Assignment 1

Philipp Epstein

10/14/2018

Introduction

This report is written for the 1st assignment of the Financial Trading Systems course 2018 at the Bond University. During the first weeks we learned how to use the Blotter Framework in R to implement a trading strategy, backtest the strategy on a set of instruments (stock, futures, etc.) and visualize meaningful results which give insights about the strategy. Moreover, we learned how to compare the strategy to another alternative (e.g. a buy and hold strategy). The trading idea, implementation and analysis and results are summarized in the following report.

Trading Idea

Smash Day by Larry Williams

The trading idea which is used for this report is based on the Smash Day strategy introduced by Larry Williams in his book "Long Term Secrets to short term trading" 2012. The basic theoretical foundation is set by the concept of autocorrelation. There are two autocorrelation principles which can be empirically observed. The positive autocorrelation which describes the fact that if we observe an upward movement it is more likely to observe an upward movement in the future. The same is valid for a negative trend. However, the negative autocorreleation can be observed on short term time periods. We can expect a price to bounce up again after a sharp down movement and vice versa. Larry William's Smash Day pattern is based on the observation of this autocorrelation behaviour. A Smash Day pattern is initialized when we observe a close which is lower than the previous day's low. This looks like a breakout to the downside. If the very next day the price moves opposite the Smash Day and trades above the high of the Smash Day this is according to Williams a buy signal. Williams explains this buy signal with a feeling of hurt felt by the public as a result of an unfulfilled breakout. The public would want to get back on track and the price responds with a reversal. The exact opposite is true for a Sell setup. The Smash Day is set up when we observe a close above the previous day's high. The sell signal is initiated when the price reverses immidiately the next day and falls below the smash day's low.

Own variation of the Smash Day trading strategy

In addition to the Smash Day pattern, I have incorporated another rule for the buy signal and 2 additional components for the sell signal. I want to make sure that I only enter long when we are observing a longlasting upward trend. In order to do this buy orders are only performed when the current price is above the exponential moving average of the last 200 days. After that the Smash Day pattern is checked. For the sell I incorporated two "harder" conditions before checking whether we observe a Smash Day (sell). The first is that a maximum holding period can be set in order to make sure that a position is not open for too long. The second sell case is when the price falls below the 200 day exponential moving average the position is liquidated. If both is not the case a check for the smash day pattern is performed. If we observe it, we sell the current position.

These variations are incorporated to reduce the risk of falling prices.

Implementation of the trading idea

Part A: Initialization

Step 1: General Setup

The environment has to be cleared in order to ensure a reproducable setup. Moreover, the blotter library and the INF361Course library are loaded.

```
# Clear Environment
rm(list=ls())

# Loading libraries
library(blotter)
library(INFT361Course)
```

Step 2: Setting the Variables

The variables set in the next section can be adjusted to test the strategy with different parameters. The critical variables which should be adjusted to test the strategy are: * daterange * emaPeriod * maxHoldingPeriod * instrumentlist * BuyHoldInstrument

```
# Set values:
startCapital <- 1e+6
transactionCost <- -20
daterange <- '2010::2014'
emaPeriod <- 200
maxHoldingPeriod <- 20

InstrumentDirectory <- "~/Desktop/R/DownloadedData/"
instrumentlist <- c("SAP.csv","DBK.csv", "HEI.csv")
BuyHoldDirectory <- "~/Desktop/R/DownloadedData/"
BuyHoldInstrument <- "DAXEX.csv"

currency("EUR")
Sys.setenv(TZ="UTC")
initdate <- '1999-12-31'
startdate <- '2000-01-01'
enddate <- '2018-12-31'
portfolioname <- "Smash Day"
accountname <- portfolioname
```

Step 3: Presetup for plotting graphs

Some setup for plotting the graphs in the visualization is performed here to ensure that the Theme is available later, when it is needed and used.

```
# Settings for graph
myTheme <- chart_theme()
myTheme$col$up.col <- 'lightblue'
myTheme$col$dn.col <- 'brown'
myTheme$col$dn.border <- 'lightgray'
myTheme$col$up.border <- 'lightgray'

# Concatenate string for EMA with input parameter
addEMAString <- paste("add_EMA(n=",emaPeriod,")",sep = "")
```

Step 4: Initializing the portfolio

The portfolio takes the instrumentlist which includes all the stocks we defined in Step 2.

```
# Clear portfolio and Account
suppressWarnings(rm("account.Smash Day","portfolio.Smash Day",
                    "account.buyhold","portfolio.buyhold",
                    pos=.blotter))

# Initialize Portfolio and Account
initPortf(portfolioname,
          instrumentlist,
          initDate=initdate,
          currency="EUR")

initAcct(accountname,
         portfolios=portfolioname,
         initDate=initdate,
         initEq=startCapital,
         currency="EUR")
```

Part B: Bar by bar processing

Step 1: Go through the data bar by bar

In this step the bar by bar processing is implemented. Thus, a for loop is implemented to loop through the instrumentlist. For each instrument in the list the data is loaded and the exponential moving average is added to the data. Furthermore, another for loop is used to loop through the dates bar by bar. In this step the defined strategy is applied.

```
for (instrument in instrumentlist) {
    LoadCourseFile(InstrumentDirectory, instrument, debugme = TRUE, dates = daterange)

    # Initialize the instrument
    stock(instrument, currency = "EUR")

    # Load the XTS file
    symbol <- get(instrument)

    # Calculate the Exponential Moving Average
    ema <- EMA(symbol$Close, n=emaPeriod)

    # Merge the xts file with the Exponential Moving Average
    symbol <- merge(symbol,ema)
    assign(instrument,symbol)
  # Starting to go bar by bar through using a "for loop"
  for (i in (emaPeriod + 1):(nrow(symbol) - 1)) {
    # Dates
    CurrentDate <- time(symbol[i])
    TomorrowDate <- time(symbol[i + 1])

    # Today's variables
    CloseToday <- as.numeric(symbol[i, "Close"])
    EMA_today <- as.numeric(symbol[i, "EMA"])
    LowToday <- as.numeric(symbol[i, "Low"])
    HighToday <- as.numeric(symbol[i, "High"])
```

```
# Yesterday's variables
LowYesterday <- as.numeric(symbol[i - 1, "Low"])
HighYesterday <- as.numeric(symbol[i - 1, "High"])

# Tomorrow's variables
OpenTomorrow <- as.numeric(symbol[i + 1, "Open"])
LowTomorrow <- as.numeric(symbol[i + 1, "Low"])
HighTomorrow <- as.numeric(symbol[i + 1, "High"])

# Config
Equity <- getEndEq(accountname, CurrentDate)
Position <-
  getPosQty(portfolioname, Symbol = instrument, Date = CurrentDate)

# Check whether we have a position
if (Position == 0) {
  # Start checking BUY rules

  # Check whether we have a Smash Day (Buy).
  # Smash Day (Buy) is when Todays Close is below Yesterdays Low.
  if (CloseToday < LowYesterday) {
    # Smash Day (Buy)

    #Check whether todays close is above today's EMA
    if (CloseToday > EMA_today) {

      # BUY RULE: If today was a smash day place a STOP BUY order
      # at todays high price for the next day.
      # (Buy tomorrow for 'price >= todays high')

      ######
      # Simulate STOP BUY order:
      ######

      # Option 1 to check: Check whether the open price tomorrow
      # is above today's high and add the transaction tomorrow at
      # tomorrow's open price.

      # Option 2 to check: Check whether today's high was lower
      # than tomorrows high and add the transaction tomorrow
      # at today's high price

      # Check Option 1
      if (OpenTomorrow > HighToday) {
        # Don't trade at the day before the last day
        if (CurrentDate != time(symbol[nrow(symbol) - 1])) {
          # Calculate the buy quantity
          BuyQuantity <- as.numeric(trunc(Equity / OpenTomorrow))
          # Add transaction
          addTxn(
            portfolioname,
            Symbol = instrument,
            TxnDate = TomorrowDate ,
```

```
            TxnPrice = OpenTomorrow,
            TxnQty = BuyQuantity,
            TxnFees = transactionCost
          )
          # Store the bar at which we placed the transaction
          BuyBar <- i
        }

      } else {
        # Check Option 2
        if (HighToday < HighTomorrow) {
          # Don't trade at the day before the last day
          if (CurrentDate != time(symbol[nrow(symbol) - 1])) {
            # Calculate the buy quantity
            BuyQuantity <- as.numeric(trunc(Equity / HighToday))
            # Add transaction
            addTxn(
              portfolioname,
              Symbol = instrument,
              TxnDate = TomorrowDate ,
              TxnPrice = HighToday,
              TxnQty = BuyQuantity,
              TxnFees = transactionCost
            )
            # Store the bar at which we placed the transaction
            BuyBar <- i
          }
        }
      }
    }
  }
} else {
  # We already have a position

  # Check the sell rules in the following order and sell at the
  # first condition which is satisfied.

  #####
  # SELL rules:
  #####

  # Rule 1: Sell if we hold the position longer than the specified
  # maximum holding period

  # Rule 2: Sell at tomorrow's opening price if the close price
  # today falls below the EMA

  # Rule 3: Sell if we meet the Smash Day (Sell) requirements.
  # Today's close must be higher than yesterday's high

  # Rule 4: If no sell rule can be applied and we reach the
  # second last day. Sell at the last day.
```

```
    # Check Rule 1:
    if ((i - BuyBar) > maxHoldingPeriod) {
      # Place the sell transaction at todays close price
      addTxn(
        portfolioname,
        Symbol = instrument,
        TxnDate = CurrentDate,
        TxnPrice = as.numeric(symbol[i, "Close"]),
        TxnQty = -Position,
        TxnFees = transactionCost
      )

    } else {
      # Check Rule 2:
      if (as.numeric(symbol[i, "Close"]) < EMA_today) {
        # Place the sell transaction at tomorrow's open price
        addTxn(
          portfolioname,
          Symbol = instrument,
          TxnDate = time(symbol[i + 1]),
          TxnPrice = OpenTomorrow,
          TxnQty = -Position,
          TxnFees = transactionCost
        )

      } else {
        # Check Rule 3:

        # Sell Rule 3: If today is a Smash Day (Sell) place an order tomorrow at todays
        # low price.

        # Simulate this behaviour:

        # Option 1 to check: Check whether the open price tomorrow is below today's
        # low and add the transaction tomorrow at tomorrow's open price.

        # Option 2 to check: Check whether today's low was larger than tomorrow's
        # low and add the transaction tomorrow at today's low price.

        # Check for Smash Day (Sell)
        if (CloseToday > HighYesterday) {
          # Check for Option 1
          if (OpenTomorrow < LowToday) {
            # Add Sell transaction tomorrow at tomorrow's open price
            addTxn(
              portfolioname,
              Symbol = instrument,
              TxnDate = time(symbol[i + 1]),
              TxnPrice = OpenTomorrow,
              TxnQty = -Position,
              TxnFees = transactionCost
            )
```

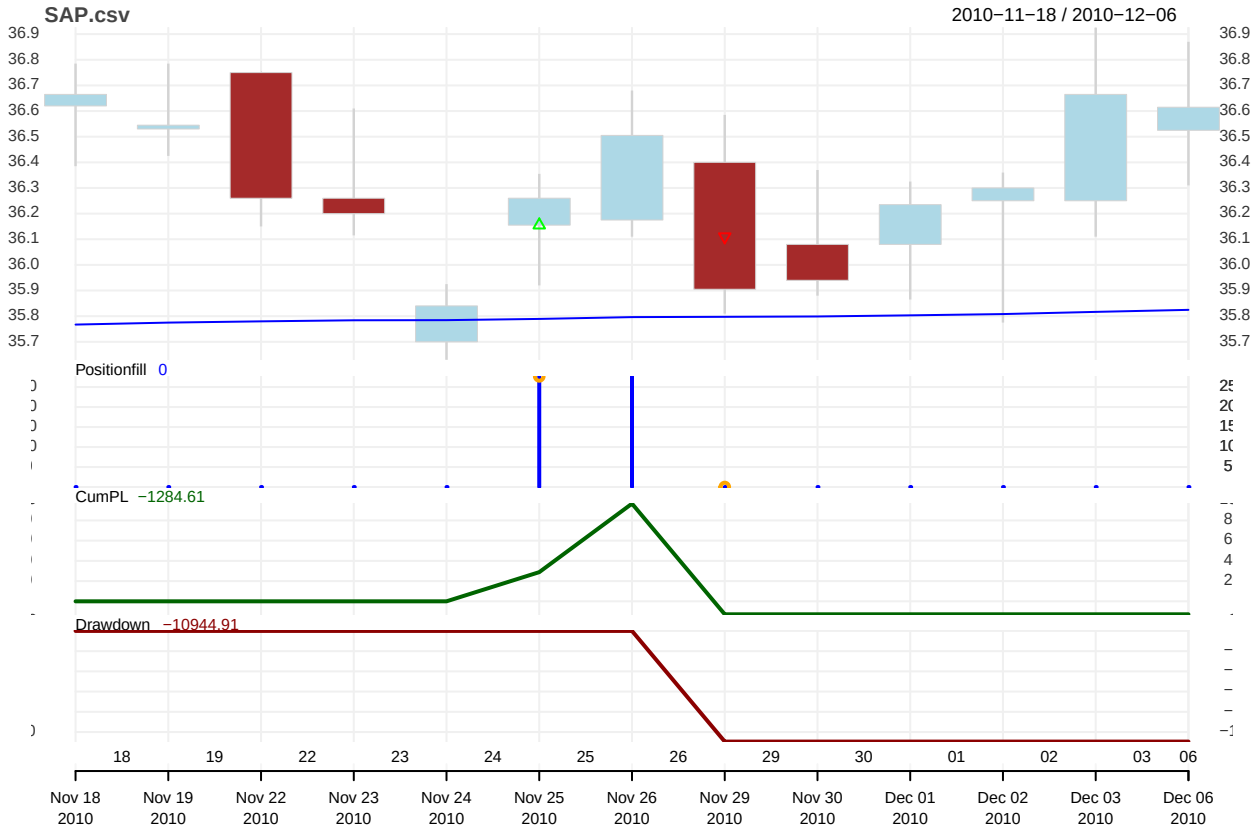
```
            } else {
              # Check for Option 2
              if (LowToday > LowTomorrow) {
                # Add Sell transaction tomorrow at today's low price
                addTxn(
                  portfolioname,
                  Symbol = instrument,
                  TxnDate = time(symbol[i + 1]),
                  TxnPrice = LowToday,
                  TxnQty = -Position,
                  TxnFees = transactionCost
                )
              }
            }
          } else {
            # Check Rule 4
            if (i == nrow(symbol) - 1) {
              # Add Sell transaction for the last day at the close price
              addTxn(
                portfolioname,
                Symbol = instrument,
                TxnDate = time(symbol[i + 1]),
                TxnPrice = as.numeric(symbol[i, "Close"]),
                TxnQty = -Position,
                TxnFees = transactionCost
              )
            }
          }
        }
      }
    }

    updatePortf(portfolioname, Symbols = instrument, Dates = CurrentDate)
    updateAcct(accountname, Dates = CurrentDate)
    updateEndEq(accountname, CurrentDate)

  } # End Bar-by-bar processing
} # End for loop for multiple instruments
```

Step 2: System Check

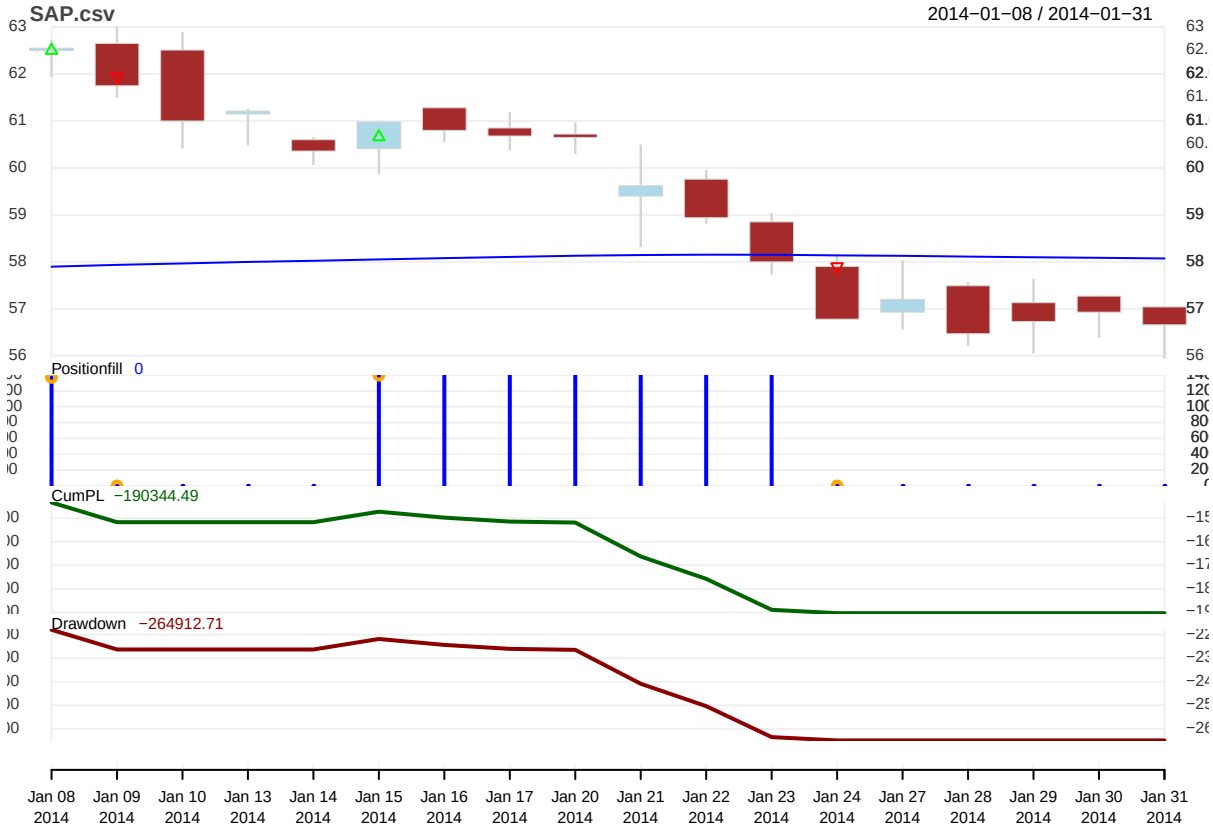
In order to make sure that the system works as designed the plots of some choosen transactions are printed in the following. The plots for the first, third and the 5th last trasnaction are plotted for every instrument in the instrumentlist. The plots can be checked manually and by this it can be ensured that the transactions were performed as expected.

