# Predicting Duration of Insurance Complaints

Anthony J. Mercure

## 1 Introduction

Insurance disputes are a notable area of concern, affecting both consumers and the regulatory bodies tasked with enforcing varying insurance industries' compliance. This investigation presents a regression analysis of complaint duration in insurance sectors, leveraging complaint data from the Texas Department of Insurance (TDI) compiled over the last 12 years. Our study includes records of complaints lodged against insurance entities, including licensed individuals and companies. Our analysis is framed around understanding factors influencing complaint resolution times, aiming to explore potential predictors, including insurance coverage types, respondent types, and previous complaint histories. We employ multiple linear regression to examine how these factors influence the duration of complaint resolution. To enhance model reliability, diagnostic tests were conducted, and transformations on some variables were applied, with some outliers and influential observations removed to ensure more robust predictive capabilities. This analysis highlights the complex dynamics within insurance complaint resolution processes, suggesting that targeted regulatory measures could improve response efficiency. This examination contributes to the field of public policy and consumer protection, offering data-driven recommendations to refine complaint management strategies and enhance consumer protection and satisfaction within the insurance industry.

The dataset comprises 258,229 complaints filed against 5,286 unique entities, including licensed individuals and companies. Of these, 202,833 complaints were not confirmed as errors, while 38,673 complaints were substantiated as errors by the entity being filed against. Key variables in the dataset include Complaint Duration, the dependent variable, calculated as the difference between the received and closed dates of a complaint. Predictor variables encompass Coverage

Type, categorized into Accident & Health, Life & Annuity, Automobile, and others; Respondent Type, distinguishing between organizations and individuals; and Confirmed Complaint, a binary indicator signifying whether the TDI confirmed the licensed entity was at fault. Additionally, the dataset captures such quantitative metrics as Time Between Complaints, representing the interval between successive complaints for the same entity, and Number of Previous Complaints, reflecting the cumulative history of complaints filed against a respondent at the given instance of the individual complaint. This comprehensive dataset provides a robust foundation for analyzing the factors influencing complaint resolution times, identifying significant predictors, and proposing data-driven operational or policy-level recommendations to enhance resolution efficiency.

## 2 Graphical Presentation and Summary Statistics

To begin the analysis, a thorough examination of the dataset was performed to uncover underlying patterns and trends. Summary statistics for the quantitative variables are presented in Table 1. These descriptive measures offer insights into the distribution, variability, and scale of the data, providing a foundation for subsequent modeling.

| **Variable** | **Min** | **25th** | **Median** | **Mean** | **75th** | **Max** | **Std. Deviation** |
|---|---|---|---|---|---|---|---|
| Complaint Duration | 1.0 | 37.0 | 72.0 | 109.3 | 126.0 | 793.0 | 118.168 |
| Time Between Complaints | 0.00 | 0.00 | 1.00 | 18.41 | 4.00 | 3389.00 | 109.2905 |
| Number of Previous Complaints | 0 | 113 | 664 | 2391 | 2598 | 20261 | 3900.639 |

Table 1: Summary Statistics for Quantitative Variables

Figure 1 depicts the distribution of complaint durations, revealing a right-skewed pattern. The majority of complaints are resolved within a shorter time frame, while a small number of outliers significantly extend the average duration. A closer examination, stratified by duration class, is illustrated in Figure 2, showcasing how resolution times are distributed across the predefined complaint categories of *Short*, *Moderate*, *Long*, and *Extended*.

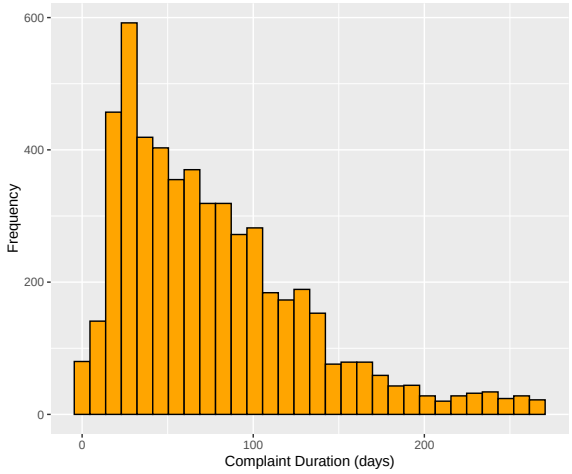


Figure 1: Histogram of Complaint Duration

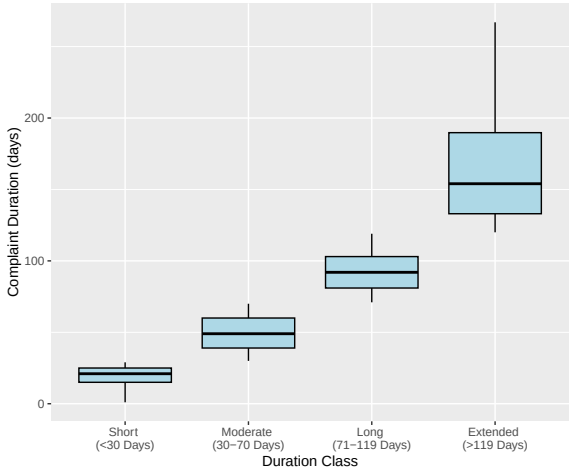


Figure 2: Boxplots of Complaint Duration

The categorical variables, particularly *Coverage Type* and *Respondent Type*, were also explored to identify their distributions and relationships to resolution time. Figure 3 illustrates the frequency of complaints across different coverage types, with *Automobile*, *Accident & Health*, and *Life & Annuity* emerging as the most common. Furthermore, Figure 4 visualizes the interaction between respondent type and coverage type, providing insights into how these factors jointly influence complaint patterns.

