Unindo KNN e Marketing Digital
Algoritmo de Recomendação

Enzo R. Tramontin[1], **Aline D. Riva**[1]

[1]Ciência da Computação – Universidade La Salle
Canoas, RS – Brazil

enzo.201910645@unilasalle.edu.br, aline.riva@unilasalle.edu.br

Abstract. *Artificial intelligence is being used by a wide variety of areas for data analysis, like recommendation, predictions, etc. One of the most popular techniques in that process is k-nearest neighbors, whose objective is to classify new data based on the classification of the most similar data. The data similarity is measured by a distance function, which calculates the difference between values of the new data and every other data in the database. The k-nearest neighbors algorithm is commonly used on data classification due to his simplicity and quality, which is directly linked to previous training as a supervised algorithm feature. This paper purposes an algorithm using k-nearest neighbors techniques to recommend new beauty focused courses to existing clients in the database.*

Resumo. *A inteligência artificial vem sendo utilizada nas mais diversas áreas para fins de análise de dados, recomendação de itens, previsões, entre outros. Uma das técnicas utilizadas neste processo é o uso do k-vizinhos mais próximos, cujo objetivo é classificar um novo dado com base na classificação dos indivíduos mais parecidos com ele. Medidos por uma função de cálculo da distância entre os valores do novo indivíduo e cada outro existente na base de dados. O algoritmo k-vizinhos mais próximos é comumente utilizado para classificação de dados devido à sua simplicidade e qualidade, sendo esta diretamente ligada ao treinamento prévio do algoritmo como característico nos algoritmos supervisionados. Este trabalho propõe um algoritmo que utiliza as técnicas do k-vizinhos mais próximos para recomendar novos cursos na área de estética a clientes já existentes na base de dados.*

1. Introdução

Com os avanços tecnológicos, cada vez mais utilizamos máquinas conectadas como facilitadoras em nosso dia-a-dia, estas que por sua vez geram uma enorme quantidade de dados todos os dias. Aproximadamente 2,5 quintilhões de bytes de dados são criados diariamente a partir de postagens em redes sociais, upload de fotos e vídeos, registros de transações comerciais, sinais de GPS, rastros de navegação e sensores de diversos tipos [Sodré 2016].

Tendo em vista a quantidade de informação gerada pelos mais diversos meios diariamente, podemos afirmar que com o passar dos anos a análise humana de tanta informação se tornará inviável e será necessário o desenvolvimento de algoritmos capazes de adquirir conhecimento automaticamente, por meio da análise de dados em massa [Vianna and Dutra 2016].

Para criação de tais algoritmos utilizamos um conjunto de técnicas, chamadas de inteligência artificial (IA), que permitem a simulação da inteligência humana por um computador. Sendo uma das ramificações da ciência da computação, a inteligência artificial compreende sistemas inteligentes modelados com características presentes no comportamento humano [Braga 2002]. Dentre suas diversas aplicações, a atividade de classificação de dados é a mais comum, pois permite uma tomada de decisão ou previsão de resultado a partir de um conhecimento obtido anteriormente. [Braga 2002].

Por conta de seu custo-benefício no mundo digital, um setor que está adotando cada vez mais técnicas de IA é o de marketing organizacional, especialmente de pequenas empresas que anteriormente eram bloqueadas dos canais de marketing tradicionais. Estas técnicas de análise de dados estruturados e não estruturados permitem que as empresas aumentem a eficácia e eficiência de seus anúncios por meio de estratégias de recomendação, direcionando os mesmos a um público alvo seleto [Stanton and Stanton 2019].

Dentre as técnicas utilizadas em algoritmos de recomendação, a k-vizinhos mais próximos (KNN) se destaca pela facilidade de implementação e qualidade de resultados, pois busca classificar um dado indivíduo com base na classificação mais comum dentre os indivíduos mais próximos dele. Este tipo de algoritmo é amplamente utilizado em tarefas analíticas, como reconhecimento de padrões, análises textuais e reconhecimento de objetos, além de apresentar resoluções intuitivas e facilmente compreensíveis por meio da visualização da vizinhança [Le et al. 2021].

