Universidade Estadual de Campinas
Departamento de Estatística

ME731 - Métodos em Análise Multivariada

Professor Aluísio de Souza Pinheiro

Comparação de Métodos de Clusterização em Valorant: Um Estudo com PAM e K-Means

Luiz Felipe de Oliveira Barbosa Nunes - 255403

Campinas - SP
Setembro de 2024

1 Introdução

A análise de dados multivariados requer, em diversos momentos, o uso de técnicas que possibilitem o agrupamento de observações semelhantes. Para alcançar esse objetivo, são aplicados métodos como K-médias, PAM (Particionamento em Medoides), Análise de Componentes Principais (PCA), entre outros, que realizam essa separação

A aplicação será realizada com base nos dados de desempenho de jogadores de Valorant, um jogo de tiro em primeira pessoa. No jogo, as partidas ocorrem em rodadas, onde a equipe atacante tenta plantar e defender uma bomba chamada "Spike", enquanto a equipe defensora busca desarmá-la ou eliminar todos os adversários para somar pontos. O formato de decisão segue o esquema "melhor de 24 rodadas", sendo a equipe que atingir 13 vitórias declarada vencedora.

Em Valorant, os jogadores são divididos em quatro classes principais: Duelistas, Controladores, Sentinelas e Iniciadores. Cada uma dessas classes desempenha um papel estratégico específico no jogo, o que influencia diretamente o desempenho dos jogadores durante as partidas.

O objetivo deste trabalho é aplicar a análise de clusterização utilizando o Particionamento em Medoides (PAM) e o K-médias nos dados de desempenho dos jogadores, para investigar se os grupos formados pelos métodos de clusterização são condizentes com as classes predefinidas do jogo.

Os scripts, códigos e o conjunto de dados utilizados neste trabalho estão disponíveis no repositório GitHub: https://github.com/LuizNunes2020/AnaliseMultivariadaME731_Trab1. Para a obtenção dos dados diretamente do site https://www.vlr.gg/stats, foi utilizada uma API (Application Programming Interface), disponível no mesmo repositório, que possibilita a extração e manipulação dos dados. A extração foi realizada no dia 23/09.

2 Metodologia

2.1 Análise de Componentes Principais

A Análise de Componentes Principais (PCA) é uma técnica de análise multivariada que objetiva realizar uma projeção ortogonal da matriz de dados em um espaço de menor dimensão, maximizando a variabilidade representada. Segundo Izenman (2008), seja $S = \frac{1}{n}(X - \bar{X})^T(X - \bar{X})$ a matriz de covariância estimada das colunas de X, onde $\bar{X}$ é a matriz cujas colunas correspondem às médias de cada coluna de X. Os autovalores e autovetores dessa matriz são, respectivamente, $\hat{\lambda}_1 \geqslant \hat{\lambda}_2 \geqslant \cdots \geqslant \hat{\lambda}_p \geqslant 0$ e $(\hat{v}_1, \hat{v}_2, \ldots, \hat{v}_p)$, onde $\hat{\lambda}_i$ está associado ao autovetor $\hat{v}_i$, para $i = 1, 2, \ldots, p$.

A melhor reconstrução de X com posto $t < p$ é dada por:

$$\hat{X}^{(t)} = \bar{X} + \sum_{i=1}^{t} \hat{v}_i \hat{v}_i^T (X - \bar{X}),$$

onde o escore da j-ésima componente principal de X é estimado por:

$$\hat{\psi}_j = \hat{v}_j^T (X - \bar{X}),$$

e a variância da j-ésima componente principal é estimada por $\hat{\lambda}_j$. Uma medida da qualidade da projeção, em termos de variabilidade explicada, é dada pela proporção da variabilidade representada pelas t primeiras componentes principais, ou seja:

$$\frac{\sum_{j=1}^{t} \hat{\lambda}_j}{\sum_{j=1}^{p} \hat{\lambda}_j}.$$

2.2 Particionamento em Medoides (PAM)

O Particionamento em Medoides, ou *Partitioning Around Medoids* (PAM) em inglês, é uma técnica de agrupamento utilizada para segmentar dados multivariados contínuos em um número definido de grupos. O método PAM seleciona algumas observações do conjunto de dados para atuar como medoides, que são os centros dos clusters. O objetivo é minimizar a soma das distâncias entre as observações e seus respectivos medoides.

