Modeling and Forecasting Reservoir Storage

Stats 415 - Final Project

Guy Dotan

December 14, 2018

Contents

Figure 1: San Joaquin River System

1 Background

In the heart of California's Central Valley is the the San Joaquin River System consisting of several major tributaries such as the Stanislaus, Tuolumne, and Merced Rivers. My father is a civil engineer and has been tasked with a project regarding the **New Melones Reservoir**, the largest reservoir in the river system with a storage of about 2.4 million acre-feet of water.

The water in the Stanislaus River cascades through 3 reservoirs: New Melones, Tulloch and Goodwin. Since Tulloch and Goodwin are usually kept full, the release from New Melones dictates how much water can be diverted for agriculture and municipal needs and how much can remain down stream for environmental consideration.

The State of California is proposing a policy where 40% of the inflow to New Melones during Spring (Feb.-June) should be released below Goodwin. Assuming the diversion will remain the same, this means that less water could be stored in New Melones. This policy is currently being challenged by farmers and cities who claim that the reduction in the amount of water stored in New Melones will adversely impact water supply during drought conditions.

Figure 2 shows the schema of these cascading reservoirs.

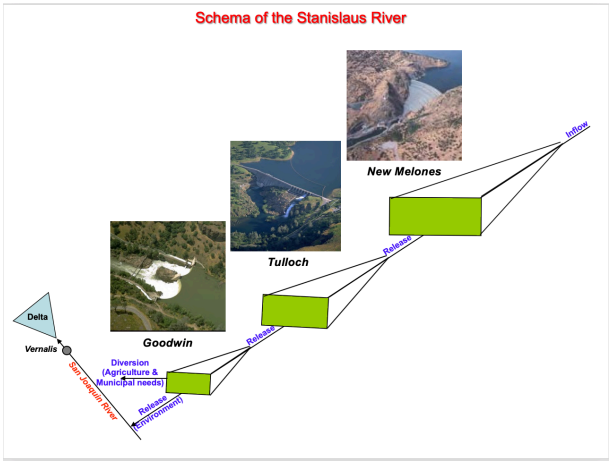

Figure 2: Stanislaus River Reservoir Schema

Our goal is to take a look at two separate sets of records. First, the historical storage levels of the New Melones Reservoir dating back to the mid-1980s. Second, what those storage levels would look like if the state were to have implemented that policy over the same time span. By comparing a forecast for each of these two models we could then come to an understanding of how much the historical projection would differ from the policy's projection.

2 The Data Set

The dataset was provided by my father who was given the information from the *California State Water Resources Control Board* (https://www.waterboards.ca.gov/). The New Melones reservoir was built in the late 1970's, but was first full in mid-1983. The historical data available spans from when it was first built up until the most recent measurements in 2018.

The policy storage data was created from a simulation done by the Water Sources Board. They took the historical trends and modeled what the impact would be with the enforcement of the policy. Unfortunately the simulation data ends in December of 2010. As a result the best data to use for this project was from March 9, 1983 to Dec. 10, 2010. However, to keep things simple, I decided to set the initial starting date as Jan. 1 1984. So our final date range is: 01/01/1984 - 12/12/2010 for a total of 9,841 observations.

The target variable used for the forecasting is the same for each model. We will be looking at volume (storage) in the reservoir measured in acre-feet. The historical data ranges from a maximum of `2367876` ac-ft to a minimum of `83631` ac-ft. The policy data ranges from a maximum of `2208511` ac-ft to a minimum of `84154` ac-ft.

A graph of the two datasets are plotted in Figure 3.

2.1 Stationarity

From the plot of the data we can see that our two time series definitely do not appear to be stationarity as both the variance and the mean seem to vary throughout the time range. The plot of both time series with their linear trends shows that policy data has a distinct negative slope while the historical trend has a distinct positive slope. The graphs and their slopes are shown in Figure 4. After running the Augmented Dickey-Fuller Test we see that the p-value for the policy time series is .422, while the p-value for the historical time series was .5158. In both cases we cannot accept the alternative hypothesis that our datasets are stationary and can conclude that they are both, in fact, non-stationary. The estimated regression line for policy is $x_t = 1226936.8 - 26.56t + w_t$. The estimated regression line for the historical data is $x_t = 1067717.1 + 34.50t + w_t$. From Table 1 and Table 2 we see that for both linear models the slopes and intercepts are statistically significant.

2.2 Fitting The Trend

In order to remove the apparent trend in the data we examined the previously mentioned linear regression. Although the slopes were statistically significant, the ADF tests did not yield any improvement in the time series trending toward stationary. Since the trend of both datasets did not resemble any simple nonlinear regressions, a smoothing spline was applied to the data sets. Figure 2 depicts the two datasets and their nicely fitted splines. Running the ADF test for each fitted spline resulted in a p-value less than .01 thus passing the test. We can see that the ACF and the PACF look significantly better on the detrended data from the spline in comparison to the original data (see Figure 6 and 7. However, there is still some apparent seasonality and the fact that both models fail the Ljung-Box test suggests there are more trends to remove.

3 Seasonality

From this point forward, I decided to only look at the historical data set as it was the one that is based on real logged data. In theory most of the same techniques used to model the historical data could be applied to the policy data set since it mainly a function of the historical set. But for convenience sake well will proceed with just the historical time series.

The first step toward eliminating the seasonal trend was to examine the periodogram of the detrended data's residuals. Spectral Analysis of the periodogram in Figure 8, suggests some peaks early on in the graph. When we convert the frequency to a cycle, we see that of the five most prominent cycle peaks, several of them are in the area of 365. This seems to suggest that a yearly seasonality trend (365 days) should be removed. Table 3 reflects this chart of the five most dominant cycles. In order to account for the yearly trend, we calculated the average storage on a per-year basis and removed those residuals from our detrended model. A graph of the yearly averages are shown in Figure 9.

There were plenty of other cycles that resulted in notable seasonality peaks in the periodogram. However, they did not seem to correlate to a familiar time pattern. Most of these cycles appeared between 1 year and 5 years, but with the yearly cycle occurring most prominently. As a result, we decided to not remove a second dominant frequency from our series.

