Modeling and Forecasting the U.S. House Price Index

Patrick Omens

Email: patrickomens@ucsb.edu

Abstract

The goal of this project is to examine the U.S. House Price Index for the purposes of analyzing the possible trends to be able to forecast the HPI years into the future. We will use the Box Jenkin's approach for the SARIMA model, and we will use spectral analysis to identify possible cyclical patterns in the HPI.

The SARIMA model, or the Seasonal Autoregressive Integrated Moving Average, is used in time series analysis to predict future data points by deducing the past seasonality and patterns in the data. Spectral analysis transforms the data from the time domain into the frequency domain. It is possible we find cyclical patterns in the House Price Index, since the HPI is an economic indicator, and we know that the economy as a whole is very cyclical.

All together, this project can help us see where the housing market will be in the future. This can be of use to us who plan on eventually purchasing a single-family home.

Introduction

I chose this dataset because I find the recent rise of cost of living, and more specifically, the upsurge of home prices in America, relevant to my own life. One day, I'd like to be a homeowner to be able to raise my future kids. Owning a house is also the most common way of building wealth in America. This is part of the reason why owning a house is part of what many people call "The American Dream." It seems like this dream is becoming harder and harder to achieve as of recently. This project can help us to get realistic, accept the situation at hand, and effectively plan for the future that most of us want to have.

The data is quarterly and spans roughly 50 years (from 1975 to 2023). This dataset has been cited in countless studies to try to find the correlation between inflation and home prices. www.whitehouse.gov did an article on September 9, 2021 titled "House Prices and Inflation" which investigated the correlation during the tail-end of the pandemic, when cost of living skyrocketed across the board.

investopedia.com also wrote an article titled "Understanding the House Price Index (HPI) and How It Is Used." This article goes into how the HPI is calculated, how to tell if a house is a good price, and what factors may change the price of a house.

Data

This dataset shows the U.S. House Price Index from Q1 1975 to Q4 2023. The frequency of the data is quarterly, and the size of the dataset is 196.The U.S. Federal Housing Finance Agency collected the House Price Index data.

The house price index is a measure of the price changes in single-family housing in the United States. The HPI is based on repeat transactions. The estimates are generated by looking at the appreciation on repeated valuations of the same property over time.

This dataset is important because the house price index is an economic indicator. It can display the overall strength of the housing market, and the strength of the entire economy as a whole. When home prices go up, this usually indicates a stronger economy, whereas when they fall, this could signal an economy that is more vulnerable. For example, we see the HPI take a significant hit during the Great Recession starting in around 2008.

https://fred.stlouisfed.org/series/USSTHPI

Methodology

Model 1

SARIMA (p, d, q) x (P, D, Q) model

For this project, we will first implement the SARIMA model. This will be done by applying the Box-Jenkins method.

First, we plot the time series data, along with the ACF and PACF. We do this in order to determine whether or not we need to transform the data.

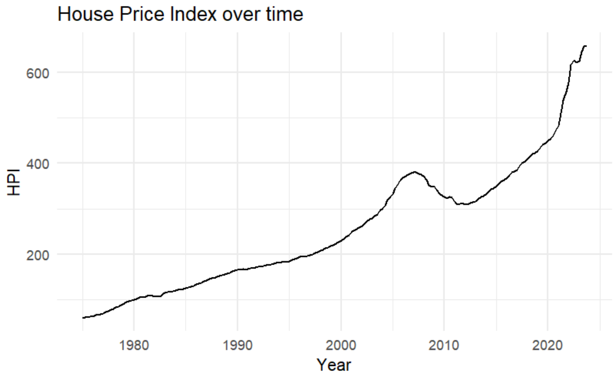
Secondly, we transform the data. The original ACF decreased very slowly, and the PACF was equal to about 1. Therefore, we differenced the logarithmic transformation of the data. However, the ACF of the new data still decreased fairly slowly, so we instead applied a second-order differencing to the data. In the newest ACF plot, the data no longer appeared to be highly correlated. We have now figured out the order of differencing, d. We later try to figure out the value for the autoregressive order (p) and the value for the moving average order (q). After this step, we will perform model diagnostics on the SARIMA model plots. We then compare the AIC and BIC of the different models, and choose the model with the lowest AIC and BIC. In this case, the model with the lowest AIC and BIC was the MA(2) model.

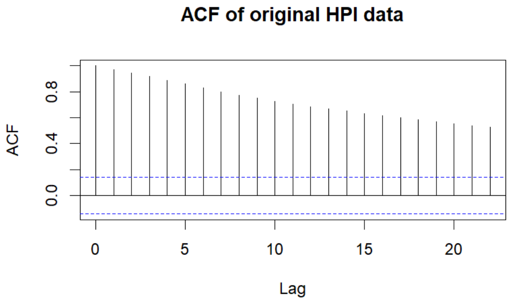
Model 2

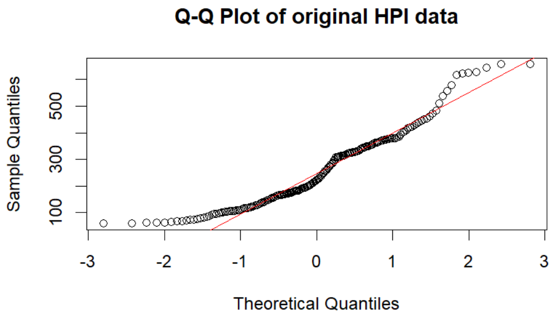
Spectral Analysis

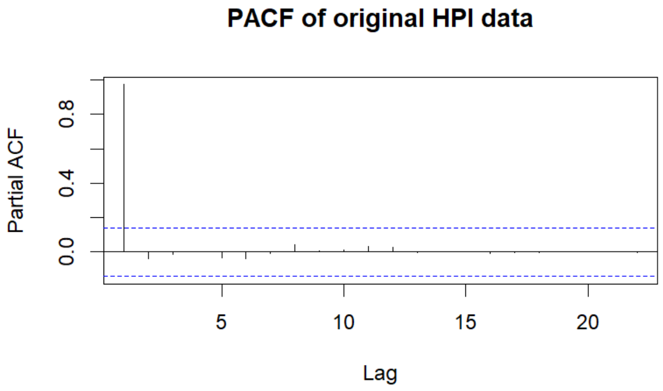
Spectral analysis is used to help us understand the possible cycles or echoing patterns in our time series data. It can show us how often these cycles can happen (frequency) and how powerful they are (amplitude). In our project, we see a clear periodic component between around 10 months and 15 months.

Results

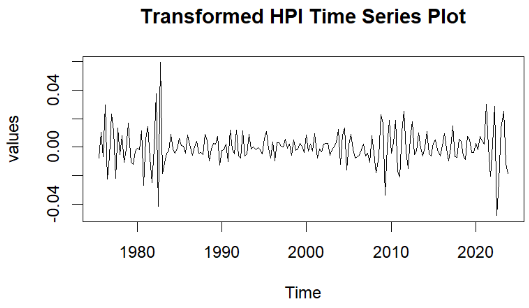


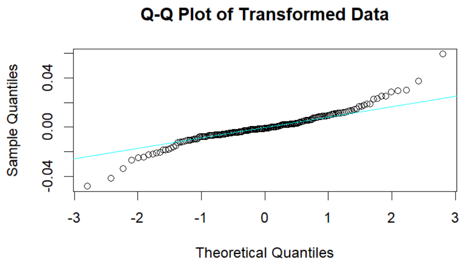


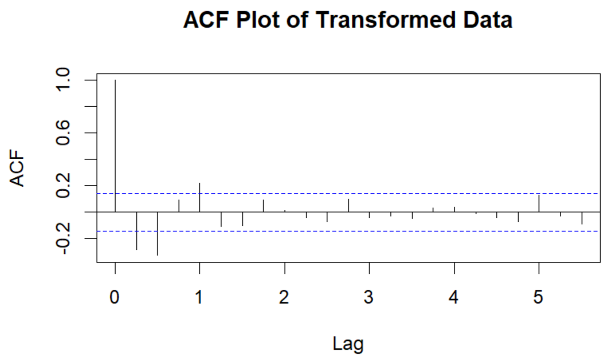


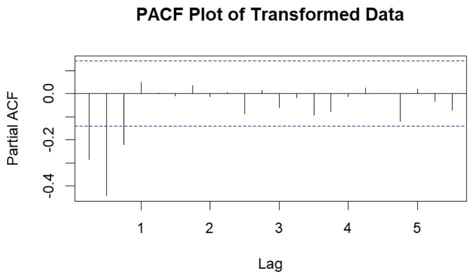


The original data, as well as its ACF, PACF, and Q-Q plot are shown above. The ACF plot has a very slow decay, and the PACF is nearly equal to 1. Because of this, we will perform differencing.