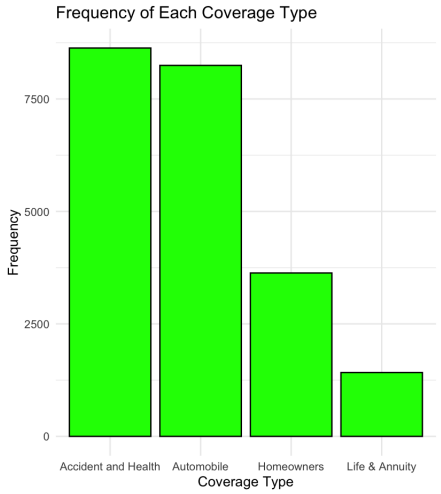


Figure 3: Bar Plot of Coverage Type Frequencies

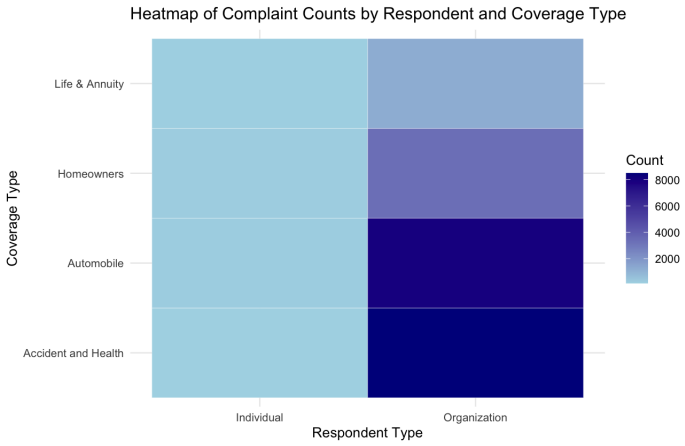


Figure 4: Heatmap of Complaint Counts by Respondent and Coverage Type

The scatter plot in Figure 5 highlights the relationship between *Complaint Duration* and *Time Between Complaints*, suggesting potential linear trends while also indicating outliers that could influence model stability.

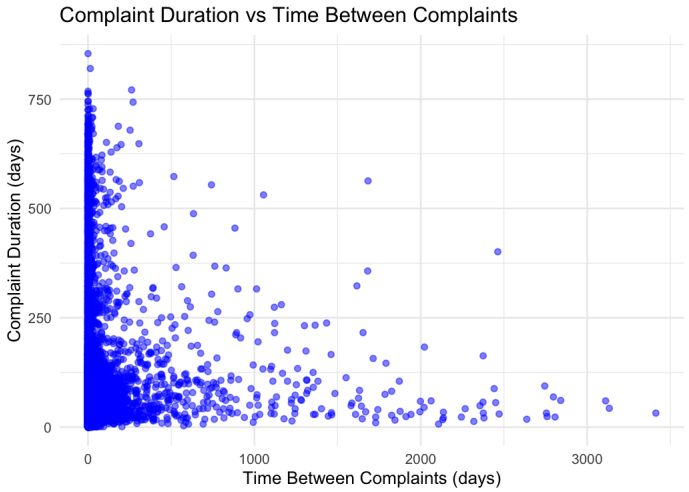


Figure 5: Scatter Plot of Complaint Duration vs. Time Between Complaints

These exploratory visuals not only confirm the variability and complexity within the dataset but also underscore the necessity of robust statistical modeling to uncover meaningful patterns. The insights from these visualizations inform the selection and transformation of variables used in the regression analysis.

## 3 Data Preprocessing & Preparation

To ensure suitability for regression analysis and to maintain data integrity, the dataset was cleaned and preprocessed. Initially, complaints with a zero duration were excluded, as such records typically represent immediately terminated complaints that provide no meaningful information about complaint resolution times that could distort the analysis. Additionally, entities with fewer than two complaints were removed from the dataset to focus the analysis on entities with significant complaint histories, providing a more robust understanding of patterns across repeat occurrences. Respondent Type, Coverage Type, and Confirmed Complaint, were carefully factorized to enhance interpretability and model compatibility. The Respondent Type variable categorized complaints based on whether they were filed against an individual or an organization. The Coverage Type

variable included the most common categories of Accident & Health, Life & Annuity, and Automobile, while Confirmed Complaint was a binary variable indicating whether the TDI confirmed the complaint as valid.

To manage efficiency while maintaining representativeness, a 4% random sample of the dataset, comprising 7,940 observations, was selected for analysis. Outliers were identified and removed using the interquartile range method, resulting in the exclusion of 2,742 extreme values across key quantitative variables. Furthermore, influential points were identified using Cook's Distance and leverage values, and 914 such observations were excluded to mitigate their influence on the regression model, thus improving its robustness and reliability. To address issues of non-linearity and heteroscedasticity, a Box-Cox variable transformation was applied to the dataset. This transformation identified an optimal lambda value of $\lambda = 0.1414$, which was subsequently used to log-transform the variables Complaint Duration and Time Between Complaints. This step was instrumental in resulting in a more symmetric distribution, stabilized variance, and addressed issues of heteroscedasticity, thereby meeting the fundamental assumptions of regression analysis.

## 4 Multiple Linear Regression

The multiple linear regression analysis was conducted to quantify the impact of both quantitative and categorical predictors on complaint resolution times. The regression model is expressed as:

$$\ln(\text{CD}) = \beta_0 + \beta_1(\ln(\text{TbC})) + \beta_2(\text{NPC}) + \beta_3(\text{RO}) + \beta_4(\text{CfN}) + \beta_5(\text{CvAH}) + \varepsilon$$

Where:

- ln(CD): Log-transformed complaint duration (response variable).
- ln(TbC): Log-transformed time between complaints.
- NPC: Number of previous complaints.
- RO: Binary indicator for respondent type (organization vs. individual).
- CfN: Binary indicator for complaint confirmation (confirmed vs. unconfirmed).

- CvAH: Categorical variable for Accident & Health coverage types.

