Forecasting Change in Annual Global Temperature

Lucas Childs

2025-05-25

Abstract

This time series analysis report sought to use Box-Jenkins methodology to forecast annual global temperature deviations compared to the long-term average temperature from 1951-1980. Using data from 1880 to 2024, I fit an ARIMA(0, 1, 2) model following Box-Jenkins methodology. Originally, after differencing the data once to eliminate trend, I considered 3 potential models based on sample ACF and PACF and AICc values. These models were: ARIMA(1, 1, 3), ARIMA(3, 1, 0), and ARIMA(0, 1, 2). The final decision was ultimately made based upon the principle of parsimony, since all models passed diagnostics tests on the models' residuals. Box-Jenkins methods were verified by checking that the data followed linearity assumptions, were stationary before conducting model building, and the residuals of the models followed Gaussian white noise. Based on the forecasts by the ARIMA(0, 1, 2) model, the global change in temperature was roughly predicted not to change from 2015 to 2024, however, the test set shows that this is not the case and that my model underestimated temperatures. The test set mostly falls within the 95% forecast interval, but the last 2 ground truth values lie above it.

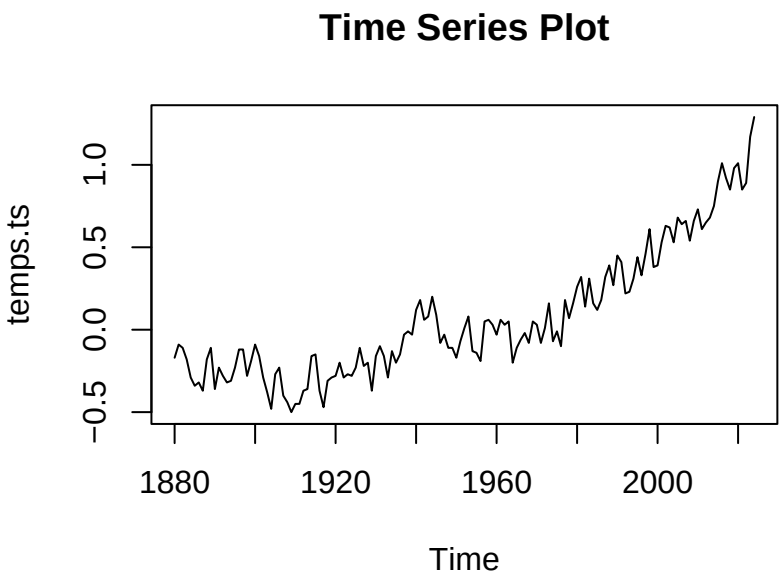
Introduction

Based on NASA's Goddard Institute for Space Sciences (GISS) Global Land-Ocean Temperature Index, I sought to forecast yearly changes in global surface temperature using training data from 1880 until 2014 and testing with data from 2015 to 2024 [1]. The dataset provided by NASA's GISS took a long-term average temperature from 1951 until 1980 and recorded deviations of global surface temperature (in Celsius) from this, ranging from 1880 until 2024. These data are important and relevant due to the continually rising global temperatures we have experienced due to global warming. The past 10 years have been the warmest years on record and Earth was about 1.47^{o}C warmer in 2024 compared to the preindustrial age [1]. The increasingly warm temperatures are alarming as rising global temperatures run the risk of inducing species extinction due to habitat loss, migration limitations, and swift climate shifts [3].

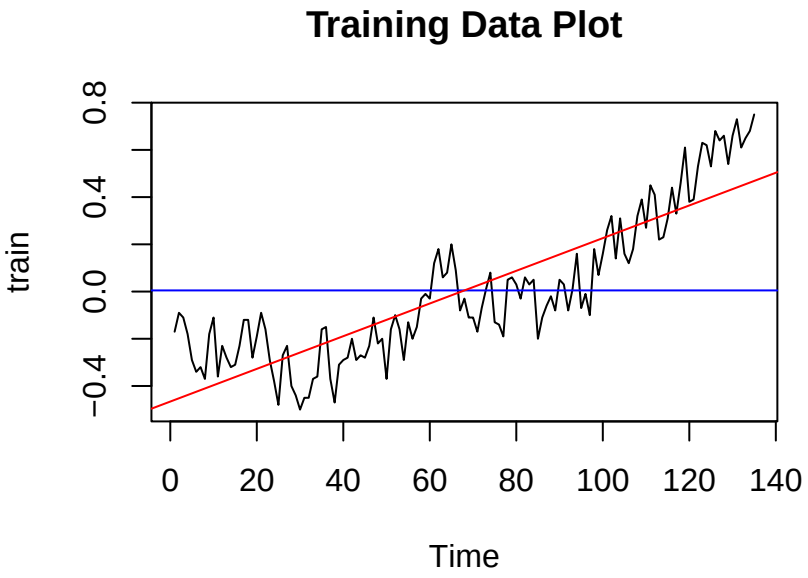
To properly forecast using an autoregressive integrated moving average (ARIMA) model, I followed Box-Jenkins methodology and utilized RStudio [2]. In my analysis I differenced the time series to remove an increasing first-order trend and applied no transformation as variance appeared constant. Once my data appeared stationary I analyzed sample ACF and PACF to hypothesize autoregressive (AR) and moving average (MA) parameters. Then I used Akaike Information Criterion corrected for bias (AICc) to compare the fit of a range of potential models. Choosing three models with the lowest AICc, I made sure they were both stationary and invertible by checking whether the AR and MA roots lied outside of the unit circle to check for stationarity and invertibility respectively. All three models were confirmed to be stationary and invertible. To further check model assumptions I ran diagnostics on the models' residuals to confirm that they resembled Gaussian white noise using a QQ Plot, the Shapiro-Wilk test, Yule-Walker estimation, and Portmanteau tests, to name a few. All models passed diagnostics and the final model, ARIMA(0, 1, 2), was chosen based on the principle of parsimony. I forecasted 10-steps ahead in which my model predicted that there would be roughly no change in global temperature from 2015–2024. However, my test set showed that temperatures during this period fluctuated and ultimately rose, exceeding the model's forecast and indicating a stronger warming trend than anticipated. All in all, my model underestimated forecasts.

Modeling and Forecasting

NASA's Global Temperature Deviations from 1880-2024:

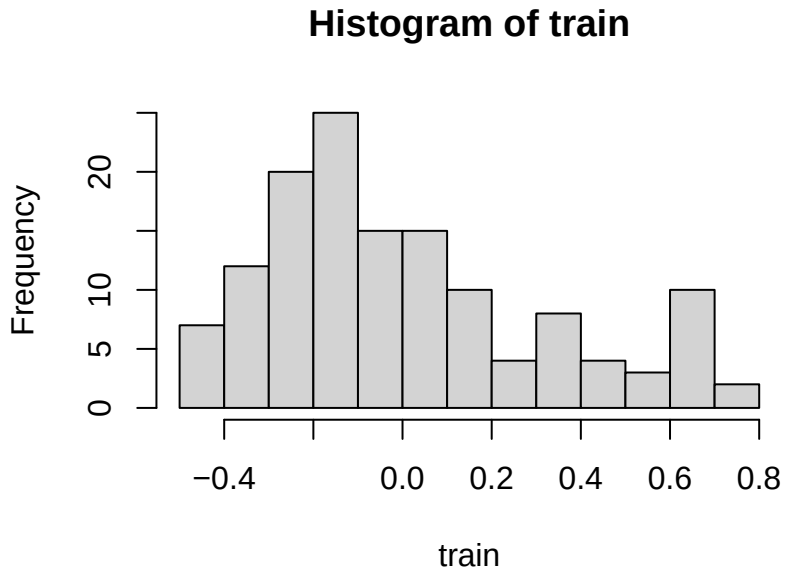


Above is a plot of the dataset which clearly does not portray stationary due to its increasing trend, however it does not appear to have seasonality. Now, I will split the data into a training and testing set, saving the last 10 observations for testing. The resulting training set is plotted below.

