Data Developer Salary Analysis

By Ian Joshua Sainani

25 Jul 2024

Data Source: https://www.kaggle.com/datasets/zeesolver/data-eng-salary-2024

Load Packages

```r
library(dplyr)
library(ggplot2)
library(statsr)
library("scales")
library(countrycode)
```

Import Data

```r
salaries_df <- read.csv("~/Library/CloudStorage/OneDrive-NanyangTechnologicalUniversity/personal projec
```

Data Exploration

Overview of the structure of salaries_df

```r
str(salaries_df)
```

```
## 'data.frame':    16534 obs. of  11 variables:
##  $ work_year         : int  2024 2024 2024 2024 2024 2024 2024 2024 2024 2024 ...
##  $ experience_level  : chr  "SE" "SE" "SE" "SE" ...
##  $ employment_type   : chr  "FT" "FT" "FT" "FT" ...
##  $ job_title         : chr  "AI Engineer" "AI Engineer" "Data Engineer" "Data Engineer" ...
##  $ salary            : int  202730 92118 130500 96000 190000 160000 400000 65000 101520 45864 ...
##  $ salary_currency   : chr  "USD" "USD" "USD" "USD" ...
##  $ salary_in_usd     : int  202730 92118 130500 96000 190000 160000 400000 65000 101520 45864 ...
##  $ employee_residence: chr  "US" "US" "US" "US" ...
##  $ remote_ratio      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ company_location  : chr  "US" "US" "US" "US" ...
##  $ company_size      : chr  "M" "M" "M" "M" ...
```

Check for missing data

```r
cat("Count of null values:", sum(is.na(salaries_df)))
```

```
## Count of null values: 0
```

View unique values in columns of interest

```r
distinct_values <- list(
  Years = unique(salaries_df$work_year),
  Experience_Levels = unique(salaries_df$experience_level),
  Employment_Types = unique(salaries_df$employment_type),
  Company_Sizes = unique(salaries_df$company_size),
  Company_Location = unique(salaries_df$company_location)
)

distinct_values
```

```
## $Years
## [1] 2024 2022 2023 2020 2021
##
## $Experience_Levels
## [1] "SE" "MI" "EN" "EX"
##
## $Employment_Types
## [1] "FT" "CT" "PT" "FL"
##
## $Company_Sizes
## [1] "M" "L" "S"
##
## $Company_Location
##  [1] "US" "AU" "GB" "CA" "NL" "LT" "DK" "FR" "ZA" "NZ" "AR" "ES" "KE" "LV" "IN"
## [16] "DE" "IL" "FI" "AT" "BR" "CH" "AE" "PL" "SA" "UA" "EG" "PH" "TR" "OM" "MX"
## [31] "PT" "BA" "IT" "AS" "IE" "EE" "MT" "HU" "LB" "RO" "VN" "NG" "LU" "GI" "CO"
## [46] "SI" "GR" "MU" "RU" "KR" "CZ" "QA" "GH" "SE" "AD" "EC" "NO" "JP" "HK" "CF"
## [61] "SG" "TH" "HR" "AM" "PK" "IR" "BS" "PR" "BE" "ID" "MY" "HN" "DZ" "IQ" "CN"
## [76] "CL" "MD"
```

Problem Formulation

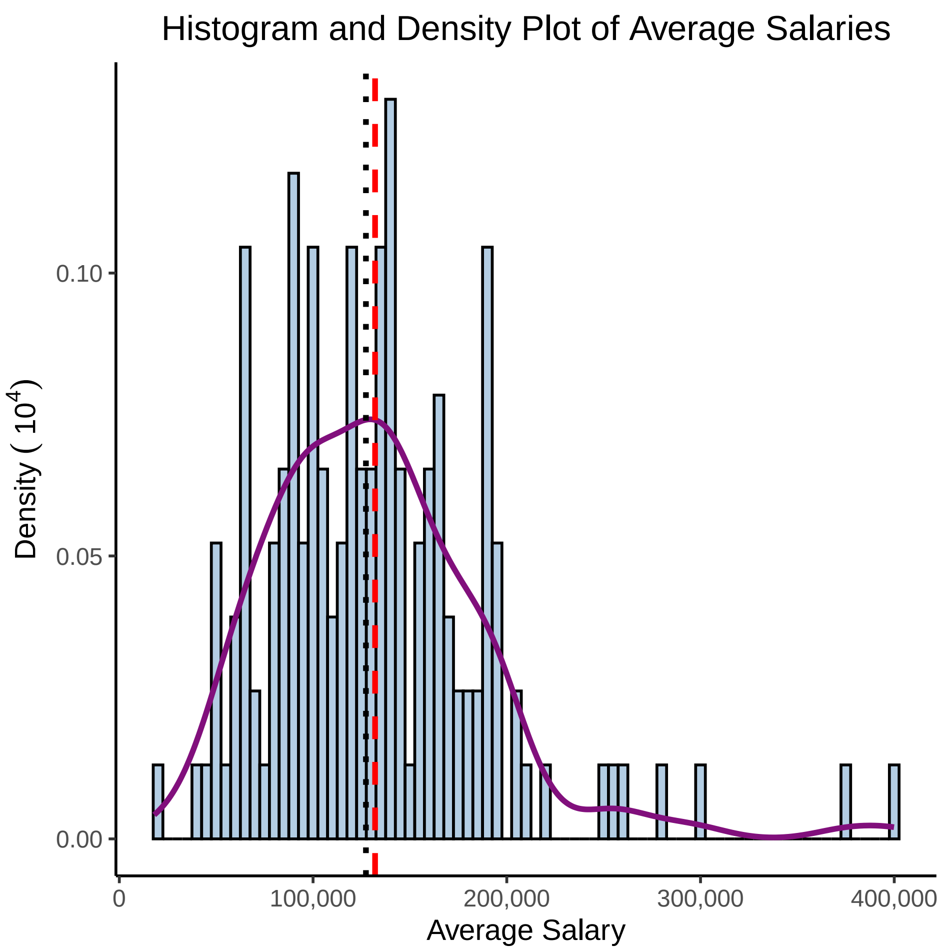
Average Salary of each Job Title

```r
average_salaries <- salaries_df %>%
  filter(employment_type == "FT") %>%
  group_by(job_title) %>%
  summarise(AvgSalary = mean(salary_in_usd), MedSalary = median(salary_in_usd),
            MinSalary = min(salary_in_usd), MaxSalary = max(salary_in_usd),
```

```r
            Std = sd(salary_in_usd)) %>%
  arrange(desc(AvgSalary))

average_salaries
```

```
## # A tibble: 153 x 6
##    job_title                        AvgSalary MedSalary MinSalary MaxSalary     Std
##    <chr>                                <dbl>     <dbl>     <int>     <int>   <dbl>
##  1 Analytics Engineering Manager      399880    399880    399880    399880      NA
##  2 Data Science Tech Lead             375000    375000    375000    375000      NA
##  3 Head of Machine Learning           299758.   330000     76309    448000 137103.
##  4 Managing Director Data Scien~      280000    280000    260000    300000  28284.
##  5 AWS Data Architect                 258000    258000    258000    258000      NA
##  6 AI Architect                       252551.   204000     99750    800000 131291.
##  7 Cloud Data Architect               250000    250000    250000    250000      NA
##  8 Director of Data Science           218775.   217000     57786    375500  72954.
##  9 Head of Data                       211860.   215000     31520    329500  66834.
## 10 Prompt Engineer                    205094.   197011     60462    600000 115091.
## # i 143 more rows
```

```r
# Create density plot
hist_avg_salaries <- ggplot(data = average_salaries, aes(x = AvgSalary)) +
                     geom_histogram(aes(y = after_stat(density*10^4)), binwidth = 5000,
                                    fill = "#b3cde3", color = "black") +
                     geom_density(aes(y = after_stat(density*10^4)),
                                  color = "#810f7c", size = 1) +
                     geom_vline(aes(xintercept = mean(AvgSalary)),
                                color = "red", size = 1,
                                linetype = "dashed") +
                     geom_vline(aes(xintercept = median(AvgSalary)),
                                color = "black", size = 1,
                                linetype = "dotted")

# Styling
hist_avg_salaries + labs(title = "Histogram and Density Plot of Average Salaries",
                         x = "Average Salary",
                         y = expression(Density~(~10^4))) +
                    scale_x_continuous(labels = comma) +
                    theme(plot.title = element_text(hjust = 0.5),
                    panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
                    panel.background = element_blank(), axis.line = element_line(colour = "black"))
```




```
# Print mean and median values
cat("Mean:", mean(average_salaries$AvgSalary), "\n")
```

```
## Mean: 132017.8
```

```
cat("Median:", median(average_salaries$AvgSalary))
```

```
## Median: 127292.8
```

```
# Standard deviation of salaries among all individual full time jobs
std_all <- salaries_df %>%
  filter(employment_type == "FT") %>%
  summarise(std = sd(salary_in_usd))

# Standard deviation of average salaries, among full time, unique job titles
std_unique <- average_salaries %>%
  summarise(std = sd(average_salaries$AvgSalary))

cat("Standard deviation (FT, All):", std_all$std, '\n')
```

```
## Standard deviation (FT, All): 68351.02
```

```r
cat("Standard deviation (FT, Unique):", std_unique$std)
```

```
## Standard deviation (FT, Unique): 57862.94
```

