Econ 144 Project #1

Ragini Srinivasan, Jiarui Song, Siyu Xiong, Adriana Villavicencio

2023-10-17

I. INTRODUCTION

The data we chose was real U.S. GDP — quarterly, not seasonally adjusted, and chained to the 2017 USD. Specifically, the observations go from Quarter 1 of 2002 to Quarter 2 of 2023. We wanted to choose data that we knew would have both trend and seasonality. Indeed, the GDP of the U.S., as is consistent with most other countries, has a general upward trend over the years.

At the same time, there is clear seasonality in the data, with GDP being significantly low during the first quarter of each year, then steadily increasing each quarter and reaching a peak by the fourth. Studies indicate that this seasonality also appears in GDP growth, with the proposed concept of "residual seasonality". Residual seasonality is the idea that GDP growth is much slower in the first quarter than in subsequent quarters (see Section IV for the reference). Economists have been unable to ascertain whether this is a result of actual economic conditions or merely misreporting/miscalculations. In our project, we hoped to explore this concept, which has lasting implications for policymakers, and see if residual seasonality is present in not just GDP growth but also GDP itself.

II. RESULTS

Preliminary Work

```
# Load appropriate packages
tinytex::install_tinytex()
```

```
## The directory /usr/local/bin is not writable. I recommend that you make it writable. See https://gitl
```

```
## Warning: Please run this command in your Terminal (password required):
##   sudo chown -R 'whoami':admin /usr/local/bin
```

```
rm(list = ls())
library(dplyr)
```

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

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

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

```
library(tidyr)
library(readr)
library(ggplot2)
library(graphics)
library(ggfortify)
library(tseries)
```

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

```
library(lubridate)
```

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

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

```
library(dynlm)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

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

```
library(data.table)
```

```
##
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:lubridate':
##
##     hour, isoweek, mday, minute, month, quarter, second, wday, week,
##     yday, year
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```r
library(tsibble)
```

```
##
## Attaching package: 'tsibble'
```

```
## The following object is masked from 'package:data.table':
##
##     key
```

```
## The following object is masked from 'package:zoo':
##
##     index
```

```
## The following object is masked from 'package:lubridate':
##
##     interval
```

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

```r
library(forecast)
```

```
## Registered S3 methods overwritten by 'forecast':
##   method                 from
##   autoplot.Arima         ggfortify
##   autoplot.acf           ggfortify
##   autoplot.ar            ggfortify
##   autoplot.bats          ggfortify
##   autoplot.decomposed.ts ggfortify
##   autoplot.ets           ggfortify
##   autoplot.forecast      ggfortify
##   autoplot.stl           ggfortify
##   autoplot.ts            ggfortify
##   fitted.ar              ggfortify
##   fortify.ts             ggfortify
##   residuals.ar           ggfortify
```

```r
library(feasts)
```

```
## Loading required package: fabletools
```

```
##
## Attaching package: 'fabletools'
```

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

```r
library(tis)
```

```
##
## Attaching package: 'tis'

## The following object is masked from 'package:forecast':
##
##     easter

## The following objects are masked from 'package:data.table':
##
##     between, month, quarter, year

## The following objects are masked from 'package:lubridate':
##
##     day, hms, month, period, POSIXct, quarter, today, year, ymd

## The following object is masked from 'package:dplyr':
##
##     between
```

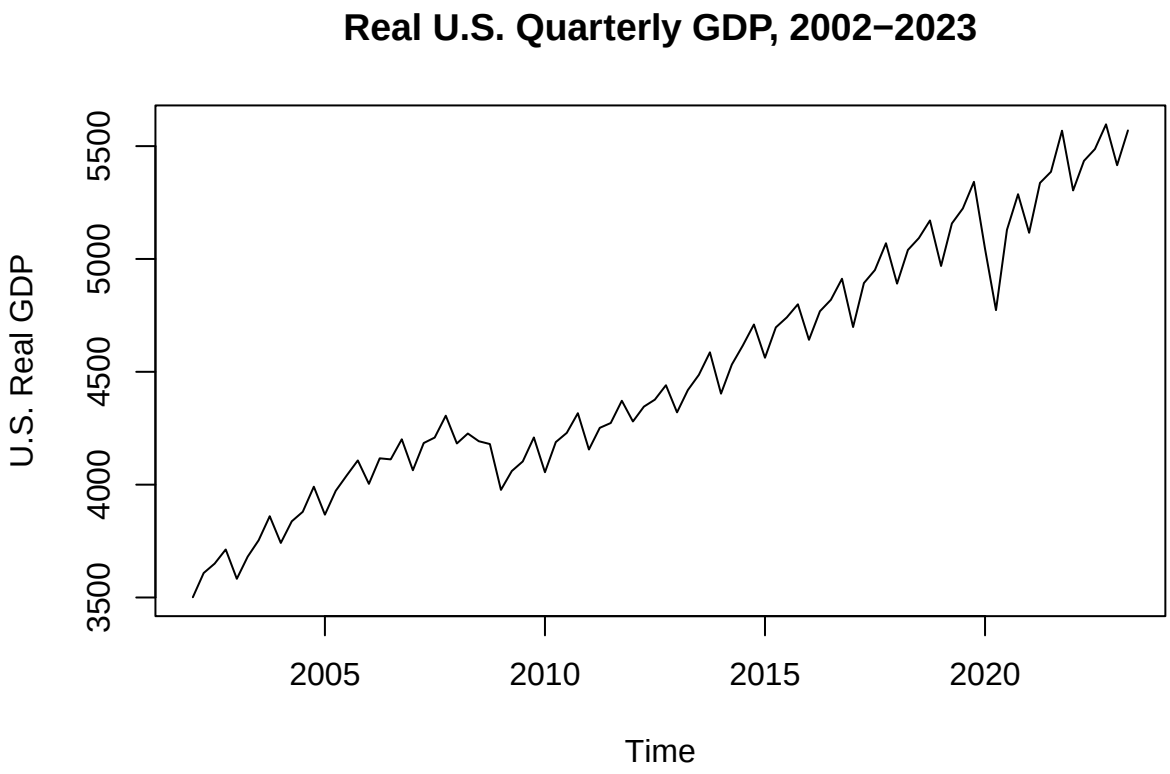
```r
# Download data on US GDP
gdp <- read_csv("/Users/raginisrinivasan/Desktop/Econ 144/Data/US_GDP.csv") # Data source in Section IV
```

```
## Rows: 86 Columns: 2

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl  (1): ND000334Q
## date (1): DATE
##
## 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
# Rename variables for clarity/concision
names(gdp) <- c("date", "gdp")
```

(a)

