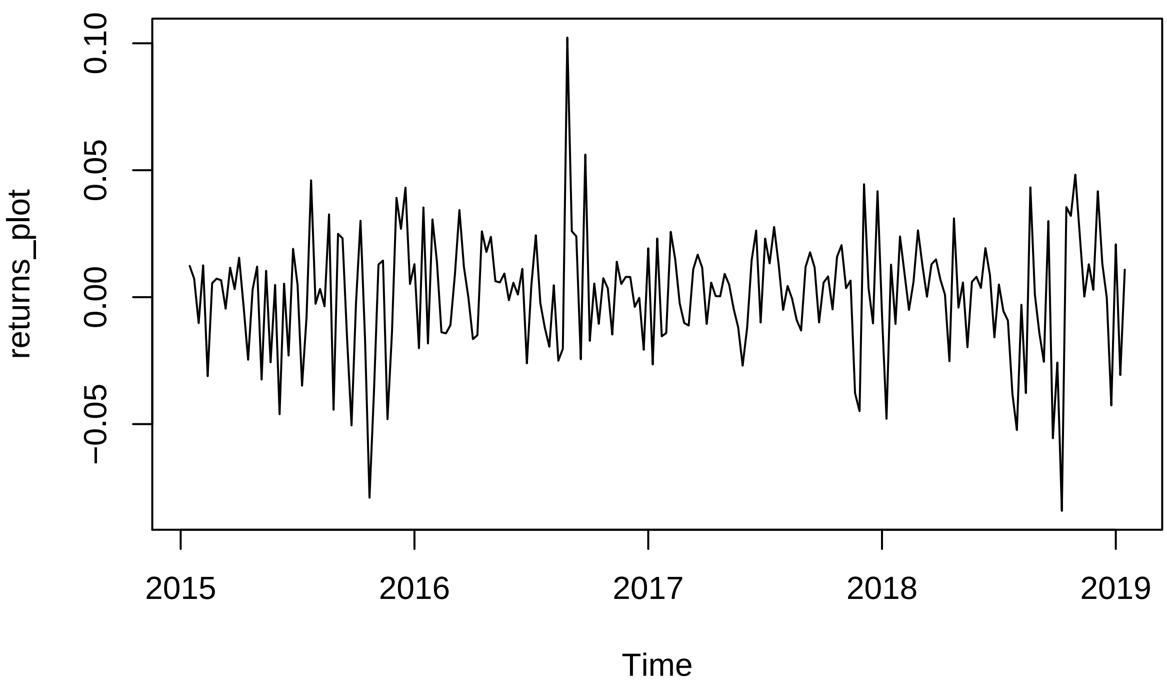
509-HW8-Xinye Xu

Q3

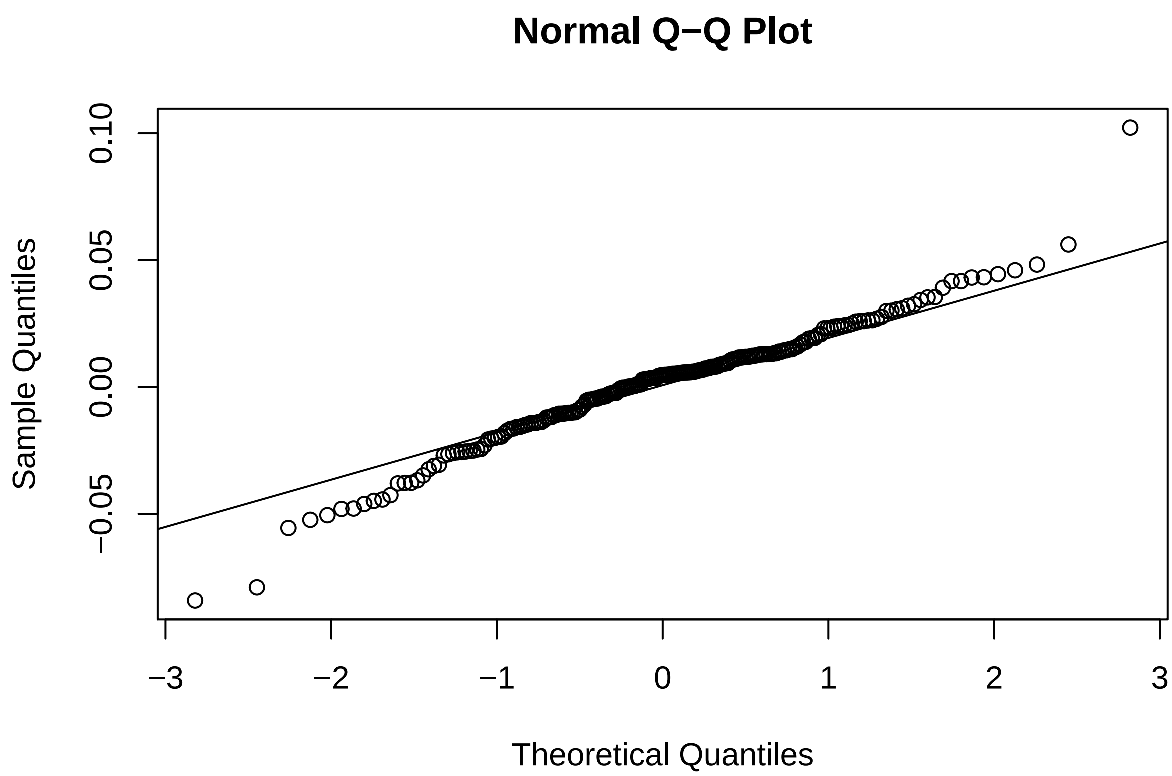
For this problem there is a data set consisting of the recent Russell 2000 weekly data over the past 4 years – it is found in the Data folder on Canvas. (a) Generate plot of the sample auto-correlation of the Russell 2000 weekly returns, and discuss what is observed from these plots. Also, carry out a Box-Ljung test on whether the auto-correlations are all zero out to lag 10, and summarize your results

