CAP-model

Jesse Keränen

1/14/2022

Prologue

In this file I take a look on one of the most widely known and used asset pricing models, namely capital asset pricing model. Main contributors in creation of Cap-model were William Sharpe and John Lintner in the sixties. In the first part of this file I show the mathematical proof of the cap-model. This derivation of cap-model is not my own work, rather I replicate the derivation from my asset management course and try to gain deeper understanding of the subject. Cap-model is really basic concept of the finance. To try something new in, in the second part of this file I perform Fama-MacBeth (1973) beta estimations to my (bad) stock price data from Fama-French-file.

Capital asset pricing-model

One thing Markowitz didn't consider was price of a single stock. In Markowitz-model we have found out the optimal portfolio for investor, i.e. portfolio with highest Sharpe ratio, and that the portfolio is same regardless of the desired return level.

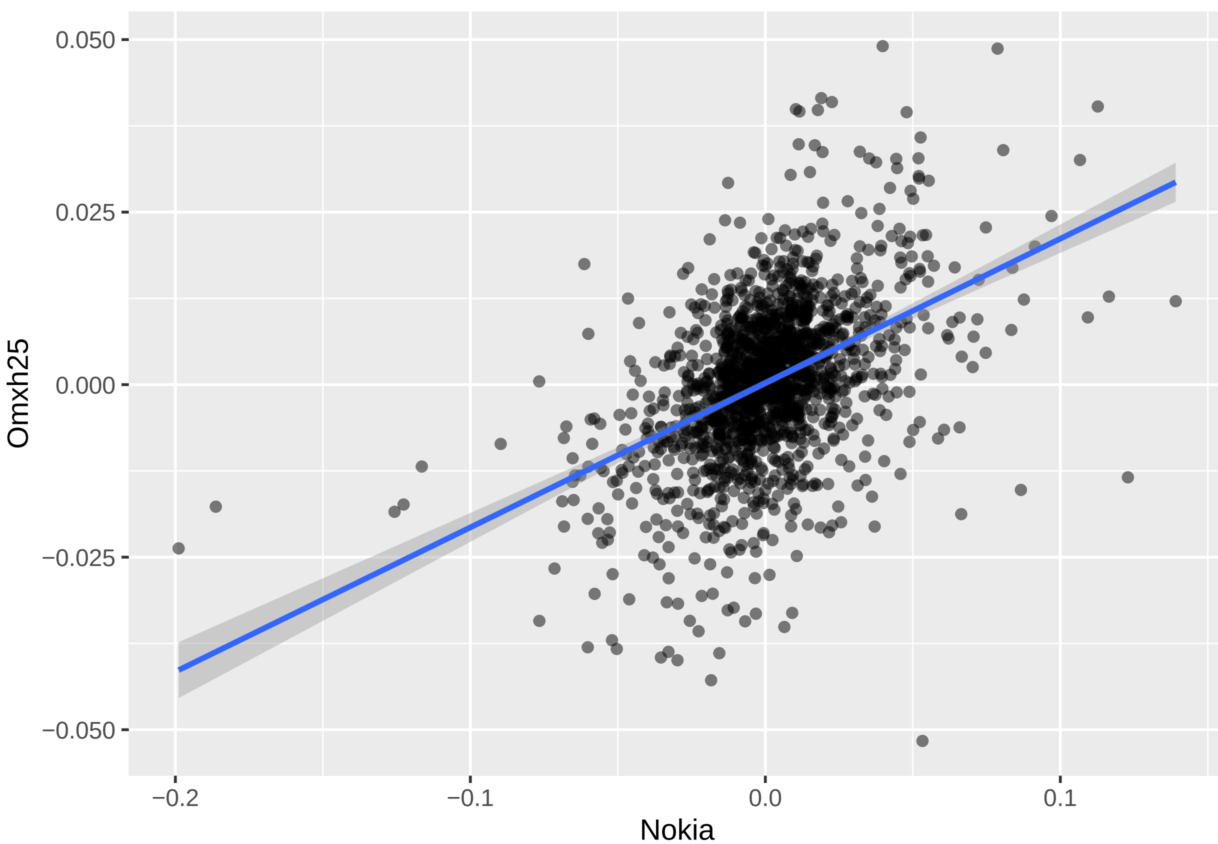
$$Sharpe\ Ratio = \frac{R_p - R_f}{\sigma_p}$$

In capital asset model we consider situation where there is more than one investor. If we have perfect financial markets and every investor has access to same information, we can expect the expectations of every investor to be same. This means vector of expected returns μ and covariance matrix $\underline{\Omega}$ are same for each investor. This also means that every investor has same optimal portfolio. This is only possible if this portfolio is market portfolio, i.e. market portfolio has highest possible Sharpe ratio.

