Assignment 1 Solution

Exploratory Data Analysis

Sina Bahrami

Part 1: Data

Description of the Data Set

The input data frame is obtained from Kaggle [1] and converted to tibble data type. It represents tech salaries. It has 1655 observations for 19 variables. The variables are index, salary id, employer name, location name, location state, location country, location latitude, location longitude, job title, job title category, job title rank, total experience years, employer experience years, annual base pay, signing bonus, annual bonus, stock value bonus, comments, submitted at. There are many problems in the data including missing values, inconsistent names for cities, etc. In the following section the cleaning process of the data is explained.

Data Cleaning

- The second part of some city names (state names after commas) are removed.
- Modifications are made to some location name like nyc to New York.
- A review of the data set reveals city names are not provided in a consistent way; many location longitude and latitude values are missing although the name of the city is available; Country name abbreviations are also missing for some of them. All these missing data are recovered by using a data set at Kaggle [2], which includes information of all world cities.
- The numeric job rank 3, 4 are converted to Senior and the 1 and 2 are converted to Junior.
- Senior, Junior, Jr, Sn are removed from job titles as they are already included in job title rank. Also the titles are trimmed and changed to title case.
- Date types are recognized in R and numbers are converted to integers where decimals are not required.
- Also further types of some columns are modified;
- The name of the companies are changed to title case;
- Stock value bonus are converted to numbers using parse_number command as some of them were provided in non-numeric format like 80k.

The focus in this analysis is on annual base pay. The summary of this variable is provided below. As shown, there are outliers in the annual base pay data that need to be removed. Also the annual values below 5000 are intuitively removed from the data set as they do not seem valid.

