CLARA PLORETTI CAPPATTO

RODRIGO ISKENDERIAN OLIVEIRA

# APRENDIZADO DE MÁQUINA APLICADO AO MERCADO FINANCEIRO BRASILEIRO

SÃO PAULO

2020

CLARA PLORETTI CAPPATTO
RODRIGO ISKENDERIAN OLIVEIRA
ORIENTADOR: THIAGO MARTINS

# APRENDIZADO DE MÁQUINA APLICADO AO MERCADO FINANCEIRO BRASILEIRO

Trabalho apresentado a Escola Politécnica da Universidade de São Paulo para obtenção do Título de Engenheiro(a) Mecatrônico(a)

SÃO PAULO

2020

Esse trabalho é dedicado às nossas famílias, que sempre estiveram presentes e nos apoiaram em nossas decisões.

AGRADECIMENTOS

Ao Professor Dr. Thiago Martins pelo apoio e incentivo a realização do nosso Trabalho de Conclusão de Curso.

Agradecemos também aos nossos amigos e colegas da Turma de 2016 da Mecatrônica, que sempre foram muito colaborativos e nos ajudaram nessa jornada e durante toda a graduação. Obrigada também às nossas famílias, que nos apoiaram não só ao longo da elaboração dessa tese, mas também por toda a nossa formação.

Agradecemos especialmente ao nosso amigo e colega de sala Felipe Machado pelos comentários ao longo do desenvolvimento da aplicação, que nos ajudaram muito na tomada de decisões.

Por fim, agradecemos aos amigos Guilherme Pouso e Victor Chacon e pela ajuda nas revisões da monografia.

RESUMO

O presente estudo apresenta o desenvolvimento de uma aplicação de gestão de uma carteira de ações do mercado brasileiro de ações (B3), no qual utiliza-se ferramentas de aprendizado de máquina. A aplicação realiza a previsão de preços diários para indicar sugestões de compra e venda. A previsão dos preços dos ativos é realizada utilizando os preços de dias anteriores em uma estrutura composta pela utilização de *Principal Component Analysis* (PCA) e pelo método de aprendizado supervisionado de *Support Vector Machine* (SVM) para regressão.

**Palavras-chave:** Previsão de ativos financeiros, Aprendizado supervisionado, Mercado brasileiro de ações.

ABSTRACT

The present study presents the development of an application for portfolio management of a group of stocks in the Brazilian Stock Market (B3), using machine learning's tools. The application forecasts the stocks daily prices to indicate buying and selling suggestions. The prediction of stock prices is made using the prices of previous days in a structure composed of Principal Component Analysis (PCA) and the Support Vector Machine (SVM) for regression, a machine learning method.

**Keywords:** Stocks price prediction, Machine learning, Brazilian stock market.

# Sumário

# Lista de Figuras

# Lista de Tabelas

# Lista de Equações

# Parte I

## Introdução

# 1 Introdução

O mundo das finanças é altamente codependente: um evento em uma parte do mundo é sentido e pode causar efeitos devastadores no mercado de ações do lado oposto do globo. Basta pensar na crise financeira de 2008, que se iniciou nos Estados Unidos com o estouro da bolha imobiliária, mas causou uma crise econômica global, derrubando os mercados da Ásia e Europa como consequência.

Um dos parâmetros mais utilizados para avaliar a situação econômica de um país é sua performance dentro do mercado financeiro mundial. A variação do índice da bolsa (no caso do Brasil, o Ibovespa) é uma referência para se compreender o estado da economia e sua integração com os outros mercados internacionais.

Considerando esses fatores e o interesse de instituições e agentes do mercado de encontrar oportunidades de investimento, a previsão de tendências de índices e ativos financeiros gerais dentro do mercado financeiro se apresenta como uma questão de grande importância.

O potencial ganho de antecipar, com sucesso, a direção de ativos é alto. Um grande exemplo deste fato é o fundo americano "*Renaissance Technologies*". A empresa, especializada no uso de modelos quantitativos e análises matemáticas e estatísticas para investimentos, é vista como um dos fundos mais bem sucedidos no mundo, com um histórico de rentabilidade exemplar. Os modelos utilizados pelo fundo consistem, essencialmente, na análise de grandes quantidades de dados e a busca de padrões para realizar previsões.

Existe, no entanto, uma grande dificuldade em antever tais tendências. Segundo (Abu-Mostafa & Atiya, 1996), a previsão de séries financeiras constitui um problema desafiador, pois estes sinais são muito ruidosos, não-estacionários e deterministicamente caóticos. Considerando essas características, a modelagem através de redes neurais aparece como uma solução interessante para esse problema, visto que essa técnica é capaz de mapear dados não lineares sem realizar grandes prerrogativas, como descrito por (Haykin, 1999).

Até o começo da década passada, o modelo de treinamento de redes neurais principalmente utilizado para prever séries temporais financeiras era o *back-propagation* (BP), desenvolvido por (Linnainmaa, 1976). Em 1995, foi desenvolvido

o algoritmo chamado de *Support Vector Machine* (SVM) por (Vapnik & Cortes, Support-Vector Networks, 1995), e este é utilizado até hoje para modelar esse tipo de problema ou para complementar o funcionamento de outros algoritmos, como realizado (Zhang, Patuwo, & Y. Hu, 1998).

Apesar dos resultados positivos de muitas pesquisas anteriores, muitos economistas ainda afirmam que existe uma impossibilidade de se prever com eficácia tendências dentro do mercado financeiro, por este ser regido pela "hipótese do mercado eficiente". Esta afirma que as direções que ativos tomam ao longo do tempo não dependem de fator algum e são, por consequência, aleatórias.

Além disso, existe ainda um desafio maior a ser superado dentro do escopo desse trabalho. (Leahy, Carciente, Stanley, & Kenett, 2014) indicam que as ações do mercado brasileiro possuem pouca correlação com o índice Bovespa e, por ser se tratar de um país emergente, as correlações são mais difíceis de serem definidas. No entanto, ele também afirma que existe uma grande correlação entre as ações que compõem o mercado e possivelmente há uma boa correlação entre essas ações e ativos de mesmo segmento em mercados internacionais.

Este trabalho irá, ao longo dos próximos capítulos, portanto, apresentar testes iniciais e resultados preliminares, que tem por objetivo avaliar a aplicação da hipótese de previsão ao mercado brasileiro. A partir destes, os seguintes capítulos irão estabelecer uma metodologia para a escolha de ferramentas, o estabelecimento e desenvolvimento da solução e, por fim, os resultados obtidos.

# 2 Objetivos

O principal objetivo deste trabalho é desenvolver um agente operador do mercado de ações (B3 – Bovespa) que seja capaz de superar o *benchmark* do mercado (índice Bovespa). Para isso, o projeto analisa o comportamento de ativos do mercado financeiro brasileiro. Além de superar o *benchmark*, tem-se como objetivo secundário a obtenção de lucro ainda que em cenários desfavoráveis para o mercado

Espera-se que o algoritmo desenvolvido seja capaz de processar grandes quantidades de dados históricos de ativos nacionais a serem estabelecidos posteriormente, retornando sugestões de compra e venda a partir de previsões para determinados ativos.

Dados os pontos apresentados anteriormente, a linguagem escolhida para o desenvolvimento da solução foi Python, que possui diversas bibliotecas com capacidade de lidar com o processamento de grandes quantidades de dados e ferramentas eficientes para trabalhar com previsão.

# 3 Requisitos de Projeto

O projeto a ser desenvolvido será, em sua essência, uma aplicação para facilitar e otimizar a ação de um usuário dentro do mercado de ações. O grupo, portanto, criou uma persona, e foi a partir das necessidades desta que foram estabelecidos os requisitos de projeto. Além disso, foi necessário considerar aspectos jurídicos de funcionamento e entrada no mercado (estabelecidos pela CVM - Comissão de Valores Mobiliários) e requisitos relacionados às corretoras, pelas quais o sistema opera.

O usuário alvo da aplicação é a pessoa que não tem ou tem pouco conhecimento prévio quando se diz respeito a mercado financeiro, ou acredita que investir no mercado de ações seja algo intimidador, portanto, não o faz, apesar de ter interesse. Além disso, ele teria interesse em saber o destino do seu investimento, saber para onde seu dinheiro vai, e poder ter poder de decisão sobre as empresas ou setores em que vai investir, a partir da estratégia fornecida pelo sistema.

Dessa forma, os requisitos estabelecidos para o sistema (RS) até a etapa atual são:

- RS0 - O sistema deve ser capaz de alcançar o *benchmark* estabelecido, no caso o índice Bovespa, sem conhecimento prévio da composição do índice;
- RS1 - O sistema deve possuir uma base de cálculo universal e que forneça informações específicas para o usuário;
- RS2 - O sistema precisa ser de fácil compreensão, permitindo com que o usuário entenda facilmente a proposta de ação que é passada para ele;
- RS3 - O sistema precisa reconhecer que a ação não foi tomada para poder adaptar a próxima sugestão de investimento;
- RS4 - O sistema deve estar de acordo com as normas estabelecidas pela Comissão de Valores Mobiliários;
- RS5 - O sistema deve levar a um desempenho satisfatório de rendimento, ainda considerando períodos de estresse de mercado.

# Parte II

## Estado da Arte

# 4 O Estado da Arte

Até o começo da década passada, o modelo de treinamento de redes neurais principalmente utilizado para prever séries temporais financeiras era o *back-propagation* (BP), desenvolvido por (Linnainmaa, 1976), que possuía uma arquitetura simples e grande capacidade de resolver problemas de natureza ruidosa, como descrito por (Tay & Cao, 2001). Em 1995, foi desenvolvido o algoritmo chamado de *Support Vector Machine* (Vapnik & Cortes, Support-Vector Networks, 1995), e este é utilizado para modelar problemas de classificação e sua extensão *Support Vector Regression* é utilizado para previsão de séries temporais ruidosas ou para complementar o funcionamento de outros algoritmos, como realizado por (Shen, Jiang, & Zhang, 2012).

Nesta revisão, aborda-se de maneira geral o desenvolvimento dos algoritmos de redes neurais para previsão de séries financeiras, bem como a seleção e filtragem dos dados utilizados para treinamento das inteligências artificiais. Por fim, analisa-se os resultados obtidos pelos autores e verifica-se a viabilidade de elaboração de um algoritmo para prever o comportamento do mercado de ações brasileiro.

## 4.1 Modelos Matemáticos

Diversos profissionais vêm, por décadas, tentando prever tendências dentro do mercado financeiro, mesmo antes do desenvolvimento em massa de computadores, linguagens de programação e as famosas redes neurais. A análise quantitativa tem sido utilizada como uma ferramenta para processamento de dados e identificação de probabilidades estatísticas em qualquer mercado.

Em 1988, o fundador da *Renaissance Technologies*, um matemático premiado e ex-decodificador da Guerra Fria, James Simons, estabeleceu seu principal e mais famoso fundo, o Medallion, que opera usando métodos matemáticos e estatísticos para manipular grandes quantidades de dados. Seu algoritmo inicial foi baseado nos modelos de Leonard Baum, principalmente o algoritmo Baum-Welch e o modelo de Hidden Markov. O primeiro descreve a probabilidade das chamadas variáveis "ocultas" que definem um estado oculto atual; e o segundo encontra a estimativa de máxima verossimilhança dos parâmetros para o modelo de Markov.

Ao longo dos anos, o uso de estatística simples tem sido substituído por redes neurais baseadas em computadores e sistemas inteligentes de processamento e filtragem de dados, que se provam mais eficientes a longo prazo.

## 4.2 Redes Neurais (RNs)

Ainda segundo (Tay & Cao, 2001), diferentemente de dados estatísticos, redes neurais são modelos direcionados pelos dados de entrada e trabalham bem com dados não paramétricos, além de conseguirem ser tolerantes a ruídos e mais flexíveis para aprenderem a dinâmica de sistemas utilizando novos tipos de dados. Utilizando abordagem de redes neurais, os autores (Kaastra, 1994), (Zhanga & Hu b, 1998) e (W.-C, Urban, & Baildridge, 1996) demonstraram que é possível obter uma performance superior aos métodos estatísticos tradicionais, obtendo erros inferiores através de diversos indicativos, como Erro Quadrado Médio (EQM) e *Erro Absoluto Médio* (EAM).

