Stat153_hw1_Xiaoying Liu

1.

```r
library(TSA)
```

```
## Warning: package 'TSA' was built under R version 3.4.4
##
## Attaching package: 'TSA'
## The following objects are masked from 'package:stats':
##
##     acf, arima
## The following object is masked from 'package:utils':
##
##     tar
```

```r
library(forecast)
```

```
## Warning: package 'forecast' was built under R version 3.4.4
## Warning in as.POSIXlt.POSIXct(Sys.time()): unknown timezone 'zone/tz/2018e.
## 1.0/zoneinfo/America/Los_Angeles'
```

```r
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.4.4
```

```r
library(gridExtra)
```

```
## Warning: package 'gridExtra' was built under R version 3.4.1
```

```r
#install.packages("gtrendsR")
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.4.4
##
## Attaching package: 'dplyr'
## The following object is masked from 'package:gridExtra':
##
##     combine
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(gtrendsR)
```

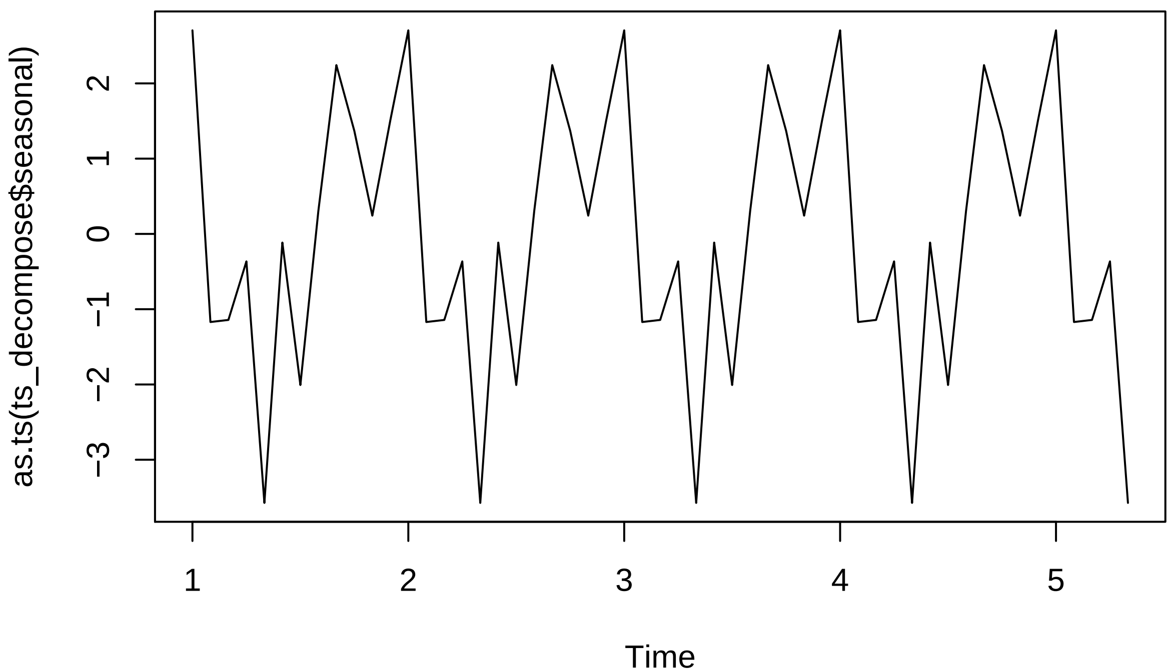
```
## Warning: package 'gtrendsR' was built under R version 3.4.4
```

```r
#microsoft=gtrends("microsoft")
#df=microsoft[[1]]
microsoft=read.csv("multiTimeline2.csv")

#examine data

microsoft=ts(microsoft)#global/past 5 years
t=time(microsoft)
class(microsoft)
```

