**Relatório <010> - < 10 - Prática: Lidando com Dados do Mundo Real (II) >**

<Kawan Machado>

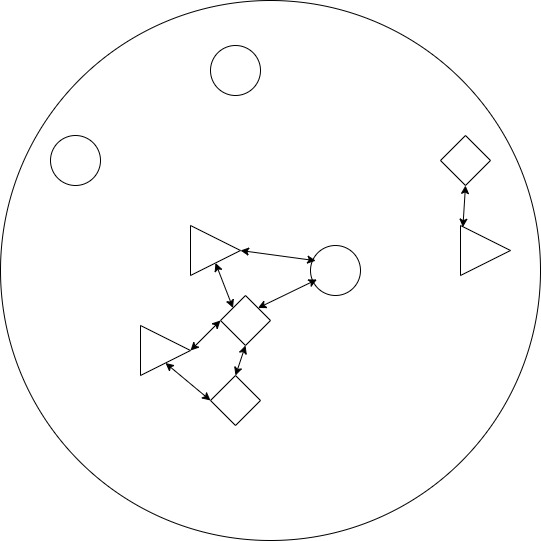
# Descrição da atividade

## More Data Mining and Machine Learning Techniques

### K-Nearest-Neighbors Concepts

Iniciando por esse conceito, fica evidente como as técnicas de aprendizado de máquina podem ser tão simples. A ideia é classificar os pontos dos dados baseado na distância entre si. Para cada nova entrada, o algoritmo classifica a distância dos vizinhos mais próximos e agrupa os baseados em suas distâncias.

FIGURA 1 - K-NEAREST-NEIGHBORS

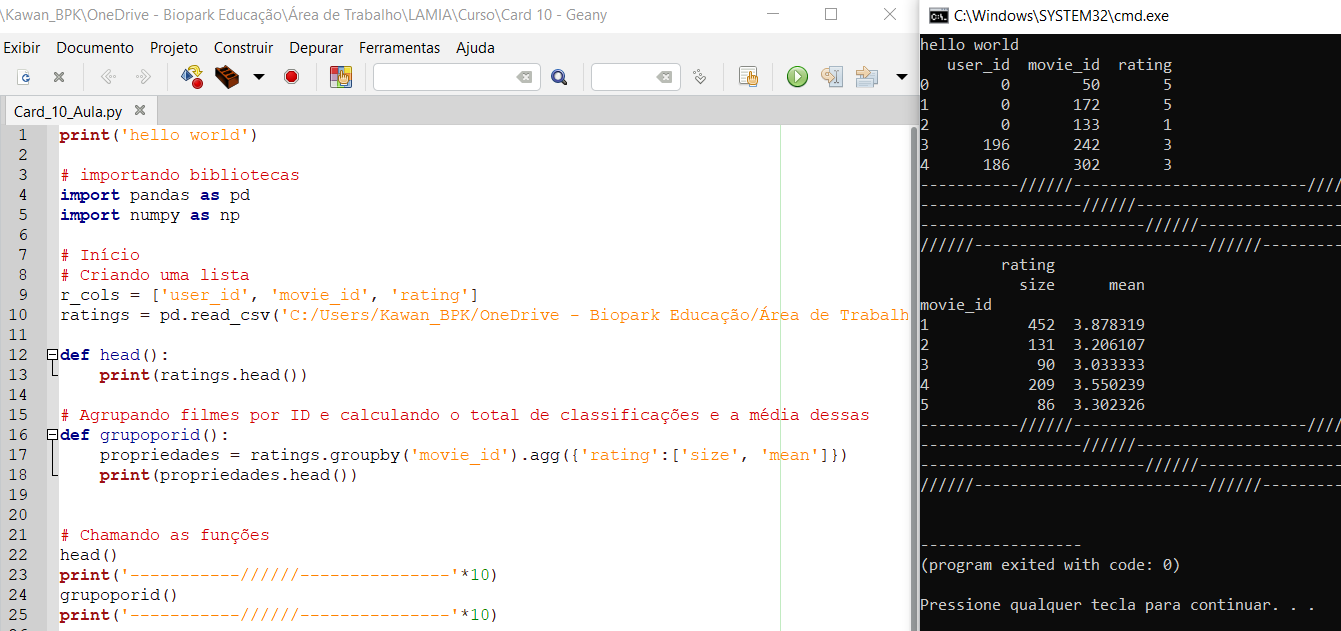
Elaborado pelo próprio autor, 2025.

