MCristinaFernandez.module05RProject

M. Cristina Fernandez

2024-11-23

The Situation

Your CEO has decided that the company needs a full-time data scientist, and possibly a team of them in the future. She thinks she needs someone who can help drive data science within then entire organization and could potentially lead a team in the future. She understands that data scientist salaries vary widely across the world and is unsure what to pay them. To complicate matters, salaries are going up due to the great recession and the market is highly competitive. Your CEO has asked you to prepare an analysis on data science salaries and provide them with a range to be competitive and get top talent. The position can work offshore, but the CEO would like to know what the difference is for a person working in the United States. Your company is currently a small company but is expanding rapidly.

```
require("tidyverse")
```

```
## Loading required package: tidyverse
```

```
## Warning: package 'dplyr' was built under R version 4.4.2
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✓ dplyr     1.1.4     ✓ readr     2.1.5
## ✓ forcats   1.0.0     ✓ stringr   1.5.1
## ✓ ggplot2   3.5.1     ✓ tibble    3.2.1
## ✓ lubridate 1.9.3     ✓ tidyr     1.3.1
## ✓ purrr     1.0.2
```

```
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

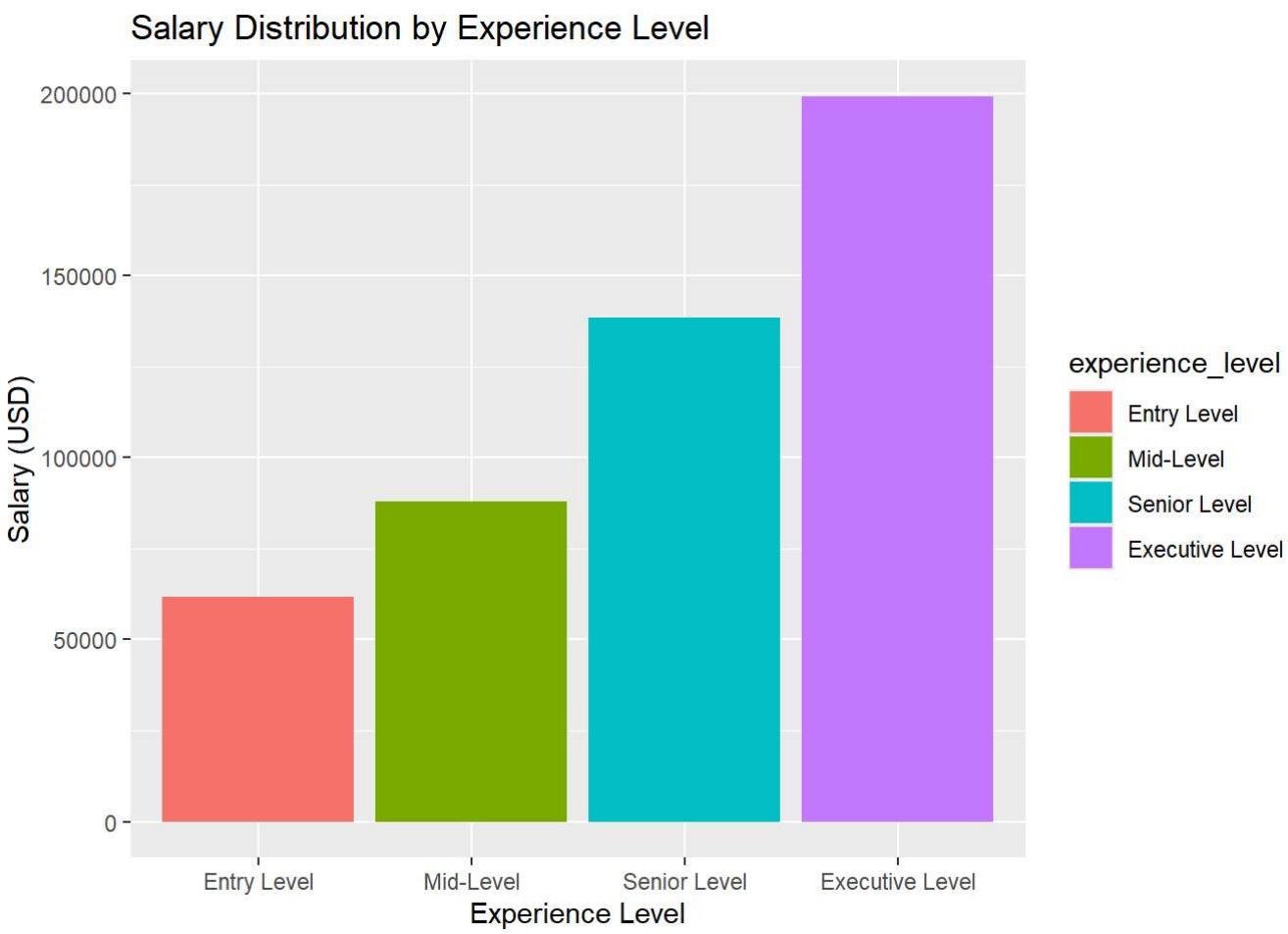
```
require("dplyr")
ds_salaries <- read.csv("C:/Users/crisf/Documents/DSE5002/project_1/r project data.csv")
view(ds_salaries)
```

I plan on looking up salaries by experience, employment type, company size, and remote ratio to see if/how these factors influence the salary ranges. I'm going to want to compare what the avg US-based salaries look like vs non-US based (the employee residence) I should see if I can determine any trends over time (perhaps look by year). Get ready for so, so many ggplot graphs.

```r
# First, I want to look at the salaries in USD by experience
# Just so I remember: EN Entry-level / Junior MI Mid-level / Intermediate SE Senior-level / Expert EX Executive-level

ds_salaries <- ds_salaries %>%
 mutate(experience_level = factor(recode(experience_level,
                                  EN = "Entry Level",     # I wanted to rename these codes so they made sense to me
                                  MI = "Mid-Level",       # I also looked up how to make sure they showed up in the right order
                                  SE = "Senior Level",
                                  EX = "Executive Level"),
                                levels = c("Entry Level", "Mid-Level", "Senior Level", "Executive Level")))

salary_by_experience <- ds_salaries %>%
  group_by(experience_level) %>%
  summarize(
    avg_salary = mean(salary_in_usd),
    median_salary = median(salary_in_usd),
    min_salary = min(salary_in_usd),
    max_salary = max(salary_in_usd)
  )

print(salary_by_experience)
```

```
## # A tibble: 4 × 5
##   experience_level avg_salary median_salary min_salary max_salary
##   <fct>                 <dbl>         <dbl>      <int>      <int>
## 1 Entry Level          61643.        56500        4000     250000
## 2 Mid-Level            87996.        76940        2859     450000
## 3 Senior Level        138617.       135500       18907     412000
## 4 Executive Level     199392.       171438.      69741     600000
```

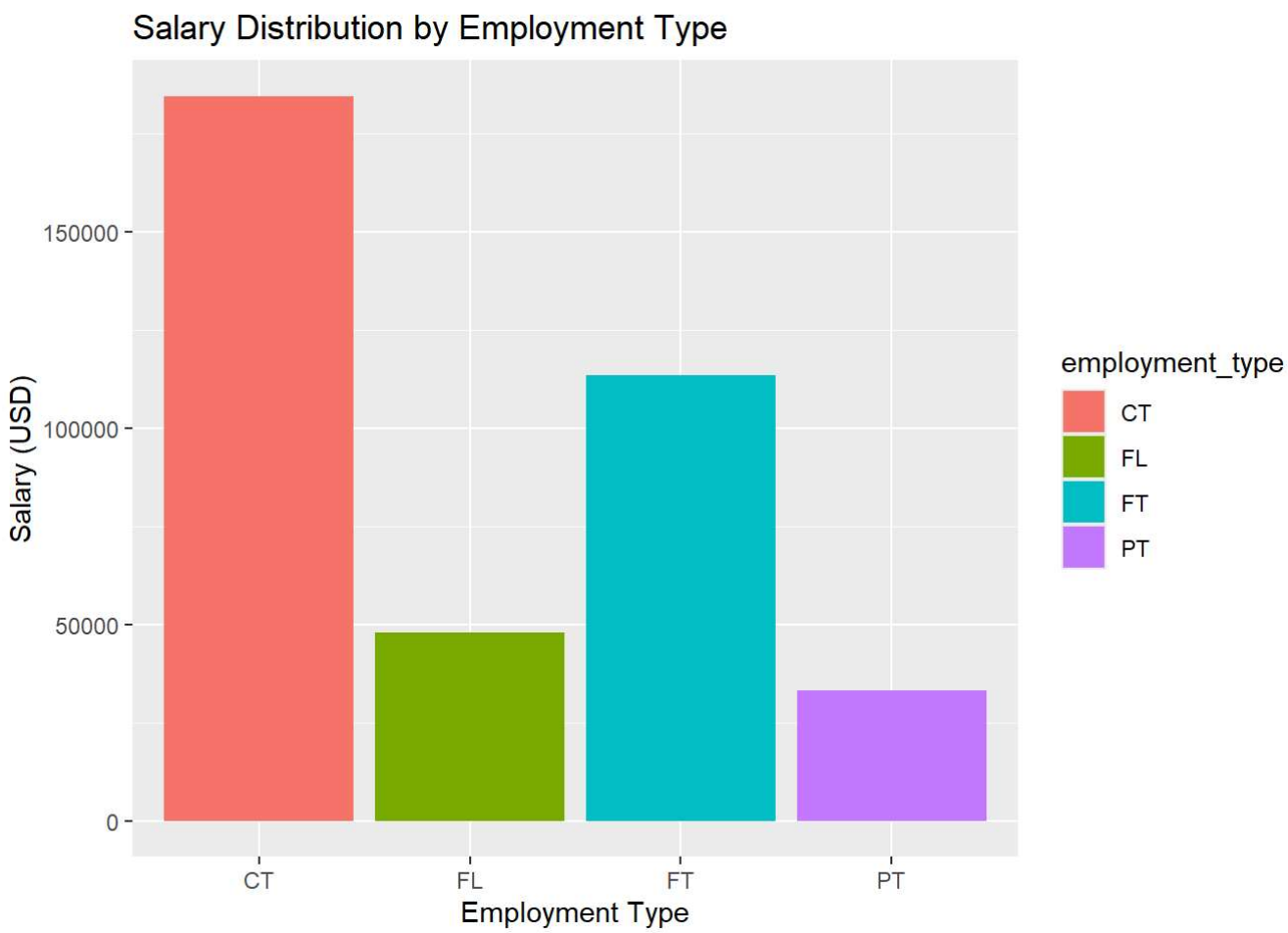
```r
ggplot(salary_by_experience, aes(x = experience_level, y = avg_salary, fill = experience_level)) +
  geom_col() +
  labs(title = "Salary Distribution by Experience Level",
       x = "Experience Level",
       y = "Salary (USD)")
```




