

PANDUAN PRAKTIS MEMBUAT APLIKASI PORTOFOLIO SAHAM DENGAN R SHINY

2024



Retno Subekti
Kuswari Hernawati
Rizky Nuraini
Dicky Ramadhan
Muhammad Nashiruddin Al Bani

KATA PENGANTAR

Puji dan syukur kami panjatkan kepada Allah Yang Maha Kuasa, atas terselesaikannya buku panduan ini yang berjudul Panduan Praktis Membuat Aplikasi Web Portofolio Saham dengan R Shiny. Buku panduan ini disusun sebagai referensi praktis dan teoritis bagi para pengembang, peneliti, mahasiswa, dan praktisi keuangan yang ingin memahami serta mengembangkan aplikasi web interaktif untuk perhitungan rasio keuangan dan analisis portofolio saham.

Dalam dunia keuangan, kemampuan untuk menganalisis dan memvisualisasikan data secara cepat dan interaktif sangatlah penting. Program R Shiny, sebagai salah satu alat yang powerful dalam analisis data, menawarkan solusi berbasis web yang dapat mempermudah proses pengambilan keputusan finansial. Buku ini disusun dengan pendekatan step-by-step yang mengajak pembaca untuk mengenal dasar-dasar pemrograman R Shiny, membangun aplikasi interaktif, hingga menerapkan berbagai kalkulasi rasio keuangan seperti Return on Assets (ROA), Debt-to-Equity Ratio, hingga Mean-Variance Analysis dalam penyusunan portofolio saham.

Kami menyampaikan terima kasih kepada berbagai pihak yang telah membantu dalam penyusunan buku panduan ini sehingga dapat selesai dengan baik. Besar harapan kami buku panduan ini dapat menjadi salah satu referensi pembelajaran dalam analisis data ataupun pembelajaran statistika keuangan. Kami menyadari bahwa dalam penulisan buku panduan ini masih terdapat kekurangan sehingga kami terbuka jika ada masukan atau perbaikan. Semoga buku ini dapat membantu pembaca untuk mengembangkan keterampilan analitis dan teknis, serta memberikan kontribusi yang signifikan bagi kemajuan di bidang teknologi keuangan.

Yogyakarta, 26 Oktober 2024

(Tim Penulis)

DAFTAR ISI

DAFTAR ISI	iii
DAFTAR GAMBAR	iv
DAFTAR TABEL	v
BAB I PENDAHULUAN	1
BAB II PROGRAM R SHINY	3
2.1 Panduan Proses Instalasi R Shiny	4
2.2 Package yang Relevan.....	6
BAB III APLIKASI WEB MENGGUNAKAN R SHINY	9
3.1 Tahapan penyusunan web menggunakan program R shiny.	9
3.2 Coding dan Hasil.....	12
BAB IV PERHITUNGAN RASIO KEUANGAN.....	20
4.1 Rasio Profitabilitas	20
4. 2 Rasio Solvabilitas.....	20
4.3 Rasio Valuasi	20
BAB V PORTOFOLIO SAHAM.....	22
5.1 Karakteristik Portofolio.....	22
5.2 Portofolio Optimal Menggunakan Metode MV	22
5.2 Implementasi dalam R Shiny	24
DAFTAR PUSTAKA	26

DAFTAR GAMBAR

Gambar 3. 1. Tampilan R Shiny Sederhana	14
Gambar 3. 2. Tampilan Dashboard Saham Modifikasi	18
Gambar 3. 3. Tampilan Menu Data Saham	18
Gambar 3. 4. Contoh Tampilan Grafik Data Saham dari Yahoo Finannce	19
Gambar 3. 5. Tampilan Menu Rasio Keuangan	19
Gambar 3. 6. Tampilan Menu Deskripsi Statistik	19

DAFTAR TABEL

Table 1. Data Keuangan Saham TKIM	21
Table 2. Data Keuangan Saham GJTL.....	21

BAB I PENDAHULUAN

Dalam beberapa tahun terakhir, perkembangan teknologi digital telah membawa perubahan signifikan dalam pengolahan dan visualisasi data keuangan, bisnis, serta investasi. Berbagai aplikasi dan perangkat lunak berbasis web telah banyak dikembangkan untuk mempermudah analisis data keuangan dan memantau portofolio saham secara interaktif. Aplikasi seperti Yahoo Finance, Bloomberg, dan TradingView menjadi contoh utama platform yang memungkinkan investor dan analis untuk melacak data pasar secara real-time, mengolah berbagai rasio keuangan, dan memvisualisasikan data historis. Kemajuan ini tidak hanya mendukung pengambilan keputusan yang lebih cepat dan akurat, tetapi juga memungkinkan aksesibilitas yang lebih luas bagi pengguna di berbagai kalangan. Dalam konteks ini, kebutuhan akan alat yang dapat disesuaikan dan terintegrasi dengan kemampuan analisis yang lebih dalam semakin meningkat.

Bersamaan dengan kemajuan tersebut, software open-source seperti Python, R, dan Tableau juga berkembang pesat, memungkinkan pengguna untuk mengembangkan aplikasi kustom sesuai kebutuhan bisnis dan investasi. R, sebagai bahasa pemrograman yang kuat di bidang statistik dan data science, menonjol dalam menyediakan alat untuk mengolah dan memvisualisasikan data secara komprehensif. Dalam analisis keuangan, R memberikan fleksibilitas dalam menghitung metrik penting seperti rasio keuangan, peramalan harga saham, hingga manajemen portofolio yang optimal. Integrasi R dengan pustaka analisis data dan paket visualisasi yang kaya, menjadikan R sangat ideal untuk diterapkan dalam pengembangan aplikasi web berbasis data keuangan.

R juga memiliki keunggulan lain dibandingkan perangkat lunak lain dalam pengembangan aplikasi analisis. Kelebihan utama R adalah dukungan komunitas yang kuat, yang memungkinkan pengguna untuk mengakses berbagai paket dan pustaka yang terus berkembang. Program ini memudahkan pemrosesan data dalam jumlah besar dan mendalam dengan kecepatan yang efisien. Pengembang aplikasi dapat memanfaatkan paket-paket seperti ``quantmod`` untuk mendapatkan data pasar secara langsung, ``PerformanceAnalytics`` untuk analisis portofolio, serta ``ggplot2`` dan ``plotly`` untuk visualisasi data dinamis. R juga mendukung berbagai format input dan output data, memudahkan integrasi dengan sumber data eksternal seperti API, Excel, dan basis data SQL, sehingga fleksibel untuk kebutuhan analisis lintas-platform.

Di antara berbagai alat pengembangan aplikasi berbasis data, R Shiny menempati posisi unik sebagai platform yang memungkinkan penyusunan aplikasi web interaktif dengan fokus pada analisis data secara instan. Dibandingkan dengan framework lain seperti Flask atau Django di Python, R Shiny lebih mudah diterapkan oleh mereka yang memiliki fokus pada analisis data daripada pengembangan web. Shiny memberikan keunggulan bagi pengguna dalam mengembangkan dashboard visual yang kompleks dengan kode minimal, sementara integrasinya yang erat dengan R membuat proses

analisis dan pengembangan lebih efisien. Hal ini membuat R Shiny menjadi pilihan unggul bagi para analis dan peneliti keuangan yang memerlukan fleksibilitas dalam membangun aplikasi web tanpa memerlukan keterampilan pemrograman web yang mendalam.