```r
# Plot time series of the data
plot(gdp$date, gdp$gdp, type = "l", main = "Real U.S. Quarterly GDP, 2002-2023", xlab = "Time", ylab =
```

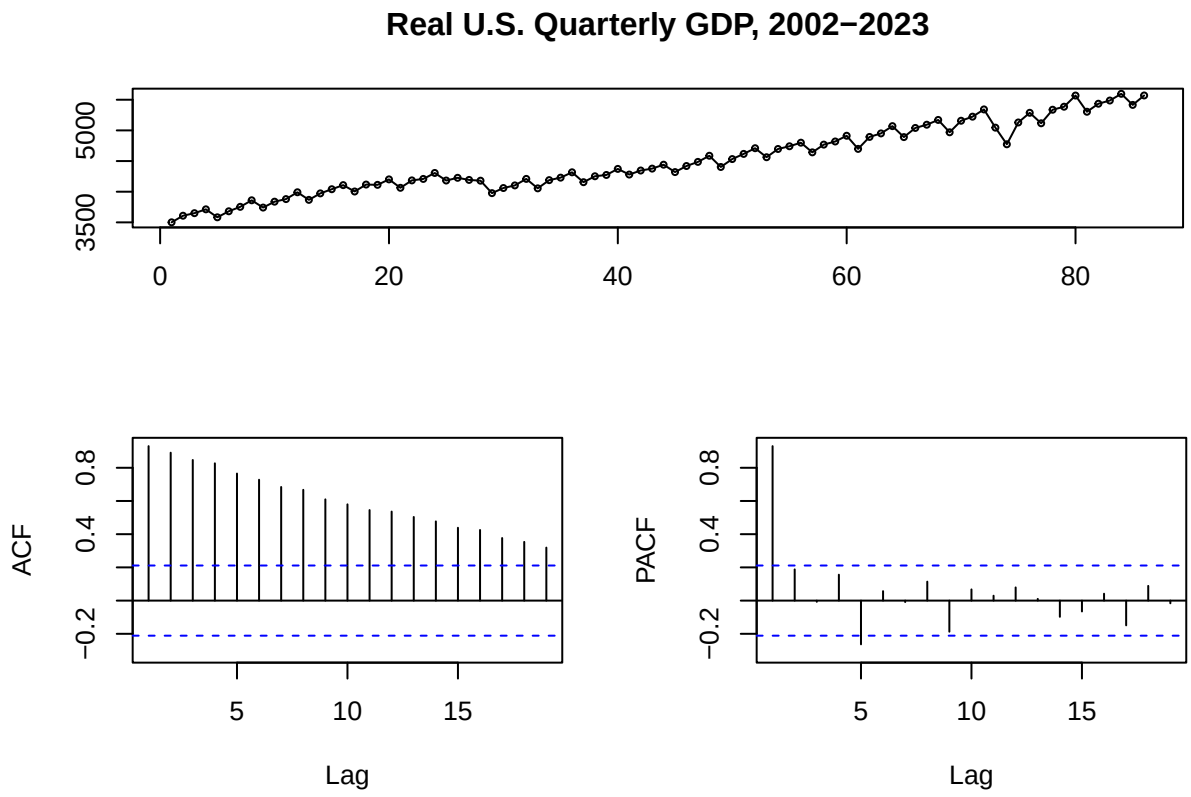


(b)

Our plot of the time series suggests that the data are not covariance stationary: the observations are steadily increasing over time, so each random variable does not have the same mean. Just with this knowledge, we can conclude that the there is no second-order weak stationarity.

(c)

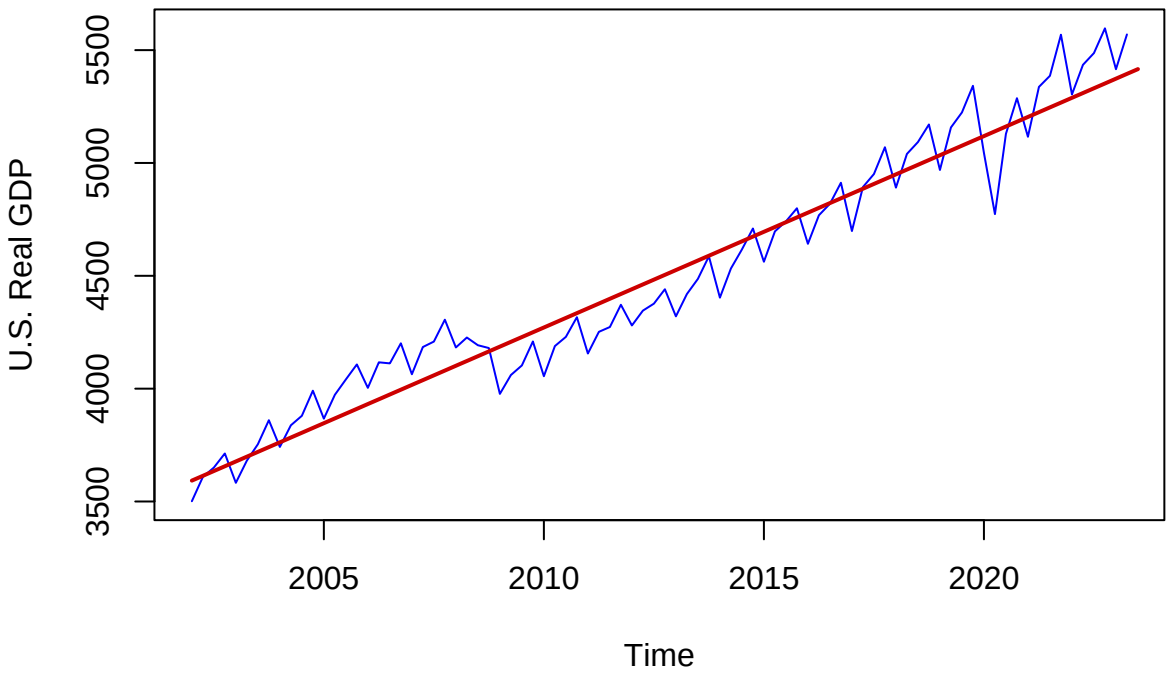
```
# Convert data into a tsibble
gdp_tsibble <- gdp %>%
  mutate(date = yearquarter(date)) %>%
  as_tibble(index = date)

# Compute the quarterly changes in GDP
gdp_tsibble <- gdp_tsibble %>%
  mutate(diff = difference(gdp))

# Plot the ACF and PACF
tsdisplay(gdp_tsibble$gdp, main = "Real U.S. Quarterly GDP, 2002-2023")
```



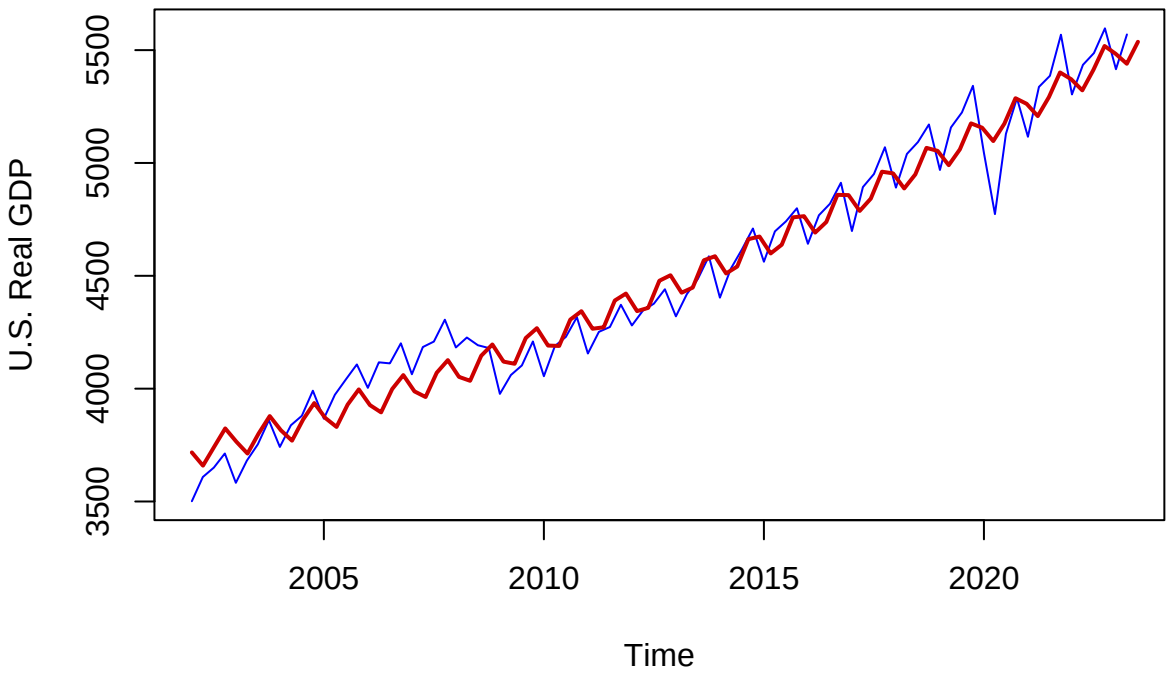
The ACF and PACF show greater autocorrelation in the data. In the ACF, we see that each observation, as a random variable, is strongly positively correlated with every single the previous random variables (or lags), dating back to a lag of 19 The PACF adds onto this by showing which autocorrelations are the strongest. We see spikes at lags of 5, 9, and (to a lesser extent) 17, and we can note that the figures are quarterly. Given that each observation strongly correlates with the values $4n$ quarters (i.e., n years) before it, we can conclude that there may be a seasonal component.

(d)