```
# Next, I want to see whether employment type influences salaries
# PT Part-time / FT Full-time / CT Contract / FL Freelance

salary_by_employment <- ds_salaries %>%
  group_by(employment_type) %>%
  summarize(
    avg_salary = mean(salary_in_usd),
    median_salary = median(salary_in_usd),
    min_salary = min(salary_in_usd),
    max_salary = max(salary_in_usd)
  )

print(salary_by_employment)
```

```
## # A tibble: 4 × 5
##   employment_type avg_salary median_salary min_salary max_salary
##   <chr>                <dbl>         <dbl>      <int>      <int>
## 1 CT                 184575        105000       31875     416000
## 2 FL                  48000         40000       12000     100000
## 3 FT                 113468.       104196.       2859     600000
## 4 PT                  33070.        18818.       5409     100000
```

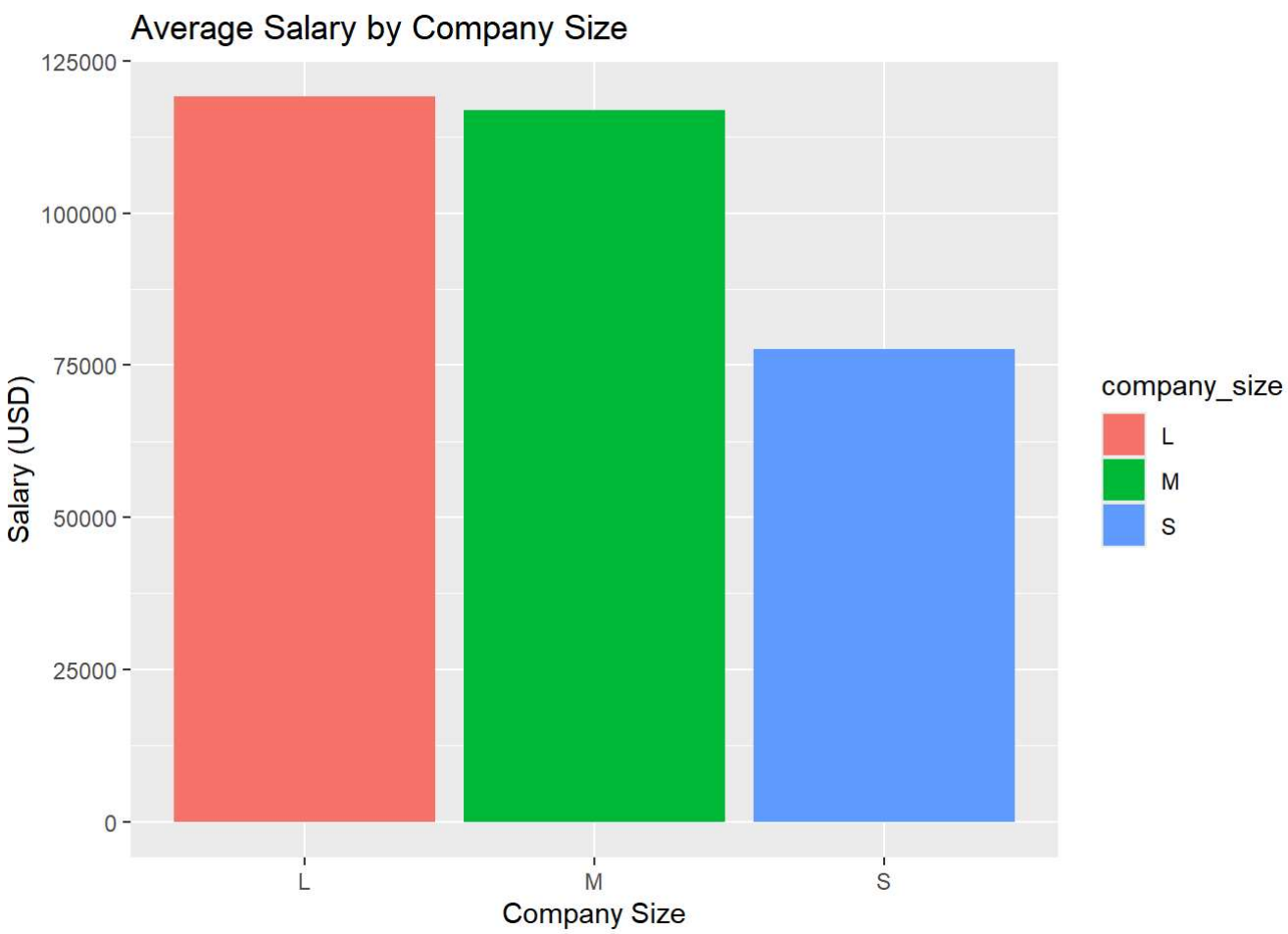
```
ggplot(salary_by_employment, aes(x = employment_type, y = avg_salary, fill = employment_type)) +
  geom_col() +
  labs(title = "Salary Distribution by Employment Type",
       x = "Employment Type",
       y = "Salary (USD)")
```




```
# How might the size of the company influence salaries?

salary_by_company_size <- ds_salaries %>%
  group_by(company_size) %>%
  summarize(
    avg_salary = mean(salary_in_usd),
    median_salary = median(salary_in_usd),
    min_salary = min(salary_in_usd),
    max_salary = max(salary_in_usd)
  )

