ARIMAX Modeling: Ice Cream Consumption

Andira Putri

In this project, we do ten exercises from: https://www.r-bloggers.com/forecasting-arimax-model-exercises-part-5/. We will be working on the `Icecream` data set from the `Ecdat` package. The data set is also downloadable as a `.csv` file on the website. To use the data set straight from the `Ecdat` package, uncomment the four lines of code with `#` in front, and comment out `Icecream=read.csv("Icecream.csv")`.

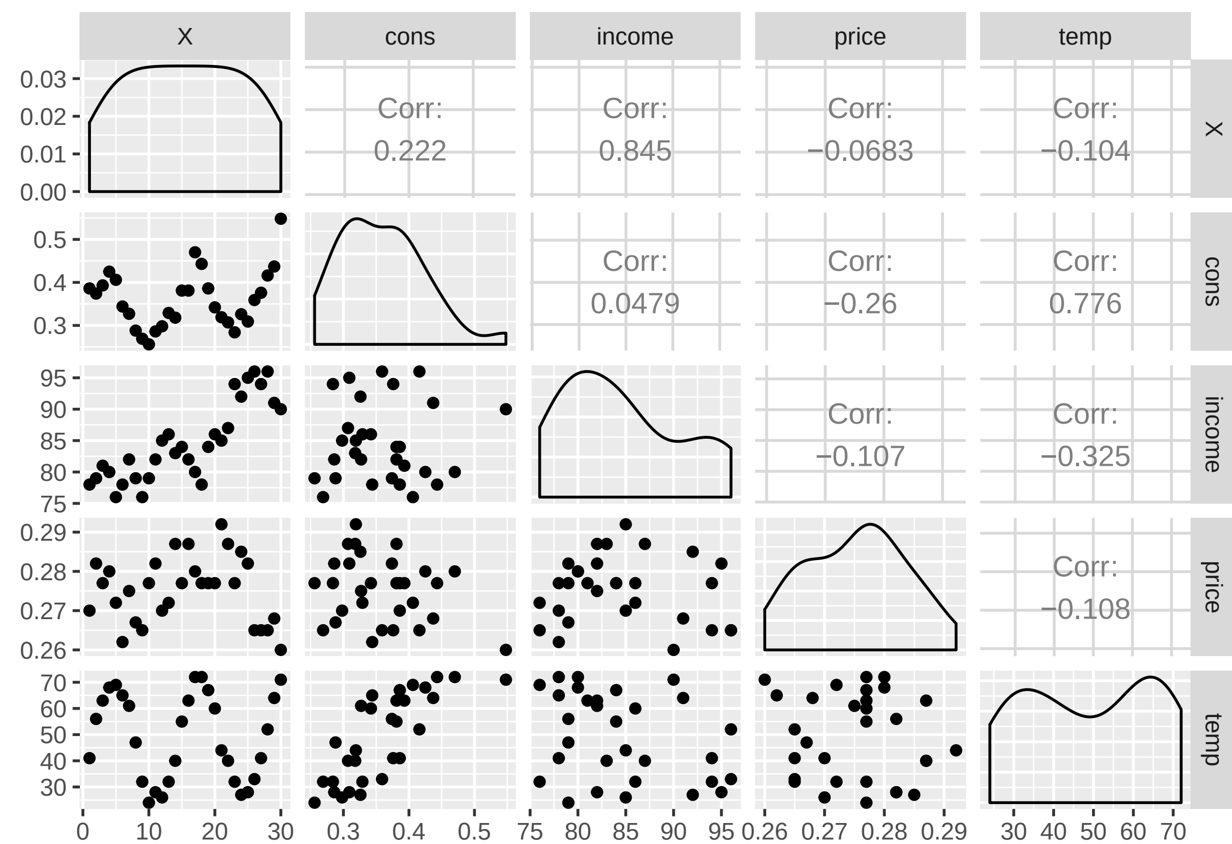
```
#library(Ecdat)
#data(Icecream)
Icecream=read.csv("Icecream.csv")
```

Exercise 1. Load the dataset, and plot the variables `cons` (ice cream consumption), `temp` (temperature), and `income`.

```
#t=1:30 #time values
#df=cbind(t,Icecream) #add time values to dataset
library(GGally)
```

```
## Loading required package: ggplot2
```

```
ggpairs(Icecream)
```



We are not paying attention to the correlation values given here- when working with time series data, we are interested in the autocorrelations.