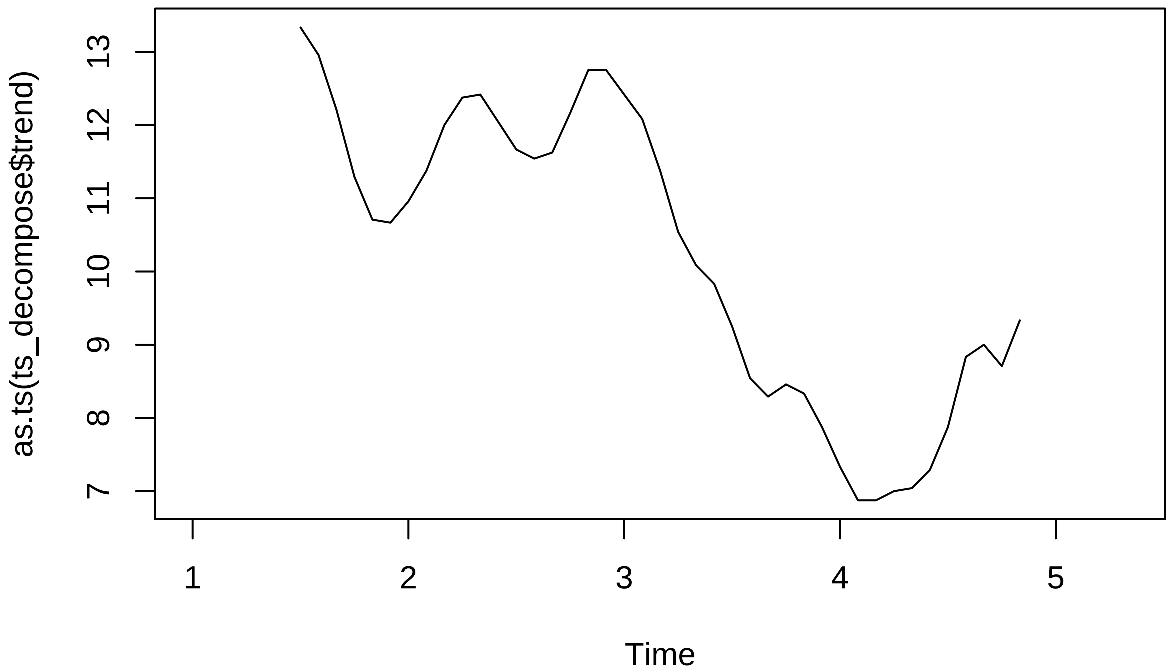
```
## [1] "ts"
```

play with data

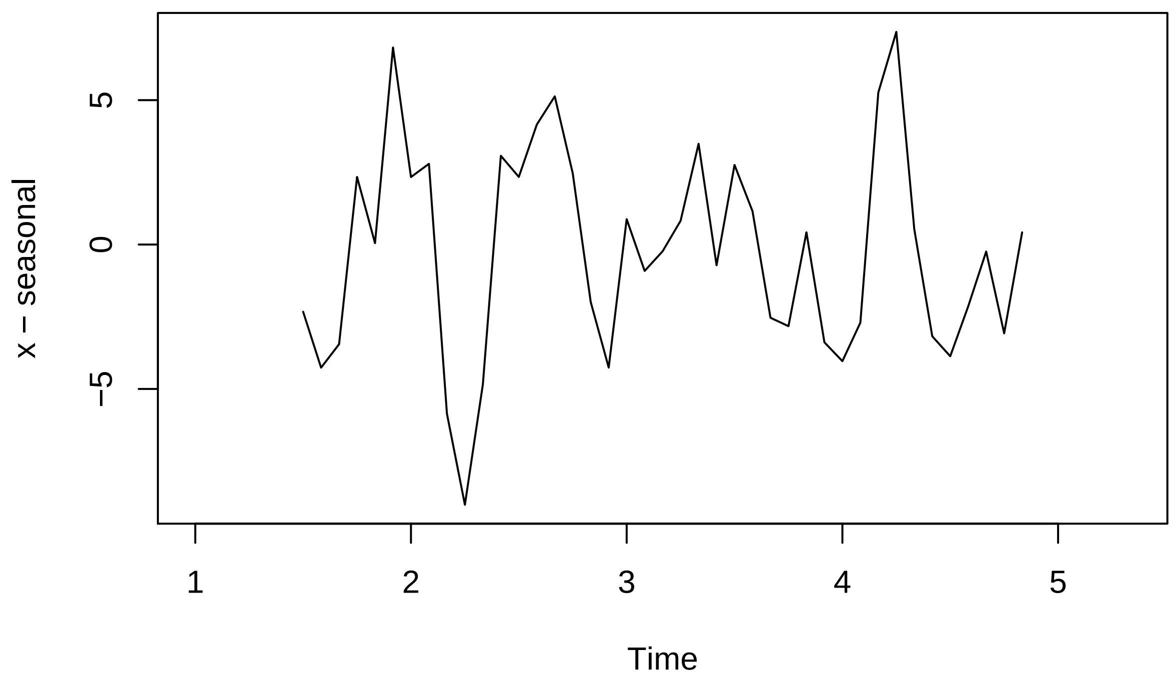
```r
ts=ts(microsoft,frequency=12)
ts_decompose=decompose(ts,"additive")
plot(as.ts(ts_decompose$seasonal))
```




```r
plot(as.ts(ts_decompose$trend))
```




```
plot(as.ts(ts_decompose$random))
```




```
ts_decompose$random
```

```
##            Jan         Feb         Mar         Apr         May         Jun
## 1           NA          NA          NA          NA          NA          NA
## 2  2.33709491  2.79542824 -5.85734954 -9.01012731 -4.84346065  3.07320602
## 3  0.87876157 -0.91290509 -0.23234954  0.82320602  3.48987269 -0.71846065
## 4 -4.03790509 -2.70457176  5.26765046  7.36487269  0.53153935 -3.17679398
## 5           NA          NA          NA          NA          NA
##            Jul         Aug         Sep         Oct         Nov         Dec
## 1 -2.32609954 -4.26359954 -3.45109954  2.34056713  0.04890046  6.82320602
```

```
## 2  2.34056713  4.15306713  5.13223380  2.46556713 -1.99276620 -4.26012731
## 3  2.75723380  1.15306713 -2.53443287 -2.82609954  0.42390046 -3.38512731
## 4 -3.86776620 -2.13859954 -0.24276620 -3.07609954  0.42390046          NA
## 5
```

153 hw questions :

1. (a)what is time plot? Compare and discuss comments(supposed to comment on 3 graphs)

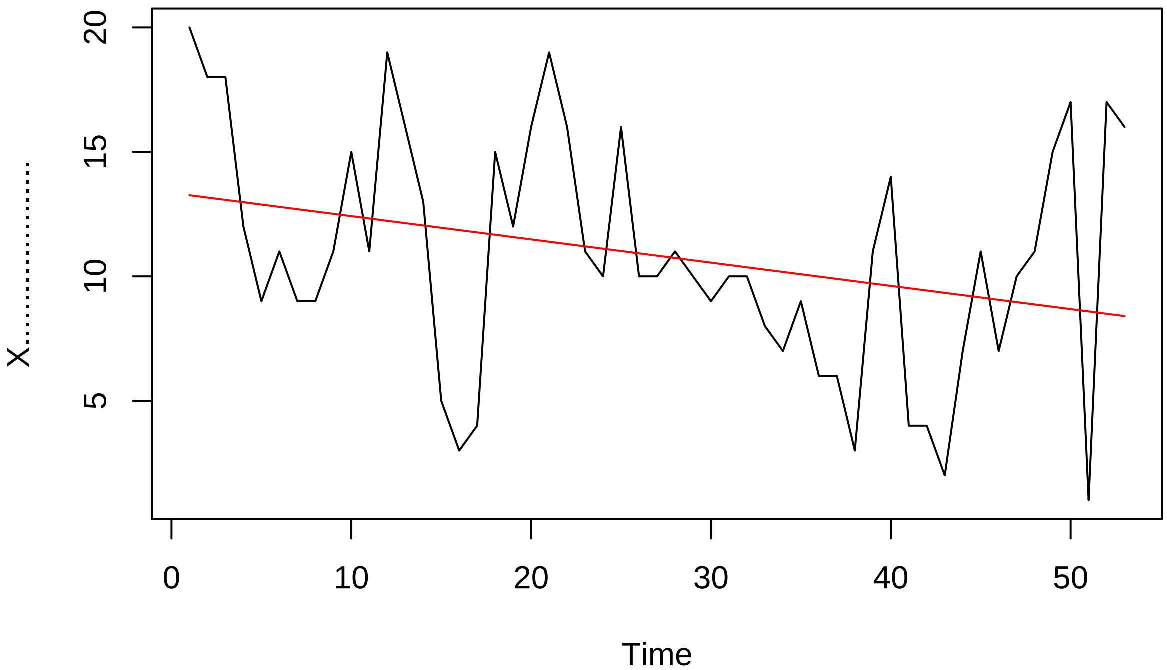
#.(b)Explain smoothing parameter? #(c) is there any trend in differenced data? #(d)isotonic trend, what does it look like?

