Nanodegree Machine Learning Engineer

Projeto Capstone

Otimizacão de Campanha de Marketing via Clusterização e Regressão

1. Definição

Visão Geral

O objetivo do projeto é aplicar o aprendizado obtido durante o programa de Nanodegree Engenheiro de Machine Learning para solucionar um problema na área do Marketing Digita aplicando algoritmos e técnicas de Machine Learning para Segmentação de Clientes.

Segmentação de e clientes é uma das tarefas mais importantes em qualquer empresa de Marketing os resultados iram influenciar as decisões de marketing e vendas de forma que seja possível oferecer serviços e produtos mais personalizados visando obter maior lucratividade no negócio. O Conceito de segmentação de marketing foi cunhado por Wendell R. Smith, que em seu artigo “Diferenciação de Produto e Segmentação de Mercado como Estratégias de Marketing Alternativas” observou “muitos exemplos de segmentação” em 1956.

O papel da segmentação de clientes no marketing digital é justamente identificar e categorizar esses grupos e em seguida entender as suas características, a partir disso, você pode criar personas, isto é, personagens únicos que representem cada grupo identificado. As personas servirão de base para que que os serviços e produtos sejam desenvolvidos ou direcionados para atender as necessidades e ter melhor desempenho do negócio.

Enunciação do Problema

No Marketing digital, uma atividade muito comum é a utilização de ferramentas de anúncios que realizam campanhas tanto em sites de busca (Google, Bing, etc) ou em redes sociais (Facebook, Instagram, etc) em resumo, esses anúncios aparecem para as pessoas que estão buscando algum serviço em questão. As plataformas fornecem opções de filtros para que esses anúncios tenham um público-alvo especifico, a pergunta feita é: **como saber o perfil do público alvo?** Esse é o problema em questão que esse projeto irá ajudar a resolver.

A métrica mais utilizada pelas empresas de Marketing Digital para mensurar o desempenho dessas campanhas é a **taxa de conversão**, que é gerada pela frequência de todos os clientes clicaram ou acessam um site ou plataforma via links e anúncios, e realizam alguma transação que gera valor ao negócio, como por exemplo, cliente acessa um site de roupas e comprar alguma peça, o cliente acessou o site por alguma fonte e realizou uma ação gerou valor (Lucro) e consequentemente gerou a taxa de conversão.

Outra métrica utilizada é **ROAS** (Retorno Sobre o Investimento Publicitário), que seria a métrica que avalia a viabilidade econômica da campanha, ela é calculada pela razão da receita pelo investimento, não é obrigatório que uma campanha com maior taxa de conversão seja sempre melhor, o ROAS também deve ser levado em consideração, é através do ROAS, a agência tem a visão genuína dos lucros e investimentos de seu cliente se tratando das campanhas publicitárias.

O objetivo é aumentar ROAS identificando os grupos que teriam a maior taxa de conversão.

Para atingir esse objetivo, serram realizados os seguintes passos:

O projeto irá seguir os seguintes passos:

1. **Análise Exploratória:** Entendimentos mais aprofundados dos dados utilizando estatística sumárias.
2. **Visualização Exploratória:** Demonstração e comentários das visualizações das caraterísticas mais relevantes identificadas na analise exploratória.
3. **Algoritmos e Técnicas:** Definição dos algoritmos e técnicas de validação.
4. **Pre-processamento de Dados**: Limpeza, Escalonamento, Tratamento de dados categóricos, Feature Engineering, identificação e tratamento de outliers e Feature Selection.
5. **Aplicação dos Algoritmos:** Utilizar a biblioteca sckitlearn para treinar os modelos.
6. **Ajuste e validação dos Algoritmos:** Melhorar o desempenho do modelo por ajustes e aplicar técnicas de validação.
7. **Conclusão:** Analise final e sugerir plano de ação utilizando os resultados encontrados para diferenciar a campanha de forma otimizada.

Métricas

Para o modelo de clusterização será utilizada validação baseada em densidade, coeficiente de silhueta é calculada com a **média** para todos os pontos utilizando a equação:

$$s = \frac{b - a}{\max(a, b)}$$

s = Coeficiente de silhueta do pronto (vai de -1 até 1)

b = Distância média do ponto e dos outros pontos dos outros clusters.

a = Distância média do ponto e dos pontos do mesmo cluster.

Para o modelo regressão será usado Root Mean Squared Error (RMSE)

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (X_{obs,i} - X_{model,i})^2}{n}}$$

Xobs = Valor Real

Xmodel = Valor estimado pelo modelo

2. Análise

Análise Exploratória

Os dados usados neste projeto são da campanha publicitária de mídia social de uma organização anônima. que foram extraídos do banco de dados do Kaggle.

Os dados se referem a uma da campanha publicitária de mídia social de uma organização anônima. O Dataset contém 1143 observações (Linhas) em 11 variáveis (Colunas). Abaixo estão as descrições das variáveis:

- **ad_id:** ID exclusivo para cada anúncio;
- **xyz_campaign_id:** ID associado a cada campanha publicitária empresa XYZ;
- **fb_campaign_id:** ID associado a campanha como o Facebook rastreia a campanha;
- **age:** Idade da pessoa a quem o anúncio foi mostrado;
- **gender:** Gênero da pessoa a quem o anúncio foi mostrado;
- **interest:** Código que especifica a categoria de qual o interesse da pessoa pertence;
- **Impressions:** Numero de vezes que anúncio foi mostrado;
- **Clicks:** Número de cliques do anuncio;
- **Spent:** Quantidade pago pela empresa XYZ para o Facebook, para mostrar o anúncio;
- **Total conversion:** Número total de pessoas que se interessaram sobre o produto ou serviço depois de ver o anúncio;
- **Approved conversion:** Número total de pessoas que compraram o produto depois de ver o anúncio.