From the plots of original returns, we can notice sometimes there will be values outside our normal interval. With QQplot, it indicates the heavier tail of the return distribution than normal. According to ACF plot and Box-Ljung, not all auto-correlations are zero out to lag 10, lag3 and 4 seems to be out of the confidence interval in ACF plot; p-value of Box-Ljung is 0.3124, suggesting that the correlations coefficients up to lag 10 is zero.

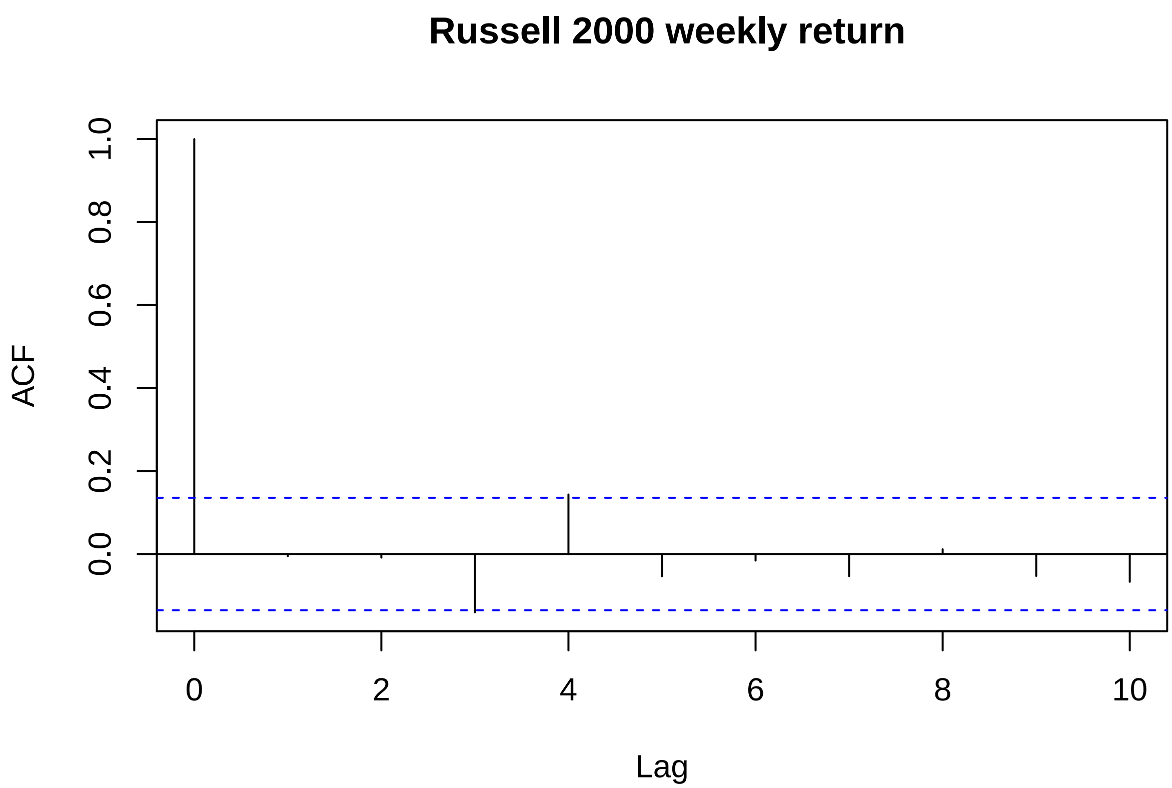
```
data = read.csv('RUT_03_2015-03_2019.csv')
return = data$Adj.Close[2:nrow(data)] / data$Adj.Close[1:nrow(data)-1] - 1