(a)fit a parametric curve to data.plot of original data and trend estimate. Time plot and correlogram of residuals. Comment on each plots

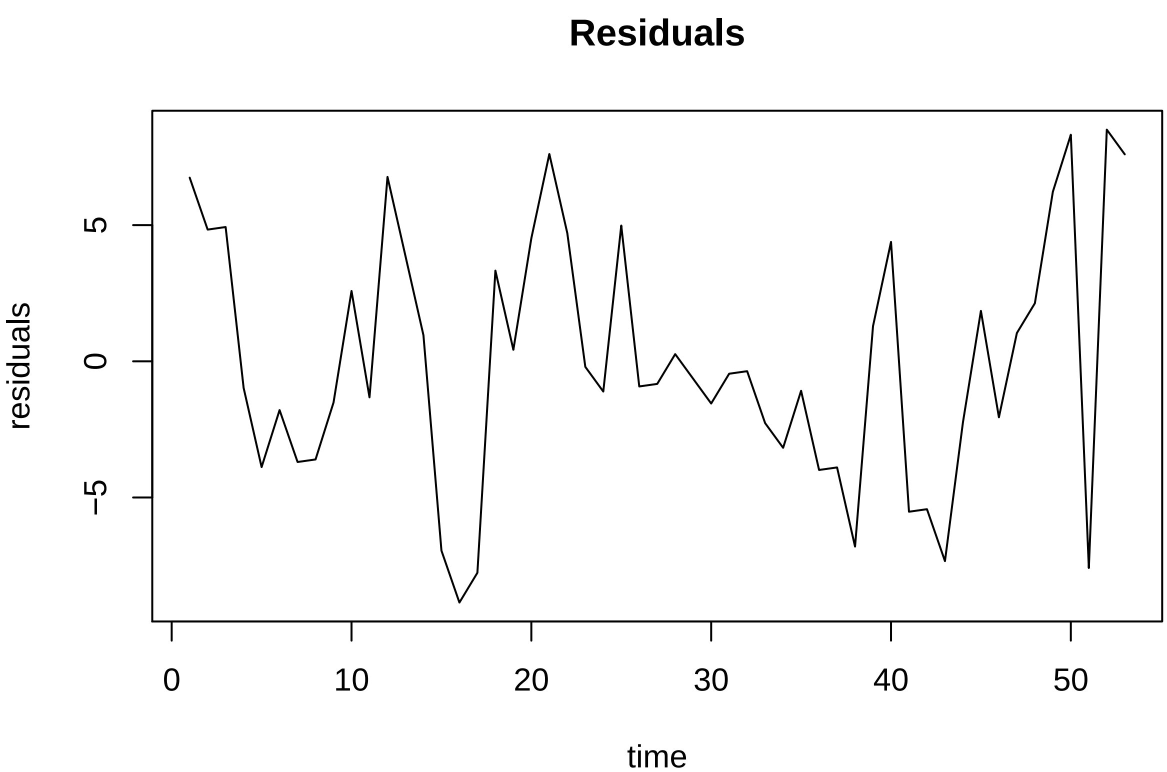
```r
#try linear_fit
linear_fit <- lm(microsoft ~ t)
linear_fit
```

```
##
## Call:
## lm(formula = microsoft ~ t)
##
## Coefficients:
## (Intercept)            t
##    13.35123     -0.09337
```

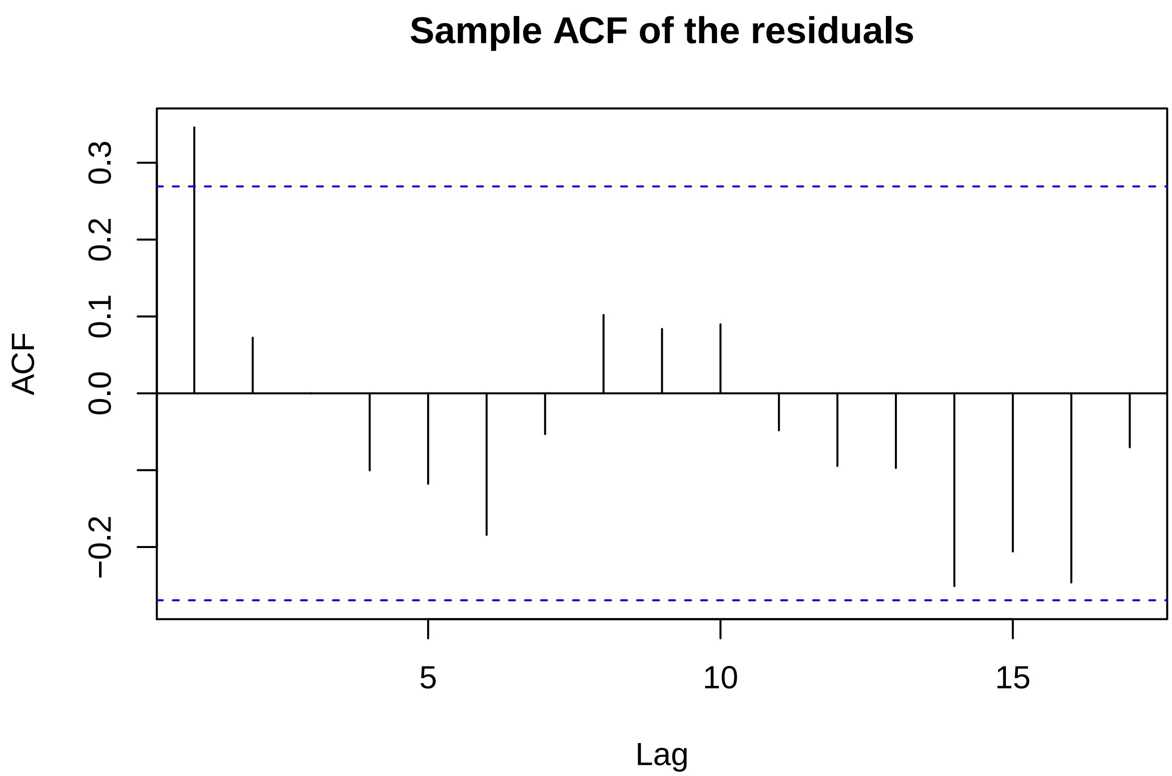
```r
#original data plot and estimate trend
plot(microsoft)
lines(as.numeric(t), linear_fit$fitted.values, col='red')
```




```r
#in the original data plot, it is hard to determine the trend. From the way of the data is collected(we
#residual plot
plot(as.numeric(t), linear_fit$residuals, type='l',
     xlab='time', ylab='residuals', main='Residuals')
```