```r
# Create and plot a linear model
gdp_ts <- ts(gdp$gdp, start = 2002.00, frequency = 4)
t <- seq(2002.00, 2023.50, length = length(gdp_ts))
m1 <- lm(gdp_ts ~ t)
plot(gdp_ts, xlab = "Time", ylab = "U.S. Real GDP", col = "blue")
lines(t, m1$fit, col = "red3", lwd = 2)
```

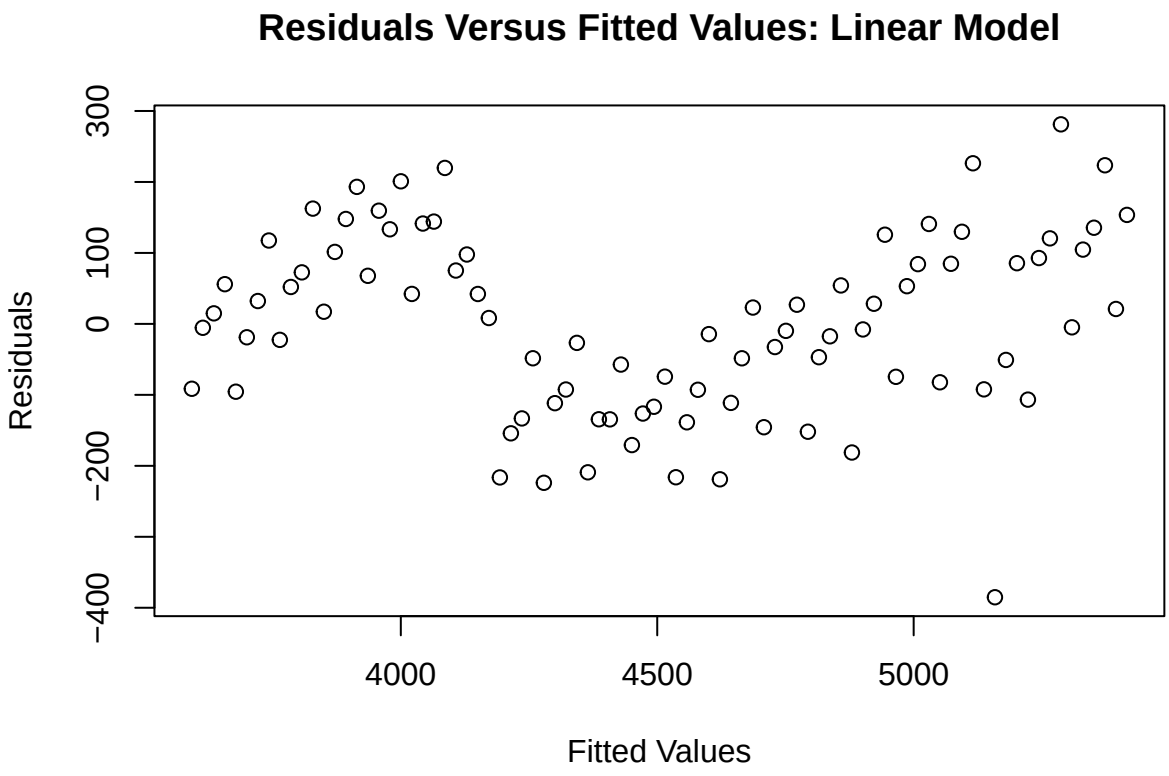



```
# Create and plot a nonlinear (quadratic + periodic) model
cost <- cos(2*pi*t)
sint <- sin(2*pi*t)
t2 <- t^2
m2 <- lm(gdp_ts ~ t + t2 + cost + sint)
plot(gdp_ts, xlab = "Time", ylab = "U.S. Real GDP", col = "blue")
lines(t, m2$fit, col = "red3", lwd = 2)
```

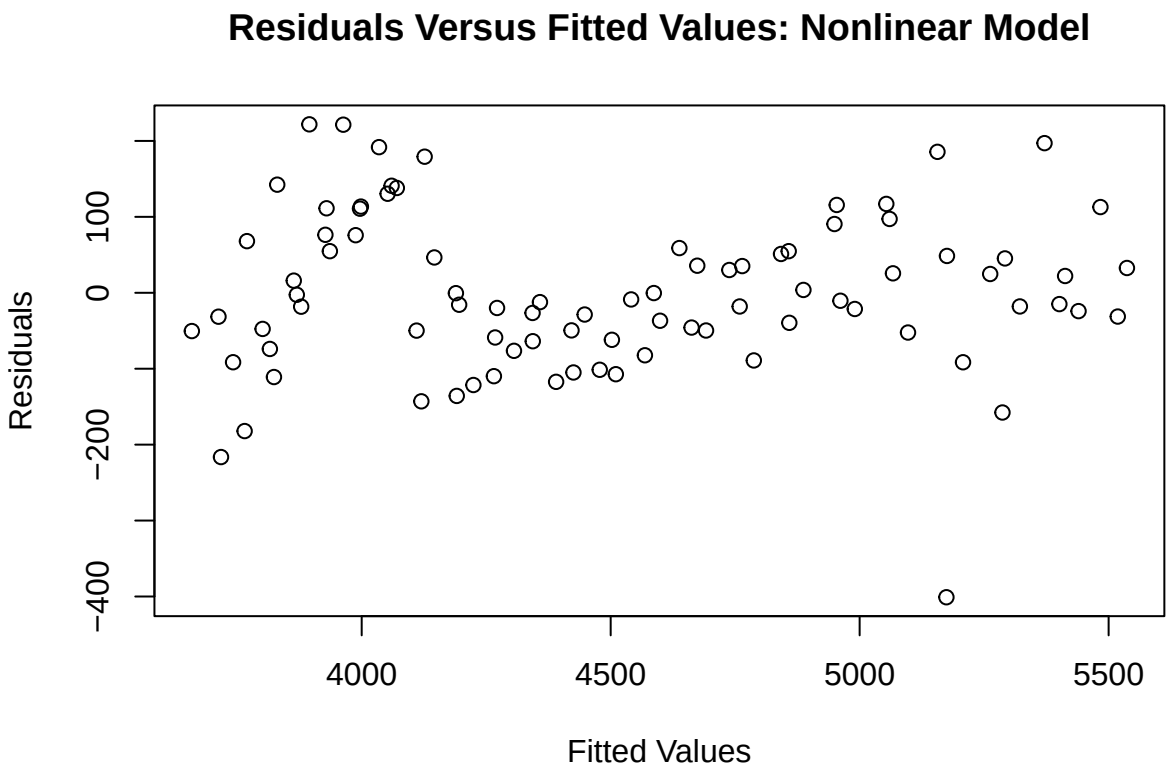


(e)

```
# Plot residuals of linear model
plot(m1$fit, m1$res, main = "Residuals Versus Fitted Values: Linear Model", xlab = "Fitted Values", ylab
```



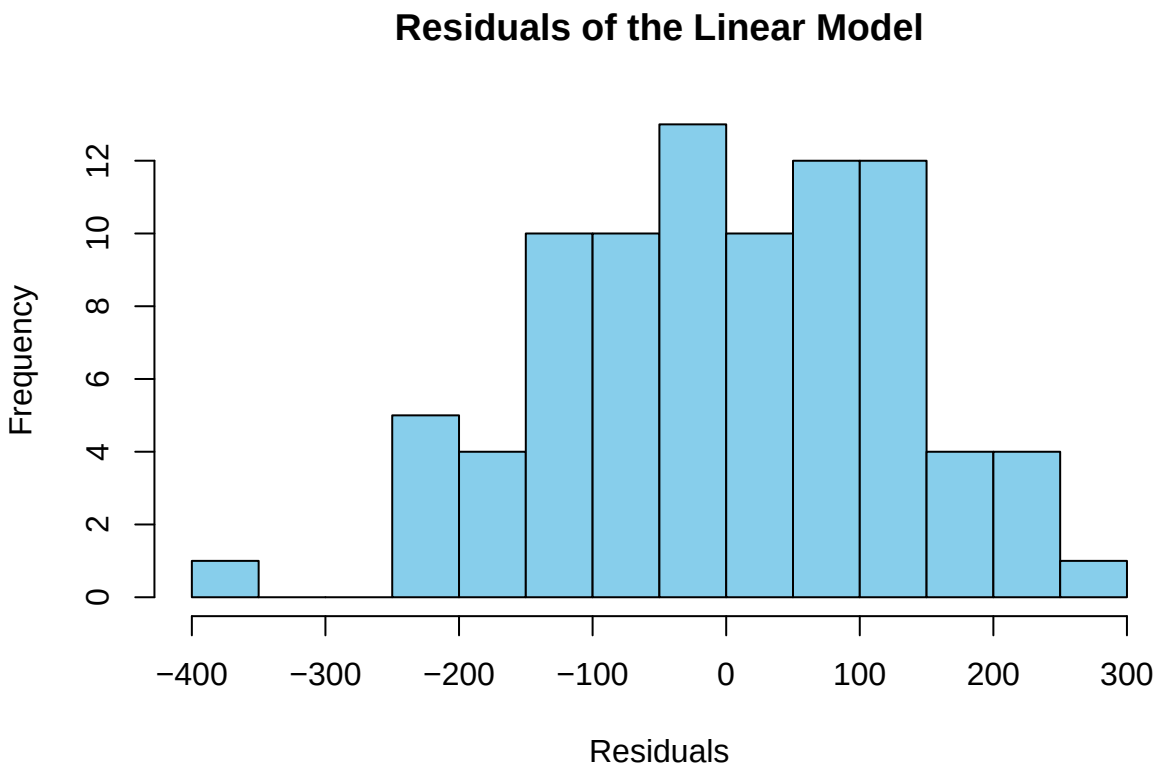