returns_plot <- ts(return, start=c(2015, 3, 23),frequency = 52) # 52 week per year
plot(returns_plot)
```



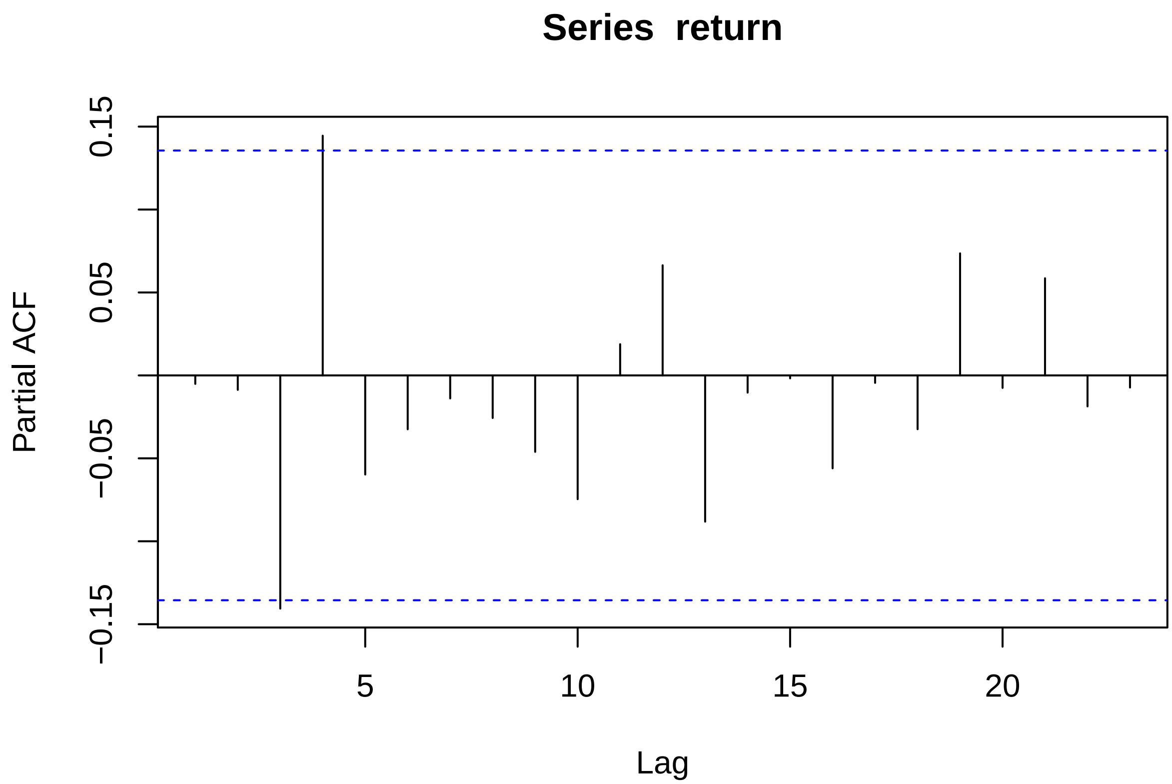

```
qqnorm(return)
qqline(return)
```




```r
acf(as.vector(return), lag=10, main=' Russell 2000 weekly return')
```




```r
pacf(return) # partial auto correlation plot
```




```
Box.test(return, lag=10, type='Ljung')  # Box-Ljung test
```

```
##
##  Box-Ljung test
##
## data:  return
## X-squared = 11.605, df = 10, p-value = 0.3124
```

(b) Derive the optimal AR model utilizing the ar function in R. Provide a comprehensive set of diagnostics and give a clear summary of findings/conclusions.

