

Mean-Variance Portfolio Analysis

PART 1

ดึงข้อมูลราคาปิดสินทรัพย์ย้อนหลังรายเดือน 3 ตัว ได้แก่ Apple, Tesla และ Microsoft

```
tokk <- c("AAPL", "TSLA", "MSFT") #Choose ticker symbols.
numtk <- length(tokk);
ustart <- "2020-01-01"; #Define start date.
uend <- "2024-1-20" #Define end date.
all_dat <- list();
for(i in 1:numtk) {
  all_dat[[i]] <- xxx <- get.hist.quote(
    instrument = tokk[i],
    start=ustart, end=uend,
    quote = c("Close"), provider = "yahoo", compression = "m")
}
```

```
time series ends 2024-01-01
time series ends 2024-01-01
time series ends 2024-01-01
```

คำนวณ Covariance matrix ของหุ้นรายเดือนทั้ง 3 ตัว

```
covmat<-cov(returns) #variance-covariance matrix
covmat
```

	APP	TSLA	MSFT
APP	0.007736051	0.013443357	0.004259853
TSLA	0.013443357	0.043190727	0.007354834
MSFT	0.004259853	0.007354834	0.004341499

คำนวณค่าเฉลี่ยของผลตอบแทนของหุ้นทั้ง 3 ตัว (Expected return)

```
mearR <- colMeans(returns) #expected return
mearR
```

	APP	TSLA	MSFT
	0.01809190	0.03047624	0.01767179

คำนวณผลตอบแทน ความเสี่ยงที่ต่ำที่สุด และน้ำหนักของหุ้นแต่ละตัว ในพอร์ตพอร์ลิโอ

```
Gport <- globalMin.portfolio(er=mearR , cov.mat=covmat)
summary (Gport)
```

```
Portfolio expected return: 0.01595922
Portfolio standard deviation: 0.06236734
Portfolio weights:
  APP  TSLA  MSFT
0.2703 -0.1426 0.8723
```

สร้างกราฟ Efficient Frontier สำหรับพอร์ตที่มีสินทรัพย์ 3 ตัว

```
sapply(as.data.frame(returns), summary)
ns <- ncol(returns)
```

```
er <- apply(returns, 2, mean) #expected return
std <- apply(returns, 2, sd) #standard deviation
er
```

```
      APP      TSLA      MSFT
0.01809190 0.03047624 0.01767179
```

```
std
```

```
      APP      TSLA      MSFT
0.08795483 0.20782379 0.06589005
```

```
covm <- cov(returns) #variance-covariance matrix
#Global minimum variance (gmV) portfolio
Am <- rbind(2*covm, rep(1, ns))
Am <- cbind(Am, c(rep(1, ns), 0))
b <- c( rep(0, ns), 1)
w.gmv <- solve(Am) %*% b #last value is lambda constraint. Hence not relevant
w.gmv <- w.gmv[-(ns+1), ] #sum of weights are 1
sum(w.gmv)
```

```
[1] 1
```

```
var.gmv <- t(w.gmv) %*% covm %*% w.gmv
sigma <- sqrt(var.gmv)
ret.gmv <- t(w.gmv) %*% er
uv <- rep(1, ns) #unit vector
w.gmv2 <- (solve(covm) %*% uv) * c(solve( t(uv) %*% solve(covm) %*% uv))
#the results yield the same weight, expected return, and standard deviation
ret.gmv #return of portfolio
```

```
      [,1]
```

```
[1,] 0.01595922
```

