An Analysis of Reservoir Levels in California (WIP)

Jeremy Chan, Fourth-Year Economics and Statistics Major

Table of Contents

1 Summary

This report contains an analysis of water reservoir levels in 11 selected California reservoirs. The ultimate goal of the report was to provide a summary of the current reservoir storage as a percentage of total capacity and to provide a forecast of these values for the year of 2016. As the report was written in May of 2016, there are already some data points from 2016 which will be compared with the forecasted points. To accomplish the task, first the data was plotted and inspected. A stationary time series was then modeled and predictions made from the residuals of that model. Lastly, the forecasts were plotted and listed in the results section. Additionally, the model for each reservoir was used to predict 2014 and compared against actual values to further test the accuracy of the models. In general, the models for 2016 were found to be not very accurate, while the 2014 models were more accurate, on average. This may indicate that the forecasting process is more accurate in non-El Niño years than El Niño years, but more research is required to come to a conclusive answer.

2 Introduction

Since 2012, California has faced severe drought conditions, the likes of which the state has not seen since the 1970s. As not only the most populous state but also the state which produces the most food[1], California faces unique challenges during a drought. It is, therefore, of great interest to track the level of drought which California faces and to make predictions on if and when the state will return to normal water levels.

According to the United States Geological Service, "a drought is a period of drier-than-normal conditions that results in water-related problems" resulting from "less than normal [rainfall] for several weeks, months, or years."[2] One key indicator of drought conditions is reservoir levels. During period of drought, reservoirs will The California Department of Water, provides reservoir information on 199 different reservoirs all throughout the state[3]. Of these, 199 reservoirs, eleven were selected for analysis in this report. The 11 that were selected were part of a group of 12 which the California Department of Water uses for their daily report on the conditions for "major

[1] http://www.ers.usda.gov/faqs.aspx

[2] http://ca.water.usgs.gov/data/drought/

[3] http://cdec.water.ca.gov/misc/resinfo.html

reservoirs."[4] These 11 reservoirs create a fairly representative picture of the overall reservoir levels in California and have been a steady indicator of drought levels. Thus, it is of interest to analyze and predict the levels in these reservoirs.

All of the data from the remaining reservoirs was retrieved through a query function from the California Department of Water Resources through the California Data Exchange Center. The data for each reservoir's capacity was queried at a monthly level, from the first recorded measurement to the most recent measurement of April 2016 and measured in acre-feet, with one acre-foot being equivalent to 325,851 gallons of water.

The ultimate goal of this report is to provide an overview of the selected reservoirs' raw storage as a percentage of its capacity, provide a forecast of these values for the 2016 calendar year for each reservoir, and to compare each reservoir's forecasted storage levels with the actual observations from 2016, up through the present time. To accomplish this, the first step is to retrieve, inspect, and clean the data which has been retrieved. Any outliers will be removed if they are deemed as incorrect readings and any non-stabilized variance will be corrected for. Additionally, if there are any reservoir readings which appear clearly non-linear, they will be adjusted for on a case-by-case basis.

The next step is to remove any deterministic component from each time series. Trend and seasonality will be removed and once stationarity has been established, a model will be fitted to the remaining residuals. From that model, a 12-month forecast for the calendar year of 2016 will be created. If the residuals from the model are normal, then a prediction interval will be provided in addition to the point forecasts.

Finally, the model will be evaluated against the actual values from 2016. Since 2016 is an El Niño year, the resultant models may not be accurate. As such, each model will also be used to forecast 2014 and compared against actual 2014 values to determine if the models are accurate for both El Niño and non-El Niño years.

3 Methods

In this section, the methods used for analysis is detailed. The analysis will be split into the following sections: Data Retrieval, Data Cleaning, Data Visualization, Removing

[4] http://cdec.water.ca.gov/cdecapp/resapp/getResGraphsMain.action

Deterministic Components, Fitting a Time Series Model, and Forecasting. Due to the number of reservoirs that are being analyzed, all of the above tasks were combined into a single function in R which simplifies the actual coding analysis aspect of this report.

3.1 Data Retrieval

All of the data contained in this report was retrieved from the California Department of Water through the California Data Exchange Center (CDEC). To automate the retrieval of this data, the package "sharpshootR" was used. The package contains a function called CDECquery which retrieves the reservoir levels for a given reservoir between a certain date range and at a certain frequency.

For this report, the following 11 reservoirs were used as a representative sample of California's overall reservoir levels (Reservoir abbreviations denoted in parentheses): Trinity Lake (CLE), Lake Shasta (SHA), Lake Oroville (ORO), New Melones (NML), Folsom Lake (FOL), Don Pedro (DNP), San Luis (SNL), Millerton Lake (MIL), Pine Flat (PNF), Castaic Lake (CAS), and Lake Perris (PRR).

The data was retrieved on a monthly basis spanning from when each reservoir was first opened up through April 2016. Each reservoir's data contained the following: date and time of each observation, year of each observation, month of each observation, the raw reservoir storage, and the ID of each reservoir (the three letter reservoir abbreviation).

After retrieving the raw storage readings for each of these reservoirs, the capacity of all California reservoirs was retrieved from the CDEC[5]. From this table of reservoir capacities, the specific capacities for our reservoirs of interest were saved within R.

The raw reservoir storage was divided by each reservoir's overall capacity, resulting in a percentage of capacity for each reservoir (henceforth referred to as a reservoir's capacity level). Capacity level was used as the primary unit of observation for ease of interpretation. Since converting to percentage of capacity is a linear transformation, it should not affect the final results of the time series model and forecast.

Note: Each reservoir was stored in a list; moving forward, an individual item in this list will be referred to as a "reservoir object" which contains all of the data for a single reservoir including date and time of each observation, year of each observation,

[5] http://cdec.water.ca.gov/misc/resinfo.html

month of each observation, the raw reservoir storage, the ID of each reservoir (the three letter reservoir abbreviation), and the reservoir's capacity level.

3.2 Data Inspection

After the data had been retrieved, each reservoir's historic capacity levels were plotted to look for the following problems: inconstant variance, large number of NA values, any sharp trend changes in the data, and outliers in the observations. Following that step, notes were made on which reservoirs needed additional inspection based on immediate visual inspection.

If inconstant variance was found, a log transformation would have been introduced to stabilize the variance so that the time series could be properly modeled. However, no times series were found to have seriously inconstant variance.

To check for NA values, each reservoir had its total number of NA values summed up. If they had a NA values that were spaced at least two values away from another NA value, the value was handled by imputation (detailed in Section 3.4).

Sharp trend changes and outliers in the data were handled on a case-by-case basis. To address a sharp trend change, historical data was consulted to see if the change in trend was due to a statewide drought or water shortage. Additionally, that same time period was cross-checked with other reservoirs to see if there was a trend across the state. All sharp trend changes and outliers were explained by either historical water occurrences or documented changes to the reservoir, thus, no further transformation was required.

3.3 Data Cleaning

To clean the data, the aforementioned NA checking process was utilized. NAs which were found to be far enough away from each other were handled by imputation. Any NA values which did not fit this process were handled on a case-by-case basis and are described in the individual results (detailed in Section 4). Additionally, as there was only reservoir with a major drop not related to state-wide water conditions, the data for that reservoir was handled in a singular manner (detailed in its individual results in Section 4).

3.4 Forecasting

To forecast the 2016 calendar year for each reservoir, a generalized function was created which would take a reservoir object in as an input. The function (henceforth referred to as forecast.all) then works to derive a stationary time series and model that time series. Forecast.all was written in such a way that it will predict the calendar year for the latest month it has data for, e.g., if your latest observation is January 2017, it will predict all of 2017. For the purposes of this project, 2016 was the predicted year.

Forecast.all first generates a time series of reservoir capacity percentages based on the first January observation and the last December observation in preparation for using sum of harmonics to remove seasonality from the time series (this abbreviated series will be henceforth referred to as "year series"). After creating the time series, any missing values are imputed by calculating an average of the two previous and two following observations around the NA value. If there are less than two values before after the missing value, a simple average of the previous and next value is used.

Next, forecast.all removes any deterministic trends from the year series. First, a first-order difference operation is used to remove any trend. Next, a sum of harmonics operation is used to remove the seasonality from the year series and fit a model. The residuals are then obtained from the de-trended and de-seasonalized year series.

These de-trended residuals are then evaluated for stationarity using both the Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test and the Augmented Dickey-Fuller (ADF) tests for stationarity. If the de-trended residuals are found to be not stationary by either of these tests, a warning is produced by forecast.all stating that the resultant series is not stationary, but the function continues the process of forecasting.

Next, the de-trended residuals are tested for independence using the Ljung-Box test for independence. If the de-trended residuals are found to be dependent, forecast.all produces a warning, indicating that other methods of de-trending the year series may be necessary.

After testing for independence, the de-trended residuals are tested for normality for the use of creating a prediction interval. The residuals are tested using the Shapiro-Wilk test and forecast.all will produce a warning if the residuals are found to be non-normal by this test. Additionally, a histogram of the residuals is produced, which is often a better indication of normality of the residuals. The histogram is displayed within the R plot panel. If the residuals are not normal, the prediction intervals may not be accurate.

Finally, the forecasting begins with forecasting the noise from the de-trended residual model. 12 periods corresponding to the 12 months of 2016 are forecasted at a 95% confidence level. The seasonal values are then fitted back onto the noise forecast, along with the upper and lower bound of the noise forecast, creating a 95% prediction interval. Finally, the seasonalized noise forecast and upper and lower seasonalized bounds are undifferenced, creating three vectors: a point forecast, an upper bound, and a lower bound.

Forecast.all then generates four separate time series: one with all of the observed values from the CDEC, one with the point forecast for 2016, one with the upper prediction bound for 2016, and one with the lower prediction bound for 2016. Finally, forecast.all produces a series of plots: the raw data, the cleaned data, the cleaned data with the forecast and prediction interval, and a zoomed-in view of the cleaned data, forecast, and prediction interval including the year predicted and the year prior.

The function thus returns four plots of the data, a histogram of the residuals, and a number of other objects (the fitted model, point forecast, lower prediction bound, upper prediction bound, and residuals) in text form to the user for analysis and presentation.

3.5 Forecast Comparison

As a method of analyzing the relative prediction power of the forecasting process, a separate forecasting function, forecast.2014 was created. Forecast.2014 is a replication of the forecast.all function, but it specifically creates a forecast for 2014, as 2014 was the last year without an El Niño and without any effects of the onset of an El Niño as 2015 had. This serves as a visual check to assess the predictive power of the forecasting process during non-El Niño years in comparison to its predictive power during El Niño years. After running each reservoir's data through forecast.2014, the graphs were inspected in comparison to the 2016 predictions.

4 Results

The results section will be divided by individual reservoir for ease of finding specific reservoir forecasts. Please refer to the table of contents for a page numbers for each reservoir.

4.1 Trinity Lake Reservoir (CLE) Forecast Results

4.1.1 Trinity Lake Reservoir (CLE) Data Retrieval

Trinity Lake Reservoir (CLE) data was retrieved from the CDEC using sharpshootR for the date range of October 1961 to April 2016 at a monthly level, totaling 655 observations. The raw reservoir storage was then divided by Trinity Lake's capacity of 2,447,650 acre-feet of storage capacity to find the capacity percentage of every observation.

4.1.2 Trinity Lake Reservoir (CLE) Data Inspection

Upon initial inspection, the data for Trinity Lake (shown below in **Figure 1**) did not appear to require any additional transformation or cleaning.

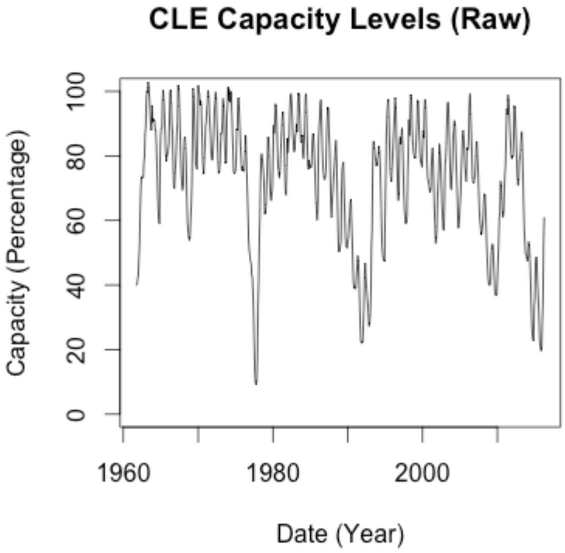


Figure 1. Raw data for Trinity Lake Reservoir (CLE).

After checking for any NA values, none were found, meaning that no imputation was required for this reservoir. Sharp drops in the late 1970s, early 1990s, and early 2010s were accounted for by statewide drought conditions during those times. As such, the observations are recorded as intended and are not incorrect measurements.

4.1.3 Trinity Lake Reservoir (CLE) Data Cleaning

No data cleaning was required for Trinity Lake capacity percentage observations. The cleaned data is presented below in **Figure 2**.

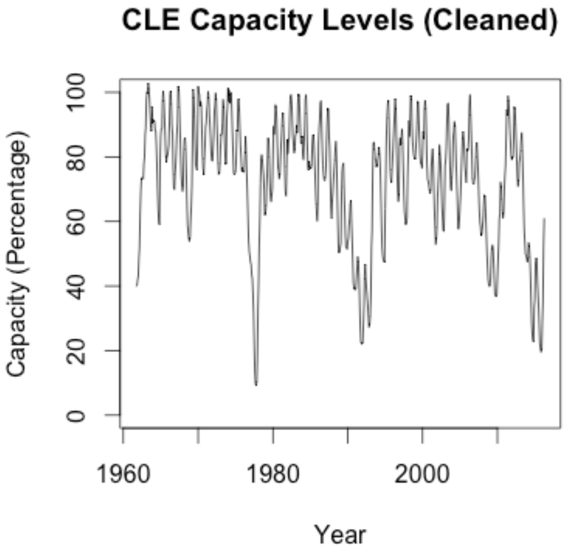


Figure 2. Cleaned data for Trinity Lake Reservoir (CLE).

4.1.4 Trinity Lake Reservoir (CLE) Forecasting

Using the forecast.all function, an ARMA(3,2) model was derived to fit the stationary data. The residuals are presented below in **Figure 3**.

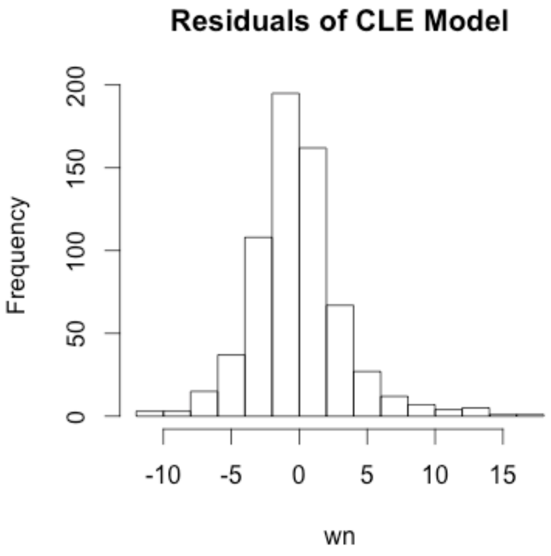


Figure 3. Histogram of residuals from the ARMA(3,2) model fitted to Trinity Lake Reservoir (CLE).

Based on the histogram, it appears that the residuals are mostly normal with a slightly right skew. According to the Shapiro-Wilk test, normality is rejected, but the histogram demonstrates that the residuals are approximately normal. The 95% prediction intervals presented in **Figure 4** below may not be accurate, but should be pretty close.

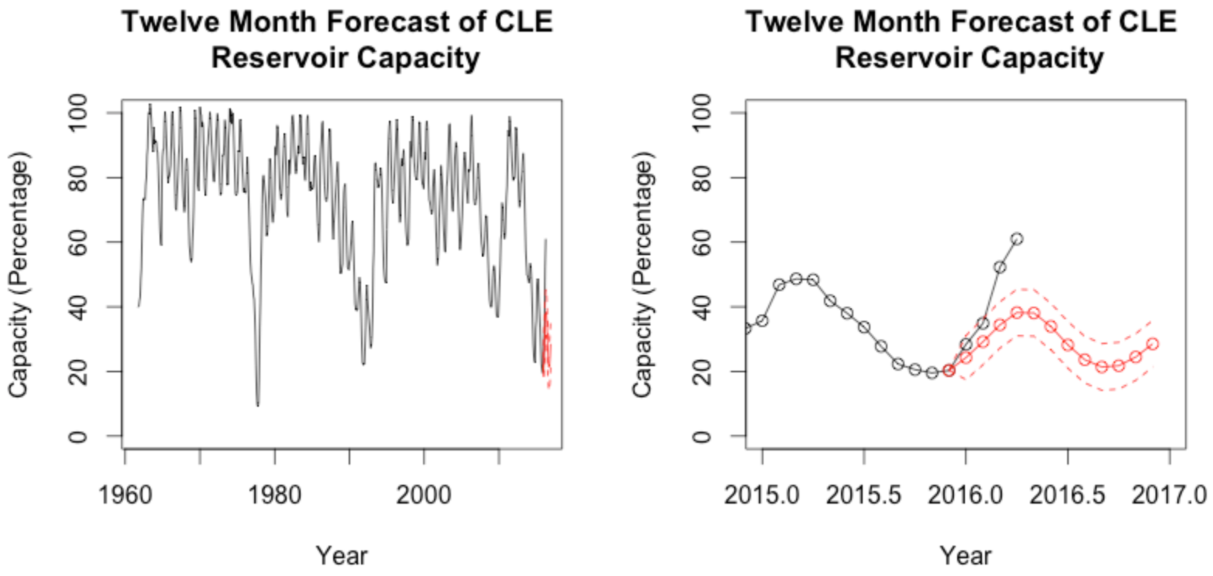


Figure 4. 2016 point forecast and 95% prediction interval for Trinity Lake Reservoir (CLE). On the left is the observed data in the black with the point forecast in solid red and the 95% prediction interval in dotted red lines. On the right is the observed data in black with the point forecast in solid red lines with month markers and the 95% prediction interval in dotted red lines, zoomed in for the time span of 2015-2017.

As seen in the graph rightmost, the forecast for 2016 is accurate through January and February, but for March and April, the forecast generally underestimates the amount of water in the Trinity Lake Reservoir. The predictions for January and February are also low, but are still within the upper bound of the 95% prediction interval. The exact numerical values for the 2016 forecast are presented along with the first four months of actual data below in **Table 1.**

Table 1. Numerical values for 2016 point forecast and 95% prediction interval for Trinity Lake Reservoir (CLE). Values reported as a percentage of total reservoir capacity.

Month	Lower 95%	Point Forecast	Upper 95%	Actual

January	17.47742	24.28285	31.08828	28.375421
February	22.05884	29.19314	36.32744	34.901395
March	27.22113	34.40041	41.57968	52.276224
April	31.02766	38.20834	45.38903	61.018405
May	30.89573	38.07831	45.26090	-
June	26.73980	33.93493	41.13006	-
July	20.99687	28.19806	35.39925	-
August	16.41748	23.62966	30.84184	-
September	14.17302	21.39087	28.60871	-
October	14.54228	21.76793	28.99359	-
November	17.24260	24.47274	31.70288	-
December	21.29829	28.53383	35.76937	-

4.1.5 Trinity Lake Reservoir (CLE) Forecast Comparison

To check the accuracy of the forecasting process for Trinity Lake Reservoir, the same modeling process was repeated for the 2014 year. The comparison between forecasts and 95% prediction interval is presented below in **Figure 5**.

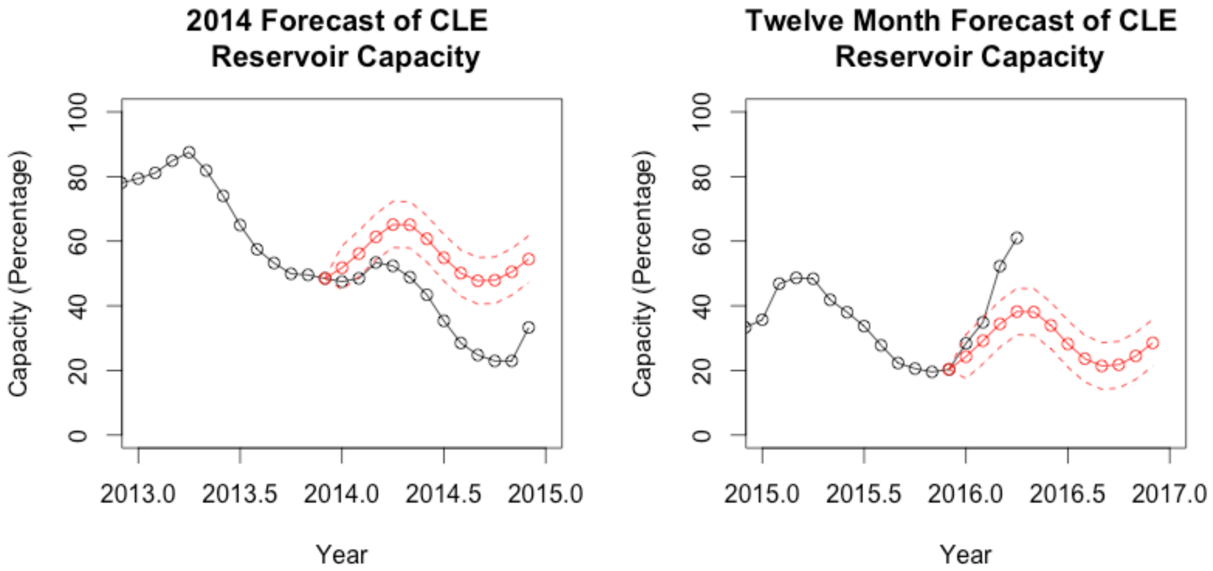


Figure 5. 2014 and 2016 point forecasts and prediction intervals for Trinity Lake. On the left is the 2014 forecast and on the right is the 2016 forecast. For both graphs, the observed data is given in black, the point forecast is given by the solid red line with month markers, and the prediction interval is given by the dotted red line.

