco. The dataset is structured with three main columns:

- **Annual**: This column represents the subjects' annual salary in thousands of dollars.
- **Prof**: This denotes the subjects' profession, with values including DS (Data Scientist), SE (Software Engineer), and BE (Bioinformatics Engineer).
- **Region**: This indicates the city where the subjects work, with values SF for San Francisco and S for Seattle.

The main question we will attempt to answer is *if there are significant differences in salaries based on profession, region, or the interaction between both these factors*. We will refer to Profession as Factor A and Region (City) as Factor B. We want to answer this question because understanding what impacts salary differences is very valuable information for the people in these professions. It can provide insights into regional market trends, inform future career and life choices, and help companies stay financially aware and competitive in a rapidly changing job market.

Since we have two factors A and B, we can conduct a Two-Factor Analysis of Variance (ANOVA) for our analysis. This specific statistical approach allows us to assess the main effects of both factors (profession and region) on the salaries, as well as any interaction effect between the two. The findings from this analysis will give us a solid understanding of how these factors influence salary variations within the tech industry.

We will begin with a summary of the data, including descriptive statistics and visualizations such as histograms, boxplot and interaction plot. We will first perform diagnostics of the data to validate and check the assumptions of our ANOVA using plots such as residual vs fitted and Q-Q plots. To formally assess these assumptions, we will conduct several diagnostic tests including the Shapiro-Wilk test to check for the normality of the residuals and Levene's test to assess the homogeneity of variances. We can then perform our Two-Factor ANOVA to statistically test the main effects of Profession and Region, and their interaction on salaries.

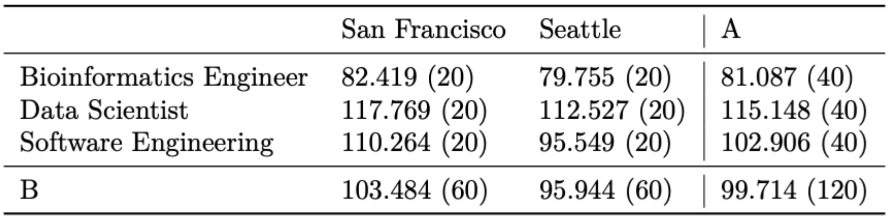
We will set our significance level (α) at 0.05 for all our tests (Shapiro-Wilk test, Levene’s test, and ANOVA itself). In our ANOVA analysis, we will be doing 4 pairwise and 2 contrast comparisons for average difference between the groups using appropriate multipliers such as Tukey, Scheffe and Bonferroni multipliers. Finally, after conducting the ANOVA, we will interpret the results and summarize our findings.

II. Summary of the Data

In this section, we aim to better understand the data by providing an overarching overview of the datasets, discussing key summary values, and utilizing various plots.

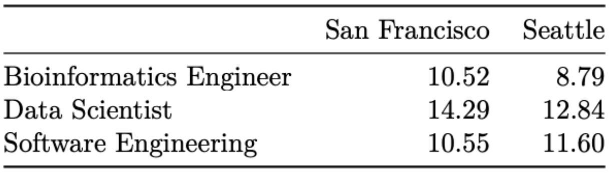
From the summary statistics, the overall mean salary across all professions and regions is approximately *$99,714* with a standard deviation of *$18,692*. We can break this down by profession and region as follows:

Table 1: Average annual salaries (in thousand of dollars)



	San Francisco	Seattle	A
Bioinformatics Engineer	82.419 (20)	79.755 (20)	81.087 (40)
Data Scientist	117.769 (20)	112.527 (20)	115.148 (40)
Software Engineering	110.264 (20)	95.549 (20)	102.906 (40)
B	103.484 (60)	95.944 (60)	99.714 (120)

Table 2: Standard deviation of annual salaries



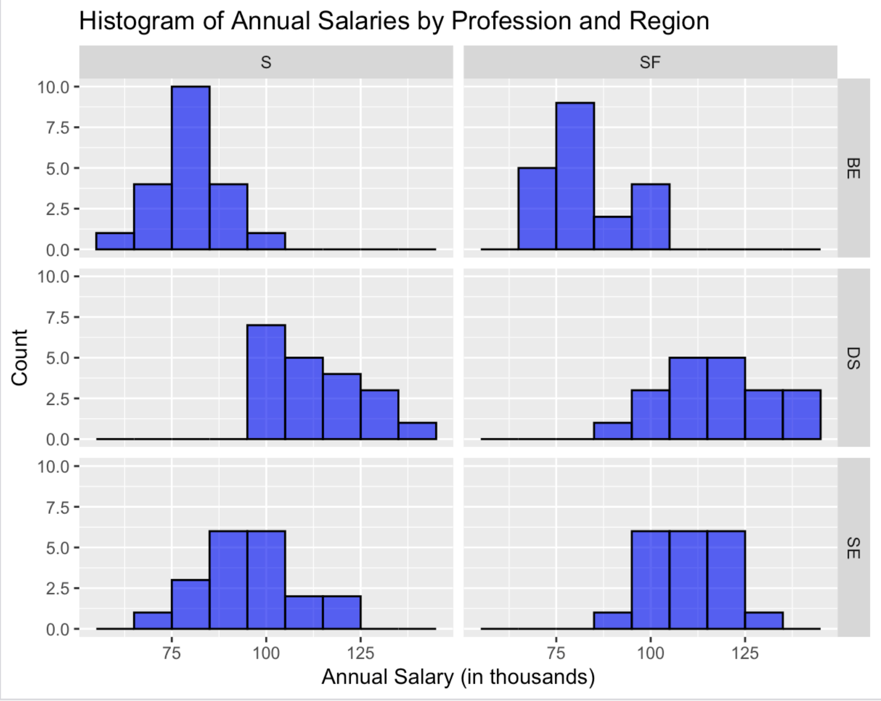
	San Francisco	Seattle
Bioinformatics Engineer	10.52	8.79
Data Scientist	14.29	12.84
Software Engineering	10.55	11.60

In table 1, The A and B in the table represent the marginal means of factor A for profession and factor B for city with the corner right the overall mean for the entire treatment group. We can see here that the Bioinformatics Engineer has the average lowest annual salary compared to other professions in both cities. We see that the average salary of the profession between the cities is similar, with one exception with Software Engineering with a significant drop in Seattle. This might suggest there might be interaction between the city for Software

Engineering jobs. Sample sizes are denoted inside the brackets. We see here that the sample sizes for all of the groups are equal and we have a balanced design.

Another summary is the standard deviation that we can see in table 2. We see that overall the standard deviation is similar to each other. This might suggest that the data have equal variance. One thing to note is that there is significant variability for the Data Scientist jobs, which might indicate that the salary for Data Scientist fluctuates more.

