



UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE CIÊNCIAS EXATAS E DA NATUREZA
CURSO DE ESTATÍSTICA

Lucas Bezerra de Souza

**Desenvolvimento de aplicativo web para apoio à gestão de risco de perda no
mercado de ações: uma aplicação baseada no pacote shiny do R**

Recife

2023

Lucas Bezerra de Souza

Desenvolvimento de aplicativo web para apoio à gestão de risco de perda no mercado de ações: uma aplicação baseada no pacote shiny do R

Trabalho de Conclusão de Curso apresentado à Coordenação do Curso de Estatística do Campus Recife da Universidade Federal de Pernambuco, na modalidade de monografia, como requisito parcial para obtenção do grau de bacharel em Estatística

Orientador (a): Prof. Dr. Vinícius Quintas Souto
Maior

Coorientador (a): Prof. Dr. Wilton Bernardino da
Silva

Recife

2023

Ficha de identificação da obra elaborada pelo autor,
através do programa de geração automática do SIB/UFPE

Souza, Lucas Bezerra de.

Desenvolvimento de aplicativo web para apoio à gestão de risco de perda no mercado de ações: uma aplicação baseada no pacote shiny do R / Lucas Bezerra de Souza. - Recife, 2023.

25 p. : il.

Orientador(a): Vinícius Quintas Souto Maior

Coorientador(a): Wilton Bernardino da Silva

Trabalho de Conclusão de Curso (Graduação) - Universidade Federal de Pernambuco, Centro de Ciências Exatas e da Natureza, Estatística - Bacharelado, 2023.

1. GARCH. 2. VaR. 3. shiny. 4. backtest. 5. webapp. I. Maior, Vinícius Quintas Souto. (Orientação). II. Silva, Wilton Bernardino da. (Coorientação). IV. Título.

310 CDD (22.ed.)

RESUMO

O presente trabalho apresenta uma proposta de aplicação *web* destinada a realizar análises quantitativas sobre ativos de renda variável nos mercados brasileiro e internacional. Em particular, os desenvolvimentos visam o mercado de ações. Neste sentido, há a visualização de uma série de ativos nacionais e internacionais, criação de carteiras de ativos e realização de *backtesting*, comparação de ativos, análise de cluster, estratégias de compra/venda baseadas em sistema de negociação, entre outros estudos automatizados. A aplicação utiliza o pacote *shiny* da linguagem R para criar uma interface amigável. Como principal resultado, existe um motor de análise automatizado que é muito útil, principalmente para investidores individuais.

Palavras-chaves: shiny, R, webapp, GARCH, VaR, backtest.

ABSTRACT

The present work presents a proposal for an application *web* designed to carry out quantitative analyses on variable income assets in the Brazilian and international markets. In particular, the developments are aimed at the stock market. In this sense, there is the visualization of a series of national and international assets, creation of portfolios of assets and carrying out *backtesting*, comparison of assets, cluster analysis, buying/selling strategies based on a trading system, among others automated studies. The application uses the *brilliant* package of the R language to create a user-friendly interface. As a main result, there is an automated analysis engine that is very useful, especially for individual investors.

Keywords: shiny, R, webapp, GARCH, VaR, backtest.

LISTA DE FIGURAS

Figura 1 – Opções para modelo de ativos nacionais	15
Figura 2 – Opções para modelo de ativos internacionais	15
Figura 3 – Tela de Analise de VaR	17
Figura 4 – Tela de comparação de ativos	18
Figura 5 – Opções para Analise de <i>Clustering</i>	19
Figura 6 – Tela de Analise de <i>Clustering</i>	19
Figura 7 – Opções para aba de criação de portfólios	20
Figura 8 – Opções para modelo de portfólio	20
Figura 9 – Tela de visualização da carteira	21
Figura 10 – Tela do <i>Trade System</i>	22

SUMÁRIO

1	INTRODUÇÃO	7
2	FUNDAMENTAÇÃO TEÓRICA	8
2.1	APLICAÇÕES WEB E O USO DO SHINY	8
2.2	VALOR EM RISCO	9
2.3	AGRUPAMENTO	9
2.4	TRADE SYSTEM	10
3	DADOS E MÉTODOS	11
3.1	DADOS	11
3.2	SHINY	11
3.3	GARCH-VAR E MEAN EXCESS LOSS	12
3.4	AGRUPAMENTO	13
3.5	TRADE SYSTEM	14
4	RESULTADOS	15
4.1	ABA <i>INPUT</i>	15
4.2	ABA ANÁLISE VAR	16
4.3	ABA COMPARAÇÃO DE ATIVOS	16
4.4	ABA ANÁLISE DE AGRUPAMENTO	16
4.5	ABA PORTFÓLIO	20
4.6	ABA TRADE SYSTEM	21
5	CONCLUSÃO	23
	REFERÊNCIAS	24

1 INTRODUÇÃO

No início da *internet* os *websites* eram formados por um conjunto de arquivos de hipertexto conectados que exibiam informação usando texto e gráficos limitados. Com o avanço das tecnologias, o crescimento da linguagem HTML (*HyperText Markup Language*) e ferramentas de desenvolvimento como *Java*, tornou-se possível a utilização da capacidade computacional junto as informações para criar aplicações cada vez mais complexas. Assim surgiam os sistemas e aplicações baseados na *Web*, os *webapps*. Atualmente os *webapps* evoluíram para ferramentas computacionais complexas que não apenas oferecem funções especializadas ao usuário, como também foram integradas aos bancos de dados corporativos e às aplicações de negócio (PRESSMAN, 2016).

O sistema de negociações sistemática (*trade systems*) foi introduzido em 1949 por Richard Donchian quando lançou um dos primeiros fundos de commodities denominados Futures Inc. O sistema gera sinais de compra e venda através de um conjunto de regras estabelecidas e realizam operações automaticamente. Sistemas de negociações automáticas se beneficiam de serem testáveis e reproduzíveis (com o uso de simulação computacional), diferente da negociação discricionária onde quem está realizando as operações decide quais ações tomar baseadas na condição atual do mercado. Negociações sistemáticas se beneficiam de possuir melhores previsões do mercado e gerenciamento de risco (BURGESS, 2019). Também reduzem viés de seleção que pode levar a possíveis perdas por decisões impulsivas.

Este trabalho tem como objetivo discutir alguns detalhes do desenvolvimento de um aplicativo *web* feito em R (R Core Team, 2023), com ajuda do pacote *shiny* (CHANG et al., 2023) para avaliar o desempenho do Valor em Risco (VaR) como gerador de sinal para negociações no mercado de ações nacionais e internacionais. O aplicativo auxilia na visualização de métricas como VaR, *excess mean loss* e médias móveis para a formação de carteiras de ativos. Assim como o agrupamento de ativos por métodos hierárquicos e não hierárquicos.

