6402 Project

2025-12-07

This R markdown file contains code for baseline ARMA Modeling techniques as well as simple forecasting based off the initial ARMA modeling results.

```r
# Load Libraries

library(readr)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```r
library(ggplot2)
library(forecast)
```

```
## Registered S3 method overwritten by 'quantmod':
##    method            from
##    as.zoo.data.frame zoo
```

```r
library(vars)
```

```
## Loading required package: MASS

## Loading required package: strucchange

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: sandwich

## Loading required package: urca

## Loading required package: lmtest
```

```
library(gridExtra)
library(Metrics)
```

```
##
## Attaching package: 'Metrics'

## The following object is masked from 'package:forecast':
##
##      accuracy
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following object is masked from 'package:gridExtra':
##
##      combine

## The following object is masked from 'package:MASS':
##
##      select

## The following objects are masked from 'package:stats':
##
##      filter, lag

## The following objects are masked from 'package:base':
##
##      intersect, setdiff, setequal, union
```

```
veh <- read_csv("/Users/niraj/Downloads/VehicleData-1 (1).csv")
```

```
## Rows: 295 Columns: 3

## -- Column specification ---------------------------------------------------
## Delimiter: ","
## chr (1): DATE
## dbl (2): Total Sales, New Orders
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
veh$DATE <- mdy(veh$DATE)
veh <- veh[order(veh$DATE), ]

sales_ts  <- ts(veh$`Total Sales`, start = c(2000,1), frequency = 12)
orders_ts <- ts(veh$`New Orders`, start = c(2000,1), frequency = 12)

ts_data <- cbind(sales_ts, orders_ts)
```

```r
# ARMA MODEL - Total Sales

library(readr)
library(lubridate)
library(forecast)
library(ggplot2)
library(gridExtra)


veh <- read_csv("/Users/niraj/Downloads/VehicleData-1 (1).csv")
```

```
## Rows: 295 Columns: 3
## -- Column specification -----------------------------------------------------
## Delimiter: ","
## chr (1): DATE
## dbl (2): Total Sales, New Orders
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
veh$DATE <- mdy(veh$DATE)
veh <- veh[order(veh$DATE), ]

# Create time series
sales_ts <- ts(
  veh$`Total Sales`,
  start = c(2000, 1),
  frequency = 12
)


# 1. Stationarity Check via Differencing

sales_diff <- diff(sales_ts)

autoplot(sales_diff) +
  ggtitle("Differenced Total Sales (Stationary Input for ARMA)") +
  theme_minimal()
```
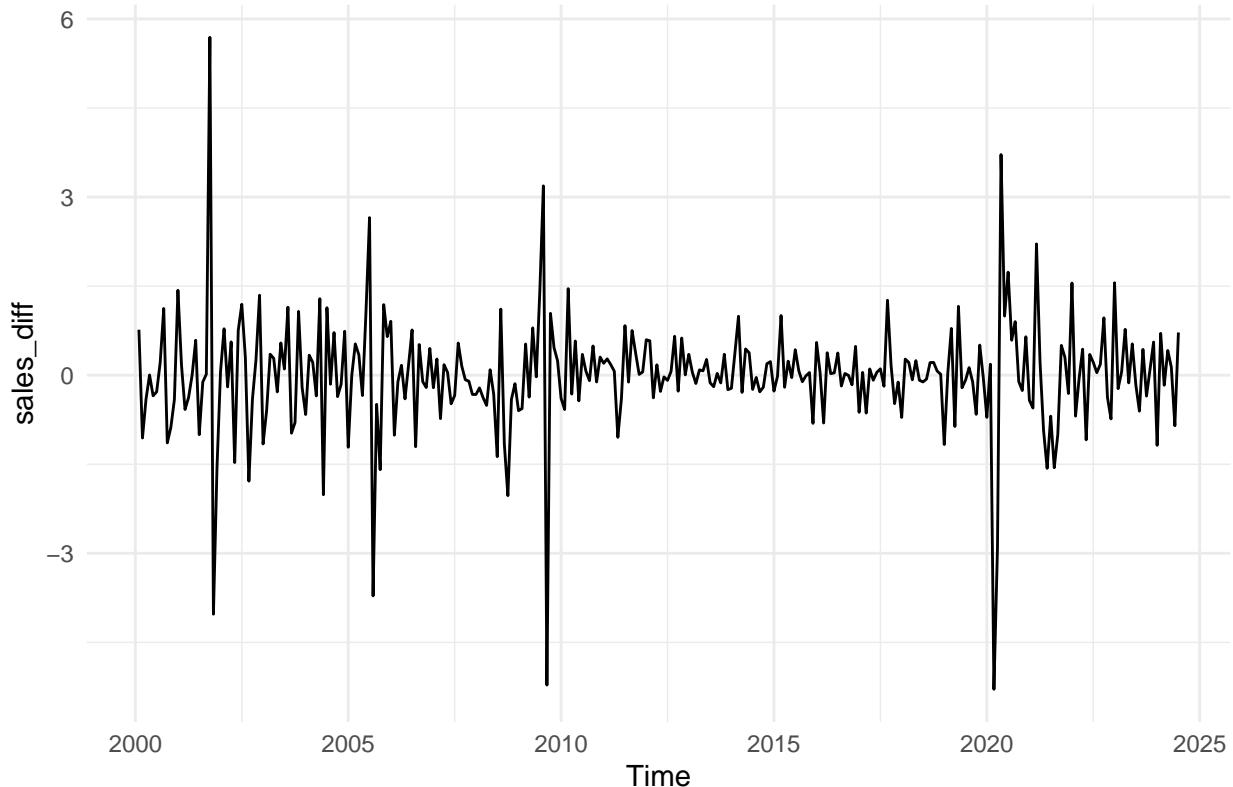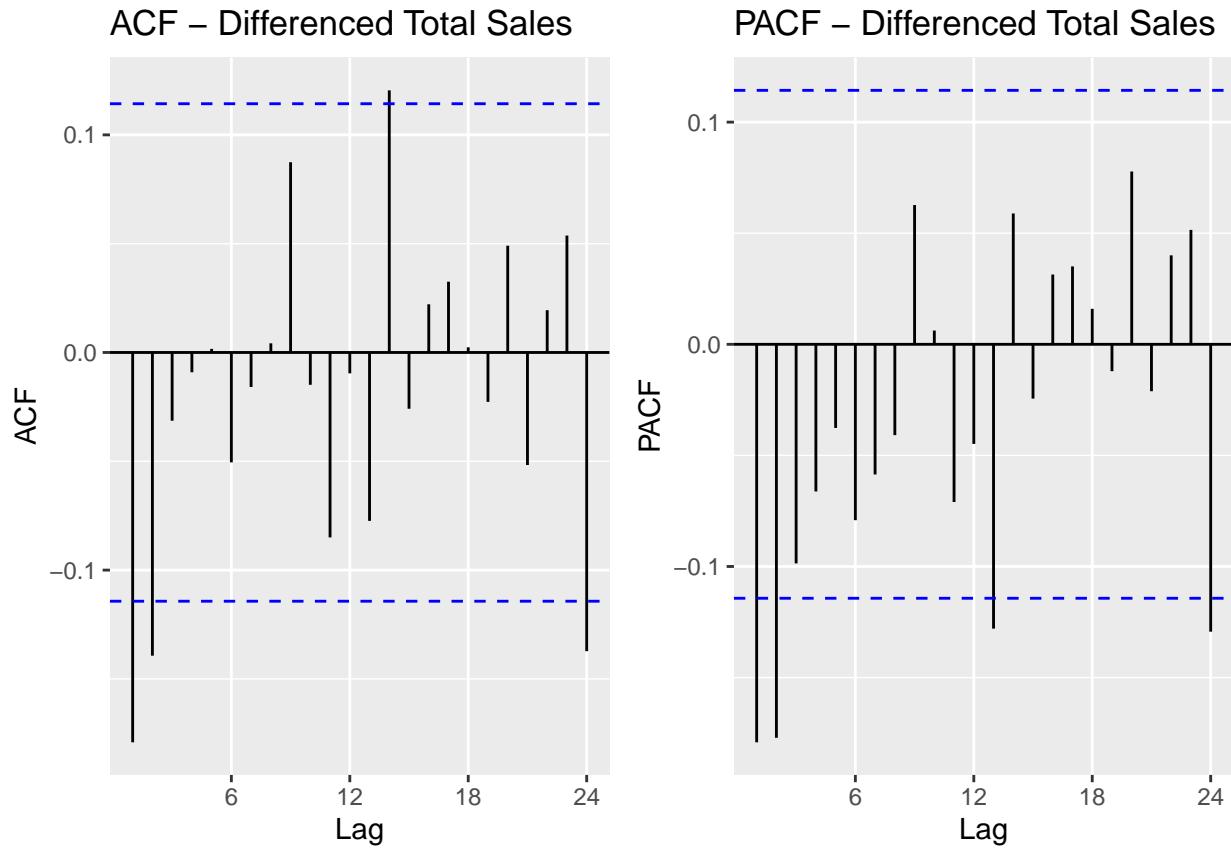
Differenced Total Sales (Stationary Input for ARMA)


```r
# 2. ACF/PACF for ARMA Order Selection

acf_sales  <- ggAcf(sales_diff)  + ggtitle("ACF - Differenced Total Sales")
pacf_sales <- ggPacf(sales_diff) + ggtitle("PACF - Differenced Total Sales")

grid.arrange(acf_sales, pacf_sales, ncol=2)
```

