hw

Nicole Huang

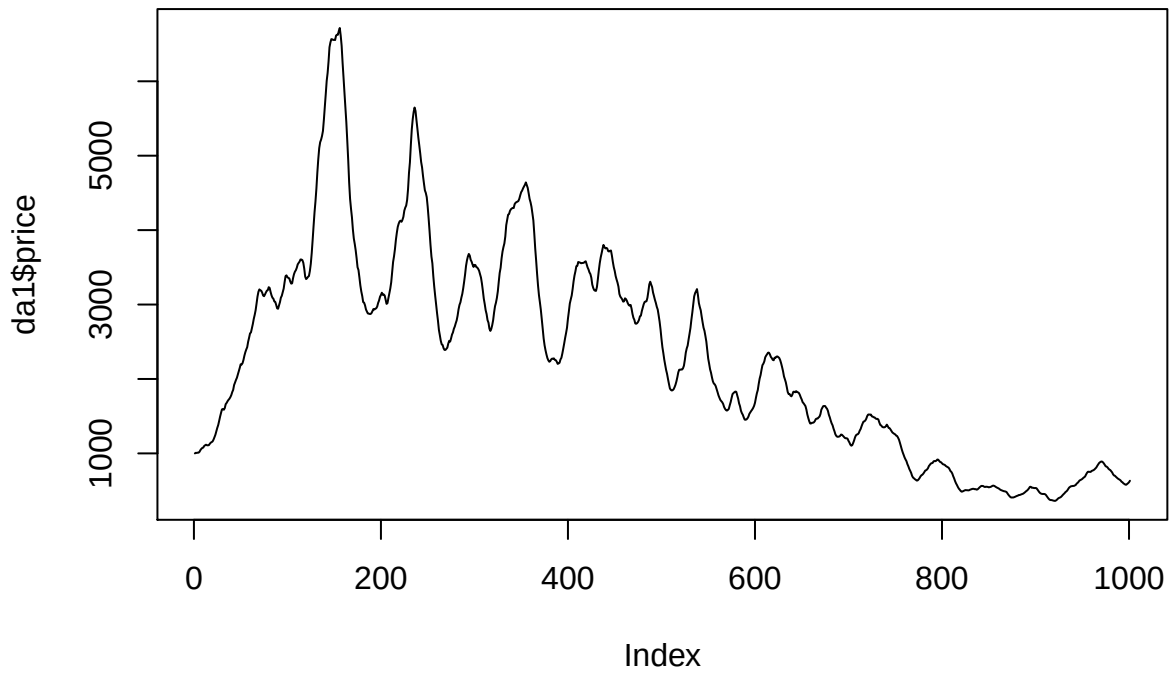
2020/04/20

```
library(maxLik)
```

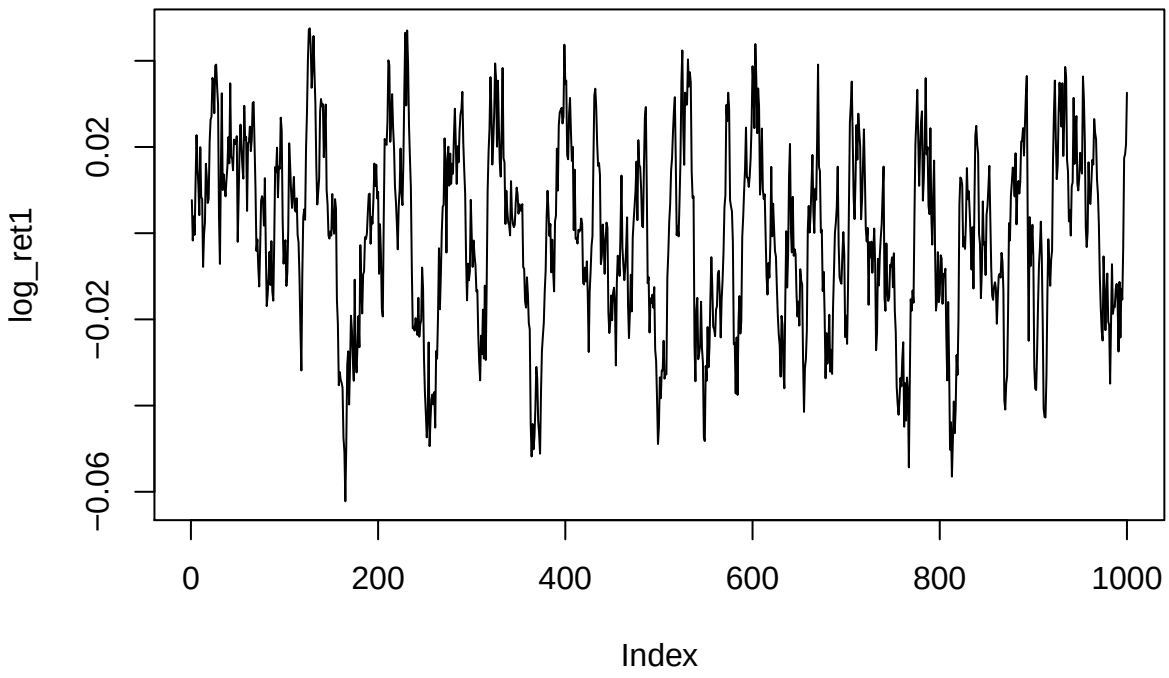
```
## Warning: package 'maxLik' was built under R version 4.0.5
```

```
## Warning: package 'miscTools' was built under R version 4.0.5
```

```
#2
da1=read.csv("data1.csv")
n1 <- length(da1$price)
sim_ret1 <- diff(da1$price)/da1$price[1:n1-1]
log_ret1 <- diff(log(da1$price))
plot(da1$price,type='l')
```




```
plot(log_ret1,type='l')
```




```
rt = log_ret1[-1]
rt_1 = log_ret1[-length(log_ret1)]
likelihoodf = function(param){
  phi0 <- param[1]
  phi1 <- param[2]
  sigma <- param[3]
  f = (1/(2 * pi * sigma)^(0.5)) * exp((-1 / (2 * sigma)) * (rt - phi0 - phi1 * rt_1)^2)
  lnf = sum(log(f))
  return(lnf + log(((1 - phi1^2) / (2 * pi * sigma ))^(0.5)*exp((-(1 - phi1^2) / (2 * sigma ))* (rt[1]
}

likeli = function(param){return(-likelihoodf(param))} #optim func:find minmum, so we find -likeli

p <- optim(c(phi0=0.5,phi1=0.5,sigma=0.5),likeli)
p
```

```
## $par
##          phi0          phi1         sigma
## -3.125592e-05  8.879247e-01  9.611325e-05
##
## $value
## [1] -3205.246
##
## $counts
## function gradient
##      430       NA
```

```
##
## $convergence
## [1] 0
##
## $message
## NULL
```

```
