

Título completo de la tesis doctoral de la UDC

Modelos de procesamiento de la información en el cerebro aplicados a sistemas conexionistas. Redes NeuroGliales Artificiales y Deep Learning.

Autor/a: Lucas Antón Pastur Romay

Tesis doctoral UDC / 2018

Directores:

Dra. Ana Belén Porto Pazos

Dr. Juan Pazos Sierra

Tutor/a: Ana Belén Porto Pazos

Programa de doctorado en Tecnologías de la Información y las Comunicaciones



UNIVERSIDADE DA CORUÑA

A mi familia, mi pareja y mis amigos

Agradecimientos

Me gustaría mostrar mi agradecimiento a todas las personas que han hecho posible el trabajo desarrollado en esta tesis. En primer lugar, a mi tutora y directora de tesis, Ana Porto, que me ha servido de guía en este maravilloso campo de la Inteligencia Artificial. Me has ayudado y enseñado mucho desde que comencé a trabajar con las neuronas y astrocitos artificiales, y gracias a ti ha sido posible realizar esta tesis. También quisiera agradecer a mi director de tesis, Juan Pazos, por su ayuda y sabios consejos.

Quiero agradecer a Alejandro Pazos, por todo el apoyo que me ha dado y ayudarme a comenzar mi carrera profesional en el campo de la investigación.

Por supuesto, quiero agradecer a todos mis compañeros del laboratorio RNASA-IMEDIR por la acogida que tuve desde el primer día. Fran, por todo el tiempo que me has dedicado y la ayuda que me has prestado, este trabajo no podría haber sido posible sin tu colaboración. Candela por todo lo que me has ayudado y enseñado desde mi primer día en el laboratorio. A mi compañero Cristian, por todo lo que me has enseñado y las conversaciones que hemos tenido, espero poder seguir aprendiendo del auténtico experto en Deep Learning. Y también quiero agradecer al resto de compañeros del laboratorio: Quique, Carlos, Adrián, Marcos, Julián, Nieves, Dani, Iván, Jose.

A mis amigos de la banda del patio por todo lo que hemos vivido y lo que nos queda por vivir: Mani, Dan, Gely, Perry, Jimmy, Jeans, Floty y Cojo. También a mis amigos del erasmus: Sergio, Pichu, Antonio, Andrea, Cordobelo y Manolo.

Y sobre todo gracias a mis padres, Isabel y Manolo, por todo lo que me habéis enseñado y el ejemplo que siempre habéis sido para mí. El cariño y la educación que me habéis dado me ha hecho como soy. A mis hermanos, Mateo y Manuel, por vuestro apoyo y cariño, y por servirme de ejemplo. Gracias a Malcom y Mirka por vuestra compañía y sabios consejos. Luz gracias por coincidir, me empujas a querer ser cada día mejor y seguir avanzando este camino juntos, viviendo una aventura constante.

Por último, agradecer al grupo RNASA-IMEDIR (Proyecto “GERIA-TIC”, Proyecto cofinanciado por GAIN- Conecta PEME (3ª edi) (IN852A 2016/10) y fondos FEDER UE, “Proyecto Colaborativo de Integración de datos Genómicos” (CICLOGEN) PI17/01826, “Técnicas de data mining y docking molecular para análisis de datos integrativos en cáncer de colon” del MINECO. “Red Gallega de Investigación en Cáncer Colorrectal” (REGICC) ED431D2017/23, “Red Gallega de medicamentos” (REGID) ED431D2017/16, Ayudas para la Consolidación y estructuración de unidades de investigación competitivas del SUGA-Xunta de Galicia y Centros Singulares (ED431G/01) dotados de fondos FEDER de la UE), la UDC y al CESGA por todos los recursos que me han permitido realizar esta tesis.

Resumen

En el campo de la Inteligencia Artificial, los sistemas conexionistas se han inspirado en las neuronas ya que, según la visión clásica de la Neurociencia, eran las únicas células con capacidad para procesar la información. Descubrimientos recientes de Neurociencia han demostrado que las células gliales tienen un papel clave en el procesamiento de la información en el cerebro. Basándose en estos descubrimientos se han desarrollado las Redes NeuroGliales Artificiales (RNGA) que cuentan con dos tipos de elementos de procesado, neuronas y astrocitos. En esta tesis se ha continuado con esta línea de investigación multidisciplinar que combina la Neurociencia y la Inteligencia Artificial. Para ello, se ha desarrollado un nuevo comportamiento de los astrocitos que actúan sobre la salida de las neuronas en las RNGA. Se ha realizado una comparación con las Redes de Neuronas Artificiales (RNA) en cinco problemas de clasificación y se ha demostrado que el nuevo comportamiento de los astrocitos mejora de manera significativa los resultados.

Tras demostrar la capacidad de los astrocitos para procesar la información, en esta tesis se ha desarrollado además una nueva metodología que permite por primera vez la creación de redes Deep Learning conteniendo miles de neuronas y astrocitos, denominadas *Deep Neuron-Astrocyte Networks* (DANAN). Tras probarlas en un problema de regresión, las DANAN obtienen mejores resultados que las RNA. Esto permitirá evaluar comportamientos más complejos de los astrocitos en las redes de *Deep Learning*, pudiendo incluso crearse redes de astrocitos en un futuro próximo.

Resumo

No campo da Intelixencia Artificial, os sistemas conexionistas inspiráronse nas neuronas xa que, segundo a visión clásica da Neurociencia, eran as únicas células con capacidade para procesar a información. Descubrimentos recentes de Neurociencia demostraron que as células gliais teñen un papel cruce no procesamento da información no cerebro. Baseándose nestes descubrimentos desenvóléronse as Redes NeuroGliales Artificiais (RNGA) que contan con dous tipos de elementos de procesado, neuronas e astrocitos. Nesta tese continuouse con esta liña de investigación multidisciplinar que combina a Neurociencia e a Intelixencia Artificial. Para iso, desenvolveuse un novo comportamento dos astrocitos que actúan sobre a saída das neuronas nas RNGA. Realizouse unha comparación coas Redes de Neuronas Artificiais (RNA) en cinco problemas de clasificación e demostrouse que o novo comportamento dos astrocitos mellora de xeito significativo os resultados.

Tras demostrar a capacidade dos astrocitos para procesar a información, nesta tese desenvolveuse ademais unha nova metodoloxía que permite por primeira vez a creación de redes *Deep Learning* contendo miles de neuronas e astrocitos, denominadas *Deep Neuron-Astrocyte Networks* (DANAN). Tras probalas nun problema de regresión, as DANAN obteñen mellores resultados cas RNA. Isto permitirá avaliar comportamentos máis complexos dos astrocitos nas redes de *Deep Learning*, podendo ata crearse redes de astrocitos nun futuro próximo.

Abstract

In the field of Artificial Intelligence, connectionist systems have been inspired by neurons and, according to the classical view of neuroscience, they were the only cells capable of processing information. The latest advances in Neuroscience have shown that glial cells have a key role in the processing of information in the brain. Based on these discoveries, Artificial NeuroGlial Networks (RNGA) have been developed, which have two types of processing elements, neurons and astrocytes. In this thesis, this line of multidisciplinary research that combines Neuroscience and Artificial Intelligence has been continued. For this goal, a new behavior of the astrocytes that act on the output of the neurons in the RNGA has been developed. A comparison has been made with the Artificial Neuron Networks (ANN) in five classification problems and it has been demonstrated that the new behavior of the astrocytes significantly improves the results.

After prove the capacity of astrocytes for information processing, in this thesis has been developed a new methodology that allows for the first time the creation of Deep Learning networks containing thousands of neurons and astrocytes, called Deep Neuron-Astrocyte Networks (DANAN). After testing them in a regression problem, the DANAN obtain better results than ANN. This allows testing more complexes astrocyte behaviors in Deep Learning networks, and even creates astrocyte networks in the near future.

1. Introducción

1.1. Cerebro e Inteligencia Artificial

El cerebro humano es el sistema más complejo del universo conocido y se considera el culmen de la evolución biológica en la Tierra. En menos de 1,5 Kg están almacenados todos nuestros recuerdos, conocimientos y habilidades. También nos permite interactuar con el medio y con nuestros iguales de maneras muy complejas. Además, la plasticidad cerebral nos facilita el aprendizaje de nuevo conocimiento y habilidades. Por todas estas razones, el estudio del cerebro humano y la mente humana ha centrado el esfuerzo de numerosos pensadores, filósofos y científicos a lo largo de la historia. Durante mucho tiempo se consideró que la mente humana no residía en el cerebro, ya que tenía un origen divino. El punto de inflexión en el estudio del cerebro se produjo a principios del siglo XX con los estudios de Ramón y Cajal y su teoría Neuronal (Ramon y Cajal, 1904). Se puede considerar que sus descubrimientos dieron lugar al nacimiento de la Neurociencia moderna, la idea de que el cerebro estaba compuesto por células individuales marcó un antes y un después en este campo.

El cerebro tiene multitud de capacidades, algunas de las más destacadas son el aprendizaje, la memoria, la imaginación o la creatividad. Estas capacidades están bastante definidas, y se engloban en un concepto más global, la inteligencia, el cual resulta más complicado de definir. El término “Inteligencia” proviene del latín *Intelligentia* que a su vez deriva de *Inteligere*, palabra compuesta de dos términos: *intus* (“entre”) y *legere* (“escoger”). Por lo que etimológicamente, la inteligencia significa “la capacidad de escoger entre varias posibilidades”. Una definición más formal de la inteligencia es: “capacidad de adquirir y aplicar conocimiento y habilidades”. Estas definiciones son elusivas, y no permiten establecer límites claros entre lo que es o no es la inteligencia, y que seres vivos podemos considerar inteligentes.

A mediados del siglo XX se establece el campo de la Inteligencia Artificial (IA) que trata de crear sistemas computacionales inteligentes (Turing, 1950). La IA se divide en IA fuerte, científica, o *Artificial General Intelligence* (AGI), y IA débil, tecnológica, o *narrow*. La AGI trata de crear sistemas que sean capaces de realizar cualquier tarea intelectual que un ser humano es capaz de realizar. Es decir, que un único sistema sea capaz de igualar, o mejorar, las capacidades intelectuales de los humanos. Se considera que para que un sistema de IA pueda igualar las capacidades intelectuales de un ser humano, sería necesario que adquiriera

algún tipo de conciencia propia que le permitiese interactuar de manera exitosa con el mundo y formarse una representación interna del propio sistema. Por otra parte, la IA tecnológica se centra en crear sistemas que realicen una única tarea de manera exitosa, y hoy en día la mayoría de los trabajos en IA se centran en este tipo de sistemas. A su vez la IA tecnológica se divide en dos ramas principales, los sistemas expertos y el *Machine Learning* (ML), o aprendizaje máquina. El ML se define como:

“*un sistema organizado puede definirse como aquel que transforma un cierto mensaje de entrada en uno de salida, de acuerdo con algún principio de transformación. Si tal principio está sujeto a cierto criterio de validez de funcionamiento, y si el método de transformación se ajusta a fin de que tienda a mejorar el funcionamiento del sistema de acuerdo con ese criterio, se dice que el sistema aprende*” (Wiener, 1975).

Los sistemas expertos tratan de crear comportamientos inteligentes en base al conocimiento de un experto en el campo que ayuda a diseñar un programa, o conjunto de reglas, para realizar una determinada tarea (Lederberg et al., 1970; Michalski et al., 2013). Por su parte, el ML clásico utiliza un conjunto de variables seleccionadas manualmente para entrenar un modelo que no hay sido programado explícitamente para esa tarea. Dentro del ML existe un campo que está inspirado en el cerebro y utiliza la neurona como elemento básico de procesado de la información. El nombre con el que se denomina este campo ha evolucionado a lo largo del tiempo, en sus orígenes se denominó cibernética, más tarde conexionismo, y en los últimos tiempos se denominan Redes de Neuronas Artificiales (RNA) (I. Goodfellow et al., 2016; Michalski et al., 2013). Este tipo de modelo emula el funcionamiento del cerebro, de manera que trata de extraer las características de manera automática a partir de los ejemplos que se le muestran en la fase de entrenamiento. Es por ello que a este campo también se le denomina *Representation Learning*, o aprendizaje de representación, ya que después de entrenar los modelos, estos representan las características de los datos en su estado interno. Dentro de los modelos de *Representation Learning* existe una división en función del tamaño, o profundidad de la red, los modelos *shallow*, o poco profundos, y los modelos de *Deep Learning*, o de aprendizaje profundo, los cuales son capaces de representar de forma jerárquica las características, al igual que hace el cerebro (I. Goodfellow et al., 2016). En la figura 1 se puede observar un diagrama de Venn con ejemplos dentro de algunas áreas de la IA.

En el trabajo de 1948 de Alan Turing, publicado en 1968 (Turing, 1948), se proponen dos tipos de modelos computacionales, denominadas máquinas desorganizadas, o *unorganized machines*, que se inspiran en el funcionamiento del cerebro humano y establecen la base teórica de las RNA. Las *unorganized machines* se dividen en: tipo A y tipo B. Las *unorganized*

machines tipo A son las más sencillas y tienen una arquitectura que una vez inicializada, no puede modificarse. Este tipo de máquina se conoce como red booleana aleatoria, o red recurrente binaria aleatoria. Las *unorganized machines* tipo B constan de neuronas artificiales y dispositivos que modifican las conexiones entre ellas. Estas máquinas pueden verse como máquina tipo A en las que las conexiones son reemplazadas por una máquina tipo A. Las *unorganized machines* tipo B son la base de las actuales RNA y de la computación evolutiva.

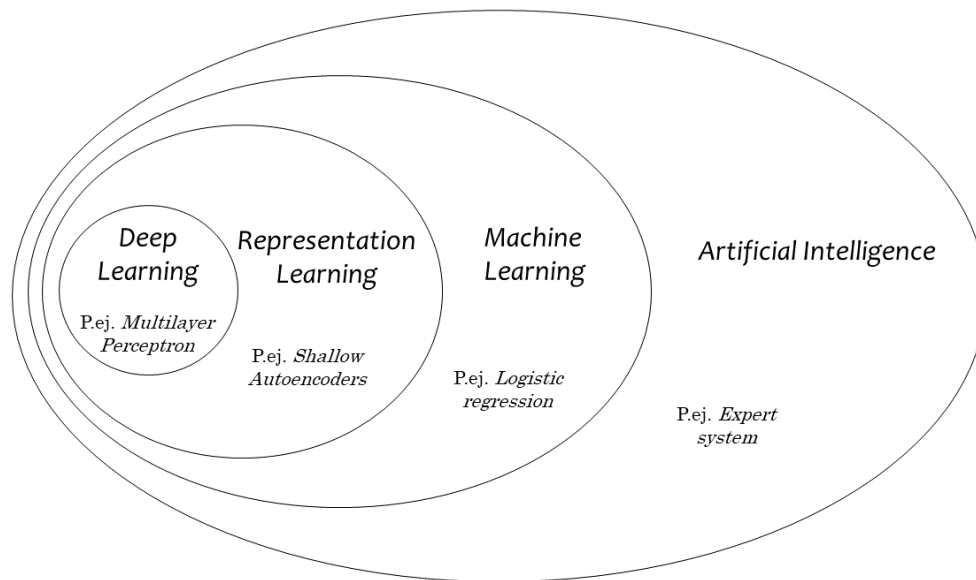


Figura 1. Diagrama de Venn en el que se muestra que el Deep Learning es un tipo de aprendizaje por representación, que a la vez es un tipo de aprendizaje máquina, el cual es una de las aproximaciones de la Inteligencia Artificial. En cada sección del diagrama de Venn se incluye un ejemplo de una tecnología de Inteligencia Artificial. Figura adaptada del libro Deep Learning (I. Goodfellow et al., 2016).

Para desarrollar modelos de IA un aspecto importante a tener en cuenta es el funcionamiento del cerebro, ya que es la única prueba que tenemos de la existencia de la inteligencia. Por tanto, estudiar la cognición de los animales juega un rol vital a la hora de desarrollar sistemas inteligentes artificiales. Este enfoque multidisciplinar presenta beneficios tanto para la IA como para la Neurociencia. En primer lugar, la Neurociencia sirve como fuente de inspiración para desarrollar nuevos algoritmos y arquitecturas, los cuales pueden ser independientes o complementarios a los métodos lógicos y matemáticos empleados hasta el momento en el campo de la IA (Hassabis et al., 2017). Durante mucho tiempo esta fue la única dirección en el flujo de conocimiento entre las dos disciplinas. Pero la IA también puede beneficiar a la Neurociencia, no sólo por el uso de las técnicas de procesamiento de datos. Los algoritmos desarrollados y empleados en la IA pueden validarse en estudios de Neurociencia y servir como inspiración para realizar nuevos experimentos en el

laboratorio. Por ejemplo, si un algoritmo empleado en IA, se descubre que también está implementado en el cerebro, sería una prueba sólida de que puede tener un papel importante en la inteligencia.

La IA, en especial las RNA, ha recorrido un largo camino desde sus orígenes, superando diversos obstáculos hasta alcanzar los resultados actuales. La última década ha sido la más fructífera en el campo de la IA gracias a los enormes avances en las RNA, en especial en *Deep Learning*. Hoy en día es uno de los campos más activos tanto en la investigación como en el desarrollo de aplicaciones. La posibilidad de creación de aplicaciones muy rentables ha provocado que las grandes compañías tecnológicas como Google, Microsoft, IBM, Apple, Facebook, Baidu, Adobe, Netflix o NVIDIA, y multitud de start-ups hayan invertido gran cantidad de recursos en *Deep Learning*. Se puede explicar el éxito actual entre otros por los siguientes motivos:

- Desarrollo exponencial de la capacidad de computación, especialmente el desarrollo de *Graphics Processing Unit* (GPU) diseñadas para entrenar RNA.
- Aumento del tamaño y calidad de las bases de datos. En los últimos años ha aumentado la cantidad de información que se almacenan en las bases de datos y la calidad de las mismas.
- Mejora de los algoritmos de entrenamiento y optimizadores (SGD (Bottou, 2010), ADADELTA (Zeiler, 2012)), funciones de activación (ReLU) (Nair & Hinton, 2010), técnicas de regularización (*Dropout* (Srivastava et al., 2014), *Batch normalization* (Ioffe & Szegedy, 2015)), mejora de las arquitecturas (AlexNet (Krizhevsky et al., 2012), VGG Net (Simonyan & Zisserman, 2014), GoogLeNet (Szegedy et al., 2015), ResNet (He et al., 2016) Generative Adversarial Nets (I. Goodfellow et al., 2014)).
- Desarrollo de plataformas abiertas para el desarrollo de modelos de Deep Learning, gracias en parte al interés de las grandes compañías por desarrollar la plataforma principal para el desarrollo de aplicaciones. Algunas de las plataformas más destacadas son: TensorFlow (Abadi et al., 2016), Theano (Bergstra et al., 2010), Caffe (Jia, 2013), Keras (Chollet, 2015), PyLearn2 (I. J. Goodfellow et al., 2013).

Todos estos avances han permitido desarrollar modelos de gran tamaño, desde la aparición de las redes multicapa, su tamaño se ha duplicado cada dos años y medio. Esto ha permitido el desarrollo de multitud de aplicaciones, y han convertido a las RNA en el algoritmo “estado-del-arte” para muchas tareas, superando incluso los resultados obtenidos por los humanos, por ejemplo en tareas de reconocimiento de audio (G. Hinton et al.,

2012) e imágenes (Farabet et al., 2013; Krizhevsky et al., 2012), tareas de generación de imágenes o texto manuscrito indistinguible de las imágenes reales o el texto escrito por humanos (Lake et al., 2015), traduciendo texto entre múltiples lenguas (Wu et al., 2016), etc. Además de superar a los humanos en tareas relativamente sencillas de reconocimiento de patrones, las RNA han obtenido resultados extraordinarios en campos más sorprendentes como el diseño de nuevo medicamentos (Dahl et al., 2014), diagnóstico de enfermedades como el melanoma (Codella et al., 2015), reconstrucción de mapas 3-D a partir de imágenes de microscopio (Kasthuri et al., 2015), detección de partículas subatómicas como el Bosón de Higgs (Baldi et al., 2014), generación de voces humanas (Oord et al., 2016), creación de imágenes con estilos artísticos (Gatys et al., 2016).

La IA y la Neurociencia han tenido una historia larga y entrelazada, a pesar de los grandes resultados obtenidos por las RNA, aún existen muchos aspectos del cerebro que pueden servir como inspiración para desarrollar nuevas técnicas y algoritmos de IA. Algunas de las áreas en las que la inspiración en la Neurociencia puede ser más útil para la IA son: autoorganización de la arquitectura, eficiencia energética, comprensión intuitiva del mundo físico, algoritmos de aprendizaje, aprendizaje eficiente, transferencia del aprendizaje, imaginación y planificación, análisis virtual del cerebro (Brooks et al., 2012; Hassabis et al., 2017). También existen grandes diferencias en cómo se consiguen desarrollar los comportamientos inteligentes en las máquinas, y en los de los seres humanos y animales en general. Para desarrollar los sistemas de IA es necesario un experto que diseñe la arquitectura, los algoritmos y el sistema de evaluación del sistema para cada tarea, además el consumo energético del cerebro es varias órdenes de magnitud inferior al de los computadores. Por ello es importante que la IA se inspire en el cerebro, pero sin tener que ser fiel a las limitaciones de crear sistemas biológicamente plausibles. Desde el punto de vista de la ingeniería informática, es necesario que los sistemas implementados funcionen correctamente, por lo que la Biología sirve como guía, pero no se trata de un requerimiento estricto. Esto hace que no sea necesario estudiar el cerebro desde todos los niveles de análisis. Tomando como ejemplo los niveles establecidos por Marr y Poggio (Marr & Poggio, 1976) para el análisis de los sistemas biológicos: nivel de implementación, nivel algorítmico y nivel computacional, solo sería necesario centrarse los dos últimos niveles ya que no es relevante el nivel de implementación que estudia los mecanismos físicos que ocurren en el sustrato biológico. Esta propuesta es defendida por alguna iniciativa importante que ha obtenido grandes resultados, como es el caso de Deep Mind y su fundador Demis Hassabis (Brooks et al., 2012; Hassabis et al., 2017). Sin embargo, otras

iniciativas como el Human Brain Project (Markram, 2012) tratan de simular el cerebro de manera biológicamente realista para así desarrollar sistemas artificiales inteligentes. En el campo de la IA existe el extremo contrario, en el que se afirma que no es necesario entender el funcionamiento del cerebro para crear sistemas inteligentes, y se utiliza el símil de que para construir aviones que pudiesen volar no fue necesario estos moviesen sus alas como los pájaros. En el trabajo presentado en esta tesis doctoral se toma el punto intermedio, es decir, se utiliza la Neurociencia como fuente de inspiración para el desarrollo de nuevos algoritmos de IA, pero sin necesidad de que éstos sean biológicamente realistas. En concreto, desde el punto de vista de la IA, uno de los aspectos más relevantes que se ha descubierto en las últimas dos décadas en el campo de la Neurociencia, ha sido el papel que juegan las células gliales en el procesamiento de la información. La doctrina neuronal otorga a las neuronas el protagonismo único en la transmisión y procesamiento de la información a través de sinapsis. Según esta teoría, las funciones asociadas a las células gliales eran de soporte trófico y metabólico de las neuronas (Ramon y Cajal, 1904; Ramón y Cajal, 1952). En las últimas décadas se ha demostrado que las células gliales, especialmente los astrocitos, participan en la transmisión y procesamiento de la información, formando parte de la sinapsis, en lo que se conoce como sinapsis tripartita (Alfonso Araque et al., 1999), compuesta por una neurona presináptica, una neurona postsináptica y un astrocito perisináptico. Además, los astrocitos pueden comunicarse entre ellos mediante ondas de calcio (A. H. Cornell-Bell et al., 1990).

En Neurociencia Computacional existen grupos que tratan de simular el funcionamiento de los astrocitos y su rol en el procesamiento de la información. Hay modelos de astrocitos individuales, de la sinapsis tripartita y de la comunicación entre astrocitos mediante ondas de calcio. Pero estos modelos tratan de ser biológicamente realistas por lo que tienen en cuenta detalles moleculares y no se centran en cómo se procesa la información (Oschmann et al., 2017).

Por otra parte, en el campo de la IA apenas hay trabajos que incorporen astrocitos artificiales a las RNA clásicas. El grupo RNASA-IMEDIR ha sido pionero en este campo, en el año 2004 Ana Porto desarrolló el primer modelo de IA que incorporaba dos tipos de elementos de procesado, las neuronas artificiales y las células gliales artificiales, para formar las denominadas Redes NeuroGliales Artificiales (RNGA) (Ana Belen Porto-Pazos et al., 2005). Estas redes utilizan un entrenamiento híbrido, existe una parte supervisada mediante algoritmos genéticos y una parte no supervisada en la que actúa el algoritmo glial, el cual modifica los pesos de las conexiones salientes de las neuronas en función de su actividad.

Las RNGA se pueden considerar un tipo híbrido de las *unorganized machines* propuestas por Alan Turing, ya que además de contar con neuronas artificiales como elementos de procesado, se incluyen astrocitos artificiales que controlan la actividad de las neuronas artificiales. Las RNGA fueron evaluadas en varios problemas de clasificación y sus resultados fueron comparados con los obtenidos por las RNA, mejorando significativamente los mismos (A. Porto et al., 2007). A partir de esta prueba de concepto del papel de la glía en el procesamiento de la información, se ha continuado con esta línea de investigación en IA inspirada en los descubrimientos de Neurociencia sobre el papel de las células gliales, especialmente los astrocitos, en el procesamiento de la información (Ana B Porto-Pazos et al., 2011b). Para ello, se han creado varios algoritmos gliales inspirados en el concepto de la sinapsis tripartita para investigar el papel de las células gliales en el procesamiento de la información, centrándose en las funciones de los astrocitos en el neocórtex (Alvarellos-González et al., 2012b).

1.2. Objetivos Generales

El objetivo general de esta tesis doctoral es desarrollar nuevos modelos de IA basándose en el papel de los astrocitos del sistema glial en el procesamiento de la información y su influencia en la inteligencia. Para ello el trabajo se basará tanto en las RNGA como en las nuevas técnicas de *Deep Learning* para la implementación y el desarrollo de nuevos algoritmos que representan las funciones de los astrocitos observadas en los experimentos de Neurociencia.

1.3. Objetivos Específicos

- Determinar la eficacia en los sistemas conexionistas de una modulación artificial astrocítica de la salida de las neuronas y compararla con las RNGA implementadas hasta el momento, las cuales actuaban sobre los pesos de las conexiones. Evaluar la eficacia del nuevo algoritmo a la hora de resolver problemas de clasificación y predicción.
- Realizar un estudio comparativo entre las nuevas Redes NeuroGliales Artificiales y las Redes de Neuronas Artificiales.
- Estudio de las diferentes arquitecturas y técnicas de *Deep Learning* para diseñar la metodología de trabajo para implementar los astrocitos artificiales en redes entrenadas con el algoritmo de retropropagación.

- Creación de redes de Deep Learning que incorporen neuronas y astrocitos artificiales, denominadas *Deep Artificial Neuron-Astrocyte Networks* (DANAN). Para ello se diseñará un algoritmo astrocítico que actúe en las redes entrenadas con retropropagación.
- Estudio comparativo de la DANAN y la red de *Deep Learning* en un problema de regresión.

1.4. Hipótesis

Los descubrimientos en el campo de Neurociencia en la última década muestran el potencial que tienen las células gliales para participar en el procesamiento de la información. Se ha comprobado que existen diversos modos de actuación de los astrocitos en el procesamiento de la información, y aún no se sabe con exactitud cómo son las interacciones entre neuronas y astrocitos, ni en qué situaciones tienen lugar, o a través de qué mecanismos.

Los resultados previos de las RNGA son prometedores, ya que se han desarrollado varios algoritmos gliales que mejoran los resultados de las RNA. Esto representa una prueba de concepto de las capacidades, y abren una nueva línea de investigación en el campo de IA. Pero en la actualidad los mejores resultados se obtienen con RNA entrenadas con algoritmos de retropropagación y técnicas de *Deep Learning*, por lo que es necesario implementar nuevos algoritmos gliales compatibles con este tipo de entrenamiento.

La hipótesis de esta tesis doctoral es: “los astrocitos participan en el procesamiento de la información por medio de la sinapsis tripartita, por tanto, se pueden desarrollar algoritmos de IA basados en el concepto de la sinapsis tripartita que pueden mejorar los resultados.”

Esta hipótesis se probó implementando diferentes algoritmos en varios tipos de redes. En primer lugar, se probó con redes sencillas, las denominadas RNGA, entrenadas con algoritmos genéticos en 5 problemas de clasificación. Después de probar la eficacia con problemas sencillos, se realizó una prueba de concepto con redes de mayor tamaño, con miles de neuronas, y entrenadas con el algoritmo de retropropagación para un problema más complejo de regresión. Estas redes se denominan DANAN por estar inspiradas en un tipo concreto de células gliales, los astrocitos, que son característicos del neocórtex, la zona del cerebro.

1.5. Estructura de la Tesis

Esta tesis doctoral se compone de 8 capítulos los cuales se dividen en varios apartados,

- **Capítulo 1: Introducción.** Se trata de la presente Introducción, en la cual se contextualiza el trabajo desarrollado, los objetivos y la hipótesis de esta tesis doctoral.
- **Capítulo 2: Fundamentos Biológicos de Neurociencia.** Descripción de los fundamentos biológicos y neurocientíficos; tipos de células en el sistema nervioso; y organización general. Análisis del cerebro como computador y teorías al respecto.
- **Capítulo 3: Fundamentos Computacionales de la Inteligencia Artificial.** Se resume el desarrollo histórico del estudio del cerebro como computador tanto desde el punto de vista de la Neurociencia Computacional como desde el de la Inteligencia Artificial. Se presentan los diferentes modelos computacionales basados en el cerebro, desde los más realistas hasta los más abstractos, centrándose en el funcionamiento de las neuronas artificiales y las RNA. También, se presenta el concepto de las Redes NeuroGliales Artificiales y se explica su funcionamiento.
- **Capítulo 4: Redes NeuroGliales Artificiales.** En este capítulo se detalla la experimentación llevada a cabo con las RNGA y el nuevo algoritmo glial desarrollado. Se presenta la metodología empleada, las herramientas utilizadas en la experimentación y los problemas seleccionados para evaluar los modelos. Además, se muestran los resultados obtenidos y se analiza la comparación con las RNA y los otros algoritmos de las RNGA.
- **Capítulo 5: *Deep Artificial Neuron-Astrocyte Networks*.** Se presenta la experimentación realizada con las RNA de Deep Learning, la metodología empleada y el problema utilizado. Se detalla el nuevo algoritmo astrocítico implementado en las denominadas DANAN.
- **Capítulo 6: Conclusiones.** Se presentan las conclusiones obtenidas de la investigación realizada en esta tesis doctoral.
- **Capítulo 8: Futuros desarrollos.** Finalmente se presentan los futuros desarrollos que se pueden derivar de los avances realizados en esta investigación.

Índice

Agradecimientos	5
Resumen.....	7
Resumo.....	9
Abstract	11
1. Introducción	13
1.1. Cerebro e Inteligencia Artificial.....	13
1.2. Objetivos Generales	19
1.3. Objetivos Específicos.....	19
1.4. Hipótesis.....	20
1.5. Estructura de la Tesis	20
Lista de figuras	27
Lista de tablas.....	29
Lista de gráficas.....	31
2. Fundamentos Biológicos	33
2.1. Sistema Nervioso.....	33
2.2. Procesamiento de la Información en el Cerebro	36
2.3. Astrocitos	38
2.4. Sinapsis Tripartita.....	40
3. Fundamentos Computacionales.....	45
3.1.1. Desarrollo Histórico.....	46
3.1.2. Modelos Neuronales Computacionales	59
3.1.2.1. Modelos Digitales Software	61
3.1.2.2. Modelos Digitales Hardware	62
3.1.2.3. Modelos Analógicos	63
3.1.2.4. Modelos Híbridos	64
3.1.3. Redes Neuronales Artificiales.....	64
4. Antecedentes y Motivación	67
4.1. Modelos gliales.....	68
4.1.1. Neurociencia Computacional	68
4.1.2. Modelos gliales en Chips Neuromórficos.....	69
4.1.3. Modelos gliales en Inteligencia Artificial.....	70
4.2. Motivación	72
5. Redes NeuroGliales Artificiales	75

5.1.	Introducción	75
5.1.1.	Fase de Diseño	77
5.1.2.	Fase de Entrenamiento	77
5.1.2.1.	Fase de Aprendizaje no Supervisado: Algoritmo Glial	77
5.1.2.2.	Fase de Aprendizaje Supervisado: Algoritmo Genético	80
5.1.3.	Fase de Validación.....	80
5.1.4.	Fase de Test.....	81
5.2.	Antecedentes Redes NeuroGliales Artificiales.....	81
5.3.	Metodología.....	84
5.3.1.	Problemas a Resolver.....	85
5.3.1.1.	Diagnóstico del Cáncer de Mama.....	85
5.3.1.2.	Señales de Ionosfera	86
5.3.1.3.	Señales de Sonar	86
5.3.1.4.	Diagnóstico de Diabetes.....	86
5.3.1.5.	Setas Venenosas.....	87
5.3.2.	Preparación de los Datos	87
5.3.3.	Diseño y Construcción de la Red	88
5.3.4.	Fase de Entrenamiento, Validación y Test	90
5.3.4.1.	Algoritmos Genéticos	90
5.3.4.2.	Algoritmo Glial con modificación de la salida	91
5.3.5.	Análisis y Organización de los Resultados.....	93
5.3.6.	Validación del Sistema	94
5.4.	Resultados	96
5.4.1.	Resultados Depresión.....	98
5.4.1.1.	Diagnóstico del Cáncer de Mama.....	98
5.4.1.2.	Señales de Ionosfera	99
5.4.1.3.	Señales de Sonar	100
5.4.1.4.	Diagnóstico de Diabetes.....	101
5.4.1.5.	Setas Venenosas.....	102
5.4.2.	Resultados Potenciación	103
5.4.2.1.	Diagnóstico del Cáncer de Mama.....	103
5.4.2.2.	Señales de Ionosfera	104
5.4.2.3.	Señales de Sonar	105
5.4.2.4.	Diagnóstico de Diabetes.....	106
5.4.2.5.	Setas Venenosas.....	107
5.4.3.	Resumen Resultados	108

5.4.4.	Comparación con Algoritmo de Conexiones	108
5.5.	Discusión	109
6.	Deep Artificial Neuron-Astrocyte Network	111
6.1.	Diseño y Metodología	112
6.1.1.	Problemas a Resolver	112
6.1.2.	Preparación de los Datos	113
6.1.3.	Diseño y Construcción de la Red	113
6.1.4.	Fase de Entrenamiento, Validación y Test	114
6.1.5.	Algoritmo Astrocítico	115
6.1.6.	Análisis y Organización de Resultados	117
6.1.7.	Validación del Sistema	117
6.2.	Resultados	117
6.3.	Discusión	119
7.	Conclusiones.....	121
8.	Futuros Desarrollos	125
9.	Bibliografía	127

Lista de figuras

Figura 1. Diagrama de Venn sobre <i>Deep Learning</i>	15
Figura 2. Tinción de Nissl y Golgi.....	34
Figura 3. Representación de la sinapsis tripartita.....	38
Figura 4. Time-lapse de un cultivo de astrocitos	39
Figura 5. Liberación de gliotransmisores por parte del astrocito	40
Figura 6. Esquema de la sinapsis tripartita.	41
Figura 7. Retracción del proceso astrocitario y crecimiento del proceso astrocítico	42
Figura 8. Representación del aumento del perímetro del astrocito	43
Figura 9. Citoarquitectura del cerebro humano	44
Figura 10. Diagrama de flujo sobre inteligencia artificial.....	45
Figura 11. Extracción de características de las imágenes con convoluciones	49
Figura 12. Gráfico de arquitecturas representativas de <i>Deep Learning</i>	50
Figura 13. Diagrama con la clasificación de los modelos neuronales	60
Figura 14. Modelo simplificado de una neurona artificial.	65
Figura 15. Función de activación sigmoidea.....	65
Figura 16. Estructura básica de una RNA.	66
Figura 17. Esquema simplificado de la sinapsis tripartita	67
Figura 18. Esquema de la arquitectura de una RNGA.....	75
Figura 19. Esquema de la metodología de las RNGA.	76
Figura 20. Retracción del proceso astrocitario debido a la estimulación	84
Figura 21. Pseudocódigo de las RNGA con modificación de la salida de la neurona.	92
Figura 22. Dropout	115
Figura 23. Pseudocódigo del funcionamiento de las DANAN.....	116

Lista de tablas

Tabla 1. Hitos de Deep Learning	51
Tabla 2. Modelos computacionales de las neuronas	59
Tabla 3. Combinaciones de activación-iteración de las RNGA.	79
Tabla 4. Arquitecturas de las redes empleadas en cada problema.....	88
Tabla 5. Parámetros de los algoritmos genéticos empleados.....	91
Tabla 6. Resultados en el problema de diagnóstico del cáncer de mama de la RNGA con el algoritmo de depresión y la RNA	98
Tabla 7. Resultados en el problema de señales de la ionosfera de la RNGA con el algoritmo de depresión y la RNA	99
Tabla 8. Resultados en el problema de señales de sonar de la RNGA con el algoritmo de depresión y la RNA	100
Tabla 9. Resultados en el problema de diagnóstico de diabetes de la RNGA con el algoritmo de depresión y la RNA	101
Tabla 10. Resultados en el problema de setas venenosas de la RNGA con el algoritmo de depresión y la RNA	102
Tabla 11. Resultados en el problema de diagnóstico del cáncer de mama de la RNGA con el algoritmo de potenciación y la RNA.....	103
Tabla 12. Resultados en el problema de señales de la ionosfera de la RNGA con el algoritmo de potenciación y la RNA.....	104
Tabla 13. Resultados en el problema de señales de sonar de la RNGA con el algoritmo de potenciación y la RNA	105
Tabla 14. Resultados en el problema de diagnóstico de diabetes de la RNGA con el algoritmo de potenciación y la RNA.....	106
Tabla 15. Resultados en el problema de setas venenosas de la RNGA con el algoritmo de potenciación y la RNA	107
Tabla 16. Estructura en Keras de la arquitectura de la red de tres capas oculta.....	114

Lista de gráficas

Gráfica 1. Resumen de los resultados obtenidos en el test por la RNA y RNGA en los cinco problemas	108
Gráfica 2. Resumen de los resultados obtenidos en el test por la RNGA que actúa sobre los pesos de las conexiones y la que actúa sobre las salidas de las neuronas para los cinco problemas analizados.....	109
Gráfica 3. Resumen de los resultados obtenidos en el test por la RNA y la DANAN con la arquitectura de una capa oculta.....	118
Gráfica 4. Resumen de los resultados obtenidos en el test por la RNA y la DANAN con la arquitectura de tres capas ocultas	118

2. Fundamentos Biológicos

2.1. Sistema Nervioso

El sistema nervioso (SN) es un conjunto altamente organizado de células que permiten a los animales interactuar con su entorno. Para ello el SN recibe información del entorno y del propio organismo. Debido a la gran cantidad de información que recibe, es necesario filtrarla para poder procesarla, integrarla, y almacenarla. En función de esta información, el SN puede generar dos tipos de respuestas: involuntarias, o reflejas, y voluntarias.