Segue um resumo das informações dos tipos de dados:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1143 entries, 0 to 1142
Data columns (total 11 columns):
ad_id                  1143 non-null int64
xyz_campaign_id        1143 non-null int64
fb_campaign_id         1143 non-null int64
age                    1143 non-null object
gender                 1143 non-null object
interest               1143 non-null int64
Impressions            1143 non-null int64
Clicks                 1143 non-null int64
Spent                  1143 non-null float64
Total_Conversion       1143 non-null int64
Approved_Conversion    1143 non-null int64
dtypes: float64(1), int64(8), object(2)
memory usage: 98.3+ K
```

Figura 1 - Resultado do método do objeto Pandas dataframe.info().

Aparentemente não tem nenhum dado faltando, os tipos de dados deveram ser investigados para validação.

Para melhor intuição dos dados, segue uma amostra das 5 primeiras linhas do conjunto de dados:

	ad_id	xyz_campaign_id	fb_campaign_id	age	gender	interest	Impressions	Clicks	Spent	Total_Conversion	Approved_Conversion	
0	708746	916	103916	30-34	M	15	7350	1	1.43	2	1	
1	708749	916	103917	30-34	M	16	17861	2	1.82	2	0	
2	708771	916	103920	30-34	M	20	693	0	0.00	1	0	
3	708815	916	103928	30-34	M	28	4259	1	1.25	1	0	
4	708818	916	103928	30-34	M	28	4133	1	1.29	1		

Figura 2 - As 5 Primeiras linhas do Dataset, resultado do método do objeto Pandas dataframe.head().

Inspecionando os primeiros elementos, podemos perceber que a colunas `'age'` e `'gender'` são categóricas, apesar que a coluna 'age' ser numérica, ela representa uma categoria de uma faixa de idade, vamos investigar se alguma variável numérica é na verdade uma categoria.

Para Finalizar a nossa exploração básica, vamos analisar as estatísticas sumárias :

	Clicks	Spent	Total_Conversion	Approved_Conversion
count	1143.000000	1143.000000	1143.000000	1143.000000
mean	33.390201	51.360656	2.855643	0.944007
std	56.892438	86.908418	4.483593	1.737708
min	0.000000	0.000000	0.000000	0.000000
25%	1.000000	1.480000	1.000000	0.000000
50%	8.000000	12.370000	1.000000	1.000000
75%	37.500000	60.025000	3.000000	1.000000
max	421.000000	639.949998	60.000000	21.000000

Figura 3 - Resumo dos 5 números gerado pelo método do objeto Pandas dataframe.describe().

Podemos observar que em `'Clicks'`, `'Spent'`, `'Total_Conversion'`, `'Approved_Conversion'` apresentam uma grande diferença entre 75 percentil para o valor máximo, isso significa se ordenarmos nossos dados de forma crescente 75% deles seriam menor que o valor apresentado no linha "75%", no caso de `'Clicks',` seria 37.5 e ultimo valor máximo, o último valor dos dados ordenados, seria 421, é notável que houve um crescimento fora do padrão, considerando que ele saiu de 0 a 35.5 recorrendo 75%, claramente temos outlies, teremos que investigar mais essa variáveis de forma visual.

Visualização Exploratória

Conforme foi observado na Figura 1 no tópico anterior existem algumas colunas deveriam ser categóricas mas estão classificados como numéricas, para confirmar isso, vamos plotar um gráfico de barras mostrando a contagem dos velórios únicos de cada variável.

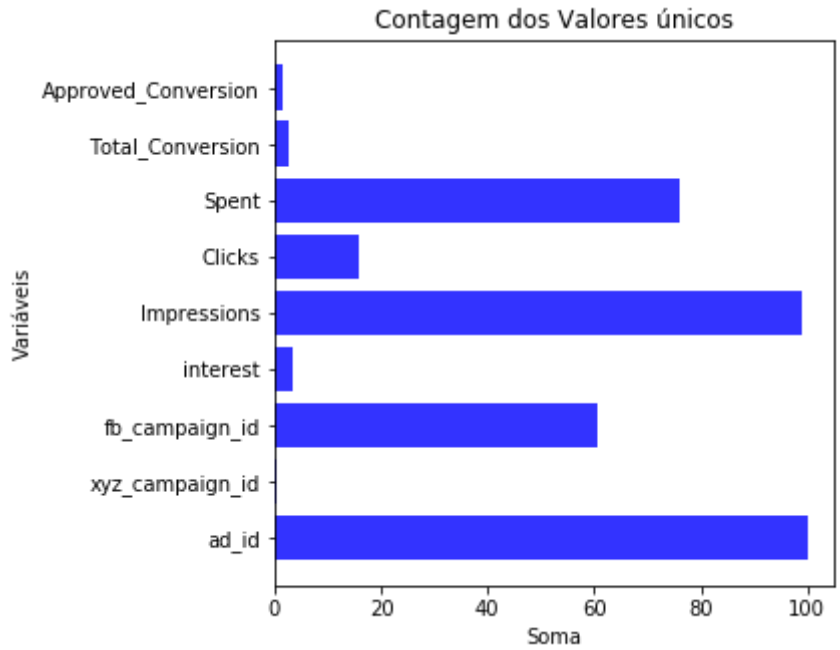


Figura 4 - Contagem dos Valores Únicos de doas as variáveis

