Randal Gasparini

Análise Posicional de Jogadores Brasileiros de Futebol Utilizando Dados GPS

Sorocaba, SP

26 de Fevereiro de 2018

Randal Gasparini

Análise Posicional de Jogadores Brasileiros de Futebol Utilizando Dados GPS

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação (PPGCC-So) da Universidade Federal de São Carlos como parte dos requisitos exigidos para a obtenção do título de Mestre em Ciência da Computação. Linha de pesquisa: Engenharia de Software e Redes de Computadores.

Universidade Federal de São Carlos – UFSCar

Centro de Ciências em Gestão e Tecnologia – CCGT

Programa de Pós-Graduação em Ciência da Computação – PPGCC-So

Orientador: Prof. Dr. Alexandre Álvaro

Sorocaba, SP

26 de Fevereiro de 2018

Gasparini, Randal

Análise Posicional de Jogadores Brasileiros de Futebol Utilizando Dados GPS / Randal Gasparini. -- 2018.
98 f. : 30 cm.

Dissertação (mestrado)-Universidade Federal de São Carlos, campus Sorocaba, Sorocaba
Orientador: Prof. Dr. Alexandre Álvaro
Banca examinadora: Prof. Dr. André Luis Debiaso Rossi, Profa. Dra. Katti Faceli
Bibliografia

1. Aprendizado de Máquina. 2. Futebol. 3. Classificação. I. Orientador. II. Universidade Federal de São Carlos. III. Título.

Ficha catalográfica elaborada pelo Programa de Geração Automática da Secretaria Geral de Informática (SIn).
DADOS FORNECIDOS PELO(A) AUTOR(A)
Bibliotecário(a) Responsável: Maria Aparecida de Lourdes Mariano – CRB/8 6979

UNIVERSIDADE FEDERAL DE SÃO CARLOS

Centro de Ciências em Gestão e Tecnologia
Programa de Pós-Graduação em Ciência da Computação

Folha de Aprovação

Assinaturas dos membros da comissão examinadora que avaliou e aprovou a Defesa de Dissertação de Mestrado do candidato Randal Gasparini, realizada em 26/02/2018:

Prof. Dr. Alexandre Alvaro
UFSCar

André L. D. Rossi

Prof. Dr. André Luis Debiaso Rossi
UNESP

Katti Faceli

Profa. Dra. Katti Faceli
UFSCar

Dedico esse trabalho aos meus pais por me ensinarem o valor da educação. À minha esposa e ao meu filho pelo apoio e compreensão durante essa importante jornada em minha vida.

Agradecimentos

Agradeço,

a Deus pelo dom da vida,

ao meu orientador pela confiança em mim depositada,

aos professores que trilharam essa jornada comigo,

ao CCGT pela estrutura de estudos oferecida,

à One Sports pela parceria e disponibilização dos dados.

Eu não falhei, encontrei 10 mil soluções que não davam certo.
(Thomas Alva Edison, inventor)

Resumo

O futebol profissional vem se transformando ao longo do tempo e busca constantemente ferramentas e dados que auxiliem a tomada de decisão, entregando informações táticas e técnicas para o time. Não diferente, a modalidade esportiva no Brasil segue a mesma tendência e os investimentos são cada vez mais consideráveis. A exemplo disso, a empresa One Sports é responsável pela captura de dados GPS de jogadores que atuam profissionalmente em determinados clubes nacionais. Uma vez que a coleta existe e a mesma é rica em atributos, esse estudo aborda a possibilidade de inferir a posição tática ideal de um jogador profissional de futebol. Desse modo, promovendo uma parceria entre uma empresa comercial e um estudo acadêmico, esse trabalho busca entender e propor métodos e técnicas para inferir o posicionamento ideal dos jogadores de futebol, adotando algoritmos de aprendizado de máquina. A base de dados contém mais de um milhão de tuplas e passou pela etapa de pré-processamento, a qual demonstrou ser fundamental e de extrema importância, uma vez que gerou novos atributos, eliminou dados incompletos e ruidosos, realizou o balanceamento das classes e removeu *outliers*, preparando assim a base para a execução dos algoritmos k-NN, árvores de decisão, regressão logística, SVM e redes neurais. Com o objetivo de ampliar o entendimento sobre o desempenho e as taxas de acerto, diferentes cenários foram considerados e testados. Houve baixa taxa de acerto quando os algoritmos trabalharam com um problema multi-classe. Os melhores resultados foram obtidos ao utilizar apenas duas classes. Os modelos k-NN e SVM, especificamente para esse estudo, foram aqueles que obtiveram as melhores taxas de acerto. É importante salientar que o SVM consumiu mais de seis horas para finalizar a sua execução, enquanto o k-NN utilizou menos de um minuto para a entrega dos resultados.

Palavras-chaves: Aprendizado de Máquina; Futebol; Classificação; GPS.

Abstract

The professional soccer is always changing and is constantly searching tools and data to help the decision-making, providing tatics and techniques to the team. In Brazil, this sport goes to same way and the investiments are considerables. The One Sports is a company that capture GPS data from professional soccer players of some brazilian teams. This set of data has a lot of features and the One Sports asked if was possible to predict the ideal position of a player. Then, was firmed a cooperation between a academic study and a comercial company. This work find to understand a propose methods and techniques to predict the ideal position of soccer player, using machine learning algorithms. The database has more of one million of tuples. It was submited to pre-processing step, what is fundamental, because generated new features, removed incomplete and noisy data, generated new balaced dataset and delete outliers, preparing the data to execution of the algorithms k-NN, decision trees, logistic regression, SVM and neural networks. With the purpose to understand the performance and accuracy, some scenarios was tested. There was poor results when executed multi-class problems. The best results come from binary problems. The models k-NN and SVM, specifically to this study, had the best accuracy. It is important to note that SVM spent more than six hours to finish your execution, and k-NN used less than one and half minute to end.

Key-words: Machine Learning; Soccer; Classification; GPS.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

GPS	Global Positioning System
Hz	Hertz
k-NN	k-nearest Neighbor
MAD	Distância Absoluta Média
MSE	Erro Quadrático Médio
PCA	Análise de Componentes Principais
SGBD	Sistema Gerenciador de Banco de Dados
SRA	Sistema de Rastreamento Automático
SVM	Support Vector Machines

Lista de símbolos

ac	Acurácia
β	Beta
err	Erro
$\hat{f}$	Função
μ	Mu
σ	Sigma
$\sum$	Somatório

Sumário

1 Introdução

1.1 Contextualização

A história esportiva remete ao surgimento do futebol brasileiro no ano de 1885, trazido da Inglaterra, berço da profissionalização dessa modalidade. A popularização aconteceu de modo significativamente rápido, a considerar a pouca difusão dos fatos e notícias àquela época. Em 1900 é fundada a primeira equipe profissional de futebol do Brasil, a Ponte Preta. (DAOLIO, 2000).

Na atualidade, é cada vez maior o nível profissional dos envolvidos com os clubes, como treinador, departamento médico, além dos próprios jogadores. A busca contínua por melhores condições físicas dos atletas e um melhor posicionamento tático são justificativas para o constante investimento nas análises de dados e na busca de padrões para os jogadores de futebol (SALVO et al., 2007; BOURKE, 2003).

O posicionamento dos jogadores de futebol é, em suma, definido prioritariamente pelo treinador do time e é parte fundamental para o seu sucesso. Essa análise e decisão são baseadas na percepção do técnico e, em alguns casos, conforme as características anatômicas e antropométricas[1] do atleta (GIL et al., 2007).

Sob o olhar da tecnologia e seus avanços, diversos recursos são utilizados para avaliar o desempenho do atleta profissional em seu treinamento e também durante os jogos. Desse modo, é possível identificar algumas tecnologias que objetivam avaliar o desempenho do atleta profissional, dentre elas: dispositivo com GPS e acelerômetro para monitorar o deslocamento do jogador, plataforma de salto para avaliar o desgaste muscular, plataforma de foto-célula para monitorar a explosão muscular, frequencímetro visando avaliar a performance do batimento cardíaco, dentre outros dispositivos de monitoramento (OKAZAKI et al., 2012).

Equipamentos especializados coletam dados individuais para posterior processamento dos mesmos, com o objetivo de adquirir informações relevantes e estratégicas para a comissão técnica. Nesse sentido, a inteligência artificial pode auxiliar fornecendo o apoio de algoritmos específicos os quais permitem a criação de sistemas de auxílio à tomada de decisão. Mais especificamente, os estudos em aprendizado de máquina possibilitam que determinados algoritmos sejam treinados a partir de entradas selecionadas, possibilitando predizer a saída mais indicada para um determinado conjunto de atributos.

Além da aplicação de algoritmos de aprendizado de máquina, é possível realizar

[1] A antropometria é o ramo das ciências humanas que estuda as medidas do corpo, particularmente o tamanho e a forma (RODRIGUEZ-AÑEZ, 2001)

também análises através de métodos não lineares. A aplicação do princípio da redução da dimensionalidade de dados não linear, possibilita a identificação de padrões e, posteriormente, a sua análise através dos eixos e distâncias dos pontos, permitindo assim a obtenção de predições para coordenadas ainda não analisadas, de modo similar ao aprendizado de máquina (ROWEIS; SAUL, 2000).

1.2 Motivação

A motivação principal desse trabalho está relacionada à possibilidade de identificar padrões nos jogadores do futebol brasileiro, uma vez que esse esporte é definido como a principal prática esportiva do país (DAOLIO, 2000). Além disso, a possibilidade de trabalhar em parceria com uma empresa especializada no segmento de captura de dados esportivos de GPS, permitiu a obtenção de uma base de dados com mais de dois milhões de tuplas.

A One Sports aplica análises gráficas aos dados coletados, conseguindo promover melhorias táticas e técnicas aos times assistidos. Buscando ampliar e aprimorar os métodos aplicados, a empresa demonstrou interesse na exploração de técnicas de aprendizado de máquina. Com isso, a principal motivação desse trabalho está na possibilidade de compreender, investigar e analisar soluções para um problema real, utilizando para isso a pesquisa científica. A junção do mundo acadêmico às necessidades reais das empresas é fundamentalmente importante em um mundo onde os requisitos empresariais são constantemente atualizados. A pesquisa e o desenvolvimento no meio acadêmico podem ser um importante modelo de cooperação entre as partes (SEGATTO-MENDES; SBRAGIA, 2002).

1.3 Aplicação Prática

O comportamento posicional de um atleta de futebol dentro de campo é fundamental para o desempenho individual e coletivo, uma vez que existe a busca pelo respeito tático do time. Além de um bom posicionamento, a capacidade de aceleração do atleta também garante um melhor rendimento (DALLAWAY, 2014).

Os dados são coletados a partir de equipamentos GPS acoplados à vestimenta dos atletas, permitindo obter, dentre outros dados, a latitude e longitude em intervalos definidos de tempo, conforme frequência do equipamento, sendo assim possível determinar o posicionamento do atleta em campo, além da sua aceleração (BARBERO-ÁLVAREZ et al., 2010).

Os dados utilizados nesse estudo foram obtidos através dos equipamentos adotados pela One Sports, os quais trabalham com uma captura de dados na velocidade de 6 Hz.

Quanto menor essa frequência, maior o volume de dados coletados por atleta (amostras por segundo).

1.4 Metodologia

Para o desenvolvimento deste trabalho foi concebida uma metodologia que pode ser visualizada na figura 1.4.

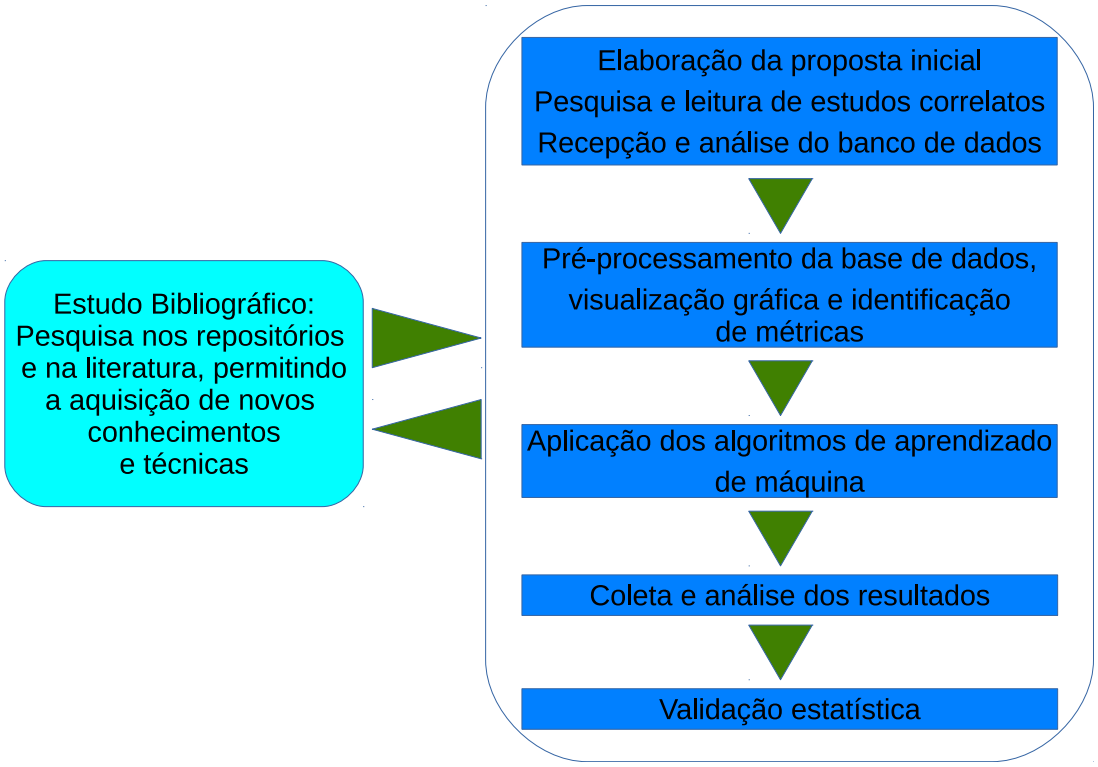

Figura 1: Ilustração da metodologia a ser adotada nesse trabalho

O detalhamento das etapas adotadas na metodologia são descritas a seguir.

Primeiramente, a busca de trabalhos relacionados foi realizada através de portais de trabalhos científicos, dentre eles: IEEE Xplore Digital[2], Scopus[3], ACM Digital Library[4] e Google Scholar[5]. Os termos buscados que originaram os resultados que embasam esse trabalho são detalhados na seção 3.

A seleção dos trabalhos correlacionados auxiliou na definição das técnicas e métodos a serem adotados nesse estudo. Essa etapa é fundamental, pois permite direcionar os esforços a fim de buscar resultados concisos e relevantes.

[2] http://ieeexplore.ieee.org
[3] https://www.scopus.com
[4] http://dl.acm.org
[5] https://scholar.google.com.br

Uma vez permitido o acesso à base de dados da One Sports, foi facilmente identificado o grande volume de tuplas, o que demanda grande poder de processamento e memória. Com isso, foi definida a arquitetura e as ferramentas computacionais a serem adotadas, as quais são detalhadas na Seção 4.

A base de dados obtida precisa de tratamentos, uma vez que possui frações de tuplas repetidas e incompletas, além de alguns dados cadastrados especificamente para testes da plataforma interna da One Sports. Essa etapa é definida nesse trabalho como pré-processamento dos dados, detalhado na Seção 4.1.

Uma vez obtida uma base mais próxima da realidade, é de considerada importância a visualização gráfica dos dados, ajudando a entender o comportamento dos mesmos, além de identificar possíveis tendências, agrupamentos e quais métricas possuem relevância para o estudo. Essa é também uma etapa importante para a identificação de possíveis tratamentos necessários aos dados. As seções 4.6 e 4.2 trazem em detalhes todo o processo.

Finalizada toda a preparação da base e de posse de dados confiáveis para o cenário em questão, as técnicas de aprendizado de máquina serão aplicadas. Toda a base será testada com mais de um algoritmo de aprendizado de máquina. As implementações adotarão a linguagem Python, auxiliada por bibliotecas específicas. Essa etapa permitirá também a validação dos resultados, identificando a capacidade de generalização dos modelos. Para tal, será adotada a técnica *10-fold*, a qual permite a obtenção das taxas de acerto dos algoritmos e seus conjuntos de dados.

1.5 Objetivo

A proposta é investigar, identificar, testar e validar um conjunto de métodos baseados nas práticas de aprendizado de máquina, como eliminação de *outliers*, normalização dos dados e aplicação dos algoritmos, com o objetivo final de determinar a posição ideal de um jogador profissional de futebol, utilizando para isso dados GPS. Uma vez realizadas as etapas previstas e, obtidos os resultados, os mesmos serão comparados e analisados entre si, estabelecendo assim possíveis vantagens competitivas entre os mesmos. Dessa forma, o objetivo desse trabalho é responder à seguinte pergunta:

Qual algoritmo de aprendizado de máquina possui melhor taxa de acerto ao inferir a posição tática de um jogador profissional do futebol brasileiro?

A resposta à pergunta de pesquisa englobará a comparação entre os algoritmos k-NN, árvores de decisão, regressão logística, máquinas de vetores de suporte (SVM) e redes neurais.

1.6 Escopo Negativo

O desenvolvimento desse trabalho é limitado à situações claras e específicas, as quais permitem o direcionamento e manutenção das propostas aqui apresentadas. Desse modo, existem cenários que não serão desenvolvidos, uma vez que não serão contemplados. A fim de evitar desvios ou gerar falsas expectativas, essas negativas são declaradas nessa seção.

Não é prevista a elaboração de uma ferramenta comercial para inferir as posições dos jogadores. Mesmo se tratando de um trabalho com a colaboração de uma empresa com fins comerciais (One Sports), o desenvolvimento e seus resultados possuem foco estritamente acadêmico. Ao final desse estudo é totalmente possível a continuidade dos trabalhos pela empresa parceira, uma vez que a mesma pode se valer dos resultados obtidos e dos experimentos aplicados. Importante ressaltar também que, a ferramenta desenvolvida durante esse trabalho, a qual é apresentada nos próximos capítulos, tem por objetivo prover apoio às execuções e coleta dos resultados.

Nenhum método inédito de aprendizado de máquina será apresentado, nem tampouco variações de modelos já existentes. O objetivo central é coletar resultados e entender o comportamento e a acurácia para cada algoritmo testado, sempre usando os dados fornecidos pela One Sports. Essa colocação não limita a seleção das melhores métricas e a transformação das mesmas, com objetivo de alcançar a melhor performance e qualidade dos algoritmos de aprendizado de máquina.

Considerando o escopo limitado e bem definido desse trabalho, além da adoção de dados que representam características de jogadores e times específicos, não haverá a proposição de um modelo a ser adotado para algoritmos de aprendizado de máquina, o qual tenha o objetivo de inferir o posicionamento ideal dos jogadores em determinado time. Os resultados obtidos são válidos para o cenário em questão, não sendo possível afirmar a obtenção de sucesso ou falha para aplicações, considerando dados coletados em outras circunstâncias e outros times.

1.7 Organização

O primeiro capítulo faz a introdução ao assunto e a contextualização das tecnologias aplicadas ao futebol, além de expor o objetivo desse trabalho. O segundo capítulo aborda os fundamentos e as técnicas utilizadas. Os trabalhos relacionados são discutidos no terceiro capítulo, o qual expõe o diferencial buscado nesse estudo. O quarto capítulo é dedicado à base de dados e seus atributos. Os métodos que foram previamente apresentados são aplicados e descritos no quinto capítulo, o qual detalha também os resultados e suas validades. Ao final desse, ocorre o encerramento do trabalho com a conclusão, as

contribuições obtidas através do desenvolvimento desse estudo, as limitações e os trabalhos futuros.

2 Fundamentos e Técnicas

2.1 Introdução

Esse capítulo aborda alguns métodos que são empregados no aprendizado de máquina. São visitados autores que abordam os processos após a aquisição dos dados, como as etapas de pré-processamento, a identificação dos atributos relevantes, a aplicação nos algoritmos de predição, a coleta dos resultados e a avaliação dos mesmos. Essas etapas são fundamentais para alcançar o objetivo traçado, norteando quanto às técnicas e os procedimentos a serem adotados.

2.2 Pré-Processamento

O pré-processamento é a etapa na qual várias técnicas são aplicadas a fim de eliminar inconsistências da base e realizar transformações necessárias para um melhor desempenho dos algoritmos de predição.

2.2.1 Eliminação de Dados

Os resultados coletados na aplicação dos algoritmos de aprendizado de máquina são impactados diretamente pela qualidade dos dados de entrada. A existência de dados com falhas de integridade impacta diretamente nas predições realizadas pelo modelo. A literatura traz situações clássicas dessas ocorrências, dentre elas:

- Dados incompletos

  Trata-se de tuplas com valores faltantes para determinados atributos que os compõem. Essa ocorrência se deve a vários motivos, dentre eles falha na coleta do dado pelo equipamento, informação opcional, informação desconhecida ou até falha no preenchimento pelo usuário ou operador. Prejudica o modelo de predição, uma vez que a falta de dados nos parâmetros pode provocar uma atribuição incorreta de classe para o problema. Uma das alternativas para esses casos é a remoção das tuplas em questão (FACELI et al., 2011).

- Dados inconsistentes

  Situações quando os valores de entrada são idênticos para duas ou mais tuplas, entretanto suas respectivas saídas são diferentes. Para esses casos o ideal é a remoção de todas as tuplas que gerem conflitos (PILA, 2001).

- Dados redundantes

  Quando são encontradas duas ou mais tuplas idênticas com o mesmo rótulo na mesma base. Essa situação pode provocar um favorecimento da classe em questão. Em geral, o ideal é a identificação e remoção dos objetos com redundância, mantendo apenas uma tupla (FACELI et al., 2011).

- Dados ruidosos

  São dados que possuem características distorcidas dos demais. Geralmente são falhas na leitura do equipamento ou mesmo erro no fornecimento da informação por parte do usuário ou operador. Tendem a confundir o algoritmo, uma vez que esses dados podem não representar a realidade. Recomenda-se a remoção das mesmas (MACHADO, 2007).

Além dessas situações, as quais são mais clássicas, é possível a existência de atributos que carreguem características mais específicas. Nesses casos, é necessário o entendimento da base e a significância de cada atributo e suas possíveis correlações.

2.2.2 Eliminação de *Outliers*

É possível que o processo da coleta dos dados capture valores não condizentes com o real, seja por erro na leitura do equipamento ou algum comportamento não esperado por parte daquele que gera os dados. Atribui-se o nome de *outliers* para esses casos, o qual será também adotado nesse estudo (BEN-GAL, 2005).

A identificação dos *outliers* indica que esses dados requerem uma atenção especial. Variando com o contexto do problema, os mesmos podem ser excluídos ou preservados. A eliminação é a saída geralmente adotada, pois mesmo que os dados em questão estejam corretos, eles fogem da média e desvio padrão, podendo confundir o modelo de predição. A análise e exclusão podem ser realizadas através de técnicas estatísticas univariada ou multivariada. Na primeira abordagem, cada atributo é analisado individualmente. Já na segunda, são consideradas também as correlações desse atributo com os demais, tornando o processo mais seguro, mas é naturalmente mais complexo (BEN-GAL, 2005).

2.2.3 Dados Desbalanceados

Para problemas multi-classes muitas vezes são encontradas bases na qual uma classe é predominantemente superior a outra em número de registros. Essa situação pode induzir o algoritmo, tornando-o tendencioso, uma vez que haverá um maior número de rótulos em relação aos demais. Situações como essas podem ser tratadas da seguinte forma: replicando as tuplas da classe minoritária, de maneira aleatória, garantido que os mesmos atributos gerem a mesma saída da classe; ou diminuindo o número de registros da classe

majoritária. É importante observar que o acréscimo de registros à classe minoritária pode provocar um superajustamento no algoritmo, enquanto a remoção de tuplas das classes majoritárias pode originar uma falta de ajustamento (FACELI et al., 2011).