print(salary_by_company_size)
```

```
## # A tibble: 3 × 5
##   company_size avg_salary median_salary min_salary max_salary
##   <chr>             <dbl>         <dbl>      <int>      <int>
## 1 L               119243.        100000       5882     600000
## 2 M               116905.        113188       4000     450000
## 3 S                77633.         65000       2859     416000
```

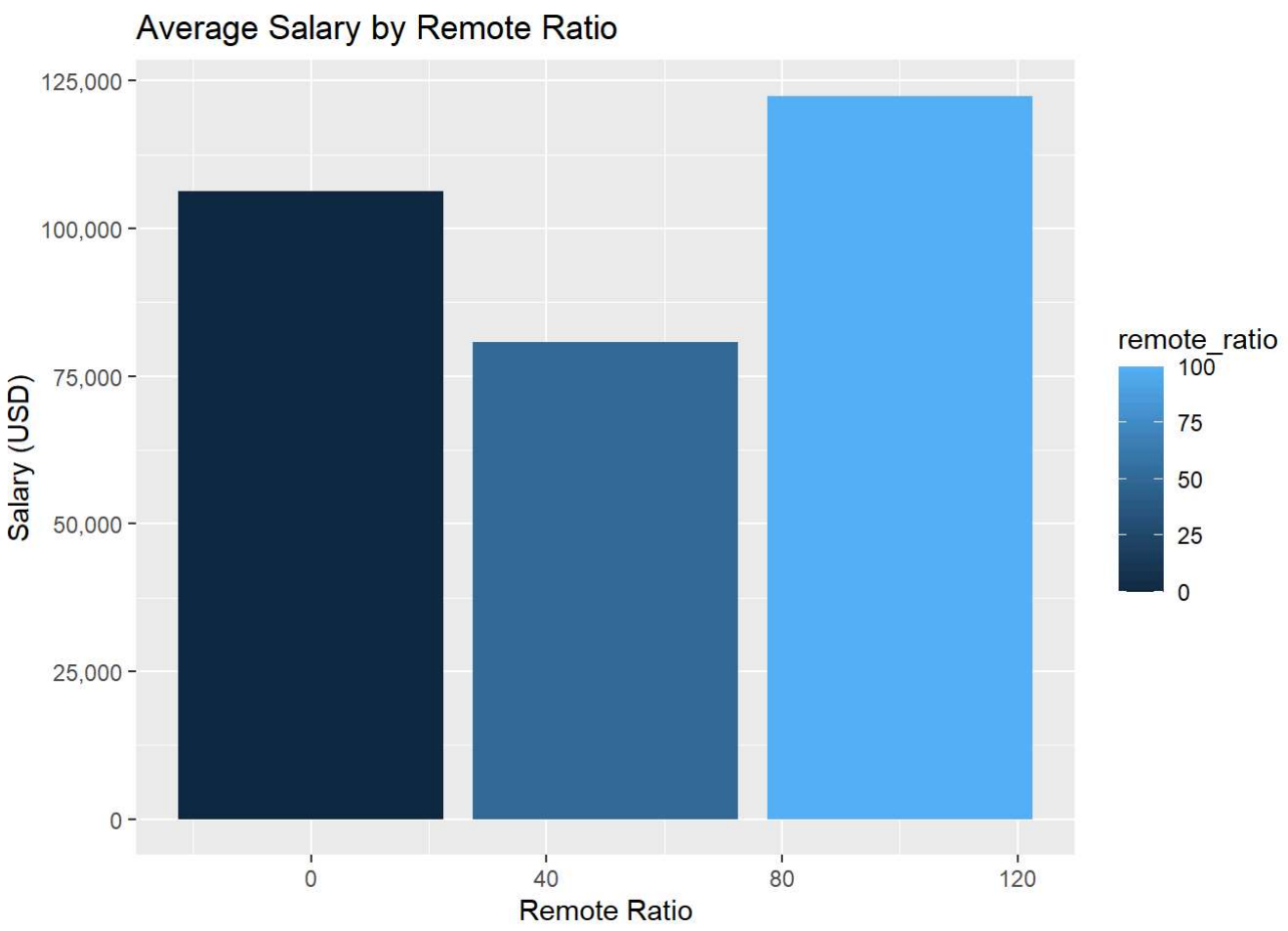
```
ggplot(salary_by_company_size, aes(x = company_size, y = avg_salary, fill = company_size)) +
  geom_col() +
  labs(title = "Average Salary by Company Size",
       x = "Company Size",
       y = "Salary (USD)")
```



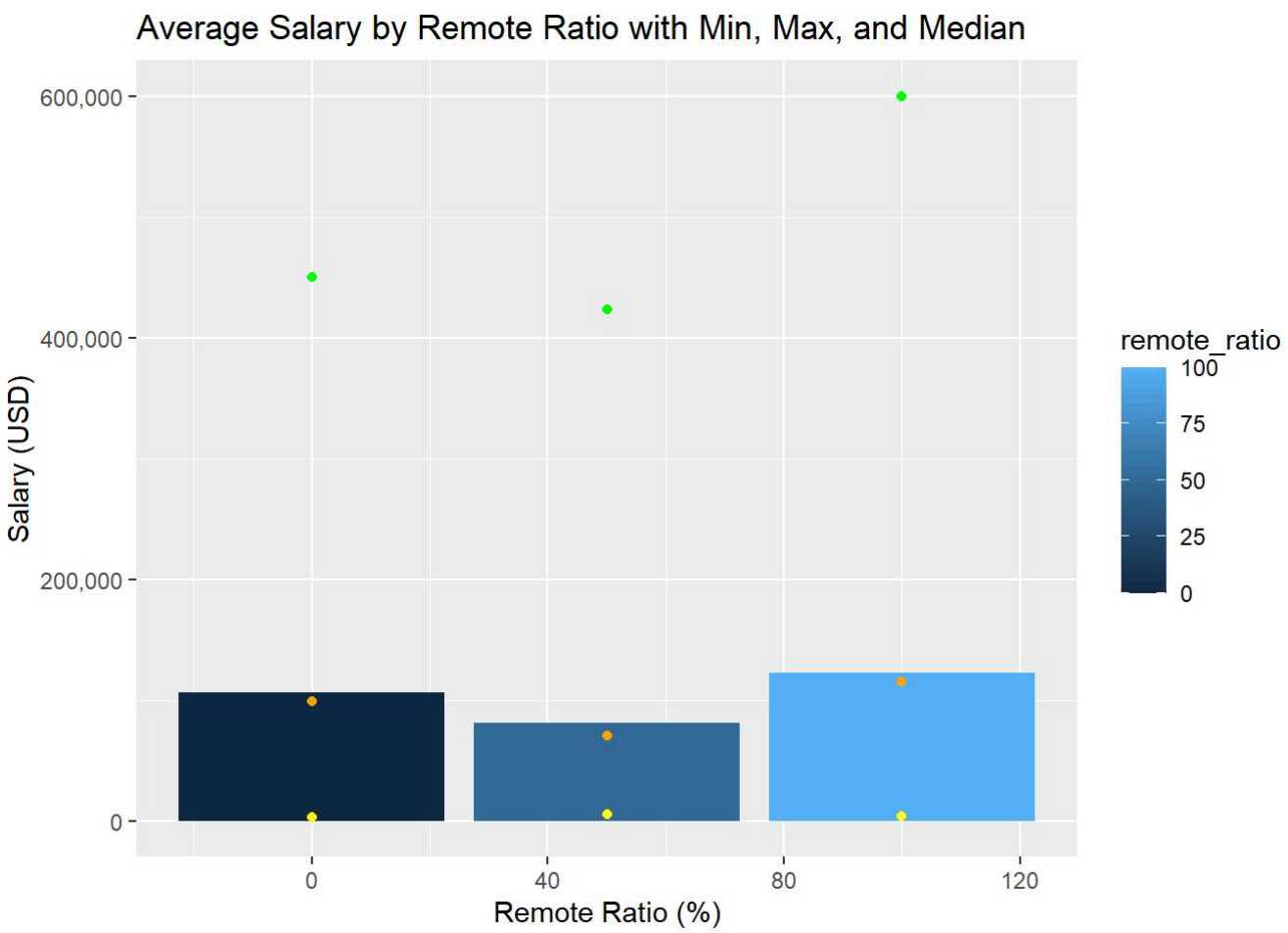

```
# What about whether the person works in the office, remotely, or hybrid?

salary_by_remote_ratio <- ds_salaries %>%
  group_by(remote_ratio) %>%
  summarize(
    avg_salary = mean(salary_in_usd),
    median_salary = median(salary_in_usd),
    min_salary = min(salary_in_usd),
    max_salary = max(salary_in_usd)
  )

print(salary_by_remote_ratio)
```

```
## # A tibble: 3 × 5
##   remote_ratio avg_salary median_salary min_salary max_salary
##          <int>      <dbl>         <int>      <int>      <int>
## 1            0    106355.         99000       2859     450000
## 2           50     80823.         69999       5409     423000
## 3          100    122457.        115000       4000     600000
```

```
ggplot(salary_by_remote_ratio, aes(x = remote_ratio, y = avg_salary, fill = remote_ratio)) +
  geom_col() +
  labs(title = "Average Salary by Remote Ratio",
       x = "Remote Ratio",
       y = "Salary (USD)") +
  scale_y_continuous(labels = function(x) format(x, big.mark = ",", scientific = FALSE))
```




```
# I didn't like the scientific notation so I asked ChatGPT how to change that

ggplot(salary_by_remote_ratio, aes(x = remote_ratio, y = avg_salary, fill = remote_ratio)) +
  geom_col() +
  geom_point(aes(y = min_salary), color = "yellow") +
  geom_point(aes(y = max_salary), color = "green") +
  geom_point(aes(y = median_salary), color = "orange") +
  labs(title = "Average Salary by Remote Ratio with Min, Max, and Median",
       x = "Remote Ratio (%)",
       y = "Salary (USD)") +
  scale_y_continuous(labels = function(x) format(x, big.mark = ",", scientific = FALSE))
```

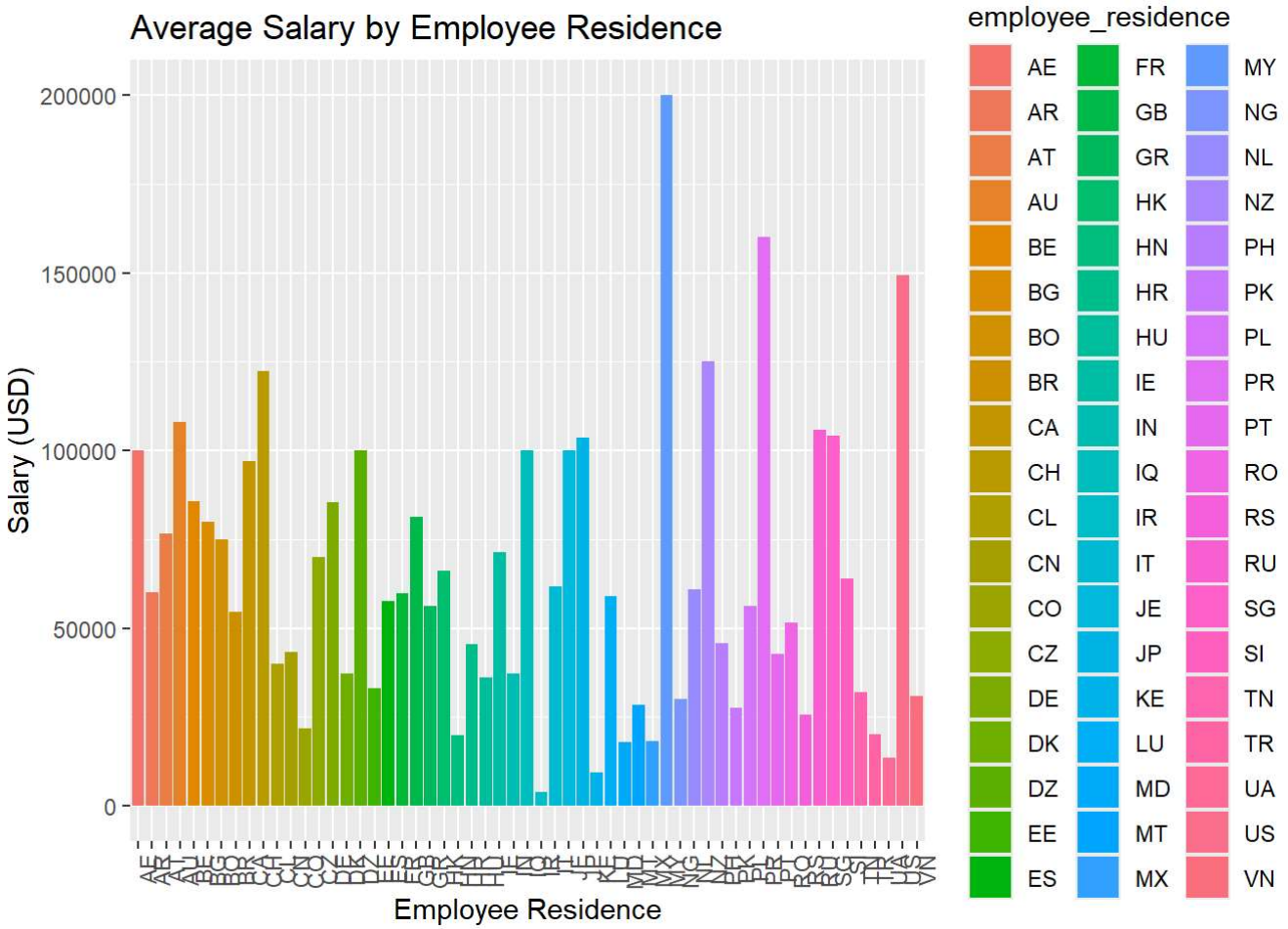



