forecasting and time series lab

Time series models are used in a wide range of applications, particularly for forecasting.

Perform analyses on a time series of California dairy data. Specifically exploring the structure of the time series and forecast the monthly production of fresh milk in the state of California.

This exploration is performed in two steps:

- Explore the characteristics of the time series data.
- Decompose the time series of monthly milk production into trend, seasonal components, and remainder components.
- Apply time series models to the remainder component of the time series.
- Forecast the production of monthly milk production for a 12 month period.

The header of the data loaded from scripts of R code to:

- The data is read from a dataset in Azure Machine Learning subscription.
- A new column, of type POSIXct, is created. POSIXct is a flexible R data-time class. The strtime function formats a text string for conversion to the date-time class.
- The Month column is converted to an ordered R factor class and unnecessary columns removed

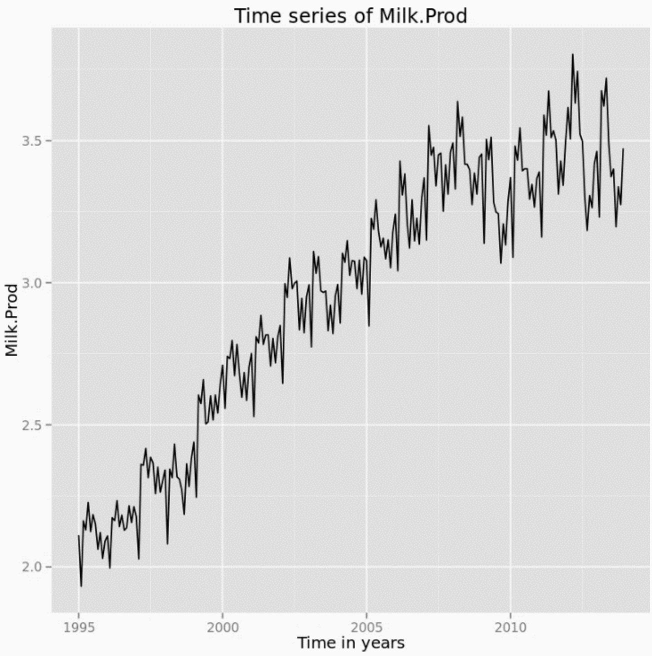
	Year	Month	Cotagecheese.Prod	Icecream.Prod	Milk.Prod	N.CA.Fat.Price	dateTime
1	1995	Jan	4.37	51.595	2.112	0.9803	1995-01-01
2	1995	Feb	3.695	56.086	1.932	0.8924	1995-02-01
3	1995	Mar	4.538	68.453	2.162	0.8924	1995-03-01
4	1995	Apr	4.28	65.722	2.13	0.8967	1995-04-01
5	1995	May	4.47	73.73	2.227	0.8967	1995-05-01
6	1995	Jun	4.238	77.994	2.124	0.916	1995-06-01

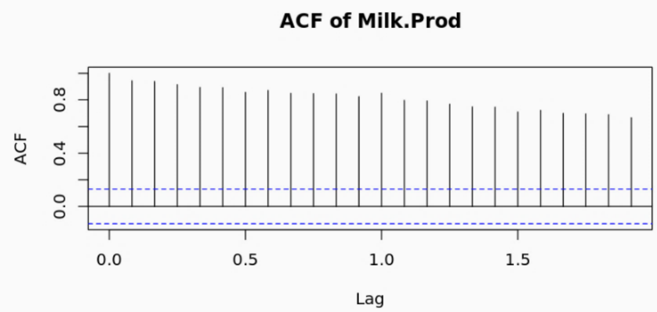
The POSIXct column is used to create the time axis on a Time Series plot of Milk Production (ggplot2)

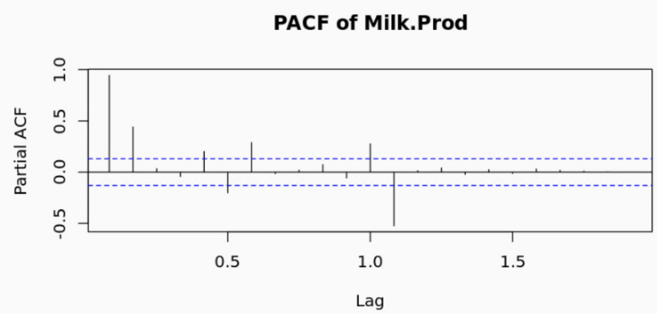
The plot shpws Milk Production increasing over the years with a decline in 2009 (the receission).

Additionally, the Time series exhibits a strong seasonal component with an annual cycle.

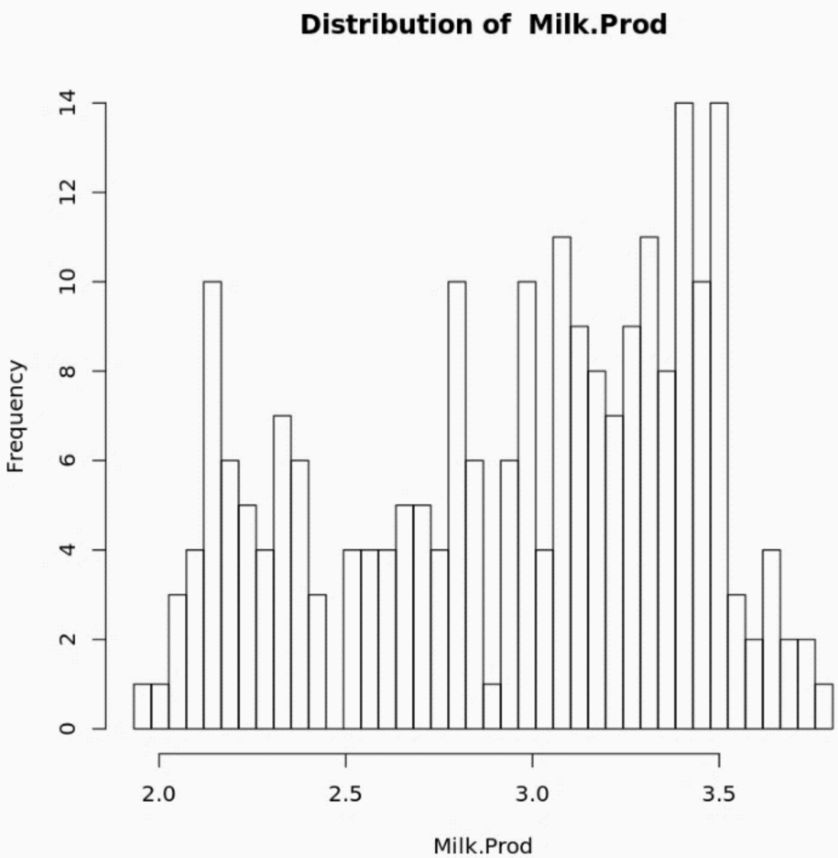
The Autocorrelation and Partial Correlation Functions are computed on the Time Series next:







The values of the ACF decays slowly between lags. This indicates considerable serial correlation between the time series values at the various lags, likely from the trend.



Autocorrelation is a fundamental property of time series.
The **Autocorrelation Function** or **ACF** provides information on the dependency of the time series values of previous values. The results of a ACF analysis is used later on to estimate the order of moving average processes. The **Partial Autocorrelation Function** or **PACF**, measures the correlation of the time series with its own lag values. Later in this lab you will use a **PACF** to estimate the order of an autoregressive process.

Plotting a histogram provides information on the distribution of values of the time series:

The histogram of the full milk production time series shows considerable dispersion. Again, such behavior is likely the result of the trend.

Simple Moving Average Decomposition of the Time Series

Time series are typically decomposed into three components: trend, seasonal, and the remainder, or residual. Trend can be modeled by several methods; beginning with a Simple Moving Average Model using the Moving Window Method. The Moving Window Method computes the average timeseries over a specified span, or order of operator.

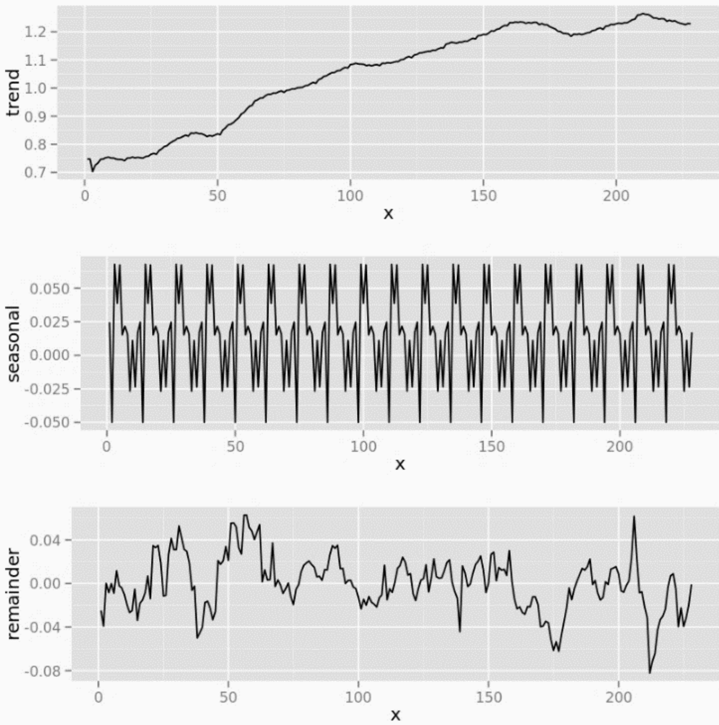
Once the trend has been removed, the seasonal component must be modeled and removed. The Seasonal Component is computed as a function of the month of the year using a linear model.

