Clusterização de Vídeos do Canal Formiga Atômica

Autor: Lucas Perondi Kist (RA: 236202)

1 Introdução

O YouTube é uma plataforma de compartilhamento de vídeos criada em 2005. Desde então, inúmeras pessoas têm criado canais nela, nos quais publicam materiais produzidos. O site conta com a opção de organizá-los em playlists, que correspondem a conjuntos com características, em princípio, semelhantes. Nesse sentido, surge a possibilidade de fazer uma clusterização desses vídeos, com base em características relacionadas a popularidade e qualidade do som, que podem servir como uma segunda abordagem para agrupá-los.

A partir dessa ótica, foram coletados dados de 111 vídeos do canal Formiga Atômica (Formigari (2014)), o qual é propriedade de João Pedro de Campos Formigari, aluno do curso de Estatística que autorizou o uso das informações para a elaboração deste artigo. Para isso, foi utilizada uma API do YouTube a partir do R (via pacote *tuber*), bem como *web scraping* para extração dos áudios.

As características utilizadas para a clusterização foram: número de dias desde a publicação do primeiro vídeo (Dias), número de visualizações (*Views*) e curtidas (*Likes*), duração em segundos (Duração), valor médio da onda de áudio do canal da esquerda (Média_l), bem como seus quantis 0,05, 0,10, ..., 0,90, 0,95, totalizando 24 atributos. A atualização dos dados ocorreu no dia 14/04/2024 e os *scripts* que podem ser utilizados para a reprodução deles podem ser encontrada em `https://github.com/lpkist/Trabalho1_ME921`.

Nesse sentido, o objeto deste trabalho é clusterizar os vídeos em grupos que, idealmente, possam ser identificados como adequados para objetivos específicos. Por exemplo, um grupo de vídeos (*cluster*) pode ser ideal para a inserção de propagandas por ter alto número de visualizações, enquanto outro pode estar relacionado a altos ruídos, o que pode estar associado a uma menor popularidade e, portanto, deve ser evitado, entre outras interpretações possíveis.

2 Materiais e Métodos

As análises foram realizadas nas linguagens de programação R (R Core Team (2018)) e Python (Van Rossum e Drake (2009)), sendo que a última foi utilizada apenas para extração dos áudios. Inicialmente, foram obtidas as informações dos vídeos através do pacote *tuber* (SOod (2020)), incluindo os *links*. A partir deles e utilizando os pacotes *pandas* e *selenium*, foi realizada a conversão e *download* dos vídeos em formato MP3, com exceção do vídeo "Transmissão ao vivo de Formiga Atômica", que não estava disponível.

Na sequência, eles foram lidos via pacote *tuneR* e foram extraídas a duração (em segundos), a média e os quantis 0,05, 0,10, ..., 0,95 das ondas sonoras do canal da esquerda de cada um deles. Como era de interesse utilizar o ruído na *clusterização*, foi realizado PCA desses quantis (em geral, quantis inferiores são negativos e os superiores, positivos). Finalmente, as unidades dos dados foram alteradas de forma que tivessem desvios-padrão semelhantes, com pequenas diferenças que refletiam a importância de cada atributo.

Com os dados estruturados, foram realizadas análises para definir o número adequado de *clusters* a serem utilizados a partir dos pacotes *cluster* (Maechler et al. (2022)) e *factoextra*. Por fim, foi aplicado o algoritmo PAM e os resultados foram reportados, com a visualização sendo realizada via multidimensional scaling clássico.

2.1 Partição em Medoides (PAM)

PAM (em português, Partição em Medoides) é um algoritmo que agrupa as observações em *clusters* a partir de k elementos centrais (medoides). Formalmente, conforme apresentado por Van der Laan et al. (2003), sejam $\boldsymbol{X}_{n\times p}$ a matriz de dados, com n observações e p características,

$\boldsymbol{D} = (d(i,j))_{n\times n} = (d(\boldsymbol{x}_i, \boldsymbol{x}_j))_{n\times n}$ a matriz de dissimilaridades, simétrica, com $d(\boldsymbol{x}_i, \boldsymbol{x}_j)$ sendo a dissimilaridade entre as observações i e j, e $\boldsymbol{M} = (M_1, M_2, ..., M_k)$ qualquer coleção de k elementos de $\boldsymbol{X}$.