El SN se divide en Sistema Nervioso Central (SNC) y Sistema Nervioso Periférico (SNP). En los seres humanos el SNC está formado por la médula espinal y el encéfalo. El encéfalo a su vez está formado por el cerebro, el cerebelo, el tronco encefálico y el bulbo raquídeo. El SNP se divide en Sistema Nervioso Periférico visceral, también denominado vegetativo o involuntario, y Sistema Nervioso Periférico somático, o voluntario. El SNP visceral está formado por los nervios que inervan los órganos internos, los vasos sanguíneos y las glándulas. El SNP somático se compone de los nervios que inervan la piel, las articulaciones y los músculos.

A lo largo del siglo XIX se realizaron grandes avances en las técnicas de microscopía y tinción que dieron lugar a la Neurociencia como disciplina científica. El alemán Franz Nissl descubrió que una clase de tintes teñían los núcleos de todas las células, así como el material que rodea el núcleo de las neuronas, los denominados cuerpos de Nissl (figura 2). La tinción de Nissl permitió realizar un gran avance en Neurociencia ya que se podían diferenciar dos tipos de células en el SN, las neuronas y las células gliales, además de ayudar a estudiar la distribución de las células en las diferentes regiones, o citoarquitectura. Los estudios de la citoarquitectura del cerebro permitieron descubrir que el cerebro consiste en diversas regiones especializadas, hoy en día sabemos que cada región lleva a cabo funciones diferentes.

Pero la tinción de Nissl presenta ciertas limitaciones para el estudio del SN ya que no es posible estudiar en detalle la estructura de las neuronas debido a que solo se tiñe su núcleo y los cuerpos de Nissl a su alrededor. En 1873, Camillo Golgi desarrolló una nueva técnica basada en una solución de cromato de plata, la denominada tinción de Golgi, que colorea un pequeño porcentaje de las neuronas (figura 2). Esta técnica hizo posible observar la estructura de la neurona con mayor detalle, diferenciándose el cuerpo neuronal, y dos tipos de ramificaciones, las dendritas y el axón.

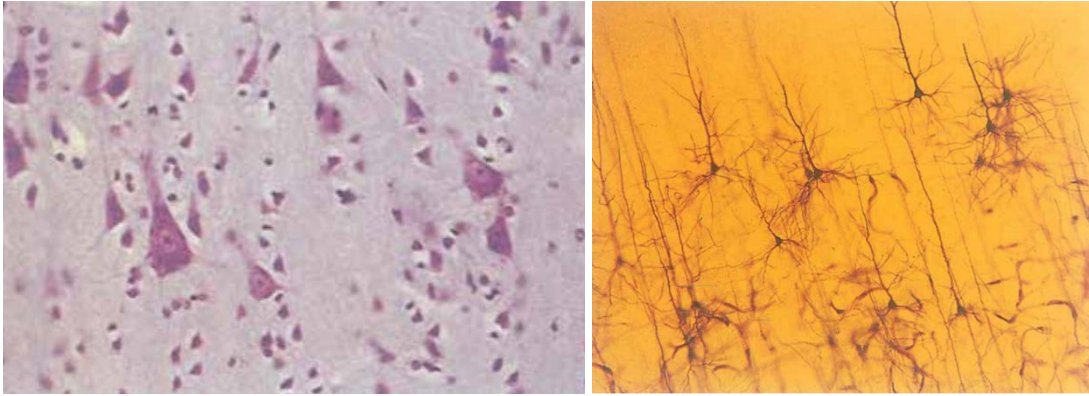


Figura 2. A la izquierda neuronas con la tinción de Nissl. A la derecha neuronas con la tinción de Golgi.

A pesar de que Camillo Golgi fue quien desarrolló la técnica de tinción, el histólogo y artista Santiago Ramón y Cajal fue quien supo aprovecharla para comprender mejor el funcionamiento del SN. Ambos emplearon la tinción de Golgi para estudiar el SN durante década, pero llegaron a conclusiones opuestas sobre su estructura. Golgi concluyó que las neuronas estaban unidas a través de las ramificaciones, formando un retículo continuo, similar al sistema circulatorio. Según la teoría de Golgi, el cerebro es una excepción a la teoría celular, que establece que la célula es el elemento funcional básico de todos los tejidos animales. Sin embargo, Santiago Ramón y Cajal extrajo una conclusión opuesta de sus estudios, según la cual las neuronas no forman un retículo continuo, sino que forman una red y se comunican mediante contacto. Esta teoría es la que se conoce como Doctrina Neuronal. A pesar de que Golgi y Ramón y Cajal tenían ideas opuestas y mantuvieron una gran rivalidad científica, ambos compartieron el premio nobel de Fisiología y Medicina en el año 1906 por sus aportaciones fundamentales, que dieron lugar al nacimiento de la Neurociencia Moderna (Bear et al., 2007).

Desde estas primeras observaciones se ha avanzado mucho en el estudio del cerebro y la caracterización de las células que componen el SN. La estructura general de las neuronas se compone de tres partes: las dendritas, el soma, o cuerpo neuronal, y el axón, aunque existen neuronas que carecen de axón. Las dendritas suelen estar muy ramificadas y forman lo que se denominan árbol dendrítico, normalmente las neuronas reciben la información a través de las dendritas. El soma representa el cuerpo celular y es la parte de mayor diámetro, además es donde se encuentra el núcleo de la célula y la mayor parte de los orgánulos celulares. El axón suele ser la parte de mayor longitud de la célula, pudiendo alcanzar un gran tamaño, por ejemplo, en los calamares puede llegar a medir varios metros, aunque la mayoría de las neuronas tienen un tamaño que oscila entre 10 y 50 micrómetros. Las células

del SN presentan una gran variabilidad y se pueden clasificar de diferentes maneras dependiendo de su morfología, función, polaridad... Algunos ejemplos de clasificación:

- Dendritas: múltiples clasificaciones dependiendo de la región, por ejemplo, en el córtex: neuronas estrelladas y neuronas piramidales. También se pueden clasificar entre las neuronas que tiene espinas dendríticas y las que no.
- Función: Neuronas primarias sensoriales, interneuronas y neuronas motoras.
- Longitud del axón: Neuronas de Golgi tipo I o neuronas de proyección, Neuronas de Golgi tipo II, o neuronas de circuito local. Las Neuronas de Golgi tipo I se pueden clasificar según el número de prolongaciones en: monopolares, bipolares y multipolares.

Aunque los estudios genéticos han demostrado que las células del SN pueden, y deberían ser clasificadas en función de la expresión genética. El ser humano cuenta con unos 20.000 genes, de los cuales 14.000 se expresan en el cerebro, y 6.000 única y exclusivamente en el cerebro.

Las células gliales fueron descubiertas por Rudolf Virchow en 1859 y durante mucho tiempo se les atribuyó la única función de “pegamento” o unión (significado en latín de la palabra glía). Pero a lo largo del siglo XX se ha descubierto que las células gliales tienen más funciones incluso que las neuronas, clasificándose en varios subtipos: astrocitos, glía mielinizante (oligodendrocitos en el SNC y Células de Schwann en el SNP), microglía, glía epitelial. Las principales características y funciones de las células gliales son las siguientes:

- Astrocitos: Se trata del tipo de glía más numeroso en el cerebro. Se ha demostrado que los astrocitos influyen en la sinapsis de diversas maneras. Por una parte, poseen receptores para la mayoría de neurotransmisores: (Perea & Araque, 2010). Lo cual permite que puedan conocer la actividad neuronal. Por otra parte, los astrocitos puede emitir gliotransmisores (similares a los neurotransmisores), que pueden afectar a la actividad neuronal. También regulan la concentración de diversos iones en el espacio sináptico e incluso pueden modificar su morfología regulando el volumen de la hendidura sináptica lo cual afecta la concentración de iones y neurotransmisores y por tanto a la intensidad de la transmisión sináptica
- Glía mielinizante: Oligodendrocitos (SNC) y Células de Schwann (SNP). La glía mielinizante se encarga de recubrir con mielina lo axones de las neuronas, lo que permite que los impulsos se transmiten de manera eficiente por los mismos. Este recubrimiento de mielina es discontinuo, y cada cierta distancia existen regiones libres de recubrimiento que se denominan nódulos de Ranvier. Los

oligodendrocitos se encuentra exclusivamente en el SNC y pueden recubrir los axones de diversas neuronas. Por su parte, las células de Schwann se localizan en el SNP y solo recubren el axón de una única neurona.

- Microglía: La microglía hace el papel del sistema inmune en el SNC (Cunningham et al., 2013), actuando como los fagocitos al eliminar los desechos, las células muertas y los agentes extraños. Además, pueden limitar la producción de neuronas corticales mediante la fagocitación de los precursores neuronales. También juegan un papel importante en la remodelación de las conexiones sinápticas y son las únicas células móviles del SN.
- Glía epitelial (o células endimarias o endimocitos). Estas células facilitan el movimiento del líquido cefalorraquídeo, interacción con los astrocitos para formar la barrera hematoencefálica, recubriendo los ventrículos del cerebro y manteniéndolo aislado.
- Glía NG2: ES último tipo de célula glial descubierta y se caracteriza por expresar el proteoglicano *chondroitin sulphate NG2*. Aunque sus funciones aún no están del todo claras, se ha comprobado que este tipo de glía mantiene la capacidad proliferativa a lo largo de toda la vida, constituyendo casi el 90% de las células con capacidad de división en el cerebro adulto (Fields, 2009) y son precursoras de neuronas, astrocitos, oligodendrocitos.

Las funciones clásicas asociadas a las células gliales eran de soporte trófico y metabólico de las neuronas (Ramon y Cajal, 1904; Ramón y Cajal, 1952). En las últimas décadas se han descubierto numerosas funciones nuevas de las células gliales. Durante el desarrollo un tipo de glía, denominado glía NG2, actúa como precursor de las neuronas (Nishiyama et al., 2009). Además, participan en el proceso de diferenciación, guía y supervivencia neuronal (Raff et al., 1993; Stevens, 2008). También juegan un papel clave en la formación, mantenimiento y regulación de las sinapsis (Pfrieger, 2002), así como en el crecimiento y guía de los axones y las dendritas (Le Roux & Reh, 1995; Shu & Richards, 2001).

2.2. Procesamiento de la Información en el Cerebro

Después de presentar algunos conceptos básicos de Neurociencia, a continuación, se analiza el procesamiento de la información en el cerebro desde el punto de vista computacional. La Neurociencia se puede dividir en diferentes categorías según el nivel en el que se centre, aunque todas están relacionadas: Neurociencia molecular, Neurociencia celular, Neurociencia de sistemas, Neurociencia del comportamiento y Neurociencia

cognitiva. Pero como se ha comentado en la introducción, para desarrollar modelos *in silico* de IA inspirados en la Neurociencia, no es necesario estudiar en detalle todos los mecanismos biológicos. Por tanto, se presentarán los mecanismos y teorías sobre el procesamiento de información desde los dos niveles superiores de la escala de Marr y Poggio (Marr & Poggio, 1976), el nivel computacional y nivel de algoritmos, ya que son los que resultan útiles para el trabajo desarrollado en esta tesis (Hassabis et al., 2017).

Existe una gran variedad de teorías sobre el funcionamiento del SN y cómo se procesa la información. Aunque la más aceptada y generalizada es la teoría neuronal, que propone que las neuronas realizan el procesamiento de la información y se comunican con otras neuronas por medio de dos tipos sinapsis, eléctricas y química. En la sinapsis eléctrica las neuronas están unidas y la corriente eléctrica puede circular de una otra, de esta manera se transmiten las señales eléctricas de forma directa. En este tipo de sinapsis la corriente puede circular en las dos direcciones, por lo que la información fluye en ambos sentidos. En las sinapsis químicas entre la neurona presináptica y la neurona postsináptica existe una separación, denominada hendidura sináptica. La neurona presináptica emite neurotransmisores a la hendidura sináptica, la neurona postsináptica capta estas señales químicas, y las integra con las que recibe de otras neuronas. Si las señales recibidas superan un umbral, se produce un potencial de acción, o *spike*, que se transmite a lo largo de la neurona y provoca que esta emita señales a otras neuronas. Se considera que estos potenciales de acción tienen un mecanismo de todo o nada, ya que solo se dispara si se superar un valor umbral. Desde el punto de vista computacional se puede considerar como un estado binario, 0 cuando está en reposos y 1 cuando se produce un potencial de acción. La capacidad de computación del cerebro humano es imposible de calcular, pero existen ciertas estimaciones. Teniendo en cuenta la teoría neuronal, y los estudios de la citoarquitectura del cerebro se puede realizar una aproximación de la capacidad de cómputo. El cerebro humano está formado por unos 86.000 millones de neuronas (Azevedo et al., 2009), y cada neurona se comunica con miles de neuronas, dando lugar a billones de sinapsis más de diez billones de sinapsis. Se estima que cada sinapsis puede realizar unas 200 operaciones por segundo, por lo que el cerebro podría realizar unos 20.000 millones de operaciones por segundo.

2.3. Astrocitos

Durante mucho tiempo se consideró que las neuronas eran las únicas células excitables del SN y que podían transmitir y procesar la información. Esta ha sido la teoría imperante durante casi 100 años, principalmente porque debido a las limitaciones técnicas sólo era posible estudiar la excitabilidad eléctrica de las células. Pero en los años ochenta se realizaron los primeros descubrimientos que pusieron en duda que las neuronas fueran las únicas células capaces de transmitir y procesar información. Gracias a los avances en las técnicas de imagen y marcadores fluorescentes, se pudo comprobar que los astrocitos también son células excitables. Varios experimentos demostraron que los astrocitos responden al glutamato con una elevación de la concentración de Ca^{2+} intracelular. Además, se observó que el glutamato podría inducir la propagación de ondas de calcio entre los astrocitos adyacentes (Charles et al., 1991; A. H. Cornell-Bell et al., 1990) (figura 3).

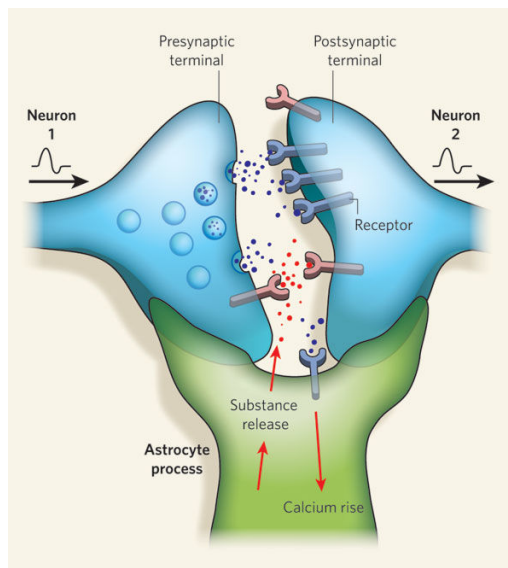


Figura 3. Representación de la sinapsis tripartita.

Existe una diferencia importante en la forma de transmitir la información de las neuronas y los astrocitos, las primeras propagan las señales mediante impulsos eléctricos que se propagan a una gran velocidad, entre 10-100 m/s. Por su parte, los astrocitos producen ondas de calcio que se propagan a menor velocidad, entre 4-20 m/s (figura 4).

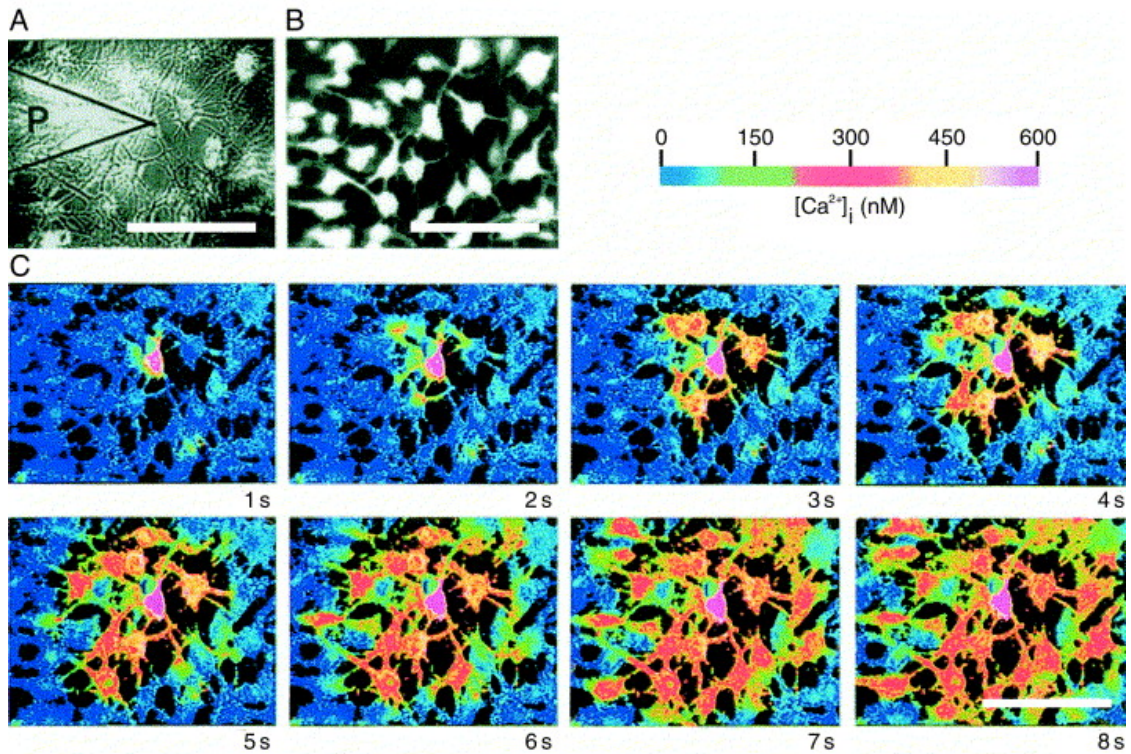


Figura 4. *Time-lapse* de un cultivo de astrocitos con un marcador fluorescente que permite observar la propagación de una onda de Ca^{2+} (Alfonso Araque et al., 1999).

Los primeros experimentos que demostraron el verdadero potencial de la glía se llevaron a cabo a principios de los años noventa. En primer lugar, en experimentos *in vitro* se observó que, al aplicar un potencial eléctrico a un astrocito en un cultivo, éste puede provocar aumentos de calcio intracelular en neuronas adyacentes (Nedergaard, 1994). En ese mismo año, se comprobó que, al aumentar la concentración de calcio intracelular de los astrocitos, éstos pueden liberar glutamato a la hendidura sináptica, lo cual produce un aumento de calcio intracelular en las neuronas adyacentes (Parpura et al., 1994).

En el año 1998, se publican los trabajos de cuatro grupos independientes sobre la capacidad de los astrocitos de modular la transmisión sináptica (Oberheim et al., 2012a). Estos estudios se realizaron en cuatro regiones diferentes del SN: (A) en las uniones neuromusculares, la modulación la realizan las células de Schwann (Bourque & Robitaille, 1998); (B) en el hipocampo de las ratas, se comprobó que los astrocitos modulan la magnitud del potencial de acción, provocando corrientes postsinápticas excitatorias o inhibitorias (Alfonso Araque et al., 1998); (C) en la retina las células gliales modulan la actividad de las neuronas ganglionares (Newman & Zahs, 1998); (D) los astrocitos del hipocampo liberan glutamato que provoca la despolarización de las neuronas piramidales de CA1 (Kang et al., 1998).

2.4. Sinapsis Tripartita

De estos trabajos sobre los astrocitos surgió la idea de un nuevo modelo sináptico, conocido como sinapsis tripartita, que considera que la comunicación sináptica se produce entre una neurona presináptica, una neurona postsináptica y un astrocito perisináptico (Araque, 1999). Este modelo asume que la liberación de neurotransmisor desde la terminal presináptica activa a la glía, la cual liberará una cantidad adicional de neurotransmisores, denominados gliotransmisores, normalmente glutamato y/o ATP (Theodosis et al., 2008) (figura 5).

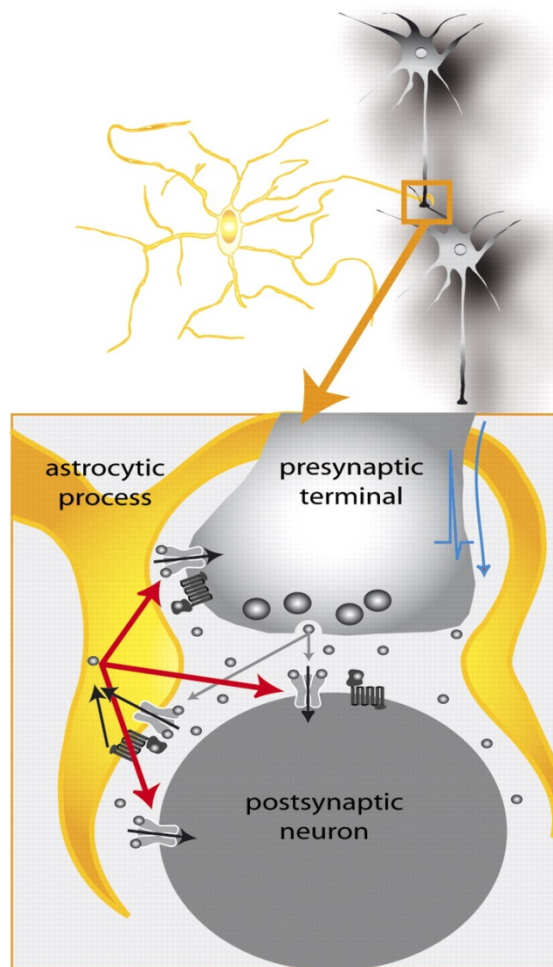


Figura 5. Liberación de gliotransmisores por parte del astrocito como respuesta a los neurotransmisores liberados por la neurona presináptica (Theodosis et al., 2008)

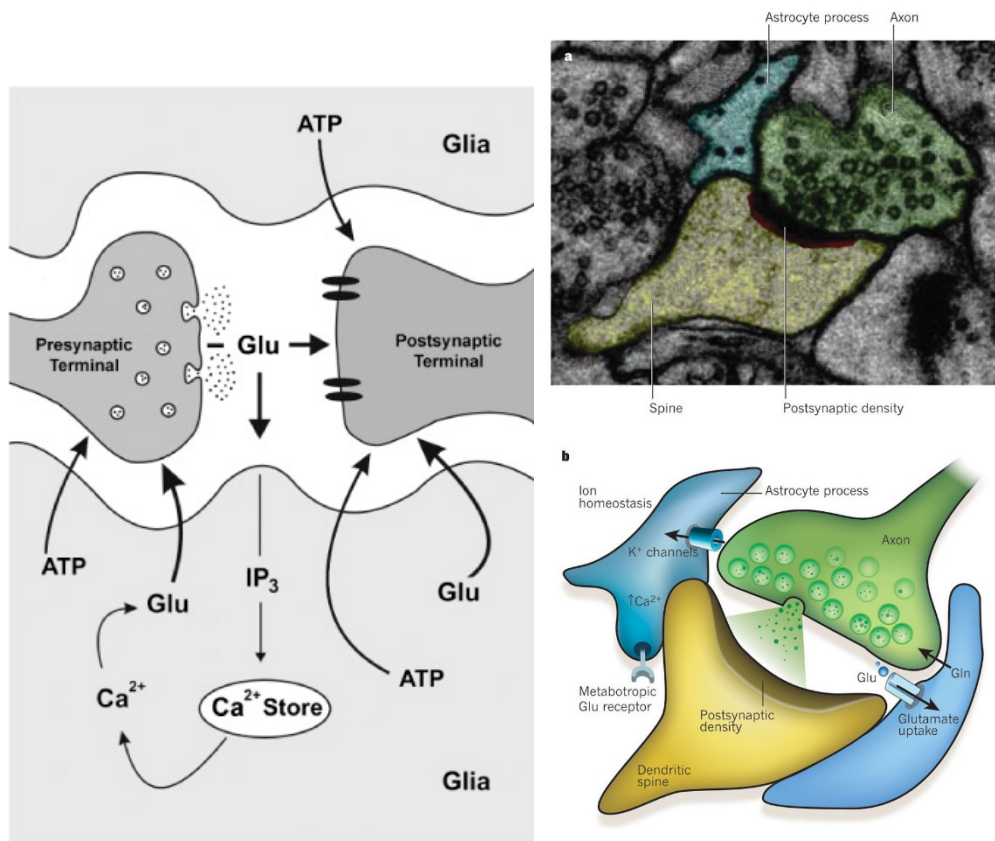


Figura 6. Esquema de la sinapsis tripartita. El astrocito responde al glutamato liberado por la terminal presináptica aumentado el calcio intracelular, esto provoca la liberación de ATP o glutamato (Sotero & Martínez-Cancino, 2010).

La actividad sináptica también puede modularse mediante la modificación de la estructura física del astrocito. Se ha demostrado que el glutamato puede causar el incremento del número de filopodios en la superficie del astrocito (A. Cornell-Bell et al., 1990). Este proceso puede afectar a la transmisión de información de varios modos: un proceso astrocítico puede aislar una sinapsis impidiendo la difusión del neurotransmisor a otras sinapsis vecinas (figura 7). En el caso de que el astrocito libere algún neurotransmisor, la modificación de la distancia entre la neurona y el astrocito afectará a la transmisión sináptica (figura 7) (Theodosis et al., 2008). El astrocito también puede aumentar su tamaño y reducir el espacio sináptico, potenciando la conexión sináptica (figura 7) (Genoud et al., 2006).

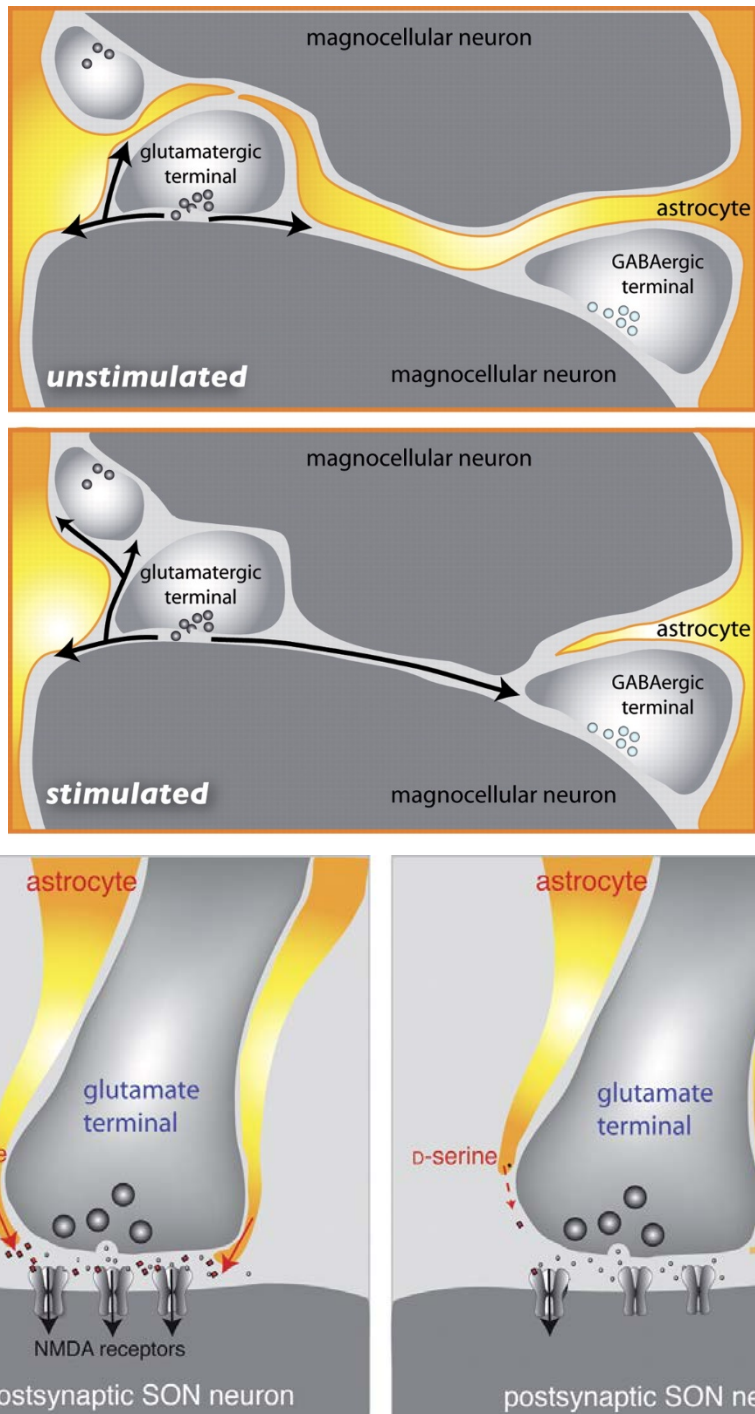


Figura 7. A. Retracción del proceso astrocitario debido a la estimulación, el glutamato liberado en la terminal glutamatérgica puede afectar a la sinapsis gabaérgica. B. El crecimiento del proceso astrocítico aumenta el efecto de la D-Serina, en la neurona postsináptica (Theodosis et al., 2008).

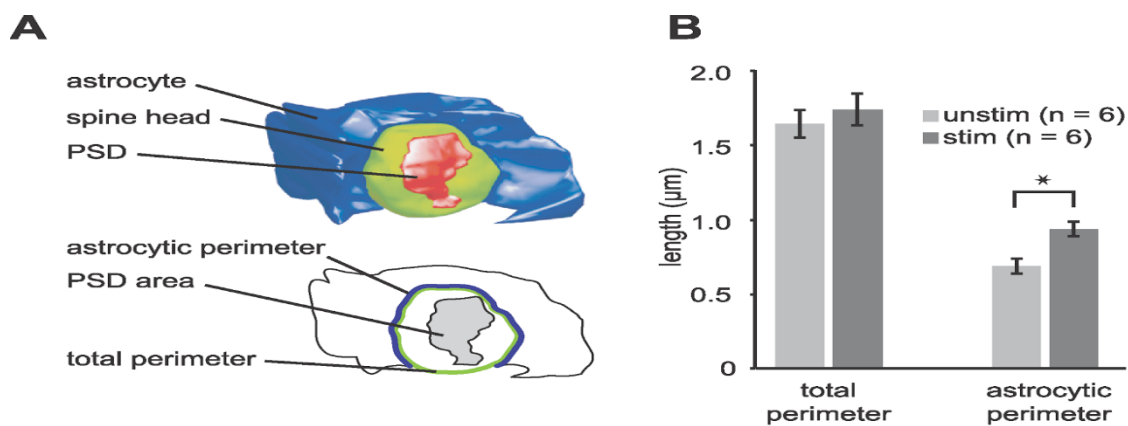


Figura 8. A. Representación 3D de una espina dendrítica (verde), un astrocito (azul) y el área que expresa la proteína “postsynaptic density” (rojo). B. Gráfica que muestra el aumento del perímetro del astrocito tras la estimulación (Genoud et al., 2006).

Los estudios de Araque, Perea y colaboradores sobre astrocitos (Perea & Araque, 2002) permiten extraer tres conclusiones importantes:

1. Los astrocitos poseen dominios funcionales que responden independientemente a distintos neurotransmisores liberados sinápticamente.
2. Los astrocitos son capaces de discriminar la actividad de distintas vías aferentes.
3. La activación sináptica de receptores astrocitarios es un proceso muy localizado espacialmente, que no resulta de un vertido incontrolado de neurotransmisores en el espacio sináptico, sino de una comunicación precisa, selectiva y controlada entre neurona y astrocitos.

El grado de especialización de los astrocitos aumenta a lo largo de la escala evolutiva. Por ejemplo, comparando la citoarquitectura del cerebro de roedores y humanos, se comprobó que los astrocitos humanos tienen casi 17 veces más volumen que los de roedores, cubren entre 270.000-2.000.000 de sinapsis y en roedores solo entre 20.000-120.000 sinapsis (Oberheim et al., 2009). Además, se demostró que los astrocitos interlaminares del córtex son exclusivos de primates (Oberheim et al., 2009, 2012b). También se observó que en humanos los astrocitos poseen dominios anatómicos que apenas se solapan un 5% con los dominios de otros astrocitos. En roedores, el grado de solapamiento de los astrocitos es menor debido a su menor tamaño (Bushong et al., 2002; Oberheim et al., 2012b; Ogata & Kosaka, 2002).

Por otra parte, la distribución de neuronas y células gliales en el SN no es homogénea. En el ser humano, el cerebelo tiene casi cuatro veces más neuronas que células gliales, en el córtex ocurre lo contrario, hay casi cuatro veces más células gliales que neuronas. Por tanto, parece que las regiones donde tienen lugar las funciones más complejas presentan un

mayor porcentaje de astrocitos (figura 9) (Azevedo et al., 2009). Aunque algunos autores afirman que la mayor proporción de astrocitos en el neocórtex se debe a los altos requerimientos metabólicos de las neuronas relacionados con el coste energético de mantener los árboles dendríticos y las proyecciones de los axones (Sherwood et al., 2006).