O KNN é um dos modelos mais simples de aprendizado de máquina, ele é classificado como aprendizado supervisionado. No próximo tópico vamos explorar a classificação de filmes baseado nos seus metadados.

### Activity Using KNN to predict a rating for a movie

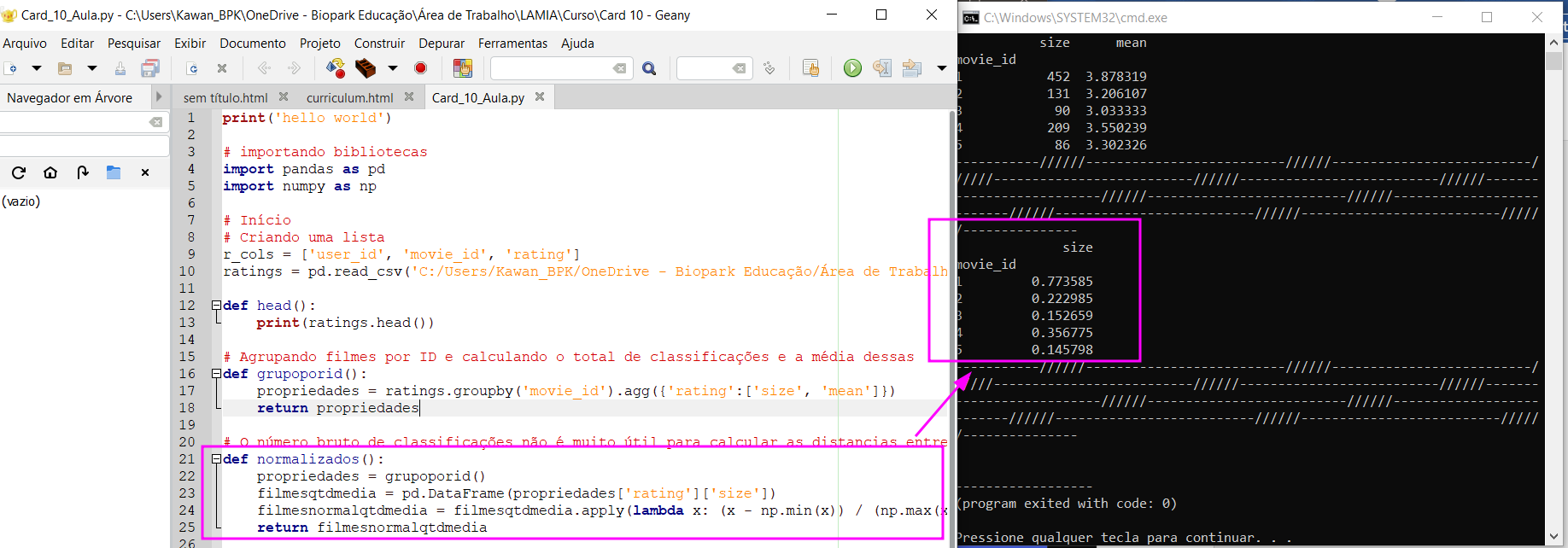
Para esse exemplo vamos tratar o data set ‘MovieLens’, com ele vamos classificar os filmes olhando os 10 mais próximos em termos de gênero e popularidade e com isso vamos conseguir predizer a avaliação média esperada de um filme do mesmo gênero. Primeiro fazemos a importação dos dados e depois calculamos a quantidade e a média de classificação dos filmes.

FIGURA 2 – IMPORTAÇÃO E PRIMEIRO CÁLCULO

Elaborado pelo próprio autor, 2025.

