UNIVERSIDADE ESTADUAL PAULISTA
"JÚLIO DE MESQUITA FILHO"
Câmpus de Presidente Prudente

BRUNO DE LIMA TELES

APLICAÇÃO DO ALGORITMO NAIVE BAYES PARA ESTIMAR DECISÕES EM UM TORNEIO DE POKER ONLINE

PRESIDENTE PRUDENTE
2023

BRUNO DE LIMA TELES

APLICAÇÃO DO ALGORITMO NAIVE BAYES PARA ESTIMAR DECISÕES EM UM TORNEIO DE POKER ONLINE

Relatório final para Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Estatística da FCT/Unesp para aproveitamento na disciplina TCC.
Orientadora: Profa. Dra. Miriam Rodrigues Silvestre.

PRESIDENTE PRUDENTE
2023

T269a

Teles, Bruno de Lima
Aplicação do algoritmo Naive Bayes para estimar decisões em um torneio de poker online / Bruno de Lima Teles. -- Presidente Prudente, 2023
36 p. : il., tabs., fotos

Trabalho de conclusão de curso (Bacharelado - Estatística) - Universidade Estadual Paulista (Unesp), Faculdade de Ciências e Tecnologia, Presidente Prudente
Orientadora: Miriam Rodrigues Silvestre

1. Naive Bayes. 2. Poker. 3. Aprendizagem de Máquina. 4. Aprendizagem Estatística. 5. Machine Learning. I. Título.

Sistema de geração automática de fichas catalográficas da Unesp.
Biblioteca da Faculdade de Ciências e Tecnologia, Presidente Prudente.
Dados fornecidos pelo autor(a).

Essa ficha não pode ser modificada.

TERMO DE APROVAÇÃO

BRUNO DE LIMA TELES

APLICAÇÃO DO ALGORITMO NAIVE BAYES PARA ESTIMAR DECISÕES EM UM TORNEIO DE POKER ONLINE

Relatório Final de Trabalho de Conclusão de Curso aprovado como requisito para obtenção de créditos na disciplina Trabalho de Conclusão do curso de graduação em Estatística da Faculdade de Ciências e Tecnologia da Unesp, pela seguinte banca examinadora:

Orientadora: _____

Profa. Dra. Miriam Rodrigues Silvestre

Departamento de Estatística

_____

Prof. Dr. Fernando Antonio Moala

Departamento de Estatística

_____

Prof. Dr. Sérgio Minoru Oikawa

Departamento de Estatística

Presidente Prudente, 4 de dezembro de 2023.

RESUMO

Padrões. Por toda história o ser humano é apaixonado em buscar padrões e correlações que procurem explicar a natureza, o mundo e o universo. E não é diferente nos dias atuais. Quando há uma identificação de um padrão no meio esportivo, automaticamente há uma vantagem competitiva. Até que ponto nossas decisões em um esporte podem ser previstas? É possível um modelo de aprendizagem estatística identificar padrões estratégicos e comportamentais dos jogadores em um torneio de Poker online sem a informação das cartas em jogo? Entende-se que o Poker não é um jogo de azar devido a suas inúmeras possibilidades de estratégias. Há uma vasta literatura que nos apresenta diversas formas lucrativas de se competir em um torneio de Poker, estudos que contribuíram para que, em 2010, o jogo fosse considerado um esporte mental. Um jogador experiente não toma decisões apenas baseando-se em suas cartas e nas cartas da mesa, o Poker é mais complexo do que apostar quando temos cartas boas e desistir quando não as temos. Aliando o conhecimento do esporte que temos hoje com um poderoso algoritmo de aprendizagem estatística chamado Naive Bayes, trabalharemos em estimar com a maior assertividade possível as decisões de um jogador de Poker sem a informação das cartas em jogo. O quão previsível pode ser um jogador para conseguir ser decifrado por um modelo estatístico?

Palavras-chave: Poker, Aprendizagem Estatística, Aprendizagem de Máquina, Naive Bayes, Decisão, Probabilidade, Teorema de Bayes, Modelo de Classificação, Métodos de Reamostragem.

ABSTRACT

Standards. Throughout history, human beings have been passionate about seeking patterns and correlations that seek to explain nature, the world and the universe. And it is no different today. When there is an identification of a standard in sports, there is automatically a competitive advantage. To what extent can our decisions in a sport be predicted? Is it possible for a statistical learning model to identify strategic and behavioral patterns of players in an online poker tournament without information on cards in play? It is understood that Poker is not a game of chance due to its countless possibilities of strategies. There is a vast literature that presents us with several profitable ways to compete in a Poker tournament, studies that contributed to, in 2010, the game being considered a mental sport. An experienced player doesn't make decisions just based on his cards and the cards on the board, Poker is more complex than betting when we have good cards and folding when we don't. Combining the knowledge of the sport that we have today with a powerful statistical learning algorithm called Naive Bayes, we will work on estimating the decisions of a Poker player as accurately as possible without information about the cards in play. How predictable can a player be to be deciphered by a statistical model?

Keywords: Poker, Statistical Learning, Machine Learning, Naive Bayes, Decision, Probability, Bayes Theorem, Classification Model, Resampling Methods

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE SIGLAS

SL – Statistical Learning
ML – Machine Learning
SB – Small Blind
BB – Big Blind
UTG – Under the Gun
MP1 – Middle Position 1
MP2 – Middle Position 2
MP3 – Middle Position 3
HJ – Hijack
CT – Cut Off
BT – Button (Dealer)

SUMÁRIO

1 INTRODUÇÃO

Em 2010, a IMSA (International Mind Sports Association) considerou o Poker como um “esporte da mente” (ESPN, 2010), assim como o Xadrez. O jogo consiste em apostas, o objetivo é perder o menor número de fichas possíveis e maximizar seu lucro adquirindo as fichas de seus oponentes. Há quem diga que o Poker não é um jogo de cartas e sim um jogo de fichas.

O seguinte estudo baseia-se no modo de jogo mais jogado no mundo: o *No Limit Texas Hold’em*. Será utilizado como amostra um torneio online no formato *Sit in Go* de nove jogadores pelo aplicativo *Pokerstars* (Figura 1).

Cada jogador recebe duas cartas fechadas e até cinco cartas podem ser viradas na mesa, o chamado “bordo” (*board*). O jogador que fizer a melhor combinação possível dentre suas cartas e as cartas da mesa ou fizer com que todos os adversários desistam de disputar o pote, leva o pote. Um ranking de combinações possíveis de cartas define o ganhador da rodada.

