Análise de Dados para Trocatine

Empresa: Trocatine

Integrantes:

Gabriela Machado Ribeiro (N°7)

Sarah de Moraes Ferreira Batista (°17)

Sumário

[Introdução 2](#_Toc181770215)

[Objetivo do aplicativo 3](#_Toc181770216)

[Objetivo da Análise Exploratória de Dados 3](#_Toc181770217)

[Objetivo da Análise de Dados (para Aprendizado de Máquina) 3](#_Toc181770218)

[Levantamento dos dados na análise exploratória 4](#_Toc181770219)

[Busca dos dados 4](#_Toc181770220)

[Descrição da base de dados de trabalho 4](#_Toc181770221)

[Faixa Etária 4](#_Toc181770222)

[Flag Filho 5](#_Toc181770223)

[Faixa Etária Filho 6](#_Toc181770224)

[Cidade 6](#_Toc181770225)

[Compra Roupa Usada 7](#_Toc181770226)

[Dificuldade Compra 7](#_Toc181770227)

[Dificuldade Infantil 8](#_Toc181770228)

[Faixa Salarial 8](#_Toc181770229)

[Forma Trocou 9](#_Toc181770230)

[Gênero 9](#_Toc181770231)

[Objetivo Compra 10](#_Toc181770232)

[Reciclagem Tecido 10](#_Toc181770233)

[Segurança Aplicativo 11](#_Toc181770234)

[Interesse 11](#_Toc181770235)

[Visualização dos Dados Sujos 12](#_Toc181770236)

[Limpeza dos dados 13](#_Toc181770237)

[Visualização dos Dados (após serem limpos) 14](#_Toc181770238)

[Condicionamento para alimentar o modelo de ML 14](#_Toc181770239)

[Condicionamento inicial 15](#_Toc181770240)

[Definição dos objetivos e das classes 15](#_Toc181770241)

[Definição dos modelos mais adequados para analisar os dados 15](#_Toc181770242)

[Descrição dos modelos selecionados 16](#_Toc181770243)

[Modelo 1: Naive Bayes (GaussianNB) 16](#_Toc181770244)

[Modelo 2: Tree (DecisionTreeClassifier) 16](#_Toc181770245)

[Modelo 3: KNN (KNeighborsClassifier) 16](#_Toc181770246)

[Aplicação dos modelos selecionados 16](#_Toc181770247)

[Análise dos Resultados 17](#_Toc181770248)

[Modelo 1: Naive Bayes 17](#_Toc181770249)

[Modelo 2: Tree 18](#_Toc181770250)

[Modelo 3: KNN 19](#_Toc181770251)

[Comparação Entre Modelos: 20](#_Toc181770252)

[Ajustes Necessários 20](#_Toc181770253)

[Identificação de Problemas: 20](#_Toc181770254)

[Mudanças na Base de Dados: 20](#_Toc181770255)

[Ajustes nos Modelos: 22](#_Toc181770256)

[Impacto das Modificações 22](#_Toc181770257)

[Modelo 1: [Nome do Modelo] 22](#_Toc181770258)

[Modelo 2: [Nome do Modelo] 22](#_Toc181770259)

[Modelo n [Nome do Modelo] 22](#_Toc181770260)

[Comparação Entre Modelos: 22](#_Toc181770261)

[Conclusão 22](#_Toc181770262)

# Introdução

Vs2 -

O Trocatine busca proporcionar um espaço seguro e de fácil acesso, para vendas e trocas de artigos infantis. Ele visa os problemas enfrentados com o descarte de resíduos têxteis, o qual é ligado a segunda indústria mais poluidora do mundo de acordo com o levantamento realizado pela Global Fashion Agenda, as indústrias têxteis. Esta situação é agravada quando é colocada a ótica em direção a famílias com crianças pequenas, já que elas crescem de forma rápida e precisam trocar frequentemente suas roupas. Além disso, as trocas de interesses durante as fases do crescimento de um indivíduo, despertam a necessidade de se desfazer de objetos que seus filhos não utilizam mais.

Logo o intuito é que o aplicativo seja utilizado por esses responsáveis que usualmente não sabem como descartar ou como proceder diante da situação, desta forma são levados frequentemente a decisão de descartar no lixo comum os itens, até mesmo os que estão em bom estado, e comprar novos itens com preços absurdos.

Vs1 -

O Trocatine busca proporcionar um espaço seguro e de fácil acesso, para vendas e trocas de artigos infantis. O intuito é que o aplicativo seja utilizado por famílias com crianças pequenas que desejam se desfazer de objetos que seus filhos não utilizam mais ou, que como crescem de forma rápida, devem trocá-los frequentemente. Entretanto as mesmas que não sabem como descartar ou como proceder diante da situação, desta forma são levadas frequentemente a decisão de descartar no lixo comum os itens, até mesmo os que estão em bom estado, e comprar novos itens com preços absurdos.

## Objetivo do aplicativo

O objetivo do aplicativo é auxiliar a introduzir a sustentabilidade na vida de famílias e incentivar a troca de itens infantis entre as pessoas que realmente precisam, desta forma evitando o descarte indevido desses itens e incentivando a economia circular. Além de disponibilizar os produtos por um preço mais acessível já que é algo essencial para as famílias as quais o nosso aplicativo atende, pela perspectiva dos preços exorbitantes os quais são ofertados os objetos no mercado infantil.

# Objetivo da Análise Exploratória de Dados

Montar, analisar e limpar um conjunto de dados relacionado ao aplicativo para ser utilizado em análise de dados exploratória e com a finalidade de alimentar uma IA.