ACF – Differenced Total Sales

PACF – Differenced Total Sales


```
# 3. Fit ARMA(p,q) Model

arma_sales <- auto.arima(
  sales_ts,
  d = 1,
  seasonal = FALSE,
  stepwise = FALSE,
  approximation = FALSE
)

summary(arma_sales)
```

```
## Series: sales_ts
## ARIMA(1,1,1)
##
## Coefficients:
##          ar1      ma1
##       0.5215  -0.7776
## s.e.  0.1125   0.0827
##
## sigma^2 = 0.8758:  log likelihood = -396.78
## AIC=799.56    AICc=799.64    BIC=810.61
##
## Training set error measures:
##                         ME      RMSE      MAE      MPE     MAPE      MASE
```

5

```
## Training set -0.01778367 0.9310896 0.5772647 -0.4350386 3.879926 0.4176031
##                          ACF1
## Training set 0.008227212
```
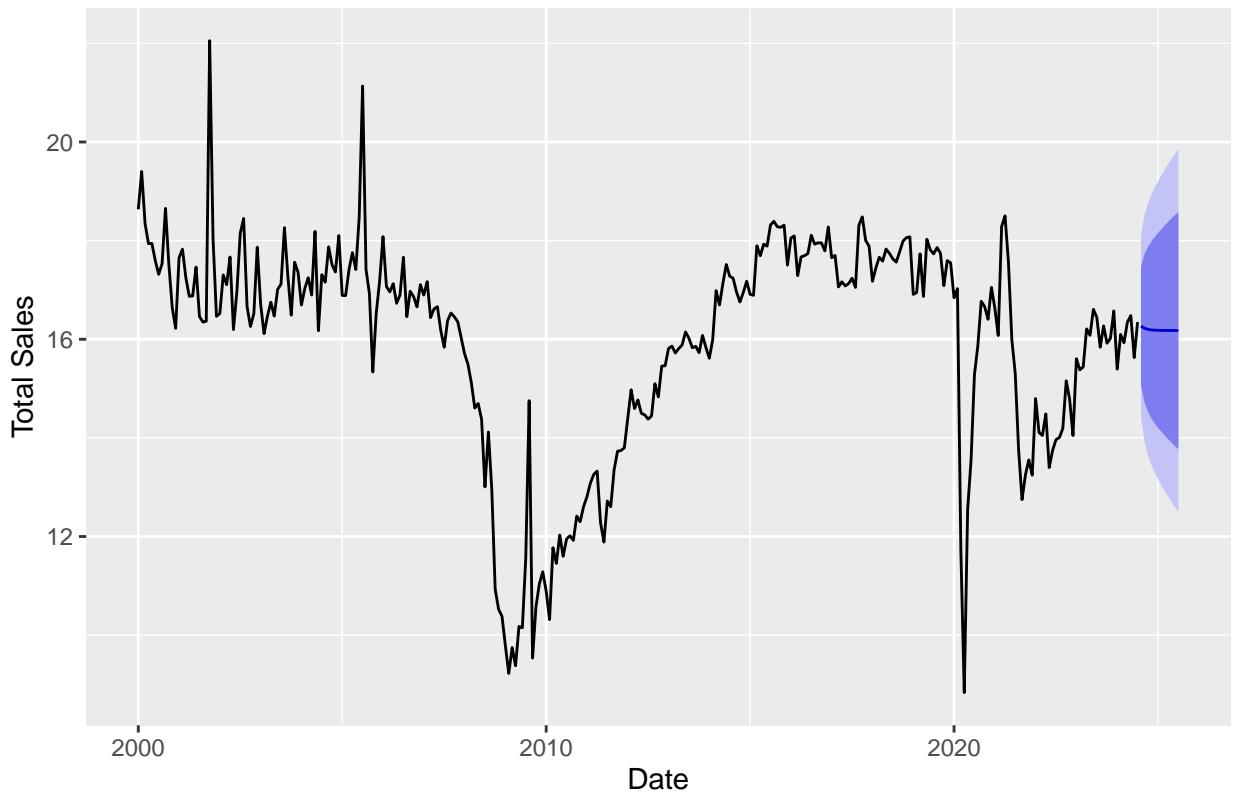
```
# 4. Forecast Next 12 Months

fc_sales <- forecast(arma_sales, h = 12)

autoplot(fc_sales) +
  ggtitle("ARMA Forecast - Total Sales") +
  xlab("Date") + ylab("Total Sales")
```
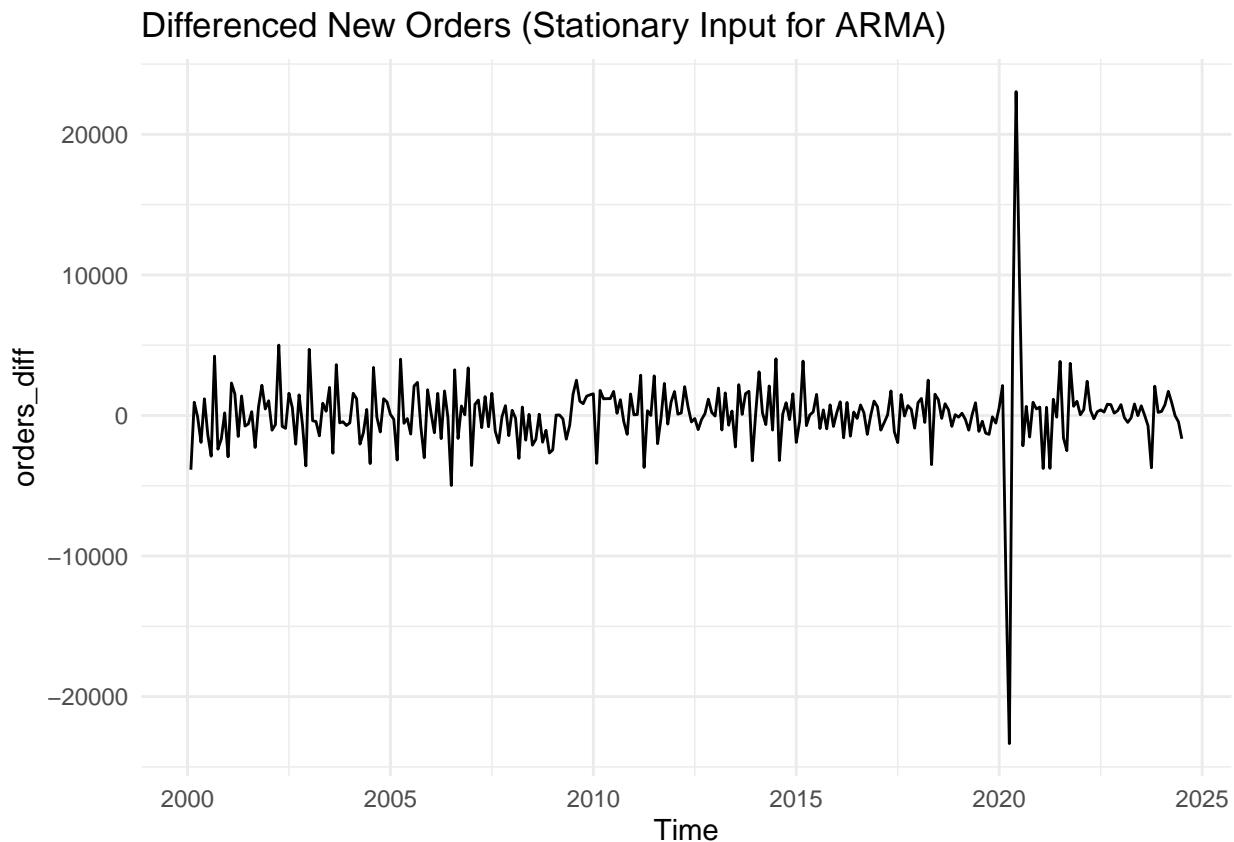


```
# ARMA MODEL - NEW ORDERS


# Create orders time series
orders_ts <- ts(
  veh$`New Orders`,
  start = c(2000, 1),
  frequency = 12
)

# 1. Stationarity Check via Differencing

orders_diff <- diff(orders_ts)
```
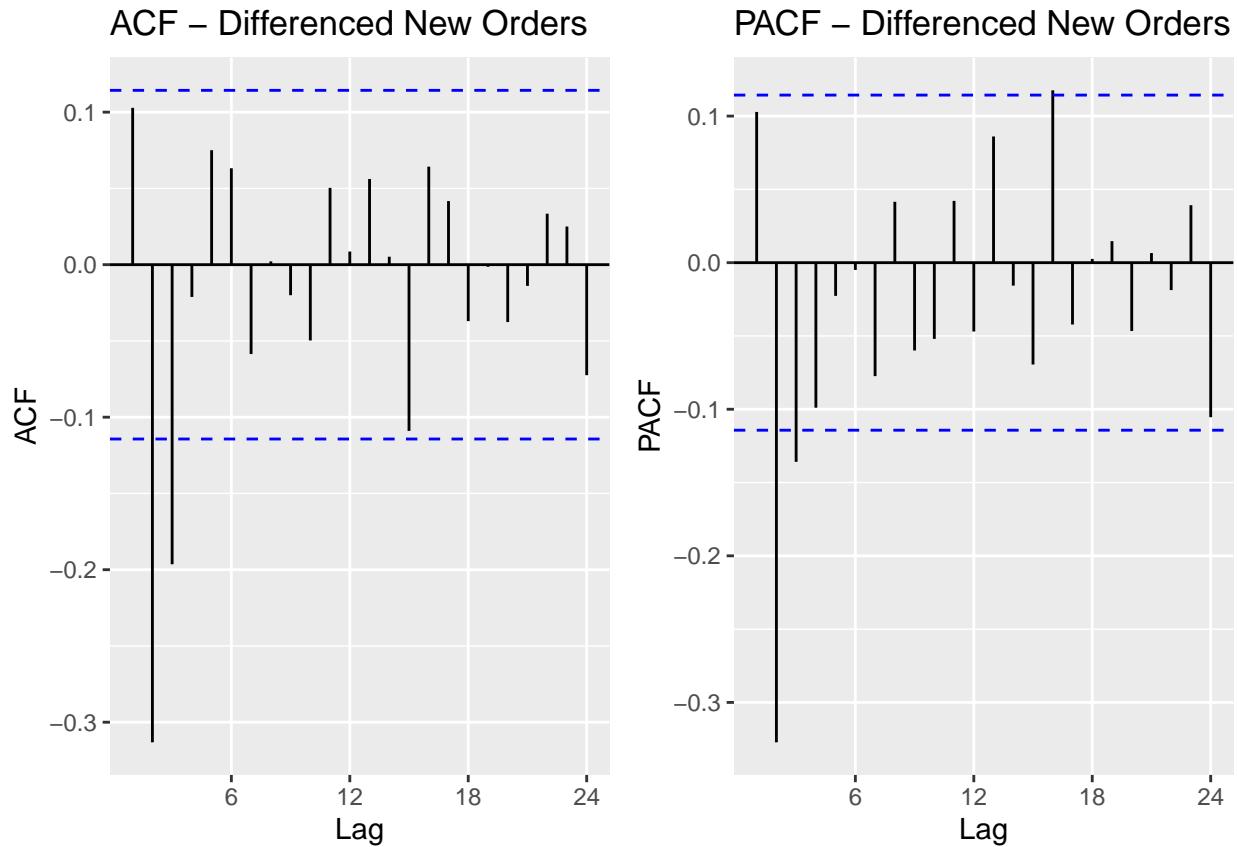
```
autoplot(orders_diff) +
  ggtitle("Differenced New Orders (Stationary Input for ARMA)") +
  theme_minimal()
```

