Assignment 05, Question 1

Jeff Nguyen

24/04/2021

University of Southern California
Marshall School of Business
FBE 543 Forecasting and Risk Analysis

#**Question 1**

```
library(quantmod)

# Set start date and end date of data
start_date <- "2014-01-01"
end_date <- "2021-03-23"

# Get data
getSymbols("JPM", src = "yahoo", , from = start_date, to = end_date)
```

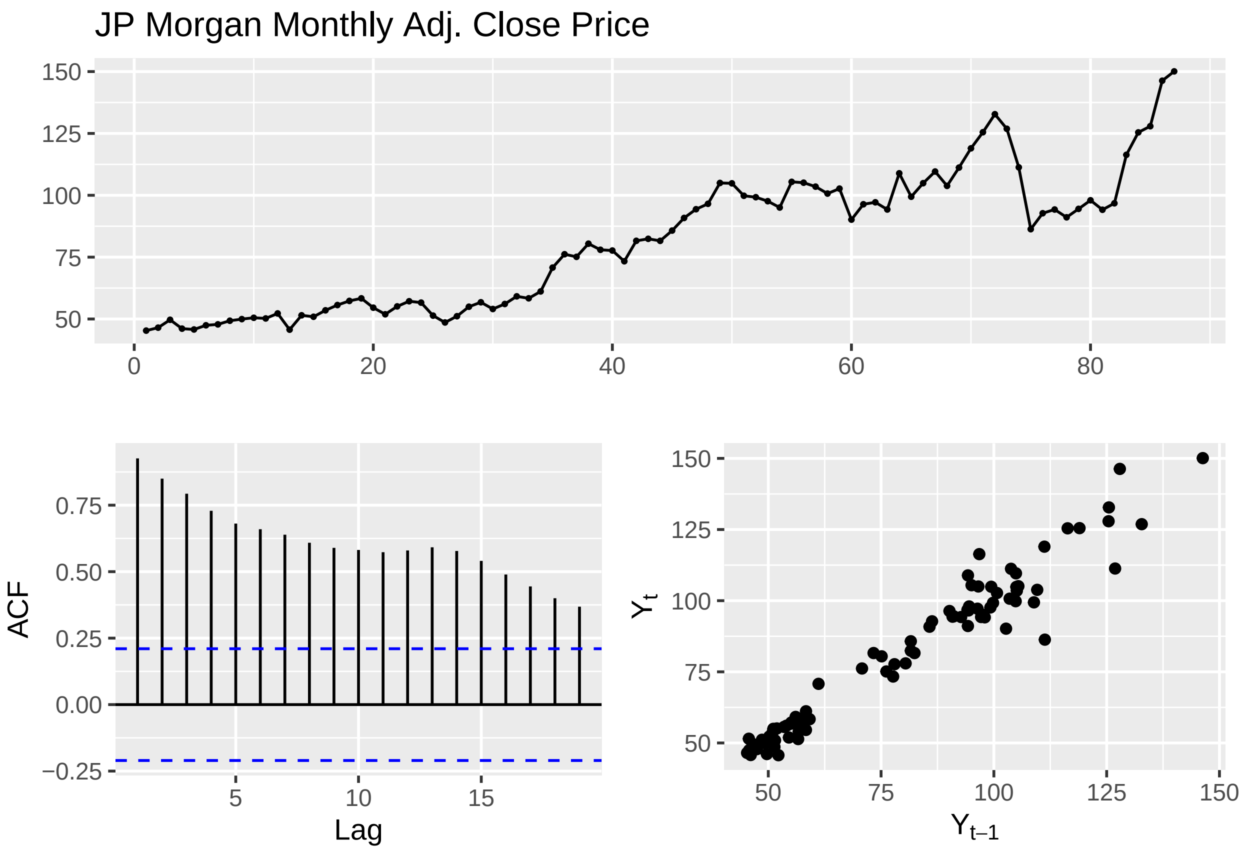
```
## [1] "JPM"
```

```
adjJPM_mo <- to.monthly(JPM)$JPM.Adjusted  # Monthly Adjusted Closing Price
rJPM_mo <- diff(log(adjJPM_mo))[-1]  # Monthly Returns
```