The standard deviations can be observed to be very large, and it can be seen to drop as we establish a grouping.

We want to then find out if there are any factors that impact the salaries of employees, and in what way.

How does experience level affect salary?

For a fair comparison, we will filter only FT employees

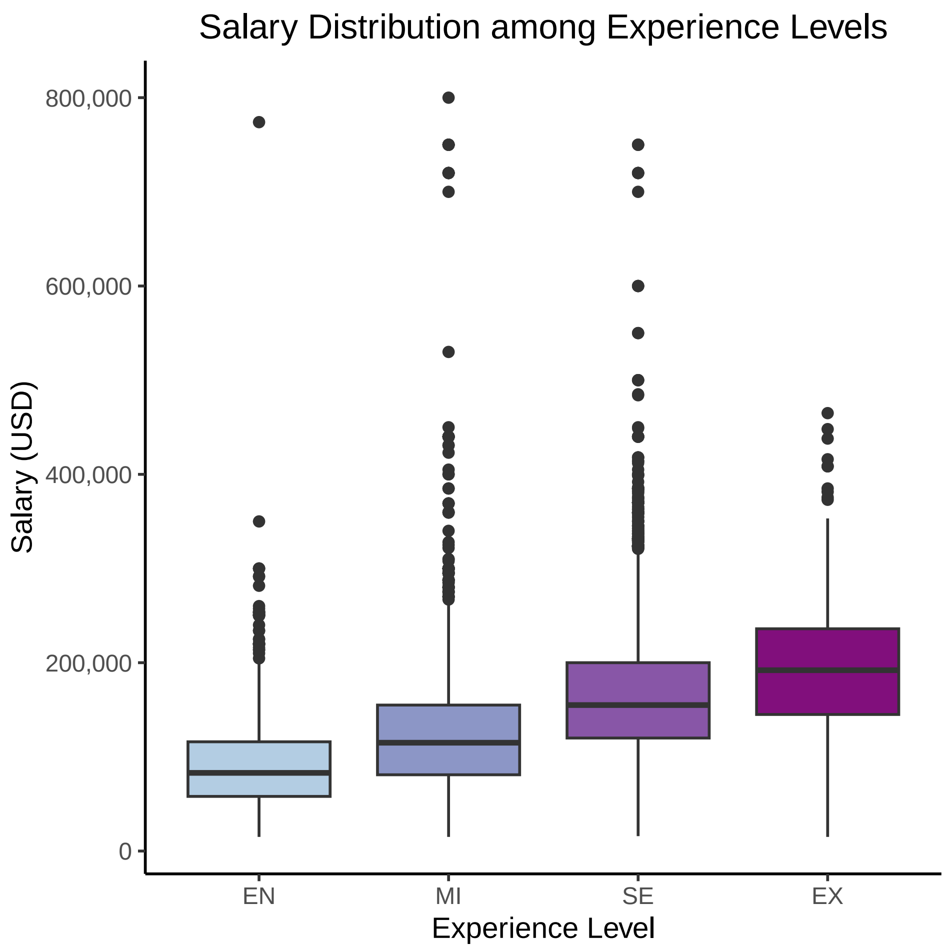
```r
# Descriptive Statistics
exp_salaries <- salaries_df %>%
  filter(employment_type == "FT") %>%
  group_by(experience_level) %>%
  summarise(min = min(salary_in_usd), q1 = quantile(salary_in_usd,0.25),
    AvgExpSalary = mean(salary_in_usd), median = median(salary_in_usd),
    q3 = quantile(salary_in_usd, 0.75), max = max(salary_in_usd),
    std = sd(salary_in_usd)) %>%
  arrange(desc(AvgExpSalary))

# Create a colour gradient
colour_gradient_exp <- c("EN" = "#b3cde3", "MI" = "#8c96c6", "SE" = "#8856a7", "EX" = "#810f7c")

# Sort experience level
exp_level <- c('EN', 'MI', 'SE', 'EX')

# Create a barplot
salary_vs_exp <- ggplot(data = exp_salaries, aes(x = factor(experience_level,
                level = exp_level), y = AvgExpSalary, fill = experience_level)) +
                scale_y_continuous(labels = comma) +
                scale_fill_manual(values = colour_gradient_exp)

# Styling
salary_vs_exp + labs(title = "Average Salary vs Experience Level", x = "Experience Level",
                     y = "Average Salary (USD)") +
                     theme(plot.title = element_text(hjust = 0.5),
                     panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
                     panel.background = element_blank(), axis.line = element_line(colour = "black"))
```


```r
# View Descriptive Statistics
exp_salaries
```

```
## # A tibble: 4 x 8
##   experience_level   min     q1 AvgExpSalary  median     q3    max    std
##   <chr>            <int>  <dbl>        <dbl>   <dbl>  <dbl>  <int>  <dbl>
## 1 EX               15000 145000      194823. 191928. 235250  465000 69772.
## 2 SE               15809 120250      163732. 155000  200000  750000 63898.
## 3 MI               15000  81500      126224. 115360  155000  800000 67040.
## 4 EN               15000  58780.      92827.  83171  117006. 774000 51583.
```

```r
# Create a boxplot
bp_exp_salaries <- ggplot(data = salaries_df, aes(x = factor(experience_level,
                  level = exp_level), y = salary_in_usd,
                  fill = experience_level)) + geom_boxplot(show.legend = FALSE) +
                  scale_y_continuous(labels = comma)+
                  scale_fill_manual(values = colour_gradient_exp)

# Styling
bp_exp_salaries + labs(title = "Salary Distribution among Experience Levels",
                      x = "Experience Level", y = "Salary (USD)") +
                    theme(plot.title = element_text(hjust = 0.5),
```

