Classificando o desempenho dos candidatos do ENEM 2021 através de dados socioeconômicos

Gabriel Lopes de Souza, Kailane Eduarda Felix da Silva, Maria Luísa Mendes de Siqueira Passos, Mário da Mota Limeira Neto, Pedro César Guimarães Rodrigues
Centro de Informática, Universidade Federal de Pernambuco, Recife, Brasil
{gls6, kefs, mlmsp, mmln, pcgr}@cin.ufpe.br

***Resumo*—O objetivo deste projeto é utilizar os modelos de aprendizado de máquina para classificar o desempenho na prova de matemática dos estudantes que realizaram o ENEM 2021, através dos seus dados socioeconômicos. Utilizaremos a base dos microdados do ENEM 2021, disponibilizada pelo próprio Governo Federal do Brasil.**

***Index Terms*—ENEM, *Naïve Bayes*, classificação, desempenho, matemática**

I. Introdução

O Exame Nacional do Ensino Médio (ENEM) foi uma política criada em 1998 pelo governo federal brasileiro e visa avaliar o desempenho escolar dos estudantes ao término da educação básica. A partir de 2009, os resultados do Exame começaram a ser usados como critério de seleção para a maioria das universidades brasileiras. Atualmente, o ENEM é o principal meio de ingresso ao ensino superior no Brasil, contando com cerca de 5 milhões de participantes inscritos por ano.

Além disso, para realizar a inscrição no exame, os candidatos precisam responder um questionário socioeconômico, o qual conta com perguntas sobre sua renda familiar, seu acesso à internet, suas ocupações e a escolaridade dos seus pais. Neste projeto, iremos explorar as descrições estatísticas desse conjunto de dados, bem como analisar probabilisticamente a correlação entre os parâmetros sociais e econômicos e o desempenho final do candidato na prova de matemática.

II. Objetivos

Neste projeto, teremos como objetivo principal a classificação do desempenho dos estudantes inscritos no ENEM, mais especificamente na prova de matemática, entre duas classes: o candidato está entre os 15% melhores avaliados ou não. Dessa forma, buscaremos, através de uma exploração analítica, com base na análise do conjunto de dados disponibilizado pelo Governo Federal, o qual conta com um banco de respostas dos alunos inscritos e participantes no ENEM 2021, a correlação entre cada um dos parâmetros utilizados e o resultado final, ou seja, tentaremos traçar uma linha lógica a respeito do que a análise estará nos fornecendo, a fim de entender e mostrar, explicitamente, como a desigualdade socioeconômica no país é refletida em Exames Nacionais.

III. Justificativa

É notório e fatídico que o Brasil é, atualmente, um dos países mais desiguais do mundo, tanto em termos sociais quanto em termos econômicos, desde a educação básica precária até a dificuldade de acesso ao ensino superior, a qual está longe de ser democrática.

Além disso, com o avanço da pandemia do Coronavírus em 2020, perpetuando até 2021, a discrepância entre a educação dos alunos mais favorecidos economicamente e dos menos favorecidos, foi ainda mais acentuada e agravada. Muito além de dinheiro, a desigualdade social também é percebida, ao passo que estudantes pretos e de regiões mais afastadas do polo econômico, foram, também, os mais afetados com tudo isso. Logo, é claro que os jovens brasileiros não tiveram as mesmas condições de estudos, especialmente nesse período, ou seja, à medida que alunos puderam assistir aulas online, no conforto de casa e continuarem seus preparativos para a prova nacional, outros não tinham computador e nem muito menos internet em casa, o que elevou, ainda mais, a evasão escolar.

Nesse sentido, é válido discutir e buscar entender de que forma essa discrepância impacta, diretamente, a educação e os resultados do ENEM 2021, o qual foi realizado durante esse período pandêmico. Assim, escolhemos analisar esse tema de forma crítica, através dos dados dados coletados, a fim de revelar muito além da disparidade de notas no exame, mas também como o contexto socioeconômico do candidato está atrelado a esse resultado.

IV. Metodologia

Neste projeto, inicialmente faremos uma análise exploratória para entender como as variáveis se comportam, bem como visualizar suas correlações e distribuições.

Adiante, iremos propor a utilização de técnicas de aprendizado de máquina para classificação das notas do exame de matemática do ENEM, tomando como parâmetros os dados socioeconômicos. Para esse fim, iremos utilizar, principalmente, o algoritmo classificador *Naïve Bayes* da biblioteca *Scikit-learn*, com objetivo de comparar os resultados deste com outros algoritmos clássicos: KNN (*K-Nearest Neighbors*), Árvore de decisão e Regressão logística.

Desenvolveremos todo o código na linguagem *Python*, em um ambiente de desenvolvimento do *Google Colaboratory*.

