

学校代码：10200

学号：2013012422



# 东北师范大学

时间序列分析期末作业

## 北京银行股票数据分析报告

学生姓名：隆征帆

所在学院：数学与统计学院

所学专业：统计学

中国·长春

2017年5月

## 摘要

本报告首先对北京银行股票收盘价序列作了平稳性分析，得出北京银行股票收盘价序列非平稳的结论。然后对北京银行股票收盘价差分序列作了平稳性分析，得出北京银行股票收盘价差分序列平稳的结论。接着对北京银行股票收盘价建立了三个不同的 *ARIMA* 模型，对这三个模型作了充分性检验，并利用拉开档次法对三个模型进行了综合评价。本报告最终选择使用 *ARIMA*(4,1,2) 模型对北京银行股票收盘价建模，并利用该模型作了超前 2 步预测。本报告预测 5 月 8 日北京银行股票收盘价将位于  $[7.91, 9.47]$  内，即 5 月 8 日北京银行股票收盘价为  $8.69 \pm 0.78$ 。

**关键字：** 时序图 纯随机性检验 自相关函数图 偏自相关函数图 扩展单位根检验 *ARIMA* 模型 模型选择 回测检验 预测均方误差 预测平均绝对误差 AIC BIC 拉开档次法

## Abstract

In this report, we first made a stationarity analysis of stock closing price series of the Bank of Beijing, and we concluded that the Bank of Beijing stock closing price series is not a stationary sequence. And then we made a stationarity analysis of the Beijing Bank's stock closing price difference sequence, and draw the conclusion that the Bank of Beijing stock price difference sequence is stable. Then, three different ARIMA models were established for the bank closing price of the Bank of Beijing. The three models were tested adequately and the three models were evaluated synthetically by using the grade method. This report eventually chose to use the ARIMA (1,1,0) model to model the Bank of Beijing stock closing price, and use the model to make the first two steps to predict. The report predicts that the closing price of the Bank of Beijing on May 8 will be in the range of [7.91,9.47], that is, the closing price of the Bank of Beijing on 8 May is  $8.69 \pm 0.78$ .

**Key words:** Temporal graph   pure randomness test   autocorrelation function graph   partial autocorrelation function graph   extended unit root test   ARIMA model   model selection   back test   test prediction mean square error   prediction mean absolute error   AIC   BIC   pull-off method

## 1 数据概况

本报告使用的数据集为：2007 年 9 月 19 日至 2017 年 5 月 4 日北京银行股票收盘价。该数据集样本量为 2471，数据来自于 *Yahoo*，由 *R* 语言 *quantmod* 包中的 *getSymbols()* 函数获得。获取该数据的 *R* 程序见附录。

## 2 股票收盘价分析

图 2-1，图 2-2，图 2-3，图 2-4 分别是股票收盘价的时序图、纯随机性检验  $p$  值图、自相关函数 (ACF) 图以及偏自相关函数 (PACF) 图。获取这些图像的 *R* 程序见附录。



