Final Project

Xie zejian Ma jiahui

Contents

1 Optimal portfolio via a simple BARRA model

1.1 Step 1

Read data:

```r
data <- read_csv("finalproj.csv",
                 col_types = cols(
  DATE=col_date(format="%Y%m%d"),
  PERMNO=col_integer()
))

factor <- read_csv("riskfactor.csv",
                    col_types = cols(
  date=col_date(format="%Y%m%d")
))
```

Apply standardize:

```r
scale2 <- function(x, na.rm = TRUE) (x - mean(x, na.rm = na.rm)) / sd(x, na.rm)

outlier.rm <- function(vec){
    scale <- quantile(vec, c(.01,.99,.5),na.rm = TRUE)
    min_out <- scale[[1]]
    max_out <- scale[[2]]
    mean_out <- scale[[3]]

    func <- function(val){
        if(is.na(val)){
            return(mean_out)
        }
        else if(val<min_out){
            return(min_out)
        }else if(val>max_out){
            return(min_out)
        }else{
            return(val)
        }
    }

    return(map(vec,func) %>% as.numeric())

}
```

```r
beg <- "1963-07-01"
end <- ymd(beg) %m+% months(120)

data_train <-  data %>%
    filter(DATE< end & DATE>beg) %>%
    mutate_at(c("logme","logbeme","r_2_12","gp","invest_asset"), outlier.rm) %>%
    mutate_at(c("logme","logbeme","r_2_12","gp","invest_asset"), scale2)
```

```r
data_train
```

```
## # A tibble: 175,629 x 9
##    DATE       PERMNO   exret mkt_beta  logme logbeme r_2_12     gp invest_asset
##    <date>      <int>   <dbl>    <dbl>  <dbl>   <dbl>  <dbl>  <dbl>        <dbl>
##  1 1963-07-31  10006 -0.0479     1.05  0.426   0.889   1.52 -0.773         1.99
```

```
##  2 1963-07-31  10014 -0.0027    0.707 -1.10   0.783  -0.170 -0.401        1.79
##  3 1963-07-31  10030 -0.0663    0.570  0.154  0.186   0.369 -0.140       -0.167
##  4 1963-07-31  10057 -0.0967    1.14  -0.451  0.186   1.31  -0.140       -0.167
##  5 1963-07-31  10102 -0.019     0.964  0.815  0.377   0.194 -0.262        0.188
##  6 1963-07-31  10137 -0.0027    0.649  1.16  -0.0794  0.255 -1.02        -0.275
##  7 1963-07-31  10145  0.0205    1.10   1.75   0.0602  0.592 -0.347        0.796
##  8 1963-07-31  10153 -0.0843    0.997  0.520  1.58    0.645 -0.753       -0.369
##  9 1963-07-31  10161 -0.0967    0.934  1.26   0.0115  0.645 -0.687        0.132
## 10 1963-07-31  10188 -0.0601    1.56  -0.532  0.944   0.469  0.566       -1.68
## # ... with 175,619 more rows
```

```
theta <- data_train %>%
    group_by(DATE) %>%
    do(tidy(lm(exret ~ mkt_beta+logme+logbeme+r_2_12+gp+invest_asset-1, data = .))) %>%
    pull(estimate) %>%
    matrix(ncol = 6,byrow=TRUE) %>%
    as_tibble() %>%
    `colnames<-`(c("mkt-beta", "logme", "logbeme", "momentum", "gp","ia")) %>%
    as.matrix()
```

1.2 Step 2

```
Time <- 120

theta.bar <- t(theta) %*% ones(Time,1)/Time
```

1.3 Step 3

```
proj_one <- ones(Time,Time)/Time

df <- theta %>% t %>%  dim %>% diff

theta.cov <- t(theta) %*%  (diag(Time)-proj_one) %*% theta / df
```

1.4 Step 4

Select PERMNO that meet requirements:

```
stk.no <- data %>%
    filter(DATE<(end%m+%months(1)) & DATE>end) %>%
    drop_na() %>%
    pull(PERMNO)

var_eta <- function(group,r){
    eta_i <- group$.resid

    return(sum((eta_i-mean(eta_i))^2/(length(eta_i)-6)))

}
```