To better visualize this, we will utilize histograms to show the distribution of a single variable (annual salaries) for the city and profession. They will provide us with a good visual representation of how frequently different salary ranges occur.

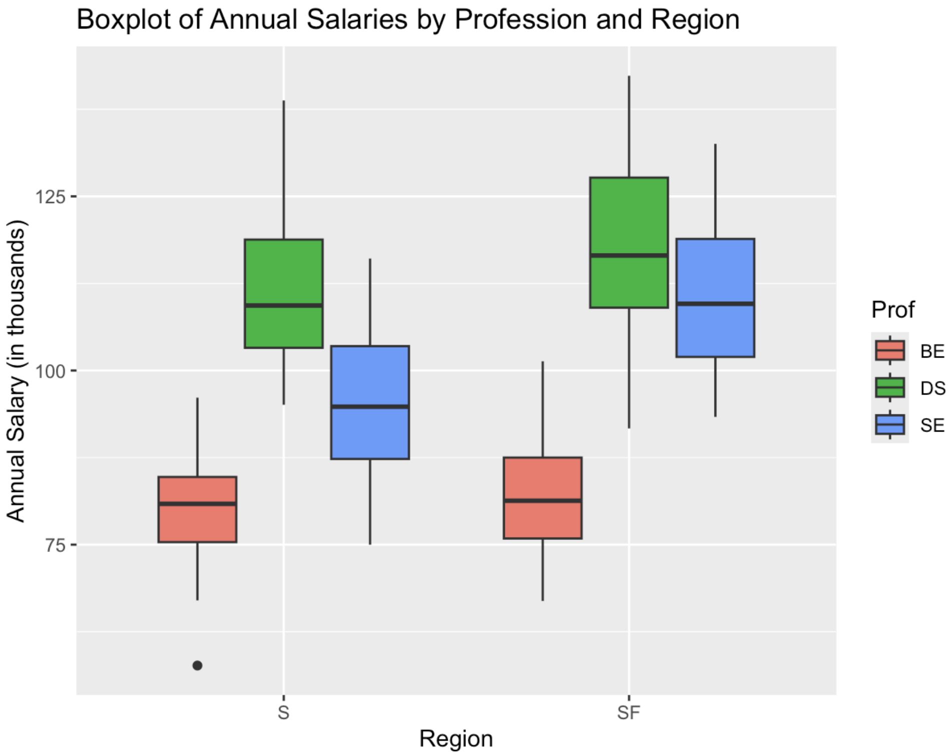


The *Bioinformatics Engineers (BE)* histogram shows that the salaries in Seattle are more clustered around the $75,000-$85,000 range. On the other hand, San Francisco has a more equal spread, suggesting that there is a larger range of salary for the job than in Seattle. If we compare to the other plots, we can see that for both Seattle and San Francisco, they have the lowest average annual salary.

The *Data Scientists (DS)* histogram in Seattle shows a spread of salaries with a significant portion in the lower $90,000-$110,000 range. In contrast, the histogram for SF is spread more evenly but is more concentrated in the higher salary range ($110,000-$130,000), reflecting generally higher pay in this region. The wider spread indicates some variability.

The *Software Engineers (SE)* histogram in Seattle shows a more symmetric distribution centered around $90,000-$100,000. However in SF, there is a slight skew towards higher salaries on both ends. This indicates that software engineers in SF have a more stable income with lower chances of getting a low or higher pay.

Next, we will utilize a series of boxplots which are useful for comparing distributions across different groups. They provide a summary of the distribution of a dataset by showing the median, interquartile range, and potential outliers in the data. In this case, the groups are professions and cities.

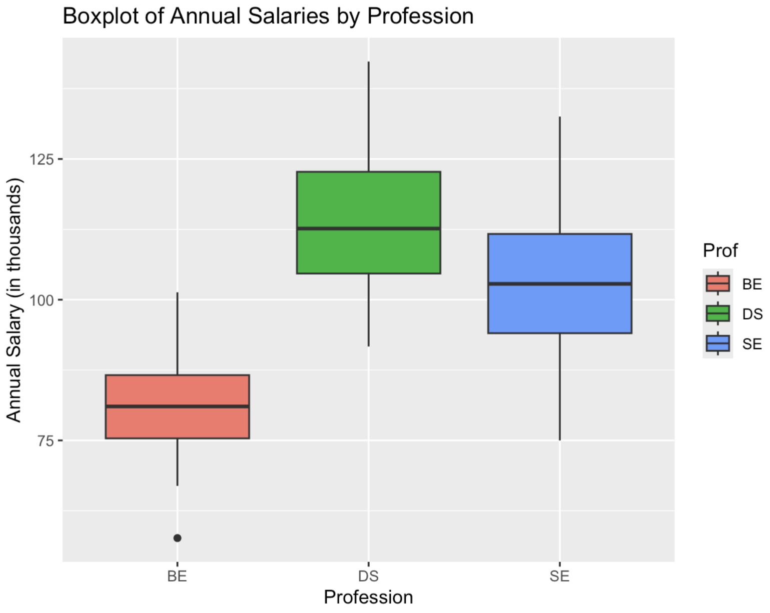


For *Bioinformatics Engineers (BE)*, the boxplot shows that the median salary in San Francisco is higher than in Seattle. Seattle has less variability as indicated by the spread of salaries (IQR) which is slightly wider in SF. Seattle also has one outlier on the lower end which could indicate a possibility of underpay.

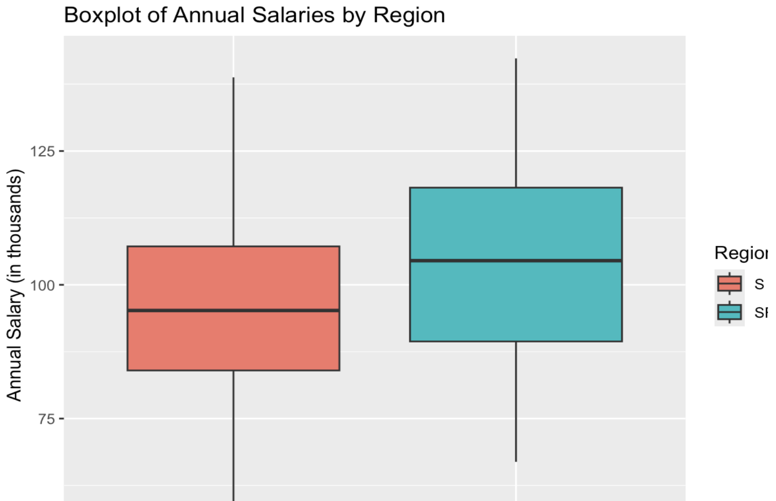
For *Data Scientists (DS)*, the boxplot reveals that the median salary in San Francisco is higher than in Seattle. The IQR is slightly wider in SF which reflects more variability which could indicate that Data Scientists jobs do not have a fixed range.

