DS\_salaries

2024-01-31

# Data Exploration and Cleaning:  
# Loaded the dataset  
df <- read.csv("ds\_salaries.csv")

# Explored dataset structure  
str(df)

## '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" ...

summary(df)

## work\_year experience\_level employment\_type job\_title   
## Min. :2020 Length:3755 Length:3755 Length:3755   
## 1st Qu.:2022 Class :character Class :character Class :character   
## Median :2022 Mode :character Mode :character Mode :character   
## Mean :2022   
## 3rd Qu.:2023   
## Max. :2023   
## salary salary\_currency salary\_in\_usd employee\_residence  
## Min. : 6000 Length:3755 Min. : 5132 Length:3755   
## 1st Qu.: 100000 Class :character 1st Qu.: 95000 Class :character   
## Median : 138000 Mode :character Median :135000 Mode :character   
## Mean : 190696 Mean :137570   
## 3rd Qu.: 180000 3rd Qu.:175000   
## Max. :30400000 Max. :450000   
## remote\_ratio company\_location company\_size   
## Min. : 0.00 Length:3755 Length:3755   
## 1st Qu.: 0.00 Class :character Class :character   
## Median : 0.00 Mode :character Mode :character   
## Mean : 46.27   
## 3rd Qu.:100.00   
## Max. :100.00

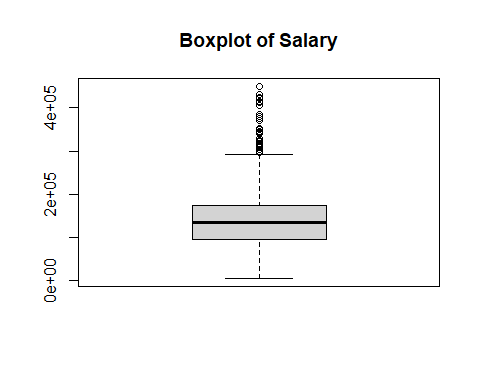
# Checked for missing values  
sum(is.na(df)) # no missing value found

## [1] 0

# Checked for extreme values  
summary(df$salary\_in\_usd)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 5132 95000 135000 137570 175000 450000

# Detected Outliers  
boxplot(df$salary\_in\_usd, main = "Boxplot of Salary")



# Calculated quartiles and IQR  
Q1 <- quantile(df$salary\_in\_usd, 0.25)  
Q3 <- quantile(df$salary\_in\_usd, 0.75)  
IQR <- Q3 - Q1

