analysis_1.1.R

jiaru2014

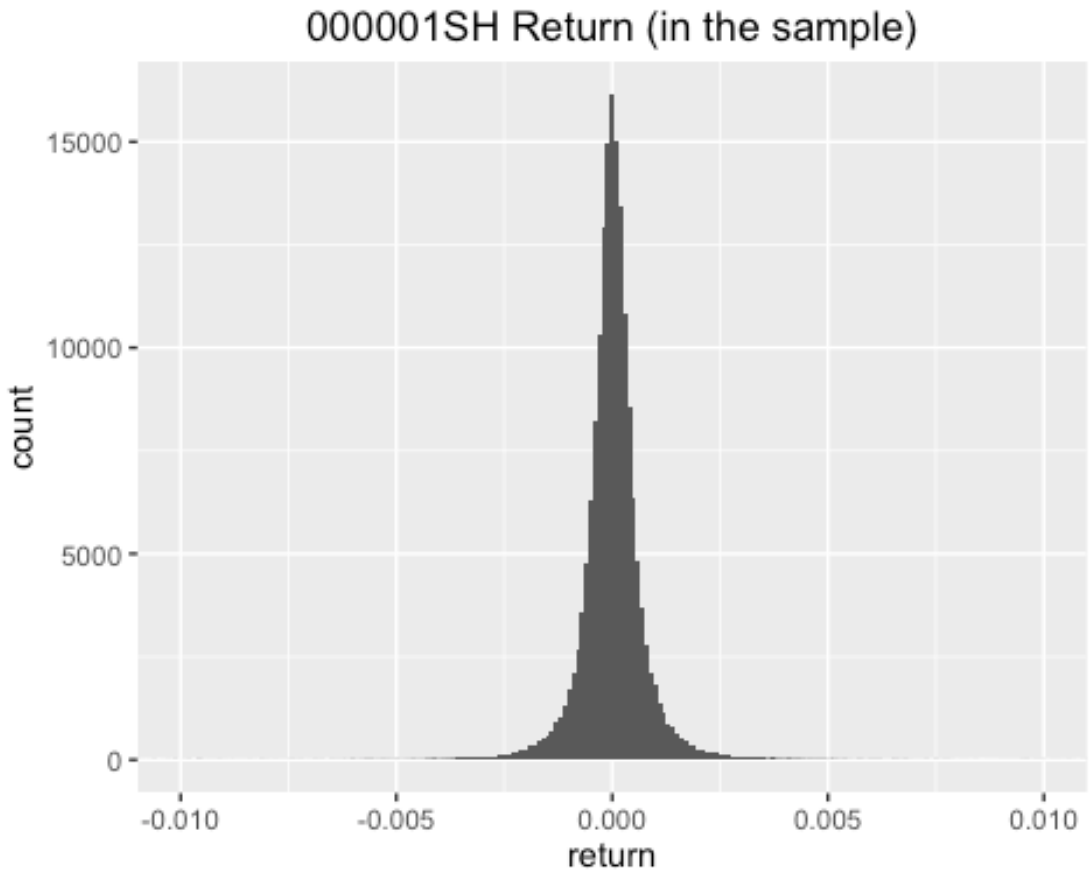
Tue Jun 7 16:24:23 2016

```
# 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))

require(data.table)

require(magrittr)  # 只是为了用 `%>%`

require(ggplot2)

require(assertthat)  # a tool for Defensive Programming

require(stringr)  # 文字处理包

# 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
```

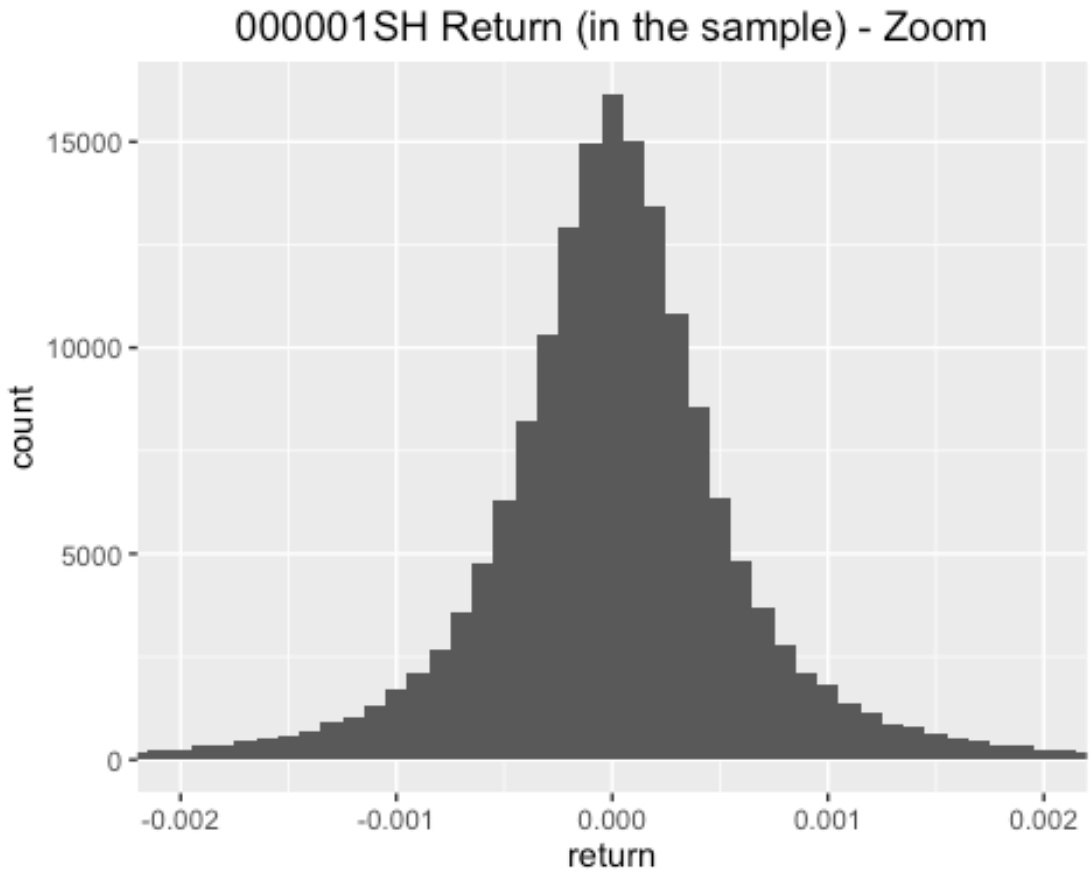
