Operação Clarividência

EXA864 - Mineração de Dados

Dicentes: Adlla Katarine, Daniel Alves e Ramon Silva
Docente: Rodrigo Tripodi Calumby

Linguagem e ferramentas utilizadas

- Python
- Spyder
- Biblioteca
  - Pandas
  - Sklearn
  - Scipy
  - Matplotlib
- PyQt5

Sistema de Busca de Personagens por Similaridade

Objetivos

- Escolha de um personagem
- Escolha de uma medida de distância
- Seleção de features(superpoderes)
- Ranking com os personagens mais similares e seus respectivos escores

Desenvolvimento

Medidas de distância

3 medidas de distâncias disponíveis com cálculo de <u>similaridade</u>

- Distância de Jaccard

$$\frac{c_{TF} + c_{FT}}{c_{TT} + c_{FT} + c_{TF}}$$

- Distância de Russell-Rao

$$\frac{n - c_{TT}}{n}$$

- Distância de Sokal-Michener

$$\frac{R}{S + R}$$

, $R = 2 * (c_{TF} + c_{FT})$ $S = c_{FF} + c_{TT}$

Demonstração

Predição de super-poderes

Objetivos

- Predizer se um determinado personagem possui ou não um determinado super-poder ou características.
- Fazer uso de Árvore de Decisão.
- Construção de preditores para: Flight, Super Strength, Accelerated Healing, Alignment e Teleportation, como o atributo extra.

Desenvolvimento

Pré-processamento

- Integração das bases de dados;
- Tratamento de valores faltantes e inconsistentes;
- Tratamento de personagens duplicados;
- Exclusão de personagens sem características;
- Renomeamento de hérois com nome iguais;

Pré-processamento

- Agrupamento de classes de alguns atributos;
- Tratamento de NaN, por classe mais frequente de cada categoria;
- Preenchimento de classes ‘-’ por negação do atributo;
- Preenchimento de classes ‘-’ por classe mais frequente de cada categoria;
- Transformação de valores nominais para numéricos;

Treinamento e Teste

- Divisão de dados para treinamento e teste;
- Uma árvore de decisão é criada para cada previsor, CART;
- Uso de GridSearchCV para validação do melhor parâmetro para a árvore;

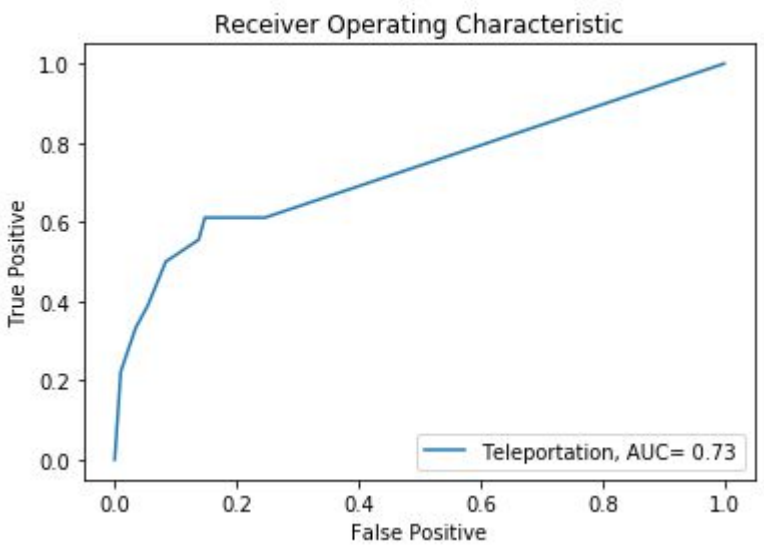
Avaliação dos classificadores e predição

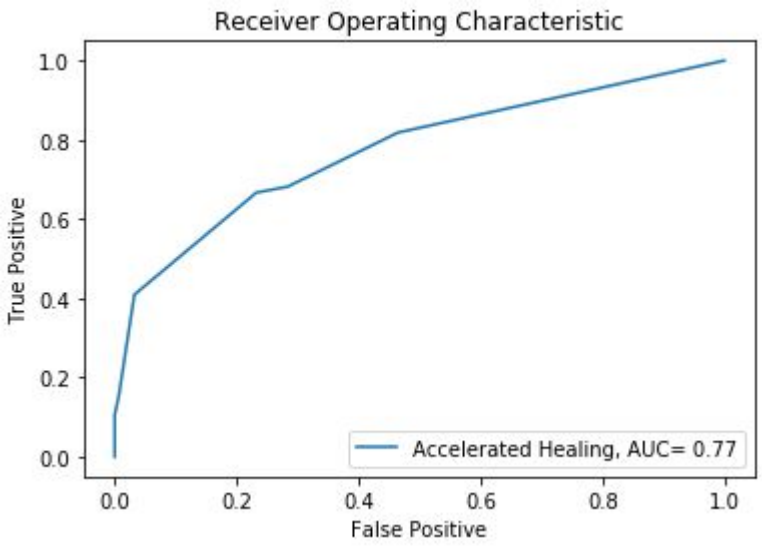
- Acurácia;
- Matriz de confusão;
- Validação cruzada;
- Curva ROC + AUC, Área abaixo da curva;