Utilizaremos as principais bibliotecas para análise e modelagem de dados construídas para o *Python*: *Scipy*, *Numpy*, *Pandas*, *Matplotlib*, *Seaborn* e *Scikit learn*.

A. Dataset

Utilizaremos os microdados do ENEM mais recente, realizado no ano de 2021. O *dataset* é provido pelo INEP (Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira).

De maneira geral, o *dataset* pode ser dividido em três partes:

- A primeira parte contém informações preenchidas pelo candidato ao fazer a inscrição para a prova. Consiste nas seguintes características: idade, gênero, estado civil, etnia, grau de escolaridade, tipo da escola (pública ou privada), estado de residência, região de residência, etc.
- A segunda parte possui respostas do questionário socioeconômico, que possui perguntas como:
  - Até que série seu pai estudou?
  - Qual é a renda mensal de sua família?
  - Quantas pessoas moram atualmente na sua casa?
  - Na sua residência tem acesso à Internet?
- A última parte é referente a nota do estudante em cada uma das provas do exame: Matemática, Redação, Linguagens, Ciências Humanas e Ciências da Natureza.

B. Processamento do dataset

O processamento que iremos aplicar no *dataset* pode ser resumido em: (1) Filtro de Instâncias; (2) *Feature Selection*; (3) *Feature Engineering*.

1) Filtro de Instâncias: Nesta etapa, teremos que remover ou preencher os valores faltantes, dado que existe uma abundância de estudantes inscritos que não compareceram à aplicação das provas. Essas instâncias não serão úteis para a nossa análise.

2) Feature Selection: Nem todos os atributos mencionados serão utilizados como *inputs* para os modelos. Essa etapa consiste em selecionar um conjunto de *features* para o modelo de aprendizado. Esse conjunto de *features* precisa ter uma dimensão razoável. Dessa forma, não iremos incluir atributos muito específicos (ex: nome da escola que o aluno estudou), pois isso prejudicaria o desempenho dos modelos.

3) Feature Engineering: Esta etapa consiste em transformações das *features* para permitir que sejam utilizadas pelos algoritmos de aprendizado e, até mesmo, melhorar o desempenho deles. Precisaremos, por exemplo, transformar dados nominais em numéricos através de um *encoding*.

C. Algoritmos de aprendizagem de máquina

1) Naïve Bayes: O classificador *Naïve Bayes* é um dos modelos mais populares no aprendizado de máquina. O termo *Naïve*, (do inglês, ingênuo) se refere à premissa central do algoritmo de que os atributos considerados são não correlacionados entre si, ou seja, ele toma como pressuposto a suposição de independência entre as variáveis do problema, e, assim, o modelo em questão realiza uma classificação probabilística de observações, caracterizando-as em classes pré-definidas. Esse modelo, como o próprio nome indica, faz o uso do teorema de Bayes como princípio fundamental, se tornando uma aplicação direta do teorema homônimo.

O teorema de Bayes, o qual recebe esse nome por ser criado pelo pastor e matemático inglês Thomas Bayes (1702-1761), determina a probabilidade de um evento acontecer, diante de um conhecimento prévio que pode estar relacionado a este evento. Por isso, esse teorema é uma fórmula matemática que utiliza a probabilidade condicional, isso significa que, na teoria da estatística e probabilidade, esse teorema é uma forma de revisão das previsões diante de sólidas evidências. Sendo assim, por estimativas baseadas em um conjunto de indícios, é possível ter uma compreensão real daquele universo de dados. O manuscrito de Bayes só foi publicado após a morte de Thomas, editado significativamente por Richard Price antes disso.

$$P(A \mid B) = \frac{P(A)P(B \mid A)}{P(B)} \tag{1}$$

Por ser muito simples e rápido, o classificador de *Naïve Bayes* possui um desempenho relativamente maior do que outros classificadores. Além disso, ele precisa de um pequeno número de dados de teste para concluir as classificações com uma boa precisão. Diante disso, temos um modelo adequado para classificação de atributos discretos, o *Naïve Bayes* tem aplicações na análise de crédito, diagnósticos médicos, busca por falhas em sistemas mecânicos e, ainda, na análise de texto conforme a frequência das palavras usadas, o que o torna, também, comumente utilizado na classificação de *emails* como *spam*.

Neste projeto, utilizaremos o classificador de Bayes da biblioteca *Scikit-learn* para classificar o desempenho dos estudantes na prova de matemática. Utilizaremos 2 classes possíveis para o *output*, correspondentes a se aluno está entre os 15% melhores avaliados ou não.