图 2-1

北京银行股票收盘价纯随机性检验p值图

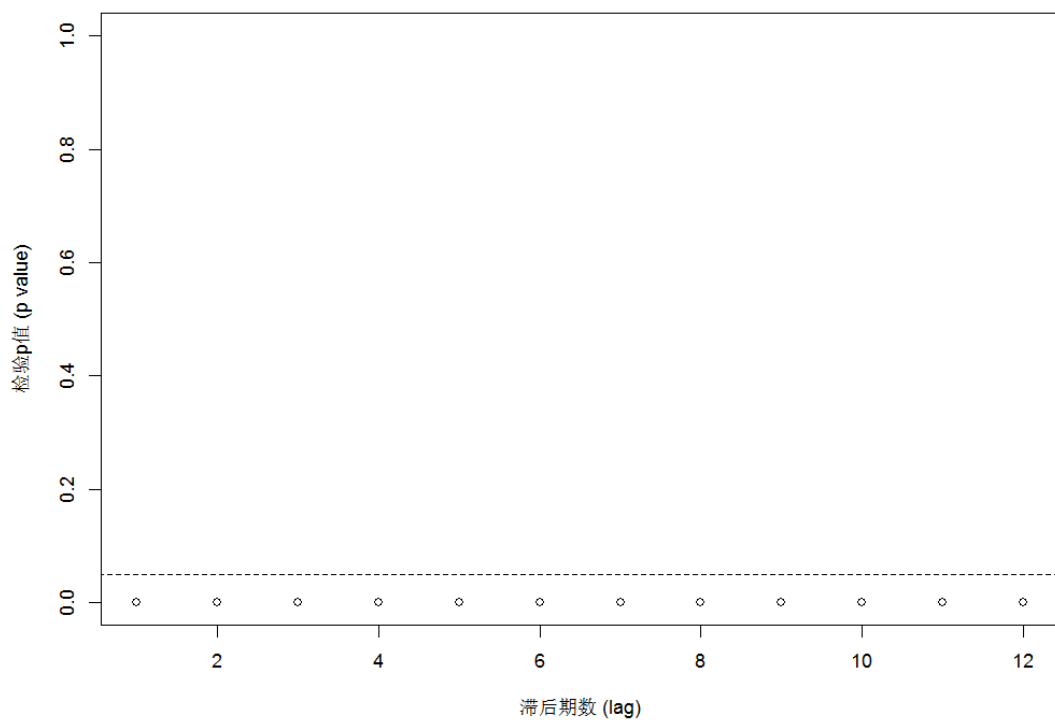


图 2-2

北京银行股票收盘价自相关图

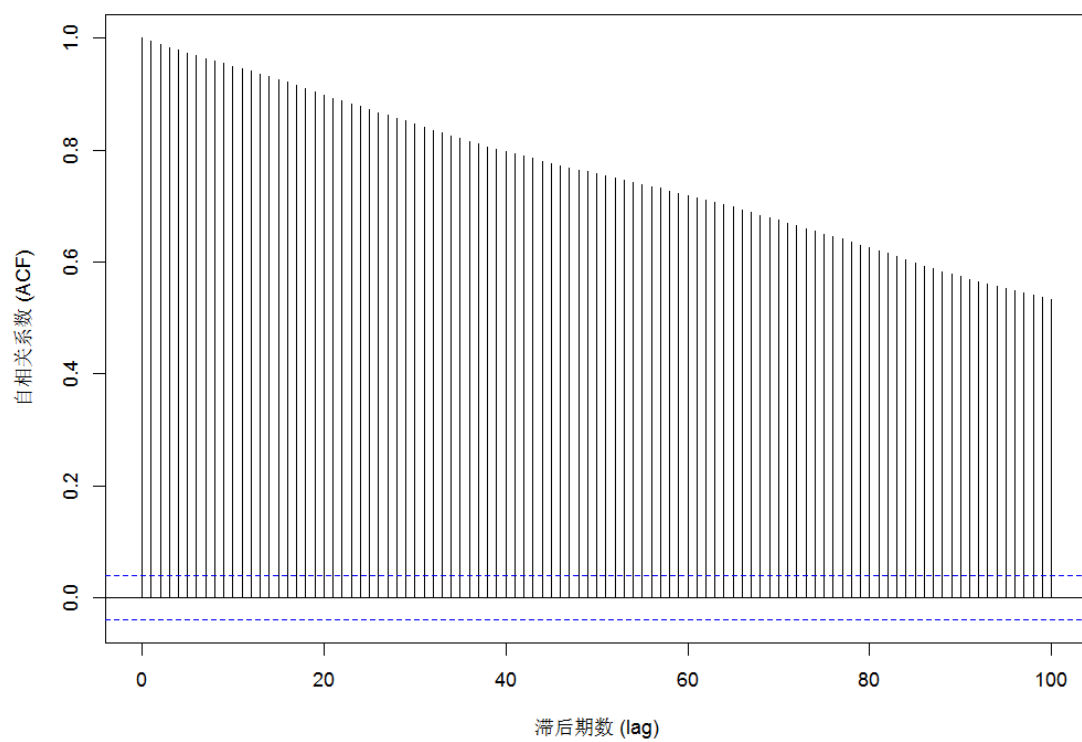


图 2-3

北京银行股票收盘价偏自相关图

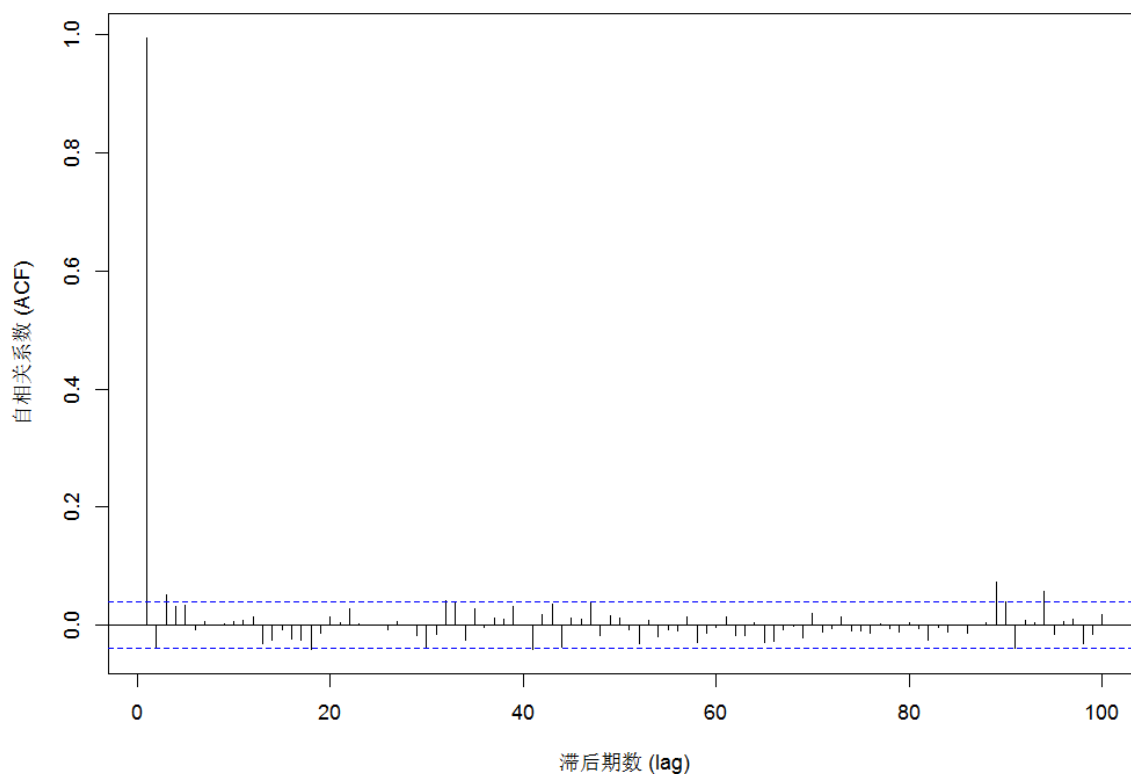


图 2-4

由图 2-2 可以知道：北京银行股票收盘价自相关函数不为 0，即北京银行股票收盘价不是一个纯随机序列，具有继续分析的价值。又由图 2-1、图 2-3、图 2-4 可以得到直观性结论：北京银行股票收盘价序列非平稳。

为进一步明确序列的非平稳性，观察图 2-4，可以得到偏自相关函数 3 阶截尾的结论，因此对  $AR(3)$  模型作扩展单位根检验，检验结果为：扩展单位根检验  $p$  值= 0.0858，不能拒绝单位根假设，认为序列非平稳。获取该检验结果的  $R$  程序见附录。

综上，北京银行股票收盘价序列非平稳，考虑使用差分方法使该非平稳序列平稳化。

### 3 股票收盘价差分序列分析

图 3-1，图 3-2，图 3-3，图 3-4 分别是股票收盘价差分序列的时序图、纯随机性检验  $p$  值图、自相关函数 (ACF) 图以及偏自相关函数 (PACF) 图。获取这些图像的  $R$  程序见附录。