```r
# Plot residuals of nonlinear model
plot(m2$fit, m2$res, main = "Residuals Versus Fitted Values: Nonlinear Model", xlab = "Fitted Values", y
```



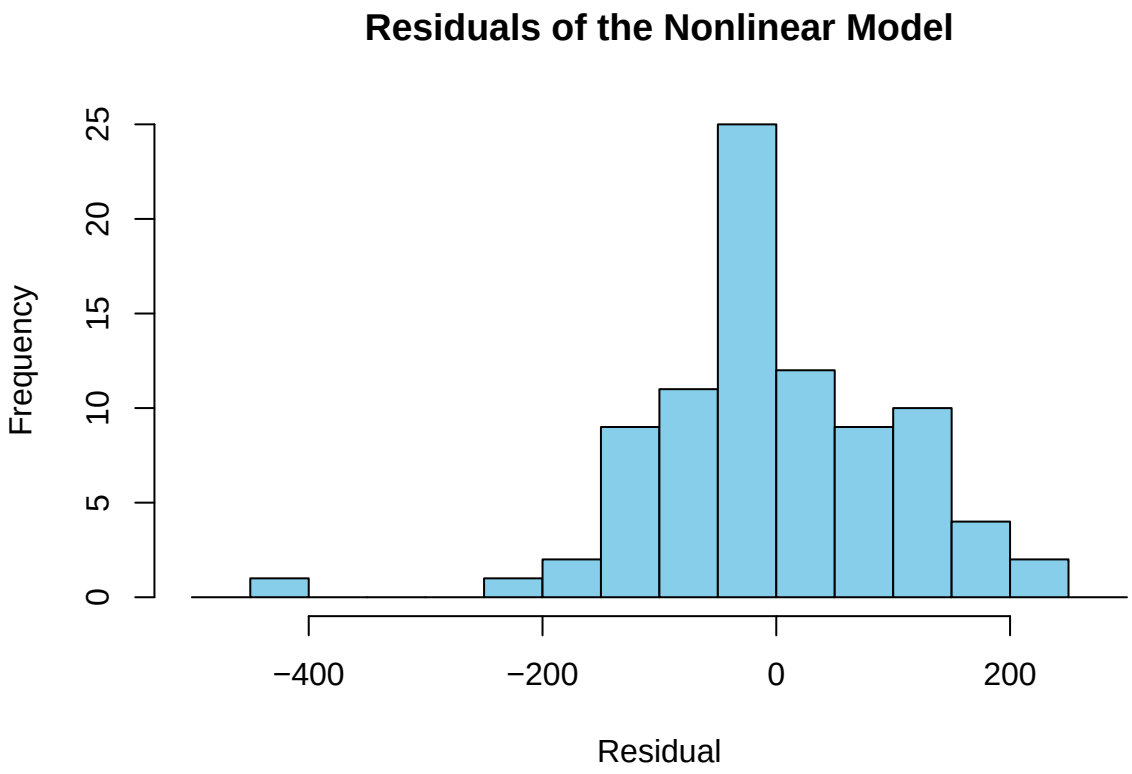
These residual plots generally show that the nonlinear model is a better fit for the time series than the linear model; the residuals are scattered more randomly in the former. In the latter, there is a strong pattern to the residuals, resembling a sinusoidal wave. While the nonlinear model does have a curved pattern in the residuals, it is much less prominent. This indicates that the quadratic + periodic model better fits the data.

(f)

```r
# Plot a histogram of the residuals for the linear model
hist(m1$residuals, main = "Residuals of the Linear Model",
         xlab = "Residuals", col = "skyblue",
         breaks = seq(-400, 300, 50))
```




```r
# Plot a histogram of the residuals for the nonlinear model
hist(m2$residuals, main = "Residuals of the Nonlinear Model",
         xlab = "Residual", col = "skyblue",
         breaks = seq(-500, 300, 50))
```



In both models, the residuals range from approximately -400 to 300. However, in the nonlinear model, except for 3 observations, all of the residuals are between -200 and 200. In the linear model, though, there are around 11 observations that appear to go above 200 or below -200. This tells us that the residuals are slightly more clustered towards the center in the nonlinear model, and the distribution looks approximately normal. This suggests that the quadratic + periodic model may be a better fit of the data.

(g)

```
par(mfrow = c(2, 2))
summary(m1)
```

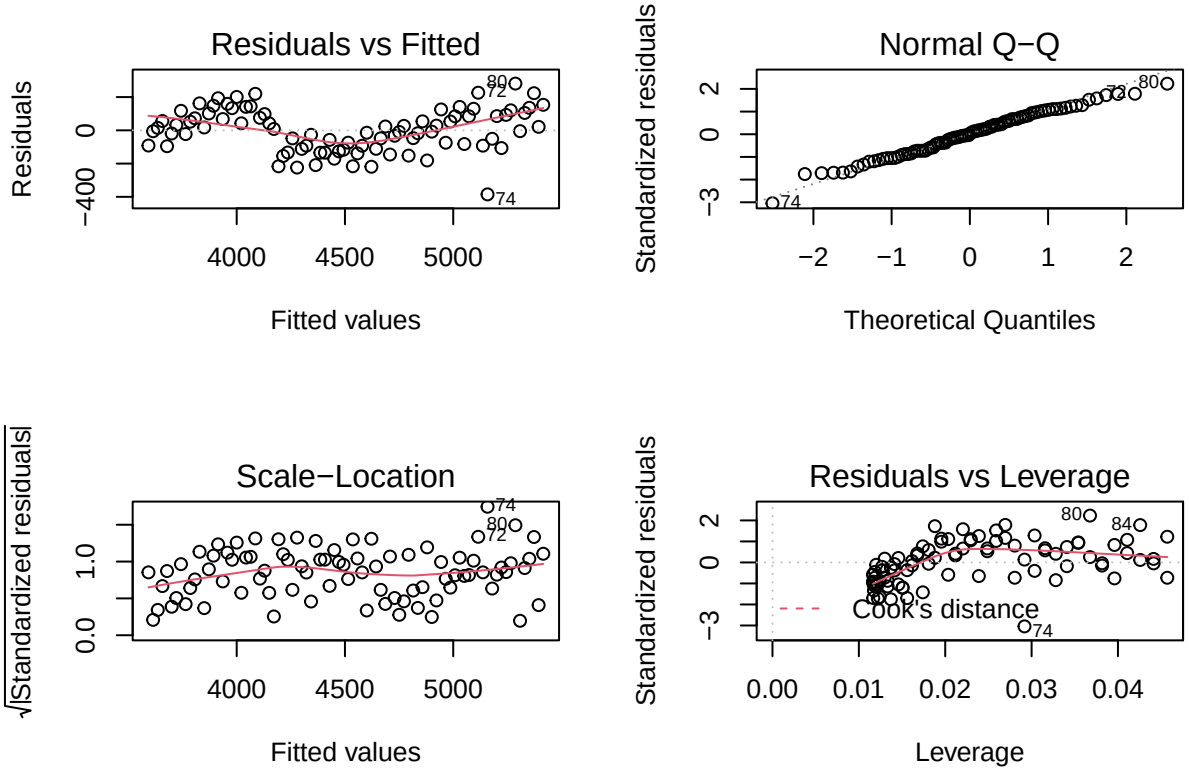
```
##
## Call:
## lm(formula = gdp_ts ~ t)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -385.17  -92.73    1.71   96.54  281.18
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.662e+05  4.441e+03  -37.42   <2e-16 ***
## t            8.481e+01  2.207e+00   38.43   <2e-16 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 128.5 on 84 degrees of freedom
## Multiple R-squared:  0.9462, Adjusted R-squared:  0.9456
## F-statistic:  1477 on 1 and 84 DF,  p-value: < 2.2e-16
```

```
anova(m1)
```

```
## Analysis of Variance Table
##
## Response: gdp_ts
##           Df   Sum Sq  Mean Sq F value    Pr(>F)
## t          1 24387077 24387077  1477.1 < 2.2e-16 ***
## Residuals 84  1386880    16510
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
plot(m1)
```




