Time Series Assignment: ARIMA Modeling

Kamil Stos & [Your Name]

Introduction

In this project, we will apply time series analysis techniques to model the Index of Industrial Production (IPI) for the alimentary industry, as reported by INSEE. Our primary objective is to use ARIMA (AutoRegressive Integrated Moving Average) models to understand and forecast industry trends. The dataset can be accessed through the following link: IPI: Industrie Alimentaire.

Data Overview

This section should provide a basic description of the dataset, including its range, the frequency of observations, and any preliminary observations regarding trends or seasonality.

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

# Load the data
data <- read.csv("data_industrie_alimentaire.csv", header = TRUE, sep = ";")

summary(data)
```

```
##    Libellé
##  Length:413
##  Class :character
##  Mode  :character
##  Indice.CVS.CJO.de.la.production.industrielle..base.100.en.2021....Industries.alimentaires..NAF.rév.
##  Length:413
##  Class :character
##  Mode  :character
##     Codes
##  Length:413
##  Class :character
##  Mode  :character
##  Indice.CVS.CJO.de.la.production.industrielle..base.100.en.2021....Industries.alimentaires..NAF.rév.
##  Length:413
##  Class :character
##  Mode  :character
##    Codes.1
##  Length:413
##  Class :character
##  Mode  :character
```

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

```r
# Load the data
data <- read.csv("data_industrie_alimentaire.csv", header = TRUE, sep = ";")

# first roww
head(data)
```

```
##        Libellé
## 1       idBank
## 2 Mises à jour
## 3      Période
## 4      2024-02
## 5      2024-01
## 6      2023-12
##   Indice.CVS.CJO.de.la.production.industrielle..base.100.en.2021....Industries.alimentaires..NAF.rév
## 1
## 2
## 3
## 4
## 5
## 6
##   Codes
## 1
## 2
## 3
## 4     A
## 5     A
## 6     A
##   Indice.CVS.CJO.de.la.production.industrielle..base.100.en.2021....Industries.alimentaires..NAF.rév
## 1
## 2
## 3
## 4
## 5
## 6
##   Codes.1
## 1
## 2
## 3
## 4
## 5       A
## 6       A
```

```r
# rename the columns as "index" and "date" and
colnames(data) <- c("date", "index1", "useless1", "index2", "useless2")

# remove the useless columns
data <- data[, c("date", "index1", "index2")]

# remove 3 first rows which contain useless data
data <- data[-c(1:3),]

# convert the index to numeric
data$index1 <- as.numeric(data$index1)
```

```r
# Part I

# convert the index to a time series
data_ts <- ts(data$index1, start = c(2010, 1), frequency = 12)

# show 2 plots at the same time
par(mfrow = c(2, 2))

# plot the time series
plot(data_ts, main = "Index", xlab = "Time", ylab = "Index", col = "blue", cex.main = 1.5, cex.lab = 1.5

# we can see that the time series is not stationary
# we can use the Augmented Dickey-Fuller test to check if the time series is stationary
adf.test(data_ts)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  data_ts
## Dickey-Fuller = -2.0625, Lag order = 7, p-value = 0.551
## alternative hypothesis: stationary
```

```r
# the p-value is 0.5 which is greater than 0.05, so we can't reject the null hypothesis at the 5% signi

# we can try to difference the time series
data_ts_diff <- diff(data_ts)

# plot the differenced time series
plot(data_ts_diff, main = "Differenced Index", xlab = "Time", ylab = "Index", col = "blue", cex.main =


adf.test(data_ts_diff)
```

```
## Warning in adf.test(data_ts_diff): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  data_ts_diff
## Dickey-Fuller = -10.8, Lag order = 7, p-value = 0.01
## alternative hypothesis: stationary
```

```r
# p-value is less than 0.01 so we can reject the null hypothesis at the 1% significance level : the tim

# check for heteroskedasticity
Box.test(data_ts_diff, lag = 20, type = "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  data_ts_diff
## X-squared = 75.75, df = 20, p-value = 2.042e-08
```

```r
# p-value is greater than 2*10^-8 so we can't reject the null hypothesis at the 5% significance level :
```

