Video Y

Checkpoint

Trabalho Realizado Por:

Miguel Proença Fernandes Ramalho Amaro, up202106985

António Maria Araújo Pinto dos Santos, up202105587

Índice

Trabalho a Desenvolver……..3

Ferramentas e Algoritmos…….4

Implementação…….5

Pré-processamento dos Dados……6

Análise dos Dados……...7

Algoritmos e Comparação…….8

Conclusão……..9

Referências Bibliográficas……..10

Trabalho a Desenvolver

O trabalho escolhido para desenvolvimento foi a opção Video. Neste trabalho é nos fornecido um dataset que contém informação relativa a aproximadamente 6000 jogos, cujos atributos são:

- id, nome, categoria, número de DLC's, número de expansões, ano de lançamento, número de seguidores, existência de um franchise, género, plataforma, empresas, pontuação média dos utilizadores, qualificação média dos utilizadores, número de reviews, resumo.

A maioria da informação que nos é fornecida já se encontra devidamente formatada, pelo que o foco principal deste projeto será no pré-processamento dos dados (análise e organização do dataset). Assim, será verificada a necessidade de eliminar, agregar ou discretizar atributos; será detetada a existência de dados em falta, procedendo para o seu tratamento (através do uso de médias, frequências ou remoção).

De seguida, passaremos ao treino de um classificador do tipo Supervised Learning. Esta tarefa será concretizada através da seleção dos atributos importantes para a classificação, separando os dados em output (classificação) e input (restantes dados), treinando o classificador de modo a prever as classificações médias dos utilizadores relativas aos jogos.

Ferramentas e Algoritmos

Sendo este um trabalho relacionado com Machine Learning, a linguagem de programação indicada para a sua realização é o Python, uma vez que disponibiliza uma grande variedade de bibliotecas relacionadas com esta área. Como tal, iremos utilizar, entre outras, as seguintes bibliotecas:

- Pandas, com o propósito de visualizar e manipular DataFrames importando o dataset “video_games.csv”;
- SciKit-Learn, que permite utilizar algoritmos de classificação e funções de pré-processamento de dados;
- MatPlotLib, de modo a serem criados tabelas e gráficos com o intuito de mais facilmente visualizar os dados e comparar os algoritmos usados.

A importação da biblioteca SciKit-Learn permite o uso intuitivo de algoritmos de Supervised Learning, bem como o ajuste dos parâmetros dos mesmos. Iremos, através desta biblioteca, implementar os seguintes algoritmos:

- “Decision Trees”, baseado na colocação de uma série de perguntas e classificação do input de acordo com as respostas dadas;
- “K-Nearest Neighbors”, baseado no agrupamento de inputs de acordo com as suas características, classificando-os de acordo com os seus vizinhos.

Este trabalho irá ser desenvolvido no ambiente Jupyter Notebook, de modo a poder visualizar plots e gráficos durante e após a programação. É, assim, criado um programa de fácil interpretação e modificação.

Implementação

O trabalho encontra-se dividido:

- **Libraries** – importação das bibliotecas a serem usadas ao longo do trabalho;
- **Pré-processamento dos Dados** – leitura do dataset e tratamento dos dados de modo a poderem ser interpretados por um algoritmo preditivo;
- **Análise dos Dados** – desenho de plots e gráficos de modo a visualizar os dados e interpretar possíveis relações entre os mesmos**;**
- **Separação dos Dados** – separação dos dados em variáveis independentes e variáveis dependentes;
- **Decision Trees** – implementação do algoritmo DecisionTreeClassifier da biblioteca sklearn;
- **K-Nearest Neighbors** – implementação do algoritmo KNeighborsClassifier da biblioteca sklearn;
- **Comparação dos Algoritmos** – obtenção de dados relacionados com as previsões de ambos os modelos e desenho de matrizes e gráficos para facilitar a comparação dos dados;
- **Extra** – implementação de outros algoritmos.

Pré-processamento dos Dados

Nesta etapa do programa começamos por ler o dataset e selecionar as colunas que contêm informação pertinente para a nossa tarefa. Assim, eliminamos as colunas “id”, “name”, “user_score” e “summary”.

De seguida averiguamos a quantidade de dados em falta. Ao constatar que se tratava de uma quantidade diminuta de dados em falta, optamos por os eliminar na sua totalidade, uma vez que não afetará o resultado final.