According to Cap-model there are two kinds of risk. Unsystematic and systematic risk. Unsystematic risk is asset specific risk that can be diversified. One example of this kind of risk could be risk associated with product company produces. Systematic risk affects whole markets and it cannot be diversified by adding more assets to the portfolio. Example of systematic risk could be tax increase. Since investor doesn't have to bear unsystematic risk, he is not rewarded with higher return for taking it. Only systematic risk is compensated with higher returns.

```
library(tidyverse)
library(ggplot2)
library(data.table)
library(readxl)

nokia <- fread("extdata/nokia.csv")
omxh25 <- as.data.table(read_excel("extdata/omxh25.xlsx"))
```

```
colnames(omxh25)[c(1, 4)] <- c("Date", "Omxh25")
colnames(nokia)[6] <- "Nokia"

omxh25[, Date := as.Date(Date)]
omxh25[, Omxh25 := as.numeric(Omxh25)]
omxh25 <- omxh25[order(Date)]
omxh25[, Return := Omxh25/shift(Omxh25) -1]

nokia[, Return := Nokia/shift(Nokia) -1]

dt <- merge(nokia[, c(1, 8)], omxh25[, c(1, 8)], by = "Date")
colnames(dt) <- c("Date", "Nokia", "Omxh25")
dt <- dt[!(is.na(Nokia)) & !(is.na(Omxh25))]

ggplot(dt, aes(Nokia, Omxh25)) + geom_point(alpha = 0.5) + geom_smooth(method = "lm")
```




```
cor(dt$Nokia, dt$Omxh25)
```

```
## [1] 0.4599023
```

Why has the tangent portfolio to be also the market portfolio? Following is how I have understood how the market equilibrium works. In Markowitz-model we showed that the tangent portfolio has the highest possible Sharpe ratio. Then investor can't improve by changing weight of any asset. We stated above that only systematic risk is rewarded. That's why we have to consider every stock as part of a portfolio, if we want to find their right prices.

Let's say we have one individual stock and tangent portfolio. Then consider situation where we form a new portfolio from this stock and the market portfolio and the Sharpe ratio of this new portfolio is lower than Sharpe ratio of our original tangent portfolio. Investor wouldn't want to include this stock to his portfolio. Since the expectations are identical for every investor no-investor would include this stock to their portfolio. This would cause lack of demand for this stock, which would then decrease price of this stock. This works also vice versa when the new portfolio has higher Sharpe ratio than original tangent portfolio. This fluctuation of asset prices would continue until all assets are appropriately priced and part of this tangent portfolio.

From findings of the Markowitz-model we can derive Cap-model formula. First we need to define some variables:

$$w_i = Wealth\ of\ investor\ i$$
$$\overline{p}_m = Vector\ of\ market\ capitalizations$$
$$\sum_i w_i = W = Total\ capitalization\ of\ the\ market$$
$$\overline{x}_m = \overline{p}\ \tfrac{1}{W} = Vector\ of\ market\ shares$$

In Markowitz-model (or more precicely Tobin-model) we showed what are the desired weights given expected returns and covariance matrix of the assets in the market. If we multiply these weights by the wealth of the investor and sum these products of from every investor together, this sum should equal market capitalization:

$$\sum_i w_i \overline{x}_i = \sum_i w_i \tfrac{\mu_i - r_f}{\overline{\pi}^t \underline{\Omega}^{-1} \overline{\pi}} \underline{\Omega}^{-1} \overline{\pi} = \overline{p}_m$$

We also know that some part of the wealth B can be invested in risk free assets. It is also stated that sum of total wealth invested in risk free assets should be zero. I believe this is because institutions issuing risk free assets are also market participants. When some one invests in risk free asset, issuer of this risk free security has short position on risk free asset.