Dessa forma, sendo $\boldsymbol{M}$ conhecido, pode-se calcular $d(\boldsymbol{x}_i, M_K)$ para cada $M_K \in \boldsymbol{M}$ e $i = 1, 2, ..., n$. Definindo $\min_{K=1,2,...,k} d(\boldsymbol{x}_i, M_K) = d_1(\boldsymbol{x}_i, \boldsymbol{M})$ e $\min^{-1}_{K=1,2,...,k} d(\boldsymbol{x}_i, M_K) = l_1(\boldsymbol{x}_i, \boldsymbol{M})$, respectivamente, o mínimo e o minimizador de $d(\boldsymbol{x}_i, M_K)$, são selecionados os medoides $\boldsymbol{M^*} = \min^{-1}_{M} \sum_{i=1}^{n} d_1(\boldsymbol{x}_i, \boldsymbol{M}$ Assim, cada medoide M^*_K identifica um grupo, definido como todos os elementos que estão mais próximos a ele do que de qualquer outro medoide. Tal agrupamento é registrado nas etiquetas $l(\boldsymbol{X}, \boldsymbol{M}^*) = (l_1(\boldsymbol{x_1}, \boldsymbol{M}^*), ..., l_1(\boldsymbol{x_n}, \boldsymbol{M}^*))$.

Para avaliar o agrupamento realizado, podem ser utilizadas as silhuetas. Para cada $\boldsymbol{x_i}$, $i = 1, ..., n$, sejam A e B, respectivamente, o *cluster* de $\boldsymbol{x_i}$ e o grupo com menor dissimilaridades média dos elementos em relação a $\boldsymbol{x_i}$ (excluindo A). Além disso, definindo $a(\boldsymbol{x_i})$ e $b(\boldsymbol{x_i})$ como sendo a média da dissimilaridade de $\boldsymbol{x_i}$ em relação a todos os elementos de A e B, respectivamente, a silhueta de $\boldsymbol{x_i}$ é $s(\boldsymbol{x_i})$, definida como sendo $s(\boldsymbol{x_i}) = \frac{b(\boldsymbol{x_i}) - a(\boldsymbol{x_i})}{\max\{a(\boldsymbol{x_i}), b(\boldsymbol{x_i})\}}$. Ademais, pode-se calcular a média das silhuetas, $\overline{s}(k) = \sum_{i=1}^{n} s(\boldsymbol{x_i})/n$, e adotar o valor de k que a maximiza como número adequado de $clusters$.

2.2 Análise de Componentes Principais (PCA)

A Análise de Componentes Principais (do inglês, PCA), é uma técnica de análise multivariada que objetiva realizar uma projeção ortogonal da matriz de dados reais em um espaço de dimensão menor, mas maximizando a variabilidade representada. Segundo Izenman (2008), seja $\boldsymbol{S} = n^{-1}(\boldsymbol{X} - \overline{\boldsymbol{X}})^T(\boldsymbol{X} - \overline{\boldsymbol{X}})$ a matriz de covariância estimada das colunas de $\boldsymbol{X}$, onde $\overline{\boldsymbol{X}}$ é uma matriz em que a i-ésima coluna corresponde à média da i-ésima coluna de $\boldsymbol{X}$, cujos autovalores e autovetores são, respectivamente, $\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq ... \geq \hat{\lambda}_p \geq 0$ e $(\hat{v}_1, \hat{v}_2, ..., \hat{v}_p)$, onde $\hat{\lambda}_i$ é associado a $\hat{v}_i$, $i = 1, 2, ..., p$.