BAB II PROGRAM R SHINY

R Shiny adalah sebuah framework berbasis R yang memungkinkan pengguna untuk membangun aplikasi web interaktif tanpa memerlukan keahlian mendalam dalam pengembangan web. Dengan Shiny, pengguna dapat membuat aplikasi yang menampilkan analisis data dan visualisasi secara real-time. Salah satu keunggulan utama Shiny adalah kemampuannya mengintegrasikan kode R, sehingga analisis dan pemodelan data dapat langsung dihubungkan ke antarmuka pengguna. Ini menjadikannya pilihan yang ideal untuk aplikasi yang membutuhkan perhitungan dinamis, seperti aplikasi keuangan, visualisasi statistik, atau pengolahan data ilmiah.

Shiny dirancang dengan dua komponen inti: *user interface (UI)* dan *server*. Komponen UI mengatur bagaimana aplikasi akan terlihat, termasuk elemen input (misalnya, dropdown, slider) dan output (seperti grafik dan tabel). Di sisi lain, komponen server menjalankan kode R yang dibutuhkan untuk memproses data dan mengatur respons dari antarmuka. Dengan struktur ini, R Shiny memudahkan pengguna untuk memisahkan tampilan aplikasi dari proses analisis, sehingga pengguna dapat fokus pada fungsi setiap komponen. Shiny juga mendukung berbagai jenis visualisasi dan grafik interaktif, yang meningkatkan pengalaman pengguna dalam memahami data. Di sisi lain, RStudio adalah Integrated Development Environment (IDE) untuk R yang dirancang khusus untuk memudahkan pemrograman dan analisis data. RStudio menyediakan berbagai alat dan antarmuka yang intuitif untuk menulis, mengelola, dan mengeksekusi kode R. Dalam konteks pengembangan Shiny, RStudio berperan sebagai lingkungan utama yang memungkinkan pengguna untuk mengedit, menjalankan, dan menguji aplikasi Shiny dengan mudah. Fitur-fitur seperti syntax highlighting, auto-completion, dan debugging tools sangat membantu dalam menulis kode yang lebih efisien dan minim kesalahan.

RStudio juga memiliki integrasi bawaan untuk Shiny, sehingga pengguna dapat memulai proyek Shiny dengan beberapa klik saja, melihat preview aplikasi langsung, dan melakukan deploy ke web secara langsung menggunakan layanan Shinyapps.io atau server lainnya. Dengan kombinasi R Shiny dan RStudio, pengguna memiliki ekosistem lengkap untuk pengembangan aplikasi web berbasis data, mulai dari pemrograman dan analisis data hingga penyebaran aplikasi ke pengguna akhir. RStudio dan Shiny bersama-sama membuka peluang besar bagi para analis data, peneliti, dan profesional di berbagai bidang untuk membangun aplikasi web interaktif yang mampu menangani analisis data secara real-time.

Dalam pengembangan aplikasi web interaktif, R Shiny dan Python memiliki pendekatan dan keunggulan yang berbeda namun saling melengkapi. R Shiny, sebagai framework yang mengintegrasikan kemampuan pemrograman statistik dengan antarmuka web, memungkinkan para analis data dan peneliti untuk membuat aplikasi dengan alur kerja yang efisien. R Shiny menawarkan kemudahan dalam memproses dan memvisualisasikan data real-time serta dalam mengimplementasikan model statistik atau machine learning dengan cara yang intuitif, terutama karena Shiny langsung mengakses seluruh ekosistem package R tanpa memerlukan konfigurasi tambahan. Pengguna yang fokus pada analisis data dapat memanfaatkan R Shiny untuk menghasilkan dashboard dan

laporan interaktif, sementara Shiny juga mendukung integrasi dengan HTML, CSS, dan JavaScript, yang dapat meningkatkan fungsionalitas dan tampilan aplikasi.

Sementara itu, Python sebagai bahasa pemrograman populer di bidang data science dan web development menawarkan framework seperti Flask dan Django untuk pengembangan aplikasi web yang lebih fleksibel. Framework Python ini memungkinkan integrasi data dengan backend yang kuat, mendukung pemrosesan data dalam skala besar, serta mendukung berbagai library machine learning seperti TensorFlow dan scikit-learn. Meskipun Flask dan Django membutuhkan keterampilan pemrograman web yang lebih tinggi dibandingkan R Shiny, Python menyediakan keunggulan dari segi skalabilitas dan integrasi dengan teknologi berbasis cloud. Kombinasi antara R Shiny dan Python juga banyak digunakan dalam pengembangan aplikasi hibrid, di mana Shiny berfungsi sebagai aplikasi analisis interaktif sementara Python dapat menangani proses backend atau machine learning yang lebih kompleks, menjadikan kedua platform ini solusi fleksibel untuk aplikasi berbasis data yang interaktif dan canggih.

2.1 Panduan Proses Instalasi R Shiny

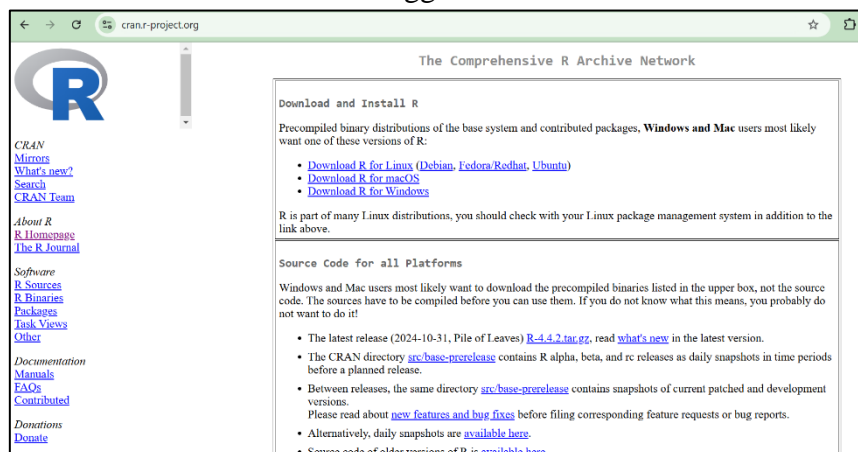
Berikut adalah struktur panduan untuk mengeksplorasi R Shiny dan fitur-fitur utamanya dalam menyusun aplikasi web keuangan:

Bagian 1: Cara Instal RStudio

Langkah pertama dalam menggunakan R Shiny adalah memastikan bahwa R dan RStudio telah terinstal dengan benar di perangkat Anda.

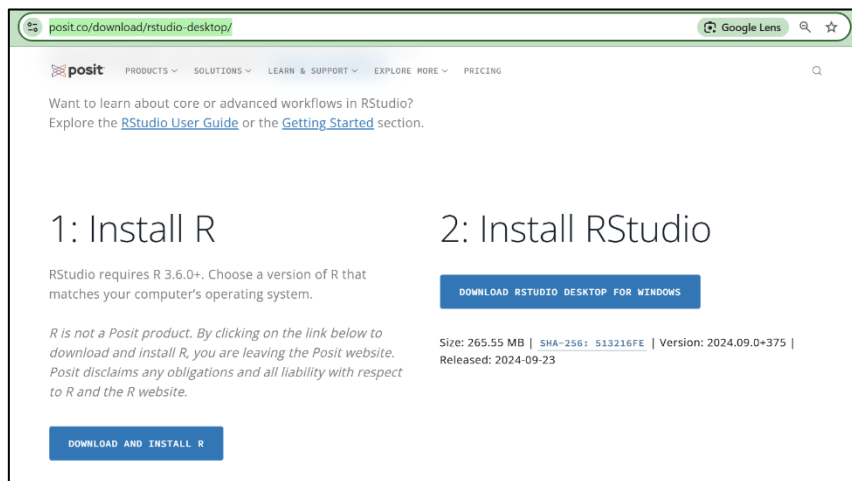
1. Instal R

- Kunjungi situs web **R Project** (<https://cran.r-project.org/>) dan unduh R sesuai dengan sistem operasi yang digunakan.
- Ikuti instruksi instalasi hingga selesai.



