pset2_R

Franklin She

2024-02-15

Part II: 2 R Questions

Question 1. Load Data

```
library(haven)

financeR_data <- read_dta("financeR.dta")
head(financeR_data)
```

```
## # A tibble: 6 × 5
##   date          r_A       r_B     r_C     r_M
##   <chr>       <dbl>     <dbl>   <dbl>   <dbl>
## 1 12/1/2004  0.0191 -0.00585   0.100   0.0386
## 2 1/1/2005  -0.0508  0.0975    0.0227  0.0325
## 3 2/1/2005  -0.0207  0.00858  -0.0170 -0.0253
## 4 3/1/2005   0.0287  0.000354  0.0625  0.0189
## 5 4/1/2005  -0.0162  0.0179   -0.0195 -0.0191
## 6 5/1/2005   0.0124 -0.125     0.0127 -0.0201
```

Question 2: Number of Variables and Observations

```
data_dimensions <- dim(financeR_data)
num_observations <- data_dimensions[1]
num_variables <- data_dimensions[2]

cat("Number of observations (rows):", num_observations, "\n")
```

```
## Number of observations (rows): 156
```

```
cat("Number of variables (columns):", num_variables, "\n")
```

```
## Number of variables (columns): 5
```

Question 3: Data Types of Variables

```
variable_classes <- sapply(financeR_data, class)

print(variable_classes)
```

```
##        date         r_A         r_B         r_C         r_M
## "character"   "numeric"   "numeric"   "numeric"   "numeric"
```

```
non_numeric_variables <- names(variable_classes[variable_classes != "numeric"])

if(length(non_numeric_variables) > 0) {
  cat("Non-numeric variable(s):", paste(non_numeric_variables, collapse = ", "), "\n")
}
```

```
## Non-numeric variable(s): date
```

Question 4: Drop Non-numeric Variable

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
financeR_data_cleaned <- select(financeR_data, -one_of(non_numeric_variables))
head(financeR_data_cleaned)
```

```
## # A tibble: 6 × 4
##       r_A       r_B     r_C     r_M
##     <dbl>     <dbl>   <dbl>   <dbl>
## 1  0.0191 -0.00585   0.100   0.0386
## 2 -0.0508  0.0975    0.0227  0.0325
## 3 -0.0207  0.00858  -0.0170 -0.0253
## 4  0.0287  0.000354  0.0625  0.0189
## 5 -0.0162  0.0179   -0.0195 -0.0191
## 6  0.0124 -0.125     0.0127 -0.0201
```

Question 5: Calculate Excess Returns

```
rf <- 0.0041

calculate_excess_returns <- function(returns) {
  excess_returns <- returns - rf
  return(excess_returns)
}

financeR_excess_returns <- as.data.frame(sapply(financeR_data_cleaned, calculate_excess_
returns))

head(financeR_excess_returns)
```

```
##            r_A          r_B          r_C         r_M
## 1  0.015031184 -0.009947938  0.095899987  0.03445962
## 2 -0.054895037  0.093351001  0.018627288  0.02839332
## 3 -0.024807292  0.004476014 -0.021137002 -0.02939047
## 4  0.024647899 -0.003745696  0.058447088  0.01480335
## 5 -0.020266300  0.013785573 -0.023603586 -0.02321766
## 6  0.008341376 -0.129013022  0.008558242 -0.02420858
```

Question 6: Descriptive Statistics and Highest Mean Return

```
descriptive_stats <- summary(financeR_excess_returns)

print(descriptive_stats)
```

```
##       r_A                 r_B                 r_C
##  Min.   :-0.174342   Min.   :-0.462632   Min.   :-0.713305
##  1st Qu.:-0.029586   1st Qu.:-0.043620   1st Qu.:-0.082507
##  Median : 0.001516   Median : 0.004074   Median :-0.005886
##  Mean   : 0.003752   Mean   : 0.001441   Mean   : 0.007011
##  3rd Qu.: 0.042496   3rd Qu.: 0.057011   3rd Qu.: 0.058521
##  Max.   : 0.129233   Max.   : 0.287812   Max.   : 1.799471
##       r_M
##  Min.   :-0.172031
##  1st Qu.:-0.020429
##  Median : 0.006967
##  Mean   : 0.001999
##  3rd Qu.: 0.025580
##  Max.   : 0.101795
```

```
mean_returns <- sapply(financeR_excess_returns, mean)

cat("The asset with the highest mean return is:", names(which.max(mean_returns)), "with
a mean return of", max(mean_returns), "\n")
```

```
## The asset with the highest mean return is: r_C with a mean return of 0.007011021
```

Question 7: Variances and Covariances