We can now see that the ACF is now, for the most part, the ACF stays within the 95% confidence interval (the blue dotted lines). These are the plots we got after second-order differencing. After differencing once, the data was still not stationary, so we had to difference again. We will now try to use the SARIMA model on the new transformed, stationary data so that we can forecast future HPI values.

```
auto_fit <- auto.arima(log_data, seasonal =TRUE)

print(auto_fit)
```

```
Series: log_data

ARIMA(1,0,2)(2,0,0)[4] with zero mean


Coefficients:

          ar1     ma1      ma2    sar1     sar2

      -0.5816  0.0723  -0.5814  0.3062  -0.0831

s.e.   0.1778  0.1619   0.0975  0.0822   0.0841


sigma^2 = 0.0001063:  log likelihood = 614.37

AIC=-1216.74   AICc=-1216.29   BIC=-1197.13
```

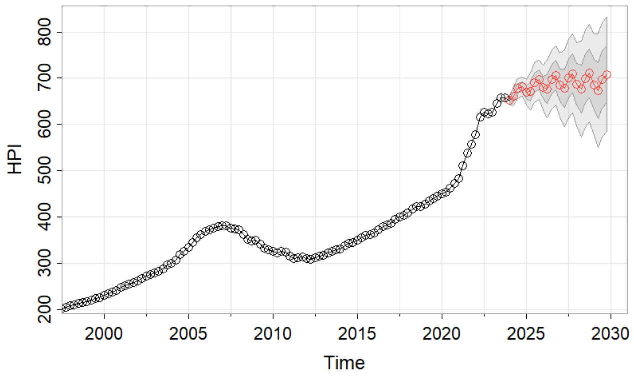
For the seasonal component, P = 2, D = 0, and Q = 0. We will keep this in mind later when we do our forecasting.

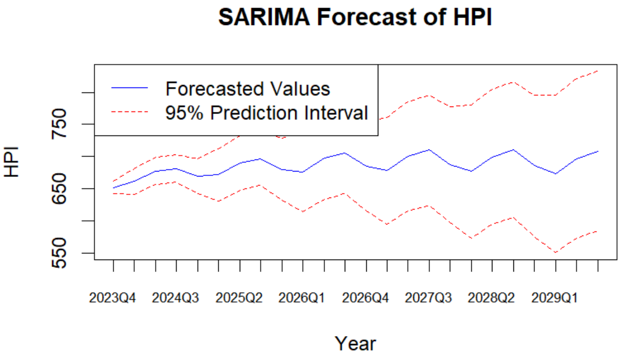
```
sarima(log_data, 1, 0, 0) # AR(1)

 Coefficients:

       Estimate     SE t.value p.value

 ar1    -0.2877 0.0691 -4.1663   0.000

 xmean   0.0000 0.0007 -0.0653   0.948


 sigma^2 estimated as 0.0001374782 on 192 degrees of freedom


 AIC = -6.022795  AICc = -6.022471  BIC = -5.972261
```

```
sarima(log_data, 0, 0, 2) # MA(2)

Coefficients:

      Estimate     SE t.value p.value

ma1    -0.4539 0.0791 -5.7406  0.0000

ma2    -0.2152 0.0839 -2.5648  0.0111

xmean   0.0000 0.0003 -0.0526  0.9581


sigma^2 estimated as 0.0001130947 on 191 degrees of freedom


AIC = -6.205582  AICc = -6.204931  BIC = -6.138204
```

This model has a lower BIC and AIC than the AR(1) model, so we will use the MA(2) model instead.

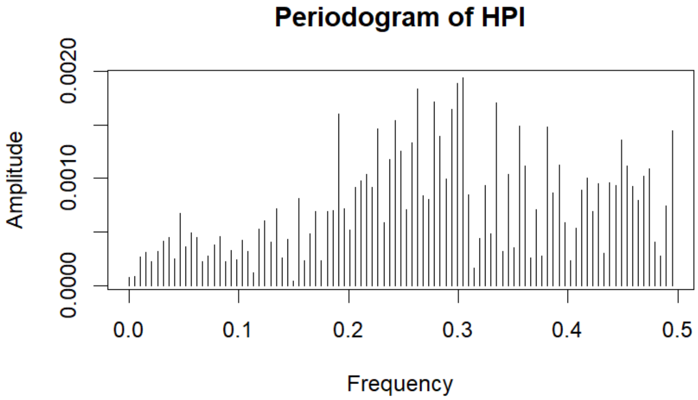
Forecasting





Unless there is an economic recession in the near future, we should expect at least a modest increase in the HPI (of around 8-9 percent) over the next six years.

Spectral Analysis



There are clear peaks around 0.2 and 0.3. This corresponds to a period of 5 quarters and 3.33 quarters, respectfully, as 1/0.2 = 5, and 1/0.3 = 3.33. This translates to around 10 to 15 months. This implies the existence of yearly cyclical trends in the House Price Index data. There are many different possible reasons for seasonal trends in house prices. In the spring and summer, for example, there are more listings. This is partly because some families buy homes during this period so that their children can be enrolled in their new schools before the school year begins.

Conclusion

As house prices in America stay increasing, many people have found it increasingly difficult to live out what many call “The American Dream.” It would be wise for young Americans to accept the plight at hand, and plan accordingly for the future. The results of this study can help to do just that. The Box-Jenkins approach helped us apply SARIMA modeling to forecast the future possible values of the HPI for the next several years. This is personally relevant to my life, given my hope of becoming a homeowner by the time I am 30 in 2030. We chose the model with the lowest BIC and AIC, and the model predicted a modest increase in the HPI in the next six years. Our spectral analysis revealed a 10 to 15 month cycle on house prices. We can say this is more or less a yearly cycle, and the seasonal nature of the housing market confirms this.

References

"All-Transactions House Price Index for the United States." *FRED*, 28 May 2024, fred.stlouisfed.org/series/USSTHPI.

Boykin, Ryan. "How Seasons Impact Real Estate Investments." *Investopedia*, Investopedia, www.investopedia.com/articles/investing/010717/seasons-impact-real-estate-more-you-think.asp. Accessed 13 June 2024.

"Housing Prices and Inflation." *The White House*, The United States Government, 30 Nov. 2021, www.whitehouse.gov/cea/written-materials/2021/09/09/housing-prices-and-inflation/.

Liberto, Daniel. "Understanding the House Price Index (HPI) and How It Is Used." *Investopedia*, Investopedia, www.investopedia.com/terms/h/house-price-index-hpi.asp. Accessed 13 June 2024.

Appendix

All the code is down below.

Modeling and Forecasting the U.S. House Price Index

Patrick Omens

2024-05-29

```
set.seed(123)

getwd()
```

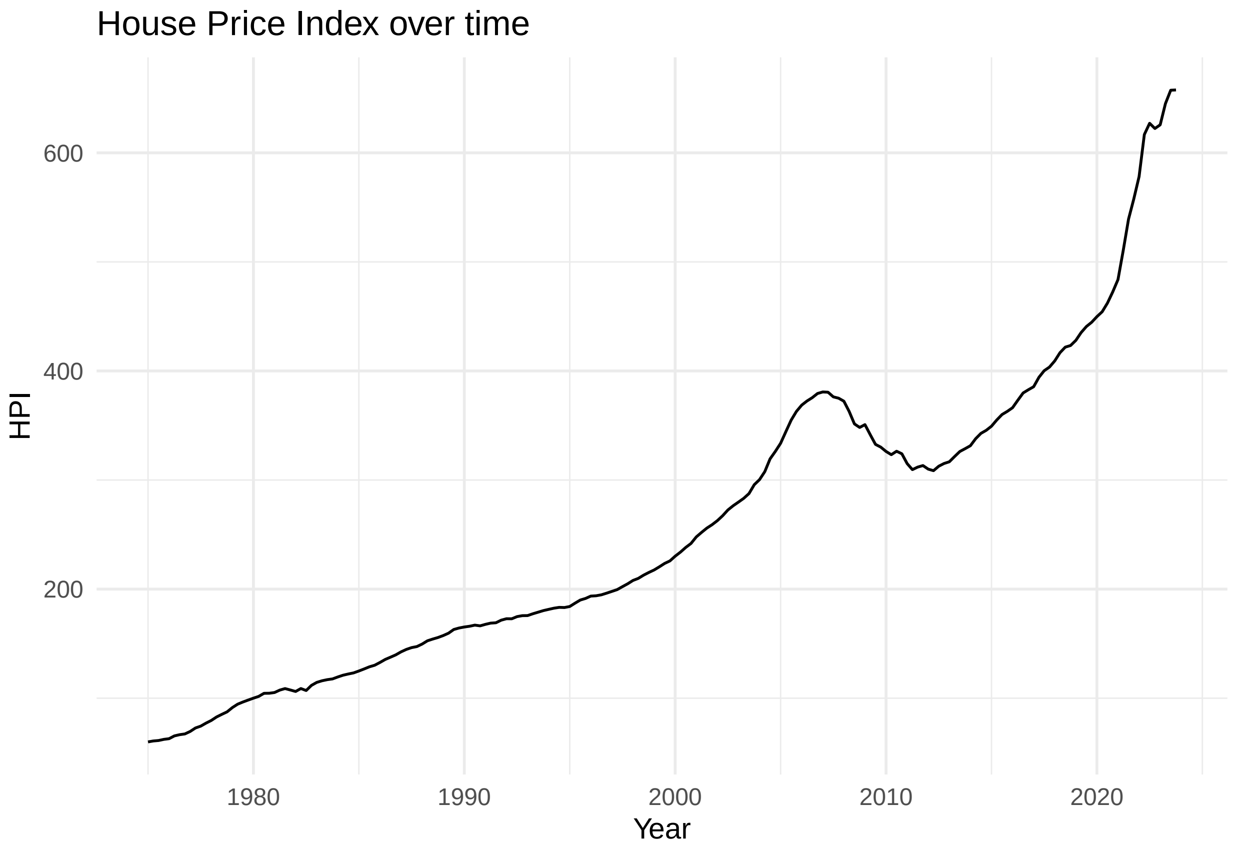
```
## [1] "C:/Users/Patrick Omens/Documents/R Studio files"
```

```
# setwd('C:/Users/Patrick Omens/R Studio files')

data = read.csv('USSTHPI.csv')



nrow(data)
```