2) KNN (K-Nearest Neighbors): O Algoritmo de KNN é um algoritmo que utiliza o aprendizado baseado em instâncias, isto é, as instâncias de treinamento brutas são usadas para fazer previsões. As previsões são feitas para uma nova instância (x) pesquisando todo o conjunto de treinamento para as K instâncias mais semelhantes (os vizinhos) e resumindo a variável de saída para essas instâncias de K.

Para determinar quais das instâncias do K no conjunto de dados de treinamento são mais semelhantes a uma nova entrada, uma medida de distância é usada. Para variáveis de entrada de valor real, a medida de distância mais popular é a distância euclidiana.

Quando o KNN é usado para classificação, a saída pode ser calculada como a classe com a maior frequência das instâncias mais semelhantes do K. Cada instância vota em sua classe e a classe com o maior número de votos é considerada a predição.

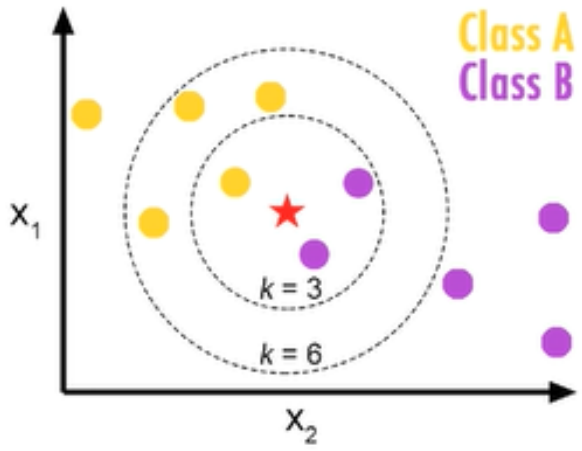

Figura 1. Como o KNN funciona visualmente

3) Regressão Logística: A regressão logística é outro algoritmo que lida com problemas de classificação. Esse modelo mede a relação entre a variável dependente categórica e uma ou mais variáveis independentes, estimando as probabilidades, usando uma função logística. Isso é, analisa diferentes aspectos ou variáveis de um objeto, para depois determinar uma classe na qual ele se encaixa melhor. No modelo de regressão logística binominal, os objetos são classificados em dois grupos ou categorias. Por exemplo, a mensagem é *spam* ou não, a imagem é colorida ou não, a célula é cancerígena ou não.

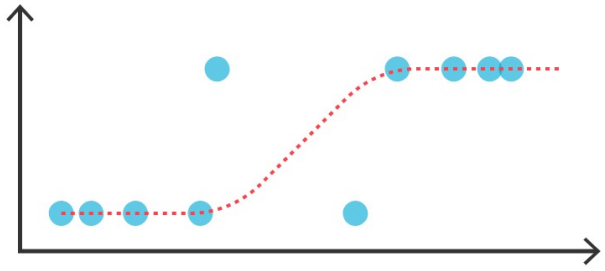

Figura 2. Exemplo de gráfico de uma regressão logística

4) Árvore de decisão: O algoritmo para árvores de decisão pode, também, ser utilizado em problemas de classificação. Nesse tipo de modelo, as informações são registradas em forma de árvore, os seus dados são distribuídos ao longo dos nós e o classificador trabalha em cima da incerteza do subconjunto de dados definido por cada nó. Na árvore de decisão, cada nó interno define uma *feature* da classificação que está em andamento, um conjunto de *features* indica uma *branch* da classificação e cada folha indica, finalmente, a classificação para determinado subconjunto daquele nó.

No algoritmo, para cada nova distribuição dos dados em novos nós, o ganho de informação é calculado, visando a redução da incerteza em relação à *features* anteriores. Essas distribuições se repetem até que as folhas tenham o resultado claro com a menor incerteza possível, a fim de classificar os objetos.

A principal vantagem desse tipo de classificador é, justamente, a sua distribuição dos dados em *features* e as regras de decisão, que são utilizadas em diferentes estágios da classificação.

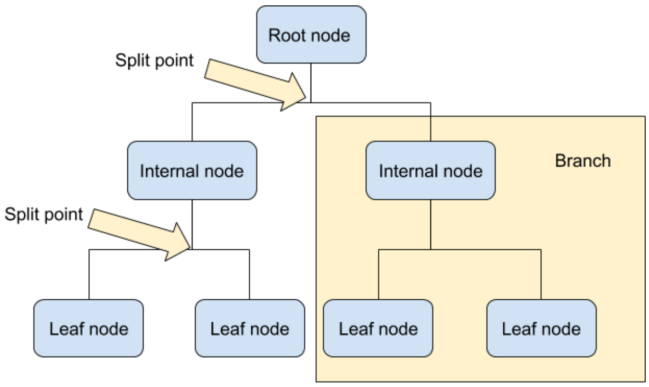

Figura 3. Exemplo estrutural de uma árvore de decisão

V. Análise Exploratória

Após fazermos uma visualização inicial da nossa base de dados, podemos observar que ela é dividida em três partes, informações preenchidas pelo candidato, respostas do questionário socioeconômico e as notas do estudante em cada uma das provas. Na nossa análise, iremos utilizar os dados que dizem respeito ao fator social e econômico do candidato e que tenham uma maior correlação com a nota de matemática.

