Solutions

Lena Onyango

2024-03-06

Import Data

```
# importing the data

library(readxl) # importing the data

library(tidyverse) # data manipulation and visualization

library(janitor) # data cleaning

data <- read_excel("Market_Prices.xlsx")

# clean the names

data <- data %>%
  clean_names()

# convert date to date

data$dates <- as.Date(data$dates)

# cleaning the season column

data <- data %>%
  mutate(seasons = ifelse(seasons == "drought", "Drought",
                          ifelse(seasons == "Drought", "Drought",
                                 ifelse(seasons == "dry", "Dry",
                                        ifelse(seasons == "Dry", "Dry","Wet")))))
```

Visualization

```
## visualization

p1 <- ggplot(data, aes(x = dates, y = bull)) + geom_line(col="blue") + theme_minimal()+
  labs(x = "Date", y = "Price")
p2 <- ggplot(data, aes(x = dates, y = cow)) + geom_line() + theme_minimal()+
  labs(x = "Date", y = "Cow Price")

p3 <- ggplot(data, aes(x = dates, y = heifer)) + geom_line() + theme_minimal()+
  labs(x = "Date", y = "Heifer Price")
```

```r
p4 <- ggplot(data, aes(x = dates, y = steer)) + geom_line() + theme_minimal()+
  labs(x = "Date", y = "Steer Price")
# combining the plots

library(patchwork)

p1 + p2 + p3 + p4 + plot_annotation(tag_levels = "a",
            title = "Visualization of the Market Price per Animal")
```

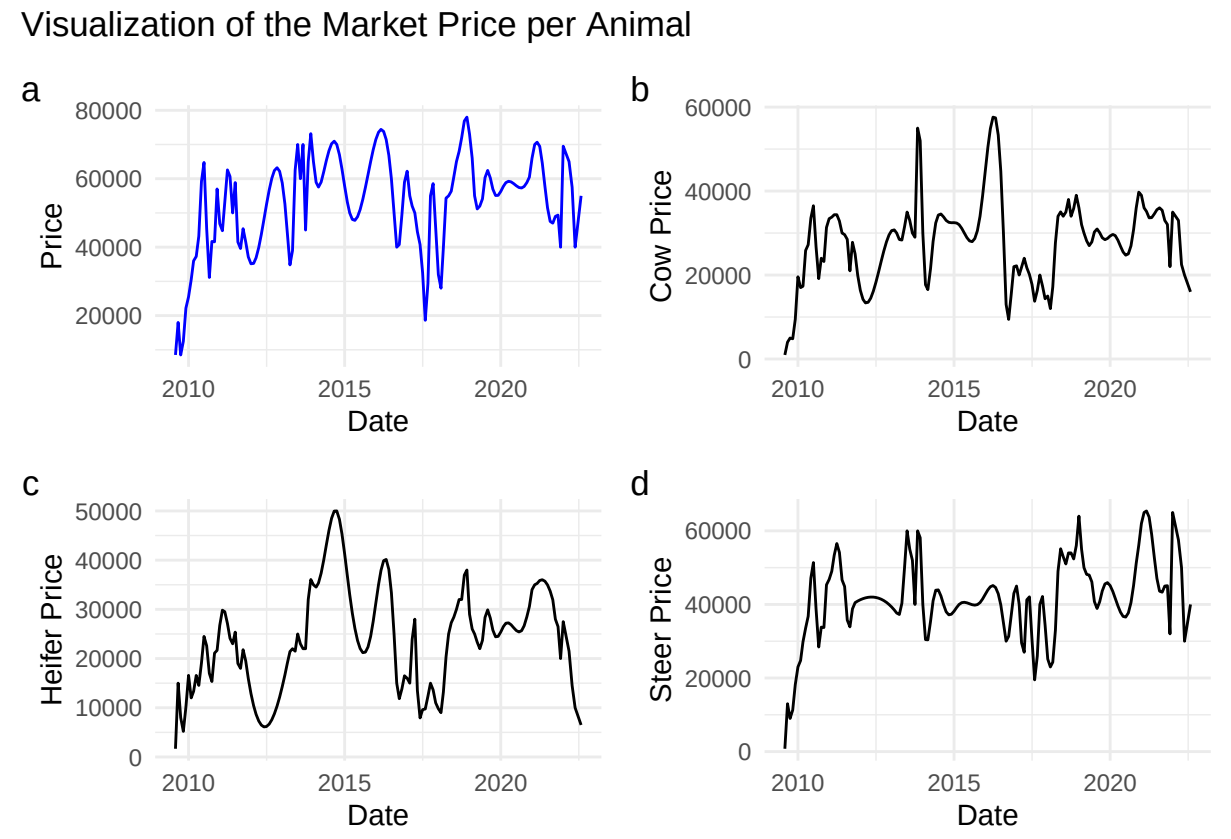


Figure 1: Visualization of the Market Price per Animal

Visualization Per Season for Different Animal

```r
p5 <- ggplot(data, aes(x = dates, y = bull)) + geom_line(col = "blue") + theme_minimal()+
  labs(x = "Date", y = "Bull Price") + facet_wrap(~seasons) +
  theme(axis.text.x = element_text(angle=45, vjust=.5, hjust=1))

p6 <- ggplot(data, aes(x = dates, y = cow)) + geom_line(col="red") + theme_minimal()+
  labs(x = "Date", y = "Cow Price")+ facet_wrap(~seasons)+
  theme(axis.text.x = element_text(angle=45, vjust=.5, hjust=1))

p7 <- ggplot(data, aes(x = dates, y = heifer)) + geom_line(col="green") + theme_minimal()+
  labs(x = "Date", y = "Heifer Price")+ facet_wrap(~seasons)+
  theme(axis.text.x = element_text(angle=45, vjust=.5, hjust=1))
```

```r
p8 <- ggplot(data, aes(x = dates, y = steer)) + geom_line(col="pink") + theme_minimal()+
  labs(x = "Date", y = "Steer Price")+ facet_wrap(~seasons)+
  theme(axis.text.x = element_text(angle=45, vjust=.5, hjust=1))
# combining the plots

library(patchwork)

p5 + p6 + p7 + p8 + plot_annotation(tag_levels = "a",
  title = "Visualization of the Market Price per Animal per Season")
```