Atualmente, a técnica de *Support Vector Machine* (Vapnik & Cortes, Support-Vector Networks, 1995) e sua extensão *Support Vector Regression* (SVR) se apresenta como uma das mais promissoras para o treinamento de redes neurais com propósito de prever séries temporais financeiras.

### 4.2.1 Back-propagation (BP)

O algoritmo conhecido como *back-propagation* é usado amplamente em *Machine Learning* no que se diz respeito ao treinamento de redes neurais para aprendizado por supervisionamento. Essencialmente, o algoritmo considera o gradiente existente entre a saída da rede neural e a saída esperada, com relação ao peso da rede, para um único par de entrada-saída. Este valor é constantemente atualizado, assim como o peso, com o objetivo de diminuir a diferença entre o valor obtido e o esperado.

Apesar de sua simplicidade, o *back-propagation* se mostra ineficiente ao ser comparado com outras formas de treinamento de redes neurais (RNs). (Gupta & Sexton, 1999) demonstram que um algoritmo genético utilizado para treinar uma rede caótica de séries temporais apresenta resultados superiores ao BP quanto a eficiência e efetividade, além de se mostrar mais simples de ser desenvolvido.

Francis (Tay & Cao, 2001) também comparou em seu artigo o uso do algoritmo de *back-propagation* ao *Support Vector Machine*, mencionado anteriormente. Ele conclui que este último método gera resultados mais precisos e satisfatórios do que os obtidos pela aplicação de BP quando se diz respeito a uma série temporal financeira.

Dessa forma, a despeito da larga utilização e da simplicidade de implementação do *back-propagation*, o grupo decidiu por descartar esta forma de treinamento de RNs em favor de alternativas mais eficientes.

### 4.2.2 Support Vector Machine (SVM)

Dentro do universo de *Machine Learning*, *Support Vector Machine* (SVM) é um modelo de aprendizado supervisionado de redes neurais que analisa dados para resolução de problemas relacionados tanto a classificação quanto a regressão.

Originalmente desenvolvida com o foco de reconhecimento de padrões, o algoritmo possui resultados melhores comparados a outras redes neurais quando se trata da performance de generalização. A performance obtida pelo uso do SVM e sua extensão *Support Vector Regression* (SVR) para resolução de problemas de regressões não lineares é mostrada como excelente por (Müller, et al., 1997).

Como descrito por (Tay & Cao, 2001), esse método SVR funciona minimizando uma função de risco ao invés da taxa de erro, o que faz com que o modelo seja capaz de ser usados em aplicações não paramétricas sob dinâmica complexa. Ele proporciona ao usuário a possibilidade de definir a tolerância ao erro desejada dentro do modelo.

## 4.3 Principal Component Analysis (PCA)

A performance de uma rede baseada em SVR pode ser melhorada quando combinada com *Principal Componente Analysis*, um método de extração de dados que transforma vetores de grandes dimensões em componentes principais não correlacionadas entre si. Para determinar a influência do PCA na previsão de séries temporais financeiras, (Chowdhury, Hossain, & Chakravarty, 2018) compararam os resultados obtidos por uma rede neural treinada somente por SVM e por outra que também utilizava PCA.

Os autores analisaram diferentes métricas de erro calculadas a partir do resultado teórico obtido e o real, tendo como base de dados os índices de 4 diferentes ações ao longo de 3600 dias. A conclusão foi de que a combinação SVR e PCA gera uma porcentagem de erro de previsão menor do que aquela obtida ao se implementar somente SVM no treinamento de redes neurais.

## 4.4 Aquisição de Dados

Antes de definir metodologia e a técnica a ser utilizada no projeto, é importante definir as informações que serão fornecidas ao sistema para treinamento e previsão das séries temporais. Segundo (Enke & Thawornwong, 2005), informações históricas disponíveis ao público podem auxiliar na previsão do preço de ativos no futuro.

Um dos principais problemas ao se tratar do mercado financeiro é o fato de que, como coloca (Abu-Mostafa & Atiya, 1996), séries temporais financeiras estão entre os sinais mais ruidosos de se prever. Assim, faz-se necessária uma filtragem ou pré-seleção de informações.

Em seu trabalho, (Enke & Thawornwong, 2005) propôs uma estratégia que consiste em computar medidas que podem ser utilizadas para calcular e quantificar, a partir de uma equação definida, a relevância de variáveis de um conjunto de dados extenso. Tais variáveis são ranqueadas de acordo com seu respectivo ganho de informação, e as que possuem maior ganho são selecionadas como entradas do sistema.

Os resultados obtidos pelo autor foram satisfatórios e mostraram que este método, combinado ao uso de redes neurais, é capaz de melhorar a performance do agente operador sem aumento de risco.

Yaser (Abu-Mostafa & Atiya, 1996), por sua vez, selecionou parâmetros de entrada para a rede neural seguindo uma análise em duas partes: fundamental e técnica. A fundamental consiste basicamente no estudo de fatores e notícias econômicas que podem influenciar oferta e procura de ativos financeiros. Índices fundamentais, como a inflação, déficit orçamentário, reservas do Banco Federal, etc., também podem servir como entradas para uma rede neural.

A análise técnica, por outro lado, envolve a observação de padrões de preços, assim como o volume de compras e vendas. Isso pode ser feito através de alguns índices, explicados e calculados por Yaser em detalhe: '*The Moving Averages Indicator*', '*Trading Volume and Open Interest Indicators*' e 'Volatilidade'. O primeiro avalia essencialmente, através de uma fórmula estipulada, se o preço de um ativo tende a subir ou a descer. O segundo mede a intensidade com a qual o preço varia: altos volumes de troca indicam o início de um movimento de preço no mercado. Por fim, volatilidade mede a intensidade de atividade do mercado, sendo volatilidades maiores são atreladas a grandes variações de preço.

# Parte III

## Metodologia

# 5 Arquitetura Inicial

O projeto desenvolvido é, essencialmente, uma ferramenta que fornece indicações de compra e venda de ativos de acordo com suas preferências. A arquitetura básica da aplicação está apresentada abaixo.

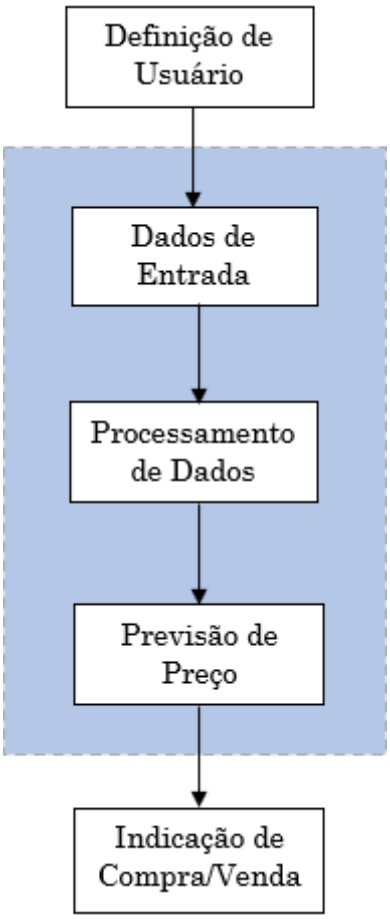


Figura 1 - Arquitetura básico da aplicação (Metodologia)

As definições de usuário são estabelecidas seguindo os requisitos da aplicação e com elas são definidos os ativos os quais serão realizadas as previsões de preço. As sugestões de compra e venda seguem uma estratégia simples: caso o preço da ação seja maior no dia seguinte, será dada a indicação de compra; caso seja menor, será dada a indicação de venda.

Os processos que constituem a região delimitada na Figura 1 são detalhados nas seções 5 e 6, onde são apresentados as análises de redução de dimensionalidade do problema, escolha e calibração do algoritmo de previsão e seleção de dados de entrada.

## 5.1 Principal Component Analisys (PCA):

A aplicação do PCA, desenvolvido por (Pearson, 1901), permite delimitar e definir correlações entre diferentes variáveis. Se tais correlações existem e são significativas, é possível reduzir o tamanho do problema, obtendo um espaço dimensional menor, mas que possui a maior parte das informações iniciais.

A primeira etapa do processo de análise de componentes principais é normalizar os dados importados. Compõe-se uma nova matriz de dados utilizando os valores normalizados e em seguida, é definida a matriz de covariância, na qual cada elemento representa a covariância entre duas características. A covariância entre dois fatores é calculada por:

Equação 1 - Covariância entre dois fatores

$$\sigma_{jk} = \frac{1}{n-1}\sum_{i=1}^{N}\left(x_{ij} - \bar{x}_j\right)\left(x_{ik} - \bar{x}_k\right)$$

A matriz de covariância, por sua vez é definida pela seguinte equação matricial:

Equação 2 – Matriz de covariância

$$C = \frac{1}{n-1}\left((X - \bar{x})^T(X - \bar{x})\right) \text{ , com } \bar{x} = \sum_{k=1}^{n} x_i$$

Na equação acima, X é matriz composta pelos dados de entrada. Em seguida, busca-se os auto-valores (λ) e auto-vetores que a compõem. Os auto-valores de C são as raízes da equação características.

Equação 3 - Polinômio característico de C

$$p_C(x) = \det(C - xI), \text{com I sendo a matriz identidade}$$

Para obter o conjunto de auto-vetores V(λ), temos que Vλ é o conjunto solução do sistema a seguir:

Equação 4 - Sistema de equações para obter o conjunto de auto-vetores

$$(C - \lambda I)X = 0$$

Ordena-se os auto-valores por módulo e tem-se que a variância explicada por cada auto-valor λ é dada por:

Equação 5 – Variância explicada pelos auto-valores

$$\text{Variância Explicada por } \lambda_j = \frac{\lambda}{\sum_j \lambda_j}$$

Cada um dos conjuntos de auto-valores e auto-vetores, ordenados por módulo, compõe uma componente principal.

## 5.2 Support Vector Regression (SVR)

O método *Support Vector Regression* (SVR) consiste em uma extensão do método de classificação *Support Vector Machine* (SVM), proposto por (Vapnik & Cortes, Support-Vector Networks, 1995). Diferente de regressões simples que tenta minimizar a taxa de erro, o SVR busca manter o erro dentro de uma margem de tolerância estabelecida (ε). Tendo um conjunto de treinamento composto pelos pares $(x_i, y_i), i = 1,2, \dots, n$, onde $x_i \in R^m$, a função objetiva do SVR consiste em minimizar R.

Equação 6 –Risco R

$$R = \frac{1}{2}\|w\|^2 + C\left(\sum_{i=1}^{n} |y - f(x,w)|_\varepsilon\right)$$

Onde $f(x,w)$ é função de regressão linear gerada pelo SVR, onde x são os dados de entrada e w vetor de peso.

Equação 7 – Função de regressão linear (SVR)

$$f(x,w) = w^T x + b$$

Definida por Vapnik, a função linear de custo ε-insensível $|y - f(x,w)|_\varepsilon$ utiliza os valores alvos y e a regressão linear.

Equação 8 - Função linear de custo ε-insensível de Vapnik

$$|y - f(x,w)|_\varepsilon = \begin{cases} 0, & |y - f(x,w)|_\varepsilon \text{ dentro da margem } \varepsilon \\ |y - f(x,w)|_\varepsilon - \varepsilon & \text{caso contrário} \end{cases}$$

Como apontado por (Chowdhury, Hossain, & Chakravarty, 2018), a estimativa de $f(x,w)$ ocorre através da minimização de $\|w\|^2$ e da soma dos custos da função ε-insensível (Equação 6). O trade-off entre o erro de aproximação e o módulo do vetor de peso $\|w\|$ é controlado pela constante C, definida previamente. A equação de Risco R (Equação 6), pode ser reescrita em função das chamadas variáveis de folga $\xi_i$ e $\xi_i^*$

que consideram pontos foram da margem ε, sendo uma por estar mais de ε abaixo do valor alvo e outra por estar mais de ε acima.