```
# That's not how I thought the salaries would trend!
```

```
# Does employee residence factor into salaries?
salary_by_residence <- ds_salaries %>%
  group_by(employee_residence) %>%
  summarize(
    avg_salary = mean(salary_in_usd),
    median_salary = median(salary_in_usd),
    min_salary = min(salary_in_usd),
    max_salary = max(salary_in_usd)
  )

print(salary_by_residence)
```

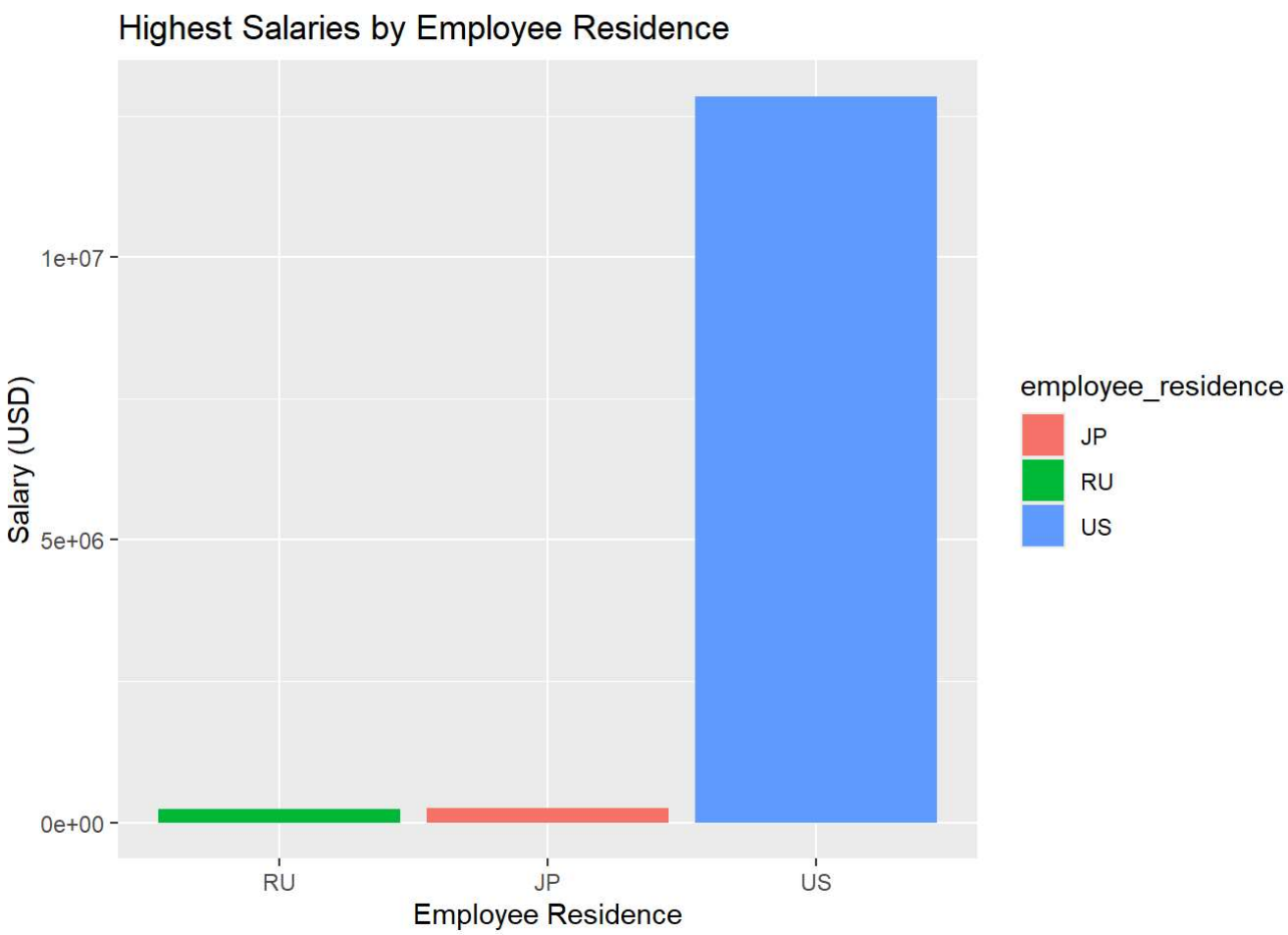
```
## # A tibble: 57 × 5
##    employee_residence avg_salary median_salary min_salary max_salary
##    <chr>                   <dbl>         <dbl>      <int>      <int>
##  1 AE                    100000        115000       65000     120000
##  2 AR                     60000         60000       60000      60000
##  3 AT                     76739.        74130       64849      91237
##  4 AU                    108043.        87425       86703     150000
##  5 BE                     85699         85699       82744      88654
##  6 BG                     80000         80000       80000      80000
##  7 BO                     75000         75000       75000      75000
##  8 BR                     54635.        21454.      12000     160000
##  9 CA                     97085.        85000       52000     196979
## 10 CH                    122346        122346      122346     122346
## # ℹ 47 more rows
```

```
ggplot(salary_by_residence, aes(x = employee_residence, y = avg_salary, fill = employee_residence)) +
  geom_col() +
  labs(title = "Average Salary by Employee Residence",
       x = "Employee Residence",
       y = "Salary (USD)") +
    theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



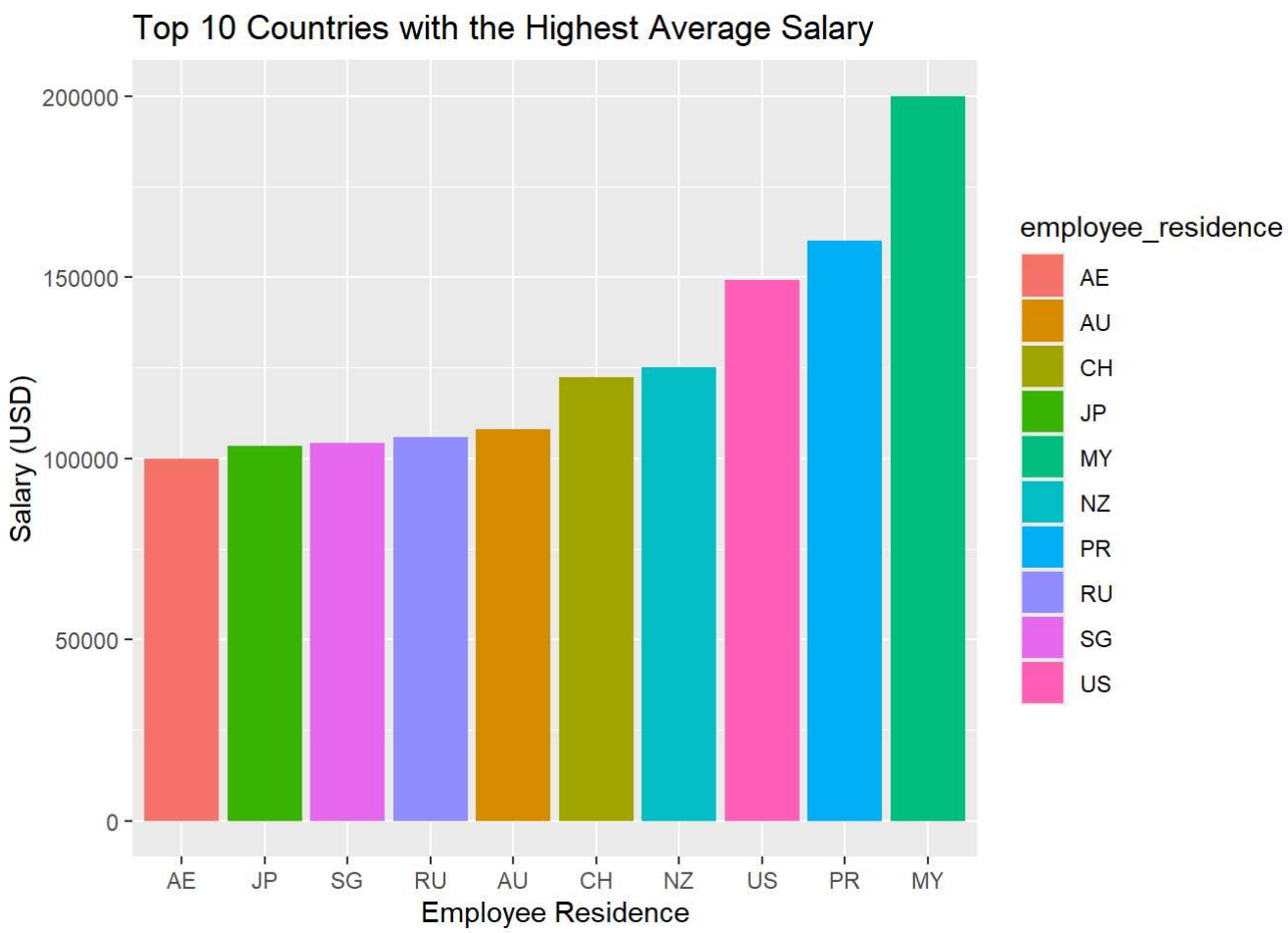

```
#This is an overwhelming plot. I'm going to try something else.
```

```
# Can I see the top 50 highest-paid people and where they happen to reside, instead?

top_50_salaries <- ds_salaries %>%
  arrange(desc(salary_in_usd)) %>%
  head(50)

ggplot(top_50_salaries, aes(x = reorder(employee_residence, salary_in_usd), y = salary_in_usd, fill = emplo
yee_residence)) +
  geom_bar(stat = "identity") +
  labs(title = "Highest Salaries by Employee Residence",
       x = "Employee Residence",
       y = "Salary (USD)")
```