The regression coefficients and their significance levels are summarized in Table 2. These results provide insights into the relationships between predictors and complaint resolution durations.

| **Coefficient** | **Estimate** | **p-value** |
|---|---|---|
| Intercept | 4.333 | <0.001*** |
| ln(Time Between Complaints) | 0.001 | 0.881 |
| Number of Previous Complaints | -0.00002 | 0.008** |
| Respondent Type (Org.) | -0.139 | 0.651 |
| Confirmed Complaint (No) | -0.225 | <0.001*** |
| Coverage Type (Accident & Health) | 0.006 | 0.812 |

Table 2: Regression Model Summary

The regression analysis reveals several important relationships:

- **Time Between Complaints:** The variable was not statistically significant ($p = 0.881$), indicating no strong evidence of an association between the time interval of successive complaints and the duration of complaint resolution.

- **Number of Previous Complaints:** This variable exhibited a statistically significant negative relationship with complaint duration ($p = 0.008$). Entities with more previous complaints tended to resolve subsequent complaints slightly faster, potentially due to improved familiarity with resolution procedures or streamlined processes.

- **Respondent Type:** When comparing organizations to individuals, the respondent type was not statistically significant ($p = 0.651$). This suggests no meaningful difference in resolution times between these two respondent categories.

- **Confirmed Complaint (No):** This variable was highly significant ($p < 0.001$), indicating that complaints not confirmed as valid by the Texas Department of Insurance were resolved significantly faster than confirmed complaints. This likely reflects simplified processes for unsubstantiated claims.

- **Coverage Type (Accident & Health):** The estimate for Accident & Health coverage type was not statistically significant ($p = 0.812$), indicating no significant difference in resolution times compared to the baseline category, Life & Annuity.

- **Coverage Type (Automobile):** This variable was excluded from the final model due to high multicollinearity with Accident & Health, which could distort coefficient estimates and reduce interpretability.

The 95% confidence intervals for the regression coefficients are presented in Table 7, providing a range within which the true population parameters are likely to lie. The confidence intervals confirm that complaint confirmation, specifically the not confirmed, and the number of previous complaints significantly impact resolution times, with unconfirmed complaints resolved faster and prior complaint history associated with slight efficiency gains. Conversely, variables like respondent type, time between complaints, and coverage type (Accident & Health) show no significant effects, highlighting their limited predictive value in explaining resolution duration.

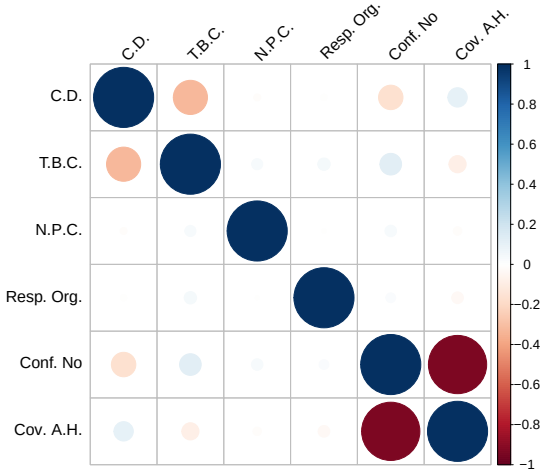


Figure 6: Correlation Matrix

| Variable | Lower 5% | Upper 5% |
|---|---|---|
| Intercept | 3.728 | 4.938 |
| T.b.C. | -0.006 | 0.007 |
| N.P.C. | 0.000005 | 0.00004 |
| Resp. Org. | -0.742 | 0.463 |
| Conf. No | -0.290 | -0.153 |
| Cov. A.H. | -0.045 | 0.058 |

Figure 7: 95% Confidence Intervals for Regression Coefficients

## 5 Regression Diagnostics

The regression diagnostics reveal mixed findings regarding model assumptions. The Residuals vs. Fitted Values plot shows no significant curvature, indicating that the linearity assumption is reasonably upheld. However, the dispersion of residuals across fitted values suggests minor

heteroscedasticity, which may warrant further consideration. The Q-Q Plot indicates that most residuals align with the theoretical quantiles, supporting approximate normality for the majority of the data. Nonetheless, deviations at the tails persist even after removing outliers and influential points, highlighting residual non-normality in extreme values that could impact inference reliability.

The Scale-Location Plot shows a relatively consistent spread of residuals across fitted values, supporting the assumption of homoscedasticity, except for a slight upward trend that suggests a minor deviation from constant variance. The Residuals vs. Leverage Plot confirms that no points exceed the Cook's distance threshold, suggesting the absence of observations with excessive influence. However, some residuals near the leverage threshold persist, even after removing influential observations. These findings collectively indicate that while the model assumptions are generally reasonable, minor issues with variance consistency and residual normality remain, which could influence the precision of estimates and predictions.

