Machine Learning aplicado na previsão de resultados de rodadas de Counter Strike Global Offensive.

Juan Victor Leal de Moraes

[1]Instituto de Ensino Superior(iCEV)
R. Dr. José Auto de Abreu, 2929 – 64055-260 – São Cristóvão, Teresina – Piauí – Brasil

Abstract. *Digital games, especially Counter-Strike Global Offensive, are conquering more and more the public and the competitive scenario, moving more than 14 million dollars, in addition to the money that circulates through the scenario, there are several variables to be used, thus conquering not only game lovers but also dice lover. Therefore, this project seeks to analyze most of the possible variables to predict which team has the best chance of achieving victory in a round of the match.*

Resumo. *Os jogos digitais, em especial o Counter Strike Global Offensive, vêm a cada dia conquistando mais o público e o cenário competitivo, movimentando mais de 14 milhões de dólares, além do dinheiro que circula pelo cenário há diversas variáveis para serem utilizadas, conquistando assim não só os amantes de jogos mas também como amante de dados. Dessa forma este projeto busca analisar grande parte das variáveis possíveis para prever qual equipe possui mais chance de alcançar a vitória em uma rodada da partida.*

1. Introdução

Este projeto inclui duas áreas que estão cada vez mais em ascensão e conectadas: esportes eletrônicos e análise de dados. Ao explorar a crescente quantidade de dados gerado diariamente por partidas e campeonatos, este projeto visa a análise preditiva de rodadas de partidas de Counter Strike Global Offensive no contexto do mercado de apostas.

1.1. Contexto

O foco deste projeto será a análise de rodadas de partidas de Counter Strike Global Offensive(CSGO). As partidas competitivas de CSGO são divididas em 30 rodadas , de dois minutos de duração cada, vence o time que alcançar 16 rodadas ganhas, caso haja empate de 15 a 15 é realizado uma prorrogação com 6 rodadas, o primeiro que ganhar 4 rodadas na prorrogação ganha a partida. Um time de CSGO é formado por 5 jogadores que jogam a partida por completo.

O Counter Strike Global Offensive foi lançado em 21 de agosto de 2012, quando conseguiu tirar o foco do Counter Strike 1.6(lançado em 15 de setembro de 2003), outro jogo da franquia que popularizou bastante em campeonatos locais de bairros e até globais, mesmo sendo um sucesso não se popularizou logo de cara, mas nos anos seguintes iniciou sua ascensão, principalmente com os campeonatos mundiais(majors).

1.2. Problemática

Com avanços da tecnologia e na área de análise de dados tem se permitido que diversos problemas de diversas áreas ganhem novas soluções. Com esse viés, o uso de análise de dados se tornou essencial nos esportes, tanto tradicionais como digitais, apresentando um enorme crescimento. Constantemente são desenvolvidos diversas novas técnicas e treinos para melhorar a performance do jogador e seu time como um conjunto. O *machine learning*, ou aprendizado de máquina, é a área que busca prever eventos futuros por meio do aprendizado passado por eventos anteriores.

O Counter Strike Global Offensive(CSGO) foi e ainda é um grande sucesso em meio ao mundo dos jogos eletrônicos. Desde o seu antecessor, o Counter Strike 1.6, que o "CS"não era apenas um jogo online, mas também uma fonte de renda para os jogadores profissionais, movimentando mais de 14 milhões de dólares em 366 torneios em 2021, e como em qualquer outro esporte as apostas em partidas e campeonatos surgiram. E com o surgimento das apostas, torna-se necessário obter bons resultados para potencializar seus lucros cada vez mais, tornando o *machine learning* um ótimo aliado para realizar previsões e usar como base para apostas.

1.3. Objetivos

O objetivo principal desse artigo é usufruir de três algoritmos de *machine learning* para prever os resultados de rodadas de partidas de CSGO, proporcionando futuramente usufruir das previsões para realizar apostas mais conscientes no cenário competitivo.

A precisão da previsão é um ponto crucial para o projeto. Encontrar a melhor combinação possível de variáveis para obter um modelo com a melhor precisão possível. Dessa forma é necessário um conhecimento bem profundo sobre a problemática a fim de realizar a melhor escolha das variáveis, para assim, superar as expectativas em questão ao resultado.

