PSTAT 131 Final Project: Model comparison for predicting the salary of a data science job

Nicholas Axl Andrian

2023-11-23

Dataset used: https://www.kaggle.com/datasets/arnabchaki/data-science-salaries-2023

In this project, we will be fitting several different machine learning algorithms to find out which method of prediction is the most accurate in getting the predicted salary(in usd).

About the data set's variables (excerpt from the kaggle site)

- work_year: The year the salary was paid.
- experience_level: The experience level in the job during the year
- employment_type: The type of employment for the role
- job_title: The role worked in during the year.
- salary: The total gross salary amount paid.
- salary_currency: The currency of the salary paid as an ISO 4217 currency code.
- salaryinusd: The salary in USD
- employee_residence: Employee's primary country of residence in during the work + year as an ISO 3166 country code.
- remote_ratio: The overall amount of work done remotely
- company_location: The country of the employer's main office or contracting branch
- company_size: The median number of people that worked for the company during the year

```r
library(dplyr)
library(randomForest)
library(gbm)
library(ISLR)
library(tree)
library(tidyverse)
library(ggplot2)
library(gridExtra)
```

Part 1: Exploratory Data Analysis

Loading the dataset

```r
salaries <- read.csv("ds_salaries.csv")
```

Checking the structure of the dataset

```r
options("max.print" = 5) # to prevent page number bloat, remove for full output
head(salaries)
```

```
##      work_year experience_level employment_type job_title salary
##      salary_currency salary_in_usd employee_residence remote_ratio
##      company_location company_size
##  [ reached 'max' / getOption("max.print") -- omitted 6 rows ]
```

```r
str(salaries)
```

```
## 'data.frame':    3755 obs. of  11 variables:
##  $ work_year         : int  2023 2023 2023 2023 2023 2023 2023 2023 2023 2023 ...
##  $ experience_level  : chr  "SE" "MI" "MI" "SE" ...
##  $ employment_type   : chr  "FT" "CT" "CT" "FT" ...
##  $ job_title         : chr  "Principal Data Scientist" "ML Engineer" "ML Engineer" "Data Scientist"
##  $ salary            : int  80000 30000 25500 175000 120000 222200 136000 219000 141000 147100 ...
##  $ salary_currency   : chr  "EUR" "USD" "USD" "USD" ...
##  $ salary_in_usd     : int  85847 30000 25500 175000 120000 222200 136000 219000 141000 147100 ...
##  $ employee_residence: chr  "ES" "US" "US" "CA" ...
##  $ remote_ratio      : int  100 100 100 100 100 0 0 0 0 0 ...
##  $ company_location  : chr  "ES" "US" "US" "CA" ...
##  $ company_size      : chr  "L" "S" "S" "M" ...
```

Already we can see an issue that needs to be worked on. Several variables seem to supposedly be read in as factors. We will finish conducting checks on the dataset before converting said columns.

Checking the summary of the dataset

```r
summary(salaries)
```

```
##    work_year    experience_level   employment_type     job_title
##      salary          salary_currency    salary_in_usd    employee_residence
##   remote_ratio    company_location   company_size
##  [ reached getOption("max.print") -- omitted 6 rows ]
```

Checking for null values

```r
colSums(is.na(salaries))
```

```
##        work_year experience_level  employment_type        job_title
##                0                0                0                0
##           salary
##                0
##  [ reached getOption("max.print") -- omitted 6 entries ]
```

Fortunately, we have no null values so imputing is not required

Checking potential factor columns for their unique values

```r
factor_cols <- salaries[, c(2, 3, 4, 6, 8, 10, 11)]

# finding unique values, referenced code from https://www.kaggle.com/code/abdulfaheem11/data-science-sa

# output ommitted to prevent too much space being taken up
sapply(factor_cols, function(col) unique(col))
```

Changing said variables to become factors

```r
salaries[, c(2, 3, 4, 6, 8, 10, 11)] <- lapply(factor_cols, factor)
str(salaries)
```

