PCA Analysis of the J200 Top 40

Wouter Bezuidenhout

```
library(pacman)
```

```
## Warning: package 'pacman' was built under R version 4.0.2
```

```
pacman::p_load("tidyverse", "fmxdat", "devtools", "tbl2xts", "lubridate",
"readr", "PerformanceAnalytics", "tidyr", "FactoMineR", "factoextra", "rmsfuns")
list.files('/code/', full.names = T, recursive = T) %>% as.list() %>% walk(~source(.))
```

PCA Application to J200

I start by doing some data wrangling. I construct a simple-weighted return per day using the weights of the index. This seemed intuitive to me as converting from daily simple returns to log returns was challenging without exact price data. I mean-centre the data and convert it to a wide format. The code for the wrangling is below.

```
T40 <- read_rds("data/T40.rds") %>% select(-J400, -Short.Name, -Sector, -Index_Name) %>%
mutate(Tickers =gsub(" SJ Equity", "", Tickers)) %>%
rename(Weights = J200) %>% mutate(WRet = Weights*Return) %>%
select(-Weights, - Return) %>% rename(return = WRet) %>%
group_by(Tickers) %>%
mutate(return = return - mean(return)) %>% ungroup()

return_mat <- T40 %>% spread(Tickers, return)

# I have calculated a weighted simple return per day.
```

Next, I use Nico's function to impute values to replace NAs. I tried to source this from my code folder but ran into an error, so I do this manually.