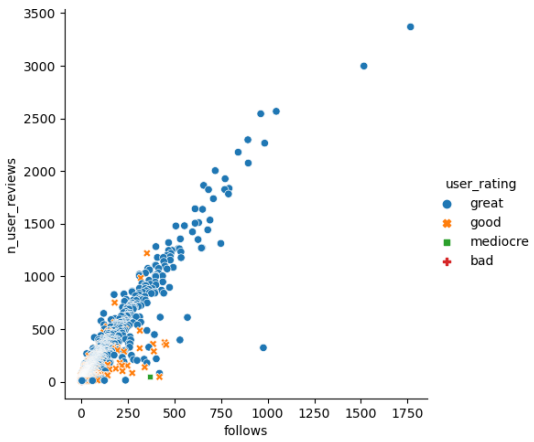
Tendo, agora, apenas os dados necessários, procedemos para a transformação dos valores categóricos (“category” e “in_franchise”) em valores numéricos, de modo a poderem ser interpretados pelo algoritmo.

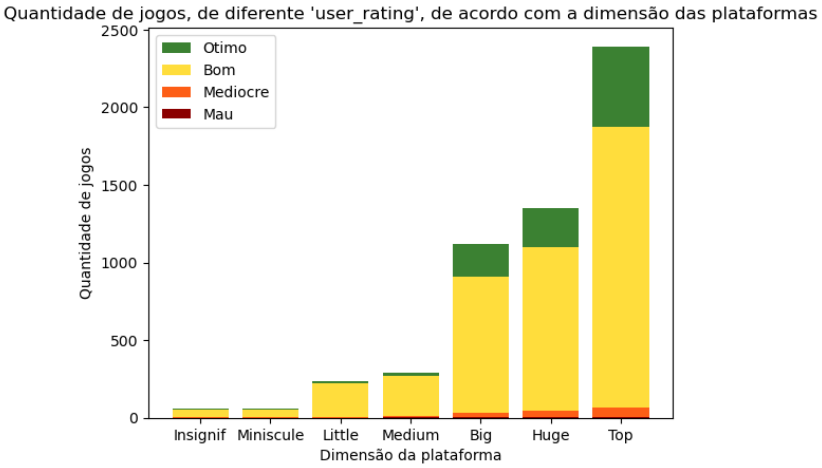
Por fim, separamos as colunas que continham demasiada informação (“data_genres”, “companies” e “platforms”) em colunas binárias dedicadas a cada informação, e agrupando-as de acordo com a sua importância.

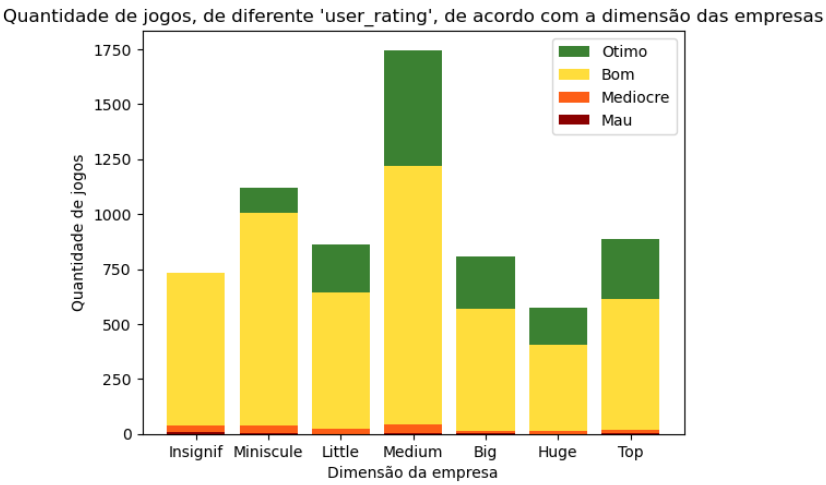
Análise dos Dados

Após extrairmos os dados necessários do dataset já processado, construímos os gráficos abaixo apresentados, sendo possível evidenciar alguns padrões, como por exemplo

- todos os jogos acima de 500 “follows” e 1300 “n_user_reviews” são classificados como great;
- existe um aumento no número de jogos com o aumento da dimensão das plataformas, porém, a proporção da quantidade de jogos de diferente classificação é semelhante para todas as dimensões de plataformas;
- empresas médias fabricam a maior quantidade de jogos.

