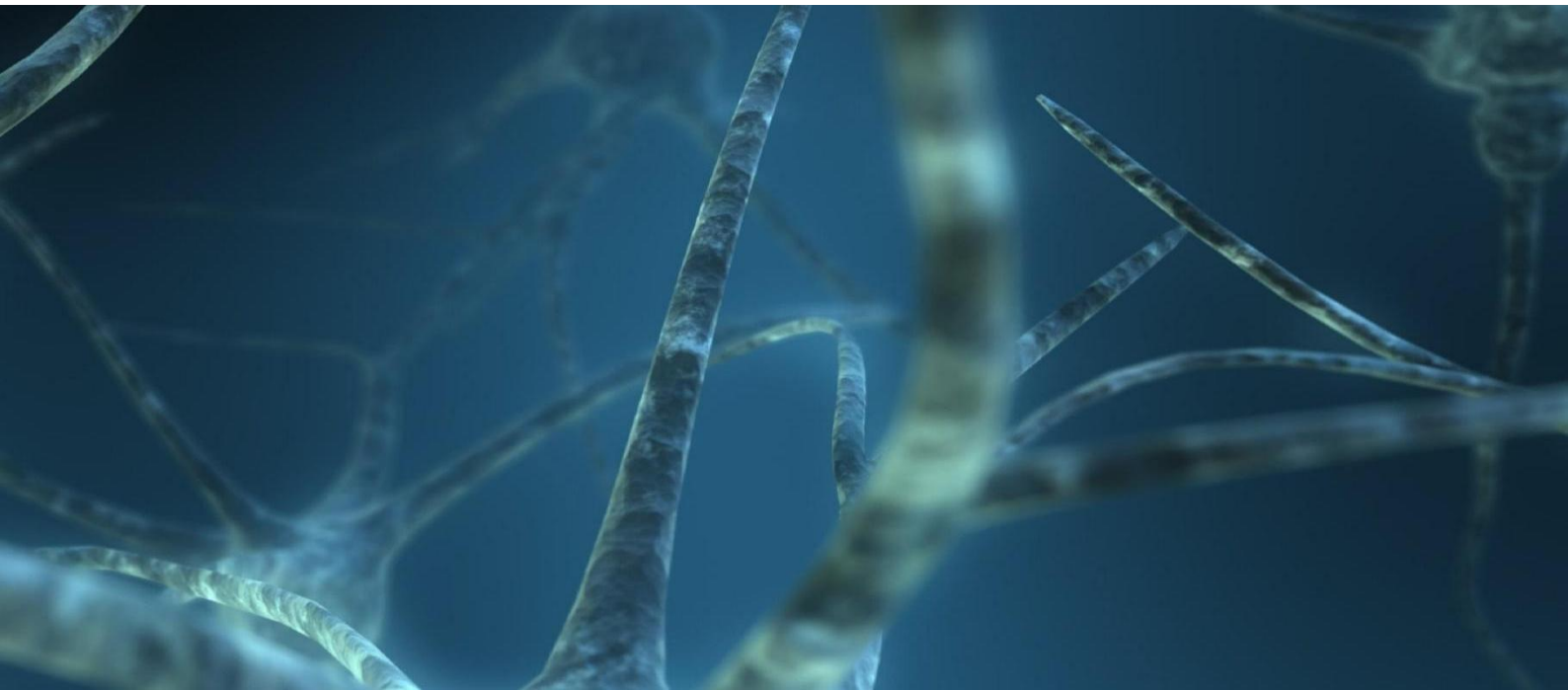


Práctica 3

Regresión Logística



Objetivos

El objetivo de esta práctica es comprender el funcionamiento de los combinadores lineales y no lineales

Temas

- Regresión Logística simple y múltiple
- Funciones de activación y de costo
- Regresión logística multiclase

Lectura

Material de Lectura: Capítulos 2 y 3 del libro Neural Networks and Deep Learning.

Regresión Logística Múltiple (varias variables de entrada y salida sigmoide en (0,1) o (-1,1))

Ejercicio 1

A partir de los ejemplos del archivo **automobile-simple.csv**, utilice todos los ejemplos para generar un modelo de 1 neurona para clasificar si un auto es ecológico:

- Elimine los registros que presenten valores faltantes y utilice el atributo eco-rating para generar un nuevo atributo binario que determine si un auto es ecológico o no. Considere que un auto es ecológico si el valor de "eco-rating" supera la media de dicho atributo.
- Luego de completar según lo indicado en a), aplique normalización y realice el entrenamiento utilizando los atributos numéricos:
 - Utilizando salida sigmoide (sigmoid) para la neurona.
 - Utilizando salida tangente hiperbólico (tanh) para la neurona.

Ejercicio 2

El archivo **semillas.csv** contiene información de granos que pertenecen a tres variedades diferentes de trigo.

Entrene una neurona no lineal para determinar si una semilla es de "TIPO 2" o no.

Utilice el 60% de los ejemplos para entrenar y el 40% para realizar el testeo.

Realice 30 ejecuciones independientes de la configuración seleccionada para respaldar sus afirmaciones referidas a la precisión obtenida tanto para el conjunto de entrenamiento como para el de testeo. Utilice un máximo de 200 iteraciones y una cota de error de $1e-03$.