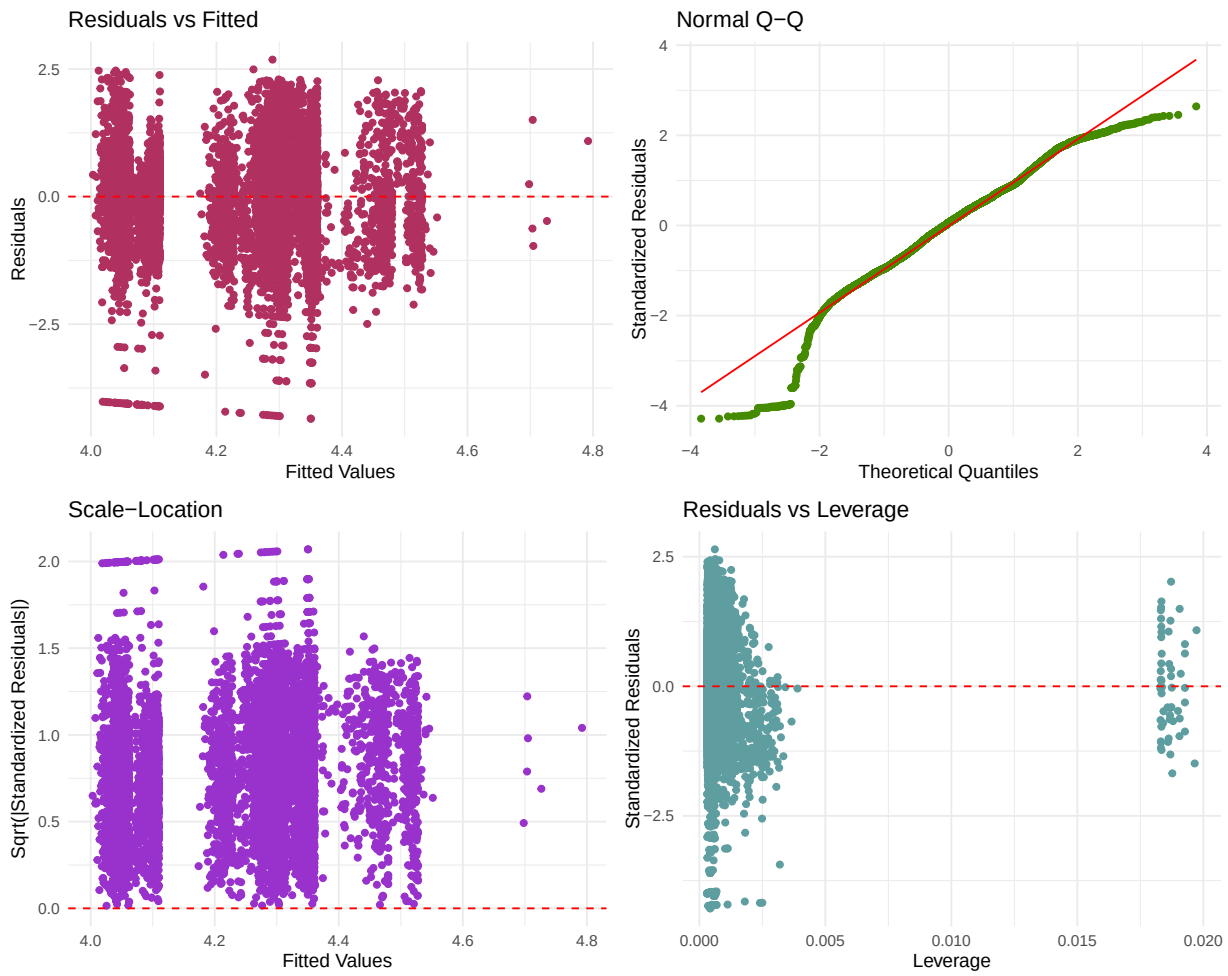


Figure 8: Regression Diagnostics Plots

## 6 Conclusion

Overall, the regression analysis of insurance complaints in Texas has provided significant insights into the factors influencing complaint resolution durations. For instance, the suggestion that entities with more prior complaints resolve subsequent complaints slightly faster, potentially due to improved procedural efficiencies. Similarly, when complaints were not confirmed as the entity's error, resolution times were significantly shorter, reflecting the streamlined handling of such cases. Despite the improvements made through transformations and diagnostic refinements, some issues related to the linearity of predictors and the overall model fit remain evident in the regression diagnostics. These challenges suggest that future work should focus on exploring advanced statistical

techniques, such as generalized least squares (GLS) or weighted least squares (WLS), which could address the identified heteroscedasticity, while robust regression methods may improve resilience against outliers and leverage points.

Furthermore, integrating machine learning approaches could address the limitations inherent in linear regression. Models such as random forests, gradient boosting, and support vector regression could capture non-linear relationships and interactions more effectively. Moreover, deep learning methods, including neural networks, could be explored for their capacity to model complex, high-dimensional data patterns. These models could be evaluated using cross-validation to ensure generalizability and compared against traditional regression techniques to assess performance gains in predictive accuracy. With these advancements, hyperparameter optimization frameworks should also be included to refine model configurations, as well as leveraging interpretability tools, to ensure that insights remain actionable for policy and operational decisions. Another means of enhancement could consider temporal patterns in complaint data using time-series analysis or recurrent neural networks to provide valuable insights into how complaint behaviors evolve over time, informing more proactive policy interventions. By addressing these avenues, future work can refine the analytical tools applied to complaint resolution data, improving the reliability of insights and supporting more effective regulatory and operational strategies in the insurance industry.

## References

Texas Department of Insurance. Insurance complaints: All data, 2024. URL `https://catalog.data.gov/dataset/insurance-complaints-all-data`. Accessed: October 21, 2024.