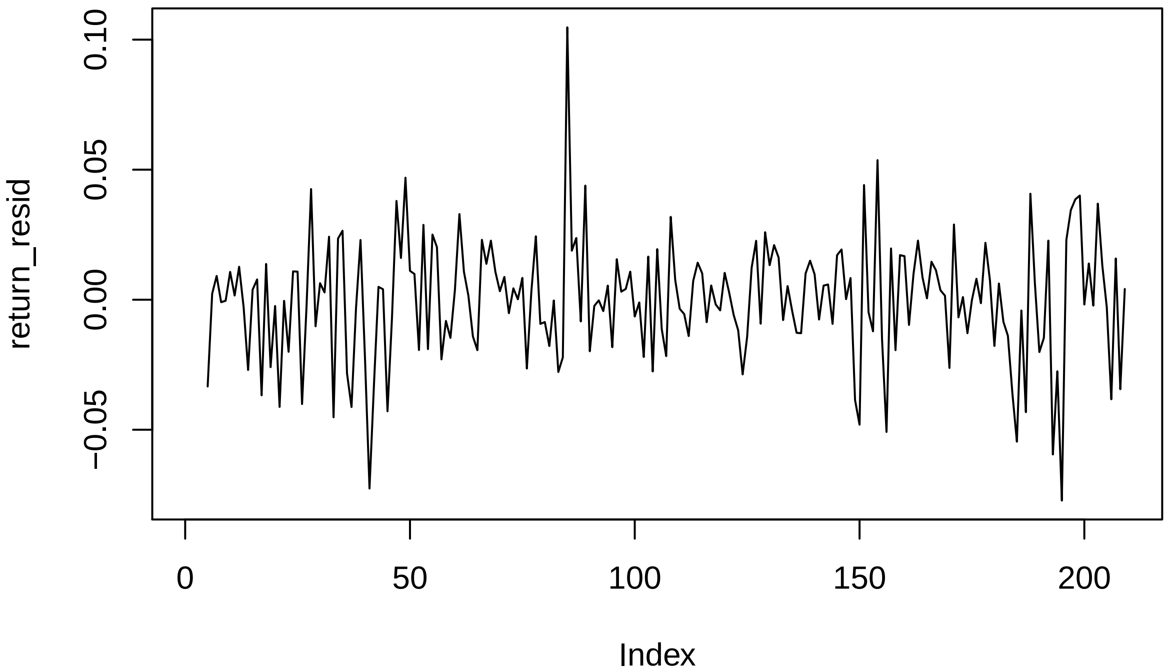
From the ar funciton, we can find that optimal AR model is lag4. Diagonize the residuals, we can notice that residuals are uncorrleated with lags in ACF. Residuals seem to have a little heavy tail, which might not be a strictly white noise. By calculating some summary figures, mean and var of residuals are -0.0001200701 0.0005423549. Both of them are quite small. Then, from arima diagonistic plots, Box-Ljung test p-values are high, which supporting that there is no dependence of residuals.