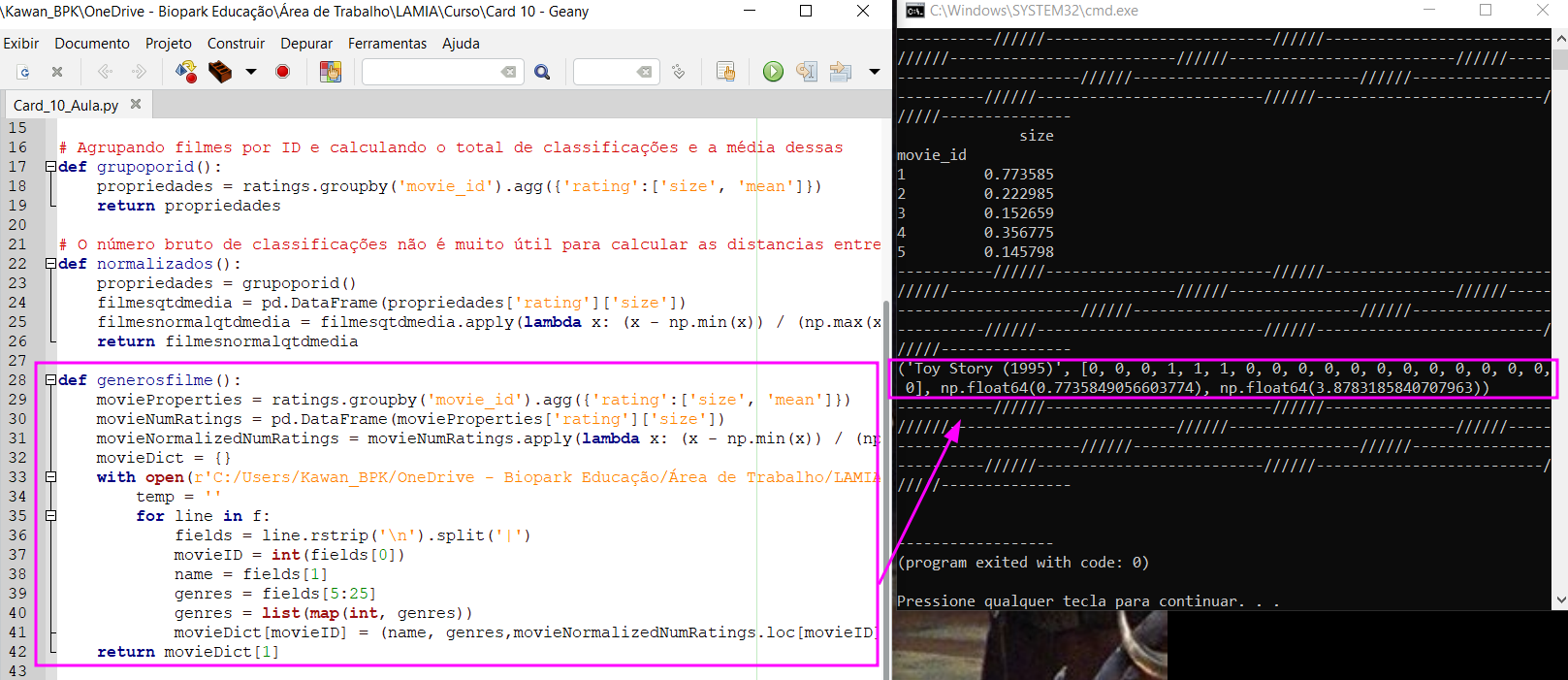
A média bruta de avaliações dos filmes pode conter problemas, pois alguns filmes tem mais avaliações que outros. Para isso efetuamos a normalização dos valores para eles ficarem entre 0 e 1, com isso conseguimos prosseguir a análise.

FIGURA 3 – NORMALIZAÇÃO DOS DADOS

Elaborado pelo próprio autor, 2025.