For the *Software Engineers (SE)*, the boxplot shows that the median salary in San Francisco is higher than in Seattle. We observe a narrower IQR for SF which suggests less variability in salaries. This indicates that on average Software Engineers receive a higher pay level in SF.

We will then proceed to divide the boxplots into smaller, more digestible data. We will split each boxplot by profession and the region.

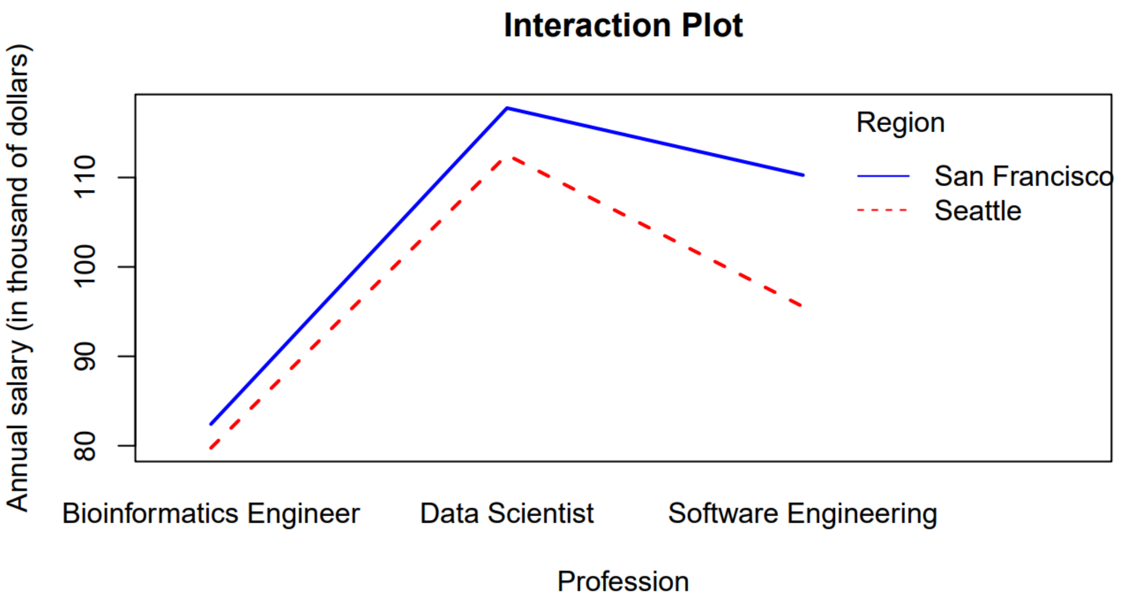


These boxplots show us that Data Scientists have the highest median salaries among the three professions. Next are Software Engineers, and then lastly Bioinformatics Engineers earning the least. This trend is consistent regardless of region. The IQRs show that Data Scientists earn the most on average and also have a wider range of salaries.



The boxplot comparing salaries by region shows us that San Francisco has a higher median salary across all professions. The IQR for San Francisco is also slightly wider which indicates more variability in overall salaries. However, overall, the higher median suggests SF has a more lucrative job market.

We want to now determine whether there is any significant interaction between the city and the profession to the annual salaries.



We see here for the interaction plot, the lines are mostly parallel to each other. We see that there might be interaction in the annual salaries for the region for Software Engineering jobs. This graph supports our findings in table 1 where there was a significant difference between the job in Seattle and San Francisco.

III. Diagnostics

In this section we will be performing model diagnostics for our ANOVA model. In Two-Factor ANOVA, we make assumptions for the error term in our model. Before we perform ANOVA analysis, we must check to see whether the assumptions of ANOVA are being met. The following assumptions are:

1.) All Y_{ij} were randomly sampled (independent)

2.) All groups are independent

3.) $\varepsilon_{ij} \sim N(0, \sigma_{\varepsilon})$ for all i = control, enlarged, and reduced

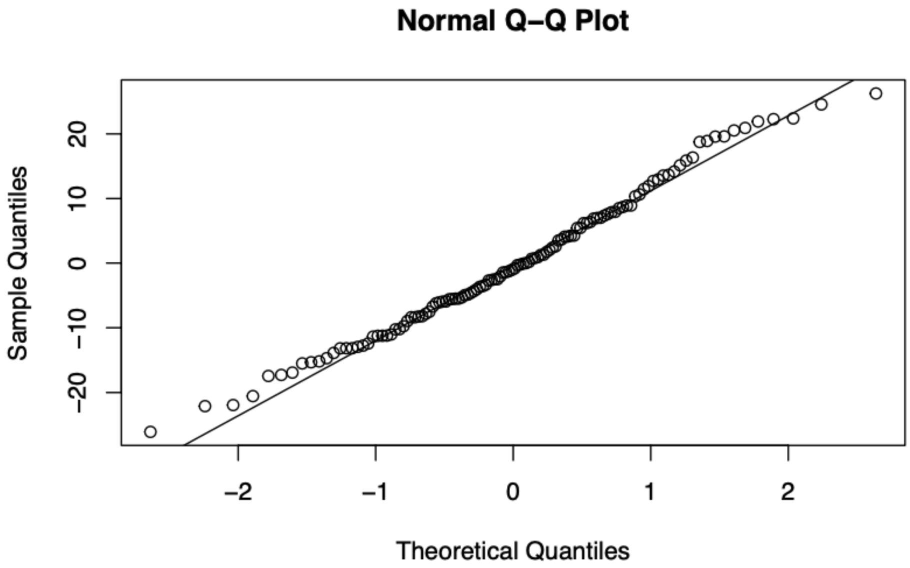
We will only be testing assumption 2 and 3 for normality and equal variance since we cannot test whether our data were sampled randomly. We will also include a formal test using the Shapiro-Wilk test and Levene's test to determine whether they are statistically significant.

Outliers also can be a potential factor that causes non-normality or non-constant variance, we aim to remove outliers via a semi-studentized or studentized approach. When we tested for outliers, we found that there were no outliers in our dataset. We will now assess whether our ε_{ij}'s in our data are normally distributed and have constant variance using the full model with interaction to account for all variability.

III.A: Testing for Normality

To check if ε_{ij}'s are normally distributed in which we can utilize two methods and we will go over the Q-Q plot first.

III.A.1 Q-Q Plot



A Q-Q plot calculated centered percentiles of our data and compares to what they should've been if the data was normal. Theoretically, a scatter plot following the straight linear line would be perfectly normally distributed.

From the Normal Q-Q plot above, we can see there are a couple of points on the tails that skew slightly higher than the normal line. However, most of the data along the middle falls along the normal line so our data appears to be normally distributed. Since interpreting plots is subjective, we will now perform the Shapiro-Wilks test..

