Econ 104 Project

Colin MacPherson, Mauricio Sammet, Leopold Arnaud

2025-02-27

Contents

Data Collection: All data is sourced from the Our World in Data — which pulls data from the UN, the World Bank, and more.

```
set.seed(123)

theme_set(theme_light())
my_df <- read_csv('climate_change_and_energy_usage.csv')
```

```
## Rows: 58 Columns: 6
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): Entity
## dbl (5): Year, carbon_growth, Annual change in primary energy consumption (%...
##
## 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.
```

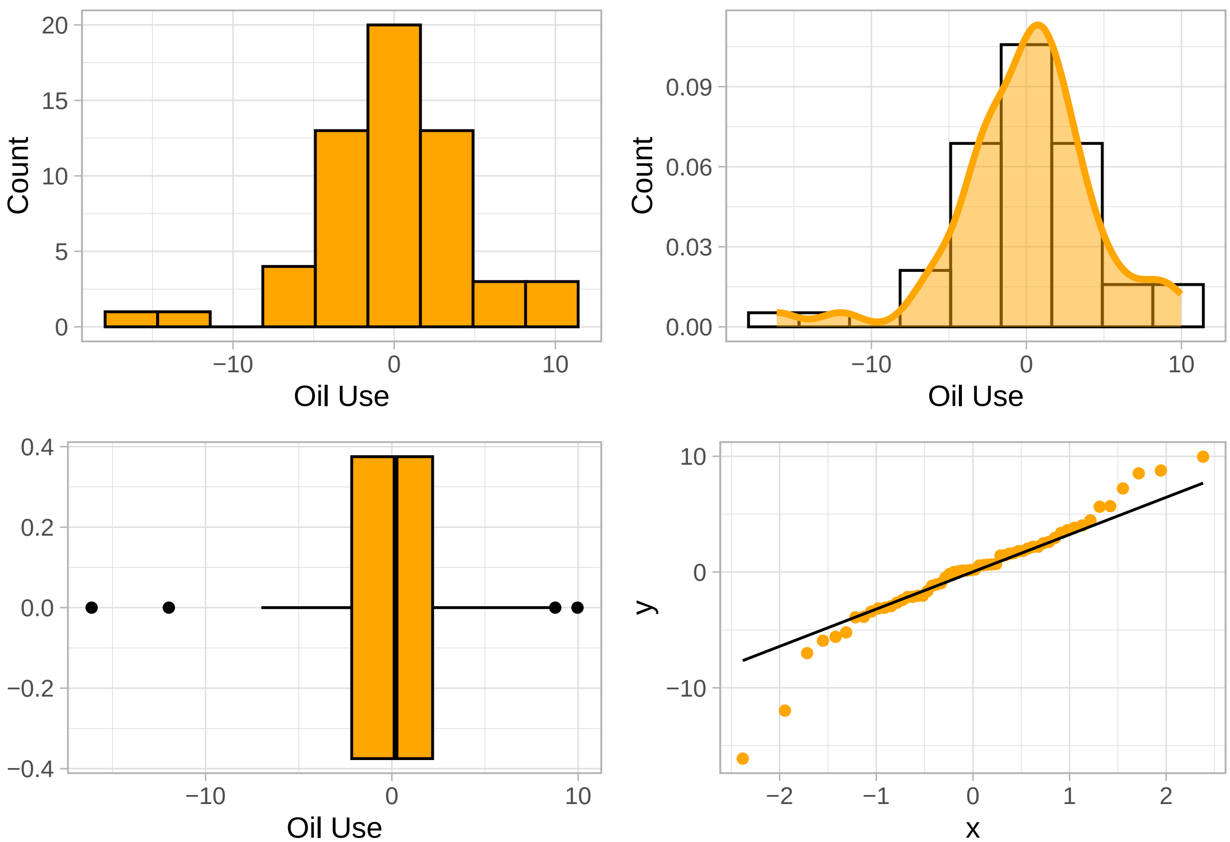
```
oil_prod_ts <- diff(ts(my_df[,6],start=1966,freq=1))
co2_ts <- ts(my_df[,3],start=1966,freq=1)
energy_ts <- ts(my_df[,4],start=1966,freq=1)
temp_ts <- ts(my_df[,5],start=1966,freq=1)

active_ts <- cbind(oil_prod_ts, co2_ts, energy_ts, temp_ts)
active_df <- data.frame(active_ts)
active_df[1,1] <- 0
active_ts[1,1] <- 0
```

Part 1: Time Series

(1) EDA

Change in Oil Production

```
oil_hist <- ggplot(active_df, aes(x = oil_prod_ts)) +
  geom_histogram(color = 'black', fill = '#ffa600', bins = round(1 + log(183, base = 2), 0)) +
  labs(x = "Oil Use", y = "Count")

oil_hist_fitted <- ggplot(active_df, aes(x = oil_prod_ts)) +
  geom_histogram(aes(y=..density..), color = 'black', fill = NA, bins = round(1 + log(183, base = 2), 0
  geom_density(lwd = 1.2, color = '#ffa600', fill = '#ffa600', alpha = 0.5) +
  labs(x = "Oil Use", y = "Count")

oil_box <- ggplot(active_df, aes(x = oil_prod_ts)) +
  geom_boxplot(color = 'black', fill = '#ffa600') +
  labs(x = "Oil Use")

oil_qq <- ggplot(active_df, aes(sample = oil_prod_ts)) +
  stat_qq(color = '#ffa600') +
  stat_qq_line()

ggarrange(oil_hist, oil_hist_fitted, oil_box, oil_qq)
```



```
summary(active_df$oil_prod_ts)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
```

```
## -16.11660  -2.15413   0.19714   0.03377   2.18856   9.96688
```

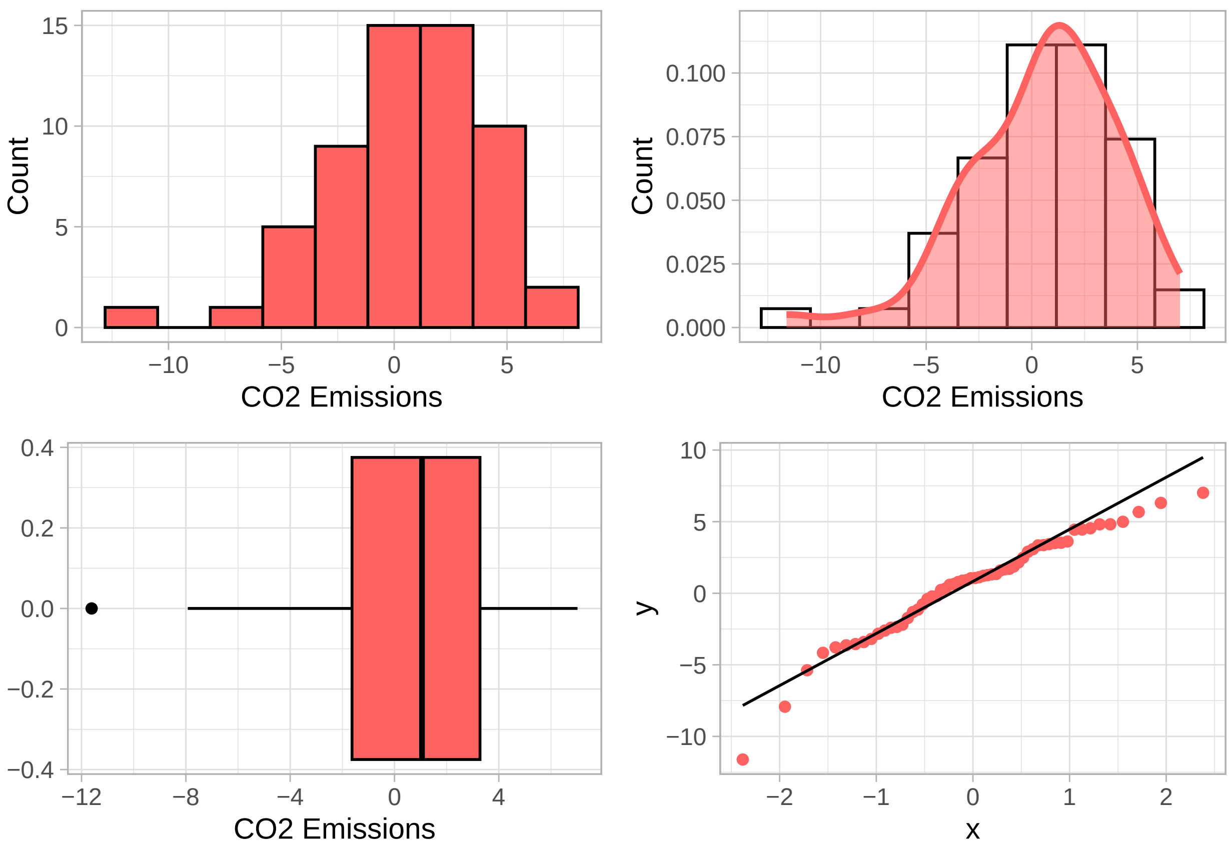
Histogram & Fitted Distribution: The graphs show a slightly left-skewed distribution. The median US oil consumption growth over the past 57 years is 0.19714%, suggesting a slight upward trend in oil production. However, there are also several years where oil use fluctuated significantly, both above and below the average growth of 0.03377%.

Boxplot: Again,he boxplot indicates a slight left skew in the data, with a few outliers beyond the median. One notable outlier is a sharp increase in oil use growth, reaching a maximum of 9.96688% in 2018. This significant rise is likely due to 2018 being the year with the highest energy consumption in U.S. history.

Q-Q Plot: The Q-Q plot shows that the data almost follows a normal distribution.

Change in Carbon Emissions

```r
co2_hist <- ggplot(active_df, aes(x = co2_ts)) +
  geom_histogram(color = 'black', fill = '#ff6361', bins = round(1 + log(183, base = 2), 0)) +
  labs(x = "CO2 Emissions", y = "Count")

co2_hist_fitted <- ggplot(active_df, aes(x = co2_ts)) +
  geom_histogram(aes(y=..density..), color = 'black', fill = NA, bins = round(1 + log(183, base = 2), 0)
  geom_density(lwd = 1.2, color = '#ff6361', fill = '#ff6361', alpha = 0.5) +
  labs(x = "CO2 Emissions", y = "Count")

co2_box <- ggplot(active_df, aes(x = co2_ts)) +
  geom_boxplot(color = 'black', fill = '#ff6361') +
  labs(x = "CO2 Emissions")

co2_qq <- ggplot(active_df, aes(sample = co2_ts)) +
  stat_qq(color = '#ff6361') +
  stat_qq_line()

ggarrange(co2_hist, co2_hist_fitted, co2_box, co2_qq)
```




```r
summary(active_df$co2_ts)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -11.6132  -1.6283   1.0547   0.5724   3.2783   7.0176
```

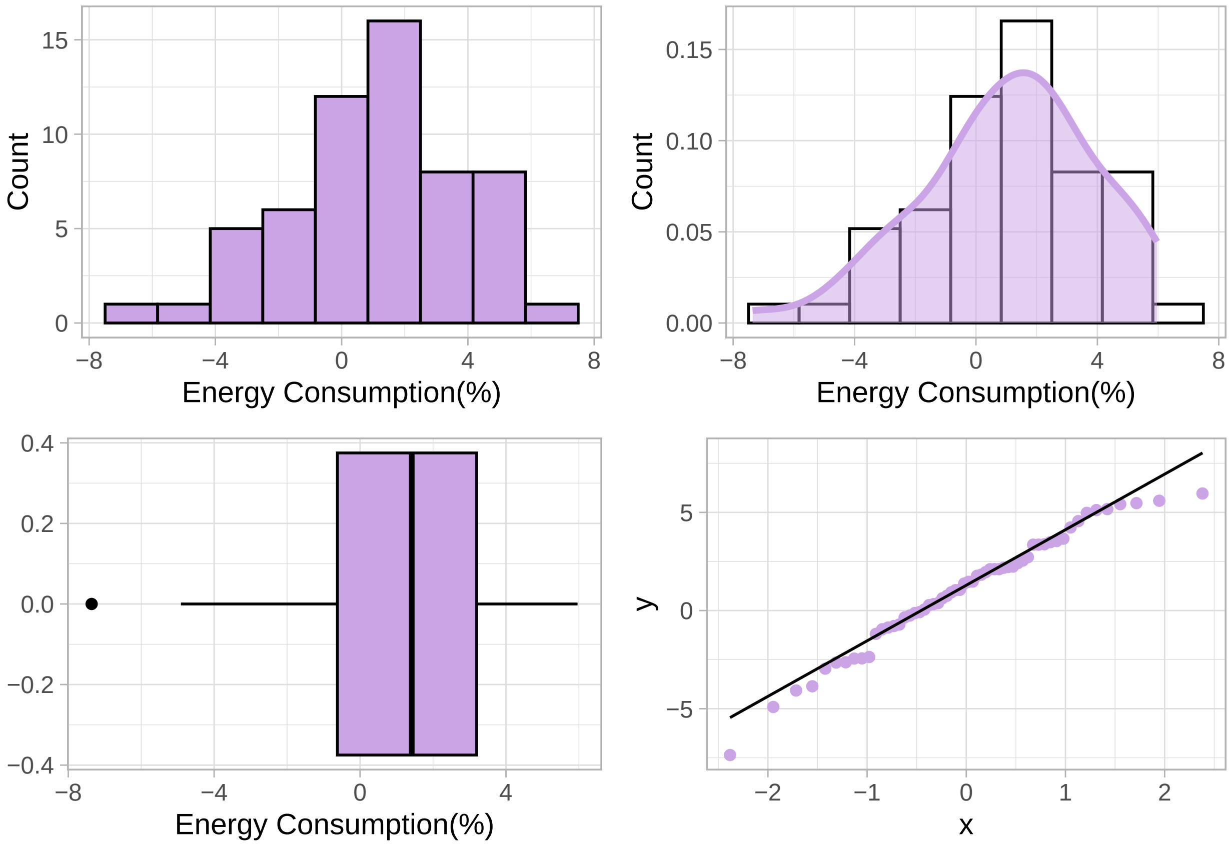
Histogram & Boxplot: From the histogram and boxplot graphs we see a distribution which is slightly left-skewed, indicating that over the years the US has had a semi-consistent growth pattern in CO2 emmissons.

Looking at the boxplot we can see over the last 57 years the US had averagely a growth of 0.5724% in CO_2 emmisions per year, and one significant outlier at -11.6132% in 2020. That outlier likely being a result of the 2020 Pandemic causing the CO_2 emmision growth to decrease.

Q-Q plot: The Q-Q plot reveals some deviation from normality, but the skewness isn't as extreme and is close enough.

Change in Primary Energy Consumption (%)

```
energy_hist <- ggplot(active_df, aes(x = energy_ts)) +
  geom_histogram(color = 'black', fill = '#cba4e6', bins = round(1 + log(183, base = 2), 0)) +
  labs(x = "Energy Consumption(%)", y = "Count")

energy_hist_fitted <- ggplot(active_df, aes(x = energy_ts)) +
  geom_histogram(aes(y=..density..), color = 'black', fill = NA, bins = round(1 + log(183, base = 2), 0
  geom_density(lwd = 1.2, color = '#cba4e6', fill = '#cba4e6', alpha = 0.5) +
  labs(x = "Energy Consumption(%)", y = "Count")

energy_box <- ggplot(active_df, aes(x = energy_ts)) +
  geom_boxplot(color = 'black', fill = '#cba4e6') +
  labs(x = "Energy Consumption(%)")

energy_qq <- ggplot(active_df, aes(sample = energy_ts)) +
  stat_qq(color = '#cba4e6') +
  stat_qq_line()

ggarrange(energy_hist, energy_hist_fitted, energy_box, energy_qq)
```




```
summary(active_df$energy_ts)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -7.3588 -0.6212  1.4178  1.0723  3.1969  5.9628
```

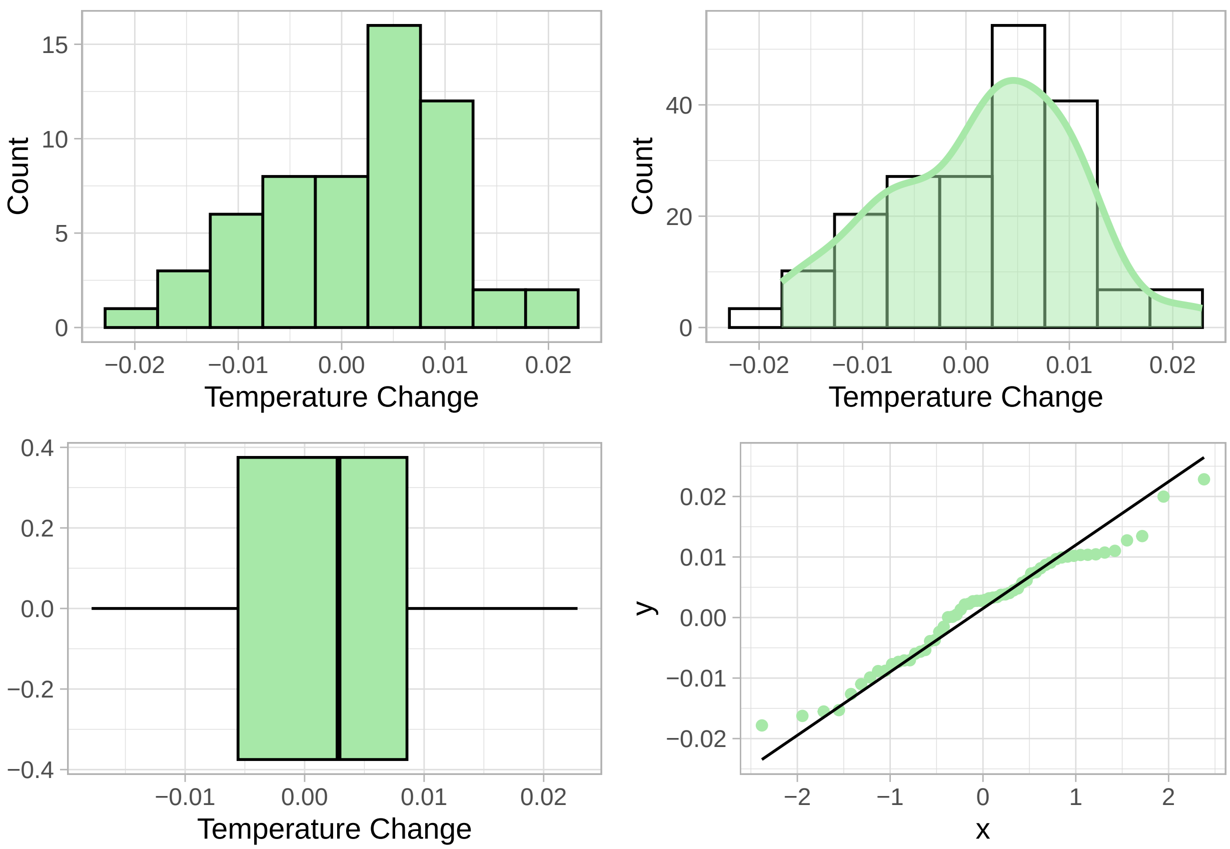
Histogram & Fitted Distribution: The graphs are slightly left-skew distributed. For most years in the U.S., the annual change in primary energy consumption typically ranged between +1% and +1.5%. But there were also a number of years that had a significant decrease in primary energy consumption.

Boxplot: The boxplot reveals that the data is slightly left-skewed. Additionally, there is one outlier in 2020, with a value of -7.3588, suggesting that this year experienced a significant decrease in primary energy consumption. This again is likely an impact of the COVID-19 pandemic and the associated global economic slowdown.

Q-Q Plot: The Q-Q plot indicates that the data nearly follows a normal distribution, with slight deviations from the plotted line at both the beginning and end of it.

Global average temperature anomaly relative to 1861-1890

```r
temp_hist <- ggplot(active_df, aes(x = temp_ts)) +
  geom_histogram(color = 'black', fill = '#a7e8a8', bins = round(1 + log(183, base = 2), 0)) +
  labs(x = "Temperature Change", y = "Count")

temp_hist_fitted <- ggplot(active_df, aes(x = temp_ts)) +
  geom_histogram(aes(y=..density..), color = 'black', fill = NA, bins = round(1 + log(183, base = 2), 0)
  geom_density(lwd = 1.2, color = '#a7e8a8', fill = '#a7e8a8', alpha = 0.5) +
  labs(x = "Temperature Change", y = "Count")

temp_box <- ggplot(active_df, aes(x = temp_ts)) +
  geom_boxplot(color = 'black', fill = '#a7e8a8') +
  labs(x = "Temperature Change")

temp_qq <- ggplot(active_df, aes(sample = temp_ts)) +
  stat_qq(color = '#a7e8a8') +
  stat_qq_line()

ggarrange(temp_hist, temp_hist_fitted, temp_box, temp_qq)
```




```r
summary(active_df$temp_ts)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -0.017823 -0.005572  0.002838  0.001532  0.008564  0.022837
```

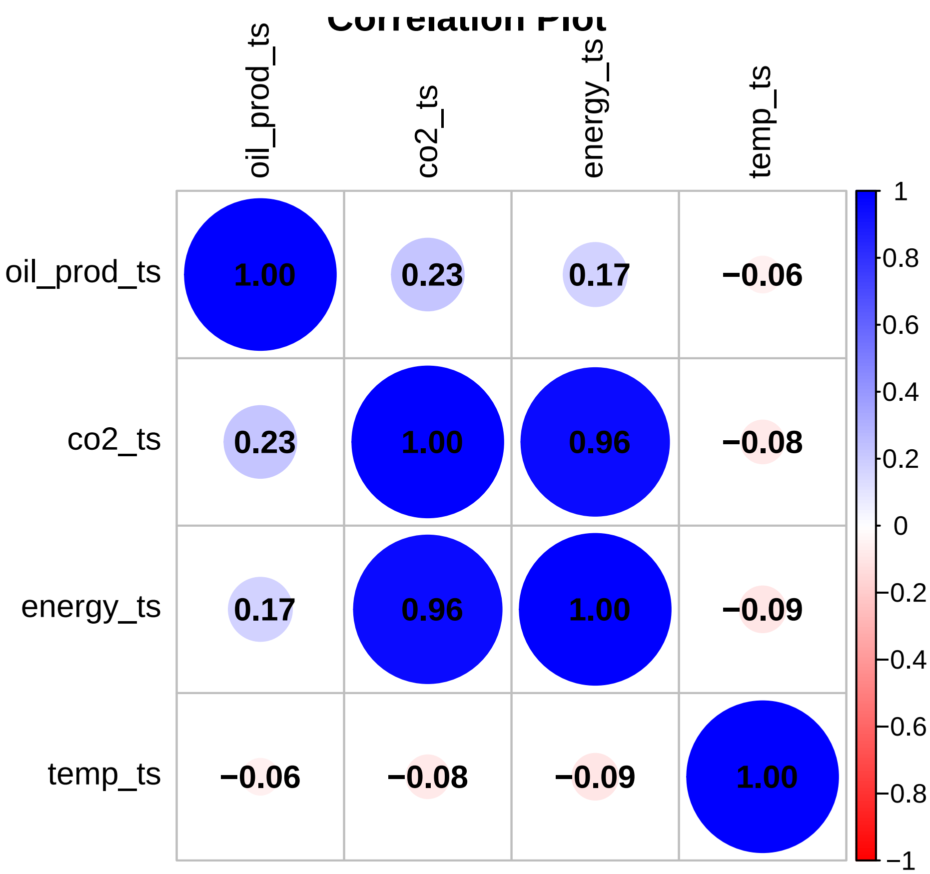
Histogram & Fitted Distribution: The graphs have a left-skewed distribution. When looking at the Global average temperature anomaly relative to 1861-1890, we see that as energy use increases, typically Global

Average Temperature follows along. The US temperature on average increased by 0.001532% per year, with a median of 0.002838%

Boxplot: The boxplot shows that the data is slightly left-skewed, but has a few outliers. The temperature changes fluctuating between -1.172342% and 0.685292.

Q-Q Plot: The Q-Q plot shows that the data almost follows a normal distribution, but seems to have some deviation in the line.

```
corr_matrix <- cor(active_df)
corrplot(corr_matrix, method = "circle", type = "full", col = colorRampPalette(c("red", "white", "blue"
         addCoef.col = "black", tl.col = "black", title = "Correlation Plot")
```



The Correlation plot shows that energy consumption is highly correlated with C02 emissions. However, the rest of the variables appear to have weak correlations with any of the other variables.

