Identificação do ICP (*Ideal Customer Profile*) na área de Tecnologia em Saúde

Felipe Fernandes
Instituto de Ciência e Tecnologia
Universidade Federal de São Paulo
São José dos Campos, Brasil
fernandes.felipe@unifesp.br

***Abstract*—O presente artigo apresenta uma análise abrangente sobre o ICP e sua área, o algoritmo de agrupamento K-Means, uma técnica popular de aprendizagem de máquina não supervisionada que se baseia na ideia de clusterização, minimizando a distância entre os elementos, e também sobre a tarefa de classificação pós agrupamento, o algoritmo utilizado para este fim (KNN) e suas métricas. São descritos os conceitos que envolvem estas técnicas, as etapas que envolvem o desenvolvimento do modelo e trabalhos relacionados.**

***Index Terms*—ICP, tecnologia, saúde, agrupamento, classificação**

I. Introdução

Nas últimas décadas, os avanços tecnológicos permitiram muitas facilidades, tanto do ponto de vista de *software* quanto de *hardware*. A capacidade de processamento de dados, por exemplo, têm auxiliado diversas pessoas de várias áreas, uma vez que cálculos impossíveis de serem feitos de forma manual são realizados em questão de segundos por dispositivos computacionais.

Isso fez com que tarefas de reconhecimento de padrões e predição de resultados se tornassem viáveis graças ao alto desempenho computacional que esses dispositivos possuem [5].

Comumente falada e cada vez mais em evidência, a Inteligência Artificial e sua área de Aprendizagem de Máquina compreendem diversos conceitos e técnicas para a constante evolução das suas aplicações no mundo real. A sua principal ideia se concentra no desenvolvimento de algoritmos e modelos capazes de aprender padrões e tomar decisões a partir de dados.

É possível afirmar que a base de dados é a matéria prima para qualquer atividade dentro desse contexto, dado que, através disso, torna-se possível extrair padrões, singularidades e outras informações a partir de um conjunto de características.

O contexto, nesse caso, é a Tecnologia em Saúde, que se utiliza das ideias e abordagens da área de IA, sanando dúvidas e fornecendo informações valiosas para os objetivos propostos.

Baseado no exposto acima, torna-se possível discorrer sobre a aplicação em problemas concretos, detalhando pontos importantes e evidenciando como esta técnica pode auxiliar o mundo real.

II. Fundamentação Teórica

A. Conceitos Fundamentais

O conceito de Perfil de Cliente Ideal (*Ideal Customer Profile*) tem suas raízes nas práticas de segmentação de mercado e marketing direcionado, que começaram a se desenvolver no início do século XX. No entanto, a formalização e popularização do ICP como uma ferramenta estratégica específica ganharam impulso nas últimas décadas, particularmente com o avanço das tecnologias de informação e a evolução das estratégias de marketing e vendas orientadas por dados.

O perfil do cliente é, basicamente, o conjunto de características e comportamentos que definem o comprador ideal de um produto ou serviço. É uma definição geral do perfil das pessoas que consomem de determinada empresa/negócio, construída com base no seu histórico de compradores e nos objetivos da marca.

Quando falamos de perfil do cliente, há sempre uma confusão entre esse termo e o ICP. O que pode-se dizer é que, apesar de muito parecidos, eles não são a mesma coisa. Isso porque mesmo semelhantes, esses conceitos nem sempre são convergentes.

O perfil do cliente, como dito, é uma mistura de características e comportamentos de tipos diferentes de clientes. O ICP, por sua vez, está mais focado nos clientes que têm mais potencial de comprar seu produto ou serviço. Essa é uma métrica importante para as estratégias comerciais de uma empresa e refere-se ao grupo de pessoas que são mais passíveis de conversão/sucesso.

O setor de Tecnologia em Saúde abrange uma vasta gama de produtos e serviços, desde sistemas de prontuários eletrônicos e telemedicina até dispositivos médicos avançados e soluções de Inteligência Artificial para diagnóstico e tratamento. A natureza complexa e especializada desse mercado exige uma compreensão profunda das necessidades, desafios e oportunidades que os clientes enfrentam.

Portanto, identificar o ICP envolve considerar diversos fatores, incluindo o porte da empresa, o segmento de mercado, o perfil demográfico dos decisores, as necessidades tecnológicas específicas, e as tendências emergentes na área de saúde.

A identificação do Perfil de Cliente Ideal é uma etapa crucial para empresas que desejam otimizar suas estratégias de

marketing e vendas, especialmente em setores dinâmicos e em constante evolução, como a Tecnologia em Saúde. Relacionar isso às tarefas de Aprendizado de Máquina faz com que seja possível definir metodologias cada vez mais sofisticadas e refinar ainda mais o processo de prospecção de clientes.

Uma vez entendidos os conceitos de ICP e Tecnologia em Saúde, pode-se partir para a etapa de entendimento das ideias por trás dos modelos de Aprendizado de Máquina.

Um dos principais conceitos a serem entendidos a partir disso são os modelos de agrupamento.

Análise de agrupamento, ou *clustering*, é o nome dado para o grupo de técnicas computacionais cujo propósito consiste em separar objetos em grupos, baseando-se nas características que estes objetos possuem. A ideia básica consiste em colocar em um mesmo grupo objetos que sejam similares de acordo com algum critério pré-determinado.

O critério baseia-se normalmente em uma função de dissimilaridade, que recebe dois objetos e retorna a distância entre eles. Os grupos determinados por uma métrica de qualidade devem apresentar alta homogeneidade interna e alta separação, isto é, os elementos de um certo conjunto são mutuamente similares e, preferencialmente, muito diferentes dos elementos de outros conjuntos.

