Furman University

Statistical Methods I in R

MTH 245

Evaluation of Contemporary Data Science Salaries

Authors:
William Foster, Jonathan Wade,
Flynn Nisbet

Supervisor:
Dr. Jordan Bounds

April 29, 2024

1 Abstract

This study investigates the determinants of salary within the data science field, utilizing a robust dataset from job postings across various geographic locations and company sizes. The analysis explores how professional experience, job title, educational background, company size, and geographic location influence salary outcomes. Advanced statistical techniques, including regression models and machine learning methods, are applied to analyze these factors comprehensively. Results indicate significant variances in compensation, influenced by factors such as experience level and job title, with higher-level positions and roles in larger companies generally commanding higher salaries. Additionally, the shift towards remote work is examined for its impact on compensation structures. This research not only provides empirical insights into the current salary landscape in data science but also serves as a valuable resource for stakeholders in the tech industry, aiding in strategic planning and decision-making regarding compensation.

2 Introduction

The rapid expansion of the data science sector has prompted a surge in demand for skilled professionals, making the understanding of salary determinants increasingly crucial for both job market entrants and established entities. This paper addresses the pertinent question of what factors most significantly influence data science salaries, aiming to shed light on the complexities of tech industry compensation and provide a foundation for both theoretical and practical applications.

Salaries in data science are presumed to be influenced by a variety of factors, including but not limited to professional experience, educational background, job title, company size, and geographic location. Additionally, the shift towards remote work and its implications on compensation structures presents a new dimension of analysis, reflecting modern work arrangements. By exploring these aspects, this study aims to provide a comprehensive overview of the components that collectively dictate salary structures in the field of data science.

This investigation utilizes a dataset containing detailed job postings, which includes information on salary figures across different countries and company sizes, varying experience levels, and diverse job titles. The dataset provides a unique opportunity to analyze real-world data and apply statistical methodologies to derive meaningful insights. Our approach combines traditional statistical techniques with advanced analytics, including regression models and machine learning, to thoroughly examine how each factor contributes to salary variations.

The significance of this study lies in its potential to impact various stakeholders in the tech industry. For employers and HR professionals, understanding these salary determinants can assist in developing strategic compensation plans that attract and retain top talent. For employees and job seekers, insights from this research can empower them with knowledge to better negotiate their compensation packages, aligning their career decisions with their financial goals and market realities.

Furthermore, this study aims to provide empirical evidence on salary determinants in a rapidly evolving field, offering a basis for future research and discussion. As the data science landscape continues to evolve with technological advancements and economic shifts, the insights derived from this study will help stakeholders stay abreast of changes and make informed decisions in a competitive job market.

3 Exploratory Data Analysis

```
knitr::opts_chunk$set(echo = TRUE)
library(tidyverse)
library(caret)
library(car)
library(knitr)
library(kableExtra)
```

```
library(patchwork)
library(scales)
library(gtsummary)
library(tidymodels)
library(MASS)
library(xtable)
library(rcompanion)
tidymodels_prefer()
```

First, let's library necessary packages and load the data into R.

```
df <- read_csv('/Users/wfoster/Desktop/ds_salaries.csv', show_col_types = FALSE)
glimpse(df)
```

```
## Rows: 607
## Columns: 12
## $ ...1               <dbl> 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 1~
## $ work_year          <dbl> 2020, 2020, 2020, 2020, 2020, 2020, 2020, 2020, 202~
## $ experience_level   <chr> "MI", "SE", "SE", "MI", "SE", "EN", "SE", "MI", "MI~
## $ employment_type    <chr> "FT", "FT", "FT", "FT", "FT", "FT", "FT", "FT", "FT~
## $ job_title          <chr> "Data Scientist", "Machine Learning Scientist", "Bi~
## $ salary             <dbl> 70000, 260000, 85000, 20000, 150000, 72000, 190000,~
## $ salary_currency    <chr> "EUR", "USD", "GBP", "USD", "USD", "USD", "USD", "H~
## $ salary_in_usd      <dbl> 79833, 260000, 109024, 20000, 150000, 72000, 190000~
## $ employee_residence <chr> "DE", "JP", "GB", "HN", "US", "US", "US", "HU", "US~
## $ remote_ratio       <dbl> 0, 0, 50, 0, 50, 100, 100, 50, 100, 50, 0, 0, 0, 10~
## $ company_location   <chr> "DE", "JP", "GB", "HN", "US", "US", "US", "HU", "US~
## $ company_size       <chr> "L", "S", "M", "S", "L", "L", "S", "L", "L", "S", "~
```

```
df <- df %>%
  mutate(
    work_year = factor(work_year),
    experience_level = factor(experience_level, levels = c("EN", "MI", "SE", "EX")),
    employment_type = factor(employment_type),
    job_title = factor(job_title),
    salary_currency = factor(salary_currency),
    employee_residence = factor(employee_residence),
    company_location = factor(company_location),
    company_size = factor(company_size, levels = c("S", "M", "L")),
    remote_ratio = as.integer(remote_ratio)  # Convert to integer if you're keeping it numeric
  )%>%
  select(-...1)
```

The dataset from Kaggle (https://www.kaggle.com/datasets/ruchi798/data-science-job-salaries) comprises data on 607 individuals, encompassing a range of features such as job title, salary, experience level, employment type, company location and size, and the proportion of remote work.

In the preprocessing phase, the data is refined by transforming pertinent variables into categorical factors and eliminating a redundant index column. Given that the dataset includes two different salary measurements, a decision will be made to retain only one for the purpose of model construction. This adjustment will be addressed in subsequent steps of the data analysis process.

3.1 Summary Table

```r
table_summary <- df %>%
  select(-starts_with("Unnamed")) %>%
  tbl_summary(
    by = "experience_level",
    type = list(
      where(is.numeric) ~ "continuous2",
      where(is.character) ~ "categorical"
    ),
    statistic = list(
      all_continuous() ~ "{min}, {max} [{mean}; {sd}]",
      all_categorical() ~ "{n} ({p}%)"
    )
  ) %>%
  modify_header(label = "Variable") %>%
  add_n() %>%
  bold_labels() %>%
  modify_caption("Summary of Variables.")

table_summary
```

Table 1: Summary of Variables.

Variable	N	**EN**, N = 88	**MI**, N = 213	**SE**, N = 280	**EX**, N = 26
work_year	607				
2020		20 (23%)	32 (15%)	18 (6.4%)	2 (7.7%)
2021		47 (53%)	90 (42%)	69 (25%)	11 (42%)
2022		21 (24%)	91 (43%)	193 (69%)	13 (50%)
employment_type	607				
CT		2 (2.3%)	1 (0.5%)	1 (0.4%)	1 (3.8%)
FL		0 (0%)	3 (1.4%)	1 (0.4%)	0 (0%)
FT		79 (90%)	206 (97%)	278 (99%)	25 (96%)
PT		7 (8.0%)	3 (1.4%)	0 (0%)	0 (0%)
job_title	607				
3D Computer Vision Researcher		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
AI Scientist		4 (4.5%)	2 (0.9%)	1 (0.4%)	0 (0%)
Analytics Engineer		0 (0%)	0 (0%)	2 (0.7%)	2 (7.7%)
Applied Data Scientist		1 (1.1%)	2 (0.9%)	2 (0.7%)	0 (0%)
Applied Machine Learning Scientist		1 (1.1%)	3 (1.4%)	0 (0%)	0 (0%)
BI Data Analyst		2 (2.3%)	3 (1.4%)	0 (0%)	1 (3.8%)
Big Data Architect		0 (0%)	0 (0%)	1 (0.4%)	0 (0%)
Big Data Engineer		3 (3.4%)	3 (1.4%)	2 (0.7%)	0 (0%)
Business Data Analyst		2 (2.3%)	3 (1.4%)	0 (0%)	0 (0%)
Cloud Data Engineer		0 (0%)	1 (0.5%)	1 (0.4%)	0 (0%)
Computer Vision Engineer		3 (3.4%)	0 (0%)	3 (1.1%)	0 (0%)
Computer Vision Software Engineer		2 (2.3%)	1 (0.5%)	0 (0%)	0 (0%)

Variable	N	**EN**, N = 88	**MI**, N = 213	**SE**, N = 280	**EX**, N = 26
Data Analyst		12 (14%)	29 (14%)	54 (19%)	2 (7.7%)
Data Analytics Engineer		1 (1.1%)	1 (0.5%)	2 (0.7%)	0 (0%)
Data Analytics Lead		0 (0%)	0 (0%)	1 (0.4%)	0 (0%)
Data Analytics Manager		0 (0%)	0 (0%)	7 (2.5%)	0 (0%)
Data Architect		0 (0%)	3 (1.4%)	8 (2.9%)	0 (0%)
Data Engineer		12 (14%)	53 (25%)	63 (23%)	4 (15%)
Data Engineering Manager		0 (0%)	1 (0.5%)	3 (1.1%)	1 (3.8%)
Data Science Consultant		5 (5.7%)	1 (0.5%)	0 (0%)	1 (3.8%)
Data Science Engineer		0 (0%)	1 (0.5%)	2 (0.7%)	0 (0%)
Data Science Manager		0 (0%)	2 (0.9%)	10 (3.6%)	0 (0%)
Data Scientist		22 (25%)	60 (28%)	61 (22%)	0 (0%)
Data Specialist		0 (0%)	0 (0%)	1 (0.4%)	0 (0%)
Director of Data Engineering		0 (0%)	0 (0%)	2 (0.7%)	0 (0%)
Director of Data Science		0 (0%)	0 (0%)	1 (0.4%)	6 (23%)
ETL Developer		0 (0%)	2 (0.9%)	0 (0%)	0 (0%)
Finance Data Analyst		0 (0%)	0 (0%)	1 (0.4%)	0 (0%)
Financial Data Analyst		1 (1.1%)	1 (0.5%)	0 (0%)	0 (0%)
Head of Data		0 (0%)	1 (0.5%)	2 (0.7%)	2 (7.7%)
Head of Data Science		0 (0%)	1 (0.5%)	0 (0%)	3 (12%)
Head of Machine Learning		0 (0%)	0 (0%)	0 (0%)	1 (3.8%)
Lead Data Analyst		0 (0%)	2 (0.9%)	1 (0.4%)	0 (0%)
Lead Data Engineer		0 (0%)	1 (0.5%)	4 (1.4%)	1 (3.8%)
Lead Data Scientist		0 (0%)	1 (0.5%)	2 (0.7%)	0 (0%)
Lead Machine Learning Engineer		0 (0%)	0 (0%)	1 (0.4%)	0 (0%)
Machine Learning Developer		1 (1.1%)	1 (0.5%)	1 (0.4%)	0 (0%)
Machine Learning Engineer		9 (10%)	12 (5.6%)	20 (7.1%)	0 (0%)
Machine Learning Infrastructure Engineer		0 (0%)	2 (0.9%)	1 (0.4%)	0 (0%)
Machine Learning Manager		0 (0%)	0 (0%)	1 (0.4%)	0 (0%)
Machine Learning Scientist		1 (1.1%)	4 (1.9%)	3 (1.1%)	0 (0%)