In comparison to the forecasts in 2016, the 2014 forecasts seem to overestimate the amount of water in the Trinity Lake reservoir. January through March of 2014 seem to fall within the 95% prediction interval, but the rest of the year is below the lower 95% bound. Thus, the forecasting process does not seem to fit very well for predicting Trinity Lake Reservoir's water capacity levels in both

4.2 Lake Shasta Reservoir (SHA) Forecast Results

4.2.1 Lake Shasta Reservoir (SHA) Data Retrieval

Lake Shasta Reservoir (SHA) data was retrieved from the CDEC using sharpshootR for the date range of October 1953 to April 2016 at a monthly level, totaling 751 observations. The raw reservoir storage was then divided by Lake Shasta's capacity of 4,552,000 acre-feet of storage capacity to find the capacity percentage of every observation.

4.2.2 Lake Shasta Reservoir (SHA) Data Inspection

Upon initial inspection, the data for Lake Shasta (shown below in **Figure 6**) did not appear to require any additional transformation or cleaning.

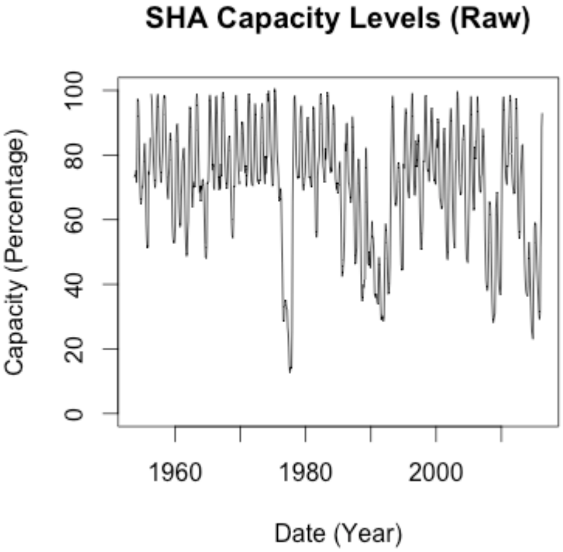


Figure 6. Raw data for Lake Shasta Reservoir (SHA).

After checking for any NA values, two were found, at indexes 31 and 196. The large number of observations in between the two NA values allowed for imputation to create values at those indexes.

Sharp drops in the late 1970s, early 1990s, and early 2010s were accounted for by statewide drought conditions during those times. As such, the observations are recorded as intended and are not incorrect measurements.

4.2.3 Lake Shasta Reservoir (SHA) Data Cleaning

No data cleaning was required for Lake Shasta capacity percentage observations. The cleaned data is presented below in **Figure 7**.

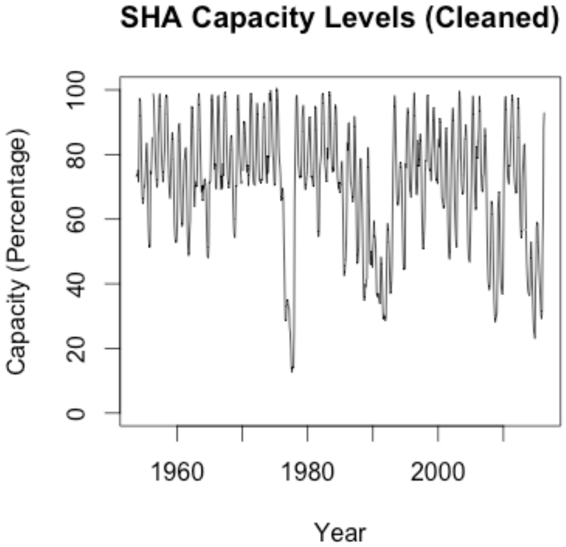


Figure 7. Cleaned data for Lake Shasta Reservoir (SHA).

4.2.4 Lake Shasta Reservoir (SHA) Forecasting

Using the forecast.all function, an ARMA(3,0) model was derived to fit the stationary data. The residuals are presented below in **Figure 8**.

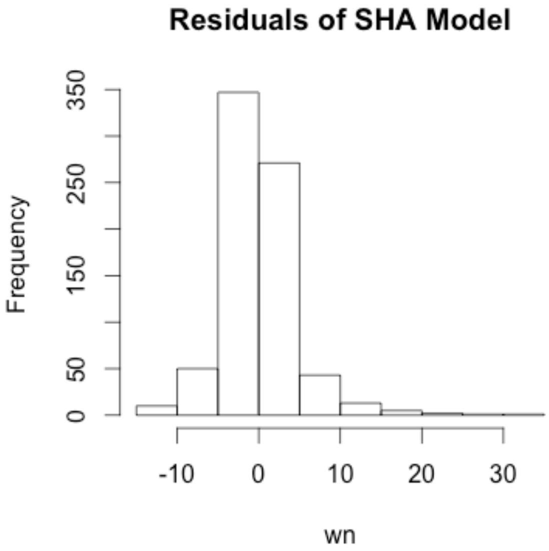


Figure 8. Histogram of residuals from the ARMA(3,0) model fitted to Lake Shasta Reservoir (SHA).

Based on the histogram, it appears that the residuals are mostly normal with a slightly right skew. According to the Shapiro-Wilk test, normality is rejected, but the histogram demonstrates that the residuals are approximately normal. The 95% prediction intervals presented in **Figure 9** below may not be accurate, but should be pretty close.

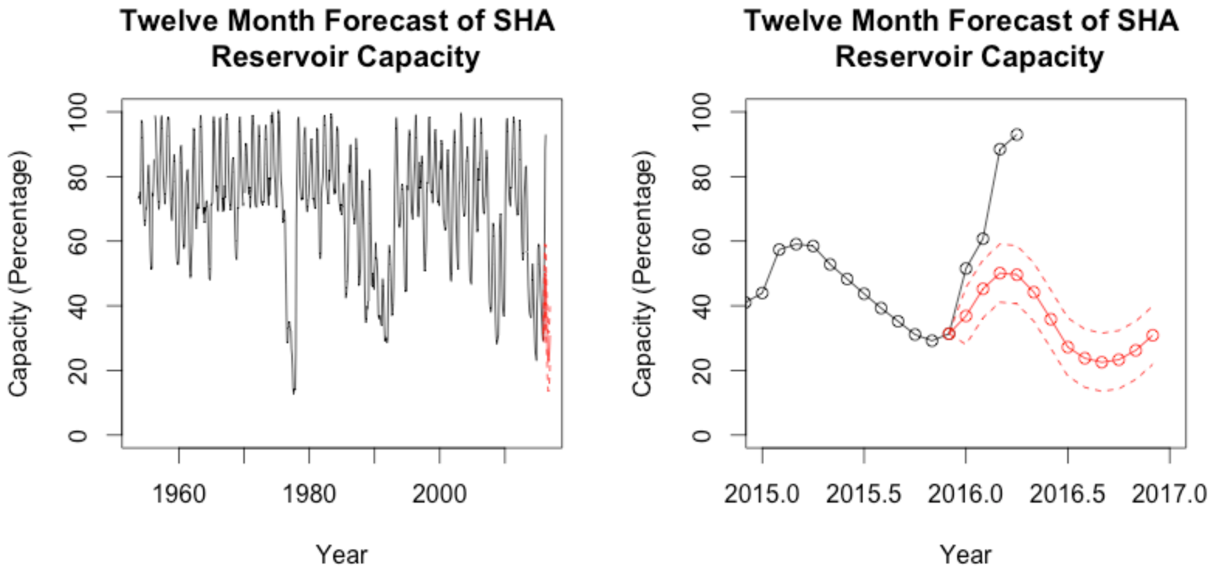


Figure 9. 2016 point forecast and 95% prediction interval for Lake Shasta Reservoir (SHA). On the left is the observed data in the black with the point forecast in solid red and the 95% prediction interval in dotted red lines. On the right is the observed data in black with the point forecast in solid red lines with month markers and the 95% prediction interval in dotted red lines, zoomed in for the time span of 2015-2017.

As seen in the rightmost graph above, the forecast for 2016 has so far not been accurate in predicting the capacity levels for Lake Shasta. The model seems to consistently underestimate capacity levels for Lake Shasta through April.

Table 2. Numerical values for 2016 point forecast and 95% prediction interval for Lake Shasta Reservoir (SHA). Values reported as a percentage of total reservoir capacity.

Month	Lower 95%	Point Forecast	Upper 95%	Actual
January	28.02210	36.86662	45.71115	51.53833
February	36.27259	45.24952	54.22646	60.76714
March	41.12166	50.10676	59.09186	88.45940
April	40.63212	49.62642	58.62072	92.99453
May	35.21288	44.20846	53.20403	-
June	26.84539	35.84112	44.83684	-
July	18.22264	27.21837	36.21411	-
August	14.78451	23.78025	32.77599	-
September	13.53565	22.53139	31.52713	-
October	14.33408	23.32982	32.32556	-
November	17.21553	26.21127	35.20701	-
December	21.88132	30.87706	39.87280	-

4.2.5 Lake Shasta Reservoir (SHA) Forecast Comparison

To check the accuracy of the forecasting process for Lake Shasta Reservoir, the same modeling process was repeated for the 2014 year. The comparison between forecasts and 95% prediction interval is presented below in **Figure 10**.

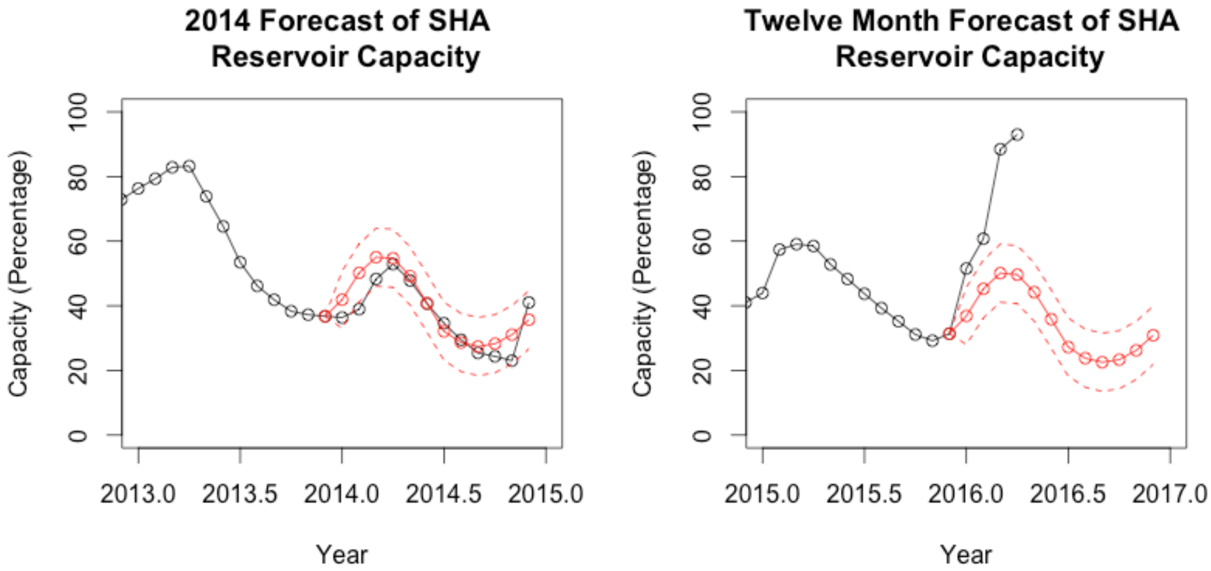


Figure 10. 2014 and 2016 point forecasts and prediction intervals for Lake Shasta (SHA). On the left is the 2014 forecast and on the right is the 2016 forecast. For both graphs, the observed data is given in black, the point forecast is given by the solid red line with month markers, and the prediction interval is given by the dotted red line.

In comparison to the forecasts in 2016, the 2014 forecasts seem much more accurate. Every observed value in 2014 falls within the 95% prediction interval or is almost exactly on the point forecast. Thus, the forecasting process seems to be more accurate in non-El Niño years than in El Niño years for Lake Shasta.

4.3 Lake Oroville Reservoir (ORO) Forecast Results

4.3.1 Lake Oroville Reservoir (ORO) Data Retrieval

Lake Oroville Reservoir (ORO) data was retrieved from the CDEC using sharpshootR for the date range of October 1967 to April 2016 at a monthly level, totaling 583 observations. The raw reservoir storage was then divided by Lake Oroville's capacity of 3,537,577 acre-feet of storage capacity to find the capacity percentage of every observation.

4.3.2 Lake Oroville Reservoir (ORO) Data Inspection

Upon initial inspection, the data for Lake Oroville (shown below in **Figure 11**) did not appear to require any additional transformation or cleaning.

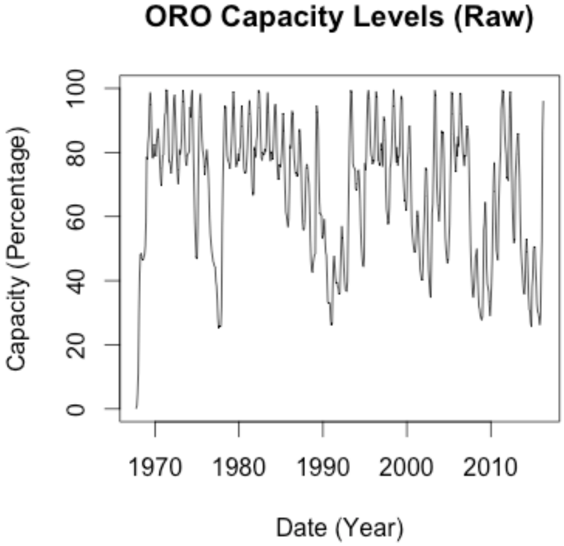


Figure 11. Raw data for Lake Oroville Reservoir (ORO).

After checking for any NA values, none were found, meaning that no imputation or data cutting was necessary.

Sharp drops in the late 1970s, early 1990s, and early 2010s were accounted for by statewide drought conditions during those times. As such, the observations are recorded as intended and are not incorrect measurements.

4.3.3 Lake Oroville Reservoir (ORO) Data Cleaning

No data cleaning was required for Lake Oroville capacity percentage observations. The cleaned data is presented below in **Figure 12**.

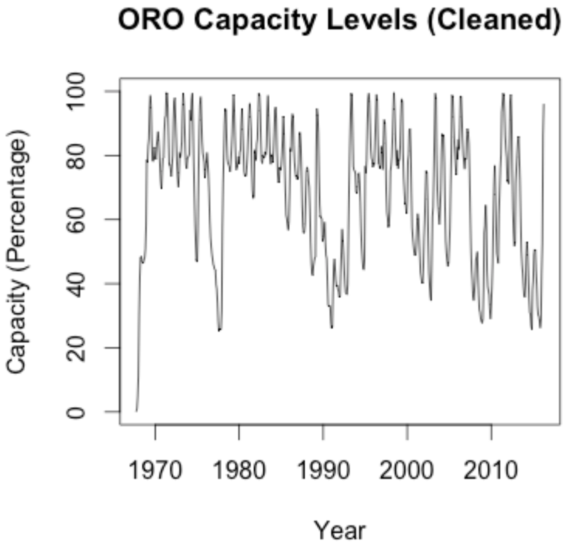


Figure 12. Cleaned data for Lake Oroville Reservoir (ORO).

4.3.4 Lake Oroville Reservoir (ORO) Forecasting

Using the forecast.all function, an ARMA(1,0) model was derived to fit the stationary data. The residuals are presented below in **Figure 13**.

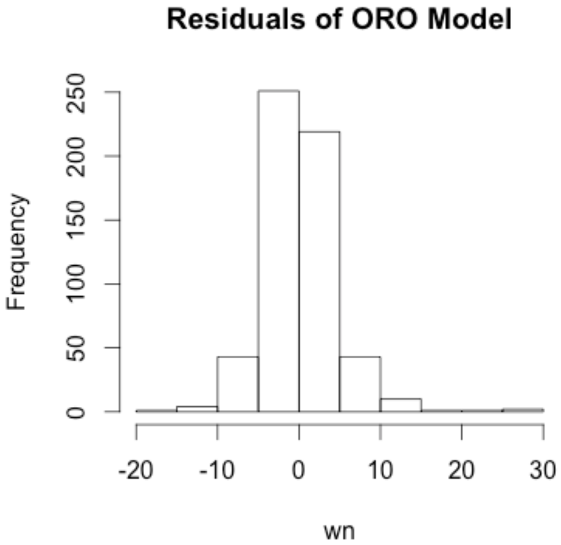


Figure 13. Histogram of residuals from the ARMA(1,0) model fitted to Lake Oroville Reservoir (ORO).

Based on the histogram, it appears that the residuals are mostly normal with a slightly right skew. According to the Shapiro-Wilk test, normality is rejected, but the histogram demonstrates that the residuals are approximately normal. The 95% prediction intervals presented in **Figure 14** below may not be accurate, but should be pretty close.

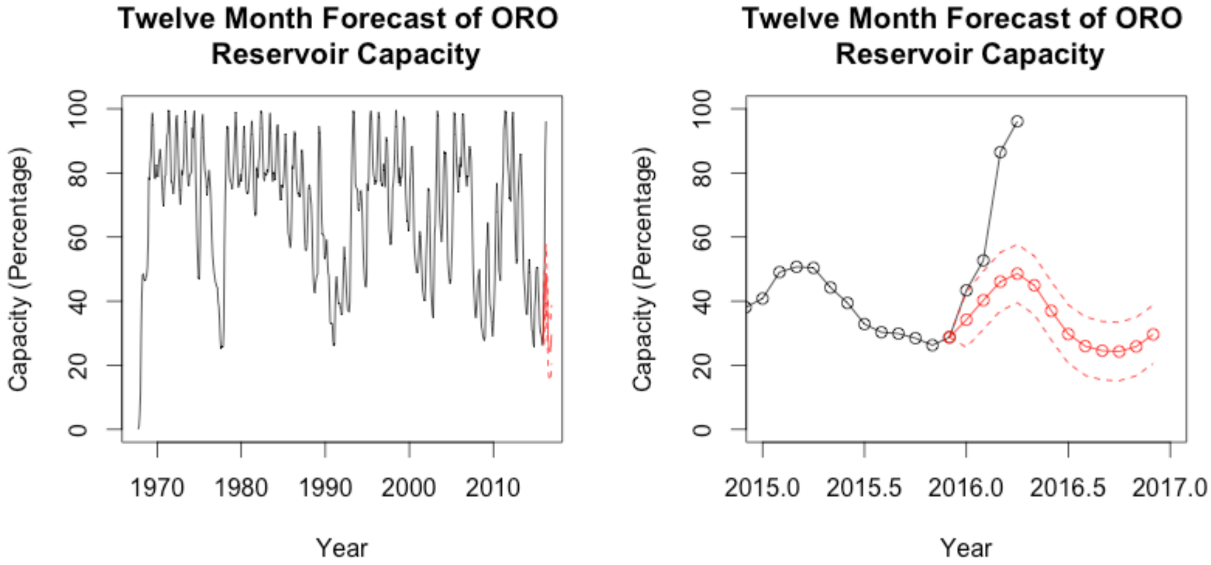


Figure 14. 2016 point forecast and 95% prediction interval for Lake Oroville Reservoir (ORO). On the left is the observed data in the black with the point forecast in solid red and the 95% prediction interval in dotted red lines. On the right is the observed data in black with the point forecast in solid red lines with month markers and the 95% prediction interval in dotted red lines, zoomed in for the time span of 2015-2017.

As seen in the rightmost graph above, the forecast for 2016 has so far not been very accurate in predicting the capacity levels for Lake Oroville. Through April, the model has consistently underestimated capacity levels, with only January's observation falling within the 95% prediction interval.

Table 3. Numerical values for 2016 point forecast and 95% prediction interval for Lake Oroville Reservoir (ORO). Values reported as a percentage of total reservoir capacity.

Month	Lower 95%	Point Forecast	Upper 95%	Actual
January	25.61116	34.21714	42.82313	43.36963
February	31.22582	40.28758	49.34934	52.71034
March	36.98575	46.09568	55.20562	86.49259

April	39.46869	48.58385	57.69901	96.10589
May	35.83631	44.95204	54.06777	-
June	27.85548	36.97127	46.08706	-
July	20.61901	29.73480	38.85060	-
August	16.85860	25.97440	35.09019	-
September	15.43127	24.54706	33.66286	-
October	15.16527	24.28107	33.39686	-
November	16.73845	25.85425	34.97005	-
December	20.52142	29.63721	38.75301	-

4.3.5 Lake Oroville Reservoir (ORO) Forecast Comparison

To check the accuracy of the model for Lake Oroville Reservoir, the same modeling process was repeated for the 2014 year. The comparison between forecasts and 95% prediction interval is presented below in **Figure 15**.

