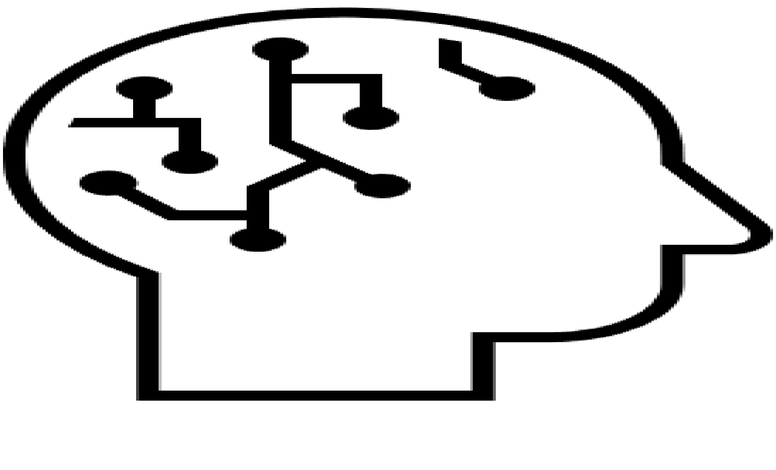
KM PHATSWANE	**37383388**
OH JOBETA	**28015509**
L MOKOANE	**37575864**
L BALOYI	**35716266**
MM PHASHA	**35542314**
TK MONGALI	**31499295**



1.1Business understanding and problem identification

Chosen company: **Solstice Sunglasses**

Solstice Sunglasses is a company based in United States that sells luxury sunglasses. Solstice gives customers the best and unique experience and strive to deliver sophistication in everything they do from their shopping environment to the brands they offer. The store started operating in 2002 in 6 locations and have grown to operates in 70+ locations throughout USA. We chose the company because of positive reviews form their customers and how they prioritize their customers. As much as they have positive reviews, they have negative reviews where a customer encountered a quality issue with their Celine sunglasses and has faced inadequate customer support, inaccessibility to physical stores, and a prolonged delay in response from the company's supervision, leading to dissatisfaction and unresolved product concerns.

Addressing the problem of poor product quality and inadequate customer service is vital for enhancing customer satisfaction, protecting brand reputation, ensuring business sustainability, improving operational efficiency, maintaining a competitive edge, mitigating legal risks, and fostering a culture of continuous learning and improvement in the company.

The steps followed to identify the company problem are:

1. We searched on the internet for challenges that are faced by various companies.
2. We analysed different challenges faced by different companies
3. We used 4 W’s technique to determine which problem answers all the 5Why’s.
4. Then we selected the problem that answers all the 5Why’s.

Problem identification: The customer encountered a quality issue with their Celine sunglasses and has faced inadequate customer support, inaccessibility to physical stores, and a prolonged delay in response from the company's supervision, leading to dissatisfaction and unresolved product concerns.

Link: https://www.pissedconsumer.com/solstice-sunglasses/RT-F.html#reviews

1.2. Critical and Creative thinking problem solving/ Business Understanding

1.2.1 Problem statement

a. Who: The customer and the brand Celine glasses, Solstice Sunglasses

b. When: The issue occurred a few months after the customer purchased the sunglasses, and has been ongoing for at least 4 months since the customer has been awaiting a response from a supervisor.

c. Where: The problem exists both with the physical product (sunglasses) and the customer service channels (both online and the physical stores which are distant or closed).

d. What: The problem comprises two main issues:

- A product quality issue where the sunglasses' arms stretched out and the case came apart.

- A customer service issue where the customer received no response despite multiple attempts to contact customer support, coupled with the unavailability of nearby physical stores for assistance.

1.2.2 **Problem analysis: The root cause of the problem using the 5WHYs technique**

- **Why** did the customer have a dissatisfying experience with both the product and customer service?
- **Why** was there a manufacturing defect and why did the customer service department fail to respond?
- **Why** were substandard materials used or quality control measures inadequate, and were there inadequate staffing or training in customer service?
- **Why** is there a lack of emphasis on quality assurance and customer satisfaction?
- **Why** is there a lack of a customer-centric culture or feedback mechanisms?