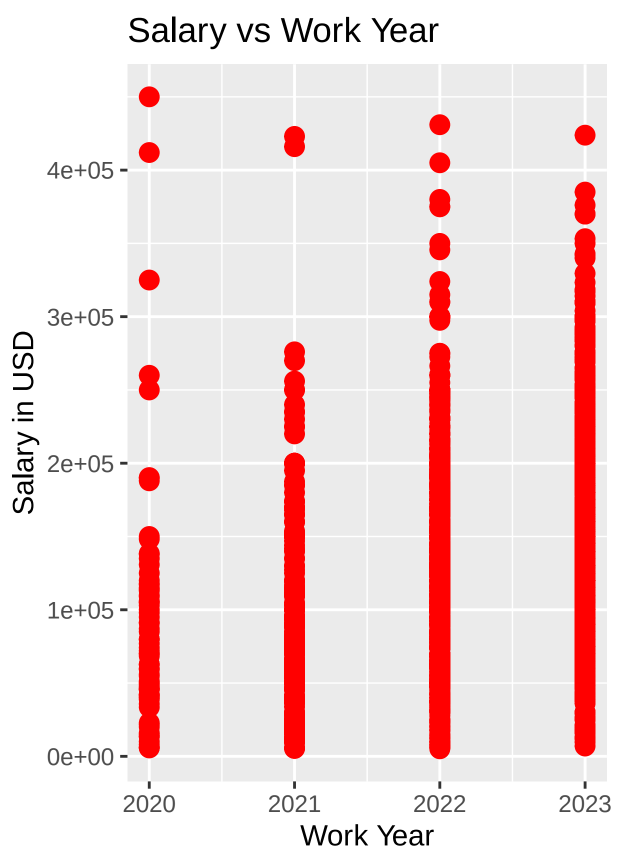
We can also drop the salary and salary_currency as we will just be using the salary_in_usd to simplify our steps. We will also drop the employee_residence, to put more focus onto the company_location instead.

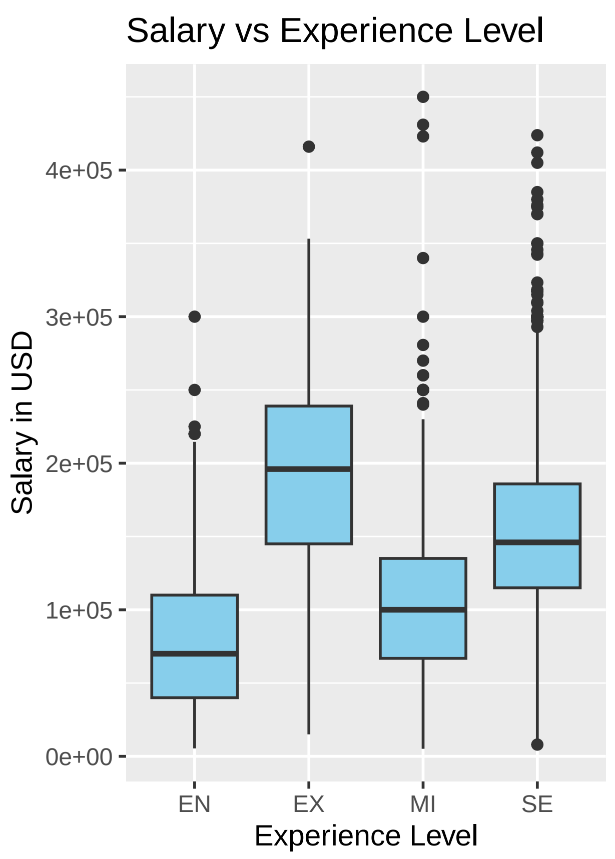
```r
salaries <- salaries[, !(names(salaries) %in% c('salary_currency','salary', 'employee_residence'))]
```

Visualization to search for patterns with regards to the salary_in_usd

Prioritizing focus on work_year, experience_level, employment_type, job_title, employee_residence, remote_ratio, company_location, company_size

