Universidade Federal do Rio Grande do Sul
Instituto de Informática

Aprendizado de Máquina
INF01017

ALGORITMOS DE AGRUPAMENTOS
K-MEANS

Gabriel Lando 291399
Juliane da Rocha Alves 285681

Porto Alegre **05 de maio de 2022**

SUMÁRIO

1. Introdução

Esse trabalho tem como objetivo aplicar o algoritmo de agrupamento *K-means* para tarefas de identificação de padrões nos dados por abordagem não-supervisionada. O algoritmo *K-means* será utilizado para detectar agrupamentos, para que, após isso, se possa realizar uma análise exploratória dos agrupamentos gerados com intuito de extrair informações não categorizadas dos dados.

O presente relatório pretende estudar quais características dos clientes ou da forma com que foi feito o contato da empresa de telemarketing estão associadas ao sucesso na venda do depósito a prazo baseado em dados coletados por uma instituição bancária portuguesa.

A linguagem de desenvolvimento escolhida para o trabalho foi Python, utilizou-se as bibliotecas do Scikit-Learn para os modelos, utilizou-se Numpy e Pandas para a manipulação dos dados, e as bibliotecas Matplot Lib e Seaborn para a geração dos gráficos e visualizações. O código foi desenvolvido utilizando o Jupyter.

2. Metodologia

2.1. Dataset

O *dataset* utilizado para a realização do trabalho foi fornecido juntamente da definição do trabalho. O arquivo consiste em um documento de texto com dados separados por tabulações. Esses dados foram adaptados de uma base de dados pública de uma instituição bancária portuguesa que visou, através de campanhas de *telemarketing*, vender depósitos a prazo para os clientes dessa instituição. Os atributos disponíveis no *dataset* estão descritos na Tabela 1.

Tabela 1: Atributos do dataset.

Atributo	Valor
age	Idade do cliente (valor numérico)
marital	Estado civil do cliente (valor categórico)
education	Nível de ensino do cliente (valor categórico)
housing.loan	Se o cliente possui um empréstimo para a casa (valor binário: 1=sim, 0=não)
personal.loan	Se o cliente possui um empréstimo pessoal (valor binário: 1=sim, 0=não)
contact	Forma com que foi feito o contato com o cliente, se

	por telefone fixo ou celular (valor categórico)
day.of.week	Dia da semana em que foi feito o contato com o cliente (valor categórico)
duration.contact	Duração do contato, em segundos (valor numérico)
number.contacts.campaign	Número de contatos feito com o cliente durante a campanha de telemarketing (valor numérico)
poutcome	Resultado da última campanha de marketing, se inexistente, sucesso ou fracasso (valor categórico)
term.deposit	Se o cliente subscreveu ao depósito a prazo (valor binário: 1=sim, 0=não)

Os dados fornecidos possuem ao todo 3836 linhas/clientes. Desses 3836 clientes, 3421 não subscreveram o depósito, e apenas 415 subscreveram. Antes de executar o algoritmo *K-means*, foi necessário preparar os dados. Todas as colunas com dados categóricos (*marital, education, contact, day.of.week* e *poutcome*) foram convertidos para o formato One-hot-encoding, onde o atributo relacionado à categoria de uma instância recebe valor 1, e os demais atributos da categoria recebem o valor 0. Após feito isso, os dados foram normalizados. Para isso, utilizou-se a função *MinMaxScaler* da biblioteca do *Scikit-Learn* para deixar os valores de todos os atributos entre -1 e 1.

2.2. Escolha do valor de K

Após a organização dos dados, foi necessário encontrar o melhor valor de clusters de agrupamento dos dados. Para isso, foi utilizado a biblioteca *Scikit-Learn* que implementa o algoritmo *k-means*. Primeiramente, foram avaliados os valores de *k* entre 1 e 15, com o intuito de investigar dentro desse intervalo qual seria o possível melhor valor de *k*, utilizando-se o método *Elbow*. O algoritmo utilizado está descrito abaixo.

```
# Clusters size. From 1 to 15
k_clusters = [i for i in range(1,16)]
results = []
labels = []

for k in k_clusters:
    k_means = KMeans(n_clusters=k, random_state=0)
    k_means.fit(data)
    results.append(k_means.inertia_)
```

A Figura 1 representa o resultado da execução do método *Elbow*. Como pode ser observado, é difícil definir qual o melhor valor de *k* a partir do resultado desse método. Entretanto, baseado na curva do gráfico, pode-se deduzir que o melhor valor de *k* se encontra entre o intervalo de *k* maior ou igual 2 e *k* menor ou igual a 5.

