CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
PROGRAMA DE PÓS-GRADUAÇÃO EM MODELAGEM MATEMÁTICA E COMPUTACIONAL

# PREVISÃO DE RESULTADOS NO FUTEBOL POR MEIO DE TÉCNICAS DE APRENDIZADO DE MÁQUINA

**FELIPE AUGUSTO PEREIRA FERNANDES**

Orientador: Flávio Vinícius Cruzeiro Martins
Centro Federal de Educação Tecnológica de Minas Gerais

Coorientador: Anísio Mendes Lacerda
Centro Federal de Educação Tecnológica de Minas Gerais

BELO HORIZONTE
FEVEREIRO DE 2019

**FELIPE AUGUSTO PEREIRA FERNANDES**

# PREVISÃO DE RESULTADOS NO FUTEBOL POR MEIO DE TÉCNICAS DE APRENDIZADO DE MÁQUINA

Dissertação apresentado ao Programa de Pós-graduação em Modelagem Matemática e Computacional do Centro Federal de Educação Tecnológica de Minas Gerais, como requisito parcial para a obtenção do título de Mestre em Modelagem Matemática e Computacional.

Área de concentração: Modelagem Matemática e Computacional

Linha de pesquisa: Sistemas Inteligentes

Orientador: Flávio Vinícius Cruzeiro Martins
Centro Federal de Educação Tecnológica de Minas Gerais

Coorientador: Anísio Mendes Lacerda
Centro Federal de Educação Tecnológica de Minas Gerais

CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
PROGRAMA DE PÓS-GRADUAÇÃO EM MODELAGEM MATEMÁTICA E COMPUTACIONAL
BELO HORIZONTE
FEVEREIRO DE 2019

Fernandes, Felipe Augusto Pereira
F363p Previsão de resultados no futebol por meio de técnicas de aprendizado de máquina / Felipe Augusto Pereira Fernandes. – 2019.
84 f.

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Modelagem Matemática e Computacional.
Orientador: Flávio Vinícius Cruzeiro Martins.
Coorientador: Anísio Mendes Lacerda.
Dissertação (mestrado) – Centro Federal de Educação Tecnológica de Minas Gerais.

1. Futebol – Aspectos econômicos – Teses. 2. Aprendizado por computador – Teses. 3. Esportes – Métodos estatísticos – Teses. 4. Apostas esportivas – Teses. I. Martins, Flávio Vinícius Cruzeiro. II. Lacerda, Anísio Mendes. III. Centro Federal de Educação Tecnológica de Minas Gerais. IV. Título.

CDD 519.3

Elaboração da ficha catalográfica pela Biblioteca-Campus II / CEFET-MG

SERVIÇO PÚBLICO FEDERAL
MINISTÉRIO DA EDUCAÇÃO
CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
COORDENAÇÃO DO PROGRAMA DE PÓS-GRADUAÇÃO EM MODELAGEM MATEMÁTICA E COMPUTACIONAL

# *"PREVISÃO DE RESULTADOS NO FUTEBOL POR MEIO DE TÉCNICAS DE APRENDIZADO DE MÁQUINA"*

Dissertação de Mestrado apresentada por **Felipe Augusto Pereira Fernandes**, em 19 de fevereiro de 2019, ao Programa de Pós-Graduação em Modelagem Matemática e Computacional do CEFET-MG, e aprovada pela banca examinadora constituída pelos professores:

Prof. Dr. Flávio Vinícius Cruzeiro Martins (Orientador)
Centro Federal de Educação Tecnológica de Minas Gerais

Prof. Dr. Anisio Mendes Lacerda
Centro Federal de Educação Tecnológica de Minas Gerais

Prof. Dr. Daniel Hasan Dalip
Centro Federal de Educação Tecnológica de Minas Gerais

Prof. Dr. Paulo Eduardo Maciel de Almeida
Centro Federal de Educação Tecnológica de Minas Gerais

Visto e permitida a impressão,

Prof. Dr. Thiago de Souza Rodrigues
Coordenador do Programa de Pós-Graduação em Modelagem Matemática e Computacional

Dedico também aos portadores de doenças auto-imunes, em especial aos portadores da Doença de Chron. Apesar de toda a dificuldade e sofrimento, ainda podemos sonhar e batalhar para alcançar nossos objetivos. A vida vale a pena ser vivida!

# Agradecimentos

Agradeço aos meus pais, Adilson José Fernandes e Maria da Penha Pereira Fernandes, e minha irmã, Bruna Regina Pereira Fernandes, por todo amor, carinho e compreensão que tiveram comigo durante toda a minha vida.

Também agradeço à Patrícia Sette Câmara Haizer, que me apoiou e teve muita paciência comigo nessa jornada.

Agradeço aos professores que tive no CEFETMG, tanto na graduação como no mestrado, pois me serviram de exemplo e inspiração acadêmica.

Agradeço aos meus amigos que me apoiaram e me incentivaram durante todos os momentos difíceis, alguns com palavras incentivadoras e outros com belas risadas.

Agradeço meu orientador, Flávio Vinícius Cruzeiro Martins, e meu co-orientador, Anísio Lacerda, pela compreensão e ajuda ao longo do estudo.

Agradeço também à CAPES, ao CNPq e à FAPEMIG pelo apoio financeiro.

*"Walk on, Walk on*
*With hope in your heart*
*And you'll never walk alone"*
(You'll Never Walk Alone, Compositores: Oscar Hammerstein li / Richard Rodgers)

# Resumo

Futebol é considerado o esporte mais popular do mundo. Por isso, existe um grande interesse em saber quem será o vencedor de uma partida com impactos sociais e econômicos ao redor do mundo. A partir de dados básicos de desempenho, extraídos dos placares dos jogos, este trabalho tem o objetivo de estudar o desempenho dos algoritmos de aprendizado de máquina aplicados à predição de jogos de futebol: vitória do mandante, empate ou vitória do visitante. Propusemos um modelo de dados para predição de jogos de futebol e investigamos os modelos de aprendizado com tarefas de regressão e classificação aplicando técnicas de redução de dimensão. Os algoritmos utilizados para regressão foram: Regressão Linear (RegLin), K vizinhos mais próximos (KNN), *Random Forest* (RF) e *Gradient Boosting* (GB). Para classificação: Regressão Logística (RegLog), *Random Forest* (RF), *Gradient Boosting* (GB), *Naive Bayes* (NB) e K vizinhos mais próximos. Para os algoritmos de regressão, o valor referência passou a ser o saldo de gols, pois é um número inteiro que varia num intervalo . O melhor algoritmo, em relação ao *F1-score*, foi o GB aplicado a tarefa de regressão utilizando o modelo proposto neste estudo com $K$ igual a $0.25$. Este modelo apresentou um *F1-score* de $0.509$ e uma acurácia de $52.78\%$. Aplicando esse modelo em apostas esportivas, ele se mostrou robusto em relação a lucratividade, com rentabilidade superior a índices de referência como IBOV e a taxa básica de juros do Brasil, a taxa SELIC.

**Palavras-chave**: Futebol, Predição, Aprendizado de Máquina, Predição de jogos de Futebol, Apostas.

# Abstract

Football is considered the most popular sport in the world. Therefore, there is a great interest in knowing who will be the winner of a match with social and economic impacts around the world. Based on basic performance data, extracted from game scores, this work has the objective of studying the performance of machine learning algorithms applied to soccer game prediction: victory of home team, draw or victory of the visitor. We propose a data model to predict soccer games and investigate machine learning models with regression and classification tasks applying size reduction techniques. The algorithms used for regression were: Linear Regression (RegLin), K-neighbors (KNN), Random Forest RF and Gradient Boosting (GB). For classification: Logistic Regression (RegLog), Random Forest (RF), Gradient Boosting (GB), Naive Bayes (NB), and K-nearest neighbors (KNN). For the regression algorithms, the reference value became the goal difference. To map the predicted goal difference to one of our classes, we create a function that receives the predicted goal difference and a parameter $K$ that must be set. If predicted goal difference is: greater than $K$ then the is home team victory; less than $-K$ then predicted class is away team victory; otherwise the class is draw. The best algorithm, considering F1-score, was the GB applied to the regression task using the model proposed in this study with $K$ equal to $0.25$. This model had an F1-score of $0.509$ and an accuracy of $52.78\%$. Applying the best model in sports betting it proved to be robust in terms of profitability, with better profitability higher than benchmark indices such as IBOV and the Brazilian interest rate, SELIC.

**Keywords**: Soccer, Football, Machine Learning, Football Prediction, Bet, Soccer Betting

# Lista de Figuras

# Lista de Tabelas

# Lista de Quadros

# Lista de Algoritmos

# Lista de Abreviaturas e Siglas

| | |
|---|---|
| CDI | Certificado de Depósito Interbancário |
| DGLM | Modelos Dinâmicos Lineares Generalizados - *Dynamic Generalized Linear Model* |
| GB | *Gradient Boosting* |
| IBOV | Índice da Bolsa de Valores de São Paulo |
| KNN | K Vizinhos mais próximos - *K Nearest Neighbours* |
| LDA | Análise Discriminante Linear - *Linear Discriminant Analysis* |
| LogReg | Regressão Logística |
| MMI | Melhor Modelo Individual |
| NB | *Naive Bayes* |
| PCA | Análise dos Componentes Principais - *Principal Component Analysis* |
| RF | *Random Forest* |
| RegLin | Regressão Linear |
| RNA | Rede Neural Artificial |
| SELIC | Sistema Especial de Liquidação e Custódia |

# Sumário

# 1 Introdução

Futebol é considerado o esporte mais popular do mundo, segundo a *Association et al. (2007)*, e no Brasil não é diferente. O antropólogo francês Christian Bromberger (2001) procurou entender melhor os motivos da popularidade do Futebol. Em seu trabalho foram levantadas quatro hipóteses consideradas por ele reducionista.

A primeira é de que o futebol é o ópio do povo, desviando a atenção da massa dos temas políticos e econômicos, servindo assim a uma alta classe capitalista. A segunda hipótese toma como base a produção da catarse de massas, unindo os participantes e rompendo a sensação do cotidiano e as hierarquias. A terceira hipótese é de que o futebol faz ressurgir comportamentos arcaicos expressos principalmente pelo conteúdo violento presente nas torcidas organizadas. A última é de que existe uma relação direta entre gostos esportivos e classes sociais. Utiliza-se o futebol como exemplo de esporte que concentraria características corporais e estéticas abominadas pelas classes mais altas, ao contrário do golfe que seria um esporte de estética mais aprazível.

Após refutar as hipóteses, Bromberger (2001) chega a conclusão de que são duas as principais fontes de popularidade do futebol: a teatralização da vida social e a identidade coletiva. De acordo com Bromberger (2001) o futebol cultua o mérito e a performance concomitante com a incerteza presente a cada jogo. O sentimento de pertencimento e mobilização social floresce nas pessoas com o auxilio da rivalidade regional e nacional. O estilo de jogo das equipes e os ideais presentes na filosofia existencial das equipes, mesmo que algumas equipes não cumpram esta filosofia, amplificam o sentimento de pertencimento. Esses fatores potencializam a popularidade do futebol, sendo assim justificativas para que seja o esporte mais popular do mundo.

Junto com toda essa popularidade vem o potencial econômico. É notória a exploração econômica do esporte, que movimenta muito dinheiro e gera diversos empregos indiretos. As empresas de comunicação como jornais e emissoras de televisão travam uma batalha diária pela audiência. Um exemplo disso é a aquisição da *Opta Sports*, maior empresa do mundo em coleta de dados estatísticos e vídeos de jogos de futebol, pela *Perform Group*, empresa líder em conteúdo digital para esportes, por quarenta milhões de libras esterlinas (OPTA, 2013).

É comum, durante as transmissões dos jogos e em programas esportivos, utilizar-se de números para demonstrar ou corroborar com questões frequentemente levantadas em debates futebolísticos. Qual time é melhor? Quem vai vencer o campeonato? Os especialistas presentes nestes programas usam dados básicos de desempenho para discutir e fazer projeções, empíricas, sobre os jogos da rodada ou sobre o futuro do campeonato

em questão.

Determinar o resultado de uma partida não é uma tarefa simples, pois o futebol é um esporte coletivo sujeito a inúmeras variáveis, como padrão de um time, esquema tático, falhas individuais de jogadores, capacidade de execução e habilidade dos envolvidos, condição do gramado, motivação e montagem da equipe, árbitros etc.

Devido a dificuldade desta tarefa, hoje as empresas estão focadas em estabelecer melhores métricas de desempenho, como melhorar o entendimento do passe no futebol (POWER et al., 2017), utilizando técnicas de Aprendizado de Máquina para melhorar o entendimento do desporto e consequentemente o auxílio à decisão de técnicos e jogadores como é dito por Dhar (2017). Corroborando com isso existem dois livros de ampla divulgação, *Soccernomics* (KUPER, 2014) e *The numbers game: why everything you know about football is wrong* (ANDERSON; SALLY, 2013), que ajudam a demonstrar o impacto que a análise estatística tem a contribuir em análises que envolvem todo o contexto do desporto.

O primeiro livro, *Soccernomics* (KUPER, 2014), trata sobre a mudança de perspectiva na análise do jogo de futebol. Ele introduz um olhar estatístico econômico sobre o jogo. Demonstra como o processo de contratação de treinadores e dirigentes é raso, normalmente desconsiderando a os dados básicos sobre o desempenho e baseando somente no sentimento dos presidentes dos clubes. Também apresenta uma perspectiva sobre a contratação de jogadores que se destacam em campeonatos de seleções internacionais, como Copa do Mundo e Eurocopa. Os autores demonstram casos em que jogadores foram contratados por preços acima do mercado e na maioria das vezes não retornaram nem perto do esperado. Por fim os autores discutem aspectos do jogo e de desempenho de seleções internacionais de acordo com o tamanho da população, faixas etárias e outros dados estatísticos populacionais e sociais dos países. Já o segundo livro,(ANDERSON; SALLY, 2013), aborda assuntos mais específicos como porque o Chelsea deveria ter contratado Darren Bent em vez de Fernando Torres, ou discute a efetividade do lançamento lateral direto para área, uma prática comum na Inglaterra nos anos 90 e que ainda é utilizada por alguns treinadores ao redor do mundo. O objetivo é mudar a mentalidade da população em relação aos números, mesmo que através de uma análise quantitativa, ajudando a remover a barreira existente em relação a matemática, auxiliando na consolidação de uma nova era de análise de dados no futebol.

Apesar de todo o esforço em mudar a visão da sociedade e dos dirigentes esportivos, os dados utilizados em massa na mídia são dados básicos de informação sobre os jogos, obtidos na sua maioria, a partir da tabela de classificação do campeonato e dos jogos rodada a rodada.

Algoritmos de aprendizado de máquina são capazes de executar tarefas importantes através de generalização por exemplos (DOMINGOS, 2012). Quanto mais dado de exemplos são disponíveis melhor se consegue generalizar, seguindo premissas de validação para

evitar o sobre-ajuste. Com a introdução dessa nova visão sobre o esporte e a o acesso à dados básicos sobre às partidas de futebol, a curiosidade de se utilizar algoritmos de Aprendizado de Máquina aplicados à tarefa de predição de jogos de futebol se tornou um possível objeto de estudo.

Na literatura existem alguns trabalhos com essa proposta, entretanto utilizam algoritmos diferentes com modelo de dados distintos e com uma base de dados relativamente pequena. Estes fatores acabam sendo uma dificuldade na continuidade do estudo da área, pois primeiro não se sabe se Aprendizado de Máquina é um bom caminho para este tipo de tarefa e a comparação entre os trabalhos se torna muito custosa, tornando-se mais um empecilho para a introdução de novos pesquisadores nesta área.

Portanto, neste trabalho, objetiva-se o uso das técnicas de Aprendizado de Máquina aplicados na classificação de jogos de futebol, propondo um modelo de entrada que usufrui de dados básicos e de fácil acesso.

## 1.1 Objetivo

A partir de dados básicos de desempenho, podendo ser extraídos das rodadas de um campeonato, investigar o desempenho de diversas técnicas de Aprendizado de Máquina utilizando uma base de dados significativa, avaliando a aplicabilidade dos mesmos.

### 1.1.1 Objetivos Específicos

1. Proposição de um modelo de variáveis para a tarefa de predição de jogos de futebol;
2. Estudo do desempenho dos algoritmos e técnicas de Aprendizado de Máquina, mais comuns e populares na área, aplicados à tarefa de predição de jogos de futebol;
3. Estudo das formas de predição aplicadas aos diferentes modelos de dados com ou sem redução de dimensão.

## 1.2 Organização do trabalho

Este trabalho está organizado de forma que o Capítulo 2 apresenta alguns estudos, que tratavam da predição de jogos de futebol usando algoritmos de Aprendizado de Máquina, previamente presentes na literatura. Também apresenta alguns trabalhos relacionados que são importantes para o entendimento de alguns temas discutidos neste estudo. No Capítulo 3 são explanados fundamentos teóricos que são utilizados e necessários para o entendimento do trabalho. No Capítulo 4 apresenta-se como a base de dados foi modelada e representada, explanando como foi o processo de construção do Modelo Proposto para predição de jogos de futebol. Já no Capítulo 5 são apresentados os experimentos realizados junto com uma avaliação dos resultados e no Capítulo 6 apresenta-se uma análise com

uma discussão dos resultados obtidos. Nesse capitulo apresenta-se um estudo sobre a rentabilidade do melhor modelo individual obtido aplicado a apostas esportivas. Por fim, no Capítulo 7, conclusões são feitas acerca do trabalho realizado e de suas aplicações, juntamento com propostas de trabalhos futuros.

# 2 Trabalhos Relacionados

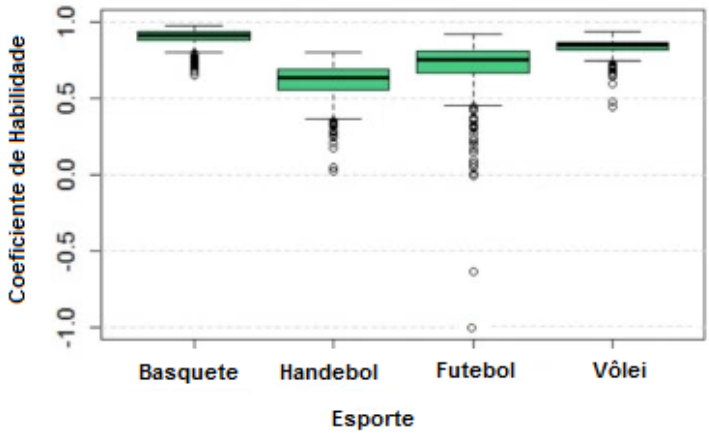
A predição de resultados esportivos é uma tarefa com dificuldade intrínseca da natureza do problema, entretanto é algo que facilitaria a tomada de decisão dos times e até o uso para apostas esportivas, sendo utilizada tanto por apostadores quanto por casa de apostas.

Aoki, Assuncao e Melo (2017) abordaram a árdua tarefa de predição de resultados esportivos, tentando quantificar o quão imprevisível é um esporte. Para isso, propuseram captar quanto do resultado é proveniente da sorte e quanto é proveniente da habilidade. Para tal, percorreram 1503 temporadas de quatro esportes entre 2007 e 2016. Todos os esportes analisados estavam sobre um sistema de pontos corridos de 2 turnos, onde uma equipe enfrenta todas as outras duas vezes: uma em casa e outra fora. No total foram: 42 ligas e 310 temporadas de basquete; 51 ligas e 328 temporadas de vôlei; 25 ligas e 234 temporadas de handebol; e 80 ligas e 631 temporadas de futebol.

Para mensurar o quão suscetível à sorte é um esporte, Aoki, Assuncao e Melo (2017) propuseram um coeficiente de habilidade, $\phi$, que objetiva indicar o quão longe os resultados de uma temporada estão do seu resultado esperado se a competição fosse simplesmente baseada em sorte. Quanto mais próximo de 1 menos baseado na sorte o esporte é.

Após a construção do modelo para cálculo de $\phi$, Aoki, Assuncao e Melo (2017) demonstram que futebol e handebol são esportes mais suscetíveis a aleatoriedade. A Figura 1 demonstra o diagrama de caixa das 1503 temporadas separadas por esporte. Por fim, os autores concluem que mesmo nos campeonatos mais competitivos a sorte está presente substancialmente, o que pode explicar parcialmente o porque modelos sofisticados e complexos tem desempenho relativamente pouco melhor que modelos simples para predição de resultados em esportes.

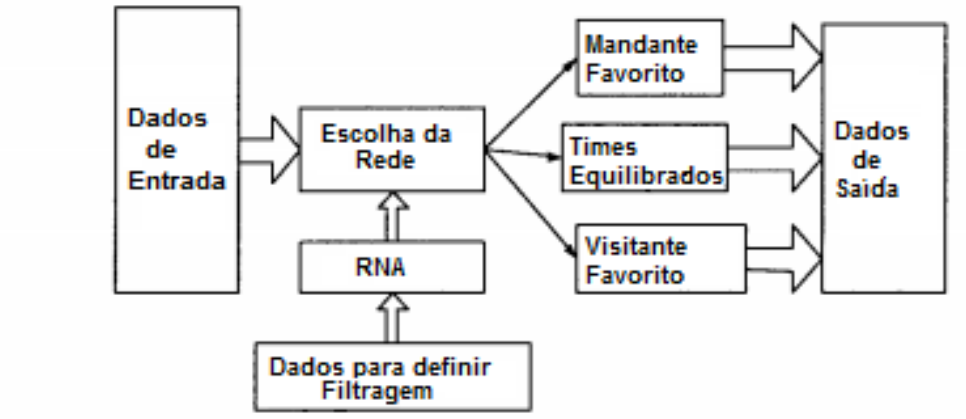
Figura 1 – Diagrama de caixa dos 1503 valores de $\phi$ em cada temporada separados por esporte.




Fonte: Traduzido e adaptdado de Aoki, Assuncao e Melo (2017)

Cheng et al. (2003) implementaram um modelo usando redes neurais artificias para a predição de jogos de futebol do campeonato italiano da primeira divisão das temporada 2001-2002, com 34 rodadas disponíveis. Para isso os autores introduziram um conceito de sistema hibrido, onde é feito um sistema que avalia o nível das equipes de uma dada partida e então direciona esta partida para uma das 3 redes: mandante mais forte, visitante mais forte e jogos equilibrados. A Figura 2 exemplifica o funcionamento do sistema híbrido. Dado um jogo a ser predito, primeiro é decidido a qual rede neural artificial (RNA) este jogo deve ser encaminhado. A predição então é dada como a saída da RNA escolhida para a partida em questão. Os autores testaram a saída desse modelo contra outros modelos baseados somente em estatísticas, como o *Elo Rating* (ELO, 1978), apresentando melhores resultados.

Figura 2 – Arquitetura do modelo proposto por Cheng et al. (2003)




Fonte: Traduzido e adaptdado de (CHENG et al., 2003)

Kumar (2013) utiliza técnicas de aprendizado de máquina para predição de jogos de futebol, identificando os atributos mais importantes das performances dos jogadores. Portanto a sua predição é baseada nesse histórico de performance dos jogadores.

Ulmer, Fernandez e Peterson (2013) propõem o uso de cinco diferentes classificadores para a predição de jogos da *Premier League*: Classificador linear com gradiente descendente, *Naive Bayes*, SVM (*Support Vector Machine*), *Hidden Markov Model* e *Random Forest*. Os melhores algoritmos, em relação a acurácia foram: Random Forest(0.50) e SVM(0.50). Os autores utilizaram 10 temporadas para treinamento (2002/2003 até 2011/2012) e duas temporadas para teste (2012/2013 e 2013/2014). Os dados de entradas eram básicos como: o time mandante e o visitante, gols marcados por cada equipe até aquela rodada.

Existem também alguns trabalhos propostos no campo da estatística Bayesiana. O trabalho feito por Diniz et al. (2017) propõe dois modelos *multinomial-Dirichlet* para predição de jogos de futebol do campeonato Brasileiro. Os autores concluem que o uso dos modelos baseados em estatística Bayesiana não são somente competitivos com os modelos baseados na estatística clássica como também são bem calibrados e tem um ajuste relativamente bom.

Já Owen (2011) apresenta um modelo dinâmico generalizado de modelos linear, do inglês (DGLMs - *Dynamic Generalized Linear Models*), que possibilita mensurar a habilidade dos times variando ao longo do tempo, já que nos modelos lineares generalizados os parâmetros do modelo, que na maioria se referem a habilidade das equipes, se mantém constante ao longo do tempo. Para isso ele utiliza do conceito de DGLM para variar os parâmetros ao longo do campeonato, assim otimizando a performance preditiva destes tipos de modelos. Nesse trabalho foram utilizados jogos do campeonato escocês das temporadas de 2003/2004 até 2005/2006. O autor conclui que o seu modelo apresenta resultados melhores que os modelos não dinâmicos.

