

北京航空航天大学
2019—2020 学年 第二学期期末

《R 语言编程及应用》
考 试 A 卷

任课教师：马 杰

班 级_____学 号_____

姓 名_____成 绩_____

考试日期:2020 年 6 月 29 日

班号 _____ 学号 _____ 姓名 _____ 成绩 _____

《R 语言编程及应用》期末考试卷

注意事项：

1、因疫情原因，本次考试为在线开卷考试，允许看书、看课件，也允许电脑打字答题——但文字概念性问题、不要直接复制/粘贴课件，请用自己的话语或举例、加以组织与编辑文字。

2、绝不允许相互抄袭，一旦发现有雷同卷、文字/用词/举例等雷同作弊情形，抄袭与提供抄袭答案的同学，都记为作弊 0 分。

试题：

一、名词解释说明……………(每个 8 分,共 24 分)

[提示：解释说明，即对名词略微详细的展开解释，不能只是简略的一句话；大约 3~6 句话，展开解释说明即可。]

1. 列表：
2. 因子：
3. 期望损失：

二、已有编码解读题 ……………(31 分)

要求：1) 请对以下每行语句，进行注释（没提供运行结果的代码行、不需要对运行结果进行判断与比较，只需要依上下文代码、说明每行代码本身的含义）；
2) 然后，概述/总结该段代码实现的主要功能。

library(quantmod)

```
getSymbols('^HSI', from='1989-12-01', to='2013-11-29')
```

```
dim(HSI)
```

```
names(HSI)
```

```
head(HSI) ; tail(HSI)
```

[提示：部分运行结果如下图]

	HSI.Open	HSI.High	HSI.Low	HSI.Close	HSI.Volume	HSI.Adjusted
2013-11-21	23661.42	23661.42	23508.79	23580.29	1620640500	23580.29
2013-11-22	23752.28	23764.41	23641.20	23696.28	1689620800	23696.28
2013-11-25	23829.97	23829.97	23673.53	23684.45	1513617400	23684.45
2013-11-26	23675.37	23751.86	23659.98	23681.28	1505522100	23681.28
2013-11-27	23653.95	23856.92	23649.80	23806.35	1588937200	23806.35
2013-11-28	23934.94	24014.81	23734.54	23789.09	1333799400	23789.09

```
chartSeries(HSI, theme='white')
```

```
ptd.HSI <- HSI$HSI.Adjusted
```

```
ptd.HSI <- na.omit(ptd.HSI)
```

```
rtd.HSI <- diff(log(ptd.HSI))*100
```

```
rtd.HSI <- rtd.HSI[-1,]
```

```
plot(rtd.HSI)
```

```
head(to.monthly(HSI)) ; tail(to.monthly(HSI))
```

[提示：部分运行结果如下图]