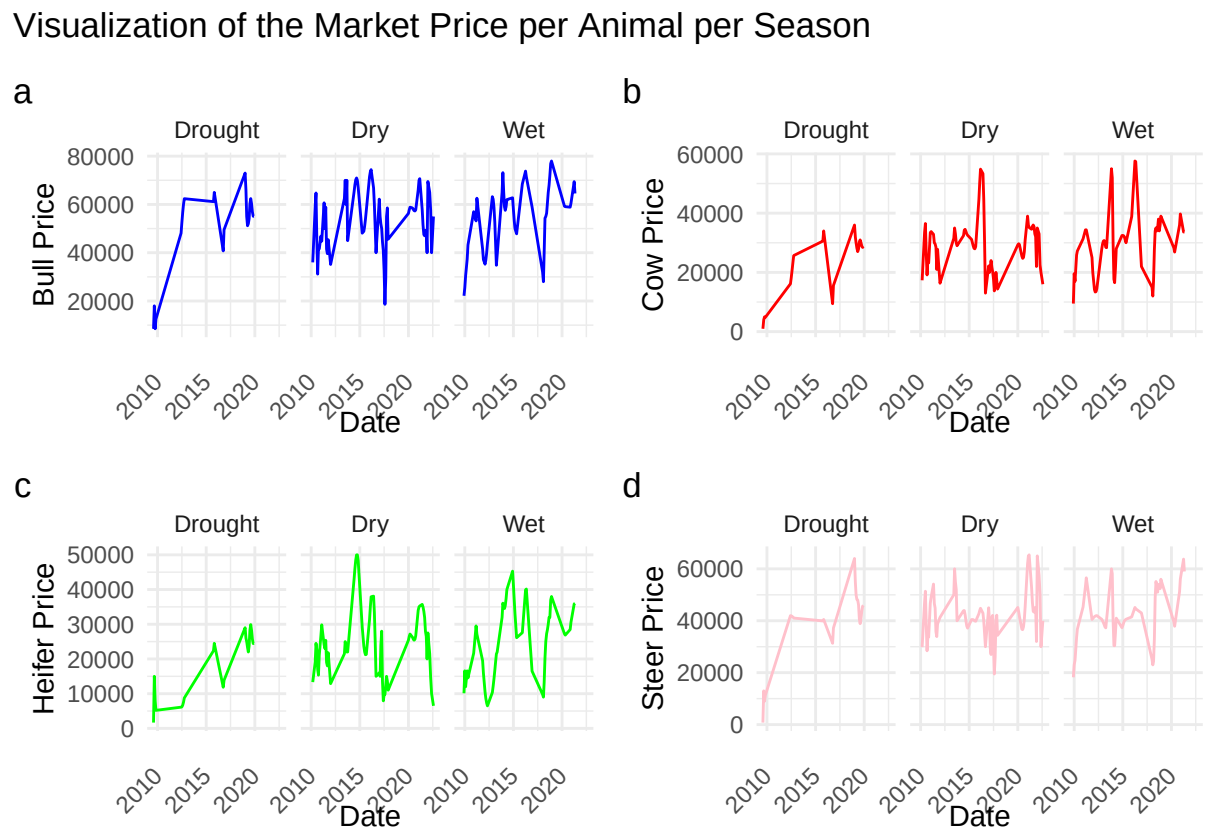


Figure 2: Visualization of the Market Price per Animal per Season

Testing for outliers

The Grubbs test allows to detect whether the highest or lowest value in a dataset is an outlier. There is no outlier in the data set since all the p-values are bigger than 0.05.

```r
library(outliers)
# bull - maximum value
test_1 <- grubbs.test(data$bull)

# bull - minimum value
test_2 <- grubbs.test(data$bull, opposite = TRUE)

# cow - maximum value
```

```
test_3 <- grubbs.test(data$cow)

# cow - minimum value
test_4 <- grubbs.test(data$cow, opposite = TRUE)


# heifer - maximum value
test_5 <- grubbs.test(data$heifer)

# heifer - minimum value
test_6 <- grubbs.test(data$heifer, opposite = TRUE)


# steer - maximum value
test_7 <- grubbs.test(data$steer)

# steer - minimum value
test_8 <- grubbs.test(data$steer, opposite = TRUE)


output <- c(test_1$p.value, test_3$p.value, test_5$p.value, test_7$p.value)

animal <- c("Bull", "Cow", "Heifer", "Steer")

data_out <- data.frame(animal, output)

colnames(data_out) <- c("Market Price", "p-value")
knitr::kable(data_out, caption = "Outlier Test for the Variables")
```

Table 1: Outlier Test for the Variables

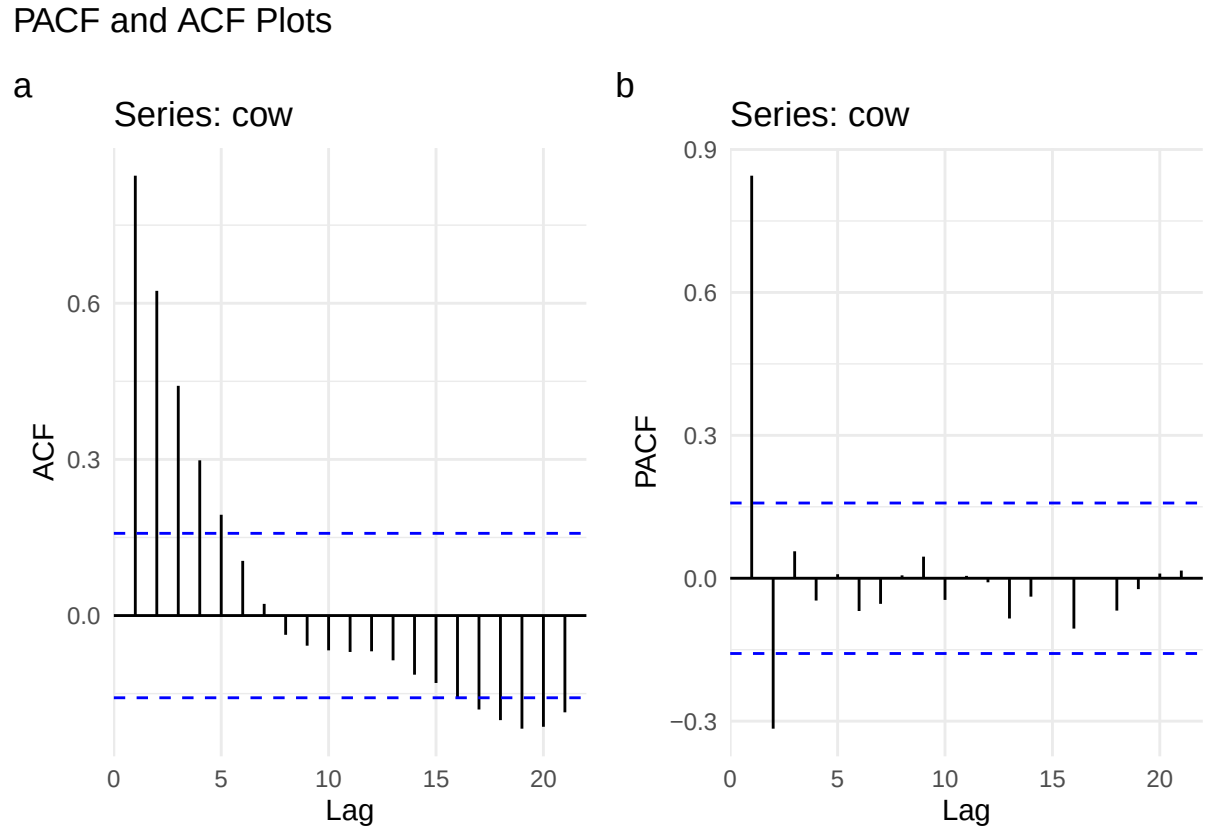
Market Price	p-value
Bull	0.0771641
Cow	0.2094533
Heifer	0.9094185
Steer	0.0071221

Modeling Livestock Prices

Cow Prices

```
library(xts) # time series modeling

library(tseries) # testing stationarity

library(forecast)

library(broom)

attach(data)

data_ts_cow <- xts(cow, dates)
```

```r
pa <- ggAcf(cow) + theme_minimal() # acf

pac <- ggPacf(cow) + theme_minimal()

pa + pac + plot_annotation(tag_levels = "a", title = "PACF and ACF Plots")
```




```r
# testing for stationarity

res_cow <- adf.test(cow)

statistic <- data.frame(res_cow$statistic, res_cow$p.value)
colnames(statistic) <- c("Statistic", "p-value")

knitr::kable(statistic, caption = "Stationarity test for cow series")
```