Os dados que serão incluídos nesse conjunto podem ser provenientes de bases de dados governamentais (Brasil, Estados Unidos, União Europeia etc.), científicas ou de sondagem própria das equipes envolvidas (Forms, por exemplo).

# Objetivo da Análise de Dados (para Aprendizado de Máquina)

Vs1 -

Utilizando a base de dados montada durante a análise exploratória de dados a finalidade da inteligência artificial será determinar se uma pessoa, com base no seu perfil, é um cliente em potencial para o aplicativo ou não. Para alcançar este objetivo, faremos testes com os diferentes tipos de modelos que foram ensinados em sala de aula, os quais são:

* KNeighborsClassifier (neighbors)
* GaussianNB (naive\_bayes)
* DecisionTreeClassifier (tree)

Liste quais os modelos que serão utilizados e o escreva o que vão prever (se uma pessoa é, ou não, um cliente em potencial para o seu APP, qual o seu nível do interesse, quais produtos podem interessar a ela, quais meios de comunicação podem atingir essa pessoa, etc.

# Levantamento dos dados na análise exploratória

## Busca dos dados

A coleta de dados feita pela equipe foi realizada por meio de um formulário idealizado de forma online. O questionário foi montado visando atender o objetivo do modelo, o qual é compreender o perfil de potenciais usuários, desta forma por meio desta pesquisa é possivel entender as preferencias e características do público que se interessa pela proposta declarada pelo aplicativo em questão, o Trocatine. A pesquisa perdurou durante **68** dias, obteve **232** respostas e demorou em média **08:03** minutos para ser preenchido por cada indivíduo, ele continha **19** perguntas pelo seu desenvolvimento.

Aqui, o grupo descreve os critérios e os métodos para buscar os dados. Por exemplo:

* Levantamento de dados por questionário criado pelo próprio grupo
* Dados em bases oficiais/do governo/da ONU
* Bases de veículos de comunicação – jornais, revistas especializadas (não serve a Caras)

Coloque aqui os questionários que a equipe de Dados fez e indique quantas pessoas responderam. Se os questionários foram feitos pelos outros integrantes da equipe do Projeto (Dev ou 1° ano), não é necessário incluir aqui.

Explique por que é importante esses dados que foram escolhidos, qual a importância deles para o trabalho. Essas justificativas podem ser colocadas em uma seção separada, ou então indicadas junto da descrição da busca de dados.

# Descrição da base de dados de trabalho

Priorizando um entendimento detalhado a respeito da base de dados, as colunas utilizadas vão ser demonstradas como seções que apresentarão: tipo dos dados, o que ele é, valores-limite e parâmetros estatísticos. Os parâmetros estatísticos seriam:

* count: o número de entradas não nulas.
* unique: o número de valores únicos.
* top: o valor mais frequente.
* freq: a frequência do valor mais comum.

As colunas apresentadas serão apenas as utilizadas após a limpeza por conta da quantidade exacerbada de colunas que havia anteriormente, ademais ao final será apresentado como está se comportando os dados dessas colunas da forma a qual eles foram coletados.

### Faixa Etária

Tipo – String

O que ele é – Indica a idade da pessoa que está respondendo a pesquisa.

Valores Limites –

Menos de 18 anos

Entre 18 e 24 anos

Entre 25 e 34 anos

Entre 35 e 44 anos

Entre 45 e 54 anos

Entre 55 e 59 anos

60+

Parâmetros Estatísticos –

Texto

Descrição gerada automaticamente

### Flag Filho

Tipo – Boolean

O que ele é – Uma indicação se o individuo tem filhos ou não.

Valores Limites –

Sim – 0

Não – 1

Parâmetros Estatísticos –

Tabela

Descrição gerada automaticamente com confiança baixa

### Faixa Etária Filho

Tipo – String

O que ele é – Se ele tiver filhos, uma indicação da idade do filho dele.

Valores Limites –

Menos de 8 anos

Entre 9 e 14 anos

Entre 15 e 17 anos

Entre 18 e 24 anos

Entre 25 e 34 anos

Entre 35 e 44 anos

Entre 45 e 54 anos

Entre 55 e 59 anos

60+

Parâmetros Estatísticos –

Texto

Descrição gerada automaticamente

### Cidade

Tipo – String

O que ele é – A cidade que a pessoa que está respondendo mora.

Valores Limites –

Poderia ser escrito de forma livre.

Parâmetros Estatísticos –

Uma imagem contendo Texto

Descrição gerada automaticamente

### Compra Roupa Usada

Tipo – Boolean

O que ele é – Verificação de aderência a compra de roupas usadas, logo se o indivíduo compraria roupas usadas.

Valores Limites –

Sim – 1

Não – 0

Parâmetros Estatísticos –

Uma imagem contendo Texto

Descrição gerada automaticamente

### Dificuldade Compra

Tipo – String

O que ele é – Dificuldades encontradas ao comprar suas próprias roupas baseado nas opções disponíveis.

Valores Limites –

Preço

Crescimento

Qualidade

Parâmetros Estatísticos –

Texto

Descrição gerada automaticamente

### Dificuldade Infantil

Tipo – String

O que ele é – Dificuldades encontradas ao comprar roupas infantis baseado nas opções disponíveis.

Valores Limites –

Preço

Crescimento

Qualidade

Parâmetros Estatísticos –

Texto

Descrição gerada automaticamente

### Faixa Salarial

Tipo – String

O que ele é – A faixa salarial que se encaixa a renda do individuo.