A visualização acima indica que `'Total_Conversion'`, `'Approved_Conversion'`, `'interest'`, `'xyz_campaign_id'` podem ser categóricas, segundo o meta dados fornecidos,`'Total_Conversion'` e `'Approved_Conversion'` seriam número total de conversões então seriam realmente dados numéricos, `'interest'`, `'interest'` seriam códigos que especificam as categoria de interesse esses terão que ser alterados e `'xyz_campaign_id'` seria o código da campanha da empresa XYZ também deverá ser alterado. `'ad_id'` tem 100% de valores únicos o que significa que está servido como índice, vamos usar o índice do DafaFrame, estão essa coluna deverá ser removida, Em resumo, **será alterado o tipo das variáveis `'interest'` e `'xyz_campaign_id'` de numérico para categórico e 'ad_id' será removido.**

Seguindo com análise, no data set existem apenas 3 campanhas, é valido entender as proporções de cada uma delas no dataset.

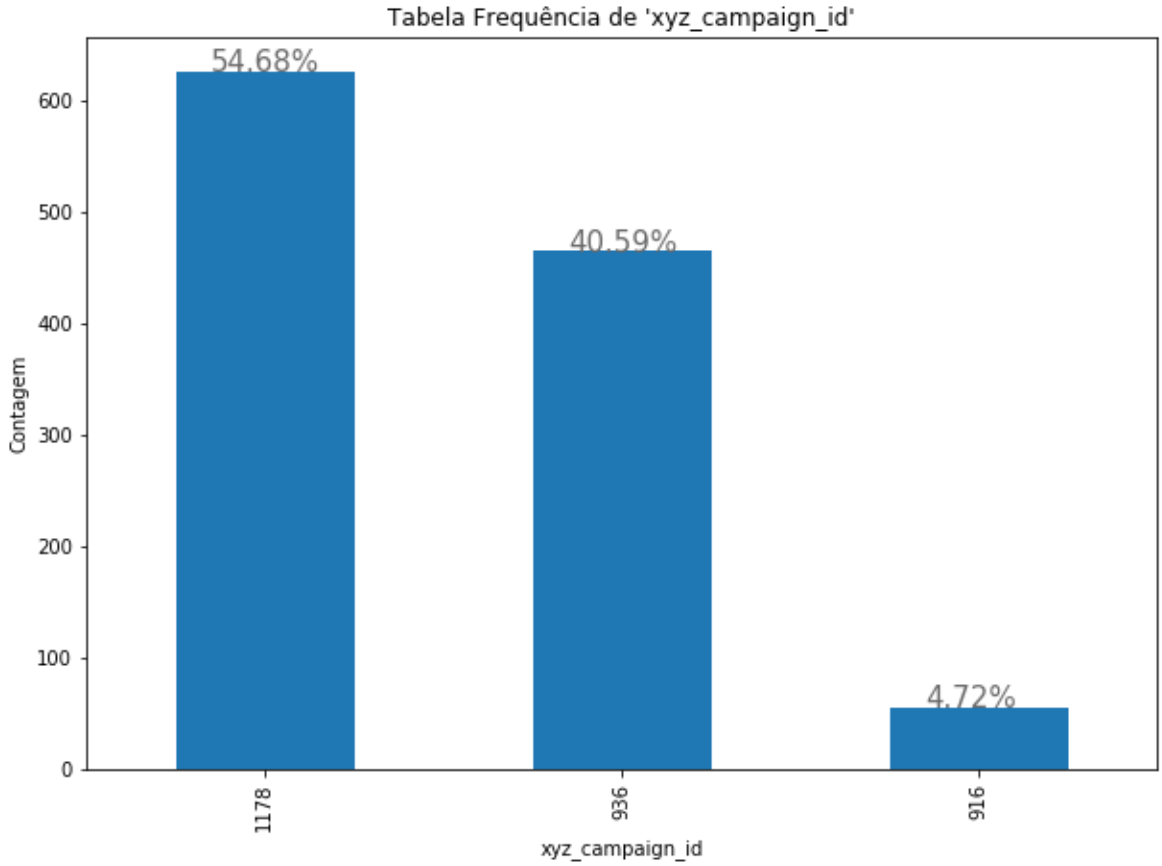


Figura 5 - Tabela Frequência de 'xyz_campaign_id'.

Variável 'xyz_campaign_id' existem apenas 3 valores únicos, o que confirma que pode ser lidado como uma variável categórica, sendo divido entre as campanhas 1178 e 936, a campanha 916 representa apenas 4.72%.

Agora vamos fazer o mesmo com a variável `'interest'`.

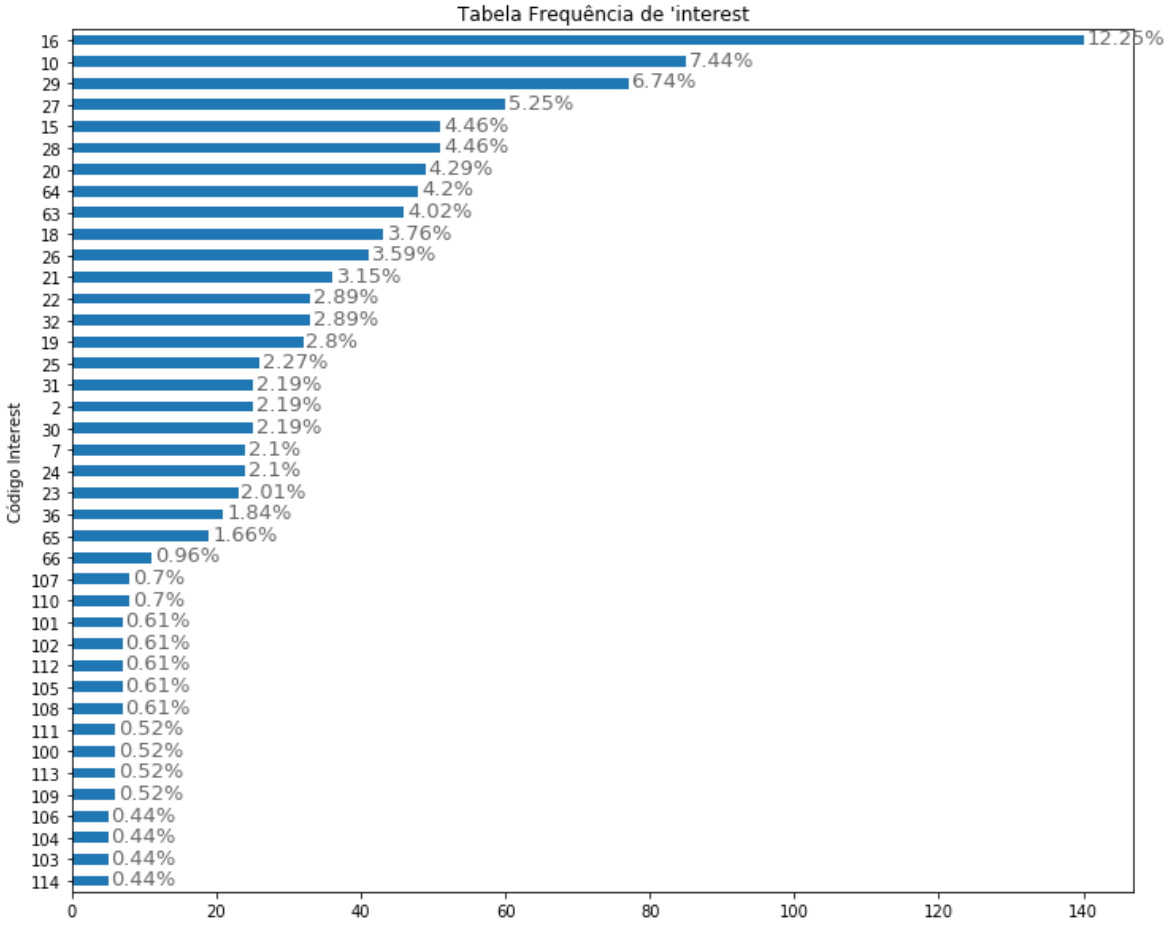
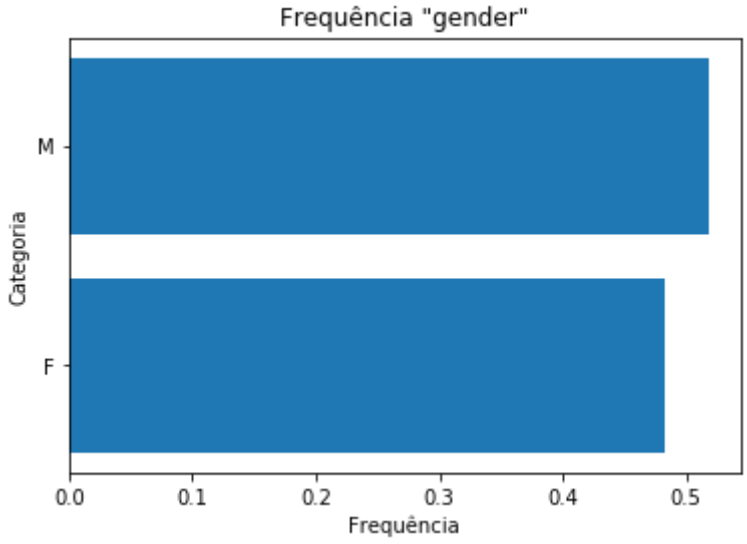
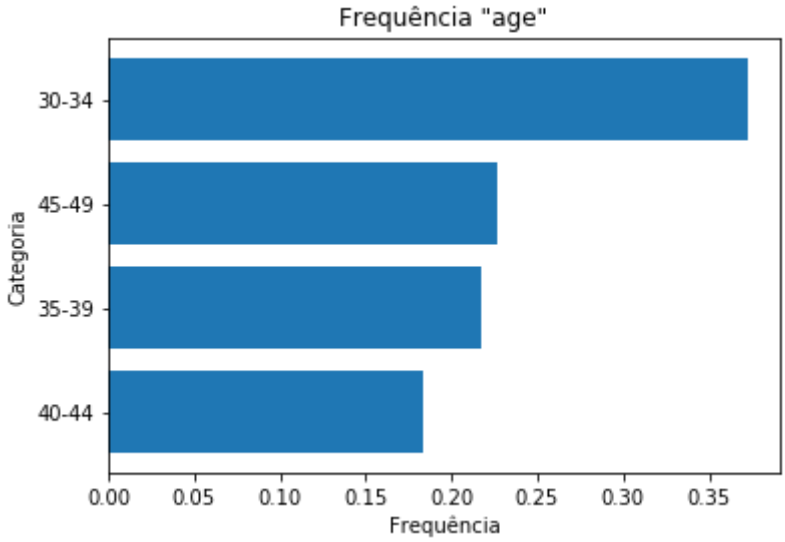


Figura 6 - Tabela Frequência de 'interest'

Apesar de a distribuição sugere que é uma variável numérica, de acordo com os meta dados fornecidos `'interest'` é um código que especifica a categoria de qual o interesse da pessoa pertence, então vamos tratar-lo como uma variável categoria extensa, podemos essa variável como a matriz características dos clusteres.

E por fim, vamos também utilizar gráficos de barra para visualizar as suas frequências das variáveis categóricas `'age'` e `'gender'` que estavam corretas na classificação do tipo de variável.





Figuras 7 e 8 - Tabela Frequência 'gender' e 'age'

Podemos afirmar que pelo menos 80% dados são representados por pessoas de 30 a 39 anos, e é dividido entre homens e mulheres, com a maioridade com idade entre 30 a 34 anos.

Para finalizar a análise de Visualização Exploratória, vamos plotar uma matriz de onde a diagonal principal são histogramas e o restante são gráficos de dispersão, para entendemos tanto a distribuição dos das variáveis numéricas e as suas relações com as outras note que esse matriz não é proveitosa com as variáveis categóricas aonde não apresentam correlação nem uma forma de distribuição definida e em seguida um gráfico de calor parar melhorar a intuição das correlações com as variáveis numéricas.

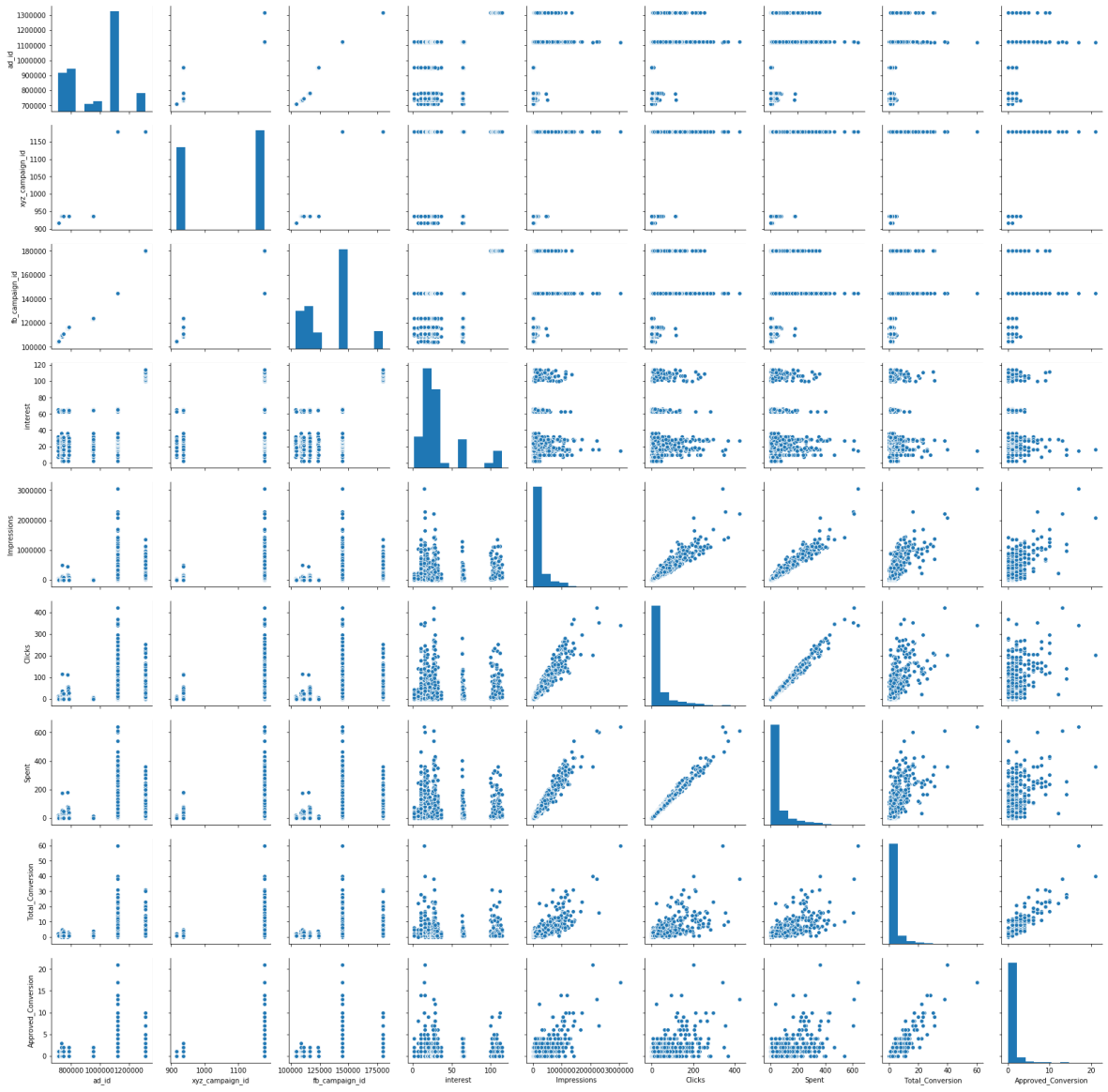


Figura 9 - Matriz de de histogramas de gráficos de dispersão, foi realizada utilizando a biblioteca Python seaborn, a função executa foi searborn.pair_plot()

Agora ficou claro na presença de outlies nas variáveis `'Clicks'`, `'Spent'`, `'Total_Conversion'`, `'Approved_Conversion'` conforme mostra a curva assimétrica direita, confirmando a análise anterior.

É possível confirmar que a variável `'interest'` é categórica por apresentar nenhuma correlação com as demais a a distribuição não definida.

Existem algumas variáveis que são fortemente correlacionadas, isso é notável quando visualizamos os gráficos de dispersão que aparentam ser uma linha, que em outras palavras , indica o quanto a variável explica o comportamento linear da outra, para um problema de aprendizagem não supervisionada isso não é incessante, que implica que as variáveis tem a mesma informação, teremos que criar novas variáveis de forma que diminua essa coração e aumente a informação, em resumo, **será necessário a criação de novas variáveis numéricas para substituir o uso das originais para melhor desempenho do modelo**.