Essas situações de desbalanceamento são mais evidentes quando a predição é baseada em problemas reais. Segundo Batista (BATISTA; PRATI; MONARD, 2004), o balanceamento nem sempre é viável quando são utilizadas bases com cardinalidade em constante mudança, pois raramente haverá uma igualdade numérica entre as classes. A técnica em questão acaba sendo aplicada de maneira forçada, provocando uma mutação na base e atrapalhando o poder de indução classificadora da mesma. É possível que nesses cenários ocorra um melhor desempenho de bases desbalanceadas, seguindo a natureza do próprio problema investigado. Entretanto, essa abordagem de não igualar a quantidade numérica das classes varia a cada problema e cenário.

Outras duas técnicas para o balanceamento são a definição de custos de classificação para as classes e a indução de modelos. O primeiro tem por objetivo atribuir um peso maior às classes minoritárias. Esse procedimento garante uma equiparação de custos sem alterar o número de registros. Para essa técnica, considerando uma rotulagem incorreta pelo algoritmo, a penalização pode ter impacto negativo superior à classe majoritária. Uma vez que o custo é maior para esse grupo, um erro pode equivaler penalizações maiores. A segunda técnica prega que o treinamento de cada classe deve ocorrer de maneira isolada, evitando assim ajustamentos do modelo para as classes com cardinalidades distintas (FACELI et al., 2011).

2.2.4 Transformação dos Dados

Processo pelo qual é realizada a transformação do dado, seja através da sua mudança de tipo, escala ou como o mesmo é descrito. Esse processo é de extrema importância, uma vez que é passível de ser encontrado na base de dados original registros de modo não compatível com aqueles esperados pelos algoritmos de predição. Para algoritmos que trabalham exclusivamente com números, como redes neurais artificiais e SVM, a conversão simbólico-numérico é imprescindível. Considerando que essa transformação não causa impacto para os outros modelos de predição, é prudente adotá-la (BATISTA, 2003)(FACELI et al., 2011).

As possíveis transformações variam com o algoritmo a ser aplicado. A seguir são listadas algumas técnicas, segundo (BATISTA, 2003):

- Normalização

  A normalização consiste em manter todos os valores de determinado atributo em um mesmo intervalo escalar. Algoritmos baseados em distância podem se tornar tendenciosos para valores altos. O mesmo ocorre nas redes neurais. A normalização

pode ser aplicada de duas maneiras distintas: por distribuição ou amplitude. O primeiro caso é mais aplicável para transformações escalares. (FACELI et al., 2011) cita como exemplo um *ranking*, onde os dados são ordenados de forma crescente, sendo os valores originais substituídos pela posição correspondente ocupada. Já a normalização por amplitude é dividida por padronização ou reescala.

Reescala: Para esse tipo de transformação os limites máximo e mínimo dos valores dos atributos selecionados são definidos entre [-1,1] ou [0,1]. A Tabela 1 exemplifica a aplicação dessa técnica, considerando os limites [-1,1] para o conjunto de dados [4, 5, 5.5, 6].

Tabela 1: Exemplificação da normalização por reescala na transformação dos dados

Valor original	Valor após normalização
4	-1
5	0
5.5	0.5
6	1

A normalização por amplitude por reescala é calculada através da equação demonstrada em 2.1.

$$valor\_novo = min + \frac{valor\_atual - menor}{maior - menor}(max - min) \tag{2.1}$$

Padronização: Transformação capaz de lidar melhor com os *outliers*. Os novos valores são calculados com a Equação 2.2.

$$valor\_novo = \frac{valor\_atual - \mu}{\sigma} \tag{2.2}$$

Após a aplicação, um novo conjunto de dados com distribuição uniforme será criado, uma vez que o espalhamento original é preservado, pois são adotadas as medidas de média (μ) e variância (σ) na transformação. Importante observar que o novo conjunto gerado terá média 0 e variância igual a 1 (FACELI et al., 2011).

A obtenção dos novos valores está disposta na Tabela 2.

- Discretização de atributos quantitativos

  Alguns algoritmos são limitados quanto aos atributos quantitativos. Essa transformação consiste na criação de dados qualitativos a partir dos dados quantitativos (BATISTA, 2003).

- Transformação de atributos qualitativos em quantitativos

Tabela 2: Exemplificação da normalização por padronização na transformação dos dados

Valor original	Valor após normalização
0	-1.336
2	-0.802
4	-0.267
6	0.267
8	0.802
10	1.336

De maneira contrária ao item anterior, alguns algoritmos têm dificuldades em trabalhar com atributos qualitativos. Essa transformação consiste em gerar novos dados quantitativos a partir de valores qualitativos. Um exemplo é a transformação de dados como grande, médio e pequeno. Um alternativa para essas situação é a conversão para 3, 2 e 1, respectivamente (BATISTA, 2003).

- Atributos de tipos de dado complexos

  Quando se faz necessária a utilização de dados com informações complexas, como data e hora, as mesmas não são passíveis de interpretação pelo algoritmo. A conversão desse tipo de dado precisa ser realizada utilizando algum tipo de referência, como adotar um padrão numérico para representar cada data e hora contida nos atributos (BATISTA, 2003).

2.3 Identificação de Atributos

A identificação e seleção ideal do conjunto de atributos é fundamental para o resultado a ser obtido. Para problemas pouco complexos e com um baixo número de atributos, a atenção dispendida pode ser relativamente baixa. Entretanto, para problemas onde a quantidade de variáveis independentes é alta, um maior tempo deve ser dedicado a essa etapa. Usualmente, nem todos os atributos são aproveitados. Dentre as questões passíveis de serem encontradas estão a dependência, relacionamento, complexidade e baixa generalização (PILA, 2001).

Dentro do universo disponível de atributos, a descoberta do grupo correto pode ser muitas vezes exaustiva. Métodos automatizados podem ser adotados nessa etapa, facilitando o processo, apesar de, muitas vezes requerer um alto custo de processamento. PILA (PILA, 2001) destaca três abordagens possíveis:

- *Embedded*

  Modelo capaz de realizar a tarefa de seleção ideal dos atributos para um determinado conjunto de modo integrado ao algoritmo de aprendizado. Segundo (FACELI

et al., 2011), um exemplo dessa técnica são as árvores de decisão.

- Filtro

  São aplicados na etapa de pré-processamento, sem qualquer relação com o algoritmo de predição. Um filtro é aplicado selecionando os k atributos que forneçam o melhor valor e possuam correlação. Embora simples, a sua vantagem está associada ao pouco processamento computacional, além de não utilizar nenhum algoritmo de predição, o que o torna menos dependente de um modelo específico.

- *Wrapper*

  Independem do algoritmo de predição, porém utiliza o mesmo como uma caixa-preta. Apenas alguns atributos são selecionados, gerando assim um novo conjunto, de menor dimensão. Em seguida o algoritmo busca a correlação da seleção com o atributo alvo. Esse processo é repetido até ser encontrado o grupo com o melhor resultado, levando em consideração a taxa de erro, o qual tende a ser o escolhido. O custo computacional é um ponto desfavorável para esse modelo, uma vez que o mesmo tende a se repetir até encontrar o melhor conjunto.

2.4 Modelos Preditivos

2.4.1 k-NN

Algoritmo relativamente simples e com entendimento facilitado pela sua baixa complexidade. Seu princípio é norteado pela distância de seus vizinhos imediatos. O treinamento consiste no cálculo e definição posicional do conjunto dos atributos selecionados, gerando elementos. A distância entre esses é calculada, determinando assim suas características de relacionamento com as classes vizinhas. Uma vez conhecidas as posições dos atributos da base com suas respectivas classes, um elemento não classificado é inserido juntamente aos demais. Nesse momento, a distância do seu k vizinhos é calculada - k é um valor inteiro, preferencialmente ímpar e não muito distante de zero. Esse algoritmo prega que o novo elemento assumirá a classe predominante ao seu entorno, limitando-se a k vizinhos. A importância de k assumir um valor ímpar está relacionado ao fator empate, evitando a ocorrência dessa situação. Já um número não distante de zero, garante uma amplitude limitada de vizinhos, evitando uma distorção da classe de saída. (WU et al., 2008).

Uma das estratégias possíveis para definir o valor ideal de k para o problema é a adoção da validação cruzada. Outra abordagem é a definição de pesos para cada vizinho. Quanto mais próximo estiver a classe vizinha do novo elemento a ser rotulado, maior o seu peso na votação. A formalização da definição da classe do novo elemento, para problemas de classificação, equivale a: $\hat{f}(x_t) \longleftarrow modal(f(x_1), f(x_2), ..., f(x_k))$ (FACELI et al., 2011).

2.4.2 Árvores de Decisão

Árvores de decisão são relativamente simples e possuem performance aceitável ao que se propõe. A ideia, de modo geral, é resolver problemas complexos através de decisões simples e encadeadas, obtendo ao final dessas o resultado. Basicamente, o algoritmo se estrutura dividindo os dados em conjuntos finitos e com a geração de novos filhos, conforme a execução do algoritmo evolui (SAFAVIAN; LANDGREBE, 1991).

A sua estruturação, em forma de árvore, permite que apenas os valores da variável buscada chegue ao nó folha, o qual é uma função. Para os demais casos, aqueles mais simples, a função assume o valor constante, a fim de minimizar o custo (FACELI et al., 2011).

Segundo (SAFAVIAN; LANDGREBE, 1991), os objetivos principais de uma árvore de decisão são:

- Classificar corretamente o maior número possível de amostras de treinamento, mantendo a capacidade de generalização.
- Ser fácil de inserir novos dados conhecidos de treinamento.
- Ter a estrutura o mais simples possível.

CART (*Classification and Regression Trees*) é um algoritmo para indução de árvores de regressão passível de ser adotado para problemas onde a variável dependente é categórica. O seu processo de construção se dá a partir de alguns passos essenciais. A sua inicialização ocorre da divisão binária da raiz. Desse modo, o nó pai passa a ter filhos. De maneira recursiva, o processo de divisão continua, aumentando a árvore. Essa rotina será repetida até que o critério de parada seja satisfeito. O modelo CART utiliza o índice Gini, o qual mede a dispersão dos dados presentes dos nós das árvores. Com isso, é possível obter a probabilidade de erro resultante entre as classes, permitindo a aplicação da poda. De maneira simplificada, a poda é o processo no qual é possível gerar novas árvores menores e de menor complexidade, encontrando assim o melhor modelo para o problema, sempre buscando o menor custo (RODRIGUES; VIEIRA; SILVA, 2013).

2.4.3 Regressão Logística

Método estatístico muito empregado em aprendizado de máquina, o qual trabalha com saídas categóricas. É passível a sua aplicação em problemas multi-classes, mas possui uma melhor adaptação às situações com classes binárias (HAIR; ANDERSON, 2005).

Trata-se de um modelo que possui forte relação entre variáveis independentes e a variável dependente e não exige, obrigatoriamente, que a distribuição dos dados ocorra de

modo normal, apesar de ser uma característica desejável. A saída gerada é um resultado probabilístico entre 0 e 1 (GONÇALVES, 2005).

O seu modelo é baseado na função sigmóide $f(z)$, definida em 2.3. Entretanto, é convencionalmente usada conforme demonstrada em 2.4.

$$f(z) = \frac{1}{1 + e^{-}z} \tag{2.3}$$

$$f(z) = \frac{e^{z}}{1 + e^{z}} \tag{2.4}$$

A partir da função logística é possível garantir uma saída entre $0 \leq f(z) \leq 1$ (GEVERT et al., 2011).

O modelo logístico pode ser obtido a partir da função logística $f(z)$, promovendo a soma das variáveis independentes, demonstrada em 2.5. Posteriormente ocorre a substituição na função 2.6.

$$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n \tag{2.5}$$

$$f(y) = \frac{e^{\beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n}}{1 + e^{\beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n}} \tag{2.6}$$

$f(y)$ é o valor probabilístico buscado para as variáveis independentes (GEVERT et al., 2011)

2.4.4 Máquinas de Vetores de Suporte

Máquinas de vetores de suporte (do inglês *support vector machines*), ou simplesmente SVM, é um conjunto de métodos de aprendizado supervisionado que busca inferir saídas únicas para problemas binários. Baseia-se na construção de hiperplanos[1] ou espaços dimensionais infinitos. Trata-se de um classificador com técnicas avançadas e de considerável robustez nas predições (BURGES, 1998).

Considerando um conjunto de atributos de entrada denominado x e uma saída conhecida, denominada y, o SVM pode ser definido como $x_i \mapsto y_i$. Uma das principais características do SVM está relacionada com a capacidade do modelo trabalhar com diferentes configurações (*kernels*). Desse modo, baseado na teoria do aprendizado estatístico, é possível fornecer respaldo matemático para a definição do melhor classificador, tomando por base os conjuntos de treinamento do domínio em questão (BURGES, 1998) (FACELI et al., 2011).

[1] linha limítrofe de separação entre as classes

O hiperplano criado pelo modelo SVM permite a distinção entre as classes. O espaço de separação é denominado de margem. A figura 2 ilustra um hiperplano linear (BURGES, 1998).

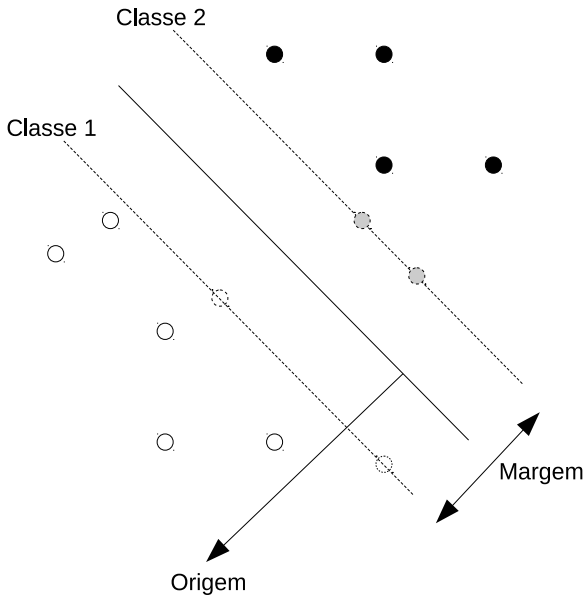

Figura 2: Hiperplano SVM linear - Imagem baseada no original (BURGES, 1998)

Além da aplicação para problemas lineares, o SVM também pode ser aplicado para cenários onde os dados não sejam facilmente separáveis, impactando diretamente na sua margem. Para esses casos, o SVM é capaz mapear um novo espaço a partir do original. A fim de auxiliar a compreensão e a disposição dos dados. O novo plano criado possui uma maior dimensão quando comparado com o original, permitindo uma melhor separação das classes (FACELI et al., 2011).

O SVM possui mais de um *kernel*, os quais, resumidamente, podem ser definidos como modelos ou configurações do algoritmo, permitindo que um mesmo problema seja tratado de formas distintas, variando conforme a escolha do *kernel* e suas definições (BURGES, 1998).

2.4.5 Redes Neurais Artificiais

Modelo inspirado na neurociência, a qual busca reproduzir, de maneira computacional, o funcionamento de uma rede neural. Seus princípios são norteados pela robustez, tolerância às falhas, capacidade de aprendizagem, paralelismo e processamento de informação incorreta[2] (RAUBER, 1997).

O modelo de redes neurais artificiais tende a ser computacionalmente custoso, tornando, muitas vezes, o processo lento. Requer também um grande volume de dados

[2] capacidade de identificação e ajuste para dados de entrada com ruídos ou divergências

para treinamento. Por outro lado é um algoritmo com uma taxa de acerto considerável e lida bem com problemas multi-classes (FACELI et al., 2011).

Uma rede neural artificial é composta por um conjunto de camadas, o qual é formado por um grupo de neurônios[3]. A estrutura de uma rede neural artificial é dada pelo seu tipo, pelas características das suas unidades de processamento, pelo sistema de interligação dos nós e pela, propriamente dita, função de aprendizagem. Existem várias formas de treinar uma rede neural artificial. Será destacado o treinamento com *backpropagation*, uma vez que é um dos mais utilizados (GEVERT et al., 2011). A demonstração das conexões de uma rede neural artificial é apresentada na figura 3.

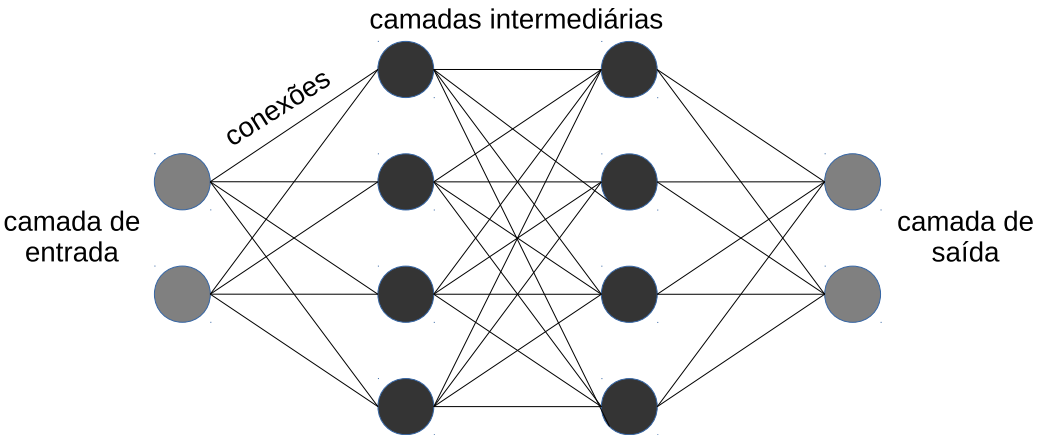

Figura 3: Rede neural artificial e suas conexões - Imagem baseada nos originais de (CARVALHO, 2005) e (FACELI et al., 2011)

Em geral, uma rede neural é definida pela camada de entrada, a qual recebe os dados, uma ou mais camadas intermediárias, as quais são responsáveis pela maior parte do processamento e, finalmente, a camada de saída, responsável pela entrega do dado. A importância da topologia de rede para esse modelo está diretamente ligada ao processamento da informação pelo neurônio, o qual gera um estímulo que é transmitido pelas suas conexões, as quais, por sua vez, possuem pesos associados, sendo fator multiplicador do estímulo recebido. Desse modo, cada neurônio é responsável por uma parte da função de ativação, resultando assim na saída da rede (GONÇALVES, 2005).

O algoritmo de *backpropagation* se destaca no ajustamento dos pesos. No processo de treinamento, o dado é apresentado à primeira camada intermediária da rede. Nesse momento, cada neurônio aciona a função de ativação, passando o resultado para o neurônio da camada seguinte. Esse processo é denominado de *forward* e busca encontrar o erro cometido pela rede, calculando a diferença dos valores de saída. Uma vez obtido o valor, inicia-se o processo de *backward*, a fim de realizar o ajustamento dos pesos de entrada, iniciando na camada de saída até a primeira intermediária. O ciclo de *backpropagation* se

[3] nome idêntico aos neurônios cerebrais, mas simulado por elementos de processamento

repete até a definição de parada, a qual pode se dar de diferentes maneiras, como número máximo de ciclos ou taxa máxima de erros. (FACELI et al., 2011).

2.5 Avaliação dos Modelos Preditivos

2.5.1 Métricas de Erro Para Classificação e Regressão

Cada modelo de predição possui suas características, considerando suas vantagens e desvantagens. Ao analisar o problema onde será empregado o algoritmo de predição, determinadas peculiaridades induzem a adoção do modelo mais condizente com o cenário. Entretanto, muitas vezes é difícil afirmar com exatidão a escolha do modelo ideal, sem a execução do mesmo. A escolha final será de um único algoritmo para um determinado problema, entretanto será necessária a execução de diferentes modelos a fim de compará-los. Além dessa definição, é preciso considerar também que cada algoritmo possui ajustes de parâmetros, os quais podem otimizar a qualidade das predições. Essas variações precisam também de medições que ajudem na escolha final. As métricas de erro são obtidas através da apresentação de objetos ainda desconhecidos ao modelo de predição (FACELI et al., 2011).

Segundo (FACELI et al., 2011), para problemas de classificação, adota-se a equação 2.7, a qual calcula a medida equivalente à função custo. A variação do erro fica entre os valores 0 e 1, sendo os extremos próximos a 0 os melhores, conforme equação 2.8. Essa medida é tomada corresponde à acurácia do modelo verificado. Quando o problema é de regressão, o erro pode ser calculado tomando por base o valor de y_i confrontado com a predição do modelo. As equações 2.9 e 2.10 fornecem o erro quadrático médio e a distância absoluta média, respectivamente. Quão menores os seus resultados, maior a probabilidade do resultado inferido estar correto.

$$err(\hat{f}) = \frac{1}{n}\sum_{i=1}^{n} I\left(y_i \neq \hat{f}(x_i)\right) \tag{2.7}$$

$$ac(\hat{f}) = 1 - err(\hat{f}) \tag{2.8}$$

$$MSE(\hat{f}) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}(x_i)\right)^2 \tag{2.9}$$

$$MAD(\hat{f}) = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{f}(x_i)\right| \tag{2.10}$$

2.5.2 Métodos de Validação

Conforme abordado em 2.5.1, a busca pelo algoritmo que melhor se adapte ao problema é de extrema importância. Após o treinamento é possível realizar a predição, propriamente dita, gerando o comparativo entre os diferentes modelos, calculando assim as suas acurácias. Para que seja possível mensurar esses valores, amostras não rotuladas precisam ser apresentadas para os modelos. Das possíveis técnicas que podem ser adotadas, duas serão apresentadas: *Holdout* e Validação Cruzada.

- *Holdout*

  Esse método realiza a divisão de todo o conjunto em dois sub-conjuntos. Geralmente a divisão ocorre na proporção 66% para treinamento e 34% para teste. Uma vez realizado o treinamento do algoritmo em questão o sub-conjunto de teste é apresentado ao mesmo, entretanto sem a rotulagem das amostras. O objetivo básico é que o algoritmo realize as predições e rotule as amostras. Uma vez finalizado esse processo, serão confrontados os dados rotulados pelo modelo de predição com os rótulos originais, permitindo assim o cálculo da acurácia e outras métricas (KOHAVI, 1995).

- Validação Cruzada

  Conhecido também por *k-fold cross-validation* o conjunto original C é fracionado em k sub-conjuntos $C_1, C_2, ..., C_k$ exclusivos e idênticos quanto ao tamanho. Considerando um rotacionamento dos sub-conjuntos, sendo i o sub-conjunto atual, o processo se dá a partir do treinamento dos $(k - i)$ sub-conjuntos e teste de predição com o sub-conjunto i, promovendo a rotatividade de i por k vezes. Ao final de cada rotação, os valores preditos corretamente e incorretamente são computados. Concluída as k rotações, todos os dados coletados são calculados e a acurácia pode ser obtida, além de outras métricas (KOHAVI, 1995).

2.6 Considerações Finais

Cada modelo de predição possui suas características, as quais devem ser consideradas no momento da escolha, tomando por base o problema, os atributos e a quantidade de amostras. Além disso, o poder computacional deve ser levando em consideração.

A escolha dos atributos deve ser feita com muito cuidado e baseada em técnicas apropriadas. Testes com diferentes combinações de variáveis independentes são também um ponto importante, pois ajudam a entender melhor o reflexo na variável dependente.

O pré-processamento se apresenta como etapa fundamental para o sucesso das predições, uma vez que dados ruidosos e inconsistências tendem a confundir o modelo, diminuindo, muitas vezes, significativamente a qualidade das inferências.

É válido pontuar algumas características dos modelos apresentados nessa seção, tomando por base (FACELI et al., 2011), (GONÇALVES, 2005), (BURGES, 1998) e (GEVERT et al., 2011):

- Algumas características do modelo kNN:

  - Capaz de lidar bem com problemas de classificação e regressão.