The remaining stocks are given by:

```
stk.list <- data_train %>%
    filter(PERMNO %in% stk.no) %>%
    group_by(PERMNO) %>%
    filter(n()>=100) %>%
    ungroup() %>%
    select(PERMNO) %>%
    distinct() %>%
    pull(PERMNO)
```

and the variance is

```
eta.cov <- data_train %>%
    group_by(DATE) %>%
    do(augment_columns(lm(exret ~ mkt_beta+logme+logbeme+r_2_12+gp+invest_asset-1, data = .),
                       data = .)) %>%
    select(DATE,PERMNO,.resid) %>%
    filter(PERMNO %in% stk.list) %>%
    group_by(PERMNO) %>%
    group_map(var_eta) %>%
    as.numeric() %>%
    diag()
```

1.5 Step 5

```
X <- data %>%
    filter(DATE<(end%m+%months(1)) & DATE>end) %>%
    filter(PERMNO %in% stk.list) %>%
    mutate_at(c("logme","logbeme","r_2_12","gp","invest_asset"), outlier.rm) %>%
    mutate_at(c("logme","logbeme","r_2_12","gp","invest_asset"), scale2) %>%
    select(c("mkt_beta","logme","logbeme","r_2_12","gp","invest_asset")) %>%
    as.matrix()
```

forecast return is:

```
r.pred <- X %*% theta.bar
```

while the real return is

```
r_real <- data %>%
    filter(DATE<(end%m+%months(1)) & DATE>end) %>%
    filter(PERMNO %in% stk.list) %>%
    pull(exret)
```

1.6 Step 6

```r
r.cov <- X %*% theta.cov %*% t(X) + eta.cov
```

1.7 Step 7a

The weight assign to each asset is given by:

```r
N <- dim(r.cov)[[1]]

e <- ones(N,1)

omega <- r.cov %*% e / (t(e) %*% r.cov %*% e )[[1,1]]
```

and the return is:

```r
t(omega) %*% r_real
```

```
##             [,1]
## [1,] 0.09981045
```

1.8 Step 7b

As hint, the parameters are given by

```r
H <- 3*r.cov
f <- -r.pred %>% as.numeric()
beq <- c(0,0,0)
Aeq <- data %>%
    filter(DATE<(end%m+%months(1)) & DATE>end) %>%
    filter(PERMNO %in% stk.list) %>%
    mutate_at(c("logme","logbeme","r_2_12","gp","invest_asset"), outlier.rm) %>%
    mutate_at(c("logme","logbeme","r_2_12","gp","invest_asset"), scale2) %>%
    mutate(e=1) %>%
    select(c("e","mkt_beta","logme")) %>%
    as.matrix() %>% t()
lb <- rep(-.02,length(r.pred))
ub <- rep(.02,length(r.pred))
```

```r
solution <- quadprog(H, f, NULL, NULL, Aeq, beq, lb, ub)

portfolio <- solution$x
```

```r
t(portfolio) %*% r_real
```

```
##              [,1]
## [1,] -0.07358881
```

1.9 Step 8

Construct the time windows for map.

```
windows <- seq(from = ymd('1963-07-01'), to = ymd('2005-01-01'), by='month')
```