Conforme descrito por Kaufman e Rousseeuw (2009), considere uma matriz de dados X na forma:

$$X_{p \times n} = \{x_1, x_2, ..., x_n\}$$

onde

$$x_i = (x_{i1}, x_{i2}, ..., x_{ip})^T, \quad i = 1, 2, ..., n.$$

O objetivo do agrupamento é organizar os dados em $k \geqslant 2$ grupos, considerando as p variáveis quantitativas de cada observação.

Seja $d(i, j)$ a medida de dissimilaridade entre as observações x_i e x_j, onde $i, j = 1, 2, ..., n$. Defina-se y_i como uma variável indicadora, que indica se o i-ésimo elemento é um medoide, e z_{ij} como uma variável indicadora que aponta se o j-ésimo elemento pertence ao grupo cujo medoide é o i-ésimo objeto.

O problema de otimização, originalmente formulado por Vinod (1969), pode ser descrito como:

$$\text{minimizar} \sum_{i=1}^{n} \sum_{j=1}^{n} d(i, j) z_{ij}$$

sujeito às seguintes restrições:

$$\sum_{i=1}^{n} z_{ij} = 1, \quad j = 1, 2, ..., n,$$
$$z_{ij} \leqslant y_i, \quad i, j = 1, 2, ..., n,$$
$$\sum_{i=1}^{n} y_i = k, \quad y_i, z_{ij} \in \{0, 1\}, \quad i, j = 1, 2, ..., n.$$

Essas restrições garantem que os dados sejam agrupados em k clusters, com um único medoide representando cada grupo. Também assegura que cada observação pertença a apenas um cluster, e que a atribuição ocorra somente quando a observação for designada a um medoide.

Outra perspectiva do método foi apresentada por Van der Laan et al. (2003). Nesta abordagem, considera-se a matriz de dissimilaridade $D = (d(i, j))_{n\times n} = (d(x_i, x_j))_{n\times n}$, que é simétrica. Define-se $M = \{M_1, M_2, ..., M_k\}$ como um subconjunto de k medoides retirados de X. Dado M, pode-se calcular $d(x_i, M_K)$ para cada $M_K \in M$ e $i = 1, 2, ..., n$.

Dessa forma, o valor mínimo da dissimilaridade entre x_i e os medoides é dado por $\min_{K=1,2,...,k} d(x_i, M_K) = d_1(x_i, M)$, e o medoide mais próximo de x_i é representado por $l_1(x_i, M)$. O conjunto de medoides M^* que minimiza a função:

$$M^* = \min_{M} \sum_{i=1}^{n} d_1(x_i, M)$$

é selecionado. Cada medoide M_K^* identifica um cluster, composto por todas as observações que estão mais próximas dele do que de qualquer outro medoide. Esse agrupamento é descrito pelas etiquetas $l(X, M^*) = (l_1(x_1, M^*), ..., l_1(x_n, M^*))$.

2.3 Determinação do Número de Grupos

Para utilizar o método PAM, é essencial definir o número de grupos k previamente. Várias técnicas foram desenvolvidas para estimar o número ideal de agrupamentos. Entre elas, destacam-se o gráfico de silhuetas e a estatística GAP, conforme mencionado por Everitt et al. (2011).

O **gráfico de silhuetas**, como descrito por Kaufman e Rousseeuw (2009), é uma ferramenta que avalia a qualidade da classificação para diferentes valores de k. Inicialmente, calcula-se a dissimilaridade média de x_i em relação aos outros elementos do grupo A, representada por $a(x_i)$, considerando que A possui ao menos dois elementos. Para cada um dos outros grupos C, onde $C \neq A$, é calculada a dissimilaridade $d(x_i, C)$, similar a $a(x_i)$, mas considerando os elementos de C.

Em seguida, determina-se $b(x_i) = \min_C d(x_i, C)$, com B sendo o grupo que minimiza essa dissimilaridade. A silhueta de x_i, $s(x_i)$, é definida por:

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

onde $s(x_i)$ varia entre -1 e 1. Quanto maior o valor de $s(x_i)$, melhor foi a classificação de x_i, indicando que está bem alocado em seu grupo. Os gráficos de silhuetas são construídos ordenando os valores de $s(x_i)$ em cada grupo, de forma decrescente, sendo que a "altura"do gráfico reflete a quantidade de elementos no grupo.