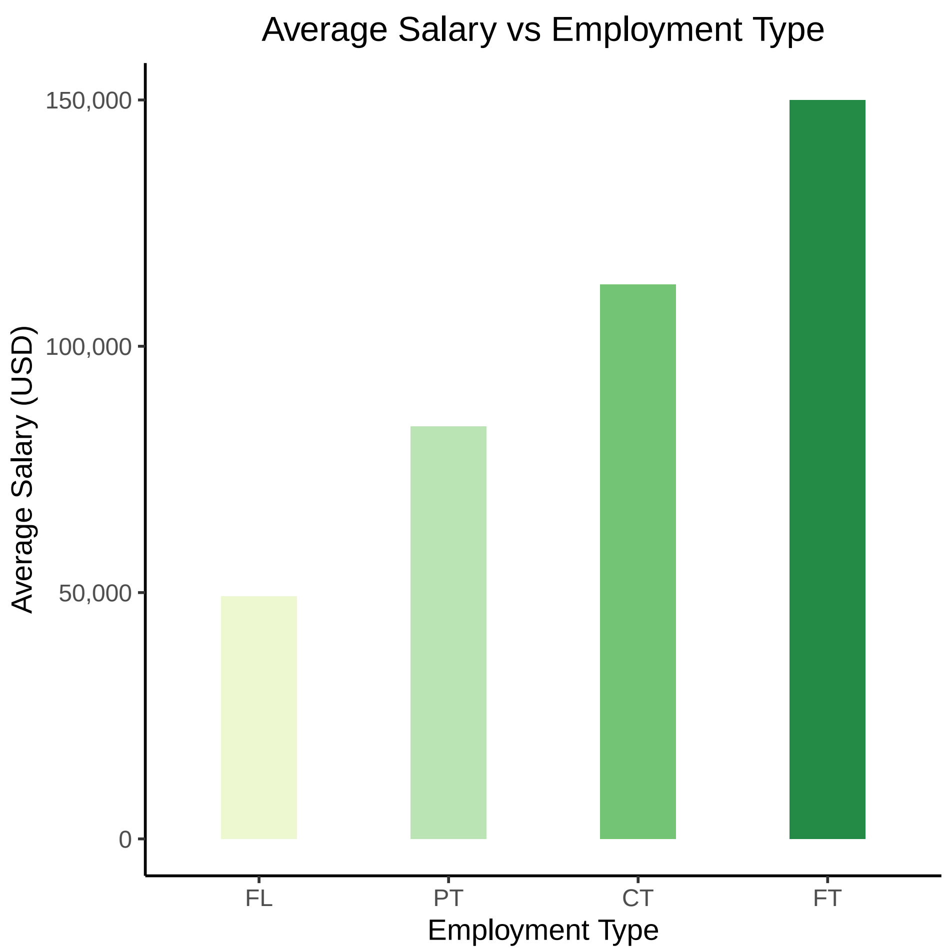
```
                       panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
                       panel.background = element_blank(), axis.line = element_line(colour = "black"))
```



How does employment type affect salary?

```
# Descriptive Statistics
emp_salaries <- salaries_df %>%
  group_by(employment_type) %>%
  summarise(min = min(salary_in_usd), q1 = quantile(salary_in_usd, 0.25),
    AvgEmpSalary = mean(salary_in_usd), median = median(salary_in_usd),
    q3 = quantile(salary_in_usd, 0.75), max = max(salary_in_usd),
    std = sd(salary_in_usd)) %>%
  arrange(desc(AvgEmpSalary))

# Create a colour gradient
colour_gradient_emp <- c("FL" = "#edf8d1", "PT" = "#bae4b3", "CT" = "#74c476", "FT" = "#238b45")

# Sort employment type
emp_type <- c('FL', 'PT', 'CT', 'FT')
```

```r
# Create a barplot
salary_vs_emp <- ggplot(data = emp_salaries, aes(x = factor(employment_type,
                level = emp_type), y = AvgEmpSalary, fill = employment_type)) +
                geom_col(width = 0.4, show.legend = FALSE) +
                scale_y_continuous(labels = comma) +
                scale_fill_manual(values = colour_gradient_emp)

# Styling
salary_vs_emp + labs(title = "Average Salary vs Employment Type", x = "Employment Type",
                     y = "Average Salary (USD)") +
                      theme(plot.title = element_text(hjust = 0.5),
                      panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
                      panel.background = element_blank(), axis.line = element_line(colour = "black"))
```

