Report

Qitian Ma

I researched on Excess Reserves of Depository Institutions (EXCSRESNS). This series is non seasonally adjusted. However, later statistical test does not show the presence of trend or seasonality.

First, I did Augmented Dicky-Fuller test, which shows the potential existence of unit root. So I differentiated the data.

From the general plot, I saw no significant trend or seasonality. Later regression on trends and seasonal dummies further confirmed this. ACF plot points to MA(1) or MA(3). PACF plots point to AR(1) or AR(3). Auto.arima function points to ARMA(1,3). Therefore, I decided to fit ARIMA(1,0,3) model.

I applied one-step-ahead cross-validation because the general plot shows that cycles do not repeat exactly the same, thus not suitable for train-validation methods. Under RMSE criteria, ARIMA(1,0,3) performs better then Exponential Average (alpha=0.6) as a baseline model. And Diebold/Mariano test further verified this.

Since the above model is applied on differentiated series, the original model is ARIMA(1,0,3).

The final fitted model is:

$$(y_t - y_{t-1}) = 0.7493(y_{t-1}-y_{t-2}) + \epsilon_t - 0.4332\varepsilon_{t-1} - 0.3373\varepsilon_{t-2} + 0.3253\varepsilon_{t-3} + 11121.40$$

The increment of neighboring months are positively correlated.

Positive noise in month corresponds to positive increment three months after and negative increment in the following two months.

```r
library(tidyverse)

## ── Attaching packages
────────────────────────────────────────────────────────────────────────
─────────────────────────────────────── tidyverse 1.3.0 ──

## ✓ ggplot2 3.3.0     ✓ purrr   0.3.3
## ✓ tibble  3.0.0     ✓ dplyr   0.8.4
## ✓ tidyr   1.0.2     ✓ stringr 1.4.0
## ✓ readr   1.3.1     ✓ forcats 0.4.0

## ── Conflicts
────────────────────────────────────────────────────────────────────────
────────────────────────────────────────── tidyverse_conflicts() ──
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(forecast)

## Registered S3 method overwritten by 'xts':
##   method     from
##   as.zoo.xts zoo

## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo

## Registered S3 methods overwritten by 'forecast':
##   method             from
##   fitted.fracdiff    fracdiff
##   residuals.fracdiff fracdiff

library(lubridate)

##
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':
##
##     date

library(urca)

raw.ts <- read_csv("EXCSRESNS.csv")

## Parsed with column specification:
## cols(
##   DATE = col_date(format = ""),
##   EXCSRESNS = col_double()
## )

raw.ts$DATE <- date(raw.ts$DATE)
raw.ts$DTBSPCKM <- as.numeric(raw.ts$EXCSRESNS)
raw.ts <- raw.ts[order(raw.ts$DATE), ]

# select year after 2009
raw.ts <- raw.ts[300:length(raw.ts$DATE),]
raw.ts <- ts(raw.ts$EXCSRESNS, start=c(2009,1), freq=12)
```