  - Algoritmo de fácil implementação e entendimento.

  - Aceita bem novas amostras, uma vez que é necessário apenas a sua incorporação à base.

  - Requer atenção ao valor de k, sendo esse, muitas vezes, determinante para o sucesso da implementação do algoritmo.

  - Geralmente não lida bem com problemas complexos e de grande dimensionalidade, pois torna-se custoso em relação ao processamento.

  - Para problemas pouco complexos é vantajoso quanto à velocidade da predição; diminui a sua eficácia de maneira inversamente proporcional à medida em que aumenta a complexidade do problema.

  - Trabalha bem com problemas multi-classes.

- Algumas características do modelo Árvores de Decisão:

  - Muito utilizado graças a sua flexibilidade e robustez.

  - Capacidade de selecionar bem os atributos, além de ser eficiente e possuir boa interpretação do problema.

  - O seu modelo promove a repetição de informações e estruturas, podendo gerar instabilidade e morosidade no processo.

- Algumas características do modelo Regressão Logística:

  - Capacidade de trabalhar com problemas multi-classes.

  - Dispensa a obrigatoriedade da distribuição normal dos dados.

  - Sua saída é probabilística.

  - Capacidade de trabalhar com problemas categóricos.

  - Seus resultados são impactados diretamente pelas escolhas das variáveis independentes.

- Algumas características do modelo SVM:

  - Possuem boa generalização.

  - Trabalha com hiperplanos, permitindo uma separação mais segura das classes.

  - Classifica apenas problemas binários.

  - Pode se tornar computacionalmente custoso para problemas complexos.

- Algumas características do modelo Redes Neurais:

  - Tende a ser computacionalmente custoso, tornando, muitas vezes, o processo lento.

  - Requer também um grande volume de dados para treinamento.

  - Capaz de lidar bem com problemas multi-classes.

O processo de validação é uma etapa fundamental a fim de obter o melhor modelo para o problema. Apesar das características de cada algoritmo de predição nortearem a escolha, apenas as validações são capazes de demonstrar na prática o modelo mais acertado.

3 Trabalhos Relacionados

A busca de trabalhos relacionados foi realizada através de portais de trabalhos científicos, dentre eles: IEEE Xplore Digital[1], Scopus[2], ACM Digital Library[3] e Google Scholar[4]. Os termos buscados que originaram os resultados que embasam esse trabalho foram: gps soccer, gps soccer reliability, motion analysis in soccer, standard deviation soccer e machine learning gps sports.

Inicialmente foi realizada a busca *adhoc* nos portais acima mencionados, tomando por base as palavras chaves. Os critérios de busca foram:

(("gps soccer") OR ("motion analysis sports") OR ("machine learning sports"))
Considerando os repositórios pesquisados e a variação de termos buscados, um considerável volume de documentos científicos foi obtido, conforme disposto na tabela 3.

Tabela 3: Quantidade de artigos encontrados por repositório

Repositório	Quantidade de Artigos
Scopus	61
ACM	15
IEEE	116
Google Scholar	169

A considerar que o total de resultados obtidos foi de 361 documentos, foi notada a importância de refinar os resultados e trabalhar apenas com aqueles correlatos a esse trabalho. Para isso foi utilizado primeiramente o critério de relevância das ferramentas de busca. Dentre esses, foram selecionados os trabalhos pertinentes ao assunto para leitura prévia. Por fim, foram lidos em sua totalidade aqueles que condiziam e que possuíam paralelos a esse estudo. Nessa etapa, foram observadas as referências bibliográficas comuns a essas publicações, sendo esse um importante indicativo de relevância no meio científico acerca do assunto. Nesse momento foi utilizada a técnica de *snowballing*. Além dessas técnicas, também foram adotadas literaturas consideradas base para o problema estudado. Com esse material definido, foi realizada a seleção final dos trabalhos, considerando aqueles que melhor condiziam e contribuíam com esse estudo. O refinamento dos documentos e suas quantidades são demonstrados na tabela 4.

Finalmente, a leitura dos conteúdos selecionados e a pesquisa acadêmica permitiram traçar um paralelo dos trabalhos já desenvolvidos e as lacunas com potencial de exploração.

[1] http://ieeexplore.ieee.org
[2] https://www.scopus.com
[3] http://dl.acm.org
[4] https://scholar.google.com.br

Tabela 4: Quantificação por etapas dos trabalhos relacionados

Etapa	**Artigos Impactados**	**Pré-Selecionados**
Busca *adhoc*	+361	361
Relevância das Ferramentas de Busca[5]	-276	85
Seleção dos trabalhos com base na introdução dos mesmos	-58	27
Leitura completa dos estudos	-4	23
Referências comuns / *snowballing*	+16	39

A seleção final resultou num total de trinta e nove publicações, variando entre livros, artigos científicos e teses. As referências desse trabalho totalizam quarenta itens, pois uma trata especificamente da abordagem de uma ferramenta comercial.

Os dados analisados nesse estudo são originados a partir de equipamentos GPS. Desse modo, as leituras iniciais se concentraram em publicações correlatas a esse assunto. Os trabalhos selecionados que embasaram esse estudo resultaram em sete documentos que abordam o tema GPS. Já a análise por vídeo teve uma seleção final de cinco trabalhos, o qual foi importante no entendimento das técnicas de análises sobre esse esporte. Uma vez que o objetivo era a aplicação dos algoritmos de aprendizado de máquina sobre esses dados, foram utilizadas nove publicações sobre o tema. Considerando a a alta cardinalidade da base utilizada nesse estudo, compõem os trabalhos relacionados um total de nove publicações relacionadas à análise de dados. Além dos já citados, seis estudos sobre a prática esportiva do futebol integram esse trabalho. Completam essa lista duas publicações sobre análise física e uma sobre a cooperação entre universidades e empresas.

Inicialmente foi possível confirmar que o futebol profissional, em âmbito mundial e igualmente no Brasil, recebe altos investimentos em tecnologia, processos, equipamentos e profissionais qualificados. Analisar os dados individuais da atuação de cada jogador em campo não é, definitivamente, uma tarefa fácil. Métodos tradicionais podem ser morosos e dispendiosos para o clube. A informação rápida e atual é um poderoso diferencial para o time, seus jogadores e a comissão técnica (CARLING et al., 2008).

As pesquisas demonstraram que o método de análise por vídeo é bastante difundido, o qual obtém a movimentação e distribuição dos jogadores, sendo assim um dos métodos passíveis de utilização para a coleta de dados. Nos anos noventa, um processo mais rudimentar já era baseado nessa mesma técnica. Os dados eram coletados através da repetição de jogos gravados em fitas de vídeo. Observadores anotavam todas as informações possíveis, como chutes a gol, faltas, passes longos, dentre outros. Em paralelo, análises físicas eram realizadas, como testes de corridas, velocidade e resistência anaeróbica dos atletas[6]. Todos esses dados eram confrontados, obtendo informações preciosas sobre o time

[6] capacidade de repetir por diversas vezes a corrida na faixa máxima de aceleração, sem perda considerável

e seus jogadores (MEYER; OHLENDORF; KINDERMANN, 2000).

Mais recentemente, a mesma técnica de utilização de imagens continua a ser empregada, mas de forma muito mais automática e autônoma, conforme (BARROS et al., 2007; SALVO et al., 2007). A captura ocorre através de modernas câmeras instaladas em pontos estratégicos e de boa amplitude visual no estádio. Esse método é denominado de Sistema de Rastreamento Automático (SRA). A captura dos quadros é de excelente qualidade visual, a fim de que sejam analisadas posteriormente por um software específico, o qual baseia seus cálculos na análise de variância, tomando por base as posições obtidas a partir das imagens.

A utilização de câmeras para monitoramento aceita variações, conforme o propósito ao qual se designa. A exemplo disso, a análise posicional dos juízes de futebol foi estudada por (D'OTTAVIO; CASTAGNA, 2001). O objetivo era entender o padrão e promover comparativos para os árbitros, o qual possui um padrão singular se comparado com os jogadores da mesma modalidade esportiva. No estudo em questão foram identificados alguns padrões e atributos, os quais podem ser verificados na Tabela 5.

Tabela 5: Padrões e atributos para árbitros de futebol através de coleta por vídeo - tabela baseada em (D'OTTAVIO; CASTAGNA, 2001)

	Períodos			
Atributos	**0-15(min)**	**30-45(min)**	**45-60(min)**	**75-90(min)**
Distância total (m)	1925 ± 211	1802 ± 197	1799 ± 195	1773 ± 193
Baixa intensidade (m)	752 ± 106	711 ± 123	736 ± 115	730 ± 107
Média intensidade (m)	502 ± 108	437 ± 98	439 ± 102	416 ± 85
Alta atividade (m)	257 ± 88	244 ± 71	236 ± 79	243 ± 80
Repouso (s)	108 ± 47	135 ± 57	133 ± 56	140 ± 55

A Tabela 5 destaca quatro atributos relacionados à intensidade da distância percorrida e um atributo com ligação temporal. A observação positiva nesse estudo específico está relacionada à possibilidade da captura de informações valiosas a partir do monitoramento por vídeo, inclusive permitindo a captura da velocidade e explosão muscular do árbitro. O lado negativo fica evidenciado na expressiva variação dos dados coletados, aumentando demasiadamente a margem de erro. Esse fator está ligado ao número de câmeras utilizado, sendo apenas duas. Segundo a ProZone®(PROZONE, 2016), é importante dimensionar corretamente a quantidade dos equipamentos de captura, observar cuidadosamente à posição de instalação dos mesmos e utilizar corretamente o software de análise das imagens, pois são fatores decisivos para a acurácia da tecnologia.

Considerando que foram analisados apenas os quinze minutos iniciais e finais de cada tempo e, apesar de sofrer uma variação considerável nos dados coletados, (D'OTTAVIO;

de velocidade (SIENKIEWICZ-DIANZENZA; RUSIN; STUPNICKI, 2009)

CASTAGNA, 2001) identificou padrões para o comportamento do árbitro durante as partidas. O período "0-15 (min)"é aquele que mais se diferencia dos outros. A justificativa está baseada no início do confronto entre os times, uma vez que existe a busca por uma imposição tática e de domínio sobre o adversário. É também o momento em que o árbitro busca entender o comportamento dos jogadores, de maneira individual e coletiva, exigindo mais da sua atenção e presença à curta distância. De modo geral, as intensidades diminuem com o desenrolar do jogo, entretanto a velocidade máxima demonstra ser ascendente, o que sinaliza um aumento nos contra-ataques. Fica evidenciado que, mesmo analisando apenas os dados comportamentais do árbitro, é possível analisar a intensidade de um jogo de futebol profissional.

Ao analisar jogadores e árbitros através da técnica do SRA é preciso considerar que os juízes não sofrem impactos físicos diretos, como faltas e divididas, raramente vão ao chão e não tem contato com a bola, como chutes ao gol, por exemplo. Apesar de se preparem para a atividade, geralmente não são profissionais com dedicação exclusiva, fazendo com que, muitas vezes, não obtenham o nível máximo em excelência física (D'OTTAVIO; CASTAGNA, 2001).

A decisão pela utilização do SRA, seja para o rastreamento dos jogadores ou dos árbitros, deve considerar, impreterivelmente, a qualidade dos equipamentos e a escolha estratégica da instalação. Contrário a isso, a acurácia da coleta poderá se demonstrar pouco confiável (BARROS et al., 2007).

É destacada por (BARBERO-ÁLVAREZ et al., 2010; HENNIG; BRIEHLE, 2000) a técnica de captura dos dados dos jogadores através de equipamentos que utilizem o sistema de posicionamento global (*Global Positioning System - GPS*). Essa tecnologia é baseada na determinação de pontos obtidos através de satélites. Um transmissor, localizado na Terra, é o responsável por enviar as coordenadas e assim obter, como resposta, o posicionamento e a localização do mesmo. Uma vez sequenciado esse processo, é possível capturar informações quanto ao deslocamento e posicionamento dos jogadores, sendo o equipamento carregado junto à sua roupa durante as atividades físicas, principalmente nos jogos oficiais. Posteriormente, as informações coletadas são transferidas para um software específico, o qual também é capaz de gerar dados quantitativos sobre os atletas.

O GPS demonstra uma tendência natural, uma vez que está difundido em grande parte da sociedade, muito graças aos *smartphones*. A maioria desses eletrônicos já vem equipado com os recursos de GPS e acelerômetro. Para atletas profissionais, o ideal é a utilização de equipamentos específicos, pois possuem melhor precisão e uma frequência maior de registro dos dados. Para atletas não profissionais, é totalmente possível utilizar os recursos dos *smartphones*, possibilitando análises e comparativos de atletas profissionais com simples praticantes de esporte por saúde e lazer (MITCHELL; MONAGHAN; O'CONNOR, 2013).

Apesar de serem modelos diferentes, ambos são capazes de determinar o trajeto executado pelo atleta, permitindo assim a obtenção de mais informações, como distância percorrida, organização posicional do time, características individuais, dentre outras. Em relação à exatidão das tecnologias de coleta aqui apresentadas, ambas possuem um nível de precisão confiável e similar entre si, desde que respeitadas as condições técnicas exigidas para cada tecnologia. Desse modo é seguro optar por qualquer um dos métodos (EDGECOMB; NORTON, 2006).

Comparando o processo de instalação dos equipamentos para as tecnologias citadas, é preciso considerar que o SRA exige um considerável investimento e preparação, uma vez que um conjunto de câmeras específicas é necessário. Somado a isso, a instalação é muitas vezes complexa, pois os pontos de fixação são estratégicos e precisam ter grande amplitude visual (PROZONE, 2016; BARROS et al., 2007). Por sua vez, o GPS demanda a necessidade de equipamentos acoplados à roupa de cada atleta, além de tecnologia e software compatíveis para a interpretação dos dados (EDGECOMB; NORTON, 2006).

Em estudo com características próximas a esse, os dados foram coletados utilizando as duas tecnologias citadas acima, entretanto em momentos diferentes. Especificamente para os treinos, adotou-se a coleta dos dados baseada em GPS. Já para as partidas oficiais, foi utilizada a captura das imagens através de câmeras apropriadas. É importante entender o cenário de aplicação para que exista nexo nessas escolhas. Os treinos geralmente são realizados em campos reduzidos[7], garantido um melhor aproveitamento das condições anaeróbicas dos atletas. Já os jogos, são sempre realizados utilizando as medidas oficiais. Considerando a amplitude visual, as câmeras são apenas adotadas nos estádios, onde existe a tecnologia contratada para captura por imagem (DALLAWAY, 2014).

Em relação ao volume de dados, ao ser analisado outro estudo, o qual adotou como equipamento para captura *smartphones* com acelerômetros - ao invés de equipamentos específicos, o mesmo analisou a prática esportiva em diferentes modalidades, sendo possível a captura numa frequência de 16 a 25 Hz. O volume de dados coletados à essa frequência foi suficiente para gerar dados significativos para o estudo em questão (MITCHELL; MONAGHAN; O'CONNOR, 2013).

Uma vez conhecido os métodos de captura adotados em outros estudos, foram analisados trabalhos com aderência no processamento dos dados capturados, almejando a geração de informações relevantes.

Os dados coletados precisam ser bem caracterizados, permitindo assim identificar as suas correlações e importância. A correta identificação desses atributos é fundamental para o sucesso da aplicação, pois as saídas sempre estão condicionadas à qualidade dos dados de entrada. Para aplicações de GPS no futebol, são relevantes aquelas que possuam

[7] usualmente nos centros de treinamento

associação com velocidade, aceleração e intensidade, dados esses passíveis de serem obtidos com a utilização do GPS nos jogadores (DALLAWAY, 2014).

EDGECOMB & NORTON (EDGECOMB; NORTON, 2006) estudaram o monitoramento de atletas de futebol australiano[8] utilizando GPS e vídeo. O objetivo proposto foi, além de entender o comportamento geral dos atletas durante uma partida, obter dados quanto ao posicionamento e deslocamento coletivo e individual, além de aplicar o comparativo entre as tecnologias de monitoramento. O estudo demonstrou que foi possível rastrear todas as ações dos jogadores em campo, propiciando a coleta de dados em um volume considerável, permitindo realizar análises computacionais posteriores. Trata-se de uma modalidade esportiva diferente da proposta nesse estudo, mas demonstra a versatilidade das tecnologias.

É preciso considerar também a forma de aplicação e análise dos dados. É possível que o processamento dos dados e obtenção das informações aconteça em tempo real, ou seja, durante a própria partida. Outra forma é a análise dos dados pós-jogo. AUGHEY & FALLOON (AUGHEY; FALLOON, 2010) observaram que, para o futebol australiano, a margem de erro é maior quando a aplicação ocorre durante o jogo, devendo o técnico ser mais cauteloso com os dados obtidos dessa forma.

Ao analisar o futebol de campo, especificamente, o processamento dos dados capturados objetiva a identificação de padrões, buscando características próximas. É possível definir, por exemplo, que jogadores com atuação no meio de campo tem por característica cobrir uma maior área do campo em distância percorrida. Ainda é possível observar, dentro desse mesmo grupo, que os "meias laterais"atuam com uma maior intensidade, quando comparado com os "meias centrais". Essas características podem ser observadas nos atributos distância e aceleração, respectivamente (DALLAWAY, 2014).

Considerando a tecnologia GPS e suas aplicações para as mais diversas áreas, inclusive no futebol, foram observados os recentes estudos publicados. O número de artigos acadêmicos relacionados com o assunto demonstrou uma crescente evolução tecnológica dos aparelhos de captura de dados esportivos, além da difusão da tecnologia no meio esportivo, evidenciando assim uma tendência crescente sobre o assunto, sendo esse um importante indicador de contemporaneidade. A fim de confirmar essa propensão, foi realizado o levantamento do número de publicações dos últimos cinco anos. É facilmente notada a tendência de curva ascendente na figura 4. Salvo uma pequena queda no em 2016, os demais anos acumularam sucessivos aumentos de publicações correlacionados a esse estudo. O levantamento anual demonstrado na figura 4 foi realizado utilizando exclusivamente o repositório Google Scholar.

[8] modalidade esportiva semelhante ao rugby, mas com adaptações e regras específicas, o que o diferencia dos outros desportos

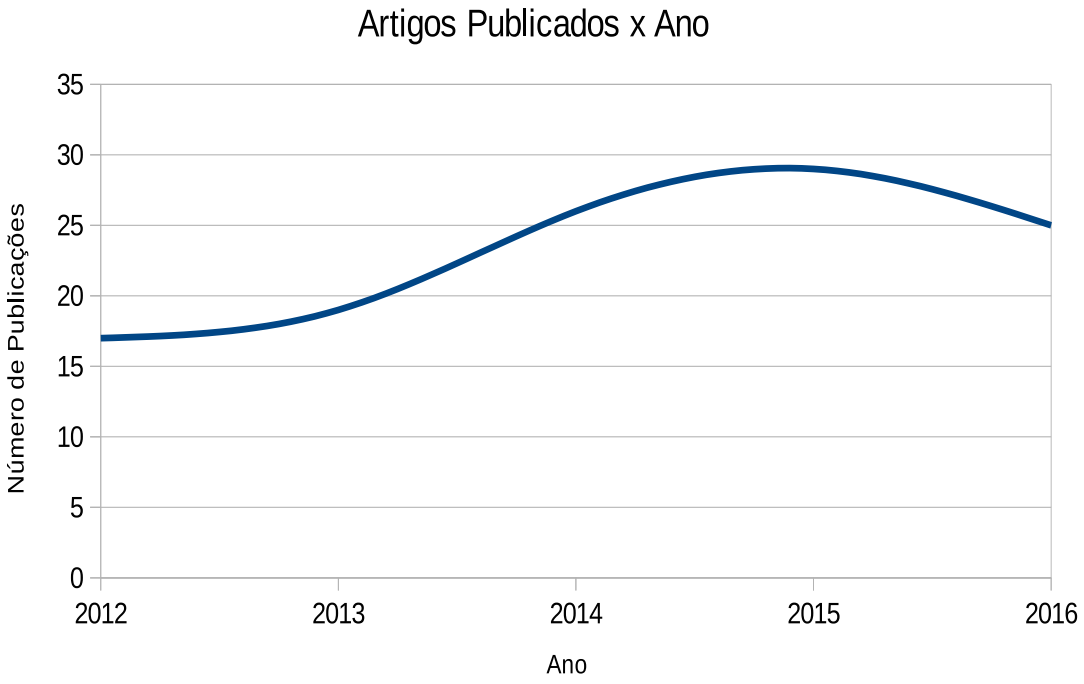

Figura 4: Quantidade de artigos publicados por ano

3.1 Considerações Finais

Os trabalhos relacionados evidenciam que o número de publicações se mantém em uma quantidade substancial ano após ano. Entretanto é possível observar também que há muito potencial a ser explorado. A aplicação do GPS em jogos oficiais é recente e suas análises ainda são poucas. Os estudos demonstram, principalmente, situações posicionais, sem grande aprofundamento.

Uma parte considerável dos trabalhos relacionados se preocupa com a tecnologia de captura e sua acurácia - em relação ao equipamento, deixando, algumas vezes, a análise detalhada dos dados em segundo plano.

Outra observação importante é que uma parcela considerável dos trabalhos aplica a mesma tecnologia aqui descrita para outras modalidades esportivas, como rugby e futebol australiano.

Ao analisar os trabalhos apresentados nesse capítulo, juntamente com aqueles que embasam as afirmações e as teorias observadas nos demais, é possível identificar a escassez de publicações que tratem sobre a aplicação de algoritmos de aprendizado de máquina utilizando dados GPS de jogadores profissionais do futebol brasileiro. São encontrados diversos trabalhos que abordam esses assuntos de maneira isolada, como aprendizado de máquina, análise posicional do time, seja por vídeo ou GPS, análise anaeróbica do atleta e

capacidade de explosão muscular. A proposta apresentada nesse trabalho visa identificar qual modelo possui maior confiabilidade na predição da posição tática do jogador, unindo assim os conceitos e aplicações de aprendizado de máquina no cenário futebolístico, com olhar específico para os atletas profissionais brasileiros, considerando o seu comportamento posicional baseado em dados GPS.

Essa pesquisa, juntamente com os trabalhos correlatos, são um forte indicativo que o assunto proposto e tratado nesse estudo é atual, com aderência acadêmica e com potencial de aplicação prática. Sob a perspectiva futura, trata-se de um assunto capaz de promover novos trabalhos, norteados pela mesma área de interesse.

4 Base de Dados e Atributos

4.1 Preparação da Base

Os dados GPS utilizados nesse estudo são reais e foram disponibilizados pela empresa OneSports[1]. Os mesmos são referentes à primeira divisão do futebol profissional brasileiro.

Os dados fornecidos estavam previamente rotulados com posições determinadas do futebol. As mesmas são detalhadas na tabela 6.

Tabela 6: Posicionamento dos jogadores de futebol que serão considerados nesse estudo (SCAGLIA et al., 1996)

Posição	Descrição
Atacante	Jogador que tem por objetivo receber a bola no campo de ataque do seu time e finalizar a jogada com o intuito de marcar o gol
Lateral Direito	Jogador que atua pelo lado direito do seu time. A sua principal função é permitir que a bola saia do setor defensivo e chegue até os jogadores do meio campo. Por possuírem características de velocistas, muitas vezes fazem a ligação direta com os jogadores de ataque
Lateral Esquerdo	Mesma função que o lateral direito, entretanto atua pelo lado esquerdo
Meia Atacante	Jogadores com atuação no meio do campo e com características de suporte direto aos atacantes do time, além de exercer pressão na saída de bola do adversário
Meia Central	Tem por função atuar na central do campo, provendo apoio ao ataque e defesa do seu próprio time
Meia Defensivo	Função similar ao meia atacante, entretanto fornece apoio mais direto à defesa da sua equipe
Zagueiro	Jogadores designados especificamente para defender o seu time e eliminar o risco de gol do adversário

Os dados foram disponibilizados no formato CSV (*Comma-separated values*). Dada a necessidade da manipulação dos mesmos e, sabendo que essa base precisaria ser conectada à uma plataforma de testes, optou-se por importá-los para o sistema gerenciador de banco de dados (SGBD) MySQL®.