$$\sum_i w_i B_i = 0$$

Return of the investor is:

$$\mu_i = (1 - B_i) r_m + B_i r_f$$

Then we can write that:

$$\sum_i w_i \tfrac{\mu_i - r_f}{\overline{\pi}^t \underline{\Omega}^{-1} \overline{\pi}} = \sum_i w_i \tfrac{(1-B_i) r_m + B_i r_f - r_f}{\overline{\pi}^t \underline{\Omega}^{-1} \overline{\pi}} = \tfrac{\sum_i w_i r_m - \sum_i w_i B_i r_m + \sum_i w_i B_i r_f - \sum_i w_i r_z}{\overline{\pi}^t \underline{\Omega}^{-1} \overline{\pi}}$$

And since $\sum_i w_i B_i = 0$ we get that:

$$\tfrac{\sum_i w_i r_m - \sum_i w_i r_f}{\overline{\pi}^t \underline{\Omega}^{-1} \overline{\pi}} = \tfrac{W(r_m - r_f)}{\overline{\pi}^t \underline{\Omega}^{-1} \overline{\pi}}$$

Now we can plug this back to our market capitalization formula:

$$\tfrac{W(r_m - r_f)}{\overline{\pi}^t \underline{\Omega}^{-1} \overline{\pi}} \underline{\Omega}^{-1} \overline{\pi} = \overline{p}_m$$

If we divide both sides in above formula by total wealth W vector of market capitalizations $\overline{p}_m$ changes to vector of market shares $\overline{x}_m$. We can also multiply both sides by covariance matrix $\underline{\Omega}$ and vector of market shares $\overline{x}_m$. so that we get variance of the market portfolio to the right side of the equation.

$$\overline{\pi} = \frac{\overline{\pi}^t \underline{\Omega}^{-1} \overline{\pi}}{r_m - r_f} \overline{x}^t \underline{\Omega}$$

$$\overline{x}_m \overline{\pi} = \frac{\overline{\pi}^t \underline{\Omega}^{-1} \overline{\pi}}{r_m - r_f} \overline{x}^t \underline{\Omega} \overline{x} = \frac{\overline{\pi}^t \underline{\Omega}^{-1} \overline{\pi}}{r_m - r_f} \sigma_m^2$$

Sum of market shares times stock premiums $\overline{x}^t \overline{\pi}$ is same as market return minus risk free eate $r_m - r_f$.

$$r_m - r_f = \frac{\overline{\pi}^t \underline{\Omega}^{-1} \overline{\pi}}{r_m - r_f} \sigma_m^2$$

$$\overline{\pi}^t \underline{\Omega}^{-1} \overline{\pi} = \frac{(r_m - r_f)^2}{\sigma_m^2}$$

Finally we get our equation for security market line by plugging above function back to our stock return premium function:

$$\overline{\pi} = \frac{r_m - r_f}{\sigma_m^2} \overline{x}_m^t \underline{\Omega}$$

Note that this is just security market line function. We have covariance matrix $\underline{\Omega}$ in our function. If want to derive function that gives us equilibrium return of the single asset we need to examine term $\overline{x}_m^t \underline{\Omega}$ one line at the time. With respect to ith line it can be written as:

$$\sum_i x_{im} \sigma_{ij} = \sum_i x_{im} E[(\tilde{r}_i - \overline{r}_i)(\tilde{r}_j - \overline{r}_j)]$$

How we can derive covariance between stock i's and market returns from above function doesn't seem too intuitive to me, but let's try to do same calculations with our example data from Markowitz-file. We only have five stock in our portfolio and we examine first of them, Apple.

```
# Simulate market returns
market_return <- merge(x, prices, by = "Company")
market_return2 <- market_return[, sum(Returns*Weight), by = Date]

# Seems to work
sum(x$Weight*cov_matrix[1,])
```

```
## [1] 0.002017826
```

```
cov(prices[Company == "AAPL", Returns], market_return2$V1)
```

```
## [1] 0.002017826
```

$$\sum_i E[(\tilde{r}_i - \overline{r}_i) x_{im} (\tilde{r}_j - \overline{r}_j)$$

$$E[(\sum_i x_{im} \tilde{r}_i - \sum_i x_{im} \overline{r}_i)(\tilde{r}_j - \overline{r}_j)]$$