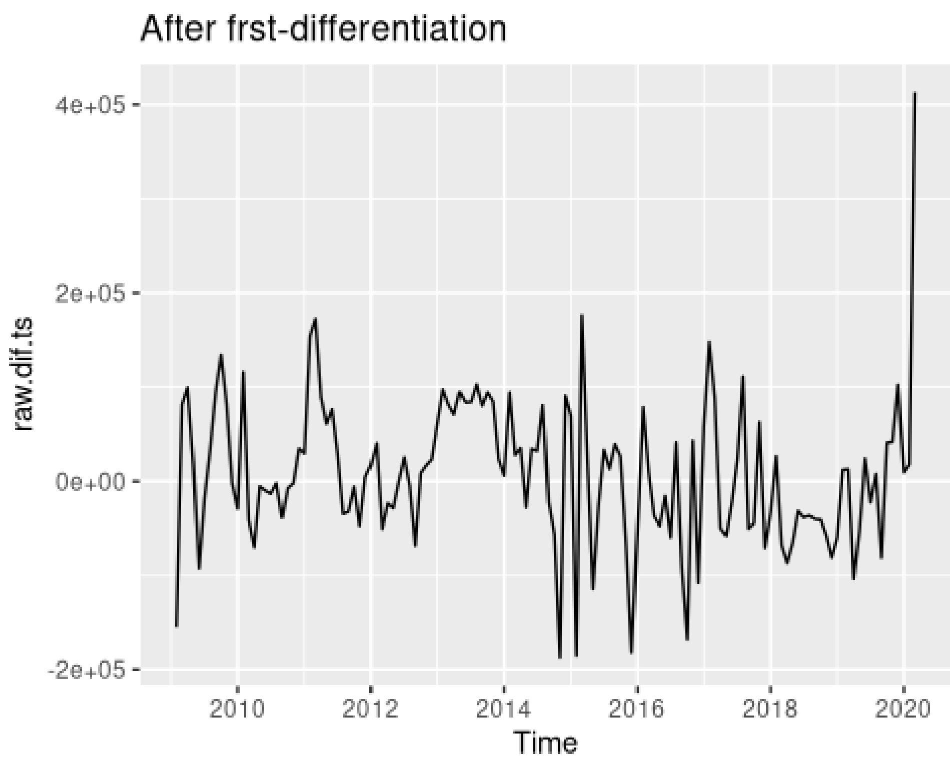
1. First we need to test the existence of unit root. The ADF test does not rule out the existence of unit root under 5% significance level (-1.6508 > -3.43). After first difference, the ADF test rules out the existence of unit oot under 5% significance level (-6.3374 < -3.43).

```
# unit root test
summary(ur.df(raw.ts, type="trend", selectlags="BIC"))

##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression trend
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + tt + z.diff.lag)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -192774  -39450    -654   43782  399265
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.294e+04  2.212e+04   2.393  0.01816 *
## z.lag.1     -2.249e-02  1.362e-02  -1.651  0.10122
## tt          -6.938e+01  2.020e+02  -0.343  0.73180
## z.diff.lag   2.567e-01  9.502e-02   2.702  0.00783 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 75780 on 129 degrees of freedom
## Multiple R-squared:  0.09159,    Adjusted R-squared:  0.07047
## F-statistic: 4.336 on 3 and 129 DF,  p-value: 0.006019
##
##
## Value of test-statistic is: -1.6508 2.0221 2.2512
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -3.99 -3.43 -3.13
## phi2  6.22  4.75  4.07
## phi3  8.43  6.49  5.47

raw.dif.ts <- diff(raw.ts)
summary(ur.df(raw.dif.ts, type="trend", selectlags="BIC"))

##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression trend
##
##
## Call:
```

```
## lm(formula = z.diff ~ z.lag.1 + 1 + tt + z.diff.lag)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -205624  -39304   -5536   38268  414963
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.176e+04  1.425e+04   1.527    0.129
## z.lag.1     -7.835e-01  1.236e-01  -6.337 3.66e-09 ***
## tt          -2.093e+02  1.831e+02  -1.143    0.255
## z.diff.lag   6.956e-02  9.756e-02   0.713    0.477
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 76230 on 128 degrees of freedom
## Multiple R-squared:  0.3081, Adjusted R-squared:  0.2919
## F-statistic:    19 on 3 and 128 DF,  p-value: 2.975e-10
##
##
## Value of test-statistic is: -6.3374 13.6964 20.4475
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -3.99 -3.43 -3.13
## phi2  6.22  4.75  4.07
## phi3  8.43  6.49  5.47
```

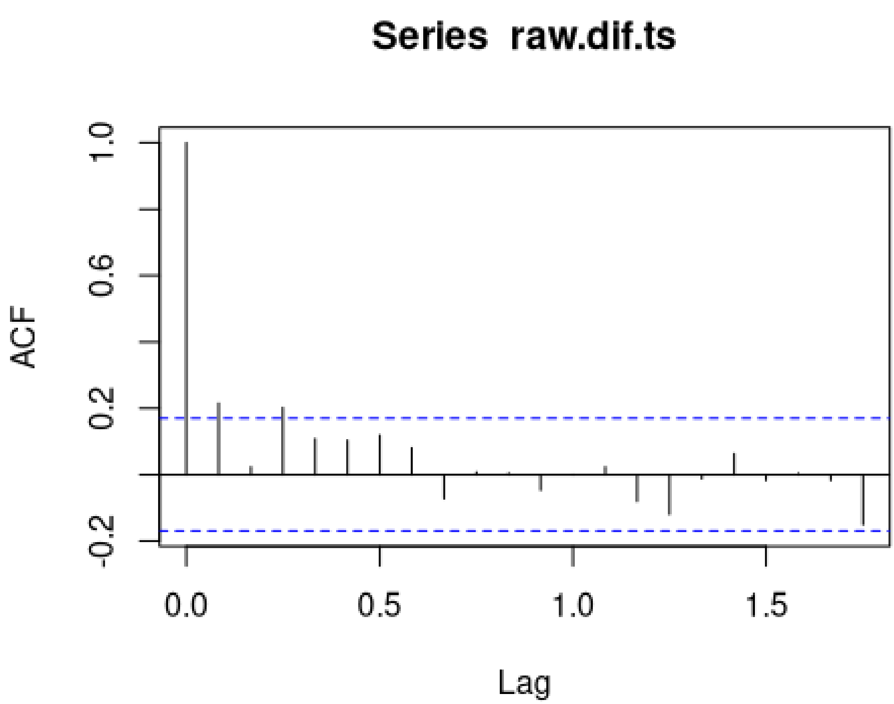
2. Simple Time Plot
   From the plot, this time series seems stationary, without trend, without seasonality, with cycle.

```
# general plot
autoplot(raw.dif.ts) +
  ggtitle("After frst-differentiation")
```