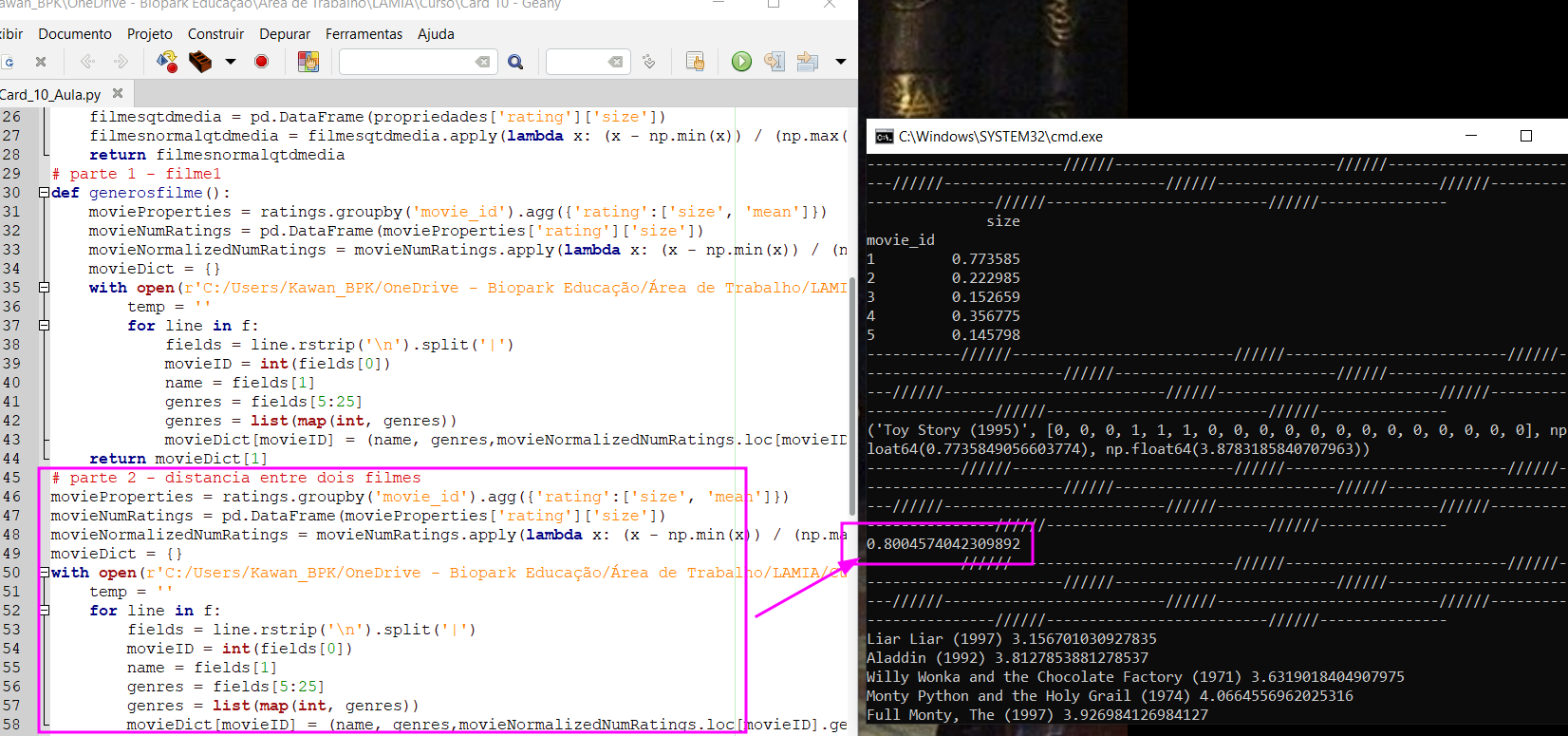
Agora precisamos importar a informação dos gêneros dos filmes que estão contidos no arquivo “u.file” e mesclar com os valores das médias de avaliação e com os valores normalizados das mesmas. Para exemplificar trouxemos os valores do item 1, que corresponde ao filme ‘Toy Story’ que se encaixa em três genêros, tem uma média de avaliações normalizada de 0,77 e uma média bruta de 3,87.

FIGURA 4 – DADOS DO FILME 1

Elaborado pelo próprio autor, 2025.