Então, a melhor reconstrução de $\boldsymbol{X}$ com posto $t < p$ é $\hat{\boldsymbol{X}}^{(t)} = \overline{\boldsymbol{X}} + \sum_{i=1}^{t} \hat{v}_i \hat{v}_i^T(\boldsymbol{X} - \overline{\boldsymbol{X}})$, o escore da j-ésima componente principal de $\boldsymbol{X}$ é estimado por $\hat{\psi}_j = \hat{v}_j^T(\boldsymbol{X} - \overline{\boldsymbol{X}})$ e a variância da j-ésima componente principal de $\boldsymbol{X}$ é estimada por $\hat{\lambda}_j$. Uma medida da qualidade da projeção, em termos de variação, é dada pela proporção da variabilidade representada pelas t primeiras componentes principais amostrais, isto é, $\frac{\sum_{j=1}^{t} \hat{\lambda}_j}{\sum_{j=1}^{p} \hat{\lambda}_j}$.

2.3 Multidimensional scaling clássico

Multidimensional scaling clássico é uma técnica utilizada para reduzir a dimensão dos dados ao realizar uma projeção que busca preservar a distância entre as observações originais. Formalmente, conforme apresentado por Trevor F. Cox (2001), o centroide é definido como a origem e a matriz de produtos internos $[\boldsymbol{B}]_{rs} = b_{rs}$ é dada por $b_{rs} = a_{rs} - a_{r.} - a_{.s} + a_{..}$, com $a_{rs} = -\frac{1}{2}d^2_{rs} = -\frac{1}{2}d^2(\boldsymbol{x}_r, \boldsymbol{x}_s)$, $a_{r.} = n^{-1}\sum_{s=1}^{n} a_{rs}$, $a_{.s} = n^{-1}\sum_{r=1}^{n} a_{rs}$ e $a_{..} = n^{-2}\sum_{r=1}^{n}\sum_{s=1}^{n} a_{rs}$.

Dessa forma, se $[\boldsymbol{A}] = a_{rs}$, então $\boldsymbol{B} = (\boldsymbol{I} - n^{-1}\boldsymbol{1}\boldsymbol{1}^{\boldsymbol{T}})\boldsymbol{A}(\boldsymbol{I} - n^{-1}\boldsymbol{1}\boldsymbol{1}^{\boldsymbol{T}})$. Sendo a decomposição espectral de $\boldsymbol{B}$ dada por $\boldsymbol{B} = \boldsymbol{V}\Lambda\boldsymbol{V}^T$, $\boldsymbol{V} = [\boldsymbol{v}_1, ..., \boldsymbol{v}_p]$ e $\Lambda = diag(\lambda_1, ..., \lambda_p)$, a projeção de $\boldsymbol{X}$ em um espaço $p' < p$ dimensional é dada por $\boldsymbol{X}^* = \boldsymbol{V}^*\Lambda^*$, onde $\boldsymbol{V^*} = [\boldsymbol{v_1}, ..., \boldsymbol{v}]$ e $\Lambda^* = diag(\lambda_1, ..., \lambda_{p'})$.

3 Resultados

Inicialmente, foi aplicado PCA aos quantis das ondas sonoras do canal da esquerda dos vídeos. Os dois primeiros autovetores são $\lambda_1 = 29149234$ e $\lambda_2 = 679842$, representando, respectivamente, 97,4% e 2,3% da variação total. A Figura 1 apresenta o *biplot* das duas primeiras componentes principais.