Valores Limites –

Classe A: superior a R$ 22 mil

Classe B: entre R$7,1 mil e R$ 22 mil

Classe C: entre R$ 2,9 mil e R$ 7,1 mil

Classe D/E: até R$ 2,9 mil

Parâmetros Estatísticos –

Texto

Descrição gerada automaticamente

### Forma Trocou

Tipo – String

O que ele é – Se a pessoa já obteve alguma experiencia com qualquer forma de troca de itens.

Valores Limites –

Trocou

Doou

Recebeu

Parâmetros Estatísticos –

Uma imagem contendo Texto

Descrição gerada automaticamente

### Gênero

Tipo – String

O que ele é – O gênero que a pessoa se identifica.

Valores Limites –

Feminino

Masculino

Prefiro não me identificar

Parâmetros Estatísticos –

Uma imagem contendo Texto

Descrição gerada automaticamente

### Objetivo Compra

Tipo – String

O que ele é – Qual seria o objetivo principal da pessoa ao realizar uma compra online.

Valores Limites –

Chegue mais rápido

Comprar mais barato

Parâmetros Estatísticos –

Texto

Descrição gerada automaticamente

### Reciclagem Tecido

Tipo – Boolean

O que ele é – Se o individuo conhece locais de coleta de tecido.

Valores Limites –

Sim – 1

Não – 0

Parâmetros Estatísticos –

Texto

Descrição gerada automaticamente

### Segurança Aplicativo

Tipo – Boolean

O que ele é – Se a pessoa se sente segura ao realizar compras online.

Valores Limites –

Sim – 1

Não – 0

Parâmetros Estatísticos –

Texto

Descrição gerada automaticamente

### Interesse

Tipo - Boolean

O que ele é – Se o objetivo do aplicativo seria algo que interessa a pessoa.

Valores Limites –

Sim – 1

Não – 0

Parâmetros Estatísticos –

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

### Visualização dos Dados Sujos

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Aqui, faça como nas atividades do primeiro semestre. Inclua coisas como:

* Tipos dos dados: inteiro, data/hora, string etc.
* Valores-limite: máximo e mínimo para dados numéricos, lista de valores para dados categóricos
* Parâmetros estatísticos que descrevem os dados: média, moda, desvio padrão, variância etc.

Sempre indique como fez a análise, seja com seu código Python próprio, ou dizendo qual a biblioteca de análise exploratória de dados (por exemplo, ydata-profiling).

No relatório, coloque os resultados da análise dos dados brutos (parâmetros estatísticos e gráficos). O código (notebook) deve ser fornecido em um anexo ou como um link para o GitHub.

Para plotagem dos dados brutos encontrados, monte um dashboard, de acordo com os requisitos da disciplina Business Intelligence (aproveite o que for desenvolvido para essa disciplina).

# Limpeza dos dados

Para a preparação dos dados, foi analisado como se comportavam os valores de cada coluna

e com base nesta visão geral foram realizadas as ações seguintes. Durante o procedimento, foi excluído as colunas de: ID, Nome, Hora da última modificação, Email, Hora de início, Hora de conclusão. Além dessas foram excluídas outras 5 colunas que eram de escrita livre, os motivos da remoção foram: ID’s duplicados, nulos em quantidade massiva, descritivos que não agregavam a pesquisa. Após a exclusão destas colunas, foi excluída também uma linha a qual os dados claramente não foram preenchidos da forma correta.

Após alterar o nome das colunas, foi encontrado inconformidades em 4 colunas: Dificuldade infantil, Dificuldade compra, Forma Trocou, Cidade. As 3 primeiras colunas estavam com informações duplicadas por conta da ordem as quais elas apareciam, logo foi criado uma ordem padrão para essas colunas, as quais seriam:

Dificuldade Infantil e Dificuldade Compra:

* Preco;
* Qualidade;
* Crescimento;
* Preco;Qualidade;
* Preco;Crescimento;
* Qualidade;Crescimento;
* Preco;Qualidade;Crescimento;

Forma Trocou:

* Doou;
* Trocou;
* Recebeu;
* Doou;Trocou;
* Doou;Recebeu;
* Trocou;Recebeu;
* Doou;Trocou;Recebeu;

A respeito da coluna cidade, foi criada uma função para consertar o regex já que os valores se repetiam com pequenas diferenças, por exemplo um espaço, e ele foi criado de acordo com as inconformidades presentes no dataset.

### Visualização dos Dados (após serem limpos)

Interface gráfica do usuário, Site

Descrição gerada automaticamente

Descrição do procedimento de limpeza e preparação dos dados para poder prosseguir para a análise preditiva, ou seja:

* quais colunas/campos foram limpos e por quê (remoção de outliers, dados claramente incorretos etc)
* gráficos que ajudem na justificativa
* como transformou os dados (mudança de tipo, arredondamento etc)

No relatório, coloque os resultados da limpeza (parâmetros estatísticos e gráficos). O código (notebook) deve ser fornecido em um anexo ou como um link para o GitHub.

Para plotagem dos dados limpos, monte um dashboard, de acordo com os requisitos da disciplina Business Intelligence (aproveite o que for desenvolvido para essa disciplina).

# Condicionamento para alimentar o modelo de ML

A preparação dos dados ocorrerá antes da execução do modelo de aprendizado de máquina. Além disso, será feita novamente após a execução do modelo, caso seja identificado que ajustes são necessários para um novo processamento.

