# Relatório de E.T.

## Relatório nº4 - *Automatic Classification*

**Curso:** METI
**Turno:** 3ª feira 08:00 > 09:30
**Grupo:** Bancada 8

**Trabalho realizado por:**

Luís Pereira, nº77984
Ruben Condesso, nº 81969

# Índice

1. **Introdução.....2**
2. **Estrutura do código.....3**
3. **Resultados obtidos.....4**
4. **Discussões de resultados .....6**

# 1. Introdução

O objetivo deste laboratório é testar a identificação automática de padrões de tráfego, de modo a identificar fluxos p2p, numa rede específica. Para a realização deste trabalho, implementámos um algoritmo de aprendizagem automática para avaliar a capacidade de generalização do classificador *Naive Bayes*, quando aplicado ao problema dado pelo enunciado sendo que, desta forma, a identificação de tráfego é realizada por um simples classificador baseado em *machine learning*.

A aprendizagem e generalização referidas foram feitas em 4 cenários diferentes, por conseguinte, cada um deles terá diferentes amostras de tráfego da rede.

Passaremos, de seguida, a explicar qual o problema em causa deste trabalho, como estruturamos o nosso código de forma a ir ao encontro da solução pretendida, e mostraremos as taxas de erro referentes a cada classe. Com base nos resultados obtidos, iremos retirar as respetivas conclusões, nomeadamente, avaliar a exatidão do classificador automático, dadas as taxas de erro calculadas.

# 2. Estrutura do código

O objetivo específico do classificador automático a implementar passa pela sua capacidade de prever o "uso" injusto de largura de banda, por parte de um IP de uma rede, devido ao enorme uso de ligações *p2p.* São fornecidos 8 *data files* com informação extraída de uma análise aos *flows* de uma rede, onde existem 4 *labeled data files* classificados com diversos campos sendo o último o mais relevante para o caso uma vez que se apresenta classificado como *p2p* ou *not p2p.* Temos então de classificar os restantes 4 *data files*, tendo por base a previsão gerada pelo classificador baseado em *Naive Bayes*.

Em primeiro lugar, foi necessário ler os dados de entrada presente em cada *data file (*função *parse*), onde é criada uma matriz m que terá, em cada coluna, os parâmetros existentes para as respetivas linhas do *data file* em causa. Depois essa matriz m é "baralhada", criando uma matriz aleatória que só difere da matriz original na ordem das linhas existentes (função *random_matrix*).

Posto isto, são criadas duas matrizes ( a partir da matriz m): a matriz *matrix_learn* e a matriz *matrix_run*. A primeira corresponderá à matriz onde o classificador irá fazer a sua aprendizagem, e a segunda matriz corresponderá às previsões do classificador, segundo essa aprendizagem. Logo a *matrix_learn* percorre apenas a primeira metade da matriz original, e a *matrix_run* irá ter a previsão para a outra metade.

Relativamente à aprendizagem da *matrix_learn*, é feita uma filtragem do último parâmetro (função *filter_by_column*). Ou seja, fazer uma divisão dos casos em que existe ligação *p2p* com os casos em que não existe.

Em relação ao uso do classificador de *Naive Bayes*, usamos a probabilidade condicionada para verificar se devemos classificar como *p2p* ou *not p2p* cada ligação. Se essa probabilidade for superior para o caso *p2p*, então iremos classificar o fluxo como *p2p*, caso contrário iremos classificar como *not p2p*, onde iremos aplicar este raciocino na matriz *matrix_run*, referida anteriormente. Desta forma, iremos fazer depois a verificação da classificação, e logo demonstrar os resultados obtidos bem como os vários tipos de erros existentes.

Por último, iremos escrever nos ficheiros *unlabeled*, os resultados obtidos pelo classificador *Naive Bayes*, numa pasta denominada como *results*.

# 3. Resultados obtidos

Passaremos agora à demonstração dos resultados que obtivemos. Para cada classificação feita calculámos a taxa de erro (*false negative + false positive)* existente na mesma, e analogamente calculámos a taxa de sucesso (*true positive+ true negative)*.