# Defined the threshold for outliers  
threshold <- 1.5 \* IQR

# Identified outliers  
outliers <- df$salary\_in\_usd < (Q1 - threshold) | df$salary\_in\_usd > (Q3 + threshold)

# Print rows corresponding to outliers  
print(df[outliers, ])

## work\_year experience\_level employment\_type  
## 34 2023 SE FT  
## 69 2023 SE FT  
## 84 2022 EN FT  
## 134 2023 SE FT  
## 146 2023 SE FT  
## 164 2023 SE FT  
## 191 2023 MI FT  
## 229 2023 EX FT  
## 359 2023 SE FT  
## 479 2023 EX FT  
## 483 2023 SE FT  
## 489 2023 SE FT  
## 529 2023 SE FT  
## 650 2023 SE FT  
## 688 2023 SE FT  
## 717 2023 SE FT  
## 794 2023 SE FT  
## 846 2023 MI FT  
## 861 2023 EX FT  
## 1008 2023 EX FT  
## 1098 2023 SE FT  
## 1100 2023 SE FT  
## 1106 2023 SE FT  
## 1117 2023 SE FT  
## 1132 2023 SE FT  
## 1154 2023 EX FT  
## 1259 2022 SE FT  
## 1287 2023 SE FT  
## 1289 2023 SE FT  
## 1312 2023 SE FT  
## 1397 2023 EX FT  
## 1422 2023 SE FT  
## 1428 2023 EX FT  
## 1459 2023 SE FT  
## 1559 2023 SE FT  
## 1594 2023 SE FT  
## 1606 2023 EX FT  
## 1676 2023 SE FT  
## 1678 2023 SE FT  
## 1723 2023 SE FT  
## 1933 2022 EX FT  
## 2012 2022 MI FT  
## 2163 2022 SE FT  
## 2280 2022 EX FT  
## 2282 2022 SE FT  
## 2332 2022 SE FT  
## 2360 2022 SE FT  
## 2375 2022 SE FT  
## 2407 2022 SE FT  
## 2503 2022 SE FT  
## 2556 2022 SE FT  
## 2671 2022 SE FT  
## 2833 2022 EX FT  
## 2857 2022 SE FT  
## 3153 2022 SE FT  
## 3411 2022 EX FT  
## 3464 2022 SE FT  
## 3469 2022 SE FT  
## 3523 2020 MI FT  
## 3676 2021 EX CT  
## 3698 2020 EX FT  
## 3748 2021 MI FT  
## 3751 2020 SE FT  
## job\_title salary salary\_currency salary\_in\_usd  
## 34 Computer Vision Engineer 342810 USD 342810  
## 69 Applied Scientist 309400 USD 309400  
## 84 AI Developer 300000 USD 300000  
## 134 Machine Learning Engineer 342300 USD 342300  
## 146 Machine Learning Engineer 318300 USD 318300  
## 164 Applied Scientist 309400 USD 309400  
## 191 Machine Learning Engineer 300000 USD 300000  
## 229 Head of Data 329500 USD 329500  
## 359 Machine Learning Engineer 304000 USD 304000  
## 479 Director of Data Science 353200 USD 353200  
## 483 Data Scientist 297300 USD 297300  
## 489 Data Scientist 317070 USD 317070  
## 529 AI Scientist 1500000 ILS 423834  
## 650 Data Architect 376080 USD 376080  
## 688 Data Science Manager 299500 USD 299500  
## 717 Data Scientist 297300 USD 297300  
## 794 Data Science Manager 299500 USD 299500  
## 846 Research Scientist 340000 USD 340000  
## 861 Data Engineer 310000 USD 310000  
## 1008 Data Engineer 310000 USD 310000  
## 1098 Data Scientist 300240 USD 300240  
## 1100 Data Scientist 300240 USD 300240  
## 1106 Data Scientist 370000 USD 370000  
## 1117 Machine Learning Engineer 323300 USD 323300  
## 1132 Data Science Manager 299500 USD 299500  
## 1154 Data Engineer 310000 USD 310000  
## 1259 Machine Learning Software Engineer 375000 USD 375000  
## 1287 Machine Learning Engineer 318300 USD 318300  
