Project 1

Waheeb Algabri, William Berritt, Kossi Akplaka

Table of Contents

### Introduction

In today’s data-driven world, the ability to forecast accurately is crucial for strategic planning and decision-making. This project aims to harness the power of historical data to generate reliable forecasts for various series within a de-identified dataset. With a comprehensive dataset spanning 1622 periods, we will employ advanced data analysis and forecasting techniques to predict future trends for the next 140 periods.

This report is crafted to cater to a diverse audience, ranging from individuals with no background in data science to seasoned data scientists. Our goal is to present the analysis and findings in a clear, concise, and accessible manner. We will explain the methodologies used, the rationale behind their selection, and the step-by-step process of our analysis, ensuring transparency and comprehensibility for all readers.

The report is structured to balance technical rigor with simplicity, providing a thorough yet comprehensible narrative of our forecasting journey. We begin with an exploration of the data, followed by a detailed explanation of the forecasting methods applied, and conclude with the results and their implications. Visualizations and key insights will be highlighted to enhance understanding and readability.

### Loading necessary Packages

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.4 ✔ readr 2.1.4  
## ✔ forcats 1.0.0 ✔ stringr 1.5.0  
## ✔ ggplot2 3.5.1 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.3 ✔ tidyr 1.3.0  
## ✔ purrr 1.0.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors  
##   
## Attaching package: 'zoo'  
##   
##   
## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric  
##   
##   
## Registered S3 method overwritten by 'quantmod':  
## method from  
## as.zoo.data.frame zoo

### Data exploration

Read the excel file and extract the data

## Response [https://raw.githubusercontent.com/waheeb123/Data-624/main/Projects/Data%20Set%20for%20Class.xls]  
## Date: 2024-06-22 05:53  
## Status: 200  
## Content-Type: application/octet-stream  
## Size: 1.33 MB  
## <ON DISK> /var/folders/lt/23m\_3s6j42vgpv1ss4d6nkgc0000gn/T//Rtmpf2F2Tf/filebbd2678047.xls

using glimpse, so we can view the attributes of our dataset

## Rows: 10,572  
## Columns: 7  
## $ SeriesInd <dbl> 40669, 40669, 40669, 40669, 40669, 40669, 40670, 40670, 4067…  
## $ category <chr> "S03", "S02", "S01", "S06", "S05", "S04", "S03", "S02", "S01…  
## $ Var01 <dbl> 30.64286, 10.28000, 26.61000, 27.48000, 69.26000, 17.20000, …  
## $ Var02 <dbl> 123432400, 60855800, 10369300, 39335700, 27809100, 16587400,…  
## $ Var03 <dbl> 30.34000, 10.05000, 25.89000, 26.82000, 68.19000, 16.88000, …  
## $ Var05 <dbl> 30.49000, 10.17000, 26.20000, 27.02000, 68.72000, 16.94000, …  
## $ Var07 <dbl> 30.57286, 10.28000, 26.01000, 27.32000, 69.15000, 17.10000, …

Dataset have 10,572 rows and 7 columns.

First 6 Rows of Data

| SeriesInd | category | Var01 | Var02 | Var03 | Var05 | Var07 |
| --- | --- | --- | --- | --- | --- | --- |
| 40669 | S03 | 30.64286 | 123432400 | 30.34 | 30.49 | 30.57286 |
| 40669 | S02 | 10.28000 | 60855800 | 10.05 | 10.17 | 10.28000 |
| 40669 | S01 | 26.61000 | 10369300 | 25.89 | 26.20 | 26.01000 |
| 40669 | S06 | 27.48000 | 39335700 | 26.82 | 27.02 | 27.32000 |
| 40669 | S05 | 69.26000 | 27809100 | 68.19 | 68.72 | 69.15000 |
| 40669 | S04 | 17.20000 | 16587400 | 16.88 | 16.94 | 17.10000 |