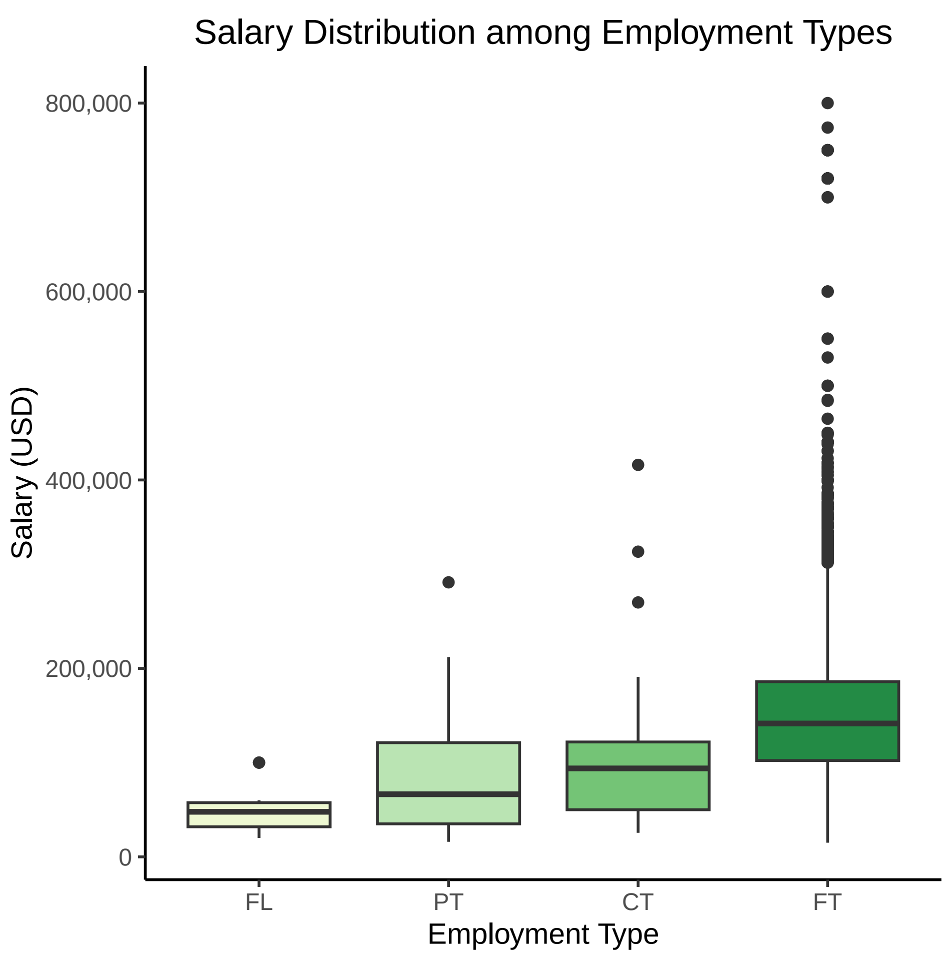



```r
# View Descriptive Statistics
emp_salaries
```

```
## # A tibble: 4 x 8
##   employment_type   min     q1 AvgEmpSalary  median     q3    max    std
##   <chr>           <int>  <dbl>        <dbl>   <dbl>  <dbl>  <int>  <dbl>
## 1 FT              15000 102225      149988. 141525  185900 800000 68351.
```

```
## 2 CT               25500  50000        112578.  93856  121902. 416000 91676.
## 3 PT               15966  35028.        83750.  66452. 121158. 291340 61774.
## 4 FL               20000  31892.        49221.  47778.  57500  100000 24997.
```

```r
# Create a boxplot
bp_emp_salaries <- ggplot(data = salaries_df, aes(x = factor(employment_type,
                  level = emp_type), y = salary_in_usd,
                  fill = employment_type)) + geom_boxplot(show.legend = FALSE) +
                  scale_y_continuous(labels = comma)+
                  scale_fill_manual(values = colour_gradient_emp)

# Styling
bp_emp_salaries + labs(title = "Salary Distribution among Employment Types",
                       x = "Employment Type", y = "Salary (USD)") +
                      theme(plot.title = element_text(hjust = 0.5),
                      panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
                      panel.background = element_blank(), axis.line = element_line(colour = "black"))
```



Does remote ratio have an impact on salary?

```
cor.test(salaries_df$remote_ratio, salaries_df$salary_in_usd)
```

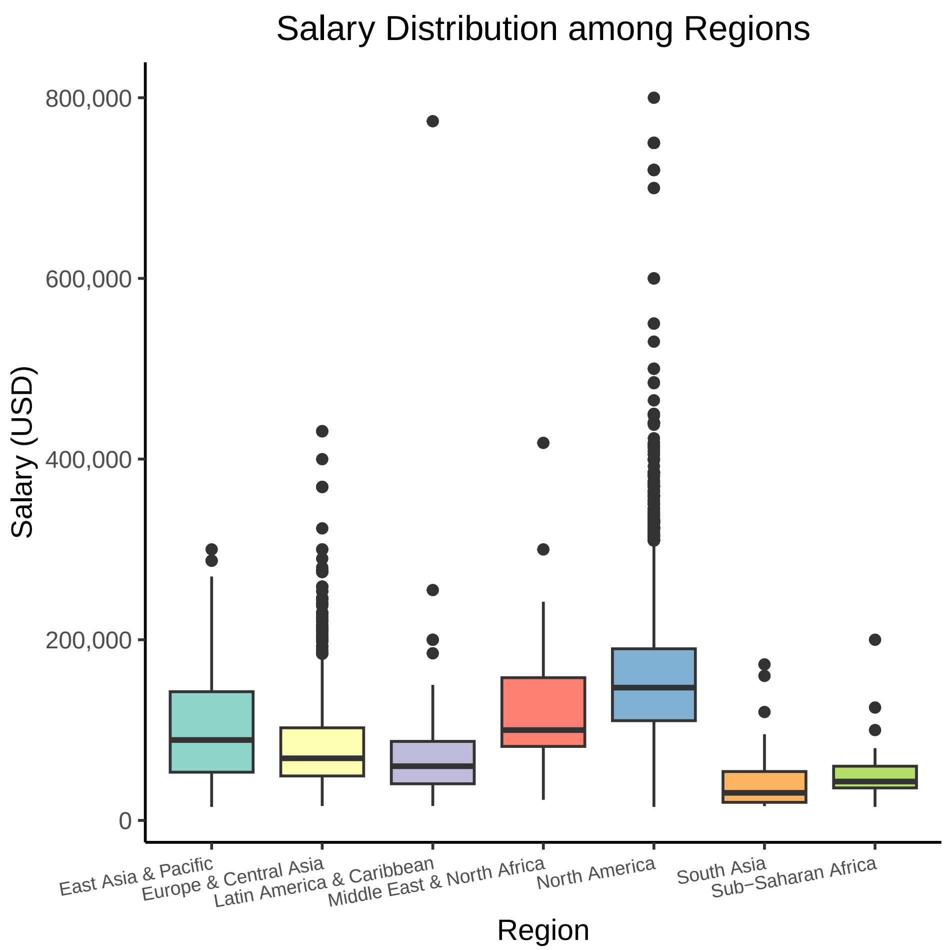
```
##
##  Pearson's product-moment correlation
##
## data:  salaries_df$remote_ratio and salaries_df$salary_in_usd
## t = -7.3781, df = 16532, p-value = 1.681e-13
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.07246839 -0.04208281
## sample estimates:
##         cor
## -0.05728887
```

With a correlation score of -0.0573, there is a very weak negative correlation between the remote ratio and salary.