```r
# 1. Define function: ============================================
# PreAnalysis(stock_num)
# Argus:
#   stock_num: 股票代码，chr 格式
# Return:
#   清理好的 data.table

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：
```

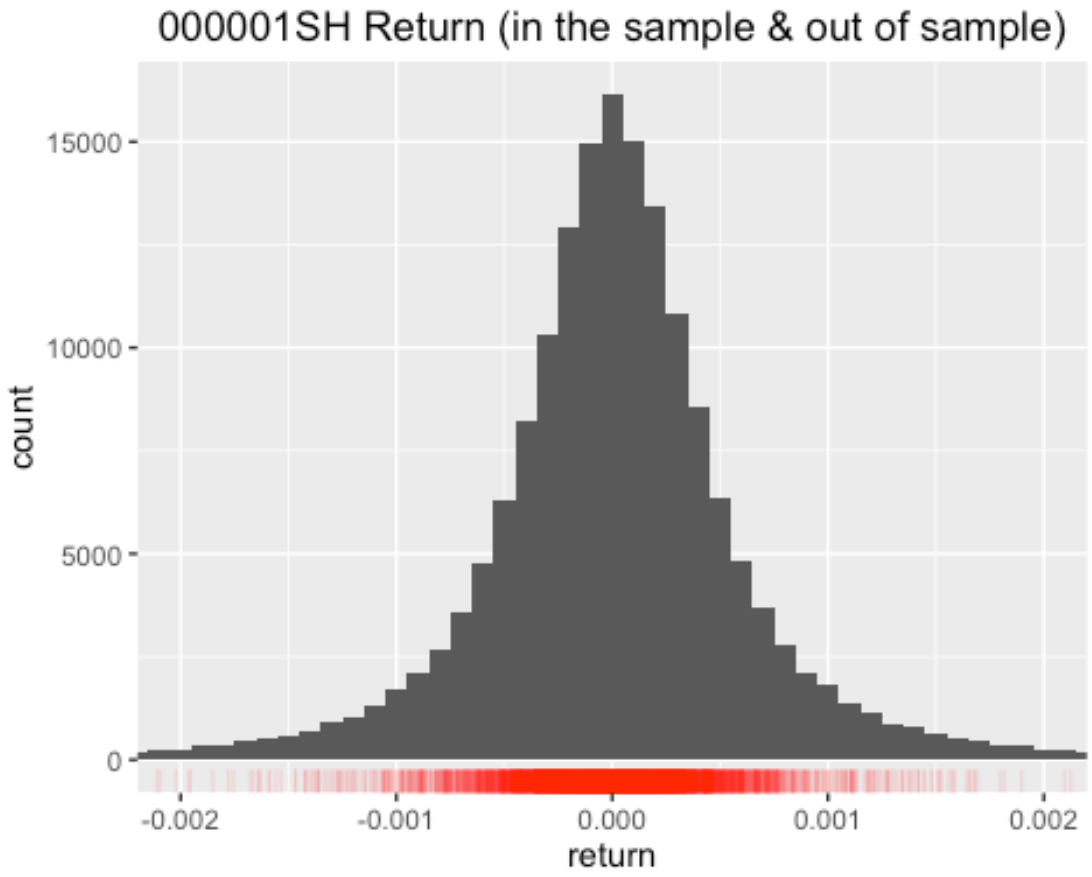
```
##       TDate   N
## 1: 20140715 240
## 2: 20150821 240

# 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-0
4