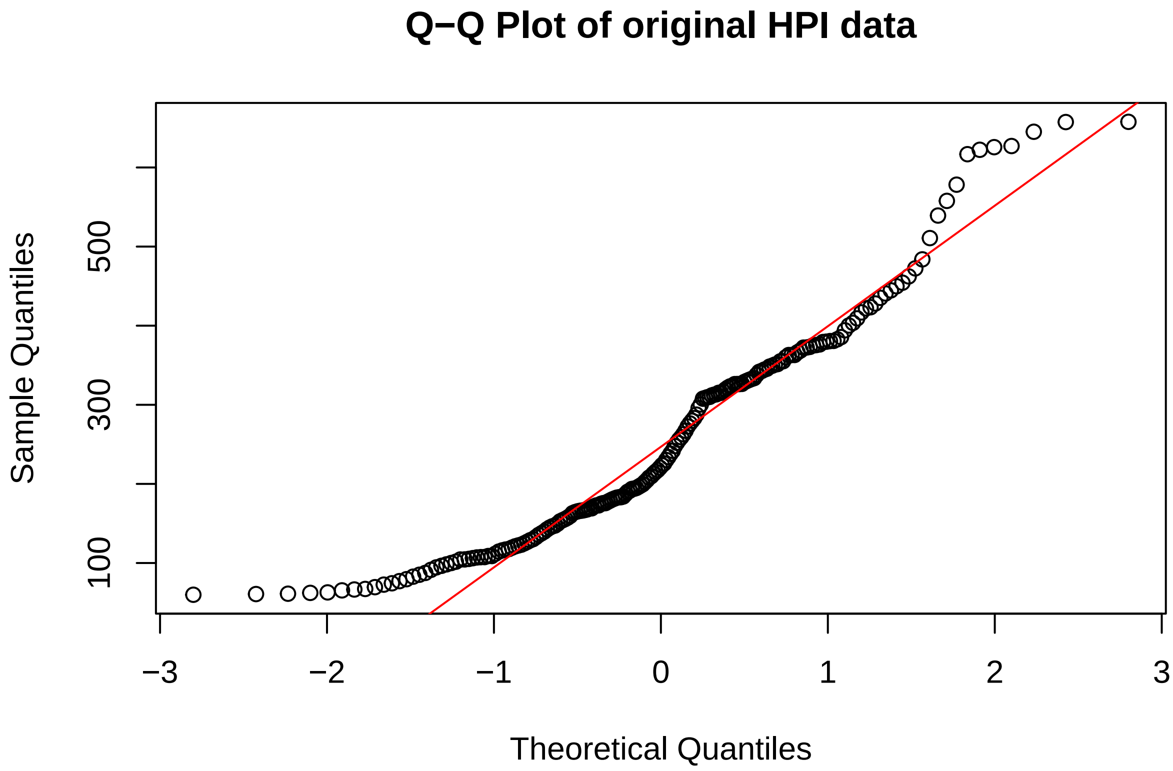
```
## [1] 196
```

```
ts_data = ts(data, start = c(1975, 1), frequency = 4)







library(ggplot2)







df <- data.frame(
  time = time(ts_data),
  value = as.numeric(ts_data[, 2])
)

# Plot using ggplot2
library(ggplot2)

ggplot(df, aes(x = time, y = value)) +
  geom_line() +
  labs(title = "House Price Index over time", x = "Year", y = "HPI") +
  theme_minimal()
```

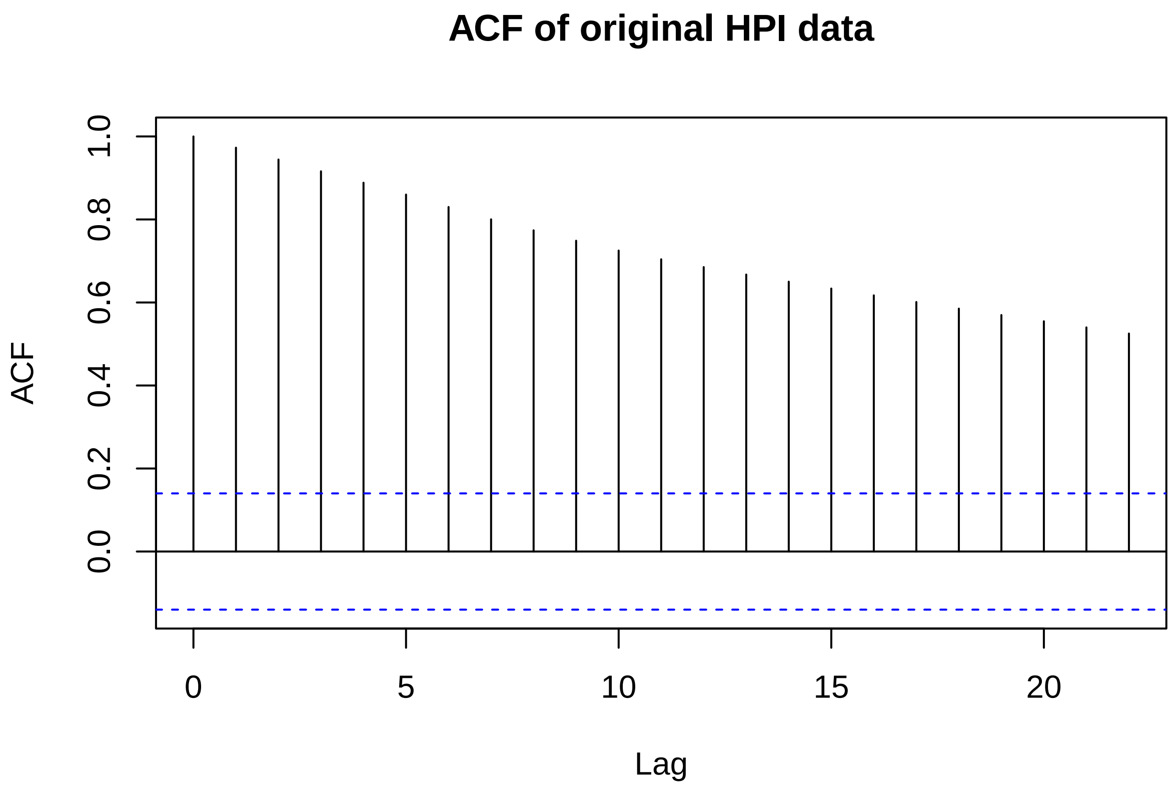
```
## Don't know how to automatically pick scale for object of type <ts>. Defaulting
## to continuous.
```



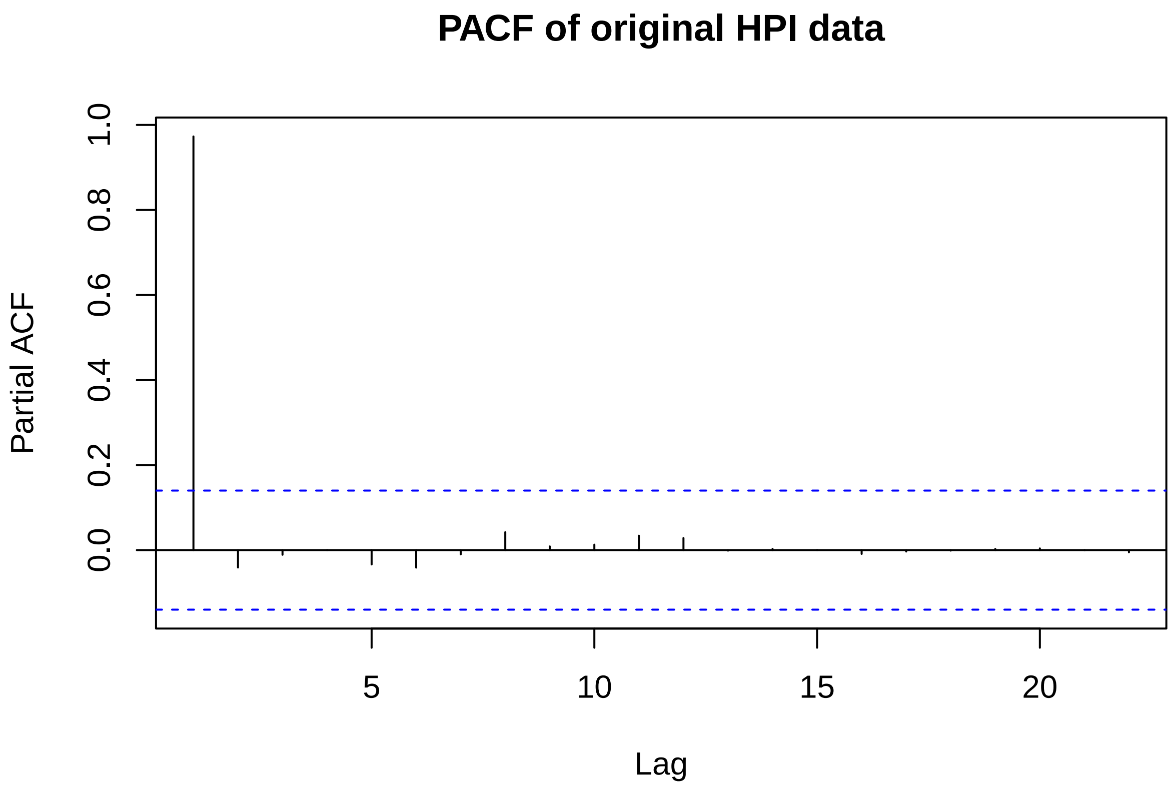
```r
qqnorm(df$value, main = "Q-Q Plot of original HPI data")
qqline(df$value, col = "red")
```



```r
acf(df$value, main = 'ACF of original HPI data')
```



```
pacf(df$value, main = 'PACF of original HPI data')
```



