

**ESCUELA POLITECNICA NACIONAL**

**FACULTAD DE INGENIERIA MECANICA**

**APLICACIÓN DE LAS REDES NEURONALES AL  
RECONOCIMIENTO DE OBJETOS EN ROBOTS  
MANIPULADORES**

**TESIS PREVIA A LA OBTENCION DEL GRADO DE MASTER  
(MSc)  
EN DISEÑO PRODUCCION Y AUTOMATIZACION**

**WILLAN LEOPOLDO MONAR MONAR**

**William.monar@epn.edu.ec**

**DIRECTOR: ING. ÁLVARO GONZALO AGUINAGA B. Ph.D.**

**alvaro.aguinaga@epn.edu.ec**

**Quito, Octubre 2014**

## DECLARACIÓN

Yo Willan Leopoldo Monar Monar, declaro que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

La Escuela Politécnica Nacional, puede hacer uso de los derechos correspondientes a este trabajo, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

---

Ing. Willan Monar Monar

## CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por el Ingeniero Willan Monar Monar, bajo mi supervisión.

---

Ing. Álvaro Aguinaga B. Ph.D.  
**DIRECTOR DE TESIS**

## AGRADECIMIENTOS

Al Dr. Álvaro Aguinaga B. MSc. Ph.D., por su aporte académico, por su conducción brillante y paciente, por sus cualidades personales y profesionales y sobre todo por ser leal amigo y mentor de la presente investigación.

Al Ing. Mario Granja R. MSc., y al Ing. Ricardo Soto MSc., y al Ing. Edwin Salvador MSc. profesores de la Escuela Politécnica Nacional, por su decidido apoyo académico, imprescindibles para la realización de la presente investigación.

Al Ing. Robin Mancheno, Gerente Técnico de la Finca Rose Success, por su gentil aporte al haber proporcionado los datos de producción de rosas, para la realización de esta investigación.

## DEDICATORIA

Con inmenso cariño a toda mi familia  
En especial a mi querida esposa Janneth y  
a mis hijitos  
Christian y Gabyta  
Mi mini equipo de vida  
Les amo.

## CONTENIDO

CERTIFICACIÓN .....	iii
DEDICATORIA .....	v
ÍNDICE DE FIGURAS .....	x
ÍNDICE DE TABLAS .....	xii
CAPÍTULO I .....	1
INTRODUCCIÓN .....	1
1.1    PLANTEAMIENTO DEL PROBLEMA .....	1
1.1.3 OBJETIVOS .....	2
1.1.4 JUSTIFICACIÓN DE LA INVESTIGACIÓN.....	3
1.1.5 HIPÓTESIS DE LA INVESTIGACIÓN.....	3
1.2    INTELIGENCIA ARTIFICIAL .....	3
1.2.1    LÓGICA CLÁSICA BINARIA.....	4
1.2.2. LÓGICA DIFUSA .....	4
1.2.3 CONJUNTO DIFUSO .....	4
1.2.4 LÓGICA DIFUSA Y RAZONAMIENTO HUMANO.....	5
1.2.5 LOS INICIOS DE LA INTELIGENCIA ARTIFICIAL .....	5
1.2.6 OBJETIVOS DE LA INTELIGENCIA ARTIFICIAL.....	6
1.2.7 PROGRAMACIÓN ORIENTADA A OBJETOS .....	6
1.2.8 PROGRAMACIÓN DE VON NEUMANN .....	6
1.2.9 LOS SENTIDOS Y LA INTELIGENCIA ARTIFICIAL .....	7
1.3    ESTADO DEL ARTE DE LAS REDES NEURONALES .....	7
1.3.1 I. A. TRADICIONAL VS. CONEXIONISTAS.....	8
1.3.2 REDES NEURONALES PARADIGMA DE PROGRAMACIÓN .....	8
1.3.3 EL CEREBRO ARTIFICIAL .....	8
1.3.4 MÉTODOS DE CONTROL NO LINEAL.....	9
1.3.5 SOFTWARE DE SIMULACIÓN PARA I. A.....	10
1.3.6 AUTOAPRENDIZAJE .....	10
1.4    FUNDAMENTOS NEURONALES .....	11
1.4.1    NEURONAS .....	11
1.4.2 CARACTERÍSTICAS MORFOLÓGICAS DE LAS NEURONAS .....	11
1.4.3 SINAPSIS NEURONAL .....	12
1.4.4 ACCIÓN MOTORA POR ACTIVIDAD NEURONAL .....	13
1.4.5    CIRCUITOS NEURONALES Y REDES NEURONALES.....	13
1.4.6 PATRÓN DE ACTIVACIÓN NEURONAL .....	14

1.4.7 REPETICIÓN DE LA ACTIVIDAD NEURONAL .....	14
1.4.8 REESTRUCTURACIÓN NEURONAL .....	15
1.4.9 VELOCIDAD DE RESPUESTA MUSCULAR .....	15
1.4.10 LA CAPACIDAD DEL SISTEMA NERVIOSO .....	15
1.5 NEURONAS ARTIFICIALES .....	16
1.5.1 ENTRENAMIENTO DE LAS NEURONAS .....	17
1.5.2 LAS FUNCIONES DE UNA NEURONA ARTIFICIAL .....	17
1.5.3 APLICACIONES EN ROBOTS MANIPULADORES .....	18
1.6 MODELO NEURONAL DE McCOLLUCH Y PITTS .....	18
1.6.1 LIMITACIONES INICIALES .....	19
1.6.2 RED NEURONAL .....	19
1.6.3 LEJOS DE LA CONCEPCIÓN CLÁSICA .....	19
1.6.4 RECREACIÓN LAS NEURONAS DE McCULLOCH Y PITTS .....	20
CAPÍTULO II .....	36
REDES NEURONALES EN LA ROBÓTICA .....	36
2.1 GENERALIDADES .....	36
2.1.1 ROBÓTICA ETIOLÓGICA .....	36
2.2 TIPOS DE ROBOTS .....	36
2.2.1 ROBOTS ANDROIDES .....	37
2.2.2 ÚLTIMA GENERACIÓN DE ROBOTS HUMANOIDES .....	37
2.2.3 ASIMO DE HONDA .....	40
2.3 ROBOTS MÓVILES .....	42
2.3.1 TELEROBÓTICA ESPACIAL .....	43
2.4 ROBOTS MANIPULADORES .....	44
2.4.1 PRIMERAS INVESTIGACIONES .....	44
2.4.2 ROBOTS TIPO MAESTRO-ESCLAVO .....	45
2.4.3 MANIPULADOR INDUSTRIAL .....	45
2.4.4 REALIMENTACIÓN SENSORIAL .....	45
2.4.5 EL BRAZO STANFORD .....	46
2.4.6 EL ROBOT PUMA .....	48
2.4.7 LA ROBÓTICA EN LA ACTUALIDAD .....	49
2.5 SISTEMAS DE CONTROL DE ROBOTS .....	53
2.6 REDES NEURONALES Y ROBOTS MANIPULADORES .....	54
CAPÍTULO III .....	57

APLICACIÓN DE LAS REDES NEURONALES AL RECONOCIMIENTO DE OBJETOS EN ROBOTS MANIPULADORES.....	57
3.1 REDES NEURONALES ARTIFICIALES .....	57
3.1.1 EL MODELO DE MATEMÁTICO DE LA NEURONA ARTIFICIAL ...	58
3.1.2 REDES NEURONALES CELULARES.....	60
3.1.3 ALGORITMOS EVOLUTIVOS .....	62
3.2 PROCESOS DE APRENDIZAJE NEURONAL.....	63
3.2.1 CONOCIMIENTO Y ENTORNO. ....	63
3.2.2 ENTORNO DE INTERÉS Y PATRONES.....	63
3.2.3 APRENDIZAJE NEURONAL .....	64
3.2.4 TAMAÑO DE LA MUESTRA.....	64
3.2.5 PREPROCESAMIENTO .....	65
3.2.6 FORMAS DE ENTRENAMIENTO.....	65
3.2.7 SOBRE ENTRENAMIENTO .....	65
3.2.8 MATRIZ DE CONFUSIÓN .....	65
3.2.9 ERROR DE APRENDIZAJE .....	66
3.2.10 EL APRENDIZAJE.....	67
3.2.11 APRENDIZAJE SUPERVISADO .....	67
3.2.12 APRENDIZAJE POR REFUERZO .....	68
3.2.13 APRENDIZAJE NO - SUPERVISADO .....	68
3.1.14 ASOCIACIÓN DE PATRONES.....	69
3.2.15 RECONOCIMIENTO DE PATRONES .....	70
3.2.16 CONTROL DE PLANTAS CON REDES NEURONALES ARTIFICIALES .....	70
3.3 EL PERCEPTRÓN.....	70
3.3.1 IMPORTANCIA HISTÓRICA.....	70
3.3.2 PERCEPTRÓN SIMPLE.....	71
3.3.3 ARQUITECTURA DE PERCEPTRONES .....	72
3.3.4 ORDEN DEL PERCEPTRÓN .....	75
3.3.5 EFICIENCIA DE LOS PERCEPTRONES .....	75
3.3.6 EL PROBLEMA DE PARIDAD.....	76
3.3.7 PERCEPTRONES MULTICAPA.....	77
3.3.8 REGLA DE RETROPROPAGACIÓN.....	77
3.4 ARQUITECTURAS NEURONALES .....	79
3.4.1 REDES MONO CAPA.....	80
3.4.2 REDES MULTICAPA.....	81



3.4.3 ARQUITECTURA NEURONAL DE DOS CAPAS .....	81
3.4.4 REDES NEURONALES DE TRES CAPAS.....	81
3.4.5 REDES NEURONALES n CAPA .....	82
3.5 CONTROL DE ROBOTS MANIPULADORES USANDO REDES NEURONALES .....	82
3.5.1 APLICACIONES EN ROBÓTICA CON NNTOL DE MATLAB .....	83
3.5.2 CINEMÁTICA DEL ROBOT .....	85
3.5.3 CINEMÁTICA DIRECTA.....	85
3.5.4 CINEMÁTICA INVERSA .....	86
3.6 APLICACIÓN DE LAS REDES NEURONALES A LA SELECCIÓN DE OBJETOS.....	87
3.6.1 MODELO DE ZHAO-HUI JIAN .....	87
3.6.2 CONTROL ROBUSTO.....	88
3.6.3 RECONOCIMIENTO DE OBJETOS .....	89
3.6.4 CLASIFICACIÓN DE OBJETOS CON REDES NEURONALES.....	89
3.6.5 REDES NEURONALES ARTIFICIALES EN MATLAB .....	89
3.6.6 HERRAMIENTA RECONOCIMIENTO DE PATRONES NPRTOOL.	92
3.6.7 APLICACIÓN EN MATLAB DE CLASIFICACIÓN DE PATRONES..	93
CAPÍTULO IV.....	115
CONCLUSIONES Y RECOMENDACIONES.....	115
4.1 CONCLUSIONES .....	115
4.2 RECOMENDACIONES.....	117
BIBLIOGRAFIA .....	120

## ÍNDICE DE FIGURAS

Figura 1.1 Diagrama básico de una neurona .....	12
Figura 1.2 Sinapsis en los terminales de las neuronas.....	13
Figura 1.3 Circuitos neuronales .....	16
Figura 1.4 Representación gráfica de la neurona de McCulloch y Pitts .....	21
Figura 2.1 Otonaroid, cuál es humano y cuál robot?.....	37
Figura 2.2 Kodomoroid modelo y robot.....	39
Figura 2.3 Dr. Hiroshi Ishiguro y Mamoru Mori .....	39
Figura 2.4 ASIMO de Honda.....	41
Figura 2.5 Innovaciones tecnológicas de ASIMO de Honda 2014 .....	43
Figura 2.6 Curiosity de la Misión Marte de la Nasa .....	44
Figura 2.7 Brazo Stanford .....	47
Figura 2.8 Robot PUMA 500.....	49
Figura 2.9 Robot manipulador quirúrgico Da Vinci .....	49
Figura 2.10 UR5 y UR10 de Global Robots.....	50
Figura 2.11 Robot 140 IRB de ABB .....	52
Figura 2.12 Robot IRB 340 de ABB .....	52
Figura 3.1 Modelo matemático de una neurona artificial de McCulloch y Pitts. ....	58
Figura 3.2 Funciones de activación .....	59
Figura 3.3 Función de Activación Sigmoidal Bipolar .....	60
Figura 3.4 Matriz de Confusión .....	66
Figura 3.5 Estimación de Error de Desempeño en Redes Neuronales .....	66
Figura 3.6 Modelo de Aprendizaje con Profesor.....	67
Figura 3.7 Modelo de Aprendizaje por Refuerzo .....	68
Figura 3.8 Modelo de Aprendizaje No-Supervisado.....	69
Figura 3.9 Modelo de un perceptrón simple según Matlab.....	71
Figura 3.10 El perceptrón como clasificador lineal .....	72
Figura 3.11 Arquitectura de Perceptrones en notación Matlab .....	73
Figura 3.12 Arquitectura de una red neuronal mono capa .....	80
Figura 3.13 Red Neuronal con Arquitectura de Tres Capas .....	82
Figura 3.14 Modelo de un robot manipulador esférico industrial .....	84
Figura 3.15 Cinemática directa y cinemática inversa .....	85
Figura 3.16 Cuadro de dialogo de nstart de Matlab .....	90
Figura 3.17 Red Neuronal para Reconocimiento de Patrones de Matlab .....	92

Figura 3.18 Ventana de inicio de nntool de matlab. ....	93
Figura 3.19 Caja de herramientas de redes neuronales en Matlab. ....	94
Figura 3.20 Creación de una Nueva Red Neuronal de Trabajo.....	95
Figura 3.21 Tipos de Redes Neuronales en Ambiente Matlab. ....	96
Figura 3.22 Caja de diálogo de la función de entrenamiento.....	97
Figura 3.23 Función de Adaptación del Aprendizaje.....	98
Figura 3.24 La Función de Rendimiento .....	98
Figura 3.25 Número de Capas .....	99
Figura 3.26 Funciones de transferencia para el sistema de control.....	100
Figura 3.27 Ventana de ingreso de los datos.....	100
Figura 3.28 Nombre de la Red de Trabajo .....	103
Figura 3.29 Entradas en inputs .....	104
Figura 3.30 New Data Created.....	104
Figura 3.31 Introducción de los Targets.....	105
Figura 3.32 Red neuronal y parámetros seleccionados.....	105
Figura 3.33 Red Perceptrón Mono capa de Tres Neuronas.....	106
Figura 3.34 Selección de la Red a analizar .....	107
Figura 3.35 Función de entrenamiento.....	107
Figura 3.36 Recursos de entrenamiento de la red perceptrón.....	108
Figura 3.37 Variación del error con cada iteración .....	108
Figura 3.38 Simulación .....	108
Figura 3.39 Re inicialización de los pesos.....	109
Figura 3.40 Ajuste de los pesos para la red Perceptron .....	109
Figura 3.41 Selección de la red Fedd-forward backpropagation .....	110
Figura 3.42 Arquitectura de la red Feed-forward backpropagation .....	110
Figura 3.43 Entrenamiento de la red .....	111
Figura 3.44 Matriz de ajuste de pesos de la red Fedd-forward backpropagation .....	111
Figura 3.45 Salidas de la red Feed-forward backpropagation .....	111
Figura 3.46 Errores en la red Feed-forward backpropagation .....	112
Figura 3.47 Error de la red Feed-forward backpropagation para la clasificación de rosas. ....	113
Figura 3.48 Estado de entrenamiento .....	113
Figura 3.49 Regresión del error durante el entrenamiento de la red .....	114

## ÍNDICE DE TABLAS

Tabla 1.1 Comportamiento una neurona MCP .....	21
Tabla 1.2 Reemplazo de los valores de la tabla 1.1 si por 1 y 0 por no. ....	22
Tabla 1.3 Salidas de una Neurona MCP con umbral 1 .....	23
Tabla 1.4 Respuesta de una Neurona MCP a Diferentes Señales .....	25
Tabla 1.5 Cambio de notación Si = 1 No = 0 para MCP Umbral = 1.....	27
Tabla 1.6 Efectos de Dos Señales Inhibitorias y Umbral $b = 0$ .....	28
Tabla 1.7 Combinación de Dos Señales Excitatorias.....	29
Tabla 1.8 Posibilidades de combinación de 2 señales con dos sensores.....	31
Tabla 1.9 Respuesta de una MCP con pesos 0.5 y -0.3 y Umbral 0.1 .....	33
Tabla 3.1 Matriz de ceros y unos para formar el patrón T .....	74
Tabla 3.2 Ejemplos del problema de paridad .....	76
Tabla 3.3 Descripción de los Comandos de nntool .....	94
Tabla 3.4 Descripción de cada ítem.....	94
Tabla 3.5 Conjunto de Entrenamiento para la Selección de Rosas.....	101
Tabla 3.6 Tabla de resultados .....	112

## RESUMEN

El objetivo general de esta investigación es determinar las aplicaciones de las redes neuronales al control de brazos robóticos manipuladores; cuyos objetivos específicos son evaluar el estado del arte de los sistemas de control basados en redes neuronales en robots manipuladores; definir los parámetros funcionales de las redes neuronales que intervienen en los sistemas de control de robots manipuladores; definir las características relevantes de objetos a ser discriminados por un sistema automático de toma de decisiones en base a redes neuronales y clasificar objetos utilizando redes neuronales en la toma de decisiones por robots manipuladores.

La inteligencia artificial tiene sus inicios como resultado de la investigación en psicología cognitiva y lógica matemática, enfocada a la explicación del trabajo mental y la construcción de algoritmos de solución de problemas genéricos, que favorece la abstracción y la generalidad, es una combinación de la ciencia de las computadoras, la biología, la fisiología y la filosofía y las herramientas fundamentales que se utilizan son la lógica difusa, algoritmos genéticos, algoritmos evolutivos, teoría del caos y las redes neuronales.

La robótica es una tecnología multidisciplinaria de vanguardia que hace uso de los recursos de ciencias como mecánica, matemáticas, electrónica, informática, energía, visión e inteligencia artificial, lógica difusa, algoritmos genéticos y redes neuronales; para que un sistema mecánico se llame robot debe necesariamente ser inteligente o debe poseer cierta inteligencia que se debe normalmente a los algoritmos computacionales asociados con su sistema de control y sensorial; existen diferentes maneras de proporcionarle de inteligencia a una máquina, sea con lógica difusa, redes neuronales o una combinación de las dos; en este trabajo se ha optado por el control de movimientos de brazos manipuladores mediante redes neuronales.

Los sistemas neuronales artificiales se basan en las propiedades no lineales de la operación sinérgica de unidades de proceso elementales en paralelo, y tienen

como característica que su función quede directamente ser trasladada a una arquitectura modular luego de un proceso de aprendizaje, de no programación, es una alternativa a la computación basada en reglas de decisión prefijadas y algoritmos codificados que resultan ineficaces, en aplicaciones de clasificación, asociación y generalización. En este trabajo se aplica el nntool de Matlab para la clasificación de rosas de la empresa Rose Succes.

## PRESENTACIÓN

La presente investigación está estructurada en cuatro capítulos que se sintetizan a continuación, En el capítulo 1 se realiza una introducción sobre el tema y la hipótesis planteada en esta tesis; se presentan los objetivos de la investigación y se fundamentan las redes neuronales artificiales, se hace un análisis a la historia del arte de las redes neuronales como un cambio de paradigma al método convencional de programación secuencial de Von Neumann y su influencia en la inteligencia artificial.

En el capítulo dos se revisan los últimos avances de la robótica analizando los tipos de robots que existen como los androides, parecidos a los seres humanos donde se hace notar lo lejos que hemos llegado con los nuevos robots modelo 2014, a tal punto que sea difícil distinguir a simple vista cual es humano y cual robot. Los robots móviles provistos de elementos que le permiten desplazarse básicamente se usan para trabajos peligrosos y los robots manipuladores o industriales, que dotados por sistemas combinados de control y redes neuronales son cada vez más autónomos.

En el capítulo tres se realiza la aplicación de las redes neuronales al reconocimiento de objetos en este caso la clasificación de rosas utilizando la herramienta nntool del toolbox de Matlab de Mathworks en robots manipuladores; Las redes neuronales artificiales son modelos matemáticos no algorítmicos que tratan de reproducir el comportamiento del cerebro humano debido a que constan de dispositivos elementales de proceso, denominados neuronas artificiales.

En el capítulo 4, se presentan las conclusiones y recomendaciones, siendo una de las más importantes la que demuestra la hipótesis planteada en esta tesis, se cumplió el objetivo general de esta investigación de determinar las aplicaciones de las redes neuronales al control de brazos robóticos manipuladores utilizando redes neuronales, con el propósito de definir las características relevantes de objetos o productos para ser utilizados como conjunto de entrenamiento, en este

caso las características cuantitativas de los grupos de rosas de producción nacional de la finca Rose Success de la provincia de Cotopaxi Ecuador, para su clasificación.



# **CAPÍTULO I**

## **INTRODUCCIÓN**

En este capítulo se presentan los objetivos de la investigación y se hace una introducción a las aplicaciones de las redes neuronales a la inteligencia artificial como un cambio de paradigma al método convencional de programación de Von Neumann.

### **1.1 PLANTEAMIENTO DEL PROBLEMA**

La Facultad de Ingeniería Mecánica de la Escuela Politécnica Nacional imparte cursos de robótica con principal aplicación en procesos industriales en mecanismos que permitan maximizar la productividad de las empresas manufactureras en nuestro país, siendo una de las aplicaciones más utilizadas en mecánica, la robótica de manipuladores.

El control de los sistemas robotizados es uno de los puntales de la base del éxito y contar con herramientas amigables en términos utilitarios específicos que permitan enseñarle al robot las funciones que debe desempeñar, basada en redes neuronales artificiales utilizando herramientas matemáticas computacionales como Matlab.

Los cursos de robótica que se imparten tanto en pregrado como en postgrado en la Facultad, tienen alta expectativa de los estudiantes por interiorizar los conocimientos, pues toda la industria del siglo XXI irá robotizando sus procesos para ser competitivos en esta sociedad global.

#### **1.1.2 SISTEMATIZACION DEL PROBLEMA**

¿Es conveniente generar un documento en términos amigables para el estudio de robótica aplicada a la Ingeniería Mecánica? el diseño de un documento acorde con las exigencias de los docentes y las expectativas de los estudiantes, utilizando notación que es de uso común a nivel mundial redundará en beneficios para los involucrados en la problemática del incremento de la productividad del País para producir un cambio de la matriz productiva.

¿Cómo se plantea el proceso para generar un documento que permita hacer más cómodo el proceso de enseñanza-aprendizaje de las redes neuronales en la Facultad? basándose en un análisis previo de la situación que se quiere resolver en referencia a los sistemas de control de robots, utilizando redes neuronales.

¿Dónde se obtiene la información para generar un documento en términos amigables para el estudio de las redes neuronales y su aplicación en la robótica? la información se obtendrá por una parte, de la bibliografía que se utiliza en la actualidad y en los apuntes de realizados por el participante de esta tesis, en clases de pregrado, postgrado y otros eventos, donde están planteados los conocimientos básicos involucrados.

¿Cuál es la ventaja del final del control de brazos industriales por redes neuronales? La ventaja competitiva será la innovación y adecuación del uso de las redes neuronales al control de brazos manipuladores, base fundamental para la implementación de un estudio a nivel nacional con resultados asequibles a todos los involucrados, reconociendo el trabajo que se ha venido haciendo en la Facultad de Ingeniería Mecánica de la Escuela Politécnica Nacional, validando sus procesos de enseñanza-aprendizaje realizados en el día a día en las aulas.

¿Qué ventaja adicional tendrá este diseño? el documento generado será la base para una aplicación más amigable de los sistemas de control de robots de brazos manipuladores basado en redes neuronales, como una aplicación en la ingeniería mecánica.

### **1.1.3 OBJETIVOS**

El objetivo general de esta investigación es determinar las aplicaciones de las redes neuronales al control de brazos robóticos manipuladores; cuyos objetivos específicos son evaluar el estado del arte de los sistemas de control basados en redes neuronales en robots manipuladores; definir los parámetros funcionales de las redes neuronales que intervienen en los sistemas de control de robots manipuladores; definir las características relevantes de objetos a ser discriminados por un sistema automático de toma de decisiones en base a redes neuronales y clasificar objetos con robots manipuladores.

#### **1.1.4 JUSTIFICACIÓN DE LA INVESTIGACIÓN**

Este trabajo de investigación se justifica, pues la Escuela Politécnica Nacional una universidad pública, referente a nivel nacional en ciencia, tecnología e innovación que forma académicos y profesionales en ingeniería y ciencias, capaces de contribuir al bienestar de la comunidad, así como: generar, difundir y transmitir el conocimiento científico y tecnológico; la misma que necesita permanentemente implementarse y actualizarse en tecnologías punteras, que le permita mantener su visión y cumplir su misión.

#### **1.1.5 HIPÓTESIS DE LA INVESTIGACIÓN**

Si el software de nntool del toolbox de Matlab es capaz de clasificar objetos de un conjunto de entrenamiento mediante redes neuronales artificiales, entonces todo un lote producción puede ser clasificado y los resultados pueden utilizarse para el control de robots manipuladores, como una aplicación de la ingeniería mecánica y su contribución para la implementación en las asignaturas de robótica en las carreras de Ingeniería Mecánica, con lo que seremos capaces de demostrar que estamos avanzando en este ámbito, siendo competentes y capaces de generar resultados técnicamente válidos.

Entre otros beneficios adicionales que contempla este estudio están las aplicaciones de las redes neuronales a nivel industrial, para la optimización de los recursos y procesos, minimización de indecisiones e improvisaciones, detección de errores, mejoramiento de procesos y entrenamiento más eficaz, minimización de sobre esfuerzos por parte de los operadores evitando largos periodos de concentración, en definitiva para incrementar la productividad de la industria y la mejora de la calidad de vida de la gente.

### **1.2 INTELIGENCIA ARTIFICIAL**

La inteligencia artificial tiene sus inicios como resultado de la investigación en psicología cognitiva y lógica matemática, enfocada a la explicación del trabajo mental y la construcción de algoritmos de solución de problemas genéricos, que favorece la abstracción y la generalidad, es una combinación de la ciencia de las computadoras, la biología, la fisiología y la filosofía y las herramientas fundamentales que se utilizan son la lógica difusa, algoritmos genéticos y las redes neuronales.

### **1.2.1 LÓGICA CLÁSICA BINARIA**

En 2013 Marco García Baturán español, escribe su artículo Computación Blanda, cuyo contenido se refiere a que el ser humano, en la búsqueda de la precisión, intentó ajustar el mundo real a modelos matemáticos rígidos y estáticos, como la lógica clásica binaria, cuando Aristóteles y sus precursores idearon sus teorías de la lógica y las matemáticas, propusieron la Ley del Centro Excluido, que se refiere a que toda proposición sólo puede ser verdadera o falsa. La nube es blanca o no blanca; definitivamente no puede ser que la nube sea blanca y no blanca.

### **1.2.2. LÓGICA DIFUSA**

“Tuvieron que pasar muchos años desde el inicio de la matemática hasta que, a mediados del siglo veinte, fue concebido el concepto de lógica difusa por el matemático azerbaiyano-iraní Lofti Asker Zadeh, de la Universidad de Berkeley, el año 1972, como una extensión de la lógica de conjuntos discretos.

Lofti Asker Zadeh inconforme con los conjuntos clásicos que sólo permiten dos opciones, la pertenencia o no de un elemento a un determinado conjunto, presentó la lógica difusa como una forma de procesar información permitiendo pertenencias parciales a ciertos conjuntos, que contrariamente a los conjuntos típicos los denominó conjuntos difusos.

La lógica difusa ya se utilizó hace más de 24 siglos cuando Parménides de Elea sugirió que una proposición podía ser verdadera y falsa al mismo tiempo, a su admirador Platón le gustó la idea y admitió una tercera región entre los polos de lo verdadero y lo falso”. [40]

### **1.2.3 CONJUNTO DIFUSO**

El concepto de conjunto difuso en la actualidad fue propuesto por Zadeh en sus artículos: Conjuntos Difusos y Semántica Difusa Cuantitativa en la revista Información y Control publicados en los años 1965 y 1971 respectivamente en donde introduce los elementos formales que terminarían componiendo la doctrina de la lógica difusa y sus aplicaciones.

La lógica difusa es una herramienta que permite el control de sistemas muy complejos, de los que no puede extraerse un modelo matemático, su estructura

es muy cercana a la forma en la que un ser humano procesa la información, ya que permite crear distinciones en los sistemas, como muy caliente o muy frío, que son entendidos por el sistema y generan una respuesta en el controlador que se esté usando, el controlador es la estructura en la que se encuentra inscrito el sistema difuso.

#### **1.2.4 LÓGICA DIFUSA Y RAZONAMIENTO HUMANO**

La lógica difusa, permite una aproximación de los sistemas computacionales al razonamiento humano, y permite el manejo de incertidumbre en las decisiones que toma. La teoría de conjuntos difusos es el marco de referencial matemático para representar y resolver la incertidumbre en el sentido de vaguedad, imprecisión, falta de información y verdad parcial.

#### **1.2.5 LOS INICIOS DE LA INTELIGENCIA ARTIFICIAL**

“El 31 de Agosto 1955, Jhon McCarthy del Dartmouth College, New Hampshire, M.L. Minsky de Harvard University, N. Rochester de I.B.M. Corporation y C.E. Shannon de Bell Telephone Laboratories, lanzaron una propuesta para reunir en el verano de 1956 a un grupo de investigadores que quisieran trabajar sobre la hipótesis de que cada aspecto del aprendizaje y cada característica de la inteligencia podían ser tan precisamente descritos que se podían crear máquinas que las simularan. El encuentro, celebrado en 1956 y conocido como la conferencia de Dartmouth, se llevó a cabo con tal éxito que en el mismo evento McCarty inventó el término Inteligencia Artificial y con él una nueva área científica de conocimiento”<sup>1</sup>.

En este año 2014 se cumplen cincuenta y ocho años de la Conferencia de Dartmouth y cada vez se ven más lejanos esos sueños, mientras más se desarrollan nuevos materiales, nuevos programas de computación, nuevas ciencias que lo abordan y estudian el problema de encontrar las minuciosas descripciones de las características del cerebro y de la mente que fue mencionado en la propuesta original de 1955 sigue vigente y más distante y es que esa es una de las características intrínsecas del cerebro humano que mientras más conoce de un tema se siente más pequeño o ignorante.

---

<sup>1</sup> FERNANDEZ-CABALLERO, A. (2006); 50 Años de la Inteligencia Artificial, Ed. Sergio Miguel Tome; Albacete, España

### **1.2.6 OBJETIVOS DE LA INTELIGENCIA ARTIFICIAL**

Uno de los principales objetivos que se busca alcanzar con la inteligencia artificial es emular el razonamiento del cerebro humano, tiene que ver con la ingeniería del software en el diseño de entornos de programación inteligente, la representación del conocimiento mediante sistemas expertos cuya función es la de razonar, para lo que se usan muchas herramientas y variadas ciencias control automático, bases de datos inteligentes, reconocimiento de patrones, comprensión y síntesis del habla y visión artificial.

Cuando se va a trabajar en inteligencia artificial pueden considerarse dos teorías: construir réplicas de las redes neuronales del cerebro humano (bottom-up) visualizando soluciones de una manera no convencional, o imitar el comportamiento del cerebro humano con un computador (top-down) de manera algorítmica tradicional. Así mismo pueden considerarse dos metodologías diferentes: la lógica difusa para la toma de decisiones bajo condiciones de incertidumbre, y las redes neuronales artificiales que consiste en una poderosa herramienta tecnológica en ciertas tareas como clasificación y reconocimiento de patrones, cuyo concepto se basa en aprender mediante la sumatoria de un elevado número de elementos simples (neuronas), trabajando y generando información en paralelo.

### **1.2.7 PROGRAMACIÓN ORIENTADA A OBJETOS**

En el estudio de los sistemas informáticos, una propiedad que discrimina los métodos de inteligencia artificial de los métodos numéricos es el uso de símbolos no precisamente matemáticos, lo que no es suficiente pues otros tipos de programas computacionales procesan también símbolos y no usan técnicas de inteligencia artificial. Un programa de inteligencia artificial no produce salidas fijas y son determinadas secuencialmente por las salidas intermedias durante las consideraciones de cada problema específico; los lenguajes orientados a objetos cumplen esta propiedad y son los más afines a la inteligencia artificial

### **1.2.8 PROGRAMACIÓN DE VON NEUMANN**

Un programa de computación clásico, determinista de John Von Neumann sigue un algoritmo definido, a pesar de ser el primero en haber concebido los sistemas expertos el cual especifica como calcular las salidas para cualquier entrada; en

cambio el comportamiento de un programa de inteligencia artificial no es descrito explícitamente por un algoritmo, la secuencia del programa es definido por el problema específico.

Los programas de inteligencia artificial consideran relaciones del mundo real y del ámbito del conocimiento en que ellos operan, a diferencia de los programas con algoritmos específicos como para cálculos científicos, los programas de inteligencia artificial pueden aún aplicarse a datos y problemas mal estructurados, por ejemplo en la resolución de conflictos como en planificación, o el análisis de tareas de un sistema real con poca información, con una solución cercana siendo un modelo estocástico.

### **1.2.9 LOS SENTIDOS Y LA INTELIGENCIA ARTIFICIAL**

La inteligencia artificial I. A. tiene que ver con varias áreas por ejemplo la robótica, el lenguaje, la visión que cada vez más se utilizan en líneas de ensamblaje, y sistemas computacionales expertos, éstos reproducen el comportamiento de razonamiento del cerebro humano; existen muchos programas que sirven por ejemplo para el diagnóstico de infecciones sanguíneas y hasta pueden prescribir un tratamiento adecuado, los que interpretan datos sísmológicos, los que interpretan el comportamiento climatológico o los que configuran equipos de alta tecnología.

Tareas que asumidas por dispositivos mecánicos generan una significativa reducción de costos, eliminan riesgos en la manipulación por seres humanos de sistemas peligrosos, eventualmente ayudan a mejorar el desempeño de las personas con capacidades especiales, y se está utilizando de manera eficaz en los procesos de mejora continua de las empresas, específicamente en el control de calidad de productos.

### **1.3 ESTADO DEL ARTE DE LAS REDES NEURONALES**

Las redes neuronales artificiales surgieron del movimiento conexionista, que nació junto con la inteligencia artificial simbólica o tradicional en los años de 1950, la inteligencia artificial simbólica se basa en que todo conocimiento se puede representar mediante combinaciones de símbolos, derivadas de otras combinaciones que representan verdades incuestionables o axiomas.

