Ângelo Ferreira

- Divisão do Código em vários ficheiros: É boa prática, nomeadamente quando o código é grande e necessita de ser feito um bom debug. No entanto, poderia ter ligado os vários scripts de forma a ser possível correr uma vez e não correr cada ficheiro de forma individual.
- Código comentado: É sempre boa prática ter o código comentado, nomeadamente quando este é partilhado para outras pessoas.
- É determinada a matrix de confusão. É algo importante para verificar a validade do modelo. No entanto poderia ter sido adicionado o cálculo da precisão do modelo. Aqui, existem várias formas de o fazer:
  https://scikit-learn.org/stable/modules/classes.html#module-sklearn.metrics
- Para facilitar a compreensão do problema/solução poderiam ter sido elaborados gráficos (matplolib ou seaborn)
- No feature selection apresentou mais que um método o que é bom para validar resultados. No entanto, não foi apresentado (através de gráficos/tabelas) as características de cada um
- Foi só utilizado um modelo na classificação. Outros modelos poderiam ter sido utilizados. Existem alguns exemplos no seguinte documento:
  https://drive.google.com/drive/folders/1Lx6v0MW2FauVmwCTOLfxWs3IyObYBish
- Foi realizado uma análise superficial aos dados em bruto o que é útil. Para a realização de uma análise mais profunda poderá ser utilizado, por exemplo o comando da biblioteca pandas pd.descrive()
  https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.describe.html
- O código em cada script está bem estruturado e simples, o que é uma boa prática.
- Uma vez que os dados em bruto não estavam balanceados, deveria ter sido realizada um balanceamento dos mesmos. Aqui, pode ser usada a biblioteca imblearn. Um exemplo pode ser encontrado abaixo:
  https://stackoverflow.com/questions/55814015/over-sampling-class-imbalance-train-test-split-found-input-variables-with-incon?rq=1

De forma geral, o projeto está bem estruturado e cumpre os objetos principais.