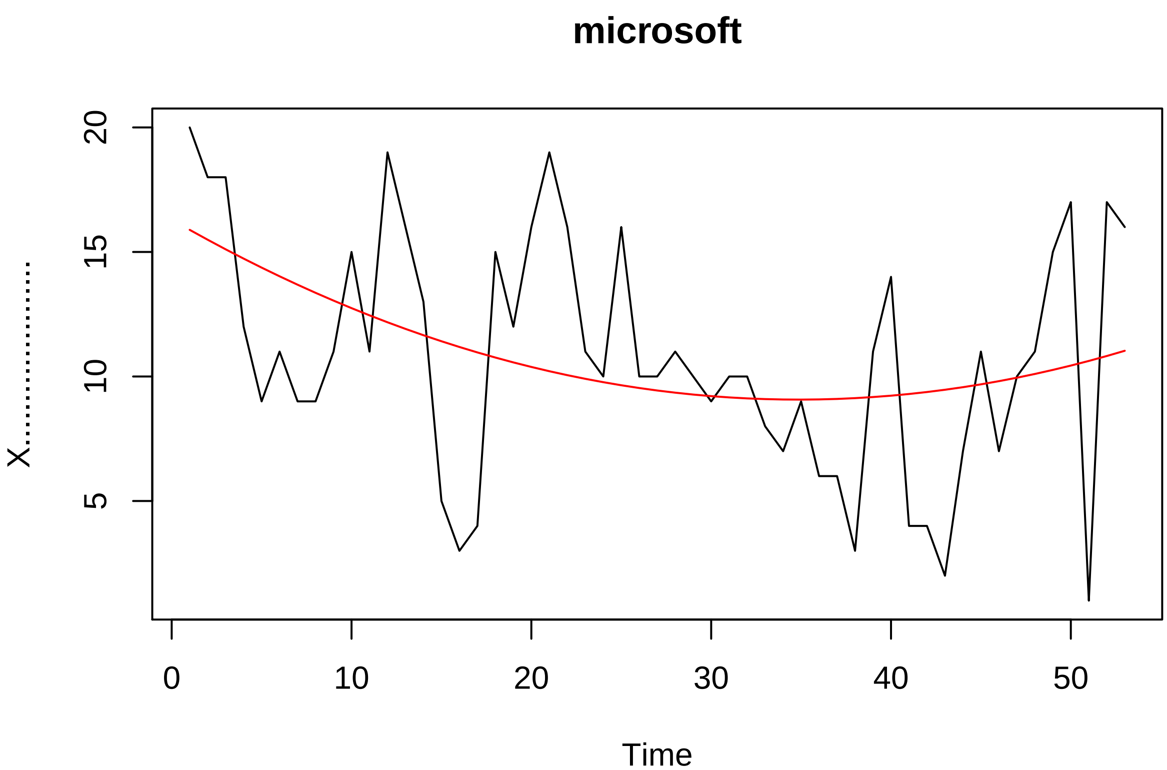



```r
#in the residual plot, we first examine if its mean is close enough to 0, mean(linear_fit$residuals), t
acf(linear_fit$residuals, main='Sample ACF of the residuals')
```

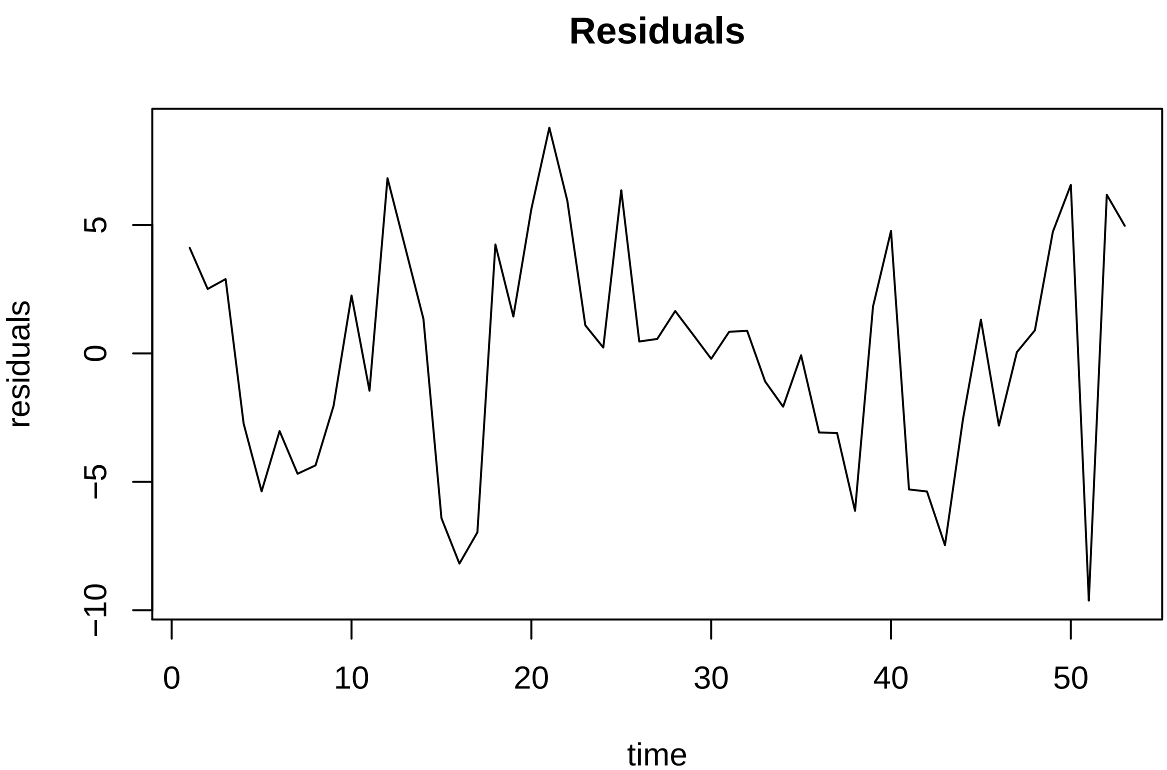



```
#acf of residuals shows that the correlation between residuals are not significantly correlated.(All wi
```