How does company region affect salary?

```
# Create new column indicating region of company location

salaries_df$company_region = countrycode(salaries_df$company_location, "iso2c","region")
```

```
# Descriptive Statistics
reg_salaries <- salaries_df %>%
  filter(employment_type == "FT") %>%
  group_by(company_region) %>%
  summarise(min = min(salary_in_usd), q1 = quantile(salary_in_usd, 0.25),
    AvgRegSalary = mean(salary_in_usd), median = median(salary_in_usd),
    q3 = quantile(salary_in_usd, 0.75), max = max(salary_in_usd),
    std = sd(salary_in_usd)) %>%
  arrange(desc(AvgRegSalary))

# Create a boxplot
bp_reg_salaries <- ggplot(data = salaries_df, aes(x = company_region, y = salary_in_usd,
                   fill = company_region)) +
                   geom_boxplot(show.legend = FALSE) +
                   scale_y_continuous(labels = comma) +
                   theme(axis.text.x = element_text(angle = 10, hjust = 1, size = 7)) +
                   scale_fill_brewer(palette = "Set3")

# Styling
bp_reg_salaries + labs(title = "Salary Distribution among Regions",
                        x = "Region", y = "Salary (USD)") +
                      theme(plot.title = element_text(hjust = 0.5),
                      panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
                      panel.background = element_blank(), axis.line = element_line(colour = "black"))
```



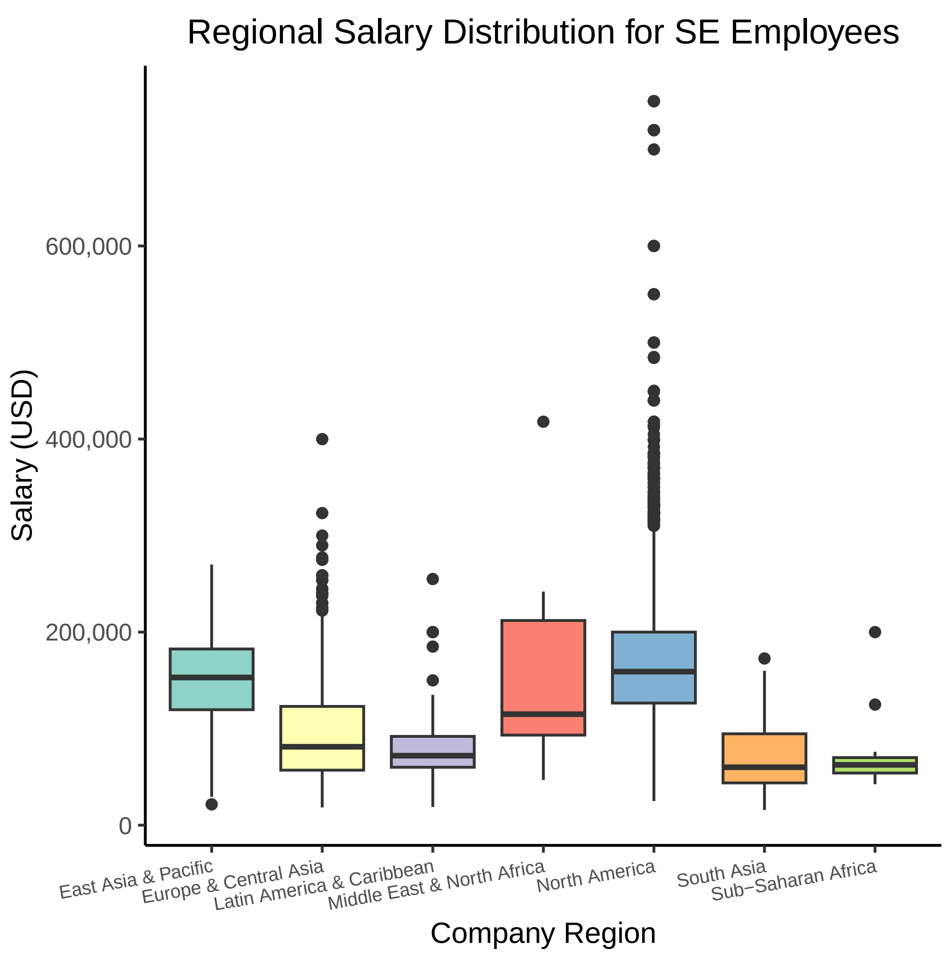

```
# View Descriptive Statistics
reg_salaries
```