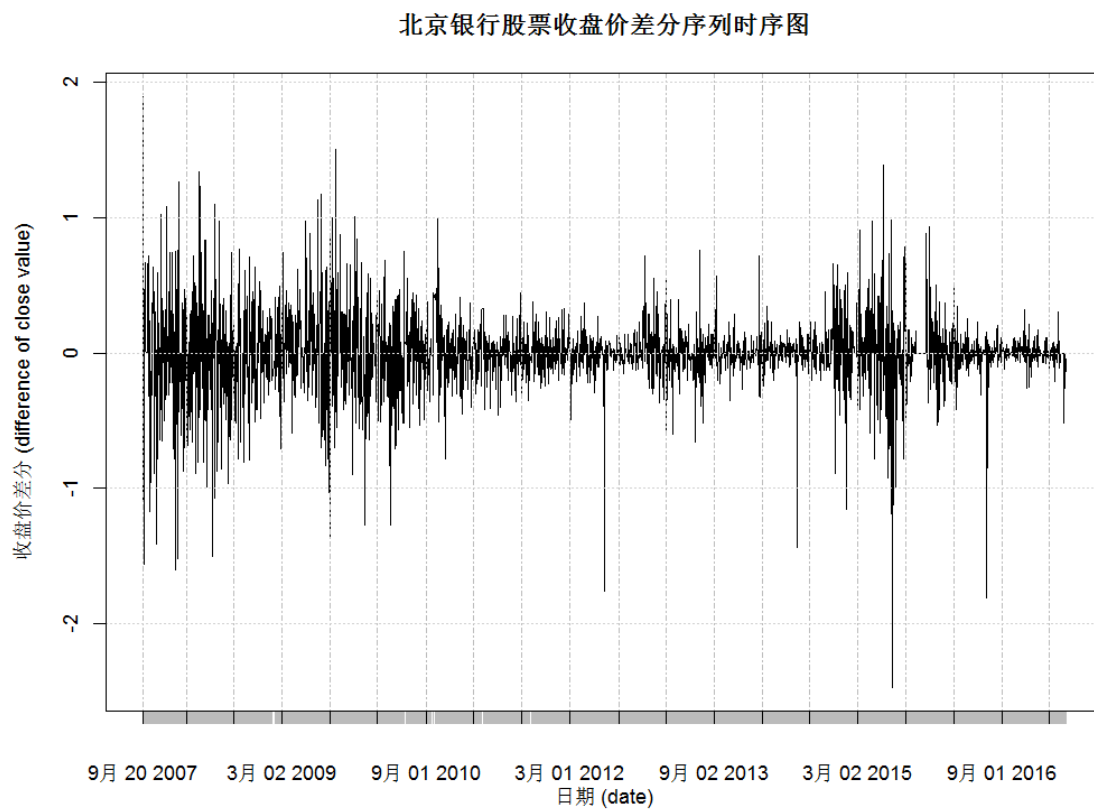


图 3-1

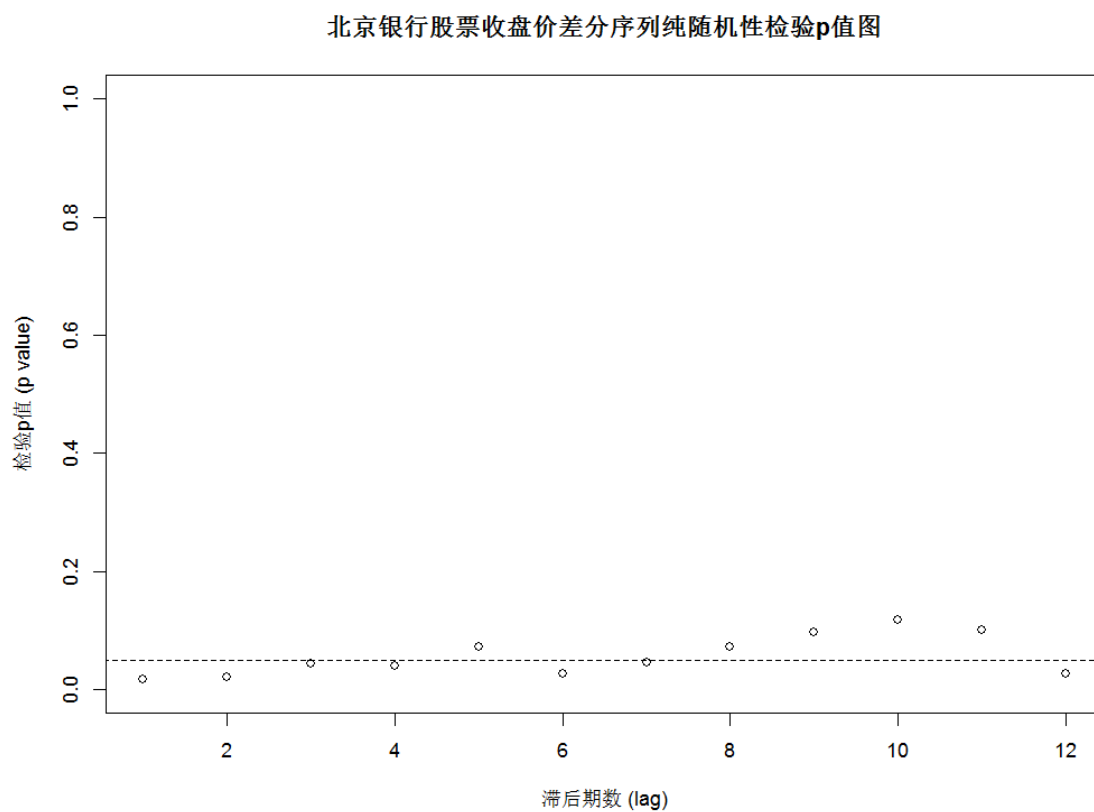


图 3-2

北京银行股票收盘价差分序列自相关图

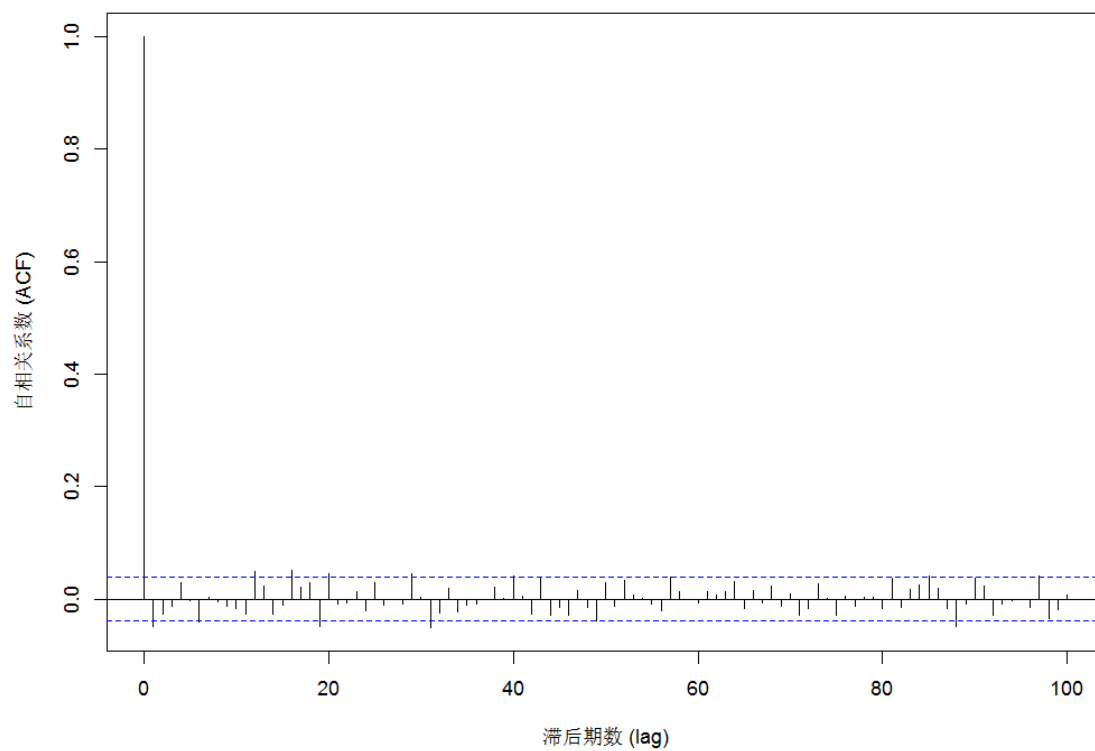


图 3-3

北京银行股票收盘价差分序列偏自相关图

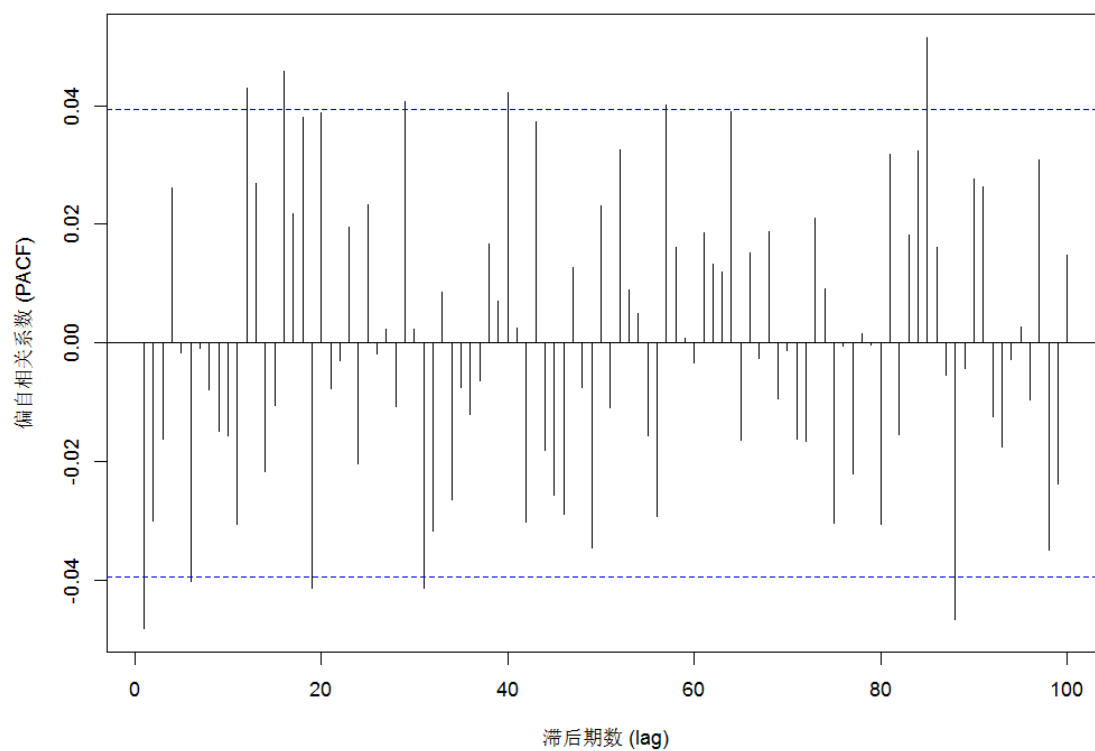


图 3-4



由图 3-2 可以知道：北京银行股票收盘价差分序列自相关函数不为 0，即北京银行股票收盘价差分序列不是一个纯随机序列，具有继续分析的价值。又由图 3-1、图 3-3、图 3-4 可以得到直观性结论：北京银行股票收盘价差分序列平稳。

为进一步明确序列的平稳性，观察图 3-4，可以得到“偏自相关函数 1 阶截尾”的结论，因此对  $AR(1)$  模型作扩展单位根检验，检验结果为：扩展单位根检验  $p$  值 = 0.01，拒绝单位根假设，认为序列平稳。获取该检验结果的  $R$  程序见附录。