```
par(mfrow = c(2, 2))
summary(m2)
```

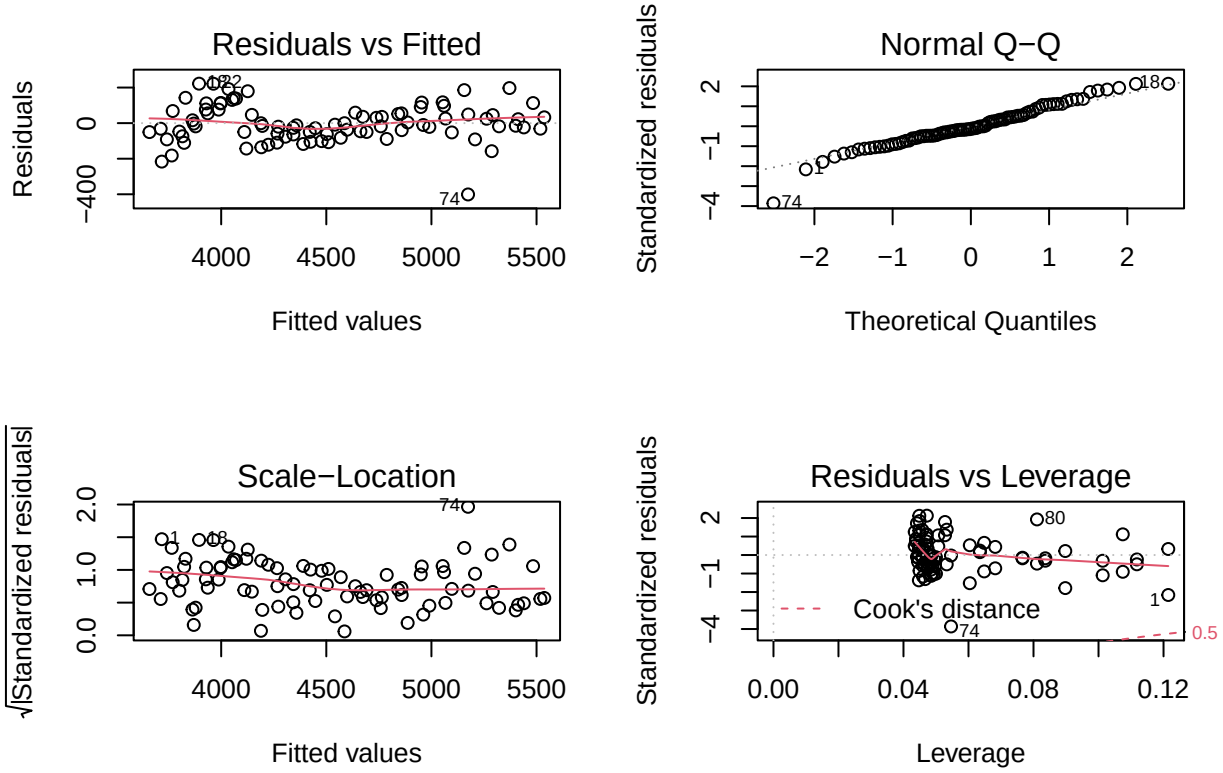
```
##
```

```
## Call:
## lm(formula = gdp_ts ~ t + t2 + cost + sint)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -400.98  -57.25  -13.61   57.86  221.83
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.333e+06  1.324e+06   4.783 7.64e-06 ***
## t           -6.374e+03  1.316e+03  -4.844 6.02e-06 ***
## t2           1.604e+00  3.269e-01   4.908 4.68e-06 ***
## cost         2.080e+00  1.620e+01   0.128    0.898
## sint        -6.929e+01  1.640e+01  -4.226 6.20e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 106.8 on 81 degrees of freedom
## Multiple R-squared:  0.9641, Adjusted R-squared:  0.9624
## F-statistic: 544.5 on 4 and 81 DF,  p-value: < 2.2e-16
```

```
anova(m2)
```

```
## Analysis of Variance Table
##
## Response: gdp_ts
##           Df   Sum Sq  Mean Sq   F value    Pr(>F)
## t          1 24387077 24387077 2137.5554 < 2.2e-16 ***
## t2         1   258841   258841   22.6877 8.241e-06 ***
## cost       1      188      188    0.0165    0.8982
## sint       1   203733   203733   17.8574 6.202e-05 ***
## Residuals 81   924118    11409
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
plot(m2)
```



The adjusted R^2 for the linear and nonlinear models are 0.9456 and 0.9624 respectively, and both have extremely low p-values and high F-statistics. This suggests that both models fit the data well, but the nonlinear model is marginally better. The one aspect of the nonlinear model that has a high p-value is the coefficient for $cos(2 * pi * t)$, indicating that simply having $sin(2 * pi * t)$ may be adequate. Additionally, if we look at the diagnostic plots, we can see that the quadratic model has more assumptions met than the linear model.

(h)

```
# Calculate the AIC and BIC values of each model
AIC(m1, m2)
```

```
##    df      AIC
## m1  3 1083.244
## m2  6 1054.331
```

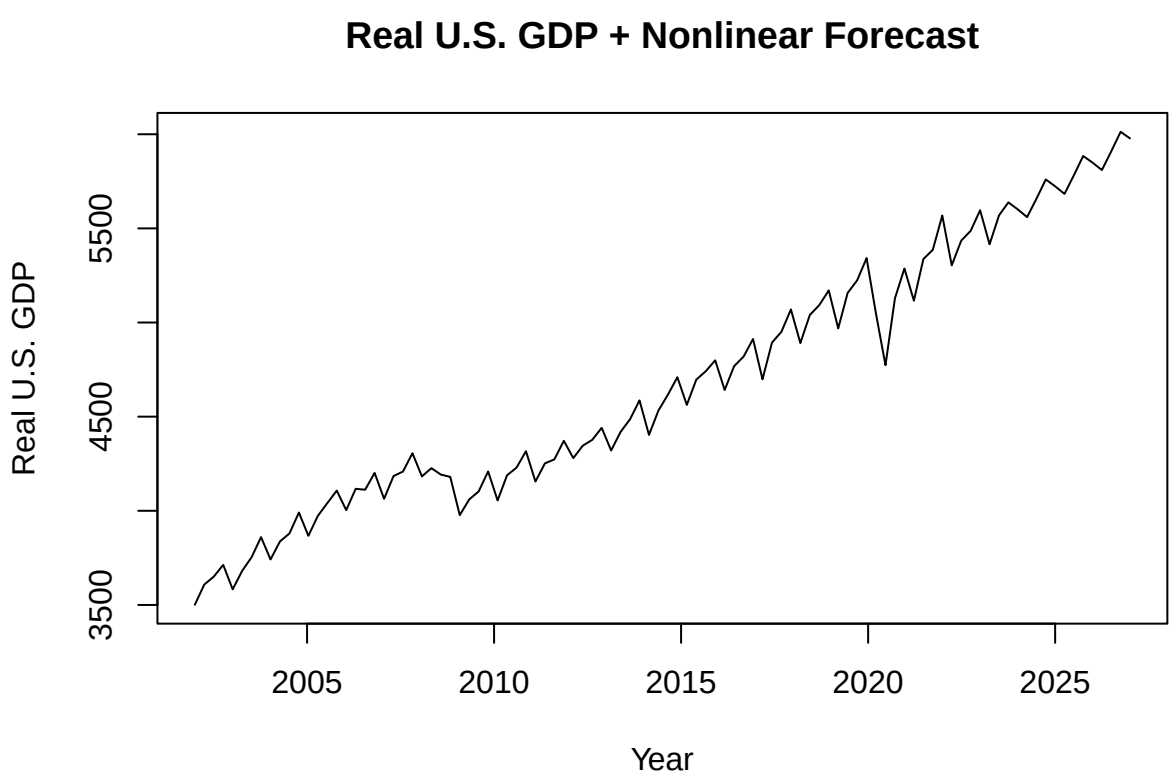
```
BIC(m1, m2)
```