Variable	N	EN, N = 88	MI, N = 213	SE, N = 280	EX, N = 26
Marketing Data Analyst		0 (0%)	0 (0%)	1 (0.4%)	0 (0%)
ML Engineer		2 (2.3%)	3 (1.4%)	1 (0.4%)	0 (0%)
NLP Engineer		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
Principal Data Analyst		0 (0%)	1 (0.5%)	1 (0.4%)	0 (0%)
Principal Data Engineer		0 (0%)	0 (0%)	2 (0.7%)	1 (3.8%)
Principal Data Scientist		0 (0%)	1 (0.5%)	5 (1.8%)	1 (3.8%)
Product Data Analyst		0 (0%)	2 (0.9%)	0 (0%)	0 (0%)
Research Scientist		4 (4.5%)	7 (3.3%)	5 (1.8%)	0 (0%)
Staff Data Scientist		0 (0%)	0 (0%)	1 (0.4%)	0 (0%)
salary	607				
Range [Mean; SD]		4,000, 4,450,000 [264,622; 650,986]	4,000, 30,400,000 [480,618; 2,448,889]	24,000, 7,000,000 [213,949; 584,283]	59,000, 6,000,000 [427,072; 1,142,630]
salary_currency	607				
AUD		1 (1.1%)	1 (0.5%)	0 (0%)	0 (0%)
BRL		0 (0%)	1 (0.5%)	1 (0.4%)	0 (0%)
CAD		1 (1.1%)	7 (3.3%)	8 (2.9%)	2 (7.7%)
CHF		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
CLP		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
CNY		0 (0%)	2 (0.9%)	0 (0%)	0 (0%)
DKK		2 (2.3%)	0 (0%)	0 (0%)	0 (0%)
EUR		20 (23%)	48 (23%)	22 (7.9%)	5 (19%)
GBP		5 (5.7%)	30 (14%)	9 (3.2%)	0 (0%)
HUF		0 (0%)	2 (0.9%)	0 (0%)	0 (0%)
INR		10 (11%)	11 (5.2%)	5 (1.8%)	1 (3.8%)
JPY		1 (1.1%)	2 (0.9%)	0 (0%)	0 (0%)
MXN		0 (0%)	1 (0.5%)	1 (0.4%)	0 (0%)
PLN		0 (0%)	3 (1.4%)	0 (0%)	0 (0%)
SGD		0 (0%)	2 (0.9%)	0 (0%)	0 (0%)
TRY		0 (0%)	2 (0.9%)	1 (0.4%)	0 (0%)
USD		48 (55%)	99 (46%)	233 (83%)	18 (69%)
salary_in_usd	607				
Range [Mean; SD]		4,000, 250,000 [61,643; 44,396]	2,859, 450,000 [87,996; 63,901]	18,907, 412,000 [138,617; 57,692]	69,741, 600,000 [199,392; 117,071]
employee_residence	607				
AE		0 (0%)	1 (0.5%)	2 (0.7%)	0 (0%)
AR		0 (0%)	0 (0%)	1 (0.4%)	0 (0%)
AT		0 (0%)	2 (0.9%)	1 (0.4%)	0 (0%)
AU		2 (2.3%)	1 (0.5%)	0 (0%)	0 (0%)
BE		0 (0%)	1 (0.5%)	1 (0.4%)	0 (0%)
BG		0 (0%)	0 (0%)	1 (0.4%)	0 (0%)
BO		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
BR		1 (1.1%)	1 (0.5%)	4 (1.4%)	0 (0%)
CA		3 (3.4%)	10 (4.7%)	14 (5.0%)	2 (7.7%)
CH		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
CL		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)

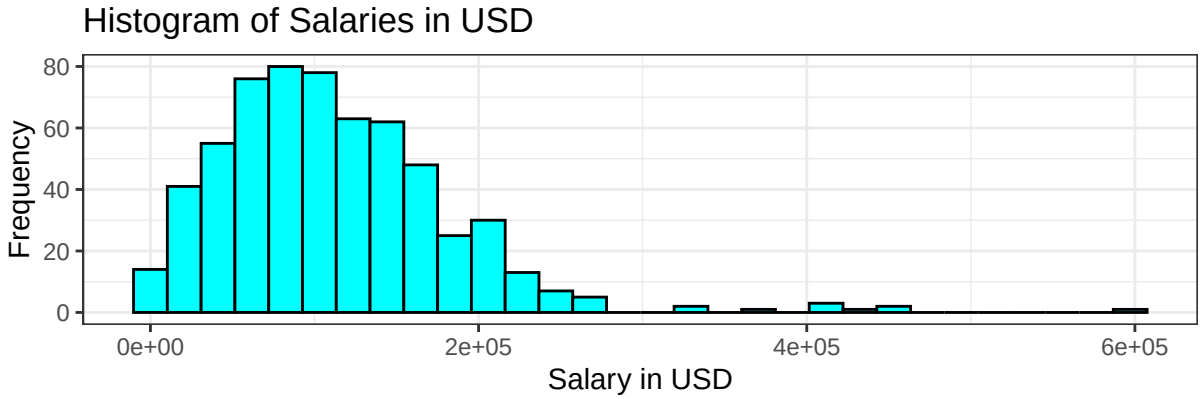
Variable	N	**EN**, N = 88	**MI**, N = 213	**SE**, N = 280	**EX**, N = 26
CN		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
CO		1 (1.1%)	0 (0%)	0 (0%)	0 (0%)
CZ		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
DE		8 (9.1%)	8 (3.8%)	7 (2.5%)	2 (7.7%)
DK		2 (2.3%)	0 (0%)	0 (0%)	0 (0%)
DZ		1 (1.1%)	0 (0%)	0 (0%)	0 (0%)
EE		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
ES		1 (1.1%)	12 (5.6%)	1 (0.4%)	1 (3.8%)
FR		5 (5.7%)	8 (3.8%)	4 (1.4%)	1 (3.8%)
GB		5 (5.7%)	30 (14%)	9 (3.2%)	0 (0%)
GR		0 (0%)	10 (4.7%)	3 (1.1%)	0 (0%)
HK		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
HN		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
HR		0 (0%)	0 (0%)	1 (0.4%)	0 (0%)
HU		0 (0%)	2 (0.9%)	0 (0%)	0 (0%)
IE		0 (0%)	0 (0%)	1 (0.4%)	0 (0%)
IN		11 (13%)	12 (5.6%)	5 (1.8%)	2 (7.7%)
IQ		1 (1.1%)	0 (0%)	0 (0%)	0 (0%)
IR		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
IT		1 (1.1%)	2 (0.9%)	0 (0%)	1 (3.8%)
JE		1 (1.1%)	0 (0%)	0 (0%)	0 (0%)
JP		2 (2.3%)	3 (1.4%)	2 (0.7%)	0 (0%)
KE		1 (1.1%)	0 (0%)	0 (0%)	0 (0%)
LU		1 (1.1%)	0 (0%)	0 (0%)	0 (0%)
MD		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
MT		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
MX		0 (0%)	1 (0.5%)	1 (0.4%)	0 (0%)
MY		0 (0%)	0 (0%)	1 (0.4%)	0 (0%)
NG		1 (1.1%)	1 (0.5%)	0 (0%)	0 (0%)
NL		2 (2.3%)	2 (0.9%)	1 (0.4%)	0 (0%)
NZ		0 (0%)	0 (0%)	1 (0.4%)	0 (0%)
PH		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
PK		4 (4.5%)	2 (0.9%)	0 (0%)	0 (0%)
PL		0 (0%)	3 (1.4%)	1 (0.4%)	0 (0%)
PR		0 (0%)	0 (0%)	1 (0.4%)	0 (0%)
PT		2 (2.3%)	3 (1.4%)	1 (0.4%)	0 (0%)
RO		0 (0%)	1 (0.5%)	1 (0.4%)	0 (0%)
RS		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
RU		0 (0%)	1 (0.5%)	1 (0.4%)	2 (7.7%)
SG		0 (0%)	2 (0.9%)	0 (0%)	0 (0%)
SI		0 (0%)	1 (0.5%)	1 (0.4%)	0 (0%)
TN		1 (1.1%)	0 (0%)	0 (0%)	0 (0%)
TR		0 (0%)	2 (0.9%)	1 (0.4%)	0 (0%)
UA		1 (1.1%)	0 (0%)	0 (0%)	0 (0%)
US		29 (33%)	77 (36%)	211 (75%)	15 (58%)
VN		1 (1.1%)	1 (0.5%)	1 (0.4%)	0 (0%)
remote_ratio	607				
Range [Mean; SD]		0, 100 [70; 38]	0, 100 [64; 43]	0, 100 [76; 40]	0, 100 [79; 35]
company_location	607				
AE		0 (0%)	1 (0.5%)	2 (0.7%)	0 (0%)
AS		1 (1.1%)	0 (0%)	0 (0%)	0 (0%)
AT		0 (0%)	3 (1.4%)	1 (0.4%)	0 (0%)