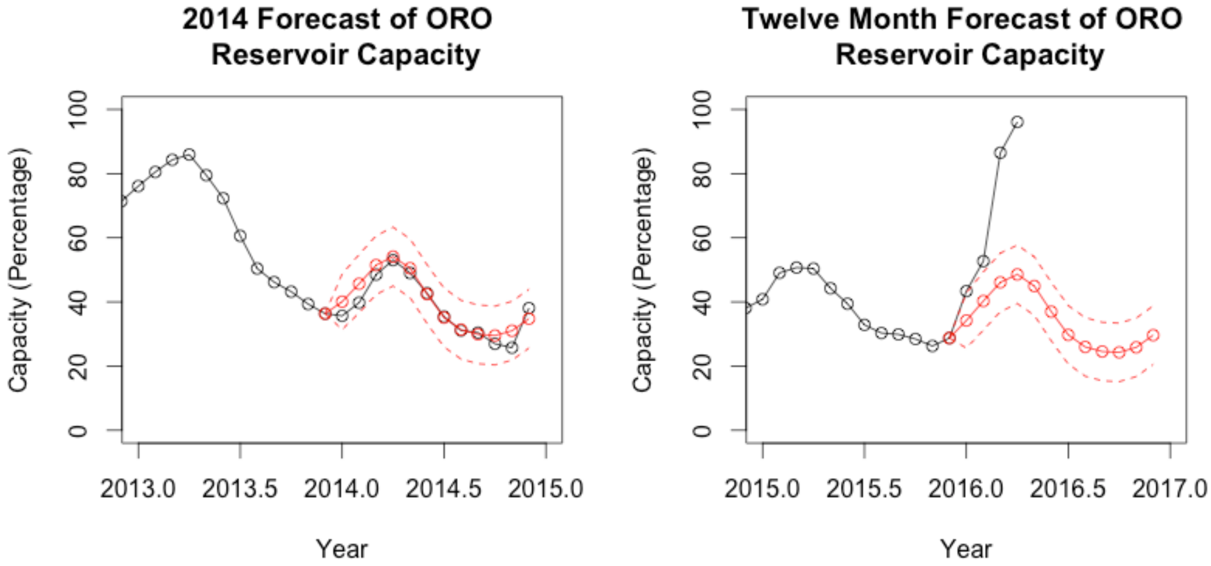


Figure 15. 2014 and 2016 point forecasts and prediction intervals for Lake Oroville (ORO). On the left is the 2014 forecast and on the right is the 2016 forecast. For both graphs, the observed data is given in black, the point forecast is given by the solid red line with month markers, and the prediction interval is given by the dotted red line.

In comparison to the forecasts in 2016, the 2014 forecasts appear to be much more accurate. Every observed value in 2014 falls within the 95% prediction interval with the predicted values with June, July, and August falling almost exactly on the point forecast. Thus, the forecasting process may be more accurate in non-El Niño years than in El Niño years for Lake Oroville.

4.4 New Melones Lake Reservoir (NML) Forecast Results

4.4.1 New Melones Lake Reservoir (NML) Data Retrieval

New Melones Lake Reservoir (NML) data was retrieved from the CDEC using sharpshootR for the date range of October 1975 to April 2016 at a monthly level, totaling 487 observations. The raw reservoir storage was then divided by New Melones Lake Reservoir's capacity of 2,400.000 acre-feet of storage capacity to find the capacity percentage of every observation.

4.4.2 New Melones Lake Reservoir (NML) Data Inspection

Upon initial inspection, the data for New Melones Lake (shown below in **Figure 16**) did not appear to require any additional transformation or cleaning.

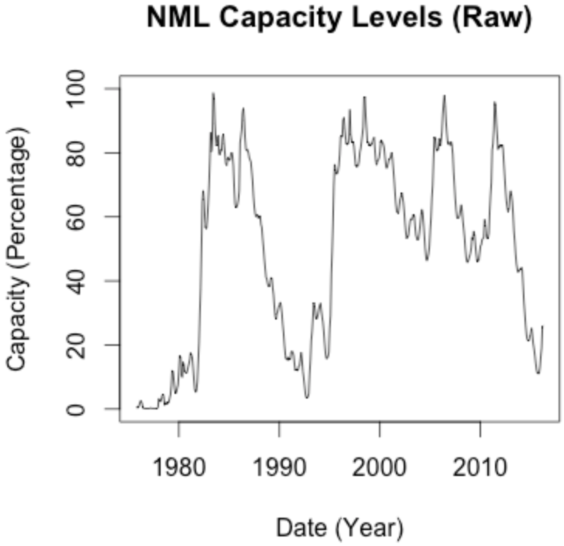


Figure 16. Raw data for Lake Oroville Reservoir (ORO).

After checking for any NA values, none were found, meaning that no imputation or data cutting was necessary.

The low initial values, drops in the early 1990s, and drops in the early 2010s were accounted for by statewide drought conditions during those times. As such, the observations are recorded as intended and are not incorrect measurements.

4.4.3 New Melones Lake Reservoir (NML) Data Cleaning

No data cleaning was required for Lake Oroville capacity percentage observations. The cleaned data is presented below in **Figure 17**.

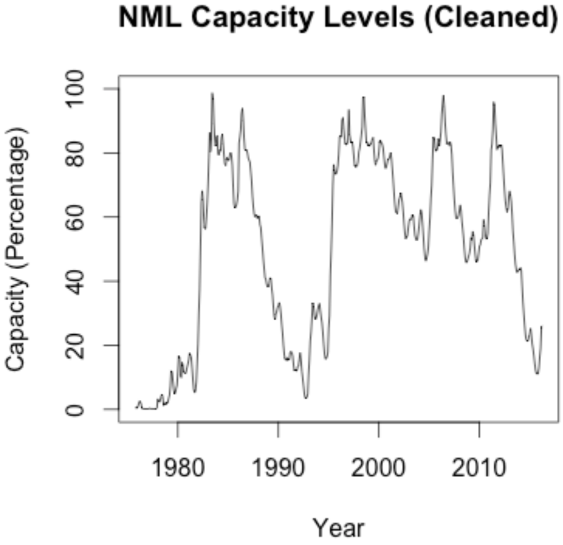


Figure 17. Cleaned data for New Melones Lake Reservoir (NML).

4.4.4 Lake Oroville Reservoir (ORO) Forecasting

Using the forecast.all function, an ARMA(1,3) model was derived to fit the stationary data. The residuals are presented below in **Figure 18**.

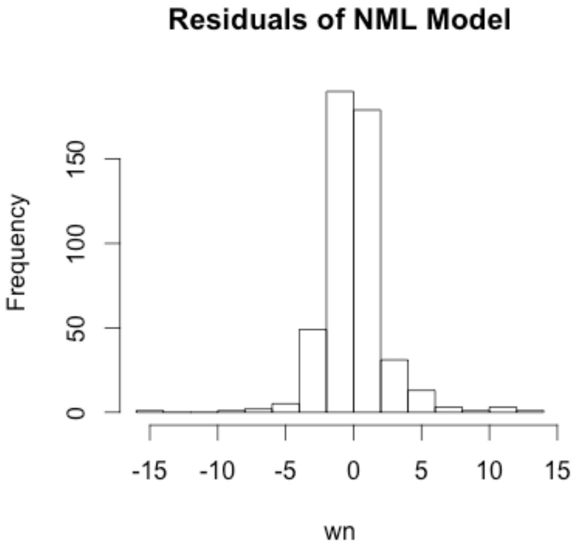


Figure 18. Histogram of residuals from the ARMA(1,3) model fitted to New Melones Lake Reservoir (NML).

Based on the histogram, it appears that the residuals are mostly normal with a slightly right skew. According to the Shapiro-Wilk test, normality is rejected, but the histogram demonstrates that the residuals are approximately normal. The 95% prediction intervals presented in **Figure 19** below may not be accurate, but should be pretty close.

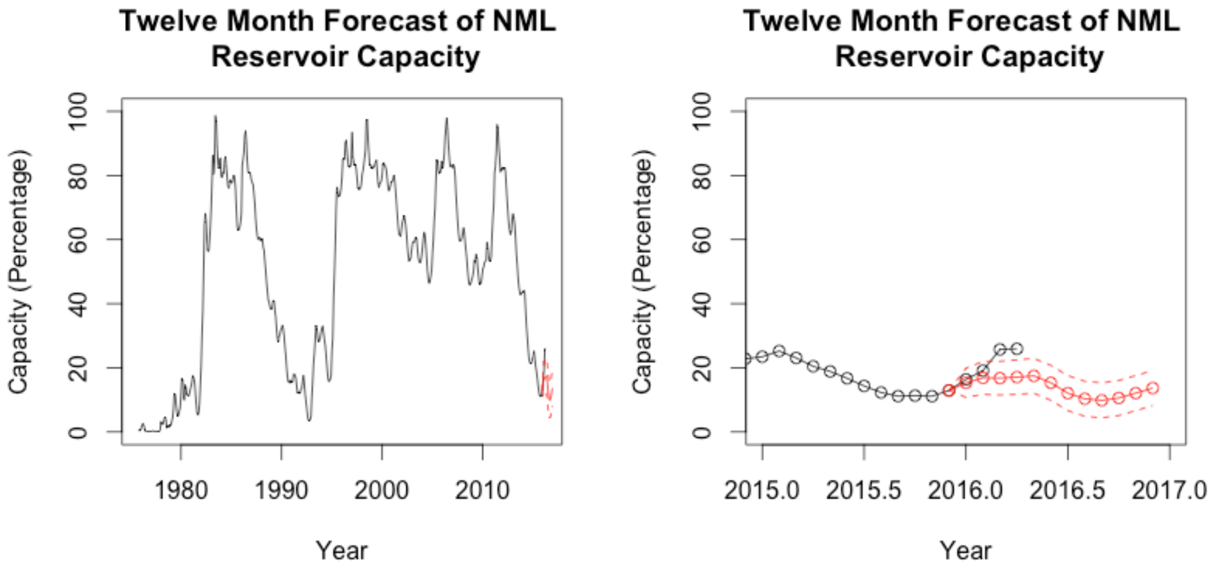


Figure 19. 2016 point forecast and 95% prediction interval for New Melones Lake Reservoir (NML). On the left is the observed data in the black with the point forecast in solid red and the 95% prediction interval in dotted red lines. On the right is the observed data in black with the point forecast in solid red lines with month markers and the 95% prediction interval in dotted red lines, zoomed in for the time span of 2015-2017.

As seen in the rightmost graph above, the forecast for 2016 was fairly accurate for January and February, but has underestimated capacity levels for March and April. January's observation fell almost exactly on the point forecast, while February's observation was slightly above the point forecast, continuing the trend of the models consistently underestimating capacity levels.

Table 4. Numerical values for 2016 point forecast and 95% prediction interval for New Melones Lake (NML). Values reported as a percentage of total reservoir capacity.

Month	Lower 95%	Point Forecast	Upper 95%	Actual
January	10.791137	15.316986	13.670318	16.37167
February	11.702429	16.843336	21.98424	19.11267
March	11.508911	16.767981	22.02705	25.72267
April	11.706703	17.081779	22.45686	25.92083
May	12.060120	17.485482	22.91084	-
June	9.868835	15.316189	20.76354	-
July	6.631219	12.088226	17.54523	-
August	4.869878	10.331131	15.79238	-
September	4.381035	9.844155	15.30728	-
October	5.111567	10.575509	16.03945	-
November	6.636709	12.101014	17.56532	-
December	8.205855	13.670318	19.13478	-

4.4.5 New Melones Lake Reservoir (NML) Forecast Comparison

To check the accuracy of the model for New Melones Lake Reservoir, the same modeling process was repeated for the 2014 year. The comparison between forecasts and 95% prediction interval is presented below in **Figure 20**.

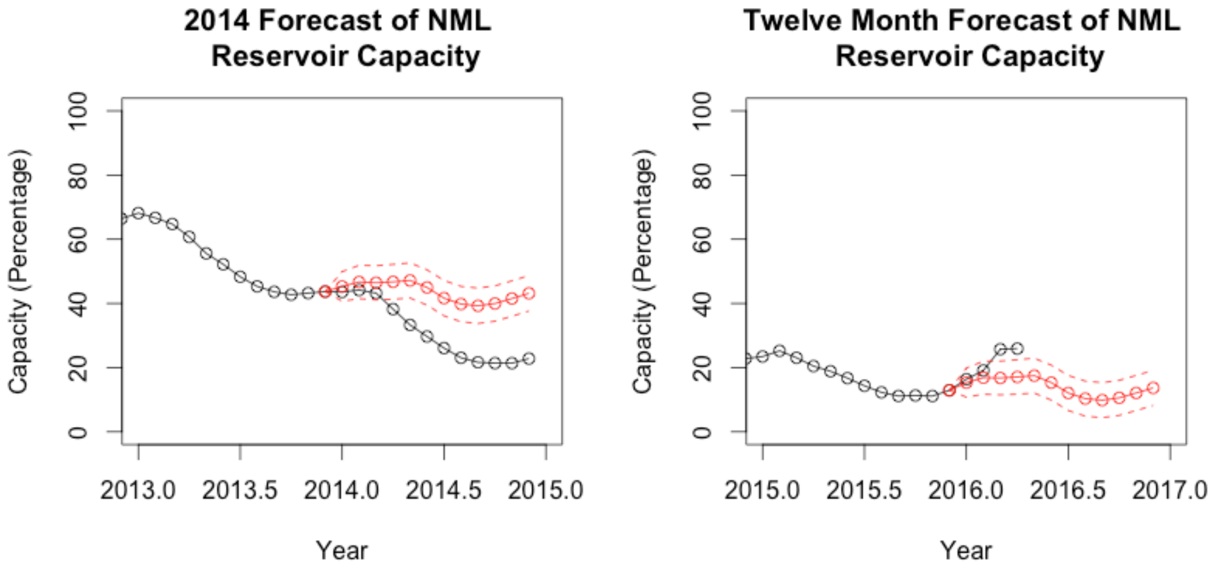


Figure 20. 2014 and 2016 point forecasts and prediction intervals for New Melones Lake Reservoir (NML). On the left is the 2014 forecast and on the right is the 2016 forecast. For both graphs, the observed data is given in black, the point forecast is given by the solid red line with month markers, and the prediction interval is given by the dotted red line.

In comparison to the forecasts in 2016, the 2014 forecasts surprisingly overestimate the true percentage capacity levels for the New Melones Lake Reservoir. Although January, February, and March fall within the 95% prediction interval, the rest of the year does not follow the forecast. Thus, the forecasting may not be the best fit for New Melones Lake Reservoir.

4.5 Folsom Lake Reservoir (FOL) Forecast Results

4.5.1 Folsom Lake Reservoir (FOL) Data Retrieval

Folsom Lake Reservoir (FOL) data was retrieved from the CDEC using sharpshootR for the date range of October 1955 to April 2016 at a monthly level, totaling 727 observations. The raw reservoir storage was then divided by Folsom Lake Reservoir's capacity of 977,000 acre-feet of storage capacity to find the capacity percentage of every observation.

4.5.2 Folsom Lake Reservoir (FOL) Data Inspection

Upon initial inspection, the data for Folsom Lake (shown below in **Figure 21**) did not appear to require any additional transformation or cleaning.

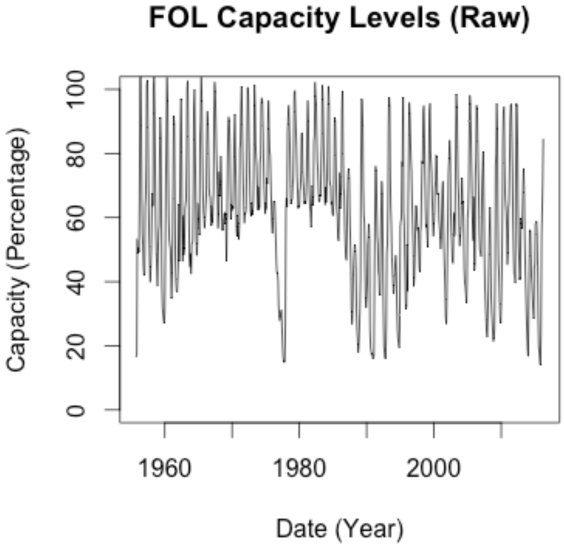


Figure 21. Raw data for Folsom Lake Reservoir (FOL).

After checking for any NA values, none were found, meaning that no imputation or data cutting was necessary.

The only major unexpected drop occurs in the late 1970s, which was a result of the statewide drought conditions during that time. As such, the observations are recorded as intended and are not incorrect measurements.

4.5.3 Folsom Lake Reservoir (FOL) Data Cleaning

No data cleaning was required for Folsom Lake capacity percentage observations. The cleaned data is presented below in **Figure 22**.

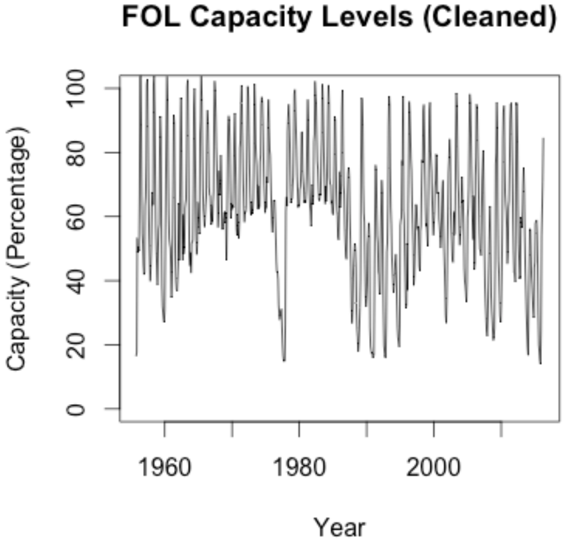


Figure 22. Cleaned data for Folsom Lake Reservoir (FOL).

4.5.4 Folsom Lake Reservoir (FOL) Forecasting

Using the forecast.all function, an ARMA(1,3) model was derived to fit the stationary data. The residuals are presented below in **Figure 23**.

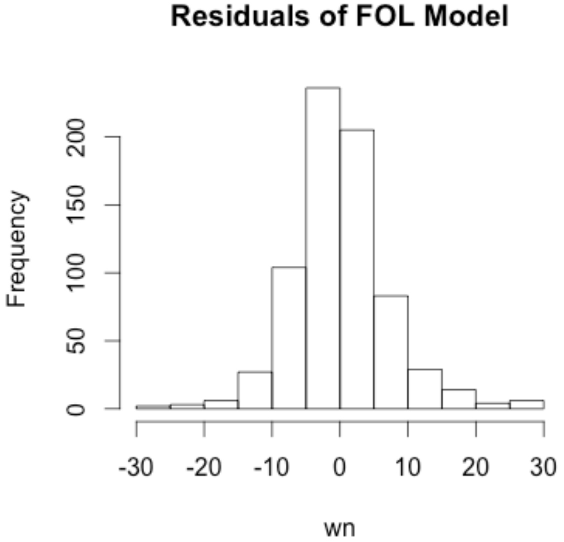


Figure 23. Histogram of residuals from the ARMA(1,3) model fitted to Folsom Lake Reservoir (FOL).

Based on the histogram, it appears that the residuals are mostly normal with a slightly right skew. According to the Shapiro-Wilk test, normality is rejected, but the histogram demonstrates that the residuals are approximately normal. The 95% prediction intervals presented in **Figure 24** below may not be accurate, but should be pretty close.

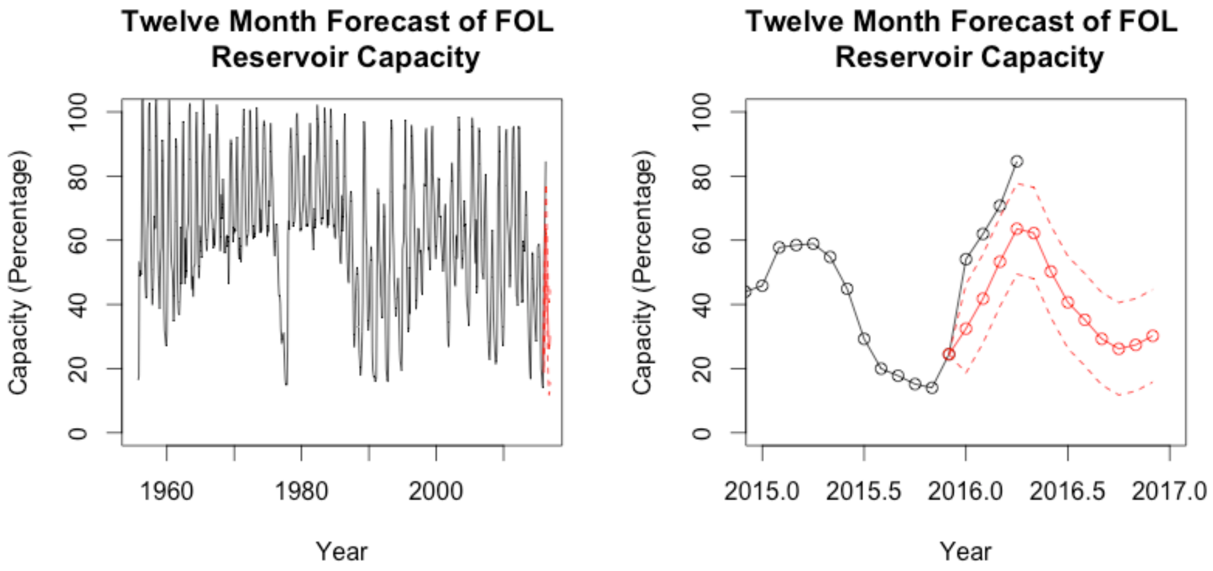


Figure 24. 2016 point forecast and 95% prediction interval for Folsom Lake Reservoir (NML). On the left is the observed data in the black with the point forecast in solid red and the 95% prediction interval in dotted red lines. On the right is the observed data in black with the point forecast in solid red lines with month markers and the 95% prediction interval in dotted red lines, zoomed in for the time span of 2015-2017.