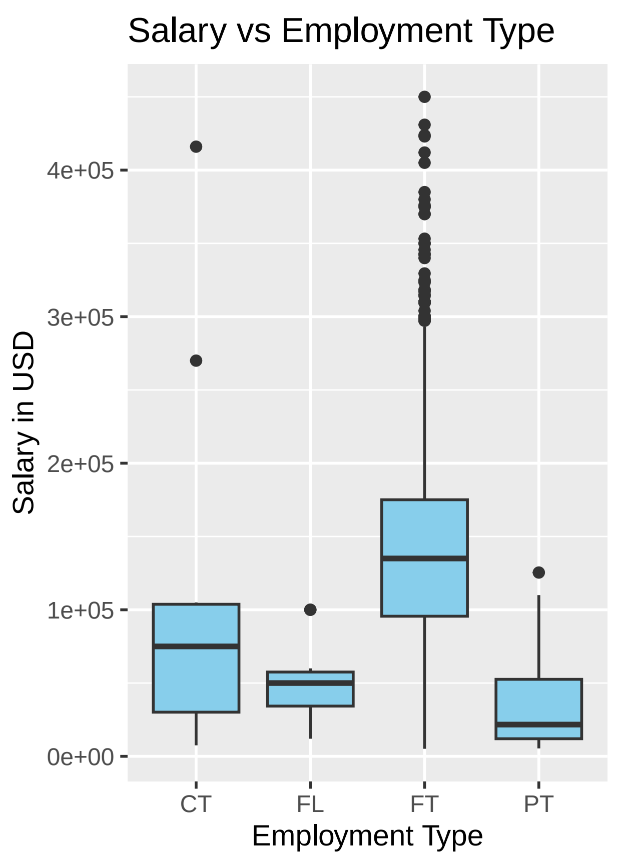
```r
yearplot <- ggplot(salaries, aes(x = work_year, y = salary_in_usd)) +
  geom_point(color = "red", size = 3) +
  labs(x = "Work Year", y = "Salary in USD", title = "Salary vs Work Year")
expplot <- ggplot(salaries, aes(x = experience_level, y = salary_in_usd)) +
  geom_boxplot(fill = "skyblue") +
  labs(x = "Experience Level", y = "Salary in USD", title = "Salary vs Experience Level")
grid.arrange(yearplot, expplot, ncol = 2)
```

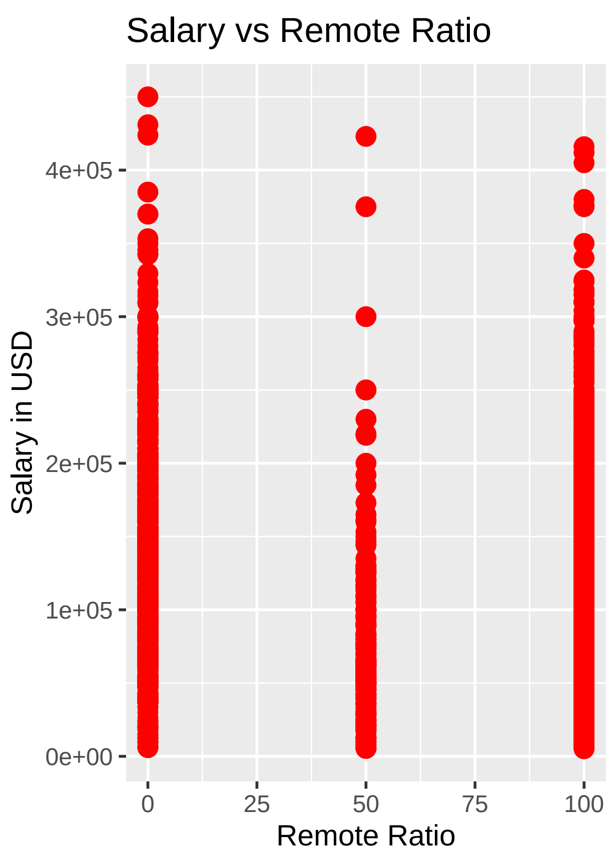




- We can see that the average salary in usd increases as the years go by, as the line congests further upwards towards the end.
- Experience level does not really show much of a trend as it goes towards seniority, We can tell though that EX has the highest average and MI has the highest peak

```
employplot <- ggplot(salaries, aes(x = employment_type, y = salary_in_usd)) +
  geom_boxplot(fill = "skyblue") +
  labs(x = "Employment Type", y = "Salary in USD", title = "Salary vs Employment Type")
remoteplot <- ggplot(salaries, aes(x = remote_ratio, y = salary_in_usd)) +
  geom_point(color = "red", size = 3, shape = 19) +
  labs(x = "Remote Ratio", y = "Salary in USD", title = "Salary vs Remote Ratio")
grid.arrange(employplot, remoteplot, ncol = 2)
```