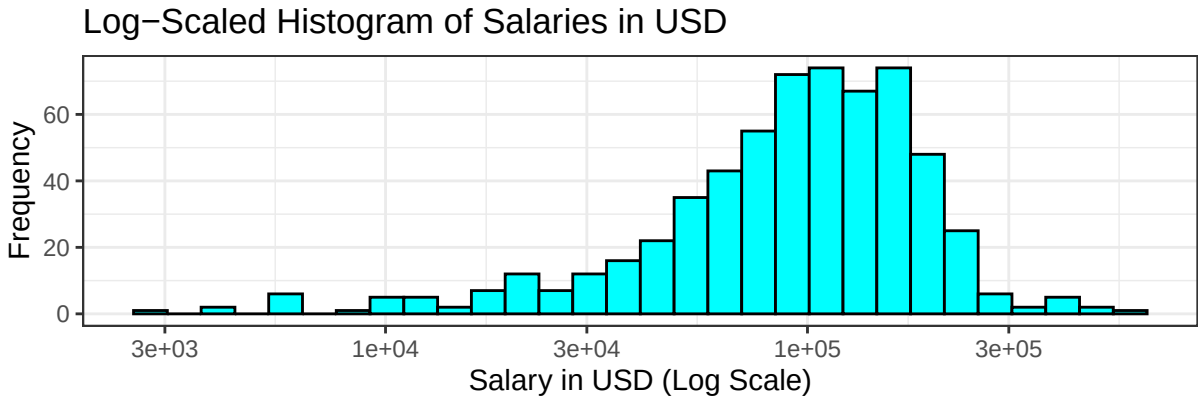
Variable	**N**	**EN**, N = 88	**MI**, N = 213	**SE**, N = 280	**EX**, N = 26
AU		2 (2.3%)	1 (0.5%)	0 (0%)	0 (0%)
BE		0 (0%)	1 (0.5%)	1 (0.4%)	0 (0%)
BR		0 (0%)	1 (0.5%)	2 (0.7%)	0 (0%)
CA		3 (3.4%)	10 (4.7%)	15 (5.4%)	2 (7.7%)
CH		1 (1.1%)	1 (0.5%)	0 (0%)	0 (0%)
CL		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
CN		1 (1.1%)	1 (0.5%)	0 (0%)	0 (0%)
CO		1 (1.1%)	0 (0%)	0 (0%)	0 (0%)
CZ		1 (1.1%)	1 (0.5%)	0 (0%)	0 (0%)
DE		11 (13%)	8 (3.8%)	7 (2.5%)	2 (7.7%)
DK		2 (2.3%)	0 (0%)	1 (0.4%)	0 (0%)
DZ		1 (1.1%)	0 (0%)	0 (0%)	0 (0%)
EE		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
ES		1 (1.1%)	10 (4.7%)	1 (0.4%)	2 (7.7%)
FR		5 (5.7%)	6 (2.8%)	4 (1.4%)	0 (0%)
GB		5 (5.7%)	30 (14%)	12 (4.3%)	0 (0%)
GR		0 (0%)	10 (4.7%)	1 (0.4%)	0 (0%)
HN		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
HR		0 (0%)	0 (0%)	1 (0.4%)	0 (0%)
HU		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
IE		0 (0%)	0 (0%)	1 (0.4%)	0 (0%)
IL		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
IN		9 (10%)	10 (4.7%)	4 (1.4%)	1 (3.8%)
IQ		1 (1.1%)	0 (0%)	0 (0%)	0 (0%)
IR		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
IT		1 (1.1%)	1 (0.5%)	0 (0%)	0 (0%)
JP		1 (1.1%)	3 (1.4%)	2 (0.7%)	0 (0%)
KE		1 (1.1%)	0 (0%)	0 (0%)	0 (0%)
LU		2 (2.3%)	1 (0.5%)	0 (0%)	0 (0%)
MD		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
MT		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
MX		0 (0%)	1 (0.5%)	2 (0.7%)	0 (0%)
MY		1 (1.1%)	0 (0%)	0 (0%)	0 (0%)
NG		1 (1.1%)	1 (0.5%)	0 (0%)	0 (0%)
NL		1 (1.1%)	2 (0.9%)	1 (0.4%)	0 (0%)
NZ		0 (0%)	0 (0%)	1 (0.4%)	0 (0%)
PK		1 (1.1%)	2 (0.9%)	0 (0%)	0 (0%)
PL		0 (0%)	3 (1.4%)	0 (0%)	1 (3.8%)
PT		1 (1.1%)	2 (0.9%)	1 (0.4%)	0 (0%)
RO		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
RU		0 (0%)	0 (0%)	0 (0%)	2 (7.7%)
SG		0 (0%)	1 (0.5%)	0 (0%)	0 (0%)
SI		0 (0%)	1 (0.5%)	1 (0.4%)	0 (0%)
TR		0 (0%)	2 (0.9%)	1 (0.4%)	0 (0%)
UA		1 (1.1%)	0 (0%)	0 (0%)	0 (0%)
US		31 (35%)	90 (42%)	218 (78%)	16 (62%)
VN		1 (1.1%)	0 (0%)	0 (0%)	0 (0%)
company_size	607				
S		29 (33%)	29 (14%)	22 (7.9%)	3 (12%)
M		30 (34%)	98 (46%)	186 (66%)	12 (46%)
L		29 (33%)	86 (40%)	72 (26%)	11 (42%)

The summary table presents a comprehensive overview of the dataset, organized according to experience levels. It illustrates the distribution of various attributes such as job titles, types of employment, salary ranges, and other relevant factors across different professional stages, including entry-level, mid-level, senior-level, and executive-level positions.

Notably, some experience levels appear infrequently within the dataset, raising concerns regarding potential issues such as overfitting. To mitigate these challenges, it may be necessary to amalgamate certain categories. This strategy will ensure more robust data analysis and model building, enhancing the reliability of the results. However, it's better understood graphically.