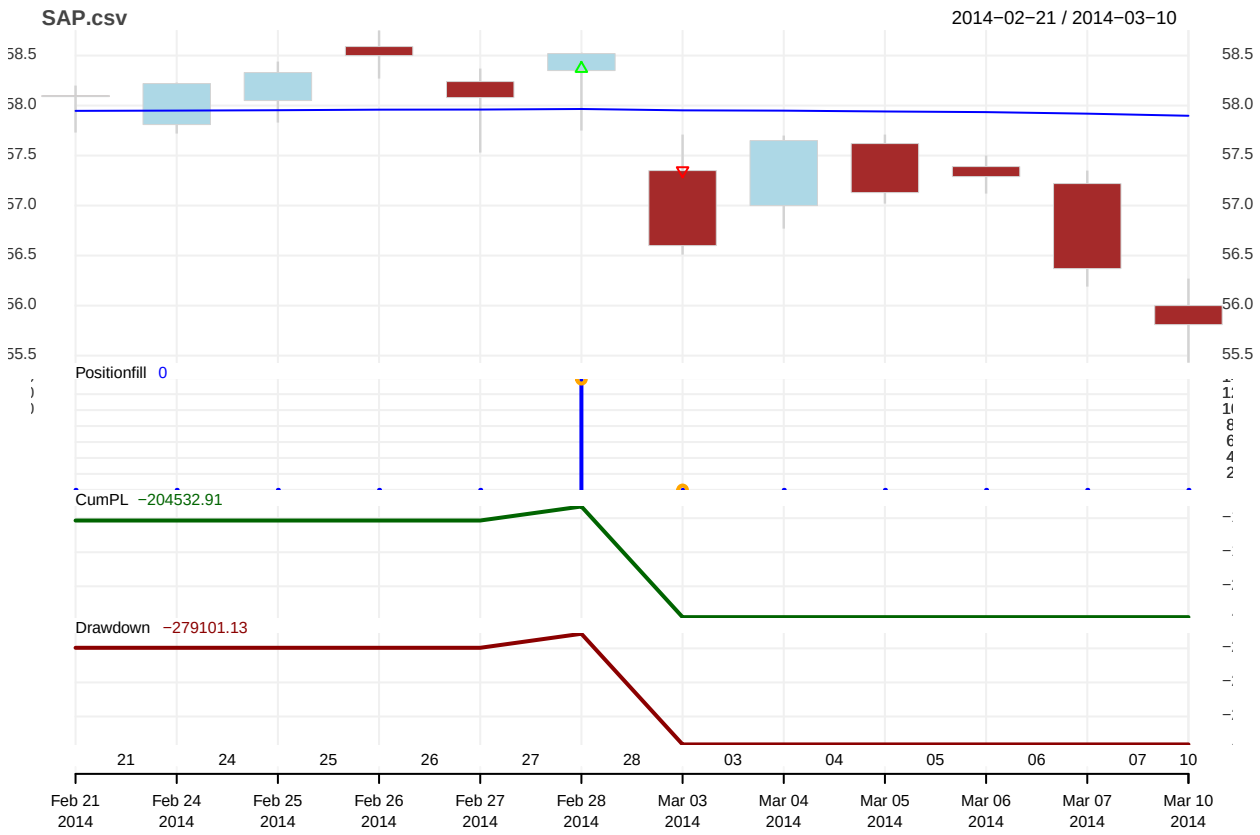
```
# Loop through all instruments in the instrumentlist
for (instrument in instrumentlist){
  rm(daterange_check)
  daterange_check <- c()
  transactionsInstrument <- getTxns(Portfolio=portfolioname,Symbol=instrument)

  # Create a list of transactions to check
  for (i in
```

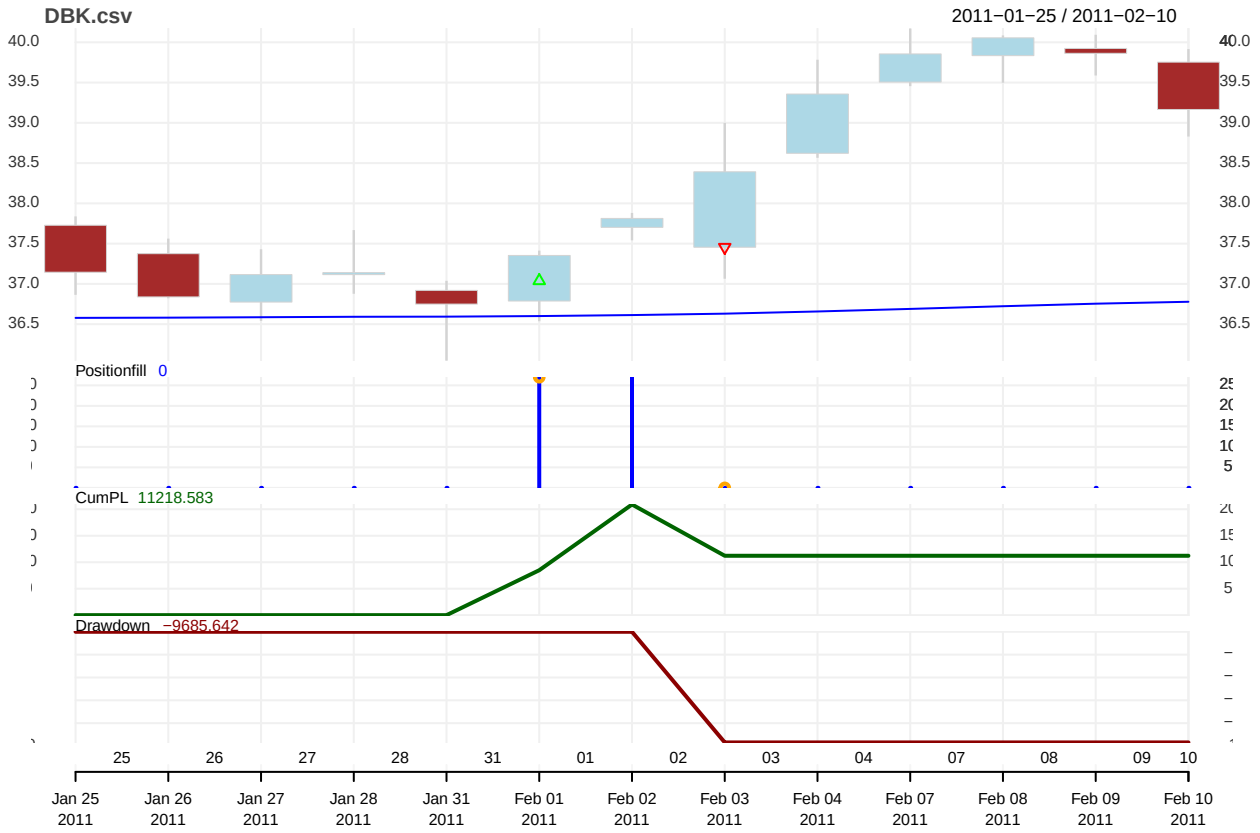
```
      c(2,6, (nrow(transactionsInstrument)-7),(nrow(transactionsInstrument)-5))) {
    from <- as.Date(index(transactionsInstrument[i,1]))-7
    to <- as.Date(index(transactionsInstrument[i+1,1]))+7
    daterange_check <- c(daterange_check, paste(from, "::", to, sep = ""))
  }
  # Plot the transactions and check them manually
  for (daterange_check_i in daterange_check){
    print(chart.Posn(portfolioname,
                     Symbol=instrument,
                     type='candlesticks',
                     theme=myTheme,
                     subset=daterange_check_i,
                     TA=addEMAString))
  }
}
```

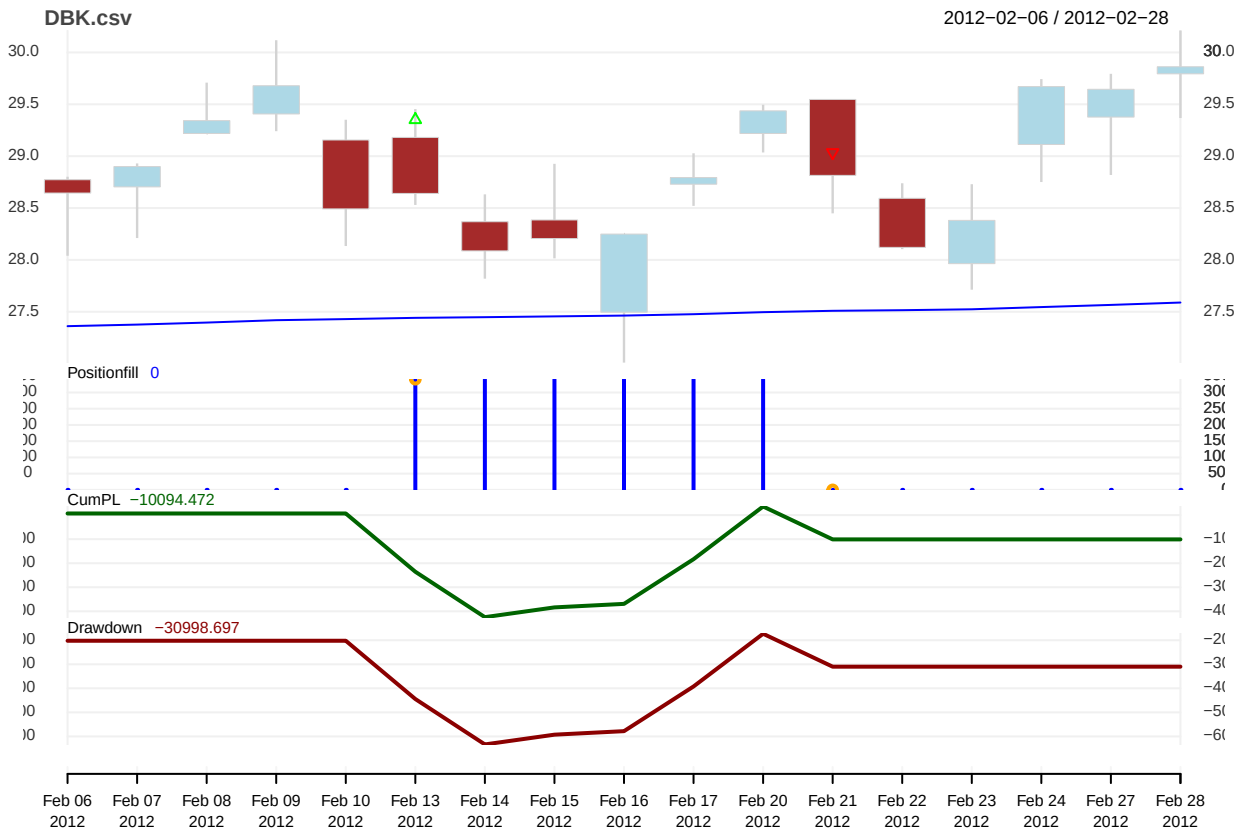


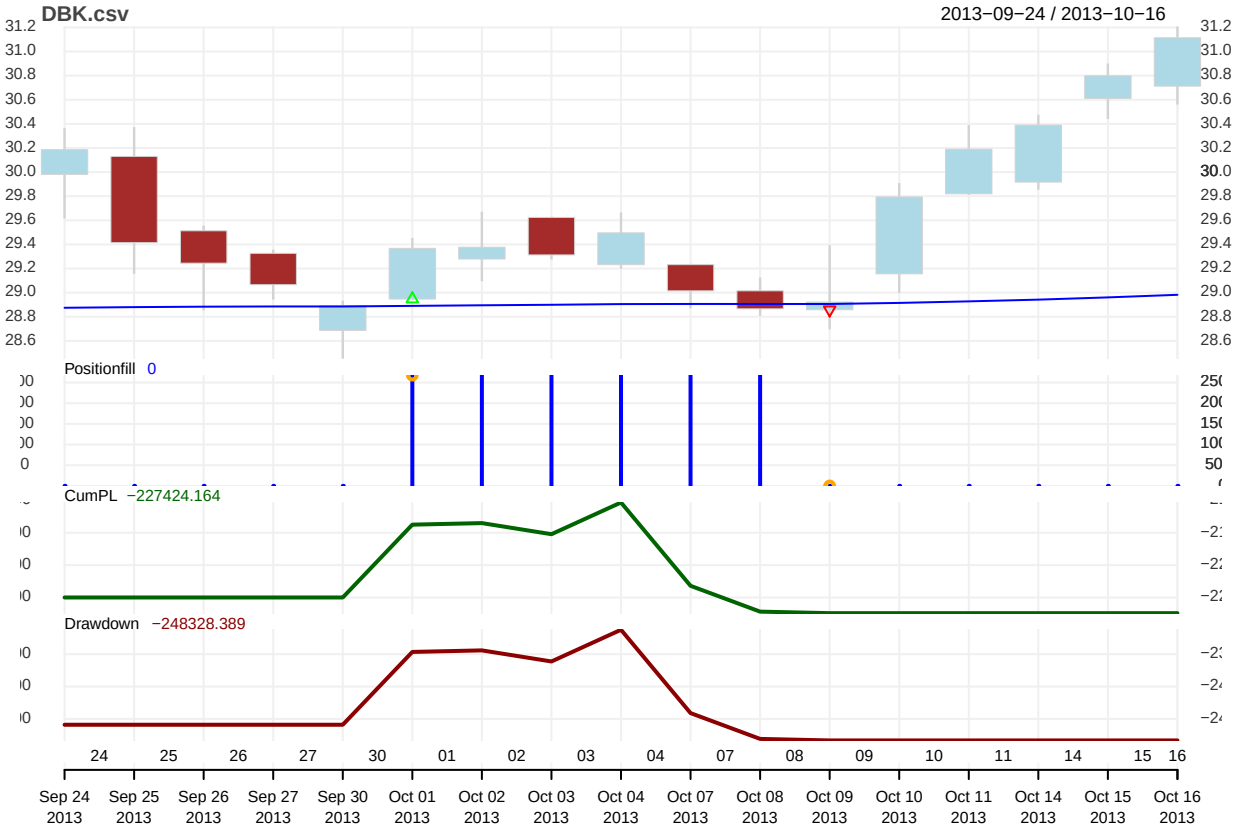

SAP.csv
2011−02−18 / 2011−04−01
Positionfill 0
CumPL 46404.335
Drawdown −28163.885
W07
W08
W09
W10
W11
W12
W13
Feb 18 2011
Feb 28 2011
Mar 07 2011
Mar 14 2011
Mar 21 2011
Mar 28 2011
Apr 01 2011


SAP.csv
2014−01−08 / 2014−01−31
Positionfill 0
CumPL −190344.49
Drawdown −264912.71
Jan 08 2014
Jan 09 2014
Jan 10 2014
Jan 13 2014
Jan 14 2014
Jan 15 2014
Jan 16 2014
Jan 17 2014
Jan 20 2014
Jan 21 2014
Jan 22 2014
Jan 23 2014
Jan 24 2014
Jan 27 2014
Jan 28 2014
Jan 29 2014
Jan 30 2014
Jan 31 2014

