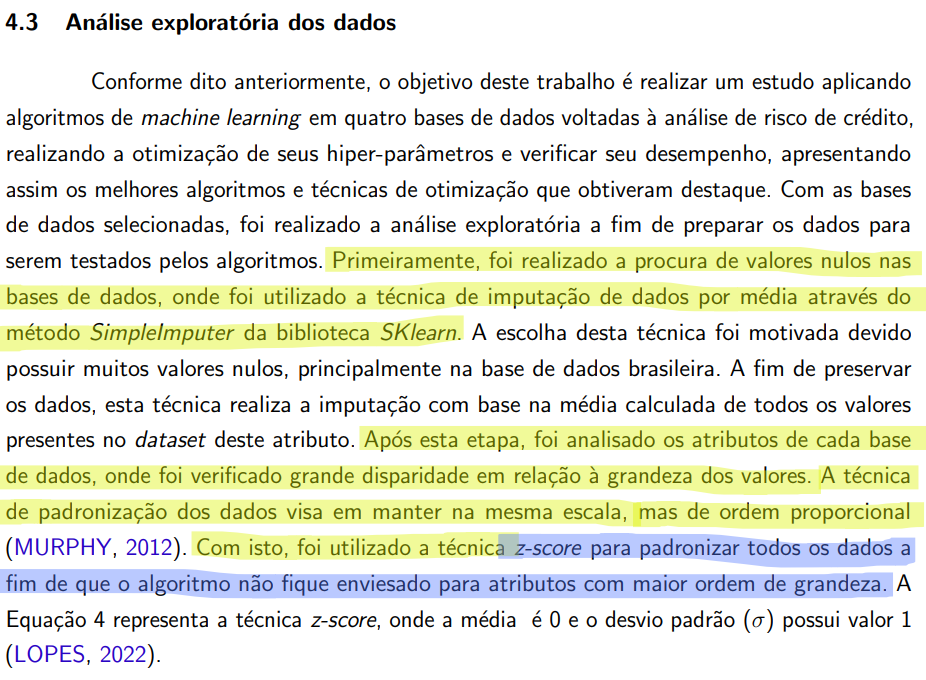
Observações base australiana

## Tratamento feito SILVA(2022) TCC

* Foi utilizado *SimpleImputer* para tratamento de valores nulos, porém na base australiana conforme a própria documentação, valores nulos foram já tratados.
* Foi feito o escalonamento dos dados para deixá-los na mesma escala utilizando *z-score*. PORÉM de acordo com documentações analisadas, StandardScaler por exemplo só funcionaria para distribuições normais
  + Como foi feito o escalonamento para variáveis categóricas?
  + Lembrando que a padronização consiste em redimensionar os dados de forma que sua média se torne 0 e desvio padrão 1.



## 

## Tratamento de categorias “raras” -> Após execução do OHE sem realizar nenhum agrupamento de categorias, notou-se um comportamento estranho, fazendo a contagem do n de registros para cada categoria sobre as novas colunas, por exemplo:

Coluna: A5, Valores: A5

8 146

11 78

9 64

3 59

6 54

1 53

4 51

13 41

7 38

14 38

2 30

10 25

5 10

12 3

Essas últimas 4 contagens após OHE geraram os dados da seguinte forma:

Coluna: A5\_2, Valores: A5\_2

0 660

2 30

Coluna: A5\_3, Valores: A5\_3

0 675

10 25

Coluna: A5\_12, Valores: A5\_12

0 687

10 3

Ou seja, as novas colunas ficaram muito esparsas, e para resolver esse problema, foi criada uma categoria responsável por agrupar frequências de categorias menores que 5%. Desta forma, a coluna A\_5 após normalização ficou assim:

Coluna: A5, Valores: A5

8 146

11 78

15 68

9 64

3 59

6 54

1 53

4 51

13 41

14 38

7 38

Com a categoria “15” (equivalente a “Outros”) agrupando os valores das categorias 2, 10, 5 e 15 e após OHE, gerando apenas uma nova coluna A5\_15, reduzindo tanto dimensionalidade quanto outliers falsos positivos.

## One Hot Encoding

* Aplicar OHE sobre variáveis categóricas com mais de uma opção (A4, A5, A6, A12) dado que possuem mais de duas opções e assim, refletir com realidade sua disposição sobre o Dataset.

## Detecção e Tratamento de Outliers

* Devido a base ser pequena, não é interessante excluir os outliers identificados dado que isso pode impactar significativamente no poder de previsibilidade do modelo
* Teste com dois métodos de detecção de outliers para compreender se o número de anomalias identificada possui muita diferença
  + Detecção por Desvio Padrão

Coluna: A1, Outliers: 0

Coluna: A2, Outliers: 0

Coluna: A3, Outliers: 0

Coluna: A4, Outliers: 383

Coluna: A5, Outliers: 0

Coluna: A6, Outliers: 0

Coluna: A7, Outliers: 0

Coluna: A8, Outliers: 0

Coluna: A9, Outliers: 0

Coluna: A10, Outliers: 0

Coluna: A11, Outliers: 0

Coluna: A12, Outliers: 690

Coluna: A13, Outliers: 0

Coluna: A14, Outliers: 0

Coluna: A15, Outliers: 0

* + Detecção por IQR:

Coluna: A1, Outliers: 0

Coluna: A2, Outliers: 18

Coluna: A3, Outliers: 17

Coluna: A4, Outliers: 165

Coluna: A5, Outliers: 0

Coluna: A6, Outliers: 215

Coluna: A7, Outliers: 63

Coluna: A8, Outliers: 0

Coluna: A9, Outliers: 0

Coluna: A10, Outliers: 79

Coluna: A11, Outliers: 0

Coluna: A12, Outliers: 65

Coluna: A13, Outliers: 13

Coluna: A14, Outliers: 113

Coluna: A15, Outliers: 0

* Detecção por Z-score com valor absoluto de Z = 3 (ou seja, caso Z > 3, o dado está a 3 desvio padrões de distância da média)

Coluna: A1, Outliers: 0

Coluna: A2, Outliers: 9

Coluna: A3, Outliers: 10

Coluna: A4, Outliers: 0

Coluna: A5, Outliers: 0

Coluna: A6, Outliers: 0

Coluna: A7, Outliers: 22

Coluna: A8, Outliers: 0

Coluna: A9, Outliers: 0

Coluna: A10, Outliers: 8

Coluna: A11, Outliers: 0

Coluna: A12, Outliers: 65

Coluna: A13, Outliers: 9

Coluna: A14, Outliers: 6

Coluna: A15, Outliers: 0

* Detecção por DBSCAN(eps=1000, min\_samples=10):

Coluna: A1, Outliers: 0

Coluna: A2, Outliers: 0

Coluna: A3, Outliers: 0

Coluna: A4, Outliers: 0

Coluna: A5, Outliers: 0

Coluna: A6, Outliers: 0

Coluna: A7, Outliers: 0

Coluna: A8, Outliers: 0

Coluna: A9, Outliers: 0

Coluna: A10, Outliers: 0

Coluna: A11, Outliers: 0

Coluna: A12, Outliers: 0

Coluna: A13, Outliers: 0

Coluna: A14, Outliers: 18

Coluna: A15, Outliers: 0

* Detecção por SOM (Rede Neural)

[Untitled0.ipynb](https://colab.research.google.com/drive/1gIFzXrjHxQPSkIY7nT7_g6OIMbk-YjYs?usp=sharing)

* O gpt sugeriu de talvez utilizar identificação por Isolation Forest. <https://medium.com/@gabrielpbreis/outliers-como-definir-detectar-e-tratar-parte-2-5240149f8f98>

olhar esse link

* K-fold validation:
  + Como apresentado nos trabalhos, foi utilizado a técnica de validação cruzada por k-fold. No dataset australiano, o valor de k, ou seja, o valor de folds (subamostras) que dividiram os dados, foi igual a 10.
  + <https://medium.com/learning-data/an-intuitive-guide-to-understanding-cross-validation-abd01bf5e879> (sobre k-fold cross-validation)

números totais com e sem validação cruzada:

Sem k-fold

Acurácia : 0.8502415458937198, Precisão : 0.8531763171834889, Recall : 0.8502415458937198, F1 : 0.8507354605577844

precision recall f1-score support

0.0 0.89 0.84 0.86 117

1.0 0.80 0.87 0.83 90

accuracy 0.85 207

macro avg 0.85 0.85 0.85 207

weighted avg 0.85 0.85 0.85 207

Com k-fold

Acurácia : 0.8357487922705314, Precisão : 0.8422686294647952, Recall : 0.8357487922705314, F1 : 0.8338038772821382

precision recall f1-score support

0.0 0.80 0.92 0.86 111

1.0 0.89 0.74 0.81 96

accuracy 0.84 207

macro avg 0.85 0.83 0.83 207

weighted avg 0.84 0.84 0.83 207

Isso implica que na verdade o modelo ainda está “fraco” em determinadas métricas, como por exemplo, a taxa de 74% de *recall* para a classe positiva. Esse número pode ser preocupante nesse ponto, dado que um dos objetivos é ter um bom *recall* para diminuir FP.

Otimização de Hiperparâmetros

* Interessante notar que a validação-cruzada é passada como parâmetro dos algoritmos de otimização
* Otimização por algoritmo genético
  + Quando criar sua própria função fitness, de acordo com a lib PyGAD, a função deve receber 3 parâmetros, sendo eles:
    - Uma instância da classe [pygad.GA](http://pygad.ga), a qual pode-se utilizar ferramentas para ajudar a obter o fitness que será retornado;
    - A solução(ões) para calcular o valor(es) fitness;

. A solução são os valores de determinado indivíduo. Por essa linha de raciocínio, podemos pensar em maximizar o valor de revocação sobre determinada configuração do modelo.

* + - O índice de solução de determinada população.
* Um ponto importante sobre a solução é que os valores contidos nela serão de acordo com a quantidade de genes determinada no momento de instanciar a classe GA
* OBJETIVO PARA OTIMIZAÇÃO: Hiperparâmetros que tiverem o melhor *recall* serão os escolhidos para a tarefa.
  + Dado que posso utilizar validação cruzada dentro da otimização, vou implementar *k-fold* e ver se consigo encontrar o melhor resultado
* O primeiro trabalho cita a utilização de validação cruzada aninhada e pesquisando sobre, é um tipo de validação “dupla” que utiliza validação cruzada e seu principal objetivo é evitar que os mesmos dados que foram utilizados para buscar os melhores hiperparâmetros, sejam reutilizados para encontrar o modelo com as melhores métricas, problema conhecido como *data leakage*
* para o modelo Random Forest com AG, foi definido como hiperparâmetros que serão otimizados: n\_estimators, max\_depth, min\_samples\_split, min\_samples\_leaf, criterion. Esses são os principais para um bom desempenho do modelo
* é possível definir um range de valores possíveis para cada gene que o indivíduo terá