Data Observation:

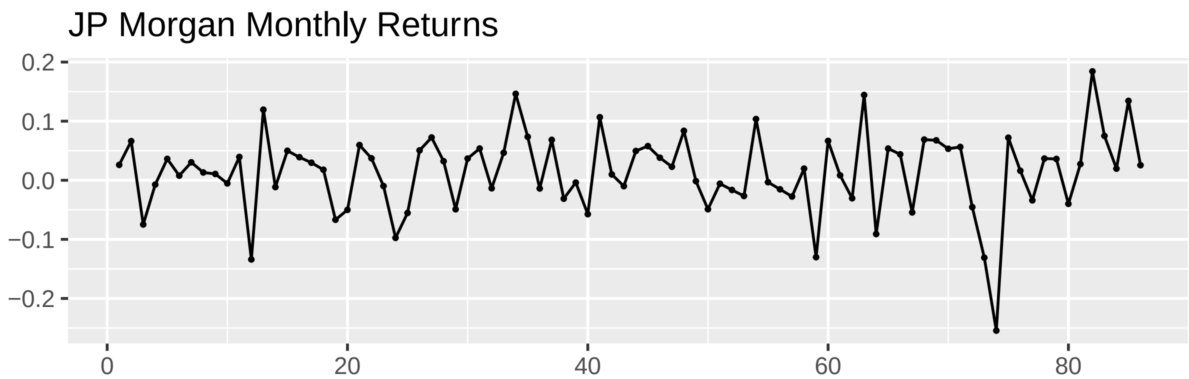
Observing monthly adjusted closing prices:

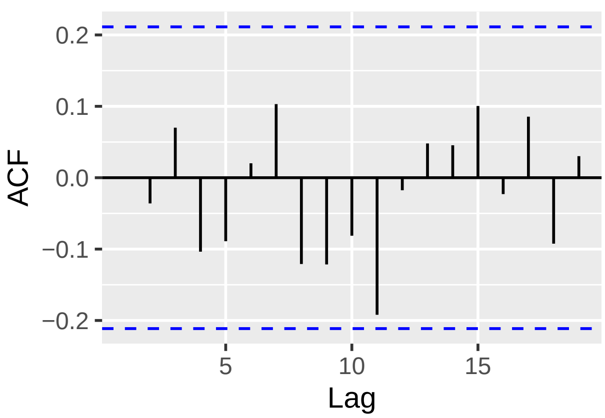
```
library(forecast)

ggtsdisplay(adjJPM_mo, main="JP Morgan Monthly Adj. Close Price", plot.type="scatter")
```

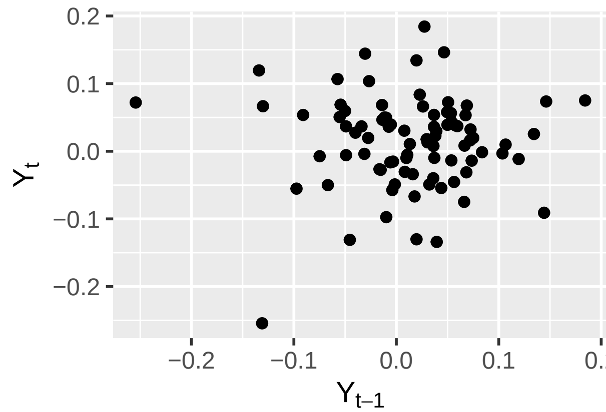


Observing monthly returns:

```
ggtsdisplay(rJPM_mo, main="JP Morgan Monthly Returns", plot.type="scatter")
```







Remarks
We can see JPM's monthly adjusted closing price' lag plots exhibit a linear pattern, implying that the data is strongly non-random and thus, a first-order autoregressive model might be appropriate.

$$y_t = \beta_0 + \beta_1 y_{t-1} + \epsilon_t \tag{1}$$

On the other hand, JPM's monthly returns's lag plot does not exhibit any obvious patterns, implying that the data is strongly random.