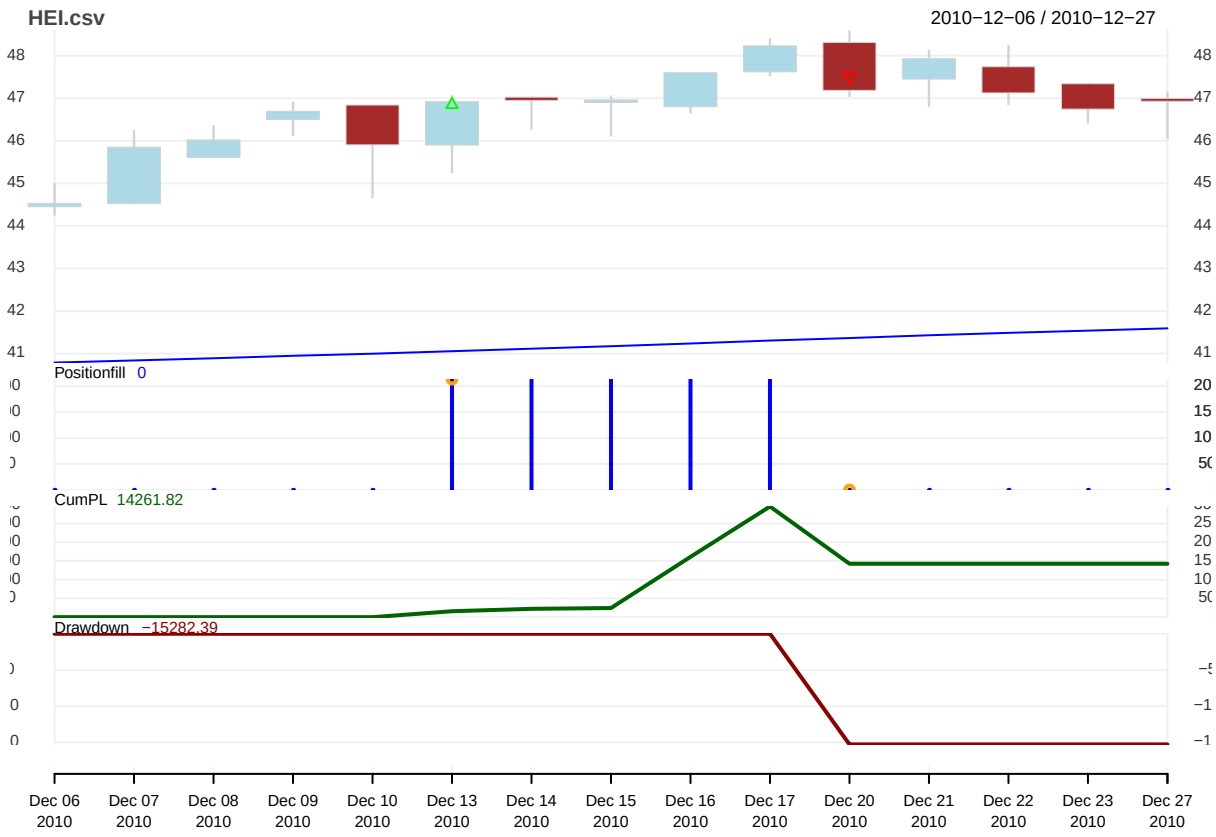

SAP.csv
2014−02−21 / 2014−03−10
58.5
58.0
57.5
57.0
56.5
56.0
55.5
Positionfill 0
CumPL −204532.91
Drawdown −279101.13
21
24
25
26
27
28
03
04
05
06
07
10
Feb 21 2014
Feb 24 2014
Feb 25 2014
Feb 26 2014
Feb 27 2014
Feb 28 2014
Mar 03 2014
Mar 04 2014
Mar 05 2014
Mar 06 2014
Mar 07 2014
Mar 10 2014

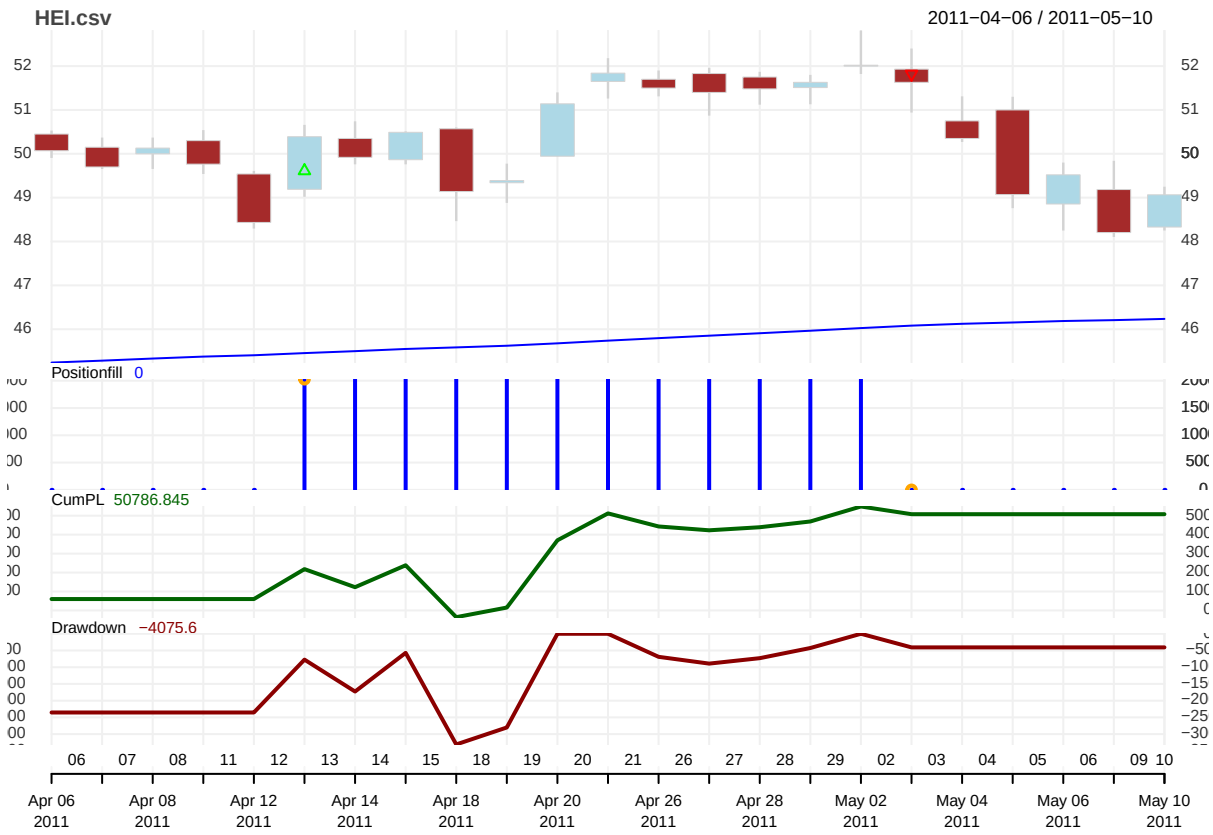

DBK.csv
2011−01−25 / 2011−02−10
40.0
39.5
39.0
38.5
38.0
37.5
37.0
36.5
Positionfill 0
CumPL 11218.583
Drawdown −9685.642
25
26
27
28
31
01
02
03
04
07
08
09
10
Jan 25 2011
Jan 26 2011
Jan 27 2011
Jan 28 2011
Jan 31 2011
Feb 01 2011
Feb 02 2011
Feb 03 2011
Feb 04 2011
Feb 07 2011
Feb 08 2011
Feb 09 2011
Feb 10 2011


DBK.csv
2012−02−06 / 2012−02−28
30.0
29.5
29.0
28.5
28.0
27.5
Positionfill 0
CumPL −10094.472
Drawdown −30998.697
Feb 06 2012
Feb 07 2012
Feb 08 2012
Feb 09 2012
Feb 10 2012
Feb 13 2012
Feb 14 2012
Feb 15 2012
Feb 16 2012
Feb 17 2012
Feb 20 2012
Feb 21 2012
Feb 22 2012
Feb 23 2012
Feb 24 2012
Feb 27 2012
Feb 28 2012

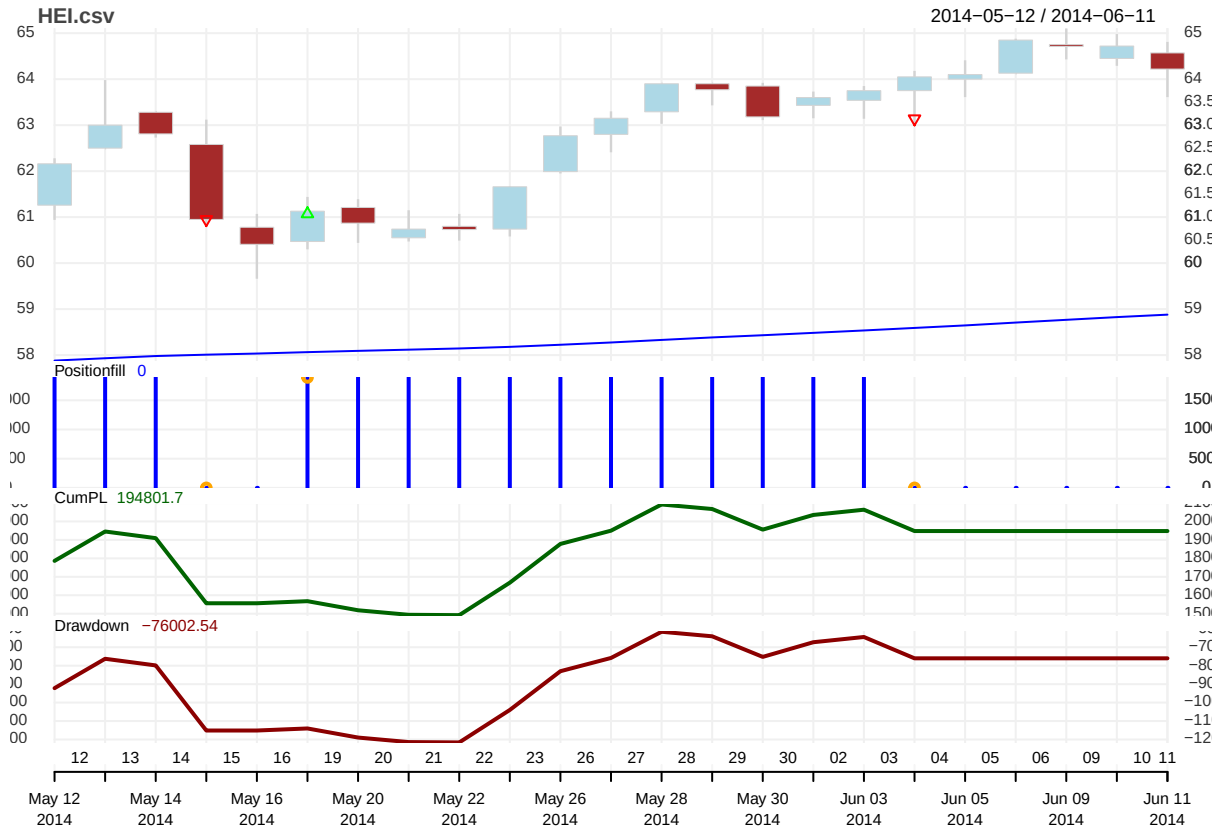

DBK.csv
2013−09−24 / 2013−10−16
Positionfill 0
CumPL −227424.164
Drawdown −248328.389
24 25 26 27 30 01 02 03 04 07 08 09 10 11 14 15 16
Sep 24 2013
Sep 25 2013
Sep 26 2013
Sep 27 2013
Sep 30 2013
Oct 01 2013
Oct 02 2013
Oct 03 2013
Oct 04 2013
Oct 07 2013
Oct 08 2013
Oct 09 2013
Oct 10 2013
Oct 11 2013
Oct 14 2013
Oct 15 2013
Oct 16 2013


DBK.csv
2013−10−15 / 2013−11−20
31.5
31.0
30.5
30.0
29.5
29.0
Positionfill 0
CumPL −285257.218
Drawdown −306161.443
W41
W42
W43
W44
W45
W46
Oct 15 2013
Oct 17 2013
Oct 21 2013
Oct 23 2013
Oct 25 2013
Oct 29 2013
Oct 31 2013
Nov 04 2013
Nov 06 2013
Nov 08 2013
Nov 12 2013
Nov 14 2013
Nov 18 2013
Nov 20 2013


HEI.csv
2010−12−06 / 2010−12−27
Positionfill 0
CumPL 14261.82
Drawdown −15282.39
Dec 06 2010
Dec 07 2010
Dec 08 2010
Dec 09 2010
Dec 10 2010
Dec 13 2010
Dec 14 2010
Dec 15 2010
Dec 16 2010
Dec 17 2010
Dec 20 2010
Dec 21 2010
Dec 22 2010
Dec 23 2010
Dec 27 2010


HEI.csv
2011−04−06 / 2011−05−10
Positionfill 0
CumPL 50786.845
Drawdown −4075.6
Apr 06 2011
Apr 08 2011
Apr 12 2011
Apr 14 2011
Apr 18 2011
Apr 20 2011
Apr 26 2011
Apr 28 2011
May 02 2011
May 04 2011
May 06 2011
May 10 2011


HEI.csv
2014−04−07 / 2014−05−22
Positionfill 18923
CumPL 149217.27
Drawdown −121586.97
W14
W15
W16
W17
W18
W19
W20
Apr 07 2014
Apr 14 2014
Apr 22 2014
Apr 28 2014
May 05 2014
May 12 2014
May 19 2014
May 22 2014