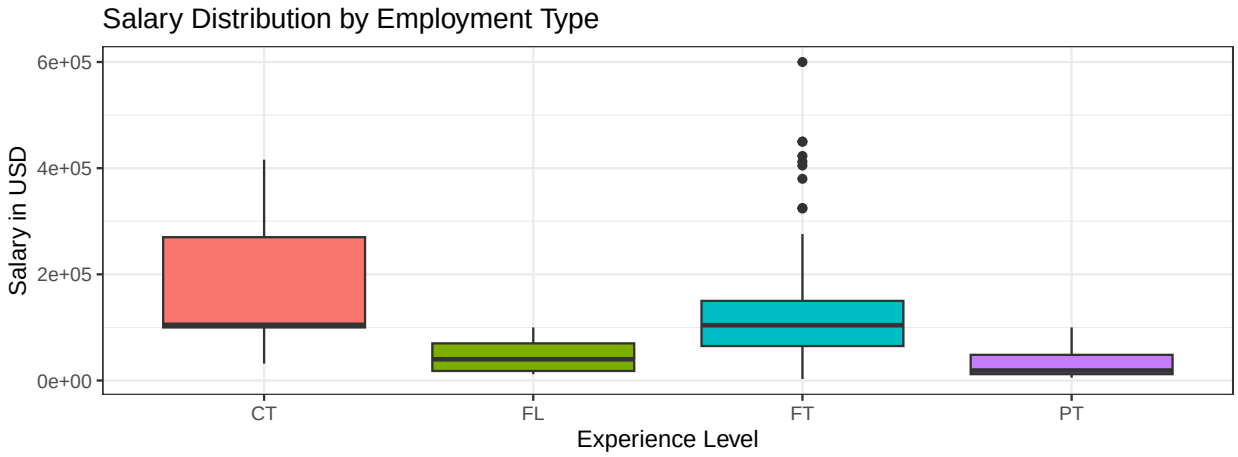
3.2 Data Visualization

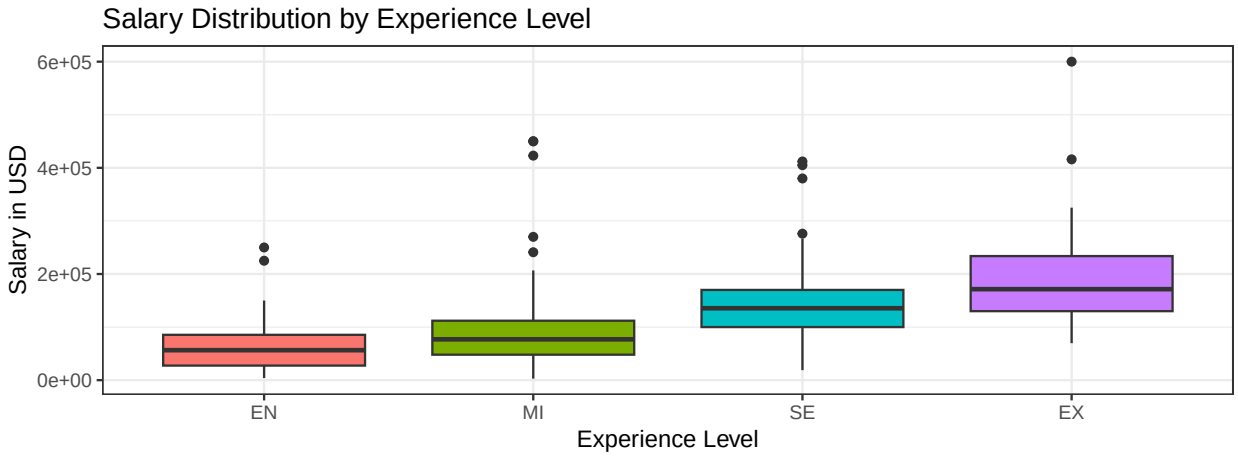


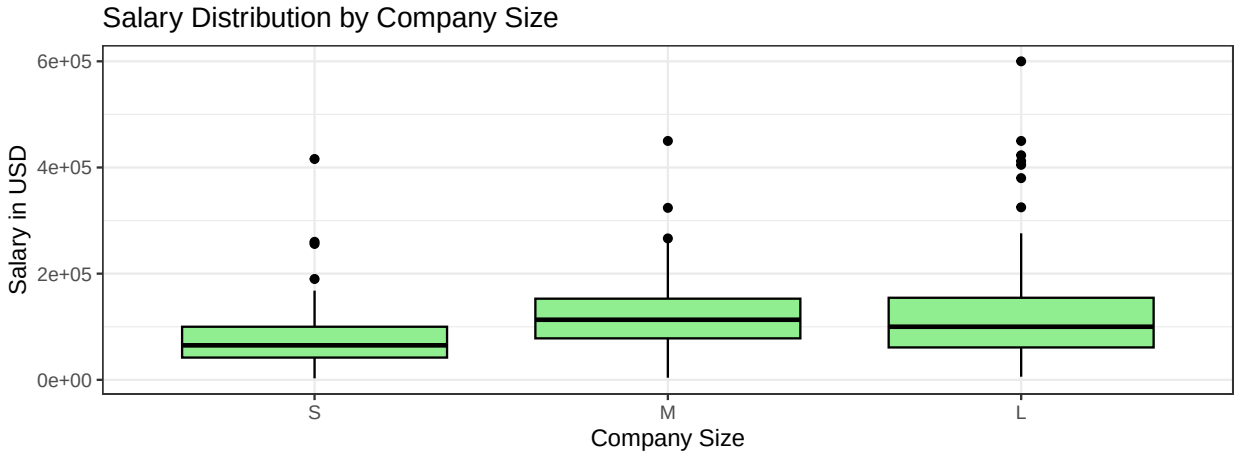


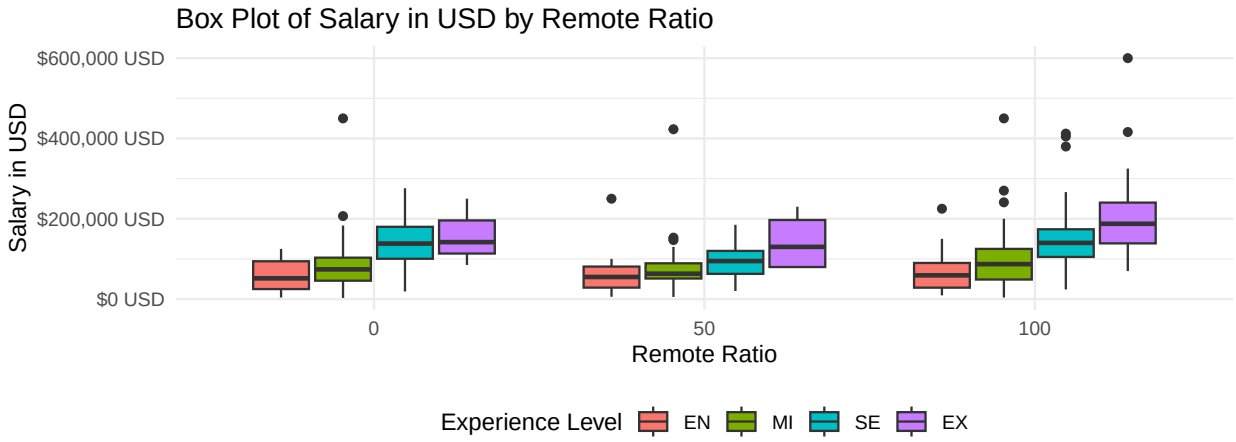
The histograms illustrate that the distribution of salaries within the dataset is right-skewed, characterized by a long tail of higher salaries extending to the right. Applying a logarithmic transformation to the salary data effectively normalizes this distribution, making it more symmetrical and statistically manageable.

This pattern indicates that although the majority of salaries cluster within a specific range, there are notable outliers that receive significantly higher compensation. Such disparities highlight the presence of highly remunerative positions that deviate from the typical salary norms.


Salary Distribution by Employment Type
6e+05
4e+05
2e+05
0e+00
Salary in USD
CT
FL
FT
PT
Experience Level


Salary Distribution by Experience Level
6e+05
4e+05
2e+05
0e+00
Salary in USD
EN
MI
SE
EX
Experience Level


Salary Distribution by Company Size
6e+05
4e+05
2e+05
0e+00
Salary in USD
S
M
L
Company Size

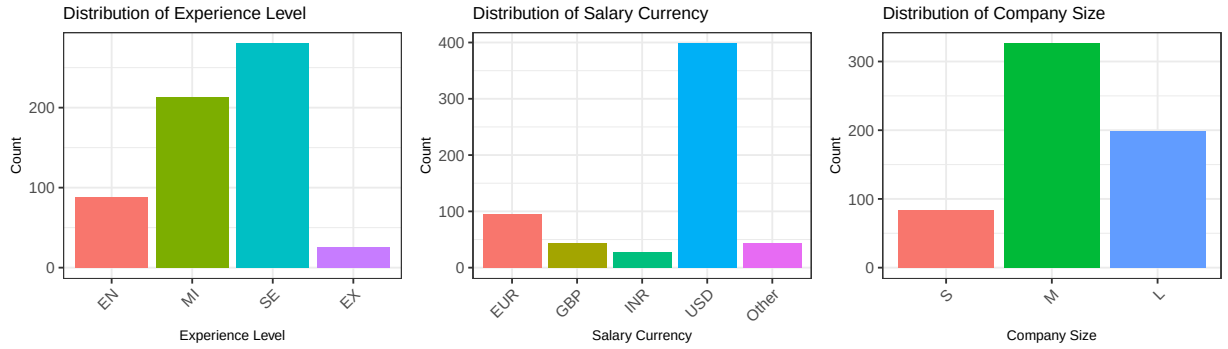


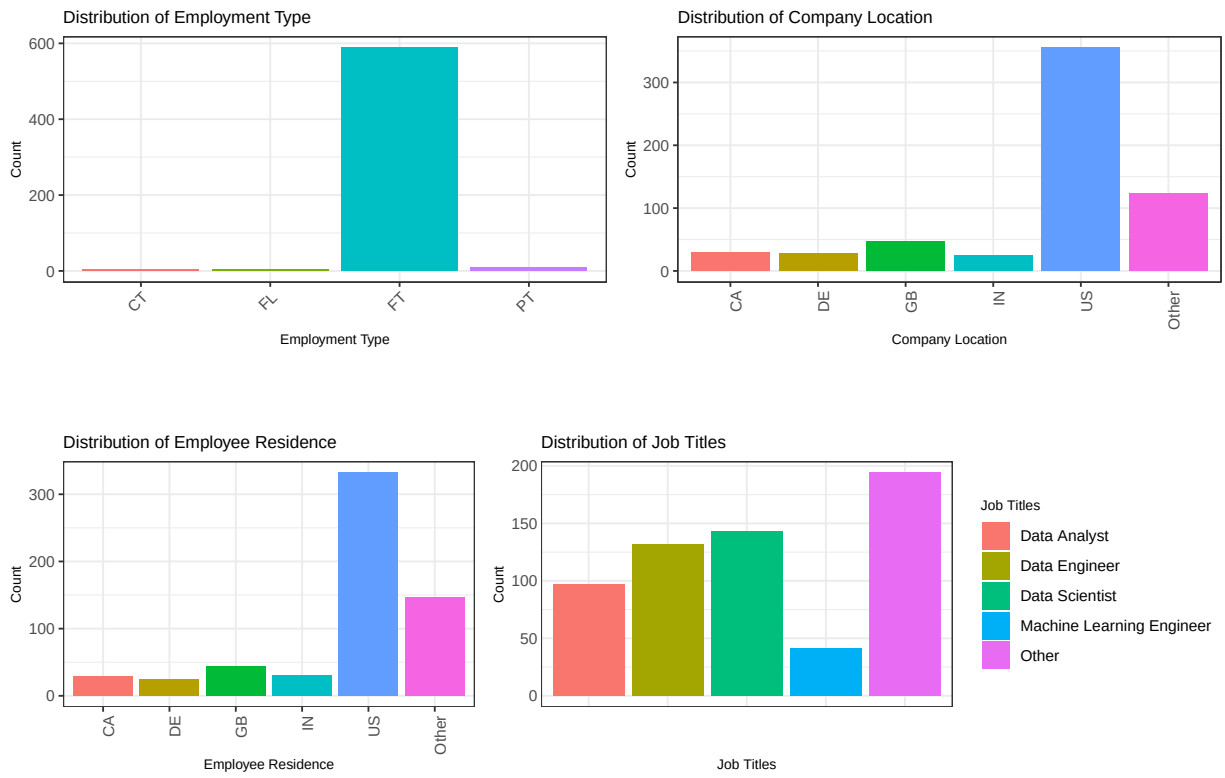
The analysis of the boxplots provides valuable insights into the salary trends across various factors within the dataset. It is evident that salaries escalate with an increase in experience levels. Entry-level positions generally present lower compensation, whereas senior and executive roles command higher salaries, reflecting the value of accumulated professional experience and responsibilities.

Regarding employment type, full-time roles consistently offer higher salaries compared to part-time or contractual positions. This discrepancy likely arises from the greater job security and benefits typically associated with full-time employment.

When examining company size, it becomes apparent that larger organizations tend to provide higher salaries. This trend may be attributed to the larger financial resources and more established compensation frameworks that bigger companies often possess.

Additionally, the analysis reveals that positions allowing for a greater proportion of remote work usually have slightly higher salaries. This could be linked to the benefits of flexibility and the cost efficiencies that remote working conditions afford both employers and employees.





The bar plots provide a detailed visual representation of the distribution of categorical variables within the dataset, yielding several key insights into employment trends. Primarily, the majority of employees are categorized as being at the senior or mid-level, indicating a workforce with a significant amount of experience and expertise. This suggests a maturity within the sector, where more seasoned professionals dominate.

In terms of employment type, full-time positions are the most prevalent, highlighting the stability and continuity that full-time roles offer to employees in the industry. This commonality underscores the sector's preference for sustained engagement and possibly greater job security and benefits that full-time employment typically entails.

Regarding the currency in which salaries are paid, the US Dollar (USD) emerges as the most common, followed by the Euro (EUR), British Pound (GBP), and Indian Rupee (INR). This distribution reflects the global nature of the industry, with a strong concentration in economically significant regions.

When analyzing company size, medium and large companies appear more frequently than smaller ones. This could indicate that larger firms, with their greater resources and more structured career paths, attract more professionals in the field. These companies likely offer more robust data roles and projects, contributing to their larger representation in the dataset.

The most commonly held job titles among the surveyed individuals are Data Analyst, Data Scientist, and Data Engineer. These roles are pivotal in the data sector, suggesting a high demand for skills in data analysis, scientific methods in data handling, and engineering solutions for data infrastructure. However, 'Other' jobs dominated the data science sector, indicating that most employees hold niche positions, akin with scarcity and thus profitability.