```
return_ar = ar(return, aic = TRUE, order.max = NULL,  method = c("mle"))
summary(return_ar)
```

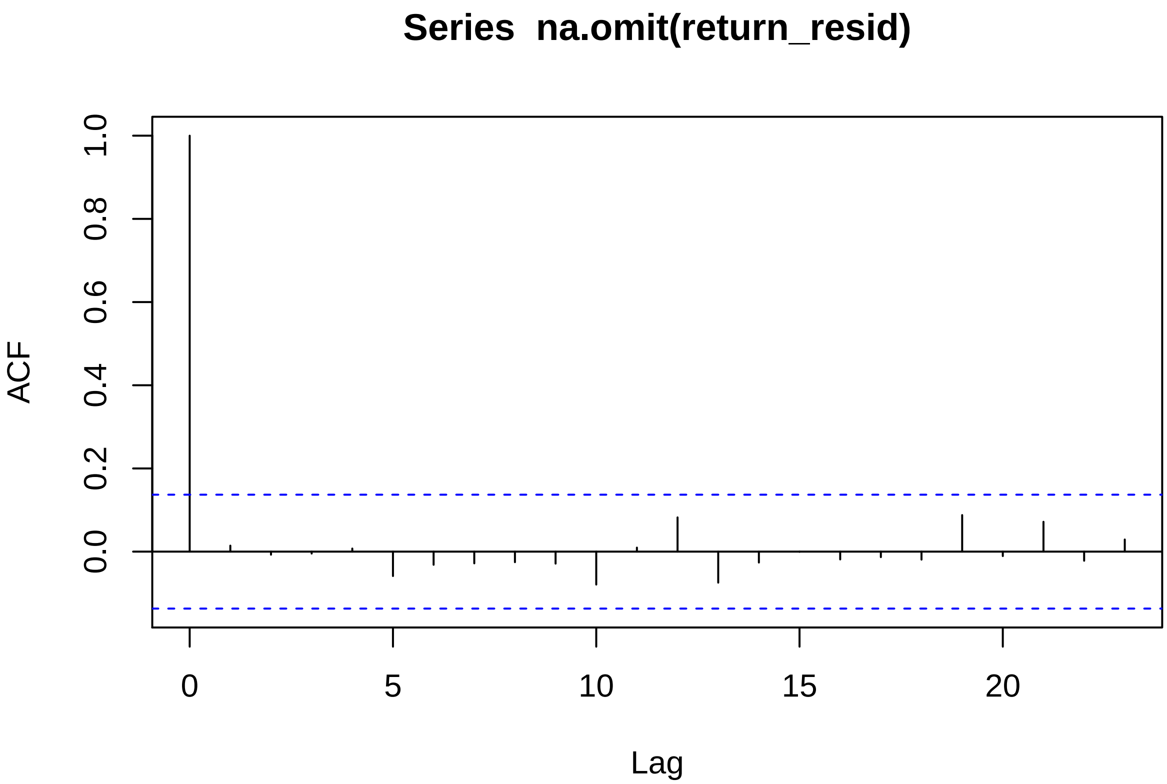
```
##             Length Class  Mode
## order         1    -none- numeric
## ar            4    -none- numeric
## var.pred      1    -none- numeric
## x.mean        1    -none- numeric
## aic          13    -none- numeric
## n.used        1    -none- numeric
## order.max     1    -none- numeric
## partialacf    0    -none- NULL
## resid       209    -none- numeric
## method        1    -none- character
```

```
## series          1    -none- character
## frequency       1    -none- numeric
## call            5    -none- call
## asy.var.coef   16    -none- numeric
```