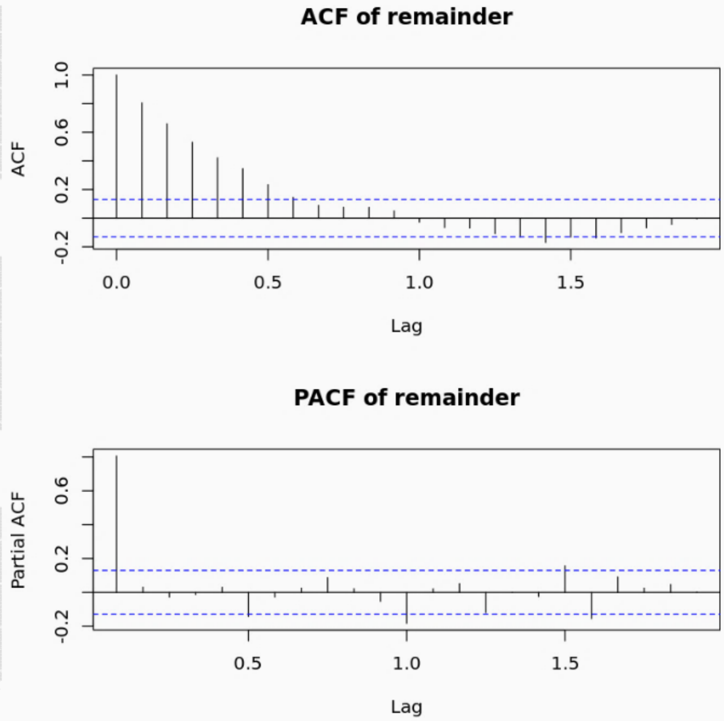
The final step is to take a Multiplicative Decomposition of the timeseries by taking a log of the values.

The resulting data frame has three components for trend, seasonal and remainder.

	trend	seasonal	remainder
1	0.7476354	0.02471208	-0.02471208
2	0.7476354	-0.04996016	-0.03911947
3	0.7030956	0.06783954	9.862991e-05
4	0.7257416	0.03872409	-0.008343715
5	0.7333367	0.06733682	-1.813267e-05
6	0.7468004	0.01530435	-0.008803681

The Decomposed Timeseries data is plotted using **ggplot2** package in R. The trend and seasonal components are clearly separated in the plot on the following page. The remainder plot appears random as expected. However, the remainder will need to be tested if stationary or not, the ACF of the Remainder will determine if the remainder is stationary or not.

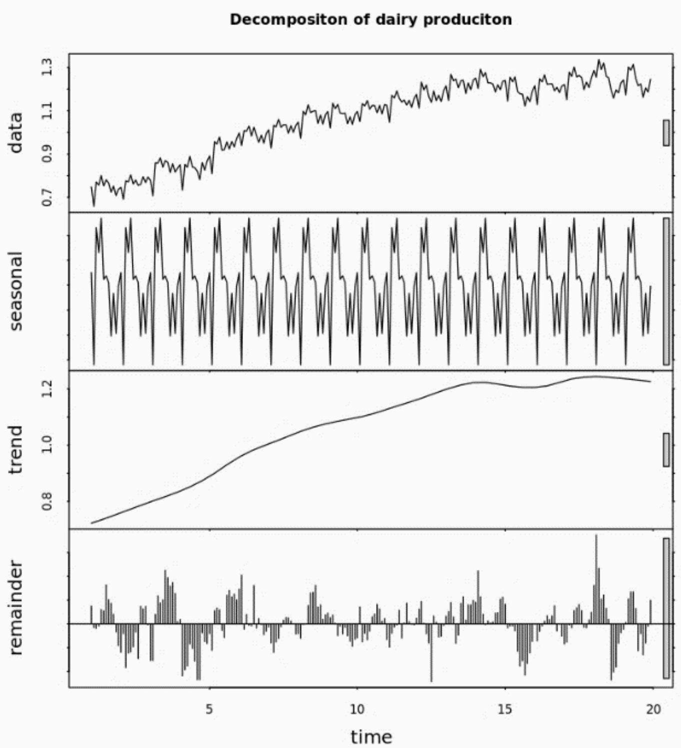


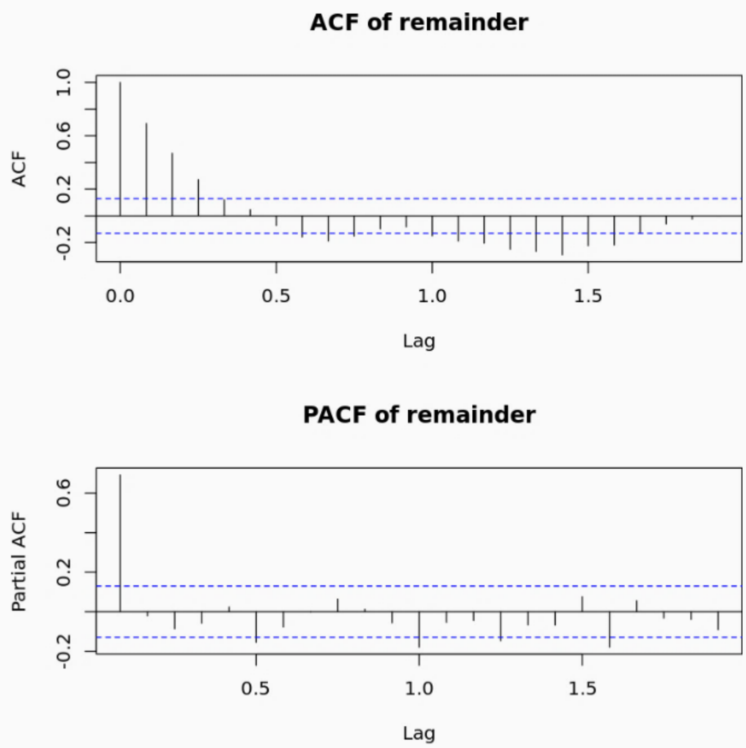


The ACF has 7 significant lag values, indicating the remainder is not, in fact, stationary.

Exploring the Multiplicative Model with Lowess

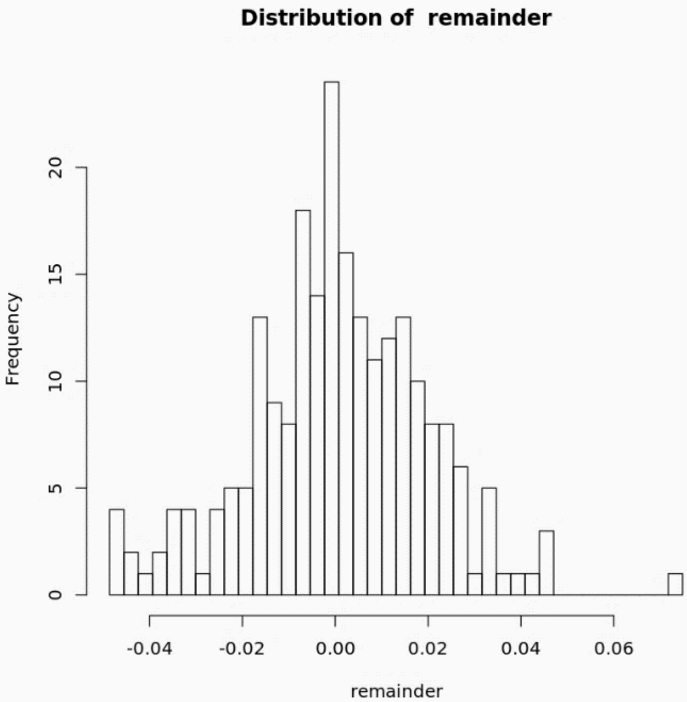
Subsequent to applying an MA model to the data, a Lowess Model will be used to determine the trend. Lowess is a sophisticated non-linear regression. The lowess trend model is combined with a moving window seasonal component model into the R **stl** function. The **stl** function decomposes the time series and the columns of the timeseries decomposition are added to the data frame.





The time series charts show the original time series along with the components of the decomposition. The trend is a bit smoother than was obtained with the simple moving average decomposition. To determine if stationary, the ACF and PCF of the remainder indicate the remainder is still not stationary.

The first 4 lag values of the ACF have significant values, indicating that the remainder series **is not stationary**. Compared to the behavior of the ACF for the simple moving average decomposition, the behavior of the remainder is improved. The Histogram and Box-Plots for the Remainder (non-seasonal residual) Distribution examined next:



The distribution of the remainder values is much closer to a Normal distribution than for the original time series created earlier. This result combined with the ACF plot shows **stl** decomposition effective.

The remainder component shows only limited variation from month to month. The differences are within the interquartile range, indicating that the seasonal model is a reasonably good fit.

Moving Average Models

Subsequent to Decomposition of the Timeseries, the process moves to constructing and testing an Autoregressive Moving Average (ARMA) Model for the Timeseries Remainder; requiring three steps:

- Create a Moving Average Model (MA)
- Create an Autoregressive Model (AR)
- Creating an Autoregressive Moving Average (ARMA) Model

Autoregressive Integrative Moving Average (ARIMA) model:

The summary statistics for the model are printed and the model object returned. By assigning values to the order of each operator, different time series models can be specified: as order of **MA** model, order of **Integrative** model, and order of **AR** model. Since the de-trended remainder is being modeled, the **include.mean** argument is set to FALSE in the **arima** function.