O condicionamento de dados será realizado antes de rodar o modelo de ML. Também será feito após rodar o modelo de ML e for constatado que é necessário algum ajuste para reprocessamento.

Este item deverá conter uma subseção para cada etapa de condicionamento dos dados.

## Condicionamento inicial

* Normalização do banco de dados (Ordinal Encoder e Label Encoder)
* OverSampling (já que foi apresentado uma pequena amostra de registros, não foi considerado o teste de UnderSampling), para teste de qual seria a melhor forma de realizar a técnica de gerar novos dados genéricos, desta forma 7 bases geradas de funções distintas são testadas.
* Avaliação dos dados preparados: Dashboard de Business Intelligence.
* Normalização, padronização
* Redução de dimensionalidade
* Escolha de uma variável dentro de um conjunto de variáveis com alta correlação  
  etc.
* Avaliação dos dados preparados: Dashboard de Business Intelligence.

AQUI COMEÇA A PARTE DE ANÁLISE DE DADOS (MÓDOLO)

# Definição dos objetivos e das classes

O objetivo projetado para este trabalho seria entender quais são os padrões apresentados pelas pessoas interessadas na proposta do aplicativo, para assim com base neste entendimento prever quais pessoas se interessariam pelo aplicativo. Desta forma, responderia à pergunta “**Você estaria interessado/a em um aplicativo com estas funções?**” que prevê as classes “**Sim**” e “**Não**”. Esta ferramenta irá agregar muito a respeito da percepção de negócio do aplicativo.

Explicar em detalhes os objetivos que desejam prever com a aplicação dos modelos. Escrever quais são as perguntas ou problemas específicos que precisam resolver com a aplicação dos modelos.

Listar e descrever a(s) resposta(s) (y) e as classes que cada uma dessas respostas vai prever.

# Definição dos modelos mais adequados para analisar os dados

Escrever uma visão geral inicial dos modelos que levantaram para serem aplicados nas análises e quais critérios utilizados para selecionar aqueles que serão aplicados nos seus dados. (Aqui é uma explicação teórica antes de aplicar os modelos aos dados)

Apresentar uma visão geral dos modelos de classificação que analisaram para aplicar aos seus dados. Explicar por que esses modelos são potenciais candidatos para resolver o seu problema de classificação e os critérios utilizados para essa seleção, considerando aspectos como:

* Natureza dos Dados: dados categóricos ou contínuos que precisam ser classificados.
* Complexidade dos Modelos: modelos simples ou modelos mais complexos.
* Objetivos de Classificação: tarefa é binária ou multi-classe.

# Descrição dos modelos selecionados

Descrever cada modelo selecionado em uma subseção a seguir (criar quantas subseções forem necessárias para os modelos selecionados). Escrever os critérios que utilizou para seleção do modelo, considerando aspectos como:

* Precisão e Robustez: capacidade do modelo de classificar corretamente novos dados e lidar com variações nos dados.
* Interpretação e Explicabilidade: facilidade em interpretar os resultados e entender como as previsões são feitas.
* Desempenho Computacional: tempo e recursos necessários para treinar e aplicar o modelo.
* Capacidade de Generalização: habilidade do modelo de manter um bom desempenho em dados não vistos.

## Modelo 1: Naive Bayes (GaussianNB)

Descrever o modelo 1.

Escrever os critérios que utilizou para seleção do modelo 1.

## Modelo 2: Tree (DecisionTreeClassifier)

Descrever o modelo 2.

Escrever os critérios que utilizou para seleção do modelo 2.

## Modelo 3: KNN (KNeighborsClassifier)

Descrever o modelo n.

Escrever os critérios que utilizou para seleção do modelo n.

# Aplicação dos modelos selecionados

A aplicação do modelo foi separada em etapas, com o objetivo de alcançar o melhor resultado possivel. A primeira seria a das separações das bases, elas foram separadas em treino (80%) e teste (20%). Essa separação foi aplicada para oito bases que serão testadas, a primeira delas seria a base original sem alterações e as outras sete são as bases geradas que foram citadas anteriormente.

Após este procedimento, foi efetuada uma validação cruzada com 5 divisões em conjunto com a seleção dos melhores hiperparâmetros utilizando a função GridSearchCV em função de apenas os dados de treino, isso foi usado como ferramenta para testar quais bases seriam melhores para cada modelo. Desta forma foi analisado os resultados de cada divisão e ajustado hiperparâmetros com base no resultado do f1-score, já que ele é uma medida que tem uma boa visão a respeito do equilíbrio entre os resultados do precision e do recall.

Explicar como cada modelo foi aplicado aos dados, como foram feitos a divisão, o treinamento e a avaliação dos modelos.

* Divisão dos Dados: detalhar a divisão dos dados em conjuntos de treinamento e teste (ex.: 80% treinamento e 20% teste, validação cruzada, etc).
* Treinamento: explicar o processo de treinamento para cada modelo, incluindo ajuste de hiperparâmetros.
* Métricas de Avaliação: apresentar as métricas utilizadas para avaliar o desempenho dos modelos, como acurácia, precisão, recall e F1-score, e descrever como essas métricas ajudam a avaliar a eficácia dos modelos.

# Análise dos Resultados

Na etapa de teste é feita uma bateria de testes, visando verificar se haveria oscilações nas métricas e seu tempo de execução. Durante a realização, foi possivel observar que as métricas não mudaram significativamente, logo apenas um dos testes será mostrado como exemplo. Enquanto os tempos de execução foram:

Gráfico, Gráfico de barras

Descrição gerada automaticamente

Discutir os resultados obtidos para cada modelo com base nas métricas de avaliação.

## Modelo 1: Naive Bayes

Métricas de Avaliação:

Gráfico

Descrição gerada automaticamente

*Resultados do f1 score apresentado durante o cross-validation realizado em cada uma das bases com o modelo de GaussianNB.*

Calendário

Descrição gerada automaticamente

*Resultados das métricas apresentadas durante o teste realizado com o modelo de GaussianNB que apresentou o melhor desempenho abaixo de 98% com a base que interagiu melhor.*

Gráfico

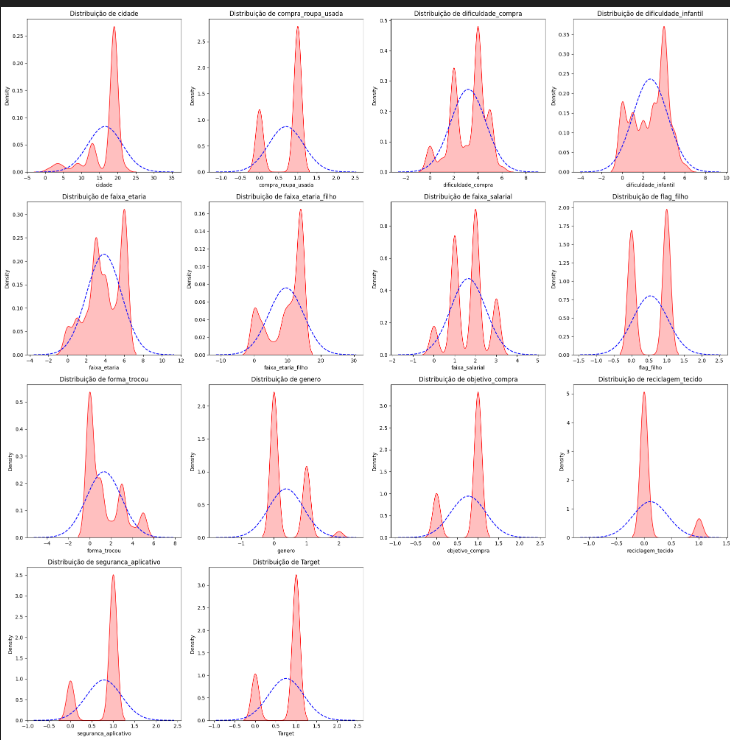
Descrição gerada automaticamente

*Matriz de confusão gerada a partir da predição do modelo de aprendizado.*

Discussão dos Resultados:

Por conta da falta de equilíbrio entre as classes, foi apresentado uma acurácia maior, todavia os resultados apresentados pela precisão e o recall da classe 0 (yes) são extremamente baixos. Essas informações demonstram a dificuldade do modelo em lidar com a classe minoritária, mas para além disso é possivel observar no gráfico de F1 score, que sua interação com todas as bases, sua dificuldade de se adaptar ao contexto do aplicativo. Isso é detectável por seu desempenho baixo em bases balanceadas, alcançando métricas maiores apenas quando está desbalanceado, pois desta forma quando aponta apenas para a classe majoritária tem a falsa sensação de um bom resultado.

Nos princípios do modelo GaussianNB os dados quantitativos deveriam seguir uma distribuição normal, quando normalizamos a base com o OrdinalEncoder ele passa a tratar as colunas qualitativas da mesma forma. A base utilizada na construção desse modelo é feita somente por colunas qualitativas, logo para verificar se este era o problema foi feito um gráfico com as distribuições dos valores de cada coluna.



*Gráfico com a distribuição dos valores de cada coluna da base de dados que o modelo Naive Bayes utiliza, sendo a linha azul a distribuição esperada e a área vermelha o que é realmente encontrado na base.*

Analisando os resultados apresentados é possivel concluir que as colunas não seguem o padrão esperado, podemos deduzir que este seria um dos motivos para o modelo não conseguir se adaptar da melhor forma.

Métricas de Avaliação: apresentar as métricas de avaliação obtidas para o modelo.

Discussão dos Resultados: analisar como o modelo performou em cada uma das métricas e o que esses resultados indicam sobre seu desempenho, considerando aspectos como:

* Equilíbrio entre Precisão e Recall: se o modelo está tendendo a obter alta precisão mas baixo recall, ou vice-versa.
* Classes Desbalanceadas: se o modelo tem dificuldade em classificar classes menos frequentes.
* Erro e Overfitting: se o modelo apresenta sinais de overfitting, ou seja, se ele tem um desempenho muito bom no conjunto de treinamento mas ruim no conjunto de teste.

## Modelo 2: Tree

Métricas de avaliação:

Gráfico, Gráfico de barras

Descrição gerada automaticamente com confiança média

*Resultados do f1 score apresentado durante o cross-validation realizado em cada uma das bases com o modelo de DecisionTreeClassifier.*

Calendário

Descrição gerada automaticamente

*Resultados das métricas apresentadas durante o teste realizado com o modelo de DecisionTreeClassifier que apresentou o melhor desempenho abaixo de 98% com a base que interagiu melhor.*

Gráfico, Gráfico de mapa de árvore

Descrição gerada automaticamente

*Matriz de confusão gerada a partir da predição do modelo de aprendizado.*

Discussão dos Resultados:

O cross-validation do modelo foi seguida uma boa estabilidade do F1 score nas bases, entre 85% e 100%, conquanto tenha apresentado algumas vezes a métrica alta demais de forma propicia a um overfitting. Durante o teste da melhor combinação entre hiperparâmetros e base, foi demonstrado um ótimo equilíbrio entre a precisão e o recall de ambas as classes. Desta forma, resultou em ótimos resultados de acurácia também e nenhuma métricas ficou abaixo de 80%.

## 

## Modelo 3: KNN

Métricas de Avaliação:

Gráfico

Descrição gerada automaticamente com confiança média

*Resultados do f1 score apresentado durante o cross-validation realizado em cada uma das bases com o modelo de KNeighborsClassifier.*

Tela preta com letras brancas

Descrição gerada automaticamente

*Resultados das métricas apresentadas durante o teste realizado com o modelo de KNN que apresentou o melhor desempenho (abaixo de 98%) com a base que interagiu.*

Gráfico

Descrição gerada automaticamente

*Matriz de confusão gerada a partir da predição do modelo de aprendizado.*

Discussão dos Resultados:

O modelo apresentou estar com o F1 score elevado na interação da maioria das bases, apresentando ter se adaptado muito bem em relação a base. Isso é possivel ser comprovado durante o teste também. Entretanto apesar de estar apresentando métricas muito boas, elas estarem tão elevadas não são necessariamente significa algo bom, o mais adequado seria estar menor que 98%.

## Comparação Entre Modelos:

O ponto forte do modelo de Naive Bayes seria exatamente o tempo que sua execução demora, mas não apresenta bons resultados nas suas métricas da classe 0 (yes). O modelo de KNN apresenta métricas extremamente altas, demonstrando ter grande potencial para o contexto do aplicativo, mas não alcançando ainda o seu melhor estado. Em paralelo, o modelo que mais se adequou seria o de árvore, demonstrando estar exatamente entre os dois extremos apontados nos outros dois modelos com um bom balanceamento do F1 score de ambas as classes. Apesar disso a árvore seria a que tem o maior tempo de execução.

Fazer a comparação dos resultados das métricas de avaliação e identificar as forças e fraquezas de cada modelo.

Comparação Direta: comparar os resultados das métricas de avaliação entre os modelos. Discuta qual modelo obteve melhor desempenho em cada métrica e quais aspectos de cada modelo contribuem para essas diferenças.

Forças e Fraquezas: identificar as forças e fraquezas de cada modelo com base nas métricas de avaliação e nos objetivos da análise.

# Ajustes Necessários

Realizar mudanças na base de dados ou nos modelos para melhorar os resultados, se necessário, e explicar as modificações realizadas e o impacto delas nos resultados.

## Identificação de Problemas:

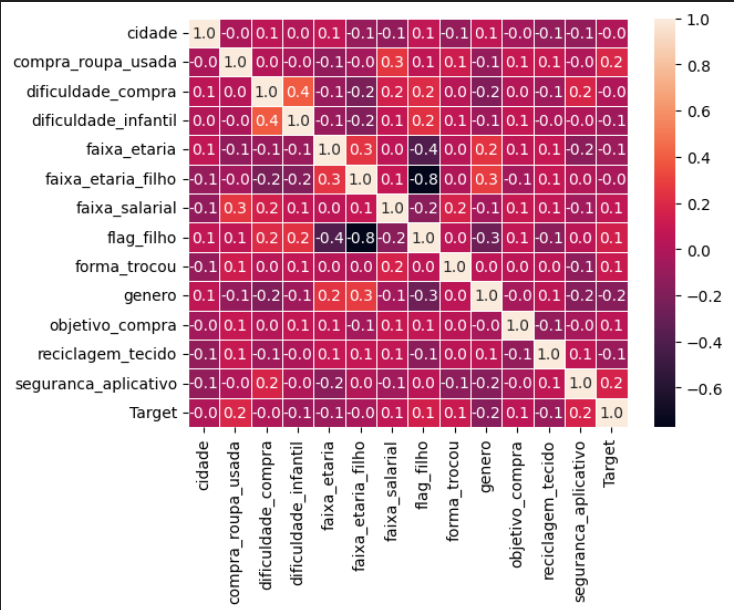
As principais questões a serem solucionadas seriam a dificuldade de classificação da classe 0 (yes) no modelo de Naive Bayes e as métricas extremamente altas do modelo de KNN.

Problemas de Desempenho: identificar problemas específicos encontrados com os modelos, como baixa acurácia, baixa precisão, ou baixa recall.

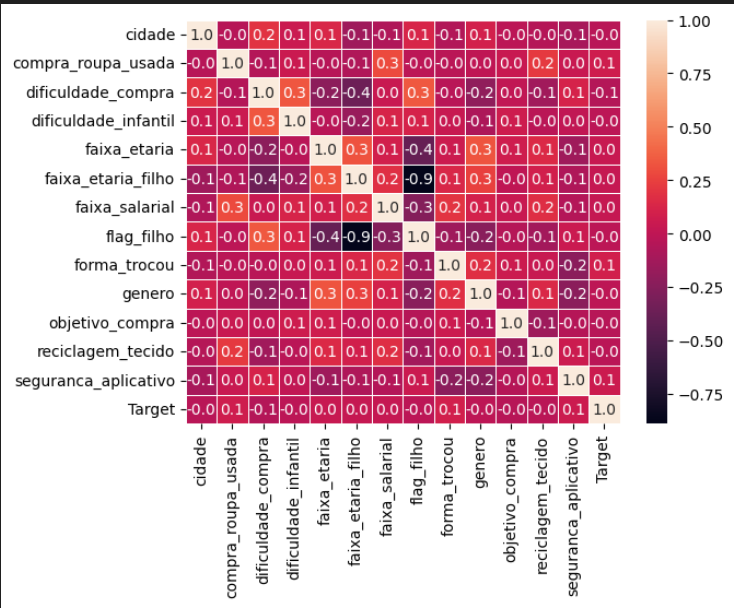
Dados Desbalanceados: se as classes estão desbalanceadas, considerar como isso pode estar afetando o desempenho do modelo.

## Mudanças na Base de Dados:

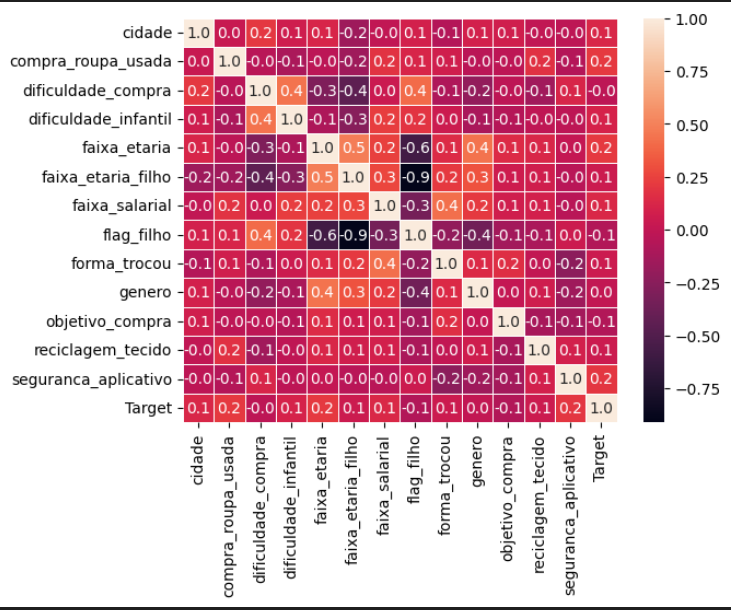
A quantidade e a maneira a qual as informações passadas podem estar impactando o desempenho do modelo do Naive Bayes e do KNN. Para identificar isso foi feito um gráfico de correlação das features de cada uma das bases:



*Base utilizada para a classificação do modelo de Naive Bayes.*



*Base utilizada para a classificação do modelo de Tree.*



*Base utilizada para a classificação do modelo de KNN.*

Foi possivel concluir que as colunas podem ser reduzidas, já que apresenta correlações em sua base. A redução de dimensionalidade auxiliaria também a fazer o modelo generalizar melhor a base e podendo levar a uma execução mais rapida, evitando o overffiting apresentado pelo modelo de KNN e diminuindo o tempo de execução do modelo de tree. Isso será alcançado utilizando o método PCA e o SelectKBest em conjunto.

A dificuldade apresentada pelo Naive Bayes poderia ser solucionada pensando em sua lógica de distribuição normal. Como foi apresentado anteriormente os valores estão fugindo do padrão que o modelo assume, todavia escalonar os dados pode ajudar a manter os valores das variáveis em uma faixa similar, melhorando o ajuste à suposição de normalidade. Desta forma padronizar a base fornecida poderia tornar ele menos influenciável por variáveis em escalas diferentes. E além de poder ajudar também no modelo de KNN que é baseado em distancias e pode ser sensível a escala de dados. Isso será alcançado utilizando o método StandardScaler.

Ajustar o pré-processamento e/ou o balanceamento das classes para melhorar o desempenho.

Pré-processamento:

* Limpeza de Dados: revisar e ajustar a limpeza de dados para remover outliers ou erros que podem estar afetando o desempenho.
* Transformação de Dados: considerar técnicas de normalização ou padronização adicionais.
* Engenharia de Características: adicionar ou modificar características para melhorar a representação dos dados, tais como, criar novas variáveis a partir das existentes ou usar técnicas de seleção de características.

Balanceamento de Classes: usar técnicas de balanceamento como oversampling (ex.: SMOTE) ou undersampling para tratar desbalanceamento de classes e melhorar o desempenho do modelo.

## Ajustes nos Modelos:

A forma a qual vai ser decidido os melhores parâmetros para os métodos, será por meio de uma pipeline que contêm as etapas que passara pela função GridSearchCV, a função também realiza uma validação cruzada com cinco divisões avaliando o F1 score utilizando apenas os dados de treino. Os parâmetros seriam:

Texto

Descrição gerada automaticamente

*Parâmetros que são testados na função GridSearchCV, utilizando uma estrutura de repetição como meio de informar quantas colunas seriam selecionadas pelo SelectKBest e quantas colunas teriam para serem reavaliadas pelo PCA após essa etapa, com o número de colunas variando de 1 a 6 que representa por volta de metade da base.*

Foi testado fazer uma reavaliação dos parâmetros dos modelos e das bases em conjunto com as novas etapas, entretanto o tempo de execução exigido para realizar essa reavaliação tornava inviável realizá-la. Logo para essas melhoras foram mantidos os parâmetros e a base do modelo que teve o maior desempenho abaixo de 98%. Além disso também foi testado ampliar o intervalo de números para o SelectKBest, entretanto não gerava o resultado esperado de combater a presença do overffiting no modelo de KNN, como a redução para no máximo 6 colunas não causaram grandes impactos nos outros modelos foi estabelecido desta forma.

