Time Series Analysis

Tejovardhan Medamarti

March 7, 2017

Overview and summary

Our objective in this assignment is to perform time series analysis on the data for one of Milk Production, or Ice Cream Production, in the CADairyProduction.csv.

We have chosen to analyse the data for the ice cream Production.

file to answer the following questions

- Is this time series stationary?
- Is there a significant seasonal component?
- For the residual from the STL decomposition of the time series what is the order of the ARMA(p,q) process that best fits?
- Forecast production for 12 months. Report both numeric values and plot the confidence intervals. Are the confidence intervals reasonably small compared to the forecast means? How do the confidence intervals behave as time moves to the future?

Note: Following packages are required to run the below packages. - forecast - repr

```
require(forecast)
require(repr)
```

Data staging:

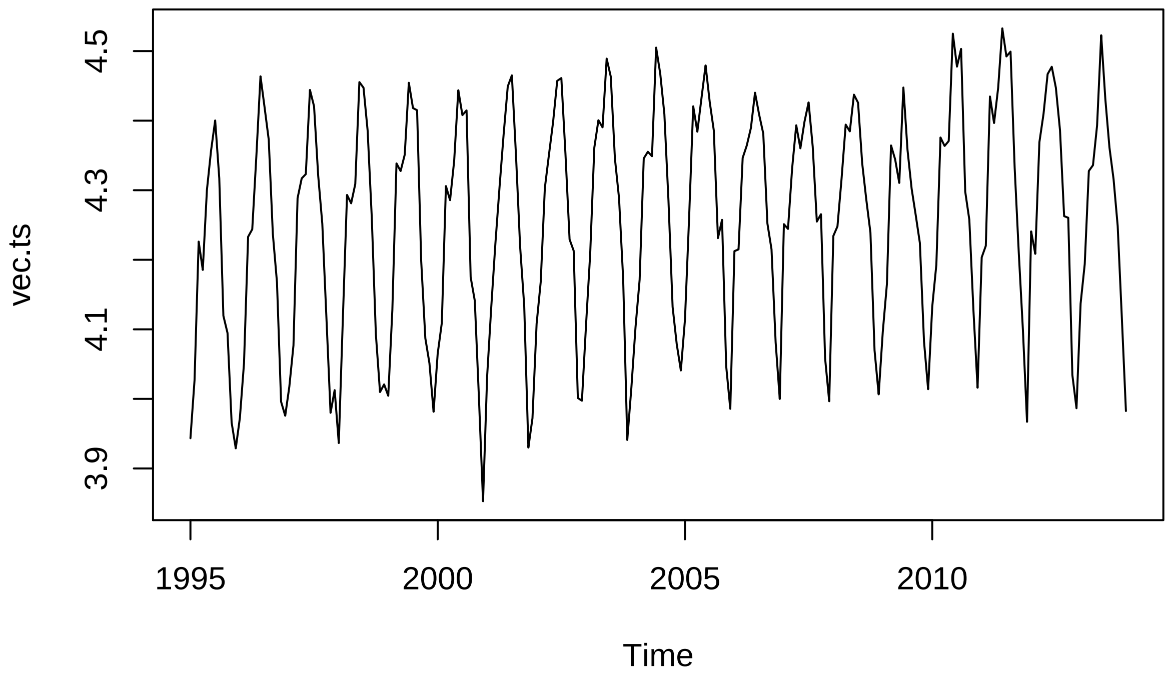
```
read.CADairy = function(file = "C:\\Tejo\\Datascience\\350\\Classwork\\DSClasswork\\Lecture8\\Assignment
    ## Read the csv file
    dairy.produce <- read.csv(file, header = TRUE,
                              stringsAsFactors = FALSE)

  }

ca.dairy = read.CADairy()
```

Since we are dealing only with the Icecream prod feature, lets create a basic time series vector vec.ts which only contains Icecream prod feature.

```
vec.ts = with(ca.dairy, log(ts(Icecream.Prod, start = c(1995, 1), freq = 12)))
attributes(vec.ts) # Note the time series attributes
```

```
## $tsp
## [1] 1995.000 2013.917   12.000
##
## $class
## [1] "ts"
```

```
options(repr.pmales.extlot.width=8, repr.plot.height=4)
plot(vec.ts) # Note the x-axis is the time attribute
```