As seen in the rightmost graph above, the forecast for Folsom Lake underestimated capacity percentage levels for January through April of 2016. Not a single observation from 2016 even falls within the 95% prediction interval.

Table 5. Numerical values for 2016 point forecast and 95% prediction interval for Folsom Lake Reservoir (FOL). Values reported as a percentage of total reservoir capacity.

Month	Lower 95%	Point Forecast	Upper 95%	Actual
January	18.63760	32.44900	46.26040	54.10061
February	27.94453	41.79150	55.63848	61.98106
March	39.37734	53.31724	67.25714	70.76469

April	49.44489	63.57261	77.70034	84.59048
May	47.98741	62.22871	76.47002	-
June	35.95113	50.26141	64.57168	-
July	26.30936	40.66163	55.01390	-
August	20.83794	35.21581	49.59369	-
September	14.86996	29.26346	43.65697	-
October	11.78026	26.18330	40.58635	-
November	12.98064	27.38951	41.79839	-
December	15.78775	30.20018	44.61262	-

4.5.5 Folsom Lake Reservoir (FOL) Forecast Comparison

To check the accuracy of the model for Folsom Lake Reservoir, the same modeling process was repeated for the 2014 year. The comparison between forecasts and 95% prediction interval is presented below in **Figure 25**.

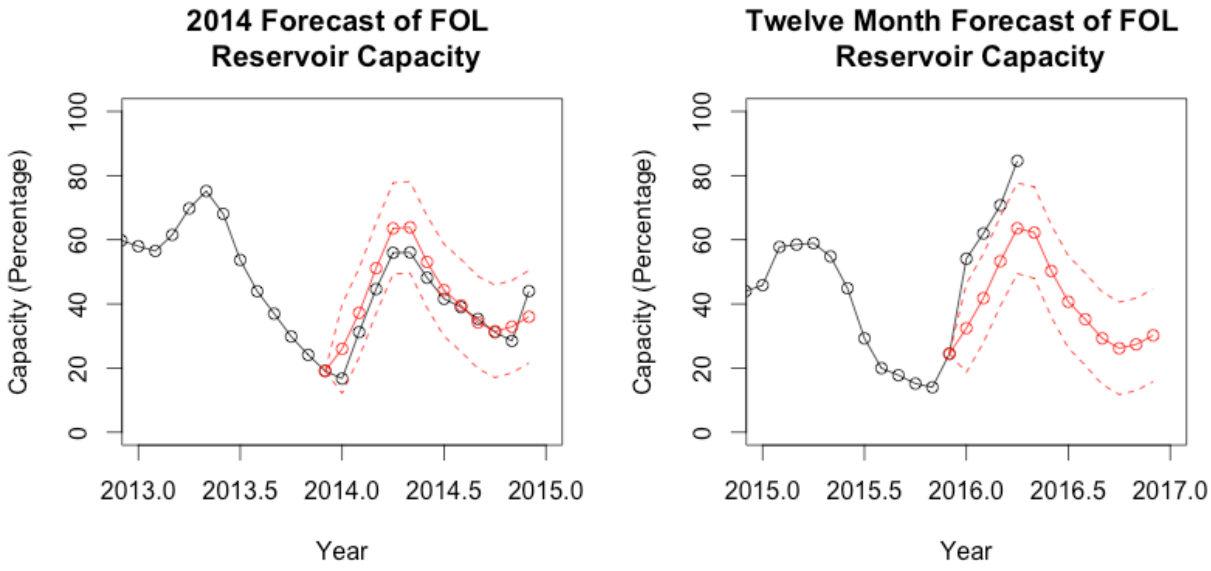


Figure 25. 2014 and 2016 point forecasts and prediction intervals for Folsom Lake Reservoir (FOL). On the left is the 2014 forecast and on the right is the 2016 forecast. For both graphs, the observed data is given in black, the point forecast is given by the solid red line with month markers, and the prediction interval is given by the dotted red line.

In comparison to the forecasts in 2016, the 2014 forecasts fit the observed values very well. All of the observed percentage capacity readings fall within the 95%

prediction interval with July through October's observations falling particularly close to the point forecasts for those respective months. Thus, the forecasting process appears to create well-fitting predictions in non-El Niño years, but not in El Niño years.

4.6 Don Pedro Reservoir (DNP) Forecast Results

4.6.1 Don Pedro Reservoir (DNP) Data Retrieval

Don Pedro Reservoir (DNP) data was retrieved from the CDEC using sharpshootR for the date range of October 1970 to April 2016 at a monthly level, totaling 547 observations. The raw reservoir storage was then divided by Don Pedro Reservoir's capacity of 2,030,000 acre-feet of storage capacity to find the capacity percentage of every observation.

4.6.2 Don Pedro Reservoir (DNP) Data Inspection

Upon initial inspection, the data for Don Pedro (shown below in **Figure 26**) did not appear to require any additional transformation or cleaning.

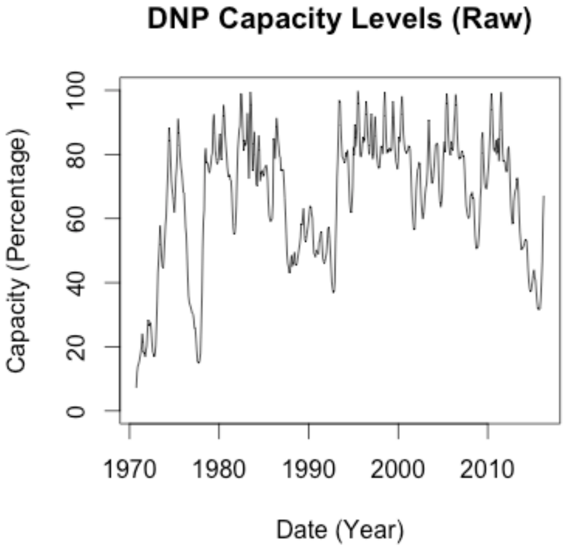


Figure 26. Raw data for Don Pedro Reservoir (DNP).

After checking for any NA values, none were found, meaning that no imputation or data cutting was necessary.

The first major drop occurring in the late 1970s coincides with statewide drought conditions. The next drop also coincides with the statewide drought which took place from 1986-1992. The sharp increase in capacity percentage immediately following the low values in 1992 is associated with the 1992-1993 El Niño. Finally, the drop in the early 2010s is representative of current California drought conditions. As such, the observations are recorded as intended and are not incorrect measurements.

4.6.3 Don Pedro Reservoir (DNP) Data Cleaning

No data cleaning was required for Don Pedro capacity percentage observations. The cleaned data is presented below in **Figure 27**.

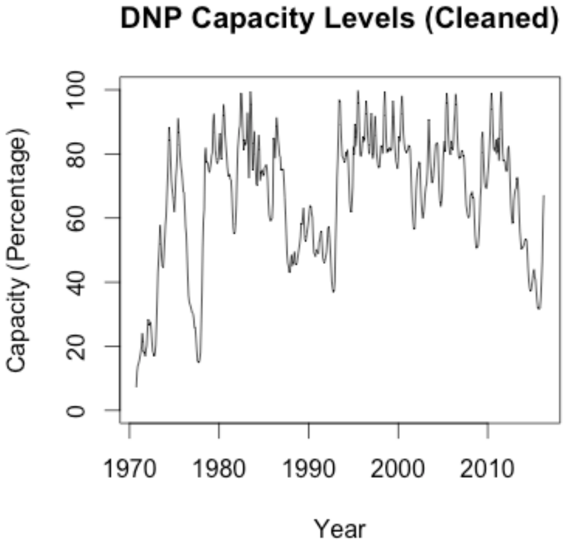


Figure 27. Cleaned data for Don Pedro Reservoir (DNP).

4.6.4 Don Pedro Reservoir (DNP) Forecasting

Using the forecast.all function, an ARMA(0,5) model was derived to fit the stationary data. The residuals are presented below in **Figure 28**.

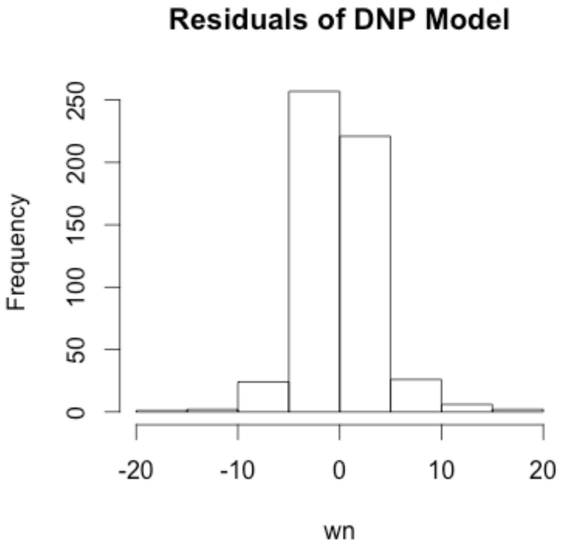


Figure 28. Histogram of residuals from the ARMA(0,5) model fitted to Don Pedro Reservoir (DNP).

Based on the histogram, it appears that the residuals are mostly normal with a slightly right skew. According to the Shapiro-Wilk test, normality is rejected, but the histogram demonstrates that the residuals are approximately normal. The 95% prediction intervals presented in **Figure 29** below may not be accurate, but should be pretty close.

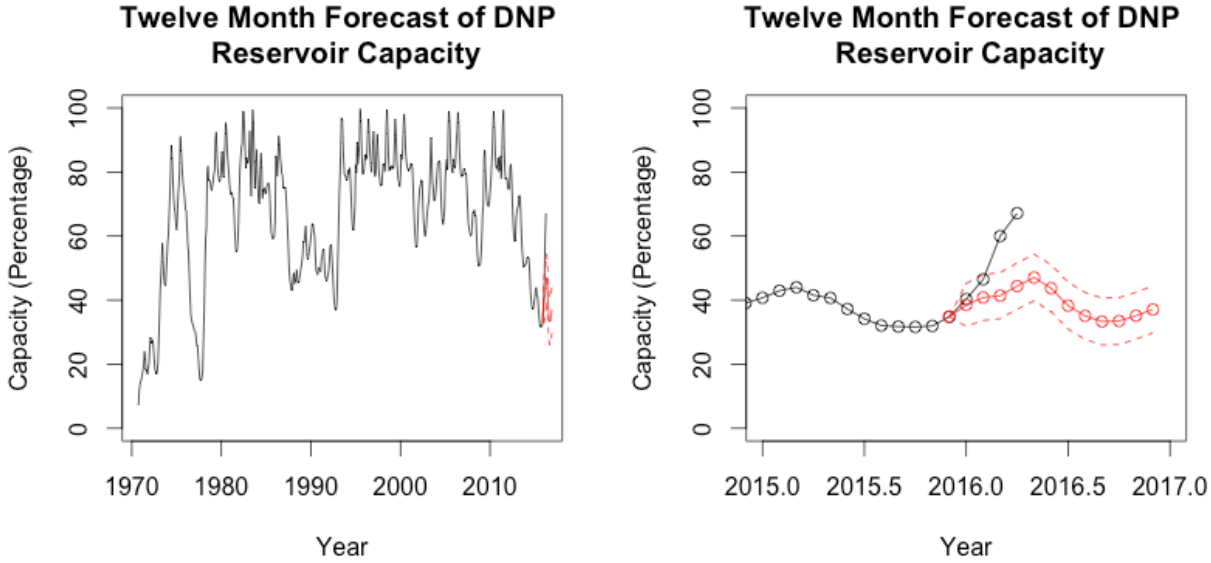


Figure 29. 2016 point forecast and 95% prediction interval for Don Pedro Reservoir (DNP). On the left is the observed data in the black with the point forecast in solid red and the 95% prediction interval in dotted red lines. On the right is the observed data in black with the point forecast in solid red lines with month markers and the 95% prediction interval in dotted red lines, zoomed in for the time span of 2015-2017.

As seen in the rightmost graph above, the forecast for Don Pedro underestimated capacity percentage levels for January through April of 2016. While January and February fell within the upper 95% prediction bound, March and April far exceeded the expectations of the forecast.

Table 6. Numerical values for 2016 point forecast and 95% prediction interval for Don Pedro Reservoir (DNP). Values reported as a percentage of total reservoir capacity.

Month	Lower 95%	Point Forecast	Upper 95%	Actual
January	31.81845	38.51913	45.21982	40.30926
February	33.63496	40.81791	48.00086	46.52000
March	34.20197	41.39491	48.58785	60.01458
April	37.20773	44.40217	51.59661	67.15256
May	39.81259	47.06435	54.31611	-
June	36.40955	43.72681	51.04407	-
July	30.96909	38.28635	45.60361	-
August	27.76446	35.08172	42.39898	-
September	26.01840	33.33566	40.65292	-
October	26.16421	33.48147	40.79873	-
November	27.82187	35.13913	42.45639	-
December	29.77831	37.09557	44.41283	-

4.6.5 Don Pedro Reservoir (DNP) Forecast Comparison

To check the accuracy of the model for Don Pedro Reservoir, the same modeling process was repeated for the 2014 year. The comparison between forecasts and 95% prediction interval is presented below in **Figure 30**.

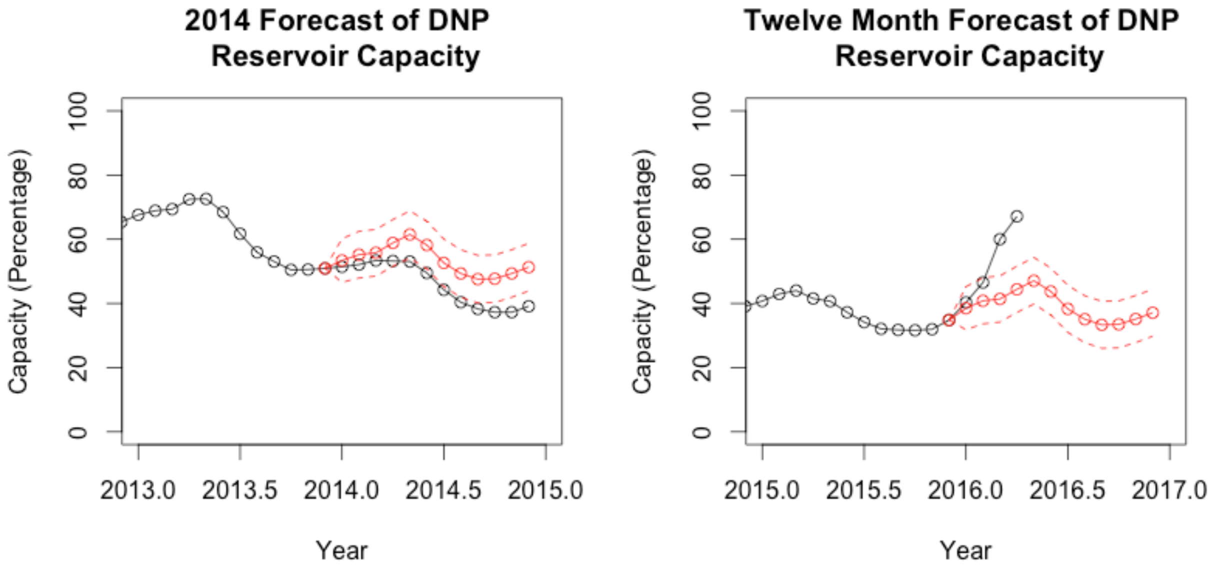


Figure 30. 2014 and 2016 point forecasts and prediction intervals for Don Pedro Reservoir (DNP). On the left is the 2014 forecast and on the right is the 2016 forecast. For both graphs, the observed data is given in black, the point forecast is given by the solid red line with month markers, and the prediction interval is given by the dotted red line.

In comparison to the forecasts in 2016, the 2014 forecast was a relatively better fit to the observed values. While only January through March of 2014 stayed relatively close the point forecast, January through April stayed within the 95% prediction interval. The rest of the year was within a roughly 5% range of the edge of the prediction interval. In general, the forecasting process overestimated percentage capacity values in 2014, but overall seemed to be a better fit, at least through April. The fit of the forecasting process seems inconclusive.

4.7 San Luis Reservoir (SNL) Forecast Results

4.7.1 San Luis Reservoir (SNL) Data Retrieval

San Luis Reservoir (SNL) data was retrieved from the CDEC using sharpshootR for the date range of October 1968 to April 2016 at a monthly level, totaling 571 observations. The raw reservoir storage was then divided by San Luis Reservoir's capacity of 2,041,000 acre-feet of storage capacity to find the capacity percentage of every observation.

4.7.2 San Luis Reservoir (SNL) Data Inspection

Upon initial inspection, the data for San Luis (shown below in **Figure 31**) did not appear to require any additional transformation or cleaning.

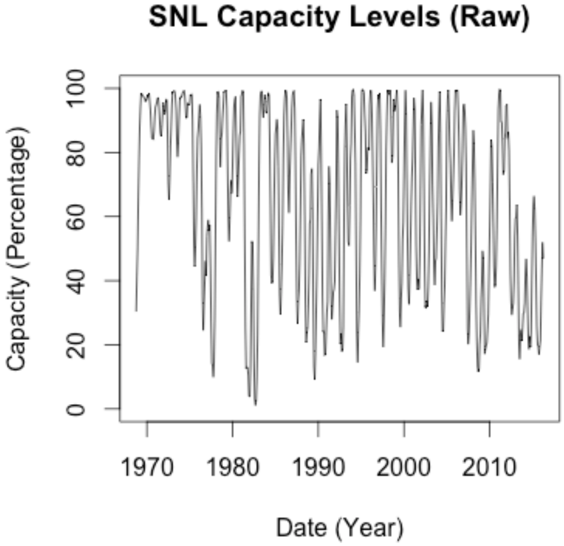


Figure 31. Raw data for San Luis Reservoir (SNL).

After checking for any NA values, none were found, meaning that no imputation or data cutting was necessary.

The historical levels of the San Luis Reservoir seem to have a relatively high variance, but the values make sense in the context of major California water events. As such, the observations are recorded as intended and are not incorrect measurements.

4.7.3 San Luis Reservoir (SNL) Data Cleaning

No data cleaning was required for San Luis capacity percentage observations. The cleaned data is presented below in **Figure 32**.

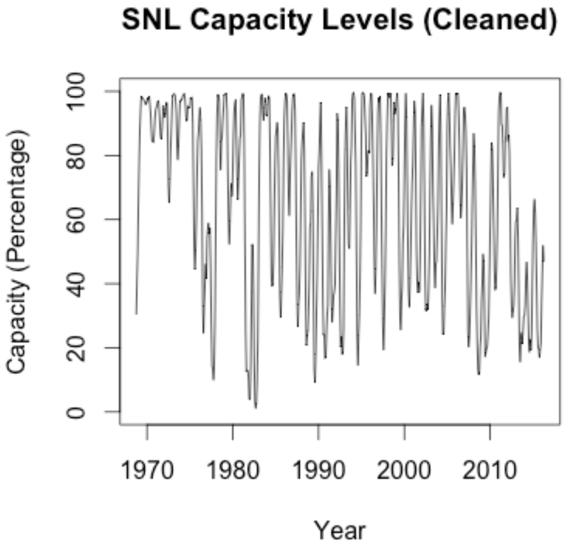


Figure 32. Cleaned data for San Luis Reservoir (SNL).

4.7.4 San Luis Reservoir (SNL) Forecasting

Using the forecast.all function, an ARMA(1,3) model was derived to fit the stationary data. The residuals are presented below in **Figure 33**.

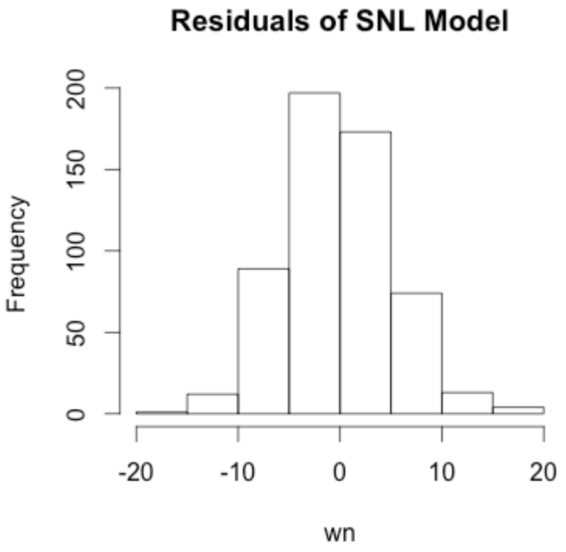


Figure 33. Histogram of residuals from the ARMA(1,3) model fitted to San Luis Reservoir (SNL).