# 7 Appendix: R Code

```
library(dplyr)
library(lubridate)
library(ggplot2)
library(broom)
library(psychometric)
library(nnet)
library(corrplot)
library(purrr)

full_dataset <- read.csv("InsuranceComplaints.csv", header = TRUE)
str(full_dataset)
head(full_dataset)

# Information on the full dataset
full_info <- str(full_dataset)

# Select variable columns
columns <- c(
  "Complaint.filed.against",
  "Complaint.number",
  "Respondent.type",
  "Confirmed.complaint",
  "Received.date",
  "Closed.date",
  "Coverage.type"
)
dataset <- full_dataset[columns]




##########################################**Data Conversions**
    ####################################################

# Convert date columns to Date type
```

```r
dataset$Received.date <- as.Date(dataset$Received.date, format = "%m/%d/%Y")
dataset$Closed.date <- as.Date(dataset$Closed.date, format = "%m/%d/%Y")

# Sort the data by 'Respondent.type' and 'Received.date'
dataset <- dataset %>%
  arrange(Respondent.type, Received.date)

# Calculate complaint duration in days
dataset <- dataset %>%
  mutate(Complaint.Duration = as.numeric(difftime(Closed.date, Received.date,
      units = "days")))

# Filter out rows with Complaint.Duration of zero
dataset <- dataset %>%
  filter(Complaint.Duration > 0)

# Remove complaints with 'Complaint.filed.against' that appears only once
dataset <- dataset %>%
  group_by(Complaint.filed.against) %>%
  filter(n() > 1) %>%
  ungroup()

# Summary statistics of the complaint duration
summary_stats <- summary(dataset$Complaint.Duration)
summary_stats

# New column for Duration.Class (categorical)
dataset <- dataset %>%
  mutate(Duration.Class = case_when(
    Complaint.Duration < 30 ~ "Short",
    Complaint.Duration >= 30 & Complaint.Duration <= 70 ~ "Moderate",
    Complaint.Duration > 70 & Complaint.Duration <= 119 ~ "Long",
    Complaint.Duration > 119 ~ "Extended"
  ))

# Calculating the 'Time.Between.Complaints' and 'Number.of.Previous.Complaints'
```

```
dataset <- dataset %>%
  group_by(Complaint.filed.against) %>%
  mutate(Time.Between.Complaints = as.numeric(difftime(Received.date, lag(Received
      .date), units = "days")),
         Number.of.Previous.Complaints = row_number() - 1) %>%
  ungroup()

# Set all null values in Time.Between.Complaints to zero
dataset <- dataset %>%
  mutate(Time.Between.Complaints = ifelse(is.na(Time.Between.Complaints), 0, Time.
      Between.Complaints))

# Filter out rows with Number.of.Previous.Complaints of zero
dataset <- dataset %>%
  filter(Number.of.Previous.Complaints > 0)

InsComp <- dataset %>%
  dplyr::select(Respondent.type, Confirmed.complaint, Coverage.type,
                Complaint.Duration, Duration.Class,
                Time.Between.Complaints, Number.of.Previous.Complaints, Complaint.
                    filed.against)

# Dataset after conversions
dataset_info <- str(InsComp)




##########################################**Category Filtering**
    ################################################

# Extract unique responses for each categorical column
unique_values <- InsComp %>%
  dplyr::select(Respondent.type, Confirmed.complaint, Coverage.type,
         Duration.Class) %>%
  map(~ unique(.x))
```

```
unique_values

# Count the occurrences of each unique response in the Complaint.filed.against
    column
against_counts <- InsComp %>%
  count(Complaint.filed.against, name = "Frequency")
against_counts

# Count the occurrences of each unique response in the Respondent.type column
respondent_type_counts <- InsComp %>%
  count(Respondent.type, name = "Frequency")
respondent_type_counts

# Count the occurrences of each unique response in the Confirmed.complaint column
confirmed_complaint_counts <- InsComp %>%
  count(Confirmed.complaint, name = "Frequency")
confirmed_complaint_counts

# Remove rows with "null" values ('')
InsComp_filtered <- InsComp %>%
  filter(!(Confirmed.complaint %in% c("")))

# Count the occurrences of each unique response in the Coverage.type column
coverage_type_counts <- InsComp %>%
  count(Coverage.type, name = "Frequency")
coverage_type_counts

# Remove rows with values 'Fire, Allied Lines & CMP', 'Homeowners', Liability',
    and 'Miscellaneous'
InsComp_filtered <- InsComp %>%
  filter(!(Coverage.type %in% c("Fire, Allied Lines & CMP", "Homeowners", "
      Liability", "Miscellaneous")))

# Filtered unique responses for each categorical column
filtered_unique <- InsComp_filtered %>%
  dplyr::select(Respondent.type, Confirmed.complaint, Coverage.type,
```

```
                Duration.Class) %>%
  map(~ unique(.x))
filtered_unique




########################################**Categorical Variable Factorization**
    ##################################

InsComp_filtered$Duration.Class <- as.numeric(factor(InsComp_filtered$Duration.
    Class,
                                                     levels = c("Short", "Moderate", "Long"
                                                         , "Extended")))

InsComp_final <- InsComp_filtered

# Factorizing 'Respondent.type'
InsComp_final <- InsComp_final %>%
  mutate(Resp.Individual = ifelse(Respondent.type == "Individual", 1, 0),
         Resp.Organization = ifelse(Respondent.type == "Organization", 1, 0))

# Factorizing 'Confirmed.complaint'
InsComp_final <- InsComp_final %>%
  mutate(Conf.Yes = ifelse(Confirmed.complaint == "Yes", 1, 0),
         Conf.No = ifelse(Confirmed.complaint == "No", 1, 0))

# Factorizing 'Coverage.type'
InsComp_final <- InsComp_final %>%