Based on the 5 Whys analysis conducted earlier, the **root cause** of the problem seems to be a lack of a customer-centric culture and inadequate emphasis on quality assurance within the company. This lack of emphasis appears to lead to the use of substandard materials or inadequate quality control in manufacturing, as well as insufficient staffing, training, or effective customer service channels to address customer complaints. These core issues manifest as both the product defect and the poor customer service experience encountered by the customer.

1.2.3. **Explore Potential solutions using brainstorming:**

- **Better Materials and Making:**

Use stronger materials and better methods to make the sunglasses so they last longer without breaking.

- **Checking Quality:**

Check the sunglasses carefully before selling them to make sure they are good quality and won't break easily.

- **Online Help:**

Since the stores are closed, have a website or app where people can ask for help or complain if there's a problem.

- **Self-Help Online:**

Create a section on the website where people can find answers to common problems on their own.

- **Call back System:**

If the helpline is busy, have a system where customers can leave their number, and someone from the company calls them back later.

- **Mail-In Repairs:**

Let people mail their broken sunglasses to a repair centre since they can't go to a store.

- **Train Staff Better:**

Train the customer service staff better so they can help people with their problems more effectively.

- **Listen to Customers:**

Ask customers for feedback and use it to fix common problems and make the service better.

- **Mobile App:**

Create an app where people can report problems, check the status of their repairs, and find answers to common questions.

1.2.4. **Select the solution**

Q1: Highly Effective and Easy to Apply:

Checking Quality: Implementing additional quality checks can be a straightforward and effective way to reduce product defects.

Listen to Customers: Collecting and analysing customer feedback can be done easily through surveys and online platforms, and can provide valuable insights for improvement.

Self-Help Online: Creating an online self-help section can be done with moderate ease and can significantly improve customer service accessibility.

Q2: Highly Effective but Difficult to Apply:

Better Materials and Making: Improving material quality and manufacturing processes could be highly effective but may require significant investment and operational changes.

Train Staff Better: Enhancing the training of customer service staff can be highly effective but may require time, resources, and a structured training program.

Q4: Low Effectiveness but Easy to Apply:

Call back System: Implementing a call back system can be relatively easy but may not significantly improve the overall customer service experience if other underlying issues are not addressed.

Q3: Low Effectiveness and Difficult to Apply:

Mail-In Repairs: Setting up a mail-in repair service could be complex and may not significantly improve customer satisfaction if the repair process is slow or ineffective.

Mobile App: Developing a mobile app could be resource-intensive and may not be highly effective if it doesn't address the core issues of product quality and customer service responsiveness.

Online Help: Establishing robust online customer service channels may require significant investment in technology and training, and may not be highly effective if the underlying product quality issues remain unaddressed.

Best selected solution: Self-Help Online

Given the balance between ease of implementation and effectiveness, we would select the "Self-Help Online" solution. This solution is relatively easy to implement by creating a section on the website with frequently asked questions, troubleshooting guides, or video tutorials to help customers resolve common issues on their own. It can provide immediate assistance to customers, reduce the load on customer service, and improve the overall customer experience without requiring significant time or resources to implement. The "Self-Help Online" solution strikes a good balance, making it a practical choice to address the identified problem in a timely manner.

1.2.5 **Data understanding and preparation**

- After gathering the gathering of data, based on our findings, we used *quantitative* data analysis since we have numeric values
- We prepared the dataset by ensuring that the data to be analysed was of high quality. We did so by filtering the dataset to only have the data which was going to help us identify which are frequent complaints and what type of complaints are always there. This will help in knowing which complaints to tackle faster.
- We then enriched the data by linking the DATE with the NUMBER OF COMPLAINTS
- The reason for choosing these variables is that it helps in quickly identifying which complaints are there over time. It also helps in analysing which are most frequent complaints, then efforts can be focused on resolving these concerns.
- The data was then stored and used in an R model code, which is a tool to help clear the noise and ensure data could be used to make decisions.