Based on the histogram, it appears that the residuals are mostly normal with a slightly right skew. According to the Shapiro-Wilk test, normality is rejected, but the histogram demonstrates that the residuals are approximately normal. The 95% prediction intervals presented in **Figure 34** below may not be accurate, but should be pretty close.

Note: The residuals from the ARMA(1,3) were not found to be independent, according to the Ljung-Box test. The following results may not be accurate.

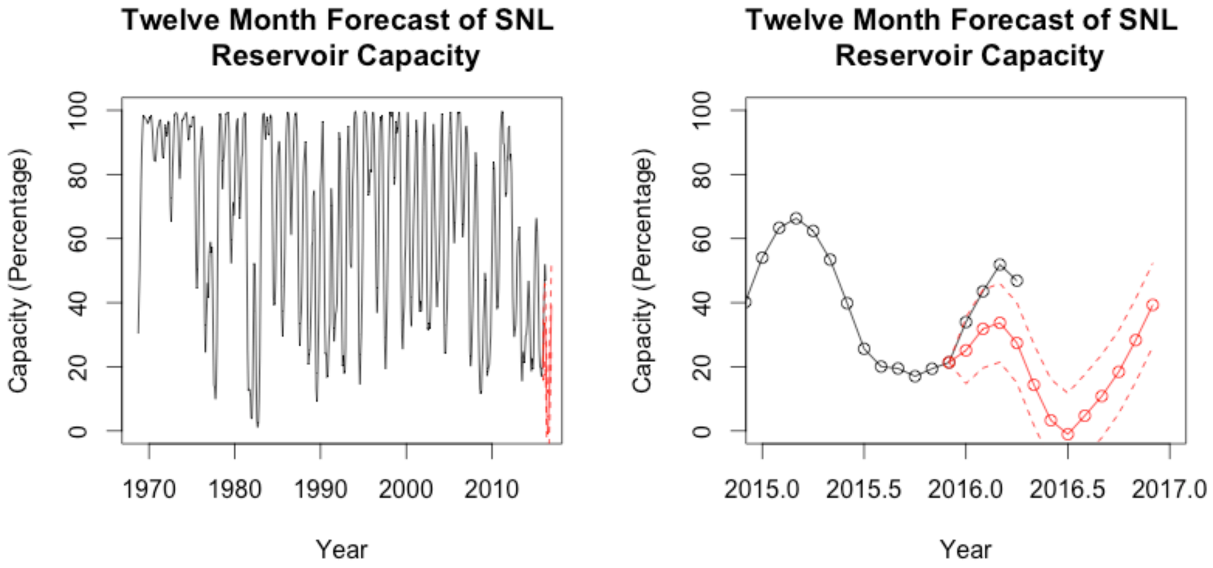


Figure 34. 2016 point forecast and 95% prediction interval for San Luis Reservoir (SNL). On the left is the observed data in the black with the point forecast in solid red and the 95% prediction interval in dotted red lines. On the right is the observed data in black with the point forecast in solid red lines with month markers and the 95% prediction interval in dotted red lines, zoomed in for the time span of 2015-2017.

As seen in the rightmost graph above, the forecast for San Luis Reservoir pretty consistently underestimated percentage capacity levels through April of 2016. While January and February were right around the very edge of the upper 95% prediction bound, March and April were beyond the bounds of the forecast, likely due to the unexpected increases rain which accompanies an El Niño year.

An interesting note for this model is that one value of the point forecast dipped below the 0% point. Further investigation is needed to determine why the predicted point forecast value for July manifested as such.

Table 7. Numerical values for 2016 point forecast and 95% prediction interval for San Luis Reservoir (SNL). Values reported as a percentage of total reservoir capacity.

Month	Lower 95%	Point Forecast	Upper 95%	Actual
January	14.771423	25.112359	35.45330	33.95502
February	19.625931	31.834724	44.04352	43.53038
March	21.438043	33.720847	46.00365	51.87011
April	14.951787	27.441896	39.93200	46.87139
May	1.750393	14.391697	27.03300	-
June	-9.449474	3.302547	16.05457	-
July	-13.877799	-1.044474	11.78885	-
August	-8.159881	4.733271	17.62642	-
September	-2.089157	10.848082	23.78532	-
October	5.451733	18.421496	31.39126	-
November	15.321399	28.315175	41.30895	-
December	26.296266	39.307780	52.31929	-

4.7.5 San Luis Reservoir (SNL) Forecast Comparison

To check the accuracy of the model for San Luis Reservoir, the same modeling process was repeated for the 2014 year. The comparison between forecasts and 95% prediction interval is presented below in **Figure 35**.

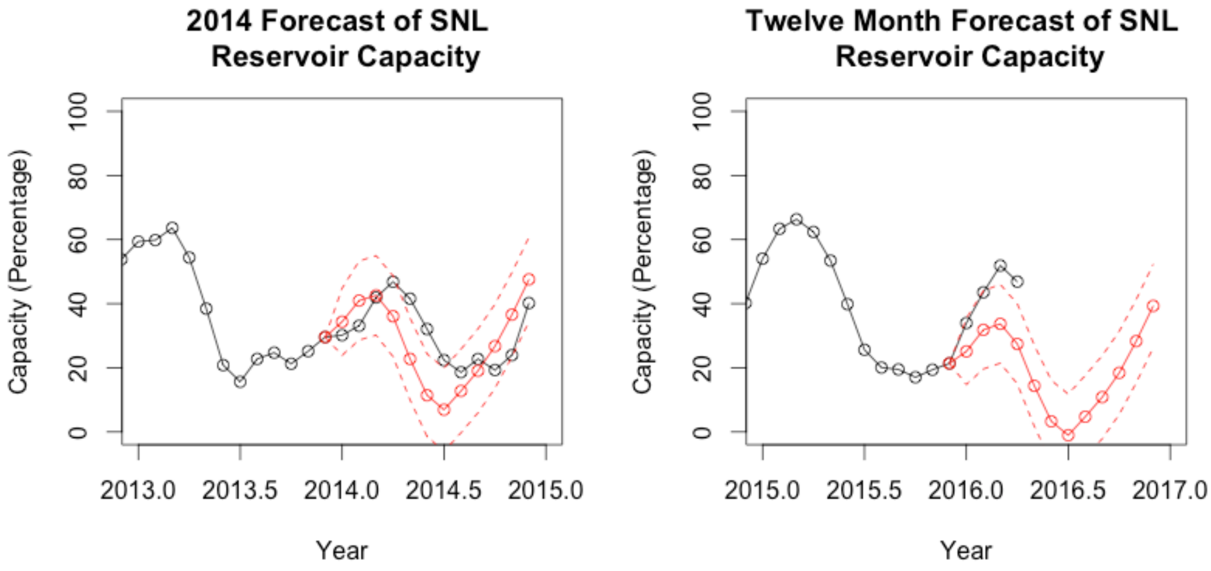


Figure 35. 2014 and 2016 point forecasts and prediction intervals for San Luis Reservoir (SNL). On the left is the 2014 forecast and on the right is the 2016 forecast. For both graphs, the observed data is given in black, the point forecast is given by the solid red line with month markers, and the prediction interval is given by the dotted red line.

The 2014 forecast seemed to be relatively closer than the 2016 forecast. In 2014, only three observations (May, June, and July) fell outside of the 95% prediction bounds, with March falling almost exactly in line with the point forecast. It is of note that were the 2014 forecast to be shifted ahead by one month, the shape of the forecast would more closely align with the observed values. Thus, the forecasting process seems to predict better in non-El Niño years for San Luis Reservoir.

4.8 Millerton Lake Reservoir (MIL) Forecast Results

4.8.1 Millerton Lake Reservoir (MIL) Data Retrieval

Millerton Lake Reservoir (MIL) data was retrieved from the CDEC using sharpshootR for the date range of October 1941 to April 2016 at a monthly level, totaling 895 observations. The raw reservoir storage was then divided by Millerton Lake Reservoir's capacity of 520,500 acre-feet of storage capacity to find the capacity percentage of every observation.

4.8.2 Millerton Lake Reservoir (MIL) Data Inspection

Upon initial inspection, the data for Millerton (shown below in **Figure 36**) did not appear to require any additional transformation or cleaning.

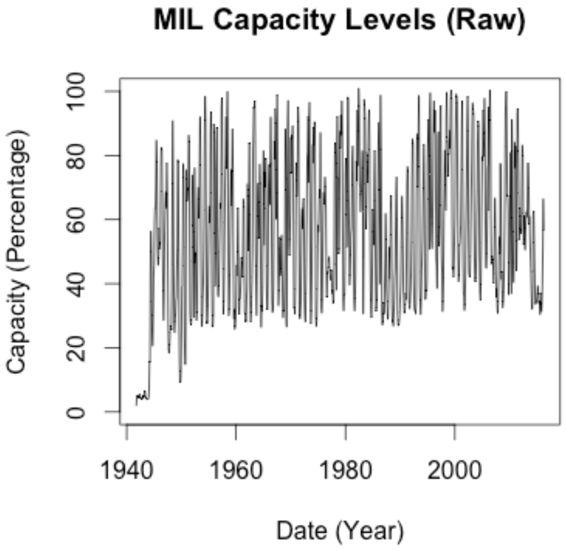


Figure 36. Raw data for Millerton Lake Reservoir (MIL).

After checking for any NA values, none were found, meaning that no imputation or data cutting was necessary.

The historical levels of the Millerton Lake Reservoir have a fairly large, but stable variance, meaning no transformation seems to be required. The sudden drop around 2006 is a result of an agreed upon draining of some of the reservoir capacity. It continues into a slightly larger decrease in capacity percentage in conjunction with the severe drought conditions of the early 2010s in California. As such, the observations are recorded as intended and are not incorrect measurements.

4.8.3 Millerton Lake Reservoir (MIL) Data Cleaning

No data cleaning was required for San Luis capacity percentage observations. The cleaned data is presented below in **Figure 37**.

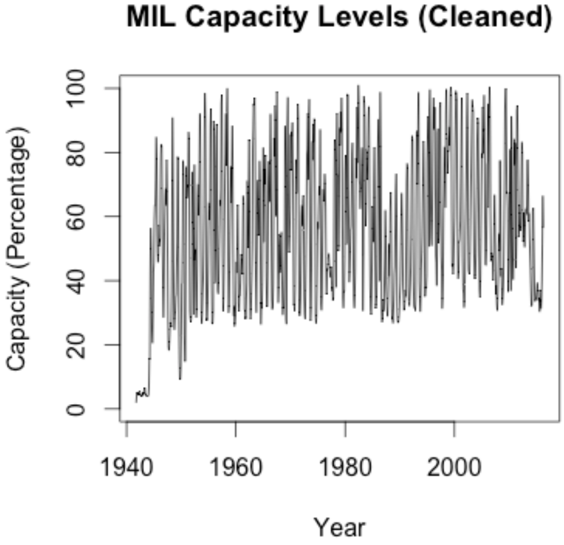


Figure 37. Cleaned data for Millerton Lake Reservoir (MIL).

4.8.4 Millerton Lake Reservoir (MIL) Forecasting

Using the forecast.all function, an ARMA(0,4) model was derived to fit the stationary data. The residuals are presented below in **Figure 38**.

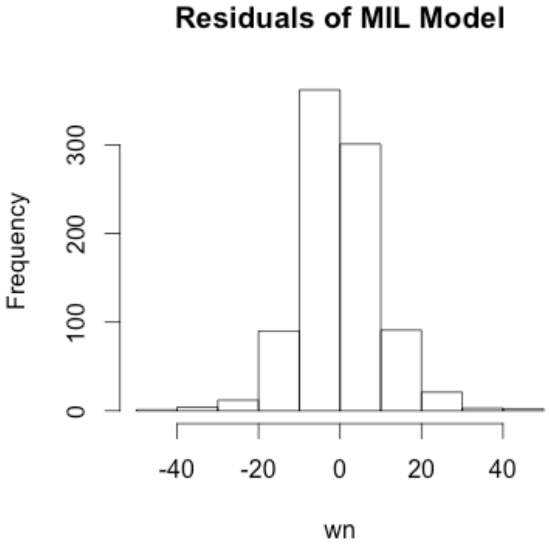


Figure 38. Histogram of residuals from the ARMA(0,4) model fitted to Millerton Lake Reservoir (MIL).

Based on the histogram, it appears that the residuals are mostly normal with a slightly right skew. According to the Shapiro-Wilk test, normality is rejected, but the histogram demonstrates that the residuals are approximately normal. The 95% prediction intervals presented in **Figure 39** below may not be accurate, but should be pretty close.

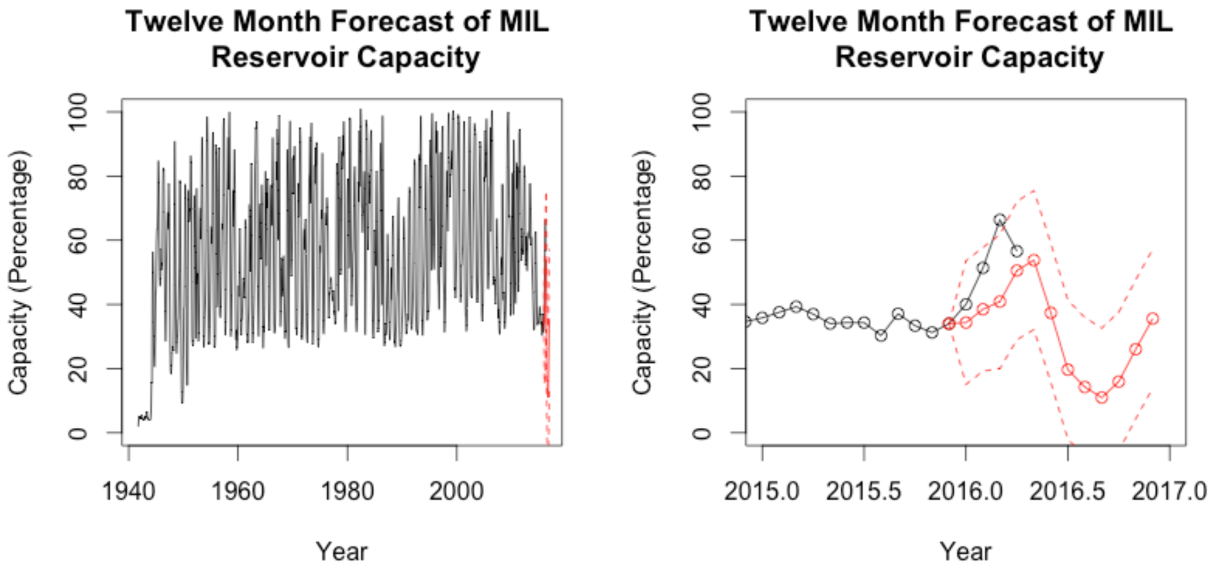


Figure 39. 2016 point forecast and 95% prediction interval for Millerton Lake Reservoir (MIL). On the left is the observed data in the black with the point forecast in solid red and the 95% prediction interval in dotted red lines. On the right is the observed data in black with the point forecast in solid red lines with month markers and the 95% prediction interval in dotted red lines, zoomed in for the time span of 2015-2017.

According to the rightmost graph, the forecast for 2016 has been relatively accurate, as the observed values January, February, and April all fall within the 95% prediction interval. Despite being outside of the prediction interval, March's observed value is extremely close to the edge of the prediction interval, so the model appears to be relatively accurate compared to some of the other reservoirs.

It is of note that the prediction interval for Millerton Lake is quite wide in comparison to many of the other reservoirs' forecasts. The forecast band is, on average, about a 40% spread between the upper and lower bounds. More investigation may be required to understand why the band is so large, but an initial guess would be that the large, constant variance of historical observations would be a major contributing factor for an especially wide prediction interval.

Table 8. Numerical values for 2016 point forecast and 95% prediction interval for Millerton Lake Reservoir (MIL). Values reported as a percentage of total reservoir capacity.

Month	Lower 95%	Point Forecast	Upper 95%	Actual
January	15.084098	34.34467	53.60523	40.02747
February	19.209684	38.49120	57.77271	51.39885
March	20.043931	40.87801	61.71208	66.37675
April	28.989956	50.55998	72.13000	56.59135
May	32.149083	53.80023	75.45138	-
June	15.720135	37.37128	59.02243	-
July	-1.964849	19.68630	41.33745	-
August	-7.394815	14.25633	35.90748	-
September	-10.681035	10.97011	32.62126	-
October	-5.701302	15.94985	37.60099	-
November	4.379901	26.03105	47.68220	-
December	13.930136	35.58128	57.23243	-

4.8.5 Millerton Lake Reservoir (MIL) Forecast Comparison

To check the accuracy of the model for Millerton Lake Reservoir, the same modeling process was repeated for the 2014 year. The comparison between forecasts and 95% prediction interval is presented below in **Figure 40**.

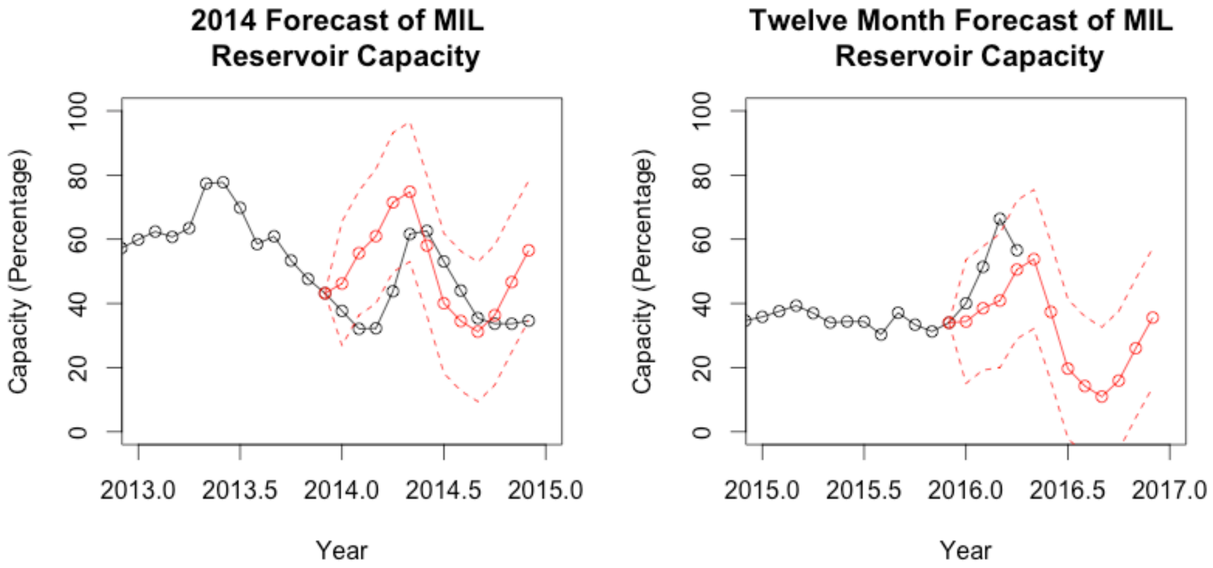


Figure 40. 2014 and 2016 point forecasts and prediction intervals for Millerton Lake Reservoir (MIL). On the left is the 2014 forecast and on the right is the 2016 forecast. For both graphs, the observed data is given in black, the point forecast is given by the solid red line with month markers, and the prediction interval is given by the dotted red line.

Through April, the 2016 forecast for Millerton Lake Reservoir seems to do better, as three of the four observations fall within the 95% prediction interval, as opposed to only one observation falling in the prediction interval for 2014. However, the rest of 2014 falls within the prediction interval, so it remains to be seen if the 2016 forecast will be as accurate as the 2014 forecast. The results thus far are inconclusive, but the accuracy seems fairly high for non-El Niño years.

4.9 Pine Flat Lake Reservoir (PNF) Forecast Results

4.9.1 Pine Flat Lake Reservoir (PNF) Data Retrieval

Pine Flat Lake Reservoir (PNF) data was retrieved from the CDEC using sharpshootR for the date range of December 1951 to April 2016 at a monthly level, totaling 773 observations. The raw reservoir storage was then divided by Pine Flat Lake Reservoir's capacity of 1,000,000 acre-feet of storage capacity to find the capacity percentage of every observation.

4.9.2 Pine Flat Lake Reservoir (PNF) Data Inspection

Upon initial inspection, the data for Pine Flat Lake (shown below in **Figure 41**) did not appear to require any additional transformation or cleaning.

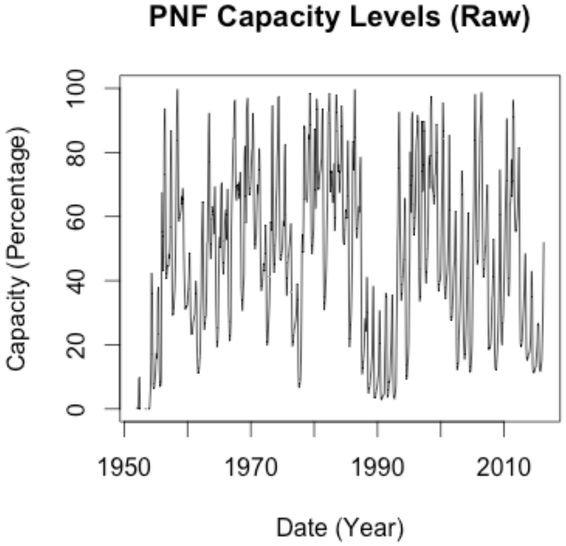


Figure 41. Raw data for Pine Flat Lake Reservoir (PNF).

