Exercises I

David Zimmermann

20-09-2016

Exercises

1. Load and get familiar with the data, tidy the data and save it in a clean format.
2. Visualise prices for selected stocks (level and indexed) and save the output as pdfs.
3. Compute returns for all stocks, visualise correlations between different selected stocks.

0) File System:

```
list.files(full.names = T, recursive = T)
```

```
##  [1] "./data/stockData.csv"     "./data/tidyStockData.csv"
##  [3] "./ex1.html"               "./ex1.pdf"
##  [5] "./ex1.Rmd"                "./exercises/README.md"
##  [7] "./plot_correlation.pdf"   "./price_index.pdf"
##  [9] "./prices.pdf"             "./R/01_load_data.R"
## [11] "./R/02_price_vis.R"       "./R/03_cor_vis.R"
## [13] "./R/createStockData.R"    "./R/functions.R"
```

1) Data Munging and Tidying

```
library(dplyr)
library(readr)
library(tidyr)

# setwd("..") # if necessary, change working directory

df <- read_csv("data/stockData.csv")

names(df)
```

```
##  [1] "date" "AAPL" "AXP"  "BA"   "CAT"  "CSCO" "CVX"  "DD"   "DIS"  "GE"
## [11] "GS"   "HD"   "IBM"  "INTC" "JNJ"  "JPM"  "KO"   "MCD"  "MMM"  "MRK"
## [21] "MSFT" "NKE"  "PFE"  "PG"   "TRV"  "UNH"  "UTX"  "V"    "VZ"   "WMT"
## [31] "XOM"
```

```
dim(df)
```

```
## [1] 4204   31
```

```
summary(df)
```

```
##       date                AAPL              AXP               BA
##  Min.   :2000-01-03   Min.   :  0.858   Min.   : 9.051   Min.   : 18.47
##  1st Qu.:2004-03-10   1st Qu.:  3.135   1st Qu.:33.734   1st Qu.: 37.11
##  Median :2008-05-13   Median : 17.264   Median :41.126   Median : 56.46
##  Mean   :2008-05-12   Mean   : 34.492   Mean   :45.841   Mean   : 63.75
##  3rd Qu.:2012-07-13   3rd Qu.: 60.460   3rd Qu.:54.051   3rd Qu.: 72.14
##  Max.   :2016-09-16   Max.   :129.181   Max.   :92.661   Max.   :151.42
##       CAT               CSCO             CVX               DD
##  Min.   :  9.884   Min.   : 7.42   Min.   : 19.18   Min.   :11.92
##  1st Qu.: 26.537   1st Qu.:15.71   1st Qu.: 28.74   1st Qu.:25.88
##  Median : 51.676   Median :18.97   Median : 56.61   Median :30.65
##  Mean   : 50.878   Mean   :21.16   Mean   : 59.78   Mean   :35.88
##  3rd Qu.: 75.488   3rd Qu.:23.21   3rd Qu.: 87.39   3rd Qu.:42.62
##  Max.   :102.594   Max.   :69.08   Max.   :122.55   Max.   :73.65
##       DIS               GE               GS               HD
##  Min.   : 11.24   Min.   : 5.222   Min.   : 46.99   Min.   : 14.77
##  1st Qu.: 21.53   1st Qu.:17.102   1st Qu.: 83.90   1st Qu.: 25.99
##  Median : 29.11   Median :22.665   Median :124.88   Median : 30.95
##  Mean   : 39.40   Mean   :21.543   Mean   :123.46   Mean   : 44.46
##  3rd Qu.: 45.15   3rd Qu.:24.956   3rd Qu.:157.71   3rd Qu.: 47.95
##  Max.   :120.04   Max.   :36.086   Max.   :222.11   Max.   :138.06
##       IBM              INTC             JNJ              JPM
##  Min.   : 43.02   Min.   : 9.328   Min.   : 22.55   Min.   :10.42
##  1st Qu.: 72.35   1st Qu.:15.898   1st Qu.: 39.88   1st Qu.:27.45
##  Median : 92.23   Median :18.931   Median : 48.31   Median :33.21
##  Mean   :108.68   Mean   :20.900   Mean   : 55.48   Mean   :35.96
##  3rd Qu.:150.20   3rd Qu.:23.170   3rd Qu.: 59.80   3rd Qu.:39.68
##  Max.   :195.36   Max.   :52.540   Max.   :124.57   Max.   :68.08
##        KO              MCD              MMM              MRK
##  Min.   :12.60   Min.   :  8.438   Min.   : 26.57   Min.   :15.71
##  1st Qu.:16.14   1st Qu.: 20.771   1st Qu.: 49.82   1st Qu.:27.08
##  Median :20.00   Median : 43.155   Median : 62.52   Median :31.22
##  Mean   :24.15   Mean   : 49.990   Mean   : 74.97   Mean   :34.22
##  3rd Qu.:33.01   3rd Qu.: 80.880   3rd Qu.: 82.20   3rd Qu.:40.26
##  Max.   :46.16   Max.   :129.632   Max.   :180.81   Max.   :63.40
##       MSFT             NKE              PFE              PG
##  Min.   :12.46   Min.   : 2.757   Min.   : 8.821   Min.   :17.45
##  1st Qu.:19.56   1st Qu.: 7.557   1st Qu.:16.052   1st Qu.:36.62
##  Median :22.67   Median :12.507   Median :18.968   Median :48.08
##  Mean   :25.91   Mean   :18.724   Mean   :20.563   Mean   :49.14
##  3rd Qu.:27.33   3rd Qu.:23.739   3rd Qu.:25.201   3rd Qu.:58.22
##  Max.   :58.17   Max.   :66.451   Max.   :37.002   Max.   :88.64
##       TRV              UNH              UTX               V
##  Min.   : 14.11   Min.   :  5.347   Min.   : 15.29   Length:4204
##  1st Qu.: 30.52   1st Qu.: 22.579   1st Qu.: 33.79   Class :character
##  Median : 39.64   Median : 40.657   Median : 52.96   Mode  :character
##  Mean   : 48.91   Mean   : 45.424   Mean   : 57.19
##  3rd Qu.: 57.64   3rd Qu.: 52.240   3rd Qu.: 74.16
##  Max.   :118.68   Max.   :143.030   Max.   :119.42
##        VZ              WMT              XOM
##  Min.   :12.64   Min.   :32.52   Min.   :21.28
```

```
##  1st Qu.:19.15   1st Qu.:39.54   1st Qu.:31.14
##  Median :22.52   Median :43.73   Median :57.49
##  Mean   :27.30   Mean   :48.95   Mean   :56.16
##  3rd Qu.:36.08   3rd Qu.:58.27   3rd Qu.:74.89
##  Max.   :56.26   Max.   :86.15   Max.   :96.96
```

```
# leave out V
df <- df %>% select(-V)

# have a look at the data again
df %>% head
```

```
## # A tibble: 6 × 30
##         date     AAPL      AXP       BA      CAT     CSCO      CVX
##       <date>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
## 1 2000-01-03 3.660058 36.79188 28.19358 15.56339 46.61910 23.54678
## 2 2000-01-04 3.351477 35.40268 28.14973 15.36334 44.00368 23.54678
## 3 2000-01-05 3.400523 34.50348 29.90361 15.72342 43.86887 23.96915
## 4 2000-01-06 3.106247 35.20644 30.21054 16.52360 43.14087 24.98986
## 5 2000-01-07 3.253385 35.71902 31.08747 17.06371 45.67539 25.42982
## 6 2000-01-10 3.196165 36.21695 30.64900 16.50359 47.37407 24.74348
## # ... with 23 more variables: DD <dbl>, DIS <dbl>, GE <dbl>, GS <dbl>,
## #   HD <dbl>, IBM <dbl>, INTC <dbl>, JNJ <dbl>, JPM <dbl>, KO <dbl>,
## #   MCD <dbl>, MMM <dbl>, MRK <dbl>, MSFT <dbl>, NKE <dbl>, PFE <dbl>,
## #   PG <dbl>, TRV <dbl>, UNH <dbl>, UTX <dbl>, VZ <dbl>, WMT <dbl>,
## #   XOM <dbl>
```

```
df_long <- gather(df, key = ticker, value = price, -date)
head(df_long)
```