Equação 9 - Risco R (em função de variáveis de folga)

$$R = \frac{1}{2}\|w\|^2 + C\left(\sum_{i=1}^{n}(\xi_i + \xi_i^*)\right), \text{com as seguintes condições de contorno:}$$

$$(w^T x_i + b) - y_i \leq \varepsilon + \xi_i$$

$$y_i - (w^T x_i + b) \leq \varepsilon + \xi_i^*$$

$$\xi_i, \xi_i^* \geq 0, \qquad i = 1,2,\dots,m$$

O problema de otimização (Equação 9) é resolvido utilizando a teoria Lagrangiana e as condições de Karush-Kuhn-Tucker, obtendo assim os fatores de peso otimizados. Para trabalhar com dados de entrada cuja expressividade excede o espaço linear, o SVR mapeia os dados de entrada $x_i \in R^m$ em espaço dimensional superior $\phi: R^n \rightarrow R^N$ para aplicar uma regressão linear neste novo espaço. Para realizar esse mapeamento, utiliza-se uma função kernel $K(x_i, x_j)$, a função kernel mais popular, que será usada nesse projeto, é a *Radial Basis Function* (RBF):

Equação 10 – Kernel RBF

$$K(x_i, x_j) = \exp\left(-\gamma\|x_i - x_j\|^2\right)$$

Na Equação 10, tem-se que γ é constante da função kernel. Os parâmetros do SVR a serem definidos são, portanto, γ e a constante de regularização C.

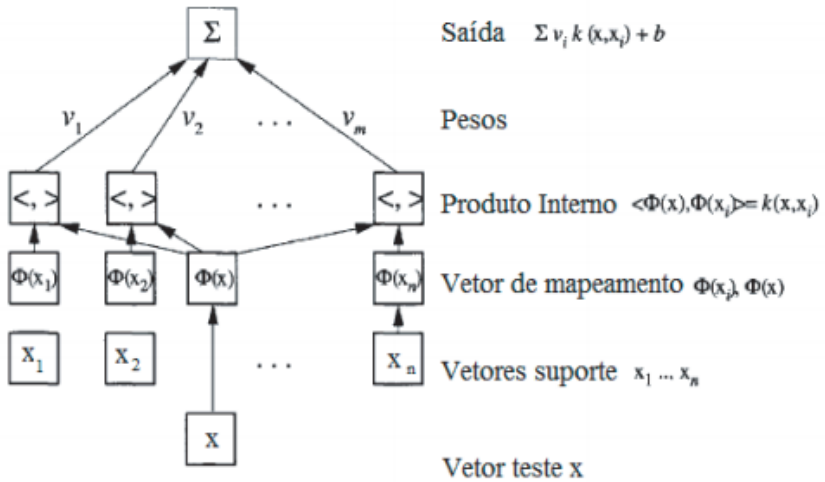


Figura 2 – Arquitetura do SVR (Dias, 2007)

# Parte IV

## Projeto

# 6 Arquitetura Final

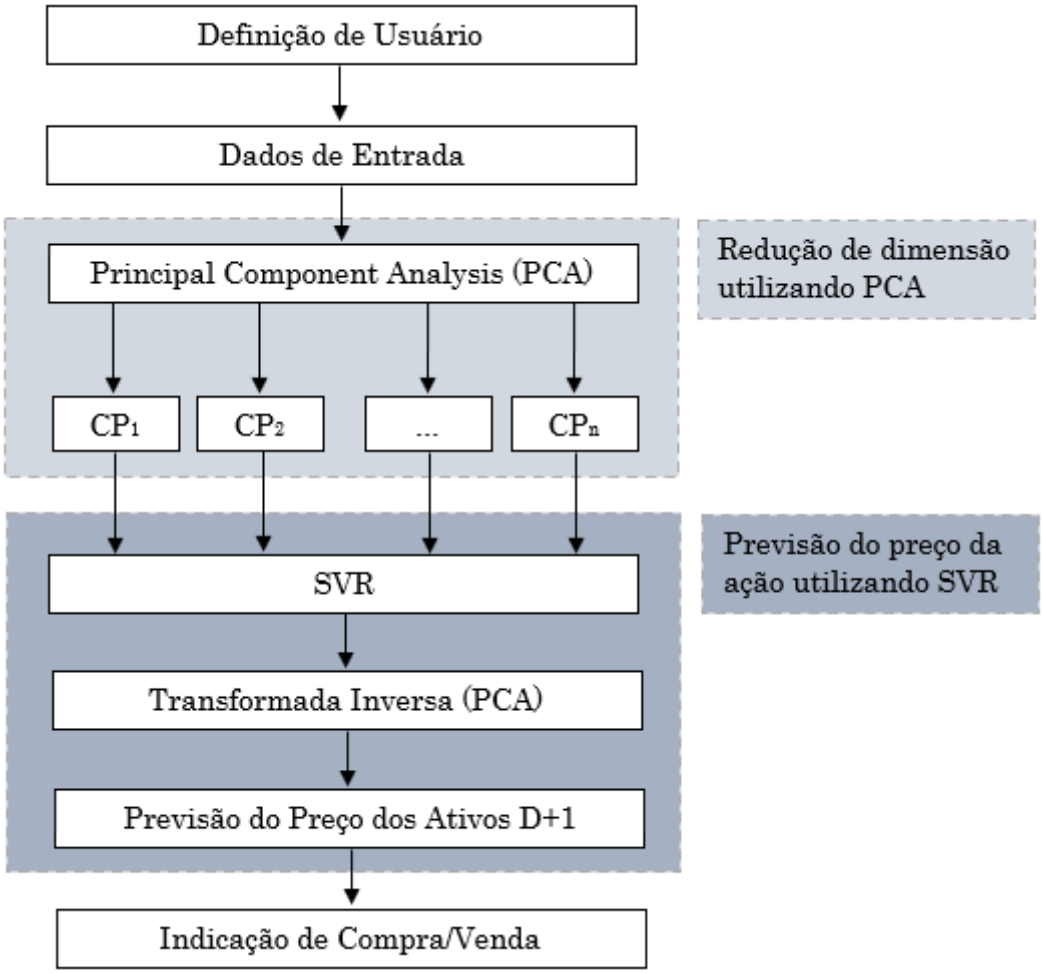


Figura 3 - Arquitetura final da aplicação

Partindo da Arquitetura Inicial descrita no capítulo 5, foram desenvolvidas as estruturas de tratamento de dados e de previsão de preços chegando até a arquitetura final.

Através das definições de usuário e dos dados de entrada dos ativos é realizada uma redução dimensional utilizando *Principal Component Analysis* (PCA) e são selecionadas as componentes para previsão utilizando o método de *Support Vector Regression*. Previsto os valores das componentes, utiliza-se a transformada inversa para obter os preços dos ativos em D+1 e consequentemente aplica-se a estratégia de compra e venda.

## 6.1 Seleção de Dados

Para realizar a escolha dos ativos que irão compor a aplicação foi realizada uma análise de liquidez entre os anos de 2016 e 2019. A análise consistiu numa verificação do volume médio negociado nesse período. O cálculo do volume médio por ativo é descrito na Equação 11, onde $VM_{i,B3}$ é o volume financeiro médio diário do ano i total do mercado a vista $VM_{i,ativo}$ é o volume financeiro médio diário do ano i do ativo.

Equação 11 – Volume médio negociado por ativo

$$VM_{ativo} = \sum_{i=2016}^{N=2019} \frac{VM_{i,ativo}}{VM_{i,B3}}$$

Foram escolhidas 17 ações da B3 classificadas em 5 setores (Tabela 1). Destaca-se a presença de empresas estatais como Petrobras (PETR4) e Vale (VALE3) que juntas compõem cerca de 15% do volume médio negociado no período e de bancos como Bradesco (BBDC3) e Banco do Brasil (BBAS3) que juntas compõem cerca de 10% do volume médio negociado no período.

Tabela 1 - Ativos disponíveis na aplicação

| Setores | | | | |
|---|---|---|---|---|
| Financeiro | Constr. Civil e Transporte | Indústria de Base | Consumo | Alimentícia |
| ITUB4 | GOLL4 | VALE3 | VVAR3 | ABEV3 |
| BBDC4 | MRVE3 | ELET3 | HGTX3 | MRFG3 |
| BBAS3 | EMBR3 | PETR4 | | JBSS3 |
| SANB11 | | CSNA3 | | BRFS3 |
| BRAP4 | | | | |

Além das ações das tradicionais empresas do setor financeiro e de base, destaca-se também a seleção alguns ativos do setor de consumo que obtiveram um bom desempenho no ano de 2019 como é o caso das varejistas Hering (HGTX3) e a Via Varejo (VVAR3). Os históricos de preços de fechamento são mostrados nas Figura 4 a Figura 8.

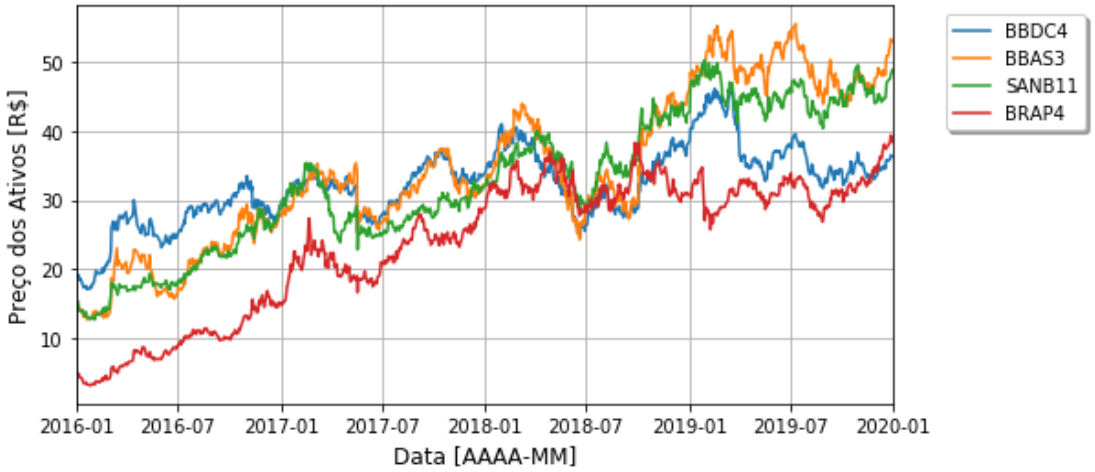


Figura 4 - Histórico de Preços (Financeiro)

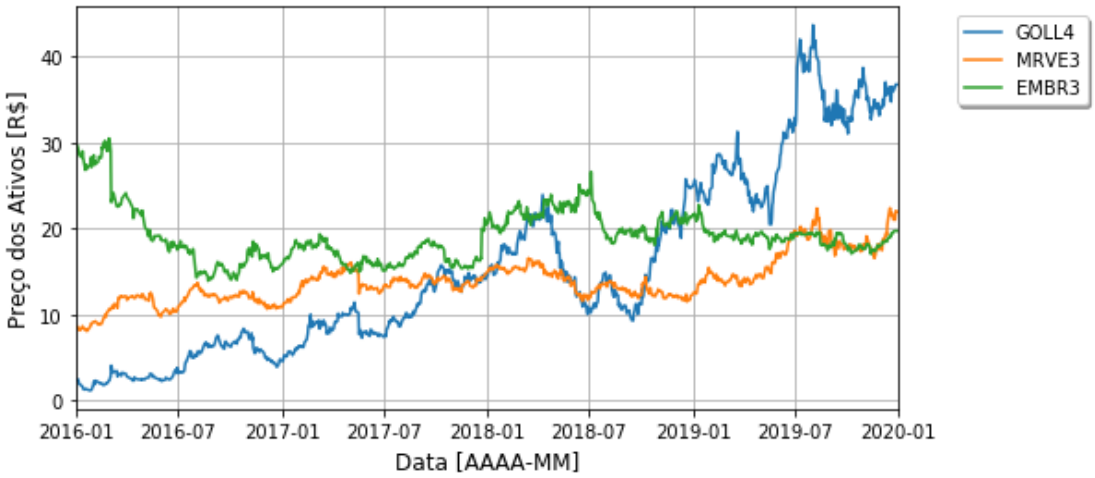


Figura 5 – Histórico de Preços (Construção Civil e Transporte)