## 1289 Data Analyst 385000 USD 385000  
## 1312 Research Scientist 370000 USD 370000  
## 1397 Head of Data Science 314100 USD 314100  
## 1422 Applied Scientist 350000 USD 350000  
## 1428 Data Engineer 310000 USD 310000  
## 1459 Data Engineer 300000 USD 300000  
## 1559 Data Science Manager 299500 USD 299500  
## 1594 Data Engineer 300000 USD 300000  
## 1606 Data Scientist 300000 USD 300000  
## 1676 Data Science Manager 297300 USD 297300  
## 1678 Data Scientist 297300 USD 297300  
## 1723 Data Engineer 310000 USD 310000  
## 1933 Data Engineer 310000 USD 310000  
## 2012 Data Analyst 350000 GBP 430967  
## 2163 Data Engineer 300000 USD 300000  
## 2280 Data Engineer 310000 USD 310000  
## 2282 Data Science Manager 299500 USD 299500  
## 2332 Research Scientist 300000 USD 300000  
## 2360 Data Science Tech Lead 375000 USD 375000  
## 2375 Data Scientist 350000 USD 350000  
## 2407 Data Engineer 315000 USD 315000  
## 2503 Data Engineer 300000 USD 300000  
## 2556 Data Architect 345600 USD 345600  
## 2671 Data Engineer 300000 USD 300000  
## 2833 Data Engineer 297500 USD 297500  
## 2857 Data Engineer 300000 USD 300000  
## 3153 Data Science Manager 300000 USD 300000  
## 3411 Data Engineer 324000 USD 324000  
## 3464 Data Analytics Lead 405000 USD 405000  
## 3469 Applied Data Scientist 380000 USD 380000  
## 3523 Research Scientist 450000 USD 450000  
## 3676 Principal Data Scientist 416000 USD 416000  
## 3698 Director of Data Science 325000 USD 325000  
## 3748 Applied Machine Learning Scientist 423000 USD 423000  
## 3751 Data Scientist 412000 USD 412000  
## employee\_residence remote\_ratio company\_location company\_size  
## 34 US 0 US M  
## 69 US 0 US L  
## 84 IN 50 IN L  
## 134 US 0 US L  
## 146 US 100 US M  
## 164 US 0 US L  
## 191 US 0 US M  
## 229 US 0 US M  
## 359 US 100 US M  
## 479 US 0 US M  
## 483 US 100 US M  
## 489 US 0 US M  
## 529 IL 0 IL L  
## 650 US 100 US M  
## 688 US 0 US M  
## 717 US 100 US M  
## 794 US 0 US M  
## 846 US 100 US M  
## 861 US 100 US M  
## 1008 US 100 US M  
## 1098 US 0 US M  
## 1100 US 0 US M  
## 1106 US 0 US M  
## 1117 US 0 US M  
## 1132 US 0 US M  
## 1154 US 100 US M  
## 1259 US 100 US M  
## 1287 US 100 US M  
## 1289 US 0 US M  
## 1312 US 0 US M  
## 1397 US 0 US M  
## 1422 US 0 US L  
## 1428 US 100 US M  
## 1459 US 0 US M  
## 1559 US 0 US M  
## 1594 US 0 US M  
## 1606 US 0 US M  
## 1676 US 100 US M  
## 1678 US 100 US M  
## 1723 US 0 US M  
## 1933 US 100 US M  
## 2012 GB 0 GB M  
## 2163 US 0 US M  
## 2280 US 100 US M  
## 2282 US 0 US M  
## 2332 US 100 US M  
## 2360 US 50 US L  
## 2375 US 100 US M  
## 2407 US 100 US M  
## 2503 US 0 US M  
## 2556 US 0 US M  
## 2671 US 0 US M  
## 2833 US 100 US M  
## 2857 US 0 US M  
## 3153 US 100 US M  
## 3411 US 100 US M  
## 3464 US 100 US L  
## 3469 US 100 US L  
## 3523 US 0 US M  
## 3676 US 100 US S  
## 3698 US 100 US L  
## 3748 US 50 US L  
## 3751 US 100 US L