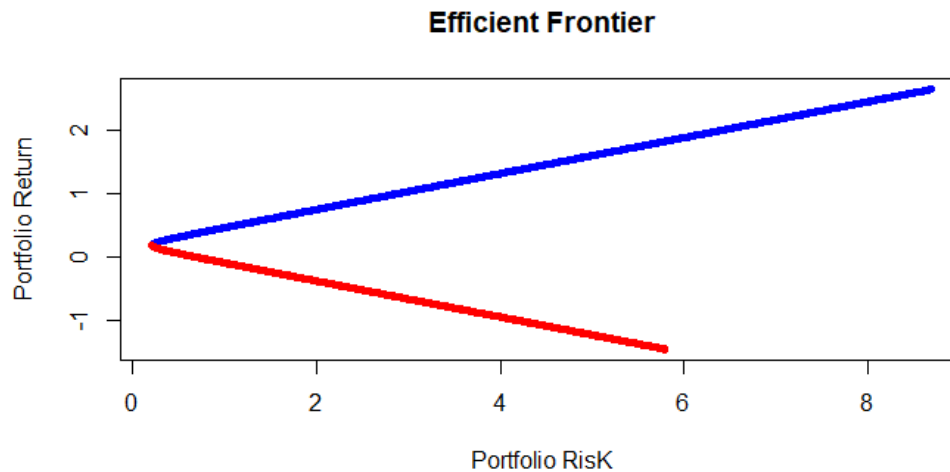
```
w1 <- .4
w <- seq(from = -5, to = 5, by = .01) #various combination of weights.
eff <- function(w1) {
  z <- w1 * w.gmv + (1 - w1) * w.ep
  c(ret = t(z) %*% er * 12, sd = sqrt(t(z) %*% covm %*% z * 12))
} #annualize expected return and standard deviation.
comb <- cbind(w, t(sapply(w, FUN = eff)))

efficient <- comb[, "ret"] > c(ret.gmv * 12) # month expected return and standard deviation.
xlim <- c(min(comb[, "sd"]), max(comb[, "sd"]))
ylim <- c(min(comb[, "ret"]), max(comb[, "ret"]))
col <- ifelse(efficient, "blue", "red")
plot(comb[, "ret"] ~ comb[, "sd"],
      col = col,
      xlim = xlim,
      ylim = ylim,
```

```

xlab = "Portfolio Risk",
ylab = "Portfolio Return",
pch = 16,
main = "Efficient Frontier",
cex = .7)

```



สร้าง Portfolio ใหม่ที่มีสินทรัพย์ 4 ตัว โดยเพิ่มหุ้น AMD

```

tckk <- c("AAPL", "TSLA", "MSFT", "AMD")
numtk <- length(tckk);
ustart <- "2020-01-01"; #Define start date.
uend <- "2024-1-20" #Define end date.
all_dat <- list();
for(i in 1:numtk) {
  all_dat[[i]] <- xxx <- get.hist.quote(
    instrument = tckk[i],
    start=ustart, end=uend,
    quote = c("Close"), provider = "yahoo", compression = "m")
}

```

```

time series ends  2024-01-01
time series ends  2024-01-01
time series ends  2024-01-01
time series ends  2024-01-01

```

```

APP <- as.xts(all_dat[[1]]) #Apple
TSLA <- as.xts(all_dat[[2]]) #Tesla
MSFT <- as.xts(all_dat[[3]]) #Microsoft
AMD <- as.xts(all_dat[[4]]) #AMD

```

คำนวณ Covariance matrix ของหุ้นรายเดือนทั้ง 4 ตัว

```

ret <- cbind(APP,TSLA,MSFT,AMD)
returns <- apply(log(ret), 2, diff)

```

```
colnames(returns) <- c("APP", "TSLA", "MSFT", "AMD")
colnames(returns)
#1 "APP" "TSLA" "MSFT" "AMD"
# plot return
Return.APP <- apply(log(APP), 2, diff)
Return.TSLA <- apply(log(TSLA), 2, diff)
Return.MSFT <- apply(log(MSFT), 2, diff)
Return.AMD <- apply(log(AMD), 2, diff)
covmat<-cov(returns) #variance-covariance matrix
covmat
```

```
      APP      TSLA      MSFT      AMD
APP  0.007736051 0.013443357 0.004259853 0.008078110
TSLA 0.013443357 0.043190727 0.007354834 0.015689452
MSFT 0.004259853 0.007354834 0.004341499 0.006528373
AMD  0.008078110 0.015689452 0.006528373 0.024491985
```

```
mearR <- colMeans(returns) #expected return
mearR
```

```
      APP      TSLA      MSFT      AMD
0.01809190 0.03047624 0.01767179 0.02649936
```