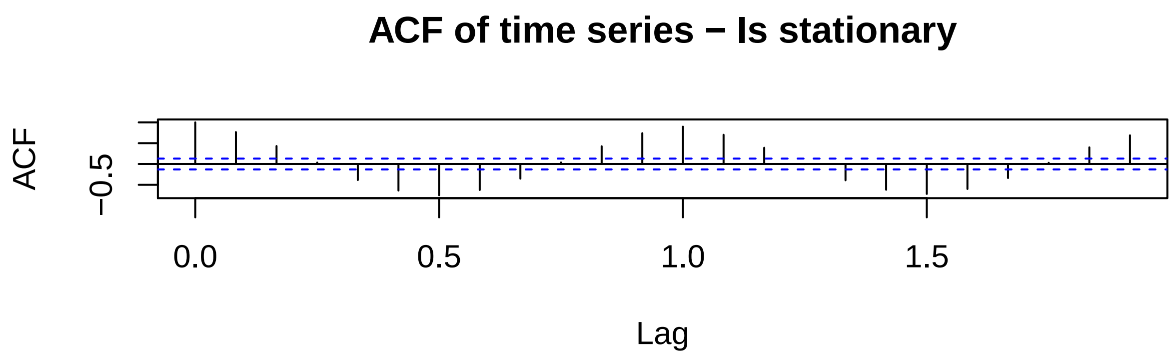


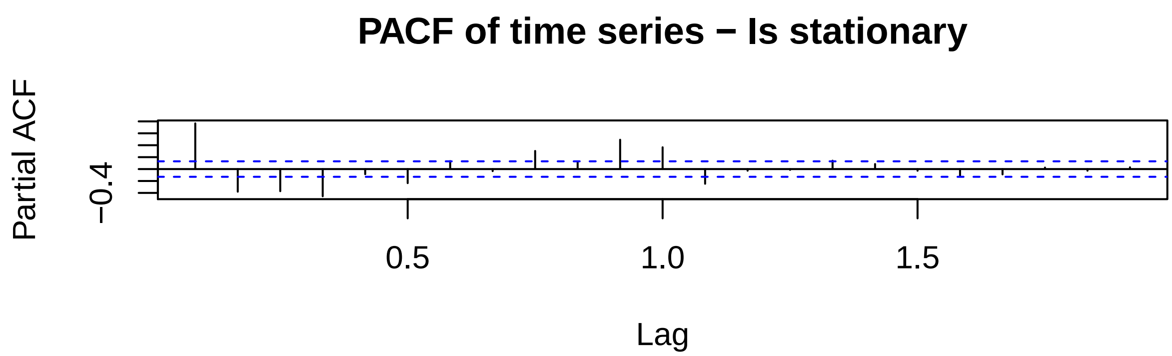
Lets create a helper functions for the visualization purpose.

```
plot.acf <- function(df, col = 'remainder', is.df =TRUE){
  if(is.df) temp <- df[, col]
  else temp <- df
  par(mfrow = c(2,1))
  acf(temp, main = paste('ACF of', col))
  pacf(temp, main = paste('PACF of', col))
  par(mfrow = c(1,1))
}

## Function for ARIMA model estimation
ts.model = function(ts, col = 'remainder', order = c(0,0,1)){
  mod = arima(ts, order = order, include.mean = FALSE)
 print(mod)
  mod
}
```

Q1: Is this time series stationary?

- Plot 1:





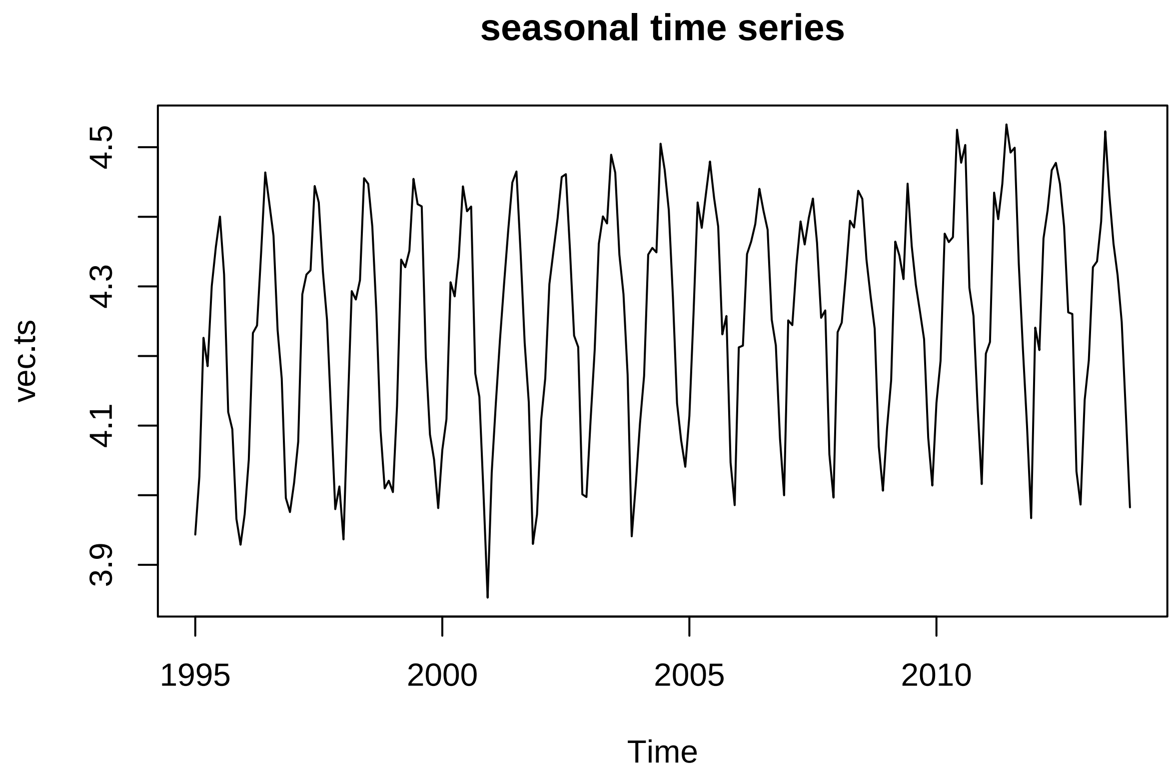

```r
Box.test(vec.ts, lag =20, type="Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  vec.ts
## X-squared = 1475, df = 20, p-value < 2.2e-16
```

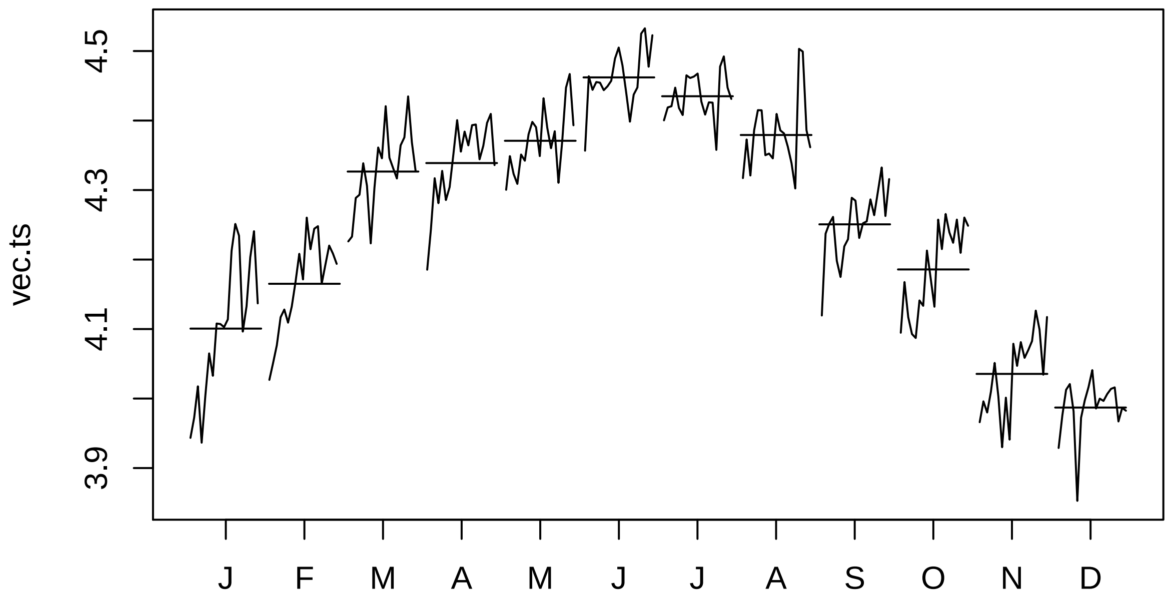
Result: Based on the above results, ACF and PACF are dying slowly and also the LJung-box test reveals the ts is stationary based on the p-value.Hence the **time series is stationary**.