Figura 6 - Histórico de Preços (Indústria de Base)

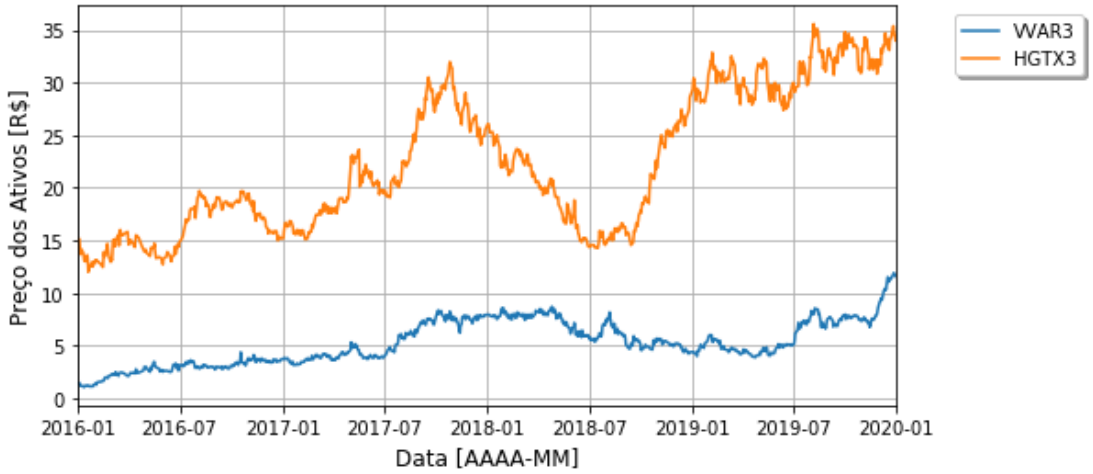


Figura 7 - Histórico de Preços (Consumo)

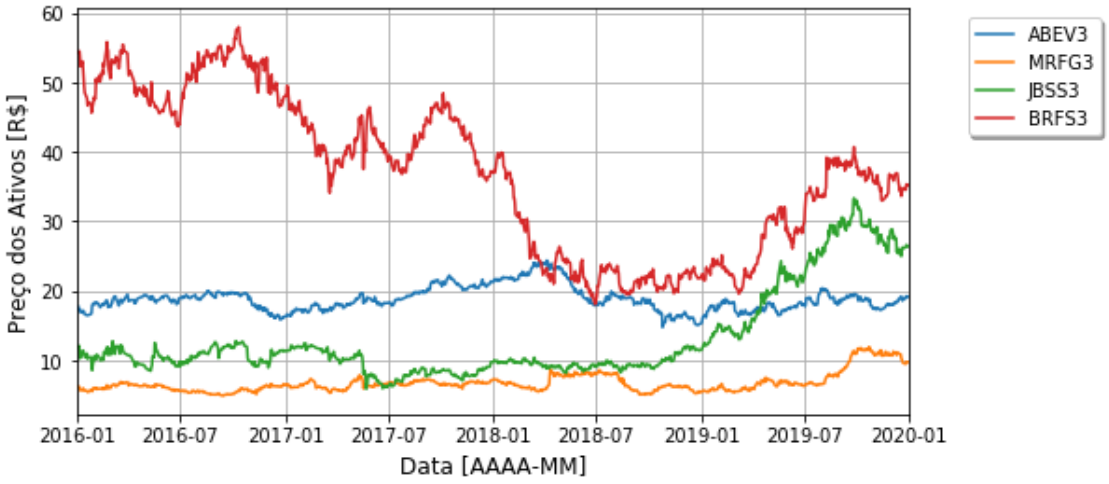


Figura 8 - Histórico de Preços (Alimentos)

Na Figura 4, é possível verificar uma boa correlação entre todos os ativos, o que já era esperado para o setor de financeiro. Na Figura 6, verifica-se boa correlação entre os ativos que compõem a indústria de base. Observa-se também boa correlação entre os ativos de transporte aéreo (EMBR3 E GOL4) na Figura 5 e uma boa correlação entre ações da indústria de frigoríficos (MRFG3 e BRFS3), principalmente nos últimos dois anos.

No período de 2018, verifica-se um fenômeno de queda em grande parte dos ativos que pode ser explicada pelo ano ser caracterizado pelas eleições presidenciais no Brasil. Após as eleições, observa-se um movimento de subida na maioria dos ativos, ainda que algumas ações possuam grande volatilidade, devido as políticas de carácter neoliberal e a expectativa de crescimento do país com o resultado das eleições.

## 6.2 Aplicação do PCA

Definidos os dados de entrada, foi realizada a análise de componentes principais.

### 6.2.1 Resultados da aplicação do PCA

Os dados de entrada foram normalizados e foi realizada a aplicação do PCA seguindo a abordagem apresentada na seção 5.1. A variância acumulada com as componentes principais é apresentada na Figura 9.

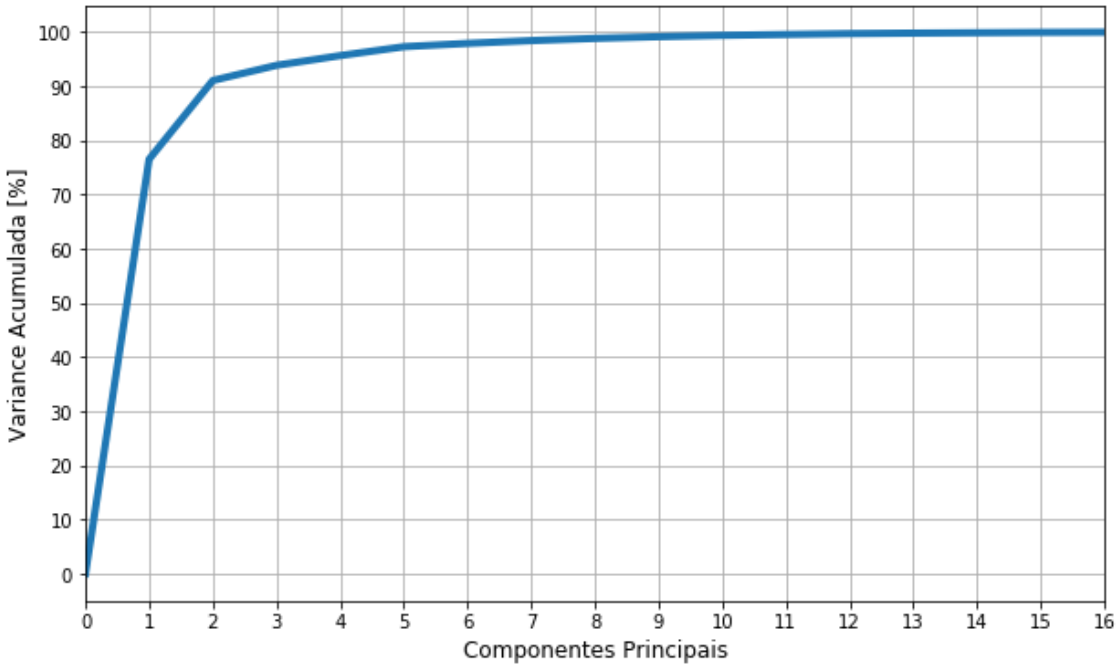


Figura 9 - Variância acumulada das componentes principais

A partir dos resultados obtidos, conclui-se que para o período de análise, as 4 primeiras componentes principais são responsáveis por aproximadamente 95% da variância total dos dados e que com as 7 primeiras componentes são capazes de descrever 98% da variância. Os resultados demonstraram que a dimensão do problema pode ser reduzida mantendo um alto nível de confiança.

Ao analisar a contribuição absoluta de cada uma das séries na composição das componentes principais (Figura 10), verificou-se que os ativos do setor financeiro são bem correlacionados com as componentes 1, 3 e 4. Destaca-se também a boa correlação de ativos do setor alimentício nas componentes 2, 3 e 4. A Figura 11

apresenta as componentes principais posteriores, nessas pode-se destacar boa correlação com ativos do setor de consumo e alimentício.

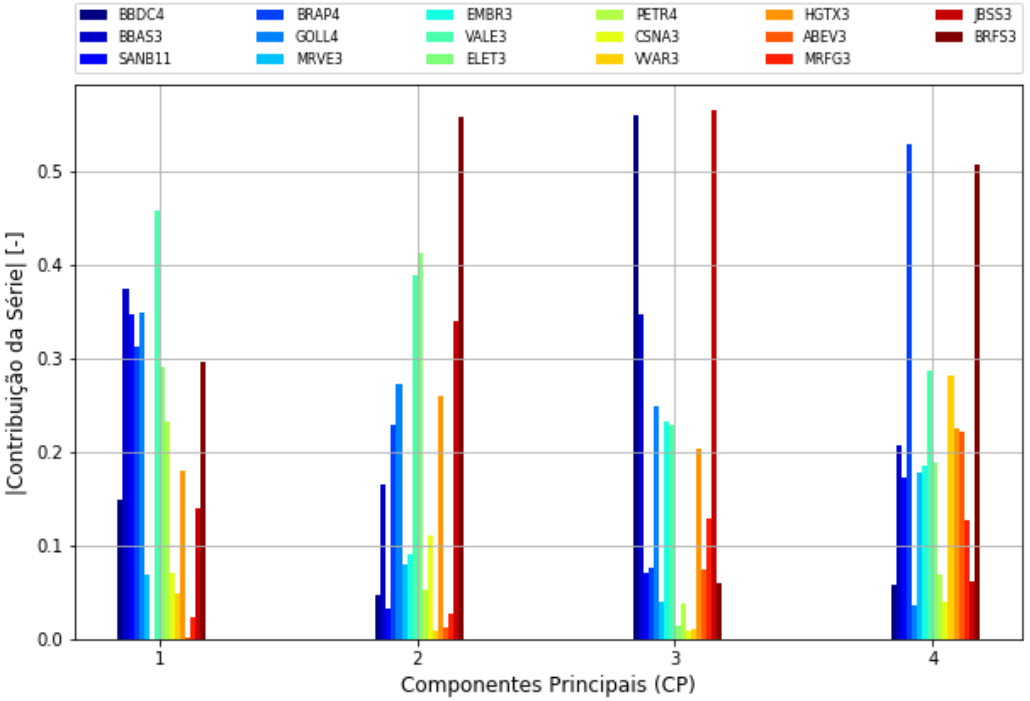


Figura 10 – Contribuição das séries temporais nas 4 primeiras CP

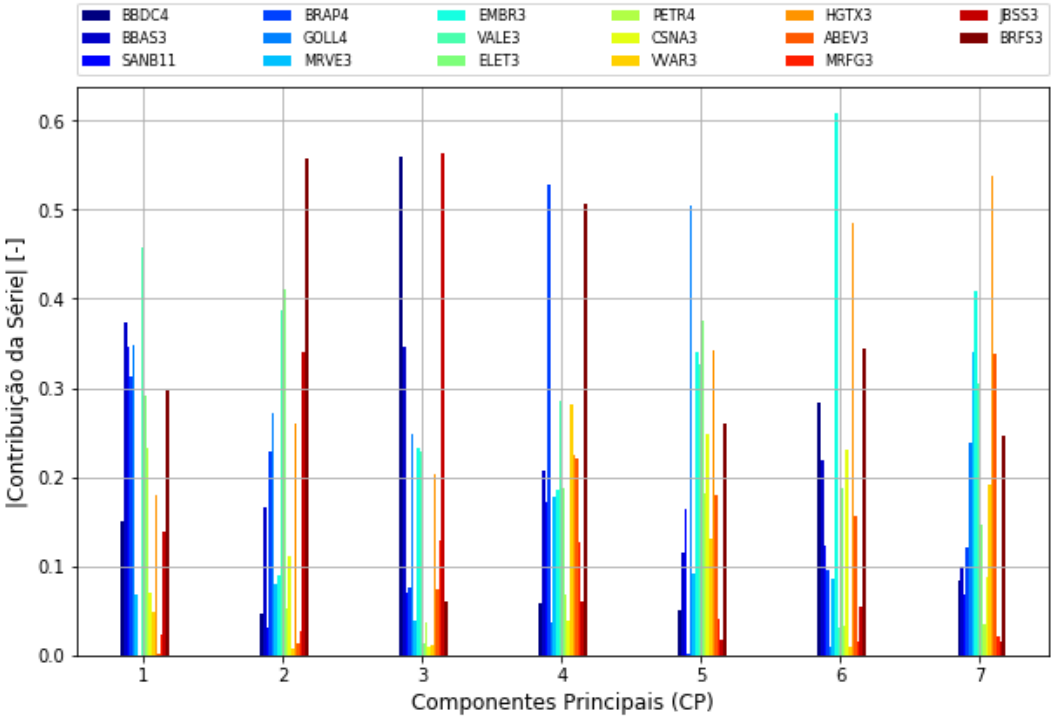


Figura 11 – Contribuição das séries temporais nas 8 primeiras CP

### 6.2.2 Análise de Sazonalidade

Buscando analisar uma possível periodicidade nos dados, foram adicionadas 24 novas séries temporais compostas de senos e cossenos, da seguinte forma:

Equação 12 – Séries Temporais Adicionadas

$$Cn(data) = N\cos(data)$$

$$Sn(data) = N\sin(data)$$

$$\text{Onde } N = 1,2,4,6,8,10 \text{ e } 12 \text{ e data} = {}^{dia\_do\_ano}/_{364}$$

As contribuições das séries temporais adicionadas para cada uma das componentes principais foram comparadas com a contribuição das séries temporais dos preços de fechamento dos ativos. A Figura 12 apresenta essa composição:

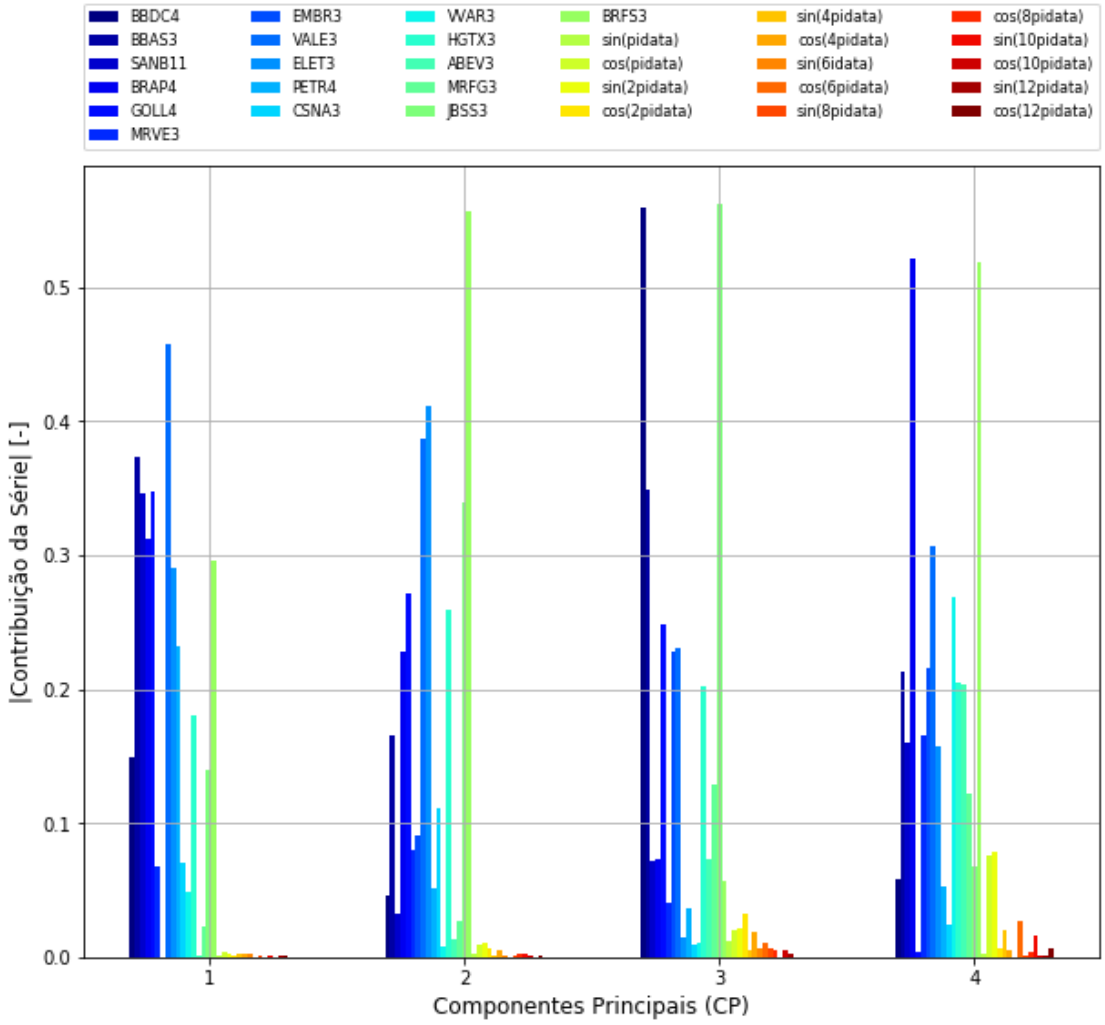


Figura 12 – Contribuição das séries nas 4 primeiras componentes principais

A contribuição das novas séries nas componentes principais foi relativamente pequena e demonstrou que qualquer contribuição sazonal é antecipada e precificada pelo mercado. Também foi possível confirmar a contribuição das séries de preços de fechamento de ativos analisadas na seção 6.2.1.

# Parte V

## Resultados

# 7 Resultados

Após a análise de componentes principais e seleção dos ativos para compor a aplicação, foram definidos uma janela rolada e os parâmetros C e ε para previsão das componentes principais.

## 7.1 Definições do Previsor

As 7 componentes principais devem ser previstas diariamente e compreende-se que cada uma delas pode possuir uma configuração de parâmetros do modelo de *Support Vector Regression* diferentes.

### 7.1.1 Janela Móvel

Considerando que as variáveis de entrada utilizadas no modelo não possuem comportamento estacionário, a abordagem por janela móvel apresenta-se como uma boa alternativa por considerar dados do passado recente para prever o comportamento futuro.

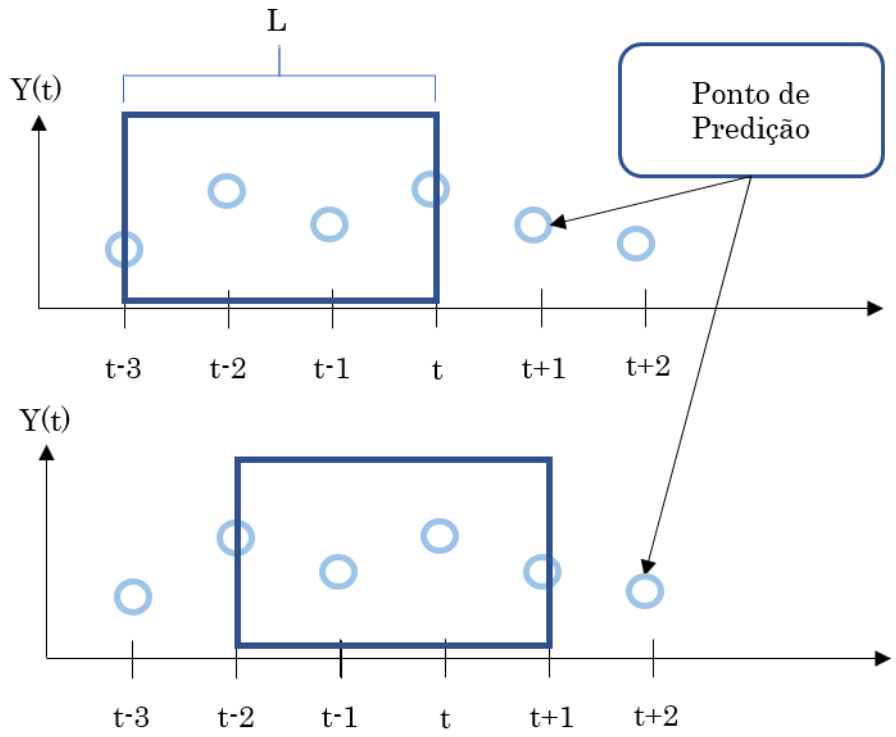


Figura 13 - Modelo de janela móvel

Para definir o tamanho da janela (L), foi analisado o erro quadrado médio (EQM) para cada uma das componentes principais, utilizando os valores das

componentes ao longo de um intervalo de teste de 58 dias (04/10/2019 a 30/12/2019). Os valores dos parâmetros do SVR foram mantidos constantes ($C = 1\ e\ \varepsilon = 0.1$) e o período de treinamento consistiu em 898 dias (04/01/2016 a 01/10/2019).

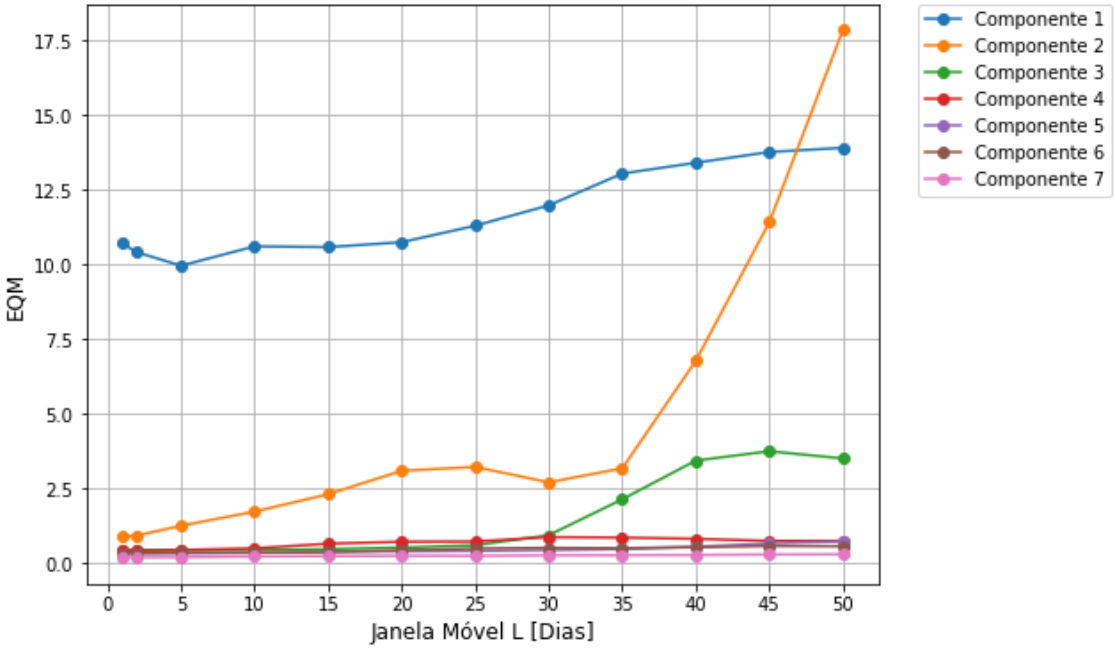


Figura 14 – EQM para cada uma das componentes ao longo do tempo

Ao analisar a Figura 14, verifica-se que as componentes possuem menor EQM para janelas móveis bem curtas (1 a 5 dias). O EQM das componentes 2 e 3 aumentam consideravelmente para janelas maiores que 30 dias.