```
##    df      BIC
## m1  3 1090.607
## m2  6 1069.057
```

Assuming that both models have positive AIC and BIC values, we aim to choose the model with the lower AIC or BIC. Consistent with our previous conclusions, we see that the nonlinear model (m2) has lower AIC and BIC values, suggesting that it is a better fit for the data. This tells us that the quadratic + polynomial model is both consistent and asymptotically efficient, compared to the purely linear model.

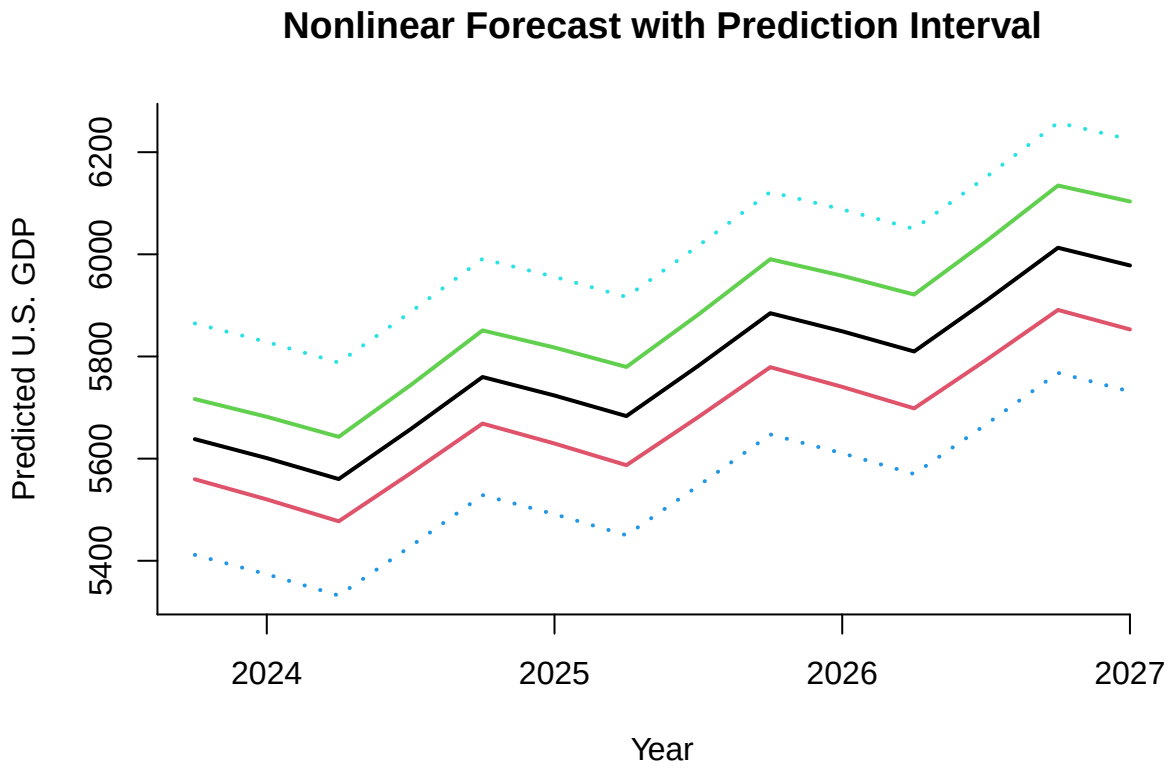
(i)

```
# Forecast 14 steps ahead using the nonlinear (quadratic + periodic) model
tn <- seq(2023.75, 2027.00, length = 14)
tn2 <- data.frame(
  t = tn,
  t2 = tn^2,
  cost <- cos(2*pi*tn),
  sint <- sin(2*pi*tn)
)
pred2 = predict(m2, tn2, se.fit = TRUE)
plot(c(t, tn2$t), c(gdp_ts, pred2$fit), type = 'l', main = "Real U.S. GDP + Nonlinear Forecast", xlab =
```




```
# Calculate the prediction and confidence intervals
pred.plim2 = predict(m2,tn2, level = 0.95, interval = "prediction")
pred.clim2 = predict(m2, tn2,level = 0.95, interval = "confidence")

# Plot the nonlinear forecast + prediction and confidence intervals
```

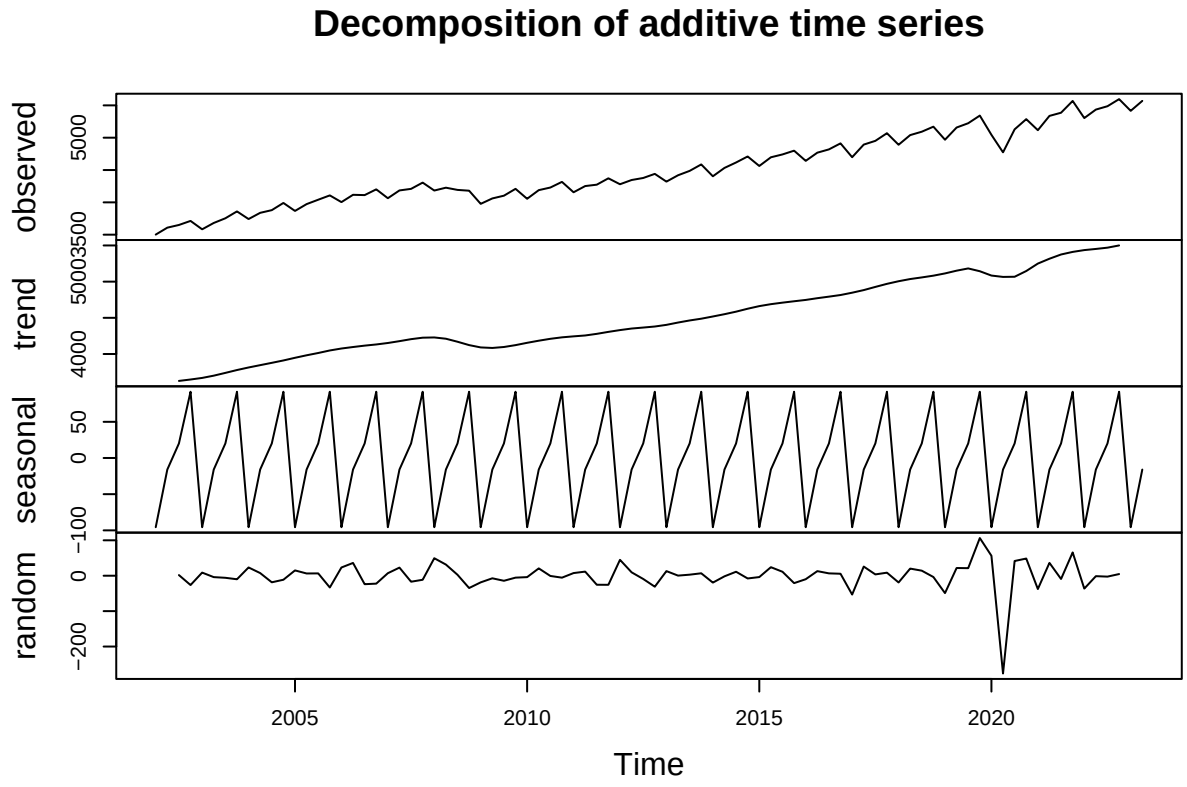
```r
matplot(tn2$t, cbind(pred.clim2, pred.plim2[,-1]),
        lty = c(1,1,1,3,3), type = "l", lwd = 2,
        main = "Nonlinear Forecast with Prediction Interval", ylab = "Predicted U.S. GDP", xlab = "Year")
axis(1, at = c(2023, 2024, 2025, 2026, 2027))
axis(2, at = seq(5200, 6400, length = 7))
```



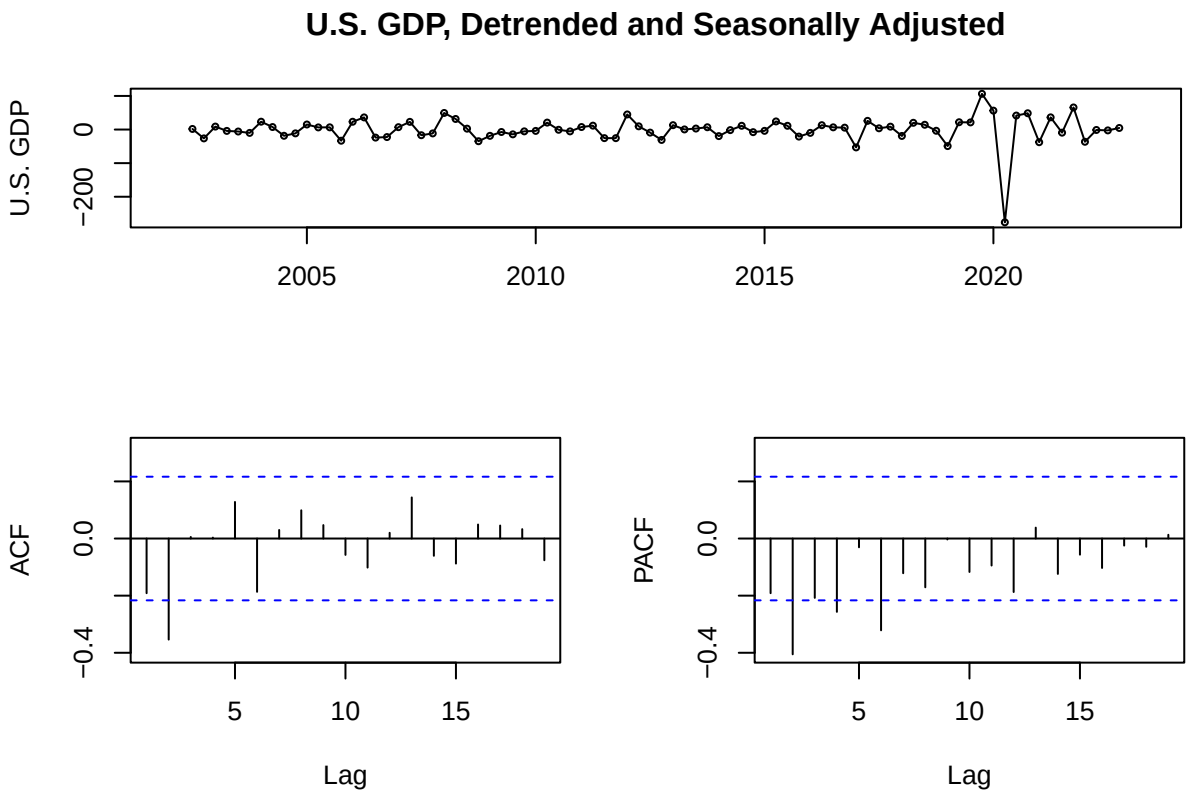
# # # # # # # # # # #

(a)