However, after checking for any NA values, 10 values were found, meaning that some data manipulation was necessary to eliminate them.

The historical levels of the Pine Flat Lake Reservoir have a fairly large, but stable variance, meaning no transformation seems to be required. As such, the observations are recorded as intended and are not incorrect measurements.

4.9.3 Pine Flat Lake Reservoir (PNF) Data Cleaning

To clean the data, the NA values had to be addressed. Finding the index of the NA values, the first 9 NA values were contained within the first 22 monthly observations, while the last one was at index 322. To correct for this, the first 22 values were dropped from the dataset and the last value was handled through imputation. The cleaned data is presented below in **Figure 42**.

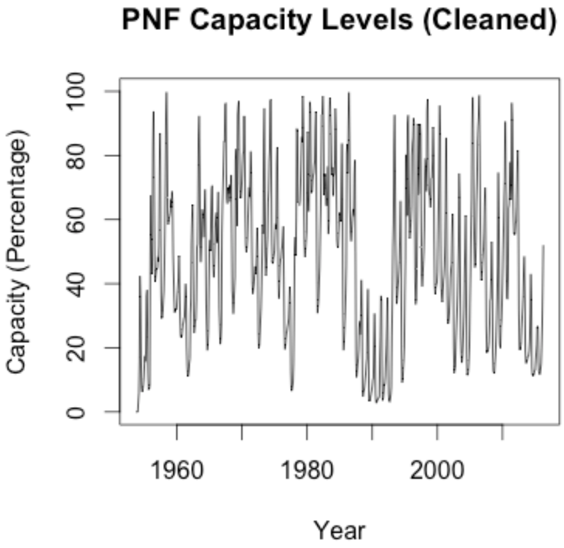


Figure 42. Cleaned data for Pine Flat Lake Reservoir (PNF).

As described above, the first 22 values have been left off of the time series and the NA value at index 322 was assigned a value by imputing the average of the preceding and following two months around the value.

4.9.4 Pine Flat Lake Reservoir (PNF) Forecasting

Using the forecast.all function, an ARMA(2,0) model was derived to fit the stationary data. The residuals are presented below in **Figure 43**.

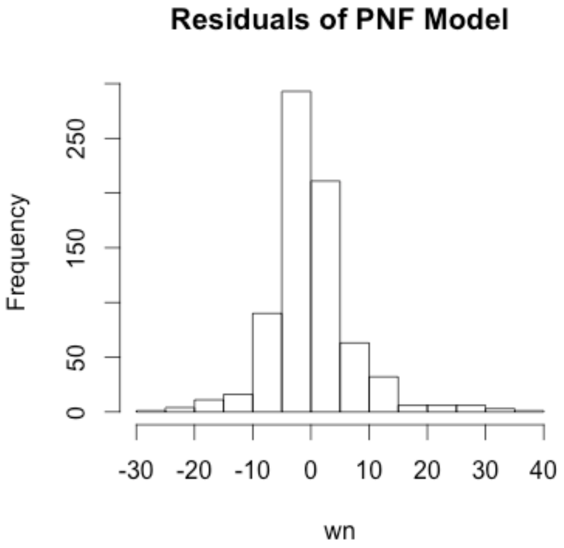


Figure 43. Histogram of residuals from the ARMA(2,0) model fitted to Pine Flat Lake Reservoir (PNF).

Based on the histogram, it appears that the residuals are mostly normal with a slightly right skew. According to the Shapiro-Wilk test, normality is rejected, but the histogram demonstrates that the residuals are approximately normal. The 95% prediction intervals presented in **Figure 44** below may not be accurate, but should be pretty close.

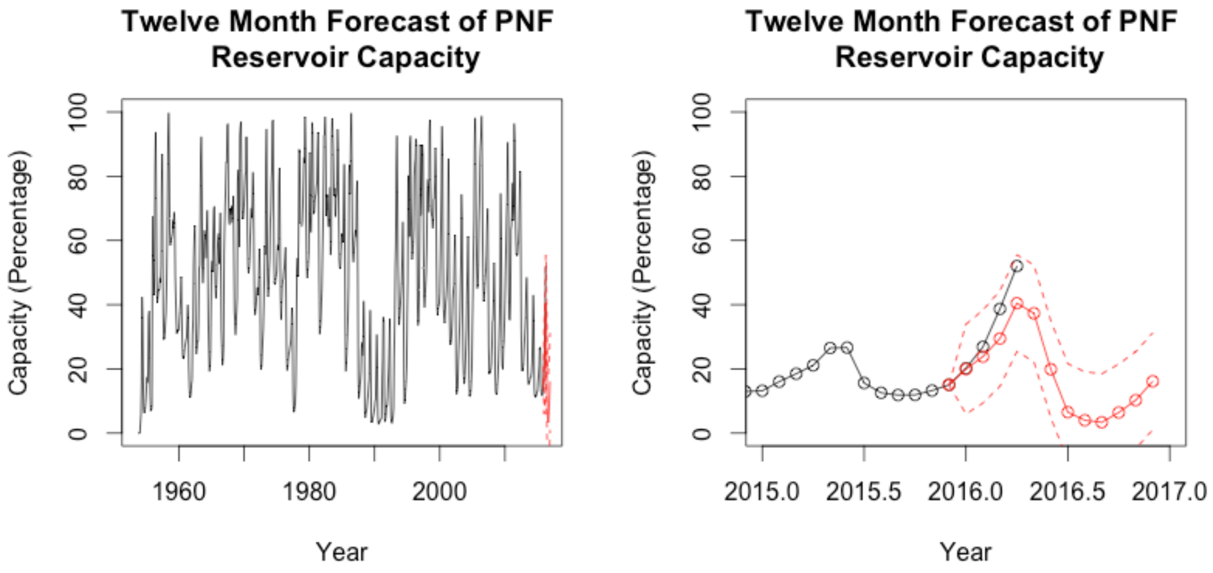


Figure 44. 2016 point forecast and 95% prediction interval for Pine Flat Lake Reservoir (PNF). On the left is the observed data in the black with the point forecast in solid red and the 95% prediction interval in dotted red lines. On the right is the observed data in black with the point forecast in solid red lines with month markers and the 95% prediction interval in dotted red lines, zoomed in for the time span of 2015-2017.

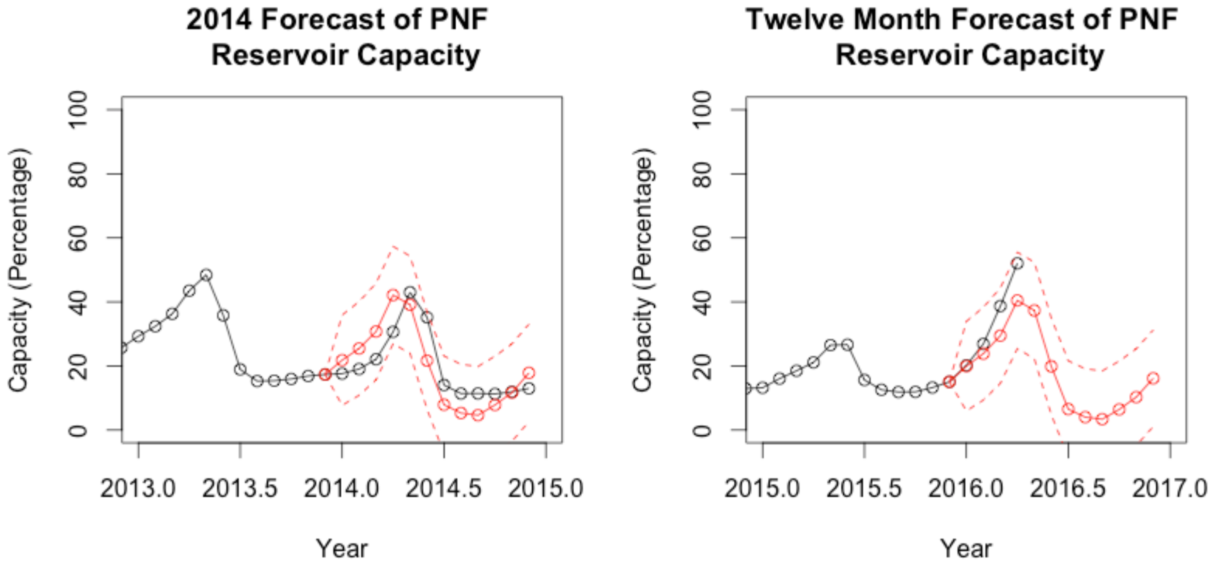
According to the rightmost graph, the 2016 forecast for Pine Flat Lake capacity percentage has been fairly accurate. The observation for January fell almost exactly on the point forecast and the observations for February through April all fell within the 95% prediction interval. Relative to most other reservoirs, the forecasts have been much more accurate, although the point forecast continues to underestimate capacity percentage, a trend among other reservoirs.

Table 9. Numerical values for 2016 point forecast and 95% prediction interval for Pine Flat Lake Reservoir (PNF). Values reported as a percentage of total reservoir capacity.

Month	Lower 95%	Point Forecast	Upper 95%	Actual
January	5.969645	19.938474	33.90730	20.1894
February	9.362830	23.832846	38.30286	26.8816
March	14.569318	29.461059	44.35280	38.6927
April	25.423740	40.473891	55.52404	52.0303
May	22.264648	37.325024	52.38540	-
June	4.783456	19.868235	34.95301	-
July	-8.555621	6.529245	21.61411	-
August	-11.074077	4.013113	19.10030	-
September	-11.705402	3.382046	18.46949	-
October	-8.657315	6.430261	21.51784	-
November	-4.823829	10.263816	25.35146	-
December	1.056789	16.144436	31.23208	-

4.9.5 Pine Flat Lake Reservoir (PNF) Forecast Comparison

To check the accuracy of the model for Pine Flat Lake Reservoir, the same modeling process was repeated for the 2014 year. The comparison between forecasts and 95% prediction interval is presented below in **Figure 45**.


2014 Forecast of PNF Reservoir Capacity

Twelve Month Forecast of PNF Reservoir Capacity

Figure 45. 2014 and 2016 point forecasts and prediction intervals for Pine Flat Lake Reservoir (PNF). On the left is the 2014 forecast and on the right is the 2016 forecast. For both graphs, the observed data is given in black, the point forecast is given by the solid red line with month markers, and the prediction interval is given by the dotted red line.

Through April, the 2016 forecast for Pine Flat Lake Reservoir seems to do around the same as the 2014 forecast, as both forecasts contain the actual observed values within the 95% prediction interval. The 2014 forecast, however, seems to do slightly better as the observed values seem closer to the point forecasts, on average, than the observed values in the 2016 forecast. Overall, the forecasting process seems to perform well.

4.10 Castaic Lake Reservoir (CAS) Forecast Results

4.10.1 Castaic Lake Reservoir (CAS) Data Retrieval

Castaic Lake Reservoir (CAS) data was retrieved from the CDEC using sharpshootR for the date range of October 1974 to April 2016 at a monthly level, totaling 499 observations. The raw reservoir storage was then divided by Castaic Lake Reservoir's capacity of 325,000 acre-feet of storage capacity to find the capacity percentage of every observation.

4.10.2 Castaic Lake Reservoir (CAS) Data Inspection

Upon initial inspection, the data for Castaic Lake Reservoir (shown below in **Figure 46**) did not appear to require any additional transformation or cleaning.

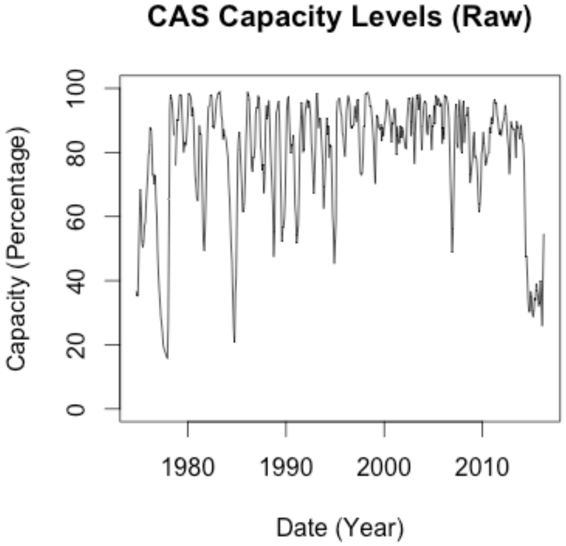


Figure 46. Raw data for Castaic Lake Reservoir (CAS).

However, after checking for any NA values, one was found at index 48. This observation will be handled by imputation as part of the forecasting process.

The historical levels of the Castaic Lake Reservoir have a fairly stable, if not small variance. No transformation seems required and the large drops in percentage capacity coincide with statewide drought conditions. As such, the observations are recorded as intended and are not incorrect measurements.

4.10.3 Castaic Lake Reservoir (CAS) Data Cleaning

To clean the data, the NA value was addressed by the forecasting process. The value was imputed as an average between the previous two months and the next two months. The cleaned data is presented below in **Figure 47**.

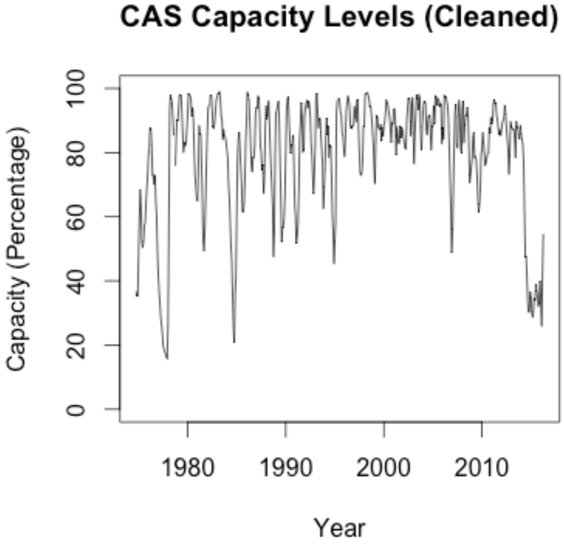


Figure 47. Cleaned data for Castaic Lake Reservoir (CAS).

4.10.4 Castaic Lake Reservoir (CAS) Forecasting

Using the forecast.all function, an ARMA(1,3) model was derived to fit the stationary data. The residuals are presented below in **Figure 48**.

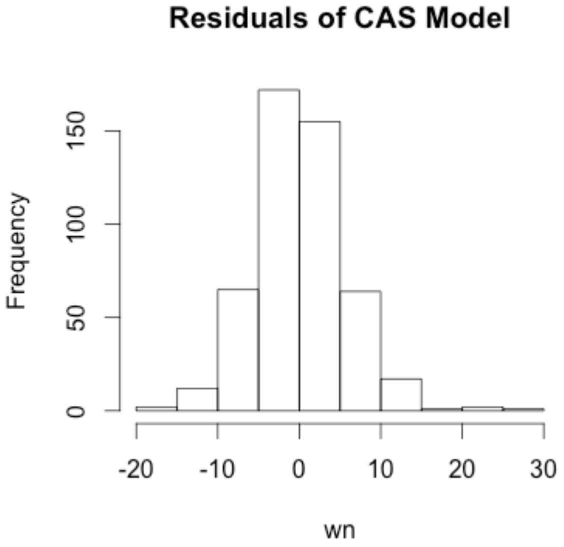


Figure 48. Histogram of residuals from the ARMA(1,3) model fitted to Castaic Lake Reservoir (CAS).

Based on the histogram, it appears that the residuals are mostly normal with a slightly right skew. According to the Shapiro-Wilk test, normality is rejected, but the histogram demonstrates that the residuals are approximately normal. The 95% prediction intervals presented in **Figure 49** below may not be accurate, but should be pretty close.

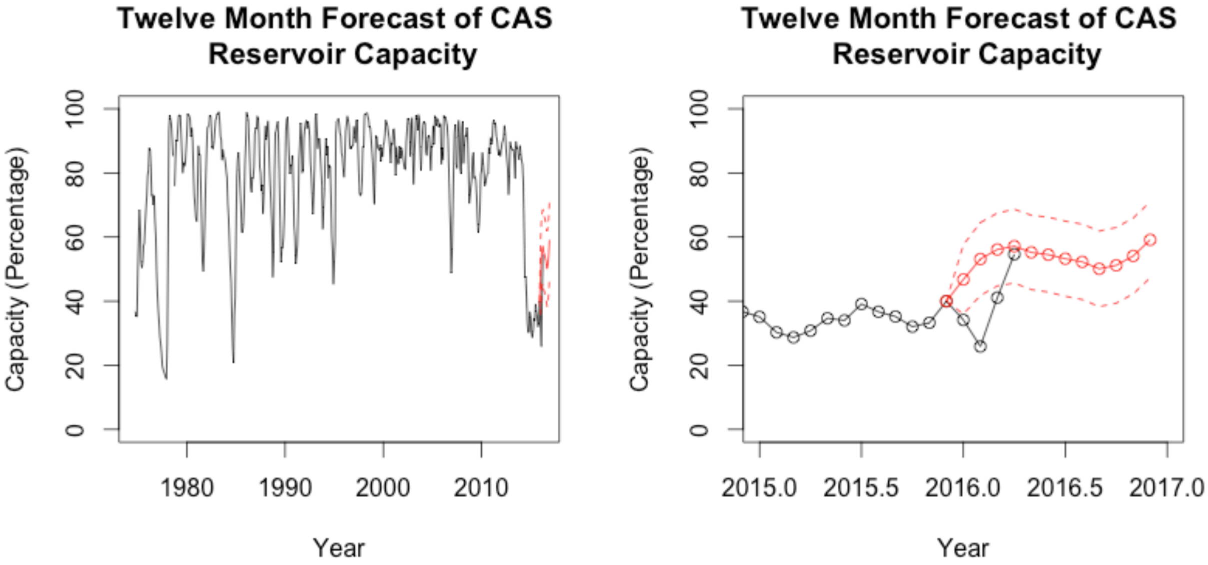


Figure 49. 2016 point forecast and 95% prediction interval for Castaic Lake Reservoir (CAS). On the left is the observed data in the black with the point forecast in solid red and the 95% prediction interval in dotted red lines. On the right is the observed data in black with the point forecast in solid red lines with month markers and the 95% prediction interval in dotted red lines, zoomed in for the time span of 2015-2017.

According to the rightmost graph, the 2016 forecast for Castaic Lake capacity percentage has not been very accurate. Only one observation has fallen within the 95% prediction interval and for all observations in general, the forecast seems to overestimate the percentage capacity levels.

Table 10. Numerical values for 2016 point forecast and 95% prediction interval for Castaic Lake Reservoir (CAS). Values reported as a percentage of total reservoir capacity.

Month	Lower 95%	Point Forecast	Upper 95%	Actual
January	35.96631	46.77772	57.58914	34.15477
February	41.77378	53.14079	64.50779	25.86985

March	44.74639	56.11501	67.48363	41.10092
April	45.64465	57.14824	68.65184	54.63631
May	43.60664	55.20827	66.80990	-
June	42.83154	54.50456	66.17758	-
July	41.52016	53.24528	64.97041	-
August	40.49572	52.25893	64.02213	-
September	38.30407	50.09514	61.88620	-
October	39.36372	51.17518	62.98664	-
November	42.28336	54.10976	65.93616	-
December	47.29740	59.13475	70.97210	-

4.10.5 Castaic Lake Reservoir (CAS) Forecast Comparison

To check the accuracy of the model for Castaic Lake Reservoir, the same modeling process was repeated for the 2014 year. The comparison between forecasts and 95% prediction interval is presented below in **Figure 50**.

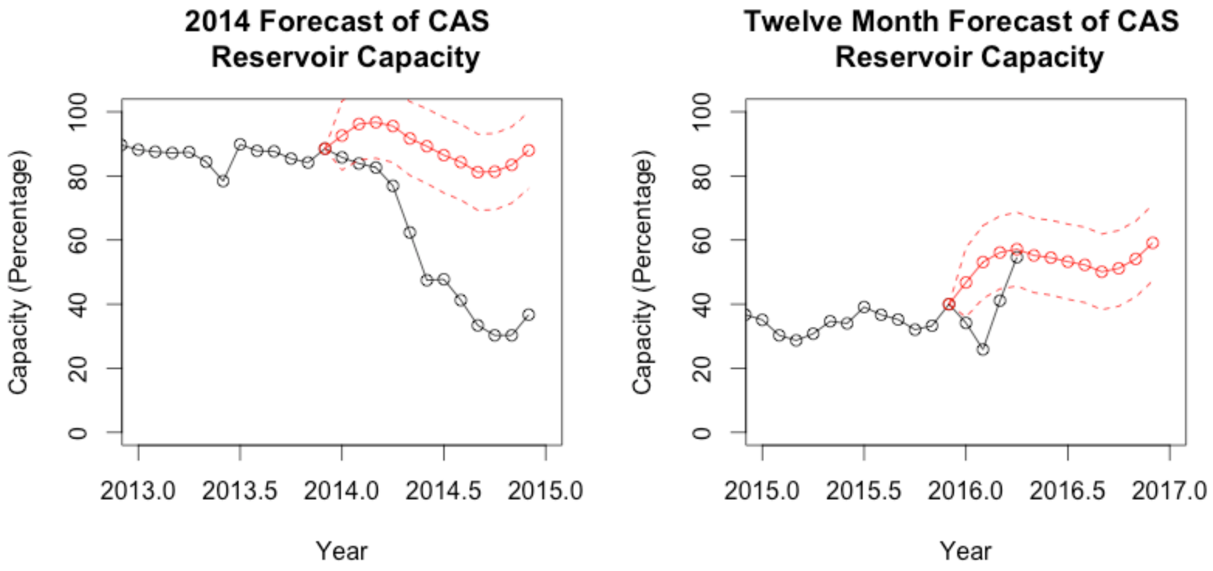


Figure 50. 2014 and 2016 point forecasts and prediction intervals for Castaic Lake Reservoir (CAS). On the left is the 2014 forecast and on the right is the 2016 forecast. For both graphs, the observed data is given in black, the point forecast is given by the solid red line with month markers, and the prediction interval is given by the dotted red line.