2. Instal RStudio

- Unduh dan instal RStudio dari situs resminya di **RStudio Download** (<https://posit.co/download/rstudio-desktop/>).
- RStudio merupakan IDE untuk R yang memudahkan pengguna dalam pengembangan aplikasi Shiny.



All Installers and Tarballs			
RStudio requires a 64-bit operating system.			
Linux users may need to import Posit's public code-signing key prior to installation, depending on the operating system's security policy.			
OS	Download	Size	SHA-256
Windows 10/11	RSTUDIO~2024.09.0~375.EXE ↕	265.55 MB	513216FE
macOS 12+	RSTUDIO~2024.09.0~375.DMG ↕	621.00 MB	54D722FD
Ubuntu 20/Debian 11	RSTUDIO~2024.09.0~375-AMD64.DEB ↕	203.93 MB	DB096050
Ubuntu 22/Debian 12	RSTUDIO~2024.09.0~375-AMD64.DEB ↕	203.92 MB	111C640B
Ubuntu 24	RSTUDIO~2024.09.0~375-AMD64.DEB ↕	203.92 MB	111C640B

3. Menginstal Paket Shiny

- Buka RStudio dan ketik perintah berikut di *Console* untuk menginstal paket Shiny:

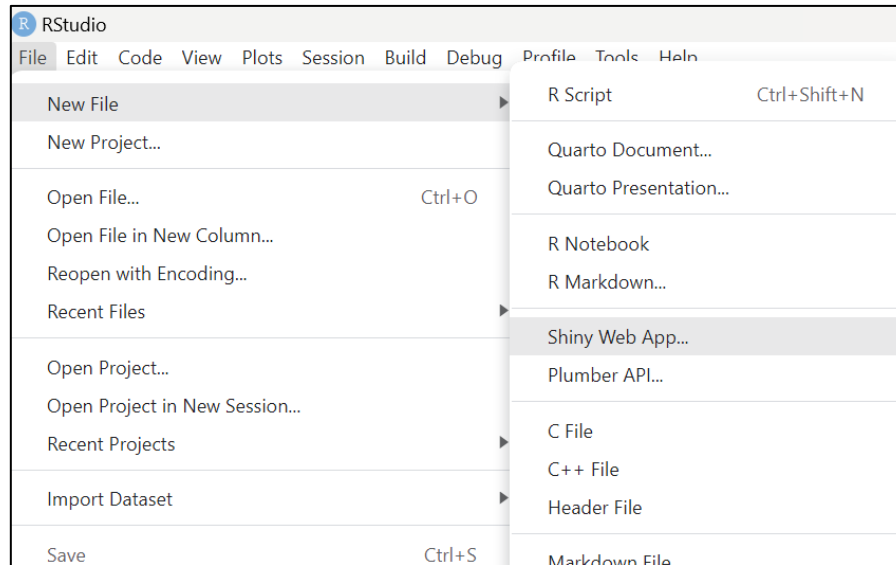
```
install.packages("shiny")
```

Bagian 2: Membuka Jendela R Shiny

Untuk memulai proyek Shiny, kita dapat menggunakan *template* bawaan (*default*) di RStudio:

1. Buka RStudio

- Pilih *File > New Project > New Directory > Shiny Web Application*.



2. Pilih Lokasi Proyek dan Nama File

- Beri nama proyek dan pilih lokasi penyimpanan.

3. Membuka File Utama Shiny

- RStudio akan membuat dua file utama, yaitu `ui.R` dan `server.R`, atau satu file `app.R` jika menggunakan struktur tunggal. Kedua file ini merupakan inti dari aplikasi Shiny yang akan kita bangun.

2.2 Package yang Relevan

Bagian 1: Menu dan Penjelasan Package untuk Pengambilan Data Saham dari Yahoo Finance

Untuk mengambil data saham dari Yahoo Finance, kita akan menggunakan beberapa package utama, yaitu `quantmod` dan `tidyquant`.

1. Menginstal dan Memuat Package

```
install.packages("quantmod")
install.packages("tidyquant")
library(quantmod)
library(tidyquant)
```

2. Mengambil Data Saham dari Yahoo Finance

- Package `quantmod` menyediakan fungsi `getSymbols()` yang memungkinkan pengambilan data saham secara otomatis.
- Contoh kode untuk mengambil data saham:

```
getSymbols("AAPL", src = "yahoo", from = "2020-01-01", to = "2023-01-01")
```

- tidyquant adalah package tambahan yang membantu dalam memproses data menjadi format yang lebih mudah dikelola.

Bagian 2: Package untuk Menampilkan Data Rasio Keuangan

Untuk menghitung rasio keuangan, kita bisa menggunakan beberapa package, di antaranya PerformanceAnalytics dan dplyr.

1. Menginstal dan Memuat Package

```
install.packages("PerformanceAnalytics")
install.packages("dplyr")
library(PerformanceAnalytics)
library(dplyr)
```

2. Menghitung Rasio Keuangan

- PerformanceAnalytics menyediakan fungsi seperti Return.calculate() untuk menghitung return, SharpeRatio() untuk rasio Sharpe, dan metrik risiko lainnya.
- dplyr membantu dalam memproses dan menganalisis data finansial, misalnya menghitung rasio leverage atau rasio profitabilitas.

3. Contoh Kode untuk Rasio Keuangan

- Contoh perhitungan rasio Sharpe:

```
returns <- Return.calculate(data_saham)
sharpe_ratio <- SharpeRatio(returns)
```

Bagian 3: Package untuk Menyusun Portofolio dengan Metode Mean-Variance

Dalam menyusun portofolio optimal dengan metode Mean-Variance, kita bisa menggunakan package PortfolioAnalytics dan ROI.

1. Menginstal dan Memuat Package

```
install.packages("PortfolioAnalytics")
install.packages("ROI")
library(PortfolioAnalytics)
library(ROI)
```

2. Menyusun Portofolio Mean-Variance

- PortfolioAnalytics menyediakan fungsi untuk menentukan alokasi aset optimal berdasarkan mean-variance.
- Langkah pertama adalah mendefinisikan portofolio dan menetapkan batas-batas alokasi setiap aset:

```
port <- portfolio.spec(assets = c("AAPL", "GOOG", "MSFT"))
port <- add.constraint(port, type = "weight_sum", min_sum = 1, max_sum = 1)
```

```
port <- add.objective(port, type = "return", name = "mean")  
port <- add.objective(port, type = "risk", name = "var")
```

3. Optimisasi Portofolio

- Setelah mendefinisikan portofolio, optimisasi dilakukan menggunakan `optimize.portfolio()` untuk mendapatkan alokasi optimal:

```
optimal_port <- optimize.portfolio(data_saham, portfolio = port, optimize_method = "ROI")
```

4. Menampilkan Hasil Portofolio Optimal

- Kita dapat menggunakan `chart.Weights()` dari `PerformanceAnalytics` untuk memvisualisasikan hasil alokasi portofolio.

```
chart.Weights(optimal_port)
```

BAB III APLIKASI WEB MENGGUNAKAN R SHINY

3.1 Tahapan penyusunan web menggunakan program R shiny.

Tahapan dalam menampilkan aplikasi r shiny yang telah dibuat menjadi sebuah web interaktif yang dapat diakses oleh pengguna lain memerlukan proses lanjutan. Berikut ini Langkah-langkah yang dapat diikuti setelah coding file aplikasi kita telah berhasil dijalankan.