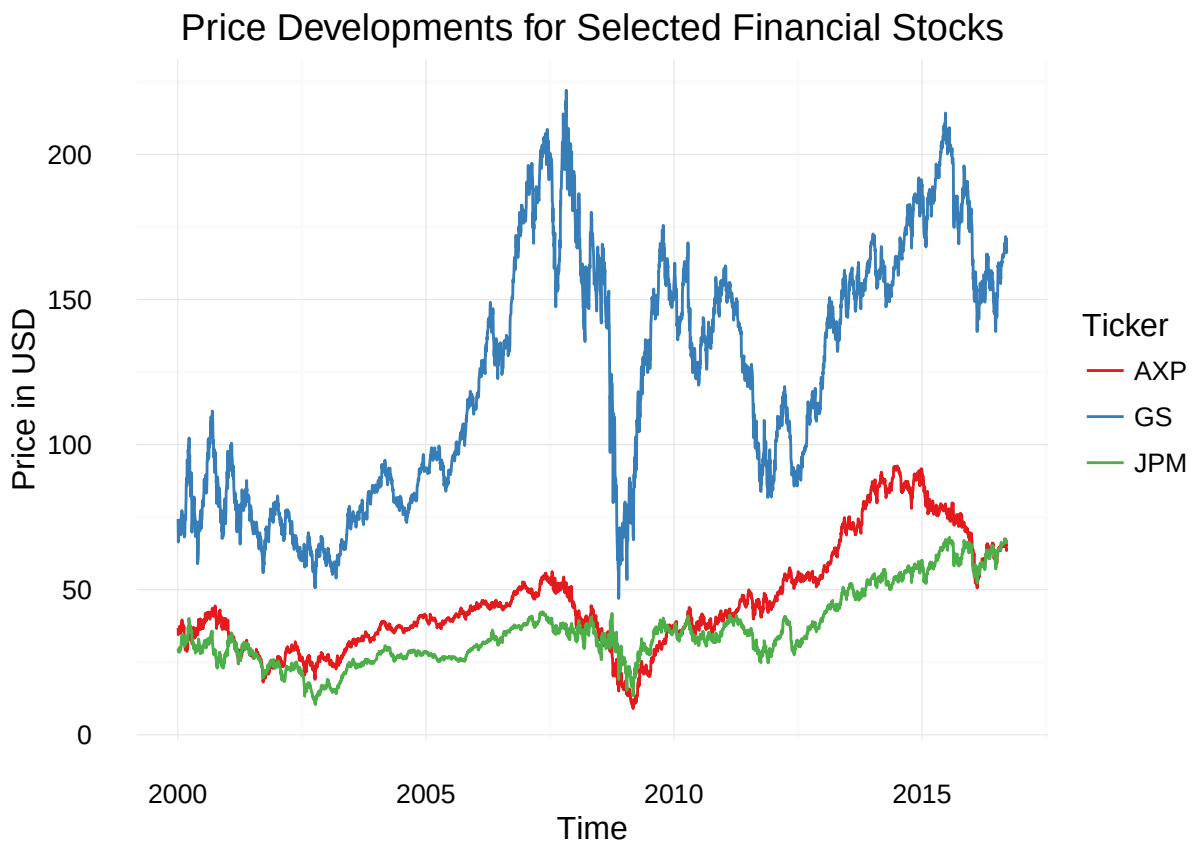
```
## # A tibble: 6 × 3
##         date ticker    price
##       <date>  <chr>    <dbl>
## 1 2000-01-03   AAPL 3.660058
## 2 2000-01-04   AAPL 3.351477
## 3 2000-01-05   AAPL 3.400523
## 4 2000-01-06   AAPL 3.106247
## 5 2000-01-07   AAPL 3.253385
## 6 2000-01-10   AAPL 3.196165
```

```
# save the tidy data
write_csv(df_long, "data/tidyStockData.csv")
```