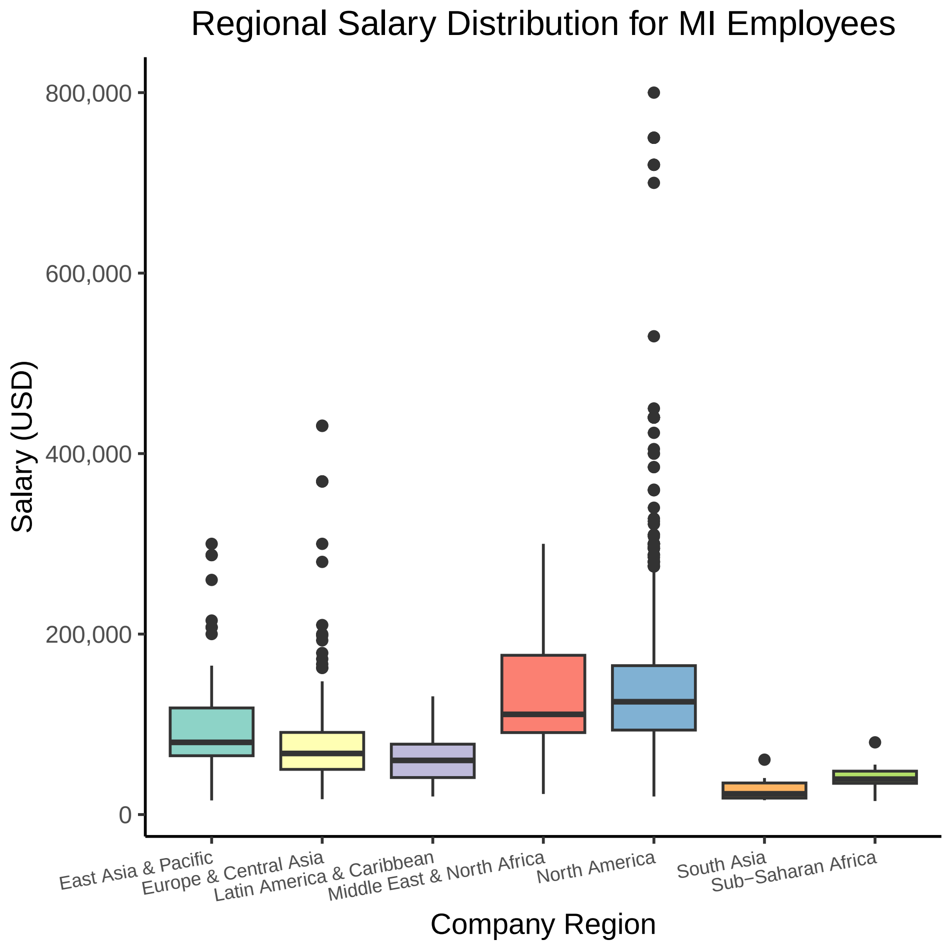
```
## # A tibble: 7 x 8
##   company_region               min      q1 AvgRegSalary median     q3    max    std
##   <chr>                      <int>   <dbl>        <dbl>  <dbl>  <dbl>  <int>  <dbl>
## 1 North America              15000 1.11e5      156748. 1.47e5 1.9 e5 800000 65673.
## 2 Middle East & North Afr~ 22800 8.34e4      129438. 1.03e5 1.69e5 417937 84246.
## 3 East Asia & Pacific        15000 5.34e4      106686. 8.90e4 1.42e5 300000 69905.
## 4 Europe & Central Asia      16455 4.92e4       84182. 6.88e4 1.03e5 430967 53244.
## 5 Latin America & Caribbe~ 16000 4.00e4       82869. 6    e4 8.80e4 774000 96778.
## 6 Sub-Saharan Africa         15000 3.67e4       53933. 4.28e4 5.70e4 200000 34165.
## 7 South Asia                 15809 2.02e4       43017. 3.17e4 5.48e4 172700 33218.
```

Drill down into regional salary distribution for each experience level

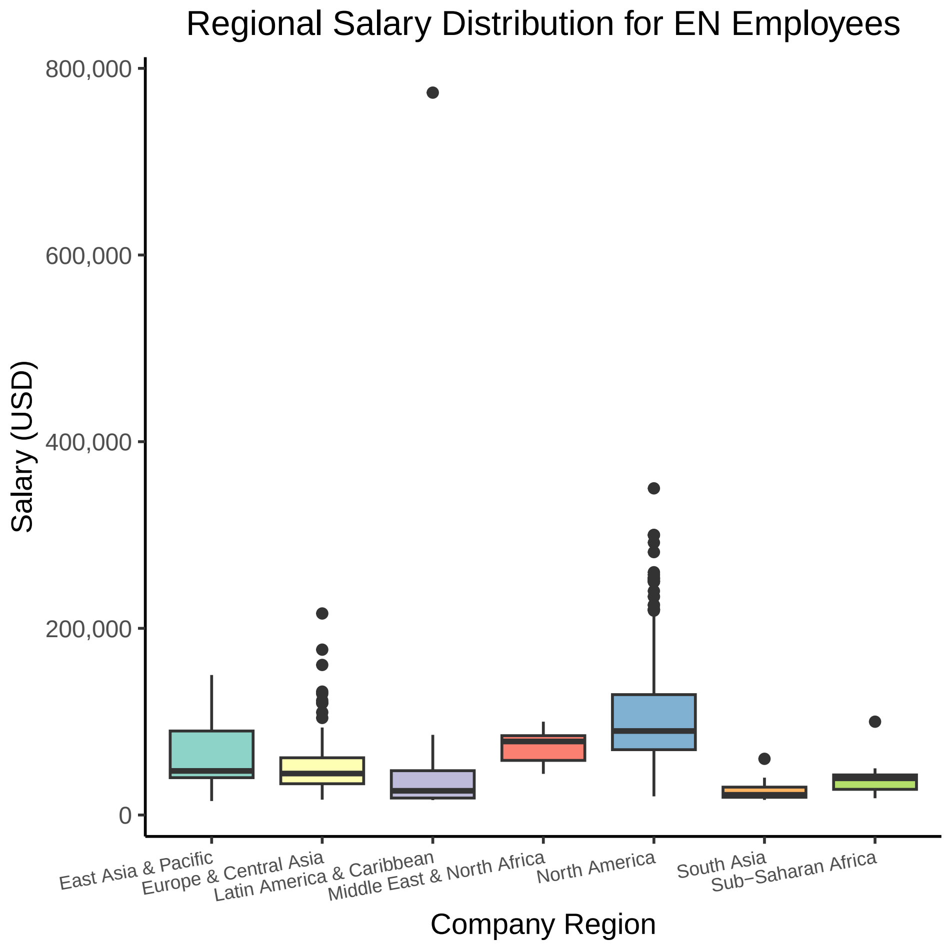
```
# Get the unique experience levels
experience_levels <- unique(salaries_df$experience_level)

# Loop through each experience level and create a boxplot
```