Combine step 1-7 to a function:

```
mv_portfolio <- function(beg){
    end <- beg %m+% months(120)

    data_train <-  data %>%
        filter(DATE< end & DATE>beg) %>%
        mutate_at(c("logme","logbeme","r_2_12","gp","invest_asset"), outlier.rm) %>%
        mutate_at(c("logme","logbeme","r_2_12","gp","invest_asset"), scale2)

    theta <- data_train %>%
        group_by(DATE) %>%
        do(tidy(lm(exret ~ mkt_beta+logme+logbeme+r_2_12+gp+invest_asset-1, data = .))) %>%
        pull(estimate) %>%
        matrix(ncol = 6,byrow=TRUE) %>%
        as_tibble() %>%
        `colnames<-`(c("mkt-beta", "logme", "logbeme", "momentum", "gp","ia")) %>%
        as.matrix()

    Time <- 120

    theta.bar <- t(theta) %*% ones(Time,1)/Time

    proj_one <- ones(Time,Time)/Time

    df <- theta %>% t %>%  dim %>% diff

    theta.cov <- t(theta) %*%  (diag(Time)-proj_one) %*% theta / df

    stk.no <- data %>%
        filter(DATE<(end%m+%months(1)) & DATE>end) %>%
        drop_na() %>%
        pull(PERMNO)

    stk.list <- data_train %>%
        filter(PERMNO %in% stk.no) %>%
        group_by(PERMNO) %>%
        filter(n()>=100) %>%
        ungroup() %>%
        select(PERMNO) %>%
        distinct() %>%
        pull(PERMNO)

    eta.cov <- data_train %>%
        group_by(DATE) %>%
        do(augment_columns(lm(exret ~ mkt_beta+logme+logbeme+r_2_12+gp+invest_asset-1, data = .),
                           data = .)) %>%
        select(DATE,PERMNO,.resid) %>%
```

```
        filter(PERMNO %in% stk.list) %>%
        group_by(PERMNO) %>%
        group_map(var_eta) %>%
        as.numeric() %>%
        diag()

    X <- data %>%
        filter(DATE<(end%m+%months(1)) & DATE>end) %>%
        filter(PERMNO %in% stk.list) %>%
        mutate_at(c("logme","logbeme","r_2_12","gp","invest_asset"), outlier.rm) %>%
        mutate_at(c("logme","logbeme","r_2_12","gp","invest_asset"), scale2) %>%
        select(c("mkt_beta","logme","logbeme","r_2_12","gp","invest_asset")) %>%
        as.matrix()

    r.pred <- X %*% theta.bar

    r_real <- data %>%
        filter(DATE<(end%m+%months(1)) & DATE>end) %>%
        filter(PERMNO %in% stk.list) %>%
        pull(exret)

    r.cov <- X %*% theta.cov %*% t(X) + eta.cov

    N <- dim(r.cov)[[1]]

    e <- ones(N,1)

    omega <- r.cov %*% e / (t(e) %*% r.cov %*% e )[[1,1]]

    pb$tick()$print()

    return(t(omega) %*% r_real)
}
cstr_portfolio <- function(beg){
    end <- beg %m+% months(120)

    data_train <-  data %>%
        filter(DATE< end & DATE>beg) %>%
        mutate_at(c("logme","logbeme","r_2_12","gp","invest_asset"), outlier.rm) %>%
        mutate_at(c("logme","logbeme","r_2_12","gp","invest_asset"), scale2)

    theta <- data_train %>%
        group_by(DATE) %>%
        do(tidy(lm(exret ~ mkt_beta+logme+logbeme+r_2_12+gp+invest_asset-1, data = .))) %>%
        pull(estimate) %>%
        matrix(ncol = 6,byrow=TRUE) %>%
        as_tibble() %>%
        `colnames<-`(c("mkt-beta", "logme", "logbeme", "momentum", "gp","ia")) %>%
        as.matrix()

    Time <- 120

    theta.bar <- t(theta) %*% ones(Time,1)/Time
```

```r
proj_one <- ones(Time,Time)/Time

df <- theta %>% t %>%  dim %>% diff

theta.cov <- t(theta) %*%  (diag(Time)-proj_one) %*% theta / df

stk.no <- data %>%
    filter(DATE<(end%m+%months(1)) & DATE>end) %>%
    drop_na() %>%
    pull(PERMNO)

stk.list <- data_train %>%
    filter(PERMNO %in% stk.no) %>%
    group_by(PERMNO) %>%
    filter(n()>=100) %>%
    ungroup() %>%
    select(PERMNO) %>%
    distinct() %>%
    pull(PERMNO)

eta.cov <- data_train %>%
    group_by(DATE) %>%
    do(augment_columns(lm(exret ~ mkt_beta+logme+logbeme+r_2_12+gp+invest_asset-1, data = .),
                       data = .)) %>%
    select(DATE,PERMNO,.resid) %>%
    filter(PERMNO %in% stk.list) %>%
    group_by(PERMNO) %>%
    group_map(var_eta) %>%
    as.numeric() %>%
    diag()

X <- data %>%
    filter(DATE<(end%m+%months(1)) & DATE>end) %>%
    filter(PERMNO %in% stk.list) %>%
    mutate_at(c("logme","logbeme","r_2_12","gp","invest_asset"), outlier.rm) %>%
    mutate_at(c("logme","logbeme","r_2_12","gp","invest_asset"), scale2) %>%
    select(c("mkt_beta","logme","logbeme","r_2_12","gp","invest_asset")) %>%
    as.matrix()

r.pred <- X %*% theta.bar

r_real <- data %>%
    filter(DATE<(end%m+%months(1)) & DATE>end) %>%
    filter(PERMNO %in% stk.list) %>%
    pull(exret)

r.cov <- X %*% theta.cov %*% t(X) + eta.cov

H <- 3*r.cov
f <- -r.pred %>% as.numeric()
A <- 0
b <- 0
beq <- c(0,0,0)
```

```
    Aeq <- data %>%
        filter(DATE<(end%m+%months(1)) & DATE>end) %>%
        filter(PERMNO %in% stk.list) %>%
        mutate_at(c("logme","logbeme","r_2_12","gp","invest_asset"), outlier.rm) %>%
        mutate_at(c("logme","logbeme","r_2_12","gp","invest_asset"), scale2) %>%
        mutate(e=1) %>%
        select(c("e","mkt_beta","logme")) %>%
        as.matrix() %>% t()
    lb <- rep(-.02,length(r.pred))
    ub <- rep(.02,length(r.pred))

    solution <- quadprog(H, f, NULL, NULL, Aeq, beq, lb, ub)

    portfolio <- solution$x

    pb$tick()$print()

    return(t(portfolio) %*% r_real)
}
```

```
pb <- progress_estimated(2*length(windows))

mv_r <- map(windows,mv_portfolio) %>% as.numeric()

cstr_r <- map(windows,cstr_portfolio) %>% as.numeric()
```

```
tibble(
    r1e=mv_r,
    r2e=cstr_r
) %>%
    mutate(
        rme=factor %>%
                filter(date>"1973-07-01") %>%
                mutate(mkt=Mkt_RF-RF) %>%
                pull(mkt),
        r3e=rme+r2e,
    ) %>% write_csv("returns.csv")
```