Como foi apresentado brevemente, existem muitos trabalhos na literatura que abordam a predição de jogos de futebol. Entretanto existem alguns pontos interessantes a serem destacados: não há semelhança na base de dados entre os trabalhos; divergência na forma que os dados são utilizados, uns usando dados históricos dos times, outros dos jogadores; ainda há uma discrepância na finalidade dos modelos, já que uns querem fazer classificação e outros regressão; e por fim, não há continuidade em técnicas de Aprendizado de Máquina com o mesmo fim. Este trabalho pretende colaborar para esta área de estudo, propondo um modelo fixo de entradas e estudando a aplicação deste modelo de dados em algoritmos de aprendizado de máquina.

# 3 Fundamentação Teórica

Neste capitulo são abordados tópicos necessários para o entendimento deste trabalho. Discute-se os fundamentos do Aprendizado de Máquina, suas técnicas e especificidades.

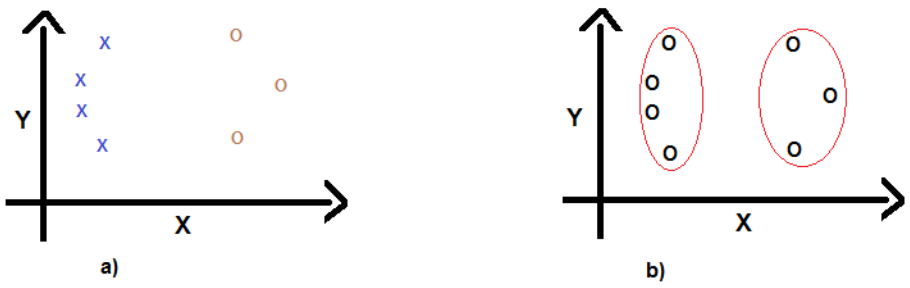
## 3.1 Aprendizado de Máquina

O termo Aprendizado de Máquina tem sido bastante discutido atualmente, entretanto, sua origem data de 1959. Arthur Samuel definiu Aprendizado de Máquina como o campo de estudo que dá aos computadores a capacidade de aprender sem terem sido explicitamente programados. (WIEDERHOLD; MCCARTHY; FEIGENBAUM, 1990)

No domínio de Aprendizado de Máquina é preciso investigar um conjunto de dados, pois é a partir deste conjunto que os padrões serão aprendidos. Dentro do contexto de aprendizado é necessário o reconhecimento de padrões para que o algoritmo se adeque aos impulsos desconhecidos e seja capaz de realizar sua tarefa de forma mais eficiente na próxima ocasião.

Existem duas formas simples de aprendizado: supervisionado e não supervisionado. No supervisionado existe a referência do resultado esperado que o algoritmo produza. Já no não supervisionado não existe essa referência direta entre o impulso desconhecido e seu resultado esperado. A Figura 3 demonstra um conjunto de dados para aprendizado supervisionado e não supervisionado. No exemplo a) pode-se observar que já existe uma definição de qual classe uma observação do conjunto de dados pertence, classes “xis” ou “bola”. Já no exemplo b) não existe uma definição explícita sobre a qual classe as observações pertencem. Então, o algoritmo de aprendizado deverá ser capaz de captar a similaridade das observações, definindo conjuntos de classes.

Figura 3 – Exemplificação de dados para problemas: a) Supervisionado e b) Não supervisionado




No processo de aprendizado supervisionado existem três etapas: treinamento, teste

e validação.

Na etapa de treinamento, apresenta-se um conjunto de dados, composto por variáveis de entrada, ao algoritmo de Aprendizado de Máquina. Este conjunto de treinamento é utilizado para identificação de padrões. Espera-se que ao fim do processo, o algoritmo tenha aprendido as relações da entrada com a saída desejada, sendo assim capaz de realizar a tarefa designada.

O processo de validação é um processo importante pois é nele que verificamos se o algoritmo de fato está aprendendo e não se sobre-ajustando, do inglês *overfitting*. Sobre-ajuste é quando o algoritmo se adequa muito bem ao conjunto de dados do treinamento, aprendendo muito sobre aqueles dados, inclusive possíveis ruídos comportamentais, e acaba perdendo sua capacidade de generalização, sendo ineficaz na tarefa designada. Outra funcionalidade da validação é a sintonia de parâmetros do algoritmo, para que o mesmo consiga atingir seu ponto ótimo.

A etapa de teste é onde verificamos a capacidade de generalização do algoritmo treinado e se o mesmo foi capaz de aprender de forma suficientemente satisfatória. Para isso, após a etapa de treinamento e validação, introduz-se um novo conjunto de dados, cujo o algoritmo não tem conhecimento algum, e verifica-se, através de uma métrica escolhida, se a tarefa está sendo executada da forma esperada e quão bom foi aprendizado.

Existem duas tarefas possíveis no aprendizado supervisionado. São elas: classificação e regressão. Ao se analisar uma variável ela pode ser qualitativa ou quantitativa. Uma variável quantitativa representa um valor numérico, uma quantidade. Como exemplo pode-se citar: o valor de um carro, peso de um produto, quantidade de gols de um time em uma partida etc. Já uma variável qualitativa indica uma qualidade, um tipo, uma classe. Sexo, cor, raça de um cachorro e variáveis binárias são alguns exemplos.

Quando a tarefa do algoritmo de aprendizado de máquina tem uma variável de resposta do tipo quantitativa, trata-se de um problema de regressão. Caso seja uma variável qualitativa, a tarefa é um problema de classificação. Portanto, em um problema de regressão a variável de resposta está num domínio continuo, onde o objetivo do algoritmo é, dado um conjunto de valores de entrada, predizer o valor da variável de resposta. Já para um problema de classificação a variável de resposta está no domínio discreto. Nesse caso um algoritmo deve, após um novo conjunto de entrada, estimar a qual classe aquele conjunto de dados de entrada pertence.

Nas próximas seções os algoritmos serão tratados com mais detalhes em suas especificidades.

## 3.2 Regressão Logística

Regressão é uma técnica que permite a verificação se duas ou mais variáveis estão relacionadas. Para entender essa relação um modelo matemático é necessário. De acordo com Montgomery, Runger e Calado (2000) o conjunto de técnicas estatísticas utilizadas para modelar e explorar relações determinísticas entre as variáveis é chamada de análise de regressão.

Para realizar uma regressão precisa-se definir as variáveis de entrada e saída. Após a coleta de dados, que é primordial para entendermos a relação entre as variáveis independentes e dependentes, aplica-se os dados no modelo matemático de aprendizado. Num processo iterativo, define-se os parâmetros desse modelo.

O modelo de Regressão Logística é utilizado para tarefas de classificação. Entretanto ele não é um classificador direto, é um modelo que produz uma probabilidade de pertencimento a uma classe particular (JAMES et al., 2013). A função que modela a regressão logística está apresentada na Equação (1)

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}} \tag{1}$$

onde $p(X)$ é saída do modelo de regressão, $\beta_0$ e $\beta_1$ são parâmetros do modelo, $X$ é a variável de entrada e $e$ é o número de Euler.

Para estimar os valores dos parâmetros do modelo é geralmente utilizado o método de estimativa por máxima verossimilhança, devido as suas melhores propriedades estatísticas (JAMES et al., 2013).

O método consiste em achar estimadores $\hat{\beta}_0$ e $\hat{\beta}_1$ de maneira que quando inseridos na equação 1 tenha-se um resultado próximo ao esperado (JAMES et al., 2013). Em um exemplo de classificador onde a saída é 1 para o sexo masculino e 0 para o feminino, os estimadores anteriormente citados, quando inseridos na Equação (1) devem fazer com que $p(X)$ tenha um valor próximo de 1 quando $X$ for um homem e um valor próximo de 0 quando $X$ for do sexo feminino. A equação que descreve este comportamento é:

$$l(\beta_0,\beta_1) = \prod_{i:y_i=1} p(x_i) \prod_{i':y_{i'}=0} (1 - p(x_{i'})). \tag{2}$$

Quando temos mais de uma variável dependente, o modelo de regressão é múltiplo. Isto significa que teremos um estimador $\beta$ para cada variável dependente.

Para problemas onde temos mais que duas classes para classificar, temos uma situação onde é criado um regressor para cada classe, onde sua tarefa é gerar uma probabilidade de pertencimento a uma classe específica. A sua equação é dada de forma

análoga a apresentada anteriormente pela seguinte equação:

$$Pr(Y = k|X) = \frac{e^{\beta_{0k}+\beta_{1k}X_1+...+\beta_{pk}X_p}}{\sum_{l=1}^{K} e^{\beta_{0l}+\beta_{1l}X_1+...+\beta_{pl}X_p}} \tag{3}$$

Na Equação (3) temos uma probabilidade para cada classe de saída $Y = k$ dado um conjunto de entrada $X$, onde $K$ é o número de classes de saída, $k$ é o indicador de uma classe especifica e $p$ é o tamanho do conjunto de entrada, onde $X_p$ é a variável de entrada com o índice $p$. Esta equação também pode ser denominada como Regressão Multinomial.

## 3.3 *Naive Bayes Classifier*

O classificador ingênuo de Bayes, ou em inglês *Naive Bayes Classifier*, é, assim como a Regressão Logística, um classificador probabilístico. Um dos motivos de seu uso comum é sua rapidez computacional, fácil implementação e sua eficácia relativa (RENNIE et al., 2003). O termo ingênuo se deve ao fato da forte suposição de independência entre as variáveis de entrada do modelo probabilístico (MURPHY, 2006).

Esse classificador usa como base o teorema de Bayes que é dado pela a Equação (4):

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}. \tag{4}$$

O classificador ingênuo de Bayes parte dessa equação e tem as seguintes premissas, de acordo com Duda, Hart e Stork (2012):

- O número de classes é conhecido, $k$;
- A probabilidade a priori $P(K_j)$ é conhecida, $j = 1,...,k$;
- O conjunto amostral de entrada, $X$, é desconhecido e não aleatório.

Quando adequamos a Equação (4) temos:

$$P(K_j|X) = \frac{P(X|K_j)P(K)}{P(X)}. \tag{5}$$

Dado $X$ o conjunto de entradas, temos $X = x_1,...,x_n$ onde $n$ é a quantidade de variáveis de entrada. Se aplicarmos o principio da independência entre as variáveis de entrada temos:

$$P(x_i|K_j,x_1,...,x_{i-1},x_{i+1},...,x_n) = P(x_i|K_j) \tag{6}$$

Aplicando a Equação (6) na Equação (5) temos:

$$P(K_j|x_1,...,x_n) = \frac{P(K_j)\prod_{i=1}^{n} P(x_i|K_j)}{P(x_1,...,x_n)} \tag{7}$$

Como $P(x_1,...,x_n)$ é sempre constante, independente da entrada podemos reduzir a Equação (7) para

$$P(K_j|x_1,...,x_n) \propto P(K_j) \prod_{i=1}^{n} P(x_i|K_j) \tag{8}$$

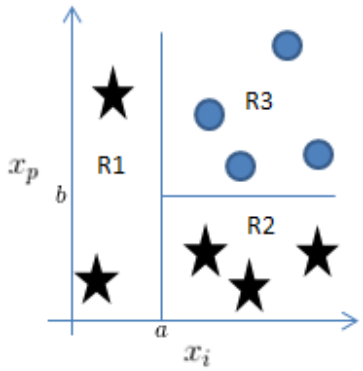
em que $K_j$ é uma classe do problema. Repetindo a Equação (8) para cada classe do problema, temos ao final, ao pegarmos o maior valor, a classe em que o conjunto de entrada $x_1,...,x_n$ pertence.

## 3.4 Árvore de Decisão

Os modelos baseados em árvore de decisão são modelos que estratificam ou segmentam o espaço das variáveis de entrada num número de regiões simples. Então para fazer uma predição de uma dada observação, um estimador pontual é usado na região na qual aquela observação pertence. Como essa divisão espacial é feita através de regras, pode-se resumir estas regras numa árvore, por isso esses métodos são conhecidos como árvores de decisão (JAMES et al., 2013). Elas podem ser usadas tanto para problemas de classificação quanto regressão.

A Figura 4 representa o espaço de variáveis particionado por uma árvore de decisão. Neste exemplo, $x_i$ e $x_p$ são as variáveis de entrada. Ao aplicarmos o método, foram criadas 3 regiões: R1, R2 e R3.

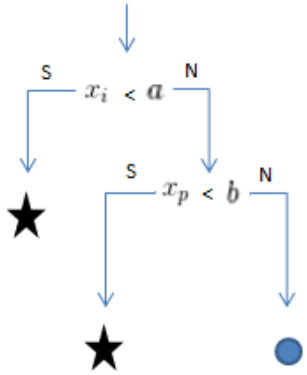
Figura 4 – Representação de um espaço particionado por uma árvore de decisão.




A Figura 5 é um fluxograma das regras de classificação da árvore de decisão deste exemplo. Dado uma observação, se $x_i < a$ o modelo pode retornar que aquela observação pertence a classe estrela. Caso contrário, verifica-se $x_p < b$. Se verdadeiro, o modelo classifica a observação como pertencente a classe estrela. Se falso, a classificação deve ser classe “bola”. Para se fazer a classificação no nó folha, pode-se usar estimadores como média e moda, entre outros.

Como pode-se observar, as árvores são simples e de fácil interpretação. Entretanto, considerando as métricas de eficiência mais comuns, geralmente uma árvore não é tão

Figura 5 – Fluxograma com as regras de decisão da árvore de decisão apresentada na Figura 4




competitiva em quanto os outros métodos de aprendizado supervisionado (JAMES et al., 2013).

Para decidir quantas regiões serão criadas, qual a ordem e os pontos de cortes no espaço de variáveis um algoritmo é seguido. Primeiro define-se uma função que representa o erro. Para o problema de classificação usa-se geralmente *Gini index* ou *cross-entropy* como variáveis que mensuram erro na classificação, pois ambas são mais sensíveis em relação ao crescimento da árvore (JAMES et al., 2013).

A Equação (9) define a fórmula para o cálculo do *Gini index*, onde $\hat{p}_{mk}$ representa o percentual de observações da $m$-ésima região que pertencem a classe $k$. Portanto, esta métrica dá uma visão sobre a variância total sobre todas as classes do problema, dessa forma indicando uma medida de pureza de uma região.

$$G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk}) \tag{9}$$

Ainda há a *cross-entropy* que é uma alternativa ao *Gini index*. A Equação (10) apresenta a equação que a define.

$$D = -\sum_{k=1}^{K} \hat{p}_{mk} log(\hat{p}_{mk}) \tag{10}$$

Após definido como o erro da classificação vai ser mensurado, é feito uma abordagem de cima para baixo e de forma gulosa que minimiza o valor do erro em cada iteração. Esse processo é chamado de divisão binária recursiva. Portanto em cada iteração são percorridas todas as variáveis de entrada, que ainda não foram usadas para cortes nos nós pais, e testamos um ponto de corte $s$ que minimizará o erro. Este processo é repetido

recursivamente, dividindo as regiões sucessivamente, até que um critério de parada seja atingido, como quantidade máxima de observações em um nó terminal.

A divisão binária recursiva pode gerar árvores largas e profundas e isto pode causar um sobre-ajuste, *overfitting*, prejudicando a capacidade de generalização do modelo. Para isso existe a técnica de poda, que é achar uma sub-árvore $A_0$ que minimizará o erro. Uma sub-árvore é também uma árvore de decisão, porem ela está contida dentro de uma árvore maior. Para a definição da melhor sub-árvore que minimiza o erro é utilizado dados de validação. A sub-árvore que apresentar o menor erro nos dados de validação será a sub-árvore $A_0$ escolhida.

### 3.4.1 *Ensemble*

*Ensemble Methods*, ou métodos de conjunto, são algoritmos de aprendizado que constroem um conjunto de classificadores e então classifica novas observações a partir de um sistema que leva em consideração todos os classificadores criados, como um sistema de voto ponderado. Esse sistema é geralmente mais acurado que os classificadores individuais que compõem o conjunto (DIETTERICH et al., 2000). As sub-seções a seguir são exemplos de ensembles.

#### 3.4.1.1 *Random Forest*

*Random Forest* utiliza de uma técnica de re-amostragem que reduz a variância do aprendizado do método de aprendizagem supervisionado (JAMES et al., 2013). Dado um conjunto de treinamento $T$ retira-se $\gamma$ amostras, construindo um modelo com cada uma delas. Ao fim existirá $\gamma$ modelos. Para realizar uma predição de uma dada observação, usa-se um estimador como média ou um sistema de voto para problemas de classificação. O sistema de voto consiste em que cada modelo tem um voto para a predição final, onde seu voto é a sua predição para a dada observação. Assim, existirá uma quantidade de votos para cada classe existente, onde a predição final será a classe com mais votos. Como no fim é usado um sistema que leva em conta todos os modelos, o dano causado pelo *overfitting* é mitigado, assim não há tanta preocupação com o tamanho das árvores, quando se tem uma quantidade de estimadores suficientemente grande (JAMES et al., 2013).

Uma característica única do *Random Forest* é que as suas regiões são decididas estocasticamente. Portanto, para cada árvore da floresta ( conjunto de todos os $\gamma$ preditores) é considerado em cada divisão binária apenas $m$ variáveis de entrada como candidatas a limitadores de região. De acordo com James et al. (2013), o valor de $m$ é definido por $m \approx \sqrt{n}$, onde $n$ é o número de variáveis de entrada.

Este fato ajuda a explorar o espaço criado pelas variáveis de entrada de uma forma melhor. Como o algoritmo de criação de uma árvore é guloso, visando sempre minimizar

o erro, pode existir uma variável que acabe atrapalhando o desenvolvimento da floresta, visto que essa variável dominaria as outras em um local, fazendo com que todas as árvores fossem de certa forma parecidas.

A quantidade de estimadores $\gamma$, profundidade das árvores, critérios de parada, função de erro são parâmetros do algoritmo que devem ser sintonizados de acordo com o problema em questão e o conjunto de dados existente.

#### 3.4.1.2 *Boosting* Trees

*Boosting* é também um método de aprendizagem, restrito as árvores de decisão. O seu funcionamento é similar ao *Random Forest*, entretanto, as árvores são criadas de uma forma sequencial, usando a informação das árvores que foram criadas anteriormente.

A principal característica deste modelo é que ele se baseia no aprendizado lento. Para isso é utilizado a ideia de que com modelos simples, pouco melhores que o aleatório (aprendizes fracos), é possível atingir resultados satisfatórios (BISHOP, 2006).

Neste método é criada uma árvore com tamanho fixo de nós folhas, que significa quantidade fixa de regiões que divide o espaço das variáveis. Aos dados de treinamento desta árvore é dado um peso, $w_n$. Após o treinamento, um novo peso é dado para a árvore de decisão seguinte. Este novo peso leva em consideração os erros cometidos pela árvore passada, assim ele usufrui dos erros passados tentando evitá-los no futuro. Após a construção do estimador $y_i$ é atribuído a ele um coeficiente de encolhimento $\alpha$. Este coeficiente ajuda a retardar mais o processo de aprendizagem (JAMES et al., 2013).

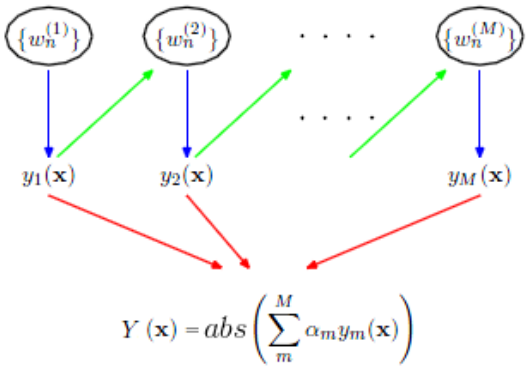
A Figura 6 demonstra o processo completo por trás do *Boosting Trees*. Para um conjunto de entrada $w_n^{(1)}$ tem-se uma árvore de decisão $y_1(x)$. Com sua construção, corrige-se a base de entrada dando um peso maior aos erros cometidos pela árvore que acabara de ser construída, $y_1(x)$, gerando uma nova base de dados $w_n^{(2)}$. Este processo é repetido $M$ vezes, ao termino do qual é feito um processo de voto para decidir a classificação final. O sistema de voto é dado pela Equação (11).

$$Y(x) = abs(\sum_{m=1}^{M} \alpha_m y_m(x))) \tag{11}$$

## 3.5 K Vizinhos mais Próximos - (KNN)

O algoritmo K Vizinhos mais Próximos, do inglês *K-Nearest Neighbor* (KNN), é de muito simples entendimento. É baseado na ideia de que um ponto no espaço de variáveis estará cercado por pontos que em sua maioria compartilharão com ele a mesma classe. A Figura 7 exemplifica o processo de classificação do KNN, com $k$ igual a 3. Neste exemplo a

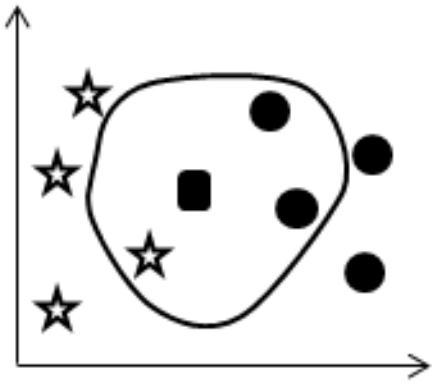
Figura 6 – Ilustração esquemática do algoritmo *Boosting*




Fonte: Adaptado de (BISHOP, 2006)

observação a ser classificada esta representada pelo retângulo, a vizinhança está demarcada onde temos 2 pontos pertencentes a classe “bola” e 1 para classe estrela.Portanto, para este exemplo, a classificação seria classe “bola”.

Figura 7 – Exemplo de uma classificação usando KNN, com $k$ igual a 3



Para realizar uma predição $Y$ de uma observação nova é preciso então calcular a distância, no espaço criada pelas variáveis de entrada, para os outros pontos já existentes e que tem-se o conhecimento de qual classe pertencem. Para isso deve-se primeiramente definir uma função para o cálculo da distância entre os pontos do conjunto. Assumindo que o conjunto de variáveis seja de tamanho $n$, a distância Euclidiana segue a seguinte equação:

$$D(y,p) = \sqrt{\sum_{i=1}^{n}(y_i - p_i)^2}, \tag{12}$$

onde $y$ é a observação que se deseja classificar, $y_i$ vai assumir o valor de cada variável de entrada dentro do somatório e $p$ assume o valor de um ponto do conjunto de treina-

mento. Pode-se utilizar diferentes equações para calcular distância entre dois pontos como: distância de Manhattan, distância de Hamming etc.

Após o cálculo da distância para todos os pontos do conjunto de treinamento, é fixado um $k$, e usa-se a base de validação para decidir qual é o melhor valor para $k$ onde se terá a maior eficácia. Pois se $k$ for muito pequeno a classificação pode ser afetada por ruídos existentes no conjunto de treinamento. Entretanto, se $k$ for muito grande, a ideia de que os indivíduos vizinhos definem a nova observação, baseando na similaridade, é invalidada. Portanto o valor de $k$ deve ser validado, usando o conjunto de validação, para que o algoritmo tenha um melhor desempenho. Vale também ressaltar que $k$ deve ser um número ímpar, pois assim evita o acontecimento de empate entre classes.

Como pode ser observado, para conjunto de dados grandes, ou com muitas variáveis de entrada, tem-se um custo computacional elevado, dependendo da infra-estrutura computacional disponível.

## 3.6 *F1-score*

F1-score é uma métrica que é a média harmônica entre precisão e revocação. Rijsbergen (1979) define precisão como os documentos relevantes recuperados dentre todos os documentos recuperados. O autor também define a revocação como os documentos relevantes recuperados por todos os documentos relevantes.

No campo da classificação, pode-se montar um quadro explanatório, Quadro 1, que possibilita a melhor visualização da predição feita pelo modelo em relação ao resultado esperado.

