MGMTMFE 400 — Problem Set 7

Tiffany Tu

1 0. Setup

```
# Core packages
library(readr)
library(dplyr)
library(lubridate)
library(tidyr)
library(ggplot2)
library(purrr)
library(tibble)

TRADING_DAYS <- 252
WEEKS_PER_YEAR <- 52
```

2 0.1 Load data

```
# ===== S&P 500 daily total return data =====
# File: 500csv.csv, columns: caldt (m/d/Y), vwretd
sp <- read_csv("500csv.csv", col_types = cols())

sp <- sp |>
  mutate(date = mdy(caldt)) |>
  arrange(date) |>
  rename(r_sp = vwretd)

head(sp)
```

```
# A tibble: 6 x 3
  caldt           r_sp date
  <chr>          <dbl> <date>
1 1/3/1972  -0.00414  1972-01-03
2 1/4/1972   0.0042   1972-01-04
3 1/5/1972   0.00935  1972-01-05
4 1/6/1972   0.00427  1972-01-06
5 1/7/1972  -0.000263 1972-01-07
6 1/10/1972 -0.00131  1972-01-10
```

```r
names(sp)
```

```
[1] "caldt" "r_sp"  "date"
```

```r
# ===== Stock + daily rf data (CRSP-style long panel) =====
# File: lecture7p_2025.csv, columns: date (m/d/Y), ticker, ..., ret*, rf, ...
raw_daily <- read_csv("lecture7p_2025.csv", col_types = cols()) |>
  mutate(date = mdy(date)) |>
  arrange(date)

head(raw_daily)
```

```
# A tibble: 6 x 18
  date       ticker permno exchcd shrcd   prc shrout     mcap    vol   bid   ask
  <date>     <chr>   <dbl>  <dbl> <dbl> <dbl>  <dbl>    <dbl>  <dbl> <dbl> <dbl>
1 1989-12-29 INTC    59328      3    11  34.5 183789  6.34e 9 7.43e5  34.2  34.5
2 1989-12-29 JNJ     22111      1    11  59.4 333135  1.98e10 3.28e5  NA    NA
3 1989-12-29 LUV     58683      1    11  24    29258  7.02e 8 6.07e4  NA    NA
4 1989-12-29 MCD     43449      1    11  34.5 369994  1.28e10 4.92e5  NA    NA
5 1989-12-29 MSFT    10107      3    11  87    55676  4.84e 9 5.40e5  87    87.2
6 1990-01-02 INTC    59328      3    11  36   183789  6.62e 9 2.00e6  35.8  36
# i 7 more variables: ret...12 <dbl>, ret...13 <dbl>, ret_excess <dbl>,
#   mktrf <dbl>, smb <dbl>, hml <dbl>, rf <dbl>
```

```r
names(raw_daily)
```

```
 [1] "date"       "ticker"     "permno"     "exchcd"     "shrcd"
 [6] "prc"        "shrout"     "mcap"       "vol"        "bid"
[11] "ask"        "ret...12"   "ret...13"   "ret_excess" "mktrf"
[16] "smb"        "hml"        "rf"
```

```
# ---- Choose which return column to use dynamically ----
# We look for columns that start with "ret" but are not "ret_excess".
ret_candidates <- names(raw_daily)[grepl("^ret", names(raw_daily)) & names(raw_daily) != "re

ret_candidates
```

```
[1] "ret...12" "ret...13"
```

```
if (length(ret_candidates) == 0) {
  stop(
    paste(
      "Could not find a usable return column in lecture7p_2025.csv.",
      "Available columns are:",
      paste(names(raw_daily), collapse = ", ")
    )
  )
}

# Use the first such return column
ret_col <- ret_candidates[1]
ret_col
```

```
[1] "ret...12"
```

```
# Pivot to wide by ticker using that return column
stock_daily_long <- raw_daily |>
  select(date, ticker, all_of(ret_col)) |>
  rename(ret = all_of(ret_col))

stock_daily_wide <- stock_daily_long |>
  mutate(ticker = as.character(ticker)) |>
  pivot_wider(names_from = ticker, values_from = ret)

# Extract rf series (one per date)
if (!("rf" %in% names(raw_daily))) {
  stop(
    paste(
      "Couldn't find a risk-free (rf) column in lecture7p_2025.csv.",
      "Available columns are:",
      paste(names(raw_daily), collapse = ", ")
    )
```

```
  )
}

rf_series <- raw_daily |>
  select(date, rf) |>
  distinct()

# Merge rf into the wide stock data
daily <- stock_daily_wide |>
  left_join(rf_series, by = "date") |>
  arrange(date)

head(daily)
```

```
# A tibble: 6 x 7
  date          INTC     JNJ      LUV     MCD     MSFT       rf
  <date>       <dbl>   <dbl>    <dbl>   <dbl>    <dbl>    <dbl>
1 1989-12-29  0.0147   0.00423 -0.00518  0.0110  0.0205  0.000003
2 1990-01-02  0.0435   0.00842  0.0104   0.0109  0.0201  0.000003
3 1990-01-03 -0.0278   0.00418 -0.0206  -0.0108  0.00563 0.000003
4 1990-01-04  0.0214   0.00208  0.00526 -0.0181  0.0294  0.000003
5 1990-01-05 -0.00699 -0.0104  -0.00524 -0.0184 -0.0245  0.000003
6 1990-01-08  0.0141   0.0168   0.0105   0.0188  0.0153  0.000003
```

```
names(daily)
```

```
[1] "date" "INTC" "JNJ"  "LUV"  "MCD"  "MSFT" "rf"
```