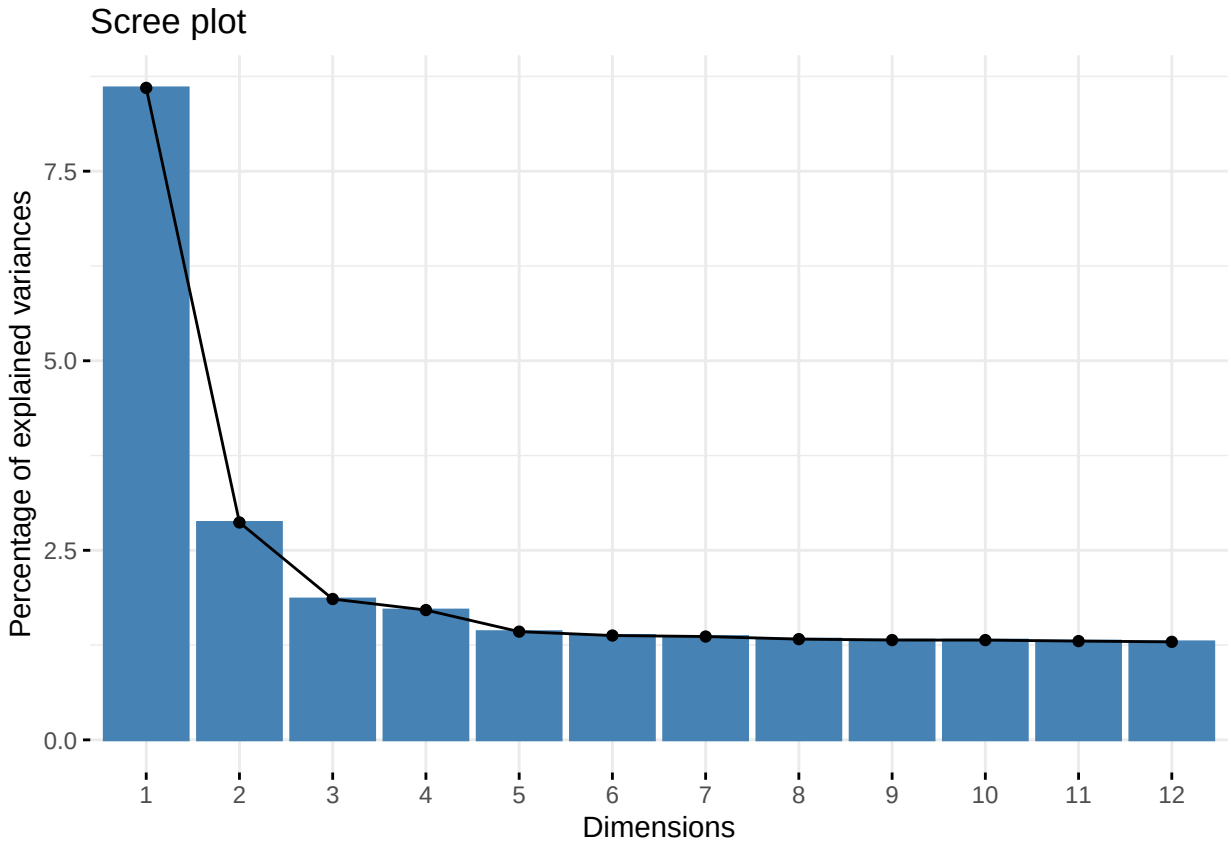
```
impute_missing_returns <- function(return_mat, impute_returns_method = "NONE",
Seed = 1234){
# Make sure we have a date column called date:
if( !"date" %in% colnames(return_mat) ) stop("No 'date' column
provided in return_mat. Try again please.")

# Note my use of 'any' below...
# Also note that I 'return' return_mat - which stops the function and returns return_mat.
if( impute_returns_method %in% c("NONE", "None", "none") ) {
if( any(is.na(return_mat)) ) warning("There are missing values in the return matrix
Consider maybe using impute_returns_method =
'Drawn_Distribution_Own' / 'Drawn_Distribution_Collective'")
return(return_mat)
}

if( impute_returns_method  == "Average") {
```

```
return_mat <-
return_mat %>% gather(Stocks, Returns, -date) %>%
group_by(date) %>%
mutate(Avg = mean(Returns, na.rm=T)) %>%
mutate(Avg = coalesce(Avg, 0)) %>%
ungroup() %>%
mutate(Returns = coalesce(Returns, Avg)) %>% select(-Avg) %>%
spread(Stocks, Returns)

    } else

if( impute_returns_method  == "Drawn_Distribution_Own") {

set.seed(Seed)
N <- nrow(return_mat)
return_mat <- left_join(return_mat %>% gather(Stocks, Returns, -date),
return_mat %>% gather(Stocks, Returns, -date) %>% group_by(Stocks) %>%
do(Dens = density(.$Returns, na.rm=T)) %>%
ungroup() %>% group_by(Stocks) %>% # done to avoid warning.
do(Random_Draws = sample(.$Dens[[1]]$x, N, replace = TRUE, prob=.$Dens[[1]]$y)),
by = "Stocks") %>%
group_by(Stocks) %>% mutate(Row = row_number()) %>%
mutate(Returns = coalesce(Returns, Random_Draws[[1]][Row])) %>%
select(-Random_Draws, -Row) %>% ungroup() %>% spread(Stocks, Returns)

        } else

if( impute_returns_method  == "Drawn_Distribution_Collective") {
set.seed(Seed)
NAll <- nrow(return_mat %>% gather(Stocks, Returns, -date))

return_mat <-
 bind_cols(
return_mat %>% gather(Stocks, Returns, -date),
return_mat %>% gather(Stocks, Returns, -date) %>%
do(Dens = density(.$Returns, na.rm=T)) %>%
do(Random_Draws = sample(.$Dens[[1]]$x, NAll, replace = TRUE,
prob=.$Dens[[1]]$y)) %>% unnest(Random_Draws)) %>%
mutate(Returns = coalesce(Returns, Random_Draws)) %>%
select(-Random_Draws) %>% spread(Stocks, Returns)

} else

if( impute_returns_method  == "Zero") {
warning("This is probably not the best idea but who am I to judge....")
                    return_mat[is.na(return_mat)] <- 0

} else
stop("Please provide a valid impute_returns_method method.
Options include:\n'Average', 'Drawn_Distribution_Own',
'Drawn_Distribution_Collective' and 'Zero'.")
}
```

Next, I turn to the PCA calculations. The challenge here is that the constituents of the top 40 change over

time. I calculate that over the sample period there were 92 stocks in the top 40. My approach is to include all 92 stocks, and then to see which PCAs drive these stocks best. I argue this is intuitive because if a stock only featured in the top40 for 1 year and never again, then it is unlikely that it will explain most of the variation of the other stocks.

```
options(scipen = 999)
return_mat <- impute_missing_returns(return_mat,
impute_returns_method = "Drawn_Distribution_Collective",
Seed = as.numeric(format( Sys.time(), "%Y%d%H%M")))

# Drop date column for this...
return_mat_Nodate <- data.matrix(return_mat[, -1])

# METHODS
Sigma <- RiskPortfolios::covEstimation(return_mat_Nodate)
Mu <- RiskPortfolios::meanEstimation(return_mat_Nodate)

pca <- prcomp(return_mat_Nodate, center = TRUE, scale. = TRUE)
scree_1 <- fviz_screeplot(pca, ncp = 12)
scree_1
```



