ARIMA(p,d,q): Implementation of Box-Jenkins Procedure

Vipul Bhatt

Application: Estimating an ARIMA(p,d,q) model for Moody's AAA rated corporate bond yield

We will implement the Box-Jenkins method and estimate an ARIMA(p,d,q) model for the US-Canada spot exchange rate. We use non-seasonally adjusted monthly data from Jan 1970 through March 2022. The figure below gives the plot of this natural logs of this yield over time:

```
library(xts)
library(readr)
data=read.csv("C:/Users/bhattvx/Dropbox/In Class/SP 2023/datasets/canada.csv")

y=ts(data$EXCAUS, start=c(1970,1), end=c(2022,3), frequency=12)

plot.xts(as.xts(y))
```

Figure 1: US-Canada Spot Exchange Rate

Our goal is to fit an ARIMA(p,d,q) to this data. We will follow the Box-Jenkins methodology which has the following three steps.

Model Identification

In this step we identify the order of integration (d), and the optimal ARMA structure (p,q)

1. Identifying *d*: We begin by testing for unit root. Before we do that, we need to de-seasonalize our data because our data is not seasonally adjusted. We can use the **decompose** function in R, and obtain the deseasonalized data. The plot below show the deasonalized data:

```
library(xts)

#### first deseaonalize data

ys=decompose(y, type="additive")

## compute deseasonalized data
y=y-ys$seasonal

plot.xts(as.xts(y), main="" )
```

Figure 2: De-seasonalized Corporate Bond Yield (AAA)

After de-seasonalizing our data we test for unit root. Table 1 reports the unit root test results for the three alternative models for our data. Because the absolute value of the test statistic is less than the abslute value of the critical value, we do not reject the null hypothesis and conclude that there is no sample evidence against the null hypothesis of unit root.

```r
library(urca)


ur.df(y,type="none", lags=10, selectlags ="AIC")
ur.df(y,type="drift", lags=10, selectlags ="AIC")
ur.df(y,type="trend", lags=10, selectlags ="AIC")
```

Table 1: Augmented Dickey-Fuller Test for Levels of Y

Test-statistic	5% Critical Value
-0.2708341	-1.95
-1.9157981	-2.86
-1.7086290	-3.41

Because we found non-stationarity in levels, we next test for unit root in first differences.

```r
library(urca)
ur.df(diff(y),type="none", selectlags ="AIC")
ur.df(diff(y),type="drift", selectlags ="AIC")
ur.df(diff(y),type="trend", selectlags ="AIC")
```

Table 2 reports the unit root test results for the first difference of our data. Now, the absolute value of the test statistic is more than the absolute value of the critical value, and so we can reject the null hypothesis. There is no unit root when we look at the first difference of our data. Hence, our data is I(1) or in other words $d = 1$.

Table 2: Augmented Dickey-Fuller Test for First Difference of Y

Test-statistic	5% Critical Value
-16.43404	-1.95
-16.42083	-2.86
-16.48439	-3.41

2. Identifying (p, q): Next we use either AIC or BIC to select the optimal lag structure for our ARMA model.

Table 3 below shows these statistics. Usin AIC the optimal structure is $p = 6, q = 6$ whereas based on BIC we get $p = 1, q = 1$.

Table 3: Optimal ARMA Structure

p	q	aic	bic
1	1	-3135.180	-3121.857
2	1	-3133.517	-3115.753
3	1	-3133.165	-3110.961
4	1	-3131.057	-3104.411
5	1	-3129.079	-3097.993
6	1	-3128.581	-3093.054
1	2	-3135.067	-3117.304
2	2	-3133.516	-3111.311
3	2	-3139.201	-3112.556
4	2	-3129.893	-3098.807

p	q	aic	bic
5	2	-3127.913	-3092.385
6	2	-3135.249	-3095.281
1	3	-3133.500	-3111.296
2	3	-3131.499	-3104.853
3	3	-3130.234	-3099.148
4	3	-3145.226	-3109.698
5	3	-3143.337	-3103.369
6	3	-3141.565	-3097.155
1	4	-3131.015	-3104.369
2	4	-3129.710	-3098.623
3	4	-3144.546	-3109.018
4	4	-3143.350	-3103.382
5	4	-3140.947	-3096.537
6	4	-3148.709	-3099.859
1	5	-3129.021	-3097.935
2	5	-3128.326	-3092.799
3	5	-3143.086	-3103.117
4	5	-3141.429	-3097.020
5	5	-3148.666	-3099.816
6	5	-3146.381	-3093.090
1	6	-3128.859	-3093.331
2	6	-3129.800	-3089.832
3	6	-3142.253	-3097.843
4	6	-3148.665	-3099.815
5	6	-3146.775	-3093.484
6	6	-3145.047	-3087.315