---

3 1. Optimal allocation: S&P 500 vs bond (A = 4)

We treat the **bond** as investing at the **daily risk-free rate `rf`**.
We match S&P dates with the risk-free rate, compute daily excess returns, then annualize.

```
# Match S&P data with daily rf
sp_rf <- sp |>
  inner_join(daily |> select(date, rf), by = "date") |>
  mutate(
```

```r
    r_bond   = rf,              # bond = risk-free asset
    r_excess = r_sp - r_bond
  )

# Annual mean and variance of excess returns
mu_e_daily  <- mean(sp_rf$r_excess, na.rm = TRUE)
var_e_daily <- var(sp_rf$r_excess,  na.rm = TRUE)

mu_e_annual  <- TRADING_DAYS * mu_e_daily
var_e_annual <- TRADING_DAYS * var_e_daily

A <- 4

w_stock <- mu_e_annual / (A * var_e_annual)
w_bond  <- 1 - w_stock

w_stock
```

```
[1] 0.9075777
```

```r
w_bond
```

```
[1] 0.09242232
```

---

4 2. Weekly and annual statistics + correlation matrix

We form **weekly simple returns** for each stock and compute:

- weekly mean & standard deviation
- annualized mean and SD
- correlation matrix of weekly returns

```r
# Identify the five tickers of interest
tickers5 <- c("MSFT", "INTC", "LUV", "MCD", "JNJ")

# Sanity check: all tickers should exist as columns in 'daily'
missing_tickers <- setdiff(tickers5, names(daily))
missing_tickers  # should be character(0); if not, something is off with the tickers.
```

```
character(0)
```

```r
# 2.1 Construct weekly stock returns (simple compounding within each week)
weekly <- daily |>
  mutate(week = floor_date(date, unit = "week", week_start = 1)) |>
  group_by(week) |>
  summarize(
    across(all_of(tickers5),
           ~ prod(1 + .x, na.rm = TRUE) - 1),
    .groups = "drop"
  )

head(weekly)
```

```
# A tibble: 6 x 6
  week          MSFT     INTC      LUV      MCD      JNJ
  <date>       <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
1 1989-12-25  0.0205  0.0147  -0.00518  0.0110   0.00423
2 1990-01-01  0.0302  0.0290  -0.0104  -0.0362   0.00421
3 1990-01-08 -0.0391 -0.00704 -0.116   -0.0338  -0.0461
4 1990-01-15  0.0929  0.0745   0.0298  -0.00778  0.00879
5 1990-01-22 -0.0359  0.0231  -0.0520  -0.0196  -0.0784
6 1990-01-29  0.0331  0.00645  0.0793   0.0640   0.0331
```

```r
# 2.2 Weekly mean & sd for each stock, plus annualized versions
weekly_stats <- weekly |>
  summarize(across(all_of(tickers5),
                   list(mean = ~ mean(.x, na.rm = TRUE),
                        sd   = ~ sd(.x,   na.rm = TRUE))))

weekly_stats_long <- weekly_stats |>
  pivot_longer(
    cols       = everything(),
```

```
    names_to  = c("stock", "stat"),
    names_sep = "_",
    values_to = "value"
  ) |>
  pivot_wider(names_from = stat, values_from = value) |>
  mutate(