คำนวณผลตอบแทน ความเสี่ยงที่ต่ำที่สุด และน้ำหนักของหุ้นแต่ละตัว ในพอร์ตฟอร์ลิโอ

```
Gport <- globalMin.portfolio(er=mearR , cov.mat=covmat)
summary (Gport)
```

```
Portfolio expected return:    0.01513697
Portfolio standard deviation: 0.06065209
Portfolio weights:
      APP      TSLA      MSFT      AMD
0.2942  -0.1245   0.9509  -0.1205
```

```
#Global minimum variance (gmw) portfolio
Am <- rbind(2*covm, rep(1, ns))
Am <- cbind(Am, c(rep(1, ns), 0))
b <- c( rep(0, ns), 1)
w.gmv <- solve(Am) %*% b #last value is lambda constraint. Hence not relevant
w.gmv <- w.gmv[-(ns+1), ] #sum of weights are 1
sum(w.gmv)
```

```
[1] 1
```

```
var.gmv <- t(w.gmv) %*% covm %*% w.gmv
sigma <-sqrt(var.gmv)
ret.gmv <- t(w.gmv) %*% er
uv <- rep(1, ns) #unit vector
w.gmv2 <- (solve(covm) %*% uv) * c(solve( (t(uv) %*% solve(covm) %*% uv)))
#the results yield the same weight, expected return, and standard deviation
ret.gmv #return of portfolio
```

```
[,1]
[1,] 0.01513697
```

#portfolio expected return

```
ret.ep <- t(er) %*% w.ep
ret.ep
```

```
[,1]
[1,] 0.05
```

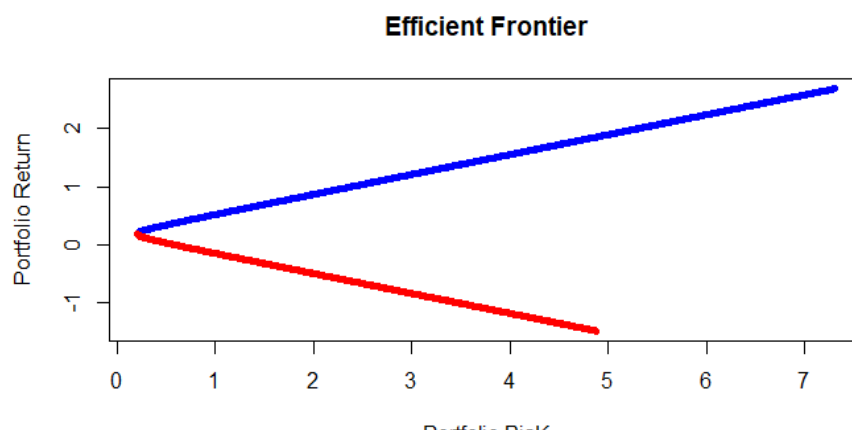
#portfolio standard deviation

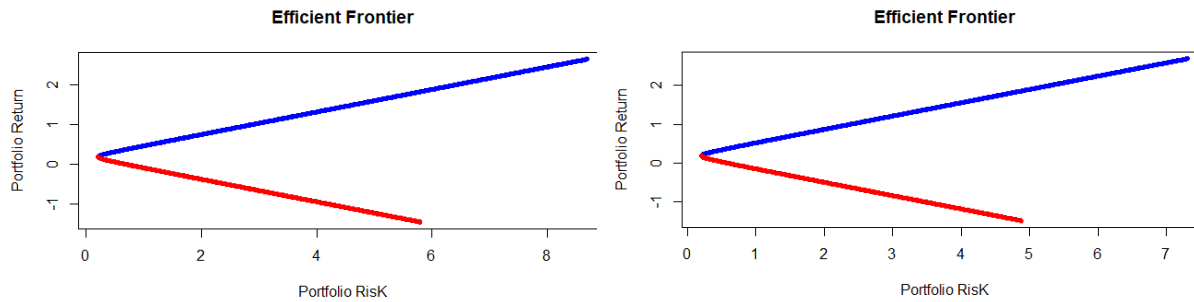
```
var.ep <- t(w.ep) %*% covm %*% w.ep
sigma_EP <- sqrt(var.ep)
sigma_EP
```

```
[,1]
[1,] 0.3565481
```

สร้างกราฟ Efficient Frontier สำหรับพอร์ตที่มีสินทรัพย์ 4 ตัว

```
w1 <- .4
w <- seq(from = -5, to = 5, by = .01) #various combination of weights.
eff <- function(w1) {
  z <- w1 * w.gmv + (1 - w1) * w.ep
  c(ret = t(z) %*% er * 12, sd = sqrt(t(z) %*% covm %*% z * 12))
} #annualize expected return and standard deviation.
comb <- cbind(w, t(sapply(w, FUN = eff)))
efficient <- comb[, "ret"] > c(ret.gmv * 12) # month expected return and standard deviation.
xlim <- c(min(comb[, "sd"]), max(comb[, "sd"]))
ylim <- c(min(comb[, "ret"]), max(comb[, "ret"]))
col <- ifelse(efficient, "blue", "red")
plot(comb[, "ret"] ~ comb[, "sd"],
      col = col,
      xlim = xlim,
      ylim = ylim,
      xlab = "Portfolio Risk",
      ylab = "Portfolio Return",
      pch = 16,
      main = "Efficient Frontier",
      cex = .7)
```