Vide a capacidade de reconhecimento de padrões e classificação do algoritmo KNN e as necessidades de direcionamento de anúncios do marketing organizacional surge a questão: Existe a possibilidade de recomendar serviços personalizados utilizando KNN?

Diante do contexto acima, a presente pesquisa tem por objetivo introduzir um algoritmo de recomendação, estruturado com uma base de dados reais de clientes e cursos de uma escola na área da estécica e que utiliza as técnicas do KNN para classificação de novos cursos e, por fim, recomendar os mesmos à clientes específicos com base nesta classificação. O termo classificação consiste na criação de um procedimento capaz de aprender e agrupar dados com base em um conjunto de atributos ligados a ele.

Na próxima seção serão apresentados os objetivos da pesquisa, seguido da metodologia, do referencial teórico, da proposição de um algoritmo de recomendação e das considerações finais da pesquisa.

2. Objetivos

2.1. Objetivo geral

Desenvolver um algoritmo de recomendação com uso da técnica KNN a fim de utilizá-lo em uma base de dados real durante o lançamento de novos cursos na área de estética. Onde o algoritmo deve classificar o curso e, com base nesta classificação, identificar potenciais clientes para recomendá-lo.

2.2. Objetivo específicos

- Analisar os dados de clientes existentes e centralizá-los em uma base de dados
- Normalizar os dados existentes
- Estudar formas de implementação do algoritmo KNN

- Estudar sobre a integração com softwares de mensageria e e-mail
- Escolher a linguagem de programação mais adequada
- Definir os critérios de classificação
- Documentar as principais funcionalidades do algoritmo

3. Metodologia

Tendo em vista que este artigo visa introduzir um algoritmo, caracteriza-se como uma pesquisa aplicada à solução de um problema específico de retenção e captação de clientes por meio do uso de um algoritmo classificador. A pesquisa aplicada é definida pela utilização, aplicação e consequências de conhecimentos científicos na resolução de problemas variados [Assis 2009].

Sendo assim, este trabalho apresentará um algoritmo capaz de identificar padrões em uma base de dados e com estes classificar novos cursos, a fim de fazer uma ligação entre os melhores clientes possíveis para um determinado curso, com objetivo de aumentar a retenção de clientes e as vendas, onde o resultado está diretamente ligado à acurácia da classificação. Caracterizando-se como uma pesquisa qualitativa que não possui comprovação numérica e estatística, mas propõe uma interpretação por meio de uma análise detalhada e consistente das afirmações, assim como em argumentações lógicas de ideias [Michel 2005].

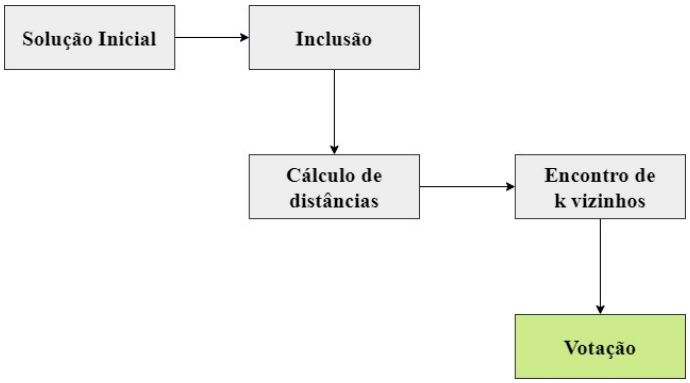

Figura 1. Diagrama de um algoritmo KNN básico

No KNN, um indivíduo é um determinado dado a ser classificado. Sendo necessário a presença de uma solução inicial ou base de dados, como característica dos algoritmos supervisionados, para que se possa classificar novos indivíduos. Após a inclusão de um novo dado ao conjunto existente, deve-se calcular a distância entre ele e cada indivíduo já existente na solução. Durante sua execução, deve ser determinado um valor k de indivíduos a serem comparados na classificação do novo indivíduo, sendo este valor de suma importância para a acurácia do algoritmo pois a classificação do novo indivíduo será a mesma da maioria dentre os comparados. Em caso de empate a classificação determinada será a do indivíduo mais próximo.