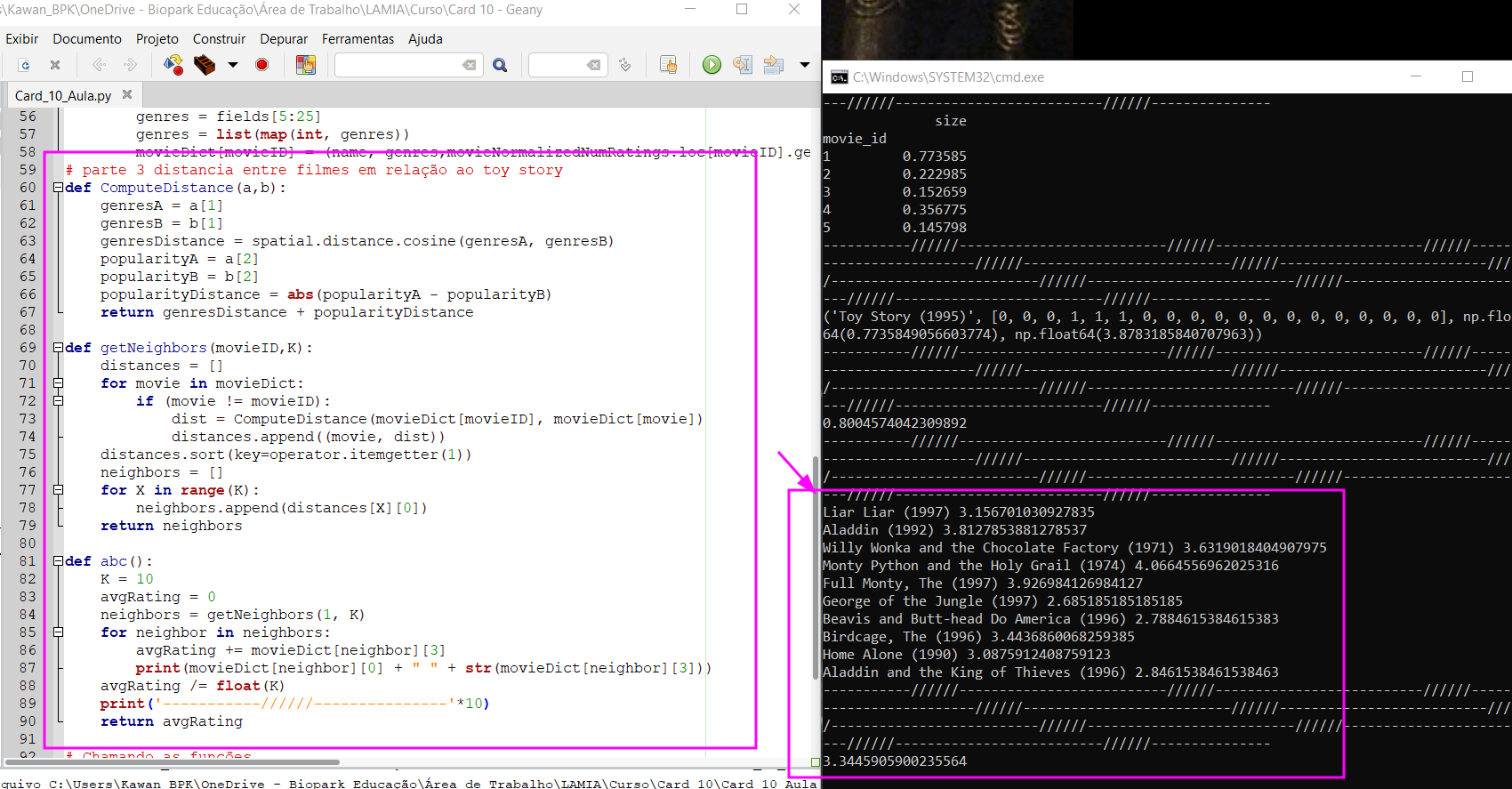
Agora vamos computar a distância entre dois filmes, no caso o filme 2 e o filme 4. O resultado é um valor de 0,80, lembrando que quanto maior menos similar um filme é de outro, nesse caso, 0,80 significa que os filmes não são parecidos.

FIGURA 5 – DISTÂNCIA ENTRE FILME 2 E 4

Elaborado pelo próprio autor, 2025.

Por fim, vamos usar o filme 1 (‘toy sotry’) para agrupar todos os filmes do mesmo gênero e calcular o valor médio de avaliação destes e com isso predizer a média de avaliação esperada para um filme do mesmo grupo.

FIGURA 6 – MÉDIA DE AVALIAÇÃO

 Elaborado pelo próprio autor, 2025.