Por fim, assim como a precisão a velocidade da previsão é um ponto importante para o projeto, pois como as rodadas duram no máximo 2 minutos, é importante uma maior agilidade para representar os resultados. Aumentando assim o desafio em escolher o modelo com bons resultados e no menor tempo possível.

2. Metodologia

2.1. Compreensão dos dados

Os dados são a a base para qualquer projeto de *machine learning*, então coletar dados com alta qualidade e em uma quantidade suficiente se torna essencial para garantir o sucesso do algoritmo. Graças a comunidade e amantes do Counter Strike, temos uma vasta base de dados para ser explorados, facilitando a análise dos dados.

Como fonte de dados, neste projeto irei utilizar a base de dados *CS:GO Round Winner Classification* disponível para livre acesso pela plataforma *Kaggle*, criado no ano de 2020, possuindo mais de 500 partidas de campeonatos de alto nível registradas, partes da partida que não são interessantes para o estudo(como aquecimentos e rodadas reiniciadas) não foram utilizados na análise.

O entendimento dos dados utilizados se torna essencial para o sucesso do projeto, para isso é necessário entender todas as variáveis que estão sendo trabalhadas no aprendizado da máquina, fazendo necessário ter total controle de o que representa cada atributo considerado, como representado na Tabela 1.

Atributo	Descrição
Tempo Restante	Tempo restante da rodada
Pontos CT	Quantidade de pontos da equipe Contra terrorista
Pontos TR	Quantidade de pontos da equipe Terrorista
Mapa	Mapa da rodada
Bomba plantada	Se a bomba foi plantada ou não
Vida dos CT	Total de vida da equipe Contra Terrorista
Vida dos TR	Total de vida da equipe Terrorista
Coletes CT	Quantos da equipe Contra Terrorista usam colete
Coletes TR	Quantos da equipe Terrorista usam colete
Capacetes CT	Quantos Contra Terroristas tem capacete
Capacetes TR	Quantos Terroristas tem capacete
Dinheiro CT	Total de dinheiro da equipe Contra Terrorista
Dinheiro TR	Total de dinheiro da equipe Terrorista
Kits desarme	Quantos CTs tem kits para desarmar a bomba
CTs vivos	Quantos CTs vivos ao fim da rodada
TRs vivos	Quantos TRs vivos ao fim da rodada
Armas CTs	Armas compradas pelos CTs
Armas TRs	Armas compradas pelos TRs
Granadas CTs	Granadas compradas pelos CTs
Granadas TRs	Granadas compradas pelos TRs

Tabela 1. Métricas usadas da base de dados

2.2. Machine Learning

2.2.1. Árvores de Decisão

As árvores de decisão funcionam de um modo bem intuitivo, dividem o conjunto inicial de dados em subconjuntos mais homogêneos e que são divididos em outro subconjunto posteriormente. São compostas por nós e folhas, o primeiro nó é considerado o ponto de partida da árvore e após ele cada nó posterior representa um teste específico no conjunto de dados para dividir em subconjuntos menores e mais homogêneos, as folhas representam o conjunto final e mais homogêneo que a árvore consegue alcançar, por isso não são mais divididas. Alguns algoritmos podem produzir árvores binárias, onde cada nó interno se ramifica somente em dois outros nós, onde outros podem produzir árvores não-binárias.

Como pontos fortes da árvore de decisão pode-se citar que são bem intuitivas e de fácil interpretação, tornando uma ótima ferramenta para a análise dos dados, além de aceitar vários tipos de variáveis. Em contrapartida, como pontos fracos, são mais propensas ao *overfitting*, já que podem produzir árvores muito grandes e complicadas que modelam um treino perfeito, mas não atendem ao modelo real.

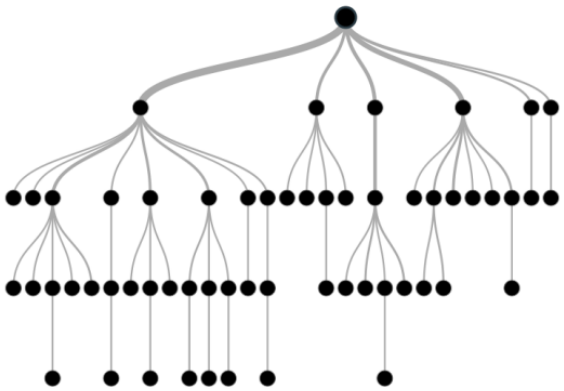

Figura 1. Exemplo de estrutura da árvore de decisão.