In the first part of our equation we have sum of each assets market weight times realized return of that asset. Then we have sum of each assets market weight times assets average return. These are naturally realized market return and average market return.

$$E[(\tilde{r}_m - \overline{r}_m)(\tilde{r}_j - \overline{r}_j)] = cov(r_m, r_j)$$

We know that beta β of a stock is covariance between stock and market divided by variance of the market. Then we finally get our capital asset pricing formula:

$$\overline{\pi}_j = \mu_j - r_f = \frac{r_m - r_f}{\sigma_m^2} cov(r_m, r_j)$$

$$\mu_j = r_f + \beta(r_m - r_f)$$

Now that we have a formula, let's calculate equilibrium and price for Nokia stock in 2008 using our data from beginning. Suppose that risk free rate is at 0.01.

```
# Convert to daily, since we have daily data
r_f <- 0.01/360
mu_j <- r_f + cov(dt$Nokia, dt$Omxh25)/var(dt$Omxh25) *
  (mean(dt$Omxh25) - r_f)

# Nokia's annual return
mu_j*360
```

```
## [1] 0.15658
```

```
mean(dt$Nokia)*360
```

```
## [1] 0.3416414
```

```
mean(dt$Omxh25)*360
```

```
## [1] 0.1549426
```

```
# With this return level what would be Nokia's price one year after end of our data?
nokia_2009 <- nokia[, tail(Nokia, 1)]*(1+mean(dt$Nokia)*360)
nokia_2009
```

```
## [1] 28.07711
```

```
# Then Correct price should be
nokia_2008 <- nokia_2009/(1+mu_j*360)
nokia_2008
```

```
## [1] 24.27598
```

According to Cap-model one should have been willing to pay anything below ~24.3 for Nokia's stock in beginning of 2008. It is easy to look back in history and say that for one year investment period investor shouldn't have pay even half of that and for even longer period not even one tenth of that price. By the time of it's creation, Cap-model groundbreaking. Nevertheless empirical tests have shown that Cap-model doesn't do too good job explaining stock returns. We have more or less data from whole Finnish stock market from 2011 to 2020 in our famafrench-file. Let's test if Cap-model is dead for recent Finnish stock market data.

Fama-Macbeth

I use equally weighted portfolios.