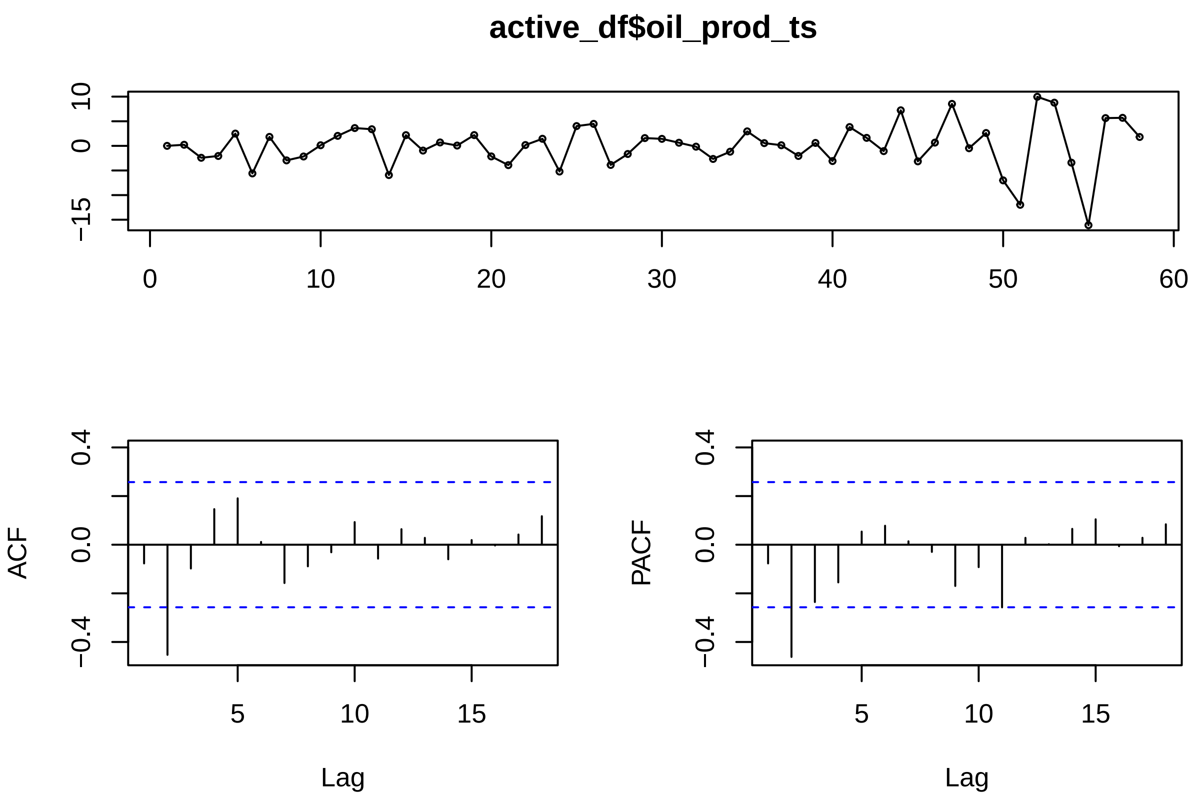
(2) tsdisplay

```r
adf.test(active_df$oil_prod_ts)
```

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

```
##
##  Augmented Dickey-Fuller Test
##
## data:  active_df$oil_prod_ts
## Dickey-Fuller = -5.7838, Lag order = 3, p-value = 0.01
## alternative hypothesis: stationary
```

```r
tsdisplay(active_df$oil_prod_ts)
```

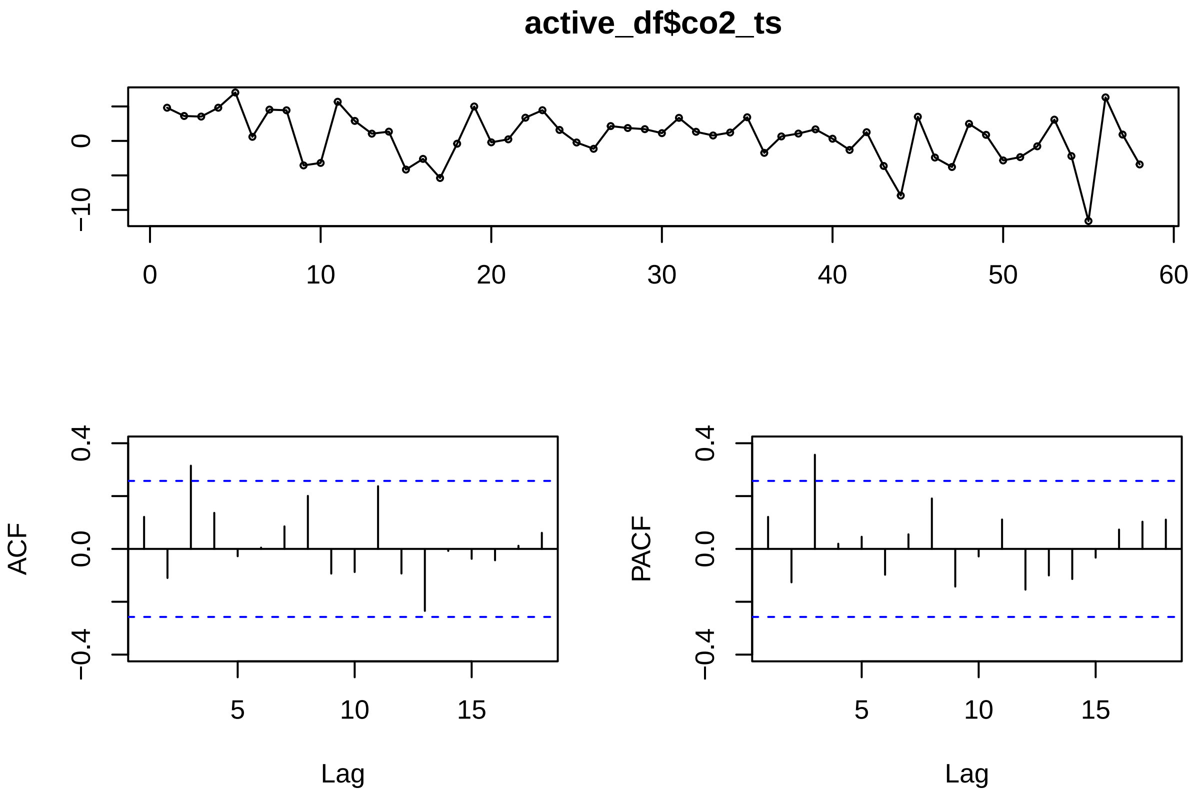


Both the ADF test and graph indicate the variable is stationary. The PACF of the oil production time series reveals some dynamics at lags 2, and 11, suggesting their significance and indicating that they should be included in our model. This implies that an Ar(1), AR(2) and potentially AR(11) model should be used for oil production.

```r
adf.test(active_df$co2_ts)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  active_df$co2_ts
## Dickey-Fuller = -3.3826, Lag order = 3, p-value = 0.0674
## alternative hypothesis: stationary
```

```r
tsdisplay(active_df$co2_ts)
```

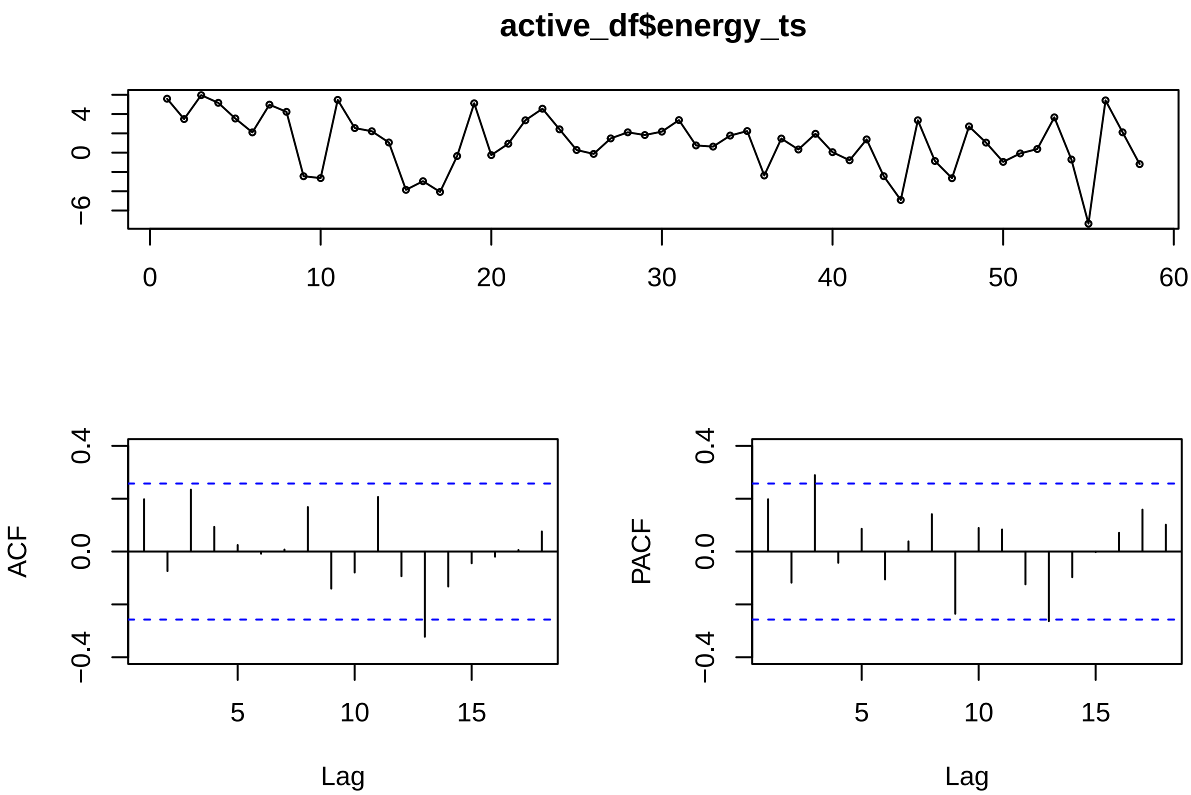


The graph appears to be stationary, however, the ADF test indicates it is just within the non-rejection zone. For this project we will assume it is stationary. When looking at the PACF of the CO2 emissions time series, lag 3 appears to be the only significant one, indicating it should be included in our model. We will run an AR(1) and AR(3) model for CO2 emissions.

```r
adf.test(active_df$energy_ts)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  active_df$energy_ts
## Dickey-Fuller = -3.6048, Lag order = 3, p-value = 0.04059
## alternative hypothesis: stationary
```

```r
tsdisplay(active_df$energy_ts)
```



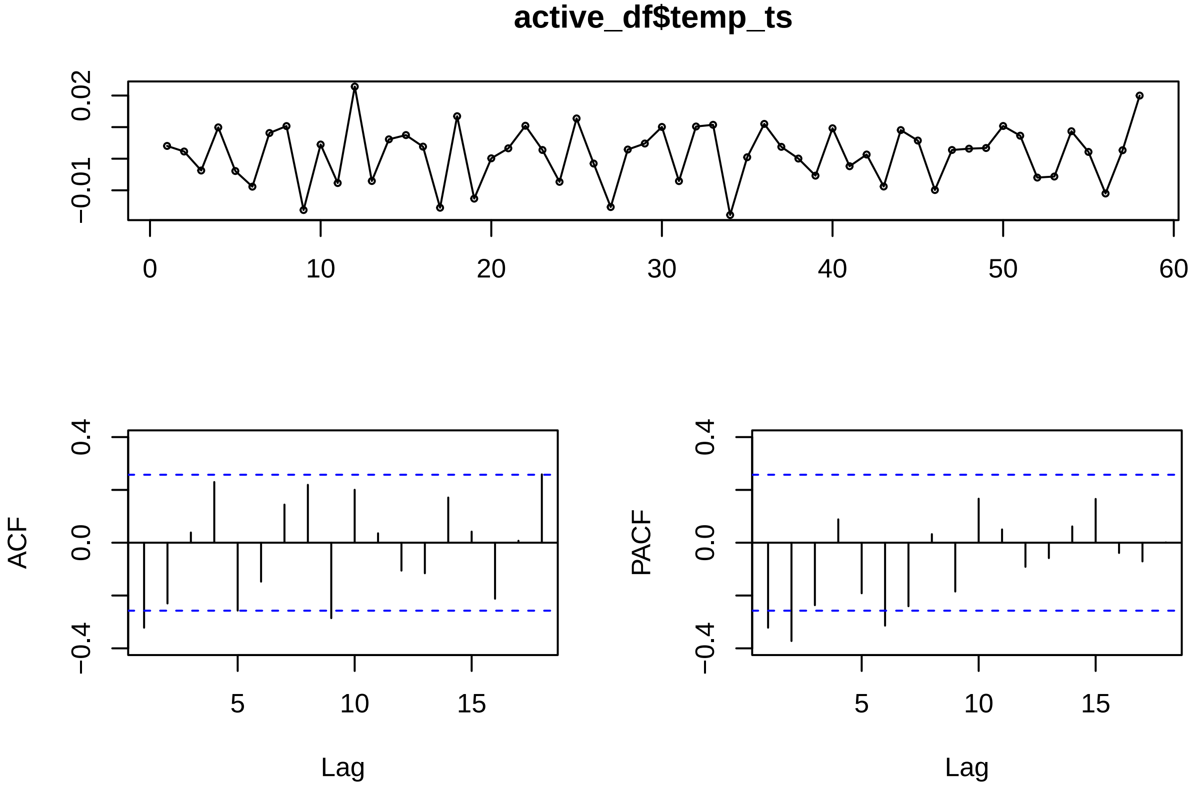
Both the ADF test and graph indicate the variable is stationary. The PACF of the Energy Consumption time series reveals some dynamics at lag 3, and (slightly) at lag 13 suggesting them to be significant and indicating they should be included in our model. This implies that an Ar(1), AR(3) and potentially AR(13) model should be used for Energy Consumption.

```
adf.test(active_df$temp_ts)
```

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

```
##
##  Augmented Dickey-Fuller Test
##
## data:  active_df$temp_ts
## Dickey-Fuller = -4.5788, Lag order = 3, p-value = 0.01
## alternative hypothesis: stationary
```

```
tsdisplay(active_df$temp_ts)
```



Both the ADF test and graph indicate the variable is stationary. The PACF of the Global Average Temperature Changes time series highlights significant dynamics at lags 1, 2, and 6, pointing to their significance and suggesting they should be incorporated into our model. This indicates that an AR(1), AR(2), and AR(6) model should be applied to model Global Average Temperature Changes.

(3) Autoregressive Models

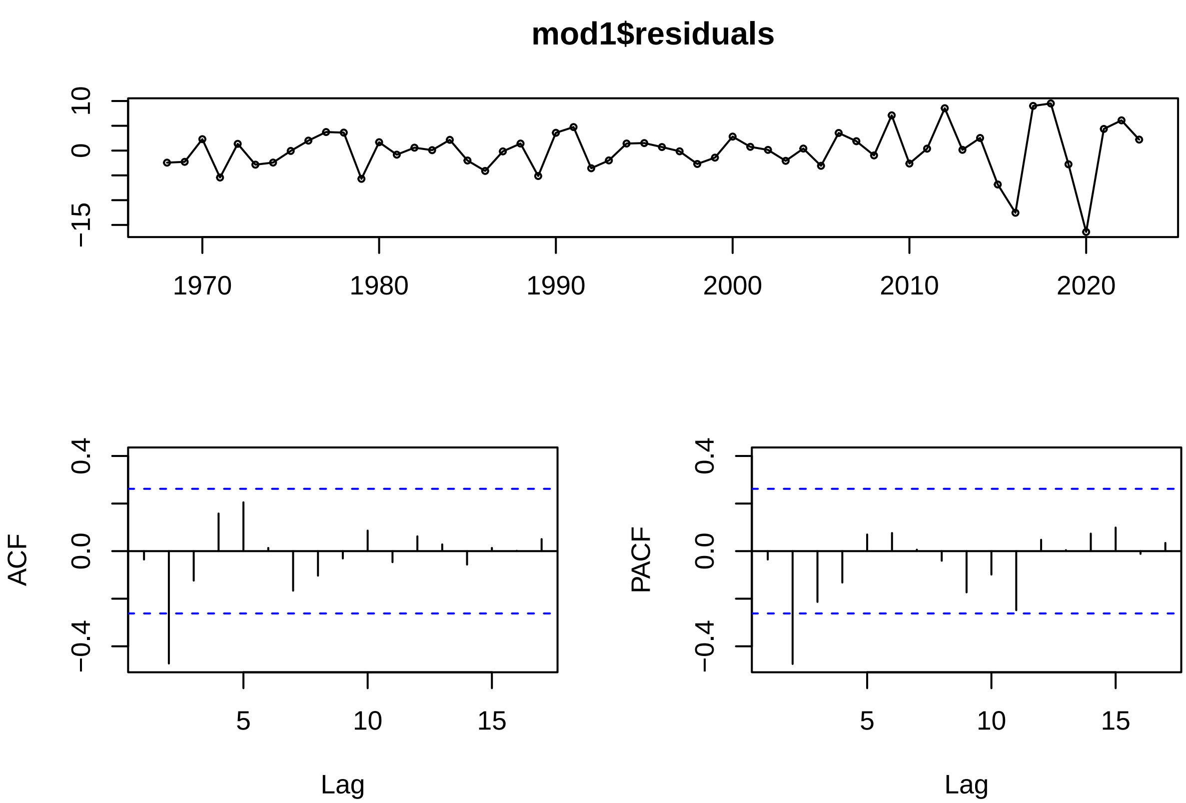
Change in Oil Production

AR(1)

```
y1 <- oil_prod_ts
mod1 <- dynlm(y1 ~ L(y1, 1))
summary(mod1)
```

```
##
## Time series regression with "ts" data:
## Start = 1968, End = 2023
##
## Call:
## dynlm(formula = y1 ~ L(y1, 1))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.4115  -2.4217   0.2692   2.2367   9.5095
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.03119    0.61398   0.051    0.960
## L(y1, 1)    -0.07723    0.13586  -0.568    0.572
##
## Residual standard error: 4.595 on 54 degrees of freedom
## Multiple R-squared:  0.005947,   Adjusted R-squared:  -0.01246
## F-statistic: 0.3231 on 1 and 54 DF,  p-value: 0.5721
```

```
tsdisplay(mod1$residuals)
```



Our

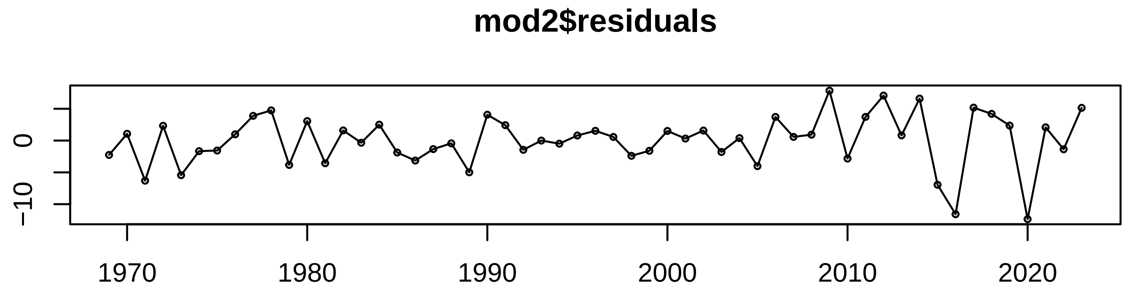
ACF and PACF residual plots for our AR(1) model demonstrate signs of serial correlation. This indicates our model has not captured all the dynamics of our oil production growth variable. Another model may be better or the use of serially correlated errors may be necessary.

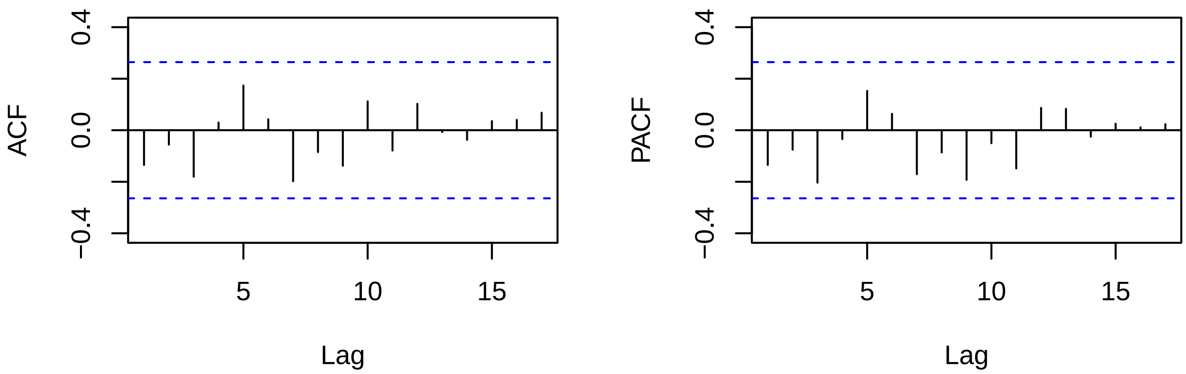
AR(2)

```
mod2 <- dynlm(y1 ~ L(y1, 1:2))
summary(mod2)
```

```
##
## Time series regression with "ts" data:
## Start = 1969, End = 2023
##
## Call:
## dynlm(formula = y1 ~ L(y1, 1:2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.3546  -1.8362   0.5656   2.3884   7.8492
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.02722    0.55553   0.049 0.961110
## L(y1, 1:2)1 -0.11796    0.12227  -0.965 0.339147
## L(y1, 1:2)2 -0.47793    0.12407  -3.852 0.000323 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.119 on 52 degrees of freedom
## Multiple R-squared:  0.2266, Adjusted R-squared:  0.1969
## F-statistic: 7.618 on 2 and 52 DF,  p-value: 0.001254
```

```
tsdisplay(mod2$residuals)
```





There are no signs of serial correlation in the errors of the AR(2) model. This may be the better suited model.

```
index <- floor(2/3 * nrow(active_df))
train_data <- active_ts[1:index, ]
test_data <- active_ts[(index+1):nrow(active_ts), ]

train_model_1 <- dynlm(oil_prod_ts~ L(oil_prod_ts, 1), data = train_data)
coef <- train_model_1$coefficients
oil_prod_ts_1 <- active_ts[index:(nrow(active_df)-1),"oil_prod_ts"]
oil_prod_ts_2 <- active_ts[(index - 1):(nrow(active_df)-2),"oil_prod_ts"]
forecast_oil <- coef[1] + coef[2]*oil_prod_ts_1
f_errors1 <- test_data[ ,1] - forecast_oil

train_model_2 <- dynlm(oil_prod_ts~ L(oil_prod_ts, 1:2), data = train_data)
coef <- train_model_2$coefficients
oil_prod_ts_2 <- active_ts[(index - 1):(nrow(active_df)-2),"oil_prod_ts"]
forecast_oil <- coef[1] + coef[2]*oil_prod_ts_1 + coef[3]*oil_prod_ts_2
f_errors2 <- test_data[, 1] - forecast_oil

rmse_1 <- sqrt(mean(f_errors1^2, na.rm = TRUE))
rmse_2 <- sqrt(mean(f_errors2^2, na.rm = TRUE))

print(paste("RMSE AR(1):", rmse_1))
```

```
## [1] "RMSE AR(1): 6.62472791436378"
```

```
print(paste("RMSE AR(2):", rmse_2))
```

```
## [1] "RMSE AR(2): 5.5983922662473"
```

The RMSE's for the model's re-affirm our suggestion that the AR(2) model is better for modeling oil production growth.

```r
BIC(train_model_1)
```

```
## [1] 339.7469
```

```r
BIC(train_model_2)
```

```
## [1] 324.7411
```

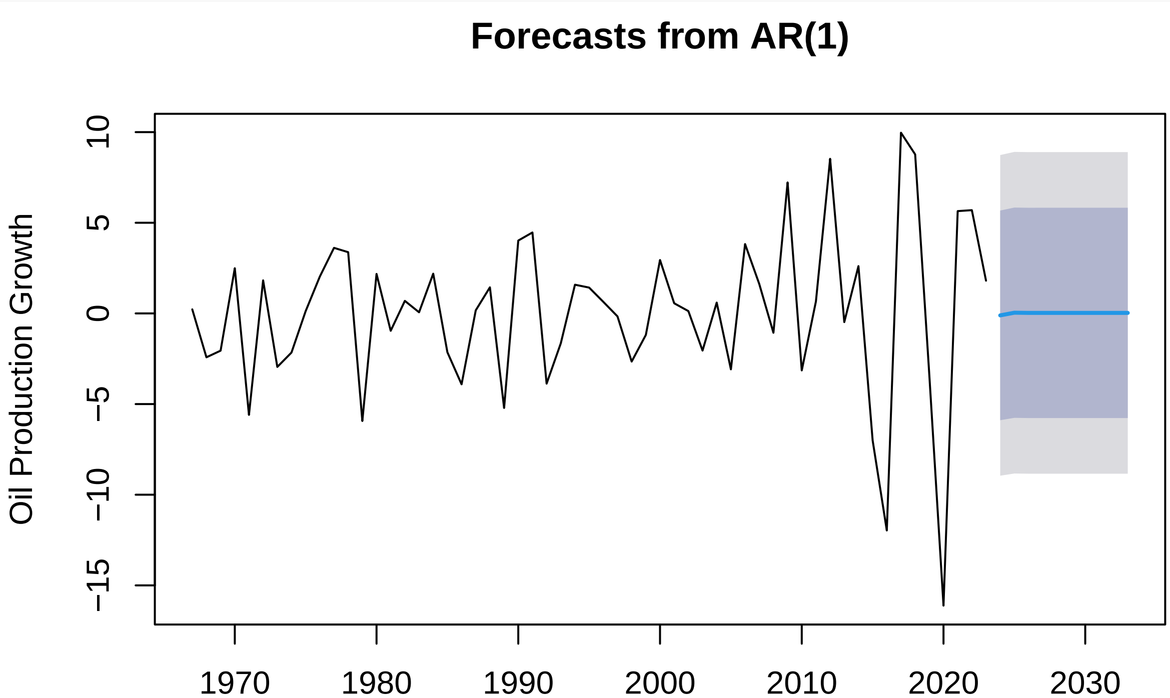
The BIC confirms our suggestion that the AR(2) model is better than the AR(1) model at capturing the dynamics for the changes in annual oil production.