Q2: Is there a significant seasonal component?? #### - Plot 2:

```r
plot(vec.ts, main = 'seasonal time series')
```



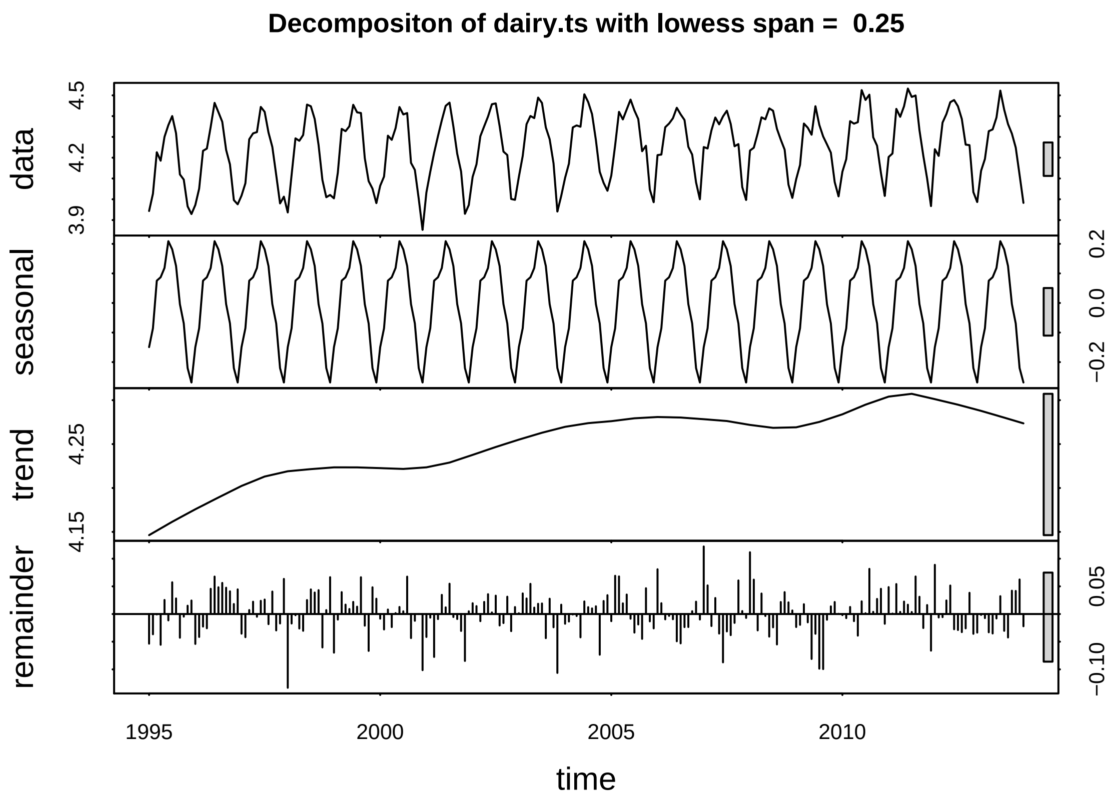

```
monthplot(vec.ts)
```



Result: Based on the above results, **time series is a significant seasonal component**.