```
var_cov_matrix <- var(financeR_excess_returns)
print(var_cov_matrix)
```

```
##               r_A           r_B          r_C          r_M
## r_A  2.905690e-03 -0.0003844852 0.0001249001 1.171093e-05
## r_B -3.844852e-04  0.0105434580 0.0132516055 2.646404e-03
## r_C  1.249001e-04  0.0132516055 0.0563354184 4.679460e-03
## r_M  1.171093e-05  0.0026464043 0.0046794598 1.615544e-03
```

```
variances <- diag(var_cov_matrix)
most_volatile_asset <- names(which.max(variances))
cat("(a) The most volatile asset is:", most_volatile_asset, "with a variance of", max(va
riances), "\n")
```

```
## (a) The most volatile asset is: r_C with a variance of 0.05633542
```

```
diag(var_cov_matrix) <- 0
highest_cov_value <- max(var_cov_matrix)
highest_cov_assets <- which(var_cov_matrix == highest_cov_value, arr.ind = TRUE)
asset_names <- rownames(var_cov_matrix)
pair_with_highest_covariance <- c(asset_names[highest_cov_assets[1, "row"]], asset_names
[highest_cov_assets[1, "col"]])
cat("(b) The two assets with the highest covariance are:", pair_with_highest_covariance
[1], "and", pair_with_highest_covariance[2], "with a covariance of", highest_cov_value,
"\n")
```

```
## (b) The two assets with the highest covariance are: r_C and r_B with a covariance of
0.01325161
```

(c)

Genworth Financial (GNW) is the most volatile asset likely due to its direct exposure to the highly fluctuating financial sector. The high covariance between Morgan Stanley (MS) and Genworth Financial (GNW) suggests a strong positive relationship, likely because both are deeply integrated within the financial industry, making their performances susceptible to similar economic and sector-specific influences.

Question 8: Risk-Return Tradeoff Analysis

```
asset_means <- sapply(financeR_excess_returns, mean)
asset_variances <- sapply(financeR_excess_returns, var)
print(asset_means)
```

```
##         r_A         r_B         r_C         r_M
## 0.003752189 0.001440923 0.007011021 0.001999263
```

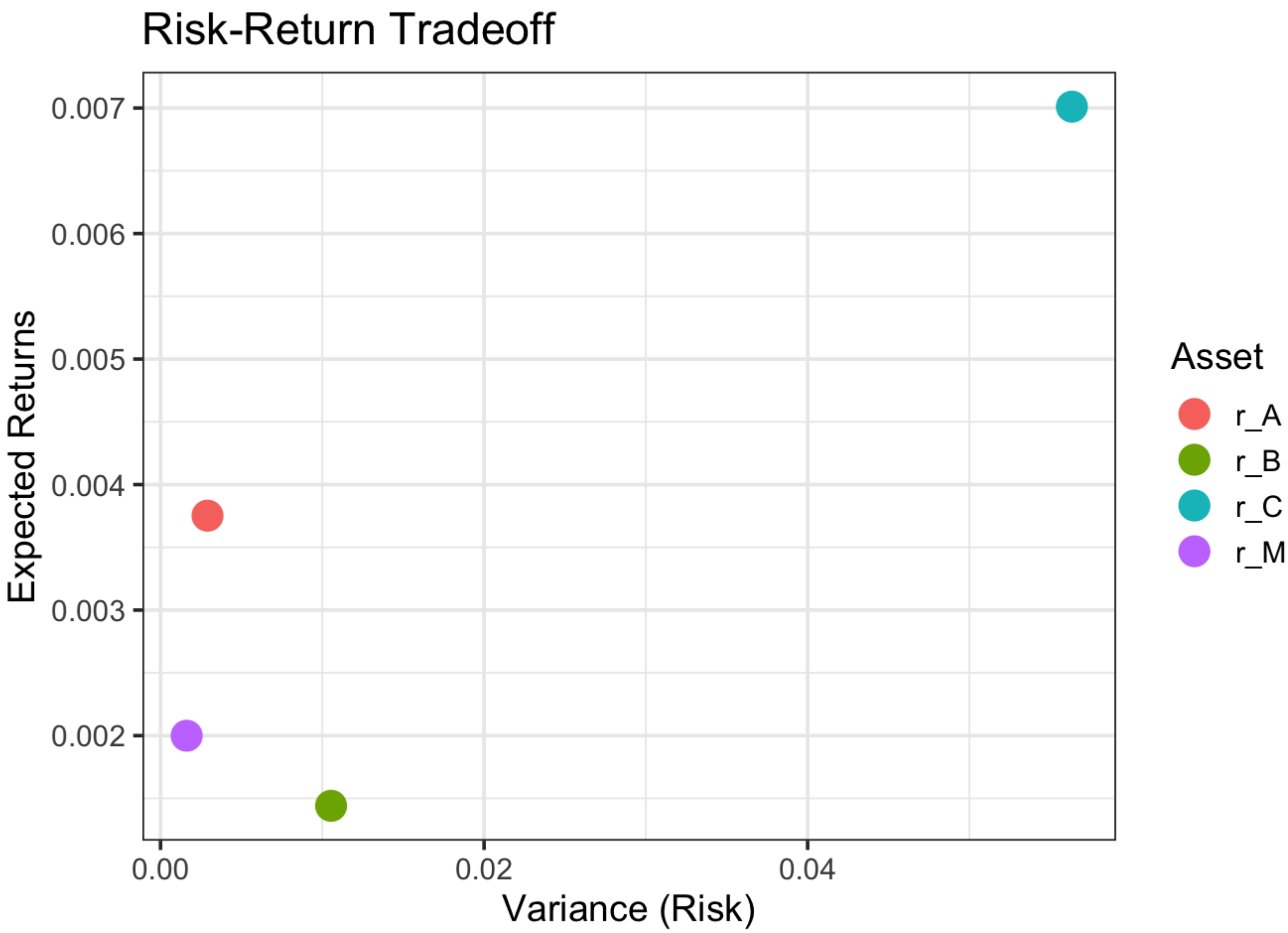
```
print(asset_variances)
```