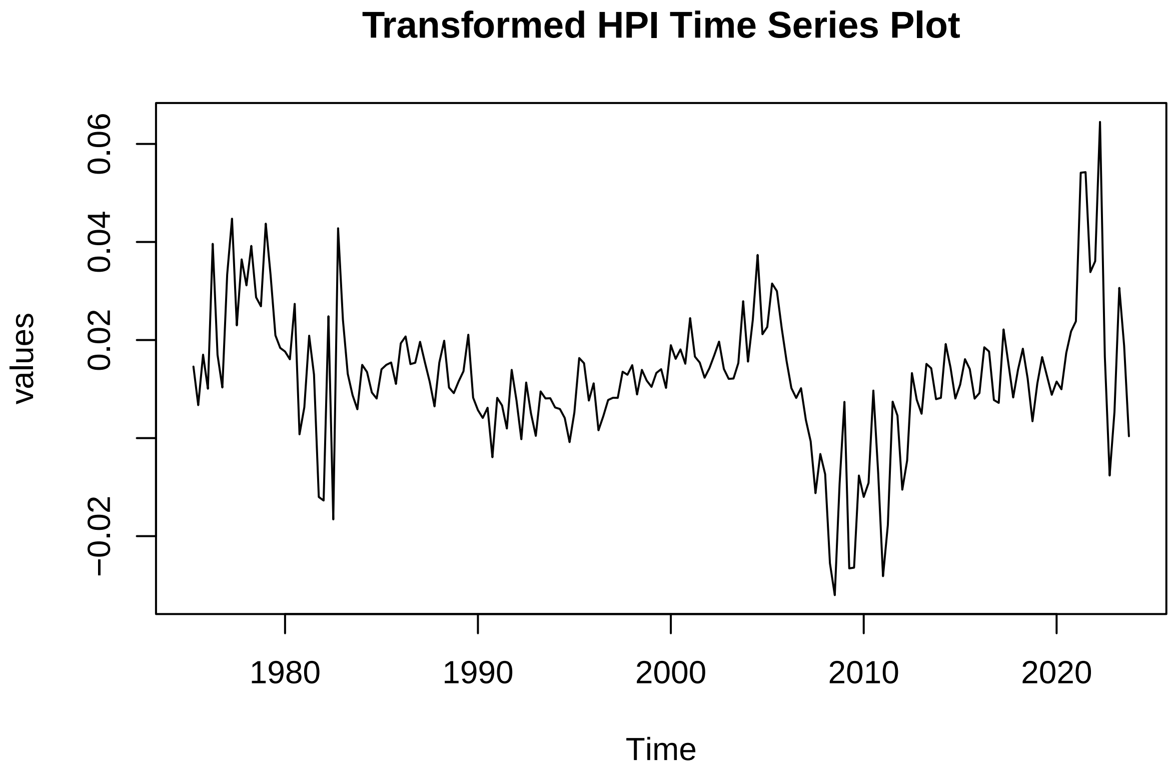
The ACF and PACF of the original time series are shown above. The ACF plot has a very slow decay. Because of this, we will perform differencing.

```
library(forecast)
```

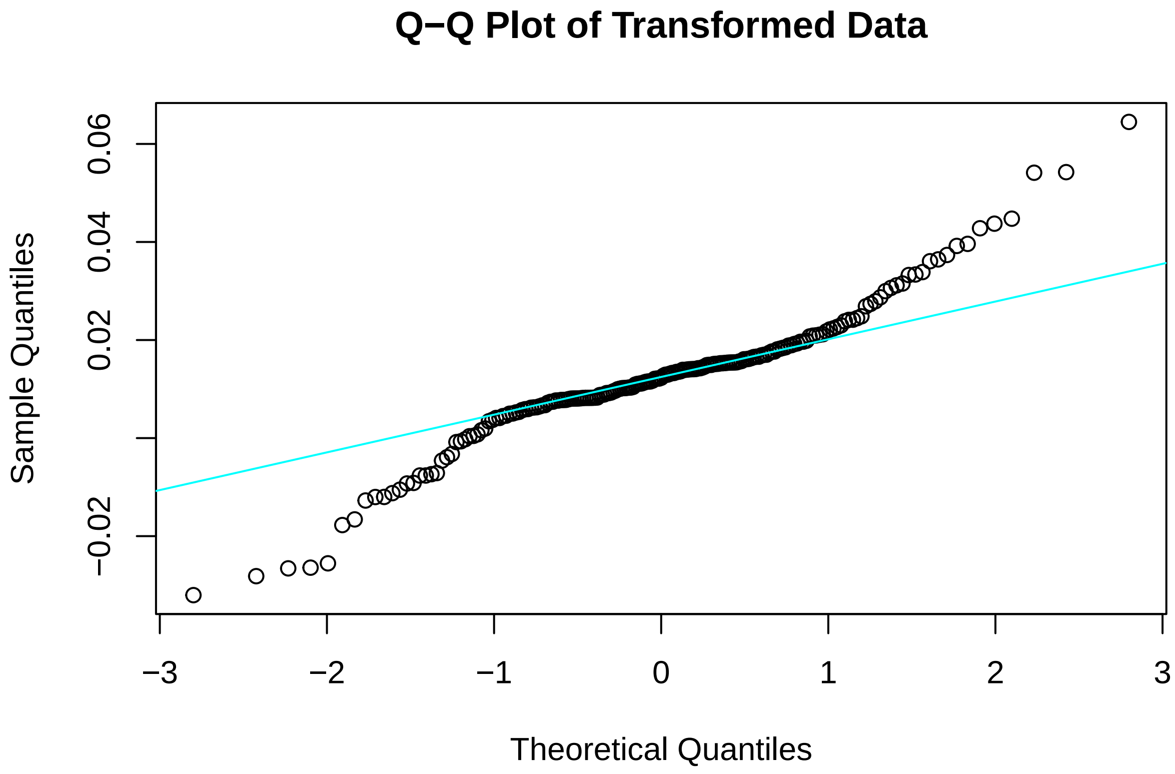
```
## Warning: package 'forecast' was built under R version 4.3.3
```

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

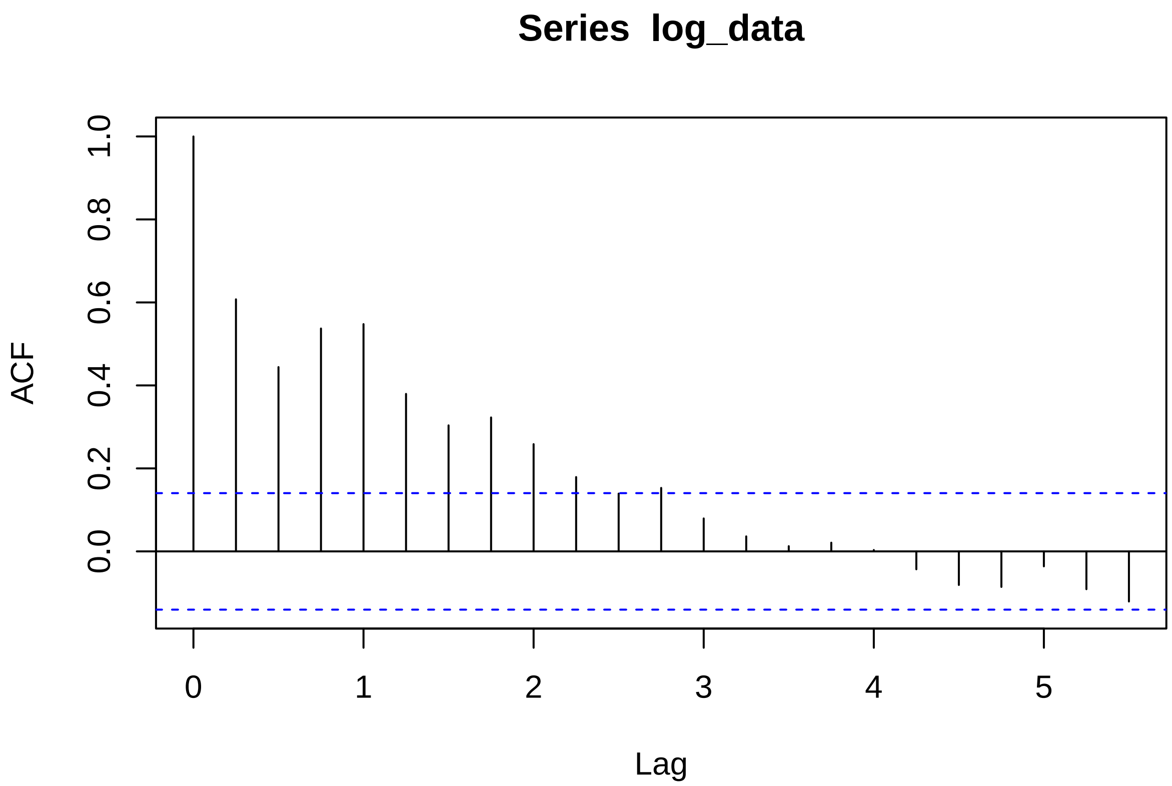
```
log_data = diff(log(ts_data[,2]))







plot(log_data, ylab = 'values', main = 'Transformed HPI Time Series Plot')
```



```r
qqnorm(log_data, main = 'Q-Q Plot of Transformed Data')
qqline(log_data, col = 'cyan')
```