Differenced New Orders (Stationary Input for ARMA)


```
# 2. ACF/PACF

acf_orders  <- ggAcf(orders_diff)  + ggtitle("ACF - Differenced New Orders")
pacf_orders <- ggPacf(orders_diff) + ggtitle("PACF - Differenced New Orders")

grid.arrange(acf_orders, pacf_orders, ncol=2)
```

ACF – Differenced New Orders

PACF – Differenced New Orders

```
# 3. Fit ARMA Model

arma_orders <- auto.arima(
  orders_ts,
  d = 1,
  seasonal = FALSE,
  stepwise = FALSE,
  approximation = FALSE
)

summary(arma_orders)
```

```
## Series: orders_ts
## ARIMA(2,1,1)
##
## Coefficients:
##          ar1      ar2      ma1
##       0.5664  -0.3664  -0.4945
## s.e.  0.1271   0.0555   0.1356
##
## sigma^2 = 6267477:  log likelihood = -2716.51
## AIC=5441.01   AICc=5441.15   BIC=5455.75
##
## Training set error measures:
##                     ME     RMSE    MAE        MPE      MAPE      MASE        ACF1
## Training set 104.6931 2486.462 1601.5 -0.1516608 4.022563 0.3843941 0.002306477
```

```r
# 4. Forecast Next 12 Months

fc_orders <- forecast(arma_orders, h = 12)

autoplot(fc_orders) +
  ggtitle("ARMA Forecast - New Orders") +
  xlab("Date") + ylab("New Orders")
```

ARMA Forecast – New Orders