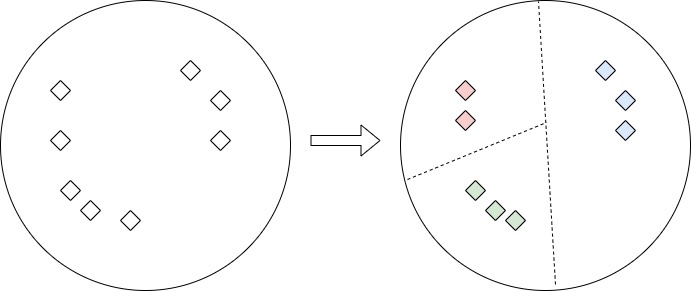
Observando a imagem acima, primeiro temos os filmes que se encaixaram no mesmo grupo do filme ‘toy story’ e abaixo temos a média de avaliações ‘3,34’, com isso podemos dizer que a avaliação média esperada para um filme que se encaixa no mesmo padrão de gênero de ‘toy story’ é de 3,34.

### Dimensionality Reduction; Principal Component Analysis (PCA)

Explorando os temos de dimensionalidades, o primeiro item que chama a atenção é a maldição da dimensionalidade, ou seja, um cálculo matemático que pode gerar infinitas dimensões, por exemplo, em um sistema de recomendações, o vetor de avaliações para cada filme pode representar uma dimensão, ou seja, cada filme possuiria sua própria dimensão, o que pode comprometer recursos computacionais.

Para tratar dessa situação, é feita a redução da dimensionalidade, um dado em grandes dimensões reduzido a uma dimensão menor preservando o máximo de variância que for possível. Uma forma de fazer isso é usando K-Means, se cada filme é a sua própria dimensão, agrupar eles reduzem a dimensão individual para uma dimensão de grupos.

FIGURA 7 – K-MEANS REDUÇÃO DE DIMENSIONALIDADES

 Elaborado pelo próprio autor, 2025.

Outra forma de diminuir a dimensionalidade é através da Análise de Componentes Principais (PCA), ela envolve matemática complexa, mas em termos simples, ela identifica autovetores nos dados em alta dimensão, os quais definem hiperplanos que dividem os dados, preservando ao máximo a variância (diferença) entre eles. Os dados então são projetados nesses hiperplanos que representam dimensões reduzidas.

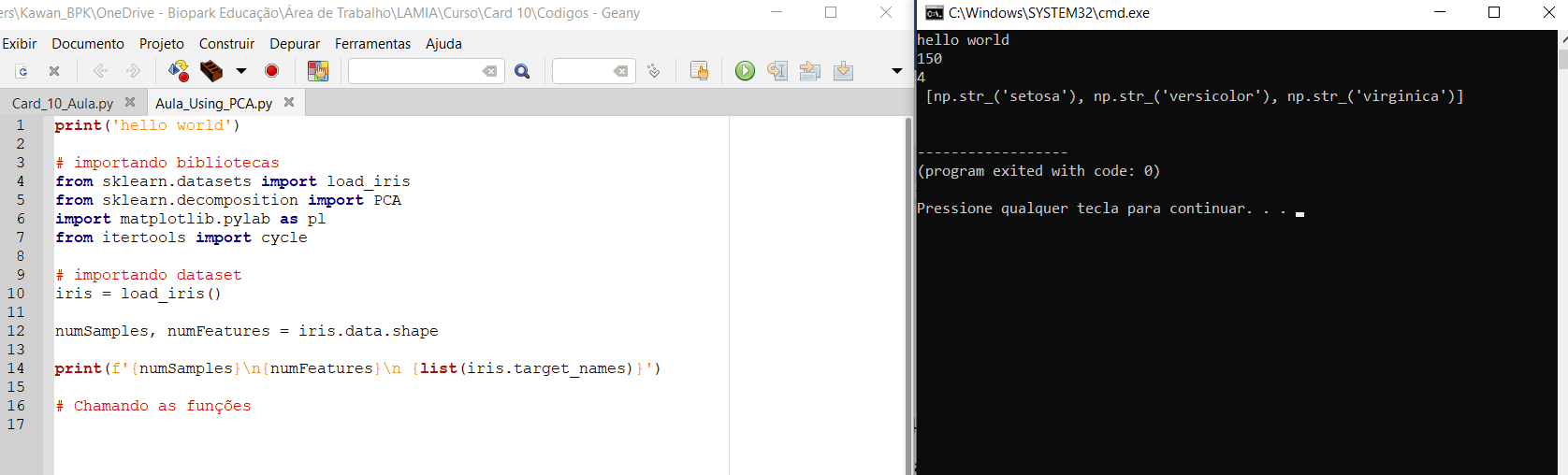
Uma implementação dessa técnica é o algoritmo, Decomposição em Valores Singulares (SVD). Essa técnica é usada para a compressão de imagens e reconhecimento facial. Um exemplo é a visualização de dados do conjunto íris, ela contém dados das pétalas e sépalas de diversas flores de íris.