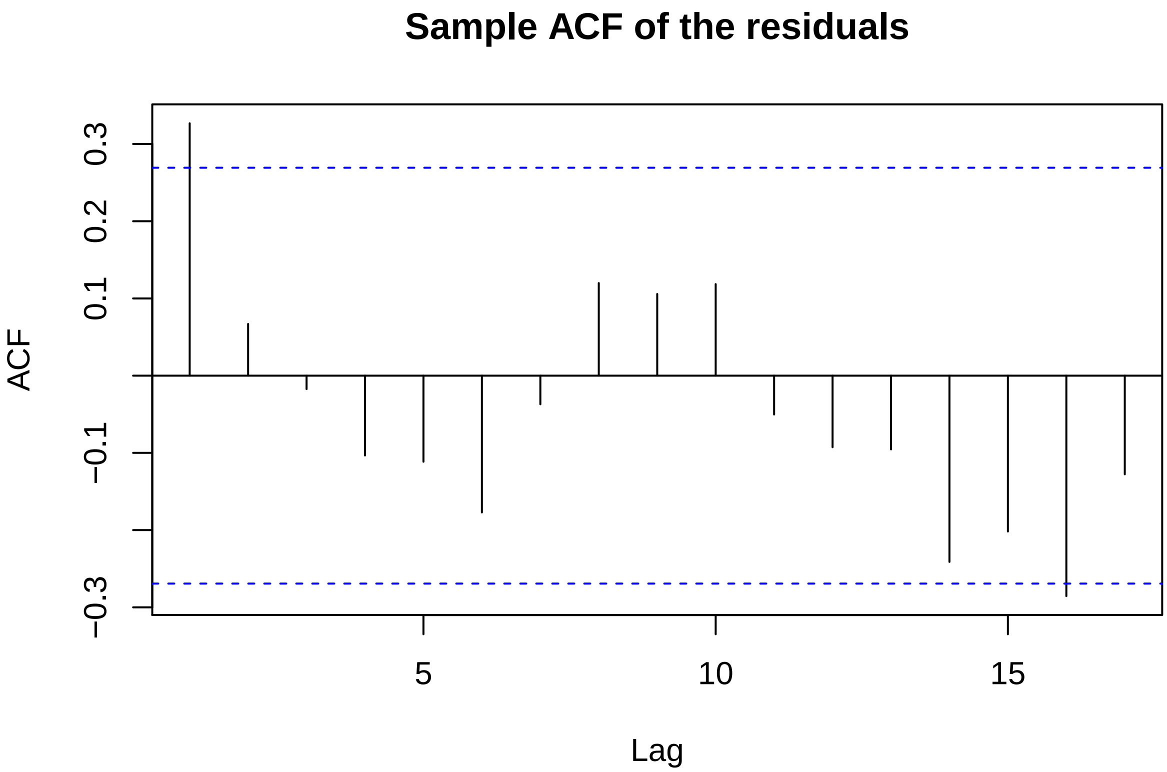
```
#try quadratic fit
quadratic_fit <- lm(microsoft ~t + I(t^2))
#plot original data and estimate trend
plot(microsoft, main='microsoft')
lines(as.numeric(t), quadratic_fit$fitted.values, col='red')
```




```
#in the original data plot, it is hard to determine the trend. From the way of the data is collected(we

#residuals plot
plot(as.numeric(t), quadratic_fit$residuals, type='l',
     xlab='time', ylab='residuals', main='Residuals')
```



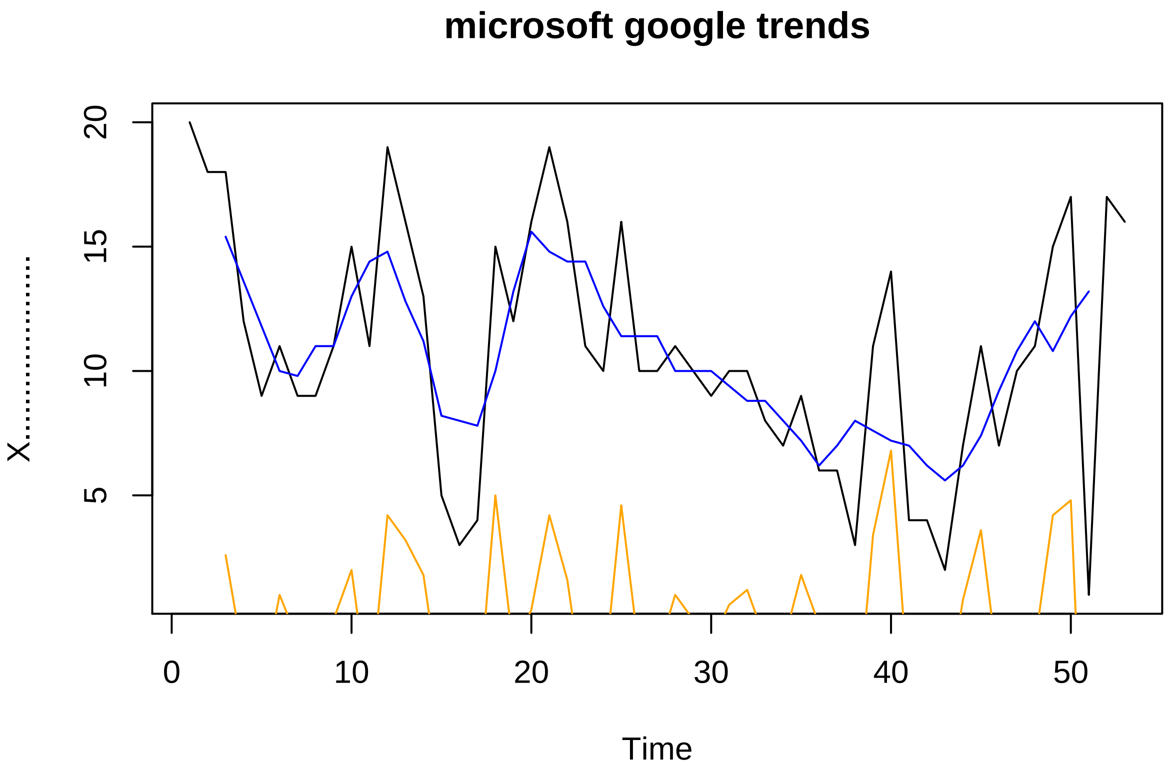
```
#in the residual plot, we first examine if its mean is close enough to 0, mean(quadratic_fit$residuals)
acf(quadratic_fit$residuals, main='Sample ACF of the residuals')
```