Uma vez importados, os trabalhos iniciais foram focados no entendimento da base. Os dados contemplavam uma cardinalidade de 3.281.948 tuplas. Desse total, foram

[1] http://www.onesports.com.br

observados conjuntos com valores inconsistentes, os quais não deveriam permanecer. Uma vez que demonstravam ser prejudiciais para o estudo, pois podem confundir os modelos, esses foram removidos, conforme detalhado na tabela 7.

Tabela 7: Limpeza de dados na base original

Motivo da Limpeza	Dado Observado	Total de Tuplas Removidas
Dados ruidosos	Distância percorrida igual a zero	978.197
Dados incompletos	Rótulo (posição do jogador) nulo	6.789

O motivo da captura de dados com inconsistências não pode ser definido nesse estudo, uma vez que não há elementos determinísticos que demonstrem com clareza a causa.

Em continuação à preparação da base, foram identificadas tuplas com problemas distintos, sendo: dados de calibração[2] dos equipamentos GPS, dados de testes, dados de treinos e dados sem relacionamento entre as tabelas do banco de dados, quando as mesmas usavam chaves estrangeiras. Novamente optou-se pela eliminação desses dados, pois são potenciais indutores de erros aos modelos de aprendizado de máquina escolhidos (FACELI et al., 2011).

A eliminação das tuplas identificadas resultou numa cardinalidade de informações GPS da ordem de 1.190.782 tuplas. É uma diminuição significativa quando comparada à base original, entretanto, a partir desse momento, é conhecido que os dados que serão analisados e testados são reais e confiáveis, gerando resultados mais seguros.

Cada tupla da base de dados contém o detalhamento momentâneo do atleta, gerando uma sequência de informações com variações mínimas de dados para cada atributo. Esses dados, permitem entender o comportamento do jogador enquanto esteve em campo. Ao realizar agrupamentos na base novas informações são obtidas, as quais estão dispostas na tabela 8.

Tabela 8: Quantificações considerando agrupamento das tuplas

Informação	Quantificação
Número de Jogos	30
Número de Times	22
Número de Atletas/Jogadores	73
Número de Campos/Estádios	16

Sabendo que cada jogador guarda uma posição definida dentro do time e que a

[2] Processo pelo qual o equipamento faz sincronismo com os satélites e ajusta o seu posicionamento longitudinal e latitudinal em relação ao campo de futebol e seus quatro cantos, obtendo assim maior precisão (YEH et al., 2006)

base encontra-se rotulada, a quantificação de tuplas por posição de atleta está disponível na tabela 9.

Tabela 9: Número de tuplas das classes dos jogadores na base de dados

Posição Rotulada	Total de Tuplas
Atacante	200.438
Lateral Direito	121.340
Lateral Esquerdo	35.240
Meia Atacante	407
Meia Central	337.945
Meia Defensivo	273.398
Zagueiro	222.014
Total de Tuplas	1.190.782

É possível observar com base na tabela 9 a existência de desbalanceamento das classes. Usualmente essa característica tem impacto negativo nos modelos preditivos, uma vez que o classificador pode se tornar tendencioso. Entretanto, é importante notar que, para problemas reais não é raro encontrar situações de desbalanceamento, uma vez que os dados oscilam conforme o tipo de necessidade de representação. Um exemplo clássico da literatura que pode ser mencionado é o algoritmo que busca inferir uma determinada doença em um grupo específico de pacientes. Comumente uma classe será predominante, dos doentes ou dos sadios, variando com o grupo analisado. Esse exemplo demonstra claramente a necessidade de treinar um determinado algoritmo com bases desbalanceadas, onde naturalmente as classes não são uniformes (BATISTA; PRATI; MONARD, 2004).

Será proposto nesse estudo uma diminuição da disparidade numérica encontrada entre as classes, realizando para isso o balanceamento. Será empregada a adoção da técnica de replicação das classes minoritárias, aumentando assim quantidade numérica dessas tuplas, exceto para a classe "Meia Atacante", a qual é muito inferior às demais e será totalmente removida, pois não existe significância numérica dessa classe no universo do problema (BATISTA; PRATI; MONARD, 2004).

4.2 Tabelas e Atributos

A base de dados obtida possui um total de quatro tabelas, as quais são descritas a seguir:

- Activities

  Contém informações relativas aos jogos. Os seus atributos fazem referência ao local, data, hora, times e duração dos jogos. A sua cardinalidade contempla 111 registros.

- Dados

  Tabela disponibilizada pela One Sports com dados sumarizados das partidas e das movimentações dos jogadores. Trata-se de dados já processados, os quais não serão utilizados nesse estudo, pois passaram por transformações e cálculos, podendo descaracterizar ou induzir os resultados. Ao total são 1.527 tuplas.

- Fields

  Possui informações sobre campos de futebol e que são passíveis de receber jogos utilizando a tecnologia GPS. Seus principais atributos são relativos ao nome do estádio e as coordenadas de latitude e longitude das quatro extremidades do campo. A sua cardinalidade é de 97 registros.

- Gps

  Principal tabela a ser utilizada nesse estudo. A mesma contém o detalhamento da movimentação de cada jogador, partida a partida, sendo a geração/captura dos dados numa frequência de 6 Hz (conforme equipamentos adotados pela One Sports). Entre os diversos atributos da tabela, destacam-se nome do jogador, posição (rótulo), distância percorrida, velocidade, aceleração, latitude, longitude e tempo instantâneo. A sua cardinalidade é de 1.190.782 tuplas.

A escolha dos atributos relevantes para esse estudo está diretamente associada aos princípios de velocidade, distância e posicionamento dos jogadores em campo. Desse modo, foram selecionados todos os atributos com dados correlatos a essas características. Os atributos previamente selecionados estão dispostos na tabela 10. Todas as colunas e dados analisados estão na base de dados nomeada como Gps.

Tabela 10: Atributos selecionados previamente com base na velocidade, distância e posicionamento em determinado período de tempo

Atributo	Tipo de Dado	Descrição
Position	Texto	Contém a posição rotulada para o jogador
Distance	Decimal	Armazena a distância percorrida pelo atleta
Speed	Decimal	Velocidade do jogador
Acceleration	Decimal	Armazena a aceleração promovida pelo atleta
Latitude	Decimal	Dado relativo ao posicionamento latitudinal do jogador em campo
Longitude	Decimal	Posicionamento longitudinal do atleta em campo
Instant_time	Hora	Contém a hora da captura dos dados referente à tupla em questão, variando em segundos

4.3 Transformações dos Atributos

O pré-processamento em uma dada base de dados tem fundamental importância, uma vez que ajusta o seu conteúdo com o intuito de obter um melhor aproveitamento do algoritmo de aprendizado de máquina. Além disso, diminui o custo computacional, evitando que sejam promovidas transformações dos dados durante a execução dos mesmos. Nesse trabalho, a etapa de pré-processamento está concentrada nas etapas de exclusão dos atributos não relevantes ao problema, eliminação de ruídos (vide 4.1), conversões e transformações dos atributos.

Essa seção destaca a preocupação da padronização, transformação e validade dos atributos, mantendo-as na mesma simbologia e com informações equivalentes e coerentes para o problema.

Todos os atributos previamente selecionados possuem seu grau de importância para esse estudo (vide 4.2), entretanto duas delas, especificamente latitude e longitude, carregam características valiosas para a análise aqui proposta. A sua importância está relacionada ao entendimento do comportamento posicional do atleta em campo, uma vez que essas coordenadas permitem conhecer cada movimento executado pelo atleta durante o jogo profissional. Apesar de sua importância e aparente impossibilidade de descartá-las, foi observada nos dados armazenados na base uma divergência do padrão dos dados entre um jogo e outro, muitas vezes do mesmo atleta. Essa situação se deve ao fato da realização dos jogos em dezesseis estádios diferentes, uma vez que os times alternam o local das suas partidas.

Para cada campo cadastrado no sistema, as latitudes e longitudes das suas quatro extremidades são armazenadas (*as quatro marcas de escanteio do campo*). Com essa informação é possível acompanhar o posicionamento do atleta em campo, uma vez que é formado um retângulo com suas referências e limites. Como a informação GPS não se repete para nenhum ponto distinto do planeta, as coordenadas de cada campo serão únicas. Desse modo, a primeira transformação necessária foi o reposicionamento de todos os jogadores dentro de um mesmo campo, garantido assim que não exista influência da variação de latitude e longitude provocada por campos distintos.

Para o reposicionamento dos atletas foi eleito o campo do Maracanã, situado na cidade do Rio de Janeiro, uma vez que o mesmo possui medidas oficiais e pode ser facilmente mapeado pela ferramenta *Maps Google*[3]. O mapeamento realizado pode ser visualizado na figura 5.

Para que fosse possível o reposicionamento no Maracanã, todos os campos que receberam os jogos originais foram mapeados, permitindo assim calcular as movimentações dos atletas. Para haver consistência no processo, foi eleito o canto superior à direção norte

[3] https://www.google.com.br/maps

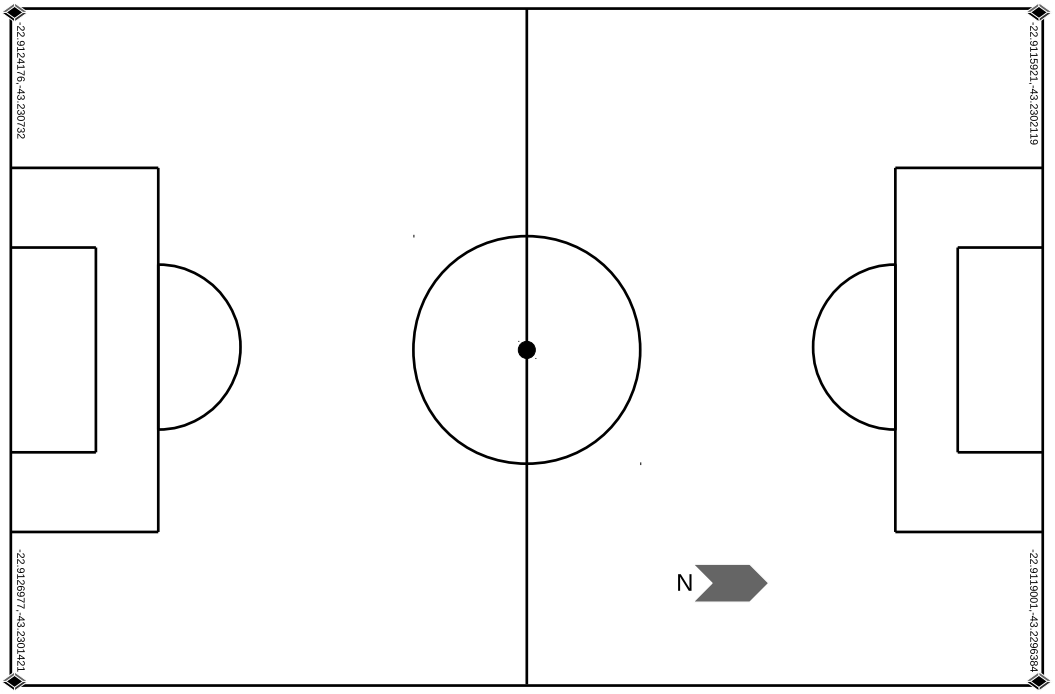

Figura 5: Mapeamento latitudinal e longitudinal das extremidades do campo do Maracanã

de cada estádio como referencial para os cálculos. O objetivo é a obtenção de dois novos dados para cada tupla: distância e ângulo do atleta - ambos relativos ao canto eleito do campo.

Para definir a distância do jogador em relação ao ponto base do campo, foi implementada a função em Python, demonstrada a seguir[4]:

```
def calcular_distancia(self,latitude1,longitude1,latitude2,longitude2):
        import numpy as np
        R = 6373.0 # raio aproximado da Terra em KM
        lat1 = np.deg2rad(latitude1)
        lon1 = np.deg2rad(longitude1)
        lat2 = np.deg2rad(latitude2)
        lon2 = np.deg2rad(longitude2)
        dlon = lon2 - lon1
        dlat = lat2 - lat1
        a = np.sin(dlat / 2)**2+np.cos(lat1)*np.cos(lat2)*np.sin(dlon / 2)**2
        c = 2 * np.arctan2(np.sqrt(a), np.sqrt(1 - a))
        distance = R * c
        return round(distance*1000,1)
```

Para que seja possível saber a posição do atleta, além da distância é preciso obter também o seu ângulo relativo ao ponto de referência que foi determinado. Desse modo, para cada tupla da base de dados foi calculada esse novo atributo utilizando a função em

[4] algoritmos adaptados a partir de funções publicadas e documentações Python

Python abaixo:

```
def calcular_angulo(self,lat1,lon1,lat2,lon2):
        import numpy as np
        bearing = np.arctan2(np.sin(lon2-lon1)*np.cos(lat2),
                  np.cos(lat1)*np.sin(lat2)-np.sin(lat1)*np.cos(lat2)*
                  np.cos(lon2-lon1))
        bearing = np.degrees(bearing)
        bearing = (bearing + 360) % 360
        return bearing
```

Considerando o número de tuplas e a necessidade de cálculo dos novos atributos, essa etapa do pré-processamento consumiu um total de 34 horas para ser executada. Ao final, foram obtidos os dados de ângulo e distância para cada tupla da base de dados, gerando assim dois novos atributos.

Conhecendo a posição do jogador no campo de origem e de posse das coordenadas dos quatro cantos do campo de destino - Maracanã, foi necessário realizar o cálculo das novas posições dos jogadores no campo de destino (latitude e longitude). Para que fosse possível o cálculo desses dados, foi adotada a utilização da biblioteca *geopy*[5] da linguagem Python. O seu uso facilita a aplicação de algoritmos baseados em geocódigos.

Foi eleito o canto superior à direção norte do Maracanã como referência para as novas posições, seguindo o mesmo padrão adotado nesse estudo. Todas as tuplas da base de dados foram percorridas, calculando as novas coordenadas do jogador em relação ao estádio do Maracanã. Uma vez que as mesmas foram obtidas, houve o registro no banco de dados, gerando dois novos atributos, nomeadas no banco de dados como *new_latitude* e *new_longitude*. A função em Python utilizada para o reposicionamento dos jogadores é demonstrada a seguir:

```
def reposition(self,lat1,lon1,distancia,bearing):
        from geopy import Point
        from geopy.distance import distance, VincentyDistance
        distancia = distancia / 1000; #converte a distancia para metros
        location = (VincentyDistance(kilometers=distancia).destination(
                   Point(lat1, lon1), bearing))
        lat2 = location.latitude
        lon2 = location.longitude
        return lat2, lon2
```

No código acima é possível observar a utilização da função VincentyDistance. Criada por Thaddeus Vincenty, a mesma objetiva calcular a distância entre dois pontos distintos, tomando por base o globo terrestre. Sua margem de erro é de no máximo 0.5 mm. A biblioteca geopy suporta essa função (VINCENTY, 1975).

[5] https://pypi.python.org/pypi/geopy

Uma vez que as coordenadas GPS são importantes para a análise aqui proposta e, todos os atletas foram reposicionados em um mesmo campo, passa a existir equivalência para esse atributo, tornando-o mais confiável. Entretanto, ao prosseguir com a preparação da base e dos atributos, foi observada uma inconsistência que pode induzir o modelo de predição ao erro. Todos os jogos registrados na base de dados são compostos de dois tempos de quarenta e cinco minutos. Cada time ocupa um lado do campo para cada tempo. Ao término desse, ambos invertem os lados. Analisando sob a perspectiva de apenas um time, quando ocorre a inversão as áreas de atuação em campo são guardadas por posições diferentes. O zagueiro passa a atuar na posição do atacante e vice-versa. O mesmo ocorre para as outras posições, salvo os jogadores de meio de campo, os quais sofrem menos impacto com essa modificação. Como na base de dados não existe uma identificação clara sobre primeiro e segundo tempo e, não há registros de qual lado do campo o time atuou, as novas coordenadas podem gerar algum tipo de confusão para o algoritmo de aprendizado de máquina.

Em busca de uma informação mais genérica do posicionamento dos jogadores em campo, optou-se pela criação de um novo atributo. Uma vez que todos estão reposicionados no mesmo estádio, foram calculados o ângulo e a distância de todas as tuplas em relação ao meio de campo. Esses novos atributos impactam de maneira menos direta nos dados, pois é possível definir a distância do jogador em campo, independente do seu lado de atuação. A figura 6 exemplifica essa situação.

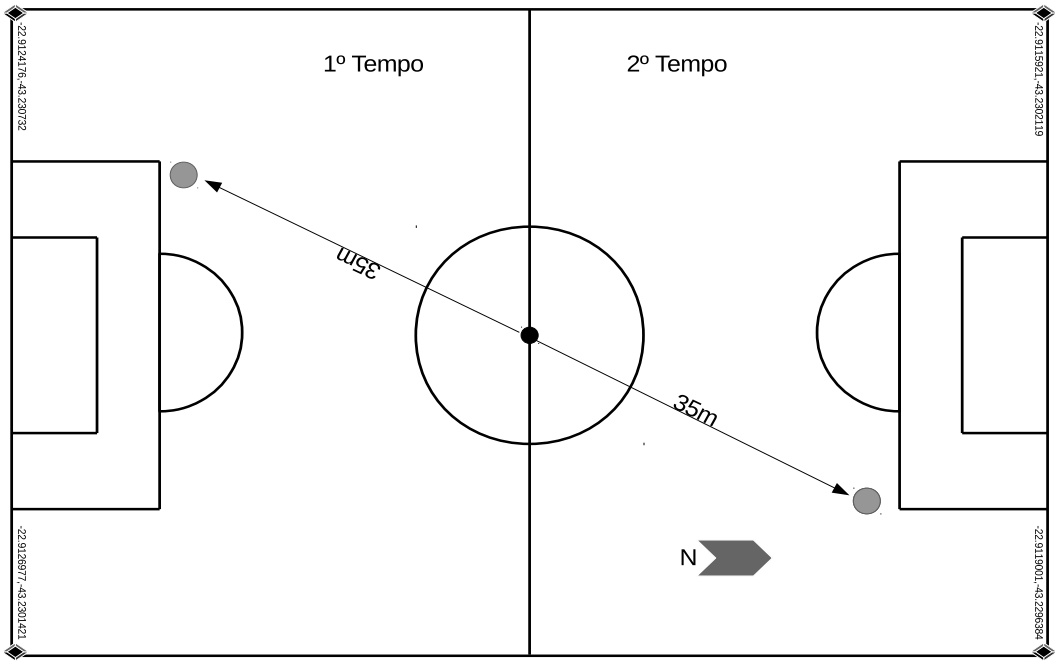

Figura 6: Distância do zagueiro em relação ao meio de campo durante o primeiro e segundo tempo

É importante observar que a mudança do lado do campo não impacta a distância que foi calculada a partir do centro, entretanto, o ângulo ainda é afetado quando as equipes invertem o gol de ataque.

É válido ressaltar também que o Maracanã possui medidas oficiais, entretanto, alguns dos estádios onde ocorreram os jogos contemplados nesse estudo são de dimensão igual ou menor, quando comparados ao Maracanã. O reposicionamento não contemplou essa compensação. Caso os atributos selecionados para a aplicação dos algoritmos sejam especificamente a latitude e longitude do reposicionamento, a informação estará falha, pois não houve o devido ajuste. Por outro lado, se os atributos utilizados forem baseados na distância e ângulo do *corner 1* (referência), os mesmos não sofrerão impacto, pois os dados são equivalentes ao campo de origem. A seleção dos atributos é demonstrado na seção a seguir.

4.4 Seleção dos Atributos

Após a realização das etapas de pré-processamento que foram identificadas como relevantes a esse trabalho, um novo número de atributos foi obtido. É importante, nesse momento, o entendimento quanto à relevância dos mesmos e, inclusive, a identificação de correlação entre esses. Buscar manualmente o melhor grupo de atributos tende a ser um processo moroso e sujeito a falhas. Tomando por base a seção 2.3, foi escolhida a técnica automatizada de seleção denominada *wrapper*. O método adotado foi a eliminação recursiva de atributos. Com o auxílio das bibliotecas correspondentes em *Python*, os dados foram submetidos ao algoritmo capaz de identificar o melhor grupo para o problema em questão. A figura 7 destaca as taxas de acerto obtidas para cada conjunto de atributos.

Na figura 7 é possível observar que a melhor taxa de acerto está relacionada ao conjunto composto por seis atributos. O algoritmo com a técnica de seleção *wrapper*, implementado em *Python*, foi executado diversas vezes, adotando diferentes combinações, entretanto nenhum outro conjunto se demonstrou superior ao seguinte: *distance*, *speed*, *acceleration*, *angle_corner*, *distance_corner* e *distance_middle*.

A escolha da técnica *wrapper* ocorreu mediante a quantidade numérica de atributos e as possíveis combinações que as mesmas geraram. Apesar de computacionalmente mais custoso, foi uma alternativa mais rápida e confiável, se comparado com os outros métodos existentes.

Conforme destacado na seção 4.3, mesmo ocorrendo o reposicionamento dos jogadores no Maracanã, as coordenadas de latitude e longitude não foram eleitas na seleção dos atributos a serem utilizados, desse modo, a distância relativa ao *corner 1* foi preservada. Para o atributo *distance middle*, o valor é diferente se comparado com o campo de origem, entretanto estão todos os jogadores realocados no Maracanã, mantendo o ponto central do

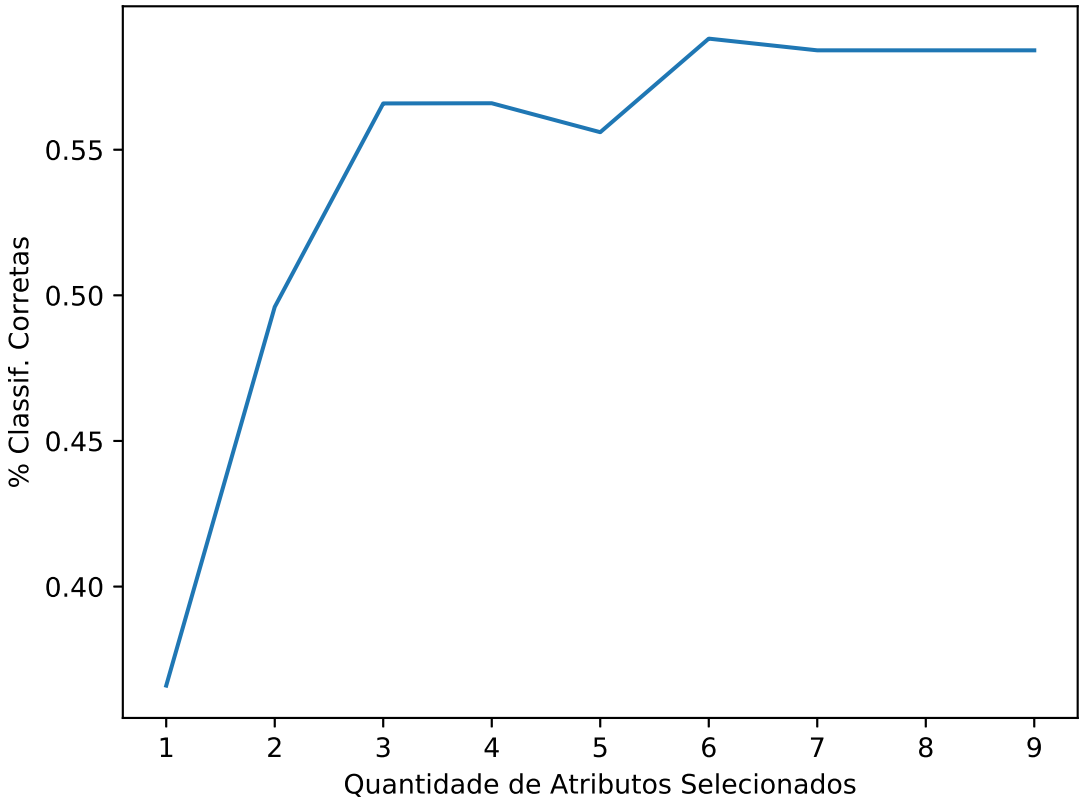

Figura 7: Aplicação do método *wrapper*

campo idêntico para esse cálculo.