Langkah-Langkah untuk menggunakan Shinyapps.io untuk Windows

1. **Buat Akun di Shinyapps.io:**

- Kunjungi [Shinyapps.io](https://shinyapps.io) dan buat akun gratis.
- Setelah mendaftar, Anda akan mendapatkan akses ke dashboard untuk mengelola aplikasi Anda.

2. **Instal Package rsconnect:**

- Buka RStudio dan pastikan Anda telah menginstal package rsconnect dengan perintah berikut:

```
install.packages("rsconnect")
```

3. **Atur Token:**

- Masuk ke akun Shinyapps.io Anda dan buka **Account Settings** untuk mendapatkan name, token, dan secret.
- Di RStudio, masukkan informasi tersebut:

```
library(rsconnect)  
rsconnect::setAccountInfo(name='your_username', token='your_token',  
secret='your_secret')
```

- Gantilah 'your_username', 'your_token', dan 'your_secret' dengan informasi yang Anda dapatkan.

4. **Deploy Aplikasi:**

- Pastikan file aplikasi Anda, misalnya app.R, sudah berjalan dengan baik di RStudio.
- Gunakan perintah berikut untuk mengunggah aplikasi Anda:

```
rsconnect::deployApp("path/to/app.R")
```

- Setelah selesai, Anda akan menerima URL yang dapat diakses oleh publik.

5. **Uji Aplikasi Anda:**

- Buka URL yang diberikan dan uji aplikasi Anda di browser untuk memastikan semuanya berfungsi dengan baik.

Langkah-Langkah untuk menggunakan Shinyapps.io untuk Linux

Untuk menjadikan aplikasi R Shiny menjadi web interaktif bagi pengguna Linux yang dapat diakses oleh pengguna lain, kita memerlukan beberapa tahapan publikasi. Berikut langkah-langkah yang bisa diikuti, misalkan file aplikasi kita diberi nama app.R:

1. Persiapan File Aplikasi

Pastikan bahwa semua kode dan package yang digunakan dalam aplikasi sudah lengkap dan berjalan dengan baik pada file app.R.

Uji aplikasi di RStudio dengan menjalankan perintah berikut untuk memastikan aplikasi dapat berjalan tanpa masalah: `R shiny::runApp("app.R")`

2. Memilih Metode Hosting

Shinyapps.io: Layanan hosting resmi dari RStudio untuk aplikasi Shiny. Mudah diakses dan tidak memerlukan server khusus.

Shiny Server: Aplikasi Shiny dapat di-host di server Linux menggunakan Shiny Server, yang tersedia dalam versi gratis dan berbayar (Shiny Server Pro).

Docker: Menggunakan container Docker untuk deployment, yang mempermudah pembagian aplikasi ke berbagai platform dan server.

3. Menggunakan Shinyapps.io untuk Publikasi

Buat akun di [Shinyapps.io](https://shinyapps.io).

Setelah akun dibuat, instal package `rsconnect` di R dan hubungkan RStudio ke akun

Shinyapps.io: `R install.packages("rsconnect") library(rsconnect)`
`rsconnect::setAccountInfo(name='your_username', token='your_token',`
`secret='your_secret')`

Untuk memublikasikan aplikasi, jalankan perintah berikut di Console: `R rsconnect::deployApp("path/to/your/app.R")`

Shinyapps.io akan mengunggah dan mengatur aplikasi sehingga dapat diakses publik melalui URL khusus. Benar sekali, langkah untuk mendapatkan dan mengatur token di Shinyapps.io perlu dijelaskan lebih rinci agar integrasi berjalan dengan lancar. Berikut penjelasan lengkap untuk langkah-langkah mendapatkan dan mengatur token untuk publikasi ke Shinyapps.io:

3a. Mendapatkan Token di Shinyapps.io

Setelah membuat akun di Shinyapps.io, buka Dashboard Shinyapps.io Anda.

Di dashboard, pilih Account Settings dan cari bagian Tokens.

Klik untuk membuat token baru. Shinyapps.io akan menghasilkan name, token, dan secret yang diperlukan untuk autentikasi.

3b. Mengonfigurasi Token di R

Setelah mendapatkan name, token, dan secret, masukkan informasi tersebut ke dalam R menggunakan perintah R

```
rsconnect::setAccountInfo(name='your_username', token='your_token',  
secret='your_secret')
```

Gantilah 'your_username', 'your_token', dan 'your_secret' dengan informasi yang telah diperoleh dari Shinyapps.io.

Dengan menambahkan token ini, RStudio dapat melakukan autentikasi untuk mengakses akun Shinyapps.io Anda, memungkinkan proses publikasi berjalan lancar. Setelah token diatur, Anda dapat melanjutkan ke langkah berikutnya untuk deploy aplikasi dengan `rsconnect::deployApp("path/to/your/app.R")`.

4. Menggunakan Shiny Server di Server Linux

Jika Anda memiliki server Linux, instal Shiny Server dengan mengunduhnya dari [Shiny Server](#). - Setelah Shiny Server terinstal, letakkan file app.R di direktori aplikasi (misalnya di /srv/shiny-server/).

- Konfigurasikan Shiny Server agar aplikasi bisa diakses melalui alamat IP server. - Pengguna dapat mengakses aplikasi melalui URL yang mengarah ke alamat IP server atau domain yang telah ditentukan.

5. Pengujian dan Pemeliharaan Aplikasi

Setelah aplikasi diterbitkan, lakukan pengujian untuk memastikan fungsi aplikasi berjalan sebagaimana mestinya.

Perbarui package atau script aplikasi jika ada perubahan. Jika menggunakan Shinyapps.io, aplikasi dapat diperbarui dengan menjalankan kembali `rsconnect::deployApp()` di console R.

Pantau penggunaan dan kinerja aplikasi, terutama jika aplikasi membutuhkan akses data dalam jumlah besar.