Figura 1 – Mesa de Poker Online

Fonte: *Pokerstars* - Editado pelo Autor (2023)

Cada decisão que um jogador toma influencia na sua longevidade dentro de um torneio. Dependendo da situação em que se encontra, o jogador pode optar por uma das cinco decisões: passar a vez (*check*), apostar (*bet*), desistir (*fold*), pagar (*call*) ou aumentar a aposta (*raise*). O jogador passa a vez ou aposta quando não há apostas anteriores a dele. Para a pesquisa, considera-se apenas as decisões que resultam de uma aposta anterior, citadas na Figura 2.

Figura 2 – Decisões da Pesquisa

Fonte: *Pokerstars* - Editado pelo Autor (2023)

Imagine que um jogador X decide se irá desistir, pagar ou aumentar uma aposta aleatoriamente. Para isso, ele joga um dado comum e caso o resultado for 1 ou 2 ele desiste, caso for 3 ou 4 ele paga e caso for 5 ou 6 ele aumenta a aposta. Nesta situação hipotética a probabilidade do jogador X desistir, pagar ou aumentar uma aposta é de $\frac{1}{3}$, são decisões equiprováveis que configuram uma distribuição uniformemente variada. Neste caso, não é possível identificar algum padrão e estimar qual será a próxima decisão do jogador em questão.

Na prática, participantes de um torneio não medem suas ações de forma aleatória, há estratégias que configuram um jogo mais lucrativo e, conscientemente ou não, os jogadores tomam suas decisões baseados em alguma informação, seja das cartas ou de outros fatores.

Não é possível afirmar que todos os jogadores que participam de um torneio online coletado aleatoriamente sigam padrões previsíveis, o pesquisador está suscetível a coletar informações de jogadores que simplesmente não conhecem o jogo ou estão tendo um primeiro contato com o esporte. Acredita-se que o modelo não irá chegar em resultados otimistas se a unidade amostral for algum destes casos. Quanto menos o jogador souber sobre o jogo de Poker, é bem possível que mais imprevisível ele seja, aproximando-se do exemplo do jogador X.

Dado esta afirmação, chega-se a uma reflexão: quanto aos jogadores experientes, não seria vantajoso traçar uma estratégia de jogo imprevisível e mesmo assim lucrativa? Essa estratégia é possível de ser aplicada? Um jogador experiente terá vantagens quando seus oponentes não identificarem suas características de jogo, deixar que as identifiquem auxilia na previsibilidade de uma aposta por valor ou um blefe, por exemplo. Conclui-se que, por suposição, na esmagadora maioria das vezes há um padrão implícito nas decisões dos jogadores, independentemente das cartas em jogo. Se um jogador, seja experiente ou não, delega seu jogo a partir de estratégias existentes que não dependam das cartas, se faz necessário existir padrões que não são facilmente identificados.

Otimizar imprevisibilidade e lucratividade é a meta de todos os jogadores profissionais. Em geral, jogadores experientes conseguem identificar alguns padrões nos seus oponentes e disfarçar suas características de jogo. No entanto, nem sempre a leitura perante o adversário estará correta e nem tudo será possível esconder dos adversários. Há diversos erros de julgamento e vieses humanos que induzem a decisões erradas dentro de um torneio de Poker.

Com o intuito de diminuir a probabilidade de analisar um jogador com menos conhecimento sobre o jogo e, consequentemente, diminuir a qualidade da precisão do modelo, deve-se analisar apenas os três primeiros colocados do *Sit in Go*, denominados como Jogador 1, Jogador 2 e Jogador 3, respectivamente em ordem de classificação.

Mas e quando falamos em aprendizagem estatística? Obviamente, os vieses destes modelos são muito menores do que os vieses humanos. Com a modelagem dos algoritmos de aprendizagem estatística é possível observar e analisar padrões que o cérebro humano não consegue identificar facilmente. Por este motivo, será utilizado um “detector de decisões": o algoritmo Naive Bayes. Trata-se de um modelo de classificação poderoso e de fácil aplicabilidade que utiliza o Teorema de Bayes para estimar futuras observações.

O objetivo é mostrar que é possível identificar padrões comportamentais e estratégicos nos jogadores em um torneio de Poker online. E o melhor indicativo de previsibilidade e identificação de padrões é estimar com sucesso qual decisão o jogador irá tomar sem a informação de suas cartas. O alvo de estudo será avaliado a partir de quatro circunstâncias submetidas naturalmente ao decorrer do torneio, são elas: *Pot Odds, Blinds,* Posição e Turno (*Street)*.

A pergunta a ser feita é: dado que o jogador A está em determinada posição, com uma determinada quantidade de *blinds* em seu *stack*, em determinado turno (*street*) e lhe foi oferecida determinada *pot odds.* Qual a decisão que o jogador A irá tomar? Observa-se que é um problema probabilístico adotando a interpretação frequentista de probabilidade, onde o número de resultados de eventos passados irá convergir assintoticamente ao verdadeiro valor da probabilidade desconhecida a partir que n se aproximar do infinito. No entanto, não é possível realizar repetições independentes e identicamente distribuídas neste experimento, ao decorrer do torneio o jogador se depara constantemente com cenários diferentes. E é a partir de problemas probabilísticos não triviais que surgem as soluções indutivas dos modelos estatísticos que utilizamos nos dias de hoje.

Alguns objetivos específicos devem ser alcançados: diminuir ao máximo os vieses possíveis que podem estar presentes no procedimento de amostragem e modelagem; aplicar o modelo Naive Bayes e esperar uma boa acurácia das estimativas; validar os resultados e ter indícios de generalizações na aplicabilidade deste modelo em outros torneios de Poker.

A pesquisa visa agregar informação para um jogador de Poker experiente e gerar lucratividade no esporte. Mostrar também que o Poker não é apenas um jogo de cartas e sim um vasto campo a ser explorado com o poder de contribuir para novos estudos.

2 REVISÃO BIBLIOGRÁFICA

2.1 Terminologias do Poker

Para melhor entendimento da pesquisa, é necessário apresentar algumas terminologias que serão úteis para o estudo.