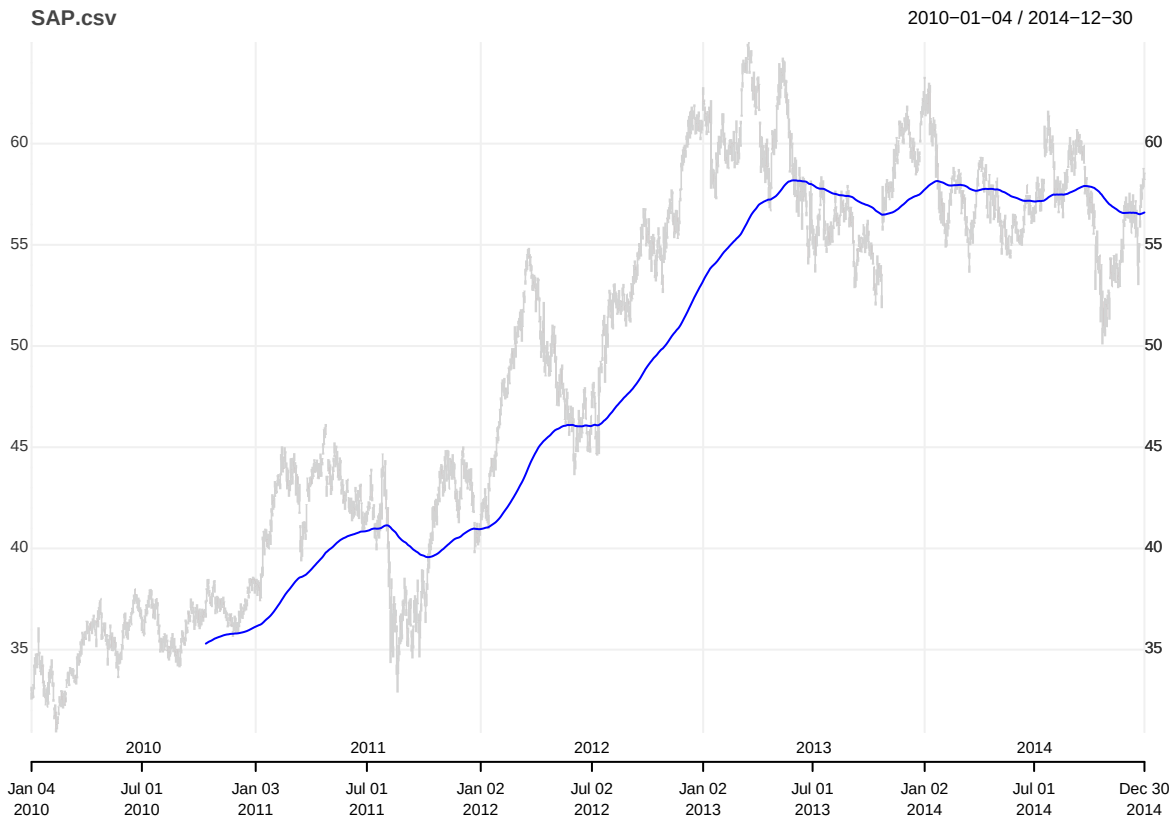


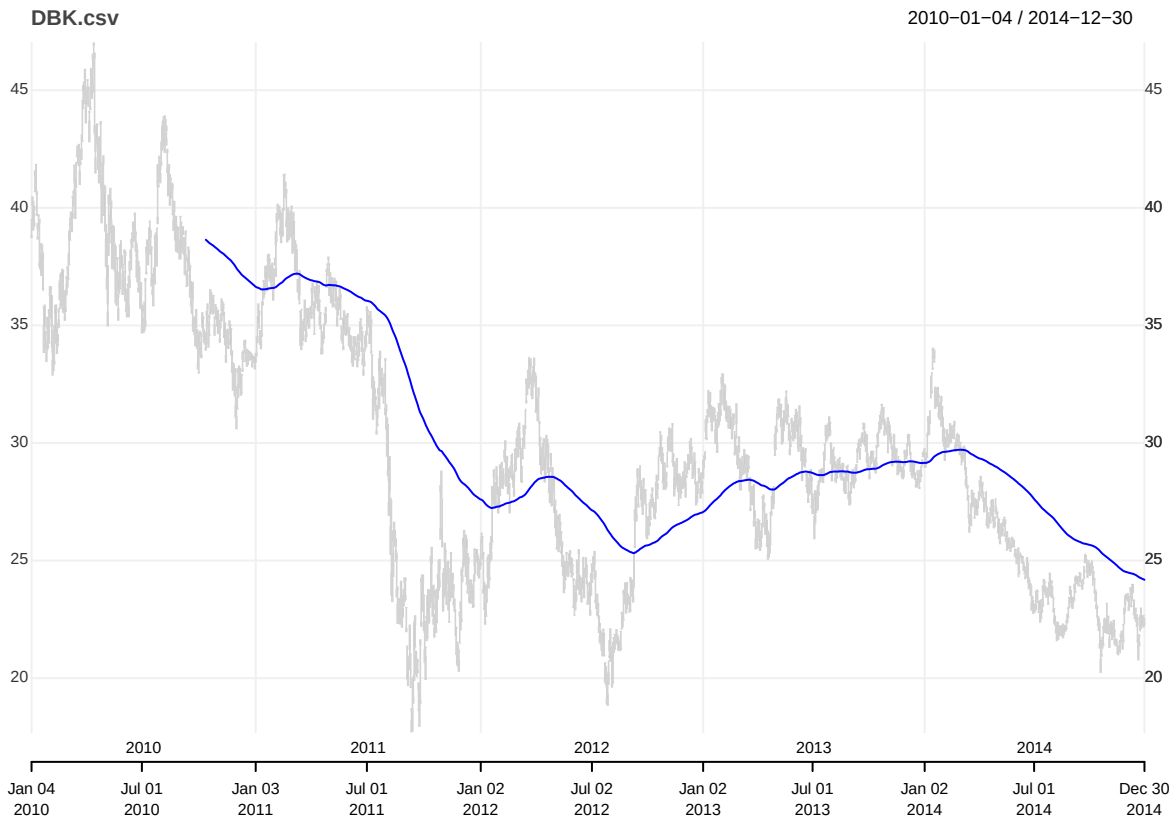
Part C: Analysis and Reporting

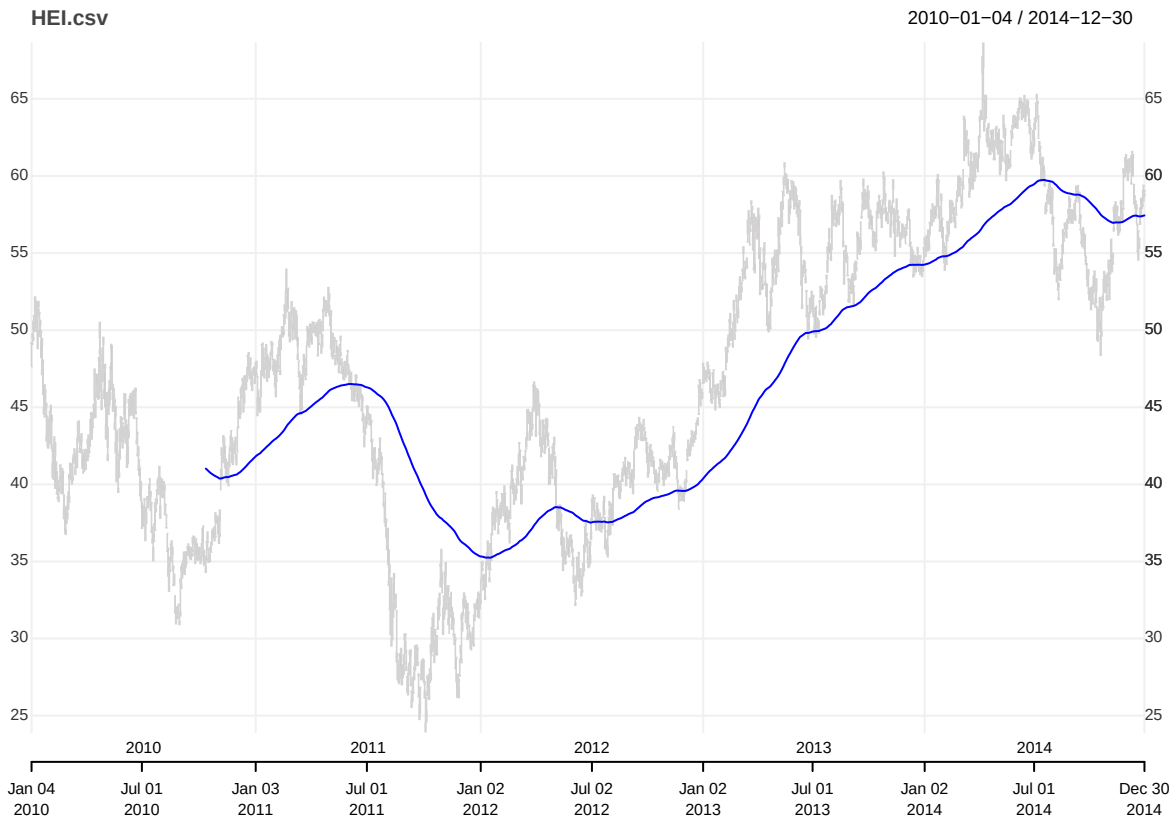
Step 1: Visualize original data

Plot of the instrument with the EMA line which indicates the general trend of the stock exponentially smoothed for the last 200 days. Moreover, the tradevolume is added below the graph. This plots can be used to get a first impression of the long term trend of the instrument and to see which general behaviour of the price can be observed

```
for (instrument in instrumentlist){
  chart <- get(instrument)
  print(chart_Series(x=chart,name=instrument,
                     theme=myTheme,
                     subset=daterange,
                     TA="add_TA((chart$EMA), on=1,type='l',col='blue')"))
}
```


SAP.csv
2010−01−04 / 2014−12−30
60
55
50
45
40
35
2010
2011
2012
2013
2014
Jan 04 2010
Jul 01 2010
Jan 03 2011
Jul 01 2011
Jan 02 2012
Jul 02 2012
Jan 02 2013
Jul 01 2013
Jan 02 2014
Jul 01 2014
Dec 30 2014


DBK.csv
2010−01−04 / 2014−12−30
45
40
35
30
25
20
2010
2011
2012
2013
2014
Jan 04 2010
Jul 01 2010
Jan 03 2011
Jul 01 2011
Jan 02 2012
Jul 02 2012
Jan 02 2013
Jul 01 2013
Jan 02 2014
Jul 01 2014
Dec 30 2014



Step 2: All transactions performed by the trading system

The following table can be used to get a better overview of the transactions performed and the exact details like the quantity, price and value per transaction.

```
for (instrument in instrumentlist){
  print(sprintf("Transactions for the instrument: %s", instrument))
  txns <- getTxns(Portfolio = portfolioname, Symbol = instrument)
  txns.pr <- txns[,c(1,2,4,6)]
  colnames(txns.pr) <- c("Quantity","Price","Value","Net realized Profit")
  print(txns.pr)
  writeLines("")
}
```

```
## [1] "Transactions for the instrument: SAP.csv"
##            Quantity  Price      Value Net realized Profit
## 1999-12-31        0  0.000        0.0               0.000
## 2010-11-25    27658 36.155   999975.0             -20.000
## 2010-11-29   -27658 36.110  -998730.4           -1264.610
## 2010-12-21    26250 38.045   998681.2             -20.000
## 2011-01-20   -26250 40.250 -1056562.5           57861.250
## 2011-02-25    24367 43.360  1056553.1             -20.000
## 2011-03-25   -24367 42.945 -1046440.8          -10132.305
## 2011-04-08    23685 44.180  1046403.3             -20.000
## 2011-04-11   -23685 43.925 -1040363.6           -6059.675
## 2011-04-13    23619 44.045  1040298.9             -20.000
## 2011-04-18   -23619 43.545 -1028489.4          -11829.500
```

```
## 2011-05-06    23659 43.470  1028456.7            -20.000
## 2011-05-12   -23659 44.380 -1049986.4          21509.690
## 2011-05-24    24243 43.310  1049964.3            -20.000
## 2011-06-01   -24243 42.775 -1036994.3         -12990.005
## 2011-06-09    24662 42.045  1036913.8            -20.000
## 2011-06-15   -24662 42.270 -1042462.7           5528.950
## 2011-07-13    24794 42.045  1042463.7            -20.000
## 2011-07-14   -24794 40.870 -1013330.8         -29152.950
## 2011-08-04    24137 41.980  1013271.3            -20.000
## 2011-08-05   -24137 38.995  -941222.3         -72068.945
## 2011-11-02    21731 43.310   941169.6            -20.000
## 2011-11-09   -21731 44.000  -956164.0          14974.390
## 2011-12-15    21910 43.640   956152.4            -20.000
## 2011-12-22   -21910 40.110  -878810.1         -77362.300
## 2012-02-28    17470 50.300   878741.0            -20.000
## 2012-03-27   -17470 53.090  -927482.3          48721.300
## 2012-03-30    17565 52.800   927432.0            -20.000
## 2012-04-13   -17565 49.550  -870345.8         -57106.250
## 2012-04-24    17672 49.250   870346.0            -20.000
## 2012-05-23   -17672 47.000  -830584.0         -39782.000
## 2012-07-24    16594 50.050   830529.7            -20.000
## 2012-08-07   -16594 52.270  -867368.4          36818.680
## 2012-08-13    16599 52.250   867297.8            -20.000
## 2012-08-28   -16599 51.680  -857836.3          -9481.430
## 2012-10-01    15354 55.870   857828.0            -20.000
## 2012-10-17   -15354 55.040  -845084.2         -12763.820
## 2012-10-24    15580 54.240   845059.2            -20.000
## 2012-11-05   -15580 56.560  -881204.8          36125.600
## 2012-11-16    15645 56.320   881126.4            -20.000
## 2012-11-30   -15645 59.840  -936196.8          55050.400
## 2012-12-19    15272 61.300   936173.6            -20.000
## 2013-01-23   -15272 59.200  -904102.4         -32091.200
## 2013-02-12    15151 59.670   904060.2            -20.000
## 2013-02-14   -15151 59.730  -904969.2            889.060
## 2013-02-19    15082 60.000   904920.0            -20.000
## 2013-02-26   -15082 59.700  -900395.4          -4544.600
## 2013-03-25    14239 63.230   900332.0            -20.000
## 2013-04-17   -14239 58.970  -839673.8         -60678.140
## 2013-11-06    14447 58.120   839659.6            -20.000
## 2013-11-08   -14447 58.530  -845582.9           5903.270
## 2013-11-22    13955 60.590   845533.5            -20.000
## 2013-11-29   -13955 61.340  -855999.7          10446.250
## 2014-01-08    13696 62.500   856000.0            -20.000
## 2014-01-09   -13696 61.940  -848330.2          -7689.760
## 2014-01-15    13984 60.660   848269.4            -20.000
## 2014-01-24   -13984 57.900  -809673.6         -38615.840
## 2014-02-28    13871 58.370   809650.3            -20.000
## 2014-03-03   -13871 57.350  -795501.8         -14168.420
## 2014-09-04    13407 59.330   795437.3            -20.000
## 2014-09-22   -13407 57.760  -774388.3         -21068.990
## 2014-12-04    13559 57.110   774354.5            -20.000
## 2014-12-05   -13559 55.880  -757676.9         -16697.570
##
## [1] "Transactions for the instrument: DBK.csv"
```

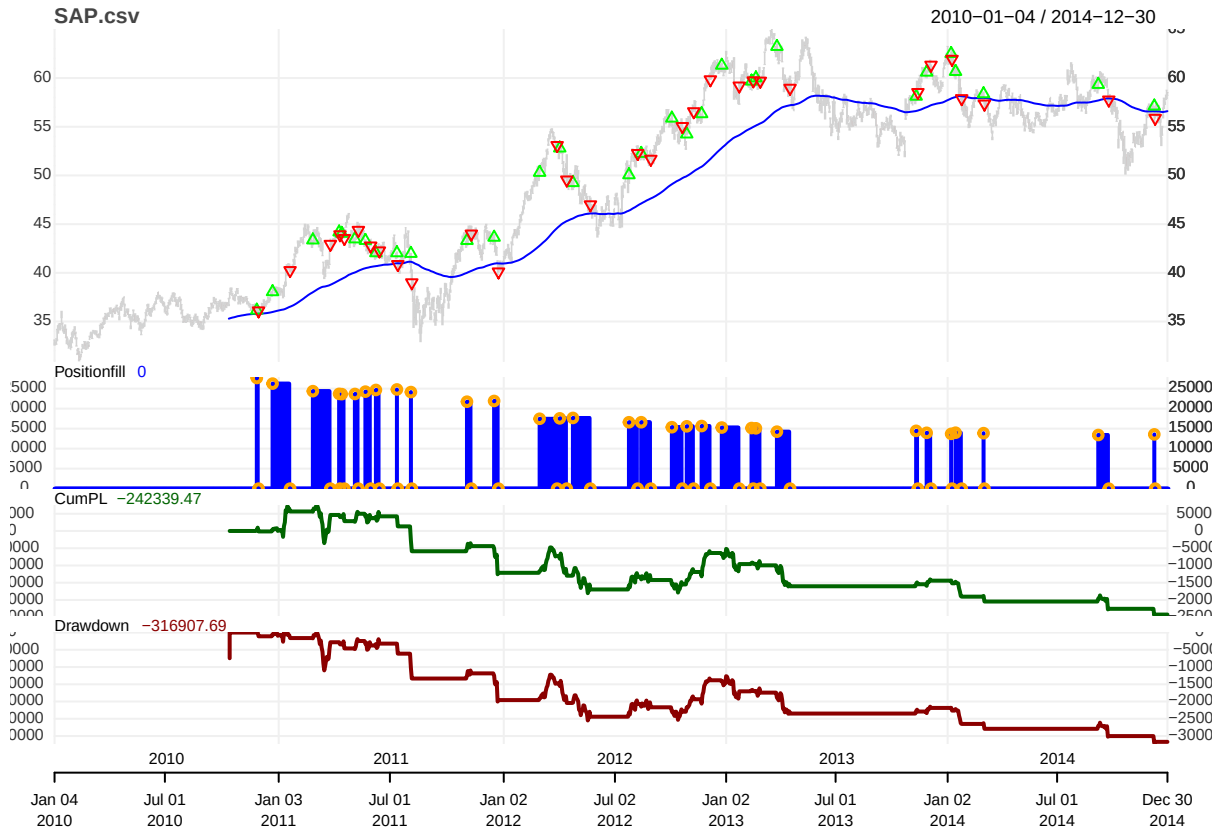
```
##            Quantity  Price      Value Net realized Profit
## 1999-12-31        0  0.000        0.0               0.000
## 2011-02-01    26999 37.038   999989.0             -20.000
## 2011-02-03   -26999 37.455 -1011247.5           11238.583
## 2011-02-25    25720 39.315  1011181.8             -20.000
## 2011-02-28   -25720 38.907 -1000688.0          -10513.760
## 2012-02-13    34093 29.351  1000663.6             -20.000
## 2012-02-21   -34093 29.036  -989924.3          -10759.295
## 2012-04-11    33239 29.781   989890.7             -20.000
## 2012-04-13   -33239 28.900  -960607.1          -29303.559
## 2012-04-17    32689 29.385   960566.3             -20.000
## 2012-04-24   -32689 28.317  -925654.4          -34931.852
## 2012-09-14    32575 28.415   925618.6             -20.000
## 2012-09-18   -32575 28.279  -921188.4           -4450.200
## 2012-11-06    30585 30.118   921159.0             -20.000
## 2012-11-07   -30585 29.841  -912687.0           -8492.045
## 2012-11-19    31961 28.555   912646.4             -20.000
## 2012-11-20   -31961 27.921  -892383.1          -20283.274
## 2012-11-29    31147 28.649   892330.4             -20.000
## 2012-12-06   -31147 29.539  -920051.2           27700.830
## 2012-12-11    31532 29.177   920009.2             -20.000
## 2013-01-08   -31532 30.169  -951288.9           31259.744
## 2013-03-14    32617 29.164   951242.2             -20.000
## 2013-03-20   -32617 27.666  -902381.9          -48880.266
## 2013-05-22    28549 31.607   902348.2             -20.000
## 2013-05-23   -28549 30.518  -871258.4          -31109.861
## 2013-05-30    28272 30.816   871230.0             -20.000
## 2013-06-05   -28272 30.867  -872671.8            1421.872
## 2013-06-07    28310 30.824   872627.4             -20.000
## 2013-06-21   -28310 28.202  -798398.6          -74248.820
## 2013-08-01    27382 29.156   798349.6             -20.000
## 2013-08-07   -27382 28.304  -775020.1          -23349.464
## 2013-10-01    26773 28.947   774998.0             -20.000
## 2013-10-09   -26773 28.858  -772615.2           -2402.797
## 2013-10-22    24551 31.467   772546.3             -20.000
## 2013-11-13   -24551 29.113  -714753.3          -57813.054
## 2014-02-05    23859 29.956   714720.2             -20.000
## 2014-02-13   -23859 30.249  -721710.9            6970.687
## 2014-02-21    23925 30.164   721673.7             -20.000
## 2014-03-03   -23925 29.368  -702629.4          -19064.300
##
## [1] "Transactions for the instrument: HEI.csv"
##            Quantity  Price      Value Net realized Profit
## 1999-12-31        0  0.000        0.0               0.000
## 2010-12-13    21346 46.845   999953.4             -20.000
## 2010-12-20   -21346 47.515 -1014255.2           14281.820
## 2011-01-03    21323 47.565  1014228.5             -20.000
## 2011-01-04   -21323 47.180 -1006019.1           -8229.355
## 2011-04-13    20278 49.610  1005991.6             -20.000
## 2011-05-03   -20278 51.820 -1050806.0           44794.380
## 2011-05-13    21710 48.400  1050764.0             -20.000
## 2011-05-16   -21710 48.295 -1048484.5           -2299.550
## 2011-05-19    21664 48.395  1048429.3             -20.000
## 2011-06-08   -21664 46.250 -1001960.0          -46489.280
```