Verificar a necessidade de ajustes nos hiperparâmetros e/ou utilização de validação cruzada, caso ainda não tenham feito:

Hiperparâmetros: realize ajuste de hiperparâmetros para otimizar o desempenho do modelo.

Validação Cruzada: usar validação cruzada para avaliar o desempenho dos modelos de forma mais robusta e evitar overfitting.

# Impacto das Modificações

Discutir novamente os resultados obtidos para cada modelo com base nas métricas de avaliação.

Gráfico, Gráfico de barras

Descrição gerada automaticamente

*Média dos F1 scores que cada modelo alcançou com a nova pipeline durante a validação cruzada.*

Gráfico, Gráfico de barras, Gráfico de mapa de árvore

Descrição gerada automaticamente

*Média dos tempos de execução de cada modelo extraída durante uma bateria de 6 vezes de teste.*

## Modelo 1: Naive Bayes (GaussianNB)

Gráfico, Gráfico de mapa de árvore

Descrição gerada automaticamente

*As colunas selecionadas após o estágio do PCA e suas principalidades.*

Calendário

Descrição gerada automaticamente

*Resultados das métricas apresentadas durante o teste realizado com a pipeline, utilizando o modelo de Naive Bayes que apresentou o melhor desempenho (abaixo de 98%) com a base que interagiu na etapa anterior.*

Gráfico, Gráfico de mapa de árvore

Descrição gerada automaticamente

*Matriz de confusão gerada a partir da predição do modelo de aprendizado.*

É possivel notar que o modelo conseguiu entender melhor a base de dados fornecida a ele após a sua padronização, desta forma a classe 0 (yes) teve melhores métricas de recall e precisão do que anteriormente, entretanto ainda não alcançou um resultado satisfatório em ambas as métricas o que resulta em um resultado baixo também no F1 score.

Discussão dos Resultados: explicar como as mudanças na base de dados e os ajustes nos modelos ajudaram a melhorar as métricas de avaliação.

## Modelo 2: Tree (DecisionTreeClassifier)

Gráfico

Descrição gerada automaticamente

*As colunas selecionadas após o estágio do PCA e suas principalidades.*

Calendário

Descrição gerada automaticamente

*Resultados das métricas apresentadas durante o teste realizado com a pipeline, usando o modelo de tree que apresentou o melhor desempenho (abaixo de 98%) com a base que interagiu na etapa anterior.*

Gráfico, Gráfico de mapa de árvore

Descrição gerada automaticamente

*Matriz de confusão gerada a partir da predição do modelo de aprendizado.*

Após a mudança, é possivel observar uma piora do desempenho do modelo de árvore, apesar de não ser uma grande piora. Isso provavelmente se deve a limitação de colunas até 6, tanto que é possivel observar na seleção de colunas que foram escolhidas a quantidade máxima.

Discussão dos Resultados: explicar como as mudanças na base de dados e os ajustes nos modelos ajudaram a melhorar as métricas de avaliação.

## Modelo 3: KNN (KNeighborsClassifier)

Interface gráfica do usuário, Gráfico

Descrição gerada automaticamente

*As colunas selecionadas após o estágio do PCA e suas principalidades.*

Uma imagem contendo Calendário

Descrição gerada automaticamente

*Resultados das métricas apresentadas durante o teste realizado com a pipeline, o modelo de KNN seria o que apresentou o melhor desempenho (abaixo de 98%) com a base que interagiu na etapa anterior.*

Gráfico

Descrição gerada automaticamente

*Matriz de confusão gerada a partir da predição do modelo de aprendizado.*

O modelo de KNN não demonstrou novamente overffiting, obtendo 94% de F1 score, apesar também selecionar a maior quantidade possivel apresenta não ter dificuldades com a redução de dimensionalidade e conseguir ter um ótimo desempenho apenas com as novas seis colunas escolhidas.

Discussão dos Resultados: explicar como as mudanças na base de dados e os ajustes nos modelos ajudaram a melhorar as métricas de avaliação.

## Comparação Entre Modelos:

O modelo de Naive Bayes aumentou suas métricas na classe 0 (yes) em relação ao que foi apresentado anteriormente, mas não suficientemente para alcançar resultados bons. Enquanto o modelo de arvore diminuiu suas métricas, além de aumentar o seu tempo de execução piorando nos dois quesitos em relação a anteriormente. Por fim foi solucionado o overffiting apresentado no modelo de KNN e houve uma grande melhora em seu tempo de execução, desta forma, em comparação aos outros modelos testados nessa pipeline, foi o modelo que obteve os melhores resultados.

Em uma comparação ampla, os dois melhores modelos testados que obtiveram os melhores resultados foram o modelo de KNN com a pipeline e o modelo de arvore antes da pipeline. Ambos tiveram F1 scores bons, com o modelo de arvore ligeiramente maior, o mesmo cenário é encontrado quando discutido em relação ao tempo de execução. Para o auxílio desta decisão foi realizado um teste A/B levando em consideração as métricas obtidas durante a validação cruzada realizada e o tempo de execução coletado durante a bateria de testes.

Apesar das melhorias as métricas se apresentaram menores do que as presentes na etapa anterior,

Fazer a comparação dos resultados das métricas de avaliação, identificar as forças e fraquezas de cada modelo.

Escolher o modelo mais adequado para seus dados e explicar, porque é o mais adequado.

# Conclusão

Resumir as principais descobertas da análise e a eficácia dos modelos aplicados.

Discutir o potencial impacto das previsões no seu projeto e as próximas etapas recomendadas para aprimorar a análise.