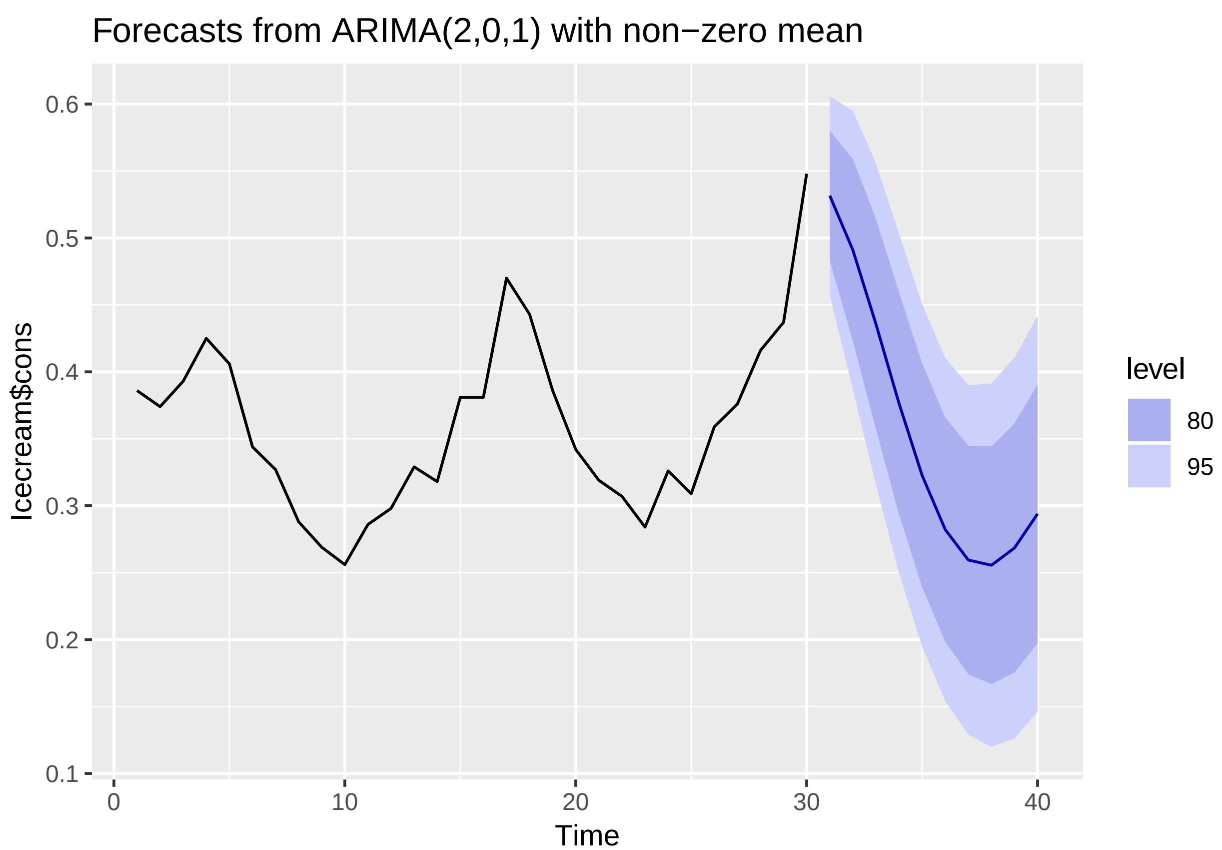
Exercise 2. Estimate an ARIMA model for the data on ice cream consumption using the auto.arima function. Then pass the model as input to the `forecast` function to get a forecast for the next 6 periods (both functions are from the forecast package).

```
library(forecast)
arima=auto.arima(Icecream$cons)
fc=forecast(arima)
fc
```

```
##    Point Forecast     Lo 80     Hi 80     Lo 95     Hi 95
## 31      0.5316113 0.4830822 0.5801405 0.4573924 0.6058303
## 32      0.4909572 0.4229683 0.5589462 0.3869771 0.5949374
## 33      0.4356571 0.3570545 0.5142598 0.3154447 0.5558695
## 34      0.3762883 0.2934119 0.4591647 0.2495398 0.5030368
## 35      0.3226146 0.2390348 0.4061944 0.1947904 0.4504388
## 36      0.2821831 0.1984980 0.3658682 0.1541977 0.4101684
## 37      0.2594562 0.1741573 0.3447550 0.1290028 0.3899095
## 38      0.2555409 0.1667694 0.3443125 0.1197766 0.3913052
## 39      0.2684755 0.1754698 0.3614812 0.1262356 0.4107154
## 40      0.2939525 0.1973362 0.3905688 0.1461906 0.4417144
```