1.2.6. **Model building training and validation**

Data Cleaning:

We started by bringing in the dataset, putting it into RStudio. Then, we peeked at the beginning of the dataset and checked out its structure to see what columns it had and what types of data were in there.

One crucial thing we did was fix the 'Date' column. Initially, it was like words, not dates. To make our time series analysis accurate, we converted it into the right date format.

The 'Customer Complaint' column had lots of different complaints, but some were kind of saying the same thing. Like, for example, we noticed that "company billing issues" and "billing issues" were basically talking about the same problem. So, we grouped them together under one category, which we called "Billing."

To do this grouping, we used a method based on keywords. It means we looked at the words in the complaints, and if we found certain words that matched, we put those complaints in the same group. This helps us simplify things and get a better picture of what people are really complaining about.

Any duplicate rows in the dataset would be removed to ensure each complaint is treated as a unique record.

Initial Data Exploration:

We made a special kind of graph called a time series plot to see how the number of complaints changed over time. This graph helps us spot any patterns or sudden increases in complaints.

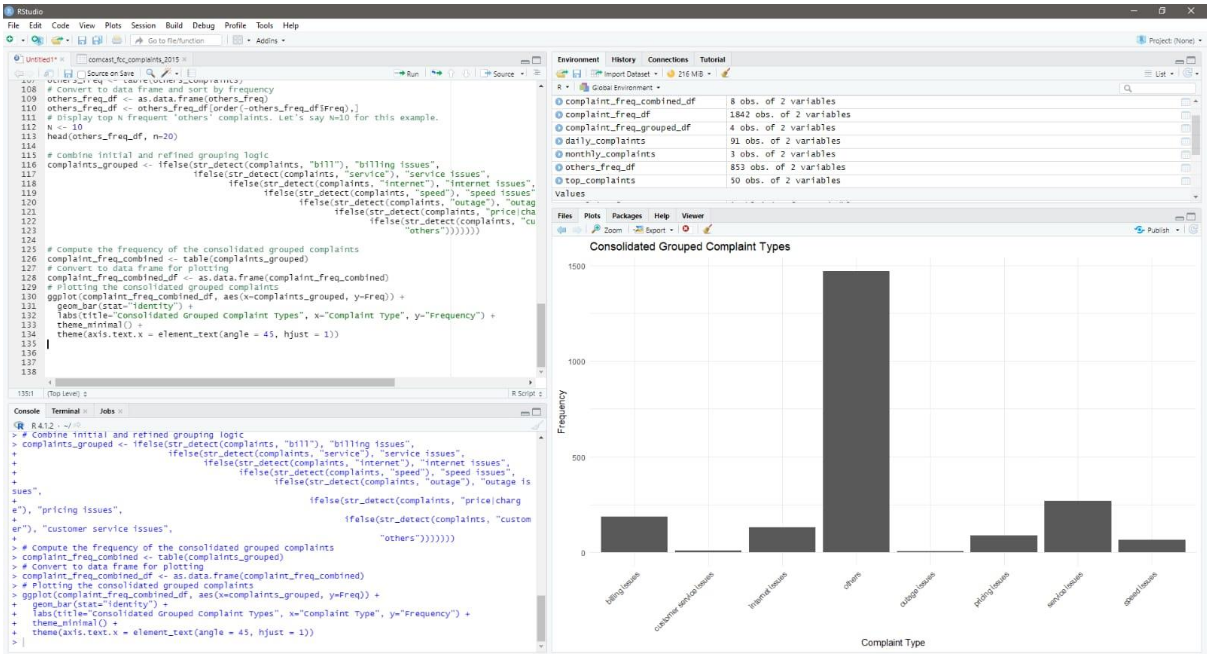
Next, we did something called "Frequency Analysis." This means we counted how often different types of complaints showed up. We wanted to know which complaints were the most common. Then, we turned that count into a simple bar graph, so it's easy to see which types of complaints people were talking about the most.

