



Nombre: Dana Carolina Ramírez Velázquez

Código: 220286547

División: Tecnologías para la integración Ciber-Humana

Carrera: Ingeniería en computación

Materia: Seminario de Inteligencia Artificial 2

Tarea: Practica 1 Ejercicio 1

Fecha: 04/09/23

Introducción

- El algoritmo del perceptrón es un algoritmo de aprendizaje supervisado utilizado en el campo de la inteligencia artificial y el machine learning. Fue desarrollado por Frank Rosenblatt en 1957 y es uno de los conceptos fundamentales en la historia de las redes neuronales y el aprendizaje automático.
- El objetivo del perceptrón es aprender a clasificar datos en dos categorías, es decir, realizar una clasificación binaria.

Desarrollo

- El desarrollo del algoritmo del perceptrón se basó en programar
- La parte de lectura de datos:

```
0 with open("./info/XOR_trn.csv", "r") as arc:
1     for line in arc.readlines():
2         data = line.split(",")
3         X1 = np.append(X1, float(data[0]))
4         X2 = np.append(X2, float(data[1]))
5         if int(data[2]) < 0:
6             Y = np.append(Y, 0)
7         else:
8             Y = np.append(Y, int(data[2]))
9 X = np.array([X1, X2])
0
```

- La parte procesamiento de datos :

```
def predict(self, X):
    p = X.shape[1]
    y_est = np.zeros(p)
    for i in range(p):
        y_est[i] = np.dot(self.w, X[:, i]) + self.b
        if y_est[i] >= 0:
            y_est[i] = 1
        else:
            y_est[i] = 0
    return y_est

def fit(self, X, Y, epochs=20):
    p = X.shape[1]
    for _ in range(epochs):
        for i in range(p):
            y_est = self.predict(X[:, i].reshape(-1, 1))
            self.w += self.eta * (Y[i] - y_est) * X[:, i]
            self.b += self.eta * (Y[i] - y_est)
```

- Y la parte del muestreo de los datos procesados:

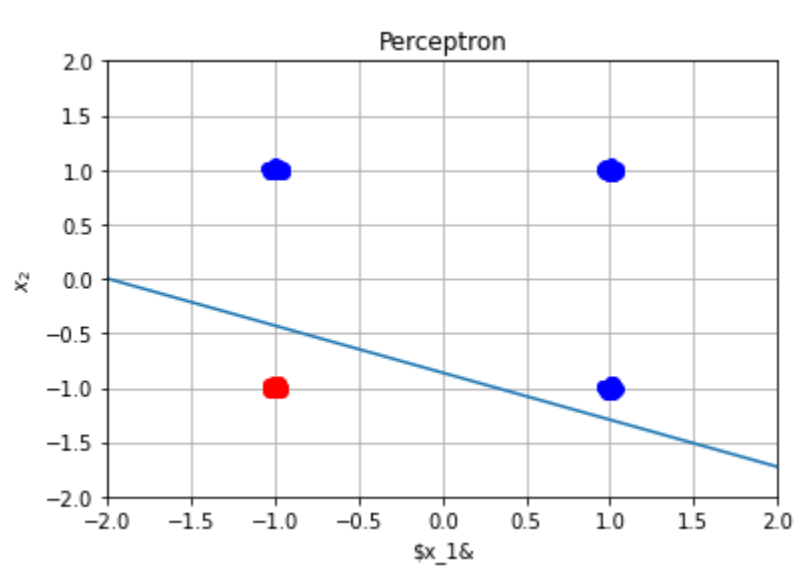
```

70 print(model.predict(X))
71
72 #Dibujo
73 p = X.shape[1]
74 for i in range(p):
75     if Y[i] == 0:
76         plt.plot(X[0, i], X[1, i], 'or')
77     else:
78         plt.plot(X[0, i], X[1, i], 'ob')
79
80 plt.title('Perceptron')
81 plt.grid('on')
82 plt.xlim([-2,2])
83 plt.ylim([-2,2])
84 plt.xlabel(r'$x_1$')
85 plt.ylabel(r'$x_2$')
86 draw_perceptron(model)
87 plt.show()

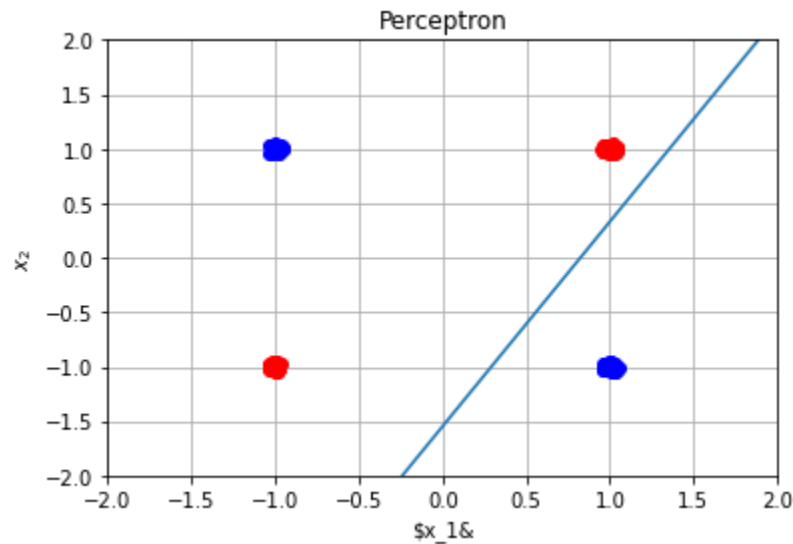
```

Conclusiones

- Lo que nos dió como resultado:
- Compuerta OR:



- Compuerta XOR:



- Podemos notar que al tratarse de datos linealmente separables, el algoritmo es capaz de clasificar los datos de forma efectiva, sin embargo, en otros casos como es el de la compuerta XOR, este algoritmo no tiene un buen desempeño.

Referencias:

- Mathivet, V. (2018). *Inteligencia artificial para desarrolladores: conceptos e implementación en c*. Ediciones ENI..
- Boden, M. A. (2017). *Inteligencia artificial*. Turner.
- Cerrillo Martínez, A. (2019). *El impacto de la inteligencia artificial en el derecho administrativo¿ nuevos conceptos para nuevas realidades técnicas?*.
- Leyva-Vázquez, M., & Smarandache, F. (2018). *Inteligencia Artificial: retos, perspectivas y papel de la Neutrosología*. Infinite Study.
- Leyva-Vázquez, M., & Smarandache, F. (2018). *Inteligencia Artificial: retos, perspectivas y papel de la Neutrosología*. Infinite Study.