Forecasting

```r
mod_1 <- ar(oil_prod_ts, aic=FALSE, order.max=1, method="ols")
forecast(mod_1, 10)
```

```
##      Point Forecast     Lo 80    Hi 80     Lo 95    Hi 95
## 2024    -0.10826707 -5.890360 5.673826 -8.951215 8.734681
## 2025     0.03955084 -5.759758 5.838860 -8.829727 8.908829
## 2026     0.02813548 -5.771276 5.827547 -8.841299 8.897570
## 2027     0.02901704 -5.770395 5.828429 -8.840418 8.898452
## 2028     0.02894897 -5.770463 5.828361 -8.840486 8.898384
## 2029     0.02895422 -5.770458 5.828366 -8.840481 8.898390
## 2030     0.02895382 -5.770458 5.828366 -8.840482 8.898389
## 2031     0.02895385 -5.770458 5.828366 -8.840482 8.898389
## 2032     0.02895385 -5.770458 5.828366 -8.840482 8.898389
## 2033     0.02895385 -5.770458 5.828366 -8.840482 8.898389
```

```r
plot(forecast(mod_1, 10),ylab = "Oil Production Growth")
```

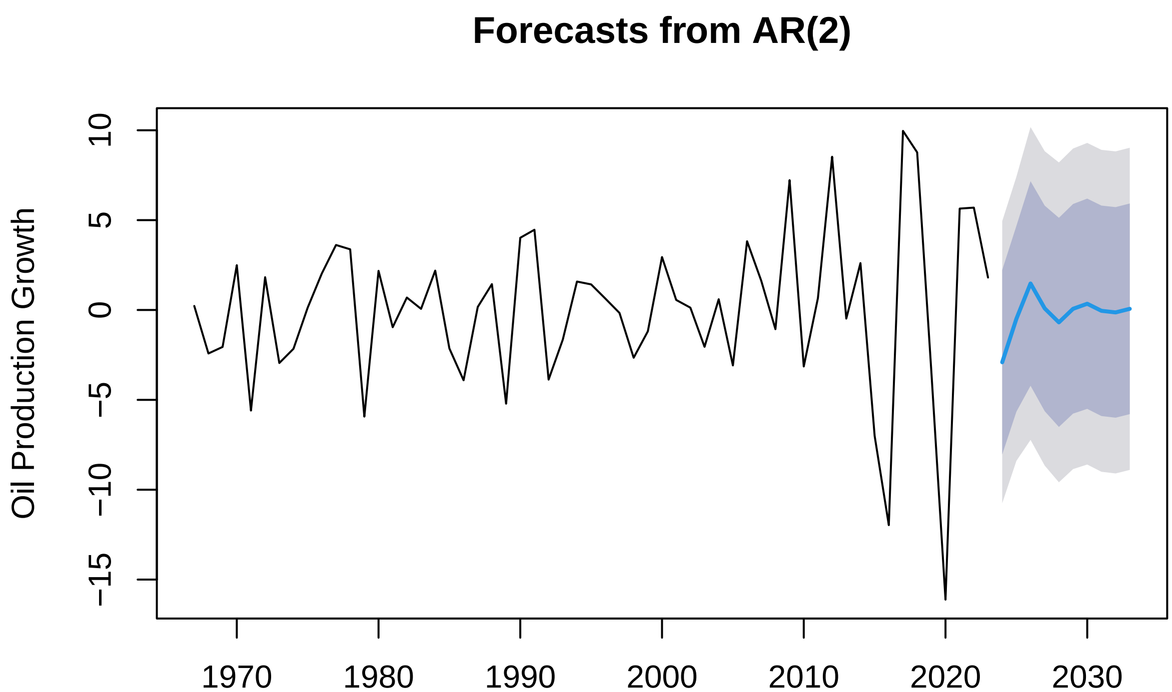



```r
mod_2 <- ar(oil_prod_ts, aic=FALSE, order.max=2, method="ols")
forecast(mod_2, 10)
```

```
##      Point Forecast     Lo 80    Hi 80      Lo 95     Hi 95
```

```
## 2024    -2.90799255 -8.040546 2.224560 -10.757555  4.941570
## 2025    -0.49282680 -5.660963 4.675309  -8.396808  7.411155
## 2026     1.47516565 -4.215316 7.165647  -7.227674 10.178006
## 2027     0.08875087 -5.630233 5.807735  -8.657681  8.835183
## 2028    -0.68827467 -6.506672 5.130122  -9.586745  8.210196
## 2029     0.06598832 -5.766065 5.898041  -8.853367  8.985344
## 2030     0.34838187 -5.502163 6.198927  -8.599255  9.296019
## 2031    -0.04541250 -5.901104 5.810279  -9.000921  8.910096
## 2032    -0.13392631 -5.992804 5.724952  -9.094307  8.826454
## 2033     0.06472029 -5.795832 5.925273  -8.898221  9.027662
```

```
plot(forecast(mod_2, 10),ylab = "Oil Production Growth")
```



Change in Carbon Emissions

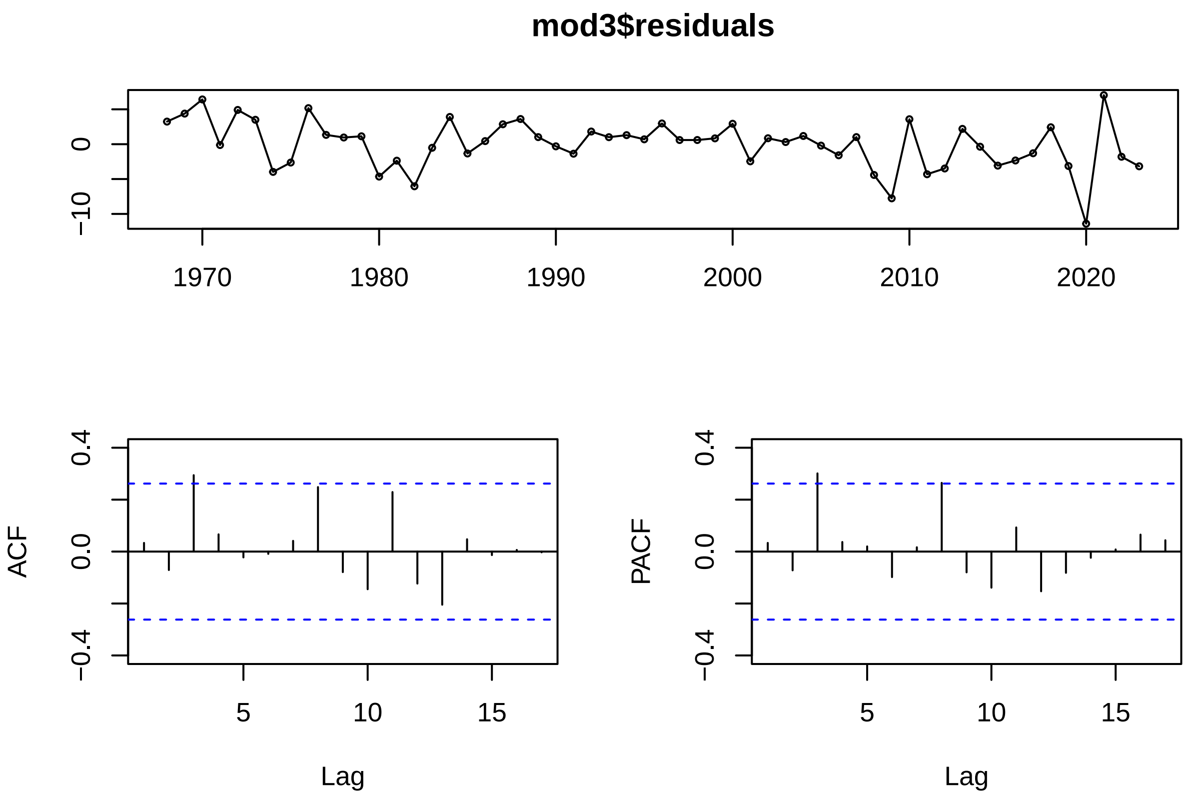
AR(2)

```
y2 <- co2_ts
mod3 <- dynlm(y2 ~ L(y2, 1:2))
summary(mod3)
```

```
##
## Time series regression with "ts" data:
## Start = 1968, End = 2023
##
## Call:
## dynlm(formula = y2 ~ L(y2, 1:2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.3990  -2.3470   0.5957   2.2517   7.0256
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)   0.4529     0.4825   0.939    0.352
## L(y2, 1:2)1   0.1247     0.1371   0.909    0.367
## L(y2, 1:2)2  -0.1276     0.1354  -0.943    0.350
##
## Residual standard error: 3.518 on 53 degrees of freedom
## Multiple R-squared:  0.0279, Adjusted R-squared:  -0.008786
## F-statistic: 0.7605 on 2 and 53 DF,  p-value: 0.4725
```

```r
tsdisplay(mod3$residuals)
```



There appears to be slight serial correlation of the errors for the AR(2) model as seen at the lags 3 and 8 in the PACF. This indicates the model may not have captured all the dynamics of the change in carbon emissions. Adding a feature or lag may improve the model.

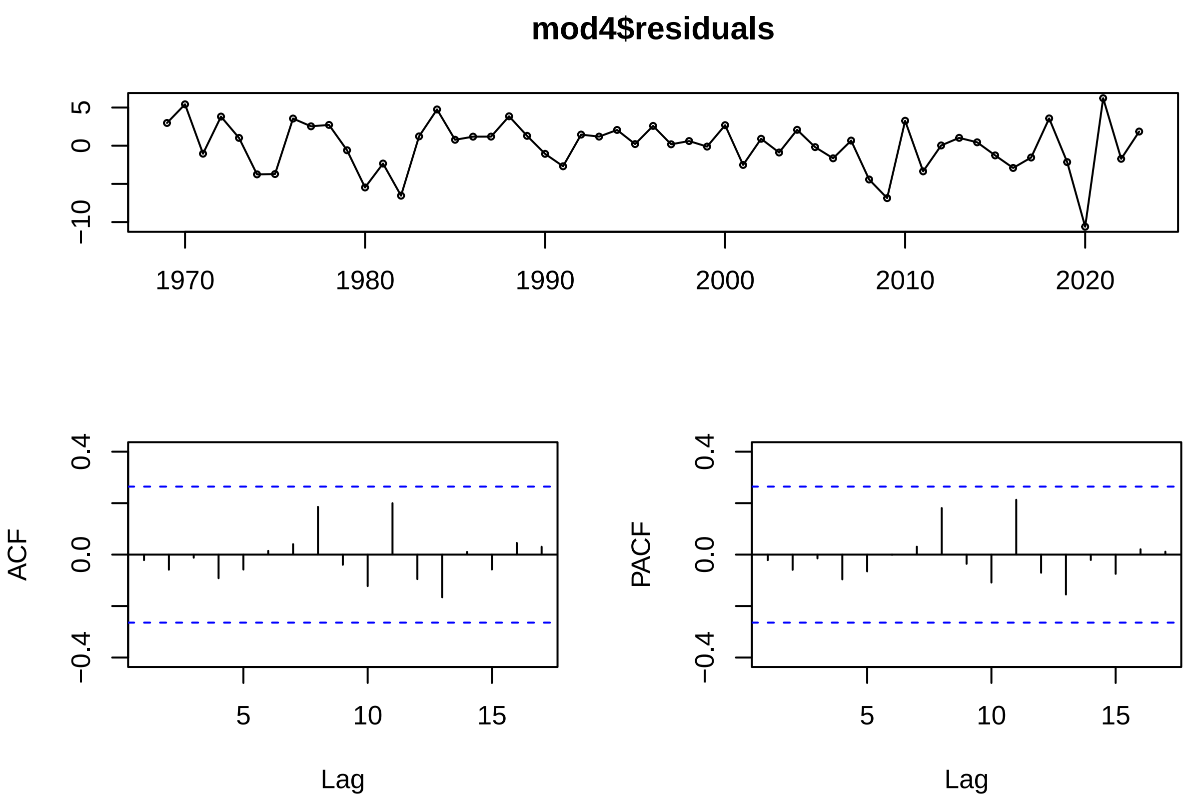
AR(3)

```r
mod4 <- dynlm(y2 ~ L(y2, 1:3))
summary(mod4)
```

```
##
## Time series regression with "ts" data:
## Start = 1969, End = 2023
##
## Call:
## dynlm(formula = y2 ~ L(y2, 1:3))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.6020  -1.6822   0.6112   2.0716   6.2200
##
## Coefficients:
```

```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.2206     0.4590   0.481  0.63290
## L(y2, 1:3)1   0.1485     0.1298   1.144  0.25814
## L(y2, 1:3)2  -0.1984     0.1296  -1.531  0.13183
## L(y2, 1:3)3   0.3762     0.1307   2.879  0.00581 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.298 on 51 degrees of freedom
## Multiple R-squared:  0.1659, Adjusted R-squared:  0.1169
## F-statistic: 3.382 on 3 and 51 DF,  p-value: 0.02506
```

```
tsdisplay(mod4$residuals)
```



There is no evidence of serial correlation indicating the model is well-specified and will produce reliable estimation results.

```
train_model_3 <- dynlm(co2_ts~ L(co2_ts, 1:2), data = train_data)
coef <- train_model_3$coefficients
co2_ts_1 <- active_ts[index:(nrow(active_df)-1),"co2_ts"]
co2_ts_2 <- active_ts[(index - 1):(nrow(active_df)-2),"co2_ts"]
co2_ts_3 <- active_ts[(index - 2):(nrow(active_df)-3),"co2_ts"]
forecast_co2 <- coef[1] + coef[2]*co2_ts_1 + coef[3]*co2_ts_2
f_errors3 <- test_data[,2] - forecast_co2

train_model_4 <- dynlm(co2_ts~ L(co2_ts, 1:3), data = train_data)
coef <- train_model_4$coefficients
forecast_co2 <- coef[1] + coef[2]*co2_ts_1 + coef[3]*co2_ts_2 + coef[4]*co2_ts_3
f_errors4 <- test_data[,2] - forecast_co2
```

```r
rmse_3 <- sqrt(mean(f_errors3^2, na.rm = TRUE))
rmse_4 <- sqrt(mean(f_errors4^2, na.rm = TRUE))

print(paste("RMSE AR(2):", rmse_3))
```

```
## [1] "RMSE AR(2): 4.23173074689738"
```

```r
print(paste("RMSE AR(3):", rmse_4))
```

```
## [1] "RMSE AR(3): 3.77268083898825"
```

The RMSE supports the suggestion that our AR(3) model is better than the AR(2) model. It's RMSE is lower which indicates that it has better prediction accuracy.

```r
BIC(train_model_3)
```

```
## [1] 312.8372
```

```r
BIC(train_model_4)
```

```
## [1] 303.2455
```

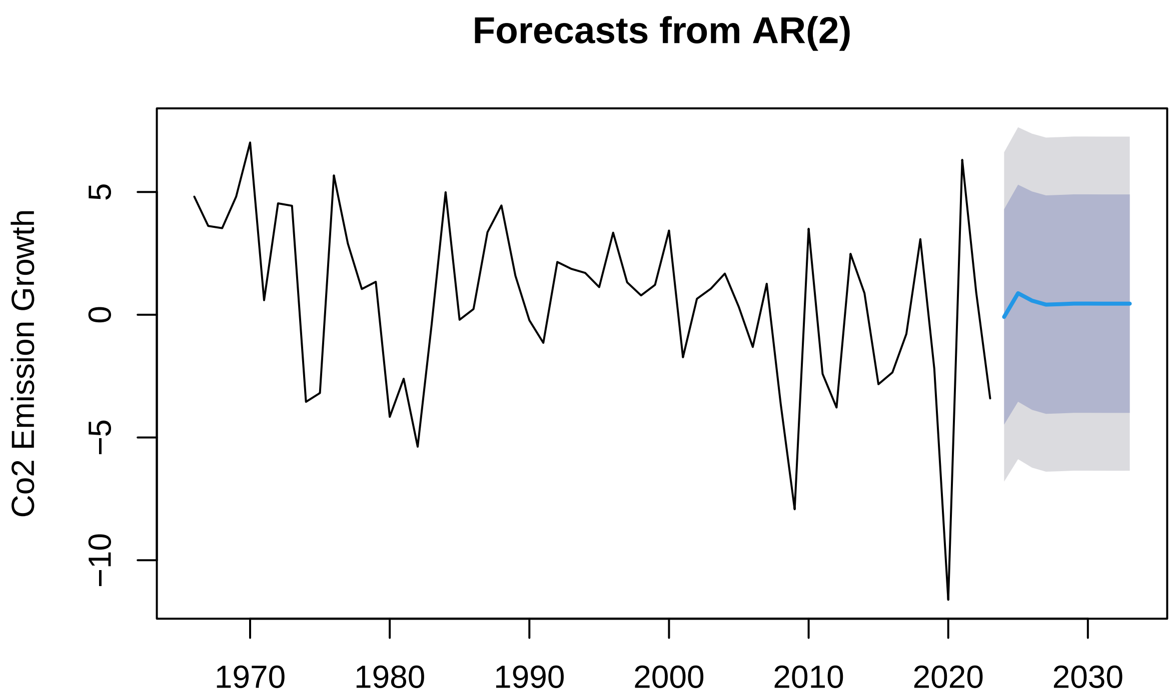
The AR(3) model is better model fit than the AR(2) model as indicated by the lower BIC.

Forecasting

```r
mod_3 <- ar(co2_ts, aic=FALSE, order.max=2, method="ols")
forecast(mod_3, 10)
```

```
##      Point Forecast     Lo 80    Hi 80     Lo 95    Hi 95
## 2024    -0.08950323 -4.476126 4.297119 -6.798264 6.619257
## 2025     0.87682499 -3.543760 5.297410 -5.883876 7.637526
## 2026     0.57360877 -3.874220 5.021437 -6.228758 7.375976
## 2027     0.41250270 -4.037257 4.862262 -6.392817 7.217822
## 2028     0.43110651 -4.018895 4.881108 -6.374583 7.236796
## 2029     0.45398287 -3.996075 4.904041 -6.351794 7.259759
## 2030     0.45446120 -3.995598 4.904520 -6.351317 7.260239
## 2031     0.45160184 -3.998458 4.901662 -6.354178 7.257382
## 2032     0.45118431 -3.998876 4.901245 -6.354596 7.256964
## 2033     0.45149711 -3.998563 4.901557 -6.354283 7.257277
```

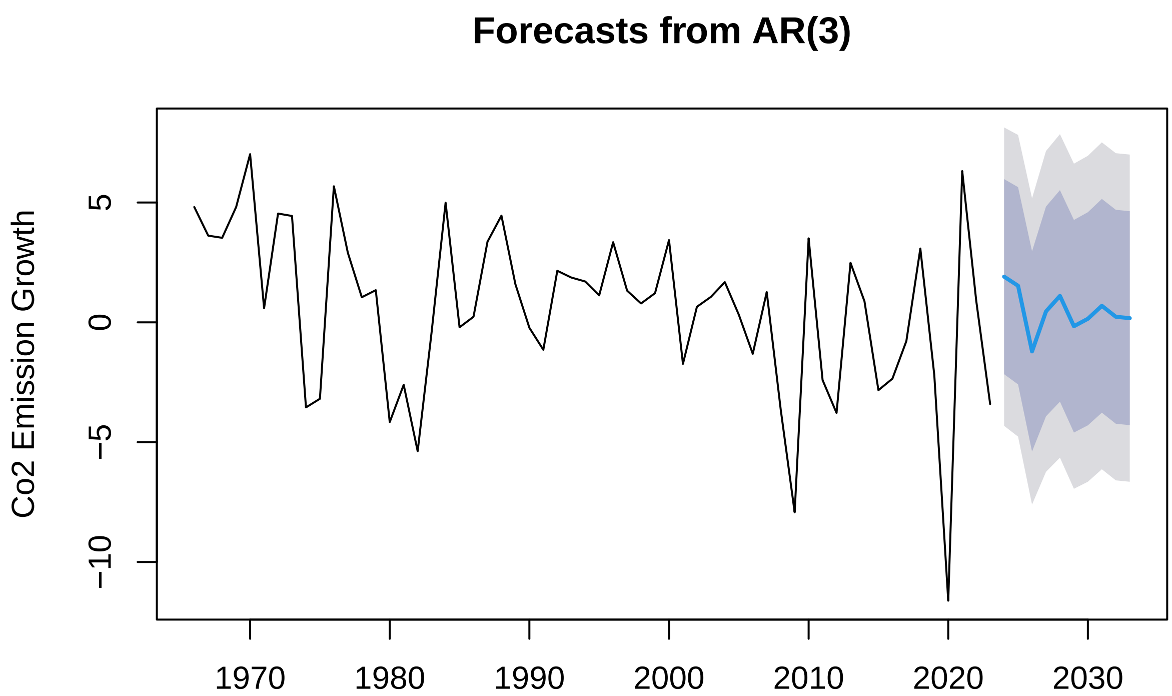
```r
plot(forecast(mod_3, 10),ylab = "Co2 Emission Growth")
```




```
mod_4 <- ar(co2_ts, aic=FALSE, order.max=3, method="ols")
forecast(mod_4, 10)
```

```
##      Point Forecast     Lo 80    Hi 80     Lo 95    Hi 95
## 2024      1.9064256 -2.164045 5.976896 -4.318822 8.131673
## 2025      1.5258565 -2.589241 5.640954 -4.767641 7.819354
## 2026     -1.2140686 -5.391309 2.963172 -7.602607 5.174470
## 2027      0.4548469 -3.921480 4.831174 -6.238168 7.147862
## 2028      1.1030801 -3.309387 5.515547 -5.645206 7.851366
## 2029     -0.1626452 -4.597526 4.272235 -6.945210 6.619919
## 2030      0.1487100 -4.297204 4.594624 -6.650729 6.948149
## 2031      0.6899541 -3.769471 5.149379 -6.130148 7.510057
## 2032      0.2323345 -4.230648 4.695317 -6.593208 7.057877
## 2033      0.1741438 -4.288895 4.637183 -6.651485 6.999773
```

```
plot(forecast(mod_4, 10),ylab = "Co2 Emission Growth")
```



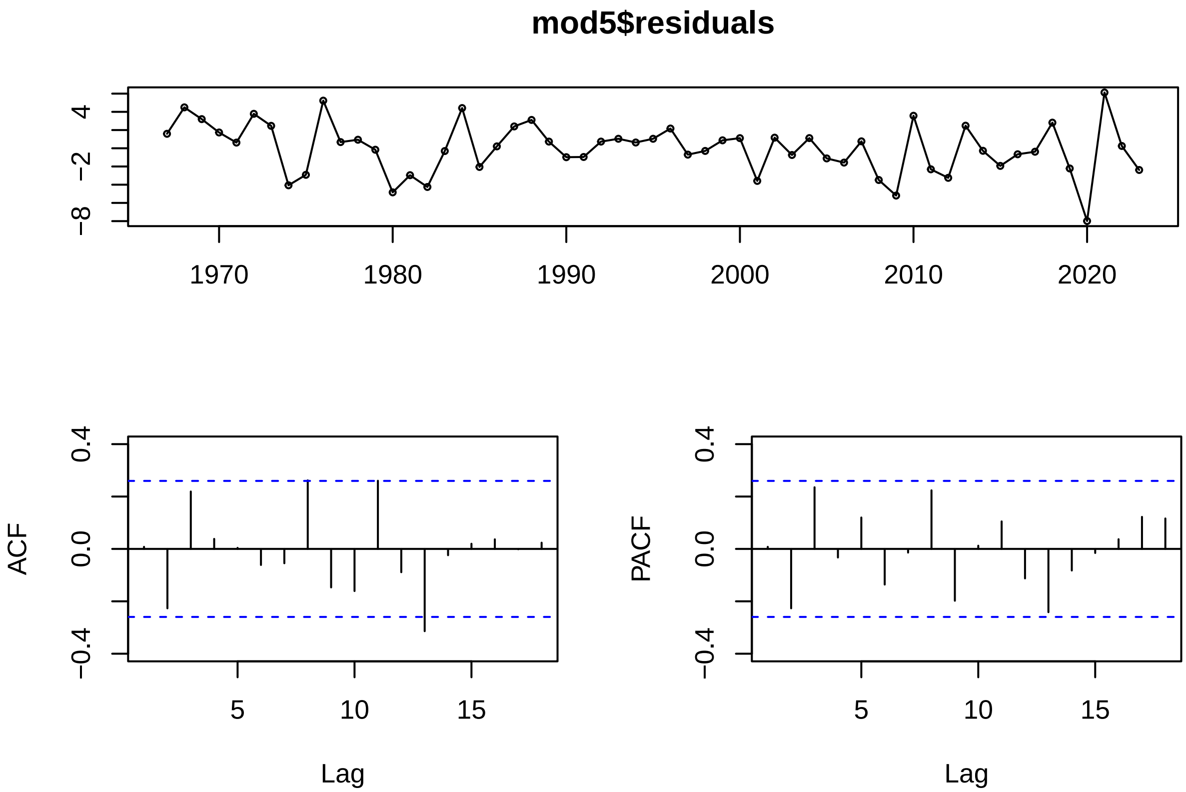
Change in Primary Energy Consumption (%)

AR(1)

```r
y3 <- energy_ts
mod5 <- dynlm(y3 ~ L(y3, 1))
summary(mod5)
```

```
##
## Time series regression with "ts" data:
## Start = 1967, End = 2023
##
## Call:
## dynlm(formula = y3 ~ L(y3, 1))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.9861 -1.9342  0.2527  1.5922  6.1239
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.7701     0.3980   1.935   0.0581 .
## L(y3, 1)      0.2005     0.1298   1.545   0.1281
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.8 on 55 degrees of freedom
## Multiple R-squared:  0.0416, Adjusted R-squared:  0.02417
## F-statistic: 2.387 on 1 and 55 DF,  p-value: 0.1281
```

```r
tsdisplay(mod5$residuals)
```



There is no evidence of serial correlation indicating the model is well-specified.