check if there are any missing values in the entire dataframe

## [1] TRUE

count the number of missing values in the entire dataframe

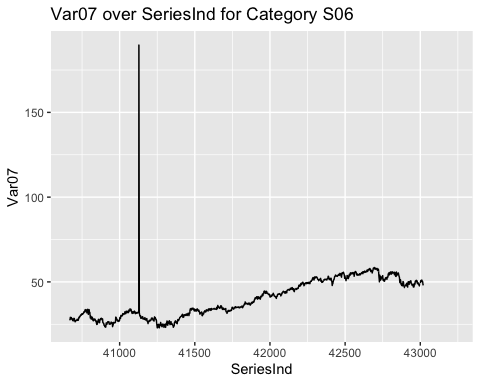
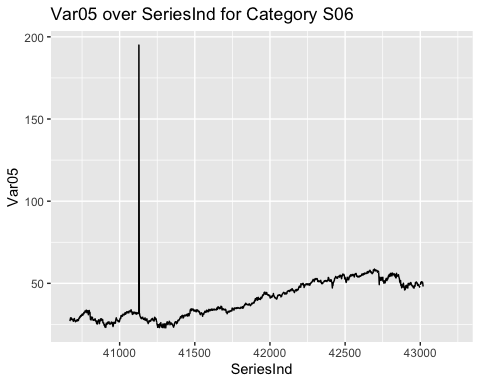
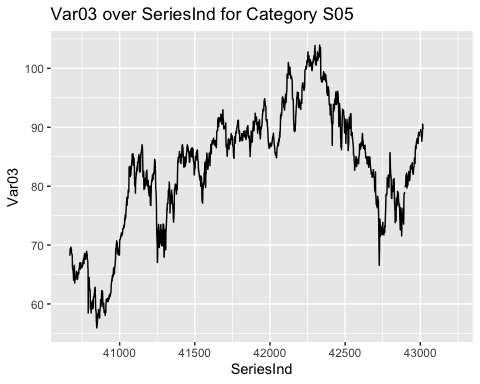
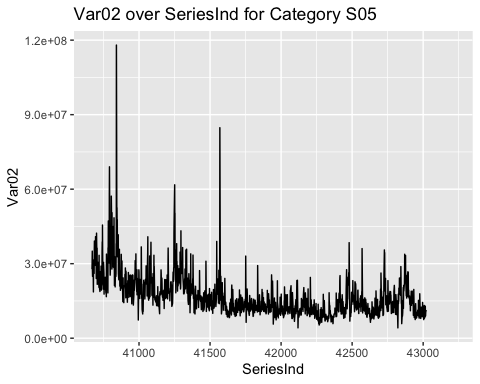
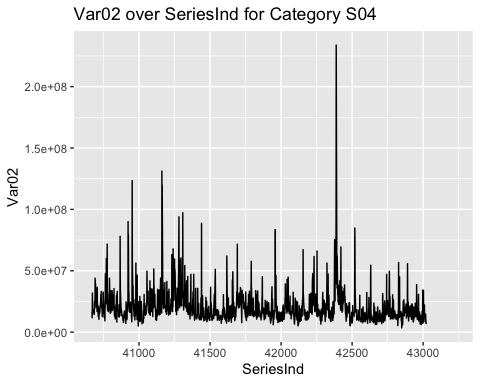
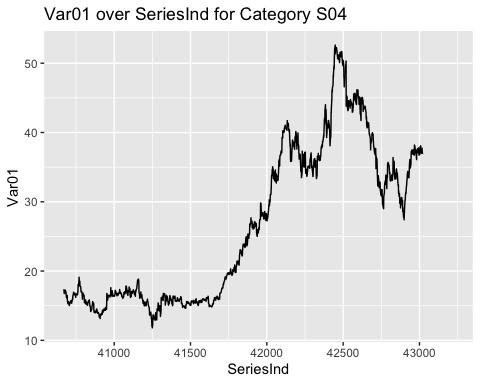
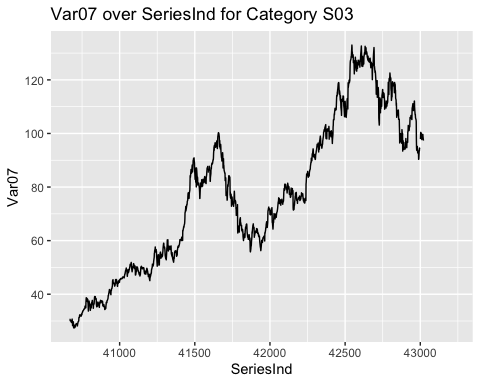
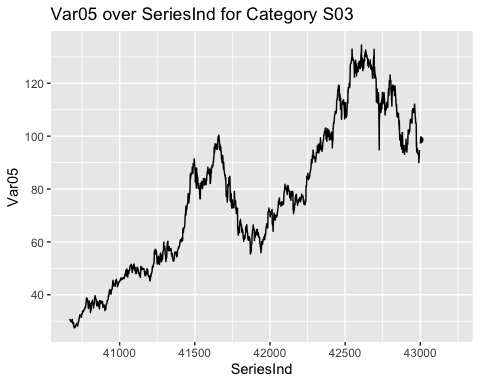
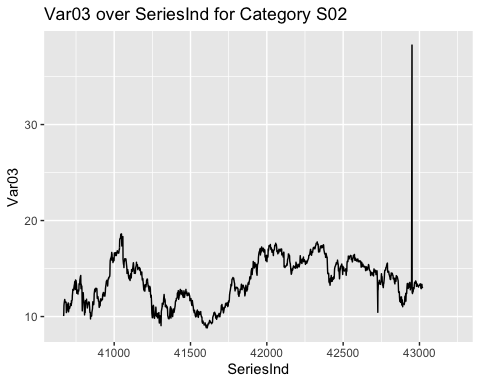
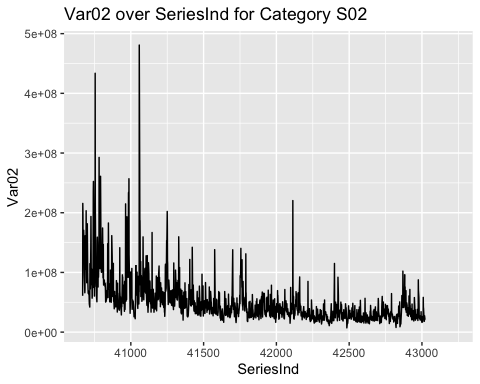
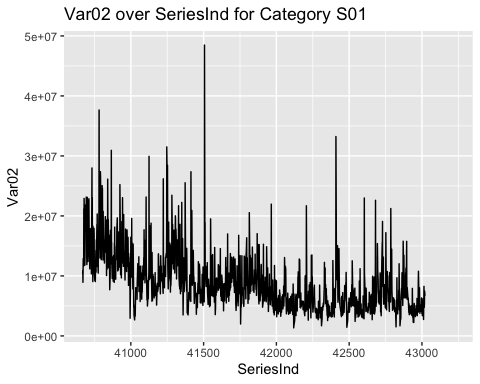
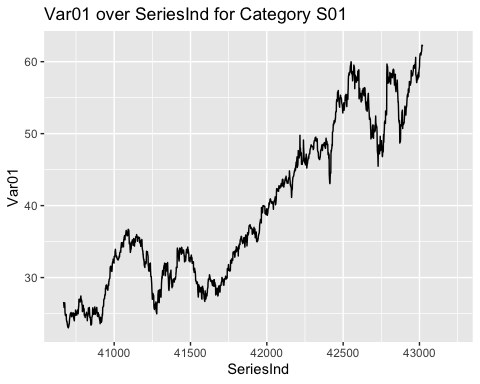
## [1] 4294