    mean_ann = mean * WEEKS_PER_YEAR,
    sd_ann   = sd   * sqrt(WEEKS_PER_YEAR)
  )

weekly_stats_long
```

```
# A tibble: 5 x 5
  stock    mean     sd mean_ann sd_ann
  <chr>   <dbl>  <dbl>    <dbl>  <dbl>
1 MSFT  0.00466 0.0401    0.243  0.289
2 INTC  0.00319 0.0499    0.166  0.360
3 LUV   0.00311 0.0475    0.162  0.342
4 MCD   0.00277 0.0306    0.144  0.220
5 JNJ   0.00246 0.0278    0.128  0.200
```

```
# 2.3 Weekly correlation matrix
cor_weekly <- weekly |>
  select(-week) |>
  cor(use = "pairwise.complete.obs")

cor_weekly
```

```
          MSFT      INTC       LUV       MCD       JNJ
MSFT 1.0000000 0.4848730 0.2841369 0.2800347 0.2716620
INTC 0.4848730 1.0000000 0.2929674 0.2457807 0.2043160
LUV  0.2841369 0.2929674 1.0000000 0.3096305 0.2399522
MCD  0.2800347 0.2457807 0.3096305 1.0000000 0.3238470
JNJ  0.2716620 0.2043160 0.2399522 0.3238470 1.0000000
```

---

5 3. Mean–variance frontier: Intel + Microsoft

We use **weekly** returns of MSFT & INTC, compute and Σ, and sweep weights in MSFT.

```r
# 3.1 Weekly returns for MSFT & INTC
weekly_IM <- weekly |>
  select(week, MSFT, INTC)

mu_IM <- colMeans(weekly_IM |> select(MSFT, INTC), na.rm = TRUE)
Sigma_IM <- cov(weekly_IM |> select(MSFT, INTC),
                use = "pairwise.complete.obs")

mu_IM
```

```
       MSFT        INTC
0.004663943 0.003187450
```

```r
Sigma_IM
```

```
             MSFT         INTC
MSFT 0.0016058072 0.0009696142
INTC 0.0009696142 0.0024902826
```

```r
# 3.2 Grid of portfolios using w_MSFT (w_INTC = 1 - w_MSFT)
w_grid <- seq(-0.5, 1.5, length.out = 400)

frontier_IM <- tibble(
  w_MSFT = w_grid,
  w_INTC = 1 - w_MSFT
) |>
  mutate(
    mean = w_MSFT * mu_IM["MSFT"] + w_INTC * mu_IM["INTC"],
    var  = w_MSFT^2 * Sigma_IM[1, 1] +
           w_INTC^2 * Sigma_IM[2, 2] +
           2 * w_MSFT * w_INTC * Sigma_IM[1, 2],
    sd   = sqrt(var)
  )

head(frontier_IM)
```

```
# A tibble: 6 x 5
  w_MSFT w_INTC    mean     var     sd
   <dbl>  <dbl>   <dbl>   <dbl>  <dbl>
1 -0.5      1.5  0.00245 0.00455 0.0675
```

```
2 -0.495   1.49 0.00246 0.00452 0.0673
3 -0.490   1.49 0.00246 0.00450 0.0671
4 -0.485   1.48 0.00247 0.00447 0.0669
5 -0.480   1.48 0.00248 0.00445 0.0667
6 -0.475   1.47 0.00249 0.00442 0.0665
```

```r
# 3.3 Minimum-variance portfolio (closed form)
one2 <- c(1, 1)
Sigma_IM_inv <- solve(Sigma_IM)
A_IM <- as.numeric(t(one2) %*% Sigma_IM_inv %*% one2)
w_mv_IM <- as.vector(Sigma_IM_inv %*% one2 / A_IM)

names(w_mv_IM) <- c("MSFT", "INTC")
w_mv_IM
```

```
     MSFT      INTC
0.7050376 0.2949624
```

```r
mu_mv_IM <- sum(w_mv_IM * mu_IM)
sd_mv_IM <- sqrt(as.numeric(t(w_mv_IM) %*% Sigma_IM %*% w_mv_IM))

mu_mv_IM
```

```
[1] 0.004228433
```