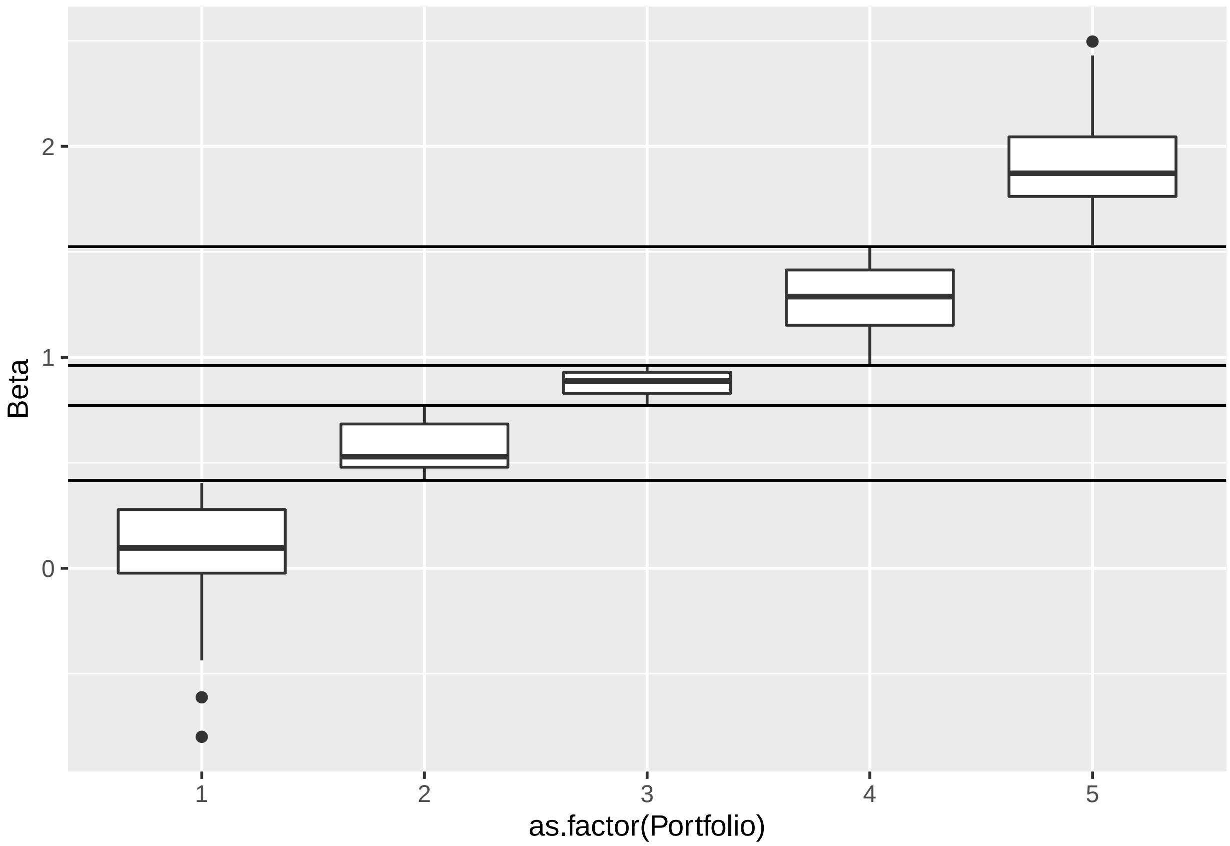
```
library(readxl)

# Read the excel file
data <- read_excel("extdata/famafrench2.xlsx")
data2 <- read_excel("extdata/factors.xlsx")

omxh <- data[, c(1, seq(2, 427, by = 3))]
r_f2 <- data.table(Dates = data2$Date, RF = data2$RF)

colnames(omxh) <- sub("_.*", "",colnames(omxh))

omxh <- as.data.table(melt(omxh, id.var = "Dates", value.name = "Price",
                      variable.name = "Company"))

# Calculate returns
omxh[, Return := as.numeric(Price)/shift(as.numeric(Price))-1, by = Company]
omxh <- drop_na(omxh)

# Calculate market returns
market_return3 <- merge(omxh[, .N, by = Dates], omxh[, sum(Return), by = Dates],
                        by = "Dates")
market_return3[, Market_Return := V1*1/N]

omxh <- merge(omxh, market_return3[, c(1, 4)], by = "Dates")
omxh[, Dates := as.Date(Dates)]
r_f2[, Dates := as.Date(Dates)]

# Dates for which we want to calculate betas
dates <- unique(omxh$Dates)[c(36:length(unique(omxh$Dates)))]

a <- data.table(Dates = as.Date(character()), Company = character(), V1 = numeric())

# Function to calculate betas for every stock
# Calculated beta using all data before given date
calculate_beta <- function(a, dt, date){
  d <- dt[Dates < date,
          if(length(Return) > 24){
            cov(Return, Market_Return)/var(Market_Return)
            }
          else {NA_real_},
          by = Company]
  d[, Dates := as.Date(date)]
  a <- rbind(a, d, fill=T)
  return(a)
}
betas <- rbindlist(lapply(dates, function(t){calculate_beta(a, omxh, t)}))
betas <- drop_na(betas)

omxh <- merge(omxh, betas, by = c("Dates", "Company"))
colnames(omxh)[ncol(omxh)] <- "Beta"

omxh <- merge(omxh, r_f2, by = "Dates")
```

```
omxh
```

```
##            Dates Company   Price      Return Market_Return       Beta     RF
##    1: 2014-02-28  FORTUM   17.16  0.07586207    0.02206663 0.58075608 0.0000
##    2: 2014-02-28   CGCBV   32.34  0.22639363    0.02206663 2.30391423 0.0000
##    3: 2014-02-28   CTY1S 13.1434  0.12647737    0.02206663 0.70163549 0.0000
##    4: 2014-02-28   ELISA   20.28  0.06568576    0.02206663 0.34354370 0.0000
##    5: 2014-02-28   FSKRS 15.8374  0.02598421    0.02206663 0.98509884 0.0000
##   ---
## 8567: 2020-03-31   LEM1S    23.9  0.00000000   -0.08156503 0.79483425 0.0012
## 8568: 2020-03-31   PIZZA    22.1  0.00000000   -0.08156503 0.04969345 0.0012
## 8569: 2020-03-31   CRA1V   13.25 -0.03284672   -0.08156503 2.35184728 0.0012
## 8570: 2020-03-31   SILMA    6.02  0.00000000   -0.08156503 0.22036430 0.0012
## 8571: 2020-03-31   TPS1V    4.65  0.00000000   -0.08156503 0.70382257 0.0012
```