Além disso, pode-se calcular a média das silhuetas de cada grupo e a média global de todas as silhuetas, representada por $s(k)$:

$$s(k) = \frac{1}{n} \sum_{i=1}^{n} s(x_i).$$

Esse valor é utilizado para escolher o número ideal de k, sendo o valor de k que maximiza $s(k)$ o mais adequado. Segundo Kaufman e Rousseeuw (2009), se $s(k) \geqslant 0,71$, é considerado que há uma estrutura de agrupamento forte, enquanto valores de $0,51 \leqslant s(k) \leqslant 0,70$ indicam uma estrutura razoável.

Por outro lado, a **estatística GAP**, proposta por Tibshirani et al. (2001), formaliza a busca por um 'cotovelo' no gráfico de critério de agrupamento que otimiza o número de grupos. Suponha que os dados foram divididos nos grupos $C_1, C_2, ..., C_k$, contendo $n_1, n_2, ..., n_k$ elementos, respectivamente. Calcula-se:

$$D_r = \sum_{x_i, x_j \in C_r} d(x_i, x_j)$$

para cada $r = 1, 2, ..., k$, e define-se:

$$W_k = \sum_{r=1}^{k} \frac{D_r}{2n_r}.$$

Dessa forma, busca-se padronizar o gráfico de $\log(W_k)$ ao compará-lo com o valor esperado sob uma determinada distribuição de referência. Definindo:

$$\text{Gap}_n(k) = \mathbb{E}_n^*\{\log(W_k)\} - \log(W_k),$$

onde $\mathbb{E}_n^*(.)$ é a esperança de uma amostra de tamanho n sob a distribuição de referência. A estimativa do número de *clusters*, $\hat{k}$, será o valor que maximiza $\text{Gap}_n(k)$.

2.4 Algoritmo

O algoritmo de Particionamento em Medoides (PAM) proposto por Kaufman e Rousseeuw é executado em duas fases principais: construção e troca.

Na fase de construção, o objetivo é selecionar k medoides iniciais:

1. Considere um objeto x_i que ainda não foi selecionado;
2. Selecione outro objeto x_j, que também não foi selecionado, e calcule a diferença entre a dissimilaridade de x_j com o objeto mais semelhante já selecionado, D_j, e a dissimilaridade de x_i com x_j, denotada por $d(x_i, x_j)$;
3. Se a diferença for positiva, x_j contribui para a escolha de x_i como medoide. A contribuição C_{ji} é dada por:
$$C_{ji} = \max\{D_j - d(x_i, x_j), 0\};$$
4. Calcule o ganho total $\sum_j C_{ji}$ ao incluir x_i como medoide;
5. Selecione o objeto x_i que maximiza $\sum_j C_{ji}$ como o próximo medoide.

Após a seleção inicial dos k medoides, a fase de troca otimiza o conjunto de medoides ao trocar elementos selecionados e não selecionados:

1. Considere um objeto não selecionado x_h e calcule sua contribuição C_{hij} para a troca de x_i (selecionado) por x_j (não selecionado), definida por:
$$C_{hij} = \begin{cases} (d(x_h, x_j) - d(x_h, x_i))I(E_h > d(x_j, x_h)) + (E_h - D_h)I(E_h \leqslant d(x_j, x_h)), & \text{se } d(x_i, x_h) = D_h \\ d(x_h, x_j) - D_h, & \text{caso contrário;} \end{cases}$$
onde E_h é a dissimilaridade entre x_h e o segundo medoide mais próximo.
2. Calcule a soma das contribuições para a troca $T_{ij} = \sum_h C_{hij}$;
3. Selecione o par (x_i, x_j) que minimiza T_{ij};
4. Se o valor mínimo de T_{ij} for menor que zero, troque x_i por x_j. Caso contrário, o algoritmo para.

2.5 Método de K-Means

O método de K-means agrupa os dados em K clusters distintos, onde cada cluster é composto por observações que compartilham características semelhantes. Essas observações são representadas por uma matriz $X_{n\times p}$, com n sendo o número de observações e p o número de variáveis. A similaridade entre as observações é medida através da distância euclidiana $d(x_i, c_j)$, onde x_i é uma observação e c_j o centroide de um cluster.

O algoritmo inicia com a seleção aleatória de K centroides iniciais $c_1, \ldots, c_K$, onde cada $c_j \in \mathbb{R}^p$. A cada iteração, cada ponto x_i é associado ao centroide c_j mais próximo, de acordo com a menor distância euclidiana. Em seguida, as etiquetas dos pontos são atualizadas com base na seguinte regra:

$$c_j = \arg\min_{c\in\mathbb{R}^p} \frac{1}{N_j} \sum_{i=1}^{n} \mathbb{1}\{i \text{ está no cluster } j\} d^2(x_i, c),$$

onde N_j é o número de observações no cluster j. Este processo é repetido até que as observações não mudem mais de cluster.