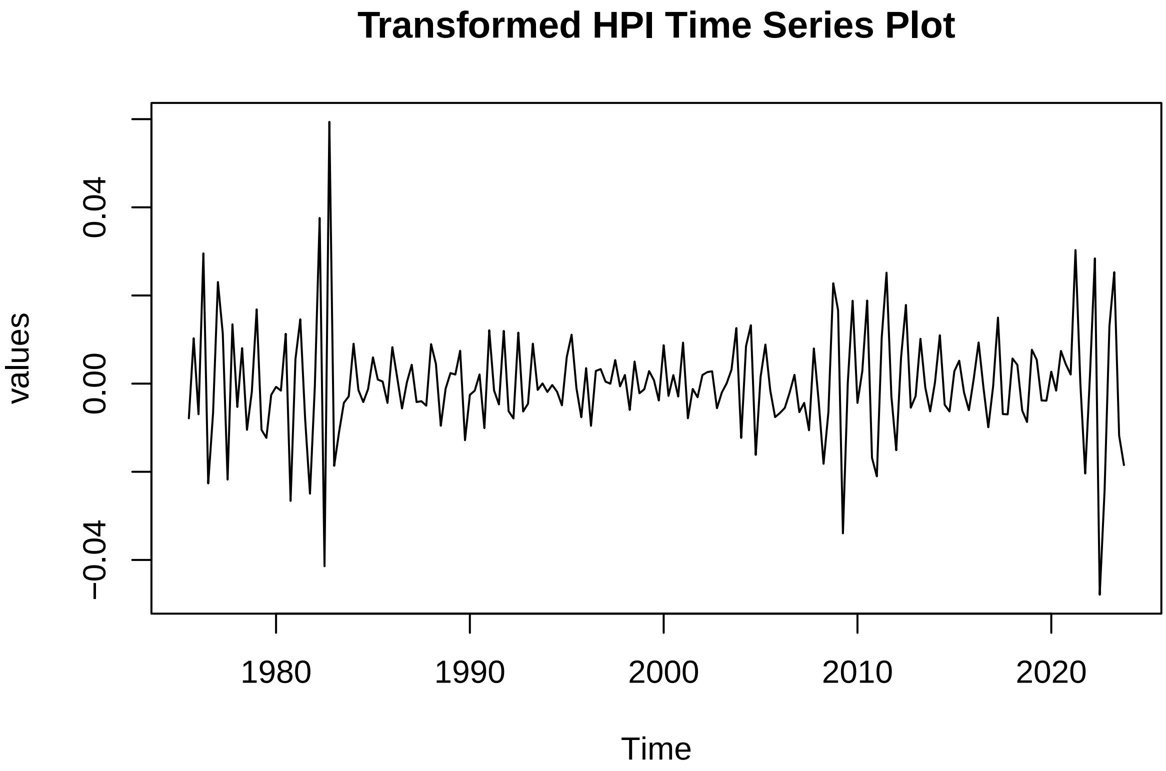

Q−Q Plot of Transformed Data
Sample Quantiles
−0.02
0.02
0.04
0.06
−3
−2
−1
0
1
2
3
Theoretical Quantiles

```
acf(log_data)
```

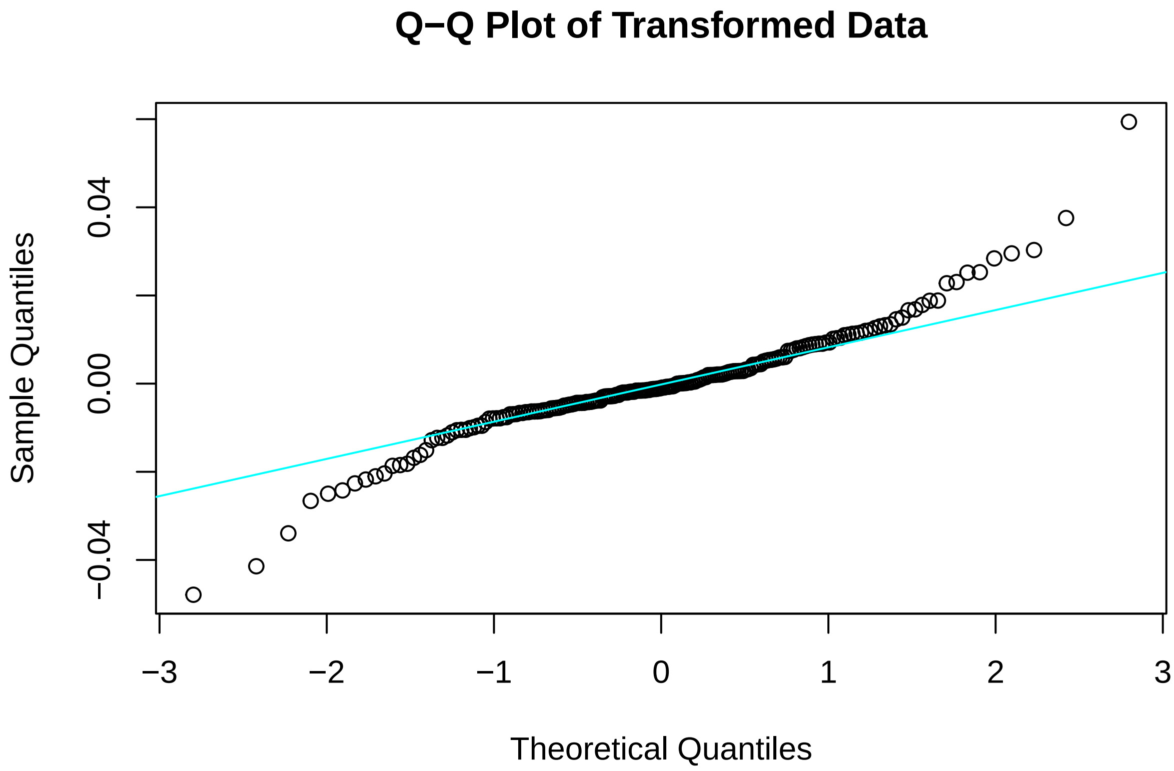


The ACF still has a very slow decay. We will instead perform second-order differencing.

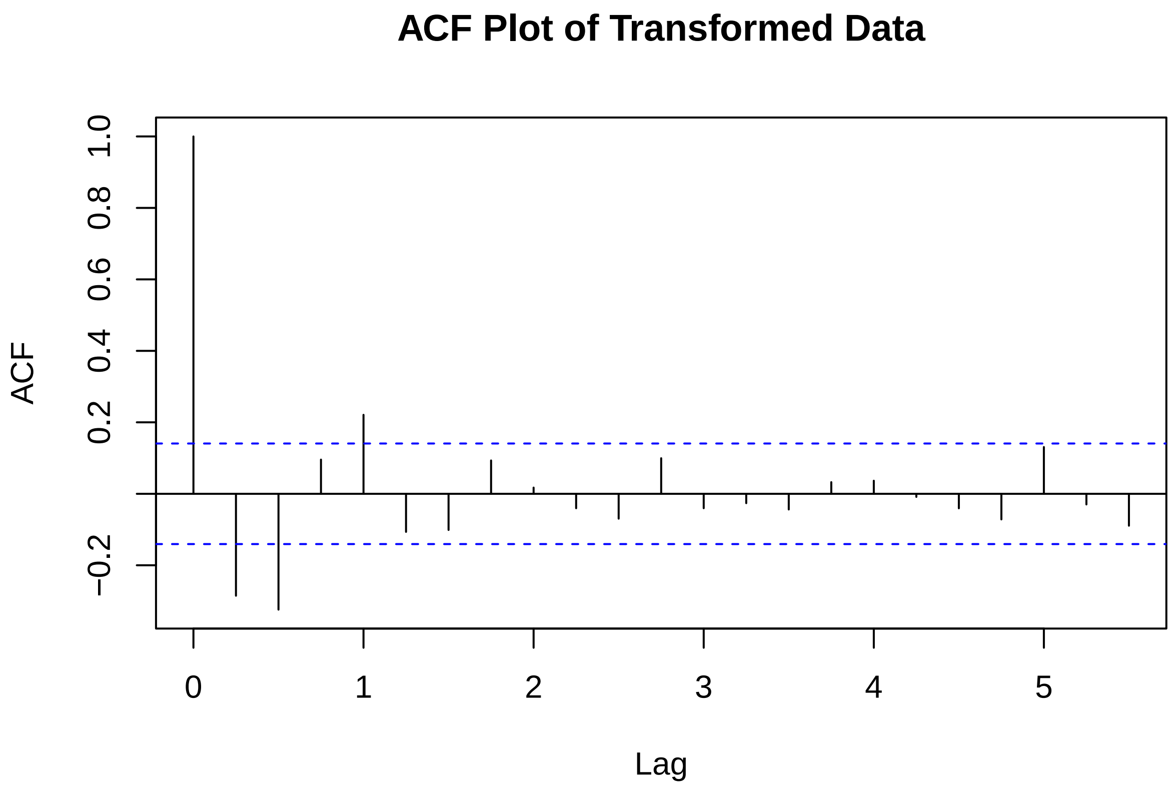
```
log_data = diff(log(ts_data[,2]), differences = 2)

plot(log_data, ylab = 'values', main = 'Transformed HPI Time Series Plot')
```