Now that we have calculated betas for every stock we can start constructing portfolios.

```
# omxh[Beta < -2 | Beta > 5, Beta := NA]
# omxh <- drop_na(omxh)

# Perform allocation each February
omxh[month(Dates) == 2, Portfolio := cut(Beta, c(-Inf, quantile(Beta, probs = seq(0.2, 0.8, by = 0.2)),

# Add same portfolio indicator for other months also
omxh[, Portfolio := na.locf(Portfolio, na.rm = FALSE), by = Company]

# Drop companies that went public in the middle of year
# This should be changed that they would be dropped if there is less than 2 years data
omxh <- drop_na(omxh)

break_points <- omxh[month(Dates) == 2, quantile(Beta, probs = seq(0.2, 0.8, by =
                0.2)), by = Dates]

# Seems to be that our allocation worked
ggplot(data = omxh[Dates == "2016-02-29"], aes(as.factor(Portfolio), Beta)) +
  geom_boxplot() + geom_hline(yintercept = break_points[Dates =="2016-02-29"]$V1)
```




```r
# Calculate portfolio returns
portfolio_return <- omxh[, mean(Return), by = list(Portfolio, Dates)]
colnames(portfolio_return)[3] <- "Portfolio_Return"

# Cumulative returns
portfolio_return[, Cum_Return := cumprod(1+Portfolio_Return), by = Portfolio]

# Check if there are patterns in returns
portfolio_return[, mean(Portfolio_Return), by = Portfolio]
```

```
##    Portfolio           V1
## 1:         2 0.0069604861
## 2:         5 0.0008010542
## 3:         1 0.0046260328
## 4:         3 0.0063879148
## 5:         4 0.0094734351
```

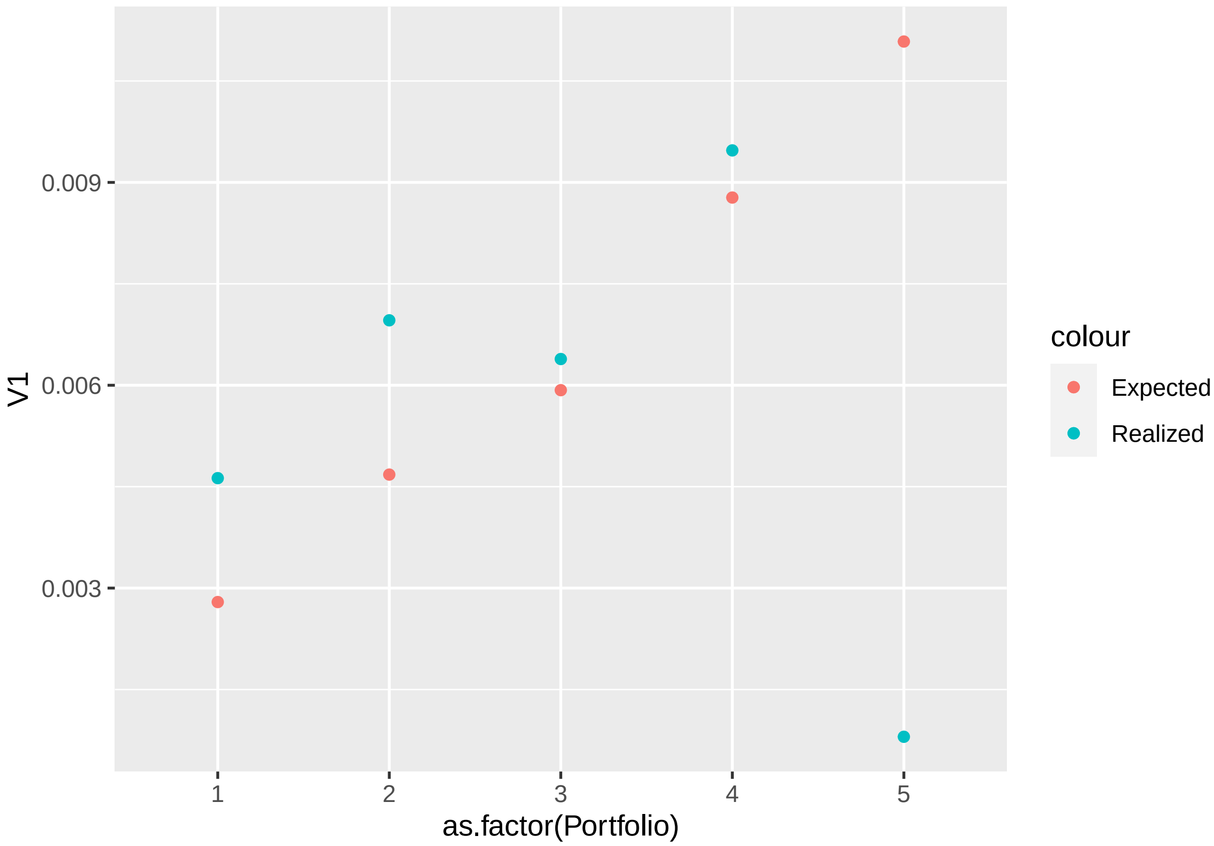
```r
ggplot(portfolio_return, aes(Dates, Cum_Return, color = as.factor(Portfolio))) + geom_line()
```

