

# MAC460 Aprendizagem Computacional - Tarefa 2

Fernando Omar Aluani (NUSP: 6797226)

September 7, 2012

## 1 Execução e Especificação do Programa

Para executar o programa, basta rodar

*./tarefa2.py <treino> <teste>*

Onde:

- *<treino>* é o caminho para o arquivo que contém o conjunto de dados que será usado para treinamento.
- *<teste>* é o caminho para o arquivo que contém o conjunto de dados que será usado para testar os classificadores.

O programa aceita conjuntos de dados de qualquer tamanho, mas espera que ele sigam o mesmo padrão do *gerador.py*: "valorObservado classe". Ele também supõe que as 3 distribuições estatísticas possíveis são uniforme contínua (*uniform*), normal (*norm*) e exponencial (*expon*), que são as que o *gerador.py* usa.

Com os dados de treinamento, o programa estima os parâmetros das 3 distribuições estatísticas para as 3 classes nos 27 casos possíveis de combinações. Com os dados de testes, o programa testa cada um dos 27 classificadores e marca se ele acertou ou errou a classificação, e então imprime a "pontuação" de cada classificador nos testes e qual deles foi o que se saiu melhor. Quando há mais de um classificador empatado com a maior pontuação, o programa escolhe um desses aleatoriamente como o se saiu melhor.

## 2 Resultado dos Testes

Eis os resultados dos testes da execução do programa para um conjunto de amostras de treinamento e outro conjunto para testes gerados pelo *gerador.py*, lembrando que a pontuação é  $(acertos)/(totaldetestes)$ , e consequentemente  $(totaldetestes - acertos) = (erros)$ :

```

Classifier(uniform/uniform/uniform) SCORE: 47/60 [C1: 15/20; C2: 20/20; C3: 12/20]
Classifier(uniform/uniform/ norm) SCORE: 48/60 [C1: 19/20; C2: 20/20; C3: 9/20]
Classifier(uniform/uniform/ expon) SCORE: 29/60 [C1: 7/20; C2: 20/20; C3: 2/20]
Classifier(uniform/ norm/uniform) SCORE: 46/60 [C1: 15/20; C2: 17/20; C3: 14/20]
Classifier(uniform/ norm/ norm) SCORE: 47/60 [C1: 19/20; C2: 17/20; C3: 11/20]
Classifier(uniform/ norm/ expon) SCORE: 27/60 [C1: 7/20; C2: 18/20; C3: 2/20]
Classifier(uniform/ expon/uniform) SCORE: 39/60 [C1: 15/20; C2: 6/20; C3: 18/20]
Classifier(uniform/ expon/ norm) SCORE: 40/60 [C1: 19/20; C2: 6/20; C3: 15/20]
Classifier(uniform/ expon/ expon) SCORE: 19/60 [C1: 7/20; C2: 10/20; C3: 2/20]
Classifier( norm/uniform/uniform) SCORE: 48/60 [C1: 16/20; C2: 20/20; C3: 12/20]
Classifier( norm/uniform/ norm) SCORE: 48/60 [C1: 19/20; C2: 20/20; C3: 9/20]
Classifier( norm/uniform/ expon) SCORE: 35/60 [C1: 12/20; C2: 20/20; C3: 3/20]
Classifier( norm/ norm/uniform) SCORE: 47/60 [C1: 16/20; C2: 17/20; C3: 14/20]
Classifier( norm/ norm/ norm) SCORE: 47/60 [C1: 19/20; C2: 17/20; C3: 11/20]
Classifier( norm/ norm/ expon) SCORE: 35/60 [C1: 12/20; C2: 20/20; C3: 3/20]
Classifier( norm/ expon/uniform) SCORE: 39/60 [C1: 15/20; C2: 6/20; C3: 18/20]
Classifier( norm/ expon/ norm) SCORE: 39/60 [C1: 18/20; C2: 6/20; C3: 15/20]
Classifier( norm/ expon/ expon) SCORE: 34/60 [C1: 12/20; C2: 17/20; C3: 5/20]
Classifier( expon/uniform/uniform) SCORE: 47/60 [C1: 15/20; C2: 20/20; C3: 12/20]
Classifier( expon/uniform/ norm) SCORE: 48/60 [C1: 18/20; C2: 20/20; C3: 10/20]
Classifier( expon/uniform/ expon) SCORE: 47/60 [C1: 17/20; C2: 20/20; C3: 10/20]
Classifier( expon/ norm/uniform) SCORE: 46/60 [C1: 15/20; C2: 17/20; C3: 14/20]
Classifier( expon/ norm/ norm) SCORE: 47/60 [C1: 18/20; C2: 17/20; C3: 12/20]
Classifier( expon/ norm/ expon) SCORE: 47/60 [C1: 17/20; C2: 20/20; C3: 10/20]
Classifier( expon/ expon/uniform) SCORE: 39/60 [C1: 15/20; C2: 6/20; C3: 18/20]
Classifier( expon/ expon/ norm) SCORE: 40/60 [C1: 18/20; C2: 6/20; C3: 16/20]
Classifier( expon/ expon/ expon) SCORE: 46/60 [C1: 17/20; C2: 17/20; C3: 12/20]

```

## ARRUMAR ISSO AQUI - RECHECAR SAIDA DO PROGRAMA

Como é possível notar, o classificador "certo" (aquele que tem a mesma distribuição das classes geradas pelo *gerador.py*, expon/uniform/norm) fez 48 acertos, sendo o classificador que menos errou junto com outros 3 classificadores que também só erraram 12 testes.