For both 2014 and 2016, neither year's forecasts seem to perform well at all. In 2014, only one observation fell within the 95% prediction interval, a pattern which 2016 has followed thus far. In both years, the forecasting process seemed to heavily overestimate the percentage capacity in Castaic Lake Reservoir. Thus, the forecasting process does not appear to work well for Castaic Lake in both El Niño and non-El Niño years.

It should be noted that there seems to be an unusual drop in percentage capacity for Castaic Lake in 2014. Further investigation should be performed to see if this drop was due to human intervention and whether the forecasts should be accordingly adjusted.

4.11 Lake Perris Reservoir (PRR) Forecast Results

4.11.1 Lake Perris Reservoir (PRR) Data Retrieval

Lake Perris Reservoir (PRR) data was retrieved from the CDEC using sharpshootR for the date range of October 1974 to April 2016 at a monthly level, totaling 499 observations. The raw reservoir storage was then divided by Lake Perris Reservoir's capacity of 131,452 acre-feet of storage capacity to find the capacity percentage of every observation.

4.11.2 Lake Perris Reservoir (PRR) Data Inspection

Upon initial inspection, the data for Lake Reservoir (shown below in **Figure 51**) required some deeper inspection due to fluctuating variance and a sudden abrupt drop around 2005.

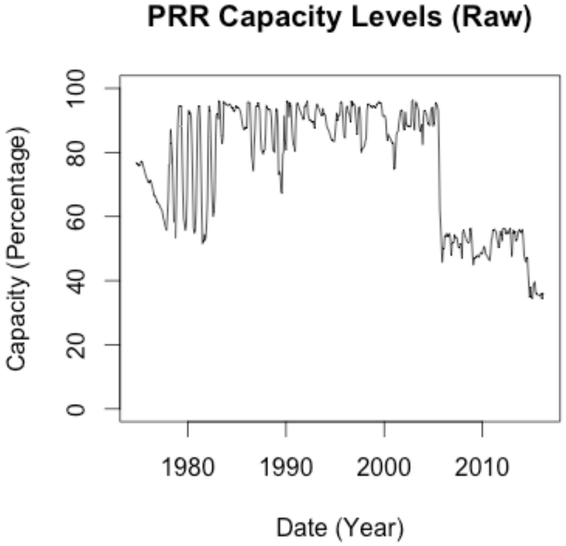


Figure 51. Raw data for Lake Perris Reservoir (PRR).

After further research, the sharp drop in 2005 was determined to have resulted from water drainage from Lake Perris which occurred over the course of several months. After consultation with a time series analysis professor, the choice was made to simply model Lake Perris' percentage capacity levels based on post-2005 data to more accurately generate future forecasts.

The drop in the early 2010s is consistent with statewide drought conditions and thus, is recorded as intended.

A single NA value was found at index in the data, but that value is in the portion of the data which will not be included in the modeling process. Consequently, the NA value will not require imputation.

4.11.3 Lake Perris Reservoir (PRR) Data Cleaning

To clean the data, all data before and during the water drainage was dropped from the dataset. The new dataset included observations 378 through 499.

The cleaned data is presented below in **Figure 52**.

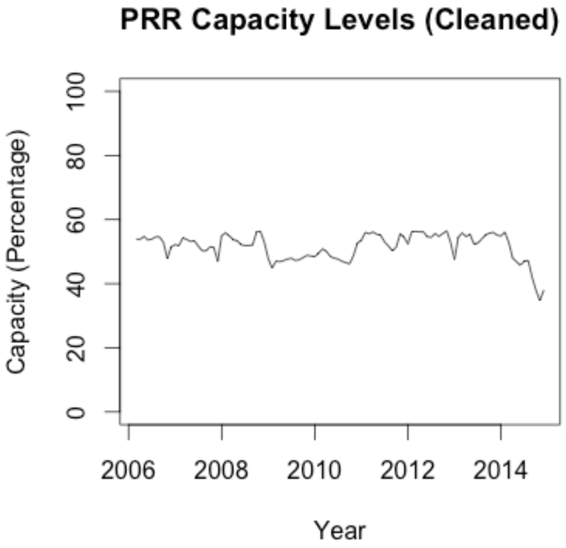


Figure 52. Cleaned data for Lake Perris Reservoir (PRR).

4.11.4 Lake Perris Reservoir (PRR) Forecasting

Before starting the forecasting process, it should be noted that because of the much smaller size of clean data for Lake Perris in comparison to other reservoirs, the forecast may be overfitted to recent observations and not accurately represent historical patterns.

Using the forecast.all function, an ARMA(0,2) model was derived to fit the stationary data. The residuals are presented below in **Figure 53**.

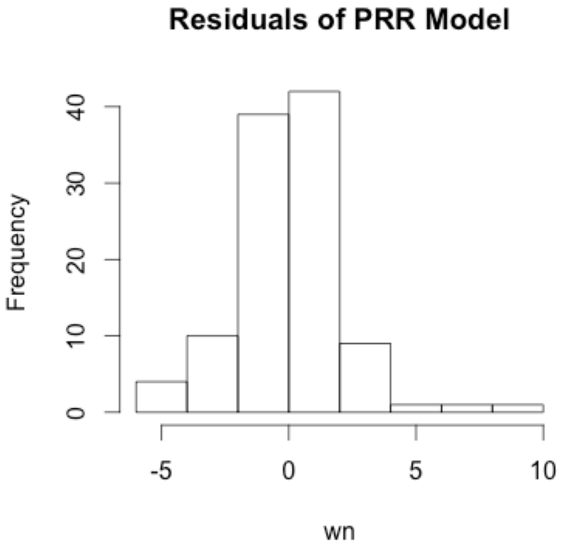


Figure 53. Histogram of residuals from the ARMA(0,2) model fitted to Lake Perris Reservoir (PRR).

Based on the histogram, it appears that the residuals are mostly normal with a slightly left skew. According to the Shapiro-Wilk test, normality is rejected, but the histogram demonstrates that the residuals are approximately normal. The 95% prediction intervals presented in **Figure 54** below may not be accurate, but should be pretty close.

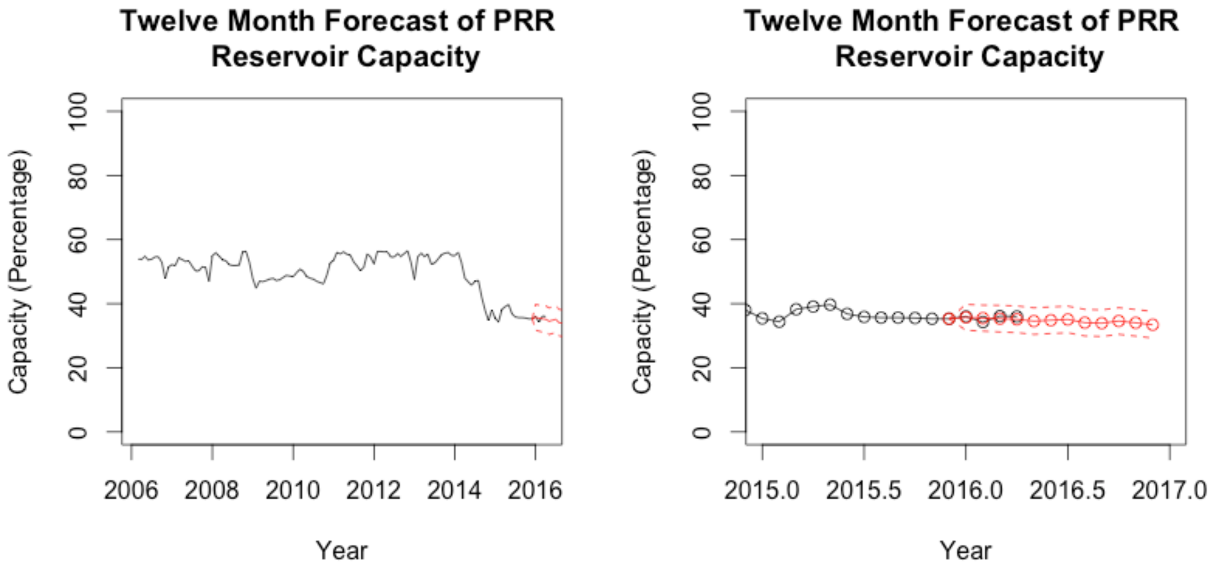


Figure 54. 2016 point forecast and 95% prediction interval for Lake Perris Reservoir (PRR). On the left is the observed data in the black with the point forecast in solid red and the 95% prediction interval in dotted red lines. On the right is the observed data in black with the point forecast in solid red lines with month markers and the 95% prediction interval in dotted red lines, zoomed in for the time span of 2015-2017.

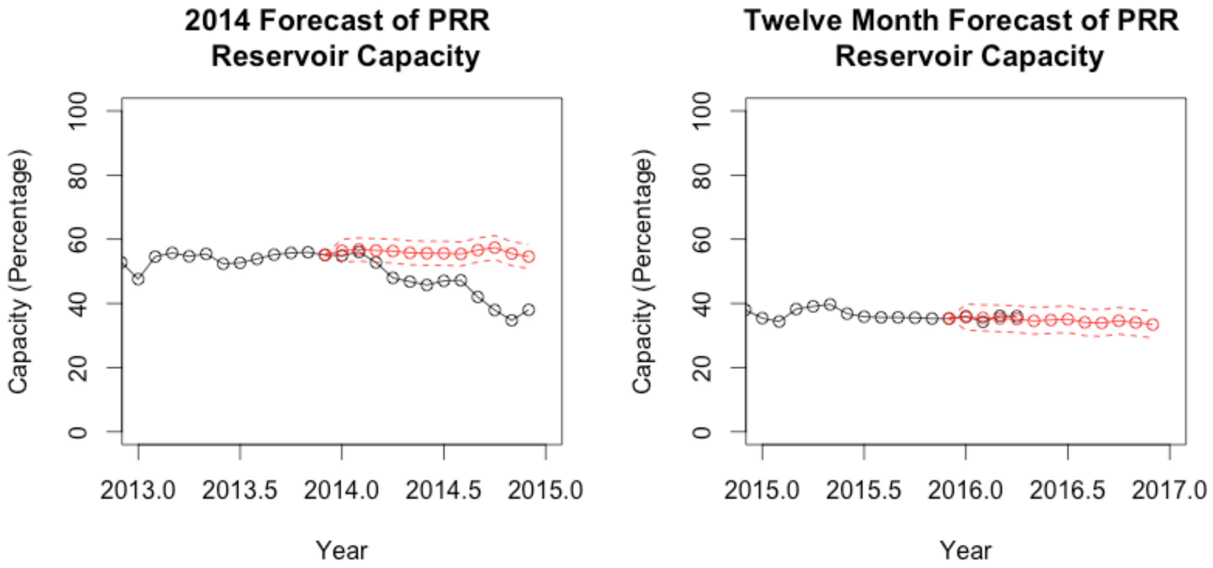
According to the rightmost graph, the 2016 forecast for Lake Perris has been extremely accurate. The prediction interval is relatively narrow in comparison to most other reservoirs and the four observations from 2016 are almost exactly in line with the point forecasts produced by this process.

Table 11. Numerical values for 2016 point forecast and 95% prediction interval for Lake Perris Reservoir (PRR). Values reported as a percentage of total reservoir capacity.

Month	Lower 95%	Point Forecast	Upper 95%	Actual
January	31.69491	35.73575	39.77659	35.99108
February	31.42529	35.48141	39.53753	34.25661
March	31.16766	35.31959	39.47152	36.11965
April	30.97247	35.12440	39.27633	35.99108
May	30.40648	34.55841	38.71034	-
June	30.68221	34.83414	38.98607	-
July	30.92457	35.07649	39.22842	-
August	29.92103	34.07296	38.22489	-
September	29.75921	33.91114	38.06307	-
October	30.43911	34.59104	38.74297	-
November	29.87313	34.02505	38.17698	-
December	29.27376	33.42569	37.57762	-

4.11.5 Lake Perris Reservoir (PRR) Forecast Comparison

To check the accuracy of the model for Lake Perris Reservoir, the same modeling process was repeated for the 2014 year. The comparison between forecasts and 95% prediction interval is presented below in **Figure 55**.


2014 Forecast of PRR Reservoir Capacity

Twelve Month Forecast of PRR Reservoir Capacity

Figure 50. 2014 and 2016 point forecasts and prediction intervals for Lake Perris Reservoir (PRR). On the left is the 2014 forecast and on the right is the 2016 forecast. For both graphs, the observed data is given in black, the point forecast is given by the solid red line with month markers, and the prediction interval is given by the dotted red line.

Interestingly, the 2016 forecast seems to be much more accurate through the first four months of 2016 than the 2014 forecast was through the same point in the year. While January and February of 2014 fell in line with the forecast, the forecast overestimated the percentage capacity in Lake Perris for the rest of the year. It is interesting that in both 2014 and 2016, the forecasts were exceptionally flat in comparison to forecasts for other reservoirs. While the forecasting process appears to be accurate for 2016, more research is required before conclusively saying that the process works well in either El Niño or non-El Niño years.

5 Conclusions

Overall, the presented method for forecasting reservoir percentage capacity levels did not seem to do well in 2016. Only two reservoirs' forecasts and prediction intervals matched up with actual observations for January through April of 2016. One additional reservoir's prediction interval contained three of four months' observations, and all the other reservoirs predicted 50% or more of the observations incorrectly. A common trend was that many of the models seemed to underestimate reservoir percentage capacity.

One idea for this discrepancy was the higher-than-usual levels of precipitation experienced in 2016 as a result of the El Niño event. To that end, the 2014 forecasts seemed to do a much better job of forecasting reservoir levels when using the same forecasting process. While not perfect, the forecasts were, on average, more accurate to actual observations. Thus, it may be the case that the forecasting process used is not accurate in years containing an El Niño event.

6 Further Considerations

There are a variety of ideas on how to further expand this research. First and foremost, it would be prudent to create a function that can quantify the accuracy of these forecasts, perhaps by using Mean Squared Error or some similar measure. Then, by using this function, a measure of relative accuracy could be obtained to compare forecasts in non-El Niño years to El Niño years. Following this, a more conclusive result on the accuracy of this forecasting process might be made.

An assumption which this project made was that trend could be accurately modeled and eliminated with a simple first-order difference. Further research might investigate modeling the trend using a quadratic function instead. Furthermore, a different model for seasonality besides sum of harmonics may be useful for further benchmarking the performance of this specific forecasting process.

Another avenue for further research would be to include spatial features into this analysis. This forecasting process was based solely on temporal elements, but given the nature of reservoirs and climate patterns within the state of California, it would be of interest to see what different forecasts a spatio-temporal analysis might produce.

7 Acknowledgements

I would like to take a moment and thank two professors for their guidance and direction for this project.

Thank you, Professor Duncan Temple Lang, for overseeing the overall progress of this project, for giving this project direction and ideas, and for extreme patience in dealing with the ever-changing direction of this project. Additionally, thank you for the miscellaneous R code assistance and for providing numerous directions in which to expand or deepen the understanding of this project.

Thank you, Professor Joshua Patrick, for helping flesh out the best ways to proceed with this analysis, providing consultation on best practices for creating these forecasts, and generating ideas in which this project could be expanded. Best of luck at Baylor University next year.

8 Code Appendix

All updated code can be found on https://github.com/jeremychan17/California-Reservoir-Project

8.1 Reservoir Forecasting Function (forecast.all())

```
# The purpose of this function is to create a twelve month 95% confidence
forecast for a given
# California reservoir. The year predicted will be the last year in the dataset.

# Expected Input: The function takes a "reservoir" object. The object is created
using the
# R package "sharpshootR." See the associated documentation for that package for
more information
# on retrieving the necessary information. A sample call to the necessary
function as well
# as a call for this function is included below.

# Sample Input:
# temp = CDECquery(x, 15, interval = "M", "1900-01-01", Sys.Date())
# forcast.all(temp)

# Expected Output: forecast.all has several outputs. The first output is three
plots: The first
# plot is a histogram of the time series model residuals, used to check for
normality of the
# residuals. The second plot is the entire time series plus the forecast and
prediction interval.
# The forecast and prediction interval are all in red lines, with the prediction
interval denoted
# by a dotted red line. The third plot is a zoomed-in plot of the prediction,
including the
# year being predicted and the previous year for reference. Again, the point
forecast is indicated
# by a red line, but this time each month is denoted by a circle. The prediction
interval
# remains as a dotted red line. Additionally, the function returns several
objects shown below:
# $model returns the time series model which was fitted to the data
# $forecast.list returns the point forecast as well as the upper and lower 95%
prediction interval
# $point.forecast returns only the point forecast
# $upper.forecast returns only the upper 95% prediction interval bound
# $lower.forecast returns only the lower 95% prediction interval bound
# $residuals returns the residuals from the fitted time series model


forecast.all = function(waterObject)
{

  # Finds first january and last december
  firstjan = grep("January", waterObject$month)[1]
  lastdec = grep("December", waterObject$month)[length(grep("December",
waterObject$month))]

  # Pulls dates
  first_jan = substr(strptime(waterObject[firstjan,]$datetime, format = "%F"), 0,
10)
  last_dec = substr(strptime(waterObject[lastdec,]$datetime, format = "%F"), 0,
10)

  # Creates Ranges
  start_range = c(as.numeric(substr(first_jan, 0, 4)), 1)
  end_range = c(as.numeric(substr(last_dec, 0, 4)), 12)

  # Creates Time Series
  year_series = ts(waterObject$cap[firstjan:lastdec], start = start_range, end =
end_range, frequency = 12)

  # Removes the last observation if it is an NA
  if(is.na(year_series[length(year_series)])==T)
  {
    year_series = year_series[1:length(year_series)-1]
  }

  # Set up variables
  x = as.vector(year_series)
  n = length(data)
  t = 1:n

  # Impute any missing values
  y = sapply(1:length(x), function(i){
    if(is.na(x[i])==T)
    {
      if(i == 2)
      {
        x[i] = (x[i-1]+x[i+1])/2
      }
      else {x[i] = (x[i-2]+x[i-1]+x[i+1]+x[i+2])/4}
    }
    else x[i] = x[i]
  })

  # Remove trend through differencing
  y = diff(y)

  # Removal seasonal component

  # Rescale t
  n = length(t)
  t = 1:length(y)
  t = (t)/n

  # Matrix of harmonics
  d = 12
  n.harm = d/2
  harm = matrix(nrow=length(t), ncol=2*n.harm)
  for(i in 1:n.harm){
    harm[,i*2-1] = sin(n/d * i *2*pi*t)
    harm[,i*2] = cos(n/d * i *2*pi*t)
  }
  colnames(harm)=
    paste0(c("sin", "cos"), rep(1:n.harm, each = 2))

  # Fit on all of the sines and cosines
```

```
dat = data.frame(y, harm)
fit = lm(y~., data=dat)

# Setup the full model and the model with only an intercept
full = lm(y~.,data=dat)
reduced = lm(y~1, data=dat)

# Stepwise regression starting with the full model
fit.back = step(full, scope = formula(reduced), direction = "both", trace = F)
resid = residuals(fit.back)

# Get back the original t so that we can plot over this range
t = 1:length(y)

# Fitting a model
library(forecast)
fit.y = auto.arima(resid, allowmean = F, trace = F, stepwise = F)
wn = resid(fit.y)

# Tests for stationarity of resulting series
library(tseries)
if(kpss.test(wn)$p.value < .05)
{
  print("Warning: Time Series is not stationary by kpss test")
}

if(adf.test(wn)$p.value > .05)
{
  print("Warning: Time series is not stationary by adf test")
}

# Test for residual independence
if((Box.test(wn, type = "Ljung-Box", lag = 24)$p.value) < .05){
  print("Warning: Residuals are not independent.")
}

# Test for residual normality
if(shapiro.test(wn)$p.value < .05){
  print("Warning: Residuals are not normal by Shapiro-Wilk Test.")
}

## Forecasting ##

# Forecast noise
fc = forecast(fit.y, h=12, level = 0.95)

# Forecast seasonality
season.fc = fit.back$fitted.values[1:12]

# Create combined seasonality and noise forecast
y.fc = c(y, season.fc+fc$mean)
bound = (fc$upper - fc$lower)/2

# Differencing Inverse to return to original values
fc.all = diffinv(y.fc, difference=1, xi =  x[1])
fc.upper = fc.all[(length(x)+1):(length(x)+12)]+bound
fc.lower = fc.all[(length(x)+1):(length(x)+12)]-bound

forecast.list = list("point.forecast" = fc.all[(length(x)+1):(length(x)+12)],
              "upper.forecast" = fc.upper, "lower.forecast" = fc.lower)

  # Objects to return
  model = fit.y
  point.forecast = forecast.list$point.forecast
  upper.forecast = forecast.list$upper.forecast
  lower.forecast = forecast.list$lower.forecast

  # Finds first january and last december
  lastobs = length(waterObject$cap)

  # Pulls dates
  firstobs = substr(strptime(waterObject[1,]$datetime, format = "%F"), 0, 10)
  lastobs = substr(strptime(waterObject[length(waterObject$cap),]$datetime,
format = "%F"), 0, 10)

  # Creates Ranges
  full_range_start = c(as.numeric(substr(firstobs, 0, 4)),
as.numeric(substr(firstobs, 6,7)))
  full_range_end = c(as.numeric(substr(lastobs, 0, 4)),
as.numeric(substr(lastobs, 6,7)))

  forecast_start = c(as.numeric(substr(last_dec, 0, 4)), 1)
  forecast_end = c(as.numeric(substr(lastobs, 0, 4)), 12)

  point_joined = c(year_series[length(year_series)], point.forecast)
  upper_joined = c(year_series[length(year_series)], upper.forecast)
  lower_joined = c(year_series[length(year_series)], lower.forecast)

  # Creates Time Series
  full_series = ts(waterObject$cap, start = full_range_start, end =
full_range_end, frequency = 12)
  point_series = ts(point_joined, start = end_range, end = forecast_end,
frequency = 12)
  upper_series = ts(upper_joined, start = end_range, end = forecast_end,
frequency = 12)
  lower_series = ts(lower_joined, start = end_range, end = forecast_end,
frequency = 12)

  # Create Plots

  # Plot Raw Data
  ID = waterObject[1,]$ID
  raw_data = ts(waterObject$cap, start = full_range_start, end = full_range_end,
frequency = 12)
  plot(raw_data, type = "l", main = paste(ID, "Capacity Levels (Raw)"), ylab =
         "Capacity (Percentage)", ylim = c(0, 100), xlab = "Year")

  # Plot Cleaned Data
  plot(full_series, type = "l", main = paste(ID, "Capacity Levels (Cleaned)"),
ylab =
         "Capacity (Percentage)", ylim = c(0, 100), xlab = "Year")

  # Residuals Plot
  hist(wn, main = paste("Residuals of", ID, "Model"))

  # Forecast Plot

  # Zoomed out
  plot(full_series, type = "l", main =
         paste("Twelve Month Forecast of", ID, "\n Reservoir Capacity"), ylab =
         "Capacity (Percentage)", ylim = c(0, 100), xlab = "Year")
  lines(point_series, col = "red", type = "l")
  lines(upper_series, col = "red", lty = 2)
  lines(lower_series, col = "red", lty = 2)

  # Zoomed In
  year = (as.numeric(substr(lastobs, 0, 4)))
  plot(full_series, xlim = c(year-1, year+1), type = "o", main =
         paste("Twelve Month Forecast of", ID, "\n Reservoir Capacity"), ylab =
         "Capacity (Percentage)", ylim = c(0, 100), xlab = "Year")
  lines(point_series, col = "red", type = "o")
  lines(upper_series, col = "red", lty = 2)
  lines(lower_series, col = "red", lty = 2)

  # Objects to return
  model = fit.y
  point.forecast = forecast.list$point.forecast
  upper.forecast = forecast.list$upper.forecast
  lower.forecast = forecast.list$lower.forecast

  # Return Call
  returnList = list(model = fit.y, forecast.list = forecast.list,
                    point.forecast = point.forecast, upper.forecast =
upper.forecast,
                    lower.forecast = lower.forecast, residuals = wn)
}
```