III.A.2 Shapiro-Wilks Test

The Shapiro-Wilks Test is a hypothesis test to test our normality assumptions. Our hypothesis for the null and alternative will be the following:

H_0: The salary dataset is normally distributed.

H_A: The salary dataset is not normally distributed.

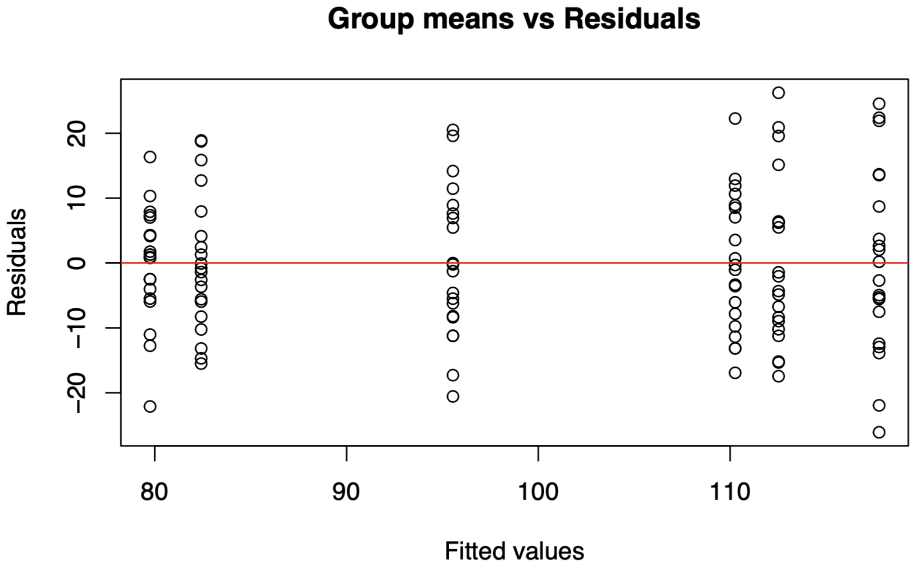
Through R, we obtained a p-value of 0.324 (Table 3.) Since our p-value is larger than our coefficient α, we fail to reject the null hypothesis and conclude that our dataset is normally distributed.

III.B: Testing for Equal Variance Variance

Similarly, there are also two ways to check f or constant variance in which the first method will be visually using a plot of the Errors vs Group means and Brown-Forsythe for the second method.

Since we have 3 groups in the profession factor (Factor A) and 2 groups in the region factor (Factor B) we will have a total of 6 groups. By plotting out the residuals for each group, we can observe the plot to see if there is constant variance.

III.B.1 Residuals ε_{ij} vs. Fitted Values Plot Y_{ij}



From the plot above, we can see that there is an approximately equal spread within each group. However, this is also subjective so we can perform the Brown-Forsythe Test.

III.B.2 Brown-Forsythe Test

Our hypothesis test will be the following:

$$H_0\text{: } \sigma_{DS,SF} = \sigma_{DS,S} = \ \sigma_{SE,SF} = \sigma_{SE,S} = \sigma_{BE,SF} = \ \sigma_{BE,S}$$

$$H_A\text{: At least one of the } \sigma_i \text{ is not equal.}$$

where with the null hypothesis states that each standard deviation is equal to each other that with σ as the standard deviation of each treatment group.

Through R, we obtained a p-value of 0.305 (Table 3.) Since our p-value is larger than our significance level α, we fail to reject the null hypothesis and conclude that our group variances are equal.

Thus, considering our diagnostic tests for normality and constant variance, we can see that our dataset does not violate the assumptions of ANOVA and we can proceed in our analysis without transformations. We will assume that each error term is independent. Below are the p values for the formal test that we have done.

Table 3: Normality and Equal Variance Test

	p value
Shapiro-Wilk	0.324
Brown-Forsythe	0.305

IV. Analysis

When we are performing Two Factor ANOVA analysis, we have to decide which model is appropriate to use amongst the following models: Factor A and B model with interaction, Factor A and B model without interaction, Factor A model only, or Factor B model only. After fitting the model for each combinations, we get:

Table 5: Model Summary

Values	AB	A + B	A	B	·
Mean of Squared Errors	133.8	138.43	151.83	337.91	349.4
Sum of Squared Errors	15252.93	16058.34	17764.09	39872.94	41578.69
Degrees of Freedom	114	116	117	118	119

Where,

AB: the model with interaction

A + B: the model with factor A and B effects without interaction

A: the model with only factor A effects

B: the model with only factor B effects

· : the empty model without any factors

We calculated the table using the following equation:

$$SSE = \sum_{i}^{ab} (n_i - 1)\, s_i^2$$

$$df\{SSE\} = n_T - \# Parameters$$

$$MSE = \frac{SSE}{df\{SSE\}}$$

where ab is the number of factor A and factor B multiplied for i treatments of A and B.

We calculated (SSE) and subsequently their degrees of freedom (d.f) and mean of squared errors (MSE) for each model (Table 5). Additionally, as an initial indicator for which model may be best fit, we calculated the conditional R^2 for each model using the formula

$$R^2\{reduced\ model\ |\ full\ model\} = \frac{(SSE_R - SSE_F)}{SSE_R}$$

These values tell us the percent error in reduction when using their respective models.

Table 6: Conditional R²

	(AB | A + B)	(A + B | B)	(A + B | A)	(A | ·)	(B | ·)
Percentages	5.02%	59.73%	9.6%	57.28%	4.1%

Each percentage represents the amount of reduction from the reduced model to the full model. We see here that there might be no interaction as the error reduction is very low when adding interaction to the model of factor A and B. We see that factor A has the most effect when adding the model to model B and from the empty model.

IV.A Factor effect testing

IV.A.1: Test for Interactions

To determine which model to use, we want to do a formal test rather than the suggestions from conditional R^2, we first want to determine if the interaction effects were statistically significant using hypothesis testing. If we consider our "full model" with interaction to be

$$Y_{ijk} = \mu_{..} + \gamma_i + \delta_i + (\gamma\delta)_{ij} + \epsilon_{ijk}$$

and our "reduced model" without interaction to be

$$Y_{ijk} = \mu_{..} + \gamma_i + \delta_i + \epsilon_{ijk}.$$

$$\text{where, } \sum_i \delta_i = 0 \ and \sum_{ij} (\gamma\delta)_{ij} = 0$$

then our hypothesis test will be

$$H_0\text{: All } (\gamma\delta)_{ij} = 0$$

$$H_A\text{: At least one } (\gamma\delta)_{ij} \neq 0$$

The F-statistic can be calculated by

$$Fs = [\frac{SSE_R - SSE_F}{df\{SSE_R\} - df\{SSE_F\}}]/MSE_F$$

and its subsequent p-value calculated by

$$df\{numerator\} = df\{SSE_R\} - df\{SSE_F\}$$

$$d.f\{denominator\} = df\{SSE_F\}$$

$$\text{p-value} = Pr\{F > F_s\}$$

where SSE_F is the SSE for the full model and SSE_R is the SSE for the reduce model.