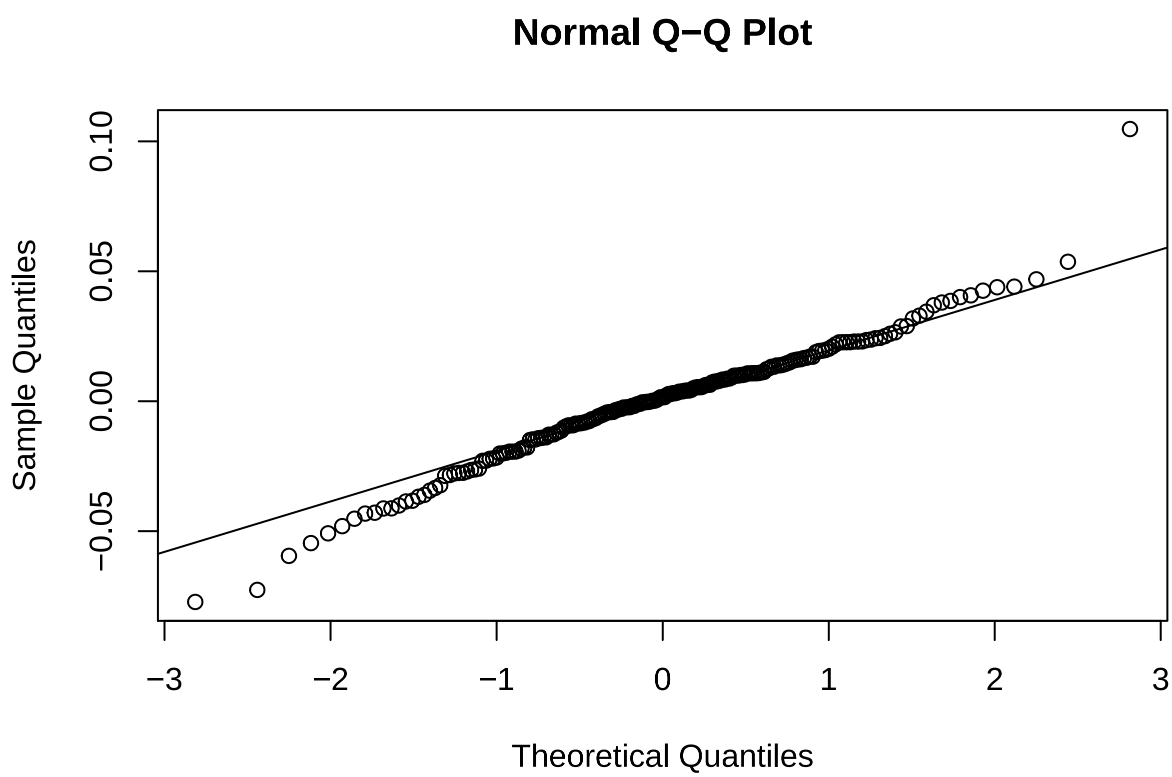
```r
# check residuals:
return_resid = return_ar$resid

plot(return_resid, type = "l")
```



```r
acf(na.omit(return_resid))
```



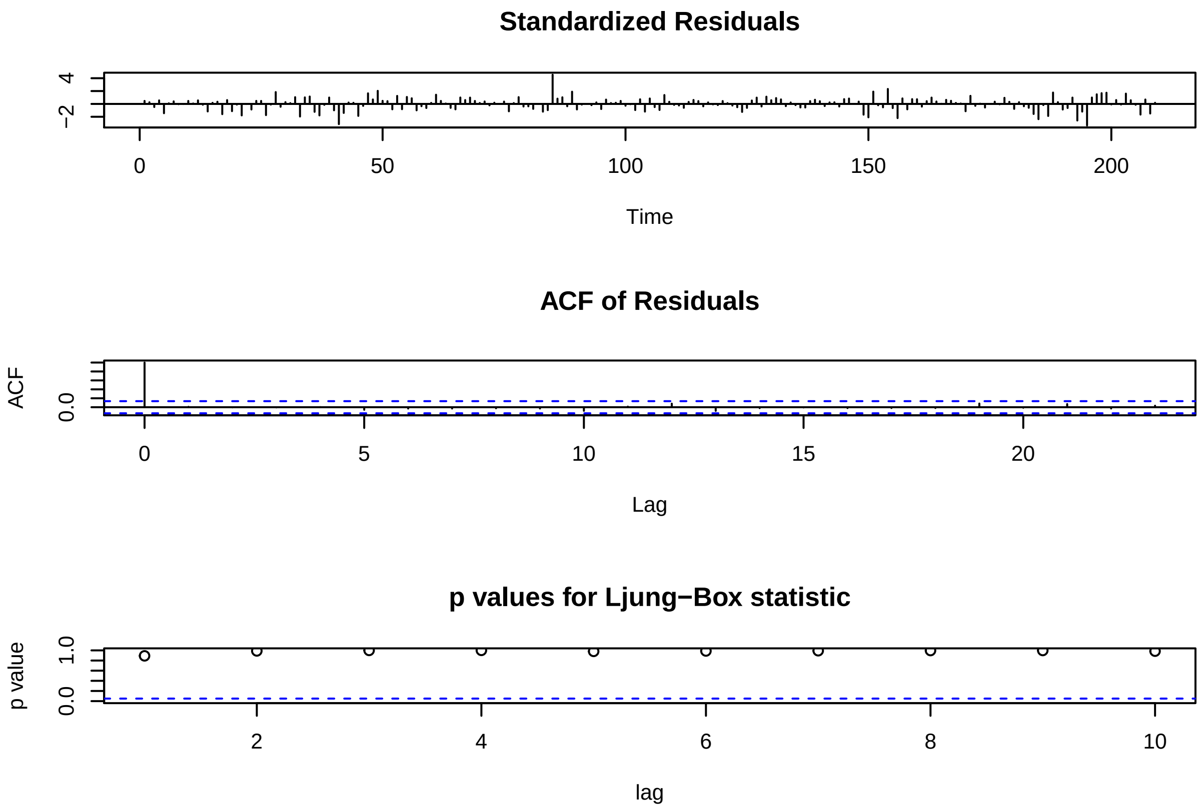

```r
qqnorm(return_resid)
qqline(return_resid)
```



```r
show(c(mean(na.omit(return_resid)), var(na.omit(return_resid))))
```

```
## [1] -0.0001200701  0.0005423549
```

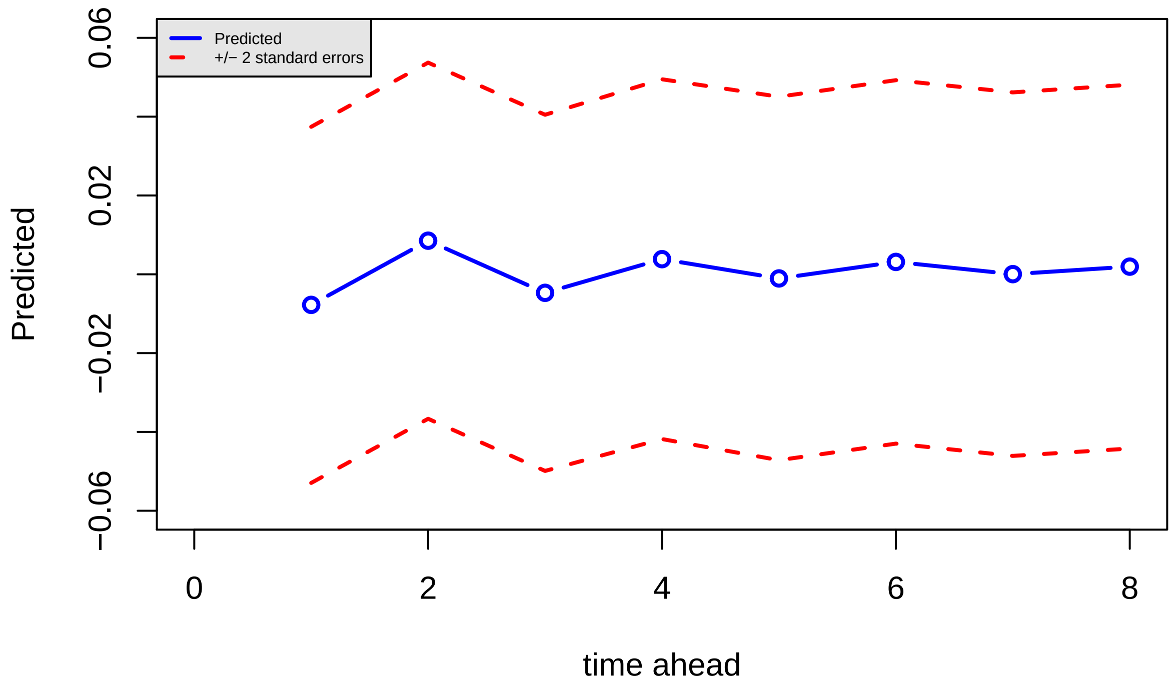
```r
# qq plot of the fitted return data
Re_arima1 <- arima(return, order = c(4,0,0), method = "ML")
tsdiag(Re_arima1)
```



(c)

Do a prediction out to 8 weeks ahead along with 95% confidence intervals – provide numerical values along with a plot of predictions/confidence intervals.

```r
xnew=return[206:209]