Exercise 3. Plot the obtained forecast with the autoplot.forecast function from the forecast package.

```
autoplot(fc)
```



Exercise 4. Use the accuracy function from the forecast package to find the mean absolute scaled error (MASE) of the fitted ARIMA model.

```r
accuracy(fc)
```

```
##                        ME       RMSE        MAE        MPE     MAPE
## Training set 0.0001020514 0.03525274 0.02692065 -0.9289035 7.203075
##                   MASE       ACF1
## Training set 0.8200619 -0.1002901
```

Our MASE is 0.82.

Exercise 5. Estimate an extended ARIMA model for the consumption data with the temperature variable as an additional regressor (using the auto.arima function). Then make a forecast for the next 6 periods (note that this forecast requires an assumption about the expected temperature; assume that the temperature for the next 6 periods will be represented by the following vector: `fcast_temp <- c(70.5, 66, 60.5, 45.5, 36, 28)`). Plot the obtained forecast.

```r
fcast_temp<-c(70.5,66,60.5,45.5,36,28)
arimax1=auto.arima(Icecream$cons,xreg=Icecream$temp)
fc2=forecast(arimax1,xreg=fcast_temp)
fc2
```

```
##    Point Forecast     Lo 80     Hi 80     Lo 95     Hi 95
## 31      0.5465774 0.5039101 0.5892446 0.4813234 0.6118313
## 32      0.5337735 0.4734329 0.5941142 0.4414905 0.6260566
## 33      0.5181244 0.4442225 0.5920263 0.4051012 0.6311476
## 34      0.4754450 0.3901105 0.5607796 0.3449371 0.6059529
## 35      0.4484147 0.3530078 0.5438217 0.3025024 0.5943270
## 36      0.4256524 0.3211393 0.5301654 0.2658135 0.5854913
```

```r
autoplot(fc2)
```

Exercise 6. Print summary of the obtained forecast. Find the coefficient for the temperature variable, its standard error, and the MASE of the forecast. Compare the MASE with the one of the initial forecast.

```r
summary(arimax1)
```

```
## Series: Icecream$cons
## Regression with ARIMA(0,1,0) errors
##
## Coefficients:
##         xreg
##       0.0028
## s.e.  0.0007
##
## sigma^2 estimated as 0.001108:  log likelihood=58.03
## AIC=-112.06   AICc=-111.6   BIC=-109.32
##
## Training set error measures:
##                       ME       RMSE        MAE      MPE     MAPE      MASE
## Training set 0.002563685 0.03216453 0.02414157 0.564013 6.478971 0.7354048
##                    ACF1
## Training set -0.1457977
```

```r
summary(fc2)
```

```
##
## Forecast method: Regression with ARIMA(0,1,0) errors
```

```
##
## Model Information:
## Series: Icecream$cons
## Regression with ARIMA(0,1,0) errors
##
## Coefficients:
##         xreg
##       0.0028
## s.e.  0.0007
##
## sigma^2 estimated as 0.001108:  log likelihood=58.03
## AIC=-112.06   AICc=-111.6   BIC=-109.32
##
## Error measures:
##                       ME       RMSE        MAE      MPE     MAPE      MASE
## Training set 0.002563685 0.03216453 0.02414157 0.564013 6.478971 0.7354048
##                    ACF1
## Training set -0.1457977
##
## Forecasts:
##    Point Forecast     Lo 80     Hi 80     Lo 95     Hi 95
## 31      0.5465774 0.5039101 0.5892446 0.4813234 0.6118313
## 32      0.5337735 0.4734329 0.5941142 0.4414905 0.6260566
## 33      0.5181244 0.4442225 0.5920263 0.4051012 0.6311476
## 34      0.4754450 0.3901105 0.5607796 0.3449371 0.6059529
## 35      0.4484147 0.3530078 0.5438217 0.3025024 0.5943270
## 36      0.4256524 0.3211393 0.5301654 0.2658135 0.5854913
```