2 Performance Evaluation

```
pfm <- read_csv("returns.csv") %>%
    mutate(rf=factor %>%
                filter(date>"1973-07-01") %>%
                pull(RF),
           r1=r1e+rf,
           r3=r3e+rf,
           r2=r2e+rf
    )
```

2.1 Annualized average

```
func <- function(vec){
  mean(vec)*12
}

knitr::kable(
    pfm %>% summarise(across(c('r1e','r3e','rme'),func)),
             longtable=TRUE)
```

r1e	r3e	rme
0.1290001	0.535452	0.0182549

2.2 Annualized standard deviations

```
func <- function(vec){
  sd(vec)*sqrt(12)
}

knitr::kable(
    pfm %>% summarise(across(c('r1e','r3e','rme'),func)),
             longtable=TRUE)
```

r1e	r3e	rme
0.2186389	0.5422801	0.1610633

2.3 Annualized Sharpe ratios

```
func <- function(vec){
    mean(vec)/sd(vec)*sqrt(12)
}

knitr::kable(
    pfm %>% summarise(across(c('r1e','r3e','rme'),func)),
             longtable=TRUE)
```

r1e	r3e	rme
0.5900143	0.9874086	0.1133399

2.4 Annualized alpha and t-stats

```
func <- function(vec){
    lms <- lm(vec~pfm$rme) %>% summary()
    alpha <- lms$coefficients['(Intercept)','Estimate']
```

```r
    c(alpha*12,lms$coefficients['(Intercept)','t value'])
}

tbl <- pfm %>% summarise(across(c('r1e','r3e'),func))
rownames(tbl) <- c('alpha','t')

knitr::kable(tbl,
             longtable=TRUE,
             row.names = TRUE)
```

	r1e	r3e
alpha	0.1074116	0.5208572
t	6.4427718	6.3663222

2.5 CAPM beta

```r
func <- function(vec){
    lms <- lm(vec~pfm$rme) %>% summary()
    b <- lms$coefficients['pfm$rme','Estimate']
    c(b,b*sd(pfm$rme)*sqrt(12))
}

tbl <- pfm %>% summarise(across(c('r1e','r3e'),func))
rownames(tbl) <- c('beta','sys vol')

knitr::kable(tbl,
             longtable=TRUE,
             row.names = TRUE)
```