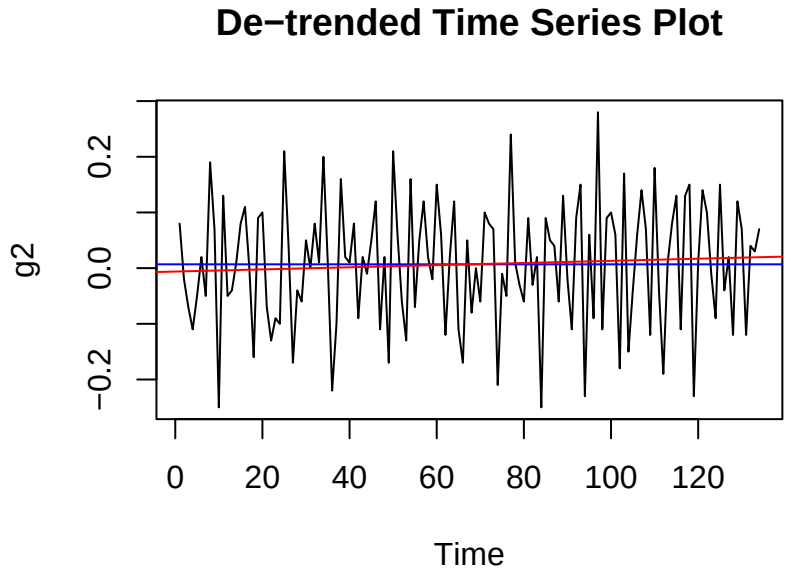


The dataset looks similar to those exmained in class; Box-Jenkins methodology should be appropriate. There appears to be a slight anomaly towards the middle of the time series in which temperature rises and then falls

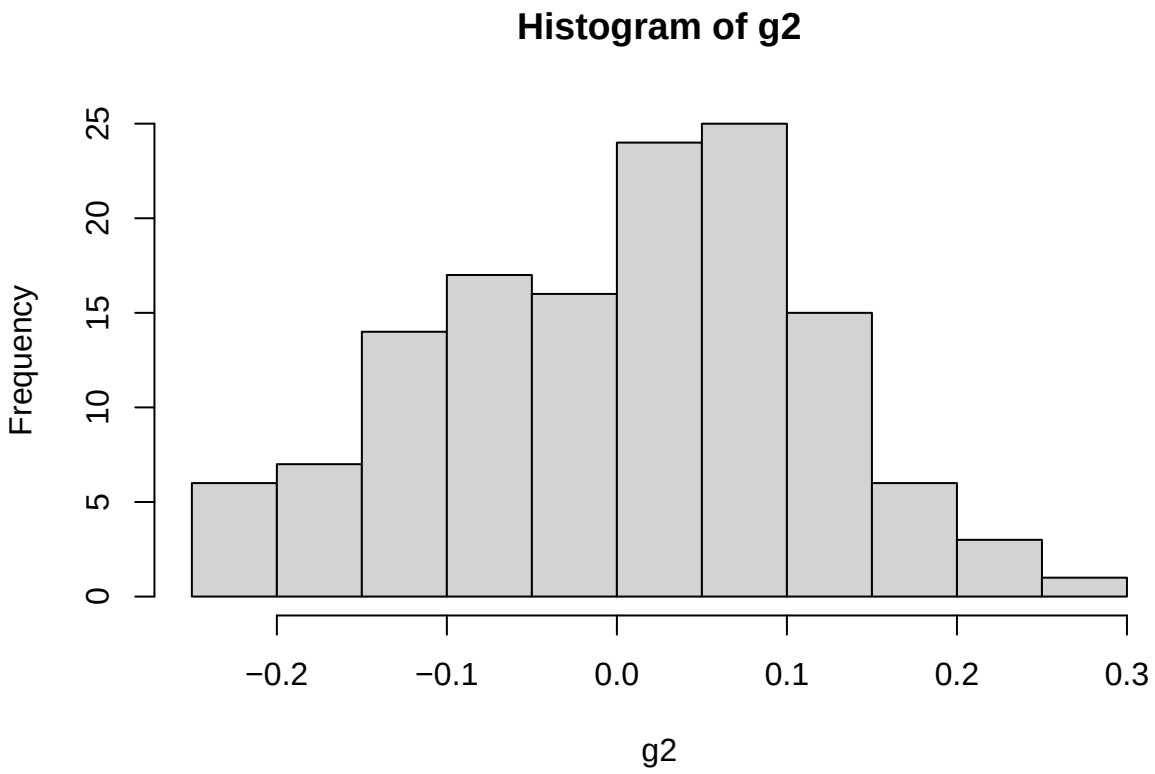
again over the span of about 15 years. Variance looks relatively stable, a constant mean can be seen with the blue line, and an upwards linear trend can be observed with the red. The histogram shown below looks skewed as well. Because of the linear trend, I will difference once at lag 1 to see if this trend is eliminated and sample variance decreases.



After differencing once at lag 1 to eliminate first order-trend, sample variance decreases from 0.09832 to 0.01248, which supports this differencing. Furthermore, the plot below of the differenced time series looks much more stationary with no trend and constant mean. Differencing for a second time at lag 1 increases the variance from 0.01248 to 0.0309, so I will keep differncing to just once.