4.5 Normalização

Após a seleção do grupo de atributos mais indicado para esse estudo, foi aplicada a normalização dos dados, uma vez que determinados algoritmos de aprendizado de máquina têm melhor desempenho com os dados normalizados.

Quando são comparadas diferentes posições de jogadores, fica nítida a variância nos dados. Se for adotada a técnica de normalização por reescala, essa característica pode ser perdida, uma vez que os dados ficarão com um intervalo escalar reduzido. Na busca da preservação dessas características foi adotada nesse estudo a normalização por padronização.

4.6 Visualização Gráfica dos Dados

A visualização gráfica é uma etapa importante, pois permite o entendimento dos dados e suas correlações. É possível também compreender o seu espalhamento, identificando possíveis *outliers*.

A aplicação da técnica *wrapper* retornou um total de seis atributos (4.4). Apenas é possível dispor os dados graficamente tomando por base dois ou três eixos, conforme o problema. Desse modo, não é possível a visualização, uma vez que o número de atributos é superior ao total de eixos possíveis (x, y e z). A fim de possibilitar a visualização gráfica dos dados, foi utilizada a técnica de Análise de Componentes Principais (PCA).

Do inglês *Principal Component Analysis*, a PCA permite a redução da dimensionalidade de um conjunto de dados. Seu objetivo é a criação das matrizes T e P', a partir dos produtos da matriz mãe X. As duas novas matrizes criadas carregam a característica de serem menores que sua mãe, uma vez que seus produtos têm origem nos dados principais de X. A aplicação da PCA permite também a rotação de eixos, mas modificando graficamente a perspectiva sobre os dados (HAIR; ANDERSON, 2005) (WOLD; ESBENSEN; GELADI, 1987).

Para que fosse possível uma visualização gráfica de todas as classes, foi aplicada a PCA. Apesar da técnica promover a rotação e uma aproximação das amostras, a figura ajuda a entender a disposição das classes. Na figura 8 é mostrado o espalhamento de todos os dados tomando por base a visualização em duas dimensões após a aplicação da PCA.

É possível observar na figura 8 que as amostras de diferentes classes compartilham praticamente o mesmo espaço. Entretanto, essa característica é comum quando aplicado a PCA. Por outro lado, é possível observar que algumas amostras se distanciam das demais, podendo caracterizar *outliers* ou mesmo características diferenciadas.

Todas as visualizações gráficas dessa seção adotaram os dados já normalizados.

A figura 9 demonstra o espalhamento dos dados de todas as classes tomando por base a visualização em três dimensões após a aplicação da PCA.

Com o objetivo de identificar a existência de possíveis *outliers*, as figuras 10 e 11 exibem o espalhamento dos dados para as classes "meia central"e "atacante", respectivamente.

Tomando por base as figuras 9, 10 e 11 é possível observar a existência de *outliers*. Desse modo, fica evidente a necessidade da eliminação dos mesmos, evitando prejuízos aos classificadores.

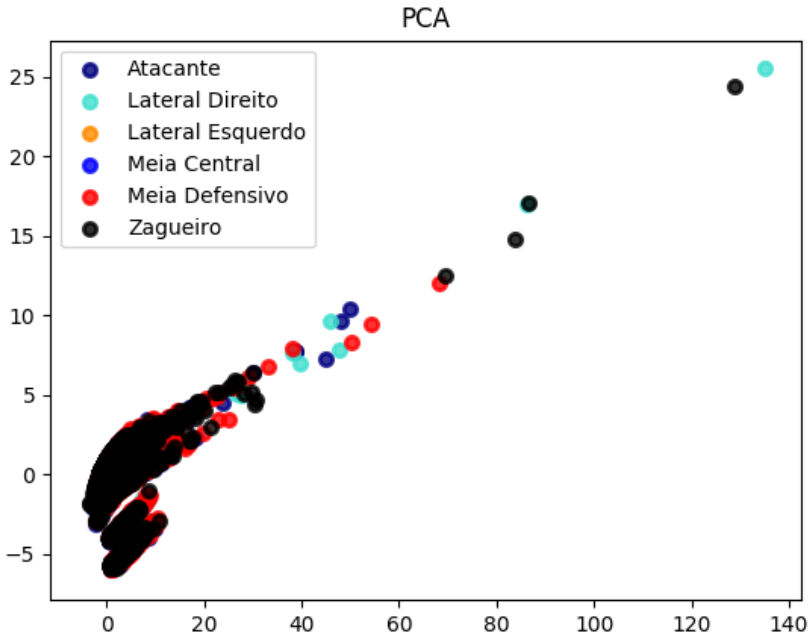

Figura 8: Dados reduzidos para 2 dimensões utilizando PCA

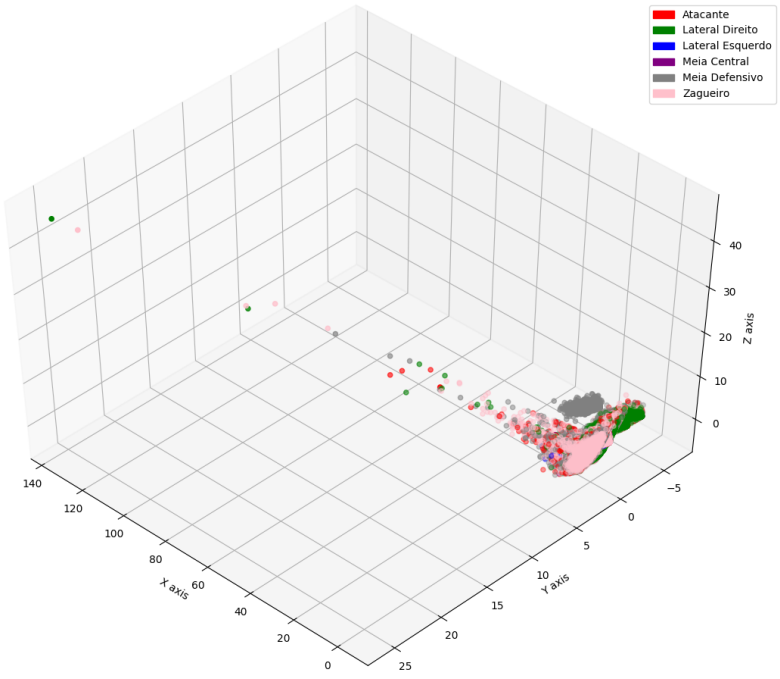

Figura 9: Todas as classes reduzidas para 3 dimensões utilizando PCA

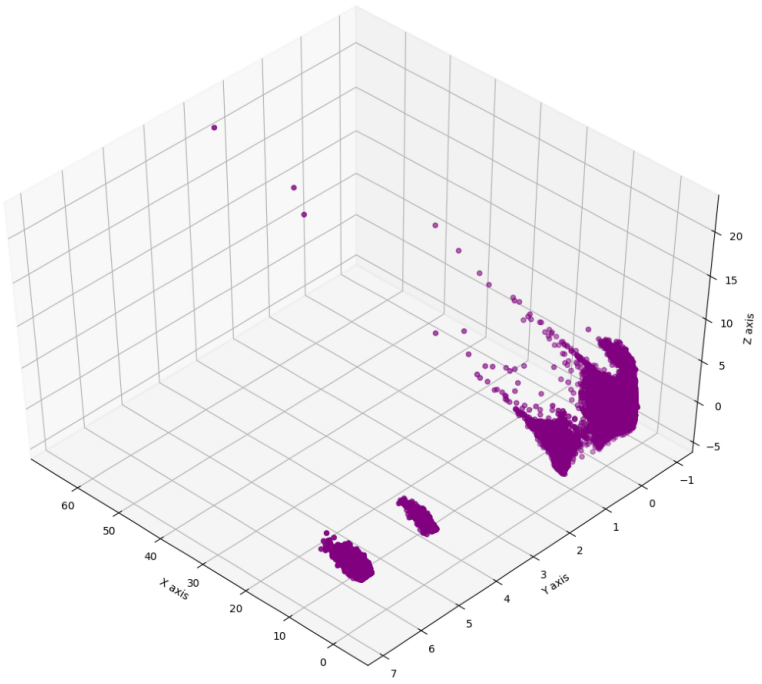

Figura 10: Classe "meia central"reduzida para 3 dimensões utilizando PCA

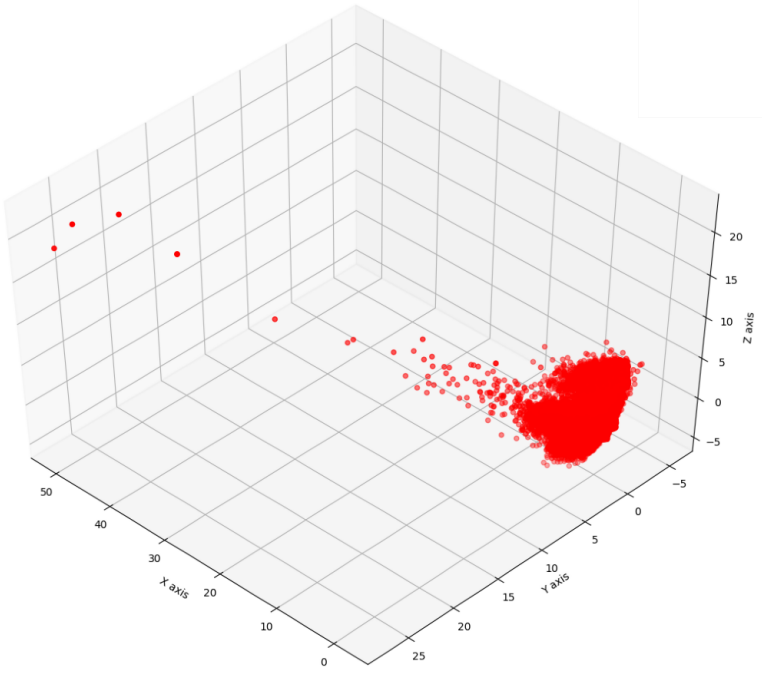

Figura 11: Classe "atacante"reduzida para 3 dimensões utilizando PCA

4.7 Eliminação de *Outliers*

Uma vez identificada a presença de *outliers* através da visualização gráfica (4.6), a eliminação dos mesmos foi necessária, conforme descrito na seção 2.2.2.

Considerando os seis atributos selecionados, foram realizados testes para identificar quais os dados que possuíam maior impacto nos *outliers*. Foi observado que a "distância"é o atributo que influenciava de maneira mais relevante nos pontos fora de padrão, quando comparado com a maioria da mesma classe. Assim, foi adotada a estratégia de eliminação das tuplas para dados identificados como *outliers* no atributo "distância". O método usado foi o baseado nos intervalos interquartis, onde o dado deve possuir valor entre os percentis 75% e 25%, adotando a equação mostrada na equação 4.1.

$$IQR = Q_3 - Q_1, \tag{4.1}$$

onde: Q_1 é o quartil inferior e Q_3 é o quartil superior.

E obtendo finalmente o *outlier*:

$$[Q_1 - 1.5.IQR, Q_3 + 1.5.IQR], \tag{4.2}$$

devendo o valor alvo estar entre o primeiro e segundo limites obtidos na equação 4.2[6].

As figuras 12, 13 e 14 exibem, respectivamente, todas as classes, apenas a "meia central"e "atacante".

Com base nas figuras, após a eliminação dos *outliers* é possível observar a diminuição significativa de dados fora da área de maior concentração. É visível também a rotação de eixos promovida pela PCA e a sobreposição de classes, especificamente na figura 12.

4.8 Balanceamento das Classes

Após a eliminação dos outliers, detalhado na seção 4.7, a base de dados ficou composta pelas classes e cardinalidades dispostas na tabela 11.

A classe "meia atacante", conforme citado anteriormente, foi eliminada completamente, uma vez que possuía uma quantidade de tuplas pouco significativa para o problema. Assim, houve uma redução total de 99.372 registros, os quais foram considerados potenciais *outliers*.

Mesmo após o processo de remoção de dados inconsistentes, é possível observar na tabela 11 a divergência numérica entre as classes. Enquanto o "meia central"possui

[6] Fórmula interquartil e codificação em Python obtidas no sítio https://www.datacrucis.com/research/find-outliers-in-an-array.html em 12/12/2017.

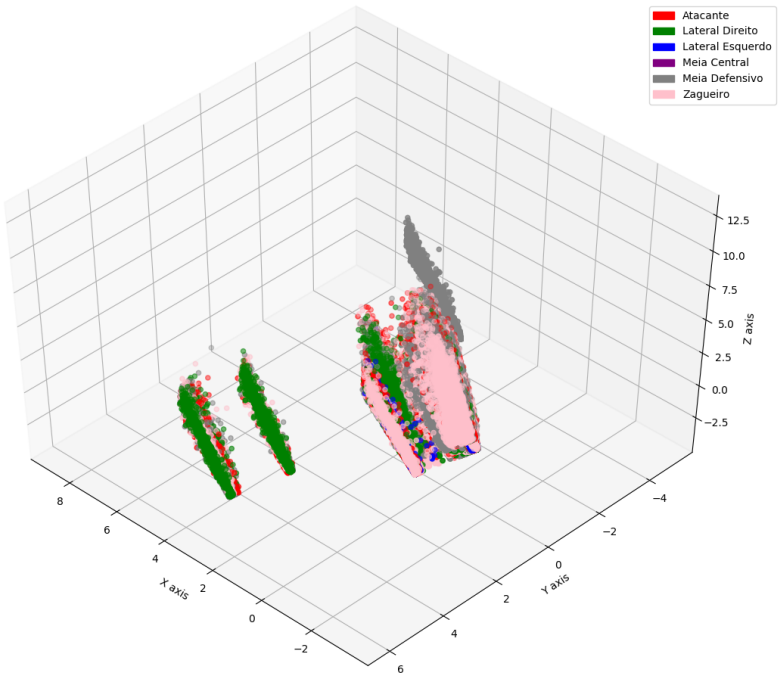

Figura 12: Todas as classes sem *outliers* reduzidas para 3 dimensões utilizando PCA

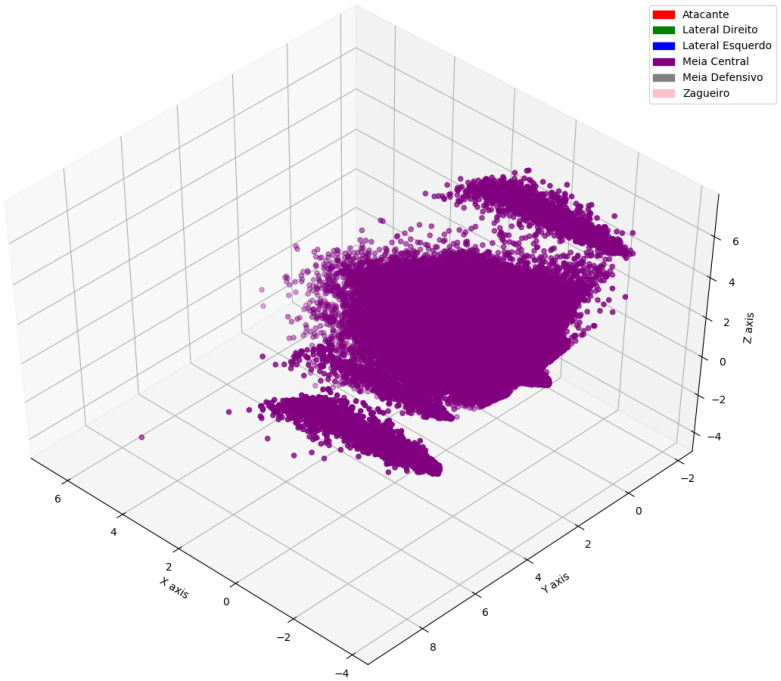

Figura 13: Classe "meia central"reduzida para 3 dimensões utilizando PCA sem *outliers*

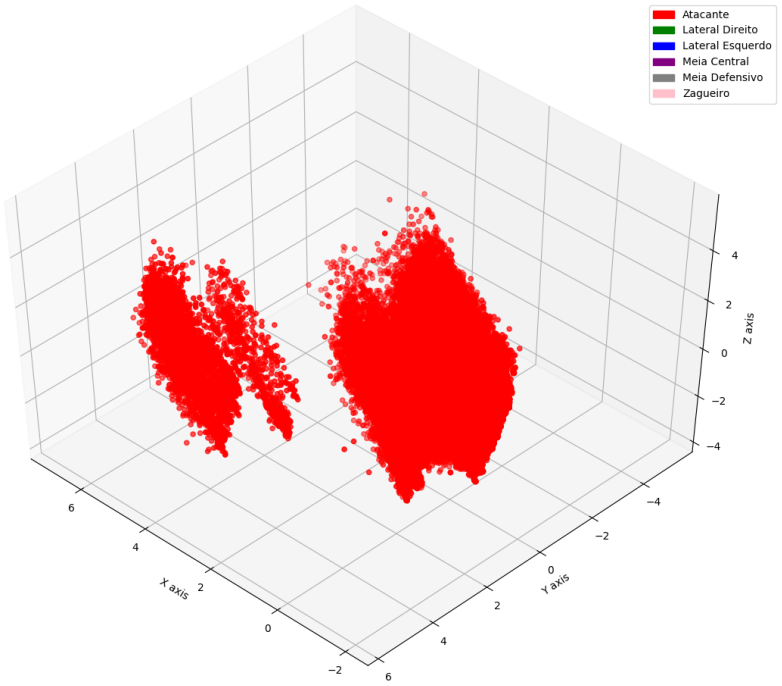

Figura 14: Classe "atacante"reduzida para 3 dimensões utilizando PCA sem *outliers*

Tabela 11: Número de tuplas das classes dos jogadores na base de dados após a eliminação dos *outliers*

Posição Rotulada	Total de Tuplas	%
Atacante	187.525	17.19
Lateral Direito	109.887	10.07
Lateral Esquerdo	31.586	2.90
Meia Atacante	0	0
Meia Central	297.617	27.27
Meia Defensivo	254.388	23.30
Zagueiro	210.407	19.27
Total de Tuplas	1.091.410	-

um total de 297.617 tuplas, a classe "lateral esquerdo"contabiliza 31.586 registros, uma diferença de aproximadamente 942.24%. Tamanha disparidade se torna uma condição favorável de tendência para o algoritmo de predição. Uma vez que o conjunto definido como "lateral esquerdo"é minoritário, é possível que as predições evitem definir novas amostras como pertencentes a essa classe, conforme detalhado na seção 2.2.3. Com o objetivo de diminuir tamanha disparidade numérica, foi realizado o balanceamento da base de dados. Foi adotado como valor ideal de tuplas por classe o número de 297.617 registros, correspondente ao "meia central". A estratégia utilizada foi a replicação dos dados reais das classes que possuíam menor número de linhas. Esse processo foi executado diretamente no banco de dados, através da interface do SGBD. De maneira aleatória, foram duplicadas as tuplas das classes minoritárias até que essas igualassem a quantidade numérica das

majoritárias. Desse modo, ao final desse processo, todas as posições rotuladas ficaram com a mesma quantidade de tuplas, ou seja, todas com 297.617, totalizando 1.785.702 registros na tabela "gps".

É relevante o fato do balanceamento ter gerado uma adição de 694.292 registros na base original, representando um acréscimo de 57.20%. Esse processo impactou os dados de maneira considerável, podendo trazer impactos positivos, ou mesmo gerar resultados com características de *underfiting* ou *overfiting*. Em busca das melhores técnicas de pré-processamento e seus resultados, foi adotado nesse estudo a aplicação dos algoritmos utilizando a base desbalanceada e balanceada, ampliando o entendimento sobre o processo como um todo.

4.9 Visualização Final dos Dados

Os dados originais foram pré-processados, conforme a lista a seguir:

- eliminação de ruídos;
- eliminação das inconsistências;
- transformação dos atributos;
- normalização por padronização;
- eliminação da classe "meia-atacante";
- eliminação dos *outliers*;
- balanceamento.

Esse processo foi importante para garantir mais consistência para os modelos de predição. A visualização final dos dados pode ser observada na figura 15 utilizando a redução PCA. Por outra perspectiva, é válido observar que após todas as etapas de pré-processamento não há uma separação clara das classes na visualização gráfica. Essa característica é um indício da complexidade da classificação que pode ser encontrada pelos algoritmos de predição, principalmente àqueles baseados em distâncias.

4.10 Considerações Finais

Este capítulo teve por objetivo apresentar a preparação da base a fim de extrair os melhores resultados dos algoritmos de aprendizado de máquina, cuja aplicação será descrita no capítulo a seguir. A eliminação de inconsistências e ruídos nos dados de entrada tendem a diminuir problemas de classificação. A transformação e geração de novos atributos podem

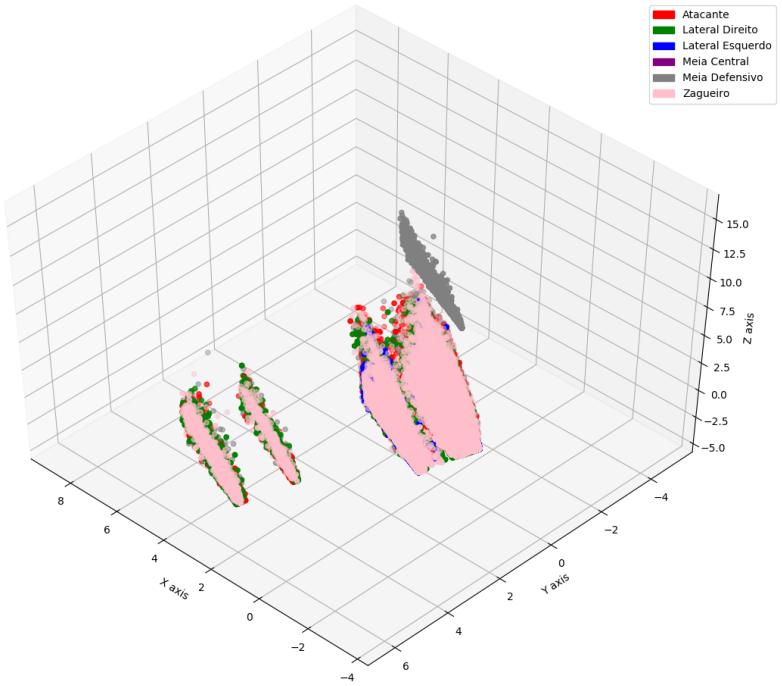

Figura 15: Todas as classes balanceadas, sem *outliers* e reduzidas para 3 dimensões utilizando PCA

favorecer e facilitar as tarefas do classificador, elucidando dados até então desconhecidos ou ocultos.

Não houve a compensação da distância do jogador para os jogos que ocorreram em estádios menores que o Maracanã. Uma vez que não foram utilizados os atributos de latitude e longitude, um possível impacto ao resultado foi minimizado, entretanto esse fator abre possibilidades para a execução de trabalhos futuros contemplando esse ajuste no posicionamento.

É importante ressaltar a grande adição de tuplas gerada pelo balanceamento. Não é possível afirmar nessa etapa, mas cabe a observação de uma mudança muito significativa no volume de dados, podendo gerar situações de *overfiting* ou *underfiting* nos modelos. Esse cenário deve ser considerado e observado nas etapas a seguir desse estudo.