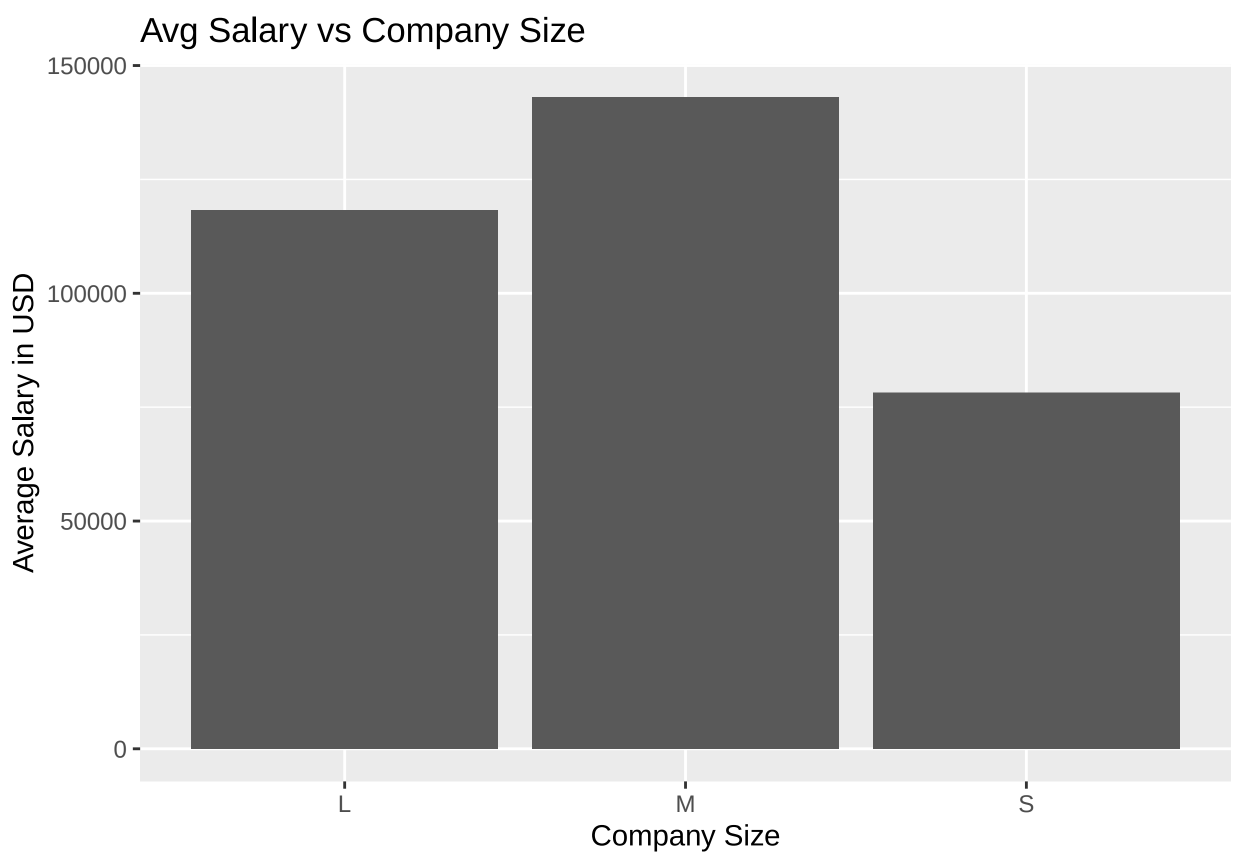




- In employment type, FT has the highest average as well as higher peaks
- Remote ratio consists of 0, 50 and 100. The highest points as well as average are in the order 0>100>50

```
usd_salary_by_size <- salaries%>%
  group_by(company_size)%>%
  summarise(Avg_sal=mean(salary_in_usd))

sizeplot <- ggplot(usd_salary_by_size, aes(x=company_size, y=Avg_sal)) +
  geom_col() +
  labs(title='Avg Salary vs Company Size', x='Company Size', y='Average Salary in USD')
sizeplot
```