```r
# Part II

# fit an ARMA(p,q) model to the differenced time series since it's stationary


# we can try to fit an ARIMA model to the time series
fit <- auto.arima(data_ts_diff)

# show the model
summary(fit)
```

```
## Series: data_ts_diff
## ARIMA(2,0,1)(2,0,1)[12] with non-zero mean
##
## Coefficients:
##          ar1     ar2      ma1    sar1     sar2     sma1     mean
##       0.2174  0.0457  -0.7717  0.4794  -0.1954  -0.5921  -0.0242
## s.e.  0.0787  0.0644   0.0589  0.1029   0.0560   0.0972   0.0118
##
## sigma^2 = 1.689:  log likelihood = -685.52
## AIC=1387.04   AICc=1387.4   BIC=1419.15
##
## Training set error measures:
##                       ME     RMSE       MAE  MPE MAPE      MASE         ACF1
## Training set 0.007035435 1.288597 0.9542203 -Inf  Inf 0.5793147 0.0009021499
```

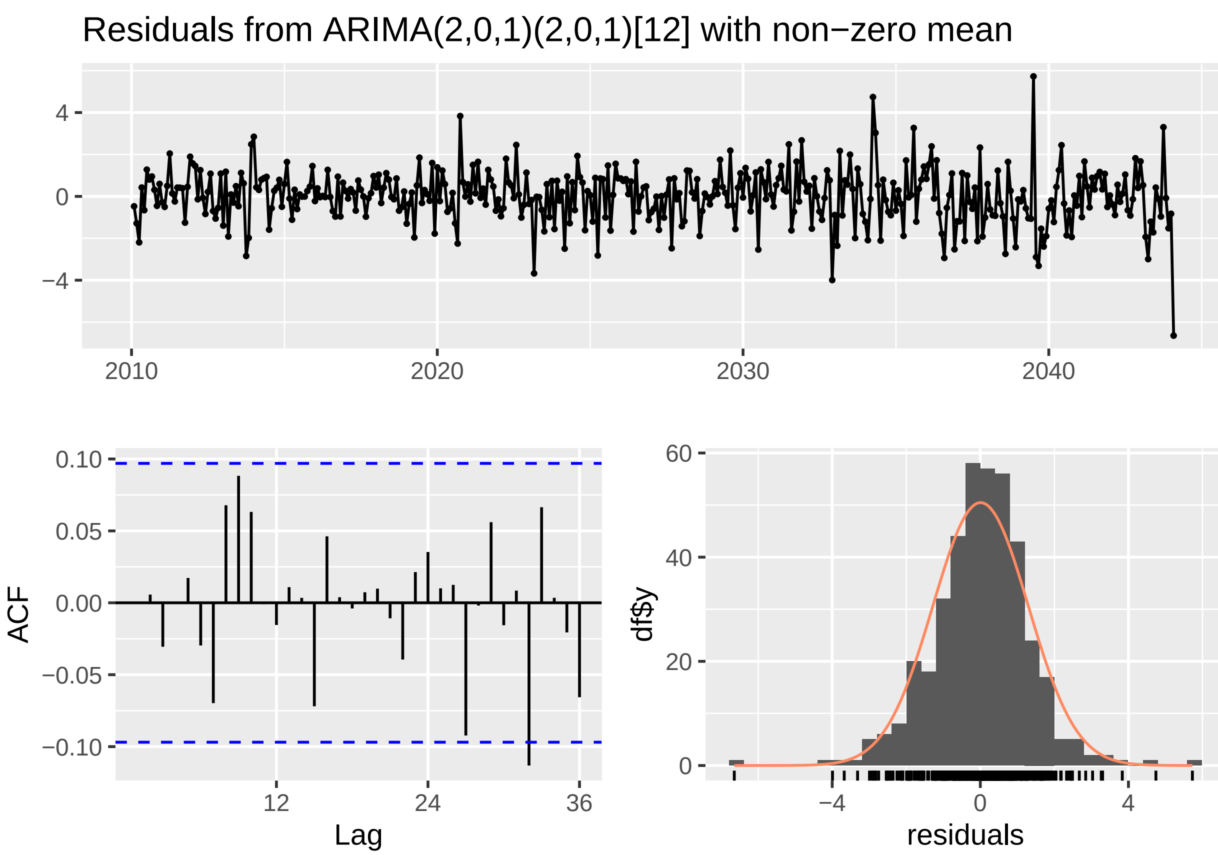
```r
# show the residuals
checkresiduals(fit)
```