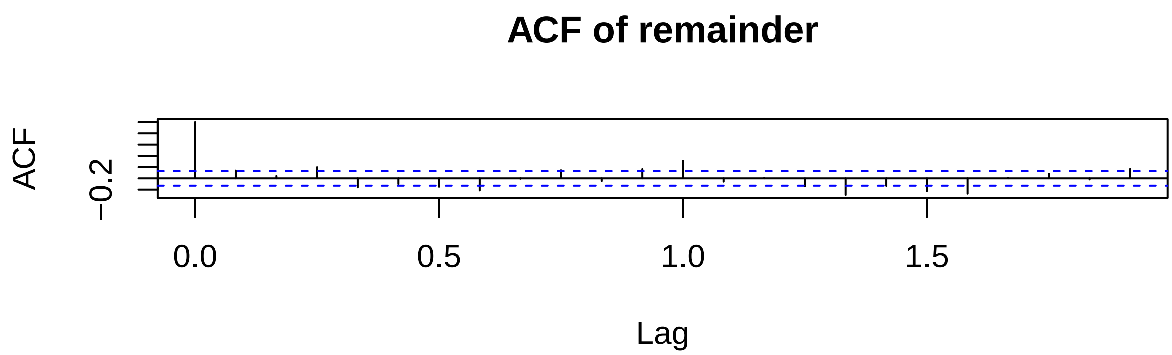
Q3: For the residual from the STL decomposition of the time series what is the order of the ARMA(p,q) process that best fits?

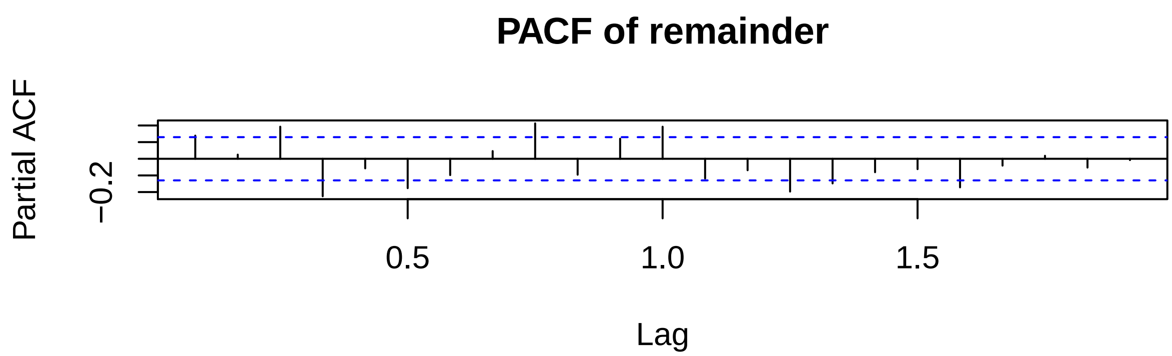
```
## Decomposition of the time series into components
ts.decomp <- function(df, col = 'dairy.ts', span = 0.25, Mult = TRUE, is.df = TRUE){
  # if(Mult) temp = log(df[, col])  else temp = ts(df[, col]
  if(is.df) temp = log(df[, col])
  else temp = df
  spans = span * length(temp)
  fit <- stl(temp, s.window = "periodic", t.window = spans)
  plot(fit, main = paste('Decompositon of',col,'with lowess span = ', as.character(span)))
  fit$time.series
}

temp = ts.decomp(vec.ts, is.df = FALSE, Mult = FALSE)
```