Os objetos são denominados exemplos, tuplas e/ou registros. Cada objeto representa uma entrada de dados que pode ser constituída por um vetor de atributos que são campos numéricos ou categóricos (tipo de campo que pode assumir um entre um conjunto de valores pré definidos) [3].

A análise de agrupamento é uma ferramenta útil para a análise de dados em muitas situações diferentes. Esta pode ser usada para reduzir a dimensão de um conjunto de dados, reduzindo uma ampla gama de objetos à informação do centro do seu conjunto.

Tendo em vista que o *clustering* é uma técnica de aprendizado não supervisionado, pode servir também para extrair características escondidas dos dados e desenvolver as hipóteses a respeito de sua natureza.

Dos diversos algoritmos utilizados na área de agrupamento, pode-se mencionar o K-Means, uma heurística de agrupamento não hierárquico que busca minimizar a distância dos elementos a um conjunto de *k* centros dado de forma iterativa por (1).

$$x = x1, x2, ..., xk \tag{1}$$

A distância entre um ponto e um conjunto de *clusters* é definida como sendo a distância do ponto ao centro mais próximo dele.

Este algoritmo é extremamente veloz, geralmente convergindo em poucas iterações para uma configuração estável, na qual nenhum elemento está designado para um cluster cujo centro não lhe seja o mais próximo.

Após essa tarefa de agrupamento, a tarefa seguinte a ser realizada para identificar novos clientes ideais é a classificação, que consiste no processo de identificar um modelo ou função que descreve diferentes classes de dados, rotulando novas instâncias conforme sua base de informações, baseando-se no valor dos atributos dos mesmos [4].

Dentre os algoritmos de classificação, o *K-Nearest Neighbors* (KNN) pode ser destacado.

Esse é um dos algoritmos mais simples e amplamente utilizados na área de Aprendizado de Máquina, se enquadrando na categoria de algoritmos supervisionados usados não só para tarefas de classificação mas também para regressão. Tem como premissa a tomada de decisões com base na semelhança entre os dados de entrada e exemplos previamente conhecidos, sem assumir um modelo predefinido para os mesmos.

A ideia principal do algoritmo é que exemplos com atributos semelhantes tendem a estar próximos uns dos outros no espaço de características. Para prever a classe ou o valor de um novo dado, o KNN analisa as *k* observações mais próximas desse novo ponto, com base em uma métrica de distância, geralmente a distância Euclidiana, mostrada em (2).

$$d(p, q) = \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2} \tag{2}$$

A classe mais comum entre essas *k* observações será a classe atribuída ao novo ponto no caso da classificação, ou a média dos valores no caso da regressão.

Assim como em outros algoritmos de classificação, o KNN é um procedimento que envolve duas fases: treinamento do modelo e previsão de dados de teste.

Na fase de treinamento, envolve apenas encontrar um *k* adequado para um determinado conjunto de dados de treinamento.

A escolha de *k* é crítica para o desempenho do KNN. Valores pequenos de *k* resultam em modelos mais complexos e podem levar ao *overfitting* (sensibilidade excessiva a dados de treinamento), enquanto valores grandes podem levar ao *underfitting* (modelo simples demais, incapaz de capturar padrões). Um método comum para escolher o valor ótimo de *k* é a validação cruzada, onde diferentes valores de *k* são testados e o que resulta no melhor desempenho é escolhido.

Na fase de previsão, a primeira etapa é uma busca por *k* pontos de dados no conjunto de dados de treinamento que são mais relevantes para uma consulta (dados de teste/amostra). Sem outras informações, os *k* pontos de dados mais relevantes são considerados os *k* vizinhos mais próximos dos dados de teste dentro do conjunto de dados de treinamento. Depois disso, uma previsão é feita com base na classe de dados de teste que ocorre com mais frequência entre os *k* vizinhos. Isso é conhecido como regra da maioria.

Do ponto de vista geral do algoritmo, isso indica que há quatro pontos principais para sua execução: computação de *k*, seleção do vizinho mais próximo, pesquisa do vizinho mais próximo e regra de classificação.

O uso do KNN faz-se presente devido ao fato de ser intuitivo e não requerer nenhum conhecimento aprofundado, além de não fazer suposições sobre a distribuição dos dados, sendo assim não-paramétrico, tornando-o útil para uma variedade de problemas.

Outro conceito de grande relevância diz respeito às medidas de avaliação de um modelo.

As medidas de avaliação de um modelo de Aprendizado de Máquina são métricas utilizadas para avaliar e quantificar o desempenho do modelo em relação aos dados de teste ou validação/treinamento. Essas medidas fornecem informações sobre a qualidade das previsões ou classificações feitas e permitem comparar diferentes modelos ou ajustes de parâmetros.

Em um problema de classificação, há duas soluções possíveis: erro ou acerto. Alguns dos conceitos mais importantes para a classificação binária gira em torno das classes que os dados preditos podem receber a partir de verdadeiro ou falso: TP, TN, FP e FN.

- **Verdadeiro Positivo (TP):** significa que a classe prevista e observada originalmente fazem parte da classe positiva;
- **Verdadeiro Negativo (TN):** significa que a classe prevista e observada originalmente fazem parte da classe negativa;
- **Falso Positivo (FP):** significa que a classe predita retornou positivo mas a original observada era negativa;
- **Falso Negativo (FN):** representa que o valor predito resultou na classe negativa mas o original observado era da classe positivo.

Uma maneira mais simples de representar dados de um método de classificação é através da Matriz de Confusão. Esta indica a quantidade de ocorrências que o modelo teve para cada uma das quatro categorias mencionadas anteriormente.

		Valor Predito	
		Sim	**Não**
Real	**Sim**	Verdadeiro Positivo (TP)	Falso Negativo (FN)
	Não	Falso Positivo (FP)	Verdadeiro Negativo (TN)

Fig. 1. Representação de uma Matriz de Confusão

Através disso, obtém-se os valores de acertos ou erros das predições com um maior embasamento.