The ACF of the remainder from the **stl** decomposition of the milk production time series had 4 significant lag values. As an initial model, you will now create an **MA** model of order 4. The summary results are on the following page:

```
Call:
arima(x = ts, order = order, include.mean = FALSE)

Coefficients:
         ma1     ma2     ma3     ma4
      0.7259  0.5308  0.2976  0.0193
s.e.  0.0659  0.0776  0.0748  0.0589

sigma^2 estimated as 0.0001876:  log likelihood = 654.34,  aic = -1298.67
```

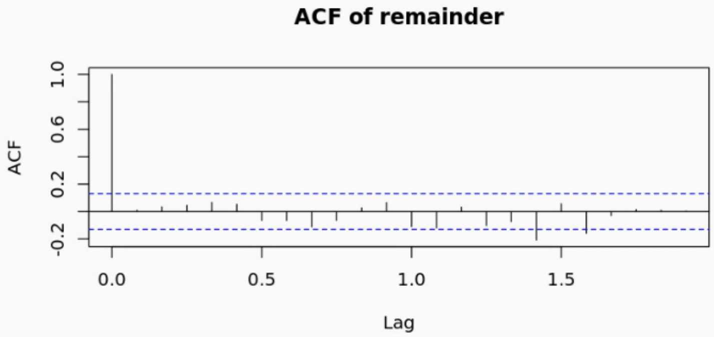
Note the SE of the **ma4** coefficient is > the value of the coefficient itself. This indicates that the value of this coefficient is **poorly determined** and should likely be set to **zero**.

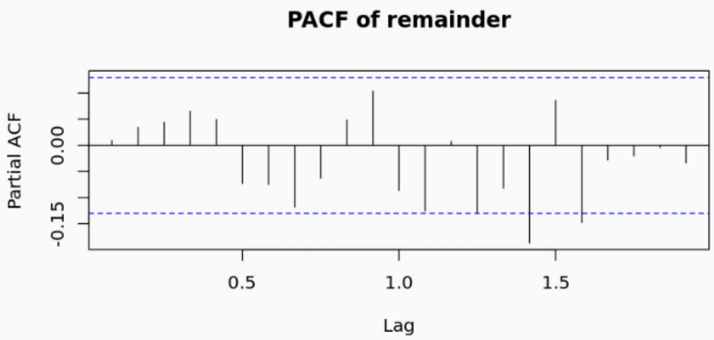
The result indicates that the order of the MA model should be reduced. Generally, the order of an MA model is reduced in unit steps until all the coefficients appear to be significant; an **MA(3)** is run next:

```
Call:
arima(x = ts, order = order, include.mean = FALSE)

Coefficients:
         ma1     ma2     ma3
      0.7224  0.5211  0.2861
s.e.  0.0645  0.0698  0.0660

sigma^2 estimated as 0.0001877:  log likelihood = 654.28,  aic = -1300.56
```

The small standard error compared to the magnitude of the coefficients indicates that the order of the model is reasonable.

To test how well this model fits the data, and produces a stationary result, plot the ACF of the residuals of the MA(3) model:





Note that only the **0 lag** of the ACF is significant and that there are no significant lags for the PACF; This indicates that the MA(3) model is a good fit.

Autoregressive Models (AR)

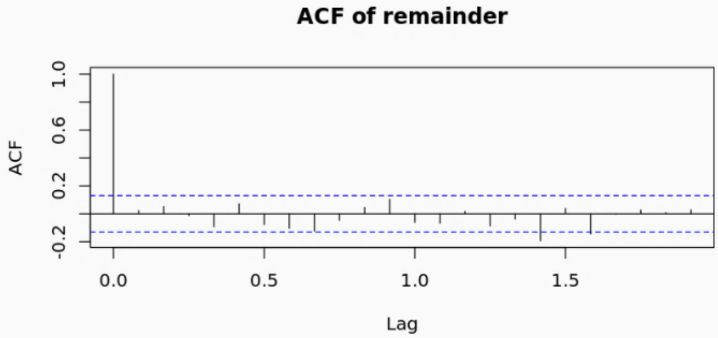
The **MA(3)** model has been shown to be effective. An Autoregressive **(AR)** Model will be tested next. The **PACF** of the reminder indicates that an **AR** model might not be the best choice. None the less, a low order **AR(2)** model might fit the data:

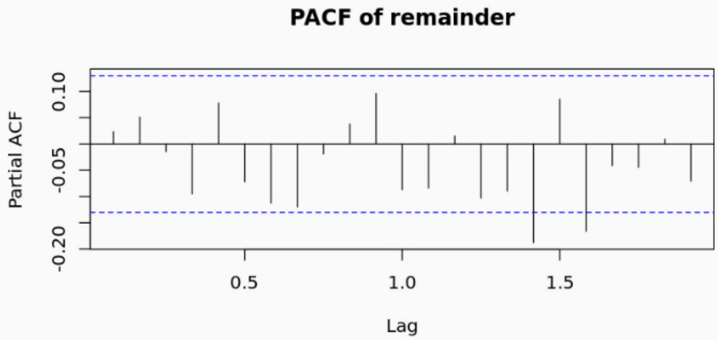
```
Call:
arima(x = ts, order = order, include.mean = FALSE)

Coefficients:
         ar1      ar2
      0.7148  -0.0288
s.e.  0.0665   0.0665

sigma^2 estimated as 0.0001899:  log likelihood = 653.04,  aic = -1300.08
```

Note that the standard error of the second coefficient is of the same magnitude as the first coefficient. The **AR(2)** model is over parameterized; an **AR(1)** will be run next:

```
Call:
arima(x = ts, order = order, include.mean = FALSE)

Coefficients:
         ar1
      0.6946
s.e.  0.0475

sigma^2 estimated as 0.00019:  log likelihood = 652.95,  aic = -1301.9
```

The standard error of the **AR(1)** model is an order of magnitude less than the value of the coefficient, which is promising. The next step is to plot the **ACF** and **PACF** of the **AR(1)** model. The graphs are found on the following page:





Note that only the **0** lag of the **ACF** is significant and that there are **no significant lags** for the **PACF**. These observations indicate that the **AR(1)** model is a good fit. Compare these results to those of the **MA(3)** model, noting that they are nearly identical. Evidently, either the **MA(3)** or **AR(1)** model is a good choice for this data.

Autoregressive Moving Average Models (ARMA)

Both **MA(3)** and **AR(1)** models are good fits to the remainder series; an Autoregressive Moving Average **(ARMA)** model will be tested next on the remainder series; starting with an **ARMA(1,3)** model:

```
Call:
arima(x = ts, order = order, include.mean = FALSE)

Coefficients:
         ar1     ma1     ma2     ma3
      0.1532  0.5750  0.4288  0.2274
s.e.  0.3874  0.3889  0.2588  0.1805

sigma^2 estimated as 0.0001876:  log likelihood = 654.39,  aic = -1298.78
```

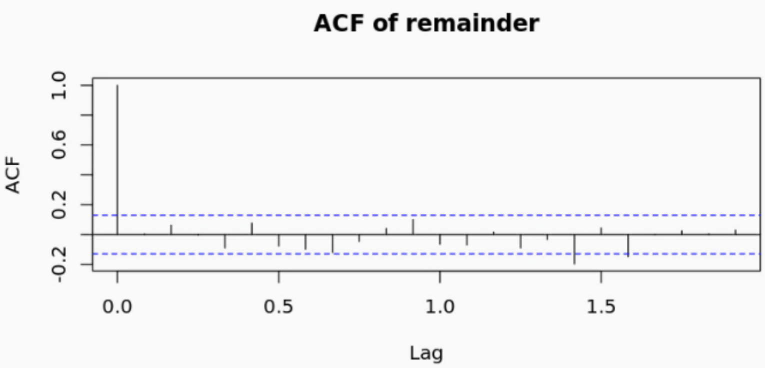
In each case, the standard error is same order of magnitude as the value of the coefficient, indicating this model as a poor fit to the data.

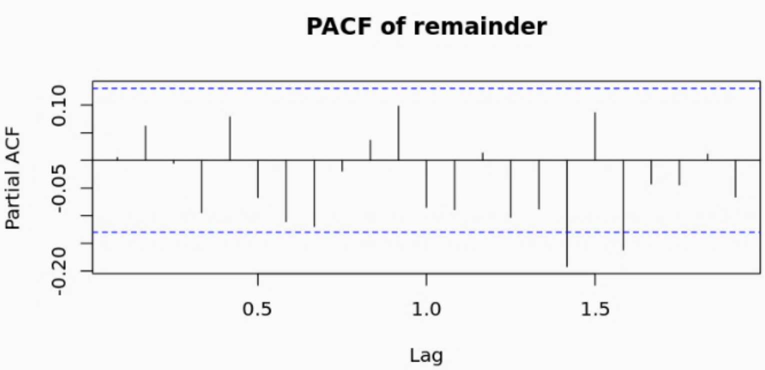
An **ARMA(1)** Model is tested next:

```
Call:
arima(x = ts, order = order, include.mean = FALSE)

Coefficients:
         ar1     ma1
      0.6777  0.0330
s.e.  0.0661  0.0856

sigma^2 estimated as 0.0001899:  log likelihood = 653.02,  aic = -1300.05
```

In both cases, the **AR(1)** and the **MA(1)** are good fits to the data. This is proven when as the **ACF** and the **PACF** indicate **no significant features** outside of **0** for the **ACF**.

Differencing Method follows in the next section.