De forma a obter resultados mais consistentes, e por conseguinte retirar conclusões mais precisas, fizemos 10 simulações para as classificações originadas, para os 4 *unlabeled data files*. Iremos mostrar a média para cada taxa, e como variou os máximos e mínimos ao longo das simulações feitas.

Referente ao ficheiro de entrada *1-labeled.dat* os resultados obtidos foram os seguintes:

```
>> simulacaoTaxaErros(10)
Media da Taxa de erro:
    0.2467

Media da Taxa de sucesso:
    0.7533

Media da Taxa de falsos negativos:
   8.4000e-04

Taxa de falsos positivos:
    0.2458

Valores gerados da taxa de erro:
    0.2382    0.2518    0.2502    0.2464    0.2438    0.2468    0.2442    0.2478    0.2430    0.2544

Valores gerados da taxa de sucesso:
    0.7618    0.7482    0.7498    0.7536    0.7562    0.7532    0.7558    0.7522    0.7570    0.7456

Valores gerados da taxa de falsos negativos:
    0.0014         0    0.0006    0.0016    0.0008    0.0022    0.0006         0    0.0012         0

Valores gerados da taxa de falsos positivos:
    0.2368    0.2518    0.2496    0.2448    0.2430    0.2446    0.2436    0.2478    0.2418    0.2544
```

Figura 1: Taxas de erros e de sucesso
com 10 simulações, para o *1-labeled.dat*

Para o ficheiro *2-labeled.dat*, obtivemos o seguinte:

```
>> simulacaoTaxaErros(10)
Media da Taxa de erro:
    0.0812

Media da Taxa de sucesso:
    0.9188

Media da Taxa de falsos negativos:
    0.0024

Taxa de falsos positivos:
    0.0788

Valores gerados da taxa de erro:
    0.0828    0.0598    0.0926    0.0650    0.0846    0.0782    0.0802    0.0932    0.0886    0.0874

Valores gerados da taxa de sucesso:
    0.9172    0.9402    0.9074    0.9350    0.9154    0.9218    0.9198    0.9068    0.9114    0.9126

Valores gerados da taxa de falsos negativos:
    0.0028    0.0028    0.0020    0.0026    0.0014    0.0018    0.0030    0.0032    0.0034    0.0014

Valores gerados da taxa de falsos positivos:
    0.0800    0.0570    0.0906    0.0624    0.0832    0.0764    0.0772    0.0900    0.0852    0.0860
```

Figura 2: Taxas de erros e de sucesso
com 10 simulações, para o 2-*labeled.dat*

Agora para o *3-labeled.dat*, obtivemos os seguintes resultados:

```
>> simulacaoTaxaErros(10)
Media da Taxa de erro:
   6.2000e-04

Media da Taxa de sucesso:
    0.9994

Media da Taxa de falsos negativos:
     0

Taxa de falsos positivos:
   6.2000e-04

Valores gerados da taxa de erro:
    0.0004         0         0    0.0004    0.0008         0    0.0008    0.0004    0.0034         0

Valores gerados da taxa de sucesso:
    0.9996    1.0000    1.0000    0.9996    0.9992    1.0000    0.9992    0.9996    0.9966    1.0000

Valores gerados da taxa de falsos negativos:
     0     0     0     0     0     0     0     0     0     0

Valores gerados da taxa de falsos positivos:
    0.0004         0         0    0.0004    0.0008         0    0.0008    0.0004    0.0034         0
```