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.     NA's
## 0.00e+00 6.10e+04 9.93e+04 2.68e+05 1.30e+05 1.56e+08        5
```

The histogram also shows the distribution of the annual base pay for all of the given countries and job categories. The distribution is right skewed and does not seem to be a normal distribution.

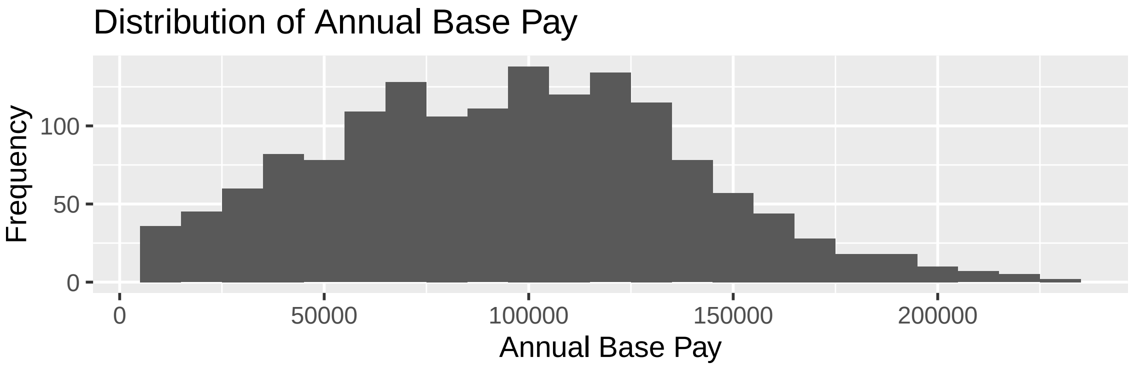
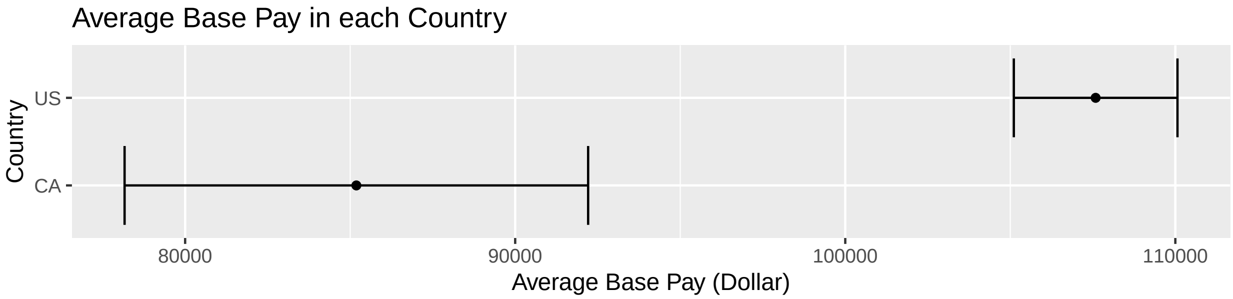


Figure 1: Annual base pay data summary and outliers

Part 2: Planning and Analysis

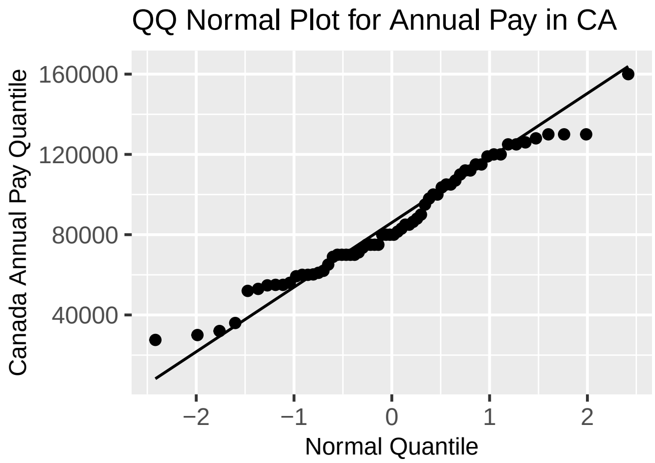
In this part an idea based on the observation is developed and formulated as a hypothesis. First overall observation of the data is carried out by creating various plots. Many of these plots are provided in the Appendix. The main observation that starts the research was the distribution of average annual salary for all of countries. Among the highest were US and Canada. As demonstrated on the world map in the appendix, it seems US has higher average than Canada. To better examine this idea, the average annual pay and their confidence intervals are visualized in the following chart.

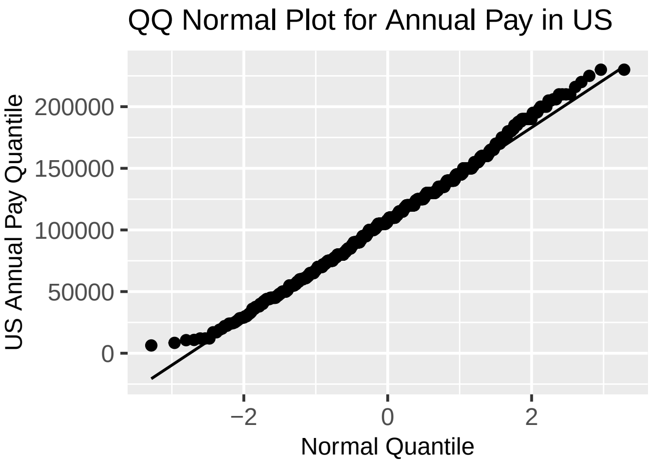


The plot of average annual pay and their CIs for both US and Canada, suggests the salaries in US are higher than Canada in average with no interference of CIs. So this subject is investigated in this analysis. The idea is formulated in form of the following hypothesis:

H_0: Average annual base pay for US and Canada are equal
H_1: Average annual base pay in US is higher than Canada

For this purpose t-test is conducted, but we need to ensure normality of data first. Therefore, first quantile quantile normal plots for annual base pay in both countries are visualized, then Shapiro-wilk tests are performed.


QQ Normal Plot for Annual Pay in CA


QQ Normal Plot for Annual Pay in US

The qq noraml plot suggest some level of normality. However, to ensure this, Shapiro-Wilk test is performed too. The p-value for Canada is 0.211532502904976 and for US is 0.011832951826525. So for US normality is rejected at $\alpha = 0.05$ but for Canada there is not enough evidence. However, both qq normal charts are close to straight lines and the sample sizes (64, 984) are large enough, so a t-test is conducted based on the current sample data.

The results of the test for both samples show that for US normality is rejected but for Canada there is not enough evidence to reject at $\alpha = 0.05$.

It is also required to check equality of variances. If they are very different the Welch t-test needs to be used instead of student t-test. For equality of variances F-test is used. The p-value is 0.00152941752518501. So, the equality of variances is also rejected. So Welch t-test is performed.