Table 2: Stationarity test for cow series

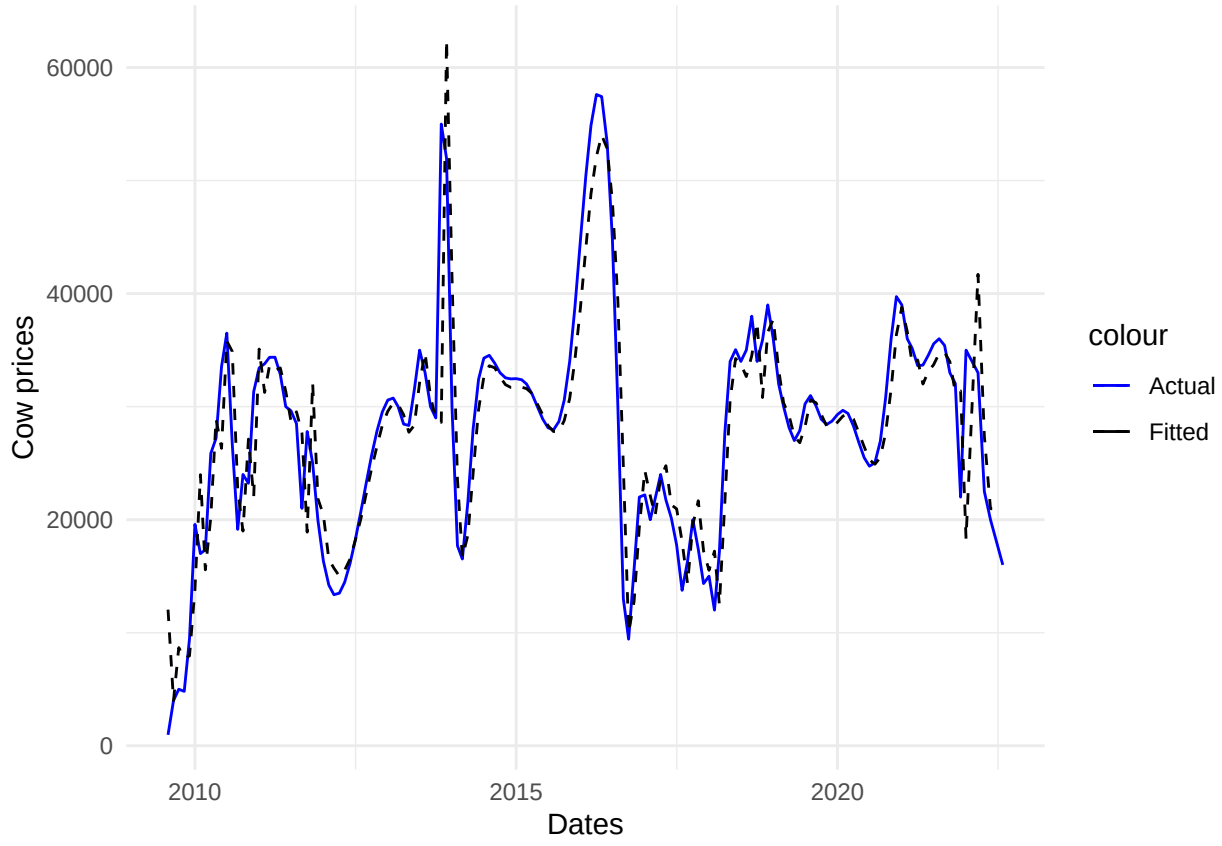
	Statistic	p-value
Dickey-Fuller	-3.890364	0.0164592

```r
# fitting the model

arima_cow <- auto.arima(data_ts_cow, seasonal = TRUE)

knitr::kable(tidy(arima_cow), caption = "Model for Cow Series")
```

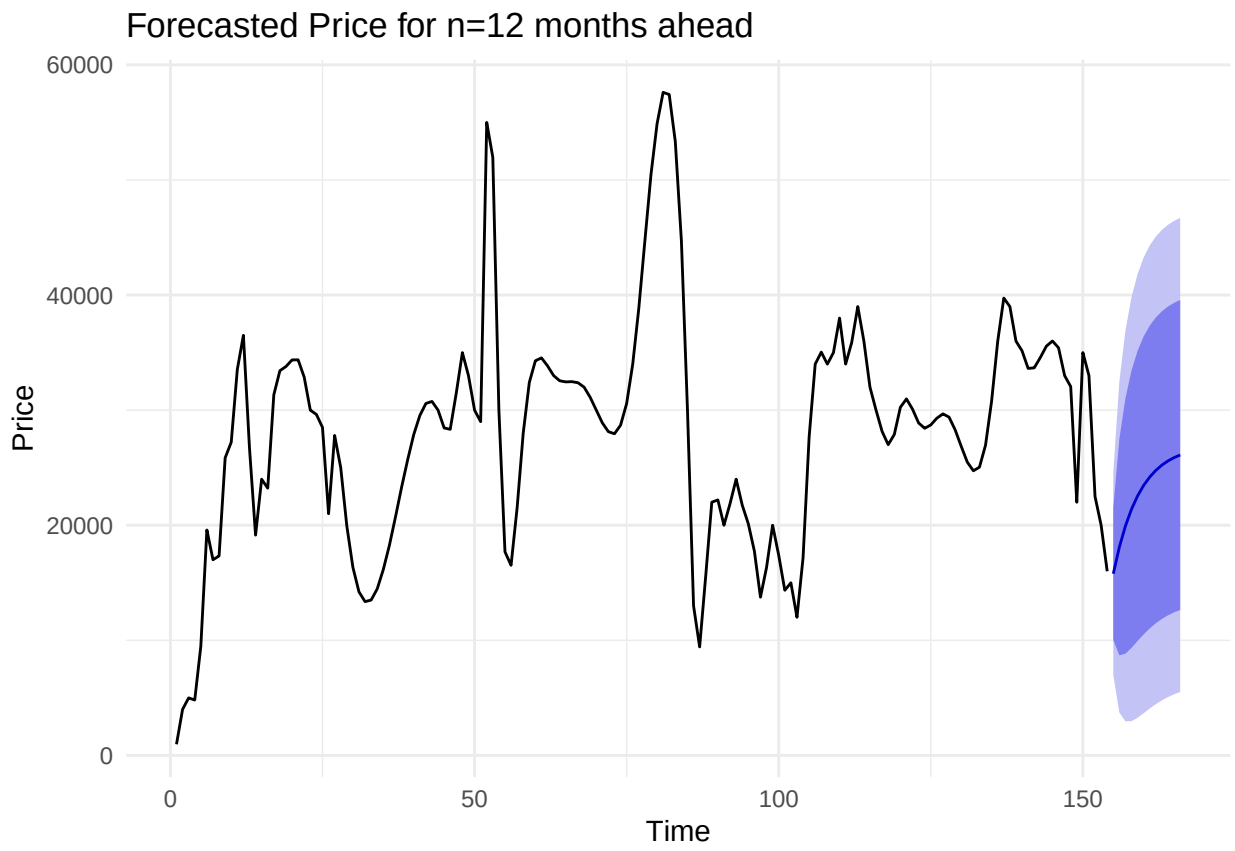
Table 3: Model for Cow Series

term	estimate	std.error
ar1	7.930143e-01	5.58524e-02
ma1	5.014095e-01	7.53200e-02
intercept	2.697881e+04	2.55064e+03

```
fitted_cows <- arima_cow$fitted

legend_title <- ""

ggplot(data, mapping=aes(x = dates)) + ylab("Cow prices")+
xlab("Dates")+ geom_line(aes(y = cow,color="actual"))+
geom_line(aes(y =fitted_cows,color="fitted"),linetype="dashed") +
theme_minimal() + scale_color_manual(values = c(actual = "blue",
fitted = "black"),labels = c(actual = "Actual", fitted = "Fitted")) +
scale_fill_manual(legend_title)
```




```
# forecasting for 12 months ahead

predCowprice = forecast(arima_cow, h=12)

autoplot(predCowprice) + theme_minimal() +
labs(x = "Time", y = "Price",
title = "Forecasted Price for n=12 months ahead")
```