### **1.3.1 I. A. TRADICIONAL VS. CONEXIONISTAS**

La inteligencia artificial tradicional asume que el conocimiento es independiente de la estructura que maneje los símbolos, en contraposición, los conexionistas intentan representar el conocimiento desde el estrato más básico de la inteligencia, lo físico, creen que el aprendizaje y el conocimiento se hallan directamente relacionados con la estructura del cerebro, concretamente con las neuronas y la interconexión entre ellas, y que trabajan en grupos, llamadas redes neuronales.

### **1.3.2 REDES NEURONALES PARADIGMA DE PROGRAMACIÓN**

Las redes neuronales artificiales son un paradigma de programación que imita el comportamiento de aprendizaje y procesamiento en forma automática de una neurona humana. Hecho importante porque permite solucionar problemas con un alto nivel de complejidad.

La solución de problemas complejos en un computador mediante algoritmos tradicionales inspirados en la filosofía de Von Neumann como una secuencia de pasos consumen demasiados recursos computacionales al tratar de resolverlos como una secuencia de pasos, ahora la evolución de un sistema computacional inspirado en el cerebro humano, utilizando redes neuronales permite que haciendo sinergias con otras herramientas como la lógica difusa, los algoritmos genéticos o los sistemas expertos, permitan soluciones altamente eficaces con bajo consumo de recurso computacional, lo cual explica su importancia en la robótica.

### **1.3.3 EL CEREBRO ARTIFICIAL**

A pesar de las desilusiones de los años 70 el desarrollo del arte continúa, y lo más avanzado que se tiene hoy en estos días es el modelaje de cerebros artificiales como por ejemplo en el centro para la neurociencia teórica de la Universidad de Waterloo ha desarrollado la simulación cerebral más grande del mundo, un grupo de investigadores canadienses han logrado crear el modelo de cerebro funcional más grande del mundo, el modelo, llamado Spaun.

Spaun es capaz de ver, recordar, pensar e incluso escribir mediante un brazo mecánico, Chris Eliasmith de la Universidad de Waterloo asegura que Spaun



vive en un mundo ficticio con físicas simuladas, la máquina utiliza únicamente su cerebro para comprender las imágenes que ve y mover su brazo virtual y dibujar con un lápiz el signo aprendido; que actúa de la misma forma en que un cerebro humano procesaría la información.

Spaun, es capaz de reconocer números, almacenarlos y después escribirlos en un papel, puede pasar las pruebas de un test de inteligencia, en comparación con otros modelos, este sistema puede realizar varias funciones. Spaun tiene 2,5 millones de neuronas y está construido de forma que éstas piensen en los patrones que se encuentran y hagan ocurrir cosas en su entorno, para la construcción fue necesario crear sistemas cerebrales específicos, para lo que hubo que recurrir a métodos de distintos campos, como la ingeniería, la programación, la biología, la filosofía, la psicología o la estadística.

### **1.3.4 MÉTODOS DE CONTROL NO LINEAL**

La Universidad de Waterloo, está ofertando su segunda escuela de verano para el 2015 cuyo principal objetivo es que los participantes aprendan a usar el paquete de simulación Nengo, utilizado para construir el modelo de cerebro funcional Spaun, que es una poderosa herramienta computacional para la simulación de sistemas cognitivos y neuronales.

Con este software se pueden llevar a cabo comprobaciones de hipótesis, implementación de funciones específicas de comportamiento con neuronas artificiales, expansión y mejora de modelos pasados, proporcionar pruebas de concepto de diferentes mecanismos neuronales, construir con modelos de neuronas enriquecidas, simular modelos a gran escala en una variedad de niveles de detalle.

Con Nengo se puede realizar modelos anatómicos y datos electrofisiológicos, establecer el modelo neuronal detallado para datos cognitivos y conductuales, utilizar variedades de modelos de células individuales dentro de un modelo a gran escala, implementar métodos de control no lineal modernos en un modelo neuronal.

### **1.3.5 SOFTWARE DE SIMULACIÓN PARA I. A.**

Existen diferentes tipos de simuladores como el Nest, Briam, Neuron y Nengo que es un paquete de software basado en gráficos y secuencias de comandos para la simulación de sistemas neuronales a gran escala, para utilizar Nengo, se definen grupos de neuronas en términos de lo que representan, y luego se procede a formar conexiones entre grupos neuronales en términos de lo que el cálculo debe realizar en esas representaciones; utiliza el marco de ingeniería neural para resolver para los pesos de las conexiones sinápticas adecuadas para lograr el resultado deseado, es compatible con varios tipos de aprendizaje, ayuda a que los modelos detallados de las neuronas enriquecidas que implementan algoritmos cognitivos de alto nivel complejas sean más eficaces, ha sido utilizado para implementar el control motor, la atención visual, memoria de serie, la selección de la acción, la memoria de trabajo, redes de atractor, el razonamiento inductivo, integración camino, y la planificación con la resolución de problemas.

### **1.3.6 AUTOAPRENDIZAJE**

En agosto de 2014, Google prepara un sistema de inteligencia artificial que se nutre de internet llamado knowledge Vault, capaz de aprender por si solo y conectar la información de la red, es la última tecnología que se encuentra ya en desarrollo en la compañía, esta bóveda del conocimiento se convertirá en la fuente de información más grande del mundo.

knowledge Vault consiste en una base de datos que almacenará información leída desde el internet de manera autónoma, agrupando gran cantidad de contenidos de manera informática, Knowledge Vault podría comenzar a relacionar lo que vaya aprendiendo y a medida que pase el tiempo, almacenar y conectar mejor los datos, en un proceso de auto-aprendizaje que le permitirá ordenar la información de manera eficiente, porque Knowledge Vault de google o Watson de IBM tratarán la información como hechos y no como cifras, por lo que será mucho más inteligente al procesarla que cualquier otra base de datos, haciendo lo que a nivel personal se hace con las redes sociales a nivel empresarial y científico.

Lo que viene a muy corto plazo será contar con tecnología integrada en sistemas portátiles de inteligencia artificial como el Google Now o Siri de Apple,

volviéndolos asistentes verdaderamente inteligentes capaces de responder cuestiones complejas, lo cual va a más allá, incluso se podría generar predicciones, basadas en los hechos históricos.

## **1.4 FUNDAMENTOS NEURONALES**

Se realiza una introducción a las neuronas cerebrales del cuerpo humano, si bien es cierto son células presentes en todo tipo de animal, aquí se centra el estudio principalmente a las funciones de éstas en el cerebro humano, considerando también que no están presentes solo en el cerebro sino a lo largo de todo el sistema nervioso.

### **1.4.1 NEURONAS**

La neurona (del griego νεῦρον, cuerda, nervio) es la unidad estructural y funcional del sistema nervioso, recibe los estímulos provenientes del medio ambiente, los convierte en impulsos nerviosos y los transmite a otra neurona, a una célula muscular o glandular donde se producirá una respuesta, son un tipo de células cuya función es la excitabilidad eléctrica de su membrana plasmática; están especializadas en la recepción de estímulos y conducción del impulso nervioso (en forma de potencial eléctrico) entre ellas o con otras, como por ejemplo las fibras musculares.

### **1.4.2 CARACTERÍSTICAS MORFOLÓGICAS DE LAS NEURONAS**

Las neuronas presentan características morfológicas típicas que sustentan sus funciones: un cuerpo celular llamado soma; prolongaciones cortas que transmiten impulsos hacia el soma, denominadas dendritas; y una prolongación larga, denominada axón, que conduce los impulsos desde el soma hacia otra neurona u órgano, un diagrama básico de una neurona se muestra en la figura 1.1.



Figura 1.1 Diagrama Básico de una Neurona<sup>2</sup>

La doctrina de la neurona, establecida por Santiago Ramón y Cajal a finales del siglo XIX, es el modelo aceptado hoy en neurofisiología, consiste en que la base de la función neurológica radica en las neuronas como entidades discretas, cuya interacción denominada sinapsis, conduce a la aparición de respuestas complejas.

### 1.4.3 SINAPSIS NEURONAL

Las neuronas tienen la capacidad de comunicarse con precisión, rapidez y a larga distancia con otras células, ya sean nerviosas, musculares o glandulares; a través de las neuronas se transmiten señales eléctricas denominadas impulsos nerviosos, estos viajan por toda la neurona comenzando por las dendritas, hasta llegar a los botones terminales, que pueden conectar con otra neurona, fibras musculares o glándulas, la figura 1.2 muestra la sinapsis en los terminales de las

---

<sup>2</sup> <http://www.cwi.nl/neuroinformatics>

neuronas, mecanismo que involucra, movimientos de proteínas cuyo inventario fue determinado por Seth Grant en Cambridge en 2010.

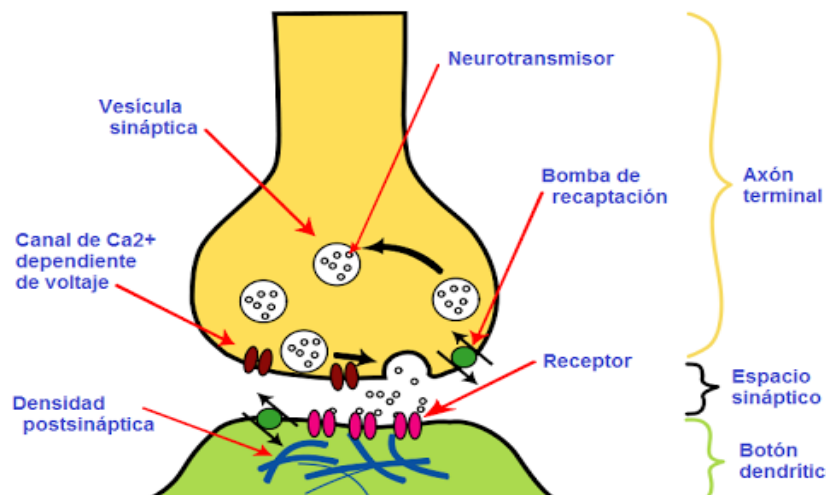


Figura 1.2 Sinapsis en los terminales de las neuronas<sup>3</sup>

#### 1.4.4 ACCIÓN MOTORA POR ACTIVIDAD NEURONAL

Las neuronas conforman e interconectan los tres componentes del sistema nervioso: sensitivo, motor e integrador o mixto; así, un estímulo que es captado en alguna región sensorial entrega cierta información que es conducida a través de las neuronas y es analizada por el componente integrador, el cual puede elaborar una respuesta, cuya señal es conducida a través de las mismas y la respuesta es ejecutada mediante una acción motora, como la contracción muscular o secreción glandular.

Las neuronas transmiten ondas de naturaleza eléctrica originadas como consecuencia de un cambio transitorio de la permeabilidad en la membrana plasmática, su propagación se debe a la existencia de una diferencia de potencial o potencial de membrana que surge debido a las distintas concentraciones de iones a ambos lados de la membrana.

#### 1.4.5 CIRCUITOS NEURONALES Y REDES NEURONALES

Las neuronas se agrupan dentro de circuitos neuronales, y la señal eléctrica, que es un potencial eléctrico, de una neurona se ve afectada por las neuronas del

<sup>3</sup> <http://upload.wikimedia.org/wikipedia/commons/a/a0/Sinapsis.png>

circuito a las que está conectada y su estado, el de una neurona dentro de un circuito cambia con el tiempo, y se ve afectada por tres tipos de influencias, las neuronas excitadoras del circuito neuronal, las neuronas inhibitoras del circuito neuronal y los potenciales externos que tienen su origen en neuronas sensoriales.

#### **1.4.6 PATRÓN DE ACTIVACIÓN NEURONAL**

La función de un determinado grupo de neuronas es alcanzar un determinado estado final en función de los estímulos externos, por ejemplo, en la percepción de color, un grupo de neuronas puede terminar en un determinado estado si el estímulo es uno y otro si el estímulo es diferente.

El número de estados estables posibles del circuito neuronal corresponde al número de patrones que puede reconocer el circuito neuronal, los trabajos de Freeman en los años 1990 aclararon que un determinado grupo de neuronas sigue un patrón de evolución caótico hasta alcanzar un determinado estado.

Un estado estable corresponde con el reconocimiento de un patrón, a nivel microscópico, el estado estable es un patrón de activación neuronal dentro de determinado circuito, en el que, el potencial de activación está cerca de un valor atractor de la neurodinámica del grupo.

#### **1.4.7 REPETICIÓN DE LA ACTIVIDAD NEURONAL**

El número de patrones reconocibles por un grupo de neuronas se puede relacionar con el número de neuronas que forman el grupo y la probabilidad de error en el reconocimiento de dicho patrón; las personas más hábiles o más entrenadas en una tarea ejecutan la misma tarea con mucha mayor precisión porque tienen un mayor número de neuronas encargadas de dicha tarea.

La repetición espaciada de una actividad refuerza las sinapsis y el número de neuronas potencialmente involucradas en una tarea, lo que refleja por ejemplo que un deportista de élite ejecute con una probabilidad de error muy pequeña determinada tarea, porque su entrenamiento hace que un mayor número de neuronas estén involucradas en dicha tarea lo que minimiza la probabilidad de error.

### **1.4.8 REESTRUCTURACIÓN NEURONAL**

El aprendizaje se da cuando por efecto de los patrones de activación reiterados, las conexiones neuronales sufren una reestructuración; ciertas conexiones sinápticas se refuerzan mientras otras se debilitan y el conocimiento que un individuo tiene del mundo se refleja en la estructura de estas conexiones; a su vez el número y el tipo de conexión determina el número de atractores disponibles de la neurodinámica de un circuito y por tanto el número de patrones diferentes que dicho circuito puede identificar.

El olvido y la pérdida de las capacidades motoras en algunas personas, tienen una base fisiológica en el debilitamiento de sinapsis poco usadas, por lo que cuando un determinado circuito neuronal se activa poco, sus sinapsis decaen y pueden perderse por lo cual el reconocimiento de cierto patrón puede llegar a perderse.

### **1.4.9 VELOCIDAD DE RESPUESTA MUSCULAR**

El impulso nervioso se transmite a través de las dendritas y el axón, la velocidad de transmisión del impulso nervioso, depende de la velocidad de conducción del axón, la cual depende a su vez del diámetro del mismo; el axón lleva el impulso en una sola dirección y es transmitido de un espacio a otro, las dendritas son las fibras nerviosas de una neurona que reciben los impulsos provenientes desde otras neuronas, en las grandes neuronas alfa de las astas anteriores de la médula espinal, las velocidades de conducción axonal pueden alcanzar hasta 120 m/s, y si se considera la talla de una persona, al impulso eléctrico le tomaría unos pocos milisegundos en recorrer desde la punta del pie hasta el cerebro.

### **1.4.10 LA CAPACIDAD DEL SISTEMA NERVIOSO**

Una red neuronal se define como una población de neuronas físicamente interconectadas o un grupo de neuronas aisladas que reciben señales que procesan a la manera de un circuito reconocible, la comunicación entre neuronas, que implica un proceso electroquímico, una vez que una neurona es excitada a partir de cierto umbral, ésta se despolariza transmitiendo a través de su axón una señal que excita a neuronas aledañas, y así sucesivamente.

El sustento de la capacidad del sistema nervioso, radica en las conexiones sinápticas; en oposición a la red neuronal, se refiere a un circuito neuronal cuando se hace mención a conjunto de neuronas que se controlan dando lugar a una realimentación; Los circuitos neuronales representan el substrato anatómico en el que se realizan todas las funciones del sistema nervioso, sensoriales, motores, cognitivos, de regulación de modulación, cada tipo presenta características particulares, que dependen de las propiedades de las neuronas que los forman y de las sinapsis que ellas forman. Figura 1.3.

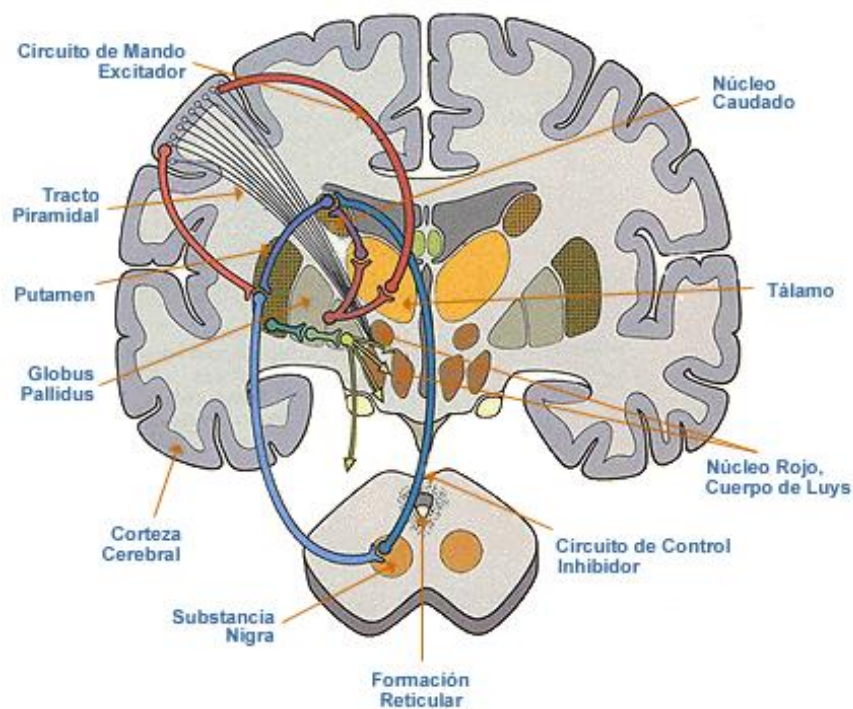


Figura 1.3 Circuitos Neuronales<sup>4</sup>

## 1.5 NEURONAS ARTIFICIALES

“La aparición de las computadoras digitales y el desarrollo de las modernas teorías del aprendizaje y el procesamiento neuronal se produjeron casi al mismo tiempo, al final de los años cuarenta”<sup>5</sup>, los PC han sido utilizados para modelar neuronas individuales y también grupos de neuronas llamadas redes neuronales, existen innumerables investigaciones neurofisiológicas y de redes neuronales artificiales en el campo investigativo biomédico.

<sup>4</sup> [http://www7.uc.cl/sw\\_educ/neurociencias/html/017.html](http://www7.uc.cl/sw_educ/neurociencias/html/017.html)

<sup>5</sup> FREEMAN J. SKAPURA D., (1993), Redes Neuronales, algoritmos, Aplicaciones y técnicas de programación. Ed, Addison Wesley. Díaz de Santos , Salamanca



A pesar de la gran ilusión inicial y el apareamiento del Perceptrón en 1958 por Frank Rosenblatt el desarrollo de las redes neuronales artificiales no llegó a tener un avance significativo pues podía haber quedado para el olvido ya que en 1969 Marvin Minsky y Seymour Papert publican el libro Perceptrons mostrando la debilidad del modelo, a no ser por gente que se mantuvo por más de una década empeñada en salir adelante con los estudios de las redes neuronales, hasta que en 1986 se publica la obra Parallel Distributed Processing de Rumelhart y McClelland orientado a temas fisiológicos y neurológicos, contiene las definiciones fundamentales del estudio de las redes neuronales artificiales.

### **1.5.1 ENTRENAMIENTO DE LAS NEURONAS**

En las primeras etapas de vida, el niño realiza el aprendizaje del cerebro, entrena sus neuronas mediante el éxito o fracaso de una acción a estímulos sensoriales, cuando cierta acción realizada en respuesta a alguna entrada sensorial es exitosa por ejemplo, al comer se sacia el hambre, las conexiones sinápticas entre un grupo de neuronas se fortalecen, de manera que cuando aparece una sensación sensorial parecida, la salida será la entrenada, de esta manera se forman fuertes conexiones entre grupos de neuronas, que pueden servir para realizar otras acciones más complejas.

El conocimiento de las redes neuronales biológicas ha dado lugar a un diseño de neuronas artificiales empleado en inteligencia artificial, estas redes funcionan porque cada neurona recibe una serie de entradas a través de interconexiones y emite una salida, la cual viene dada por tres funciones que son: de propagación, de activación y transferencia.

### **1.5.2 LAS FUNCIONES DE UNA NEURONA ARTIFICIAL**

La función de propagación que por lo general consiste en la sumatoria de cada entrada multiplicada por el peso de su interconexión; con este valor de suma ponderada se calcula una función de activación, que será la salida que dará la neurona; las dos funciones de activación más usadas son el escalón y la sigmoidea, se diferencian en que la sigmoidea (por su forma de S) es continua y diferenciable en todos sus puntos y la función escalón no.

Una función de activación, que modifica a la anterior y que puede no existir, siendo en este caso la salida la misma función de propagación; y una función de

transferencia, que se aplica al valor devuelto por la función de activación, se utiliza para acotar la salida de la neurona y generalmente viene dada por la interpretación que se quiera darle a dichas salidas para lo que se necesita conocer de antemano el rango de las respuestas que se espera pueda aprender el sistema inteligente para lo que se apoya en los conocimientos específicos sobre lo que se desee discriminar.

### **1.5.3 APLICACIONES EN ROBOTS MANIPULADORES**

Actualmente se está utilizando para el control de brazos articulados robóticos para la selección de productos por su calidad o presentación, por ejemplo el matiz aceptable de coloración de frutas, rosas así como el tamaño u otras características usualmente variables no discretas o no, sobre las cuales el sistema de control pueda decidir sobre la clasificación de un producto; dependiendo de las especificaciones dadas por los usuarios o clientes y de acuerdo con una persona experta el sistema computarizado, apoyándose en cálculos iterativos en el ambiente nntool de Matlab, el sistema robotizado podrá llegar a aprender.

### **1.6 MODELO NEURONAL DE McCOLLUCH Y PITTS**

En 1943 Warren S. McCulloch, un neurocientífico, y Walter Pitts, matemático estudioso de la lógica, publicaron el artículo: "Un cálculo lógico de las ideas inmanentes en la actividad nerviosa"<sup>6</sup>. En este trabajo McCulloch y Pitts trataron de explicar cómo el cerebro podría producir patrones altamente complejos mediante el uso de muchas células básicas que están conectadas entre sí, estas células básicas del cerebro que son las neuronas; McCulloch y Pitts dieron un modelo muy simplificado de lo que sería una neurona artificial en su artículo.

El modelo de una neurona, conocida como una neurona de MCP por sus siglas McCulloch y Pitts, ha hecho una importante contribución al desarrollo de las redes neuronales artificiales, la cual es el modelo matemático de las características clave de las neuronas biológicas.

---

<sup>6</sup> [http://centrodeartigo.com/articulos-para-saber-mas/article\\_41787.html](http://centrodeartigo.com/articulos-para-saber-mas/article_41787.html)

### **1.6.1 LIMITACIONES INICIALES**

El modelo original neuronal de McCulloch y Pitts, tenía limitaciones, se agregaron características adicionales que le permitió aprender y el siguiente desarrollo importante en las redes neuronales fue el concepto de un perceptrón que fue presentado por Frank Rosenblat en 1958 esencialmente el perceptrón es una neurona de McCulloch y Pitts donde las entradas se pasan primero a través de algunos “pre-procesadores”, que reciben el nombre de unidades de asociación, éstas detectan la presencia de ciertas características específicas en las entradas, un perceptrón estaba destinado a ser un dispositivo de reconocimiento de patrones, y las unidades de asociación corresponden a la característica o detectores de patrones.

### **1.6.2 RED NEURONAL**

Se puede comprar una neurona de McCulloch y Pitts en una tienda de electrónica, conocidos como unidades lógicas umbral, un grupo de neuronas de McCulloch y Pitts que están conectadas entre sí se denomina una red neuronal artificial; en cierto sentido, el cerebro es una red neuronal muy grande, tiene miles de millones de neuronas, y cada neurona está conectada a miles de otras neuronas. McCulloch y Pitts mostraron cómo codificar cualquier proposición lógica de una red adecuada de las neuronas de MCP, y así en teoría todo lo que se pueda hacer con un ordenador también se puede hacer con una red de neuronas MCP.

McCulloch y Pitts demostraron que todas las redes de neuronas MCP codifican una proposición lógica, así que si el cerebro fuera una red neuronal, entonces podría codificar un programa de ordenador complicado, pero la neurona MCP no es una neurona real es sólo un modelo simplificado por lo que hay tener cuidado al sacar conclusiones sobre las neuronas reales basados en las propiedades de las neuronas artificiales.

### **1.6.3 LEJOS DE LA CONCEPCIÓN CLÁSICA**

Un estudio de la neurona McCulloch Pitts MCP es esclarecedor por varias razones la historia temprana del desarrollo de lo que se convertiría en los impresionantes programas de redes neuronales artificiales utilizados en la actualidad, se nota la evolución de un modelo informático que poco a poco se

aleja de la concepción clásica de un ordenador digital para la aparición de una computadora no clásica que están haciendo una contribución significativa en la investigación que se está haciendo para comprender mejor el funcionamiento del cerebro humano, se aprecia la contribución de vital importancia que los matemáticos han hecho al estudio científico de la mente.

#### **1.6.4 RECREACIÓN LAS NEURONAS DE McCULLOCH Y PITTS<sup>7</sup>**

Con la finalidad de recrear las neuronas de McCulloch y Pitts, por sus siglas MCP, a continuación se expone un ejemplo tomado de la publicación digital THE MIND PROJECT CURRICULUM cuya autoría es de Michael Marsalli.

##### **1.6.4.1 Reconocimiento de Objetos con una neurona MCP**

suponiendo que hay una neurona en el cerebro de un ave que tiene dos receptores, que están conectados a los ojos, si el ave ve un objeto redondo, se envía una señal al primer receptor, pero si es de cualquier otra forma, no se envía ninguna señal, así que el primer receptor es un detector de redondez; y, si el ave ve un objeto púrpura, se envía una señal al segundo receptor de la neurona, pero si el objeto es de cualquier otro color, entonces no se envía ninguna señal; así que el segundo receptor es un detector de color púrpura.

El primer receptor corresponde a la pregunta ¿el objeto es redondo? el segundo receptor corresponde a la pregunta ¿el objeto es púrpura?, una neurona de MCP que le dijera al ave a comer un arándano, en otras palabras, se desea que la neurona envíe una señal de comer, si el objeto es a la vez redondo y púrpura, pero la neurona no enviará ninguna señal si el objeto no es ni redondo ni púrpura.

Por tanto, el ave sólo va a comer un objeto si la neurona envía una señal apropiada. La figura 1.3 muestra la representación gráfica de la neurona de McCulloch y Pitts con el ejemplo del ave y Tabla 1.1 muestra cómo se comportaría una neurona MCP en varios casos.

---

<sup>7</sup> <http://www.mind.ilstu.edu/curriculum/modOverview.php?modGUI=212>

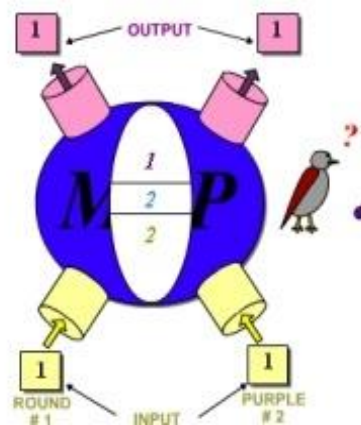


Figura 1.4 Representación gráfica de la neurona de McCulloch y Pitts<sup>8</sup>

Tabla 2.1 Respuesta de una neurona MCP.

Objeto	Púrpura	Redondo	Comer
Arándano	Si	Si	Si
Pelota de Golf	No	Si	No
Flor Violeta	Si	No	No
Salchicha	No	No	No

Considerando que todas las señales enviadas a la neurona y la señal que se envía son todas señales sí o no, esta característica de todo o nada es uno de los supuestos que McCulloch y Pitts hicieron sobre el funcionamiento de una neurona real.

El Modelo de McCulloch y Pitts también supone que de alguna manera una neurona real suma las señales de todos sus receptores, y se decide si se debe enviar un sí o un no, respuesta basada en el total de las señales que recibe; si el total de las señales recibidas es lo suficientemente alta, la neurona envía una señal de sí; de lo contrario, la neurona envía una señal de no; con el fin de sumar las señales que la neurona está recibiendo, se utiliza el número 1 para un sí y el número 0 para un no; como se indica en la tabla 1.2, obteniendo a una tabla de verdad and.

<sup>8</sup> <http://www.mind.ilstu.edu/curriculum/>

Tabla 1.3 Reemplazo de los valores de la tabla 1.1 si por 1 y 0 por no.

Objeto	Púrpura	Redondo	Comer
Arándano	1	1	1
Pelota de Golf	0	1	0
Flor Violeta	1	0	0
Salchicha	0	0	0

La manera de decidir si el total de las señales recibidas es suficientemente alto McCulloch y Pitts utilizaron un número que llamaron umbral, cada neurona tiene su propio umbral que se compara con la suma de las señales recibidas, si la suma es mayor que o igual al umbral, entonces la neurona enviará un 1 (es decir, una señal de sí), y si la suma es menor que el umbral, entonces la neurona enviará un 0 (es decir un no), la neurona está respondiendo a la pregunta ¿la suma de señales recibida es mayor o igual al umbral?.

#### 1.6.4.2 Umbral de Activación

Con el fin de ver cómo funciona el umbral, se puede suponer que se tiene una neurona MCP con dos receptores conectados a los ojos de un ave, el un receptor es un detector de redondez y el otro es un detector de púrpura, del mismo modo que en el ejemplo anterior, como se desea que la neurona instruya al ave a comer arándanos pero no pelotas de golf, se necesita un umbral suficientemente alto como para que se garantice que las dos propiedades están presentes, sea entonces un umbral de 2.

Si el ave ve un arándano, entonces el detector púrpura envía un 1 y el detector de redondez envía un 1, es así que la neurona añade estas señales para obtener una entrada combinada de  $1 + 1 = 2$ , la neurona toma esta entrada total de 2 y lo compara con el umbral de 2, debido a que la entrada total ( $= 2$ ) es mayor que o igual al umbral ( $= 2$ ), la neurona MCP se activará y enviará una señal de salida de 1 (lo que significa, comer).

Pero, ¿cómo será el comportamiento del pájaro en presencia de una pelota de golf o en presencia de otros objetos?, decidir qué objetos puede comer y cuáles

evitar. Suponiendo que dentro del pájaro existe una simple neurona MCP que influye en lo que come y lo que no come, la neurona MCP está configurada actualmente con un umbral de 2.

Puesto que es factible de cambiar el umbral se puede colegir cómo la neurona controla el comportamiento alimenticio del ave, cómo el ave decide qué hacer con cuatro diferentes objetos que se le presentan. Se puede dar entonces diferentes valores para el umbral para producir diferentes tipos de comportamiento en el ave. ¿Dónde establecer el umbral de lo que el pájaro se comería 3 de los 4 objetos? ninguno de los objetos? todos los cuatro objetos?

El comportamiento de esta ave artificial depende de dos factores: la información que recibe del medio ambiente a través de sus dos detectores sensoriales y el umbral que se establece; los detectores pueden ser considerados como unos buscadores de propiedades particulares, la entrada combinada a la neurona de los detectores será mayor o menor dependiendo de cómo las propiedades sean detectadas por los sensores; sea o no la entrada total de los sensores será un número lo suficientemente alto como para hacer que la neurona se dispare es decir, para dar una salida de 1 en lugar de 0, depende de qué tan alto es el umbral.

En la tabla 1.3 en se observa que si se establece el umbral en 1, lo que el ave haría sería no solo comer el arándano que es su alimento natural sino que además se comería o pretendería comer las flores violetas y las pelotas de golf, lo que se ha hecho es una tabla de verdad OR inclusiva.

Tabla 1.4 Salidas de una Neurona MCP con umbral 1

Objeto	Púrpura	Redondo	Total	$b \geq 1$	Comer
Arándano	1	1	2	Si	1
Pelota Golf	0	1	1	Si	1
Flor Violeta	1	0	1	Si	1
Salchicha	0	0	0	No	0

Una neurona McCulloch y Pitts que sirve sólo para números positivos como entrada y se dice que la señal es de tipo excitatoria, sin embargo se puede dar a la neurona una mayor flexibilidad mediante la adición de los valores de entrada negativos y por tanto la señal puede ser también inhibitoria.

#### **1.6.4.3 Señales Excitatorias**

A las señales procedentes de los receptores del ave del ejemplo se pueden agregar otras señales que provienen otros receptores, estos tipos de señales se llaman excitatorias porque excita la neurona a enviar su propia señal; Las señales propias más las señales excitatorias que una neurona recibe, permitirá que el total se acercará más hasta el umbral de la neurona, y así más cerca estará la neurona de enviar su señal. Así como la neurona recibe señales cada vez más excitantes, se vuelve más y más excitada, hasta que se alcanza el umbral, y la neurona envía su propia señal.

#### **1.6.4.4 Señales Inhibitorias**

Existe también otro tipo de señal que la excitatoria, la cual tiene el efecto opuesto en una neurona, estas otras señales se denominan señales inhibitorias, y tienen el efecto de inhibir la neurona para el envío de una señal, cuando una neurona recibe una señal inhibidora, con la cual no se activará, se vuelve menos excitada, y necesitará más señales excitadoras para alcanzar el umbral, el efecto de las señales inhibitorias es el restar de la suma de las señales excitatorias, volviendo a la neurona más relajada y moviendo la neurona lejos de su umbral.

Se describe un ejemplo de una neurona de McCulloch y Pitts con una señal inhibitoria siguiendo en el mismo esquema del ave del ejemplo anterior: se considerará un tipo particular de ave, por ejemplo un petirrojo, esta ave tiene plumas rojas en su pecho y se siente seguro alrededor de los objetos de color rojo, incluyendo criaturas rojas como un cardenal, suponiendo que el cerebro del ave (artificial) tiene una neurona con dos receptores conectados a los ojos, normalmente esta ave huirá de cualquier otra criatura que ve; si el ave ve a otra criatura, una señal excitatoria se enviará al primer receptor, que tratará de hacerle huir, así que el primer receptor es un detector de criatura, y que excita al ave a huir, sin embargo, si la criatura que el ave ve es de color rojo un inhibidor de señal será enviada al segundo receptor, lo que evitará que el ave huya, así



que el segundo receptor es un detector de color rojo, y que inhibe que el ave huya.