The p-value of the one-sided t-test is 4.3095180879273e-08, so H_0 is not rejected i.e. there is not enough evidence that the average of annual base pay in Canada is less than US.

Part 3: Conclusion

The input data for tech salaries required lots of cleaning steps like correction of city names, job titles, rank and many of the data were missing, although some of the data were restored. The first observation of the data on the world map revealed a higher average in US compared to Canada. This was further investigated by visualizing averages and CIs. Based on that, a hypothesis was formed that the average annual base pay in US is higher than Canada, however, later, the null hypothesis could not be rejected. It is also worth mentioning that normality of annual base pay of US sample was rejected, but based on straight quantile quantile normal plots as well as relatively large sample sizes, t-test was performed.

References

1. Telle, Brandon. “Tech Salaries.” Kaggle, 4 Dec. 2022, https://www.kaggle.com/datasets/thedevastator/know-your-worth-tech-salaries-in-2016.

2. SimpleMaps.com. “World Cities Database.” Kaggle, 26 Mar. 2022, https://www.kaggle.com/datasets/juanmah/world-cities.

Appendix

Codes

```
knitr::opts_chunk$set(echo = TRUE, message=FALSE, Warning=FALSE,
                      fig.width=3, fig.height=2)
options(digits=2)
remove(list=ls())
library(readr)
library(tibble)
library(dplyr)
library(ggplot2)
library(magrittr)
library(tidyverse)
library(stringdist)
library(pastecs)
library(car)
# Takes a column and detects outliers
detect_outlier <- function(x) {

  Quantile1 <- quantile(x, probs=.25, na.rm = TRUE)
  Quantile3 <- quantile(x, probs=.75, na.rm = TRUE)
  IQR = Quantile3-Quantile1
  x > Quantile3 + (IQR*1.5) | x < Quantile1 - (IQR*1.5)
}

# Takes a data frame and a column and removes the outliers from the
# original data frame
remove_outlier <- function(dataframe,
                           columns=names(dataframe)) {
  # for loop to traverse in columns vector
  for (col in columns) {
    # remove observation if it satisfies outlier function
    dataframe <- dataframe[!detect_outlier(dataframe[[col]]), ]
  }
  return(dataframe)
}


setwd("C:/Users/sinab/OneDrive/MMSc/MSCI 718/Session 3/Assignment 1")

# Salary data set is read and for some variables the correct type is preset.
data_types <- c("index"="integer", "salary_id"="integer",
                "annual_base_pay" = "numeric",
                "total_experience_years" = "numeric",
                "employer_experience_years" = "numeric",
                "annual_bonus"="numeric", "annual_base_pay"="numeric",
                "stock_value_bonus"="numeric", "signing_bonus"="numeric")
df_cities <- as_tibble(read_csv(file = "worldcities.csv",
                                show_col_types = FALSE))