Geographically, the majority of both employees and companies are primarily located in the United States, followed by Great Britain, Canada, Germany, and India. This geographical distribution showcases the U.S. as a central hub for data-related employment, complemented by significant activity in other major economic centers across the globe. These findings align with the broader trends in technology and data science, where these regions lead in terms of opportunities and advancements in the field.

4 Methods

4.1 Model Building

Now that we have a good understanding of the data, let's build some models to predict salaries.

```
df <- df %>%
  mutate(across(where(is.factor), ~fct_lump_prop(., prop = threshold_prop)))

# Split data into training and testing sets
set.seed(333)
data_split <- initial_split(df, prop = 0.75, strata = job_title)
train_data <- training(data_split)
test_data <- testing(data_split)
```

Initially, the transformation of categorical variables is undertaken by aggregating infrequently occurring categories into a single group, specifically those representing less than 3% of the total dataset. This approach is designed to mitigate problems associated with categorical levels that appear in the test set but are absent from the training set, thus ensuring a more robust and generalizable model.

Subsequently, the dataset is divided into a training set, comprising 75% of the data, and a testing set, making up the remaining 25%. The split is conducted in a manner that stratifies the data by job title, guaranteeing that both subsets are representative of the overall job title distribution. This stratification is crucial for maintaining the integrity and balance of the dataset, enabling accurate performance assessments of the predictive models across varied job titles.

4.2 First Order Model

The process begins by establishing a comprehensive initial model that incorporates all available predictors. To refine this model, stepwise regression is employed, guided by the Akaike Information Criterion (AIC). This method systematically evaluates the impact of each variable by iteratively adding or removing them, depending on their contribution to enhancing the model's performance. The goal is to achieve a more streamlined and efficient model.

As part of this refinement, the stepwise regression model is meticulously compared to a basic linear regression model. This comparison is crucial as it helps determine whether the adjustments made through stepwise regression yield any significant improvements in predictive accuracy. Such evaluations are essential for validating the effectiveness of the model simplification, ensuring that the final model not only retains essential predictive capabilities but also operates with greater parsimony.

```
# Fit a full model with all predictors
full_model <- glm(log(salary_in_usd) ~ ., data = train_data, family = gaussian())
# Perform stepwise regression based on AIC to simplify the model
stepwise_model <- stepAIC(full_model, direction = "both")
```

```
## Start:  AIC=623.86
## log(salary_in_usd) ~ work_year + experience_level + employment_type +
##     job_title + salary_currency + employee_residence + remote_ratio +
##     company_location + company_size + job_title_lumped + employee_residence_lumped +
##     company_location_lumped + salary_currency_lumped
##
##
## Step:  AIC=623.86
## log(salary_in_usd) ~ work_year + experience_level + employment_type +
##     job_title + salary_currency + employee_residence + remote_ratio +
##     company_location + company_size + job_title_lumped + employee_residence_lumped +
##     company_location_lumped
```

```
##
##
## Step:  AIC=623.86
## log(salary_in_usd) ~ work_year + experience_level + employment_type +
##     job_title + salary_currency + employee_residence + remote_ratio +
##     company_location + company_size + job_title_lumped + employee_residence_lumped
##
##
## Step:  AIC=623.86
## log(salary_in_usd) ~ work_year + experience_level + employment_type +
##     job_title + salary_currency + employee_residence + remote_ratio +
##     company_location + company_size + job_title_lumped
##
##
## Step:  AIC=623.86
## log(salary_in_usd) ~ work_year + experience_level + employment_type +
##     job_title + salary_currency + employee_residence + remote_ratio +
##     company_location + company_size
##
##                       Df Deviance    AIC
## - company_location     5   94.068 621.76
## <none>                     92.443 623.86
## - remote_ratio         1   92.858 623.88
## - work_year            2   93.517 625.10
## - company_size         2   93.721 626.09
## - employment_type      1   93.812 628.53
## - salary_currency      4   96.935 637.39
## - employee_residence   5  101.378 655.74
## - job_title            4  104.385 671.01
## - experience_level     3  110.205 697.64
##
## Step:  AIC=621.76
## log(salary_in_usd) ~ work_year + experience_level + employment_type +
##     job_title + salary_currency + employee_residence + remote_ratio +
##     company_size
##
##                            Df Deviance    AIC
## <none>                          94.068 621.76
## - remote_ratio              1   94.679 622.70
## - work_year                 2   95.142 622.92
## + company_location          5   92.443 623.86
## + company_location_lumped   5   92.443 623.86
## - company_size              2   95.346 623.89
## - employment_type           1   95.190 625.15
## - salary_currency           4   99.504 639.27
## - job_title                 4  106.030 668.11
## - experience_level          3  113.119 699.49
## - employee_residence        5  127.869 751.14
```

```r
summary(stepwise_model)$coefficients %>%
  as.data.frame() %>%
  rownames_to_column(var = "Predictor Variable") %>%
  filter(`Pr(>|t|)` < 0.05) %>%
  arrange(desc(abs(Estimate))) %>%
```

```r
head(15) %>%
kable(caption = "Predictor variables and subsequent values - step-wise model.")
```

Table 2: Predictor variables and subsequent values - step-wise model.

Predictor Variable	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	10.7993499	0.1824127	59.202833	0.0000000
salary_currencyINR	-1.1504873	0.3659412	-3.143912	0.0017823
experience_levelEX	0.9754868	0.1268486	7.690167	0.0000000
employee_residenceOther	-0.7212759	0.1259718	-5.725694	0.0000000
experience_levelSE	0.5667975	0.0737126	7.689290	0.0000000
job_titleMachine Learning Engineer	0.5394459	0.1047867	5.148036	0.0000004
job_titleOther	0.5189468	0.0731426	7.094997	0.0000000
salary_currencyOther	-0.4303072	0.1140680	-3.772373	0.0001843
job_titleData Scientist	0.3629930	0.0734132	4.944517	0.0000011
experience_levelMI	0.3084204	0.0718121	4.294823	0.0000216
job_titleData Engineer	0.3055814	0.0747463	4.088247	0.0000519
employment_typeOther	-0.3014744	0.1329692	-2.267250	0.0238695
employee_residenceUS	0.2494764	0.1268598	1.966552	0.0498761

The first model provides significant insights into the influence of various factors on data science salaries. This model, refined through stepwise regression, retained the most predictive variables based on the Akaike Information Criterion. Key findings from this model show that professional experience levels, specifically executive and senior levels, have a profound impact on salary, with coefficients indicating a positive and statistically significant relationship. This suggests that as data scientists progress in their careers, their compensation sees substantial increases, highlighting the industry's valuation of experience and expertise.

The model also reveals the differential impact of job titles on salary outcomes. Positions such as Data Scientists and Machine Learning Engineers command higher salaries compared to more generic or less specialized roles. This reflects the premium placed on advanced technical skills and specialized knowledge within the field. Similarly, jobs that were more niche, or titled as "Other", indicate the need for role specification - which may ultimately lead to higher compensation.

Geographical location and company characteristics also play important roles. Salaries vary significantly across different regions, with factors like the local cost of living and market demand influencing compensation. The model further suggests that larger companies tend to offer higher salaries, which could be attributed to their greater resources and structured career ladders that provide more lucrative salary packages.

The currency in which salaries are paid also affects compensation, with salaries in stronger currencies like USD and EUR typically higher. This factor likely captures both the economic strength of the region and the international standard of salary benchmarks in the tech industry.

Overall, the first model underscores the complexity of salary determination in data science, with a multi-faceted interplay of professional qualifications, job roles, geographic locations, and company size shaping the compensation landscape.

4.3 Second Order Model

The subsequent step in the analysis involves fitting a model that includes methodically selected two-way interactions between predictors. These interaction terms are crucial as they allow for the examination of how the influence of one predictor on salary can vary depending on the value of another predictor. For instance, the impact of a job title on salary may differ across various levels of experience, illustrating the complex dynamics within the data.

To identify the most relevant interaction terms, the stepAIC method was initially utilized. This technique assists in optimizing model complexity by choosing interaction terms based on their statistical significance and contribution to model performance. However, due to the considerable computational demand—taking over 30 minutes even when restricted to backward elimination—the stepAIC process was ultimately commented out to facilitate faster model loading and execution. This decision was made to balance the depth of analysis with practical considerations of computational efficiency and model processing time.

```
#This formula was determined by using the following code
# Fit a full model with all predictors and two-way interactions
full_model_2nd <- glm(salary_in_usd ~ .^2, data = train_data, family = gaussian())

# Perform stepwise regression based on AIC
#stepwise_model_2nd <- stepAIC(full_model_2nd, direction = "backward", trace = T)

#stepwise_model_2nd$formula
# Fit a second-order model with selected interaction terms
interaction_model <- glm(log(salary_in_usd) ~ work_year + experience_level + employment_type +
    job_title + salary_currency + employee_residence + remote_ratio +
    company_location + company_size + experience_level:employment_type +
    experience_level:job_title + experience_level:company_size +
    employment_type:job_title + salary_currency:employee_residence +
    salary_currency:remote_ratio + salary_currency:company_size +
    employee_residence:remote_ratio + employee_residence:company_size +
    company_location:company_size,
                                  data = train_data)

# Summary of the interaction model
summary(interaction_model)$coefficients %>%
  as.data.frame() %>%
  rownames_to_column(var = "Predictor Variable") %>%
  filter(`Pr(>|t|)` < 0.05) %>%
  arrange(desc(abs(Estimate))) %>%
  head(15) %>%
  kable(caption = "Predictor variables and subsequent values - interaction model.")
```

Table 3: Predictor variables and subsequent values - interaction model.