3.2 Coding dan Hasil

Coding 1: Tampilan dasar

```
library(shiny)
library(quantmod)
library(openxlsx) # Untuk membaca dan menulis Excel

# Fungsi untuk memperbarui B1 dengan kode saham dan membaca hasil perhitungan
# dari sheet
update_excel_and_read_results <- function(file_path, stock_code) {
  # Membuka file Excel
  wb <- loadWorkbook(file_path)

  # Menulis kode saham ke sel B1 pada sheet pertama
  writeData(wb, sheet = 1, x = stock_code, startCol = 2, startRow = 1)

  # Simpan perubahan file Excel
  saveWorkbook(wb, file_path, overwrite = TRUE)

  # Membaca hasil perhitungan dari sheet pertama (misal sel B2, B3, dst.)
  data <- read.xlsx(file_path, sheet = 1, cols = c(1, 5), rows = c(2:27)) # Sesuaikan
  range sel yang dibaca

  # Menyusun data ke dalam format tabel untuk Shiny
  colnames(data) <- c("Keterangan", "Nilai")

  return(data)
}

# UI (User Interface)
ui <- fluidPage(
  titlePanel("Data Saham dan Rasio Keuangan dari Excel"),
  sidebarLayout(
    sidebarPanel(
      textInput("stock", "Masukkan Kode Saham", value = ""), # Input simbol saham
      dateRangeInput("dates", "Pilih Periode",
        start = Sys.Date() - 30,
        end = Sys.Date(), # Pilih rentang tanggal
      fileInput("file1", "Pilih File Excel Rasio Keuangan",
        accept = c(".xlsx")),
      actionButton("goButton", "Ambil Data")
    ),
    mainPanel(
      plotOutput("stockPlot"), # Output grafik harga saham
      tableOutput("stockTable"), # Output tabel harga saham
      tableOutput("excelTable") # Output tabel rasio dari Excel
    )
  )
)
```

```

)

# Server (Logic Aplikasi)
server <- function(input, output) {

  # Reactive untuk memperbarui Excel dan mengambil data rasio
  excelData <- eventReactive(input$goButton, {
    req(input$file1) # Memastikan file diunggah
    file_path <- input$file1$datapath

    # Memperbarui file Excel dengan kode saham dan membaca hasil rasio
    update_excel_and_read_results(file_path, input$stock)
  })

  # Mengambil data saham dari Yahoo Finance
  stockData <- eventReactive(input$goButton, {
    req(input$stock) # Pastikan ada kode saham
    getSymbols(input$stock, src = "yahoo",
      from = input$dates[1],
      to = input$dates[2],
      auto.assign = FALSE)
  })

  # Menampilkan Grafik Saham
  output$stockPlot <- renderPlot({
    stock <- stockData()
    chartSeries(stock, theme = chartTheme("white"), type = "line")
  })

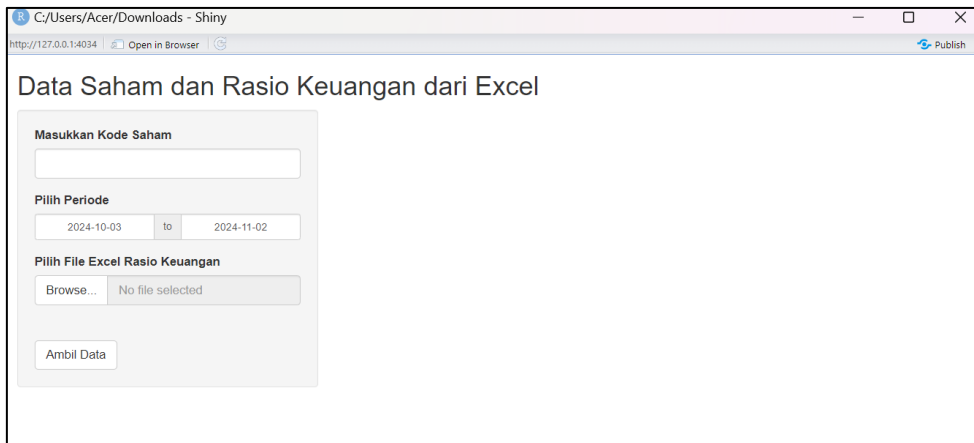
  # Menampilkan Data Saham dalam Tabel
  output$stockTable <- renderTable({
    stock <- stockData()
    tail(stock) # Menampilkan 6 data terakhir dari saham
  })

  # Menampilkan Data Rasio Keuangan dari Excel
  output$excelTable <- renderTable({
    excelData() # Memanggil fungsi untuk menampilkan hasil perhitungan
  })
}

# Jalankan Aplikasi
shinyApp(ui = ui, server = server)

```

Gambar:



Gambar 3. 1. Tampilan R Shiny Sederhana

Coding 2 : Tampilan lebih berwarna dan menu terpisah

```
library(shiny)
library(shinydashboard)
library(quantmod)
library(openxlsx) # Untuk membaca dan menulis Excel
library(moments) # Untuk menghitung skewness dan kurtosis
library(DT)      # Untuk menampilkan tabel interaktif

# UI (User Interface)
ui <- dashboardPage(
  skin = "green",
  dashboardHeader(title = "Dashboard Saham dan Analisis"),
  dashboardSidebar(
    sidebarMenu(
      menuItem("Dashboard", tabName = "dashboard", icon = icon("dashboard")),
      menuItem("Data Saham", tabName = "data_saham", icon = icon("line-chart")),
      menuItem("Rasio Keuangan", tabName = "rasio_keuangan", icon =
icon("calculator")),
      menuItem("Deskripsi Statistik", tabName = "deskripsi_statistik", icon =
icon("table"))
    )
  ),
  dashboardBody(
    tabItems(
      # Tab 1: Data Saham
      tabItem(tabName = "data_saham",
        fluidPage(
          titlePanel("Data Saham dari Yahoo Finance"),
          sidebarLayout(
            sidebarPanel(
```

```

saham      textInput("stock", "Masukkan Kode Saham", value = ""), # Input simbol
           dateRangeInput("dates", "Pilih Periode",
                           start = Sys.Date() - 30,
                           end = Sys.Date()), # Pilih rentang tanggal
           actionButton("goButton", "Ambil Data")
        ),
        mainPanel(
          plotOutput("stockPlot"), # Output grafik harga saham
          tableOutput("stockTable") # Output tabel harga saham
        )
      )
    ),
    # Tab 2: Rasio Keuangan
    tabItem(tabName = "rasio_keuangan",
            fluidPage(
              titlePanel("Rasio Keuangan dari Excel"),
              sidebarLayout(
                sidebarPanel(
                  textInput("stockRasio", "Masukkan Kode Saham", value = ""), # Input
simbol saham untuk pembaruan file Excel
                  fileInput("file1", "Pilih File Excel Rasio Keuangan",
                           accept = c(".xlsx")),
                  actionButton("goButtonRasio", "Perbarui dan Tampilkan Rasio")
                ),
                mainPanel(
                  DTOutput("dataTable"), # Output tabel data keuangan
                  tableOutput("ratiosTable") # Output tabel rasio keuangan
                )
              )
            ),
    # Tab 3: Deskripsi Statistik Return
    tabItem(tabName = "deskripsi_statistik",
            fluidPage(
              titlePanel("Deskripsi Statistik Return Saham"),
              sidebarLayout(
                sidebarPanel(
                  actionButton("calcButton", "Hitung Statistik Deskriptif")
                ),
                mainPanel(
                  tableOutput("statTable") # Output tabel statistik deskriptif
                )
              )
            )
  )

```

```

    )
  )
)
)

# Server (Logic Aplikasi)
server <- function(input, output) {

  # Mengambil data saham dari Yahoo Finance
  stockData <- eventReactive(input$goButton, {
    req(input$stock)
    getSymbols(input$stock, src = "yahoo",
      from = input$dates[1],
      to = input$dates[2],
      auto.assign = FALSE)
  })

  # Menghitung Return Saham
  stockReturns <- eventReactive(input$calcButton, {
    stock <- stockData()
    daily_return <- dailyReturn(Cl(stock)) # Menghitung return harian
    return(daily_return)
  })

  # Menampilkan Grafik Saham
  output$stockPlot <- renderPlot({
    stock <- stockData()
    chartSeries(stock, theme = chartTheme("white"), type = "line")
  })

  # Menampilkan Data Saham dalam Tabel
  output$stockTable <- renderTable({
    stock <- stockData()
    tail(stock) # Menampilkan 6 data terakhir
  })

  # Membaca data dari file Excel yang diunggah
  financialData <- eventReactive(input$goButtonRasio, {
    req(input$file1) # Memastikan file diunggah
    file_path <- input$file1$datapath

    # Membaca data dari file Excel
    data <- read.xlsx(file_path, sheet = 1)

    return(data)
  })

  # Menampilkan Data Rasio Keuangan dari Excel