Um mesmo algoritmo pode ser aplicado a diversas soluções, desde que se tenha uma base de dados confiável e a representação de um indivíduo esteja clara. As seguintes seções explicarão a representação de um indivíduo e o funcionamento de cada fase da implementação do algoritmo, são elas: solução inicial, inclusão de um novo indivíduo, cálculo de distâncias, encontro dos vizinhos mais próximos e votação.

4. Referencial Teórico

4.1. K-Vizinhos mais Próximos

O algoritmo k-vizinhos mais próximos é um algoritmo supervisionado de aprendizagem de máquina que, utiliza da experiência adquirida através da fase de treinamento para gerar mais conhecimento e ser capaz de prever informações faltantes em novos dados [Shalev-Shwartz and Ben-David 2014]. Após a definição da base de testes, o algoritmo utiliza a classificação mais comum entre os indivíduos/vizinhos mais próximos do novo para classificá-lo. A Figura 1 representa as etapas para execução de um algoritmo KNN simples.

4.1.1. Representação de um Indivíduo

Um indivíduo dentro do contexto do KNN é um dado já classificado ou a ser classificado, onde um conjunto de indivíduos já classificados representa a solução inicial. Ele pode ser visto como uma tradução do problema original para o mundo digital, possuindo cada característica da solução dividida em um vetor por exemplo, onde cada coluna representa um possível fator determinante na classificação do indivíduo e portanto indicará sua posição na solução. A representação de um indivíduo pode ser traduzida como uma análise de um problema real, ligando-o ao mundo digital por meio da transformação de conceitos em dados que permitam a exploração do espaço de busca delimitado pelas definições [Eiben and Smith 2003]. Abaixo teremos a Figura 2 representando um indivíduo simples no formato de um vetor de dados, contendo traduções de características de um diagnóstico de exemplo.

Febre	Dores	Náusea	Diarreia	Diagnóstico
Sim	Não	Não	Sim	Doente

Figura 2. Indivíduo simples como um string de valores reais

Após a classificação de um indivíduo ele passará a fazer parte da base de dados e poderá ser utilizado para classificar novos indivíduos.

4.1.2. Solução Inicial

O algoritmo KNN é um exemplo de algoritmo supervisionado. Sendo assim, devemos partir de um treinamento em uma base de dados com características definidas e pré-rotuladas para que se possa reconhecer padrões e classificar novas informações [Silva 2017]. Para permitir uma visualização dos dados da solução inicial e do funcionamento do algoritmo, podemos alocar o conjunto de dados em um plano onde a posição de cada indivíduo depende de suas características, vale ressaltar que o posicionamento é independente da classificação. A Figura 3 demonstra os dados de uma solução inicial traduzida a um plano.