4 Model Fitting

With our model detrended from the smoothing spline and then the yearly seasonality removed, I was comfortable to proceed with the ARIMA fitting. Beginning with the ACF and the PACF analysis an ARIMA of (3,0,1) looked to be appropriate. Running the auto.arima function resulted in settling on the ARIMA parameters of (3,0,1). Running some basic diagnostics on our ARIMA(3,0,1) revealed a standardized residuals chart without any noticeable problems, a remarkably improved ACF of the residuals compared to our original time series, but unfortunately it did not quite pass the Ljung-Box test. That said, this model has come a long ways from the raw data set that we began with. Figure 10 shows the results from the diagnostics while Figure 11 shows a deeper analysis of our model's residuals and its relatively normal distribution.

5 Model Forecasting

In order to test the effectiveness of this model we split the data into training and a test set. The training set consists of the all of the data up until the final year of our data set. Therefore the test set consists of the final 365 days in the model. Applying our ARIMA(3,0,1) model to the training set leaves us with predicted values for the final 365 days of the year. Unfortunately, this predict does not seem to very accurately account for some of the expected variation in our model. Instead it just hits the average and just remains at that trend. We have not demonstrated very much predictive power from this model and would require more analysis in order to apply this forecast to predictions beyond the scope of our dataset. Figure 12 shows the model predictions.

6 Further Analysis

Our results from this exercise were rather disappointing. Even though the spectral analysis heavily implied that the yearly trend was the most dominant influence in our time series, removing the yearly averages improved our model but did not make it robust enough for predictive power. Intuitively it does seem that adding in a season-by-season trend could be useful since we are aware that waterflow is based on seasons. Even though the Spectral Analysis did not suggest the significance of sub-yearly trends, it still might have been worth exploring based on common sense.

In addition, one of the challenges of modeling reservoir data is that it does reflect human intervention. While riverflow might be completely due to natural causes, reservoir storage is not. Changes in the water demands in the surrounding cities as well as climate changes would greatly influence how much water was diverted from the dam. It would be difficult to isolate which agricultural, environmental, or political trend could explain the changes in reservoir usage.

Finally, as for the policy time series, further analysis might actually be able to fit a model to it quite well as the policy series was built almost entirely off the historical trends. Further research could attempt to detrend the policy model by using the historical model. However, doing this might be rather trivial as we would essentially be reverse engineering the dataset and how it was generated in the first place.

Unfortunately, the practical purpose of this exercise was to attempt to determine a 5-10 year forecast for both the historical time series and the policy. Doing so would have allowed us to make a prediction as to what the water levels could be reasonably expected to occur down the road. Then from there determine if this policy will make a severe impact to those that require the water outflow from New Melones Reservoir. With this current model, extrapolation does not seem achievable and further digging into the intricacies of our data might reveal a better approach to counteract some of the noise in this time series.

7 Graphs and Figures

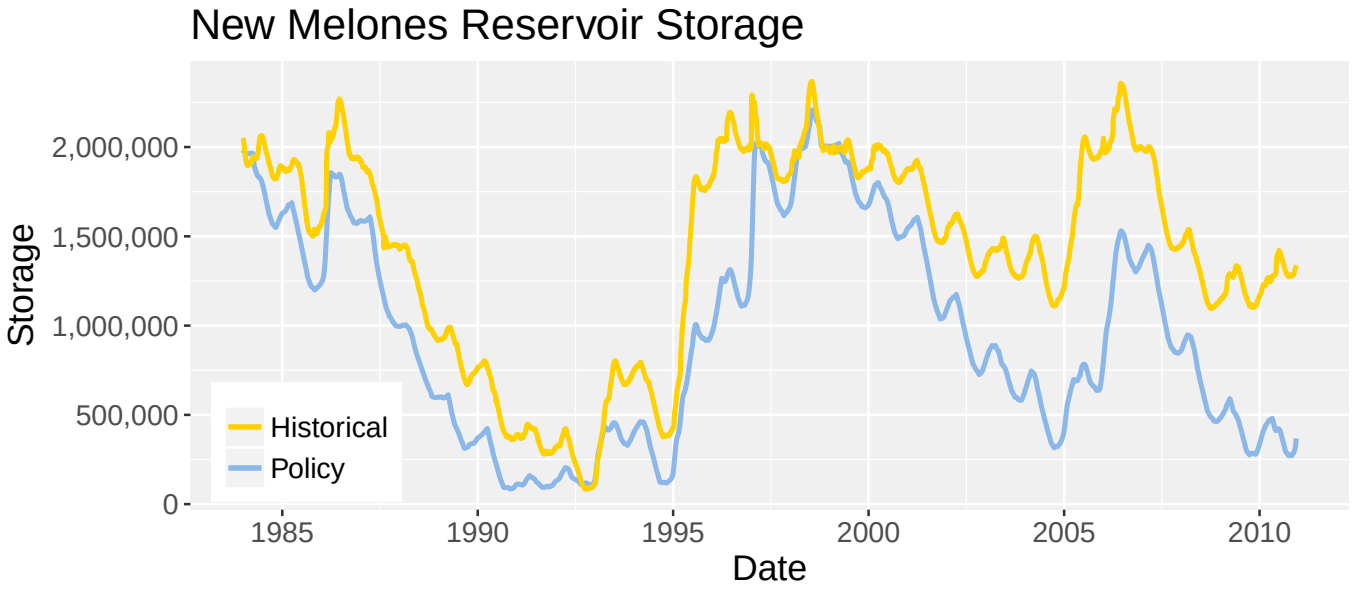

Figure 3: New Melones Reservoir Storage (Historical vs. Policy)