综上，北京银行股票收盘价差分序列平稳，考虑使用  $ARMA(p, q)$  模型对北京银行股票收盘价差分序列进行分析，这等价于使用  $ARIMA(p, d, q)$  模型对北京银行股票收盘价序列进行分析。观察图 3-3、图 3-4，得到“北京银行股票收盘价差分序列  $ACF$  拖尾， $PACF$  1 步截尾”的结论，因此判断  $ARIMA(p, d, q)$  模型阶数为  $p = 1, d = 1, q = 0$ 。

## 4 股票收盘价序列建模

通过观察图 3-3、图 3-4，我们判断  $ARIMA(p, d, q)$  模型阶数为  $p = 1, d = 1, q = 0$ ，而通过  $R$  语言 *forecast* 程序包中的 *auto.arima()* 函数得到  $ARIMA(p, d, q)$  模型阶数为  $p = 4, d = 1, q = 2$ ，另外鉴于指数平滑模型（即  $ARIMA(0, 1, 1)$  模型）是时间序列建模时的常用模型，因此我们决定对股票收盘价序列建立三个模型，它们分别是： $ARIMA(1, 1, 0)$ 、 $ARIMA(4, 1, 2)$ 、 $ARIMA(0, 1, 1)$ 。

图 4-1，图 4-2，图 4-3 分别是  $ARIMA(1, 1, 0)$  模型、 $ARIMA(4, 1, 2)$  模型、 $ARIMA(0, 1, 1)$  模型的充分性检验结果。