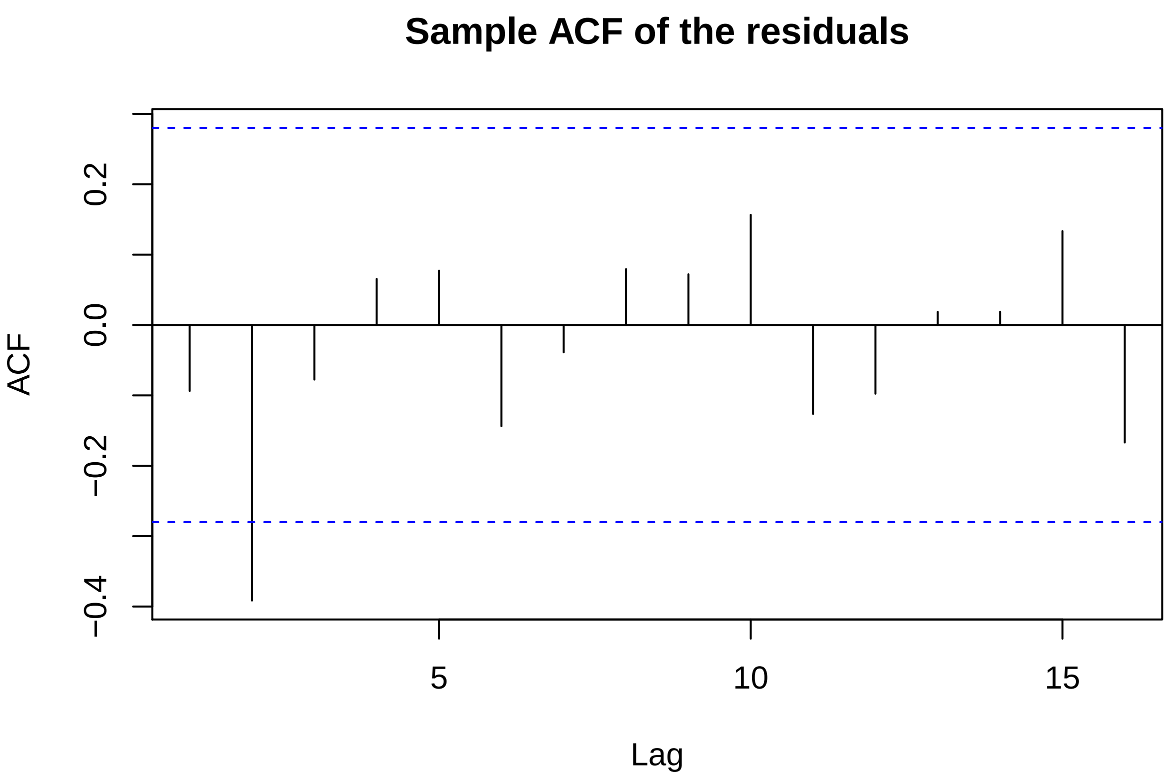

```
#in the residual plot, we first examine if its mean is close enough to 0, mean(linear_fit$residuals), t
#acf of residuals shows that the correlation between residuals are not significantly correlated.(All wi
```

(b)Smooth the trend. Choose smoothing parametr. 3 plots with comments.

```
#smoothing to get a general idea
microsoft_smoothed <- stats::filter(microsoft, rep(1/5, 5), sides=2, method='convolution')
#I choose the
#plot original data and trend estimate
plot(microsoft, main='microsoft google trends')
lines(as.numeric(t), microsoft_smoothed, col='blue')
#there is no obvious trend after smoothing the original data.I tried several smoothing parametr,and cho
#residuals plot
residuals=microsoft-microsoft_smoothed
lines(as.numeric(t),residuals,col="orange")
```



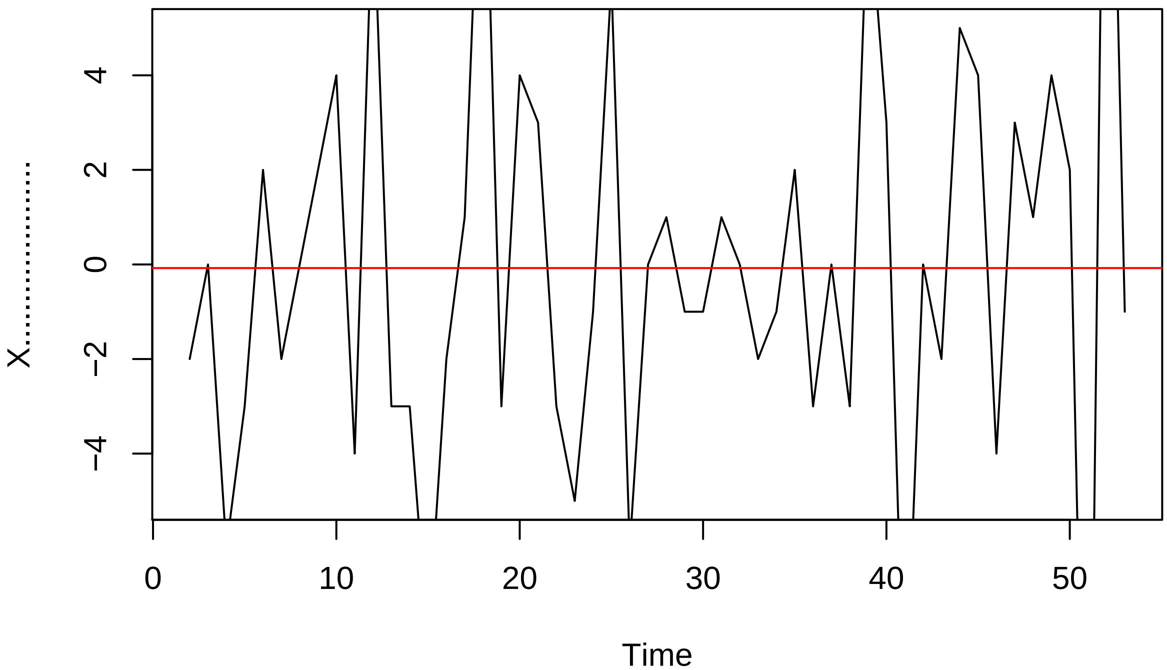

```
residuals <- residuals[!is.na(residuals)]
acf(residuals,main='Sample ACF of the residuals')
```




```
#From residual and acf plot, we can see there is a significant correlation for a certain lag h(h is bet
```

(c)differencing data. Is there any trend?

