

**CENTRO UNIVERSITÁRIO SALESIANO DE SÃO PAULO**

**UNIDADE LORENA**

**CURSO DE ENGENHARIA DE COMPUTAÇÃO**

**INTELIGÊNCIA ARTIFICIAL**

**PROFESSOR ME. WALMIR DUQUE**

Breno Ryan de Andrade Fernandes

Daniel Marton Barbosa

Lucas Gabriel dos Santos Moraes

Mariana Gonçalves de Freitas Ribeiro

João Vitor Ferreira Azevedo Pereira

**"Agrupamento (Clustering) /** **Inteligência Artificial Generativa e PLN"**

# Sumário

# Introdução ................................................................................................ 3

# 1. Capítulo 1 - Agrupamento (Clustering) no Python ........................... 3

1.1 - Business Understanding .................................................................... 3

1.2 ­­­­­­­- Data Understanding e Data Preparation ............................................ 16

1.3 – Modeling e Evalution ......................................................................... 19

**2. Capítulo 2 – Inteligência Artificial Generativa – Processamento de Linguagem Natural (PLN) ............................................................................................ 27**

2.1 – Definição de texto a ser trabalhado, explicação do seu gênero textual e objetivos de negócio ................................................................................................. 27

2.2 - PLN - Remoção de Ruídos, Homogeneização Stopwords ............... 8

2.3 - PLN - Stemming / Lemmatization ..................................................... 30

2.4 - Chunk / Embedding .......................................................................... 31

**Conclusão ............................................................................................... 10**

**Referências Bibliográficas .................................................................... 10**

# Introdução

O trabalho a seguir tem como foco a utilização de técnicas de data science e machine learning com o modelo proposto em sala de agrupamento também chamado de clustering. Embora haja diversos tipos de clustering utilizamos o clusters baseados em centróides para a facilitação e na mais otimista visão, a resolução de um problema de classificação de dados desconhecidos em grupos criados por similaridade para uma futura análise e possível catálogo destes por um profissional especializado nos dados trabalhados.

**1. Capítulo 1 - Agrupamento (Clustering) no Python**

1.1 Business Understanding

Nosso projeto teve como objetivo entender e segmentar pinguins com base em suas características físicas (como comprimento e profundidade do bico), a fim de facilitar a análise biológica, como a diferenciação entre espécies ou populações. Nesse sentido, o modelo de escolhido foi o agrupamento (clustering) para identificar padrões de grupos de pinguins com características semelhantes podendo prever o cluster ao qual novos pinguins pertencem.

De acordo com as tarefas de Data Science e de maneira breve, visto que alguns pontos serão mais esclarecidos ao avançar no texto temos:

**1. Pré-Processamento de Dados**

* **Normalização/Padronização**:

O código utiliza o StandardScaler para padronizar os dados. Essa tarefa é essencial para modelos baseados em distância, como KMeans, pois evita que variáveis com escalas diferentes dominem o cálculo.

* **Divisão de Dados**:

A divisão entre conjuntos de treinamento e teste (Percentage Split) é um passo importante na validação de modelos, mesmo para clustering não supervisionado.

* **Transformação dos Dados**:

Transformação dos dados em uma forma numérica adequada para análise.

**2. Clustering (Agrupamento)**

* **Técnicas de Clusterização**:

Implementação do algoritmo KMeans com diferentes métodos de inicialização (random, k-means++).

Uso de diferentes métricas de distância: Euclidiana e Manhattan.

* **Análise do Número Ótimo de Clusters**:

O uso do WCSSE e o método do cotovelo para determinar o número ideal de clusters faz parte de um processo exploratório para avaliação de modelos.

**3. Avaliação do Modelo**

* **Métricas de Avaliação Interna**:

WCSSE (Within-Cluster Sum of Squares): Avalia a compactação dos clusters.

Silhouette Score: Mede a separação entre clusters e o quão bem os dados estão agrupados.

* **Validação Cruzada (Cross-Validation)**:

Ainda que o clustering seja uma técnica não supervisionada, o código adaptou a ideia de validação cruzada para avaliar diferentes configurações de clusterização.

**4. Visualização**

* **Método do Cotovelo (Elbow Method)**:

Produz um gráfico visual para ajudar na seleção do número ideal de clusters.

* **Exploração de Dados**:

O código sugere a visualização de características médias dos clusters, o que pode ser feito por gráficos.