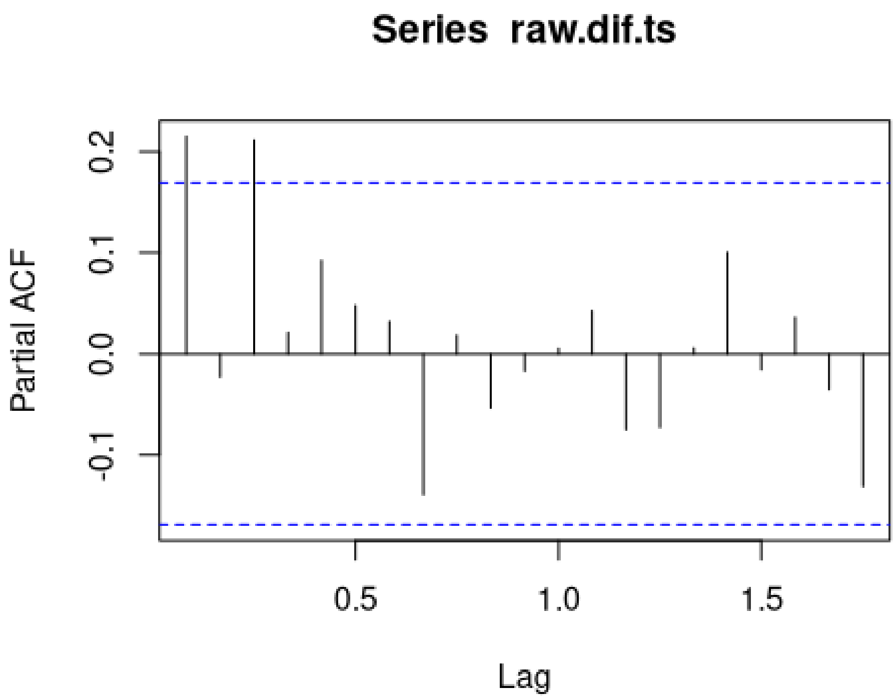


3. The ACF-PACF plot points to AR 1/3 MA 1/3 and shows not seasonality or unit root.

```
# plot acf pacf
acf(raw.dif.ts) # MA(1) or MA(3)
```




```
pacf(raw.dif.ts) # AR(3)
```



4. The statistics do not support trend or seasonality and suggest ARMA(1,3)

```
# trend and seasonality
raw.dif.trend.lin <- tslm(raw.dif.ts~trend+season)

summary(raw.dif.trend.lin)
```

```
##
## Call:
## tslm(formula = raw.dif.ts ~ trend + season)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -220852  -40702   -6033   50914  356411
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  26877.3    26571.6   1.012   0.3138
## trend         -276.1      174.0  -1.587   0.1151
## season2      28693.6    32454.4   0.884   0.3784
## season3      67100.1    32450.2   2.068   0.0408 *
## season4     -14271.2    33177.5  -0.430   0.6679
## season5     -29716.7    33169.8  -0.896   0.3721
## season6      -7076.0    33162.9  -0.213   0.8314
## season7      -1538.6    33157.0  -0.046   0.9631
## season8      19985.9    33152.0   0.603   0.5477
## season9     -27339.8    33147.9  -0.825   0.4111
## season10     -9439.6    33144.7  -0.285   0.7763
## season11    -10130.2    33142.4  -0.306   0.7604
## season12    -22157.9    33141.0  -0.669   0.5050
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 77720 on 121 degrees of freedom
## Multiple R-squared:  0.1323, Adjusted R-squared:  0.04623
## F-statistic: 1.537 on 12 and 121 DF,  p-value: 0.1199
```

```
auto.arima(raw.dif.ts)
```

```
## Series: raw.dif.ts
## ARIMA(1,0,3) with zero mean
##
## Coefficients:
##          ar1      ma1      ma2     ma3
##       0.7493  -0.4412  -0.3399  0.3267
## s.e.  0.1475   0.1592   0.1069  0.0987
##
## sigma^2 estimated as 5.589e+09:  log likelihood=-1692.14
## AIC=3394.28   AICc=3394.75   BIC=3408.77
```