1) *Sit'n Go:* um torneio de Poker online. Geralmente são torneios muito jogados pois são mais curtos e podem ser bem lucrativos se jogados em um volume alto.
2) *Stack:* o montante de fichas que o jogador possui.
3) *Street:* o termo utilizado para denominar os turnos que cada rodada de Poker possui. A mão se inicia com um turno Pré Flop e, enquanto houverem no mínimo dois jogadores disputando o pote, a mão prossegue para o Flop, onde são viradas as três primeiras cartas na mesa. Após o Flop vem o Turn, onde é virada a quarta carta e, por fim, o River, onde é virada a quinta carta.
4) *Blinds:* os blinds regem uma aposta mínima e obrigatória em um jogo de Poker. Se os blinds são 50/100, a aposta mínima de cada jogador na mesa deve ser 100 fichas em caso de disputa de pote. Apenas duas posições são obrigadas a jogar essas quantias, o Small Blind que paga 50 e o Big Blind que paga 100. Dito isso, se algum jogador possuir um stack de 1000 fichas, quer dizer que possui 10 big blinds ou, resumidamente, 10 blinds. Os blinds vão aumentando a cada nível do torneio.
5) *Pot Odds:* são uma expressão matemática que lhe dá a razão de risco-recompensa para pagar uma aposta. (Moshman, p. 15). Um valor planejado para auxiliar na tomada de decisões em um jogo de Poker. Se temos um pote de 100, e um jogador aposta 50, teremos um pote de 150, ou seja, uma pot odds de 3-para-1. Traduzindo para porcentagem, este exemplo resulta em 25%, pois devemos fazer o cálculo de $1/(3+1)$, este seria, em proporção, seu valor de investimento no pote caso aceite pagar a aposta. Se suas chances de ganhar a mão (*odds*) forem maiores ou próximas de 25%, aconselha-se permanecer na disputa do pote, se forem menores, o aconselhado é desistir pois você estará investindo um valor maior do que suas chances de ganhar.

2.2 Algoritmos

Um algoritmo é um conjunto de passos que definem a forma como uma tarefa é executada (Brookshear, p. 15). Aplicando à programação, ao abordarmos algoritmos com modelagem e aprendizagem estatística, conseguimos implementá-los com o intuito de inferir observações futuras de uma variável de interesse e parâmetros dentro de uma pesquisa. Geralmente, os algoritmos de aprendizagem estatística são úteis quando o pesquisador se depara com uma enorme quantidade de dados, no entanto também podem ser utilizados para amostras menores.

O pesquisador Leo Breiman definiu em seu artigo *Statistical modeling: The Two Cultures* duas culturas no uso de modelos estatísticos. A primeira seria a cultura de modelagem de dados, predominantemente inserida na comunidade estatística e fundamentada a partir de probabilidade e inferência, supõe que o modelo está correto dado que as suposições e pressupostos feitos no início da pesquisa estejam corretas também. A segunda cultura seria a de modelagem algorítmica, predominante na comunidade de aprendizagem de máquina e ciência de dados. Esta abordagem assume que o modelo ajustado pode ou não estar correto em suas suposições, o interesse é apenas verificar se o modelo usado para criar algoritmos de previsão estão prevendo bem ou não. Geralmente, pesquisadores testam vários algoritmos para um determinado problema e o que estiver prevendo melhor é utilizado.

2.3 Aprendizagem Estatística (Statistical Learning)

O termo Statistical Learning (SL) também pode ser chamado de Machine Learning (ML). Relatos históricos afirmam que o termo Machine Learning surgiu na década de 1950 criado pelo cientista da computação Arthur Lee Samuel, criando um algoritmo de auto aprendizagem computacional. Traduzindo para o português, Machine Learning significa Aprendizagem de Máquina, um sistema de análise de dados que automatiza modelos estatísticos. O programador “ensina” o algoritmo a prever resultados baseados nas informações que lhe é fornecida, informações essas provenientes de uma base de dados dado uma ou mais amostras.

Dentro do universo de Machine Learning temos vários algoritmos utilizando métodos estatísticos com o objetivo de estimar classes de uma variável resposta de

interesse, realizar regressões e até mesmo problemas de agrupamentos. As aprendizagens podem ser definidas por três tipos.

2.3.1 Aprendizagem Supervisionada

Os algoritmos de aprendizagem supervisionada procuram identificar características similares e diferentes entre as variáveis da base de dados para assim poder classificá-las com o intuito de estimar o comportamento de variáveis dependentes. Dadas observações $(X_{1,}Y_1),\ldots,(X_{n,}Y_m)$, o objetivo é construir um modelo para prever Y_i usando X_i. Com essa classificação o algoritmo cria uma estrutura que consiga percorrer por esses agrupamentos e definir, caso tenha um novo elemento na base de dados, onde este elemento vai se encaixar.

2.3.2 Aprendizagem Não Supervisionada

São métodos descritivos que classificam automaticamente os dados, você não tem um fator de supervisão para informar em qual classe pertence determinado elemento. Dadas observações $X_1,\ldots,X_n$, o objetivo é descobrir alguma estrutura baseada na similaridade. Esse tipo de aprendizagem necessita de uma análise para determinar o significado dos padrões encontrados na base de dados. Um exemplo comumente utilizado é a análise de *clusters*.

2.3.3 Aprendizagem Por Reforço

É um aprendizado baseado em aprender com o decorrer do tempo, pois o algoritmo precisa errar para aprender. Pode-se aprender por meio de iterações com situações que o pesquisador submete o experimento ou situações derivadas do ambiente. O algoritmo deste tipo de aprendizado precisa mapear as situações negativas e positivas e procurar aprimorar seus resultados.

2.4 Modelos de Regressão e Classificação

Há dois tipos de modelos que podem ser utilizados para prever o comportamento de uma variável de interesse, são chamados de *modelos de regressão* e *modelos de classificação*.

Os modelos de regressão são utilizados quando a variável resposta é quantitativa, podem ser ajustados por modelos de regressão linear, quadrática, logarítmica entre outras. Atenta-se que apesar do nome, a regressão logística se trata de um modelo de classificação. Os modelos de classificação levam esse nome pois o interesse é prever o comportamento de variáveis resposta qualitativas com K classes. As classes ou categorias de uma variável qualitativa são todos os possíveis resultados que essa variável pode obter, tendo como exemplo a variável resposta do estudo, temos que Y possui três classes: *Fold*, *Call* e *Raise*.

Se as variáveis preditoras são qualitativas ou quantitativas é geralmente considerado menos importante. A maioria dos métodos de aprendizagem estatístico podem ser aplicados independentemente do tipo da variável preditora, desde que quaisquer preditores qualitativos sejam devidamente tratados antes que a análise seja realizada (James, Gareth, p. 29).

O objetivo ao aplicar esses modelos de classificação é estimar observações futuras da variável de interesse com a maior assertividade possível, no ramo da aprendizagem é necessário o uso de algoritmos computacionais e métodos estatísticos para alcançar os objetivos definidos pelo pesquisador e determinar o desempenho do modelo ajustado.

2.5 Métodos de Reamostragem (Simulação)

Os métodos de reamostragem são fundamentais dentro de pesquisas que envolvem grupos de treinamento e teste. Estes métodos envolvem desenhar repetidamente amostras de um conjunto de treinamento e ajustar um modelo de interesse para cada amostra definida, obtendo informações que não seriam possíveis utilizando apenas uma amostra (James, Gareth, p. 197).