With model AB as our full model with interaction and model A+B as our reduced model without interactions, we can use the values obtained from Table 7 to generate our factor effect tests.

Table 7: Factor Effect Test

	(AB | A + B)	(A + B | B)	(A + B | A)	(A | ·)	(B | ·)
F Statistic	3.01	86.01	12.32	78.43	5.05
p value	0.053236	< 2e-16	0.000638	< 2e-16	0.026514

Since, the p-value for model (AB | A +B) is 0.053236 > α, we fail to reject the null hypothesis and conclude that interaction effects are not statistically significant. Thus, we will need to test for Factor A and B effects to see whether we should use a two factor model without interactions.

IV.B: Test for Factor A and B Effects

In the test for Factor A effects, our “full model” will be

$$Y_{ijk} = \mu.. + \gamma_i + \delta_j + \epsilon_{ijk}$$

and “reduced model” to be

$$Y_{ijk} = \mu.. + \delta_j + \epsilon_{ijk}$$

$$\text{where, } \sum_i \delta_i = 0 \ and \sum_i \gamma_i$$

with hypothesis tests

$$H_0\text{: All } \gamma_i = 0 \text{ for all i}$$

$$H_A\text{: At least one } \gamma_i \neq 0.$$

The test for Factor B effects is similar to the test for Factor A effects in that the full model will be the same. However, the "reduced model" is instead

$$Y_{ijk} = \mu_{..} + \gamma_i + \epsilon_{ijk}$$

$$\text{where, } \sum_i \gamma_i$$

with hypothesis tests

$$H_0\text{: All } \delta_j = 0 \text{ for all i}$$

$$H_A\text{: At least one } \delta_j \neq 0.$$

Similar to the test for interactions, we will be using the same formulas to calculate the F-statistic and p-value. Given the results from Table 7, we see the tests for Factor A effect (A + B | B) and Factor B effect (A + B | A) have p-values of < 2e-16 and 0.000638 respectively. Since both p-values < α, we can reject the null hypotheses for these tests and conclude that both Factor A and Factor B effects exist. Therefore, from our hypothesis testing, the appropriate model to use would be the Factor A and B model with no interactions.

IV.C Confidence Intervals

In this section, we will also be constructing 95% simultaneous confidence intervals for 4 pairwise comparisons and 2 contrasts with our best model of factor A and factor B without interaction from our analysis. We construct them to determine whether there is a statistically significant result for the average differences.

IV.C.1: Pairwise Confidence Intervals

The four pairwise comparisons of interest are as follows:

$$\mu_{San\ Francisco} - \mu_{Seattle}$$

$$\mu_{Data\ Scientist} - \mu_{Software\ Engineer}$$

$$\mu_{Data\ Scientst} - \mu_{Bioinformatics\ Engineer}$$

$$\mu_{Software\ Engineer} - \mu_{Bioinformatics\ Engineer}$$

where μ is the average of the annual salaries of the cities and profession. We will be using the sample mean as an estimate of the population mean.

To determine the multiplier we use when constructing our confidence intervals, we can calculate each one and then choose the smallest (Table 8.) We use the following formulas to calculate each multiplier:

Tukey multiplier = T = $\frac{1}{\sqrt{2}} q_{1-\alpha;\ a;\ nT-a}$ where q is the studentized range distribution

Scheffe multiplier = S = $\sqrt{(a\ -\ 1)F_{1-\alpha;\ a-1;\ nT-a}}$ where F is the F distribution

Bonferroni multiplier = B = $t_{1-\frac{\alpha}{(2g)};\ nT-a}$ where t is the student t distribution and g is the number of simultaneous comparisons.

with each α = 0.05, a = number of groups, and nT = total sample size in the data. After calculating for each group a = 2 for City and a = 3 for profession, we get:

Table 8: Multiplier

	Value
Tukey City	1.981
Tukey Profession	2.374
Scheffe City	1.981
Scheffe Profession	2.480
Bonferonni	2.537

We found the Tukey multiplier was the lowest for the profession factor and Tukey and Scheffe were both the lowest for the city factor. For simplicity, we will proceed in constructing our confidence intervals with the Tukey multiplier for both factors. Using the general formula

$$\overline{Y}_{ij.} - \overline{Y}_{i'j'.} \pm T\sqrt{MSE(1/n_{ij} + 1/n_{i'j'})}$$

where $\overline{Y}_{ij.}$ and $\overline{Y}_{i'j'.}$ are the two group means we are comparing it to for group i and j, T is the Tukey multiplier, and n_{ij} and $n_{i'j'}$ are the two group sample sizes. Our resulting pairwise confidence intervals are as follows:

Table 9: Tukey Pairwise Difference 95% Confidence Interval

	Estimate	Lower Bound	Upper Bound
San Francisco | Seattle	7.540	3.285	11.795
Data Science | Software Engineering	12.242	5.996	18.488
Data Science | Bioinformatics Engineer	34.061	27.815	40.307
Software Engineering | Bioinformatics Engineer	21.819	15.573	28.065

IV.C.2: Confidence Intervals for Contrasts

Next, we were interested in analyzing 95% confidence intervals for the following two contrasts:

$$\frac{\mu_{Data\ Scientist}+\mu_{Software\ Engineer}}{2} - \mu_{Bioinformatics\ Engineer}$$

$$\frac{\mu_{Bioinformatics\ Engineer}+\mu_{Software\ Engineer}}{2} - \mu_{Data\ Scientist}$$

where μ is the average of the annual salaries of the profession. We will also be using the sample marginal mean as an estimate for contrast confidence intervals.

Since we are constructing confidence intervals for contrasts, we can use either the Bonferroni or Scheffe multiplier. Using the formula we get:

Table 10: Multiplier

	Value
Scheffe	2.48
Bonferonni	2.54

We picked which one would give us the narrowest interval which is Scheffe. Using the Scheffe multiplier and the formula

$$\sum_i \sum_j c_{ij}\overline{Y}_{ij.} \pm S\sqrt{MSE(1/a^2)\sum_i \sum_j c_{ij}^2 / n_{ij}}$$

where c are contrast coefficients, a is the factor A levels which is 3 for profession., our resulting confidence intervals for contrasts are then

Table 11: Scheffe Contrast Difference 95% Confidence Interval

		Lower	Upper
	Estimate	Bound	Bound
Data Science + Software Engineering | Bioinformatics Engineer	27.940	22.290	33.590
Bioinformatics Engineer + Software Engineering | Data Science	-8.475	-14.125	-2.825

V. Interpretation: Key Findings from ANOVA

The Two-Factor ANOVA that we conducted on the salary dataset provided clear evidence of the significant effects that both profession and region have on the salaries of technology workers in Seattle and San Francisco. First, we confirmed that the assumptions of ANOVA were met: the data passed tests for normality (p-value of 0.324 from the Shapiro-Wilks test) and equal variance (p-value of 0.305 from the Brown-Forsythe test), allowing us to proceed with the ANOVA without the need for data transformation.