1. Test for the stationarity of the adjusted closing prices for JPM.

We run Augmented Dickey Fuller Test for JPM. Recall that the null hypothesis for Dickey-Fuller Test is that a unit root is present in our autoregressive model, meaning the variable is a non-stationary variable.

```
library(aTSA)

adf.test(adjJPM_mo)
```

```
## Augmented Dickey-Fuller Test
## alternative: stationary
##
## Type 1: no drift no trend
##      lag  ADF p.value
## [1,]   0 2.44   0.990
## [2,]   1 2.17   0.990
```

```
## [3,]   2 2.11   0.990
## [4,]   3 1.89   0.984
## Type 2: with drift no trend
##      lag  ADF p.value
## [1,]   0 2.21    0.99
## [2,]   1 2.05    0.99
## [3,]   2 2.12    0.99
## [4,]   3 1.80    0.99
## Type 3: with drift and trend
##      lag  ADF p.value
## [1,]   0 3.29    0.99
## [2,]   1 3.17    0.99
## [3,]   2 3.29    0.99
## [4,]   3 3.05    0.99
## ----
## Note: in fact, p.value = 0.01 means p.value <= 0.01
```

We can observe $p - value = .99 > .05$. Thus, we fail to reject the null hypothesis. In other words, JPM monthly adjusted closing price has a unit root and therefore, is a non-stationary variable.

2. Test for the stationarity of the returns for JPM.

Similarly, we run (A)DF test for JPM's monthly returns:

```
adf.test(rJPM_mo)
```

```
## Augmented Dickey-Fuller Test
## alternative: stationary
##
## Type 1: no drift no trend
##      lag   ADF p.value
## [1,]   0  8.79    0.99
## [2,]   1 12.44    0.99
## [3,]   2 14.48    0.99
## [4,]   3 16.64    0.99
## Type 2: with drift no trend
##      lag  ADF p.value
## [1,]   0  9.1    0.99
## [2,]   1 13.1    0.99
## [3,]   2 15.3    0.99
## [4,]   3 17.9    0.99
## Type 3: with drift and trend
##      lag   ADF p.value
## [1,]   0  9.05    0.99
## [2,]   1 12.98    0.99
## [3,]   2 15.25    0.99
## [4,]   3 17.78    0.99
## ----
## Note: in fact, p.value = 0.01 means p.value <= 0.01
```

We can observe $p - value = .99 > .05$. Thus, we fail to reject the null hypothesis. In other words, JPM monthly returns has a unit root and therefore, is a non-stationary variable.

3. Run the best ARIMA model for JPM returns.

We run auto ARIMA:

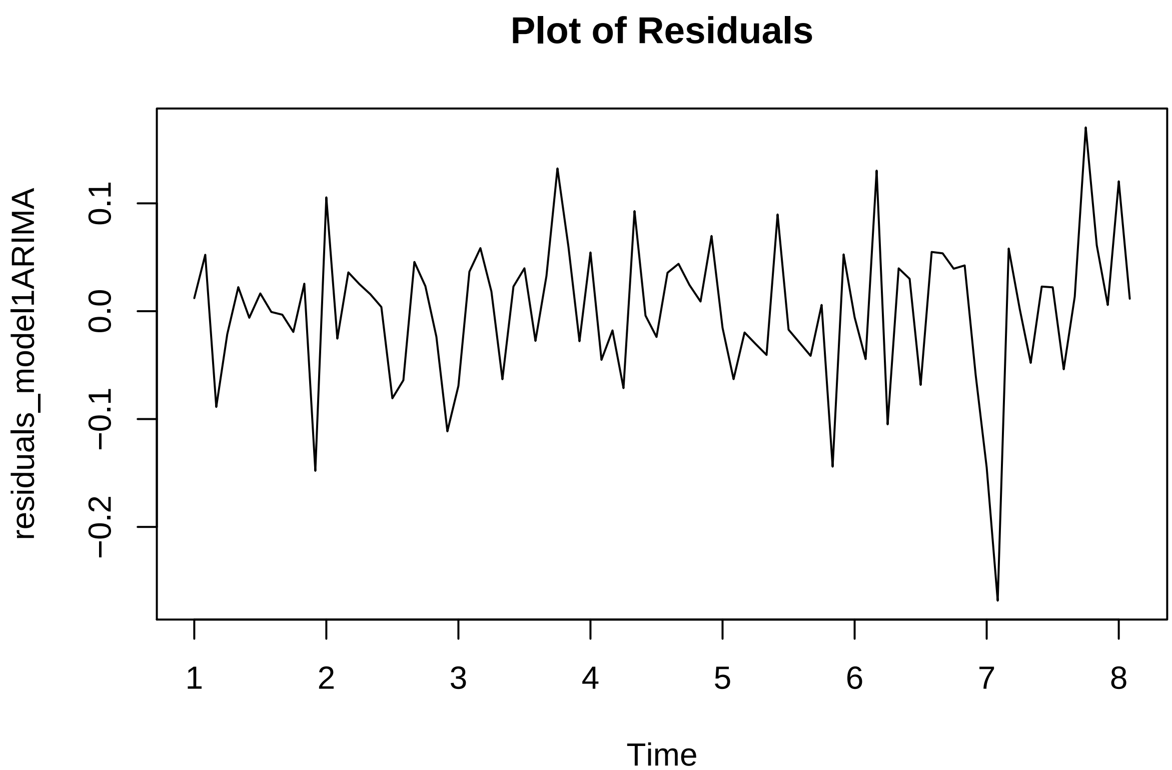
```
model1ARIMA <- auto.arima(rJPM_mo)
summary(model1ARIMA)
```

```
## Series: rJPM_mo
## ARIMA(0,0,0) with non-zero mean
##
## Coefficients:
##         mean
##       0.0139
## s.e.  0.0072
##
## sigma^2 estimated as 0.004527:  log likelihood=110.57
## AIC=-217.15   AICc=-217   BIC=-212.24
##
## Training set error measures:
##                        ME       RMSE        MAE      MPE     MAPE      MASE
## Training set 6.867467e-19 0.06689178 0.04978467 117.7615 122.9537 0.6406739
##                     ACF1
## Training set 0.001628326
```

We can observe that ARIM(0,0,0) model with non-zero mean represents a White Noise model with independent and identically distributed data.

4. Test for the existence of heteroskedasticity on the residuals of the JPM's ARIMA model.

```
residuals_model1ARIMA <- model1ARIMA$residuals
plot(residuals_model1ARIMA, main="Plot of Residuals")
```




```
residualsSq_model1ARIMA <- residuals_model1ARIMA^2
```

Testing for hetereoscedacity on the residuals of the ARIMA model above using Breusch-Pagan Test:

```
library(lmtest)

reg_residualsSq_model1ARIMA <- lm(residualsSq_model1ARIMA ~ rJPM_mo)
bp_residualsSq_model1ARIMA <- bptest(reg_residualsSq_model1ARIMA)

bp_residualsSq_model1ARIMA
```

```
##
##  studentized Breusch-Pagan test
##
## data:  reg_residualsSq_model1ARIMA
## BP = 6.1131, df = 1, p-value = 0.01342
```

We can observe $p - value = .01343 < .05$, thus we reject the null-hypothesis and assume hetereoscedacity of the residuals of the ARIMA model for monthly JPM stock returns.

Similarly, we attempt a Goldfield-Quant test:

```
gq_residualsSq_model1ARIMA <- gqtest(reg_residualsSq_model1ARIMA)
gq_residualsSq_model1ARIMA
```

```
##
##  Goldfeld-Quandt test
##
```

```
## data:  reg_residualsSq_model1ARIMA
## GQ = 5.5116, df1 = 41, df2 = 41, p-value = 1.313e-07
## alternative hypothesis: variance increases from segment 1 to 2
```

We can observe similar result with the Breusch-Pagan test as $p - value = 1.308e - 07 < .05$, i.e. we reject the null hypothesis and assume hetereoscedacity for the residuals of the ARIMA model for monthly returns.