O objetivo é reamostrar o grupo de treinamento da pesquisa a fim de obter hiperparâmetros generalizados para o modelo e observar seus comportamentos quando aplicados em diversas amostras. Se tivermos apenas um valor para acurácia em um modelo de classificação, por exemplo, não é possível determinar com

confiança que se obtivemos outras amostras este valor se manteria aproximadamente equivalente. Isto engloba também que muitas vezes, quando selecionamos apenas uma amostra aleatoriamente para os grupos de treinamento e teste algumas informações que estiverem no grupo de treinamento poderiam ser mais úteis para o aprendizado do modelo se estivessem no grupo de teste ou vice e versa. O método a ser utilizado nesta pesquisa será o bootstrap.

2.6 Naive Bayes

Naive Bayes é um algoritmo estatístico utilizado em aprendizagem de máquina para tarefas de classificação e previsão. O algoritmo se baseia no teorema de Bayes, desenvolvido por Thomas Bayes no século 18 (Maarseveen, p. 2). Como mencionado, um algoritmo é uma sucessão de passos para executar uma tarefa, iremos apresentar os passos que devem ser seguidos para a realização do algoritmo.

- **Passo 1**: **Preparação dos dados**

O primeiro passo para qualquer algoritmo de aprendizagem estatística é tratar os dados e variáveis devidamente para serem incrementadas no modelo. No caso do Naive Bayes, precisamos preparar nossos dados em um formato que seja facilmente processado pelo algoritmo computacional. Isso significa transformar, agrupar ou normalizar dados para que sejam corretamente analisados pelo algoritmo.

- **Passo 2: Selecionar Variáveis**

Quando nos deparamos com uma base de dados prontamente formada, nem todas as variáveis coletadas são úteis para a utilização no modelo. Ou mesmo nem todas as classes de uma variável qualitativa são importantes para o problema de pesquisa. Deve-se então extrair as melhores variáveis e as melhores informações de cada variável para serem modeladas. Alguns métodos podem ser úteis para verificar quais variáveis ou classes estão sendo mais importantes para a ocorrência da variável resposta de interesse.

- **Passo 3: Cálculo da Probabilidade**

Tratados os dados e já definidas as variáveis explicativas e resposta do modelo, devemos começar os cálculos das probabilidades de interesse. A princípio, a primeira informação que devemos obter são as probabilidades a priori das classes da variável resposta. É o equivalente a proporção de cada classe pelo grupo de treinamento obtido. Observe o exemplo considerando o “Jogador 1” da pesquisa com um grupo de treinamento de 50.

$$\pi_1 = P(Fold) = \frac{28}{50} = 0{,}56$$
$$\pi_2 = P(Call) = \frac{18}{50} = 0{,}36$$
$$\pi_3 = P(Raise) = \frac{4}{50} = 0{,}08$$

- **Passo 4: Previsão**

Obtidas as probabilidades das classes da variável resposta, podemos usar o teorema de Bayes para calcular a probabilidade de cada classe condicionadas às variáveis explicativas.

$$P(Y = k|X = x) = \frac{\pi_k f_k(x)}{\sum_{j=1}^{K} \pi_j f_j(x)}$$

onde,

$$f_k(x) = f_{k1}(x_1).f_{k2}(x_2),\ldots,f_{kp}(x_p)$$

Observa-se que a multiplicação só é possível pelo pressuposto de que as variáveis explicativas são independentes. A partir deste cálculo conseguimos definir a expressão da probabilidade a posteriori.

$$P(Y = k|X = x) = \frac{\pi_k f_{k1}(x_1).f_{k2}(x_2),\ldots,f_{kp}(x_p)}{\sum_{j=1}^{K} \pi_j f_{j1}(x_1).f_{j2}(x_2),\ldots,f_{jp}(x_p)}$$

Para o cálculo acima ser realizado, deve-se ter sido criado uma tabela destas probabilidades para fins de classificação quando testarmos uma nova observação

derivada de um grupo de teste. Para a criação da tabela, houve uma categorização das variáveis quantitativas utilizando o mesmo método utilizado na criação de distribuições de frequência por classes. Segue como exemplo a criação de classes para a variável independente X_1.

Tabela 1 - Tabela de Distribuição de Frequência X_1

Classes	Frequência Absoluta	Frequência Relativa
0,02 -- | 0,07	1	0,0024
0,07 -- | 0,12	35	0,0833
0,12 -- | 0,17	15	0,0357
0,17 -- | 0,22	52	0,1238
0,22 -- | 0,27	72	0,1714
0,27 -- | 0,32	72	0,1714
0,32 -- | 0,37	34	0,081
0,37 -- | 0,42	130	0,3095
0,42 -- | 0,47	9	0,0214
TOTAL	420	1

Fonte: Autor (2023)

Tabela 2: Tabela de Probabilidades

Y/X1	(0.02;0.07]	(0.07;0.12]	(0.12;0.17]	(0.17;0.22]	(0.22;0.27]	(0.27;0.32]	(0.32;0.37]	(0.37;0.42]	(0.42;0.47]	TOTAL
Fold (28)	0	0,0357	0,0357	0,0714	0,2857	0,2143	0,1071	0,2143	0,0357	0,9999
Call (18)	0	0,0555	0,1111	0,2222	0,4444	0,0555	0,0555	0,0555	0	0,9997
Raise (4)	0	0	0	0,25	0,5	0	0	0,25	0	1

Y/X2	(2.6;13]	(13;23.4]	(23.4;33.8]	(33.8;44.2]	(44.2;54.6]	(54.6;65]	(65;75.4]	(75.4;85.8]	(85.8;96.2]	(96.2;106.6]	(106.6;117]	TOTAL
Fold (28)	0	0,5357	0,1071	0,0714	0,1071	0	0,1071	0,0714	0	0	0	0,9998
Call (18)	0,1667	0,1667	0,0555	0,1667	0,3333	0	0,1111	0	0	0	0	1
Raise (4)	0	0,25	0,25	0	0,25	0	0,25	0	0	0	0	1

Y/X3	UTG	MP1	MP2	MP3	HJ	CT	BT	SB	BB	TOTAL
Fold (28)	0,1428	0,0714	0,0714	0	0,1071	0,1786	0,1786	0,1786	0,0714	0,9999
Call (18)	0	0,0555	0	0	0,2778	0,0555	0,2222	0,3333	0,0555	0,9998
Raise (4)	0	0	0	0,25	0,5	0	0	0	0,25	1

Y/X4	Pré	Flop	Turn	River	TOTAL
Fold (28)	0,7143	0,1428	0,0714	0,0714	0,9999
Call (18)	0,7222	0,1111	0,1667	0	1
Raise (4)	0,5	0,25	0	0,25	1

Fonte: Autor (2023)

A partir dessa tabela serão feitas as classificações e por fim o cálculo das estimativas na expressão de probabilidade a posteriori. Como explicado no tópico de aprendizagem supervisionada, dependendo das informações da nova observação que

será testada, deve-se percorrer por esses agrupamentos e encontrar a maior probabilidade entre as três classes calculadas e, por fim, definir a estimativa para aquela observação, realizando, assim, uma previsão. Um exemplo mais prático deste cálculo será mostrado no tópico de Metodologia.

- **Passo 5: Validação**

A etapa de validação serve para descrever e verificar o desempenho ou performance do modelo. Estatísticas e métricas de desempenho são verificadas para concluir se nosso modelo está ou não sendo assertivo e tendo resultados significativos.

2.7 Métricas de Desempenho

A matriz de confusão é uma matriz de visualização dos acertos e erros dos algoritmos e modelos utilizados. Algoritmos que trabalham para prever variáveis com classes, a partir de uma contagem, utilizam a matriz de confusão e suas métricas para entender o quão eficiente foi o modelo utilizado para aquela determinada amostra. A disposição das informações em uma matriz de confusão nos permite avaliar de uma maneira simples e prática os falsos positivos (FP), falsos negativos (FN), verdadeiros positivos (VP) e verdadeiros negativos (VN) que resultaram do nosso modelo. Observe a figura abaixo:

Figura 3 – Matriz de Confusão 2x2

	PREVISTO	
REAL	VERDADEIRO	FALSO
VERDADEIRO	VP	FN
FALSO	FP	VN

Fonte: Autor (2023)

Os falsos positivos e falsos negativos são os erros que resultaram do modelo, portanto são os valores que o algoritmo estimou e não condizem com as verdadeiras observações. Já os verdadeiros positivos e verdadeiros negativos são os valores corretamente previstos pelo modelo, ou seja, que condizem com a realidade.

2.6.1 Métricas

- **Acurácia**

$$Acurácia = \frac{VP + VN}{VP + VN + FP + FN}$$

A acurácia trata de todos os erros, portanto o algoritmo divide os acertos pelo número total da amostra. Uma acurácia alta pode indicar uma boa modelagem.

- **Precisão**

$$Precisão = \frac{VP}{VP + FP}$$

A precisão é a taxa de acertos positivos do modelo em relação aos verdadeiros casos positivos.

- **Recall (True Positive Rate)**

$$Recall = \frac{VP}{VP + FN}$$

Relação entre os valores verdadeiros positivos da amostra.

- **Specificity (True Negative Rate)**

$$Specificity = \frac{VN}{VN + FP}$$

Relação entre os valores verdadeiros negativos da amostra.

- **F1 Score**

$$F1\ Score = 2 * \frac{Precisão * Recall}{Precisão + Recall}$$

Um modelo com um bom valor de F1 Score nos indica um modelo muito assertivo em suas previsões, consequentemente, houve uma boa taxa de Recall.

3 METODOLOGIA

3.1 Coleta dos Dados

Foi selecionado aleatoriamente um torneio online e gratuito no formato *Sit in Go* de nove jogadores pelo aplicativo *Pokerstars.* A cada rodada de apostas foram coletadas informações de todos os jogadores em mesa, consecutivamente. No total, considerando todos os jogadores, foram coletadas 621 observações. Se a ação estivesse no Jogador 8 por exemplo, era anotado as informações de todos os jogadores envolvidos na mão. Jogadores que desistissem de disputar o pote, ou seja, tomavam a decisão de *Fold* não eram mais observados até que aquela mão acabasse. Toda a modelagem e tratamento dos dados foram realizados no software R. Observe-se as variáveis selecionadas para a coleta de dados e algumas observações coletadas:

Tabela 3 - Base de Dados

Nível	Mão	Jogador	Posição	Street	Blinds	Pot Odds	Decisão
1	1	5	UTG	Pré	50	0,4	Fold
1	1	9	MP1	Pré	50	0,4	Call
1	1	7	MP2	Pré	50	0,29	Call
1	1	3	MP3	Pré	50	0,22	Fold
1	1	1	HJ	Pré	50	0,22	Raise
1	1	2	CT	Pré	50	0,27	Fold
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
4	45	5	CT	River	9,89	0,24	Fold
4	46	5	UTG	Pré	9,89	0,4	Raise
4	46	3	CT	Pré	9,7	0,46	Fold
4	46	1	BT	Pré	15,37	0,46	Fold
4	46	2	SB	Pré	21,37	0,45	Fold
4	46	4	BB	Pré	32,17	0,44	Fold

Fonte: Autor (2023)

3.2 Variáveis Selecionadas

- Variável Resposta (Dependente)

Y: Decisão. $\Omega = \{(Fold), (Call), (Raise)\}$

- Variáveis Explicativas (Independentes)

X_1: *Pot Odds*. $\Omega = \{x_1 : 0 \leq x_1 \leq 1\}$
X_2: *Blinds*. $\Omega = \{x_2 : 0 \leq x_2 \leq \infty\}$
X_3: Posição. $\Omega = \{(UTG), (MP1), (MP2), (MP3), (HJ), (CO), (BT)\}$
X_4: *Street.* $\Omega = \{(Pré\ Flop), (Flop), (Turn), (River)\}$

3.3 Pré-processamento

As práticas de pré-processamento realizam um tratamento e limpeza dos dados e é fundamental para criar bons modelos. Foram retirados da amostras todos os elementos NA, resultando em uma diminuição de n para 420. Foram testados resultados com e sem a transformação das variáveis qualitativas e não houve diferença no desempenho no modelo. Tratados os dados, faremos a divisão da amostra total por jogador. Como segue nas tabelas seguintes:

Tabela 4 - Tabela de Distribuição de Frequência “Jogador 1”

Jogador 1	Fold	Call	Raise	TOTAL
Frequência Absoluta	38	24	6	68
Frequência Relativa	0,559	0,353	0,088	1

Fonte: Autor (2023)

Tabela 5 - Tabela de Distribuição de Frequência “Jogador 2”

Jogador 2	Fold	Call	Raise	TOTAL
Frequência Absoluta	36	10	6	52
Frequência Relativa	0,692	0,192	0,116	1

Fonte: Autor (2023)

Tabela 6 - Tabela de Distribuição de Frequência “Jogador 3”

Jogador 3	Fold	Call	Raise	TOTAL
Frequência Absoluta	38	30	2	70
Frequência Relativa	0,543	0,428	0,029	1

Fonte: Autor (2023)

3.4 Algoritmo Naive Bayes

Tomaremos como exemplo de explicação do procedimento do algoritmo uma única observação do grupo de teste para o “Jogador 1”. A partir do cálculo das probabilidades, conseguimos construir a seguinte tabela:

Tabela 7 - Tabela de Probabilidades “Jogador 1”

Y/X1	(0.02;0.07]	(0.07;0.12]	(0.12;0.17]	(0.17;0.22]	(0.22;0.27]	(0.27;0.32]	(0.32;0.37]	(0.37;0.42]	(0.42;0.47]	TOTAL
Fold (28)	0	0,0357	0,0357	0,0714	0,2857	0,2143	0,1071	0,2143	0,0357	0,9999
Call (18)	0	0,0555	0,1111	0,2222	0,4444	0,0555	0,0555	0,0555	0	0,9997
Raise (4)	0	0	0	0,25	0,5	0	0	0,25	0	1

Y/X2	(2.6;13]	(13;23.4]	(23.4;33.8]	(33.8;44.2]	(44.2;54.6]	(54.6;65]	(65;75.4]	(75.4;85.8]	(85.8;96.2]	(96.2;106.6]	(106.6;117]	TOTAL
Fold (28)	0	0,5357	0,1071	0,0714	0,1071	0	0,1071	0,0714	0	0	0	0,9998
Call (18)	0,1667	0,1667	0,0555	0,1667	0,3333	0	0,1111	0	0	0	0	1
Raise (4)	0	0,25	0,25	0	0,25	0	0,25	0	0	0	0	1

Y/X3	UTG	MP1	MP2	MP3	HJ	CT	BT	SB	BB	TOTAL
Fold (28)	0,1428	0,0714	0,0714	0	0,1071	0,1786	0,1786	0,1786	0,0714	0,9999
Call (18)	0	0,0555	0	0	0,2778	0,0555	0,2222	0,3333	0,0555	0,9998
Raise (4)	0	0	0	0,25	0,5	0	0	0	0,25	1

Y/X4	Pré	Flop	Turn	River	TOTAL
Fold (28)	0,7143	0,1428	0,0714	0,0714	0,9999
Call (18)	0,7222	0,1111	0,1667	0	1
Raise (4)	0,5	0,25	0	0,25	1

Fonte: Autor (2023)

Selecionamos a primeira observação do grupo de validação para mostrar como é realizado o cálculo da estimativa, temos que:

Tabela 8: Primeira Observação Grupo Teste “Jogador 1”

Posição	Street	Blinds	Pot Odds
HJ	Turn	45	0.33

Fonte: Autor (2023)

A partir das informações das variáveis explicativas, o algoritmo começa a etapa de classificação e posteriormente o cálculo das probabilidades a posteriori.

Tabela 9 - Etapa de Classificação “Jogador 1”

Y/X1	(0.02;0.07]	(0.07;0.12]	(0.12;0.17]	(0.17;0.22]	(0.22;0.27]	(0.27;0.32]	(0.32;0.37]	(0.37;0.42]	(0.42;0.47]	TOTAL
Fold (28)	0	0,0357	0,0357	0,0714	0,2857	0,2143	0,1071	0,2143	0,0357	0,9999
Call (18)	0	0,0555	0,1111	0,2222	0,4444	0,0555	0,0555	0,0555	0	0,9997
Raise (4)	0	0	0	0,25	0,5	0	0	0,25	0	1

Y/X2	(2.6;13]	(13;23.4]	(23.4;33.8]	(33.8;44.2]	(44.2;54.6]	(54.6;65]	(65;75.4]	(75.4;85.8]	(85.8;96.2]	(96.2;106.6]	(106.6;117]	TOTAL
Fold (28)	0	0,5357	0,1071	0,0714	0,1071	0	0,1071	0,0714	0	0	0	0,9998
Call (18)	0,1667	0,1667	0,0555	0,1667	0,3333	0	0,1111	0	0	0	0	1
Raise (4)	0	0,25	0,25	0	0,25	0	0,25	0	0	0	0	1

Y/X3	UTG	MP1	MP2	MP3	HJ	CT	BT	SB	BB	TOTAL
Fold (28)	0,1428	0,0714	0,0714	0	0,1071	0,1786	0,1786	0,1786	0,0714	0,9999
Call (18)	0	0,0555	0	0	0,2778	0,0555	0,2222	0,3333	0,0555	0,9998
Raise (4)	0	0	0	0,25	0,5	0	0	0	0,25	1

Y/X4	Pré	Flop	Turn	River	TOTAL
Fold (28)	0,7143	0,1428	0,0714	0,0714	0,9999
Call (18)	0,7222	0,1111	0,1667	0	1
Raise (4)	0,5	0,25	0	0,25	1

Fonte: Autor (2023)

Terminada a classificação, o algoritmo realiza o cálculo das probabilidades a posteriori utilizando as probabilidades condicionais referentes às variáveis explicativas por classes.

$$P_F(x) = p_{F1}(0{,}32 \leq x_1 < 0{,}27).p_{F2}(44{,}2 \leq x_2 < 54{,}6).p_{F3}(x_3 = HJ).p_{F4}(x_4 = Turn)$$
$$P_F(x) = 0{,}1071.0{,}1071.0{,}1071.0{,}0714 = 0{,}000087$$

$$P_C(x) = p_{C1}(0{,}32 \leq x_1 < 0{,}27).p_{C2}(44{,}2 \leq x_2 < 54{,}6).p_{C3}(x_3 = HJ).p_{C4}(x_4 = Turn)$$
$$P_C(x) = 0{,}0555.0{,}3333.0{,}2778.0{,}1667 = 0{,}000857$$

$$P_R(x) = p_{R1}(0{,}32 \leq x_1 < 0{,}27).p_{R2}(44{,}2 \leq x_2 < 54{,}6).p_{R3}(x_3 = HJ).p_{R4}(x_4 = Turn)$$
$$P_R(x) = 0.0{,}25.0{,}5.0 = 0$$

Considerando as probabilidades a priori do “Jogador 1 “que definimos no tópico de Naive Bayes, temos que:

$$P(Y = Fold|X = x) = \frac{0{,}56 * 0{,}000087}{0{,}56 * 0{,}000087 + 0{,}36 * 0{,}000857 + 0{,}08 * 0} = \frac{0{,}000049}{0{,}000357}$$

$$= 0{,}137$$

$$P(Y = Call|X = x) = \frac{0{,}36 * 0{,}000857}{0{,}000357} = \frac{0{,}000308}{0{,}000357} = 0{,}863$$

$$P(Y = Raise|X = x) = \frac{0}{0{,}000357} = 0$$

Portanto, toma-se como estimativa para a primeira observação do grupo de validação a decisão *Call*, pois foi o evento que obteve a maior probabilidade. Neste caso o algoritmo acertou, de fato o “Jogador 1” optou por pagar a aposta. Este procedimento é feito para as 18 observações de teste para depois podermos verificar o desempenho do modelo.

4 RESULTADOS

Para o Jogador 1, conseguimos obter as seguintes medidas de desempenho para uma amostra de $n = 68$, com 50 observações de treinamento e 18 para teste, equivalente à um grupo de treinamento de 75%.

Figura 4 - Matriz de Confusão “Jogador 1”

	Referencia		
Predição	Fold	Call	Raise
Fold	8	4	0
Call	1	3	0
Raise	0	0	2

Fonte: Autor (2023)

Tabela 10 - Métricas de Desempenho “Jogador 1”

Accuracy	0,7222
95% CI	(0,4652; 0,9031)
No Information Rate	0,5
P-Value	0,04813
Kappa	0,5109

	Fold	Call	Raise
Sensitivity	0,8889	0,4286	1
Specificity	0,5556	0,9091	1
Pos Pred Value	0,6667	0,75	1
Neg Pred Value	0,8333	0,7143	1
Prevalence	0,5	0,3889	0,1111
Detection Rate	0,4444	0,1667	0,1111
Detection Prevalence	0,6667	0,2222	0,1111
Balanced Accuracy	0,7222	0,6688	1

Fonte: Autor (2023)

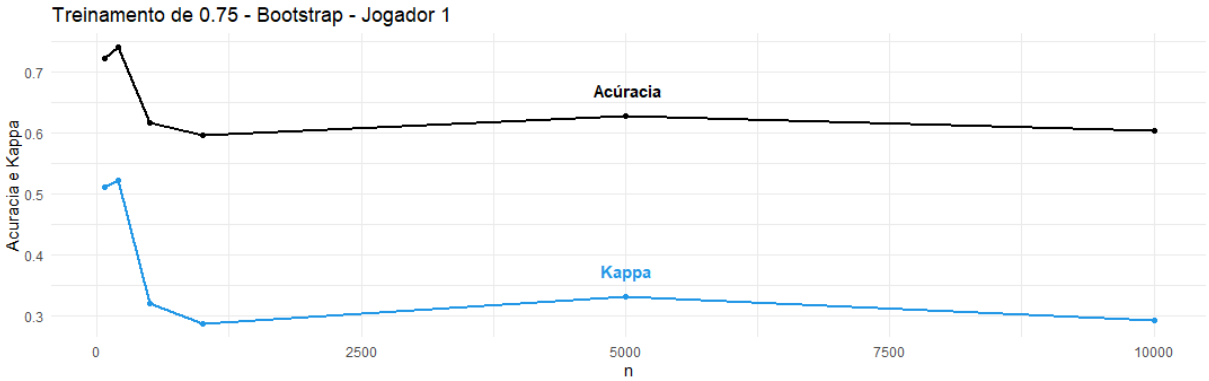
Para obter uma melhor validação dos resultados, fizemos um bootstrap da amostra do “Jogador 1”. Todas as amostras de tamanho n foram submetidas a um valor de 75% para treinamento. Neste método de reamostragem nosso objetivo foi verificar a variação do valor de acurácia e kappa resultantes de amostras maiores. O valor de kappa, não mencionado no texto, varia de -1 a 1 e seria o equivalente a dizer se meus resultados se aproximam de acertos meramente aleatórios (positivos próximos de 0) ou, de fato, foram resultados significativos. Um valor próximo de 1 significa que o modelo está performando bem e acertando observações a partir de suas classificações e um valor próximo de -1 significa que o modelo está performando pior do que se os valores fossem estimados aleatoriamente. Este método será utilizado para os outros dois jogadores também.

Tabela 11 - Bootstrap “Jogador 1”

n	Acurácia	Kappa
68	0,7222	0,5109
200	0,74	0,5228
500	0,616	0,32
1000	0,596	0,2872
5000	0,628	0,331
10000	0,6038	0,2924

Fonte: Autor (2023)

Figura 5 - Gráfico Bootstrap “Jogador 1”



Fonte: Autor (2023)

Para o Jogador 2, conseguimos obter as seguintes medidas de desempenho para uma amostra de $n = 52$, com 39 observações de treinamento e 13 para teste, equivalente à um grupo de treinamento de 75%.

Figura 6 - Matriz de Confusão “Jogador 2”

	Referência		
Predição	Fold	Call	Raise
Fold	8	0	1
Call	0	2	0
Raise	2	0	0

Fonte: Autor (2023)

Tabela 12 - Métricas de Desempenho “Jogador 2”

Accuracy	0.7692
95% CI	(0.4619, 0.9496)
No Information Rate	0.7692
P-Value	0.6485
Kappa	0.4658

	Fold	Call	Raise
Sensitivity	0.8000	1	0
Specificity	0.6667	1	0.8333
Pos Pred Value	0.8889	1	0
Neg Pred Value	0.5000	1	0.9091
Prevalence	0.7692	0.1538	0.0769
Detection Rate	0.6154	0.1538	0
Detection Prevalence	0.6923	0.1538	0.1538
Balanced Accuracy	0.7333	1	0.4167

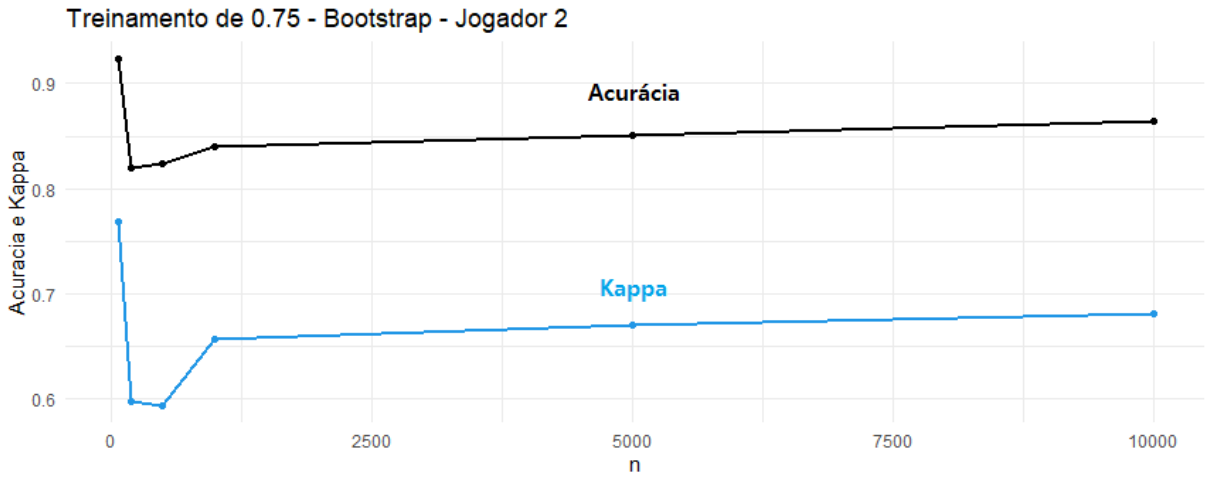
Fonte: Autor (2023)

Tabela 13 - Bootstrap “Jogador 2”

n	Acurácia	Kappa
52	0.9231	0.7679
200	0.8200	0.5968
500	0.8240	0.5934
1000	0.8400	0.6566
5000	0.8511	0.6698
10000	0.8648	0.6814

Fonte: Autor (2023)

Figura 7 - Gráfico Bootstrap “Jogador 2”



Fonte: Autor (2023)

Para o Jogador 3, conseguimos obter as seguintes medidas de desempenho para uma amostra de $n = 70$, com 52 observações de treinamento e 18 para teste, equivalente à um grupo de treinamento de 75%.

Figura 8 - Matriz de Confusão “Jogador 3”

	Referência		
Predição	Fold	Call	Raise
Fold	6	4	0
Call	4	4	0
Raise	0	0	0

Fonte: Autor (2023)

Tabela 14 - Métricas de Desempenho “Jogador 3”

Accuracy	0.5556
95% CI	(0.3076, 0.7847)
No Information Rate	0.5556
P-Value	0.5966
Kappa	0.1

	Fold	Call	Raise
Sensitivity	0.6000	0.5000	NA
Specificity	0.5000	0.6000	1
Pos Pred Value	0.6000	0.5000	NA
Neg Pred Value	0.5000	0.6000	NA
Prevalence	0.5556	0.4444	0
Detection Rate	0.3333	0.2222	0
Detection Prevalence	0.5556	0.4444	0
Balanced Accuracy	0.5500	0.5500	NA

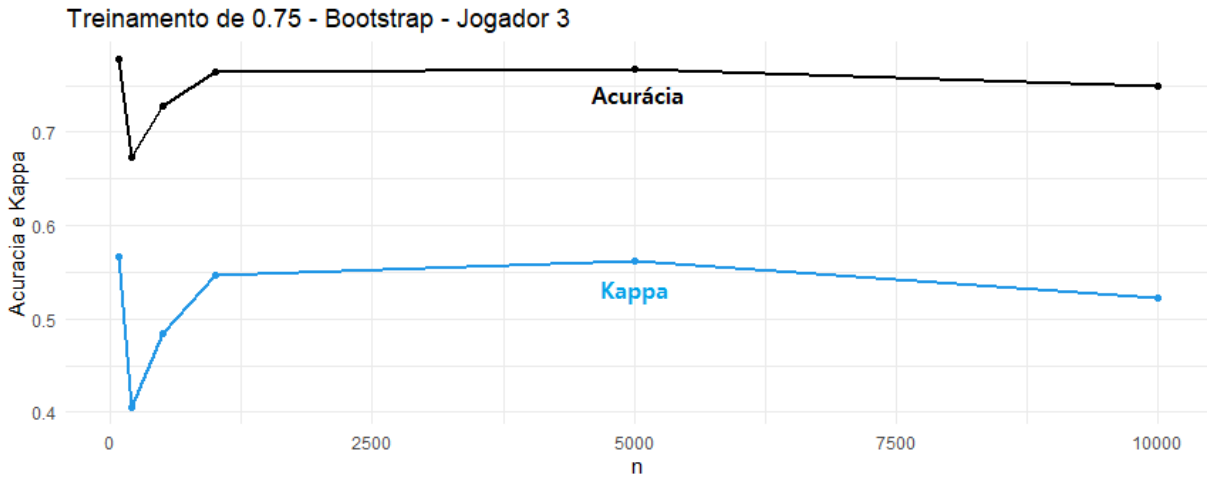
Fonte: Autor (2023)

Tabela 15 - Bootstrap “Jogador 3”

n	Acurácia	Kappa
70	0.7778	0.5663
200	0.6735	0.4056
500	0.7280	0.4837
1000	0.7640	0.5470
5000	0.7672	0.5620
10000	0.7492	0.5222

Fonte: Autor (2023)

Figura 9 - Gráfico Bootstrap “Jogador 3”



Fonte: Autor (2023)

5 CONCLUSÃO

Conclui-se que com estes resultados avaliados para os três jogadores alvos se faz necessário, muito possivelmente, acrescentar mais variáveis independentes ao modelo. Por mais que o método de reamostragem bootstrap mostre resultados otimistas, quando analisamos o modelo de Naive Bayes com o tamanho amostral original de cada jogador, se mostra bem inconstante, por vezes sendo um modelo assertivo e por vezes sendo um modelo não assertivo. Essa inconstância mostra a instabilidade do modelo, fazendo com que não possamos confiar cegamente em sua aplicação caso queiramos implementar em um torneio de Poker online real.

Observa-se também que o modelo irá performar melhor ou pior dependendo do jogador analisado, alguns jogadores podem ser mais previsíveis que outros. No entanto, no geral a pesquisa obteve resultados satisfatórios.

6 REFERÊNCIAS

HILGER, Matthew; **Texas Hold'em Odds e Probabilidades; Estratégias de Limit, No-Limit e Torneios:** tradução Karen Dias**.** Belo Horizonte: Raise Editora, 2012.

MOSHMAN, Collin; **Heads-Up: No-Limit Hold'em Poker**: tradução Karen Dias Fernandes. Belo Horizonte: Raise Editora. 2010.

MATOS, Paulo; **Relatório Técnico "Métricas de Avaliação";** USP, UFSCAR, UNIMEP; Setembro, 2009.

ROSS, Sheldon; **Probabilidade: um curso moderno com aplicações**; tradutor: Alberto Resende De Conti. - 8ª Edição - Porto Alegre: Bookman, 2010.

JAMES, R. Barry; **Probabilidade: um curso intermediário.** 1ª Edição. Rio de Janeiro: Projeto Euclides. 1981.

JAMES, Gareth; **An Introduction to Statistical Learning: with applications in R.** 2ª Edição. 2009.

HASTIE, Trevor; **The Elements of Statistical Learning: Data Mining, Inference, and Prediction**. 2ª Edição. Stanford, California: Springer. 2008.

BROOKSHEAR, J. Glenn; **Ciência da Computação: Uma Visão Abrangente**. 13ª Edição. Bookman. 2013.

BREIMAN, Leo; **Statistical Modeling: The Two Cultures**. Vol. 16, No. 3, 199-231. Statistical Science. 2001.

MAARSEVEEN, Henri; **Naive Bayes: The Foundation of Machine Learning.** Kindle. 2023.