The coefficient for temperature is 0.0028, and its standard error is 0.0007. The MASE is 0.7354, which is lower than the MASE of our initial forecast without the temperature regressor.

Exercise 7. Check the statistical significance of the temperature variable coefficient using the the `coeftest` function from the `lmtest` package. Is the coefficient statistically significant at 5% level?

```r
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```r
coeftest(arimax1)
```

```
##
## z test of coefficients:
##
##       Estimate Std. Error z value  Pr(>|z|)
## xreg 0.0028453  0.0007302  3.8966 9.756e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The p-value is 0.00009756, so the coefficient is statistically significant at the 5% level.

Exercise 8. The function that estimates the ARIMA model can input more additional regressors, but only in the form of a matrix. Create a matrix with the following columns:

- values of the temperature variable,
- values of the income variable,
- values of the income variable lagged one period,
- values of the income variable lagged two periods.

Print the matrix. Note: the last three columns can be created by prepending two NA's to the vector of values of the income variable, and using the obtained vector as an input to the embed function (with the dimension parameter equal to the number of columns to be created).

```r
#create matrix one column at a time
#start with temperature
temp=matrix(Icecream$temp,ncol=1)
inc=c(NA,NA,Icecream$income)
in1=embed(inc,3)
df2=cbind(temp,in1)
print(df2)
```

```
##       [,1] [,2] [,3] [,4]
##  [1,]   41   78   NA   NA
##  [2,]   56   79   78   NA
##  [3,]   63   81   79   78
##  [4,]   68   80   81   79
##  [5,]   69   76   80   81
##  [6,]   65   78   76   80
##  [7,]   61   82   78   76
##  [8,]   47   79   82   78
##  [9,]   32   76   79   82
## [10,]   24   79   76   79
## [11,]   28   82   79   76
## [12,]   26   85   82   79
## [13,]   32   86   85   82
## [14,]   40   83   86   85
## [15,]   55   84   83   86
## [16,]   63   82   84   83
## [17,]   72   80   82   84
## [18,]   72   78   80   82
## [19,]   67   84   78   80
## [20,]   60   86   84   78
## [21,]   44   85   86   84
## [22,]   40   87   85   86
## [23,]   32   94   87   85
## [24,]   27   92   94   87
## [25,]   28   95   92   94
## [26,]   33   96   95   92
## [27,]   41   94   96   95
## [28,]   52   96   94   96
## [29,]   64   91   96   94
## [30,]   71   90   91   96
```

Exercise 9. Use the obtained matrix to fit three extended ARIMA models that use the following variables as additional regressors:

- temperature, income,
- temperature, income at lags 0, 1,
- temperature, income at lags 0, 1, 2.

Examine the summary for each model, and find the model with the lowest value of the Akaike information criterion (AIC). Note that the AIC cannot be used for comparison of ARIMA models with different orders of integration (expressed by the middle terms in the model specifications) because of a difference in the number of observations. For example, an AIC value from a non-differenced model, ARIMA(p,0,q), cannot be compared to the corresponding value of a differenced model, ARIMA(p,1,q).