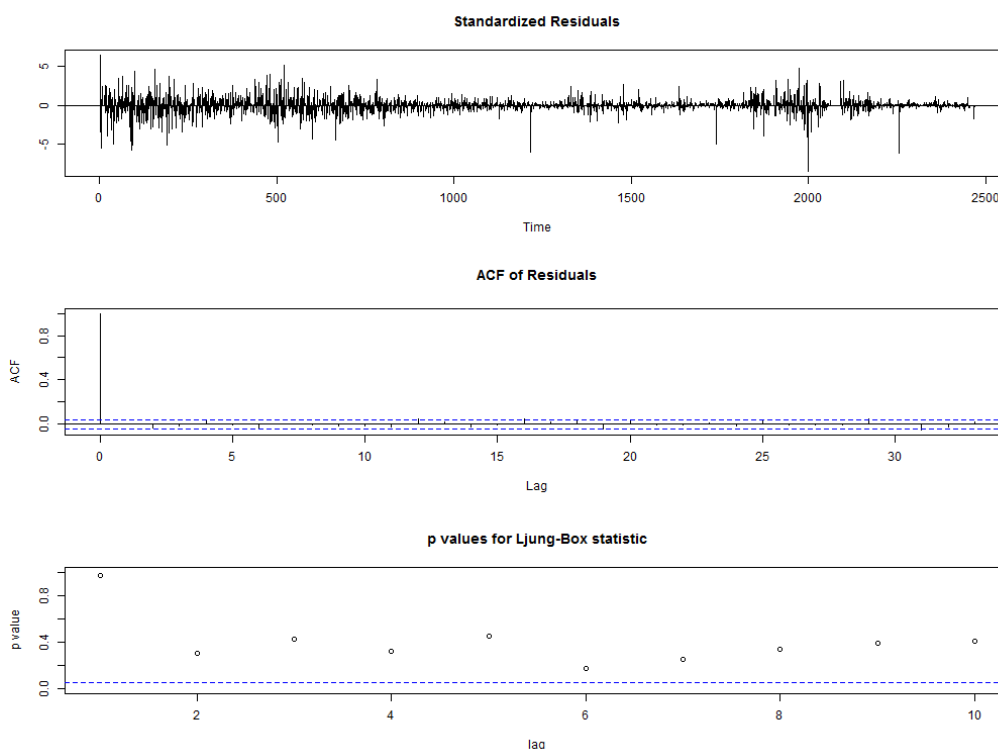


图 4-1  $ARIMA(1, 1, 0)$  模型充分性检验结果

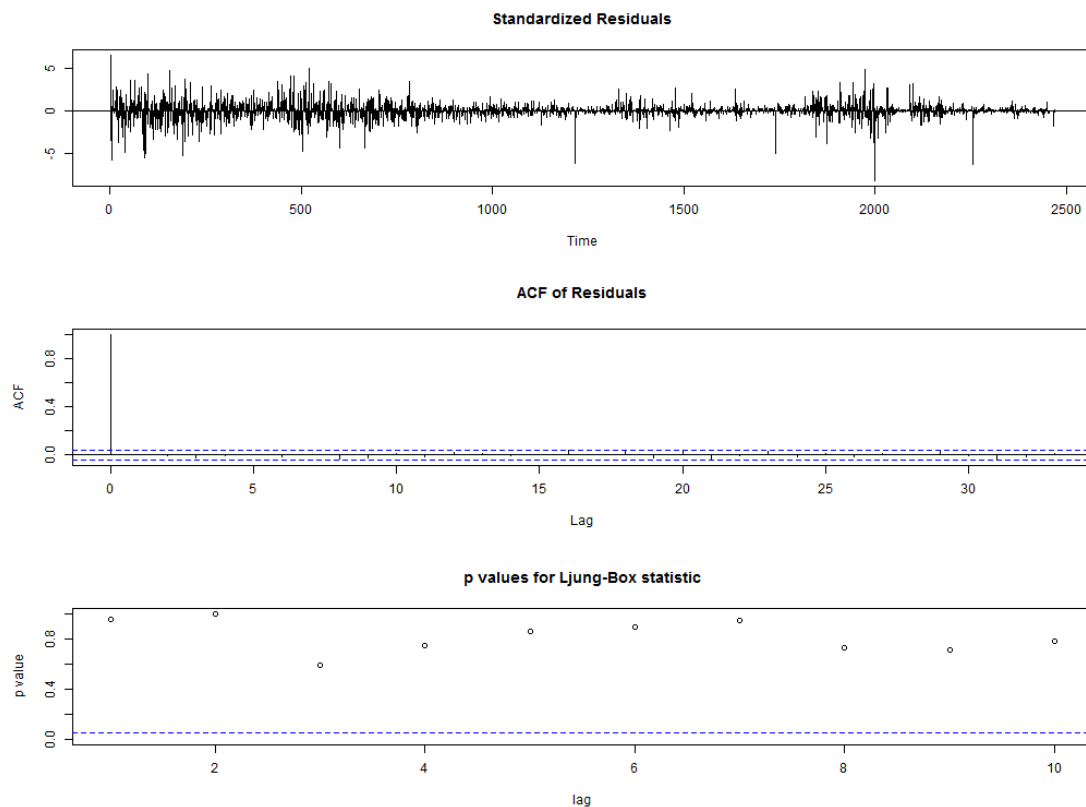


图 4-2 ARIMA(4,1,2) 模型充分性检验结果

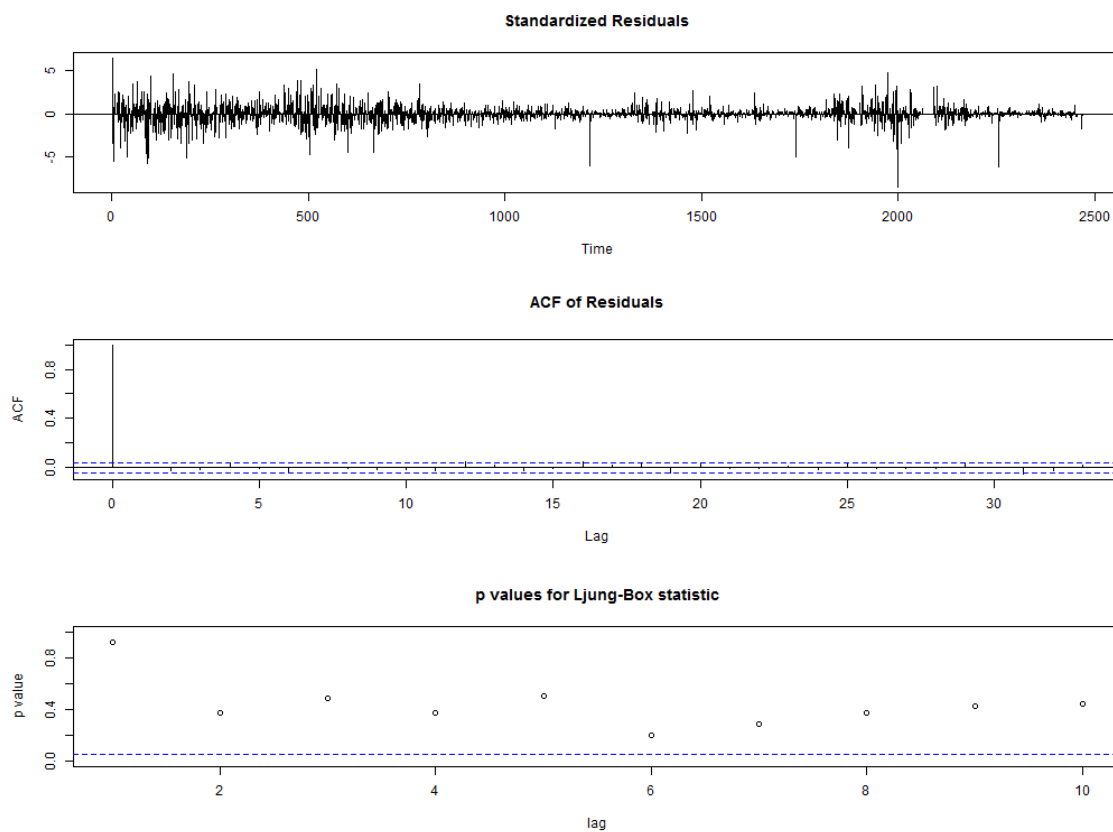


图 4-3 ARIMA(0,1,1) 模型充分性检验结果

由图 4-1, 图 4-2, 图 4-3 可以得到:  $ARIMA(1,1,0)$  模型、 $ARIMA(4,1,2)$  模型、 $ARIMA(0,1,1)$  模型的残差序列为纯随机序列, 因而三个模型均是充分的, 即它们很好的提取了序列的相关信息。

## 5 股票收盘价模型选择

我们选取了由“回测检验”得到的模型的预测均方误差 ( $MSFE$ )、预测平均绝对误差 ( $MAFE$ ) 以及模型的  $AIC$  值、模型的  $BIC$  值共四项指标, 由“拉开档次法”对上面建立的三个模型进行了综合评价 (有关拉开档次法的原理见文献[4])。表 5-1 为三个模型的预测效果指标, 表 5-2 为由  $R$  及  $Matlab$  得到的模型综合评价结果, 评价结果按综合评价值由小到大排序 (因为各项指标均为极小型指标), 另外需要说明的是: 为了应用来开档次法, 表 5-2 中各项指标在表 5-1 的基础上作了标准化处理。

