



Trabajo Práctico N°3

Perceptrón Simple y Multicapa

Sistemas de Inteligencia Artificial

Alan Pierri
Rodrigo Ramele
Eugenia Piñeiro
Marina Fuster
Luciano Bianchi
Santiago Reyes
Marco Scilipoti
Paula Oseroff

Ejercicio 1

Implementar el algoritmo de perceptrón simple (**sin librerías**) con **función de activación escalón** y utilizar el mismo para aprender los siguientes problemas:

Función lógica “AND” con entradas:

$$x = \{-1, 1\}, \{1, -1\}, \{-1, -1\}, \{1, 1\}$$

y salida esperada:

$$y = \{-1, -1, -1, 1\}$$

Función lógica “XOR” con entradas:

$$x = \{-1, 1\}, \{1, -1\}, \{-1, -1\}, \{1, 1\}$$

y salida esperada:

$$y = \{1, 1, -1, -1\}$$

¿Qué puede decir acerca de los problemas que puede resolver el perceptrón simple escalón en relación a los problemas lógicos planteados en la consigna?

Ejercicio 2

Implementar el algoritmo de perceptrón simple lineal y perceptrón simple no lineal (**sin librerías**). Utilizar ambos para aprender a clasificar los datos en el archivo “TP3-ej2-conjunto.csv”.

- Comparar la capacidad de **aprendizaje** del perceptrón simple lineal y del perceptrón simple no lineal para aprender la función cuyas muestras están en los archivos.
- Elegir el perceptrón (perceptrón lineal o - exclusivo- no lineal) que haya dado mejores resultados de acuerdo a su potencial de aprendizaje y evaluar la capacidad de **generalización**.
- Para resolver el ítem 2, analizar las siguientes preguntas:
 - ¿Cómo elegirían el mejor conjunto de entrenamiento?
 - ¿Qué efecto tiene dicha elección en la capacidad de generalización?

Tip: buscar cross validation

Ejercicio 3

Implementar el algoritmo de perceptrón multicapa (**sin librerías**) y resolver las siguientes consignas:

- **Función lógica XOR:** Aprendizaje de la función presentada en el primer ejercicio, con los mismos datos y misma salida esperada. Este ejercicio es una herramienta para validar la correcta implementación de la red neuronal. NO debe incluirse en la presentación.
- **Discriminación de paridad:** determinar si un número es par/impar, utilizando el conjunto de datos presente en el archivo "TP3-ej3-digitos.txt" (dígitos de 7x5 dimensiones).
- **Discriminación de dígito:** Determinar qué dígito se corresponde con la entrada a la red. Por ejemplo, si alimentamos al perceptrón multicapa con una imagen del dígito "7", la salida esperada será "7" (la salida puede tomar valores entre 0 y 9). Nótese que se utiliza el mismo perceptrón multicapa, con una salida de 10 neuronas.
 - Una vez que la red haya aprendido, utilizar patrones correspondientes a los dígitos del conjunto de datos, con sus píxeles afectados por ruido. Evaluar los resultados.

Requerimientos mínimos:

- Análisis de hiperparámetros relevantes para cada ejercicio.
- Métricas de evaluación pertinentes y claramente explicadas para cada problema.
- Elección de arquitectura e hiperparámetros.
- Implementación de optimizadores debe incluir Gradiente Descendente, Momentum y Adam, para resolución del ejercicio 3.
- Conclusiones finales.

Tip: los análisis deben tener en cuenta las enseñanzas del TP-0.

Ejercicio 4 (OPCIONAL)

Usando el conjunto de datos MNIST, disponible en el siguiente [link](#), atacar el problema de clasificación de dígitos utilizando el **perceptrón multicapa implementado por el equipo (sin librerías)**. Este ejercicio es LIBRE y tiene como objetivo que se enfrenten a un problema abierto, de una complejidad mucho mayor a los anteriores. Deberán exponer cómo lo abordan y cómo este problema les resultó distinto al anterior (también de dígitos).

Algunas aclaraciones:

- Aquellos equipos presentando este ejercicio opcional pueden disponer de 5 minutos extra durante su presentación (en adicional a los 30 minutos originales). De todas formas, traten de presentar los resultados más interesantes así como los desafíos principales con los cuales se encontraron.
- Recomendamos no abordar este ejercicio antes de haber completado la resolución de los ejercicios anteriores.