```r
# Perform and plot an additive decomposition
gdp_decomp_a <- decompose(gdp_ts, type = "additive")
plot(gdp_decomp_a)
```



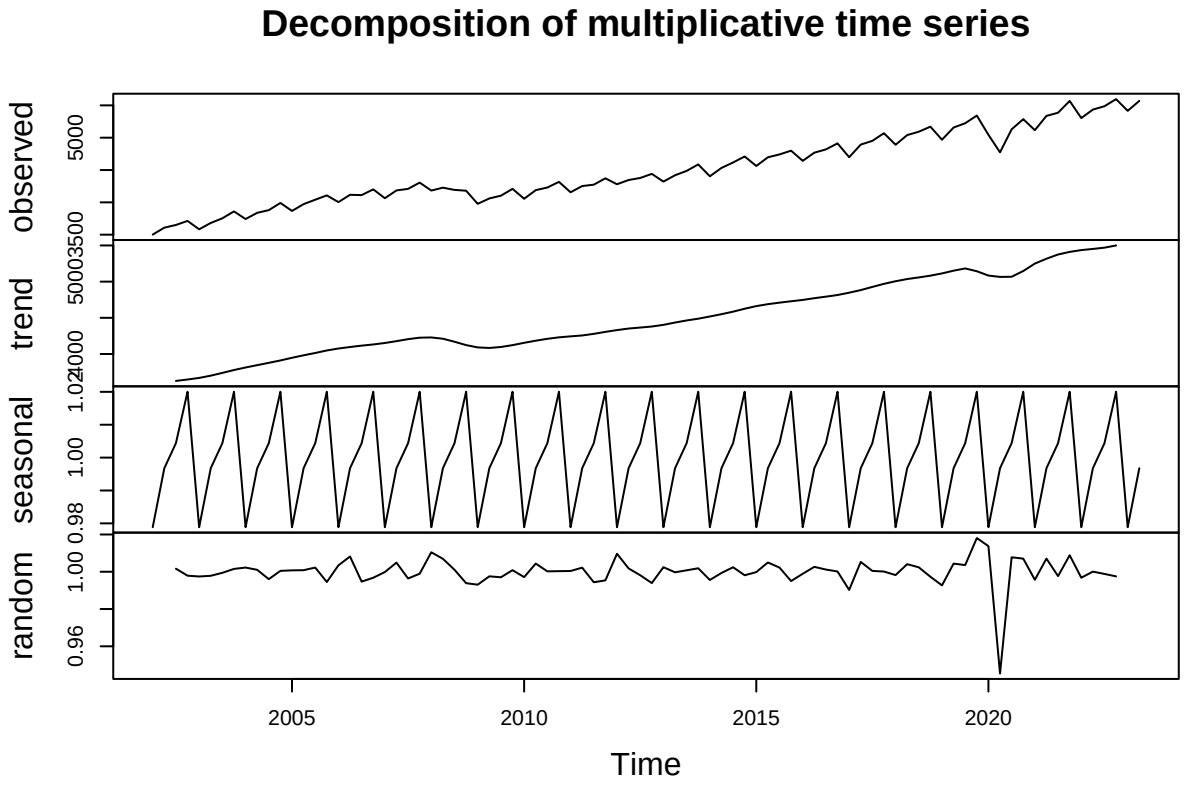

```
# Remove trend and seasonality, and plot the resulting series, along with its ACF and PACF
gdp_random_a <- gdp_decomp_a$random
tsdisplay(gdp_random_a, main = "U.S. GDP, Detrended and Seasonally Adjusted", ylab = "U.S. GDP")
```



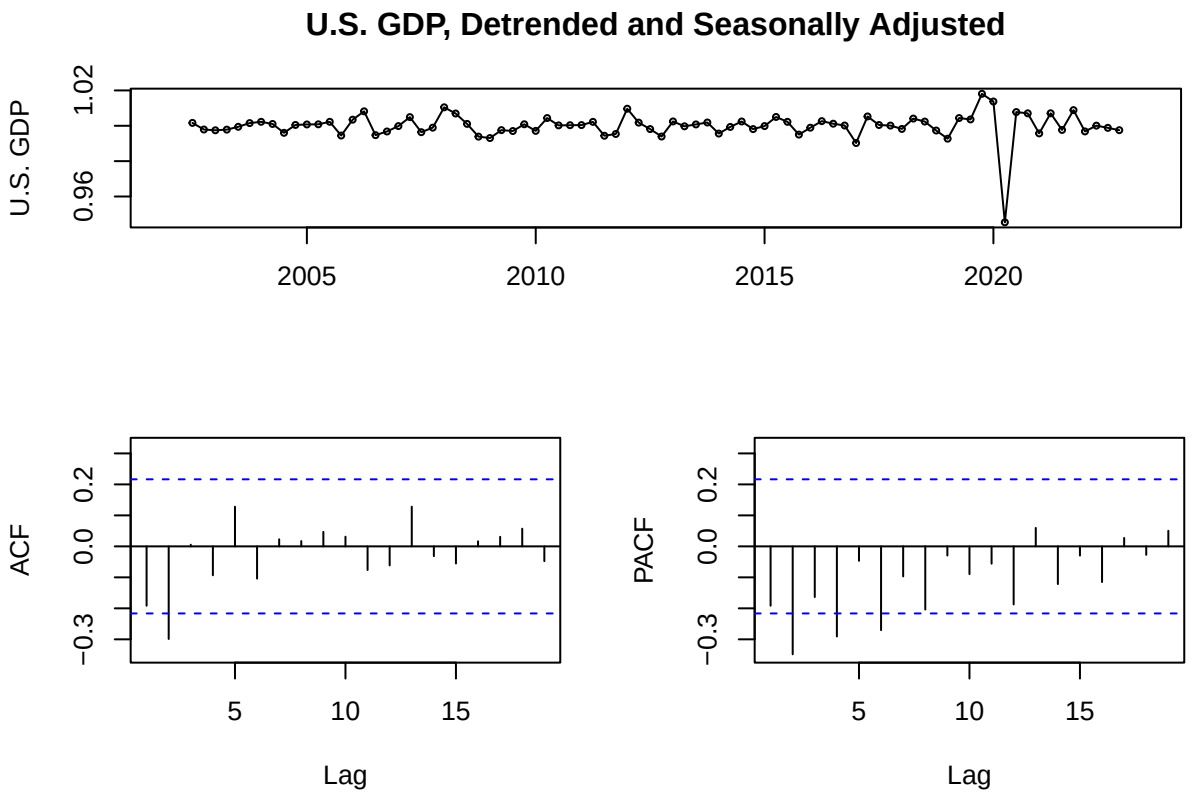
The additive decomposition shows us that the trend of real U.S. GDP is steadily increasing; there are two small dips that correspond to recessions (2008 financial crisis and 2020 COVID pandemic). Seasonality is also present, with GDP being lower in the first quarter of each year and reaching a peak by the fourth quarter. Upon examining the detrended and seasonally adjusted data (i.e, the "random" component of the time series), we see that the series indeed looks random, centered around 0. There is a sharp downward spike in 2020, which again corresponds to the COVID pandemic and resulting recession. The ACF suggests little autocorrelation in the series, except for lag 2. Therefore, there are some remaining autocorrelation after removing seasonality and trend. The PACF is significant for lags 2, 4, and 6, which suggests there might be some seasonality not captured during the decomposition.

(b)

```
# Perform and plot a multiplicative decomposition
gdp_decomp_m <- decompose(gdp_ts, type = "multiplicative")
plot(gdp_decomp_m)
```