5 Experimentos e Resultados

5.1 Proposta

Esse capítulo detalha a aplicação de algoritmos específicos de aprendizado de máquina. A proposta está baseada na execução dos algoritmos e coleta dos resultados utilizando a técnica validação cruzada com dez partições. A metodologia consistirá na busca pelo modelo mais indicado para o cenário proposto nesse estudo, considerando a taxa de acerto, a viabilidade da execução e a aplicação para os problemas multi-classe e binário (apenas duas classes envolvidas).

5.2 Metodologia Experimental

Todos os algoritmos foram aplicados utilizando a linguagem *Python* com suporte do banco de dados *MySQL*®. A execução ocorreu em ambiente web, utilizando para tal o servidor Apache. As versões dos *softwares* utilizados estão disponíveis na tabela 12.

Tabela 12: Versão dos *softwares* utilizados na aplicação dos algoritmos

Produto/Serviço	Versão
Apache	2.0
MySQL®	5.7.20
Python	2.7.13

A fim de evitar resultados tendenciosos, todas as execuções utilizaram a base de dados pré-processada, conforme demonstrado na seção 4. Adicionalmente, os dados foram dispostos de modo aleatório, sendo apresentados aos algoritmos desse modo.

Com o objetivo de apoiar execução dos algoritmos, a análise de dados e todo o processo aplicado nesse estudo, foi desenvolvida uma ferramenta em *Python* com *HTML*. Seu propósito específico foi agir como um facilitador das muitas execuções necessárias durante todas as etapas envolvidas. As imagens 16 e 17 demonstram a ferramenta desenvolvida e alguns dos seus recursos.

As execuções ocorreram num computador portátil com a seguinte configuração: Core i7 - 7ª geração, 8 GB RAM, 1 TB HD, placa de vídeo NVidia 2GB e sistema operacional Linux Ubuntu 16.04.

Em relação à linguagem *Python*, as principais bibliotecas utilizadas foram: *mysqldb*[1],

[1] http://mysql-python.sourceforge.net/MySQLdb.html

matplotlib[2], *numpy*[3], *mpl_toolkits*[4] e *sklearn*[5].

A execução da validação cruzada para diferentes algoritmos de aprendizado de máquina permitirá a realização das análises dos resultados obtidos, indicando os modelos com melhor ajuste para o cenário apresentado. Uma vez que as posições dos jogadores serão inferidas e, idealizando uma aplicação prática e real, as posições definidas pelos algoritmos podem ser utilizadas pelos treinador e comissão técnica das respectivas equipes de futebol. A base de dados resultante do pré-processamento possui um total de seis classes, as quais passarão pelo processo de validação cruzada. A composição desses registros será denominada como *"problema multi-classe"*. É válido ressaltar que inferir posições, considerando um problema multi-classe é relevante academicamente, uma vez que produz resultados passíveis de análise. Entretanto, é preciso considerar que a função de um treinador exige conhecimento prévio e experiência na área, portanto, raramente o mesmo será incapaz de apontar as características encontradas nos jogadores que estão sob o seu comando. Buscando aproximar esse estudo da realidade do futebol, é relevante considerar também a possibilidade da incerteza entre duas posições ideais, onde a dúvida do treinador persista e, assim, um algoritmo de aprendizado de máquina possivelmente poderá ajudá-lo nessa tarefa. Nesse trabalho, as execuções que envolverem apenas duas classes serão denominadas "problema binário".

Especificamente para a segunda abordagem não serão confrontadas todas as classes de duas em duas, mas apenas aquelas que possuam correlações, uma vez que, por serem similares, são passíveis de gerar dúvidas no treinador quanto ao posicionamento ideal do atleta.

A fim de uma melhor compreensão envolvendo todo o cenário desse estudo, serão aplicados também os algoritmos para duas classes opostas, mais especificamente atacante e zagueiro.

As classes eleitas para comparação são as que seguem:

- Lateral Direito e Lateral Esquerdo

  Ambos atuam nas laterais do campo, entretanto em lados opostos

- Meia Central e Meia Defensivo

  Ambos atuam no meio do campo, entretanto o primeiro na área central e o segundo um pouco mais recuado

- Atacante e Zagueiro

[2] https://matplotlib.org/
[3] http://www.numpy.org/
[4] http://matplotlib.org/1.4.3/mpl_toolkits/index.html
[5] http://scikit-learn.org/stable/

Figura 16: Visualização parcial da ferramenta desenvolvida com o intuito de auxiliar as execuções dos algoritmos 1/2

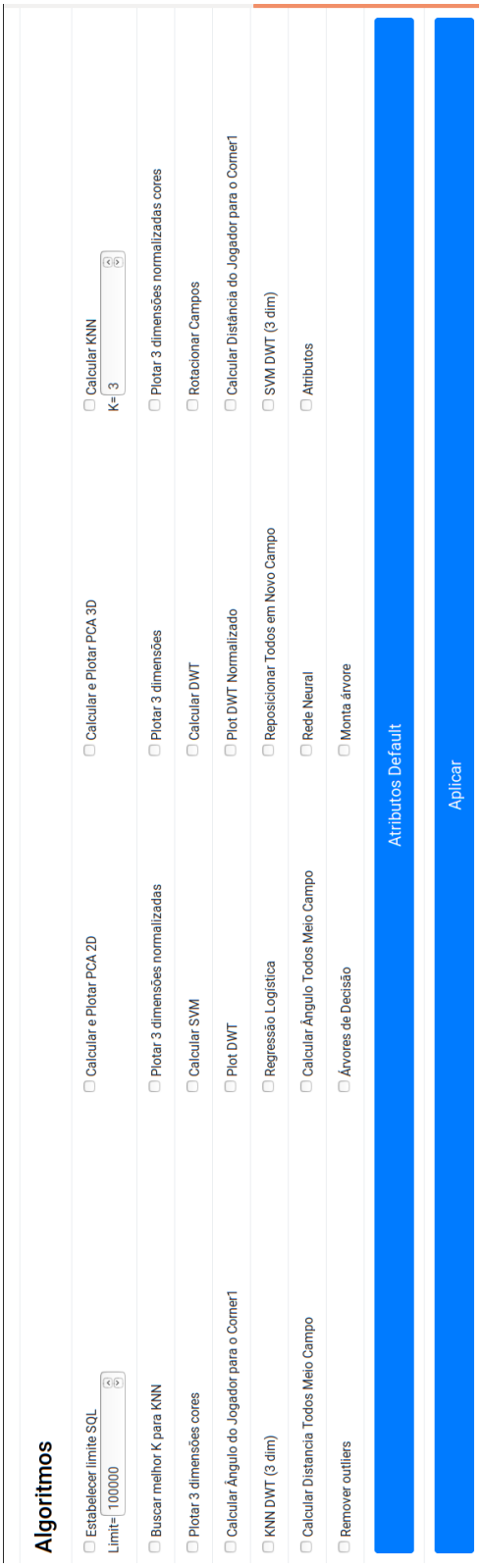

Figura 17: Visualização parcial da ferramenta desenvolvida com o intuito de auxiliar as execuções dos algoritmos 2/2

O primeiro tem característica de atacar a meta adversária, enquanto o segundo objetiva defender a sua própria meta. Realmente são posições contrárias de atuação, entretanto, academicamente pode ser interessante essa análise, possibilitando um melhor entendimento quanto ao comportamento dos algoritmos para o cenário proposto.

A tabela 13 apresenta a quantidade de tuplas envolvida em cada execução com a base desbalanceada.

Tabela 13: Quantidade de tuplas envolvida em cada execução com a base desbalanceada

Classe 1	Quantidade	Classe 2	Quantidade	Total
Lateral Direito	109.887	Lateral Esquerdo	31.586	141.473
Meia Central	297.617	Meia Defensivo	254.388	552.005
Atacante	187.525	Zagueiro	210.407	397.932

A base balanceada adotou a cardinalidade de 297.617 tuplas para cada classe, mantendo o padrão do estudo.

Os algoritmos aplicados foram k-NN, árvores de decisão, regressão logística, SVM e redes neurais.

A fim de explorar melhor as possibilidades desse estudo, todas as execuções da metodologia experimental serão realizadas adotando a base balanceada e também a desbalanceada. Essa medida foi adotada para melhor entender o comportamento dos algoritmos e seus resultados, além do impacto do balanceamento sobre o problema.

Para cada algoritmo foi criada uma ou mais funções, as quais utilizam as respectivas bibliotecas em *Python*. Outras funções auxiliam as tarefas de execução, como a seleção das amostras e rótulos, carregamento dos dados e a geração de visualizações gráficas.

Cada algoritmo utilizado nessa metodologia gerou quatro resultados distintos, uma vez que as execuções ocorreram para os problemas multi-classe e binário, com as bases balanceada e desbalanceada.

5.3 Aplicação e Resultados

São detalhados nessa seção as configurações dos algoritmos, suas execuções e os resultados obtidos.

5.3.1 Bases Distintas

Ao realizar o balanceamento, muitas amostras foram duplicadas, uma vez que foi adotada a técnica de replicação das classes minoritárias. Esse processo gerou uma

amostra idêntica fazendo com que, por muitas vezes, o melhor resultado represente o próprio elemento, ou seja, a amostra duplicada a fim de promover o balanceamento.

Uma vez que o cenário se demonstrou tendencioso com a base balanceada e existe a possibilidade, na validação cruzada, do melhor elemento ser a sua própria amostra duplicada, para todo algoritmo de aprendizado de máquina executado na metodologia experimental serão adotadas duas bases: balanceada e desbalanceada. Desse modo será possível coletar os resultados e promover um melhor entendimento sobre o impacto que o balanceamento trouxe para o problema aqui abordado.

5.3.2 k-NN

O primeiro algoritmo executado foi o k-NN. Nesse modelo é de extrema importância a escolha adequada do valor de k, uma vez que o número de vizinhos é determinante para a definição da classe. Para não incorrer na utilização de um valor aleatório de k, foi adotada a busca pelo valor ideal, promovendo a execução do algoritmo repetidas vezes com diferentes valores para k. Utilizando uma fração dos dados balanceados, o mesmo iniciou a busca assumindo o valor 3, sendo finalizado em 49. A cada execução o valor de k foi incrementado com dois. O valor ideal obtido para k nesse estudo foi 3, conforme pode ser visualizado na figura 18.

A adoção de um baixo valor para k garante que apenas seus vizinhos mais próximos sejam considerados. O valor sempre ímpar, mantém o critério de desempate. Não foram adotados pesos por distância, fazendo assim com que todos os vizinhos tenham a mesma importância na votação. Não foram realizados testes com valores diferentes para k, além de 3, uma vez que a figura 18 ilustra um menor desempenho sempre que o seu valor é aumentado.

Uma vez definido o valor ideal de k, foi iniciada a execução do algoritmo, utilizando a técnica de validação cruzada com dez partições. A taxa de acerto, utilizando a base balanceada, foi de 52.27%, consumindo aproximadamente um minuto e meio de execução. Entretanto, foi encontrado um cenário tendencioso, fazendo com que esse resultado não seja considerado confiável. A técnica de balanceamento da base consistiu na replicação das tuplas das classes minoritárias. Considerando o processo de validação cruzada e, sabendo que o k-NN baseia a sua classificação através da distância dos vizinhos mais próximos, é certo que o balanceamento torna o processo tendencioso, principalmente para esse modelo, conforme abordado anteriormente nesse estudo.

A fim de identificar o impacto que o balanceamento traz, especificamente para o k-NN, foi realizada uma nova busca pelo melhor k, ainda utilizando a base balanceada. O critério de partida foi alterado, considerando k igual a 1. O resultado obtido foi exatamente 1 para k. Desse modo, fica claro que o balanceamento impactou diretamente os resultados

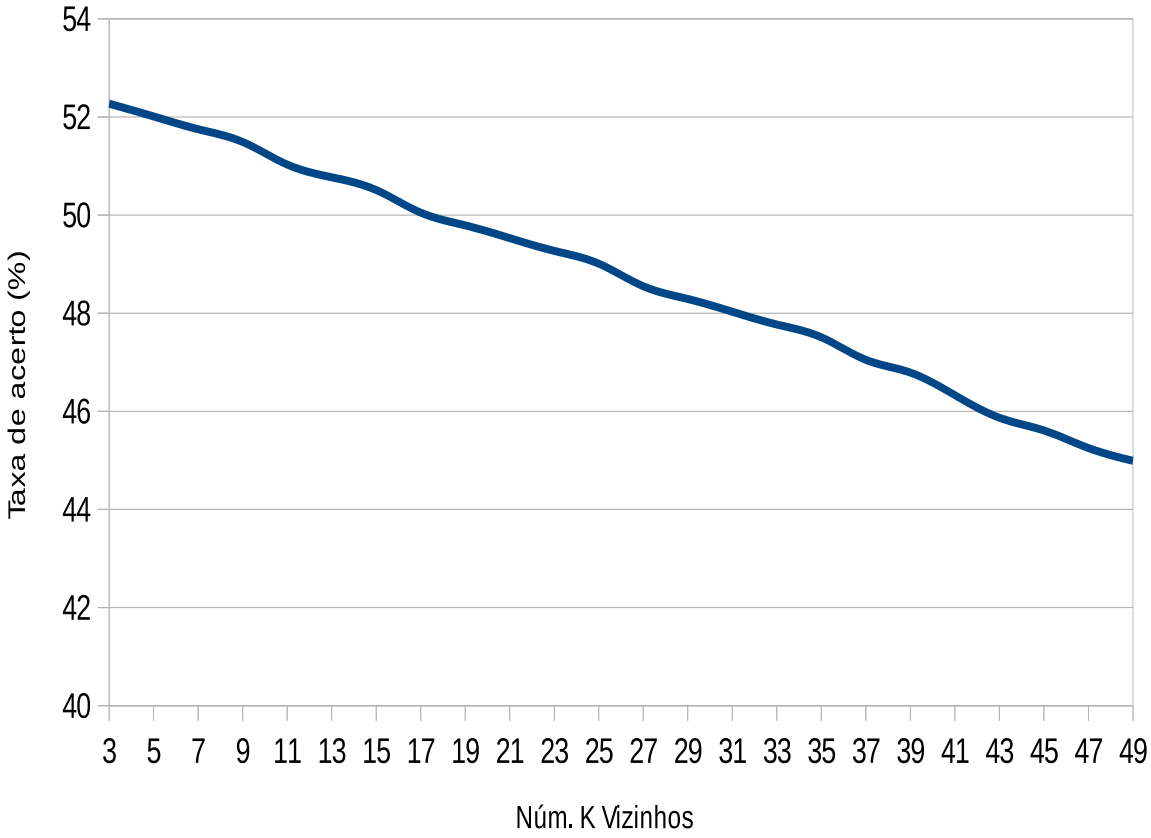

Figura 18: Busca do melhor k para o algoritmo k-NN com base balanceada iniciando com 3 vizinhos

obtidos pelo k-NN, pois a distância ideal é a própria amostra duplicada, a qual foi posteriormente calculada pela validação cruzada. A figura 19 ilustra o valor ideal de k iniciando em 1.

Para a base balanceada, o melhor k é igual a 1, tornando-o um algoritmo *1*-NN. Dessa forma, fica claro que o balanceamento tornou o modelo não confiável. Assim sendo, os resultados para a base balanceada não serão considerados na análise do k-NN.

Diante do exposto, apenas o resultado do k-NN para a base desbalanceada foi considerado. A figura 20 ilustra a busca pelo melhor k para o cenário proposto. A sua taxa de acerto, para k igual a 9, foi 41.85%, consumindo aproximadamente um minuto para a sua execução.

Apesar de um algoritmo relativamente simples, o seu tempo de execução foi muito positivo, uma vez que consumiu menos de dois minutos para ambos cenários. A taxa de acerto pode ser considerada relevante, uma vez que houve mais de 40% de acerto, considerando a existência de cinco classes distintas na base de dados em questão.

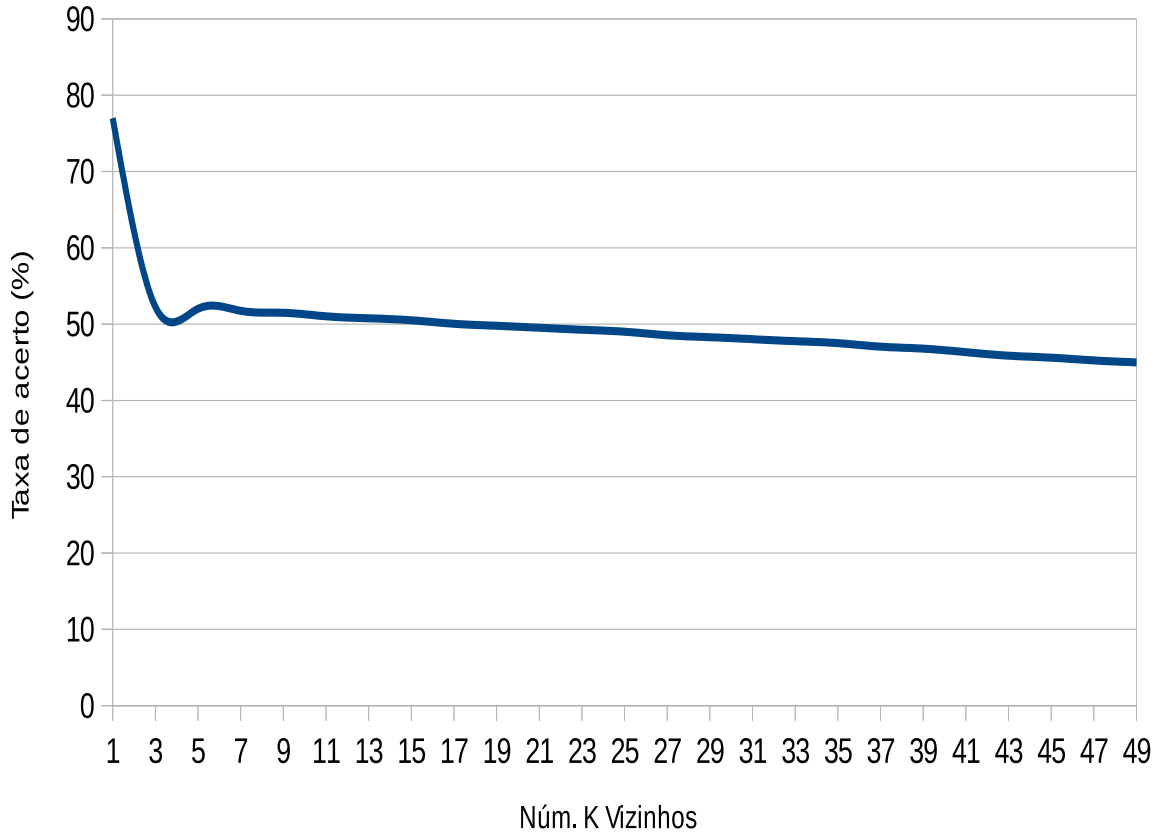

Figura 19: Busca do melhor k, iniciando em 1, para o algoritmo k-NN com base balanceada

5.3.3 Árvores de Decisão

O segundo algoritmo executado foi o de árvores de decisão. Esse modelo foi impactado pelo grande volume de registros, com isso a composição da árvore se tornou extensa. A figura 21 ilustra a sua composição, entretanto com um número reduzido de dados, objetivando apenas a ilustração gráfica através da ferramenta desenvolvida para apoiar esse estudo.

Diferentemente do cenário encontrado no k-NN, onde houve um cenário tendencioso ocasionado pelo balanceamento, os algoritmos apresentados a seguir não sofreram com essa questão de maneira tão impactante. No caso das árvores de decisão, o algoritmo executado na base balanceada foi capaz de predizer 14.80% em aproximados sete minutos. Na base desbalanceada a taxa de acerto alcançou 13.40% após aproximados cinco minutos.

As árvores propõem uma separação finita e simplificada da sua estrutura. Um problema multi-classe e com amostras semelhantes, como o apresentado, torna árdua a tarefa do algoritmo. A diferença percentual das taxas de acerto pode ser considerada pouco significativa, totalizando apenas 1.4%.

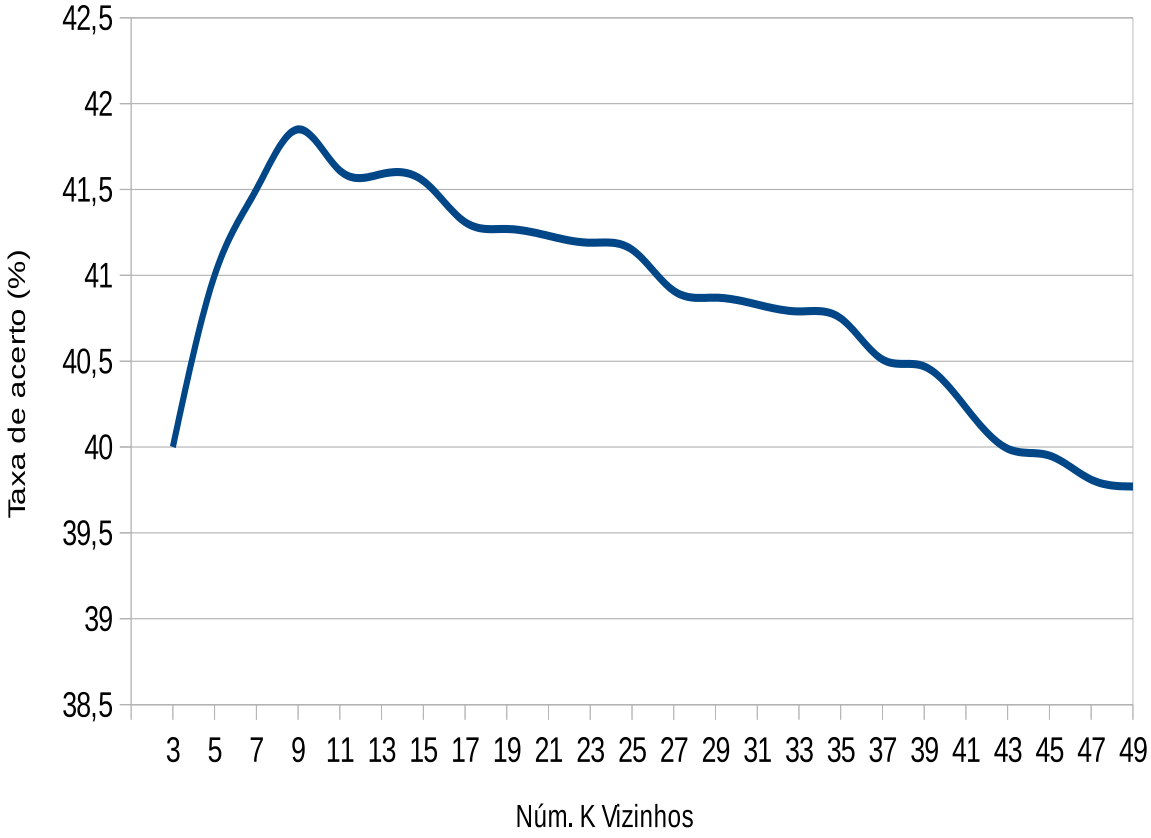

Figura 20: Busca do melhor k para o algoritmo k-NN com base desbalanceada

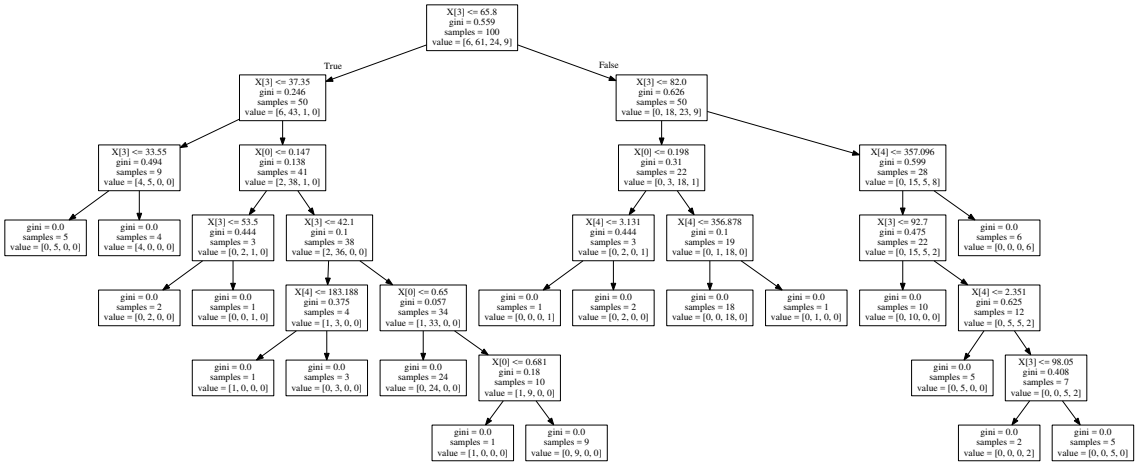

Figura 21: Demonstração reduzida da montagem da árvore de decisão utilizando a ferramenta de apoio desse estudo

A princípio, analisando exclusivamente as taxas de acerto obtidas através da validação cruzada, pode-se considerar pouco relevante os resultados retornados pelas árvores de decisão para o problema proposto. São taxas de acerto baixas para o problema analisado, pouco auxiliando na tomada de decisão.