# The reference data set for information of cities is read for recovery of the
```

```r
# missing or non-standard information in the salary data set
df_jobs <- as_tibble(read_csv(file = "salaries_clean.csv",
                              show_col_types = FALSE, col_types = data_types))
```

```
## Warning: One or more parsing issues, call ‘problems()‘ on your data frame for details,
## e.g.:
##   dat <- vroom(...)
##   problems(dat)
```

```r
# The second part of some city names (state names) are removed.

df_jobs$location_name %<>% str_split_i(",", 1)
df_jobs$location_name %<>% str_trim()

# modification of location names

nyc_variants <- c("new york city", "nyc")
df_jobs <- df_jobs %>%
  mutate(location_name = ifelse(str_to_lower(location_name) %in%
                                  nyc_variants, "New York", location_name))

# Name of the cities in the salary list are matched by max 1 character distance
# with those in the reference data set of cities and their index is stored

city_index <- amatch(str_to_lower(df_jobs$location_name),
                     str_to_lower(df_cities$city_ascii),
                     nomatch = NA,
                     maxDist = 1)
# The recovered city names and their associated longitude and latitude are
# stored in the salary data set.

df_jobs %<>%
  mutate(location_name =
           case_when(not(is.na(city_index)) ~
                       df_cities$city_ascii[city_index]),
         location_country =
           case_when(not(is.na(city_index)) ~ df_cities$iso2[city_index],
         is.na(city_index) ~ location_country),
         location_latitude =
           case_when(not(is.na(city_index)) ~ df_cities$lat[city_index],
         is.na(city_index) ~ location_latitude),
         location_longitude =
           case_when(not(is.na(city_index)) ~ df_cities$lng[city_index],
         is.na(city_index) ~ location_longitude))

# The numeric job rank 3, 4 are converted to Senior and the 1 and 2 are
# converted to Junior

df_jobs %<>%
  mutate(job_title_rank =
           case_when(
             job_title_rank=="1" | job_title_rank=="2" ~ "Junior",
             job_title_rank=="3" | job_title_rank=="4" ~ "Senior"))
```

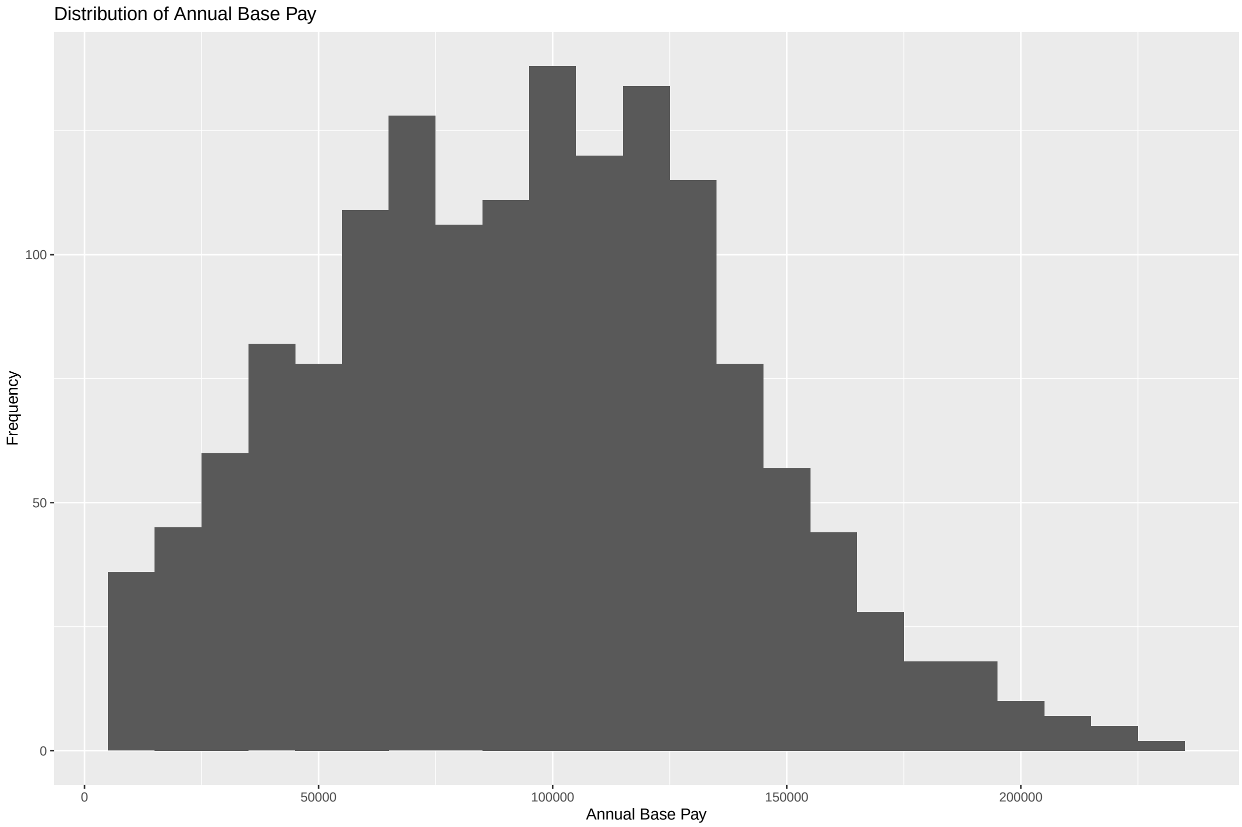
```
# Senior, Junior, Jr, Sn are removed from job titles as they are already
# included in job title rank. Also the titles are trimmed and changed to title
# case

df_jobs <- df_jobs %>%
  mutate(job_title = str_remove( str_to_lower(job_title), "senior")) %>%
  mutate(job_title = str_remove( str_to_lower(job_title), "junior")) %>%
  mutate(job_title = str_remove( str_to_lower(job_title), "jr")) %>%
  mutate(job_title = str_remove( str_to_lower(job_title), "sr")) %>%
  mutate(job_title = str_trim(str_to_title(job_title)))