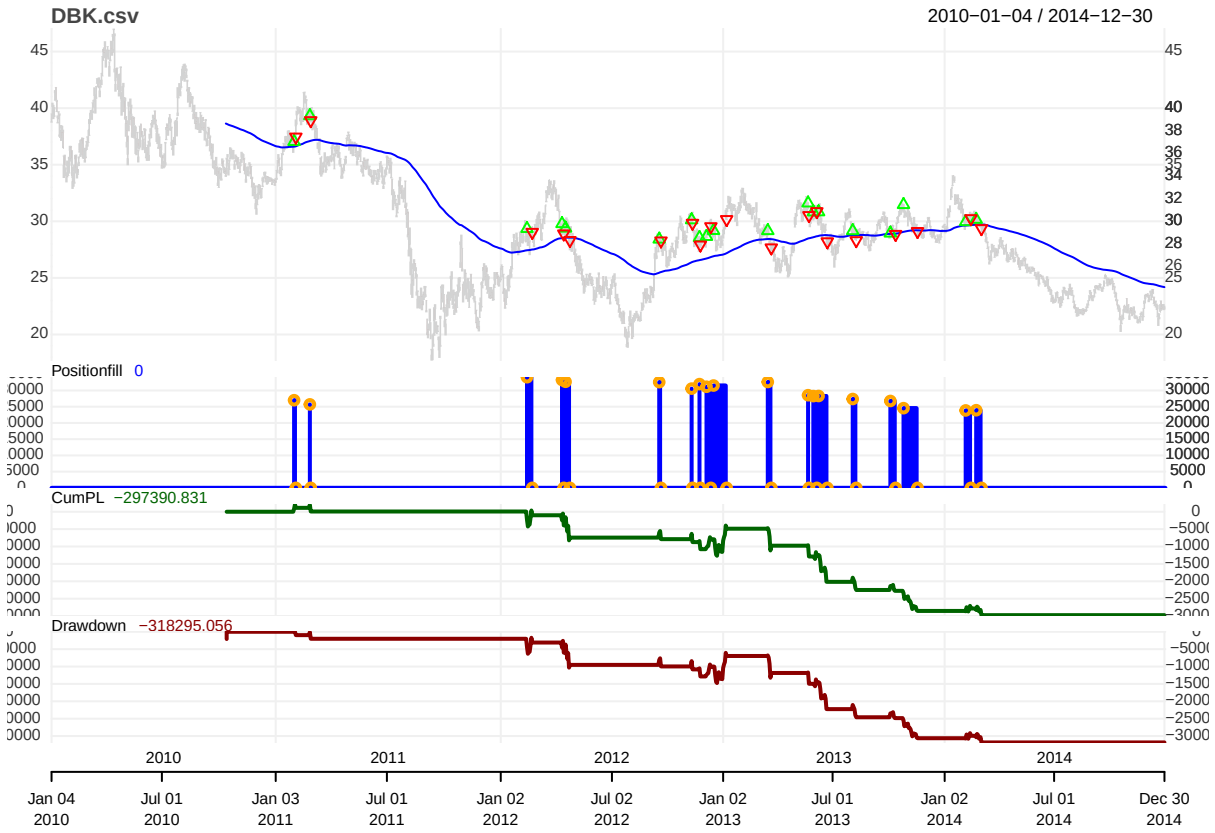
```
## 2012-01-31    26315 38.075  1001943.6            -20.000
## 2012-02-28   -26315 40.250 -1059178.8          57215.125
## 2012-03-21    23700 44.690  1059153.0            -20.000
## 2012-04-02   -23700 44.480 -1054176.0          -4997.000
## 2012-04-11    24549 42.940  1054134.1            -20.000
## 2012-04-13   -24549 42.880 -1052661.1          -1492.940
## 2012-08-01    27376 38.450  1052607.2            -20.000
## 2012-08-02   -27376 37.800 -1034812.8         -17814.400
## 2012-09-03    25409 40.725  1034781.5            -20.000
## 2012-09-11   -25409 42.900 -1090046.1          55244.575
## 2012-09-14    24733 44.070  1089983.3            -20.000
## 2012-09-17   -24733 43.660 -1079842.8         -10160.530
## 2012-09-19    24772 43.590  1079811.5            -20.000
## 2012-09-20   -24772 43.035 -1066063.0         -13768.460
## 2012-10-01    25375 42.010  1066003.8            -20.000
## 2012-10-19   -25375 41.055 -1041770.6         -24253.125
## 2012-11-08    23822 43.730  1041736.1            -20.000
## 2012-11-09   -23822 42.390 -1009814.6         -31941.480
## 2013-01-09    21376 47.240  1009802.2            -20.000
## 2013-02-04   -21376 46.670  -997617.9         -12204.320
## 2013-03-11    18432 54.120   997539.8            -20.000
## 2013-03-21   -18432 56.970 -1050071.0          52511.200
## 2013-03-25    18335 57.270  1050045.4            -20.000
## 2013-03-28   -18335 56.580 -1037394.3         -12671.150
## 2013-04-02    18190 57.200  1040468.0            -20.000
## 2013-04-04   -18190 56.660 -1030645.4          -9842.600
## 2013-04-19    20230 50.790  1027481.7            -20.000
## 2013-05-13   -20230 56.700 -1147041.0         119539.300
## 2013-05-24    19514 58.780  1147032.9            -20.000
## 2013-05-29   -19514 58.610 -1143715.5          -3337.380
## 2013-07-04    22442 50.960  1143644.3            -20.000
## 2013-07-05   -22442 50.350 -1129954.7         -13709.620
## 2013-07-31    20396 55.400  1129938.4            -20.000
## 2013-08-06   -20396 57.200 -1166651.2          36692.800
## 2013-08-08    20268 57.560  1166626.1            -20.000
## 2013-08-12   -20268 56.880 -1152843.8         -13802.240
## 2013-09-02    21661 53.220  1152798.4            -20.000
## 2013-09-03   -21661 52.920 -1146300.1          -6518.300
## 2013-09-23    19491 58.810  1146265.7            -20.000
## 2013-09-27   -19491 58.050 -1131452.6         -14833.160
## 2013-10-01    19618 57.670  1131370.1            -20.000
## 2013-10-16   -19618 58.130 -1140394.3           9004.280
## 2013-11-19    20244 56.330  1140344.5            -20.000
## 2013-12-02   -20244 57.180 -1157551.9          17187.400
## 2014-02-19    19975 57.950  1157551.2            -20.000
## 2014-03-19   -19975 59.900 -1196502.5          38931.250
## 2014-04-14    18961 63.100  1196439.1            -20.000
## 2014-05-15   -18961 60.950 -1155672.9         -40786.150
## 2014-05-19    18923 61.070  1155627.6            -20.000
## 2014-06-04   -18923 63.140 -1194798.2          39150.610
## 2014-06-16    18489 64.620  1194759.2            -20.000
## 2014-06-20   -18489 64.500 -1192540.5          -2238.680
## 2014-11-21    20589 57.920  1192514.9            -20.000
## 2014-12-08   -20589 60.100 -1237398.9          44864.020
```

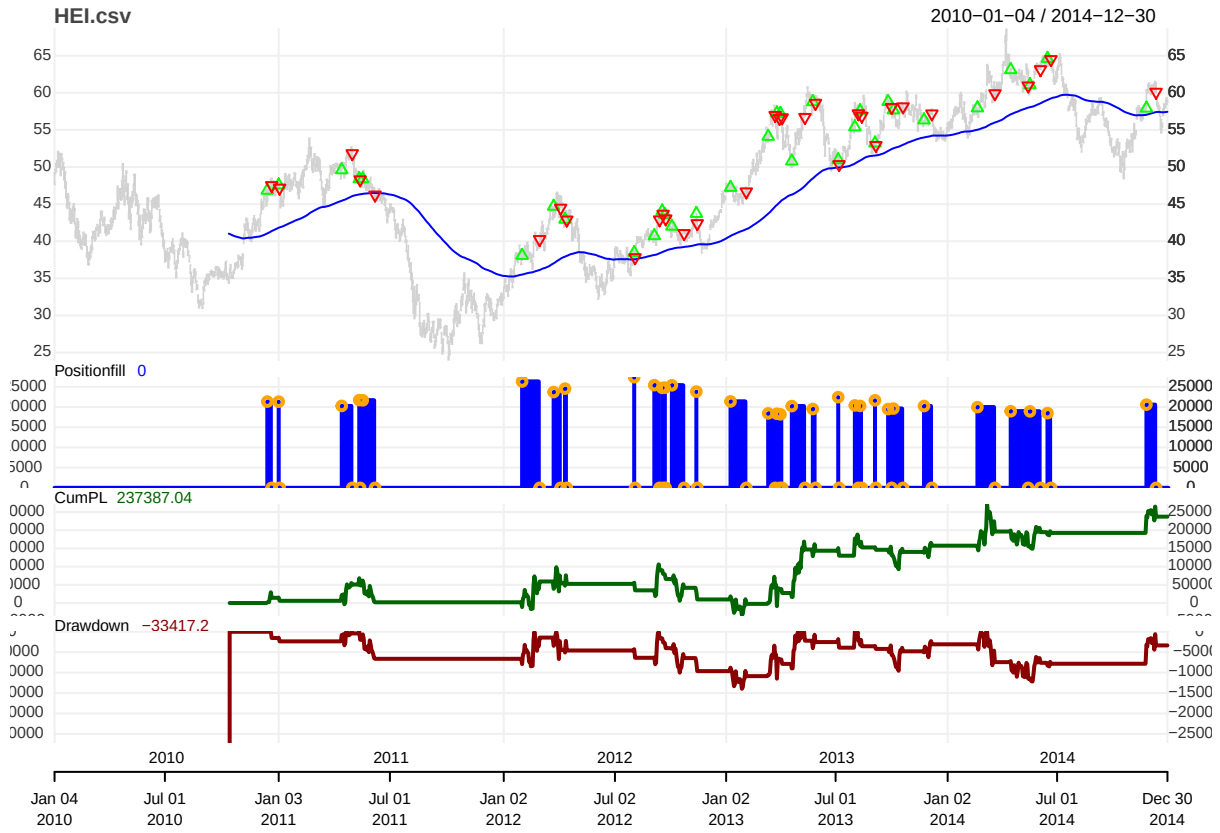
Step 3: Graph which visualize transactions

The following graphs show the combined view of the performance of the Smash Day trading system. It visualizes the trades (buy-transactions are visualized in green and sell-transactions are visualized in red). Moreover, the size of the blue squares indicates the size of the position (height) and the holding duration of the position (width). The green line shows the cummulative net profit curve, while the red line indicates the drawdown on each day compared to the last reached high.

```
# Plot graph with indicators for transaction
for (instrument in instrumentlist){
  chart.Posn(portfolioname,
             Symbol=instrument,
             type='candlesticks',
             theme=myTheme,
             subset=daterange,
             TA=addEMAString)
}
```




DBK.csv
2010−01−04 / 2014−12−30
Positionfill 0
CumPL −297390.831
Drawdown −318295.056
2010
2011
2012
2013
2014
Jan 04 2010
Jul 01 2010
Jan 03 2011
Jul 01 2011
Jan 02 2012
Jul 02 2012
Jan 02 2013
Jul 01 2013
Jan 02 2014
Jul 01 2014
Dec 30 2014



Step 4: Performance Statistics

The following table summarizes some important trading statistics for all instruments. This statistic is just an excerpt of the overall statistics which can be calculated. It is printed here as it can be assumed that they are interesting for investers to rate the trading strategy.

```
library(PerformanceAnalytics)
# Get returns for the account
rets <- PortfReturns(Account=accountname)
rownames(rets) <- NULL
tstats <- tradeStats(Portfolio=portfolioname, Symbols=instrumentlist)
for (i in 1:nrow(tstats)) {
  trades.tab <- cbind(
    c("Trades", "Win Percent", "Loss Percent","W/L Ratio"),
    c(tstats[i,"Num.Trades"],
      round(tstats[i,"Percent.Positive"],2),
      round(tstats[i,"Percent.Negative"],2),
      round((tstats[i,"Percent.Positive"]/tstats[i,"Percent.Negative"]),2)))
  trades1 <- trades.tab
  rownames(trades1) <- c("Trades",
                         "Win Percent",
                         "Loss Percent",
                         "W/L Ratio")
  trades1 <- trades1[,2]
  print(row.names(tstats[i,]))
  print(trades1)
```

```
  writeLines("")
  }
```

```
## [1] "SAP.csv"
##       Trades  Win Percent Loss Percent    W/L Ratio
##         "31"       "35.48"       "64.52"        "0.55"
##
## [1] "DBK.csv"
##       Trades  Win Percent Loss Percent    W/L Ratio
##         "19"       "26.32"       "73.68"        "0.36"
##
## [1] "HEI.csv"
##       Trades  Win Percent Loss Percent    W/L Ratio
##         "32"        "37.5"        "62.5"         "0.6"
```

Step 5: Calculate statistics of the Portfolio and all instruments in the portfolio

```
## Performance Metrics
```

```
##                                DBK        HEI         SAP
## Cumulative Return      -0.26579663 0.21682542 -0.22999224
## Annualized Return      -0.07011898 0.04725806 -0.05964256
## Annualized Sharp Ratio -0.95521079 0.33821921 -0.63281163
## Calmar Ratio           -0.24961855 0.35569424 -0.20792977
```

```
## Risk Metrics
```

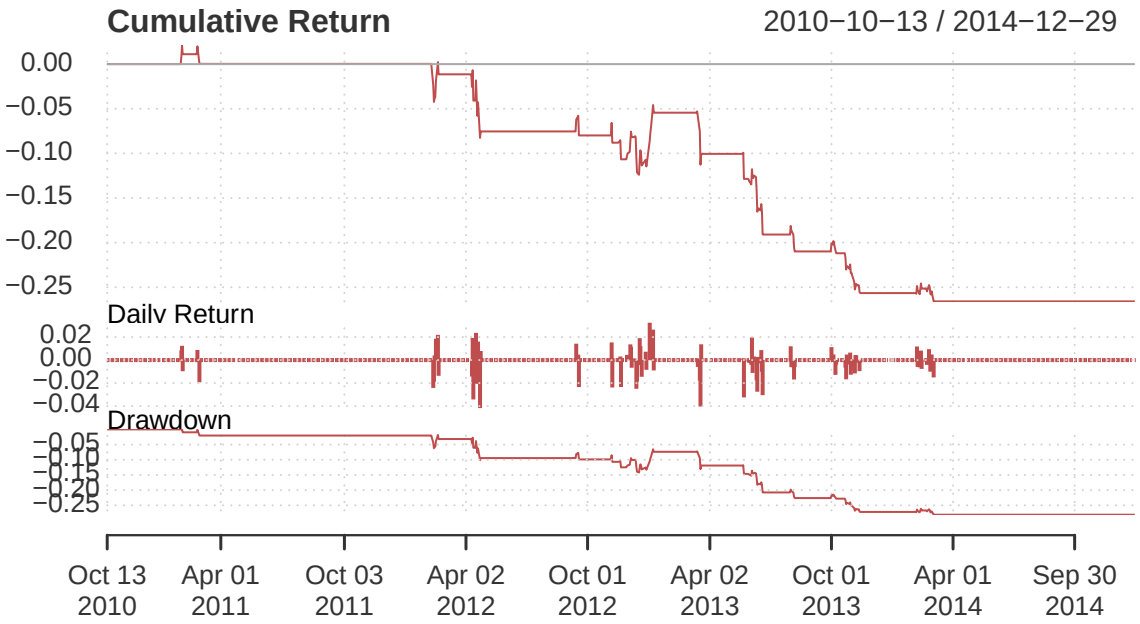
```
##                             DBK         HEI         SAP
## Annualized StdDev  0.073406817  0.13972612  0.09425010
## Max Drawdown       0.280904541  0.13286147  0.28683994
## Value-at-Risk     -0.002306355 -0.01314211 -0.00849015
## Conditional VaR   -0.014045335 -0.02102629 -0.01757923
```