##           50%           60%           70%           80%           90%

##  5.993403e-06  1.167950e-04  2.410418e-04  4.068348e-04  7.158247e-0
4
##          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
```



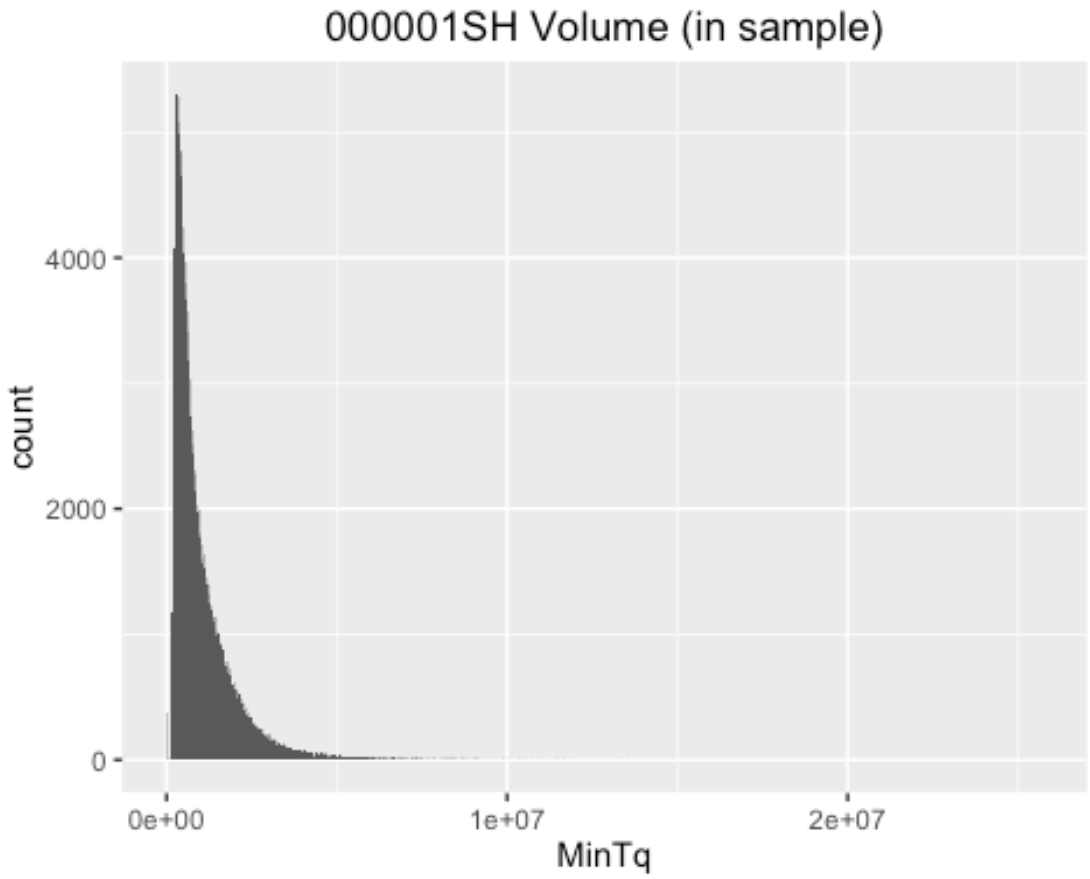

```r
# 还是画 return 的直方图，调整一下 x 坐标轴：
c2 <- ggplot(data = dt_in, aes(x = return)) +
  geom_histogram(binwidth = 1e-4) +
  coord_cartesian(xlim = c(-2e-3, 2e-3)) +