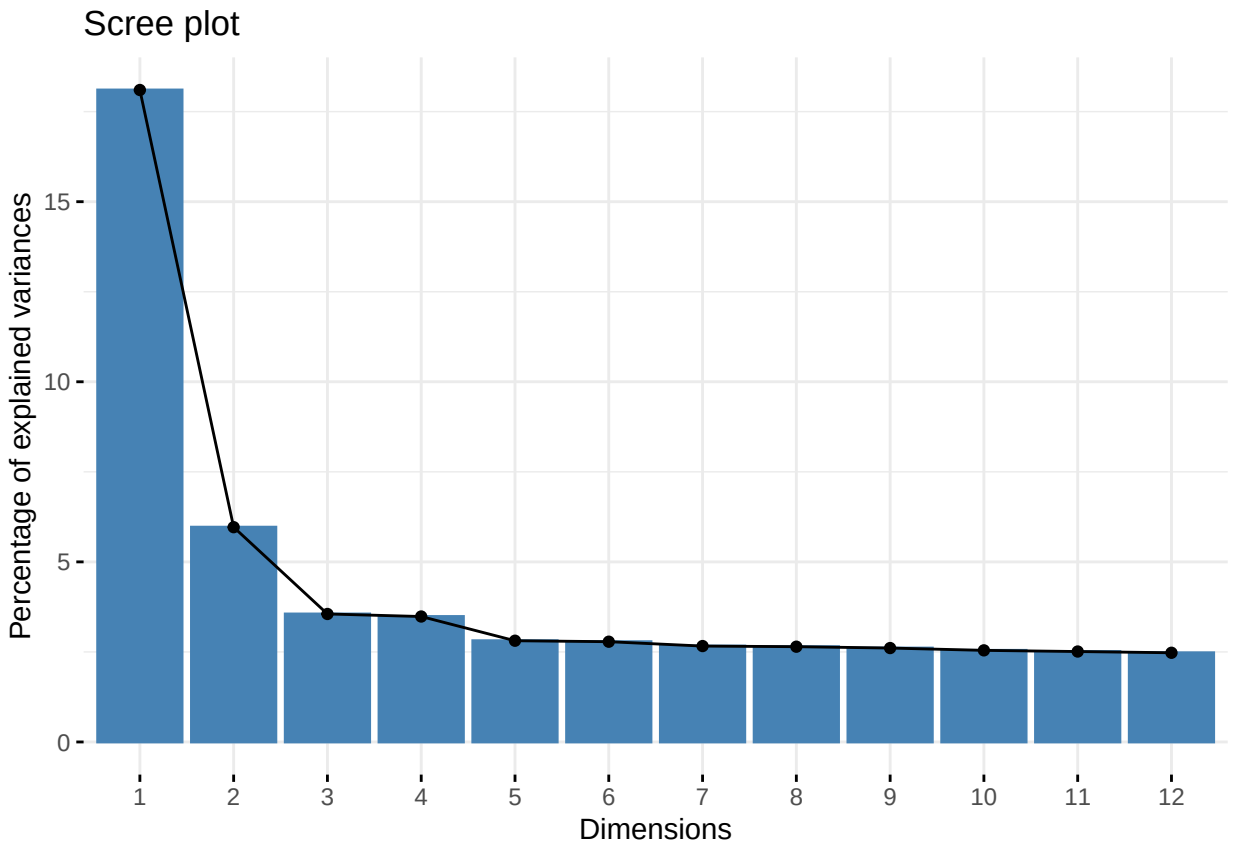
I see that the top 40 PCAs explain 58 percent of the variance of the 92 stocks. The scree plot above is the best visual illustration of the contribution of the top 10 PCA factors to explaining the variation. The top PCA factor explains 8,5 percent. In retrospect, a more focused approach with fewer PCAs would be more intuitive.

More Focused PCA

I take a snapshot of the top40 on the 1st of January 2020, and then I use these 41 stocks (41, not 40), to create a PCA overtime. I implement the exact procedure that I did in the previous PCA using Nico's function to impute missing values. My results show that I have 41 PCA factors, where the 80 percent of the variation is explained using the best 25 PCA factors and 50 percent of the variation is explained using the best 12 PCA factors.

