

TP FINAL SIA

Contenido

Introducción:.....	2
Aclaraciones importantes	4

Introducción:

En los laboratorios de OpenRoberto SRL se nos ha solicitado brindar ayuda en el desarrollo del robot móvil “Roberto”. Para lo cual se ha nos ha solicitado a nuestro grupo de trabajo el diseño y desarrollo de una red neuronal que le permita al robot “Roberto” tomar la decisión correcta al momento de esquivar obstáculos en su recorrido.

El robot cuenta con:

- Cuatro sensores de proximidad que le permiten detectar obstáculos adelante, atrás, a la izquierda y a la derecha a una determinada distancia preestablecida.
- Dos motores servo mecánicos ubicados en los laterales del robot y una rueda loca que le sirve de apoyo.



Debemos tener en cuenta que en base a los sensores que posee “Roberto” podrían darse 16 posibles combinaciones ($16 = 2^4$) y que para cada combinación cada uno de los dos motores podría dar marcha atrás o adelante.

El comportamiento del robot esta dado por la siguiente tabla:

PATRON	S1	S2	S3	S4	M1	M2
P1	1	1	1	1	-1	-1
P2	-1	1	1	1	-1	1
P3	1	1	-1	-1	1	-1
P4	-1	-1	-1	-1	1	1
P5	1	-1	1	1	1	-1
P6	1	1	-1	1	-1	1
P7	1	1	1	-1	1	-1

Tabla con patrones de entrenamiento

Es decir que cuando los sensores detecten un objeto que se encuentra a una distancia inferior a la predeterminada se dirá que el objeto se encuentra cerca y esto se representará por medio de un 1 y cuando se detecte un objeto que se encuentra a una distancia mayor que la preestablecida se dirá que el objeto se encuentra lejos y esto será dispuesto en la tabla de entrenamiento por medio del valor -1; como ya se dijo: dependiendo de dichas lecturas los motores podrán dar marcha atrás (representado por -1) o adelante (representado por un 1) .

Entonces se nos pide:

1. Decidir si el diseño de la red neuronal debe ser de tipo Perceptron o de tipo multicapa Backpropagation.
2. En base a la red elegida determinar:
 - a. La cantidad de neuronas en la capa de entrada.
 - b. La cantidad de neuronas en la capa de salida.
 - c. La cantidad de capas ocultas y neuronas en las mismas (si el tipo de red lo requiere).
 - d. Los pesos iniciales.
 - e. La función de transferencia o activación.
3. Desarrollar la red aplicando el algoritmo correspondiente y utilizando el lenguaje de programación que se desee. Se solicita desarrollar para el algoritmo elegido la versión “vectorizada” y sin “vectorizar”. De manera que se pueda comparar el rendimiento de ambas implementaciones.
4. Entrenar la red desarrollada con la tabla de entrenamiento y mostrar las salidas correspondientes (y relevantes) del estado final de la red.
5. Determinar el mínimo error posible de la red, si una vez entrenada, se le vuelven a ingresar los patrones de entrenamiento. Se puede probar con diferentes topologías de Red y realizar una comparativa.
6. Simular la utilización del robot manipulando la API REST <https://www.agurait.com/ubp/sia/2022/roberto/> :
 - a. GET: por ejemplo devuelve un JSON con el siguiente contenido: [{"Resp":{"S1":"-1", "S2":"-1", "S3":"-1", "S4":"1", "Desc":""}}] el cual indica el valor de la lectura de los sensores para todas las combinaciones restantes.
 - b. POST: por ejemplo debe enviarse un JSON con el siguiente formato: [{"S1":"-1", "S2":"-1", "S3":"-1", "S4":"1", "M1":"1", "M2":"1"}] es decir el valor que fue devuelto por los sensores del robot en la consulta anterior (al hacer el GET) acompañado por los valores que fueron devueltos por la red neuronal (ya entrenada) para esas respectivas entradas obtenidas (al hacer GET) en el paso anterior.

- i. Si el POST fue **exitoso** la API devolverá: [{"Resp": "OK", "Desc": ""}]
- ii. Si el POST fue **NO exitoso** la API devolverá: [{"Resp": "KO", "Desc": "Una descripción de error"}]

Aclaraciones importantes

El TP puede ser realizado en grupos de hasta 4 integrantes y deben estar todos los integrantes en la mesa examinadora.

Se pueden solicitar al profesor, luego de que finalice el cursado, (por TEAMS) clases de consulta referidas al TP final.

Se destaca que en la mesa de examen final se espera:

1. Presentación de un informe en formato de documento PDF con la prolijidad acorde. (10%)
2. Explicación teórico/práctica del trabajo practico por parte de **todos** los integrantes del grupo (40%)
3. Construir una aplicación que corra sin presentar errores. (20%)
4. Que dicha aplicación cuente con una interfaz gráfica funcional de manera que permita “pronosticar” correctamente las decisiones del robot “Roberto” y cuente con alguna funcionalidad extra que los integrantes del grupo crean necesaria. (30%)