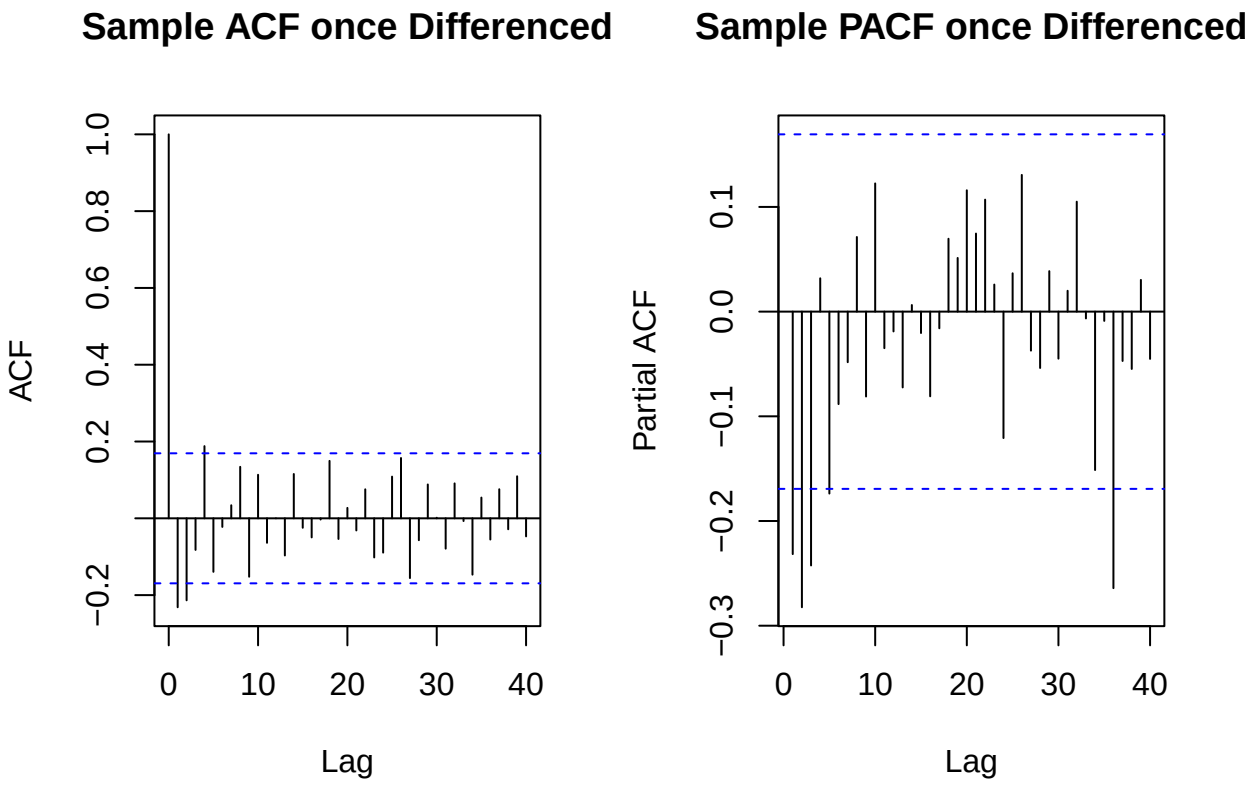


Looking at the histogram below, the data appears to be much more symmetric.



Thus, it can be concluded that the data is now stationary.

Now, we can analyze sample autocorrelation function (ACF) and sample partial autocorrelation function (PACF) to conduct preliminary model identification. Both plots are show on the page below.



Most ACF values that fall outside of the confidence interval are fairly close to the band. However the most significant values are at lags 1 and 2, with borderline significance at lag 4. It is also worth noting that the ACF values outside the CI band are barely outside of it. The scale is very small, so even the largest ACF value at lag 1 is less than 0.1 outside of the interval.

Sample PACF values fall outside the confidence interval at lags 1, 2, 3, and 36. The PACF value at lag 36 is concerning because it is a significant value at a large lag, however due to the small scale again and the fact that R uses approximations of Bartlett's formula to display the confidence intervals in the `acf()` and `pacf()` functions, its practical significance is questionable.

Thus, potential models are ARIMA with $d = 1$, $p = 1, 2, 3$, and $q = 1, 2$, or 4.

After running a loop testing the different AICc values (with `MuMIn`'s `AICc()` function) of models using maximum likelihood estimation for $p = 0, 1, 2, 3$ and $q = 0, 1, 2$, I selected three potential models that gave the lowest AICc values.

The smallest AICc value was -224.92 for ARIMA(1, 1, 3), the second smallest AICc = -224.37 for ARIMA(3, 1, 0), and the third smallest AICc = -224.33 for ARIMA(0, 1, 2). Thus, I have identified 3 potential models to fit to the data. I additionally fitted models using maximum likelihood estimate with $p = 36$ and $q = 0, 1, 2, 3, 4$ and each model had much higher AICc values, so they will not be considered.

The three fitted models are shown below:

Table 1: ARIMA(1, 1, 3) ML Estimated Parameters

	Estimate	Std.Error
ar1	-0.9396	0.0580
ma1	0.6329	0.0994
ma2	-0.5213	0.0811
ma3	-0.2909	0.0786

Table 2: ARIMA(3, 1, 0) ML Estimated Parameters

	Estimate	Std.Error
ar1	-0.3555	0.0841
ar2	-0.3427	0.0841
ar3	-0.2304	0.0835

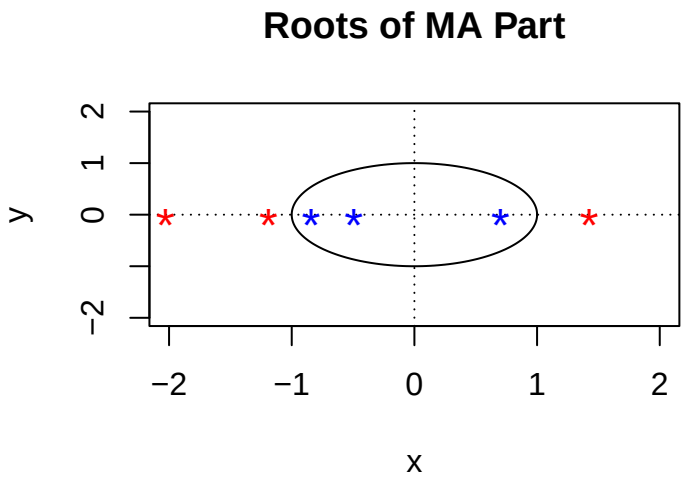
Table 3: ARIMA(0, 1, 2) ML Estimated Parameters

	Estimate	Std.Error
ma1	-0.3698	0.0812
ma2	-0.2038	0.0742

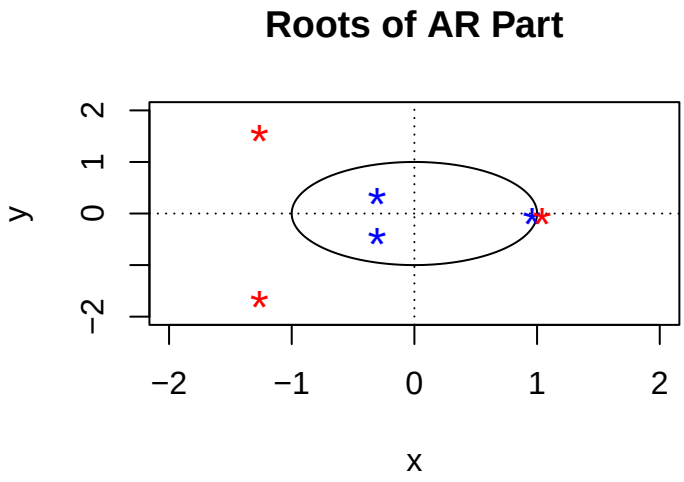
Every model has statistically significant coefficient estimates. If we build confidence intervals for them: $[\hat{\theta} - 2 * s.e. \ , \ \hat{\theta} + 2 * s.e.]$ and test the hypothesis $H_0 : \theta = 0$, each estimate does not lie within the interval (using $\hat{\theta}$ as an example coefficient).

