



TAREA #2B: DEEP LEARNING

Prof. NIBALDO RODRÍGUEZ A.



OBJETIVO

- Implementar y evaluar el rendimiento de un modelo Aprendizaje Profundo (DL) usando algoritmo Descenso Gradiente Estocástico para clasificar diez tipos de severidad de fallos de un motor eléctrico.



DATA : Train

- Formato: **train_x.csv** : (D,N), donde:
 - **D-filas** : número de atributos.
 - **N-columns** : números de muestras.
- Formato: **train_y.csv** : (C,N)
 - **C=10-filas** : etiqueta binaria para cada clase.
 - **N-columns**: números de muestras.



DATA: Test

- Formato: **test_x.csv** : (D,N)

- ☐ **D-filas** : número de atributos
- ☐ **N-columns** : números de muestras

- Formato: **test_y.csv** : (C,N)

- ☐ **C=10-filas** : etiqueta binaria para cada clases
- ☐ **N-columns**: números de muestras.

FASE 1: Pre-Tuning

■ train.py:

Inicialización de pesos con valores aleatorios

$$r = \sqrt{\frac{6}{n_i + n_{i-1}}}$$
$$w^{(i)} = \text{rand}(n_i, n_{i-1}) \times 2 \times r - r$$

n_i : Nodos capa siguiente

n_{i-1} : Nodo capa previa



FASE 1: Pre-Tuning

■ **train.py:**

☐ **Archivos de Salida:**

■ **costo_softmax.csv. :**

☐ N-filas por 1-columna

■ **Pesos del Deep Learning.**

☐ w_dl.npz



FASE 1: Pre-Tuning

- **test.py**

- **Archivos de Salida:**

 - **metrica_dl.csv.**

 - F-scores para cada una de las 10 clases.

 - F-score promedio.

Test.py: Métrica:

$$F - score (j) = 2 \times \frac{Pr ecision (j) \times Re call (j)}{Pr ecision (j) + Re call (j)}$$

$$Precision (i) = \frac{CM_{i,i}}{\sum_{j=1}^{n_L} CM_{i,j}}, \quad i = 1, \dots, n_L = 10$$

$$Re call (j) = \frac{CM_{j,j}}{\sum_{i=1}^{n_L} CM_{i,j}}, \quad j = 1, \dots, n_L = 10$$

$$avgFscore = \frac{1}{10} \sum_{i=1}^{10} Fscore (i)$$

CM(i,j) : Matriz de confusión



Configuración: AE-Apilados

■ **cnf_sae.csv:**

- Línea 1: Tamaño del Bloque (batch) :64
- Línea 1: Máximo Iteraciones : 60
- Línea 2: Tasa de aprendizaje : 0.1
- Línea 3: Nodos Oculto AE1 : 200
- Línea 4: Nodos Oculto AE2 : 150
- Línea 5: Nodos Oculto AE3 : 100
- ...
-



Configuración : Softmax

■ **cnf_softmax.csv**

- Línea 1: Máximo Iteraciones : 2000
- Línea 2: Tasa aprendizaje (mu) : 0.1



ENTREGA

- **Martes 26/Octubre/2021**

- ☐ Hora : 09:00 am

- ☐ Lugar : Aula Virtual del curso

- **Lenguaje Programación:**

- ☐ Python version: 3.7.6 window (anaconda)

- numpy

- panda



OBSERVACIÓN:

- Si un Grupo no Cumple con los requerimientos funcionales y no-funcionales, entonces la nota máxima será igual a 3,0 (tres coma cero).