表 5-1 模型预测效果指标

	预测均方误差	预测平均绝对误差	AIC	BIC
$ARIMA(1,1,0)$	0.01345	0.07128	885.82	889.63
$ARIMA(4,1,2)$	0.01343	0.07228	878.14	911.02
$ARIMA(0,1,1)$	0.01345	0.07133	885.47	889.28

表 5-2 模型综合评价结果

	预测均方误差	预测平均绝对误差	AIC	BIC	综合评价值
$ARIMA(4,1,2)$	-1.15470	1.15356	-1.15376	1.15459	-1.15423
$ARIMA(0,1,1)$	0.57735	-0.53241	0.53653	-0.59135	0.55694
$ARIMA(1,1,0)$	0.57735	-0.62115	0.61723	-0.56324	0.59729

由表 5-2, 我们选择模型  $ARIMA(4,1,2)$  作为我们最终的模型, 该模型表达式为:

$$x_t = 1.1712x_{t-1} - 1.1232x_{t-2} + 0.9233x_{t-3} + 0.0266x_{t-4} + 0.0021x_{t-5} \\ + a_t + 0.2224a_{t-1} - 0.9407a_{t-2}, a_t \sim N(0, 0.0830)$$

下面我们将使用该模型作超前 2 步预测, 得到 5 月 8 日北京银行股票收盘价的预测值。

## 6 股票收盘价预测

通过  $R$  语言  $forecast$  程序包中的  $forecast()$  函数得到  $ARIMA(4,1,2)$  模型超前 2 步预测结果为 8.69, 其 95% 置信区间为 [7.91,9.47]。因此预测 5 月 8 日北京银行股票收盘价将位于 [7.91,9.47] 内, 即 5 月 8 日北京银行股票收盘价为  $8.69 \pm 0.78$ 。

## 7 参考文献

- [1] Box.G.E.P, Jenkins G.M., Time Series Analysis,Forecasting and Control, Holden-Day, 1976
- [2] Jonathan D.Cryer, Kung-Sik Chan, Time Series Analysis with Applications in R, Second Edition, Springer Science+Business Media, 2008
- [3] Norman Matloff, The Art of R Programing:A Tour of Statistical Software Design, William Pollock,2011
- [4] 郭亚军, 综合评价理论、方法及应用[M], 北京: 科学出版社, 2007

## 8 附录：R 程序代码及 Matlab 程序代码

在保证已于 D 盘创建名为“北京银行”的文件夹，并保证安装了 quantmod、xts、tseries 三个程序包，以及网络通畅的前提下，运行以下 R 程序代码，将在目录“D:\北京银行\”下生成如图 8-1 所示的 18 个文件，本报告中使用的所有图片及检验结果均来自于这些文件。



图 8-1

本报告使用的 R 程序代码以及 matlab 程序代码已经上传至百度云盘，下载链接为：

- ①北京银行股票分析 R 语言程序：<http://pan.baidu.com/s/1kUAlTkj>
- ②非线性规划求解的 matlab 程序 1（evaluation.m）：<https://pan.baidu.com/s/1hs5O0wG>
- ③非线性规划求解的 matlab 程序 2（fun.m）：<https://pan.baidu.com/s/1skGLmhB>
- ④非线性规划求解的 matlab 程序 3（mycon.m）：<http://pan.baidu.com/s/1o8dlWd8>

以下是程序代码的详情

### 8.1 R 程序代码

```
rm(list=ls()) #清空内存
de.outloc <- "D:\\北京银行\\" #默认输出路径
stock.name <- "北京银行" #股票名称
```

```
##
##第一部分：股票数据的获取
##
```

**#1.1 创建股票数据的下载 (load) 和输出 (output) 函数 stock.lo****#stock.name 为股票名称, stock.symbol 为股票代码****#start.time 为起始时间, end.time 为截止时间, outloc 为输出路径**

```
stock.lo <- function(stock.name,stock.symbol,start.time,end.time,outloc){
```

```
  #1.1.1 加载 “quantmod” 程序包 (由 Jeffry 开发)
```

```
  #利用该包, 可从雅虎财经 (yahoo)、谷歌财经 (google)、圣路易斯联邦储备银行的联邦储备经济数据库 (PRED) 下载金融数据
```

```
  library(quantmod)
```

```
  #1.1.2 由 getSymbols 函数下载股票数据
```

```
  setSymbolLookup(OBJ = list(name=stock.symbol,src="yahoo",from=start.time,to=end.time))
  getSymbols("OBJ")
```

```
  #1.1.3 改股票数据英文列名为中文, 并将数据输出为一个 csv 文件, 文件名为股票名称
```

```
  da <- as.matrix(OBJ) #将 xts/zoo 对象转换为 matrix 对象
  colnames(da) <- c("开盘价","最高价","最低价","收盘价","成交量","调整后收盘价")
  write.csv(da,paste(outloc,stock.name,".csv",sep=""))
```

```
}
```

**#1.2. 下载并输出北京银行从 2007.09.19 到 2017.05.04 的股票数据, 文件保存在默认输出路径: D:\北京银行\**

```
stock.lo("北京银行","601169.SS","2007-09-19","2017-05-04",de.outloc)
```

```
##
##第二部分：股票数据的读取及分析前的准备工作
##
```

**#2.1 读入股票数据**

```
da <- read.csv(paste(de.outloc,stock.name,".csv",sep=""),header=T)
```

**#2.2 将 data.frame 对象转换成 xts/zoo 对象并提取第 5 列 (收盘价) 数据作为待分析数据**

```
date <- as.character(da[,1]) #将股票数据第一列 (时间) 转变为字符串类型
library(xts)
da <- xts(da[,5],order.by=as.Date(date,format="%Y/%m/%d"))
```

**#2.3 创建图形输出函数，i 为输出文件标号，name 为图形名称的核心部分**

```
PNG <- function(i,name){
  png(paste(de.outloc,i,"",stock.name,"",股票收盘价
",name,".png",sep=""),width=1024,height=768,pointsize=18)
}
```

**#2.4 创建时序图绘制函数**

```
SERIES <- function(xn,yn,is.d=FALSE){
  if(is.d==TRUE){
    plot(dda,xlab=xn,ylab=yn,main=paste(stock.name,"股票收盘价差分序列时序图",sep=""))
  }else{
    plot(da,xlab=xn,ylab=yn,main=paste(stock.name,"股票收盘价时序图",sep=""))
  }
}
```

**#2.5 创建纯随机性检验 p 值图绘制函数**

```
PV <- function(x,y,xn,yn,is.d=FALSE){
  if(is.d==TRUE){
    plot(x,y,xlab=xn,ylab=yn,ylim=c(0,1),main=paste(stock.name,"股票收盘价差分序列纯随机
性检验 p 值图",sep=""))
  }else{
    plot(x,y,xlab=xn,ylab=yn,ylim=c(0,1),main=paste(stock.name,"股票收盘价纯随机性检验 p 值
图",sep=""))
  }
}
```