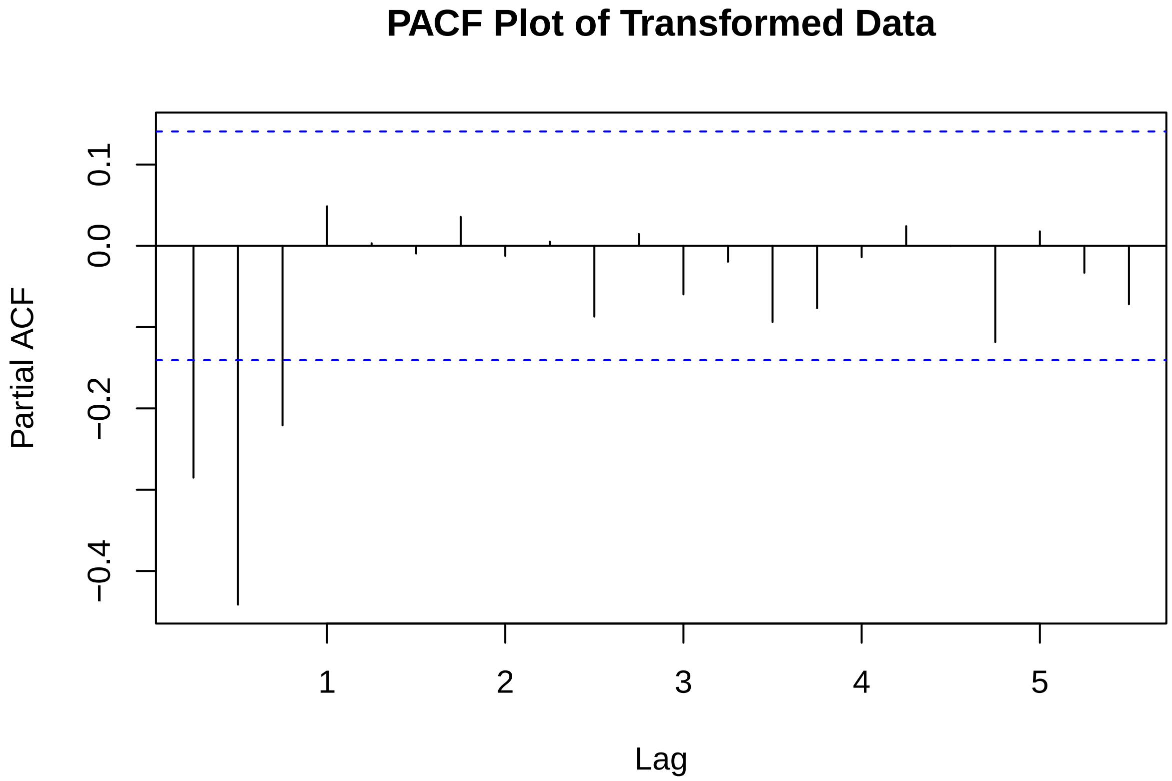
```
qqnorm(log_data, main = 'Q-Q Plot of Transformed Data')
qqline(log_data, col = 'cyan')
```



```r
acf(log_data, main = 'ACF Plot of Transformed Data')
```



```
pacf(log_data, main = 'PACF Plot of Transformed Data')
```



```
auto_fit <- auto.arima(log_data, seasonal = TRUE)
print(auto_fit)
```

```
## Series: log_data
## ARIMA(1,0,2)(2,0,0)[4] with zero mean
##
## Coefficients:
##           ar1     ma1      ma2    sar1     sar2
##       -0.5816  0.0723  -0.5814  0.3062  -0.0831
## s.e.   0.1778  0.1619   0.0975  0.0822   0.0841
##
## sigma^2 = 0.0001063:  log likelihood = 614.37
## AIC=-1216.74   AICc=-1216.29   BIC=-1197.13
```

For the seasonal component, P = 2, D = 0, Q = 0. We will keep this in mind later when we do our forecasting.

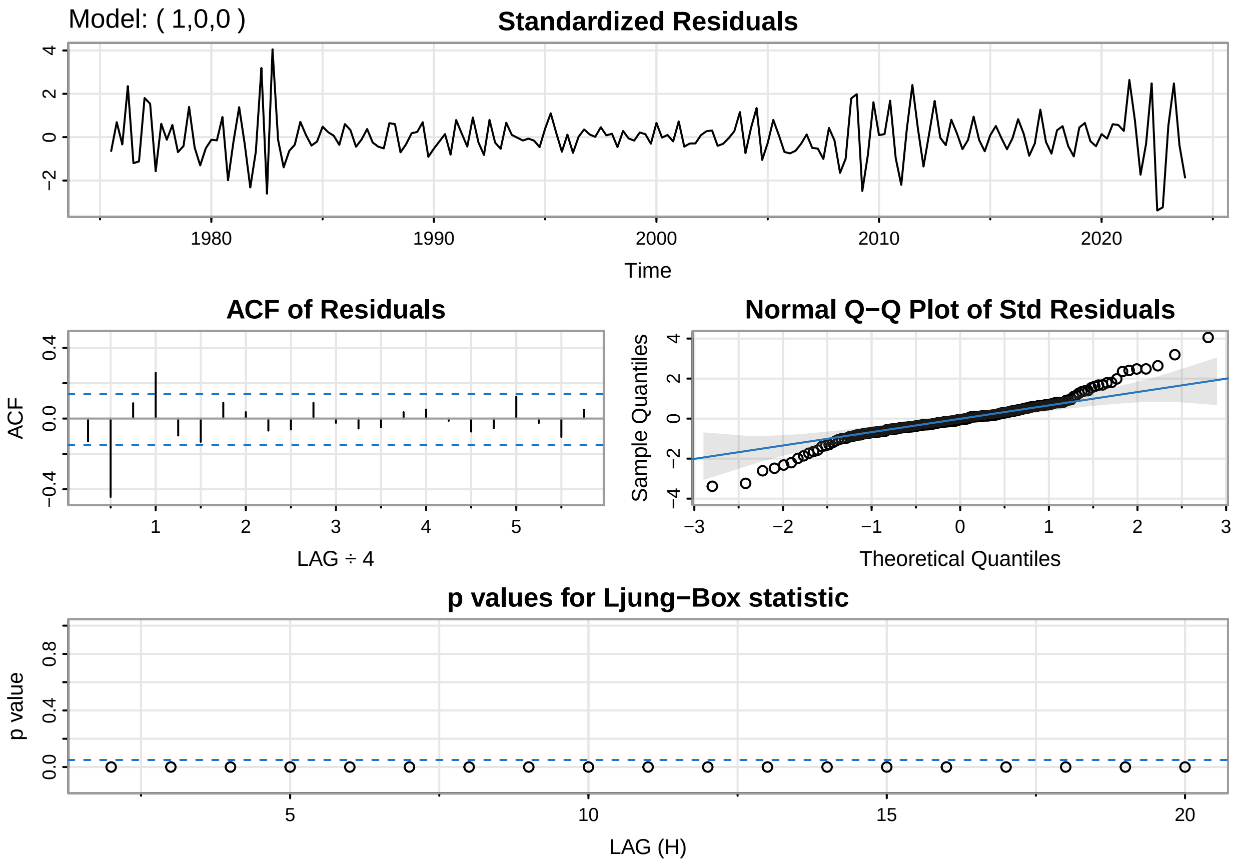
```
library(astsa)
```

```
## Warning: package 'astsa' was built under R version 4.3.2
```

```
##
## Attaching package: 'astsa'
## The following object is masked from ’package:forecast’:
##
##     gas
```

```
sarima(log_data, 1, 0, 0)
```

```
## initial  value -4.401437
## iter   2 value -4.444485
## iter   3 value -4.444495
## iter   3 value -4.444495
## iter   3 value -4.444495
## final  value -4.444495
## converged
## initial  value -4.445797
## iter   2 value -4.445800
## iter   2 value -4.445800
## iter   2 value -4.445800
## final  value -4.445800
## converged
## <><><><><><><><><><><><><><>
##
## Coefficients:
##       Estimate     SE t.value p.value
## ar1    -0.2877 0.0691 -4.1663   0.000
## xmean   0.0000 0.0007 -0.0653   0.948
##
## sigma^2 estimated as 0.0001374782 on 192 degrees of freedom
##
## AIC = -6.022795  AICc = -6.022471  BIC = -5.972261
##
```