	HSI.Open	HSI.High	HSI.Low	HSI.Close	HSI.Volume	HSI.Adjusted
十二月 1989	2748	2962	2735	2836.6	0	2836.6
一月 1990	2836	2889	2732	2751.6	0	2751.6
二月 1990	2751	2982	2697	2952.0	0	2952.0
三月 1990	2951	3027	2862	2997.0	0	2997.0

```
ptm.HSI <- to.monthly(HSI)$HSI.Adjusted
```

```
rtn.HSI <- diff(log(ptm.HSI))*100
```

```
rtn.HSI <- rtn.HSI[-1,]
```

```
plot(rtm.HSI)

detach(package:quantmod)

class(rtm.HSI) ; dim(rtm.HSI) ; head(rtm.HSI) ; tail(rtm.HSI)

outsample <- gsub(' ', '', substr(index(rtm.HSI), 4, 8)) %in% '2013'

insample <- !outsample

(rtm.insample <- rtm.HSI[insample])

(rtm.outsample <- rtm.HSI[outsample])

Box.test(rtm.insample, lag=12, type='Ljung-Box')

Box.test(rtm.insample^2, lag=12, type='Ljung-Box')

FinTS::ArchTest(x=rtm.insample, lags=12)


epst <- rtm.insample - mean(rtm.insample)

par(mfrow=c(1,2))

acf(as.numeric(epst)^2, lag.max=20, main='平方序列')

pacf(as.numeric(epst)^2, lag.max=20, main='平方序列')


library(fGarch)
GARCH.model_1 <- garchFit(~garch(1,1), data=rtm.insample, trace=FALSE)

GARCH.model_2 <- garchFit(~garch(2,1), data=rtm.insample, trace=FALSE)
```

```
GARCH.model_3 <- garchFit(~garch(1,1), data=rtm.insample, cond.dist='std',  
trace=FALSE)
```

```
GARCH.model_4 <- garchFit(~garch(1,1), data=rtm.insample, cond.dist='sstd',  
trace=FALSE)
```

```
GARCH.model_5 <- garchFit(~garch(1,1), data=rtm.insample, cond.dist='ged',  
trace=FALSE)
```

```
GARCH.model_6 <- garchFit(~garch(1,1), data=rtm.insample, cond.dist='sged',  
trace=FALSE)
```

```
summary(GARCH.model_1)
```

```
summary(GARCH.model_2)
```

```
plot(GARCH.model_1)
```

```
slotNames(GARCH.model_1)
```

```
mode(GARCH.model_1)
```

```
class(GARCH.model_1)
```

```
typeof(GARCH.model_1)
```

```
(vol_1 <- fBasics::volatility(GARCH.model_1))
```

```
(GARCH.model_1@sigma.t)
```

```
(GARCH.model_1@fitted)
```

```
sres_1 <- residuals(GARCH.model_1, standardize=TRUE)
```

```
vol_1.ts <- ts(vol_1, frequency=12, start=c(1990, 1))
```

```
sres_1.ts <- ts(sres_1, frequency=12, start=c(1990, 1))
```

```
par(mfcol=c(2,1))
```

```
plot(vol_1.ts, xlab='年', ylab='波动率 sigma(t)')
```

```
plot(sres_1.ts, xlab='年', ylab='标准化残差')
```

```
par(mfrow=c(2,2))
```

```
acf(sres_1, lag=24)
```

```
pacf(sres_1, lag=24)
```

```
acf(sres_1^2, lag=24)
```

```
pacf(sres_1^2, lag=24)
```

```
FinTS::ArchTest(x=sres_1, lags=12)
```

```
par(mfrow=c(1,1))
```

```
qqnorm(sres_1)
```

```
qqline(sres_1)
```

```
pred.model_1 <- predict(GARCH.model_1, n.ahead = 11, trace = FALSE, mse = 'cond',  
plot=FALSE)
```

```
pred.model_2 <- predict(GARCH.model_2, n.ahead = 11, trace = FALSE, mse = 'cond',  
plot=FALSE) [提示：与上一行类同的以下几行代码，不需要注释]
```

```
pred.model_3 <- predict(GARCH.model_3, n.ahead = 11, trace = FALSE, mse = 'cond',  
plot=FALSE)
```

```
pred.model_4 <- predict(GARCH.model_4, n.ahead = 11, trace = FALSE, mse = 'cond',  
plot=FALSE)
```

```
pred.model_5 <- predict(GARCH.model_5, n.ahead = 11, trace = FALSE, mse = 'cond',  
plot=FALSE)
```

```
pred.model_6 <- predict(GARCH.model_6, n.ahead = 11, trace = FALSE, mse = 'cond',  
plot=FALSE)
```

```
names(pred.model_1)
```

```
predVol_1 <- pred.model_1$standardDeviation
```

```
predVol_2 <- pred.model_2$standardDeviation
```

[提示：与上一行类同的以下几行代码，不需要注释]

```
predVol_3 <- pred.model_3$standardDeviation
```

```
predVol_4 <- pred.model_4$standardDeviation
```

```
predVol_5 <- pred.model_5$standardDeviation
```

```
predVol_6 <- pred.model_6$standardDeviation
```

```
et <- abs(rtm.outsample - mean(rtm.outsample))
```

```
rtd.HSI.2013 <- rtd.HSI['2013']
```

```
head(rtd.HSI.2013) ; tail(rtd.HSI.2013)
```

[提示：部分运行结果如下图]

```

                HSI.Adjusted
2013-01-02  2.807538438
2013-01-03  0.370876316
2013-01-04 -0.288937559