Suponiendo un escenario en que el ave ve un gato negro, ¿qué pasaría? el detector de criatura va a enviar una señal excitadora a la neurona, y el detector de color rojo no enviaría ninguna señal entonces el pájaro huiría. Suponiendo un segundo escenario en que el ave ve un cardenal macho, el detector de criatura va a enviar una señal excitadora a la neurona, y el detector de color rojo enviaría una señal inhibitoria, así que el pájaro no huiría, debido a que la señal inhibitoria cancelaría la señal excitatoria. En un tercer escenario el objeto que se presenta al ave ve una salchicha, el detector de criatura no envía ninguna señal, en cambio el detector de color rojo envía una señal inhibitoria puesto que es de color rojo por tanto el ave no huiría. La tabla 1.4 muestra cómo esta neurona con las señales excitadoras e inhibitorias trabajaría en los casos descritos.

Tabla 1.5 Respuesta de una Neurona MCP a Diferentes Señales

<b>Objeto</b>	<b>Criatura</b>	<b>Rojo</b>	<b>Huye</b>
<b>Gato negro</b>	Si	No	Si
<b>Cardenal</b>	Si	Si	No
<b>Salchicha</b>	No	Si	No

A continuación se muestra cómo estas señales inhibitorias y excitatorias funcionan cuando la neurona MCP compara estas señales a su umbral, igual que antes, se va a utilizar un 1 si se envía una señal excitatoria, y un 0 si no se envía ninguna señal excitatoria; pero ahora se va a utilizar un -1 cuando se envía una señal inhibitoria, y un 0 si no se envía ninguna señal inhibitoria, debido a que se está utilizando un -1 para una señal inhibitoria, cuando se añade una señal inhibitoria a la suma de todas las señales recibidas, el efecto de la señal inhibitoria sobre el total es restar un 1 (la adición de un -1 es la mismo que restar un 1) así que cuando una neurona calcula el efecto total de sus señales, que se sumará un 1 al total de cada una de las señales excitadoras y añadir un -1 al total de cada una de sus señales inhibitorias, si el total de las señales excitadoras y las señales inhibitorias es mayor que o igual al umbral, entonces la neurona

enviará un 1 y si el total de las señales excitadoras y las señales inhibitoras es menor que el umbral, entonces la neurona enviará un 0.

Hay que notar de que esto no es como McCulloch y Pitts manejan el efecto de un inhibidor de señal, se ha cambiado su enfoque con el fin de facilitar la transición a las redes neurales modernas, de hecho de acuerdo a McCulloch y Pitts, si una neurona recibe una señal inhibitora, entonces no va a enviar una señal, es decir el efecto de cualquier señal inhibitoria es inducir a la neurona a enviar un 0.

Suponiendo que se tiene una neurona MCP conectada a un detector de criatura que envía una señal excitatoria y un detector de color rojo que envía una señal inhibitoria y suponiendo también que el umbral es 1, se puede haber elegido otro número para el umbral, para cada objeto en la Tabla 1.4, se puede calcular el total de las señales mediante la adición de un 1 para cada señal excitadora y un -1 para cada señal inhibitora, luego se compara el total al umbral.

Si el ave ve un gato negro, entonces el detector de criatura, que es excitatorio, envía un 1, porque el gato es una criatura, el detector de color rojo, que es inhibitorio, envía un 0, debido a que el gato no es rojo, porque hay una señal excitatoria, el total es  $1 + 0 = 1$ , al comparar este total de 1 con el umbral, debido que es igual a 1, la neurona enviará un 1, por lo que el ave huirá.

Si el ave ve un cardenal macho, el detector de criatura, que es excitatorio envía un 1, porque el cardenal es una criatura, el detector de color rojo, que es inhibitorio, envía un -1 ya que el cardenal es de color rojo, puesto que hay una señal excitatoria y una señal inhibitora, el total es de  $1 + -1 = 0$ , comparando este total de 0 con el umbral, debido a que 0 es menor que el umbral de 1, la neurona MCP enviará un 0, por lo que el ave no huirá.

Si el ave ve una salchicha, entonces el detector de criatura, que es excitatorio, envía un 0, porque la salchicha no es una criatura, el detector de color rojo, que es inhibitorio, envía un -1, porque la salchicha es de color rojo, debido a que no hay señal excitadora y una señal inhibitora, el total es  $0 + -1 = -1$ . Comparando este total de -1 con el umbral, debido a que -1 es menor que el umbral de 1, la neurona MCP enviará un 0, por lo que el ave no huirá. La tabla 1.5 muestra este cambio de notación.

Tabla 1.6 Cambio de notación Si = 1 No = 0 para MCP Umbral = 1

Objeto	Creatura	Rojo	Total	>=1	Huye
	Exitatoria	Inhibitoria			
<b>Gato negro</b>	1	0	1	Si	1
<b>Cardenal</b>	1	-1	0	No	0
<b>Salchicha</b>	0	-1	-1	No	0

Se puede hacer una neurona MCP que haga exactamente lo contrario de la primera neurona considerada; ahora se puede ver qué otros resultados se puede producir con una neurona MCP mediante el uso de diferentes combinaciones de señales excitatorias y señales inhibitorias con varios umbrales; por ahora sólo se tendrá en cuenta las neuronas que reciben dos señales. Entonces se puede formular la siguiente interrogante ¿Hay algún límite en el tipo de neuronas de MCP que se pueda hacer?, de hecho hay algunos límites, con sólo dos señales, sólo hay tres posibles combinaciones de señales excitatorias e inhibitorias.

Existen dos maneras de ajustar una neurona de MCP; se puede hacer señales ya sea excitatoria o inhibitoria, y se puede cambiar el umbral, al hacer diferentes opciones para estos ajustes, se puede hacer que las neuronas de MCP produzcan una variedad de resultados; por ejemplo, suponiendo que se desea construir una neurona que haga que el ave coma objetos que no son ni púrpura ni redondos, en otras palabras, se trata de una neurona que hace exactamente lo contrario de lo que se hizo en el primer ejemplo. (Ver tabla 1.2).

Se podría hacer tanto el detector de color púrpura y del detector de redondez sean señales inhibitorias y establecer el umbral en 0, si el objeto es un arándano, hay dos señales inhibidoras enviadas, así que el total es igual a  $-1 + (-1) = -2$ , comparando  $-2$  con el umbral de 0, se ve que  $-2$  es menor que 0, por lo que la neurona de enviará un 0, y el ave no comerá el arándano, los resultados de hacer lo contrario a lo propuesto en la tabla 1.2 se muestran en la tabla 1.6

Tabla 1.7 Efectos de Dos Señales Inhibitorias y Umbral  $b = 0$ 

Objeto	Púrpura Inhibitoria	Redondo Inhibitoria	Total	$\geq 0$	Comer
Arándano	-1	-1	-2	No	0
Pelota Golf	0	-1	-1	No	0
Flor Violeta	-1	0	-1	No	0
Salchicha	0	0	0	Si	1

El número de posibles umbrales es infinito, pero sólo hay unos pocos tipos de resultados que una neurona MCP puede producir, ya que sólo envía un 0 o 1, las neuronas MCP también están limitadas por el número de señales que reciben, si se obedece las señales de los tipos de detectores y sólo se concentra en si el detector envía una señal, se puede ver que sólo hay cuatro combinaciones de dos señales: ambos detectores envían una señal, la primera envía una señal y el segundo detector no, el segundo detector envía una señal y el primer detector no, y ninguno de los dos detectores envían una señal; en consecuencia, se demuestra que hay dieciséis posibles maneras de producir un 0 o 1 de cada uno de los cuatro pares de señales, así que hay a lo sumo dieciséis posibles tipos de neuronas de MCP con dos receptores. A continuación se considera en tipo particular de combinación de estos dieciséis la O exclusiva.

#### 1.6.4.5 OR Exclusiva

Sea un ave con dos detectores conectados a una neurona, el primer detector enviará una señal si el objeto es una criatura de cuatro patas, y el segundo detector envía una señal si el objeto es de color verde, se desea hacer una neurona de MCP que indicará al ave a huir si el objeto es de cuatro patas o verde, pero no ambos. Se puede empezar por tratar dos señales excitatorias con un umbral de 1 para lo que se va a considerar cuatro objetos: una rana, una serpiente verde, un gato negro, y una flor violeta, estos objetos dan las cuatro combinaciones posibles de 0 y 1 para cada par de señales, dado que todas las señales son excitatorias, el total es simplemente la suma de las dos señales, esto producirá los resultados que se muestran en la tabla 1.7.

Tabla 1.8 Combinación de Dos Señales Excitatorias

<b>Objeto</b>	<b>4 Patas Excitatoria</b>	<b>Verde Excitatoria</b>	<b>Total</b>	<b>&gt;=1</b>	<b>Huye</b>
<b>Rana</b>	1	1	2	Si	1
<b>Serpiente Verde</b>	0	1	1	Si	1
<b>Gato Negro</b>	1	0	1	Si	1
<b>Flor Violeta</b>	0	0	0	No	0

Si bien esto está cerca de lo que se desea, pero no lo es, así que el ave huirá de un objeto que es de cuatro patas, como el gato, o de un objeto que es de color verde, como la serpiente, pero el ave también huirá de cualquier objeto que es a la vez de cuatro patas y verde, lo que no se desea, así que esta combinación de señales y el umbral no funciona; lo que se trata de hacer es una neurona de MCP que produzca un 1 si cualquiera de las señales es un 1, pero produce un 0 si las dos señales son 0 o 1, el tipo particular de neurona MCP que se está tratando de hacer se llama una neurona MCP OR exclusiva, ya que envía un 1 si recibe un 1 de una señal o de la otra, pero no ambas, una OR exclusiva tendría 0,1,1,0 en la última columna de la Tabla 1.7. También es una OR inclusiva que envía un 1 si recibe un 1, o bien la señal de ambos, una O inclusiva tendría 1,1,1,0 en la última columna de la Tabla 1.7, ya se hizo antes una neurona MCP OR inclusiva en la Tabla 1.3.

Si se prueba las diferentes combinaciones de señales inhibitoras con varios umbrales de excitación, se va a ver que es imposible construir una neurona de MCP que produce una OR exclusiva, se puede demostrar matemáticamente que ninguna combinación de señales y los umbrales puede producir la OR exclusiva, es así que hay una limitación fundamental del tipo de resultados de una sola neurona MCP puede producir.

Con el fin de producir resultados más complicados como la OR exclusiva hay que empezar a conectar las neuronas de MCP entre sí para formar redes neuronales, resulta que la OR exclusiva se puede hacer mediante el uso de tres

neuronas de MCP cada una con dos receptores. Dos de las neuronas de MCP cada una tiene un receptor unido al primer detector y el otro receptor conectado al segundo detector, así que la misma señal de cada detector se envía a ambas neuronas MCP simultáneamente, cada una de estas neuronas de MCP a continuación, envían su señal a uno de los receptores de la tercera neurona MCP, si las distintas señales y los umbrales se eligen correctamente, esta red aceptará dos señales y enviar una señal que produce una OR exclusiva, en otras palabras, se puede producir una red neuronal de tres neuronas MCP que produzca una OR exclusiva.

Otra forma de aumentar los posibles resultados de una neurona MCP es utilizar varios receptores, se podría pensar a las neuronas de MCP con tres o más receptores, entonces se podría enlazar estas neuronas de MCP en una red neuronal, al hacer sólo esto, McCulloch y Pitts demostraron que hay una red neuronal de las neuronas de MCP que produce cualquier combinación de 1 y 0. Por tanto, mientras que hay una limitación fundamental con una sola neurona MCP, esta limitación puede ser superada mediante la conexión de las neuronas de MCP individuales juntas en una red neuronal, esto muestra cómo resultados complejos se pueden obtener mediante la conexión de un gran número de piezas muy simples (las neuronas MCP), tal vez esto explica cómo nuestro cerebro puede hacer cosas asombrosas, a pesar de que es una recopilación de un gran número de células básicas (neuronas reales) que están conectadas a un gran número de otras células básicas.

#### **1.6.4.6 Cómo Aprende una Neurona de McCulloch y Pitts**

Hasta aquí si se deseaba encontrar una neurona de MCP que tenía un cierto tipo de comportamiento, se tenía prácticamente que adivinar la combinación de señales y el umbral al que trabajarían; ahora se va a ver cómo una neurona MCP puede aprender, cómo producir el tipo de comportamiento que se desea, pero primero se tiene que hacer que la neurona MCP sea un poco más flexible, lo que se va a hacer no fue hecho por McCulloch y Pitts, pero es un paso importante en el desarrollo de las redes neuronales modernas.

Suponiendo que se tiene una neurona MCP conectada a dos detectores, sea el primer detector X1 y el segundo detector de X2 para abreviar, no se considerará

exactamente lo que los detectores están detectando, para notación se escribirá un 1 si el detector envía una señal a la neurona, y se va a escribir un 0 si el detector no envía ninguna señal, hay cuatro situaciones posibles para los dos detectores.

Ambos detectores pueden enviar una señal, en cuyo caso se tiene que  $X_1$  es 1 y  $X_2$  es 1, el primer detector envía una señal, pero el segundo detector no envía ninguna señal, en cuyo caso  $X_1$  es 1 y  $X_2$  es 0, el primer detector no envía ninguna señal, pero el segundo detector envía una señal, en cuyo caso  $X_1$  y  $X_2$  es 0 y 1. Finalmente, ningún detector envía una señal, en cuyo caso  $X_1$  y  $X_2$  son 0. Se puede resumir estas cuatro posibilidades en la tabla 1.8.

Tabla 1.9 Posibilidades de combinación de 2 señales con dos sensores

$X_1$	$X_2$
1	1
1	0
0	1
0	0

Ahora se desea considerar una manera de tomar en cuenta la posibilidad de que uno de nuestros detectores tenga más influencia en la neurona de MCP que el otro detector, en otras palabras, es posible que la señal de un detector puede ser mucho más fuerte que la otra señal, la forma en que se va a hacer esto es mediante la asignación de un peso a cada detector, el peso de un detector indicará cuánta influencia de la señal del detector tiene en la neurona de MCP.

El peso será un número decimal menor a 1, y puede ser positivo o negativo, un peso positivo significa que la señal es excitatoria, y un peso negativo significa que la señal es inhibidora, se usará  $W_1$  como abreviatura para el peso del primer detector, y se usará  $W_2$  como abreviatura para el peso del segundo detector, también se necesitará un umbral, igual que antes, el umbral también será un número decimal. A continuación se verá cómo la neurona MCP utiliza estos pesos.

De acuerdo a lo revisado en 1.5, se ha añadido las señales entrantes y compara la suma al umbral, se hará esto, excepto que en lugar de utilizar un 1 para una señal de entrada de un detector, se utilizará el peso para ese detector, si no se envía ninguna señal, se utilizará un 0, como se muestra en el ejemplo siguiente.

Suponiendo que el peso  $W_1$  es de 0,5, el peso  $W_2$  es -0,3, y el umbral es 0,1, estas son opciones sólo convenientes para los pesos y el umbral, cómo esta neurona MCP vaya a funcionar correctamente en cada una de las cuatro combinaciones posibles de los dos detectores.

Si ambos detectores envían una señal, es decir,  $X_1$  es 1 y  $X_2$  es 1, entonces se volvería a tomar el peso para el primer detector y el peso para el segundo detector y sumarlos, en este caso, el peso del primer detector es de 0,5 y el peso del segundo detector es -0,3, entonces si se suma se obtiene  $0,5 + (-0,3) = 0,2$ , ahora comparando esta cifra con el umbral de 0,1 debido a que el total de 0,2 es mayor que el umbral de 0,1, la neurona enviará una señal 1.

Si el primer detector envía una señal, pero el segundo detector no envía ninguna señal, es decir,  $X_1$  es 1 y  $X_2$  es 0, entonces sólo tiene que utilizar el peso del primer detector, y un 0 para el segundo detector, dado que el peso para el primer detector es de 0,5, cuando se añade las señales se obtiene  $0,5 + 0 = 0,5$ , comparando con el umbral de 0,1, debido a que el total de 0,5 es mayor que el umbral de 0,1, la neurona enviará una señal 1.

Si el primer detector no envía ninguna señal, pero el segundo detector envía una señal, es decir,  $X_1$  es 0 y  $X_2$  es 1, entonces se utiliza un 0 para el primer detector, y se usa el peso para el segundo detector, dado que el peso para el segundo detector es -0,3, cuando se añade las señales se obtiene  $0 + (-0,3) = -0,3$ , comparando esta cifra con el umbral de 0,1, debido a que el total -0,3 es menor que el umbral de 0,1, la neurona enviará un 0.

Finalmente, si ninguno de los detectores envía una señal, es decir,  $X_1$  es 0 y  $X_2$  es 0, luego el total es  $0 + 0 = 0$  otra vez entre el total y el umbral de 0,1 debido a que el total de 0 es menor que el umbral de 0,1, la neurona enviará un 0. De manera que si  $X_1$  es 1 y  $X_2$  es 1, entonces la neurona produce un 1. Si  $X_1$  es 1 y  $X_2$  es 0, entonces la neurona produce una 1. Si  $X_1$  es 0 y  $X_2$  es 1, la neurona produce un 0. Finalmente, si  $X_1$  es 0 y  $X_2$  es 0, entonces la neurona produce un



0, toda esta información se puede resumir en la tabla 1.9, donde se utiliza una función F como forma abreviada de la salida de la neurona MCP.

Tabla 1.10 Respuesta de una MCP con pesos 0.5 y -0.3 y Umbral 0.1

X1	X2	F
1	1	1
1	0	1
0	1	0
0	0	0

Afortunadamente, no hay que hacer todos los cálculos anteriores para las otras neuronas de MCP, pues existen programas computacionales como el nntools de Matlab que se pueden producir la tabla anterior cuando se ingresa los pesos y el umbral.

Una vez que se tiene las neuronas de MCP con pesos, se va a mostrar cómo una neurona MCP puede aprender para producir el comportamiento que se desee, este fue el siguiente gran paso en el desarrollo de la red neuronal moderna, la idea básica se debió a el psicólogo Donald Hebb en 1949, Hebb estaba tratando de explicar la manera de aprender en términos del comportamiento de las neuronas humanas.

Se pensó que si un detector de forma reiterada y persistente envía una señal a una neurona, haría que la neurona envíe su propia señal, entonces de alguna manera la conexión desde el detector a la neurona se vería reforzada, se puede traducir esta idea en una regla aprendizaje para una neurona MCP con pesos. En referencia al primer ejemplo (tabla 1.2), se tiene un ave con una neurona conectada a dos detectores, el primer detector envía una señal si el objeto es redondo, el segundo detector envía una señal si el objeto es de color púrpura, lo que se desea es que la neurona envíe una señal sólo cuando un objeto es a la vez redondo y de color púrpura, por lo que el ave sólo va a comer todo el tiempo, los objetos redondos y de color púrpura.

Esta neurona tiene ahora un umbral y dos pesos, un peso para cada detector, aquí se va a utilizar la idea de Hebb en la neurona para producir las señales correctas, si el objeto es redondo y de color púrpura y la neurona está enviando

una señal no se va a cambiar nada; si el objeto es redondo y púrpura y la neurona no está enviando una señal, se añadirá 1 a los pesos en los detectores, y se resta 1 desde el umbral; si el objeto no es redondo y púrpura y la neurona está enviando una señal, entonces se va a restar 1 los pesos de los detectores que también están enviando una señal, y se va a añadir 1 al umbral; si el objeto no es redondo y púrpura y la neurona no está enviando una señal, entonces no va a cambiar nada.

Nótese que cuando la neurona tiene el comportamiento correcto, no se cambiará nada, pero cuando la neurona no tiene el comportamiento correcto, se trata de cambiar los pesos y los umbrales de tal manera que se pueda ayudar a la neurona a producir el comportamiento correcto.

Cuando la neurona debería estar enviando una señal, pero lo está haciendo, se aumenta el peso de los detectores que están enviando una señal, y se baja el umbral, esto hace que el total de los pesos sea más grande y el umbral más pequeño, por lo que el total es más cercano a la superación del umbral, por lo tanto, la neurona está ahora enviando una señal, o se está poniendo más cerca de enviar una señal.

Cuando la neurona está enviando una señal, pero no debe, se bajan los pesos de los detectores que están enviando una señal, y se aumenta el umbral, esto hace que el total de los pesos sea más pequeño y el umbral más grande, por lo que el total es mayor que el umbral, así la neurona ya no envía una señal, o se está poniendo más cerca de no enviar una señal.

Desafortunadamente, un ajuste de los pesos puede no ser suficiente para producir el comportamiento que se desea en todas las circunstancias, puede que se tenga que repetir el ajuste de los pesos muchas veces por todas las circunstancias hasta que la neurona siempre produzca el comportamiento correcto en cada circunstancia esto es muy parecido a aprender; cuando se entera de alguna tarea, a menudo se tiene que practicar muchas veces hasta que se lo hace bien.

Una vez que se ha visto cómo funciona una neurona de McColluch y Pitts individual y se entrenara para un comportamiento deseado, se está listo para

considerar el siguiente desarrollo en la historia de las redes neuronales: el perceptrón que se analiza detalladamente en 3.3.

## **CAPÍTULO II**

### **REDES NEURONALES EN LA ROBÓTICA**

#### **2.1 GENERALIDADES**

La robótica es una tecnología multidisciplinaria de vanguardia que hace uso de los recursos de ciencias como mecánica, matemáticas, electrónica, informática, energía, visión e inteligencia artificial, lógica difusa, algoritmos genéticos y redes neuronales; para que un sistema mecánico se llame robot debe necesariamente ser inteligente o debe poseer cierta inteligencia que se debe normalmente a los algoritmos computacionales asociados con su sistema de control y sensorial; existen diferentes maneras de proporcionarle de inteligencia a una máquina, sea con lógica difusa, redes neuronales o una combinación de las dos; en este trabajo se ha optado por el control de movimientos de brazos manipuladores mediante redes neuronales.

##### **2.1.1 ROBÓTICA ETIOLÓGICA**

El ser humano ha buscado crear máquinas que puedan realizar las mismas tareas que él, su meta va más allá, lograr que éstas no sólo reproduzcan conductas inteligentes, sino que se realicen utilizando los mismos principios que se usan en los seres vivos y en particular en el hombre, esta ciencia llamada robótica etiológica o fisiológica pretende que la naturaleza indique los caminos, y los robots permiten a los investigadores entender algunas funciones imposibles de desentrañar directamente a través de la experimentación animada.

#### **2.2 TIPOS DE ROBOTS**

Existen diversos tipos de robots tales como los androides, que son los que se parecen a los seres humanos; los móviles, que están provistos de patas o llantas para desplazarse y se emplean para el transporte de productos en cadenas de producción; los médicos, que son básicamente prótesis que se adaptan al cuerpo humano; los tele operadores, que se controlan remotamente por un operador humano y son útiles en entornos peligrosos y los robots manipuladores o industriales; todos los robots están relacionados entre sí y algunos entran en más

de una clasificación; en esta investigación se considera a los robots manipuladores o industriales, sin embargo se hace una descripción de los otros tipos.

### 2.2.1 ROBOTS ANDROIDES

Son robots que se parecen a los seres humanos como que se tratara de muñecos y mientras más avanza la ciencia de materiales, se obtienen cada vez mejores acabados superficiales produciendo cada vez mejores semejanzas entre el material artificial y el natural, de manera que si se quiere determinar una diferencia entre el ser humano y la persona resulta a estas alturas del desarrollo de la ciencia algo difícil de apreciar a simple vista, más si los movimientos del robot pueden también simularse ya con bastante acercamiento a la realidad.

### 2.2.2 ÚLTIMA GENERACIÓN DE ROBOTS HUMANOIDES

En la figura 2.1 se observa una fotografía de una modelo humana y su robot Otonaroid una de las últimas creaciones del Dr. Hiroshi Ishiguro, donde la pregunta que se hace Mary Ann Rosen en su página web del 24 de junio de 2014, es cuál es humano y cuál es la máquina, donde se muestra el parecido; tomado de una muestra en el Museo Nacional de Ciencias Emergentes e Innovación (Miraikan) en Tokio que es un escaparate de algunos de los robots humanoides más realistas jamás vistos.

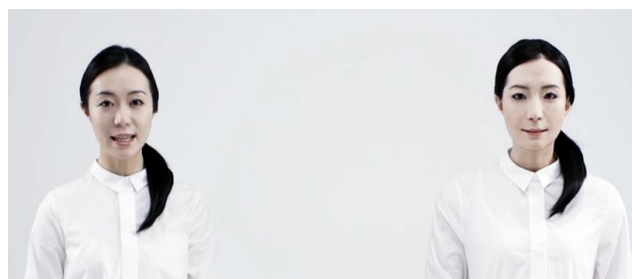


Fig. 2.1 Otonaroid, ¿Cuál es humano y cuál robot?<sup>9</sup>

Russon de la película Blade Runner tiene razón al pensar que estos robots androides de Japón, permiten vislumbrar el futuro cuando se refiere a tener miedo, el futuro de robots diseñados genéticamente mostrados en la película de Harrison Ford en 1982, es una realidad cada vez más cercana.

---

<sup>9</sup> <http://www.miraikan.jst.go.jp/en/exhibition/future/robot/android.html>

En el museo Miraikan Nacional de Innovación y Ciencias de Tokio se inauguró una exposición permanente denominada Androide, la novedad está en que los visitantes tendrán la oportunidad de interactuar con los androides más avanzados del mundo, donde se acaban de incorporar tres nuevos androides como personal del mismo, Otonaroid, Kodomoroid y Telenoid.

El Dr. Hiroshi Ishiguro profesor del Departamento de Sistemas de Innovación de la Universidad de Osaka, que ha estado desarrollando robots para más de 20 años, dice lo que significa para él, llevar a cabo estos desarrollos tiene una implicación científica y tecnológica importante, pero con repercusiones en el lado humano que vale la pena tomar en cuenta: “Hacer androides se trata de explorar lo que significa ser humano, examinar lo que es la emoción, lo que es la conciencia, lo que es el pensamiento”<sup>10</sup>. Esta obra de arte tiene la intención de conseguir que los visitantes a pensar más sobre el futuro de la humanidad.

Las interacciones de los visitantes con estos robots permitirán a los investigadores a explorar la interacción humano-robot, y para examinar lo que significa ser humano, Otonaroid se asemeja a una mujer adulta, y de acuerdo con el museo, ella ha sido contratada por el Miraikan como un robot comunicador de ciencias, los visitantes pueden interactuar con este robot al tener una conversación cara a cara con ella, y con el tiempo los investigadores esperan que la gente se acostumbre a establecer comunicación con los androides, de hecho en Japón desde el primer año de ingreso a la escuela los niños interactúan de una manera natural con los robots androides educativos.

La figura 2.2 muestra, una presentadora de noticias robot androide que se asemeja a una chica, Kodomoroid es un robot que se asemeja a una chica, es presentadora de noticias robot, capaz de recitar cualquier noticia recogida de todo el mundo en una variedad de voces e idiomas, las 24 horas del día.

---

<sup>10</sup> Hiroshi Ishiguro, "Telenoid : A Teleoperated Android with a Minimalistic Human Design", *In Robo Business Europe*, Billund, Denmark, May, 2014.



Fig. 2.2 Kodomoroid modelo y robot<sup>11</sup>

La figura 2.3 muestra una fotografía de Dr. Hiroshi Ishiguro y Mamoru Mori Director del museo posando con Kodomoroid y Otonaroid.



Fig. 2.3 Dr. Hiroshi Ishiguro y Mamoru Mori<sup>12</sup>

Los creadores de los robots utilizaron un tipo especial de silicona que se parece a la piel humana para replicar el cuerpo de la modelo, y gastar cientos de horas de moldeado, pintura y elaboración de una réplica fiel de los rasgos faciales de la modelo, al mismo tiempo, la maquinaria y programación compleja se ponen juntos de modo que cada parte del robot está programado para replicar gestos humanos realistas y las inflexiones faciales, de modo que el robot es capaz de moverse e interactuar de una forma tan realista como sea posible.

En el 2005 el Dr. Ishiguro se refería a su trabajo con robots “He desarrollado muchos robots antes, pero pronto me di cuenta de la importancia de su apariencia. Una apariencia similar a la humana da un robot una fuerte sensación de presencia. Repliee Q1Expo puede interactuar con la gente. Puede responder a la gente tocarlo. Es muy satisfactorio, aunque obviamente tenemos un largo

<sup>11</sup> <http://www.miraikan.jst.go.jp/en/>

<sup>12</sup> <http://www.miraikan.jst.go.jp/en/>

camino por recorrer”<sup>13</sup>, ahora casi 10 años después está haciendo realidad su sueño de darles personalidad y una verdadera apariencia humana con Otonaroid, Kodomoroid y Telenoid sus últimas creaciones.

### **2.2.3 ASIMO DE HONDA**

Otro robot androide y emblema del museo de ciencias de Tokio es ASIMO de Honda, ellos proponen la siguiente interrogante. ¿Por qué crear un robot humanoide?. El sueño parece simple, diseñar un robot que puede duplicar las complejidades del movimiento humano y realmente ayudar a la gente, al principio parecía una tarea fácil lo que no fue sí de ningún modo, ASIMO tomó más de dos décadas de persistente estudio, investigación y ensayo y error antes de los ingenieros de Honda lograron su sueño de crear un robot humanoide avanzado.

En 1986, los ingenieros de Honda se propusieron crear un robot que camina, los primeros modelos (E1, E2, E3) se centraron en el desarrollo de las piernas que puedan simular el caminar de un ser humano, la siguiente serie de modelos (E4, E5, E6) se centraron en la estabilización de caminar y subir escaleras; a continuación, se añadieron la cabeza, el cuerpo y los brazos al robot para mejorar el equilibrio y la funcionalidad.

El primer robot humanoide de Honda, P1 fue de 6 pies y 2 pulgadas de altura, y 386 libras. P2 mejorado con un diseño más amigable, la mejora de caminar, subir y bajar escaleras, y movimientos automáticos inalámbricos. El modelo P3 fue aún más compacto, de 5 pies y 2 pulgadas de altura y un peso de 287 libras.

ASIMO es la culminación de dos décadas de investigación en robótica humanoide por los ingenieros de Honda, este robot androide puede correr, caminar en pendientes y superficies irregulares, girar suavemente, subir escaleras, y alcanzar y agarrar objetos, también puede comprender y responder a comandos de voz simples, tiene la capacidad de reconocer la cara de un grupo seleccionado de personas, usando una cámara como sus ojos, puede asignar su entorno y registrar los objetos estacionarios, también puede evitar los obstáculos en movimiento a medida que avanza a través de su medio ambiente.

---

<sup>13</sup> Whitehouse, David, "[Japanese develop 'female' android](#)", BBC News, 27 July 2005



Dado que el desarrollo continúa Honda ASIMO demuestra a todo el mundo para alentar e inspirar a los jóvenes estudiantes a estudiar las ciencias, y en el futuro, ASIMO puede servir como otro conjunto de ojos, oídos, manos y piernas para todo tipo de personas necesitadas, algún día ASIMO puede ayudar con tareas importantes como la asistencia a las personas de edad o una persona confinada a una cama o una silla de ruedas, ASIMO también podría realizar ciertas tareas que son peligrosas para los seres humanos, tales como la lucha contra los incendios o la limpieza de derrames tóxicos. La figura 2.4 muestra la última versión de ASIMO presentada en Bruselas el 7 de julio de 2014.

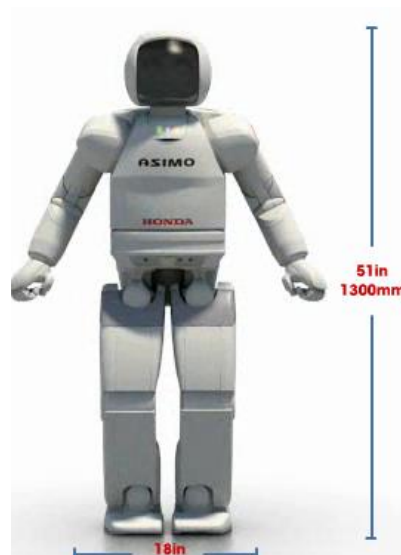


Fig. 2.4 ASIMO de Honda<sup>14</sup>

Los ingenieros de Honda ASIMO crearon el robot con 34 grados de libertad que le permiten caminar y realizar tareas muy parecidas a un ser humano con flexibilidad óptima, construido con materiales ligeros cuya estructura es de aleación de magnesio y plástico, provisto de potentes ordenadores y servomotores. Las especificaciones técnicas de este robot han ido cambiando desde su primera versión ya hace casi 10 años, fundamentalmente en lo referente a la disminución notable de su peso y el control o sistema inteligente incorporado.

El 7 de julio de 2014, en Bruselas, Bélgica que es la sede de las comunicaciones Honda Robótica en Europa, se realizó una presentación de ASIMO, frente a una

---

<sup>14</sup> <http://asimo.honda.com>

multitud de medios de comunicación reunidos, el totalmente nuevo ASIMO hizo su debut europeo, Honda espera que en el futuro será capaz de ayudar a las personas necesitadas, incorpora varios avances significativos respecto a sus predecesores, estos incluyen la mejora de la inteligencia, una mayor destreza manual lo que le permite abrir una botella y verter una bebida, así como la capacidad de correr más rápido, correr hacia atrás, subir y bajar escaleras sin problemas, incluso saltar.

Los avances de Honda van a la vanguardia hacia la creación de un robot para uso práctico en el entorno del hogar, o donde se requiere asistencia, ASIMO se introdujo por primera vez a Europa en el año 2003 en un simposio científico titulado: Alta Tecnología de Inteligencia, El Desafío de Robots Humanoides, celebrado en la Universidad Técnica de Darmstadt, desde entonces.

ASIMO ha viajado por toda Europa con una gira que ha incluido muchos eventos científicos y robótica, también se utiliza ampliamente para inspirar a los jóvenes acerca de la ciencia y la ingeniería, asistiendo a eventos como el Picnic Ciencia en Varsovia, eventos de ciencias al aire libre más grande del mundo y el evento Greenlight for Girls en Bruselas, para inspirar a las mujeres jóvenes a considerar un futuro en las matemáticas, la ciencia, la tecnología y la ingeniería.

### **2.3 ROBOTS MÓVILES**

En este tipo de robots se encuentran todos aquellos autómatas que tengan cualquier forma y normalmente están provistos de motores, turbinas, ruedas, patas o garras para su desplazamiento autónomo y tienen innumerables aplicaciones desde vehículos de exploración espacial, aviones no tripulados o drones, cohetes, misiles, naves marinas, etc.