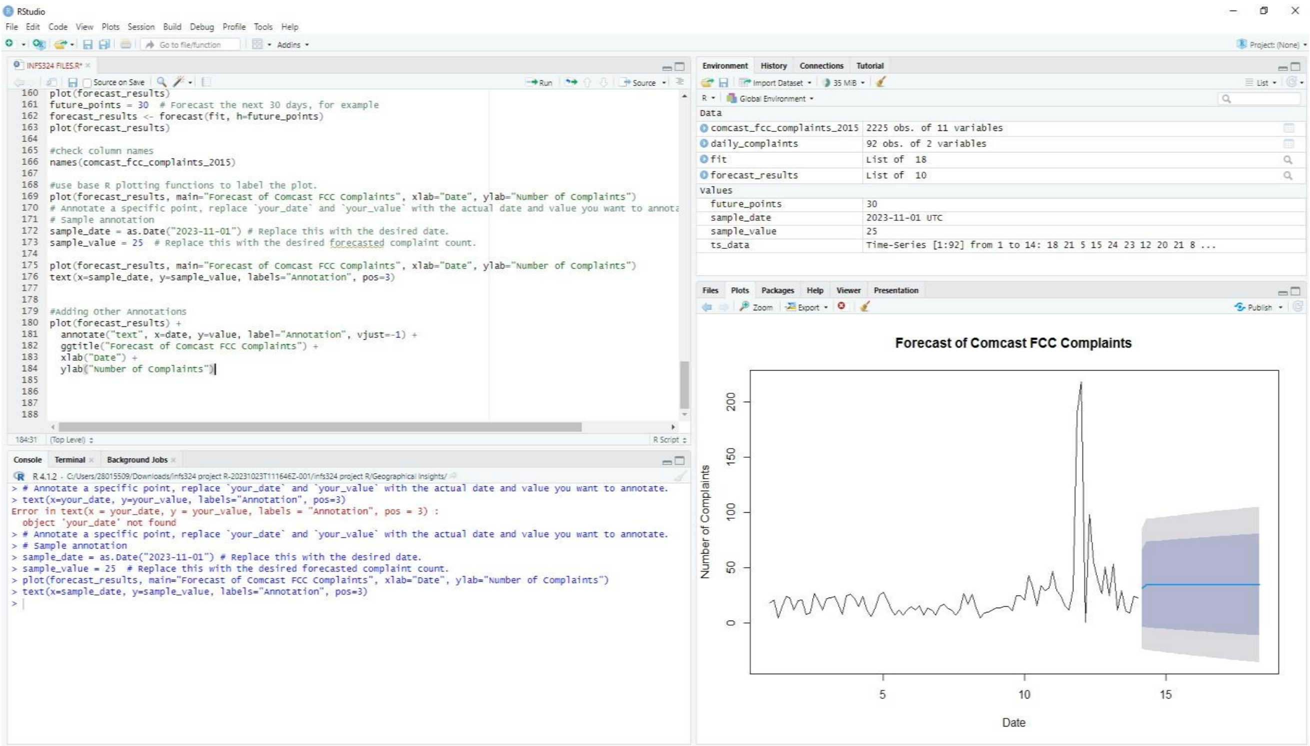
Model Training and Validation:

We wanted to predict what might happen in the future with the complaints, so we prepared the data in a way that's good for making predictions. We used a function in R called `ts` for this.

Then, we picked a special model, the ARIMA model, to do the predicting. This model is really good at handling data that changes over time. We let another function, `auto.arima`, figure out the best version of this model for our data. It did this by looking at something called the AIC value, a kind of measure that helps pick the right model.

