

Aplicación Real del Perceptron y su Aplicación Real

El modelo del Perceptron es una representación muy acertada de la neurona biológica, y permite realizar dentro del “Aprendizaje Automático” soluciones de Clasificación binaria (exclusivamente dos clases).

Lectura inicial:

El estudiante debe de realizar la lectura del archivo “*Clase_Perceptron_Caso_AND.ipynb*” que se encuentra disponible en el curso del aula digital del entorno UAO Virtual.

Una vez realizado el entendimiento previo, la actividad consiste en replicar el ejercicio de clasificación de la compuerta AND usando el Perceptron para el conjunto de datos Iris dataset. Esta vez, en particular, deberá de usar la siguiente celda para carga el Iris dataset:

```
df = pd.read_csv('https://archive.ics.uci.edu/ml/'  
                'machine-learning-databases/iris/iris.data', header=None)  
df.head()
```

Observe con detalle que se ha realizado una carga directa consultando una página web en particular, es decir sin necesidad de tener la base de datos de manera local.

Una vez obtenido la variable “*df*” que es de tipo DataFrame de Pandas, responder las siguientes preguntas:

1. Organice los datos provenientes de “*df*” de tal manera que cree una matriz de características la cual debe denominar “*X*” y debe de estar organizada de la siguiente manera: La columna 1 debe de ser la longitud del Sépalo y la columna 2 debe de ser la Longitud del Pétalo, para las muestras de la clase Setosa y Versicolor exclusivamente. Verifique que la dimensión de “*X*” debe de ser (100,2).
2. Cree una variable llamada “*y*” donde contenga los valores 1 y -1 (Vector de clases) que representen las clases Setosa y Versicolor respectivamente. Verifique que la dimensión de “*y*” debe de ser (1,100). Tenga presente que las primeras 50 muestras de la variable “*X*” deben de corresponder a la misma clase Setosa o Versicolor, como lo haya decidido organizar en el ítem 1.
3. Realice un gráfico el cual le permita graficar La longitud del Pétalo vs la Longitud del Sépalo para ambas clases usando etiquetas y la leyenda que permita diferenciar cada clase.
4. Realice un proceso de entrenamiento en 5 ocasiones distintas usando la clase “Perceptron” en la cual varíe el coeficiente de aprendizaje “*eta*” desde 0.3 haciendo incrementos de 0.1 hasta 0.7 y en las 5 ocasiones almacene los últimos valores de convergencia w_1 , w_2 y b . En las 5 ocasiones use un número de iteraciones “ $n_iter=20$ ”.

Sobre el mismo gráfico desarrollado en el ítem 3, superponga las 5 ecuaciones lineales provenientes de los valores obtenidos en el ítem 4. Recuerde que la ecuación de separación es:

$$X_2 = -\frac{w_1}{w_2} \cdot X_1 - \frac{b}{w_2}$$

5. Grafique en una sola figura el error obtenido por los 5 procesos de entrenamiento.
6. Tome el que usted considere (justifique su respuesta) el mejor entrenamiento de los 5 realizados anteriormente y evalúe las siguientes nuevas muestras usando la función “*predict*” y llene la columna faltante:

Muestra nueva	Longitud del Sépalo	Longitud del Pétalo	Clase obtenida
1	6.5	2	
2	5.5	1	
3	5.5	5	
4	7	4	

7. Tenga la libertad de modificar y crear una nueva clase llamada “Perceptron2” en la cual en este caso utilice la función de activación **Escalón Binario Unipolar (0,1)**. Repita Todo el taller desde el ítem 2 (dado que tendrá que cambiar el vector “*y*”) usando esta nueva función “Perceptron2”.

¿Qué puede concluir con los resultados obtenidos comparado con lo realizado anteriormente en “Perceptron”? que claramente usa la función de activación **Escalón Binario Bipolar (-1,1)**.

8. ¿Qué implicaciones tendría si no se usara el hiperparámetro **bias**? Consideraría que serviría para realizar la clasificación del ejercicio:
 - Compuerta AND realizado en “*Clase_Perceptron_Caso_AND.ipynb*”
 - Iris dataset dispuesto en este taller.

¿Qué puede concluir?

Todo el desarrollo debe de ser realizado en un notebook de Jupyter extensión .ipynb usando comentarios para explicar brevemente las preguntas planteadas.