Haciendo su debut europeo en el simposio Alta Tecnología de Inteligencia, El Desafío de Robots Humanoides fue, el  $\beta$  UNI-CUB (beta), un nuevo dispositivo de movilidad personal que cuenta con sistema de rueda motriz omnidireccional de Honda y utiliza la tecnología de control de balance acumulado de una amplia investigación de Honda en robots humanoides bípedos.

Existen algunas innovaciones tecnológicas de honda como el Honda's Stride Management Assist que es un dispositivo mecánico está diseñado para ayudar a aquellas personas con músculos de las piernas debilitadas, pero que todavía son capaces de caminar, un motor ayuda a levantar cada pierna a la altura del muslo mientras se desplaza hacia adelante y hacia atrás, permitiendo alargar la zancada del usuario, por lo que le es más fácil cubrir distancias más largas y con mayor rapidez. Entre las novedades también se presentó un vehículo bípodo para transporte individual y una prótesis para pierna, como se observa en la figura 2.5



Fig. 2.5 Innovaciones tecnológicas de ASIMO de Honda 2014<sup>15</sup>

### 2.3.1 TELEROBÓTICA ESPACIAL

Una de las aplicaciones muchos más aprovechadas de la robótica, y que ha seguido maravillando al hombre, es la telerobótica en el espacio extraterrestre, la organización más importante dentro de este aspecto, y que ha marcado un rumbo avanzado en cuanto a tecnologías e investigaciones, es la NASA (National Aeronautics and Space Administration).

El Programa de Telerobótica Espacial de la NASA, está diseñado para desarrollar capacidades en telerobótica para la movilidad y manipulación a distancia, uniendo la robótica y las teleoperaciones y creando nuevas tecnologías. Los requerimientos de tecnología de la robótica espacial pueden ser caracterizados por la necesidad del control manual y automático, tareas no repetitivas, tiempo de espera entre el operador y el manipulador, manipuladores flexibles con dinámicas complejas, nueva locomoción, operaciones en el

<sup>15</sup> <http://asimo.honda.com>

espacio, y la habilidad para recuperarse de eventos imprevistos. El Curiosity que llegó a Marte es el mayor exponente de este tipo, ver figura 2.6 donde se aprecia la cámara y el brazo robótico.

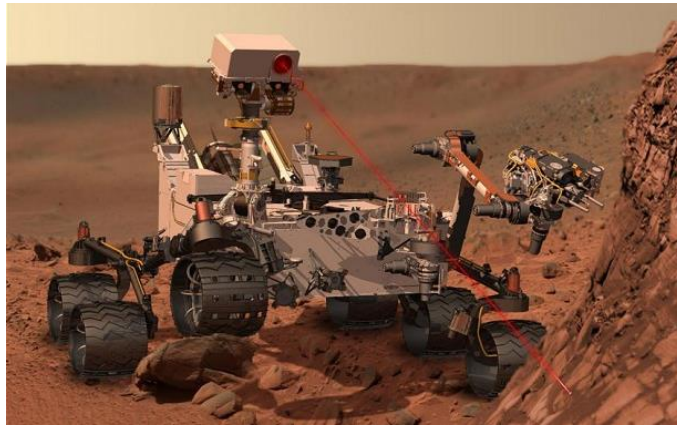


Fig. 2.6 Curiosity de la Misión Marte de la Nasa<sup>16</sup>

## 2.4 ROBOTS MANIPULADORES<sup>17</sup>

Los primeros dispositivos que responden a lo que hoy se conoce como robots manipuladores, inicialmente eran llamados manipuladores teleoperados, y se desarrollaron con el objetivo de manejar elementos radioactivos sin riesgo para el operador, la palabra robot fue utilizada por primera vez en 1921, por el escritor checo Karel Capek en su obra Rossum's Universal Robot, su origen es la palabra eslava robota, que se refiere al trabajo realizado de manera forzada.

### 2.4.1 PRIMERAS INVESTIGACIONES

Las primeras investigaciones que condujeron a los robots manipuladores para ensamble en la industria tanto automotriz como electrónica, se remontan al período que siguió a la Segunda Guerra Mundial, a fines de la década de 1940, comenzaron programas de investigación en Oak Ridge, donde uno de los proyectos más importantes desarrollados fue el conocido proyecto Manhattan y Argonne National Laboratories en USA, para desarrollar manipuladores mecánicos controlados de forma remota para manejar materiales peligrosos radiactivos.

---

<sup>16</sup> <http://www.nasa.gov/mission>

<sup>17</sup> <http://robotica.blogspot.com/2007/10/historia-de-la-robotica.html>

### **2.4.2 ROBOTS TIPO MAESTRO-ESCLAVO**

Los robots articulados eran del tipo maestro-esclavo, diseñados para reproducir los movimientos de mano y brazos realizados por un operador humano; el manipulador maestro estaba guiado por el usuario a través de una secuencia de movimientos, mientras que el manipulador esclavo duplicaba a la unidad maestra a manera de un pantógrafo; luego, se añadió la realimentación de la fuerza acoplada mecánicamente al movimiento de forma que el operador podía sentir las fuerzas que se desarrollaban entre el manipulador y su medio ambiente.

### **2.4.3 MANIPULADOR INDUSTRIAL**

Investigaciones de sistemas más sofisticados capaces de operaciones repetitivas autónomas; a mediados de los años 1950, George Devol desarrolló un dispositivo de transferencia articulada, un manipulador cuya operación podía ser programada y que podía seguir una secuencia de movimientos determinados por las instrucciones en el programa.

Posteriores desarrollos del concepto de Devol y Engelberger condujeron al primer robot industrial o manipulador industrial construido por Unimation Inc. en 1958, la clave de este robot era el uso de un ordenador en conjunto con un manipulador para producir una máquina que podía ser enseñada, para realizar una serie de tareas en forma automática, al contrario que las máquinas automáticas de uso dedicado, estos robots se podían reprogramar y cambiar de herramienta a un costo relativamente bajo para efectuar otros trabajos cuando cambiaban los requisitos de fabricación.

### **2.4.4 REALIMENTACIÓN SENSORIAL**

En década de los años de 1960 se mejoró la flexibilidad de los brazos articulados mediante el uso de realimentación sensorial, H. A. Ernst publicó el desarrollo de una mano mecánica controlada por ordenador con sensores táctiles, dispositivo llamado MH-1, podía sentir bloques y usar esta información para controlar la mano de manera que apilaba bloques sin la ayuda de un operador, este robot es uno de los primeros capaz de tener una conducta adaptativa en un entorno no estructurado; el sistema manipulador consistía en un ANL modelo 8, con seis

grados de libertad, controlado por un ordenador TX-O mediante una interface; este programa de investigación evolucionó como parte del proyecto MAC, y se le añadió una cámara de televisión para comenzar la investigación sobre la percepción en la máquina.

Tomovic y Boni desarrollaron una mano prototipo provista con un sensor de presión que detectaba un objeto y proporcionaba una señal de realimentación de entrada a un motor para iniciar uno de los modelos de aprehensión, una vez que la mano estaba en contacto con el objeto, se enviaba a un ordenador información proporcional a su tamaño y peso mediante sensores de presión; en 1963, la American Machine y Foundry Company (AMF) introdujo el robot comercial Versatran.

A fines de la década de 1960, McCarthy y su equipo en el Stanford Artificial Intelligence Laboratory publicaron el desarrollo de un ordenador con manos, ojos y oídos (manipuladores, cámaras de televisión y micrófonos), mostraron un sistema que reconocía mensajes hablados, veía bloques distribuidos sobre una mesa y los manipulaba de acuerdo a instrucciones, en estos años, Pieper estudió el problema cinemático de un manipulador controlado por ordenador, mientras que Kahn y Roth (1971) analizaban la dinámica y el control.

#### **2.4.5 EL BRAZO STANFORD**

En 1968 la compañía japonesa Kawasaki Heavy Industries negoció una licencia con Unimation para sus robots y en 1969 se desarrolló un camión experimental por General Electric para la Armada Americana, el mismo año se desarrolló el brazo Boston y al año siguiente el brazo Stanford, que estaba equipado con una cámara y controlado por ordenador.

Trabajos más avanzados en robótica se efectuaron con estos brazos utilizándolos como robots manipuladores, un experimento con el brazo Stanford consistía en apilar automáticamente bloques de acuerdo con diversas estrategias, lo que era un trabajo sofisticado para un robot, en 1974, Cincinnati Milacron introdujo su primer robot industrial controlado por ordenador el T3 o The

Tomorrow Tool, que podía levantar más de 45 kg así como seguir a objetos móviles en una línea de montaje. La figura 2.7 muestra un brazo Stanford.



Fig. 2.7 Brazo Stanford<sup>18</sup>

El laboratorio de inteligencia artificial, Intelligence Laboratory del MIT, trabajó sobre los aspectos de inteligencia artificial de la realimentación de fuerzas. En el Draper Laboratory Nevins investigaron técnicas sensoriales basadas en el control coordinado de fuerza y posición, este trabajo desarrolló la instrumentación de un dispositivo de control remoto coordinado de fuerza y posición (remote center compliance RCC) que se unió a la placa de montaje de la última articulación del manipulador.

Bejczy, en el Jet Propulsion Laboratory, desarrolló una técnica de control de par, basada en ordenador; En 1970 se introdujo el uso de sensores para facilitar las operaciones manipulativas, en Stanford, Bolles y Paul, utilizando realimentación tanto visual como de fuerza, mostraron que un brazo Stanford controlado por ordenador conectado a un Digital PDP-10 efectuaba el montaje de bombas de agua de automóvil. Will y Grossman en IBM desarrollaron un manipulador controlado por ordenador con sensores de contacto y fuerza para realizar montajes mecánicos. Inoue, usó su brazo Stanford ampliado para proyectos de exploración espacial.

---

<sup>18</sup> <http://www.roboticafya.blogspot.com/>

#### 2.4.6 EL ROBOT PUMA

En la década de 1960 también se sustituyó el acoplo mecánico por sistemas eléctricos e hidráulicos en manipuladores tales como el Handyman de General Electric y el Minotaur I construido por General Mills, se puso este nombre en base al brazo manipulador de Víctor Scheinman, el robot manipulador más famoso, el robot PUMA. A fines de la década de 1960 y principios de los 70 se diseñan los primeros robots móviles con algún grado de autonomía.

El brazo robótico manipulador PUMA (Programmable Universal Machine for Assembly, o Programmable Universal Manipulation Arm) es un brazo robot industrial desarrollado por Víctor Scheinman en la empresa pionera en robótica Unimation, Inicialmente desarrollado para GM, este brazo robot nació de los diseños iniciales inventados por Scheinman mientras se encontraba en el MIT y en la Universidad de Stanford.

Unimation produjo PUMAs durante algunos años hasta que fue absorbida por Westinghouse en 1980, que luego pasó a la empresa suiza Staubli en 1988 que se dedica a soluciones en mecatrónica en las áreas de textil, conectores y robótica; Nokia Robotics, manufacturó brazos robots PUMA durante los años 1980, siendo el PUMA-650 el modelo más popular.

Nokia también ha desarrollado su propio robot - NS-16, robot que manejaba cargas de hasta 16 kg con una velocidad de 1 m / s; finalmente Nokia vendió su división de robótica en 1990, y en el año 2002, General Motors Controls Robotics and Welding (CRW) donó el prototipo original del brazo robot PUMA al Museo Nacional de Historia Americana, reconociéndose así su importancia en el desarrollo de la robótica. La figura 2.8 muestra un robot PUMA 500.





Fig. 2.8 Robot PUMA 500<sup>19</sup>

La figura 2.9 muestra un sistema quirúrgico desarrollado por Intuitive Surgical, se trata del robot Da Vinci, para cirugías complejas con mínimas invasiones para el cuerpo humano; es un manipulador de varios brazos utilizado en cirugías del Centro Hospitalario de Mónaco.



Fig. 2.9 Robot manipulador quirúrgico Da Vinci<sup>20</sup>

## 2.4.7 LA ROBÓTICA EN LA ACTUALIDAD

En la actualidad, octubre de 2014, muchas empresas en el mundo se dedican a la producción de robots, a continuación se mencionan algunos de los más interesantes desde la óptica de esta tesis del control por redes neuronales de brazos articuladores robóticos; por ejemplo, los robots universal UR5 con

<sup>19</sup> <http://roucab.blogspot.com/>

<sup>20</sup> <http://www.chpg.mc/fr/centre-hospitalier/>

capacidad de 5 Kg. y 850 mm de alcance y un peso de 60 Kg.; el UR10 con una capacidad de 10 Kg y 1300 mm de alcance, con un peso de 70 Kg.; son dos robots de la marca Global Robots Ltd. que tienen el sistema de programación más intuitiva que nunca se ha visto, diseñados para ser de fácil instalación, estos robots tienen una serie de características clave innovadoras que los diferencia de otros robots en el mercado: bajo precio 18 500 y 23 500 libras esterlinas, monofásico, ligero, invertible y montable en pared, fácil de programar puede ser enseñado a mover el brazo con la mano, y tiene un sistema controlador muy compacto.

El UR5 es perfecto para muchas aplicaciones de los robots estándar, tales como mantenimiento y limpieza de máquinas herramientas, recogida, colocación, montaje, paletizado, embalaje y desembalaje, encolado, sellado y muchas otras operaciones; además de las ventajas únicas hacen que estos robots sean ideales para el entrenamiento y la educación, publicidad, eventos de relaciones públicas, los efectos especiales y el trabajo en proyecciones de la películas, la automatización de gran flexibilidad y otras situaciones en las que un robot industrial estándar es demasiado grande, pesado, peligroso y complicado de configurar. Ver figura 2.10.



Fig. 2.10 UR5 y UR10 de Global Robots<sup>21</sup>

ABB es el líder mundial en robótica y han sido en gran parte de los últimos 40 años, más de 190000 robots ABB se han instalado en todo el mundo, con muchos robots desde la década de 1980 hoy sigue trabajando, tienen una envidiable reputación por su fiabilidad, a partir de S3 hasta el controlador más

---

<sup>21</sup> <http://www.robotsltd.co.uk/>

reciente IRC5, estos robots utilizan lenguaje de programación RAPID de ABB utilizado en controladores S4 y posteriores es reconocido como uno de los más potentes y adaptables en el mundo de los robots.

El ABB IRB 140 es un robot pequeño y excepcionalmente rápido, preciso y potente, a diferencia de muchos otros robots de este tamaño el IRB 140 está sólidamente construido y es una máquina industrial de todo el poder de 810 mm de altura es compatible en un espacio muy pequeño, incluso dentro de otra máquina, con una carga útil de 5 kg el 140 mm de alcance es perfecto para máquinas herramientas, para recoger el lugar de trabajo, así como para antorchas MIG y TIG de soldadura, amolado ligero, corte, desbarbado y pulido son algunas de las aplicaciones de estos brazos robotizados.

Una indicación de la verdadera naturaleza de la IRB 140 es el uso de un completo armario de control ABB IRC5 M2004. Esto significa que la instalación es sencilla y diseñada para un entorno industrial adecuado, no es algo que a menudo se puede decir de los tipos de brazos de robot más pequeños. Normalmente se suministra con una tarjeta de E/S de 16 entradas y 16 salidas puede ser equipado con muchos más si es necesario. E/S entradas y salidas analógicas también están disponibles. El controlador puede funcionar también como un PLC mediante el control y el seguimiento de las señales de E/S. El controlador se ha construido con una cadena de doble circuito de seguridad (de 0 V y + 24 V) lo que significa que el robot puede ejecutarse sin necesidad de dispositivos de desconexión de emergencia adicionales.

Debido a la excelente conectividad controladores que se puede utilizar fácilmente con programas suministrados comprar un CAD sistema de software que convierte CAM modelos CAD en senderos robot.

El brazo 140 IRB está disponible como un robot totalmente cerrado con protección IP67 o con Clean room 10 protección. Se ha instalado en muchas fábricas de todo el mundo desde 1999 y demostrado ser excepcionalmente fiable, incluso en circunstancias muy exigentes. La precisión de la posición indicada es  $\pm 0,03$  mm y el 140 IRB tiene muy buena trayectoria, se puede montar en una variedad de maneras, de hecho, se puede montar en cualquier ángulo, una gran ventaja para conseguir el mejor fuera de alcance. Existen muchos usos

diferentes para los robots, en una amplia gama de industrias. Ellos son, en principio, un simple pedazo de equipo y por lo tanto, altamente adaptable. Lo último de ABB un conjunto de brazos se muestra en la figura 2.11.

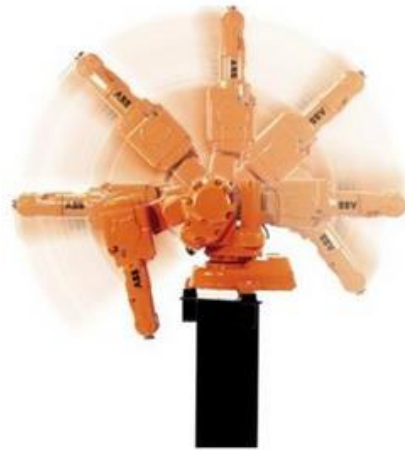


Figura 2.11 Robot 140 IRB de ABB<sup>22</sup>

El robot industrial de 6 ejes es el tipo más flexible y se puede utilizar de muchas maneras diferentes. Además hay robots más especializados, como los de ABB IRB 340, figura 2.12 son robots de recogida y colocación que se utilizan en aplicaciones muy específicas, algunas aplicaciones son muy simples y se han hecho miles de veces antes, pero se encuentran más y más industrias que están dando vuelta a la automatización y más específicamente robots, para incrementar la producción y calidad al tiempo que reduce costos y residuos.



Figura 2.12 Robot IRB 340 de ABB<sup>23</sup>

---

<sup>22</sup> <http://www.robotsltd.co.uk/>

<sup>23</sup> <http://www.robotsltd.co.uk/>

Los precios de los robots que se utilizan ahora son tan bajos y las presiones sobre las empresas que fabrican tan grandes que ya no son del dominio exclusivo de la industria automotriz, para cada aplicación se enumeran cientos de posibilidades de derivados y afines, con cientos de miles de robots instalados en todo el mundo en industrias tan diversas como aserraderos, manufactura de todo tipo, farmacéuticos, que proporcionan grandes ganancias en productividad, seguridad, economía y calidad.

## **2.5 SISTEMAS DE CONTROL DE ROBOTS**

Los sistemas de control desarrollados para los robots van desde los controles tradicionales proporcional derivativo PD, proporcional integrativo derivativo PID, controles del tipo adaptable y los controladores robustos, diversos tipos de control se han utilizado en el tiempo para los robots, sin embargo debe reiterarse el uso del término robot, como un sistema inteligente que aprende, desarrolla conocimiento y aplica conocimiento, obteniendo como resultado la autonomía para realizar determinadas tareas sin la asistencia externa, sino que responde a estímulos tal como los humanos, por tanto el control inteligente de esta nueva era de la robótica se basa sobre principios irracionales como la lógica difusa o las redes neuronales artificiales.

En la Revista Tecnológica de la ESPOL-RTE, Vol. 26, de Diciembre de 2013 aparece un artículo Implementación en C# de una Red Neuronal Artificial en el control locomotor de una hormiga robótica publicado por Jorge A. Párraga, Marcos C. Anzules, Cristhian E. Ramírez y Lincoln E. Santander, de la Carrera Informática, de la Escuela Superior Politécnica Agropecuaria de Manabí, Campus El Limón, Calceta, Ecuador.

Párraga y su equipo demuestran la habilidad de las redes neuronales artificiales en el control del movimiento de las hormigas, según sus autores, se creó una aplicación de software que simuló su comportamiento, se analizaron las características y movimientos posibles de la hormiga robótica, crearon los sensores, codificaron el espacio matricial, posición y generación de feromonas, elaboraron la interfaz gráfica con IDE Visual Studio 2010, realizaron pruebas,

bajo diferentes condiciones como ubicación del robot hormiga, feromonas, y estado de los sensores.

En la International Journal of Agriculture and Crop Sciences de Junio de 2013 se publicó lo que en el Departamento de Maquinaria Agrícola de la Facultad de Agricultura en la universidad Tabriz en Irán, Ali Reshadsedghi y Asghar Mahmoudi desarrollaron un trabajo de investigación en 2012 para detectar y clasificar las diferentes variedades de almendras en función del grosor de la cáscara y la dureza, utilizando un sistema acústico de impacto combinado con las redes neuronales artificiales (RNA).

Reshadsedghi y Mahmoudi desarrollaron los modelos de redes neuronales artificiales para lo que tomaron señales de sonido de 6000 almendras, 1.000 muestras para cada genotipo el ensayo consistía en dejar caer por gravedad pepas de almendra sobre una placa de impacto de acero y la señal acústica generada por el sistema es capturada por un micrófono y procesada por un PC, características como la amplitud, fase y espectro de potencia se extrajeron de diferentes almendras para el análisis de la señal de sonido en el tiempo y la frecuencia por medio de transformada rápida de Fourier, principal método de análisis de componentes que se utiliza para la obtención de características como el vector de entrada a los modelos de redes neuronales artificiales.

El modelo óptimo de las redes neuronales artificiales fue seleccionado una red neuronal de perceptrón multicapa, que exhibe una precisión más alta y mínimo error (MSE) en el set de datos de validación cruzada, para confirmar la potencialidad del sistema para la distinción y la clasificación de las variedades de almendras en tres clases, duro, semi-blando y suave.

## **2.6 REDES NEURONALES Y ROBOTS MANIPULADORES**

En el año 2000 en las cartas de procesadores neuronales aparece una publicación de la Kluber Academic Publisher en Holanda de Teixeira, Braga y Menezes respecto del control de un robot manipulador usando redes neuronales artificiales con adaptación en línea que tiene un mejor desempeño respecto de

un control usando redes neuronales fuera de línea, trabajo realizado en un robot PUMA 560.

Texeira y su equipo demuestran que la diferencia de usar control convencional mediante controladores PD y PID frente al uso de control por redes neuronales es abismal, pues el control fue implementado usando un perceptrón multicapa con un algoritmo de aprendizaje tipo back propagation para controlar el torque en las articulaciones utilizando el modelo de la dinámica inversa.

Los recientes avances en la automatización industrial han exigido una comprensión más profunda y la necesidad de desarrollo de métodos más robustos y fiables para el control de la máquina robótica industrial por excelencia el brazo manipulador, desde el punto de vista de sistemas de control, la robótica de manipuladores presenta sistemas altamente no lineales, multi-variable y con acoplamiento dinámico.

Un gran número de técnicas de control se encuentra en la literatura para controlar estas máquinas aunque es bastante común y ampliamente utilizado, las técnicas de control convencionales no dan resultados satisfactorios para controlar sistemas robóticos complejos.

Los métodos convencionales de control son basados en las ecuaciones dinámicas para lo que hay que calcular las fuerzas y pares de torsión en tiempo real. Los métodos basados en Sistema de Estructura Variable se refieren a modelos más simples pero, a pesar de ser robusta, pueden resultar en desviaciones en relación con trayectorias óptimas, como consecuencia de perturbaciones en la dinámica del sistema.

“La necesidad de más estrategias de control eficiente ha exigido el desarrollo de sistemas adaptativos de control, donde la capacidad de adaptación, la generalización, y la posibilidad de la aplicación del aprendizaje en tiempo real hacen de redes neuronales artificiales (RNA) una importante alternativa”<sup>24</sup>.

Se encontró en la literatura una tesis de maestría realizada por Enrique Martínez en 2008 en el Departamento de Mecatrónica del Centro Nacional de

---

<sup>24</sup> [http://www.cpdee.ufmg.br/~apbraga/MeusArtigos/Journals/12\\_RoselitoNPL.pdf](http://www.cpdee.ufmg.br/~apbraga/MeusArtigos/Journals/12_RoselitoNPL.pdf)

Investigación y Desarrollo Tecnológico CEDIMET en México que trata del control de un robot tipo PUMA de tres grados de libertad que realiza la implementación de una técnica de control alternativa basada en redes neuronales celulares para generar patrones de oscilación que llevan a cabo la función de un generador central de patrones GCP y una señal de control que permita que el robot siga la trayectoria definida por el GCP para provocar movimientos oscilatorios periódicos tratando de emular el movimiento de un brazo humano.

A continuación se detallan algunos tipos de controles utilizados en diferentes trabajos referentes al control de brazos robóticos articulados: Se usó en una tesis de grado en México en el CEDINET en 2005 para el “control de un robot de 3 gdl un control convencional PID más un control inteligente usando la metodología Núcleo Híbrido de Transición de Estado”<sup>25</sup>.

En 2007 otra tesis de control inteligente vía Núcleo Híbrido de Transición de Estado que partiendo de controladores Multi-input Single-Output, llegó a utilizar control multivariable multi-input Multi-Output, en las articulaciones de la cintura y el hombro con lo que implementaron controles desacoplados y acoplados en las articulaciones.

En otra tesis del 2008 usan estereoscopía y estimación de pose para el posicionamiento de un brazo robótico con dos técnicas de análisis de video mediante una cámara externa y otra acoplada al efector.

---

<sup>25</sup> Martínez E. (2008) Control de un robot utilizando un robot tipo PUMA



## **CAPÍTULO III**

### **APLICACIÓN DE LAS REDES NEURONALES AL RECONOCIMIENTO DE OBJETOS EN ROBOTS MANIPULADORES**

#### **3.1 REDES NEURONALES ARTIFICIALES**

Las redes neuronales artificiales son modelos matemáticos no algorítmicos que tratan de reproducir el comportamiento del cerebro humano debido a que constan de dispositivos elementales de proceso, denominados neuronas artificiales, una de las definiciones es: “una red neuronal artificial es un procesador distribuido, masivamente paralelo, que tiene una predisposición natural a almacenar conocimiento experimental y hacerlo disponible para su utilización”<sup>26</sup>.

Los sistemas neuronales artificiales se basan en las propiedades no lineales de la operación sinérgica de unidades de proceso elementales en paralelo, y tienen como característica que su función quede directamente ser trasladada a una arquitectura modular luego de un proceso de aprendizaje, de no programación, que es una alternativa a la computación basada en reglas de decisión prefijadas y algoritmos codificados que resultan ineficaces, en aplicaciones de clasificación, asociación y generalización.

Con las redes neuronales artificiales se busca la solución de problemas complejos, no como una secuencia de pasos, sino como la evolución de unos sistemas de computación inspirados en el funcionamiento del cerebro humano, y dotados de inteligencia, los cuales son la combinación de elementos simples de proceso interconectados, denominados neuronas artificiales, que operando de forma paralela, consiguen resolver problemas relacionados con el reconocimiento de formas o patrones, predicción, codificación, control y optimización entre otras aplicaciones.

---

<sup>26</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

Las redes neuronales artificiales son un campo de investigación amplio de la inteligencia artificial, contando con innumerables aplicaciones, entre las que se destacan: el aprendizaje automático, el modelado de patrones, el reconocimiento del habla y el procesamiento digital de imágenes.

### 3.1.1 EL MODELO DE MATEMÁTICO DE LA NEURONA ARTIFICIAL<sup>27</sup>

El modelo matemático de neurona artificial de McCulloch y Pitts consiste de una unidad con función de activación tipo escalón (binaria) como se observa en la Figura 3.1.

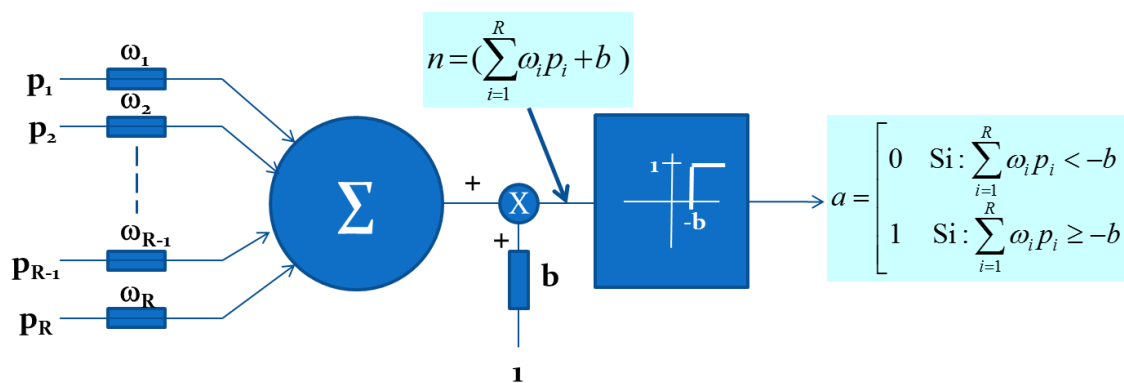


Figura 3.1 Modelo matemático de una neurona artificial de McCulloch y Pitts.<sup>28</sup>

Donde:

$P$  = Vector de entrada de dimensión  $R \times 1$

$W$  = Vector de los pesos sinápticos cuya dimensión es  $1 \times R$

$b$  = Umbral de activación de la neurona (bias)

$X$  = Función de transferencia o de activación

$a$  = Salida 1 o 0

El modelo matemático de una neurona artificial es análogo al modelo matemático que tendrían las neuronas reales, las conexiones físicas representarían las

<sup>27</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

<sup>28</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

dendritas, los pesos de ponderación a las sinapsis y el umbral de activación, de la actividad del soma; los pesos actúan como excitadores (positivos) o inhibidores (negativos) en función de su signo; la neurona artificial realiza una función de proceso, realizando el producto escalar de las entradas vector  $P$  por los pesos sinápticos vector  $W$ , los mismos que van actualizándose iterativamente hasta que la neurona o la red neuronal ha aprendido.

Si la sumatoria es mayor o igual al valor del umbral ( $-b$ ), la neurona se activa produciendo como salida el valor 1, y si es menor que el umbral, la neurona permanecerá inactiva y su salida será 0. La función de activación o de transferencia que originalmente propuso el lógico matemático Walter Pitts de tipo escalón unitario, que en el transcurrir del tiempo ha sido cambiada por otras funciones que producen mejores resultados, las más utilizadas son: Rampa, Gaussiana o sigmoidea, mono o bipolares, siendo la sigmoidea bipolar especialmente utilizada para el reconocimiento de patrones, como se observa en las figuras 3.2 y 3.3.

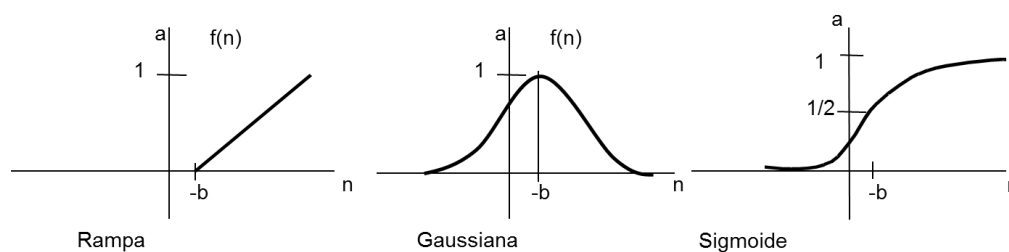


Figura 3.2 Funciones de activación<sup>29</sup>

<sup>29</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

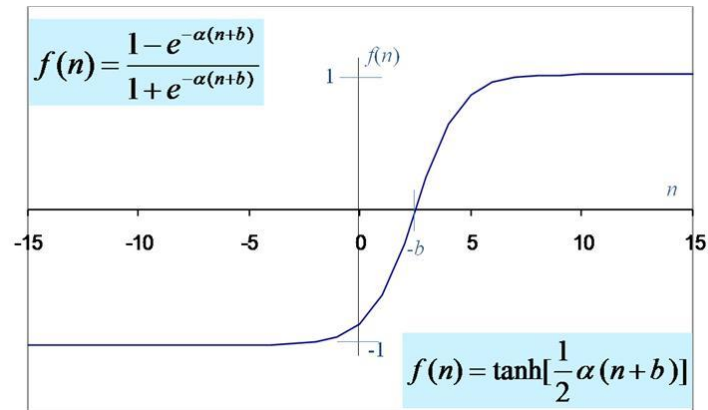


Figura 3.3 Función de Activación Sigmoidea Bipolar<sup>30</sup>

### 3.1.2 REDES NEURONALES CELULARES

Los modelos de redes neuronales artificiales utilizadas en el procesamiento digital de imágenes, es muy usado el modelo llamado red neuronal celular, se basa en los principios de la lógica celular y se centra en el procesamiento analógico y digital de imágenes, como procesamiento morfológico, filtrado espacial, filtrado en frecuencia y análisis en tiempo discreto de imágenes.

Las redes neuronales celulares fueron concebidas en la Universidad de California en Berkeley; por Leon O. Chua en 1987; los fundamentos en los cuales se basa Chua para generar su idea son: las redes neuronales artificiales, y los autómatas celulares; de las redes neuronales artificiales saca la capacidad de procesamiento asíncrono en paralelo, la dinámica en el tiempo y la interacción global de los elementos de la red; y de los autómatas celulares saca la idea de distribuir los elementos de procesamiento, también llamadas células, en rejillas regulares y permitir que la comunicación de cada célula con las otras se lleve a cabo a nivel local.

Las redes neuronales celulares se componen de un gran número de células o elementos de cómputo no lineales, organizados en capas periódicas bidimensionales rectangulares, mismas que pueden ser replicadas para tener redes neuronales celulares multicapa, que tienen conectividad local a diferencia de otros tipos de redes neuronales, cada célula puede recibir señales externas y de células próximas de su misma capa dentro de un entorno; son sistemas

<sup>30</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

dinámicos no lineales formados por la agrupación de procesadores elementales con una distribución espacial regular y donde las interacciones directas entre los distintos procesadores son de carácter local. A pesar de esto, las redes neuronales celulares pueden realizar tareas globales, por propagación de las señales por toda la red.

El comportamiento de la red neuronal celular está regido por un sistema de ecuaciones diferenciales no lineales, las que describen las contribuciones entre celdas interconectadas y las influencias mutuas entre sus estados y salidas, las contribuciones están afectadas por parámetros de realimentación, control, y umbral, los cuales van a particularizar el sistema para la realización de una tarea; en el caso de redes uniformes, estos parámetros serán independientes de la posición de la celda en la matriz de modo que se pueden definir plantillas matriciales de clonación que representan los pesos de realimentación y control respectivamente.

Para el análisis de imágenes las redes neuronales celulares utilizadas son arreglos bidimensionales, son tres modelos de redes neuronales celulares: de una sola capa, multicapa y en tiempo discreto; la red neuronal celular de una sola capa es la más utilizada y es el modelo original; el modelo de red neuronal celular de una sola capa, basa su éxito en su simplicidad ya que consta de un arreglo de células que están descritas por un conjunto simple de ecuaciones, debido a la alta cantidad de elementos que procesan los datos de manera paralela y continua.