  ggtitle(paste(stock_num, "Return (in the sample) - Zoom"))
c2
```



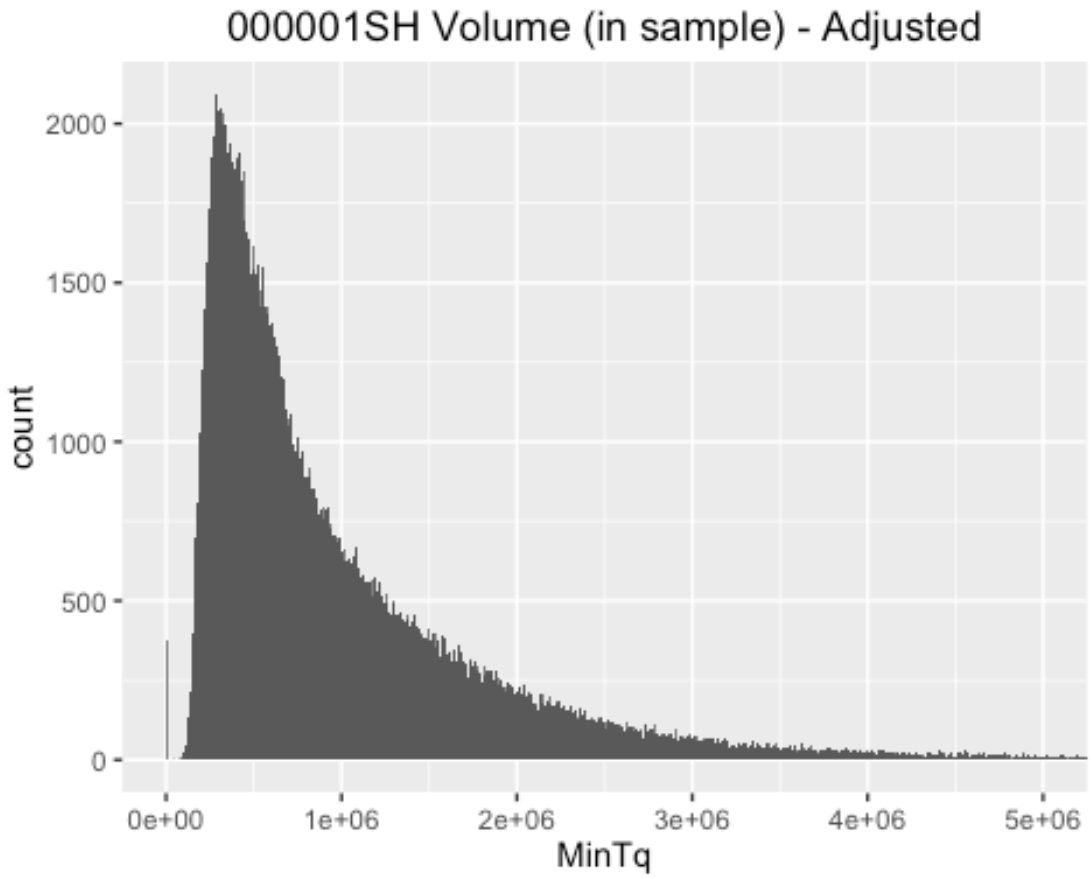

```
# 然后看一下out of sample 的 return 在什么位置:
c2 + geom_rug(
  data = dt_out, aes(x = return),
  col = "red", alpha = 0.1
) + ggtitle(paste(stock_num, "Return (in the sample & out of sample)"))
```



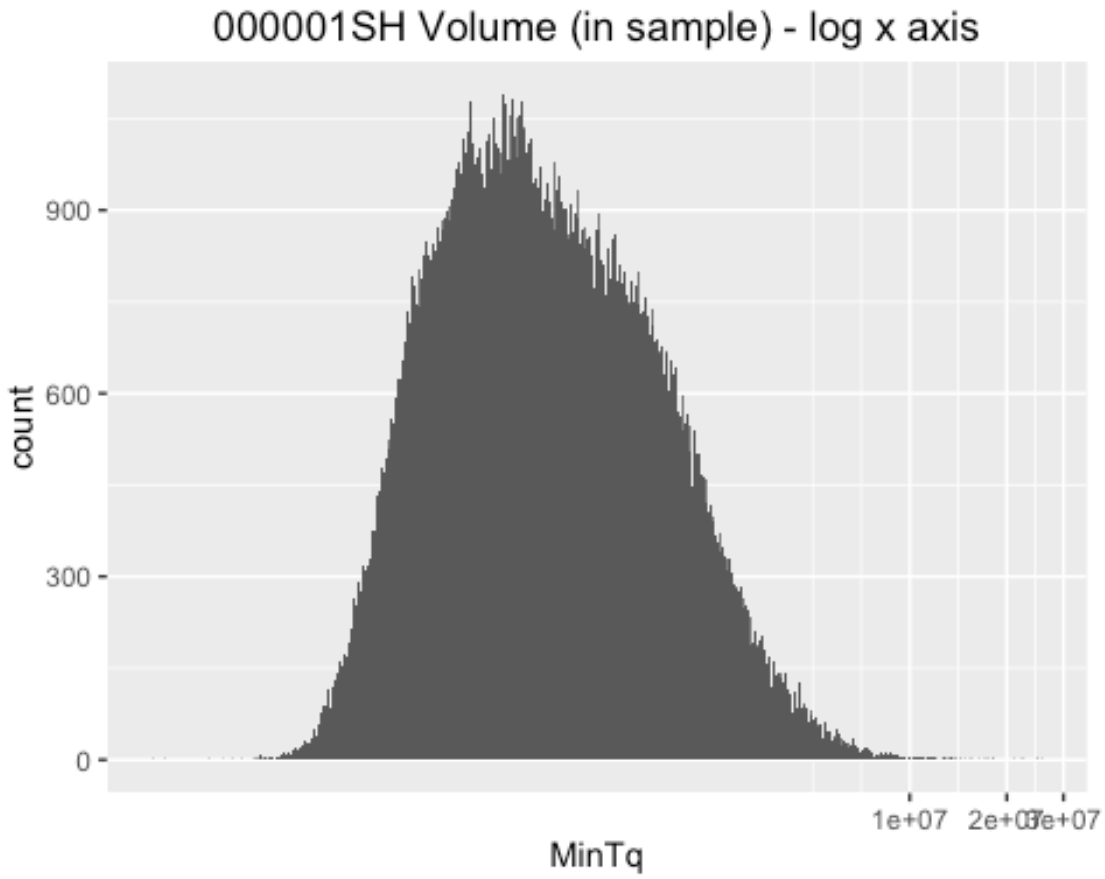

```
# 4. 研究成交量分布  ================================================
=