Outra métrica simples porém importante é a acurácia, que tem como intuito avaliar o percentual de acertos do modelo. Pode ser obtida facilmente através da razão entre a quantidade de acertos e o total de entrada:

$$acuracia = \frac{acertos}{entradas} \tag{3}$$

Utilizando os conceitos anteriormente apresentados, pode-se obtê-la também pela seguinte fórmula:

$$acuracia = \frac{TP + TN}{TP + FN + TN + FP} \tag{4}$$

Dentre as demais métricas, pode-se mencionar outras duas essenciais para este estudo: precisão e *F-score*.

Diferenciando-se da acurácia, a precisão avalia a quantidade de verdadeiros positivos sobre a soma de todos os outros valores positivos, isto é:

$$precisao = \frac{TP}{TP + FP} \tag{5}$$

Por outro lado, a *F-measure*, *F-score* ou F1 é uma média harmônica calculada com base em outras duas métricas: a própria precisão e também a revocação (*recall*), que indica das amostras positivas, quantas o modelo conseguiu classificar corretamente. A *F-score* pode ser obtida a partir da seguinte equação:

$$F1 = \frac{2 * precisao * revocacao}{precisao + revocacao} \tag{6}$$

Além dessas medidas, dependendo do problema e do contexto, podem ser utilizadas outras métricas específicas, como Área Sob a Curva ROC (AUC-ROC), *Log Loss*, *Mean Absolute Error* (MAE), *Root Mean Squared Error* (RMSE), dentre outras.

Por fim, é válido ressaltar que a escolha das medidas de avaliação adequadas depende do tipo de problema, classes envolvidas, desbalanceamento das mesmas, consequências dos erros de previsão e metas específicas do projeto.

B. Trabalhos Relacionados

O artigo "*Modeling Ideal Customer Profile for Maximum Return on Investment*" de Bhushan Ekbote explorou as melhores práticas para desenvolver o Perfil de Cliente Ideal (ICP) e apresentou modelos matemáticos para otimizar a segmentação de mercado e cobertura de vendas. Através de análises qualitativas e quantitativas, o ICP identificou os clientes mais valiosos para uma empresa.

O trabalho discutiu a diferença entre ICP e persona do comprador, coleta de dados de clientes e utilização de ferramentas online. Modelos de regressão e análise de correlação foram usados para aplicar o ICP ao mercado total endereçável (TAM), dessa forma concluindo que este método de modelagem é único porque além de analisar o banco de dados de clientes para fornecer percepções significativas, aplica estas aos dados totais do mercado, economizando dinheiro, possibilitando uma reconfiguração mais barata e impulsionando as empresas tanto em eficiência quanto em eficácia.

O artigo "*Integration K-Means Clustering Method and Elbow Method for Identification of The Best Customer Profile Cluster*", publicado na *IOP Conference Series: Materials Science and Engineering*, explorou a aplicação combinada dos métodos de *clustering* K-Means e heurística *Elbow* para identificar o melhor perfil de *cluster* de clientes.

A pesquisa, conduzida por Syakur et al. (2018), visou otimizar a segmentação de perfis de clientes, utilizando-se dos métodos mencionados acima para agrupar os dados e determinar o número ideal de *clusters*. O método *Elbow* ajudou a identificar o ponto onde a soma dos erros quadrados dentro dos *clusters* não diminuiu significativamente com o aumento deles, indicando o número ótimo de agrupamentos.

Os resultados demonstraram que ao aplicar essa abordagem, foi possível obter *clusters* mais coerentes e úteis para análise de perfis de clientes, facilitando a identificação de segmentos de mercado relevantes. A técnica mostrou-se eficaz para melhorar a precisão da segmentação e a compreensão das características dos clientes.

É válido ressaltar, porém, que o artigo não discutiu a utilização de um algoritmo de classificação durante o desenvolvimento do modelo/estudo. Isso mostra que é possível aprimorar o processo de criação do modelo a depender do objetivo.

Em suma, ambos os artigos corroboram com a ideia de que encontrar o ICP utilizando-se dos conceitos da Inteligência Artificial é válido e traz resultados satisfatórios tanto do ponto de vista do modelo, quanto da facilitação na análise dos dados e investigação do comportamento dos *clusters*.

C. Objetivo

A partir da ideia de agrupamento e classificação e da perspectiva de *Design Science* [2], um paradigma de pesquisa com foco no desenvolvimento e validação de conhecimento prescritivo em ciência da informação, é possível abordar a seguinte tarefa: *criação de um modelo de aprendizado que nos aproxime da identificação do Perfil de Cliente Ideal na área de Tecnologia em Saúde.*

III. Metodologia Experimental

Revisando a literatura e baseando-se no processo de descoberta [1] e nos passos que permeiam a ideia geral do Aprendizado de Máquina, foram definidas sete fases necessárias relacionadas ao processamento dos dados:

1) Limpeza dos Dados
2) Integração dos Dados
3) Seleção dos Dados
4) Transformação dos Dados
5) Mineração dos Dados
6) Avaliação dos Dados
7) Visualização dos Dados

Com essas informações, pode-se partir para o processo de validação do projeto, usando a linguagem *Python* e seus *frameworks*, tais como *Pandas*, *Numpy*, *Seaborn*, *MatPlotLib*, *SciKit-Learn* e afins.

Em relação à base de dados, serão utilizados dados verdadeiros de uma empresa da área, para que o modelo seja desenvolvido em cima de informações do mundo real.