5.3.4 Regressão Logística

A regressão logística aplicada é baseada no método estatístico, o qual busca inferir saídas categóricas. O seu modelo permite trabalhar com problemas multi-classes, como o aplicado, entretanto seus melhores resultados são obtidos em cenários limitados a duas classes, conforme disposto em 2.4.3.

Com o objetivo de ilustrar o modelo da sigmóide $f(z)$, foram selecionadas apenas duas classes[6]: atacante e zagueiro. Sabendo que o classificador assume uma saída baseada em $0 \leq f(z) \leq 1$, a figura 22 ilustra a função tomando por base os dados balanceados desse estudo.

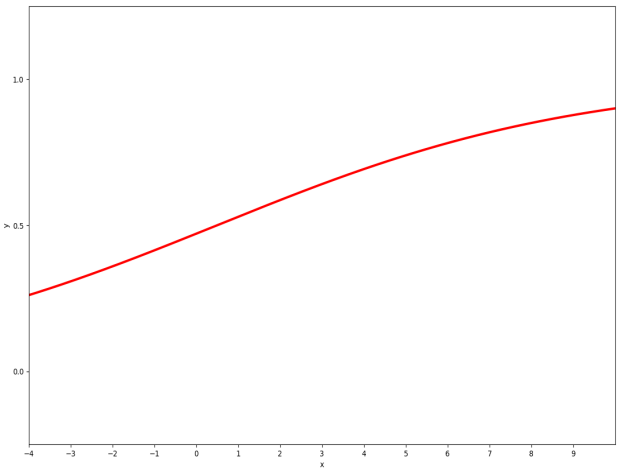

Figura 22: Gráfico da função sigmóide das classes zagueiro e atacante

Tomando por base a função sigmóide, a figura 23 exibe o espalhamento das duas classes e as suas classificações no modelo. Como é possível observar, a sobreposição é intensa, além de uma separação pouco precisa, o que leva a baixas taxas de acurácia.

Apesar de um melhor desempenho quando comparado com as árvores de decisão, a regressão logística pouco a superou. Para a base com balanceamento, o tempo consumido foi de aproximadamente dez minutos. A taxa de acerto obtida foi de 17.42%. Repetindo a execução, mas adotando a base desbalanceada, o tempo total foi de aproximadamente sete

[6] tornando mais fácil a visualização gráfica

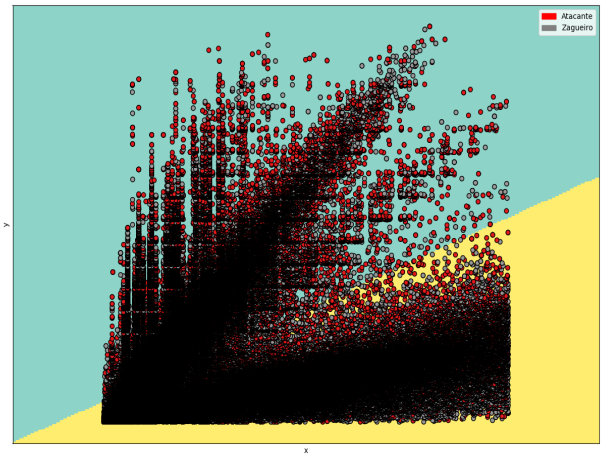

Figura 23: Separação das classes zagueiro e atacante na regressão logística

minutos, gerando uma taxa de acerto de 16.11%. Novamente é pouco notável a melhora da taxa de acerto entre as duas bases. Nesse caso, o ganho foi de 1.31%.

5.3.5 SVM

O modelo SVM exigiu um alto custo computacional, cenário já esperado, considerando que o algoritmo trabalha com problemas binários, conforme destacado na seção 2.4.4. A execução para o problema multi-classe foi possível através da repetição do processo de classificação, finalizando apenas ao término do confronto das classes, sempre em pares alternados.

A execução do algoritmo SVM foi consideravelmente extensa, consumindo aproximadamente quatorze horas e dezenove minutos. Tamanho tempo é justificável pela complexidade e escolha dos modelos adotados por ele. O resultado obtido, se comparado com as árvores de decisão e regressão logística, fez jus ao tempo, retornando uma taxa de acerto de 56.11%, para a base balanceada. Seguindo os mesmos critérios adotados anteriormente, entretanto com a base desbalanceada, a sua execução consumiu aproximadamente onze horas e quarenta e dois minutos, gerando uma taxa de acerto de 42.14%.

O *kernel* utilizado na execução do SVM foi o *RBF - Radial Basis Function*. A sua escolha, dentre outras possíveis, foi em razão deste trabalhar com números reais baseados em distância. Para calcular o intervalo entre os pontos, foi utilizado o método Euclidiano. A figura 24 demonstra a aplicação do algoritmo SVM para as classes lateral direito e lateral esquerdo, uma vez que a visualização gráfica é facilitada para problemas binários. A fim de melhorar a visualização, uma amostragem parcial foi selecionada, servindo exclusivamente como ilustração para um entendimento mais fácil da separação criada.

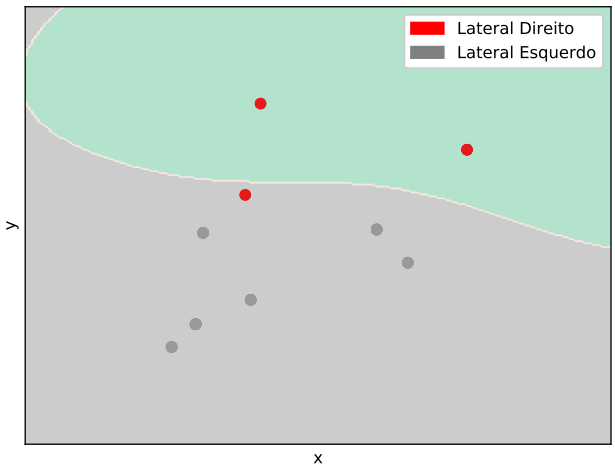

Figura 24: Separação das classes utilizando SVM com *kernel* RBF

Ao utilizar a validação cruzada como método para obtenção das taxas de acerto e, ao comparar o SVM com os demais modelos apresentados até o momento, é possível notar que o mesmo foi capaz de apresentar resultados consistentes para o problema multi-classe.

Quanto ao tempo consumido, é importante reforçar que os resultados foram obtidos através da validação cruzada. Esse método promove diversos treinamentos na base, considerando o número de partições. Num cenário de aplicação final, o treinamento ocorreria uma vez apenas, inferindo somente as amostras não rotuladas. Essa situação garante a entrega do resultado de modo direto e extremamente mais veloz.

5.3.6 Redes Neurais

Para as redes neurais com *backpropagation* foi adotado um modelo com uma camada de entrada com seis parâmetros, os quais correspondem aos atributos selecionados. São duas camadas ocultas, sendo a primeira com doze neurônios, seguida de outra com seis. A camada de saída é responsável pela definição de uma classe única. A figura 25 ilustra o leiaute da rede neural.

O leiaute escolhido foi definido após repetidas execuções com diferentes valores para as camadas. Ao analisar os resultados, a configuração em questão foi a qual obteve o melhor desempenho.

O algoritmo de redes neurais consumiu aproximadamente vinte minutos para que o seu processamento fosse totalmente finalizado utilizando a base balanceada, retornando uma taxa de acerto de 16.66%. Para a base desbalanceada o processamento consumiu aproximadamente quatorze minutos, fornecendo ao seu final uma taxa de acerto de 10.06%.

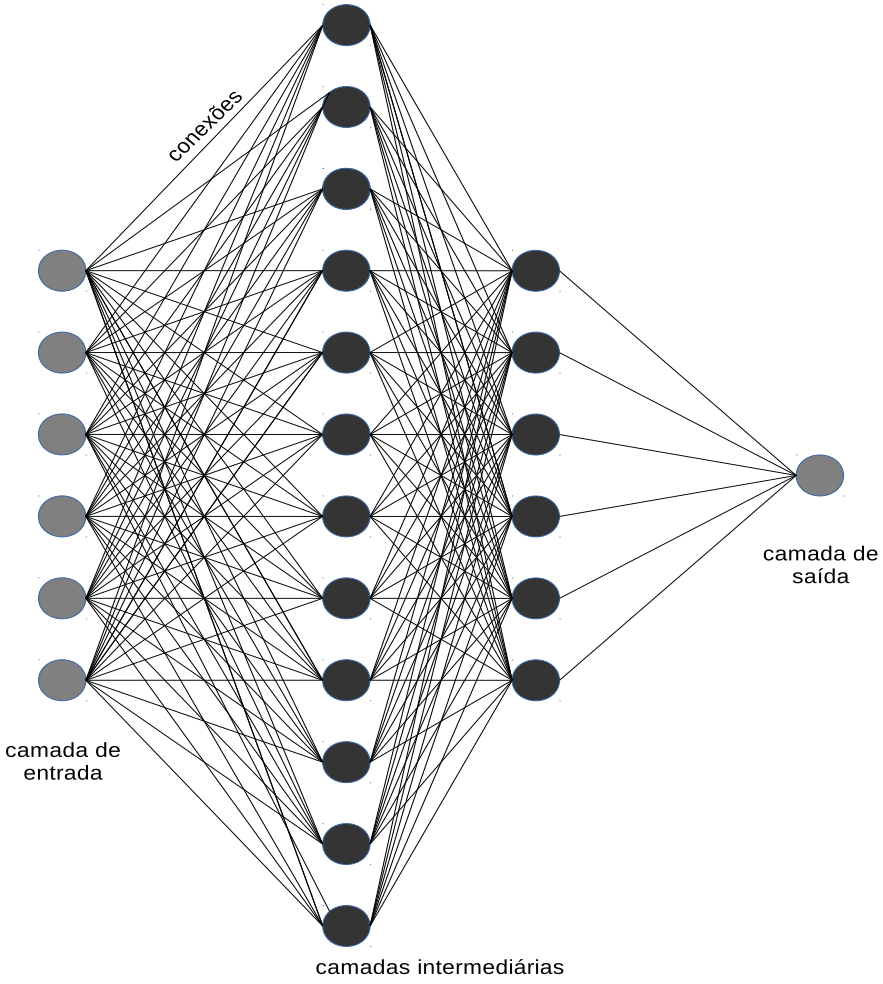

Figura 25: Rede neural utilizada na execução do algoritmo

Novamente são encontrados resultados com baixa assertividade, mas com um tempo de execução aceitável.

A tabela 14 apresenta todos os resultados coletados. O modelo SVM obteve as melhores taxas de acerto para o problema apresentado até o momento. Apesar da sua superioridade ser incontestável nesse quesito, o seu tempo de execução, aliado ao custo computacional, o tornam um modelo pouco convidativo. A sua adaptação a um problema multi-classe é possível, mas a sua eficiência foi drasticamente afetada nos cenários apresentados.

Tabela 14: Aplicação dos algoritmos para inferir a posição do jogador - multi-classe

Algoritmo	Taxa de Acerto(%)	Tempo de Execução ≈
k-NN - Balanceada	~~52.27~~	1.5 min.
k-NN - Desbalanceada	41.85	1 min
Árvores de Decisão - Balanceada	14.80	7 min.
Árvores de Decisão - Desbalanceada	13.40	5 min.
Regressão Logística - Balanceada	17.42	10 min.
Regressão Logística - Desbalanceada	16.11	7 min.
SVM - Balanceada	56.11	14 horas e 19 min.
SVM - Desbalanceada	42.14	11 horas e 42 min.
Redes Neurais - Balanceada	16.66	20 min.
Redes Neurais - Desbalanceada	10.06	14 min.

5.3.7 Problema Binário

Já destacado em 5.2, na prática seria pouco usual um questionamento total do treinador quanto à posição ideal de um jogador de futebol. Considerando que esse cargo é ocupado apenas por pessoas preparadas e com vivência no futebol, é plausível dúvidas relativas quanto ao desempenho do atleta em duas posições distintas. Uma vez que não é possível medir qualidade e adaptação através da quantidade de gols marcados por determinado jogador, modelos de aprendizado de máquina podem ser potencialmente positivos para auxiliá-lo na resposta a essa pergunta. Nesse intuito, os algoritmos foram novamente executados. A base foi limitada a apenas duas posições alvo dos jogadores, executando os algoritmos com os dados balanceados e desbalanceados.

Todos os algoritmos seguiram os mesmos padrões, modelos e valores antes executados. Os valores de k para o algoritmo k-NN também não sofreram alterações. As posições confrontadas e seus detalhamentos encontram-se na seção 5.2.

As taxas de acerto estão dispostas na tabela 15, bem como o tempo aproximado de execução.

Considerando a mesma base utilizada no estudo como um todo, mas realizando uma análise binária das posições dos jogadores, os resultados coletados apresentam taxas de acertos mais expressivas, conforme demonstrado na tabela 15. A última coluna contém o valor médio do tempo de execução, considerando cada posição confrontada.

Os valores obtidos para o k-NN utilizando a base balanceada foram novamente descartados, pois continuam tendenciosos e, por consequência, não são confiáveis.

Ao analisar as colunas que confrontam as posições dos jogadores na tabela 15, fica evidente que, para todos os algoritmos utilizados na metodologia experimental, a melhor taxa de acerto é obtida na comparação entre lateral direito e lateral esquerdo.

Tabela 15: Taxas de acerto dos algoritmos para inferir a posição do jogador - classes binárias

	Lat Dir. x Esq.	Meia Cent. x Def.	Atacante x Zagueiro	Tempo de Execução Médio≈
k-NN - Balanceada	~~97.29~~%	~~65.36~~%	~~68.96~~%	0.7 min.
k-NN - Desbalanceada	91.21%	64.76%	66.31%	0.5 min.
Árvores de Decisão - Balanceada	63.91%	40.85%	35.86%	4.5 min.
Árvores de Decisão - Desbalanceada	67.39%	41.63%	35.46%	3 min.
Regressão Logística - Balanceada	62.76%	49.78%	51.60%	0.5 min.
Regressão Logística - Desbalanceada	77.67%	53.61%	53.10%	0.4 min.
SVM - Balanceada	96.36%	59.25%	60.33%	6 horas e 23 min.
SVM - Desbalanceada	91.50%	59.27%	60.54%	4 horas e 47 min.
Redes Neurais - Balanceada	71.28%	58.51%	54.87%	5 min.
Redes Neurais - Desbalanceada	78.16%	57.63%	55.29%	3.5 min.

Considerando que as duas posições atuam em lados opostos do campo e metade dos atributos são relacionados ao posicionamento, é possível a existência de uma situação tendenciosa. Entretanto, ao analisar a dinâmica de jogo e o fato da base de dados carregar informações do primeiro e segundo tempo, onde as equipes alternam os lados do campo, faz com que haja uma equivalência de informações, conforme demonstrado na figura 26.

O ângulo e distância relativos ao *corner_1* são consistentes, uma vez que ocorre a troca de lado pelas equipes e as posições dos laterais são invertidas. A distância do meio de campo também é mantida, não induzindo os resultados, mesmo se tratando de atributos de localização.

Em relação às demais posições, os laterais atuam de modo mais consistente, uma vez que alternam entre defesa, meio e ataque, mas sempre juntos às linhas laterais. A criação desse corredor virtual faz com que os dados sofram variações menores, mas com uma separação clara entre as classes, explicando a superioridade na taxa de acerto. É preciso considerar também os demais atributos, os quais se baseiam em velocidade e distância, uma vez que laterais se deslocam mais e com menor intensidade. Ao utilizar todos os atributos, mesclando velocidade e posicionamento, os algoritmos foram capazes de predizer com maior exatidão se o jogador possui características de atuação na lateral esquerda ou direita (SALVO et al., 2007; SCAGLIA et al., 1996).

Os resultados obtidos para a aplicação dos algoritmos entre meia central e meia defensivo obtiveram taxas menores de acerto. Diferentemente do que ocorre com os laterais,

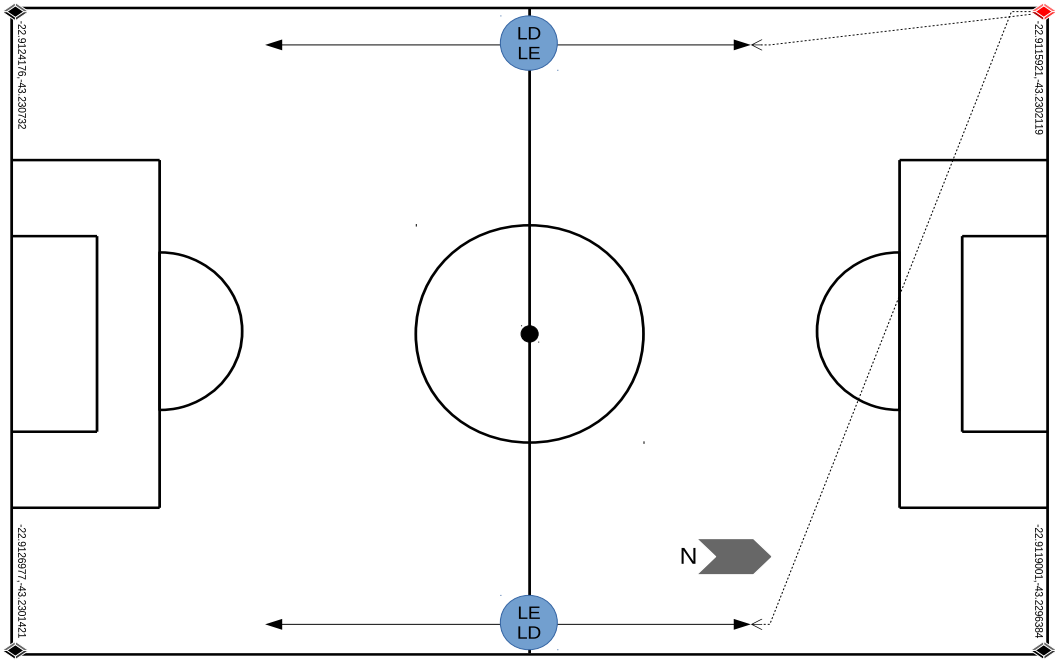

Figura 26: LE - Lateral Esquerco, LD - Lateral Direito. Equivalência de ângulo e posicionamento para os laterais

são posições centrais. Desse modo, nem sempre é possível guardar uma área delimitada, conforme demonstrado na tabela 6. O meio de campo é uma região compartilhada, sendo que as ações são tomadas pelos jogadores conforme o andamento da partida. Assim, a predição dos algoritmos foi afetada, uma vez que três atributos são baseados em posicionamento.

Apesar da obtenção de uma taxa de acerto menor, os resultados obtidos variaram entre 64.76% e 40.85%. Tomando por base o melhor resultado, é possível considerá-lo um acerto relevante, principalmente se comparado com o problema multi-classe.

A última aplicação ocorreu entre zagueiro e atacante. Apesar de atuarem em lados opostos do campo e com objetivos diferentes, ambos possuem algumas características semelhantes. As duas posições têm atuação direto com a meta, sendo um a favor e outro contra. Costumam percorrer menores distâncias em campo, entretanto as suas explosões musculares são maiores. Assim, os dados gerados para essas duas posições distintas possuem semelhanças, o que torna a tarefa da predição mais difícil (SALVO et al., 2007; SCAGLIA et al., 1996).

As taxas de acerto variaram entre 66.31% e 35.46%, conforme disposto na tabela 15. O melhor resultado traz relevância para esse estudo, principalmente ao considerar que são posições onde os dados gerados possuem similaridades.

Ao analisar o tempo de execução para os problemas multi-classe e binário, o SVM foi o algoritmo que gerou maior custo computacional dentre todas as execuções. É

preciso considerar também que, a técnica utilizada foi a validação cruzada. Esse processo é exaustivo, uma vez que ocorrem repetidos treinos com a base. Considerando um cenário de predição específico, o treinamento ocorreria apenas uma vez, ficando disponível para novas predições, as quais ocorrem em menor tempo. Entretanto, na proposta apresentada nesse estudo, a técnica *10*-fold foi computacionalmente custosa para o SVM, quando considerado o tempo consumido pelos demais algoritmos.

Outra comparação importante a ser analisada é o fator balanceamento. No pré-processamento foi adotada essa técnica, entretanto a mudança de cardinalidade foi muito considerável. Foi adotada a execução duplicada de todos os algoritmos, sempre com as bases balanceada e desbalanceada, permitindo entender melhor o comportamento e o quanto de ganho ou perda foi computado.

Para o problema multi-classe, o balanceamento agiu de modo positivo. Todos os resultados foram melhores quando adotada a técnica. O ganho médio entre os algoritmos considerados foi de 5.82%.

No problema binário houve prejuízos quando o balanceamento foi adotado. Desse modo, para o problema binário aplicado na metodologia experimental, não é recomendado o balanceamento. A única exceção aplica-se ao algoritmo SVM, o qual obteve desempenho superior ou relativamente igual à base desbalanceada.

Considerando que o problema binário se aproxima mais da realidade e que o desbalanceamento foi, de modo geral, desfavorável, a sua adoção não é indicada nesse estudo, alinhando assim com a orientação de que o balanceamento pode não trazer ganhos positivos em circunstâncias alinhadas com a realidade (BATISTA; PRATI; MONARD, 2004).

Em uma situação onde fosse condizente a aplicação do problema multi-classe, o SVM com balanceamento teria a melhor taxa de acerto. Com um resultado 14.26% menor, o k-NN desbalanceado seria o segundo algoritmo mais indicado. Entre os eleitos é preciso ressaltar que o primeiro consumiu mais de quatorze horas, contra apenas um minuto e meio do segundo. Apesar de uma diferença considerável da taxa de acerto, o custo computacional é muito discrepante entre os modelos.

Para a proposta desse estudo, considerando a base de dados utilizada, o pré-processamento aplicado e todas as etapas envolvidas, as quais foram descritas ao decorrer dos capítulos, o algoritmo que demonstra ser o mais indicado é o k-NN sem balanceamento da base. O segundo melhor é o SVM com balanceamento, entretanto é válido reforçar que o custo computacional é muito divergente entre os dois modelos.

A proposta abordou dois problemas diferentes: multi-classe e binário. Para ambos, os dois melhores algoritmos foram o k-NN e SVM. Isso indica uma tendência de melhor adaptação dos modelos ao cenário encontrado nesse estudo.