เปรียบเทียบ Efficient Frontier ระหว่างพอร์ตที่มีสินทรัพย์ 3 ตัวกับพอร์ตที่มีสินทรัพย์ 4 ตัว

ถ้าดูจากรูปจะเห็นว่ากราฟพอร์ตที่มีสินทรัพย์ 3 ตัว (ทางซ้าย) กับพอร์ตที่มีสินทรัพย์ 4 ตัว (ทางขวา) แทบไม่มีความต่างอย่างเห็นได้ชัดมาก จึงจะสรุปจากค่าที่คำนวณได้ดังนี้

- **พอร์ตที่มีสินทรัพย์ 3 ตัว (Apple, Tesla, Microsoft):**

- Portfolio expected return (Global Minimum Variance Portfolio): 0.01596
- Portfolio standard deviation: 0.06237

พอร์ตที่มีสินทรัพย์ 4 ตัว (เพิ่ม AMD):

- Portfolio expected return (Global Minimum Variance Portfolio): 0.01514
- Portfolio standard deviation: 0.06065

สรุปได้ว่าการเพิ่ม AMD ช่วยลด Portfolio Standard Deviation ของพอร์ตโฟลิโอขั้นต่ำ (Minimum Variance Portfolio) จาก 0.06237 เป็น 0.06065 AMD ทำให้ Efficient Frontier ขยับลงเพราะ expected return ลดลงและขยับไปทางซ้ายเพราะมีความเสี่ยงต่ำลง

Efficient Frontier ของพอร์ต 4 สินทรัพย์ให้ตัวเลือกที่ดีกว่าในแง่ของความเสี่ยงสำหรับผลตอบแทนที่ใกล้เคียงกัน เมื่อเทียบกับพอร์ต 3 สินทรัพย์

PART 2

ขั้นที่ 1 สร้างพอร์ตที่มีประสิทธิภาพซึ่งให้ผลตอบแทนที่คาดหวัง 5% ต่อปี

```
u0 <- 0.05 #desired expected return of the portfolio 5%
if(u0 < ret.gmv) {
  message("#u0 should be greater than return on minimum variance portfolio
")
}
M <- cbind(er, uv)
B <- t(M) %*% solve(covm) %*% M
mu.tilde <- c(u0, 1)
#weight of efficient portfolio
w.ep <- solve(covm) %*% M %*% solve(B) %*% mu.tilde
w.ep
```

```
[,1]
APP -3.976664
TSLA 2.655238
MSFT 2.321426
```

#portfolio expected return

```
ret.ep <- t(er) %*% w.ep
ret.ep
```

```
[,1]
[1,] 0.05
```

#portofolio standard deviation

```
var.ep <- t(w.ep) %*% covm %*% w.ep
sigma_EP <- sqrt(var.ep)
sigma_EP
```

```
[,1]
[1,] 0.4223308
```

ขั้นที่ 2 .กำหนดอัตราผลตอบแทนสินทรัพย์ปลอดความเสี่ยง (Risk-Free Rate) เท่ากับ 2% ต่อปี
ระบุ Tangency Portfolio (พอร์ตที่มี Sharpe Ratio สูงสุด) โดยใช้สินทรัพย์ทั้งสาม ระบุสัดส่วนการลงทุน
ของสินทรัพย์แต่ละตัวและคำนวณ Sharpe Ratio ของพอร์ตนี้

```
#SR = (mu - rf) / sigma
rfree <- .02 / 12 #define the risk-free interest rate = 2%
(w.sr <- solve(covm) %*% (er - rfree) / c(t(uv) %*% solve(covm) %*% (er - rfree))) #weights of optimal portfolio
```

```
[,1]
APP 0.044775638
TSLA 0.005943841
MSFT 0.949280521
```

```
(ret.sr <- t(w.sr) %*% er ) #expected return
```

```
[1]
[1,] 0.0177667

(var.sr <- t(w.sr)%*% covm %*% w.sr) #variance of portfolio

[1]
[1,] 0.004381586
```