A. Visualização de dados

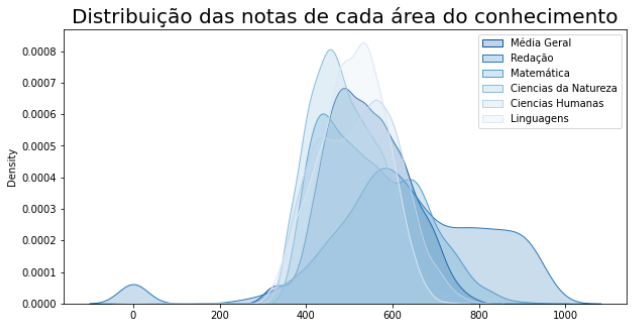

Figura 4.

1) Verificando a área que mais se aproxima da Média geral: Como podemos ver na figura 6, a distribuição das notas de matemática é muito próxima da distribuição das notas gerais, representada na figura 5, por isso, vamos utilizar a nota de matemática como base para nossas próximas análises.

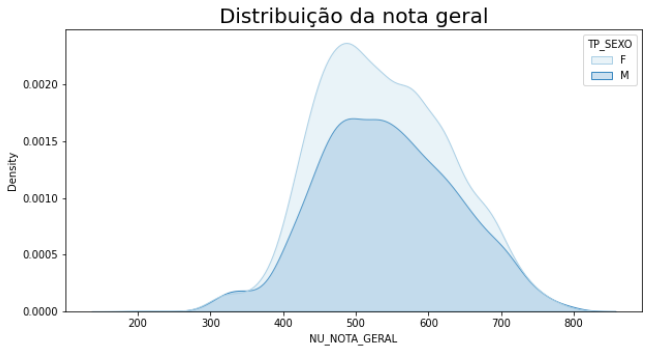

Figura 5.

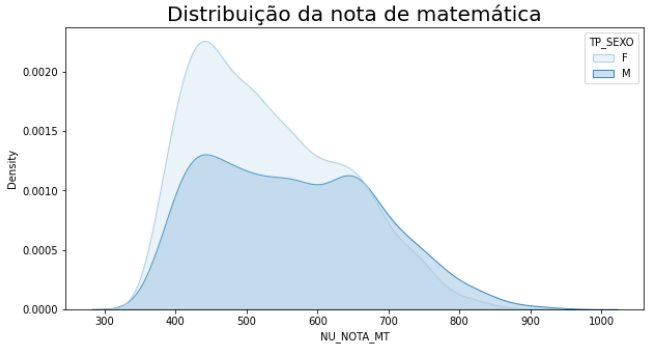

Figura 6.

2) Verificando a correlação e a distribuição dos fatores socioeconômicos com a nota de matemática: Podemos observar, pelas figuras a seguir, que existe uma grande relação entre fatores sociais e econômicos e a nota de matemática, sendo as pessoas com maior renda as que detêm das maiores notas, além de apresentar uma distribuição de notas com valor maior que os demais. Isso se deve, também, pela correlação entre renda e acesso à tecnologia, fazendo com que aqueles que têm computadores em casa tenham maiores notas. Ainda, podemos ver que a escolaridade dos pais afeta, diretamente, a escolaridade do filho, de tal forma que, quanto menor for o exemplo de estudos dentro de casa, mais propenso o jovem é a não se dedicar, também, ao ensino, impactando sua nota do ENEM.

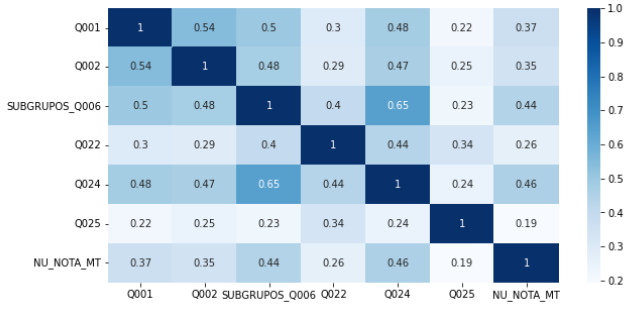

Figura 7.