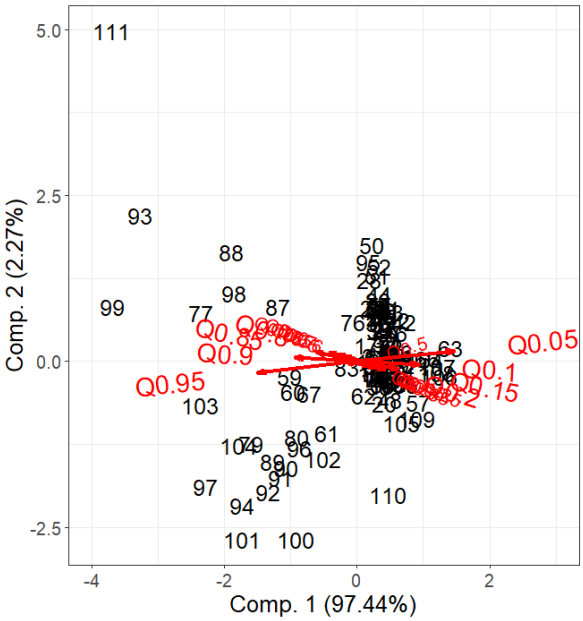

Figura 1: Biplot das duas primeiras componentes principais

A Tabela 1 apresenta o mínimo, quartis, média, máximo e desvio-padrão (DP) das variáveis utilizadas na clusterização, onde CP1_l corresponde à primeira componente principal dos quantis. Já as transformações aplicadas às colunas podem ser encontradas na Tabela 2, assim como os novos DPs.

Tabela 1: Medidas-resumo das variáveis utilizadas

	Dias	*Views*	*Likes*	Duração	Média_l	CP1_l
Mínimo	0,0	4,00	2,00	10,08	-45,77225	-19944,8
1º Quartil	70,5	21,00	3,00	305,32	-0,02444	662,4
Mediana	96,0	35,00	4,00	482,76	0,00433	1845,2
Média	194,0	56,32	4,82	560,86	-0,82670	0,0
3º Quartil	295,5	70,50	6,00	576,37	0,03264	2743,2
Máximo	1536,0	311,00	21,00	4664,78	32,58934	7607,8
DP	194,18	61,76	3,31	573,24	7,70	5423,49

Tabela 2: Transformações aplicadas a cada coluna e novos desvios-padrão

	Dias	*Views*	*Likes*	Duração	Média_l	CP1_l
$f(x) =$	$x/210$	$x/3$	$x/55$	$x/580$	$x/7,5$	$x/5300$
Novo DP	0,92	1,12	1,10	0,99	1,03	1,02

As distâncias euclidianas entre os vídeos, calculadas a partir das colunas transformadas, são apresentadas na Figura 2. Nela, cores mais claras correspondem a distâncias maiores. Já a Figura 3 ilustra o tamanho médio da silhueta das observações após aplicar o PAM para alguns valores de k, estando destacado o número de *clusters* que maximiza esse valor. Além disso, as silhuetas das observações para esse valor de k estão representados na Figura 4, cuja média é de 0,51, indicando estrutura de agrupamentos razoável.