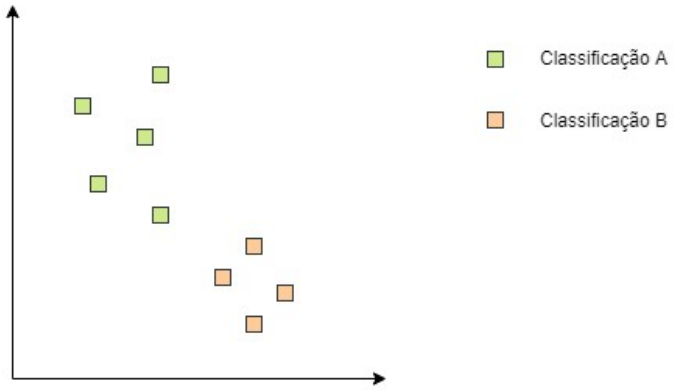

Figura 3. Solução inicial em um plano

4.1.3. Inclusão

Assim como os indivíduos da solução inicial, ao incluirmos um novo indivíduo não classificado na base ele terá uma posição baseada em suas características. O algoritmo KNN trabalha com o conceito de aprendizagem em instância, onde a classificação de um novo indivíduo dependerá de sua similaridade com os demais já classificados que compõem a base de testes [Buani et al. 2009]. A Figura 4 representa a inclusão de um novo indivíduo na base de dados.

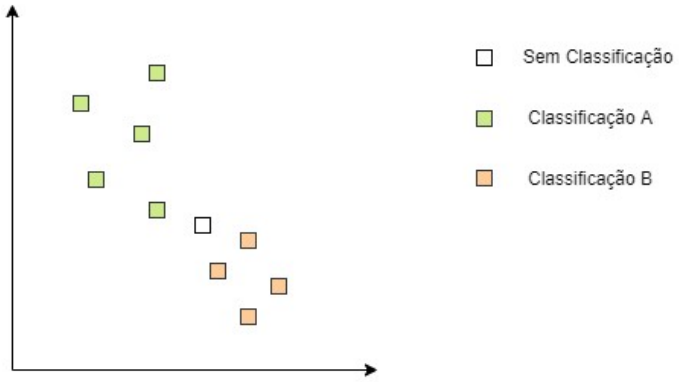

Figura 4. Inclusão de um novo indivíduo em uma base de dados

4.1.4. Cálculo de Distância

Após a inclusão de um novo indivíduo na base, para poder seguir com a classificação deve-se calcular a distância entre os demais dados já classificados e o novo. Na implementação do algoritmo KNN se utiliza o cálculo da distância euclidiana, baseada no teorema de Pitágoras [Amaral 2016]. Onde quanto mais próxima a distância entre dois indivíduos mais parecidos eles serão e quanto menor a distância maior será a acurácia na classificação de um novo indivíduo [Silva 2017].

$$d(i,j) = \sqrt{\sum_{k-1}^{n} (p_{ik} - p_{jk})^2}$$

Figura 5. Função de cálculo da distância Euclidiana [Silva 2005]

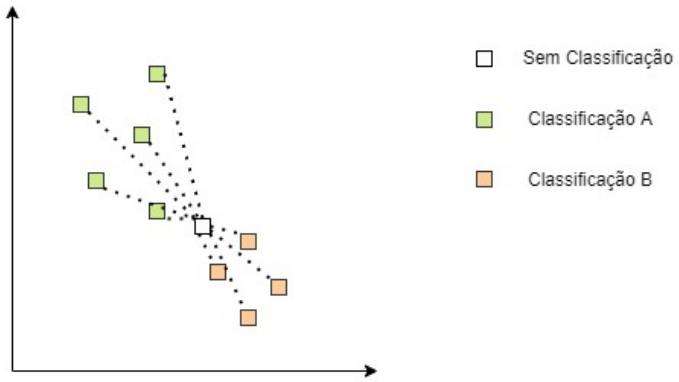

Figura 6. Demonstração da distância entre os dados existentes e o novo

4.1.5. Encontro dos vizinhos e votação

O algoritmo k-vizinhos mais próximos utiliza a classificação mais comum dentre os vizinhos do novo indivíduo para definir seu rótulo. Portanto é necessário determinar uma quantidade k de vizinhos a serem comparados nesta classificação, buscando utilizar valores muito menores que a quantidade total de indivíduos [Silva 2017].

Figura 7. Demonstração dos k-vizinhos mais próximos para k = 3

A Figura acima apresenta a classificação de um novo dado levando em consideração os 3 vizinhos mais próximos (k = 3). Como existem 2 vizinhos de classificação B e apenas 1 de classificação A, o novo registro será classificado como B.