Parameter estimation

Now we can estimate our final model. Using the BIC our best model is ARIMA(1,1,1) given by:

$$y_t = \phi_0 + \phi_1 y_{t-1} + \theta_1 \epsilon_{t-1} + \epsilon_t$$

```r
library(forecast)

final=Arima(y,order=c(1,1,1), include.mean = T)

final
```

Table 4 shows the estimated model:

Table 4: Estimation Results for ARIMA(1,1,1)

	ϕ_1	θ_1
Estimated Coefficient	0.0490	0.1764
Std. Errors	0.1523	0.1489

Model evaluation

We now test whether the residuals are white noise using the BG test for serial correlation.

```r
library(lmtest)

### obtain residuals as lm object for leter use in bgtest

fe = lm(residuals(final) ~ 1)
a=rep(0,12)
b=rep(0,12)

### look for serial correlation from order 1 to 12
for (p in seq(along = rep(1,12))) {
a[p]=bgtest(fe,order=p)$statistic
b[p]=bgtest(fe,order=p)$p.value
}

results=data.frame(cbind(1:12, a,b))

results
```

Table 5 below reports the test statistic and the p-value.

Table 5: Breusch-Godfrey test for serial correlation

order	Test-statistic	$p - value$
1	0.0004	0.9840
2	0.0268	0.9867
3	0.6509	0.8847
4	1.8612	0.7613
5	1.8923	0.8638
6	2.8194	0.8311
7	3.7565	0.8074
8	3.8983	0.8662
9	5.2559	0.8115
10	7.3353	0.6935
11	15.1168	0.1772
12	17.2026	0.1421

The results confirm that upto 12th order of serial correlation we do not find any correlation in residuals. Hence, they are white noise.

Forecast based on the final model

Finally, we can use the estimated ARIMA(1,1,1) to forecast the future yield. The plot below shows the forecast for next 6 months and Table 6 shows the forecast values along with the confidence bands. Note that even though you are estimating the model in first difference to address non-stationarity, R's Arima() function automatically report the forecast for the levels and shows convergence to the mean of the data and not its first difference.

```r
y_f=forecast(final,h=6)
y_f
```

Table 6: Forecast for next 6 months

	Point Forecast	Lo 80	Hi 80	Lo 95	Hi 95
Apr 2022	0.9890	0.9637	1.0142	0.9503	1.0276
May 2022	0.9889	0.9489	1.0289	0.9278	1.0500
Jun 2022	0.9889	0.9382	1.0396	0.9113	1.0665
Jul 2022	0.9889	0.9293	1.0485	0.8978	1.0800
Aug 2022	0.9889	0.9216	1.0562	0.8860	1.0918
Sep 2022	0.9889	0.9147	1.0631	0.8754	1.1024

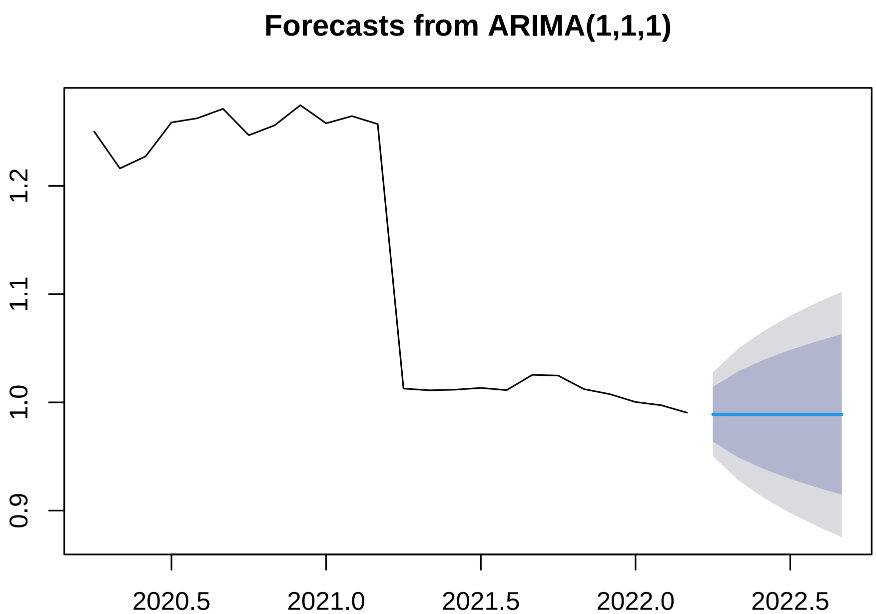

Figure 3: Forecast of Corporate Bond Yield (AAA)