Quadro 1 – Quadro de diagnóstico preditivo

| | | Resposta Real | |
|---|---|---|---|
| | | Positivo | Negativo |
| Predição | Positivo | Verdadeiro Positivo | Falso Positivo |
| | Negativo | Falso Negativo | Verdadeiro Negativo |

Usando o Quadro 1 para contextualizar as definições de precisão, revocação e acurácia tem se:

$$Prec = \frac{VP}{VP + FP} \tag{13}$$

$$Rev = \frac{VP}{VP + FN} \tag{14}$$

$$Acu = \frac{VP + VN}{VP + FN + VP + VN} \tag{15}$$

onde a Equação (13) é a definição de precisão, a Equação (14) a de revocação, Equação (15) a de acurácia e $VP$ significa Verdadeiro Positivo, $FP$ Falso Positivo, $VN$ Verdadeiro Negativo e $FN$ Falso Negativo.

A equação correspondente à métrica *F1-score* é dada pela seguinte equação:

$$F1 - score = 2.\frac{Prec.Rev}{Prec + Rev} \tag{16}$$

A métrica *F1-score* leva em consideração o tipo de erro cometido pelo preditor, não tratando o erro de uma forma única como a métrica acurácia. Em alguns cenários um Falso Negativo é um erro cujo impacto em uma decisão é crucial. Um exemplo é o caso de presença de uma doença ou gravidez, pois neste cenário, o modelo predizer que não há presença da doença, quando na realidade ela existe, é um erro cujo impacto é severo. Como a métrica *F1-score* leva em consideração os tipos de erros em seu cálculo, ela acaba sendo mais confiável em muitos cenários e por isso é amplamente utilizada.

Para problemas com mais de 2 classes, calcula-se a precisão e a revocação para cada classe. Após agrega-se esses valores na métrica *F1-score* através de um estimador pontual. Neste trabalho utilizamos a média aritmética, seguindo a Equação (16) substituindo $Prec$ pela médias das precisões de cada classe, $Rev$ pela média das revocação de cada classe.

## 3.7 Redução de Dimensão

Bellman (2013) denominou como *curse of dimensionality*, maldição da dimensionalidade, o problema causado pelo aumento exponencial do volume de dados necessários quando se adiciona novas dimensões no espaço de variáveis Euclidiano.

Um problema de aprendizado de máquina com muitas dimensões é um problema com muitas implicações e dificuldades. Uma delas é que em espaços de alta dimensões, o conceito de proximidade ou similaridade perde sentido, já que a proporção do vizinho mais próximo de um objeto sobre seu vizinho mais distante se aproxima do valor 1. Isto significa dizer que em espaços de altas dimensões as observações são aproximadamente equidistantes (SAMMUT; WEBB, 2011).

Portanto a maldição da dimensionalidade afeta a capacidade de aprendizagem dos algoritmos, ou por tornar o efeito da similaridade ineficiente, ou pela necessidade de investigação de um montante exponencial de dados para o treinamento.

Para mitigar os efeitos da alta dimensionalidade é comumente utilizado a seleção das variáveis de entrada ou a utilização de métodos de redução de dimensão. Ambos acabam diminuindo a dimensão do conjunto de variáveis de entrada. O primeiro mantém os valores das variáveis, simplesmente retirando algumas que não contribuem de maneira significativa para o entendimento das classes. Já a segunda forma transforma um conjunto de variáveis de entrada de tamanho $n$ em um outro de novas variáveis, cujo tamanho é $p$, preservando as características das informações presentes no conjunto de variáveis original.

As subseções seguintes explicam duas formas de redução de dimensão que transformam um conjunto de entrada em outro com valores diferentes.

### 3.7.1 PCA

*Principal Component Analysis* (PCA), ou Análise de Componentes Principais, é uma técnica de redução de dimensão, ou simplificação, de um conjunto de dados. Foi desenvolvida por Pearson (1901) e o método consiste em a partir de características originais correlacionadas obter um novo conjunto de dado com novas características que não são correlacionadas, usando transformações lineares. As novas variáveis são chamadas de componentes principais, sendo que a quantidade de componentes é sempre menor ou igual ao número de variáveis do conjunto original.

A construção dos componentes principais tenta extrair a maior variabilidade dos dados possível. Como cada componente deve ser não correlacionada com a outra, dentro desse novo espaço de variáveis isso significa que os vetores dos componentes devem ser ortogonais. Dessa forma o primeiro componente representará a maior variabilidade do conjunto de dados original, o segundo componente principal explicará a segunda maior variabilidade dos dados e assim por diante. Geralmente apenas alguns dos primeiros principais componentes já se responsabilizam por grande parte da variabilidade do dado.

Para explicar o cálculo dos componentes principais serão apresentados os princípios básicos.

#### 3.7.1.1 Covariância

A covariância é uma medida de relação linear entre as variáveis. Se o sinal da covariância é positiva as duas variáveis seguem o mesmo comportamento, se uma aumenta a outra também. Caso seja negativa, significa que se uma aumenta o seu valor a outra diminui. E por fim, se o valor da covariância for nulo, então não existe relação entre as variáveis.

A formula que a define é:

$$cov(A,B) = \frac{\sum_{i=1}^{n}(a_i\mu_a)(b_i\mu_b)}{n-1}, \tag{17}$$

onde $\mu_a$ e $\mu_b$ são as médias das variáveis $A$ e $B$ respectivamente.

Para três ou mais variáveis o cálculo de todas as covariâncias pode ser feito através de uma matriz de covariância, definida pela Equação (18).

$$cov(A,B,C) = \begin{pmatrix} cov(A,A) & cov(A,B) & cov(A,C) \\ cov(B,A) & cov(B,B) & cov(B,C) \\ cov(C,A) & cov(C,B) & cov(C,C) \end{pmatrix} \tag{18}$$

#### 3.7.1.2 Autovalor e Autovetor

Seja $Z$ uma matriz de ordem $n \times n$, $\lambda$ é o autovalor de $Z$ se existe um vetor $v$ tal que

$$Zv = \lambda v, \tag{19}$$

onde $v$ é chamado de autovetor de $Z$ que é associado a $\lambda$. Pode-se reescrever a Equação (19) como

$$(Z - \lambda I)v = 0, \tag{20}$$

onde $I$ é a matriz identidade. Para existir um vetor $v$ que satisfaça a Equação (20), $(Z - \lambda I)$ deve ser não inversível. Portanto, o determinante de $p(\lambda) = det(Z - \lambda I) = 0$. Para se calcular o i-ésimo autovetor $v_i$ associado a um autovalor $\lambda_i$ basta resolver o sistema $p(\lambda) = 0$.

#### 3.7.1.3 Algoritmo PCA

O primeiro passo é selecionar a base de dados, $X$ de tamanho $n$, que será aplicada ao PCA. Então calcula-se a matriz de covariância das variáveis de entrada, $x_i, \ldots, x_n$. Após o cálculo da matriz, calcula-se os autovalores e autovetores da matriz de covariância. Feito isso, escolhe-se os componentes principais que mais trarão variabilidade dos dados. E então mapeia-se todos os dados da base de dados $X$ para a nova base de dados $X_{pca}$.

Esse processo pode ser melhor visualizado no algoritmo 1.

**Algoritmo 1:** PCA

**Entrada:** Base de dados $X$, Quantidade de variáveis $n$, Quantidade de Componentes $p$
**Saída:** Base de dados $X_{pca}$
**início**
  $M$ = matriz de covariância das entradas $x_i, \ldots, x_n$
  $A_{val}$= Autovalores de $M$
  $A_{vet}$= Autovetores de $M$
  Seleciona os $p$ primeiros autovetores, em relação a variabilidade
  $X_{pca}$ = $X.A_{vet}$
**fim**
**retorna** $X_{pca}$

### 3.7.2 LDA

A Análise Discriminante Linear, do inglês *Linear Discriminant Analysis* (LDA), que também é conhecida por discriminantes lineares de Fisher, é uma técnica de estatística multivariada. Ela vem sendo utilizada com sucesso como técnica de redução dimensional para problemas de classificação (YU; YANG, 2001).

Sabe-se que a direção dos autovetores de PCA não necessariamente indicam a melhor direção para uma classificação. Baseando-se em projeções das amostras para um novo espaço vetorial, o LDA foi proposto com um critério de maximização da separação entre duas ou mais classes. O objetivo é encontrar um novo espaço para as novas amostras, $X_{LDA}$, em que a razão entre determinante da matriz inter-classe $S_b$ e a matriz intra-classe $S_w$ seja maximizada (BELHUMEUR; HESPANHA; KRIEGMAN, 1997).

Este método projeta um espaço de n-dimensões em um de $K-1$ dimensões, onde $K$ é a quantidade de classes existentes no problema. Para isso, primeiro calcula-se a matriz de dispersão intra-classe $S_w$. As equações 21, 22 e 23 definem o cálculo da matriz de dispersão intra-classe, onde: $\omega_i$ são as observações do conjunto de dados que pertencem a classe $i$ e $N_i$ é a quantidade de observações pertencentes a classe $i$.

$$S_w = \sum_{i=1}^{K} S_i \tag{21}$$

$$S_i = \sum_{x \in \omega_i} (x - \mu_i)(x - \mu_i)^T \tag{22}$$

$$\mu_i = \frac{1}{N_i} \sum_{x \in \omega_i} x \tag{23}$$

Para a matriz de dispersão entre-classes, $S_b$ é definida pela Equação (24) e Equação (25).

$$S_b = \sum_{i=1}^{K} N_i(\mu_i - \mu)(\mu_i - \mu)^T \tag{24}$$

$$\mu = \frac{1}{N} \sum_{\forall x} x \tag{25}$$

onde $N$ é o número total de observações presente na base de dados. A matriz de dispersão total é dada por $S_t = S_w + S_b$.

Por fim basta reduzir o espaço em $K-1$ vetores $w_i$, que são vetores colunas da matriz $W = [w_1|...|w_{k-1}]$, onde a definição dos novos valores no espaço reduzido, ou

valores das amostras projetadas, será dada pela Equação (26).

$$y_{LDA} = W^T x \tag{26}$$

Com a definição de uma base $W$ precisa-se agora definir a melhor base algébrica $W$ que maximizará a separabilidade das classes. De forma análoga calcula-se $\hat{S}_b, S_w, \hat{\mu}_i, \hat{\mu}$.

O objetivo é maximizar a razão entre dispersão entre-classe $\hat{S}_b$ e intra-classe $\hat{S}_w$. Como ambos são matrizes, é usado o determinante para a determinação da função objetivo $J(W)$. Por manipulações algébricas conseguimos definir que $\hat{S}_w = W^T S_w W$ e $\hat{S}_b = W^T S_b W$.

$$J(W) = \frac{|\hat{S}_b|}{|\hat{S}_w|} = \frac{|W^T S_b W|}{|W^T S_w W|} \tag{27}$$

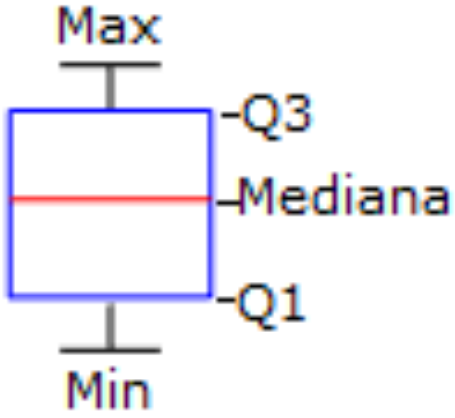
Portanto a projeção ótima é uma matriz $W^*$ cujas colunas são autovetores associados aos maiores autovalores e que pode ser obtida resolvendo o problema dado pela Equação (28).

$$W^* = [w_1^* | \dots | w_{K-1}^*] = argmax \left[ \frac{W^T S_b|}{W^T S_w W} \right] \implies (S_b - \lambda_i S_w) w_i^* = 0 \tag{28}$$

## 3.8 Diagrama de Caixa

O Diagrama de Caixa é uma ferramenta de visualização da distribuição de um conjunto de dados contínuo e unimodal, por meios de quartis. O diagrama é composto pelo valor minimo, primeiro quartil, mediana, terceiro quartil e o valor máximo, de um conjunto de dados. A Figura 8 demonstra uma caixa presente no diagrama.

Figura 8 – Explicação dos valores presentes para a formação do diagrama de caixa. Onde Q significa Quartil.



O diagrama de caixa é utilizado para visualizar de forma resumida o conjunto de dados e realizar comparações entre grupos de dados. Quando é colocado diversas caixas num mesmo diagrama, podemos comparar o conjunto de dados. Se não houver interseção entre o corpo da caixa, composto pelo Q1 e Q3, podemos dizer que os grupos em questão são diferentes. Caso haja intersecção não pode-se afirmar nada.

# 4 Metodologia

Este capítulo descreve a metodologia e modelagem proposta para o problema tratado nessa dissertação e a representação dos dados servirá como entrada para os algoritmos de Aprendizado de Máquina.

A base de dados utilizada neste trabalho foi extraída do sitio eletrônico Kaggle (2016), um site de competições de modelos preditivos. A comunidade de usuários é muito ativa, fomentando sempre o conhecimento.

A base de dados utilizada contém mais de vinte e cinco mil jogos de futebol de onze ligas europeias de 2008 a 2016. Ela consiste em 7 conjunto de dados: *Country* (País), *League* (Liga), *Match* (Jogo), *Player* (Jogador), *Player_Attributes* (Atributo dos Jogadores), *Team* (Times) e *Team_Attributes* (Atributo dos Times). Os dados mais relevantes para esse trabalho referente aos dados da partida contidos no conjunto de dados *Match*, pois são os atributos que descrevem o que ocorreu numa dada partida.

Os principais campos referente a uma partida são: *home team goals* (gols do time mandante), *away team goals* (gols do time visitante), *league_id* (id da liga), *season* (temporada), *stage* (rodada), *date* (data) e *bet365 odds* (prêmio pagos pela Bet365). Estes são atributos que nos dão mais informações e também há poucos valores nulos nestas colunas. Portanto, este conjunto de dados será utilizada para realizar criações de novas características, ou *features*, para melhorar a descrição de um jogo.

As *odds*, ou prêmios pagos, são multiplicadores associados a um investimento na casa de aposta, que caso sua aposta seja vencedora, o seu retorno será dado de acordo com este fator. Para seu cálculo é seguido a equação $odds = \frac{1}{Prob(e)}$, onde $e$ é um evento associado a *odds*. Suponhamos que o time mandante tenha $odds = 1.45$. Isto significa que a casa de aposta avalia que o time mandante tem $68.97\%$ de chances de sair vencedor ao final da partida. Estes valores serão úteis pois dão uma dimensão entre a diferença de qualidade entre as equipes. Para definição desses valores, as casas de apostas levam em consideração alguns índices de desempenho, mas dão um peso maior para as avaliações de seus *oddmakers*, os especialistas daquelas ligas e times em questão.

## 4.1 Representação dos dados

Os campos previamente citados fornecem informações básicas sobre o que aconteceu em uma partida. Entretanto é possível derivar novos dados que auxiliam a descrever melhor uma partida.

Como futebol é um esporte suscetível a várias aleatoriedades, tais como: fatores

físicos, técnicos e psicológicos que influenciam o desempenho de uma equipe, mensurar a qualidade de uma equipe, transformando essa qualidade em um número, é algo altamente subjetivo e complicado.

Em um jogo de futebol pode-se dividir uma equipe em três setores: defesa, meio de campo e ataque. Em geral, os especialistas afirmam que times com defesa e meio de campo fortes tendem a sofrer menos gols, enquanto times com ataque e meio de campo fortes tendem a fazer muitos gols. Portanto, indicadores de qualidade da defesa, meio de campo e ataque dos times parecem ser entradas interessantes para a modelagem desse problema, pois auxiliam a diminuir a subjetividade.

A partir dos dados disponíveis e seguindo a ideia de tentar minimizar a subjetividade pode-se descrever melhor uma partida com as seguintes variáveis para cada time:

- Número de vitórias, empates e derrotas;
- Gols marcados, sofridos, média de gols marcados, média de gols sofridos e média de gols num jogo onde essa equipe estava envolvida;
- Número de jogos em que:
  - marcou pelo menos 1 gol;
  - sofreu pelo menos 1 gol;
  - marcou e sofreu pelo menos 1 gol;
  - teve pelos menos 2 gols na partida em que o time estava envolvido;
  - teve pelos menos 3 gols na partida em que o time estava envolvido;
  - teve pelos menos 4 gols na partida em que o time estava envolvido;
  - venceu sem sofrer gol;
  - perdeu sem marcar um gol;
  - não marcar um gol;
  - não sofreu gol (*Clean Sheet*).

É sabido que o mando de campo tem influência no desempenho de um time, pois jogar ao lado de sua torcida ou sob a pressão da torcida adversária exerce um efeito psicológico no time. Portanto para a predição de um jogo é interessante avaliar o desempenho do time mandante nos jogos em que ele atuou em casa e o desempenho do time visitante nos jogos em que atuou fora de casa.

No futebol, suspensões, lesões e transações de jogadores entre equipes é algo muito comum durante um campeonato. A mudança de um elenco durante um campeonato pode afetar o desempenho da equipe. Para que a rede seja capaz de captar essas variações é interessante levar em consideração o desempenho da equipe nos últimos jogos.

Em relação ao impacto da suspensão, lesões e transferências de jogadores, avaliar a performance de um time desde o início da temporada não parece ser a melhor maneira. Estes indicadores estatísticos de desempenho, submetidos a uma análise temporal mais re-

cente, considerando somente os últimos 5 jogos disputados, é possível visualizar a mudança de valores destes indicadores quando é aplicado essa janela de tempo no desempenho. Parece ser mais fácil predizer um próximo comportamento quando se analisa somente os últimos eventos. A Figura 9 demonstra os gols por jogo de uma forma global, desde o início da temporada. Já a Figura 10 demonstra o indicador gols a favor por jogo de uma forma local, considerando somente os últimos 5 jogos. As figuras esclarecem essa diferença nos valores dos índices quando aplicamos essa janela de tempo no desempenho.

Figura 9 – Gols marcados por jogo do Liverpool na temporada 2014/2015.

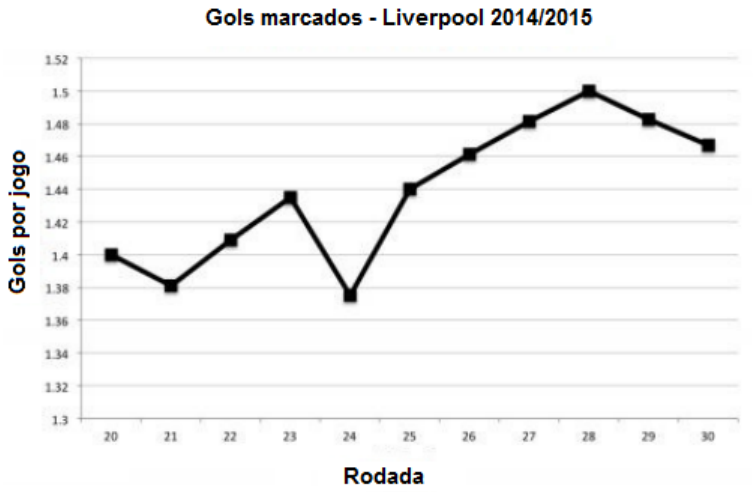
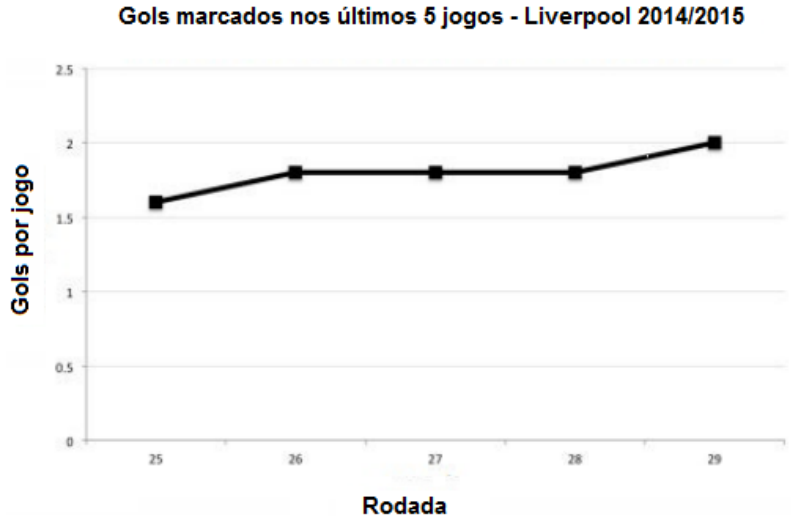



Figura 10 – Gols marcados por jogo nos últimos 5 jogos do Liverpool na temporada 2014/2015.




Ainda pode-se acrescentar o campo referente à diferença em dias do jogo em questão para o último jogo, assim criando uma variável para o mandante e uma para o visitante. A intenção é captar o fator de cansaço, pois uma partida disputada após um

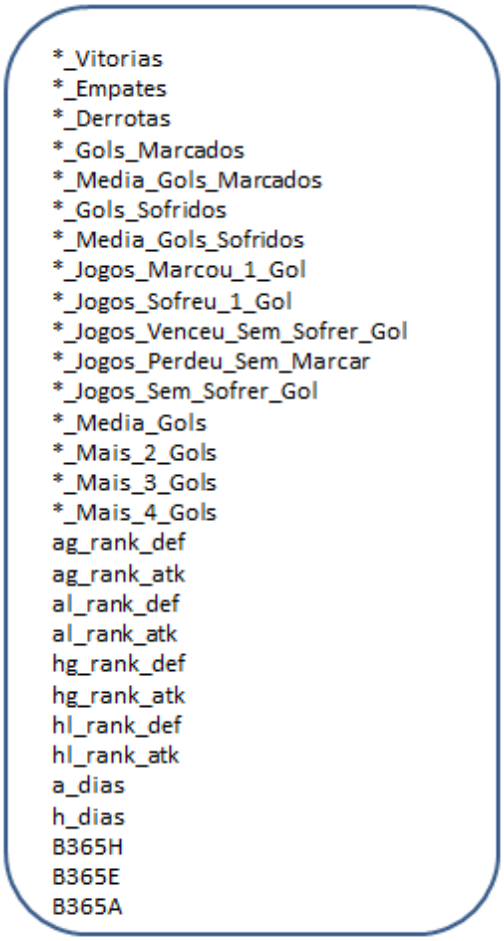
curto período de descanso pode afetar o rendimento dos atletas, a tornando uma variável interessante a ser adicionada ao modelo de entradas.

Outro fator interessante a se considerar é a performance dos setores defensivos e ofensivos quando comparados aos outros times da liga. Para isso, um *rank* pode ser utilizado para captar a influência dos setores em relação a todas as equipes do campeonato.

Finalizando, o valor das *odds* da Bet365 também pode ser utilizado, pois fornece uma dimensão entre a diferença de qualidade das equipes. A Bet365 é uma das maiores casas de apostas do mundo, portanto sua análise sobre a qualidade das equipes tem um valor subjetivo interessante a ser agregado.

Combinando todas as ideias previamente descritas temos um modelo de entradas descrito pela Figura 11, onde o prefixo $h$ significa mandante, $a$ significa visitante, $g$ global, $l$ local e $c$ significa somente os jogos daquela equipe sob a mesma condição que o jogo em questão, mandante ou visitante. O $*$ significa todas as combinações possíveis para os prefixos: $hg$,$hgc$,$hl$,$hlc$,$ag$,$agc$,$al$,$alc$.

Figura 11 – Modelo de entradas proposto neste estudo.



### 4.1.1 Base de Dados

Tanto para o Modelo Primário e Modelo Proposto de dados o processo de tratamento e separação em treino, validação, teste e avaliação seguiram as mesmas premissas. Para cada temporada, os cinco primeiros jogos de cada equipe não são jogos que entram como exemplos para a base de dados. Logo, eles não são jogos que entrarão como referência para os algoritmos.

Os dados de teste são utilizados para definir a performance dos algoritmos em relação a métrica *F1-score*. Com essa definição, as comparações de performance são feitas utilizando os dados de teste. Já os dados de avaliação são utilizados justamente para medir a performance do Melhor Modelo Individual- (MMI) aplicado a dados que não foram usados em nenhuma etapa do processo de definição do MMI.

A base de dados consiste em 8 temporadas consecutivas, 2008/2009 ... 2015/2016, das 7 principais ligas europeias: Inglesa, Francesa, Alemã, Italiana, Holandesa, Portuguesa e Espanhola.

As 5 primeiras temporadas são separadas como treinamento. A sexta temporada é utilizada como validação para os modelos. A sétima é utilizada como teste. Já a oitava, a de avaliação, é utilizada para mensurar a acurácia do Melhor Modelo Individual (MMI) e sua possível rentabilidade.

A Tabela 1 apresenta a distribuição de classes nos dados utilizados no treinamento, validação, teste e desempenho.

