

**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 ............................................ 4

1.3 – Modeling e Evalution ......................................................................... 4

**2. Capítulo 2 – Inteligência Artificial Generativa – Processamento de Linguagem Natural (PLN) ............................................................................................ 8**

2.1 – Definição de texto a ser trabalhado, explicação do seu gênero textual e objetivos de negócio ................................................................................................. 8

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

2.3 - PLN - Stemming / Lemmatization ..................................................... 8

2.4 - Chunk / Embedding .......................................................................... 9

**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.

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):

- 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

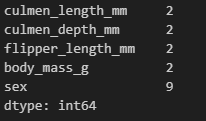
Descrição gerada automaticamente

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

Elbow Method para Determinação de K

O método do cotovelo (Elbow) 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: O código utiliza o método `k-means++` para inicialização, que é geralmente melhor do que a inicialização aleatória, pois tende a gerar clusters mais estáveis.

- Distância: A medida de distância utilizada é a Euclidiana, que é a mais comum em K-means e a Manhattan.

Medida de distância Euclidiana: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

Tabela de Estratégias de Testes

| Parâmetro | Valor 1 | Valor 2 | Valor 3 |

|----------------------|-----------------|-----------------|----------------|

| Inicialização | random | k-means++ | |

| Medida de Distância | Euclidiana | Manhattan | |

| Técnica de Validação | Percentage Split| Cross Validation| |

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

Texto

Descrição gerada automaticamente

Visualização e Avaliação dos Clusters

Avaliação do Modelo

Você pode utilizar métricas como WCSSE. (Within-Cluster Sum of Squares) para avaliar a qualidade do modelo, que é o critério usado no método do cotovelo.

- WCSSE: O valor de WCSS diminui à medida que o número de clusters aumenta, e isso pode ser analisado a partir do gráfico do cotovelo.

Qualidade dos Clusters

Os clusters formados podem ser analisados em relação às suas características médias, que ajudam a explicar o significado de cada grupo. Um exemplo seria analisar a média das variáveis para cada cluster:

```python

import pandas as pd

cluster\_means = pd.DataFrame(Xpenguins\_scaled, columns=['Comprimento do Culmen', 'Profundidade do Culmen']).groupby(kmeans.labels\_).mean()

print(cluster\_means)

```

Melhor Modelo

Você pode comparar os resultados dos diferentes modelos (com diferentes valores de K, inicialização, ou distância) e escolher o modelo com a menor WCSS e melhor capacidade de generalização (por exemplo, via Cross Validation).

Conclusão

O código fornecido parece atender a grande parte dos requisitos, mas algumas adaptações, como a alteração do cálculo da distância ou mais experimentos com diferentes inicializações e validações, seriam necessárias para uma análise completa.

# 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.

### 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")

### 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.

# Conclusão

Deve ser fundamentada nos resultados e na discussão, contendo deduções lógicas correspondentes aos objetivos propostos. A conclusão constitui-se de uma resposta as hipóteses enunciadas. Portanto, o autor manifesta seu ponto de vista sobre os resultados obtidos. Devem surgir aqui também as dificuldades encontradas e recomendações futuras, para o aprimoramento do curso de Inteligência Artificial.

# 6. Referências Bibliográficas

Devem seguir as Normas ABNT para Monografias

**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