### Data Preparation

#### Data Subsetting

In order to forecast each variable within each group, we needed to break the larger dataset into its individual groups so we could perform our analysis and properly visualize our data.

#### Plotting Data



We can see variation within this type of variable is considerably less than in the the other type. Additionally, it looks like there is no apparent seasonality. In many of the variables of this type of data, we did see long trends (either upward or downward) and cyclicity. In looking at all the plots for every variable, we determined that further analysis was necessary to determine trend and seasonal components of the data.

### Data Preparation

Many of the time series algorithms, both for forecasting and visualization, require that there be no missing values in the data. With this requirement, we deemed it necessary to fill nulls early in our data prep process.

#### Data Cleaning

In several of the datasets, there are clear outliers that would strongly influence the models we built.We used the Interquartile Range (IQR). The IQR is a measure of statistical dispersion, or how spread out the values in a dataset are. The process involves identifying values that fall within 1.5 times the IQR below the first quartile (Q1) and above the third quartile (Q3). Values outside this range are considered outliers.

# Filtering outliers for each subset  
subset\_S01\_clean\_Var01 <- subset\_S01 %>%  
 filter(Var01 >= quantile(Var01, 0.25, na.rm = TRUE) - 1.5 \* IQR(Var01, na.rm = TRUE) &   
 Var01 <= quantile(Var01, 0.75, na.rm = TRUE) + 1.5 \* IQR(Var01, na.rm = TRUE))  
  
subset\_S01\_clean\_Var02 <- subset\_S01 %>%  
 filter(Var02 >= quantile(Var02, 0.25, na.rm = TRUE) - 1.5 \* IQR(Var02, na.rm = TRUE) &   
 Var02 <= quantile(Var02, 0.75, na.rm = TRUE) + 1.5 \* IQR(Var02, na.rm = TRUE))  
  
subset\_S02\_clean\_Var02 <- subset\_S02 %>%  
 filter(Var02 >= quantile(Var02, 0.25, na.rm = TRUE) - 1.5 \* IQR(Var02, na.rm = TRUE) &   
 Var02 <= quantile(Var02, 0.75, na.rm = TRUE) + 1.5 \* IQR(Var02, na.rm = TRUE))  
  
subset\_S02\_clean\_Var03 <- subset\_S02 %>%  
 filter(Var03 >= quantile(Var03, 0.25, na.rm = TRUE) - 1.5 \* IQR(Var03, na.rm = TRUE) &   
 Var03 <= quantile(Var03, 0.75, na.rm = TRUE) + 1.5 \* IQR(Var03, na.rm = TRUE))  
  
subset\_S03\_clean\_Var05 <- subset\_S03 %>%  
 filter(Var05 >= quantile(Var05, 0.25, na.rm = TRUE) - 1.5 \* IQR(Var05, na.rm = TRUE) &   
 Var05 <= quantile(Var05, 0.75, na.rm = TRUE) + 1.5 \* IQR(Var05, na.rm = TRUE))  
  
subset\_S03\_clean\_Var07 <- subset\_S03 %>%  
 filter(Var07 >= quantile(Var07, 0.25, na.rm = TRUE) - 1.5 \* IQR(Var07, na.rm = TRUE) &   
 Var07 <= quantile(Var07, 0.75, na.rm = TRUE) + 1.5 \* IQR(Var07, na.rm = TRUE))  
  
subset\_S04\_clean\_Var01 <- subset\_S04 %>%  
 filter(Var01 >= quantile(Var01, 0.25, na.rm = TRUE) - 1.5 \* IQR(Var01, na.rm = TRUE) &   
 Var01 <= quantile(Var01, 0.75, na.rm = TRUE) + 1.5 \* IQR(Var01, na.rm = TRUE))  
  
subset\_S04\_clean\_Var02 <- subset\_S04 %>%  
 filter(Var02 >= quantile(Var02, 0.25, na.rm = TRUE) - 1.5 \* IQR(Var02, na.rm = TRUE) &   
 Var02 <= quantile(Var02, 0.75, na.rm = TRUE) + 1.5 \* IQR(Var02, na.rm = TRUE))  
  