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
```



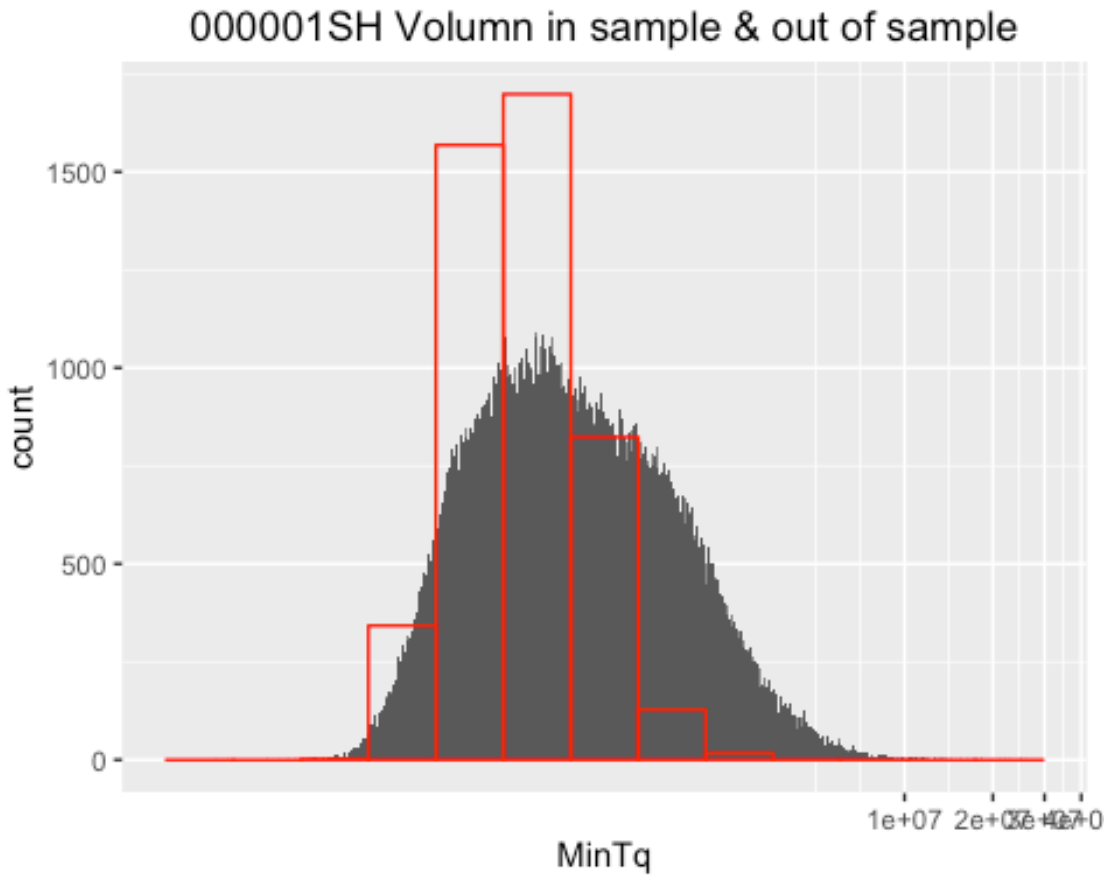

```
# 调整一下 x 轴坐标和 binwidth:
c4 <- ggplot(data = dt_in, aes(x = MinTq)) +
  geom_histogram(binwidth  = 1e4) +
  coord_cartesian(xlim = c(0, 5e6)) + # <<-- 不断调整 x 轴坐标和 binwidth
  ggtitle(paste(stock_num, "Volume (in sample) - Adjusted"))
c4
```




```
# 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 = "l
og", "log10", ...