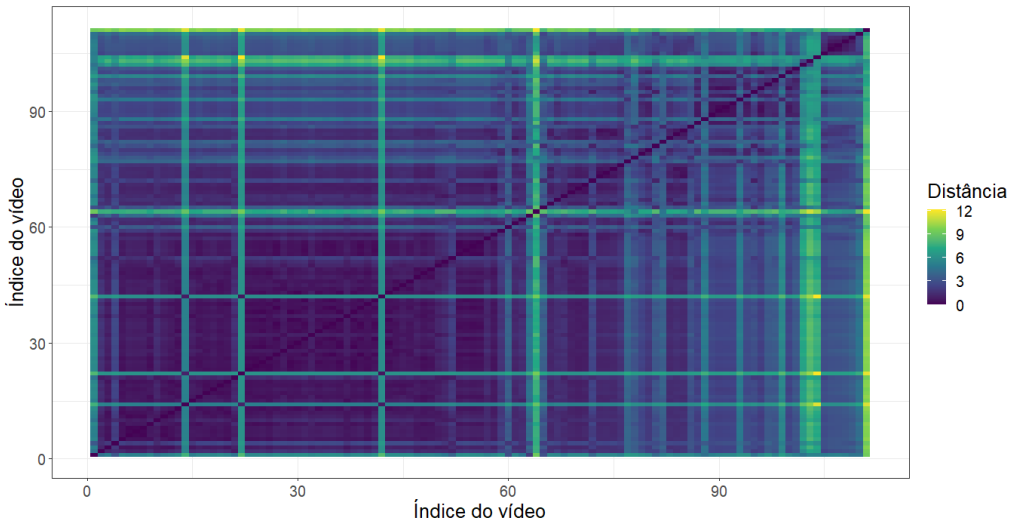

Figura 2: Distância euclidiana entre os vídeos

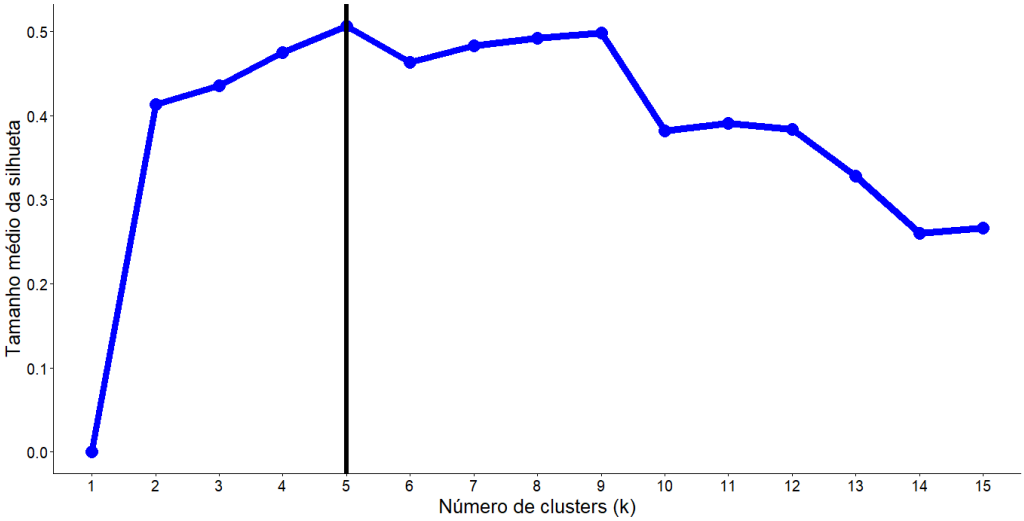

Figura 3: Comprimento médio da silhueta para alguns valores de k

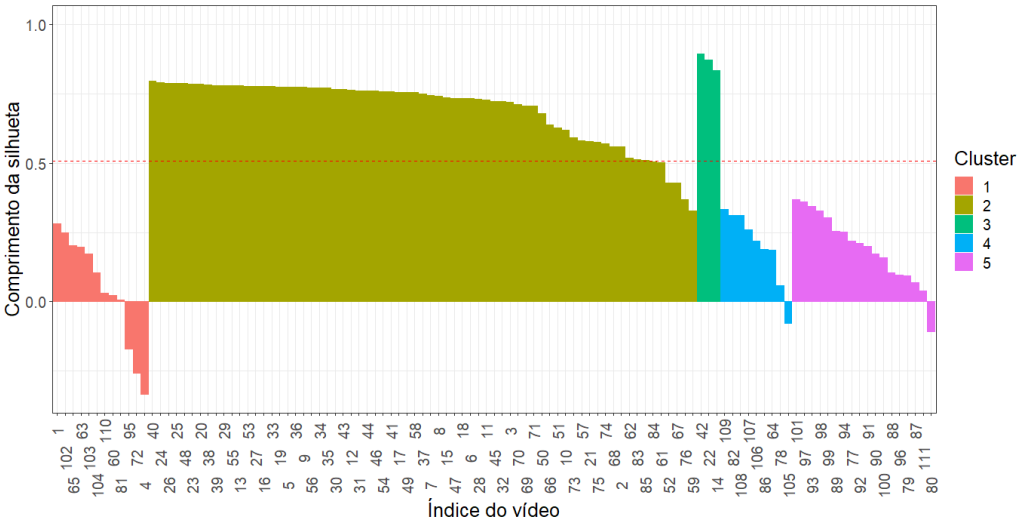

Figura 4: Silhuetas de cada observação após aplicar o PAM com o melhor k

A Figura 5 ilustra a aplicação do multidimensional scaling clássico aos dados, com a cor correspondendo ao *cluster* a que cada observação pertence. Além disso, os tamanhos dos *clusters* 1,