subset\_S05\_clean\_Var02 <- subset\_S05 %>%  
 filter(Var02 >= quantile(Var02, 0.25, na.rm = TRUE) - 1.5 \* IQR(Var02, na.rm = TRUE) &   
 Var02 <= quantile(Var02, 0.75, na.rm = TRUE) + 1.5 \* IQR(Var02, na.rm = TRUE))  
  
subset\_S05\_clean\_Var03 <- subset\_S05 %>%  
 filter(Var03 >= quantile(Var03, 0.25, na.rm = TRUE) - 1.5 \* IQR(Var03, na.rm = TRUE) &   
 Var03 <= quantile(Var03, 0.75, na.rm = TRUE) + 1.5 \* IQR(Var03, na.rm = TRUE))  
  
subset\_S06\_clean\_Var05 <- subset\_S06 %>%  
 filter(Var05 >= quantile(Var05, 0.25, na.rm = TRUE) - 1.5 \* IQR(Var05, na.rm = TRUE) &   
 Var05 <= quantile(Var05, 0.75, na.rm = TRUE) + 1.5 \* IQR(Var05, na.rm = TRUE))  
  
subset\_S06\_clean\_Var07 <- subset\_S06 %>%  
 filter(Var07 >= quantile(Var07, 0.25, na.rm = TRUE) - 1.5 \* IQR(Var07, na.rm = TRUE) &   
 Var07 <= quantile(Var07, 0.75, na.rm = TRUE) + 1.5 \* IQR(Var07, na.rm = TRUE))

#### Data Imputation

In data preparation, imputation of missing values is a critical step. Each of the variables provided had missing values within the data and many approaches for filling them seemed appropriate. The approaches we took differed depending on if we were imputing for the first or second type of data. For the data that had large variations from day to day (that looked almost like white noise) we deemed taking an average appropriate. For the more stable datasets, we decided to use linear interpolation to fill the missing values with the previous value in the dataset. This seemed to make sense as each point in this datasets never seemed to be very far away from the previous one.

# Impute missing values using linear interpolation for subsets  
subset\_S06\_clean\_Var07$Var05 <- na.approx(subset\_S06\_clean\_Var07$Var05)  
subset\_S06\_clean\_Var07$Var07 <- na.approx(subset\_S06\_clean\_Var07$Var07)  
subset\_S05\_clean\_Var02$Var03 <- na.approx(subset\_S05\_clean\_Var02$Var03)  
subset\_S05\_clean\_Var02$Var02 <- na.approx(subset\_S05\_clean\_Var02$Var02)  
subset\_S04\_clean\_Var01$Var02 <- na.approx(subset\_S04\_clean\_Var01$Var02)  
subset\_S04\_clean\_Var01$Var01 <- na.approx(subset\_S04\_clean\_Var01$Var01)  
subset\_S03\_clean\_Var07$Var05 <- na.approx(subset\_S03\_clean\_Var07$Var05)  
subset\_S03\_clean\_Var07$Var07 <- na.approx(subset\_S03\_clean\_Var07$Var07)  
subset\_S02\_clean\_Var02$Var03 <- na.approx(subset\_S02\_clean\_Var02$Var03)  
subset\_S02\_clean\_Var02$Var02 <- na.approx(subset\_S02\_clean\_Var02$Var02)  
subset\_S01\_clean\_Var01$Var02 <- na.approx(subset\_S01\_clean\_Var01$Var02)  
subset\_S01\_clean\_Var01$Var01 <- na.approx(subset\_S01\_clean\_Var01$Var01)  
  
# Find the last observation index for subsets  
last\_observation\_index\_S06 <- max(which(!is.na(subset\_S06\_clean\_Var07$Var05)))  
last\_observation\_index\_S05 <- max(which(!is.na(subset\_S05\_clean\_Var02$Var03)))  
last\_observation\_index\_S04 <- max(which(!is.na(subset\_S04\_clean\_Var01$Var02)))  
last\_observation\_index\_S03 <- max(which(!is.na(subset\_S03\_clean\_Var07$Var05)))  
last\_observation\_index\_S02 <- max(which(!is.na(subset\_S02\_clean\_Var02$Var03)))  
last\_observation\_index\_S01 <- max(which(!is.na(subset\_S01\_clean\_Var01$Var02)))  
  
