analysis\_1.1.R

jiaru2014

Tue Jun 7 16:24:23 2016

# 分析股票交易数据  
# Author: JiaRu   
# Email: jiaru2014@126.com  
# Version 1.1  
# 1. 增加成交量的分析   
# 2. 改进excel文件读取方式，增加异常处理机制。  
# 3. 把单个股票数据的读取、清洗、初步处理的过程封装成函数  
# PreAnalysis(stocknumber)，以后分析其他股票时可以直接调用。  
# ===================================================================  
  
rm(list = ls())  
  
require(gdata) # gdata::read.xls(, colClass = rep("character", 5))

## Loading required package: gdata

## gdata: read.xls support for 'XLS' (Excel 97-2004) files ENABLED.

##

## gdata: read.xls support for 'XLSX' (Excel 2007+) files ENABLED.

##   
## Attaching package: 'gdata'

## The following object is masked from 'package:stats':  
##   
## nobs

## The following object is masked from 'package:utils':  
##   
## object.size

require(data.table)

## Loading required package: data.table

##   
## Attaching package: 'data.table'

## The following object is masked from 'package:gdata':  
##   
## last

require(magrittr) # 只是为了用 `%>%`

## Loading required package: magrittr

require(ggplot2)

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 3.2.4

require(assertthat) # a tool for Defensive Programming

## Loading required package: assertthat

require(stringr) # 文字处理包

## Loading required package: stringr

# 0. 列出数据集中所有的股票代码 ========================================  
  
stocks <-   
 list.files("JP\_stock\_file", pattern = "^.+[^part1|2]\\.xls$") %>%   
 str\_replace(pattern = "\\.xls", replacement = "")  
cat("以下是数据集中所有可分析的股票代码：", stocks, sep = "\n")

## 以下是数据集中所有可分析的股票代码：  
## 000001SH  
## 000006  
## 000539  
## 000829  
## 399001SZ  
## 600030  
## 601390  
## 601398  
## 601766  
## 601989

# 1. Define function: ===============================================  
# PreAnalysis(stock\_num)  
# Argus:   
# stock\_num: 股票代码，chr格式  
# Return:  
# 清理好的data.table  
  
PreAnalysis <- function(stock\_num)  
{  
 # 检查参数 ---------------------  
 stock\_num <- as.character(stock\_num)  
 assert\_that(  
 stock\_num %in% stocks  
 )  
 message("Call:", "股票代码：", stock\_num)  
   
 # 读文件 -------------------------------  
   
 tryCatch(  
 {  
 fseq <- .Platform$file.sep  
 file\_name <- paste0("JP\_stock\_file", fseq, stock\_num, ".xls")  
 file\_name1 <- paste0("JP\_stock\_file", fseq, stock\_num, ".part1", ".xls")  
 file\_name2 <- paste0("JP\_stock\_file", fseq, stock\_num, ".part2", ".xls")  
   
 message("正在读取文件 part 1 / 3 ...")  
 df <- read.xls(file\_name, colClass = rep("character", 5))  
 message("正在读取文件 part 2 / 3 ...")  
 df1 <- read.xls(file\_name1, colClass = rep("character", 5))  
 message("正在读取文件 part 3 / 3 ...")  
 df2 <- read.xls(file\_name2, colClass = rep("character", 5))  
   
 dt <- rbind(df, df1, df2) %>% as.data.table()   
 message("文件读取成功！")  
 },  
 error = function(e) paste("读取文件错误：", e)  
 )  
   
 # 检查异常值 ----------------------------  
 # 看有没有重复的行。  
 if (anyDuplicated(dt)){  
 setkey(a, "TDate", "MinTime")  
 dt <- unique(dt)  
 message("数据集中有重复行，已删除重复行。")  
 } else {  
 message("没有重复行。")  
 }  
   
 # 看每一天时间是否齐全。  
 # 股票交易时间：上午时段9:30-11:30，下午时段13:00-15:00  
 # 所以正常来说每天应该有 4\*60+1 = 241 个一分钟（行）  
 checktime <- dt[, .N, by = TDate][N != 241]  
 if (nrow(checktime) == 0) {  
 message("每个交易日的交易信息都是完整的！")  
 } else {  
 message("以下交易日缺少部分交易信息：")  
 print(checktime)  
 }  
 # 整理数据格式 ----------------------------  
 # 转换日期时间  
 dt[, DateTime := as.POSIXct(paste(TDate, MinTime), format = "%Y%m%d %H%M", tz = "PRC")]  
 dt[, TDate := as.Date(TDate, "%Y%m%d")]  
   