`'fb_campaign_id'` é muito correlacionada com `'ad_id'`, como anteriormente havíamos definido a remoção de `'ad_id'`, será removido também `'fb_campaign_id'`.

Algoritmos e Técnicas

Para casos de segmentação de clientes, normalmente não temos rótulos para treinar um algoritmo de aprendizagem supervisionada, portanto, será utilizado o algoritmo de aprendizagem não supervisionada `K-Prototype`, que é um algoritmo combina a funcionalizada de do clássico algoritmo K-Means e K-modes podendo lidar tanto com vaiáveis numéricas e categóricas sem a necessidade da transformação das variáveis categorias em colunas binárias, para entender melhor como funciona o K-Protoype, existe um artigo do site medium aqui.

O único parâmetro que vamos trabalhar no K-Protorype é o K, que é o numero de centroides (Clusters) que o algoritmo irá dividir dados, para encontrar o valor ótimo de cluster, será treinados de 2 a 6 clusteres medindo métrica de avaliação coeficiente de silhueta, em seguida será plotado um gráfico para visualizar qual o melhor número de clusters.

Como K-Prototpye trata os dados números da mesma forma que do K-Means, medindo a distância entre os pontos, é importante os dados sejam colocados na mesma escala e de forma reduzida, todos os dados serram reduzidos a escala logarítmica com uma tratativa adicional dos dados que tem o valor igual a 0, os mesmo serram substituídos por 0.01 aproximadamente -4 na escala logarítmica.

Em seguida aplicado uma técnica de redução de dimensionalidade chamada PCA que tornará possível a visualização para visualizar os clusters, que é esperado que os 2 principais componentes que representem pelo menos 80% da variabilidade dos dados.

Após os resultados obtidos da clusterização, será treinado regressão linear e Árvore Regressara de forma que podemos avaliar qual cluster influencia mais no aumento do ROAS e quais característica são mais importantes, será utilizando a técnica de validação busca em grade e validação cruzada para evitar que os modelos fiquem sobreajustados (Overfiting).

Modelo de referência

Modelo de benchmark O coeficiente de silhueta varia de -1 a 1, porém não existe um valor benchmark para ser comparado, apenas durante a análise devemos escolher a quantidade de clursters referente ao ponto de inflexão dos score encontrados no coeficiente de silhueta. Para a regressão, podemos utilizar a desvio padrão para ter como referência o RMSE.

3. Metodologia

Pré-processamento dos dados

Para organizar o Pré-processamento dos dados, essa atividade foi dividida 4 em partes.

1 - Alterar o tipo das variáveis 'interest' e 'xyz_campaign_id' de numérico para categórico e remoção 'ad_id' e 'fb_campaign_id'

2 - Criar novas variáveis utilizando 'Clicks' , 'Spent','Impressions' e remover as variáveis numéricas inicias.

3 - Escalonamento e Tratamento de Outlies.

4 - Seleção de atributos (Feature Selection).

Alterar o tipo das variáveis 'interest' e 'xyz_campaign_id' de numérico para categórico e remoção 'ad_id' e 'fb_campaign_id'.

A atividade foi realizada por código utilizando funções do Pandas, o resultado final da alteração pode ser confirmado Abaixo:

```
Lista das variáveis numéricas:
['xyz_campaign_id', 'age', 'gender', 'interest']

Lista das variáveis numéricas:
Index(['Impressions', 'Clicks', 'Spent', 'Total_Conversion',
       'Approved_Conversion'],dtype='object')
```

Criar nova variável utilizando 'Clicks' , 'Spent','Impressions' e remover as variáveis numéricas inicias

Foram criados alguns KPI padrões que são utilizados no marketing digital:

Taxa de cliques "Click-Through-Rate" (CTR): Esta é a porcentagem de quantas impressões se tornaram cliques. Mede a atratividade do anúncio. Um Benchmark para essa KPI seria 2% para ser razoável.

Taxa de Conversão "Conversion-Rate" (CR): Esta é a porcentagem de cliques que resultam em uma "conversão". Conversão é determinado por um objetivo que é definido para a campanha. O que mede efetivamente a efetividade anúncio de campanha. Para dataset do projeto, serão criados 2 taxas CR por temos 2 variáveis distintas de conversão, uma que o objetivo é despertar o interesse e a outra em comprar o produto/serviço.

OBS: Taxa de Conversão será divido em `'TCR'` e `'ACR'` pelo motivo que o dataset tem 2 variáveis distintas de conversão `'Total_Conversion'` e `'Approved_Conversion'`.

No total, as teremos 3 vamos trabalhar com 3 variáveis numéricas,

- **CTR:** Click-Through Rate;
- **ACR:** Aproved Conversion Rate;
- **TCR:** Total Conversion Rate.

Tratamento de Outliers e Escalonamento.

Essa atividade será divida em 3 partes:

- Identificação de outliers: Será utilizado Bloxplot para visualizar esses pontos, lembrando que Boxplot por padrão, identifica como outlier as pontos que estão a 1.5 desvios padrões do IQR(Intervalo inter Quartil).
- Análise: Será avaliado se esses pontos fazem sentido, ver a representatividade dos dados
- Tratamento: Será decidido os pontos serram mantidos, alterados ou removidos nessa ordem de importância, o escalonamento deve acontecer para reduzir também reduzir o efeito de outlies

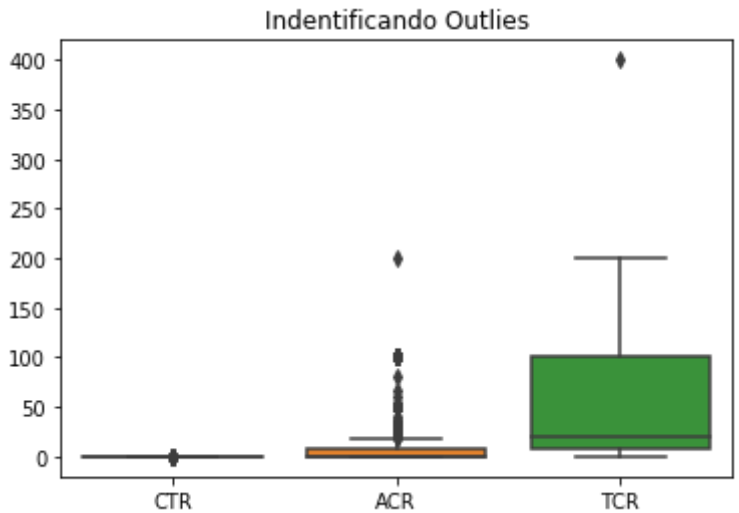


Figura 10 - Boxplot das variáveis numéricas, os pontos que estão fora das caixas são os outliers.

Todas as variáveis aparentam ter outlies, para entender melhor intuição desses pontos, um código foi usado para mensurar a quantidade de proporção dos outliers, segue o resultado:

```
No total, 220 (19.25%) linhas são outliers unicos, sendo 8 (0.7%) em pelo menos
1 variável
```

Existe uma quantidade expressiva de outliers, apenas remover esses pontos iria impactar demais no dataset, além disso, eles aparatam ser dados legítimos, a premissa adotada para interpretar esses pontos foi que algumas campanhas que estão rodando a mais tempo e por isso receberam mais `'impressions'` e `'clicks'`. Esses outliers serão tratados de uma forma matemática, de forma que serão colocados em uma escala logarítmica(mas primeiro é preciso tratar os zeros do data set, na escala logarítmica, a valor zero vale menos infinito), onde irá diminuir a diferença dos valores extremos

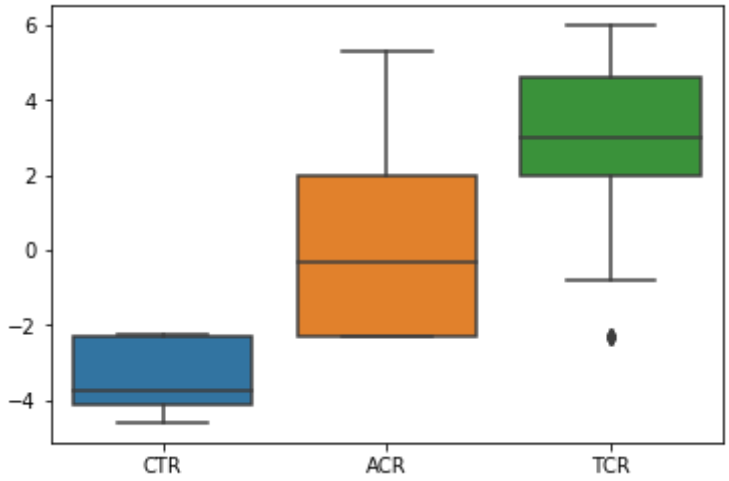


Figura 11 - Boxplot dos dados numéricos após serem colocados na escala logarítmica.

É possível perceber a redução dos pontos fora das caixas, segue o resultado do código que mensura o quantitativo dos outliers