El modelo multicapa es un conjunto de elementos arreglados en capas lo que hace posible que cada una trabaje persiguiendo un objetivo en particular y el resultado final será la sumatoria de los resultados parciales; la variedad de fenómenos que pueden ser analizados se verá incrementada pues la generalización se consigue mediante el empleo de varias variables de estado en cada célula en lugar de una sola.

Las redes neuronales celulares en tiempo discreto, fueron propuestas por Hubert Harrer, en 1992, como una clase de red neuronal celular que puede describirse por un algoritmo recursivo para actuar en períodos discretos de tiempo.

La morfogénesis es el estudio de procesos que tienen que ver con la formación de patrones que de la naturaleza, se han realizado, hay diversos trabajos relacionados, el más importante es el de Alan Turing en 1952, demostró como un modelo de reacción-difusión permite generar algunos patrones; de la misma manera que con las redes neuronales artificiales, los trabajos realizados con las redes neuronales celulares están relacionados con procesamiento de imágenes; simulación de modelos complejos principalmente en biología y encriptación de datos.

En 2012 Martínez J. y Ferrández J. proponen el diseño y la implementación sobre hardware reconfigurable de una arquitectura para la emulación en tiempo real de redes neuronales celulares; eligen la aproximación con mejores prestaciones para desarrollar el algoritmo computacional que describe la arquitectura hardware de la red neuronal celular, desde el punto de vista de implementación del hardware, la principal característica es la capacidad que tiene para procesar la información en flujo de datos y en tiempo real, demuestran la implementación de un sistema de cómputo distribuido, que permite el desarrollo de aplicaciones de procesamiento de video en tiempo real con redes neuronales celulares multicapa grandes y complejas, el sistema procesa la información de video en flujo de datos y proporciona una salida de video estándar compatible con el formato VGA industria.

### **3.1.3 ALGORITMOS EVOLUTIVOS**

Los algoritmos evolutivos son métodos de solución y optimización que se basan en la evolución biológica, en ellos se mantiene un conjunto de entidades que representan posibles soluciones, las cuales se mezclan, y compiten entre sí, de manera que las que tienen capacidad de adaptación prevalecen en el tiempo, evolucionando hacia mejores soluciones, son una rama de la inteligencia artificial, utilizados básicamente en problemas con espacios de búsqueda muy grandes y no lineales, en los que otros métodos consumen excesivo recurso computacional.

Las entidades soluciones al problema se denominan individuos o cromosomas, y el conjunto, población, los individuos son modificados por operadores genéticos, como el sobrecruzamiento que consiste en la mezcla de la

información varios individuos; la mutación es un cambio aleatorio en los individuos; y la selección de los individuos que sobrevivirán y conformarán la siguiente generación, los individuos que representan las mejores soluciones al problema tienen mayor posibilidad de sobrevivir y la población irá mejorando gradualmente.

### **3.2 PROCESOS DE APRENDIZAJE NEURONAL<sup>31</sup>**

Aprendizaje Hebbiano del psicólogo Donald Hebb: “si las neuronas de ambos lados de una sinapsis se activan simultáneamente repetidas veces, el valor de la sinapsis se incrementa”<sup>32</sup>; el aprendizaje neuronal es la adaptación de los pesos de sus conexiones sinápticas, tendiente a disminuir el error entre la salida output y la salida deseada target, por una entrada input.

#### **3.2.1 CONOCIMIENTO Y ENTORNO.**

El proceso de aprendizaje se desarrolla en secuencia, la red neuronal se estimula por muestras de su entorno; cambia por el estímulo aplicado y responde al entorno, mostrando el entrenamiento, para que la red neuronal aprenda necesita saber qué información se tiene que hacer explícita y, cómo se debe codificar la información para su uso; el conocimiento del entorno, comprende dos tipos de información, el conocimiento previo y el estado actual; y, Las mediciones de los sensores. Las mediciones son ruidosas y propensas a errores debidos a imperfecciones de los sensores y de los sistemas de medición, las observaciones son símbolos que pueden ser suficientemente útiles como para conformar una fuente de información, de manera que se puede seleccionar conjuntos de patrones para entrenar (Training Set), comprobar el aprendizaje (Testing Set) y verificar la operación de una red neuronal artificial.

#### **3.2.2 ENTORNO DE INTERÉS Y PATRONES**

Los objetos que constituyen el entorno de interés pueden ser datos, señales visuales imágenes, audibles, eléctricas, estados de un sistema o cualquier objeto susceptible de ser representado, por tanto aprendido, reconocido o clasificado y se representan por patrones que resulta ser una abstracción de los atributos de

---

<sup>31</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

<sup>32</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

un objeto de interés, sintetiza sus rasgos característicos, se lo representa por un vector n-dimensional que integra un conjunto de mediciones obtenidas de los atributos.

“La representación del conocimiento mediante patrones y la selección del correspondiente conjunto de vectores constituyen las principales tareas del proceso de diseño de soluciones con redes neuronales, y son la clave de su operación”<sup>33</sup>.

### 3.2.3 APRENDIZAJE NEURONAL

La red neuronal debe aprender por entrenamiento, a través de un conjunto de vectores de entrada cuyas respuestas o salidas sean conocidas, y es ahí precisamente donde entra el criterio del experto, para determinar la salida correcta, pueden existir ejemplos cuya señal de entrada, no requieran de profesor para determinar la salida deseada.

Dado un conjunto de ejemplos, la capacidad de aprender es una función de tres factores: la tolerancia al error (e), el número de rasgos presentes en los ejemplos (n) y el tamaño de la regla necesaria para discriminar (t). H. Banda escribe: “Un sistema es capaz de aprender un concepto si, dado un conjunto de ejemplos positivos y negativos, puede producir un algoritmo que pueda clasificar correctamente en el futuro, con probabilidad inversamente proporcional a la tolerancia especificada para el error”<sup>34</sup>, por tanto “si el número de ejemplos requeridos para el entrenamiento de un sistema está dado por un polinomio  $p(e,n,t)$ , entonces se dice que el concepto es susceptible de ser aprendido”.<sup>35</sup>

### 3.2.4 TAMAÑO DE LA MUESTRA

Según Desmond H. Foley<sup>36</sup>, para minimizar el error del sistema, el número de ejemplos, elegidos para conformar el conjunto de entrenamiento debe ser al

---

<sup>33</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

<sup>34</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

<sup>35</sup> <http://plato.stanford.edu/entries/learning-formal/>

<sup>36</sup> Foley, D. H. *Considerations of Sample and Feature Size*. IEEE Transactions on Information Theory, Vol IT-18, 1972, pp. 618.



menos tres veces el número de rasgos utilizados para representar cada caso, lo que quiere decir que es igual o mayor a tres veces la dimensión del vector del patrón característico.

### **3.2.5 PREPROCESAMIENTO**

Para un entrenamiento más eficiente se aplican ciertos pasos de pre procesamiento a los vectores de entrada y salida esperadas como el análisis del poder discriminante de los atributos; escalamiento de entradas y salidas (rango entre -1 y +1); normalización de entradas y salidas (media cero y desviación estándar unitaria) y análisis de componentes principales (reducción dimensional); para lo que se asume normalidad de los datos.

### **3.2.6 FORMAS DE ENTRENAMIENTO**

Entrenamiento incremental, si la presentación de ejemplos a la red y el ajuste de los pesos de las conexiones sinápticas y los valores de los umbrales de decisión, son actualizados cada vez que un vector de entrada es presentado a la red; o entrenamiento por lotes si los pesos de las conexiones sinápticas y los valores de los umbrales de decisión, son actualizados después de que todos los de vectores de entrada han sido presentados a la red (época).

### **3.2.7 SOBRE ENTRENAMIENTO**

El sobre entrenamiento es uno de los inconvenientes durante el entrenamiento de redes neuronales; el error toma un valor muy pequeño, sin embargo si nuevos vectores son presentados a la red y el error es grande, lo que ocurre es que la red ha memorizado los ejemplos de entrenamiento, pero no tiene capacidad de generalización a situaciones nuevas.

### **3.2.8 MATRIZ DE CONFUSIÓN**

La matriz de confusión se utiliza para describir los errores de una red neuronal cuando clasifica un conjunto de vectores dados de un problema particular para

el cual fue entrenada; en la matriz de confusión que se presenta en la figura 3.4, los grupos de clasificación son las filas y las columnas, donde los valores de la diagonal representan los casos correctamente clasificados, mientras que los valores fuera de la diagonal representan los errores cometidos.

		Grupos o Clases				
		1	2	3	4	5
Grupos o Clases	1	$c_{11}$	$e_{12}$	$e_{13}$	$e_{14}$	$e_{15}$
	2	$e_{21}$	$c_{22}$	$e_{23}$	$e_{24}$	$e_{25}$
	3	$e_{31}$	$e_{32}$	$c_{33}$	$e_{34}$	$e_{35}$
	4	$e_{41}$	$e_{42}$	$e_{43}$	$c_{44}$	$e_{45}$
	5	$e_{51}$	$e_{52}$	$e_{53}$	$e_{54}$	$c_{55}$

Figura 3.4 Matriz de Confusión<sup>37</sup>

### 3.2.9 ERROR DE APRENDIZAJE

Se puede construir la función error y determinar una zona de error optimista si se utiliza el conjunto de entrenamiento como conjunto de prueba y error pesimista si se utiliza un conjunto de prueba diferente al conjunto de entrenamiento, por tanto en algún punto entre los dos errores estará el error real de la red neuronal; la figura 3.5 muestra la estimación del error de desempeño de las redes neuronales según Hugo Banda.

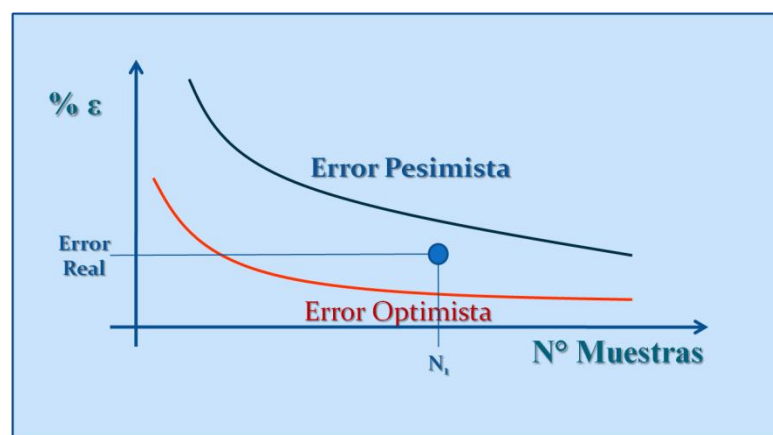


Figura 3.5 Estimación de Error de Desempeño en Redes Neuronales<sup>38</sup>

<sup>37</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

<sup>38</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

### 3.2.10 EL APRENDIZAJE

Una red neuronal artificial tiene la habilidad para aprender y mejorar su eficiencia en el transcurso del tiempo, el aprendizaje puede ser de dos tipos: algoritmos de aprendizaje y paradigmas de aprendizaje, en este caso, al igual que en los seres humanos, una red neuronal artificial aprende con la ayuda de un profesor o sin él; esto es aprendizaje supervisado y el aprendizaje que se realiza sin profesor, se le clasifica en aprendizaje por refuerzo y aprendizaje no supervisado.

### 3.2.11 APRENDIZAJE SUPERVISADO

En el aprendizaje supervisado, se requiere de un profesor con conocimiento del entorno, el profesor determina el conjunto de vectores que representan ese conocimiento, la red ve el entorno por medio de la codificación hecha por el profesor, que es capaz de proveer a la red la respuesta deseada para cada vector de entrada usado para el entrenamiento y los parámetros de la red son ajustados por el error, que es la diferencia entre la respuesta deseada (target) y la respuesta actual de la red. La figura 3.6 muestra el proceso de aprendizaje de una red neuronal con profesor.

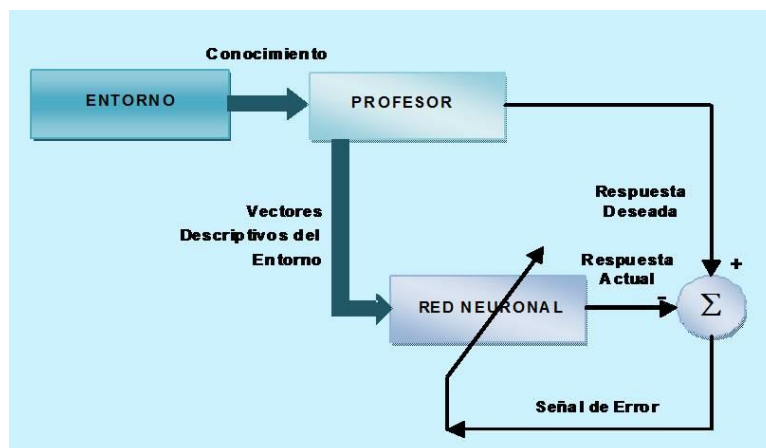


Figura 3.6 Modelo de Aprendizaje con Profesor<sup>39</sup>

Los ajustes de los pesos de la red se realizan en forma iterativa, de modo que el comportamiento de la red sea similar al del profesor, la emulación se califica utilizando criterios estadísticos, por ejemplo mínimos cuadrados, entonces el

<sup>39</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

conocimiento del profesor es transferido a la red, como un código sub-simbólico y almacenado en forma de pesos sinápticos, lo que resulta en una memoria de largo plazo, con lo que la red está lista para actuar por sí misma.

### 3.2.12 APRENDIZAJE POR REFUERZO

El aprendizaje por refuerzo se realiza por interacción continua con el entorno y se centra alrededor de un crítico, encargado de convertir una señal de refuerzo primario recibida, en una señal de refuerzo heurístico, procedimiento que permite al sistema de aprendizaje interactuar con el entorno y desarrollar la habilidad de realizar una tarea predeterminada. La figura 3.7 muestra el modelo de aprendizaje por refuerzo.

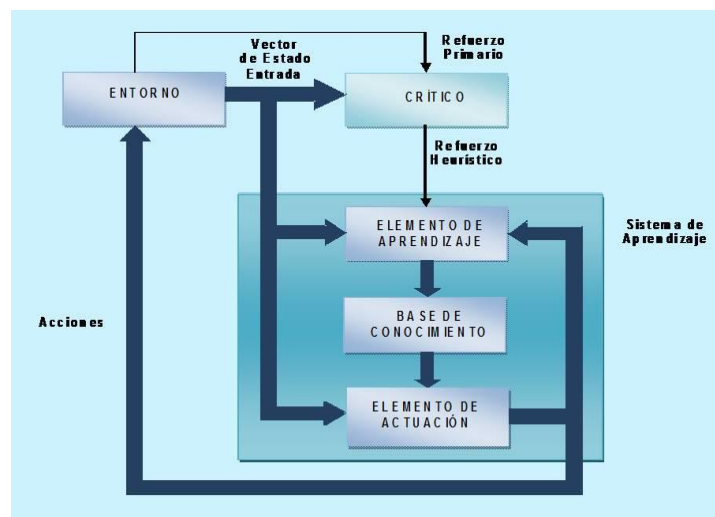


Figura 3.7 Modelo de Aprendizaje por Refuerzo<sup>40</sup>

### 3.2.13 APRENDIZAJE NO - SUPERVISADO

En el aprendizaje no supervisado, no hay ni un profesor ni un crítico para supervisar el proceso, no existen ejemplos de la función a ser aprendida por la red; en este caso se provee una medida, que determina la calidad de la representación que debe ser aprendida por la red, y sus pesos sinápticos son optimizados con respecto a esa medida, en base a la estadística de las entradas, la red aprende a formar sus representaciones para los rasgos característicos de las entradas y genera nuevas clases automáticamente; comúnmente se usa el aprendizaje competitivo, el sistema de aprendizaje está conformado por una

<sup>40</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

capa de entrada que recibe los datos y una capa en la que las neuronas compiten unas con otras por responder a los datos de entrada, y la neurona con mayor número de entradas se activa, mientras que las otras se desactivan. La figura 3.8 muestra el modelo de aprendizaje no-supervisado.

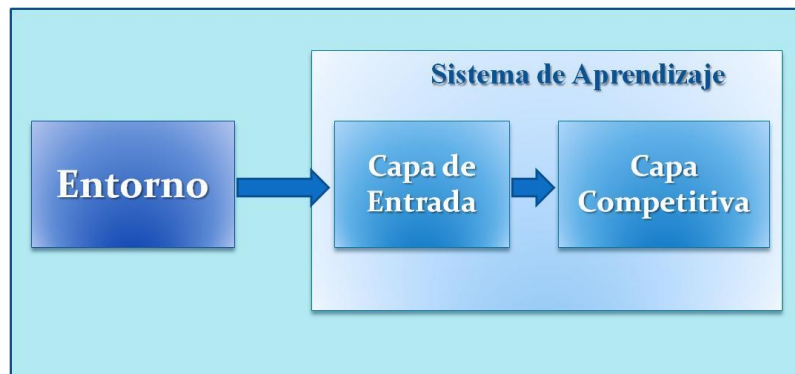


Figura 3.8 Modelo de Aprendizaje No-Supervisado<sup>41</sup>

Las tareas de aprendizaje definen la arquitectura de red y el algoritmo de aprendizaje, las más importantes han sido consideradas en el nntool de Matlab, sin embargo puede generarse una arquitectura en un ambiente de programación particular pues la universalidad de las redes neuronales permite.

### 3.1.14 ASOCIACIÓN DE PATRONES

La memoria humana puede aprender por aprendizaje asociativo y puede ser de dos tipos: autoasociación y heteroasociación; si se requiere que la red neuronal artificial aprenda un conjunto de patrones (vectores) por la presentación repetitiva y luego, al presentarle una versión incompleta o ruidosa, la red es capaz de recuperar y entregar en su salida el patrón completo, se dice que el aprendizaje es por autoasociación en aprendizaje no supervisado; sin embargo “para que una operación sea confiable el número de patrones aleatorios debe ser menor al 14% del total de neuronas de la memoria auto asociativa”<sup>42</sup>. En cambio por heteroasociación, la red asocia un conjunto arbitrario de patrones de entrada, con otro conjunto arbitrario de patrones de salida, para el aprendizaje supervisado.

<sup>41</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

<sup>42</sup> Müller B and Reinhardt J. *Neural Networks - An Introduction*. Physics of Neural Networks Series. Springer-Verlag, Germany, 1991

### **3.2.15 RECONOCIMIENTO DE PATRONES**

“El reconocimiento de patrones es un proceso en el cual un patrón recibido es asignado a uno de un número prescrito de clases, es un proceso de tipo estadístico, en el que los patrones son representados por puntos en un espacio de decisión n-dimensional, cuyos ejes son los n atributos codificados en los vectores de entrada, el espacio de decisión está dividido en regiones, cada región está asociada a una clase, las fronteras de decisión se determinan durante el proceso de entrenamiento”<sup>43</sup>.

### **3.2.16 CONTROL DE PLANTAS CON REDES NEURONALES ARTIFICIALES**

Una red neuronal artificial es capaz de realizar el control de una planta, donde planta es un proceso o un sistema, que debe ser mantenido en control, para lo que es necesario conocer sus parámetros, que son variables y dependen del punto de operación, por lo que la red neuronal artificial, constituye un sistema idóneo para su determinación dinámica, sea por aprendizaje directo, o indirecto, según Hugo Banda.

## **3.3 EL PERCEPTRÓN<sup>44</sup>**

Rosenblatt en 1958 desarrolló una aplicación práctica de la neurona de McCulloch y Pitts, utilizando el modelo de aprendizaje de Hebb, hizo una máquina capaz de reconocer objetos de forma visual, considerándose la primera red neuronal artificial de la historia.

### **3.3.1 IMPORTANCIA HISTÓRICA**

El siguiente desarrollo importante en las redes neuronales luego del concepto de modelo de neurona artificial de McCulloch y Pitts fue el concepto de perceptrón de Frank Rosenblatt en 1958, el perceptrón es una neurona de McCulloch y Pitts donde las entradas se pasan primero a través de algunos pre-procesadores, (unidades de asociación), que detectan la presencia de características específicas en las entradas; el perceptrón es un sistema de reconocimiento de patrones.

---

<sup>43</sup> BANDA, H. (2014), Inteligencia Artificial Principios y Aplicaciones. EPN.

<sup>44</sup> BANDA, H. (2014), Inteligencia Artificial Principios y Aplicaciones. EPN.

David Hubel y Torsten Wiesel en 1962 estudian la corteza visual del gato, descubrieron que había grupos de células en la corteza que respondían a la presencia de ciertas características de los cuerpos en el campo de la visión del gato, un grupo de células respondía a la presencia del borde en un cuerpo, otro grupo de células respondía a la presencia de una línea en una determinada dirección; concluyeron que estos grupos de células son las unidades de asociación del perceptrón de Rosenblatt.

Una unidad de asociación es una neurona de McCulloch y Pitts cuya salida es 1 si recibe un patrón específico de entradas, y 0 para los demás patrones de entradas, cada unidad de asociación tendrá un cierto número de entradas que se seleccionan de todas las entradas al perceptrón, por tanto el número de entradas a una unidad de asociación particular es menor o igual al número total de entradas; la salida de cada unidad de asociación se convierte en entrada a una neurona McCulloch y Pitts, y la salida de ésta, es la salida del perceptrón; en conclusión, un perceptrón es una capa de las neuronas de McCulloch y Pitts, y todas estas neuronas envían sus salidas a una sola neurona McCulloch y Pitts.

### 3.3.2 PERCEPTRÓN SIMPLE

La figura 3.9 muestra un perceptrón simple de  $R$  entradas a una neurona cuya función de transferencia produce una salida ( $a$ )  $a = 1$  o  $a = 0$ . La entrada  $p_0 = 1$ , multiplicada por el peso  $b$ , facilita el entrenamiento, sesgando horizontalmente la función de transferencia; la transición de 0 a 1 sucede en el punto del eje horizontal en el que el resultado del producto escalar del vector de pesos  $W$  por el vector de patrones  $p$  es igual a  $-b$  (umbral de decisión bias).

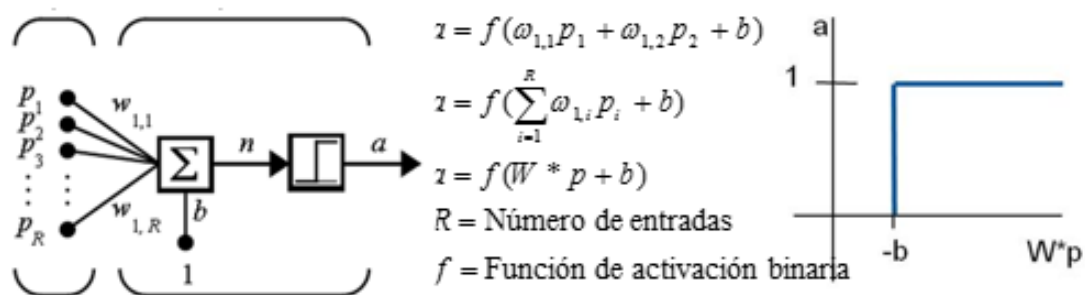


Figura 3.9 Modelo de un perceptrón simple según Matlab<sup>45</sup>

<sup>45</sup> <http://www.mathworks.com/>

El perceptrón simple tiene la habilidad de clasificar los vectores de entrada, dividiendo el espacio de atributos (feature space) en dos regiones delimitadas por la línea dada por:  $W * p + b = 0$  ; ver figura 3.10

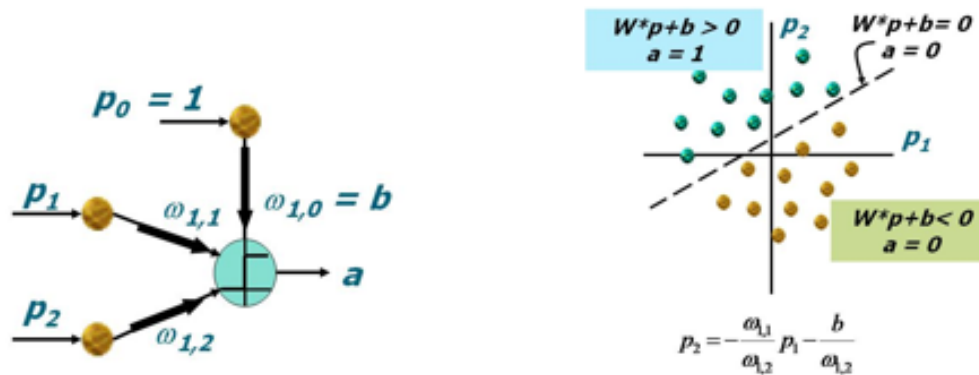


Figura 3.10 El perceptrón como clasificador lineal<sup>46</sup>

En Neural Network Toolbox Demos de Matlab, se encuentra como ejemplo la regla de aprendizaje del perceptrón, Perceptron Learning Rule, con el cual se puede interactuar posicionando puntos blancos y negros que se los pueda separar con una línea; al seleccionar la opción del umbral de decisión (biases) y seleccionando valores aleatorios para pesos y umbral de decisión, se comprueba que al utilizar el entrenamiento incremental, Learn, la red encuentra una solución para separar los grupos con la línea negra entrecortada de la figura 3.9, mientras que si se usa la opción de entrenamiento por lotes, Train, la red no encuentra solución y por medio de la interfaz gráfica, el demo entrega los valores de las variables en el entrenamiento.

### 3.3.3 ARQUITECTURA DE PERCEPTRONES

Una red de perceptrones se configura como una capa de  $S$  neuronas conectadas a  $R$  entradas a través del conjunto de pesos  $W_{i,j}$ . Los subíndices indican que el peso  $W$  conecta la entrada  $j$  a la neurona  $i$ . La regla de aprendizaje del perceptrón, sólo puede entrenar una sola capa de neuronas.

<sup>46</sup> <http://www.mathworks.com/>



Las redes de perceptrones tienen varias limitaciones, por ejemplo las salidas de las neuronas sólo pueden tener valores de 0 o 1, y los perceptrones sólo pueden clasificar conjuntos de vectores linealmente separables, la figura 3.11 muestra la arquitectura del perceptrón en notación Matlab.

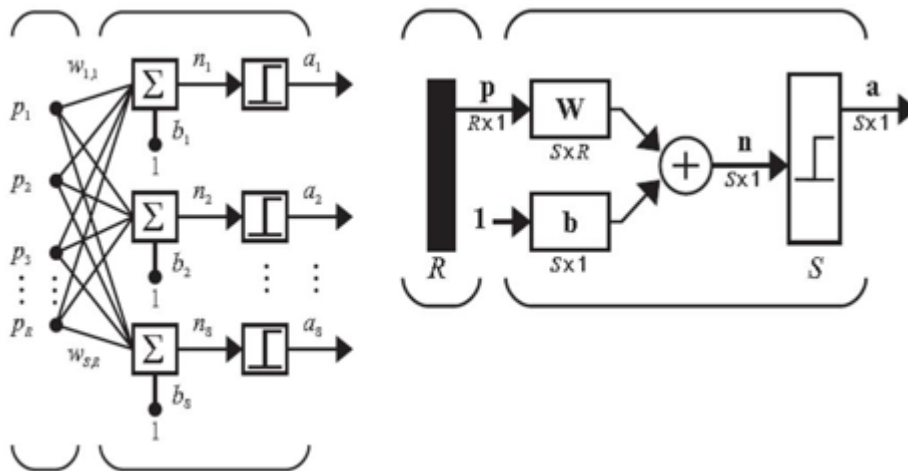


Figura 3.11 Arquitectura de Perceptrones en notación Matlab<sup>47</sup>

Donde:

$R$  = Número de entradas del vector de patrones  $p$

$S$  = Número de neuronas en la capa 1

$W$  = Matriz de pesos  $S$  filas x  $R$  columnas

$b$  = Vector de  $S$  valores de umbral.

### 3.3.3.1 Reconocimiento de Patrones del Perceptrón<sup>48</sup>

En 1.5.3 se vio que una sola neurona McCulloch y Pitts no puede producir la función or exclusiva, pero es posible construir una red de neuronas perceptrón que la produzca.

<sup>47</sup> <http://www.mathworks.com/>

<sup>48</sup> <http://www.mind.ilstu.edu/curriculum/modOverview.php?modGUI=212>

Primero, se crea una neurona MCP que produce un 1 si la primera entrada es 1 y la segunda entrada es 0, y produce un 0 para todos los demás pares de entradas, esta neurona es una unidad de asociación que detecta el patrón (1,0); luego se hace otra neurona que produce un 1 si la primera entrada es 0 y la segunda entrada es 1, y produce 0 para los demás pares de entradas, esta neurona reconoce el patrón (0,1).

Después se crea una tercera neurona que produce un 0 si ambas entradas son 0, y produce un 1 para las otras entradas, si se usa las salidas de las dos primeras neuronas como las entradas a la tercera neurona, se tiene un perceptrón que produce un 1 si el par de entradas es (1,0) o (0,1), y producirá un 0 para los demás pares de entradas, esto será el perceptrón or exclusivo, el perceptrón reconoce los dos patrones (1,0) y (0,1).

Para el reconocimiento de patrones se hizo matrices de unos y ceros, como por ejemplo para reconocer el patrón de unos y ceros de la matriz de la tabla 3.1, se observa que los unos forman una letra T.

Tabla 3.1 Matriz de ceros y unos para formar el patrón T

1	1	1
0	1	0
0	1	0

Es posible diseñar un perceptrón que puede reconocer este patrón, es decir, el perceptrón producirá un 1 para este patrón, y se producirá un 0 para cualquier otro patrón de unos y ceros, tal perceptrón tendría nueve entradas, una para cada uno de los cuadrados de la matriz, se podría diseñar el perceptrón con una unidad de asociación de nueve entradas para cada posible patrón de unos y ceros, y la neurona MCP final tendría de salida 1 sólo cuando la unidad T patrón de asociación envíe su señal, lo que sería muy ineficiente, porque hay  $2^9$  posibles patrones de unos y ceros en esta matriz, así que este perceptrón tendría 512 unidades de asociación, y cada unidad asociación tendrían nueve entradas, que sería un gran perceptrón para una tarea muy elemental.

La cuestión es ¿se puede construir un perceptrón más pequeño y eficiente que detecte el patrón de T? si, es posible diseñar un detector de T mucho más eficiente al reducir el número de unidades de asociación y el número de entradas a cada unidad de la asociación, sólo hay tres unidades de asociación que tienen tres entradas cada una, la primera unidad de asociación podría detectar el patrón de (1,1,1) a partir de las tres entradas en la primera fila, la segunda unidad de asociación podría detectar el patrón de (0,1,0) a partir de las tres entradas en la segunda fila, y el tercera unidad de asociación también podría detectar el patrón (0,1,0) a partir de las tres entradas en la tercera fila, así se tendría un perceptrón con sólo tres unidades de la asociación, y cada unidad asociación tendría tres entradas, entonces este es un perceptrón mucho más eficiente que el primero.

### **3.3.4 ORDEN DEL PERCEPTRÓN**

Una manera de medir la eficiencia de un perceptrón es considerar el mayor número de entradas de todas las unidades de asociación, esto se denomina el orden del perceptrón, la primera forma de concebir el perceptrón detector de la matriz de unos en forma de T descrito en 3.3.3.1, muy ineficiente porque cada unidad de asociación tiene nueve entradas, pero el segundo detector tiene tres, porque cada una de sus unidades de asociación tiene sólo tres entradas.

### **3.3.5 EFICIENCIA DE LOS PERCEPTRONES**

Cualquier imagen en blanco y negro digital puede ser considerada como una matriz de unos y ceros, y el ejemplo del patrón T de la tabla 3.1, demuestra que los perceptrones pueden ser diseñados para reconocer cualquier patrón en una serie de unos y ceros, la eficiencia del perceptrón se vuelve más importante cuando el tamaño de la matriz se hace más grande, como en el ejemplo del modelo T, que era posible diseñar un perceptrón bastante eficiente que reconoce el patrón, pero este tipo de soluciones eficientes no siempre son posibles, esta fue la conclusión notable y decepcionante de un perceptrón escrito en el libro Perceptrons de Marvin Minsky y Seymour Papert lo que realmente detuvo la investigación de esta ciencia por un lapso de tiempo casi de 13 años entre 1969 y 1982.

Minsky y Papert demostraron que hay un problema de reconocimiento de patrones simples que no puede ser resuelto de manera eficiente con un

perceptrón, ya que el problema puede ser resuelto con un perceptrón, pero los perceptrones seleccionados pueden no ser eficientes.

### 3.3.6 EL PROBLEMA DE PARIDAD

El problema se puede explicar así, contar en una matriz dada de unos y ceros, un perceptrón que tiene una matriz de tamaño dado y tiene la salida 1 si el número de unos en la matriz es impar, o la salida 0 si el número de unos es par, para explicar el problema de la paridad se va a considerar matrices con tres filas y tres columnas, a continuación se muestran tres ejemplos en la tabla 3.2

Tabla 3.2 Ejemplos del problema de paridad

1	1	1
0	1	0
0	1	0

1	1	0
0	1	0
0	1	0

1	0	0
0	0	0
0	1	1

La matriz de la izquierda tiene 5 unos, que es un número impar, la salida será 1, el patrón mostrado en la matriz del centro tiene 4 unos por tanto su salida es 0, y la matriz de la derecha tiene 3 unos por tanto su salida sería 1.

Es posible diseñar un perceptrón de orden nueve que resuelva este problema, pero esto es ineficiente ya que el número total de entradas es 9, Minsky y Papert demostraron que cualquier perceptrón que pueda resolver este problema debe tener orden 9 y que además puede resolver el problema de paridad para una matriz con  $n$  filas y  $n$  columnas debe tener orden  $2^n$ , que es el número total de entradas de la matriz, lo que significaría que no hay manera eficiente de resolver el problema de paridad con un perceptrón.

Minsky y Papert dieron otros ejemplos de problemas de reconocimiento de patrones que demostraron la limitación de los perceptrones con pruebas matemáticas, pero sólo consideraron perceptrones con unidades de asociación y una neurona de McCulloch y Pitts de salida; una manera de evitar las limitaciones de los perceptrones de una sola capa es vincular a redes de perceptrones en varias capas, pero Minsky y Papert sugirieron que estos perceptrones multicapa tenían sus propias dificultades, especialmente la falta de

un buen algoritmo de aprendizaje; por estas razones, el interés por los perceptrones, y redes neuronales en general, disminuyó en gran medida por bastante tiempo, sin embargo, las dificultades del enfoque de múltiples capas fueron finalmente superados.