c5
```

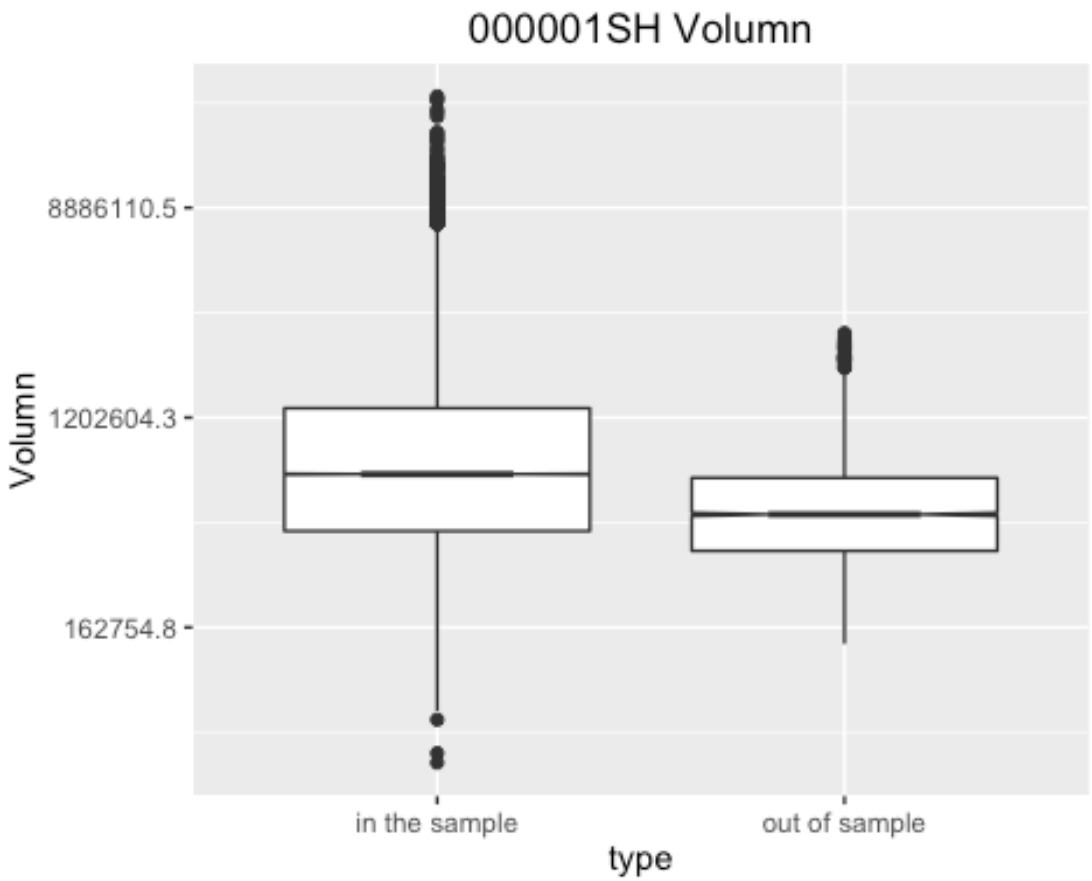



```
# 加入样本外预测
m <- 1/nrow(dt_in[MinTq > 0])*nrow(dt_out[MinTq > 0])
c6 <- c5 +
  geom_histogram(
    data = dt_out[MinTq > 0], aes(x = MinTq),
    bins = 500*m, col = "red", alpha = 0
  ) +
  ggtitle(paste(stock_num, "Volumn", "in sample & out of sample"))
c6
```

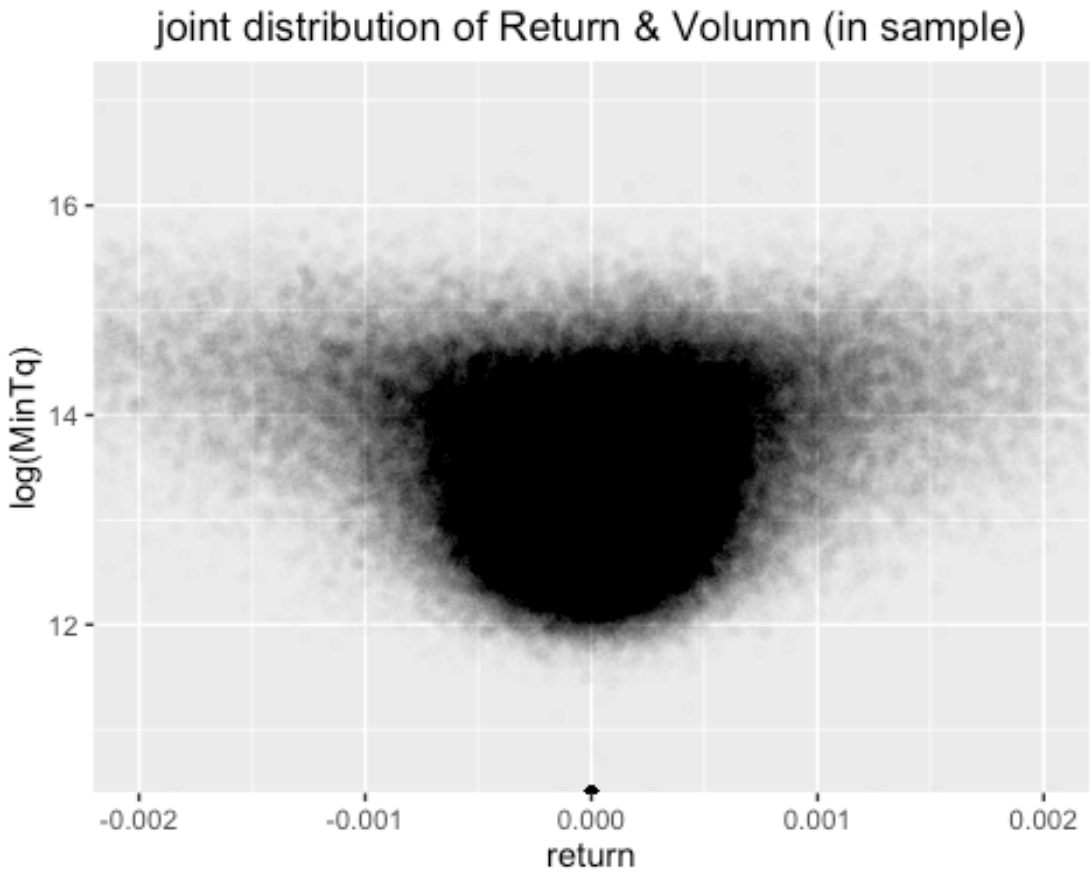



```