O cálculo do EQM foi realizado como mostrado na equação 13, com $Y_P$ sendo os valores das componentes principais previstas e $Y_R$ como os valores reais das componentes, ambos ao longo do período de teste.

Equação 13 – Erro quadrado médio

$$EQM = \frac{1}{N}\sum_{i=1}^{N}(Y_P - Y_R)^2$$

A janela móvel de 5 dias apresentou os menores valores de EQM e por isso foi a janela escolhida para fase de calibração dos parâmetros C e ε.

### 7.1.2 Definição de C e ε

Definida a janela móvel, foram considerados um intervalo para cada um dos coeficientes C e ε para cada uma das componentes principais utilizadas.

Equação 14 – Intervalos de busca para C e ε

$$C = e^{x}, onde\ x = [1,2,3,4,5,6,7,8,9,10]$$

$$\varepsilon = e^{y}, onde\ y = [-5,-2,0]$$

Para o mesmo período de teste da Seção 7.1.1 , foi analisado o erro médio quadrado (Equação 13) de cada uma das combinações de constante de regularização C e tolerância ε.

Tabela 2 - Grid de combinações (C,ε)

| C | ε | i | C | ε | i | C | ε | i |
|---|---|---|---|---|---|---|---|---|
| $e^{1}$ | $e^{-5}$ | 0 | $e^{1}$ | $e^{-2}$ | 1 | $e^{1}$ | $e^{0}$ | 2 |
| $e^{2}$ | $e^{-5}$ | 3 | $e^{2}$ | $e^{-2}$ | 4 | $e^{2}$ | $e^{0}$ | 5 |
| $e^{3}$ | $e^{-5}$ | 6 | $e^{3}$ | $e^{-2}$ | 7 | $e^{3}$ | $e^{0}$ | 8 |
| $e^{4}$ | $e^{-5}$ | 9 | $e^{4}$ | $e^{-2}$ | 10 | $e^{4}$ | $e^{0}$ | 11 |
| $e^{5}$ | $e^{-5}$ | 12 | $e^{5}$ | $e^{-2}$ | 13 | $e^{5}$ | $e^{0}$ | 14 |
| $e^{6}$ | $e^{-5}$ | 15 | $e^{6}$ | $e^{-2}$ | 16 | $e^{6}$ | $e^{0}$ | 17 |
| $e^{7}$ | $e^{-5}$ | 18 | $e^{7}$ | $e^{-2}$ | 19 | $e^{7}$ | $e^{0}$ | 20 |
| $e^{8}$ | $e^{-5}$ | 21 | $e^{8}$ | $e^{-2}$ | 22 | $e^{8}$ | $e^{0}$ | 23 |
| $e^{9}$ | $e^{-5}$ | 24 | $e^{9}$ | $e^{-2}$ | 25 | $e^{9}$ | $e^{0}$ | 26 |
| $e^{10}$ | $e^{-5}$ | 27 | $e^{10}$ | $e^{-2}$ | 28 | $e^{10}$ | $e^{0}$ | 29 |

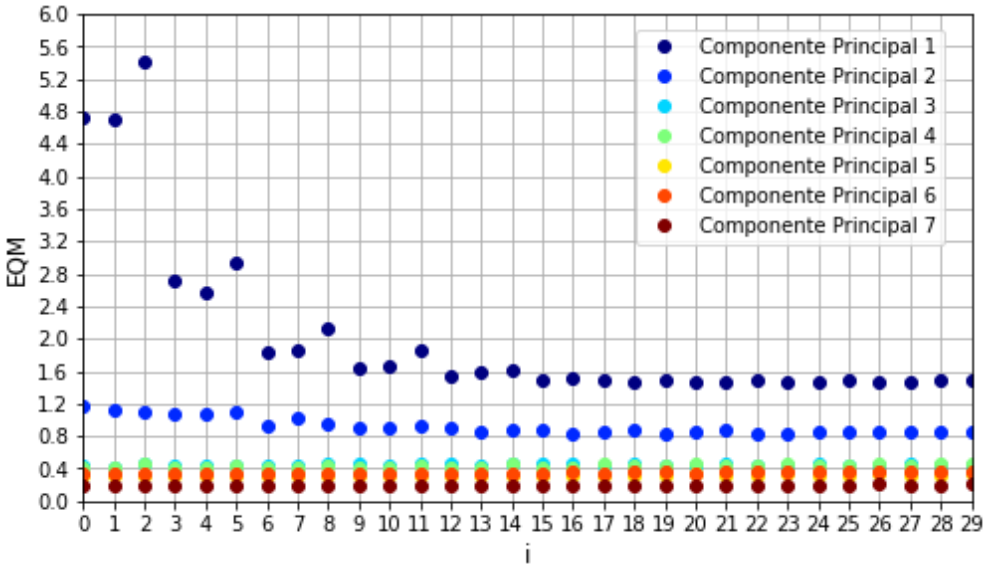


Figura 15 - EQM para as combinações de C e ε (dados de teste)

A partir da Figura 15, verifica-se que para todas as componentes principais os valores de $C \geq e^5$ apresentam os menores valores de EQM no período de teste, para todas as tolerâncias. Na figura 16, é possível ver com mais detalhes a região de valores de i que apresentam as melhores combinações para C e ε.

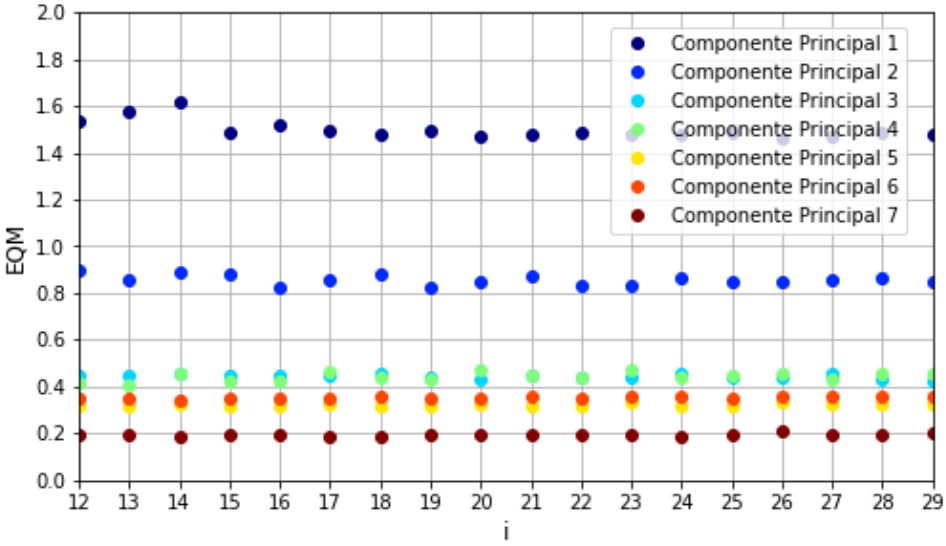


Figura 16 - EQM para as melhores combinações de C e ε (dados de teste)

Buscando evitar os efeitos de *overfitting* e *underfitting* e garantir o poder de previsibilidade do algoritmo, foram observados os erros quadrados médios também o período de treino. Para obter um ajuste satisfatório aos dados de treino e um poder de previsibilidade alto, foram selecionadas as combinações com os menores valores de EQM, considerando que os valores de treino deveriam possuir EQM inferior, porém o mais próximo possível dos valores de teste.

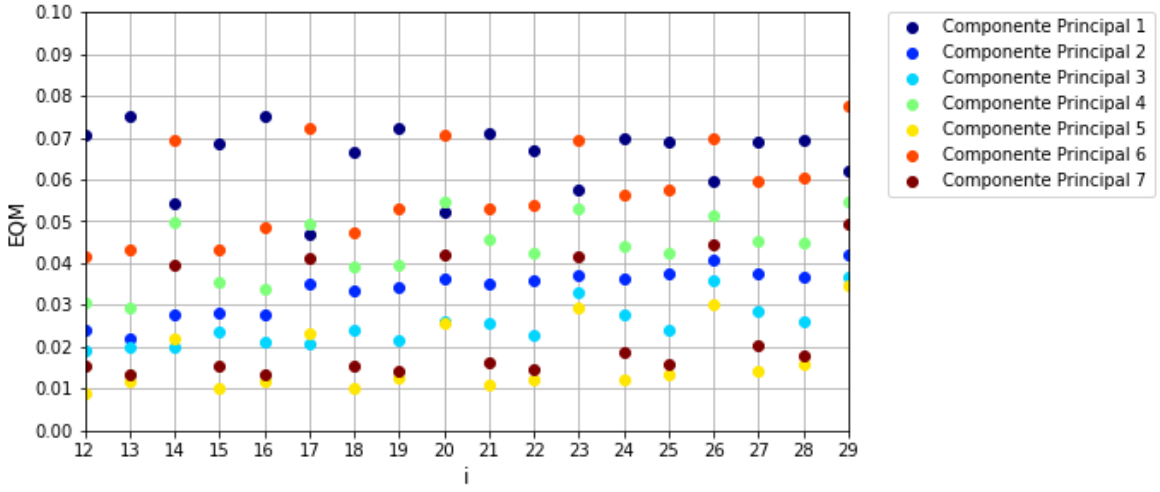


Figura 17 - EQM para as melhores combinações de C e ε (dados de treino)

A partir da Figura 17, verifica-se que para as melhores combinações, todos os EQM dos dados de treino são inferiores aos de teste o que demonstra que o problema de *underfitting* pode ser descartado. Considerando os valores reais das componentes principais (Figura 18), nota-se que tanto para o período de teste quanto para o período de treino o EQM é relativamente pequeno o que mostra que o algoritmo é promissor com relação ao poder de previsibilidade.

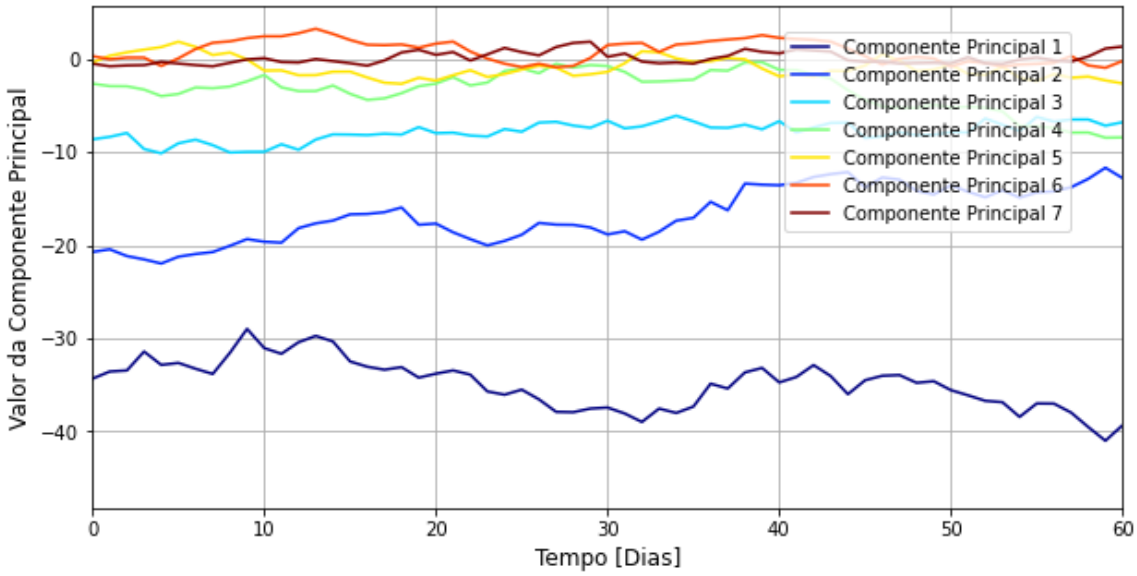


Figura 18 - Componentes Principais (período de teste)

Por fim, foram escolhidas as combinações de constante de regularização e tolerância e essas são apresentadas na tabela a seguir:

Tabela 3 - Combinações das constantes C e ε

| | CP 1 | CP 2 | CP 3 | CP 4 | CP 5 | CP 6 | CP 7 |
|---|---|---|---|---|---|---|---|
| **C** | $e^{7}$ | $e^{6}$ | $e^{7}$ | $e^{6}$ | $e^{5}$ | $e^{5}$ | $e^{6}$ |
| **ε** | $e^{0}$ | $e^{-2}$ | $e^{0}$ | $e^{-2}$ | $e^{-2}$ | $e^{0}$ | $e^{0}$ |

## 7.2 Estratégia

Como apresentado na Seção 5, foi definida a estratégia de compra e venda diária a partir da previsão do algoritmo. Utilizando o valor disponível em uma carteira inicial, divide-se o valor financeiro igualmente entre os ativos que devem ser comprados, mantendo assim a exposição da carteira para cada um dos ativos mais igualmente distribuída. Com esse método e os diversos ativos disponíveis, o algoritmo é capaz de reduzir risco não-sistêmico e definir, portanto, um portfólio mais diversificado.