O presente trabalho está dividido em 4 capítulos. O capítulo 2 onde é feita uma revisão da literatura sobre aplicativos *shiny*, valor de risco, agrupamentos e *trade system*. O capítulo 3 onde será discutida a metodologia usada para a construção do aplicativo. O capítulo 4 onde são apresentadas as funcionalidades do aplicativos. Por fim o capítulo 5 onde é discutido o estado atual da aplicação e possíveis melhorias que possam ser implementadas.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 APLICAÇÕES WEB E O USO DO SHINY

O *shiny*(CHANG et al., 2023) é um *framework* que permite a criação de aplicativos *web* usando a linguagem de programação R(R Core Team, 2023). Aplicativos *web* são aplicativos que podem ser acessados através da internet e são desenvolvidos usando linguagens de programação que podem ser acessados por navegadores(AL-FEDAGHI, 2011). Portanto aplicativos *web* podem ser acessados a partir de qualquer sistema operacional que possua um navegador, sem a necessidade de instalações. Assim todos os usuários tem acesso a mesma versão do aplicativo fazendo com que não tenhamos problemas de compatibilidade.

Uma aplicação *web* pode ser vista como uma extensão de um sistema *web* (CONALLEN, 1999). Em um sistema *web* para acessar um documento, o usuário deve usar um navegador que faz requisições através da *internet* que são processadas por um servidor *web* que monitora a atividade da rede. O servidor então recebe a requisição, localiza em seus arquivos o que está sendo requisitado e então retorna o documento esperado pelo navegador.

O *shiny* torna simples o compartilhamento de análises e gráficos os usuários podem interagir (BEELEY, 2013). O *shiny* foi desenvolvido para ser uma ferramenta que nos permite criar aplicativos *web* complexos sem nenhum conhecimento de HTML, CSS(*Cascading Style Sheets*) ou JavaScript. Porém isto não limita os aplicativos a serem feitos de forma predefinida, o *shiny* possui funções para que possamos utilizar essas outras linguagens de programação para fazer aplicativos da maneira que desejarmos.

Uma aplicação *shiny* consiste em dois componentes: um objeto de interface de usuário chamada de *UI* e uma função de *server*. A *UI* contém o código que controla a aparência do aplicativo, ou seja, o que o usuário vai visualizar como os painéis, as abas, as cores, os botões, etc. A função do *server* contém as relações entre as entradas da interface de usuário e as saídas que vão ser exibidas para o usuário. Logo, na função do servidor é onde podemos carregar dados, ajustar modelos e gerar visualizações para o usuário (BENÍTEZ; COLL-SERRANO; BOLÓS, 2021).

2.2 VALOR EM RISCO

No gerenciamento de risco de mercado, um dos valores padrões de referência é o valor em risco (VaR)(DUFFIE; PAN, 1997). Para uma dada janela de tempo t e um nível de confiança τ o VaR é um valor que quantifica as perdas que podem ocorrer no período de tempo t que excedam o nível $1 - \tau$. É importante saber que o valor de risco é uma medida única e resumida do risco normal do mercado (LINSMEIER; PEARSON, 1996), sendo assim somente mais uma ferramenta para tomada de decisões a cerca dos riscos de um portfólio. Portfólio é um conjunto de bens financeiros, como ações, títulos de renda fixa, imóveis, moedas, etc., de propriedade de um investidor. Outra medida de risco útil proposta por (ARTZNER et al., 1999) é conhecida como *expected shortfall* (ES). O ES é definido como a perda esperada quando esta perda é maior que o valor de risco, portanto o ES é uma informação adicional ao VaR.

As técnicas para estimação do VaR podem ser classificadas como modelos paramétricos, não paramétricos e semi paramétricos. Para os modelos paramétricos é preciso assumir uma distribuição para os retornos, e então estimar seus parâmetros para que possamos usar os quantis da distribuição estimada. Como exemplo temos os modelos auto regressivos ARCH e GARCH (BOLLERSLEV, 1986; ENGLE, 1982). Nos modelos não paramétricos a distribuição dos retornos pode ser obtida empiricamente, sem precisar de especificação a prévia. Por fim, os modelos semi-paramétricos são modelos que possuem componentes paramétricos e não paramétricos, como exemplo temos modelos de regressão quantílica como proposto por (KOENKER; BASSETT, 1978).

O procedimento para se estimar o VaR é geralmente baseado em modelos GARCH. Primeiro se ajusta um modelo GARCH para os retornos e então precisamos escalar essa estimação usando algum quantil ou a média truncada das inovações (GAO; SONG, 2008).

2.3 AGRUPAMENTO

Na área de aprendizado de máquina temos dois tipos de algoritmos, os supervisionados e os não supervisionados. Os algoritmos supervisionados são aqueles que precisam ser alimentados com pares de entradas e saídas conhecidas (MOHRI; ROSTAMIZADEH; TALWALKAR, 2018). Essas saídas são chamadas de rótulos que podem ser valores numéricos ou categorias. Porém nem sempre estes rótulos estão disponíveis fazendo necessário a utilização de algoritmos não supervisionados. Nos modelos de aprendizagem não supervisionados o algoritmo não tem

conhecimento dos rótulos e tenta realizar inferências a partir dos próprios dados.

Entre os algoritmos de aprendizagem não supervisionado um dos mais populares é o *K-means*. O objetivo do algoritmo *K-means* é dividir os dados em k conjuntos de forma que a soma de quadrados dentro de cada conjunto seja minimizada. Logo, este algoritmo é um procedimento de busca local e seu desempenho depende completamente das condições iniciais (PENA; LOZANO; LARRANAGA, 1999).

Outra estratégia para realizar agrupamento de dados de forma não supervisionada são os métodos de agrupamento não hierárquicos. Existem dois tipos de agrupamento hierárquico os aglomerativos e os divisivos. Os métodos de agrupamentos aglomerativos consistem em construir a união de árvores binárias. Cada observação começa em uma folha e então são agrupados dois a dois de modo que a distância entre os elementos seja a menor. O processo é repetido até chegarmos à raiz da árvore que contém todas as observações (NIELSEN; NIELSEN, 2016). Os métodos divisivos realizam o procedimento contrario começando da raiz e então dividindo as observações em conjuntos disjuntos, o procedimento é repetido até que tenha apenas 1 elemento em cada conjunto.

2.4 TRADE SYSTEM

Um *trade system* é um conjunto de regras que definem condições para compra e venda sem ambiguidade. Essas condições são geralmente baseadas em indicadores técnicos ou na combinação deles. O objetivo principal de um *trade system* é fazer o gerenciamento de risco e aumentar os lucros de operações financeiras (FXSTREET, 2023). Para saber a eficiência de um *trade system* é preciso realizar simulações baseadas em dados históricos, essas simulações são conhecidas como *backtesting*. A ideia é que o *trade system* que tiver uma boa performance no passado ,ou seja, gere o maior lucro, tem mais chances de ter uma boa performance no futuro. Porém, o *backtest* é apenas uma simulação e não garante o comportamento futuro do *trade system* (MAIER-PAAPE; PLATEN, 2014).

3 DADOS E MÉTODOS

3.1 DADOS

Os dados foram obtidos a partir da plataforma Yahoo Finance (<https://finance.yahoo.com/?guccounter=1>), que disponibiliza de maneira gratuita dados históricos (diários) de preços de abertura, fechamento, máximo, mínimo e volume transacionado em ativos do mercado financeiro brasileiro e internacional.