Step 6: Visualize returns of the trading strategy for every instrument

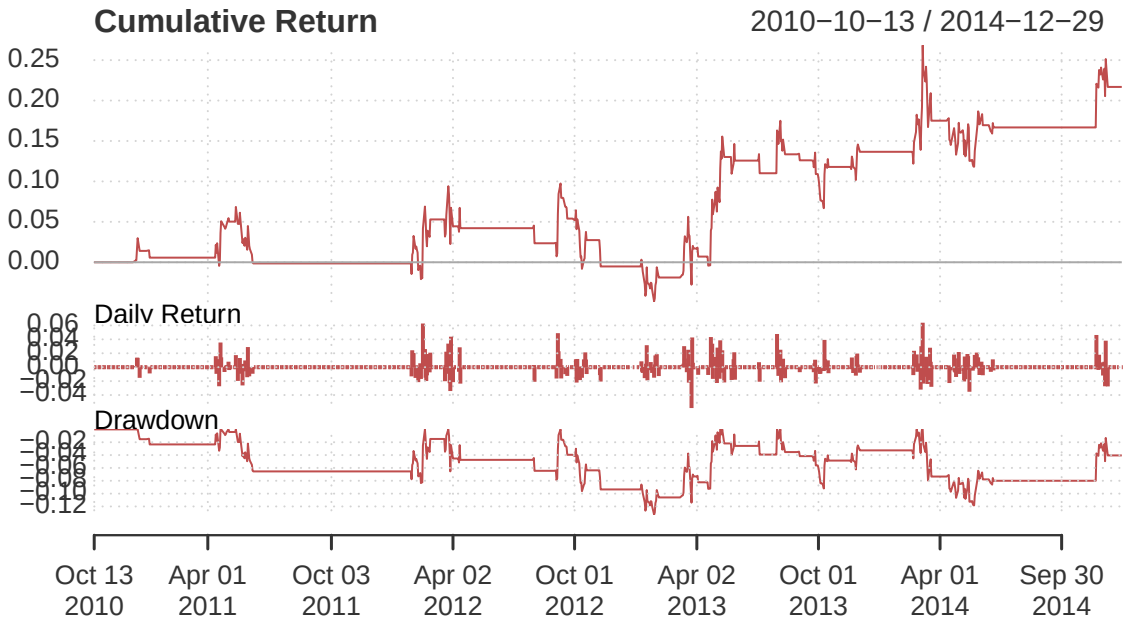
The following plots visualizes some important information about the returns of the strategy applied to the instruments on the instrumentlist. The cummulative return over the entire period, the daily returns and the drawdown for each instrument is illustrated.

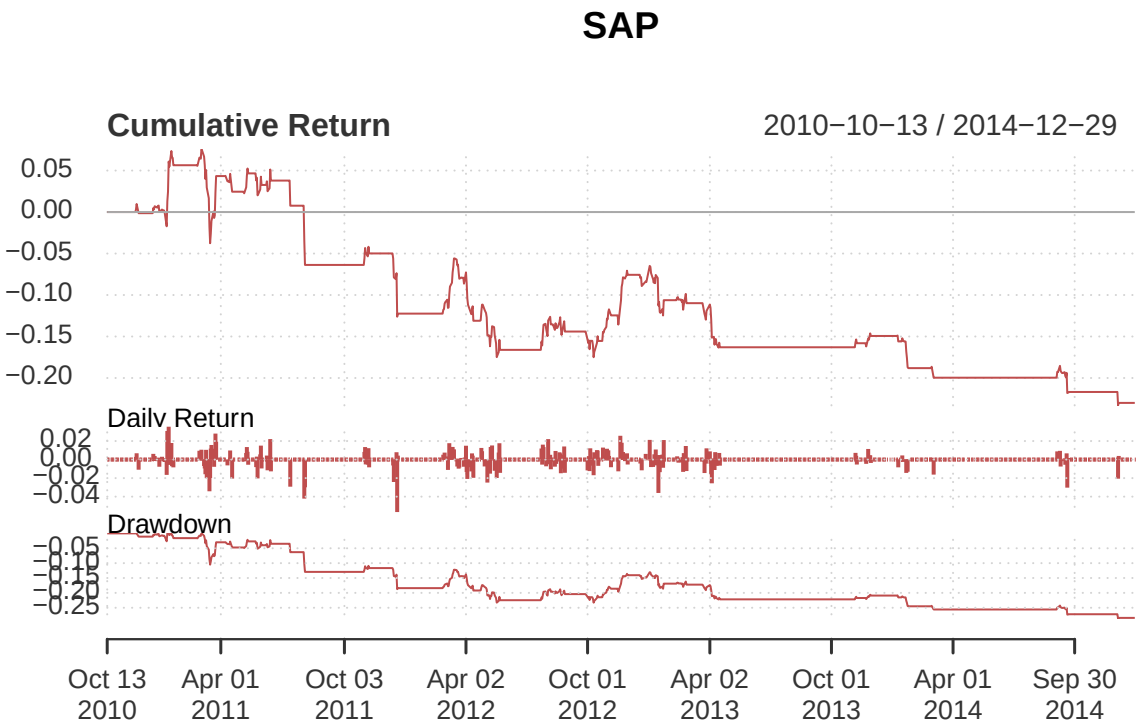
```
for (i in 1:ncol(rets)){
  charts.PerformanceSummary(rets[,i],colorset=rainbow12equal,lwd=1,main=substr(colnames(rets[,i]),1,3))
}
```

DBK


Cumulative Return
2010−10−13 / 2014−12−29
0.00
−0.05
−0.10
−0.15
−0.20
−0.25
Dailv Return
0.02
0.00
−0.02
−0.04
Drawdown
−0.05
−0.10
−0.15
−0.20
−0.25
Oct 13 2010
Apr 01 2011
Oct 03 2011
Apr 02 2012
Oct 01 2012
Apr 02 2013
Oct 01 2013
Apr 01 2014
Sep 30 2014

HEI


Cumulative Return
2010−10−13 / 2014−12−29
0.25
0.20
0.15
0.10
0.05
0.00
Daily Return
0.06
0.04
0.02
0.00
−0.02
−0.04
Drawdown
−0.02
−0.12
Oct 13
2010
Apr 01
2011
Oct 03
2011
Apr 02
2012
Oct 01
2012
Apr 02
2013
Oct 01
2013
Apr 01
2014
Sep 30
2014



Compare with Buy and Hold Strategy

In order to compare the trading strategy properly we need to define a benchmark against which we can measure the results. In this case a simple buy and hold strategy is used. At the first date of the trading period we place a buy order and sell our position at the last day of the selected period. In order to do this we create a new Portfolio and a new Account. As stocks from the DAX (German Stocks Market Index) are used in the strategy, the buy and hold strategy is performed on an DAX ETF.

Step 1: Perform the Buy and Hold Strategy

```
# Any objects, in case there was a buyhold
# portfolio initialized before are removed
suppressWarnings(try(rm(list=c("account.buyhold",
                               "portfolio.buyhold"),
                        pos=.blotter)))

# The Buy and hold instrument is loaded
LoadCourseFile(BuyHoldDirectory,BuyHoldInstrument,debugme=TRUE,dates=daterange)
# The Buy and hold instrument is initialized
stock(BuyHoldInstrument,currency="EUR")

BuyHoldSymbol<-get(BuyHoldInstrument)

# The portfolio and account "buyhold" is initialized
initPortf("buyhold",
          BuyHoldInstrument,
```

```
           initDate=initdate,
           currency="EUR")
initAcct("buyhold",
         portfolios="buyhold",
         initDate=initdate,
         initEq=startCapital,
         currency="EUR")

# The first date of the defined daterange is selected
currentdate <- first(time(BuyHoldSymbol))

# The close price at this date is selected
closeprice <- as.numeric(Cl(BuyHoldSymbol[currentdate,]))

# Calculate the unitsize we can buy with our startingcapital
unitsize <- as.numeric(trunc(startCapital/closeprice))

# Place the transaction for the instrument at the first date
addTxn("buyhold",
       Symbol=BuyHoldInstrument,
       TxnDate=currentdate,
       TxnPrice=closeprice,
       TxnQty=unitsize,
       TxnFees=transactionCost)

# Select the last date of the daterange period
lastdate <-last(time(BuyHoldSymbol))

# Select the price at the last date
lastprice <- as.numeric(Cl(BuyHoldSymbol[lastdate,]))

# Sell the position at the last date of the daterange
addTxn("buyhold",
       Symbol=BuyHoldInstrument,
       TxnDate=lastdate,
       TxnPrice=lastprice,
       TxnQty=-unitsize,
       TxnFees=transactionCost)

# update portfolio and account
updatePortf(Portfolio="buyhold")
updateAcct(name="buyhold")
updateEndEq(Account="buyhold")
```

Step 2: Visualize the Buy and Hold strategy

We can see that we hold the position from the first until the last date. The cumulative profits are visualized by the green line.

```
chart.Posn("buyhold",Symbol=BuyHoldInstrument, theme=myTheme)
```

Step 3: Compare the returns of the trading strategy with the buy and hold strategy

In order to compare the results of both strategies, we calculat the returns for the buy and hold strategy and combine them with the returns of the trading strategy which were calculated before.

```
rets.bh <- PortfReturns(Account='buyhold')
returns <- cbind(rets,rets.bh)
colnames(returns) <- substr(colnames(returns),1,3)
```

In order to compare the performance between the strategy investment in one of the stocks and the alternative buy and hold strategy of the DAX ETF, the following graphs visualize the relative performance between each of the stocks used and the DAX ETF.

```
table.Stats(returns)
```

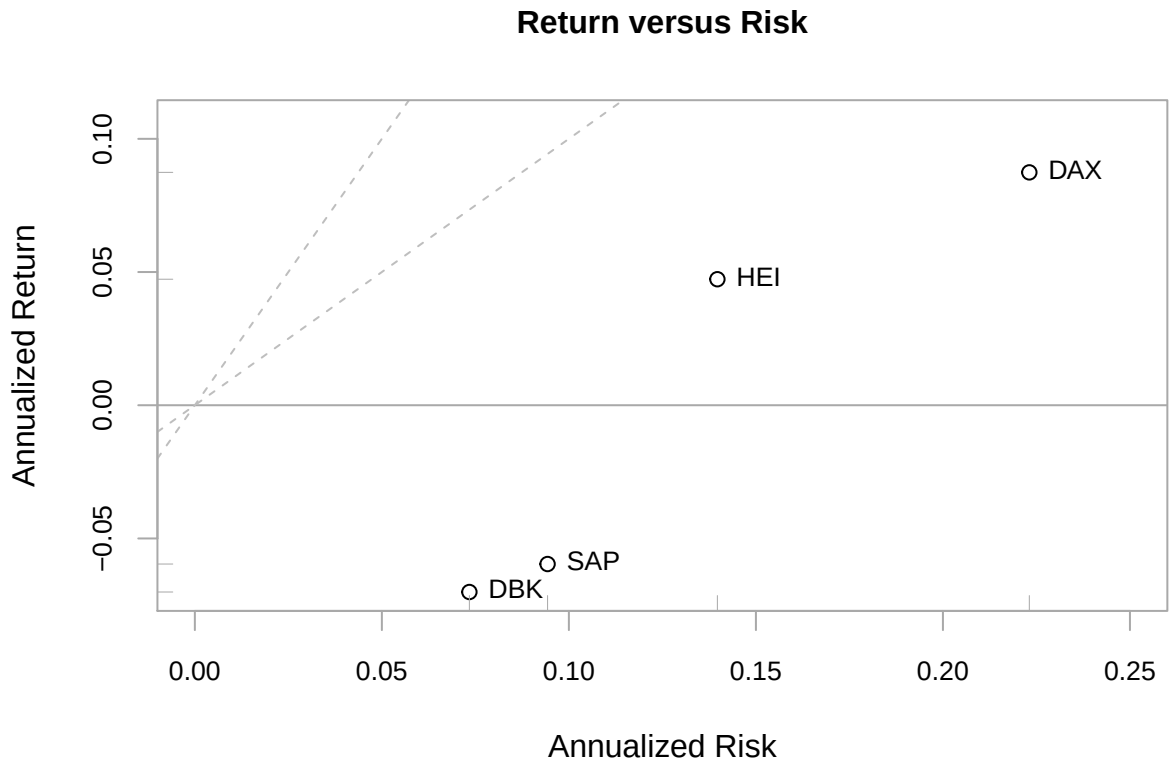
```
##                       DBK       HEI       SAP       DAX
## Observations    1071.0000 1071.0000 1071.0000 1272.0000
## NAs              201.0000  201.0000  201.0000    0.0000
## Minimum           -0.0417   -0.0587   -0.0566   -0.0535
## Quartile 1         0.0000    0.0000    0.0000   -0.0067
## Median             0.0000    0.0000    0.0000    0.0012
## Arithmetic Mean   -0.0003    0.0002   -0.0002    0.0004
## Geometric Mean    -0.0003    0.0002   -0.0002    0.0003
## Quartile 3         0.0000    0.0000    0.0000    0.0082
## Maximum            0.0325    0.0639    0.0354    0.0519
## SE Mean            0.0001    0.0003    0.0002    0.0004
## LCL Mean (0.95)   -0.0006   -0.0003   -0.0006   -0.0003
## UCL Mean (0.95)    0.0000    0.0007    0.0001    0.0012
```

```
## Variance           0.0000    0.0001    0.0000    0.0002
## Stdev              0.0046    0.0088    0.0059    0.0141
## Skewness          -2.4601    1.4016   -1.5151   -0.2034
## Kurtosis          28.6833   13.4157   18.2203    1.3153
```

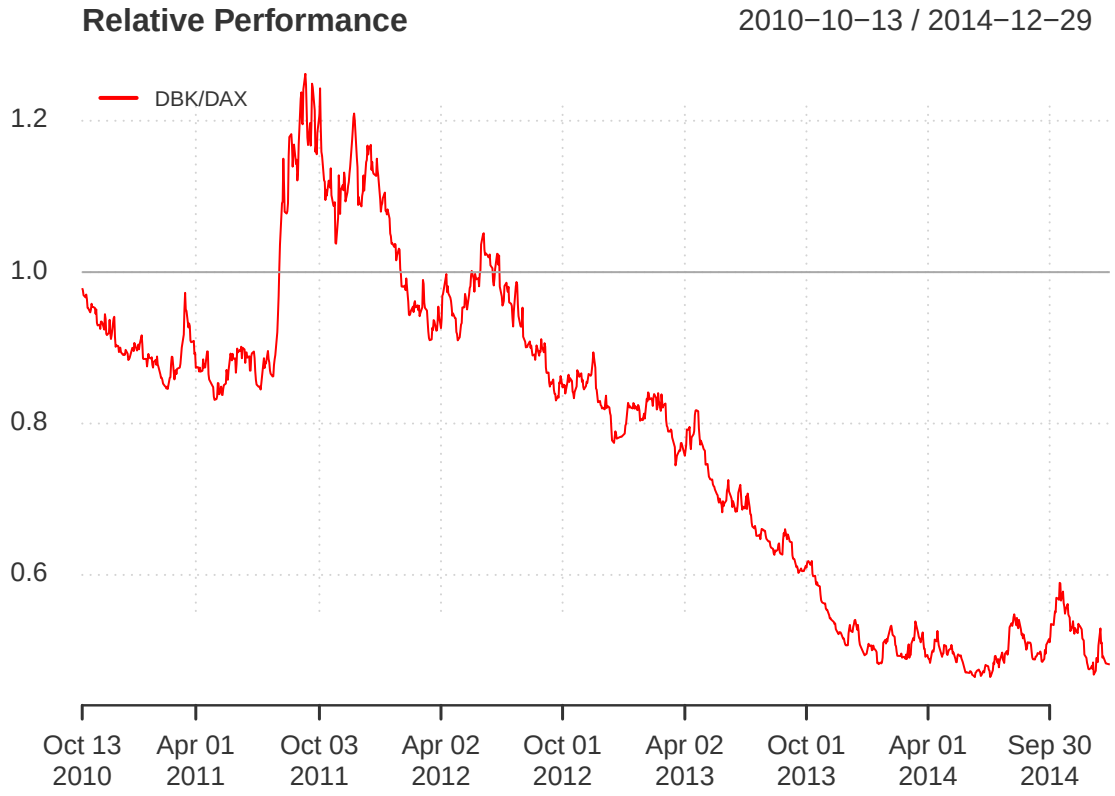
```r
table.AnnualizedReturns(returns)
```