**5. Análise Exploratória dos Clusters**

* **Interpretação das Características dos Clusters**:

Cálculo das médias das variáveis em cada cluster para entender o perfil de cada grupo.

* **Perfilação dos Grupos**:

Identificação de padrões ou perfis dentro dos dados com base nos clusters.

**6. Aplicação em Cenários Reais**

* **Segmentação de Dados**:

Agrupamento de espécies (como no caso dos pinguins).

* **Detecção de Padrões**:

Identificação de padrões escondidos nos dados para suporte à tomada de decisão.

* **Análise de Dados Multivariados**:

O código pode ser considerado apropriado para explorar e agrupar dados com várias dimensões.

**7. Significado dos Clusters**:

A explicação do significado de cada cluster é baseada nas características médias de cada grupo.

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

* Cada cluster representa grupos com diferentes características médias em termos de massa, tamanho do bico e comprimento das asas, provavelmente correspondendo a diferenças entre espécies ou subgrupos de pinguins.

**8. Possíveis melhorias**

Com pequenas adaptações, ele também poderia ser usado para:

* **Detecção de Anomalias**.
* **Redução de Dimensionalidade** (como prévia ao clustering)

Exemplo de como os dados estavam antes do nosso trabalho:

Gráfico, Gráfico de dispersão

Descrição gerada automaticamente

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

**Requisito Funcional 1**: Identificar clusters de pinguins com características semelhantes



**Requisito Funcional 2**: Permitir a previsão de cluster para novos pinguins inseridos

Texto

Descrição gerada automaticamente

**Requisito Funcional 3**: Exibir visualizações gráficas dos clusters formados

Medida de Distância Euclidiana

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

Visualização

Gráfico, Gráfico de dispersão

Descrição gerada automaticamente

Medida de distância Manhattan

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

Visualização:

Gráfico, Gráfico de dispersão

Descrição gerada automaticamente

**Requisito Funcional 4:** Determinar o número ideal de clusters usando o método Elbow

Texto

Descrição gerada automaticamente

Para a visualização:

Texto

Descrição gerada automaticamente

Visualização:

Gráfico, Gráfico de linhas

Descrição gerada automaticamente

**Requisito Funcional 5**: Criar clusters e avaliar os resultados com base nas características físicas dos pinguins

Texto

Descrição gerada automaticamente

**Requisito Funcional 6**: Uso de distância de Manhattan para formar cluster alternativos

Texto

Descrição gerada automaticamente

Tela preta com letras brancas

Descrição gerada automaticamente

Requisitos Não Funcionais:

- O modelo precisa ser eficiente para grandes volumes de dados, uma vez que esse número tende a crescer.

- Precisa ser escalável para poder agregar mais variáveis ou espécies.

1.2 Data Understanding e Data Preparation

Variáveis (Features):

Valores mínimos

Texto

Descrição gerada automaticamente

Valores maxímos

Texto

Descrição gerada automaticamente

Valores médios

Texto

Descrição gerada automaticamente

- culmen\_length\_mmtipo float — Representando o comprimento do bico.

- culmen\_depth\_mmtipo float — Representando a profundidade do bico.

- flipper\_length\_mm tipo float — Representando o comprimento da barbatana.

- body\_mass\_g tipo float — Peso do pinguim.

- Excluímos da análise a feature “sex”, pois nesse caso a diferença entre um sexo ou outro não influência o tipo de espécie.

Target e Classes:

- Não há variável alvo (target) porque o problema é de agrupamento (clustering).

- Os clusters formados (0, 1 e 2) são as "classes" que o algoritmo deve identificar.

Preparação de Dados:

Verificamos como o estava o dataset com esse input:

Texto

Descrição gerada automaticamente

Obtivemos esse output:

Tela de vídeo game

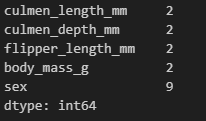
Descrição gerada automaticamente

Como demonstra a imagem existem valores nulos (NaN).

Aplicamos outra verificação:



Output:



Normalizamos utilizando a classe StandardScaller da biblioteca scikit-learn para fazer o tratamento:

Texto

Descrição gerada automaticamente

Nova verificação:



Output:

Texto

Descrição gerada automaticamente

1.3 Modeling e Evaluation

Método do Cotovelo para Determinação de K

O método do cotovelo (Elbow Method) é utilizado para encontrar o número ideal de clusters. Esse método nos revela que a partir de certo ponto o aumento de clusters se torna menos significante do que de início.

