Exploração de Dados
Ciência de Dados para a Segurança

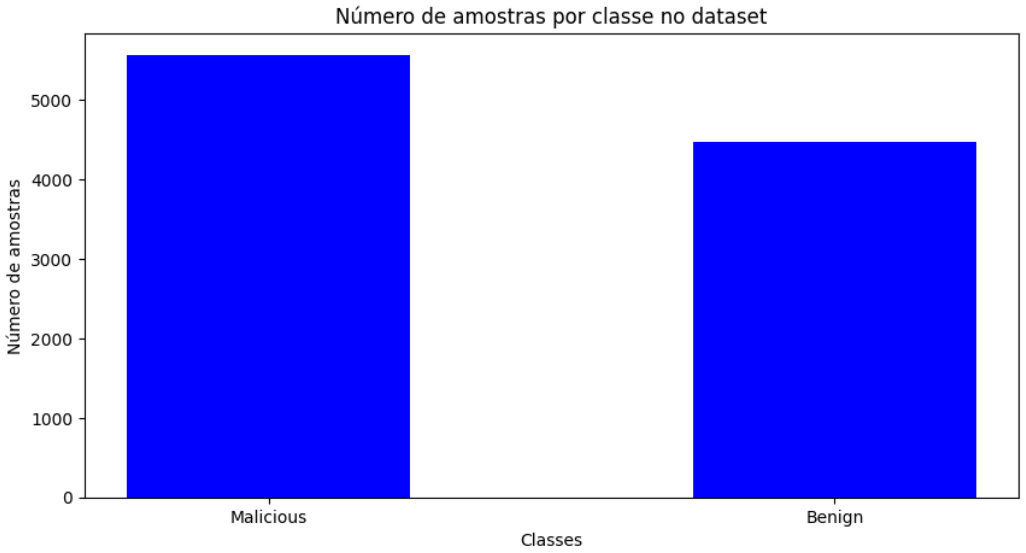
Vinicius Gabriel Machado - GRR20182552

A partir da exploração do dataset PDFMalware2022, o seguinte vetor de características foi extraído:

pdfsize
metadata size
xref Length
title characters
images
text
header
obj
endobj
stream
endstream
xref
trailer
startxref
ObjStm
JS
Javascript
AA
OpenAction
Acroform
JBIG2Decode
RichMedia
launch
XFA,
Colors
classe

A última, classe, indica a classificação da amostra em um espaço binário. Assumindo os valores Malicious (malicioso) e Benign (benigno). O primeiro indica amostras maliciosas que tiveram características parecidas com as benignas na classificação feita na criação do dataset. E a segunda, amostras benignas que tiveram características que as aproximaram de maliciosas.

A distribuição dos dados é a seguinte:



Segue abaixo o código que deu origem a esta figura, implementado em pyhon:

```
1.  #!/usr/bin/python
2.
3.  import sys
4.  import csv
5.  import numpy as np
6.  import matplotlib.pyplot as plt
7.
8.  classname = "Class"
9.  classes = ["Malicious", "Benign"]
10. count = [0, 0]
11.
12. def main(filename, delim):
13.     # abre e le o arquivo pelo nome da coluna
14.     file = open(filename, encoding="utf-8")
15.     csvreader = csv.DictReader(file, delimiter=delim)
16.
17.     # conta o numero de instancias de cada classe
18.     for row in csvreader:
19.         if (row[classname] == classes[0]):
20.             count[0] = count[0] + 1
21.         if (row[classname] == classes[1]):
22.             count[1] = count[1] + 1
23.
24.     # plota o grafico de barras
25.     fig = plt.figure(figsize = (10, 5))
26.     plt.bar(list(classes), list(count), color ='blue', width = 0.5)
27.     plt.xlabel("Classes")
28.     plt.ylabel("Número de amostras")
29.     plt.title("Número de amostras por classe no dataset")
30.     plt.savefig("figure.png")
31.
32.
```

```
33.
34. if __name__ == "__main__":
35.     if (len(sys.argv) != 3):
36.         print("Error - Correct usage:")
37.         print(str(sys.argv[0]), " <FILE> <DELIMITER>")
38.         sys.exit()
39.
40.     main(str(sys.argv[1]), str(sys.argv[2]))
```

A execução foi feita com a seguinte linha de comando:

python3 plot.py dados.csv ,

O primeiro parâmetro indica o nome do arquivo de dados (um csv) e a segunda o delimitador utilizado no arquivo.