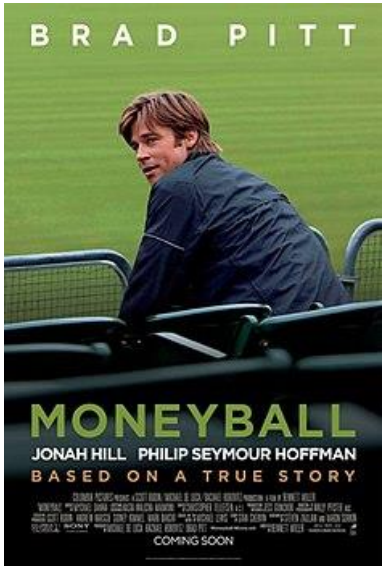
TRABALHO FINAL
MACHINE LEARNING I

PROBLEMA REAL QUE PODE SER RESOLVIDO COM UM ALGORITMO SUPERVISIONADO DE MACHINE LEARNING

Machine Learning I

DESCRIÇÃO DO PROBLEMA: MODELO DE CLASSIFICAÇÃO DE PLAYOFFS PARA TIMES DE BASEBALL

No filme '*Moneyball',* um treinador de baseball trabalha com um estatístico para formar um time vencedor, mesmo tendo verba reduzida e jogadores considerados de segunda categoria.



Esse filme é baseado em uma história real, e a atividade de aplicação de métodos estatísticos e ciência de dados na área de esportes é chamada de *Sports Statistics.*

Com crescente interesse em Sports Statistics, decidimos criar um modelo que identifica quais times de baseball se classificam ou não para os *playoffs.*

Os playoffs são uma segunda fase composta por 7 jogos eliminatórios. Nesse jogos, as 10 melhores equipes competem para se tornarem a campeã daquele ano.

Machine Learning I

FIAP

DESCRIÇÃO DO PROBLEMA: MODELO DE CLASSIFICAÇÃO DE PLAYOFFS PARA TIMES DE BASEBALL

Com um modelo que indique se um time da Major League Baseball (MLB) irá ou não se classificar para os playoffs, já é possível entender o que é importante para se classificar para os playoffs, quais times estão mais ou menos próximos de conseguirem, e ter condições iniciais para avaliar o desempenho do próprio time.

Se a resposta do modelo para o time em questão é que ele não vai se classificar, é viável saber qual das características determinantes para essa classificação precisa de maior atenção e acompanhamento para que se obtenha uma melhora de performance.

Esse tipo de informação fornece um direcionamento para o time e também auxilia na tomada de decisão de investidores externos.

SOLUÇÃO: ÁRVORE DE DECISÃO E KNN

Por se tratar de um problema em que é necessário prever se um time se classificará ou não para os Playoffs, os algoritmos de classificação vistos em aula como Árvore de Decisão e K-Vizinhos Mais Próximos (KNN) são boas opções para solucionar o problema.

A vantagem da árvore de decisão é que ela é explicável, o que facilita o entendimento do modelo e possibilita a geração de insights sobre os dados. Além disso, ela é mais robusta em relação aos dados, não sendo alterada por pontos extremos, fora da curva ou com diferença de escala.
A desvantagem da árvore é que ela é bastante passível de sobreajuste. O sobreajuste pode ou não ser uma questão para os dados.

O outro modelo que pode ser usado é o KNN. O KNN, é intuitivo, simples de usar, flexivel, tem apenas um hiperparâmetro de número de vizinhos (k) e algumas métricas de distância que podem ser escolhidas de acordo com o formato dos dados.
Por outro lado, e diferentemente da árvore, o KNN é sensível ao formato dos dados, pontos extremos, pontos fora da curva, dados faltantes e escala. Ele não lida bem com altas dimensionalidades, é mais custoso e é necessário escolher o número de vizinhos k baseado em testes, o que pode ser um pouco arbitrário. O KNN também é sensível ao desbalanceamento de classes.

APLICAÇÃO DOS ALGORITMOS DE MACHINE LEARNING VISTOS EM AULA EM UM PROBLEMA REAL

Machine Learning I

Para demostrar a aplicação desses modelos (Árvore de Decisão e KNN) em dados reais, escolhemos um dataset que está disponível no Kaggle, mas foi retirado do site da Major League Baseball (MLB). Ele possui dados históricos dos times de baseball e suas respectivas métricas de partidas.

São elas:

- Runs Scored (RS): corridas pontuadas por completar as bases
- Runs Allowed (RA): corridas pontuadas contra o arremessador
- Wins (W): total de vitórias do arremessador
- On-Base Percentage(OBP): mede o quão frequentemente um rebatedor atinge a base (rebatimentos + interferências + número de vezes atingido pelo arremessador / número de rodadas completadas)
- Slugging Percentage (SLG): performance do rebatedor (total de bases atingidas / total rebatimentos)
- Batting Average (BA): rebatimento médio
- Games Played (G): total de partidas jogadas
- Opponent On-Base Percentage (OOBP): mede o quão frequentemente o rebatedor oponente atinge a base
- Opponent Slugging Percentage (OSLG): performance do rebatedor oponente

Além disso, o dataset também possui a feature binária que diz se o time foi ou não para os Playoffs.

Machine Learning I

A lista completa de features e suas explicações pode ser vista acessando este link.

Para resolver o problema, fizemos uma análise exploratória dos dados para entender as features e escolher aquelas que iam para o modelo, testamos diversos modelos de árvore e fizemos uma busca em grade de hiperparâmetros com validação cruzada. Também entendemos qual seria o melhor k para o KNN baseado na métrica f1-score.

Escolhemos o f1-score pois queriamos obter o máximo de precisão e revocação. Percebemos que a nossa variável target era desbalanceada, realizamos testes mudando a métrica para precisão sem diferenças consideráveis nos resultados, então acabamos por manter o f1-score.

Os notebooks podem ser acessados neste link.
Há também o link para o github do projeto: https://github.com/paciencia/moneyball_sandbox

No próximo slide, apresentaremos os resultados da análise.

RESULTADOS - ÁRVORE DE DECISÃO

O melhor modelo foi o modelo de Árvore de Decisão com profundidade máxima de 2 níveis e mínimo de 50 folhas. O modelo apresentou a mesma acurácia de treino e teste, com 88%. Abaixo apresentamos o report de classificação:

	precision	recall	f1-score	support
0	0.94	0.90	0.92	290
1	0.68	0.79	0.73	80
accuracy			0.88	370
macro avg	0.81	0.84	0.83	370
weighted avg	0.88	0.88	0.88	370

Para a classe 1, a melhor precisão foi de 68, recall de 79, e f1 de 73.

Repare que tivemos apenas 80 amostras da classe 1 para testar. Provavelmente, aumentando o dataset, ou mudando para outro tipo de algoritmo com mais árvores, teremos resultados melhores.

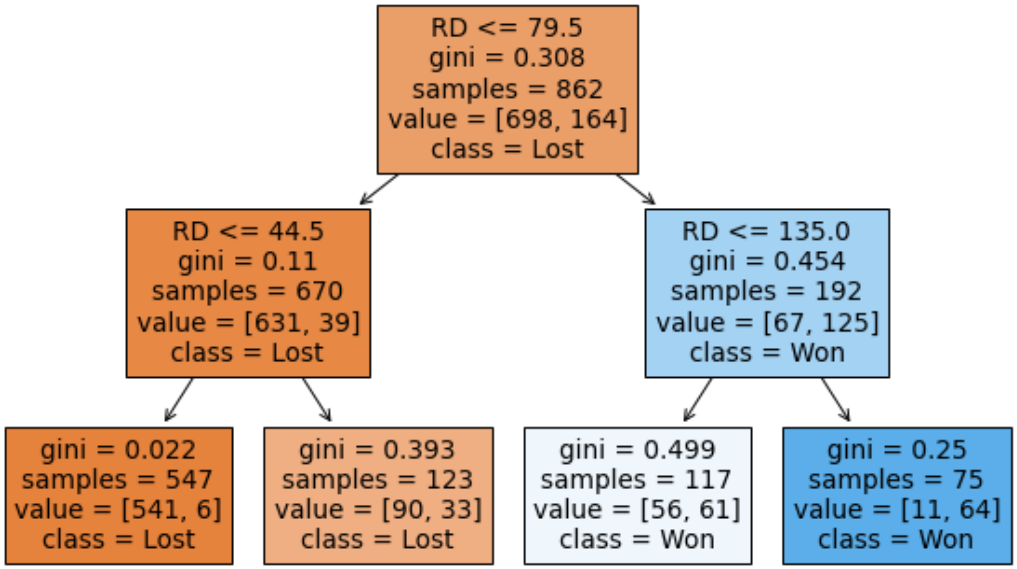
RESULTADOS

Esta foi a matriz de confusão do modelo:

Ela classificou corretamente 63 dos 80 times que foram para os playoffs. Errando para 17 entradas.

RESULTADOS

Quanto a árvore de decisão do modelo,



É possivel perceber que a única feature levada em consideração foi a RD. RD significa Result Difference, ou diferença de resultados, e foi criada a partir da diferença entre pontuações que acontecem quando a pessoa que joga a bola para o rebatedor de baseball comete falta e pontuações que acontecem quando o rebatedor acerta a bola. O que mostra, que, de acordo com esse modelo simples de árvore, obter uma pontuação <= 135 ao acertar a bola, descontando possíveis erros do arremessador parece ser decisivo para ganhar os Playoffs.

RESULTADOS - KNN

Para o KNN, o melhor K obtido (testando de 0 a 20) e sem sobreajuste, foi K=19:

	K	ACC_TRAIN	ACC_TEST	F1
18	19	0.883	0.889	0.713
10	11	0.892	0.884	0.707
16	17	0.887	0.884	0.691
17	18	0.885	0.878	0.672
9	10	0.884	0.878	0.676
14	15	0.882	0.878	0.676
8	9	0.889	0.876	0.676

K=19 também apresentou uma acurácia de treino igual a de teste, variando apenas na terceira casa decimal, como podemos ver na primeira linha destacada da tabela acima. O f1 ficou em 0.71, sendo 0.02 menor do que o f1 do modelo de árvore.

RESULTADOS

O report de classificação para o KNN, obteve uma precisão de 0.81 para a classe 1, 0.13 maior do que a precisão do modelo de árvore. No entanto, também apresentou uma revocação 0.15 menor, com 0.64. E um f1 menor de 0.71 por 0.02, uma vez que o f1 da ávore foi de 0.73.

	precision	recall	f1-score	support
0	0.91	0.96	0.93	290
1	0.81	0.64	0.71	80
accuracy			0.89	370
macro avg	0.86	0.80	0.82	370
weighted avg	0.88	0.89	0.88	370

Machine Learning I

RESULTADOS

Quanto a matriz de confusão,

O KNN acabou acertando 51 das 80 amostras de times que foram para os Playoffs. Errando para 29 entradas.

Como a Árvore de decisão errou para 17 entradas e o KNN para 29, por fim, escolhemos o algoritmo de árvore de decisão.

PROBLEMA REAL QUE **NÃO** PRECISA DE MACHINE LEARNING PARA SER RESOLVIDO

DESCRIÇÃO DO PROBLEMA: IDENTIFICAÇÃO DE POSSÍVEIS FRAUDES

Um problema presente em diversos segmentos da economia é a fraude, tanto para empresas (desvio de dinheiro, manipulações financeiras) como para governos (evasão de tributos/impostos). Quando uma fraude acontece, ela pode levar uma empresa ou governo à perda de credibilidade, valor de mercado e até à falência.

Há algumas abordagens que utilizam Machine Learning para resolver o problema de identificar fraudes, principalmente com algoritmos de clusterização. No entanto, há uma solução que não está relacionada à Machine Learning, mas apenas à análise de frequência e ao formato da distribuição dos primeiro números de movimentações financeiras, seu nome é: lei de Newcomb-Benford.

Saber que a lei de Newcomb-Benford também pode ser usada para detecção de fraudes é importante, pois é um método baseado em contagem, mais rápido e barato, que não precisa do uso de ML e que já aponta as possíveis fraudes.

FIAP

SOLUÇÃO: LEI DE NEWCOMB-BENFORD

A Lei de Newcomb-Benford, também é conhecida como a lei do primeiro digito. Ela se baseia em uma observação sobre a distribuição dos dígitos que podem aparecer, por exemplo, em patrimônios ou movimentações de contas.

A aplicação da lei consiste em comparar a distribuição de frequência dos primeiros dígitos dos dados analisados com a sua distribuição de acordo com a lei de Benford. Ao fazer isso, já aparecem os resultados de frequência anômalos que precisam ser investigados.

A fórmula da lei é:

$$P(d) = \log_{10}(d+1) - \log_{10}(d) = \log_{10}\left(\frac{d+1}{d}\right) = \log_{10}\left(1+\frac{1}{d}\right)$$

No qual d é um primeiro dígito que varia de 1 a 9.

No caso de um patrimônio, se a frequência do dígito 1 é abaixo de 30%, que geralmente é a frequência do número 1 na distribuição de Benford, isso já seria um indicativo de que é necessário investigar a diferença e que pode ter ocorrido fraude ou erro.

SOLUÇÃO: LEI DE NEWCOMB-BENFORD

Assim como todo o método, a lei de Benford possui suas limitações. O seu bom uso é dependente de certas propriedades que as distribuições dos dados e os números analisados precisam ter. Listamos abaixo tipos de distribuições que já se sabe que obedecem ou não à lei de Benford.

Distribuições que obedecem à lei de Benford:

- possuem média > mediana e enviesamento positivo;
- são compostas de números que resultam da combinação de outros números, como quantidade * preço;
- são dados variados de transações, vendas, reembolsos, montantes.

Distribuições que não obedecem à lei de Benford:

- possuem números sequenciais, como identificadores de cheques e notas fiscais;
- tem números que são decididos com base no pensamento humano, como o valor de um preço de 1,99 ou 250,00 reais;
- são contas que cumprem propósitos específicos de empresas, por exemplo, uma conta que seja específica para reembolsos de 100 reais;
- são compostas por números sem diferentes ordens de magnitude.

Machine Learning I

FIAP

FONTES

http://lycofs01.lycoming.edu/~sprgene/M400/BenfordsLaw.pdf

https://www.revistaespacios.com/a14v35n07/14350720.html

https://www.forbes.com/sites/taxnotes/2021/08/19/can-benfords-law-detect-tax-fraud/?sh=690bca124d70

https://en.wikipedia.org/wiki/Benford%27s_law