2) Data Vis 1

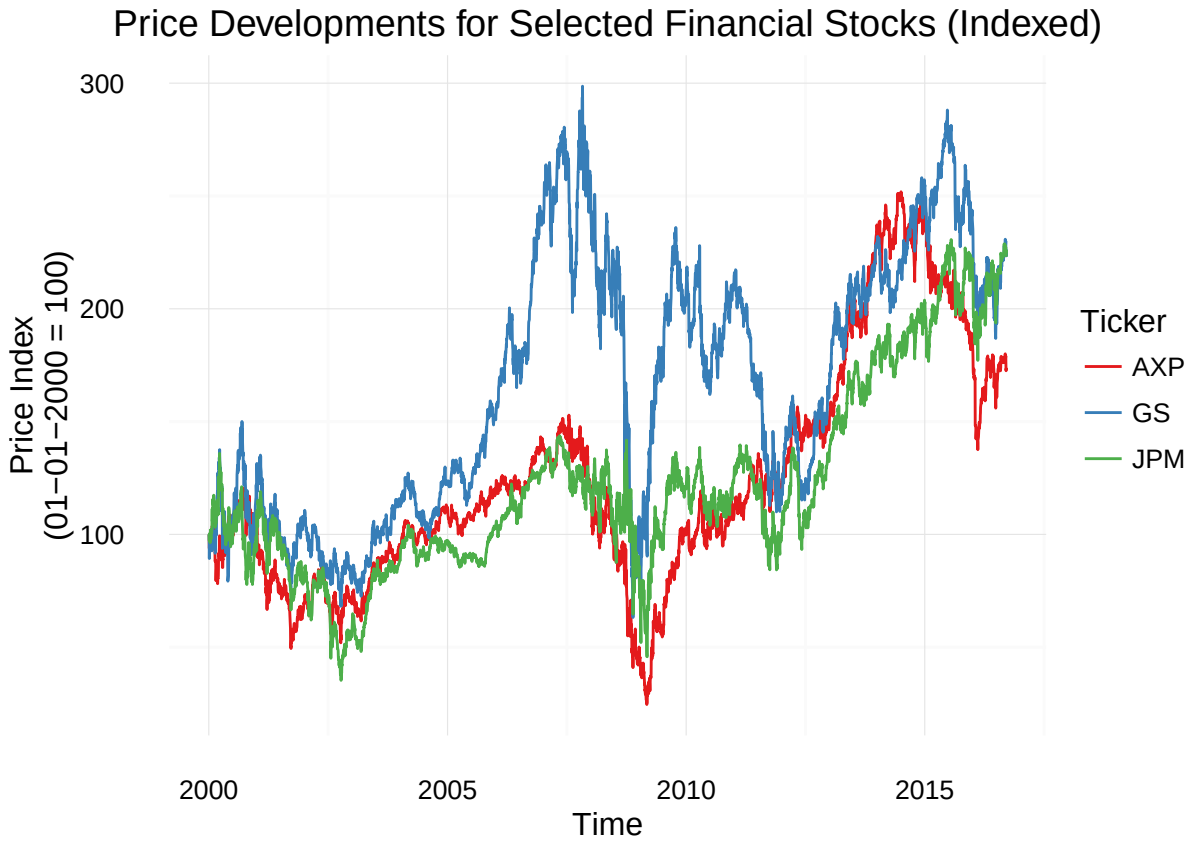
```
library(ggplot2)
library(RColorBrewer)

# setwd("..") # if necessary, change working directory
```

```r
df <- read_csv("data/tidyStockData.csv")
df <- df %>% mutate(date = as.Date(date))

# 1 Filter selected Stocks
ticker_selection <- c("GS", "JPM", "AXP")
df_selection <- df %>% filter(ticker %in% ticker_selection)

# 2 Compute indexed prices
df_selection <- df_selection %>% group_by(ticker) %>%
  mutate(price_index = (price / price[1]) * 100)

# 3 A) Visualise Prices
plot_prices <- ggplot(df_selection, aes(x = date, y = price, color = ticker)) +
  geom_line() +
  labs(x = "Time", y = "Price in USD",
       title = "Price Developments for Selected Financial Stocks") +
  scale_color_manual(name = "Ticker",
                     values = brewer.pal(length(ticker_selection), "Set1")) +
  theme_minimal()

plot_prices # show plot
```




```r
ggsave("prices.pdf", plot_prices)

# 3 B) Visualise Indexed Prices
plot_idx_prices <- ggplot(df_selection, aes(x = date, y = price_index,
```

```
                                          color = ticker)) +
  geom_line() +
  labs(x = "Time", y = "Price Index\n(01-01-2000 = 100)",
       title = "Price Developments for Selected Financial Stocks (Indexed)") +
  scale_color_manual(name = "Ticker",
                     values = brewer.pal(length(ticker_selection), "Set1")) +
  theme_minimal()

plot_idx_prices # show plot
```




```
ggsave("price_index.pdf", plot_idx_prices)
```