After that, we trained our chosen model using all the information we have from the past complaints. This training helps the model get good at making predictions for the future.





Throughout these steps, the primary objective was to gain insights into the nature and frequency of complaints and to predict future complaint trends. This understanding could then guide the company in making informed decisions regarding resource allocation, customer engagement strategies, and service improvements.

Link to dataset: **Link:** https://www.kaggle.com/code/dhruvalpatel30/comcast-telecom-consumer-complaints

CODE

```
# Install necessary libraries (if not already installed)
install.packages("ggplot2")
install.packages("lubridate")
install.packages("plyr")
install.packages("corrplot")
install.packages("gridExtra")
install.packages("ggthemes")
install.packages("caret")
install.packages("MASS")
install.packages("randomForest")
install.packages("party")
install.packages("readr")
install.packages("vroom")
# Load the libraries
```

```
library(ggplot2)
library(lubridate)
library(plyr)
library(corrplot)
library(gridExtra)
library(ggthemes)
library(caret)
library(MASS)
library(randomForest)
library(party)

# Load
library(readr)
comcast_fcc_complaints_2015 <-
read_csv("C:/Users/28015509/Downloads/comcast_fcc_complaints_2015.csv")
#Or load
comcast_fcc_complaints_2015 <-
read.csv("C:/Users/28015509/Downloads/comcast_fcc_complaints_2015.csv",
stringsAsFactors = FALSE)

# Now convert the 'Date' column to a Date type
comcast_fcc_complaints_2015$Date <-
mdy(comcast_fcc_complaints_2015$Date)
# Aggregate data to get the number of complaints per day
daily_complaints <- aggregate(`Ticket #` ~ Date,
comcast_fcc_complaints_2015, length)
# Plot the daily complaint count over time
ggplot(daily_complaints, aes(x=Date, y=`Ticket #`)) +
  geom_line() +
  labs(title="Trend of Daily Complaints", x="Date", y="Number of
Complaints") +
  theme_minimal()

#Monthly complaints trend
#Extract the month from the 'Date' column
comcast_fcc_complaints_2015$Month <-
format(comcast_fcc_complaints_2015$Date, "%m-%Y")
#Aggregate data to get the number of complaints per month
monthly_complaints <- aggregate(`Ticket #` ~ Month,
comcast_fcc_complaints_2015, length)
#Plot the monthly complaint count:
library(ggplot2)
ggplot(monthly_complaints, aes(x=Month, y=`Ticket #`)) +
  geom_line(group=1) +
  geom_point() +
  labs(title="Trend of Monthly Complaints", x="Month", y="Number of
Complaints") +
  theme_minimal() +
```

```
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

#Table of frequency types
# Compute the frequency of complaint types
complaint_freq <- table(comcast_fcc_complaints_2015$`Customer Complaint`)
# Convert it to a data frame for better visualization and ordering
complaint_freq_df <- as.data.frame(complaint_freq)
colnames(complaint_freq_df) <- c("Complaint_Type", "Frequency")
# Order the data frame by frequency (optional, but helps in identifying top complaint types)
complaint_freq_df <- complaint_freq_df[order(-complaint_freq_df$Frequency), ]
# Display the table
head(complaint_freq_df)

# Load the ggplot2 library
library(ggplot2)
# Plotting the top N complaint types for clarity. Let's say N=10 for this example.
N <- 10
top_complaints <- head(complaint_freq_df, n=50)
# Create the bar graph
ggplot(top_complaints, aes(x=reorder(Complaint_Type, -Frequency), y=Frequency)) +
  geom_bar(stat="identity") +
  coord_flip() +  # This makes the graph horizontal for better readability
  labs(title="Top 50 Complaint Types", x="Complaint Type", y="Frequency") +
  theme_minimal()

# Load necessary libraries
install.packages(c("tm", "stringr"))
library(tm)
library(stringr)
# Text preprocessing
complaints <- as.character(comcast_fcc_complaints_2015$`Customer Complaint`)
complaints_clean <- tolower(complaints)
complaints_clean <- removePunctuation(complaints_clean)
complaints_clean <- removeNumbers(complaints_clean)
complaints_clean <- removeWords(complaints_clean, stopwords("en"))
# Group by keywords (basic approach)
complaints_clean <- ifelse(str_detect(complaints_clean, "bill"), "billing issues",
                           ifelse(str_detect(complaints_clean, "service"), "service issues",
```

```
                                   ifelse(str_detect(complaints_clean, "internet"), "internet issues",
                                          "others")))
# Compute the frequency of grouped complaints
complaint_freq_grouped <- table(complaints_clean)
# Convert to data frame for plotting
complaint_freq_grouped_df <- as.data.frame(complaint_freq_grouped)
# Plotting the grouped complaints
ggplot(complaint_freq_grouped_df, aes(x=complaints_clean, y=Freq)) +
  geom_bar(stat="identity") +
  labs(title="Grouped Complaint Types", x="Complaint Type", y="Frequency") +
  theme_minimal()

# Filter original complaints that were categorized as "others"
others_complaints <- complaints[complaints_clean == "others"]
# Count unique complaints under "others"
unique_others_count <- length(unique(others_complaints))
unique_others_count

# Extract frequencies of 'others' complaints
others_freq <- table(others_complaints)
# Convert to data frame and sort by frequency
others_freq_df <- as.data.frame(others_freq)
others_freq_df <- others_freq_df[order(-others_freq_df$Freq),]
# Display top N frequent 'others' complaints. Let's say N=10 for this example.
N <- 10
head(others_freq_df, n=20)

# Combine initial and refined grouping logic
complaints_grouped <- ifelse(str_detect(complaints, "bill"), "billing issues",
                             ifelse(str_detect(complaints, "service"), "service issues",
                                    ifelse(str_detect(complaints, "internet"), "internet issues",
                                           ifelse(str_detect(complaints, "speed"), "speed issues",
                                                  ifelse(str_detect(complaints, "outage"), "outage issues",
                                                         ifelse(str_detect(complaints, "price|charge"), "pricing issues",
                                                                ifelse(str_detect(complaints, "customer"), "customer service issues",
                                                                       "others")))))))