+ From this plot we can also see that medium sized companies pay the largest on average, followed by large then small

Tabling the 6 highest and lowest paying jobs

```
options("max.print" = 6)
top_6_job_salaries<-salaries%>%
  group_by(job_title)%>%
  summarise(Avg_Sal=mean(salary_in_usd))%>%
  arrange(desc(Avg_Sal))%>%
  head()
top_6_job_salaries
```

```
## # A tibble: 6 x 2
##   job_title                Avg_Sal
##   <fct>                      <dbl>
## 1 Data Science Tech Lead   375000
## 2 Cloud Data Architect     250000
## 3 Data Lead                212500
## 4 Data Analytics Lead      211254.
## 5 Principal Data Scientist 198171.
## 6 Director of Data Science 195141.
```

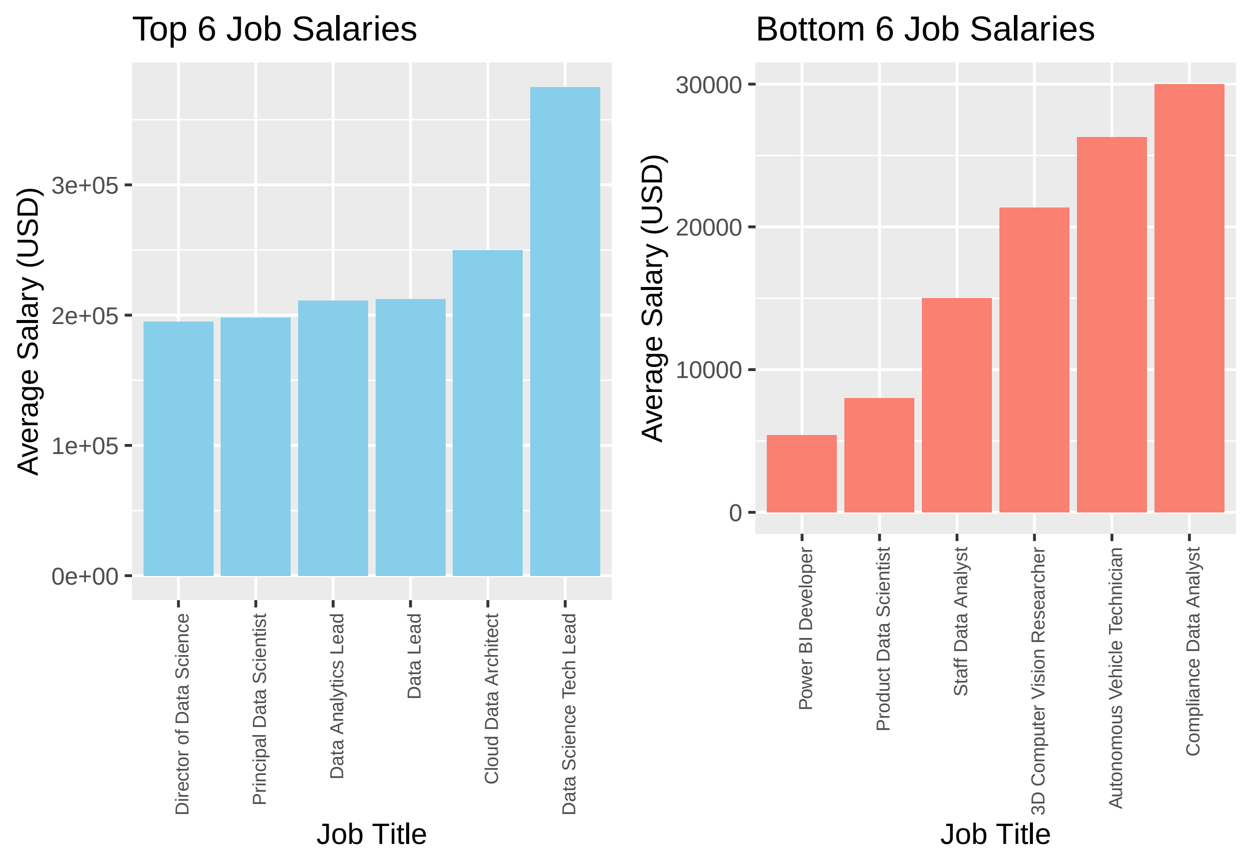
```
bottom_6_job_salaries<-salaries%>%
  group_by(job_title)%>%
  summarise(Avg_Sal=mean(salary_in_usd))%>%
```

```r
  arrange(Avg_Sal)%>%
  head()
bottom_6_job_salaries
```

```
## # A tibble: 6 x 2
##   job_title                     Avg_Sal
##   <fct>                           <dbl>
## 1 Power BI Developer               5409
## 2 Product Data Scientist           8000
## 3 Staff Data Analyst              15000
## 4 3D Computer Vision Researcher  21352.
## 5 Autonomous Vehicle Technician  26278.
## 6 Compliance Data Analyst        30000
```