- Plot 3:

```
plot.acf(temp[,3],is.df=FALSE)
```

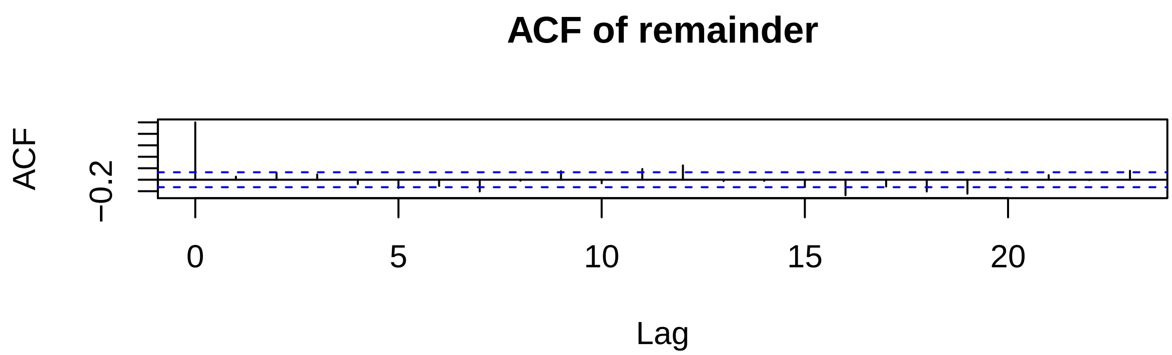


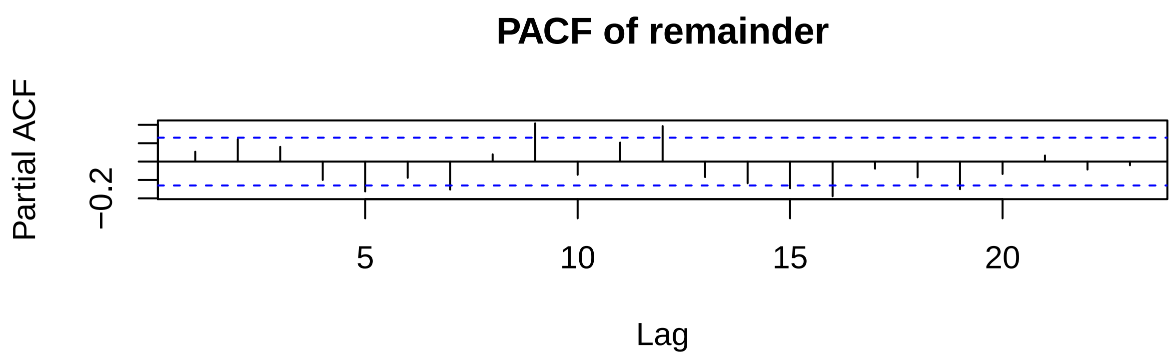



```
iceCream.arima = ts.model(temp[, 3], col = 'ARIMA model for icecream production', order = c(2,0,2))##
```

```
##
## Call:
## arima(x = ts, order = order, include.mean = FALSE)
##
## Coefficients:
##           ar1      ar2     ma1     ma2
##       -1.6287  -0.7684  1.7825  0.8681
## s.e.   0.0967   0.0852  0.0900  0.0823
##
## sigma^2 estimated as 0.001481:  log likelihood = 418.91,  aic = -827.83
```

```
plot.acf(iceCream.arima$resid[-1],is.df=FALSE)
```





Result: Based on the above results, we are getting best fit at order c(2,0,2) for the ARMA(p,q) process.

Q4: Forecast production for 12 months. Report both numeric values and plot the confidence intervals.

- Are the confidence intervals reasonably small compared to the forecast means?
- How do the confidence intervals behave as time moves to the future?

```
fit.icecream = auto.arima(vec.ts, max.p=3, max.q=3,
                          max.P=2, max.Q=2, max.order=5, max.d=2, max.D=1,
                          start.p=0, start.q=0, start.P=0, start.Q=0)
summary(fit.icecream)
```