**#2.6 创建 ACF 图绘制函数**

```
ACF <- function(xn,yn,is.d=FALSE,la=100){
  if(is.d==TRUE){
    acf(dda,lag=la,xlab=xn,ylab=yn,main=paste(stock.name,"股票收盘价差分序列自相关图
",sep=""))
  }else{
    acf(da,lag=la,xlab=xn,ylab=yn,main=paste(stock.name,"股票收盘价自相关图",sep=""))
  }
}
```

**#2.7 创建 PACF 图绘制函数**

```
PACF <- function(xn,yn,is.d=FALSE,la=100){
  if(is.d==TRUE){
    pacf(dda,lag=la,xlab=xn,ylab=yn,main=paste(stock.name,"股票收盘价差分序列偏自相关图
",sep=""))
  }else{
    pacf(da,lag=la,xlab=xn,ylab=yn,main=paste(stock.name,"股票收盘价偏自相关图",sep=""))
  }
}
```

```
    pacf(da,lag=la,xlab=xn,ylab=yn,main=paste(stock.name,"股票收盘价偏自相关图",sep=""))
  }
}
```

```
##  
##第三部分：股票收盘价序列分析  
##
```

### #3.1 股票收盘价时序图

```
PNG(1,"时序图")  
SERIES("日期 (date)", "收盘价 (close value)")  
dev.off()
```

### #3.2 股票收盘价纯随机性检验 p 值图

```
Bote.p <- rep(0,12)  
lag <- 1:12  
for(i in 1:12){  
  Bote.p[i] <- Box.test(da,lag=i,type="Ljung")$p.value  
}  
x <- seq(0,13,0.01)  
y <- rep(0.05,length(x))  
PNG(2,"纯随机性检验 p 值图")  
PV(lag, Bote.p, "滞后期数 (lag)", "检验 p 值 (p value)")  
lines(x,y,lty=2)  
dev.off()
```

### #3.3 股票收盘价自相关图

```
PNG(3,"自相关图")  
ACF("滞后期数 (lag)", "自相关系数 (ACF)")  
dev.off()
```

### #3.4 股票收盘价偏自相关图

```
PNG(4,"偏自相关图")  
PACF("滞后期数 (lag)", "自相关系数 (ACF)")  
dev.off()
```

##

##第四部分：平稳性统计检验，由第一部分偏自相关图确定增广单位根检验滞后阶数

##

#股票收盘价单位根检验(基于第三部分的偏自相关图，选择 AR(3)模型作扩展单位根检验)

```
library(tseries)
adt.p <- adf.test(da,"stationary",3)$p.value
if(adt.p >= 0.05){
  ters <- paste("扩展单位根检验 p 值 = ",round(adt.p,4),"，不能拒绝单位根假设，认为序列非平
  稳")
}else{
  ters <- paste("扩展单位根检验 p 值 = ",round(adt.p,4),"，拒绝单位根假设，认为序列平稳")
}
write.table(ters,paste(de.outloc,"5. 股票收盘价单位根检验结果.txt",sep=""),sep="")
```

##

##第五部分：股票收盘价差分序列时序图、纯随机性检验 p 值图、自相关图、偏自相关图

##

## #5.1 一阶差分

```
dda <- diff(da)[-1]
```

## #5.2 一阶差分时序图

```
PNG(6,"差分序列时序图")
SERIES("日期 (date)", "收盘价差分 (difference of close value)",is.d=T)
dev.off()
```

## #5.3 一阶差分纯随机性检验 p 值图

```
for(i in 1:12){
  Bote.p[i] <- Box.test(dda,lag=i,type="Ljung")$p.value
}
PNG(7,"差分序列纯随机性检验 p 值图")
PV(lag,Bote.p, "滞后期数 (lag)", "检验 p 值 (p value)",is.d=T)
lines(x,y,lty=2)
dev.off()
```

## #5.4 一阶差分自相关图

```
PNG(8,"差分序列自相关图")
```



```
ACF("滞后期数 (lag)","自相关系数 (ACF)",is.d=T)
dev.off()
```

### #5.5 一阶差分偏自相关图

```
PNG(9,"差分序列偏自相关图")
PACF("滞后期数 (lag)", "偏自相关系数 (PACF)",is.d=T)
dev.off()
```

```
##
##第六部分：差分序列平稳性统计检验，由第五部分偏自相关图确定增广单位根检验滞后阶数
##
```

### #一阶差分单位根检验

```
library(tseries)
dadt.p <- adf.test(dda,"stationary",1)$p.value
if(dadt.p >= 0.05){
  dters <- paste("扩展单位根检验 p 值 = ",round(dadt.p,4),"，不能拒绝单位根假设，认为序列非
平稳")
}else{
  dters <- paste("扩展单位根检验 p 值 = ",round(dadt.p,4),"，拒绝单位根假设，认为序列平稳")
}
write.table(dters,paste(de.outloc,"10. 股票收盘价差分序列单位根检验结果.txt",sep=""))
```

```
##
##第七部分：ARIMA 自动定阶及待选模型的确定
##
```

### #7.1 观察 ACF 图、PACF 图，定 ARIMA 模型阶数为 1,1,0

### #7.2 自动识别 ARIMA 模型阶数为 4,1,2

```
library(forecast)
pdq <- auto.arima(da)$arma
pdq <- c(pdq[1],1,pdq[2])
names(pdq) <- c("p","d","q")
write.table(pdq,paste(de.outloc,"11. ARIMA 自动定阶函数定阶结果.txt",sep=""))
```

### #7.3 另外增加一个指数平滑模型，其等价于阶数为 0,1,1 的 ARIMA 模型

```
##  
##第八部分：建模与模型充分性检验  
##
```

### #8.1 建模

```
ARIMA1 <- arima(da,order=c(1,1,0))  
ARIMA2 <- arima(da,order=c(4,1,2))  
ARIMA3 <- arima(da,order=c(0,1,1))
```

### #8.2 模型充分性检验

```
PNG(12.1,"模型 1 充分性检验结果")  
tsdiag(ARIMA1)  
dev.off()  
PNG(12.2,"模型 2 充分性检验结果")  
tsdiag(ARIMA2)  
dev.off()  
PNG(12.3,"模型 3 充分性检验结果")  
tsdiag(ARIMA3)  
dev.off()
```

```
##  
##第九部分：模型选择  
##
```

### #9.1 回测检验函数 (da 为原始数据, cr.mo 为建模函数, tesi 为初始测试集样本量)

```
back.test <- function(da,cr.mo,tesi=100){  
  n <- nrow(da)  
  trda <- da[1:(n-tesi),] #初始训练数据集  
  teda <- da[(n-tesi+1):n,] #初始测试数据集  
  sfe <- rep(0,tesi) #平方误差存储向量  
  afe <- rep(0,tesi) #绝对误差存储向量  
  bias <- rep(0,tesi) #偏差存储向量  
  library(forecast) #加载 forecast 包  
  for(i in 1:(tesi-2)){  
    model <- cr.mo(trda) #建模  
    prs <- forecast(model,h=2) #预测  
    sfe[i] <- (prs$mean[2]-teda[2])^2 #平方误差  
    afe[i] <- abs(prs$mean[2]-teda[2]) #绝对误差  
    trda <- da[1:(n-tesi+i),] #更新训练数据集  
    teda <- teda[-1,] #更新测试数据集  
  }  
}
```

```

msfe <- mean(sfe) #均方误差
mafe <- mean(afe) #平均绝对误差
ml <- cr.mo(da) #原始数据建模
par.nu <- length(ml$coe) #未知参数个数
#模型 AIC 值(已在建模列表 ml 中，直接输出)
AIC <- ml$aic
#模型 BIC 值(未在建模列表 ml 中，计算公式为：-2*对数似然+log(样本量)*未知参数个数)
BIC <- -2*ml$loglik+log(n)*par.nu
#预测效果指标输出
c(msfe,mafe,AIC,BIC)
}