# Based on the index of cities in the reference data set, the correct values
# from the reference data set are restored into the salary data set. with some
# further data type modification.

df_jobs %<>%
  mutate(stock_value_bonus = as.integer(stock_value_bonus),
         employer_name = str_to_title(employer_name),
         annual_base_pay = as.integer(annual_base_pay),
         total_experience_years = as.integer(total_experience_years),
         job_title = str_to_title(job_title),
         submitted_at = as.POSIXct(str_c(as.character(submitted_at),":00"),
                                   tz="", tryFormats="%m/%d/%y %H:%M:%OS"))
```

```
## Warning in mask$eval_all_mutate(quo): NAs introduced by coercion to integer
## range
```

```
# Exploring annual base pay distribution and its normality

summary(df_jobs$annual_base_pay)
```

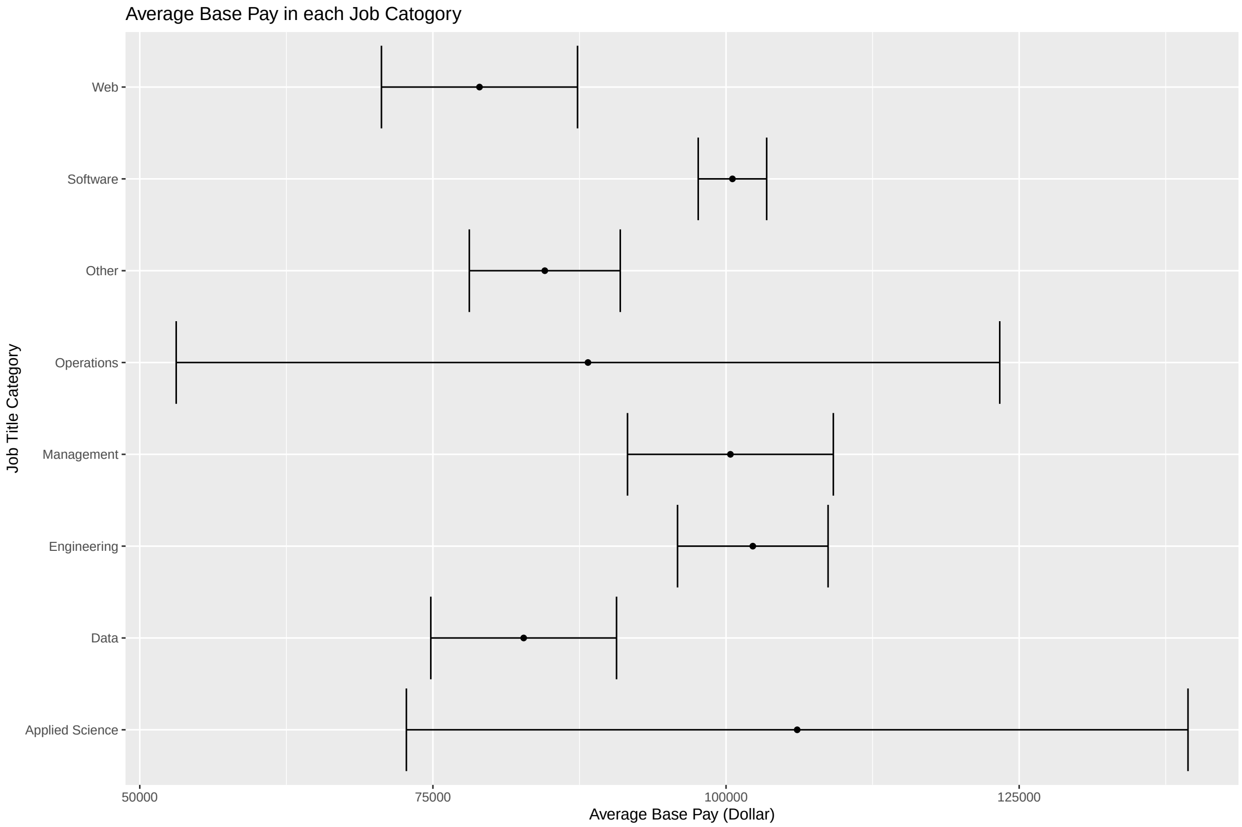
```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.     NA’s
## 0.00e+00 6.10e+04 9.93e+04 2.68e+05 1.30e+05 1.56e+08        5
```

```
df_jobs_nooutlier <- remove_outlier(df_jobs, "annual_base_pay")

# Despite removing outliers there are still some very low annual base pay.
# Based on intuition, annual base pay below 5000 are removed too.

df_jobs_nooutlier %<>% filter(annual_base_pay>5000)
ggplot(df_jobs_nooutlier) +
  geom_histogram(aes(annual_base_pay), binwidth=10000, na.rm=TRUE) +
  labs(x="Annual Base Pay", y="Frequency",
       title="Distribution of Annual Base Pay")
```