```
No total, 8 (0.7%) linhas são outliers unicos, sendo 0 (0.0%) em pelo menos 1
variável
```

A técnica aplicada foi extremamente eficaz, nossa proporção de outliers caíram de 19.25% para 0.7%, o fato de após ter transformado na escala logarítmica as os valores ainda se apresentam como outlies, eles deveram ser removidos já que 0.7% não apresenta um impacto significativo nos dados.

Seleção de atributos (Feature Selection).

Para melhorar a performance do modelo e evitar que ele possa sofrer erro por viés de uma variável que esteja altamente correlacionada, iremos testar as variáveis numéricas usando a seguinte metodologia:

Vamos remover uma variável numérica e treinar um modelo de regressão para tentar prever utilizando as variáveis restantes, iremos validar o modelo utilizando a métrica R^2 (coeficiente de determinação) que mede o quão bom está o modelo, se o modelo tiver um alto R2 significa que a variável está perfeitamente sendo prevista , o que implica que ela não é relevante para identificar caraterísticas únicas, já que outras variáveis tenham a mesma informação que a mesma.

```
R^2: CTR 0.2643576350294625
R^2: ACR 0.0323099263641875
R^2: TCR 0.28641186728694623
```

Não há nenhuma variável que foi completamente prevista pelas outras, o que significa que o modelo corre menos risco de sofrer erro de viés por alguma variável.

O mesmo modelo de teste será realizado com as variáveis categorias, mas agora utilizando o teste de independência Qui-Quadrado é usado para descobrir se existe uma relação entre as variáveis categóricas, enquanto menor o valor (próximo de zero) siginfica que uma váreial está extremamente relacionada a outra.

	xyz_campaign_id	age	gender	interest
xyz_campaign_id	0.000000	0.002322	2.435071e-04	7.700568e-14
age	2.321753e-03	0.000000	1.187763e-01	9.879450e-01
gender	2.435071e-04	0.118776	6.051966e-248	1.366011e-01
interest	7.700568e-14	0.987945	1.366011e-01	0.000000

Figura 12 - Matriz do p-valor do teste de independência Qui-Quadrado

Avaliando os resultados do teste Qui-Quadrado, pode observar que as variáveis `'xyz_campaign_id'` e `'gender'` tem relação entre todas as variáveis, sendo assim, selas serão removidas para que o modelo não sofra viés.

Implementação

O processo de implementação foi dividido em 2 estágios:

- Clusterização - Kprotoype;
- Regressão - Regressão Linear e Árvore de decisão Aleatória.

Kprotoype

O processo de implantação do modelo de clusterização seguiu as seguintes etapas

1. Transformação dos dados de treino em numpy arrayspara melhor desempenho computacional do algoritmo
2. Loop de treinamento de 2 a 6 clusters calculando o coeficiente de silhueta a cada passo do loop e armazenando o resultado em uma lista
3. Plotagem de um gráfico de linha com as valores do coeficiente de silhueta e número de clusters
4. Treinamento com o número de cluster otimizado via análise do gráfico dos coeficientes de silhueta
5. Aplicação do PCA para redução em 2 dimensões
6. Plotagem dos clusters utilizando os componentes principais do PCA

Regressão Linear e Árvore de Decisão Aleatória.

O processo e implementação dos modelos de regressão seguiram os seguintes passos.

1. Criação de variável dummy dos rótulos obtidos após a clusterização;
2. Criação de um nova variável 'ROAS_Rate' para avaliação dos clusters;
3. Seleção das variável a serem treinadas;
4. Divisão dos dados em variável alvo e variáveis preditoras e transformação em numpy arrays;

5. Divisão dos dados em conjunto de treino e teste;
6. Treinamento do algoritmo de regressão linear LassoCV utilizando o conjunto de treino;
7. Predição utilizando o conjunto de teste, a armazenando os resultados;
8. Impressão das coeficientes regularizados pelo LassoCV';
9. Cálculo da métrica RMSE, correlação e p valor utilizando o conjunto de teste;
10. Criação dos dicionário dos parâmetros a serem testados do algoritmo de Árvore de Decisão Aleatória;
11. Aplicação GridSearchCV utilizando os dicionário de parâmetros, 10 folds de validação cruzada e a métrica de avaliação R^2 (Coeficiente de Determinação).
12. Cálculo da métrica RMSE, correlação e p valor utilizando o conjunto de teste.

Refinamento

O Refinamento do algoritmo de cauterização foi realizado criado um loop de treinamento onde o algoritmo treinou 5 vezes mudando o valor d e K (número de clusters) de 2 a 6, em cada etapa do loop, foi-se calculado o coeficiente de silhueta e armazenado em uma lista, após o fim do loop, é plotado um gráfico de linha utilizando a lista onde foi armazenado o coeficiente de silhueta, com isso é possível identificar qual o melhor número de clusters.

Para o algoritmos de regressão, ambos sofreram uma refinamento automático, o a regressão de LassoCV que utiliza as técnica de regularização e validação que consiste em diminuir a influência das variáveis que não estão agregando valor ao modelo, o algoritmo de Árvore de Decisão Aleatória foi utilizado a técnica de de busca em grade com validação cruzada (GridSearchCV), que encontra a melhor combinação dos parâmetros selecionados avaliando com uma métrica definida, além de prevenir a sobre ajustagem do modelo pela validação cruzada.

4. Resultados

Avaliação e Validação do Modelo

O resultado final do modelo de clusterização obteve o Coeficiente de Silhueta máximo de 0.557 com 3 centroides, segue o gráfico que mostra o valor do Coeficiente de Silhueta ao longo que o número de cluster aumenta.

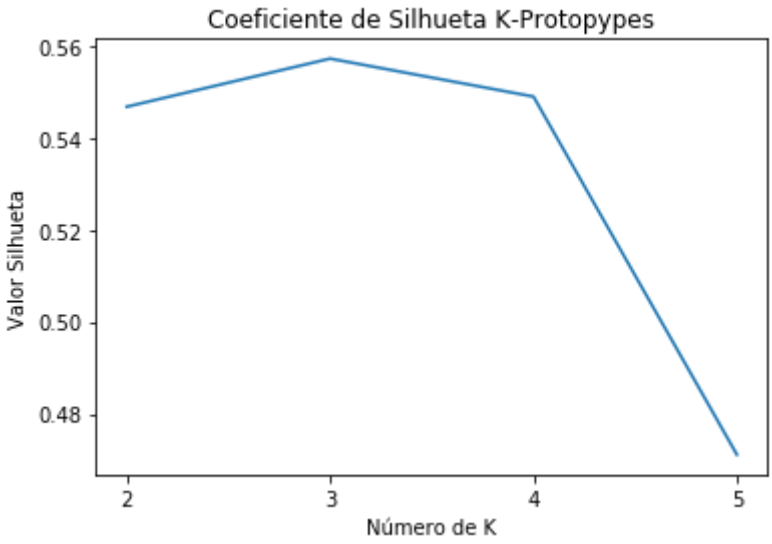


Figura 13 - Coeficientes de Silhueta para os diferentes valores de K

Como é possível perceber, o numero de K=3 teve a maior pontuação de Silhueta, esse valor escolhido para o modelo final

Justificativa

Para entender qual é o melhor o cluster é o mais interessante para o negócio, foi necessário utilizar 2 regressões, o motivo é que a primeira regressão(Regressão Linear) teve um rendimento aceitável RMSE: `11.05` sendo que o desvio padrão na variável alvo é `24.43`, porém a correlação de Pearson não foi muito alto `0.722` o que pode ser expressivo porém não totalmente, Os coeficientes da Regressão Lasso mostrou que o Cluster 0 tem influência positiva e o Cluster 1 tem influência negativa em predizer o `ROAS_Rate` conforme mostra o output:

```
training_columns = ["Spent","Approved_Conversion","ACR","CTR","TCR",0,1]

print(clf.coef_)
[-6.85658999e-03  3.01566506e-01  2.85344718e-01 -2.35421254e+02
  1.00470829e-01  1.18376643e+01 -1.10052194e+00]
```

Os resultados na Árvore de Decisão Repressora foram mais interessantes, o com RMSE menor de `2.23` e correlação de Pearson `0.99` o modelo mostra muito mais preciso, outra informação adicional foi quando foi avaliado a importância das variáveis :

```
training_columns = ["Spent","Approved_Conversion","ACR","CTR","TCR",0,1]