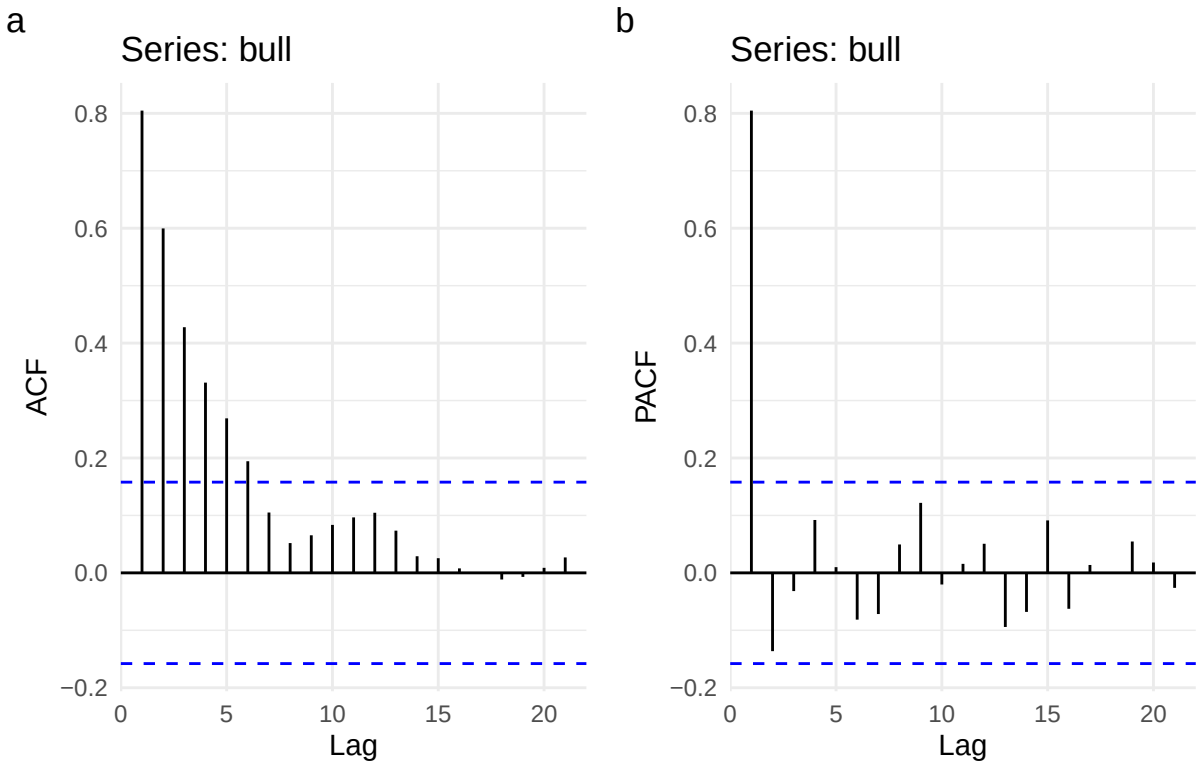

```
predCowprice$mean
```

```
## Time Series:
## Start = 155
## End = 166
## Frequency = 1
##  [1] 15787.62 18104.04 19940.99 21397.72 22552.92 23469.02 24195.50 24771.60
##  [9] 25228.46 25590.76 25878.07 26105.90
```

Bull Prices

```
library(xts) # time series modeling

library(tseries) # testing stationarity

library(forecast)

library(broom)

attach(data)

data_ts_bull <- xts(bull, dates)

pa <- ggAcf(bull) + theme_minimal() # acf

pac <- ggPacf(bull) + theme_minimal()
```

```
pa + pac + plot_annotation(tag_levels = "a", title = "PACF and ACF Plots")
```




```
# testing for stationarity

res_bull <- adf.test(bull)

statistic <- data.frame(res_bull$statistic, res_bull$p.value)
colnames(statistic) <- c("Statistic", "p-value")

knitr::kable(statistic, caption = "Stationarity test for bull series")
```

Table 4: Stationarity test for bull series

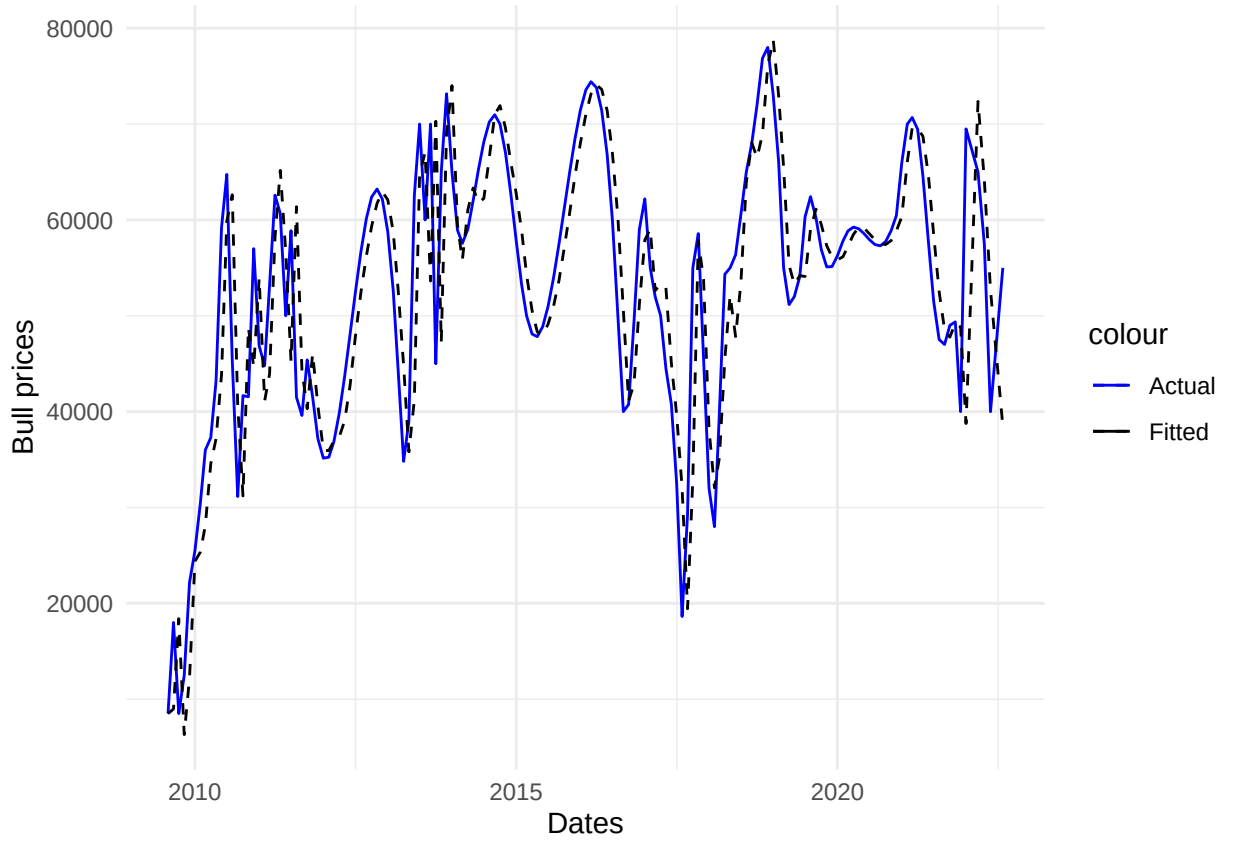
	Statistic	p-value
Dickey-Fuller	-4.426988	0.01

```
# fitting the model

arima_bull <- auto.arima(data_ts_bull, seasonal = TRUE)

knitr::kable(tidy(arima_bull), caption = "Model for Bull Series")
```