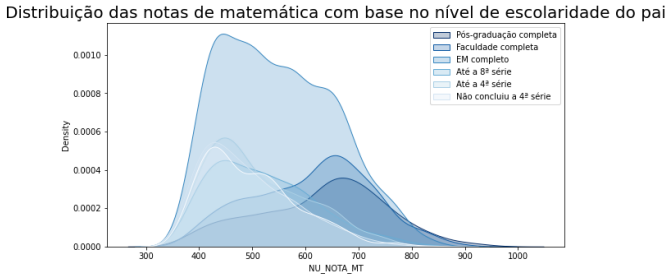

Figura 8.

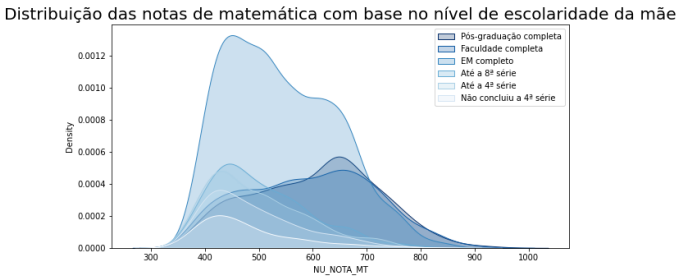

Figura 9.

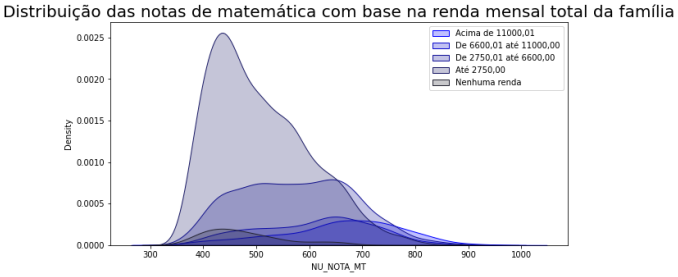

Figura 10.

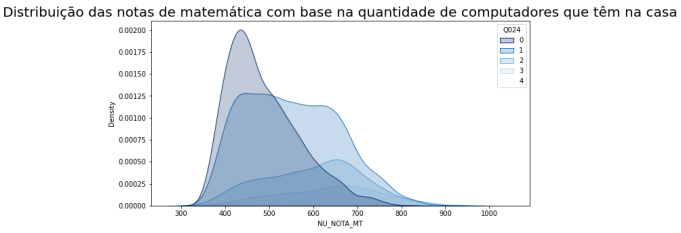

Figura 11.

3) Verificando a correlação e a distribuição de alguns dados do candidato e a nota de matemática: Podemos ver, através das imagens a seguir, que a maior parte dos candidatos é jovem, tem idade abaixo de 24 anos e que a escola possui correlação direta com as notas em matemática, existindo uma vantagem para os estudantes de escola particular, que detém das maiores notas. A cor da pele também influencia a nota do candidato, de forma que as pessoas brancas possuem as

maiores notas, embora sejam as que, em quantidade, estão no final da lista.

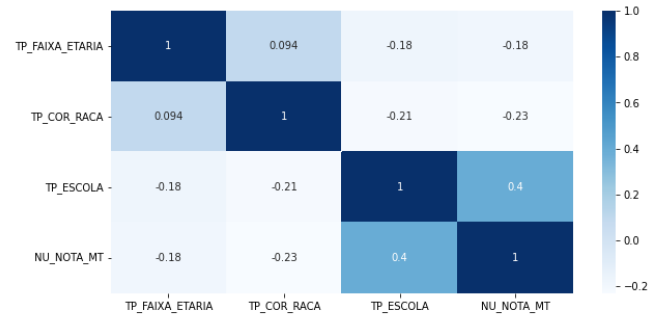

Figura 12.

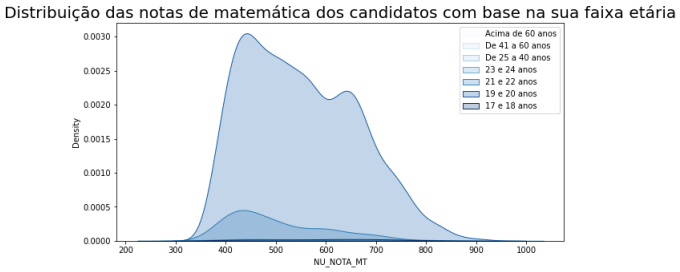

Figura 13.

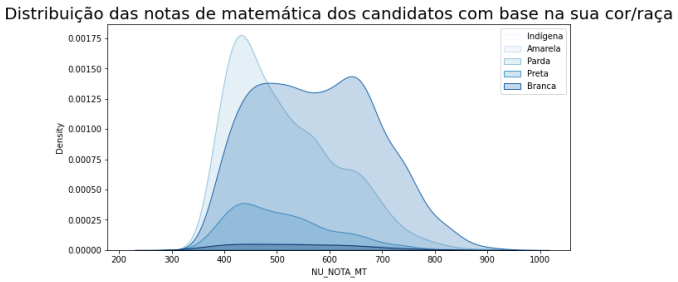

Figura 14.