Plotting the 6 highest and lowest paying jobs

```r
top6jobplot <- ggplot(top_6_job_salaries, aes(x = reorder(job_title, Avg_Sal), y = Avg_Sal)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  labs(title = "Top 6 Job Salaries", x = "Job Title", y = "Average Salary (USD)")
bot6jobplot <- ggplot(bottom_6_job_salaries, aes(x = reorder(job_title, Avg_Sal), y = Avg_Sal)) +
  geom_bar(stat = "identity", fill = "salmon") +
  labs(title = "Bottom 6 Job Salaries", x = "Job Title", y = "Average Salary (USD)")
top6jobplot <- top6jobplot + theme(axis.text.x = element_text(angle = 90, vjust = 0.75, size=7, hjust=1
bot6jobplot <- bot6jobplot + theme(axis.text.x = element_text(angle = 90, vjust = 0.75, size=7, hjust=1
grid.arrange(top6jobplot,bot6jobplot, ncol = 2)
```



+ We can tell that the data science tech lead job has the highest average pay by far + we can also tell that the power bi developer has the lowest pay out of all the jobs

Tabling the 6 highest and lowest paying company locations

```
top_6_loc<-salaries%>%
  group_by(company_location)%>%
  summarise(Avg_Sal=mean(salary_in_usd))%>%
  arrange(desc(Avg_Sal))%>%
  head()
top_6_loc
```

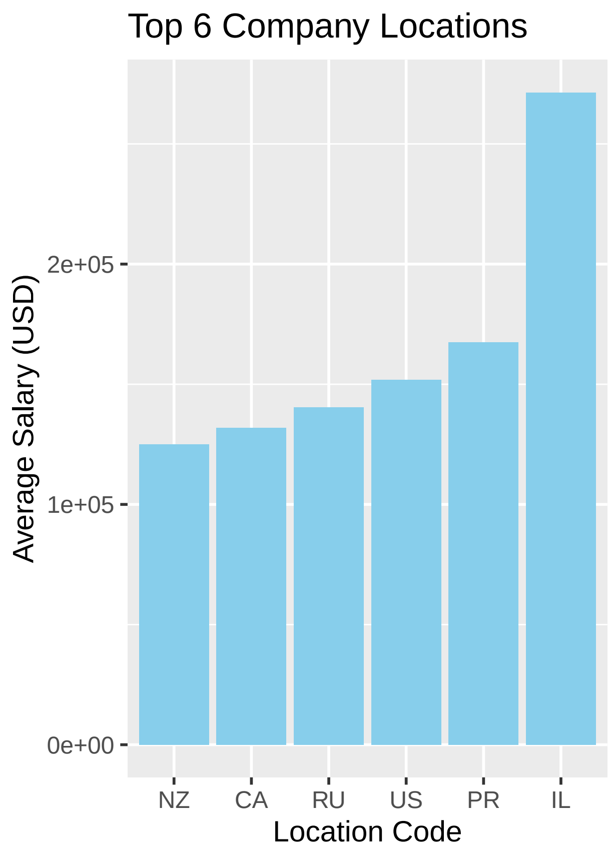
```
## # A tibble: 6 x 2
##   company_location Avg_Sal
##   <fct>              <dbl>
## 1 IL               271446.
## 2 PR               167500
## 3 US               151822.
## 4 RU               140333.
## 5 CA               131918.
## 6 NZ               125000
```