[0.21229948 0.12958379 0.34712678 0.08143894 0.10262069 0.06863613
 0.0582942 ]
```

Lembrando com esses valores são em proporções, o resulto mostra que que o algoritmo não leva em consideração o cluster , mas sim as outras variáveis, como por exemplo a variável `"ACR"` é a mais importante. Com essas informações é possível agora avaliar qual o cluster tem o melhor valor para o negocio (ROAS_Rate).

Sabendo que na regressão de lasso, o cluster 0 tem um coeficiente positivo contra negativo do cluster 1 e na Árvore de decisão aleatória os mais importantes são `"ACR"` fica claro que o cluster que tem maior valor é o Cluster 0.

5. Conclusão

Visualização Livre

O objeto da técnica de redução e dimensionalidade PCA era transformar nossos dados em 2 dimensões para que seja possível a visualização dos clusters, levando em consideração as boas práticas, onde o PC1 e PC2 (Primeiro componente e segundo componente) devem representar pelo menos 80% da variabilidade dos dados, junto a isso, segue gráficos que descreem quantitativamente cada cluster mostrando quanto a diferença me média dos clusteres em relação a média total do Dataset.

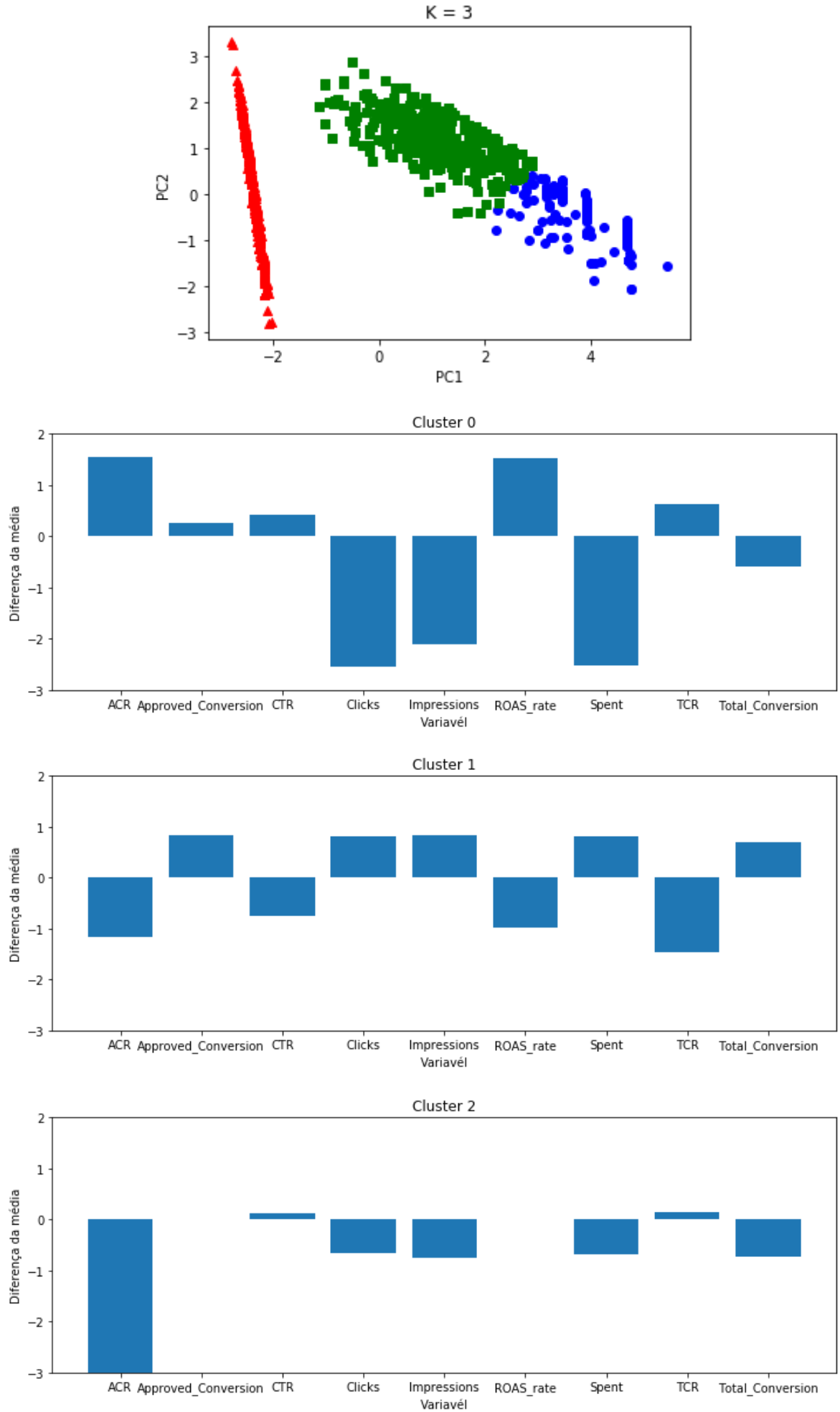


Figura 14,15,16 e 17 - Resultado da separação do modelo e característica quantitativas em relação a média

É possível perceber que o cluster 2 tem uma grande disparidade com a média geral em relação ao *'ACR'* e fica muito próximo da média nas outras variáveis, fica claro que esse grupo não tem nenhuma `'Approved_Conversion'`. No cluster 0 e 1, temos `'Approved_Conversion'` sendo o 1 com valores muito acima da média porém com custo maior e `ACR`, podendo ser interpretado casos pouco eficiência já que se precisou de muito mais 'clicks' e consequentemente mais `'Spent'`, o cluster 0 temos bem menos de média em `'Approved_Conversion'` porém ao considerável menor custo, indicando uma boa eficiência O ROAS do cluster 0 e 1 estão opostos (acima e abaixo da média), inicialmente podemos acreditar que o cluster 0 é o mais indicado para receber investimento já que mostra que o ROAS acima da média.

Reflexão

De uma forma geral, o projeto seguiu os seguinte passos

1. Entendimento do problema
2. Aquisição de dados
3. Exploração de Processamento
4. Modelagem
5. Interpretação dos dados

Os passos que demandaram mais tempo foram o 3 e o 4, Durante o Processamento eu tive que estudar bastante sobre Marketing digital para realmente entender melhor como as variáveis funcionavam que era inviável analisar utilizando as variáveis nativas, o que fez eu fazer vários testes com várias combinações de Feature Engineering, isso me fez entender que o entendimento do negócio no qual você está analisando representa até mais do que o conhecimento de técnicas de Machine Learning.

Durante a modelagem, eu tive que me aprofundar bastante nos algoritmos de clusterização e descobrir o quanto ele é sensível a outliers e escala de dados, o que me sempre voltar no processamento para ajustar e ter um resultado confiável, não encontrei muitas técnicas para validar a robustez do método de aprendizagem não supervisionada, já que aprendizagem supervisionada utilizei as boas práticas para garantir a robustez dos modelos.

No final do projeto, me fez acreditar que marketing digital não é tão simples analisar, já que estamos lidando com dados que são gerados do comportamentos de clientes, entender quais e como as variáveis antes de começar algum projeto que envolva Machine Learning é algo que também requer conhecimento do negócio aliada ao conhecimento técnico de Data Science.

Aperfeiçoamento

As variáveis categóricas não se mostram muito uteis por terem sido representadas em números, por conta disso não foi extensivamente inclusa nas analises e modelagem, as variáveis que se referiam a idade e sexo estavam muito balanceadas e não foi possível encontrar nenhuma tendência que fosse relevante, para melhorar o modelo, a coluna `"Interest"` deveriam ter menos casos únicos e qualitativos, de forma que poderíamos processar e gerar características mais humanas em cada cluster, dessa forma o resultado final não iria mostrar apenas como melhor investir nas campanhas publicitárias, mas sim quais são as características chave que podem agregar mais mais valor ao negócio.