Análise de Dados para *Olimpo*

Apolo

Sofia Elisabete Rosa Domingues – n°19

2°G Tech

Sumário

[Introdução 3](#_Toc174100038)

[Objetivo do aplicativo 3](#_Toc174100039)

[Objetivo do trabalho de Análise de Dados 3](#_Toc174100040)

[Levantamento dos dados 3](#_Toc174100041)

[Busca dos dados 3](#_Toc174100042)

[Justificativa de uso 3](#_Toc174100043)

[Descrição da base de dados encontrada 3](#_Toc174100044)

[Limpeza dos dados 4](#_Toc174100045)

# Introdução

O Olimpo surgiu da iniciativa dos integrantes do primeiro ano do Apolo de resolver o problema da escassez de cooperação e integração em comunidades. Nosso público-alvo são maiores de idade. Essa escolha foi baseada no fato de que essas pessoas apresentam necessidades da vida adulta, um maior senso comunitário e um alto uso de redes sociais de acordo com o IBGE em 2022.

## Objetivo do aplicativo

O Olimpo almeja aumentar a organização comunitária dentro de comunidades, incentivando a coesão social por meio de chats e publicações dentro do próprio aplicativo. Ademais, o Olimpo visa incentivar a economia ambiental e local dentro das comunidades que poderão anunciar seus itens usados. Dessa forma, o Olimpo impactará de maneira positiva a vida de seus usuários, seja esse impacto ambiental, governamental ou social.

Recomenda-se citar aqui que os integrantes do 1° ano já fizeram a especificação do aplicativo e que os detalhes sobre essa parte podem ser encontrados no material desenvolvido por eles para documentação do trabalho.

# Objetivo do trabalho de Análise de Dados

O objetivo do trabalho de análise de dados para o Olimpo é transformar grandes volumes de informações brutas em insights acionáveis que podem orientar decisões estratégicas para determinar se uma pessoa é um cliente em potencial para o Olimpo ou não. Através da coleta, organização e interpretação de dados, a análise permite identificar padrões e tendências de perfis, proporcionando uma compreensão mais profunda dos processos e comportamentos. Essa abordagem analítica facilita a resolução de problemas, otimização de processos e a formulação de estratégias baseadas nessas análises, ajudando nossa equipe, o Apolo, a tomar decisões informadas e a alcançar seus objetivos com maior eficácia.

# Levantamento dos dados

## Busca dos dados

Variás pesquisas foram realizadas com nosso publico-alvo, moradores de comunidades, para entender suas necessidades e prioridades dentro de seu convivio social. A equipe do Apolo escolheu criar um formulário próprios com perguntas especificas com maior relevância para o grupo e treinamento de nossa IA, que visa prever possiveis clientes. O formulário recebeu aproximadamente 100 respostas.

# Descrição da base de dados encontrada

1. **Id** (*int*): Identificador único de cada resposta.
2. **Hora de início** (*datetime*): Data e hora em que a resposta começou.
3. **Hora de conclusão** (*datetime*): Data e hora em que a resposta foi concluída.
4. **Email** (*string*): Email do participante (anônimo).
5. **Nome** (*float*): Nome do participante (valores ausentes são armazenados como NaN).
6. **Termo de Anonimato** (*string*): Indica se o termo de anonimato foi aceito.
7. **Em que cidade você vive atualmente?** (*string*): Cidade onde o participante vive.
8. **Qual é seu tipo da sua moradia?** (*string*): Tipo de moradia (ex., apartamento ou casa).
9. **Com quantas pessoas você divide sua casa/apartamento?** (*string*): Número de pessoas com quem divide a moradia (contém textos como "Nenhuma").
10. **Nível de satisfação com a comunicação** (*int*): Escala numérica de satisfação com a comunicação na comunidade.
11. **Frequência de contato com vizinhos** (*string*): Frequência de contato com vizinhos (ex., "Raramente").
12. **Acredita na importância da comunicação comunitária?** (*string*): Opinião sobre a importância da comunicação na comunidade.
13. **Uso de plataforma de comunicação** (*string*): Indica se o participante usa alguma plataforma comunitária.
14. **Nível de satisfação com a plataforma** (*float*): Satisfação com a plataforma comunitária (ausências registradas como NaN).
15. **Faixa etária** (*string*): Faixa etária do participante (ex., "Maior ou igual a 43 anos").
16. **Qualidade de vida** (*string*): Avaliação subjetiva da qualidade de vida (ex., "Boa" ou "Ótima").
17. **Situação no mercado de trabalho** (*string*): Situação de trabalho atual (ex., "Empregado").
18. **Renda familiar mensal** (*string*): Faixa de renda mensal (ex., "Classe C").
19. **Animais de estimação** (*string*): Indica se o participante possui pets.
20. **Implementação do aplicativo** (*string*): Intenção de implementar o aplicativo sugerido.

A análise de dados e código de ML foi feita por um código Python próprio criado pelos integrantes do Apolo. O repositório GitHub em que o modelo está armazenado pode ser encontrado através desse link: https://github.com/Olimpo-Org/olimpo-ml

# Limpeza dos dados

A base de dados coletada possui um total de 20 colunas, sendo dessas:

* 12 pertencem a X (outras)
* 1 pertence a Y (implementa\_app)
* 6 não possuem relevânca para X ('ID', 'Hora de início', 'Hora de conclusão', 'Email', 'Nome', 'Termo de Anonimato', 'Nivel de Satisfação com a Plataforma Utilizada')

Após a análise macro da base, começamos a limpar e extrair somente as informações necessárias para o nosso modelo, seguindo o seguinte passo a passo:

1. Primeiro, renomeamos todas as colunas para que fosse mais fácil na hora de utiliza-las dentro do modelo. Texto

   Descrição gerada automaticamente
2. Após renomear os campos, notamos a necessidade de limpar completamente a coluna “Cidade”, pois ela não foi coletada utilizando um padrão de escrita, portanto, existiam diversas respostas iguais com formatações diferentes. Texto

   Descrição gerada automaticamente
3. Depois disso, removi a parte descritiva da coluna “Frequência de contato com vizinhos**”** e extraí somente a parte relevante com as categorias(‘Ocasionalmente’, ‘Raramente’, etc). Texto

   Descrição gerada automaticamente
4. Por fim, removi as colunas indesejadas, como descritas no início dessa seção. Interface gráfica do usuário, Texto, Site

   Descrição gerada automaticamente

Condicionamento para alimentar o modelo de ML

Os condicionamentos iniciais do modelo de ML foram:

* + - Normalização de dados (via zscore)
    - Oversampling, pois a base de dados é muito pequena, e não faria sentido a aplicação do undersampling

É importante notar que foi testado o PCA, porém foi descartado devida a pequena quantidade de features (colunas) apresentadas na base.

AQUI COMEÇA A PARTE DE ANÁLISE DE DADOS (MÓDOLO)

# Definição dos objetivos e das classes

# O objetivo do modelo é prever se uma pessoa é ou não potencial usuário do aplicativo Olimpo baseado nas informações dadas pelo usuário.

# Definição dos modelos mais adequados para analisar os dados

## Natureza dos dados:

Apesar dos dados fornecidos serem, tanto quantitativos, como qualitativos, após o tratamento de toda a base, todos os dados são transformados em dados quantitativos discretos, ou seja, são números que representam valores e quantidades, que geralmente são usados para analisar padrões, médias e correlações

## Complexidade dos Modelos:

**Modelos simples:** são algoritmos de aprendizado de máquina que têm uma estrutura básica e são rápidos e fáceis de interpretar. Exemplos incluem **regressão linear** e **Naive Bayes**. Eles são ideais para dados pequenos ou situações em que interpretabilidade e rapidez são mais importantes do que a precisão.

**Modelos complexos**: têm estruturas mais sofisticadas, como as **redes neurais profundas** ou **florestas aleatórias**. Esses modelos conseguem capturar relações complexas nos dados, mas exigem mais poder computacional e podem ser difíceis de interpretar.

## Diferença entre ambos:

* **Desempenho**: Modelos complexos tendem a ter maior precisão, especialmente em dados grandes e não lineares. Modelos simples funcionam bem em dados estruturados ou problemas onde as relações são mais diretas.
* **Interpretação**: Modelos simples são mais fáceis de entender e interpretar, enquanto modelos complexos são como uma "caixa preta", tornando difícil saber como as decisões são feitas.
* **Tempo e Recursos**: Modelos simples são rápidos e exigem menos recursos. Modelos complexos demandam mais tempo e poder computacional.

Essencialmente, escolher entre um modelo simples e um complexo depende do equilíbrio entre precisão, interpretabilidade e recursos disponíveis.

**Objetivos de Classificação:**

* Classificação Binária: Todos os modelos somente tentam prever 2 classes, “Sim” e “Não”

Parte superior do formulário

Parte inferior do formulário

# 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: Gaussian
* Esse modelo pertence à família de classificadores Naive Bayes, que se baseiam no **Teorema de Bayes** e assumem independência entre as variáveis. O GaussianNB é específico para dados contínuos que seguem uma distribuição **normal (ou Gaussiana)**, o que significa que ele calcula a probabilidade de cada classe com base na média e variância dos atributos. É um modelo rápido e eficiente, especialmente em conjuntos de dados grandes e onde a suposição de independência entre variáveis faz sentido.
* Modelo 2: Tree Decision
* Esse modelo constrói uma árvore onde cada nó representa uma decisão baseada em uma **condição de um atributo**. O objetivo é dividir o conjunto de dados em partes menores e mais homogêneas em termos de classe-alvo. Ele funciona por meio de divisões sucessivas (em atributos) que maximizam a pureza das folhas da árvore, usando métricas como **gini** ou **entropia**. O Decision Tree é intuitivo e fácil de interpretar, mas pode ser suscetível a **overfitting**, especialmente em dados pequenos.
* Modelo 3: K-Neighbours
* O KNN é um **modelo de aprendizado supervisionado baseado em instâncias**, ou seja, ele faz previsões comparando novos dados com os dados de treinamento. Basicamente, para prever a classe de uma nova observação, o modelo procura os “K” exemplos mais próximos (vizinhos) e atribui a classe mais comum entre esses vizinhos. A distância entre as instâncias é geralmente medida por **métricas como a distância Euclidiana**. O KNN é simples e não exige treinamento pesado, mas pode ser mais lento com muitos dados, pois calcula distâncias em tempo real.

# Aplicação dos modelos selecionados

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

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

Comparação Entre Modelos:

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:

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:

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:

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.

Modelo 1: [Nome do Modelo]

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: [Nome do Modelo]

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

Modelo n [Nome do Modelo]

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:

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.