时间序列分析Lab4 实验报告

和泳毅 PB19010450

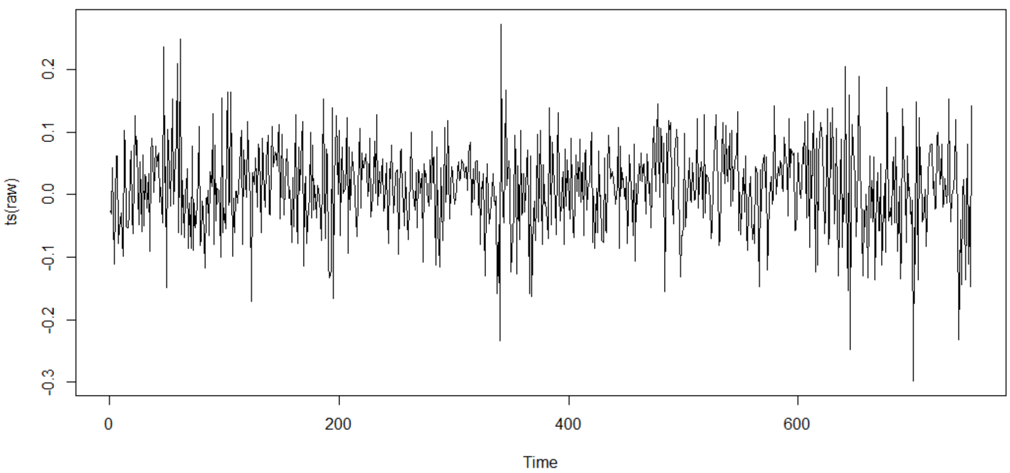
3.6

（a）

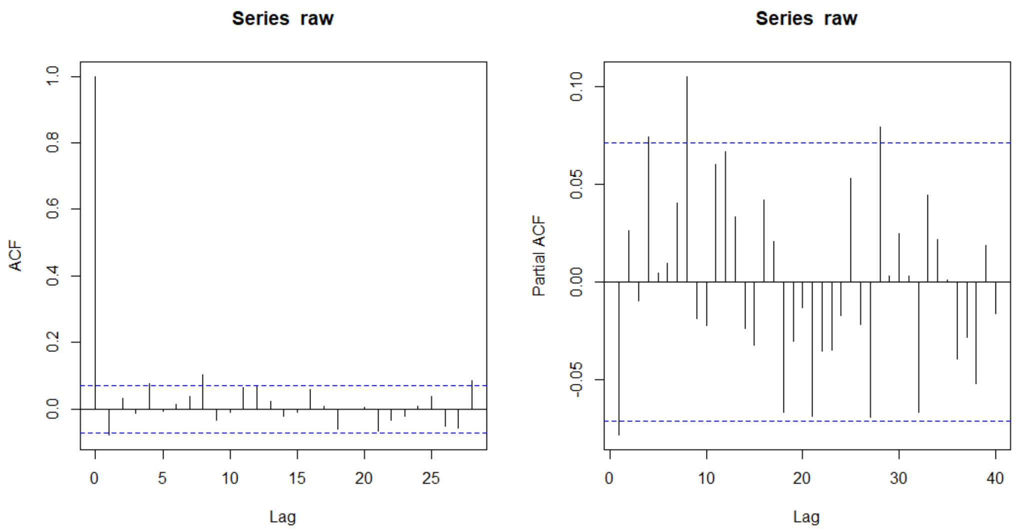
读入数据后计算对数收益率：

da	num [1:751] -0.02593 -0.03053 0.04331 -0.10566 -0.00847 ...
raw	num [1:751] -0.02627 -0.03101 0.0424 -0.11167 -0.00851 ...