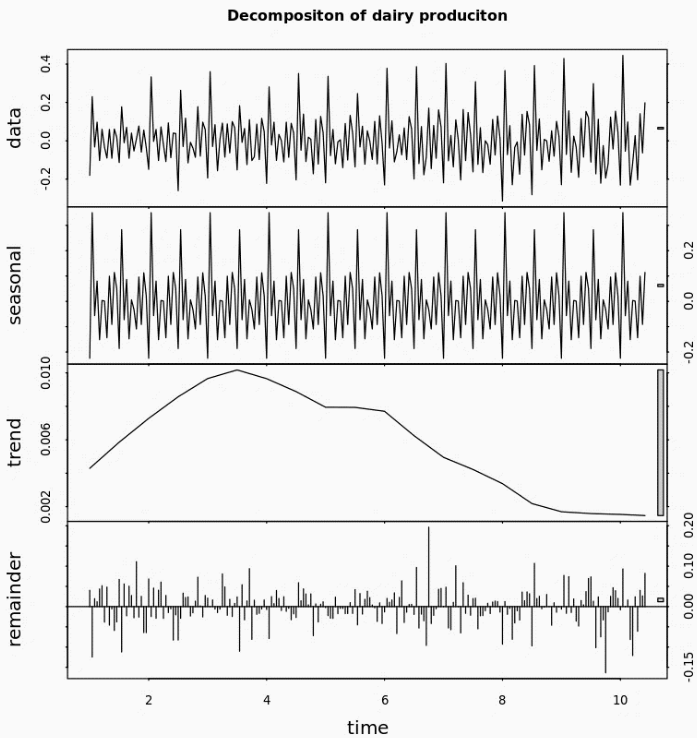
Exploring the Difference Series

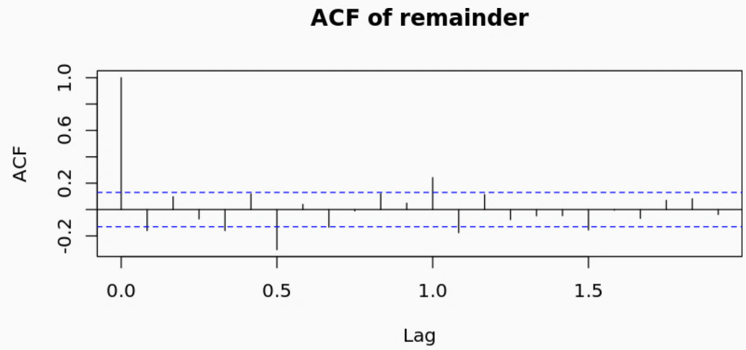
Difference series is a method to remove trend from a time series. The difference can be computed for any number of lag values, depending on the order of the trend. In this case a first order difference series is used to model the trend in the milk production.

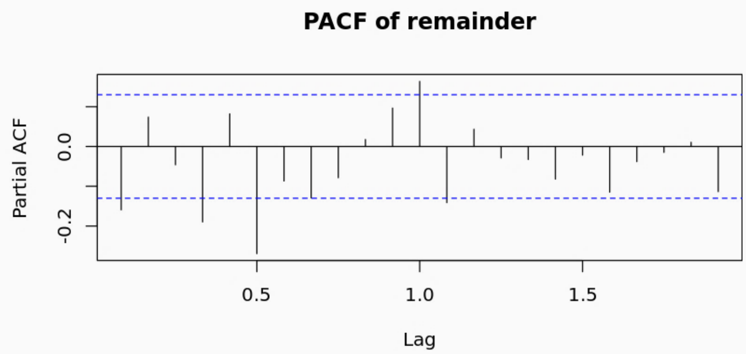
Note the difference series is necessarily of length one less than the original series.

The stl decomposition of the difference series is computed next. Considering working with a difference series, which has positive and negative values, an additive model can be used. No logarithm is taken. The decomposition of the difference series is on the following page:

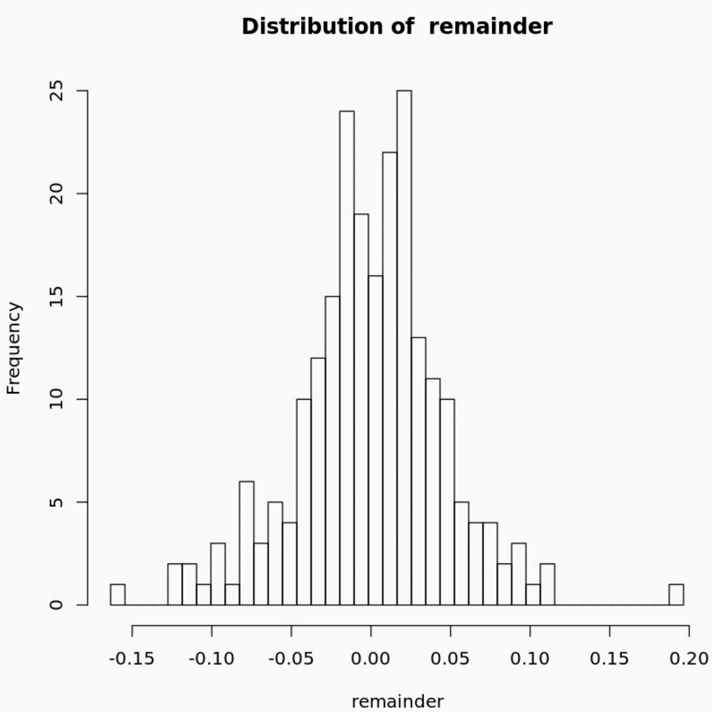
The difference series is shown in the upper most plot. Note the small magnitude of the remaining trend indicating that the first order difference model removed most of the trend. However, the seasonal series exhibits a pattern with a 24 month cycle which is a bit odd.

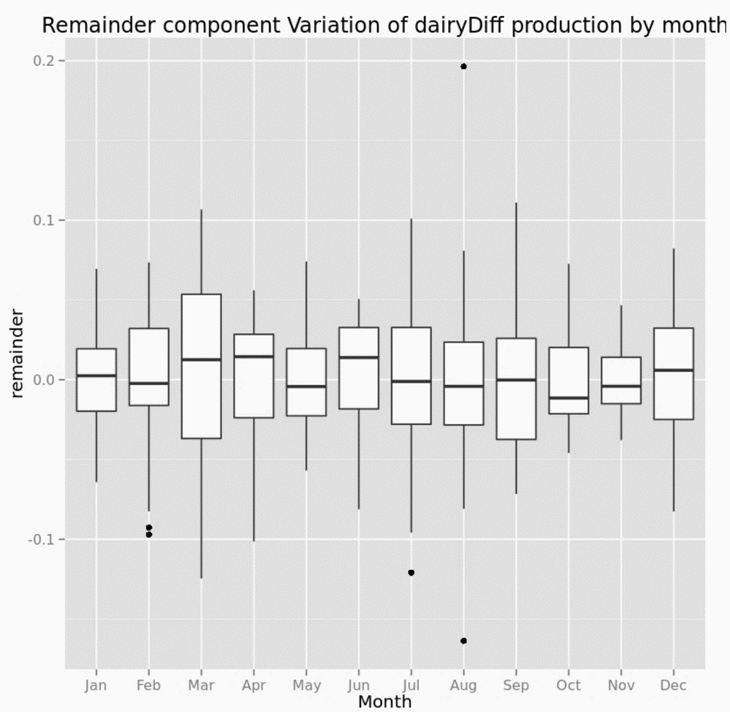






The **ACF** and **PACF** represents a couple of significant features at **0**, **0.5**, and **1.0**. The data does not appear to be stationary, but is close. The distribution looks relatively normal with slight variation.





It is clear from the exploration of the **ARMA** model, that the remainder of the decomposition of the dairy production time series is not stationary.

Autoregressive Integrative Moving Average Model

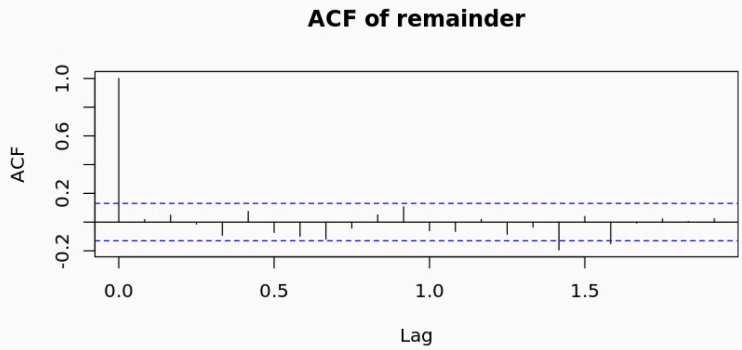
The remainder series is modeled with an autoregressive integrative moving average (ARIMA) model.

An **ARIMA(1, 1, 1)** Model is initially run:

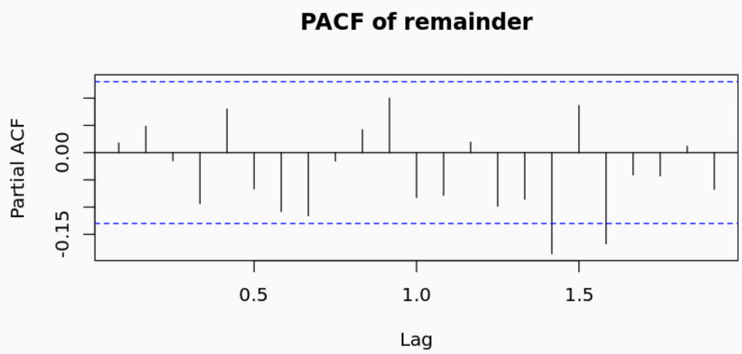
```
Call:
arima(x = ts, order = order, include.mean = FALSE)

Coefficients:
         ar1      ma1
      0.7015  -1.0000
s.e.  0.0482   0.0114

sigma^2 estimated as 0.0001908:  log likelihood = 648.11,  aic = -1290.22
```

The standard error of the **AR1** coefficient is only about half its value. This model seems to be a reasonable fit.

Next, the ACF and PACF of the model are plotted to determine if stationary:



Note that only the **0** lag of the **ACF** is significant and that there are no significant lags for the **PACF**. These observations indicate that the **ARIMA(1,1,1)** model is a good fit. Comparing these results to those of the **MA(3)** and **AR(1)** models, they are nearly identical. The **ARIMA(1,1,1)** model is a good choice for this data as well.



Modeling and Forecasting

After exploring the properties of the Decomposed Timeseries, the forecasts of Dairy Production are computed next. The **R Forecast** package is used to forecast the next 12 months of Diary Production.