Gráfico, Gráfico de linhas

Descrição gerada automaticamente

Este gráfico ajuda a identificar o ponto onde a diminuição da WCSS (within-cluster sum of squares) começa a ser mais suave, indicando o número ideal de clusters. Percebe-se que a partir de 3 clusters a diferença não diminuiu significativamente como anteriormente, por isso definimos em nosso modelo o K = 3.

Teste de Parâmetros

|  |  |  |
| --- | --- | --- |
| Inicialização | Medida de Distância | Resultado |
| Random | Euclidiana | Texto  Descrição gerada automaticamente |
| Kmeans | Euclidiana | Texto  Descrição gerada automaticamente |
| Random | Manhattan | Texto  Descrição gerada automaticamente |
| Kmeans | Manhattan | Texto  Descrição gerada automaticamente |

Medida de distância Euclidiana (Random e kmeans) :Interface gráfica do usuário, Texto

Descrição gerada automaticamente

Medida de distância Manhattan:

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

Tela preta com letras brancas

Descrição gerada automaticamente

Nesse ponto da análise o que nos trouxe a curiosidade foi que somente com a inicialização Random e a distância Manhattan é que o “Novo pinguim” foi classificado para o cluster 2 e sendo esse teste o que teve o melhor índice de WCSS ~= 531.

Tabela de Estratégias de Testes

|  |  |  |  |
| --- | --- | --- | --- |
| Inicialização | Distância | Validação | Resultado |
| Random | Euclidiana | Percentage\_split |  |
| Kmeans | Euclidiana | Percentage\_split |  |
| Random | Manhattan | Percentage\_split |  |
| Kmeans | Manhattan | Percentage\_split |  |
| Random | Euclidiana | Cross\_validation |  |
| Kmeans | Euclidiana | Cross\_validation |  |
| Random | Manhattan | Cross\_validation |  |
| Kmeans | Manhattan | Cross\_validation |  |

Métrica Silhouette Score

Utilizamos o Silhouette Score pois é uma medida que leva em conta tanto a coesão quanto a separação dos clusters. Para isso tivemos que importar da biblioteca Sklearn. Por exemplo, se chamarmos a coesão de x e separação de y o Silhouette Score para um único ponto é obtido pela fórmula:

**s = (y-x)/(max(x,y))**

O valor dessa métrica varia de menos um a um. Valores próximos de 1 indicam que o ponto está bem ajustado ao seu próprio cluster e mal ajustado aos clusters vizinhos. Se o valor é próximo de zero, temos um ponto que está próximo a um limite de decisão entre dois clusters. Já um valor negativo indica que o ponto pode ter sido atribuído ao cluster errado. Além disso, podemos calcular o Silhouette score para todo o conjunto de pontos.

Testando com um Novo Pinguim (usando um dataset com apenas uma linha)

Texto

Descrição gerada automaticamente

Melhor Modelo ?

Depende.

O WCSSE mede a compactação dos clusters: quanto menor o WCSSE, maispróximos estão os dados dentro de cada cluster. Um WCSSE menor é, portanto, desejável, mas não deve ser analisado isoladamente. Precisamos equilibrar o WCSSE com outras métricas, como o Silhouette Score, que avalia separação e coesão.

Revisão do Modelo Random + Manhattan

* Silhouette Score: 0.3836 (menor que os modelos baseados em Euclidean, que têm 0.3962).
* WCSSE: 373.71, o menor entre todos os modelos.

O baixo WCSSE sugere clusters compactos, mas o Silhouette Score mais baixo indica que esses clusters não estão tão bem separados quanto os modelos baseados em Euclidean.

Se priorizarmos a compactação (WCSSE):

O modelo **Random + Manhattan** é o melhor.

Justificativa: Ele apresenta o **menor WCSSE** (373.71), o que significa que os clusters são mais compactos, mesmo que estejam ligeiramente menos separados.

Se priorizarmos a separação (Silhouette Score):

O modelo **KMeans++ com Euclidean** é o melhor.

Justificativa: Ele apresenta o **maior Silhouette Score** (0.3962), indicando clusters bem separados e compactos.

O melhor modelo depende do objetivo específico:

Se a separação clara entre clusters for mais relevante (por exemplo, para fins interpretativos ou classificatórios), escolha KMeans++ com Euclidean.