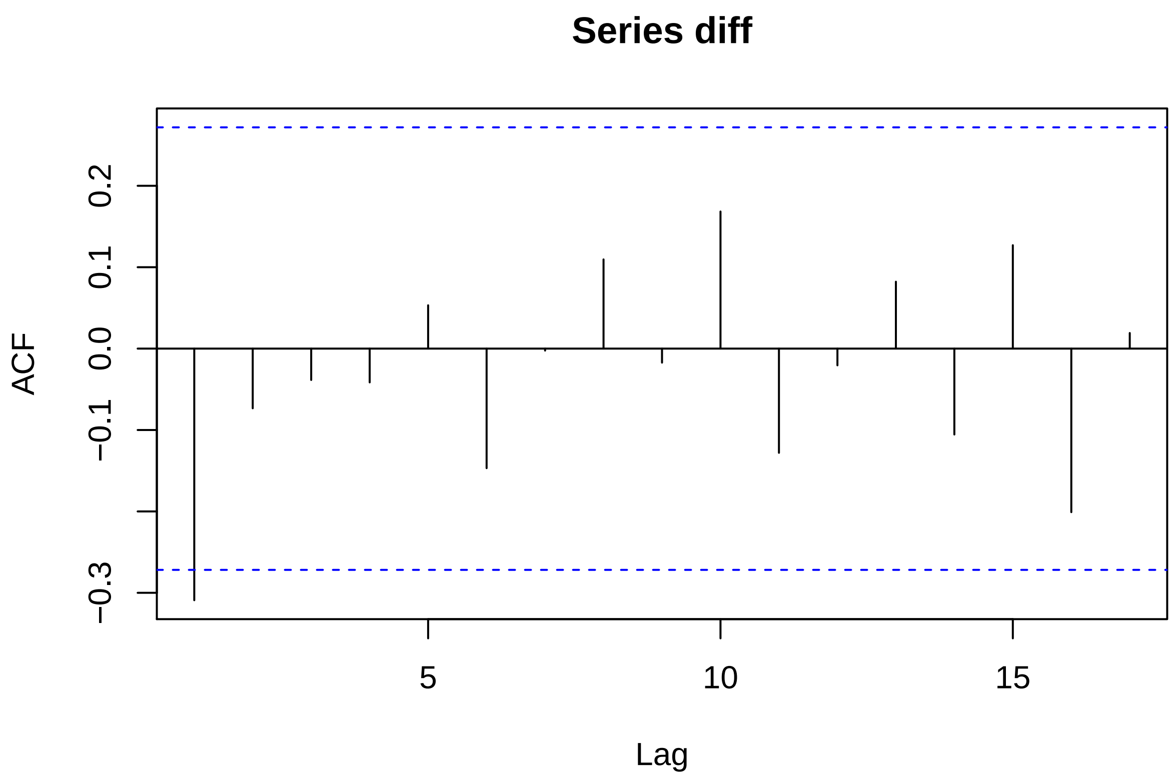
```
diff<- diff(microsoft,differences=1,lag=1)
plot(diff,ylim=c(-5,5))
abline(h=mean(diff), col='red')
```




```
diff_pred <- microsoft[length(microsoft)] + (microsoft[length(microsoft)] - microsoft[1]) / (length(mic
diff_pred
```

```
## [1] 15.92308
```