```

## #9.2 模型个数

```
k <- 3
```

## #9.3 各模型预测效果指标存储矩阵 RS

```
RS <- matrix(0,k,4,dimnames=list(rep("",k),c("预测均方误差","预测平均绝对误差","AIC","BIC")))
```

## #9.4 模型 1 名称及建模函数

```
mdn1 <- "ARIMA(1,1,0)"
md1 <- function(da){
  arima(da,order=c(1,1,0))
}

```

## #9.5 模型 2 名称及建模函数

```
mdn2 <- "ARIMA(4,1,2)"
md2 <- function(da){
  arima(da,order=c(4,1,2))
}

```

## #9.6 模型 3 名称及建模函数

```
mdn3 <- "ARIMA(0,1,1)"
md3 <- function(da){
  arima(da,order=c(0,1,1))
}

```

## #9.7 由回测检验函数计算各模型预测效果

```

for(i in 1:3){
  #计算各预测效果指标
  RS[i,] <- back.test(da,eval(parse(text = paste("md",i,sep=""))))
  #将模型名称作为 RS 矩阵行名
  rownames(RS)[i] <- eval(parse(text = paste("mdn",i,sep="")))
}

```

**#9.8 输出 RS 矩阵到外部文件：13. 模型预测效果指标.csv**

```
write.csv(RS,paste(de.outloc,"13. 模型预测效果指标.csv",sep=""))
```

**#9.9 利用拉开档次法对模型进行综合评价并输出结果到外部文件：14. 模型综合评价结果.csv**

```
sel <- read.csv(paste(de.outloc,"13. 模型预测效果指标.csv",sep=""),header=T)
```

```
rownames(sel) <- sel[,1]
```

```
sel <- sel[,-1]
```

```
sel <- scale(sel)
```

```
H <- t(sel) %*% sel
```

```
W <- c(0.7071,0.0000,0.7071,0.0000) #由 matlab 得到的权重向量
```

```
W <- W/sum(W) #指标权重归一化
```

```
PRO <- function(x){
```

```
    sum(t(as.matrix(x))*W)
```

```
}
```

```
sel <- cbind(sel,apply(sel,1,PRO))
```

```
colnames(sel)[5] <- "综合评价值"
```

```
zpj <- sel[order(sel[,5]),]
```

```
write.csv(zpj, paste(de.outloc,"14. 模型综合评价结果.csv",sep=""))
```

**#9.10 综合各指标，决定选择模型 3：ARIAM(4,1,2)**

```
##_____
```

```
##第十部分：模型预测
```

```
##_____
```

**#用 ARIMA(4,1,2)模型预测并输出预测结果到外部文件：15. 模型预测结果.csv**

```
library(forecast)
```

```
foca <- forecast(ARIMA2,h=2)
```

```
pe <- foca$mean[2]
```

```
iel <- foca$lower[4]
```

```
ieu <- foca$upper[4]
```

```
len <- round((ieu-iel)/2,2)
```

```
fors <- paste("预测 2017 年 5 月 8 日股票收盘价所在区间为","[,round(iel,2),",",round(ieu,2),"],","即  
2017 年 5 月 8 日股票价格为",round(pe,2),"±",len,sep="")
```

```
write.table(fors,paste(de.outloc,"15. 模型预测结果.txt" ,sep=""))
```

## 8.2 Matlab 程序代码

以下 matlab 代码用于求解如下非线性规划问题

$$\begin{aligned} & \max \omega^T H \omega \\ & s. t. \begin{cases} \omega^T \omega = 1 \\ \omega > 0 \end{cases} \end{aligned}$$

这里  $\omega = (\omega_1, \omega_2, \omega_3, \omega_4)'$  为待求的预测均方误差、预测平均绝对误差、AIC、BIC 的权重系数向量,  $H = A^T A$  为 3 个模型各预测效果指标的相关系数矩阵,  $A$  为 3 个模型各预测效果指标数据标准化后构成的矩阵, 利用拉开档次法对模型进行综合评价等价于求解上述规划问题, 确定各指标的权重。

### 8.2.1 evaluation.m

%%该 m 文件适用于带线性、非线性约束的非线性目标函数优化问题

%%它还需两个 m 文件, fun.m 用于定义目标函数, mycon.m 用于给出非线性等式、不等式约束

%%一、目标函数及约束条件的定义

%1.目标函数: 由函数 fun 定义, 存于 fun.m 中

objective = @fun;

%2.线性等式 (Aeq x = beq)、不等式 (Aineq x <= bineq) 约束

Aeq = [];

beq = [];

Aineq = [];

bineq = [];

%3.非线性等式、不等式约束: 由函数 mycon 给出, 存于 mycon.m 中

nonlcon = @mycon;

%4.待求变量的迭代初始值(x0)、上下界(lb <= x <= ub)

x0 = [1/2;1/2;1/2;1/2];

lb = [1e-10;1e-10;1e-10;1e-10];

ub = [];

%%二、使用 GlobalSearch 对规划问题作全局最优化

%1.优化选项设置, 这里使用 fmincon 非线性优化函数, 算法为内点法 (interior-point method)

opts = optimoptions(@fmincon,'Algorithm','interior-point');

%2.创建优化问题: createOptimProblem('solverName','ParameterName',ParameterValue,...)

%solverName 为求解器名, 这里为 fmincon, ParameterName 为参数名 (共 10 个), ParameterValue

为参数值

```
problem
createOptimProblem('fmincon','objective',objective,'Aineq',Aineq,'bineq',bineq,'Aeq',Aeq,'beq',beq,'nonlc
on','nonlcon','x0',x0,'lb',lb,'ub',ub,'options',opts);
```

%3.执行优化

```
gs = GlobalSearch;
[x,f] = run(gs,problem)
```

### 8.2.2 fun.m

%H 矩阵由 R 求得

```
function f=fun(x)
f= -(x'*[2 -1.998030526    1.998370835 -1.99980247
-1.998030526    2 -1.999983866    1.996586081
1.998370835 -1.999983866    2 -1.997039159
-1.99980247 1.996586081 -1.997039159    2
]*x);
```

### 8.2.3 mycon.m

```
function [c,ceq]=mycon(x)
c = 0; %非线性等式约束
ceq = x'*x-1; %非线性不等式约束
```