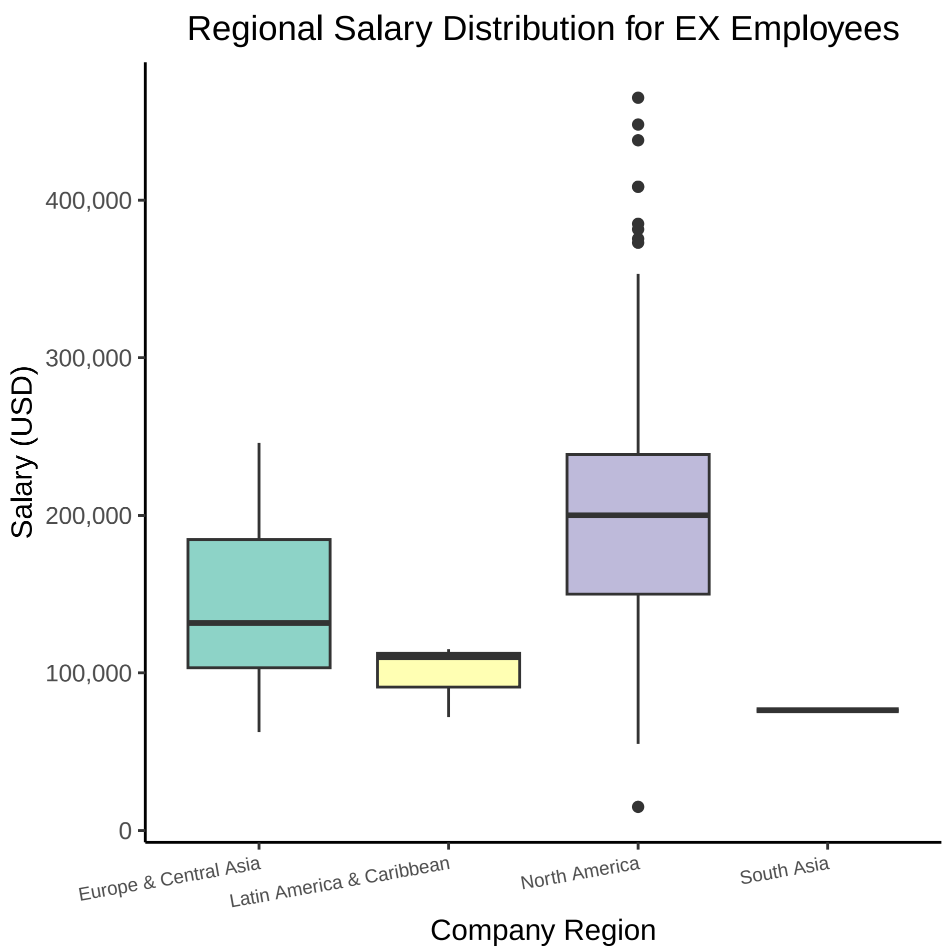
```
for (x in experience_levels) {
  # Subset data for the current experience level and employment type FT
  subset_df <- subset(salaries_df, experience_level == x & employment_type == "FT")

    # Create the boxplots
    plots <- ggplot(data = subset_df, aes(x = company_region, y = salary_in_usd, fill = company_region)
      geom_boxplot(show.legend = FALSE) +
      scale_y_continuous(labels = comma) +
      scale_fill_brewer(palette = "Set3") +
      labs(title = paste("Regional Salary Distribution for", x, "Employees"),
      x = "Company Region", y = "Salary (USD)") +
      theme(axis.text.x = element_text(angle = 10, hjust = 1, size = 7),
            panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
            panel.background = element_blank(), axis.line = element_line(colour = "black"),

    print(plots)
}
```





Regional Salary Distribution for MI Employees
Salary (USD)
800,000
600,000
400,000
200,000
0
East Asia & Pacific
Europe & Central Asia
Latin America & Caribbean
Middle East & North Africa
North America
South Asia
Sub−Saharan Africa
Company Region


Regional Salary Distribution for EN Employees
800,000
600,000
400,000
200,000
0
Salary (USD)
East Asia & Pacific
Europe & Central Asia
Latin America & Caribbean
Middle East & North Africa
North America
South Asia
Sub−Saharan Africa
Company Region



Does the size of the company impact the salary of their employees?

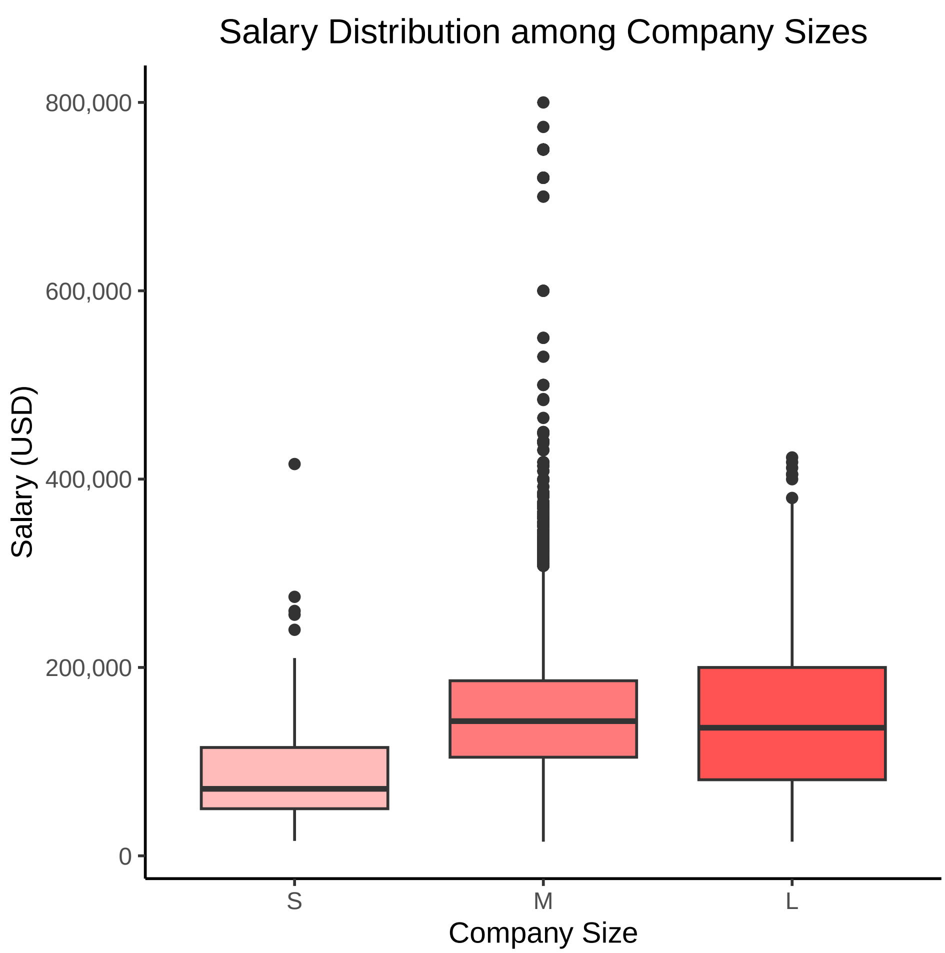
```
# Descriptive Statistics
size_salaries <- salaries_df %>%
  filter(employment_type == "FT") %>%
  group_by(company_size) %>%
  summarise(min = min(salary_in_usd), q1 = quantile(salary_in_usd, 0.25),
    AvgSizeSalary = mean(salary_in_usd), median = median(salary_in_usd),
    q3 = quantile(salary_in_usd, 0.75), max = max(salary_in_usd),
    std = sd(salary_in_usd))

# Create a colour gradient
colour_gradient_size <- c("S" = "#ffbaba", "M" = "#ff7b7b", "L" = "#ff5252")

# Sort company size
size_level <- c('S', 'M', 'L')

# Create a barplot
salary_vs_size <- ggplot(data = size_salaries, aes(x = factor(company_size, level = size_level),
                  y = AvgSizeSalary, fill = company_size)) +
```

```r
                   geom_col(width = 0.4, show.legend = FALSE) +
                   scale_y_continuous(labels = comma) +
                   scale_fill_manual(values = colour_gradient_size)

# Styling
salary_vs_size + labs(title = "Average Salary vs Company Size", x = "Company Size",
                      y = "Average Salary (USD)") +
                       theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
                       panel.background = element_blank(), axis.line = element_line(colour = "black"),
```


```r
# View Descriptive Statistics
size_salaries
```

```
## # A tibble: 3 x 8
##   company_size   min      q1 AvgSizeSalary median     q3    max    std
##   <chr>        <int>   <dbl>         <dbl>  <dbl>  <dbl>  <int>  <dbl>
## 1 L            15000  82304.       142023. 136000 202100 423000 73429.
## 2 M            15000 105000        151197. 143225 185900 800000 67800.
## 3 S            15809  50510.        87775.  76078 115000 275000 52891.
```