# 注：
# 变量 m 的意思是 in-sample 样本量和 out-of-sample 样本量的比值，它用来调整
# out-of-sample 直方图（红色的部分）的 bins。因为 in-sample 样本量和 out-of-s
ample
# 的样本量不同，直接把它们的直方图叠在一起，y 轴上的 freqency 不具有可比性。
# 所以我想的办法是调整 bins，使得 y 轴上的 freqency 可以比较。
#
# 看起来样本外和样本内的成交量分布并不一样：样本外的成交量分布更为集中

# 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")) +
```

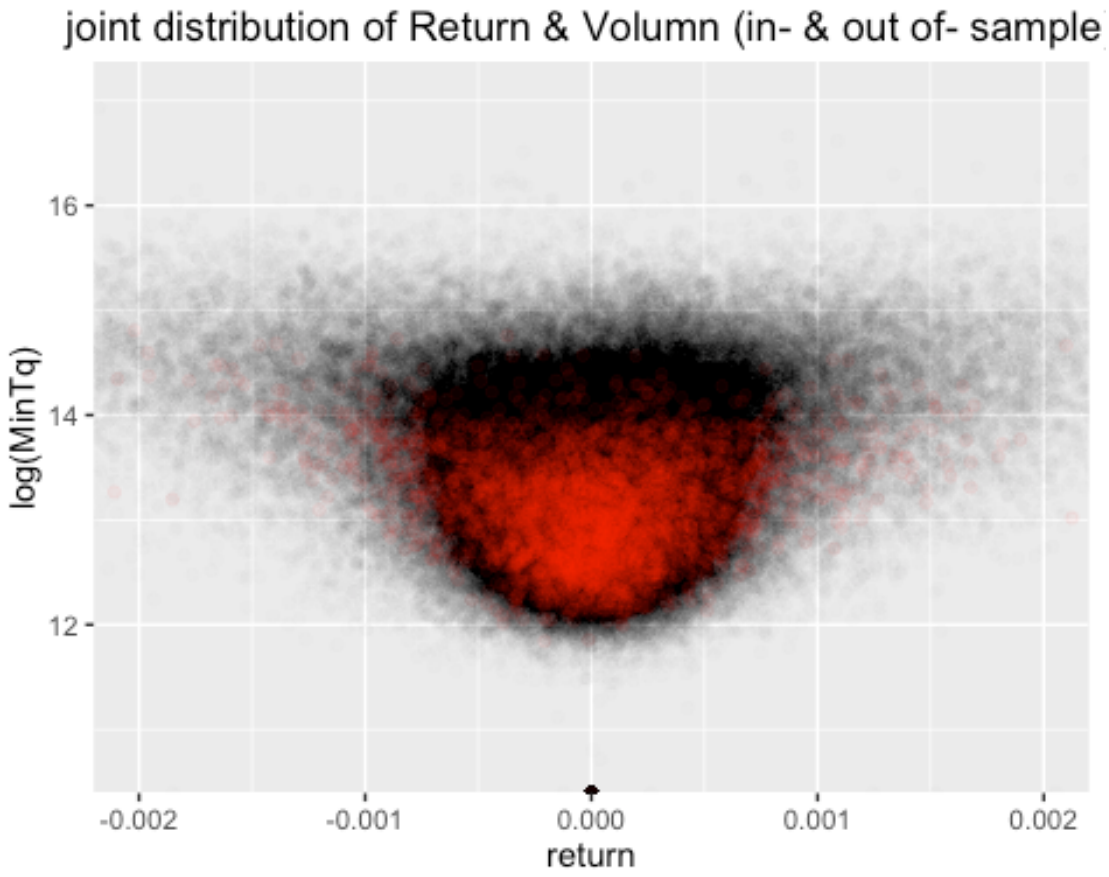
```
  scale_x_discrete(labels = c("in the sample", "out of sample"))