Now let's check stationarity and invertibility of ARIMA(1, 1, 3), ARIMA(3, 1, 0), and ARIMA(0, 1, 2):

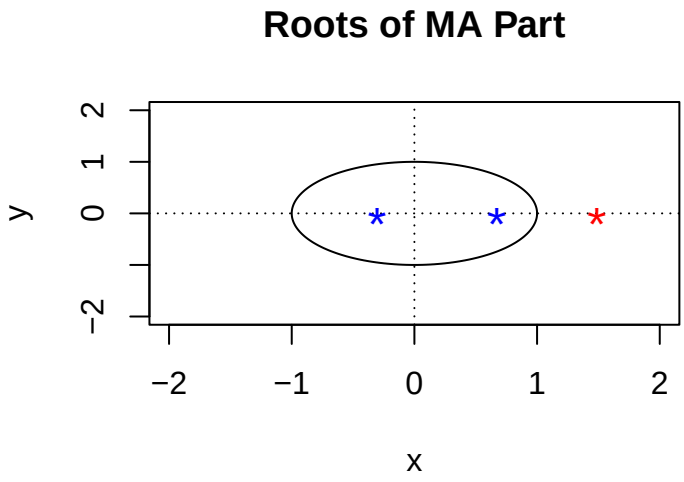
ARIMA(1, 1, 3) is stationary since $|\hat{\phi}_1| = 0.94 < 1$. Additionally, the model is invertible since all roots lie outside the unit circle. This can be visualized like so, where the red points are the roots and blue their inverses:



ARIMA(3, 1, 0) is invertible since it only consists of autoregressive parts. Additionally, ARIMA(3, 1, 0) is stationary because all AR roots lie outside of the unit circle. We can visualize this like so:



ARIMA(0, 1, 2) is stationary since it only consists of MA terms. Additionally, ARIMA(0, 1, 2) is invertible because all roots lie outside of the unit circle. This can be visualized like so (the second root is out of frame at about -3.3):

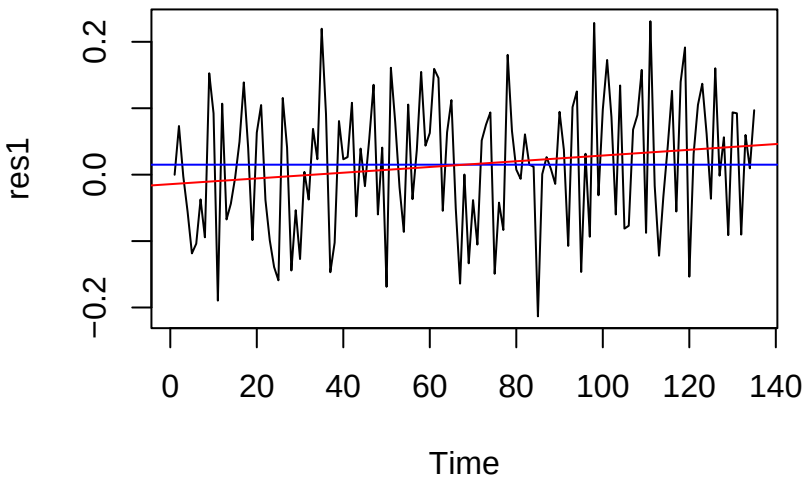


Diagnostic Checking:

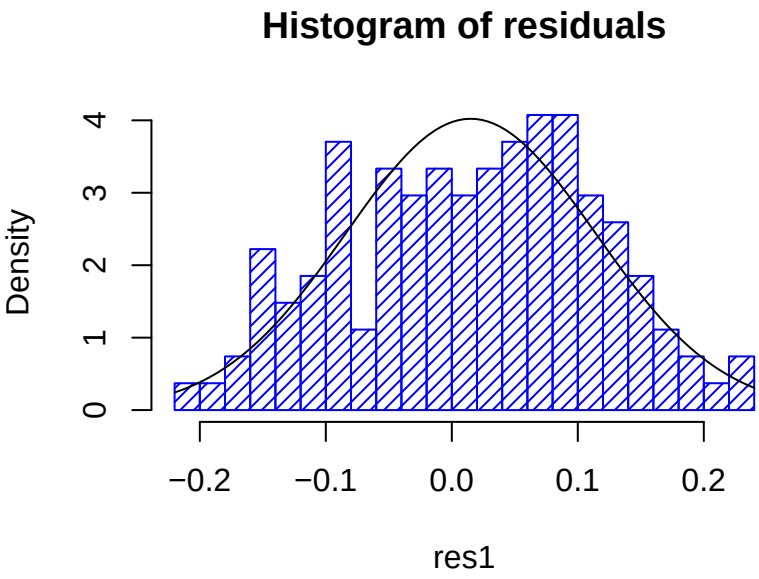
Let's check diagnostics of all three models. If all models pass, we'll choose the model with the lowest number of coefficients by the principle of parsimony.

ARIMA(1, 1, 3) diagnostics:

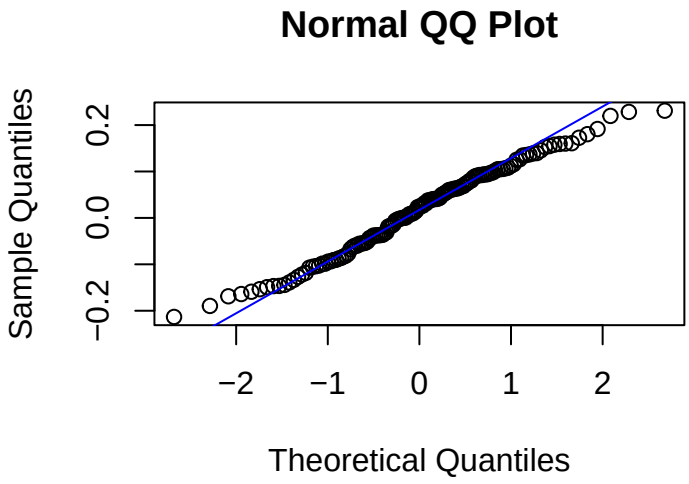
Below is a plot of the residuals of ARIMA(1, 1, 3), which can be seen to roughly follow white noise with no trend and mean: 0.015, variance: 0.00984. If differenced again, the variance of the residuals increases to 0.02066, so we know there is no trend to the residuals.



The histogram of the residuals below looks roughly normal and symmetric.