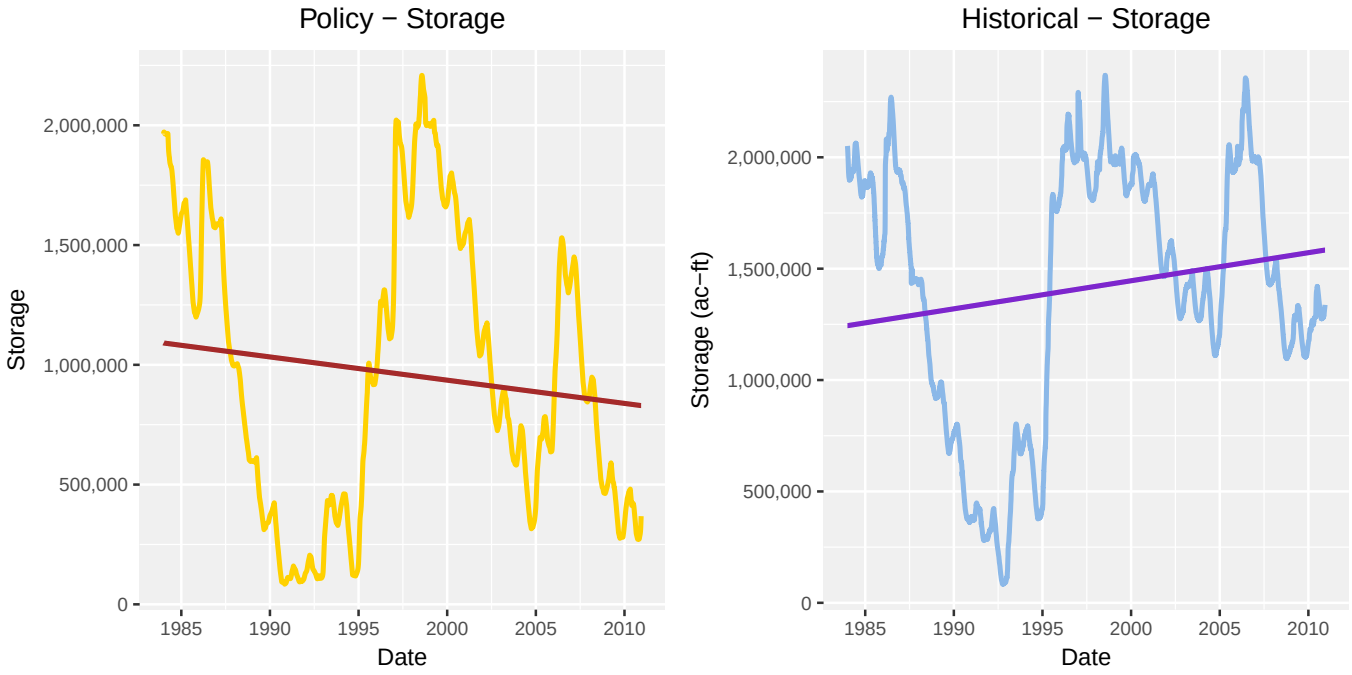

Figure 4: Trend Line for the Time Series

Table 1: Linear model for policy

Estimate	Std. Error	t value	Pr(>|t|)
1226936.80252	21896.504549	56.03345	0
-26.55893	2.099892	-12.64776	0

Table 2: Linear model for historical

Estimate	Std. Error	t value	Pr(>|t|)
1067717.09720	21129.85391	50.53121	0
34.50407	2.02637	17.02753	0