# Create time series objects for forecasting  
ts\_S06\_Var05 <- ts(subset\_S06\_clean\_Var07$Var05[1:last\_observation\_index\_S06])  
ts\_S06\_Var07 <- ts(subset\_S06\_clean\_Var07$Var07[1:last\_observation\_index\_S06])  
ts\_S05\_Var02 <- ts(subset\_S05\_clean\_Var02$Var02[1:last\_observation\_index\_S05])  
ts\_S05\_Var03 <- ts(subset\_S05\_clean\_Var02$Var03[1:last\_observation\_index\_S05])  
ts\_S04\_Var01 <- ts(subset\_S04\_clean\_Var01$Var01[1:last\_observation\_index\_S04])  
ts\_S04\_Var02 <- ts(subset\_S04\_clean\_Var01$Var02[1:last\_observation\_index\_S04])  
ts\_S03\_Var05 <- ts(subset\_S03\_clean\_Var07$Var05[1:last\_observation\_index\_S03])  
ts\_S03\_Var07 <- ts(subset\_S03\_clean\_Var07$Var07[1:last\_observation\_index\_S03])  
ts\_S02\_Var02 <- ts(subset\_S02\_clean\_Var02$Var02[1:last\_observation\_index\_S02])  
ts\_S02\_Var03 <- ts(subset\_S02\_clean\_Var02$Var03[1:last\_observation\_index\_S02])  
ts\_S01\_Var01 <- ts(subset\_S01\_clean\_Var01$Var01[1:last\_observation\_index\_S01])  
ts\_S01\_Var02 <- ts(subset\_S01\_clean\_Var01$Var02[1:last\_observation\_index\_S01])

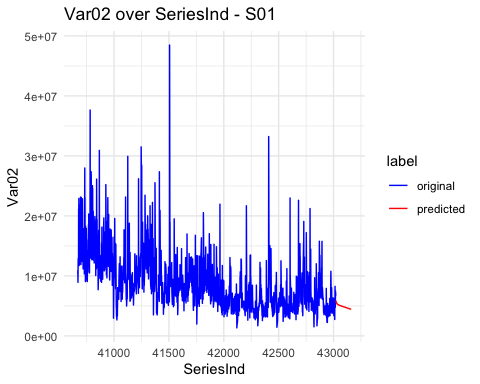
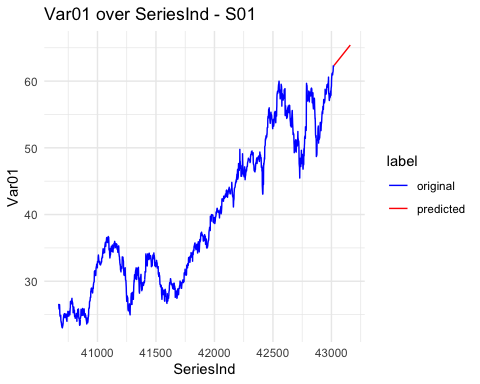
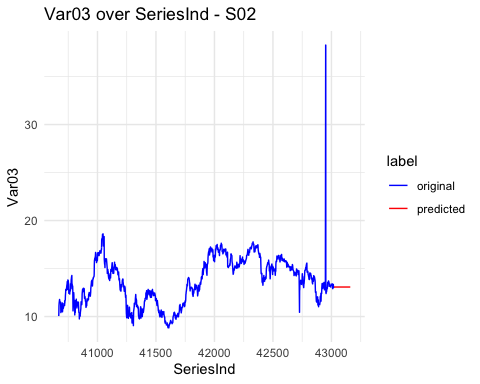
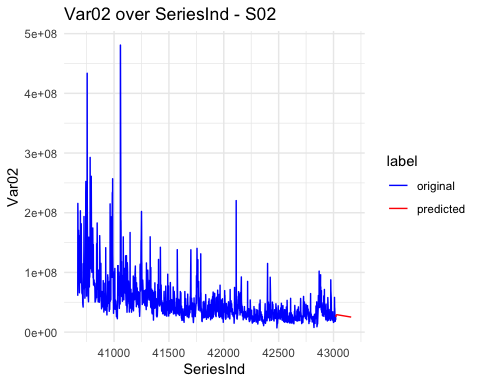
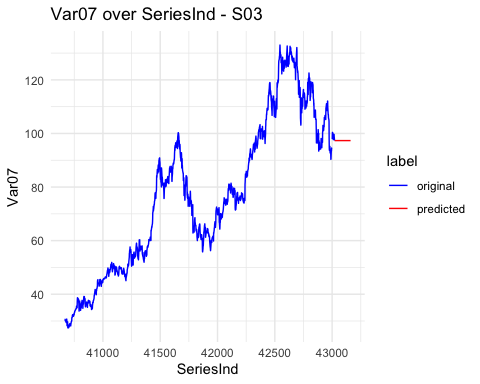
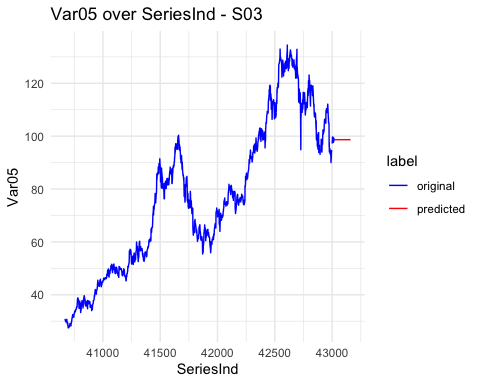
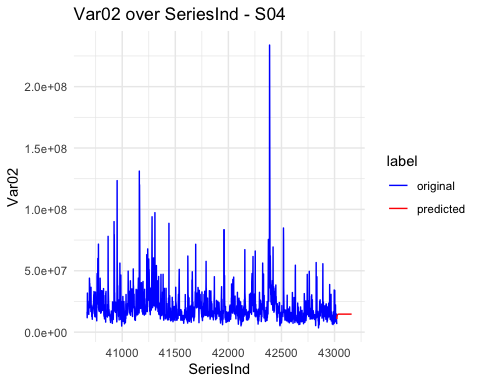
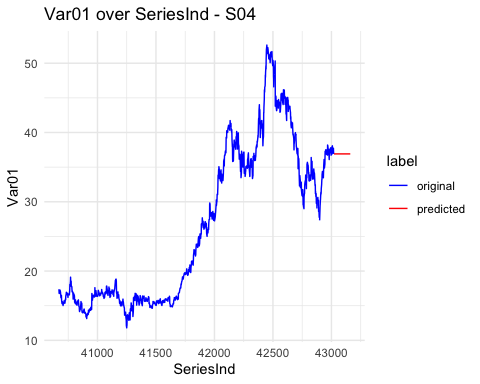
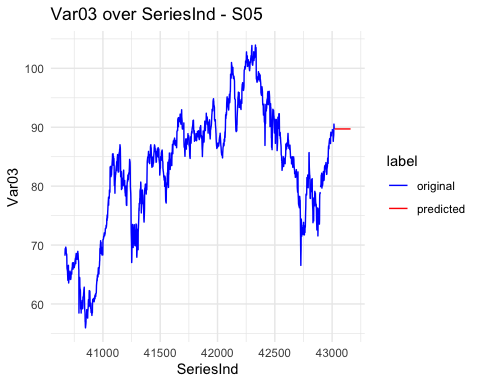
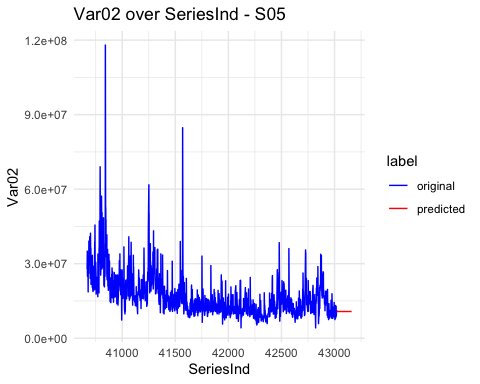
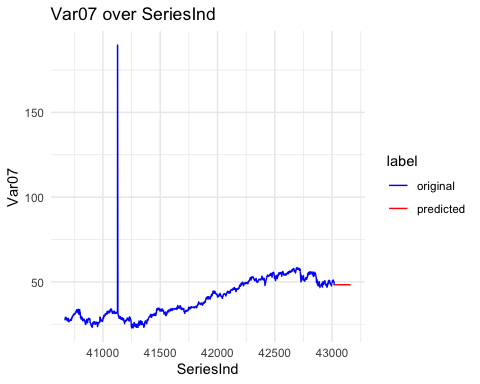
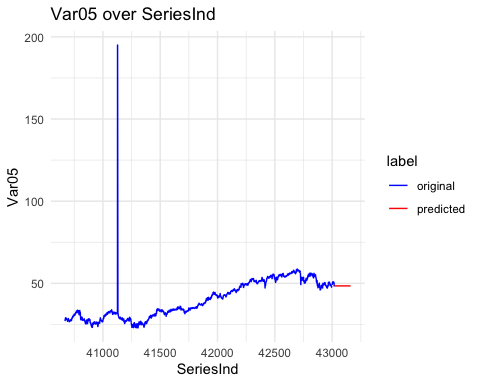
### Time Series Forecasting