# Replaced outliers with median of salary\_in\_usd directly in the same column  
df$salary\_in\_usd[outliers] <- median(df$salary\_in\_usd, na.rm = TRUE) #repeated all the outliers steps twice to get rid of all outliers

#Data Visualization:  
# Check the structure of 'experience\_level'   
str(df$experience\_level)

## chr [1:3755] "SE" "MI" "MI" "SE" "SE" "SE" "SE" "SE" "SE" "SE" "SE" "SE" ...

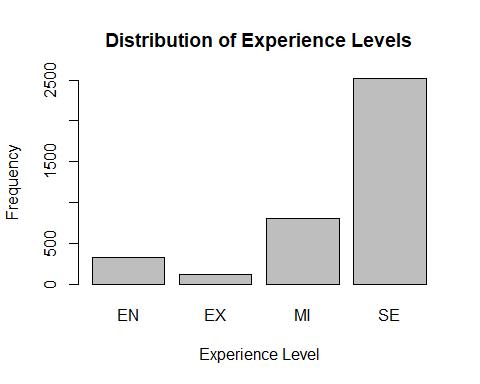
# Check summary statistics of 'experience\_level'  
summary(df$experience\_level)

## Length Class Mode   
## 3755 character character

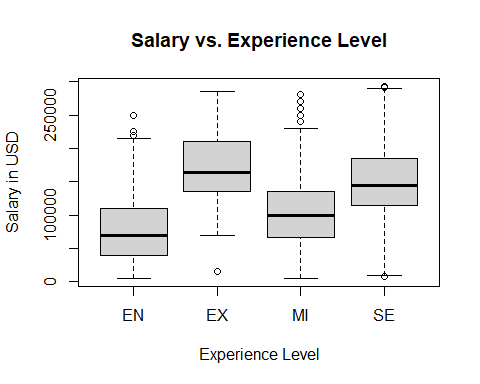
# Convert 'experience\_level' to a factor  
df$experience\_level <- as.factor(df$experience\_level)

# Convert 'experience\_level' to an ordered factor  
df$experience\_level <- factor(df$experience\_level, levels = c("EN", "EX", "MI", "SE"), ordered = TRUE)

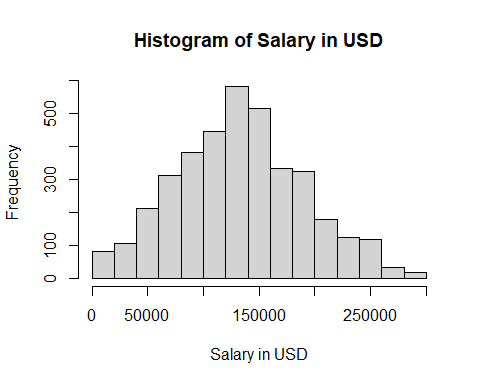
# Bar plot for 'experience\_level'  
barplot(table(df$experience\_level), main = "Distribution of Experience Levels", xlab = "Experience Level", ylab = "Frequency")



# Plot the scatter plot  
plot(df$experience\_level, df$salary\_in\_usd,   
 main = "Salary vs. Experience Level",   
 xlab = "Experience Level",  
 ylab = "Salary in USD")



# Histogram for 'salary\_in\_usd'  
hist(df$salary\_in\_usd, main = "Histogram of Salary in USD", xlab = "Salary in USD")



#Descriptive Statistics:  
# Descriptive statistics for salary  
mean\_salary <- mean(df$salary\_in\_usd)  
median\_salary <- median(df$salary\_in\_usd)  
sd\_salary <- sd(df$salary\_in\_usd)

# Print results  
cat("Mean Salary:", mean\_salary, "\n")

## Mean Salary: 134275.4

cat("Median Salary:", median\_salary, "\n")

## Median Salary: 135000

cat("Standard Deviation:", sd\_salary, "\n")

## Standard Deviation: 57503.7

#Hypothesis Testing:  
# ANOVA for 'salary\_in\_usd' and 'employment\_type'  
anova\_model <- aov(salary\_in\_usd ~ employment\_type, data = df)  
summary(anova\_model)

## Df Sum Sq Mean Sq F value Pr(>F)   
## employment\_type 3 2.468e+11 8.228e+10 25.37 3.05e-16 \*\*\*  
## Residuals 3751 1.217e+13 3.244e+09   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

The p-value is highly significant, indicating that at least one of the employment types has a different mean salary. However, ANOVA does not specify which specific groups are different from each other.

If the overall ANOVA is significant, you can perform post-hoc tests or pairwise comparisons to identify which specific groups differ in mean 'salary\_in\_usd'.

The output from the pairwise t-test with Bonferroni correction is presenting p-values for comparisons between different levels of 'employment\_type.' The p-values are adjusted to account for multiple comparisons.