2, 3, 4 e 5 são, respectivamente, 12, 69, 3, 9 e 18. A Tabela 3 apresenta os atributos originais de

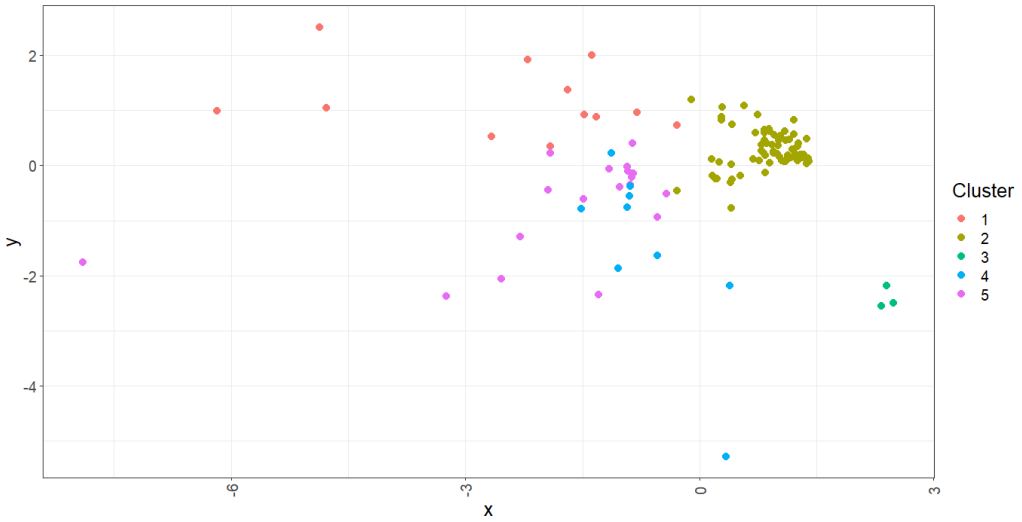

Figura 5: Multidimensional scaling clássico aplicado aos dados transformados

cada uma das cinco observações selecionadas como medoides após a aplicação do algoritmo PAM utilizando $k = 5$. Os títulos dos vídeos dos medoides são, respectivamente: “Vídeo zueira! Vejam a descrição”, “Star Wars: The Force Unleashed II! Ep 18”, “3 Star Wars: The Force Unleashed II! Ep 20”, “Minecraft! Sky wars Ep 2”e “Minecraft! Desafios de um amigo Ep 1”(o mais recente).

Tabela 3: Atributos originais dos medoides selecionados pelo algoritmo PAM

Medoide	Dias	*Views*	*Likes*	Duração	Média_l	CP1_l
1	202,00	162,00	10,00	25,03	0,34	2078,59
2	82,00	26,00	3,00	394,34	-0,08	2203,09
3	84,00	8,00	3,00	547,53	-43,92	3816,10
4	509,00	87,00	7,00	1186,82	-0,00	6469,68
5	343,00	62,00	6,00	315,07	0,21	-6868,83

4 Discussão

A partir do *biplot* apresentado na Figura 1, nota-se que é suficiente utilizar apenas a primeira componente principal, que representa 97,4% da variabilidade total dos dados. Além disso, a partir dos tamanhos e direções das setas, percebe-se que ela é uma média ponderada dos valores absolutos dos quatro quantis inferiores (valores negativos) e dos quatro superiores (valores positivos). Dessa forma, conclui-se que ela representa o ruído dos vídeos, assim como era desejado.

Em relação à Tabela 1, é nítido que os atributos variam em escalas bastante diferentes. Assim, as variáveis foram transformadas, conforme apresentado na Tabela 2, de modo que tivessem desvios-padrão semelhantes, mas dando mais importância a variações em *Views* e *Likes* (associados à popularidade) do que em Dias para o cálculo da distância euclidiana. Isso ocorreu porque foi considerado que as duas primeiras são mais relevantes para caracterização de um vídeo nos moldes estabelecidos.