```

```

output$dataTable <- renderDT({
  req(financialData())
  datatable(financialData())
})

# Menghitung rasio keuangan dan menampilkannya
output$ratiosTable <- renderTable({
  req(financialData())
  data <- financialData()

  # Menghitung rasio keuangan
  ratios <- data.frame(
    NPM = data$Laba_Bersih / data$Pendapatan,
    ROA = data$Laba_Bersih / data$Total_Aset,
    ROE = data$Laba_Bersih / data$Ekuitas,
    PBV = data$Harga_Saham / data$Nilai_Buku,
    DER = data$Total_Utang / data$Ekuitas
  )

  return(ratios)
})

# Menampilkan Statistik Deskriptif Return Saham
output$statTable <- renderTable({
  returns <- stockReturns()

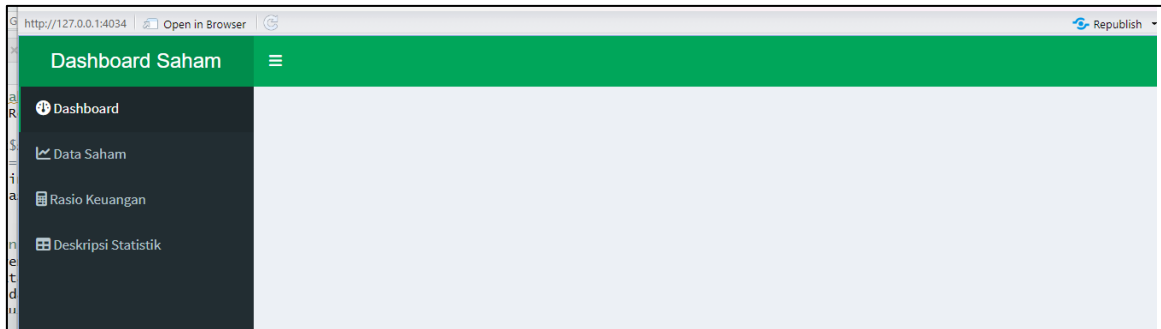
  # Menghitung statistik deskriptif
  stats <- data.frame(
    Statistika = c("Mean", "Median", "Skewness", "Kurtosis", "Minimum",
"Maximum"),
    Nilai = c(
      mean(returns),
      median(returns),
      skewness(returns),
      kurtosis(returns),
      min(returns),
      max(returns)
    )
  )

  return(stats) # Mengembalikan dataframe dalam format tabel
})
}

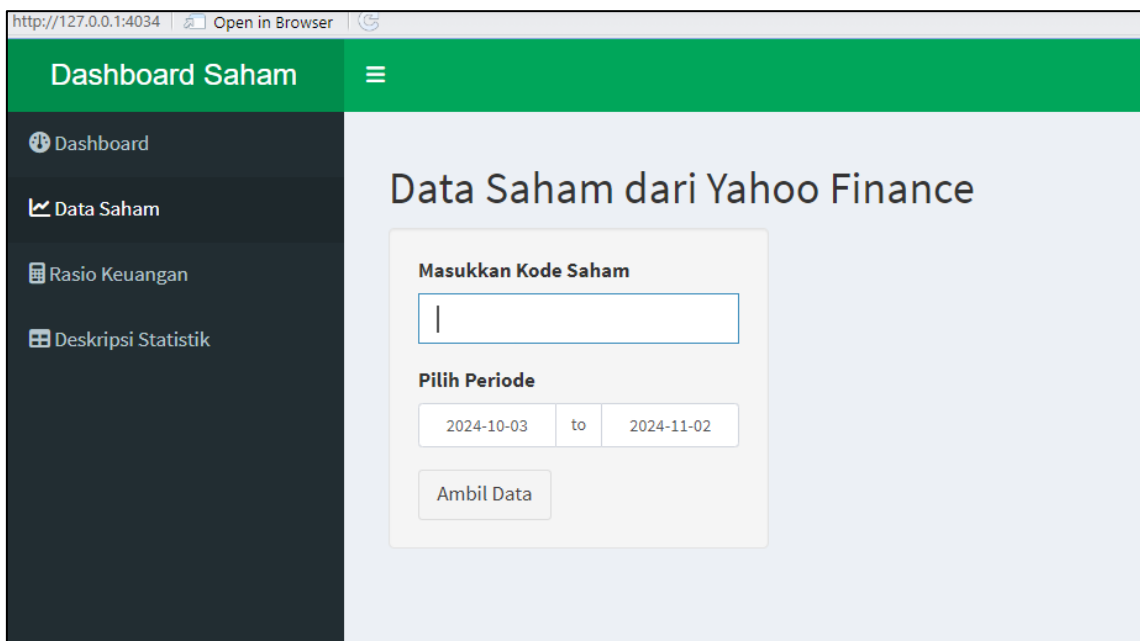
# Jalankan Aplikasi
shinyApp(ui = ui, server = server)

```

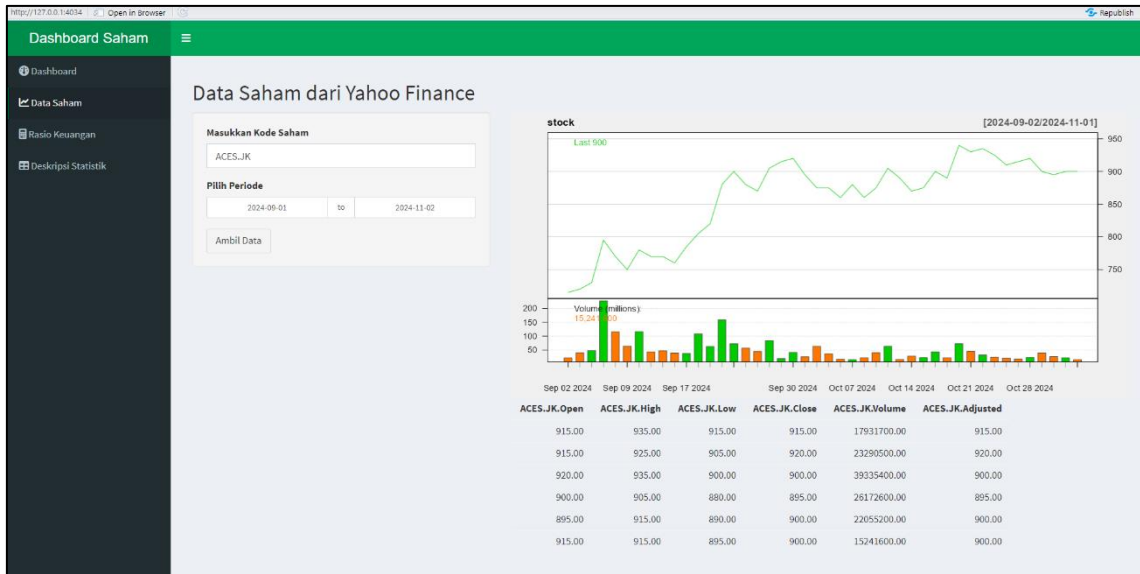
Gambar:



Gambar 3. 2. Tampilan Dashboard Saham Modifikasi



Gambar 3. 3. Tampilan Menu Data Saham



Gambar 3. 4. Contoh Tampilan Grafik Data Saham dari Yahoo Finannce

Dashboard Saham

Dashboard

Data Saham

Rasio Keuangan

Deskripsi Statistik

Rasio Keuangan dari Excel

Masukkan Kode Saham

aces

Pilih File Excel Rasio Keuangan

Browse...

datacobarasio.xlsx

Upload complete

Perbarui dan Tampilkan Rasio

Show10entries

Search:

	Laba_Bersih	Pendapatan	Total_Aset	Ekuitas	Harga_Saham	Nilai_Buku	Total_Utang
1	1000000	5000000	20000000	8000000	50	40	12000000
2	1500000	7000000	25000000	9000000	60	50	14000000
3	1200000	6000000	22000000	8500000	55	45	13000000