A nível de complexidade, especificamente para esse estudo e para a base adotada, apesar de adaptativos, os modelos são divergentes. O k-NN é baseado na técnica de distância. Isso o torna extremamente simples e eficaz, uma vez que os atributos selecionados possuem correlações de posicionamento, os quais influenciam no espalhamento dos dados. Já o SVM é uma técnica aprimorada e que demanda alto processamento, o que de fato ocorreu na seção 5.2. O fato de ser complexa a torna capaz de lidar com problemas dessa natureza. O *kernel* utilizado foi o RBF, o qual também é baseado em distância e pode ter favorecido o resultado positivo. Foi um modelo que se adaptou bem, entregando resultados expressivos.

5.4 Ameaças à Validade

A realização desse trabalho buscou eliminar ao máximo dados tendenciosos, algoritmos com parametrização incorreta, dentre outras ameaças à validade. Apesar desse cuidado, existem situações que não puderam ser eliminadas no contexto desse estudo, como influência do padrão devido às características antropométricas dos jogadores, possíveis falhas de leitura no equipamento GPS, amostras rotuladas incorretamente pelo treinador e falha no cadastramento do campo onde ocorreu a partida.

5.5 Considerações Finais

A ferramenta web que foi desenvolvida permitiu a utilização das bibliotecas em Python, garantindo um maior número de testes e variações, buscando os melhores ajustes nos algoritmos e, consequentemente taxas de acerto mais expressivas. A escolha dessa linguagem foi acertada, uma vez que se demonstrou preparada e com recursos suficientes para todas as execuções que se fizeram necessárias.

Os resultados foram positivos, pois vieram ao encontro da proposta, onde foi possível coletar resultados. A aplicação dos algoritmos, mais especificamente, demonstrou quais caminhos podem ser seguidos. Dos cinco executados, dois se destacaram em suas taxas de acerto, a saber: k-NN e SVM.

O k-NN, apesar de ser um algoritmo simples e fácil, todas as amostras precisam ser calculadas a cada predição, o que pode tornar o processo muito custoso quando adotadas bases de dados com alta cardinalidade.

O SVM, apesar de exigir um alto custo computacional para o seu treinamento, o mesmo está apto para inferir novas amostras após essa operação. Desse modo, novas predições ocorrem de modo rápido, uma vez que um novo treinamento é dispensável.

Esse capítulo também deixou clara a importância de tratar o problema num escopo bem definido. Foram baixas as taxas de acerto para o problema multi-classe, se comparados com o problema binário.

6 Conclusão e Trabalhos Futuros

Análises por GPS e por vídeo, de modo geral, são difundidas e amplamente adotadas no meio esportivo profissional. O aprendizado de máquina, por sua vez, vem em uma ascendente nos últimos anos. O volume de trabalhos publicados é muito considerável para as duas áreas. Esse estudo teve como desafio a união dessas duas vertentes, buscando entender a viabilidade e os resultados alcançados na predição do posicionamento ideal dos jogadores profissionais do futebol brasileiro.

Esse trabalho foi centrado no questionamento sobre qual algoritmo de aprendizado de máquina obtém a melhor taxa de acerto para inferir a posição tática de um jogador profissional do futebol brasileiro. A fim de encontrar a resposta mais confiável, considerando o cenário proposto, diversas técnicas e modelos foram aplicados.

A execução dos algoritmos, a qual de fato é a responsável pela entrega dos resultados, foi ajustada para a proposta desse estudo. Entretanto, os modelos sofreram impacto direto pelo pré-processamento, uma vez que novos atributos foram gerados a partir dos iniciais.

Dos seis atributos utilizados nos algoritmos, metade foi gerado a partir do pré-processamento. Esse é um fator que sinaliza a importância da etapa de preparação da base. A obtenção desses novos atributos apenas foi possível após a análise do problema e da base de dados, identificando, inicialmente, a necessidade de remapear os campos onde ocorreram os jogos e reposicionar os jogadores em um único campo. Esse processo não foi suficiente para entender o comportamento dos jogadores, uma vez que não era possível identificar em qual lado o atleta estava atuando, pois não havia a identificação de primeiro e segundo tempo do jogo. A solução para esse problema foi justamente a criação de novos atributos. Esses permitiram gerar variáveis independentes sem correlação com o lado de atuação.

Analisando o estudo como um todo, fica evidente a relevância da etapa de pré-processamento, a qual consumiu a maior parte do tempo. A execução dos algoritmos de aprendizado de máquina não foram menos importantes, entretanto utilizaram bibliotecas específicas, sendo promovidas pequenas parametrizações a fim de alinhá-los com o problema.

Desse modo, considerando o cenário encontrado, a proposta traçada e todas as aplicações executadas, é possível concluir que, especificamente nesse estudo, a etapa de pré-processamento é tão ou mais impactante do que a aplicação dos algoritmos de aprendizado de máquina.

A abordagem multi-classe, considerando a proposta específica desse estudo, não obteve êxito. Além das baixas taxas de acerto, o problema tratado não condiz com a

realidade do futebol profissional brasileiro.

Analisando as taxas de acerto obtidas no problema binário, os resultados obtidos atingiram níveis satisfatórios e que podem ser considerados condizentes com a proposta. Os algoritmos k-NN e SVM obtiveram as melhores taxas de acerto.

Uma vez que os melhores desempenhos foram alcançados com os modelos k-NN e SVM com *kernel* RBF, é importante evidenciar que ambos trabalharam suas predições baseadas em distâncias (Euclidianas). Essa é uma característica importante presente nos dois algoritmos e pode ser responsável por taxas de acerto mais elevadas.

Quanto ao balanceamento realizado no pré-processamento e testado por toda a execução, as melhoras variaram com o problema e com o algoritmo executado. Algumas vezes o balanceamento trouxe ganhos aos resultados, sendo que em outras fez com que a taxa de acerto fosse impactada negativamente. É possível observar que é um ganho pouco significativo, mas não existem elementos suficientes para recomendar ou refutar a utilização dessa técnica nesse trabalho.

Uma vez que não foram utilizados os atributos de latitude e longitude nesse estudo, os prejuízos nos resultados foram minimizados quanto à falta de compensação do tamanho dos campos para o reposicionamento no Maracanã. Sugere-se então um trabalho futuro onde a diferença seja calculada e compensada, permitindo um melhor entendimento dos resultados.

O estudo, como um todo, atingiu o seu objetivo, uma vez que entendeu o problema, definiu estratégias - as quais permitiram a geração de novas informações relevantes, determinou etapas de pré-processamento da base, executou os algoritmos propostos e obteve resultados passíveis de análise.

6.1 Contribuições

O desenvolvimento desse trabalho foi centrado em duas vertentes principais: a abordagem acadêmica e a solução de um problema real ligado à uma empresa. Esse cenário, por si só, é interessante, uma vez que demonstra ser possível a união da academia com o mundo empresarial, podendo haver cooperação e resultados positivos para ambos. Essa é a primeira contribuição desse trabalho, demonstrando ser positiva essa integração.

Considerando que os atributos constantes na base de dados bruta não eram suficientes, algumas técnicas foram aplicadas a fim de gerar novos dados, os quais auxiliaram em melhores taxas de acerto. Desse modo, houve uma contribuição quanto às abordagens que permitiram a obtenção de novos atributos.

Ficou claro também que a tratativa de um problema multi-classe pode não ser relevante no futebol profissional brasileiro. Os resultados demonstram baixa taxa de acerto

quando todas as posições de jogadores são incluídas nas predições da validação cruzada. Como contribuição, recomenda-se a comparação de classes limitando-se a um problema binário.

Dos algoritmos executados no problema binário, segundo as taxas de acerto obtidas, recomenda-se a utilização dos algoritmos k-NN ou SVM. É válido destacar que, especificamente para esse estudo e utilizando a validação cruzada, o primeiro modelo é mais simples de ser implantado e a sua execução consome apenas algumas dezenas de segundos.

Finalmente, e não menos importante, uma das contribuições muito significativas desse trabalho está relacionada à possibilidade de replicação desse estudo para os cenários acadêmico e empresarial. Qualquer empresa que trabalhe com um cenário parecido com o exposto pode replicar as técnicas e os fundamentos aqui apresentados.

6.1.1 Periódicos

Durante a pesquisa, aplicações e escrita desse trabalho foi realizada a publicação de um artigo na Revista Brasileira de Computação Aplicada. O mesmo faz uma abordagem similar ao apresentado nesse estudo, entretanto em menor escala e com menos técnicas aplicadas. A referência está disposta a seguir:

GASPARINI, Randal; ÁLVARO, Alexandre. Análise entre algoritmos de aprendizado de máquina para suportar a predição do posicionamento do jogador de futebol. Revista Brasileira de Computação Aplicada, [S.l.], v. 9, n. 2, p. 70-83, jul. 2017. ISSN 2176-6649.

6.2 Limitações e Trabalhos Futuros

Os resultados obtidos podem ser considerados satisfatórios considerando todo o cenário apresentado. Entretanto, no decorrer do trabalho algumas limitações foram identificadas, as quais estão elencadas abaixo:

- Base de dados única

  Todas as execuções foram realizadas com uma base de dados única. Esse fato pode tornar todo o processo tendencioso ou pouco adaptativo a outros cenários.

- Execução de um número limitado de algoritmos

  Atualmente existe um grande número de algoritmos de aprendizado de máquina, os quais utilizam diferentes técnicas. Foram selecionados cinco modelos específicos, uma vez que era necessário definir o escopo desse trabalho, considerando o tempo hábil para a execução de todas as tarefas. Uma vez que o número de algoritmos de aprendizado de máquina limitou-se a cinco, é possível que resultados diferentes sejam

alcançados com a execução de outros modelos ou através da adoção de parâmetros diferentes, além dos demonstrados aqui.

O trabalho aqui apresentado abordou um conjunto limitado de modelos, o qual permitiu a obtenção de algumas respostas específicas. Há possibilidade de continuidade do projeto, buscando entender melhor o contexto e aprimorar as técnicas. Alguns dos trabalhos futuros que podem ser desenvolvidos são listados a seguir:

- Balanceamento dos dados

  O balanceamento aplicado demonstrou, de modo geral, pouca eficácia. A utilização de outras técnicas de balanceamento podem gerar ganhos mais consistentes.

- Geração de Atributos

  A geração de novos atributos abordou questões específicas, entretanto é possível que novas análise resultem em diferentes variáveis independentes, as quais podem carregar correlações mais importantes.

- Aplicação de algoritmos

  A aplicação dos algoritmos de aprendizado de máquina foi limitada a cinco, mas existem outros modelos que podem se adaptar melhor ao problema.

- Reposicionamento dos jogadores

  O reposicionamento dos jogadores foi realizado mediante a uma série de cálculos, realocando todos no Maracanã. Não houve a compensação relativa às diferenças das metragens dos campos de origem e destino. Um trabalho futuro é possível de ser realizado a fim de melhor entender o impacto das diferenças dos tamanhos do campo e as técnicas para a realocação dos jogadores num único campo.

- Criação de algoritmo específico

  Considerando que o k-NN foi um dos algoritmos com melhor taxa de acerto e sua complexidade é relativamente baixa, é plausível idealizar um novo modelo de predição baseado nesse, o qual poderia sofrer ajustes a fim de melhor se adaptar ao cenário.

- Busca por novas repostas

  O trabalho apresentado se concentrou em predizer a posição ideal de um jogador. Considerando a riqueza da base de dados, novas informações importantes podem ser extraídas da mesma. Abordagens diferentes das propostas desse estudo também são possíveis, como análises coletivas, análises de posicionamento dos jogadores por setores, detecção de sobrecarga física, dentre outros.

Referências

AUGHEY, R. J.; FALLOON, C. Real-time versus post-game gps data in team sports. *Journal of Science and Medicine in Sport*, v. 13, n. 3, p. 348 – 349, 2010. ISSN 1440-2440. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1440244009001005>. Citado na página 50.

BARBERO-ÁLVAREZ, J. C. et al. The validity and reliability of a global positioning satellite system device to assess speed and repeated sprint ability (rsa) in athletes. *Journal of Science and Medicine in Sport*, Elsevier, v. 13, n. 2, p. 232–235, 2010. ISSN 1440-2440. Disponível em: <http://dx.doi.org/10.1016/j.jsams.2009.02.005>. Citado 2 vezes nas páginas 26 e 48.

BARROS, R. M. et al. Analysis of the distances covered by first division brazilian soccer players obtained with an automatic tracking method. *Journal of Sports Science and Medicine*, p. 233–242, 2007. ISSN 1303-2968. Disponível em: <http://hdl.handle.net/11449/69706>. Citado 3 vezes nas páginas 47, 48 e 49.

BATISTA, G. E. A. P. A.; PRATI, R. C.; MONARD, M. C. A study of the behavior of several methods for balancing machine learning training data. *SIGKDD Explor. Newsl.*, ACM, New York, NY, USA, v. 6, n. 1, p. 20–29, jun. 2004. ISSN 1931-0145. Disponível em: <http://doi.acm.org/10.1145/1007730.1007735>. Citado 3 vezes nas páginas 33, 55 e 87.

BATISTA, G. E. d. A. P. *Pré-processamento de dados em aprendizado de máquina supervisionado*. Tese (Doutorado) — Universidade de São Paulo, 2003. Disponível em: <http://www.teses.usp.br/teses/disponiveis/55/55134/tde-06102003-160219/en.php>. Citado 3 vezes nas páginas 33, 34 e 35.

BEN-GAL, I. Outlier detection. In: *Data mining and knowledge discovery handbook*. [S.l.]: Springer, 2005. p. 131–146. Citado na página 32.

BOURKE, A. The dream of being a professional soccer player. *Journal of Sport and Social Issues*, v. 27, n. 4, p. 399–419, 2003. Disponível em: <http://dx.doi.org/10.1177/0193732503255478>. Citado na página 25.

BURGES, C. J. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, v. 2, n. 2, p. 121–167, 1998. ISSN 1573-756X. Disponível em: <http://dx.doi.org/10.1023/A:1009715923555>. Citado 4 vezes nas páginas 15, 38, 39 e 43.

CARLING, C. et al. The role of motion analysis in elite soccer. *Sports Medicine*, v. 38, n. 10, p. 839–862, 2008. ISSN 1179-2035. Disponível em: <http://dx.doi.org/10.2165/00007256-200838100-00004>. Citado na página 46.

CARVALHO, A. Redes neurais artificiais. *ICMC*, v. 29, p. 05–09, 2005. Citado 2 vezes nas páginas 15 e 40.

DALLAWAY, N. Movement profile monitoring in professional football. July 2014. Disponível em: <http://etheses.bham.ac.uk/5044/>. Citado 3 vezes nas páginas 26, 49 e 50.

DAOLIO, J. As contradições do futebol brasileiro. *Futebol: paixão e política. Rio de Janeiro: DP&A*, p. 29–44, 2000. Disponível em: <http://www.educadores.diaadia.pr.gov.br/arquivos/File/2010/artigos_teses/EDUCACAO_FISICA/artigos/contradicoes-do-futebol.pdf>. Citado 2 vezes nas páginas 25 e 26.

D'OTTAVIO, S.; CASTAGNA, C. Analysis of match activities in elite soccer referees during actual match play. *The Journal of Strength & Conditioning Research*, LWW, v. 15, n. 2, p. 167–171, 2001. Citado 3 vezes nas páginas 17, 47 e 48.

EDGECOMB, S.; NORTON, K. Comparison of global positioning and computer-based tracking systems for measuring player movement distance during australian football. *Journal of Science and Medicine in Sport*, v. 9, n. 1, p. 25 – 32, 2006. ISSN 1440-2440. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1440244006000053>. Citado 2 vezes nas páginas 49 e 50.

FACELI, K. et al. *Inteligência artificial: uma abordagem de aprendizado de máquina*. Grupo Gen - LTC, 2011. ISBN 9788521618805. Disponível em: <https://books.google.com.br/books?id=4DwelAEACAAJ>. Citado 13 vezes nas páginas 15, 31, 32, 33, 34, 36, 37, 38, 39, 40, 41, 43 e 54.

GEVERT, V. G. et al. Modelos de regressão logística, redes neurais e support vector machine (svm s) na análise de crédito a pessoas jurídicas. *RECEN-Revista Ciências Exatas e Naturais*, v. 12, n. 2, p. 269–293, 2011. ISSN 2175-5620. Citado 3 vezes nas páginas 38, 40 e 43.

GIL, S. M. et al. Physiological and anthropometric characteristics of young soccer players according to their playing position: relevance for the selection process. *The Journal of Strength & Conditioning Research*, LWW, v. 21, n. 2, p. 438–445, 2007. ISSN 1064-8011. Disponível em: <http://journals.lww.com/nsca-jscr/Fulltext/2007/05000/PHYSIOLOGICAL_AND_ANTHROPOMETRIC_CHARACTERISTICS.26.aspx>. Citado na página 25.

GONÇALVES, E. B. *Análise de risco de crédito com o uso de modelos de regressão logística, redes neurais e algoritmos genéticos*. Tese (Doutorado) — Universidade de São Paulo, 2005. Citado 3 vezes nas páginas 38, 40 e 43.

HAIR, J.; ANDERSON, R. T. *RL: BLACK, WC Análise Multivariada de Dados. 5ª edição*. [S.l.]: Porto Alegre: Bookman, 2005. Citado 2 vezes nas páginas 37 e 63.

HENNIG, E.; BRIEHLE, R. Game analysis by gps satellite tracking of soccer players. *Archives of Physiology and Biochemistry*, SWETS ZEITLINGER PUBLISHERS PO BOX 825, 2160 SZ LISSE, NETHERLANDS, v. 108, n. 1-2, p. 44–44, 2000. Citado na página 48.

KOHAVI, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. In: *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1995. (IJCAI'95), p. 1137–1143. ISBN 1-55860-363-8. Disponível em: <http://dl.acm.org/citation.cfm?id=1643031.1643047>. Citado na página 42.

MACHADO, E. L. Um estudo de limpeza em base de dados desbalanceada e com sobreposiçao de classes. 2007. Disponível em: <http://repositorio.unb.br/handle/10482/1397>. Citado na página 32.

MEYER, T.; OHLENDORF, K.; KINDERMANN, W. Longitudinal analysis of endurance and sprint abilities in elite german soccer players. *Deutsche Zeitschrift für Sportmedizin*, v. 7, n. 8, p. 271–277, 2000. ISSN 2510-5264. Disponível em: <https://www.researchgate.net/publication/282684850_Longitudinal_analysis_of_endurance_and_sprint_abilities_in_elite_German_soccer_players>. Citado na página 47.

MITCHELL, E.; MONAGHAN, D.; O'CONNOR, N. E. Classification of sporting activities using smartphone accelerometers. *Sensors*, v. 13, n. 4, p. 5317–5337, 2013. ISSN 1424-8220. Disponível em: <http://www.mdpi.com/1424-8220/13/4/5317>. Citado 2 vezes nas páginas 48 e 49.

OKAZAKI, V. H. A. et al. Ciência e tecnologia aplicada à melhoria do desempenho esportivo. *Revista Mackenzie de Educação Física e Esporte*, v. 11, n. 1, 2012. ISSN 1980-6892. Disponível em: <http://cev.org.br/biblioteca/ciencia-tecnologia-aplicada-melhoria-desempenho-esportivo>. Citado na página 25.

PILA, A. D. *Seleção de atributos relevantes para aprendizado de máquina utilizando a abordagem de Rough Sets*. Tese (Doutorado) — Universidade de São Paulo, 2001. Disponível em: <http://www.teses.usp.br/teses/disponiveis/55/55134/tde-13022002-153921/en.php>. Acesso em: 29 out. 2017. Citado 2 vezes nas páginas 31 e 35.

PROZONE. *Find your Sports Data*. 2016. Disponível em: <http://prozonesports.stats.com>. Acesso em: 11 set. 2016. Citado 2 vezes nas páginas 47 e 49.

RAUBER, T. W. Redes neurais artificiais. 1997. Disponível em: <https://inf.ufes.br/~thomas/pubs/eri98.pdf>. Acesso em: 03 abr. 2017. Citado na página 39.

RODRIGUES, V.; VIEIRA, F.; SILVA, I. Mineração de dados para estimativas de mortalidade pré-abate de frangos de corte. *Archivos de Zootecnia*, scieloes, v. 62, p. 469 – 472, 09 2013. ISSN 0004-0592. Disponível em: <http://scielo.isciii.es/scielo.php?script=sci_arttext&pid=S0004-05922013000300015&nrm=iso>. Citado na página 37.

RODRIGUEZ-AÑEZ, C. R. A antropometria e sua aplicação na ergonomia. *Revista Brasileira de Cineantropometria & Desenvolvimento Humano*, v. 3, n. 1, p. 102–108, 2001. ISSN 1415-8426. Disponível em: <http://portalbiocursos.com.br/ohs/data/docs/51/20-_A_ANTROPOMETRIA_E_SUA_APLICAYYO_NA_ERGONOMIA.pdf>. Citado na página 25.

ROWEIS, S. T.; SAUL, L. K. Nonlinear dimensionality reduction by locally linear embedding. *Science*, American Association for the Advancement of Science, v. 290, n. 5500, p. 2323–2326, 2000. Citado na página 26.

SAFAVIAN, S. R.; LANDGREBE, D. A survey of decision tree classifier methodology. *IEEE Transactions on Systems, Man, and Cybernetics*, v. 21, n. 3, p. 660–674, May 1991. ISSN 0018-9472. Citado na página 37.

SALVO, V. D. et al. Performance characteristics according to playing position in elite soccer. *International journal of sports medicine*, Georg Thieme Verlag KG Stuttgart, New York, NY, USA, v. 28, n. 3, p. 222–227, March 2007. ISSN 0172-4622. Disponível em: <https://doi.org/10.1055/s-2006-924294>. Citado 4 vezes nas páginas 25, 47, 85 e 86.

SCAGLIA, A. J. et al. Escolinha de futebol: uma questão pedagógica. *Motriz*, v. 2, n. 1, p. 36–43, 1996. ISSN 1980-6574. Disponível em: <http://www.periodicos.rc.biblioteca.unesp.br/index.php/motriz/article/view/6513>. Acesso em: 03 abr. 2017. Citado 4 vezes nas páginas 17, 53, 85 e 86.

SEGATTO-MENDES, A. P.; SBRAGIA, R. O processo de cooperação universidade-empresa em universidades brasileiras. *Revista de Administra&ccdeil; ão da Universidade de São Paulo*, v. 37, n. 4, October 2002. ISSN 0080-2107. Disponível em: <https://www.researchgate.net/publication/311426283_O_processo_de_cooperacao_universidade-empresa_em_universidades_brasileiras>. Citado na página 26.

SIENKIEWICZ-DIANZENZA, E.; RUSIN, M.; STUPNICKI, R. Resistência anaeróbica de jogadores de futebol. *Fitness & performance journal*, Colégio Brasileiro de Atividade Física, Saúde e Esporte, n. 3, p. 199–203, 2009. ISSN 1676-5133. Disponível em: <https://dialnet.unirioja.es/descarga/articulo/2977271.pdf>. Citado na página 47.

VINCENTY, T. Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations. *Survey Review*, Taylor and Francis, v. 23, n. 176, p. 88–93, 1975. Disponível em: <http://dx.doi.org/10.1179/sre.1975.23.176.88>. Citado na página 59.

WOLD, S.; ESBENSEN, K.; GELADI, P. Principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, v. 2, n. 1, p. 37 – 52, 1987. ISSN 0169-7439. Proceedings of the Multivariate Statistical Workshop for Geologists and Geochemists. Disponível em: <http://www.sciencedirect.com/science/article/pii/0169743987800849>. Citado na página 63.

WU, X. et al. Top 10 algorithms in data mining. *Knowledge and Information Systems*, v. 14, n. 1, p. 1–37, Jan 2008. ISSN 0219-3116. Disponível em: <https://doi.org/10.1007/s10115-007-0114-2>. Citado na página 36.

YEH, T. K. et al. Construction and uncertainty evaluation of a calibration system for gps receivers. *Metrologia*, v. 43, n. 5, p. 451, 2006. Disponível em: <http://stacks.iop.org/0026-1394/43/i=5/a=017>. Citado na página 54.