3.2 SHINY

Uma aplicação *shiny* é composta por dois *scripts*, um para a interface gráfica, conhecido como *UI* (interface de usuário), e um para a parte lógica do aplicativo, conhecido como servidor. No *script ui.R* temos as instruções para a interface de usuário, ou seja, definimos o posicionamento dos gráficos, tabelas, botões e todas as opções de entrada que o usuário pode inserir. Cada elemento deste *script* deve possuir um nome único para que possa ser identificado pelo servidor para realizar as operações desejadas.

O *shiny* utiliza por padrão um *framework* conhecido como *bootstrap* para gerar a interface do usuário (ALLAIRE, 2021). O *bootstrap* é um *framework* que organiza o *layout* do nosso aplicativo. Assim, podemos customizar a aparência do nosso aplicativo com os diversos temas disponíveis. Podemos usar as classes específicas do *bootstrap* para alterar o visual de elementos de entrada e saída do nosso aplicativo.

Nosso aplicativo faz o uso do pacote *shinydashboard*(CHANG; Borges Ribeiro, 2021) para gerar um *layout* com barra de navegação vertical. Para visualização de gráficos é utilizada a biblioteca *plotly*(SIEVERT, 2020), que gera gráficos interativos em *Python*.

A parte lógica do aplicativo, que fica no arquivo *server.R*, é onde fazemos os cálculos a partir dos dados inseridos pelo usuário na interface gráfica. O *shiny* permite ao desenvolvedor fazer o uso de programação reativa. Geralmente um *script* comum no R é executado linha por linha de cima para baixo, com o uso da programação reativa o aplicativo *shiny* aguarda um evento específico acontecer para poder reagir a este evento e executar o código à medida que for necessário, assim reduzindo a quantidade de operações realizadas pelo aplicativo e otimizando nossos servidor.

Para executar nosso aplicativo precisamos da função *shinyApp* para conectar as funções

da interface de usuário e do servidor. Nesta função podemos definir opções de IP (protocolo de internet), porta onde o aplicativo sera acessado e o padrão de url (*Uniform Resource Locator*) para nosso aplicativo.

3.3 GARCH-VAR E MEAN EXCESS LOSS

Para estimar o VaR precisamos escolher uma distribuição condicional paramétrica para o retorno dos ativos. Então, o τ -ésimo quantil da distribuição escolhida sera o VaR considerando uma probabilidade de cauda $\tau \in (0, 1)$. Considerando uma série de retornos diferidos da média (*demeaned returns*) $\{R_t^{dm}\}$, a lei de probabilidade $\mathbf{P}(R_t^{dm} \leq \text{VaR}_\tau | \mathcal{F}_{t-1}) = F_t(\text{VaR}_\tau)$ é condicional ao conjunto de informações \mathcal{F}_{t-1} (σ -álgebra) no tempo $t - 1$.

Supondo que $\{R_t^{dm}\}$ siga o processo estocástico $R_t^{dm} = \mu_t + \epsilon_t = \mu_t + \sigma_t z_t$, onde $\mu_t = \mathbf{E}(R_t^{dm} | \mathcal{F}_{t-1}) = 0$, $\sigma_t^2 = \mathbf{E}(\epsilon_t^2 | \mathcal{F}_{t-1})$ e $\{z_t\} = \{\epsilon_t / \sigma_t\}$, em que $\{z_t\}$ possui função de distribuição condicional $G_t(z) = \mathbf{Pr}(z_t \leq z | \mathcal{F}_{t-1})$. Assim dada uma probabilidade τ definimos o VaR como o quantil condicional $F_t(\text{VaR}_{\tau t}) = \tau$, que pode ser estimado a partir da função quantílica

$$\text{VaR}_{\tau t} = F_t^{-1}(\tau) = \sigma_t G_t^{-1}(\tau). \quad (3.1)$$

Assim, o modelo VaR pode ser obtido a partir da escolha de $F_t(\cdot)$, μ_t , σ_t^2 e $G_t(\cdot)$.

Geralmente para se obter VaR no período t utilizamos o modelo GARCH para estimar a volatilidade σ_t . O modelo GARCH(p, q) é definido por

$$\sigma_t^2 = \gamma_0 + A(\omega)(R_t^{dm})^2 + B(\omega)\sigma_t^2,$$

onde $A(\omega) = \gamma_1\omega + \dots + \gamma_q\omega^q$ e $B(\omega) = \beta_1\omega + \dots + \beta_p\omega^p$, em que p e q são escolhidos a partir dos componentes autorregressivo e de médias moveis da série, respectivamente, ω é o operador de defasagem.

A família de modelos GARCH geralmente apresenta uma boa performance para a modelagem de variância condicional, (HANSEN; LUNDE, 2005) analisou diversos modelos de volatilidade diferentes e concluiu que não existem evidências de que outros modelos possuam melhor performance que o GARCH(1, 1). Em muitas situações a distribuição normal gaussiana é utilizada para estimar o VaR. Porém (ANGELIDIS; BENOS; DEGIANNAKIS, 2004), após avaliar diversas variações do modelo GARCH para a estimação do VaR, concluiu que distribuições leptocúrticas, em especial a distribuição t de student, são mais apropriadas que a distribuição normal para a modelagem do VaR.

No *backtesting* do VaR podemos escolher uma janela de estimação considerando a razão de violação. A razão de violação é definida pela proporção de violações ao VaR comparado com o número esperado de violações, quando consideramos todo o período de um passo de previsão do VaR. A razão de violação ideal é 1, ou seja, esperamos que ao considerar um $\text{VaR}_{95\%}$ e 100 previsões, tenhamos apenas 5 violações. Porém, nem sempre é possível obter uma razão de violação exatamente igual a 1, mas podemos definir um intervalo aceitável para esta medida. Logo, podemos procurar uma janela de estimação onde a razão de violação fique o mais próximo de 1.

A função *mean excess loss* (ZIEGEL, 2006) é definida como o valor esperado da perda excedente a um valor limiar u , ou seja, $e(u) = E(X - u | X > u)$. Usaremos esta função para fazer agrupamento de ativos, dando uma ideia a mais para a formação de portfólios.

3.4 AGRUPAMENTO

O algoritmo *K-means* utilizado será o proposto por (HARTIGAN; WONG, 1979). O objetivo do *K-means* é particionar um conjunto de n pontos em k grupos de modo que cada ponto pertença a um conjunto com melhor soma de quadrados. Esses grupos são formados iterativamente a partir dos próprios pontos que são divididos até atingir o número especificado de k grupos. O *K-means* é um método de agrupamento não-hierárquico.

O agrupamento dos ativos pode ser utilizado para acelerar o processo de formação de portfólios como proposto por (KUMARI et al., 2019). Faremos o agrupamento dos ativos considerando métricas como o VaR, VaR absoluto, *mean excess loss*, VaR - 5%, etc. Assim podemos ter uma visão de quais ativos apresentam um risco semelhante podendo formar portfólios com diferentes níveis de risco para maximizar o lucro.

Para realizar o agrupamento hierárquico usaremos o critério proposto por (WARD, 1963) para minimizar a variância dentro de cada conjunto. Para utilizar esse critério no primeiro passo cada conjunto contém apenas 1 ponto e então a cada passo formam-se novos conjuntos são formados com pontos de modo que cada novo ponto adicione o menor valor possível de variância ao conjunto.