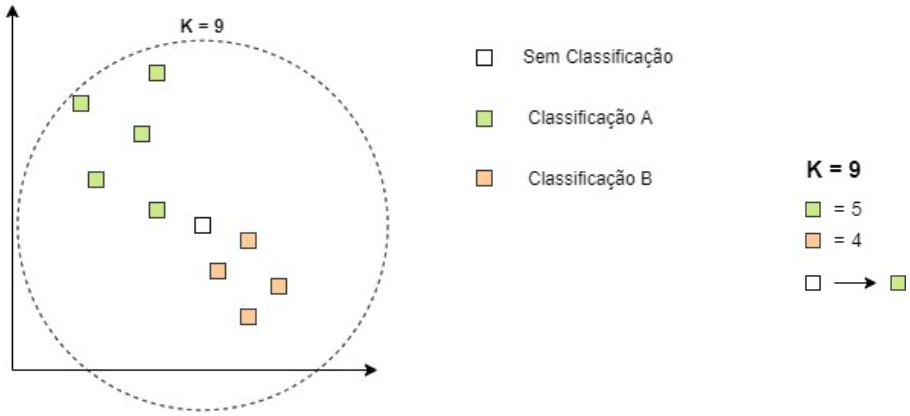

Figura 8. Demonstração dos k-vizinhos mais próximos para k = 9

A Figura acima apresenta um caso atípico onde o valor de k está muito acima do normal, ocasionando uma redução na acurácia do algoritmo. Foi levado em consideração os 9 vizinhos mais próximos (k = 9) para classificar um novo dado. Como existem 5 vizinhos de classificação A e apenas 4 de classificação B, o novo registro será classificado como A.

5. Implementação e Testes

Nesta seção do artigo será apresentada a implementação do algoritmo, que utiliza a técnica KNN no cadastro de novos cursos para identificar semelhanças entre o curso sendo inserido e os existentes na base de dados e assim encontrar potenciais clientes, sendo estes os clientes dos 'k' cursos mais semelhantes ao novo. Abaixo temos as principais tecnologias utilizadas, as definições para entendimento do protótipo seguido da implementação das principais funções e resultados obtidos.

5.1. Python

A linguagem de programação utilizada no algoritmo foi python, na versão 3.7, devido à grande quantidade de bibliotecas de inteligência artificial disponíveis na linguagem e sua performance [Python 2018]. Dentre as disponíveis, foram utilizadas as bibliotecas:

a) pandas para manipulação da base de dados [Pandas 2022];
b) numpy para cálculos [NumPy 2022];
c) scikit-learn para treinamento e avaliação do modelo [Scikit-learn 2019].

5.2. Base de Dados

A base de dados escolhida para implementação inicial do algoritmo foi utilizar arquivos de extensão .CSV, sendo estes arquivos de texto onde a ',' separa os valores. Utilizando a biblioteca pandas, disponível no python, é possível manipular facilmente este tipo de arquivo com funções específicas para .CSV [Pandas 2022].