```
# Remove trend and seasonality, and plot the resulting series, along with its ACF and PACF
gdp_decomp_m <- gdp_decomp_m$random
tsdisplay(gdp_decomp_m, main = "U.S. GDP, Detrended and Seasonally Adjusted", ylab = "U.S. GDP")
```



The multiplicative decomposition produces results very similar to those found in the additive decomposition. We see that the U.S. GDP trend is steadily increasing and very seasonal by the quarter. The detrended, seasonally adjusted component of the series looks randomly distributed, centered around 0 (with one dip in 2020, likely due to the COVID pandemic). The ACF suggests little autocorrelation in the series, except for lag 2. Therefore, there are some remaining autocorrelation after removing seasonality and trend. The PACF is significant for lag 2, 4, and 6, which suggests there might be some seasonality not captured during the decomposition.

(c)

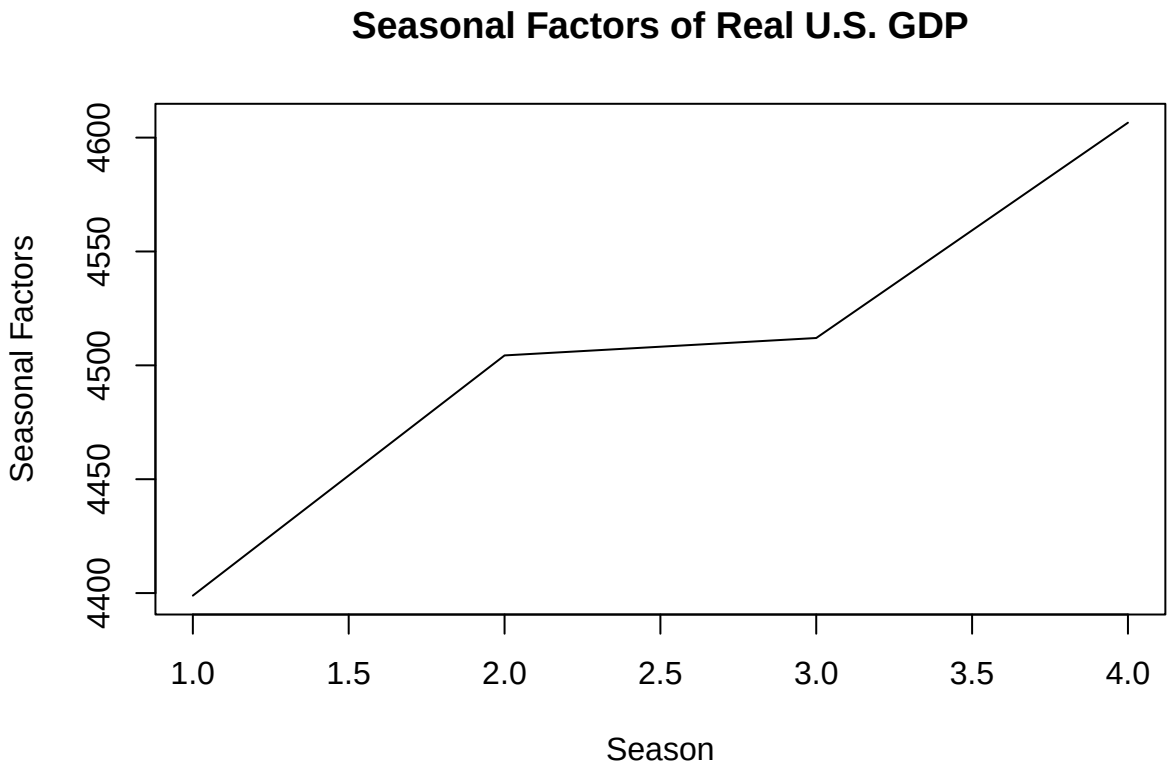
According to the decomposition plots, both produce the exact same result, with the same patterns for trend, seasonality, and randomness. Both of them shows an upward trend, a quarterly seasonality with unchanged amplitude, and a residual with a drop around 2020. However, though both decompositions produce the same results, we would prefer additive over multiplicative, as it is more efficient, computationally inexpensive, and easier to interpret in an economic context.

(d)

Based on the two decompositions, our models for the cycle would be similar, because additive and multiplicative decomposition produce the same patterns in trend, seasonality, and randomness.

(e)

```
# Fit a model with only the presence of seasonality and remove y-intercept
season_lm <- tslm(gdp_ts ~ season + 0)

# Take the seasonal dummy and plot the seasonal factors
plot(season_lm$coef, type = 'l', main = "Seasonal Factors of Real U.S. GDP", xlab = "Season", ylab = "S
```

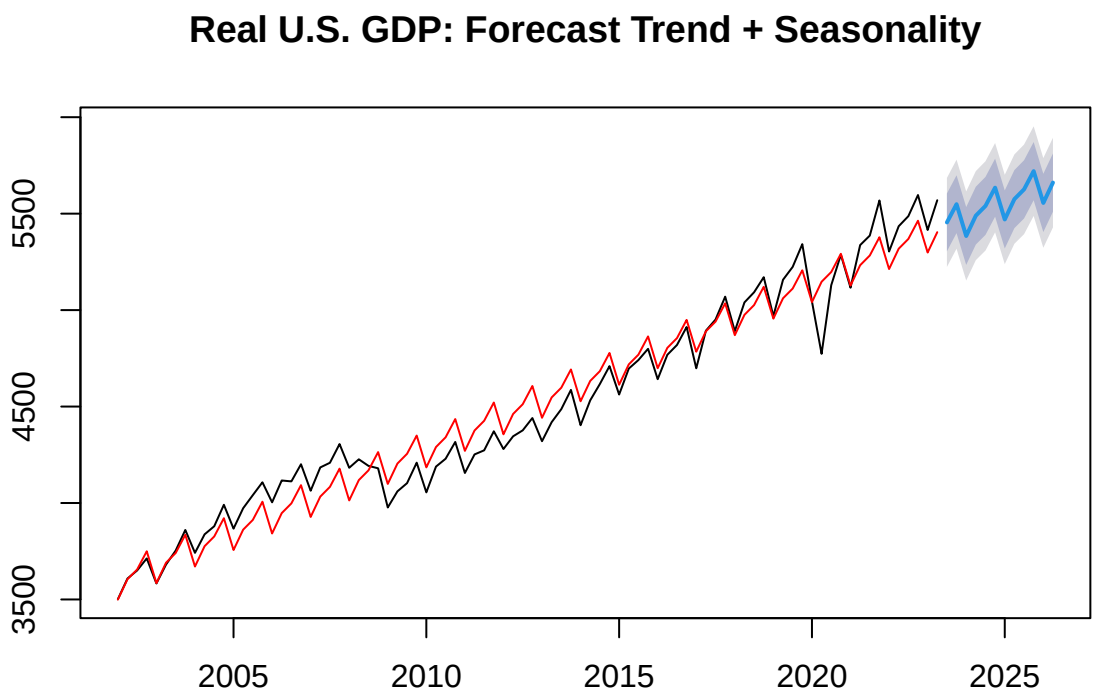


The seasonal factors show that U.S. GDP is generally increasing throughout the year. The first quarter is the lowest, consistent with the idea of "residual seasonality" as discussed in Section I. Then, the second quarter's US GDP increases sharply. From the second quarter to the third quarter, GDP stays stable with only a little increase. Moving forward to the fourth quarter, GDP experiences another sharp increase.

(f)

Based on our previous analysis, we think that a model with both trend and seasonality would be the most suitable to our data.

```
# Fit a model that contains both trend and seasonality
trend_season_lm = tslm(gdp_ts ~ trend + season)

# Plot the data, forecast 12-steps ahead, and respective fit
plot(forecast(trend_season_lm, h = 12), main = "Real U.S. GDP: Forecast Trend + Seasonality")
lines(trend_season_lm$fitted.values,col="red")
```



III. CONCLUSIONS & FUTURE WORK

Overall, our final model (quadratic + periodic) was an accurate representation of real U.S. GDP from 2002-2023. At the same time, it is relatively easy to interpret: in the long run, U.S. GDP is increasing at a quadratic rate, but within each year, it experiences clear seasonal fluctuations by the quarter.

Our series decomposition delved more into these trend and seasonality components, with both additive and multiplicative models representing the series well. The decomposition revealed (1) a general upward trend and (2) seasonality in which GDP is abnormally low during the first quarter and gradually increases over the next three quarters, with a particularly sharp increase from the third to fourth quarters. This is consistent with the concept of "residual seasonality", which has reaching implications for policymakers and economists.

As for our forecast, we can't yet pinpoint its accuracy until we know the future values, but given that our nonlinear model was a close fit of the data, we hope that our forecast will maintain that accuracy.

In terms of areas for improvement, we could use a longer time period and examine pre-2002 data to see if the trend and seasonality are as accurate as we suspect. By expanding the time period, we could also gain insight into recessions and their effects on GDP. We saw that the 2008 financial crisis and 2020 COVID pandemic resulted in sharp drops in GDP, so it could be beneficial to closely examine recessions in the last several decades.

IV. REFERENCES

Our data is sourced from the St. Louis Federal Reserve's database (FRED): https://fred.stlouisfed.org/series/ND000334Q

We also referenced an article from the St. Louis Fed about residual seasonality in Section I: https://www.stlouisfed.org/on-the-economy/2019/january/closer-look-residual-seasonality-gdp-growth