5. Find the historical measure of the volatility of the JPM's returns.

Recall the historical volatility measure:

$$\begin{aligned} \sigma^2_{hist} &= \frac{\sum(r - \overline{r})^2}{n-1} \\ \text{where } r &= \text{monthly stock returns} \\ n &= \text{number of observations} \end{aligned} \tag{2}$$

```
mean_rJPM_mo <- mean(rJPM_mo)
sigmaSq_hist <- sum((rJPM_mo - mean_rJPM_mo)^2 / (length(rJPM_mo) - 1))
sigmaSq_hist
```

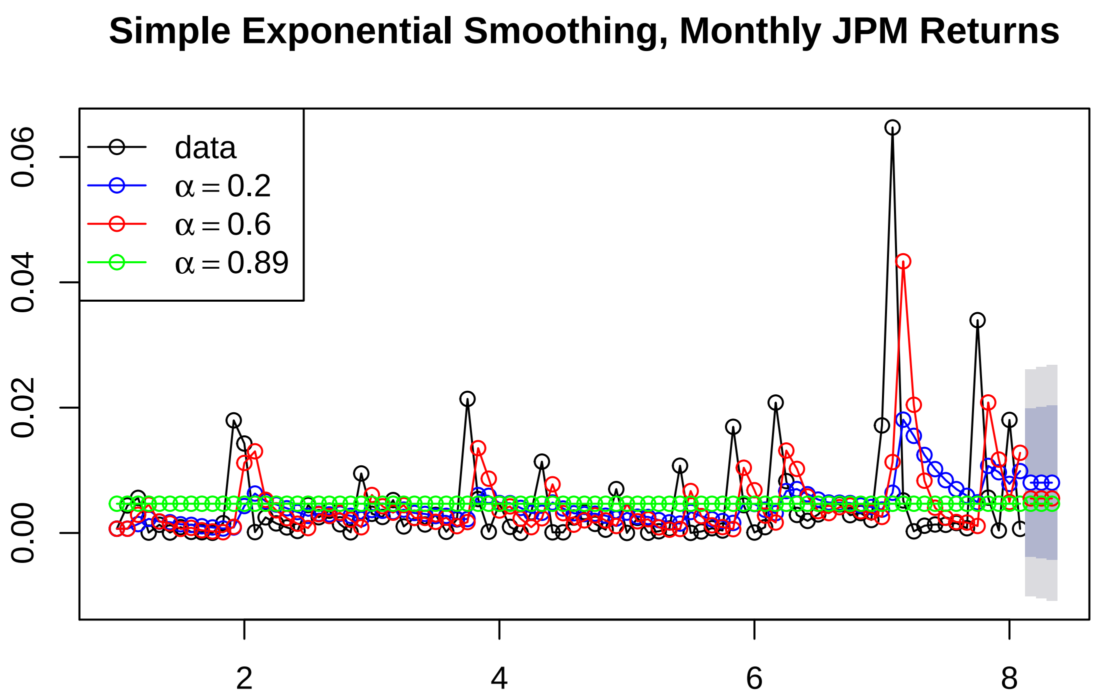
```
## [1] 0.004527152
```

Thus, $\sigma^2_{hist} = 0.0045272$.

6. Find the Exponentially Weighted Moving Average (EWMA) measure of the volatility of JPM's returns.

The EWMA measure of volatility is the exponential smoothing of the square of JPM's monthly returns. We have:

```
rJPM_mo_Sq <- rJPM_mo^2

#Simple Exponential Smoothing
fit1 <- ses(rJPM_mo_Sq, alpha=0.2, initial="simple", h=3)
fit2 <- ses(rJPM_mo_Sq, alpha=0.6, initial="simple", h=3)
fit3 <- ses(rJPM_mo_Sq, h=3)
plot(fit1, main="Simple Exponential Smoothing, Monthly JPM Returns", fcol="white", type="o")
lines(fitted(fit1), col="blue", type="o")
lines(fitted(fit2), col="red", type="o")
lines(fitted(fit3), col="green", type="o")
lines(fit1$mean, col="blue", type="o")
lines(fit2$mean, col="red", type="o")
lines(fit3$mean, col="green", type="o")
legend("topleft",lty=1, col=c(1,"blue","red","green"),
       c("data", expression(alpha == 0.2), expression(alpha == 0.6),
         expression(alpha == 0.89)),pch=1)
```



Recall:

$$\sigma^2_{EWMA} = (1 - \lambda) \sum \lambda (r - \overline{r})^2 \tag{3}$$
$$\text{where } \lambda = .94$$

```
lambda <- .94
sigmaSq_EWMA <- (1-lambda) * sum(lambda * (rJPM_mo - mean_rJPM_mo)^2)
sigmaSq_EWMA
```

```
## [1] 0.02170317
```

We can see that $\sigma^2_{EWMA} = 0.0217032$.