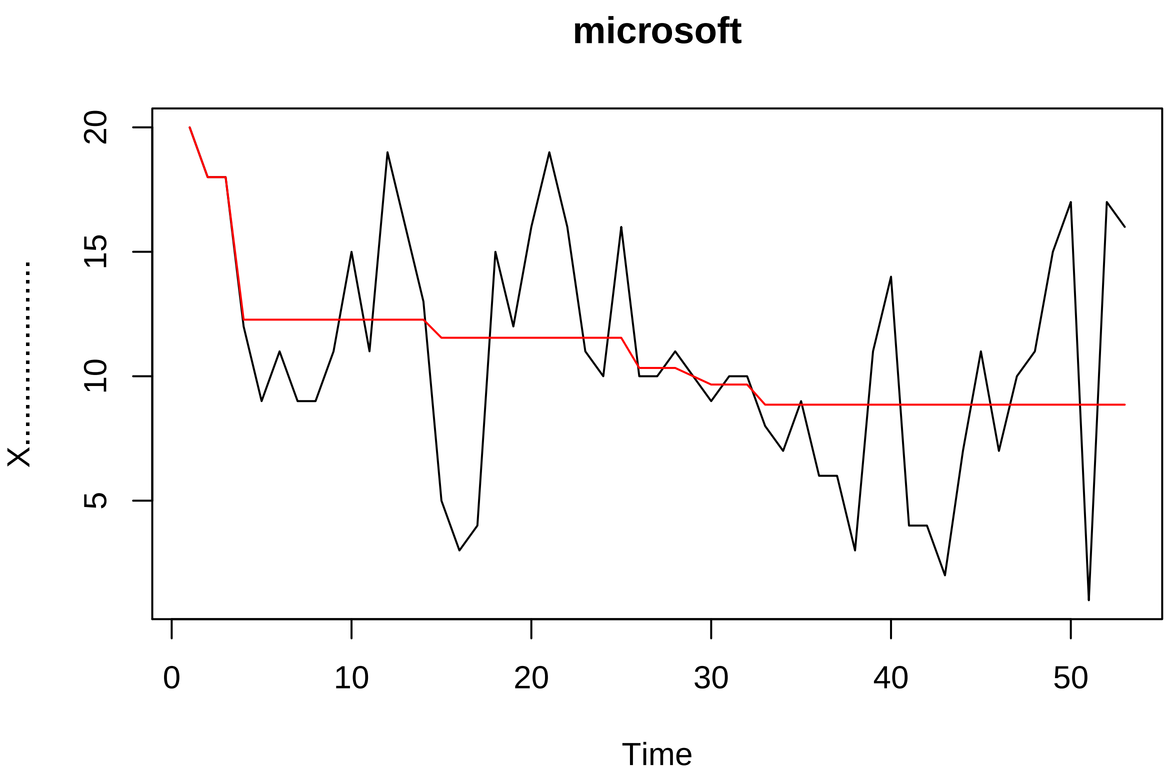
```
acf(diff)
```



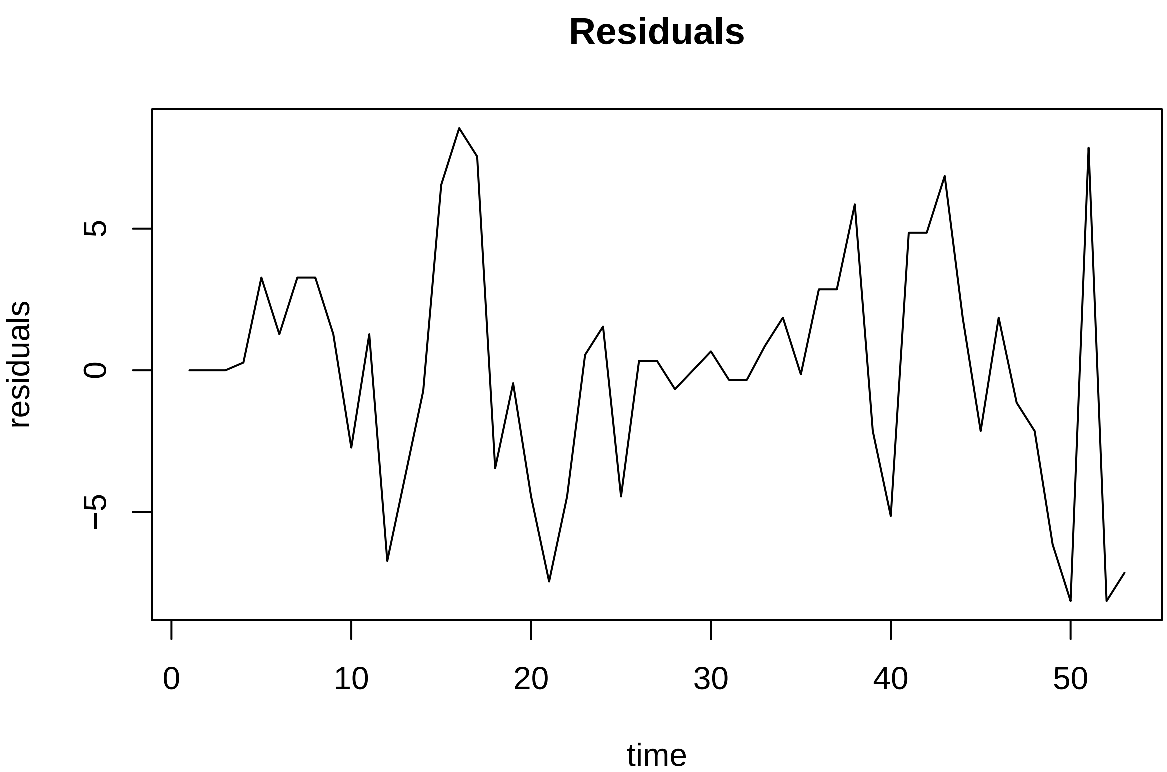

```
#lag=1 exceeds the blue bar significantly, this means that differding model does not capture all inform
```

(d)isotonic estimation. plot and comment.

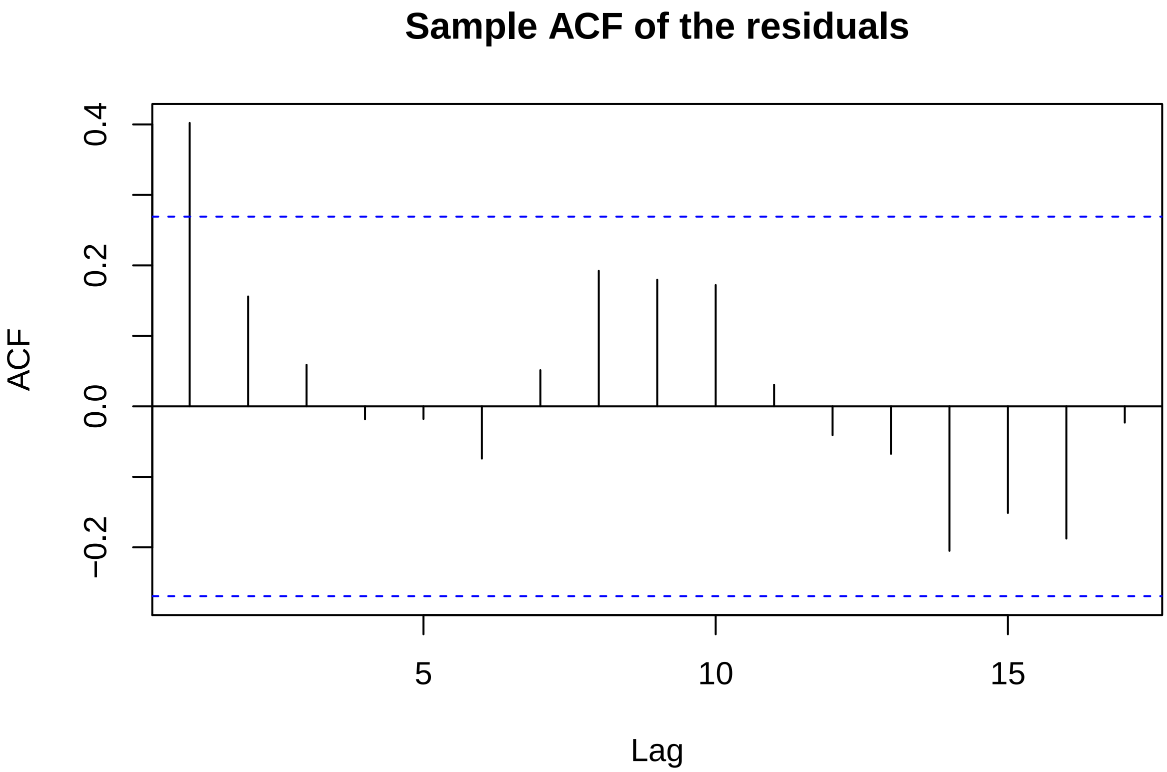
```
iso_fit <- isoreg(x=t, y=-microsoft)
plot(microsoft, main='microsoft')
lines(as.numeric(t), -iso_fit$yf, col='red')
```




```
#the iso_fit plot shows that there is a descending trend.
plot(as.numeric(t), iso_fit$y-iso_fit$yf, type='l',
     xlab='time', ylab='residuals', main='Residuals')
```



```r
acf(as.numeric(iso_fit$y - microsoft), main='Sample ACF of the residuals')
```




```r
#acf of residuals the diff shows a white-noise like pattern.We allow 5% to be outside the blue band. It
```