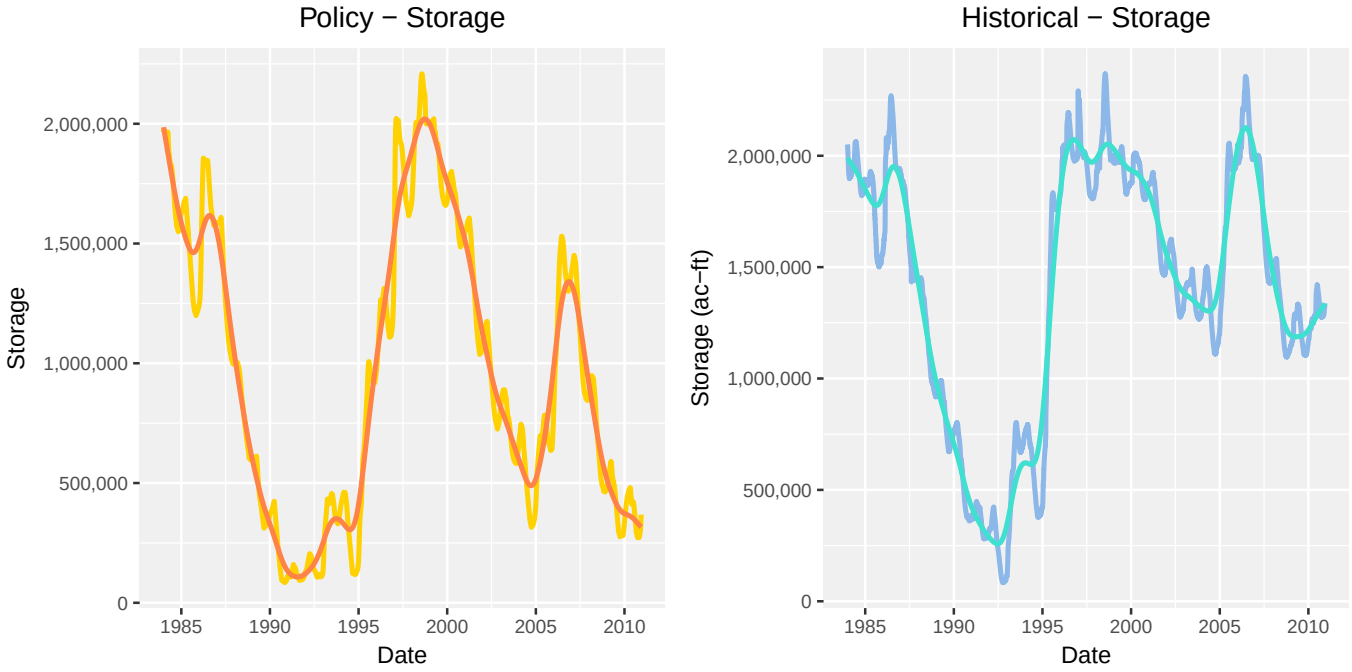

Figure 5: Smoothing Spline for the Time Series

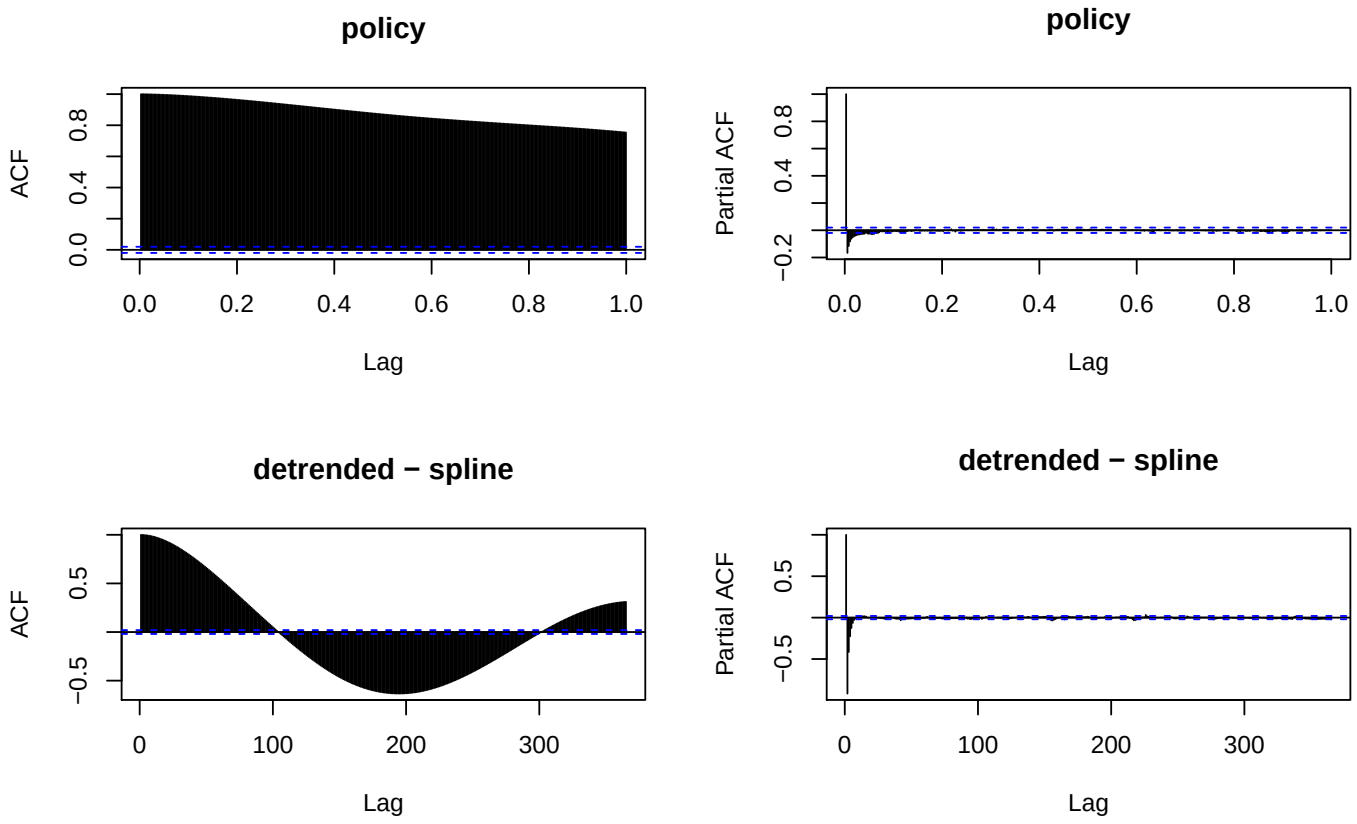

Figure 6: ACF and PACF for Original and Detrended Policy Series

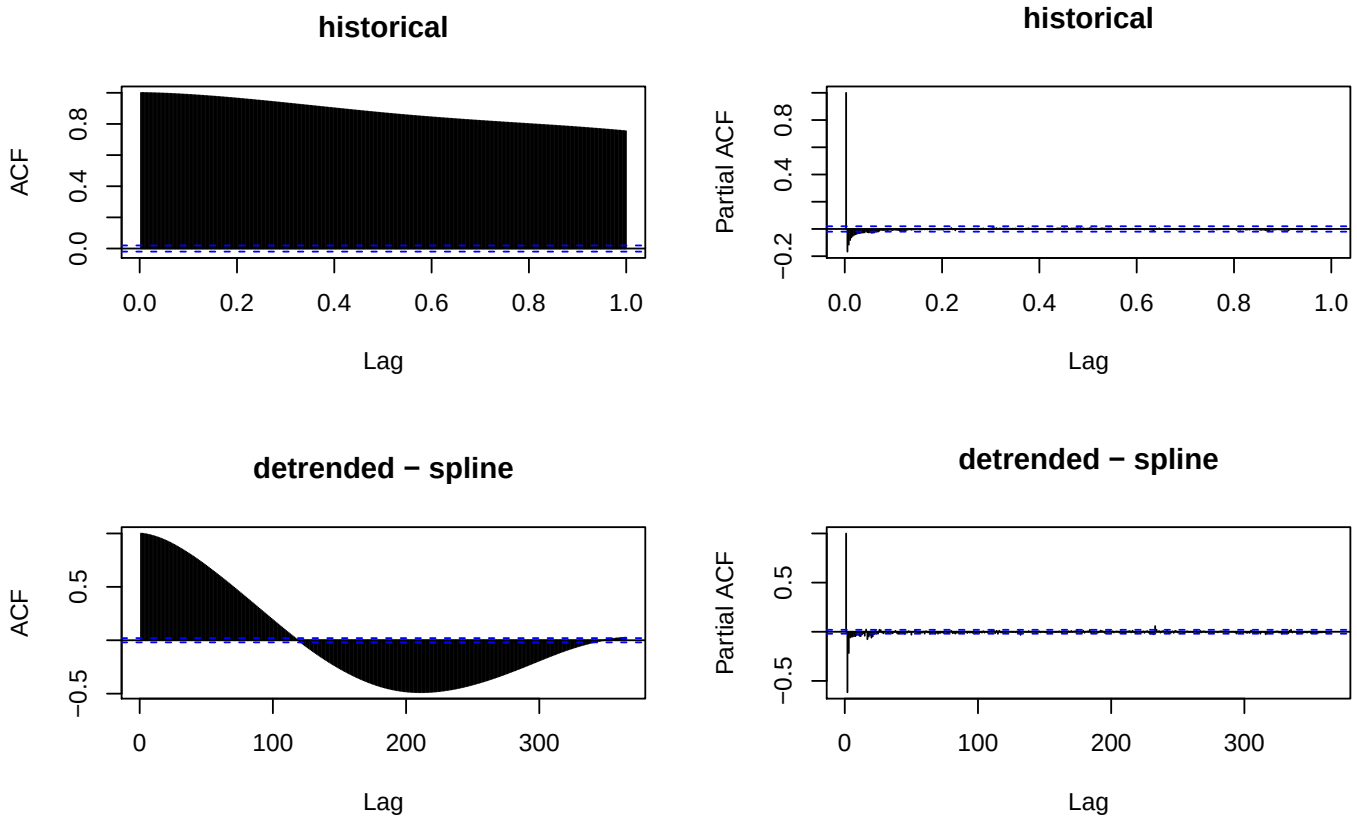

Figure 7: ACF and PACF for Original and Detrended Historical Series

Table 3: Top 5 Dominant Cycles for Historical Time Series

freq	spectrum	cycle
0.0027	13164075473268	370.3704
0.0016	6546760682407	625.0000
0.0011	5162549763809	909.0909
0.0028	4412472876301	357.1429
0.0012	3657850473763	833.3333