Showing 1 to 3 of 3 entries

Previous1Next

NPM	ROA	ROE	PBV	DER
0.20	0.05	0.12	1.25	1.50
0.21	0.06	0.17	1.20	1.56
0.20	0.05	0.14	1.22	1.53

Gambar 3. 5. Tampilan Menu Rasio Keuangan

Statistika	Nilai
Mean	0.01
Median	0.00
Skewness	1.09
Kurtosis	4.12
Minimum	-0.03
Maximum	0.09

Gambar 3. 6. Tampilan Menu Deskripsi Statistik

BAB IV PERHITUNGAN RASIO KEUANGAN

Rasio keuangan adalah alat penting dalam analisis kinerja perusahaan yang digunakan oleh investor dan analis untuk menilai kesehatan finansial suatu perusahaan. Rasio-rasio ini memberikan informasi singkat mengenai aspek keuangan seperti profitabilitas, likuiditas, dan solvabilitas perusahaan. Berikut adalah beberapa rasio keuangan yang umum digunakan:

4.1 Rasio Profitabilitas

Rasio profitabilitas menunjukkan kemampuan perusahaan untuk menghasilkan laba. Dua contoh umum adalah:

- **Return on Assets (ROA)**

ROA mengukur efisiensi perusahaan dalam menghasilkan laba dari total aset yang dimilikinya. Rumusnya adalah:

$$ROA = \frac{\text{Laba Bersih}}{\text{Total Aset}} \times 100\%$$

di mana:

- *Laba Bersih*: Laba yang diperoleh setelah pajak.
- *Total Aset*: Jumlah semua aset yang dimiliki perusahaan.

- **Return on Equity (ROE)**

ROE mengukur laba yang diperoleh dari ekuitas pemegang saham. Rumusnya adalah:

$$ROE = \frac{\text{Laba Bersih}}{\text{Ekuitas Pemegang Saham}} \times 100\%$$

4.2 Rasio Solvabilitas

Rasio ini menilai kemampuan perusahaan untuk memenuhi kewajiban jangka panjangnya. Contohnya adalah:

- **Debt to Equity Ratio (DER)**

DER membandingkan jumlah utang terhadap ekuitas perusahaan, sebagai ukuran struktur modal perusahaan:

$$DER = \frac{\text{Total Utang}}{\text{Ekuitas Pemegang Saham}}$$

4.3 Rasio Valuasi

Rasio valuasi sering digunakan untuk mengukur seberapa mahal atau murah harga saham suatu perusahaan relatif terhadap nilainya. Salah satunya adalah:

- **Price to Book Value (PBV)**

PBV mengukur nilai pasar suatu perusahaan relatif terhadap nilai bukunya, yang dihitung sebagai:

$$PBV = \frac{\text{Harga Pasar per Saham}}{\text{Nilai Buku per Saham}}$$

di mana:

- *Harga Pasar per Saham*: Harga terkini saham di pasar.
- *Nilai Buku per Saham*: Nilai ekuitas dibagi jumlah saham yang beredar.

Contoh 1.

Table 1. Data Keuangan Saham TKIM

Tahun	Laba_Bersih	Pendapatan	Total_Aset	Ekuitas	Harga Saham	Nilai_Buku	Total_Utang
2023	2,655,724,146,000	1.65789E+13	55,733,524,002,000	36,830,784,352,000	7334.87	11830.43	18,902,739,650,000
2022	7,224,475,240,000	1.78309E+13	55,276,446,560,000	34,593,159,616,000	7000.36	11111.68	20,683,286,944,000
2021	3,546,112,636,000	1.46272E+13	45,144,665,852,000	25,071,282,764,000	7489.44	8053.16	20,073,383,088,000
2020	2,062,287,602,000	1.20463E+13	42,726,199,092,000	21,027,967,731,000	9861.42	6754.4	21,698,231,361,000

Contoh 2.

Table 2. Data Keuangan Saham GJTL

Tahun	Laba_Bersih	Pendapatan	Total_Aset	Ekuitas	Harga Saham	Nilai_Buku	Total_Utang
2023	1,167,268,000,000	16,970,663,000,000	18,975,738,000,000	8,351,903,000,000	1030.5681	2396.67	10,623,835,000,000
2022	190,572,000,000	17,170,492,000,000	19,016,012,000,000	7,225,675,000,000	560.7927	2077.01	11,790,337,000,000
2021	79,896,000,000	15,344,138,000,000	18,449,075,000,000	6,967,889,000,000	660.1485	2000.45	11,481,186,000,000
2020	318,914,000,000	13,434,592,000,000	17,781,660,000,000	6,855,147,000,000	648.8592	1966.24	10,926,513,000,000

BAB V PORTOFOLIO SAHAM

Portofolio saham adalah kumpulan saham atau aset yang dimiliki oleh seorang investor dengan tujuan diversifikasi risiko dan memaksimalkan return. Dalam pembentukan portofolio, penting untuk mempertimbangkan hubungan antara return dan risiko dari aset-aset tersebut.

5.1 Karakteristik Portofolio

5.1.1 Return Portofolio

Return portofolio dihitung berdasarkan return masing-masing aset dan proporsi dana yang diinvestasikan dalam setiap aset. Jika suatu portofolio terdiri dari n saham dengan return R_1, R_2, \dots, R_n , dan proporsi investasi pada masing-masing saham tersebut adalah w_1, w_2, \dots, w_n , maka return portofolio (R_p) didefinisikan sebagai:

$$R_p = \sum_{i=1}^n w_i R_i$$

di mana: - w_i : Bobot atau proporsi investasi pada saham ke- i (dengan $\sum_{i=1}^n w_i = 1$). - R_i : Return dari saham ke- i .

5.1.2 Risiko Portofolio

Risiko portofolio diukur dengan variansi atau standar deviasi return portofolio. Jika saham-saham dalam portofolio tidak berkorelasi sempurna, diversifikasi dapat mengurangi risiko keseluruhan. Variansi portofolio (σ_p^2) dihitung sebagai:

$$\sigma_p^2 = \sum_{i=1}^n \sum_{j=1}^n w_i w_j \sigma_{ij}$$

di mana: - σ_{ij} : Kovarians antara return saham i dan j . - Jika $i = j$, maka $\sigma_{ij} = \sigma_i^2$, yang merupakan variansi return saham ke- i .

Jika kita ingin menghitung standar deviasi portofolio, maka dihitung akar kuadrat dari variansi portofolio:

$$\sigma_p = \sqrt{\sigma_p^2}$$

5.1.2 Korelasi Antar Saham dalam Portofolio

Korelasi (ρ_{ij}) antara dua saham dalam portofolio mengukur seberapa erat pergerakan return kedua saham tersebut. Korelasi berkisar dari -1 (berlawanan sempurna) hingga +1 (searah sempurna). Korelasi ini penting karena saham dengan korelasi rendah atau negatif dapat mengurangi risiko portofolio secara keseluruhan.

5.2 Portofolio Optimal Menggunakan Metode MV

Pembentukan portofolio Mean-Variance bertujuan untuk menemukan kombinasi aset yang memberikan return maksimum dengan risiko minimum. Metode ini menggunakan pendekatan matematis untuk meminimalkan variansi portofolio di bawah kendala tertentu. Dalam kasus ini, kita akan membatasi solusi tanpa *short sales*, yang berarti bobot investasi pada setiap aset tidak boleh negatif.

5.2.1 Portofolio Mean-Variance tanpa Short Sales

Jika portofolio terdiri dari n aset, dengan masing-masing aset memiliki:

- **Expected return** μ_i untuk aset ke- i ,
- **Variansi** σ_i^2 untuk aset ke- i ,
- **Kovarians** σ_{ij} antara aset i dan j .