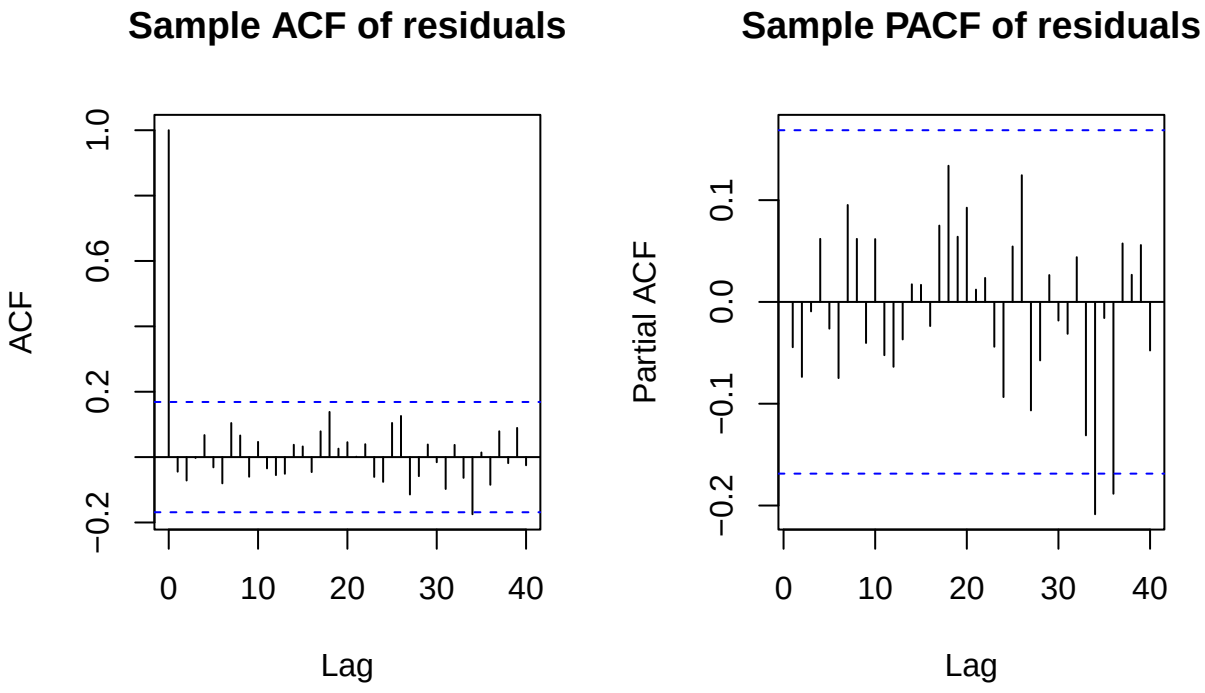


Additionally, it can be seen that the residuals follow the the theoretical line of normality in the QQ Plot, only deviating from it slighly at each tail.



The residuals also pass the Shapiro-Wilk test for normality with an insignificant p value of 0.197.

Next, I will check the sample ACF and PACF of the residuals to verify if they look like white noise. The plot below shows that all sample ACF values lie within the confidence interval, suggesting no significant autocorrelation. The PACF values are also mostly within bounds, with small spikes near lags 34 and 36. However, these exceed the confidence limits by a very small margin. It's important to note that the confidence intervals produced by R's `acf()` and `pacf()` functions are approximate, and tend to underestimate the true variability that Bartlett's formula more accurately captures.



Furthermore, running Yule-Walker estimation with the `ar()` command should select an AR model of order 0 to correspond to white noise, which it does.

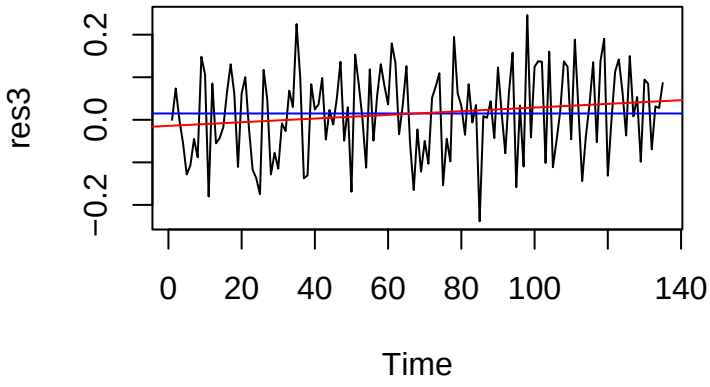
All Portmanteau tests passed at the 95% significance level with a Box-Pierce p-value of 0.6501, Ljung-Box p-value of 0.6055, and a McLeod-Li p-value of 0.6423. Thus, there is not sufficient evidence to reject the null hypotheses that the residuals resemble white noise and exhibit no non-linear dependence. Therefore, ARIMA(1, 1, 3) passes all diagnostic checks.

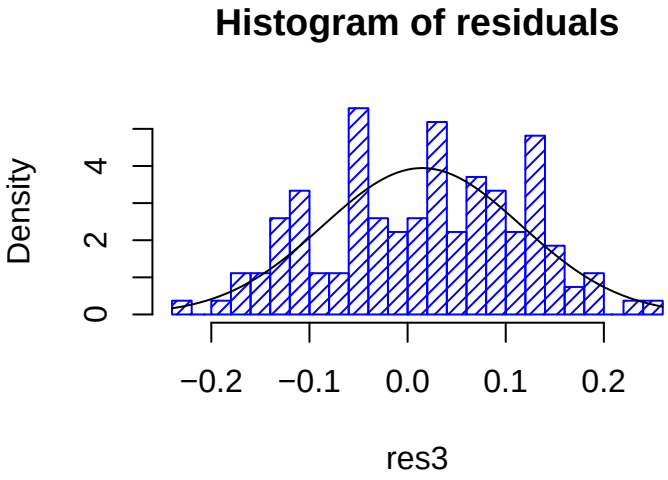
ARIMA(3, 1, 0) diagnostics:

ARIMA(3, 1, 0) also passes all diagnostics checks similar to ARIMA(1, 1, 3). For the sake of space, I will not include the figures or report the exact p-values. Their code can be found in the Appendix under "ARIMA(3, 1, 0) diagnostics".

ARIMA(0, 1, 2) diagnostics:

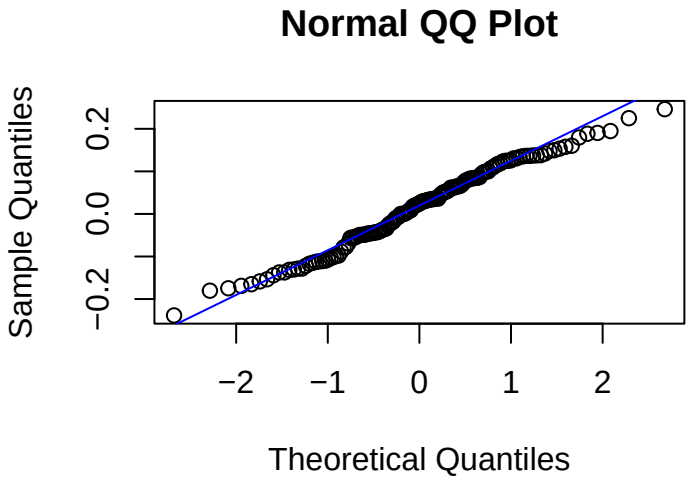
Below is a plot of the residuals of ARIMA(0, 1, 2), which can be seen to roughly follow white noise with no trend and mean: 0.01486, variance: 0.01023. If differenced again, the variance of the residuals increases to 0.021, so we know there is no trend to the residuals.





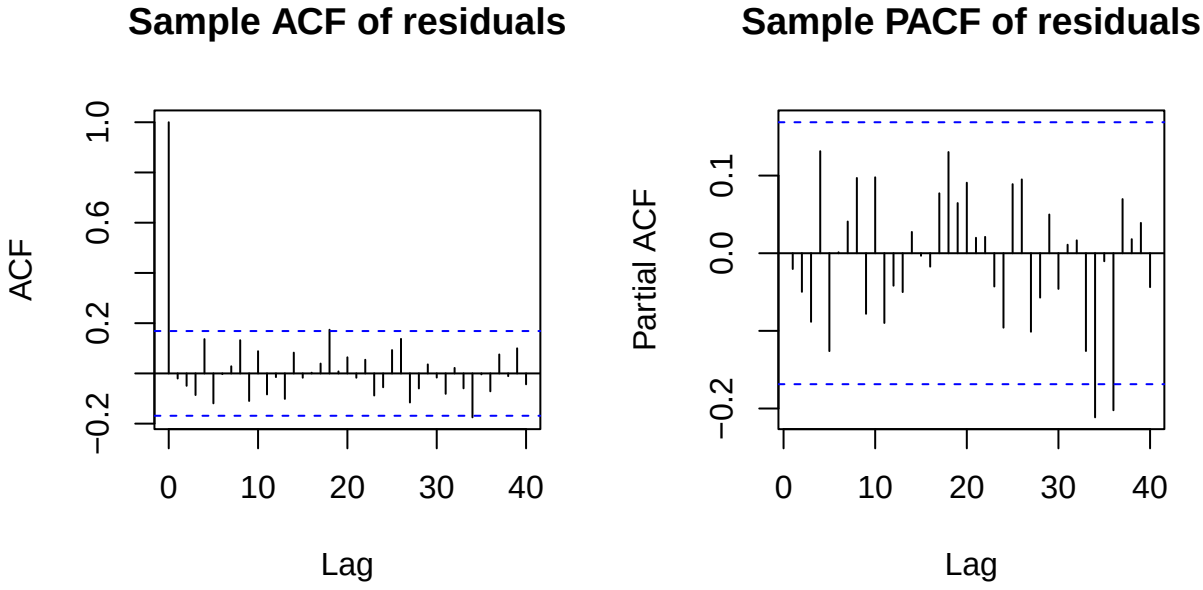
The histogram of the residuals above appear slightly normal and symmetric. Let's run other tests for normality before coming to any conclusions.