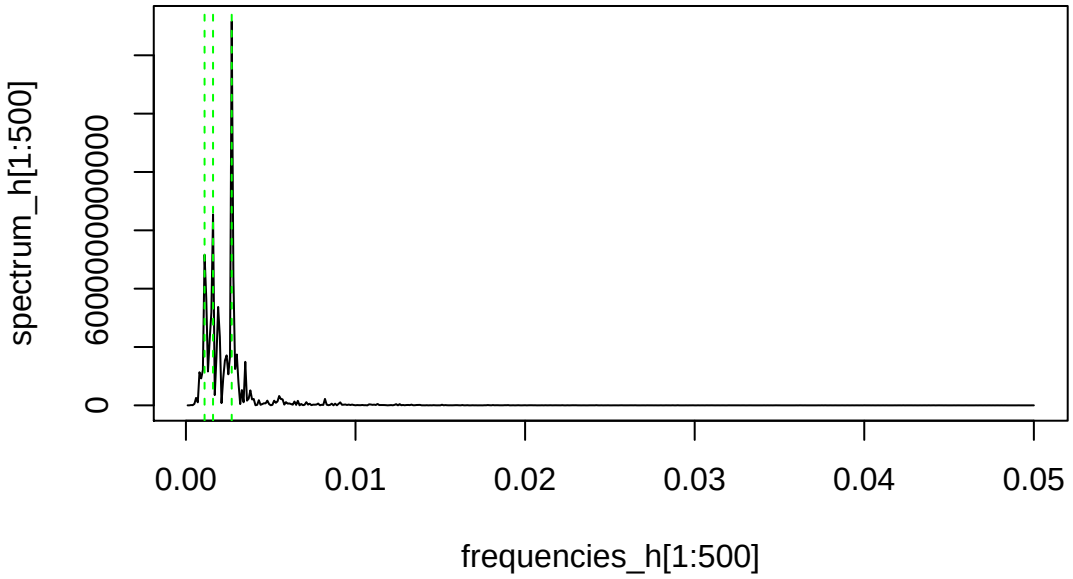

Figure 8: Periodogram for Historical Time Series

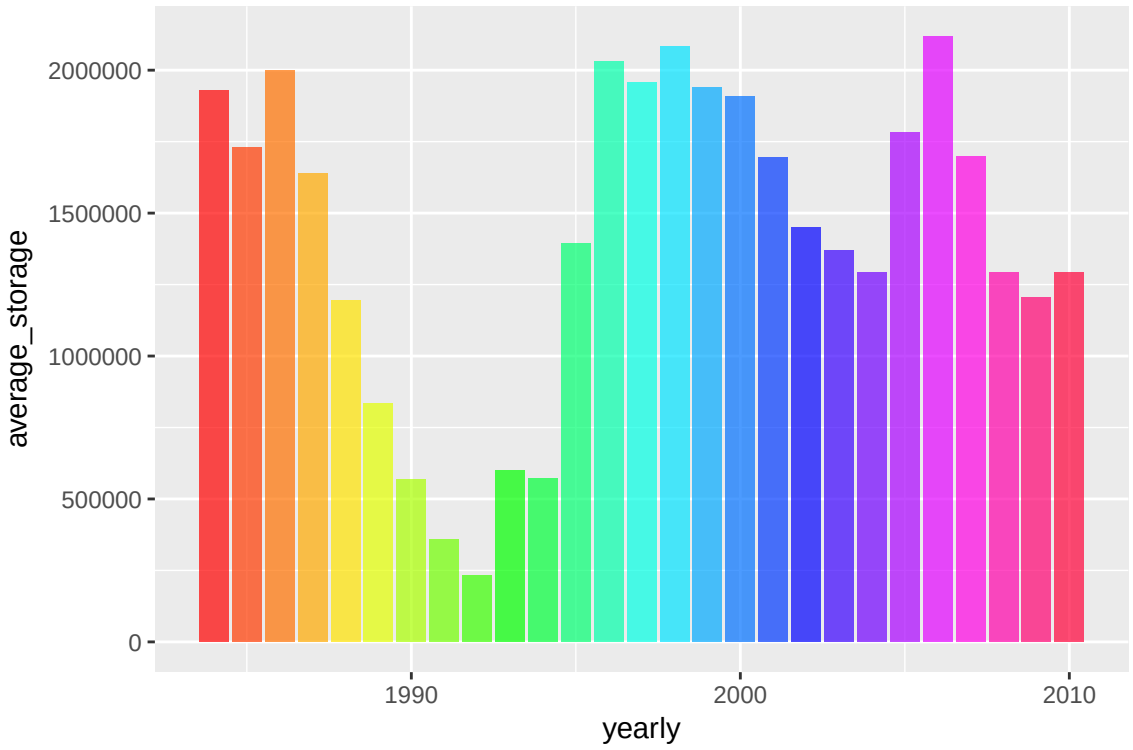

Figure 9: Plot of Yearly Averages

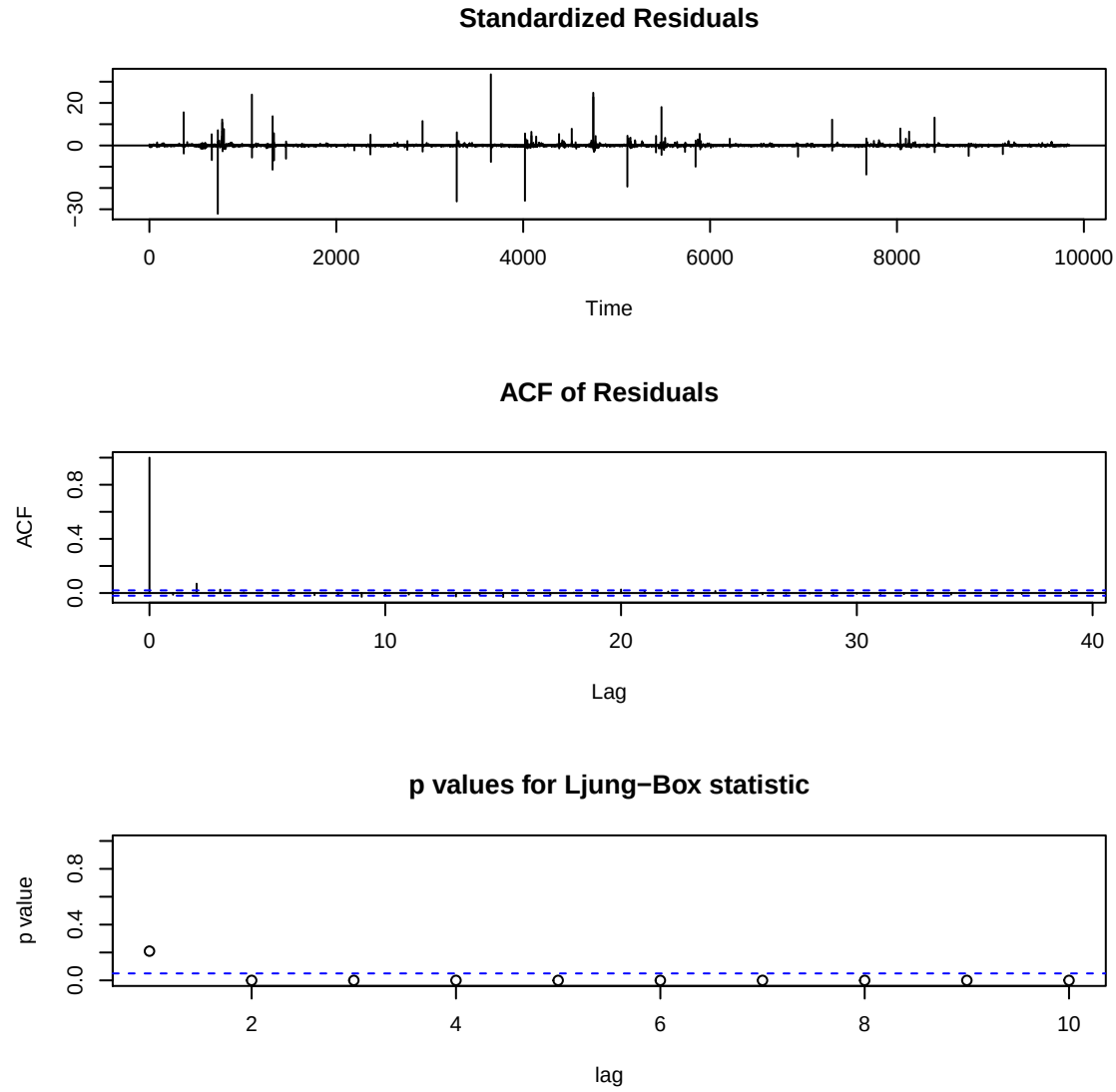

Figure 10: ARIMA(3,0,1) Model Diagnostics

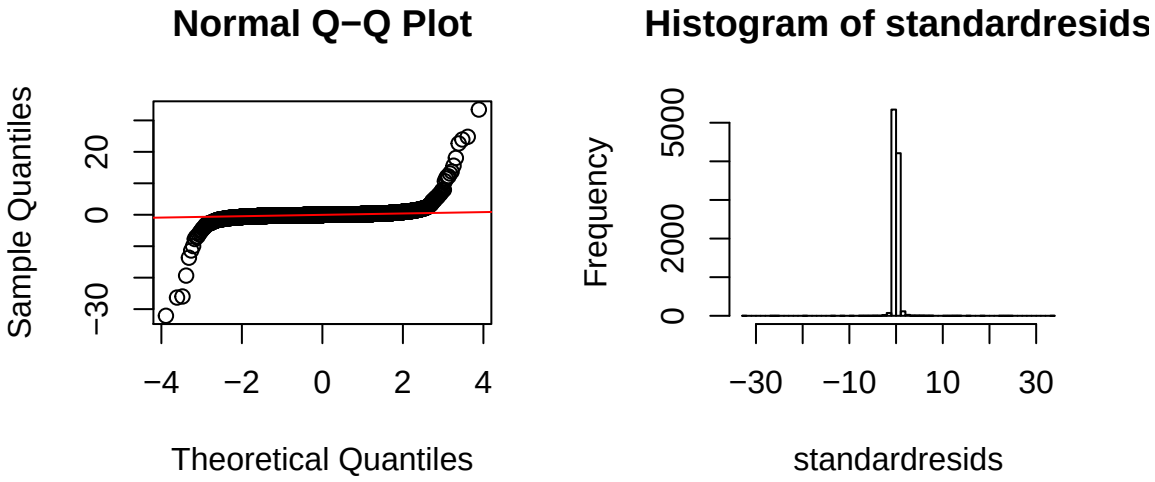

Figure 11: Analysis of the Residuals

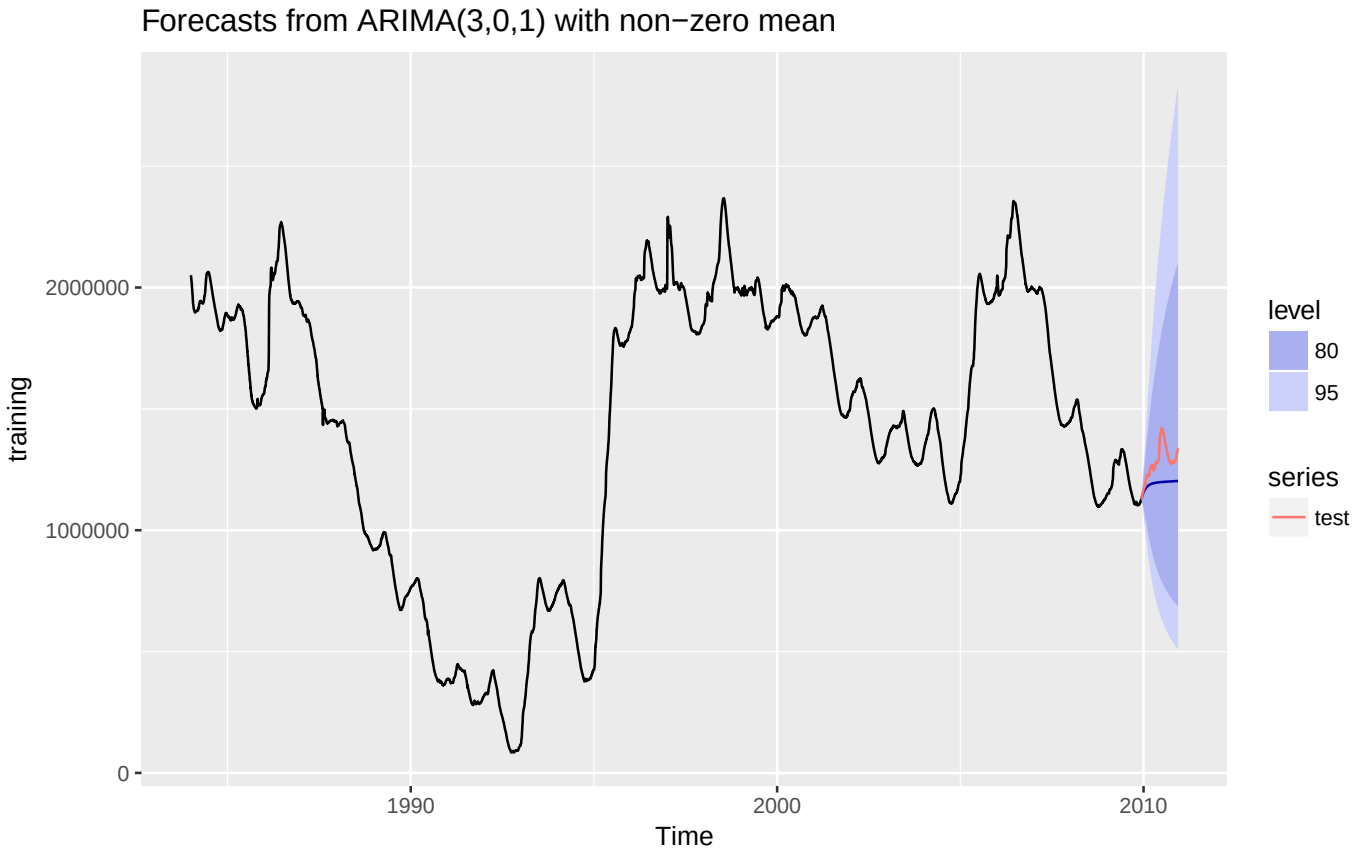

Figure 12: Analysis of the Residuals

8 Codebook

```
# load packages
library(ggplot2)    # pretty plots
library(gridExtra)  # ggplot formating
library(dplyr)      # transform data
library(tseries)    # time series
library(scales)     # scaling
library(astsa)      # time series analysis
library(TSA)        # more time series analysis
library(knitr)      # knitr format
library(kableExtra) # table building
library(FitAR)      # ts modeling
library(forecast)   # ts forecasing

## Set working directory folder
setwd("~/Documents/UCLA MAS/2018 Fall/STATS 415/Final Project")

# read in files
hist <- read.csv(file = "Data/historical_new_melones.csv", stringsAsFactors = F)
pol <- read.csv(file = "Data/policy_new_melones.csv", stringsAsFactors = F)

# format DATE as date
hist$DATE <- as.Date(hist$DATE, format = "%Y-%m-%d")
pol$DATE <- as.Date(pol$DATE, format = "%Y-%m-%d")

# merge sets
dat <- cbind(hist, pol)
dat <- dat[,c(1,2,3,6)]
names(dat)[3:4] <- c("STORAGE_HISTORICAL", "STORAGE_POLICY")

dat <- dat[dat$DATE >= as.Date("1984-01-01") & dat$DATE <= as.Date("2010-12-10"),]

# create time series - historical
hist.ts <- ts(as.numeric(dat$STORAGE_HISTORICAL), frequency = 365.25, start=1984)
summary(hist.ts)

# create time series - policy
pol.ts <- ts(as.numeric(dat$STORAGE_HISTORICAL), frequency = 365.25, start=1984)
summary(pol.ts)


### DETRENDING ###
# create spline - historical
mod_spline_hist <- smooth.spline(dat$INDEX, dat$STORAGE_HISTORICAL, df = 25)
pred_spline_hist <- data.frame(storage_pred = predict(mod_spline_hist, dat$INDEX)$y, date=dat$DATE)

# create spline - policy
mod_spline_pol <- smooth.spline(dat$INDEX, dat$STORAGE_POLICY, df = 25)