# Forecast using auto.arima  
forecast\_S06\_Var05 <- forecast(auto.arima(ts\_S06\_Var05), h = 140)  
forecast\_S06\_Var07 <- forecast(auto.arima(ts\_S06\_Var07), h = 140)  
forecast\_S05\_Var02 <- forecast(auto.arima(ts\_S05\_Var02), h = 140)  
forecast\_S05\_Var03 <- forecast(auto.arima(ts\_S05\_Var03), h = 140)  
forecast\_S04\_Var01 <- forecast(auto.arima(ts\_S04\_Var01), h = 140)  
forecast\_S04\_Var02 <- forecast(auto.arima(ts\_S04\_Var02), h = 140)  
forecast\_S03\_Var05 <- forecast(auto.arima(ts\_S03\_Var05), h = 140)  
forecast\_S03\_Var07 <- forecast(auto.arima(ts\_S03\_Var07), h = 140)  
forecast\_S02\_Var02 <- forecast(auto.arima(ts\_S02\_Var02), h = 140)  
forecast\_S02\_Var03 <- forecast(auto.arima(ts\_S02\_Var03), h = 140)  
forecast\_S01\_Var01 <- forecast(auto.arima(ts\_S01\_Var01), h = 140)  
forecast\_S01\_Var02 <- forecast(auto.arima(ts\_S01\_Var02), h = 140)  
  