### **3.3.7 PERCEPTRONES MULTICAPA**<sup>49</sup>

Un perceptrón multicapa consta de un número  $n$  de entradas  $x_i$ , una o varias capas ocultas de neuronas, y una capa de neuronas de salida, las entradas se envían a la primera capa, y la salida de cada una de estas se convierte en la entrada para la siguiente capa; finalmente, la capa de salida produce las salidas,  $y_i$ , del perceptrón; las entradas y salidas pueden ser números decimales, no sólo unos y ceros; además, las neuronas utilizadas son similares a las neuronas de McCulloch y Pitts, pero en lugar de enviar un 0 o un 1, estas calculan una suma ponderada de sus entradas y envían un número decimal entre 0 y 1, que es una función de la suma ponderada.

### **3.3.8 REGLA DE RETROPROPAGACIÓN**<sup>50</sup>

Esta es la regla de aprendizaje del perceptrón, lo que fue una de las grandes ventajas del perceptrón de capa única fue la facilidad con la que se podría aplicar la regla de aprendizaje para ajustar los pesos hasta obtener una salida deseada, parece no haber una manera sistemática para ajustar los pesos entre las capas ocultas en un perceptrón multicapa, sin embargo, este obstáculo fue superado por Rumelhart, Hinton y Williams en 1986 con el desarrollo de la regla de retropropagación; reglas similares se habían obtenido por Werbos en 1974, Parker en 1985, y LeCun en 1985, el trabajo de procesamiento paralelo distribuido, editado por Rumelhart y McClelland fue lo que hizo de la retropropagación la regla de aprendizaje más popular para perceptrones multicapa.

La regla de retropropagación comprende dos pasos a través de las capas del perceptrón, un paso hacia adelante y uno atrás, un número dado de entradas se alimentan a la red y la salida obtenida, se resta de la salida deseada para obtener el error, el mismo que se alimenta hacia atrás a través del sistema de la capa de

---

<sup>49</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

<sup>50</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

salida a la última capa oculta y así sucesivamente a las entradas, de ahí el nombre de propagación hacia atrás; en cada paso los pesos se ajustan a fin de hacer la salida obtenida más cercana de la salida deseada; con la presentación de muchas entradas con las salidas deseadas y luego aplicar retropropagación del error, el perceptrón, se dice que ha aprendido y está capacitado para producir los resultados deseados con un mayor grado de exactitud.

El perceptrón se entrena con ejemplos de comportamiento deseado, mediante la presentación de una secuencia de  $Q$  pares entrada-salida  $(p, t)$  de modo que el error  $(e)$ , va disminuyendo secuencialmente dado por la diferencia entre el vector de salida deseado  $(t)$  y las salidas reales de las neuronas  $(a)$ :  $e = t - a$ .

La red neuronal artificial perceptrón aprende de la siguiente manera: Inicializar todos los pesos  $w_{ij}$  y umbrales de decisión  $b_i$  con valores aleatorios normalmente fraccionarios menores que 1, para  $q = 1$  hasta  $q = Q$ , presentar un par entrada, salida  $(p^q, t^q)$  y calcular la respuesta de las neuronas  $a^q$  al patrón  $p^q$  como indica la ecuación 3.1

$$a^q = W * p^q + b \quad (3.1)^{51}$$

Calcular las matrices y realizar los ajustes de los pesos  $W_{ij}$  y de los valores del umbral de decisión  $b$  (bías):

$$\Delta W = e \cdot (p^q)^T \quad (3.2)^{52}$$

$$b = e = t^q - a^q \quad (3.3)^{53}$$

Si el error es mayor que un valor dado como aceptable, iterar hasta alcanzar un valor aceptable, de acuerdo con el Dr. Hugo Banda: “El perceptrón aprende mediante entrenamiento incremental, es decir las correcciones a los pesos y valores de umbral se realizan cada vez que se presenta un patrón a la entrada, la regla de aprendizaje del perceptrón ha demostrado que converge a una

---

<sup>51</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

<sup>52</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

<sup>53</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

solución en un número finito de iteraciones, si los vectores de entrada son linealmente separables”<sup>54</sup>.

La característica principal del modelo neuronal de McCulloch y Pitts es que la suma ponderada de las señales de entrada es comparada con un umbral para determinar la salida de la neurona, cuando la suma es mayor o igual al umbral, la salida es igual a 1 y cuando la suma es menor que el umbral, la salida es 0.

### **3.4 ARQUITECTURAS NEURONALES**<sup>55</sup>

En las neuronas del cerebro humano, los neurotransmisores alteran la condición eléctrica de la membrana celular en el extremo de las dendritas o en cualquier otro lugar del cuerpo de la neurona; y, dependiendo de la naturaleza del neurotransmisor liberado, el efecto es excitador o inhibitorio para la neurona receptora; en el cuerpo o soma de la neurona se suman los estímulos recibidos a través de todos los pares sinápticos y si se supera un cierto valor de potencial de activación, la neurona se activa y se genera un impulso que se transmite a lo largo del axón.

La comunicación de naturaleza electroquímica en cada par sináptico, produce que la velocidad de respuesta de una neurona biológica sea relativamente mucho más baja que los microprocesadores de silicio y el rango de frecuencia de los mensajes son de orden de un Kilohercio, dando como resultado un tiempo de respuesta en el orden de los milisegundos.

La arquitectura neuronal más utilizada para el reconocimiento de patrones es el perceptrón multicapa, entrenada con el algoritmo de retropropagación del error, sin embargo algunos inconvenientes pueden ocurrir al igual que con otras arquitecturas como el dilema de la elasticidad-plasticidad.

El dilema de la elasticidad-plasticidad es retener lo aprendido o adquirir nuevo conocimiento o por el dilema ruido-saturación o como aprender dando igual

---

<sup>54</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

<sup>55</sup> BANDA, H. Inteligencia Artificial Principios y Aplicaciones. EPN. 2014

importancia a todas las señales y en muchos casos es bueno que la red neuronal de este tipo sea considerada como caja negra en otros no, puede ser necesario averiguar el funcionamiento interno de la red.

Existen diversas arquitecturas de redes neuronales artificiales, según la forma cómo se interconectan las neuronas y cómo fluyen las señales en el interior de la red: Estructuras Monocapa, Estructuras Multicapa, Estructuras en Malla, Estructuras con conexiones hacia adelante (feedforward), Estructuras con conexiones hacia adelante y hacia atrás (recurrentes); y, Radiales.

Algunos ejemplos de arquitecturas neuronales más comunes pueden observarse en libro Inteligencia Artificial de Hugo Banda o en la Guía de usuario de Nntool de Matlab.

### 3.4.1 REDES MONO CAPA

Las redes neuronales mono capa son redes con una sola capa, para unirse las neuronas crean conexiones laterales para conectarse con otras neuronas, pueden ser de conexiones autorrecurrentes que permiten que las neuronas tengan conexiones a sí mismas y no autorrecurrentes; de las autorrecurrentes, las más representativas son la red BRAIN-STATE-IN-A-BOX, memorias asociativas lineales, Additive Grossberg y Shuting Grossbergx; y, de las arquitecturas de conexiones no autorrecurrentes, la red de Hopfield, las máquinas estocásticas de Botzmann y Cauchy; las redes neuronales monocapa, son más utilizadas en circuitos eléctricos; la figura 3.12 muestra la arquitectura de una red neuronal monocapa.

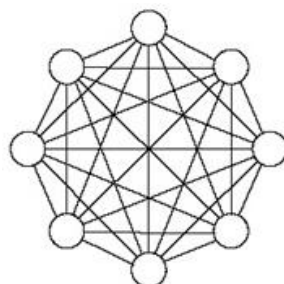


Figura 3.12 Arquitectura de una red neuronal mono capa<sup>56</sup>

---

<sup>56</sup> <http://www.redes-neuronales.com.es>



### **3.4.2 REDES MULTICAPA**

La arquitectura de redes neuronales multicapa están formadas por varias capas de neuronas, las capas están ordenadas por el orden en que reciben la señal desde la entrada hasta la salida se denominan conexiones feedforward o hacia delante; y si algunas capas están también unidas desde la salida hasta la entrada en orden inverso en que viajan las señales de información, se llaman conexiones feedback o hacia atrás y hacia adelante, las mismas que pueden tener o no conexiones laterales.

Las redes con conexiones hacia adelante contienen solo conexiones entre capas hacia delante, lo que implica que una capa no puede tener conexiones a una que reciba la señal antes que ella; las redes con conexiones hacia adelante y atrás si pueden tener conexiones de capas hacia atrás y por tanto la información puede regresar a capas anteriores.

### **3.4.3 ARQUITECTURA NEURONAL DE DOS CAPAS**

Este tipo de arquitectura se caracteriza por tener dos capas de neuronas, una de entrada y una de salida; en el caso de las redes feedforward o hacia delante sin conexiones laterales, las más conocidas son las adaline, madeline, el perceptrón simple y el drive reinforcement; las que tienen conexiones laterales son las LVQ o learning vector quantizer y las TPM o topology preserving map.

Las redes neuronales bicapa con conexiones hacia adelante y atrás, sin conexiones laterales más conocidas son BAM o bidireccional associative memory, asaptative bam, las temporal associative memory o TAM y las fuzzy asociative memory o FAM; ejemplos de redes neuronales que si tienen conexiones laterales son las BAM y las ART adaptative resonance theory.

### **3.4.4 REDES NEURONALES DE TRES CAPAS**

Las redes neuronales con conexiones hacia adelante sin conexiones laterales más conocidas son adaptative euristic critic, máquina de Cauchy y máquina de Boltzman, y las que sí tienen conexiones laterales además de la de Cauchy y de Boltzman la courier propagation. Las máquinas de cauchy y de boltzman pueden

ser de conexiones hacia adelante y atrás. La figura 3.13 muestra la arquitectura de tres capas.

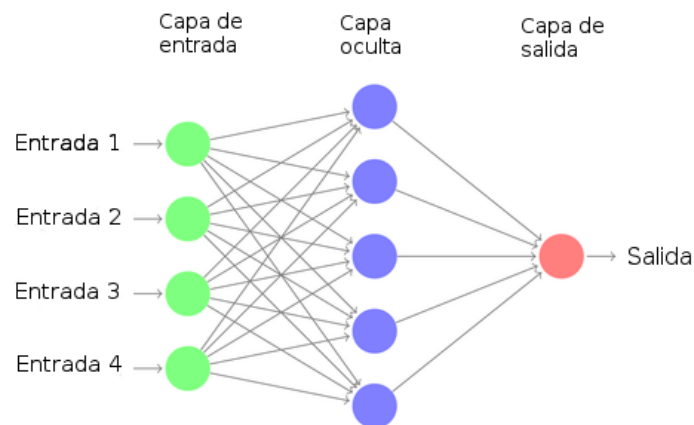


Figura 3.13 Red Neuronal con Arquitectura de Tres Capas<sup>57</sup>

### 3.4.5 REDES NEURONALES n CAPA

En este tipo de arquitectura la más reconocida es la que tiene conexiones hacia adelante back propagation y de la arquitectura con conexiones hacia adelante y atrás las cognition y las neocognition. En el libro Inteligencia Artificial de Hugo Banda puede observarse una distribución gráfica sencilla de la clasificación de las redes neuronales por su arquitectura.

### 3.5 CONTROL DE ROBOTS MANIPULADORES USANDO REDES NEURONALES

Existen varios trabajos acerca de control de robots manipuladores usando redes neuronales como el de Peña Martínez 2008, en el cual se realiza el control de un robot tipo PUMA de tres grados de libertad 3DOF (3 degree of free ) (cintura, hombro y codo) utilizando celdas neuronales analógicas que permite emular el comportamiento del brazo humano considerando el paradigma del generador central de patrones, que es capaz de reproducir los patrones requeridos para controlar las extremidades de los animales, implementa el modelo matemático de una red neuronal celular para obtener un generador central de patrones que genera las señales de excitación apropiadas que permitan provocar movimientos

<sup>57</sup> <http://taycalbacete.blogspot.com>

periódicos del brazo, similar al de una persona que camina, pues el patrón de actividad locomotora se debe a un patrón de actividad neuronal.

El principal componente del sistema motriz es el generador central de patrones es un circuito neuronal que produce el patrón motriz rítmico sin necesidad de sensores que permitan la retroalimentación del sistema de control, tal como el patrón motriz rítmico que se localiza en la columna vertebral en los vertebrados o en los ganglios en los invertebrados.

Los robots con capacidades sensoriales constituyen la última generación de este tipo de máquinas, el uso de estos robots en los ambientes industriales es muy escaso debido a su elevado costo, a pesar de todo, la investigación sobre los aparatos sensoriales está en pleno apogeo, lo que conducirá seguramente a un abaratamiento de éstos y a un aumento de su potencia y de sus capacidades, hoy en día se usan en cadenas de embotellado para comprobar si las botellas están llenas o si la etiqueta está bien colocada.

### **3.5.1 APLICACIONES EN ROBÓTICA CON NNTOOL DE MATLAB**

La aplicación exitosa de las redes neuronales artificiales, es que está dada para la solución de problemas en los cuales los recursos computacionales convencionales no son eficaces, problemas que requieren de razonamiento cualitativo y cuantitativo complejo y/o cuando el número de datos es extremadamente alto y son multivariados o existe ruido y errores o se tienen datos incompletos; se usan las redes neuronales artificiales para reconocimiento de patrones; validación, procesamiento e información de señales; procesos de monitoreo; diagnóstico y control de sistemas no lineales complejos como el caso de la solución de la cinemática inversa del robot.

La aplicación de las redes neuronales al control de brazos robóticos manipuladores para la clasificación de objetos, los cuales cuando se requiere

resolver por el modelo de la cinemática inversa resulta ser un sistema no lineal complejo para lo que se usa, nntool del Toolbox Neuro de la Plataforma Matlab®, y distribuida por The Mathworks Inc. Que es una caja de herramientas para la solución de este tipo de problemas, la cual es una eficaz aplicación para diseñar, implementar, visualizar y simular redes neuronales que resuelven los problemas no lineales con relativa facilidad, en este caso particular para la clasificación de objetos. La figura 3.14 muestra la estructura de un robot industrial.

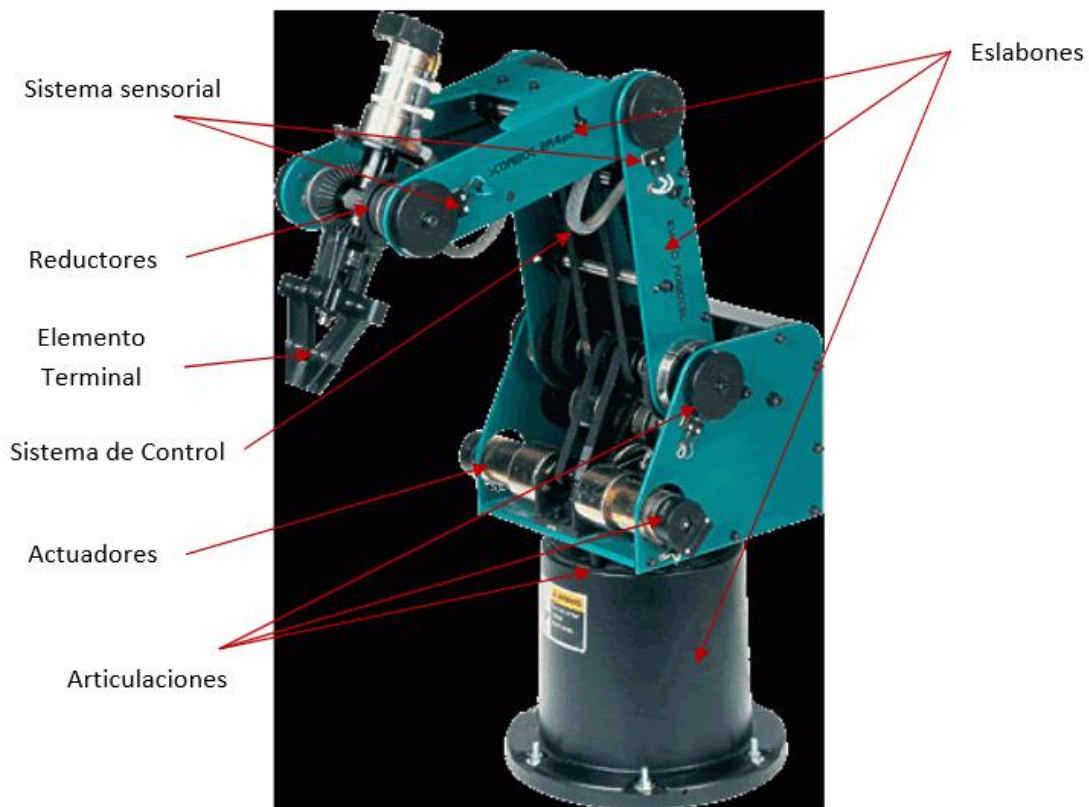


Figura 3.14 Modelo de un robot manipulador esférico industrial<sup>58</sup>

El uso de los parámetros de Denavit – Hartenberg DH sirve para el análisis cinemático directo e inverso de cualquier configuración de un robot y existe una fórmula general que permite calcular la localización del extremo efector del robot conocido como algoritmo de Denavit-Hartenberg.

<sup>58</sup> MOYA, 2010

### 3.5.2 CINEMÁTICA DEL ROBOT

El estudio cinemático del robot considera el movimiento como una función del tiempo con respecto a un sistema de referencia sin considerar las fuerzas que intervienen y la relación entre la posición y la orientación del extremo final o efector del robot con los valores que toman sus coordenadas articulares.

Los dos problemas a resolver en la cinemática del robot son: el problema de la cinemática directa y de la cinemática inversa; la cinemática directa consiste en determinar la posición y orientación del efector del robot, con respecto a un sistema de coordenadas de referencia normalmente el fijo o solidario a la tierra, siendo conocidos los valores de las articulaciones y la geometría de los elementos del robot; la cinemática inversa, resuelve la configuración angular del robot para una posición y orientación del extremo conocidas. La figura 3.15 muestra gráficamente la relación entre los parámetros y la cinemática del robot.

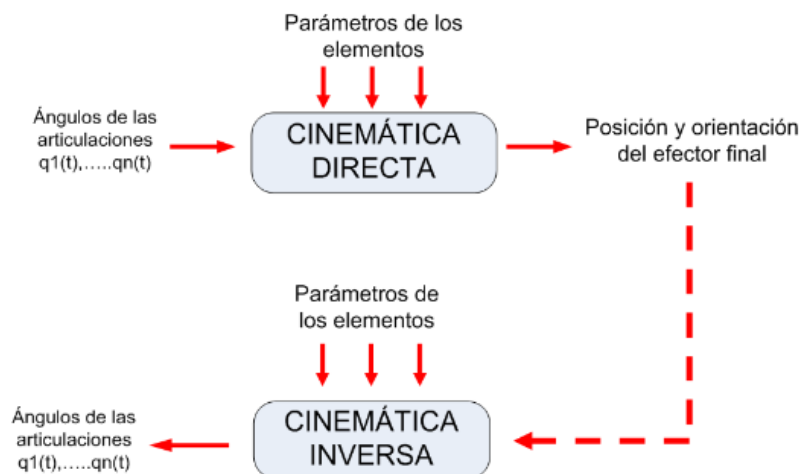


Figura 3.15 Cinemática directa y cinemática inversa<sup>59</sup>

### 3.5.3 CINEMÁTICA DIRECTA

Las variables articulares son leídas directamente por los sensores de la unidad de control del robot, el modelo cinemático directo presenta información relativa a la localización del extremo del robot. El método de Denavit - Hartenberg sirve

<sup>59</sup> MOYA, 2010

para describir y representar la geometría de los elementos del robot, con respecto a un sistema de referencia fijo que utiliza una matriz de transformación homogénea para describir la relación entre dos eslabones adyacentes, el problema cinemático directo consiste en encontrar una matriz de transformación homogénea que relacione la localización del extremo del robot con respecto al sistema de coordenadas de su base, además la cinemática establece las relaciones entre las velocidades del movimiento de las articulaciones y las del efector, esta relación viene dada por el método diferencial de la matriz Jacobiana.

El método matricial de Denavit - Hartenberg establece la localización que debe tomar cada sistema de coordenadas ligado a cada eslabón  $i$  de una cadena articulada; para sistematizar la obtención de las ecuaciones cinemáticas de la cadena, Matlab proporciona una herramienta para los cálculos, en el diseño de robots, la Toolbox de Robótica desarrollado por Peter Corke en 1996.

#### **3.5.4 CINEMÁTICA INVERSA**

La solución de la cinemática inversa consiste en encontrar los valores que deben tomar las coordenadas articulares del robot para que su extremo alcance una posición determinada, es decir, dada la posición del efector final del robot, la solución de la cinemática inversa consiste en calcular todos los posibles conjuntos de ángulos entre las articulaciones que podrían usarse para obtener la posición deseada.

Las ecuaciones que rigen la cinemática inversa dependen del tipo de robot, el problema de la cinemática inversa es más complejo que la cinemática directa ya que las ecuaciones no son lineales, sus soluciones no son cerradas e incluso, puede haber una o más soluciones; la existencia o no de la solución define el espacio de trabajo del robot, la ausencia de una solución significa que el robot no puede alcanzar la posición deseada, porque se encuentra fuera del espacio de trabajo o fuera de los rangos permisibles de cada una de sus articulaciones.

La toolbox de robótica de Matlab tiene la función `ikine`, que mediante métodos iterativos, resuelve el problema de la cinemática inversa del robot, que por utilizar un algoritmo iterativo y no una solución cerrada que puede consumir excesivos

recursos computacionales y tardar en converger a una solución o no converger y en el caso de haber soluciones múltiples proporcionará una de ellas; para su solución, en el manual de Peter Corke las novedades y cambios son los que se muestran a continuación.

La solución de la cinemática inversa se pueden encontrar para robots con 2, 3 o 6 grados de libertad en Serial Link.ikinesym; que es nuevo de 2014, la función ikine6s para una serie de casos específicos: robot sin desfase del hombro, robot con desplazamiento del hombro (puede tener configuración zurdo / derecho), el brazo de Stanford (tercera articulación prismática), brazo Puma 560. El código anterior hizo un montón de hipótesis aplicables al Puma, lo que provocó los errores de otros robots de 6 ejes con muñecas esféricas, esto está en Robotics Toolbox 9.9 for Matlab® Copyright© Peter Corke 2014.

### **3.6 APLICACIÓN DE LAS REDES NEURONALES A LA SELECCIÓN DE OBJETOS**

En las últimas tres décadas, actividades intensivas de investigación se han dedicado al diseño de sistemas de control robusto y adaptativo, sistemas de control para robots con el fin de superar inconvenientes del sistema de control causados por los errores en el modelo y parámetros inciertos, y un gran número de investigaciones, cuyos resultados han sido reportados, sin embargo, casi en todas partes asocian el control complicado en el diseño del sistema y dificultades en el control de sistema aplicación especialmente para robots manipuladores industriales.

#### **3.6.1 MODELO DE ZHAO-HUI JIAN**

En el Journal IJCA aparece un artículo en 2008 sobre un controlador de red neuronal que fue diseñado para controlar los movimientos de un robot manipulador scara a lo largo de trayectorias específicas, en base a un trabajo realizado por Zhao-Hui Jiang del Departamento de Ingeniería de Sistemas Mecánicos del Instituto Tecnológico de Hiroshima, en el que se llegan a conclusiones como que para que el modelo funcione correctamente requiere un modelo dinámico correcto y también los parámetros precisos del robot, pues en

la práctica, cada modelo dinámico contiene algunos grados de incorrección y cada parámetro se asocia con algunos grados de error lo que finalmente lleva a errores de seguimiento de trayectoria y posicionamiento, incluso llegando a causar la inestabilidad del sistema.

En este trabajo Jian propuso un esquema de control basado en la tecnología de redes neuronales artificiales con realimentación lineal para el seguimiento de una trayectoria planificada, el sistema de control está diseñado con dos subsistemas paralelos por separado, un controlador lineal, y otro es el controlador de red neuronal, en el que el seguimiento de la trayectoria y la minimización del error con la actualización de los pesos, para generación de la fuerza y par de torsión requerido por el sistema dinámico.

### **3.6.2 CONTROL ROBUSTO**

El control robusto en robótica utiliza más de un método y más de un algoritmo para resolver los problemas más complejos y hacer de las máquinas cada vez más cercanas y amigables para el ser humano, por ejemplo en 2012 el International Journal of Computer Applications publica un artículo de Akash Dutt Dubey, R. Mishra y A.K. Jha, titulado “Optimización del tiempo de trabajo de un Robot Manipulador utilizando Redes Neuronales Artificiales y Algoritmos Genéticos”<sup>60</sup>.

La propuesta es un método evolutivo para optimizar el tiempo de trabajo de los robots manipuladores, las tareas pueden planificarse con respecto a las articulaciones del robot o con respecto al extremo del robot bajo restricciones cinético-dinámicas, el algoritmo genético se implementa para optimizar los parámetros asociados con el perfil de la trayectoria de movimiento seleccionado; los resultados optimizados fueron tomados como datos de entrenamiento para entrenar una red neuronal artificial que utilizaron para obtener el tiempo de la tarea, la velocidad, aceleraciones y pares de torsión requeridos por cada motor para llevar a cabo una tarea dada; el método adoptado en este estudio se puede aplicar a cualquier manipulador con eslabones rígidos y cinemática conocida y

---

<sup>60</sup> Akash Dutt Dubey, R.b:Mishra y A.K. Jha “Optimización del tiempo de trabajo de un Robot Manipulador utilizando Redes Neuronales Artificiales y Algoritmos Genéticos”. 2012



modelos dinámicos con movimientos a lo largo de las trayectorias especificadas con evasión de obstáculos; La cinemática y dinámica del robot y el método de optimización fueron desarrollados en Matlab.

### **3.6.3 RECONOCIMIENTO DE OBJETOS**

En 2011, aparece una publicación donde se muestra lo relevante que resulta la aplicación de las redes neuronales artificiales en el control de un brazo manipulador robótico para el reconocimiento de objetos en la Revista Ciencias Técnicas Agropecuarias, de los autores González S. León J. y Estrada Y. de la Universidad Agraria de La Habana; titulado “Sistema para el Control Inteligente de un Robot Manipulador”[28], los autores, desarrollaron una aplicación para el control inteligente de un robot manipulador de seis grados de libertad, mediante visión artificial, para la identificación de objetos por reconocimiento de patrones utilizando redes neuronales artificiales y para resolver el problema de la cinemática inversa del robot, desarrollaron un modelo matemático basado en trigonometría.

### **3.6.4 CLASIFICACIÓN DE OBJETOS CON REDES NEURONALES**

El objetivo del presente trabajo es mostrar la capacidad de la herramienta computacional nntool de Matlab para resolver una aplicación que permita dotar de inteligencia a un robot manipulador, a partir del reconocimiento de patrones utilizando redes neuronales para su clasificación, la aplicación es capaz de identificar objetos previamente definidos, y determinar su posición.

### **3.6.5 REDES NEURONALES ARTIFICIALES EN MATLAB**

Para trabajar en el reconocimiento de patrones y la selección de productos, que en este caso particular son productos agrícolas, un clasificador de rosas se cuenta con la toolbox nntool de Matlab, antes de desarrollar, cualquier trabajo se puede familiarizar con la herramienta entrando al ambiente de trabajo de Matlab y pulsar nntstart que indica los pasos a seguir para la utilización de la herramienta aparece el cuadro de dialogo que se indica en la figura 3.16.

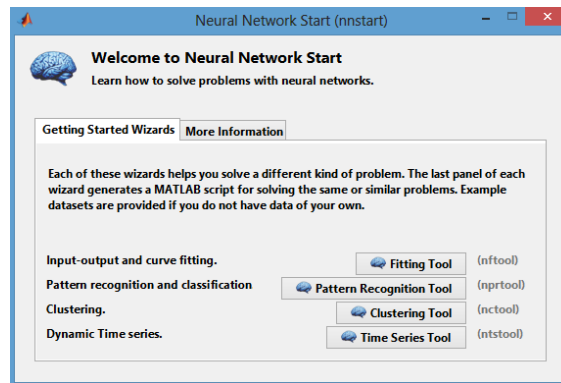


Figura 3.16 Cuadro de dialogo de nnstart de Matlab

Bienvenido a iniciando el trabajo con redes neuronales artificiales, aprender cómo solucionar problemas con redes neuronales con Matlab; cada uno de botones mostrados permite resolver un problema diferente y el último panel de cada pantalla genera una secuencia de comandos de Matlab para la solución de los mismos o problemas similares, y a manera de ejemplo conjuntos de datos son proporcionados si no se dispone datos reales.

La herramienta para problemas de ajuste (fitting), si se quiere una red neuronal para asignar entre un conjunto de datos de las entradas numéricas y un conjunto de objetivos numéricos, ejemplos de este tipo de problemas incluyen la estimación de los precios de las casas comerciales, de estas variables de entrada como la tasa de impuesto, proporción alumno / profesor en las escuelas o la tasa de criminalidad; la estimación de los niveles de emisión de motores basados en mediciones de consumo de combustible y la velocidad (engine\_dataset); o predecir el nivel de grasa corporal de un paciente sobre la base de mediciones de cuerpo (bodyfat\_dataset). La herramienta de montaje de red neuronal ayudará a seleccionar los datos, crear y entrenar a una red, y evaluar su desempeño usando error cuadrático medio y análisis de regresión.

Una red de alimentación directa de dos capas con función de activación sigmoideas, con una capa de neuronas oculta y las neuronas de salida lineal (FITNET), puede adaptarse a los problemas de asignación multidimensionales con datos consistentes y suficientes neuronas en su capa oculta; la red será entrenada con el algoritmo backpropagation Levenberg-Marquardt (trainlm), a no ser que no haya suficiente memoria, en cuyo caso se utilizará backpropagation

gradiente conjugado escalado (trainscg), la función de ajuste es el proceso de entrenamiento de una red neuronal con un conjunto de entradas con el fin de producir un conjunto asociado de salidas deseadas.

Una vez que la red neuronal está ajustada a los datos, se realiza la generalización de la relación entrada-salida y pueden ser utilizados para generar salidas para insumos para los que no fue entrenada, este conjunto de datos se puede utilizar para demostrar cómo puede ser una red neuronal entrenada para estimar la relación entre dos conjuntos de datos.

En los problemas de agrupamiento (clustering), se desea una red neuronal para agrupar datos por similitud, por ejemplo: la segmentación del mercado realizado por agrupación de la gente de acuerdo con sus patrones de compra; bases de datos se pueden hacer mediante la partición de datos en subconjuntos relacionados; o el análisis bioinformático, tales como agrupar genes con patrones de expresión relacionados; la herramienta Clustering Neural Network ayuda a seleccionar los datos, crear y entrenar a una red, y evaluar su desempeño usando una variedad de herramientas de visualización.

Las series de tiempo (time series) se utiliza para la predicción, es una especie de filtrado dinámico, en el que se utilizan valores pasados de una o más series de tiempo para predecir valores futuros, se utilizan redes neuronales dinámicas, que incluyen líneas de retardo se utilizan para el filtrado no lineal y predicción; hay muchas aplicaciones para la predicción, por ejemplo, un analista financiero podría querer predecir el valor futuro de una acción, bono u otro instrumento financiero, un ingeniero puede ser que desee para predecir el fracaso inminente de un motor a reacción. Los modelos de predicción también se utilizan para la identificación del sistema (o modelado dinámico), para construir modelos dinámicos de sistemas físicos, modelos dinámicos que son importantes para el análisis, la simulación, seguimiento y control de una variedad de sistemas, incluyendo sistemas de fabricación, procesos químicos, robótica y sistemas aeroespaciales. Time Series es una herramienta que permite resolver tres tipos de problemas de series de tiempo no lineales.

### 3.6.6 HERRAMIENTA RECONOCIMIENTO DE PATRONES NPRTOOL

Cuando se hace clic en el botón pattern recognition tool de la caja de diálogo mostrada en la figura 3.16, aparece el cuadro de dialogo que se muestra en la figura 3.17 donde se indica que en los problemas de reconocimiento de patrones, se tiene una red neuronal para clasificar entradas en un conjunto de categorías objetivo. Esta herramienta ayudará a seleccionar los datos, crear y entrenar a una red; y, evaluar su desempeño; ejemplos de este tipo, pueden ser el reconocer la viña de procedencia, de una botella de vino tinto especial, basado en el análisis químico (wine\_dataset); o clasificar un tumor como benigno o maligno, basada en la uniformidad de tamaño de celda, el espesor macizo, la mitosis (cancer\_dataset). El modelo de red neuronal es una herramienta de reconocimiento que ayuda a seleccionar los datos, crear y entrenar a una red, y evaluar su desempeño utilizando matrices cruzadas entropía y confusión.