The R **forecast** package contains the **auto.arima** function which automatically steps through the **ARIMA** model parameters to find the best fit to the data. The **ARIMA** model used in the **forecast** package also includes modeling of seasonal differences. The **auto.arima** function has multiple arguments, specifying the parameter range values to search. The first argument is a time series object of class **ts**. The code acts as follows:

- Creates a time series of class **ts**.
- Automatically finds and computes an **ARIMA** model.
- Prints a summary of the **ARIMA** model.

```
Series: temp
ARIMA(0,1,1)(0,1,2)[12]

Coefficients:
          ma1     sma1    sma2
      -0.1506  -0.9076  0.1129
s.e.   0.0743   0.0794  0.0838

sigma^2 estimated as 0.0002547:  log likelihood=577.8
AIC=-1147.6   AICc=-1147.41   BIC=-1134.12

Training set error measures:
                        ME       RMSE        MAE         MPE    MAPE      MASE
Training set -0.0003536657 0.01549906 0.01109068 -0.01955938 1.05342 0.2902694
                    ACF1
Training set 0.005145456
```

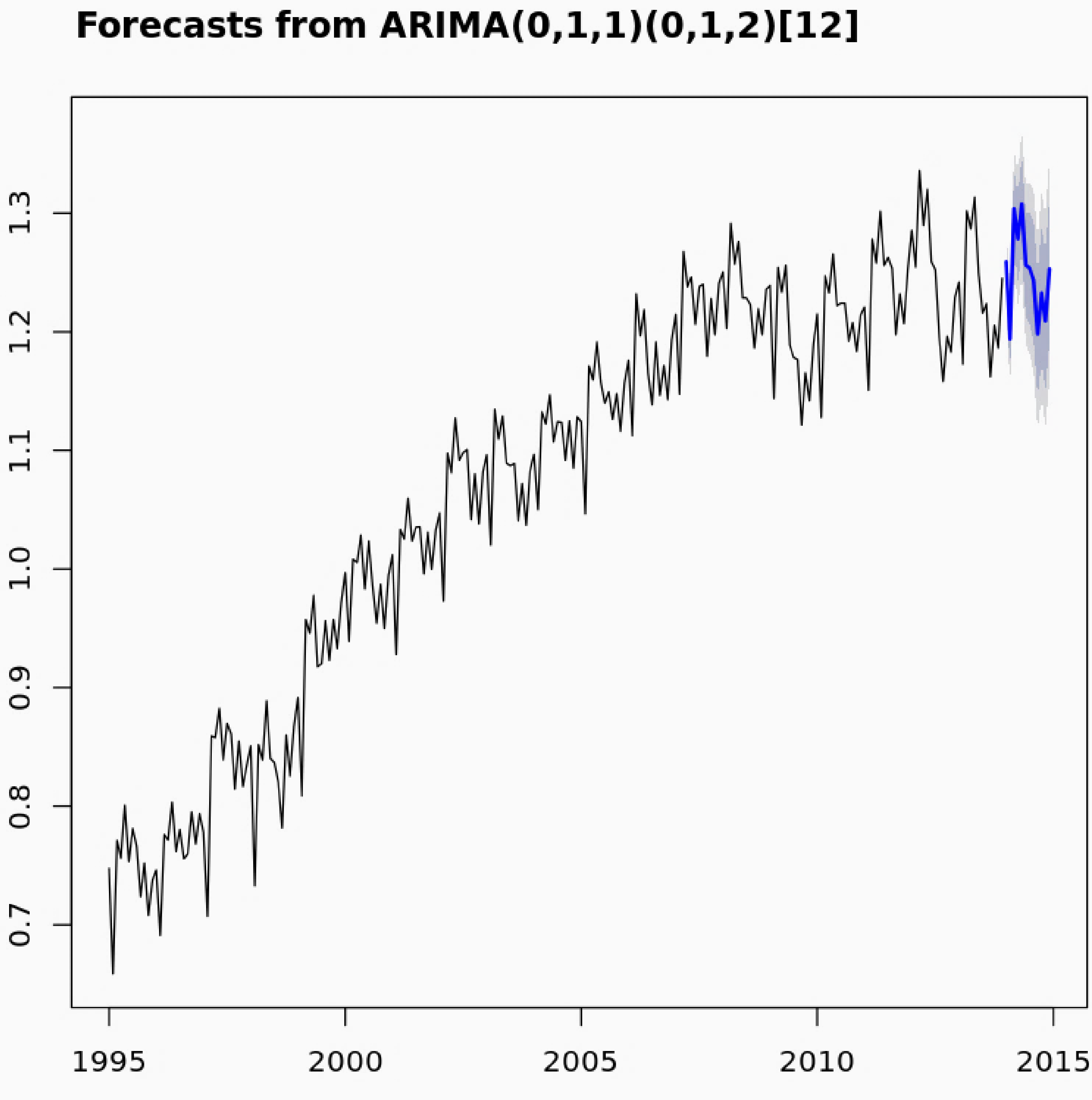
- The model uses an **MA(2)** model for the seasonal difference. The coefficients of this model, **sma1** and **sma2**, along with their standard errors can be seen in the summary.
- The model of the remainder is and **MA(1)** model. The coefficient and its standard error can be seen in the summary above.
- Error metrics, including **RMSE**, are provided in the summary. Notice that the **RMSE** is much smaller than the values of the milk production time series indicating good model performance.

The **forecast** function is used to compute the forecast of the next 12 months using the model created using **auto.arima**:

```
Forecast method: ARIMA(0,1,1)(0,1,2)[12]

Model Information:
Series: temp
ARIMA(0,1,1)(0,1,2)[12]

Coefficients:
          ma1     sma1    sma2
      -0.1506  -0.9076  0.1129
s.e.   0.0743   0.0794  0.0838

sigma^2 estimated as 0.0002547:  log likelihood=577.8
AIC=-1147.6   AICc=-1147.41   BIC=-1134.12

Error measures:
                        ME       RMSE        MAE         MPE    MAPE      MASE
Training set -0.0003536657 0.01549906 0.01109068 -0.01955938 1.05342 0.2902694
                    ACF1
Training set 0.005145456

Forecasts:
         Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
Jan 2014       1.259101 1.238648 1.279555 1.227820 1.290382
Feb 2014       1.193990 1.167154 1.220825 1.152948 1.235031
Mar 2014       1.303803 1.271835 1.335772 1.254912 1.352695
Apr 2014       1.278470 1.242086 1.314854 1.222825 1.334115
May 2014       1.307799 1.267480 1.348118 1.246136 1.369462
Jun 2014       1.256228 1.212325 1.300130 1.189084 1.323371
Jul 2014       1.253456 1.206240 1.300671 1.181246 1.325665
Aug 2014       1.243199 1.192889 1.293509 1.166256 1.320141
Sep 2014       1.198112 1.144887 1.251337 1.116711 1.279512
Oct 2014       1.232666 1.176677 1.288655 1.147039 1.318293
Nov 2014       1.209167 1.150544 1.267789 1.119512 1.298821
Dec 2014       1.252931 1.191789 1.314074 1.159422 1.346440
```

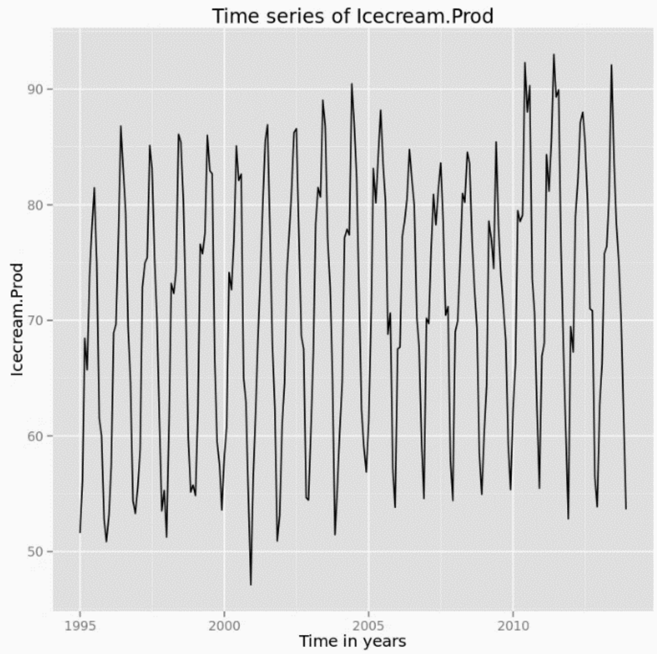
Much of the summary is the same as before. A 12 month forecast is printed below the model summary. There is a point forecast (the expected value) along with 80 and 95 percent confidence intervals. Note, that the confidence intervals generally get wider for forecasts further out in time. It is not surprising that the forecast has more uncertainty as time increases from the present.

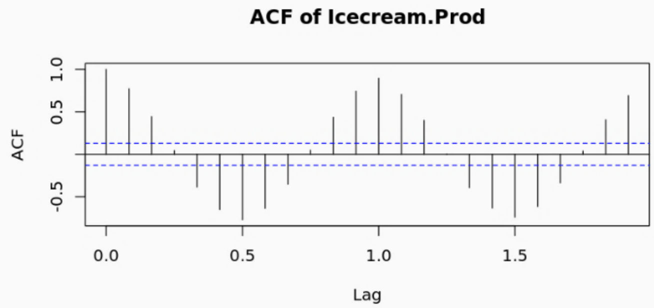
The forecast is plotted on the following page:

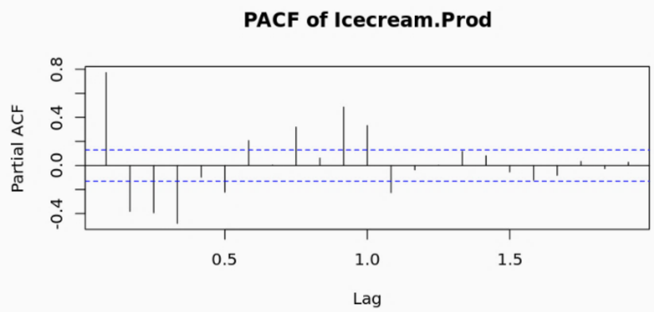


The original time series of milk production is shown in black in the plot above. The forecast is shown in **Blue**. The 80 and 95 percent confidence intervals are shown in lighter shades of **blue-gray**.