其中da为简单收益率，raw为对数收益率。绘制时序图：



绘制ACF和PACF：



ACF图显示一阶时的数值超过了5%置信区间，接下来进一步验证序列的相关性，我们分别选择 lag = 1 和 lag = 12 进行 Ljung-Box 检验：

```
> Box.test(raw, lag = 1, type = 'Ljung')

    Box-Ljung test

data:  raw
X-squared = 4.6403, df = 1, p-value = 0.03123

> Box.test(raw, lag = 12, type = 'Ljung')

    Box-Ljung test

data:  raw
X-squared = 27.236, df = 12, p-value = 0.007144
```

p-value均小于5%，说明序列在5%置信水平下显著自相关。下面对模型进行拟合，在拟合之前先进行平稳性检验：

```
> adf.test(raw)

    Augmented Dickey-Fuller Test

data:  raw
Dickey-Fuller = -7.7125, Lag order = 9, p-value = 0.01
alternative hypothesis: stationary
```

序列是平稳的，根据ACF图，选择MA(1)模型：

```
arima(x = raw, order = c(0, 0, 1))

Coefficients:
          ma1  intercept
      -0.0744     0.0108
s.e.   0.0353     0.0024

sigma^2 estimated as 0.004963:  log likelihood = 926.65,  aic = -1847.29
```

(b)

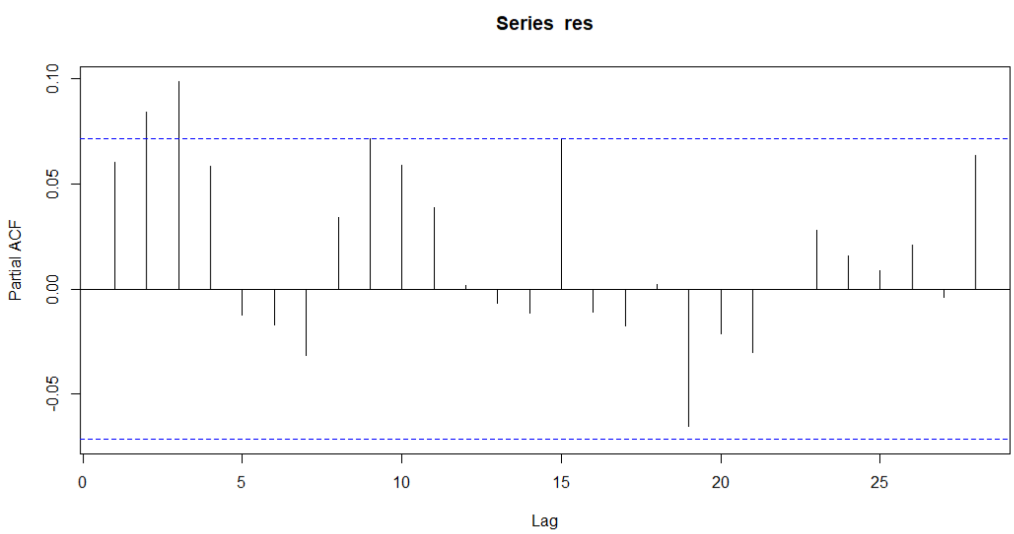
下面对模型残差平方的6个间隔和12个间隔做Ljung-Box 检验：

```
> Box.test(m$residuals ^ 2, lag = 6, type = 'Ljung')

    Box-Ljung test

data:  m$residuals^2
X-squared = 21.613, df = 6, p-value = 0.001423

> Box.test(m$residuals ^ 2, lag = 12, type = 'Ljung')

    Box-Ljung test

data:  m$residuals^2
X-squared = 32.27, df = 12, p-value = 0.001257
```

p-value均显著小于5%，说明序列具有明显的ARCH效应。

(c)

绘制PACF图:



选取ARCH(3)模型：

```
Title:
 GARCH Modelling

Call:
 garchFit(formula = ~arma(0, 1) + garch(3, 0), data = raw, trace = F)

Mean and Variance Equation:
 data ~ arma(0, 1) + garch(3, 0)
<environment: 0x00000212d1413748>
 [data = raw]

Conditional Distribution:
 norm

Coefficient(s):
        mu         ma1       omega      alpha1      alpha2      alpha3
 0.0120080  -0.0751724   0.0040608   0.0197029   0.0748576   0.0842135

Std. Errors:
 based on Hessian

Error Analysis:
         Estimate  Std. Error  t value Pr(>|t|)
mu      0.0120080   0.0023572    5.094  3.5e-07 ***
ma1    -0.0751724   0.0357058   -2.105   0.0353 *
omega   0.0040608   0.0003239   12.536  < 2e-16 ***
alpha1  0.0197029   0.0347459    0.567   0.5707
alpha2  0.0748576   0.0374821    1.997   0.0458 *
alpha3  0.0842135   0.0387548    2.173   0.0298 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Log Likelihood:
 933.9829    normalized:  1.243652

Description:
 Mon May 16 13:36:19 2022 by user: Yorick He


Standardised Residuals Tests:
                                Statistic p-Value
 Jarque-Bera Test   R    Chi^2  25.34371  3.138226e-06
 Shapiro-Wilk Test  R    W      0.994162  0.005351761
 Ljung-Box Test     R    Q(10)  16.51536  0.08579957
 Ljung-Box Test     R    Q(15)  27.59655  0.02423537
 Ljung-Box Test     R    Q(20)  33.85984  0.02708862
 Ljung-Box Test     R^2  Q(10)  9.371613  0.4972363
```

```
 Ljung-Box Test     R^2  Q(15)  14.96658  0.4538275
 Ljung-Box Test     R^2  Q(20)  16.47866  0.6865206
 LM Arch Test       R    TR^2   11.94917  0.4497707

Information Criterion Statistics:
      AIC       BIC       SIC      HQIC
-2.471326 -2.434404 -2.471452 -2.457100
```

代码：

```
library(fBasics)
library(tseries)
library(fGarch)
da = read.table("D:/USTC/时间序列分析/data/m-mrk4608.txt", header = T)
[,2]
raw = log(1 + da)

plot(ts(raw))
par(mfrow = c(1, 2))
acf(raw)
pacf(raw,lag.max = 40)

Box.test(raw, lag = 1, type = 'Ljung')
Box.test(raw, lag = 12, type = 'Ljung')

adf.test(raw)
m = arima(raw, order = c(0, 0, 1))

Box.test(m$residuals ^ 2, lag = 6, type = 'Ljung')
Box.test(m$residuals ^ 2, lag = 12, type = 'Ljung')

res =m$residuals ^ 2
par(mfrow = c(1, 1))
pacf(res)

model = garchFit(~ arma(0, 1) + garch(3, 0), data = raw, trace = F)
summary(model)
```

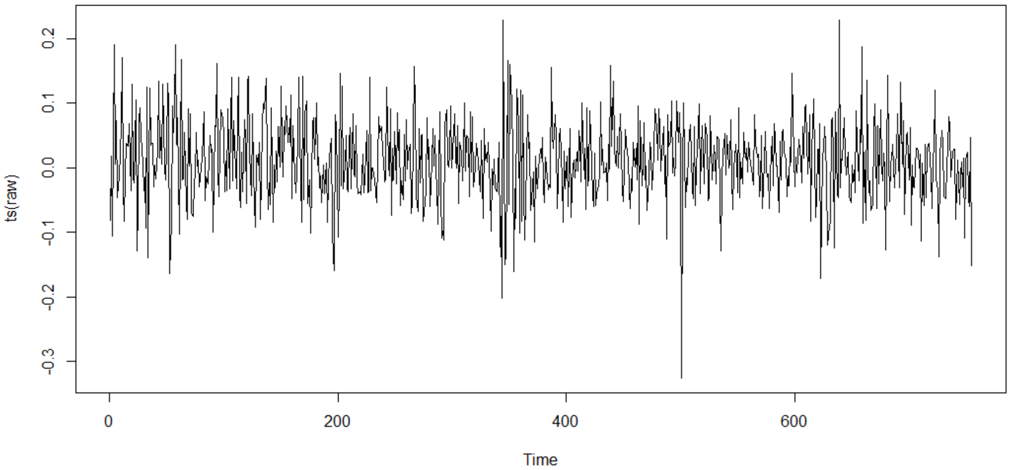
3.7

(a)

读入数据，将简单收益率转化为对数收益率：

da	num [1:755] -0.07792 0.01859 -0.1 0.20988 0.00513 ...
raw	num [1:755] -0.08113 0.01842 -0.10536 0.19052 0.00511 ...