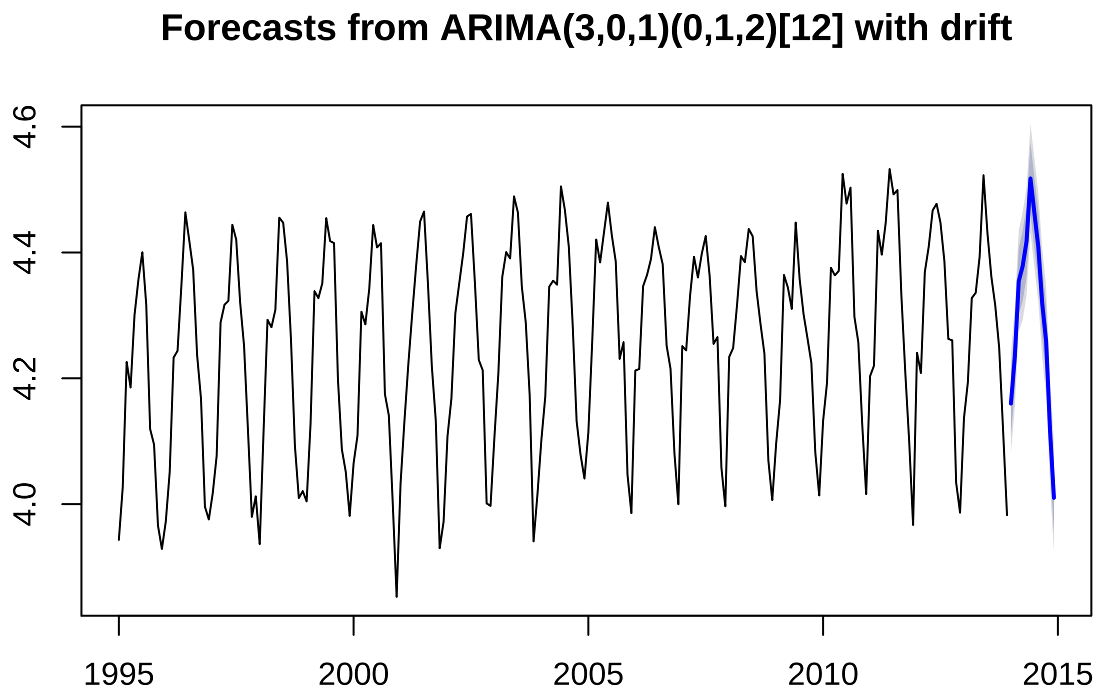
```
## Series: vec.ts
## ARIMA(3,0,1)(0,1,2)[12] with drift
##
## Coefficients:
##           ar1     ar2     ar3     ma1     sma1     sma2  drift
##       -0.1996  0.1658  0.3746  0.4086  -0.5043  -0.2039  6e-04
## s.e.   0.1767  0.0755  0.0636  0.1925   0.0699   0.0672  2e-04
##
## sigma^2 estimated as 0.001583:  log likelihood=389.7
## AIC=-763.4   AICc=-762.7   BIC=-736.4
##
## Training set error measures:
##                       ME       RMSE        MAE        MPE      MAPE
## Training set 0.001360524 0.03809705 0.02972694 0.02906281 0.7001261
##                   MASE         ACF1
```

```
## Training set 0.7583233 -0.004410837
```

```r
icecream.forecast = forecast(fit.icecream, h=12)
summary(icecream.forecast)
```

```
##
## Forecast method: ARIMA(3,0,1)(0,1,2)[12] with drift
##
## Model Information:
## Series: vec.ts
## ARIMA(3,0,1)(0,1,2)[12] with drift
##
## Coefficients:
##          ar1     ar2     ar3     ma1     sma1     sma2  drift
##      -0.1996  0.1658  0.3746  0.4086  -0.5043  -0.2039  6e-04
## s.e.  0.1767  0.0755  0.0636  0.1925   0.0699   0.0672  2e-04
##
## sigma^2 estimated as 0.001583:  log likelihood=389.7
## AIC=-763.4   AICc=-762.7   BIC=-736.4
##
## Error measures:
##                       ME       RMSE        MAE        MPE      MAPE
## Training set 0.001360524 0.03809705 0.02972694 0.02906281 0.7001261
##                   MASE         ACF1
## Training set 0.7583233 -0.004410837
##
## Forecasts:
##          Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## Jan 2014       4.159970 4.108976 4.210965 4.081981 4.237960
## Feb 2014       4.233898 4.181801 4.285995 4.154223 4.313574
## Mar 2014       4.354554 4.302074 4.407033 4.274293 4.434814
## Apr 2014       4.378331 4.322309 4.434353 4.292653 4.464010
## May 2014       4.417831 4.361797 4.473864 4.332135 4.503527
## Jun 2014       4.517751 4.461459 4.574044 4.431659 4.603843
## Jul 2014       4.462807 4.406146 4.519469 4.376151 4.549463
## Aug 2014       4.409070 4.352409 4.465731 4.322414 4.495726
## Sep 2014       4.319325 4.262580 4.376070 4.232540 4.406109
## Oct 2014       4.259274 4.202501 4.316048 4.172446 4.346103
## Nov 2014       4.116758 4.059984 4.173532 4.029930 4.203587
## Dec 2014       4.010205 3.953413 4.066997 3.923350 4.097061
```

```r
plot(icecream.forecast)
```



Conclusion:

Based on the above results, Confidence intervals are relatively small when compared to the forecasted means and they behave as the time moves.