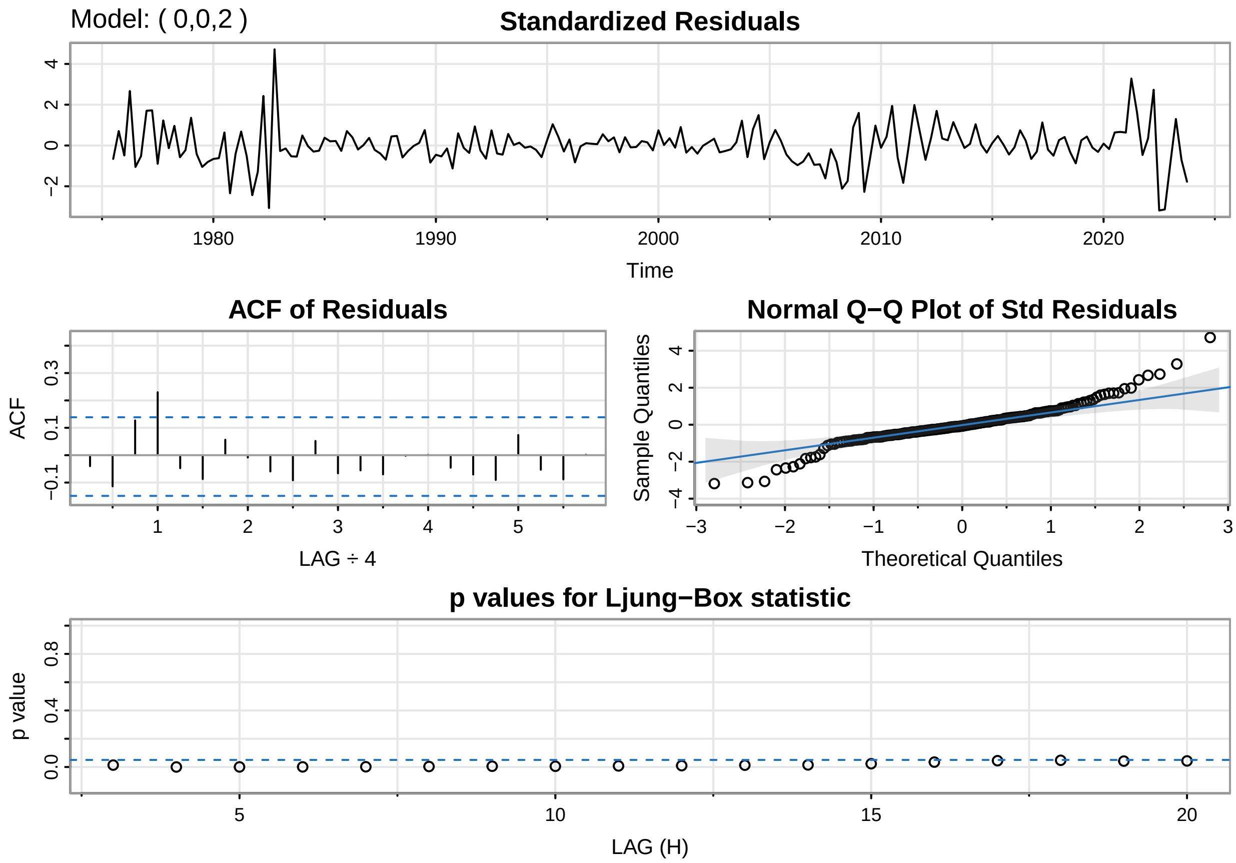
The AIC for the AR(1) model is -6.023, and the BIC is -5.972.

```
sarimaMod = arima(log_data, order = c(1, 0, 0),
                  seasonal = list(order = c(0, 0, 0)))

sarimaMod
```

```
##
## Call:
## arima(x = log_data, order = c(1, 0, 0), seasonal = list(order = c(0, 0, 0)))
##
## Coefficients:
##           ar1  intercept
##       -0.2877      0e+00
## s.e.   0.0691      7e-04
##
## sigma^2 estimated as 0.0001375:  log likelihood = 587.21,  aic = -1168.42
```

```
sarima(log_data, 0, 0, 2)
```

```
## initial  value -4.402976
## iter   2 value -4.529355
## iter   3 value -4.535399
## iter   4 value -4.540436
## iter   5 value -4.542984
## iter   6 value -4.543331
## iter   7 value -4.543356
## iter   7 value -4.543356
## iter   7 value -4.543356
## final  value -4.543356
## converged
## initial  value -4.542344
## iter   2 value -4.542346
## iter   3 value -4.542347
## iter   4 value -4.542348
## iter   4 value -4.542348
## iter   4 value -4.542348
## final  value -4.542348
## converged
## <><><><><><><><><><><><><><>
##
## Coefficients:
##       Estimate     SE t.value p.value
## ma1    -0.4539 0.0791 -5.7406  0.0000
## ma2    -0.2152 0.0839 -2.5648  0.0111
## xmean   0.0000 0.0003 -0.0526  0.9581
##
## sigma^2 estimated as 0.0001130947 on 191 degrees of freedom
##
## AIC = -6.205582  AICc = -6.204931  BIC = -6.138204
##
```



The AIC for the MA(2) model is -6.206, and the BIC is -6.138.

This model has a lower BIC and AIC than the AR(1) model, so we will use the MA(2) model instead.

```
sarimaMod2 = arima(log_data, order = c(0, 0, 2), seasonal =
                     list(order = c(0, 0, 0)))
sarimaMod2
```

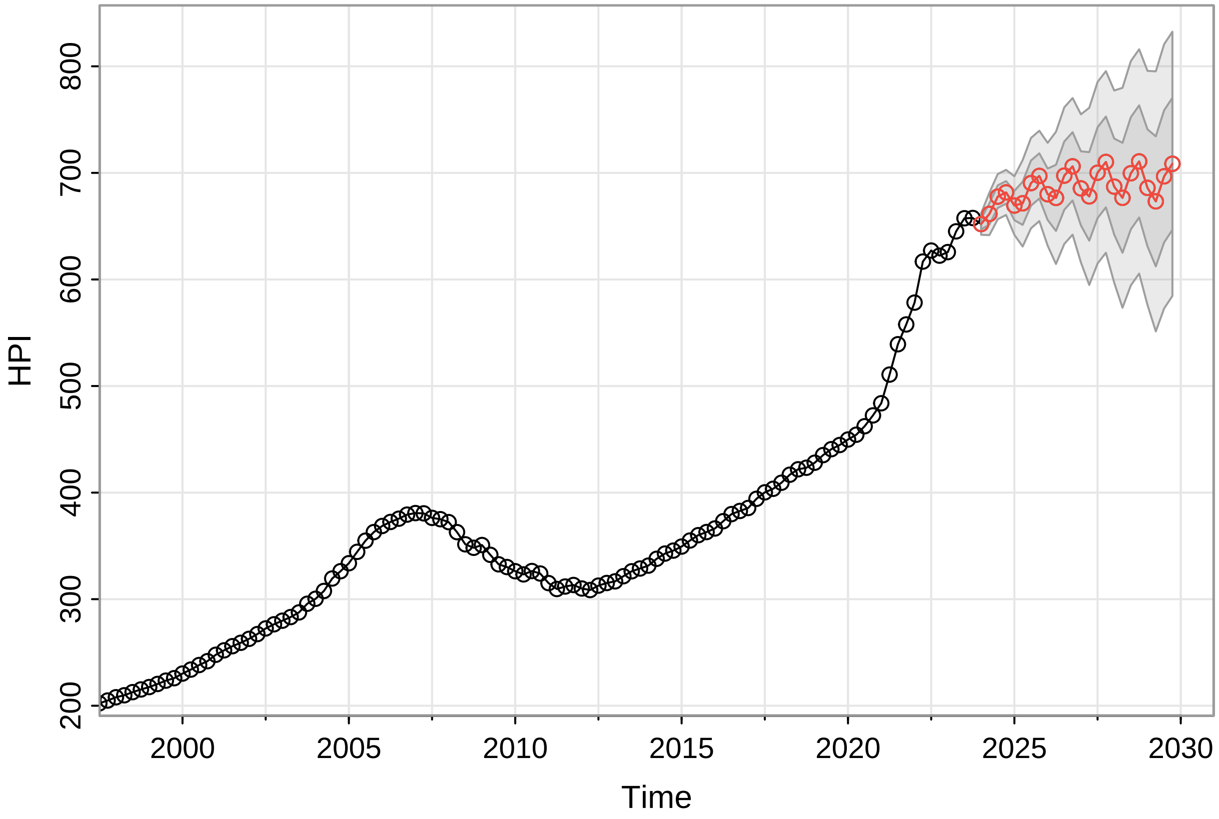
```
##
## Call:
## arima(x = log_data, order = c(0, 0, 2), seasonal = list(order = c(0, 0, 0)))
##
## Coefficients:
##           ma1      ma2  intercept
##       -0.4539  -0.2152      0e+00
## s.e.   0.0791   0.0839      3e-04
##
## sigma^2 estimated as 0.0001131:  log likelihood = 605.94,  aic = -1203.88
```

```
HPI = ts_data[,2]
```