# Create dataframe for forecasts  
forecasts\_df\_S06 <- data.frame(  
 SeriesInd = (subset\_S06\_clean\_Var07$SeriesInd[last\_observation\_index\_S06] + 1):(subset\_S06\_clean\_Var07$SeriesInd[last\_observation\_index\_S06] + 140),  
 category = rep("S06", 140),  
 Var05 = forecast\_S06\_Var05$mean,  
 Var07 = forecast\_S06\_Var07$mean)  
forecasts\_df\_S05 <- data.frame(  
 SeriesInd = (subset\_S05\_clean\_Var02$SeriesInd[last\_observation\_index\_S05] + 1):(subset\_S05\_clean\_Var02$SeriesInd[last\_observation\_index\_S05] + 140),  
 category = rep("S05", 140),  
 Var02 = forecast\_S05\_Var02$mean,  
 Var03 = forecast\_S05\_Var03$mean)  
forecasts\_df\_S04 <- data.frame(  
 SeriesInd = (subset\_S04\_clean\_Var01$SeriesInd[last\_observation\_index\_S04] + 1):(subset\_S04\_clean\_Var01$SeriesInd[last\_observation\_index\_S04] + 140),  
 category = rep("S04", 140),  
 Var01 = forecast\_S04\_Var01$mean,  
 Var02 = forecast\_S04\_Var02$mean)  
forecasts\_df\_S03 <- data.frame(  
 SeriesInd = (subset\_S03\_clean\_Var07$SeriesInd[last\_observation\_index\_S03] + 1):(subset\_S03\_clean\_Var07$SeriesInd[last\_observation\_index\_S03] + 140),  
 category = rep("S03", 140),  
 Var05 = forecast\_S03\_Var05$mean,  
 Var07 = forecast\_S03\_Var07$mean)  
forecasts\_df\_S02 <- data.frame(  
 SeriesInd = (subset\_S02\_clean\_Var02$SeriesInd[last\_observation\_index\_S02] + 1):(subset\_S02\_clean\_Var02$SeriesInd[last\_observation\_index\_S02] + 140),  
 category = rep("S02", 140),  
 Var02 = forecast\_S02\_Var02$mean,  
 Var03 = forecast\_S02\_Var03$mean)  
forecasts\_df\_S01 <- data.frame(  
 SeriesInd = (subset\_S01\_clean\_Var01$SeriesInd[last\_observation\_index\_S01] + 1):(subset\_S01\_clean\_Var01$SeriesInd[last\_observation\_index\_S01] + 140),  
 category = rep("S01", 140),  
 Var01 = forecast\_S01\_Var01$mean,  
 Var02 = forecast\_S01\_Var02$mean)  
  
# Remove the last 140 rows from subset\_S06 to append forecasts  
n\_rows\_S06 <- nrow(subset\_S06)  
subset\_S06 <- subset\_S06[1:(n\_rows\_S06 - 140), ]  
n\_rows\_S05 <- nrow(subset\_S05)  
subset\_S05 <- subset\_S05[1:(n\_rows\_S05 - 140), ]  
n\_rows\_S04 <- nrow(subset\_S04)  
subset\_S04 <- subset\_S04[1:(n\_rows\_S04 - 140), ]  
n\_rows\_S03 <- nrow(subset\_S03)  
subset\_S03 <- subset\_S03[1:(n\_rows\_S03 - 140), ]  
n\_rows\_S02 <- nrow(subset\_S02)  
subset\_S02 <- subset\_S02[1:(n\_rows\_S02 - 140), ]  
n\_rows\_S01 <- nrow(subset\_S01)  
subset\_S01 <- subset\_S01[1:(n\_rows\_S01 - 140), ]

### Combine Original and Forecasted Data

# Combine original and forecasted data  
combined\_df\_S06 <- rbind(subset\_S06, forecasts\_df\_S06)  
combined\_df\_S05 <- rbind(subset\_S05, forecasts\_df\_S05)  
combined\_df\_S04 <- rbind(subset\_S04, forecasts\_df\_S04)  
combined\_df\_S03 <- rbind(subset\_S03, forecasts\_df\_S03)  
combined\_df\_S02 <- rbind(subset\_S02, forecasts\_df\_S02)  
combined\_df\_S01 <- rbind(subset\_S01, forecasts\_df\_S01)  
  
# Add a label column to differentiate original and predicted data points  
n\_rows\_combined\_S06 <- nrow(combined\_df\_S06)  
combined\_df\_S06$label <- "original"  
combined\_df\_S06$label[(n\_rows\_combined\_S06 - 139):n\_rows\_combined\_S06] <- "predicted"  
n\_rows\_combined\_S05 <- nrow(combined\_df\_S05)  
combined\_df\_S05$label <- "original"  
combined\_df\_S05$label[(n\_rows\_combined\_S05 - 139):n\_rows\_combined\_S05] <- "predicted"  
n\_rows\_combined\_S04 <- nrow(combined\_df\_S04)  
combined\_df\_S04$label <- "original"  
combined\_df\_S04$label[(n\_rows\_combined\_S04 - 139):n\_rows\_combined\_S04] <- "predicted"  
n\_rows\_combined\_S03 <- nrow(combined\_df\_S03)  
combined\_df\_S03$label <- "original"  
combined\_df\_S03$label[(n\_rows\_combined\_S03 - 139):n\_rows\_combined\_S03] <- "predicted"  
n\_rows\_combined\_S02 <- nrow(combined\_df\_S02)  
combined\_df\_S02$label <- "original"  
combined\_df\_S02$label[(n\_rows\_combined\_S02 - 139):n\_rows\_combined\_S02] <- "predicted"  
n\_rows\_combined\_S01 <- nrow(combined\_df\_S01)  
combined\_df\_S01$label <- "original"  
combined\_df\_S01$label[(n\_rows\_combined\_S01 - 139):n\_rows\_combined\_S01] <- "predicted"  
  