 # EndPrc 和 MinTq 转换成数字格式  
 dt[, EndPrc := as.numeric(EndPrc)]  
 dt[, MinTq := as.integer(MinTq)]  
   
 # 计算收益率  
 dt[, return := c(NA, diff(EndPrc)/EndPrc[-.N])]  
   
 return(dt)   
}  
  
# 2. 研究股票 "000001SH" ====================================================  
  
stock\_num <- "000001SH"  
dt <- PreAnalysis(stock\_num)

## Call:股票代码：000001SH

## 正在读取文件 part 1 / 3 ...

## 正在读取文件 part 2 / 3 ...

## 正在读取文件 part 3 / 3 ...

## 文件读取成功！

## 没有重复行。

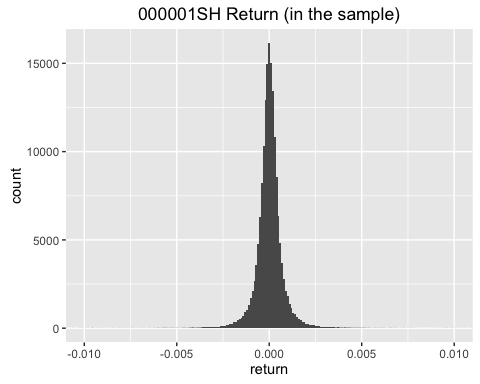
## 以下交易日缺少部分交易信息：

## TDate N  
## 1: 20140715 240  
## 2: 20150821 240

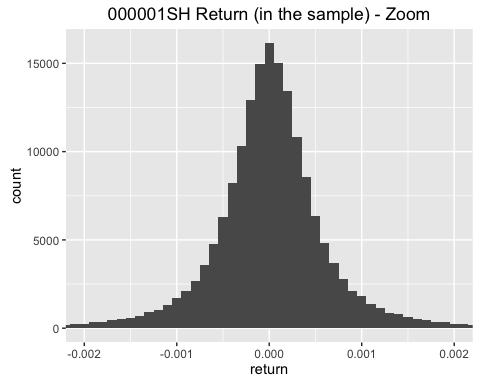
# 3. 研究收益率分布 ====================================================  
# 区分in-sample和out-sample, 看一下in sample的收益率分布，  
# 再看out of sample 的收益率处于in sample收益率分布的哪个位置。  
  
# 以"2016-05-01"为界，区分in-sample和out-of-sample  
# 也可以用其他方法分界：个数or比例  
  
bp\_time <- as.Date("2016-05-01") #以日期为界  
bp\_num <- 100L # 100个  
bp\_prop <- 1e-4 # 千分之一  
  
dt\_in <- dt[TDate <= bp\_time][-1] # in sample  
dt\_out <- dt[TDate > bp\_time] # out of sample  
  
# 看一下return的分布  
quantile(dt\_in$return, probs = seq(0, 1, 0.1))

## 0% 10% 20% 30% 40%   
## -2.068449e-01 -7.013617e-04 -3.971231e-04 -2.274981e-04 -1.021244e-04   
## 50% 60% 70% 80% 90%   
## 5.993403e-06 1.167950e-04 2.410418e-04 4.068348e-04 7.158247e-04   
## 100%   
## 3.394122e-01

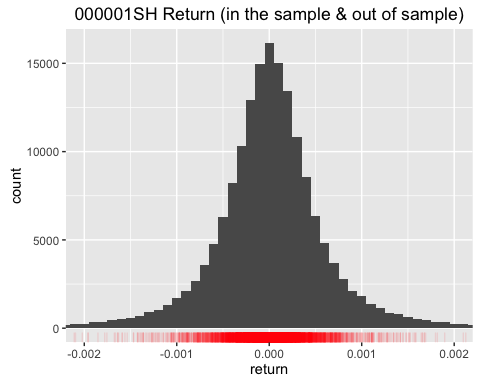
# 画return的直方图  
c1 <- ggplot(data = dt\_in, aes(x = return)) +  
 geom\_histogram(binwidth = 1e-4) +  
 coord\_cartesian(xlim = c(-1e-2, 1e-2)) +  
 ggtitle(paste(stock\_num, "Return (in the sample)"))   
c1



# 还是画return的直方图，调整一下x坐标轴：  
c2 <- ggplot(data = dt\_in, aes(x = return)) +  
 geom\_histogram(binwidth = 1e-4) +  
 coord\_cartesian(xlim = c(-2e-3, 2e-3)) +  
 ggtitle(paste(stock\_num, "Return (in the sample) - Zoom"))   
c2



# 然后看一下out of sample 的 return 在什么位置：  
c2 + geom\_rug(  
 data = dt\_out, aes(x = return),  
 col = "red", alpha = 0.1  
) + ggtitle(paste(stock\_num, "Return (in the sample & out of sample)"))