Esses algoritmos de agrupamento estão disponíveis na biblioteca do R `cluster` (MAECHLER et al., 2022) e serão utilizados para visualizar possíveis conjuntos de ativos para a composição de carteiras.

3.5 TRADE SYSTEM

Um *trade system* é um conjunto de regras que define condições para entrada e saída para uma negociação. As regras de entrada e saída não podem ser ambíguas. Para este trabalho usaremos uma estratégia de compra e venda que utiliza sinais de cruzamento. Assim, quando o preço cruza a curva de algum indicador, como o VaR ou médias moveis podemos escolher realizar uma operação de compra ou venda. Usaremos o pacote `quantstrat` (PETERSON et al., 2023) disponível para o R para realizar o *backtesting*.

4 RESULTADOS

4.1 ABA INPUT

O aplicativo possui uma tela inicial onde o usuário escolhe um intervalo para a razão de violação associada às estimativas VaR e utiliza-la como critério de parada na avaliação VaR em *backtesting*. O aplicativo permite análise automatizada de ativos nacionais e internacionais seguindo essa ótica. Adicionalmente, é possível analisar ativos listados especificamente nos seguintes índices: Dow Jones, Dow Jones Global Index, S&P 400, S&P 500 e S&P 600. O web app permite o ajuste dos preços dos ativos internacionais convertido para o real brasileiro, possibilitando uma comparação entre ativos nacionais e internacionais. As Figuras 1 e 2 ilustram as telas de análise VaR disponíveis no *app*.

Figura 1 – Opções para modelo de ativos nacionais

Figura 2 – Opções para modelo de ativos internacionais

Ao fim da execução da aba *INPUT*, o aplicativo forma um banco de dados com todas as informações úteis às análises de várias outras abas do aplicativo. Os dados são armazenados em formato *.rdata* e são disponibilizados em uma pasta específica do *app*. A depender da quantidade de ativos analisados, executar a aba *INPUT* pode levar cerca de 2 a 8 horas. As análises podem ser atualizadas diariamente, considerando um horizonte de análise em frequência diária.

4.2 ABA ANÁLISE VAR

Após executar a aba *INPUT* (definindo-se as entradas mostradas em tela), os ativos selecionados ficam disponíveis para serem analisados na ótica das séries VaR e da função *mean excess loss*. Nesse sentido, o usuário escolhe o ativo e visualiza um gráfico de *candles* com ajuste de média móvel (de período escolhido pelo usuário) bem como a curva representando a série 1% VaR, estimada pelo método GARCH-VaR (BERNARDINO et al., 2018). É possível analisar ativos nacionais e internacionais. O segundo gráfico apresenta os retornos e a curva do 1% VaR. O terceiro gráfico mostrado em tela resume a linha de excedente de perda média (*mean excess loss*), estimada para o ativo em análise e no período (diário) escolhido pelo usuário. As demais informações mostradas são um box-plot dos retornos (diários) e estatísticas sumárias da análise VaR realizada para o ativo selecionado. Uma ilustração dessa aba é mostrada na Figura 3.

4.3 ABA COMPARAÇÃO DE ATIVOS

A aba de comparação de ativos permite ao usuário obter uma visão comparativa entre 2 ou mais ativos. A aba exibe box-plots comparativos, curvas de excedentes de perdas médias, linhas GARCH-VaR estimadas e valores médios dessas séries entre os ativos escolhidos pelo usuário. A Figura 4 ilustra os resultados mostrados nessa aba do aplicativo.

4.4 ABA ANÁLISE DE AGRUPAMENTO

A aba de Análise de *Clustering* utiliza o método *K-means* para agrupar os ativos em 4 grupos utilizando como medida de similaridade as métricas VaR, $e(d)$ (*mean excess loss*, distância entre VaR e retorno, distância entre VaR e 5%, dentre outras. A Figura 5 ilustra as opções de medida

de similaridade utilizada na formação dos agrupamentos entre a lista de ativos selecionada pelo usuário. O resultado da análise de *Clustering* indica os grupos de ativos listados por nível de risco, de acordo com a medida de similaridade escolhida para a análise. A Figura6 mostra as saídas do app quando utiliza-se a análise de agrupamentos. Na parte superior, mostram-se os ativos listados por grupos e níveis de risco. Para podermos visualizar os grupos utilizaremos a análise de componentes principais (ABDI; WILLIAMS, 2010). No terceiro gráfico mostra-se a distribuição dos ativos entre os setores e o último gráfico representa o dendrograma da análise de grupos obtido em uma análise de agrupamento hierárquico.