Gráfico, Gráfico de dispersão

Descrição gerada automaticamente



Se a compactação intra-cluster for mais importante como, minimizar custos ou distância em análises operacionais escolha Random + Manhattan.Gráfico, Gráfico de dispersão

Descrição gerada automaticamente



Em cenários gerais, a separação e compactação equilibradas fazem do KMeans++comEuclidean a melhor escolha. No entanto, se o objetivo específico priorizar compactação extrema, o Random + Manhattan é o vencedor.

Conclusão

Podemos dizer que nosso trabalho foi bem desafiador ainda que em teoria não se tratasse de uma técnica (clustering de centróides) tão difícil uma vez que utilizamos o tipo de clustering mais fácil para se implementar e compreender para profissionais fora da área de TI. Todavia, acreditamos termos feito um bom serviço o qual nos dispomos a fazer. Os dados foram normalizados, tratados e utilizados em diversos testes diferentes e por mais que se há modificações a serem feita para um desempenho e qualidade maior estamos contentes.

# 2. Capítulo 2 - Inteligência Artificial Generativa

## 2.1 - Definição de texto a ser trabalhado, explicação do seu gênero textual e objetivos de negócio

O texto utilizado foi obtido por meio do dataset “Notícias publicadas no Brasil“ disponibilizado pelo site Kaggle, esse conjunto de dados foi criado a partir da raspagem de dados (web scraping) em diversos sites como G1 e SportTV feita pelo criador do conjunto. Dentro desse conjunto há alguns temas específicos de notícias (política, esporte, economia e famosos), optamos por filtrar o tema em “esporte” pelo fato de ser uma temática de cunho mais próximo a grande maioria de público não necessitando de repertório mais técnico para lidar com assunto como “economia”, ou causar alguma polêmica com “política” e ou textos, geralmente de menor profundidade intelectual “famosos”. Segundo o site Mundo Educação Notícia é um texto do campo jornalístico cuja função é relatar acontecimentos cotidianos, com a forte presença de elementos narrativos e descritivos. A notícia é um dos gêneros textuais mais dinâmicos e lidos na atualidade. Partindo disso a escolha do gênero textual se deu justamente por sua estrutura previsível e sua linguagem formal facilitou-se o processamento (remoção de ruídos). Os objetivos de negócio foram a filtragem e categorização de subtemas para melhor análise, resumo de notícias para facilitar a leitura, podendo se obter alguns insights, analisar de tendências a procura de identificar palavras-chave ou temas que estão se destacando nas notícias de esportes ao longo do tempo. Além de uma personalização de conteúdo desenvolvendo um sistema de recomendação para exibir notícias relevantes ao interesse dos usuários.

### 2.2 - PLN - Tratamento do texto: Remoção de Ruídos, Homogeneização e Stopwords

Como supracitado acima, antes mesmo de iniciarmos o processo de tratamento do texto necessitamos fazer a filtragem do dataset para o uso apenas, do que em nosso raso entendimento, julgamos fundamental ao começarmos o trabalho propriamente dito com o recurso do framework Pandas extraímos o CSV e a partir dele criamos nosso dataset filtrado somente com o assunto “esportes”. Para o tratamento de ruídos foi utilizada a biblioteca nativa do python Re para lidar com expressões regulares, formatação de das palavras em minúsculas evitando assim diferenciação a exemplo “Tenista” e “tenista” e suas inúmeras variações somado ao processo de tokenização utilizamos a biblioteca NLTK (Natural Language ToolkIt) através do corpus padrão “stopwords” adaptado para língua portuguesa, juntamente com a nossa lista de palavras stopwords customizadas, “a, e, i, o, u, né, aí, tá, então”, por se tratar de um texto jornalístico não tivemos muitos problemas ou mesmo necessidades especiais como gírias, coloquialismo tal como “né e tu”, valores numéricos ou caracteres especiais “!,@, $” também foram tratados e outras particularidades que nossa linguagem cotidiana utiliza sem perceber. Por último, foi feito testes para encontrar similaridade entre textos após a integração de modelos de embeddings fornecidos

pela biblioteca Sentence Transformers e bancos de dados vetoriais disponibilizados por outra biblioteca chamada ChromaDB. Assim, acreditamos termos feito um texto com alguma qualidade de homogeneização.