pred_spline_pol <- data.frame(storage_pred = predict(mod_spline_pol, dat$INDEX)$y, date=dat$DATE)

hist.spline <- dat[,c("DATE","STORAGE_HISTORICAL")] %>%
  mutate(line_fit = as.numeric(pred_spline_hist$storage_pred)) %>%
  mutate(resids = STORAGE_HISTORICAL - line_fit) %>%
  select(resids)

detrended_hist <- as.ts(hist.spline$resids)
```

```r
adf.test(detrended_hist)

pol.spline <- dat[,c("DATE","STORAGE_POLICY")] %>%
  mutate(line_fit = as.numeric(pred_spline_pol$storage_pred)) %>%
  mutate(resids = STORAGE_POLICY - line_fit) %>%
  select(resids)

detrended_pol <- as.ts(pol.spline$resids)
adf.test(detrended_pol)

# test Ljung-Box test
LBQPlot(detrended_pol)
LBQPlot(detrended_hist)


## ACF/PACF detrended
# ACF/PACF - policy
par(mfrow=c(2, 2))
acf(pol.ts, lag.max = 365, main='policy')
pacf(pol.ts, 365, main='policy')
acf(detrended_pol, 365, main='detrended - spline')
pacf(detrended_pol, 365, main='detrended - spline')

# ACF/PACF - historical
par(mfrow=c(2, 2))
acf(hist.ts, lag.max = 365, main='historical')
pacf(hist.ts, 365, main='historical')
acf(detrended_hist, 365, main='detrended - spline')
pacf(detrended_hist, 365, main='detrended - spline')


### SPECTRAL ANALYSIS ###
# Periodogram
pgram_hist <- spec.pgram(detrended_hist, taper = 0, log ="no", detrend = FALSE , main = "")
frequencies_h <- pgram_hist$freq
spectrum_h <- pgram_hist$spec

# Spectral values to find high frequency points
freqdf <- data.frame(freq = frequencies_h, spectrum = spectrum_h) %>%
  arrange(desc(spectrum_h)) %>%
  head(100) %>%
  mutate(cycle = 1/freq)


# now our top cycles are all weekly
# HISTORICAL
yearly_ts_hist <-  dat[,c(2,3)] %>%
  mutate(line_fit = as.numeric(pred_spline_hist$storage_pred)) %>%
  mutate(resids = STORAGE_HISTORICAL - line_fit) %>%
  mutate(yearly = year(DATE)) %>%
  group_by(yearly) %>%
  summarise(average_resid = mean(resids), average_storage = mean(STORAGE_HISTORICAL))
yearly_ts_hist

#remove yearly cycle