A escolha do número de clusters K é um aspecto crucial no algoritmo K-Médias. Um valor apropriado de K minimiza a variação intra-cluster, ou seja, a soma das distâncias quadradas entre as observações e seus centroides. A variação intra-cluster total pode ser calculada pela fórmula:

$$\text{tot.intracluster} = \sum_{k=1}^{K} \sum_{x_j \in c_k} (x_j - \mu_k)^2,$$

onde μ_k representa a média das observações no cluster k. A ideia é escolher o menor valor de K que ainda proporciona uma baixa variação intra-cluster, garantindo uma boa separação entre os clusters e minimizando a dispersão interna.

3 Aplicação no conjunto de dados

O banco de dados utilizado possui um total de $p = 11$ variáveis e $n = 406$ observações, cada uma representando o desempenho de um jogador. As variáveis incluídas na análise são as seguintes:

- **rating**: pontuação geral do jogador, refletindo sua performance geral.
- **average_combat_score**: pontuação média de combate do jogador por rodada.
- **kill_deaths**: razão entre o número de kills (abates) e mortes.
- **kill_assists_survived_traded**: porcentagem de rodadas em que o jogador conseguiu um abate, assistência, sobreviveu ou foi trocado.
- **average_damage_per_round**: dano médio causado por rodada.
- **kills_per_round**: número médio de kills por rodada.
- **assists_per_round**: número médio de assistências por rodada.
- **first_kills_per_round**: número de primeiros abates por rodada.
- **first_deaths_per_round**: número de primeiras mortes por rodada.
- **headshot_percentage**: porcentagem geral de tiros na cabeça (*headshots*).
- **clutch_success_percentage**: porcentagem de sucesso em situações de *clutch*.

Observação: Uma situação de *clutch* refere-se a momentos críticos em que uma jogada decisiva pode definir o resultado final da partida. Esses momentos frequentemente ocorrem sob alta pressão, como nas fases finais de uma rodada ou quando o placar está equilibrado.

3.1 Análise de Componentes Principais (PCA) nos dados

Para uma melhor compreensão da estrutura dos dados, foi realizada a Análise de Componentes Principais (PCA). Essa técnica permitiu identificar as variáveis que mais contribuíram para a variação total dos dados. As cinco variáveis que apresentaram as maiores cargas nas componentes principais foram:

- **kills_per_round**
- **average_combat_score**
- **average_damage_per_round**
- **kill_deaths**
- **rating**

Essas variáveis foram selecionadas para as análises subsequentes de clusterização.

3.2 Determinação do número de agrupamentos do PAM

A escolha do número ideal de agrupamentos foi realizada com base na metodologia proposta por Kaufman e Rousseeuw (2009). Esse procedimento envolve a aplicação sucessiva do algoritmo PAM para diferentes valores de k, seguida do cálculo da métrica $\bar{s}(k)$, que avalia a qualidade do agrupamento para cada k. Após calcular os valores de $\bar{s}(k)$, eles são ordenados, e o valor de k que maximiza $\bar{s}(k)$ é escolhido como o número adequado de clusters.

A Tabela 1 apresenta os resultados dessa abordagem, com os valores de $\bar{s}(k)$ organizados de forma decrescente. A partir dos resultados, observa-se que o número ideal de agrupamentos é $k = 2$, que possui $\bar{s}(k) = 0.5185$, o que indica uma separação razoável entre os grupos.

Tabela 1: Valores de k e $\bar{s}(k)$ para diferentes números de clusters

Número de grupos(k)	**Média das silhuetas** $\bar{s}(k)$
2	0.5185
5	0.4858
4	0.4791
3	0.4789
6	0.4499
7	0.4429

3.3 Determinação do número de agrupamentos do K-Means

A escolha do número de clusters K no algoritmo K-Means é realizada com base na minimização da variação intra-cluster, que corresponde à soma das distâncias quadradas entre as observações e os respectivos centroides. O objetivo é identificar o menor valor de K que ainda garanta uma boa separação entre os grupos. Essa seleção é comumente feita por meio do *método do cotovelo*, onde se busca o ponto a partir do qual o aumento no número de clusters não resulta em uma diminuição significativa da variabilidade intra-cluster.