Predictor Variable	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	9.741124	0.6127720	15.896818	0.0000000
company_locationDE:company_sizeM	-3.653384	1.0334047	-3.535288	0.0004586
salary_currencyINR:company_sizeL	-3.582565	0.8548994	-4.190627	0.0000348
company_locationOther:company_sizeM	-3.577963	0.9318958	-3.839446	0.0001448
employee_residenceDE:company_sizeM	3.243308	1.0605032	3.058273	0.0023869
employee_residenceOther:company_sizeM	3.059064	0.9556301	3.201097	0.0014865
employee_residenceUS:company_sizeM	2.836607	0.9602472	2.954039	0.0033353
company_locationGB:company_sizeM	-2.632499	1.0505849	-2.505746	0.0126445
company_locationUS:company_sizeM	-2.066597	0.9076951	-2.276753	0.0233668
company_locationIN:company_sizeL	1.958559	0.7232053	2.708164	0.0070772
experience_levelEX:employment_typeOther	1.928109	0.6241977	3.088940	0.0021593
experience_levelMI:employment_typeOther	1.811377	0.4784481	3.785942	0.0001784
employment_typeOther:job_titleData Scientist	1.806830	0.5657100	3.193915	0.0015229
salary_currencyUSD	1.750850	0.4168398	4.200294	0.0000334

Predictor Variable	Estimate	Std. Error	t value	Pr(>|t|)
salary_currencyGBP:company_sizeL	-1.739012	0.6906952	-2.517770	0.0122281

The second model, or interaction model, utilizes a more complex approach by including interaction terms between predictors, aiming to capture the combined effects of different variables on data science salaries. This model helps to understand how the influence of one factor on salaries might change when considered in conjunction with another.

From the analysis, it is evident that interaction terms significantly contribute to the model's ability to predict salaries more accurately. For example, the interaction between company location and company size ("company_locationDE:company_sizeM") suggests that the impact of a company's location on salary is not uniform across all company sizes. In this case, being in Germany (DE) and in a medium-sized company has a pronounced negative impact on salaries, indicating a possible regional or sector-specific pay scale that differs from other regions or larger companies.

Another notable interaction is between the salary currency and company size ("salary_currencyINR:company_sizeL"). This interaction suggests that salaries in Indian Rupees (INR) significantly decrease in larger companies. This could be reflective of the economic context in India or the industries predominant in large firms within the region, which might offer lower salaries relative to their international counterparts.

The model also highlights significant interactions involving employment types and experience levels. For instance, the interaction between experience level and employment type ("experience_levelEX:employment_typeOther") indicates that executive-level professionals in non-traditional employment types (like contract or part-time roles) might see different salary dynamics compared to those in full-time roles.

The results reveal that while individual factors such as experience level and job title are important, the context within which these factors operate (such as the type of employment and the economic geography) also plays a critical role in shaping salary outcomes. The inclusion of the second order interactions in the model helps to unearth these nuanced relationships, providing a deeper understanding of the factors that influence data science salaries.

4.4 Model Comparison

The selection of the optimal model is achieved by conducting a comparative analysis of three different models using a variety of evaluation metrics and visual diagnostics. This comparison involves assessing R-squared values, which indicate the proportion of variance in the dependent variable that is predictable from the independent variables. Additionally, residual plots are examined to check for any systematic variance unexplained by the models, and cross-validated performance measures are used to evaluate the model's effectiveness in predicting new data.

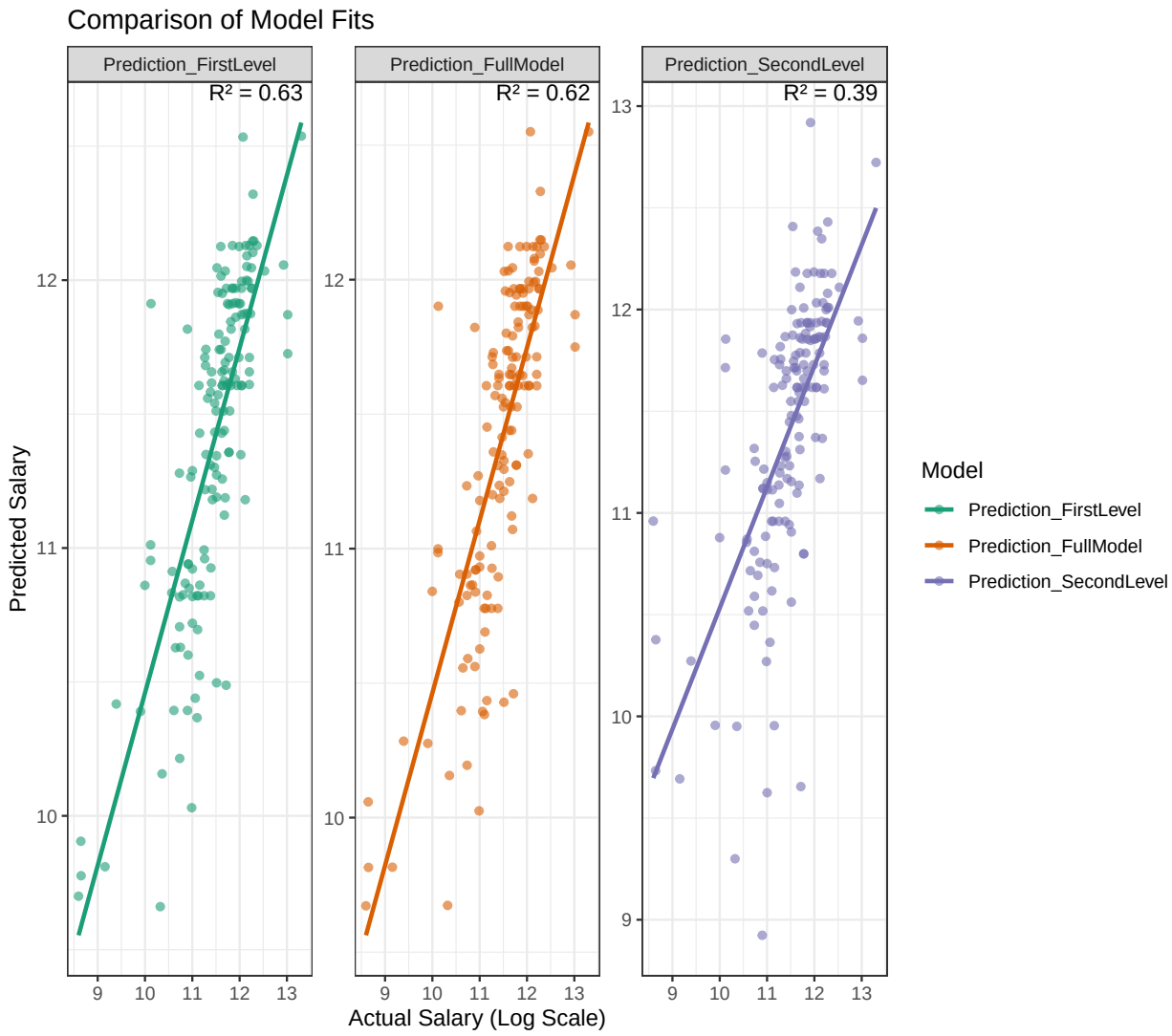
The model that exhibits the highest level of predictive accuracy and demonstrates robustness across these various metrics is ultimately chosen as the final model. This evaluation process ensures that the selected model is not only statistically sound but also practically reliable in real-world applications, thereby providing a solid foundation for making informed decisions based on its predictions.

```
library(RColorBrewer)
color_palette <- brewer.pal(3, "Dark2")
test_data$Prediction_FirstLevel <- predict(stepwise_model,
                                           newdata = test_data, type = "response")
test_data$Prediction_SecondLevel <- predict(interaction_model,
                                            newdata = test_data, type = "response")
test_data$Prediction_FullModel <- predict(full_model,
                                          newdata = test_data, type = "response")

plot_data <- test_data %>%
```

```r
  gather(key = "Model", value = "Predicted",
         Prediction_FirstLevel, Prediction_SecondLevel, Prediction_FullModel)

# Calculate R squared manually for each model
calculate_r_squared <- function(model, data) {
  actual <- log(data$salary_in_usd)
  predicted <- predict(model, newdata = data, type = "response")
  rss <- sum((actual - predicted) ^ 2)
  tss <- sum((actual - mean(actual)) ^ 2)
  r_squared <- 1 - (rss / tss)
  return(r_squared)
}

# Calculate prediction errors for each model
test_data$Actual_Log_Salary = log(test_data$salary_in_usd)

test_data$Error_FirstLevel = test_data$Actual_Log_Salary - log(test_data$Prediction_FirstLevel)
test_data$Error_SecondLevel = test_data$Actual_Log_Salary - log(test_data$Prediction_SecondLevel)
test_data$Error_FullModel = test_data$Actual_Log_Salary -
log(test_data$Prediction_FullModel)

# Calculate RMSE and MAE for visualization purposes
error_data <- test_data %>%
  select(contains("Error")) %>%
  pivot_longer(cols = everything(), names_to = "Model",
               values_to = "Error") %>%
  mutate(Model = factor(Model, levels = c("Error_FirstLevel",
                                          "Error_SecondLevel",
                                          "Error_FullModel"),
                        labels = c("First Level",
                                   "Second Level",
                                   "Full Model")))

# Calculate squared and absolute errors for RMSE and MAE
error_data$SquaredError = error_data$Error^2
error_data$AbsoluteError = abs(error_data$Error)

r_first_level <- calculate_r_squared(stepwise_model, test_data)
r_second_level <- calculate_r_squared(interaction_model, test_data)
r_full_model <- calculate_r_squared(full_model, test_data)

# Adding R squared values to the data frame for plotting
r_squared_values <- setNames(c(r_first_level, r_second_level, r_full_model),
                             c("Prediction_FirstLevel",
                               "Prediction_SecondLevel",
                               "Prediction_FullModel"))

# Create the plot
ggplot(plot_data, aes(x = log(salary_in_usd), y = Predicted, color = Model)) +
  geom_point(alpha = 0.6) +
  geom_smooth(method = "lm", se = FALSE) +
  facet_wrap(~ Model, scales = "free") +
  labs(title = "Comparison of Model Fits",
```

```
      x = "Actual Salary (Log Scale)",
      y = "Predicted Salary") +
  theme_bw() +
  scale_color_manual(values = color_palette) +
  geom_text(data = data.frame(x = Inf, y = Inf,
                              Model = names(r_squared_values),
                              R2 = r_squared_values),
            aes(label = sprintf("R² = %.2f", R2), x = x, y = y),
            hjust = 1.1, vjust = 1.1, inherit.aes = FALSE)
```