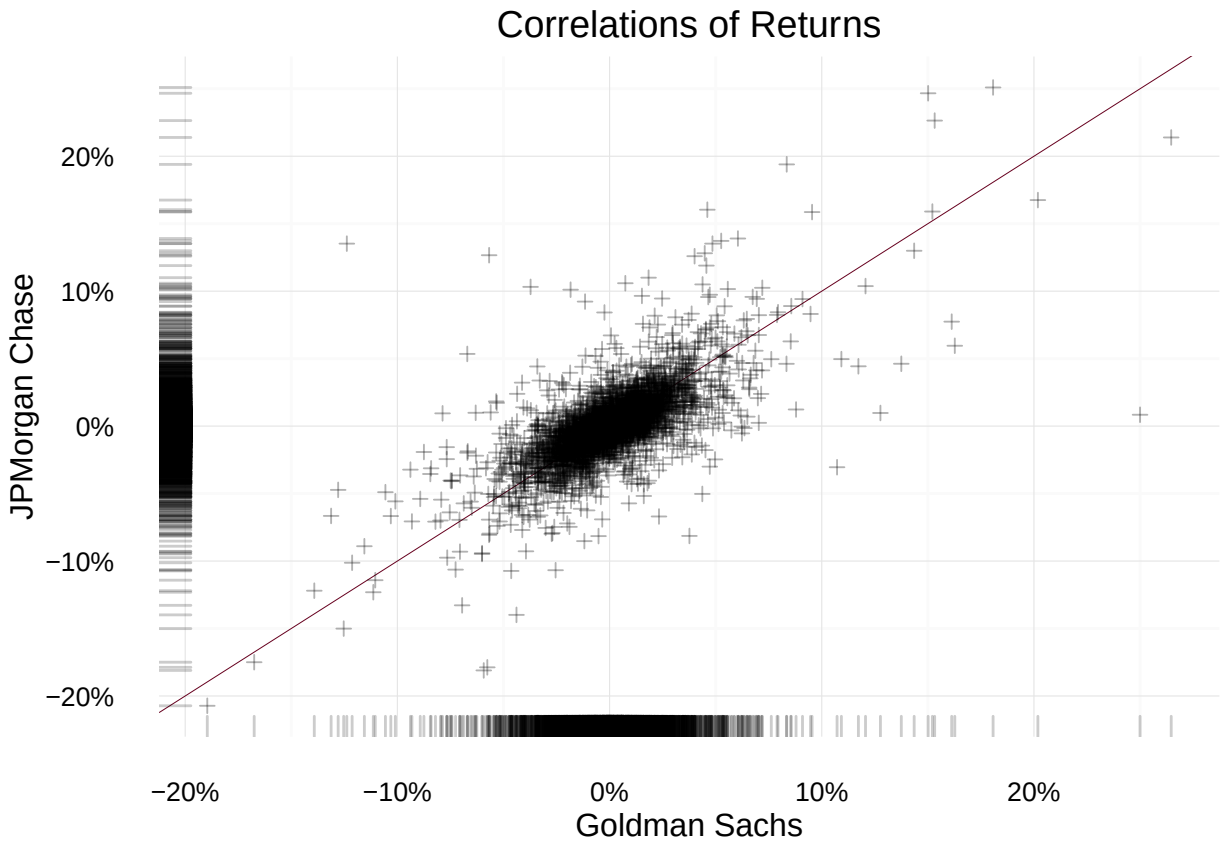
3) Data Vis 2

```
library(RColorBrewer)
library(scales) # for percent axis

# setwd("..") # if necessary, change working directory

df <- read_csv("data/tidyStockData.csv")
df <- df %>% mutate(date = as.Date(date))

# 1 Compute returns
```

```r
df <- df %>% group_by(ticker) %>% mutate(ret = price / lag(price) - 1)

df_selection <- df %>% filter(ticker %in% c("GS", "JPM"))

df_selection_wide <- spread(df_selection %>% select(date, ticker, ret),
                            key = ticker, value = ret)

# 2 Compute Correlations
cor(df_selection_wide %>% na.omit %>% select(GS, JPM))
```

```
##            GS       JPM
## GS  1.0000000 0.6998404
## JPM 0.6998404 1.0000000
```

```r
# 3 Visualise Correlations
plot_cor <- ggplot(df_selection_wide, aes(x = GS, y = JPM)) +
  geom_abline(slope = 1, intercept = 0, color = "#67001f", size = 0.1) +
  geom_point(shape = 3, alpha = 0.3) +
  geom_rug(alpha = 0.2) +
  scale_x_continuous(labels = percent) +
  scale_y_continuous(labels = percent) +
  labs(x = "Goldman Sachs", y = "JPMorgan Chase",
       title = "Correlations of Returns") +
  theme_minimal()

plot_cor # show plot
```



```r
ggsave("plot_correlation.pdf", plot_cor)
```