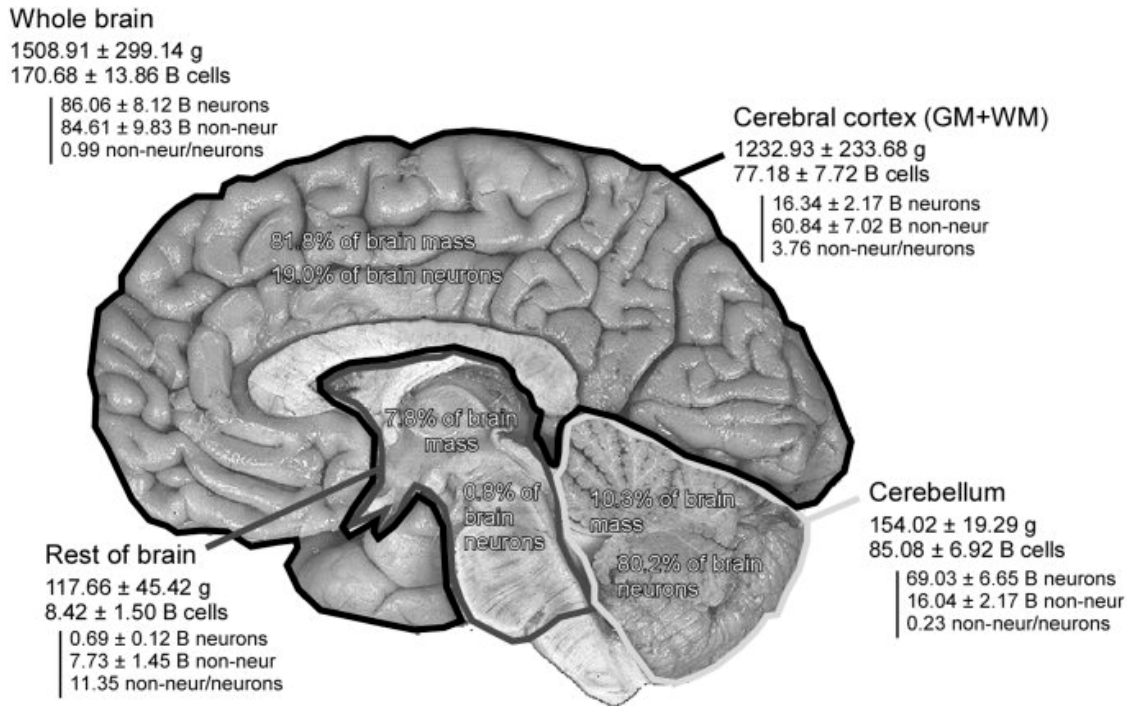


Figura 9. Citoarquitectura del cerebro humano (Azevedo et al., 2009).

Esta tesis doctoral se basa en los descubrimientos sobre las funciones de los astrocitos en el procesamiento de la información y en el concepto de la sinapsis tripartita. Como se ha comentado el objetivo es poder favorecer el avance de la IA y a su vez evaluar la validez de algunas hipótesis planteadas sobre la importancia de los astrocitos en el procesamiento de la información, y por tanto en la inteligencia. A continuación, se presentarán los fundamentos computacionales de esta tesis, comenzando con un breve resumen histórico de los en IA y su relación con la Neurociencia.

3. Fundamentos Computacionales

La IA es un campo muy amplio, y como se ha explicado en la introducción, existen dos grandes ramas, la IA científica y la IA tecnológica. A su vez la IA tecnológica, se divide en sistemas expertos y *Machine Learning*. Los modelos empleados en esta tesis doctoral se engloban dentro del *Machine Learning*, y a su vez dentro del *Representation Learning*. En concreto son RNA, tanto *shallow* como de *Deep Learning* (figura 10). Por ello a continuación se presenta un breve resumen histórico de la evolución de estos modelos y su relación con la Neurociencia.

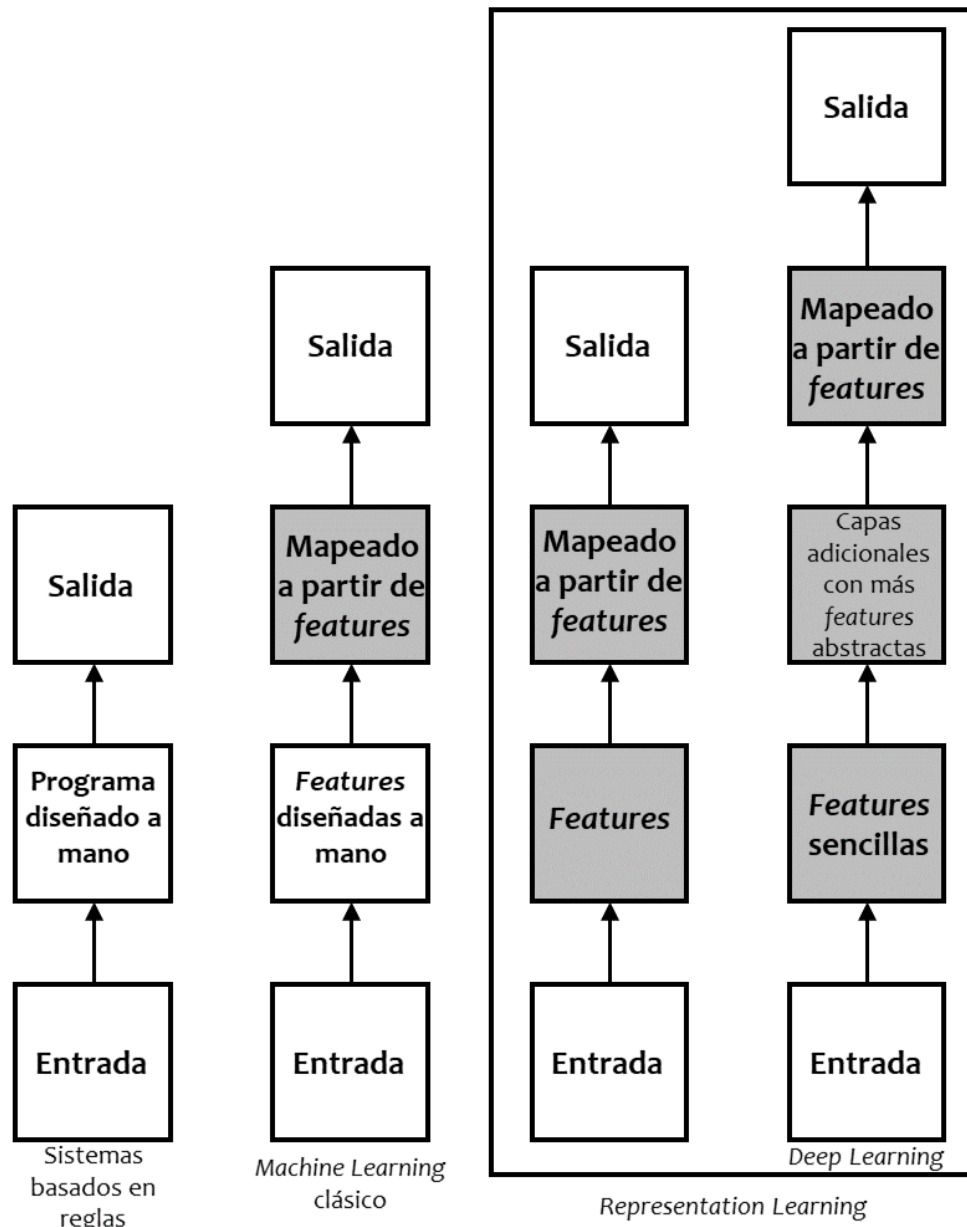


Figura 10. Diagrama de flujo con las diferentes partes de los sistemas de Inteligencia Artificial tecnológica en el que se pueden observar las similitudes y diferencias que existen entre ellos. Las cajas en gris muestran los componentes que se pueden aprender automáticamente de los datos.

Adaptación del libro *Deep Learning* (I. Goodfellow et al., 2016).

3.1.1. Desarrollo Histórico

En 1943 McCulloch y Pitts publicaron "*A logical calculus of the ideas immanent in nervous activity*" (McCulloch & Pitts, 1943), en el que proponían los primeros modelos neuronales computacionales con el objetivo de reproducir las capacidades extraordinarias del cerebro intentando imitar la estructura y los principios básicos de funcionamiento de las neuronales. Estos modelos neuronales, o neuronas artificiales, están representados por unidades lógicas con un umbral, que reciben entradas binarias con unos pesos asociados, integran estas entradas, y producen una salida binaria que depende del valor umbral. La interconexión de varias neuronas artificiales forma lo que se denomina como Red de Neuronas Artificiales. Este primer modelo de RNA presentaba un problema importante, la necesidad de asignar los parámetros de cada célula (conexiones, umbrales, pesos de las conexiones, etc.) de manera manual. Esto suponía una gran limitación porque los cálculos necesarios aumentan exponencialmente al aumentar el número de elementos.

En 1948 Alan Turing escribió un trabajo sobre la IA titulado "Intelligent Machinery", que no fue publicado hasta 1968 (Turing, 1948). En este artículo propuso dos nuevos tipos de modelos computacionales: las *unorganized machines* tipo A y tipo B. Estos modelos representan establecen los fundamentos de los sistemas conexionistas, aunque a largo de la historia no ha sido reconocido de la manera adecuada.

Donald Hebb publicó el libro "The organization of Behavior" (Hebb, 1949), en el que propuso la denominada "regla de Hebb" o Teoría de la Asamblea Celular, según la cual "*neurons that fire together, wire together*", es decir, cuando dos neuronas que se activan a la vez, se fortalece su conexión. La teoría de Hebb supuso un gran avance ya que sentó las bases de un aspecto fundamental del comportamiento de las redes neuronales del cerebro, la plasticidad neuronal.

En la década de los 50, Frank Rosenblatt combinó la idea del aprendizaje de Hebb con las Células de McCulloch-Pitts para proponer modelo del *Perceptron* multicapa (Rosenblatt, 1957). Este tipo de RNA supone un gran avance ya que los pesos de las conexiones se ajustaban con un proceso de aprendizaje mediante ejemplos por medio de un conjunto de patrones, denominados patrones de entrenamiento. Así, introducimos en la red un conjunto de valores para los que se conoce la salida correcta, la cual se compara con la salida que proporciona la red, y en caso de ser incorrecta, se ajustan los pesos de las conexiones según un algoritmo matemático.

A finales de la década de los 50, Bernard Widrow y Marcian Hoff (Widrow & Hoff, 1960), desarrollaron los modelos llamados ADALINE y MADALINE (*Multiple ADaptive LINear*

Elements). ADALINE fue diseñado para reconocer patrones binarios, en concreto el sistema analizaba la transmisión de bits de una línea de teléfono y trataba de predecir el siguiente bit. MADALINE es una red de tres capas de ADALINE, y fue la primera RNA aplicada a problemas del mundo real, usando filtros adaptativos para eliminar los ecos de las líneas de teléfonos.

Estos primeros modelos han dado lugar a la rama conexionista de la IA, la cual ha evolucionado mucho desde sus orígenes y hoy en día existe un crecimiento exponencial en las aplicaciones con RNA. Sin embargo, ha sufrido tiempos difíciles, sobre todo el denominado invierno de la IA que duró desde los años 70 hasta mediados de los 80 estuvo marcado por la publicación del libro “*Perceptrons*” de Marvin Minsky y Seymour Paper (Minsky & Paper, 1969) en el cual demostraban que el perceptron era incapaz de aprender la función lógica llamada disyunción exclusiva (XOR). Esta es una función no lineal simple, y las RNA de una capa oculta son incapaces de resolverla ya que solo pueden resolver funciones lineales.

Durante la década de los 70 y 80 apenas existieron investigaciones y proyectos sobre RNA, pero en 1986 se publicó un artículo que ayudaría a salir del invierno de la IA. El tataranieta de George Bool, Geoff Hinton, junto a David Rumelhart y Ronald Williams propusieron un algoritmo de aprendizaje para las RNA que revolucionaría el campo, el denominado “Algoritmo de Retropropagación” (Rumelhart et al., 1986). Este algoritmo ajusta los pesos de las conexiones de la red tratando de minimizar la diferencia entre el vector de salida de la red y el vector de salida deseado. El algoritmo funciona en varios, primero propagando hacia adelante la entrada de la red, luego calculando la derivada de la función de pérdida y finalmente propagando hacia atrás los errores para modificar los parámetros de las capas anteriores. Este proceso se repite varias veces hasta que el error se estabiliza y alcanza un mínimo. Esta técnica de entrenamiento permite crear RNA de varias capas ocultas, superando el problema de la resolución de funciones no lineales.

En los años 80 se publicaron los primeros trabajos con otro tipo de RNA, las denominadas redes neuronales de convolución (RNC). Estas redes están inspiradas en el sistema visual y su organización jerárquica por capas, cada una con mayor nivel de abstracción. Se considera que las RNC tiene su origen en el modelo conocido como Neocognitron (Fukushima & Miyake, 1982). En los 90 se crearán diversas aplicaciones que utilizaban RNC, principalmente para tareas de reconocimiento de patrones, especialmente imágenes y audio. Uno de los modelos más destacados es la conocida como LeNet-5, una red de 7 capas diseñada para reconocer dígitos manuscritos (LeCun et al., 1998).

Durante la década de los 80 también se desarrollaron otro de los principales modelos de RNA, las Redes de Neuronas Artificiales Recurrentes (RNAR). Como su propio nombre indica, estas redes cuentan con conexiones recurrentes, lo que las hace ideales para problemas en los cuales existen entradas de datos secuenciales como son lenguaje o la voz.

A finales de los años 90 se desarrolló una técnica de computación evolutiva para el entrenamiento de las RNA, son los denominados algoritmos genéticos (Yao & Liu, 1997). Su funcionamiento está inspirado en la selección natural, partiendo de una población inicial de RNA, en cada generación se seleccionan las redes con menor error para cruzarlas y/o mutarlas para dar lugar a una nueva red con elementos de ambos progenitores.

Se puede considerar que en el siglo XX se establecieron las arquitecturas básicas de las RNA, y se desarrollaron los principales algoritmos. Pero había varios motivos para las limitadas aplicaciones de las RNA, los principales fueron: la falta de poder computacional, la escasez de datos, y limitación de las técnicas y algoritmos para diseñar las RNA.

La idea de construir RNA no es nueva, pero existía un obstáculo a la hora de entrenar redes de gran tamaño, es el denominado problema de la desaparición del gradiente (*vanish gradient*). Esta limitación existe porque cuando se entrenan redes con muchas capas empleando el algoritmo de retropropagación para optimizar los pesos, los valores del gradiente del error se reducen exponencialmente a medida que aumenta la profundidad de la red.

La última década ha sido la más fructífera en el campo de la IA gracias a los enormes avances en las RNA. Se han desarrollado nuevas técnicas y algoritmos de entrenamiento, lo cual ha resuelto en gran medida el problema de la desaparición del gradiente, permitiendo crear RNA de gran tamaño que obtienen resultados sorprendentes. Estas redes tienen diversos niveles de representación de los datos (figura 11), desde los más sencillos a los más abstractos, es por ello que se les denomina métodos de Aprendizaje Profundo, o *Deep Learning*. Al igual que en el cerebro, existe una integración jerárquica de la información en capas cada vez más abstractas.

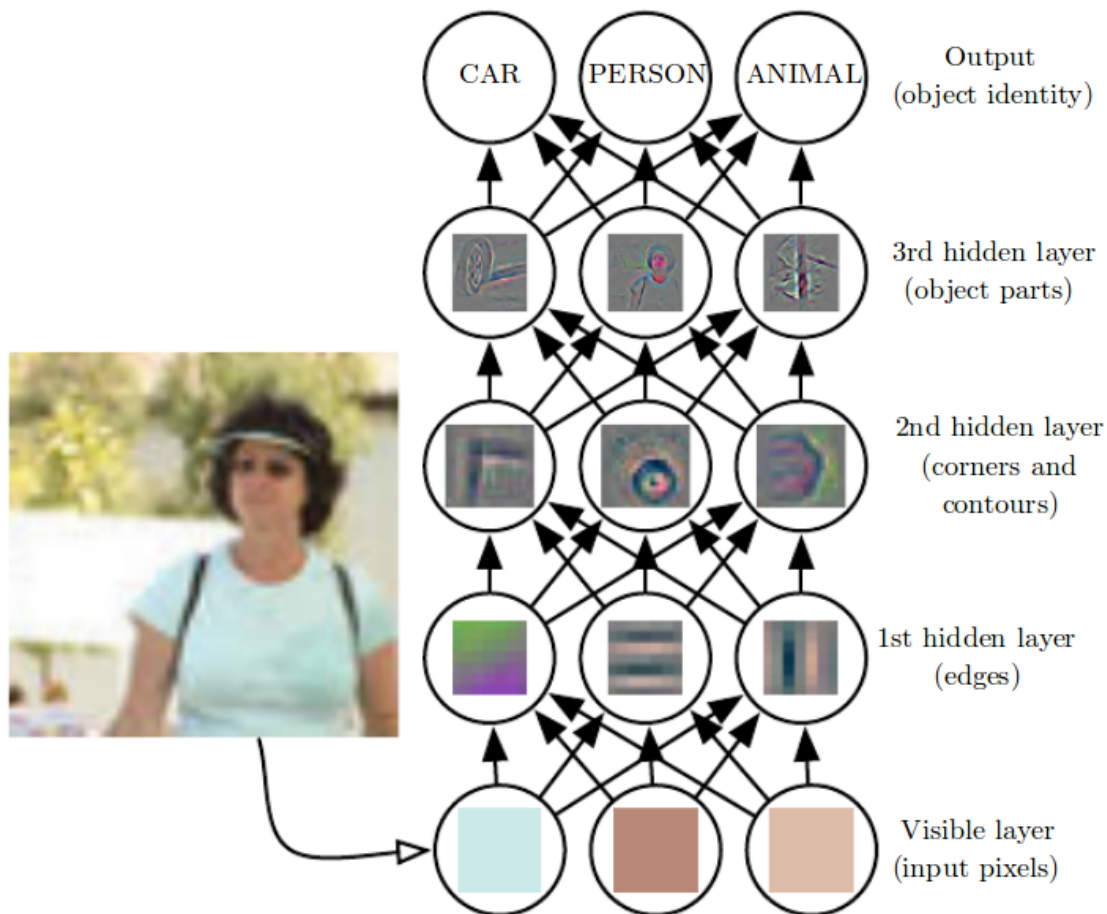


Figura 11. Extracción de características de las imágenes con una RNC (Zeiler & Fergus, 2014).

Se considera que el trabajo que marcó el inicio de la era del *Deep Learning* fue publicado en el año 2006, por Geoff Hinton y Ruslan (G. E. Hinton & Salakhutdinov, 2006), en el cual desarrollaron un nuevo método de preentrenamiento no supervisado capa a capa para inicializar los pesos de las RNA, y posteriormente realizar un ajuste fino de los mismos mediante un entrenamiento supervisado. Desde esta publicación se han realizado enormes avances a nivel teórico en el campo de *Deep Learning*. Además, hay que destacar que el crecimiento exponencial en la capacidad de computación y en los datos disponibles han facilitado mucho la investigación y desarrollo de aplicaciones basados en RNA.

La metodología de preentrenamiento no supervisado permitió sacar partido de la gran cantidad de datos no etiquetados que existían. Por ejemplo, en el reconocimiento de voz existían grandes volúmenes de datos para poder entrenar los modelos, es por ello que fue uno de los primeros campos en el que las RNA entrenadas con técnicas de *Deep Learning* obtuvieron los mejores resultados, estableciendo el algoritmo estados-del-arte.

Uno de los hitos en el campo se marcó en la competición *Large Scale Visual Recognition Challenge* (LSVRC). Esta competición se creó en el año 2010, y usa una base de datos

conocida como ImageNet creada por un grupo de la universidad de Stanford. ImageNet contiene millones de imágenes etiquetadas, y el objetivo de la competición es crear modelos de ML que realicen predicciones acertadas sobre las etiquetas de las imágenes. En los primeros años el porcentaje de error se encontraba alrededor del 28%, pero en el año 2012, Alex Krizhevsky, Ilya Sutskever, y Geoff Hinton publicaron una RNC que redujo a la mitad el error hasta el 16% (Krizhevsky et al., 2012) (figura 12). Esta red fue entrenada utilizando diversas técnicas que se convirtieron en imprescindibles para los modelos posteriores de *Deep Learning*. Una de las claves fue optimizar el entrenamiento en GPUs, pero además introdujeron la técnica de Dropout (Srivastava et al., 2014) para reducir el sobreentrenamiento y permitir la creación de redes de mayor tamaño, y utilizaron una función de activación lineal rectificadora (*Rectified Linear Unit*, o ReLU) (Nair & Hinton, 2010).

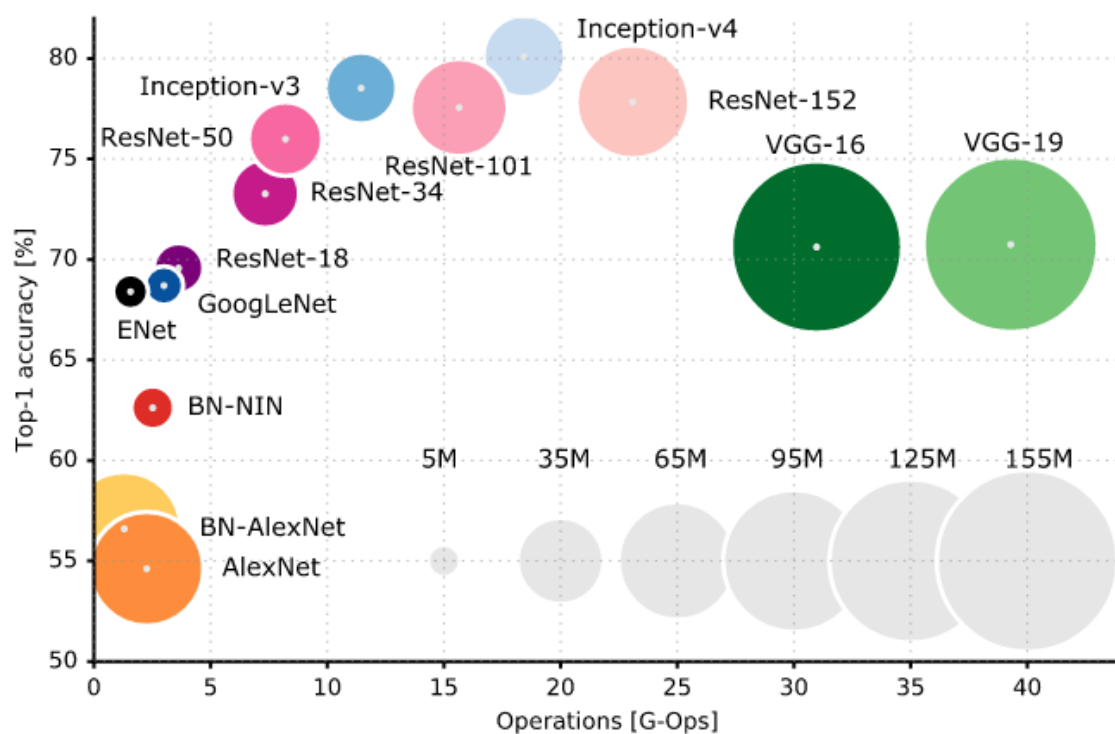


Figura 12. Gráfico con algunas arquitecturas representativas de *Deep Learning* utilizadas en la competición de *ImageNet*. En el eje de la x se muestra la capacidad de cálculo de las redes y en el eje de la y los resultados en la competición. El tamaño de los círculos representa el número de parámetros de la red (Canziani et al., 2016).

Otro de los hitos de la IA se produjo en el año 2017, con la victoria de un sistema basado en *Deep Learning*, denominado DeepStack, al juego de cartas heads-up no-limit Texas hold'em ante jugadores profesionales, venciendo tras jugar más de 44.000 manos (Moravčík

et al., 2017). En ese mismo año, otro sistema basado en *Deep Learning* denominado Libratus, venció a 4 jugadores profesionales, tras 120.000 manos jugadas, Libratus consiguió 1.8 millones de dólares (Brown & Sandholm, 2017). La dificultad del *heads-up no-limit Texas hold'em* consiste en que se trata de un juego con información imperfecta, ya que se trata de un juego de azar en el que los jugadores no disponen de toda la información. En la tabla 1 se pueden ver algunos de los hitos del *Deep Learning*.

Tabla 1. Competiciones y año en el que las redes de Deep Learning han obtenido los mejores resultados.

Tarea [Año]	Competición
Reconocimiento de texto manuscritos	MNIST, Arabic HWX (IDSIA)
Segmentación de imágenes volumétricas del cerebro [2009]	Connectomics (IDSIA, MIT)
OCR <i>in the Wild</i> [2011]	StreetView House Numbers (NYU y otros)
Reconocimiento de señales de tráfico [2011]	GTSRB competition (IDSIA, NYU)
Reconocimiento de acciones humanas [2011]	Hollywood II dataset (Stanford)
Detección de la mitosis en células del cáncer de pecho [2011]	MITOS (IDSIA)
Reconocimiento de objetos [2012]	ImageNet competition (Toronto)
Análisis de escenas [2012]	Stanford bgd, SiftFlow, Barcelona datasets (NYU)
Reconocimiento de voz [2012]	Acoustic modeling (IBM and Google)
Reconocimiento de texto asiático manuscrito [2013]	ICDAR competition (IDSIA)
Detección de peatones [2013]	INRIA datasets and others (NYU)
Análisis de escenas en imágenes complejas [2013]	NYU RGB-D dataset (NYU)
Videojuegos de Atari [2013]	2600 Atari games (Google DeepMind Technologies)
Juego de Go [2016]	AlphaGo vs Human World Champion (Google DeepMind Technologies)

Uno de los campos más interesantes para las aplicaciones de las RNA es la medicina, por ejemplo, en el diagnóstico de enfermedades. Investigadores de la universidad de Stanford crearon un sistema de *Deep Learning* capaz de diagnosticar enfermedades de la piel mejor que los propios dermatólogos (Esteva et al., 2017). En el año 2017 publicaron un artículo en el que presentaban su trabajo con una base de datos con casi 130.00 imágenes clínicas que utilizaron para entrenar una RNA capaz de reconocer más de 2.000 enfermedades.

Todos estos avances han provocado que cada vez sea más difícil definir los comportamientos necesarios para considerar un sistema inteligente. Tareas o problemas que se creían que solo podían ser resueltos por seres inteligentes, se ha demostrado que las RNA los pueden resolver, y muchas veces incluso mejor que lo seres humanos más expertos en el campo.

El gran objetivo de la IA es crear un sistema que pueda interactuar de manera inteligente en un entorno real. Para ello no es suficiente con las RNA y los algoritmos clásicos de entrenamiento, ya que estos utilizan arquitecturas diseñadas para el problema y datos etiquetados, representando la *narrow* IA. El aprendizaje por refuerzo, *Reinforcement Learning* (RL), trata de cubrir la necesidad de aprender a través de la interacción con el entorno, imitando la manera normal de aprender de los seres vivos. En el marco de trabajo del RL, existen tres componentes principales: el agente, el entorno, y el objetivo que se le marca al agente. Hoy en día, el agente entrenado con RL suele ser una RNA diseñada con técnicas de *Deep Learning*. Pero hay que señalar que el agente no recibe ningún tipo de instrucción que le enseñe cómo lograr su objetivo. Para lograr su objetivo el agente tiene que interactuar con el entorno, ya sea el mundo real o un entorno virtual. Para ello necesita realizar observaciones del mismo mediante algún tipo de sensor, real o artificial respectivamente. Estas observaciones representan información parcial del entorno, y en base a esta información el agente debe realizar acciones que maximicen de alguna manera las posibilidades de conseguir su objetivo.

Sin embargo, existen un gran número de complicaciones a la hora de desarrollar un sistema de estas características. En el campo de la robótica se ha comprobado que este objetivo es mucho más complicado de lo que se pensaba. Es un reto de ingeniería crear sistemas que se puedan desplazar e interactuar con el medio, el simple hecho de desplazarse o coger un objeto resultan tareas muy difíciles de realizar para un robot. Por tanto, la idea de desarrollar y evaluar posibles comportamientos inteligentes en los robots no es posible en estos momentos. Por esta razón, el campo de la IA ha comenzado a trabajar en entornos virtuales, lo cual reduce enormemente los costes y permite evaluar con mayor detalle los avances.

En los últimos años se ha trabajado mucho en el campo del RL, y los avances más importantes los ha realizado DeepMind, fundada por Demis Hassabis, Shane Legg y Mustafa Suleyman, en 2010. La empresa fue adquirida por Google en 2014 por alrededor de 500 millones de dólares. Esta empresa en lugar de desarrollar los entornos virtuales para entrenar a los sistemas mediante RL, decidió utilizar la fuente que más entornos virtuales

dispone, los videojuegos. Los videojuegos representan una opción perfecta, para desarrollar y evaluar los algoritmos de RL. Además de existir una fuente ilimitada de datos para el entrenamiento, no existe un sesgo en la evaluación, y normalmente los videojuegos cuenta con algún sistema de puntuación que permiten evaluar la actuación del agente en el entorno de manera objetiva. Además, los videojuegos han sido diseñados para ser jugados por humanos, por lo que la actuación del agente puede ser comparada con la de los humanos.

Para desarrollar los primeros modelos de RL, DeepMind seleccionó una colección de más de 50 juegos clásicos de la Atari 2600, entre los que están el conocido Space Invaders o el Breakout. El sistema de IA, denominada Deep Q-Network, solo recibe los píxeles como entrada y el objetivo de maximizar la puntuación, pero el sistema no tiene conocimiento de las reglas u objetivos, ni tampoco de los posibles movimientos en cada juego. Además, DeepMind se propuso que un que un mismo sistema debería jugar a todos los juegos. En el 2015 publicaron en Nature los resultados de este proyecto, demostrando que el sistema podía alcanzar un nivel muy superior a los humanos (Mnih et al., 2013).

Uno de los acontecimientos más importantes en el campo de la IA tuvo lugar en el año 2016 en Seúl, Corea del Sur, donde AlphaGo, un sistema informático desarrollado por DeepMind (Silver et al., 2016), consiguió vencer al juego oriental Go a Lee Sedo, 18 veces campeón del mundo y considerado el mejor jugador de la última década. El Go es un juego ancestral de origen chino que tiene más de 3.000 años de historia, y actualmente existen unos 40 millones de jugadores. Este evento tuvo un impacto cultural muy importante, alrededor de 280 millones de espectador, más de 35.000 artículos de prensa y la venta de tableros de Go se multiplicó por 10.

El juego del Go es extremadamente complejo, y existen aproximadamente 10^{170} posibles configuraciones del tablero, más que átomos en el universo conocido. Además, resulta imposible crear una función de evaluación que determine qué jugador está ganando en cada momento. Los expertos consideran que esta victoria se adelantó unos 10 años a las predicciones, ya que en el Go no es posible obtener una victoria por fuerza bruta, como en el caso del ajedrez, y es necesario cierta “intuición” para reducir el espacio de búsqueda del movimiento óptimo en cada momento. La victoria de Deep Blue frente a Gary Kasparov fue una proeza tecnológica desde el punto de vista computacional. Pero desde el punto de vista de la IA, Deep Blue no es comparable a Alpha Go porque varias razones. Además de la dificultad del propio juego, el Go es más complejo que el ajedrez. Los propios sistemas no son comparables, Deep Blue está diseñado para jugar única y exclusivamente al ajedrez,

mientras que Alpha Go tiene un diseño que le permite adaptarse a nuevos dominios con facilidad.

La arquitectura de AlphaGo cuenta con dos RNA, una *Policy Network* (Distribución de probabilidades sobre los movimientos) y una *Value Network* (Real numbers: 0 White - 1 Black). La *Policy Network* fue entrenada con miles de partidas de jugadores online, esta red trata de predecir qué movimiento hará el jugador en cada momento. La *Value Network* representa la función de evaluación, y se entrenó jugando 30 millones de veces contra sí misma, en cada partida sabe quién ganó y todos los movimientos intermedios.

AlphaGo se diseñó para aprender de la experiencia, sin tener ninguna idea preconcebida del juego o la estrategia. Esto provocó que su estilo de juego sea único y rompa con las convicciones en ciertos aspectos. Un ejemplo ilustrativo es el ya famoso movimiento 37 de la segunda partida frente a Lee Sedo. En el Go existe una convención que considera que, en los comienzos de una partida, si un jugador sitúa la pieza en las tres líneas más próximas al borde del tablero, su objetivo es conquistar esa parte del borde del tablero. Sin embargo, si el jugador sitúa la pieza en la cuarta línea, su objetivo es mantener sus opciones de conquistar esa zona del centro. En la historia del Go profesional, no se concibe situar la pieza en la 5ª línea, y esto fue justo lo que realizó AlphaGo en el movimiento 37 de la segunda partida, lo cual cambió el curso de la partida. Este movimiento ha pasado a la historia del Go y ha provocado que muchos jugadores profesionales de Go se replanteen su manera de jugar.

A finales del año 2017, DeepMind publicó una nueva versión denominada Alpha Go Zero, que supuso una nueva revolución en el campo del RL (Silver et al., 2017). El sistema se denominó Alpha Go Zero, ya que el sistema fue entrenado sin ningún conocimiento humano, ni para aprender de partidas jugadas entre humanos, ni jugando contra humanos para mejorar. Alpha Go Zero se entrenó jugando contra sí mismo, y ha sido capaz de vencer por 100-0 a la versión anterior de Alpha Go.

Como se ha comentado, la IA y la Neurociencia han tenido una historia larga y entrelazada, sin embargo, en los últimos tiempos la comunicación y colaboración entre estos dos campos ha dejado de ser tan frecuente. En sus orígenes estos dos campos estaban muy relacionados y muchos investigadores trabajaban en ambas disciplinas.

A lo largo de la historia de la IA numerosos avances han estado inspirados en descubrimientos de la Neurociencia como por ejemplo la estructura jerárquica de representación de la información en las redes de *Deep Learning*, o la técnica de *Dropout* inspirada en la capacidad del cerebro de tolerancia a fallos, o falta de elementos de

procesado, o errores. Pero en la actualidad, parece que los investigadores en IA no tienen tan en cuenta los descubrimientos de Neurociencia como fuente de ideas para desarrollar nuevos algoritmos, probablemente debido a que se han centrado en explotar y desarrollar los éxitos del *Deep Learning* y el RL. Pero en el fondo muchos de los desarrollos más recientes en IA han estado inspirados por ideas y descubrimientos de Neurociencia, algunos de los ejemplos más importantes son:

- **Atención:** El cerebro no es un sistema global uniforme, sino que se compone de diversos subsistemas que están especializados en determinadas tareas, procesando de manera simultánea multitud de estímulos, y siendo capaz de centrar su atención en los estímulos más relevantes en cada momento (Hassabis et al., 2017). Esta idea se ha trasladado al campo de la IA en los últimos tiempos. Por ejemplo, las redes de convolución clásicas procesaban todos los píxeles de las imágenes de entrada de manera simultánea. Recientemente se han desarrollado nuevos modelos capaces de centrarse en las regiones más importantes de la imagen, ignorando el resto de píxeles. Estas redes de convolución son capaces de realizar tareas de reconocimiento de múltiples objetos en una misma imagen (Ba et al., 2014; Xu et al., 2015). Otro ejemplo es el caso de las RNA que pueden centrar los mecanismos de atención en su propio estado interno, seleccionando la información que debe ser leída en su estado interno. Esta técnica ha ayudado a conseguir grandes avances en el campo de la traducción (Bahdanau et al., 2014) y las tareas de memoria y razonamiento (Graves et al., 2016). Por último, los mecanismos basados en la atención han servido para desarrollar modelos generativos que permiten crear imágenes sintéticas realistas basándose en su estado interno (Hong et al., 2016)
- **Memoria episódica:** Una parte muy importante de la Neurociencia se centra en los diferentes aspectos de la memoria y su relación con la inteligencia. Existen diferentes tipos de memoria y uno de los más importantes es la memoria episódica la cual permite almacenar nuestras experiencias de manera rápida (Hassabis et al., 2017). Este tipo de memoria está asociada con el lóbulo temporal medio, en especial la región del hipocampo. En este aspecto de la memoria se basa uno de los hitos recientes en IA, la DQN desarrollada por DeepMind para jugar a videojuegos de la Atari 2600 (Mnih et al., 2013). Una de las claves del éxito de la DQN era que podía almacenar un subconjunto de los datos del entrenamiento, para luego utilizarlos de manera offline y aprender de los errores y aciertos de los eventos pasados, permitiendo optimizar el aprendizaje. Además, se demostró que se

producía una mejora sustancial de los resultados cuando se favorecía la utilización offline de los eventos con una mayor recompensa (Schaul et al., 2015), al igual que se ha demostrado que ocurre en el hipocampo que parece favorecer los eventos que producen mayores niveles de refuerzo (Singer & Frank, 2009).

- **Memoria de trabajo:** Una de las características más importantes de la inteligencia humana es la habilidad para mantener y manipular información de manera activa, es lo que se conoce como memoria de trabajo. Este aspecto de la memoria también ha sido imitado para desarrollar arquitecturas de RNA, las denominadas *Long-Short-Term Memory Networks* (LSTM) (Hochreiter & Schmidhuber, 1997), las cuales pueden almacenar la información en un estado fijo. En las LSTM clásicas las funciones de control y almacenamiento estaban íntimamente ligadas, pero los modelos más recientes presentan módulos separados para el control y el almacenaje (Graves et al., 2016).
- **Aprendizaje continuo:** El cerebro es capaz de aprender de manera continua desde el nacimiento hasta el fallecimiento. Por lo que puede aprender nuevas tareas sin olvidar las que ya dominaba. Sin embargo, las RNA han presentado grandes dificultades para imitar esta característica del cerebro, sufriendo lo que se denominó “olvido catastrófico” (French, 1999; McClelland et al., 1995). Esto se debe a que las RNA dominan una tarea alcanzando un estado estable de los parámetros, pero cuando se intenta que aprendan una nueva tarea se modifican los parámetros cambiando su configuración y olvidando la tarea anterior.