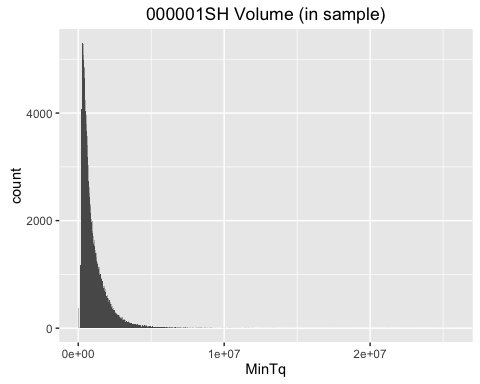
# 4. 研究成交量分布 ====================================================  
  
quantile(dt\_in$MinTq, probs = seq(0, 1, 0.1))

## 0% 10% 20% 30% 40% 50%   
## 0.0 275748.6 360874.6 452786.4 563812.0 699033.0   
## 60% 70% 80% 90% 100%   
## 887465.6 1147106.2 1517382.8 2149724.4 25728261.0

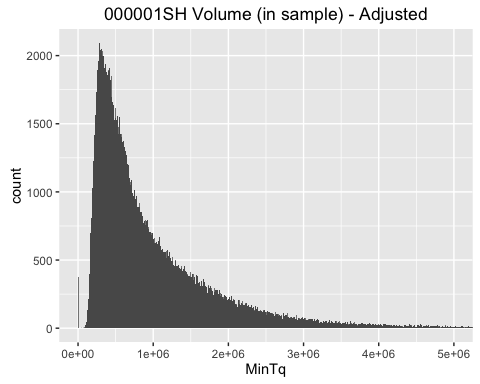
quantile(dt\_out$MinTq, probs = seq(0, 1, 0.1))

## 0% 10% 20% 30% 40% 50% 60%   
## 0.0 266065.2 312875.2 363175.8 417264.4 477310.0 550952.2   
## 70% 80% 90% 100%   
## 629515.0 742992.2 937669.4 2693441.0

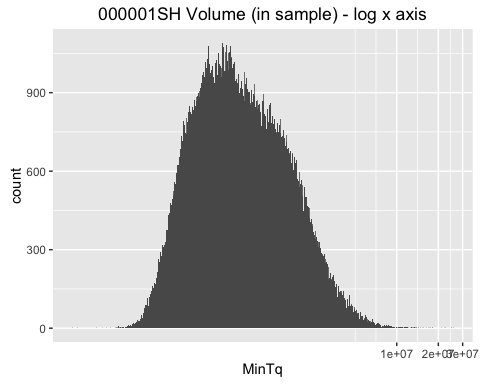
c3 <- ggplot(data = dt\_in, aes(x = MinTq)) +  
 geom\_histogram(bins = 1000) +  
 ggtitle(paste(stock\_num, "Volume (in sample)"))  
c3



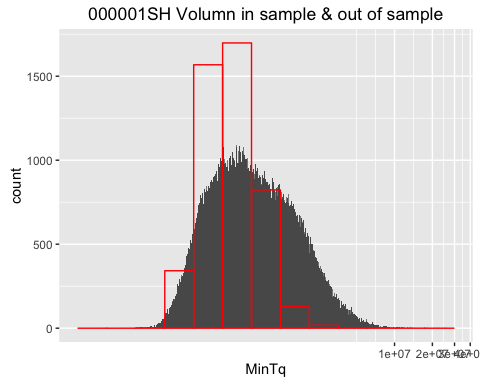
# 调整一下 x轴坐标和binwidth:  
c4 <- ggplot(data = dt\_in, aes(x = MinTq)) +  
 geom\_histogram(binwidth = 1e4) +  
 coord\_cartesian(xlim = c(0, 5e6)) + # <<-- 不断调整x轴坐标和binwidth  
 ggtitle(paste(stock\_num, "Volume (in sample) - Adjusted"))  
c4



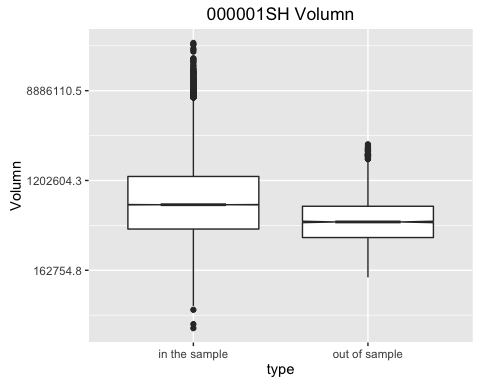
# 观察上面的直方图，发现成交量的分布很像对数正态分布，  
# 将x轴改为对数坐标轴，重新画上边的图：  
# 去掉成交量为0的值  
  
c5 <- ggplot(data = dt\_in[MinTq > 0], aes(x = MinTq)) +  
 geom\_histogram(bins = 500) +  
 coord\_cartesian() +   
 ggtitle(paste(stock\_num, "Volume (in sample) - log x axis")) +  
 scale\_x\_continuous(trans = "log1p") # <<-- 也试过其他参数：trans = "log", "log10", ...  
c5