Tabela 1 – Tabela com a distribuição de ocorrência de cada classe nos dados de treino, validação, teste e avaliação.

| Dado | Percentual de Distribuição | | |
|---|---|---|---|
| | Mandante | Empate | Visitante |
| Treino | 47.16% | 25.35% | 27.49% |
| Validação | 46.49% | 23.95% | 29.56% |
| Teste | 45.59% | 25.36% | 29.05% |
| Avaliação | 44.57% | 25.54% | 29.89% |

### 4.1.2 Classificação e Regressão

Para os algoritmos cuja tarefa for de classificação eles objetivam classificar um jogo em vitória do mandante, empate ou vitória do visitante.

Os algoritmos que tem como tarefa a regressão precisam um valor referência. Para isso, foi criado uma variável que serve de valor referência, $Y_{reg}$ para os algoritmos cuja tarefa é de regressão. Essa variável alvo é dada pela Equação (29).

$$Y_{reg} = GolsMandante - GolsVisitante \tag{29}$$

Como $Y_{reg}$ é a diferença de gols, os modelos de regressão retornam também em sua saída uma diferença de gol esperada para a partida. Portanto, precisa-se de uma função que converte a diferença de gols em classes: vitória do mandante, empate e vitória do visitante.

$$Classifica(x,K) = \begin{cases} \text{Vitória Mandante, se } x > K \\ \text{Empate, se } -K < x < K \\ \text{Vitória Visitante, se } x < -K \end{cases} \tag{30}$$

Na Equação (30) o valor de $x$ é referente a saída dos algoritmos de Aprendizado de Máquina e o valor $K$ refere-se a um parâmetro que define uma margem de segurança no saldo de gols para definir uma classe.

Para exemplificar, utilizou-se $K = 0$. Nesse caso, um jogo seria classificado como empate se a saída do modelo dissesse exatamente que o resultado do $Y_{reg}$ predito fosse $0$. Se a saída $Y_{reg}$ predita for maior que $0$ seria vitória do mandante e se fosse menor que $0$ seria vitória do visitante. Portanto, quando se treina um modelo de regressão para predição de jogos de futebol, deve-se sintonizar o melhor valor de $K$. Neste trabalho, os valores de $K$ variaram entre $0.25, 0.5, 0.75$ e $1$.

Para as tarefas de Classificação serão utilizados os algoritmos: *Naive Bayes*, Regressão Logistica, KNN, *Random Forest* (RF) e *Gradient Boosting*,(GB). Já para as tarefas de Regressão serão utilizados os algoritmos: *Random Forest* - (RF) e *Gradient Boosting* - (GB), Regressão Linear e KNN.

# 5 Avaliação Experimental

Neste estudo será utilizado os algoritmos de Aprendizado de Máquina para classificação: Regressão Logística (LogReg), *Naive Bayes* (NB), *Random Forest* (RF), *Gradient Boosting* (GB) e *KNN*. Para cada algoritmo um processo de sintonia dos parâmetros é realizada. Executa-se cada algoritmo uma vez para cada combinação possível dos parâmetros sujeito a variação. Utiliza-se a base de dados de validação para seleção do conjunto de parâmetros que maximizem o desempenho de cada algoritmo. Os melhores algoritmos, com o seu conjunto de parâmetros definidos na etapa anterior, são então levados para o teste. Os resultados apresentados neste capitulo são referentes ao desempenho na etapa de teste. O Quadro 2 enumera os parâmetros que serão validados na etapa de sintonia. Se o algoritmo também se aplica a regressão, os parâmetros que devem ser sintonizados seguem os mesmos. Já o Quadro 3 demonstra quais algoritmos utilizados neste trabalho são estocásticos e quais não são.

Quadro 2 – Quadro para sintonia de parâmetros de cada algoritmo

| Algoritmo | Parâmetros |
|---|---|
| LogReg | Algoritmo de minimização: 'newton-cg','sag','liblinear' e 'lbfgs'<br>C= 0.001, 0.01, 0.1, 1, 10, 100 e 1000 |
| RF | Estimadores: 100 a 1000, variando de 50 em 50. |
| GB | Estimadores: 100 a 1000, variando de 50 em 50.<br>Taxa de Aprendizado: 0.0001, 0.001, 0.01 e 0.1 |
| KNN | N. Vizinhos: 1 até 100, variando de 2 em 2<br>Métodos para calculo dos vizinhos: ball_tree, kd_tree e força bruta. |
| NB | Nenhum. |

Quadro 3 – Quadro que apresenta quais algoritmos utilizados nesse estudo são estocásticos.

| Algoritmo | Estocástico |
|---|---|
| RegLog | Sim |
| RegLin | Não |
| RF | Sim |
| GB | Sim |
| KNN | Não |
| NB | Não |

Os experimentos realizados para este estudo foram feitos na linguagem *python* utilizando as bibliotecas *pandas* e *scikit-learn*. O computador possui um processador Intel i3-4170 3.70GHz com 8GB de memória RAM, placa de video GeForce GTX 750TI e SSD 120GB.

Para evitar dependência temporal entre os dados de treino e teste é selecionado, dentro das 8 temporadas disponíveis, numeradas de 0 a 7 em ordem cronológica, as ligas 0 a 4 são utilizadas para o treinamento, 5 para validação e as temporadas 6 e 7 para teste, sendo que a temporada 6 foi utilizada como dado de teste nos experimentos que definem os melhores modelos e a temporada 7 para o experimento de análise do rendimento do modelo.

As 5 primeiras rodadas de cada temporada para cada liga são deixadas de fora da base de dados, pois é uma fase onde acredita-se que a aleatoriedade é mais exacerbada devido à atletas fora de forma, nível baixo de entrosamento e ainda ausência ou pouca quantidade de dados prévios para formação das *features* do modelo proposto.

## 5.1 Classificação

Levando-se em consideração a tarefa de classificação, foi utilizada para avaliar a performance dos algoritmos a métrica *F1-score*. Foram selecionadas 15 sementes (1 até 15) e para cada modelo foram feitas 15 execuções de cada algoritmo, uma para cada semente.

Todos os diagramas de caixas apresentados nas subseções futuras tem um nível de significância de 5%. No diagrama de caixa, quando não existe interseção entre as caixas, pode-se dizer que existe uma diferença entre os conjuntos de dados observados para os grupos analisados.

### 5.1.1 Modelo Primário

Para o Modelo Primário, que consiste em valores de entradas puros, sem nenhuma modificação. Ele consiste no id do time mandante, id do time visitante, id do campeonato, temporada, rodada e as *odds* da Bet365 B365H, B365D e B365A.

A Figura 12 é o diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário. Como pode-se observar, o algoritmo com o melhor desempenho utilizando este modelo foi o GB, pois apresentou o maior valor referente ao F1 *Score* e sua caixa não tem interseção com nenhuma outra.

### 5.1.2 Modelo Primário usando PCA

O outro modelo possível é utilizar técnicas de redução de dimensão do modelo de entrada. A Figura 13 apresenta as execuções dos algoritmos utilizando a técnica de redução PCA no Modelo Primário.

A Figura 13 deixa claro que o algoritmo KNN tem melhor desempenho para esse tipo de modelo de dados, pois apresentou o maior valor referente ao F1 *Score*.

Figura 12 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário

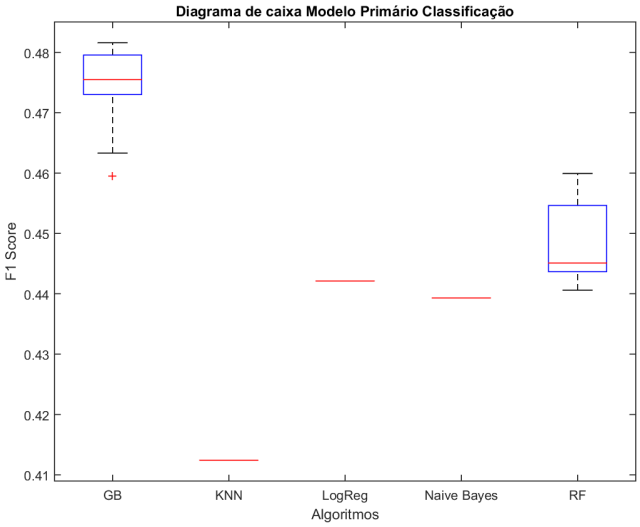
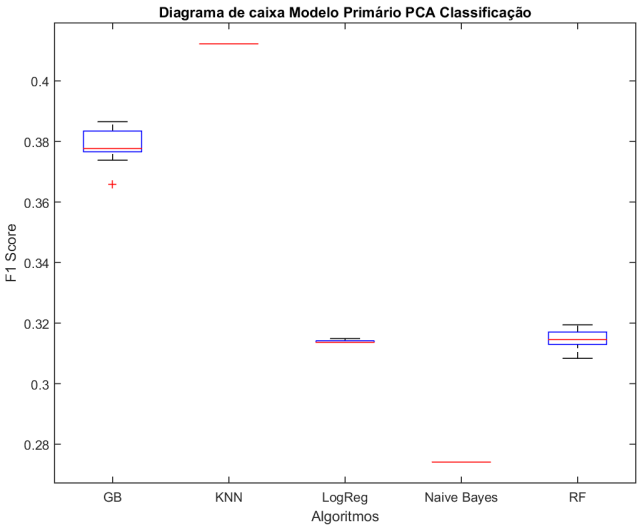


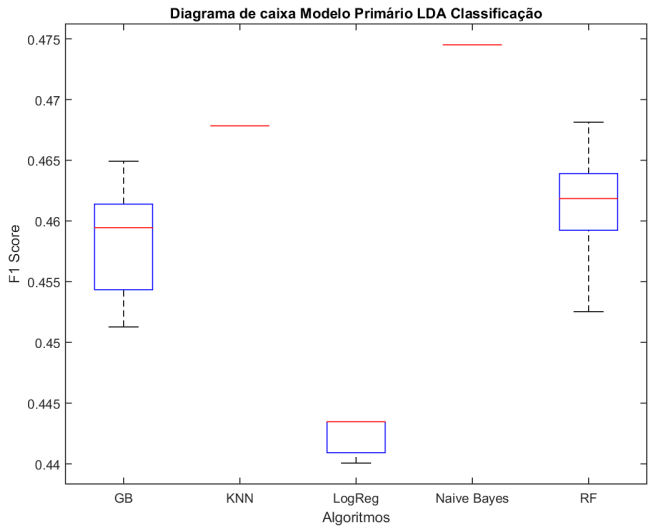

Figura 13 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário com redução de dimensão PCA




### 5.1.3 Modelo Primário usando LDA

A outra forma de redução de dimensão é utilizando a técnica LDA. A Figura 14 apresenta as execuções dos algoritmos com redução LDA no Modelo Primário de dado.

Figura 14 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário com redução de dimensão LDA

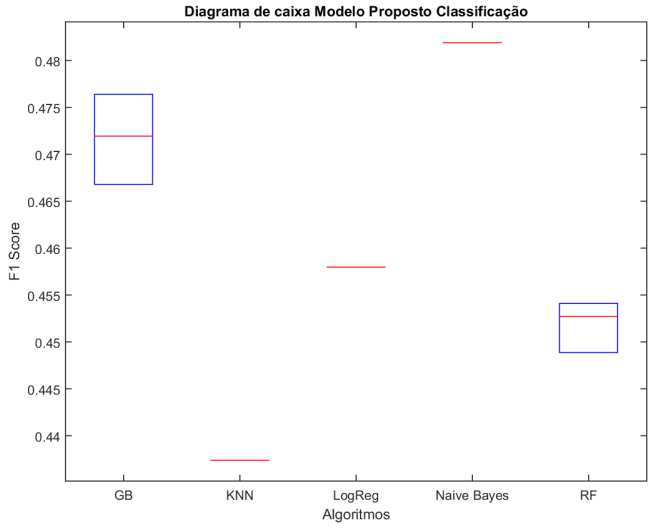



É visível que o modelo NB foi o de melhor desempenho para este modelo de dados com a tarefa de classificação.

### 5.1.4 Modelo Proposto

O Modelo Proposto de entrada, explicado no Capítulo 4 e ilustrado na Figura 11, tem o desempenho dos algoritmos ilustrados através da Figura 15.

Figura 15 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto

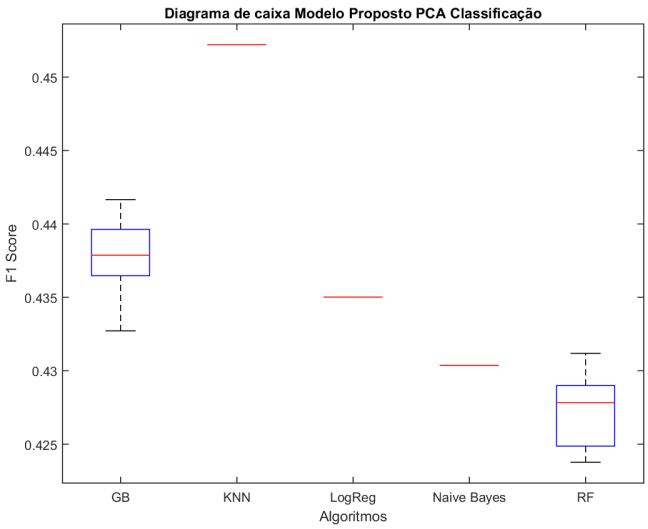



Analisando à Figura 15 pode-se verificar que o algoritmo NB apresentou melhor desempenho.

### 5.1.5 Modelo Proposto usando PCA

Utilizando-se a técnica de redução PCA no Modelo Proposto obteve-se o desempenho exemplificado na Figura 16.

Figura 16 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto com redução de dimensão PCA




É fácil perceber que o algoritmo KNN é o que obteve o melhor desempenho em relação a métrica F1.

### 5.1.6 Modelo Proposto usando LDA

Também pode-se utilizar a técnica de redução LDA no Modelo Proposto. A Figura 17 demonstra os resultados colhidos depois da execução dos algoritmos.

O algoritmo KNN foi que apresentou melhor performance para este tipo de modelo de dados.

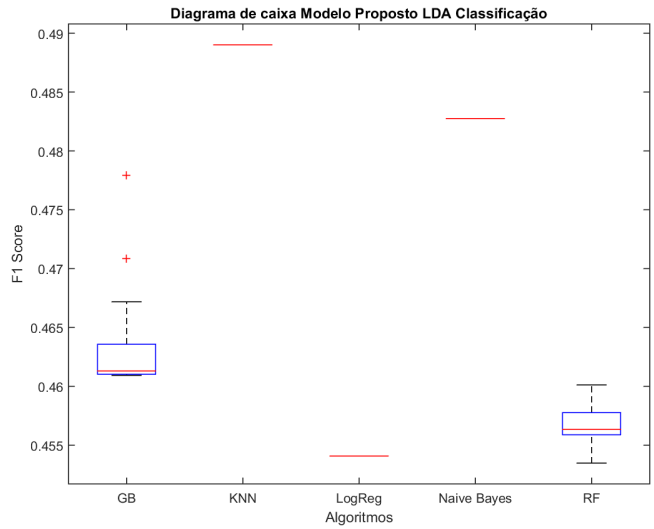
### 5.1.7 Comparação de Algoritmos e Modelos

Para cada tipo de modelo, existem pelo menos um algoritmo que apresenta melhor desempenho. O Quadro 4 apresenta todos os melhores algoritmos para cada tipo de modelo.

Primeiramente, a Figura 18 apresenta o diagrama de caixa do conjunto modelo algoritmo. Cada caixa representa o desempenho do melhor algoritmo submetido a um modelo de dados.

A Figura 18 deixa claro que o conjunto Modelo Proposto com LDA e algoritmo KNN apresentam melhor desempenho dentro de todos os modelos e algoritmos cuja tarefa é classificação apresentados nessa seção.

Figura 17 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto com redução de dimensão LDA




Quadro 4 – Quadro com os melhores algoritmos para cada tipo de modelo e um código de identificação para o conjunto modelo algoritmo.

| Modelo | Algoritmo |
|---|---|
| Primário | GB |
| Primário PCA | KNN |
| Primário LDA | NB |
| Proposto | NB |
| Proposto PCA | KNN |
| Proposto LDA | KNN |

## 5.2 Regressão

Levando-se em consideração a tarefa de regressão, utilizamos para avaliar a performance dos algoritmos a métrica *F1-score*. Foram selecionados 15 sementes (1 até 15) e para cada modelo foram feitas 15 execuções de cada algoritmo, uma para cada semente.

Como exemplificado na seção 4.1.2, temos que levar em consideração o valor de ajuste $K$, pois o modelo de regressão tem como referência o saldo de gols da partida. Este valor $K$ estabelece um limite para a classificação do jogo em vitória do mandante, vitória do visitante e empate. Portanto, para os conjuntos de modelo de dados e algoritmo, o valor de k foi sintonizado.

Todos os diagramas de caixas apresentados nas subseções futuras tem um nível de significância de 5%.

Figura 18 – Diagrama de caixa da execução dos algoritmos com melhor desempenho nos modelos de dados apresentados.

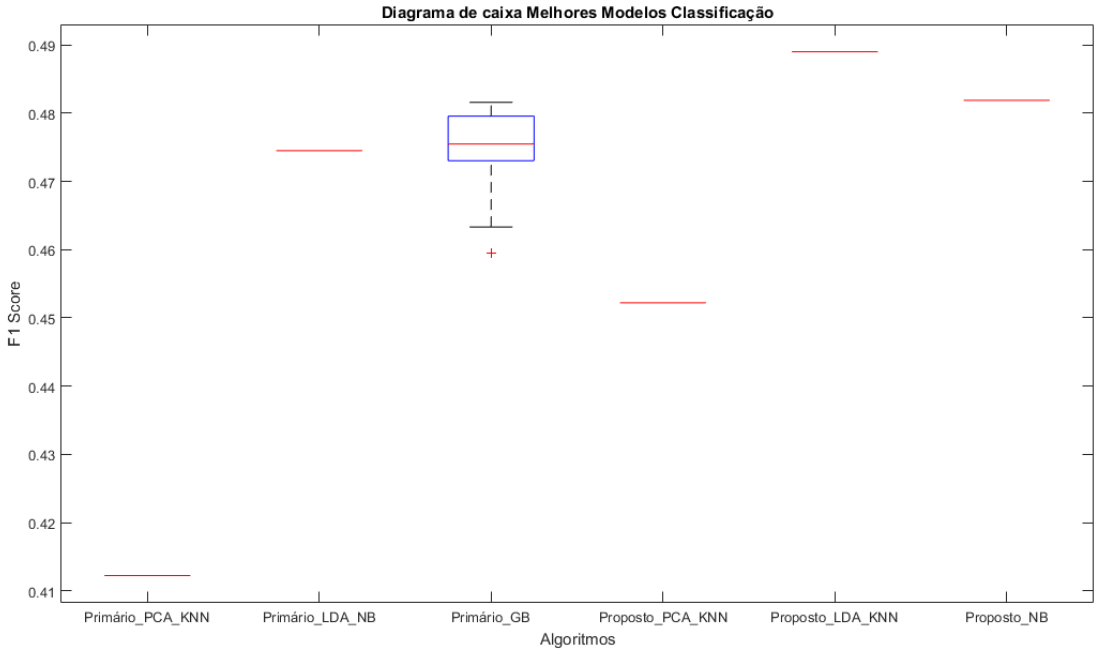
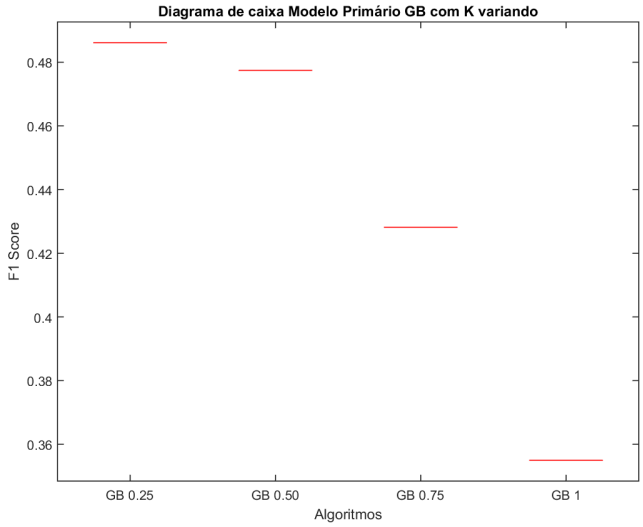



Figura 19 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário com $K$ variando para o algoritmo GB.



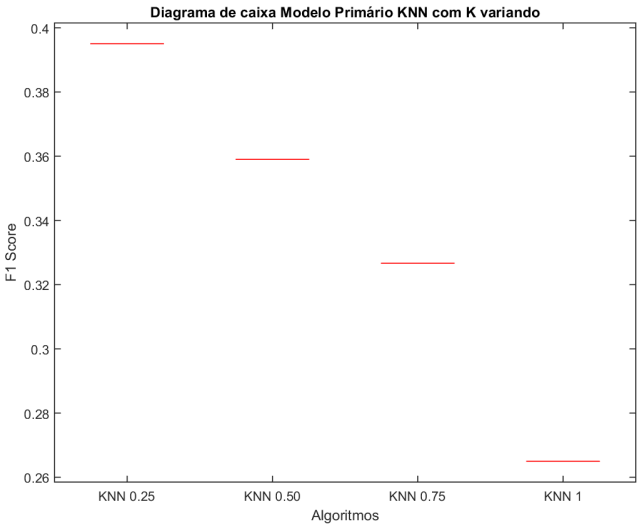

## 5.2.1 Modelo Primário

### 5.2.1.1 Gradient Boosting

A Figura 19 é o diagrama de caixa da execução do algoritmo GB utilizando o Modelo Primário com a tarefa de regressão e com $K$ variando. Como pode-se observar, o melhor valor de $K$ foi $0.25$.

#### 5.2.1.2 KNN

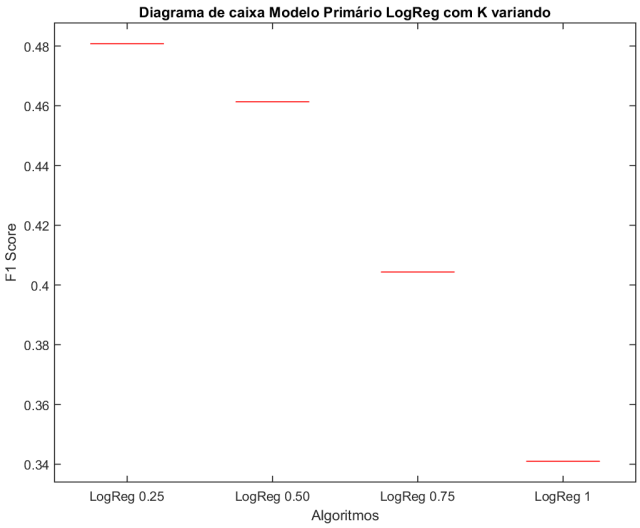
Figura 20 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário com $K$ variando para o algoritmo KNN.




A Figura 20 é o diagrama de caixa da execução do algoritmo KNN utilizando o Modelo Primário com a tarefa de regressão e com $K$ variando. Como pode-se observar, o melhor valor de $K$ foi $0.25$.

#### 5.2.1.3 Regressão Linear

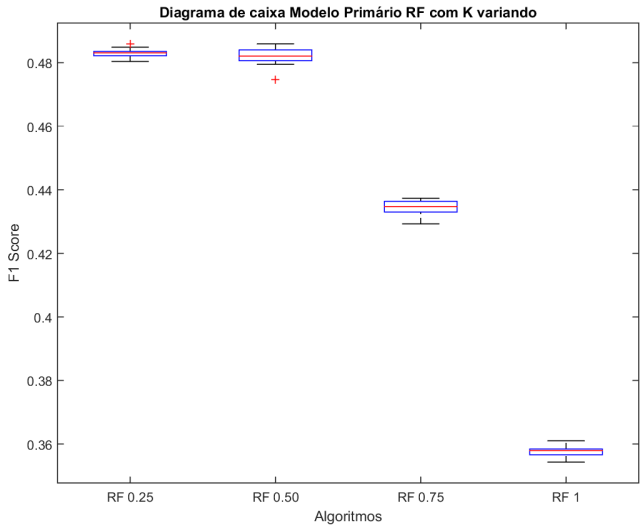
Figura 21 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário com $K$ variando para o algoritmo Regressão Linear.




A Figura 21 é o diagrama de caixa da execução do algoritmo RegLin utilizando o Modelo Primário com a tarefa de regressão e com $K$ variando. Como pode-se observar, o melhor valor de $K$ foi $0.25$.