The results from our analysis indicated that both the profession and region had significant impacts on salary. When we tested for the interaction effect in the model, we found the p-value to be 0.053236, suggesting that interaction effect between profession and region was not statistically significant. We then proceed to test whether there is a factor A and B effect. The profession factor had an extremely significant p-value of less than 2×10^−16 indicating that salaries differ significantly across the three professions analyzed. The region factor also had a significant p-value of 0.000638, indicating a factor B effect between the two regions, San Francisco and Seattle. These results led to us rejecting the null hypotheses for both factors, concluding that there are factor A and B effects for the model and determined that to be the best fit model.

The pairwise and contrast comparisons further supported our findings. Using 0.05 as our alpha, we are 95% confident that our data lies in the differences. Data Scientists earned significantly more than both

Software Engineers and Bioinformatics Engineers. For example, the confidence interval for the salary difference between Data Scientists and Software Engineers did not include zero, supporting the conclusion that Data Scientists earn more. Similarly, the comparison between San Francisco and Seattle showed that workers in San Francisco earn significantly more than those in Seattle, with the confidence interval for this difference also excluding zero. These pairwise comparisons provided strong evidence of the disparities in salaries between the different professions and regions. In addition to pairwise comparisons, we also constructed contrasts confidence intervals to compare combinations of the professions and regions. For example, the contrast comparing the combined salaries of Data Scientists and Software Engineers against Bioinformatics Engineers provided further insight into how these groups differ in salary. It shows that Software Engineers and Data Scientist jobs have similar annual salaries. The confidence intervals for these contrasts were narrow and did not include zero, emphasizing the significant differences in salaries among the groups. Overall, the Two-Factor ANOVA analysis revealed that both profession and region are significant factors of salary in the technology industry.

VI. Conclusion

In conclusion, the Two-Factor ANOVA analysis resulted in us finding out that both profession and region significantly impact the salaries of technology workers in Seattle and San Francisco. The analysis showed that Data Scientists earn significantly more than Software Engineers and Bioinformatics Engineers, and that employees in San Francisco earn significantly more than those in Seattle. We also concluded that Software Engineers and Data Scientists annual salaries are similar to each other. The lack of a significant interaction effect suggests that these factors influence salaries independently, meaning that the effect of one factor does not depend on the level of the other. Our results provide a basis for understanding salary dynamics in the technology sector, emphasizing the significant role that both profession and region play in shaping salary outcomes for those in technology.

Appendix: R codes

Part 1:

```
library(ggplot2)
library(car)
library(dplyr)
library(MASS)

data <- read.csv("E:/Study/STA106/Project2/Helicopter.csv")

#boxplot
ggplot(data, aes(x = Shift, y = Count)) +
  geom_boxplot(fill = "lightblue") +
  labs(title = "Boxplot of Helicopter Calls by Shift",
       x = "Shift",
       y = "Number of Calls") +
  theme_minimal()

anovaModel <- aov(Count ~ Shift, data = data)


#diagnostics
par(mfrow = c(2, 2))
plot(anovaModel, which = 1)
plot(anovaModel, which = 2)


#shapiro test
shapiOrigin <- shapiro.test(residuals(anovaModel))
print(shapiOrigin)

#brown test
brownOrigin <- leveneTest(Count ~ Shift, data = data, center = median)
print(brownOrigin)


#remove outliers
dataa <- boxplot(data$Count ~ data$Shift)
outliers <- dataa$out
datanoout <- data[!(data$Count %in% outliers), ]



#boxcox transformation and check 0
boxResult <- boxcox(lm(Count ~ Shift, data = data))
boxLamda <- boxResult$x[which.max(boxResult$y)]
print(boxLamda)

boxResultnoout <- boxcox(lm(Count ~ Shift, data = datanoout))
boxLamdanoout <- boxResultnoout$x[which.max(boxResultnoout$y)]
print(boxLamdanoout)

data$boxCount <- (data$Count^boxLamda - 1) / boxLamda
datanoout$boxCount <- (datanoout$Count^boxLamdanoout - 1) / boxLamdanoout



# Diag plots without outliers
anovanoout <- aov(Count ~ Shift, data = datanoout)
par(mfrow = c(2, 2))
plot(anovanoout, which = 1)
plot(anovanoout, which = 2)
```

```
# Shapiro test for no outliers
shapironoout <- shapiro.test(residuals(anovanoout))
print(shapironoout)

# Brown test for no out
brownnoout <- leveneTest(Count ~ Shift, data = datanoout, center = median)
print(brownnoout)

#diag plots for Box-Cox
anovaBox <- aov(boxCount ~ Shift, data = data)
par(mfrow = c(2, 2))
plot(anovaBox, which = 1)
plot(anovaBox, which = 2)

# Shapiro for boxcox
shapirobox <- shapiro.test(residuals(anovaBox))
print(shapirobox)

#brown for box
brownbox <- leveneTest(boxCount ~ Shift, data = data, center = median)
print(brownbox)


#diag on Box no out
anovaBoxnoout <- aov(boxCount ~ Shift, data = datanoout)
plot(anovaBoxnoout, which = 1)
plot(anovaBoxnoout, which = 2)

# Shapiro box no out
shapiroBoxnoout <- shapiro.test(residuals(anovaBoxnoout))
print(shapiroBoxnoout)

#brown box no out
brownBoxnoout <- leveneTest(boxCount ~ Shift, data = datanoout, center =
median)
print(brownBoxnoout)
```