```

```
(rv <- sqrt(aggregate(rtd.HSI.2013^2, by=substr(index(rtd.HSI.2013), 1, 7), sum)))
```

```
predVol
```

```
round(rbind(predVol_1,predVol_2,predVol_3,predVol_4,predVol_5,predVol_6,
```

```
as.numeric(et), as.numeric(rv)), digits=3)
```

```
colnames(predVol) <- 1:11
```

```
rownames(predVol) <- c('GARCH(1,1)-N模型','GARCH(1,2)-N模型','GARCH(1,1)-t模型',
'GARCH(1,1)-st模型', 'GARCH(1,1)-GED模型','GARCH(1,1)-SGED模型','残差绝对值',
'已实现波动率')
```

```
print(predVol)
```

[提示：运行结果如下图]

	1	2	3	4	5	6	7	8	9	10	11
GARCH(1,1)-N模型	5.003	5.260	5.493	5.708	5.906	6.090	6.260	6.419	6.567	6.706	6.836
GARCH(1,2)-N模型	4.706	4.709	5.114	5.392	5.659	5.897	6.115	6.314	6.497	6.666	6.822
GARCH(1,1)-t模型	5.309	5.500	5.677	5.841	5.995	6.139	6.274	6.401	6.520	6.633	6.739
GARCH(1,1)-st模型	5.348	5.528	5.695	5.850	5.995	6.130	6.257	6.376	6.488	6.593	6.692
GARCH(1,1)-GED模型	5.136	5.347	5.542	5.723	5.891	6.048	6.194	6.331	6.460	6.581	6.695
GARCH(1,1)-SGED模型	5.197	5.395	5.578	5.747	5.905	6.051	6.187	6.314	6.434	6.546	6.651
残差绝对值	4.143	3.474	3.620	1.503	1.968	7.799	4.624	1.138	4.623	1.065	2.041
已实现波动率	3.508	4.114	3.929	4.778	4.374	6.013	5.397	4.634	4.070	3.745	4.378

三、代码编程应用题..... (45 分)

Specialty 玩具公司的订单数量方案决策

Specialty 公司销售大量新款有创意的玩具。由于每年 12 月许多家庭要为孩子准备节日礼物，故 Specialty 公司决定从 10 月份起将新款玩具投放市场。为保证玩具能在商场如期到货， Specialty 公司需要在每年 6 月、即向制造商下订单。玩具市场需求瞬息万变：若某新款玩具推出时很受欢迎，则可能有很高的销量；但若该款玩具并不讨喜产生滞销，大量积货将不得不降价促销。因此，事前下订单多少，是 Specialty 公司运营面临的主要问题。

Specialty 公司计划本年度推出一款天气熊的新产品（产品性能描述，略），管理层团队建议准备的订单量分别为 15000、18000、24000、28000 只，不同方案代表着管理团队对新产品的市场潜力存在相当大的分歧，不同订单方案存货出清的概率不同、潜在利润也不同。进货每个天气熊的成本为 16 美元， Specialty 公司将以 24 美元价格出售；若节日后仍有存货，则只能以 5 美元的价格清仓去库存。根据以往同类产品的历史销售数据， Specialty 公司产品销量的资深专家们估计，天气熊需求量的期望水平为 20000 只、且在 10000~30000 只之间的概率为 95%。

请根据以上题意，假设天气熊的需求量分布近似服从正态分布的情况下，在线编程计算解决以下问题：

1) 若依管理层团队建议准备的订单量，4 种方案下、存货出清概率各自是多少？
(10 分)

2) 假设市场需求恰好为期望水平 20000 只，请计算 4 种订单方案下各自的潜在利润是多少？
(15 分)

3) 依管理层团队建议准备的几种订单量未必是科学的，请你依题意中的客观条件，确定 Specialty 玩具公司的最佳订单数量方案，以及你的方案下的公司潜在利润。
(20 分)

[提示：解题之前，先简单剖析一下解决问题的思路、然后再编代码实现；另，由于可在线使用电脑 R 软件，以上答题中可能用到的函数 `uniroot()`、`optimize()`、或 `optim()`，可自行在线查询其语法、举例与使用方法。]