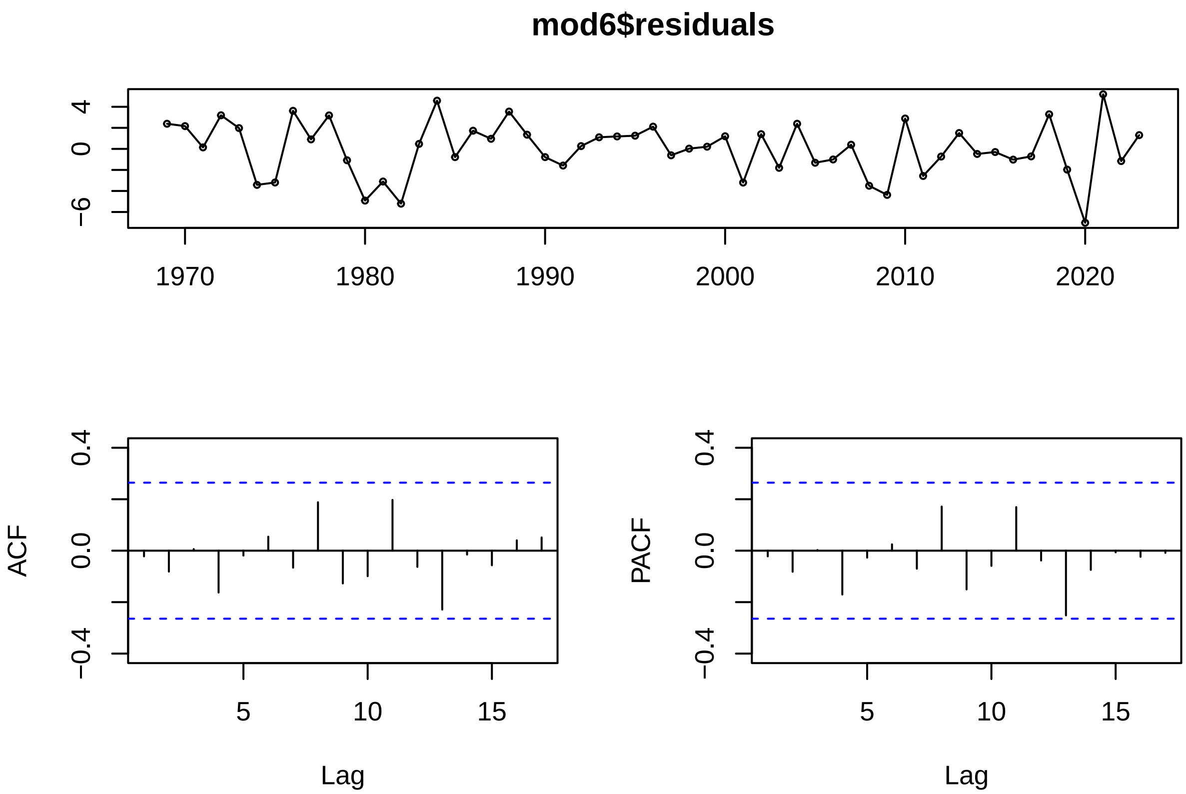
AR(3)

```r
mod6 <- dynlm(y3 ~ L(y3, 1:3))
summary(mod6)
```

```
##
## Time series regression with "ts" data:
## Start = 1969, End = 2023
##
## Call:
## dynlm(formula = y3 ~ L(y3, 1:3))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.0241 -1.2397  0.2073  1.6171  5.1901
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.5654     0.4078   1.386   0.1716
## L(y3, 1:3)1    0.2209     0.1290   1.712   0.0930 .
## L(y3, 1:3)2   -0.2345     0.1309  -1.792   0.0791 .
## L(y3, 1:3)3    0.3065     0.1284   2.386   0.0208 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.64 on 51 degrees of freedom
```

```
## Multiple R-squared:  0.1494, Adjusted R-squared:  0.0994
## F-statistic: 2.987 on 3 and 51 DF,  p-value: 0.0396
```

```
tsdisplay(mod6$residuals)
```



There is no evidence of serial correlation indicating the model is well-specified.

```
train_model_5 <- dynlm(energy_ts~ L(energy_ts, 1), data = train_data)
coef <- train_model_5$coefficients
energy_ts_1 <- active_ts[index:(nrow(active_df)-1),"energy_ts"]
forecast_energy <- coef[1] + coef[2]*energy_ts_1
f_errors5 <- test_data[,3] - forecast_energy

train_model_6 <- dynlm(co2_ts~ L(co2_ts, 1:3), data = train_data)
coef <- train_model_6$coefficients
energy_ts_2 <- active_ts[(index - 2):(nrow(active_df)-3),"energy_ts"]
energy_ts_3 <- active_ts[(index - 2):(nrow(active_df)-3),"energy_ts"]
forecast_energy <- coef[1] + coef[2]*energy_ts_1 + coef[3]*energy_ts_2 + coef[4]*energy_ts_3
f_errors6 <- test_data[, 3] - forecast_energy

rmse_5 <- sqrt(mean(f_errors5^2, na.rm = TRUE))
rmse_6 <- sqrt(mean(f_errors6^2, na.rm = TRUE))

print(paste("RMSE (1-lag model):", rmse_5))
```

```
## [1] "RMSE (1-lag model): 3.19091142049151"
```

```
print(paste("RMSE (3-lag model):", rmse_6))
```

```
## [1] "RMSE (3-lag model): 2.8842168615168"
```

The RMSE is marginally smaller for the AR(3) model indicating that it has higher prediction accuracy than the AR(1) model.

```
BIC(train_model_5)
```

```
## [1] 289.2282
```

```
BIC(train_model_6)
```

```
## [1] 303.2455
```

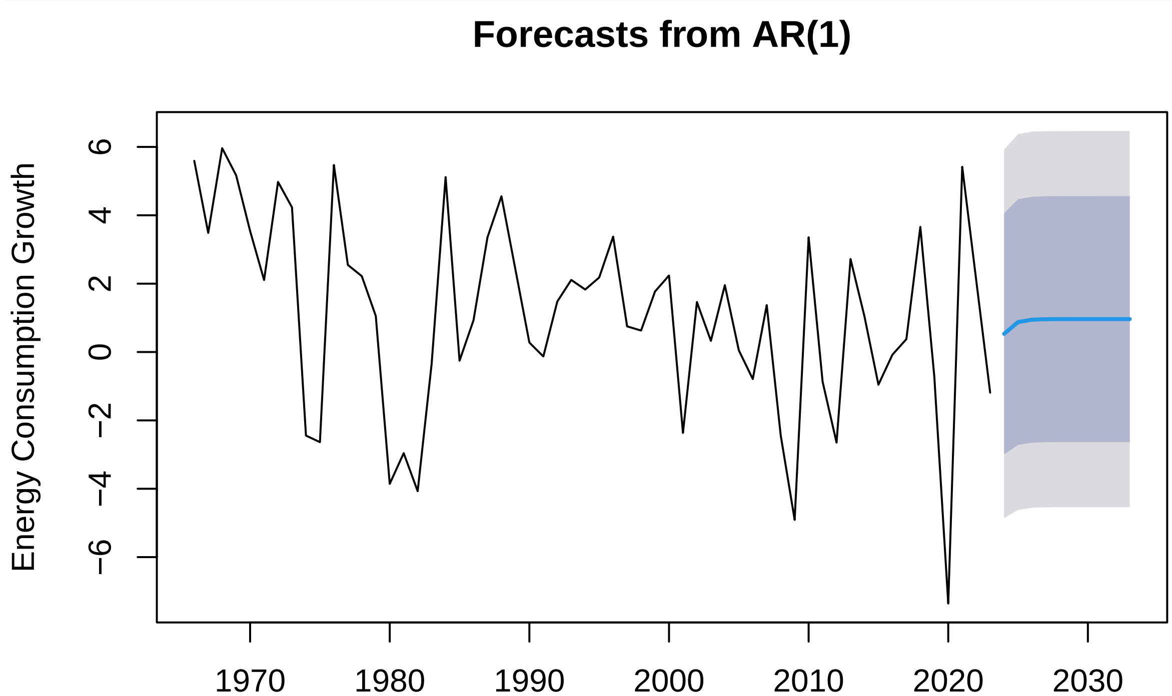
Despite the RMSE results, the BIC indicates that the AR(1) model is a better fit than the AR(3) model, potentially due to the fact that BIC punishes models that are over-fitted.

Forecasting

```
mod_5 <- ar(energy_ts, aic=FALSE, order.max=1, method="ols")
forecast(mod_5, 10)
```

```
##      Point Forecast      Lo 80    Hi 80     Lo 95    Hi 95
## 2024      0.5316845 -2.993126 4.056495 -4.859048 5.922417
## 2025      0.8766693 -2.718297 4.471635 -4.621357 6.374695
## 2026      0.9458409 -2.651917 4.543599 -4.556455 6.448137
## 2027      0.9597102 -2.638160 4.557580 -4.542757 6.462178
## 2028      0.9624911 -2.635384 4.560366 -4.539983 6.464965
## 2029      0.9630486 -2.634826 4.560923 -4.539426 6.465523
## 2030      0.9631604 -2.634714 4.561035 -4.539314 6.465635
## 2031      0.9631829 -2.634692 4.561058 -4.539292 6.465657
## 2032      0.9631874 -2.634687 4.561062 -4.539287 6.465662
## 2033      0.9631883 -2.634687 4.561063 -4.539286 6.465663
```

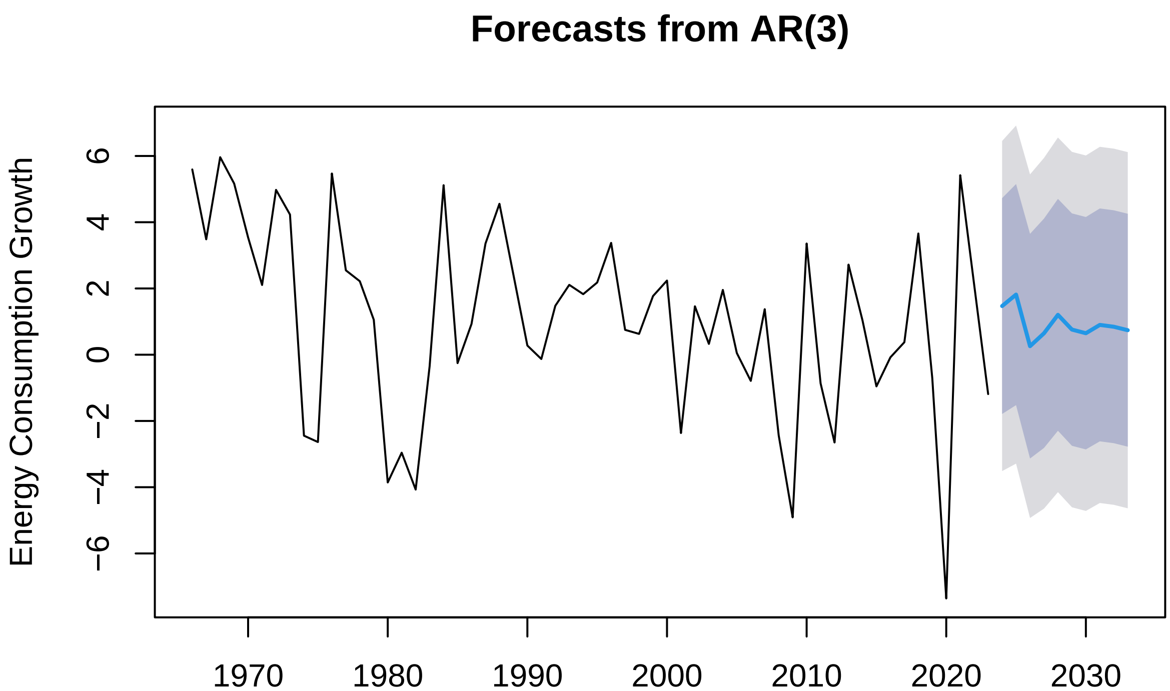
```
plot(forecast(mod_5, 10),ylab = "Energy Consumption Growth")
```




```
mod_6 <- ar(energy_ts, aic=FALSE, order.max=3, method="ols")
forecast(mod_6, 10)
```

```
##      Point Forecast      Lo 80    Hi 80     Lo 95    Hi 95
## 2024      1.4689249 -1.789405 4.727255 -3.514261 6.452111
## 2025      1.8150526 -1.521830 5.151935 -3.288268 6.918374
## 2026      0.2575284 -3.133750 3.648807 -4.928985 5.444041
## 2027      0.6469012 -2.815110 4.108912 -4.647788 5.941590
## 2028      1.2041845 -2.296105 4.704474 -4.149047 6.557416
## 2029      0.7586320 -2.749512 4.266776 -4.606612 6.123876
## 2030      0.6488831 -2.859495 4.157261 -4.716718 6.014484
## 2031      0.8999069 -2.615500 4.415313 -4.476444 6.276257
## 2032      0.8445343 -2.671021 4.360089 -4.532043 6.221112
## 2033      0.7398091 -2.776058 4.255676 -4.637245 6.116864
```

```r
plot(forecast(mod_6, 10),ylab = "Energy Consumption Growth")
```



Global average temperature anomaly relative to 1861-1890

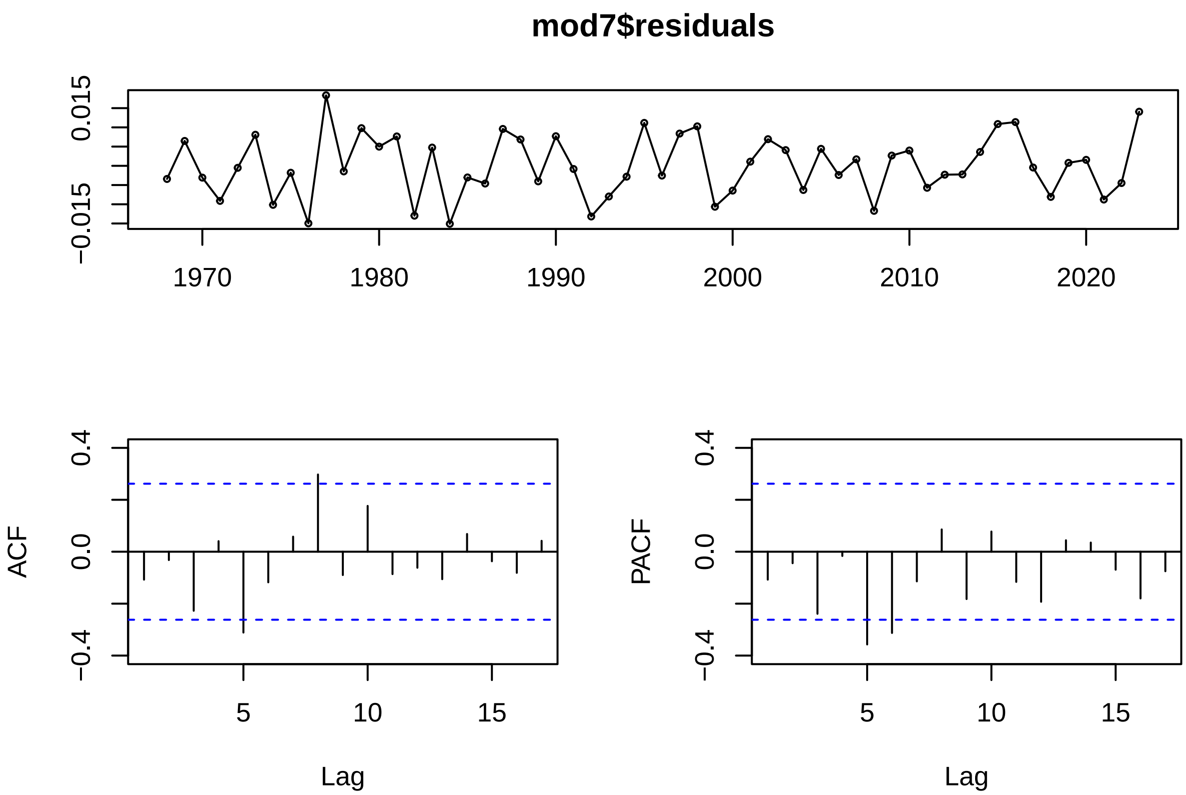
AR(2)

```r
y4 <- temp_ts
mod7 <- dynlm(y4 ~ L(y4, 1:2))
summary(mod7)
```

```
##
## Time series regression with "ts" data:
## Start = 1968, End = 2023
##
## Call:
## dynlm(formula = y4 ~ L(y4, 1:2))
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0150851 -0.0048815 -0.0006722  0.0065638  0.0183351
##
## Coefficients:
```

```
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.002560   0.001098   2.331 0.023600 *
## L(y4, 1:2)1 -0.501476   0.131444  -3.815 0.000357 ***
## L(y4, 1:2)2 -0.427311   0.131351  -3.253 0.001989 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.007997 on 53 degrees of freedom
## Multiple R-squared:  0.2606, Adjusted R-squared:  0.2327
## F-statistic: 9.338 on 2 and 53 DF,  p-value: 0.0003356
```

```r
tsdisplay(mod7$residuals)
```



The PACF exhibits signs of serial correlation at the lags 5 and 6. This indicates that the model is not well-specified.

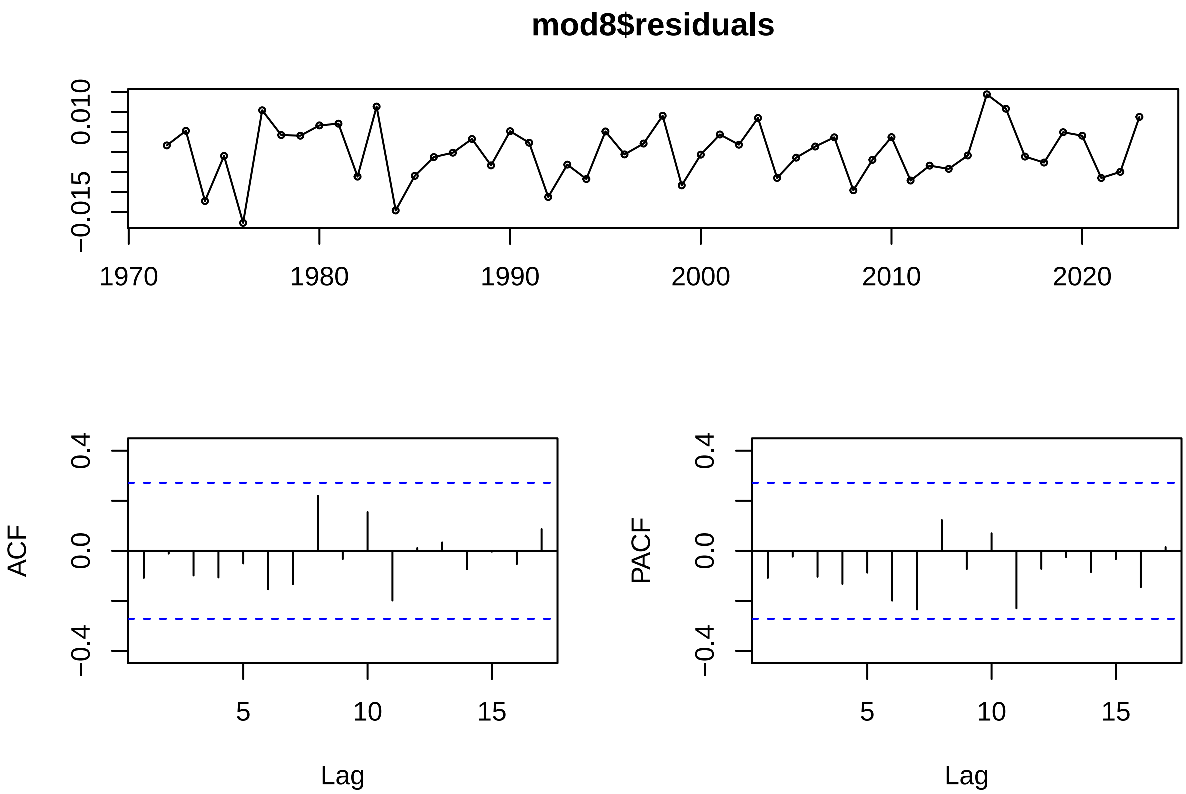
AR(6)

```r
mod8 <- dynlm(y4 ~ L(y4, 1:6))
summary(mod8)
```

```
##
## Time series regression with "ts" data:
## Start = 1972, End = 2023
##
## Call:
## dynlm(formula = y4 ~ L(y4, 1:6))
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0177072 -0.0044148 -0.0004043  0.0045018  0.0143812
```

```
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.004954   0.001312   3.776 0.000464 ***
## L(y4, 1:6)1 -0.628993   0.139370  -4.513 4.56e-05 ***
## L(y4, 1:6)2 -0.550349   0.155927  -3.530 0.000973 ***
## L(y4, 1:6)3 -0.443461   0.173206  -2.560 0.013885 *
## L(y4, 1:6)4 -0.206046   0.173285  -1.189 0.240653
## L(y4, 1:6)5 -0.416509   0.158716  -2.624 0.011818 *
## L(y4, 1:6)6 -0.387225   0.141513  -2.736 0.008862 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.007345 on 45 degrees of freedom
## Multiple R-squared:  0.4414, Adjusted R-squared:  0.367
## F-statistic: 5.928 on 6 and 45 DF,  p-value: 0.0001272
```

```r
tsdisplay(mod8$residuals)
```