It can be seen that the residuals follow the the theoretical line of normality in the QQ Plot, only deviating from it slighly at each tail.



The residuals also pass the Shapiro-Wilk test for normality with an insignificant p value of 0.13246.

Next, checking the sample ACF and PACF of the residuals below verifies that they look like white noise. The sample ACF plot shows that all values lie within the confidence interval, suggesting no significant autocorrelation. The PACF values are also mostly within bounds, with small spikes near lags 34 and 36. However, these exceed the confidence limits by a very small margin. So, once again, because of R's confidence interval approximation, we can most likely take these values to be insignificant.



Yule-Walker estimation confirms that an AR model would be of order 0, further confirming white noise assumptions.

All Portmanteau tests passed at the 95% significance level with a Box-Pierce p-value of 0.2895, Ljung-Box p-value of 0.2386, and a McLeod-Li p-value of 0.5831. Thus, there is not sufficient evidence to reject the null hypotheses that the residuals resemble white noise and exhibit no non-linear dependence. Therefore, ARIMA(0, 1, 2) passes all diagnostic checks.

Model fitting conclusions:

Among the three candidate models, all passed diagnostic checking, were stationary and invertible models, and had similar AICc values. ARIMA(1, 1, 3) has the lowest AICc, but ARIMA(0, 1, 2) offers a more parsimonious alternative with an AICc close to that of the more complex ARIMA(1, 1, 3) model. Given that AICc can favor overparameterized models (i.e. overestimate p) and based on the principle of parsimony emphasized by Professor Feldman, I selected ARIMA(0, 1, 2) as the final model [4].

It is worth noting that the sample PACF of the once differenced time series suggested part of an autoregressive structure to the data with potential values of $p = 1, 2, \text{or } 3$ appearing most significant.

The final model equation is thus:

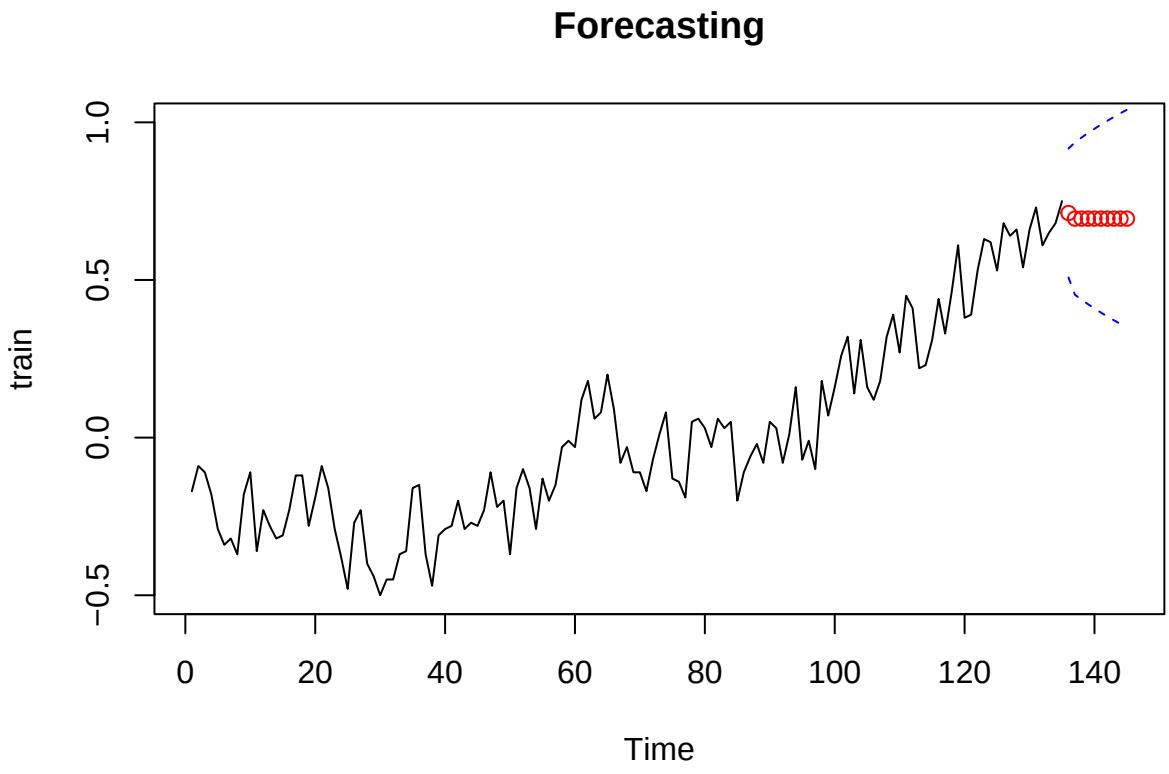
$$X_t(1 - B) = (1 - 0.3698B - 0.2038B^2)Z_t$$

which simplifies to:

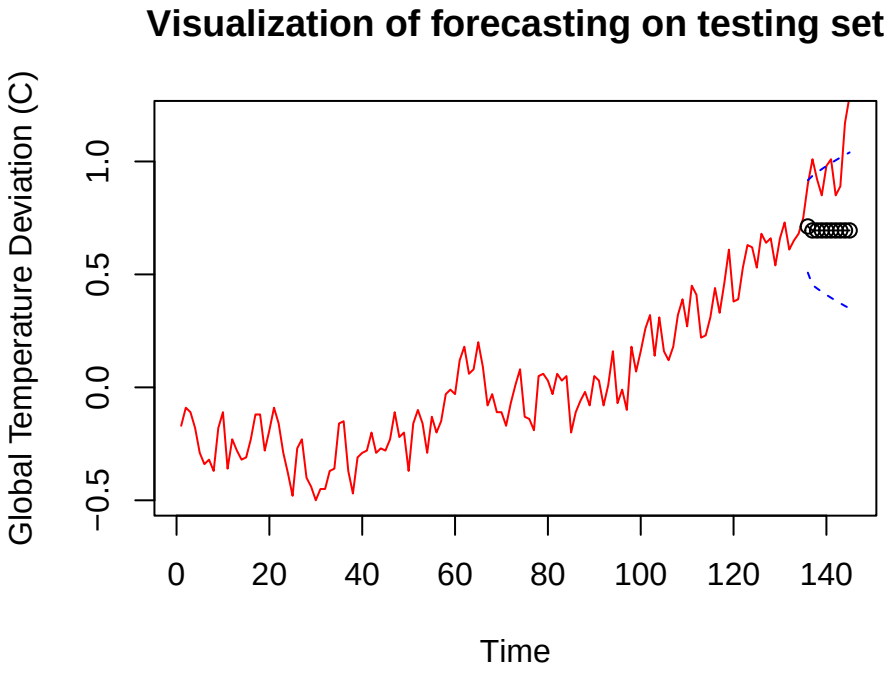
$$X_t - X_{t-1} = Z_t - 0.3698Z_{t-1} - 0.2038Z_{t-2}$$