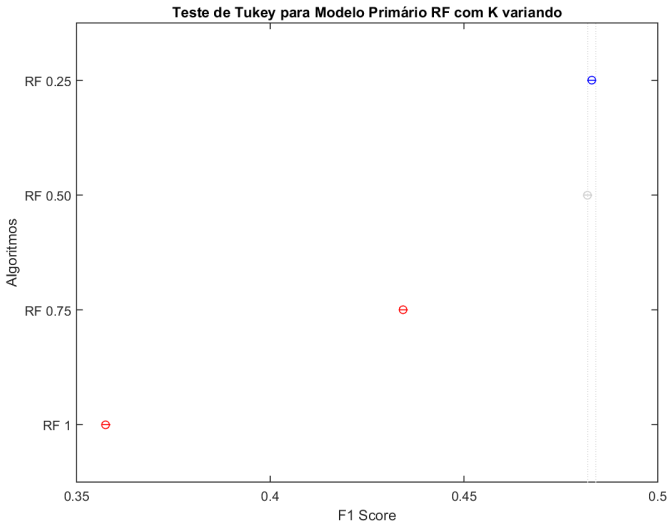
#### 5.2.1.4 Random Forest

Figura 22 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário com $K$ variando para o algoritmo RF.




A Figura 22 é o diagrama de caixa da execução do algoritmo RF utilizando o Modelo Primário com a tarefa de regressão e com $K$ variando. Como pode-se visualizar, existe uma indecisão a respeito dos valores de $K$ iguais a $0.25$ e $0.5$. A Figura 23 demonstra o teste de Tukey com a finalidade de verificar se há uma diferença significativa estatística entre os modelos apresentados.

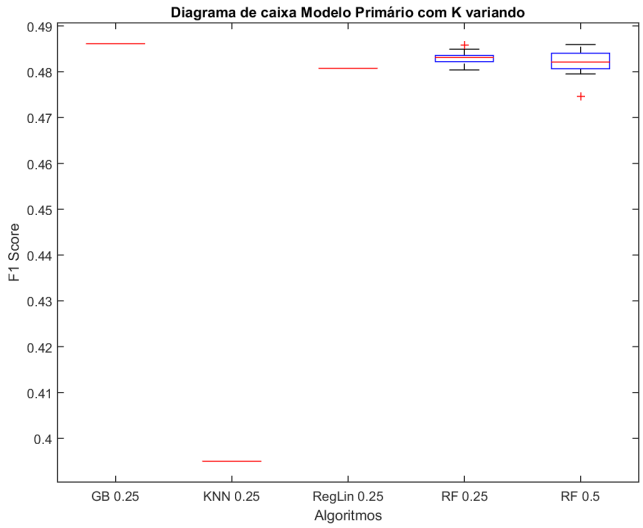
Figura 23 – Teste de Tukey para o Modelo Primário com $K$ variando para o algoritmo de RF.




Como pode-se observar, não pode-se dizer estatisticamente que existe um valor de $K$ melhor entre $0.25$ e $0.5$.

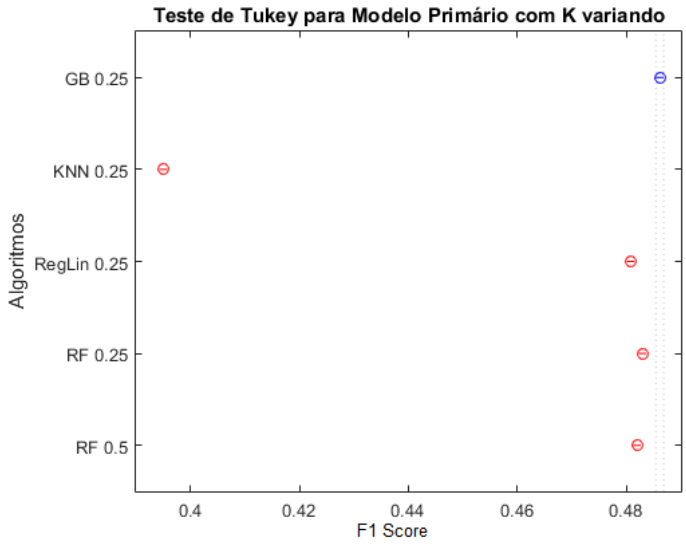
#### 5.2.1.5 Melhor Algoritmo com K variando

Figura 24 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário com $K$ variando para o algoritmo RF.




A Figura 24 é o diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário com a tarefa de regressão e com $K$ variando. É dificil observar graficamente se há ou não uma interseção entre as caixas do "GB 0.25"e do "RF 0.25"e "RF 0.5". Para que não paire uma dúvida, o teste de Tukey foi realizado. A Figura 25 demonstra o resultado do teste graficamente.

Figura 25 – Teste de Tukey para os melhores algoritmos no Modelo Primário com $K$ variando na tarefa de regressão.

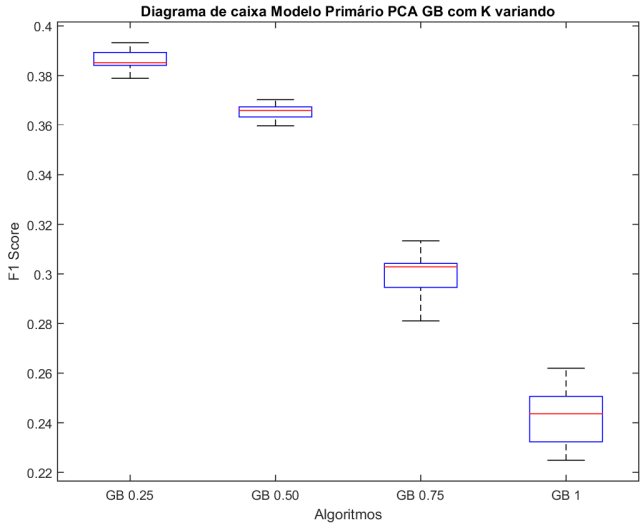



Como pode-se observar, o algoritmo GB com $K$ igual a 0.25 apresentou melhor performance para o Modelo Proposto.

### 5.2.2 Modelo Primário com PCA

#### 5.2.2.1 Gradient Boosting

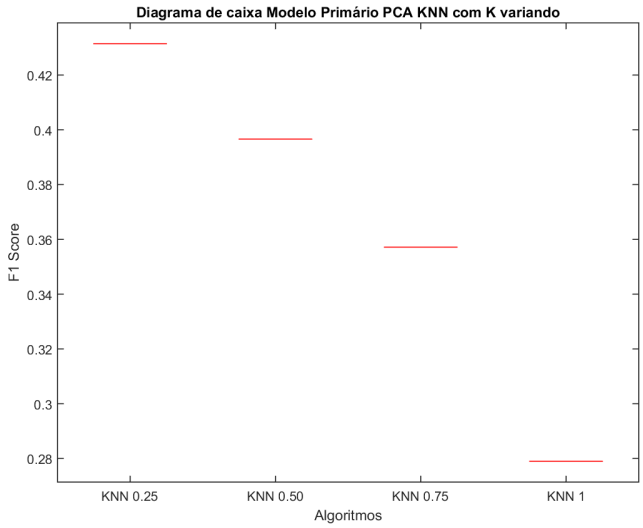
Figura 26 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário PCA com $K$ variando para o algoritmo GB.




A Figura 26 é o diagrama de caixa da execução do algoritmo GB utilizando o Modelo Primário com a tarefa de regressão e com $K$ variando. Como pode-se observar, o melhor valor de $K$ foi $0.25$.

#### 5.2.2.2 KNN

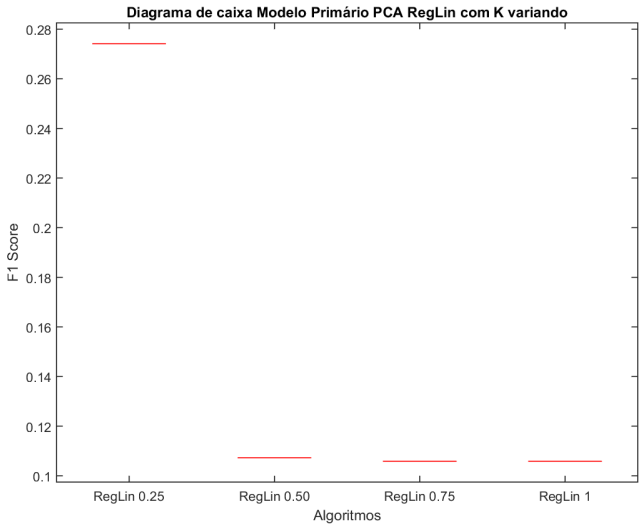
Figura 27 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário com $K$ variando para o algoritmo KNN.



A Figura 27 é o diagrama de caixa da execução do algoritmo KNN utilizando o Modelo Primário com a tarefa de regressão e com $K$ variando. Como pode-se observar, o melhor valor de $K$ foi $0.25$.

#### 5.2.2.3 Regressão Linear

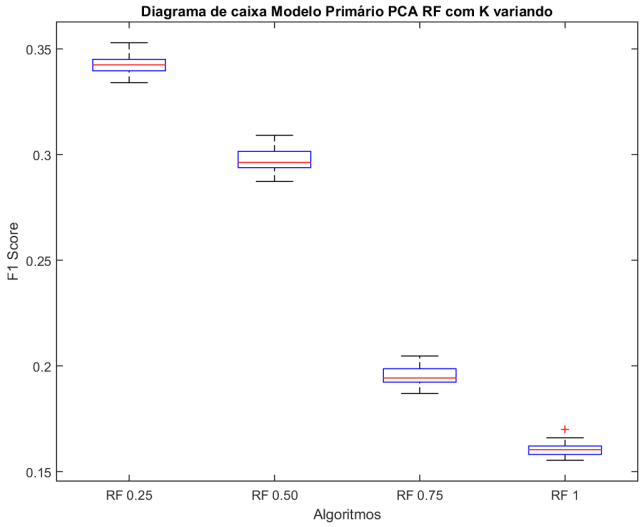
Figura 28 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário PCA com $K$ variando para o algoritmo Regressão Linear.




A Figura 28 é o diagrama de caixa da execução do algoritmo RegLin utilizando o Modelo Primário com a tarefa de regressão e com $K$ variando. Como pode-se observar, o melhor valor de $K$ foi $0.25$.

#### 5.2.2.4 Random Forest

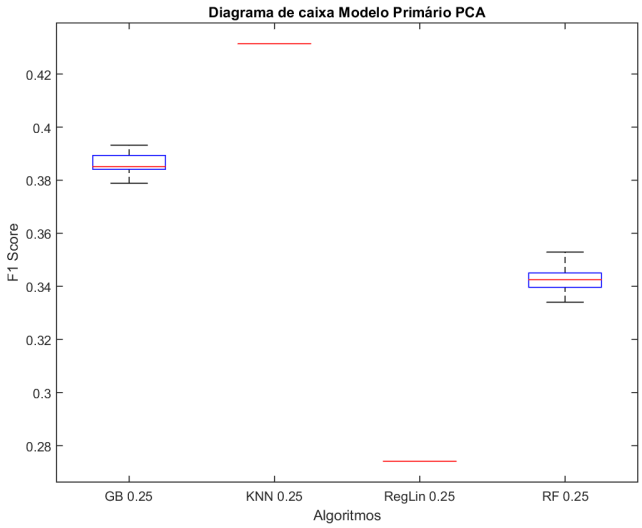
Figura 29 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário PCA com $K$ variando para o algoritmo RF.




A Figura 29 é o diagrama de caixa da execução do algoritmo RF utilizando o Modelo Primário com a tarefa de regressão e com $K$ variando. Como pode-se observar, um valor de $K$ igual a $0.25$ apresenta maior valor de métrica.

#### 5.2.2.5 Melhor Algoritmo com K variando

Figura 30 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário PCA com $K$ variando para o algoritmo RF.



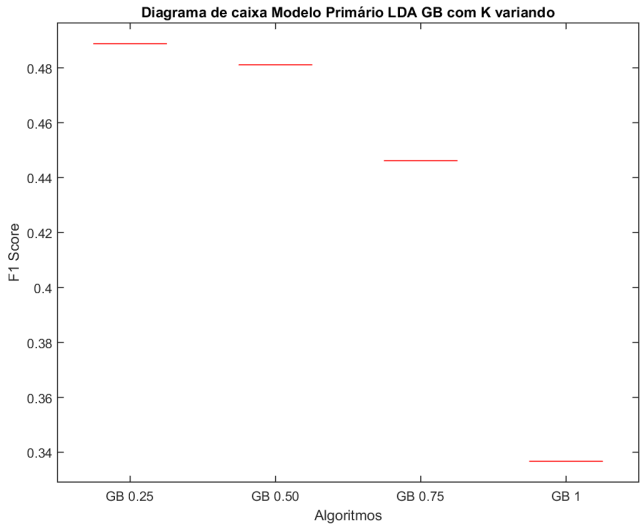

A Figura 30 é o diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário com PCA cuja a tarefa dos algoritmos era de regressão e com $K$ variando. O

algoritmo KNN com $K$ igual a 0.25 apresentou melhor desempenho em relação à metrica *F1-score*.

### 5.2.3 Modelo Primário com LDA

#### 5.2.3.1 Gradient Boosting

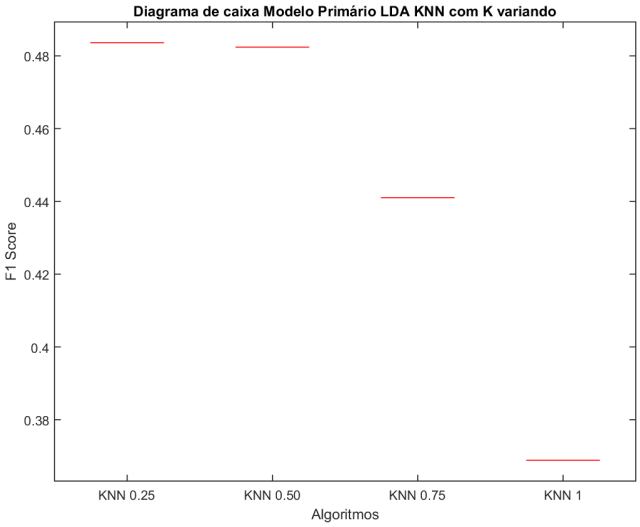
Figura 31 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário LDA com $K$ variando para o algoritmo GB.




A Figura 31 é o diagrama de caixa da execução do algoritmo GB utilizando o Modelo Primário com LDA cuja tarefa é regressão e com $K$ variando. Como pode-se observar, o melhor valor de $K$ foi $0.25$.

#### 5.2.3.2 KNN

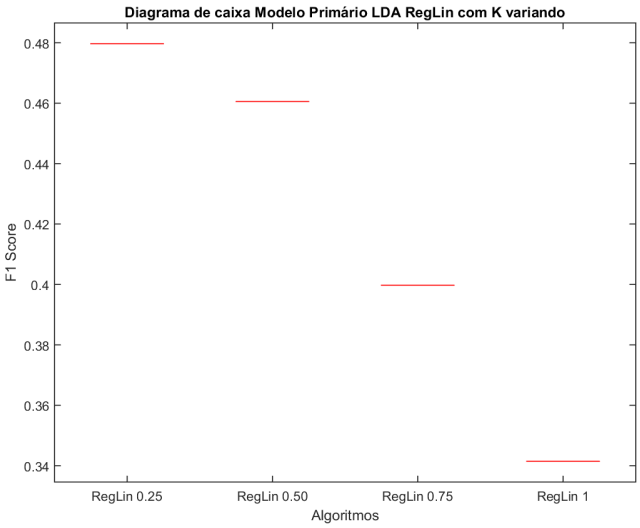
Figura 32 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário LDA com $K$ variando para o algoritmo KNN.




A Figura 32 é o diagrama de caixa da execução do algoritmo KNN utilizando o Modelo Primário com LDA cuja tarefa é regressão e com $K$ variando. Como pode-se observar, o melhor valor de $K$ foi $0.25$.

#### 5.2.3.3 Regressão Linear

Figura 33 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário LDA com $K$ variando para o algoritmo Regressão Linear.

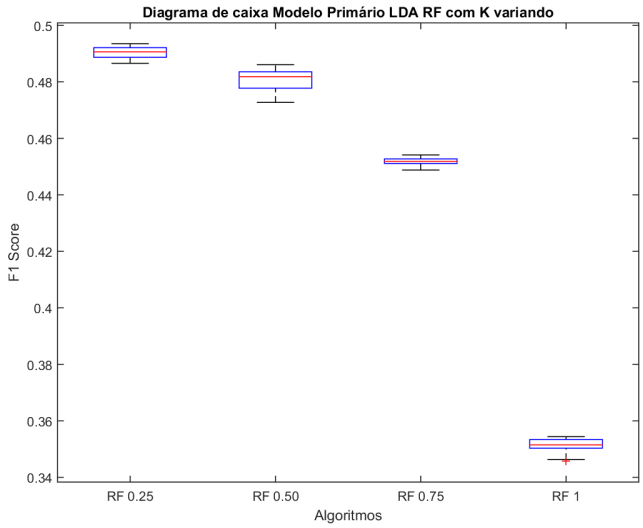



A Figura 33 é o diagrama de caixa da execução do algoritmo RegLin utilizando o Modelo Primário com LDA cuja tarefa é regressão e com $K$ variando. Como pode-se

observar, o melhor valor de $K$ foi $0.25$.

#### 5.2.3.4 Random Forest

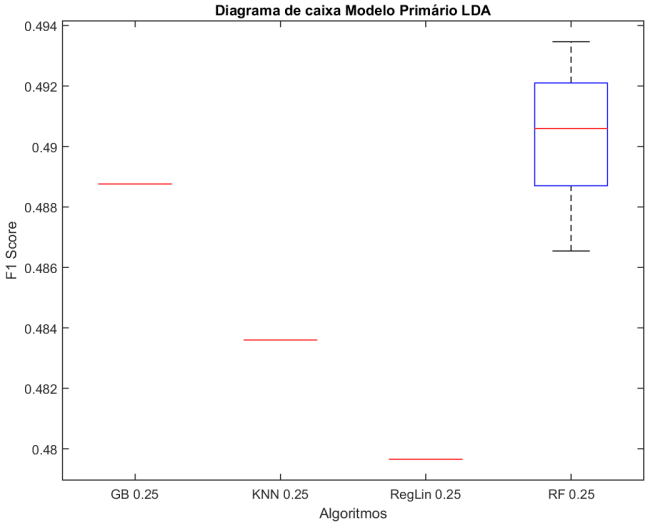
Figura 34 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Primário LDA com $K$ variando para o algoritmo RF.




A Figura 34 é o diagrama de caixa da execução do algoritmo RF utilizando o Modelo Primário com LDA cuja tarefa é regressão e com $K$ variando. Como não interseção das caixas,o melhor valor de $K$ é $0.25$.

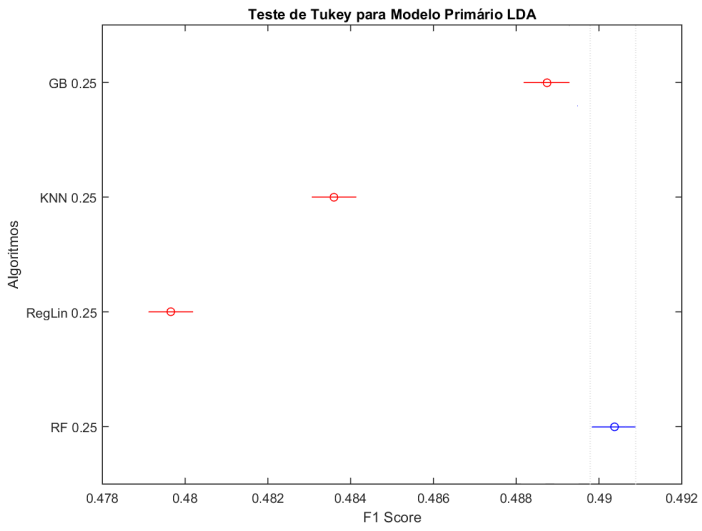
#### 5.2.3.5 Melhor Algoritmo com K variando

Figura 35 – Diagrama de caixa da execução dos melhores algoritmos utilizando o Modelo Primário LDA com $K$ variando.



A Figura 35 é o diagrama de caixa da execução dos melhores algoritmos utilizando o Modelo Primário LDA com a tarefa de regressão e com $K$ variando. A avaliação gráfica se há ou não uma interseção entre as caixas do "GB 0.25" e do "RF 0.25" é complicada. Para que não exista dúvidas, o teste de Tukey foi realizado. A Figura 36 demonstra o resultado do teste graficamente.

Figura 36 – Teste de Tukey para os melhores algoritmos no Modelo Primário com LDA e $K$ variando na tarefa de regressão.

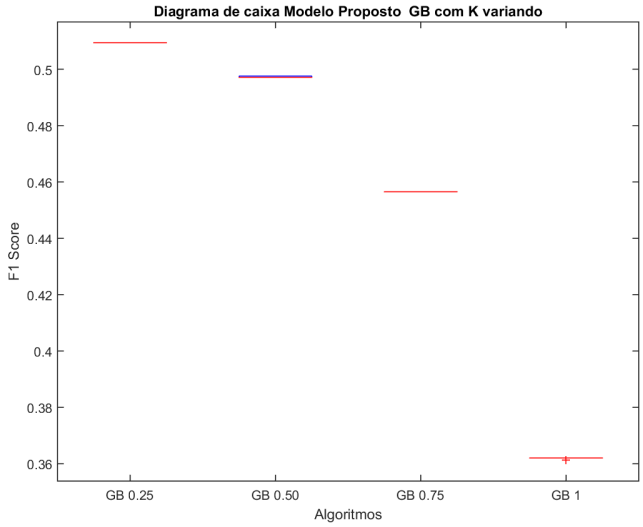



Como pode-se observar, o algoritmo RF com $K$ igual a 0.25 apresentou melhor performance para o Modelo Primário com LDA.

### 5.2.4 Modelo Proposto

#### 5.2.4.1 Gradient Boosting

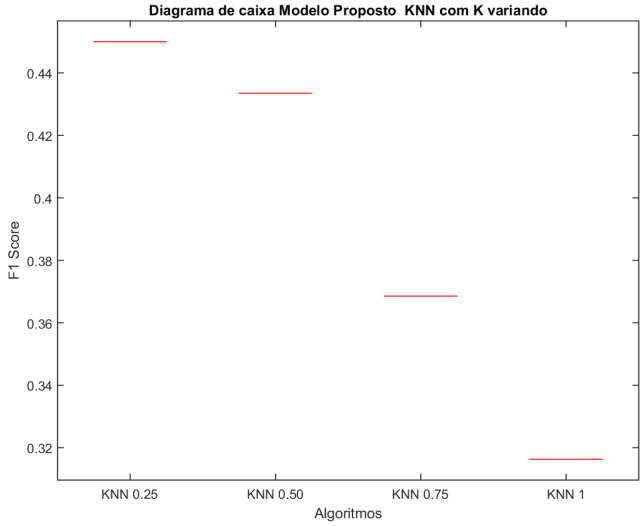
Figura 37 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto com $K$ variando para o algoritmo GB.




A Figura 37 é o diagrama de caixa da execução do algoritmo GB utilizando o Modelo Proposto com a tarefa de regressão e com $K$ variando. Como pode-se observar, o melhor valor de $K$ foi $0.25$.

#### 5.2.4.2 KNN

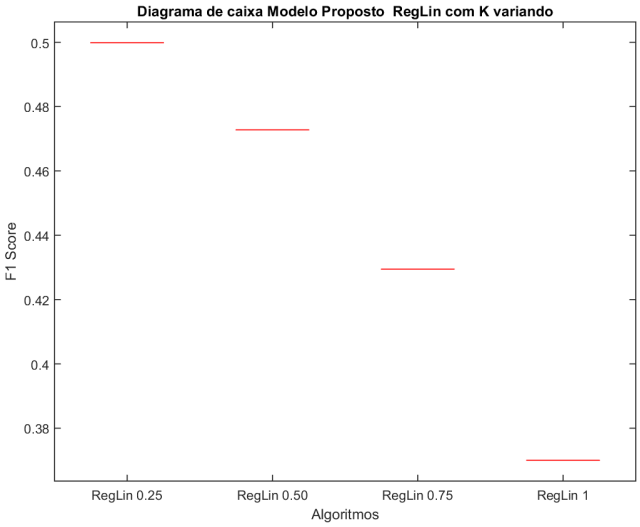
Figura 38 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto com $K$ variando para o algoritmo KNN.