There are no signs of serial correlation and therefore the indication is that it is well-specified.

```r
train_model_7 <- dynlm(temp_ts~ L(temp_ts, 1:2), data = train_data)
coef <- train_model_7$coefficients
temp_ts_1 <- active_ts[index:(nrow(active_df)-1),"temp_ts"]
temp_ts_2 <- active_ts[(index - 1):(nrow(active_df)-2),"temp_ts"]
forecast_temp <- coef[1] + coef[2]*temp_ts_1 + coef[3]*temp_ts_2
f_errors7 <- test_data[,4] - forecast_temp

train_model_8 <- dynlm(temp_ts~ L(temp_ts, 1:6), data = train_data)
coef <- train_model_8$coefficients
```

```r
temp_ts_3 <- active_ts[(index - 2):(nrow(active_df)-3),"temp_ts"]
temp_ts_4 <- active_ts[(index - 3):(nrow(active_df)-4),"temp_ts"]
temp_ts_5 <- active_ts[(index - 4):(nrow(active_df)-5),"temp_ts"]
temp_ts_6 <- active_ts[(index - 5):(nrow(active_df)-6),"temp_ts"]
forecast_temp <- coef[1] + coef[2]*temp_ts_1 + coef[3]*temp_ts_2 + coef[4]*temp_ts_3 + coef[5]*temp_ts_4
f_errors8 <- test_data[, 4] - forecast_temp

rmse_7 <- sqrt(mean(f_errors7^2, na.rm = TRUE))
rmse_8 <- sqrt(mean(f_errors8^2, na.rm = TRUE))

print(paste("RMSE (1-lag model):", rmse_7))
```

```
## [1] "RMSE (1-lag model): 0.00668642777057184"
```

```r
print(paste("RMSE (2-lag model):", rmse_8))
```

```
## [1] "RMSE (2-lag model): 0.00618855307005518"
```

The RMSE suggests the AR(6) model is more accurate in prediction compared to the AR(1) model.

```r
BIC(train_model_7)
```

```
## [1] -368.876
```

```r
BIC(train_model_8)
```

```
## [1] -339.3624
```

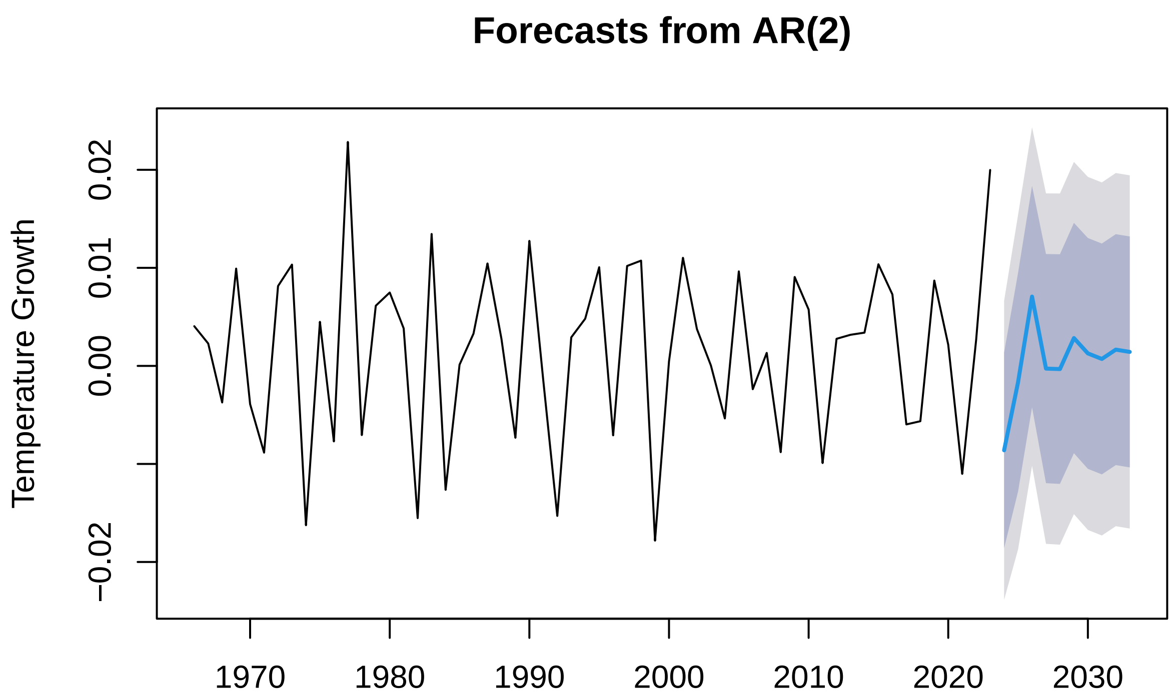
The BIC suggests the AR(1) model is a better fit compared to the AR(6) model, perhaps because the BIC penalizes models with more predictors.

Forecasting

```r
mod_7 <- ar(temp_ts, aic=FALSE, order.max=2, method="ols")
forecast(mod_7, 10)
```

```
##      Point Forecast        Lo 80       Hi 80       Lo 95       Hi 95
## 2024  -0.0086097762 -0.018579867 0.001360315 -0.02385771 0.006638162
## 2025  -0.0016619545 -0.012815442 0.009491533 -0.01871974 0.015395833
## 2026   0.0070721577 -0.004218261 0.018362576 -0.01019505 0.024339362
## 2027  -0.0002766660 -0.011962864 0.011409532 -0.01814916 0.017595831
## 2028  -0.0003235907 -0.012034679 0.011387498 -0.01823415 0.017586973
## 2029   0.0028401727 -0.008905901 0.014586246 -0.01512390 0.020804242
## 2030   0.0012736739 -0.010498294 0.013045642 -0.01673000 0.019277345
## 2031   0.0007073246 -0.011064644 0.012479293 -0.01729635 0.018710997
## 2032   0.0016607169 -0.010115914 0.013437348 -0.01635009 0.019671519
## 2033   0.0014246211 -0.010353207 0.013202450 -0.01658801 0.019437255
```

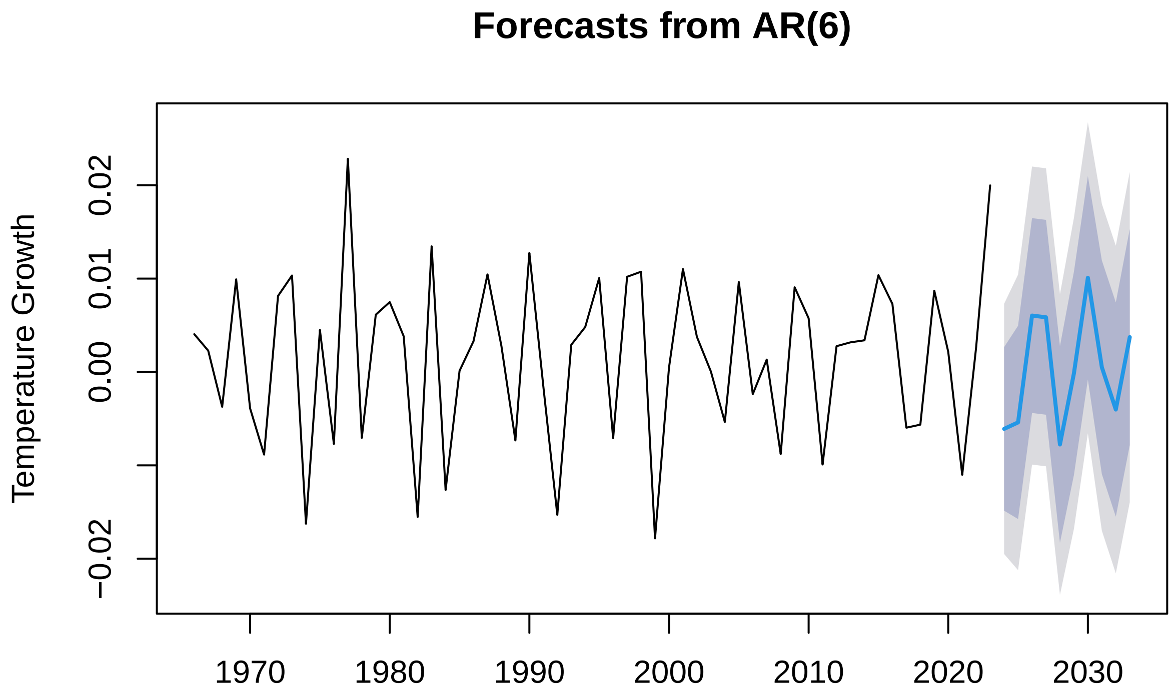
```r
plot(forecast(mod_7, 10),ylab = "Temperature Growth")
```




```r
mod_8 <- ar(temp_ts, aic=FALSE, order.max=6, method="ols")
forecast(mod_8, 10)
```

```
##      Point Forecast         Lo 80       Hi 80        Lo 95       Hi 95
## 2024  -0.0060960561 -0.0148529985 0.002660886 -0.019488644 0.007296531
## 2025  -0.0054001977 -0.0157453745 0.004944979 -0.021221780 0.010421385
## 2026   0.0060418561 -0.0043916623 0.016475375 -0.009914833 0.021998546
## 2027   0.0058558576 -0.0045776608 0.016289376 -0.010100832 0.021812547
## 2028  -0.0077673266 -0.0182922109 0.002757558 -0.023863748 0.008329095
## 2029  -0.0001488447 -0.0110352266 0.010737537 -0.016798129 0.016500440
## 2030   0.0100904523 -0.0007972676 0.020978172 -0.006560878 0.026741783
## 2031   0.0005016800 -0.0109369889 0.011940349 -0.016992254 0.017995614
## 2032  -0.0040269058 -0.0154954153 0.007441604 -0.021566477 0.013512666
## 2033   0.0037344340 -0.0078303394 0.015299207 -0.013952361 0.021421229
```

```r
plot(forecast(mod_8, 10),ylab = "Temperature Growth")
```


Forecasts from AR(6)
Temperature Growth
0.02
0.01
0.00
−0.02
1970
1980
1990
2000
2010
2020
2030

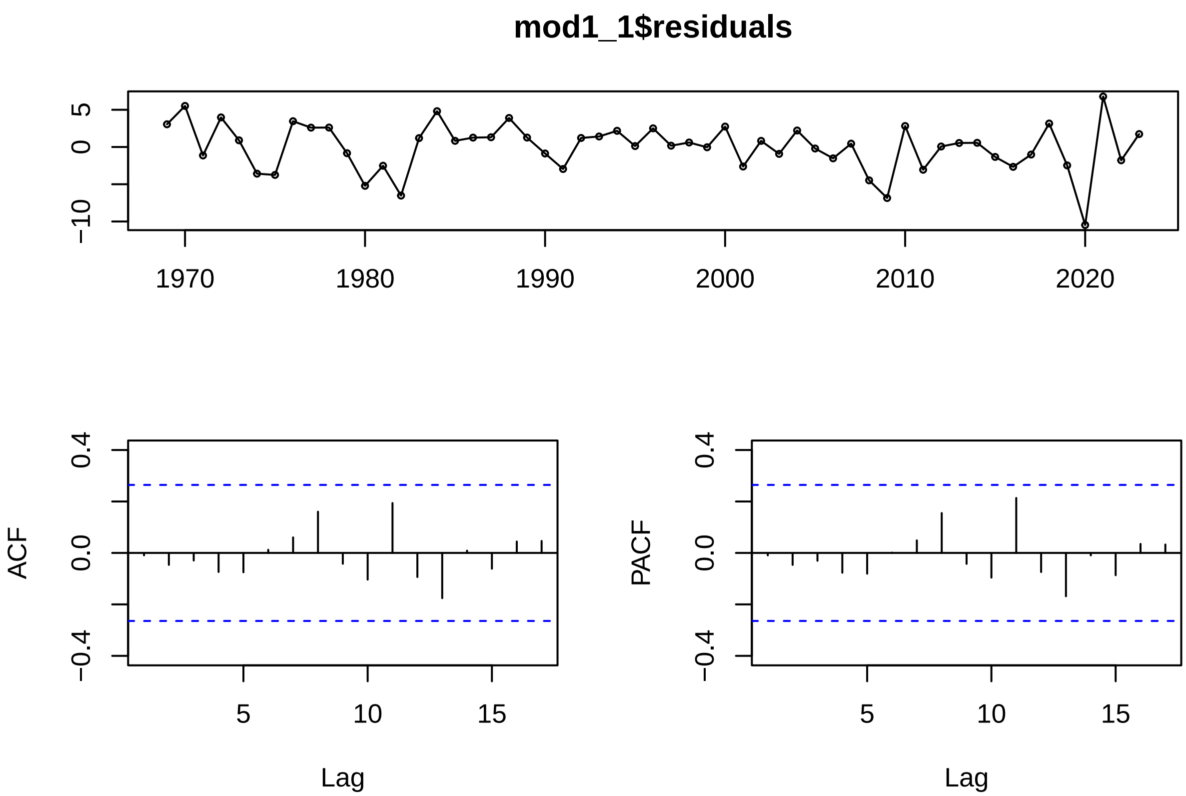
(4) Autoregressive Distributed Lag Models

Change in Co2 Emissions

```
mod1_1 <- dynlm(co2_ts~ L(co2_ts, 1:3) + L(oil_prod_ts, 1))
mod1_2 <- dynlm(co2_ts~ L(co2_ts, 1:2) + L(energy_ts, 3))
```

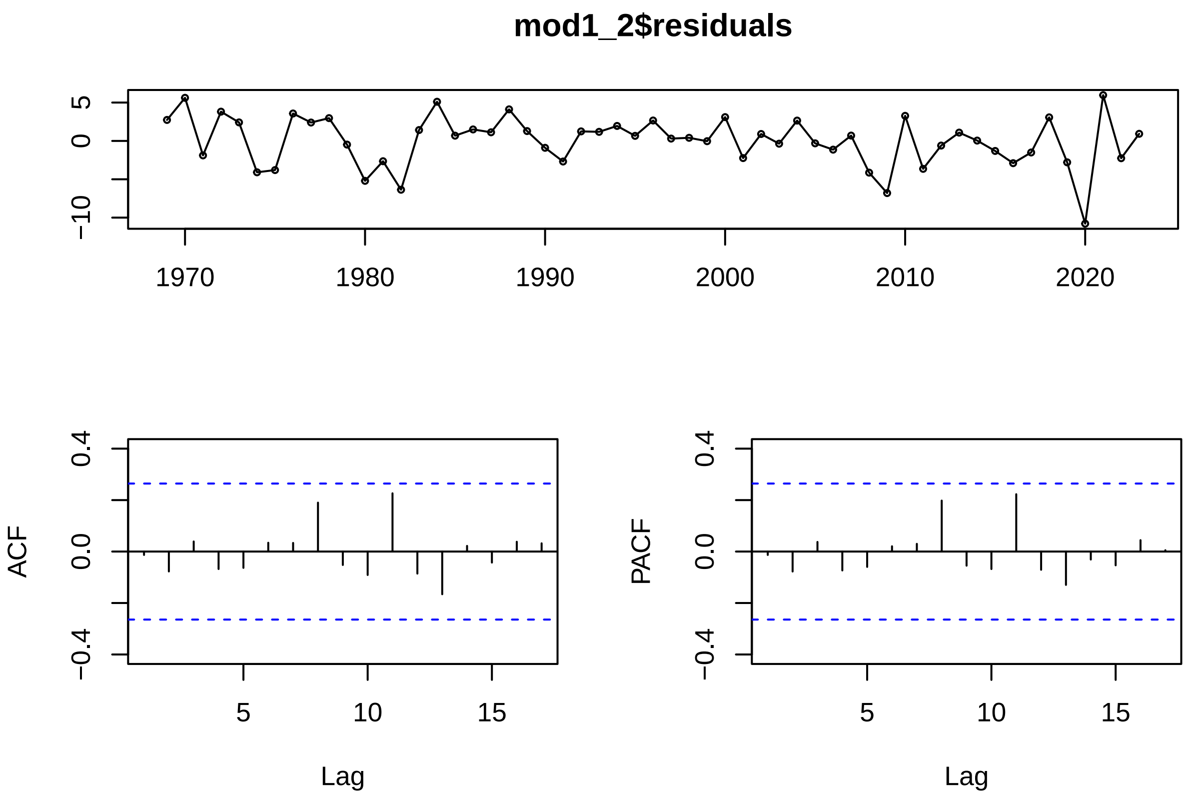
We chose to test two models. The first being an ARDL(3,1) and the second being an ARDL(2,3).

```
tsdisplay(mod1_1$residuals)
```



The ACF and PACF show no signs of serial correlation.

```
tsdisplay(mod1_2$residuals)
```



Again for model 2, The ACF and PACF show no signs of serial correlation.

```
train_model_1_1 <- dynlm(co2_ts~ L(co2_ts, 1:3) + L(oil_prod_ts, 1), data = train_data)
coef <- train_model_1_1$coefficients
co2_ts_1 <- active_ts[index:(nrow(active_ts)-1),"co2_ts"]
co2_ts_2 <- active_ts[(index - 1):(nrow(active_ts)-2),"co2_ts"]
co2_ts_3 <- active_ts[(index - 2):(nrow(active_ts)-3),"co2_ts"]
oil_prod_ts_1 <- active_ts[index:(nrow(active_ts)-1),"oil_prod_ts"]
forecast_co2 <- coef[1] + coef[2]*co2_ts_1 + coef[3]*co2_ts_2 + coef[4]*co2_ts_3 + coef[5]*oil_prod_ts_
f_errors1_1 <- test_data[,2] - forecast_co2

train_model_1_2 <- dynlm(co2_ts~ L(co2_ts, 1:2) + L(energy_ts, 3), data = train_data)
coef <- train_model_1_2$coefficients
energy_ts_3 <- active_ts[(index - 2):(nrow(active_ts)-3),"energy_ts"]
forecast_co2 <- coef[1] + coef[2]*co2_ts_1 + coef[3]*co2_ts_2 + coef[4]*energy_ts_3
f_errors1_2 <- test_data[, 2] - forecast_co2

rmse_1_1 <- sqrt(mean(f_errors1_1^2, na.rm = TRUE))
rmse_1_2 <- sqrt(mean(f_errors1_2^2, na.rm = TRUE))

print(paste("RMSE ARDL(3,1):", rmse_1_1))
```

```
## [1] "RMSE ARDL(3,1): 3.74269879551345"
```

```
print(paste("RMSE ARDL(2,3):", rmse_1_2))
```

```
## [1] "RMSE ARDL(2,3): 3.77609165789345"
```

The RMSE for our first model is marginally smaller indicating that it may be slightly more accurate for prediction.

```
BIC(train_model_1_1)
```

```
## [1] 307.0367
```

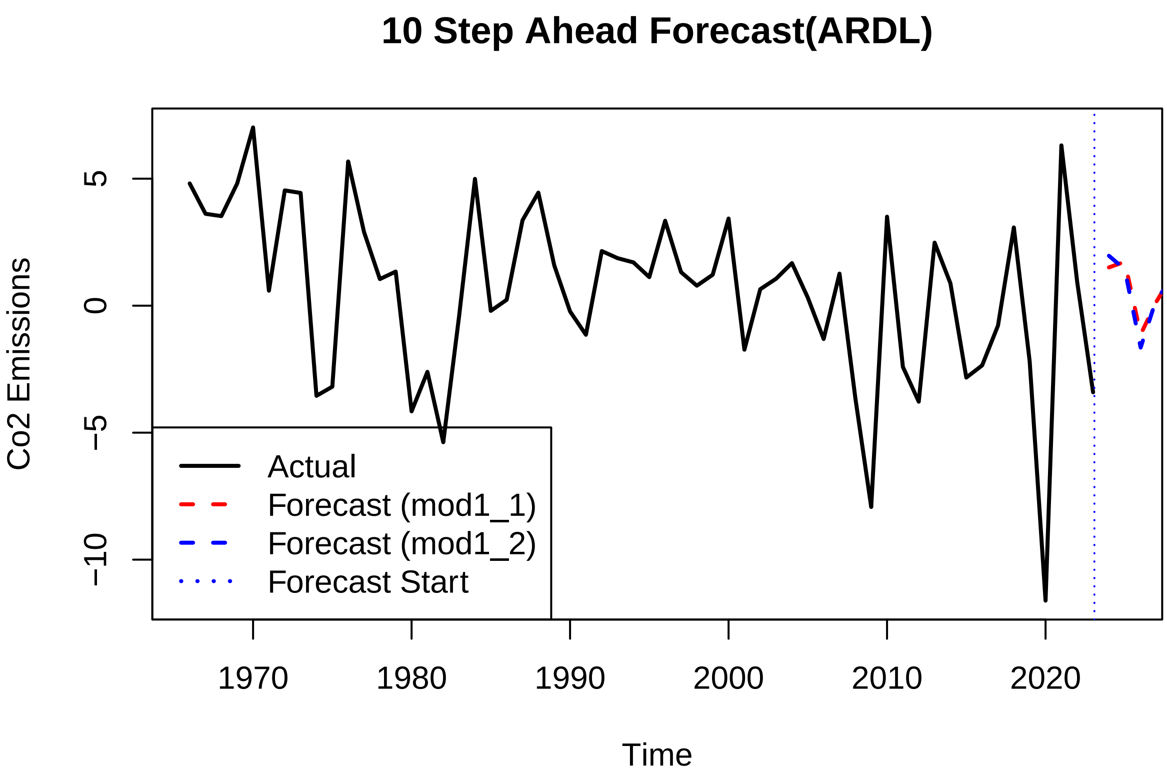
```
BIC(train_model_1_2)
```

```
## [1] 304.7521
```

The BIC for the second model is smaller than the BIC for the first model which is an indication that it may be a better fit. Both models are similarly effective at modeling carbon emissions but if forced to pick one, we would choose the 2nd because the BIC is smaller.