O contexto é dado a seguir: a empresa possui soluções de saúde que conseguem reduzir drasticamente o custo de seus clientes com convênio. Através de diversas estratégias e abordagens realizadas, a estratégia adotada permite que o custo com planos de saúde e relacionados não aumente gradativamente.

Portanto, determinar um ICP (ou algo próximo disso) é de extrema importância para os envolvidos, já que isso faz com que a empresa consiga oferecer seus serviços diretamente ao público alvo que possui essa dor.

A base de dados original possui 560 amostras (clientes) e 56 atributos, sendo a maioria do tipo não numérica, embora hajam também valores numéricos. Por questões de privacidade, os nomes dos clientes da base foram substituídos por nomes genéricos.

Baseando-se nas informações discutidas até o momento, tornou-se possível a criação de um diagrama de blocos que descrevesse as etapas do *pipeline* experimental para a execução do trabalho:

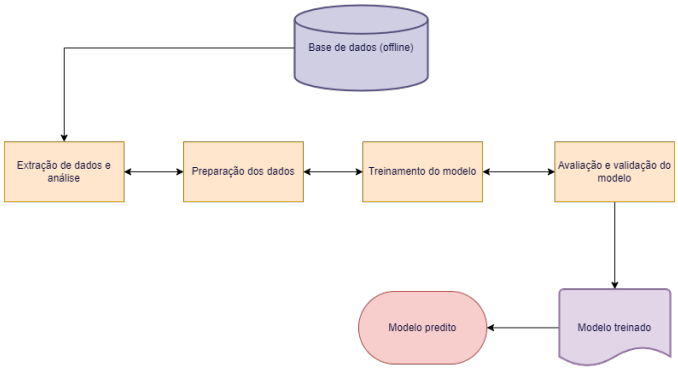

Fig. 2. Etapas do desenvolvimento do experimento

As etapas estão separadas da seguinte forma:

- **Base de dados (*offline*):** necessário reunir os dados relevantes para o problema em questão. Isso envolve a busca por conjuntos de dados existentes, coleta de dados de sensores ou dispositivos, extração de informações de bancos de dados, entre outras fontes.
- **Extração de dados e análise:** necessário realizar uma análise exploratória para compreender melhor o conjunto de dados. Isso envolve a identificação e o entendimento das diferentes variáveis presentes, a análise de sua distribuição e características, a detecção de possíveis problemas, como valores ausentes ou discrepantes, e a identificação de correlações entre as variáveis.
- **Preparação dos dados:** os dados precisam ser preparados para o treinamento do modelo. Isso pode incluir a remoção de valores ausentes, a normalização ou padronização das variáveis, a codificação de variáveis categóricas, a seleção de características relevantes e a divisão dos dados em conjuntos de treinamento, validação e teste.
- **Treinamento do modelo:** fase em que o modelo é construído e treinado com base nos dados preparados. Isso envolve a escolha do algoritmo adequado para o problema em questão, a definição dos hiperparâmetros do modelo e a execução do processo de treinamento. Durante o treinamento, o modelo aprenderá a mapear os padrões nos dados e ajustar seus parâmetros internos para realizar previsões ou tomar decisões.
- **Avaliação e validação do modelo:** necessário avaliar o desempenho do modelo para verificar sua capacidade de generalização em dados não vistos anteriormente. Isso pode ser feito usando técnicas como validação cruzada ou outras. Além disso, métricas adequadas são utilizadas para medir o desempenho do modelo.
- **Modelo treinado:** após a etapa de treinamento e validação, o modelo é considerado treinado. Isso significa que ele foi ajustado aos dados disponíveis e está pronto para fazer previsões ou tomar decisões com base em novos dados de entrada. O modelo contém os parâmetros

aprendidos durante o treinamento e se encontra em um estado em que pode ser usado para inferências.

- **Modelo predito:** quando o modelo treinado é alimentado com novos dados de entrada, ele usa o conhecimento adquirido durante o treinamento para fazer previsões ou classificações. Essa etapa envolve a passagem dos dados de entrada pelo modelo e a obtenção de uma resposta ou resultado predito. Dependendo do tipo de problema, o modelo pode gerar uma previsão numérica, uma classificação em categorias ou até mesmo uma probabilidade associada a cada classe. O resultado obtido é a resposta do modelo às informações fornecidas.

Definidas as etapas relacionadas aos dados e ao experimento, pode-se especificar cada uma delas atribuídas ao contexto do objetivo proposto:

1) **Preparação dos Dados**
   a) Leitura dos dados
   b) Limpeza dos dados
   c) Transformação dos dados
2) **Análise Exploratória dos Dados**
   a) Visualização dos dados
   b) Identificação de padrões
3) **Seleção de Recursos (*features*)**
   a) Identificação de recursos relevantes
4) **Treinamento do Modelo**
   a) Aplicação do algoritmo (K-Means)
5) **Avaliação do Modelo**
   a) Validação dos agrupamentos
   b) Identificação dos *clusters* ideais
6) **Otimização e Ajuste do Modelo**
   a) Ajuste de hiperparâmetros

É esperado que após essas etapas, já seja possível determinar os *clusters* ideais (isto é, os agrupamentos que serão alvos na classificação), caracterizando-os e definindo seus perfis (via característica média ou outra estatística/descrição).

Para esse fim, técnicas de redução de dimensionalidade a fim de aprimorar a visualização e entendimento dos agrupamentos são muito úteis e devem ser utilizadas. Dentre as existentes, pode-se destacar duas: PCA (*Principal Component Analysis*) e t-SNE (*t-Distributed Stochastic Neighbor Embedding*).