Table 5: Model for Bull Series

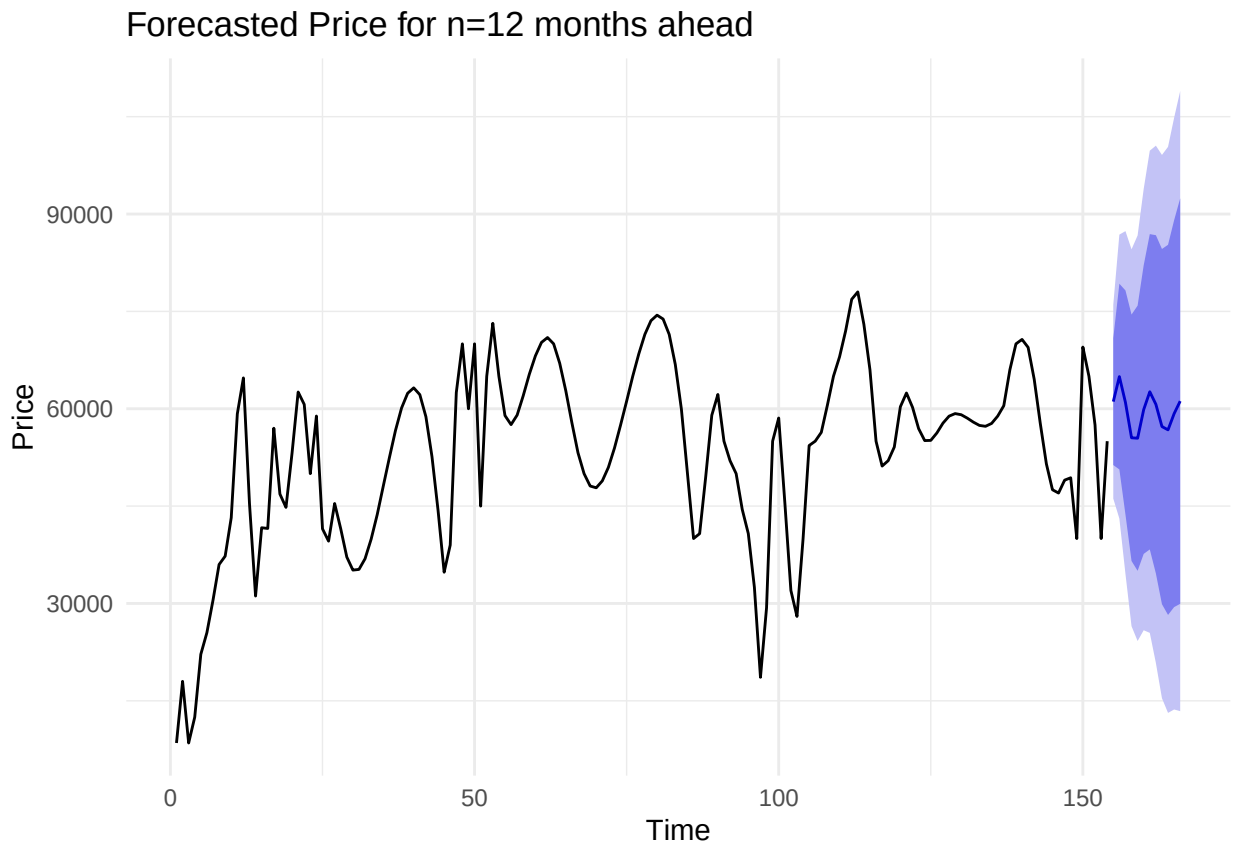
term	estimate	std.error
ar1	0.5015065	0.2432574
ar2	-0.7523113	0.2124897
ar3	-0.0886546	0.1216700
ma1	-0.4331070	0.2318392
ma2	0.6381231	0.2094140

```
fitted_bull <- arima_bull$fitted

legend_title <- ""

ggplot(data, mapping=aes(x = dates)) + ylab("Bull prices")+
xlab("Dates")+ geom_line(aes(y = bull,color="actual"))+
geom_line(aes(y =fitted_bull,color="fitted"),linetype="dashed") +
theme_minimal() + scale_color_manual(values = c(actual = "blue",
fitted = "black"),labels = c(actual = "Actual", fitted = "Fitted")) +
scale_fill_manual(legend_title)
```




```
# forecasting for 12 months ahead

predbullprice = forecast(arima_bull, h=12)

autoplot(predbullprice) + theme_minimal() +
```

```
labs(x = "Time", y = "Price",
title = "Forecasted Price for n=12 months ahead")
```



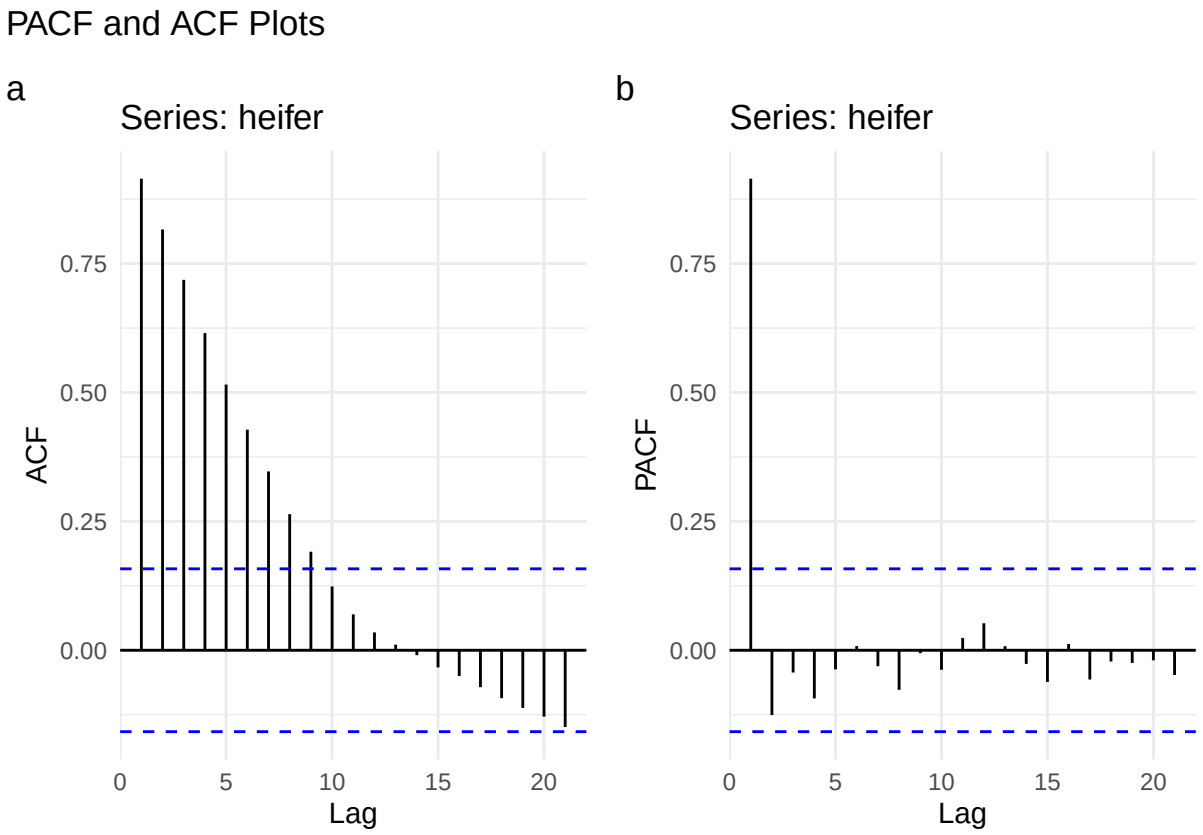

```
predbullprice$mean
```

```
## Time Series:
## Start = 155
## End = 166
## Frequency = 1
##  [1] 61099.75 64954.21 60968.52 55529.15 55458.03 59867.81 62615.08 60681.62
##  [9] 57254.24 56746.38 59241.56 61178.83
```