Bobot atau proporsi investasi pada masing-masing aset dilambangkan sebagai w_i , di mana:

$$\sum_{i=1}^n w_i = 1, \quad w_i \geq 0 \quad \text{untuk semua } i.$$

Return ekspektasi portofolio, $E(R_p)$, dapat dihitung sebagai:

$$E(R_p) = \sum_{i=1}^n w_i \mu_i$$

Risiko portofolio, dinyatakan dalam variansi, adalah:

$$\sigma_p^2 = \sum_{i=1}^n \sum_{j=1}^n w_i w_j \sigma_{ij}$$

Masalah optimisasi Mean-Variance tanpa short sales dapat dirumuskan sebagai:

$$\min \sigma_p^2 = \sum_{i=1}^n \sum_{j=1}^n w_i w_j \sigma_{ij}$$

dengan kendala:

$$\sum_{i=1}^n w_i = 1 \quad \text{dan} \quad w_i \geq 0 \quad \text{untuk semua } i.$$

Solusi ini dapat ditemukan menggunakan metode optimisasi kuadratik, yang memastikan nilai w_i optimal untuk memberikan portofolio dengan risiko minimum pada tingkat return tertentu.

5.2 Implementasi dalam R Shiny

Berikut adalah langkah-langkah utama untuk mengimplementasikan optimisasi Mean-Variance tanpa short sales dalam R Shiny:

1. **Mengimpor Data Saham:** Anda bisa menggunakan data saham yang tersedia melalui *Yahoo Finance* dengan paket *quantmod* atau *tidyquant*.
2. **Menghitung Return dan Matriks Kovarians:** Untuk setiap saham, hitung return harian dan buat matriks kovarians antar return.
3. **Membuat Fungsi Optimisasi:** Gunakan fungsi `quadprog::solve.QP` untuk menyelesaikan masalah optimisasi kuadratik dengan kendala tanpa short sales.
4. **Membangun UI dan Server R Shiny:** Desain antarmuka untuk input data saham dan periode historis, serta menampilkan hasil portofolio optimal dalam bentuk grafik.

Coding implementasi R Shiny untuk portofolio Mean-Variance

```
library(shiny)
library(quantmod)
library(quadprog)
library(ggplot2)

# Fungsi optimisasi Mean-Variance tanpa short sales
optimize_portfolio <- function(returns) {
  cov_matrix <- cov(returns)
  n_assets <- ncol(returns)

  dvec <- rep(0, n_assets) # Vector untuk optimisasi kuadratik
  Amat <- cbind(rep(1, n_assets), diag(1, n_assets)) # Kendala
  bvec <- c(1, rep(0, n_assets)) # Batas kendala (tanpa short sales)

  result <- solve.QP(Dmat = cov_matrix, dvec = dvec, Amat = Amat, bvec = bvec, me
q = 1)
  weights <- result$solution # Bobot optimal
  return(weights)
}

# UI untuk aplikasi R Shiny
ui <- fluidPage(
  titlePanel("Optimisasi Portofolio Mean-Variance tanpa Short Sales"),
  sidebarLayout(
    sidebarPanel(
      textInput("stocks", "Masukkan kode saham (pisahkan dengan koma)", "AAPL,
MSFT, GOOG"),
      numericInput("period", "Periode Historis (hari)", 252, min = 50),
      actionButton("optimize", "Hitung Portofolio Optimal")
    ),
    mainPanel()
  )
)
```

```

mainPanel(
  plotOutput("portfolioPlot"),
  tableOutput("weightsTable")
)
)
)

# Server untuk aplikasi R Shiny
server <- function(input, output) {
  observeEvent(input$optimize, {
    stocks <- strsplit(input$stocks, ",")[[1]]
    getSymbols(stocks, src = "yahoo", from = Sys.Date() - input$period)

    returns <- do.call(merge, lapply(stocks, function(sym) dailyReturn(Cl(get(sym))
  )))
  weights <- optimize_portfolio(returns)

  output$weightsTable <- renderTable({
    data.frame(Saham = stocks, Bobot = weights)
  })

  output$portfolioPlot <- renderPlot({
    barplot(weights, names.arg = stocks, main = "Portofolio Optimal", col = "skyblue")
  })
})
}

# Jalankan aplikasi
shinyApp(ui = ui, server = server)

```

- **optimize_portfolio**: Fungsi ini menerima data return dari beberapa saham, menghitung matriks kovarians, dan menjalankan optimisasi kuadratik untuk menghasilkan bobot portofolio yang optimal tanpa short sales.
- **UI Shiny**: Pengguna dapat memasukkan kode saham dan periode historis yang diinginkan.
- **Plot dan Tabel**: Menampilkan bobot optimal setiap saham dalam bentuk tabel dan grafik batang.

DAFTAR PUSTAKA

- Jogiyanto. (2010). *Teori Portofolio dan Analisis Investasi (7th ed.)*. BPFE-Yogyakarta.
- Chang, W. (2018). *R Graphics Cookbook: Practical Recipes for Visualizing Data*. Sebastopol, CA: O'Reilly Media.
- Tsay, R. S. (2010). *Analysis of Financial Time Series*. New Jersey: John Wiley & Sons.
- Wickham, H., & Grolemund, G. (2017). *R for Data Science: Import, Tidy, Transform, Visualize, and Model Data*. Sebastopol, CA: O'Reilly Media.

TIM PENULIS



Dr . Retno Subekti, M.Sc sebagai salah satu penulis buku panduan ini merupakan dosen di Prodi Statistika, Departemen Pendidikan Matematika FMIPA UNY. Beliau memperoleh gelar doktor dan master pada tahun 2023 dan 2008 di Universitas Gadjah Mada. Hingga kini, fokus beberapa penelitian dan artikel yang ditulisnya adalah di bidang statistika terapan dan keuangan

Dr . Kuswari Hernawati, M.Kom sebagai salah satu penulis buku panduan ini merupakan dosen di Prodi Pendidikan Matematika, Departemen Pendidikan Matematika FMIPA UNY. Beliau memperoleh gelar doktor dan master pada tahun 2023 dan 2003 di UNY dan UGM. Penulis memiliki ketertarikan dalam pemanfaatan teknologi dalam pembelajaran.



Rizky Nur'aini, S.Stat., M.Stat sebagai salah satu penulis buku panduan ini merupakan dosen di Prodi Statistika, Departemen Pendidikan Matematika FMIPA UNY. Beliau memperoleh gelar sarjana di Universitas Brawijaya pada tahun 2019 dan master pada tahun 2023 di Institut Teknologi Sepuluh Nopember. Fokus penelitian dan artikel yang ditulisnya adalah bidang statistika terapan.

Dicky Ramadhan, S.Si sebagai salah satu penulis buku panduan ini merupakan alumni mahasiswa Prodi Matematika, Departemen Pendidikan Matematika FMIPA UNY. Beliau memperoleh gelar sarjana di Universitas Negeri Yogyakarta pada tahun 2024. Penulis memiliki ketertarikan dalam penerapan metode komputasi matematika dalam pemodelan keuangan.



Muhammad Nashiruddin Al Bani salah satu penulis buku panduan ini adalah mahasiswa di Prodi Statistika, Departemen Pendidikan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Yogyakarta (UNY). Penulis sedang menyelesaikan studi S1 nya saat ini. Selain mengikuti perkuliahan, penulis juga aktif dalam aktifitas lainnya seperti mengikuti magang, berorganisasi baik intra maupun ekstra kampus hingga publikasi buku panduan di bidang Pengabdian Masyarakat serta penulis tertarik menganalisis masalah dalam pemodelan keuangan. Beberapa hasil karya penulis telah dipublikasikan dalam HAKI ataupun dikirimkan sebagai bentuk hasil kompetisi lomba Program Kreativitas Mahasiswa.