```
# Obtaining average base pay for each job category and their s.e. for
# evaluation of confidence interval and plotting average and confidence
# interval for average of annual base pay of each job title category

df_job_pay <- df_jobs_nooutlier %>%
  filter(!is.na(annual_base_pay) & !is.na(total_experience_years) &
           !is.na(job_title_category)) %>%
  group_by(job_title_category) %>%
  summarize(n_observation = n(), avg_base_pay=mean(annual_base_pay),
            se=sd(annual_base_pay)/sqrt(n()))

df_job_pay %>%
  ggplot(aes(x=avg_base_pay, y=job_title_category)) +
  geom_point() +
  geom_errorbar(aes(x=avg_base_pay, xmax=avg_base_pay+1.96*se,
                    xmin=avg_base_pay-1.96*se, fill = job_title_category)) +
  labs(x = "Average Base Pay (Dollar)", y = "Job Title Category",
       title = "Average Base Pay in each Job Catogory")
```

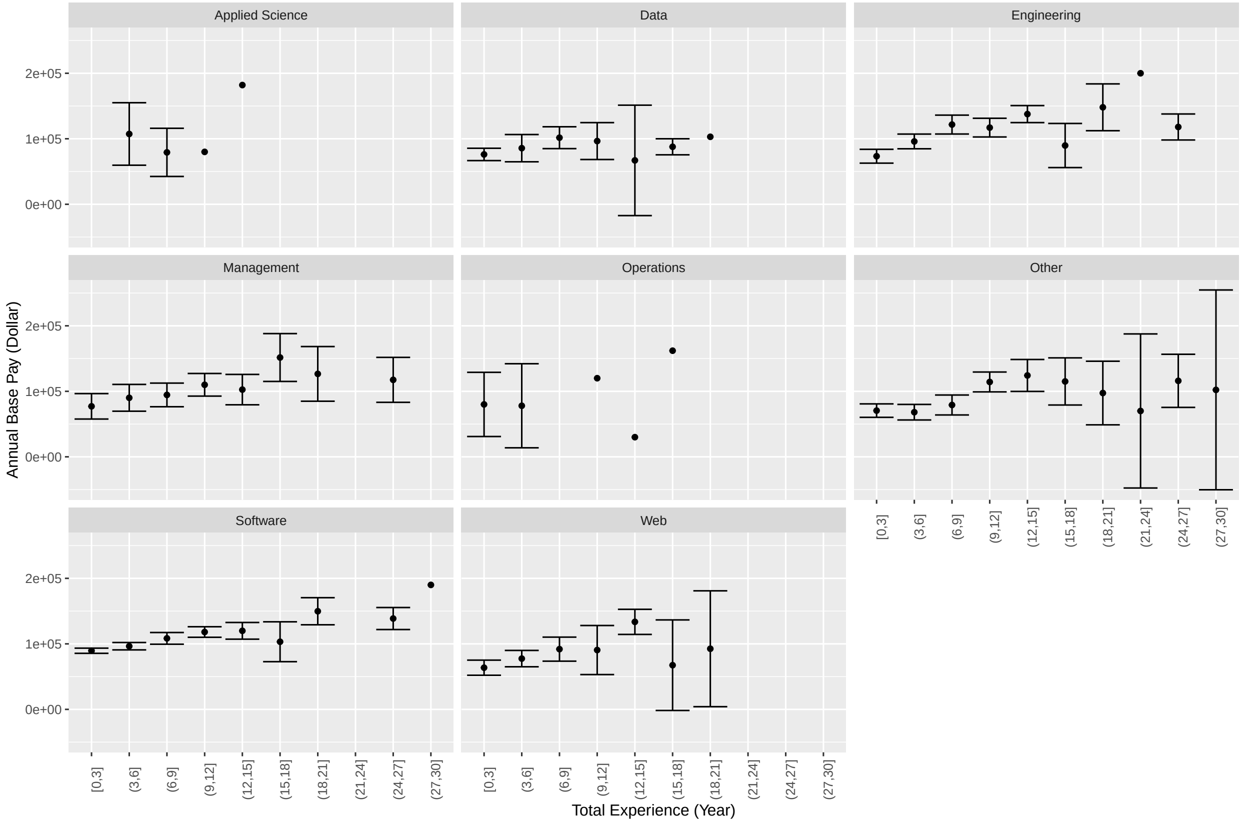
```
## Warning in geom_errorbar(aes(x = avg_base_pay, xmax = avg_base_pay + 1.96 * :
## Ignoring unknown aesthetics: fill
```



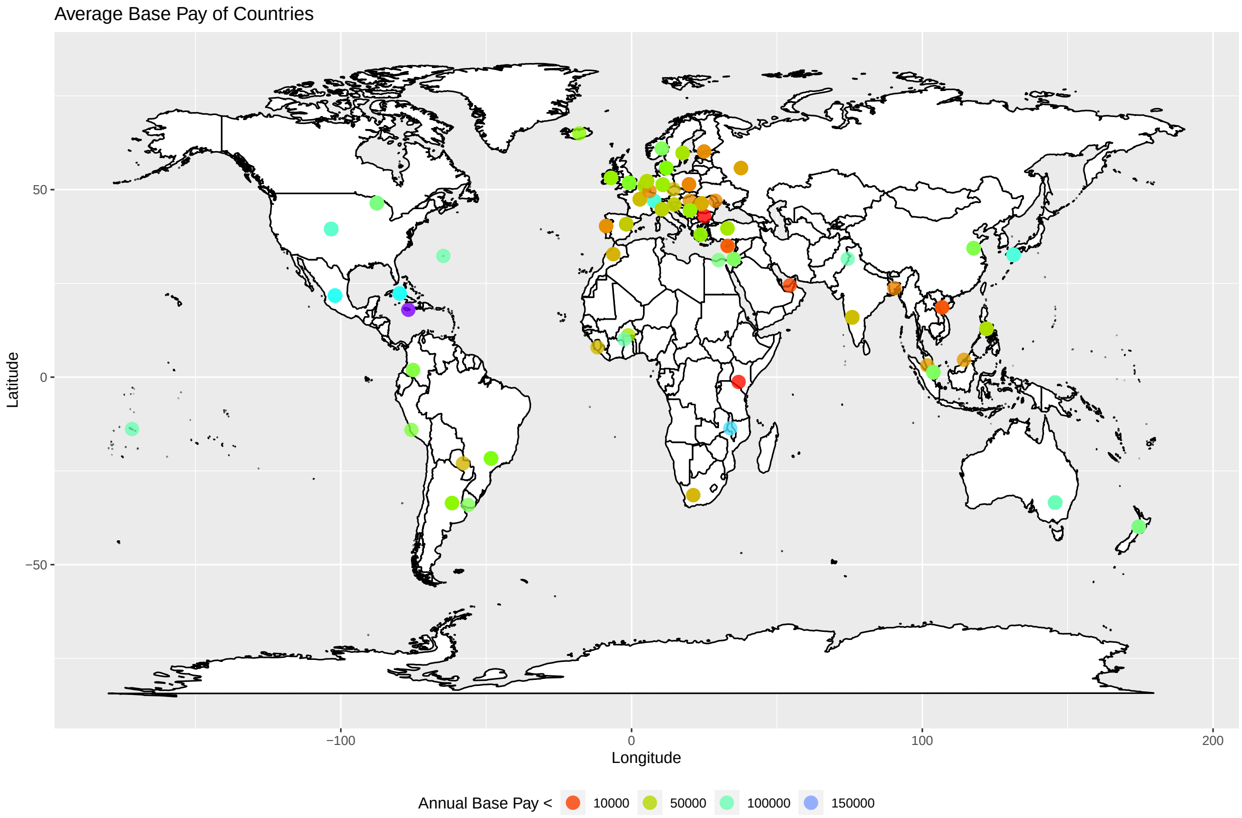
```
# Plotting annual base pay for each 3-year intervals of total work experience
# in the range of [0,30] for each job title category after removing outliers of
# pay