2.3. Regressão Logística

A regressão logística é um algoritmo de classificação que conta com uma alta interpretação dos resultados. Para modelar a probabilidade de um evento ser afetado por uma ou mais variáveis a regressão logística utiliza a função logística. A função logística mapeia as entradas em valores de 0 e 1, assim permitindo que se interprete as observações em probabilidade.

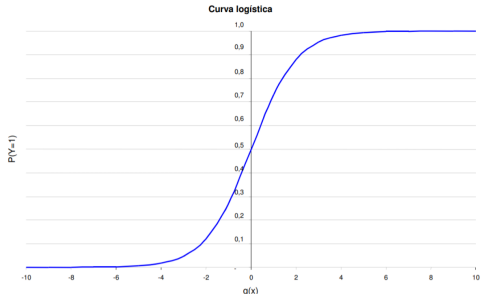

Figura 2. Curva realizada pela função logística.

2.4. Máquina de Vetores de Suporte

Máquina de suporte de vetores(SVM, do inglês *support vector machines*) é um algoritmo de aprendizagem supervisionada, não-linear, onde o conjunto de treino é passado como pontos no espaço, mapeado de forma em que os exemplos de cada classe sejam separados em um espaço o mais amplo possível. Em outras palavras, o SVM encontra uma linha de separação, chamada de hiperplano, entre a observação de cada classe.

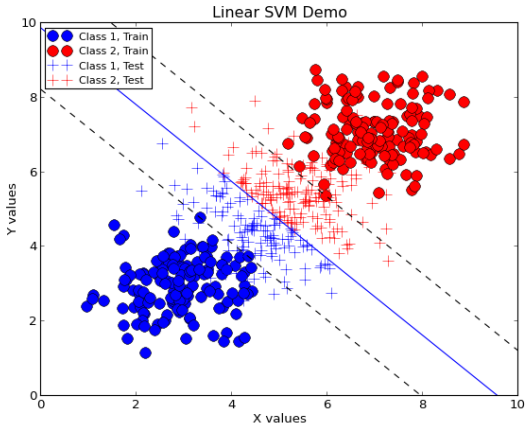

Figura 3. Exemplo do algoritmo SVM.

2.5. Avaliação do Resultado

Para avaliarmos a eficácia de nosso modelo, consideramos a acurácia obtida, que é a proporção de dados que foram previstos corretamente previstos, sejam eles verdadeiros positivos ou verdadeiros negativos. Verdadeiros positivos são quando o modelo prevê o resultado esperado, assim como também há o falso positivo onde o modelo prevê um resultado diferente do esperado, o mesmo vale para o verdadeiro negativo e falso negativo. Em meio ao *machine learning* é comum que ocorra o *overfitting*, que é quando a acurácia está bastante alta, mas só representa que o modelo se sai muito bem com dados do treinamento mas não tanto em relação a dados de teste, que são dados que ele nunca teve contato. Para prevenir o overfitting vamos explorar a acurácia do treinamento e do teste. Para realizar o treinamento e testes dos modelos dividimos o conjunto de dados em 2 conjuntos, o conjunto de treino, que representa 70% do conjunto total, e o conjunto de testes, que representa 30% do conjunto de dados total. Com isso obtemos uma maior precisão no momento da avaliação dos resultados.

2.5.1. Resultados com Árvores de Decisão

Com o modelo da árvore de decisão obtivemos um resultado bastante satisfatório, obtendo uma acurácia de 0,97(97%) com os dados de treinamento, e uma acurácia de 0,81(81%) comparando os resultados gerados com o esperado. Executando o modelo em um tempo de 44 segundos.

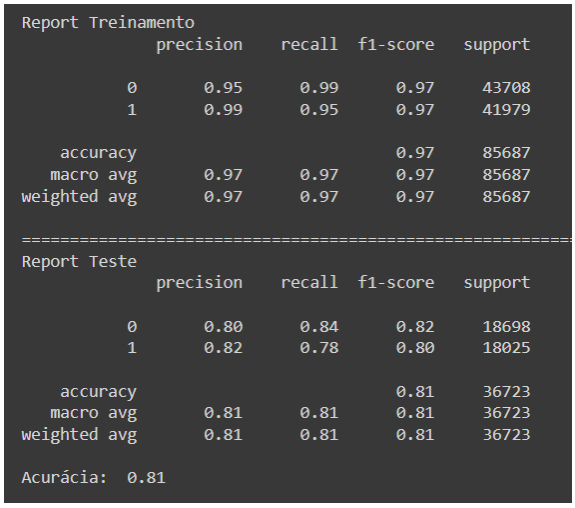

Figura 4. Dados do Classification Report da árvore de decisão.