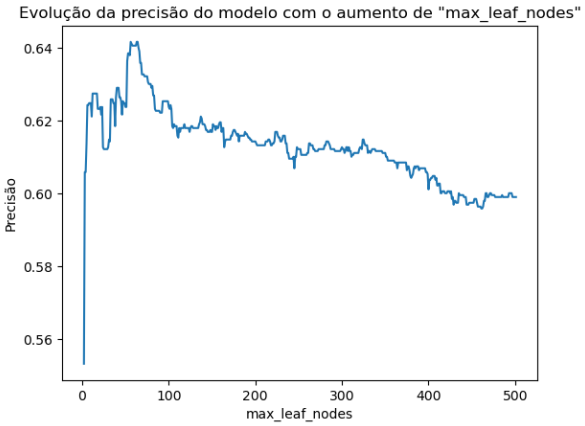


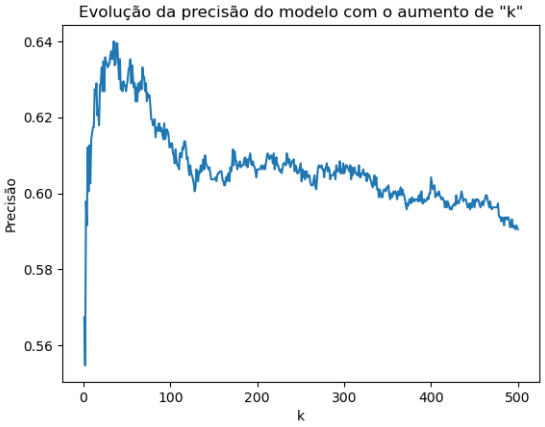




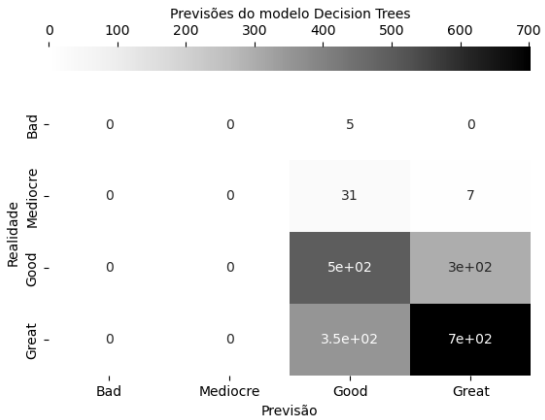
Algoritmos e Comparação

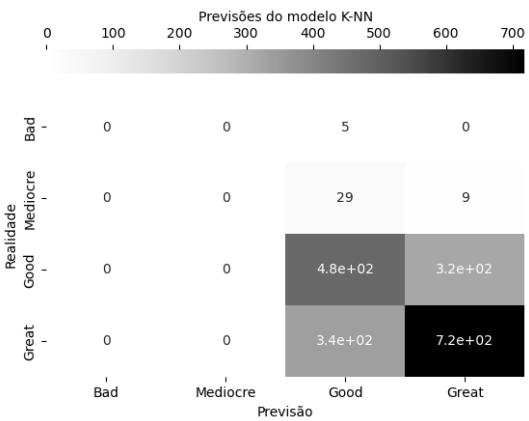
Os algoritmos utilizados neste trabalho forma implementados diretamente da biblioteca sklearn, e os seus atributos foram modificados numa tentativa de obter os melhores resultados possíveis. Para tal, construímos 2 gráficos que permitem visualizar o valor da “accuracy” em função da alteração dos parâmetros dos algoritmos. São:





Através da leitura dos gráficos acima, podemos concluir que, no melhor dos casos, a “accuracy” de ambos os algoritmos ronda dos 65%. De forma a melhor visualizar onde os algoritmos cometeram erros criamos o “confusion matrix” de cada algoritmo. Temos:





Conclusão

Após uma análise dos resultados, verificamos que os dois algoritmos obtiveram uma “accuracy” média um pouco abaixo do desejado. Todavia, com uma leitura dos gráficos criados a partir da extração dos dados do dataset, é possível concluir que, embora sejam exibidos certos padrões, não é possível estabelecer fortes relações entre os dados fornecidos e os resultados, justificando a “accuracy” não muito alta.

De qualquer modo, este projeto contribuiu para uma forte melhoria na compreensão de manipulação de dados, bem como na importância da criação de suportes visuais para compreender os dados com que trabalhamos.

Referências Bibliográficas

https://scikit-learn.org/stable/tutorial/index.html

https://towardsdatascience.com/an-introduction-to-pandas-in-python-b06d2dd51aba

https://archive.ics.uci.edu/ml/index.php

https://www.youtube.com/playlist?list=PLzMcBGfZo4-mP7qA9cagf68V06sko5otr

https://www.youtube.com/watch?v=HMSwljTtAlc

https://www.youtube.com/watch?v=Wqmtf9SA_kk

https://www.youtube.com/watch?v=sgQAhG5Q7iY

https://www.youtube.com/watch?v=wxS5P7yDHRA

https://www.youtube.com/watch?v=XmSlFPDjKdc

https://www.youtube.com/watch?v=pqNCD_5r0IU

https://www.youtube.com/watch?v=0Lt9w-BxKFQ