Forecast:

```
forecast_recursive <- function(model, data, n_ahead = 10) {
  coefficients <- coef(model)
  last_values <- tail(data, length(coefficients)-1)
  forecasts <- numeric(n_ahead)

  for(i in 1:n_ahead){
    forecasts[i] <- coefficients[1] + sum(coefficients[2:length(coefficients)] * rev(last_values))
    last_values <- c(last_values[-1], forecasts[i])}

  return(forecasts)
}

forecasts_mod1_1 <- forecast_recursive(mod1_1, co2_ts, n_ahead=10)
forecast_start <- end(co2_ts) + c(0, 1)
forecast_ts_mod1_1 <- ts(forecasts_mod1_1, start=forecast_start, frequency=1)

forecasts_mod1_2 <- forecast_recursive(mod1_2, co2_ts, n_ahead=10)
forecast_ts_mod1_2 <- ts(forecasts_mod1_2, start=forecast_start, frequency=1)

plot(co2_ts, xlim=c(start(co2_ts)[1], end(co2_ts)[1]+2), ylim=range(c(co2_ts,forecasts_mod1_1, forecast
lines(forecast_ts_mod1_1,col="red",lwd=2,lty=2)
lines(forecast_ts_mod1_2,col="blue",lwd=2,lty=2)
abline(v=end(co2_ts)[1]+(end(co2_ts)[2]/12),col="blue",lty=3)
legend("bottomleft",legend=c("Actual","Forecast (mod1_1)","Forecast (mod1_2)","Forecast Start"),col=c("
```

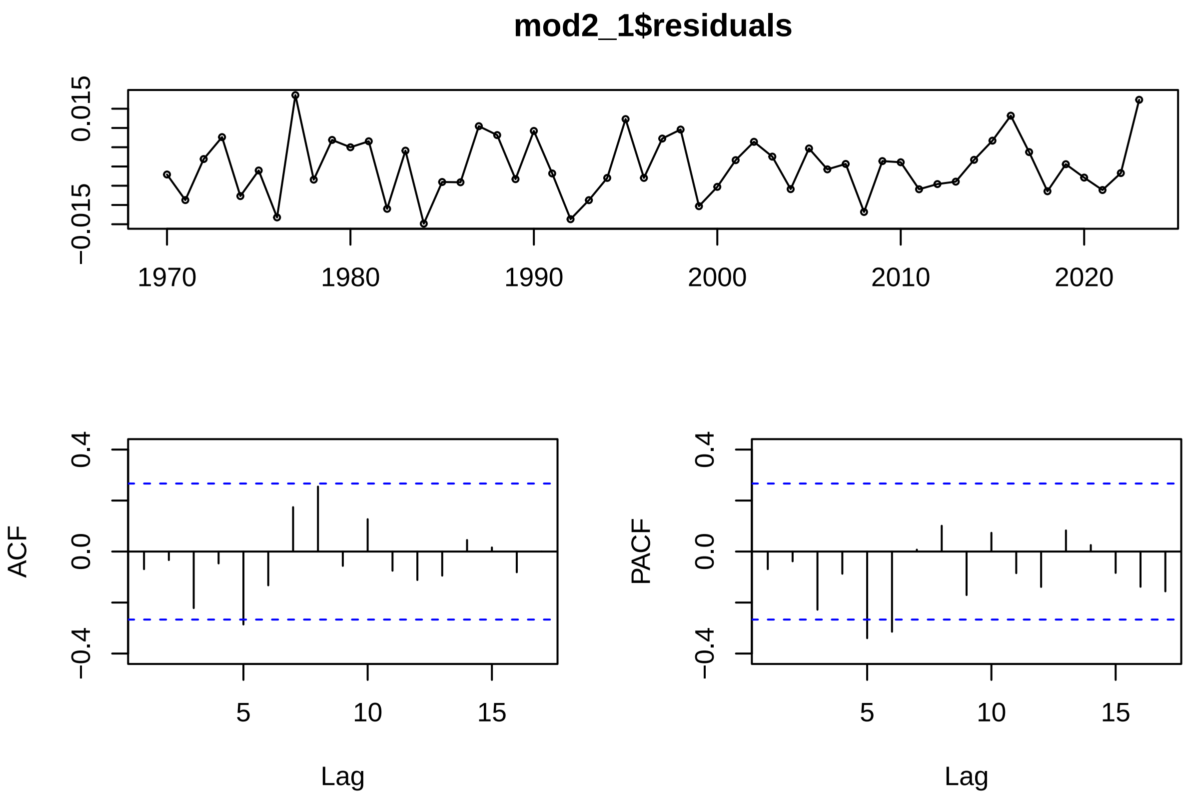


Change in Global Temperature

```
mod2_1 <- dynlm(temp_ts~ L(temp_ts, 1:2) + L(oil_prod_ts, 1:3))
mod2_2 <- dynlm(temp_ts~ L(temp_ts, 1:2) + L(energy_ts, 1:2))
```

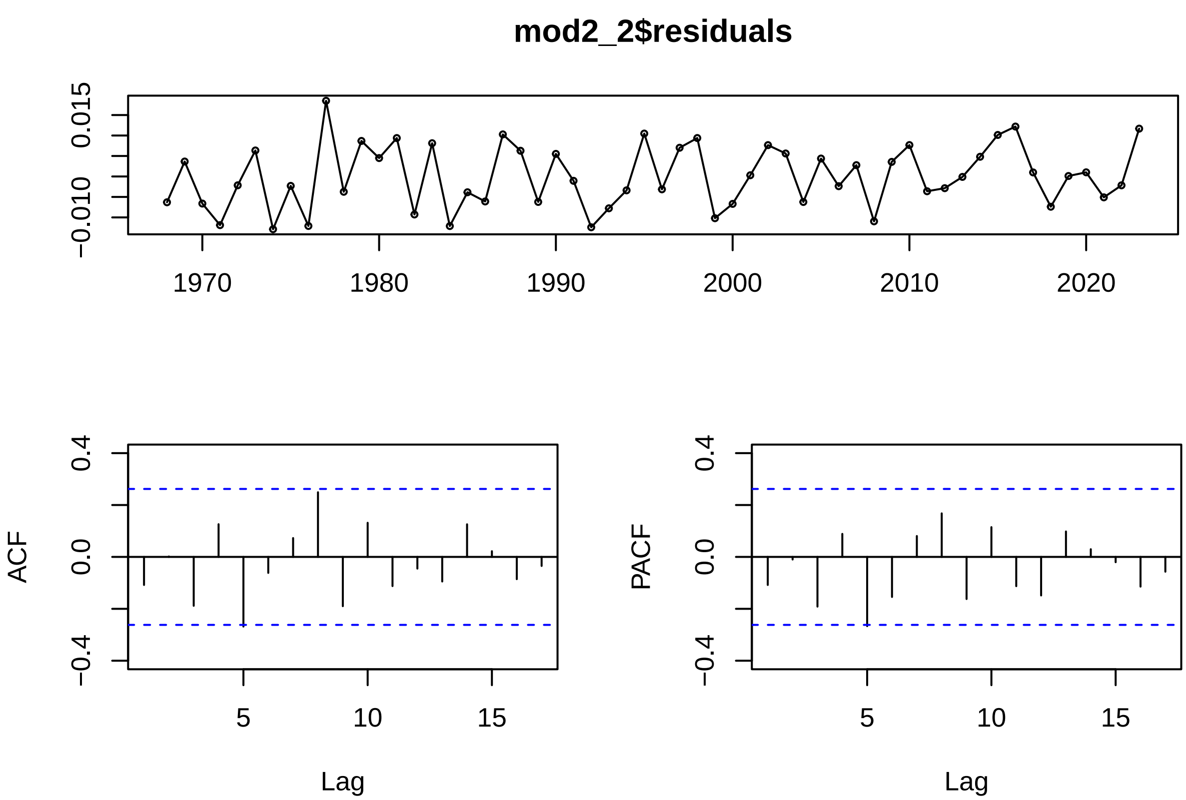
We chose two models: ARDL(2,3) for oil production as the DL and ARDL(2,2) for energy consumption as the DL.

```
tsdisplay(mod2_1$residuals)
```



There is evidence of serial correlation from the ACF and PACF which indicates this may not be the best model.

```
tsdisplay(mod2_2$residuals)
```



Lag 5 presents signs of serial correlation, however it is only just over the threshold so we are ignoring it.

```
train_model_2_1 <- dynlm(temp_ts~ L(temp_ts, 1:2) + L(oil_prod_ts, 1:3), data = train_data)
coef <- train_model_2_1$coefficients
temp_ts_1 <- active_ts[index:(nrow(active_ts)-1),"temp_ts"]
temp_ts_2 <- active_ts[(index - 1):(nrow(active_ts)-2),"temp_ts"]
oil_prod_ts_1 <- active_ts[index:(nrow(active_ts)-1),"oil_prod_ts"]
oil_prod_ts_2 <- active_ts[(index - 1):(nrow(active_ts)-2),"oil_prod_ts"]
oil_prod_ts_3 <- active_ts[(index - 2):(nrow(active_ts)-3),"oil_prod_ts"]
forecast_temp <- coef[1] + coef[2]*temp_ts_1 + coef[3]*temp_ts_2 + coef[4]*oil_prod_ts_1 + coef[5]*oil_
f_errors2_1 <- test_data[,4] - forecast_temp

train_model_2_2 <- dynlm(co2_ts~ L(temp_ts, 1:2) + L(energy_ts, 1:2), data = train_data)
coef <- train_model_2_2$coefficients
energy_ts_1 <- active_ts[index:(nrow(active_ts)-1),"energy_ts"]
energy_ts_2 <- active_ts[(index - 1):(nrow(active_ts)-2),"energy_ts"]
forecast_temp <- coef[1] + coef[2]*temp_ts_1 + coef[3]*temp_ts_2 + coef[4]*energy_ts_1 + coef[5]*energy_
f_errors2_2 <- test_data[,4] - forecast_temp

rmse_2_1 <- sqrt(mean(f_errors2_1^2, na.rm = TRUE))
rmse_2_2 <- sqrt(mean(f_errors2_2^2, na.rm = TRUE))

print(paste("RMSE ARDL(2,3):", rmse_2_1))
```

```
## [1] "RMSE ARDL(2,3): 0.00670009778931435"
```

```
print(paste("RMSE ARDL(2,2):", rmse_2_2))
```

```
## [1] "RMSE ARDL(2,2): 1.05563064801098"
```

The RMSE for model 1 is significantly smaller than the RMSE for model 2 indicating it may be the better model in regards to prediction accuracy.

```
BIC(train_model_2_1)
```

```
## [1] -344.8936
```

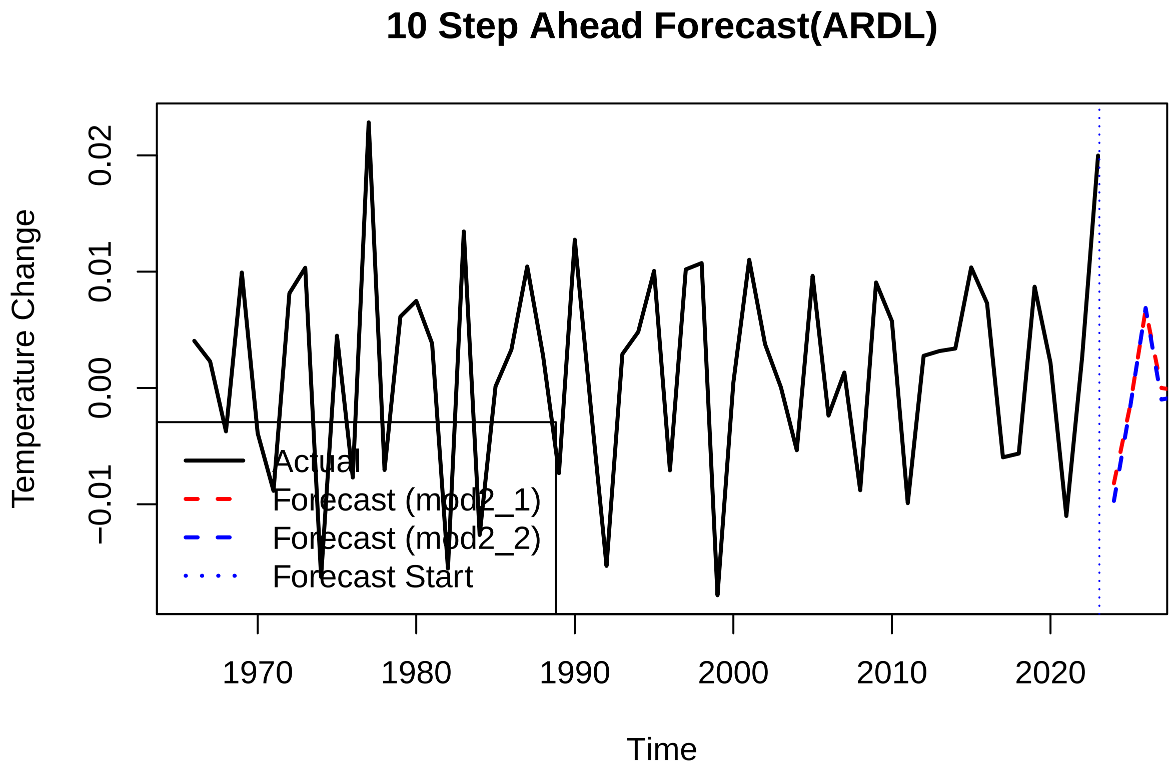
```
BIC(train_model_2_2)
```

```
## [1] 319.3337
```

The BIC for model 1 is significantly smaller than the BIC for model 2, so we can conclusively say it is the better model.

Forecasts:

```
forecasts_mod2_1 <- forecast_recursive(mod2_1, temp_ts, n_ahead=10)
forecast_start <- end(temp_ts) + c(0, 1)
forecast_ts_mod2_1 <- ts(forecasts_mod2_1, start=forecast_start, frequency=1)

forecasts_mod2_2 <- forecast_recursive(mod2_2, temp_ts, n_ahead=10)
forecast_ts_mod2_2 <- ts(forecasts_mod2_2, start=forecast_start, frequency=1)

plot(temp_ts, xlim=c(start(temp_ts)[1], end(temp_ts)[1]+2), ylim=range(c(temp_ts,forecasts_mod2_1, fore
lines(forecast_ts_mod2_1,col="red",lwd=2,lty=2)
lines(forecast_ts_mod2_2,col="blue",lwd=2,lty=2)
abline(v=end(co2_ts)[1]+(end(co2_ts)[2]/12),col="blue",lty=3)
legend("bottomleft",legend=c("Actual","Forecast (mod2_1)","Forecast (mod2_2)","Forecast Start"),col=c("
```


10 Step Ahead Forecast(ARDL)
Temperature Change
0.02
0.01
0.00
−0.01
Actual
Forecast (mod2_1)
Forecast (mod2_2)
Forecast Start
1970
1980
1990
2000
2010
2020
Time

(5) Vector Autoregressive Models

```r
full_df <- read_csv('carbon_temp_1900.csv')
```

```
## Rows: 175 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): Entity
## dbl (3): Year, carbon_growth, temp_growth
##
## 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.
```

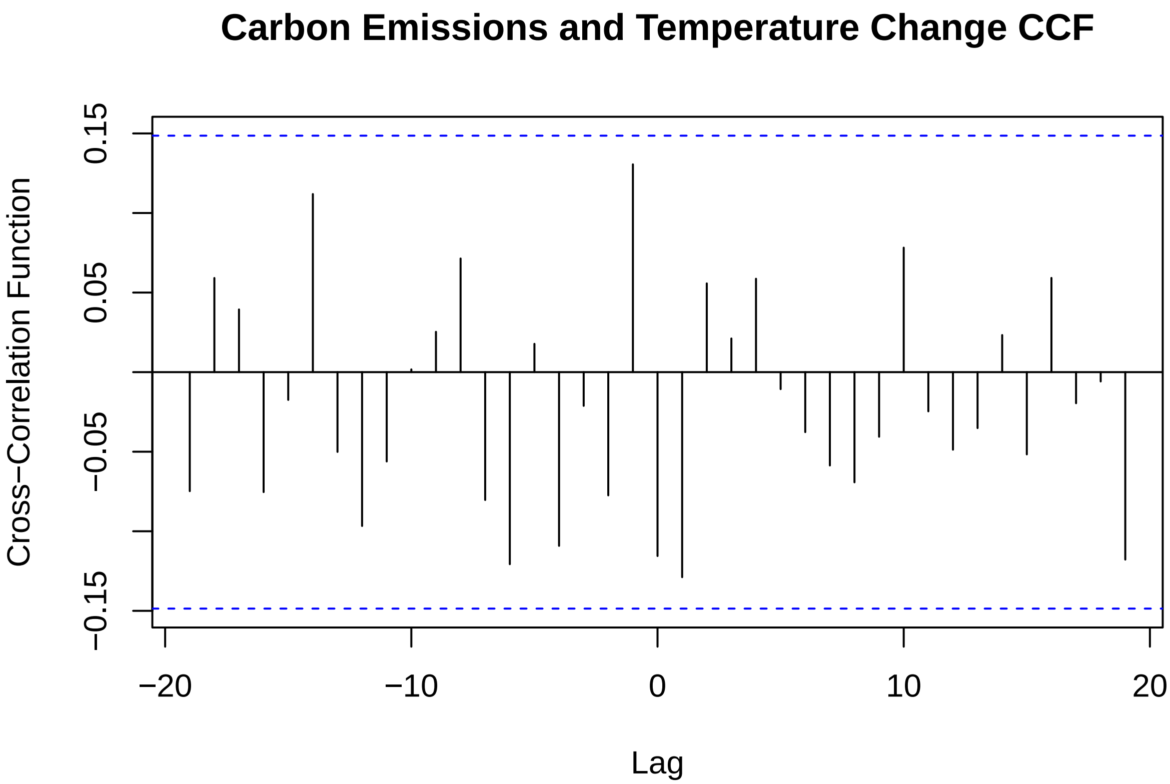
```r
temp_1900 <- full_df[4]
temp_1900 <- temp_1900[-175, ]
carbon_1900 <- full_df[3]
carbon_1900 <- carbon_1900[-175, ]
carbon_1900_ts <- ts(carbon_1900, start=1850, freq=1)
temp_1900_ts <- ts(temp_1900, start=1850, freq=1)

var_df <- data.frame(cbind(carbon_1900_ts, temp_1900_ts))
VARselect(var_df, lag.max = 10)
```

```
## $selection
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      3      3      1      3
##
## $criteria
##                   1            2            3            4            5
## AIC(n) -5.531773152 -5.582264708 -5.657679244 -5.634155238 -5.626354000
## HQ(n)  -5.485733078 -5.505531251 -5.550252404 -5.496035015 -5.457540395
## SC(n)  -5.418363405 -5.393248463 -5.393056500 -5.293925995 -5.210518260
## FPE(n)  0.003958995  0.003764174  0.003490971  0.003574484  0.003603136
##                   6            7            8            9           10
## AIC(n) -5.610125351 -5.634516855 -5.620216055 -5.602508147 -5.576565555
## HQ(n)  -5.410618362 -5.404316483 -5.359322300 -5.310921010 -5.254285035
## SC(n)  -5.118683113 -5.067468118 -4.977560820 -4.884246414 -4.782697324
## FPE(n)  0.003663052  0.003576068  0.003629279  0.003696318  0.003796262
```

The selection criterion suggest using a lag of 3 (all except SC).

```r
var_mod <- VAR(var_df, p=3)
```

```r
ccf(as.vector(carbon_1900_ts), as.vector(temp_1900_ts), ylab="Cross-Correlation Function", main = "Carb
```



There are no significant lags which suggests neither time series variable has a significant influence on the other.

```r
grangertest(carbon_1900_ts ~ temp_1900_ts, order = 14)
```

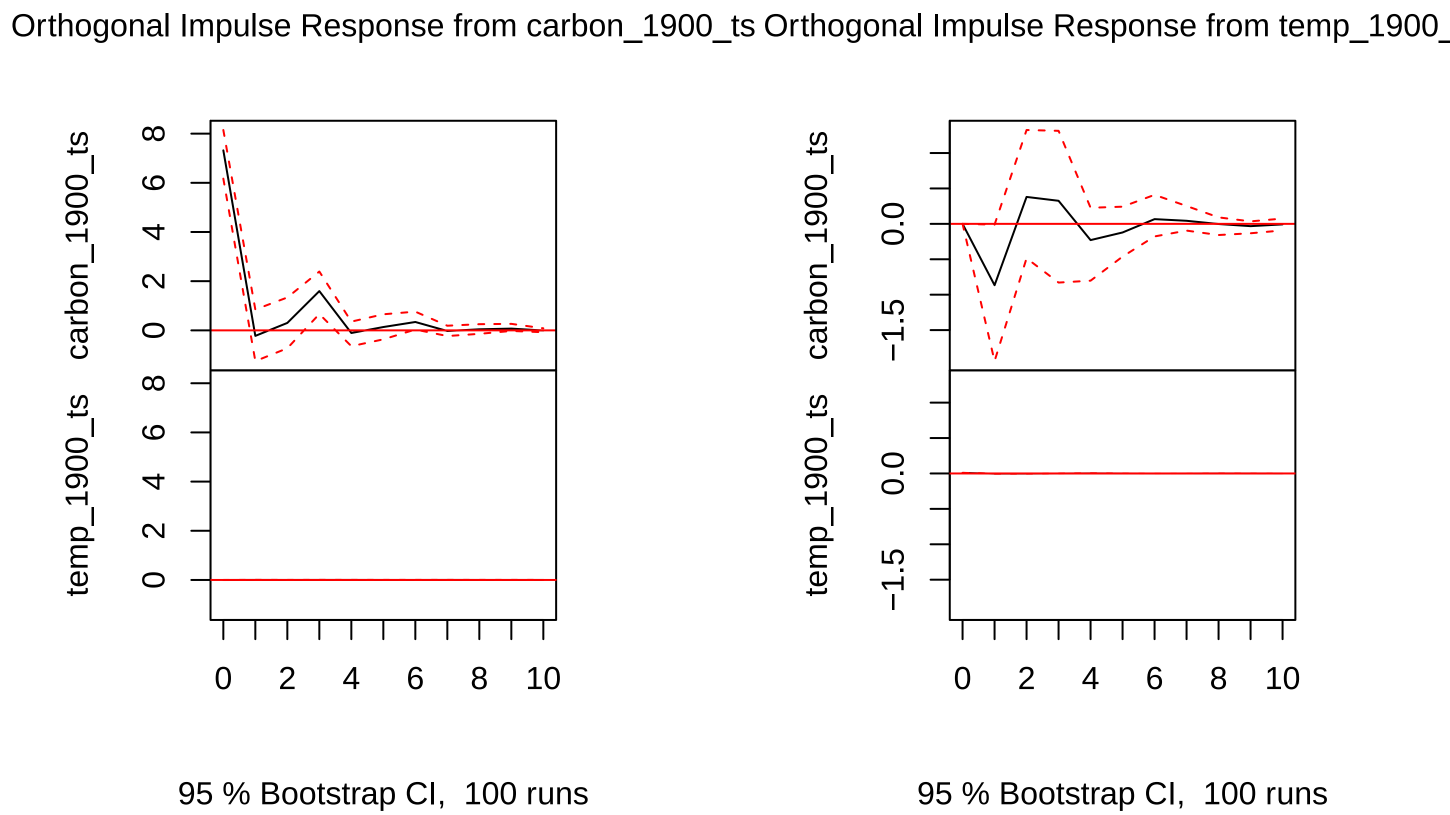
```
## Granger causality test
##
## Model 1: carbon_1900_ts ~ Lags(carbon_1900_ts, 1:14) + Lags(temp_1900_ts, 1:14)
## Model 2: carbon_1900_ts ~ Lags(carbon_1900_ts, 1:14)
##   Res.Df  Df      F Pr(>F)
## 1    131
## 2    145 -14 0.5949 0.8652
```

```r
grangertest(temp_1900_ts ~ carbon_1900_ts, order = 14)
```

```
## Granger causality test
##
## Model 1: temp_1900_ts ~ Lags(temp_1900_ts, 1:14) + Lags(carbon_1900_ts, 1:14)
## Model 2: temp_1900_ts ~ Lags(temp_1900_ts, 1:14)
##   Res.Df  Df      F  Pr(>F)
## 1    131
## 2    145 -14 1.9915 0.02298 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The Granger test suggests that carbon emissions have a greater influence on temperature rather than the other way around. This, however, is barely significant so we can assume the influence is not very strong.

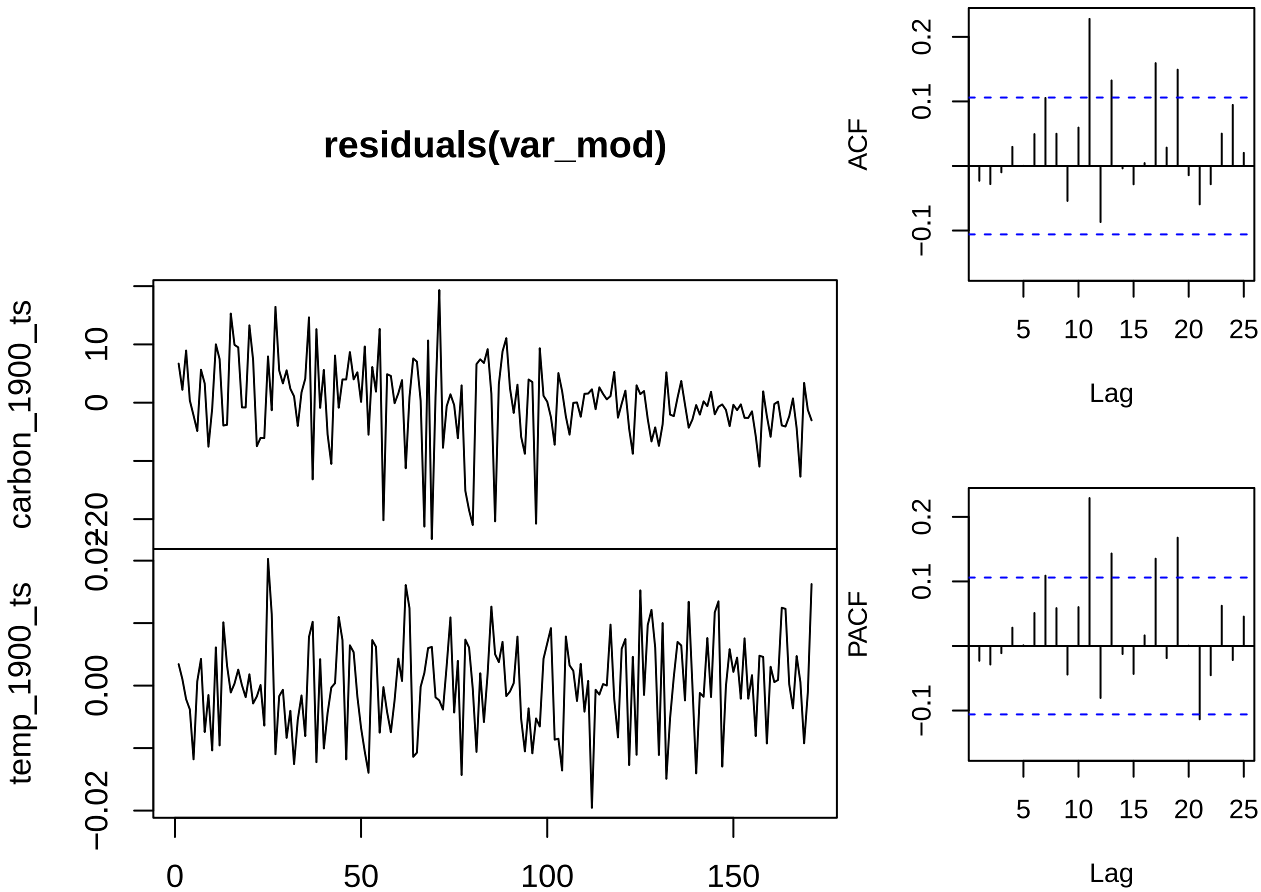
```r
plot(irf(var_mod, n.ahead=10))
```



From the IRF from carbon emissions, we can see that it has a large but short-lived effect on itself. It does not seem to influence temperature.

From the IRF from temperature, we can see that it has a little to no effect on itself. It does appear to have an effect on carbon emissions, however, due to the high degree of uncertainty from our wide confidence intervals — and the fact they cross zero — we cannot take anything meaningful from it.