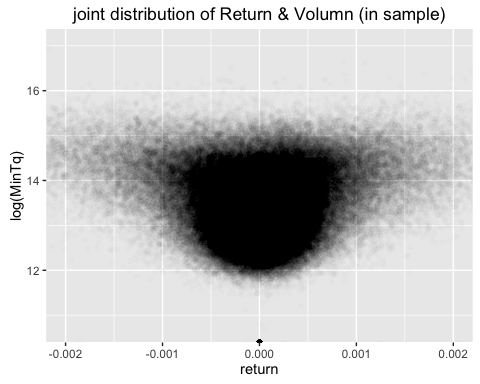
# 加入样本外预测  
m <- 1/nrow(dt\_in[MinTq > 0])\*nrow(dt\_out[MinTq > 0])  
c6 <- c5 +   
 geom\_histogram(  
 data = dt\_out[MinTq > 0], aes(x = MinTq),  
 bins = 500\*m, col = "red", alpha = 0  
 ) +   
 ggtitle(paste(stock\_num, "Volumn", "in sample & out of sample"))  
c6



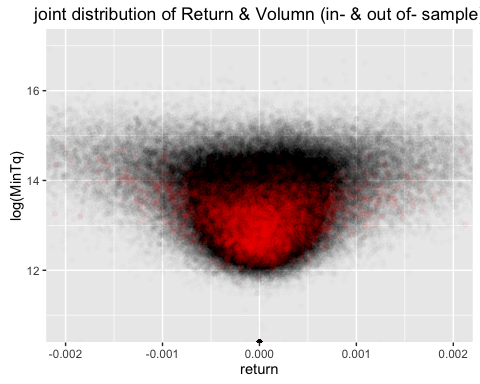
# 注：  
# 变量 m 的意思是in-sample样本量和out-of-sample样本量的比值，它用来调整  
# out-of-sample直方图（红色的部分）的bins。因为in-sample样本量和out-of-sample  
# 的样本量不同，直接把它们的直方图叠在一起，y轴上的freqency不具有可比性。  
# 所以我想的办法是调整bins，使得y轴上的freqency可以比较。  
#   
# 看起来样本外和样本内的成交量分布并不一样：样本外的成交量分布更为集中  
  
# 另一种可视化方法：boxplot  
  
dt\_boxplot <- rbind(  
 data.table(type = "in", Volumn = dt\_in$MinTq),  
 data.table(type = "out", Volumn = dt\_out$MinTq)  
)   
dt\_boxplot[, type := as.factor(type)]  
  
c7 <- ggplot(dt\_boxplot[Volumn > 0], aes(x = type, y = Volumn)) +   
 geom\_boxplot(notch = TRUE) +  
 #geom\_jitter(col = "yellow", alpha = 0.01) +   
 scale\_y\_continuous(trans = "log") +  
 ggtitle(paste(stock\_num, "Volumn")) +  
 scale\_x\_discrete(labels = c("in the sample", "out of sample"))  
c7



# 结论和上面的一样，样本外的成交量数据均值和方差都小于样本内成交量数据。  
  
  
# 5.收益率和成交量的联合分布 ====================================================  
  
c8 <- ggplot(dt\_in, aes(x = return, y = log(MinTq))) +   
 coord\_cartesian(xlim = c(-2e-3, 2e-3)) +   
 geom\_point(alpha = 0.01) +   
 ggtitle("joint distribution of Return & Volumn (in sample)")  
c8



# 加入样本外数据  
  
c9 <- c8 + geom\_point(  
 data = dt\_out, mapping = aes(x = return, y = log(MinTq)),  
 col = "red", alpha = 0.05  
 ) + ggtitle("joint distribution of Return & Volumn (in- & out of- sample)")  
c9



# 非参数方法估计联合密度：  
c10 <- ggplot() +  
 geom\_density2d(data = dt\_out, aes(x = return, y = log(MinTq)), col = "red") +  
 geom\_density2d(data = dt\_in, aes(x = return, y = log(MinTq)))  
c10

## Warning: Removed 1 rows containing non-finite values (stat\_density2d).

## Warning: Removed 376 rows containing non-finite values (stat\_density2d).