Buscando fugir de pequenas flutuações e do intervalo de erro de predição, verificou-se que considerar como critério, um valor mínimo de percentual de aumento de preço do ativo para indicação compra, apresentou bons resultados. Foram verificados um intervalo de 0 a 5% e o valor de 1% apresentou os melhores resultados nos períodos analisados. Os resultados são apresentados nas Seções 7.3.2 e 7.3.3.

## 7.3 Resultados dos Testes

Tendo definidos os parâmetros do previsor para cada uma das componentes principais, o próximo passo foi implementar a estratégia.

Inicialmente, são apresentados os resultados utilizando o mesmo período de teste que foi utilizado na calibração, em seguida, são apresentados os resultados para 2020 e, por fim, é realizada uma análise global do desempenho da estratégia.

### 7.3.1 Definições

Para realizar a análise de desempenho foram estabelecidas algumas variáveis que permitiram analisar o desempenho da aplicação.

Para analisar o comportamento de cada ativo que compunha a carteira (e da carteira como um todo), foram criadas variáveis de cota para cada ativo e para carteira geral.

Equação 15 – Cálculo da cota de um ativo

$$Cota_{Ativo}(n) = \prod_{i=1}^{n}(1 + Previsao\ (P_{ativo,i+1} - P_{ativo,i})/P_{ativo,i}))$$

Na equação, "n" são os dias ao longo do período de teste, $P_{ativo,i+1}$ e $P_{ativo,i}$ são os preços reais dos ativos nos dias i+1 e i e "Previsao" é uma variável booleana que

vale 1 nos dias que a posição está comprada na carteira e 0 nos dias em que não há posição.

Para a cota da carteira, foi utilizado a diferença entre as posições nos dias i e i+1 e o saldo remanescente ($Saldo_i$). A posição é calculada utilizando a somatória da quantidade de cada ativo $Q_{ativo,i}$ multiplicada pelo preço real do ativo $P_{ativo,i}$.

Equação 16 – Posição de um ativo

$$Posição_i = \sum_{ativo=1}^{m} (Q_{ativo,i} P_{ativo,i})$$

Considerando que o número de ações que pode ser comprado de cada ativo é um número inteiro, o saldo remanescente no dia i ($Saldo_i$) consiste no valor total do financeiro que sobra após serem estabelecidas as posições no dia i.

Equação 17 – Cálculo da cota da carteira

$$Cota_{Carteira}(n) = \prod_{i=1}^{n} [1 + (Pos_{i+1} + Saldo_{i+1} - (Pos_i + Saldo_i))/(Pos_i + Saldo_i))]$$

Além das cotas, foi criada uma variável de rentabilidade acumulada para a carteira que permite analisar a evolução do patrimônio alocado ao longo do tempo. Tanto a rentabilidade acumulada quanto a cota da carteira, permitem verificar a qualidade da alocação das posições ao longo do tempo, mas a utilização da cota favorece a análise comparativa com relação ao *benchmark* utilizado.

Equação 18 – Rentabilidade acumulada da carteira

$$Rent_{Carteira}(n) = Saldo_0 \, Cota_{Carteira}(n)$$

O cálculo da rentabilidade acumulada da carteira depende da posição e saldo remanescente logo antes de iniciar o período de teste. Para os resultados a seguir, considera-se que o valor disponível para início da estratégia seja o equivalente a soma de 1 lote de cada ativo disponível (Tabela 1) na aplicação considerando o preço de fechamento do último dia de treino. Os ativos utilizados na aplicação são negociados em lotes padrões de 100 unidades, portanto as negociações são dadas em múltiplos de 100.

### 7.3.2 Resultados da carteira (out/2019 – dez/2019)

Como forma de analisar o desempenho de cada um dos ativos da carteira ao longo do período de teste, foi utilizada a Equação 15. No gráfico a seguir podemos ver o comportamento das cotas de cada ativo:

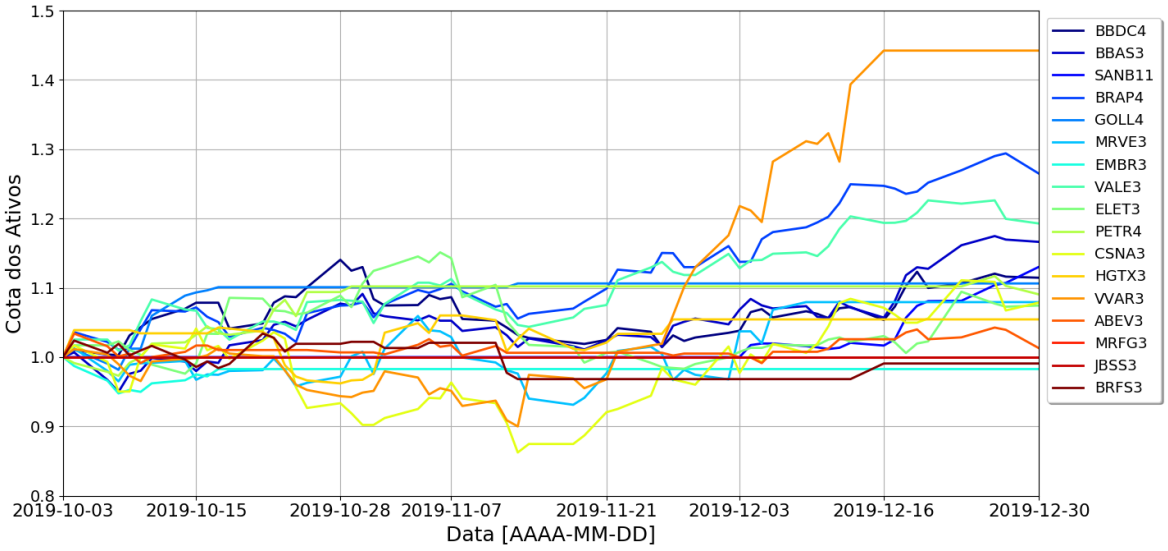


Figura 19 - Evolução das cotas dos ativos

Utilizando as cotas, verifica-se que as posições realizadas ao longo da aplicação da estratégia foram capazes de gerar alguma rentabilidade (Cota > 1) com quase todos os ativos.

Um dos requisitos definidos para esse projeto, apresentado na Seção 3, é o de que a estratégia deveria ser capaz de superar o *benchmark* do Ibovespa durante o período de utilização. Para examinar o desempenho comparativamente ao Índice, realizamos o cálculo da cota da carteira ao longo do período (Equação 17) e utilizamos a Equação 15 para definir a cota para o índice.

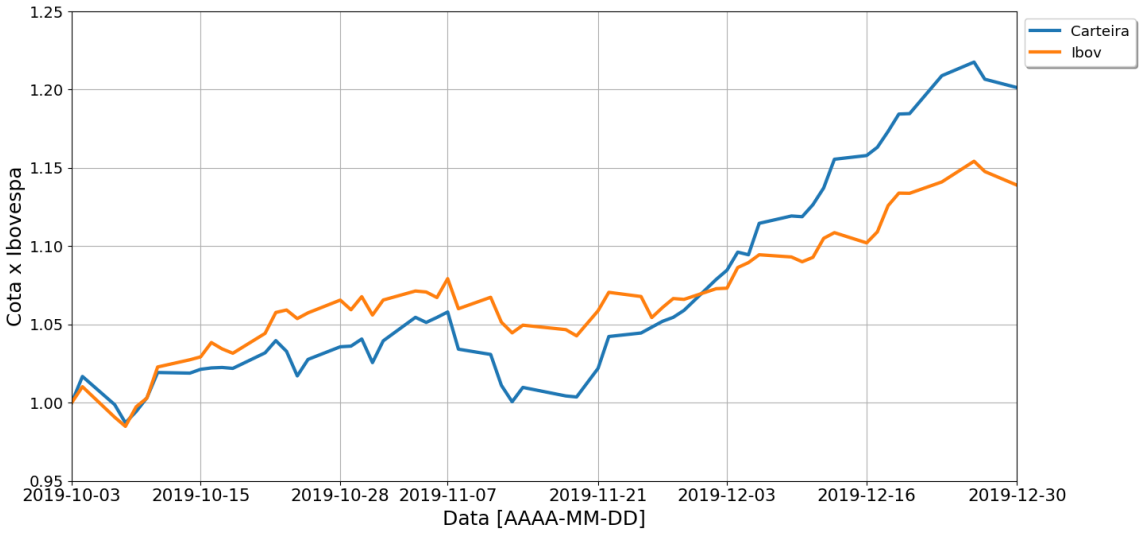


Figura 20 - Evolução da cota da certeira em relação ao Ibovespa

Durante a primeira metade do período analisado, a rentabilidade da carteira permaneceu abaixo do Bovespa, acompanhando as variações do índice. A partir do início de dezembro, no entanto, a rentabilidade da estratégia passa a crescer mais que o Ibovespa.

Considerando que os ativos considerados são negociados em lotes base de 100 ações. Foi considerado um patrimônio inicial de R$ 48.415,00 que seria equivalente a compra de 1 lote de cada ativo no primeiro dia de teste.


Figura 21 - Rentabilidade acumulada da carteira

Ao fim do período, a rentabilidade acumulada foi de R$ 9.747,00 que representa um ganho de 20,13% em relação ao patrimônio inicial em 3 meses de aplicação.

### 7.3.3 Resultados da carteira (2020/2019)

No ano de 2020, presenciamos um período de *stress* econômico global causado pela pandemia do Coronavírus. Buscando reduzir o contágio, diversos países determinaram políticas de restrição de mobilidade e de confinamento, fechando restaurantes, escolas, comércios e indústrias. Houve uma grande redução da atividade econômica global e aumento nos índices de desemprego, gerando uma recessão.

Entre os meses de fevereiro e março, verificaram-se quedas expressivas nos índices de bolsas globais, incluindo o Ibovespa. Dentro deste contexto, continuamos a testar o desempenho da aplicação desenvolvida, para determinar se ela era capaz de performar dentro do mercado em um cenário de crise econômica.

O gráfico da Figura 22 apresenta o comportamento das cotas de cada um dos ativos ao longo do período.