  mutate(Cov.AccidentHealth = ifelse(Coverage.type == "Accident␣and␣Health", 1, 0)
      ,
         Cov.Automobile = ifelse(Coverage.type == "Automobile", 1, 0),
         Cov.LifeAnnuity = ifelse(Coverage.type == "Life␣&␣Annuity", 1, 0))

# Set "Individual" as the complaint type reference
InsComp_final <- InsComp_final %>%
```

```
  dplyr::select(-Resp.Individual)

# Set "Yes" as the complaint type reference
InsComp_final <- InsComp_final %>%
  dplyr::select(-Conf.Yes)

# Set "Life & Annuity" as the complaint type reference
InsComp_final <- InsComp_final %>%
  dplyr::select(-Cov.LifeAnnuity)

str(InsComp_final)

# Take a random sample of 4% of entries from InsComp
set.seed(123)
InsComp_final <- InsComp_final %>% sample_frac(0.04)

# Information on the final dataset
final_info <- str(InsComp_final)
head(InsComp_final)

write.csv(InsComp_final, "InsComp_final.csv", row.names = FALSE)




##########################################**Regression Model**
    ##################################################
library(dplyr)
library(lubridate)
library(ggplot2)
library(broom)
library(psychometric)
library(nnet)
library(corrplot)
library(purrr)
library(xtable)
```

```
library(car)
library(psych)
library(lmtest)
library(lessR)
library(MASS)

InsComp_final <- read.csv("InsComp_final.csv", header = TRUE)
head(InsComp_final)
str(InsComp_final)

CD <- InsComp_final$Complaint.Duration
CD <- InsComp_final$Complaint.Duration
TbC <- InsComp_final$Time.Between.Complaints
TbC <- ifelse(TbC <= 0, TbC + .001, TbC) # Adding .001 in case of zero values
NPC <- InsComp_final$Number.of.Previous.Complaints
RO <- InsComp_final$Resp.Organization
CfN <- InsComp_final$Conf.No
CvAH <- InsComp_final$Cov.AccidentHealth
CvAu <- InsComp_final$Cov.Automobile

fit <- lm(CD ~ TbC + NPC + RO + CfN + CvAH + CvAu,
          data = InsComp_final)
summary(fit)

layout(matrix(c(1,2,3,4),2,2))
plot(fit)

# Box-Cox Transformation (BoxCox_Results.png)
boxcox_results <- boxcox(fit, lambda = seq(-2, 2, by = 0.1))
lambda_values <- boxcox_results$x
log_likelihoods <- boxcox_results$y
optimal_lambda <- lambda_values[which.max(log_likelihoods)]
cat("Optimal␣lambda␣value:", optimal_lambda, "\n") # \lamda = 0.1414141


# log-transformation of Complaint.Duration
```

```
# ln_CD <- CD^optimal_lambda
ln_CD <- log(CD)

# log-transformed Time.between.Complaints
ln_TbC <- log(TbC)

fit_log <- lm(ln_CD ~ ln_TbC + NPC + RO + CfN + CvAH + CvAu,
              data = InsComp_final)
summary(fit_log)

layout(matrix(c(1,2,3,4),2,2))
plot(fit_log)

# Test the form of the Model (Linear Form) (Model_Linearity.png)
fitted <- fitted(fit_log)
residuals <- residuals(fit_log)
# plot(fitted, residuals,
#      main = "Residuals vs. Fitted",
#      xlab = "Fitted values",
#      ylab = "Residuals")
# abline(h = 0, col = "red")
#
# # Test of Normality Assumption
# # (Model_Resid_Hist.png)
# hist(residuals, main = "Histogram of Residuals",
#      xlab = "Residuals", col = "orange")
#
# # (Model_QQ.png)
# qqnorm(residuals)
# qqline(residuals, col = "red", lwd = 2)
#
# # Test of Constant Variance Assumption (Model_Scale_Location.png)
# plot(fitted, sqrt(abs(residuals)),
#      main = "Scale-Location",
#      xlab = "Fitted values",
#      ylab = "Square Root of |Residuals|")
```

```
# abline(h = 0, col = "red")
#
# # Test of Outliers (Model_Resid_Boxplot.png)
# boxplot(residuals, main = "Boxplot of Residuals", col = "lightgreen", horizontal
    = TRUE)
# SummaryStats(residuals)

# Function to remove outliers using IQR
remove_outliers <- function(data, column) {
  Q1 <- quantile(data[[column]], 0.25, na.rm = TRUE)
  Q3 <- quantile(data[[column]], 0.75, na.rm = TRUE)
  IQR_value <- Q3 - Q1

  lower_bound <- Q1 - 1.5 * IQR_value
  upper_bound <- Q3 + 1.5 * IQR_value

  # Calculate the number of rows before filtering
  before_count <- nrow(data)

  # Filter the data
  filtered_data <- data %>%
    filter(data[[column]] >= lower_bound & data[[column]] <= upper_bound)

  # Calculate the number of rows after filtering
  after_count <- nrow(filtered_data)

  # Print the number of rows removed
  cat("Outliers removed for", column, ":", before_count - after_count, "\n")

  return(filtered_data)
}

# Remove outliers for quantitative variables
InsComp_final <- remove_outliers(InsComp_final, "Complaint.Duration") # 761
    outliers removed
InsComp_final <- remove_outliers(InsComp_final, "Time.Between.Complaints") # 11226
```

```
    outliers removed
InsComp_final <- remove_outliers(InsComp_final, "Number.of.Previous.Complaints") #
    755 outliers removed

# Test of Multicollinearity (CorrPlot.png)
numeric_data <- InsComp_final %>%
  dplyr::select_if(is.numeric) %>%
  dplyr::select(-Complaint.Duration, -Duration.Class)
corr_matrix <- cor(numeric_data)
cor(numeric_data)
corrplot(corr_matrix) # Disregard Cov.Automobile because of high correlation with
    Cov.AccidentHealth

# Test of Influential Observations
lev <- hatvalues(fit_log)
cooks_distance <- cooks.distance(fit_log)

# # (Model_Influence_Plot.png)
# influencePlot(fit_log)
#
# # (Model_Resid_Lev.png)
# plot(lev, residuals,
#      main = "Residuals vs. Leverage",
#      xlab = "Leverage",
#      ylab = "Residuals")
# abline(h = 0, col = "red")

cook_cutoff <- 4 / length(residuals)
# abline(v = cook_cutoff, col = "blue", lty = 2)


influential_points <- which(cooks_distance > cook_cutoff | lev > (2 * mean(lev)))

# Calculate the number of influential points
num_influential_points <- length(influential_points)
cat("Number␣of␣influential␣points:", num_influential_points, "\n") # 914
```