Plotting Ice Cream Production

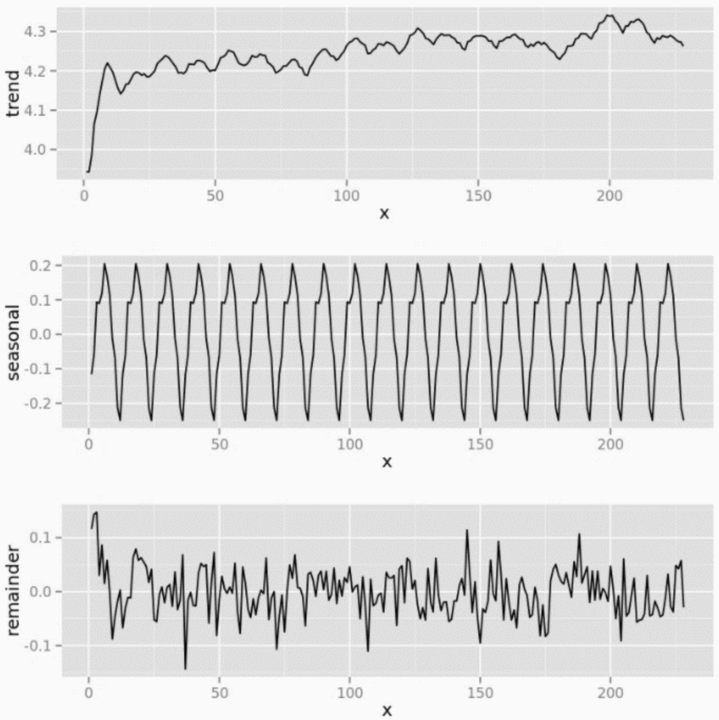


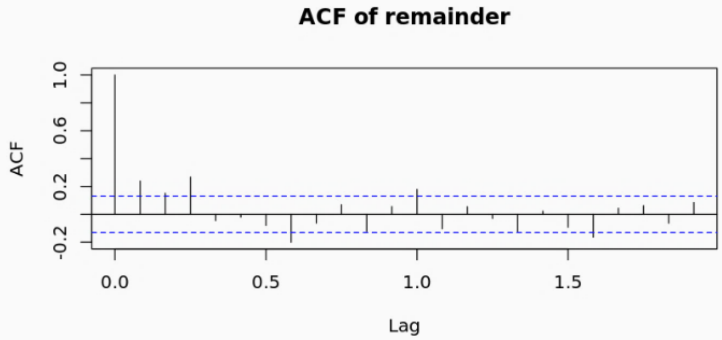


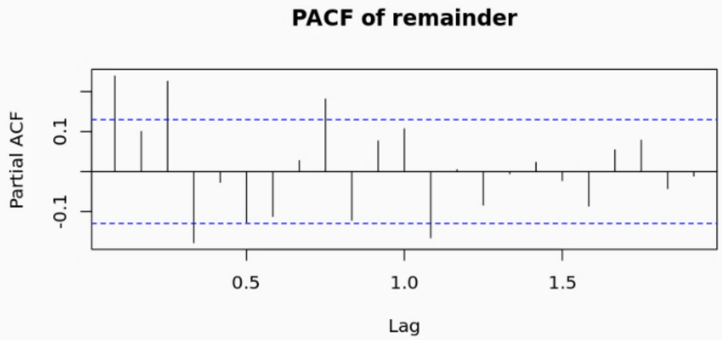


Simple Moving Average Decomposition of the Time Series

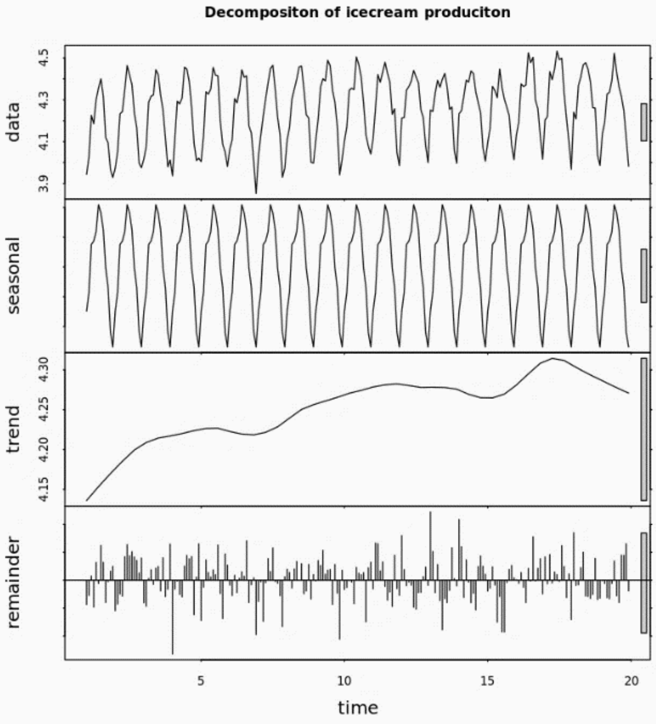
	trend	seasonal	remainder
1	3.943425	-0.1161525	0.1161525
2	3.943425	-0.05987141	0.1433329
3	3.985155	0.09401897	0.1469729
4	4.065486	0.08971517	0.03023243
5	4.095473	0.1186856	0.08625111
6	4.13646	0.205117	0.01505456

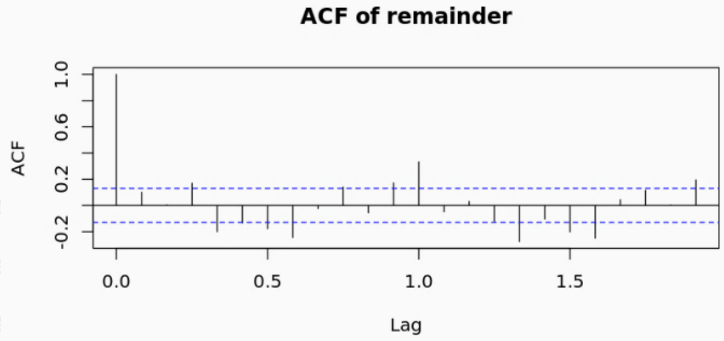


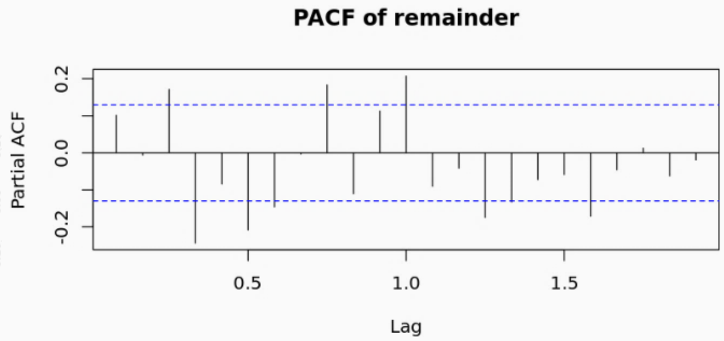


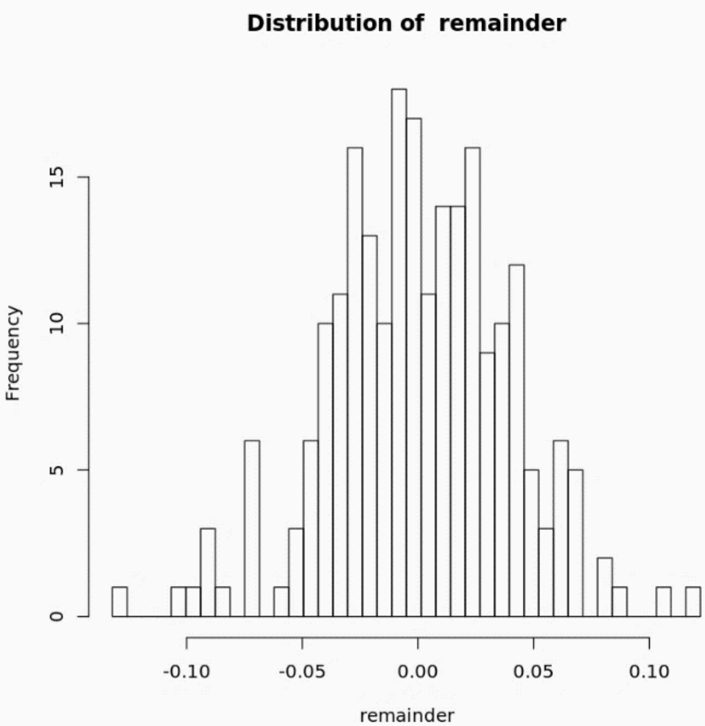


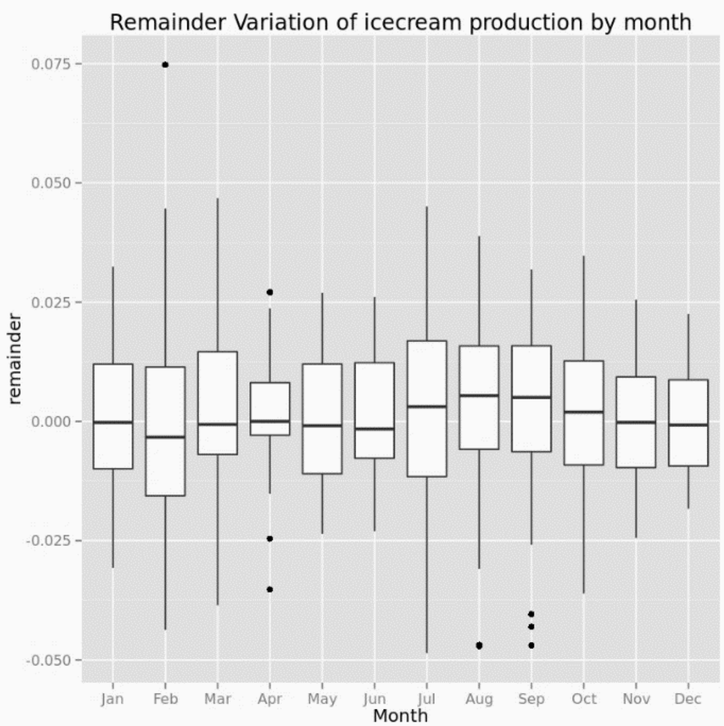
Exploring the Multiplicative Model with Lowess