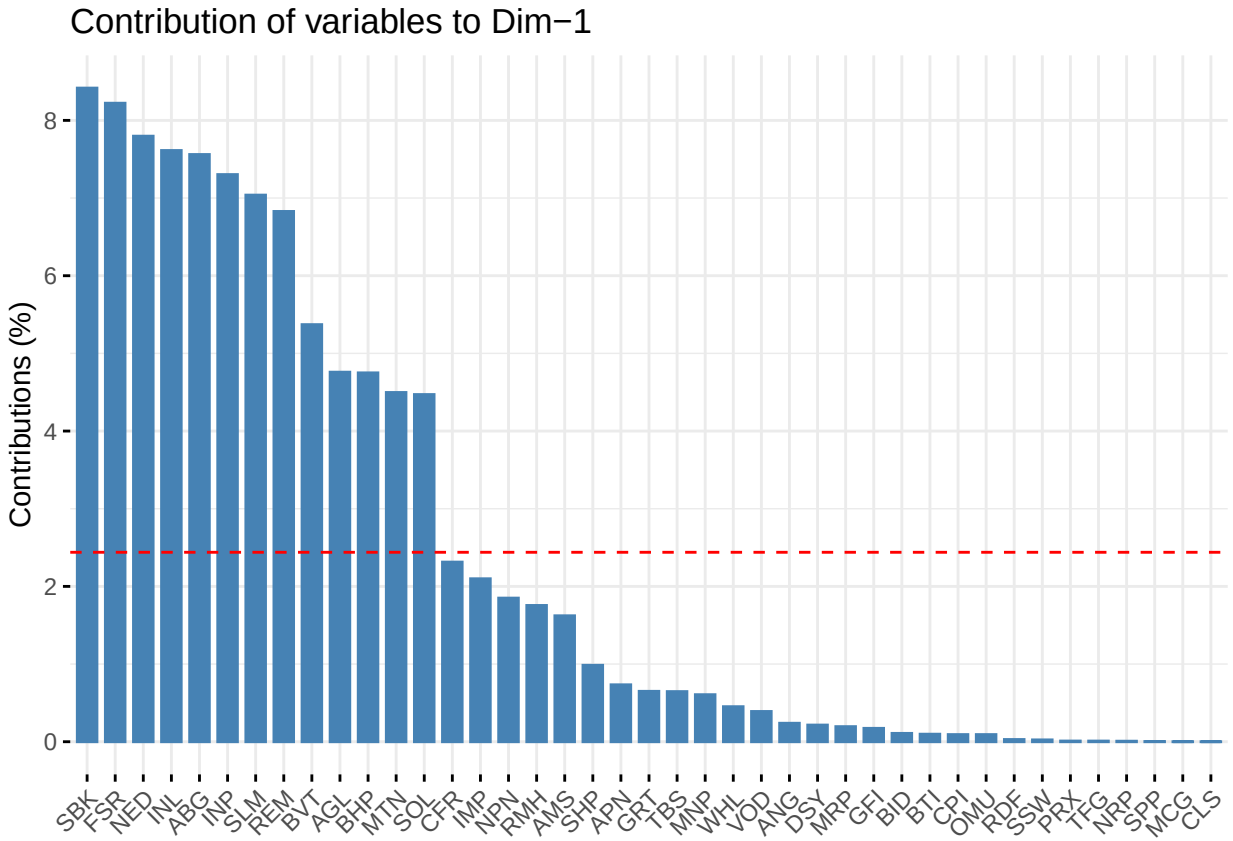
The scree plot below shows that the

```
Top40_today <- read_rds("data/T40.rds") %>%
select(-J400, -Short.Name, -Sector, -Index_Name) %>%
mutate(Tickers =gsub(" SJ Equity", "", Tickers)) %>%
filter(date > ymd(20200101)) %>%
filter(date < ymd(20200103)) %>%
pull(Tickers)

T40_new <- read_rds("data/T40.rds") %>%
select(-J400, -Short.Name, -Sector, -Index_Name) %>%
mutate(Tickers =gsub(" SJ Equity", "", Tickers)) %>%
rename(Weights = J200) %>% mutate(WRet = Weights*Return) %>%
select(-Weights, - Return) %>% rename(return = WRet) %>%
filter(Tickers %in% Top40_today) %>% group_by(Tickers) %>%
mutate(return = return - mean(return)) %>% ungroup()

return_mat_new <- T40_new %>% spread(Tickers, return)

return_mat_new <- impute_missing_returns(return_mat_new,
impute_returns_method = "Drawn_Distribution_Collective",
Seed = as.numeric(format( Sys.time(), "%Y%d%H%M")))

# Drop date column for this...
return_mat_new_Nodate <- data.matrix(return_mat_new[, -1])

# METHODS
Sigma <- RiskPortfolios::covEstimation(return_mat_Nodate)
Mu <- RiskPortfolios::meanEstimation(return_mat_Nodate)
# summary(pca_new)
pca_new <- prcomp(return_mat_new_Nodate, center = TRUE, scale. = TRUE)

scree_2 <- fviz_screeplot(pca_new, ncp = 12)
scree_2
```