A partir disso, os procedimentos envolvendo a tarefa de classificação podem ser executados, tendo como próximos passos:

1) **Etiquetar dados com base nos *clusters* ideais**
   a) Criação de rótulos para treino e teste
2) **Treinamento do Classificador**
   a) Divisão dos dados
   b) Escolha do algoritmo (KNN)
   c) Treinamento do modelo
   d) Avaliação

Para a avaliação de todos os modelos é comumente utilizada a validação cruzada (*cross-validation*), nesse caso fornecida pela biblioteca *sklearn*.

A técnica que envolve a validação cruzada é a *k-fold*, onde o conjunto de dados é dividido em k subconjuntos (chamados de *folds*). Após a execução da validação nestas partições, retorna uma métrica de performance para cada *folder* de teste. Por padrão, essa métrica é a acurácia, adicionando-se também a *F-score*.

Faz-se o uso também da ideia de *predict*, que permite obter as predições do modelo ao invés de apenas as métricas finais. Neste caso, cada predição será obtida para o conjunto de teste de cada uma das partições.

Em outras palavras, se o parâmetro passado for igual a 5, o modelo será treinado em 4 partições e testado em 1, que gera as predições. Ao final das execuções, os resultados são concatenados e retornados.

Outra ideia utilizada é a de *stratify*, que retorna os índices da partição de maneira aleatória e estratifica os dados de acordo com as classes do problema. Em outras palavras, balanceia a quantidade de amostras de cada classe entre a partição a fim de equilibrar a base de dados. É particularmente útil quando se está trabalhando com dados desbalanceados e deseja-se garantir que ambos os conjuntos de dados representem corretamente as proporções das classes.

Feito isso, é esperado que os clientes sejam identificados e segmentados de forma a satisfazer o objetivo proposto.

IV. Resultados

Nesta seção, apresenta-se os resultados da análise realizada a partir da clusterização e classificação dos clientes da base de dados, tendo como objetivo definir um possível ICP e/ou traçar um perfil de cliente. Os resultados obtidos são apresentados a seguir.

Com o objetivo de iniciar a clusterização, primeiramente foi necessário padronizar os dados para dimensionar os recursos e deixá-los na mesma escala.

Após isso, utilizou-se o método *Elbow* para determinar o número ideal de *clusters* que devem ser gerados. Isso foi feito utilizando-se do Erro Quadrático Total (SSE).

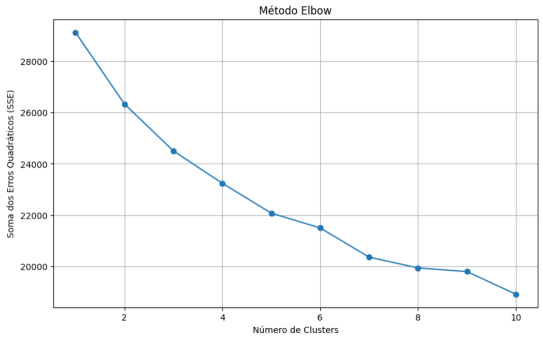

Fig. 3. Gráfico do método *Elbow*

Baseado tanto no gráfico quanto na experimentação, o número de *clusters* foi definido em 3.

O K-Means, posteriormente, foi utilizado para ajudar a segmentar os dados em grupos distintos e entender a estrutura

geral dos dados, sendo útil para encontrar padrões que podem ser explorados mais a fundo.

Tendo o algoritmo executado e os *clusters* gerados, podemos visualizar melhor aqueles produzidos e os atributos usados na clusterização.

Para isso, as duas técnicas de redução de dimensionalidade já mencionadas anteriormente foram usadas para melhorar a exibição: PCA e t-SNE.

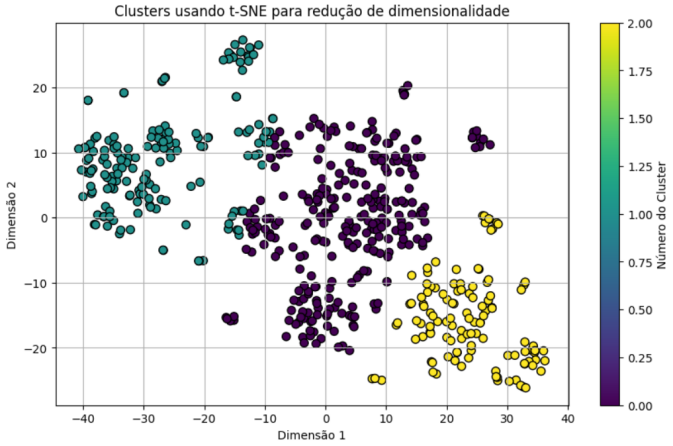

Fig. 4. Visualização dos *clusters* gerados a partir do t-SNE

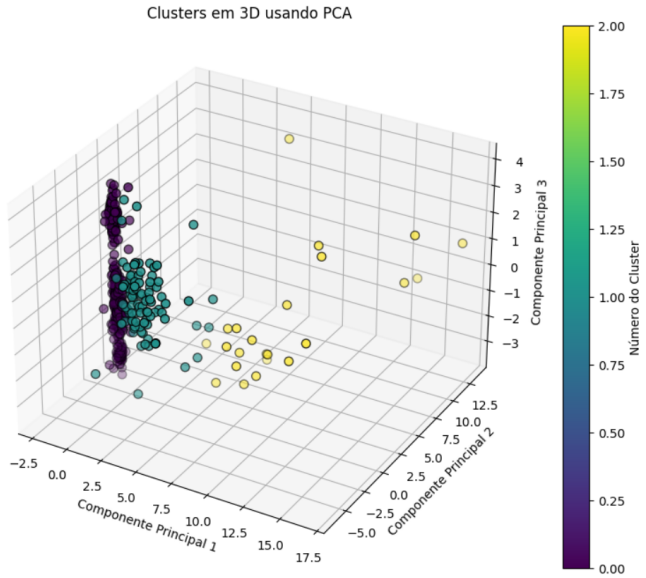

Fig. 5. Visualização dos *clusters* gerados a partir do PCA

Do ponto de vista dos *clusters*, as seguintes percepções puderam ser feitas:

- **Cluster 0:** definido como "clientes com potencial não explorado", possui alta realização de reuniões e participação de decisores, mas pouca ou nenhuma produção de diagnósticos e propostas, maior quantidade de vidas e taxa de sinistralidade. Apesar dos bons indicadores iniciais, essas empresas não avançam para etapas decisivas, mas possuem um alto potencial de receita.
- **Cluster 1:** definido como "clientes com avanço significativo", possui alta taxa de sucesso em reuniões e diagnósticos, alta participação de decisores, ticket médio e quantidade de vidas menores. Empresas neste *cluster* estão progredindo bem nas etapas de venda, mas o potencial de receita é menor devido ao ticket médio mais baixo.
- **Cluster 2:** definido como "clientes com baixo potencial", possui alta média de perda de negócios, pouca ou nenhuma progressão nas etapas de vendas e baixa participação de decisores. Essas empresas apresentam o menor potencial de conversão, sugerindo que os esforços aqui devem ser minimizados ou reavaliados.

A partir disso, os *clusters* relevantes foram definidos como sendo apenas 0 e 1 e, portanto, é o foco da aplicação do KNN.

Da perspectiva da aplicação do *K-Nearest Neighbors* a partir dos *clusters* de interesse, os seguintes passos foram executados:

1) Separação dos dados e atribuição da verdade sobre eles
2) Divisão os dados em conjunto de Treinamento e Teste
3) Treinamento do modelo
4) Validação do modelo através de validação cruzada
5) Avaliação do desempenho no conjunto de Teste

Baseado nisso e tendo executado o algoritmo, a performance foi medida baseada nas métricas e resultados expostos a seguir.

O primeiro valor abordado é o **cross-validation score**. Essa métrica representa o valor da acurácia do modelo em cada uma das cinco iterações da validação cruzada. Os valores variam entre 0.6 e 0.88461538, indicando que o modelo tem uma performance consistente, mas não perfeita.

```
[0.65384615, 0.65384615, 0.88461538, 0.6, 0.84]
```

Fig. 6. *Cross-Validation Scores*

Outro dos valores é o **mean cross-validation score**, o valor médio da acurácia nas cinco iterações da validação cruzada. Com uma média de aproximadamente 0.726, o modelo novamente indica um desempenho razoável, mas não excelente.

```
[0.7264615384615385]
```

Fig. 7. *Mean Cross-Validation Score*

O próximo dos resultados é a já mencionada **matriz de confusão**, onde é possível ver a quantidade de classificações corretas e incorretas por classe; mostra os seguintes valores:

- **13 verdadeiros negativos:** o modelo previu corretamente a classe 0 (não cliente ideal) em 13 casos.
- **7 falsos positivos:** o modelo previu incorretamente a classe 1 (cliente ideal) para 7 casos que eram da classe 0.

Fig. 8. Matriz de Confusão

- **3 falsos negativos:** o modelo previu incorretamente a classe 0 para 3 casos que eram da classe 1 (cliente ideal).
- **2 verdadeiros positivos:** o modelo previu corretamente a classe 1 em 2 casos.

Outro resultado que fornece bastante informação a respeito das métricas é o ***classification report***, o relatório de classificação para precisão, *recall* e *F1-score*.

	precision	recall	f1-score	support
0	0.81	0.65	0.72	20
1	0.22	0.40	0.29	5
accuracy			0.60	25
macro avg	0.52	0.53	0.50	25
weighted avg	0.69	0.60	0.63	25

Fig. 9. *Classification Report*

- Para a classe 0 (cliente não ideal):
  - **Precisão (0.81):** 81% das predições de classe 0 estavam corretas.
  - ***Recall* (0.65):** 65% dos casos que realmente eram classe 0 foram corretamente identificados.
  - ***F1-score* (0.72):** combinando precisão e *recall* em uma única métrica, faz 72% ser um valor razoável para a classe 0.
- Para a classe 1 (cliente ideal):
  - **Precisão (0.22):** 22% das predições de classe 1 estavam corretas.
  - ***Recall* (0.40):** 40% dos casos que realmente eram classe 1 foram corretamente identificados.
  - ***F1-score* (0.29):** é um valor baixo para a classe 1.
- ***Accuracy* (0.60):** o modelo classificou corretamente 60% do conjunto de teste.
- ***Macro avg*:** média das métricas de precisão, *recall* e *F1-score* sem levar em conta a distribuição das classes. Os valores razoáveis (0.52 para *precision*, 0.53 para *recall*, e 0.50 para *f1-score*) refletem o mau desempenho na classe minoritária (classe 1).
- ***Weighted avg*:** média ponderada das métricas, considerando a distribuição das classes. Embora a *weighted avg* seja um pouco melhor, ela ainda reflete o desequilíbrio e a dificuldade do modelo em prever a classe 1.

As imagens abaixo são representações gráficas dos valores obtidos durante a execução do modelo. Para exibir as informações de desempenho do modelo de KNN e facilitar a interpretação, os gráficos abaixo foram utilizados.

