Inteligência Artificial – Base de Dados Census

Lucas Miranda Eltz, Lucas da Mata Nascimento, Luís Fábio Barros de Arruda, Rafael Hiroshi Kanasiro Pereira e Rayanne Costa Andrade

Orientador do Projeto: Gabriela Oliveira Biondi

Universidade São Judas Tadeu (USJT)

Rua Taquari, 546 – Mooca – São Paulo – SP – Brasil

Resumo. *Esse artigo busca apresentar a análise realizada na base de dados census aplicando algumas técnicas e métodos de inteligência Artificial a fim de chegarmos em um melhor resultado e custo benefício, para isso realizamos a aplicação de alguns algoritmos como árvore de decisão, regressão logística, KNN entre outras, também a comparação entre os mesmo e em todas as aplicações testamos diversas vezes para usar as melhores variáveis em todas as técnicas apresentadas.*

Palavras-chave: *Acurácia, target, base de dados, algoritmos.*

1. Sobre a Base de Dados

A base de dados census é conhecida por prever se a renda excedeu U$50 mil/ano com base no conjunto de dados de pessoas dos Estados Unidos, aparentemente possui registros limpos. Temos mais de 32 mil linhas com 15 colunas para serem analisadas e informações como: idade, classe de trabalho, raça, relação, educação, ocupação entre outras.

1.2. Variável Target

Escolhemos como variável target a idade, para prever pessoas idosas com base nas análises da base de dados onde separamos a idade mínima 60 anos e máxima 90 anos, pois são pessoas que requerem um maior cuidado e atenção, prevendo a idade com as colunas educação, relação e estado civil.

2. Comitê de Classificadores

Temos como objetivo focar em pessoas com idades mais avançadas que automaticamente estão dentro do grupo de risco do vírus COVID-19, consequentemente precisando de um maior cuidado, especialmente no momento em que estamos. Assim

facilitando a identificação desse público alvo e conectar com pessoas que querem ajudar, de maneira 100% virtual, oferecendo um serviço delivery de produtos/serviços essenciais.

3. Procedimentos de Pré-Processamento

- Verificamos campos vazios na base para exclusão ou tratamento, porém não encontramos registros faltantes.
- Definimos um range de idades para analisar sendo, idade mínima 60 anos e máxima de 90 anos.
- Transformamos a variável idade para binária, assim criando uma nova variável.
- Apagamos algumas colunas que não iríamos utilizar como: capital de ganho e perda, país nativo, ocupação, gênero e raça.
- Escolhemos as colunas que iríamos utilizar para os algoritmos como: educação, relação, estado civil, além da renda.
- Separamos a base de dados entre treino e teste, assim os primeiros 25000 registros são para treino e o restante do conjunto para teste.

4. Algoritmos Utilizados

4.1. Árvore de Decisão

Uma árvore de decisão é como um mapa dos possíveis resultados de uma série de escolhas relacionadas. Estas árvores são treinadas de acordo com um conjunto de treino (exemplos previamente classificados) e posteriormente, outros exemplos são classificados de acordo com essa mesma árvore.

Nos resultados percebemos que o treino da árvore é claramente influenciado pela variável max_depth, usamos 9 como profundidade para os testes, a fim de obter melhores resultados. Apesar dos treinos cada vez melhores, a acurácia fica entre 0.93% e 0.94% e em determinados momentos, os testes começam subindo a acurácia e logo voltam a ficar com 0.93%. Acreditamos que a profundidade "ideal" está entre 2 e 10 com uma taxa de 0.94 de acurácia.

4.2. Regressão Logística

A regressão logística tem como objetivo prever e estimar a probabilidade associada à ocorrência de determinado evento sendo que os resultados da análise ficam contidos no intervalo de zero a um.