```
#temperature and income
arimax2=auto.arima(Icecream$cons,xreg=df2[,1:2])
summary(arimax2)
```

```
## Series: Icecream$cons
## Regression with ARIMA(1,0,0) errors
##
## Coefficients:
##          ar1   xreg1   xreg2
##       0.5672  0.0034  0.0023
## s.e.  0.1944  0.0006  0.0004
##
## sigma^2 estimated as 0.001125:  log likelihood=60.67
## AIC=-113.34   AICc=-111.74   BIC=-107.73
##
## Training set error measures:
##                        ME       RMSE        MAE        MPE     MAPE
## Training set -0.001314202 0.03181969 0.02393378 -0.8731766 6.505105
##                   MASE        ACF1
## Training set 0.7290753 -0.03519526
```

```
#temperature and income at lags 0,1
arimax3=auto.arima(Icecream$cons,xreg=df2[,1:3])
summary(arimax3)
```

```
## Series: Icecream$cons
## Regression with ARIMA(0,0,0) errors
##
## Coefficients:
##       intercept   xreg1    xreg2   xreg3
##         -0.1990  0.0035  -0.0003  0.0048
## s.e.     0.0896  0.0004   0.0021  0.0020
##
## sigma^2 estimated as 0.0009752:  log likelihood=60.96
## AIC=-111.92   AICc=-109.42   BIC=-104.92
##
## Training set error measures:
##                        ME       RMSE        MAE        MPE     MAPE
## Training set 2.679755e-17 0.02956819 0.02357644 -0.5004769 6.670631
##                   MASE      ACF1
## Training set 0.7181897 0.2350634
```

```
#temperature and income at lags 0,1,2
arimax4=auto.arima(Icecream$cons,xreg=df2[,1:4])
```

```r
summary(arimax4)
```

```
## Series: Icecream$cons
## Regression with ARIMA(0,0,0) errors
##
## Coefficients:
##       intercept   xreg1    xreg2   xreg3   xreg4
##          -0.224  0.0033  -0.0006  0.0012  0.0044
## s.e.      0.084  0.0004   0.0019  0.0024  0.0019
##
## sigma^2 estimated as 0.0008329:  log likelihood=61.12
## AIC=-110.25   AICc=-106.6   BIC=-101.84
##
## Training set error measures:
##                          ME       RMSE        MAE        MPE     MAPE
## Training set -6.155789e-16 0.02726996 0.02381773 -0.4481717 6.841168
##                   MASE      ACF1
## Training set 0.7255402 0.2716385
```

The order of integration (d value) of all ARIMAX models are the same, so we can use AIC values to find the best model. The first model with regressors temperature and lag=0 income has the lowest AIC value, so it is the best.

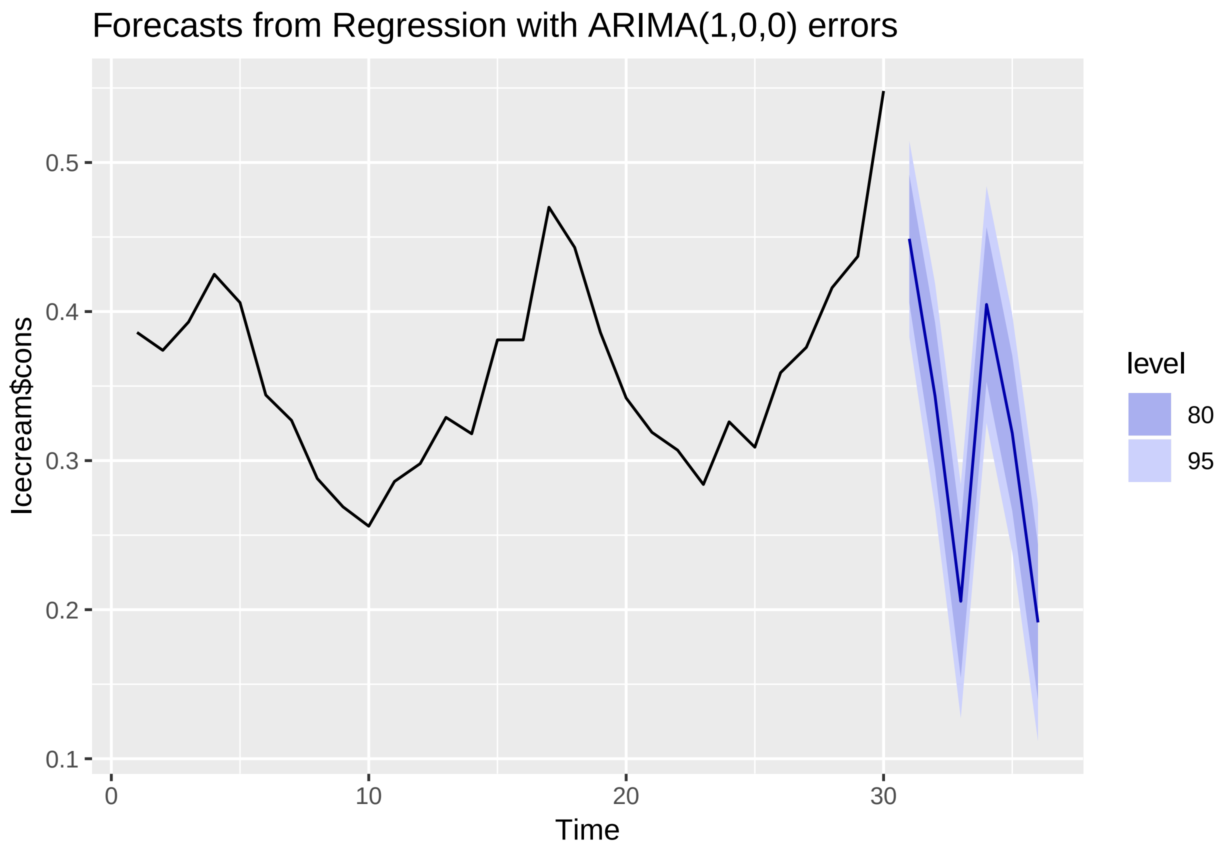
Exercise 10. Use the model found in the previous exercise to make a forecast for the next 6 periods, and plot the forecast. (The forecast requires a matrix of the expected temperature and income for the next 6 periods; create the matrix using the `fcast_temp` variable, and the following values for expected income: 91, 91, 93, 96, 96, 96). Find the mean absolute scaled error of the model, and compare it with the ones from the first two models in this exercise set.