Heifer Prices

```
library(xts) # time series modeling

library(tseries) # testing stationarity

library(forecast)

library(broom)

attach(data)

data_ts_heifer <- xts(heifer, dates)
```

```r
pa <- ggAcf(heifer) + theme_minimal() # acf

pac <- ggPacf(heifer) + theme_minimal()

pa + pac + plot_annotation(tag_levels = "a", title = "PACF and ACF Plots")
```




```r
# testing for stationarity

res_heifer <- adf.test(heifer)

statistic <- data.frame(res_heifer$statistic, res_heifer$p.value)
colnames(statistic) <- c("Statistic", "p-value")

knitr::kable(statistic, caption = "Stationarity test for heifer series")
```

Table 6: Stationarity test for heifer series

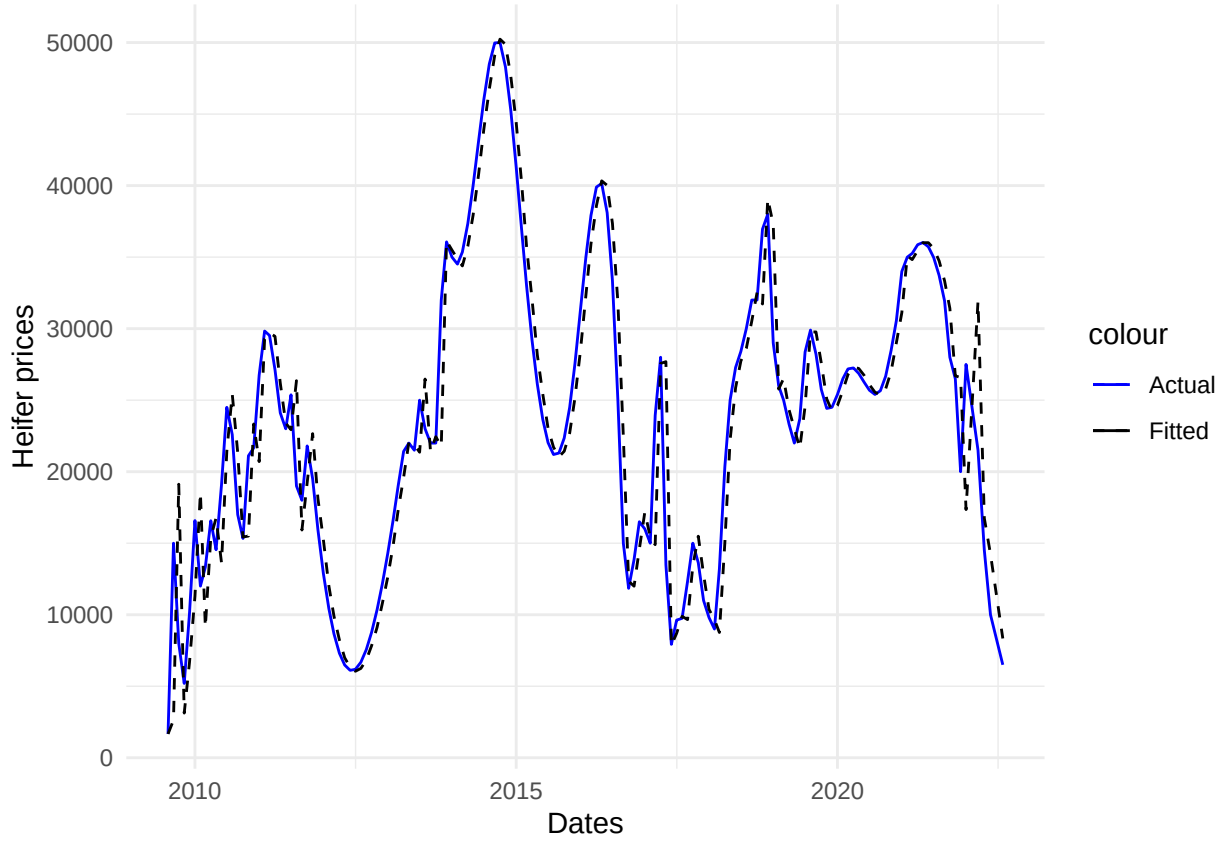
	Statistic	p-value
Dickey-Fuller	-2.424803	0.3993143

```r
# fitting the model

arima_heifer <- auto.arima(data_ts_heifer, seasonal = TRUE)

knitr::kable(tidy(arima_heifer), caption = "Model for Heifer Series")
```