c7
```



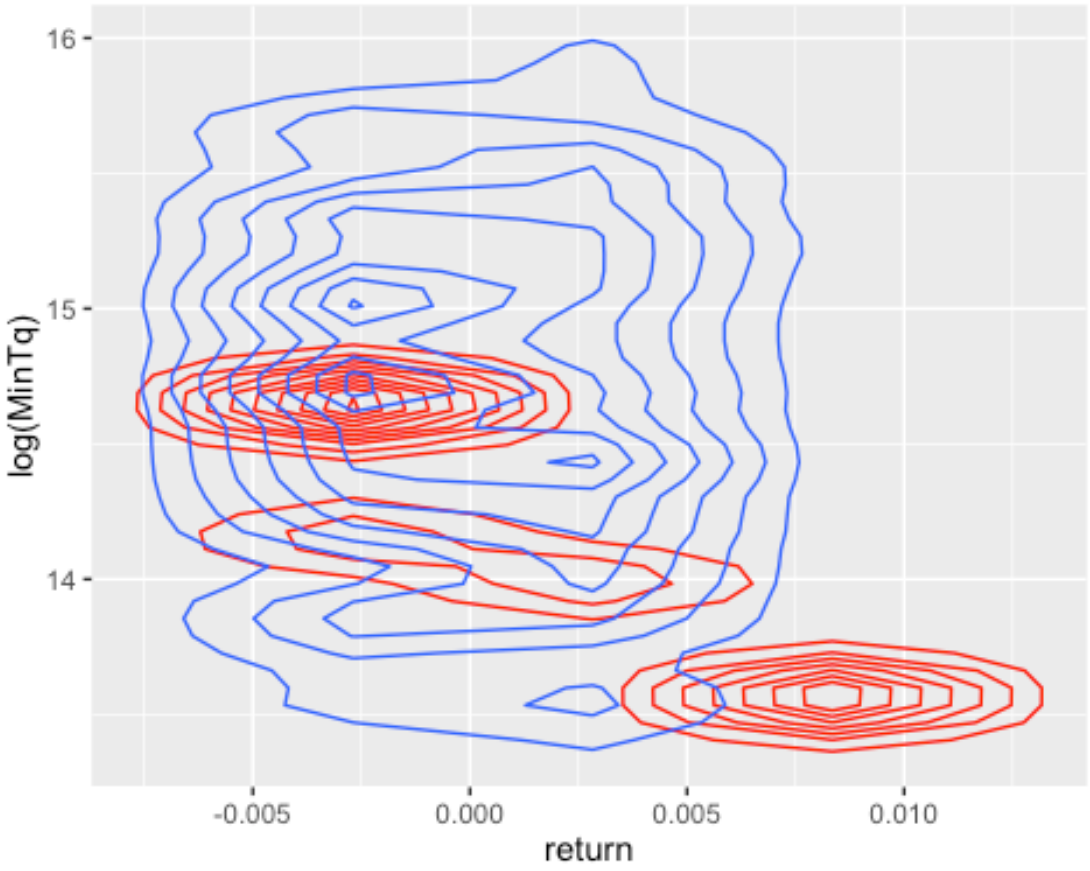

```
# 结论和上面的一样，样本外的成交量数据均值和方差都小于样本内成交量数据。


# 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- sam
ple)")
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_density2
d).

## Warning: Removed 376 rows containing non-finite values (stat_density
2d).
```


16
15
14
log(MinTq)
-0.005
0.000
0.005
0.010
return