A Figura 38 é o diagrama de caixa da execução do algoritmo KNN utilizando o Modelo Proposto com a tarefa de regressão e com $K$ variando. Como pode-se observar, o melhor valor de $K$ foi $0.25$.

#### 5.2.4.3 Regressão Linear

Figura 39 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto com $K$ variando para o algoritmo Regressão Linear.




A Figura 39 é o diagrama de caixa da execução do algoritmo RegLin utilizando o Modelo Proposto com a tarefa de regressão e com $K$ variando. Como pode-se observar, o melhor valor de $K$ foi $0.25$.

#### 5.2.4.4 Random Forest

A Figura 40 é o diagrama de caixa da execução do algoritmo RF utilizando o Modelo Proposto com a tarefa de regressão e com $K$ variando. Como pode-se observar, o valor de $K$ iguais a $0.25$ apresentou melhor métrica.

#### 5.2.4.5 Melhor Algoritmo com K variando

A Figura 41 é o diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto com a tarefa de regressão e com $K$ variando.

Como pode-se observar, o algoritmo GB com $K$ igual a 0.25 apresentou melhor performance para o Modelo Proposto.

Figura 40 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto com $K$ variando para o algoritmo RF.

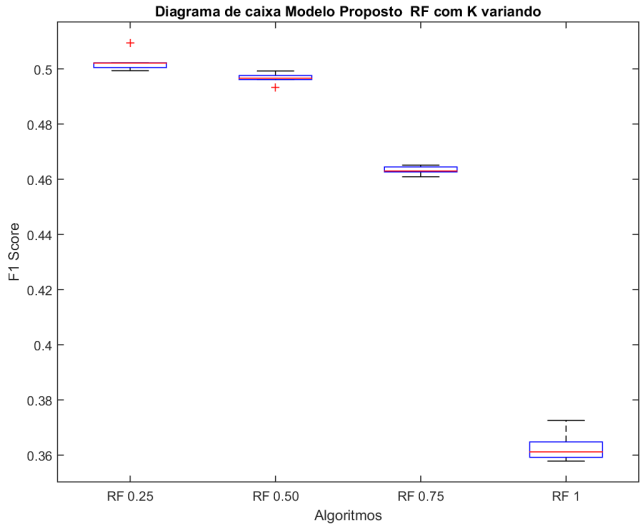
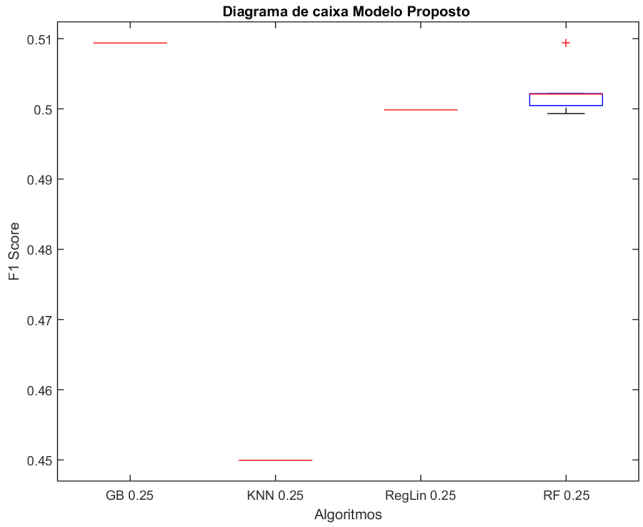


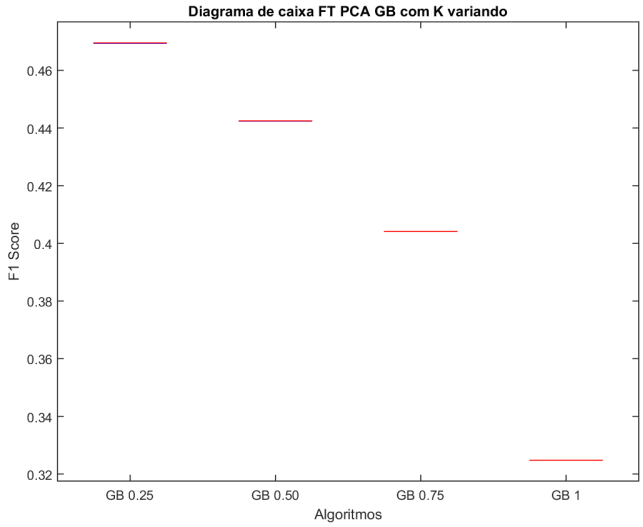

Figura 41 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto com $K$ variando para o algoritmo RF.



### 5.2.5 Modelo Proposto com PCA

#### 5.2.5.1 Gradient Boosting

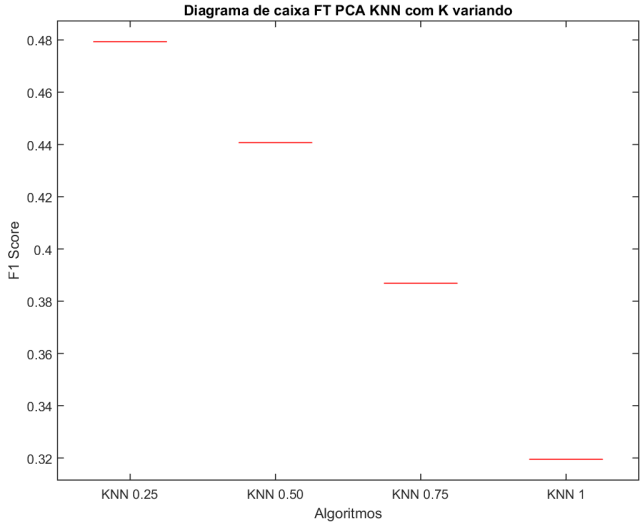
Figura 42 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto PCA com $K$ variando para o algoritmo GB.




A Figura 42 é o diagrama de caixa da execução do algoritmo GB utilizando o Modelo Proposto com a tarefa de regressão e com $K$ variando. Como pode-se observar, o melhor valor de $K$ foi $0.25$.

#### 5.2.5.2 KNN

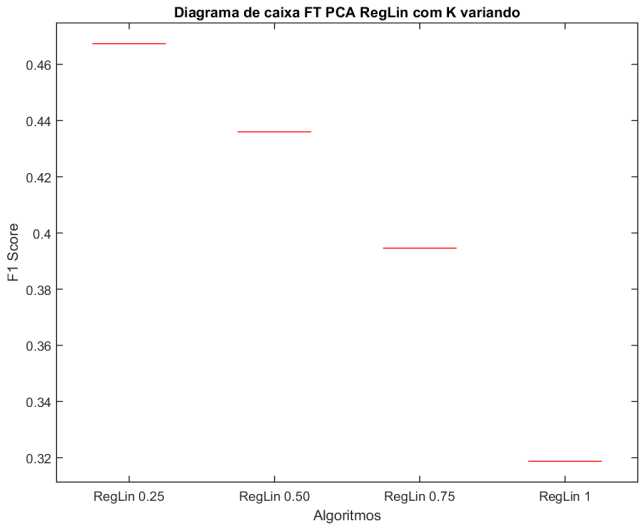
Figura 43 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto com $K$ variando para o algoritmo KNN.



A Figura 43 é o diagrama de caixa da execução do algoritmo KNN utilizando o Modelo Proposto com a tarefa de regressão e com $K$ variando. Como pode-se observar, o melhor valor de $K$ foi $0.25$.

#### 5.2.5.3 Regressão Linear

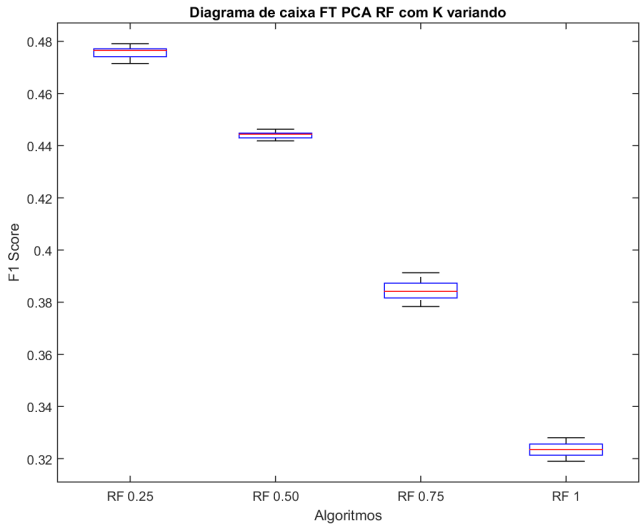
Figura 44 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto PCA com $K$ variando para o algoritmo Regressão Linear.




A Figura 44 é o diagrama de caixa da execução do algoritmo RegLin utilizando o Modelo Proposto com a tarefa de regressão e com $K$ variando. Como pode-se observar, o melhor valor de $K$ foi $0.25$.

#### 5.2.5.4 Random Forest

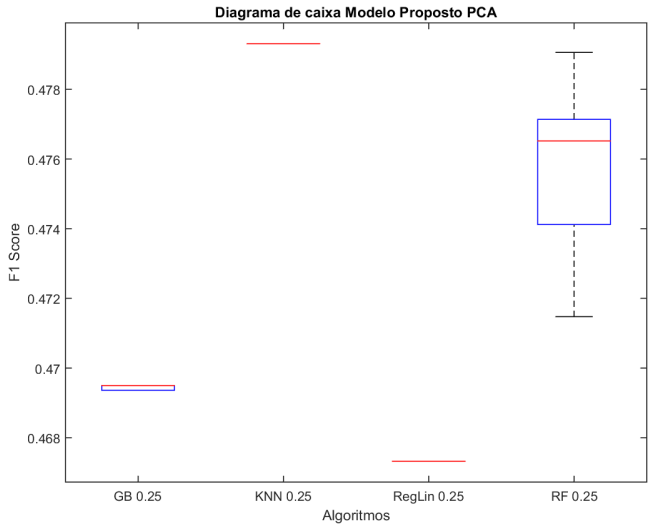
Figura 45 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto PCA com $K$ variando para o algoritmo RF.




A Figura 45 é o diagrama de caixa da execução do algoritmo RF utilizando o Modelo Proposto com a tarefa de regressão e com $K$ variando. Como pode-se observar, existe uma indecisão a respeito dos valores de $K$ iguais a $0.25$ e $0.75$.

#### 5.2.5.5 Melhor Modelo com K variando

Figura 46 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto PCA com $K$ variando para o algoritmo RF.



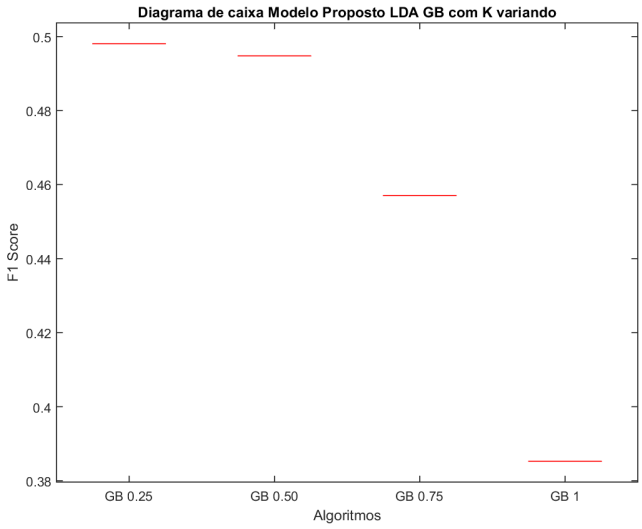

A Figura 46 é o diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto com a tarefa de regressão e com $K$ variando. O algoritmo KNN com $K$ igual a

0.25 apresentou melhor desempenho em relação à metrica *F1-score*.

### 5.2.6 Modelo Proposto com LDA

#### 5.2.6.1 Gradient Boosting

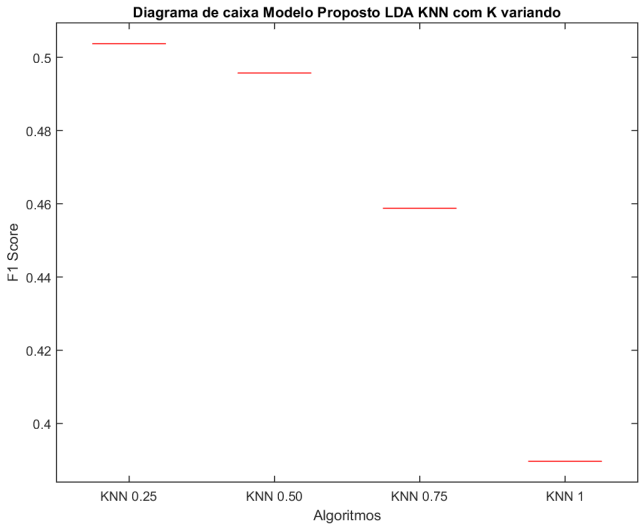
Figura 47 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto LDA com $K$ variando para o algoritmo GB.




A Figura 47 é o diagrama de caixa da execução do algoritmo GB utilizando o Modelo Proposto com a tarefa de regressão e com $K$ variando. Como pode-se observar, o melhor valor de $K$ foi $0.25$.

#### 5.2.6.2 KNN

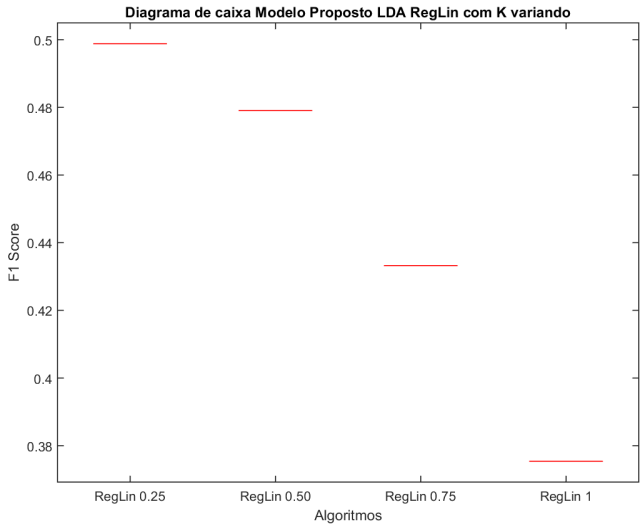
Figura 48 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto LDA com $K$ variando para o algoritmo KNN.



A Figura 48 é o diagrama de caixa da execução do algoritmo KNN utilizando o Modelo Proposto com a tarefa de regressão e com $K$ variando. Como pode-se observar, o melhor valor de $K$ foi $0.25$.

#### 5.2.6.3 Regressão Linear

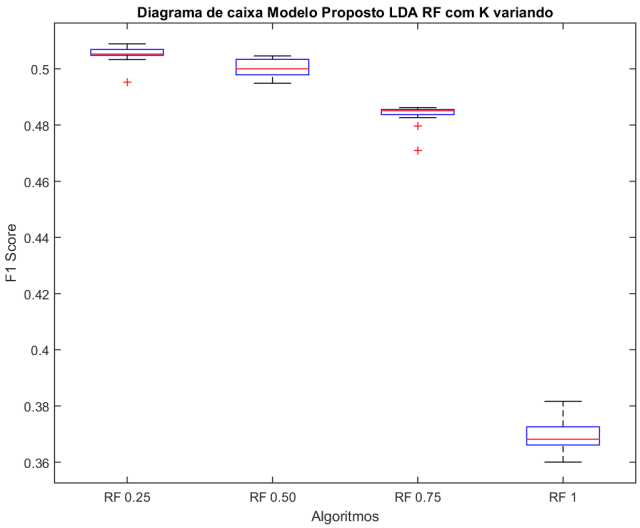
Figura 49 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto LDA com $K$ variando para o algoritmo Regressão Linear.




A Figura 49 é o diagrama de caixa da execução do algoritmo RegLin utilizando o Modelo Proposto com a tarefa de regressão e com $K$ variando. Como pode-se observar, o melhor valor de $K$ foi $0.25$.

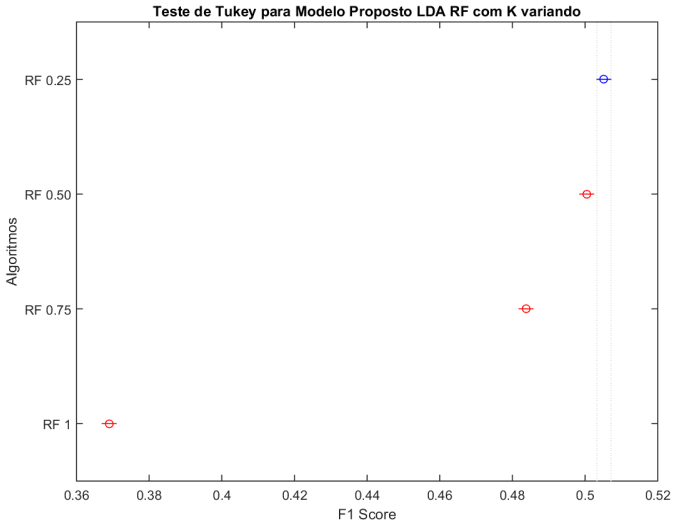
#### 5.2.6.4 Random Forest

Figura 50 – Diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto LDA com $K$ variando para o algoritmo RF.




A Figura 50 é o diagrama de caixa da execução do algoritmo RF utilizando o Modelo Proposto com a tarefa de regressão e com $K$ variando. Como pode-se observar, existe uma indecisão a respeito dos valores de $K$ iguais a $0.25$ e $0.75$.

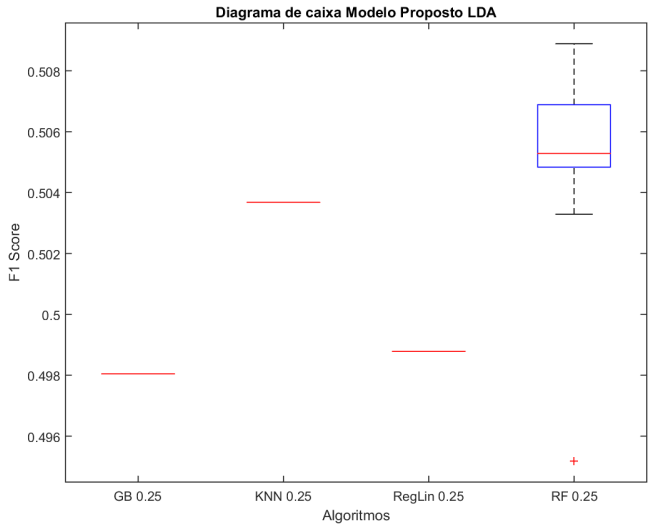
Figura 51 – Teste de Tukey para os algoritmos de RF no Modelo Proposto com LDA e $K$ variando na tarefa de regressão.




A Figura 51 demonstra que o o melhor valor de $K$ para o algoritmo RF com Modelo Proposto com LDA é de $0.25$

#### 5.2.6.5 Melhor Algoritmo com K variando

Figura 52 – Diagrama de caixa da execução dos melhores algoritmos utilizando o Modelo Proposto LDA com $K$ variando.




A Figura 52 é o diagrama de caixa da execução dos algoritmos utilizando o Modelo Proposto LDA com a tarefa de regressão e com $K$ variando. A avaliação gráfica é clara e demonstra que o algoritmo RF com $K$ igual a $0.25$ representa o melhor algoritmo para o Modelo Proposto com LDA .

### 5.2.7 Comparação de Algoritmos e Modelos

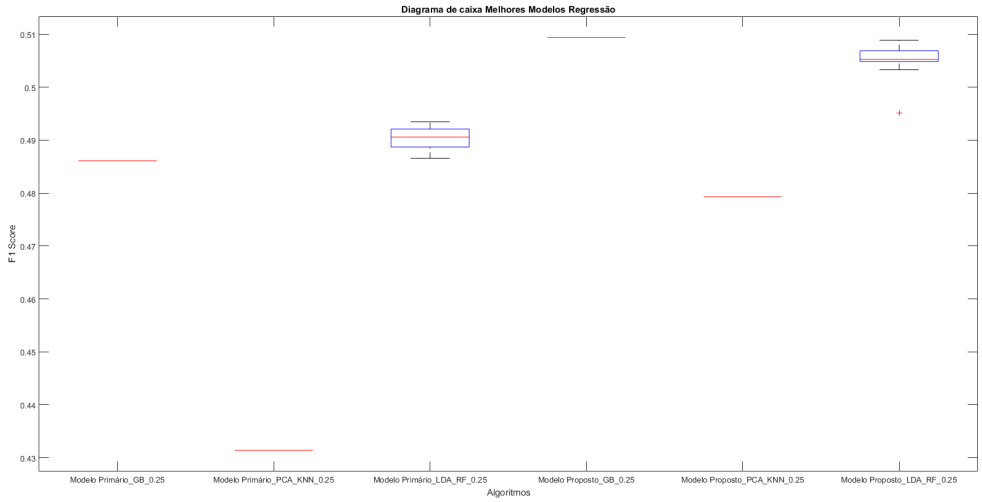
Para cada tipo de modelo, existe pelo menos um algoritmo que apresenta melhor desempenho. O Quadro 4 apresenta todos os melhores algoritmos para cada tipo de modelo.

Quadro 5 – Quadro com os melhores algoritmos para cada tipo de modelo e um código de identificação para o conjunto modelo algoritmo.

| Modelo | Algoritmo |
|---|---|
| Primário | GB 0.25 |
| Primário PCA | KNN 0.25 |
| Primário LDA | RF 0.25 |
| Proposto | GB 0.25 |
| Proposto PCA | KNN 0.25 |
| Proposto LDA | RF 0.25 |

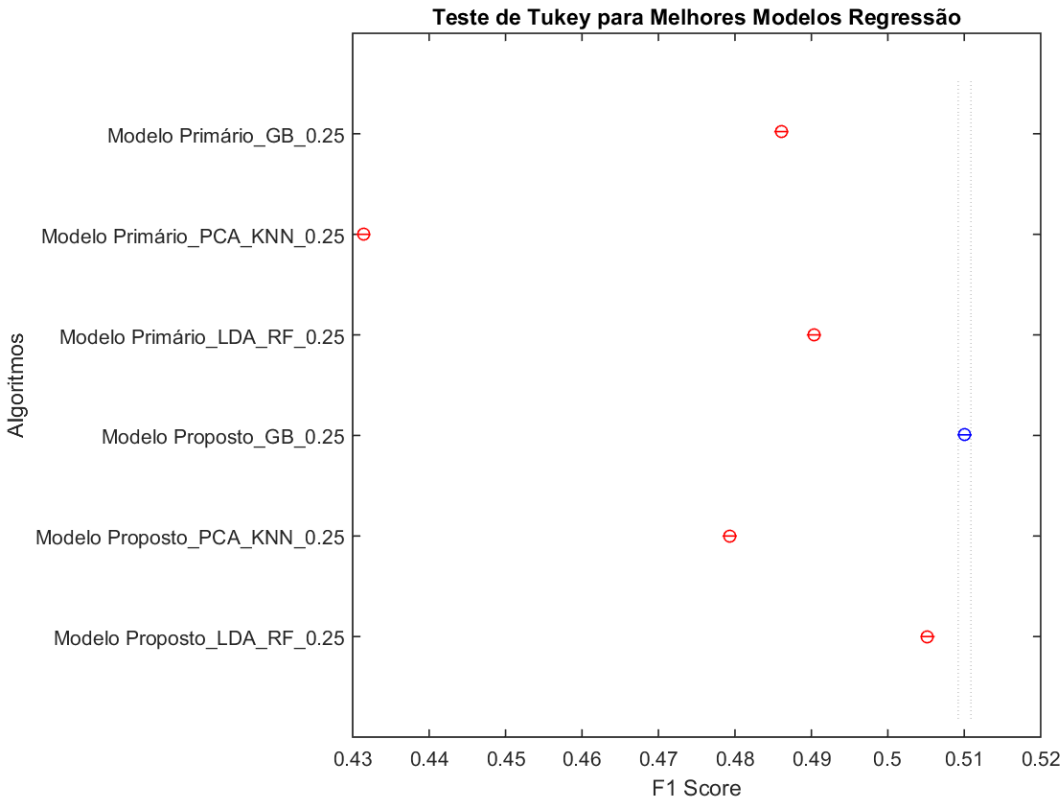
Primeiramente, a Figura 53 apresenta o diagrama de caixa do conjunto modelo algoritmo. Cada caixa representa o desempenho do melhor algoritmo submetido a um modelo de dados.

Figura 53 – Diagrama de caixa da execução dos algoritmos com melhor desempenho nos modelos de dados apresentados.