summary(df_jobs$total_experience_years)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##       0       3       5       7      10      56      47
```

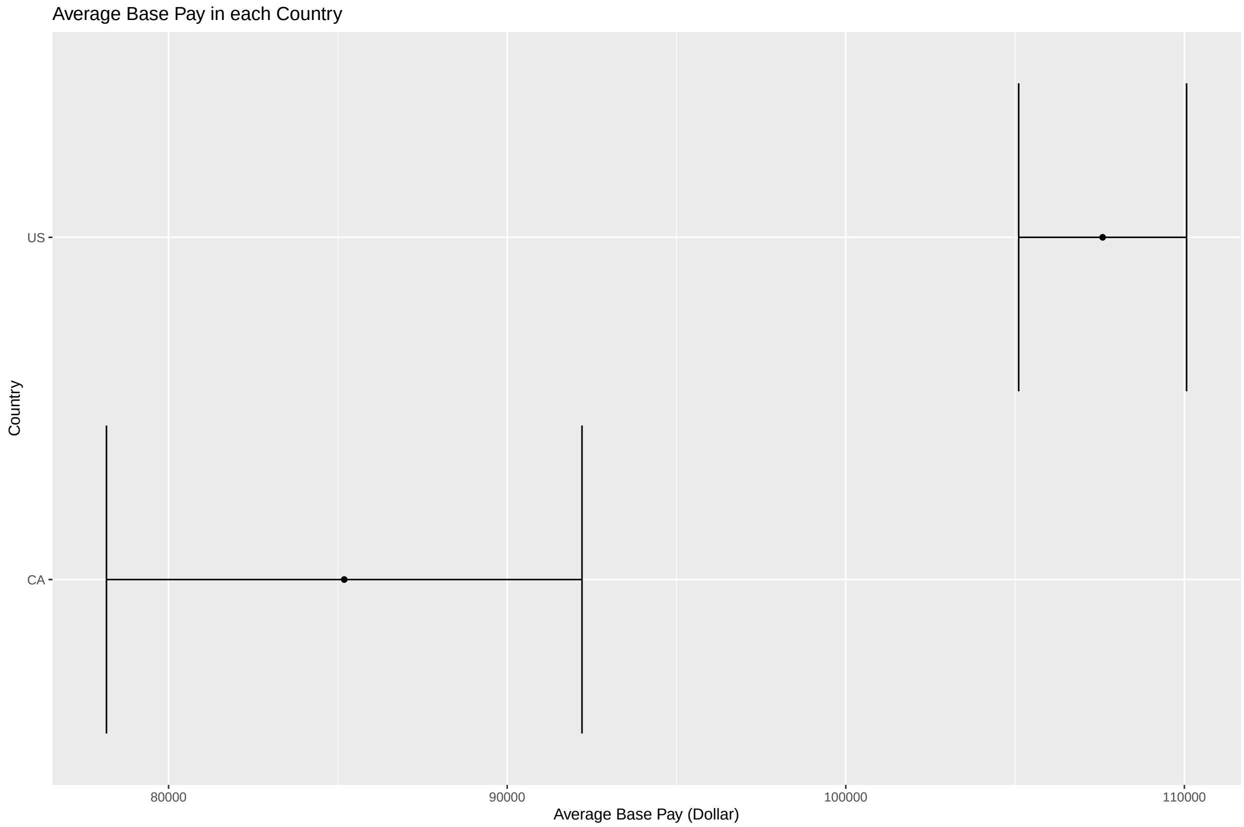
```
df_jobs_nooutlier %>%
  select(total_experience_years, annual_base_pay, job_title_category) %>%
  filter(!is.na(annual_base_pay) & !is.na(total_experience_years) &
           !is.na(job_title_category) & total_experience_years <= 30) %>%
  mutate(experience_ranges=cut(total_experience_years, seq(0,30,3),
                               include.lowest=TRUE)) %>%
  group_by(experience_ranges, job_title_category) %>%
  summarize(avg = mean(annual_base_pay),
            se=sd(annual_base_pay)/sqrt(n())) %>%
  ggplot() +
  geom_point(aes(x=experience_ranges, y=avg)) +
  geom_errorbar(aes(x=experience_ranges, y=avg,
              ymin=avg-1.96*se,
              ymax=avg+1.96*se)) +
  facet_wrap(vars(job_title_category)) +
  guides(color = guide_legend(title = "Job Title Category")) +
  labs(x="Total Experience (Year)", y="Annual Base Pay (Dollar)") +
  theme(axis.text.x=element_text(angle=90))
```



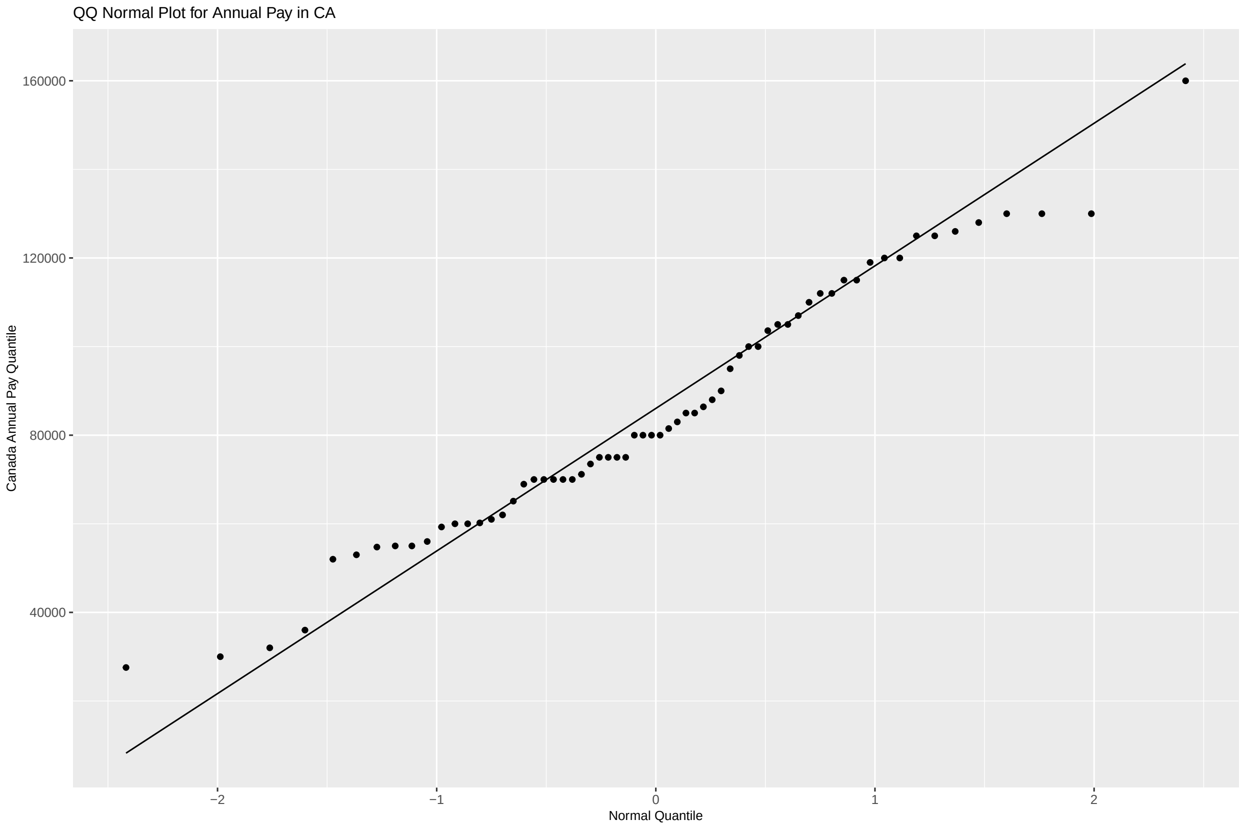
```
df <- df_jobs_nooutlier %>%
  select(annual_base_pay, job_title_category) %>%
  filter(str_to_lower(job_title_category) %in%
           c("engineering", "management")) %>%
  filter(!is.na(annual_base_pay))