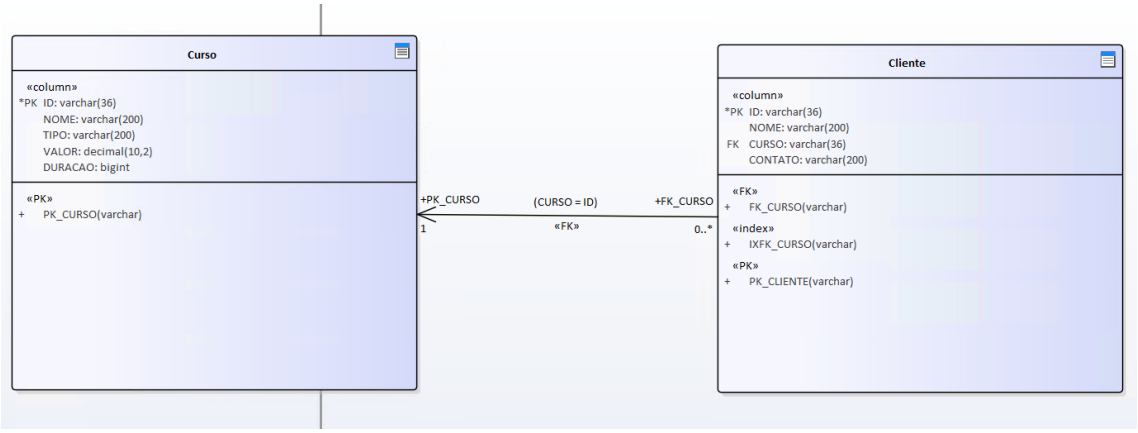

Figura 9. Modelagem da base de dados

A Figura 9 apresenta a modelagem da base de dados, onde foram criadas as entidades curso e cliente. Ambas estão conectadas pela chave estrangeira (FK) curso, onde o atributo curso na entidade cliente representa o identificador (ID) de um curso existente.

5.2.1. Cursos

Figura 10. Representação de um curso em vetor

A Figura 10 representa uma aproximação da entidade de cursos, traduzida a um vetor. Onde um curso possui um identificador inteiro, um nome do tipo texto, um valor decimal, um tipo texto e uma duração do tipo inteiro. A base inicial possui 13 cursos cadastrados, distribuidos nos tipos olhos, cabelos e unhas.

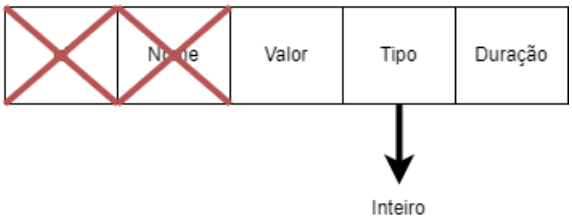

Figura 11. Normalização da entidade curso

A Figura 11 apresenta a normalização da entidade curso, preparando-a para uso no algoritmo. Foram retirados os campos Id e Nome pois não tem relevância na classificação e mantê-los pode prejudicar os resultados. Como o campo tipo é um texto, é necessário traduzi-lo ao tipo inteiro por necessidade da biblioteca numpy durante o cálculo de distância de cada atributo [NumPy 2022].

5.2.2. Clientes

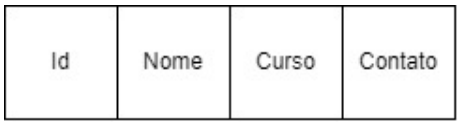

Figura 12. Representação de um cliente em vetor

A Figura 12 representa uma aproximação da entidade de clientes, traduzida também a um vetor. Onde um cliente possui um identificador inteiro, um nome do tipo texto, um curso do tipo texto ligado ao identificador de um curso existente e um contato do tipo texto. A base inicial possui 44 clientes cadastrados, distribuidos entre os cursos existentes. Como os clientes não serão classificados pelo algoritmo não é necessário normalizar a entidade.

5.3. Funções

5.3.1. Inclusão

```
def incluir(mostrar_cursos=False, valor=0, tipo=0, duracao=0):
    global cursos
    novo_curso = np.array([valor, tipo, duracao])
    distancias = get_distancias(novo_curso)
    vizinhos = get_vizinhos(distancias)
    clientes = get_clientes(vizinhos)

    if mostrar_cursos:
        print(get_cursos(vizinhos))

    return clientes
```

Figura 13. Função de inclusão

A Figura 13 apresenta a função de inclusão, cujo objetivo é incluir um novo curso na base de dados 'cursos', de acesso global, e recomendá-lo aos clientes dos cursos mais próximos deste, sendo necessário unir todas as demais funções para isto. Para criar um vetor que representará um novo curso, a função recebe por parâmetro o valor, tipo e duração dele, nota-se que o nome do curso não é informado pois seu valor não tem importância na determinação dos vizinhos. Abaixo temos a implementação das funções de cálculo de distância e encontro de vizinhos, chamadas após a criação do vetor e necessárias para determinar os clientes da recomendação, que serão retornados pela função.