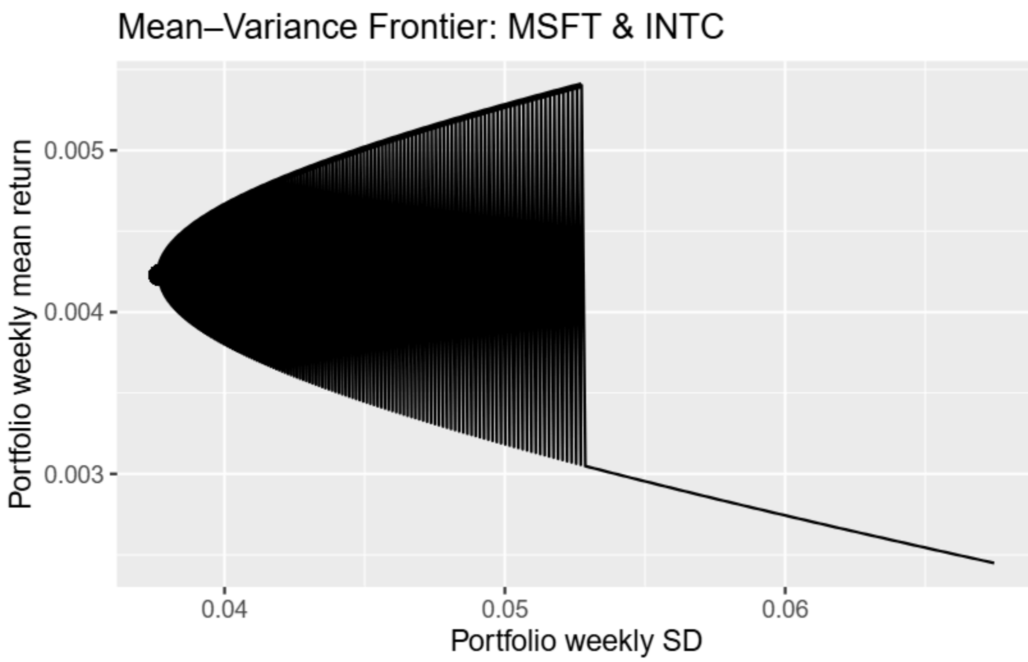
```r
sd_mv_IM
```

```
[1] 0.03765839
```

```r
# Mark efficient part (portfolios with mean >= min-variance mean)
frontier_IM <- frontier_IM |>
  mutate(efficient = mean >= mu_mv_IM)
```

```r
# 3.4 Plot MSFT-INTC frontier
ggplot(frontier_IM, aes(x = sd, y = mean)) +
  geom_line() +
  geom_line(data = subset(frontier_IM, efficient),
            linewidth = 1.1) +
  geom_point(aes(x = sd_mv_IM, y = mu_mv_IM),
```

```r
              size = 3) +
  labs(
    x = "Portfolio weekly SD",
    y = "Portfolio weekly mean return",
    title = "Mean-Variance Frontier: MSFT & INTC"
  )
```