A Figura 53 não deixa claro o algoritmo que apresenta melhor desempenho, cuja tarefa é regressão, apresentados nessa seção. Portanto, um teste de Tukey foi realizado para verificar se existe uma diferença significativa entre os algoritmos.

Figura 54 – Teste de Tukey para os algoritmos de RF no Modelo Proposto com LDA e $K$ variando na tarefa de regressão.

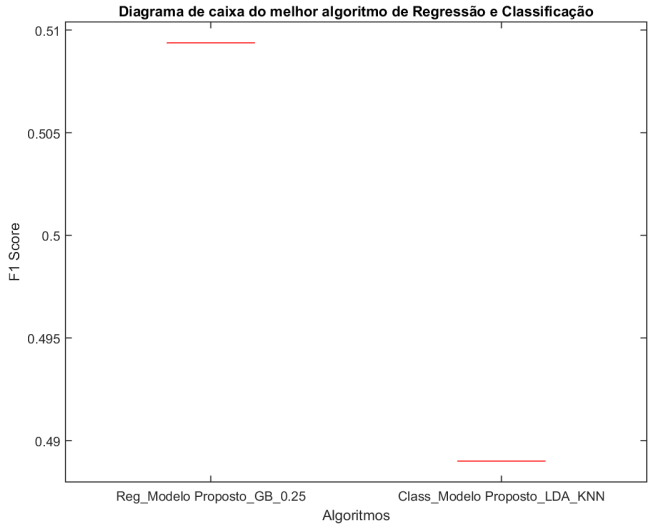


A Figura 54 demonstra que o o melhor algoritmo para regressão é o algoritmo GB que usa o Modelo Proposto como base de dados com um $K$ igual a $0.25$.

## 5.3 Classificação vs Regressão

Nas seções anteriores foi demonstrado o comportamento dos algoritmos nos modelos de dados distintos para ambas as tarefas: classificação e regressão. Analisando os valores da métrica *F1-score* obtidas tem-se uma noção de qual é a melhor tarefa para os preditores. Para facilitar, a Figura 55 apresenta, através de um diagrama de caixa, os melhores modelos para cada tarefa lado a lado para que uma análise possa ser feita.

Figura 55 – Diagrama de caixa da execução dos algoritmos com melhor desempenho nos modelos de dados apresentados.




Como pode-se observar, a tarefa de regressão apresentou melhor desempenho em relação a métrica *F1-score*, portanto o algoritmo GB aplicado a tarefa de regressão com $K$ igual a $0.25$ submetido ao Modelo Proposto será denominado agora Melhor Modelo Individual (MMI). Este modelo apresentou uma métrica *F1-score* de $0.509$.

Pode-se também observar que os algoritmos de regressão, GB e RF, com $K$ variando apresentaram um comportamento melhor em relação a métrica *F1-score* do que os modelos com tarefa de classificação direta.

# 6 Análise e Discussão dos Resultados

Como já foi explanado anteriormente neste trabalho na seção de Capítulo 2 a tarefa de predição em jogos de futebol é muito árdua. Para obter uma melhor noção de desempenho dos modelos em relação a qualidade das predições, utilizou-se a métrica *F1-score*. Entretanto, como o objetivo final da utilização dos modelos é classificar qual o resultado do jogo entre vitória do mandante, empate ou vitória do visitante a métrica acurácia é de mais fácil entendimento e facilita mensurar possíveis aplicabilidades. Ela não foi utilizada no processo de treinamento do modelo pois o problema é muito suscetível a aleatoriedade e acurácia não nos diz sobre a qualidade da predição em relação a relevância da predição.

Partindo de modelos classificadores simples podemos ter um que sempre irá predizer a classe de maior ocorrência nos dados treino. Como demonstrado na Tabela 1, este modelo sempre irá predizer vitória do mandante. Só este fato nos daria uma acurácia de $45.59\%$ e uma métrica *F1-score* de $0.286$, utilizando os dados de teste.

Um modelo, doravante denominado Modelo Ingênuo, que sempre classifica o resultado levando em consideração o favorito a vencer a partida, sendo o favorito definido pelo premio pago por aposta da Bet365, apresenta uma acurácia de $54.16\%$ e uma métrica *F1-score* $0.463$.

Levando em consideração humanos, encontra-se na literatura poucos registros de medição de performance na classificação de uma partida de futebol. Pettersson e Nyquist (2017) fizeram dois levantamentos acerca da acurácia dos humanos ao predizer resultados de jogos de futebol. O primeiro diz respeito a uma competição entre jornais suecos. Eles fazem predições sobre os jogos do campeonato sueco, rodada a rodada. A Tabela 2 feita por Pettersson e Nyquist (2017) demonstra a acurácia para o ano de 2015.

Pettersson e Nyquist (2017) também fizeram um levantamento sobre a acurácia de usuários no aplicativo *Forza Football*. Este permite que usuários façam predições sobre o resultado do jogo. O Quadro 6 demonstra à taxa de acerto de fãs de futebol em jogos entre 01/06/2015 e 01/04/2017, levando-se em consideração a quantidade de predições feitas em cada jogo.

Os últimos dados apresentados nos parágrafos anteriores apresentam melhor o contexto de avaliação de métrica para o problema de predição de jogos de futebol.

O modelo MMI (Melhor Modelo Individual) apresentou uma acurácia de $52.78\%$ nos dados de teste e de $52.00\%$ nos dados de avaliação. Comparando-se com a métrica *F1-score* do MMI, $0.509$ , em relação ao Modelo Ingênuo observa-se uma melhora de $9.99\%$.

Tabela 2 – Competidores no desafio dos 10 jornais suecos em 2015 e suas respectivas acurácias.

| Jornal | Acurácia |
|---|---|
| Sydsvenska dagladet | 46.75% |
| Expressen | 46.60% |
| Aftonbladet | 46.45% |
| Dala-Demokratin | 46.45% |
| Borlänge Tidning | 45.86% |
| Vi Tippa | 45.86% |
| TT | 45.71% |
| Dagens Spel | 45.56% |
| Skånska Dagbladet | 45.12% |
| Dagens Nyheter | 43.05% |

Fonte: Traduzido de Pettersson e Nyquist (2017)

Quadro 6 – Acurácia dos usuários do *Forza Football*

| Qnt. Usuários | Total de Jogos | Acurácia |
|---|---|---|
| Qualquer | 83963 | 47.49% |
| >100 | 40916 | 50.92% |
| >500 | 17490 | 52.57% |
| >1000 | 10520 | 54.34% |
| >5000 | 2858 | 59.13% |
| >10000 | 1215 | 59.50% |
| >20000 | 317 | 55.21% |

Fonte: Traduzido e adaptado de Pettersson e Nyquist (2017)

Portanto, o MMI melhora um modelo baseado em especialistas das casas de apostas numa métrica que leva em consideração o desbalanceamento de classes e a qualidade da predição, perdendo somente $2{,}16\%$ em acurácia. Troca-se acurácia por melhor distribuição dos acertos entre as classes.

## 6.1 Rentabilidade

Um estudo que também pode ser feito com MMI é se este modelo é lucrativo, quando aplicado a apostas esportivas. Para tal, utilizou-se os dados de teste para estudar o comportamento do modelo.

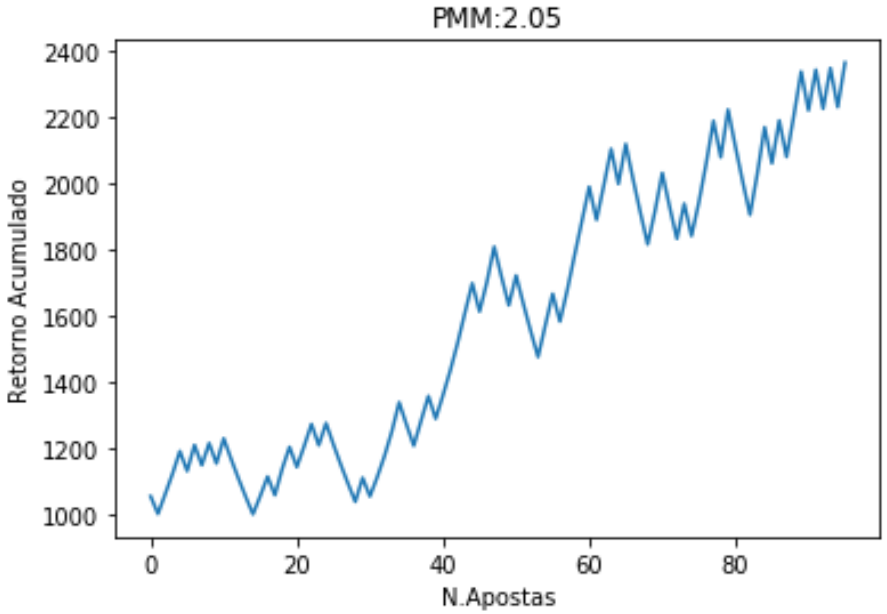
Neste estudo separou-se os jogos em que o MMI classificou como vitória do mandante, empate e vitória do visitante. Para cada classe de classificação, foi medido a rentabilidade do modelo aplicada nas apostas esportivas, fixando um capital inicial de R$1000,00 reais, arriscando sempre 5% do capital atual na aposta. Para definir um critério de quando apostar ou não, variou-se o valor do Prêmio Pago Mínimo (PPM) de entrada entre 1.1 e 2.1,

variando de 0.05. Exemplificando um cenário em que o PPM esteja em 1.85 mas o prêmio pago pela casa de aposta seja menor, a aposta não é feita. Caso o premio seja superior ao PPM a aposta é efetuada.

Para a classe de jogos em que o MMI classificou como vitória do mandante, variando o valor do PPM de entrada, em nenhum cenário se tornou lucrativo. O mesmo foi verdade para a classe de empate. Já para a classe de vitória do visitante obteve-se um cenário interessante.

Para a vitória do visitante, na maioria dos valores do PPM, o MMI se mostrou lucrativo. Entretanto o melhor valor para PPM, que maximizou o lucro, foi $2.05$. A Figura 56 apresenta o gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de $2.05$. Nele foram feitas um total de 96 apostas que retornaram um montante final total de $R\$2361.80$, resultando num lucro sobre o capital inicial de $136.18\%$. Os demais gráficos correspondentes aos outros PPM se encontram no Apêndice A.

Figura 56 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 2.05 nos dados de Teste

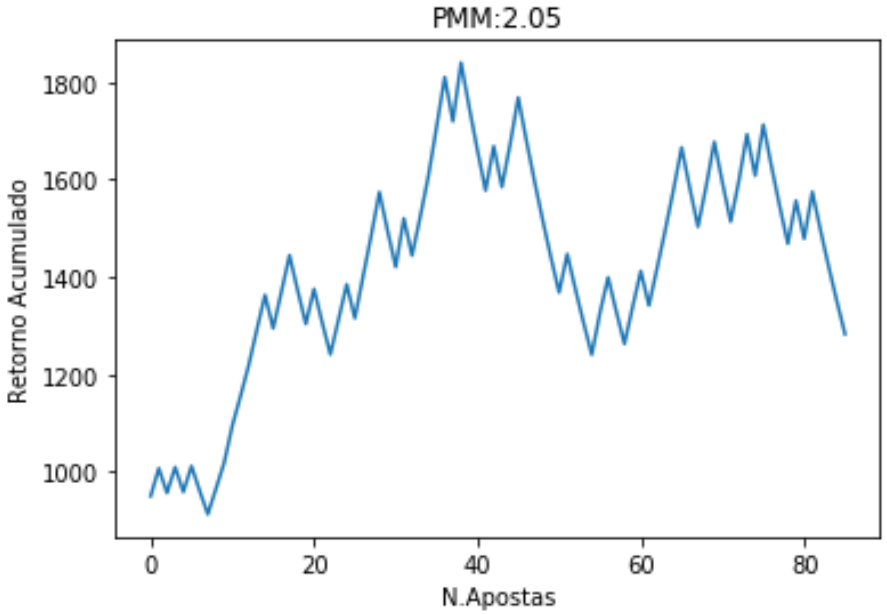



Um retorno sobre o capital nesse nível em um período de tempo de 1 ano é algo muito interessante. Portanto, testou-se o MMI com os dados de avaliação, que o modelo nunca os conheceu anteriormente.

A Figura 57 demonstra o comportamento do MMI com PPM de $2.05$ quando aplicado nos dados de avaliação. Ao final do período o montante resultante foi de $R\$1282.72$, resultando um lucro sobre o capital de $28{,}8\%$.

Nos dados de avaliação o retorno percentual caiu significativamente. Entretanto, ainda assim o MMI se mostrou com uma boa lucratividade. A Tabela 3 demonstra um

Figura 57 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 2.05 para os dados de Avaliação




comparativo de rentabilidade da taxa SELIC, do CDI e do IBOV nos mesmos períodos em que o MMI foi avaliado: 01 de agosto de 2015 até 31 de julho de 2016.

Tabela 3 – Ativos e suas taxas de rentabilidade no período de 01/08/2015 até 31/07/2016

| Ativo | Taxa |
|---|---|
| IBOV | 12.67% |
| CDI | 14.01% |
| SELIC | 14.03% |
| MMI | 28.8% |

Fonte: Taxas SELIC e CDI extraídas de Brasil (2018). Taxa IBOV extraída de Investing.com (2018)

Mesmo com a rentabilidade do MMI caindo nos dados de avaliação, o retorno obtido foi maior que as principais taxas aplicadas no mercado no mesmo período. Com relação a grande diferença de rentabilidade entre o teste e o dado de avaliação, a primeira coisa que deve-se levar em consideração é que rentabilidade passada não garante rentabilidade futura. Sabendo disso, olha-se os dados referentes ao MMI com PPM de $2.05$ nos dados de teste e de avaliação. Nos dados de teste o MMI obteve uma acurácia de $55.2\%$ e um prêmio pago médio em suas apostas de $2.19$ num universo de 96 apostas. Já para o dado de avaliação o MMI obteve uma acurácia de $50.0\%$ com prêmio pago médio de $2.18$ num universo de 89 apostas.

A queda vista na acurácia, uma variação de $5\%$, é uma variação dentro de um intervalo razoável de possibilidade. Na Seção A.2 estão todos os gráficos do MMI com o PPM variando nos mesmos moldes que foram utilizados para definir o valor de PPM que

maximiza o lucro nos dados de teste, aplicados aos dados de avaliação. Neles consegue-se observar que somente quando o valor do PPM é de $2.05$ o modelo é lucrativo. Isso aumenta a confiança de que o MMI com PPM de $2.05$ é um modelo rentável e de certa forma robusto.

# 7 Conclusão

Este trabalho investigou a tarefa de predição de jogos de futebol fazendo um levantamento do estado da arte. Levantado os algoritmos mais comuns, este trabalho propôs um modelo de dados primário para a tarefa de predição de jogos futebol. Também abordou o problema de predição de duas formas: classificação e regressão. Para a tarefa de classificação foi utilizado LogReg, KNN, RF, GB e NB. Já para a tarefa de regressão foi utilizado Regressão Linear, KNN, RF e GB. Estes modelos foram aplicados às variações dos modelos de dados para cada uma das tarefas designadas: classificação e regressão.

Foi mostrado que é possível utilizar técnicas de Aprendizado de Máquinas no futebol com uma abordagem prática e útil, uma vez que o modelo MMI demonstrou ser lucrativo tanto na base de testes quanto na de avaliação.

Devido a falta de trabalhos similares na literatura, os resultados apresentados neste trabalho poderão servir como um incentivo para o desenvolvimento de futuras pesquisas. Além disso, foi disponibilizado um modelo de dados gratuito e acessível, e um estudo de performance de algumas técnicas de Aprendizado de Máquinas, facilitando assim, a aplicação e comparação de novos modelos e novos algoritmos.

Além disso, estabeleceu-se uma relação entre a taxa de acerto dos seres humanos e o desempenho dos algoritmos. Olhando por esse prisma, foi possível verificar que os algoritmos não ficam pra trás, tendo taxas de acerto no mesmo patamar de assertividade. Um estudo comparativo sobre a assertividade de técnicas de Aprendizado de Máquina e seres humanos pode ser feito no futuro.

Ao fim do trabalho pode-se concluir:

- O melhor modelo de dados com seu respectivo algoritmo foi o *Gradient Boosting* aplicado a tarefa de regressão com $K$ igual a 0.25 no Modelo Proposto. Este modelo que fora denominado MMI (Melhor Modelo Individual), apresentou métrica *F1-score* de 0.509 e uma acurácia de 52.78%;
- O MMI se mostrou robusto em relação à lucratividade quando aplicado em apostas esportivas, sempre que ele prediz vitória do visitante e o prêmio pago pela aposta pelas casas de apostas é superior a 2.05;
- A tarefa de regressão demonstrou ser mais adequada à predição de jogos de futebol, quando se leva em consideração a métrica *F1-score*;
- Para os modelos sem utilização de técnicas de redução de dimensão os algoritmos derivados do *ensemble* com árvore de decisão se sobressaíram, tanto para classificação quanto regressão;
- Para os modelos de dados com redução de dimensão, para a técnica PCA o algoritmo

KNN foi o melhor considerando tanto o Modelo Primário quanto o Modelo Proposto, para as duas tarefas.

## 7.1 Trabalhos Futuros

Como trabalhos futuros a partir deste trabalho pode-se:

- Otimizar o processo de lucratividade do MMI. Nele estipulou-se que sempre fosse apostado 5% do capital total em cada aposta executada. Um trabalho que otimize a gestão de capital é interessante;
- Aumentar a sensibilidade do valor de $K$ nos modelos cuja tarefa é de regressão. Neste trabalho variou-se entre 0, 0.25, 0.5, 0.75 e 1. Como visto no Quadro 5 todos os valores de $K$ para os melhores foram de 0.25. Então, como trabalho futuro, aumentar a sensibilidade e variar o valor de $K$ entre 0 e 0.25;
- Coletar mais dados e criar outros modelos de Aprendizado de Máquina que utilizem também a saída do MMI para realizar predições;
- Com mais dados, pode-se também incluir um estudo do comportamento de Redes Neurais Artificiais nos modelos propostos;
- Construir modelos que tenham como objetivo a predição do números de gols numa partida de futebol, pois quanto mais modelos que gerariam características sobre uma partida mais fácil num futuro vai ser a tarefa de predição de jogos de futebol.
- Um estudo comparativo sobre a assertividade do modelo baseado em aprendizado de máquinas e especialistas em futebol.

# Referências


ANDERSON, C.; SALLY, D. **The numbers game: why everything you know about football is wrong**. [S.l.]: Penguin UK, 2013. Citado na página 2.

AOKI, R.; ASSUNCAO, R. M.; MELO, P. O. de. Luck is hard to beat: The difficulty of sports prediction. **arXiv preprint arXiv:1706.02447**, 2017. Citado 2 vezes nas páginas 5 e 6.

ASSOCIATION, F. I. de F. et al. Fifa big count 2006: 270 million people active in football. **Retrieved February**, v. 20, p. 2012, 2007. Citado na página 1.

BELHUMEUR, P. N.; HESPANHA, J. P.; KRIEGMAN, D. J. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. **IEEE Transactions on pattern analysis and machine intelligence**, IEEE, v. 19, n. 7, p. 711–720, 1997. Citado na página 21.

BELLMAN, R. **Dynamic programming**. [S.l.]: Courier Corporation, 2013. Citado na página 18.

BISHOP, C. M. **Pattern recognition and machine learning (information science and statistics) springer-verlag new york**. [S.l.]: Springer, 2006. Citado 2 vezes nas páginas 15 e 16.

BRASIL, B. C. do. **BCB Calculadora do Cidadao**. 2018. Disponível em: <https://www3.bcb.gov.br/CALCIDADAO/publico/exibirFormCorrecaoValores.do?method=exibirFormCorrecaoValores&aba=5>. Citado na página 62.

BROMBERGER, C. **Significación de la pasión popular por los clubes de fútbol**. [S.l.]: Libros del Rojas, 2001. Citado na página 1.

CHENG, T. et al. A new model to forecast the results of matches based on hybrid neural networks in the soccer rating system. In: IEEE. **Computational Intelligence and Multimedia Applications, 2003. ICCIMA 2003. Proceedings. Fifth International Conference on**. [S.l.], 2003. p. 308–313. Citado 2 vezes nas páginas viii e 6.

DHAR, V. **What Is the Role of Artificial Intelligence in Sports?** [S.l.]: Mary Ann Liebert, Inc. 140 Huguenot Street, 3rd Floor New Rochelle, NY 10801 USA, 2017. Citado na página 2.

DIETTERICH, T. G. et al. Ensemble methods in machine learning. **Multiple classifier systems**, Springer, v. 1857, p. 1–15, 2000. Citado na página 14.

DINIZ, M. A. et al. Comparing probabilistic predictive models applied to football. **arXiv preprint arXiv:1705.04356**, 2017. Citado na página 7.

DOMINGOS, P. A few useful things to know about machine learning. **Communications of the ACM**, 2012. Citado na página 2.

DUDA, R. O.; HART, P. E.; STORK, D. G. **Pattern classification**. [S.l.]: John Wiley & Sons, 2012. Citado na página 11.

ELO, A. E. **The rating of chessplayers, past and present**. [S.l.]: Arco Pub., 1978. Citado na página 6.

INVESTING.COM. **Índice Bovespa Histórico de Taxas - Investing.com**. 2018. Disponível em: <https://br.investing.com/indices/bovespa-historical-data>. Citado na página 62.

JAMES, G. et al. **An introduction to statistical learning**. [S.l.]: Springer, 2013. v. 112. Citado 5 vezes nas páginas 10, 12, 13, 14 e 15.

KAGGLE. **European Soccer Database| Kaggle**. 2016. Disponível em: <https://www.kaggle.com/hugomathien/soccer>. Citado na página 24.

KUMAR, G. Machine learning for soccer analytics. **University of Leuven**, 2013. Citado na página 7.

KUPER, S. **Soccernomics: Why England Loses, Why Spain, Germany, and Brazil Win, and Why the US, Japan, Australia and Even Iraq Are Destined to Become the Kings of the World's Most Popular Sport**. [S.l.]: Nation Books, 2014. Citado na página 2.

MONTGOMERY, D. C.; RUNGER, G. C.; CALADO, V. **Estatística Aplicada E Probabilidade Para Engenheiros .** [S.l.]: Grupo Gen-LTC, 2000. Citado na página 10.

MURPHY, K. P. Naive bayes classifiers. **University of British Columbia**, 2006. Citado na página 11.

OPTA. **Opta NEWS Opta acquired by PERFORM Group**. 2013. Disponível em: <http://www.optasports.com/news-area/news-opta-acquired-by-perform-group.aspx>. Citado na página 1.

OWEN, A. Dynamic bayesian forecasting models of football match outcomes with estimation of the evolution variance parameter. **IMA Journal of Management Mathematics**, Oxford University Press, v. 22, n. 2, p. 99–113, 2011. Citado na página 7.

PEARSON, K. 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**, Taylor & Francis, 1901. Citado na página 19.

PETTERSSON, D.; NYQUIST, R. Football match prediction using deep learning. 2017. Citado 2 vezes nas páginas 59 e 60.

POWER, P. et al. Not all passes are created equal: Objectively measuring the risk and reward of passes in soccer from tracking data. In: ACM. **Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. [S.l.], 2017. p. 1605–1613. Citado na página 2.

RENNIE, J. D. et al. Tackling the poor assumptions of naive bayes text classifiers. In: **Proceedings of the 20th international conference on machine learning (ICML-03)**. [S.l.: s.n.], 2003. p. 616–623. Citado na página 11.

RIJSBERGEN, C. van. Chapter 7. **Information Retrieval**, p. 178–180, 1979. Citado na página 17.

SAMMUT, C.; WEBB, G. I. **Encyclopedia of machine learning**. [S.l.]: Springer Science & Business Media, 2011. Citado na página 18.

ULMER, B.; FERNANDEZ, M.; PETERSON, M. **Predicting Soccer Match Results in the English Premier League**. [S.l.]: Stanford University, 2013. Citado na página 7.

WIEDERHOLD, G.; MCCARTHY, J.; FEIGENBAUM, E. Arthur samuel: pioneer in machine learning. **Communications of the ACM**, Association for Computing Machinery, Inc., v. 33, n. 11, p. 137–139, 1990. Citado na página 8.

YU, H.; YANG, J. A direct lda algorithm for high-dimensional data—with application to face recognition. **Pattern recognition**, Pergamon, v. 34, n. 10, p. 2067–2070, 2001. Citado na página 21.