其中da为简单收益率，raw为对数收益率。绘制时序图：



对对数收益率进行间隔为6和12的 Ljung-Box 检验 ：

```
> Box.test(raw,lag = 6,type = 'Ljung')

    Box-Ljung test

data:  raw
X-squared = 13.996, df = 6, p-value = 0.02968

> Box.test(raw,lag = 12,type = 'Ljung')

    Box-Ljung test

data:  raw
X-squared = 27.688, df = 12, p-value = 0.006143

```

发现p-value均小于5%，说明在5%的显著水平下可认为对数收益率在间隔为6和12时存在相关性。

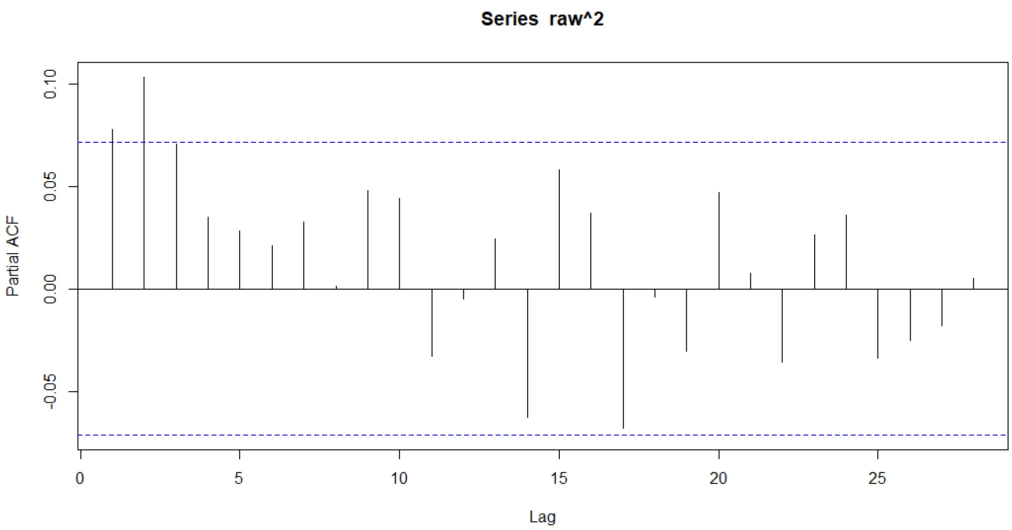
对对数收益率的平方做 Ljung-Box 检验：

```
> Box.test(raw^2,lag = 6,type = 'Ljung')

    Box-Ljung test

data:  raw^2
X-squared = 24.447, df = 6, p-value = 0.0004321

> Box.test(raw^2,lag = 12,type = 'Ljung')

    Box-Ljung test

data:  raw^2
X-squared = 32.664, df = 12, p-value = 0.001092
```

发现p-value均小于5%，说明在5%的显著水平下可认为对数收益率存在显著的ARCH效应。

(b)

绘制对数收益率平方的PACF图：



选取ARCH(2) 模型:

```
Title:
 GARCH Modelling

Call:
 garchFit(formula = ~arma(0, 0) + garch(2, 0), data = raw, trace = F)

Mean and Variance Equation:
 data ~ arma(0, 0) + garch(2, 0)
<environment: 0x00000212d044e8f0>
 [data = raw]

Conditional Distribution:
```

```
 norm

Coefficient(s):
      mu     omega    alpha1    alpha2
0.010615  0.003228  0.078122  0.128041

Std. Errors:
 based on Hessian

Error Analysis:
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.010615    0.002205    4.814 1.48e-06 ***
omega   0.003228    0.000256   12.609  < 2e-16 ***
alpha1  0.078122    0.044993    1.736   0.0825 .
alpha2  0.128041    0.053228    2.406   0.0162 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Log Likelihood:
 1017.364    normalized:  1.347502

Description:
 Mon May 16 13:42:02 2022 by user: Yorick He


Standardised Residuals Tests:
                                Statistic p-Value
 Jarque-Bera Test   R    Chi^2  41.06987  1.207233e-09
 Shapiro-Wilk Test  R    W      0.9917295 0.0003164345
 Ljung-Box Test     R    Q(10)  19.81877  0.0310142
 Ljung-Box Test     R    Q(15)  29.37285  0.01439508
 Ljung-Box Test     R    Q(20)  32.61738  0.03714796
 Ljung-Box Test     R^2  Q(10)  9.053344  0.5270487
 Ljung-Box Test     R^2  Q(15)  16.91694  0.3238542
 Ljung-Box Test     R^2  Q(20)  24.24292  0.2319419
 LM Arch Test       R    TR^2   10.14719  0.6030498

Information Criterion Statistics:
      AIC       BIC       SIC      HQIC
-2.684407 -2.659895 -2.684463 -2.674965
```