- Utilizando 'sigmoid' como función de activación.
- Utilizando 'tanh' como función de activación.

Con los resultados obtenidos complete la siguiente tabla

Funcion	alfa	Iteraciones	Accuracy train	Ite accuracy	Veces (datos Train)	Accuracy test	Veces (Datos Test)
sigmoid	0,3						
	0,2						
	0,1						
tanh	0,3						
	0,2						
	0,1						

donde

- **Iteraciones** es la cantidad promedio de iteraciones realizadas en las 30 ejecuciones
- **Accuracy train** es el promedio de las 30 tasas de acierto obtenidas en las distintas ejecuciones.
- **Ite accuracy** es el número de iteración promedio en el que se encontró el máximo accuracy por primera vez.
- **Veces (datos train)** indica cuántas de las 30 ejecuciones alcanzaron el 100% de acierto durante el entrenamiento.
- **Accuracy test y Veces (datos Test)** contienen lo mismo que Accuracy train y "Veces (datos train)" pero sobre los datos de testeo, respectivamente.

Ejercicio 3

Se ha realizado un análisis químico a tres tipos distintos de vinos producidos en una misma región de Italia. El número de muestras considerado es el siguiente:

- Tipo 1: 59 muestras
- Tipo 2: 71 muestras
- Tipo 3: 48 muestras

El archivo **Vinos.csv** permite observar los resultados de este análisis. Cada fila representa una muestra distinta y está formada, en primer lugar, por el número del tipo al cual pertenece el vino analizado seguido por los 13 atributos que lo caracterizan.

Por ejemplo, la siguiente fila:

2, 12.29, 3.17, 2.21, 18, 88, 2.85, 2.99, 0.45, 2.81, 2.3, 1.42, 2.83, 406

es el resultado del análisis de un vino correspondiente al tipo 2 (1er. valor de la fila) seguido por 13 valores separados por comas que indican los niveles de las mediciones realizadas a dicho vino.

a) Entrene una red neuronal formada por una única neurona para clasificar los vinos de Tipo 1

Realice 30 ejecuciones independientes utilizando el 50%, 60%, 70%, 80% y 90% de los ejemplos como entrenamiento y el resto como testeo. Para cada porcentaje, indique la cantidad promedio de ejemplos correctamente clasificados en entrenamiento y en testeo. Calcule también el promedio y el desvío de la cantidad de iteraciones realizadas.

Utilice un máximo de 400 iteraciones y velocidades de aprendizaje 0.1, 0.2 y 0.3.

Analice los resultados obtenidos utilizando:

- i. Función de activación 'sigmoid' y función de costo 'ECM (error cuadrático medio)
- ii. Función de activación 'sigmoid' y función de costo 'EC_binaria' (entropía cruzada binaria)
- iii. Función de activación 'tanh' y función de costo 'ECM (error cuadrático medio)

Regresión Logística multiclase

Ejercicio 4

Se busca predecir si el tipo de fármaco que se debe administrar a un paciente afectado de rinitis alérgica es el habitual o no. Se dispone de información correspondiente a las historias clínicas de pacientes atendidos previamente

- a) Numere los atributos nominales y utilice el archivo **drugs_train.csv** para entrenar un modelo con 5 neuronas que sea capaz de predecir el tipo de fármaco que se debe administrar a un paciente afectado de rinitis alérgica.
- b) Luego utilice el archivo **drugs_test.csv** para medir la calidad del modelo.

Ejercicio 5

Utilice una red neuronal formada por una única capa de salida de 3 neuronas para clasificar las muestras de vino del archivo Vinos.csv descripto en el ejercicio anterior.

Realice 30 ejecuciones independientes utilizando el 60% y 80% de los ejemplos como entrenamiento y el resto como testeo.

Utilice un máximo de 300 épocas y velocidades de aprendizaje 0.1, 0.2 y 0.3. Calcule para cada ejecución la cantidad de épocas que la función de pérdida queda por encima de la cota $1e-05$ y muestre el promedio sobre todas las ejecuciones.

Complete la siguiente tabla con los resultados de las siguientes configuraciones

- Función de activación 'sigmoid' y función de costo 'ECM' (error cuadrático medio).
- Función de activación 'sigmoid' y función de costo 'EC_binaria' (entropía cruzada binaria).
- Función de activación 'tanh' y función de costo 'ECM' (error cuadrático medio).
- Capa 'Softmax' y función de costo 'EC' (entropía cruzada).

Train-test	Función y Costo	alfa	Promedio Iteraciones Cota	Promedio Accuracy train	Promedio Accuracy test
80-20	'sigmoid' 'ECM'	0,3			
		0,2			
		0,1			
	'sigmoid' 'EC_binaria'	0,3			
		0,2			
		0,1			
	tanh 'ECM'	0,3			
		0,2			
		0,1			
	'Softmax' 'EC'	0,3			
		0,2			
		0,1			
60-40	'sigmoid' 'ECM'	0,3			
		0,2			
		0,1			
	'sigmoid' 'EC_binaria'	0,3			
		0,2			
		0,1			
	tanh 'ECM'	0,3			
		0,2			
		0,1			
	'Softmax' 'EC'	0,3			
		0,2			
		0,1			