# Apêndices

# APÊNDICE A – Gráficos de rentabilidade do MMI classificando vitória do visitante

## A.1 Gráficos usando dados de Teste

Nesta seção se encontra os dados de rentabilidade provenientes dos experimentos de rentabilidade do MMI quando classifica vitória do visitante variando o valor do PPM para os dados de Teste.

Figura 58 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.1 para os dados de Teste

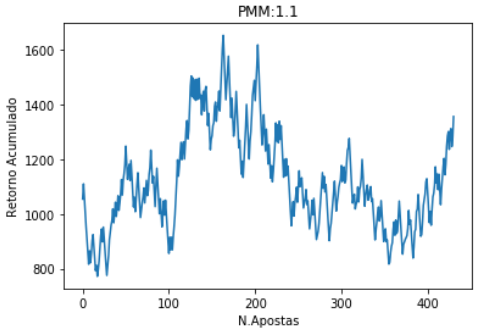



Figura 59 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.15 para os dados de Teste

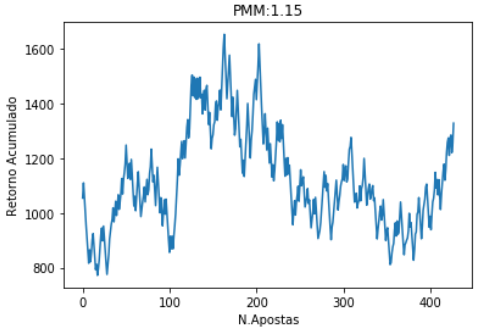


Figura 60 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.2 para os dados de Teste

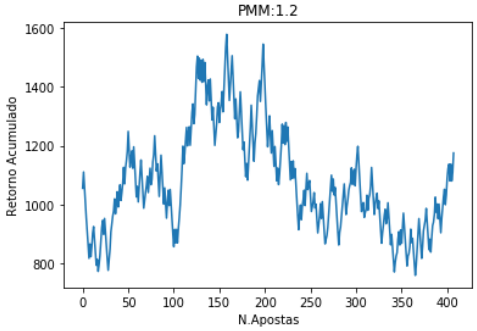



Figura 61 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.25 para os dados de Teste

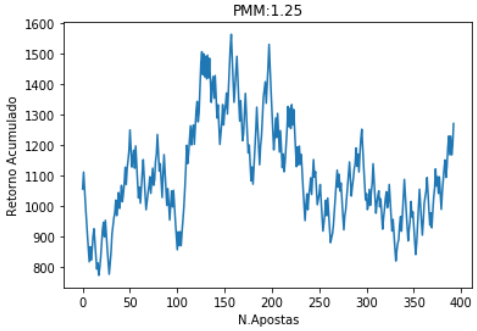



Figura 62 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.3 para os dados de Teste

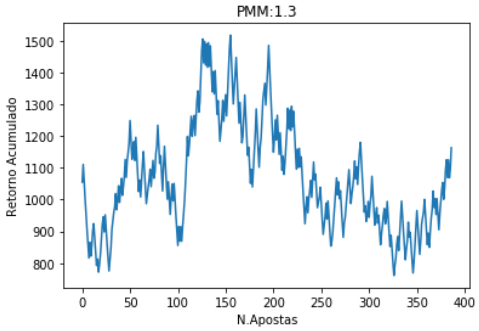


Figura 63 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.35 para os dados de Teste

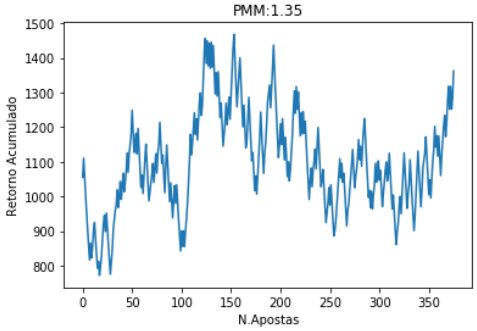



Figura 64 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.4 para os dados de Teste

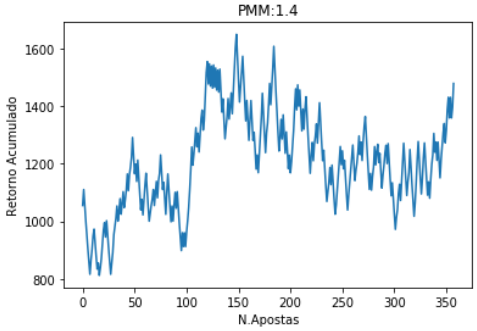



Figura 65 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.45 para os dados de Teste

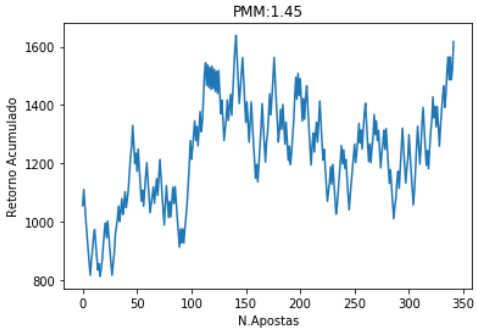


Figura 66 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.5 para os dados de Teste

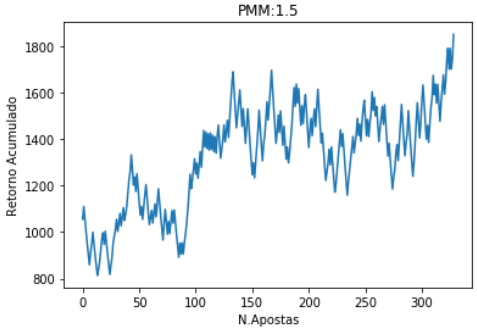



Figura 67 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.55 para os dados de Teste

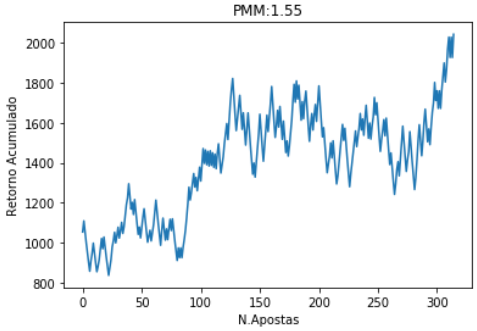



Figura 68 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.6 para os dados de Teste

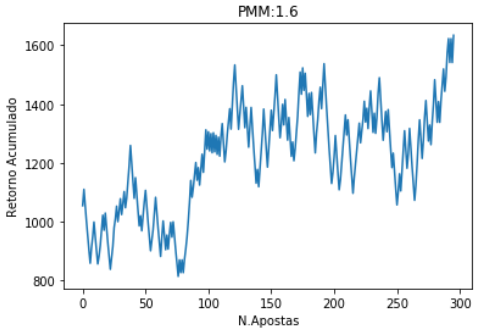


Figura 69 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.65 para os dados de Teste

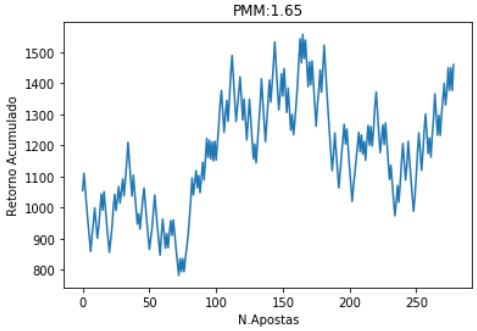



Figura 70 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.7 para os dados de Teste

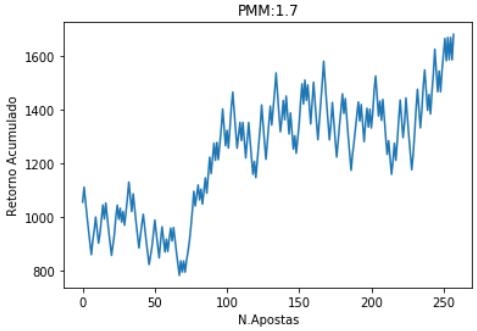



Figura 71 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.75 para os dados de Teste

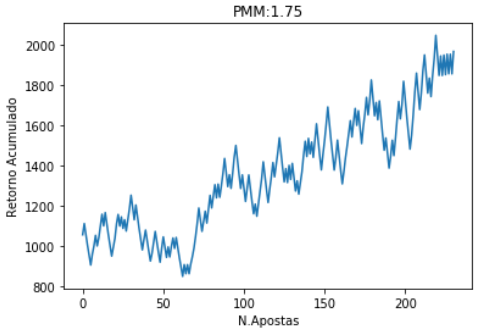


Figura 72 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.8 para os dados de Teste

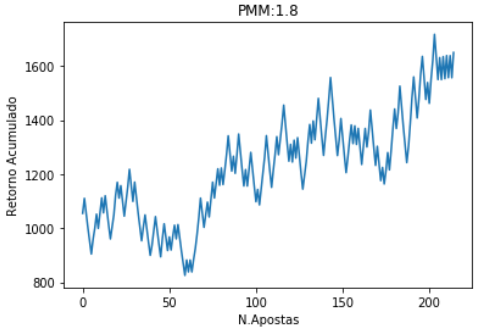



Figura 73 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.85 para os dados de Teste

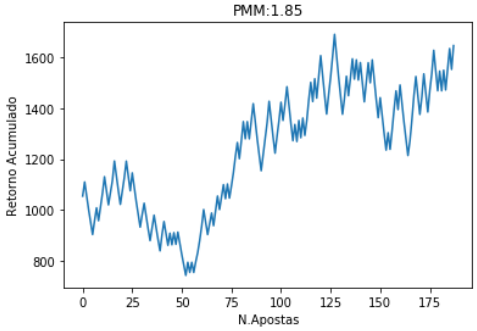



Figura 74 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.9 para os dados de Teste

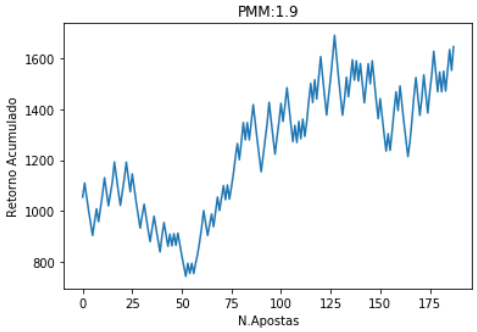


Figura 75 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.95 para os dados de Teste

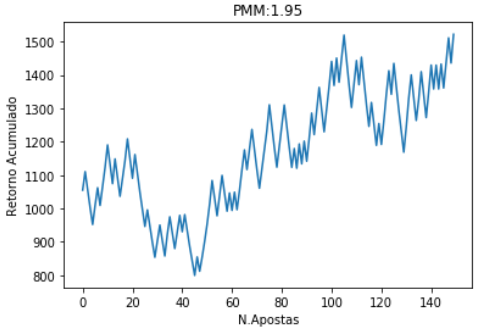



Figura 76 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 2.00 para os dados de Teste

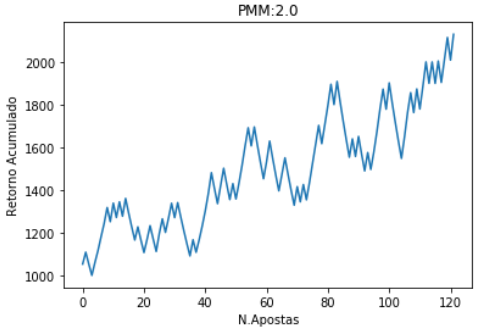



Figura 77 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 2.05 para os dados de Teste

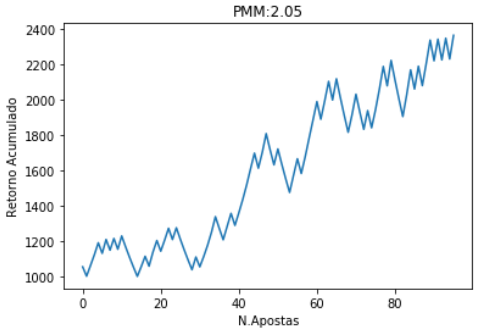
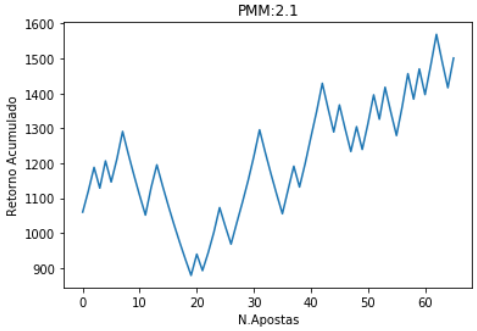


Figura 78 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 2.1 para os dados de Teste




## A.2 Gráficos usando dados de Avaliação

Nesta seção se encontra os dados de rentabilidade provenientes dos experimentos de rentabilidade do MMI quando classifica vitória do visitante variando o valor do PPM para os dados de Avaliação.

Figura 79 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.1 para os dados de Avaliação

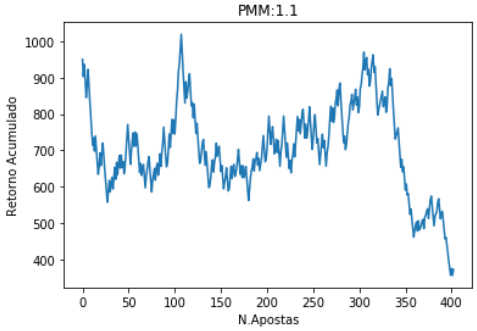



Figura 80 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.15 para os dados de Avaliação

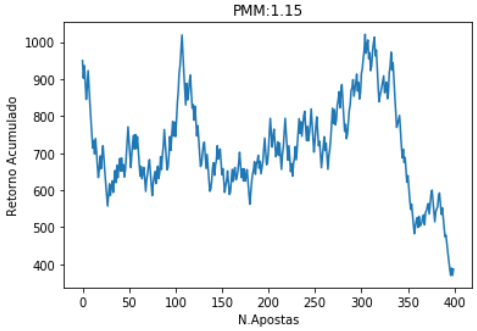


Figura 81 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.2 para os dados de Avaliação

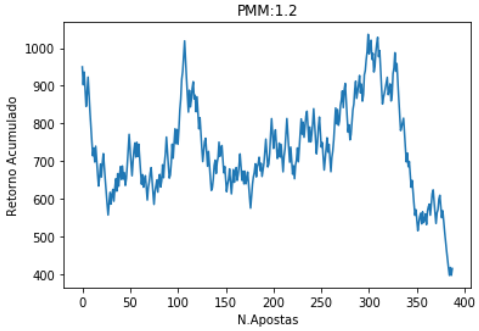



Figura 82 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.25 para os dados de Avaliação

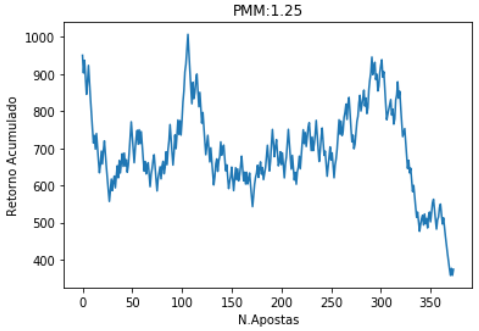



Figura 83 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.3 para os dados de Avaliação

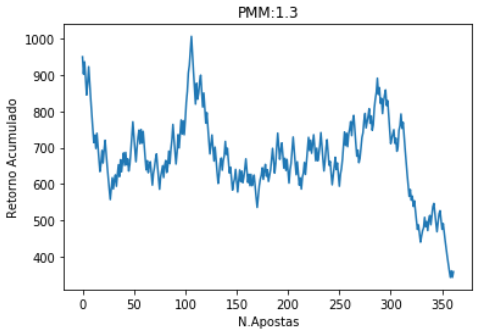


Figura 84 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.35 para os dados de Avaliação

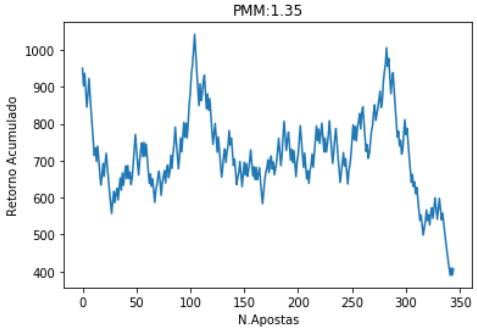



Figura 85 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.4 para os dados de Avaliação

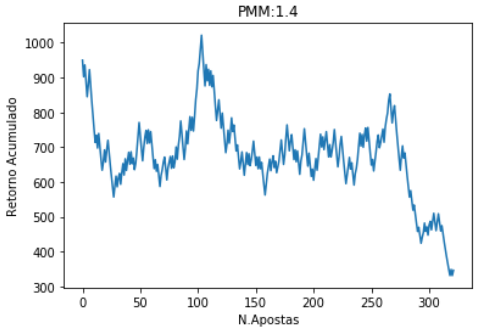



Figura 86 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.45 para os dados de Avaliação

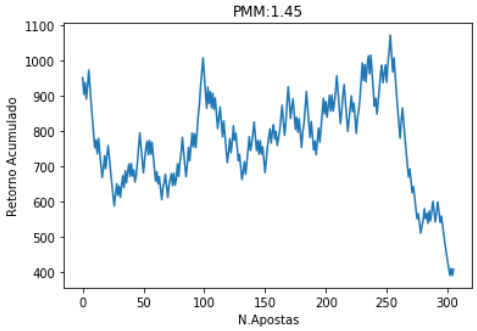


Figura 87 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.5 para os dados de Avaliação

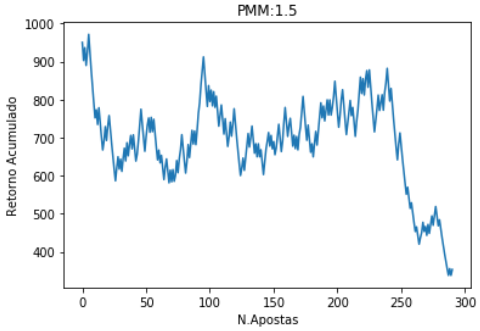



Figura 88 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.55 para os dados de Avaliação

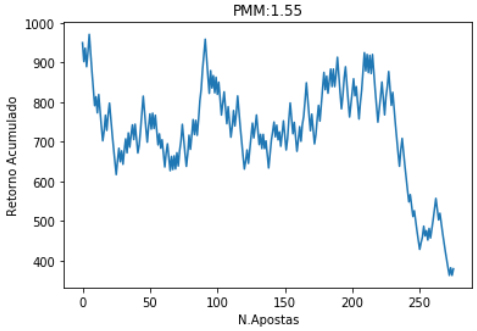



Figura 89 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.6 para os dados de Avaliação

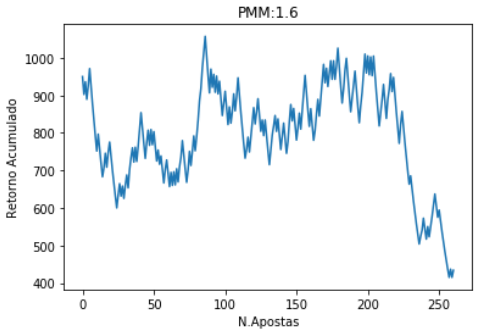


Figura 90 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.65 para os dados de Avaliação

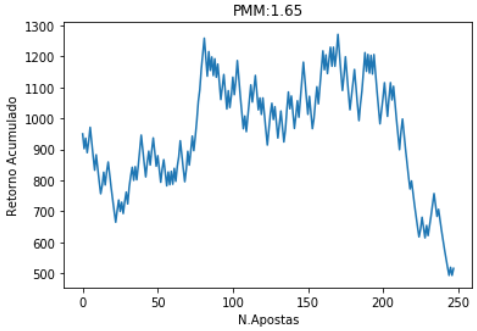



Figura 91 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.7 para os dados de Avaliação

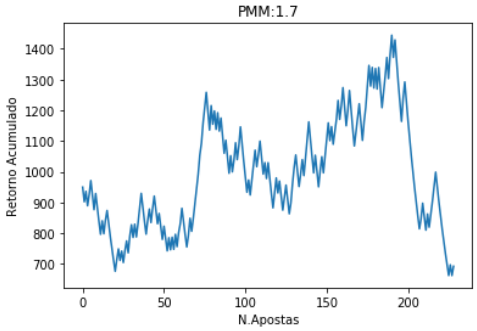



Figura 92 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.75 para os dados de Avaliação

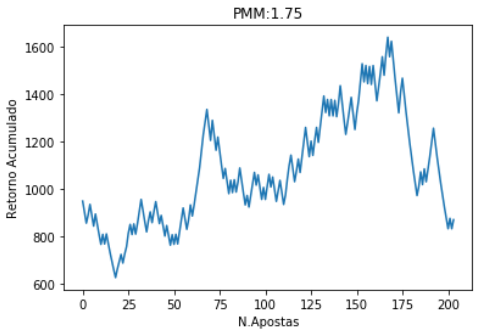


Figura 93 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.8 para os dados de Avaliação

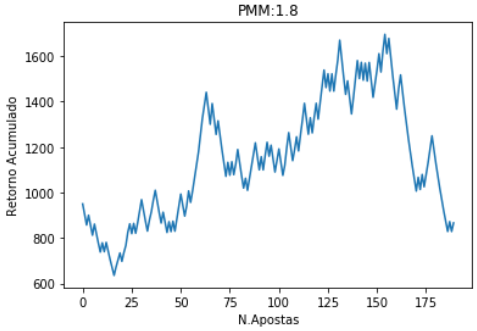



Figura 94 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.85 para os dados de Avaliação

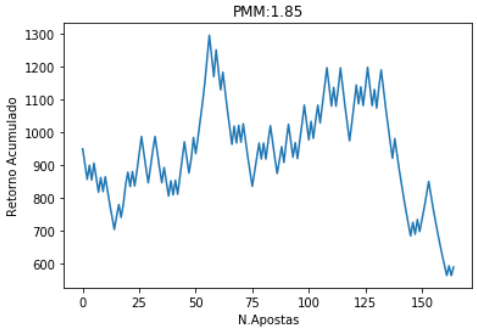



Figura 95 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.9 para os dados de Avaliação

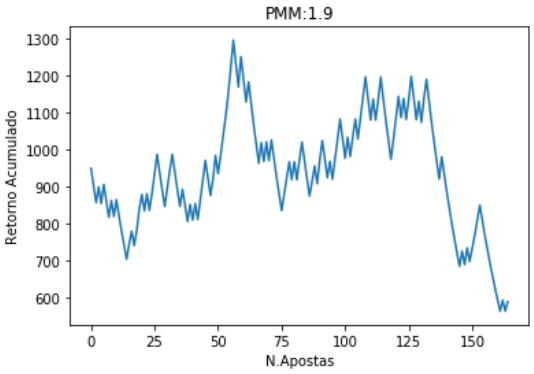


Figura 96 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 1.95 para os dados de Avaliação

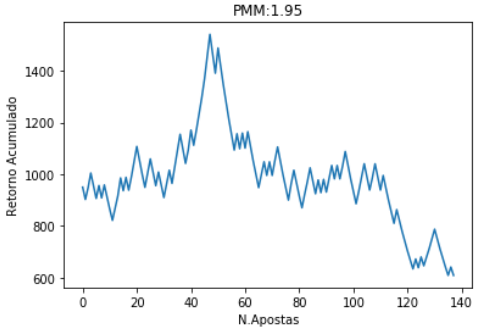



Figura 97 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 2.00 para os dados de Avaliação

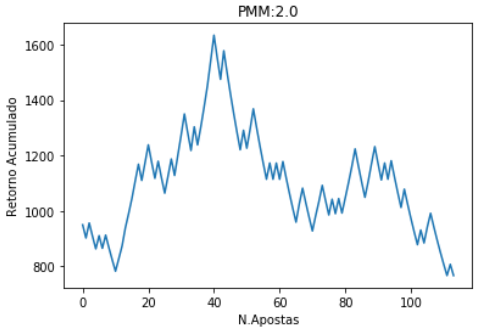



Figura 98 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 2.05 para os dados de Avaliação

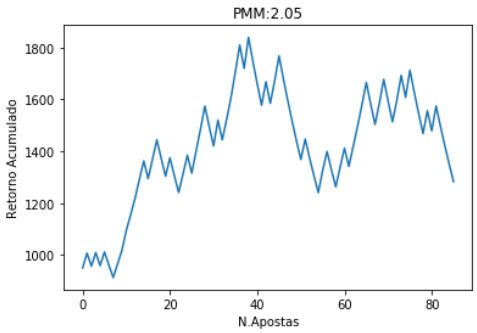


Figura 99 – Gráfico de rentabilidade do MMI quando classifica vitória do visitante com PPM de 2.1 para os dados de Avaliação