5.3.2. Cálculo de Distância

```
def get_distancias(novo_curso):
    global cursos
    x = cursos.values
    return np.linalg.norm(x - novo_curso, axis=1)
```

Figura 14. Função de cálculo de distância

A Figura 14 apresenta a função de cálculo de distância, que utiliza a biblioteca numpy (np) e sua classe de álgebra linear (linalg) para encontrar a norma vetorial da diferença entre os valores dos cursos na base de dados e o novo curso sendo inserido. Retornando por fim a distância entre o novo curso e os demais.

5.3.3. Encontro dos vizinhos

```
def get_vizinhos(distancias):
    global k
    return distancias.argsort()[:k]
```

Figura 15. Função de encontro de vizinhos

A Figura 15 apresenta a função de encontro dos vizinhos, que recebe o vetor de distâncias e utiliza a função argsort da biblioteca numpy para ordenar as distâncias. Esta ordenação é necessária para determinar quais são os k, acessado de forma global, valores mais próximos do curso sendo inserido. Seu retorno é um recorte do vetor de distância, sendo retornados apenas os k valores mais próximos.

5.3.4. Correlação dos valores

```
def get_correlacao(coluna):
    global cursos
    return cursos.corr()[coluna]
```

Figura 16. Função de correlação

A Figura 16 apresenta a função de correlação dos valores, que recebe uma coluna a ser analisada como parâmetro e utiliza a função corr da biblioteca pandas para determinar o quão relacionados estão os valores da coluna analisada com os valores das demais colunas na base de dados. Esta função não é utilizada na rotina de inclusão mas fornece uma informação importante pois está diretamente ligada ao resultado da mesma, pois apresenta os valores mais impactantes na determinação dos vizinhos mais próximos de um novo curso. Seu retorno é um vetor contendo valores decimais entre 0 e 1, onde quanto mais próximo de 1 mais relacionadas as colunas estão.

5.4. Resultados

Dados do Curso			Resultado			
Valor	Tipo	Duracao	Olhos	Cabelos	Unhas	Clientes
500	Cabelos	12	2	0	1	5
800	Olhos	80	1	1	1	11
1100	Cabelos	100	0	2	0	15
300	Unhas	10	1	0	2	20
350	Olhos	10	1	0	2	16
300	Olhos	20	1	0	2	20
450	Unhas	20	2	0	1	5

Figura 17. Tabela de Resultados

A Figura 17 apresenta os resultados obtidos nos testes do algoritmo, onde em apenas 2 resultados de um total de 7 a recomendação foi feita, em sua maioria, à clientes do mesmo tipo de curso do que estava sendo inserido. Isto ocorre devido ao peso que os campos de valor e duração tem atualmente na base de dados, sendo praticamente irrelevante o valor inserido no tipo pois este tem pouca correlação com os demais. Para melhorar os resultados pode-se balancear a base de dados, melhorar o peso de cada atributo do curso e revisar o treinamento do algoritmo.

Dados do Curso - Pós Normalização				Resultado Detalhado por Curso		
Valor (Faixa)	Tipo	Duracao (Faixa)		Valor (Faixa)	Tipo	Duracao (Faixa)
2	Cabelos	2		2	Olhos	2
				2	Olhos	2
				2	Unhas	2
3	Olhos	4		5	Olhos	5
				2	Olhos	2
				4	Cabelos	3
4	Cabelos	5		5	Olhos	5
				5	Cabelos	5
				4	Cabelos	3
1	Unhas	1		2	Unhas	2
				1	Unhas	1
				2	Unhas	1
2	Olhos	1		1	Olhos	1
				2	Olhos	2
				2	Olhos	1
1	Olhos	2		1	Olhos	1
				2	Olhos	2
				2	Olhos	2
2	Unhas	2		2	Unhas	2
				2	Unhas	2
				2	Unhas	1