5. ARMA(1,3) beats Exponential Average (by RMSE and Diebold/Mariano test)

```
# model
arima.mod <- function(x, h){
  temp <- Arima(order=c(1,0,3), x)
  temp2 <- tryCatch(forecast(temp, h=h), error=function(e){print("error");
NA})
  temp2
}

eCV <- tsCV(raw.dif.ts, arima.mod, h=1)
rmseCV <- sqrt( mean( eCV^2,na.rm=TRUE))

exptrend.mod <- function(x, h){
  ses(x, h=1, alpha=0.6)
}

eCV.exp <- tsCV(raw.dif.ts, exptrend.mod, h=1)
rmseCV.exp <- sqrt( mean( eCV.exp^2,na.rm=TRUE))

rmseCV < rmseCV.exp
```

```
## [1] TRUE
```

```
# Diebold/Mariano test
res.exp <- ses(raw.dif.ts, alpha=0.6)$res
arima.mod <- arima(raw.dif.ts, order=c(1,0,3))
res.arma <- arima.mod$res
dm.test(res.exp, res.arma, alternative="greater")
```

```
##
##  Diebold-Mariano Test
##
## data:  res.expres.arma
## DM = 2.3539, Forecast horizon = 1, Loss function power = 2, p-value =
## 0.01002
## alternative hypothesis: greater
```

```
summary(arima.mod)
```

```
##
## Call:
## arima(x = raw.dif.ts, order = c(1, 0, 3))
##
## Coefficients:
##          ar1      ma1      ma2     ma3  intercept
##       0.7358  -0.4332  -0.3373  0.3253   11121.40
## s.e.  0.1542   0.1646   0.1072  0.0986   13177.33
##
## sigma^2 estimated as 5.395e+09:  log likelihood = -1691.78,  aic = 3395.56
##
## Training set error measures:
##                    ME     RMSE      MAE      MPE     MAPE      MASE
## Training set 544.9274 73448.24 53990.28 824.2871 858.2841 0.8485887
##                      ACF1
## Training set -0.009979304
```

6. Prediction Plot
   ARIMA may spit NA since MLE does not necessarily converge. However, from the available data, the prediction result is good.

```
# Prediction Plot
store <- c()
n=1
while(n<=length(raw.dif.ts)){
  temp <- tryCatch(Arima(order=c(1,0,3), raw.dif.ts[1:n]), error=function(e)
{NA})
  temp2 <- tryCatch(forecast(temp, h=1), error=function(e){NA})
  store <<- c(store, ifelse(any(is.na(temp2)), NA, temp2$mean))
  n <- n+1
}

## Warning in sqrt(z[[2L]] * object$sigma2): NaNs produced

## Warning in forecast.forecast_ARIMA(temp, h = 1): Upper prediction
intervals are
## not finite.

## Warning in sqrt(z[[2L]] * object$sigma2): NaNs produced

## Warning in forecast.forecast_ARIMA(temp, h = 1): Upper prediction
intervals are
## not finite.

## Warning in sqrt(z[[2L]] * object$sigma2): NaNs produced

## Warning in forecast.forecast_ARIMA(temp, h = 1): Upper prediction
intervals are
## not finite.

## Warning in forecast.forecast_ARIMA(temp, h = 1): Upper prediction
intervals are
## not finite.

store <- ts(store, start=c(2009,2), freq=12) # The first input series is
just 2009-1 and the prediction is 2009-2

autoplot(raw.dif.ts) +
```

```
  autolayer(store, series="one-step ahead prediction", color="red") +
  ggtitle("True (black) vs Predicted (red)")
```

```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```