```r
tsdisplay(residuals(var_mod))
```



The residuals show indicate the model is not well-specified. There are a significant number of lags outside of the threshold indicating the existence of serial correlation.

```
index_var <- floor(2/3 * nrow(var_df))
train_data_var <- var_df[1:index_var, ]
test_data_var <- var_df[(index_var+1):nrow(var_df), ]

VARselect(train_data_var, lag.max = 10)
```

```
## $selection
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      3      1      1      3
##
## $criteria
##                   1            2            3            4            5
## AIC(n) -5.248759028 -5.258656683 -5.320664043 -5.285994430 -5.245384017
## HQ(n)  -5.187654895 -5.156816460 -5.178087731 -5.102682030 -5.021335528
## SC(n)  -5.097998325 -5.007388844 -4.968889068 -4.833712320 -4.692594771
## FPE(n)  0.005254193  0.005203017  0.004891389  0.005066147  0.005279709
##                   6           7            8            9           10
## AIC(n) -5.201570201 -5.24718575 -5.215019466 -5.171881578 -5.113339440
## HQ(n)  -4.936785623 -4.94166508 -4.868762709 -4.784888733 -4.685610505
## SC(n)  -4.548273819 -4.49338224 -4.360708813 -4.217063790 -4.058014515
## FPE(n)  0.005521597  0.00528247  0.005464772  0.005718484  0.006080182
```

```
train_mod_var <- VAR(train_data_var, p = 3)
```

```
var_forecast <- predict(train_mod_var, n.ahead = nrow(test_data_var))
predictions <- as.data.frame(var_forecast$fcst)

mse_carbon_1900_ts <- mean((test_data_var[,1] - predictions$carbon_1900_ts.fcst)^2)
mse_temp_1900_ts <- mean((test_data_var[,2] - predictions$temp_1900_ts.fcst)^2)

rmse_carbon_1900_ts <- sqrt(mse_carbon_1900_ts)
rmse_temp_1900_ts <- sqrt(mse_temp_1900_ts)

print(paste("RMSE Carbon:", rmse_carbon_1900_ts))
```

```
## [1] "RMSE Carbon: 4.72601464410711"
```

```
print(paste("RMSE Temp:", rmse_temp_1900_ts))
```

```
## [1] "RMSE Temp: 0.00905919173391068"
```

The RMSE for temp is significantly smaller than the RMSE for carbon emissions. Based on this one alone we could conclude that it it is more accurate. However, this is most likely due to the fact that temperature growth is a lot smaller than the growth in carbon emissions.

```
print(paste("RMSE Carbon:", rmse_carbon_1900_ts / mean(carbon_1900_ts)))
```

```
## [1] "RMSE Carbon: 1.64085409345783"
```

```
print(paste("RMSE Temp:", rmse_temp_1900_ts / mean(temp_1900_ts)))
```

```
## [1] "RMSE Temp: 15.6560959072437"
```

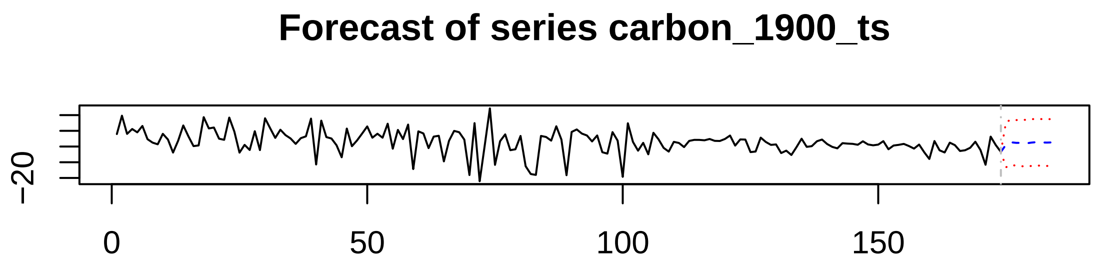
Following a mean normalization, we find that the predictions for carbon emissions growth are far more accurate.

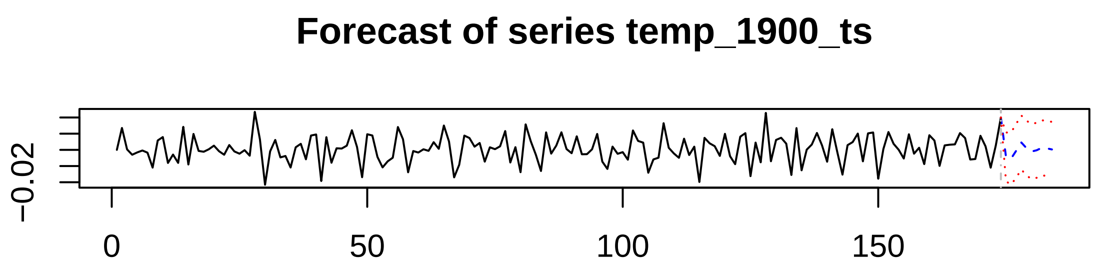
```
BIC(train_mod_var$varresult$carbon_1900_ts,train_mod_var$varresult$temp_1900_ts)
```

```
##                                       df       BIC
## train_mod_var$varresult$carbon_1900_ts  8  836.2316
## train_mod_var$varresult$temp_1900_ts    8 -755.2975
```

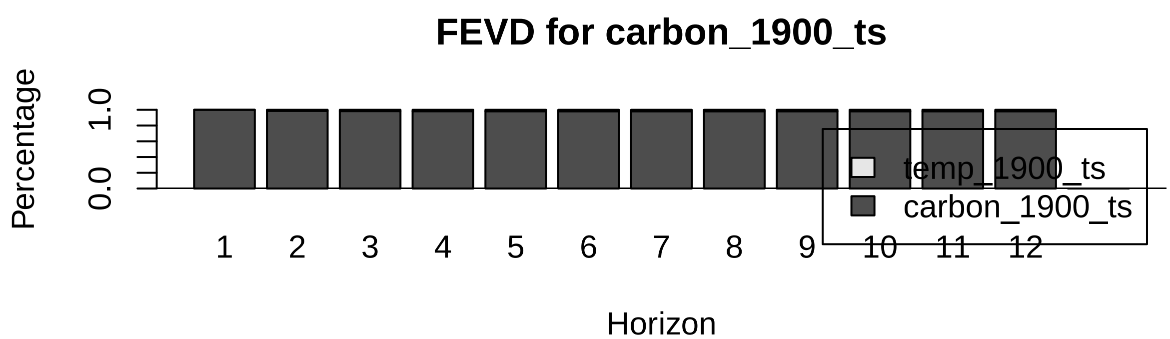
Conversely, the BIC for temperature is far smaller than the BIC for carbon emissions. This generally would mean the temperature model is far superior, however in this case, it is most likely due to the lack of standardization between the variables.

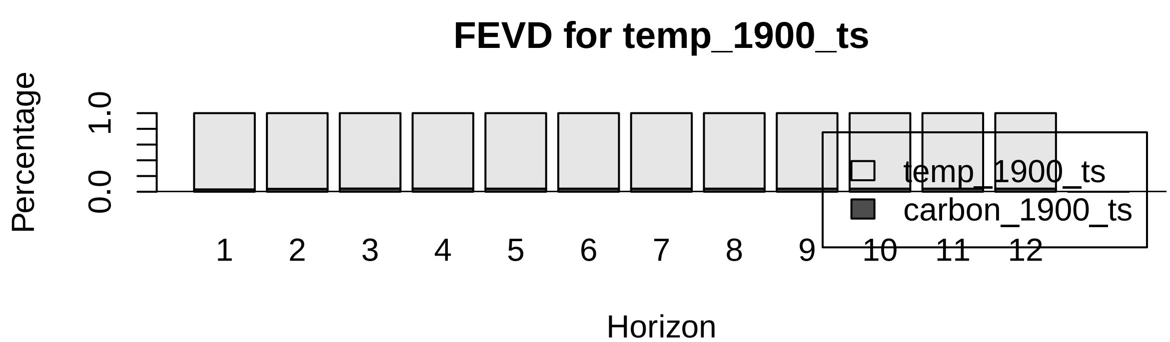
```
var_predict <- predict(object=var_mod, n.ahead=10)
plot(var_predict)
```





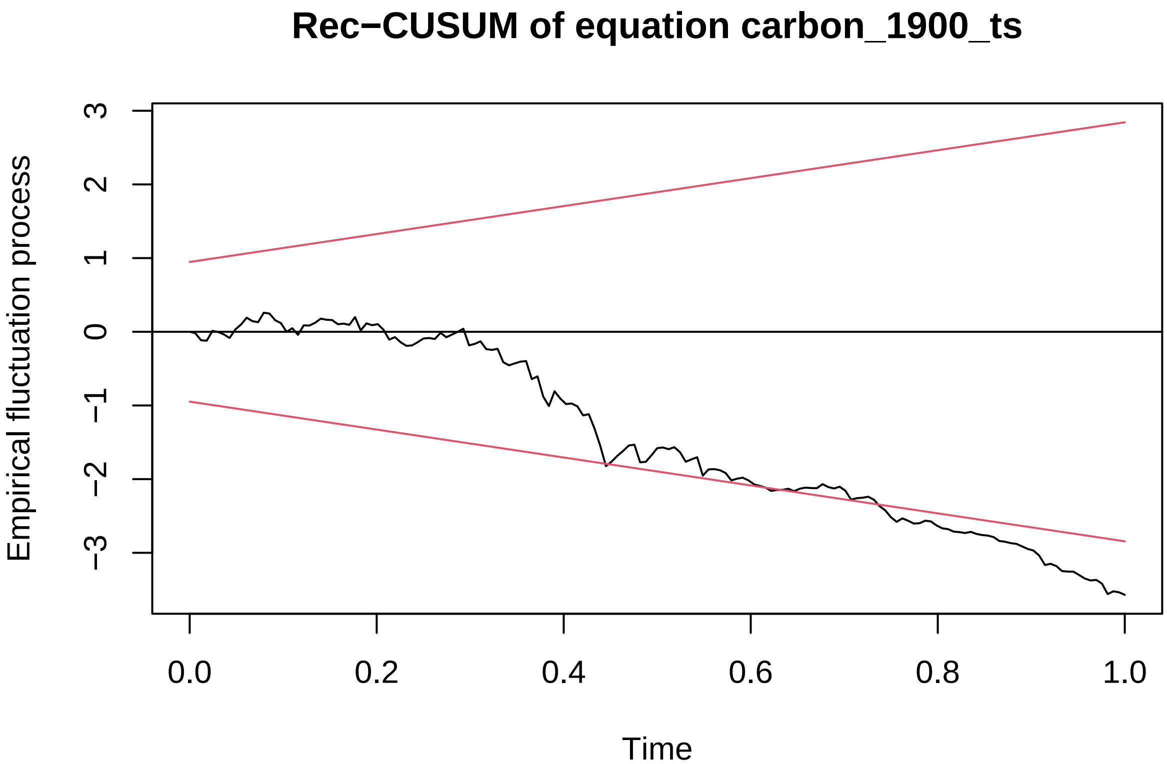

```
plot(fevd(var_mod, n.ahead = 12))
```

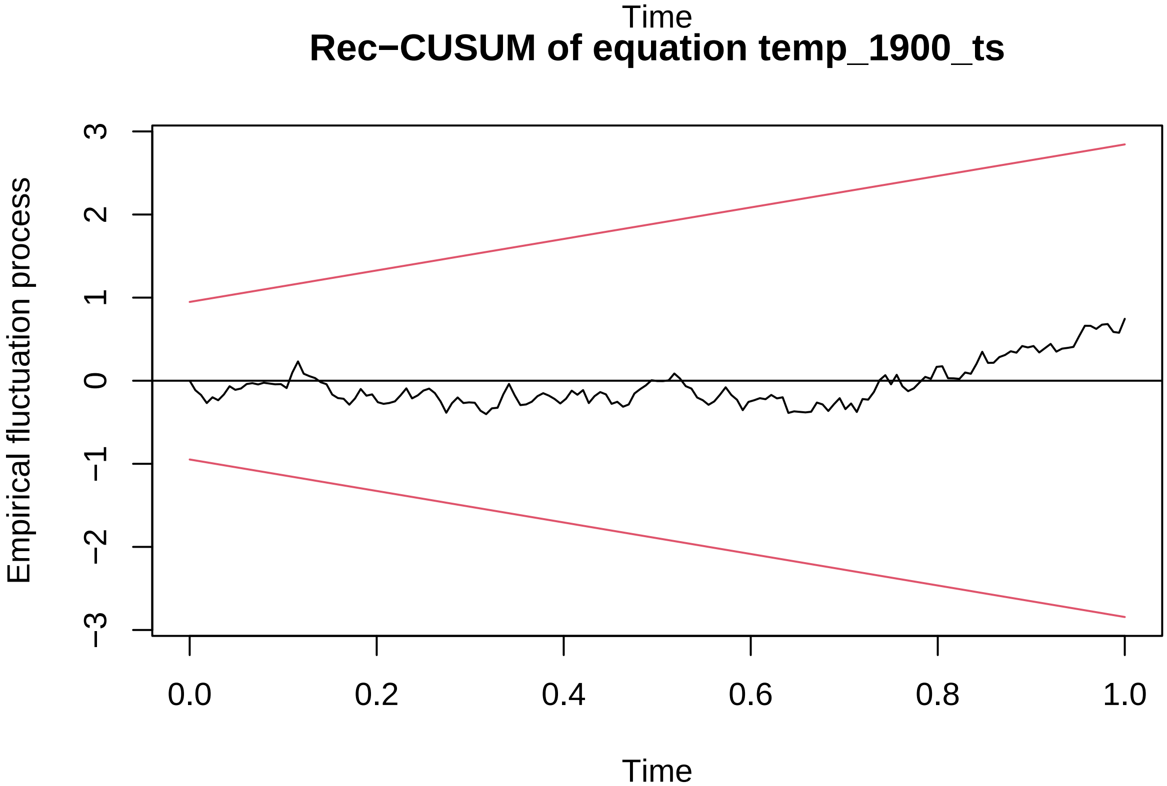




Most of the forecast error variance is explained by the variables themselves rather than the other. This suggests strong auto-correlation for both variables.

```
plot(stability(var_mod, type = "Rec-CUSUM"), plot.type="single")
```





For the rec-cusum graph for carbon emissions, the line crosses the confidence bounds indicating the model has instability.

The rec-cusum graph for temperature does not cross the bounds so we conclude there is no structural instability.

(6) Conclusion

Starting with the AR models, we found that the Autoregressive models for our response variables (carbon emission and global average temperature) were fairly good models. We found that carbon emissions can best be modeled by an AR(3) model while temperature can best be modeled by an AR(1) model. These models did not show signs of serial correlation and outperformed competing AR(n) models. With this being said, the mean normalized RMSEs for both models were close to 4. Since both our variables are measures of growth with averages close to zero it indicates the models may not be accurate predictors.

For modeling carbon emissions, the ARDL(2,3) with energy consumption as the other predictor was the better ARDL model. For temperature, the ARDL(2,3) with oil production as the other predictor was the better ARDL model. These models, however, had more evidence of serial correlation and/or were less favorable when looking at RMSE and BIC.

The VAR model showed little to no sign of usefuleness due to the lack of meaningful correlation between the two response variables. Despite intuition which may suggest that carbon emissions and temperature are linked, we found they had very little influence on one another.

From this, we conclude that the best model for each variable would be the Autoregressive models. They provided the most accurate results and passed the diagnostics necessary to provide robust estimates and predictions.

Part 2: Panel Data

(1) Introduction

We are using the (WHO) Life Expectancy dataset, to answer the economic question of: How do economic factors impact life expectancy across different regions or countries? Life expectancy is a key indicator for a country's population's overall health and well-being. Understanding how economic factors influence health outcomes can help in assessing the long-term sustainability of a country's development and the effectiveness of its policies. This can ultimately aid us to help individuals live a more satisfactory lifestyle for the longest time possible.

(2) EDA

```r
p_GDP <- powerTransform(GDP ~ 1, data = Life_Expectancy_Data, family = "bcPower")
summary(p_GDP)
```

```
## bcPower Transformation to Normality
##    Est Power Rounded Pwr Wald Lwr Bnd Wald Upr Bnd
## Y1    0.0325        0.03        0.014       0.0511
##
## Likelihood ratio test that transformation parameter is equal to 0
##  (log transformation)
##                             LRT df       pval
## LR test, lambda = (0) 11.89692  1 0.00056227
##
## Likelihood ratio test that no transformation is needed
##                             LRT df       pval
## LR test, lambda = (1) 7326.282  1 < 2.22e-16
```

```r
p_Te <- powerTransform(Total.expenditure ~ 1, data = Life_Expectancy_Data, family = "bcPower")
summary(p_Te)
```

```
## bcPower Transformation to Normality
##    Est Power Rounded Pwr Wald Lwr Bnd Wald Upr Bnd
## Y1    0.6009         0.6       0.5373       0.6644
##
## Likelihood ratio test that transformation parameter is equal to 0
##  (log transformation)
##                            LRT df       pval
## LR test, lambda = (0) 369.573  1 < 2.22e-16
##
## Likelihood ratio test that no transformation is needed
##                             LRT df       pval
## LR test, lambda = (1) 144.6698  1 < 2.22e-16
```

```
## bcPower Transformation to Normality
##    Est Power Rounded Pwr Wald Lwr Bnd Wald Upr Bnd
## Y1    2.5296        2.53       2.4135       2.6458
##
## Likelihood ratio test that transformation parameter is equal to 0
##  (log transformation)
##                             LRT df       pval
## LR test, lambda = (0) 4114.212  1 < 2.22e-16
##
## Likelihood ratio test that no transformation is needed
##                            LRT df       pval
```

```
## LR test, lambda = (1) 1057.59  1 < 2.22e-16
```

Transformations

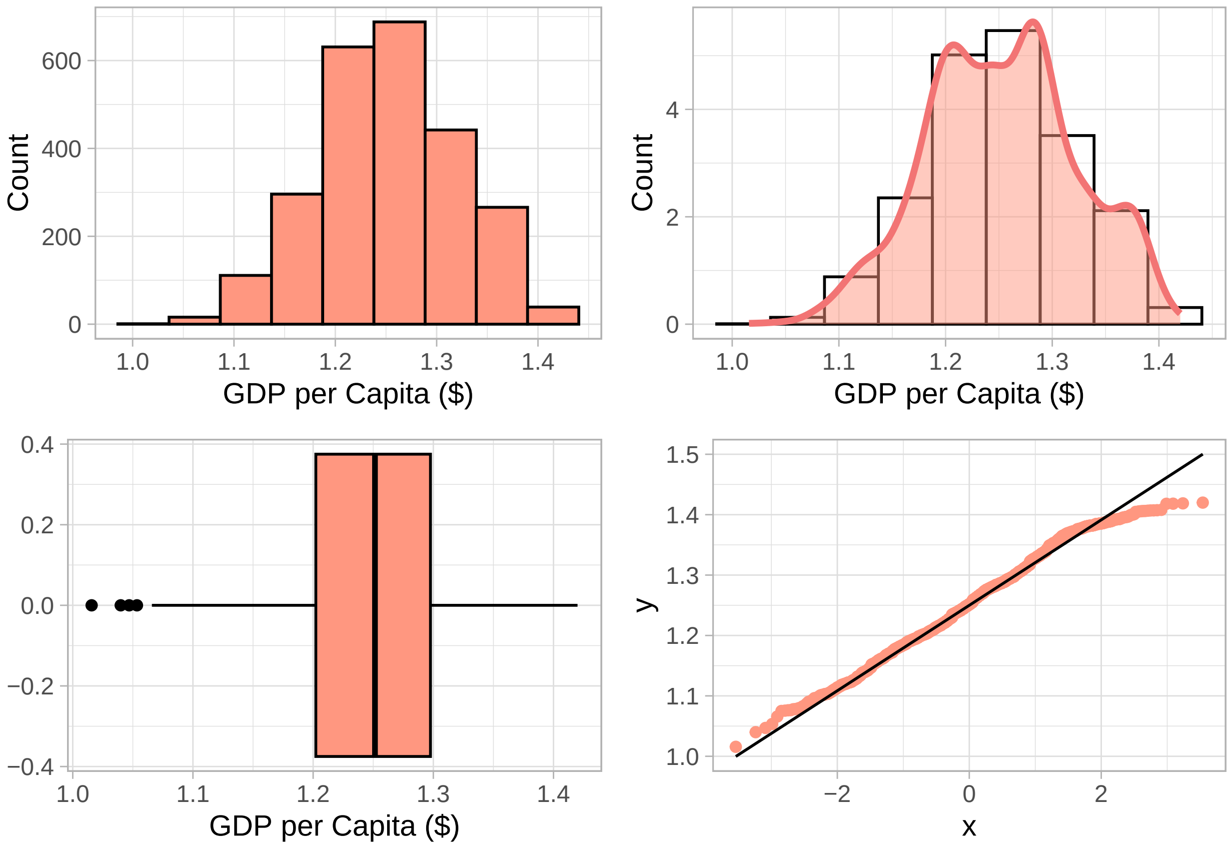
```r
Life_Expectancy_Data$GDPT <- (Life_Expectancy_Data$GDP)^0.03 # log transformation
Life_Expectancy_Data$Total.expenditureT <- (Life_Expectancy_Data$Total.expenditure)^0.6

summary(Life_Expectancy_Data, c("Life.expectancy", "GDPT", "Total.expenditureT", "Schooling"))
```

```
##    Country               Year         Status          Life.expectancy
##  Length:2938        Min.   :2000   Length:2938        Min.   :36.30  
##  Class :character   1st Qu.:2004   Class :character   1st Qu.:63.10  
##  Mode  :character   Median :2008   Mode  :character   Median :72.10  
##                     Mean   :2008                      Mean   :69.22  
##                     3rd Qu.:2012                      3rd Qu.:75.70  
##                     Max.   :2015                      Max.   :89.00  
##                                                       NA's   :10     
##  Adult.Mortality infant.deaths       Alcohol       percentage.expenditure
##  Min.   :  1.0   Min.   :   0.0   Min.   : 0.0100   Min.   :    0.000    
##  1st Qu.: 74.0   1st Qu.:   0.0   1st Qu.: 0.8775   1st Qu.:    4.685    
##  Median :144.0   Median :   3.0   Median : 3.7550   Median :   64.913    
##  Mean   :164.8   Mean   :  30.3   Mean   : 4.6029   Mean   :  738.251    
##  3rd Qu.:228.0   3rd Qu.:  22.0   3rd Qu.: 7.7025   3rd Qu.:  441.534    
##  Max.   :723.0   Max.   :1800.0   Max.   :17.8700   Max.   :19479.912    
##  NA's   :10                       NA's   :194                            
##   Hepatitis.B       Measles              BMI        under.five.deaths
##  Min.   : 1.00   Min.   :     0.0   Min.   : 1.00   Min.   :   0.00  
##  1st Qu.:77.00   1st Qu.:     0.0   1st Qu.:19.30   1st Qu.:   0.00  
##  Median :92.00   Median :    17.0   Median :43.50   Median :   4.00  
##  Mean   :80.94   Mean   :  2419.6   Mean   :38.32   Mean   :  42.04  
##  3rd Qu.:97.00   3rd Qu.:   360.2   3rd Qu.:56.20   3rd Qu.:  28.00  
##  Max.   :99.00   Max.   :212183.0   Max.   :87.30   Max.   :2500.00  
##  NA's   :553                        NA's   :34                       
##      Polio       Total.expenditure   Diphtheria       HIV.AIDS     
##  Min.   : 3.00   Min.   : 0.370    Min.   : 2.00   Min.   : 0.100  
##  1st Qu.:78.00   1st Qu.: 4.260    1st Qu.:78.00   1st Qu.: 0.100  
##  Median :93.00   Median : 5.755    Median :93.00   Median : 0.100  
##  Mean   :82.55   Mean   : 5.938    Mean   :82.32   Mean   : 1.742  
##  3rd Qu.:97.00   3rd Qu.: 7.492    3rd Qu.:97.00   3rd Qu.: 0.800  
##  Max.   :99.00   Max.   :17.600    Max.   :99.00   Max.   :50.600  
##  NA's   :19      NA's   :226       NA's   :19                      
##       GDP              Population        thinness..1.19.years
##  Min.   :     1.68   Min.   :3.400e+01   Min.   : 0.10       
##  1st Qu.:   463.94   1st Qu.:1.958e+05   1st Qu.: 1.60       
##  Median :  1766.95   Median :1.387e+06   Median : 3.30       
##  Mean   :  7483.16   Mean   :1.275e+07   Mean   : 4.84       
##  3rd Qu.:  5910.81   3rd Qu.:7.420e+06   3rd Qu.: 7.20       
##  Max.   :119172.74   Max.   :1.294e+09   Max.   :27.70       
##  NA's   :448         NA's   :652         NA's   :34          
##  thinness.5.9.years Income.composition.of.resources   Schooling    
##  Min.   : 0.10      Min.   :0.0000                  Min.   : 0.00  
##  1st Qu.: 1.50      1st Qu.:0.4930                  1st Qu.:10.10  
##  Median : 3.30      Median :0.6770                  Median :12.30  
##  Mean   : 4.87      Mean   :0.6276                  Mean   :11.99  
```