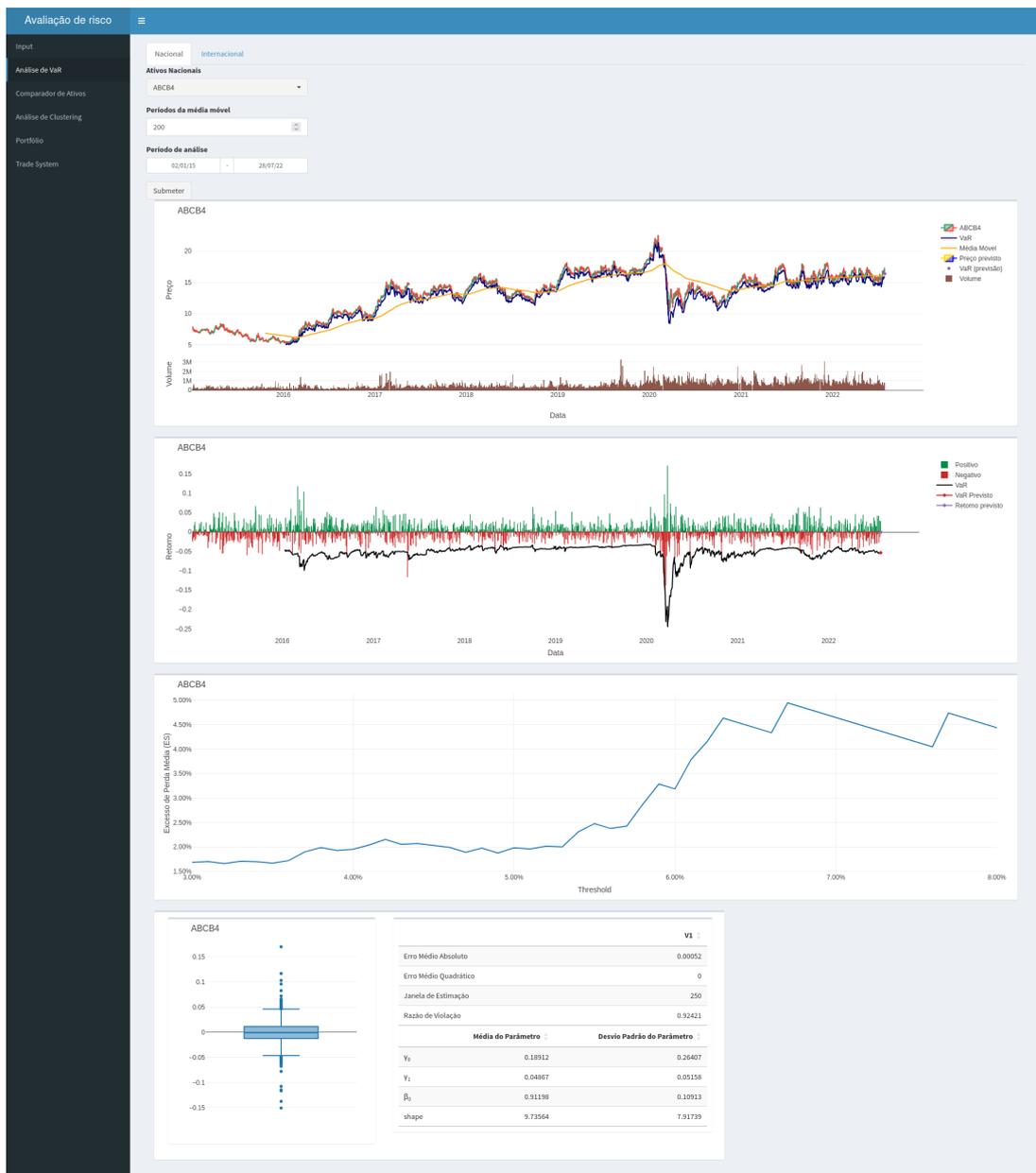


Figura 3 – Tela de Análise de VaR

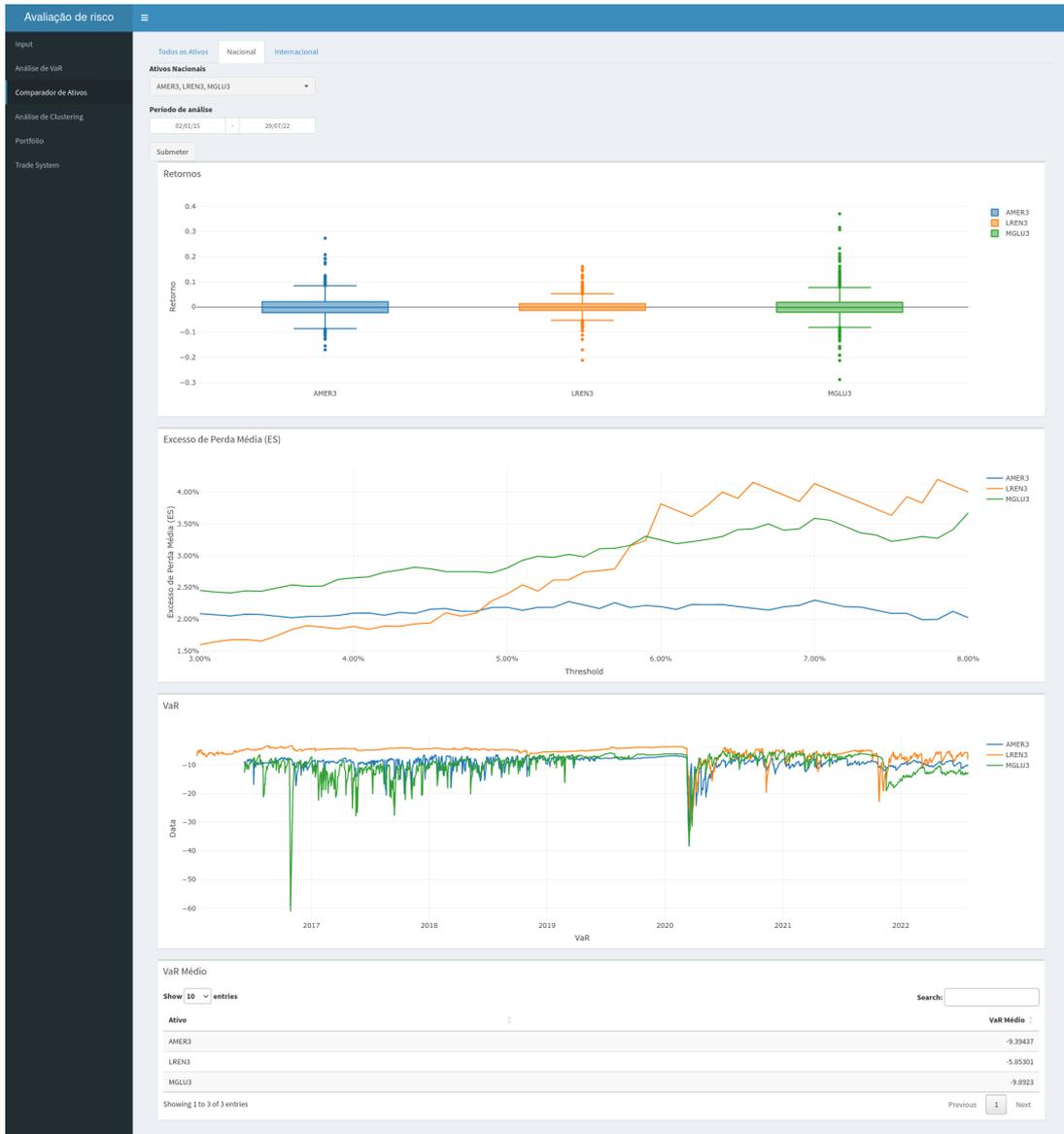


Figura 4 – Tela de comparação de ativos

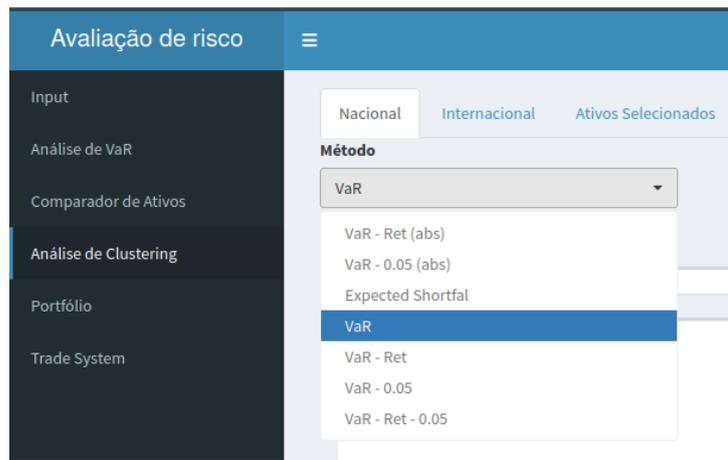


Figura 5 – Opções para Análise de Clustering

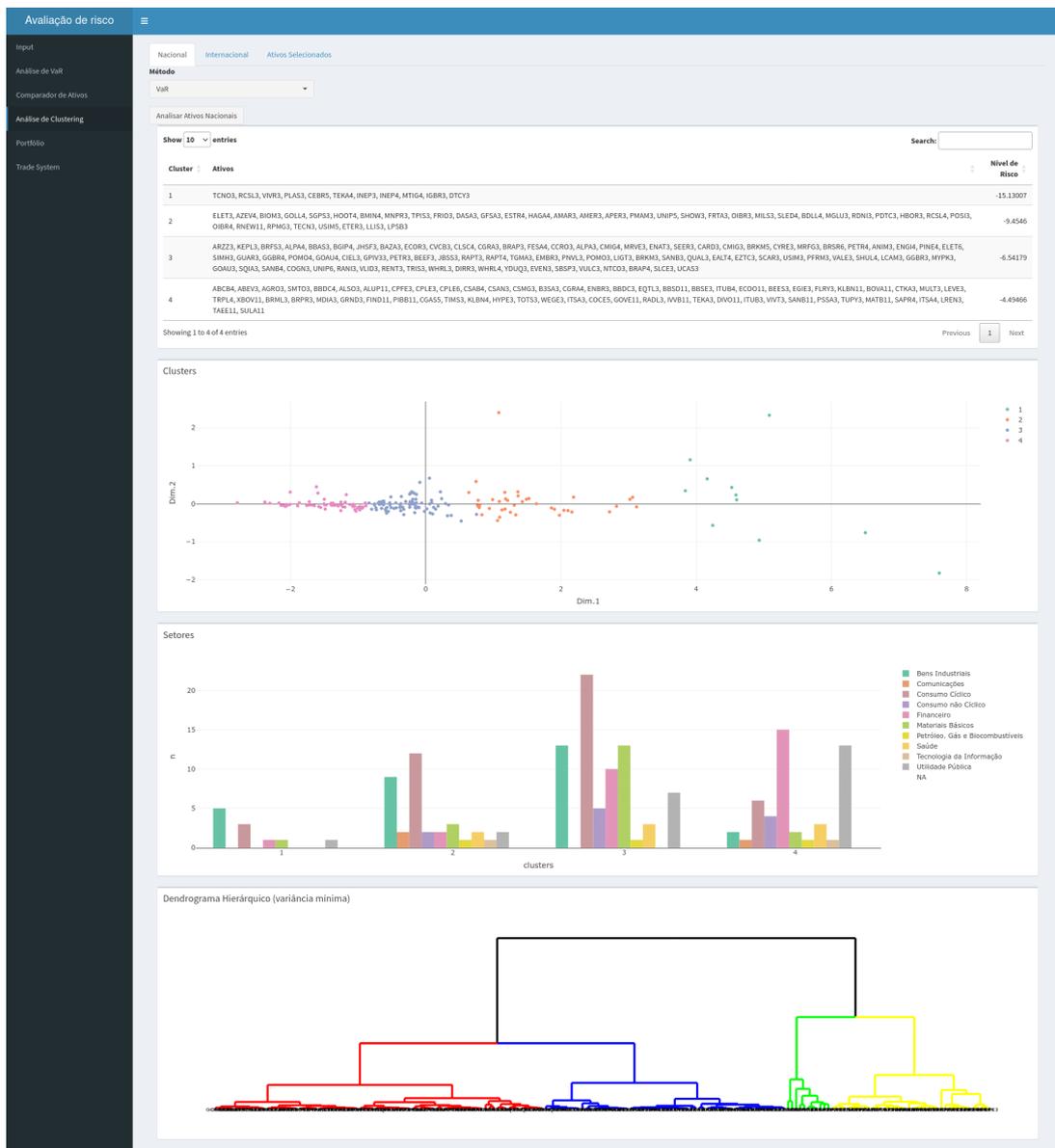


Figura 6 – Tela de Análise de Clustering

4.5 ABA PORTFÓLIO

A aba portfólio é dividida em três partes. A primeira permite a criação de carteiras que podem ser utilizadas para comparação e análise de VaR. Na formação de um portfólio, é possível escolher os pesos dos ativos na carteira ou utilizar a opção de ter pesos iguais (carteira ingênua). A Figura 7 ilustra o procedimento no *app*.

A segunda sub aba é semelhante a tela inicial de análise VaR (Seção 4.2), em que é possível escolher as janelas e estimação e um intervalo para a razão de violação do modelo para a análise GARCH-VaR do portfólio (ver Figura 8). Por fim, a visualização do GARCH-VaR do portfólio é vista na terceira sub aba (Figura 9).

The screenshot shows the 'Avaliação de risco' application interface. On the left is a dark sidebar with navigation options: 'Input', 'Análise de VaR', 'Comparador de Ativos', 'Análise de Clustering', 'Portfólio', and 'Trade System'. The main content area has a blue header with 'Avaliação de risco' and a hamburger menu icon. Below the header are three tabs: 'Criar Portfólio', 'Análise VaR', and 'Visualização'. The 'Criar Portfólio' tab is selected. It contains the following elements:

- Buttons: 'Criar Portfólio', 'Análise VaR', 'Visualização'.
- Field: 'Nome do portfólio' (empty text input).
- Field: 'Ativos' (dropdown menu showing 'AALR3, ABCB4, ABEV3').
- Field: 'Período' (date range from 27/10/16 to 12/01/22).
- Checkbox: 'Pesos iguais' (unchecked).
- Table with columns 'Ativo' and 'Peso':

	Ativo	Peso
1	AALR3	0.00
2	ABCB4	0.00
3	ABEV3	0.00
- Button: 'Criar Portfólio' at the bottom.

Figura 7 – Opções para aba de criação de portfólios

The screenshot shows the 'Avaliação de risco' application interface with the 'Análise VaR' sub-tab selected. The main content area has three tabs: 'Criar Portfólio', 'Análise VaR', and 'Visualização'. The 'Análise VaR' tab is selected. It contains the following elements:

- Buttons: 'Criar Portfólio', 'Análise VaR', 'Visualização'.
- Field: 'Carteira' (dropdown menu showing 'portfolio1').
- Field: 'Limite Inferior' (0.7).
- Field: 'Limite Superior' (1.4).
- Field: 'Período de análise' (date range from 29/04/11 to 28/07/22).
- Fields: 'Janela de estimação 1' through 'Janela de estimação 6' (values: 250, 300, 350, 400, 450, 500).
- Button: 'Submeter' at the bottom.