	r1e	r3e
beta	1.1826086	0.7995027
sys vol	0.1904749	0.1287706

2.6 Annualized idiosyncratic volatility

```r
func <- function(vec){
    lms <- lm(vec~pfm$rme) %>% summary()
    sd(lms$residuals)*sqrt(12)
}

tbl <- pfm %>% summarise(across(c('r1e','r3e'),func))

knitr::kable(tbl,
             longtable=TRUE)
```

r1e	r3e
0.1073418	0.5267692

2.7 Goodness of fit

```
func <- function(vec){
    lms <- lm(vec~pfm$rme) %>% summary()
    lms$r.squared
}

tbl <- pfm %>% summarise(across(c('r1e','r3e'),func))

knitr::kable(tbl,
             longtable=TRUE)
```

r1e	r3e
0.7589632	0.0563879

2.8 Information ratio

```
func <- function(vec){
    lms <- lm(vec~pfm$rme) %>% summary()
    alpha <- lms$coefficients['(Intercept)','Estimate']
    alpha/sd(lms$residuals)

}

tbl <- pfm %>% summarise(across(c('r1e','r3e'),func))

knitr::kable(tbl,
             longtable=TRUE)
```

r1e	r3e
0.2888629	0.2854353

2.9 Maximal Drawdown

```
func <- function(vec){
  cumwl <- cumprod(1+vec)
  dd <- drawdown(cumwl,FALSE,FALSE)
  max(dd)
}

tbl <- pfm %>%
    summarise(across(c('r1','r3'),func))

knitr::kable(tbl,
             longtable=TRUE)
```

r1	r3
170.8243	8544606

In percentage:

```
func <- function(vec){
  cumwl <- cumprod(1+vec)
  dd <- drawdown(cumwl,TRUE,FALSE)
  max(dd)
}

tbl <- pfm %>%
    summarise(across(c('r1','r3'),func))

knitr::kable(tbl,
             longtable=TRUE)
```

r1	r3
0.5994997	0.8188984

The cumulative wealth series of the two strategies:

```
x=1:nrow(pfm)
y=cumprod(1+pfm$r1)
ggplot(data= NULL, aes(x = x, y = y)) +
  geom_point() +
  geom_smooth()
```

```
x=1:nrow(pfm)
y=cumprod(1+pfm$r3)
ggplot(data= NULL, aes(x = x, y = y)) +
  geom_point() +
  geom_smooth()
```

2.10 Maximal Recovery Period

```
func <- function(vec){
  cumwl <- cumprod(1+vec)
  dd <- drawdown(cumwl,FALSE,FALSE)
  rec <- which(dd==0)
  rec <- c(rec %>% diff(),length(vec)-rec[length(rec)])
  max(rec)
}
tbl <- pfm %>%
    summarise(across(c('r1','r3'),func))

knitr::kable(tbl,
             longtable=TRUE)
```