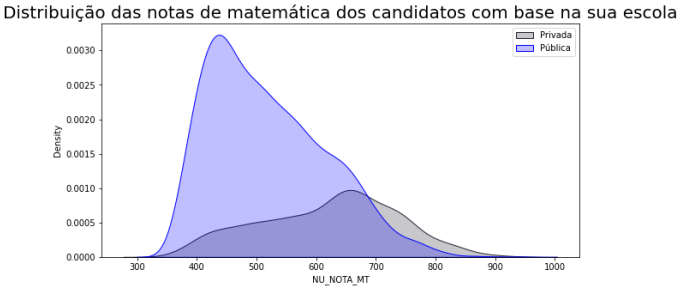

Figura 15.

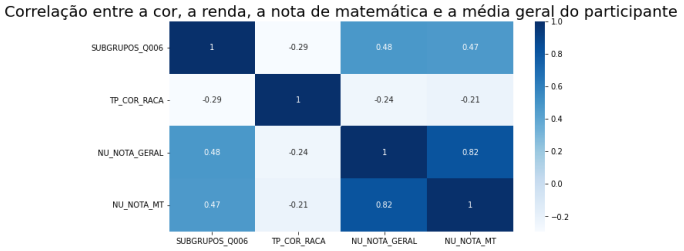

Figura 16.

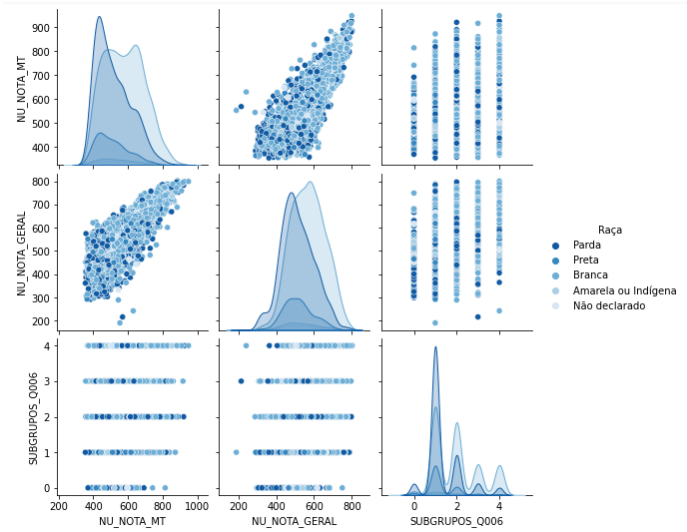

Figura 17.

4) A intercessão entre renda, raça e desempenho acadêmico no acesso ao ensino superior no Brasil: Podemos observar, pelas imagens 16 e 17 acima, que existe uma relação entre a cor dos participantes e sua renda e de ambas com a nota do participante, quanto maior for a sua renda, maior é a sua nota. Percebe-se, também, que, até mesmo nas mesmas classes de renda, há uma diferença de nota considerável por conta da cor da pele, onde as pessoas brancas têm as maiores notas e as pretas com menos de um salário mínimo têm as menores notas. Por fim, também podemos ver que a maioria dos candidatos é branco e possui uma renda entre 2750 e 6600 reais.

5) Distribuição das notas de matemática pelo Brasil: Analisando a figura abaixo, conseguimos perceber que as maiores notas de matemática estão localizadas na região sudeste e na região sul, justamente onde estão concentrados os maiores polos econômicos do país, ao passo que as menores notas estão concentradas na região norte. Isso nos mostra que as regiões com maior poder de compra e desenvolvidas socioeconomicamente, detém das maiores notas e que a região mais pobre e com menos recursos apresenta as menores notas, o que confirma e ratifica as nossas análises anteriores.

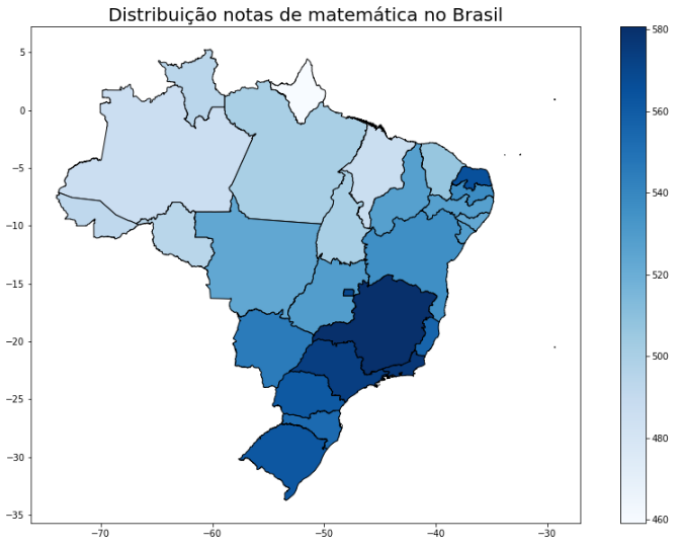

Figura 18.