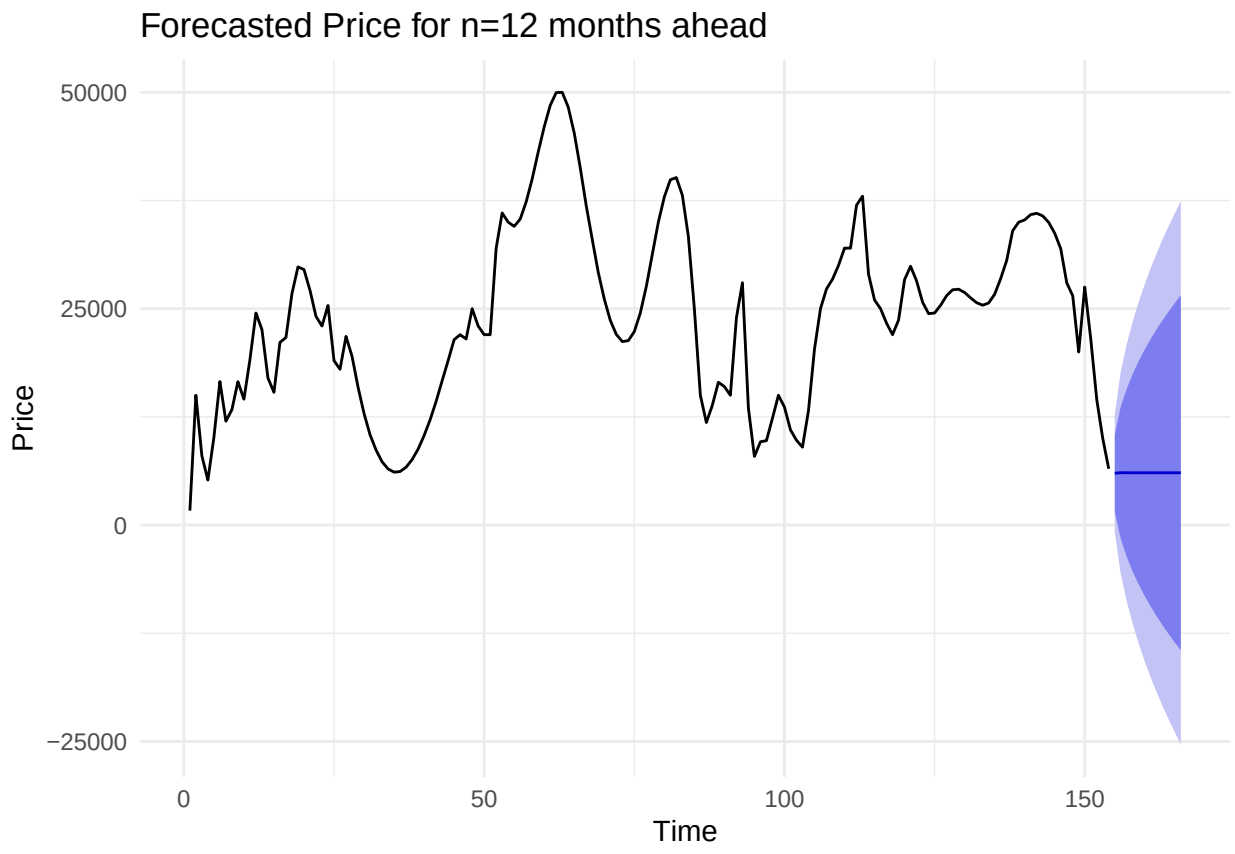
Table 7: Model for Heifer Series

term	estimate	std.error
ar1	-0.1324426	0.1710755
ma1	0.5329434	0.1425751

```
fitted_heifer <- arima_heifer$fitted

legend_title <- ""

ggplot(data, mapping=aes(x = dates)) + ylab("Heifer prices")+
xlab("Dates")+ geom_line(aes(y = heifer,color="actual"))+
geom_line(aes(y =fitted_heifer,color="fitted"),linetype="dashed") +
theme_minimal() + scale_color_manual(values = c(actual = "blue",
fitted = "black"),labels = c(actual = "Actual", fitted = "Fitted")) +
  scale_fill_manual(legend_title)
```




```
# forecasting for 12 months ahead

predheiferprice = forecast(arima_heifer, h=12)

autoplot(predheiferprice) + theme_minimal() +
labs(x = "Time", y = "Price",
title = "Forecasted Price for n=12 months ahead")
```



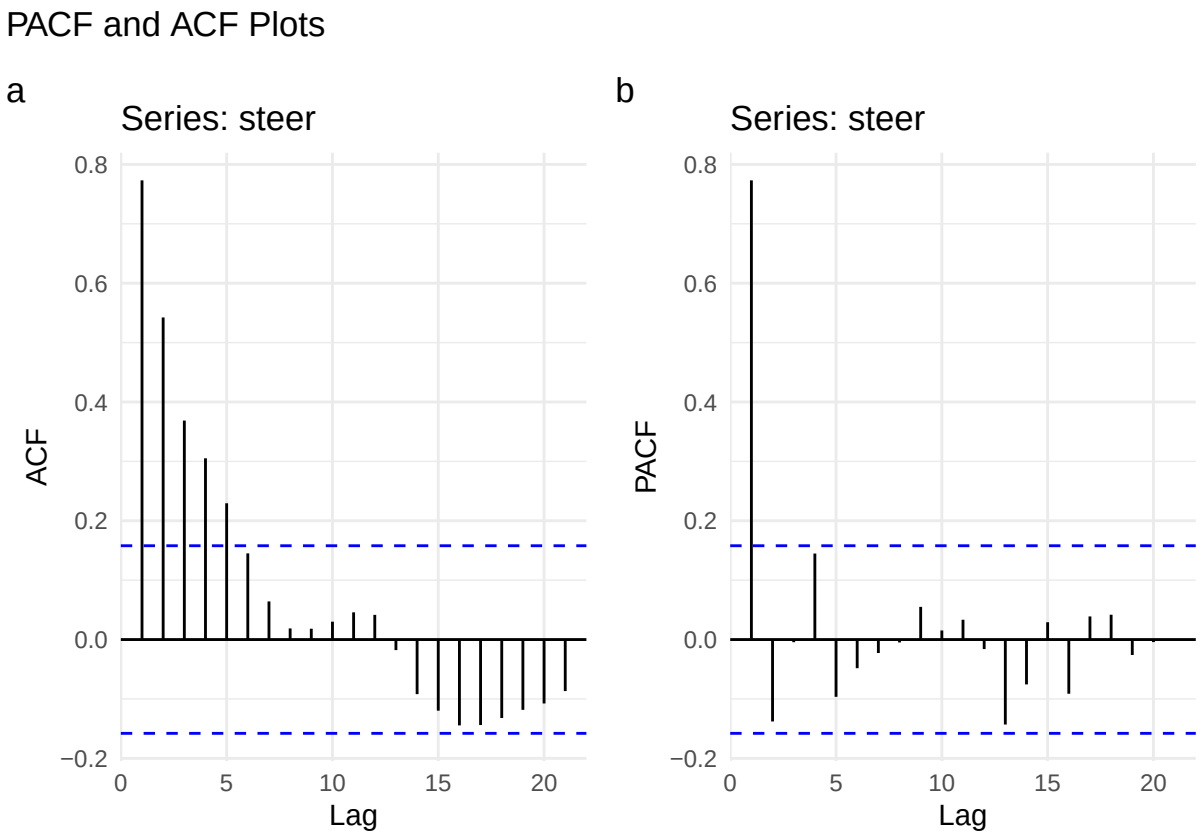

```
# the actual forecasted values are

predheiferprice$mean
```

```
## Time Series:
## Start = 155
## End = 166
## Frequency = 1
##  [1] 5982.398 6050.951 6041.872 6043.074 6042.915 6042.936 6042.933 6042.934
##  [9] 6042.934 6042.934 6042.934 6042.934
```

Steer Price

```
library(xts) # time series modeling

library(tseries) # testing stationarity

library(forecast)

library(broom)

attach(data)

data_ts_steer <- xts(steer, dates)

pa <- ggAcf(steer) + theme_minimal() # acf
```

```
pac <- ggPacf(steer) + theme_minimal()

pa + pac + plot_annotation(tag_levels = "a", title = "PACF and ACF Plots")
```




```
# testing for stationarity

res_steer <- adf.test(steer)

statistic <- data.frame(res_cow$statistic, res_cow$p.value)
colnames(statistic) <- c("Statistic", "p-value")

knitr::kable(statistic, caption = "Stationarity test for steer series")
```