forecasts <- predict(return_ar,newdata=xnew,n.ahead = 8, se.fit = TRUE)
plot(c(1:8),forecasts$pred,type='b',lty=1,xlab='time ahead',
 ylab='Predicted',xlim=c(0,8),ylim=c(-0.06,0.06),lwd=2,col='blue')
lines(c(1:8),forecasts$pred+1.96*forecasts$se,lty=2,lwd=2,col='red')
lines(c(1:8),forecasts$pred-1.96*forecasts$se,lty=2,lwd=2,col='red')
legend('topleft', c("Predicted","+/- 2 standard errors"), lty=c(1,2),
lwd=c(2,2),col=c("blue","red"), bg="gray90", cex = 0.5)
```




```
# Confidence interval
confidence_interval_lower_bound = forecasts$pred - 1.96*forecasts$se
confidence_interval_upper_bound = forecasts$pred + 1.96*forecasts$se
confidence_interval = cbind(confidence_interval_lower_bound,confidence_interval_upper_bound)
show(confidence_interval)
```

```
## Time Series:
## Start = 5
## End = 12
## Frequency = 1
##    confidence_interval_lower_bound confidence_interval_upper_bound
##  5                     -0.05294505                      0.03742609
##  6                     -0.03666224                      0.05371833
##  7                     -0.04989628                      0.04048532
##  8                     -0.04180278                      0.04948947
##  9                     -0.04716458                      0.04506120
## 10                     -0.04297130                      0.04925588
## 11                     -0.04608056                      0.04616241
## 12                     -0.04421529                      0.04810406
```