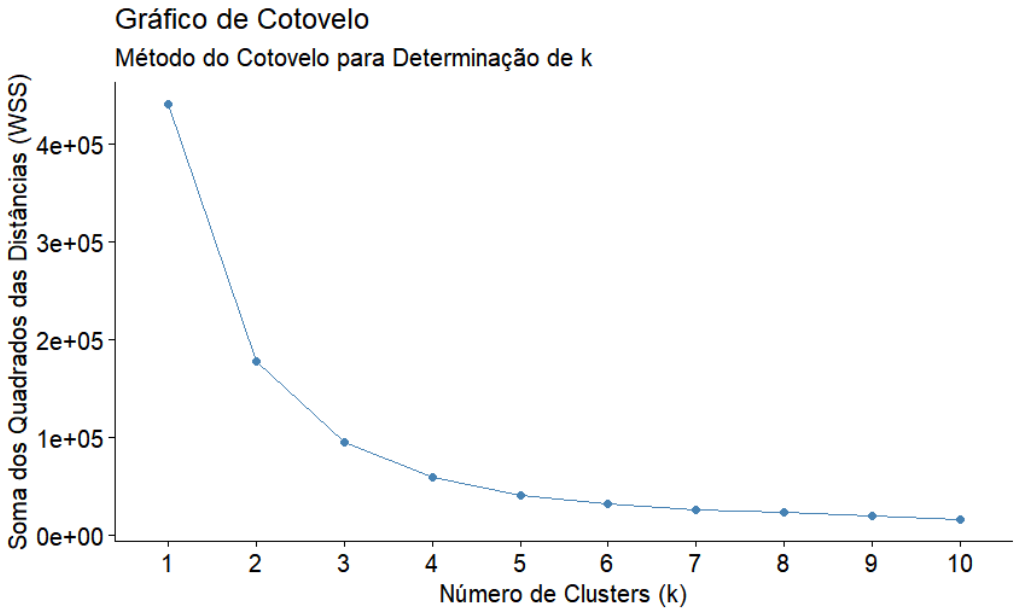

Figura 1: Método do Cotovelo

Portanto, com base no gráfico do cotovelo, observamos que o ponto de inflexão ocorre em três clusters (K = 3).

3.4 Resultados do PAM

Após definir o número de clusters, a técnica PAM foi aplicada, resultando na seleção de dois medoides que melhor representam os grupos. As observações selecionadas como representantes dos clusters correspondem às linhas 20 e 228 do conjunto de dados. A Figura 2 apresenta as faces de Chernoff dessas duas observações, e podemos observar que elas são significativamente distintas, o que reforça a separação entre os grupos.

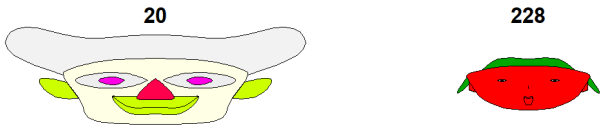

Figura 2: Faces de Chernoff das observações selecionadas como medoides.

Denotando os grupos como 1 e 2, cujos medoides correspondem às observações 20 e 228, respectivamente, é possível calcular estatísticas descritivas que resumem o agrupamento realizado. A Tabela 2 apresenta essas estatísticas, e a partir dela observa-se que o grupo 2 é maior que o grupo 1. Além disso, a dissimilaridade média no grupo 1 é superior à do grupo 2, indicando maior variabilidade dentro deste grupo. O valor máximo de dissimilaridade no grupo 1 também é consideravelmente mais alto do que no grupo 2.

Tabela 2: Resumo do agrupamento realizado

Grupo	**Tamanho**	**Dissimilaridade Máxima**	**Dissimilaridade Média**	**Diâmetro**	**Separação**
1	177	113.56	18.17	132.93	1.057
2	214	76.99	14.37	96.51	1.057

As faces de Chernoff dos elementos atribuídos ao grupo 1 estão representadas na Figura 3, enquanto as dos elementos do grupo 2 são exibidas nas Figuras 4 e 5. É possível notar que, apesar de haver certa diversidade entre os elementos, os indivíduos dentro de cada grupo apresentam maior similaridade entre si.