```
##  3rd Qu.: 7.20      3rd Qu.:0.7790                  3rd Qu.:14.30
##  Max.   :28.60      Max.   :0.9480                  Max.   :20.70
##  NA's   :34         NA's   :167                     NA's   :163
##       GDPT        Total.expenditureT
##  Min.   :1.016   Min.   :0.5507
##  1st Qu.:1.202   1st Qu.:2.3859
##  Median :1.252   Median :2.8578
##  Mean   :1.253   Mean   :2.8469
##  3rd Qu.:1.298   3rd Qu.:3.3479
##  Max.   :1.420   Max.   :5.5887
##  NA's   :448     NA's   :226
```

GDP

```r
GDP_hist <- ggplot(Life_Expectancy_Data, aes(x = GDPT)) +
  geom_histogram(color = 'black', fill = '#ff9780', bins = round(1 + log(183, base = 2), digits = 0)) +
  labs(x = "GDP per Capita ($)", y = "Count")

GDP_hist_fitted <- ggplot(Life_Expectancy_Data, aes(x = GDPT)) +
  geom_histogram(aes(y = after_stat(density)), color = 'black', fill = NA, bins = round(1 + log(183, ba
  geom_density(lwd = 1.2, color = '#f27474', fill = '#ff9780', alpha = 0.5) +
  labs(x = "GDP per Capita ($)", y = "Count")

GDP_box <- ggplot(Life_Expectancy_Data, aes(x = GDPT)) +
  geom_boxplot(color = 'black', fill = '#ff9780') +
  labs(x = "GDP per Capita ($)")

GDP_qq <- ggplot(Life_Expectancy_Data, aes(sample = GDPT)) +
  stat_qq(color = '#ff9780') +
  stat_qq_line()

ggarrange(GDP_hist, GDP_hist_fitted, GDP_box, GDP_qq)
```




```
summary(Life_Expectancy_Data$GDPT)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   1.016   1.202   1.252   1.253   1.298   1.420     448
```

Comments: Histogram & Fitted Distribution: These graphs show that the distribution of GDP per capita is slightly left-skewed. The fitted distribution is close to a normal distribution, with a median valued at 1.252.

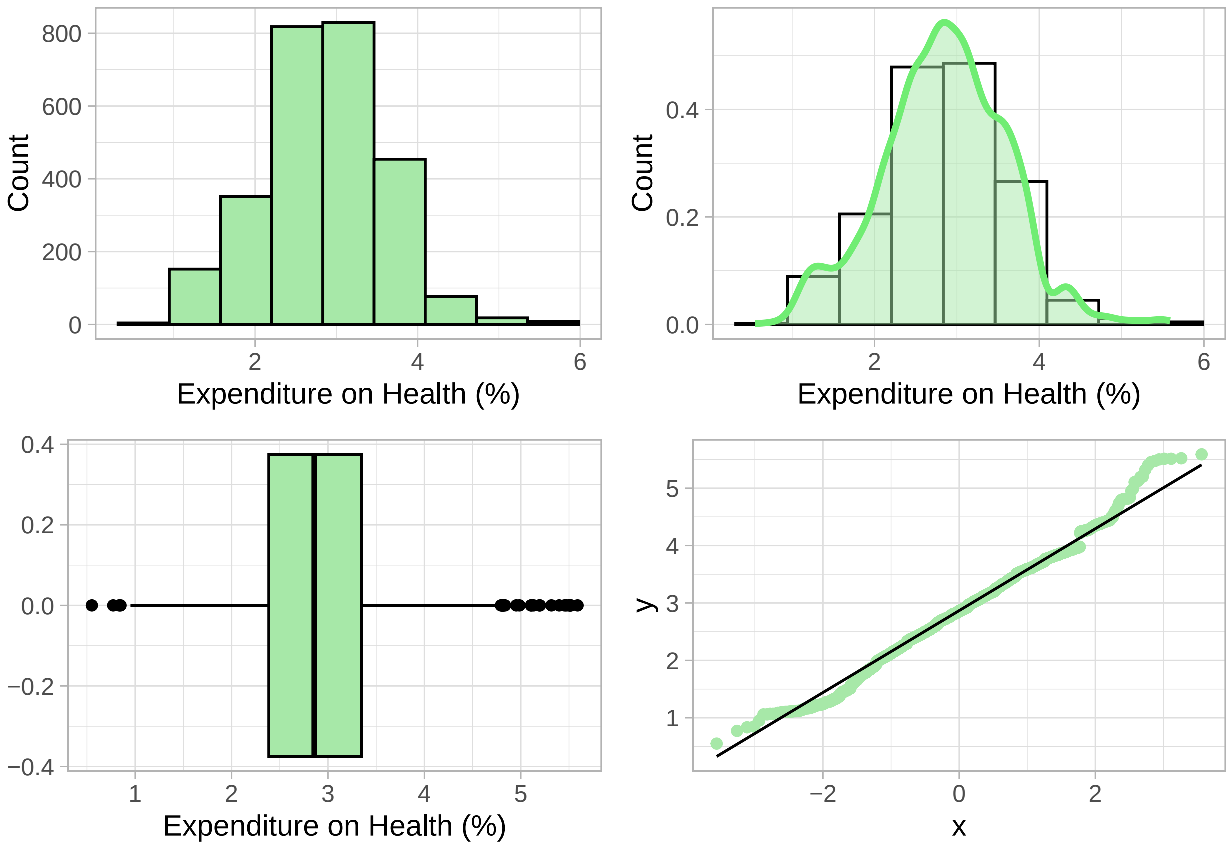
Boxplot: The boxplot re-emphasizes the left-skewness of the distribution. It also shows that there are a few number of outliers — the top 3 countries in terms of GDP per capita, which are the separated from the rest of the outliers, are Luxembourg, Qatar, and Switzerland. However, it is not surprising to see outliers, due to the great number of relatively poor countries and handful of extremely wealthy ones.

Q-Q Plot: The QQ-plot confirms the normality of the distribution, with very few outliers but some deviation at the tail-end.

Total expenditure

```
Te_hist <- ggplot(Life_Expectancy_Data, aes(x = Total.expenditureT)) +
  geom_histogram(color = 'black', fill = '#a7e8a8', bins = round(1 + log(183, base = 2), 0)) +
  labs(x = "Expenditure on Health (%)", y = "Count")

Te_hist_fitted <- ggplot(Life_Expectancy_Data, aes(x = Total.expenditureT)) +
  geom_histogram(aes(y=..density..), color = 'black', fill = NA, bins = round(1 + log(183, base = 2), 0
  geom_density(lwd = 1.2, color = '#70ed73', fill = '#a7e8a8', alpha = 0.5) +
  labs(x = "Expenditure on Health (%)", y = "Count")

Te_box <- ggplot(Life_Expectancy_Data, aes(x = Total.expenditureT)) +
```

```r
  geom_boxplot(color = 'black', fill = '#a7e8a8') +
  labs(x = "Expenditure on Health (%)")

Te_qq <- ggplot(Life_Expectancy_Data, aes(sample = Total.expenditureT)) +
  stat_qq(color = '#a7e8a8') +
  stat_qq_line()

ggarrange(Te_hist, Te_hist_fitted, Te_box, Te_qq)
```




```r
summary(Life_Expectancy_Data$Total.expenditureT)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##  0.5507  2.3859  2.8578  2.8469  3.3479  5.5887     226
```

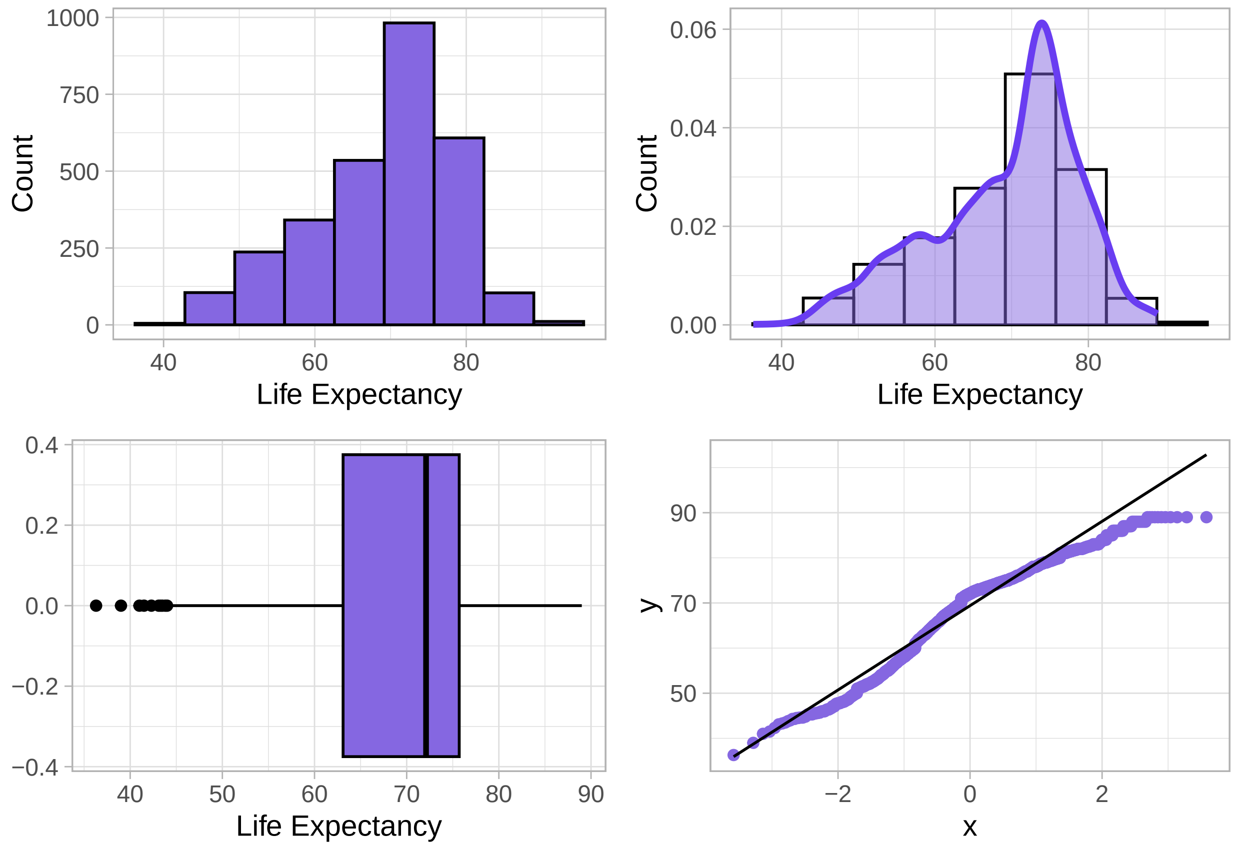
Comments: Histogram & Boxplot: From the histogram and boxplot graphs we see a distribution which is slightly right-skewed, indicating that more countries spend a lower percentage of their government expenditure on health. We see that most countries with lower spending on health, tend to focus funds on other sectors such as defense, education, and infrastructure needs.

Q-Q plot: The Q-Q plot reveals some slight deviation from normality.

Life Expectancy

```r
Le_hist <- ggplot(Life_Expectancy_Data, aes(x = Life.expectancy)) +
geom_histogram(color = 'black', fill = '#8567e1', bins = round(1 + log(183, base = 2), 0)) +
labs(x = "Life Expectancy", y = "Count")
Le_hist_fitted <- ggplot(Life_Expectancy_Data, aes(x = Life.expectancy)) +
geom_histogram(aes(y=..density..), color = 'black', fill = NA, bins = round(1 + log(183, base = 2), 0))
```

```
geom_density(lwd = 1.2, color = '#693df1', fill = '#8567e1', alpha = 0.5) +
labs(x = "Life Expectancy", y = "Count")
Le_box <- ggplot(Life_Expectancy_Data, aes(x = Life.expectancy)) +
geom_boxplot(color = 'black', fill = '#8567e1') +
labs(x = "Life Expectancy")
Le_qq <- ggplot(Life_Expectancy_Data, aes(sample = Life.expectancy)) +
stat_qq(color = '#8567e1') +
stat_qq_line()
ggarrange(Le_hist, Le_hist_fitted, Le_box, Le_qq)
```




```
summary(Life_Expectancy_Data$Life.expectancy)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   36.30   63.10   72.10   69.22   75.70   89.00      10
```

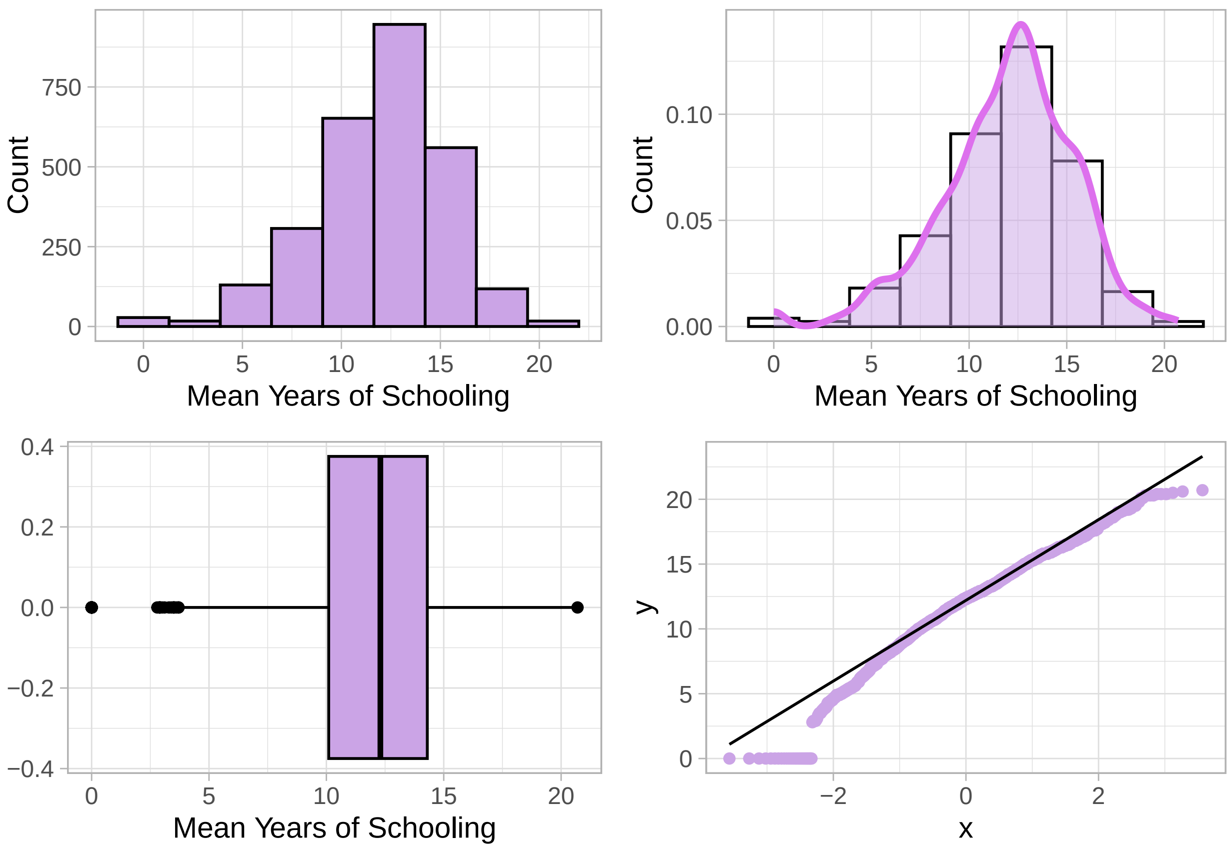
Comments: Histogram & Fitted Distribution: The graphs are left-skewed with the majority of countries lying between the life expectancies of 70 and 75. There are also a significant number of countries that are below the mean of 69.22.

Boxplot: Again, the boxplot shows that the data is slightly left-skewed. There are a few outliers, but the most significant one is Haiti with a life expectancy of 36.30 in 2010; this can be explained by the Cholera Outbreak along with poor sanitation in 2010.

Q-Q Plot: The Q-Q plot shows that the data almost follows a normal distribution. While the other visuals suggest a transformation may be necessary, this graph does not.

Schooling

```
school_hist <- ggplot(Life_Expectancy_Data, aes(x = Schooling)) +
geom_histogram(color = 'black', fill = '#cba4e6', bins = round(1 + log(183, base = 2), 0)) +
labs(x = "Mean Years of Schooling", y = "Count")
school_hist_fitted <- ggplot(Life_Expectancy_Data, aes(x = Schooling)) +
geom_histogram(aes(y=..density..), color = 'black', fill = NA, bins = round(1 + log(183, base = 2), 0))
geom_density(lwd = 1.2, color = '#dd70ed', fill = '#cba4e6', alpha = 0.5) +
labs(x = "Mean Years of Schooling", y = "Count")
school_box <- ggplot(Life_Expectancy_Data, aes(x = Schooling)) +
geom_boxplot(color = 'black', fill = '#cba4e6') +
labs(x = "Mean Years of Schooling")
school_qq <- ggplot(Life_Expectancy_Data, aes(sample = Schooling)) +
stat_qq(color = '#cba4e6') +
stat_qq_line()
ggarrange(school_hist, school_hist_fitted, school_box, school_qq)
```




```
summary(Life_Expectancy_Data$Schooling)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00   10.10   12.30   11.99   14.30   20.70     163
```

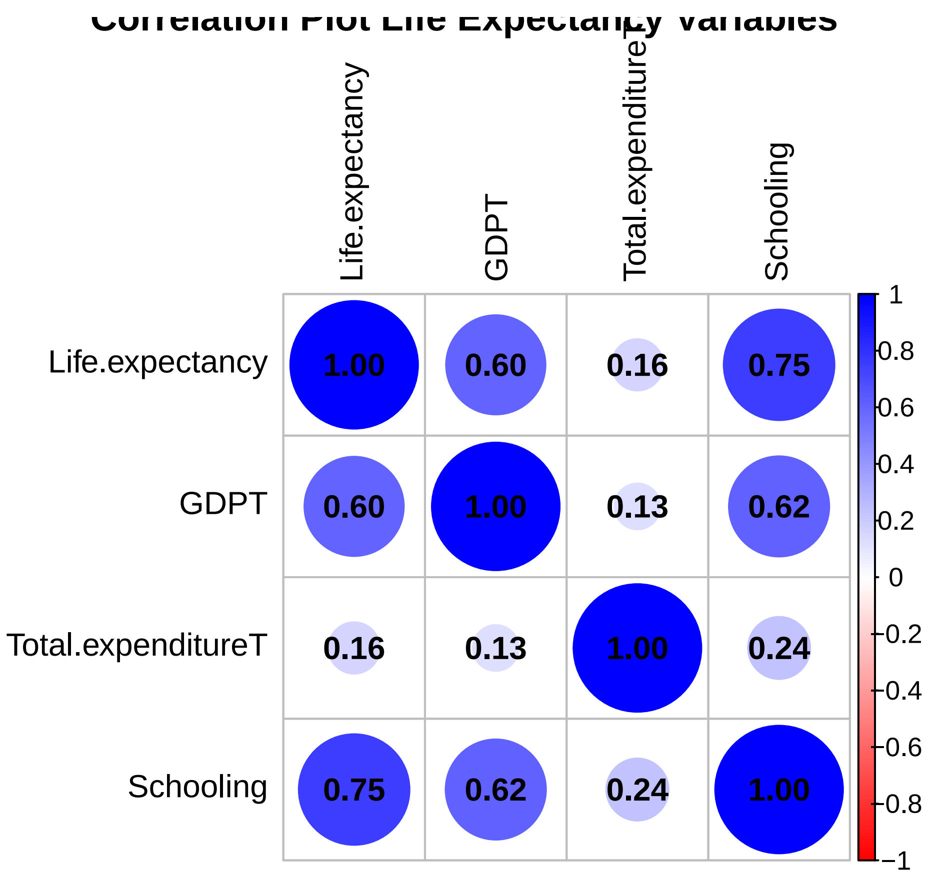
Comments: Histogram & Fitted Distribution: This graph's distribution appears slightly left-skewed; indicating that a majority of countries have a moderate-to-high average number of schooling years. The distribution means that most countries are located around the mean (11.99).

Boxplot: The boxplot re-inforces the normality of the distribution; the median sits at 12.30 which is approximately in the middle of the IQR. The boxplot also shows us that 50% of countries have a level of schooling between 10.1 and 14.3 years. The whiskers are also almost equal in length; however, the left whisker is slightly lengthier, signifying that there are more countries in the lower quartile of schooling.

Q-Q Plot: The QQ-plot confirms the normality of the distribution, with very few outliers around the lower and higher extremes.

Corrplot

```
corr_matrix <- cor(Life_Expectancy_Data[,c("Life.expectancy", "GDPT", "Total.expenditureT", "Schooling")
corrplot(corr_matrix, method = "circle", type = "full", col = colorRampPalette(c("red", "white", "blue")
         addCoef.col = "black", tl.col = "black", title = "Correlation Plot Life Expectancy Variables")
```



The Correlation Plot shows that Life Expectancy is highly correlated with the other variables, along with schooling and GDPT also showing significant correlations with them. However, Total Expenditure appears to have weak correlations with any of the other variables.

(3) Models

```
library(dplyr)

Life_Expectancy_Data <- read.csv('Life Expectancy Data.csv')

LED <- c("Afghanistan", "Albania", "Algeria", "Bosnia and Herzegovina",
                        "Brazil", "Bulgaria", "Chile", "China", "Colombia", "El Salvador")

filtered_data <- Life_Expectancy_Data %>%
  filter(Country %in% LED)
```

```r
final_filtered_data <- filtered_data %>%
  filter(Year >= 2000 & Year <= 2014)
```

Panel conversion

```r
LED <- read.csv('Life Expectancy Data.csv')
library(plm)

LED$GDPT <- (LED$GDP)^0.03
LED$Total.expenditureT <- (LED$Total.expenditure)^0.6
LE.pd <- pdata.frame(LED, index = c("Country", "Year"))
```

Pooled model:

```r
mreg.pooled <-lm(Life.expectancy ~ GDPT + Total.expenditureT +
Schooling, data = LE.pd)
```

Fixed Effects:

```r
mreg.fixed <- plm(Life.expectancy ~ GDPT + Total.expenditureT +
Schooling, data = LE.pd, model = "within")
```

Random Effects:

```r
mreg.random <- plm(Life.expectancy ~ GDPT + Total.expenditureT +
Schooling, data = LE.pd, model = "random")
```

```r
phtest(mreg.fixed, mreg.random)
```

Test which model is better

```
##
##  Hausman Test
##
## data:  Life.expectancy ~ GDPT + Total.expenditureT + Schooling
## chisq = 289.73, df = 3, p-value < 2.2e-16
## alternative hypothesis: one model is inconsistent
```

HO: Random Effects model is a better fit than the Fixed Effects Model. P-value is less than 0.05, so we reject the null and choose the Fixed Effects model.

```r
f_test <- pFtest(mreg.fixed, mreg.pooled)
print(f_test)
```

F-test to compare Pooled vs Fixed Effects Model

```
##
##  F test for individual effects
##
## data:  Life.expectancy ~ GDPT + Total.expenditureT + Schooling
## F = 85.327, df1 = 156, df2 = 2167, p-value < 2.2e-16
```

```
## alternative hypothesis: significant effects
```

HO: Pooled model is a better fit than the Fixed model. The p-value is significant and less than 0.05, so we reject the null and conclude the Fixed Effects model is preferred.

Conclusion: Based on the results from the statistical tests and effect plots conducted, the Fixed Effects model is preferred over the other models. The p-value from the Hausman test is less than 0.05, leading us to reject the null hypothesis. This indicates that the Fixed Effects model provides a better fit compared to the Random Effects model.

Additionally, the F-test comparing the Pooled model with the Fixed Effects model also yielded a p-value less than 0.05, further supporting the choice of the Fixed Effects model over the Pooled model. Therefore, we conclude that the Fixed Effects model is the most appropriate model for analyzing life expectancy based on GDP, total expenditure, and schooling.