```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(2,0,1)(2,0,1)[12] with non-zero mean
## Q* = 14.633, df = 18, p-value = 0.687
##
## Model df: 6.   Total lags used: 24
```

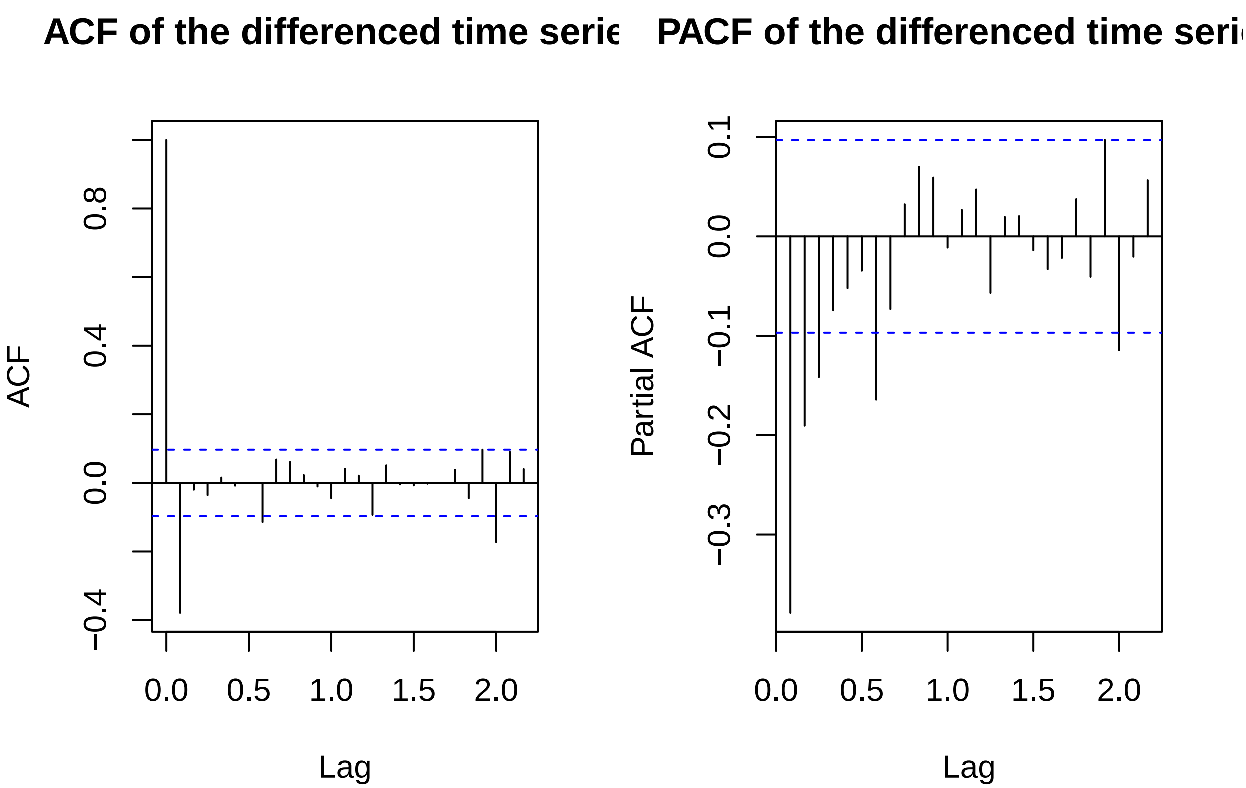
```r
# p-value is greater than 0.05 so we can't reject the null hypothesis at the 5% significance level : th

# we can forecast the time series

# checking the ACF and PACF
par(mfrow = c(1, 2))
```




```r
acf(data_ts_diff, main = "ACF of the differenced time series")
pacf(data_ts_diff, main = "PACF of the differenced time series")
```



rmarkdown::render("projet_serie_temp.Rmd")