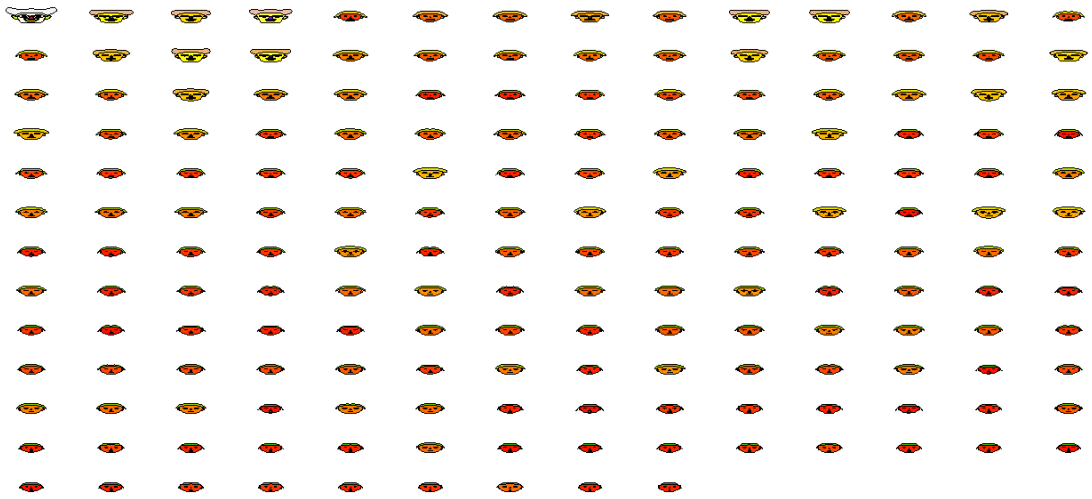

Figura 3: Faces de Chernoff dos elementos do primeiro grupo

Figura 4: Faces de Chernoff dos elementos do segundo grupo

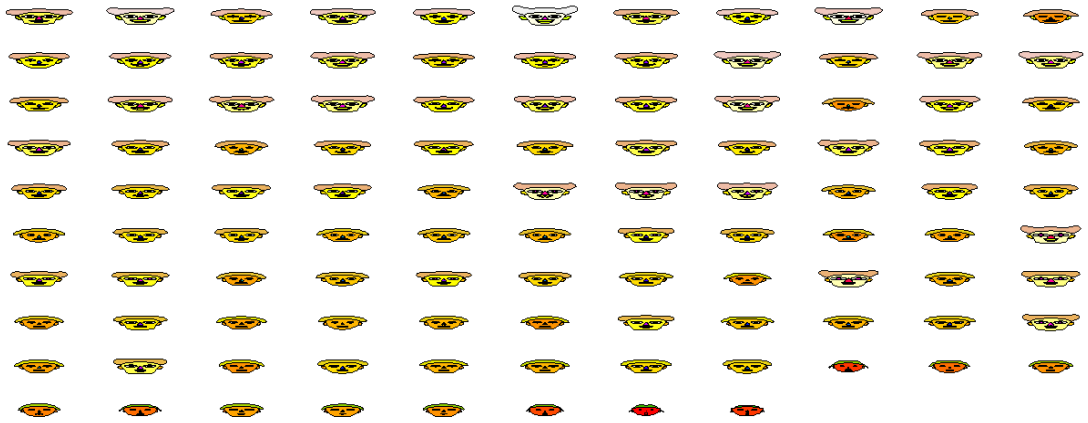

Figura 5: Faces de Chernoff dos elementos do segundo grupo

3.5 Resultados do K-Means

Nesta seção, apresentamos os resultados obtidos com o algoritmo K-Means, incluindo o gráfico de silhueta e a distribuição dos clusters. A seguir, o gráfico de silhueta é mostrado, com sua respectiva largura média de silhueta para cada cluster. Esse gráfico permite avaliar a coesão interna dos grupos formados, onde valores maiores indicam uma melhor separação entre os clusters.