ttest_res <-  t.test(str_to_lower(df$job_title_category) == "management",
                     str_to_lower(df$job_title_category) == "engineering",
                     conf.level=0.95)
df_countries <- df_jobs_nooutlier %>%
  select(location_country, location_longitude,
         location_latitude, annual_base_pay) %>%
  group_by(location_country) %>%
  summarize(avg=mean(annual_base_pay),
            long=mean(location_longitude), lat=mean(location_latitude),
            country=location_country)

worldmap <- map_data("world")
ggplot(data=worldmap) +
  geom_polygon(aes(x=long, y=lat, group=group), fill="white", color="black") +
  geom_point(data=na.omit(df_countries), aes(x=long, y=lat,
                color=avg), size=4, alpha=0.8) +
  scale_color_gradientn(colors = rainbow(4),
                        breaks=c(10000,50000,100000,150000,220000)) +
  labs(x="Longitude", y="Latitude",
       title="Average Base Pay of Countries") +
  guides(color = guide_legend(title = "Annual Base Pay <")) +
  theme(legend.position = "bottom")
```



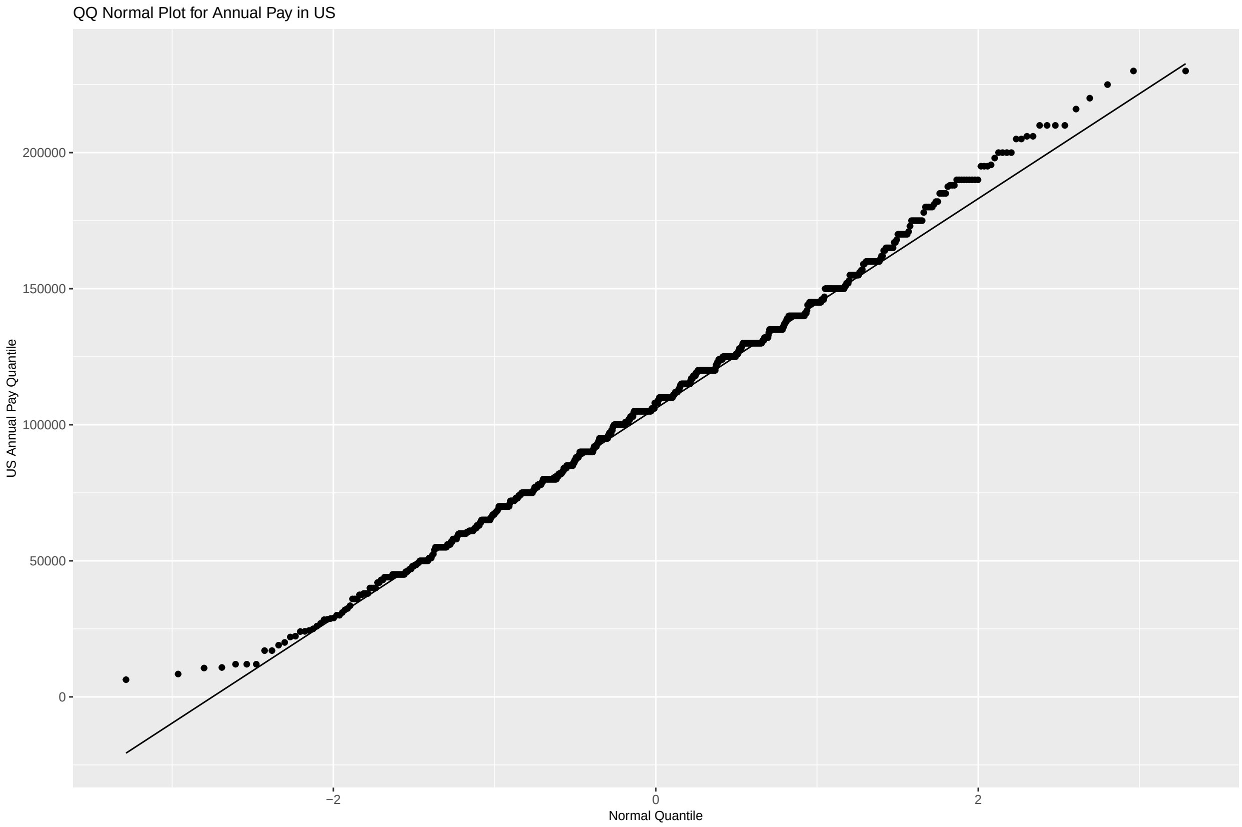
```
# Obtaining average base pay for US and Canada
USvsCA <- df_jobs_nooutlier %>%
  select(location_country, annual_base_pay) %>%
  na.omit(data=.data) %>%
  filter(location_country %in% c("CA","US"))

USvsCA %>%
  group_by(location_country) %>%
  summarize(n_observation = n(), avg=mean(annual_base_pay),
            se=sd(annual_base_pay)/sqrt(n())) %>%
  ggplot(aes(x=avg, y=location_country)) +
  geom_point() +
  geom_errorbar(aes(x=avg, xmax=avg+1.96*se, xmin=avg-1.96*se)) +
  labs(x = "Average Base Pay (Dollar)", y = "Country",
       title = "Average Base Pay in each Country")
```



```
USvsCA %>%
  filter(location_country=="CA") %>%
  ggplot(aes(sample=annual_base_pay)) +
  geom_qq() +
  geom_qq_line() +
  labs(x="Normal Quantile", y="Canada Annual Pay Quantile",
       title="QQ Normal Plot for Annual Pay in CA") +
  theme(title = element_text(size=9))
```



```
USvsCA %>%
  filter(location_country=="US") %>%
  ggplot(aes(sample=annual_base_pay)) +
  geom_qq() +
  geom_qq_line() +
  labs(x="Normal Quantile", y="US Annual Pay Quantile",
       title="QQ Normal Plot for Annual Pay in US") +
  theme(title = element_text(size=9))
```



```r
CA <- USvsCA %>%
  filter(location_country=="CA")
CA_res <- shapiro.test(CA$annual_base_pay)

US <- USvsCA %>%
  filter(location_country=="US")
US_res <- shapiro.test(US$annual_base_pay)
x <- as.vector(CA$annual_base_pay)
y <- as.vector(US$annual_base_pay)
ttest_res <- t.test(x, y, var.equal=FALSE, alternative="less")
```