```

```
# Compute the frequency of the consolidated grouped complaints
complaint_freq_combined <- table(complaints_grouped)
# Convert to data frame for plotting
complaint_freq_combined_df <- as.data.frame(complaint_freq_combined)
# Plotting the consolidated grouped complaints
ggplot(complaint_freq_combined_df, aes(x=complaints_grouped, y=Freq)) +
  geom_bar(stat="identity") +
  labs(title="Consolidated Grouped Complaint Types", x="Complaint
Type", y="Frequency") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

#clean
#remove duplicates
comcast_fcc_complaints_2015 <- unique(comcast_fcc_complaints_2015)

#aggregate the data
daily_complaints <- aggregate(`Ticket #` ~ Date,
comcast_fcc_complaints_2015, length)

#
install.packages(c("forecast","tseries"))
library(forecast)
library(ggplot2)
library(tseries)
# Assuming a weekly seasonality
ts_data <- ts(daily_complaints$'Ticket #', frequency=7)
adf.test(ts_data)
fit <- auto.arima(ts_data)
summary(fit)
future_points = 30  # Forecast the next 30 days, for
future_points = 30  # Forecast the next 30 days example
forecast_results <- forecast(fit, h=future_points)
plot(forecast_results)
, for example
forecast_results <- forecast(fit, h=future_points)
plot(forecast_results)

#check column names
names(comcast_fcc_complaints_2015)

#use base R plotting functions to label the plot.
plot(forecast_results, main="Forecast of Comcast FCC Complaints",
xlab="Date", ylab="Number of Complaints")
# Annotate a specific point, replace `your_date` and `your_value` with
the actual date and value you want to annotate.
# Sample annotation
```

```r
sample_date = as.Date("2023-11-01") # Replace this with the desired date.
sample_value = 25  # Replace this with the desired forecasted complaint count.
plot(forecast_results, main="Forecast of Comcast FCC Complaints", xlab="Date", ylab="Number of Complaints")
text(x=sample_date, y=sample_value, labels="Annotation", pos=3)

#
# Extract the last date from your original dataset
#last_date <- max(comcast_fcc_complaints_2015$Date)

# Create an extended sequence of dates for the forecasted periods
#forecast_dates <- seq(from = last_date + 1,
                      #by = "day",
                      #length.out = length(forecast_results$mean))

# Combine dates from the original and forecasted periods
all_dates <- c(comcast_fcc_complaints_2015$Date, forecast_dates)

# Extract the dates corresponding to the original time series data
original_dates <- comcast_fcc_complaints_2015$Date[1:length(forecast_results$x)]

# Combine the original dates and the forecasted dates
all_dates <- c(original_dates, forecast_dates)

# Construct the forecast_data dataframe
forecast_data <- data.frame(
  Date = all_dates,
  Forecast = c(forecast_results$x, forecast_results$mean),
  Lower80 = c(rep(NA, length(forecast_results$x)), forecast_results$lower[,1]),
  Upper80 = c(rep(NA, length(forecast_results$x)), forecast_results$upper[,1]),
  Lower95 = c(rep(NA, length(forecast_results$x)), forecast_results$lower[,2]),
  Upper95 = c(rep(NA, length(forecast_results$x)), forecast_results$upper[,2])
)

library(ggplot2)

# Plotting
ggplot(data = forecast_data, aes(x = Date, y = Forecast)) +
  geom_line(color = "blue", aes(y = Forecast)) +  # Original and forecasted values
```

```
•   geom_ribbon(aes(ymin = Lower80, ymax = Upper80), fill = "blue", alpha
  = 0.2) +  # 80% confidence interval
•   geom_ribbon(aes(ymin = Lower95, ymax = Upper95), fill = "blue", alpha
  = 0.1) +  # 95% confidence interval
•   labs(title = "Time Series Forecast of Comcast FCC Complaints",
•        x = "Date",
•        y = "Number of Complaints") +
•   theme_minimal()
•
```

