# Search turning points in the trading history of a stock

Stock prices fluctuate everyday. It is impossible to pinpoint the bottom or peak of a stock. But by analyzing the trading history of a stock, we can figure a some unusual signals in the trading history of a stock. By analyzing the time series data of the stock trading history, we can easily figure out major turning points. Through analyzing stock data daily, we may be able to watch out when a major turning point appears. The following R program to analyze and visualize turning points in stock trading history. The program is used with MySQL database. The results of data analysis will be stored into the MySQL database and be summarized in the MySQL. After that, summary information will be loaded to R for plotting. We only present the R script here. The MySQL database structure will not be displayed.

```
library(tseries) # used for runs.test  library(DBI) # required for RMySQL
library(RMySQL) # for interacting with MySQL database    #===========
functions ==============  # execute query on a MySQL database  dbQuery <-
function(Query)  {    mysql <- dbDriver("MySQL")
con<-dbConnect(mysql,dbname="stocks",username="username",password="pass
word") # create connection    dbSendStatement(con,Query)    dbDisconnect(con)
#return(res)  }    # read a table from the MySQL database  dbRead <-
function(tablename)  {    mysql <- dbDriver("MySQL")
con<-dbConnect(mysql,dbname="stocks",username="username",password="pass
word") # create connection    myresult <- dbReadTable(con, tablename)
dbDisconnect(con)    return(myresult)  }    # read stock list from the MySQL
database  dbReadStockList <- function(tablename)  {    mysql <-
dbDriver("MySQL")
con<-dbConnect(mysql,dbname="stocks",username="username",password="pass
word") # create connection    mystocks <- dbReadTable(con, tablename)
dbSendQuery(con,"call drop_stocktables();")    dbDisconnect(con)
return(mystocks[,2])  }    # write analysis result to the MySQL database  dbWrite <-
function(tablename, myvar)  {    mysql <- dbDriver("MySQL")
con<-dbConnect(mysql,dbname="stocks",username="username",password="pass
word") # create connection    dbWriteTable(con, tablename, myvar,
overwrite=TRUE)    dbDisconnect(con)  }    # conduct the runs test for a given time
series  myruns <- function(vobs, n1, n2)  {    n <- nrow(vobs)    c3 <- rep(-1,n1)
c4 <- rep(1,n2)      rank <- c(c3,c4)    rank <- as.factor(rank)       dt <-
data.frame(vobs,rank)    dt <- dt[order(dt$vobs),]       res <- runs.test(dt$rank)
return(data.frame(statistic=res$statistic, p.value=res$p.value))  }    # do multiple
runs test for a given data set   # by setting different break point  mytests <-
function(obs)  {    res <- data.frame(varn=NA, nobs=NA, ndays=NA, date1=NA,
date2=NA, varname=NA, varvalue=NA, standardN=NA, p.value=NA)    nvar <-
length(obs)    nobs <- nrow(obs)    for (i in 2:(nvar-1))    {      vobs <- obs[,i]      #for
(n1 in (as.integer(nobs/2 - nobs/4)+1):(as.integer(nobs/2)))      for (n1 in 3:5)      {
```

```
    test <- myruns(vobs, n1, nobs-n1)        res <- rbind(res,data.frame(varn=i,
nobs=nrow(obs), ndays=n1, date1=obs$Date[1], date2=obs$Date[n1],
varname=colnames(obs[i]), varvalue=vobs[n1], standardN=unname(test$statistic),
p.value=unname(test$p.value)))      }    }    res <- res[-1,]    return(res)  }
myanalysis <- function(stocksymbol)  {    stockdata <-
read.csv(paste(stocksymbol,"csv",sep="."))    stockdata1 <- subset(stockdata,
as.POSIXct(stockdata$Date)>=as.POSIXct("2016-10-01"))      # set data frame for
output    myres <- data.frame(varn=NA, nobs=NA, ndays=NA, date1=NA,
date2=NA, varname=NA, varvalue=NA, standardN=NA, p.value=NA)    # construct
data set    data <- stockdata[1:(nrow(stockdata)-1),]    data1 <-
stockdata[2:nrow(stockdata),]    data2 <- data    # compute the up or down values
 for (i in 2:length(data)){       data2[,i] <- data[,i] - data1[,i]    }    data <- data2    #
   while (nrow(data) >= 10)    {      ngroups = nrow(data)      if (ngroups > 20) {
ngroups = 20 }      for (j in seq(10,ngroups,by=2))      {        mo <- data[1:j,]
res1 <- mytests(mo)        myres <- rbind(myres, res1)      }    # romove the latest
one     data <- data[-1,]    }      # remove the first row that includes NAs    myres <-
myrses[-1,]    # write to csv file    write.csv(myres,"runstest.csv")    # write to
MySQL data base    dbWrite(stocksymbol, myres)  }    #========== Main
program ==============  # Search turning point of any stocks automatically
  setwd("C:\\data\\stocks")  # import a stock list from the MySQL database
mystocks <- dbReadStockList("_stocks")  # conduct analysis for all stocks  for
(stocksymbol in mystocks)   {   print(stocksymbol)   myanalysis(stocksymbol)  }    #
summarize turningpoint data  dbQuery("call summarize_turningpoint();")    # look at
the general trend of turning point  tpcount <- dbRead("view_tpcount")
tpcount$tp_date <- as.Date(tpcount$tp_date, "%Y-%m-%d")  plot(tpcount$counts ~
tpcount$tp_date, xaxt="n", type="l")  axis(1, tpcount$tp_date,
format(tpcount$tp_date, "%b %d"), cex.axis = .7)      # look at the turning point of
each company  tpcompanycount <- dbRead("view_tpcompanycount")
tpcompanycount$tp_date <- as.Date(tpcompanycount$tp_date)
tpcompanycount$stocksymbol <- as.factor(tpcompanycount$stocksymbol)
#plot(tpcompanycount[,2:4])  #
https://www.r-bloggers.com/conditioning-and-grouping-with-lattice-graphics/
library(lattice)  xyplot(countsum ~ tp_date | factor(stocksymbol),
data=tpcompanycount, pch=19,        main="Turning Point Significance Count
Sum", xlab="date",  ylab="count",layout=c(3,1),type=c("p","g"))
```