Figura 3: Taxas de erros e de sucesso
com 10 simulações, para o 3-*labeled.dat*

Finalmente, para o *4-labeled.dat* obtivemos os seguintes resultados:

```
>> simulacaoTaxaErros(10)
Media da Taxa de erro:
    0.1593

Media da Taxa de sucesso:
    0.8407

Media da Taxa de falsos negativos:
   4.0000e-04

Taxa de falsos positivos:
    0.1589

Valores gerados da taxa de erro:
    0.1638    0.1552    0.1538    0.1646    0.1500    0.1626    0.1612    0.1668    0.1594    0.1560

Valores gerados da taxa de sucesso:
    0.8362    0.8448    0.8462    0.8354    0.8500    0.8374    0.8388    0.8332    0.8406    0.8440

Valores gerados da taxa de falsos negativos:
         0    0.0002    0.0008    0.0002    0.0016         0         0    0.0002    0.0004    0.0006

Valores gerados da taxa de falsos positivos:
    0.1638    0.1550    0.1530    0.1644    0.1484    0.1626    0.1612    0.1666    0.1590    0.1554
```

Figura 4: Taxas de erros e de sucesso
com 10 simulações, para o 4-*labeled.dat*

Como foi referido anteriormente, segue na pasta *results*, os resultados obtidos pelo classificador *Naive Bayes*, relativamente ao preenchimento dos ficheiros *unlabaled*, ou seja, classificando como *2p2* ou *not p2p*, cada ligação.

# 4. Discussões de resultados

Passando agora para a discussão dos resultados obtidos no ponto 3, iremos analisar a exatidão do classificador automático baseando-nos nas taxas de erros calculadas nas figuras 1, 2, 3 e 4. Vamos ter em consideração a seguinte tabela (retirada dos *slides* do professor), como auxílio de raciocínio:

| Class output / Input | X | $\overline{X}$ |
|---|---|---|
| X | *tp* | *fn* |
| $\overline{X}$ | *fp* | *tn* |

tp - true positives, fn - *false negatives*, tb - *true negatives*, fp - *false positives*

Vamos também ter em conta a seguintes fórmula, de forma a justificar a discussão de resultados:

$$Accuracy = 1 - E_{rate};$$

Aplicando a fórmula da exatidão aos resultados obtidos para os respetivos *data files*, obtemos o seguinte:

| *Data_file* | Exatidão |
|---|---|
| 1-*labeled.dat* | 0.7533 |
| *2-labeled.dat* | 0.9188 |
| 3-*labeled.dat* | 0.9994 |
| 4-*labeled.dat* | 0.8407 |

Podemos verificar que a exatidão do classificador automático ainda varia consideravelmente, entre os diferentes *data files*. Se consideramos uma boa exatidão acima dos 85%, verificamos que o classificador fez uma previsão positiva do uso "injusto" de largura banda para os *data files* 2 e 3, sendo que o *data file* 4 aproxima-se bastante (84,07%), inversamente, o *data file* 1 encontra-se relativamente afastado (75,33%), onde aqui temos de considerar que a previsão não foi exata.

Considerando a figura 1, um resultado que se destaca dos demais é o valor da taxa de falsos positivos (0.2458). O facto deste valor ser demasiado elevado, levará a que seja previsto em demasia a existência de tráfego "justo", pois irá haver muitos casos classificados como ligação *p2p*, onde na verdade deveriam ser classificados inversamente. Na figura 4, podemos ver que o valor da taxa de falsos positivos também é consideravelmente alto, apesar de ser mais baixo do que relativamente ao valor presente na figura 1

Em todos os casos, os valores das taxas de falsos negativos são muito reduzidos, por conseguinte, concluímos que não existe uma grande ineficácia na previsão, na medida que não existem muitos casos onde a ligação é *p2p*, e a classificação prevista corresponde a *not p2p*. Nos *data files* 2 e 3 ambas as taxas são muito baixas, devido à elevada taxa de sucesso, principalmente no *data file* 3.

Olhando, agora, para os valores calculados referentes a cada taxa de erro, em cada *data file*, ao longo das 10 simulações, verificamos que existe uma variância muito reduzida entre o valor máximo e o valor mínimo, o que leva a crer que os resultados obtidos são sempre muito constantes. Tal acontece, devido à simplicidade do algoritmo em causa.