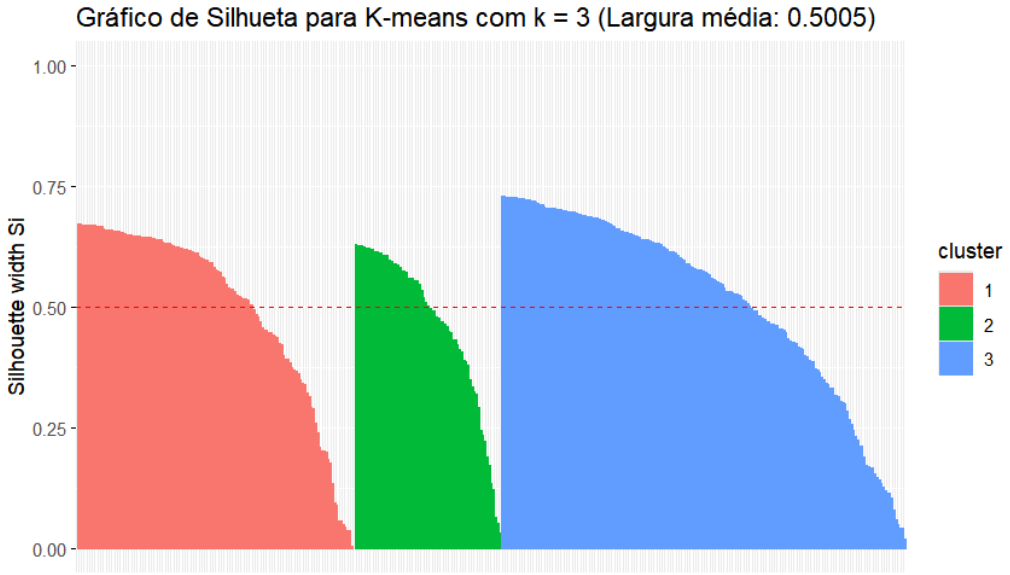

Figura 6: Gráfico de Silhueta para K-Means com $k = 3$.

A Tabela 3 apresenta o número de elementos em cada cluster e as respectivas larguras médias de silhueta:

Tabela 3: Resultados do K-Means: Tamanho dos clusters e largura média da silhueta.

Cluster	**Tamanho**	**Largura média da silhueta**
1	131	0.50
2	69	0.46
3	191	0.52

Por fim, a Figura 7 exibe a visualização dos clusters gerados pelo K-Means, com cada ponto representando uma observação e os centroides dos clusters destacados.

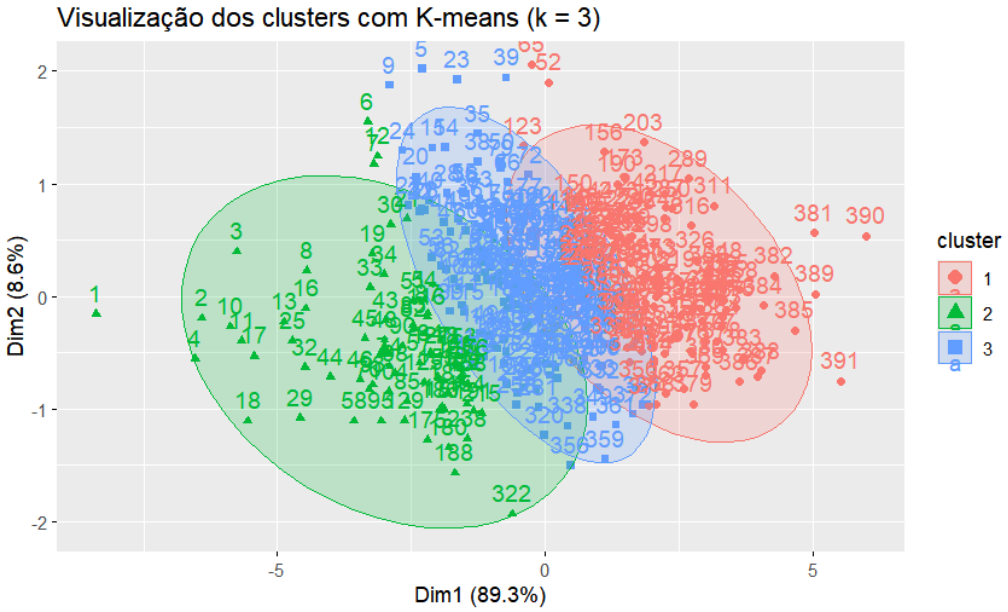

Figura 7: Visualização dos Clusters gerados pelo K-Means com $k = 3$.

4 Conclusão

Neste trabalho, foram aplicados dois métodos de clusterização, PAM e K-Means, ao conjunto de dados de desempenho de jogadores de Valorant. Ambos os métodos se mostraram adequados mas com características e desempenhos distintos.

O método PAM se destacou por selecionar diretamente os elementos mais representativos de cada cluster, os medoides, permitindo uma clara interpretação dos grupos. Além disso, a separação entre os grupos foi razoável, conforme a análise de silhuetas. No entanto, o alto custo computacional do PAM foi um limitador, especialmente em bases de dados maiores (como foi o nosso caso).