Análise Experimental

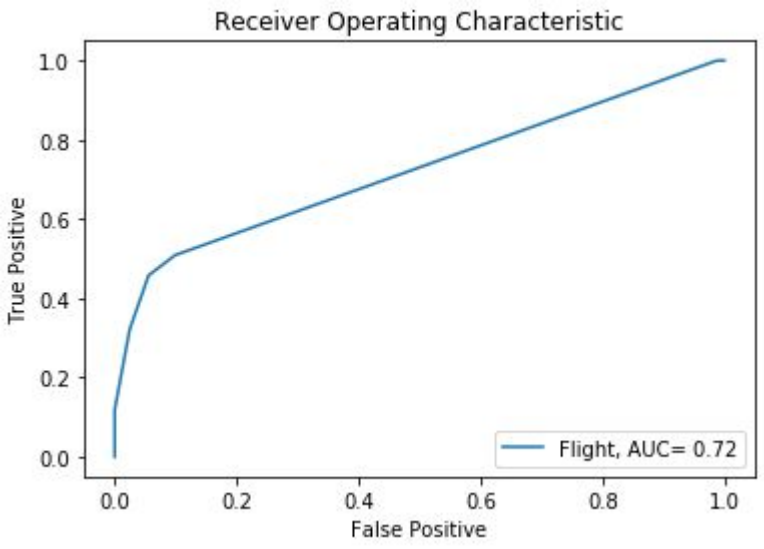
https://docs.google.com/spreadsheets/d/1qNIKBnvCtn4EX5UeICNb2wuQY943pB_PFvKTejs8xOg/edit#gid=0

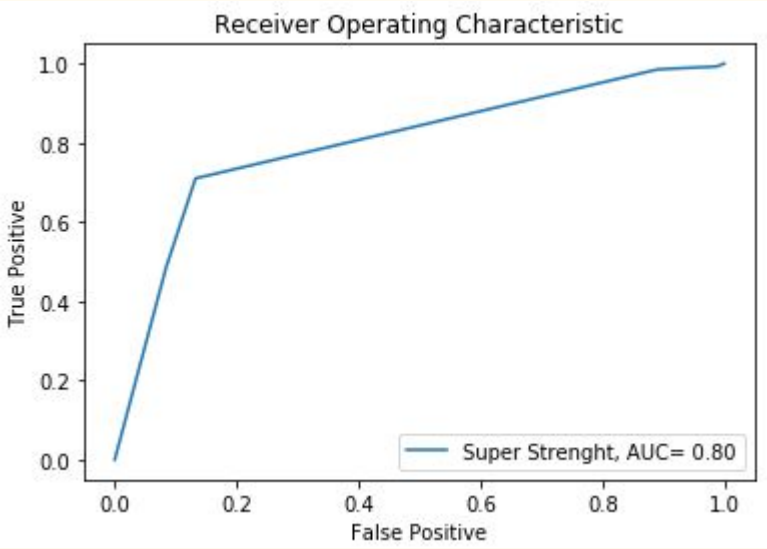
ROC + AUC


Receiver Operating Characteristic
True Positive
False Positive
0.0
0.2
0.4
0.6
0.8
1.0
Teleportation, AUC= 0.73

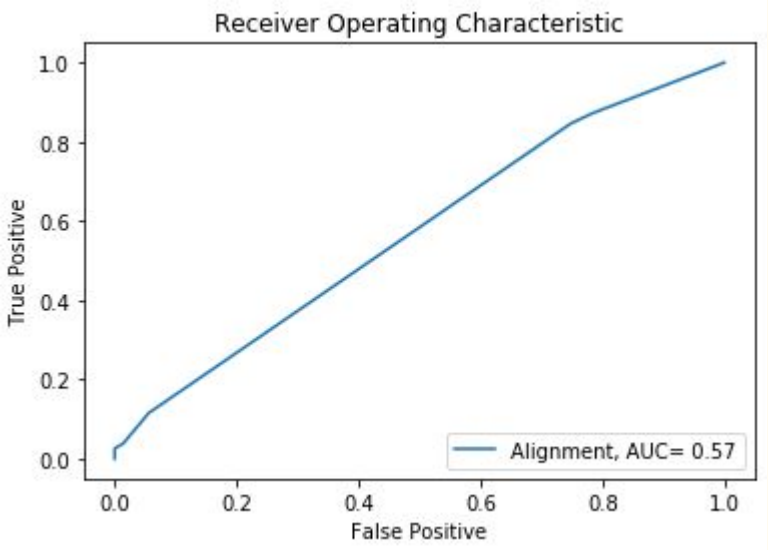

Receiver Operating Characteristic
True Positive
False Positive
0.0
0.2
0.4
0.6
0.8
1.0
Accelerated Healing, AUC= 0.77

ROC + AUC


Receiver Operating Characteristic
True Positive
False Positive
0.0
0.2
0.4
0.6
0.8
1.0
Flight, AUC= 0.72


Receiver Operating Characteristic
True Positive
False Positive
0.0
0.2
0.4
0.6
0.8
1.0
Super Strenght, AUC= 0.80

ROC + AUC


Receiver Operating Characteristic
True Positive
False Positive
0.0
0.2
0.4
0.6
0.8
1.0
Alignment, AUC= 0.57

Conclusão

Todos os requisitos foram atingidos de forma simples e eficaz. Houve dificuldades por ter sido a primeira aplicação do grupo feita em Python. Como a procura e uso dos cálculos de similaridade, em que a biblioteca apresentava a maioria como cálculo de dissimilaridade. E também, o pré-processamento dos atributos nas bases de dados e para uni-las, gerou um contratempo.

Referências

- Granatyr, Jones. Machine Learning e Data Science com Python de A a Z. Disponível em: https://www.udemy.com/machine-learning-e-data-science-com-python-y/.
- Documentação Pandas. Disponível em: https://pandas.pydata.org/pandas-docs/stable/.
- Documentação Scipy Distance. Disponível em: https://docs.scipy.org/doc/scipy/reference/spatial.distance.html.
- Scikit-Learn. Disponível em: https://scikit-learn.org/stable/.
- Matplotlib. Disponível em: https://matplotlib.org.