Pero aún existen muchos aspectos del cerebro y descubrimientos de Neurociencia que pueden ser explotados por la IA, algunos de los más relevantes son:

- **Autoorganización de la arquitectura de las redes neuronales:** El cerebro humano ha evolucionado gracias a la selección natural a lo largo de millones de años, lo que ha optimizado su arquitectura y capacidad de adaptación a los cambios, presentando tolerancia a errores o fallos en su estructura. Como ya se ha comentado, una de las técnicas clave del *Deep Learning* se basa en esta capacidad del cerebro, el *Dropout* (Srivastava et al., 2014) trata de imitar la tolerancia del cerebro a los posibles fallos o muerte de las neuronas. Pero a pesar de esta técnica, las RNA aún necesitan del conocimiento experto para diseñar su arquitectura. La flexibilidad de la arquitectura de las redes es uno de los aspectos clave para poder crear sistemas adaptables a nuevas condiciones y tareas.

- **Eficiencia energética:** Las RNA necesitan de grandes recursos computacionales por lo que consumen una gran cantidad de energía, sin embargo, el cerebro humano es uno de los sistemas más eficientes, consumiendo unas 400 Kcal al día, lo que representa unos 25 Watts de energía. Los avances en GPUs han reducido el consumo considerablemente y en la actualidad se trabaja con chips neuromórficos (L.A. Pastur-Romay et al., 2017) para crear sistemas más eficientes que imiten la estructura del cerebro.
- **Comprensión intuitiva del mundo físico:** El cerebro es un órgano que ha evolucionado en un entorno natural, por tanto, su estructura está optimizada para interactuar en este medio físico recibiendo cierto tipo de estímulos. Esto hace que el cerebro posea unas capacidades innatas que le permite aprender fácilmente conceptos relacionados con el mundo físico como son el espacio, el tiempo, la objetividad o los números (P. W. Battaglia et al., 2013). Estos conceptos son muy complicados de desarrollar en los sistemas de IA, pero se ha comenzado a avanzar para intentar solventar estos retos. Por ejemplo, nuevos modelos de *Deep Learning* son capaces de interpretar escenas y descomponer los objetos de manera individual y establecer las relaciones que existen entre ellos (P. Battaglia et al., 2016).
- **Algoritmos de aprendizaje:** El cerebro a lo largo de la evolución ha establecido unos sistemas de recompensa y evaluación que le permiten realizar un aprendizaje autónomo, aunque también exista un aprendizaje familiar y cultural que establezca ciertas recompensas para favorecer determinados comportamientos. Por su parte, en IA es necesario establecer por parte del investigador un sistema de evaluación de los resultados. Las RNA clásicas necesitan de una gran cantidad de datos etiquetados, en muchos casos millones de ejemplos, para ser entrenadas. Aunque con el RL se suple en parte esta carencia permitiendo que la red aprenda de la interacción con el medio, aún es necesario establecer algún objetivo para que el agente pueda interactuar con el entorno maximizando esta variable objetivo.
- **Aprendizaje eficiente:** El cerebro necesita muy poca información para crear una representación de un concepto y memorizarlo, por ejemplo, en la mayoría de los casos nos basta con ver una sola vez un objeto para reconocerlo la próxima vez que lo vemos, es lo que se denomina como *one-shot learning* (Gallistel & King, 2009). Este es uno de los aspectos más difíciles de implementar en las RNA, ya que existe un balance entre la flexibilidad del sistema para aprender nueva información y la estabilidad para no olvidar la información ya aprendida (Hassabis et al., 2017). Pero

existen algunos avances en IA mediante técnicas que emplean tanto modelos probabilísticos (Lake et al., 2015), como modelos generativos (Rezende et al., 2016). Otra línea de investigación intentar desarrollar redes que sean capaces de “aprender a aprender”, lo que se conoce como deuteroprendizaje, adquiriendo conocimiento en nuevas tareas mediante la experiencia previa en tareas semejantes, tratando de llevar a cabo el *one-shot learning* (Santoro et al., 2016), y acelerando el aprendizaje en RL (Wang et al., 2016).

- **Transferencia del aprendizaje:** El cerebro tiene la capacidad de transferir y extrapolar el conocimiento de un dominio a otro, es lo que se conoce como aprendizaje por transferencia, o *Transfer Learning* (Holyoak & Thagard, 1997). Existen trabajos en IA que desarrollaron modelos capaces de realizar inferencias *zero-shot* sobre formas nuevas no presentes en el entrenamiento basándose en representaciones compositivas (Higgins et al., 2016).
- **Imaginación y planificación:** La habilidad de imaginar y planificar resulta esencial para el éxito en la mayoría de las tareas. La IA intenta imitar estas capacidades del cerebro, por ejemplo, el AlphaGo emplea un árbol de búsqueda de Montecarlo (Browne et al., 2012) para tratar de predecir el valor futuro de una función de recompensa (Silver et al., 2016).
- **Representación conceptual:** El cerebro se compone de diferentes regiones especializadas y estas a su vez tienen circuitos o redes que representan determinados objetos, conceptos, ideas o personas. En el córtex entorrinal de los mamíferos se ha descubierto que existe un *grid code* con campos receptivos que representan y cubren el espacio físico mediante patrones hexagonales (Rowland et al., 2016).
- **Análisis virtual del cerebro:** Uno de los problemas de los modelos de *Deep Learning*, es que son cómo cajas negras, debido a su gran complejidad resulta casi imposible comprender su funcionamiento interno y como se producen los procesos de aprendizaje y decisión. La IA puede basarse en las técnicas de Neurociencia para intentar de comprender mejor el funcionamiento interno de los sistemas. En esta línea se ha realizado cierto avance, por ejemplo, técnicas empleadas en Neurociencia de reducción de dimensionalidad se han utilizado para analizar RNA (Zahavy et al., 2016).

3.1.2. Modelos Neuronales Computacionales

Los modelos de la neurona presentados hasta el momento son simplificaciones muy básicas pero muy efectivas computacionalmente. Las neuronas artificiales de las RNA funcionan de una manera muy sencilla, suman las entradas que reciben multiplicándolas por los pesos de las conexiones y aplican una función de activación para producir una salida. Pero existe una gran variedad de modelos neuronales, aunque las RNA emplean los modelos más sencillos. Los modelos de mayor complejidad presentan una diferencia clave y es que a la hora de modelar el funcionamiento de la neurona se tiene en cuenta el nivel de activación de la neurona y el fenómeno de los potenciales de acción, o *spike*. Las redes que se crean con estos modelos neuronales se les denomina *Spiking Neural Networks* (SNN) y tiene un mayor realismo biológico (Gerstner & Kistler, 2002; Maass & Bishop, 1998). En la tabla 2 se puede observar un resumen de los más importantes en orden de complejidad crecientes, expresándose el coste computacional de simular una neurona en operaciones de coma flotante por segundo, *Floating Points Operations Per Second* (FLOPS).

Tabla 2. Modelos computacionales de las neuronas y los FLOPS necesarios para simular una neurona.

Modelos de Neurona	FLOPS por neurona
Integrar y disparar (<i>Integrate and fire</i>)	5
Integrar y disparar cuadrático (<i>Quadratic integrate and fire</i>)	7
Integrar y disparar o ráfaga (Integrate and fire or burst)	13
Modelo de Izhikevich	13
Modelo de Hodgkin-Huxley	1200

El modelo más sencillo empleado en las SNN es el de “Integrar y disparar”, *Integrate and fire* y solo emplea 5 FLOPS. Aunque el modelo más óptimo para simular un gran número de neuronas de manera realistas es el Modelo de Izhikevich (Izhikevich, 2003), el cual es una versión mejorada del modelo *Leaky integrate and fire* (Burkitt, 2006). Por último, el modelo de Hodgkin-Huxley, propuesto por Alan Lloyd Hodgkin y Andrew Huxley en 1952 (Hodgkin & Huxley, 1952) y que les valió para ganar el Premio Nóbel en fisiología y medicina en 1963, presenta mayor complejidad y realismo biológico ya que simula las corrientes de Ca⁺, K⁺ y Na⁺. El modelo de Hodgkin-Huxley se puede aplicar a modelos

3D de las neuronas, pero los cálculos computacionales aumentan exponencialmente volviéndose demasiado complejos ya que se necesita integrar el espacio y el tiempo. Para simplificar los cálculos se emplean modelos de compartimentos (Herz et al., 2006) y la Teoría del cable, propuesta por Wilfrid Rall (Rall, 1989), para simular la transmisión del impulso nervioso a través de las dendritas y axones.

Hay que señalar que los modelos neuronales se pueden implementar tanto en *software* como en *hardware*, en la figura 13 se presenta un esquema de la clasificación de los modelos en función de su implementación. Hoy en día los modelos realistas solo pueden ser implementados en software. Por otra parte, los modelos *hardware* que se inspiran en el funcionamiento de las neuronas, se denominan chips neuromórficos. Los modelos neuronales en *hardware* se pueden clasificar en tres grupos dependiendo del tipo de computación que realicen: modelos digitales, modelos analógicos y modelos híbridos. A continuación, se presentan las características más importantes de cada modelo y los proyectos más relevantes que se están llevando a cabo actualmente.

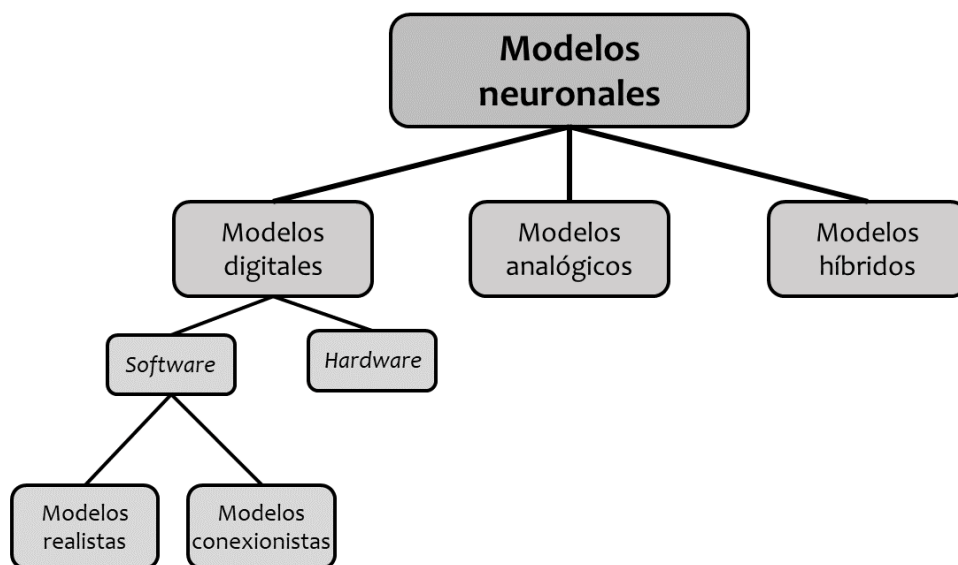


Figura 13. Diagrama con la clasificación de los modelos neuronales: digitales, analógicos e híbridos. Los modelos digitales se pueden dividir en los que se implementan en software y en hardware, a su vez los modelos en software se dividen en modelos realistas y conexionistas.

3.1.2.1. Modelos Digitales Software

Los modelos realistas se implementan generalmente en *software* ya que tratan de simular el cerebro con gran detalle, teniendo en cuenta la estructura de las células, los canales iónicos, etc. Estas simulaciones tienen utilidad para poder estudiar el cerebro desde una perspectiva neuroquímica, por ejemplo, analizando el efecto de un determinado fármaco, realizando modelos de neurodegeneración o daño cerebral, etc. También se puede estudiar el cerebro a nivel de las comunicaciones entre neuronas o regiones cerebrales, a nivel molecular para tratar de comprender cómo se crean y modifican las sinapsis, y a nivel electrofisiológico para simular la generación de potenciales de acción y corrientes iónicas mediante ecuaciones matemáticas.

El proyecto más ambicioso basado en modelos realistas es el Human Brain Project (HBP). La Unión Europea aprobó en el año 2013 el HBP como parte de la convocatoria FET Flagship con un presupuesto de 1.000 millones de euros y supone una continuación del Blue Brain Project, el cual comenzó en el año 2005 en la École Polytechnique Fédérale de Lausanne (EPFL) en Suiza y el investigador principal es Henry Markram. El objetivo principal del proyecto es simular de manera realista un cerebro humano completo en el año 2023. Este objetivo resulta muy ambicioso y atractivo, pero muchos consideran irrealista los plazos debido a las limitaciones tecnológicas. Para alcanzar los objetivos el HBP necesita una gran infraestructura computacional ya que emplean los modelos neuronales basados en los desarrollados por Hodgkin-Huxley pero de mayor complejidad. Para ello cuenta con la colaboración de IBM y actualmente trabajan con supercomputadoras Blue Gene/Q, aunque dentro del HBP se incluyen varios subproyectos para desarrollar chips neuromórficos que puedan soportar los modelos realistas con los que trabajan. En concreto, el subproyecto *Neuromorphic Many-Core system* desarrolla hardware digital, el chip SpiNNaker (Khan et al., 2008) y el subproyecto *Neuromorphic Physical Model* trabaja con hardware analógico, el proyecto BrainScaleS (Schemmel et al., 2010).

Existe otro proyecto muy interesante que emplea modelos digitales software para crear un sistema con una arquitectura semejante al cerebro, es el proyecto SPAUN (*Semantic Pointer Architecture Unified Network*) (Eliasmith et al., 2012). Este proyecto emplea modelos *Leaky Integrate-and-Fire* y simula 2.5 millones de neuronas y 60 millones de sinapsis. SPAUN tiene el objetivo de desarrollar un sistema capaz de realizar tareas perceptuales, cognitivas y motoras en el mundo real a través de un brazo robótico. Para ello el sistema se compone de subsistemas que simulan las funciones de diferentes regiones cerebrales.

A pesar de la gran utilidad que tienen los modelos realistas para la Neurociencia, estos modelos presentan una utilidad limitada para ayudar a comprender el funcionamiento global del cerebro ya que al crear modelos con tanto nivel detalle resultan computacionalmente muy costosos, limitando el número de neuronas que se pueden simular actualmente. Para evaluar nuevas teorías o hipótesis sobre el procesamiento de la información resulta más interesante crear modelos sencillos *in silico* y centrarse en los ya mencionados niveles superiores de Marr y Poggio (Marr & Poggio, 1976) ya que no es necesario tener en cuenta cómo se implementan los procesos en un sustrato biológico.

3.1.2.2. Modelos Digitales Hardware

Los modelos digitales emplean la computación binaria y tratan de crear arquitecturas hardware que simulan a las neuronas. Para ello emplean circuitos integrados de aplicación específica (ASICs), *Field Programmable Gate Array* (FGPA), o nuevas arquitecturas específicas basados en tecnologías 3D.

El HBP cuenta con el subproyecto SpiNNaker en el que se ha desarrollado un chip con 18 procesadores ARM968. Este chip es flexible y soporta diferentes modelos neuronales, tanto las neuronas artificiales de las RNA como los modelos de Izhikevich o *Leaky integrate-and-fire* de las SNN, y la comunicación se realiza con la técnica de Address Event Representation (AER). Cada CPU puede simular unas 1.000 neuronas Izhikevich o *Leaky integrate-and-fire*. En el año 2015 se creó una cabina con 5760 chips capaz de simular 100 millones de neuronas con unas 1.000 sinapsis por neurona. El SpiNNaker presenta una gran versatilidad y se ha utilizado para diversas aplicaciones, por ejemplo, interfaz con el sistema SPAUN, reconocimiento de caracteres con RNA, visión artificial con redes neuronales de convolución, interfaz con sensores AER, integración con plataformas robóticas, etc. El SpiNNaker pretenden servir como plataforma de simulación en el HBP y ya se han simulado circuitos corticales con 10.000 neuronas sencillas en tiempo real. La extensión final sería un sistema con 10 cabinas y un total de 1 millón de procesadores ARM (L.A. Pastur-Romay et al., 2017).

Por otra parte, uno de los proyectos más avanzados en este campo es la iniciativa DARPA SyNAPSE (*System of Neuromorphic Adaptive Plastic Scalable Electronics*) y está liderado por IBM y el investigador principal es Dharmendra Modha (Merolla et al., 2014). En el proyecto se ha desarrollado el chip TrueNorth, el cual presenta una arquitectura muy eficiente no Von Neumann, ya que no constan de una unidad de procesado y una de memoria, sino que el procesado y almacenamiento de la información se realiza en el mismo elemento, la

neurona. Para ello emplean una versión mejorada de los modelos de la neurona *Leaky integrate-and-fire*, y un solo chip puede simular 1 millón de neuronas y 4.000 millones de sinapsis en tiempo real. Recientemente se ha creado un sistema que combina 128 chips para simular un total de 128 millones de neuronas. Las neuronas se conectan mediante la técnica de AER. En este proyecto se ha desarrollado un paradigma de programación basado en *Corelets*, que represente un programa en el chip TrueNorth que se representa únicamente por las entradas y salidas del chip, mientras que los detalles internos del estado del chip se encapsulan. A partir de las combinaciones de estos *Corelets*, se pueden crear programas más complejos y librerías. Estas librerías cuentan con funciones algebraicas, lógicas, funciones temporales, filtros lineales, convoluciones de datos 1D, 2d y 3D, transformadas de Fourier, etc. Estas funciones permiten crear Redes Neuronales Artificiales de Convolución, *Restricted Boltzmann Machines*, *Liquid State Machine* como los modelos ocultos de Markov, o *Hidden Markov Model*, etc. Existe la posibilidad de mapear, en sentido matemático, una RNA a la arquitectura de TrueNorth para así poder implementar los pesos en el chip sin necesidad de realizar el entrenamiento. Esto permite ahorrar tiempo y costes en el desarrollo de nuevas aplicaciones. Hoy en día ya existen multitud de aplicaciones para TrueNorth, por ejemplo: reconocimiento de voz, detección de movimiento, prevención de colisiones, etc.

3.1.2.3. Modelos Analógicos

En los modelos neuromórficos analógicos no existe separación entre el software y el hardware ya que la configuración del propio hardware es la que realiza las operaciones y se puede modificar a sí misma. Por tanto, no emplean el sistema binario, sino que realizan una computación mediante señales analógicas, es decir emplean valores continuos para procesar la información basados en los valores de las resistencias de los chips. Esto permite que el procesamiento sea mucho más eficiente, pero a la vez lo hace susceptible al entorno y al proceso de fabricación.

Otro subproyecto del HBP trata de desarrollar esta tecnología, es el denominado BrainScaleS (*Brain-inspired Multiscale Computation in Neuromorphic Hybrid System*) (Schemmel et al., 2012). En este proyecto se ha desarrollado el chip HICANN (*High Input Count Neural Network*) que puede simular en un *Wafer* 200.000 neuronas del tipo *Adaptive exponential integrate-and-fire* y 50 millones de sinapsis 10.000 veces más rápido que el tiempo real. El objetivo del proyecto para 2022 es crear un sistema con 5.000 *Wafers* para poder simular hasta 5 billones de neuronas.

3.1.2.4. Modelos Híbridos

Los modelos híbridos *hardware* tratan de aprovechar lo mejor de los modelos digitales y los analógicos, normalmente el procesado de la información es analógica y la comunicación es digital. Un ejemplo de esta tecnología es el chip Neurogrid desarrollado por la universidad de Stanford (Benjamin et al., 2014). Este chip es capaz de simular 1 millones de neuronas y 1 billón de sinapsis en tiempo real consumiendo tan solo 3 Watts. El Neurogrid utiliza una simulación de las neuronas en dos niveles, por una parte, se utiliza un modelo *Quadratic integrated-and-fire* para el cuerpo neuronal y las dendritas se simulan con 4 canales de Hodgkin-Huxley. Este sistema simula todos los elementos neuronales con circuitos electrónicos compartidos excepto el cuerpo neuronal.

Los chips neuromórficos resultan una opción muy interesante para desarrollar hardware que resulte eficiente para las RNA, pero económica y técnicamente resulta muy complicado poder implementar y evaluar nuevos algoritmos basados en el papel de los astrocitos en el procesamiento de la información. En el caso de los modelos realistas, resultan útiles para estudiar aspectos concretos del funcionamiento de los astrocitos, su comunicación con otros astrocitos y sus interacciones con las neuronas, pero no permiten analizar y evaluar la influencia concreta que puedan tener los astrocitos en el procesamiento de la información. Por ello, en esta tesis doctoral se ha decidido trabajar con los modelos conexionistas, basándose en las RNA para implementar los nuevos algoritmos astrocíticos, en primer lugar, con modelos *shallow*, empleando Redes Neurogliales Artificiales RNGA, y posteriormente con modelos más complejos de *Deep Learning*, desarrollando las *Deep Artificial Neuron-Astrocyte Networks* (DANAN) (Lucas Antón Pastur-Romay et al., 2016).

3.1.3. Redes Neuronales Artificiales

Los modelos neuronales empleados en las RNA son los más sencillos lo cual permite crear redes muy grandes y consumiendo pocos recursos computacionales. En la figura 14 se puede ver el funcionamiento de una neurona artificial, la cual recibe unos valores de entrada (X_1, X_2, \dots, X_n), que se multiplican por el peso de cada conexión y se realiza un sumatorio de todos los productos para calcular su valor neto (*net*). Normalmente los pesos tienen asociado una constante denominada *bias*, o sesgo, que facilita el proceso de entrenamiento. A continuación, se aplica una función de activación a este valor neto para producir el valor de salida de la neurona (y).

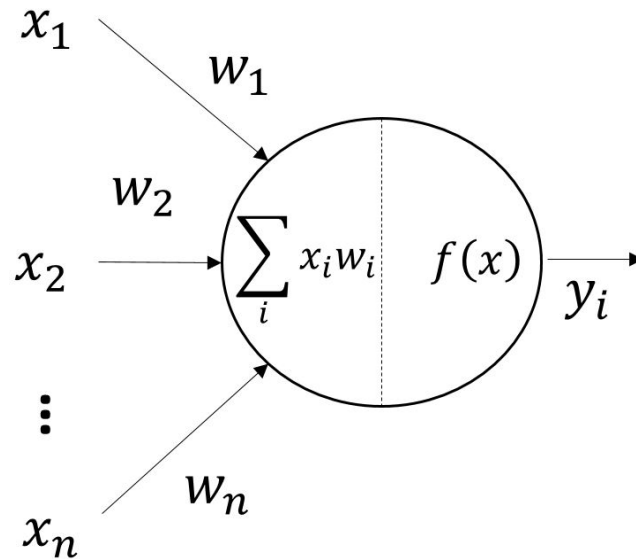


Figura 14. Modelo simplificado de una neurona artificial.

Existen diferentes funciones de activación, la más simple es la lineal que multiplica el valor neto por una constante para producir la salida. Pero actualmente la función de activación más utilizada es la *Rectified Linear Unit*, o ReLU, (Nair & Hinton, 2010) ya que computacionalmente es muy eficiente y a la vez permite realizar transformaciones no lineales. Existen otras funciones como la sigmoidea (figura 15), umbral, hiperbólica-tangente, etc.

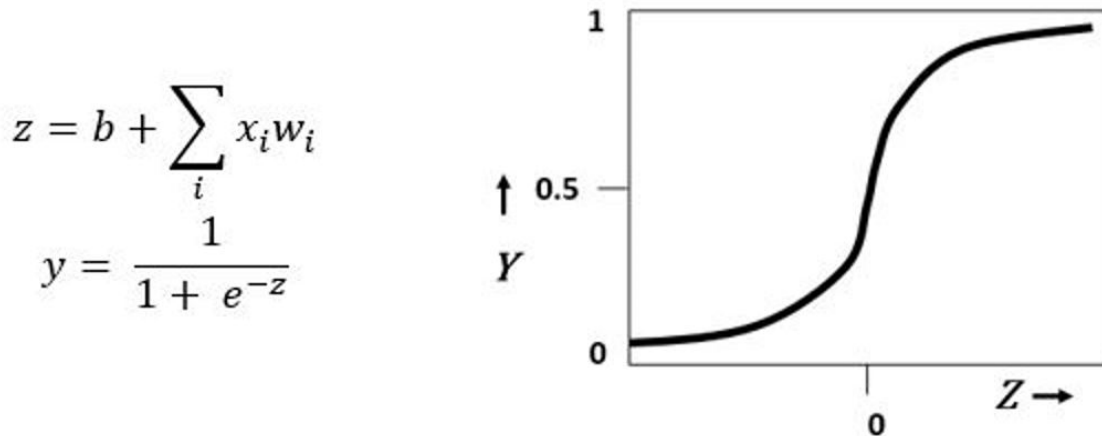


Figura 15. Función de activación sigmoidea, a la izquierda su fórmula y a la derecha la representación gráfica.

Las RNA están formadas generalmente por una capa de entrada, una capa de salida y un número variable de capas ocultas (figura 16). Los modelos más sencillos son las redes totalmente conectadas hacia adelante, es decir presentan conexiones hacia todos los elementos de la capa siguiente. Pero las neuronas también se pueden conectar a neuronas

de su misma capa o de las capas anteriores. Además, pueden presentar conexiones sólo hacia algunas neuronas de las otras capas.

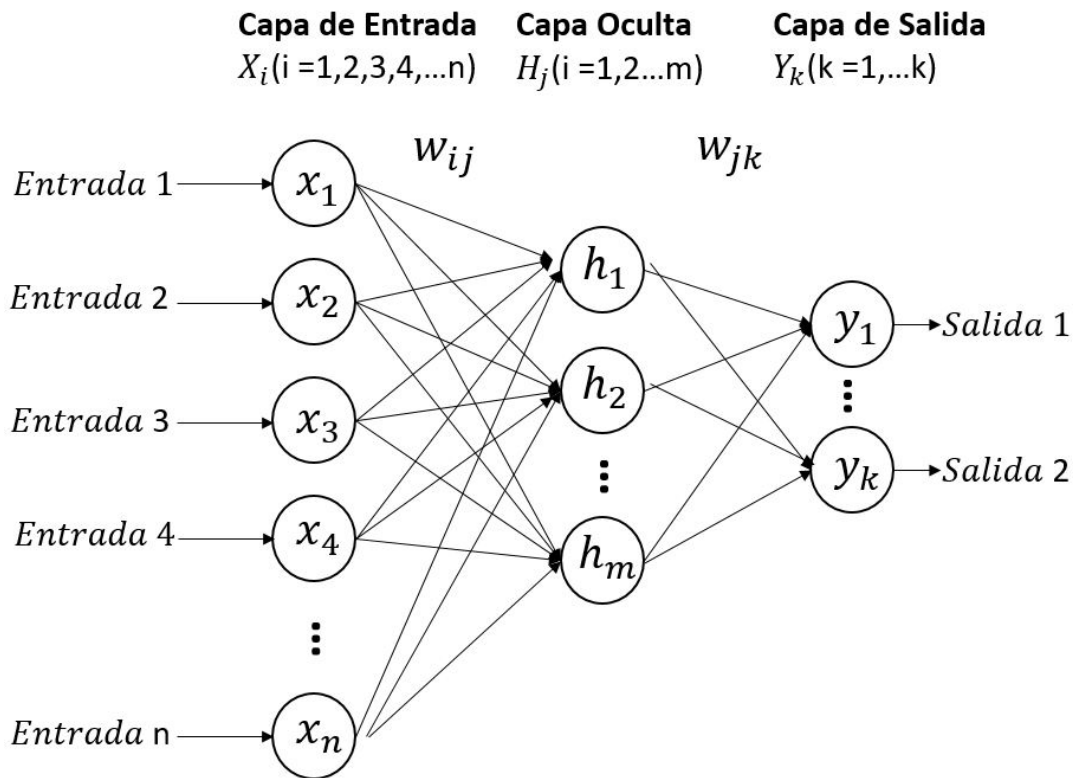


Figura 16. Estructura básica de una RNA.

Las RNA se pueden clasificar en cuatro grandes grupos: RNA totalmente conectadas hacia adelante, RNA convolucionales, RNA recurrentes y las RNA con memoria externa. Las RNA convolucionales se caracterizan por tener capas de convolución, las cuales solo están conectadas con una región de la capa anterior. En el caso de las RNA recurrentes, poseen conexiones entre neuronas de la misma capa o de capas anteriores. Por último, las RNA con memoria externa poseen neuronas artificiales con la función exclusiva de almacenar la información. Aunque esta clasificación no es del todo correcta, dado que es una tipología y hoy en día existen redes que combinan los diferentes tipos.

Las RNA han demostrado en los últimos años en el enorme potencial que tienen y muchas aplicaciones que usamos en el día a día están desarrolladas utilizando técnicas de *Deep Learning*. Como se ha comentado, estos avances también han servido para comprobar que el funcionamiento de las RNA tiene cierta semejanza con el cerebro. Por ello, en esta tesis se pretende desarrollar modelos de IA más realistas y que incluyan los dos tipos de células del SN con capacidad de procesar la información: las neuronas y los astrocitos. A continuación, se presentan los antecedentes que han servido como base para desarrollar esta tesis.

4. Antecedentes y Motivación

La idea de que las células gliales, en especial los astrocitos, participan en el procesamiento de la información ha servido como base para desarrollar una gran variedad de modelos computacionales, desde los más realistas hasta los más abstractos. Una de las mayores dificultades en el estudio de la sinapsis tripartita es la diversidad que existe en las interacciones entre las neuronas y los astrocitos. En la figura 17 se puede observar un esquema simplificado de la sinapsis tripartita, la neurona presináptica se comunica mediante neurotransmisores tanto con la neurona postsináptica como con el astrocito perisináptico. El astrocito capta los neurotransmisores que producen unas corrientes de Ca^{+2} locales que provocan la liberación de gliotransmisores que pueden afectar tanto a la neurona presináptica como postsináptica. Las corrientes de Ca^{+2} locales también pueden generar corrientes de Ca^{+2} globales que se propaguen a otros astrocitos vecinos.

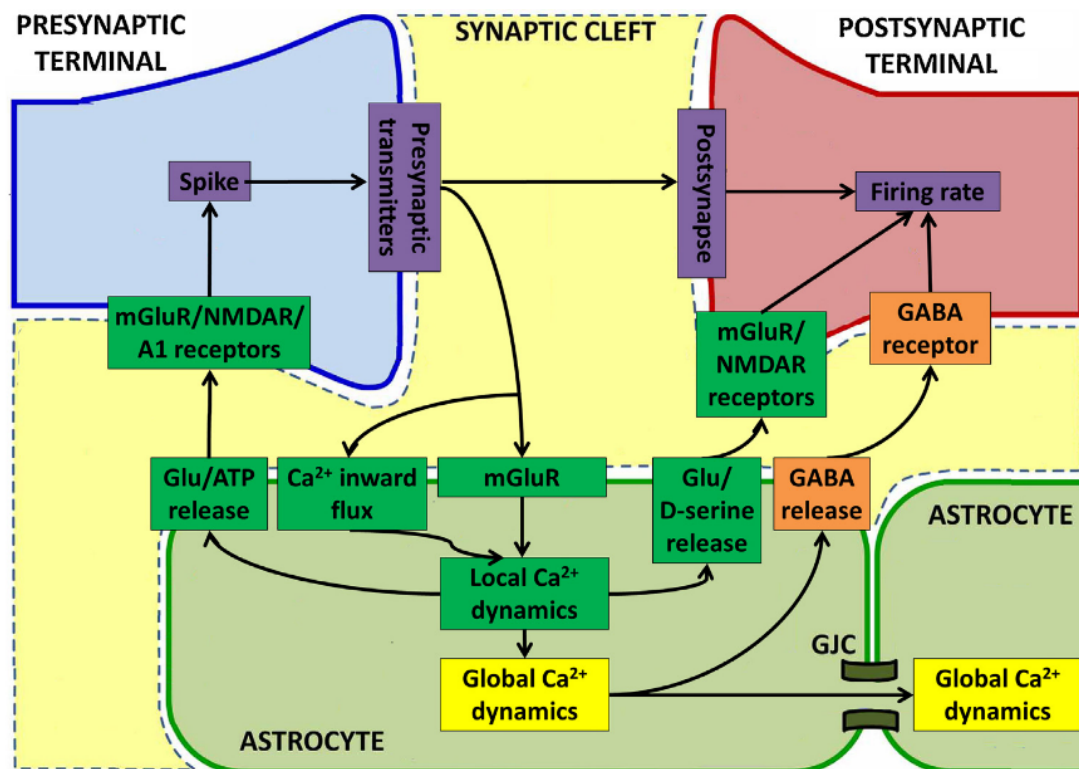


Figura 17. Esquema simplificado de la sinapsis tripartita con las múltiples interacciones que tienen lugar entre las neuronas y los astrocitos. En azul se representa el terminal de la neurona presináptica, en rojo el terminal de la neurona postsináptica, en verde el astrocito perisináptico y en amarillo el espacio sináptico (Oschmann et al., 2017).

4.1. Modelos gliales

A continuación, se presentan los diferentes tipos de modelos computacionales con células gliales que representan los antecedentes del trabajo desarrollado en esta tesis. Estos modelos se dividen en tres grupos: modelos gliales en Neurociencia Computacional, modelos gliales en chips neuromórficos y modelos gliales en IA. El trabajo de esta tesis se centra en los modelos gliales en IA, pero los modelos de Neurociencia Computacional y los chips neuromórficos han servido como ejemplo de los posibles comportamientos de las células gliales que pueden implementarse en los modelos computacionales.

4.1.1. Neurociencia Computacional

En el campo de la Neurociencia Computacional se han creado modelos realistas a varias escalas y de diferentes procesos, en los cuales se representan biofísica y bioquímicamente las células. Estos modelos sirven para estudiar el papel de los astrocitos tanto en el cerebro sano, como su rol en diferentes enfermedades. En los últimos años, los trabajos en este campo se han centrado en las funciones que desempeñan los astrocitos en el procesamiento de la información y la plasticidad. Siguiendo la clasificación realizada por F. Oschmann *et al* (Oschmann et al., 2017), los modelos realistas de los astrocitos se dividen en cuatro niveles de complejidad creciente:

- **Modelos de astrocitos:** Los modelos de células gliales individuales han servido para estudiar la creación y propagación de las señales de Ca^{+2} , así como el efecto que tienen en estas señales de Ca^{+2} los cambios en la concentración de diferentes agonistas. Otros modelos se centran en la emisión de gliotransmisores por parte de los astrocitos, los cuales pueden afectar tanto a la neurona presináptica como a la neurona postsináptica. En los experimentos *in vitro* e *in vivo* se ha comprobado que los astrocitos pueden emitir un gliotransmisores, los cuales son muy semejantes a los neurotransmisores, y que pueden ser captados por las neuronas y provocar diferentes respuestas (De Pittà et al., 2012; Postnov et al., 2008).
- **Modelos de la sinapsis tripartita:** La primera prueba de la transferencia de información entre neuronas y astrocitos se obtuvo gracias a los cambios observados en la concentración extracelular de iones provocados por la actividad neuronal (Oschmann et al., 2017), existen diferentes modelos que tratan de simular estos comportamientos (Nadkarni & Jung, 2007; Wade et al., 2011). En estos modelos se representan un astrocyto y un pequeño número de neuronas, normalmente una o dos. En los momentos de alta actividad neuronal, se disparan un gran número de

potenciales de acción lo que provoca una acumulación de potasio en el espacio extracelular. Este aumento de potasio provoca la despolarización de los astrocitos y la apertura de canales de Ca^{+2} dependientes del voltaje. Otros modelos se centran en el estudio de las dinámicas de potasio ya que, durante los momentos de actividad neuronal, los astrocitos acumulan el potasio extracelular (Øyehaug et al., 2012). También existen modelos que estudian el proceso de comunicación entre los astrocitos y las neuronas por medio de gliotransmisores. Por último, algunos trabajos tienen un objetivo más global y crear modelos para estudiar el papel de los astrocitos en la potenciación/depresión a largo plazo (Wade et al., 2011).

- **Modelos de redes de astrocitos:** Los modelos de redes de astrocitos se centran en la comunicación entre astrocitos por medio de ondas de Ca^{+2} , modelizando desde unos pocos astrocitos a cientos de ellos. Existen dos mecanismos para la propagación de estas ondas, una señalización intracelular mediante IP_3 y una señalización extracelular mediante ATP y glutamato.
- **Modelos de redes de astrocitos y neuronas:** Los modelos de redes de astrocitos y neuronas son los que presentan mayor complejidad y para ello simulan un mayor número de neuronas y astrocitos.

4.1.2. Modelos gliales en Chips Neuromórficos

Actualmente existen varios grupos de investigación que trabajan en el desarrollo de chips neuromórficos en los que se implementan modelos de neuronas y astrocitos. Desde el año 2011 (Ghaderi et al., 2011), en la Universidad de Southern California se lleva a cabo el proyecto *Biomimetic Real-Time Cortex* (BioRC) en el cual se ha diseñado el primer chip neuromórfico CMOS que simula los microdominios de la glía (Joshi et al., 2011) e incluye varias neuronas interconectadas en pequeñas redes. Los astrocitos influyen en la actividad de las neuronas estimulando los disparos de potenciales de acción. Sin la actuación de los astrocitos, las neuronas postsinápticas no tendrían el potencial suficiente para disparar. El circuito representa un modelo básico de las interacciones entre las neuronas y los astrocitos, pero se incluyen tanto neurotransmisores como gliotransmisores, y las variaciones en las concentraciones de calcio que se inducen en los astrocitos. En el proyecto BioRC se han conseguido simular diferentes comportamientos observados en los laboratorios de Neurociencia. Por ejemplo, la recepción de neurotransmisores por parte de los astrocitos (Yilda Irizarry-Valle et al., 2013), o las corrientes lentas en las neuronas

provocadas por los astrocitos (Y Irizarry-Valle & Parker, 2014; Yilda Irizarry-Valle & Parker, 2015).

La Universidad de Tehran y la Universidad de Kermanshah, Irán, colaboran en un proyecto para el desarrollo de chip neuromórficos digitales para estudiar las interacciones entre neuronas y astrocitos (Hayati et al., 2016; Ranjbar & Amiri, 2015). Para ello emplean los modelos de la neurona de Izhikevich y así simulan los patrones que generan los potenciales de acción. Los astrocitos y las dinámicas de calcio se simulan con un modelo propuesto por Postnov y colegas (Postnov et al., 2009). Estos modelos fueron diseñados primero en Matlab y más tarde se simularon en circuitos FGPA. Para simular las señales entre las neuronas y los astrocitos se empleó la técnica *Single Constant Multiply*, así como aproximaciones lineales para aumentar la eficiencia.

En la Universidad de Ulster, Irlanda del Norte, trabajan en el proyecto EMBRACE (*Novel Field Programmable Neural Network Architecture*) que tiene como objetivo desarrollar hardware neuromórfico inspirado en los mecanismos del cerebro de detección y reparación automática (Harkin et al., 2009). Para ello han desarrollado tanto modelos realistas como chips neuromórficos con FGPA empleando una arquitectura jerárquica de la red de neuronas y astrocitos denominada SANN (*Spiking Astrocyte-Neuron Networks*). Esta arquitectura permite una mayor eficiencia en la comunicación ya que cuenta con dos niveles de intercambio de información, uno local y otro global, lo que le permite establecer un balance óptimo en el flujo de información entre la red local y global de astrocitos. La SANN se ha aplicado para el control de un vehículo robotizado mediante la frecuencia de los potenciales de acción de las neuronas. La red de neuronas y astrocitos tiene la capacidad de detectar fallos internos y repararlos. Cuando menos del 20% de las sinapsis asociadas con una neurona son defectuosas, el rendimiento del sistema no se ve afectado. Si el 80% de las sinapsis son defectuosas, el sistema muestra una reducción marginal del rendimiento, y esta reducción es mucho menor, comparada con la de los sistemas convencionales de tolerancia a errores. Esta es la primera aplicación real en la que se demuestra la capacidad de auto reparación de una SANN.

4.1.3. Modelos gliales en Inteligencia Artificial

En el campo de la IA los primeros modelos compuestos por neuronas y astrocitos fueron los desarrollados por Ana Porto en 2004 y se denominaron Redes NeuroGliales Artificiales (Porto-Pazos *et al*, 2004). Este trabajo se basó en el concepto de la sinapsis tripartita para incluir a los astrocitos como elemento de procesado. Para implementar las RNGA fue

necesario desarrollar una metodología adecuada (Ana Belen Porto-Pazos et al., 2005). Estos modelos fueron evaluados con problemas simples de clasificación, obteniendo resultados prometedores mejorando la actuación de las RNA (Porto-Pazos et al., 2007).

La metodología desarrollada para la implementación RNGA permitió crear redes multicapa compuestas por dos elementos de procesado: neuronas y astrocitos. La función de las células gliales es la de modular los pesos de las conexiones entre neuronas, simulando la actuación de los astrocitos en el córtex cerebral (Porto-Pazos et al., 2011a).

Las RNGA emplean un método de entrenamiento híbrido, que combina una primera fase de aprendizaje no supervisado, en la que actúan los astrocitos, con una segunda fase de aprendizaje supervisado, en la que se emplean algoritmos genéticos. Primero actúa el sistema glial, para ello se realizan n iteraciones con el mismo patrón de entrenamiento, y la neurona se tiene que activar m veces para que también lo haga el astrocito. Así, se puede simular la lentitud astrocítica con respecto a la rapidez de respuesta de las neuronas y, por tanto, el paso del tiempo necesario para que los astrocitos actúen. Cuando se activan los astrocitos, potencian las conexiones excitadoras, con pesos positivos ($w > 0$). Las conexiones inhibitorias, $w < 0$, se deprimen. Si el astrocito no se activa, las conexiones excitadoras se atenúan y las conexiones con pesos negativos se potencian.

Posteriormente, en la segunda fase de entrenamiento actúan los algoritmos genéticos. A partir de una población inicial de pesos de las RNGA, en cada generación se combinan las redes con menor error para dar lugar a nuevos individuos. Además, se pueden producir mutaciones que introducen variaciones, que en caso de ser positivas podrán transmitirse a la descendencia.

Durante la fase de entrenamiento se realiza de forma conjunta la validación de la red, con un conjunto de patrones de validación, diferente al de entrenamiento, para asegurarse que la red mantiene la capacidad de generalización y no se produce un sobreentrenamiento. La red solo modifica los pesos en función del error con los patrones de entrenamiento, mientras que el error de validación sirve para seleccionar la configuración de pesos óptima que se empleará en la fase de test. Los primeros resultados obtenidos con RNGA fueron satisfactorios con respecto a los obtenidos con RNA (Alvarellos-González et al., 2012a; Porto-Pazos et al., 2007; Porto-Pazos, 2008). Debido a la variedad de interacciones entre los astrocitos y las neuronas, se desarrollaron varios algoritmos gliales de actuación para evaluar las diferentes posibilidades. Se demostró que los astrocitos artificiales aumentan la eficacia de la red en problemas de clasificación/predicción simples y complejos sin aumentar, o incluso reduciendo, la velocidad de aprendizaje. Asimismo, se

verificó que la mejora en la eficacia no se debe al aumento de los elementos de procesado, si no exclusivamente a la presencia de astrocitos (Porto-Pazos et al., 2011b). También se comprobó que la influencia de los astrocitos depende del problema y de la arquitectura de la red, cuanto mayor complejidad, más eficaces son las RNGA respecto a las RNA. Finalmente, hay que resaltar que los resultados obtenidos hasta el momento muestran que la eficacia obtenida depende sobre todo de las propiedades del sistema glial, tanto el algoritmo neuroglial empleado, como la combinación de activación-iteración o la intensidad de la modificación glial (Alvarellos-González et al., 2012b).

En los últimos años, otros grupos han desarrollado RNGA pero que, en lugar de centrarse en la comunicación astrocitos-neurona, han simulado las ondas de calcio que generan los astrocitos para comunicarse (Ikuta et al., 2010, 2011, 2012, 2014, 2016).

Finalmente, en un trabajo presentado por Marcin Korytkowski en 2016, se desarrolló un parámetro que representa la glía, el cual modifica los pesos de las conexiones de una red de convolución preentrenada (Korytkowski, 2016).

4.2. Motivación

Esta tesis doctoral se centra en el concepto de la sinapsis tripartita y la capacidad de los astrocitos para procesar la información. Para estudiar los posibles modos de actuación de los astrocitos se han planteado dos objetivos principales de complejidad creciente. En primer lugar, se ha implementado un nuevo tipo de actuación de las células gliales en las RNGA inspirada en los descubrimientos en Neurociencia, ya que se ha demostrado que existe una gran variedad de posibles mecanismos de interacción entre los astrocitos y las neuronas. Hasta el momento, la glía actuaba modificando los pesos de las conexiones y el nuevo algoritmo glial propuesto actúa sobre las salidas de las neuronas. Como punto de partida de esta tesis doctoral se ha tomado la metodología desarrollada por Ana Porto y el grupo RNASA-IMEDIR (Porto-Pazos, 2004) para la implementación de las RNGA.

En segundo lugar, se ha planteado el objetivo de desarrollar una metodología para la creación de la primera red de *Deep Learning* formada por neuronas artificiales y astrocitos. Para ello se ha diseñado e implementado un nuevo tipo de algoritmo astrocítico compatible con el entrenamiento con el algoritmo de retropropagación y las técnicas de *Deep Learning*. Esto permitirá explotar el potencial de los astrocitos en redes de gran tamaño gracias a los avances que se han producido en el campo del *Deep Learning*. Al entrenar redes de mayor tamaño se podrán probar comportamientos más complejos de los astrocitos, por ejemplo, haciendo que un único astrocito controle la actividad de un gran número de neuronas de

manera simultánea, o regulando de manera diferente las neuronas en función de la capa en la que se encuentren. También será posible implementar una red de astrocitos que se comunique como ocurre en el cerebro con las ondas de calcio (A. H. Cornell-Bell et al., 1990). Además, existe la posibilidad de integrar el nuevo algoritmo glial con los modelos más avanzados de redes como las LSTM (Hochreiter & Schmidhuber, 1997) o las *Generative Adversarial Neural Networks* (I. J. Goodfellow et al., 2014).

En el siguiente capítulo se presenta el trabajo realizado con las RNGA, exponiendo la metodología empleada, los resultados obtenidos y las conclusiones de los mismos. En el capítulo 5 se expone el nuevo algoritmo astrocítico implementado, la metodología diseñada, así como los resultados y conclusiones.

5. Redes NeuroGliales Artificiales

5.1. Introducción

En este capítulo se presenta el trabajo realizado con las RNGA y el nuevo algoritmo implementado. Para ello se ha partido de la metodología desarrollada por Ana Porto y su equipo durante los últimos años. Cuando se creó la primera RNGA en el año 2004 no existían trabajos similares por lo que se tuvo que implementar una nueva metodología que se caracteriza por un aprendizaje híbrido con dos fases de entrenamiento. La primera fase es no supervisada y en ella actúan las células gliales modificando los pesos en función de los algoritmos gliales implementados. La segunda fase del entrenamiento es supervisada, los AA.GG. buscan los pesos óptimos utilizando en los obtenidos en la primera fase, para ello tratan de minimizar el error cuadrático medio. La metodología diseñada por Porto *et al* tiene como objetivo poder comparar la eficacia de las RNGA y comparar los resultados con las RNA entrenadas en las mismas condiciones, a excepción del algoritmo glial.

La arquitectura de las redes empleadas es multicapa totalmente conectadas hacia delante sin conexiones recurrentes y diseñadas para la resolución de problemas de clasificación. Las RNGA cada neurona tiene un astrocito que monitoriza su actividad, en la figura 18 se puede observar un esquema de la arquitectura.

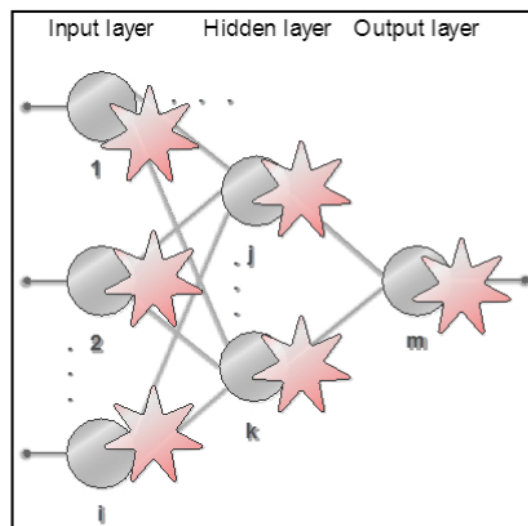


Figura 18. Esquema de la arquitectura de una RNGA.

A continuación, se explica en detalle el funcionamiento de las RNGA y la metodología desarrollada por Porto y el grupo RNASA-IMEDIR. El proceso es semejante al de otros

SC, en primer lugar, se diseña la arquitectura de la red y sus funciones de activación, las cuales dependen del tipo de problema que se trate, en este caso son problemas de clasificación. A continuación, se realiza la fase de entrenamiento y validación de manera alterna, y finalmente se lleva a cabo el test. En la figura 19 se puede observar un diagrama de la metodología.

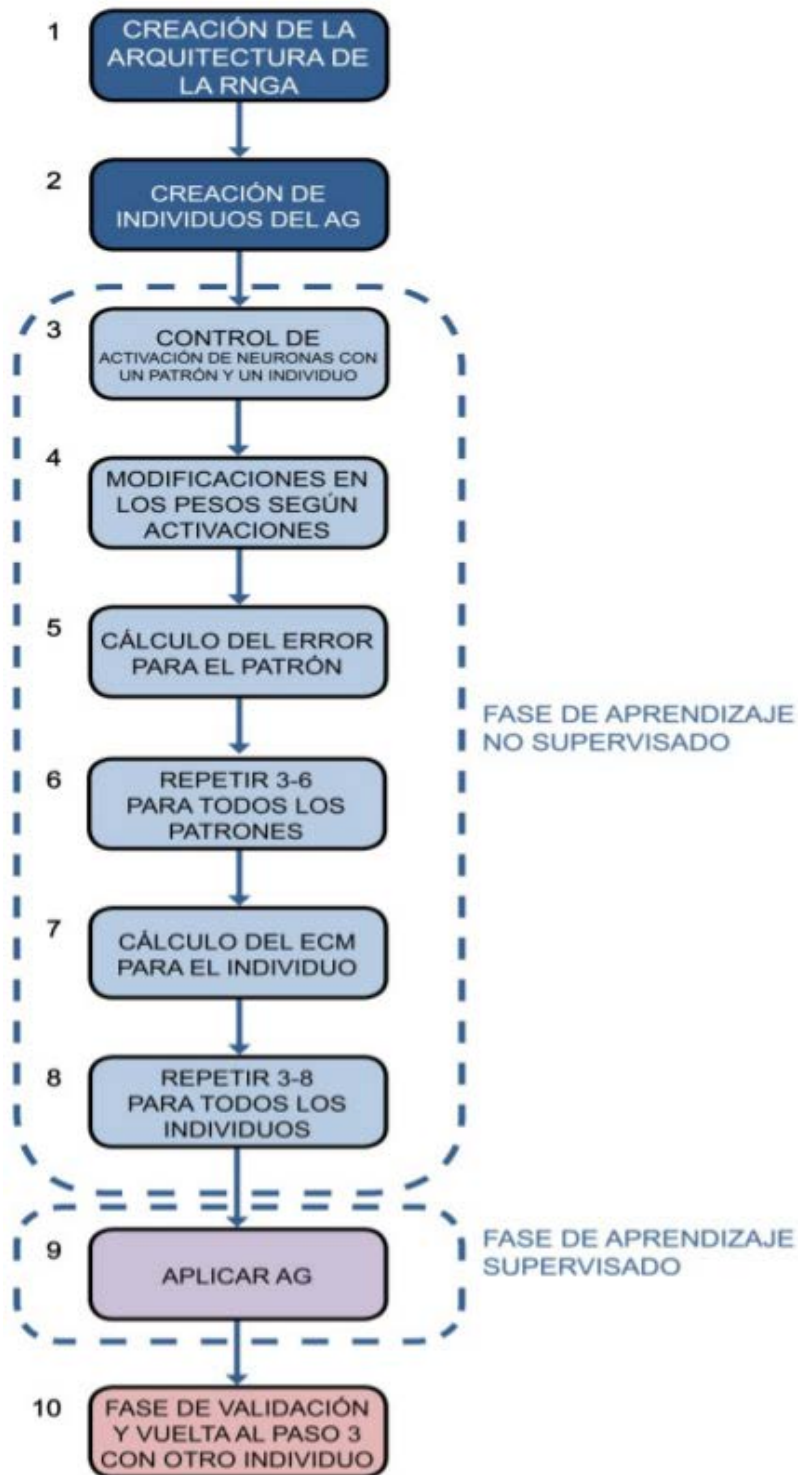


Figura 19. Esquema de la metodología de las RGA.

5.1.1. Fase de Diseño

Las RNGA se han diseñado para la resolución de problemas de clasificación. Como se ha mencionado, se emplea una arquitectura multicapa totalmente conectadas hacia delante sin conexiones recurrentes.

5.1.2. Fase de Entrenamiento

El entrenamiento de las RNGA es híbrido, consta de una fase de aprendizaje no supervisado, en la que actúa el algoritmo glial para modificar los valores de los pesos de las conexiones en función de la actividad de las neuronas, y una fase de aprendizaje supervisado en la que se emplean los AA.GG para tratar de optimizar los pesos de la red obtenidos en la primera fase. En la figura 19 esta fase de entrenamiento se corresponde con los pasos 3-9.

5.1.2.1. Fase de Aprendizaje no Supervisado: Algoritmo Glial

La fase de aprendizaje no supervisado se basa en el algoritmo glial que trata de simular el comportamiento de los astrocitos observado en el cerebro y su papel en la sinapsis tripartita. En la metodología desarrollada se combina esta fase no supervisada con la fase supervisada mediante el algoritmo genético (AG). Para la actuación de los AG es necesario tener un conjunto de individuos, denominado población, por lo que el primer paso es crear esta población. Esta población está formada por individuos, cada uno de los cuales representa todos los pesos de las conexiones de la RNGA, por lo que cada individuo tiene tantos valores como pesos tenga la RNGA. Los valores de cada individuo se modifican cada vez que se pasa un patrón de entrenamiento. Los patrones se pasan un número determinado de veces o iteraciones. De esta manera se trata de simular la diferencia en la escala temporal en la que procesan la información los astrocitos respecto a las neuronas. En los trabajos desarrollados por Porto *et al*, se establecieron las reglas por las cuales actúan los astrocitos artificiales, determinando unos valores y unos mecanismos de actuación óptimos (Porto-Pazos et al., 2007, 2009; Romero-Porta et al., 2004). A continuación, se detalla el funcionamiento del algoritmo glial y los valores óptimos para los diferentes parámetros gliales.

El algoritmo glial desarrollado actúa de la siguiente manera: cada patrón se pasa a la red n veces, si la neurona se activa m veces en las n iteraciones, la glía considerará que la neurona

está activa y lo tendrá en cuenta para modificar sus pesos. Los valores óptimos para realizar las modificaciones de los pesos se obtuvieron tras las experimentaciones realizadas por Ana Porto y el grupo RNASA-IMEDIR (Porto-Pazos et al., 2005; Porto-Pazos et al., 2007; Romero-Porta et al., 2004), y son los siguientes:

- Incrementar un 25% el valor del peso de las conexiones que salen de las neuronas que se activan n veces en las m iteraciones.
- Disminuir un 50% el valor del peso de las conexiones que salen de las neuronas que no se activan n veces en las m iteraciones.

Por ejemplo, considerando una combinación de activación-iteración 2-4, cada patrón se pasa 4 veces a la red y el algoritmo glial incrementa un 25% el peso de las conexiones que salen de las neuronas que se activen 2 veces. En este caso se considera que la actividad neuronal es suficiente para que el astrocito se active y modifiquen los pesos de las conexiones salientes. Por el contrario, los pesos de las conexiones que salen de las neuronas que se activen menos de 2 veces, se reducen un 50%. Por tanto, la actividad neuronal no es suficiente como para activar al astrocito, por lo se reducen los valores de las conexiones.

Este modo de actuación representa una de las maneras de funcionar de los astrocitos observadas en el cerebro, necesitando que las neuronas tengan cierta actividad para que los astrocitos actúen en la sinapsis tripartita. Las iteraciones necesarias tratan de representar las neuronas con alta actividad, y de esta manera se consigue que los astrocitos actúen en una escala temporal diferente que las neuronas.

Los valores asignados para los incrementos y decrementos de los pesos se obtuvieron tras la experimentación realizada por Ana Porto (Porto Pazos, 2008), estos valores son óptimos y consiguen buenos resultados. Aunque es posible que existan combinaciones mejores, ya que no es abordable probar todas las combinaciones existentes. Los experimentos en los laboratorios de Neurociencia justifican esta elección, ya que al ser el incremento por refuerzo menor que el decremento, solo se reforzarán las conexiones de neuronas que presenten una actividad constante.

El número de activaciones e iteraciones también se determinó experimentalmente (Porto, 2004), se comprobó que con más de ocho iteraciones se incrementa demasiado el tiempo de simulación y no se obtiene una mejora significativa de los resultados. Por otra parte, con menos de dos iteraciones no da tiempo a que la glía actúe. Los valores óptimos de activación-iteración obtenidos por Porto *et al*, han sido los presentado en la tabla 3.

Tabla 3. Combinaciones de activación-iteración de las RNGA.

Activación	Iteración
2	4
2	6
3	6
3	8

En cuanto al proceso de entrenamiento, para cada patrón se realizan las iteraciones que determine el algoritmo glial, y después de realizar las modificaciones por parte de los astrocitos, se calcula el error de la red para ese patrón, que se obtiene de la diferencia entre la salida obtenida y la salida deseada. El error obtenido para cada patrón se almacena y tras pasar todos los patrones, se calcula el error cuadrático medio (ECM) para ese individuo. Este proceso se repite para todos los individuos de la población. Se considera que esta fase es no supervisada ya que la actuación del algoritmo glial y las modificaciones que realiza no tienen en cuenta la salida que produce la red y su error, tan solo actúa en función de la actividad de cada neurona de manera individual, simulando los mecanismos de actuación de los astrocitos observados en el cerebro.

En resumen, la fase de aprendizaje no supervisado se compone de los siguientes pasos:

1. Generación aleatoria de los individuos iniciales, cada individuo contiene los pesos de las conexiones de la red.
2. Se introduce un patrón y para cada neurona se calcula el valor neto y el valor de activación.
3. La glía contabiliza las neuronas que se activan, y se repite el ciclo el número de iteraciones que marque el algoritmo glial.
4. Las neuronas que se activen el número necesario de veces, harán que actúe la glía y se aumente el valor de sus pesos. Las neuronas que no se activen verán reducidos sus pesos. Hay que señalar que estas modificaciones son temporales, ya que al introducir el siguiente patrón el individuo es el mismo que al comienzo, las modificaciones producidas por la glía no perduran y solo afectan al error obtenido.

5. Tras realizar las iteraciones correspondientes, se calcula el error obtenido por el individuo para ese patrón, es decir, la diferencia entre la salida deseada y la obtenida.
6. Este proceso se repite con todos los patrones y con todos los errores obtenidos, se calcula el ECM del individuo 1.
7. Para cada individuo de la población se realiza el proceso anterior, para obtener un vector con todos los individuos y su ECM asociado. Este vector será el que utilice el AG en la fase de aprendizaje supervisado para realizar las mutaciones y cruces.

Este sería el resumen de la fase de aprendizaje no supervisado y los mecanismos de actuación del algoritmo glial el cual no tiene en cuenta la salida deseada, sino que solo considera las activaciones de las neuronas para realizar las modificaciones de los pesos.

5.1.2.2. Fase de Aprendizaje Supervisado: Algoritmo Genético

En la fase de aprendizaje supervisado actúa el algoritmo genético empleando los individuos generados en la fase anterior. Para ello, se ordenan los individuos en función de su ECM y se emplean los correspondientes operadores genéticos que realizan los cruces y mutaciones. A continuación, se seleccionan los nuevos individuos que se emplearán en una nueva fase de aprendizaje no supervisado. Esta fase es supervisada ya que el AG tiene en cuenta el error cometido por la red para seleccionar los individuos, por lo que realiza los cruces y mutaciones de acuerdo a ese error. Se repetirá este proceso hasta que se alcance el mínimo ECM o se ejecuten un número determinado de generaciones.

5.1.3. Fase de Validación

La fase de validación se realiza de manera conjunta al entrenamiento y sirve para evaluar la red con unos patrones diferentes a los empleados para el entrenamiento. De esta manera se busca evitar el sobreentrenamiento, tratando de encontrar el balance justo entre el aprendizaje y la capacidad de generalización de la red.

En la fase de validación se emplea el proceso de parada temprana, que sirve para detener el entrenamiento en el momento óptimo, evitando que la red memorice los patrones de entrenamiento, perdiendo así su capacidad de generalización y la posibilidad de clasificar correctamente los patrones nuevos no empleados en el entrenamiento. Para poder realizar la parada temprana es necesario evaluar el error de la red con el conjunto de validación después de realizar cada ciclo de fase de entrenamiento no supervisado y supervisado. Las

configuraciones de los pesos deberán guardarse, para poder disponer de las configuraciones previas de la red. Así, si se alcanza un punto en el que la red comienza a aumentar de manera continuada su error en el conjunto de validación, se podrá parar el entrenamiento y recuperar la configuración que obtuvo el menor error en el conjunto de validación.

5.1.4. Fase de Test

Tras la fase de entrenamiento y validación, se obtienen las RNGA con una configuración de pesos optimizada para obtener el mínimo error posible. En la fase de test se utilizarán estos individuos para evaluar la capacidad de generalización de la red, es decir como es la actuación con patrones completamente nuevos, diferentes de los empleados en la fase de entrenamiento y validación.

Existe una diferencia importante entre las RNGA y otros modelos de redes, y es que el algoritmo glial continúa actuando de la misma manera en la fase de test. Por lo que los astrocitos serán parte integral del modelo en todas las fases y participan de manera activa y continua en el procesado de la información. De tal manera, que los patrones de test también deberán ser presentados durante un número determinado de iteraciones para que el algoritmo glial pueda actuar sobre los pesos de las conexiones en función de la actividad de las neuronas. Este modo de actuación tiene un mayor realismo biológico y guarda mayor relación con los procesos de aprendizaje que ocurre en el cerebro, al contrario que los modelos típicos de RNA en los que el aprendizaje sólo ocurre en la fase de entrenamiento. El algoritmo glial permite realizar un aprendizaje activo incluso en la fase de test, como se ha comprobado en experimentos neurocientíficos, el cerebro mantiene su plasticidad y capacidad de aprendizaje de forma que se puede adaptar a nuevas situaciones y realizando un aprendizaje constante.

5.2. Antecedentes Redes NeuroGliales Artificiales

La metodología presentada ha sido desarrollada por Ana Porto y el grupo RNASA-IMEDIR desde el año 2004, y las RNGA se han evaluado y comparado a las RNA en varios trabajos (Porto-Pazos et al., 2007). En primer lugar, se realizó una comparación en la resolución de problemas de clasificación sencillos (MUX y clasificación de *Flor de Iris*). Las RNG fueron entrenadas con el método de aprendizaje híbrido presentado en el apartado anterior, y las RNA se entrenaron mediante AA.GG. Las arquitecturas de la red y todos los parámetros del AG se mantuvieron constante para los dos métodos. Además, se emplearon

las mismas poblaciones iniciales de individuos (tres poblaciones) y la misma semilla de generación aleatoria (con la que se realiza la selección de individuos para el cruce y la mutación). El número de generaciones máxima también era el mismo para las RNA y las RNGA. De esta manera se garantiza la validez de la comparación realizada entre ambos tipos de redes.

Para el problema del MUX se emplearon dos arquitecturas, una de tres capas y otra de cinco, estableciendo el número máximo de generaciones en 4000. En todos los casos las RNGA mejoraron los resultados de las RNA, pero en el caso de la arquitectura de cinco capas las diferencias eran incluso más significativas, lo que podría sugerir que el algoritmo glial tiene es más efectivo en redes de mayor complejidad.

El problema de *Flor de Iris* es más complejo que el de MUX, y se trata de un problema de clasificación múltiple. En este caso los resultados obtenidos con las RNGA también fueron superiores a las RNA y además el ECM mínimo se alcanzó en un número menor de generaciones.

Estos resultados demostraron el potencial de las RNGA para resolver problemas de clasificación, mejorando los resultados obtenidos con RNA y reduciendo el tiempo necesario para alcanzar el mínimo ECM. Tras esta prueba de concepto, se realizó una comparación más exhaustiva y detallada entre las RNGA y las RNA, para ello se llevaron a cabo pruebas con varios de problemas de clasificación y varias arquitecturas. En concreto, se evaluaron con 4 problemas del *University of California Irvine Machine Learning Repository* (UCI-MLR) (Lichman, 2013): enfermedades coronarias, cáncer de mama, señales ionosfera y el ya citado problema de *Flor de Iris*. Los resultados de este trabajo muestran que las RNGA mejoran los resultados en los problemas del cáncer de mama y el de ionosfera. En el problema de enfermedad coronaria obtiene peores resultados y en el caso de la *Flor de Iris* son similares. Además, se comprobó que la mejora no se debe a la presencia de un mayor número de elementos de procesado en las RNGA. Para ello, se realizaron pruebas con RNA con el mismo número de neuronas que elementos de procesado tenían las RNGA, suma de las neuronas y astrocitos artificiales. Los resultados obtenidos muestran que no existe correlación entre el aumento en el número de neuronas y la mejora de los resultados. En este estudio también se pudo demostrar que la influencia de las células gliales se correlaciona con la complejidad de la red. Se realizaron pruebas con arquitecturas de una, dos y tres capas ocultas. Los resultados muestran que cuanto mayor es el número de capas, mejores son los resultados de las RNGA comparado con las RNA independientemente del problema.

Por último, se han desarrollado varias versiones del algoritmo glial basándose en los diferentes comportamientos observados en los experimentos de Neurociencia. El algoritmo presentando hasta el momento se denomina de potenciación ya que incrementa las conexiones de las neuronas activas y decreta las conexiones de las neuronas no activas.

Se han realizado varias versiones de este algoritmo que tienen en cuenta otros factores:

Los algoritmos se evaluaron con los problemas de MUX y *Flor de Iris*, y el algoritmo glial de activaciones no consecutivas y pesos ilimitados obtuvo los mejores resultados en el problema de *Flor de Iris*.

Además, se desarrolló otro algoritmo denominado de depresión que actúa de manera contraria al de potenciación, ya que decreta las conexiones de las neuronas activas e incrementa las conexiones de las neuronas no activas. Estos dos algoritmos se compararon en los cuatro problemas ya empleados del UCI-MLR: enfermedades coronarias, cáncer de mama, señales ionosfera y *Flor de Iris*. Los resultados muestran que el tipo de algoritmo glial, potenciación o depresión, que produce mejores resultados depende del problema analizado.

Basándose en estos trabajos previos, el objetivo es desarrollar un nuevo algoritmo glial inspirado en los mecanismos de actuación de los astrocitos observados en el cerebro. Hasta el momento el algoritmo glial actuaba exclusivamente en los pesos de las conexiones sináptica. Biológicamente esta actuación representa una modificación del espacio sináptico producida por una variación en el volumen del astrocito perisináptico que envuelve las sinapsis.

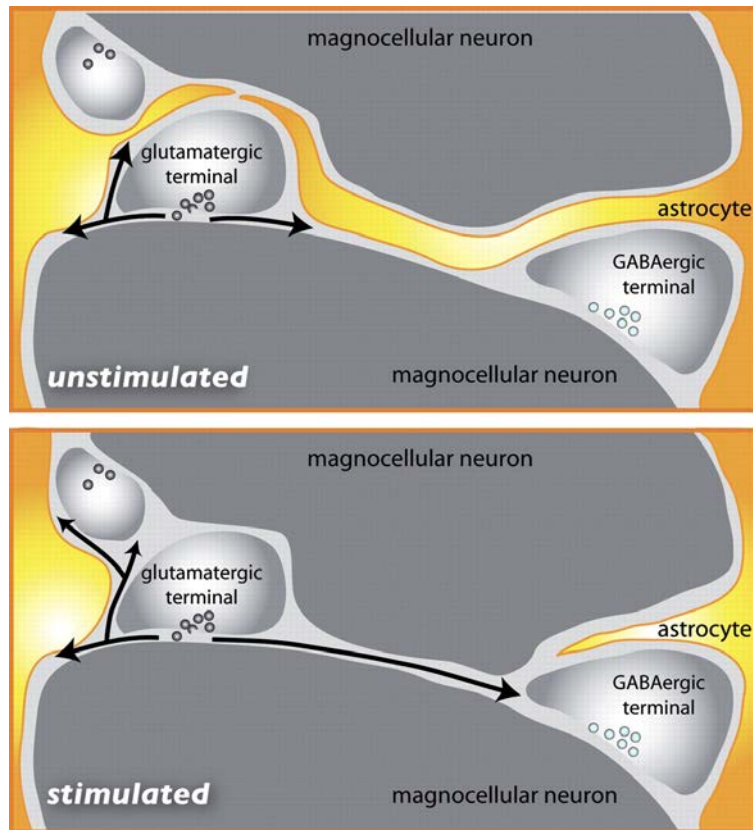


Figura 20. Retracción del proceso astrocitario debido a la estimulación, el glutamato liberado en la terminal glutamatergica puede afectar a la sinapsis gabaérgica. Este comportamiento ha servido como inspiración para implementar el algoritmo glial en el cual los astrocitos actúan sobre los pesos de las conexiones (Theodosis et al., 2008).

Sin embargo, en la sinapsis tripartita los astrocitos cuentan con varios mecanismos de actuación. Como se ha mencionado en el apartado de Fundamentos, los astrocitos pueden comunicarse con la neurona presináptica por medio de gliotransmisores. Mediante esta comunicación los astrocitos son capaces de influir en la salida de la neurona, modulando los neurotransmisores que emiten. En este comportamiento se inspira el nuevo algoritmo desarrollado para el funcionamiento de las RNGA, en el cual las células gliales modifican la salida de la neurona en lugar de los pesos de las conexiones. A continuación, se presenta la metodología empleada para llevar a cabo la experimentación y así comparar el nuevo algoritmo glial frente al ya existente. Además, se comparará con las RNA para comprobar si esta nueva versión del algoritmo glial también mejora los resultados.

5.3. Metodología

La metodología empleada para implementar el nuevo algoritmo glial es propia del grupo de investigación RNASA-IMEDIR y ha sido desarrollada y mejorada a lo largo de los años desde que Ana Porto realizó los primeros trabajos con RNGA en el año 2004.

Para el diseño e implementación de estos modelos computacionales es importante seguir un proceso metodológico explícito para mantener el rigor científico y obtener resultados válidos. La metodología empleada marca los siguientes pasos generales:

- Identificación del problema a resolver.
- Preparación de los datos.
- Diseño y construcción de la red.
- Fase de entrenamiento, validación y test.
- Organización y análisis de resultados.
- Validación del sistema.

5.3.1. Problemas a Resolver

Para la comparación entre las RNGA y las RNA se emplearon cinco problemas de clasificación del UCI-MLR: cáncer de mama, señales de ionosfera, señales de sonar, diagnóstico de diabetes y setas venenosas. Estos problemas se han seleccionado porque han sido utilizados previamente por el grupo de investigación RNASA-IMEDIR para la evaluación de las RNA y uno de los objetivos de esta tesis es comparar los resultados del nuevo algoritmo glial con los desarrollados hasta el momento. A continuación, se describen los diferentes problemas, las variables que contienen y la distribución de los datos.

5.3.1.1. Diagnóstico del Cáncer de Mama

El problema del cáncer de mama consiste en clasificar los tumores en malignos o benignos a partir de las variables histológicas que normalmente miden los especialistas para emitir su diagnóstico (Mangasarian et al., 1990). Los datos fueron tomados por el Dr. William H. Wolberg en el “University of Wisconsin Hospitals, Madison” y pertenecen a un total de 699 pacientes, de los cuales 458 (65,5%) se corresponden a la clase negativa, es decir el tumor es benigno, y 241 (34,5%) a la clase positiva, el tumor es maligno. Para cada caso se han medido 9 variables histológicas, las cuales toman valores en una escala del 1 al 10. Las variables son las siguientes:

- Espesor (*Clump thickness*)
- Uniformidad del tamaño de la célula (*Uniformity of cell size*)
- Uniformidad de la forma de la célula (*Uniformity of cell shape*)
- Adhesión marginal (*Marginal adhesion*)
- Tamaño individual de la célula (*Single epithelial cell size*)

- Núcleo desnudo (*Bare nuclei*)
- Cromatina laxa (*Bland chromatin*)
- Nucléolos normales (*Normal nucleoli*)
- Mitosis (*Mitoses*)

5.3.1.2. Señales de Ionosfera

Los datos de este problema fueron recogidos por un array de 16 antenas de alta frecuencia con un poder de transmisión total del orden de 6,4 kilowatios (Sigillito et al., 1989). El sistema está situado en Goose Bay, Labrador, Canadá y los datos procesados por Vince Sigillito *et al* del *Space Physics Group, Applied Physics Laboratory* de la *Johns Hopkins University*. Las antenas tienen como objetivo electrones libres en la ionosfera. En este problema se trata de clasificar la señal recibida en “buenas” o “malas”. Se consideran señales “buenas” aquellas que muestran evidencia de algún tipo de estructura en la ionosfera. Por su parte, en aquellas etiquetadas como “malas”, las señales pasan a través de la ionosfera por lo que no muestran evidencia de algún tipo de estructura.

La señal recibida se procesa con una función de autocorrelación en base al tiempo del pulso electromagnético y el número de pulsos. La señal está representada por un total de 34 variables y hay un total de 351 patrones.

5.3.1.3. Señales de Sonar

En este problema el objetivo es clasificar la señal de sonar en función del objeto que la emite, en este caso una mina o una roca (Gorman & Sejnowski, 1988). Para ello, se integra la energía en una determinada banda de frecuencia respecto a un periodo de tiempo. Los datos de las minas se tomaron mediante varias medidas a diferentes ángulos y bajo varias condiciones de un objeto metálico. En el caso de las rocas, se realizaron las medidas bajo las mismas condiciones. Existen un total de 60 variables normalizadas con valores entre 0 y 1. La base de datos consta de un total de 208 ejemplos, 111 de minas y 97 de rocas.

5.3.1.4. Diagnóstico de Diabetes

Esta base de datos fue recogida por el *National Institute of Diabetes and Digestive and Kidney Diseases* y el objetivo es diagnosticar la diabetes en función de 8 variables (Smith et al., 1988). Todos los pacientes era mujeres de al menos 21 años de edad, del grupo indígena Pima, nativos americanos del área del actual estado de Arizona, Estados Unidos. En total

hay 538 casos, de los cuales 183 son ejemplos de mujeres con diabetes y 355 de mujeres que no padecen diabetes. Las variables están normalizadas entre 0 y 1 y son las siguientes:

1. Número de embarazos.
2. Concentración de glucosa en plasma a las dos horas en un test oral de tolerancia a la glucosa.
3. Presión sanguínea diastólica (mm Hg).
4. Espesor del pliegue cutáneo del tríceps (mm).
5. Insulina en suero a las 2 horas (mu U/ml).
6. Índice de masa corporal (Peso en Kg/altura en metros al cuadrado).
7. Función pedigree de la diabetes.
8. Edad (en años).

5.3.1.5. Setas Venenosas

En este problema el objetivo es clasificar las setas en venenosas o no en función de 22 variables (Schlimmer, 1981). Existen 5644 casos, de los cuales 2156 corresponden a setas venenosas y 3488 a setas no venenosas. Todas las variables utilizadas para crear esta base de datos son nominales, pero han sido normalizadas con valores entre 0 y 1. Estas variables representa características físicas como la forma, color, superficie de las diferentes partes tales como el sombrero, el himenio, el tallo o el velo. Además, existen variables con información sobre el hábitat o la distribución de la población.

5.3.2. Preparación de los Datos

Los conjuntos de datos de los cinco problemas se han organizado siguiendo las reglas estándar PROBEN1 (Prechelt, 1994), de tal forma que el 50% de los ejemplos se utilizan en el entrenamiento, un 25% para la validación y un 25% para el test. Además, para comprobar y validar adecuadamente los resultados, se empleó la técnica 5x2cv cross-validation. En total se crearon 10 conjuntos distintos de patrones usando 5 iteraciones del método de validación cruzada 5x2cv (Dietterich, 1998). Además, se emplearon 10 poblaciones iniciales distintas de 150 individuos cada una, cada individuo está compuesto por los valores de pesos de todas las conexiones de la red. Esto da lugar a 100 combinaciones de conjuntos de patrones y poblaciones iniciales. De esta manera, los

resultados obtenidos no dependen del orden de presentación de los casos o de la distribución de los ejemplos en los diferentes conjuntos ni de los pesos iniciales de la red.

5.3.3. Diseño y Construcción de la Red

Para poder realizar una comparación adecuada entre las RNA y las RNGA, es necesario mantener constantes todos los parámetros posibles de las redes. Por ejemplo, todas las neuronas emplean una función de activación tangente. En cada problema el número de neuronas en la capa de entrada se corresponde con el número de variables en ese problema. En el caso de la capa de salida, ya que son problemas de clasificación binaria, en todos los casos solo hay una neurona con un valor umbral de 0,5. En cuanto a las capas ocultas de la red, para cada problema se ha seleccionado las arquitecturas de red de una capa oculta y dos capas ocultas que mejores resultados ha obtenido en los trabajos previos del grupo RNASA-IMEDIR. En el caso de las arquitecturas de tres capas ocultas, se ha empleado la misma para todos los problemas. En la tabla 4 se puede observar las arquitecturas empleadas para cada problema. En el problema del diagnóstico de cáncer, las redes de una capa oculta tienen 7 neuronas; las de dos capas ocultas tienen 7 en una capa y 5 en la otra; las de 3 capas ocultas tienen 12 en la primera, 8 en la capa intermedia y 4 en la última. Para el problema de la ionosfera, las redes de una capa oculta tienen 9 neuronas; las de dos capas ocultas tienen 9 en una capa y 4 en la otra; las de 3 capas ocultas tienen 12 en la primera, 8 en la capa intermedia y 4 en la última. En el problema del sonar, las redes de una capa oculta tienen 6 neuronas; las de dos capas ocultas tienen 6 en una capa y 4 en la otra; las de 3 capas ocultas tienen 12 en la primera, 8 en la capa intermedia y 4 en la última. Para el problema del diagnóstico de diabetes las redes de una capa oculta tienen 4 neuronas; las de dos capas ocultas tienen 6 en una capa y 4 en la otra; las de 3 capas ocultas tienen 12 en la primera, 8 en la capa intermedia y 4 en la última. Por último, en el caso del problema de las setas venenosas, las redes de una capa oculta tienen 7 neuronas; las de dos capas ocultas tienen 12 en una capa y 7 en la otra; las de 3 capas ocultas tienen 12 en la primera, 8 en la capa intermedia y 4 en la última.

Tabla 4. Arquitecturas de las redes empleadas en cada problema.

	Arquitectura		
Problema	Una capa	Dos capas	Tres capas
Cáncer	7	7-5	12-8-4

Ionosfera	9	9-4	12-8-4
Sonar	6	6-4	12-8-4
Diabetes	4	6-4	12-8-4
Setas	7	12-7	12-8-4

Para el desarrollo de la experimentación con RNGA, en los trabajos previos del grupo RNASA-IMEDIR, Ana Porto y su equipo han desarrollado una metodología que permite paralelizar la ejecución de las pruebas en el Centro de Supercomputación de Galicia (CESGA) (“Centro de Supercomputación de Galicia (CESGA),” 2017). En concreto se ha empleado el sistema SVG que cuenta con un sistema de colas *Sun Grid Engine*. El SVG permite que cada usuario pueda tener en el sistema de colas hasta 800 procesos de maneras simultánea, los cuales entrarán en ejecución en alguno de los nodos de la máquina en base a un sistema de prioridades, que tiene en cuenta: el tiempo de espera en la cola, el tiempo de ejecución ya empleado por un mismo usuario, y los requerimientos de memoria y tiempo del mismo. Esto ha permitido poder realizar la experimentación en un tiempo razonable, ya que, debido al alto coste computacional y al gran número de pruebas, es posible paralelizar las mismas para optimizar los procesos.

La aplicación para el desarrollo de las simulaciones emplea el lenguaje de programación C ya que permite un mayor control sobre el uso de la memoria empleada. Esto permite una mayor eficiencia y velocidad en la ejecución de las simulaciones. La aplicación se compone de una serie de scripts y archivos de parámetros, los cuales se organizan en unos directorios con una estructura específica. Los directorios genéricos son los siguientes:

- **entrenar_linux**: directorio donde se encuentran los archivos de código fuente, compilados y ejecutables del código de la aplicación de simulación, que permiten realizar la fase de entrenamiento y validación.
- **entrenar_linux_test**: en este directorio está compilado el código fuente para llevar cabo la fase de test.
- **new_test**: contiene una serie de programa que automatizan el análisis de los resultados del test.

Para cada prueba se crea un directorio específico que contiene los siguientes subdirectorios:

- **fParametros**: en esta carpeta se sitúan los archivos con los parámetros de cada red. En el caso de las RNGA, existe un archivo para cada combinación de parámetros de activación-iteración.

- **poblaciones:** ubicación de los archivos de las 10 poblaciones diferentes con los 150 individuos cada una.
- **patrones:** se organiza en 10 subdirectorios, cada uno de los cuales tiene una distribución diferente de los conjuntos de datos para el entrenamiento, validación y test.
- **resultados:** se compone de 10 subdirectorios, con los resultados para uno de los conjuntos de patrones. En los ficheros de resultados (*log*) se guardan los valores de: número de generación, error de entrenamiento, error de validación y tiempo en alcanzarse.
- **ejecutables:** contiene los scripts necesarios para enviar los trabajos a la cola de procesos *Sun Grid Engine* del sistema SVG del CESGA (“Centro de Supercomputación de Galicia (CESGA),” 2017).

Una vez que se han organizado todos los archivos y directorios específicos, las pruebas son enviadas a la cola *Sun Grid Engine* del sistema SVG del CESGA para su ejecución. A continuación, se detalla el proceso de entrenamiento de las RNA y las RNGA, así como el funcionamiento concreto del algoritmo glial implementado.

5.3.4. Fase de Entrenamiento, Validación y Test

Como se ha mencionado, el entrenamiento de las RNA es supervisado y se realiza mediante AA.GG. Las RNGA tienen un entrenamiento híbrido, una fase no supervisada en la que actúa el algoritmo glial y una fase supervisada mediante AA.GG. En este apartado se presenta en primer lugar los parámetros de los AA.GG empleados en la experimentación, y a continuación se detalla el funcionamiento del algoritmo glial.

5.3.4.1. Algoritmos Genéticos

En todos los problemas se han mantenido constantes todos los parámetros de los AA.GG, tanto para las RNA como para las RNGA, excepto el número máximo de generaciones que ha sido superior para las RNA ya que se ha demostrado que tardan más tiempo en alcanzar la configuración óptima de pesos (Porto-Pazos et al., 2011b). En los problemas de cáncer, ionosfera, diabetes y setas, las generaciones de la RNA fueron 10.000 y las de la RNGA 3.000. En el problema del sonar se asignaron más generaciones, 15.000 para la RNA y 5.000 para la RNGA, ya que el número de variables es elevado y las redes necesitan más

tiempo para entrenarse. El resto de parámetros de los AA.GG empleado a lo largo de toda la experimentación están representados en la tabla 5.

Tabla 5. Parámetros de los algoritmos genéticos empleados.

Parámetro del AA.GG	Valores
Número de puntos de mutación	1
Número de puntos de cruce	0
Tasa de cruce	90%
Tasa de mutación	10%
Tipo de sustitución/inserción	Peor individuo
Tipo de selección	Torneo
Tamaño de ventana del torneo	2

5.3.4.2. Algoritmo Glial con modificación de la salida

Como se ha comentado en apartados anteriores, basándose en los trabajos previos desarrollados por Ana Porto y el grupo RNASA-IMEDIR, se ha implementado un nuevo algoritmo glial (figura 21) que presenta diferencias respecto a los trabajos previos (Alvarellós-González et al., 2012b; Porto-Pazos et al., 2011b; Porto-Pazos et al., 2005, 2007; Porto-Pazos, 2008; Romero-Porta et al., 2004).

```

(...)
Para cada individuo:
{
  Para cada patrón:
  {
    Para cada iteración:
    {
      Procesar patrón:
      {
        Evaluar patrón con la red
        Modificaciones astrocíticas:
        Incremento/Decremento de salida de la neurona
      }
    }
  }
}
(...)

```

Figura 21. Pseudocódigo del funcionamiento de las RNGA con modificación de la salida de la neurona.

En primer lugar, el algoritmo glial implementado actúa modificando el valor de salida de las neuronas, en lugar de los pesos de las conexiones como se hacía hasta el momento. Esto simula varios comportamientos observados en los astrocitos reales en los experimentos de laboratorio:

- Capacidad de los astrocitos de producir un aumento, o reducción, de la cantidad de neurotransmisor liberado en la terminal sináptica por parte de la neurona. Esta comunicación con la neurona se produce mediante la liberación de gliotransmisores como ATP o glutamato, que es captado por la neurona presináptica por medio de receptores A1, mGluR o NMDAR (Alfonso Araque et al., 1999; Fiacco & McCarthy, 2004; Kang et al., 1998; Newman & Zahs, 1998; Robitaille, 1998; Theodosis et al., 2008).
- Liberación de una cantidad extra de gliotransmisor, como glutamato, D-serina o GABA, que es captado por la neurona postsináptica mediante receptores GABA, mGluR o NMDAR (A Araque et al., 1998; Kang et al., 1998; Oberheim et al., 2012b).
- Captación de neurotransmisor del espacio sináptico por parte del astrocito, lo cual reduce la concentración en la hendidura sináptica (Tzingounis & Wadiche, 2007).

En segundo lugar, el nuevo algoritmo glial implementado incorpora un astrocito en la capa de salida, que puede modificar directamente la clasificación que realiza la red. Hasta el momento, en las RNGA no existía una célula glial en la capa de salida, ya que esta no tiene

conexiones hacia otras neuronas. Por tanto, con esta versión del algoritmo glía, es posible realizar un ajuste fino de la salida de la red, modificando directamente la salida y aumentando las posibilidades de éxito en la clasificación.

Por último, cuando los astrocitos se activan potencian tanto las sinapsis inhibitoras como las excitadoras. Como se ha comentado previamente, en los algoritmos gliales implementados hasta el momento, cuando los astrocitos se activan potencian las conexiones excitadoras, con pesos positivos, (concretamente se multiplican por 1,25) y deprimen las inhibitoras, con pesos negativos (multiplicándolas por 0,75). Si el astrocito no se activa, multiplica las conexiones excitadoras por 0,5, deprimiéndolas, las conexiones inhibitoras se potencian multiplicándolas por 1,5. Estos valores numéricos indicados fueron obtenidos tras numerosas pruebas y confirmados por Alfonso Araque *et al.* En el nuevo algoritmo glial simula el comportamiento observado en el hipocampo de rata, donde los astrocitos producen, en las colaterales de Schaffer, tanto potenciación a corto plazo (Alfonso Araque & Navarrete, 2010; Fiacco & McCarthy, 2004; Perea & Araque, 2007) como depresión a corto plazo (Serrano et al., 2006). Para simular este comportamiento, se ha creado un parámetro, denominado modificador glial, asociado a cada astrocito. Se han desarrollado dos versiones del algoritmo, una basada en el algoritmo previo de potenciación y otro en el de depresión:

- Algoritmo de potenciación: En el caso del algoritmo de potenciación, cuando el astrocito se activa, se multiplica por 1,25 el modificador glial, que actúa sobre la salida de la neurona, aumentando el valor de la salida de la neurona. Si el astrocito no se activa, se multiplica por 0,5 el modificador glial, reduciendo el valor de la salida de la neurona.
- Algoritmo de depresión: En el caso del algoritmo de depresión, cuando el astrocito se activa, se multiplica por 0,5 el modificador glial, que actúa sobre la salida de la neurona, reduciendo el valor de la salida de la neurona. Si el astrocito no se activa, se multiplica por 1,25 el modificador glial, aumentando el valor de la salida de la neurona.

5.3.5. Análisis y Organización de los Resultados

Para el análisis de los resultados obtenidos en la fase de entrenamiento, validación y test se ha empleado un programa de Matlab desarrollado por el grupo RNASA-IMEDIR que, junto con una plantilla de Excel, permite generar un único archivo con todos los resultados

de cada prueba a partir de los archivos de resultados. Para facilitar la visualización y el análisis de los resultados, en estos archivos se generan tablas con los resultados organizados.

5.3.6. Validación del Sistema

Las diferencias entre los resultados obtenidos por las RNA y las RNGA se han analizado aplicando el test de Wilcoxon de rangos asignados. Se ha realizado un contraste de hipótesis mediante este test estadístico para establecer si las diferencias entre los resultados obtenidos por ambos tipos de redes son o no estadísticamente significativos. A continuación, se introduce brevemente los métodos estadísticos empleados.

Una hipótesis estadística es una conjetura sobre uno o varios parámetros de interés de un modelo de probabilidad. Para comprobar si la conjetura puede ser acertada con cierto nivel de confianza, se realiza un contraste de hipótesis a partir de los datos muestrales obtenidos que permite aceptar o rechazar la hipótesis planteada, o Hipótesis Nula (H_0). Existe una Hipótesis Alternativa (H_1) que se acepta si se rechaza H_0 . Se asume que la hipótesis correcta es H_0 y que no es necesario probarla. Las hipótesis estadísticas pueden ser:

- Paramétricas: se trata de afirmaciones sobre valores de los parámetros poblacionales desconocidos. Es simple si especifica un único valor para cada parámetro poblacional desconocido, y compuesto si asigna un conjunto de valores posibles a parámetros poblacionales desconocidos.
- No paramétricas: en este caso son conjeturas sobre características estadísticas de la población de estudio.

Para realizar el contraste de hipótesis es necesario seleccionar un estadístico de contraste que mide la discrepancia entre la información de la muestra y la hipótesis H_0 . El estadístico de contraste debe seguir una distribución conocida cuando H_0 sea cierta, de tal modo que sea posible distinguir una discrepancia grande que tenga una probabilidad muy pequeña de ocurrir cuando H_0 sea cierta, y una discrepancia pequeña que tenga una probabilidad grande de ocurrir cuando H_0 sea cierta. A continuación, es necesario establecer el valor o nivel de significación (α) del contraste de hipótesis, donde $\alpha = P(\text{rechazar } H_0 | H_0 \text{ es cierta})$. Se calcula el valor del estadístico d^\wedge asociado a la muestra (valor crítico del contraste), si d^\wedge es pequeño (pertenecer a la región de aceptación) se acepta H_0 , y si d^\wedge es grande (pertenecer a la región de rechazo) se rechaza H_0 .

El p-valor, o nivel crítico, es la probabilidad de obtener una discrepancia con H_0 mayor o igual que el valor crítico d^\wedge cuando H_0 es correcta: si $\alpha \geq \text{p-valor}$ entonces se rechaza H_0 .

Generalmente, un p-valor menor de 0,01 permite rechazar H_0 , si es mayor de 0,1 se debe aceptar H_0 , y entre 0,01 y 0,1 no se considera concluyente.

Se trata de una hipótesis estadística paramétrica y compuesta, que determina un contraste de hipótesis unilateral por la derecha (de una cola). Las muestras son pareadas ya que aparecen como distintas observaciones realizadas sobre los mismos datos. En concreto, las observaciones surgen de las 100 combinaciones del *cross-validation*, de los 10 conjuntos de datos por las 10 poblaciones iniciales. A estas 100 combinaciones se les aplican las RNA (variable X) y las RNGA (variable Y) como sistemas de resolución. No se deben considerar a X e Y como variables independientes ya que existe una dependencia clara entre las dos variables. Si se denomina d_i a la diferencia entre las observaciones antes y después de incluir la glía artificial ($d_i = x_i - y_i$) siendo μ la media de la muestra, H_0 sería igual

a:

$$H_0: \mu_x = \mu_y \quad \leftrightarrow \quad \mu_x - \mu_y = 0 \quad \leftrightarrow \quad \mu_d = 0$$

Para contrastar que las RNGA obtienen mejores resultados que las RNA, se pueden emplear diferentes test de hipótesis, para este trabajo se ha seleccionado el test no paramétrico de Wilcoxon de rangos signados.

El contraste de Wilcoxon (Wilcoxon, 1945) es una técnica no paramétrica adecuada cuando la suposición de normalidad no es válida. Las dos arquitecturas empleadas, RNA y RNGA, producen unos resultados que se consideran apareados ya que emplean la misma población inicial y la misma distribución del conjunto de datos. Por tanto, existen 100 parejas de valores (X_i, Y_i) generados por la arquitectura sin glía y con glía. Para poder realizar un test paramétrico sería necesario que existiese normalidad de las diferencias pareadas d_i , en este caso no es así, por lo que es necesario realizar el test no paramétrico de Wilcoxon de rangos signados. A continuación se presentan los pasos necesarios para realizarlo (Moore & McCabe, 1989):

1. Ordenar todos los valores absolutos de las diferencias ($|d_i|$) entre las observaciones pareadas de menor a mayor.
2. Asignar a la diferencia en valor absoluto más pequeña rango 1, a la siguiente en tamaño, 2, y así sucesivamente, sin tener en cuenta el signo. Cuando el valor absoluto de dos o más diferencias sea el mismo, se asignará a cada uno el promedio de los rangos que se asignarían si las diferencias se distinguieran.
3. A cada rango ($r(|d_i|)$: valor del orden del dato en el conjunto) se le asigna el signo de la diferencia.

4. Calcular la suma de rangos positivos por un lado (w_+), y la de rangos negativos por otro (w_-).
5. La suma de los rangos con diferencias negativas omitido su signo es el estadístico de contraste que se suele denotar con la letra W (o T).

Las hipótesis empleadas por este contraste para determinar si existen diferencias entre las muestras son:

- H_0 : La media del error obtenido por las RNGA es igual o mayor al obtenido por las RNA.
- H_1 : La media del error obtenido por las RNGA es menor al obtenido por las RNA.

Cuanto mayor será la diferencia de las sumas, mayor será la evidencia de que las dos poblaciones son distintas. En este caso, la hipótesis nula se puede rechazar a favor de la alternativa sólo si W_+ es pequeña y W_- es grande.

Para realizar el test de Wilcoxon de los rangos signados se ha empleado el paquete MASS de R (Ripley et al., 2012).

5.4. Resultados

En este apartado se presentan los resultados obtenidos para cada problema y se realiza la comparación entre las RNA y las RNGA con las arquitecturas de una, dos y tres capas ocultas. En el caso de las RNGA se presentan los resultados obtenidos con el algoritmo de depresión y con el algoritmo de potenciación. También se comparan los resultados entre el algoritmo glial implementado, que actúa sobre las salidas de las neuronas, y el algoritmo glial previo, que modifica los pesos de las conexiones, esta comparación se realiza con la arquitectura de una capa oculta.

Antes de presentar los resultados, conviene aclarar los siguientes aspectos, algunos de los cuales ya han sido detallados previamente:

1. Para cada problema y cada arquitectura, la comparación entre las RNA y las RNGA se presenta como el porcentaje de acierto medio de las 100 combinaciones, en los conjuntos de entrenamiento, validación y test. Las 100 combinaciones se generan de combinar los 10 conjuntos distintos de patrones, usando la técnica 5x2cv y empleando 10 poblaciones distintas de pesos iniciales.
2. Para cada problema se ha establecido un número máximo de generaciones, o “tiempo de parada”, que ha sido superior para las RNA ya que necesitan un mayor número de iteraciones para alcanzar la configuración de pesos óptima.

3. El entrenamiento de las RNGA se realiza empleando cuatro combinaciones gliales de activación-iteración (2-4, 2-6, 3-6 y 3-8). En las tablas se presentarán tanto los valores medios para todas las combinaciones, como el valor de cada combinación glial de manera independiente.
4. En los gráficos de barras se simboliza con asterisco el nivel de significación obtenido al realizar el test no paramétrico de Wilcoxon de rangos signados para las diferencias en el porcentaje de acierto medio alcanzado por las RNA y las RNGA. En función del p-valor obtenido en el test, se establece la siguiente correspondencia:
p-valor < 0,0025 * 3 sigmas, las diferencias son significativas;
p-valor < 0,00005 ** 4 sigmas, las diferencias son muy significativas;
p-valor < 0,0000005 *** 5 sigmas, las diferencias son extremadamente significativas;
5. En los gráficos que se presente el porcentaje de acierto medio, se mostrará el valor de la media de los 100 casos ($n=100$) \pm el Error estándar medio (E.E.M).

A continuación, se presentan los resultados obtenidos para los problemas de diagnóstico del cáncer de mama, señales de ionosfera, señales de sonar, diagnóstico de diabetes y setas venenosas. Se muestra para cada problema la comparación de las RNGA con el algoritmo de depresión frente a las RNA, y la comparación de las RNGA con el algoritmo de potenciación frente a las RNA. además, se presenta la comparativa de los resultados obtenidos con las nuevas RNGA que modifican las salidas de las neuronas frente a las RNGA en las cuales se actúa en los pesos de las conexiones.

5.4.1. Resultados Depresión

5.4.1.1. Diagnóstico del Cáncer de Mama

En la tabla 6, se presentan los resultados obtenidos por las RNA y las RNGA, con el algoritmo de depresión, en el problema de diagnóstico del cáncer de mama, en el conjunto de entrenamiento, validación y test. En este problema se han probado las siguientes arquitecturas: una capa oculta con 7 neuronas, dos capas ocultas que tienen 7 neuronas en una capa y 5 en la otra y 3 capas ocultas que tienen 12 en la primera, 8 en la capa intermedia y 4 en la última. Los resultados que se muestran son la media del porcentaje de acierto para las 10 poblaciones iniciales y los 10 conjuntos de datos. Los resultados de las RNGA se presentan de manera independiente para las combinaciones de activación-iteración y para la mejor combinación.

En este caso los mejores resultados, tanto en el entrenamiento, validación y test, los obtiene la RNGA con la arquitectura de tres capas ocultas, superando el 98% de acierto medio tanto en el entrenamiento como en la validación, y superando el 97% en el test. Aunque con las arquitecturas de una y dos capas ocultas las RNA obtiene mejores resultados que las RNGA.

Tabla 6. Resultados en el problema de diagnóstico del cáncer de mama de la RNGA con el algoritmo de depresión y la RNA, obtenidos en el entrenamiento, validación y test, con las arquitecturas de una, dos y tres capas ocultas. En rojo se muestran los mejores resultados con esa arquitectura, y en negro las combinaciones de RNGA que obtiene mejores resultados que las RNA.

% medio de acierto			RNGA					RNA
			24	26	36	38	MEJOR	
CÁNCER	Entrenamiento	1 capa [7]	94,87	94,87	94,34	94,27	96,47	97,19
		2 capas [7_5]	94,80	95,26	94,84	95,09	96,18	97,64
		3 capas [12_8_4]	96,37	97,38	97,68	96,13	98,05	97,66
	Validación	1 capa [7]	94,84	95,09	94,94	94,39	96,95	97,65
		2 capas [7_5]	95,40	95,15	95,41	95,13	97,18	98,14
		3 capas [12_8_4]	96,73	97,30	98,19	95,82	98,68	98,52
	Test	1 capa [7]	92,74	93,19	92,67	92,40	95,12	96,01
		2 capas [7_5]	93,25	93,43	93,33	93,42	95,47	96,68
		3 capas [12_8_4]	95,34	95,95	96,58	94,54	97,37	96,95

5.4.1.2. Señales de Ionosfera

En la tabla 7 se muestran los resultados de las RNA y las RNGA, con el algoritmo de depresión, para el problema de señales de la ionosfera. En este caso las redes de una capa oculta tienen 9 neuronas, las de dos capas ocultas tienen 9 en una capa y 4 en la otra y las de 3 capas ocultas tienen 12 en la primera, 8 en la capa intermedia y 4 en la última.

En este problema las RNGA consiguen mejores resultados para todas las arquitecturas tanto en la validación como en el test. El mejor resultado en el test se obtiene con la arquitectura de dos capas ocultas con un abierto medio de 81,22%, cuatro puntos porcentuales superior a la mejor RNA. En el entrenamiento la RNGA obtiene mejores resultados con la arquitectura de una y dos capas ocultas, y la RNA con la de tres capas ocultas.

Tabla 7. Resultados en el problema de señales de la ionosfera de la RNGA con el algoritmo de depresión y la RNA, obtenidos en el entrenamiento, validación y test, con las arquitecturas de una, dos y tres capas ocultas. En rojo se muestran los mejores resultados con esa arquitectura, y en negro las combinaciones de RNGA que obtiene mejores resultados que las RNA.

% medio de acierto		RNGA					RNA	
		24	26	36	38	MEJOR		
IONOSFERA	1 capa [9]	Entrenamiento	86,22	85,42	84,85	87,98	89,34	86,37
	2 capas [9_4]		85,52	85,89	83,84	87,19	89,31	86,71
	3 capas [12_8_4]		84,83	85,40	85,58	85,13	88,03	88,06
	1 capa [9]	Validación	89,20	88,33	88,81	90,23	92,44	90,46
	2 capas [9_4]		88,99	87,98	88,06	88,70	91,93	89,54
	3 capas [12_8_4]		89,39	88,54	88,86	89,26	92,66	90,94
	1 capa [9]	Test	72,78	71,90	71,12	76,00	79,74	76,40
	2 capas [9_4]		74,84	78,64	75,38	75,37	81,22	75,66
	3 capas [12_8_4]		72,51	73,45	73,84	73,69	79,85	76,83

5.4.1.3. Señales de Sonar

Los resultados de la RNA y la RNGA con el algoritmo de depresión en el problema de señales de sonar se presentan en la tabla 8. En este problema las arquitecturas de una capa oculta tienen 6 neuronas, las de dos capas ocultas tienen 6 en una capa y 4 en la otra y las de 3 capas ocultas tienen 12 en la primera, 8 en la capa intermedia y 4 en la última.

En este problema los mejores resultados en la validación y el test los obtiene las RNGA, alcanzando un 78,49% en test, mejorando en casi un 5% el mejor acierto medio obtenido por las RNA, con un 73,90% de acierto medio.

Tabla 8. Resultados en el problema de señales de sonar de la RNGA con el algoritmo de depresión y la RNA, obtenidos en el entrenamiento, validación y test, con las arquitecturas de una, dos y tres capas ocultas. En rojo se muestran los mejores resultados con esa arquitectura, y en negro las combinaciones de RNGA que obtiene mejores resultados que las RNA.

% medio de acierto			RNGA					RNA
			24	26	36	38	MEJOR	
SÓNAR	1 capa [6]	Entrenamiento	83,24	82,63	81,97	83,58	84,98	85,53
	2 capas [6_4]		82,19	82,37	83,47	82,79	85,24	84,82
	3 capas [12_8_4]		82,03	82,92	83,45	81,60	85,42	87,06
	1 capa [6]	Validación	74,56	73,74	74,30	73,79	79,42	74,67
	2 capas [6_4]		72,21	71,37	72,63	70,47	77,47	73,79
	3 capas [12_8_4]		71,21	71,91	73,19	70,88	77,72	75,42
	1 capa [6]	Test	72,44	72,24	71,17	71,22	78,49	73,24
	2 capas [6_4]		70,93	69,34	70,83	69,34	76,46	72,59
	3 capas [12_8_4]		69,95	69,54	69,95	68,85	75,61	73,90

5.4.1.4. Diagnóstico de Diabetes

En la tabla 9 se presentan los resultados de la RNA y la RNGA con el algoritmo de depresión en el problema de diagnóstico de diabetes, para el cual se han empleado una arquitectura de una capa oculta que tiene 4 neuronas, la de dos capas ocultas tiene 6 en una capa y 4 en la otra y la de 3 capas ocultas tiene 12 en la primera, 8 en la capa intermedia y 4 en la última.

En este problema la RNA obtiene mejores resultados en entrenamiento, validación y test con todas las arquitecturas, excepto con tres capas en el conjunto de test.

Tabla 9. Resultados en el problema de diagnóstico de diabetes de la RNGA con el algoritmo de depresión y la RNA, obtenidos en el entrenamiento, validación y test, con las arquitecturas de una, dos y tres capas ocultas. En rojo se muestran los mejores resultados con esa arquitectura, y en negro las combinaciones de RNGA que obtiene mejores resultados que las RNA.

% medio de acierto		RNGA					RNA	
		24	26	36	38	MEJOR		
DIABETES	1 capa [4]	Entrenamiento	72,91	72,47	72,24	72,97	74,98	75,79
	2 capas [6_4]		71,59	72,70	71,36	71,97	73,61	75,61
	3 capas [12_8_4]		73,00	73,32	73,02	73,55	74,76	75,09
	1 capa [4]	Validación	69,28	68,75	68,70	69,05	71,92	72,72
	2 capas [6_4]		68,10	69,11	68,17	67,93	70,44	72,05
	3 capas [12_8_4]		68,54	68,23	68,23	68,59	71,26	71,43
	1 capa [4]	Test	66,28	65,76	65,50	65,92	69,27	70,12
	2 capas [6_4]		65,56	65,96	65,94	65,82	68,16	68,19
	3 capas [12_8_4]		65,52	65,63	65,12	65,58	68,49	68,38

5.4.1.5. Setas Venenosas

Los resultados del problema de las setas venenosas se presentan en la tabla 10, mostrándose la comparación de la RNA frente a la RNGA con el algoritmo de depresión. En este problema, las redes de una capa oculta tienen 7 neuronas, las de dos capas ocultas tienen 12 en una capa y 7 en la otra y las de 3 capas ocultas tienen 12 en la primera, 8 en la capa intermedia y 4 en la última.

En la tabla 10 se puede observar que la RNGA obtiene mejores resultados para todas las arquitecturas en el entrenamiento, validación y test, superando el 98,5% de acierto medio en el test.

Tabla 10. Resultados en el problema de setas venenosas de la RNGA con el algoritmo de depresión y la RNA, obtenidos en el entrenamiento, validación y test, con las arquitecturas de una, dos y tres capas ocultas. En rojo se muestran los mejores resultados con esa arquitectura, y en negro las combinaciones de RNGA que obtiene mejores resultados que las RNA.

% medio de acierto			RNGA					RNA
			24	26	36	38	MEJOR	
SETAS	1 capa [7]	Entrenamiento	97,12	96,96	96,77	96,77	98,57	98,10
	2 capas [12_7]		95,42	95,55	95,78	95,65	97,61	97,36
	3 capas [12_8_4]		95,48	95,85	95,63	95,96	97,69	97,23
	1 capa [7]	Validación	96,96	96,78	96,53	96,49	98,50	97,97
	2 capas [12_7]		95,05	95,07	95,40	95,31	97,30	97,21
	3 capas [12_8_4]		94,89	95,26	95,00	95,44	97,33	97,00
	1 capa [7]	Test	96,72	97,00	97,10	97,28	98,59	98,26
	2 capas [12_7]		96,88	97,13	97,09	96,52	98,75	97,85
	3 capas [12_8_4]		96,86	97,14	96,96	97,02	98,55	98,40

5.4.2. Resultados Potenciación

5.4.2.1. Diagnóstico del Cáncer de Mama

En la tabla 11 se muestran los resultados obtenidos por las RNA y las RNGA con el algoritmo de potenciación. En este problema las RNGA obtienen mejores resultados con todas las arquitecturas en el entrenamiento, validación y test. Los mejores resultados los obtiene la arquitectura de una capa oculta con un 97,67% de acierto medio. La mejor combinación de activación-iteración de las RNGA es la 24 que obtiene un 97,11%.

Tabla 11. Resultados en el problema de diagnóstico del cáncer de mama de la RNGA con el algoritmo de potenciación y la RNA, obtenidos en el entrenamiento, validación y test, con las arquitecturas de una, dos y tres capas ocultas. En rojo se muestran los mejores resultados con esa arquitectura, y en negro las combinaciones de RNGA que obtiene mejores resultados que las RNA.

% medio de acierto			RNGA					RNA
			24	26	36	38	MEJOR	
CÁNCER	1 capa [7]	Entrenamiento	97,82	97,87	97,63	97,74	97,98	97,19
	2 capas [7_5]		97,00	97,39	97,24	97,32	97,74	97,64
	3 capas [12_8_4]		96,37	97,38	97,68	96,13	98,05	97,66
	1 capa [7]	Validación	98,52	98,62	98,64	98,39	99,00	97,65
	2 capas [7_5]		97,08	97,57	97,66	97,31	98,53	98,14
	3 capas [12_8_4]		96,73	97,30	98,19	95,82	98,68	98,52
	1 capa [7]	Test	97,11	96,98	96,95	96,83	97,67	96,01
	2 capas [7_5]		95,65	96,18	96,14	95,74	97,15	96,68
	3 capas [12_8_4]		95,31	95,95	96,58	94,54	97,37	96,95

5.4.2.2. Señales de Ionosfera

Los resultados obtenidos por la RNA y la RNGA con el algoritmo de potenciación para el problema de señales de ionosfera se muestran en la tabla 12. La RNGA obtiene los mejores resultados, en concreto con la arquitectura de dos capas ocultas obtiene un 80,89% de acierto medio en test, frente a la RNA que obtiene el mejor resultado de 76,83% con tres capas ocultas.

Tabla 12. Resultados en el problema de señales de la ionosfera de la RNGA con el algoritmo de potenciación y la RNA, obtenidos en el entrenamiento, validación y test, con las arquitecturas de una, dos y tres capas ocultas. En rojo se muestran los mejores resultados con esa arquitectura, y en negro las combinaciones de RNGA que obtiene mejores resultados que las RNA.

% medio de acierto			RNGA					RNA
			24	26	36	38	MEJOR	
IONOSFERA	1 capa [9]	Entrenamiento	84,82	85,27	85,11	86,36	88,68	86,37
	2 capas [9_4]		85,52	85,89	83,84	87,21	89,31	86,71
	3 capas [12_8_4]		82,51	85,56	83,94	85,07	86,84	88,06
	1 capa [9]	Validación	89,37	89,87	89,51	90,39	92,59	90,46
	2 capas [9_4]		88,99	87,98	88,06	88,71	91,93	89,54
	3 capas [12_8_4]		88,64	90,43	89,86	90,22	93,00	90,94
	1 capa [9]	Test	70,16	68,20	70,02	66,42	72,37	76,40
	2 capas [9_4]		74,75	74,75	74,32	75,38	80,89	75,66
	3 capas [12_8_4]		71,67	73,84	72,22	74,60	79,49	76,83

5.4.2.3. Señales de Sonar

En la tabla 13 se muestran los resultados obtenidos por las RNA y las RNGA con el algoritmo de potenciación para el problema de las señales de sonar. Los mejores resultados los obtiene la RNGA con la arquitectura de una capa oculta, alcanzando un 76,66% de acierto medio en test, frente al mejor resultado de la RNA de 73,90% con tres capas ocultas.

Tabla 13. Resultados en el problema de señales de sonar de la RNGA con el algoritmo de potenciación y la RNA, obtenidos en el entrenamiento, validación y test, con las arquitecturas de una, dos y tres capas ocultas. En rojo se muestran los mejores resultados con esa arquitectura, y en negro las combinaciones de RNGA que obtiene mejores resultados que las RNA.

% medio de acierto			RNGA					RNA
			24	26	36	38	MEJOR	
SÓNAR	1 capa [6]	Entrenamiento	85,90	82,34	82,48	80,45	85,68	85,53
	2 capas [6_4]		82,40	79,84	81,74	81,34	84,26	84,82
	3 capas [12_8_4]		81,98	79,42	83,39	81,58	83,87	87,06
	1 capa [6]	Validación	74,23	72,07	71,02	70,95	77,67	74,67
	2 capas [6_4]		73,16	70,98	71,26	70,28	76,93	73,79
	3 capas [12_8_4]		71,14	70,42	71,49	70,79	76,72	75,42
	1 capa [6]	Test	71,51	69,76	68,61	69,39	76,66	73,24
	2 capas [6_4]		70,29	68,20	69,59	65,98	75,15	72,59
	3 capas [12_8_4]		67,27	67,29	67,54	68,00	73,59	73,90

5.4.2.4. Diagnóstico de Diabetes

Los resultados de las RNA y la RNGA con el algoritmo de potenciación para el problema del diagnóstico de diabetes se muestran en la tabla 14. Se observa que las RNGA obtienen mejores resultados con todas las arquitecturas en el entrenamiento, validación y test. Se alcanza un 75,36% de acierto medio en test, un 4% más que el mejor resultado de la RNA con un 70,12%. Las mejores combinaciones de activación-iteración de la RNGA son la 24 y 36, las cuales superan en el test los resultados de las RNA.

Tabla 14. Resultados en el problema de diagnóstico de diabetes de la RNGA con el algoritmo de potenciación y la RNA, obtenidos en el entrenamiento, validación y test, con las arquitecturas de una, dos y tres capas ocultas. En rojo se muestran los mejores resultados con esa arquitectura, y en negro las combinaciones de RNGA que obtiene mejores resultados que las RNA.

% medio de acierto			RNGA					RNA
			24	26	36	38	MEJOR	
DIABETES	1 capa [4]	Entrenamiento	77,93	75,79	77,86	73,03	78,20	75,79
	2 capas [6_4]		77,24	74,67	75,51	72,98	77,77	75,61
	3 capas [12_8_4]		74,82	74,82	74,93	73,99	76,87	75,09
	1 capa [4]	Validación	74,84	72,19	73,71	69,27	76,14	72,72
	2 capas [6_4]		73,72	70,71	71,85	69,22	75,05	72,05
	3 capas [12_8_4]		71,25	70,77	72,91	69,05	74,34	71,43
	1 capa [4]	Test	72,73	71,68	73,00	69,37	75,36	70,12
	2 capas [6_4]		71,58	67,51	69,25	65,90	73,47	68,19
	3 capas [12_8_4]		70,16	68,20	70,02	66,42	72,37	68,38

5.4.2.5. Setas Venenosas

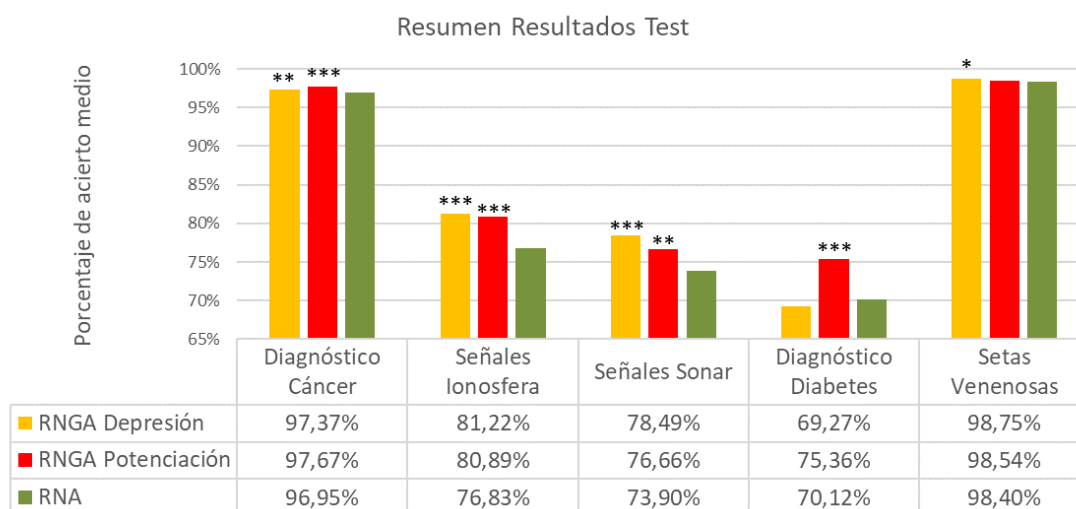
En la tabla 15 se muestran los resultados de la RNA y la RNGA con el algoritmo de potenciación. El mejor resultado en el test lo obtiene la RNGA con la arquitectura de dos capas ocultas con un 98,54% de acierto medio. Con una capa y tres capas ocultas, las RNA obtiene mejores resultados que las RNGA. En el entrenamiento y la validación la RNGA obtiene mejores resultados con todas las arquitecturas.

Tabla 15. Resultados en el problema de setas venenosas de la RNGA con el algoritmo de potenciación y la RNA, obtenidos en el entrenamiento, validación y test, con las arquitecturas de una, dos y tres capas ocultas. En rojo se muestran los mejores resultados con esa arquitectura, y en negro las combinaciones de RNGA que obtiene mejores resultados que las RNA.

% medio de acierto			RNGA					RNA
			24	26	36	38	MEJOR	
SETAS	1 capa [7]	Entrenamiento	97,64	97,81	97,84	97,77	98,82	98,10
	2 capas [12_7]		97,17	96,76	96,70	96,86	98,41	97,36
	3 capas [12_8_4]		96,78	96,34	96,15	96,60	97,98	97,23
	1 capa [7]	Validación	97,32	97,38	97,49	97,40	98,66	97,97
	2 capas [12_7]		96,68	96,23	96,31	96,43	98,11	97,21
	3 capas [12_8_4]		96,18	95,71	95,56	95,92	97,58	97,00
	1 capa [7]	Test	97,11	96,98	96,96	96,83	97,67	98,26
	2 capas [12_7]		96,86	97,14	96,96	97,02	98,54	97,85
	3 capas [12_8_4]		97,14	96,64	96,70	97,10	98,38	98,40

5.4.3. Resumen Resultados

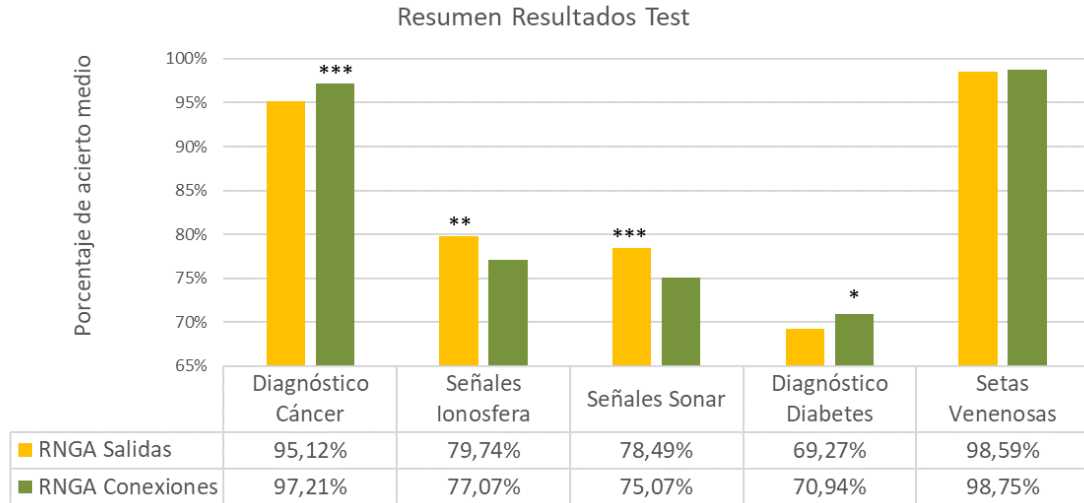
En la gráfica 1 se muestran el resumen de los resultados obtenidos por la RNA y los dos algoritmos de las RNGA en los cinco problemas analizados. Se puede observar que en cuatro de los cinco problemas analizados las RNGA, tanto el algoritmo de depresión como el de potenciación, obtienen una mejoría muy significativa en el porcentaje de acierto en el test. En el problema de las setas venenosas la mejora es significativa únicamente para el algoritmo de depresión.



Gráfica 1. Resumen de los resultados obtenidos en el test por la RNA (en verde) y RNGA con el algoritmo de depresión (en amarillo) y potenciación (en rojo) para los cinco problemas analizados. El nivel de significación se muestra con asteriscos: * equivale a un p-valor < 0,0025 o 3 sigmas, las diferencias son significativas; ** equivale a un p-valor < 0,00005 o 4 sigmas, las diferencias son muy significativas; y *** equivale p-valor < 0,0000005 o 5 sigmas, las diferencias son extremadamente significativas.

5.4.4. Comparación con Algoritmo de Conexiones

En la gráfica 2 se muestran los resultados obtenidos por la RNGA con la modulación que modifica los pesos de las conexiones y la nueva modulación implementada que actúa sobre el valor de la salida de las neuronas. En este caso se muestra la comparación únicamente con los resultados obtenidos con la red de una capa oculta y el algoritmo de depresión. Se observa que dependiendo del problema parece más adecuado un tipo de modulación u otra. En el caso de la modulación de las conexiones, se obtiene mejores resultados en los problemas del diagnóstico del cáncer de mama y diagnóstico de diabetes. La modulación de las salidas obtiene diferencias significativas en los problemas de señales de la ionosfera y señales de sonar.



Gráfica 2. Resumen de los resultados obtenidos en el test por la RGA que actúa sobre los pesos de las conexiones (verde) y la que actúa sobre las salidas de las neuronas (amarillo) para los cinco problemas analizados. El nivel de significación se muestra con asteriscos: * equivale a un p-valor < 0,0025 o 3 sigmas, las diferencias son significativas; ** equivale a un p-valor < 0,00005 o 4 sigmas, las diferencias son muy significativas; y *** equivale p-valor < 0,0000005 o 5 sigmas, las diferencias son extremadamente significativas.

5.5. Discusión

Los resultados obtenidos muestran la capacidad de las RGA de mejorar de manera significativa los resultados obtenidos por las RNA en problemas de clasificación (gráfica 1). En todos los problemas analizados, las RGA obtuvieron diferencias muy significativas con respecto a las RNA. En cuatro de los cinco problemas, el nivel de significación superó las 5 sigmas, es decir, que los resultados de las RGA son superiores a los de las RNA con una probabilidad del 99,99995%. En los problemas de diagnóstico del cáncer y de setas venenosas, la mejora no supera el 1% en el porcentaje de acierto ya que tanto las RNA como las RGA consiguen un porcentaje de acierto muy alto, a pesar de ello las diferencias son significativas estadísticamente. En los problemas de las señales de ionosfera, sonar y diagnóstico de diabetes, las RGA superan a las RNA en varios puntos porcentuales.

La comparación con la modulación glial desarrollada anteriormente (gráfica 2) muestra que dependiendo del problema es más adecuado un tipo de modulación u otra. Al igual que ocurre en el cerebro, en unos casos es más adecuado actuar sobre las conexiones y en otros sobre las salidas de las neuronas.

6. Deep Artificial Neuron-Astrocyte Network

En este capítulo se presenta la metodología desarrollada para el entrenamiento de redes de *Deep Learning* que incorporan neuronas y astrocitos, denominadas *Deep Artificial Neuron-Astrocyte Network* (DANAN). Para ello ha sido necesario diseñar una nueva metodología ya que las redes de *Deep Learning* se entrenan generalmente mediante el algoritmo de retropropagación del error. Hasta el momento los algoritmos gliales desarrollados por el grupo RNASA-IMEDIR sólo eran compatibles con el entrenamiento mediante algoritmos genéticos, no pudiendo emplearse con redes entrenadas con el algoritmo de retropropagación del error. La metodología desarrollada para la creación de DANAN consta de una primera fase de entrenamiento supervisado mediante el algoritmo de retropropagación del error y una fase posterior no supervisado en la que actúan los astrocitos artificiales.

El objetivo de este trabajo es probar la capacidad de los astrocitos artificiales de mejorar la eficacia de las redes de *Deep Learning*. Este objetivo no es sencillo ya que, como se ha mencionado anteriormente, en la última década se han realizado grandes avances en este campo, desarrollándose multitud de algoritmos y técnicas para mejorar los resultados, optimizando enormemente su funcionamiento. Sin embargo, esta prueba de concepto permitiría desarrollar futuros algoritmos gliales más complejos ya que existen más posibilidades de actuación en las redes de *Deep Learning* que cuentan con miles de neuronas artificiales.

El lenguaje de programación empleado para la implementación de las redes ha sido *Python*, se ha seleccionado este lenguaje por varias razones. En primer lugar, existen varios marcos de trabajo, o *frameworks*, desarrollados en este lenguaje que facilitan el diseño y entrenamiento de las redes de *Deep Learning*. El trabajo de esta tesis doctoral se ha realizado con *Keras* (Chollet, 2015), utilizando *TensorFlow* (Abadi et al., 2016) como *backend*. Además, al utilizar estos marcos de trabajo es posible paralelizar de manera más sencilla el entrenamiento de las redes en los procesadores de las GPU para reducir los tiempos de entrenamiento. Para el desarrollo de este trabajo se ha contado con los recursos proporcionados por el CESGA (“Centro de Supercomputación de Galicia (CESGA),” 2017), en este caso GPUs K40 y K80, y una GPU Nvidia Titan X proporcionada por Nvidia para el desarrollo de la línea de investigación de Ana Porto y el grupo RNASA-IMEDIR.

6.1. Diseño y Metodología

Para el diseño de la experimentación se han seguido los mismos pasos generales empleados en la metodología de las RNGA, que son los siguientes:

- Identificación del problema a resolver.
- Preparación de los datos.
- Diseño y construcción de la red.
- Fase de entrenamiento, validación y test.
- Organización y análisis de resultados.
- Validación del sistema.

6.1.1. Problemas a Resolver

Una de las dificultades que presenta el entrenamiento de las redes de *Deep Learning* es la necesidad de una gran cantidad de datos y de recursos computacionales para el entrenamiento. Por ello, la selección del problema ha sido complicada, ya que ha sido necesario encontrar un equilibrio entre la cantidad de datos suficiente para obtener buenos resultados, y el tiempo necesario para el entrenamiento de las redes.

Se ha seleccionado una base de datos generada mediante un proyecto del *Eurostat*, financiado por la Unión Europea (UE), denominada LUCAS (*Land Use and Cover Area frame Statistical survey*) (Tóth et al., 2013) con el objetivo de armonizar los datos recogidos sobre los usos de los suelos y sus propiedades. Los datos fueron obtenidos en un total de 23 países de la UE, con un total de 250.000 puntos de muestreo. En cada punto se recogieron 5 muestras de la capa superior del suelo, en los primeros 30 cm, siguiendo unos protocolos estandarizados. Todo el proceso ha sido asegurado por expertos del Joint Research Centre, y las muestras analizadas en el mismo laboratorio acreditado para evitar las discrepancias entre los laboratorios. A las muestras se les realizó un análisis espectral, además, se midió la concentración de diferentes elementos: carbono orgánico (CO), N, P, K, Ag, Al, etc. También se complementa con otros datos como el pH, la distribución del tamaño de las partículas, el tipo de suelo, y otros datos geográficos. La concentración de CO se ha medido mediante combustión seca (ISO 10694: 1995), se expresa en g/Kg y sus valores se distribuyen entre 0 a 600 g C/Kg.

De esta base de datos se han tomado los valores del análisis espectral, un total de 4200 valores por muestra y un total de 19036 muestras. La variable objetivo son los valores de la concentración de CO, se trata por tanto de un problema de regresión, en el cual a partir de

los valores del espectro hay que predecir la concentración de CO. Los valores del espectro se encuentran entre 0 y 1,5 por lo que no ha sido necesario normalizarlos. La variable objetivo, el contenido de CO expresado en g/Kg, varía entre 0 y 600, por lo que se ha realizado una normalización entre 0 y 1.

6.1.2. Preparación de los Datos

Para este problema los datos se han dividido mediante un 5 *folds crossvalidation* por lo que existen 5 conjuntos diferentes de entrenamiento, validación y test. Este número de conjuntos es suficiente ya que se dispone de una gran cantidad de datos por lo que no es necesario dividir los datos en 10 conjuntos como en los problemas empleados en las RNGA. Además, que consumiría mucho tiempo realizar el entrenamiento con 10 conjuntos diferentes. La división se ha realizado de tal manera que la distribución de los datos sea semejante en cada conjunto ya que al tratarse de un problema de regresión existen mayores diferencias entre los valores objetivo.

Los datos se han dividido en un 60% para entrenamiento, 11422 muestras, un 20% para validación y un 20% para test, 3807 muestras en cada caso.

6.1.3. Diseño y Construcción de la Red

Existe una gran variedad de tipos de redes de *Deep Learning*, por sencillez se ha realizado la experimentación con arquitecturas multicapa, totalmente conectadas hacia delante, sin conexiones recurrentes. En este caso se han seleccionado una arquitectura con una capa oculta y otra de tres capas ocultas para comparar la eficacia de la implementación de los astrocitos artificiales en redes de diferente complejidad. Previamente se realizaron pruebas con diferentes arquitecturas para escoger las dos RNA que obtuvieron los mejores resultados. La arquitectura de una capa oculta se compone de 1000 neuronas artificiales en la capa oculta y la arquitectura de tres capas ocultas consta de 1600 neuronas en la primera capa oculta, 800 en la segunda y 200 en la tercera.

A pesar de que el número de capas es el mismo que en el caso de las RNGA, el número de neuronas por capas es varios órdenes de magnitud superior. El número total de parámetros de las redes de tres capas es 8.162.801, en el caso de la arquitectura de una capa el número de parámetros es aproximadamente la mitad, 4.202.001 (tabla 16). Este tipo de redes no sería posible entrenarlas con los algoritmos genéticos, ni con los algoritmos de entrenamiento con retropropagación, es por ello que se emplean las técnicas de *Deep Learning* para permitir optimizar este gran número de parámetros de manera eficiente.

Tabla 16. Estructura en Keras de la arquitectura de la red de tres capas oculta.

Tipo de capa	Formato de salida de la capa	Número de parámetros	Conectada
<i>Dense 1</i>	(<i>None</i> , 1600)	6721600	<i>Input</i>
<i>Activation 1</i>	(<i>None</i> , 1600)	0	<i>Dense 1</i>
<i>Dropout 1</i>	(<i>None</i> , 1600)	0	<i>Activation 1</i>
<i>Dense 2</i>	(<i>None</i> , 800)	1280800	<i>Dropout 1</i>
<i>Activation 2</i>	(<i>None</i> , 800)	0	<i>Dense 2</i>
<i>Dropout 2</i>	(<i>None</i> , 800)	0	<i>Activation 2</i>
<i>Dense 3</i>	(<i>None</i> , 200)	160200	<i>Dropout 2</i>
<i>Activation 3</i>	(<i>None</i> , 200)	0	<i>Dense 3</i>
<i>Dropout 3</i>	(<i>None</i> , 200)	0	<i>Activation 3</i>
<i>Dense 3</i>	(<i>None</i> , 1)	201	<i>Dropout 3</i>

6.1.4. Fase de Entrenamiento, Validación y Test

Como se ha mencionado previamente las redes han sido entrenadas mediante el algoritmo de retropropagación. Los valores de todos los parámetros se han mantenido constantes en todas las pruebas. Se ha establecido un máximo de 100.000 épocas, en cada época se le presentan una vez el conjunto de patrones de entrenamiento. Para optimizar el entrenamiento, se han empleado la técnica *de mini-batches*. En lugar de presentar el conjunto completo de entrenamiento y calcular el error, se divide el conjunto de entrenamiento en 100 *mini-batches*, de tal manera que es más sencillo calcular el error. Para la optimización de los pesos de la red se ha empleado una técnica conocida como Adadelta (*An Adaptive Learning Rate Method*) (Zeiler, 2012), con un ratio de aprendizaje inicial de 0,001 y un ratio de aprendizaje final de 0,0001.

Por último, para evitar el sobreentrenamiento y favorecer la capacidad de generalización de la red se ha empleado la técnica de *Dropout* (Srivastava et al., 2014). Esta técnica consiste en que durante el entrenamiento se eliminan temporalmente y de manera aleatoria neuronas junto con sus conexiones (figura 22). El valor de *Dropout* es 0,5, es decir, existe un 50% de posibilidades de que una neurona de la red se elimine.

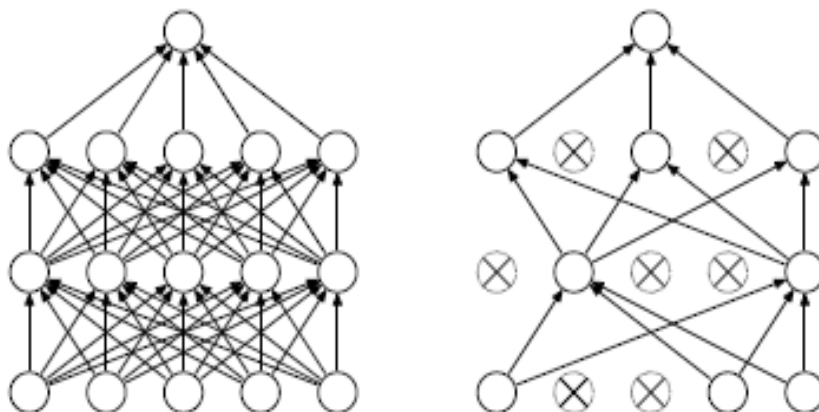


Figura 22. A la izquierda se muestra una RNA básica, a la derecha un ejemplo de la aplicación de la técnica de dropout.

El algoritmo de optimización tiene como función de costo, o *cost function*, el valor de la raíz cuadrada del error cuadrático medio, o *Root Mean Square Error* (RMSE). El RMSE se suele emplear en los problemas de regresión y frente al error cuadrático medio tiene la ventaja de magnificar y penalizar los errores grandes cometidos por el modelo.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{i=n} (y_i - \hat{y}_i)^2}{n}}$$

En la ecuación, y_i es el valor predicho para la observación i , \hat{y}_i es el valor real de la medida de la observación i y n es el número de observaciones. El valor de RMSE se expresa en la misma escala que los datos originales, por lo que es sencillo de interpretar, el único paso necesario para facilitar la interpretación es la desnormalización de los valores.

6.1.5. Algoritmo Astrocítico

Las DANAN tienen un entrenamiento híbrido, como se ha comentado, en primer lugar, hay una fase supervisada con el algoritmo de retropropagación del error y una segunda fase no supervisada mediante el algoritmo astrocítico. La fase de entrenamiento supervisado es similar al de las redes sin astrocitos, por lo que se toman los pesos obtenidos mediante el entrenamiento con el algoritmo de retropropagación. Sobre estos pesos se aplica el algoritmo astrocítico en una fase de entrenamiento no supervisada. El funcionamiento de los astrocitos artificiales se detalla en el pseudocódigo de la figura 23 y actúa de forma similar en la fase de entrenamiento, validación y test. Se realiza un número determinado de iteraciones con cada patrón y el astrocito contabiliza el número de veces que se activa la neurona. Si la neurona se activa n veces, el astrocito se activa y modifica el peso de las conexiones saliente por un valor. Después de realizar diversas pruebas, se han establecido

dos combinaciones de valores de activación-iteración que obtuvieron los mejores resultados, 2-6 y 3-6. También se probaron dos versiones del algoritmo, una de potenciación que aumentaba los pesos de las conexiones de las neuronas activas y reducía el de las neuronas no activas, y uno de depresión que actúa al revés, aumentando los pesos de las neuronas no activas y reduciendo el de las neuronas activas. Los mejores resultados se obtuvieron con el algoritmo de potenciación y con unos valores de 1,0025 para el aumento de los pesos de las conexiones y 0,995 para la reducción de los pesos de las conexiones. Estos valores se han obtenido tras realizar pruebas con varias combinaciones de estos parámetros del algoritmo astrocítico, basándose en el conocimiento previo y los trabajos previos del grupo RNASA-IMEDIR, dirigidos por Ana Porto y con la colaboración de Alfonso Araque.

```

(...)

Para cada patrón:
{
    Para cada iteración:
    {
        Para cada capas:
        {
            Para cada neurona:
            {
                Comprobar si la neurona se activa con el patron

                Si la neurona se activa:
                {
                    Sumar +1 al contador de activaciones

                    Si el contador es igual a m iteraciones:
                    {
                        Modificar los pesos de las conexiones
                    }
                    Si no:
                    {
                        No realizar ninguna modificación
                    }
                }
                Si no:
                {
                    No modificar el contador de activaciones
                }
            }
        }
    }
}

(...)

```

Figura 23. Pseudocódigo del funcionamiento de las DANAN.

6.1.6. Análisis y Organización de Resultados

Los resultados se han evaluado en función del valor del error cuadrático medio, explicado en el apartado anterior, y del coeficiente de determinación, o R^2 (Steel & Torrie, 1960). Esta medida determina la capacidad del modelo para predecir los resultados, es decir la proporción de la variación explicada por el modelo (Cameron & Windmeijer, 1997).

6.1.7. Validación del Sistema

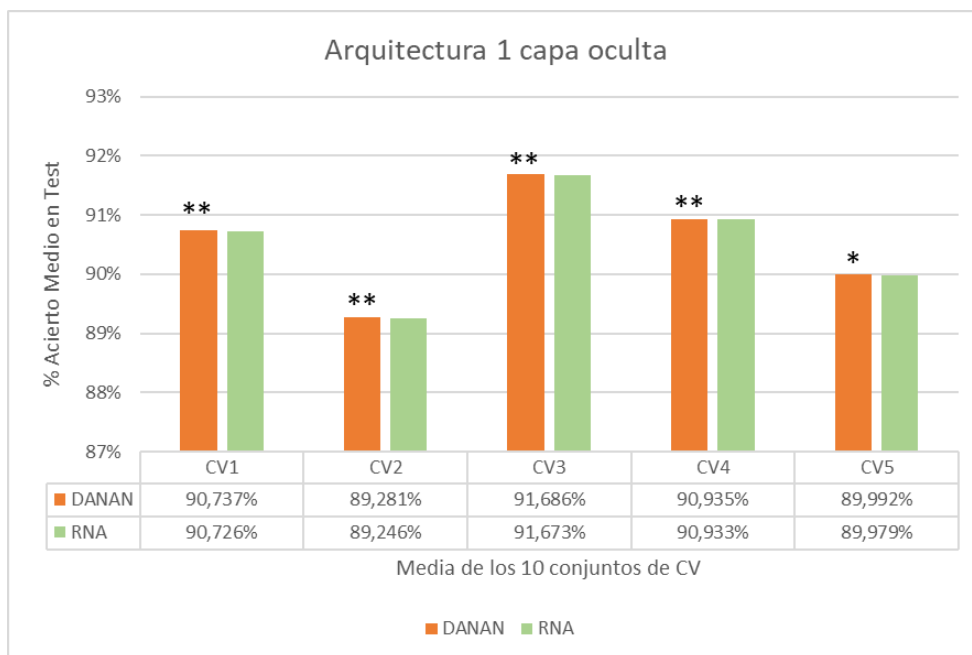
En este caso también se ha utilizado el test no paramétrico de Wilcoxon de rangos signados, el cual se ha explicado en el capítulo anterior. Sin embargo, el número de medidas es diferente ya que no se cuenta con 10 poblaciones iniciales y 10 conjuntos diferentes de test. Como se ha mencionado, se ha realizado un *5 folds crossvalidation*, para poder contar con medidas suficientes para realizar la comparación y que ésta pudiese ser significativa, se ha dividido cada conjunto de test en 10 subconjuntos. De esta manera se obtienen 50 medidas de R^2 para realizar el test no paramétrico de Wilcoxon de rangos signados

En este caso la hipótesis nula, H_0 , es que *“los resultados obtenidos por las RNA entrenadas con el algoritmo de retropropagación y técnicas de Deep Learning son mejores o iguales a los obtenidos por las DANAN”*.

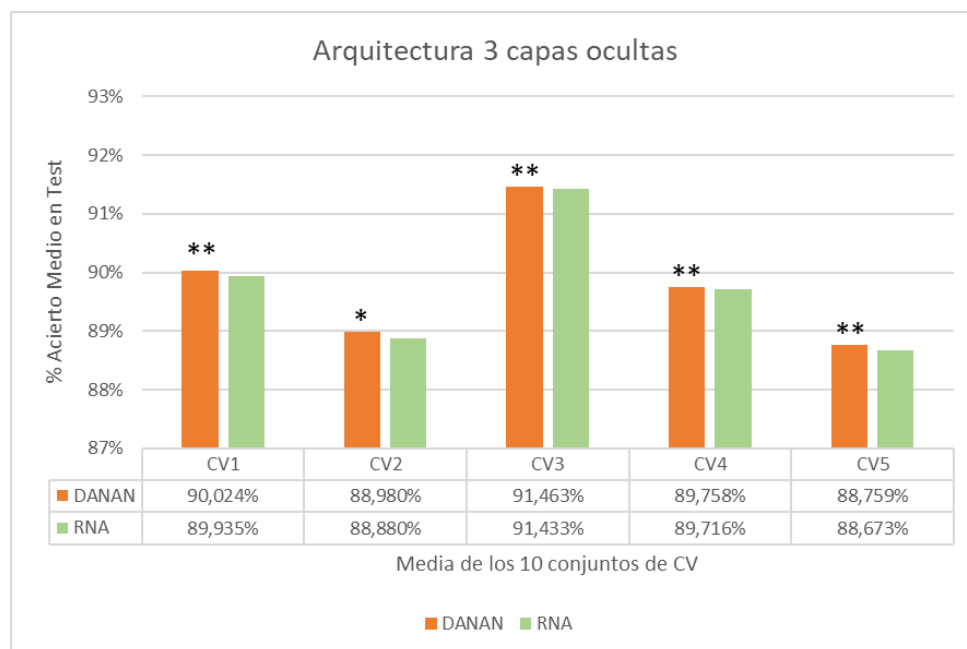
- H_0 : El R^2 obtenido por las DANAN es igual o menor al obtenido por las RNA.
- H_1 : La R^2 obtenido por las DANAN es mayor al obtenido por las RNA.

6.2. Resultados

Los resultados obtenidos con las DANAN tanto con la arquitectura de una capa oculta como la de tres capas ocultas mejoran los obtenidos con la RNA de manera significativa. En todos los conjuntos del *cross-validation* las DANAN obtienen mejores resultados para ambas arquitecturas. En las gráficas 3 y 4 se muestra el valor medio del R^2 para los 10 subconjuntos de test obtenidos para cada *fold* del *crossvalidation*.



Gráfica 3. Resumen de los resultados obtenidos en el test por la RNA (en verde) y la DANAN (en naranja) con la arquitectura de una capa oculta. El porcentaje de acierto representa la media obtenida en los 10 subconjuntos generados a partir de cada *fold* del *crossvalidation*. El nivel de significación se muestra con asteriscos: * equivale a un p-valor < 0,0025 o 3 sigmas, las diferencias son significativas y ** equivale a un p-valor < 0,00005 o 4 sigmas, las diferencias son muy significativas.



Gráfica 4. Resumen de los resultados obtenidos en el test por la RNA (en verde) y la DANAN (en naranja) con la arquitectura de tres capas ocultas. El porcentaje de acierto representa la media obtenida en los 10 subconjuntos generados a partir de cada *fold* del *crossvalidation*. El nivel de significación se muestra con asteriscos: * equivale a un p-valor < 0,0025 o 3 sigmas, las diferencias son significativas y ** equivale a un p-valor < 0,00005 o 4 sigmas, las diferencias son muy significativas.

El valor del test no paramétrico de Wilcoxon de rangos signados para la comparación de los resultados para una capa oculta da un p-valor de $4,399e-10$ (gráfica 3), y un p-valor de $4,965e-10$ para tres capas ocultas (gráfica 4). El test de Wilcoxon se realizó comparando los 50 valores de test obtenidos por las RNA y las DANAN con cada una de las arquitecturas. En base a los resultados obtenidos en el test de Wilcoxon, se rechaza la hipótesis nula y se acepta la hipótesis alternativa, “los resultados obtenidos con las DANAN son mejores que los obtenidos por las RNA”

6.3. Discusión

Los resultados obtenidos muestran la capacidad de las DANAN de mejorar los resultados, aunque las diferencias en el porcentaje de acierto son pequeñas, en todos los casos son significativas estadísticamente. Hay que señalar que, en esta implementación, la actuación de los astrocitos se realiza en redes con los pesos obtenidos en el entrenamiento con el algoritmo de retropropagación, esto permite ahorrar mucho tiempo, ya que no es necesario entrenar redes nuevas con el algoritmo astrocítico. Pero las diferencias obtenidas en el porcentaje de acierto entre las DANAN y las RNA son reducidas en comparación con los problemas analizados para las RGA. Esto se debe a que las redes tienen unos pesos muy similares, pero hay que destacar que en todos los casos la actuación de los astrocitos mejora los resultados y el test no paramétrico de Wilcoxon de rangos signados muestra un p-valor significativo.

Además, el tiempo de computación para la aplicación de los astrocitos es muy reducido, apenas unos minutos, comparado con el tiempo de entrenamiento completo de la red que puede durar varios días.

7. Conclusiones

Los algoritmos desarrollados se inspiran en los descubrimientos recientes sobre la actuación de las células gliales en el procesamiento de la información. En concreto, el concepto de la sinapsis tripartita sirve como base para los modos de actuación propuestos. Sin embargo, hay que señalar que los algoritmos desarrollados no tratan de reflejar fielmente el comportamiento biológico de las células gliales. El objetivo de esta investigación ha sido demostrar una posible vía en el desarrollo de las arquitecturas de los sistemas conexionistas, avanzado desde los modelos más sencillos de redes entrenadas mediante algoritmo genético, hasta la implementación en redes de *Deep Learning*.

En la primera fase del estudio, se desarrolló un modo de actuación de las células gliales, que modifican la salida de las neuronas en lugar de los pesos de las conexiones. Este comportamiento tiene una base neurocientífica, ya que ambos tipos de actuaciones han sido observadas en experimentos. La nueva versión de las RNGA ha sido evaluada en 5 problemas de clasificación: cáncer de mama, señales de ionosfera, señales de sonar, diagnóstico de diabetes y setas venenosas. Se han probado tres arquitecturas diferentes, con una, dos y tres capas ocultas. El análisis de los resultados de este estudio ha permitido extraer las siguientes conclusiones:

- Se ha demostrado que la eficiencia de las RNA multicapa, conectadas hacia delante, sin conexiones laterales, totalmente conectadas y entrenadas mediante algoritmos genéticos en problemas de clasificación, aumenta con la introducción de astrocitos artificiales que modifican la salida de las neuronas de manera no supervisada. Estos resultados concuerdan con los hallazgos científicos que demuestran que los astrocitos participan en el procesamiento de la información en el cerebro mediante la sinapsis tripartita, permitiendo una comunicación bidireccional con las neuronas. Los resultados experimentales obtenidos que avalan esta afirmación son:
 - En cuatro de los cinco problemas analizados (diagnóstico del cáncer de mama, señales de ionosfera, señales de sonar y setas venenosas), la mejor combinación de RNGA con el algoritmo de depresión obtiene resultados significativamente mejores que las RNA. En los problemas de señales de ionosfera y señales de sonar las diferencias han sido muy significativas, superiores a cinco sigmas. En el problema del diagnóstico de diabetes las diferencias no han sido significativas.