Nesses dados temos o comprimento e a largura das pétalas e o comprimento e a largura das sépalas. Isso significam 4 dimensões, se quisermos plotar isso em um gráfico 2D ou 3D, precisamos diminuir as dimensões. O PCA reduz as 4 dimensões para 2 dimensões preservando ao máximo a variação, permitindo plotar os dados em um gráfico 2D, mantendo a distinção entre as espécies.

### Activity PCA Example with the Iris data set

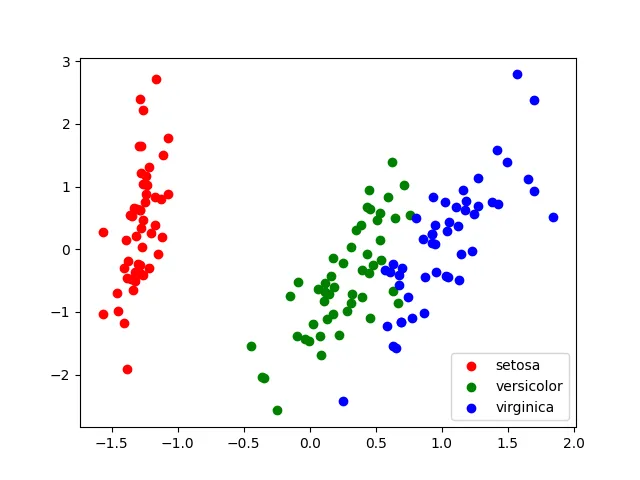
Vamos reduzir a dimensionalidade do dataset ‘iris’ lembrando que temos dados de quatro dimensões, o comprimento e a largura de sépalas e pétalas para quatro tipo de Iris (setosa, versicolor e verginica), com 150 observações.

FIGURA 8 – DATASET IRIS

 Elaborado pelo próprio autor, 2025.

Reduziremos a dimensionalidade dos dados para duas dimensões. Após isso precisamos ver o quanto de variância foi preservada para PC1 e PC2, nesse caso, 0.92 e 0.05, resultando numa variância total preservada de 0.97, isso significa que 97% da informação de quatro dimensões foi preservada em duas dimensões. Se a variância for muito baixa os dados vão apresentar muitos ruídos. Depois plotamos tudo em um gráfico.

FIGURA 9 – GRÁFICO COM DIMENSÕES REDUZIDAS

 Elaborado pelo próprio autor, 2025.

### Data Warehousing Overview ETL and ELT

### Pac-Man Example.html

### Python Markov Decision Process Toolbox.html

### Cat and Mouse Example.html

### Reinforcement Learning

### Activity Reinforcement Learning & Q-Learning with Gym

### Understanding a Confusion Matrix

### Measuring Classifiers (Precision, Recall, F1, ROC, AUC)

## Dealing with Real-World Data

### Measuring Classifiers (Precision, Recall, F1, ROC, AUC)

### Activity K-Fold Cross-Validation to avoid overfitting

### Data Cleaning and Normalization

### Activity Cleaning web log data

### Normalizing numerical data

### Activity Detecting outliers

### Feature Engineering and the Curse of Dimensionality

### Imputation Techniques for Missing Data

### Handling Unbalanced Data Oversampling, Undersampling, and SMOTE

### Binning, Transforming, Encoding, Scaling, and Shuffling

# Conclusões

# Referencias

As referências foram apenas os vídeos da atividade.