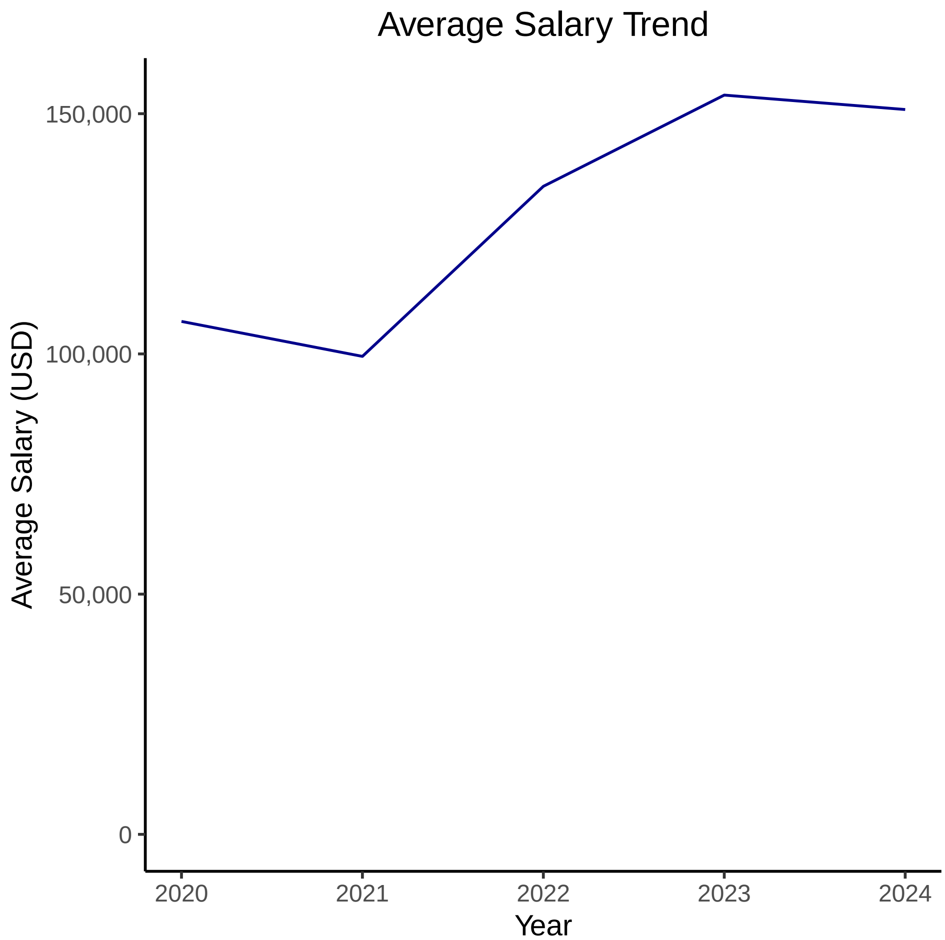
```r
# Create a boxplot
bp_size_salaries <- ggplot(data = salaries_df, aes(x = factor(company_size,
                  level = size_level), y = salary_in_usd,
                  fill = company_size)) + geom_boxplot(show.legend = FALSE) +
                  scale_y_continuous(labels = comma)+
                  scale_fill_manual(values = colour_gradient_size)

# Styling
bp_size_salaries + labs(title = "Salary Distribution among Company Sizes",
                       x = "Company Size", y = "Salary (USD)") +
                      theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
                      panel.background = element_blank(), axis.line = element_line(colour = "black"),
```



How has the average salary changed over the years?

```r
# Descriptive Statistics
years_salaries <- salaries_df %>%
  filter(employment_type == "FT") %>%
  group_by(work_year) %>%
```

```r
  summarise(min = min(salary_in_usd), q1 = quantile(salary_in_usd, 0.25),
    AvgYearSalary = mean(salary_in_usd), median = median(salary_in_usd),
    q3 = quantile(salary_in_usd, 0.75), max = max(salary_in_usd),
    std = sd(salary_in_usd))

# Create a line plot
line_years_salaries <- ggplot(data = years_salaries, aes(x = work_year, y = AvgYearSalary)) +
                       geom_line(color = "darkblue") +
                       scale_y_continuous(labels = comma, limits = c(0,NA))

# Styling
line_years_salaries + labs(title = "Average Salary Trend",
                       x = "Year", y = "Average Salary (USD)") +
                      theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
                      panel.background = element_blank(), axis.line = element_line(colour = "black"),
```




```r
# View Descriptive Statistics
years_salaries
```

```
## # A tibble: 5 x 8
##   work_year   min     q1 AvgYearSalary median     q3     max     std
```

```
##        <int> <int>  <dbl>         <dbl>  <dbl>  <dbl>  <int>  <dbl>
## 1       2020 15000  49268       106760.  87000 120000 450000 84380.
## 2       2021 15000  54202        99486.  86369 140000 423000 63013.
## 3       2022 15000  95000       134883. 132320 173000 430967 57612.
## 4       2023 15680 109400       153867. 145000 190000 750000 65275.
## 5       2024 17598 100000       150864. 140000 186153 800000 73655.
```