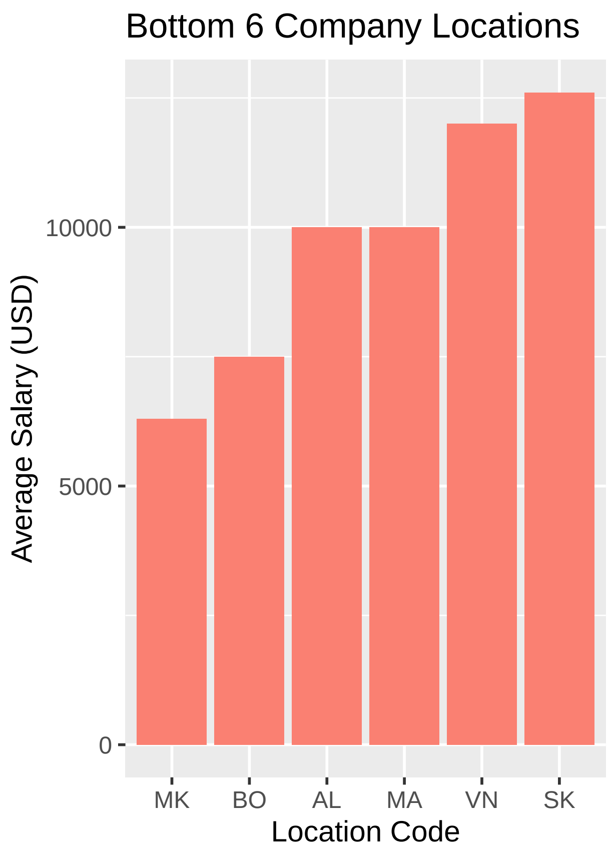
```
bottom_6_loc<-salaries%>%
  group_by(company_location)%>%
  summarise(Avg_Sal=mean(salary_in_usd))%>%
  arrange(Avg_Sal)%>%
  head()
bottom_6_loc
```

```
## # A tibble: 6 x 2
##   company_location Avg_Sal
##   <fct>              <dbl>
## 1 MK                  6304
## 2 BO                  7500
## 3 AL                 10000
## 4 MA                 10000
## 5 VN                 12000
## 6 SK                 12608
```

Plotting the 6 highest and lowest paying company locations

```
top6locplot <- ggplot(top_6_loc, aes(x = reorder(company_location, Avg_Sal), y = Avg_Sal)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  labs(title = "Top 6 Company Locations", x = "Location Code", y = "Average Salary (USD)")
bot6locplot <- ggplot(bottom_6_loc, aes(x = reorder(company_location, Avg_Sal), y = Avg_Sal)) +
  geom_bar(stat = "identity", fill = "salmon") +
  labs(title = "Bottom 6 Company Locations", x = "Location Code", y = "Average Salary (USD)")
top6resplot <- top6locplot + theme(axis.text.x = element_text(angle = 90, vjust = 0.75, size=7, hjust=1
bot6resplot <- bot6locplot + theme(axis.text.x = element_text(angle = 90, vjust = 0.75, size=7, hjust=1
grid.arrange(top6locplot,bot6locplot, ncol = 2)
```





+ It seems that IL has the highest paying jobs when it comes to company location + MK has the lowest paying job out of all the locations by far

Part 2: Problem formulation and preparation for statistical learning algorithms