# Plot Var05 over SeriesInd for subset\_S06  
plot\_SO6\_var05 <- ggplot(combined\_df\_S06, aes(x = SeriesInd, y = Var05, color = label)) +  
 geom\_line() +  
 labs(title = "Var05 over SeriesInd", x = "SeriesInd", y = "Var05") +  
 scale\_color\_manual(values = c("original" = "blue", "predicted" = "red")) +  
 theme\_minimal()  
plot\_SO6\_var07 <- ggplot(combined\_df\_S06, aes(x = SeriesInd, y = Var07, color = label)) +  
 geom\_line() +  
 labs(title = "Var07 over SeriesInd", x = "SeriesInd", y = "Var07") +  
 scale\_color\_manual(values = c("original" = "blue", "predicted" = "red")) +  
 theme\_minimal()  
plot\_S05\_Var02 <- ggplot(combined\_df\_S05, aes(x = SeriesInd, y = Var02, color = label)) +  
 geom\_line() +  
 labs(title = "Var02 over SeriesInd - S05", x = "SeriesInd", y = "Var02") +  
 scale\_color\_manual(values = c("original" = "blue", "predicted" = "red")) +  
 theme\_minimal()  
plot\_S05\_Var03 <- ggplot(combined\_df\_S05, aes(x = SeriesInd, y = Var03, color = label)) +  
 geom\_line() +  
 labs(title = "Var03 over SeriesInd - S05", x = "SeriesInd", y = "Var03") +  
 scale\_color\_manual(values = c("original" = "blue", "predicted" = "red")) +  
 theme\_minimal()  
plot\_S04\_Var01 <- ggplot(combined\_df\_S04, aes(x = SeriesInd, y = Var01, color = label)) +  
 geom\_line() +  
 labs(title = "Var01 over SeriesInd - S04", x = "SeriesInd", y = "Var01") +  
 scale\_color\_manual(values = c("original" = "blue", "predicted" = "red")) +  
 theme\_minimal()  
plot\_S04\_Var02 <- ggplot(combined\_df\_S04, aes(x = SeriesInd, y = Var02, color = label)) +  
 geom\_line() +  
 labs(title = "Var02 over SeriesInd - S04", x = "SeriesInd", y = "Var02") +  
 scale\_color\_manual(values = c("original" = "blue", "predicted" = "red")) +  
 theme\_minimal()  
plot\_S03\_Var05 <- ggplot(combined\_df\_S03, aes(x = SeriesInd, y = Var05, color = label)) +  
 geom\_line() +  
 labs(title = "Var05 over SeriesInd - S03", x = "SeriesInd", y = "Var05") +  
 scale\_color\_manual(values = c("original" = "blue", "predicted" = "red")) +  
 theme\_minimal()  
plot\_S03\_Var07 <- ggplot(combined\_df\_S03, aes(x = SeriesInd, y = Var07, color = label)) +  
 geom\_line() +  
 labs(title = "Var07 over SeriesInd - S03", x = "SeriesInd", y = "Var07") +  
 scale\_color\_manual(values = c("original" = "blue", "predicted" = "red")) +  
 theme\_minimal()  
plot\_S02\_Var02 <- ggplot(combined\_df\_S02, aes(x = SeriesInd, y = Var02, color = label)) +  
 geom\_line() +  
 labs(title = "Var02 over SeriesInd - S02", x = "SeriesInd", y = "Var02") +  
 scale\_color\_manual(values = c("original" = "blue", "predicted" = "red")) +  
 theme\_minimal()  
plot\_S02\_Var03 <- ggplot(combined\_df\_S02, aes(x = SeriesInd, y = Var03, color = label)) +  
 geom\_line() +  
 labs(title = "Var03 over SeriesInd - S02", x = "SeriesInd", y = "Var03") +  
 scale\_color\_manual(values = c("original" = "blue", "predicted" = "red")) +  
 theme\_minimal()  
plot\_S01\_Var01 <- ggplot(combined\_df\_S01, aes(x = SeriesInd, y = Var01, color = label)) +  
 geom\_line() +  
 labs(title = "Var01 over SeriesInd - S01", x = "SeriesInd", y = "Var01") +  
 scale\_color\_manual(values = c("original" = "blue", "predicted" = "red")) +  
 theme\_minimal()  
plot\_S01\_Var02 <- ggplot(combined\_df\_S01, aes(x = SeriesInd, y = Var02, color = label)) +  
 geom\_line() +  
 labs(title = "Var02 over SeriesInd - S01", x = "SeriesInd", y = "Var02") +  
 scale\_color\_manual(values = c("original" = "blue", "predicted" = "red")) +  
 theme\_minimal()

### Visualization of Forecasted Data



### Conclusion

In this project we walked through a full time series analysis to understand a de-identified dataset and ultimately to generate forecasts for 12 individual variables.