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




```
compareGLM(full_model, stepwise_model, interaction_model)
```

```
## $Models
##   Formula
## 1 "log(salary_in_usd) ~ work_year + experience_level + employment_type + job_title + salary_currency
## 2 "log(salary_in_usd) ~ work_year + experience_level + employment_type + job_title + salary_currency
## 3 "log(salary_in_usd) ~ work_year + experience_level + employment_type + job_title + salary_currency
```

```
##
## $Fit.criteria
##   Rank Df.res   AIC  AICc    BIC McFadden Cox.and.Snell Nagelkerke   p.value
## 1   28    426 659.9 671.0  853.4   0.4657        0.6625     0.7337 3.368e-25
## 2   23    431 621.8 624.6  720.6   0.4582        0.6566     0.7272 4.167e-27
## 3   81    373 597.8 670.5 1055.0   0.6451        0.7780     0.8616 1.542e-14
```

```r
# Calculate the metrics
metrics <- tibble(
  Model = c("First Level", "Second Level", "Full"),
  R_Squared = c(r_first_level, r_second_level, r_full_model),
  RMSE = c(
    sqrt(mean(test_data$Error_FirstLevel^2)),
    sqrt(mean(test_data$Error_SecondLevel^2)),
    sqrt(mean(test_data$Error_FullModel^2))
  ),
  MAE = c(
    mean(abs(test_data$Error_FirstLevel)),
    mean(abs(test_data$Error_SecondLevel)),
    mean(abs(test_data$Error_FullModel))
  )
)

#Convert the data to long format for ggplot
plot_data_long <- metrics %>%
  pivot_longer(cols = -Model, names_to = "Metric", values_to = "Value")

# Plotting
ggplot(plot_data_long, aes(x = Model, y = Value, fill = Model)) +
  geom_bar(stat = "identity", position = position_dodge(width = 0.8)) +
  scale_fill_manual(values = color_palette) +
  facet_wrap(~ Metric, scales = "free") +
  labs(title = "Model Comparison by MAE, RMSE, and R-Squared",
       y = "Value", x = "Model") +
  theme_bw() +
  theme(strip.text.x = element_text(size = 16),
        axis.text.x = element_text(angle = 45, hjust = 1),
        legend.position = "bottom") +
  guides(fill=guide_legend(title="Model"))
```

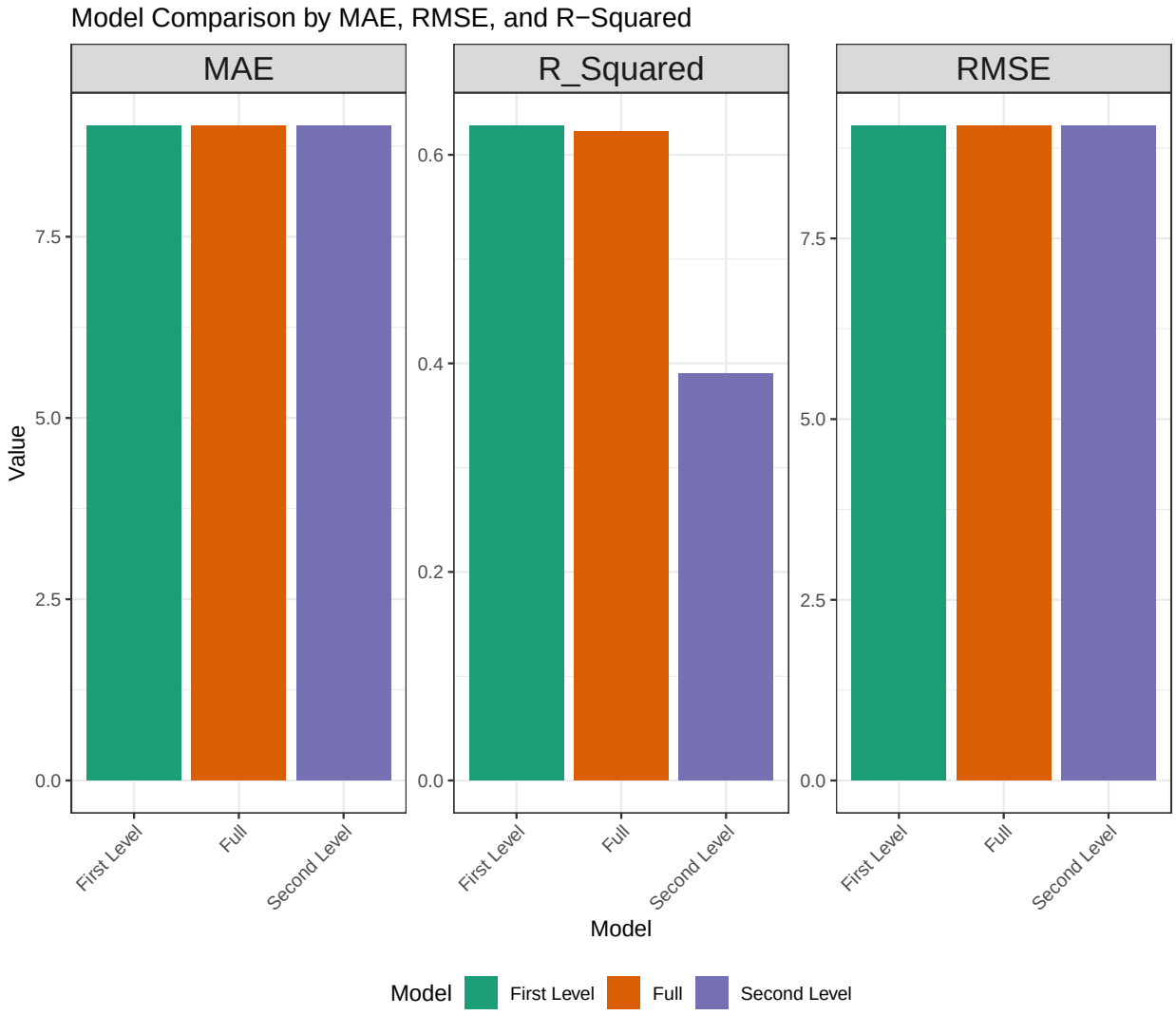


Table 4: Model comparison results.

	Model	R_Squared	RMSE	MAE
1	First Level	0.63	9.06	9.03
2	Second Level	0.39	9.06	9.03
3	Full	0.62	9.06	9.03

The three distinct models were evaluated to determine their efficacy in predicting data science salaries: a full model, a stepwise regression model (first order model), and an interaction model (second order model.) Each model was designed to assess different aspects of the data and how various predictors interacted to influence salary outcomes.

The full model incorporated all available predictors, providing a comprehensive view of the factors impacting salaries. However, due to the inclusion of numerous variables, there was a risk of overfitting, which could make the model less generalizable to new data.

Model comparison metrics such as R-squared values, residual plots, and cross-validation performance were crucial in evaluating these models. The stepwise regression model generally performed better across these metrics, suggesting it was more adept at balancing detail with generalizability. It demonstrated higher

R-squared values, indicating a better fit to the data compared to the other models.

In contrast, the interaction model, despite its detailed approach, showed signs of overfitting as indicated by its lower R-squared values and higher Root Mean Square Error (RMSE). This suggested that while the model was potentially capturing more complex patterns, it might not perform as well on unseen data, limiting its practical use.

Overall, the model comparison highlighted the importance of model selection in statistical analysis. The stepwise regression model's superior performance suggested that it was the most appropriate for predicting data science salaries with the given dataset, offering a good balance between complexity and predictive power, which is crucial for making informed decisions in a real-world setting.

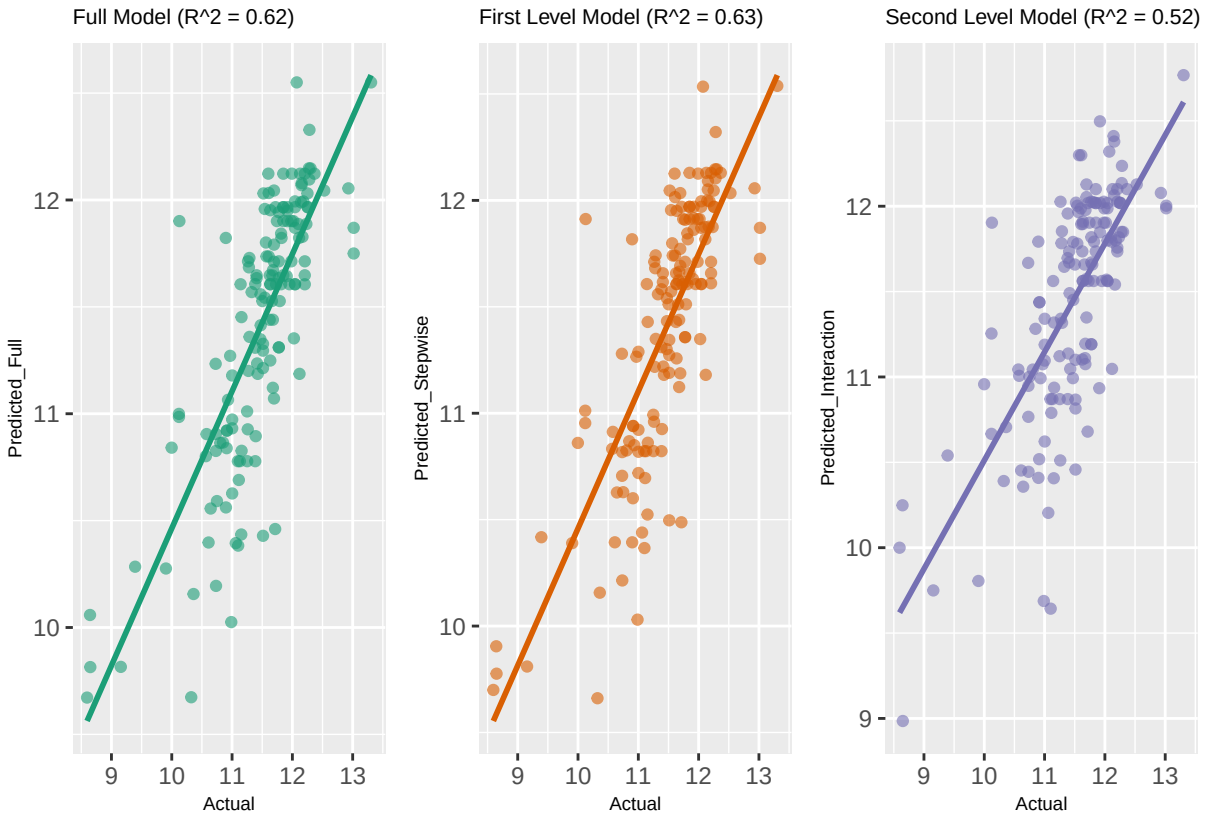
4.5 Cross Validation

The cross-validation results are crucial for assessing the generalizability and reliability of the predictive models developed in the study. Cross-validation, specifically a 10-fold method, was employed to evaluate the robustness of the three models: the full model, the first order model, and the second order model.