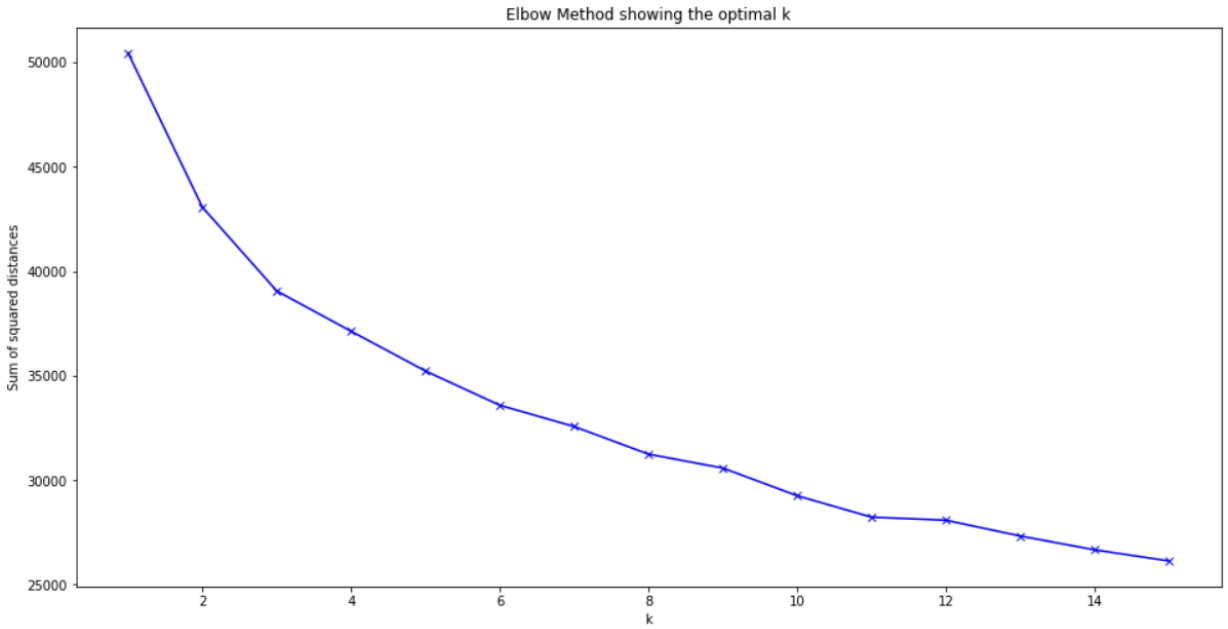

Figura 1: Método Elbow mostrando o valor ótimo de k.

Para ajudar a decidir qual quantidade de *clusters* traria a melhor classificação dos dados foi decidido criar uma visualização usando os 4 clusters gerados com base no gráfico da Figura 1. O código a seguir foi utilizado para executar o algoritmo *K-means*, com o algoritmo PCA para reduzir a dimensionalidade dos dados, com os 4 valores de clusters, alterando-se apenas o valor *n_clusters* no construtor da classe *KMeans* da biblioteca utilizada.

```
k_means = KMeans(n_clusters=5, random_state=0)
k_means.fit(data)
labels = k_means.predict(data)

pca = PCA(n_components=2)
principalComponents = pca.fit_transform(data)
principalDf = pd.DataFrame(data = principalComponents, columns =
["principal component 1", "principal component 2"])
finalDf = pd.concat([principalDf, pd.DataFrame(data=labels,
columns=["target"])], axis = 1)
```

A Figura 2 representa a execução do *K-means* com 2 clusters. Essa configuração foi descartada pelo grupo, pois ela foi diretamente influenciada pelo campo *term.deposit*, que informa se o cliente realizou ou não a compra do depósito a prazo. Sendo assim, os *clusters* possuem forte tendência a separar os dados de acordo com essa categoria, dificultando a análise dos dados e inferência de resultados pelo grupo.

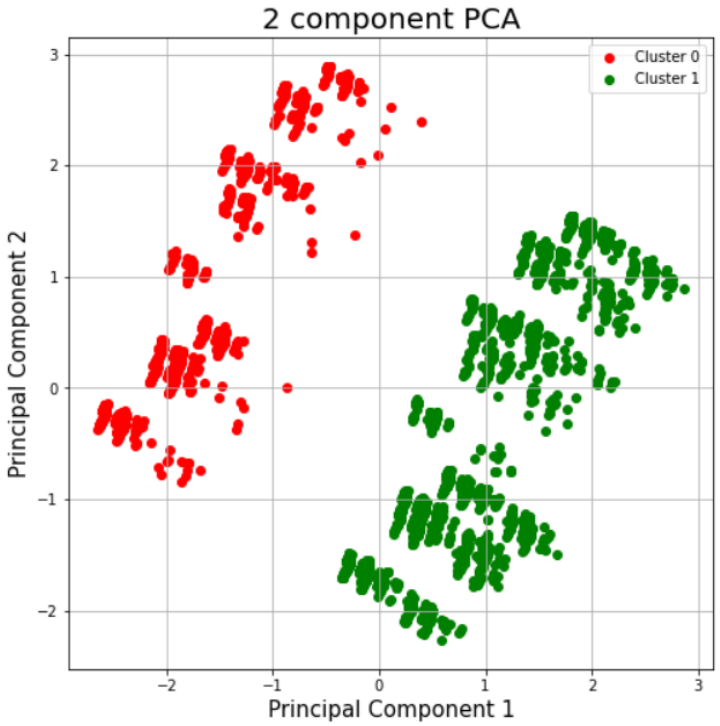

Figura 2: K-means com 2 clusters

A Figura 3 representa a execução do *K-means* com 3 clusters. Essa configuração também foi descartada pelo grupo, pois, analisando-se o gráfico, percebeu-se que o *Cluster 2*, identificado pela cor azul, possui dados sobre a região do *Cluster 1*, identificado de verde. Assim como no caso anterior, isso poderia dificultar a inferência de resultados pelo grupo.

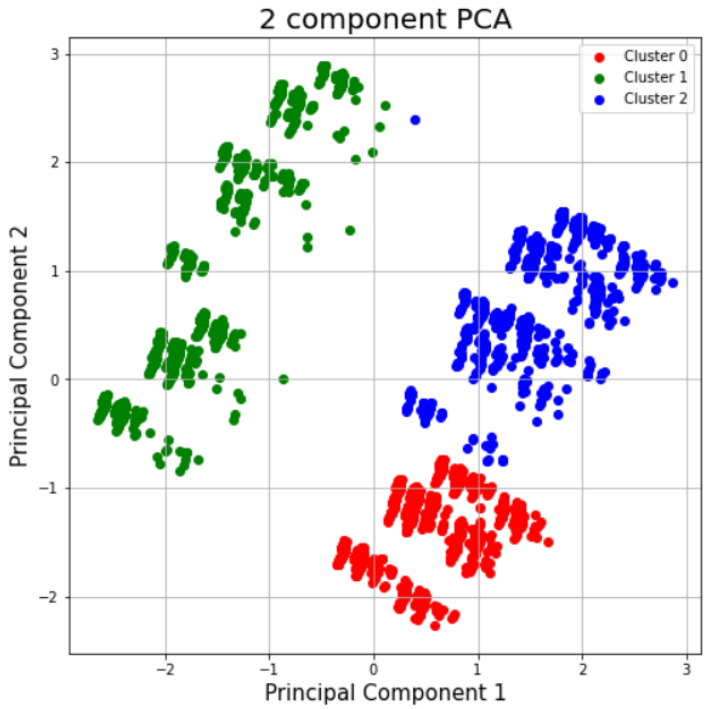

Figura 3: K-means com 3 clusters

A Figura 4 representa a execução do *K-means* com 4 clusters. O grupo optou por essa opção dada a clara diferenciação das regiões no gráfico. Ao escolher-se o valor de *k* igual a 4, a extração de *insights* se tornou mais óbvia pelo grupo, como pode ser visto na seção de Resultados do presente trabalho.

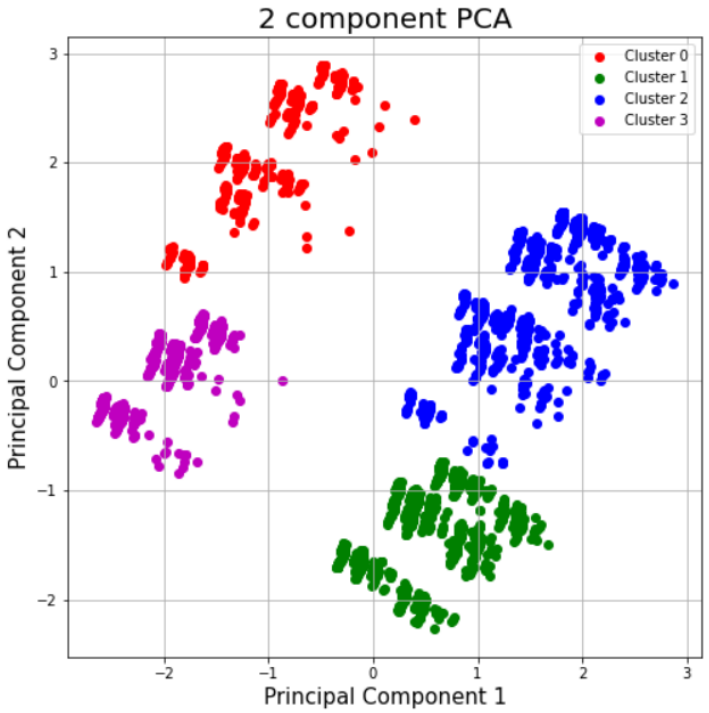

Figura 4: K-means com 4 clusters

A Figura 5 representa a execução do *K-means* com 5 clusters. Essa configuração também foi descartada pelo grupo, visto que havia uma sobreposição dos *Clusters 3* e *4*, representados nas cores magenta e ciano, respectivamente. Essa sobreposição de regiões poderia dificultar a inferência de resultados pelo grupo.

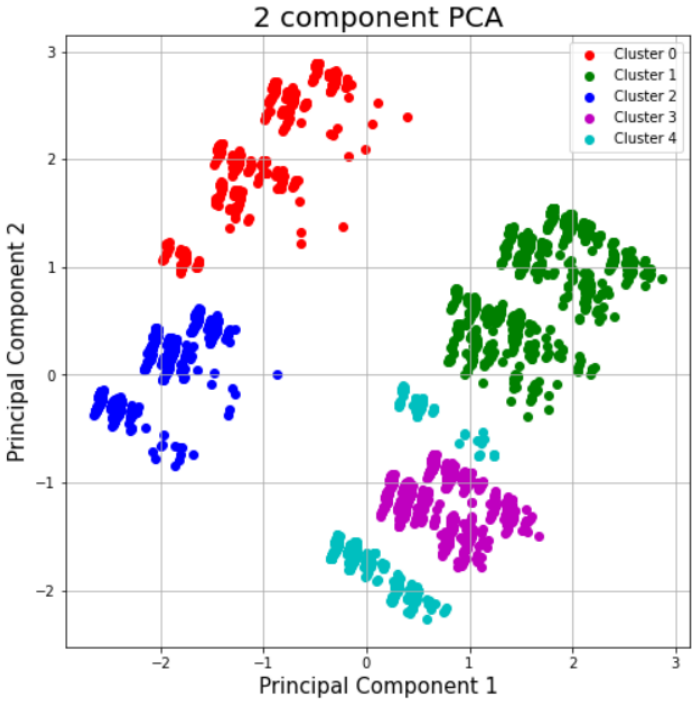

Figura 5: K-means com 5 clusters

3. Resultados

Após a escolha final do número de clusters, executou-se novamente o algoritmo *K-means* com K igual a 4. Com o resultado do algoritmo, adicionou-se a informação do número dos clusters aos dados iniciais para que pudéssemos compreender qual a semelhança e diferença entre os clusters.

```
k_means = KMeans(n_clusters=4, random_state=0)
k_means.fit(data)
labels = k_means.predict(data)

df_ = pd.concat([df, pd.DataFrame(data=labels, columns=["target"])],
axis=1)
```

Como podemos observar na Figura 6, o clusters 1 e 2 são os clusters que possuem um maior número de pessoas que subscreveram o depósito, sendo que o cluster 2 tem uma absorção de aproximadamente 15% (150 subscreveram o depósito

de um total de 1000, aproximadamente) e o cluster 1 tem uma absorção de 13,79% (150 subscreveram o depósito de um total de 1450, aproximadamente). Logo, os usuários no cluster 2 parecem ter uma chance maior de subscrever o depósito se comparado aos outros clusters. Entretanto é válido observar que mesmo que as chances dos clientes do cluster 2 sejam maiores, ainda assim é uma absorção baixa. Como há muitos clientes que não subscreveram no dataset, esse comportamento é esperado.

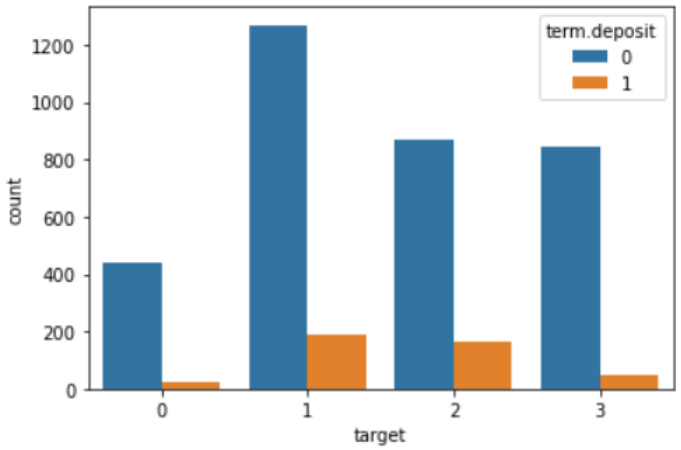

Figura 6: Número de clientes que subscreveram o depósito por cluster

Agora que já sabemos que os clusters 1 e 2 são os clusters que têm as maiores chances de possuírem clientes que subscrevem, podemos observar o que esses dois clusters têm em comum e de diferentes dos demais e entre si. Como podemos observar na Figura 7, a esmagadora maioria dos clientes que se encontram no cluster 1 são divorciados, quanto que os clientes do cluster 2 são divididos entre solteiros e divorciados, sendo os solteiros em maior número.

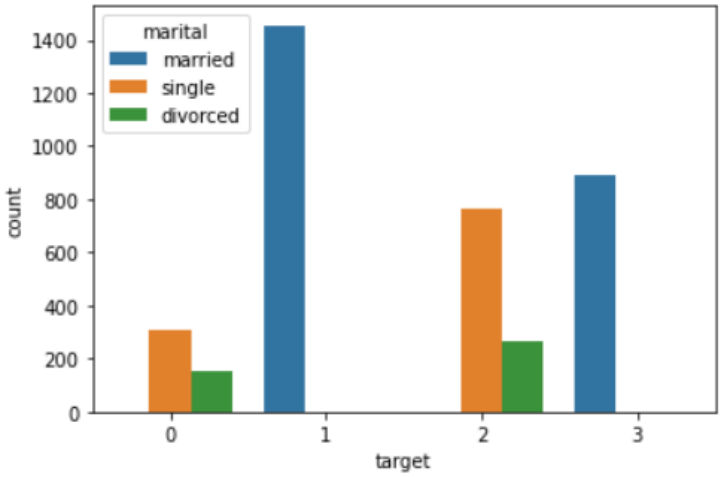

Figura 7: Número de clientes casados, solteiros e divorciados por cluster

Também podemos observar na Figura 8 que a relação idade por clusters mostra que a mediana para o cluster 1 é de aproximadamente 40 anos, e que a

mediana do cluster 2 é de aproximadamente 30 anos. Comparando a Figura 7 e 8, podemos observar que como o cluster 1 possui muitas pessoas casadas, é razoável que a faixa de idade para as pessoas nesse cluster seja em torno dos 40 anos, enquanto que para o cluster 2, onde possui mais solteiros, a faixa de idade é mais próximo dos 30 anos, com exceção dos outliers presentes para esse cluster, com clientes na faixa dos 60 a 90 anos, podendo ser os clientes divorciados.

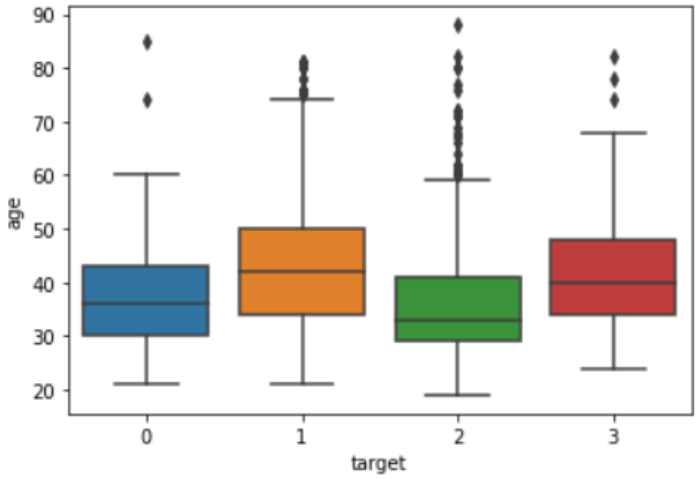

Figura 8: Relação da idade dos clientes por clusters

Como podemos observar na Figura 9, a quantidade de clientes que possuem empréstimo para casa tem um maior número no cluster 1. Dado que este cluster possui em sua esmagadora maioria clientes casados, faz sentido que as pessoas casadas peçam um empréstimo para casa, dado que estão indo morar com seus cônjuges. O cluster 2, por sua vez, está bem dividido entre não possuir um empréstimo para casa e possuí-lo.

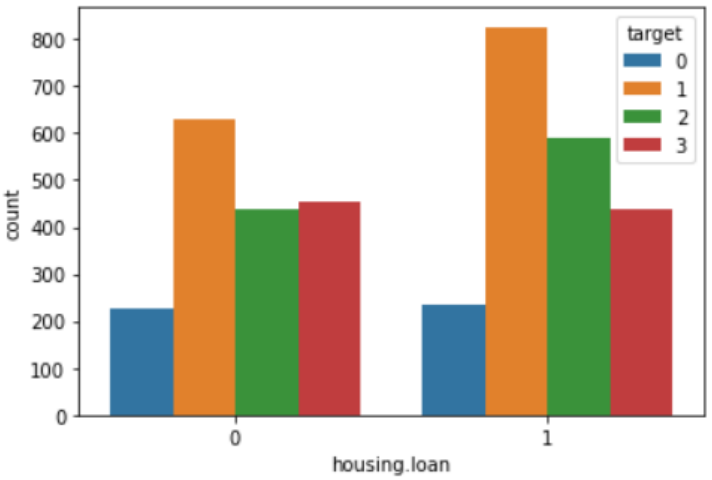

Figura 9: Contagem de clientes que possuem empréstimos para casa por clusters

A Figura 10 mostra a quantidade de clientes que possuem um empréstimo pessoal por cliente. A grande maioria dos clientes não possuem um empréstimo pessoal, especialmente os clientes no cluster 1.

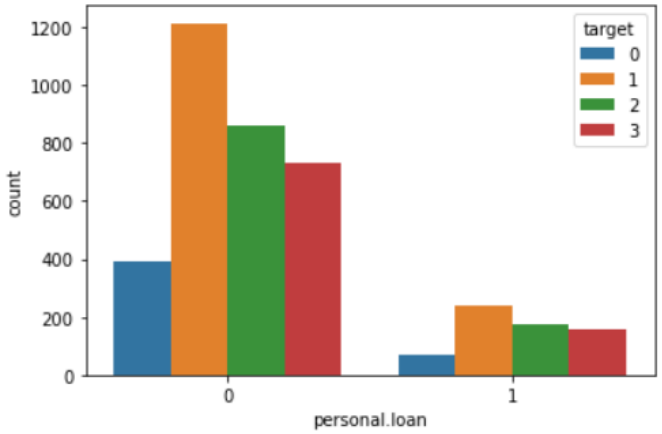

Figura 10: Contagem de clientes que possuem empréstimo pessoal por clusters

A Figura 11 mostra a contagem de usuários que foram contatados por telefone ou celular por clusters. Os clientes que se encontram nos clusters 1 e 2 foram contatos, em sua esmagadora maioria, por celular, enquanto que os clientes classificados nos clusters 0 e 3 foram contatados por telefone. Dado que os clusters 1 e 2 possuem um maior número de clientes que subscreveram o depósito, em uma futura campanha, para esses clientes, talvez seja mais efetivo contatar via celular diretamente.

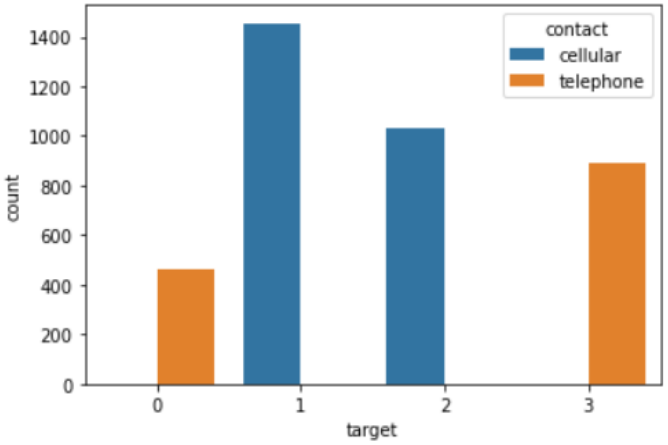

Figura 10: Contagem de clientes que foram contatados através do telefone ou celular por clusters

A Figura 11 mostra a contagem de clientes que foram contatados durante a semana por clusters. Os clientes do cluster 1 e 2 foram contatados, em sua maioria, na Quinta-feira e Segunda-feira. Já os clientes do cluster 2 foram mais contatados na Quinta-feira e Quarta-Feira. Dado que os clusters 1 e 2 possuem clientes com mais chances de subscrever o depósito, seria interessante levar os dias da semana para contatar os clientes em consideração.

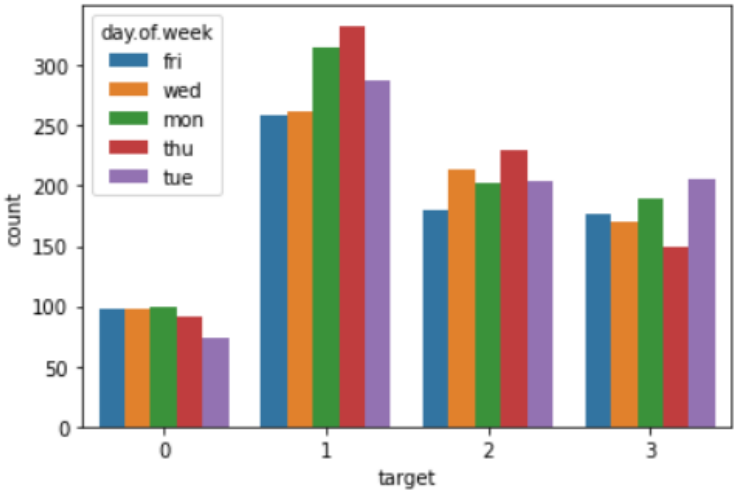

Figura 11: Contagem de clientes que foram contatados durante a semana por clusters

Como podemos observar na Figura 12, a mediana de tempo de duração das ligações é abaixo de 500s (8 minutos, aproximadamente), sendo que os clusters 1 e 2 possuem um maior tempo de ligação se comparado aos outros clusters. Pode ser devido ao fato de que por possuírem mais clientes que subscreveram o depósito, precisaram ficar mais tempo na ligação para finalizar a subscrição. Também podemos observar que os dados possuem ligação de mais 3000s (50 minutos), o que pode ser um erro dos dados, ou, caso não seja um erro, pode significar que alguns clientes tiveram que ficar mais tempo para resolver algum problema, o que poderia ser investigado pela empresa de *telemarketing*.

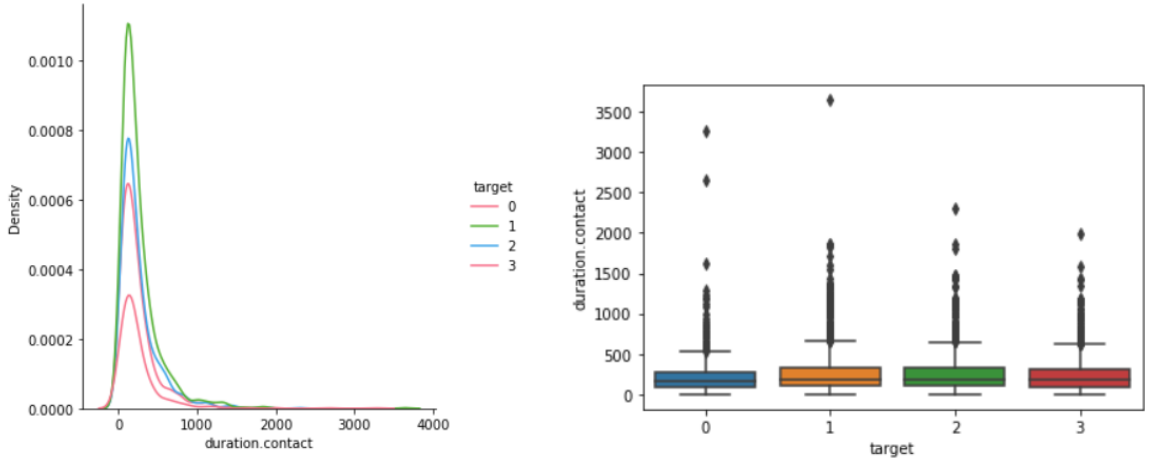

Figura 12: Distribuição da duração da ligação de contato durante a campanha por clusters

A Figura 13 mostra a contagem de usuários relacionado com o número de contatos realizados por cluster. Como podemos observar, a grande maioria dos clientes foram contatados apenas uma vez, principalmente os clientes que se encontram nos clusters 1 e 2.

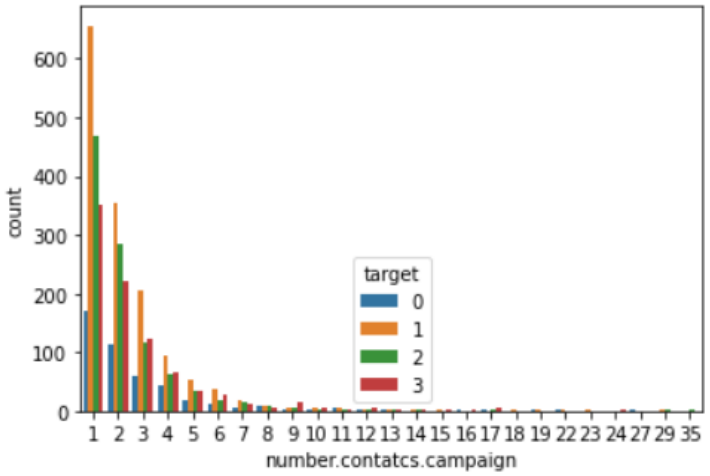

Figura 13: Contagem de clientes e o número de contatos por cluster

Como podemos observar na Figura 14, as campanhas que tiveram sucesso estão concentradas nos clusters 1 e 2, o que já se era esperado dado que os mesmos clusters concentram o maior número de clientes que subscreveram o depósito.

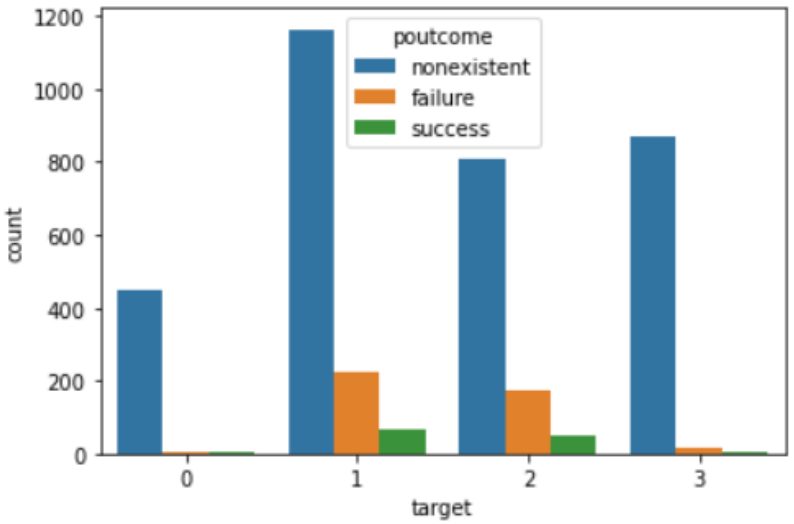

Figura 14: Relação do sucesso da campanha por cluster

A Figura 15 mostra a matriz de relação de pares de atributos por clusters. A última coluna da matriz contém a relação dos clientes que subscreveram ou não o depósito em relação aos outros atributos, separado por cluster. Como já havíamos observado anteriormente, os clientes que subscreveram se encontram em sua maioria nos clusters 1 e 2. Também podemos observar que a relação de empréstimo de casa e empréstimo pessoal não parecem afetar a questão de subscrever ou não. Os clusters 1 e 2 estão espalhados de uma forma que não deixa claro se a relação empréstimo e depósito estão ligados, aparentemente não estão.