O K-Means, por outro lado, demonstrou-se mais eficiente em termos de tempo de execução. A escolha do número de clusters foi feita pelo método do cotovelo, e a análise de silhuetas mostrou uma boa separação dos clusters. No entanto, o K-Means é sensível à presença de outliers e à escolha inicial dos centroides. Mesmo assim, eu escolheria o método de K-Means para análise de clusterização desse banco de dados.

Com base nas análises de clusterização, é possível afirmar que foi viável identificar grupos de jogadores com base em seu desempenho. No entanto, não há nenhuma correspondência entre os clusters gerados pelos métodos de clusterização e as classes de agentes do Valorant (Duelistas, Controladores, Sentinelas e Iniciadores). O agrupamento foi feito com base nas estatísticas de desempenho, e essas variáveis não necessariamente mapeiam diretamente os papéis estratégicos das classes de agentes.

5 Discussão Crítica

5.1 em relação ao PAM

De acordo com Swarndeep Saket e Pandya (2016), a facilidade de entendimento e implementação está entre os principais benefícios do método PAM. Ele se fundamenta em um conceito intuitivo, onde se busca identificar as observações que melhor representam os grupos nos quais os dados serão agrupados. Kaufman e Rousseeuw (2009) propõem um algoritmo simples para atribuir os grupos aos dados, tornando a execução do método relativamente fácil.

Além disso, os autores destacam que o PAM é rápido e garante convergência em um número finito de iterações. No entanto, apesar dessa vantagem, o método não é eficiente quando aplicado a grandes volumes de dados (ou seja, quando n é muito elevado).

Outro ponto positivo do PAM é sua menor sensibilidade a outliers, já que ele trabalha diretamente com as observações do conjunto de dados. Isso torna o método mais robusto em relação a valores atípicos e discrepantes, diferenciando-o de outros métodos de agrupamento.

Por outro lado, uma desvantagem importante é seu custo computacional, que tende a ser maior em comparação com outros algoritmos de clusterização. O tempo de execução do algoritmo depende das escolhas iniciais dos medoides, o que pode resultar em uma otimização mais lenta.

5.2 em relação ao K-Means

O K-Means é amplamente conhecido por sua simplicidade e eficiência na tarefa de agrupamento de dados. Sua implementação é direta e o conceito por trás do algoritmo — a minimização da soma das distâncias quadradas entre as observações e os centroides — é fácil de entender e aplicar, como apontado por Everitt et al. (2011). Além disso, o K-Means tem a vantagem de ser altamente eficiente em termos de tempo de execução, especialmente quando aplicado a grandes conjuntos de dados, uma característica que o diferencia do PAM.

No entanto, o K-Means apresenta algumas limitações importantes. Primeiramente, o algoritmo é sensível à escolha inicial dos centroides, o que pode levar a diferentes soluções finais dependendo das condições iniciais.

Outro aspecto crítico é a suposição implícita do K-Means de que os clusters possuem uma forma esférica e de tamanhos similares, o que pode não ser verdadeiro em muitos conjuntos de dados.

Adicionalmente, o K-Means é extremamente sensível a outliers. Como o algoritmo utiliza a distância euclidiana, a presença de valores extremos pode distorcer os centroides, prejudicando a qualidade do agrupamento. Para conjuntos de dados com muitos outliers, seria recomendável o uso de algoritmos alternativos como o DBSCAN, que são mais robustos a esses pontos extremos.

Referências

[1] B. S. Everitt, S. Landau, M. Leese, e D. Stahl. *Cluster Analysis, 5ª edição*. John Wiley and Sons, 2011.

[2] L. Kaufman e P. J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley and Sons, 2009.

[3] A. Saxena, M. Prasad, A. Gupta, N. Bharill, O. P. Patel, A. Tiwari, M. J. Er, W. Ding, e C.-T. Lin. *A review of clustering techniques and developments*. Neurocomputing, 267:664–681, 2017.

[4] J. Swarndeep Saket e S. Pandya. *An overview of partitioning algorithms in clustering techniques*. International Journal of Advanced Research in Computer Engineering and Technology (IJARCET), 5(6):1943–1946, 2016.

[5] H. D. Vinod. *Integer programming and the theory of grouping*. Journal of the American Statistical Association, 64(326):506–519, 1969.

[6] A. Julian Izenman. *Modern Multivariate Statistical Techniques: Regression, Classification, and Manifold Learning*. Springer Science Business Media, 2008.

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

[8] https://www.vlr.gg/stats. Acesso em: 23 set. 2024.