O primeiro deles é a própria **matriz de confusão** que, embora já tenha sido exibida, teve uma nova visualização:

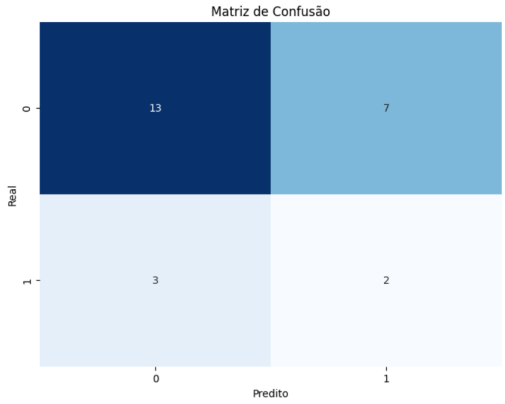

Fig. 10. Matriz de Confusão para perfil ideal e não ideal

Nessa configuração de matriz de confusão para classificação binária, a posição acompanha o seguinte padrão:

- 0,0 indica a contagem de verdadeiros negativos (TN);
- 0,1 indica a contagem de falsos positivos (FP);
- 1,0 indica a contagem de falsos negativos (FN);
- 1,1 indica a contagem de verdadeiros positivos (TP).

Outro gráfico utilizado foi a **Curva ROC (*Receiver Operating Characteristic*)** e **AUC (*Area Under the Curve*)**, que mostra o desempenho do classificador para vários limiares de probabilidade. É geralmente o ideal para visualizar a taxa de verdadeiros positivos versus a taxa de falsos positivos.

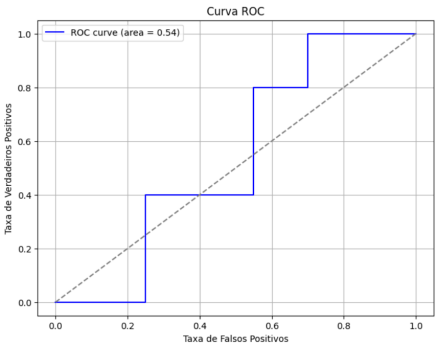

Fig. 11. Curva ROC com área igual a 0.42

O gráfico de **distribuição de probabilidades** também foi usado. Esse é um histograma gerado para comparar a distribuição das probabilidades preditas para clientes ideais e não ideais, ajudando a visualizar onde o modelo se confunde.

É possível concluir a partir dos resultados e gráficos que o modelo apresenta uma boa performance para a classe dominante (”0”), com uma precisão e *recall* altos. O *score* médio de validação cruzada de 72,6% indica que o modelo pode ser

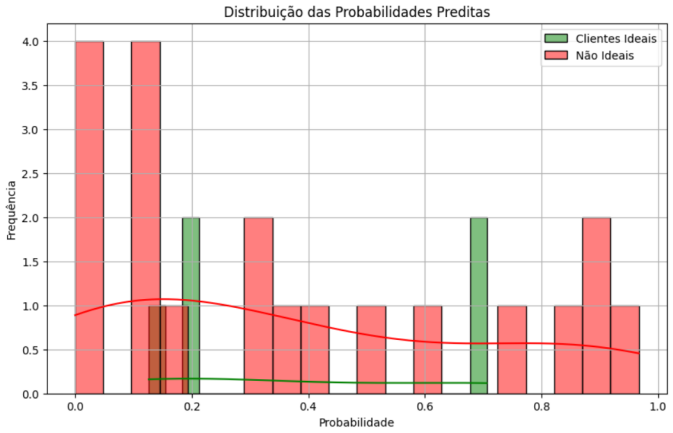

Fig. 12. Distribuição das Probabilidades de clientes ideais e não ideais

aprimorado para que seja robusto em diferentes divisões dos dados.

A principal fraqueza do modelo é a baixa capacidade de prever a classe "1". A precisão e o *recall* baixos mostram que a condição para o cliente ideal é crítica e sugerem que o modelo precisa ser ajustado.

Mesmo fazendo uso de estratégias de rebalanceamento de classes - como *oversampling* através do SMOTE -, reavaliação das *features* utilizadas, ajuste dos hiperparâmetros através de busca em grade e afins, ainda sim o modelo teve uma performance que refletiu o desbalanceamento de classes na base de dados.

Outro ponto importante é o fato de o modelo não atribuir uma probabilidade de o cliente ser ideal maior do que 70%, corroborando com o desbalanceamento já mencionado e a impossibilidade de prever clientes potenciais com uma maior confiabilidade.

V. Finalização

A combinação híbrida de um algoritmo não supervisionado (K-Means) com um algoritmo sob supervisão (KNN) permite uma abordagem mais robusta na criação do modelo, possibilitando um bom desempenho do modelo desenvolvido.

Porém, observando o comportamento e performance do modelo, percebe-se que esse bom desempenho só ocorreu para a classe maioritária (0 ou não ideal), fazendo com que seja necessário ajustes em diferentes partes do processo para aprimorar o resultado final do modelo.

Do ponto de vista da utilização do K-Means para pré-processamento e geração de *features*, teve-se uma gama variada de ações visando facilitar a tarefa de classificação, como:

- agrupar os dados e usar seus centróides como possíveis *features*;
- encontrar características ocultas/abstratas dos dados e usar estas como entrada para o KNN;
- pré-processar os dados e aplicar o KNN separadamente em cada *cluster*.

Desta forma, até o momento de aplicar o algoritmo de classificação, tornou-se fácil a adaptação do modelo de acordo com o comportamento dos dados e o resultado fornecido pelo mesmo.

Portanto, para que o resultado final da tarefa de criação de um modelo de aprendizado que aproxime a identificação do Perfil de Cliente Ideal na área de Tecnologia em Saúde seja satisfatório como um todo, serão necessárias modificações baseadas nas percepções dos especialistas e envolvidos, para possibilitar que seja entregue um protótipo treinado capaz de prever se um cliente é ideal ou não, baseando-se nas características dele mesmo e de outros.

References

[1] L. M. R. Vasconcelos and C. L. Carvalho, "Aplicação de regras de associação para mineração de dados na web," Universidade Federal de Goiás, Technical Report, 2004.

[2] R. Wieringa, *Design Science Methodology for Information Systems and Software Engineering*. Springer, 2014. doi: 10.1007/978-3-662-43839-8.