now calculate portfolio betas

```
library(tidyverse)

portfolio_return[, Portfolio_Beta := cov(Portfolio_Return,
  market_return3[Dates >=
  "2014-02-28"]$Market_Return)/ var(market_return3[Dates >=
  "2014-02-28"]$Market_Return), by = Portfolio]

market_return3[, Dates := as.Date(Dates)]

portfolio_return <- merge(portfolio_return, market_return3[, .(Dates, Market_Return)], by = "Dates") %>%

portfolio_return[, Expected_Return := mean(RF) + Portfolio_Beta*(mean(Market_Return) - mean(RF)), by =

portfolio_return[, .(mean(Expected_Return), mean(Portfolio_Return)), by = Portfolio]
```

```
##    Portfolio          V1           V2
## 1:         2 0.004678799 0.0069604861
## 2:         5 0.011083458 0.0008010542
## 3:         1 0.002793379 0.0046260328
## 4:         3 0.005927371 0.0063879148
## 5:         4 0.008776273 0.0094734351
```

```
ggplot(portfolio_return[, .(mean(Expected_Return), mean(Portfolio_Return)), by = Portfolio], aes(as.fac
```




```r
omxh <- merge(omxh, portfolio_return, by = c("Dates", "Portfolio"))

# Run a cross sectional regression for each date
cross_sect <- lapply(dates, function(t){lm(data = omxh[Dates == t], Return ~ Portfolio_Beta)})


coefficients <- lapply(cross_sect, function(t){coef(t)})
coefficients <- as.data.table(do.call(rbind, coefficients))
```

According to Fama and MacBeth (1973) we get our time series estimates and their t-statistics by:

begin

$$\overline{\gamma}_j = \frac{1}{T}\sum_{t=1}^{T}\gamma_{j,t}$$

$$t - stat(\overline{\gamma}_j) = \frac{\overline{\gamma}_j}{\sigma(\overline{\gamma}_j)/\sqrt{n}}$$

end

```r
n <- nrow(coefficients)

coefficients[, .(mean(`(Intercept)`), mean(Portfolio_Beta))]
```

```
##              V1           V2
## 1: 0.007332042 -0.001597963
```

```r
coefficients[, .(mean(`(Intercept)`)/(sd(`(Intercept)`)/sqrt(n)),
                 mean(Portfolio_Beta)/(sd(Portfolio_Beta)/sqrt(n)))]
```

```
##          V1         V2
## 1: 2.112105 -0.3442311
```