VI. Análise dos Resultados

A. Métricas para avaliar um classificador

1) Acurácia: A acurácia é uma boa indicação geral de como o modelo performou. Porém, pode haver situações em que ela é enganosa. Por exemplo, na criação de um modelo de identificação de fraudes em cartões de crédito, o número de casos considerados como fraude pode ser bem pequeno em relação ao número de casos considerados legais. Para colocar em números, em uma situação hipotética de 280000 casos legais e 2000 casos fraudulentos, um modelo simplório que simplesmente classifica tudo como legal obteria uma acurácia de 99,3%. Ou seja, você estaria validando como ótimo um modelo que falha em detectar fraudes. Nesse sentido, para o nosso modelo, não é uma métrica ideal, já que o nosso *dataset* é bastante desbalanceado, ou seja, existem muitas ocorrências de pessoas que não atingiram o top 15% da nota de matemática e uma pequena ocorrência de pessoas que atingiram o top 15%. Dessa forma, nesse contexto, o modelo poderia simplesmente chutar que todas as pessoas não estão no top 15% e mesmo assim teria uma acurácia grande.

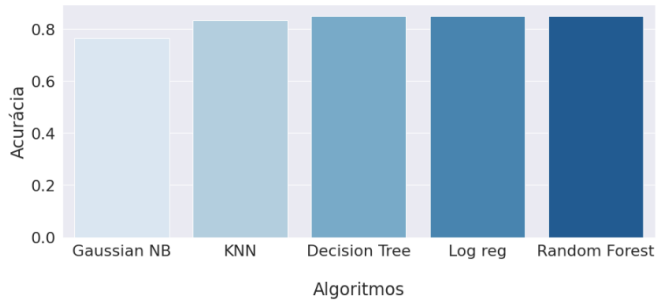

Figura 19. Comparando acurácias de cada modelo

2) Precisão: A precisão pode ser usada em uma situação em que os Falsos Positivos são considerados mais prejudiciais que os Falsos Negativos. Por exemplo, ao classificar uma ação como um bom investimento, é necessário que o modelo esteja correto, mesmo que acabe classificando bons investimentos como maus investimentos (situação de Falso Negativo) no processo. Ou seja, o modelo deve ser preciso em suas classificações, pois a partir do momento que consideramos um investimento bom quando na verdade ele não é, uma grande perda de dinheiro pode acontecer. Entretanto, de forma análoga ao que acontece com a acurácia, para o nosso modelo, a precisão também não é uma métrica ideal.

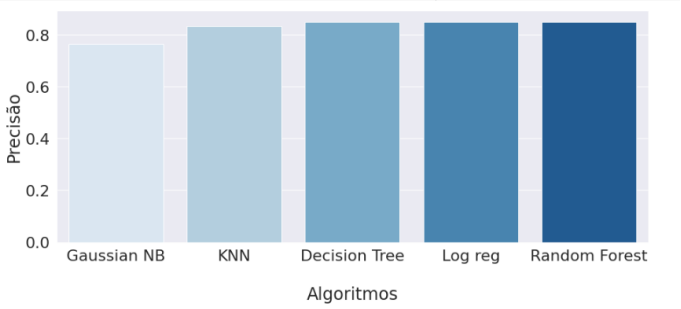

Figura 20. Comparando precisão de cada modelo

3) Matriz de confusão: Uma matriz de confusão é uma tabela que indica os erros e acertos do seu modelo, comparando com o resultado esperado, possui duas linhas e duas colunas que informam o número de verdadeiros positivos, falsos negativos, falsos positivos e verdadeiros negativos. É uma maneira bem interessante de visualizar o desempenho dos modelos classificadores por classe.