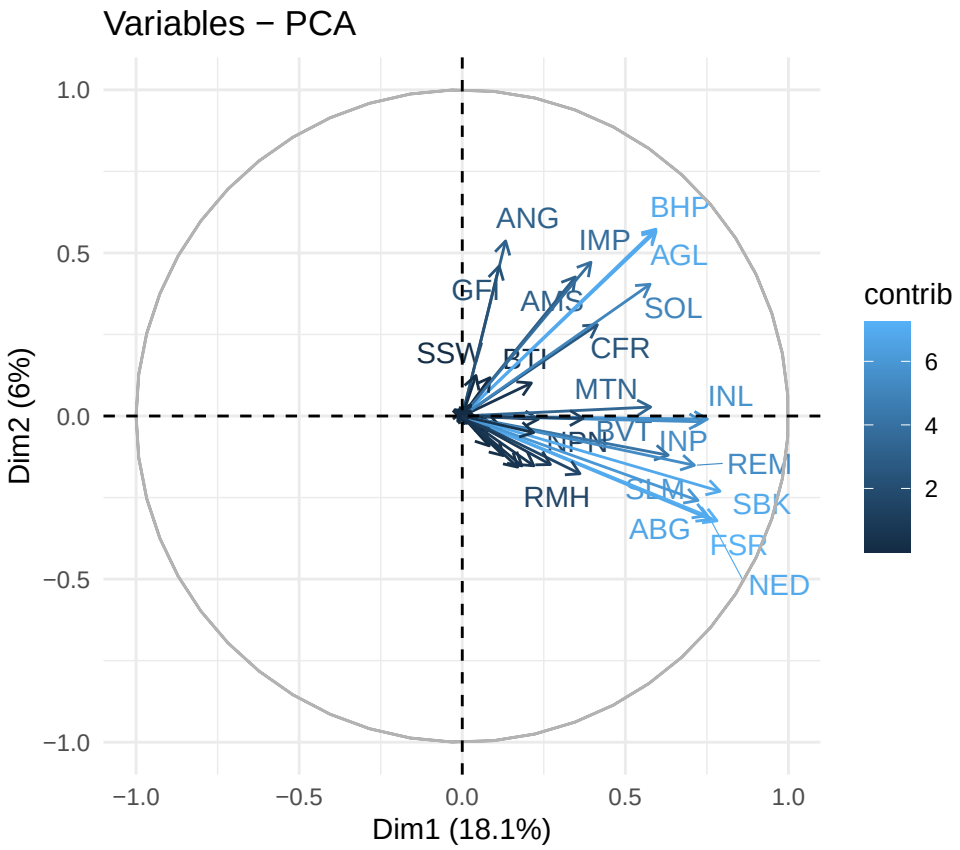
Lastly, the figure below shows how the individual stocks contribute to the first PCA factor. This first PCA factor explains 18 percent of the movement of the top 40 index as I have captured it. Interestingly, in this PCA factor, one can see the largest contributors are SBK (Standard Bank), FSR (FirstRand), NED (Nedbank), INL (Investec), ABG (Absa), INP (Investec), SLM (Sanlam), and REM (Remgro). These are all part of financials, and are mostly banks, except Remgro, Sanlam and parts of Investec. This is an intuitive finding, because, if correlated financials constitute a large part of the top 40, then using these as PCA's to explain the variation of the top 40.

```
fviz_contrib(pca_new, choice = "var", axes = 1)
```



The following figure shows which stocks contribute to PCA 1 and PCA 2. This is a visualization of the same argument that I made above. The stocks drive largely in the same direction, North-East,

```
fviz_pca_var(pca_new, col.var = "contrib", repel = T) + theme_minimal()
```


Variables − PCA
1.0
0.5
0.0
−0.5
−1.0
Dim2 (6%)
−1.0
−0.5
0.0
0.5
1.0
Dim1 (18.1%)
ANG
BHP
IMP
AGL
GFI
AMS
SOL
SSW
BTI
CFR
MTN
INL
RVT
INP
REM
RMH
SLM
SBK
ABG
FSR
NED
contrib
6
4
2