

Profesor: Dr. Oldemar Rodríguez Rojas
 Análisis de Datos 2
 Fecha de Entrega: Sábado 3 de diciembre a las 9am
 Instrucciones:

- Las tareas son estrictamente individuales.
- Tareas idénticas se les asignará cero puntos.
- Todas las tareas tienen el mismo valor en la nota final del curso.
- Cada día de entrega tardía tendrá un rebajo de 20 puntos.

TAREA NÚMERO 14

- **Ejercicio 1:** [30 puntos] Dada una Red Neuronal RNN tipo LSTM, donde $\tau(x)$ se define como sigue:

$$\tau(x) = \begin{cases} 1 - \frac{1}{1+x} & \text{si } x \geq 0 \\ -1 + \frac{-1}{1-x} & \text{si } x < 0 \end{cases}$$

Además,

$$\begin{aligned} f_t &= \tau(W_{hf}h_{t-1} + W_{xf}x_t) \\ i_t &= \tau(W_{hi}h_{t-1} + W_{xi}x_t) \\ o_t &= \tau(W_{ho}h_{t-1} + W_{xo}x_t) \\ g_t &= \tanh(W_{hg}h_{t-1} + W_{xg}x_t) \\ h_0 &= \vec{0} \end{aligned}$$

$$\begin{aligned} c_t &= f_t \odot c_{t-1} + i_t \odot g_t \\ h_t &= o_t \odot \tanh(c_t) \\ c_0 &= \vec{0} \end{aligned}$$

Además, se tiene que:

$$\begin{aligned} x_1 &= \begin{pmatrix} 2 \\ 7 \\ 3 \\ 1 \end{pmatrix}, \quad x_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix}, \quad W_{xf} = \begin{pmatrix} 2 & 5 & 0 & 3 \\ 0 & 3 & 8 & 2 \\ 3 & 0 & 1 & 1 \\ 0 & 1 & 2 & 0 \\ 1 & 1 & 2 & 1 \end{pmatrix}, \quad W_{hf} = \begin{pmatrix} 1 & 0 & 1 & 1 & 2 \\ 1 & 1 & 4 & 1 & 2 \\ 3 & 0 & 1 & 1 & 1 \\ 1 & 1 & 2 & 0 & 9 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix} \\ W_{xi} &= \begin{pmatrix} 2 & 5 & 0 & 3 \\ 0 & 3 & 8 & 2 \\ 3 & 0 & 1 & 1 \\ 0 & 1 & 2 & 0 \\ 1 & 1 & 2 & 1 \end{pmatrix}, \quad W_{hi} = \begin{pmatrix} 1 & 0 & 1 & 1 & 2 \\ 1 & 1 & 4 & 1 & 2 \\ 3 & 0 & 1 & 1 & 1 \\ 1 & 1 & 2 & 0 & 9 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix} \end{aligned}$$

$$W_{xo} = \begin{pmatrix} 2 & 0 & 0 & 3 \\ 0 & 3 & 0 & 2 \\ 3 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}, \quad W_{ho} = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 2 \\ 3 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 9 \\ 1 & 1 & 0 & 1 & 1 \end{pmatrix}$$

$$W_{xg} = \begin{pmatrix} 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}, \quad W_{hg} = \begin{pmatrix} 1 & 0 & 0 & 1 & 2 \\ 1 & 1 & 0 & 1 & 2 \\ 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \end{pmatrix}$$

1. Calcule h_1 y h_2 .

2. Prediga $y_2 = W_{hy}h_2$ para $W_{hy} = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \end{pmatrix}$.

- **Ejercicio 2:** [35 puntos] Descomprima el archivo `emociones.zip`. Este conjunto de audios tiene 7 clases: `Disgustado`, `Enojado`, `Feliz`, `Neutral`, `Sorprendido`, `Temeroso` y `Triste`. El objetivo de este ejercicio es identificar la emoción que expresa la persona cuando habla. Con la tabla de datos realice lo siguiente:

1. Cargue los audios contenidos en el archivo `emociones.zip` y obtenga la etiqueta correspondiente para cada audio.
2. Realice un preprocesamiento de los datos con ayuda de la técnica de MFCC.
3. Divida la tabla utilizando un 90 % para entrenamiento y un 10 % para pruebas. Aplique el one-hot-encoding a las etiquetas.
4. Genere una estructura de Redes Neuronales LSTM. La estructura puede definirla a su conveniencia.
5. Configure el modelo con la función de costo `categorical_crossentropy`, la función de optimización `adam` y la métrica `accuracy`.
6. Entrene el modelo con los siguientes parámetros `epochs=100` y `batch size=32`. Esto puede tardar un rato. Establezca el valor de 0 para el parámetro `verbose` para omitir la salida en consola.
7. Utilice el método `evaluate()` para evaluar la precisión del modelo.
8. Obtenga y cargue 10 audios de una persona hablando, pueden ser de personas conocidas o tomados de internet. Luego con ayuda del modelo realice una predicción de esos 10 audios. Muestre y comente los resultados.

- **Ejercicio 3:** [35 puntos] En este ejercicio vamos a intentar predecir la letra que sigue del alfabeto griego. Cabe resaltar que no existe un mapeo exacto del abecedario al nuestro y que el número de letras es inferior al nuestro también.

Copie y pegue el siguiente código y utilicelo como set de datos.

```
alfabetoGriego = αβγδεζηθικλμνξοπρστυφχψω.
```

Para esto realice lo siguiente:

1. Defina un conjunto con los diferentes caracteres que conforman el set.
2. Cree un diccionario que permita definir la equivalencia entre caracter e índice, así como su inverso.
3. Haga los pares de entrada y salida para entrenar el modelo y transforme al formato esperado por la red.
4. Normalice los datos y transforme la variable a predecir a formato **One-Hot**.
5. Haga el modelo usando la función de activación **softmax** y con las capas que considere necesarias. Utilice la función de optimización **RMSProp**, la función de costo **categorical_crossentropy** y las métricas **accuracy**.
6. Haga un resumen del modelo.
7. Haga una predicción entrenando el modelo con 500 **epochs** y un tamaño de lote igual a 1.
8. Genere la matriz de confusión.
9. Calcule la precisión global. Interprete la calidad de los resultados.
10. Repita el ejercicio 5 pero esta vez utilice como optimizador la función **adam**. Compare resultados.