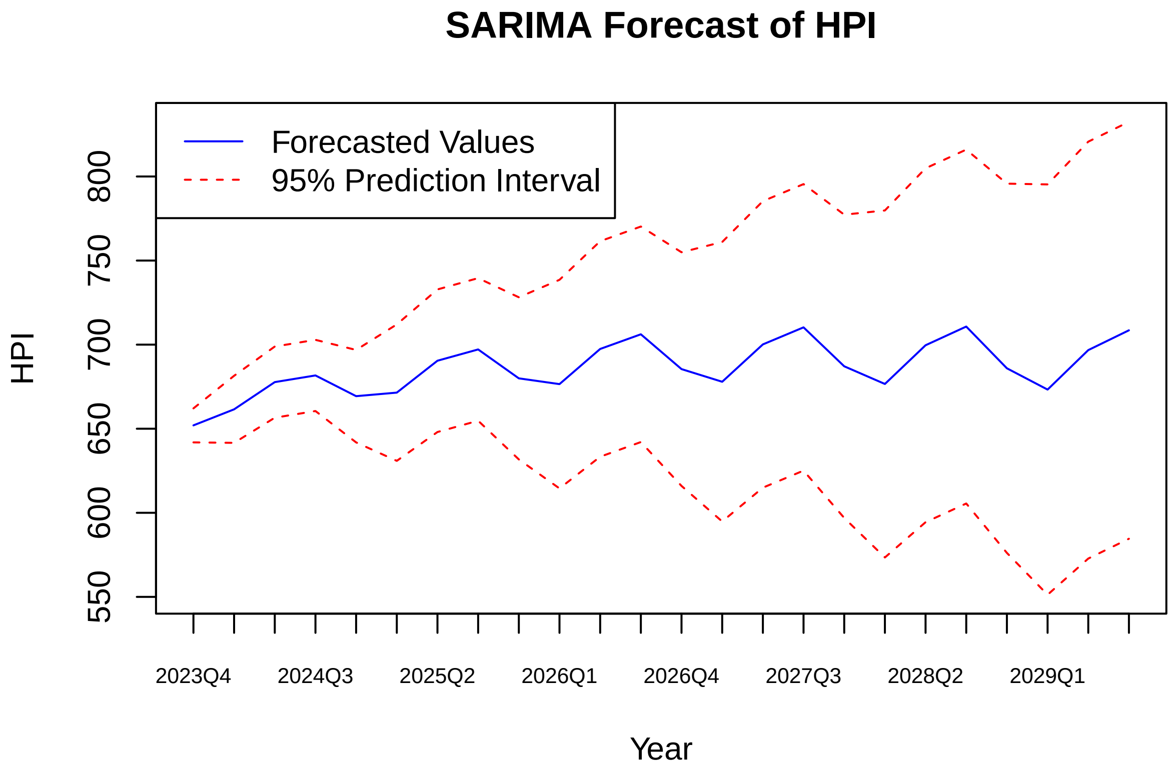
```
sarima.for(HPI, n.ahead = 24, p = 0, d = 0, q = 2, P = 2, D = 0,
           Q = 0, S = 4)
```

```
## $pred
##          Qtr1     Qtr2     Qtr3     Qtr4
## 2024 651.9957 661.5374 677.7175 681.6957
## 2025 669.3623 671.4696 690.4370 697.1581
## 2026 679.9613 676.5303 697.4393 706.1526
## 2027 685.4595 677.9369 700.1459 710.2799
## 2028 687.1302 676.6189 699.6422 710.7626
## 2029 685.9457 673.2855 696.7567 708.5342
##
## $se
##           Qtr1      Qtr2      Qtr3      Qtr4
## 2024  5.056942  9.969573 10.578745 10.578745
## 2025 13.747522 20.285747 21.195361 21.195361
## 2026 24.107479 31.016129 32.040238 32.040238
## 2027 34.736540 41.549637 42.597002 42.597002
## 2028 45.082529 51.589969 52.613673 52.613673
## 2029 54.894876 61.003270 61.979391 61.979391
```

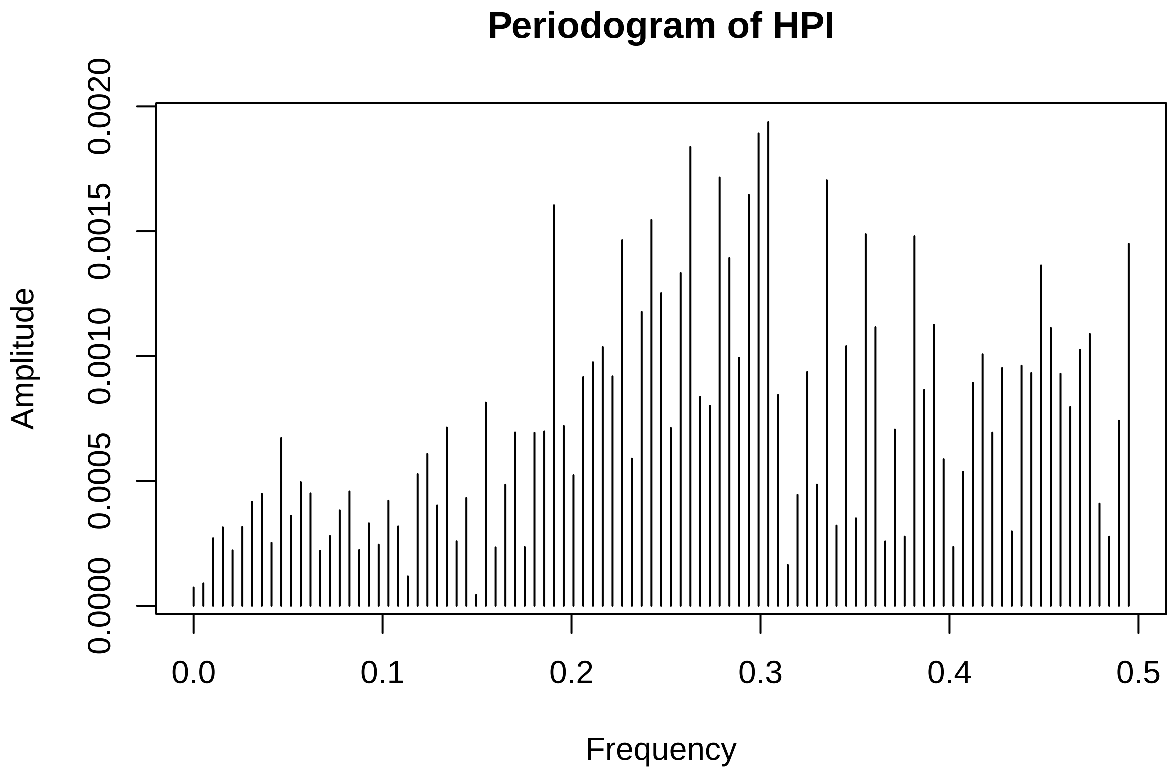
```
forecast = sarima.for(HPI, n.ahead = 24, p = 0, d = 0, q = 2, P = 2, D = 0,
           Q = 0, S = 4)
```



```
# Extract forecasted values and standard errors
forecasted_values <- forecast$pred
forecasted_se <- forecast$se

# Generate the time points for the forecast period
forecast_start <- end(ts_data[,2])[1] +
  (end(ts_data[,2])[2] - 1) / frequency(ts_data[,2])
forecast_time_points <- seq(forecast_start,
                            by = 1 / frequency(ts_data[,2]),
                            length.out = length(forecasted_values))

# Convert time points to year labels
years <- floor(forecast_time_points)
quarters <- (forecast_time_points - years) * 4 + 1
year_labels <- paste(years, "Q", quarters, sep = "")

# Plot the forecasted values
plot(forecast_time_points, forecasted_values, type = "l", col = "blue",
     ylim = range(c(forecasted_values + 2 * forecasted_se,
                    forecasted_values - 2 * forecasted_se)),
     xlab = "Year", ylab = "HPI", main = "SARIMA Forecast of HPI", xaxt = "n")

# Add custom x-axis with year labels
axis(1, at = forecast_time_points, labels = year_labels, cex.axis = 0.7)

# Add prediction intervals
lines(forecast_time_points, forecasted_values + 2 * forecasted_se,
      col = "red", lty = 2)
lines(forecast_time_points, forecasted_values - 2 * forecasted_se,
      col = "red", lty = 2)

# Add legend
legend("topleft", legend = c("Forecasted Values", "95% Prediction Interval"),
       col = c("blue", "red"), lty = c(1, 2))
```



```
# Spectral analysis

fftData = fft(log_data)

lenData = length(log_data)
ampData = Mod(fftData) / lenData

freqData = (0:(lenData - 1)) / lenData

plot(freqData[1:(lenData/2)], ampData[1:(lenData/2)], type = 'h',
     main = 'Periodogram of HPI',
     xlab = 'Frequency', ylab = 'Amplitude')
```


Periodogram of HPI
Amplitude
0.0000
0.0005
0.0010
0.0015
0.0020
0.0
0.1
0.2
0.3
0.4
0.5
Frequency