

**Israel Emmanuel Jiménez Lopez.**

*Alumno*

**Reconocimiento de patrones.**

*Curso*

**Jesús Alejandro Flores Hernández**

*Profesor*

# Funcionamiento explicado del archivo "persimple\_cn\_GenAleat\_de\_CTok.sce"

|  |  |
| --- | --- |
| Código. | Explicación. |
| nct=20; // número de puntos  x=2\*rand(2,nct)-1; // genera coordenadas (x,y) aleatorias en el rango [-1,1] | Se generan 20 puntos bidimensionales.  Se grafican con plot(x1,y1,'\*');. |
| F=[1;-2]; // representa la recta y = 2x como vector normal (A=1, B=-2) x2=linspace(-1,1,100);  y2 = 2 \* x2;  plot(x2,y2,'r'); // Línea roja | Esta es la **recta de referencia** usada para clasificar los puntos: y−2x=0y - 2x = 0y−2x=0 ⇒ y=2xy = 2xy=2x |
| l(i) = -F(2)\*x1(i) - y1(i);  class\_F(i)=sign(l(i)); | Se clasifica cada punto como +1 o -1 dependiendo de qué lado de la recta se encuentra.  Se grafican en azul o verde según su clase. |
| g(i) = sign(w(2)\*y1(i)+w(1)\*x1(i)); | Calcula la salida del perceptrón para cada punto.  Compara con la clasificación correcta (class\_F) y busca los errores. |
| ind = find(g ~= class\_F);  while ~isempty(ind)  // Ajuste de pesos con la primera muestra mal clasificada  w(1) = w(1) + (class\_F(ind(1))-g(ind(1)))\*x1(ind(1));  w(2) = w(2) + (class\_F(ind(1))-g(ind(1)))\*y1(ind(1));  ...  end | Se corrigen los pesos hasta que todos los puntos estén bien clasificados.  Finalmente, se traza la nueva recta aprendida por el perceptrón. |

Este es un código generado en Scilab que implementa un perceptrón simple para clasificar puntos generados aleatoriamente dentro de un cuadrado de coordenadas entre -1 y 1. Se imprime el vector de pesos final w. Se dibuja la línea de decisión del perceptrón en verde.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.