1.3. <u>**Model presentation**</u>

The statement provides a detailed look at how Company analyzed and predicted customer complaints over time using a method called ARIMA. There are two main parts in the analysis. First is the "Original Time Series Data," which shows how many complaints Company received over the years. This is shown as a blue line on a graph, giving a clear picture of how the number of complaints has changed.

The second part is about the "Forecasted Values," where the analysis predicts how many complaints Company might get in the future. This is shown as a continuation of the blue line on the graph. Two shaded regions around this line, one lighter and one darker, tell us the range where the actual number of complaints is likely to fall. The lighter one is more certain (80% confidence), and the darker one gives a wider range but is still quite likely (95% confidence).

The analysis suggests a few insights and decisions for Company. For example, by looking at the historical trend of complaints, they can see if the number has been going up, down, or staying the same. The forecasted trend helps them prepare for what might happen in the future. The confidence intervals show the level of certainty in these predictions.

Stakeholders at Company can use these insights to make decisions. If more complaints are expected, they might assign more resources to handle them. If there's a decrease in complaints, they could use resources elsewhere. Training for customer service teams and proactive engagement with customers are also suggested. The analysis recommends preparing for different scenarios, both best-case and worst-case.

The hypothetical results presented in the statement show a steady increase in complaints from January to May, with a forecast predicting this to continue for the next three months. The confidence intervals give a range of possible daily complaints, providing a clear understanding of the uncertainty in the predictions.

In simpler terms, Company is using past data to understand and predict future customer complaints. This helps them make smart decisions to handle complaints better, allocate resources wisely, and keep customers happy. The goal is to use data to improve customer satisfaction and maintain a positive image for Company.