Modeling and Forecasting Ice cream Production

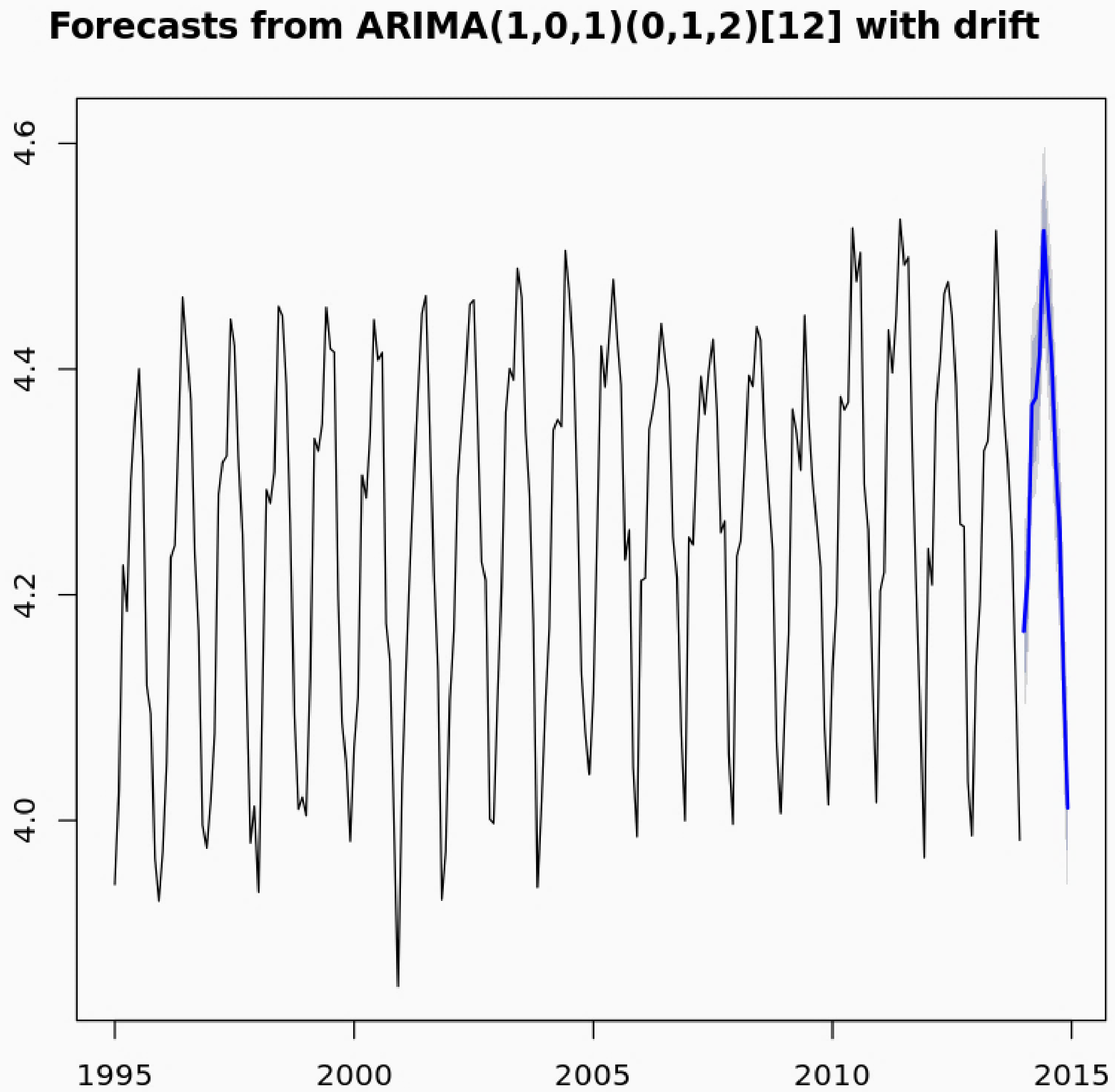
The forecast the production of **icecream** for a 12 month period following these steps.

- Create a time series object with **frequency = 12** and **start = 1995** from the **Icecream.Prod** column of the **dairy** data frame.
- Fit a model with the **auto.arima** function, following the hint given below.
- Print a summary of the model. What is the order of the **MA** and **AR** components of the seasonal and remainder models? Note there will be a drift term, which accounts for linear trend in the time series.
- Compute the forecast of icecream production for the next 12 months.
- Plot the forecast and note the behavior.

The **ts**, **auto.arima**, **summary**, **forecast**, and **plot** functions are used to create the new model, the summaries are printed to make the plots. **Hint** use the time series of icecream production as the first argument to the **auto.arima** function. The other arguments from the milk production model can be copied and pasted, but **max.p = 1** to prevent having an over-parameterized model.

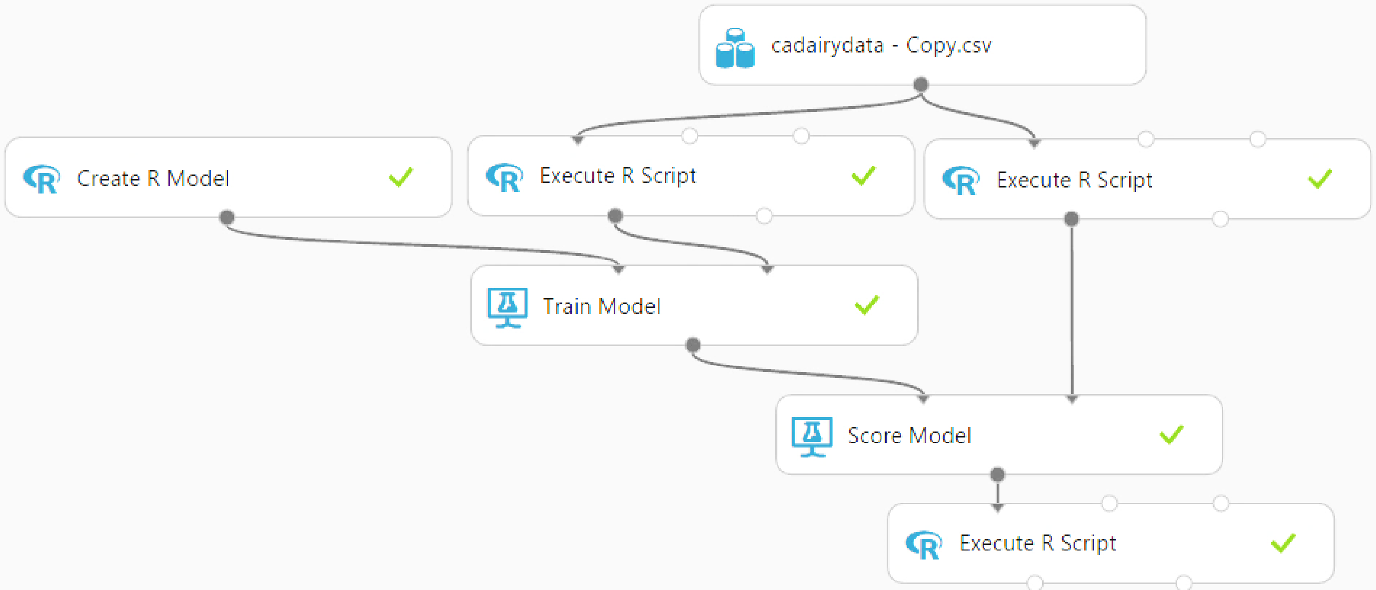
```
Series: tempIC
ARIMA(1,0,1)(0,1,2)[12] with drift

Coefficients:
         ar1      ma1     sma1     sma2  drift
      0.8676  -0.6761  -0.5038  -0.2193  6e-04
s.e.  0.0696   0.0955   0.0694   0.0663  2e-04

sigma^2 estimated as 0.001759:  log likelihood=359.75
AIC=-707.5   AICc=-707.1   BIC=-687.25

Training set error measures:
                      ME       RMSE        MAE        MPE      MAPE      MASE
Training set 0.001246958 0.03967756 0.03089997 0.02618669 0.7287523 0.7882469
                    ACF1
Training set -0.03374896
```

```
Forecast method: ARIMA(1,0,1)(0,1,2)[12] with drift

Model Information:
Series: tempIC
ARIMA(1,0,1)(0,1,2)[12] with drift

Coefficients:
         ar1      ma1     sma1     sma2  drift
      0.8676  -0.6761  -0.5038  -0.2193  6e-04
s.e.  0.0696   0.0955   0.0694   0.0663  2e-04

sigma^2 estimated as 0.001759:  log likelihood=359.75
AIC=-707.5   AICc=-707.1   BIC=-687.25

Error measures:
                      ME       RMSE        MAE        MPE      MAPE      MASE
Training set 0.001246958 0.03967756 0.03089997 0.02618669 0.7287523 0.7882469
                    ACF1
Training set -0.03374896

Forecasts:
         Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
Jan 2014       4.167588 4.113846 4.221330 4.085397 4.249780
Feb 2014       4.216966 4.162247 4.271684 4.133281 4.300651
Mar 2014       4.368172 4.312730 4.423615 4.283381 4.452964
Apr 2014       4.374599 4.318619 4.430580 4.288984 4.460215
May 2014       4.412652 4.356269 4.469035 4.326422 4.498883
Jun 2014       4.522673 4.465990 4.579357 4.435983 4.609364
Jul 2014       4.459753 4.402844 4.516662 4.372718 4.546788
Aug 2014       4.408627 4.351549 4.465705 4.321333 4.495920
Sep 2014       4.320613 4.263408 4.377818 4.233125 4.408100
Oct 2014       4.256768 4.199468 4.314069 4.169135 4.344402
Nov 2014       4.117332 4.059960 4.174704 4.029589 4.205075
Dec 2014       4.010979 3.953553 4.068405 3.923154 4.098805
```