คำนวณ Sharp-ratio

```
sr = (ret.sr - (0.02/12)) /sqrt(var.sr)

sr

[1]
[1,] 0.2432267
```

ขั้นที่ 3. ใช้สินทรัพย์ปลอดความเสี่ยง (ที่ให้ผลตอบแทน 2% ต่อปี) และ Tangency Portfolio ที่ได้จากขั้นตอนที่ 2 นำทฤษฎี Two-Fund Separation มาใช้เพื่อหาสัดส่วนที่เหมาะสมสำหรับสินทรัพย์แต่ละตัวในพอร์ตที่ให้ผลตอบแทนที่คาดหวัง 5% ต่อปี และระบุสัดส่วนการลงทุนของสินทรัพย์แต่ละตัว (รวมถึงสินทรัพย์ปลอดความเสี่ยง) อย่างชัดเจน

```
# คำนวณสัดส่วนการลงทุนใน Tangency Portfolio และสินทรัพย์ปลอดความเสี่ยง
wt <- (u0 - rfree) / (ret.sr - rfree) #สัดส่วนการลงทุนใน Tangency Portfolio
wt

[1]
[1,] 0.1552791

#สัดส่วนสินทรัพย์ปลอดความเสี่ยง
wf <- 1 - w_t
wf

[1] 0.8447208

# คำนวณสัดส่วนการลงทุนในสินทรัพย์แต่ละตัว
weights_portfolio <- w_t * c(w.sr["APP",1], w.sr["TSLA",1], w.sr["MSFT",1])
weights_portfolio

  APP      TSLA      MSFT
0.0069527245 0.0009229547 0.1474035024
```

ขั้นที่ 4 เปรียบเทียบพอร์ตที่ได้จากขั้นตอนที่ 3 (โดยใช้ทฤษฎี Two-Fund Separation) กับพอร์ตที่มีประสิทธิภาพที่ได้จากขั้นตอนที่ 1 อภิปรายความแตกต่างในเรื่องสัดส่วนการลงทุน ความเสี่ยง และลักษณะผลตอบแทน

การเปรียบเทียบระหว่างพอร์ตที่ได้จากขั้นตอนที่ 3 (ใช้ทฤษฎี Two-Fund Separation) กับพอร์ตที่มีประสิทธิภาพที่ได้จากขั้นตอนที่ 1 นั้นสามารถทำได้โดยพิจารณาจากหลายๆ ด้าน เช่น สัดส่วนการลงทุน ความเสี่ยง และลักษณะของผลตอบแทนของแต่ละพอร์ต

1. สัดส่วนการลงทุน:

พอร์ตที่ได้จากขั้นตอนที่ 1 (พอร์ตที่มีประสิทธิภาพ): สัดส่วนการลงทุนในแต่ละสินทรัพย์คือ:

- APP: -3.98%
- TSLA: 2.66%
- MSFT: 2.32%

พอร์ตนี้มุ่งเน้นไปที่ผลตอบแทนที่สูงขึ้นด้วยการลงทุนในสินทรัพย์ที่มีผลตอบแทนสูงและความเสี่ยงสูง แต่มีผลตอบแทนที่คาดหวัง 5% ต่อปี

พอร์ตที่ได้จากขั้นตอนที่ 3 (Two-Fund Separation): สัดส่วนการลงทุนคือ:

- สัดส่วนการลงทุนใน Tangency Portfolio: 15.53%
- สัดส่วนการลงทุนในสินทรัพย์ปลอดความเสี่ยง: 84.47%

พอร์ตนี้เน้นการจัดการความเสี่ยงด้วยการลงทุนในสินทรัพย์ปลอดความเสี่ยงถึง 84.47% และส่วนที่เหลือ 15.53% ลงใน Tangency Portfolio ที่ให้ผลตอบแทนสูงสุดต่อความเสี่ยงที่ยอมรับ (Sharpe Ratio) จากสินทรัพย์ทั้งสามในขั้นตอนที่ 2

2. ความเสี่ยง:

- พอร์ตที่มีประสิทธิภาพ: จากการคำนวณความเสี่ยง (Standard Deviation) ของพอร์ตนี้, ค่า Standard Deviation = 42.23% ซึ่งแสดงถึงความเสี่ยงที่ค่อนข้างสูง
- พอร์ต Two-Fund Separation: ความเสี่ยงของพอร์ตนี้จะต่ำกว่าพอร์ตที่มีประสิทธิภาพมาก เนื่องจากสัดส่วนของสินทรัพย์ปลอดความเสี่ยงสูงถึง 84.47% ทำให้พอร์ตนี้มีความเสี่ยงต่ำกว่ามากเมื่อเทียบกับพอร์ตที่มีประสิทธิภาพ

3. ผลตอบแทน:

- พอร์ตที่มีประสิทธิภาพ: ผลตอบแทนที่คาดหวังจากพอร์ตนี้คือ 5% ต่อปี
- พอร์ต Two-Fund Separation: ผลตอบแทนที่คาดหวังจากพอร์ตนี้คือประมาณ 1.78% ต่อปี ซึ่งต่ำกว่าพอร์ตที่มีประสิทธิภาพ แต่มีความเสี่ยงที่ต่ำกว่า

ขั้นที่ 5 วิเคราะห์ข้อดีและข้อเสียของแต่ละวิธีในบริบทของกลยุทธ์การลงทุน

1. พอร์ตที่มีประสิทธิภาพ

ข้อดี:

- ผลตอบแทนที่สูงกว่า: พอร์ตที่มีประสิทธิภาพมุ่งเน้นการให้ผลตอบแทนที่สูงถึง 5% ต่อปี ซึ่งเป็นการตั้งเป้าหมายที่ชัดเจนและสูงกว่าในระยะยาว
- การปรับแต่งพอร์ตให้เหมาะสม: การใช้ผลตอบแทนที่คาดหวัง 5% และการใช้สินทรัพย์ที่มีความเสี่ยงสูง (เช่น APP, TSLA) ทำให้พอร์ตมีการปรับแต่งให้เหมาะสมกับเป้าหมายของนักลงทุนที่ต้องการผลตอบแทนสูง

ข้อเสีย:

- พอร์ตนี้มีความเสี่ยงค่อนข้างสูง เนื่องจากการลงทุนในสินทรัพย์ที่มีความผันผวนสูง เช่น APP ทำให้พอร์ตอาจมีความผันผวนที่สูงกว่าพอร์ตที่เน้นสินทรัพย์ปลอดภัยความเสี่ยง
- พอร์ตนี้ไม่ได้นำสินทรัพย์ปลอดภัยความเสี่ยงมาใช้ในกลยุทธ์ ทำให้พอร์ตอาจจะมีความเสี่ยงมากเกินไปสำหรับนักลงทุนที่ไม่พร้อมรับความเสี่ยงสูง
- แม้ว่าเป้าหมายผลตอบแทนจะเป็น 5% แต่ในระยะสั้นหรือในช่วงตลาดที่ผันผวน ผลตอบแทนอาจจะไม่เป็นไปตามที่คาดการณ์ไว้

2. พอร์ต Two-Fund Separation

ข้อดี:

- ด้วยการใช้สินทรัพย์ปลอดภัยความเสี่ยง พอร์ตนี้จะมีความเสี่ยงต่ำมาก และช่วยป้องกันการขาดทุนที่อาจเกิดขึ้นจากการลงทุนในสินทรัพย์ที่มีความเสี่ยงสูง
- การลงทุนใน Tangency Portfolio ที่มี Sharpe Ratio สูงสุดทำให้พอร์ตนี้มีความเสี่ยงที่เหมาะสมและสามารถเพิ่มผลตอบแทนให้ได้ในระดับที่ยอมรับได้ โดยไม่ต้องรับความเสี่ยงสูงเกินไป
- พอร์ตนี้เหมาะสำหรับนักลงทุนที่ต้องการลดความผันผวนในพอร์ตและพร้อมที่จะยอมรับผลตอบแทนที่ต่ำกว่าในระยะสั้นเพื่อแลกกับความเสี่ยงที่ต่ำกว่าในระยะยาว

ข้อเสีย:

- ผลตอบแทนที่ต่ำ: การมีสัดส่วนการลงทุนในสินทรัพย์ปลอดภัยความเสี่ยงสูงทำให้ผลตอบแทนโดยรวมต่ำกว่าเมื่อเทียบกับพอร์ตที่มีประสิทธิภาพ