Here's how to interpret the results:

* CT vs. FL: The p-value for the comparison between 'CT' (Contract) and 'FL' (Freelance) is 1.000, indicating that there is no significant difference in the mean 'salary\_in\_usd' between these two employment types.
* CT vs. FT: The p-value for the comparison between 'CT' and 'FT' (Full-Time) is 0.035. This p-value is less than the significance level (commonly 0.05), suggesting that there is a significant difference in the mean 'salary\_in\_usd' between these two employment types.
* CT vs. PT: The p-value for the comparison between 'CT' and 'PT' (Part-Time) is 0.252, indicating that there is no significant difference in the mean 'salary\_in\_usd' between these two employment types.
* FL vs. FT: The p-value for the comparison between 'FL' and 'FT' is very small (2.2e-05), indicating a highly significant difference in the mean 'salary\_in\_usd' between Freelance and Full-Time employment types.
* FL vs. PT: The p-value for the comparison between 'FL' and 'PT' is 1.000, suggesting no significant difference in the mean 'salary\_in\_usd' between Freelance and Part-Time employment types.
* FT vs. PT: The p-value for the comparison between 'FT' and 'PT' is 2.9e-11 (very close to zero), indicating a highly significant difference in the mean 'salary\_in\_usd' between Full-Time and Part-Time employment types.

In summary, based on the Bonferroni-adjusted p-values:

* There is a significant difference in the mean 'salary\_in\_usd' between Contract and Full-Time employment types.
* There is a highly significant difference in the mean 'salary\_in\_usd' between Freelance and Full-Time, as well as between Full-Time and Part-Time employment types.
* These results provide insights into the differences in salary based on different employment types.

# Pairwise t-test for 'salary\_in\_usd' and 'employment\_type'  
pairwise.t.test(df$salary\_in\_usd, df$employment\_type, p.adjust.method = "bonferroni")

##   
## Pairwise comparisons using t tests with pooled SD   
##   
## data: df$salary\_in\_usd and df$employment\_type   
##   
## CT FL FT   
## FL 1.000 - -   
## FT 0.035 2.4e-05 -   
## PT 0.262 1.000 3.6e-11  
##   
## P value adjustment method: Bonferroni

#Predictive Modeling:  
#To calculate the Gain Ratio for each feature  
library(rpart)  
library(caret)

library(ggplot2)

# Create a decision tree model  
tree\_model <- rpart(salary\_in\_usd ~ experience\_level + employment\_type + job\_title + remote\_ratio + company\_size,data = df)

# Calculate Gain Ratio using varImp  
variable\_importance <- varImp(tree\_model, scale = FALSE)

# Print variable importance  
print(variable\_importance)

## Overall  
## company\_size 0.28216235  
## employment\_type 0.08280606  
## experience\_level 0.34661382  
## job\_title 0.49626415  
## remote\_ratio 0.18826119

Interpretation of the Output:

* company\_size: Has a moderate importance (0.282) in the model. It contributes to the decision-making process but is not the most influential.
* employment\_type: Has a lower importance (0.082) compared to other variables. It contributes less to the decision-making process.
* experience\_level: Has a moderate importance (0.346). It is a relatively influential variable in the model.
* job\_title: Has the highest importance (0.496) among the variables. It is a key factor in determining salary predictions.
* remote\_ratio: Has a moderate importance (0.188). It contributes to the decision-making process but is not the most influential.

Conclusion:

The model places the most importance on the job\_title variable, indicating that it plays a crucial role in predicting salary\_in\_usd.

Other variables, such as experience\_level and company\_size, also contribute to the model's performance but to a lesser extent.

It's essential to consider these variable importance values in the context of your specific goals and domain knowledge. They provide insights into which features are more influential in the decision-making process of the model.