Part 2:

```
# II. Summary
# histograms, boxplots, sample means, standard deviations
salary <- read.csv("Salary.csv")
alpha <- 0.05

salary$Prof[salary$Prof == "DS"] <- "Data Scientist"
salary$Prof[salary$Prof == "SE"] <- "Software Engineering"
salary$Prof[salary$Prof == "BE"] <- "Bioinformatics Engineer"
salary$Region[salary$Region == "S"] <- "Seattle"
salary$Region[salary$Region == "SF"] <- "San Francisco"

salary_means <- tapply(salary$Annual, list(salary$Prof, salary$Region), mean)
means_region <- tapply(salary$Annual, salary$Region, mean)
means_prof <- tapply(salary$Annual, salary$Prof, mean)
means_overall <- mean(salary$Annual)

salary_means <- cbind(salary_means, means_prof)
salary_means <- rbind(salary_means, c(means_region, means_overall))
salary_means <- round(salary_means, 3)
Mean = as.vector(salary_means[1:3,1:2])
```

```r
salary_size <- tapply(salary$Annual, list(salary$Prof, salary$Region), length)
size_region <- tapply(salary$Annual, salary$Region, length)
size_prof <- tapply(salary$Annual, salary$Prof, length)
size_overall <- length(salary$Annual)

salary_size <- cbind(salary_size, size_prof)
salary_size <- rbind(salary_size, c(size_region, size_overall))
salary_size <- matrix(paste("(", salary_size, ")", sep=""), nrow =
nrow(salary_size), ncol = ncol(salary_size))

mean_size <- matrix(paste(salary_means, salary_size, sep=" "), nrow =
nrow(salary_size), ncol = ncol(salary_size))

salary_sd <- tapply(salary$Annual, list(salary$Prof, salary$Region), sd)
SD = as.vector(salary_sd)

colnames(mean_size) <- c("San Francisco", "Seattle", "A")
rownames(mean_size) <- c("Bioinformatics Engineer", "Data Scientist", "Software
Engineering","B")

kable(mean_size, caption = "Average annual salaries (in thousand of dollars)")
%>%
  column_spec(3, border_right = T) %>%
  row_spec(3, hline_after = T)
kable(round(salary_sd, 2), caption = "Standard deviation of annual salaries")


#Annual, Prof, Region
ggplot(data = salary, aes(Annual)) +
  geom_histogram(aes(fill = Prof), bins = 30) +
  #geom_point(aes(y = Prof)) +
  facet_grid(Prof ~ ., switch = "y") +
  labs(
    title = "Histogram of salaries by profession",
    x = "Annual salary (in thousands of dollars)",
    y = "Amount of subjects",
    fill = "Job"
  ) +
  theme_minimal()

ggplot(data = salary, aes(Annual)) +
  geom_histogram(aes(fill = Region), bins = 30) +
  facet_grid(Region ~ ., switch = "y") +
  labs(
    title = "Histogram of salaries by region",
    x = "Annual salary (in thousands of dollars)",
    y = "Amount of subjects",
    fill = "City"
  ) +
  theme_minimal()

ggplot(data = salary, aes(Annual, interaction(Region, Prof))) +
  geom_boxplot(aes(col = Region)) +
  labs(
    title = "Boxplot of salaries",
    x = "Annual salary (in thousands of dollars)",
    y = "Location & Job",
  ) +
  theme_minimal()

salary_data <- data.frame(
  City = rep(c("San Francisco", "Seattle"), each = 3),
  Profession = c("Bioinformatics Engineer", "Data Scientist", "Software
Engineering"),
  Mean = Mean,
  SD = SD
)

ggplot(salary_data, aes(Profession, Mean, fill = City), ) +
  geom_bar(stat = "identity", position = "dodge") +
  geom_errorbar(aes(ymin = Mean - SD, ymax = Mean + SD),
                position = position_dodge(0.9), width = 0.25) +
  labs(
    y = "Average annual salaries",
    title = "Bar graph of annual salaries (in thousand dollars)"
  )

Region <- salary$Region

interaction.plot(
  x.factor = salary$Prof,
  trace.factor = Region,
  response = salary$Annual,
  col = c("blue", "red"),  # Colors for trace factor
  lty = 1:2,               # Line types for trace factor
  pch = c(19, 17),         # Point symbols for trace factor
  main = "Interaction Plot", # Title
  xlab = "Profession",
  ylab = "Annual salary (in thousand of dollars)",
  legend = T,           # Show legend
  lwd = 2,                 # Line width
)

# III. Diagnostics
# Normality Test
salary_model = lm(Annual ~ Prof * Region, data = salary)
ei <- salary_model$residuals

par(mfrow = c(1, 1))
qqnorm(ei)
qqline(ei)

# Shapiro-Wilk test
SWtest = shapiro.test(ei)

# Equal Variance Test
plot(salary_model$fitted.values, salary_model$residuals, xlab = "Fitted
values",
     ylab = "Residuals")
abline(h = 0, col = "red")
title("Group means vs Residuals")

# Brown-Forsythe test
the.BFtest = leveneTest(Annual ~ Prof * Region, data = salary, center=median)
p.val = the.BFtest[[3]][1]


# P value < alpha = not normal.
p_value <- rbind(SWtest$p.value, p.val)
diagnostics <- format.pval(p_value, 3)
diagnostics <- rbind(diagnostics[1], diagnostics[2])
colnames(diagnostics) <- c("p value")
rownames(diagnostics) <- c("Shapiro-Wilk", "Brown-Forsythe")

kable(diagnostics, caption = "Normality and Equal Variance Test")
par(mfrow = c(1, 1))
```

```
# Outliers via Semi-studentized/standardized residuals
SSE <- sum((salary_model$residuals)^2)
MSE <- SSE / salary_model$df.residual

eij.star <- salary_model$residuals / sqrt(MSE)
g <- 5 # Account for 5 outliers
t.cutoff= qt(1-alpha/(2*g), salary_model$df.residual)
CO.eij = which(abs(eij.star) > t.cutoff)
CO.eij

# studentized/standardized residuals
rij = rstandard(salary_model)
CO.rij = which(abs(rij) > t.cutoff)
#CO.rij

outliers = CO.rij
new.data = salary[-outliers,]
# no outliers were detected.
cat("No outliers were detected using semi-studentized/standardized residuals")

# Effect sizes
eta_result <- eta_squared(salary_model, partial = FALSE)
colnames(eta_result)[3] <- "95% CI"
colnames(eta_result)[4] <- "Lower Bound"
colnames(eta_result)[5] <- "Higher Bound"
eta_result[1] <- c("Profession", "Region", "Interaction")
eta_result[2] <- round(eta_result[2], 2)
eta_result[4] <- round(eta_result[4], 2)
kable(eta_result, caption = "Effect sizes for ANOVA")

# IV. Analysis
# Conditional / Partial R^2 & Formal Test
cond_R2 <- function(full, reduced) {
  return((reduced - full) / reduced)
}