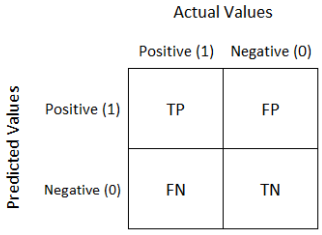

Figura 21. Como funciona uma matriz de confusão

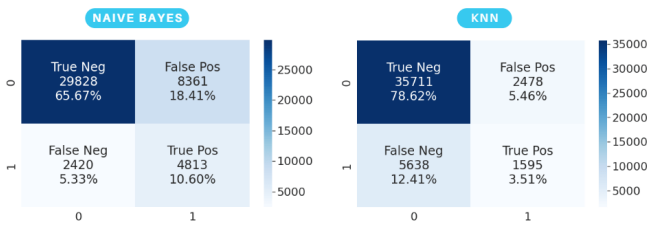

Figura 22. Comparando a matriz de confusão nesses modelos

4) Area Under the ROC Curve: A área sob uma curva ROC, abreviada como AUC, é um valor escalar único que mede o desempenho geral de um classificador binário. O valor da AUC está na faixa [0.5–1.0], onde o valor mínimo representa o desempenho de um classificador aleatório e o valor máximo corresponderia a um classificador perfeito, com taxa de erro de classificação equivalente a zero e que funciona para 100% dos casos. A AUC é uma medida geral robusta para avaliar o desempenho de classificadores de pontuação pois seu cálculo se baseia na curva ROC completa e, portanto, envolve todos os limites de classificação possíveis. Para o nosso modelo, é uma maneira bem interessante de avaliar, justamente porque não é tão simplista quanto a precisão e acurácia, além de levar em conta a matriz de confusão, de falsos positivos, falsos negativos. Para essa métrica, queremos chegar no valor mais próximo de 1.0, assim como todas as outras.

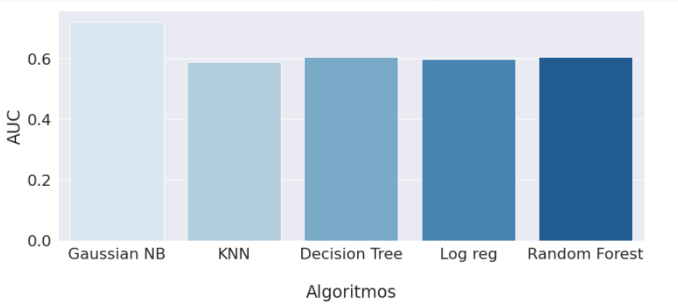

Figura 23. Comparando a AUC de cada modelo

VII. Conclusões finais

Tivemos a experiência de trabalhar com vários modelos de *Machine Learning* e desenvolver um projeto repleto de desafios. Esse conjunto de dados se trata de uma situação que todos nós já passamos, que é o vestibular. Analisar e fazer modelagens em dados que fazem sentido pra nós e pra a nossa realidade foi, sem dúvidas, o mais interessante.

Através da análise exploratória, ficou visível que o Exame Nacional do Ensino Médio reflete as desigualdades do nosso país. Além disso, identificamos muitos pontos de melhorias sobre o uso dos algoritmos classificadores, já que não obtivemos o resultado esperado em uma importante métrica de avaliação de desempenho.

Uma das possíveis explicações para o *Naive Bayes* performar melhor que outros modelos mais sofisticados é sobre o tamanho da amostra que estávamos utilizando, bem como sobre a distribuição das ocorrências positivas (o candidato está nos 15%) e as ocorrências negativas (o candidato não está nos 15%).

Continuaremos trabalhando nesse conjunto de dados para disponibilizar esse trabalho para quem se interessar, principalmente pelo seu caráter crítico, mas também pela quantidade de conceitos estatísticos essenciais que trabalhamos nesse projeto.

Referências

[1] Classificador Naive bayes da biblioteca Scikit Learn [https://scikit-learn.org/stable/modules/naive_bayes.html]
[2] Naive Bayes, Clearly Explained [https://www.youtube.com/watch?v=O2L2Uv9pdDA]
[3] Logistic Regression [https://en.wikipedia.org/wiki/Logistic_regression]
[4] O que é árvore de decisão [https://www.lucidchart.com/pages/pt/o-que-e-arvore-de-decisao]
[5] O Algorítmo K-Nearest Neighbors (KNN) Em Machine Learning [https://portaldatascience.com/o-algoritmo-k-nearest-neighbors-knn-em-machine-learning/]
[6] Modelos de Machine Learning: uma comparação entre os modelos [https://medium.com/gbtech/modelos-de-machine-learning-uma-compara%C3%A7%C3%A3o-entre-os-modelos-parte-1-c772661c7163]
[7] Microdados do Enem [https://www.gov.br/inep/pt-br/acesso-a-informacao/dados-abertos/microdados/enem]
[8] Um candidato de 98 anos fez o ENEM!- Análise de Dados [https://www.youtube.com/watch?v=Gn6Fm3iDbd8t=570s]
[9] Desigualdade educacional no Brasil é agravada pela pandemia [https://sites.ufop.br/lamparina/blog/desigualdade-educacional-no-brasil-é-agravada-pela-pandemia]
[10] Métricas de avaliação [https://vitorborbarodrigues.medium.com/métricas-de-avaliação-acurácia-precisão-recall-quais-as-diferenças-c8f05e0a513c]