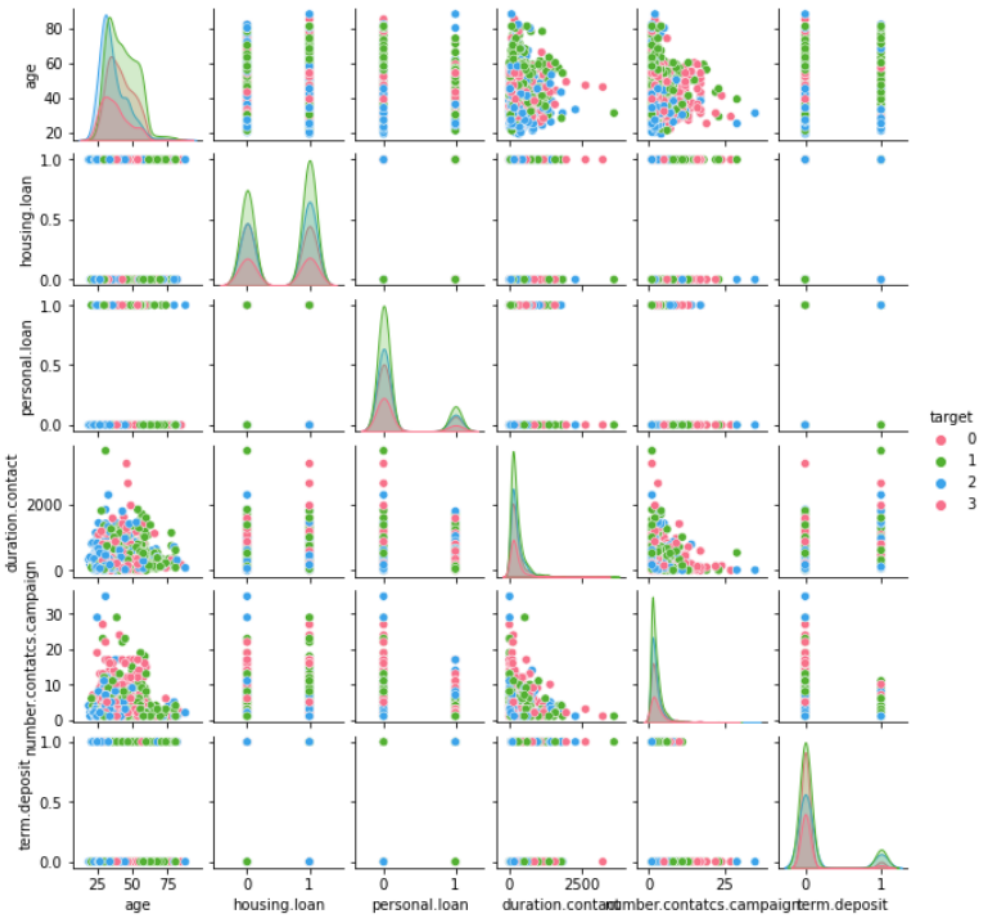

Figura 15: Matriz de relação de pares de atributos por clusters

A Figura 16 mostra a relação do dia da semana com a subscrição do depósito por clusters. Os clientes que subscreveram do cluster 1 estão em sua maioria concentrados na Segunda-feira, enquanto que os clientes do cluster 2 que subscreveram estão mais concentrados na Sexta-feira.

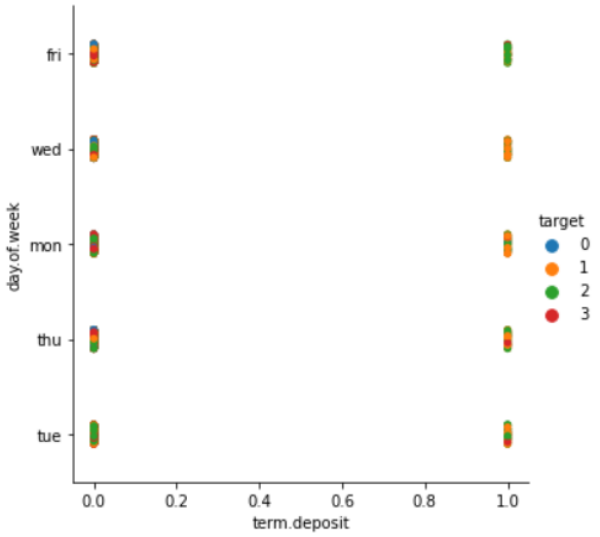

Figura 16: Relação dia da semana com o depósito por clusters

A Figura 17 mostra a quantidade de clientes em cada uma das formações por cluster. Como podemos observar, o cluster 1 possui um maior número de clientes com educação básica e graduação. Já o cluster 2 possui mais clientes com graduação.

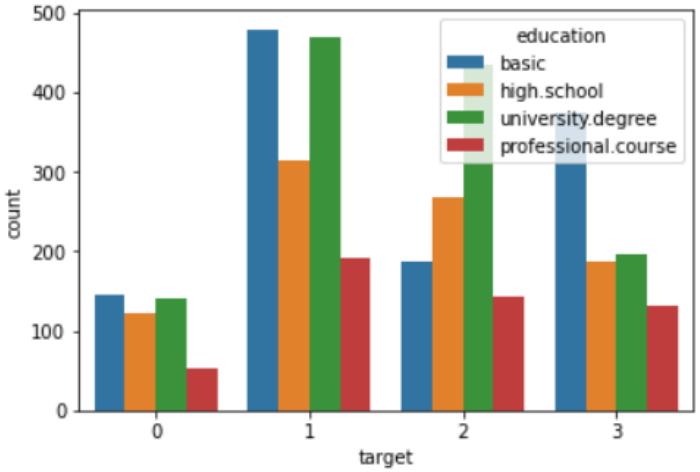

Figura 17: Contagem de clientes por educação por clusters

4. Considerações finais

Após analisar cada um dos clusters gerados e os clientes em cada um dos clusters, podemos afirmar que os clientes que se encontram nos clusters 1 e 2 têm uma maior probabilidade de subscrever o depósito, se comparados aos outros clientes. Em futuras iniciativas de marketing para os mesmos clientes, seria interessante voltar a contatar os clientes dos clusters 1 e 2 observando o melhor dia da semana, o uso do celular ao invés do telefone, a faixa etária dos clientes e possivelmente focando as iniciativas para o público de estado civil casado, solteiro, ou divorciado.

O desenvolvimento do trabalho foi muito importante para o entendimento do funcionamento dos algoritmos por agrupamento para problemas não-supervisionado, mais especificamente o *K-means*, e poder observar na prática que os dados em cada um dos clusters gerados possuem algumas semelhanças entre si, elucidando a importância desses algoritmos para problemas não-supervisionados.

Todos os dois trabalhos desenvolvidos pelo grupo durante o semestre podem ser encontrados no GitHub.