Table 8: Stationarity test for steer series

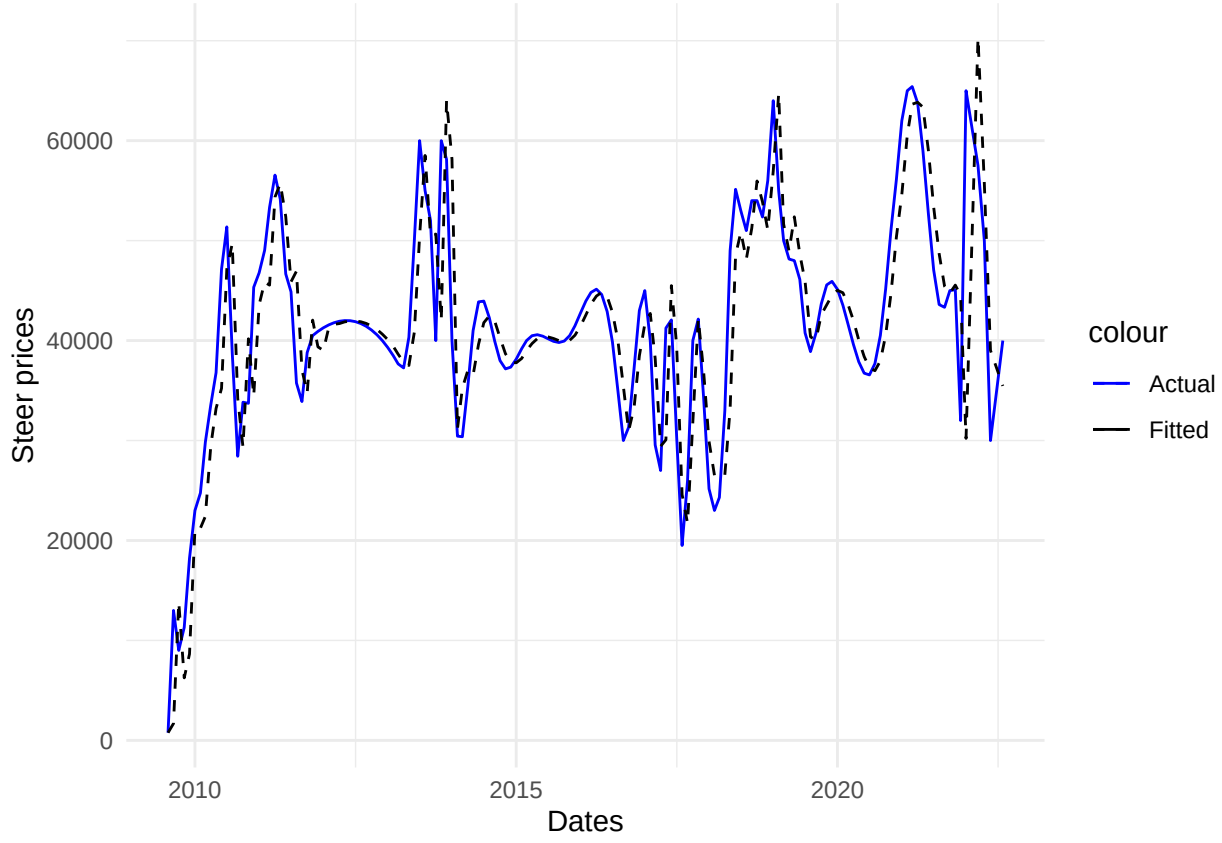
	Statistic	p-value
Dickey-Fuller	-3.890364	0.0164592

```
# fitting the model

arima_steer <- auto.arima(data_ts_steer, seasonal = TRUE)

knitr::kable(tidy(arima_steer), caption = "Model for Steer Series")
```

Table 9: Model for Steer Series

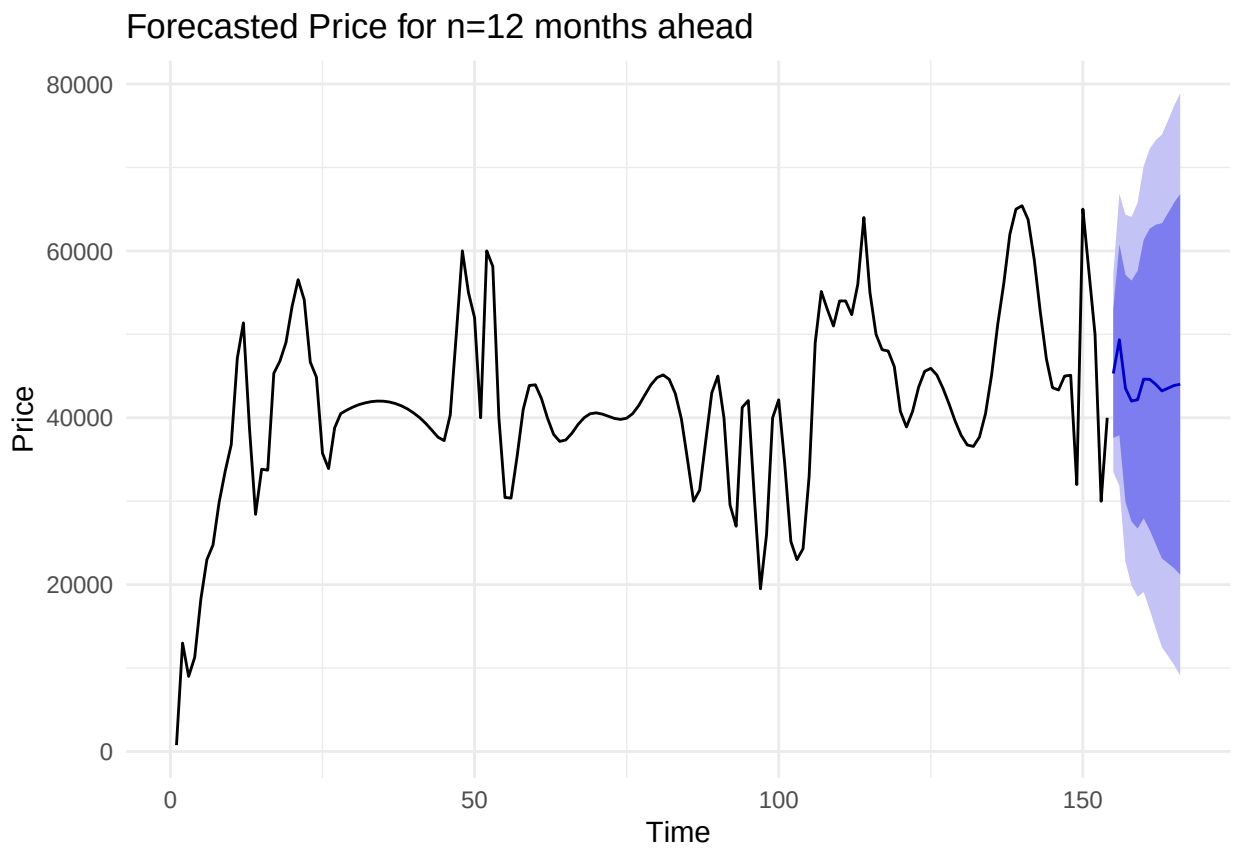
term	estimate	std.error
ar1	0.0946290	0.0822846
ar2	-0.1514552	0.0819577
ar3	-0.2980063	0.0815630
ar4	0.1207096	0.0856923

```
fitted_steer <- arima_steer$fitted

legend_title <- ""

ggplot(data, mapping=aes(x = dates)) + ylab("Steer prices")+
  xlab("Dates")+ geom_line(aes(y = steer,color="actual"))+
geom_line(aes(y =fitted_steer,color="fitted"),linetype="dashed") +
theme_minimal() + scale_color_manual(values = c(actual = "blue",
fitted = "black"),labels = c(actual = "Actual", fitted = "Fitted")) +
  scale_fill_manual(legend_title)
```




```
# forecasting for 12 months ahead

predsteerprice = forecast(arima_steer, h=12)

autoplot(predsteerprice) + theme_minimal() +
labs(x = "Time", y = "Price",
```

```
title = "Forecasted Price for n=12 months ahead")
```




```
# the actual forecasted values are

predsteerprice$mean
```

```
## Time Series:
## Start = 155
## End = 166
## Frequency = 1
##  [1] 45305.12 49347.39 43532.16 41995.79 42166.91 44636.70 44600.40 43986.45
##  [9] 43218.49 43547.75 43873.80 44009.53
```