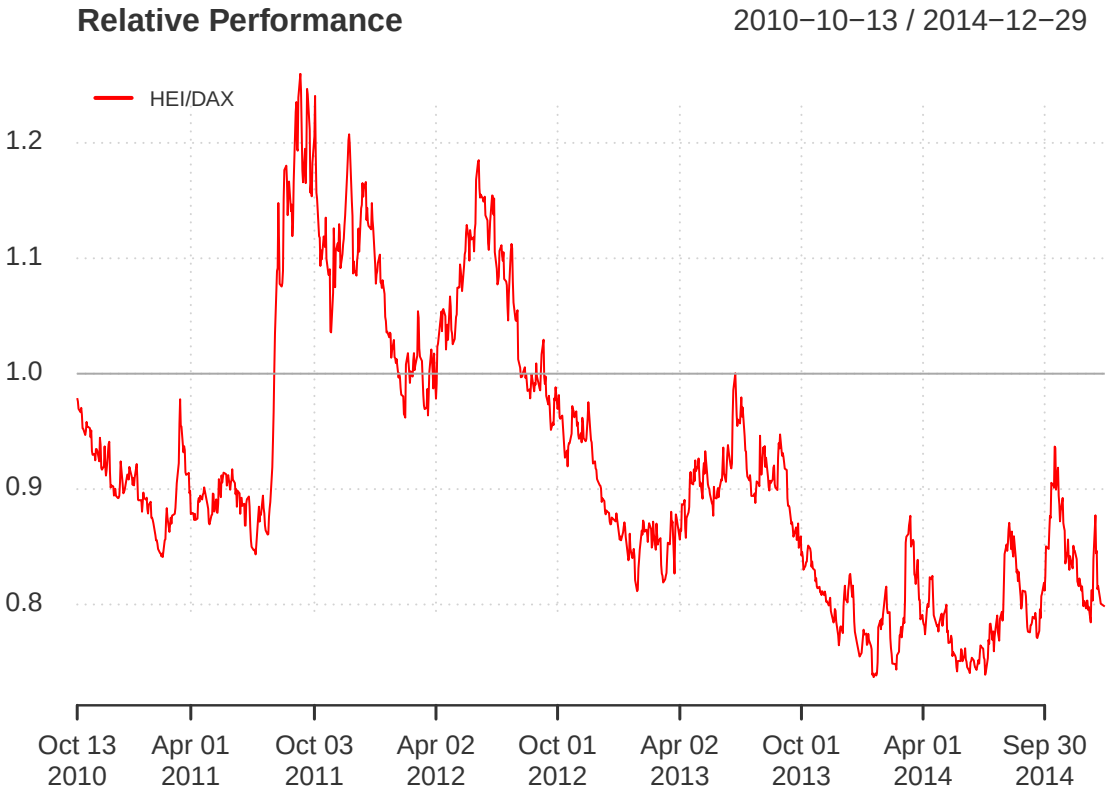
```
##                                DBK    HEI     SAP    DAX
## Annualized Return          -0.0701 0.0473 -0.0596 0.0874
## Annualized Std Dev          0.0734 0.1397  0.0943 0.2231
## Annualized Sharpe (Rf=0%) -0.9552 0.3382 -0.6328 0.3916
```

```r
# charts.PerformanceSummary(returns,geometric=FALSE,wealth.index=TRUE)
chart.RiskReturnScatter(returns,Rf=0,
                        add.sharpe=c(1,2),
                        xlim=c(0,0.25),
                        main="Return versus Risk"
                        )
```




```r
for (i in colnames(returns[,-which(names(returns) == "DAX")])){

  print(chart.RelativePerformance(returns[,i],returns["DAX"],
                          colorset=c("red","blue"),
                          lwd=1,
                          legend.loc="topleft"))
}
```

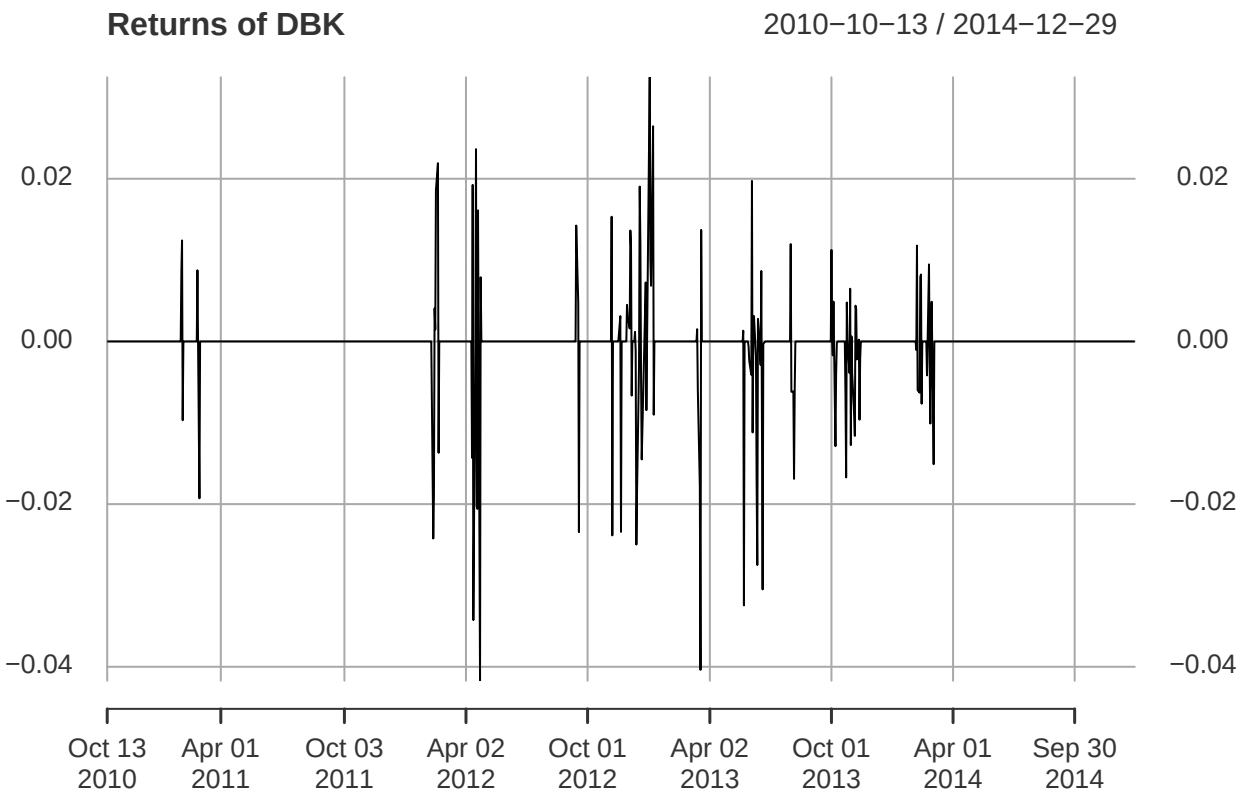

Relative Performance
2010−10−13 / 2014−12−29
DBK/DAX
1.2
1.0
0.8
0.6
Oct 13
2010
Apr 01
2011
Oct 03
2011
Apr 02
2012
Oct 01
2012
Apr 02
2013
Oct 01
2013
Apr 01
2014
Sep 30
2014

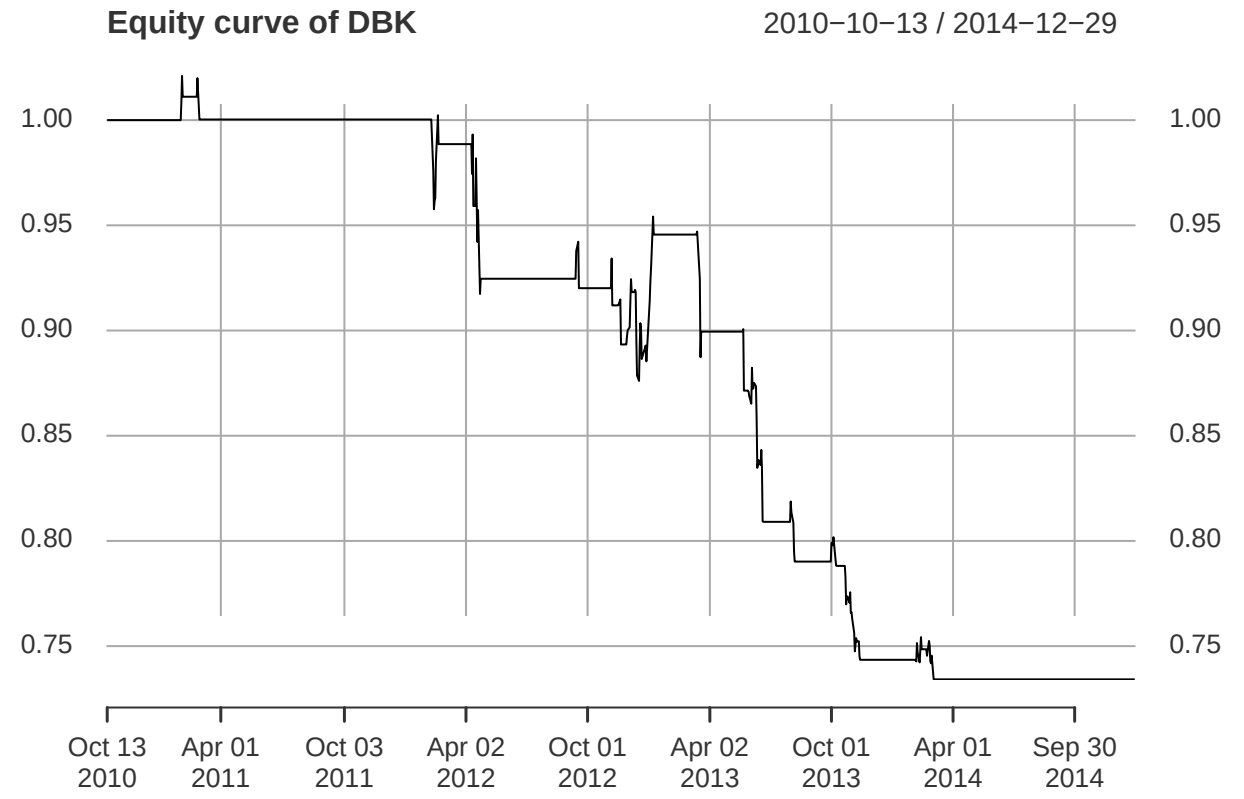

Relative Performance
2010−10−13 / 2014−12−29
HEI/DAX
1.2
1.1
1.0
0.9
0.8
Oct 13
2010
Apr 01
2011
Oct 03
2011
Apr 02
2012
Oct 01
2012
Apr 02
2013
Oct 01
2013
Apr 01
2014
Sep 30
2014

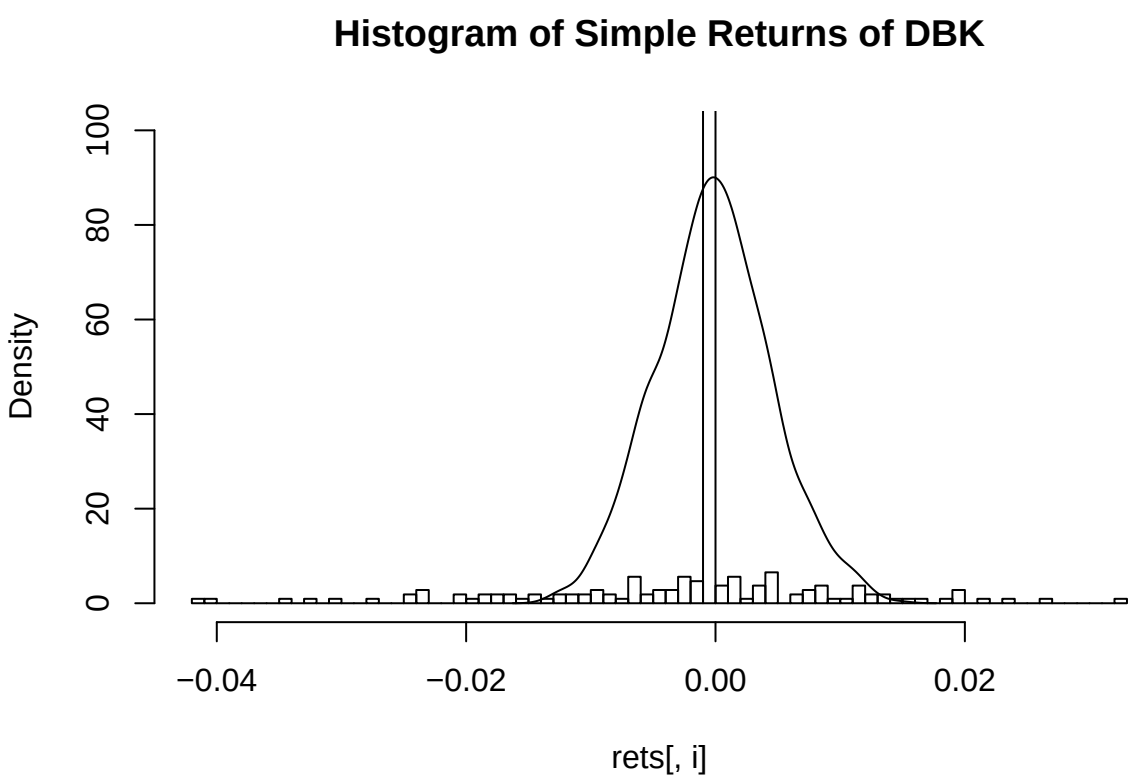
Calculations and visualizations based on returns

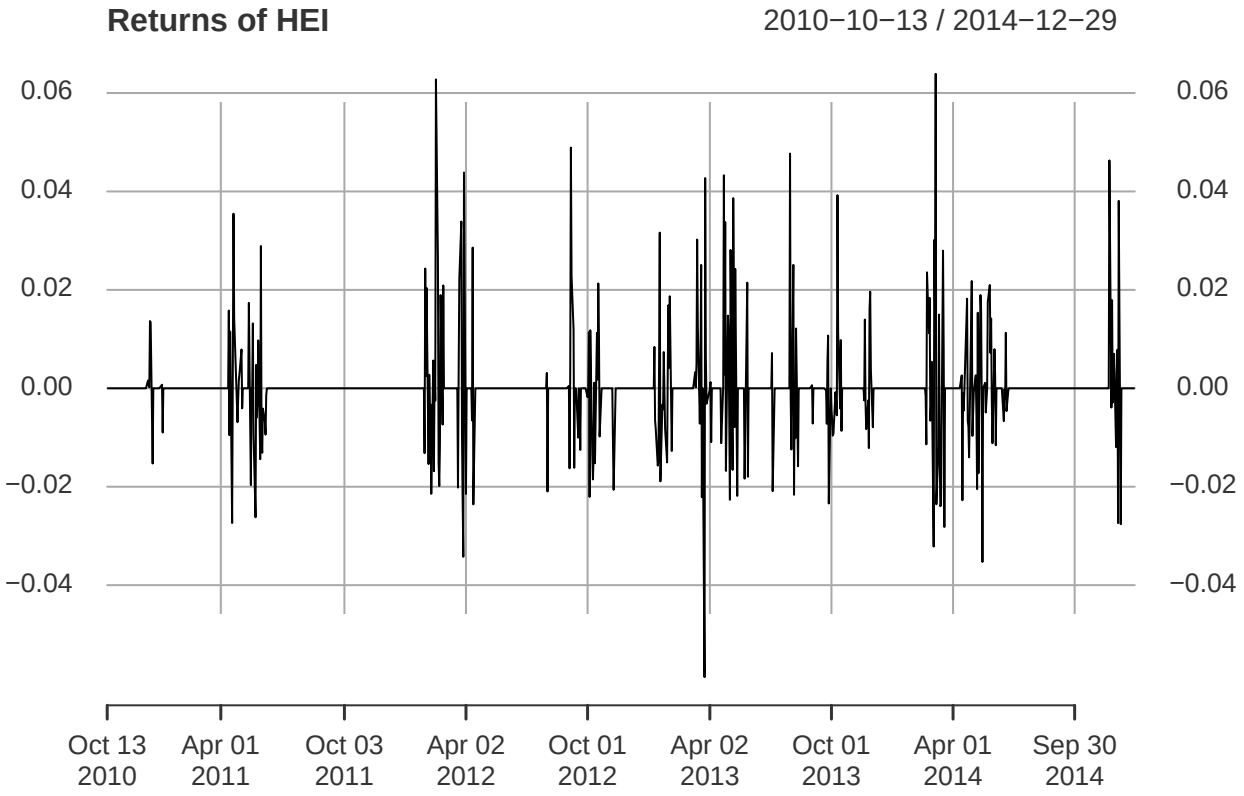
The visualization of returns, equity curve, value at risk and some other measure which can be calculated based on the returns of the trading strategy, can be observed in the following graphs.