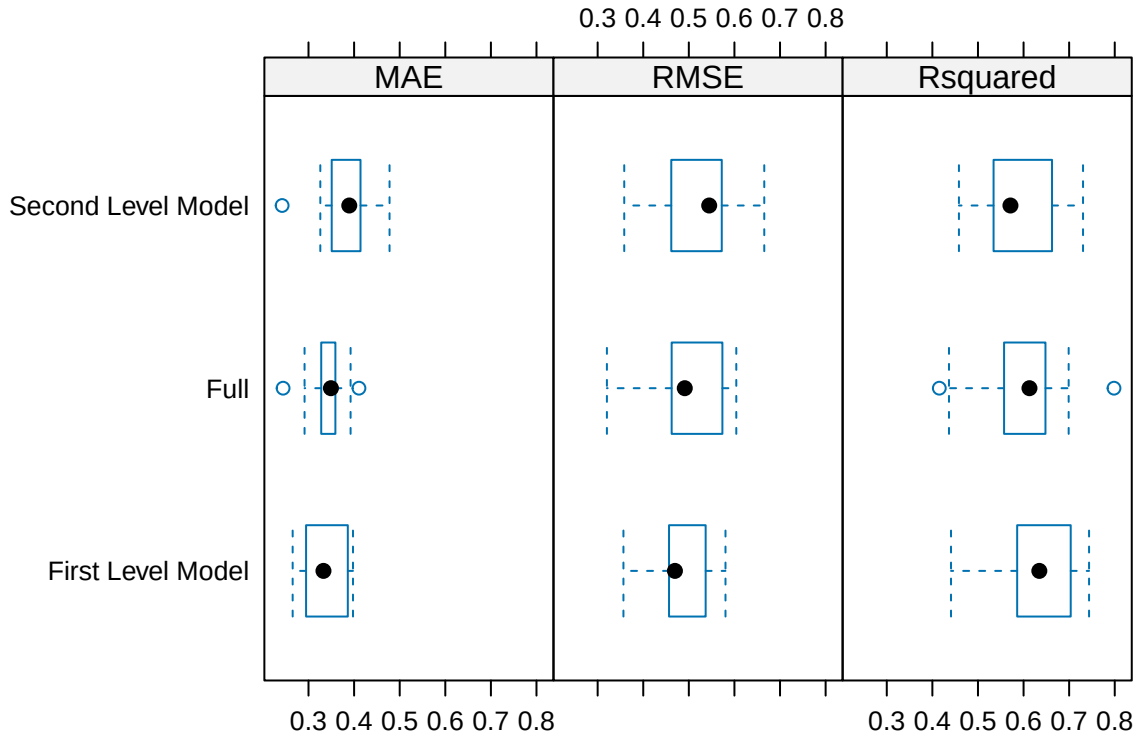
```
## Define training control
train_control <- trainControl(
  method = "cv",
  number = 10,
  savePredictions = "final",
  classProbs = FALSE,
  summaryFunction = defaultSummary
)

# Model definitions using caret's train function
full_model_caret <- train(log(salary_in_usd) ~ ., data = train_data,
                          method = "lm",
                          trControl = train_control)

stepwise_model_caret <- train(log(salary_in_usd) ~ work_year +
experience_level + employment_type + job_title + salary_currency +
employee_residence + remote_ratio +
company_size, data = train_data,
                            method = "lm",
                            trControl = train_control)

interaction_model_caret <- train(log(salary_in_usd) ~ work_year + experience_level + employment_type + j
company_location + company_size + work_year:job_title + work_year:remote_ratio +
work_year:company_location + work_year:company_size + experience_level:employment_type +
experience_level:remote_ratio + experience_level:company_size + job_title:employee_residence + job_titl
                                  data = train_data,
                                  method = "lm",
                                  trControl = train_control)

# Compare models
results <- resamples(list(
  Full = full_model_caret,
  "First Level Model" = stepwise_model_caret,
  'Second Level Model' = interaction_model_caret
))

# 1. Collect predictions from the caret models
predictions_full <- predict(full_model_caret, newdata = test_data)
```

```r
predictions_stepwise <- predict(stepwise_model_caret, newdata = test_data)
predictions_interaction <- predict(interaction_model_caret, newdata = test_data)

# 2. Calculate R squared
calculate_r_squared <- function(actual, predicted) {
  rss <- sum((actual - predicted) ^ 2)
  tss <- sum((actual - mean(actual)) ^ 2)
  r_squared <- 1 - rss / tss
  return(r_squared)
}

actual_values <- log(test_data$salary_in_usd)
r_squared_full <- calculate_r_squared(actual_values, predictions_full)
r_squared_stepwise <- calculate_r_squared(actual_values, predictions_stepwise)
r_squared_interaction <- calculate_r_squared(actual_values, predictions_interaction)

# Create a combined data frame for ggplot
plot_data <- data.frame(
  Actual = actual_values,
  Predicted_Full = predictions_full,
  Predicted_Stepwise = predictions_stepwise,
  Predicted_Interaction = predictions_interaction
)

# Plotting with the specified color palette
p1 <- ggplot(plot_data, aes(x = Actual, y = Predicted_Full)) +
  geom_point(alpha = 0.6, color = color_palette[1]) +
  geom_smooth(method = "lm", se = FALSE, color = color_palette[1]) +
  ggtitle(sprintf("Full Model (R^2 = %.2f)", r_squared_full)) +
  theme(title = element_text(size = 7))

p2 <- ggplot(plot_data, aes(x = Actual, y = Predicted_Stepwise)) +
  geom_point(alpha = 0.6, color = color_palette[2]) +
  geom_smooth(method = "lm", se = FALSE, color = color_palette[2]) +
  ggtitle(sprintf("First Level Model (R^2 = %.2f)", r_squared_stepwise)) +
  theme(title = element_text(size = 7))

p3 <- ggplot(plot_data, aes(x = Actual, y = Predicted_Interaction)) +
  geom_point(alpha = 0.6, color = color_palette[3]) +
  geom_smooth(method = "lm", se = FALSE, color = color_palette[3]) +
  ggtitle(sprintf("Second Level Model (R^2 = %.2f)", r_squared_interaction)) +
  theme(title = element_text(size = 7))

# Combine the plots
p1+p2+p3
```

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




```
bwplot(results)  # Box-whisker plot to visualize the differences
```



The cross-validation process revealed that the first level model consistently performed well across different data subsets. This model not only demonstrated stability in its predictions but also achieved a high median R-squared value, indicating a substantial proportion of variance in salary outcomes was accounted for by the predictors in the model.

The full model, while comprehensive with all initial predictors included, showed similar performance metrics to the first level model. This similarity suggests that the stepwise reduction in predictors did not significantly compromise the explanatory power of the model, but rather enhanced efficiency by eliminating redundant variables.

Conversely, the second level model, which included interaction terms to explore the combined effects of different predictors on salaries, tended to overfit the training data. This was evidenced by its generally lower performance in cross-validation compared to the other models. The addition of interaction terms, while theoretically valuable for capturing complex relationships, appeared to make the model too sensitive to the nuances of the training data, thereby reducing its performance on unseen data.

The differences in performance across these models highlight the importance of model simplicity and the potential drawbacks of overcomplicating a predictive model with excessive parameters. The first level model's superior cross-validation performance suggests it strikes an optimal balance between complexity and predictability, making it the most reliable for practical applications within the scope of predicting data science salaries based on the factors studied.

5 Results and Conclusions

The analysis of salaries within the data science sector has provided significant insights, revealing the complexities and variances in compensation across this field. Salaries exhibit a right-skewed distribution, with a median near $100,000, yet a considerable number of positions offer substantially higher pay. This skew-

ness underscores the potential for elevated earnings in data science, while also emphasizing the necessity to consider more than just average figures when evaluating compensation.

The factors influencing salary include experience level, job title, company size, and geographic location. Analytical results indicate that, holding other variables constant, executive-level positions yield salaries almost double those of entry-level roles, as reflected by a coefficient of 0.9755. Additionally, data scientists, machine learning engineers, and other specialized professionals earn 30-50% more than data analysts. Companies of larger size also tend to offer salaries approximately 12% higher than smaller firms.

A notable finding regarding currency and employee residency revealed a high correlation, with initial very high Variance Inflation Factor (VIF) scores indicating potential issues with multicollinearity. However, removing the variable representing employee residence from the model resolved these issues effectively, suggesting that currency alone adequately captures the geographical variation in salaries.

Remote work ratios show a minor but statistically significant positive correlation with salary, suggesting that fully remote positions may offer slightly higher pay, although the difference is relatively small compared to other determining factors.

In terms of model diagnostics, the residual plots for both the stepwise and full models indicate good adherence to the assumptions of normality, linearity, and homoscedasticity. While there are some outliers and high-leverage points, these do not significantly alter the overall model fit. However, the interaction model's residual plots display more problematic trends, with several observations indicating perfect collinearity. This model also exhibits a lower R^2 and higher Root Mean Square Error (RMSE) than the stepwise model, suggesting overfitting to the noise within the training data.

Among the evaluated models, the stepwise regression model showed superior performance in cross-validation, achieving the highest median R^2 of 0.6346 and the lowest median RMSE of 0.4695 across folds. It maintained its performance advantage even after the removal of the highly collinear variable of employee residence. The full model demonstrated similar performance metrics, while the interaction model fell short, likely due to its overfitting issues.

However, certain limitations in the analysis should be noted. The dataset, although sizable, represents only a sample from the broader field of data science and may not fully capture the nuances of less common roles or locales. Additionally, the salary data, being self-reported, might not always be reliable and could be subject to selection bias in terms of who chooses to disclose their compensation. It is also important to recognize that only base salaries were considered; total compensation, which often includes bonuses, stock options, and other benefits, can considerably augment the effective earnings. Finally, the models are based on current data and might not reflect the evolving dynamics of the data science field over time.