Forecasting in Microsoft AzureML

DAT203.3x: Milk Production Forecast



Statistics

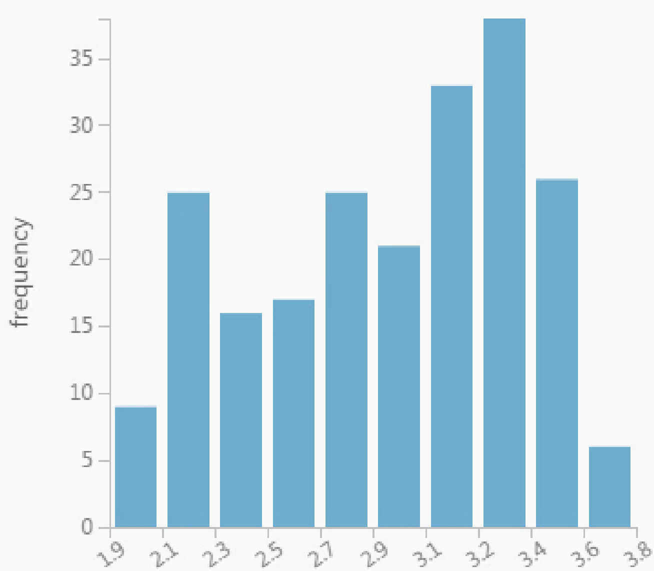
Mean	2.9199
Median	3.002
Min	1.932
Max	3.804
Standard Deviation	0.4742
Unique Values	205
Missing Values	0
Feature Type	Numeric Feature

Visualizations

Milk.Prod

Histogram

compare to None



Standard Output

RWorker pushed "port1" to R workspace.
Beginning R Execute Script

[1] 56000
Loading objects:
  port1
[1] "Loading variable port1..."

Standard Error

R reported no errors.

Statistics

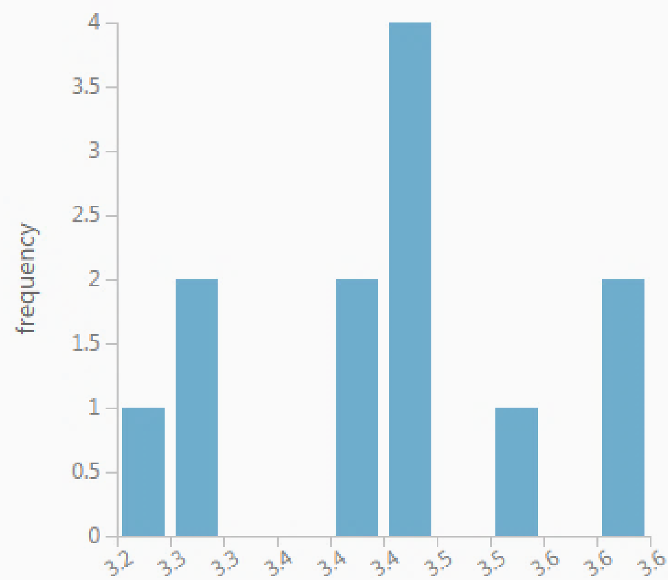
Mean	3.4477
Median	3.464
Min	3.2465
Max	3.637
Standard Deviation	0.1184
Unique Values	12
Missing Values	0
Feature Type	Numeric Feature

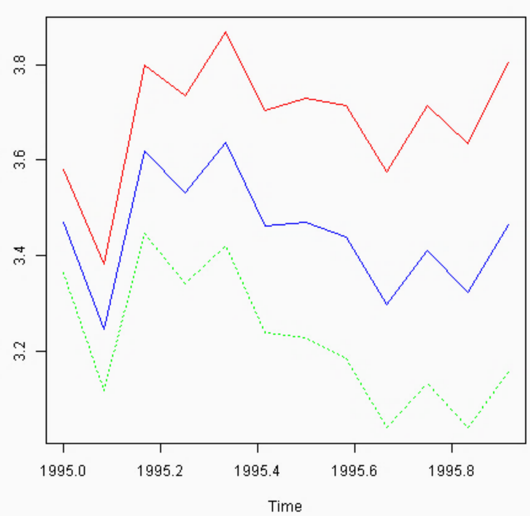
Visualizations

forecast

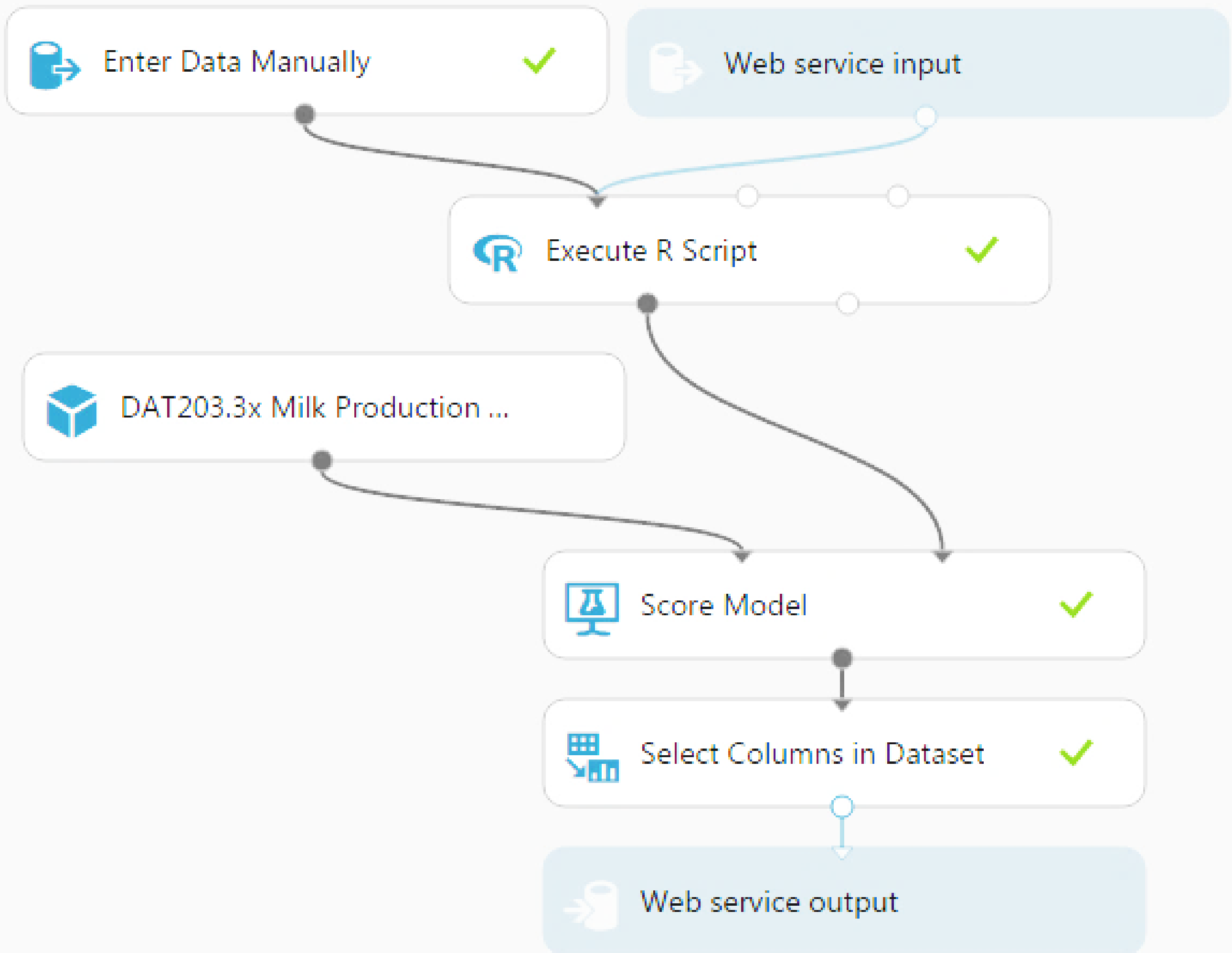
Histogram

compare to None





DAT203.3x: Milk Production Forecast [Predictive Exp.]



Summary

This lab worked with and analyzed time series data.

Specifically the following:

- Examined the properties of time series objects.
- Plotted time series data.
- Decomposed time series data into its trend, seasonal, and remainder components.
- Modeled the remainder components as AR, MA, ARMA and ARIMA models.
- Created and evaluated difference series methods.
- Constructed and evaluated a forecasting model.

forecast	upper95	lower95
3.469836	3.580358	3.362726
3.246524	3.382647	3.115879
3.619187	3.800553	3.446476
3.532431	3.734513	3.341284
3.636989	3.868202	3.419597
3.463174	3.703532	3.238416
3.469722	3.729337	3.22818
3.439218	3.71405	3.184723
3.296013	3.575274	3.038566
3.410717	3.715334	3.131075
3.323451	3.634856	3.038725
3.464765	3.804016	3.155769