### Carregamento dos dados

Texto

Descrição gerada automaticamente

Teste para conferir as colunas antes da filtragem

Texto

Descrição gerada automaticamente

Criação do novo dataframe somente com os dados relacionados ao tema esportes



Tratamento das stopwords

Texto

Descrição gerada automaticamente

Formatação das palavras em minúsculas, limpeza, lematização e tokenização

Texto

Descrição gerada automaticamente

### Tratamento de valores ausentes

Texto

Descrição gerada automaticamente

### 2.3 - PLN - Stemming / Lemmatization

Sobre a escolha entre Stemming ou lemmatization a resposta foi avaliada com certa intuição, uma vez que ao se utilizar do stemming (radicalização) ganhamos com maior performance visto que há uma grande redução em certas palavras ao se utilizar apenas o radical, no entanto por se tratar de um texto informativo optamos pela lematização garantindo uma maior compreensão e integridade da escrita original para tal foi utilizada a biblioteca Spacy com o modelo pré-treinado spacy.load("pt\_core\_news\_sm")

Importação do spacy para lematizar e tokenizar

Texto

Descrição gerada automaticamente

### 2.4 - Chunk / Embedding

Quanto ao chunk limitamos em manter a quantidade máxima de “pedaços de texto” e o número máximo de sobreposição de caracteres conforme foi utilizado em sala de aula por conta da própria natureza do texto trabalhado, uma vez que poderíamos reduzir redundância apenas reduzindo o overlap ou aumentarmos a complexidade semântica do texto aumentando o número de chunks, essa última parte foi mais bem vista em esboços desse trabalho quando utilizamos texto do tipo romance onde a mais figuras de linguagem, metáfora e poesia.

Divisão do texto em chunks

Texto

Descrição gerada automaticamente

# Conclusão

# 6. Referências Bibliográficas

**GLOBO ESPORTE.** Nadal precisa de mais de duas horas para vencer número 74 do mundo. *ge.globo*, 13 jan. 2014. Disponível em: <https://ge.globo.com/tenis/noticia/2014/01/nadal-precisa-de-mais-de-duas-horas-para-vencer-numero-74-do-mundo.html>. Acesso em: 16 nov. 2024.

**DATA GEEKS.** Embeddings. *Data Geeks*, [s.d.]. Disponível em: https://www.datageeks.com.br/embeddings/. Acesso em: 16 nov. 2024.

**ELASTIC.** O que é Word Embedding? *Elastic*, [s.d.]. Disponível em: <https://www.elastic.co/pt/what-is/word-embedding>. Acesso em: 16 nov. 2024.

ANCHIÊTA, Rafael; NETO, Francisco A. R.; MARINHO, Jeziel C.; MOURA, Raimundo. **PLN: das técnicas tradicionais aos modelos de deep learning**. In: **TÍTULO DO LIVRO**. [s.d.]. Capítulo 1. *Sociedade Brasileira de Computação*. Disponível em: <https://books-sol.sbc.org.br/index.php/sbc/catalog/download/79/341/600?inline=1>. Acesso em: 16 nov. 2024.

**BARBOSA, Wellington.** **Processamento de Linguagem Natural (PLN).** *Wellbar - Computação*, 17 jan. 2023. Mini curso de Processamento de Linguagem Natural. Disponível em: <https://www.youtube.com/watch?v=q8NomaZCpjs&list=PLxKmxm_IWVlQCoWxMr-0WoB1gz3o0MhUl> Acesso em: 16 nov. 2024

ESCOLA DNC. **Clusterização: o guia definitivo para análise de dados não supervisionada**. Escola DNC, [s.d.]. Disponível em: <https://www.escoladnc.com.br/blog/clusterizacao-o-guia-definitivo-para-analise-de-dados-nao-supervisionada/>. Acesso em: 18 nov. 2024.

DATA GEEKS. **Clustering**. Data Geeks, [s.d.]. Disponível em: <https://www.datageeks.com.br/clustering/>. Acesso em: 18 nov. 2024.

AWARI. **Aprenda sobre clustering em machine learning**. Awari, [s.d.]. Disponível em: <https://awari.com.br/aprenda-sobre-clustering-em-machine-learning/>. Acesso em: 18 nov. 2024.

MAGALHÃES, Lúcia Helena de; SOUZA, Renato Rocha. **Agrupamento automático de notícias de jornais on-line usando técnicas de machine learning para clustering de textos no idioma português**. *Múltiplos Olhares em Ciência da Informação*, [s.d.]. Disponível em: <https://periodicos.ufmg.br/index.php/moci/article/view/19170/16237>. Acesso em: 18 nov. 2024