Figura 18. Tabela de Resultados Detalhada - Pós Balanceamento

A Figura 18 exibe os resultados obtidos após a normalização e balanceamento dos dados. Para isto, foram definidas 5 faixas nos campos valor e duração para reduzir seu peso durante a classificação de um novo curso. Observa-se que após esta alteração na base, em apenas 1 resultado de um total de 7 a recomendação foi feita incorretamente, sendo consideravelmente superior à aproximação anterior e enfatizando a importância do balanceamento de variáveis em algoritmos de inteligência artificial. Como forma de melhorar ainda mais a acurácia da solução, pode-se utilizar uma base de dados com mais dados disponíveis para treino do algoritmo.

6. Considerações Finais

Este artigo propõe um algoritmo que utiliza a técnica KNN para, a partir de uma base de dados inicial, recomendar um curso novo na área de estética para os clientes dos cursos mais próximos a este sendo inserido. Seu objetivo é encontrar potenciais clientes e melhorar a retenção de clientes anteriores.

Em trabalhos futuros pode-se implementar uma forma de contato automática com os clientes, via email ou whatsapp. Também pode-se implementar uma interface para facilitar a inserção de dados com telas de acesso aos cursos, clientes e resultados, além de utilizar um banco de dados real a fim de melhorar a manipulação dos dados.

Referências

Amaral, F. (2016). Aprenda mineração de dados: Teoria e prática. *São Paulo: Alta Books*.

Assis, M. (2009). Metodologia do trabalho científico. *Editora Universitária UFPB*, 3.

Braga, M. (2002). Indução automática de Árvores de decisão. *UFSC*.

Buani, B. et al. (2009). Aplicação do algoritmo dos k-vizinhos mais próximos para seleção de características da morfologia de asas de abelhas sem ferrão. *Universidade de São Paulo*, page 5.

Eiben, A. and Smith, J. (2003). Introduction to evolutionary computing. *Springer*.

Le, L. et al. (2021). Knn loss and deep knn. *IOS Press*, 182:95 – 110.

Michel, M. (2005). Metodologia e pesquisa científica: um guia prático para acompanhamento da disciplina e elaboração de trabalhos monográficos. *São Paulo: Atlas*.

NumPy (2022). Numpy documentation. https://numpy.org/doc/stable/.

Pandas (2022). Pandas documentation. https://pandas.pydata.org/docs/.

Python (2018). Python 3.7 documentation. https://docs.python.org/3.7/.

Scikit-learn (2019). Documentation of scikit-learn. https://scikit-learn.org/0.21/documentation.html.

Shalev-Shwartz, S. and Ben-David, S. (2014). Understanding machine learning: From theory to algorithms. *Cambridge: Cambridge University Press*.

Silva, C. (2017). Avaliação da qualidade da informação na área da saúde: Aplicação de algoritmos de aprendizado de máquina. *UCS*.

Silva, L. D. (2005). Uma aplicação de Árvores de decisão, redes neurais e knn para a identificação de modelos arma não-sazonais e sazonais. *Pontifícia Universidade Católica do Rio de Janeiro*, pages 53 – 55.

Sodré, L. (2016). Big data estratégico: Um framework para gestão sistêmica do ecossistema big data. *COPPE, UFRJ*.

Stanton, A. and Stanton, W. (2019). Closing the skills gap: Finding skilled analytics professionals for a dynamically changing data-driven environment. *Applied Marketing Analytics*, 5:170 – 184.

Vianna, W. and Dutra, M. (2016). Big data e gestão da informação: Modelagem do contexto decisional apoiado pela sistemografia. *Revista Informação e Informação, Londrina*, 21(1):185 – 212.