```r
fcast_inc<-c(91,91,93,96,96,96)
temp_inc<-matrix(fcast_temp,fcast_inc,nrow=6,ncol=2)
arimaxfc=forecast(arimax2,xreg=temp_inc)
summary(arimaxfc)
```

```
##
## Forecast method: Regression with ARIMA(1,0,0) errors
##
## Model Information:
## Series: Icecream$cons
## Regression with ARIMA(1,0,0) errors
##
## Coefficients:
##          ar1   xreg1   xreg2
##       0.5672  0.0034  0.0023
## s.e.  0.1944  0.0006  0.0004
##
## sigma^2 estimated as 0.001125:  log likelihood=60.67
## AIC=-113.34   AICc=-111.74   BIC=-107.73
##
## Error measures:
##                        ME       RMSE        MAE        MPE     MAPE
## Training set -0.001314202 0.03181969 0.02393378 -0.8731766 6.505105
##                   MASE        ACF1
## Training set 0.7290753 -0.03519526
```

```
##
## Forecasts:
##    Point Forecast     Lo 80     Hi 80     Lo 95     Hi 95
## 31      0.4488284 0.4058440 0.4918128 0.3830894 0.5145674
## 32      0.3433342 0.2939170 0.3927513 0.2677572 0.4189112
## 33      0.2056198 0.1543044 0.2569352 0.1271396 0.2841000
## 34      0.4047710 0.3528597 0.4566824 0.3253794 0.4841626
## 35      0.3183453 0.2662437 0.3704470 0.2386628 0.3980279
## 36      0.1914464 0.1392837 0.2436091 0.1116705 0.2712224
```

```r
autoplot(arimaxfc)
```



Our MASE is 0.7291, and it is the lowest among the models without independent regressors and with just temperature as a regressor.