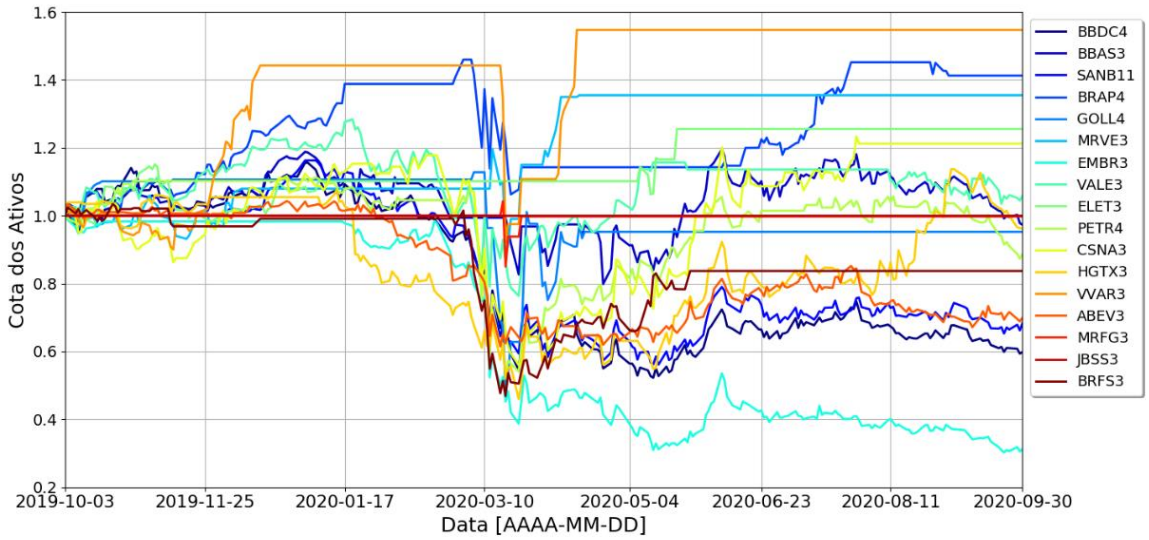


Figura 22 - Evolução da cota dos ativos 2019/2020

Destaca-se que os ativos de pior desempenho, como é o caso da EMBR3, compõem alguns dos setores mais afetados pela pandemia do Coronavírus, no caso, devido às restrições e suspenções de viagens aéreas.

O gráfico a seguir apresenta o desempenho da cota da carteira e do índice ao longo do período:

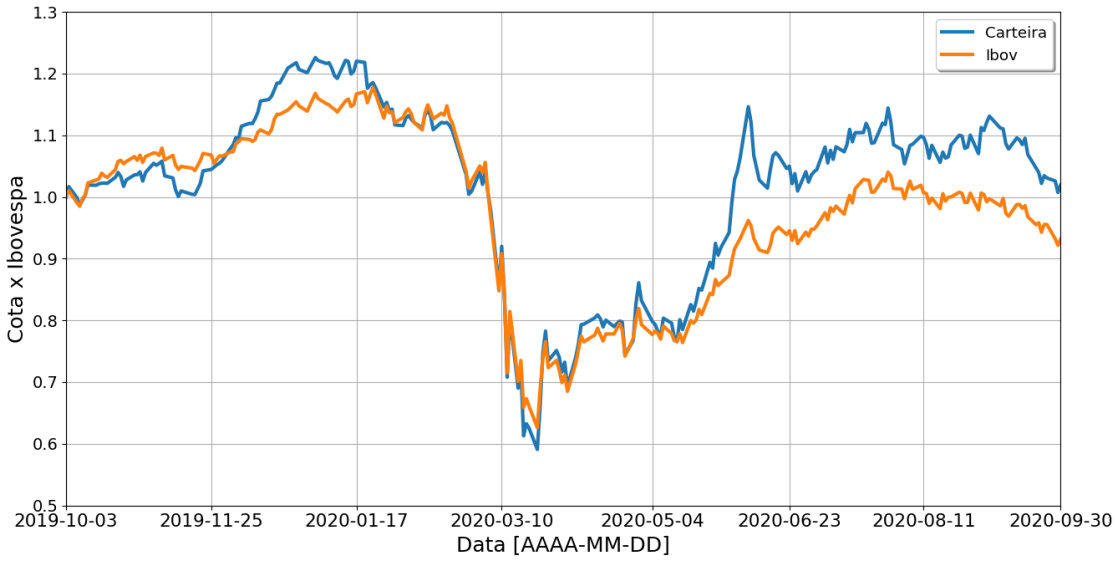


Figura 23 - Evolução da cota da carteira 2019/2020

Verifica-se um comportamento semelhante ao encontrado na seção 7.3.2, no qual o a carteira acompanha bem a variação do índice. Ao fim do período, o rendimento da carteira se encontra acima do *benchmark*.

Considerando o mesmo patrimônio inicial da seção 7.3.2., também se verificou a rentabilidade acumulada no período. A rentabilidade da carteira foi de 1.92%.


Figura 24 - Rentabilidade acumulada no período 2019/2020

### 7.3.4 Desempenho da Estratégia

O desempenho da estratégia desenvolvida pelo grupo foi avaliado por dois parâmetros: *Alpha* e *Sharpe Ratio*.

Criado por William Sharpe em 1966, o *Sharpe Ratio* se tornou uma das medidas mais usadas no mundo das finanças para se analisar uma relação de risco-retorno. Ele descreve o quanto a mais de retorno um investidor recebe pela volatilidade, ou risco, ao qual ele se submete ao manter sua posição em um ativo ou fundo.

A medida é determinada pela divisão da diferença entre o retorno médio de um determinado portfólio ($Rx$) pelo retorno do benchmark pelo desvio padrão (volatilidade) do portfólio:

Equação 19 - Cálculo do *Sharpe Ratio*

$$S(x) = \frac{Rx - Rf}{\sigma}$$

Neste caso, considera-se o investimento livre de risco o próprio benchmark estabelecido (índice Bovespa).

O *Sharpe Ratio* anualizado calculado para os resultados obtidos pela carteira no ano de 2019 foi de 1.4. Isso implica que há um prêmio de risco considerável: 1.4 pontos de rentabilidade acima do benchmark por unidade de risco tomada.

Quando passamos para o período de 2019/2020, a carteira passou a ter um rendimento menor, mas ainda sim positivo, enquanto o Ibovespa teve rendimento negativo. Calculando o *Sharpe* da estratégia no período, obteve-se o valor de 1.05 e considerando o cenário de *stress* mundial global, a estratégia manteve-se performando de forma satisfatória, oferecendo um retorno esperado acima do benchmark maior que 1 por unidade de risco tomada.

O *Alpha,* por sua vez, também mede a performance de um investimento em relação a um certo *benchmark*, porém de uma forma um pouco mais simples. O *Alpha* é simplesmente a diferença entre o retorno de um determinado ativo, ou grupo de ativos, e o retorno do benchmark, sendo tal variação chamada de retorno excedente. O *Alpha*, portanto, apresenta o quanto melhor ou pior determinado investimento performou em relação ao ativo escolhido como base para um período determinado.

No período de 2019 analisado, o rendimento do Bovespa foi de 13,89%, enquanto o da carteira gerada pelo grupo foi de 20,13%, o que gera um *Alpha* de 6.24% Na análise realizada durante 2019/2020, o retorno da carteira foi de 1.92%, enquanto o do índice foi de -6,84%. Dessa forma, temos um *Alpha* de 4.9%.

# Parte V

## Considerações Finais

# 8 Considerações Finais

Ao analisar os resultados obtidos, pode-se concluir que as ferramentas de *Machine Learning* utilizadas juntamente com o método de Componentes Principais se apresentam como uma ferramenta poderosa para a previsão de séries temporais financeiras. Possíveis problemas de *underfitting* foram descartados rapidamente e nos períodos de teste analisados também não foram encontrados sinais significativos de *overfitting*, mostrando que o modelo aprendeu as relações existentes no período de treino e apresentou poder preditivo.

No início desta monografia, foram determinados alguns requisitos para o projeto baseados nos objetivos definidos. O primeiro, considerado o mais importante, era o de que a aplicação desenvolvida, juntamente com a estratégia de investimento, deveria superar o *benchmark* do índice Bovespa, sem ter conhecimento prévio de sua composição. Como foi apresentado no tópico anterior, a carteira determinada pelo grupo performou melhor que o Ibovespa durante os dois períodos analisados.

Outro requisito também relacionado ao desempenho da carteira hipotética foi que o algoritmo deveria ser capaz de performar ainda em cenários de estresse de mercado. No período de 2020, verificou-se um cenário de estresse causado pela pandemia do Coronavírus, e foi possível verificar que a carteira ainda performou acima do benchmark nesse período. As métricas de *Sharpe* e *Alpha* comprovaram a consistência do modelo e a capacidade de gerar rentabilidade acima do benchmark.

O futuro do projeto consiste em continuar avaliando o desempenho do algoritmo ao longo do tempo e desenvolver uma interface para que um usuário leigo possa utilizar a aplicação, como por exemplo um aplicativo para dispositivo móvel (referente ao requisito RS2). O modelo atual já é capaz de realizar as indicações de compra e venda utilizando uma base universal, e com o desenvolvimento da plataforma para usuário, pode-se escolher quais setores do mercado deseja-se atuar (Tabela 1).

Uma outra possibilidade futura é implementar novas estratégias dentro da aplicação, com o objetivo de adaptar a aplicação para utilização de usuários que desejam investimentos menos arrojados. Até o momento, atua-se somente no mercado comprando e vendendo ações, sem considerar outros produtos para alocação, portanto um plano para o futuro é a incorporação investimentos em renda

fixa para que se possa reduzir a exposição à mercados de grande volatilidade em períodos de estresse.

De modo geral, o projeto foi capaz de atingir os objetivos propostos e se mostrou uma ferramenta poderosa para gestão de um portfólio composto por ativos do mercado brasileiro de ações.

# Referências


Abu-Mostafa, Y. S., & Atiya, A. F. (1996). Intriduction to Financial Forecasting. *Applied Inteligente*, 205-213.

Chowdhury, U. N., Hossain, M. T., & Chakravarty, S. K. (Janeiro de 2018). Short-Term Financial Time Series Forecasting Integrating Principal Component Analysis and Independent Component Analysis with Support Vector Regression. *Journal of Computer and Communications* , pp. 51-67.

Dias, M. S. (Fevereiro de 2007). O uso de máquina de suporte vetorial para regressão (SVR) na estimação da estrutura a termo da taxa de juros do Brasil. *Dissertação de Mestrado*.

Enke, D., & Thawornwong, S. (2005). The use of Data Mining and Neural Networks for Forecasting Stock Market Returns. *Expert Systems with Applications*, 927-940.

Gupta, J. N., & Sexton, R. S. (1999). Comparing backpropagation with a genetic algorithm for neural network training. *Omega*, 679-684.

Haykin, S. S. (1999). *Neural Networks: A Comprehensive Foundation.* University of Michigan: Prentice Hall.

Kaastra, I. (1994). Forecasting Futures Trading Volume Using Neural Networks. *Journal of Futures Markets*, 71.

Leahy, S., Carciente, S. L., Stanley, H. E., & Kenett, D. Y. (2014). Structure and Dynamics of the Brazilian Stock Market: A Correlation Analysis. *Avaliable at SSRN: https://ssrn.com/abstract=2484648* , 23.

Linnainmaa, S. (1976). Taylor Expansion of the Accumulated Rounding Error. *BIT Numerical Mathematics*, 146-160.

Müller, K. -R., Smola, A. J., Rätsch, G., Schölkopf, B., Kohlmorgen, J., & Vapnik, V. (1997). Predicting time series with support vector machines. *International Conference on Artificial Neural Networks*, 999-1004.

Pearson, K. (1901). LIII. On Lines and Planes of Closest Fit to Systems of Points in Space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 559-52.

Shen, S., Jiang, H., & Zhang, T. (2012). Stock Market Forecasting Using Machine Learning Algorithms. *Department of Electrical Engineering, Stanford University*.

Tay, F. E., & Cao, L. (2001). Application of Support Vector Machines in Financial Time Series Forecasting. *The International Journal of Management Science*, 309-317.

Vapnik, V. (2000). *The Nature of Statistical Learning Theory.* New York: Springer-Verlag.

Vapnik, V., & Cortes, C. (1995). Support-Vector Networks. *Machine Learnig, 20*, pp. 273-297.

W.-C, C., Urban, T., & Baildridge, G. (1996). A Neural Network Approach to Mutual Fund net Asset Value Forecasting. *Omega*, 205-215.

Zhang, G., Patuwo, B. E., & Y. Hu, M. (1998). Forecasting with Artificial Neural Networks: The State of the Art. *International Journal of Forecasting*, 35-62.

Zhanga, G., & Hu b, M. Y. (1998). Neural Network Forecasting of the British Pound/US Dollar Exchange Rate. *Omega*, 495-506.