7. Test whether the two measures of the volatility, historical and EWMA, are statistically the same.

We can see that $\sigma^2_{hist} = 0.0045272 \neq 0.0217032 = \sigma^2_{EWMA}$. Thus, they are not the same.

8. Estimate an auto-regression model of the volatility for JPM's returns using r^2 and log(High/low) as measures of volatility. Use the model to forecast next three-periods conditional variance.

Estimating ARIMA model of JPM's monthly returns volatility using r^2

```
fit4 <- auto.arima(rJPM_mo_Sq)
fit4
```

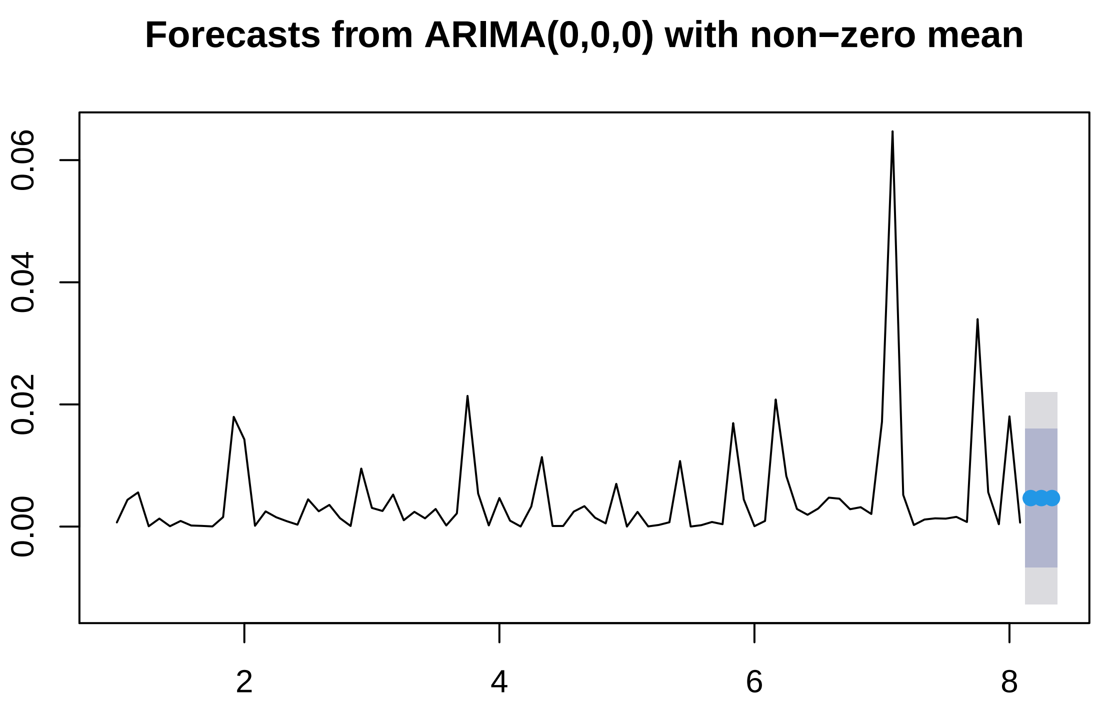
```
## Series: rJPM_mo_Sq
## ARIMA(0,0,0) with non-zero mean
##
## Coefficients:
##          mean
##        0.0047
## s.e.   0.0010
##
## sigma^2 estimated as 7.859e-05:  log likelihood=284.88
## AIC=-565.76   AICc=-565.62   BIC=-560.85
```