# Build a linear regression predictive model  
model <- lm(salary\_in\_usd ~ experience\_level + employment\_type + job\_title + company\_size, data = df)

#Split out new training and test datasets:  
train\_indices <- createDataPartition(df$salary\_in\_usd, p = 0.8, list = FALSE)  
train\_data <- df[train\_indices, ]   
test\_data <- df[-train\_indices, ] #the model has been trained on the full dataset with all factor levels for job\_title

# Make predictions on the test set  
predictions <- predict(model, newdata = test\_data)

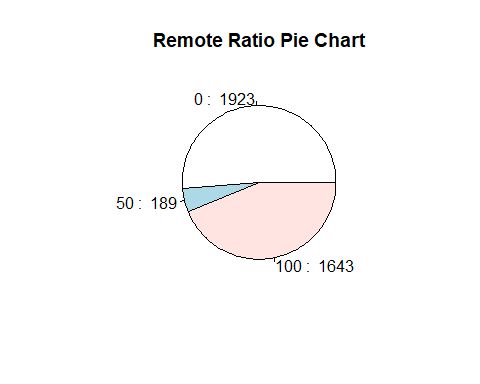
# Add predictions to the test\_data dataframe  
test\_data$predicted\_salary <- predictions

# Evaluate the model performance  
mse <- mean((test\_data$salary\_in\_usd - predictions)^2)   
rmse <- sqrt(mse)   
cat("Root Mean Squared Error (RMSE):", rmse, "\n")

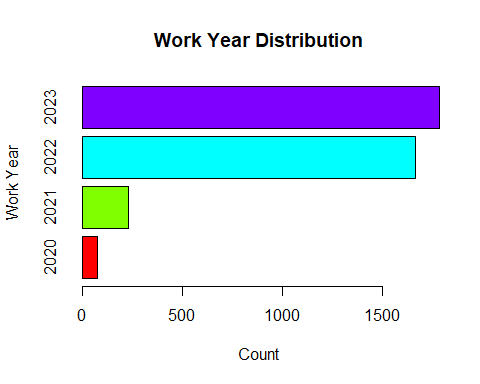
## Root Mean Squared Error (RMSE): 47626.27

# Save the updated dataframe with predictions to a CSV file  
write.csv(test\_data, "predictions.csv", row.names = FALSE)

#Visualization:  
# Count the frequencies of remote\_ratio  
remote\_counts <- table(df$remote\_ratio)  
  
# Create a pie chart  
pie(remote\_counts, labels = paste(names(remote\_counts), ": ", remote\_counts), main = "Remote Ratio Pie Chart")



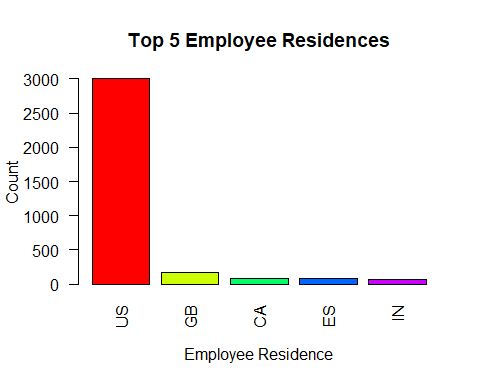
# Count the frequencies of work\_year  
work\_year\_counts <- table(df$work\_year)  
  
# Create a bar chart  
barplot(work\_year\_counts, col = rainbow(length(work\_year\_counts)), main = "Work Year Distribution", xlab = "Count", ylab = "Work Year", horiz = TRUE)



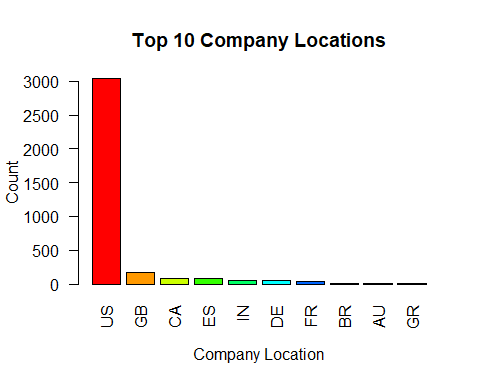
# Count the frequencies of employee\_residence  
residence\_counts <- table(df$employee\_residence)