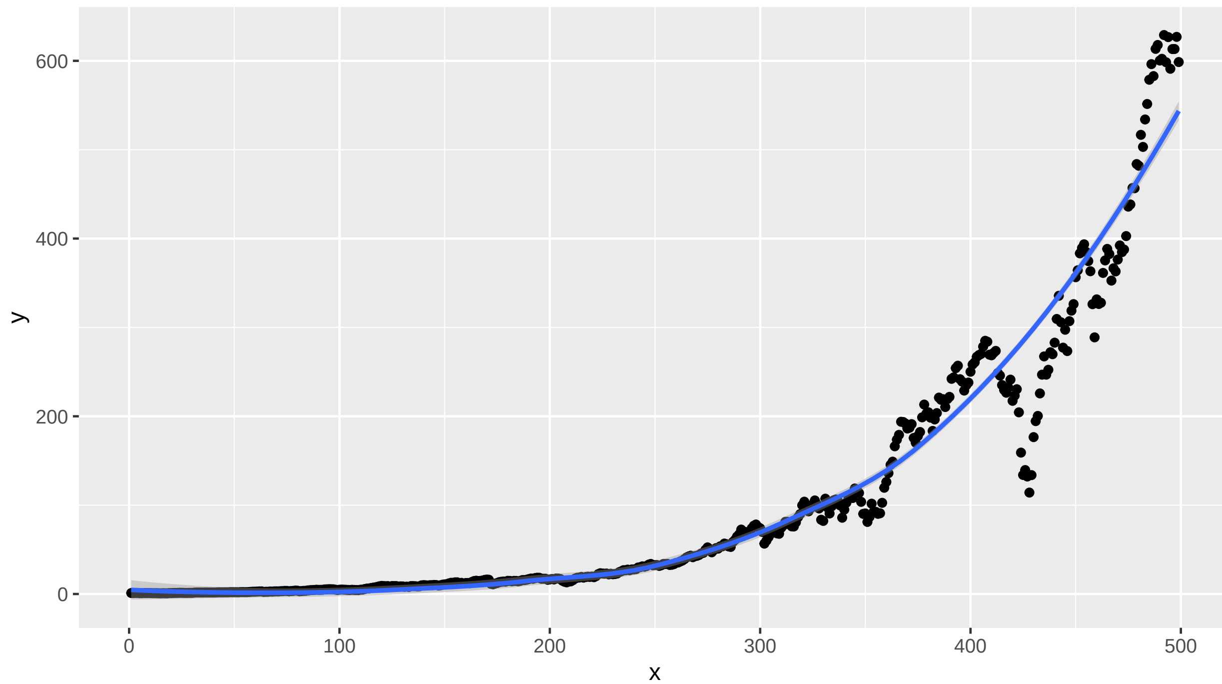

Figure 1: cumulative wealth series of r1

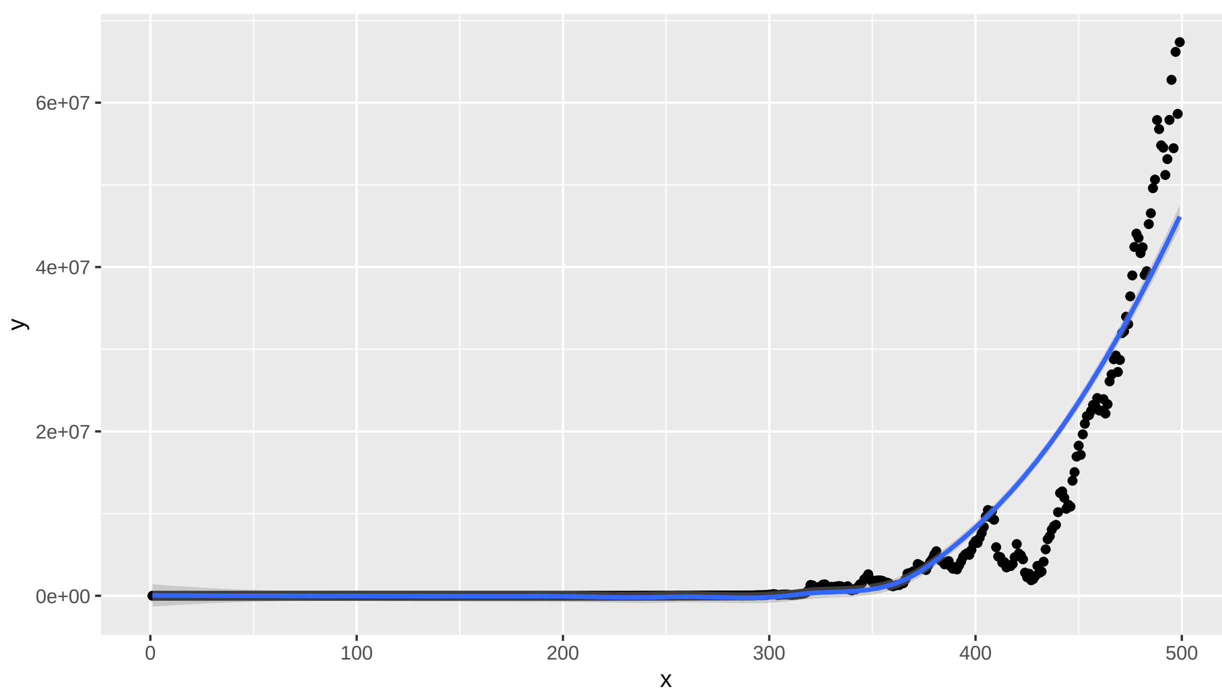

Figure 2: cumulative wealth series of r3

r1	r3
34	35