Figura 8 – Opções para modelo de portfólio

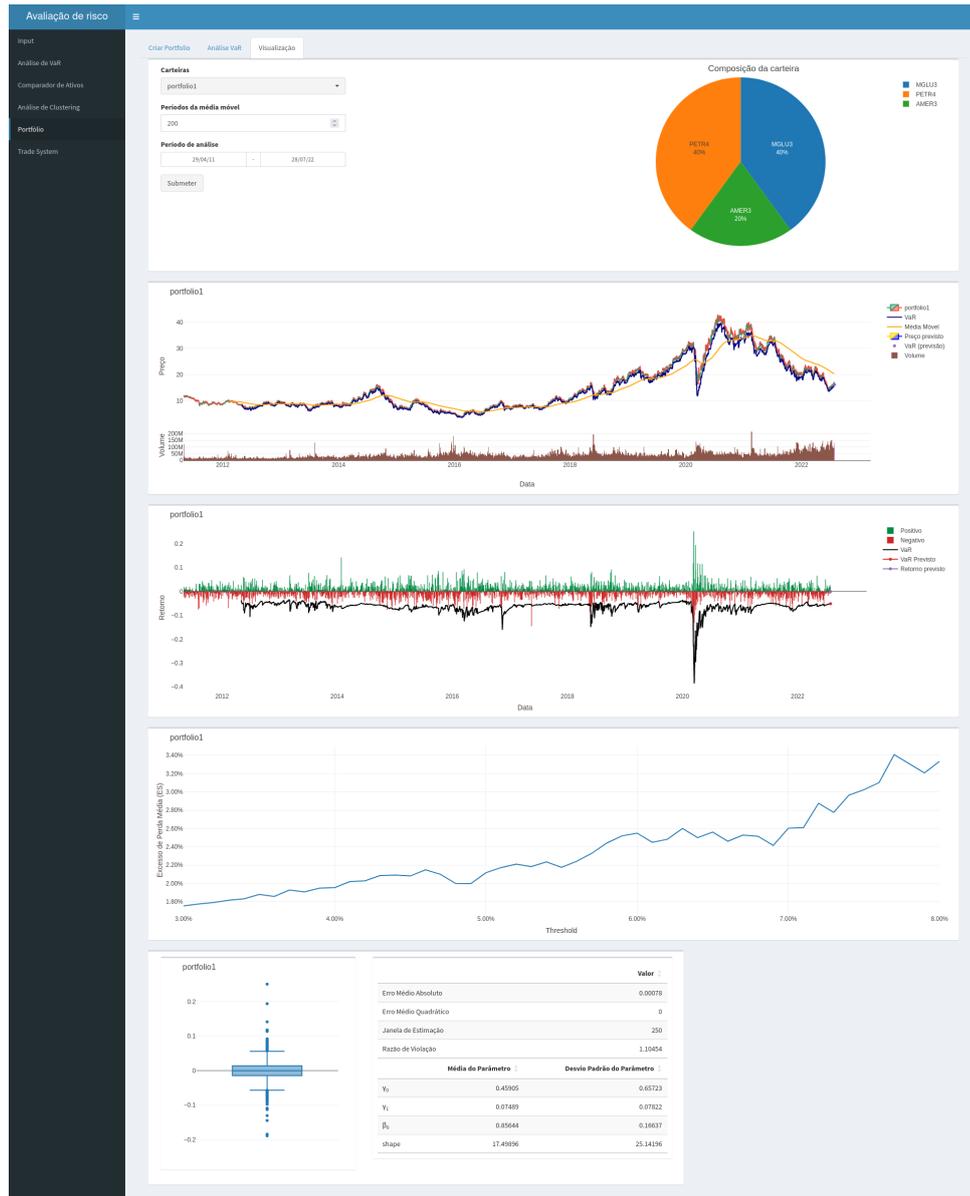


Figura 9 – Tela de visualização da carteira

4.6 ABA TRADE SYSTEM

A aba *trade system* permite que o usuário escolha um ativo nacional/internacional ou um portfólio e realize a definição das regras de compra/venda com *stop loss* definido com base na série VaR estimada pelo método GARCH-VaR. Nessa aba é possível escolher os sinais de compra e venda, definidos pelas relações entre o preço de fechamento (*Close*), o VaR e a media móvel de período escolhido. O usuário também define uma quantidade de ações a ser adquirida/vendida em caso de sinal de compra/venda, sendo possível escolher comprar todo o capital disponível ou vender todas as ações em carteira sempre que um sinal de compra/venda ocorrer. Depois de realizada a simulação o *app* mostra os gráficos de patrimônio ao longo do

tempo, quantidade de ações, valor alocado em ações, curva de lucro e saldo em caixa (sem alocação em investimento). A Figura 10 ilustra a estrutura da aba no *web app*.

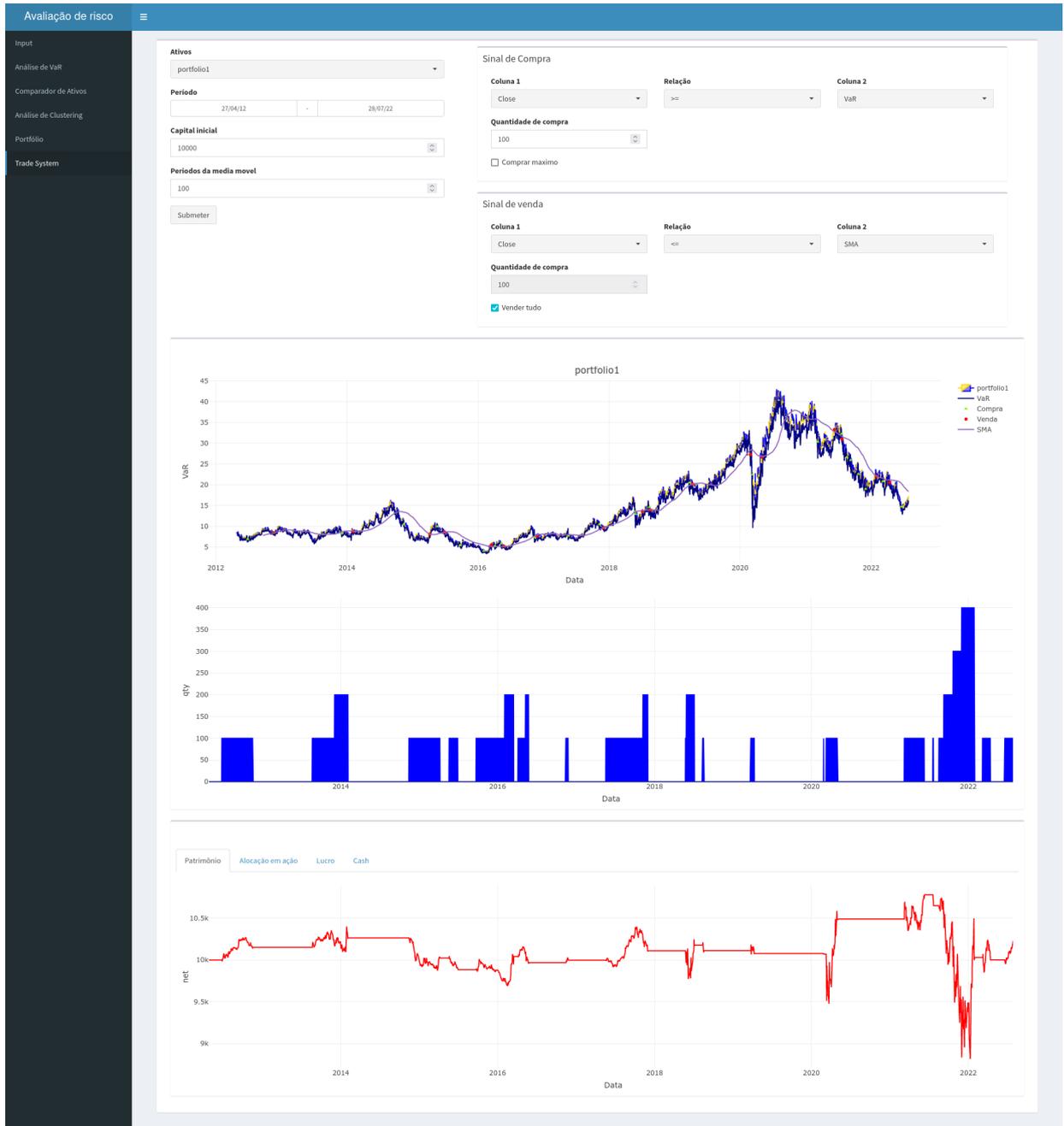


Figura 10 – Tela do *Trade System*

5 CONCLUSÃO

O presente trabalho apresenta as ferramentas utilizadas para o funcionamento de um aplicativo para visualização de séries de ativos financeiros e realização de simulações de sistemas de negociações baseada no GARCH-VaR (BERNARDINO et al., 2018), assim permitindo que sem conhecimento de programação o usuário possa realizar testes de estratégias de negociação.