We can see the model is again, a White Noise model. Forecasting the next three periods:

```
fcast4 <-forecast::forecast(fit4, h=3)
fcast4
```

```
##       Point Forecast         Lo 80      Hi 80       Lo 95      Hi 95
## Mar 8    0.004668495 -0.006692241 0.01602923 -0.01270625 0.02204324
## Apr 8    0.004668495 -0.006692241 0.01602923 -0.01270625 0.02204324
## May 8    0.004668495 -0.006692241 0.01602923 -0.01270625 0.02204324
```

```
plot(fcast4)
```



Estimating ARIMA model of JPM's monthly returns volatility using $\ln H/L$

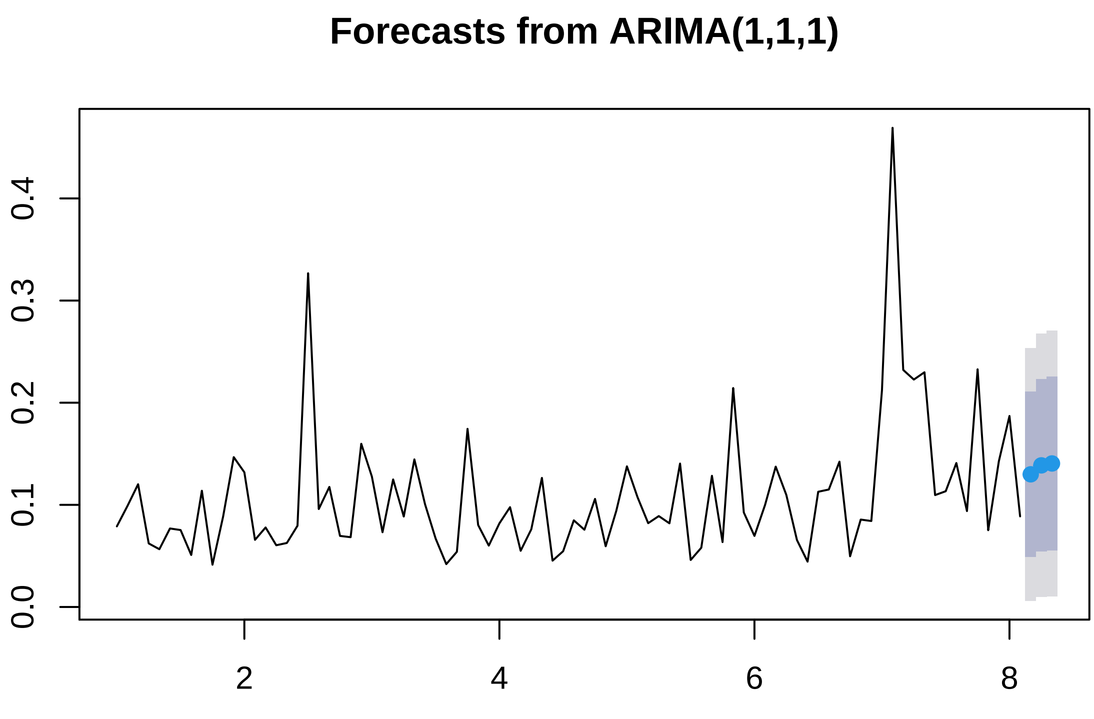
```
lnJPM_HL <- log(to.monthly(JPM)$JPM.High/to.monthly(JPM)$JPM.Low)[-1]

fit5 <- auto.arima(lnJPM_HL)
fit5
```

```
## Series: lnJPM_HL
## ARIMA(1,1,1)
##
## Coefficients:
##          ar1      ma1
##       0.2125  -0.9220
## s.e.  0.1230   0.0558
##
## sigma^2 estimated as 0.003987:  log likelihood=114.45
## AIC=-222.91   AICc=-222.61   BIC=-215.58
```

```
fcast5 <-forecast::forecast(fit5, h=3)
fcast5
```

```
##       Point Forecast      Lo 80     Hi 80       Lo 95     Hi 95
## Mar 8      0.1298851 0.04896932 0.2108009 0.006135100 0.2536351
## Apr 8      0.1386425 0.05438197 0.2229030 0.009777162 0.2675078
## May 8      0.1405033 0.05548773 0.2255189 0.010483209 0.2705234
```

```
plot(fcast5)
```