```
##         r_A         r_B         r_C         r_M
## 0.002905690 0.010543458 0.056335418 0.001615544
```

```
library(ggplot2)

risk_return_df <- data.frame(Asset = names(asset_means), Er = asset_means, Var = asset_v
ariances)

ggplot(risk_return_df, aes(x = Var, y = Er, color = Asset)) +
  geom_point(size = 5) + theme_bw(base_size = 14) +
  ggtitle("Risk-Return Tradeoff") +
  xlab("Variance (Risk)") + ylab("Expected Returns")
```



(d)

The risk-return tradeoff displayed in the figure indicates that Genworth Financial (r_C) offers the highest expected return but also comes with the highest level of risk (variance), suggesting a potential reward for investors willing to accept greater volatility. Conversely, the S&P 500 index (r_M) demonstrates the lowest risk and offers lower expected returns, which is typical for a diversified market index. The tradeoff between risk and expected return is a fundamental concept in finance, where higher risk is typically associated with the potential for higher returns, as investors demand compensation for bearing additional risk.

Question 9: Sharpe Ratio Calculation

```
calculate_sharpe_ratio <- function(return, variance, rf) {
  sharpe_ratio <- (return - rf) / sqrt(variance)
  return(sharpe_ratio)
}

sharpe_ratios <- sapply(1:ncol(financeR_excess_returns), function(i) {
  calculate_sharpe_ratio(asset_means[i], asset_variances[i], rf)
})

names(sharpe_ratios) <- names(asset_means)
print(sharpe_ratios)
```

```
##          r_A          r_B          r_C          r_M
## -0.006452367 -0.025896396  0.012264635 -0.052265164
```

```
cat("The asset with the best estimated risk-reward tradeoff according to the Sharpe rati
o is:", names(which.max(sharpe_ratios)), "\n")
```