Forecasting:

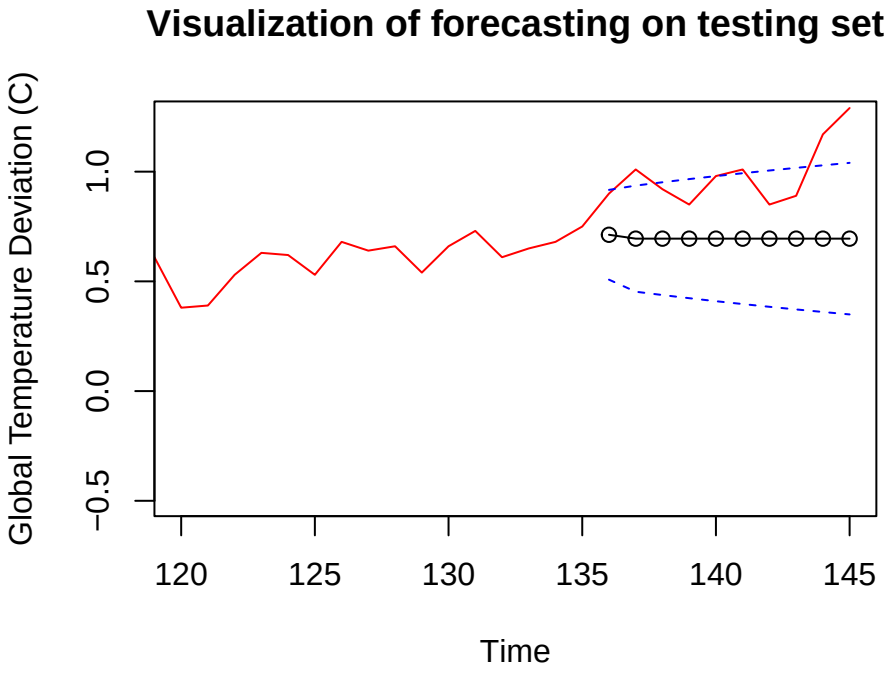
Below is the plot depicting the final model's 10-step ahead forecasts. The black line represents the training data and the red points are the forecasted points, with the dotted blue lines representing the forecast interval.



Now, adding the true global temperature deviations for the forecasted values onto the graph (in red), it can be seen that the model underestimates these values.



We can zoom in for a closer look, where the black points represent the model's forecasting:



All in all, the model's forecasts predicted no change in global temperature from 2015–2024. It successfully met the objective of Box–Jenkins methodology by producing a statistically sound univariate forecast based on historical patterns. However, the testing data show that this prediction errs slightly, as temperatures fluctuate and ultimately increase. The test set mostly falls within the 95% forecast interval, though it rises above it in the final two data points. Given this emerging deviation, future work could involve employing a multivariate time series model or more complex approaches to better capture underlying climate dynamics. Thank you for reading!

References

[1] GISS, NASA. “Global Surface Temperature.” NASA, NASA, 29 Jan. 2025, climate.nasa.gov/vital-signs/global-temperature/?intent=121.

[2] Posit team (2025). RStudio: Integrated Development Environment for R. Posit Software, PBC, Boston, MA. URL http://www.posit.co/.

[3] Hansen, James, et al. "Global temperature change." Proceedings of the National Academy of Sciences 103.39 (2006): 14288-14293.

[4] Feldman, Raya. PSTAT W 174, University of California, Santa Barbara. Spring 2025.