8.2 2014 Reservoir Forecasting Function (forecast.2014())

```
# The purpose of this function is to create a twelve month 95% confidence
forecast for a given
# California reservoir. The year predicted is 2014.

# Expected Input: The function takes a "reservoir" object. The object is created
using the
# R package "sharpshootR." See the associated documentation for that package for
more information
# on retrieving the necessary information. A sample call to the necessary
function as well
# as a call for this function is included below.

# Sample Input:
# temp = CDECquery(x, 15, interval = "M", "1900-01-01", Sys.Date())
# forcast.all(temp)

# Expected Output: forecast.all has several outputs. The first output is three
plots: The first
# plot is a histogram of the time series model residuals, used to check for
normality of the
# residuals. The second plot is the entire time series plus the forecast and
prediction interval.
# The forecast and prediction interval are all in red lines, with the prediction
interval denoted
# by a dotted red line. The third plot is a zoomed-in plot of the prediction,
including the
```

```
# year being predicted and the previous year for reference. Again, the point
forecast is indicated
# by a red line, but this time each month is denoted by a circle. The prediction
interval
# remains as a dotted red line. Additionally, the function returns several
objects shown below:
# $model returns the time series model which was fitted to the data
# $forecast.list returns the point forecast as well as the upper and lower 95%
prediction interval
# $point.forecast returns only the point forecast
# $upper.forecast returns only the upper 95% prediction interval bound
# $lower.forecast returns only the lower 95% prediction interval bound
# $residuals returns the residuals from the fitted time series

forecast.2014 = function(waterObject)
{
  # Finds first january and last december
  firstjan = grep("January", waterObject$month)[1]
  lastdec = grep("December", waterObject$month)[length(grep("December",
waterObject$month))]

  # Pulls dates
  first_jan = substr(strptime(waterObject[firstjan,]$datetime, format = "%F"), 0,
10)
  last_dec = substr(strptime(waterObject[lastdec,]$datetime, format = "%F"), 0,
10)

  # Creates Ranges
  start_range = c(as.numeric(substr(first_jan, 0, 4)), 1)
  end_range = c(as.numeric(2013), 12)

  # Creates Time Series
  # year_series = ts(waterObject$cap, start = start_range, end = end_range,
frequency = 12)
  year_series = ts(waterObject$cap[firstjan:lastdec], start = start_range, end =
end_range, frequency = 12)

  # Set up variables
  x = as.vector(year_series)
  n = length(data)
  t = 1:n

  # Impute any missing values
  y = sapply(1:length(x), function(i){
    if(is.na(x[i])==T)
    {
      if(i == 2)
      {
        x[i] = (x[i-1]+x[i+1])/2
      }
      else {x[i] = (x[i-2]+x[i-1]+x[i+1]+x[i+2])/4}
    }
    else x[i] = x[i]
  })

  # Remove trend through differencing
  y = diff(y)

  # Removal seasonal component
  # Rescale t
  n = length(t)
  t = 1:length(y)
  t = (t)/n

  # Matrix of harmonics
  d = 12
  n.harm = d/2
  harm = matrix(nrow=length(t), ncol=2*n.harm)
  for(i in 1:n.harm){
    harm[,i*2-1] = sin(n/d * i *2*pi*t)
    harm[,i*2] = cos(n/d * i *2*pi*t)
  }
  colnames(harm)=
    paste0(c("sin", "cos"), rep(1:n.harm, each = 2))

  # Fit on all of the sines and cosines
  dat = data.frame(y, harm)
  fit = lm(y~., data=dat)

  # Setup the full model and the model with only an intercept
  full = lm(y~.,data=dat)
  reduced = lm(y~1, data=dat)

  # Stepwise regression starting with the full model
  fit.back = step(full, scope = formula(reduced), direction = "both", trace = F)
  resid = residuals(fit.back)

  # Get back the original t so that we can plot over this range
  t = 1:length(y)

  # Fitting a model
  library(forecast)
  fit.y = auto.arima(resid, allowmean = F, trace = F, stepwise = F)
  wn = resid(fit.y)

  # Tests for stationarity of resulting series
  library(tseries)
  if(kpss.test(wn)$p.value < .05)
  {
    print("Warning: Time Series is not stationary by kpss test")
  }

  if(adf.test(wn)$p.value > .05)
  {
    print("Warning: Time series is not stationary by adf test")
  }

  # Test for residual independence
  if(Box.test(wn, type = "Ljung-Box", lag = 24)$p.value < .05){
    print("Warning: Residuals are not independent.")
  }

  # Test for residual normality
  if(shapiro.test(wn)$p.value < .05){
    print("Warning: Residuals are not normal by Shapiro-Wilk Test.")
  }
```

```
  ## Forecast ##

  # Forecast noise
  fc = forecast(fit.y, h=12, level = 0.95)

  # Forecast seasonality
  season.fc = fit.back$fitted.values[1:12]

  # Create combined seasonality and noise forecast
  y.fc = c(y, season.fc+fc$mean)
  bound = (fc$upper - fc$lower)/2

  # Differencing Inverse to return to original values
  fc.all = diffinv(y.fc, difference=1, xi =  x[1])
  fc.upper = fc.all[(length(x)+1):(length(x)+12)]+bound
  fc.lower = fc.all[(length(x)+1):(length(x)+12)]-bound

  forecast.list = list("point.forecast" = fc.all[(length(x)+1):(length(x)+12)],
                       "upper.forecast" = fc.upper, "lower.forecast" = fc.lower)

  # Objects to return
  model = fit.y
  point.forecast = forecast.list$point.forecast
  upper.forecast = forecast.list$upper.forecast
  lower.forecast = forecast.list$lower.forecast

  # Finds first january and last december
  lastobs = length(waterObject$cap)

  # Pulls dates
  firstobs = substr(strptime(waterObject[1,]$datetime, format = "%F"), 0, 10)
  lastobs = substr(strptime(waterObject[length(waterObject$cap),]$datetime,
format = "%F"), 0, 10)

  # Creates Ranges
  full_range_start = c(as.numeric(substr(firstobs, 0, 4)),
as.numeric(substr(firstobs, 6,7)))
  full_range_end = c(2014, 12)

  forecast_start = c(2014, 1)
  forecast_end = c(2014, 12)

  point_joined = c(year_series[length(year_series)], point.forecast)
  upper_joined = c(year_series[length(year_series)], upper.forecast)
  lower_joined = c(year_series[length(year_series)], lower.forecast)

  # Creates Time Series
  full_series = ts(waterObject$cap, start = full_range_start, end =
full_range_end, frequency = 12)
  point_series = ts(point_joined, start = end_range, end = forecast_end,
frequency = 12)
  upper_series = ts(upper_joined, start = end_range, end = forecast_end,
frequency = 12)
  lower_series = ts(lower_joined, start = end_range, end = forecast_end,
frequency = 12)

  # Create Plots

  # Plot Raw Data
  ID = waterObject[1,]$ID
  index_2015 = grep(2015, waterObject$year)[1]
  raw_data = ts(waterObject$cap[0:index_2015], start = full_range_start, end =
full_range_end, frequency = 12)
  plot(raw_data, type = "l", main = paste(ID, "Capacity Levels (Raw)"), ylab =
         "Capacity (Percentage)", ylim = c(0, 100), xlab = "Year")

  # Plot Cleaned Data
  plot(full_series, type = "l", main = paste(ID, "Capacity Levels (Cleaned)"),
ylab =
         "Capacity (Percentage)", ylim = c(0, 100), xlab = "Year")

  # Residuals Plot
  hist(wn, main = paste("Residuals of 2014", ID, "Model"))

  # Forecast Plot

  # Zoomed out
  plot(full_series, type = "l", main =
         paste("2014 Forecast of", ID, "\n Reservoir Capacity"), ylab =
         "Capacity (Percentage)", ylim = c(0, 100), xlab = "Year")
  lines(point_series, col = "red", type = "l")
  lines(upper_series, col = "red", lty = 2)
  lines(lower_series, col = "red", lty = 2)

  # Zoomed In
  year = (as.numeric(substr(lastobs, 0, 4)))
  plot(full_series, xlim = c(2013, 2015), type = "o", main =
         paste("2014 Forecast of", ID, "\n Reservoir Capacity"), ylab =
         "Capacity (Percentage)", ylim = c(0, 100), xlab = "Year")
  lines(point_series, col = "red", type = "o")
  lines(upper_series, col = "red", lty = 2)
  lines(lower_series, col = "red", lty = 2)

  # Objects to return
  model = fit.y
  point.forecast = forecast.list$point.forecast
  upper.forecast = forecast.list$upper.forecast
  lower.forecast = forecast.list$lower.forecast

  # Return Call
  returnList = list(model = fit.y, forecast.list = forecast.list,
                    point.forecast = point.forecast, upper.forecast =
upper.forecast,
                    lower.forecast = lower.forecast, residuals = wn)
}
```

8.3 Data Retrieval and Forecasting Code

~~~
---
title: "Reservoir_v3"
author: "Jeremy Chan"
date: "April 21, 2016"
output: html_document
---

Libraries
```{r}
~~~

```
library(sharpshootR)
library(XML)
library(gdata)
```

List of major reservoirs | Reservoir ID:

Trinity Lake | CLE
Lake Shasta | SHA
Lake Oroville | ORO
New Melones | NML
Folsom Lake | FOL
Don Pedro | DNP
San Luis | SNL
Millerton Lake | MIL
Pine Flat | PNF
Castaic Lake | CAS
Lake Perris | PRR

Data Retrieval
```{r}
resList = c("CLE", "SHA", "ORO", "NML", "FOL", "DNP", "SNL", "MIL", "PNF", "CAS",
"PRR")
rawData = lapply(resList, function(x) CDECquery(x, 15, interval = "M", "1900-01-
01", "2016-04-30"))

# Capacity table
u = "http://cdec.water.ca.gov/misc/resinfo.html" # Reservoir URL
table = readHTMLTable(u)[1] # Pulls Reservoir Table
table = table[[1]] # Takes first Column
names(table) = c("ID", "Dam", "Lake", "Stream", "Capacity") # Renames Columns of
Table
capacityList = table$Capacity # Pulls List of all Capacities
positions = sapply(1:length(resList), function(x) grep(resList[x], table$ID)) #
Obtains positions of all reservoirs of interest
finalCapList = capacityList[c(positions)] # List of capacities from reservoirs of
interest
finalCapList = as.numeric(gsub(",", "", finalCapList)) # Turns list into numerics

# Adds Capacity and computes capacity for each reservoir
for(i in 1:length(resList))
{
  rawData[[i]]$ID = resList[[i]] #adds ID Column
  rawData[[i]]$cap = (rawData[[i]]$value/finalCapList[[i]])*100 #Computes
capacity %
}

resData = rawData # resData will be the cleaned data

keep(rawData, resData, forecast.all, forecast.2014, finalCapList, sure = T)
```

Plotting Raw Data
```{r}
plot(rawData[[1]]$datetime, rawData[[1]]$cap, type = "l", main = "CLE Capacity
Levels (Raw)",
     ylab = "Capacity (Percentage)", xlab = "Date (Year)", ylim = c(0, 100))
plot(rawData[[2]]$datetime, rawData[[2]]$cap, type = "l", main = "SHA Capacity
Levels (Raw)",
```

~~~~
     ylab = "Capacity (Percentage)", xlab = "Date (Year)", ylim = c(0, 100))
plot(rawData[[3]]$datetime, rawData[[3]]$cap, type = "l", main = "ORO Capacity
Levels (Raw)",
     ylab = "Capacity (Percentage)", xlab = "Date (Year)", ylim = c(0, 100))
plot(rawData[[4]]$datetime, rawData[[4]]$cap, type = "l", main = "NML Capacity
Levels (Raw)",
     ylab = "Capacity (Percentage)", xlab = "Date (Year)", ylim = c(0, 100))
plot(rawData[[5]]$datetime, rawData[[5]]$cap, type = "l", main = "FOL Capacity
Levels (Raw)",
     ylab = "Capacity (Percentage)", xlab = "Date (Year)", ylim = c(0, 100))
plot(rawData[[6]]$datetime, rawData[[6]]$cap, type = "l", main = "DNP Capacity
Levels (Raw)",
     ylab = "Capacity (Percentage)", xlab = "Date (Year)", ylim = c(0, 100))
plot(rawData[[7]]$datetime, rawData[[7]]$cap, type = "l", main = "SNL Capacity
Levels (Raw)",
     ylab = "Capacity (Percentage)", xlab = "Date (Year)", ylim = c(0, 100))
plot(rawData[[8]]$datetime, rawData[[8]]$cap, type = "l", main = "MIL Capacity
Levels (Raw)",
     ylab = "Capacity (Percentage)", xlab = "Date (Year)", ylim = c(0, 100))
plot(rawData[[9]]$datetime, rawData[[9]]$cap, type = "l", main = "PNF Capacity
Levels (Raw)",
     ylab = "Capacity (Percentage)", xlab = "Date (Year)", ylim = c(0, 100))
plot(rawData[[10]]$datetime, rawData[[10]]$cap, type = "l", main = "CAS Capacity
Levels (Raw)",
     ylab = "Capacity (Percentage)", xlab = "Date (Year)", ylim = c(0, 100))
plot(rawData[[11]]$datetime, rawData[[11]]$cap, type = "l", main = "PRR Capacity
Levels (Raw)",
     ylab = "Capacity (Percentage)", xlab = "Date (Year)", ylim = c(0, 100))


```

Data Cleaning
```{r}
# Checks for NAs
# If only one or two NAs (that are spaced apart), imputation in forecast.all will
handle it
sum(is.na(resData[[1]]$cap)) # CLE no NA
sum(is.na(resData[[2]]$cap)) # SHA 2 NA
  which(is.na(resData[[2]]$cap)) # SHA 31, 196 # Far enough away that the
imputation will handle
sum(is.na(resData[[3]]$cap)) # ORO no NA
sum(is.na(resData[[4]]$cap)) # NML no NA
sum(is.na(resData[[5]]$cap)) # FOL no NA
sum(is.na(resData[[6]]$cap)) # DNP no NA
sum(is.na(resData[[7]]$cap)) # SNL no NA
sum(is.na(resData[[8]]$cap)) # MIL no NA
sum(is.na(resData[[9]]$cap)) # PNF 10 NA
  which(is.na(resData[[9]]$cap)) # PNF 10, 11, 12, 13, 14, 15, 16, 21, 22, 322 #
Just drop 10-22
sum(is.na(resData[[10]]$cap)) # CAS 1 NA
  which(is.na(resData[[10]]$cap)) # CAS 48 # Imputation will handle
sum(is.na(resData[[11]]$cap)) # PRR 1 NA
  which(is.na(resData[[11]]$cap)) # PRR 48 # Imputation will handle

# Cleaning PNF
# Since most of the NA values are near the beginning, we will simply drop values
1-22 and start at 23.
resData[[9]] = resData[[9]][23:length(resData[[9]]$cap),]
~~~~

~~~
# Fixes all but 322

# Cleaning PRR
# Since there was a draining in 2005, we will only use observations from past
that point
# Drop stabilizes at 378
# Drop all observations before 378
resData[[11]] = resData[[11]][378:length(resData[[11]]$cap),]

```

Forecasting 2016
```{r}
forc_CLE = forecast.all(resData[[1]]) # 2/4
forc_SHA = forecast.all(resData[[2]]) # 0/4
forc_ORO = forecast.all(resData[[3]]) # 1/4
forc_NML = forecast.all(resData[[4]]) # 2/4
forc_FOL = forecast.all(resData[[5]]) # 0/4
forc_DNP = forecast.all(resData[[6]]) # 2/4
forc_SNL = forecast.all(resData[[7]]) # 2/4
forc_MIL = forecast.all(resData[[8]]) # 3/4
forc_PNF = forecast.all(resData[[9]]) # 4/4
forc_CAS = forecast.all(resData[[10]]) # 1/4
forc_PRR = forecast.all(resData[[11]]) # 4/4
```

Forecasting 2014
```{r}
forc_CLE_2014 = forecast.2014(resData[[1]])
forc_SHA_2014 = forecast.2014(resData[[2]])
forc_ORO_2014 = forecast.2014(resData[[3]])
forc_NML_2014 = forecast.2014(resData[[4]])
forc_FOL_2014 = forecast.2014(resData[[5]])
forc_DNP_2014 = forecast.2014(resData[[6]])
forc_SNL_2014 = forecast.2014(resData[[7]])
forc_MIL_2014 = forecast.2014(resData[[8]])
forc_PNF_2014 = forecast.2014(resData[[9]])
forc_CAS_2014 = forecast.2014(resData[[10]])
forc_PRR_2014 = forecast.2014(resData[[11]])
```

Model Checking
```{r}
forc_CLE$model
forc_CLE_2014$model # same model
forc_SHA$model
forc_SHA_2014$model # different models
forc_ORO$model
forc_ORO_2014$model # same model
forc_NML$model
forc_NML_2014$model # same model
forc_FOL$model
forc_FOL_2014$model # different model
forc_DNP$model
forc_DNP_2014$model # same model
forc_SNL$model
forc_SNL_2014$model # same model
forc_MIL$model
forc_MIL_2014$model # same model
~~~

~~~
forc_PNF$model
forc_PNF_2014$model # different models
forc_CAS$model
forc_CAS_2014$model # different models
forc_PRR$model
forc_PRR_2014$model # different models
```
~~~