```
## The asset with the best estimated risk-reward tradeoff according to the Sharpe ratio
is: r_C
```

Question 10: Portfolio Diversification

(a)

Diversification can reduce the portfolio's overall risk if the returns on GLD and the S&P 500 are not perfectly correlated. The rule for the variance of a sum (i.e., the portfolio variance) indicates that if two assets do not move exactly in tandem (less than perfect correlation), combining them can lead to a portfolio variance that is lower than the weighted average of the individual variances, thereby reducing risk.

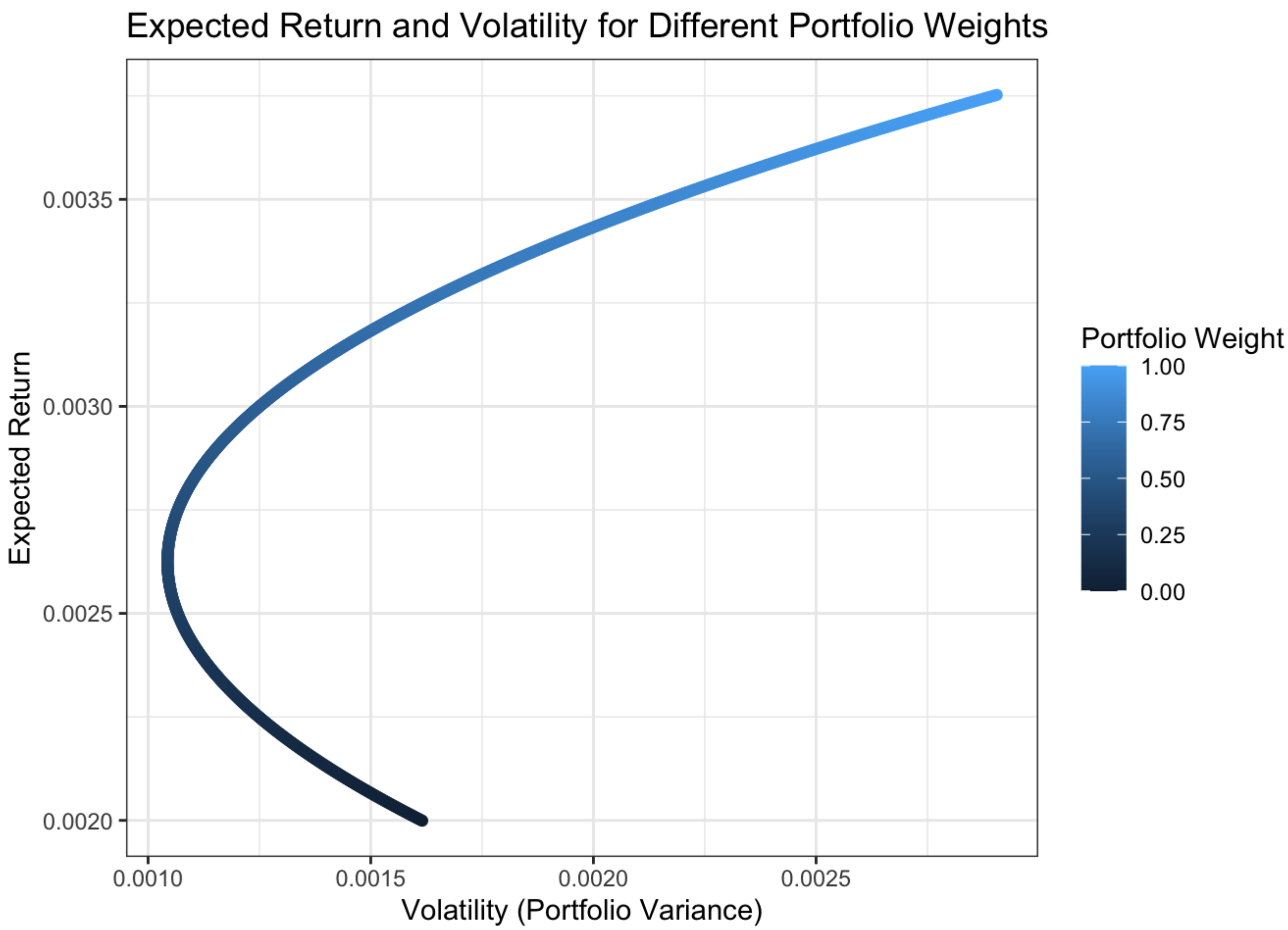
```
omega <- seq(from = 0, to = 1, length.out = 1000)

mean_GLD <- asset_means['r_A']
mean_SP500 <- asset_means['r_M']
var_GLD <- asset_variances['r_A']
var_SP500 <- asset_variances['r_M']
cov_GLD_SP500 <- var_cov_matrix['r_A', 'r_M']

portfolio_df <- data.frame(omega = omega)

# Calculate expected return and variance for each weight combination
portfolio_df$expected_return <- (omega * mean_GLD) + ((1 - omega) * mean_SP500)
portfolio_df$portfolio_variance <- (omega^2 * var_GLD) + ((1 - omega)^2 * var_SP500) +
(2 * omega * (1 - omega) * cov_GLD_SP500)

ggplot(portfolio_df, aes(x = portfolio_variance, y = expected_return, color = omega)) +
  geom_point() +
  theme_bw() +
  labs(title = "Expected Return and Volatility for Different Portfolio Weights",
       x = "Volatility (Portfolio Variance)",
       y = "Expected Return",
       color = "Portfolio Weight")
```



(c)

The risk-return tradeoff for the portfolio versus holding only SPDR Gold or only the S&P 500 index can be evaluated by comparing their Sharpe ratios, expected returns, and volatilities. If the combined portfolio offers a higher Sharpe ratio than either of the individual assets, it suggests a better risk-adjusted return, making it a more attractive choice for investors.

Given the estimates, an investor would never invest 100% in the S&P 500 index as putting at that position, putting more portfolio weight on GLD gives a higher expected return while lowering volatility.