Os cenários de testes se dividem em dois: prever a faixa etária de 0 a 20 anos e outra faixa etária de 60 a 80 anos. Os resultados obtidos mostram que a aplicação da regressão logística para a faixa etária dos 0 a 20 anos possuem uma precisão alta, enquanto para a faixa etária dos 60 a 80 anos, a precisão é baixa. Há também uma diferença de precisão ao utilizar os dados de teste e de treino. Os dados de teste possuem uma precisão maior significativa comparado aos dados de treino. Aplicamos todos os procedimentos de pré- processamento como metodologia utilizada para comparação e resultados.

4.3. Técnica KNN

A aplicação do KNN tem como objetivo classificar, onde é baseado no quão similar é um dado do vetor do outro.

Para iniciar os testes aplicamos todos os procedimentos de pré processamento e separamos a base de dados entre treino e teste, procuramos pelo melhor classificador dentro de um range. Usamos como resultado 1 para maiores de 60 anos que é nosso objetivo de classificação e 0 para o restante do conjunto de idades.

Assim, treinamos ajustando métricas e número de vizinhos para encontrar diferentes resultados, exibindo os preditos. Os resultados obtidos não variam muito sendo 0 e 1, mostrando que não importa o número de vizinhos ou métrica utilizada o predito sempre exibirá um conjunto de idades parecido.

4.4. Naive Bayes

Os classificadores Naive Bayes são probabilísticos, baseados no conhecido teorema de Bayes com um pressuposto forte (e ingênuo = naive) de independência entre as variáveis (ou características) onde há uma família de algoritmos para treinar esses classificadores, todos assumindo a tal independência.

No início do algoritmo aplicamos todos os procedimentos de pré processamento, bem como separamos a base de dados entre treino e teste. Usamos como resultado 1 para maiores de 60 anos que é nosso objetivo de classificação e 0 para o restante do conjunto de idades.

Primeiro treinamos nosso modelo com o Gaussiano, Multinomial e Bernoulli e exibimos os arrays com os resultados 0 e 1. No resultado do treino tivemos uma acurácia

para o Gaussiano 79,64% sendo o menor resultado de acurácia pelo modelo classificado pela distribuição normal.

Os outros dois modelos Multinomial e Bernoulli estão com acurácia maior que 0.93% sendo um número considerável para acurácia, onde eles treinam por contagem discreta e o outro é útil para vetores com números binários, por conta disso o número da acurácia está alto, assim são os dois modelos indicados para o nosso treinamento.

Comparando o treinamento entre os modelos KNN e Naive Bayes, a acurácia pelo modelo KNN se saiu melhor sendo 96,55% contra 81,42% do modelo Naive Bayes e olhando as médias dos dois modelos, o resultado foi parecido com 80.00% nos dois.

4.5. Redes Neurais

A aplicação das redes neurais artificiais é um modelo de aprendizagem de máquina inspirado na rede neural biológica. Após aplicarmos todos os procedimentos de pré processamento, usamos como resultado 1 para maiores de 60 anos que é nosso objetivo de classificação e 0 para o restante do conjunto de idades.

Carregamos nosso conjunto de dados com o Keras separando a base de dados entre treino e teste, redimensionamos os dados e fizemos um cast para float32. Normalizamos os dados entre 0 e 1, convertemos para matrizes binárias, para então passarmos alguns modelos como o “relu” e “softmax”, por fim treinamos os modelos e exibimos os resultados como f-score e acurácia do modelo.

Nos resultados, temos o f-score de 0.1002 e uma acurácia de 0.96, assim o modelo realizou uma boa classificação, além desses resultados tivemos uma perda de 0.2196 e um tempo de 8s 17ms/step.

5. Estratégias de Avaliação

Escolhemos algumas estratégias como acurácia, precisão, f-score, recall e também a matriz de confusão, além de considerarmos o tempo de treinamento em cada modelo para avaliação de desempenho dos modelos utilizados. Um pouco das definições de cada uma: a acurácia procura indicar uma performance geral do do modelo. Dentre todas as classificações, quantas o modelo classificou corretamente.

