

## Trabalho Prático I

Data de Entrega: 03 de Agosto 2018

O Trabalho Prático I visa implementar o Algoritmo de Aprendizagem Perceptron (PLA) descrito nos exercícios computacionais do **Homework #1**. Disponível em: <https://work.caltech.edu/homework/hw1.pdf>

Você deverá implementar o PLA na linguagem de sua escolha (preferencialmente em Matlab).

Um relatório prático sucinto deve ser escrito e entregue com o seguinte conteúdo:

- Você deve informar o seu nome no topo do relatório (primeira informação);
- Uma breve introdução sobre o assunto;
- O código fonte da implementação devidamente comentado;
- Resultados alcançados e sua interpretação dos mesmos.

\* O código deverá ser apresentado e executado em tempo oportuno, para fins de avaliação, com possível arguição do mesmo. A ser agendado.

## Descrição da tarefa

\*Uma tradução livre do enunciado presente no **Homework #1** foi gerada. Caso ocorram dúvidas na interpretação recorram ao texto original em inglês.

### O Algoritmo de Aprendizagem Perceptron

Neste problema, você vai criar a sua função *target* (alvo)  $f$  e o conjunto de dados  $D$  para verificar como o Algoritmo de Aprendizagem Perceptron trabalha. Utilize  $d = 2$  para você visualizar o problema, e assuma  $\mathcal{X} = [-1,1] \times [-1,1]$  com probabilidade uniforme de seleção  $\mathbf{x} \in \mathcal{X}$ .

Em cada execução, escolha uma reta aleatória no plano como sua função *target*  $f$  (faça isso utilizando dois valores aleatórios, pontos uniformemente distribuídos em  $[-1,1] \times [-1,1]$  gerando a reta que passa entre eles) na qual de um lado a reta mapeia  $+1$  e do outro  $-1$ . Escolha as entradas  $\mathbf{x}_n$  de uma base de dados de pontos aleatórios (uniformemente em  $\mathcal{X}$ ), e avalie a função *target* em cada  $\mathbf{x}_n$  e obtenha a correspondente saída  $y_n$ .

Para cada execução use o Algoritmo de Aprendizagem Perceptron (PLA) para encontrar  $g$ . Inicie o PLA como o vetor de pesos  $\mathbf{w}$  zerado (todos os pesos iguais a zero) e em cada iteração o algoritmo deverá escolher um ponto aleatório a partir do conjunto de pontos classificados incorretamente. Nós estamos interessados em dois valores: número de iterações que o PLA precisa para convergir a  $g$ , e a divergência entre  $f$  e  $g$  na qual  $\mathbb{P}[f(\mathbf{x}) \neq g(\mathbf{x})]$  (a probabilidade que  $f$  e  $g$  vão divergir na classificação de um ponto aleatório). Você pode calcular exatamente esta probabilidade ou você pode gerar uma aproximação gerando uma grande quantidade de conjuntos separados de pontos para estimá-la.

A fim de obter uma estimativa confiável para estes dois valores, você deve repetir o experimento por 1000 execuções (cada uma como foi especificado acima) e tomar a média sobre estas execuções.

1) Para  $N = 10$ . Em média quantas iterações são necessárias para que o PLA convirja para  $N = 10$  pontos treinados? Apresente o valor aproximado de seu resultado ( resultado próximo a media | sua resposta - opção | é próxima de 0).  
a) 1            b) 15            c) 300            d) 5000            e) 10000

2) Qual a opção mais se aproxima de  $\mathbb{P}[f(\mathbf{x}) \neq g(\mathbf{x})]$  para  $N = 10$ ;  
a) 0.001            b) 0.01            c) 0.1            d) 0.5            e) 0.8

3) Agora, teste  $N = 100$ . Em média quantas iterações são necessárias para que o PLA convirja para  $N = 100$  pontos de treinamento? Informe o valor mais próximo ao seu resultado.  
a) 50            b) 100            c) 500            d) 1000            e) 5000

4) Qual a opção mais se aproxima de  $\mathbb{P}[f(\mathbf{x}) \neq g(\mathbf{x})]$  para  $N = 100$ ;  
a) 0.001            b) 0.01            c) 0.1            d) 0.5            e) 0.8