A aba inicial permite que o usuário ajuste um modelo que se adéque melhor a serie dos retornos a partir da razão de violação. A aba de grupamento e comparação de ativos pode dar um direcionamento sobre como montar carteiras. A aba de portfólios permite que as carteiras sejam criadas e o modelo ajustado para que possa ser utilizada no *trade system*.

A revisão da literatura introduziu as principais ferramentas utilizadas pelo aplicativo. O pacote *shiny* do R torna fácil a criação de uma interface gráfica para o usuário. Assim permitindo a utilização da vasta biblioteca de pacotes estatísticos disponíveis para ajuste de modelos, simulações e criação de gráficos interativos.

Para dar continuidade ao trabalho apresentado os estudos futuros serão direcionados a otimização, melhorias gráficas e de adição de novas funcionalidades. Por conta das mudanças constante do mercado, é importante que o aplicativo seja constantemente atualizado.

REFERÊNCIAS

- ABDI, H.; WILLIAMS, L. Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2010.
- AL-FEDAGHI, S. Developing web applications. *International Journal of Software Engineering and Its Applications*, v. 5, 2011.
- ALLAIRE. *Application layout guide*. 2021. Disponível em: <<https://shiny.posit.co/r/articles/build/layout-guide/>>.
- ANGELIDIS, T.; BENOS, A.; DEGIANNAKIS, S. The use of garch models in var estimation. *Statistical methodology*, v. 1, n. 1-2, p. 105–128, 2004.
- ARTZNER, P.; DELBAEN, F.; EBER, J.-M.; HEATH, D. Coherent measures of risk. *Mathematical finance*, Wiley Online Library, v. 9, n. 3, p. 203–228, 1999.
- BEELEY, C. *Web Application Development with R Using Shiny*. United Kingdom: Packt Publishing Ltd, 2013.
- BENÍTEZ, R.; COLL-SERRANO, V.; BOLÓS, V. J. dear-shiny: an interactive web app for data envelopment analysis. *Sustainability*, MDPI, v. 13, n. 12, p. 6774, 2021.
- BERNARDINO, W.; BRITO, L.; OSPINA, R.; MELO, S. A garch-var investigation on the brazilian sectoral stock indices. *Brazilian Review of Finance*, v. 16, n. 4, p. 573–610, 2018.
- BOLLERSLEV, T. Generalized autoregressive conditional heteroskedasticity. *Journal of econometrics*, v. 31, n. 3, p. 307–327, 1986.
- BURGESS, N. An introduction to algorithmic trading: Opportunities & challenges within the systematic trading industry. *Available at SSRN 3466213*, 2019.
- CHANG, W.; Borges Ribeiro, B. *shinydashboard: Create Dashboards with 'Shiny'*. [S.l.], 2021. R package version 0.7.2. Disponível em: <<https://CRAN.R-project.org/package=shinydashboard>>.
- CHANG, W.; CHENG, J.; ALLAIRE, J.; SIEVERT, C.; SCHLOERKE, B.; XIE, Y.; ALLEN, J.; MCPHERSON, J.; DIPERT, A.; BORGES, B. *shiny: Web Application Framework for R*. [S.l.], 2023. R package version 1.7.4.1. Disponível em: <<https://CRAN.R-project.org/package=shiny>>.
- CONALLEN, J. Modeling web application architectures with uml. *Communications of the ACM*, v. 42, n. 10, p. 63–70, 1999.
- DUFFIE, D.; PAN, J. An overview of value at risk. *Journal of Derivatives*, v. 4, n. 3, p. 7–49, 1997.
- ENGLE, R. F. Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation. *Econometrica: Journal of the Econometric Society*, JSTOR, p. 987–1007, 1982.
- FXSTREET, L. C. *Definition of a Trading System - Learning Center*. 2023. Disponível em: <<https://learningcenter.fxstreet.com/education/learning-center/unit-3/chapter-1/definition-of-a-trading-system/index.html>>.

- GAO, F.; SONG, F. Estimation risk in garch var and es estimates. *Econometric Theory*, v. 24, n. 5, p. 1404–1424, 2008.
- HANSEN, P. R.; LUNDE, A. A forecast comparison of volatility models: does anything beat a garch (1, 1)? *Journal of applied econometrics*, v. 20, n. 7, p. 873–889, 2005.
- HARTIGAN, J. A.; WONG, M. A. Algorithm as 136: A k-means clustering algorithm. *Journal of the royal statistical society. series c (applied statistics)*, v. 28, n. 1, p. 100–108, 1979.
- KOENKER, R.; BASSETT, G. Regression quantiles. *Econometrica: journal of the Econometric Society*, v. 46, p. 33–50, 1978.
- KUMARI, S. K.; KUMAR, P.; PRIYA, J.; SURYA, S.; BHURJEE, A. K. Mean-value at risk portfolio selection problem using clustering technique : A case study. *AIP Conference Proceedings*, v. 2112, n. 1, p. 020178, 06 2019. ISSN 0094-243X. Disponível em: <<https://doi.org/10.1063/1.5112363>>.
- LINSMEIER, T. J.; PEARSON, N. D. *Risk measurement: An introduction to value at risk*. 1996.
- MAECHLER, M.; ROUSSEEUW, P.; STRUYF, A.; HUBERT, M.; HORNIK, K. *cluster: Cluster Analysis Basics and Extensions*. [S.l.], 2022. R package version 2.1.4 — For new features, see the 'Changelog' file (in the package source). Disponível em: <<https://CRAN.R-project.org/package=cluster>>.
- MAIER-PAAPE, S.; PLATEN, A. Backtest of trading systems on candle charts. *arXiv preprint arXiv:1412.5558*, 2014.
- MOHRI, M.; ROSTAMIZADEH, A.; TALWALKAR, A. *Foundations of Machine Learning*. [S.l.]: MIT press, 2018.
- NIELSEN, F.; NIELSEN, F. Hierarchical clustering. *Introduction to HPC with MPI for Data Science*, Springer, p. 195–211, 2016.
- PENA, J. M.; LOZANO, J. A.; LARRANAGA, P. An empirical comparison of four initialization methods for the k-means algorithm. *Pattern Recognition Letters*, v. 20, n. 10, p. 1027–1040, 1999.
- PETERSON, B. G.; ULRICH, J.; MACKIE, J.; HUMME, J.; CARL, P. *quantstrat: Quantitative Strategy Model Framework*. [S.l.], 2023. R package version 0.24.
- PRESSMAN, R. S. *Engenharia de Software : Uma Abordagem Profissional*. Porto Alegre: AMGH EDITORA LTDA., 2016.
- R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria, 2023. Disponível em: <<https://www.R-project.org/>>.
- SIEVERT, C. *Interactive Web-Based Data Visualization with R, plotly, and shiny*. Chapman and Hall/CRC, 2020. ISBN 9781138331457. Disponível em: <<https://plotly-r.com>>.
- WARD, J. H. Hierarchical grouping to optimize an objective function. *Journal of the American statistical association*, v. 58, n. 301, p. 236–244, 1963.
- ZIEGEL, E. R. Loss models: From data to decisions. *Technometrics*, v. 48, n. 2, p. 317, 2006.