Temos também a precisão que dentre todas as classificações de classe positiva que o modelo fez, quantas estão corretas e o recall que entre todas as situações de classe

positiva como valor esperado, quantas estão corretas, e por fim o f-score que seria a média harmônica entre precisão e recall.

Cada métrica utilizada tem sua peculiaridade que devemos considerar no momento da avaliação. Sem separar como melhor ou pior, entendemos que cada métrica se adapta melhor conforme o problema. Nas técnicas utilizadas, os resultados são superiores a 0.90% tanto para precisão como para f-score e a acurácia foi superior a 0.92% o que significa que os todos os modelos de certa forma realizaram boas classificações.

6. Comparativo de Desempenho dos Algoritmos

Modelo	Precisão	Recall	F1-score	Acurácia
Árvore de Decisão	92%	94%	92%	94%
Regressão Logística	92%	94%	92%	94%
Técnica KNN	92%	94%	92%	94%
Naive Bayes	92%	46%	57%	46%
Redes Neurais	10%	10%	10%	96%

7. Discussão da Comparação

Para tornar uma comparação justa entre todos os modelos utilizamos o mesmo processamento e divisão da base de dados na parte do treino e teste.

Os algoritmos classificadores possuem resultados similares e com alta porcentagem de acerto, com exceção do Naive Bayes e Redes Neurais, cujo os resultados apresentaram os piores desempenhos. Assim, utilizando tanto a árvore de decisão, regressão logística ou a técnica KNN encontramos bons resultados de acurácia e nas demais avaliações.

8. Conclusão

No mundo globalizado em que vivemos, onde as informações atravessam fronteiras com velocidade espantosa, a inteligência artificial vem se tornando cada vez mais comum, assim fazendo parte do dia-a-dia de muitas organizações, tornando essa necessidade capaz de influenciar diretamente nos negócios. Baseado nesse e em outros aspectos, este projeto teve o objetivo de realizar uma análise e comparação

simples utilizando diferentes modelos de algoritmos, mas que tem um poder muito grande e pode ser utilizado para muitos fins.

A partir da análise realizada concluímos que o pré processamento utilizado foi o mesmo para tornar uma comparação de resultados justa entre os modelos e além disso os modelos possuem sua particularidade o que leva a resultados diferentes.

Trazemos alguns resultados, como: acurácia superior a 94% em todos os modelos, de certa forma realizaram boas classificações. Também concluímos que o modelo de Naive Bayes e Redes neurais estão com os piores resultados, entendemos que não são modelos ideais para o nosso problema.

Para definir o algoritmo que melhor resolve o seu problema é sempre necessário analisar alguns fatores, como: complexidade, tempo e quantidade de dados. Existem várias formas de chegar a um resultado utilizando o Machine Learning, por esse motivo é importante selecionar o mais adequado ao problema em questão. Devemos também sempre levar em questão e analisar fatores importantes como a periculosidade de um erro e suas consequências, bem como o tempo levado para chegar a um resultado.

Referências

[1] Código realizado com a linguagem Python via colab: https://colab.research.google.com/drive/1z76S_RcYXyekhBsMkkr7FJ24L73VMICU?usp=sharing. NOV.2020.

[2] Learning Data Science — Predict Adult Income with Decision Tree: https://itnext.io/learning-data-science-predict-adult-income-with-decision-tree-ae8dd57a76cc. OUT.2020.

[3] Base de dados census utilizada: https://archive.ics.uci.edu/ml/datasets/census+income. SET.2020.

[4] Métricas de Avaliação - Qual a diferença de cada métrica https://medium.com/@vitorborbarodrigues/m%C3%A9tricas-de-avalia%C3%A7%C3%A3o-acur%C3%A1cia-precis%C3%A3o-recall-quais-as-diferen%C3%A7as-c8f05e0a513c. NOV.2020.