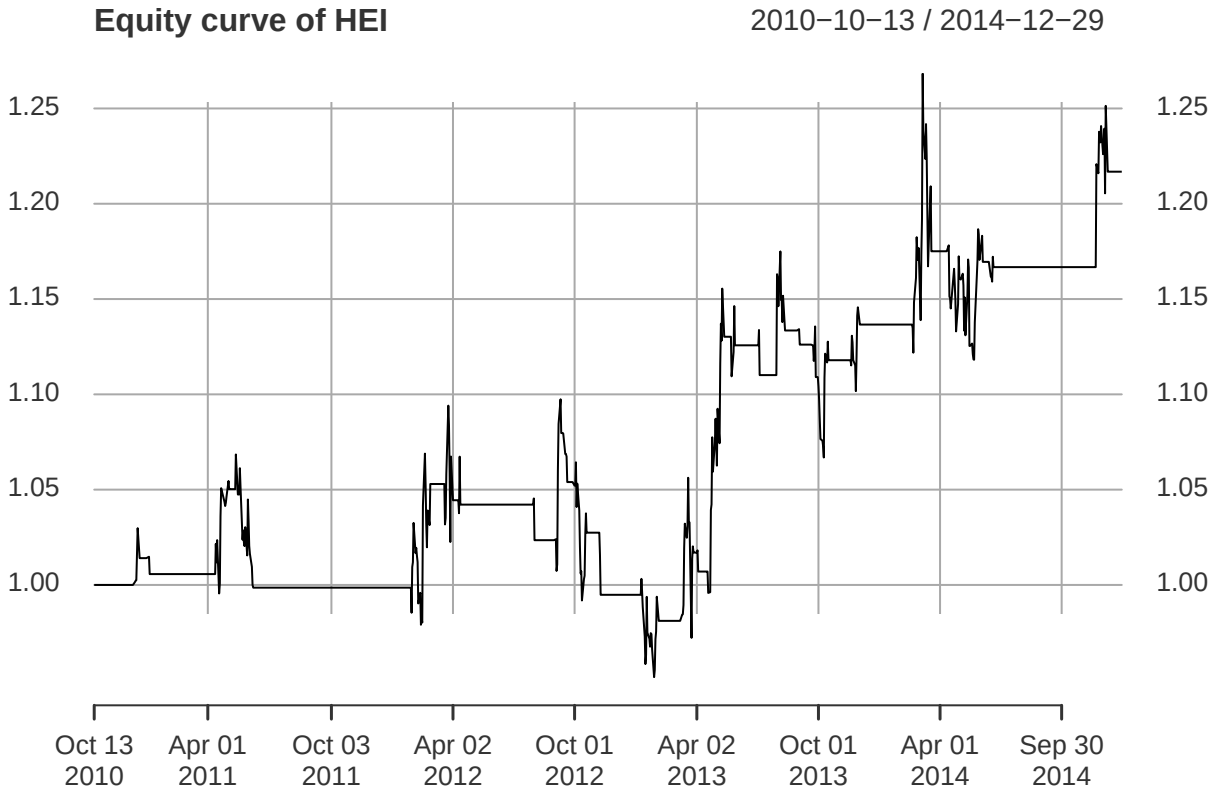
Returns and Equity Curve

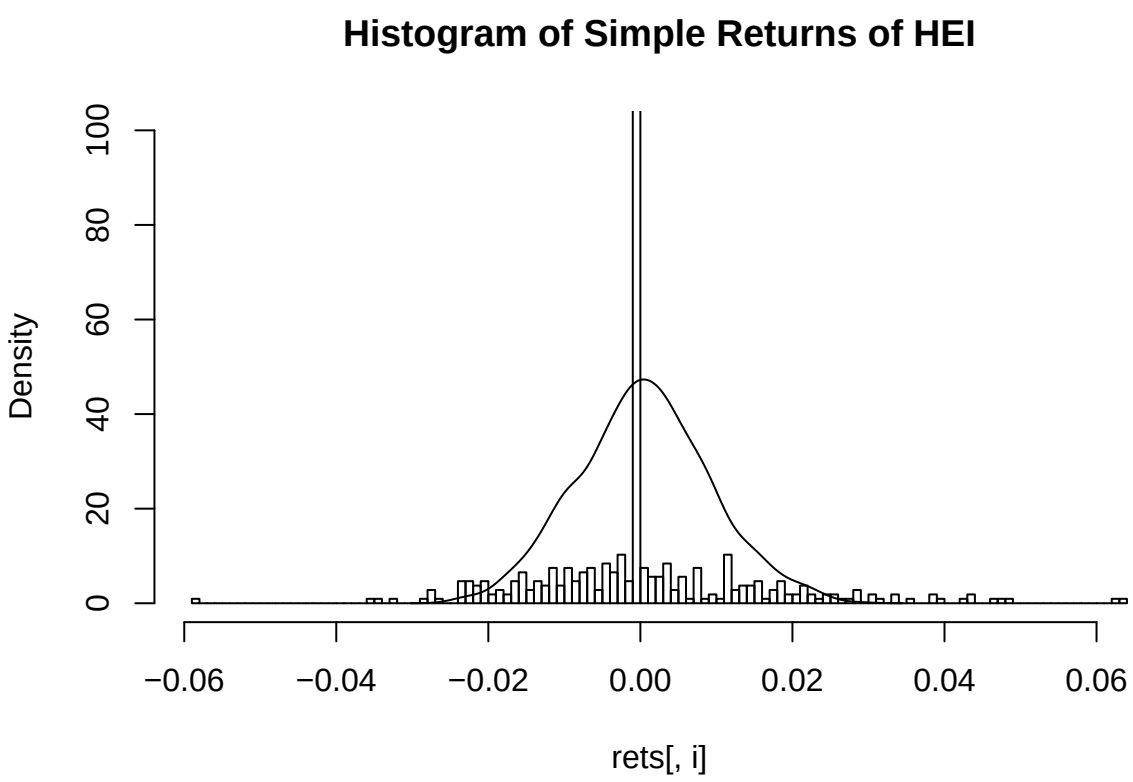


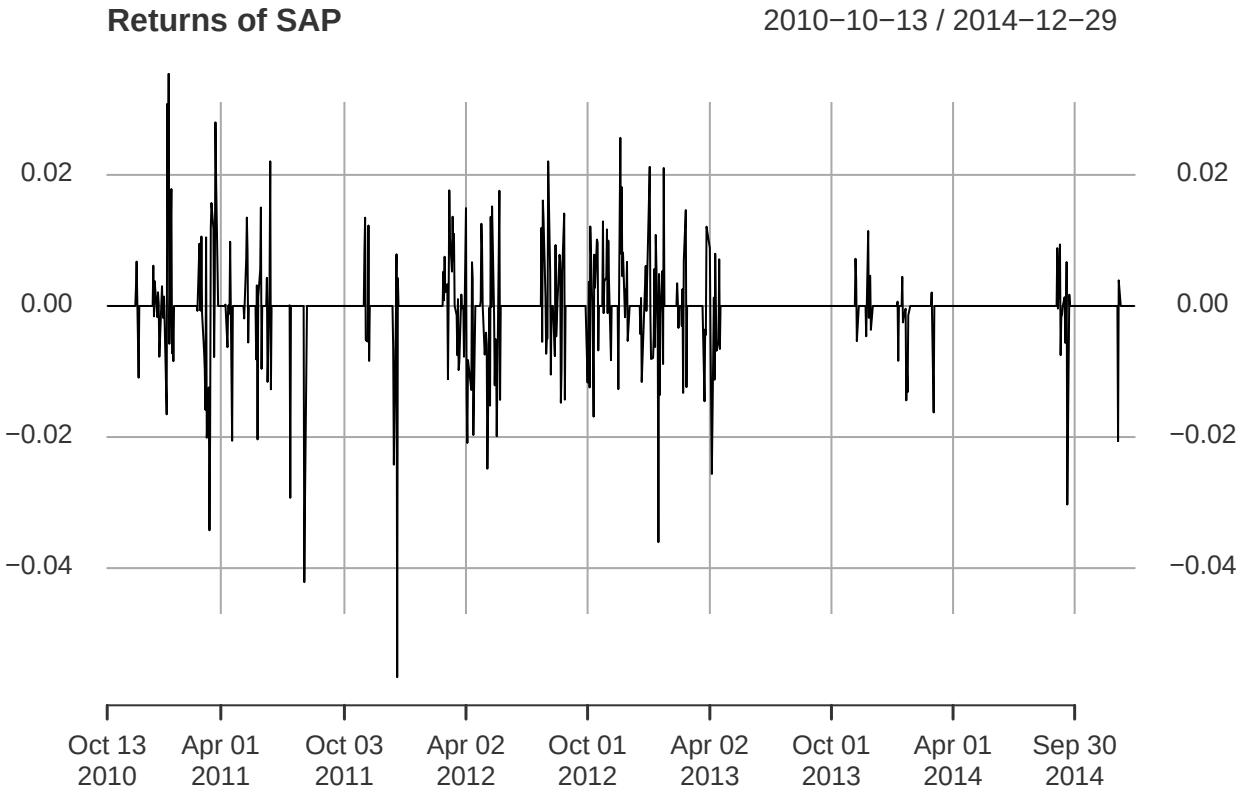


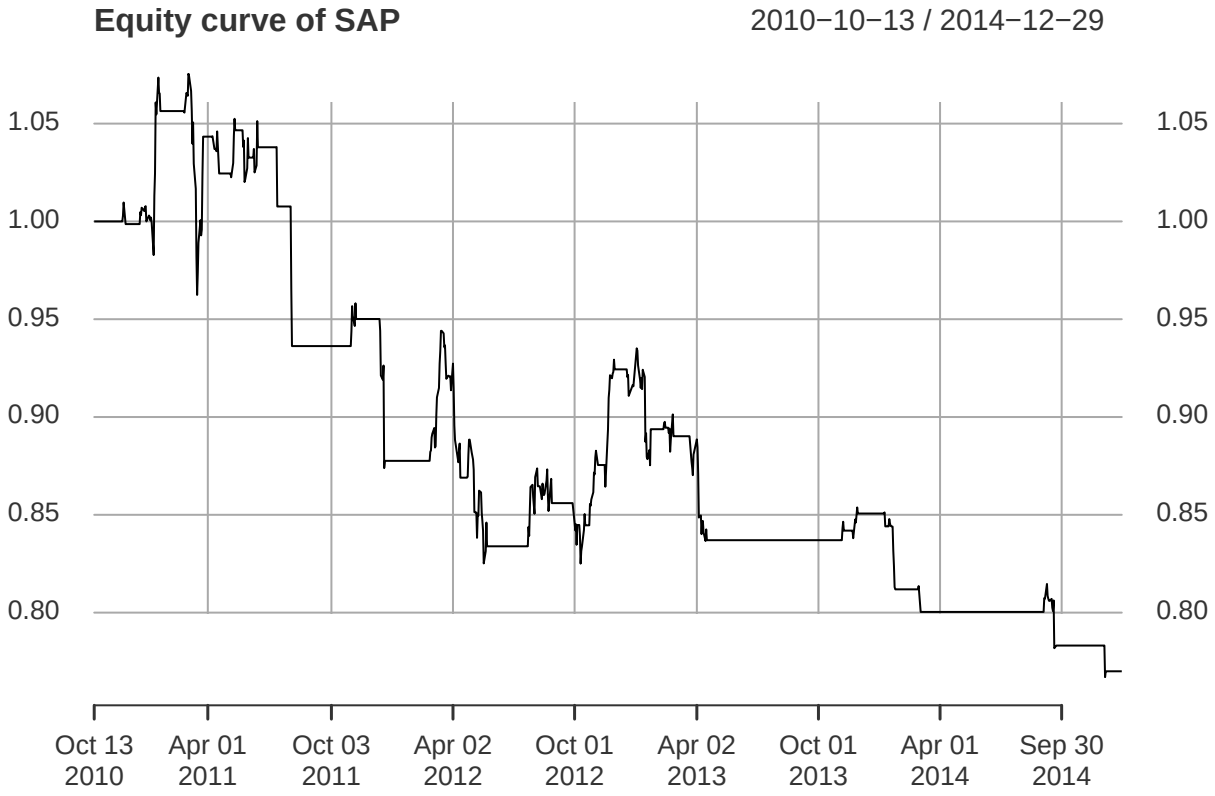

Histogram of Simple Returns of DBK
Density
0
20
40
60
80
100
−0.04
−0.02
0.00
0.02
rets[, i]

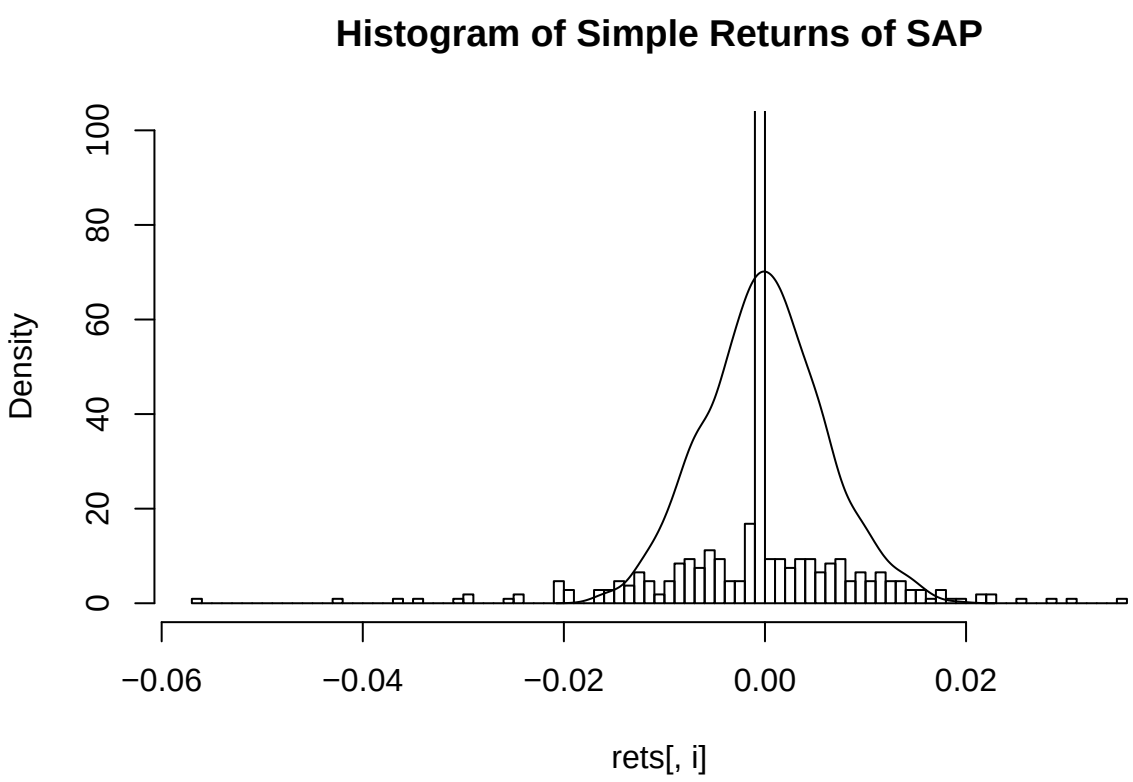

Returns of HEI
2010−10−13 / 2014−12−29
0.06
0.04
0.02
0.00
−0.02
−0.04
Oct 13
2010
Apr 01
2011
Oct 03
2011
Apr 02
2012
Oct 01
2012
Apr 02
2013
Oct 01
2013
Apr 01
2014
Sep 30
2014


Equity curve of HEI
2010−10−13 / 2014−12−29
1.25
1.20
1.15
1.10
1.05
1.00
Oct 13 2010
Apr 01 2011
Oct 03 2011
Apr 02 2012
Oct 01 2012
Apr 02 2013
Oct 01 2013
Apr 01 2014
Sep 30 2014


Histogram of Simple Returns of HEI
Density
0
20
40
60
80
100
−0.06
−0.04
−0.02
0.00
0.02
0.04
0.06
rets[, i]


Returns of SAP
2010−10−13 / 2014−12−29
0.02
0.00
−0.02
−0.04
Oct 13
2010
Apr 01
2011
Oct 03
2011
Apr 02
2012
Oct 01
2012
Apr 02
2013
Oct 01
2013
Apr 01
2014
Sep 30
2014


Equity curve of SAP
2010−10−13 / 2014−12−29
1.05
1.00
0.95
0.90
0.85
0.80
Oct 13
2010
Apr 01
2011
Oct 03
2011
Apr 02
2012
Oct 01
2012
Apr 02
2013
Oct 01
2013
Apr 01
2014
Sep 30
2014



Value at Risk

The historical "Value at Risk" measures the risk of loss for investments. More specifically, it describes the maximum amount we would expect to lose per day with a confidencelevel of 99%, given an investment of 10000.

```
## [1] "Investment: 10000"
```

```
## [1] "Confidence Level: 0.99"
```

```
## [1] "Given a 10000 investment in DBK we would expect a maximum loss of 214.4 per day."
## [1] "Given a 10000 investment in HEI we would expect a maximum loss of 235.75 per day."
## [1] "Given a 10000 investment in SAP we would expect a maximum loss of 207.11 per day."
```

Conclusion and Suggestions