yearly_removets_hist <- dat[,c(2,3)] %>%
  mutate(line_fit = pred_spline_hist$storage_pred) %>%
```

```
  mutate(resids = STORAGE_HISTORICAL - line_fit) %>%
  mutate(yearly = year(DATE))

test <- left_join(x = yearly_removets_hist, y=yearly_ts_hist[,c(1,2)], by = "yearly")
yearly_removets_hist <- test %>%
  mutate(resids_nocycle = resids - average_resid)

detrend_decyclets_hist <- yearly_removets_hist$resids_nocycle


# look at series
yearly_removets_hist %>%
  ggplot(aes(x=DATE, y = resids_nocycle)) +
  geom_line()

#adf test
adf.test(detrend_decyclets_hist)


#bar chart of avgs
xxx <- as.data.frame(yearly_ts_hist)
g <- ggplot(xxx, aes(yearly,average_storage))
g+geom_col(fill = rainbow(n = length(xxx$yearly)), alpha = .7)


#cycle removed
acf(detrend_decyclets_hist, lag.max = 365)
pacf(detrend_decyclets_hist, lag.max = 365)

acf(detrend_decyclets_hist, lag.max = 365)
pacf(detrend_decyclets_hist, lag.max = 365)

LBQPlot(detrend_decyclets_hist)


### ARIMA SELECTION ###
# Loop to search for best ARIMA model
n = length(dat[,2])
AIC = matrix(cbind(rep(0,5), 5), nrow = 5, ncol = 5) -> AICc -> BIC
for (p in 1:5){
  for (q in 1:5){
    fit = arima(detrend_decyclets_hist, order = c(p, 0, q))
    AIC[p,q] = AIC(fit)
    BIC[p,q] = AIC(fit, k = log(length(n)))
  }
}
hist_models <- as.data.frame(AIC)

# model selecting
model_hist <- auto.arima(detrend_decyclets_hist, stepwise = F,
                         approximation = F, max.order = 5, allowdrift = T, trace=T)
thearima <- sarima(detrend_decyclets_hist, 3,0,1)

# model diagnostics
tsdiag(model_hist)
standardresids <- rstandard(model_hist)
qqnorm(standardresids)
```

```r
qqline(standardresids, col = "red")
hist(standardresids, breaks= 50)


### FORECAST ###
#train and test
training <- subset(hist.ts, end=length(hist.ts)-365)
test <- subset(hist.ts, start=length(hist.ts)-364)
mod.train <- Arima(training, order=c(3,0,1), lambda=0)

fore <- predict(model, 365)

mod.train %>%
  forecast(h=365) %>%
  autoplot() + autolayer(test)
```