Da Figura 2, é visível a existência de um *cluster* maior com as primeiras observações (até em torno do índice 75), outro com as últimas (próximas do índice 108) e alguns intermediários. Além disso, vale destacar que algumas observações estão distantes de todas, o que indica que podem ser vídeos atípicos. O tamanho médio das silhuetas após a aplicação do PAM, método mais robusto a valores atípicos do que o K-means, para alguns valores de k, ilustrado na Figura 3 revela que o número ideal de *clusters* a serem utilizados é $k = 5$.

Dessa forma, foi ajustado o algoritmo usando esse valor de k, resultando no gráfico de silhuetas apresentado na Figura 4. A partir de sua análise, conclui-se que há uma estrutura de agrupamentos bastante razoável, com o *cluster* 2 sendo muito maior do que os outros e o 3 sendo bem menor do

que os demais. Apesar disso, alguns elementos dos demais possuem silhueta negativa, indicando possível falta de ajuste ou ocorrência de vídeos atípicos nesses casos.

A visualização dos agrupamentos produzidos em uma dimensão menor, mas com a tentativa de preservar as distâncias, pode ser encontrada na Figura 5. A partir dela, percebe-se que o *cluster* 3 é formado por três vídeos distantes dos demais, enquanto há uma aparente mistura dos de número 1, 4 e 5 (o que justifica a observação da Figura 4, além de alguns valores atípicos.

A Tabela 3 apresenta os atributos dos cinco medoides selecionados. A partir dela, nota-se que o primeiro agrupamento é formado por vídeos mais curtos, mas com altas *Views* e *Likes*; o terceiro, por vídeos com Média_l, *Views* e *Likes* muito menores do que os demais; o quarto, por vídeos recentes, longos, populares e com muito ruído positivos; o quinto, por vídeos com muito ruído negativo; e o segundo, por elementos intermediários.

5 Conclusão

Com base nas análises apresentadas, pode-se concluir que há uma estrutura de *clusters* razoável nos dados, os quais parecem estar divididos em cinco agrupamentos. Além disso, a transformação dos atributos originais foi essencial para que fosse possível considerá-los como um todo e utilizando ponderações razoáveis das suas importâncias relativas.

Ademais, foi possível interpretar cada um dos cinco agrupamentos encontrados, sendo, respectivamente, vídeos curtos e populares; com covariáveis médias; onda média e popularidade mais baixas; recentes, longos e com muitos *Views*, *Likes* e ruído positivos; e com muito ruído negativo. Desta forma, o objetivo inicial do artigo - de encontrar *clusters* de vídeos, com características interpretáveis - foi atingido.

Por fim, a partir dessa análise, conclui-se que vídeos semelhantes aos do agrupamento 1 podem ser úteis para aumentar o alcance do canal, enquanto os do quarto podem ser adequados para monetização, por serem longos, mas populares. Por outro lado, os do 2 e 3 não são tão interessantes de serem produzidos, por serem menos populares.

Referências

Formigari, J. (2014). Formiga Atômica. `https://www.youtube.com/@formigaatomica4294`. [Online; acessado em 14/04/2024].

Izenman, A. J. (2008). *Modern Multivariate Statistical Techniques: Regression, Classification, and Manifold Learning*. Springer Texts in Statistics. Springer-Verlag New York, 1 edition.

Maechler, M., Rousseeuw, P., Struyf, A., Hubert, M., e Hornik, K. (2022). *cluster: Cluster Analysis Basics and Extensions*. R package version 2.1.4 — For new features, see the 'Changelog' file (in the package source).

R Core Team (2018). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

SOod, G. (2020). *tuber: Access YouTube from R*. R package version 0.9.9.

Trevor F. Cox, M. C. (2001). *Multidimensional scaling*. Monographs on statistics and applied probability 88. Chapman Hall/CRC, 2nd ed edition.

Van der Laan, M., Pollard, K., e Bryan, J. (2003). A new partitioning around medoids algorithm. *Journal of Statistical Computation and Simulation*, 73(8):575–584.

Van Rossum, G. e Drake, F. L. (2009). *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA.