Clusterização dos Conteúdos Top 10 Semanais da Netflix

Ana Julia Cunha E Silva - 236038

Maio 2024

1 Introdução

Este relatório tem como propósito aplicação de técnicas de clustering baseado em modelos em um banco de dados sobre os conteúdos da Netflix global. Foi utilizada uma técnica de clustering baseado em modelos, modelo de mistura de Normais e comparado com o kmeans, para os mesmos dados com o mesmo tratamento.

Os dados foram fornecidos pela própria Netflix nesse site e são atualizados semanalmente às segundas-feiras com dados da semana anterior(Segunda a domingo). As variáveis para a modelagem foram **weekly-views** (Quantidade de Views na semana, arredondados na casa dos 100 mil), **runtime** (tempo de execução) e **TituloQtd** (Quantidade de caracteres no título).

Os pacotes utilizados no Rstudio foram: *tidyverse*, *mclust*, *readxl*, *covRobust*, *factoextra* e *cluster*. O banco possui ao todo 6040 obsevações, onde apenas 771 foram consideradas, e 11 variáveis, das quais 2 foram utilizadas e uma foi derivada de outra variável para a modelagem. Os códigos e a base de dados podem ser encontrados no repositório do github nesse link.

2 Materiais e Métodos

Os dados fornecidos pela Netflix contavam com o titulo da obra, com base nisso houve o interesse de ver se há alguma relação com o tamanho do titulo e o consumo da obra, por isso foi criada a variável **TituloQtd**. Como os dados são semanais, um título pode aparecer mais de uma vez na base pois ficou mais de uma semana no top 10, com isso em mente houve o filtro para considerar apenas a semana em que o título teve seu ápice de visualizações e aqueles com tempo de exibição maior ou igual a 1,0.

O tratamento dos dados ocorreu por meio da aplicação do log em duas variáveis e a raiz quadrada da restante, isso porquê **runtime** e **TituloQtd** tinham valores muito concentrados perto do 0 e **weekly-views** possui valores na casa dos milhões com uma disparidade muito grande. Vale lembrar que a Netflix só começou a contabilizar o tempo de exibição a partir da semana do dia 20 de junho de 2023.

Para modelar esses dados usaremos técnicas de agrupamento baseado em modelos, onde os dados são considerados como misturas de distribuições probabilísticas, no caso em específico Normais Multivariadas.

2.1 Algoritmo EM

O algoritmo EM (expectation-maximization) é um método de otimização iterativo usado para estimar parâmetros com base na máxima verossimilhança, sempre maximizando a cada iteração. Sua importância é imprescindível para clustering de modelos de mistura de distribuição.

O funcionamento do algoritmo se divide em duas partes, passo-E e passo-M. No passo-E calcula-se a esperança da log verossimilhança, enquanto no passo-M maximizamos a esperança calculada anteriormente. Esses dois passos acontecem em cada iteração até que haja a convergência. Os valores inicias que o algoritmo usa no pacote *mclust* são obtidas por meio de cluster hierárquico, e são de extrema importância para estimação dos parâmetros para o Modelo de Mistura de Normais.

2.2 Modelo de Mistura de Normais(Gaussianas)

Diferente de algoritmos baseados somente em distância como o Kmeans, cluster hierárquico ou PAM que não consideram fortemente as probabilidades e incertezas das atribuições de cada observação no cluster, os algoritmos baseados em modelos consideram. Para o MMG(Modelo de Mistura de Gaussianas) os dados pertencem a K Normais Multivarias com p observações, vetor de médias μ e matriz de variâncias-covariâncias Σ.

A matriz Σ é a mais importante para a modelagem pois é ela quem caracteriza geometricamente cada cluster por meio de 4 parâmetros, familia, volume, orientação do eixo e forma. Cada parâmetro tem pelo menos 2 duas configurações, a família pode ser Univariada, Elipsoidal, esférica ou diagonal, o volume e a forma podem ser ou variados ou iguais(equal) e por fim a orientação do eixo pode ser alinhada com algum eixo(x ou y) ou variado ou igual. Cada parametrização tem um nome especifico, nas notas de aula pode ser encontrada a tabela descrevendo cada parametrização.

Para definir qual a parametrização ideal e qual o número de clusters, foi utilizado o BIC(Bayesian Information Criterion) que faz uma aproximação da verossimilhança integrada.

2.3 Tratamento de Outliers

Existem algumas formas de lidar com Outliers no modelo de mistura, uma delas é pensar como um processo de Poisson onde pensamos nos dados como uma mistura de outliers e os clusters como normais. O pacote *mclust* tem uma opção para lidar com os ruídos, porém defini como padrão que seguem a distribuição Uniforme, portanto precisamos do pacote *covRobust* para identificar misturas de outliers.

2.4 PAM, Elbow e Silhueta

O PAM (Partition Around Method) é semelhante ao Kmeans pois seu propósito é agrupar com base na distância das observações para um ponto específico(centroide).

Existem algumas metodologias para descobrir qual o melhor número de clusters para modelos baseados em distância como o PAM e o Kmeans, nesse trabalho foi utilizado o elbow e a silhueta. O método do Elbow("cotovelo") busca encontrar o melhor número de cluster de acordo com a soma dos quadrados totais dentro dos clusters. A silhueta mede o grau de pertencimento dos elementos dentro de um grupo de acordo com a distância entre os elementos.

3 Resultados

Na figura 1 temos a visualização dos dados dois a dois, já tratados com logaritmo, raiz quadrada e filtros prontos para a modelagem. Podemos ver algumas elipses em alguns gráficos como runtime x weekly-views, dois circulos em runtime x TituloQtd.

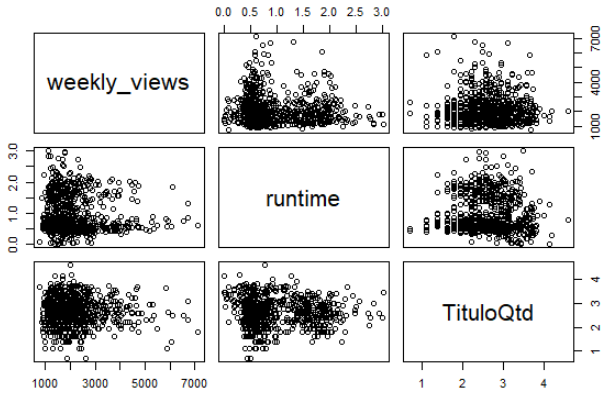

Figure 1: Dispersão dois a dois

Na figura 2 temos dois gráficos de BIC para ajudar a decidir a parametrização e o número de clusters que se ajusta melhor aos dados, na esquerda temos o BIC completo e na direita o BIC do modelo com redução de ruído(tratamento de outlier).

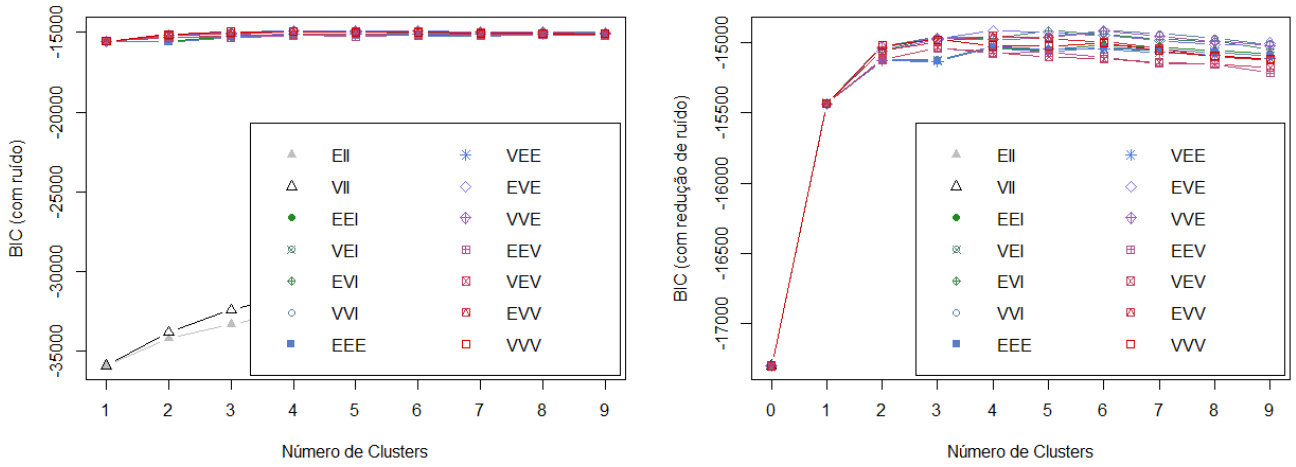

Figure 2: Comparativo BIC entre modelos

Aplicando o modelo, na tabela 1 temos as medidas resumo dos dois modelos feitos. O modelo completo trata-se de um EVI com 4 clusters, enquanto o com redução de ruído é um EVI com 5 clusters(mais os outliers), LogVerossim. é a log verossimilhança do modelo. O R trás na saída o valor do ICL também, mas ele não foi levado em consideração nesse trabalho.

Table 1: Tabela Resumo dos Modelos

Modelo	**Parametrização**	**Clusters**	**LogVerossim.**	**BIC**
Completo	EVI	4	-3894.74	-7949.025
Redução de Ruído	EVI	5	-3876.458	-7965.642

As figuras 3 e 4 mostra um pouco de como o modelo se comporta conforme aumenta a quantidade de dados, aumenta a medida da incerteza em cada um dos modelos (completo e redução de ruído). Vemos que ambos são preocupantes, o modelo de redução de ruído é mais linear porém tem uma incerteza maior, sendo assim um ponto benéfico para o modelo completo.

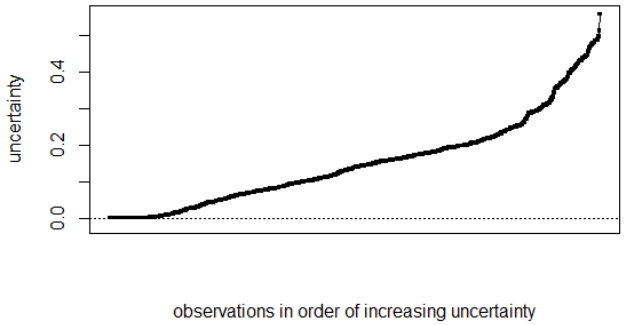

Figure 3: Incerteza: Completo

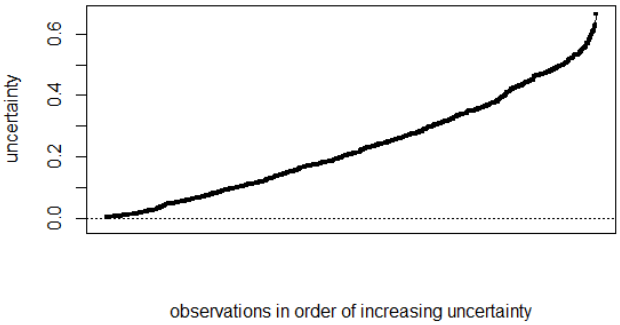

Figure 4: Incerteza: Redução de ruído

Finalmente nas figuras 5 e 6, respectivamente, conseguimos ver os clusters dois a dois por variável e quais observações ficaram mais no "limite" entre os clusters(as bolinhas maiores). As bolinhas pretas/cinza escuro na figura 4 são os outliers.

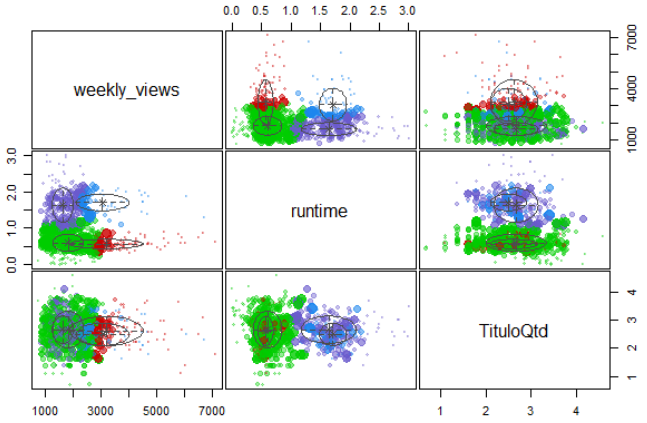

Figure 5: Dispersão: Incerteza no modelo completo

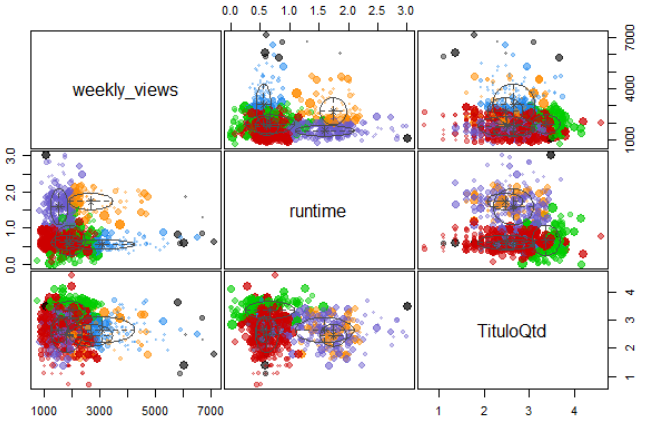

Figure 6: Dispersão: Incerteza na redução de ruído

Explorando um pouco mais a fundo os modelos, as figuras 7 e 8 trazem as densidades dos clusters numa representação que lembra curvas de nível. Cada circulo mais afastado pode ser interpretado como um cluster.

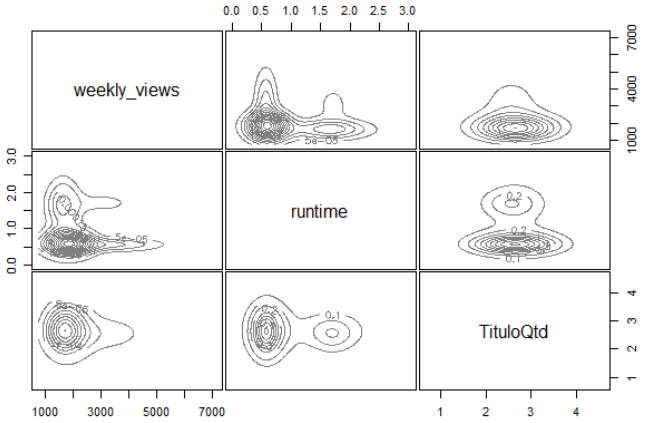

Figure 7: Densidade: modelo completo

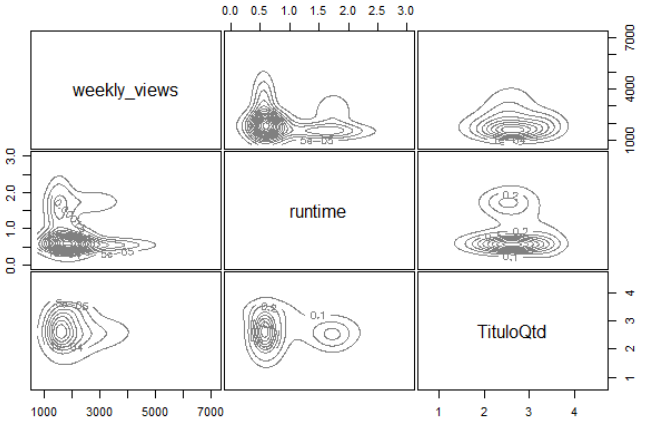

Figure 8: Densidade: redução de ruído

Por fim, fazendo um estudo de perfil de cada cluster em relação as variáveis, na figura 9 modelo completo e na figura 10 com redução de ruído.

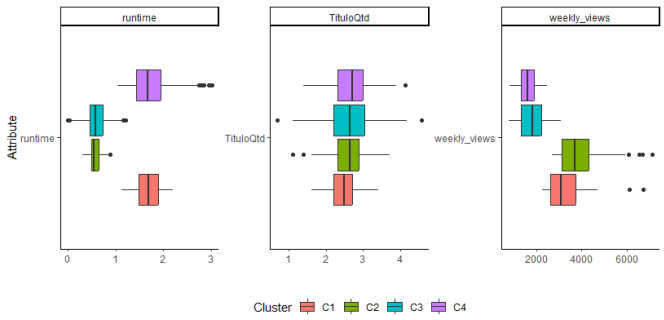

Figure 9: Análise de perfil: Completo

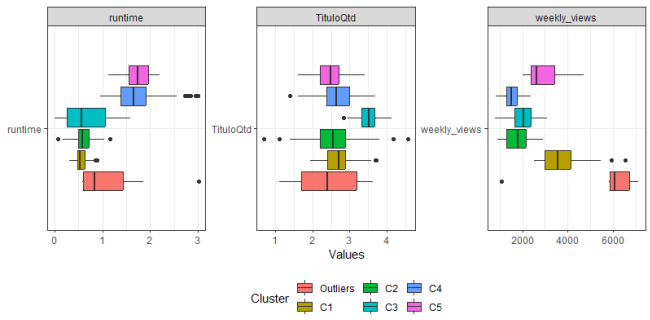

Figure 10: Análise de perfil: Redução de ruído

Com intuito de comparar a clusterização com outra técnica mais "simples" as figuras 11, 12 e 13 mostram uma rápida clusterização usando Kmeans e PAM. Começando com o estudo do número de clusters na figura 11 temos o elbow e na figura 12 o scree plot da silhueta, no elbow o número de clusters ideal é 2 ou 3 enquanto na silhueta o ideal é 2.

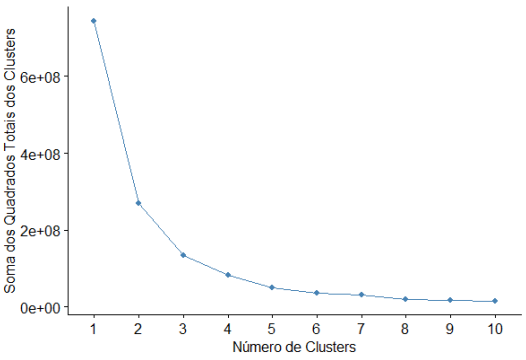

Figure 11: Elbow

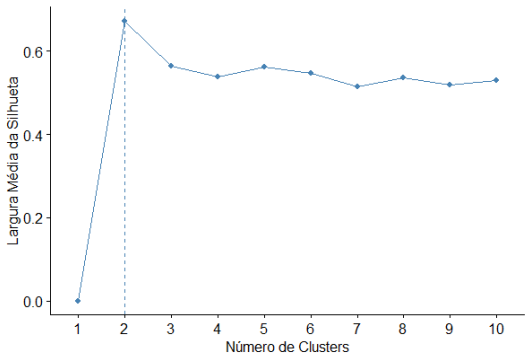

Figure 12: Silhueta

Com o estudo rápido do número de clusters definimos o ideal como 2 grupos, a figura 13 mostra uma visualização das obras por cluster. A função *fviz-cluster* calcula PCA para reduzir a dimensionalidade para conseguir gráficar, temos 36,2% da variabilidade dos dados explicado pela primeira componente(dimensão) e 34,9% na segunda componente.

Figure 13: Clusters no Kmeans

4 Discussão

A intuição inicial ao ver os dados na figura 1, é de que haviam dois grandes clusters ou no máximo três normais. Ao calcular o BIC chegamos que o recomendável eram 4 grupos, analisando os dados completos sem se importar com os outliers. Já quando levamos em consideração uma mistura de outliers, o número de clusters aumentou e o tipo "EVI" do modelo continuou o mesmo. A incerteza(figura 3 e 4) aumentando bastante em ambos os modelos era preocupante, próximo da metade da quantidade de observações a incerteza já estava acima dos 0,2 no modelo com tratamento de outliers, enquanto no modelo completo estava em torno de 0,15. Portanto o modelo completo, com menos cluster parecia mais adequado que o concorrente.

Analisando um pouco mais dos modelos nos dados para ter certeza de qual era melhor, na figura 5 vemos que runtime x weekly-views conseguem separar bem os 4 clusters, enquanto as mesmas variaveis porem comparadas com TituloQtd ficam com as observações com grande incerteza. A situação piora para 5 clusters na figura 6, muitas

incertezas e clusters misturados entre si. As densidades nas figuras 7 e 8 parecem ter de 2 a 3 clusters no máximo, exceto para runtime x weekly-views, onde novamente é possível ver os 4 grupos.

Quanto a interpretabilidade dos grupos, na figura 9 vemos que em relação ao tempo de exibição e as views semanais cada cluster tem uma característica diferente, infelizmente a quantidade de caracteres no título não tem muita diferença entre os clusters. O C1 tem tempo de exibição grande e poucas views, provavelmente são filmes. C2 tem pouco tempo de exibição e poucas views, provavelmente séries limitadas ou apenas os últimos colocados do ranking semanal. C3 é o contrário do C1, assim como C4 é o contrário do C2.

Já para o modelo com tratamento de outliers, na figura 10 vemos que a quantidade de caracteres do título é diferente no C3 e a mesma nos outros 4 clusters. Os outliers tem um perfil de muitas visualizações. Comparando com o modelo completo, C1 (modelo 2 - redução de ruído) se parece com C2(modelo 1 - completo), C4 com C4, C5(modelo 2) com C1(modelo 1) enquanto C2 e C3(modelo 2) parecem com C3 (modelo 1) mas C3 ficou com os títulos compridos.

A aplicação do método mais simples, baseado apenas na distância dos dados, trouxe à análise dos dados em dois grandes grupos, porém a interpretabilidade não ficou muito boa a ponto de conseguir concluir alguma coisa. Os resultados com o Kmeans poderiam ficar melhor se houvesse o tratamento especifico para a técnica.

Em conclusão, o tamanho do título não mostrou importância entre os grupos, é possível pensar nos dados como dois grandes grupos divididos em outros dois subgrupos (mostrados nos modelos de MMG). A interpretação é melhor quando consideramos apenas 4 grupos, o modelo pode não ter as melhores métricas mas se mostrou decente.

5 Referências

[1] Materiais das Aulas do Professor Guilherme.

[2] Pessoa, D. O algoritmo EM em estimação de densidades. 2009. Disponível em: link. Acessado em: 26/05/2024 às 15:26.

[3] Boehmke, B. Greenwell, B. Hands-On Machine Learning with R. 2022. Disponível em: link. Acessado em: 25/05/2024 às 21:28.