result <- maxLik(likelihoodf, start = c(0.5,0.5,0.5))
result
```

```
## Maximum Likelihood estimation
## Newton-Raphson maximisation, 15 iterations
## Return code 8: successive function values within relative tolerance limit (reltol)
## Log-Likelihood: 3205.246 (3 free parameter(s))
## Estimate(s): -3.277728e-05 0.8879218 9.612063e-05
```

```
grad = function(param){
          phi0 <- param[1]
          phi1 <- param[2]
          sigma <- param[3]
        f1=(-1/(2*sigma))*(sum((-2)*(rt-phi0-phi1*rt_1 ))+2*(1-phi1^2)*(rt[1]-phi0/(1-phi1))*(-1/(1-phi
        f2=(phi1/(1-phi^1))-(-1/(2*sigma))*(sum((-2)*(rt_1)*(rt-phi0-phi1*rt_1 ))-2*phi1*(rt[1]-phi0/(
        f3=-(n1/2*sigma)-(1/(2*sigma)^2)*((sum(rt-phi0-phi1*rt_1)^2+(1-phi1^2)*(rt[1]-phi0/(1-phi1))^2
        return(c(f1,f2,f3))
}

optim(c(phi0=0.5,phi1=0.5,sigma=0.5),likeli,grad)
```

```
## $par
##          phi0          phi1         sigma
## -3.125592e-05  8.879247e-01  9.611325e-05
##
## $value
## [1] -3205.246
##
## $counts
## function gradient
##      430       NA
##
## $convergence
## [1] 0
##
## $message
## NULL
```

```
model=ar(log_ret1,order.max = 1,method = 'mle')
#model=arima(log_ret1,c(1,0,0))
m=c(model$ar,model$var.pred)
print(m)
```

```
## [1] 8.880462e-01 9.613293e-05
```