

PRIMER ENTREGA: REPORTE PERCEPTRON MULTICAPA

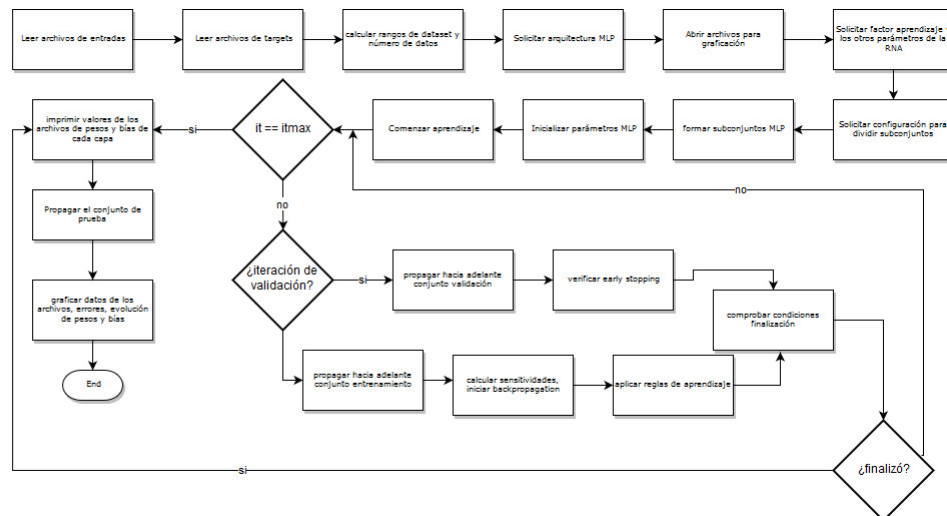
Equipo 4

AmezcuA Arevalo Bruno
Martínez Cerón José Emmanuel
Gómez Reus Jorge

Escuela Superior de Cómputo
Instituto Politécnico Nacional, México
Marco Antonio Moreno Armendariz

1 Introducción

Text



1.1 Seudocódigo

Descripción del algoritmo

Algorithm 1 MLP Entrenamiento

```

1: procedure LEER ARCHIVO DE ENTRADAS
2: end procedure
3:
4: procedure LEER ARCHIVO TARGETS
5: end procedure
6:
7: procedure CALCULAR RANGOS Y BUMERO DE DATOS
8: end procedure
9:
10: arq = input(Ingresa la arquitectura del MLP:)
11: fns = input (Ingresa las funciones de activación)
12: Abrir los archivos para gratificación
13:
14: for (i=1; i< NumeroDeCapas; i++) do
15:     archivos_pesos += 1
16:     archivos_bias += 1
17:     crear/abrir archivo pesos
18:     crear/abrir archivo bias
19: end for
20: procedure SOLICITAR PARAMETROS DE LA RNA(valor del factor de aprendizaje, error
    de época, itmax, etc)
21: end procedure
22:
23: procedure SOLICITAR CONFIGURACION PARA DIVIDIR SUBCONJUNTOS(80-10-10, 70-
    15-15)
24: end procedure
25:
26: procedure FORMAR SUBCONJUNTOS
27: end procedure
28:
29: procedure INICIALIZAR MATRIZ DE PESOS Y EL VECTOR BIAS CON VALAORES ALEATO-
    RIOS ENTRE -1 Y 1
30: end procedure
31:
32: W = cell(número de capas, 1)
33: b = cell(número de capas, 1)
34: %% Salidas de la capa
35: a = cell(número de capas, 1)
36: %% Sensitivities
37: S = cell (número de capa, 1)
38: F_m = cell(número de capas, 1)
39: for (j=1; i<=numeroDeCapas; i++) do
40:     Wi = rand() % (1) -1;
41:     bi = rand() % (1) -1;
42:     guardar archivo
43: end for

```

Algorithm 2 MLP Aprendizaje

```

early_stopping = false
2: error_val = 0
   error_apr = 0
4: num_itval = 0
   for (it_max) do
6:   Verificar error en validación
   if (it % itval != 0) then
8:     for (Datos de Entrenamiento) do
10:      procedure PROPAGAR DATOS DE ENTRENAMIENTO
12:      end procedure
14:    end for

16:    for (Número de capas) do
18:      procedure PROPAGAR CADA DATO
20:      end procedure
22:      procedure CALCULAR ERRORES
24:      end procedure

26:      procedure CALCULAR SENSITIVIDADES
28:      end procedure

30:      procedure BACKPROPAGATION
32:      end procedure

34:      procedure OBTENER FUNCIÓN DE ACTIVACIÓN PARA LA CAPA
36:      end procedure

38:      procedure APLICAR REGLAS DE APRENDIZAJE
40:      end procedure

42:    end for
44:    procedure CALCULAR EL ERROR DE APRENDIZAJE Y GUARDARLO
46:    end procedure

48:  else
50:    for (datos validación) do
52:      procedure PROPAGAR HACIA ADELANTE LOS ELEMENTOS DEL CONJUNTO
DE VALIDACIÓN
54:      end procedure

56:      for (Número de capas) do
58:        procedure PROPAGAR CADA DATO
60:        end procedure
62:      end for
64:      procedure CALCULAR ERROR DE VALIDACIÓN
66:      end procedure

68:    end for
70:    procedure VERIFICAR ERRORES PARA EARLY STOPPING
72:    end procedure
74:  end if
76:  procedure COMPROBAR CONDICIONES DE FINALIZACIÓN
78:  end procedure
80: end for

```

Algorithm 3 MLP Prueba

```

if (early_stopping == true) then
  procedure IMPRIMIR EN UN ARCHIVO LOS ÚLTIMOS VALORES DE PESOS Y BIAS DE
  CADA CAPA
3:   end procedure

  for ( doNúmero de capas)
6:   Imprimir dato de pesos
   Imprimir dato de bias
  end for
9: end if
  procedure PROPAGAR CONJUNTO DE PRUEBA
  end procedure
12: for (Número de elementos de prueba) do
  for (Número de capas) do
15:   procedure PROPAGAR CADA DATO
   end procedure
  end for
18: end for
  procedure GRAFICAR EL CONJUNTO DE PRUEBA Y TARGETS CONTRA RESULTADOS
  DE LA RED
  end procedure

```