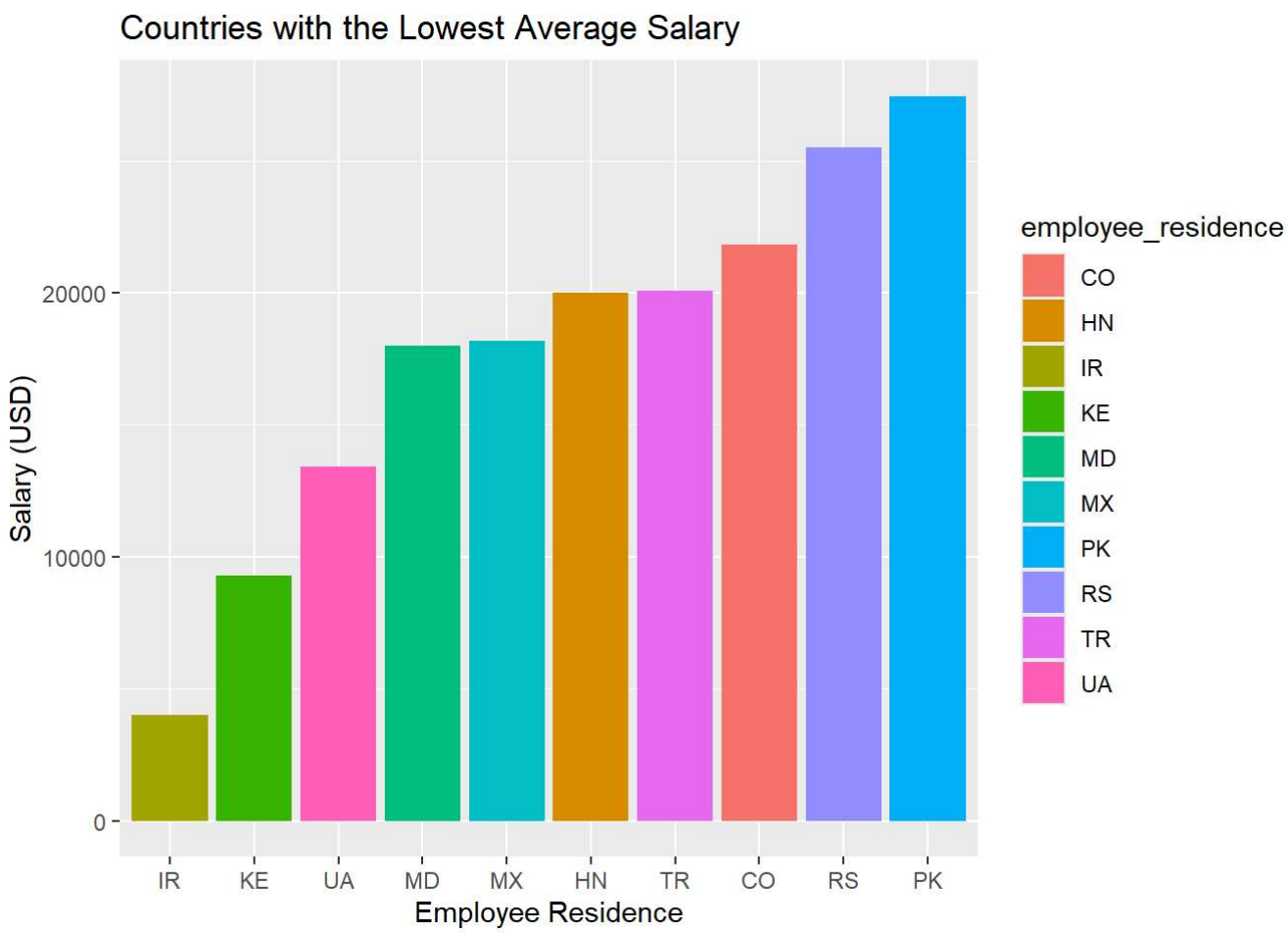



```
#I don't think I gleaned anything useful from this, but I'll keep it in here, just the same.
```

```
# Looking at just the countries with the highest average salary

top_10_salaries <- salary_by_residence %>%
  arrange(desc(avg_salary)) %>%
  head(10)

ggplot(top_10_salaries, aes(x = reorder(employee_residence, avg_salary), y = avg_salary, fill = employee_re
sidence)) +
  geom_col() +
  labs(title = "Top 10 Countries with the Highest Average Salary",
       x = "Employee Residence",
       y = "Salary (USD)")
```




```
# Now I'm just being nosy

btm_10_salaries <- salary_by_residence %>%
  arrange(avg_salary) %>%
  head(10)

ggplot(btm_10_salaries, aes(x = reorder(employee_residence, avg_salary), y = avg_salary, fill = employee_re
sidence)) +
  geom_col() +
  labs(title = "Countries with the Lowest Average Salary",
       x = "Employee Residence",
       y = "Salary (USD)")
```



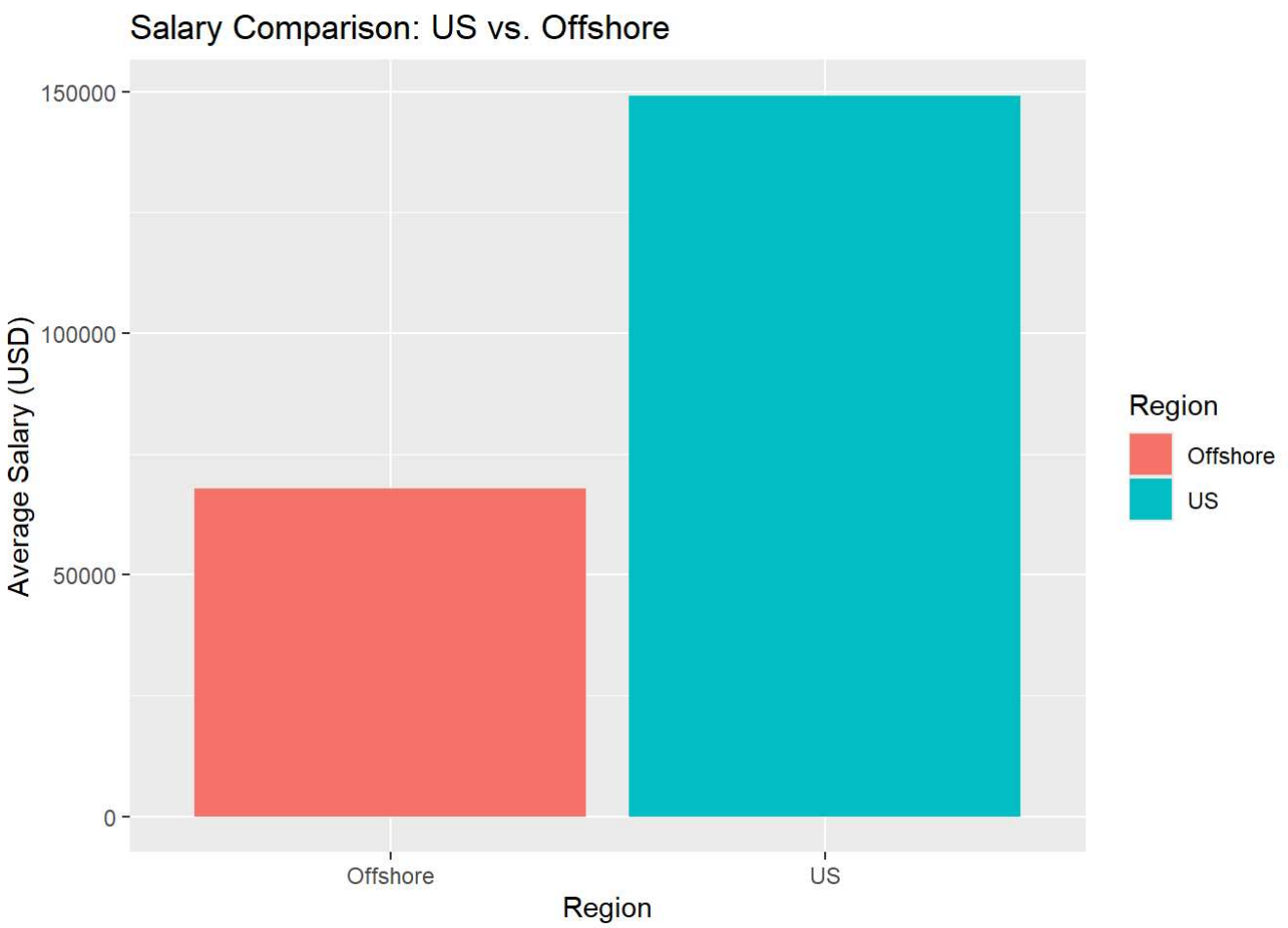

```
# I want to see how US salaries compared with non-US salaries overall

us_data <- ds_salaries %>% filter(employee_residence == "US")
offshore_data <- ds_salaries %>% filter(employee_residence != "US")

us_salary_avg <- mean(us_data$salary_in_usd)
offshore_salary_avg <- mean(offshore_data$salary_in_usd)

print(us_salary_avg)
```

```
## [1] 149194.1
```

```
print(offshore_salary_avg)
```

```
## [1] 67754.04
```

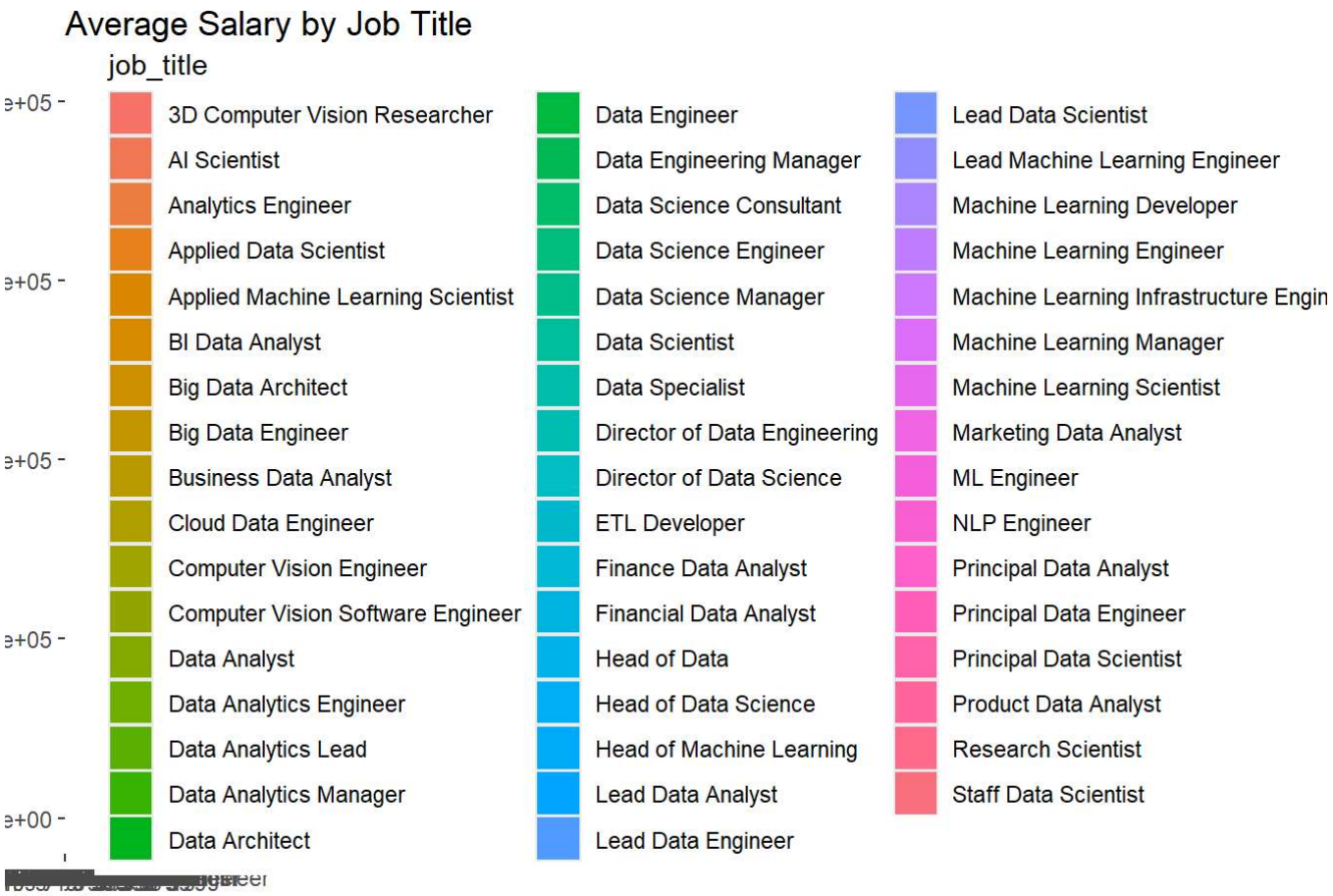
```
salary_comparison <- data.frame(      # I asked the internet for help fixing this one up
  Region = c("US", "Offshore"),
  Avg_Salary = c(us_salary_avg, offshore_salary_avg)
)

ggplot(salary_comparison, aes(x = Region, y = Avg_Salary, fill = Region)) +
  geom_bar(stat = "identity") +
  labs(title = "Salary Comparison: US vs. Offshore",
       x = "Region",
       y = "Average Salary (USD)")
```




```
# Does job title make a difference?

salary_by_job_title <- ds_salaries %>%
  group_by(job_title) %>%
  summarize(
    avg_salary = mean(salary_in_usd),
    median_salary = median(salary_in_usd),
    min_salary = min(salary_in_usd),
    max_salary = max(salary_in_usd)
  )

print(salary_by_job_title)
```

```
## # A tibble: 50 × 5
##    job_title                     avg_salary median_salary min_salary max_salary
##    <chr>                              <dbl>         <dbl>      <int>      <int>
##  1 3D Computer Vision Researcher       5409          5409        5409       5409
##  2 AI Scientist                       66136.        45896       12000     200000
##  3 Analytics Engineer                175000        179850      135000     205300
##  4 Applied Data Scientist            175655        157000       54238     380000
##  5 Applied Machine Learning Scie…    142069.        56700       31875     423000
##  6 BI Data Analyst                    74755.        76500        9272     150000
##  7 Big Data Architect                 99703         99703       99703      99703
##  8 Big Data Engineer                  51974         41306.       5882     114047
##  9 Business Data Analyst              76691.        70912       18442     135000
## 10 Cloud Data Engineer               124647        124647       89294     160000
## # ℹ 40 more rows
```

```
ggplot(salary_by_job_title, aes(x = job_title, y = avg_salary, fill = job_title)) +
  geom_col() +
  labs(title = "Average Salary by Job Title",
       x = "Job Title",
       y = "Salary (USD)")
```




```
# I can't really tell by this graph
# Also didn't realize there were so many job titles
# I think cleaning this data is a little beyond my scope at the moment
```

```
# I asked ChatGPT for alternative ways to display this data

library(knitr)
kable(salary_by_job_title, caption = "Average Salary by Job Title")
```

Average Salary by Job Title

job_title	avg_salary	median_salary	min_salary	max_salary
3D Computer Vision Researcher	5409.00	5409.0	5409	5409
AI Scientist	66135.57	45896.0	12000	200000
Analytics Engineer	175000.00	179850.0	135000	205300
Applied Data Scientist	175655.00	157000.0	54238	380000
Applied Machine Learning Scientist	142068.75	56700.0	31875	423000

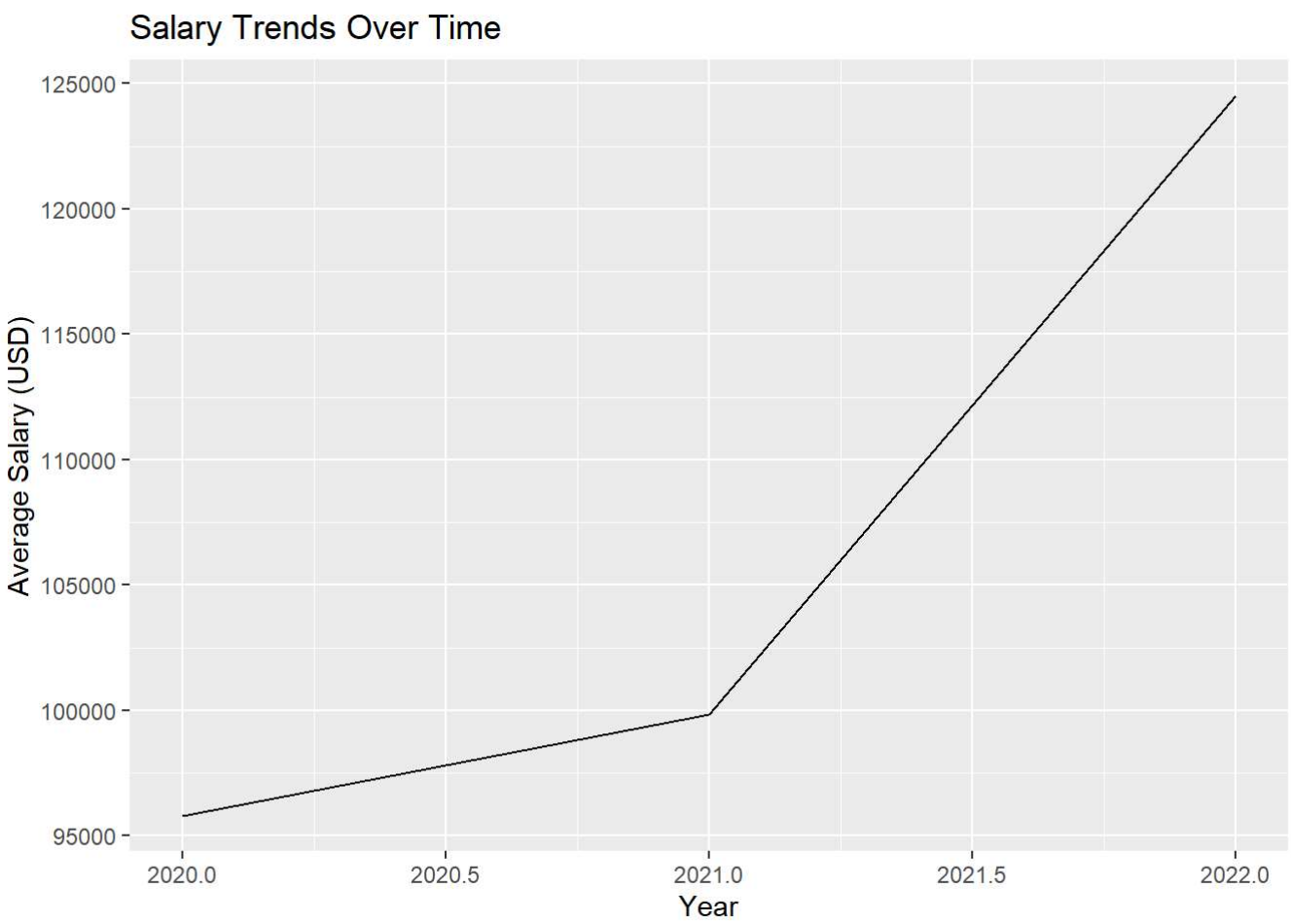
job_title	avg_salary	median_salary	min_salary	max_salary
BI Data Analyst	74755.17	76500.0	9272	150000
Big Data Architect	99703.00	99703.0	99703	99703
Big Data Engineer	51974.00	41305.5	5882	114047
Business Data Analyst	76691.20	70912.0	18442	135000
Cloud Data Engineer	124647.00	124647.0	89294	160000
Computer Vision Engineer	44419.33	26304.5	10000	125000
Computer Vision Software Engineer	105248.67	95746.0	70000	150000
Data Analyst	92893.06	90320.0	6072	200000
Data Analytics Engineer	64799.25	64598.5	20000	110000
Data Analytics Lead	405000.00	405000.0	405000	405000
Data Analytics Manager	127134.29	120000.0	105400	150260
Data Architect	177873.91	180000.0	90700	266400
Data Engineer	112725.00	105500.0	4000	324000
Data Engineering Manager	123227.20	150000.0	59303	174000
Data Science Consultant	69420.71	76833.0	5707	103000
Data Science Engineer	75803.33	60000.0	40189	127221
Data Science Manager	158328.50	155750.0	54094	241000
Data Scientist	108187.83	103691.0	2859	412000
Data Specialist	165000.00	165000.0	165000	165000
Director of Data Engineering	156738.00	156738.0	113476	200000
Director of Data Science	195074.00	168000.0	130026	325000
ETL Developer	54957.00	54957.0	54957	54957
Finance Data Analyst	61896.00	61896.0	61896	61896
Financial Data Analyst	275000.00	275000.0	100000	450000
Head of Data	160162.60	200000.0	32974	235000
Head of Data Science	146718.75	138937.5	85000	224000
Head of Machine Learning	79039.00	79039.0	79039	79039
Lead Data Analyst	92203.00	87000.0	19609	170000
Lead Data Engineer	139724.50	121593.5	56000	276000
Lead Data Scientist	115190.00	115000.0	40570	190000
Lead Machine Learning Engineer	87932.00	87932.0	87932	87932
ML Engineer	117504.00	70537.5	15966	270000
Machine Learning Developer	85860.67	78791.0	78791	100000

job_title	avg_salary	median_salary	min_salary	max_salary
Machine Learning Engineer	104880.15	87932.0	20000	250000
Machine Learning Infrastructure Engineer	101145.00	58255.0	50180	195000
Machine Learning Manager	117104.00	117104.0	117104	117104
Machine Learning Scientist	158412.50	156500.0	12000	260000
Marketing Data Analyst	88654.00	88654.0	88654	88654
NLP Engineer	37236.00	37236.0	37236	37236
Principal Data Analyst	122500.00	122500.0	75000	170000
Principal Data Engineer	328333.33	200000.0	185000	600000
Principal Data Scientist	215242.43	173762.0	148261	416000
Product Data Analyst	13036.00	13036.0	6072	20000
Research Scientist	109019.50	76263.5	42000	450000
Staff Data Scientist	105000.00	105000.0	105000	105000

```
# I want to look at salary trends over time, now that I have a basic idea of how various factors have influ
enced average salaries
# This chart calculates the average, median, minimum, and maximum salaries for all individuals in the data
frame by the year each salary was paid

salary_by_year <- ds_salaries %>%
  group_by(work_year) %>%
  summarize(
    avg_salary = mean(salary_in_usd),
    median_salary = median(salary_in_usd),
    min_salary = min(salary_in_usd),
    max_salary = max(salary_in_usd)
  )

print(salary_by_year)
```

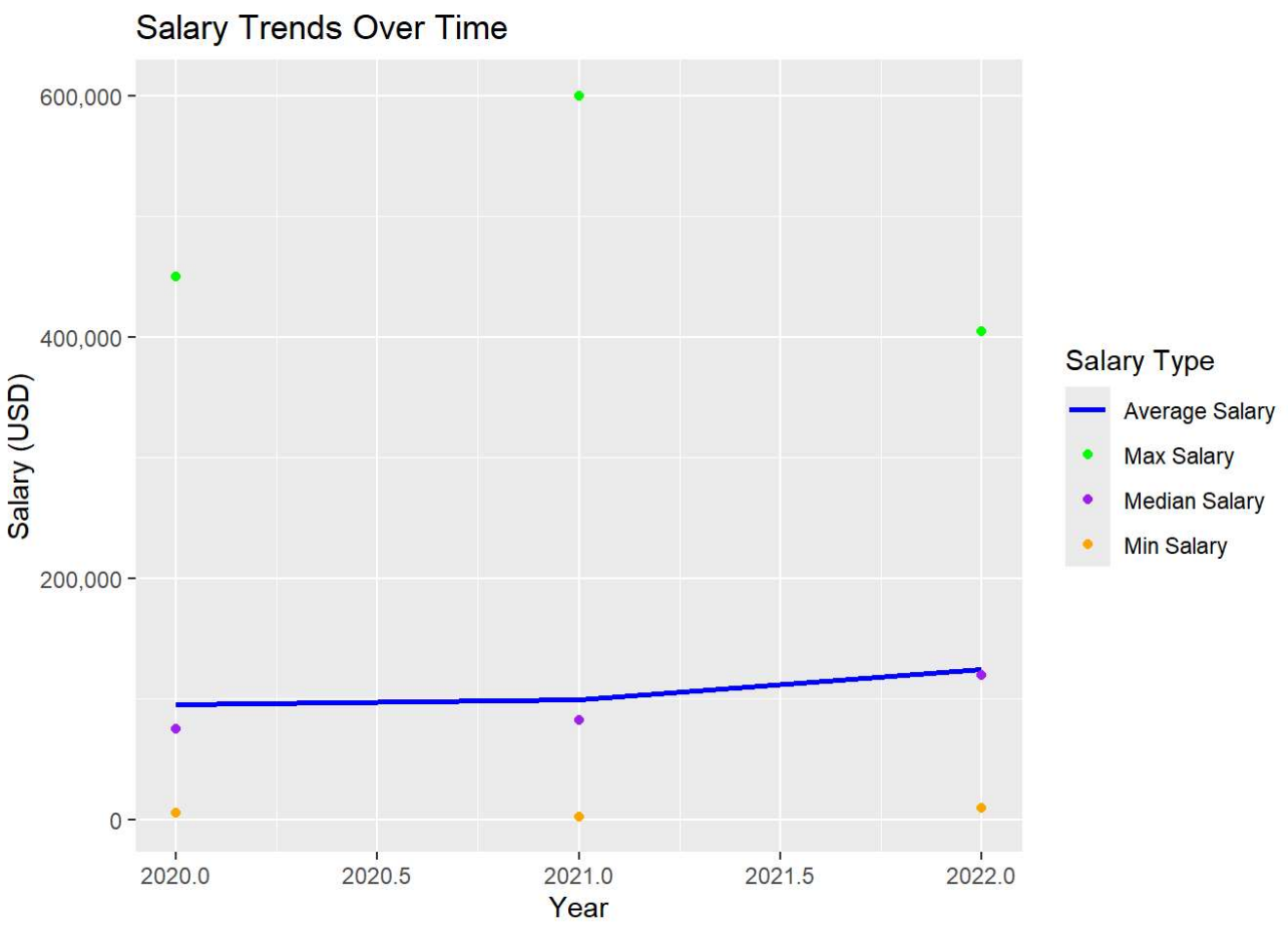
```
## # A tibble: 3 × 5
##   work_year avg_salary median_salary min_salary max_salary
##       <int>      <dbl>         <dbl>      <int>      <int>
## 1      2020     95813          75544       5707     450000
## 2      2021     99854.         82528       2859     600000
## 3      2022    124522.        120000      10000     405000
```

```
ggplot(salary_by_year, aes(x = work_year, y = avg_salary)) +
  geom_line() +
  labs(title = "Salary Trends Over Time",
       x = "Year",
       y = "Average Salary (USD)")
```



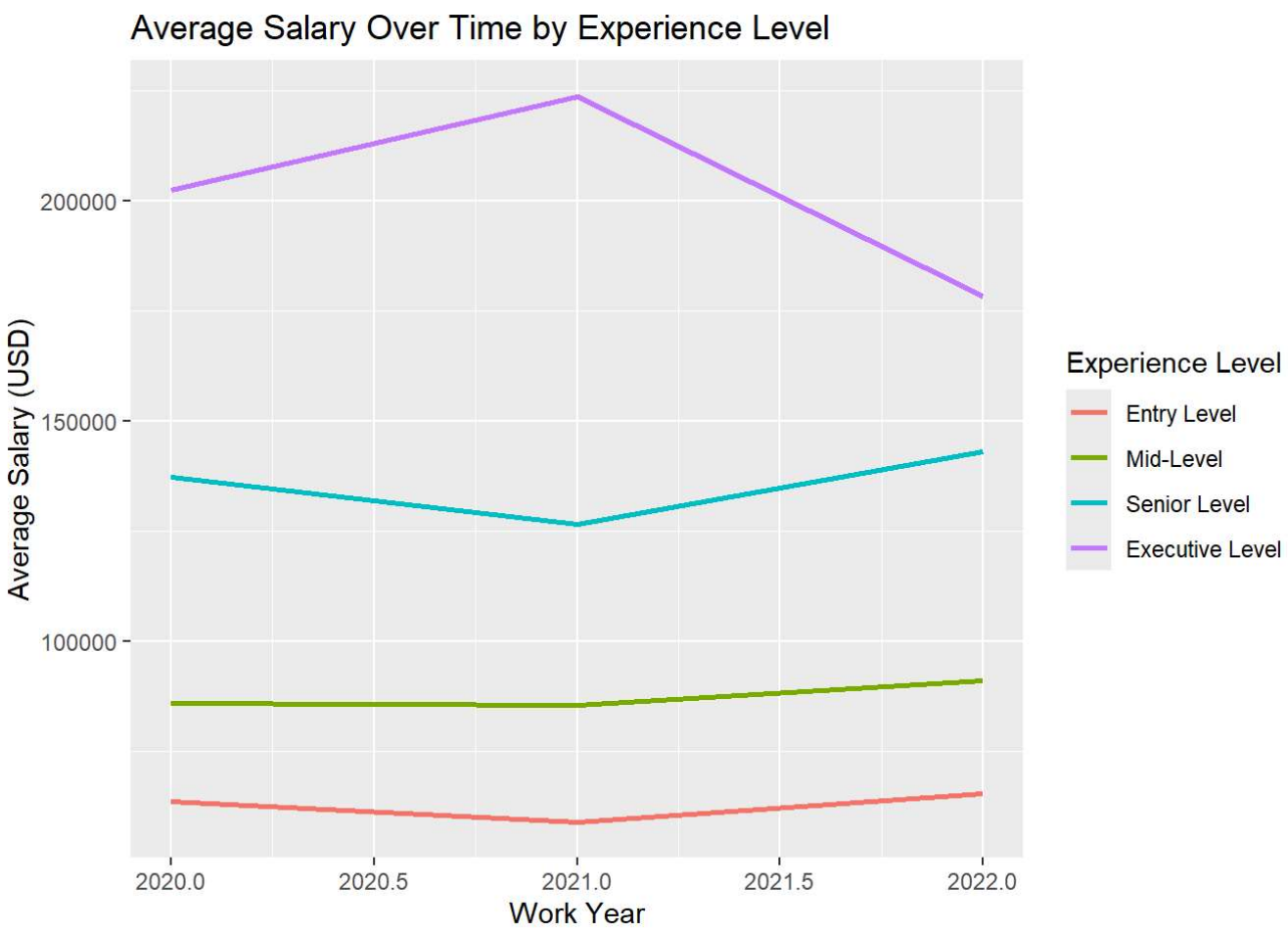

```
# and that is a massive leap from 2021 to 2022!
```

```
# I wanted to create a plot that also showed the minimum, maximum, and median salaries, so I consulted abou
t this one
# The change in average salary doesn't look as severe with this view because of the inclusion of the max sa
laries
# There was one individual who make $600k in 2021
# It is interesting to see that, while the average salaries are highest in 2022, the maximum salaries are t
he lowest of the 3 years
# Viewing the data this way shows that the overall upward trend in salaries isn't as steep as it looked in
the plot above, when we were only looking at the year

ggplot(salary_by_year, aes(x = work_year)) +
  geom_line(aes(y = avg_salary, color = "Average Salary"), linewidth = 1) +
  geom_point(aes(y = min_salary, color = "Min Salary")) +
  geom_point(aes(y = max_salary, color = "Max Salary")) +
  geom_point(aes(y = median_salary, color = "Median Salary")) +
  labs(title = "Salary Trends Over Time",
       x = "Year",
       y = "Salary (USD)",
       color = "Salary Type") +
  scale_color_manual(values = c("Average Salary" = "blue",
                                "Min Salary" = "orange",
                                "Max Salary" = "green",
                                "Median Salary" = "purple")) +
  scale_y_continuous(labels = function(x) format(x, big.mark = ",", scientific = FALSE))
```



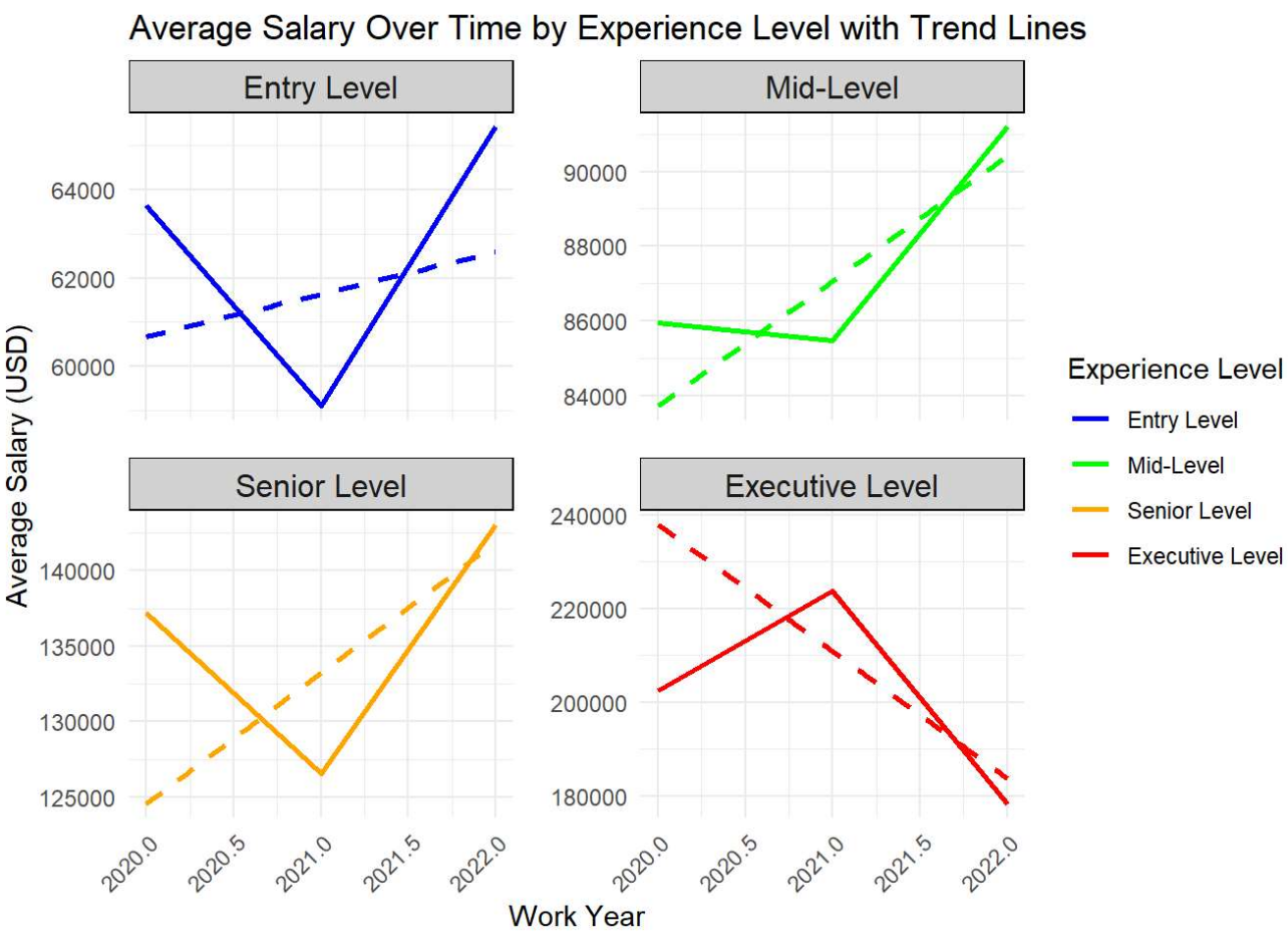

```
# I want to consolidate some of the information I'd gathered from above
# This one is supposed to show the average salaries by the experience level over time
ggplot(ds_salaries, aes(x = work_year, y = salary_in_usd, color = experience_level)) +
  geom_line(stat = "summary", fun = "mean", size = 1) +
  labs(
    title = "Average Salary Over Time by Experience Level",
    x = "Work Year",
    y = "Average Salary (USD)",
    color = "Experience Level"
  )
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## ℹ Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```


Average Salary Over Time by Experience Level
Average Salary (USD)
200000
150000
100000
2020.0
2020.5
2021.0
2021.5
2022.0
Work Year
Experience Level
Entry Level
Mid-Level
Senior Level
Executive Level

```
# I wanted to take a look at this same information broken down a little further
# Clearly, I asked ChatGPT for help with this one, too, because I couldn't make it work on my own and never would have figured this out
# But I wanted to see the salary trends for each experience level a little more closely so we can make some business decisions that way


ggplot(ds_salaries, aes(x = work_year, y = salary_in_usd, color = experience_level)) +
  geom_line(stat = "summary", fun = "mean", size = 1) +  # Plot average salary by year
  geom_smooth(method = "lm", aes(group = experience_level), se = FALSE, linetype = "dashed", size = 1) +  # Add trend lines
  facet_wrap(~ experience_level, scales = "free_y", ncol = 2) +  # Facets with free y-axis scales and 2 columns
  labs(
    title = "Average Salary Over Time by Experience Level with Trend Lines",
    x = "Work Year",
    y = "Average Salary (USD)",
    color = "Experience Level"
  ) +
  scale_color_manual(values = c("Entry Level" = "blue",
                                "Mid-Level" = "green",
                                "Senior Level" = "orange",
                                "Executive Level" = "red")) +  # Custom colors for experience levels
  theme_minimal() +  # Clean theme for better visualization
  theme(
    strip.text = element_text(size = 12),   # Adjust size of facet labels
    strip.background = element_rect(fill = "lightgray"),  # Adjust background of facet labels
    panel.spacing = unit(1, "lines"),  # Add space between panels
    axis.text.x = element_text(angle = 45, hjust = 1)  # Rotate x-axis labels for better readability
  )
```

```
## `geom_smooth()` using formula = 'y ~ x'
```


Average Salary Over Time by Experience Level with Trend Lines
Entry Level
Mid-Level
Senior Level
Executive Level
Average Salary (USD)
64000
62000
60000
90000
88000
86000
84000
140000
135000
130000
125000
240000
220000
200000
180000
2020.0
2020.5
2021.0
2021.5
2022.0
Work Year
Experience Level
Entry Level
Mid-Level
Senior Level
Executive Level