*Influential Points*

```
# Remove influential points
InsComp_cleaned <- InsComp_final[-influential_points, ]
str(InsComp_cleaned) # 4679 observations remaining

CD <- InsComp_cleaned$Complaint.Duration
ln_CD <- log(CD)
TbC <- InsComp_cleaned$Time.Between.Complaints
TbC <- ifelse(TbC <= 0, TbC + .001, TbC) # Adding .001 in case of zero values
ln_TbC <- log(TbC)
NPC <- InsComp_cleaned$Number.of.Previous.Complaints
RO <- InsComp_cleaned$Resp.Organization
CfN <- InsComp_cleaned$Conf.No
CvAH <- InsComp_cleaned$Cov.AccidentHealth

# Model with no categorical variables
fit_num_only <- lm(ln_CD ~ ln_TbC + NPC,
                       data = InsComp_cleaned)
summary(fit_num_only)

# Full Model
fit_log_cleaned <- lm(ln_CD ~ ln_TbC + NPC + RO + CfN + CvAH,
                       data = InsComp_cleaned)
summary(fit_log_cleaned)

confint(fit_log_cleaned)

numeric_data <- InsComp_cleaned %>%
  dplyr::select_if(is.numeric) %>%
  dplyr::select(-Complaint.Duration, -Duration.Class)

# Compute the correlation matrix
custom_abbreviations <- c("C.D." = "Complaint.Duration",
                          "T.B.C." = "Time.Between.Complaints",
                          "N.P.C." = "Number.of.Previous.Complaints",
```

```
                          "Resp. Org." = "Resp.Organization",
                          "Conf. No" = "Conf.No",
                          "Cov. A.H." = "Cov.AccidentHealth")

# Replace colnames and rownames with custom abbreviations
colnames(cor_matrix) <- names(custom_abbreviations)
rownames(cor_matrix) <- names(custom_abbreviations)

# Plot with custom abbreviations
corrplot(cor_matrix, method = "circle",
         tl.col = "black",
         tl.srt = 45)
```

```
fitted = fitted(fit_log_cleaned)
resid = residuals(fit_log_cleaned)
std_resid = rstandard(fit_log_cleaned)
leverage = hatvalues(fit_log_cleaned)
cooks_distance = cooks.distance(fit_log_cleaned)

library(ggplot2)
library(patchwork)

# Residuals vs Fitted (Model_ResidvFit.pdf)
plot1 <- ggplot(mapping = aes(x = fitted, y = resid)) +
  geom_point(color = "maroon") +
  geom_hline(yintercept = 0, color = "red", linetype = "dashed") +
  labs(x = "Fitted Values", y = "Residuals")
plot1

# Normal Q-Q (Model_QQ.pdf)
```

```
plot2 <- ggplot(mapping = aes(sample = std_resid)) +
  stat_qq(color = "chartreuse4") +
  stat_qq_line(color = "red") +
  labs(x = "Theoretical Quantiles", y = "Standardized Residuals")
plot2

# Scale-Location (Model_ScaleLocation.pdf)
plot3 <- ggplot(mapping = aes(x = fitted, y = sqrt(abs(std_resid)))) +
  geom_point(color = "darkorchid") +
  geom_hline(yintercept = 0, color = "red", linetype = "dashed") +
  labs(x = "Fitted Values", y = "Sqrt(|Standardized Residuals|)")
plot3

# Residuals vs Leverage (Model_ResidvLev.pdf)
plot4 <- ggplot(mapping = aes(x = leverage, y = std_resid)) +
  geom_point(color = "cadetblue") +
  geom_hline(yintercept = 0, color = "red", linetype = "dashed") +
  labs(x = "Leverage", y = "Standardized Residuals")
plot4

# Combine plots
combined_plot <- (plot1 | plot2) / (plot3 | plot4)
combined_plot # Full_DiagPlots.pdf




##########################################**Small & Large Organizations Regression
    Models**#######################

# Calculate the frequency of complaints filed against each entity
against_counts <- InsComp_cleaned %>%
  count(Complaint.filed.against, name = "Frequency")

# Merge the frequency back into the main dataset
InsComp_cleaned <- InsComp_cleaned %>%
```

```
  left_join(against_counts, by = "Complaint.filed.against")

# Categorize companies into Small, Medium, and Large
InsComp_cleaned <- InsComp_cleaned %>%
  mutate(
    Category = case_when(
      Frequency <= 10 ~ "Small",
      Frequency > 10 & Frequency <= 50 ~ "Medium",
      Frequency > 50 ~ "Large"
    )
  )

# Split the dataset into three groups
InsComp_small <- InsComp_cleaned %>% filter(Category == "Small")
InsComp_medium <- InsComp_cleaned %>% filter(Category == "Medium")
InsComp_large <- InsComp_cleaned %>% filter(Category == "Large")

# Verify the split
cat("Number␣of␣small␣organizations:", nrow(InsComp_small), "\n")
cat("Number␣of␣medium␣organizations:", nrow(InsComp_medium), "\n")
cat("Number␣of␣large␣organizations:", nrow(InsComp_large), "\n")

# # Save the datasets to CSV files
# write.csv(InsComp_small, "InsComp_small.csv", row.names = FALSE)
# write.csv(InsComp_medium, "InsComp_medium.csv", row.names = FALSE)
# write.csv(InsComp_large, "InsComp_large.csv", row.names = FALSE)

# Summary of counts in each category
summary_counts <- InsComp_cleaned %>%
  group_by(Category) %>%
  summarise(Total_Entities = n_distinct(Complaint.filed.against),
            Total_Complaints = n())

print(summary_counts)

##############Small Companies###################
```

```
InsComp_small <- remove_outliers(InsComp_small, "Complaint.Duration")
InsComp_small <- remove_outliers(InsComp_small, "Time.Between.Complaints")
InsComp_small <- remove_outliers(InsComp_small, "Number.of.Previous.Complaints")

CD <- InsComp_small$Complaint.Duration
ln_CD <- log(CD)
TbC <- InsComp_small$Time.Between.Complaints
TbC <- ifelse(TbC <= 0, TbC + .001, TbC) # Adding .001 in case of zero values