---

6 4. Mean–variance frontier: all 5 stocks vs MSFT+INTC

We now include **all five stocks** and compute the multi-asset efficient frontier using the usual A,B,C,D formulas.

```r
# 4.1 Covariance and means for all 5 stocks (weekly)
assets_all <- weekly |>
  select(week, all_of(tickers5))

mu_all <- colMeans(assets_all |> select(-week), na.rm = TRUE)
Sigma_all <- cov(assets_all |> select(-week),
```

```
                 use = "pairwise.complete.obs")

mu_all
```

```
       MSFT        INTC         LUV         MCD         JNJ 
0.004663943 0.003187450 0.003110521 0.002774155 0.002464539 
```

```
Sigma_all
```

```
             MSFT         INTC          LUV          MCD          JNJ
MSFT 0.0016058072 0.0009696142 0.0005406880 0.0003429629 0.0003023738
INTC 0.0009696142 0.0024902826 0.0006942504 0.0003748527 0.0002832014
LUV  0.0005406880 0.0006942504 0.0022549916 0.0004493706 0.0003164942
MCD  0.0003429629 0.0003748527 0.0004493706 0.0009340649 0.0002749141
JNJ  0.0003023738 0.0002832014 0.0003164942 0.0002749141 0.0007715021
```

```
# 4.2 Efficient frontier for 5 stocks
Sigma_all_inv <- solve(Sigma_all)
one5 <- rep(1, length(mu_all))

A_all <- as.numeric(t(one5) %*% Sigma_all_inv %*% one5)
B_all <- as.numeric(t(one5) %*% Sigma_all_inv %*% mu_all)
C_all <- as.numeric(t(mu_all) %*% Sigma_all_inv %*% mu_all)
D_all <- A_all * C_all - B_all^2

mu_min_all <- min(mu_all)
mu_max_all <- max(mu_all)

mu_target_all <- seq(mu_min_all * 0.8,
                     mu_max_all * 1.2,
                     length.out = 200)

frontier_all <- map_dfr(mu_target_all, function(mu_p) {
  lambda1 <- (C_all - B_all * mu_p) / D_all
  lambda2 <- (A_all * mu_p - B_all) / D_all
  w <- Sigma_all_inv %*% (lambda1 * one5 + lambda2 * mu_all)
  tibble(
    mean = mu_p,
    sd   = sqrt(as.numeric(t(w) %*% Sigma_all %*% w))
  )
})
```

```r
# Minimum-variance portfolio for all 5
w_mv_all <- as.vector(Sigma_all_inv %*% one5 / A_all)
names(w_mv_all) <- names(mu_all)
w_mv_all
```

```
      MSFT       INTC        LUV        MCD        JNJ
0.11722289 0.04244688 0.05578919 0.32319786 0.46134318
```

```r
mu_mv_all <- sum(w_mv_all * mu_all)
sd_mv_all <- sqrt(as.numeric(t(w_mv_all) %*% Sigma_all %*% w_mv_all))

mu_mv_all
```

```
[1] 0.002889151
```

```r
sd_mv_all
```