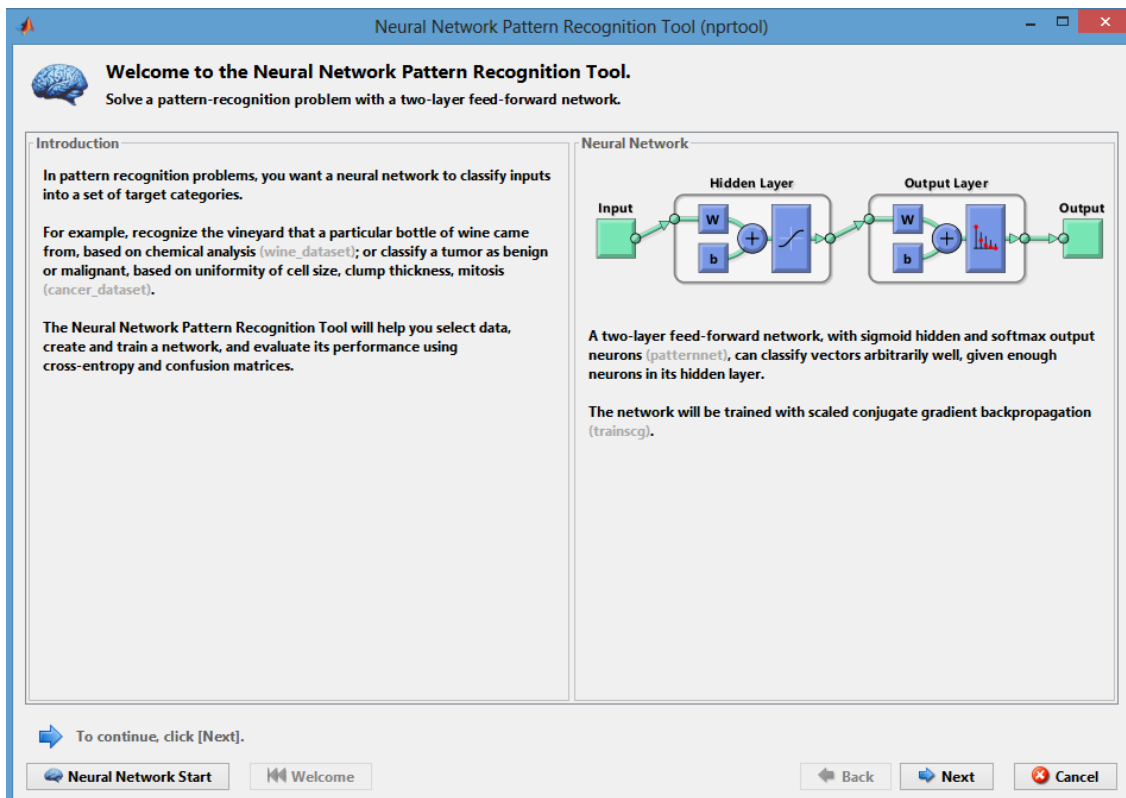


Figura 3.17 Red Neuronal para Reconocimiento de Patrones de Matlab

Una red feed-forward de dos capas, con función de transferencia tipo sigmoidea y con una capa oculta y neuronas de salida Softmax (patrón de red), puede

clasificar arbitrariamente vectores, dado suficientes neuronas en su capa oculta, la red se entrena con escalado backpropagation de gradiente conjugado (trainscg).

### 3.6.7 APLICACIÓN EN MATLAB DE CLASIFICACIÓN DE PATRONES

Las aplicaciones exitosas de redes neuronales artificiales, está dada para la solución de problemas en la medida que la tecnología computacional convencional no resulta eficaz, en los cuales se requiere de un razonamiento cualitativo y cuantitativo complejo y cuando el número de datos son extremadamente altos a la vez son multivariados y existe ruido y errores o existen datos incompletos; generalmente se usan las redes neuronales artificiales para reconocimiento de patrones, validación de señales, procesos de monitoreo, diagnósticos, procesamiento de señales e información y control de sistemas no lineales complejos.

Matlab ofrece una caja de herramientas para la solución de este tipo de problemas que se conoce como Neural Network Toolbox™, la cual es una caja de herramientas para diseñar, implementar, visualizar y simular redes neuronales. [The Mathworks Inc, 2009].

Previo al trabajo se ingresa en Matlab y en la ventana de inicio que se presenta en la figura 3.18, es aquí desde donde se puede iniciar un nuevo proyecto, al tipear nntool, enter en la ventana de comandos, se despliega la siguiente caja de diálogo, para iniciar el trabajo.

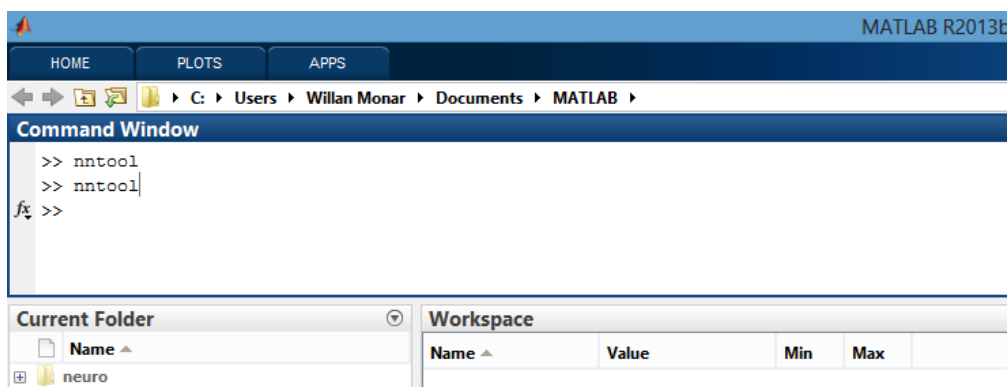


Figura 3.18 Ventana de inicio de nntool de matlab.

Una vez que se ha escrito en f>> el comando nntool, aparece una ventana que permite realizar un nuevo proyecto con redes neuronales, utilizando esta herramienta como muestra la figura 3.19.

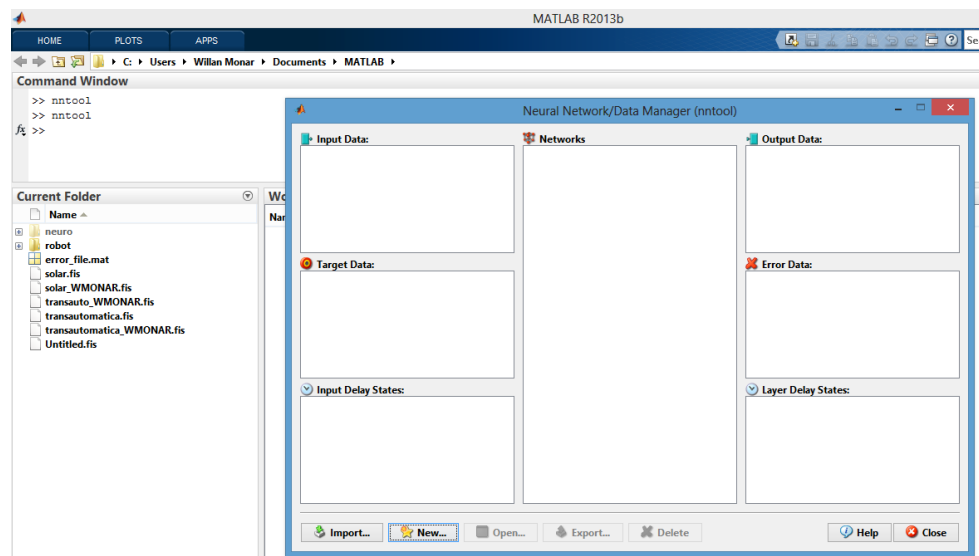


Figura 3.19 Caja de herramientas de redes neuronales en Matlab.

Para empezar a trabajar en un nuevo problema hacer lo siguiente: importar desde una base de datos o crear y los datos de entrada y los objetivos del espacio de trabajo con el comando [Import], crear una nueva red con el comando [New], seleccionar la red en la lista de red y dar clic en [open], seleccionar la pestaña de formación en la ventana de red, poner los datos de selección de entrada de entrenamiento y de salida, y dar clic en [Train], también puede utilizar la ventana de red para simular la red, o realizar otras tareas como la reiniciación y edición de pesos. En las tablas 3.3 y 3.4 se muestra la descripción de cada comando y la descripción de cada ítem respectivamente.

Tabla 3.3 Descripción de los Comandos de nntool

[Import] [Importar]	Importa datos y redes de área de trabajo o de un archivo.
[New] [Nuevo]	Permite crear una red o datos dentro de un cuadro de diálogo.
[Open] [Abrir]	Abre los datos o red seleccionados para la visualización y edición.
[Export] [Exportar]	Exporta datos y redes de área de trabajo o de un archivo.
[Delete] [Borrar]	Elimina los datos o red seleccionados.
[Help] [Ayuda]	Abre esta ventana.

Tabla 3.4 Descripción de cada ítem

Networks= Redes	Lista de redes.
Inputs= Entradas	Datos para presentar a una red.
Targets= Objetivos	Los datos que definen las salidas de red deseadas.
Outputs= Salidas	Respuesta de una red a sus entradas.
Errors= Errores	Diferencia entre objetivos y resultados.
Input Delays= Retrasos de entrada	Retardo estados de entrada para las redes
Layer Delay= Retrasos de capa	Estados para redes con retrasos de capa.

Al hacer clic en New aparece en la siguiente pantalla un cuadro de diálogo que permite crear un nuevo ambiente de trabajo o una nueva red de trabajo; en este ambiente se debe identificar con el nombre de esta red de trabajo, permite seleccionar el tipo de red de trabajo, las training functions o funciones de formación, la función de adaptación y aprendizaje, la función de rendimiento, el número de capas de redes neuronales, las propiedades para cada capa, el número de neuronas por capa, y la función de transferencia, como se muestra en la figura 3.20.

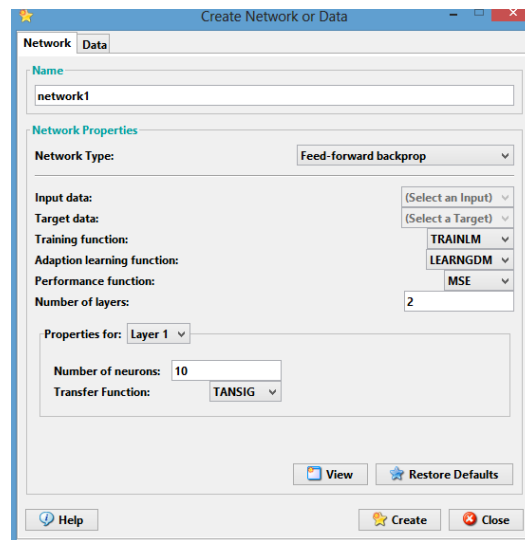


Figura 3.20 Creación de una Nueva Red Neuronal de Trabajo.

### 3.6.7.1 Tipo de Red de Trabajo

Las redes neuronales pueden ser del tipo supervisada y no supervisada o auto supervisada, las redes de trabajo supervisadas requieren de pares de datos entrada-salida correcta, mismas que se usan para predicción, evaluación o

generalización, en el entrenamiento aprenden a asociar un conjunto de datos de entrada con su correspondiente conjunto de salida; por otro lado una red del tipo auto supervisada clasifica patrones de entrada internamente y no requiere de resultados esperados por lo que su capacidad respecto de la red de aprendizaje supervisado es mínima se usan para clasificación y tipos de reconocimiento las más utilizadas son las redes Kohonen.

El tipo de red de trabajo para un determinado proyecto se puede seleccionar de un menú, las más utilizadas están la feed-forward backpropagation, que aparece por de fall o cualquier otro tipo a seleccionarse como por ejemplo la misma Forward backpropagation en cascada, fedd forward distributed time delay, feed forward time delay, generalized progression, Hopfield, layer recurrent, competitive, Elman backpropagation linear layer (desinglas), linear layer (train), LVQ, NARX, NARX Series-parallel, probabilistic, radial basis (exact fit), radial basis (fewer neurons), Self-organizing map, Como que indica la figura 3.21.

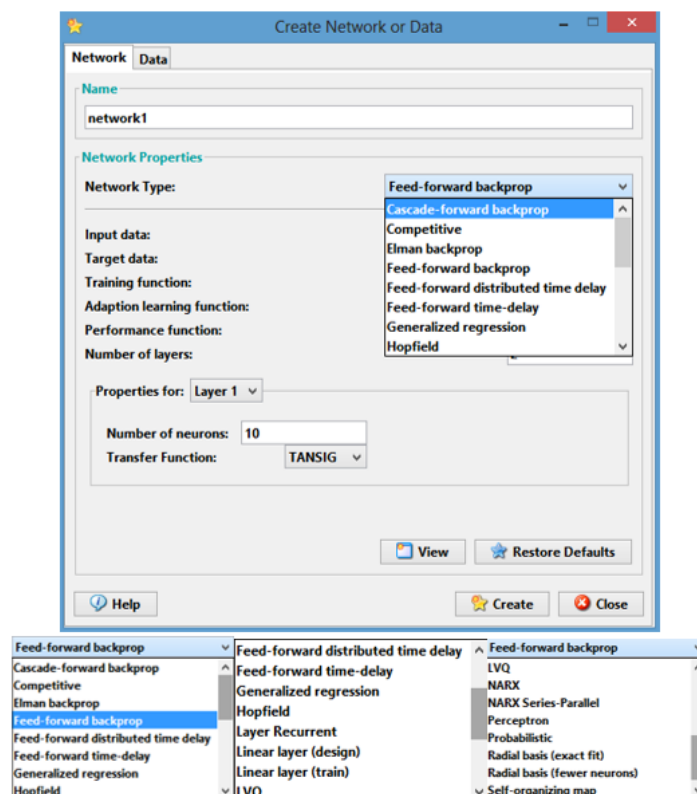


Figura 3.21 Tipos de Redes Neuronales en Ambiente Matlab.



### 3.6.7.2 Funciones de Entrenamiento

Las funciones de entrenamiento, de aprendizaje conocidas como training functions, son el modelo matemático que permite las iteraciones de los pesos y los bías, para que la red neuronal, aprenda en base en base de los datos proporcionados por el autor de entre las cuales se puede seleccionar: TRAINBFG, TRAINBR, TRAINCGB, TRAINCGP, TRAIINGD, TRAIINGDM, TRAIINGDA, TRAIINGDX, TRAINLM, TRAINOSS, TRAINR, TRAINRP, TRAINSCG; la caja de diálogo se muestra en la figura 3.22

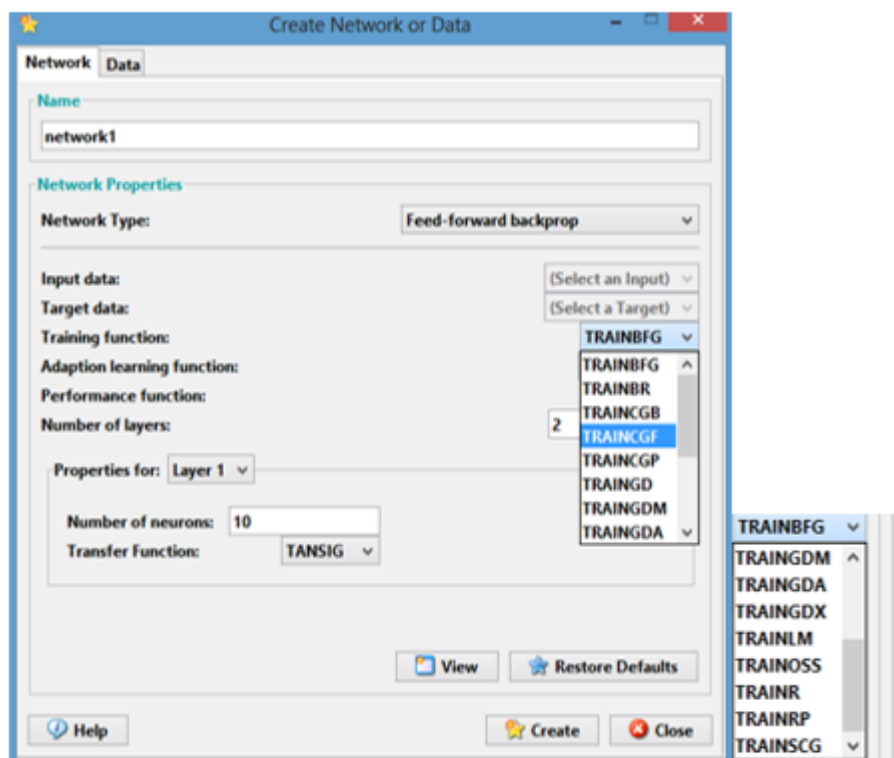


Figura 3.22 Caja de diálogo de la función de entrenamiento

### 3.6.7.3 La Funcion de Adaptación

El software permite seleccionar entre dos funciones de adaptación del aprendizaje, LEARNGD y LEARNGDM como se muestra en la figura 3.23.

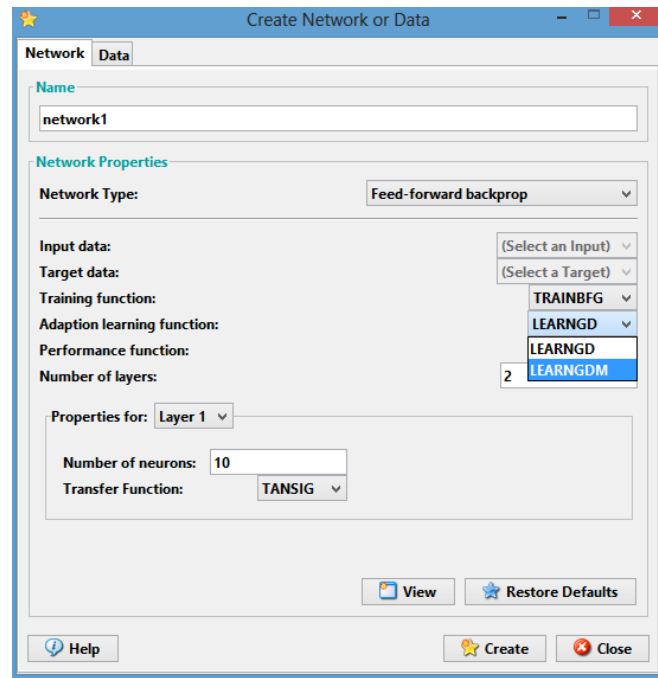


Figura 3.23 Función de Adaptación del Aprendizaje.

#### 3.6.7.4 La Función de Rendimiento

Las funciones de rendimiento o performance function pueden ser MSE, MSEREG y SSE. Como se muestra en la figura 3.24.

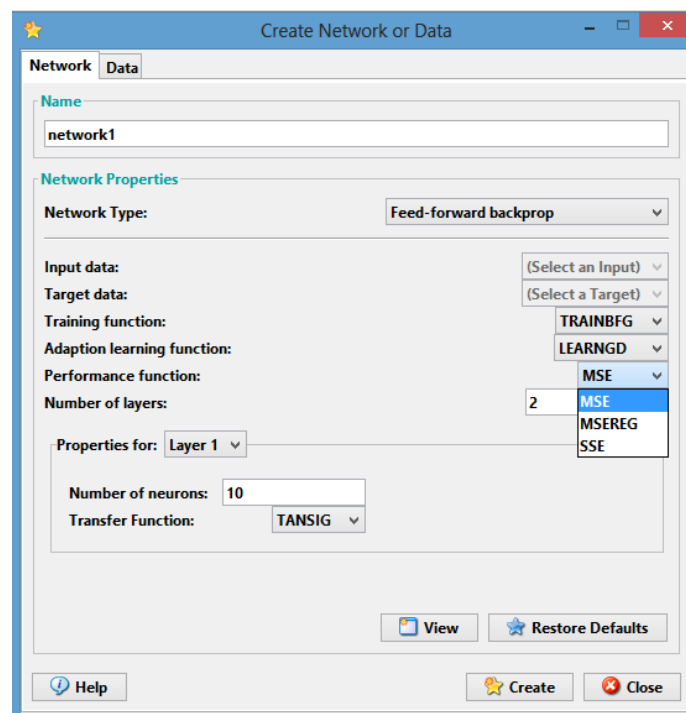


Figura 3.24 La Función de Rendimiento

### 3.6.7.5 Número de Capas

En esta ventana se puede seleccionar el número de capas de la red neuronal entre una o dos capas, como lo muestra la figura 3.25.

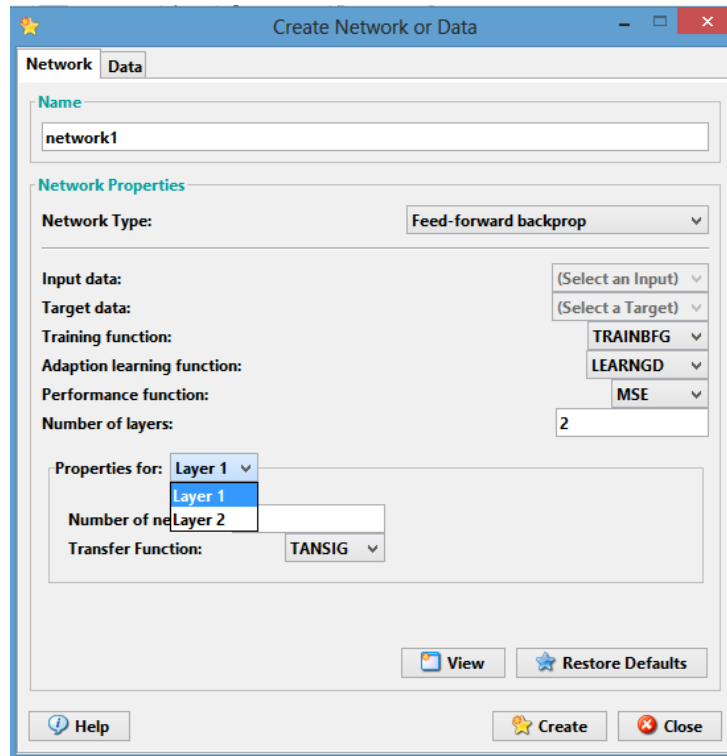


Figura 3.25 Número de Capas

### 3.6.7.6 Número de Neuronas

Se puede escoger cualquier número de neuronas dependiendo de la complejidad del sistema de control y el grado de incertidumbre y precisión que se desee manejar, así el número de elementos del vector de entrada, es decir el número de características extraídas es igual al número de neuronas de entrada, de igual manera para la salida el número de neuronas en la capa de salida depende de la selección del vector o vectores de salida.

### 3.6.7.7 Función de Activación

El software presenta las opciones para la función de transferencia del sistema de control las funciones LOGSIN, PURELIN y TANSIG, como se muestra en la figura 3.26.

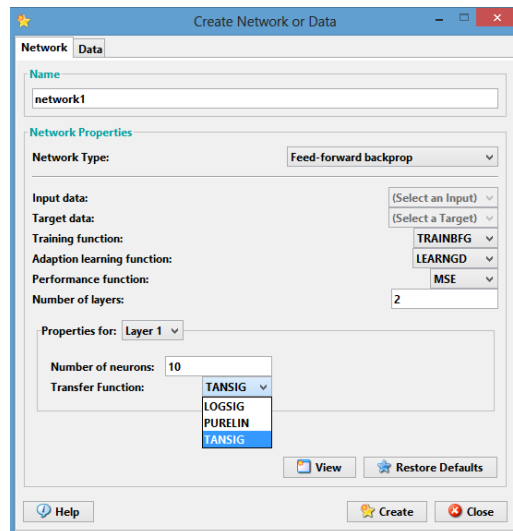


Figura 3.26 Funciones de transferencia para el sistema de control

Si se presiona Help el mensaje indica lo siguiente: Elija una arquitectura, a continuación, seleccione sus valores de propiedad, si la red tiene un número variable de capas, puede ver y editar las propiedades de una capa mediante la selección de la capa del menú desplegable proporcionado; una vez que se seleccionan todos los parámetros funcionales de la red neuronal que se va a utilizar se minimiza esta pantalla y se regresa a la anterior en la cual se introduce los datos: las entradas, los objetivos o targets, etc. Como se muestra en la figura 3.27.

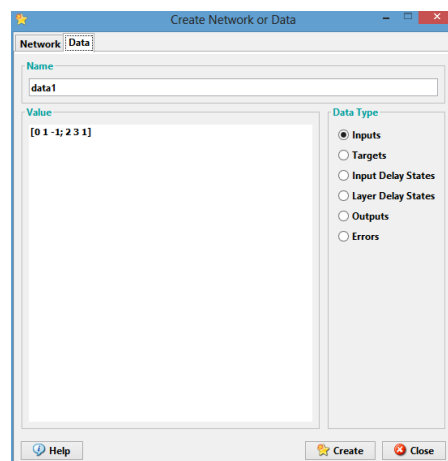


Figura 3.27 Ventana de ingreso de los Datos

Hay seis tipos de datos, la mayoría de las veces sólo se necesitará crear entradas y objetivos o targets para entrenar una red, las entradas son los datos que se presentarán a una red, los objetivos son los datos que definen las salidas

deseadas, las salidas son la respuesta de una red a sus entradas; los errores son la diferencia entre objetivos y resultados. Los retrasos de entrada son los estados de las redes con las demoras de entrada y retrasos de capa son los retardos de estado de capas para redes con retrasos de capa.

Por ejemplo para definir un conjunto de cuatro vectores de dos elementos de una matriz con cuatro columnas y dos filas.  $[0 \ 1 \ 4 \ 2; -1 \ 2 \ -1 \ 0]$ , para algunas redes, los datos necesitan ser dispuestos en una secuencia. El siguiente ejemplo muestra cómo representar una secuencia de vectores de dos elementos utilizando una matriz.  $\{[-1, 1] [0; 2] [0.5; -3]\}$ .

### 3.6.7.8 Clasificación de Rosas Utilizando NNtool de Matlab

Dentro de la clasificación de rosas en las categorías de exportación y flor nacional (aquella que no cumple con las características de exportación) existen elementos de análisis cuantitativos y cualitativos, al igual elementos de tipo fitosanitario que eliminan tallos con malas características para su posterior comportamiento destructivo o de apariencia a la llegada al cliente final, es importante que se obtenga una tabla con los atributos por categoría, de aquellas características de interés.

En este trabajo se seleccionan los parámetros cuantitativos que son más representativos para la clasificación en tres categorías que son flores tipo exportación, Milva, Esperance y Anastacia. En la tabla 3.5 se muestra el conjunto de entrenamiento para la selección de rosas.

Tabla 3.5 Conjunto de Entrenamiento para la Selección de Rosas.

MUESTRAS	Longitud de botón	Grosor de tallo	Largo de tallo	MILVA	ESPERRANCE	ANASTACIA
1	6	0,9	52	1	0	0
2	6	0,5	48	1	0	0
3	6,5	0,8	46	1	0	0
4	6	0,9	51	1	0	0
5	6	1	48	1	0	0
6	5,5	0,9	50	1	0	0
7	6	1	59	1	0	0
8	6	1	54	1	0	0
9	6	0,8	53	1	0	0
10	6	1,5	53	1	0	0
11	5	0,9	61	1	0	0
12	5,5	0,8	44	1	0	0

13	5,5	0,8	54	1	0	0
14	6	1	44	1	0	0
15	6	1	58	1	0	0
16	6	0,6	46	1	0	0
17	5,5	0,6	51	1	0	0
18	5,5	0,6	50	1	0	0
19	5,5	0,6	50	1	0	0
20	6	1	49	1	0	0
1	6,5	0,7	58	0	1	0
2	6	0,8	45	0	1	0
3	6	0,6	46	0	1	0
4	7	0,7	53	0	1	0
5	7	0,7	59	0	1	0
6	7	0,8	63	0	1	0
7	6,5	0,8	40	0	1	0
8	7	0,8	50	0	1	0
9	6,5	0,8	54	0	1	0
10	7,5	0,7	60	0	1	0
11	7,5	0,8	60	0	1	0
12	6,5	0,8	65	0	1	0
13	7	0,7	51	0	1	0
14	7,5	0,6	59	0	1	0
15	6	0,6	51	0	1	0
16	7	0,6	60	0	1	0
17	6	0,5	57	0	1	0
18	7	0,5	55	0	1	0
19	6	0,6	50	0	1	0
20	7	0,9	61	0	1	0
1	7	1	55	0	0	1
2	7,5	0,8	54	0	0	1
3	8	1	76	0	0	1
4	7	0,7	48	0	0	1
5	7,5	0,9	65	0	0	1
6	6	0,7	59	0	0	1
7	7,5	0,7	50	0	0	1
8	7	0,7	47	0	0	1
9	7,5	1	74	0	0	1
10	7	0,6	54	0	0	1
11	6	0,6	44	0	0	1
12	7,5	1	79	0	0	1
13	8	0,7	61	0	0	1
14	7	1	65	0	0	1
15	6,5	0,7	45	0	0	1
16	7	0,8	51	0	0	1
17	7,5	0,8	65	0	0	1

18	7,5	1	69	0	0	1
19	8	1	87	0	0	1
20	7	0,7	59	0	0	1

Con este conjunto de entrenamiento seleccionado por un experto se ingresa a Matlab, y en la pantalla principal de comandos se tipea `ntool` y `enter`, y se despliega el cuadro de dialogo que se muestra en la sección 3.6.7 en las figuras 3.19 y 3.20 que permite crear una nueva red neuronal.

Al hacer clic en `New`, se despliega el cuadro de diálogo que se muestra en en la figura 3.28, donde se coloca el nombre de la red de trabajo, por ejemplo en este caso la red se llamará `rosesuccess`, y además permite introducir los datos y los targets en la pestaña `Data`.

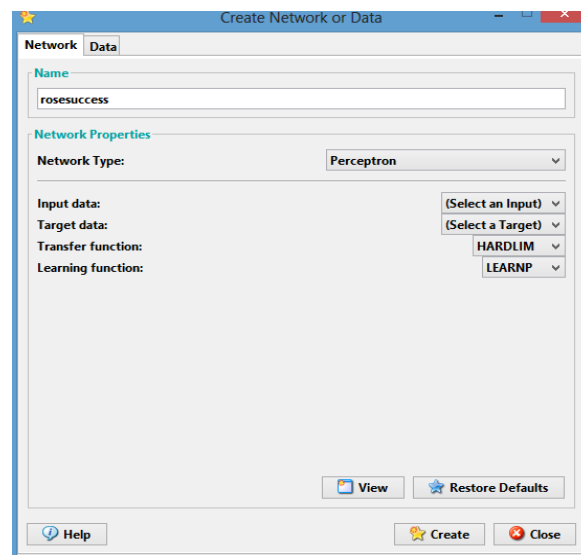


Figura 3.28 Nombre de la Red de Trabajo

En la caja `Data` se introducen los datos, de entrenamiento de la red, las entradas correspondientes a los atributos de la muestra de rosas en `inputs` como se observa en la figura 3.29. Y una vez introducidos los datos se hace clic en `create`, teniendo precaución de que los datos queden en una sola fila, cada dato se separa del siguiente con un espacio blanco y cada set de datos se separa con un





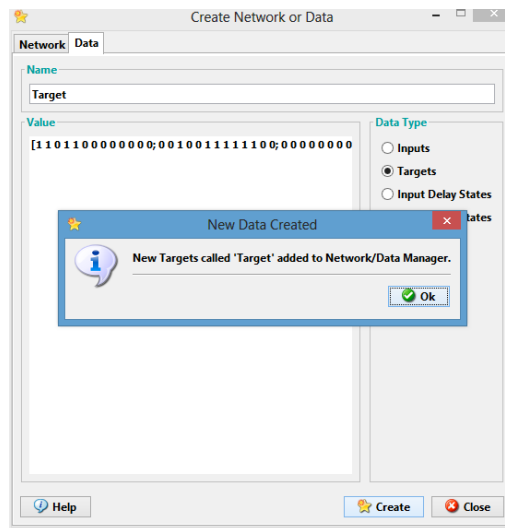


Fig. 3.31 Introducción de los Targets

Una vez que se han cargado al programa las entradas y los targets, se regresa al cuadro de diálogo de la figura 3.19, es decir a la red de trabajo, y se selecciona el tipo de red con la que se va a trabajar, se introducen las entradas y los targets, se selecciona el tipo de función de transferencia así como la función de aprendizaje. En este caso se elige el perceptrón y la función de transferencia la Hardlim o escalón esto es variando de cero a uno. Y como función de aprendizaje la función learnp como se muestra en la figura 3.32 y además se cargan los datos esto es las entradas y los targets.

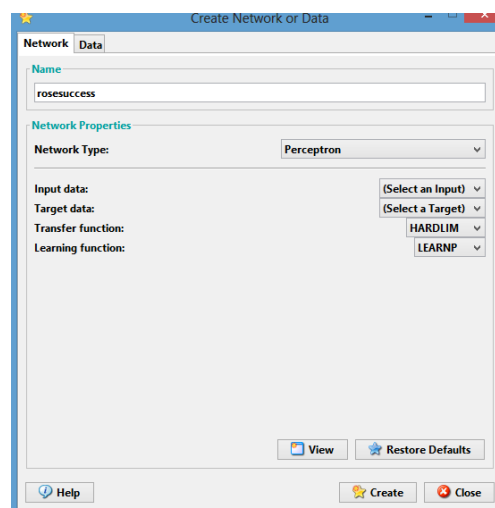


Figura 3.32 Red neuronal y parámetros seleccionados.

Si se pulsa la pestaña view se observa la red perceptrón que está trabajando está constituida por tres entradas, que son longitud del botón en centímetros, el largo de la rama en centímetros y diámetro del tallo valorados por el experto conocedor de las rosas. Esta red neuronal seleccionada, en este caso un perceptrón, tiene una capa con tres neuronas y tiene tres salidas que se pusieron al inicio que son exportación, producción nacional y desecho. La red neuronal perceptrón para este caso se muestra en la figura 3.33.

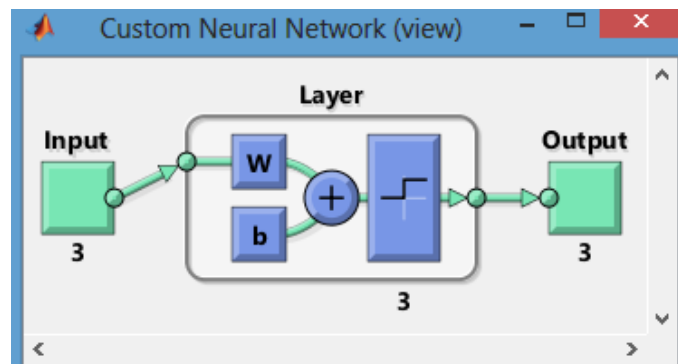


Figura 3.33 Red Perceptrón Mono capa de Tres Neuronas

Al hacer doble clic sobre el nombre de la red que se desea ver, en la caja de diálogo de la figura 2.34, aparece otra caja de dialogo que se muestra en la figura 3.35 donde se pueden apreciar las pestañas de la función de entrenamiento, la de simulación, la fase de adaptación la fase de re inicialización de los pesos sinápticos y la edición de los mismos. Al proceder a pulsar la pestaña train network (entrenamiento de la red) se podrá observar las salidas Outputs y el error que se despliegan para determinar si es la red seleccionada la adecuada o hay que seleccionar otro tipo de red que minimice el error. Se espera que las salidas sean las mismas targets entonces se podrá concluir que la red ha aprendido o que está entrenada para comportarse de la misma manera cuando se le coloquen las entradas reales y tenga que calcular las salidas y el error en cuyo caso deberá ser muy pequeño.

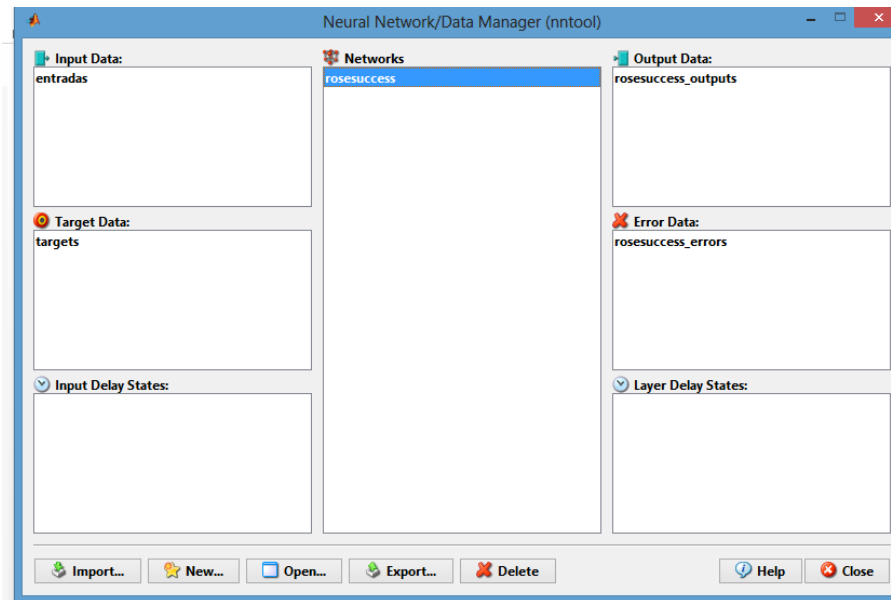


Figura 3.34 Selección de la Red a analizar

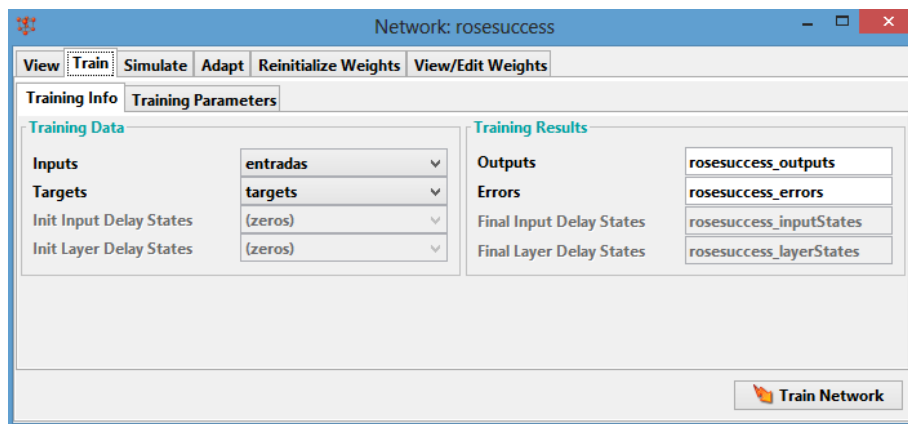


Figura 3.35 Función de entrenamiento

Al pulsar train network en la pantalla de la figura 3.35 la red neuronal empieza el entrenamiento y se puede observar el resultado del entrenamiento en la caja de diálogo mostrada en la figura 3.36 a continuación donde se puede ver que para que la red quede entrenada se necesitaron 1000 iteraciones, lo que a la máquina lo tomó un tiempo de 18 segundos y el rendimiento de esta red es 0.667, esto es del 67%, por lo que el error siendo la diferencia entre las salidas calculadas y los targets se esperaba que debía ser lo más cercano a cero y ahora se tiene 0.33 que se considera muy elevado, por lo que se concluye que esta red perceptrón no es la más adecuada para esta aplicación.



Se puede ver la re inicialización de los pesos en la figura 3.39

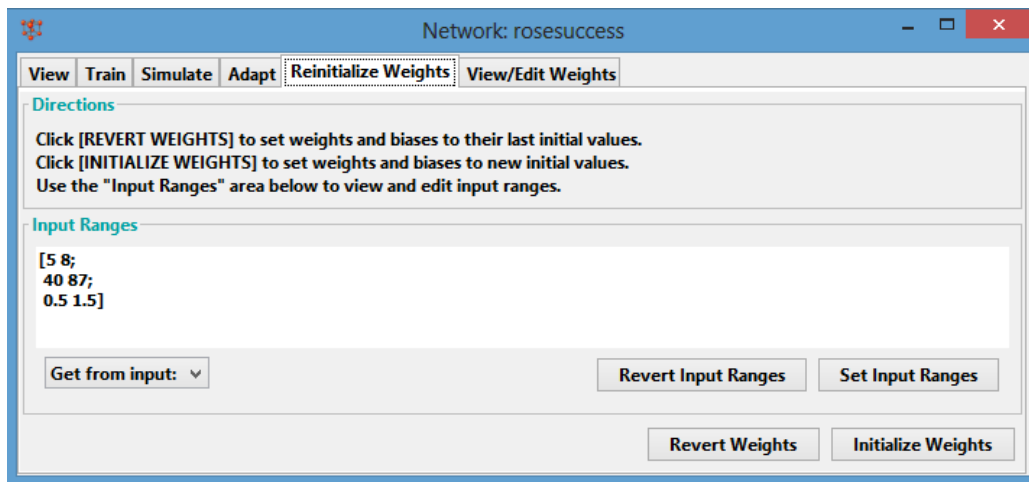


Figura 3.39 Re inicialización de los pesos

En la figura 3.40 se observa los ajustes de los pesos.

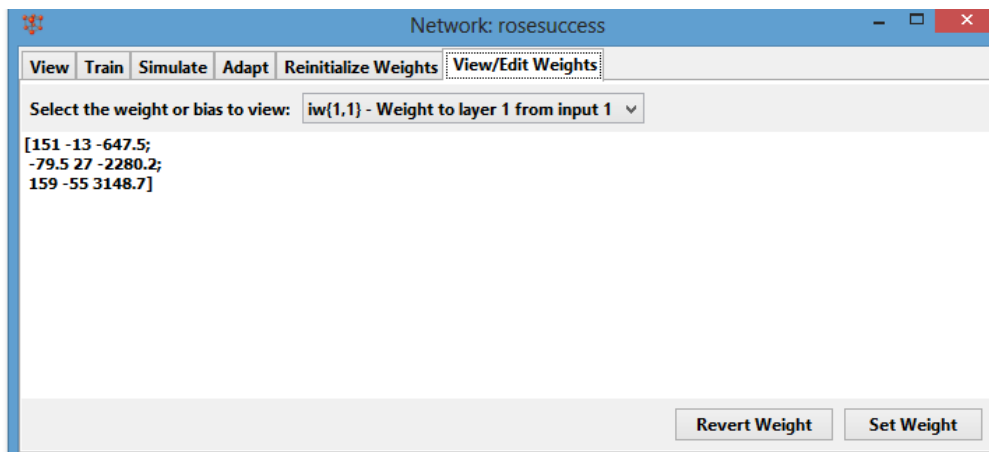


Figura 3.40 Ajuste de los pesos para la red Perceptrón

Con todo lo observado con la red perceptrón se concluye que para esta aplicación de selección de rosas no es un modelo satisfactorio pues las salidas después del entrenamiento no son las mismas que los targets, por lo que se va a probar con otra red neuronal, y analizar los parámetros analizados hasta aquí para el perceptrón.

Se procede a seleccionar la red neuronal Feed-forward backpropagation y se procede de la misma forma que con la red perceptrón y se obtienen los siguientes resultados: En la figura 3.41 se muestra la nueva selección que corresponde a una red neuronal Feed-forward backpropagation de 2 capas de neuronas con diez neuronas en la capa visible que está por default.

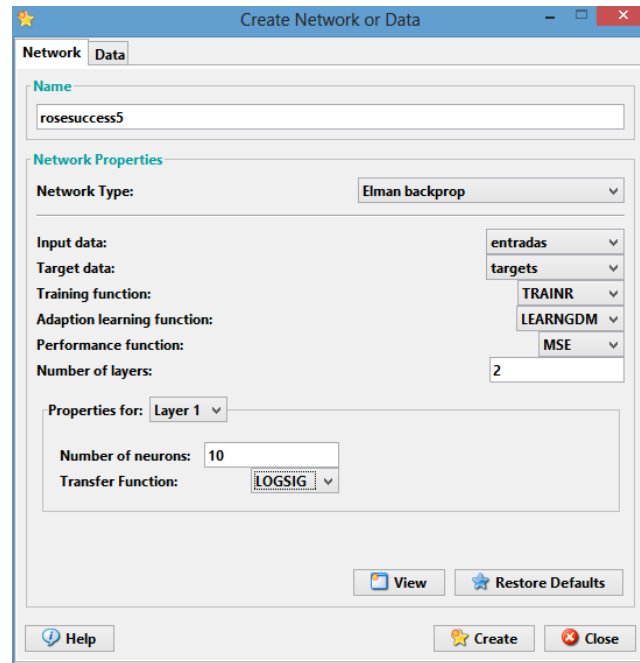


Figura 3.41 Selección de la red Fedd-forward backpropagation

La arquitectura de la red Feed-forward backpropagation se muestra en la figura 3.42 donde se observa que esta red tiene tres entradas, tres salidas y dos capas de neuronas con 10 neuronas en la capa visible y tres neuronas en la capa oculta, y en los dos casos la función de transferencia o activación es una función tangente sigmoidea bipolar.

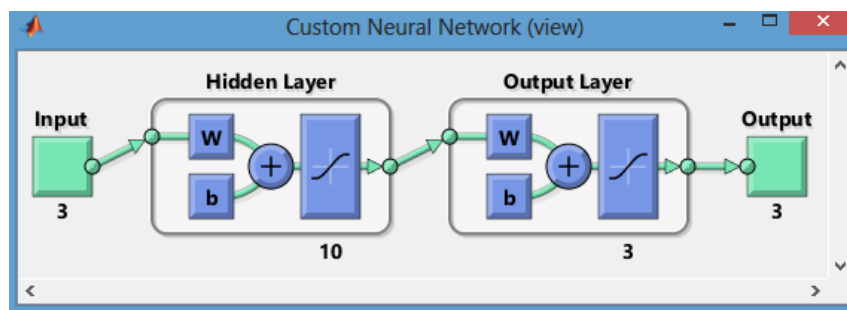


Figura 3.42 arquitectura de la red Feed-forward backpropagation

Pasando luego a la fase de entrenamiento de esta red, utilizando las mismas entradas y targets ya introducidos antes, se hace clic en train network, como se observa en la figura 3.43.

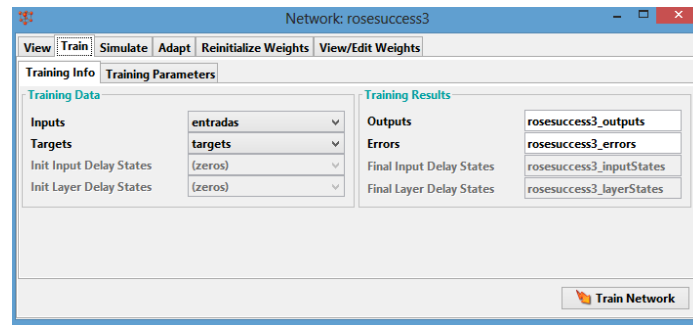


Figura 3.43 Entrenamiento de la red

En la figura 3.44 se puede observar la matriz de ajuste de los pesos de esta red Feed-forward backpropagation.

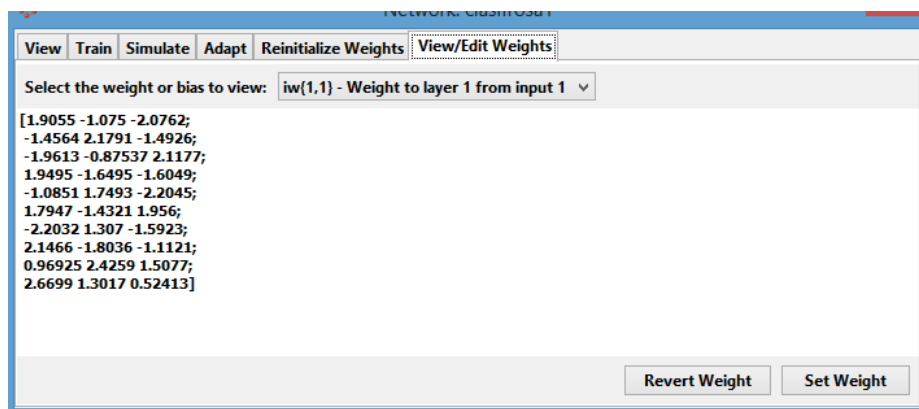


Figura 3.44 Matriz de ajuste de pesos de la red Feed-forward backpropagation

En la figura 3.45 se observa una captura de pantalla de los resultados del entrenamiento de la red feed-forward backpropagation cuyas salidas son muy parecidas a los targets introducidos al inicio, por lo que se concluye que esta red es mucho más adecuada para este caso de clasificación de las características de rosas con criterios de exportación, producción nacional y desecho.

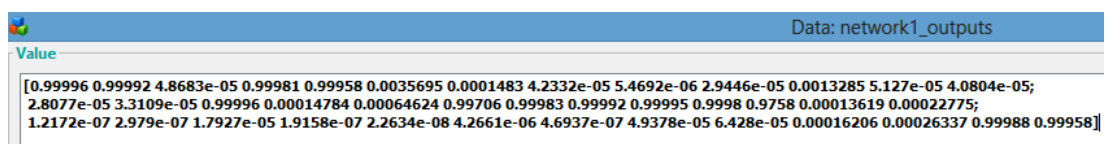


Figura 3.45 Salidas de la red Feed-forward backpropagation

En la tabla 3.6 se han colocado los targets, es decir las salidas deseadas, las puestas por un experto en las tres filas superiores y en la figura 3.45 están los valores calculados esto es las salidas luego del entrenamiento y se puede comparar y se tienen prácticamente los mismos valores de los targets, en esta tabla solo se incluyen los valores del primer vector de entrada, con lo que queda comprobado que la red ha aprendido.

Tabla 3.6 Tabla de Resultados

Target y salidas después del entrenamiento
111111110001110000101100101100001010000000000000010100000000
00000000111100011110100110100111001000100001010000010000000000
00000000000000000000000000000000010001101111010111100011111111

En la figura 3.46 se observa una captura de pantalla de los errores después del entrenamiento de la red Fedd-forward backpropagation, observándose que efectivamente son muy pequeños con lo que se concluye que esta red es más apropiada que la perceptrón.

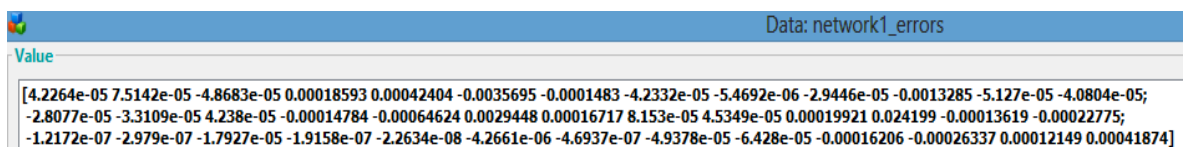


Figura 3.46 Errores en la red Feed-forward backpropagation

En la figura 3.47 se observa el comportamiento del error de la red feed-forward backpropagation de la clasificación de rosas, temiéndose en azul el error pesimista, en rojo el error optimista y en verde el error real de la red se ve que con 4 iteraciones es suficiente.



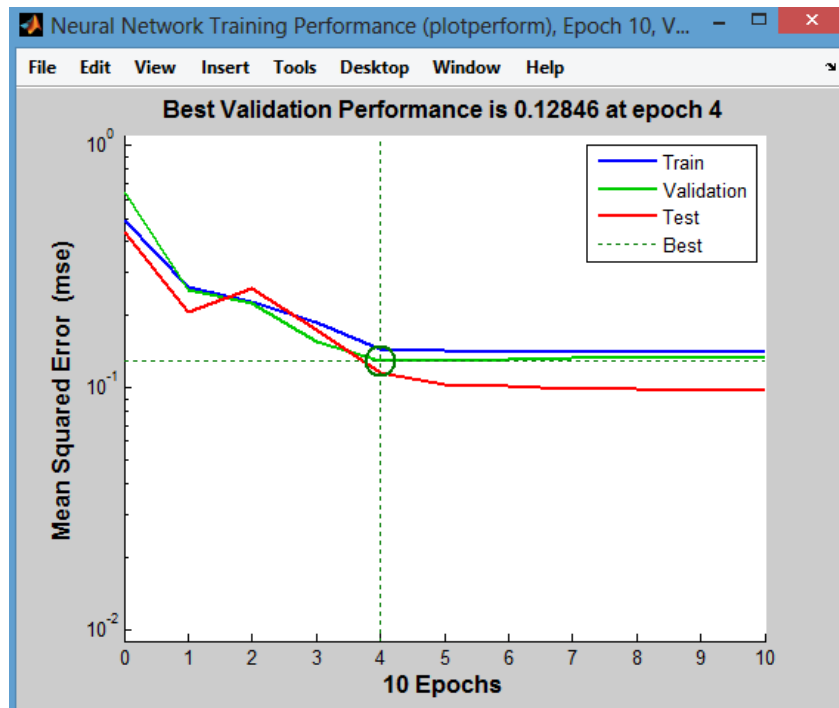


Figura 3.47 Error de la red Feed-forward backpropagation para la clasificación de rosas.

El estado de entrenamiento de la clasificación de rosas por la red neuronal Fedd-forward backpropagation se puede observar en la figura 3.48.

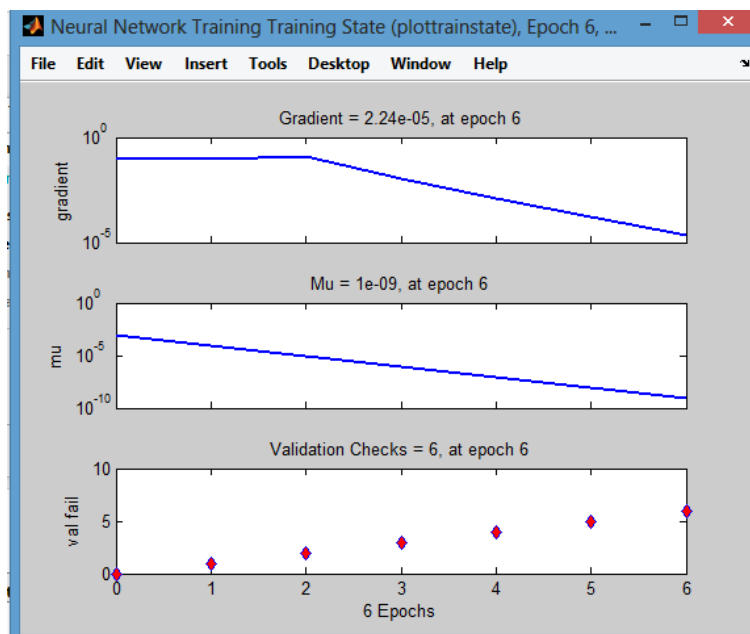


Figura 3.48 Estado de Entrenamiento

La regresión del error durante el entrenamiento, observada en el entrenamiento de la red Feed-forward backpropagation, se muestra en la figura 3.49

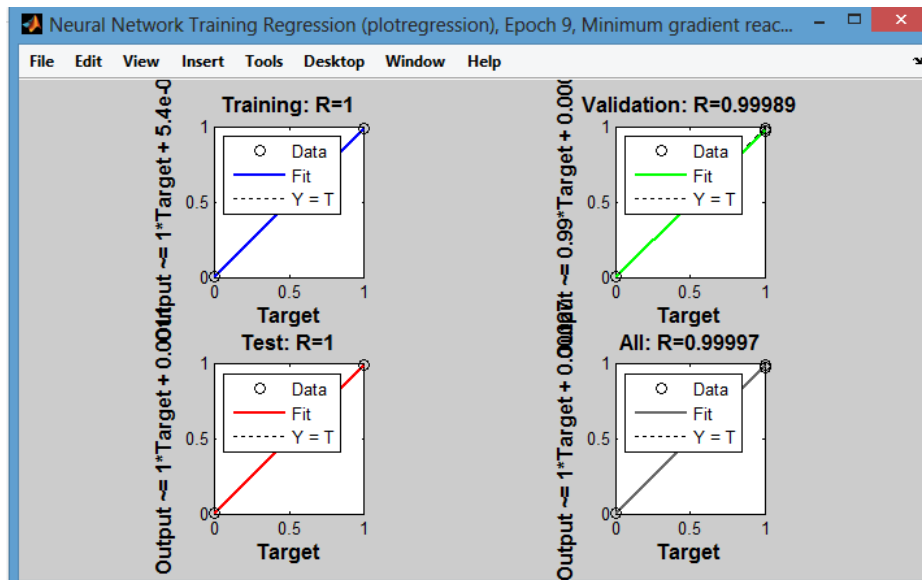


Figura 3.49 Regresión del error durante el entrenamiento de la red

Después de haber realizado el análisis de comportamiento durante el entrenamiento, la adaptación y la simulación de varias redes neuronales artificiales, se puede observar que es más efectiva para la clasificación de rosas la Feed-forward backpropagation con lo que se demuestra que la utilización de la herramienta nntool de Matlab es lo bastante poderosa para este tipo de trabajos.

## CAPÍTULO IV

### CONCLUSIONES Y RECOMENDACIONES

#### 4.1 CONCLUSIONES

Se cumplió el objetivo general de esta investigación de determinar las aplicaciones de las redes neuronales al control de brazos robóticos manipuladores utilizando redes neuronales, como una buena alternativa a la solución de problemas complejos multivariados no lineales como es el caso de la cinemática inversa de un brazo manipulador industrial.

Se cumplió el objetivo de evaluar el estado del arte de los sistemas de control basados en redes neuronales en robots manipuladores, para lo que se revisaron los trabajos presentados en investigaciones de los últimos años, llegando a la conclusión de que en estos 60 años de vida de las redes neuronales artificiales se ha avanzado bastante, sin embargo cada vez son más grandes los interrogantes que se van planteando y que el camino por recorrer es todavía muy extenso.

Se cumplió el objetivo de definir los parámetros funcionales de las redes neuronales que intervienen en los sistemas de control, especialmente de los sistemas de control aplicados a los robots manipuladores industriales que son de interés para la ingeniería mecánica, para lograr crecimiento y optimización de los procesos de la industria, lo que permitirá que cada vez se involucren más máquinas que realicen con efectividad tareas que demanden alta concentración y por periodos largos de personas, mejorando la calidad de vida de la población.

Se cumplió con el objetivo de definir las características relevantes de objetos o productos para ser utilizados como conjunto de entrenamiento, en este caso las características cuantitativas de los grupos de rosas de producción nacional, para la selección de la aplicación del presente trabajo a ser discriminados por un sistema automático de toma de decisiones en base a redes neuronales y su clasificación con robots manipuladores.

Se cumplió con el objetivo de esta propuesta, generar un documento que contenga la información necesaria para el manejo del toolbox de Matlab nntool, desde el punto de vista de las aplicaciones de las redes neuronales artificiales para el control de brazos robóticos industriales, para la clasificación de productos.

Se cumplió con el objetivo de modelar una red neuronal que utilizando un conjunto de entrenamiento seleccionado por un experto en rosas de exportación, que sea capaz de clasificar con un alto grado de precisión, esto es con error tendiente a cero, de acuerdo con las características que definen el tipo de producto que la empresa produce.

Los entornos productivos, comercial y científico en los que los sistemas informáticos recogen enormes cantidades de datos son ideales para la aplicación de las redes neuronales, esta técnica es accesible para muchos profesionales que no tienen un alto nivel de experiencia en computación, el software de redes neuronales se usa actualmente en muchas aplicaciones financieras, de la salud y clasificación de productos.

Las diferentes alternativas para seleccionar la red neuronal más adecuada que ofrece la herramienta nntool de Matlab en pocos intentos, la hace muy amigable, pues procesa la información en forma paralela y aprende mediante ejemplos presentados durante el entrenamiento, siendo adecuado para redes de aplicaciones donde se necesite procesar información compleja; las redes neuronales artificiales son capaces de extraer características esenciales de los ejemplos aprendidos, para procesar correctamente entradas incompletas o distorsionadas.

La capacidad de procesamiento de las red neuronales multicapa es adquirida durante el proceso de aprendizaje, el cual consiste en variar los valores de los pesos sinápticos de las neuronas de la red, de acuerdo con un algoritmo de entrenamiento, de tal forma que se obtengan las salidas deseadas para todos los ejemplos de entrada; de esta forma todo el conocimiento la red neuronal artificial multicapa se encuentra distribuido en los valores de los pesos sinápticos de sus neuronas.

La librería de redes neuronales de Matlab la Neural Network Toolbox provee diversas funciones para implementar la red neuronal que más se adapte a la aplicación que se desee realizar, como en el caso de la clasificación de productos agropecuarios, ahora que después del petróleo son productos de alto impacto en la matriz productiva de nuestro País, permitiendo definir el número de capas, el número de neuronas por capa, la función de transferencia o de activación en cada capa, el rango de valores de aprendizaje y la regla de aprendizaje.

Una de las grandes ventajas de las redes neuronales artificiales, es que aprenden la naturaleza paralela de las redes neuronales naturales, para reducir el tiempo requerido por un procesador secuencial, para determinar la salida adecuada a partir de una entrada.

El nntool de Matlab puede implementarse fácilmente y responde rápidamente a las entradas, su programación consiste en presentarle un grupo de ejemplos sin necesidad de desarrollar ningún algoritmo, la salida se genera por la propagación de las señales a través de sus capas, no necesitando comparar la entrada con los valores de una tabla para poder decidir cuál es la salida correcta.

Las dificultades en la adquisición de datos para el reconocimiento de objetos, es decir, la rotación, traslación, reflexión, escalonamiento, sesgado, oclusión, iluminación y presencia de múltiples objetos definen el alcance de la aplicación en desarrollo.

Las redes neuronales artificiales tienen un amplio uso en ingeniería pero a su vez quedan múltiples problemas abiertos siendo por ello un campo de investigación todavía muy intenso y extenso.

## **4.2 RECOMENDACIONES**

Las redes neuronales artificiales deben considerarse como una alternativa en la solución de aquellos problemas en los cuales no se han obtenido buenos resultados utilizando los métodos convencionales.

Se recomienda que el conjunto de datos de entrenamiento sean precisos y confiables por lo que deben ser seleccionados por un experto en la materia, para obtener resultados confiables, pues en industrias donde se producen gran

cantidad de datos, estos pueden ser incompletos o estar mal, por lo que la red a aplicarse debe ser lo suficientemente robusta para garantizar resultados exitosos.

Se recomienda estudiar la importancia real tanto teórica como práctica de la dualidad red neuronal-lógica difusa que puede ser mucho más eficiente que solo utilizar redes neuronales o lógica difusa por separado, para la solución del problema de reconocimiento y selección de objetos utilizando brazos robóticos articulados.

Se recomienda que para resolver el problema de reconocimiento y selección de objetos con redes neuronales artificiales, primero seleccionar una arquitectura de red neuronal apropiada, escoger un conjunto de ejemplos para entrenar la red, lo que es la fase de entrenamiento o aprendizaje de la red, luego probar la capacidad de la red entrenada para resolver un problema utilizando un conjunto de datos de prueba, esto es la fase de prueba de la red y evaluar la operación de la red con un conjunto de datos de verificación, que no hayan sido utilizados en el entrenamiento o las pruebas en la fase de generalización.

Para las aplicaciones de las redes neuronales en la selección de objetos por brazos robóticos articulados la mejor opción es utilizar redes neuronales de aprendizaje supervisado a pesar que tienen limitaciones siempre resultan la mejor opción, así como también se apliquen diversos métodos combinados de solución, por ejemplo las redes de la teoría de resonancia adaptativa o arquitecturas con sistemas adaptativos de inferencia neuro-fuzzy.

Si se selecciona el perceptrón multicapa para el reconocimiento y selección de objetos, los ejemplos utilizados para el entrenamiento deben seleccionarse adecuadamente para que la red adquiera la capacidad de generalización.

Para el reconocimiento de objetos se debe considerar ciertas dificultades inherentes al observador como rotación, traslación, reflexión, escalonamiento, sesgado, oclusión, iluminación y la presencia de múltiples objetos en una misma imagen, dependiendo de la aplicación para que se desee desarrollar un sistema de reconocimiento de imágenes, estas dificultades resultan de mayor o menor relevancia.

La adquisición y procesamiento de imágenes resulta una etapa que define la manera en que la red neuronal interactúa con el escenario bajo estudio, es muy importante la adecuación de los datos como entradas al sistema; procesos de binarización de imágenes en blanco y negro, de adecuación del número de píxeles que componen una imagen y el tipo de estructuras de datos a utilizar deben ser previamente considerados, el método regionprops de Matlab ayuda en la mejora del procesamiento de imágenes disminuyendo el impacto de las dificultades para el reconocimiento de objetos.

Para poder ejercer control sobre brazos robóticos computarizados industriales deben considerarse muchos factores, métodos y metodologías que integrados hacen inteligente a una máquina, el éxito está en lograr la combinación adecuada de esas técnicas que incluyen, sensorica, visión artificial, lógica difusa, redes neuronales, algoritmos genéticos, y los actuadores.

## BIBLIOGRAFIA

1. ACOSTA, M. (2000), Tutorial sobre redes neuronales aplicadas en Ingeniería eléctrica y su implementación en un sitio WEB. Tesis. UTP. Colombia.
2. AGUINAGA, A. (2014), Robótica, EPN. Quito.
3. AKASH, D. MISHRA, D. JHA, A. (2012), Optimización del tiempo de trabajo de un Robot Manipulador utilizando Redes Neuronales Artificiales y Algoritmos Genéticos. International Journal of Computer Applications.
4. BANDA, H. (2014), Inteligencia Artificial Principios y Aplicaciones. EPN..
5. BARRIENTOS, A. PEÑÍN, L. BALAGER C. ARACIL R. (1997). Fundamentos de Robótica. Ed. Mc Graw Hill. España
6. BONILLA, M.; (2005), Reconocimiento de caracteres mediante redes neuronales con MATLAB. Tesis. EPN.
7. BUSTAMANTE, E. (2007), El sistema nervioso, desde las neuronas hasta el cerebro humano, ed. Universidad de Antioquia. Antioquia.
8. CARRERA E. (1993), Reconocimiento de sonidos vocálicos empleando redes neuronales artificiales. Tesis. EPN.
9. CHANG, N. (2014), Diseño y simulación de un robot articular con seis grados de libertad utilizando el toolbox Robotics de Matlab para fortalecer las clases teóricas realizando prácticas de laboratorio con el software presentado en este proyecto. Tesis. EPN. Quito.
10. CORKE, P. (2014), Robotics TOOLBOX. Australia: CSIRO
11. CRAIG, J. (2008), Introduction to Robotics: Mechanics and Control, Ed. Pearson Education. USA.
12. DEMUTH, H. BEALE, M. HAGAN, M. (2009), Neural Network Toolbox™ 6 User's Guide, the Math Works, Massachusetts.
13. GUERRERO. F., (2013), Modelación, simulación y control de sistemas aéreos no tripulados utilizando inteligencia artificial. Tesis. EPN.
14. GRANJA, M. (2014), Comparación del método de Denavit-Hartenberg estándar con el modificado en la modelación de la robótica. Tesis. EPN.
15. HAYKIN, S. (2008), Neural Networks and Learning Machines, 3rd Ed. Prentice Hall. USA.



16. KOHONEN, T. (1987), *Self-Organization and Associative Memory*, 2nd Edition, Berlin: Springer-Verlag.
17. MARTÍNEZ, E. (2008), *Control de un robot utilizando un robot tipo PUMA Cedinet. Mx. Tesis.*
18. MARTINEZ, J. FERRANDEZ, J. (2012), *Diseño e implementación sobre hardware reconfigurable de una arquitectura para emulación en tiempo real de redes neuronales celulares*, Tesis, Universidad Politécnica de Cartagena.
19. MOYA, D. (2014), *Modelo y Análisis de un robot manipulador industrial esférico aplicando MATLAB*. Quito: Tesis EPN.
20. MULLER, B. REINHARDT J. (1991), *Neural Networks - An Introduction*. Physics of Neural Networks Series. Springer-Verlag, Germany.
21. OLLERO, A. (2001), *Robótica: manipuladores y robots móviles*; Marcombo, México.
22. REYES. L. TAPIA L. (2010), *Diseño e implementación de un programa basado en redes neuronales para el control de máquinas trifásicas de inducción*; Tesis. EPN.
23. RODRÍGUEZ. A.; (2006), *Técnicas para la visualización del habla*. Tesis. EPN.
24. ROSENBLATT, F. (1962), *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*. Spartan Books, USA.
25. SAHA S. K. (2008), *Introduction to Robotics* Tata, McGraw-Hill Education, Nueva Delhi.
26. SANCHEZ, A. (2014), *Robótica Visión Artificial*. EPN.
27. SCHIFFMAN W. GEFFERS H., (1993), *Adaptive control of dynamic systems by backpropagation networks*. Neural Networks 6.
28. GONZÁLEZ, S. LEON J. ESTRADA Y. (2011), *Sistema para el control inteligente de un robot manipulador*. Universidad Agraria de La Habana.
29. <http://www.chpg.mc/fr/centre-hospitalier/presentation-generale/une-haute-technologie-a-votre-service>
30. <http://www.palisade-lta.com/NeuralTools/5/tips/es/gs/4.asp>

31. [http://scielo.sld.cu/scielo.php?pid=S2071-00542011000200005&script=sci\\_arttext](http://scielo.sld.cu/scielo.php?pid=S2071-00542011000200005&script=sci_arttext)
32. [http://www.cpdee.ufmg.br/~apbraga/MeusArtigos/Journals/12\\_RoselitoNPL.pdf](http://www.cpdee.ufmg.br/~apbraga/MeusArtigos/Journals/12_RoselitoNPL.pdf)
33. <http://www.mathworks.com/products/neuralnet>
34. <http://www.petercorke.com/RTB/robot.pdf>
35. <http://www.redes-neuronales.com.es/tutorial-redes-neuronales/Las-redes-neuronales-monoCapa.htm>
36. <http://www.globalrobots.com/default.aspx>
37. <http://www.fayerwayer.com/2014/06/museo-de-tokio-incorpora-androides-a-su-personal/>
38. <http://www.ibtimes.co.uk/human-machine-life-like-android-robots-japan-show-glimpses-future-1453992>
39. <http://medicineworld.org/stories/lead/3-2011/activity-of-single-neurons-during-seizures.html>
40. [http://www.limoncellodigital.com/2013\\_03\\_01\\_archive.html](http://www.limoncellodigital.com/2013_03_01_archive.html)