ln_TbC <- log(TbC)
NPC <- InsComp_small$Number.of.Previous.Complaints
RO <- InsComp_small$Resp.Organization
CfN <- InsComp_small$Conf.No
CvAH <- InsComp_small$Cov.AccidentHealth

fit_small <- lm(ln_CD ~ ln_TbC + NPC + RO + CfN + CvAH,
                data = InsComp_small)
summary(fit_small)
layout(matrix(c(1, 2, 3, 4), 2, 2)) # Small_DiagPlot.png
plot(fit_small)

###############Medium Companies###################

InsComp_medium <- remove_outliers(InsComp_medium, "Complaint.Duration")
InsComp_medium <- remove_outliers(InsComp_medium, "Time.Between.Complaints")
InsComp_medium <- remove_outliers(InsComp_medium, "Number.of.Previous.Complaints")

CD <- InsComp_medium$Complaint.Duration
ln_CD <- log(CD)
TbC <- InsComp_medium$Time.Between.Complaints
TbC <- ifelse(TbC <= 0, TbC + .001, TbC) # Adding .001 in case of zero values
ln_TbC <- log(TbC)
NPC <- InsComp_medium$Number.of.Previous.Complaints
RO <- InsComp_medium$Resp.Organization
CfN <- InsComp_medium$Conf.No
CvAH <- InsComp_medium$Cov.AccidentHealth
```

```
fit_medium <- lm(ln_CD ~ ln_TbC + NPC + RO + CfN + CvAH,
                 data = InsComp_medium)
summary(fit_medium)
layout(matrix(c(1, 2, 3, 4), 2, 2)) # Medium_DiagPlot.png
plot(fit_medium)

##############Large Companies###################

InsComp_large <- remove_outliers(InsComp_large, "Complaint.Duration")
InsComp_large <- remove_outliers(InsComp_large, "Time.Between.Complaints")
InsComp_large <- remove_outliers(InsComp_large, "Number.of.Previous.Complaints")

fit_large <- lm(ln_CD ~ ln_TbC + NPC + RO + CfN + CvAH + CvAu,
                 data = InsComp_large)
summary(fit_large)
layout(matrix(c(1, 2, 3, 4), 2, 2)) # Large_DiagPlot.png
plot(fit_large)


##########################################**Graphs & Summary Statistics**
    ########################################

# Histogram of Complaint Duration (CompDur_Histogram.pdf)
ggplot(InsComp_final, aes(x = Complaint.Duration)) +
  geom_histogram(bins = 30, color = "black", fill = "orange") +
  labs(x = "Complaint␣Duration␣(days)",
       y = "Frequency")

# Boxplot of Complaint Duration by Duration Class (CompDur_Boxplot.pdf)
ggplot(InsComp_final, aes(x = factor(Duration.Class,
                                     labels = c("Short␣\n(<30␣Days)",
                                                "Moderate␣\n(30-70
                                                    ␣Days)",
                                                "Long␣\n(71-119␣
                                                    Days)",
```

```
                                                        "Extended␣\n(>119␣
                                                            Days)")),
                            y = Complaint.Duration)) +
  geom_boxplot(fill = "lightblue", color = "black") +
  labs(x = "Duration␣Class",
       y = "Complaint␣Duration␣(days)")

# Bar Plot of Coverage Type Frequencies (CovType_BarPlot.png)
ggplot(InsComp_final, aes(x = Coverage.type)) +
  geom_bar(fill = "green", color = "black") +
  labs(title = "Frequency␣of␣Each␣Coverage␣Type",
       x = "Coverage␣Type",
       y = "Frequency") +
  theme_minimal()

# Scatter Plot of Complaint Duration vs. Time Between Complaints (CompDur_
    TimeBComp_Scatter.png)
ggplot(InsComp_final, aes(x = Time.Between.Complaints, y = Complaint.Duration)) +
  geom_point(color = "blue", alpha = 0.5) +
  labs(title = "Complaint␣Duration␣vs␣Time␣Between␣Complaints",
       x = "Time␣Between␣Complaints␣(days)",
       y = "Complaint␣Duration␣(days)") +
  theme_minimal()

# Residual Plot for Log-Transformed Regression Model (LogCompDur_ResPlot.png)
ggplot(fit_log, aes(.fitted, .resid)) +
  geom_point(alpha = 0.5, color = "maroon") +
  geom_hline(yintercept = 0, linetype = "dashed") +
  labs(title = "Residuals␣vs␣Fitted␣for␣Log-Transformed␣Model",
       x = "Fitted␣Values",
       y = "Residuals") +
  theme_minimal()

# Density Plot of Complaint Duration by Respondent Type (CompDur_ResType_Density.
    png)
ggplot(InsComp_final, aes(x = Complaint.Duration, fill = Respondent.type)) +
```

```
  geom_density(alpha = 0.6) +
  labs(title = "Density of Complaint Duration by Respondent Type",
       x = "Complaint Duration (days)",
       y = "Density") +
  theme_minimal()

# Heatmap of Complaint Counts by Respondent and Coverage Type (Cov_Res_Heatmap.png
    )
InsComp_final %>%
  count(Respondent.type, Coverage.type) %>%
  ggplot(aes(x = Respondent.type, y = Coverage.type, fill = n)) +
  geom_tile(color = "white") +
  scale_fill_gradient(low = "lightblue", high = "darkblue") +
  labs(title = "Heatmap of Complaint Counts by Respondent and Coverage Type",
       x = "Respondent Type",
       y = "Coverage Type",
       fill = "Count") +
  theme_minimal()

# Diagnostic plots (Diag_Plots.png)
layout(matrix(c(1,2,3,4),2,2))
plot(fit_log)

# Summary Statistics for Quantitative Variables
summary(InsComp_final$Complaint.Duration)
sd(InsComp_final$Complaint.Duration)
summary(InsComp_final$Time.Between.Complaints)
sd(InsComp_final$Time.Between.Complaints)
summary(InsComp_final$Number.of.Previous.Complaints)
sd(InsComp_final$Number.of.Previous.Complaints)
```