Appendix

```r
# load TS data of yearly Global Temperature changes (C) from 1880-2024
global <- read.table("Global Temp Dev Nasa.txt")

# convert to a Time Series object using just the annual mean temp change variable
temps.ts <- ts(global$V2, start = c(1880), frequency = 1)

# plot time series with mean and trend line
plot.ts(temps.ts, main = "Time Series Plot")
abline(h=mean(temps.ts), col = "blue")
index <- time(temps.ts)
trend1 <- lm(temps.ts ~ index)
abline(trend1, col = "red")
```

```r
length(temps.ts)

# use the last 10 observations for validation
global_test <- temps.ts[(length(temps.ts) - 9):length(temps.ts)]
train <- temps.ts[1:(length(temps.ts) - 10)]

# time series plot of the training set
plot.ts(train, main = "Training Data Plot")
abline(h=mean(train), col = "blue")
index2 <- as.numeric(1:length(train))
trend2 <- lm(train ~ index2)
abline(trend2, col = "red")
```

```r
# histogram of the training set
hist(train)
```

```r
# once differenced data and sample variance
var1 <- var(train)
g2 <- diff(train)
var2 <- var(g2)

# twice differenced data and sample variance
g3 <- diff(g2)
var3 <- var(g3)

# de-trended time series plot with mean and trend line
plot.ts(g2, main = "De-trended Time Series Plot")
abline(h=mean(g2), col = "blue")
index3 <- as.numeric(1:length(g2))
trend3 <- lm(g2 ~ index3)
abline(trend3, col = "red")
```

```r
hist(g2)
```

```r
# comparing sample ACF and sample PACF
par(mfrow = c(1, 2))
acf(g2, lag.max=40, main = "Sample ACF once Differenced")
pacf(g2, lag.max=40, main = "Sample PACF once Differenced")
```

```r
# qpcR package was discontinued so I am using the MuMIn package instead
library(MuMIn)
# run a loop for possible ARIMA models
for (i in 0:3){
  for (j in 0:4){
    print(i)
    print(j)
    print(AICc(arima(train, order = c(i,1,j), method = "ML")))}}

# testing AICc values for models with p=36, they remain commented because of runtime
#AICc(arima(train, order = c(36, 1, 0), method = "ML"))
#AICc(arima(train, order = c(36, 1, 1), method = "ML"))
#AICc(arima(train, order = c(36, 1, 2), method = "ML"))
#AICc(arima(train, order = c(36, 1, 3), method = "ML"))
#AICc(arima(train, order = c(36, 1, 4), method = "ML"))

# ARIMA(1, 1, 3)
fit1 <- arima(train, order = c(1, 1, 3), method = "ML")
coefs <- coef(fit1)
se <- sqrt(diag(fit1$var.coef))
# Create summary table
summary_table1 <- data.frame(
  Estimate = round(coefs, 4),
  Std.Error = round(se, 4))

# ARIMA(3, 1, 0)
fit2 <- arima(train, order = c(3, 1, 0), method = "ML")
coefs2 <- coef(fit2)
se2 <- sqrt(diag(fit2$var.coef))
# create summary table
summary_table2 <- data.frame(
  Estimate = round(coefs2, 4),
  Std.Error = round(se2, 4))

# ARIMA(0, 1, 2)
fit3 <- arima(train, order = c(0, 1, 2), method = "ML")
coefs3 <- coef(fit3)
se3 <- sqrt(diag(fit3$var.coef))
# create summary table
summary_table3 <- data.frame(
  Estimate = round(coefs3, 4),
  Std.Error = round(se3, 4))

# display tables
knitr::kable(summary_table1, caption = "ARIMA(1, 1, 3) Model Estimated Parameters")
knitr::kable(summary_table2, caption = "ARIMA(3, 1, 0) Model Estimated Parameters")
knitr::kable(summary_table3, caption = "ARIMA(0, 1, 2) Model Estimated Parameters")

# plot roots visualization
source("plot.roots.R")
plot.roots(NULL, polyroot(c(1, 0.6329, -0.5213, -0.2909)), main = "Roots of MA Part")
plot.roots(NULL, polyroot(c(1, -0.3555, -0.3427, -0.2304)), main = "Roots of AR Part")
```

```r
plot.roots(NULL, polyroot(c(1, -0.3698, -0.2038)), main = "Roots of MA Part")

# diagnostics
res1 <- residuals(fit1)
mean <- mean(res1)
var <- var(res1)
res1_diff <- diff(res1)
var_d <- var(res1_diff)

# plot residuals
plot.ts(res1)
abline(h=mean(res1), col = "blue")
index <- as.numeric(1:length(res1))
trend <- lm(res1 ~ index)
abline(trend, col = "red")
```

```r
# histogram of residuals
hist(res1, density=20, breaks=20, col="blue", prob=TRUE,
main = 'Histogram of residuals')
m <- mean(res1)
sd <-sqrt(var(res1))
curve( dnorm(x, m, sd), add=TRUE)

# QQ Plot
qqnorm(res1, main = 'Normal QQ Plot')
qqline(res1, col="blue")
```

```r
# Shapiro-Wilk Test
shap <- shapiro.test(res1)

# sample P/ACF
par(mfrow = c(1, 2))
acf(res1, lag.max = 40, main = "Sample ACF of residuals")
pacf(res1, lag.max = 40, main = "Sample PACF of residuals")
```

```r
# Yule-Walker estimation
ar(res1, aic = TRUE, order.max = NULL, method = c("yule-walker"))

# Portmanteau Tests
# lag is sqrt(n=135) = 12
Box.test(res1, lag = 12, type = c("Box-Pierce"), fitdf = 4)
Box.test(res1, lag = 12, type = c("Ljung-Box"), fitdf = 4)
# McLeod-Li test (fitdf = 0)
Box.test(res1^2, lag = 12, type = c("Box-Pierce"), fitdf = 0)

# ARIMA(3, 1, 0) diagnostics
res2 <- residuals(fit2)
mean2 <- mean(res2)
var2 <- var(res2)
res2_diff <- diff(res2)
var_d2 <- var(res2_diff)

plot.ts(res2)
```

```r
abline(h=mean(res2), col = "blue")
index <- as.numeric(1:length(res2))
trend <- lm(res2 ~ index)
abline(trend, col = "red")

hist(res2, density=20, breaks=20, col="blue", prob=TRUE,
main = 'Histogram of residuals')
m <- mean(res2)
sd <-sqrt(var(res2))
curve( dnorm(x, m, sd), add=TRUE)
```

```r
qqnorm(res2, main = 'Normal QQ Plot')
qqline(res2, col="blue")

shapiro.test(res2)

par(mfrow=c(1, 2))
```

```r
acf(res2, lag.max = 40, main = "Sample ACF of residuals")
pacf(res2, lag.max = 40, main = "Sample PACF of residuals")
```

```r
ar(res2, aic = TRUE, order.max = NULL, method = c("yule-walker"))

# lag is sqrt(n=135) = 12
Box.test(res2, lag = 12, type = c("Box-Pierce"), fitdf = 3)

Box.test(res2, lag = 12, type = c("Ljung-Box"), fitdf = 3)

# McLeod-Li test (fitdf = 0)
Box.test(res2^2, lag = 12, type = c("Box-Pierce"), fitdf = 0)

# ARIMA(0, 1, 2) diagnostics
res3 <- residuals(fit3)
mean3 <- mean(res3)
var3 <- var(res3)
res3_diff <- diff(res3)
var_d3 <- var(res3_diff)

# residuals plot
plot.ts(res3)
abline(h=mean(res3), col = "blue")
index <- as.numeric(1:length(res3))
trend <- lm(res1 ~ index)
abline(trend, col = "red")

# residuals histogram
hist(res3, density=20, breaks=20, col="blue", prob=TRUE,
main = 'Histogram of residuals')
m <- mean(res3)
sd <-sqrt(var(res3))
curve( dnorm(x, m, sd), add=TRUE)
```

```r
# QQ Plot
qqnorm(res3, main = 'Normal QQ Plot')
qqline(res3, col="blue")

# Shapiro-Wilk Test
shap3 <- shapiro.test(res3)

# residuals sample P/ACF
par(mfrow = c(1, 2))
```

```r
acf(res3, lag.max = 40, main = "Sample ACF of residuals")
pacf(res3, lag.max = 40, main = "Sample PACF of residuals")
```

```r
# Yule-Walker estimation
ar(res3, aic = TRUE, order.max = NULL, method = c("yule-walker"))

# Portmanteau tests
# lag is sqrt(n=135) = 12
Boxp3 <- Box.test(res3, lag = 12, type = c("Box-Pierce"), fitdf = 2)

Ljungp3 <- Box.test(res3, lag = 12, type = c("Ljung-Box"), fitdf = 2)
# McLeod-Li test (fitdf = 0)
McL3 <- Box.test(res3^2, lag = 12, type = c("Box-Pierce"), fitdf = 0)

# forecasting
library(forecast)
pred.tr <- predict(fit3, n.ahead = 10)
# 95% CI for prediction
U.tr = pred.tr$pred + 2*pred.tr$se
L.tr = pred.tr$pred - 2*pred.tr$se

# plot the predictions & CI
ts.plot(train, xlim = c(1, length(train)+10), ylim = c(min(train), 1),
        main = "Forecasting")
lines(U.tr, col = "blue", lty = "dashed")
lines(L.tr, col = "blue", lty = "dashed")
points((length(train)+1):(length(train)+10), pred.tr$pred, col = "red")

# adding true global temp deviations onto the graph
ts.plot(as.numeric(temps.ts), xlim = c(1, length(train)+10),
        ylim = c(min(temps.ts), 1.2), col = "red",
        ylab = "Global Temperature Deviation (C)",
        main = "Visualization of forecasting on testing set")
lines(U.tr, col = "blue", lty = "dashed")
lines(L.tr, col = "blue", lty = "dashed")
points((length(train)+1):(length(train)+10), pred.tr$pred, col = "black")
lines((length(train)+1):(length(train)+10), pred.tr$pred, col = "black")
```

```r
# zoomed in forecast with ground truths
ts.plot(as.numeric(temps.ts), xlim = c(120, length(train)+10),
        ylim = c(min(temps.ts), 1.25), col = "red",
        ylab = "Global Temperature Deviation (C)",
        main = "Visualization of forecasting on testing set")
```

```r
lines(U.tr, col = "blue", lty = "dashed")
lines(L.tr, col = "blue", lty = "dashed")
points((length(train)+1):(length(train)+10), pred.tr$pred, col = "black")
lines((length(train)+1):(length(train)+10), pred.tr$pred, col = "black")
```