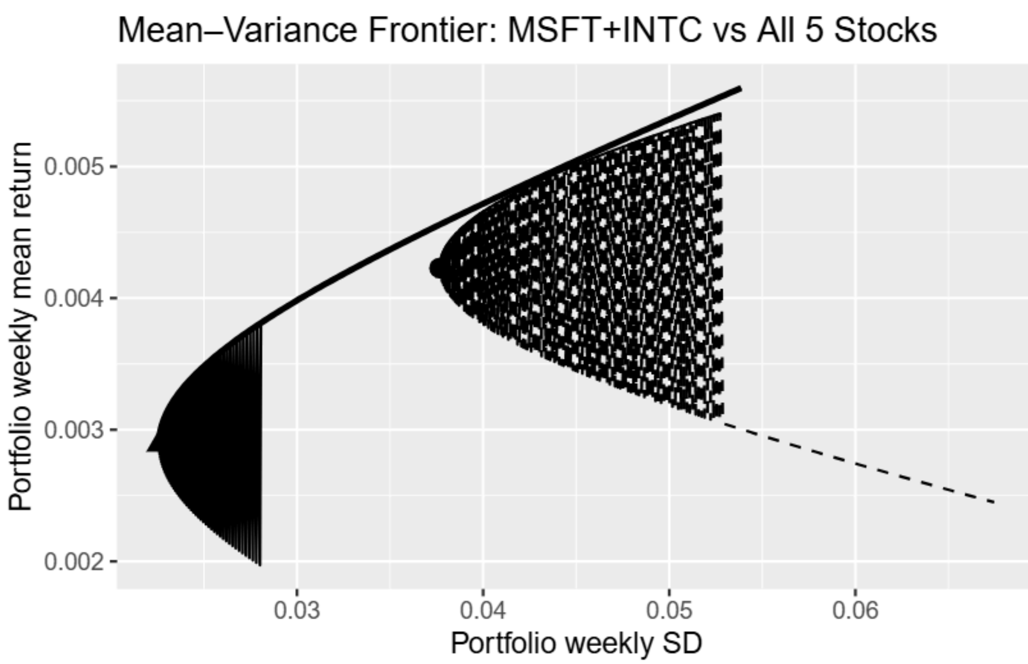
```
[1] 0.02258101
```

```r
frontier_all <- frontier_all |>
  mutate(efficient = mean >= mu_mv_all)
```

```r
# 4.3 Plot MSFT+INTC frontier vs all-5 frontier
frontier_IM_plot <- frontier_IM |>
  transmute(mean, sd, efficient_IM = efficient)

ggplot() +
  # 2-asset frontier (MSFT+INTC)
  geom_line(data = frontier_IM_plot,
            aes(x = sd, y = mean),
            linetype = "dashed") +
  geom_line(data = subset(frontier_IM_plot, efficient_IM),
            aes(x = sd, y = mean)) +
  geom_point(aes(x = sd_mv_IM,  y = mu_mv_IM),
             size = 3) +
  # 5-asset frontier
  geom_line(data = frontier_all,
            aes(x = sd, y = mean)) +
  geom_line(data = subset(frontier_all, efficient),
```

```r
            aes(x = sd, y = mean),
            linewidth = 1.1) +
  geom_point(aes(x = sd_mv_all, y = mu_mv_all),
             size = 3, shape = 17) +
  labs(
    x = "Portfolio weekly SD",
    y = "Portfolio weekly mean return",
    title = "Mean-Variance Frontier: MSFT+INTC vs All 5 Stocks"
  )
```



---

7 5. Tangent portfolios (with risk-free asset)

Now we incorporate the **weekly risk-free rate** and compute the **maximum-Sharpe (tangent) portfolios** for:

- MSFT + INTC only
- All 5 stocks

```r
# 5.1 Weekly risk-free returns from daily rf
weekly_rf <- daily |>
  select(date, rf) |>
  mutate(week = floor_date(date, unit = "week", week_start = 1)) |>
  group_by(week) |>
  summarize(
    rf_week = prod(1 + rf, na.rm = TRUE) - 1,
    .groups = "drop"
  )

# Combine weekly stock returns with weekly rf
weekly_full <- weekly |>
  inner_join(weekly_rf, by = "week")

head(weekly_full)
```

```
# A tibble: 6 x 7
  week          MSFT     INTC      LUV      MCD      JNJ    rf_week
  <date>       <dbl>    <dbl>    <dbl>    <dbl>    <dbl>      <dbl>
1 1989-12-25  0.0205  0.0147  -0.00518  0.0110   0.00423 0.00000300
2 1990-01-01  0.0302  0.0290  -0.0104  -0.0362   0.00421 0.0000120
3 1990-01-08 -0.0391 -0.00704 -0.116   -0.0338  -0.0461  0.0000150
4 1990-01-15  0.0929  0.0745   0.0298  -0.00778  0.00879 0.0000150
5 1990-01-22 -0.0359  0.0231  -0.0520  -0.0196  -0.0784  0.0000150
6 1990-01-29  0.0331  0.00645  0.0793   0.0640   0.0331  0.0000150
```

```r
# 5.2 Tangent portfolio for MSFT + INTC

IM_data <- weekly_full |>
  select(MSFT, INTC, rf_week)

mu_IM <- colMeans(IM_data[, c("MSFT", "INTC")], na.rm = TRUE)
Sigma_IM <- cov(IM_data[, c("MSFT", "INTC")],
                use = "pairwise.complete.obs")

rf_bar <- mean(IM_data$rf_week, na.rm = TRUE)  # avg weekly rf

Sigma_IM_inv <- solve(Sigma_IM)
excess_mu_IM <- mu_IM - rf_bar

w_t_IM_unnorm <- Sigma_IM_inv %*% excess_mu_IM
```

```r
w_t_IM <- as.vector(w_t_IM_unnorm / sum(w_t_IM_unnorm))
names(w_t_IM) <- c("MSFT", "INTC")

w_t_IM
```

```
      MSFT       INTC
0.93489939 0.06510061
```

```r
mu_T_IM <- sum(w_t_IM * mu_IM)
sd_T_IM <- sqrt(as.numeric(t(w_t_IM) %*% Sigma_IM %*% w_t_IM))
Sharpe_IM <- (mu_T_IM - rf_bar) / sd_T_IM

mu_T_IM
```

```
[1] 0.004567823
```

```r
sd_T_IM
```

```
[1] 0.03914224
```

```r
Sharpe_IM
```

```
[1] 0.1165704
```

```r
# 5.3 Tangent portfolio for all 5 stocks

mu_all <- colMeans(weekly_full[, tickers5],
                   na.rm = TRUE)
Sigma_all <- cov(weekly_full[, tickers5],
                 use = "pairwise.complete.obs")

Sigma_all_inv <- solve(Sigma_all)
one5 <- rep(1, length(mu_all))

excess_mu_all <- mu_all - rf_bar

w_t_all_unnorm <- Sigma_all_inv %*% excess_mu_all
w_t_all <- as.vector(w_t_all_unnorm / sum(w_t_all_unnorm))
names(w_t_all) <- names(mu_all)

w_t_all
```

```
      MSFT        INTC         LUV         MCD         JNJ
0.38919099 -0.01683036  0.05956721  0.27226444  0.29580772
```

```
mu_T_all <- sum(w_t_all * mu_all)
sd_T_all <- sqrt(as.numeric(t(w_t_all) %*% Sigma_all %*% w_t_all))
Sharpe_all <- (mu_T_all - rf_bar) / sd_T_all

mu_T_all
```

```
[1] 0.003431137
```

```
sd_T_all
```

```
[1] 0.02461142
```

```
Sharpe_all
```

```
[1] 0.1392094
```

---

8 6. Optimal overall mix given risk aversion A = 3.5

We now assume **risk aversion A = 3.5** and use the **all-5-stock tangent portfolio**.

For a mean–variance investor with a risk-free asset and a tangent portfolio, the optimal fraction of wealth invested in the tangent portfolio (call it `y_star`) is:

- proportional to the Sharpe ratio of the tangent portfolio, and
- inversely proportional to both risk aversion `A_opt` and the variance `sd_T_all^2`.

We compute it directly below, then back out the weights.

```
A_opt <- 3.5

y_star <- (mu_T_all - rf_bar) / (A_opt * sd_T_all^2)
y_star
```

```
[1] 1.616083
```

```
# Weights in risky stocks
w_risky_opt <- y_star * w_t_all
w_risky_opt
```

```
       MSFT        INTC         LUV         MCD         JNJ 
 0.62896513 -0.02719927  0.09626559  0.44000206  0.47804998 
```

```
# Weight in the risk-free asset
w_rf_opt <- 1 - y_star
w_rf_opt
```

```
[1] -0.6160835
```

```
# Sanity check: total weight in all assets (risky + rf)
sum(w_risky_opt) + w_rf_opt
```

```
[1] 1
```