(c)

划分数据集并预测：

```
> predict(model2, 5)
  meanForecast  meanError standardDeviation
1   0.01092165 0.07074067        0.07074067
2   0.01092165 0.06003729        0.06003729
3   0.01092165 0.06422518        0.06422518
4   0.01092165 0.06316346        0.06316346
5   0.01092165 0.06359692        0.06359692
```

(d)

建立模型：

```

*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model : sGARCH(2,0)
Mean Model  : ARFIMA(0,0,0)
Distribution    : norm

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.033477    0.026012  1.28696 0.198106
archm  -0.366184    0.420217 -0.87142 0.383526
omega   0.003196    0.000255 12.53235 0.000000
alpha1  0.077059    0.043603  1.76728 0.077182
alpha2  0.130298    0.052178  2.49717 0.012519

Robust Standard Errors:
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.033477    0.022023   1.5201  0.12849
archm  -0.366184    0.347874  -1.0526  0.29251
omega   0.003196    0.000291  10.9745  0.00000
alpha1  0.077059    0.052722   1.4616  0.14385
alpha2  0.130298    0.067558   1.9287  0.05377

LogLikelihood : 1013.199
```

因为 arch_mean 检验的p-value大于5%，所以不能拒绝原假设，风险溢价为0 。

(e)

EGARCH模型 ：

```
*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model : eGARCH(1,1)
Mean Model  : ARFIMA(0,0,0)
Distribution    : norm

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.009826    0.002150  4.57001 0.000005
omega  -0.840954    0.420366 -2.00053 0.045443
alpha1 -0.012217    0.028979 -0.42158 0.673331
beta1   0.847866    0.075872 11.17494 0.000000
gamma1  0.173974    0.055693  3.12381 0.001785

Robust Standard Errors:
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.009826    0.002035  4.82880 0.000001
omega  -0.840954    0.460686 -1.82544 0.067935
alpha1 -0.012217    0.049376 -0.24743 0.804573
beta1   0.847866    0.084248 10.06398 0.000000
gamma1  0.173974    0.065290  2.66464 0.007707

LogLikelihood : 1015.314
```

预测结果为：

```
*------------------------------------*
*       GARCH Model Forecast         *
*------------------------------------*
Model: eGARCH
Horizon: 5
Roll Steps: 0
Out of Sample: 0

0-roll forecast [T0=1972-01-21 08:00:00]:
      Series   Sigma
T+1 0.009826 0.05882
T+2 0.009826 0.05945
T+3 0.009826 0.05998
```

```
T+4 0.009826 0.06044
T+5 0.009826 0.06083
```

代码：

```
library(fBasics)
require(fGarch)
require(rugarch)
da = read.table("./data/m-3m4608.txt", header = T)[,2]
raw = log(1 + da)
plot(ts(raw))

Box.test(raw,lag = 6,type = 'Ljung')
Box.test(raw,lag = 12,type = 'Ljung')

Box.test(raw^2,lag = 6,type = 'Ljung')
Box.test(raw^2,lag = 12,type = 'Ljung')

pacf(raw^2)
model = garchFit(~ arma(0, 0) + garch(2, 0), data = raw, trace = F)
summary(model)

train = raw[1:750]
model2 = garchFit(~ arma(0, 0) + garch(2, 0), data = train, trace = F)
predict(model2, 5)

model3 = ugarchfit(ugarchspec(variance.model = list(model =
"sGARCH",garchOrder= c(2, 0)),
                              mean.model = list(armaOrder = c(0, 0),
archm = T)),train)
show(model3)

model4 = ugarchfit(ugarchspec(variance.model = list(model =
"eGARCH",garchOrder= c(1, 1)),
                              mean.model = list(armaOrder = c(0, 0))),
train)
show(model4)
ugarchforecast(model4, n.ahead = 5)
```