11. Run an ARCH and/or GARCH model on JPM's returns data.

```
library(rugarch)

# Write Specification of Your GARCH Model using "sGrach" or standard GARCH Mode.
garch1 <- ugarchspec(variance.model=list(model="sGARCH", garchOrder=c(1, 1)), mean.model=list(armaOrder=
```

```r
# Fit the Model to Data
garch1_rJPM_mo <- ugarchfit(spec=garch1, data=rJPM_mo)
garch1_rJPM_mo
```

```
##
## *---------------------------------*
## *          GARCH Model Fit        *
## *---------------------------------*
##
## Conditional Variance Dynamics
## -----------------------------------
## GARCH Model  : sGARCH(1,1)
## Mean Model   : ARFIMA(1,0,1)
## Distribution : std
##
## Optimal Parameters
## ------------------------------------
##         Estimate  Std. Error    t value Pr(>|t|)
## mu      0.013436    0.002007   6.693492 0.000000
## ar1     0.870117    0.050415  17.258924 0.000000
## ma1    -1.000000    0.060591 -16.504035 0.000000
## omega   0.000094    0.000279   0.337728 0.735568
## alpha1  0.000000    0.020798   0.000001 0.999999
## beta1   0.999000    0.065574  15.234635 0.000000
## shape   2.649275    0.598007   4.430173 0.000009
##
## Robust Standard Errors:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.013436    0.002424  5.54355 0.000000
## ar1     0.870117    0.041770 20.83103 0.000000
## ma1    -1.000000    0.091327 -10.94965 0.000000
## omega   0.000094    0.000557  0.16945 0.865441
## alpha1  0.000000    0.108415  0.00000 1.000000
## beta1   0.999000    0.072439 13.79094 0.000000
## shape   2.649275    0.817581  3.24038 0.001194
##
## LogLikelihood : 118.7638
##
## Information Criteria
## ------------------------------------
##
## Akaike       -2.5992
## Bayes        -2.3994
## Shibata      -2.6111
## Hannan-Quinn -2.5188
##
## Weighted Ljung-Box Test on Standardized Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                    0.02446  0.8757
## Lag[2*(p+q)+(p+q)-1][5]   0.36478  1.0000
## Lag[4*(p+q)+(p+q)-1][9]   1.03487  0.9997
## d.o.f=2
```

```
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                     0.8334  0.3613
## Lag[2*(p+q)+(p+q)-1][5]    3.0258  0.4024
## Lag[4*(p+q)+(p+q)-1][9]    3.7037  0.6394
## d.o.f=2
##
## Weighted ARCH LM Tests
## ------------------------------------
##             Statistic Shape Scale P-Value
## ARCH Lag[3]    0.4186 0.500 2.000  0.5176
## ARCH Lag[5]    0.9749 1.440 1.667  0.7404
## ARCH Lag[7]    1.2622 2.315 1.543  0.8676
##
## Nyblom stability test
## ------------------------------------
## Joint Statistic:  1.5486
## Individual Statistics:
## mu     0.04538
## ar1    0.14555
## ma1    0.16915
## omega  0.06780
## alpha1 0.06816
## beta1  0.05730
## shape  0.06792
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:          1.69 1.9 2.35
## Individual Statistic:     0.35 0.47 0.75
##
## Sign Bias Test
## ------------------------------------
##                    t-value    prob sig
## Sign Bias            0.683 0.49656
## Negative Sign Bias   1.727 0.08794   *
## Positive Sign Bias   0.770 0.44352
## Joint Effect         4.264 0.23429
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## ------------------------------------
##   group statistic p-value(g-1)
## 1    20     14.47       0.7559
## 2    30     14.39       0.9892
## 3    40     36.50       0.5844
## 4    50     29.16       0.9891
##
##
## Elapsed time : 0.1716671
```

12. Do a three-period ahead forecast of the conditional variance.

```
# Forecast Model
predict_rJPM_mo <- ugarchboot(garch1_rJPM_mo,n.ahead=3, method=c("Partial", "Full")[1])
plot(predict_rJPM_mo, which=2)
```