2.5.2. Resultados com Regressão Logística

Realizando o modelo de Regressão Logística obtivemos um resultado de uma acurácia de 0,75(75%) com os dados de treinamento e acurácia de 0,75(75%) com os dados de teste. Levando um tempo de 5 segundos para executar o modelo.

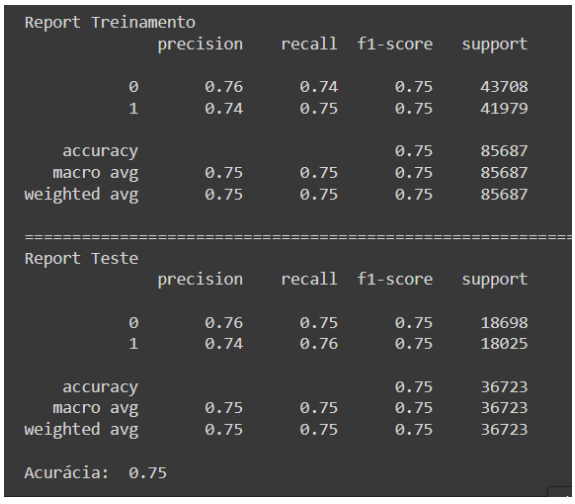

Figura 5. Dados do Classification Report da Regressão Logística.

2.5.3. Resultados com SVM

Com o modelo de Máquina de vetores de suporte obtivemos um resultado considerável, obtendo uma acurácia de 0,81(81%) com os dados de treinamento, e uma acurácia de 0,79(79%) comparando os resultados gerados com o esperado. Executando o modelo em um tempo de 40 minutos.

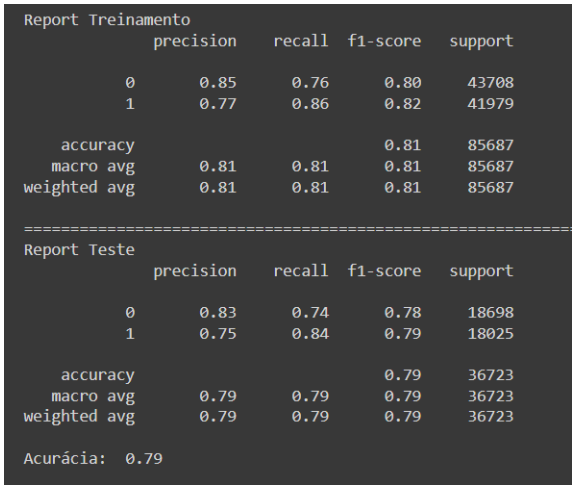

Figura 6. Dados do Classification Report do SVM.

3. Conclusões

O projeto utilizando os modelos de árvores de decisão, regressão logística e máquina de vetores de suporte, mostraram todos resultados otimistas e consideráveis na previsão do resultado esperado. Com a Tabela 2 podemos comparar melhor os resultados obtidos.

Modelo	Acurácia	Tempo Gasto
Árvores de Decisão	81%	44s
Regressão Logística	75%	5s
SVM	79%	40m

Tabela 2. Comparativo de resultado dos modelos

Comparando os resultados obtidos podemos observar que o modelo SVM pode ser descartado, pois como a agilidade é um ponto crucial para nosso projeto o seu tempo de 40 minutos não se encaixa nesse requisito. Com isso concluímos que os modelos de árvores de decisão e regressão logística se encaixam bem nos requisitos, pois possuem um bom tempo de execução e uma acurácia aceitável, ficando a critério de qual projeto irão ser utilizados para uma melhor decisão de qual modelo é mais adequado.

4. Referências

MCKINNEY, Wes. Python para análise de dados: Tratamento de dados com Pandas, NumPy e IPython. Novatec Editora, 2019.

GÉRON, Aurélien. Mãos à Obra: Aprendizado de Máquina com Scikit-Learn & TensorFlow. Alta Books, 2019.

LUNELLI, Lucas Mariani. Previsão de resultado de jogos da NBA com algoritmos de machine learning. 2020. Tese de Doutorado.