# Sort the counts in descending order  
sorted\_counts <- sort(residence\_counts, decreasing = TRUE)

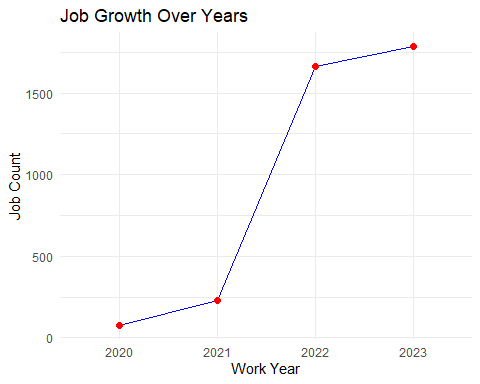
# Take the top 5 counts and corresponding residences  
top\_residences <- names(sorted\_counts)[1:5]  
top\_counts <- sorted\_counts[1:5]  
  
# Created a bar chart for the top 5 residences  
barplot(top\_counts, col = rainbow(length(top\_counts)), main = "Top 5 Employee Residences", names.arg = top\_residences, xlab = "Employee Residence", ylab = "Count", las = 2)



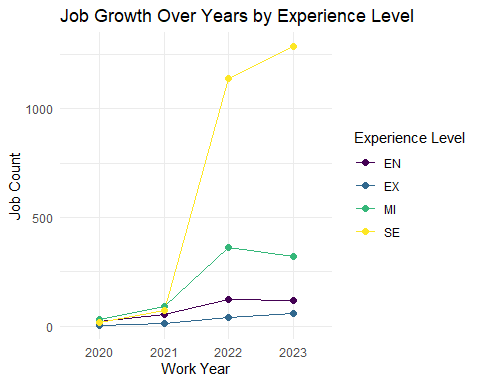
# Count the frequencies of company\_location  
location\_counts <- table(df$company\_location)  
  
# Sort the counts in descending order  
sorted\_counts <- sort(location\_counts, decreasing = TRUE)  
  
# Take the top 10 counts and corresponding locations  
top\_locations <- names(sorted\_counts)[1:10]  
top\_counts <- sorted\_counts[1:10]  
  
# Create a bar chart for the top 10 company locations  
barplot(top\_counts, col = rainbow(length(top\_counts)), main = "Top 10 Company Locations", names.arg = top\_locations, xlab = "Company Location", ylab = "Count", las = 2)



# Convert 'work\_year' to a factor for better ordering  
df$work\_year <- as.factor(df$work\_year)  
  
# Create a line chart for job growth by year  
ggplot(df, aes(x = work\_year, group = 1)) +  
 geom\_line(stat = "count", aes(y = ..count..), color = "blue") +  
 geom\_point(stat = "count", aes(y = ..count..), color = "red", size = 2) +  
 labs(title = "Job Growth Over Years",  
 x = "Work Year",  
 y = "Job Count") +  
 theme\_minimal()



# Create a line chart for job growth by experience level  
ggplot(df, aes(x = work\_year, group = experience\_level, color = experience\_level)) +  
 geom\_line(stat = "count", aes(y = ..count..)) +  
 geom\_point(stat = "count", aes(y = ..count..), size = 2) +  
 labs(title = "Job Growth Over Years by Experience Level",  
 x = "Work Year",  
 y = "Job Count",  
 color = "Experience Level") +  
 theme\_minimal()



library(dplyr)

# Calculate the percentage of remote jobs per year  
remote\_percentage <- df %>%  
 group\_by(work\_year, remote\_ratio) %>%  
 summarise(job\_count = n()) %>%  
 mutate(percentage = job\_count / sum(job\_count) \* 100)

# Create a bar chart for the percentage of remote vs in-person jobs per year  
ggplot(remote\_percentage, aes(x = work\_year, y = percentage, fill = remote\_ratio)) +  
 geom\_bar(stat = "identity", position = "stack") +  
 labs(title = "Percentage of Remote vs In-Person Jobs Per Year",  
 x = "Work Year",  
 y = "Percentage",  
 fill = "Remote Ratio") +  
 scale\_y\_continuous(labels = scales::percent\_format(scale = 1)) +  
 theme\_minimal()