[3] A. D. Gordon, *Classification*, Chapman and Hall, 1981.

[4] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, 2nd ed., Morgan Kaufmann Publishers, 2006.

[5] T. Fawcett and F. Provost, *Data Science para negócios: O que você precisa saber sobre mineração de dados e pensamento analítico de dados*. Alta Books Editora, 2018.

[6] A. Pacheco, "Introdução ao Scikit-learn - Parte 3: avaliando a qualidade do modelo via cross-validation," 2021. [Online]. Available: http://computacaointeligente.com.br/outros/intro-sklearn-part-3/.

[7] A. Pacheco, "Avaliação de modelos, cross-validation e data leakage," 2021. [Online]. Available: http://computacaointeligente.com.br/conceitos/avaliando-performance-cross-validation/.

[8] A. R. Gonçalves, "Máquina de Vetores Suporte," [Online]. Available: https://andreric.github.io/files/pdfs/svm.pdf.

[9] G. B. Junior, "Máquinas de Vetores Suporte," [Online]. Available: https://nca.ufma.br/ geraldo/vc/12.SVM.pdf.

[10] G. M. O. Junior, "Máquina de Vetores Suporte: estudo e análise de parâmetros para otimização de resultado," Monografia (Bacharel em Ciência da Computação), Universidade Federal de Pernambuco, 2010. [Online]. Available: https://www.cin.ufpe.br/ tg/2010-2/gmoj.pdf.

[11] J. S. Russel and P. Norvig, *Artificial Intelligence: A modern Approach*. Prentice Hall, 1995.

[12] J. F. G. Monico, A. P. D. Póz, M. Galo, M. C. dos Santos, and L. C. de Oliveira, "Acurácia e Precisão: revendo os conceitos de forma acurada," *Boletim de Ciências Geodésicas*, vol. 15, no. 3, pp. 469-483, Jul. 2009. [Online]. Available: https://www.redalyc.org/pdf/3939/393937709010.pdf.

[13] R. S. Leal, "Métricas Comuns em Machine Learning: como analisar a qualidade de chat bots inteligentes — métricas," 2017. [Online]. Available: https://medium.com/as-m%C3%A1quinas-que-pensam/m%C3%A9tricas-comuns-em-machine-learning-como-analisar-a-qualidade-de-chat-bots-inteligentes-m%C3%A9tricas-1ba580d7cc96.

[14] D. Mariano, "Métricas de avaliação em machine learning," 2021. [Online]. Available: https://diegomariano.com/metricas-de-avaliacao-em-machine-learning/.

[15] E. Bertran, "O que é Precisão e Revocação," 2020. [Online]. Available: https://medium.com/computando-arte/o-que-%C3%A9-precis%C3%A3o-e-revoca%C3%A7%C3%A3o-b0b991b67cde.

[16] D. Almeida, "Perfil de Cliente Ideal (ICP): o que é e como definir?" Salesforce, 2021. [Online]. Available: https://www.salesforce.com/br/blog/icp-perfil-de-cliente-ideal/.

[17] CINNECTA, "Perfil do melhor cliente: importância e como definir o seu ICP." [Online]. Available: https://cinnecta.com/conteudos/perfil-do-cliente/.

[18] B. Ekbote, "Modeling Ideal Customer Profile for Maximum Return on Investment," 2017. [Online]. Available: https://ssrn.com/abstract=3670461.

[19] M. A. Syakur, B. K. Khotimah, E. M. S. Rochman, and B. D. Satoto, "Integration K-Means Clustering Method and Elbow Method For Identification of The Best Customer Profile Cluster," in *IOP Conference Series: Materials Science and Engineering*, vol. 336, p. 012017, 2018. doi: 10.1088/1757-899X/336/1/012017.

[20] R. Linden, "Técnicas de Agrupamento," *Revista de Sistemas de Informação da FSMA*, no. 4, pp. 18-36, 2009. [Online]. Available: https://www.fsma.edu.br/si/edicao4/FSMA_SI_2009_2_Tutorial.pdf.

[21] E. M. Real, "Comportamento de Aplicações de Realidade Virtual Distribuídas por meio de Clusters," M.Sc. thesis, Centro Universitário Campo Limpo Paulista (UNIFACCAMP), 2019. [Online]. Available: https://www.cc.faccamp.br/Dissertacoes/Eduardo_Machado_Real_dissertacao.pdf.

[22] K. Tabianan, S. Velu, and V. Ravi, "K-Means Clustering Approach for Intelligent Customer Segmentation Using Customer Purchase Behavior Data," *Sustainability*, vol. 14, 2022. doi: 10.3390/su14127243.

[23] O. Kramer, "K-Nearest Neighbors," in *Dimensionality Reduction with Unsupervised Nearest Neighbors*, vol. 51, Intelligent Systems Reference Library, Springer, 2013. [Online]. Available: https://doi.org/10.1007/978-3-642-38652-7_2.

[24] S. Zhang, X. Li, M. Zong, X. Zhu, and D. Cheng, "Learning k for kNN Classification," in *Proceedings of the 26th ACM International Conference on Information and Knowledge Management (CIKM)*, ACM, 2017. doi: 10.1145/2990508.

[25] S. Zhang, "Challenges in KNN Classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 6, pp. 2418-2432, 2021. doi: 10.1109/TKDE.2020.2990508.

[26] L. E. Peterson, "K-nearest neighbor," *Scholarpedia*, vol. 4, no. 2, 2009. doi: 10.4249/scholarpedia.1883.

[27] I. José, "KNN (K-Nearest Neighbors) #1," Medium, 2020. [Online]. Available: https://medium.com/brasil-ai/knn-k-nearest-neighbors-1-e140c82e9c4e.