F_stat <- function(full, reduced, df_F, df_R) {
  MSE_F <- full / df_F
  (reduced - full) / (df_R - df_F) / MSE_F
}

# Factor A as Profession, Factor B as Region
AB <- lm(Annual ~ Prof * Region, salary)
A.B <- lm(Annual ~ Prof + Region, salary)
A <- lm(Annual ~ Prof, salary)
B <- lm(Annual ~ Region, salary)

e.mean <- mean(salary$Annual)
e.SSE <- sum((salary$Annual - e.mean)^2)
nT <- nrow(salary)

models <- list(AB = AB, "A + B" = A.B, A = A, B = B)
models_df <- c(AB$df.residual, A.B$df.residual, A$df.residual, B$df.residual,
nT - 1)
SSE <- sapply(models, function(model) sum(model$residuals^2))
SSE <- c(SSE, e.SSE)


R_cond <- matrix(ncol = length(SSE))
pattern_full <- c(1, 2, 2, 3, 4)
pattern_reduced <- c(2, 4, 3, 5, 5)

R_cond <- c()
```

```r
F_test <- c()
p_value <- c()

for (i in 1:length(SSE)) {
  R_cond <- cbind(R_cond, cond_R2(SSE[pattern_full[i]],
SSE[pattern_reduced[i]]))
  F_test <- cbind(F_test, F_stat(SSE[pattern_full[i]], SSE[pattern_reduced[i]],
models_df[pattern_full[i]],
                                models_df[pattern_reduced[i]]))
  p_value <- cbind(p_value, 1 -pf(F_test[i], models_df[pattern_reduced[i]] -
models_df[pattern_full[i]],
                                 models_df[pattern_full[i]]))
}

R_cond <- round(R_cond, 4) * 100
R_cond <- sapply(R_cond, function(values) paste(values, "%", sep=""))
F_test <- round(F_test, 2)
p_value <- format.pval(p_value, digits = 3)
test_result <- rbind(R_cond)
header <- c("(AB | A + B)", "(A + B | B)", "(A + B | A)", "(A |  ·)", "(B |  ·)")
colnames(test_result) <- header
rownames(test_result) <- c("Percentages")

test_result2 <- rbind(F_test, p_value)
colnames(test_result2) <- header
rownames(test_result2) <- c("F Statistic", "p value")

MSE <- round(SSE / models_df, 2)
MSE_m <- t(matrix(c("Mean of Squared Errors", MSE)))
SSE <- round(SSE, 2)
SSE_m <- t(matrix(c("Sum of Squared Errors", SSE), byrow = T))
df_m <- t(matrix(c("Degrees of Freedom", models_df)))
colnames(SSE_m) <- c("Values", "AB", "A + B", "A", "B", "·")


kable(rbind(MSE_m, SSE_m, df_m), caption = "Model Summary")
kable(test_result, caption = "Conditional R²",
      table.attr = 'style="text-align: center;"')
kable(test_result2, caption = "Factor Effect Test")

# IV.II Analysis
# consider 6 confidence intervals total, 4 of which are pairwise, and two of
which are contrasts
best_model <- A.B
MSE <- sum((best_model$residuals)^2) / best_model$df.residual

nt <- nrow(salary)
a <- length(unique(salary$Prof))
b <- length(unique(salary$Region))
g <- 4 # 4/2 for pairwise & contrast
df_SSE <- best_model$df.residual

get_tukey <- function(amount_factor) {
  return(qtukey(1-alpha,amount_factor, df_SSE)/sqrt(2)) # Pairwise comparison
}
get_scheffe <- function(amount_factor) {
  return(sqrt((amount_factor-1)*qf(1-alpha, amount_factor-1, df_SSE))) #
Contrast comparison
}
get_bonf <- function(g) {
  return(qt(1-alpha/(2*g), df_SSE)) # Can be used anywhere
}

give.me.CI = function(ybar,ni,ci,MSE,multiplier, amount_factor) {
  if(sum(ci) != 0 &sum(ci !=0 ) != 1 ) {
    return("Error - you did not input a valid contrast")
  } else if(length(ci) != length(ni)) {
    return("Error - not enough contrasts given")
  } else {
      estimate = sum(ybar*ci)
      SE = sqrt(MSE*(1/amount_factor^2)*sum(ci^2/ni))
      CI = estimate + c(-1,1)*multiplier*SE
      result = c(estimate,CI)
      names(result) = c("Estimate","Lower Bound","Upper Bound")
      return(result)
  }
}

city_factor = 2
prof_factor = 3
g = 4
ci <- c(1, -1)

mult <- c(get_tukey(city_factor), get_tukey(prof_factor),
          get_scheffe(city_factor), get_scheffe(prof_factor), get_bonf(g))
mult <- round(mult, 3)
names(mult) <- c("Tukey City", "Tukey Profession", "Scheffe City", "Scheffe
Profession", "Bonferonni")
mult_df <- data.frame( # Changing to df for naming.
  Value = mult
)
kable(mult_df, caption = "Multiplier")

ni <- c(60, 60) # Since it is a balanced design
# Between Cities (SF vs S)
city_avg <- salary_means[4,1:2]
SF_S <- give.me.CI(city_avg, ni, ci, MSE, get_tukey(city_factor), city_factor)

ni <- c(40, 40)
# Between Profession
# DS vs SE
DS_SE <- c(salary_means[2,3], salary_means[3,3])
DS_SE <- give.me.CI(DS_SE, ni, ci, MSE, get_tukey(prof_factor), prof_factor)

# DS vs BE
DS_BE <- c(salary_means[2,3], salary_means[1,3])
DS_BE <- give.me.CI(DS_BE, ni, ci, MSE, get_tukey(prof_factor), prof_factor)

# SE vs BE
SE_BE <- c(salary_means[3,3], salary_means[1,3])
SE_BE <- give.me.CI(SE_BE, ni, ci, MSE, get_tukey(prof_factor), prof_factor)

pair_wise <- rbind(SF_S, DS_SE, DS_BE, SE_BE)
pair_wise <- round(pair_wise, 3)
rownames(pair_wise) <- c("San Francisco        | Seattle",
                         "Data Science         | Software Engineering",
                         "Data Science         | Bioinformatics Engineer",
                         "Software Engineering | Bioinformatics Engineer")
kable(pair_wise, caption = "Tukey Pairwise Difference 95% Confidence Interval")



# Contrast
mult <- c(get_scheffe(prof_factor), get_bonf(g))
mult <- round(mult, 2)
names(mult) <- c("Scheffe", "Bonferonni")
mult_df <- data.frame( # Changing to df for naming.
  Value = mult
```

```
)
kable(mult_df, caption = "Multiplier")

g = 2
ni <- c(40, 40, 40)
ci <- c(1/2, 1/2, -1)

# DS + SE vs BE
DSE_BE <- c(salary_means[2,3], salary_means[3,3], salary_means[1,3])
DSE_BE <- give.me.CI(DSE_BE, ni, ci, MSE, get_scheffe(prof_factor),
prof_factor)

# BE + SE - DS
BES_DS <- c(salary_means[1,3], salary_means[3,3], salary_means[2,3])
BES_DS <- give.me.CI(DSE_BE, ni, ci, MSE, get_scheffe(prof_factor),
prof_factor)

contrast <- rbind(DSE_BE, BES_DS)
contrast <- round(contrast, 3)
rownames(contrast) <- c("Data Science + Software Engineering | Bioinformatics
Engineer",
                        "Bioinformatics Engineer + Software Engineering | Data
Science")
kable(contrast, caption = "Scheffe Contrast Difference 95% Confidence
Interval")
```