- En cuatro de los cinco problemas analizados (diagnóstico del cáncer de mama, señales de ionosfera, señales de sonar y diagnóstico de diabetes), la mejor combinación de RNGA con el algoritmo de potenciación obtiene resultados significativamente mejores que las RNA. En los problemas de diagnóstico del cáncer de mama, señales de ionosfera y diagnóstico de diabetes las diferencias han sido muy significativas, superando las cinco sigmas. En el problema de las setas venenosas las diferencias no han sido significativas.
- Las combinaciones de activación-iteración que obtienen mejores resultados depende del problema analizado y la arquitectura empleada. Lo cual está en consonancia con las observaciones de los diversos modos de interacción entre la glía y las neuronas y las variaciones observadas en las diferentes regiones cerebrales y estados del cerebro.
- El tipo algoritmo glial (potenciación o depresión) que obtiene mejores resultados depende del problema analizado. Estas diferencias han sido observadas en el cerebro, donde dependiendo de las condiciones y la región cerebral, la glía puede tanto potenciar como deprimir las sinapsis.
 - Los resultados obtenidos muestran que, el algoritmo de depresión obtiene mejores resultados en los problemas de diagnóstico del cáncer de mama, señales de sonar y setas venenosas. Mientras que, en los problemas de señales de ionosfera y diagnóstico de diabetes, el algoritmo de potenciación obtiene mejores resultados.
- El modo de actuación óptimo por parte de los astrocitos depende del problema analizado. En el caso de la actuación de la glía sobre los pesos de las conexiones, se obtiene mejores resultados en los problemas del diagnóstico del cáncer de mama y diagnóstico de diabetes. La actuación de la glía sobre las salidas de las neuronas obtiene mejores resultados en los problemas de señales de la ionosfera y señales de sonar. Estos resultados parecen coherentes con las evidencias neurocientíficas que muestran que los astrocitos pueden modificar tanto las conexiones sinápticas, como las neuronas.

La segunda fase del estudio se ha centrado en el desarrollo de una nueva metodología compatible con las redes entrenadas con el algoritmo de retropropagación del error y técnicas de *Deep Learning*. La metodología empleada hasta el momento por las RGA no era compatible con el entrenamiento mediante el algoritmo de retropropagación, lo que

impedía poder obtener resultados satisfactorios con redes de más de unas decenas de neuronas. Por ello, ha sido necesario el diseño de un nuevo modo de actuación de los astrocitos compatible con redes de miles de neuronas para poder resolver problemas de mayor complejidad. En concreto, la nueva implementación se ha evaluado en el problema de predicción de carbono orgánico en suelos a partir de señales del espectro NIR. Este problema tiene una gran complejidad ya que como datos de entrada se toman 4200 variables. Además, la variable objetivo es un valor continuo de concentración de carbono orgánico, por lo que se trata de un problema de regresión. El estudio realizado con las redes de Deep Learning permite extraer las siguientes conclusiones:

- La actuación de los astrocitos puede mejorar los resultados obtenidos con redes complejas entrenadas con el algoritmo de retropropagación y técnicas de *Deep Learning*. La implementación de las DANAN muestra un modo de actuación de los astrocitos rápido y sencillo, pero a la vez eficaz, que permite mejorar los resultados de las redes de *Deep Learning*. La metodología desarrollada para la implementación de los astrocitos es un primer paso en la integración con redes más complejas. La diferencia fundamental con los trabajos previos es que esta actuación de los astrocitos es compatible con los algoritmos de retropropagación.
- La actuación de los astrocitos es compatible con redes de mayor tamaño lo que permite resolver problemas más complejos. La nueva arquitectura denominada DANAN abre la vía de la integración con otros tipos de redes de Deep Learning como las redes de convolución, las redes recurrentes.

8. Futuros Desarrollos

Los resultados obtenidos con las RNGA y las DANAN, permiten extraer unas conclusiones que refuerzan las hipótesis neurocientíficas en las que se ha basado su desarrollo, a la vez que aportan más evidencia sobre la utilidad de la incorporación de astrocitos artificiales en los modelos de Inteligencia Artificial. El trabajo desarrollado en esta tesis abre nuevas líneas de investigación al implementar un modo de actuación de los astrocitos que es compatible con el algoritmo de retropropagación y las técnicas de *Deep Learning*. Algunos de los posibles futuros desarrollos que se plantean como continuación de este trabajo, son los siguientes:

- **Implementación de nuevas actuaciones de los astrocitos.** La actuación de los astrocitos implementada en las RNGA y las DANAN se centra en la comunicación con las neuronas, modificando los pesos de las conexiones o la salida de las neuronas. El objetivo de implementar una metodología compatible con el algoritmo de retropropagación y las técnicas de *Deep Learning* es poder desarrollar en el futuro actuaciones más complejas de los astrocitos. Por una parte, se pueden desarrollar astrocitos más complejos que actúen sobre un mayor número de neuronas, al igual que ocurre en el cerebro donde se ha observado que un único astrocito puede llegar a actuar sobre más de un millón de sinapsis simultáneamente. También se pueden implementar una actuación diferente de los astrocitos en función de la capa en la que actúen, por ejemplo, modificando más las neuronas que se encuentran cerca de la capa de entrada y menos las que están próximas a la capa de salida o viceversa. Por otra parte, al existir un mayor número de astrocitos se puede desarrollar una comunicación entre los astrocitos que simule las ondas de Ca^{+2} observadas en el cerebro. De esta manera la actuación de los astrocitos no será dependiente de la actividad de una única neurona ya que la comunicación con otros astrocitos permitirá que las modificaciones integren información de otras sinapsis.
- **Desarrollo de una metodología compatible con otras arquitecturas.** Los trabajos llevados a cabo hasta el momento se centraron exclusivamente en redes multicapa, conectadas hacia delante, sin conexiones laterales, totalmente conectadas. Sería interesante evaluar el efecto de los astrocitos con otras arquitecturas más complejas. Por ejemplo, redes de convolución cuya arquitectura guarda similitud con el sistema visual y están especialmente diseñadas para la resolución de problemas de imagen. Otra opción sería la combinación con las redes

recurrentes como las *Long-Short Term Memory* que cuentan con una memoria interna lo que las hace adecuadas para trabajar con series temporales.

- **Estudio de la actuación de los astrocitos en redes preentrenadas.** La metodología desarrollada en esta tesis para la implementación de las DANAN permite diseñar una actuación de los astrocitos compatible con las redes preentrenadas, ya que pueden actuar sobre la configuración de pesos obtenidos en la fase de entrenamiento con el algoritmo de retropropagación del error. Algunas arquitecturas en las que sería interesante probar la actuación de los astrocitos son la GoogleLeNet, AlexNet, VGG18, InceptionV3, ya que han obtenido los mejores resultados en competiciones de reconocimiento de imagen. Por tanto, esta sería una buena manera de evaluar la capacidad de los astrocitos de mejorar los resultados de modelos sumamente complejos, con millones de neuronas, en una competición altamente competitiva.

9. Bibliografía

- Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., ... Devin, M. (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv Preprint arXiv:1603.04467*.
- Alvarellos-González, A., Pazos, A., & Porto-Pazos, A. B. (2012a). Computational models of neuron-astrocyte interactions lead to improved efficacy in the performance of neural networks. *Computational and Mathematical Methods in Medicine, 2012*.
- Alvarellos-González, A., Pazos, A., & Porto-Pazos, A. B. (2012b). Computational models of neuron-astrocyte interactions lead to improved efficacy in the performance of neural networks. *Computational and Mathematical Methods in Medicine, 2012*, 476324. <https://doi.org/10.1155/2012/476324>
- Araque, A., & Navarrete, M. (2010). Glial cells in neuronal network function. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences, 365*(1551), 2375–2381. <https://doi.org/10.1098/rstb.2009.0313>
- Araque, A., Parpura, V., Sanzgiri, R., & Haydon, P. (1999). Tripartite synapses: glia, the unacknowledged partner. *Trends in Neurosciences, 22*(5), 208–215. Retrieved from <http://www.sciencedirect.com/science/article/pii/S0166223698013496>
- Araque, A., Parpura, V., Sanzgiri, R. P., & Haydon, P. G. (1998). Glutamate-dependent astrocyte modulation of synaptic transmission between cultured hippocampal neurons. *The European Journal of Neuroscience, 10*(6), 2129–2142. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/9753099>
- Araque, A., Sanzgiri, R. P., Parpura, V., & Haydon, P. G. (1998). Calcium elevation in astrocytes causes an NMDA receptor-dependent increase in the frequency of miniature synaptic currents in cultured hippocampal neurons. *Journal of Neuroscience, 18*(17), 6822–6829.
- Azevedo, F. A. C., Carvalho, L. R. B., Grinberg, L. T., Farfel, J. M., Ferretti, R. E. L., Leite, R. E. P., ... Herculano-Houzel, S. (2009). Equal numbers of neuronal and nonneuronal cells make the human brain an isometrically scaled-up primate brain. *Journal of Comparative Neurology, 513*(5), 532–541.
- Azevedo, F. A. C., Carvalho, L. R. B., Grinberg, L. T., Farfel, J. M., Ferretti, R. E. L., Leite, R. E. P., ... Herculano-Houzel, S. (2009). Equal numbers of neuronal and nonneuronal cells make the human brain an isometrically scaled-up primate brain. *The Journal of Comparative Neurology, 513*(5), 532–541. <https://doi.org/10.1002/cne.21974>
- Ba, J., Mnih, V., & Kavukcuoglu, K. (2014). Multiple object recognition with visual

- attention. *arXiv Preprint arXiv:1412.7755*.
- Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv Preprint arXiv:1409.0473*.
- Baldi, P., Sadowski, P., & Whiteson, D. (2014). Searching for exotic particles in high-energy physics with deep learning. *Nature Communications*, 5.
- Battaglia, P., Pascanu, R., Lai, M., & Rezende, D. J. (2016). Interaction networks for learning about objects, relations and physics. In *Advances in Neural Information Processing Systems* (pp. 4502–4510).
- Battaglia, P. W., Hamrick, J. B., & Tenenbaum, J. B. (2013). Simulation as an engine of physical scene understanding. *Proceedings of the National Academy of Sciences*, 110(45), 18327–18332.
- Bear, M. F., Connors, B. W., & Paradiso, M. A. (2007). *Neuroscience* (Vol. 2). Lippincott Williams & Wilkins.
- Benjamin, B. V., Gao, P., McQuinn, E., Choudhary, S., Chandrasekaran, A. R., Bussat, J. M., ... Boahen, K. (2014). Neurogrid: A mixed-analog-digital multichip system for large-scale neural simulations. *Proceedings of the IEEE*, 102(5), 699–716. <https://doi.org/10.1109/JPROC.2014.2313565>
- Bergstra, J., Breuleux, O., Bastien, F., Lamblin, P., Pascanu, R., Desjardins, G., ... Bengio, Y. (2010). Theano: A CPU and GPU math compiler in Python. In *Proc. 9th Python in Science Conf* (pp. 1–7).
- Bottou, L. (2010). Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010* (pp. 177–186). Springer.
- Bourque, M., & Robitaille, R. (1998). Endogenous peptidergic modulation of perisynaptic Schwann cells at the frog neuromuscular junction. *The Journal of Physiology*, 512(1), 197–209.
- Brooks, R., Hassabis, D., Bray, D., & Shashua, A. (2012). Is the brain a good model for machine intelligence? *Nature*, 482(7386), 462.
- Brown, N., & Sandholm, T. (2017). Superhuman AI for heads-up no-limit poker: Libratus beats top professionals. *Science*, eaao1733.
- Browne, C. B., Powley, E., Whitehouse, D., Lucas, S. M., Cowling, P. I., Rohlfshagen, P., ... Colton, S. (2012). A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in Games*, 4(1), 1–43.
- Burkitt, A. N. (2006). A review of the integrate-and-fire neuron model: I. Homogeneous synaptic input. *Biological Cybernetics*, 95(1), 1–19.

- Bushong, E. A., Martone, M. E., Jones, Y. Z., & Ellisman, M. H. (2002). Protoplasmic astrocytes in CA1 stratum radiatum occupy separate anatomical domains. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience*, 22(1), 183–192. <https://doi.org/22/1/183> [pii]
- Cameron, A. C., & Windmeijer, F. A. G. (1997). An R-squared measure of goodness of fit for some common nonlinear regression models. *Journal of Econometrics*, 77(2), 329–342.
- Canziani, A., Paszke, A., & Culurciello, E. (2016). An analysis of deep neural network models for practical applications. *arXiv Preprint arXiv:1605.07678*.
- Centro de Supercomputación de Galicia (CESGA). (2017). Retrieved from <http://www.cesga.es/>
- Charles, A. C., Merrill, J. E., Dirksen, E. R., & Sanderson, M. J. (1991). Intercellular signaling in glial cells: calcium waves and oscillations in response to mechanical stimulation and glutamate. *Neuron*, 6(6), 983–992.
- Chollet, F. (2015). Keras: Deep learning library for theano and tensorflow. URL: <https://keras.io/k>.
- Codella, N., Cai, J., Abedini, M., Garnavi, R., Halpern, A., & Smith, J. (2015). Deep Learning, Sparse Coding, and SVM for Melanoma Recognition in Dermoscopy Images. In L. Zhou, L. Wang, Q. Wang, & Y. Shi (Eds.), *Machine Learning in Medical Imaging SE - 15* (Vol. 9352, pp. 118–126). Springer International Publishing. https://doi.org/10.1007/978-3-319-24888-2_15
- Cornell-Bell, A. H., Finkbeiner, S. M., Cooper, M. S., & Smith, S. J. (1990). Glutamate induces calcium waves in cultured astrocytes: long-range glial signaling. *Science*, 247(4941), 470.
- Cornell-Bell, A., Thomas, P. G., & Smith, S. J. (1990). The excitatory neurotransmitter glutamate causes filopodia formation in cultured hippocampal astrocytes. *Glia*, 3(5), 322–334. <https://doi.org/10.1002/glia.440030503>
- Cunningham, C. L., Martínez-Cerdeño, V., & Noctor, S. C. (2013). Microglia regulate the number of neural precursor cells in the developing cerebral cortex. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience*, 33(10), 4216–4233. Retrieved from <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3711552&tool=pmcentrez&rendertype=abstract>
- Dahl, G. E., Jaitly, N., & Salakhutdinov, R. (2014). Multi-task Neural Networks for QSAR Predictions. *Machine Learning; Learning; Neural and Evolutionary Computing*.

Retrieved from <http://arxiv.org/abs/1406.1231>

- De Pittà, M., Volman, V., Berry, H., Parpura, V., Volterra, A., & Ben-Jacob, E. (2012). Computational quest for understanding the role of astrocyte signaling in synaptic transmission and plasticity. *Frontiers in Computational Neuroscience*, 6.
- Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10(7), 1895–1923.
- Eliasmith, C., Stewart, T. C., Choo, X., Bekolay, T., DeWolf, T., Tang, Y., ... Rasmussen, D. (2012). A large-scale model of the functioning brain. *Science (New York, N.Y.)*, 338(6111), 1202–1205. <https://doi.org/10.1126/science.1225266>
- Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639), 115–118.
- Farabet, C., Couprie, C., Najman, L., & LeCun, Y. (2013). Learning hierarchical features for scene labeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8), 1915–1929.
- Fiacco, T. A., & McCarthy, K. D. (2004). Intracellular astrocyte calcium waves in situ increase the frequency of spontaneous AMPA receptor currents in CA1 pyramidal neurons. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience*, 24(3), 722–732.
- Fields, R. D. (2009). *The Other Brain: From Dementia to Schizophrenia, How New Discoveries about the Brain Are Revolutionizing Medicine and Science*. Simon and Schuster. Retrieved from <https://books.google.com/books?id=2nmHpXPmV80C&pgis=1>
- French, R. M. (1999). Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3(4), 128–135.
- Fukushima, K., & Miyake, S. (1982). Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. In *Competition and cooperation in neural nets* (pp. 267–285). Springer.
- Gallistel, C. R., & King, A. P. (2009). Memory and the Computational Brain: Why Cognitive Science Will Transform Neuroscience.
- Gatys, L. A., Ecker, A. S., & Bethge, M. (2016). Image style transfer using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2414–2423).
- Genoud, C., Quairiaux, C., Steiner, P., Hirling, H., Welker, E., & Knott, G. W. (2006). Plasticity of astrocytic coverage and glutamate transporter expression in adult mouse

- cortex. *PLoS Biology*, 4(11), 2057–2064.
- Gerstner, W., & Kistler, W. M. (2002). Mathematical formulations of Hebbian learning. *Biological Cybernetics*, 87(5–6), 404–415.
- Ghaderi, V. S., Allam, S. L., Ambert, N., Bouteiller, J. M. C., Choma, J., & Berger, T. W. (2011). Modeling neuron-glia interactions: from parametric model to neuromorphic hardware. In *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE* (pp. 3581–3584). IEEE.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press. Retrieved from <http://www.deeplearningbook.org>
- Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... Bengio, Y. (2014). Generative Adversarial Networks. *arXiv Preprint arXiv:1406.2661*.
- Goodfellow, I. J., Warde-Farley, D., Lamblin, P., Dumoulin, V., Mirza, M., Pascanu, R., ... Bengio, Y. (2013). Pylearn2: a machine learning research library. *arXiv Preprint arXiv:1308.4214*.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems* (pp. 2672–2680).
- Gorman, R. P., & Sejnowski, T. J. (1988). Analysis of hidden units in a layered network trained to classify sonar targets. *Neural Networks*, 1(1), 75–89.
- Graves, A., Wayne, G., Reynolds, M., Harley, T., Danihelka, I., Grabska-Barwińska, A., ... Agapiou, J. (2016). Hybrid computing using a neural network with dynamic external memory. *Nature*, 538(7626), 471–476.
- Harkin, J., Morgan, F., McDaid, L., Hall, S., McGinley, B., & Cawley, S. (2009). A reconfigurable and biologically inspired paradigm for computation using network-on-chip and spiking neural networks. *International Journal of Reconfigurable Computing*, 2009, 2.
- Hassabis, D., Kumaran, D., Summerfield, C., & Botvinick, M. (2017). Neuroscience-inspired artificial intelligence. *Neuron*, 95(2), 245–258.
- Hayati, M., Nouri, M., Haghiri, S., & Abbott, D. (2016). A digital realization of astrocyte and neural glial interactions. *IEEE Transactions on Biomedical Circuits and Systems*, 10(2), 518–529.
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).
- Hebb, D. (1949). *The Organization of Behaviour: A Neuropsychological Theory*.

- Herz, A. V. M., Gollisch, T., Machens, C. K., & Jaeger, D. (2006). Modeling single-neuron dynamics and computations: a balance of detail and abstraction. *Science*, *314*(5796), 80–85.
- Higgins, I., Matthey, L., Glorot, X., Pal, A., Uria, B., Blundell, C., ... Lerchner, A. (2016). Early visual concept learning with unsupervised deep learning. *arXiv Preprint arXiv:1606.05579*.
- Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A., Jaitly, N., ... Sainath, T. N. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, *29*(6), 82–97.
- Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, *313*(5786), 504–507.
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.
- Hodgkin, A. L., & Huxley, A. F. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology*, *117*(4), 500–544. <https://doi.org/10.1113/jphysiol.1952.sp004764>
- Holyoak, K. J., & Thagard, P. (1997). The analogical mind. *American Psychologist*, *52*(1), 35.
- Hong, S., Oh, J., Lee, H., & Han, B. (2016). Learning transferrable knowledge for semantic segmentation with deep convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 3204–3212).
- Ikuta, C., Uwate, Y., & Nishio, Y. (2010). Multi-Layer Perceptron Having Neuro-Glia Network, 189–192.
- Ikuta, C., Uwate, Y., & Nishio, Y. (2011). Performance and features of multi-layer perceptron with impulse glial network. In *Neural Networks (IJCNN), The 2011 International Joint Conference on* (pp. 2536–2541). IEEE.
- Ikuta, C., Uwate, Y., & Nishio, Y. (2012). Multi-layer perceptron with positive and negative pulse glial chain for solving two-spirals problem. In *Neural Networks (IJCNN), The 2012 International Joint Conference on* (pp. 1–6). IEEE.
- Ikuta, C., Uwate, Y., & Nishio, Y. (2014). Multi-Layer Perceptron with Local Glia Connection.
- Ikuta, C., Uwate, Y., Nishio, Y., & Yang, G. (2016). Multi-layer perceptron with pulse glial chain. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, *99*(3), 742–755.
- Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by

- reducing internal covariate shift. In *International Conference on Machine Learning* (pp. 448–456).
- Irizarry-Valle, Y., & Parker, A. C. (2014). Astrocyte on neuronal phase synchrony in CMOS. *Circuits and Systems (ISCAS), 2014 IEEE International Symposium on*. <https://doi.org/10.1109/ISCAS.2014.6865115>
- Irizarry-Valle, Y., & Parker, A. C. (2015). An astrocyte neuromorphic circuit that influences neuronal phase synchrony. *IEEE Transactions on Biomedical Circuits and Systems*, 9(2), 175–187. <https://doi.org/10.1109/TBCAS.2015.2417580>
- Irizarry-Valle, Y., Parker, A. C., & Joshi, J. (2013). A CMOS neuromorphic approach to emulate neuro-astrocyte interactions. In *Proceedings of the International Joint Conference on Neural Networks*. <https://doi.org/10.1109/IJCNN.2013.6707076>
- Izhikevich, E. M. (2003). Simple model of spiking neurons. *IEEE Transactions on Neural Networks*, 14(6), 1569–1572.
- Joshi, J., Parker, A. C., & Tseng, K.-C. (2011). An in-silico glial microdomain to invoke excitability in cortical neural networks. In *Circuits and Systems (ISCAS), 2011 IEEE International Symposium on* (pp. 681–684). IEEE.
- Kang, J., Jiang, L., Goldman, S. A., & Nedergaard, M. (1998). Astrocyte-mediated potentiation of inhibitory synaptic transmission. *Nature Neuroscience*, 1(8), 683–692. <https://doi.org/10.1038/3684>
- Kasthuri, N., Hayworth, K. J., Berger, D. R., Schalek, R. L., Conchello, J. A., Knowles-Barley, S., ... Jones, T. R. (2015). Saturated reconstruction of a volume of neocortex. *Cell*, 162(3), 648–661.
- Khan, M. M., Lester, D. R., Plana, L. A., Rast, A., Jin, X., Painkras, E., & Furber, S. B. (2008). SpiNNaker: mapping neural networks onto a massively-parallel chip multiprocessor. In *Neural Networks, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence)*. *IEEE International Joint Conference on* (pp. 2849–2856). Ieee.
- Korytkowski, M. (2016). A Novel Convolutional Neural Network with Glial Cells. In *International Conference on Artificial Intelligence and Soft Computing* (pp. 670–679). Springer.
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. In F. Pereira, C. J. C. Burges, L. Bottou, & K. Q. Weinberger (Eds.), *Advances in Neural Information Processing Systems 25* (pp. 1097–1105). Curran Associates, Inc. Retrieved from <http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf>
- Lake, B. M., Salakhutdinov, R., & Tenenbaum, J. B. (2015). Human-level concept learning

- through probabilistic program induction. *Science*, 350(6266), 1332–1338.
- Le Roux, P. D., & Reh, T. A. (1995). Astroglia demonstrate regional differences in their ability to maintain primary dendritic outgrowth from mouse cortical neurons in vitro. *Developmental Neurobiology*, 27(1), 97–112.
- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2324.
- Lederberg, J., Sutherland, G. L., Buchanan, B. G., & Feigenbaum, E. A. (1970). A heuristic program for solving a scientific inference problem: summary of motivation and implementation. In *Theoretical Approaches to Non-Numerical Problem Solving* (pp. 401–409). Springer.
- Lichman, M. (2013). University of California, Irvine Machine Learning Repository. Retrieved from <http://archive.ics.uci.edu/ml>
- Maass, W., & Bishop, C. M. (1998). Spiking Neurons. In *NC* (pp. 16–20).
- Mangasarian, O. L., Setiono, R., & Wolberg, W. H. (1990). Pattern recognition via linear programming: Theory and application to medical diagnosis. *Large-Scale Numerical Optimization*, 22–31.
- Markram, H. (2012). The human brain project. *Scientific American*, 306(6), 50–55.
- Marr, D., & Poggio, T. (1976). From understanding computation to understanding neural circuitry.
- McClelland, J. L., McNaughton, B. L., & O'reilly, R. C. (1995). Why there are complementary learning systems in the hippocampus and neocortex: insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, 102(3), 419.
- McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4), 115–133. <https://doi.org/10.1007/BF02478259>
- Merolla, P. a, Arthur, J. V, Alvarez-Icaza, R., Cassidy, A. S., Sawada, J., Akopyan, F., ... Modha, D. S. (2014). A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science (New York, N.Y.)*, 345(6197), 668–673. <https://doi.org/10.1126/science.1254642>
- Michalski, R. S., Carbonell, J. G., & Mitchell, T. M. (2013). *Machine learning: An artificial intelligence approach*. Springer Science & Business Media.
- Minsky, M., & Paper, S. (1969). *Perceptrons*. MIT Press, Cambridge.
- Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., &

- Riedmiller, M. (2013). Playing atari with deep reinforcement learning. *arXiv Preprint arXiv:1312.5602*.
- Moore, D. S., & McCabe, G. P. (1989). *Introduction to the Practice of Statistics*. WH Freeman/Times Books/Henry Holt & Co.
- Moravčík, M., Schmid, M., Burch, N., Lisý, V., Morrill, D., Bard, N., ... Bowling, M. (2017). DeepStack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, *356*(6337), 508–513.
- Nadkarni, S., & Jung, P. (2007). Modeling synaptic transmission of the tripartite synapse. *Physical Biology*, *4*(1), 1–9. <https://doi.org/10.1088/1478-3975/4/1/001>
- Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)* (pp. 807–814).
- Nedergaard, M. (1994). Direct signaling from astrocytes to neurons in cultures of mammalian brain cells. *Science (New York, N.Y.)*, *263*(5154), 1768–1771. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/8134839>
- Newman, E. A., & Zahs, K. R. (1998). Modulation of neuronal activity by glial cells in the retina. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience*, *18*(11), 4022–4028.
- Nishiyama, A., Komitova, M., Suzuki, R., & Zhu, X. (2009). Polydendrocytes (NG2 cells): multifunctional cells with lineage plasticity. *Nature Reviews Neuroscience*, *10*(1), 9–22.
- Oberheim, N. A., Goldman, S. A., & Nedergaard, M. (2012a). Heterogeneity of astrocytic form and function. *Methods in Molecular Biology*, *814*, 23–45.
- Oberheim, N. A., Goldman, S. A., & Nedergaard, M. (2012b). Heterogeneity of astrocytic form and function. *Methods in Molecular Biology (Clifton, N.J.)*, *814*, 23–45. https://doi.org/10.1007/978-1-61779-452-0_3
- Oberheim, N. A., Takano, T., Han, X., He, W., Lin, J. H. C., Wang, F., ... Nedergaard, M. (2009). Uniquely hominid features of adult human astrocytes. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience*, *29*(10), 3276–3287.
- Ogata, K., & Kosaka, T. (2002). Structural and quantitative analysis of astrocytes in the mouse hippocampus. *Neuroscience*, *113*(1), 221–233.
- Oord, A. van den, Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., ... Kavukcuoglu, K. (2016). Wavenet: A generative model for raw audio. *arXiv Preprint arXiv:1609.03499*.
- Oschmann, F., Berry, H., Obermayer, K., & Lenk, K. (2017). From in silico astrocyte cell

- models to neuron-astrocyte network models: A review. *Brain Research Bulletin*.
- Øyehaug, L., Østby, I., Lloyd, C. M., Omholt, S. W., & Einevoll, G. T. (2012). Dependence of spontaneous neuronal firing and depolarisation block on astroglial membrane transport mechanisms. *Journal of Computational Neuroscience*, *32*(1), 147–165.
- Parpura, V., Basarsky, T. A., Liu, F., Jefitnija, K., Jefitnija, S., & Haydon, P. G. (1994). Glutamate-mediated astrocyte-neuron signalling. *Nature*, *369*(6483), 744–747. <https://doi.org/10.1038/369744a0>
- Pastur-Romay, L. A., Cedrón, F., Pazos, A., & Porto-Pazos, A. B. (2016). Deep Artificial Neural Networks and Neuromorphic Chips for Big Data Analysis: Pharmaceutical and Bioinformatics Applications. *Int. J. Mol. Sci*, *17*, 1313.
- Pastur-Romay, L. A., Porto-Pazos, B. A., Cedrón, F., & Pazos, A. (2017). Parallel computing for brain simulation. *Current Topics in Medicinal Chemistry*, *17*(14), 1646–1668.
- Perea, G., & Araque, A. (2002). Communication between astrocytes and neurons: a complex language. *Journal of Physiology, Paris*, *96*(3–4), 199–207. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/12445897>
- Perea, G., & Araque, A. (2007). Astrocytes potentiate transmitter release at single hippocampal synapses. *Science (New York, N.Y.)*, *317*(5841), 1083–1086. <https://doi.org/10.1126/science.1144640>
- Perea, G., & Araque, A. (2010). GLIA modulates synaptic transmission. *Brain Research Reviews*, *63*(1–2), 93–102. <https://doi.org/10.1016/j.brainresrev.2009.10.005>
- Pfriefer, F. W. (2002). Role of glia in synapse development. *Current Opinion in Neurobiology*.
- Porto-Pazos, A. B., Alvarellos-González, A., & Montañes Pazos, F. (2009). Artificial NeuroGlial Networks. In *Encyclopedia of Artificial Intelligence*. <https://doi.org/10.4018/978-1-59904-849-9.ch026>
- Porto-Pazos, A. B., Araque, A., Rabuñal, J., Dorado, J., & Pazos, A. (2007). A new hybrid evolutionary mechanism based on unsupervised learning for Connectionist Systems. *Neurocomputing*, *70*(16–18), 2799–2808. <https://doi.org/10.1016/j.neucom.2006.06.010>
- Porto-Pazos, A. B., Pazos, A., & Araque, A. (2005). Artificial neural networks based on brain circuits behaviour and genetic algorithms. In *Computational Intelligence and Bioinspired Systems* (pp. 99–106). Springer.
- Porto-Pazos, A. B., Veiguela, N., Mesejo, P., Navarrete, M., Alvarellos, A., Ibáñez, O., ... Araque, A. (2011a). Artificial astrocytes improve neural network performance. *PLoS One*, *6*(4).

- Porto-Pazos, A. B., Veiguela, N., Mesejo, P., Navarrete, M., Alvarellós, A., Ibáñez, O., ... Araque, A. (2011b). Artificial astrocytes improve neural network performance. *PLoS One*, 6(4), e19109. <https://doi.org/10.1371/journal.pone.0019109>
- Porto, A., Araque, A., Rabuñal, J., Dorado, J., & Pazos, A. (2007). A new hybrid evolutionary mechanism based on unsupervised learning for Connectionist Systems. *Neurocomputing*, 70(16–18), 2799–2808. <https://doi.org/10.1016/j.neucom.2006.06.010>
- Porto, A. B. (2004). *Modelos Computacionales para optimizar el Aprendizaje y el Procesamiento de la Información en Sistemas Adaptativos: Redes Neurogliales Artificiales (RR.NG.AA.)*. Tesis doctoral. Universidade da Coruña.
- Porto, A., Pazos, A., & Araque, A. (2005). Artificial neural networks based on brain circuits behaviour and genetic algorithms. In *Computational Intelligence and Bioinspired Systems* (pp. 99–106). Springer.
- Porto Pazos, A. B. (2008). *Advancing artificial intelligence through biological process applications*. IGI Global.
- Postnov, D. E., Koreshkov, R. N., Brazhe, N. A., Brazhe, A. R., & Sosnovtseva, O. V. (2009). Dynamical patterns of calcium signaling in a functional model of neuron–astrocyte networks. *Journal of Biological Physics*, 35(4), 425–445.
- Postnov, D. E., Ryazanova, L. S., Brazhe, N. A., Brazhe, A. R., Maximov, G. V., Mosekilde, E., & Sosnovtseva, O. V. (2008). Giant glial cell: New insight through mechanism-based modeling. *Journal of Biological Physics*, 34(3–4), 441–457.
- Prechelt, L. (1994). Proben1: A set of neural network benchmark problems and benchmarking rules.
- Raff, M. C., Barres, B. A., Burne, J. F., Coles, H. S., Ishizaki, Y., & Jacobson, M. D. (1993). Programmed cell death and the control of cell survival: Lessons from the nervous system. *Science*, 262(5134), 695–700. Retrieved from <http://www.scopus.com/inward/record.url?eid=2-s2.0-0027373764&partnerID=40&md5=654197a23314d3a6135982549e5e15db>
- Rall, W. (1989). Cable theory for dendritic neurons. In *Methods in neuronal modeling* (pp. 9–92). MIT press.
- Ramon y Cajal, S. (1904). *Textura del Sistema Nervioso del Hombre y de los Vertebrados* (Vol. 2). Madrid Nicolas Moya.
- Ramón y Cajal, S. (1952). *Histologie du système nerveux de l'homme & des vertèbres*. Consejo Superior de Investigaciones Científicas. Instituto Ramón y Cajal.
- Ranjbar, M., & Amiri, M. (2015). On the role of astrocyte analog circuit in neural frequency

- adaptation. *Neural Computing and Applications*, 1–13. <https://doi.org/10.1007/s00521-015-2112-8>
- Rezende, D., Danihelka, I., Gregor, K., & Wierstra, D. (2016). One-shot generalization in deep generative models. In *International Conference on Machine Learning* (pp. 1521–1529).
- Ripley, B., Hornik, K., Gebhardt, A., & Firth, D. (2012). Functions and datasets to support Venables and Ripley, “Modern Applied Statistics with S”(2002): package “MASS.” *Accessed through Http://cran. R-Project. Org.*
- Robitaille, R. (1998). Modulation of synaptic efficacy and synaptic depression by glial cells at the frog neuromuscular junction. *Neuron*, 21(4), 847–855. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/9808470>
- Romero-Porta, F., Mesejo, P., Ibáñez, O., & Porto-Pazos, A. B. (2004). Optimización mediante Computación Evolutiva de la interacción Neurona-Astrocyto en Redes Neurogliales Artificiales.
- Rosenblatt, F. (1957). *The perceptron, a perceiving and recognizing automaton Project Para.* Cornell Aeronautical Laboratory.
- Rowland, D. C., Roudi, Y., Moser, M.-B., & Moser, E. I. (2016). Ten years of grid cells. *Annual Review of Neuroscience*, 39, 19–40.
- Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088), 533–536.
- Santoro, A., Bartunov, S., Botvinick, M., Wierstra, D., & Lillicrap, T. (2016). One-shot learning with memory-augmented neural networks. *arXiv Preprint arXiv:1605.06065*.
- Schaul, T., Quan, J., Antonoglou, I., & Silver, D. (2015). Prioritized experience replay. *arXiv Preprint arXiv:1511.05952*.
- Schemmel, J., Brüderle, D., Grübl, A., Hock, M., Meier, K., & Millner, S. (2010). A wafer-scale neuromorphic hardware system for large-scale neural modeling. In *ISCAS 2010 - 2010 IEEE International Symposium on Circuits and Systems: Nano-Bio Circuit Fabrics and Systems* (pp. 1947–1950). <https://doi.org/10.1109/ISCAS.2010.5536970>
- Schemmel, J., Grübl, A., Hartmann, S., Kononov, A., Mayr, C., Meier, K., ... Scholze, S. (2012). Live demonstration: A scaled-down version of the brainscales wafer-scale neuromorphic system. In *Circuits and systems (ISCAS), 2012 IEEE international symposium on* (p. 702). IEEE.
- Schlimmer, J. (1981). Mushroom records drawn from The Audubon Society field guide to north American mushrooms. *GH Lincoff (Pres), New York*.
- Serrano, A., Haddjeri, N., Lacaille, J.-C., & Robitaille, R. (2006). GABAergic network

- activation of glial cells underlies hippocampal heterosynaptic depression. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience*, 26(20), 5370–5382. <https://doi.org/10.1523/JNEUROSCI.5255-05.2006>
- Sherwood, C. C., Stimpson, C. D., Raghanti, M. A., Wildman, D. E., Uddin, M., Grossman, L. I., ... Hof, P. R. (2006). Evolution of increased glia-neuron ratios in the human frontal cortex. *Proceedings of the National Academy of Sciences of the United States of America*, 103(37), 13606–13611.
- Shu, T., & Richards, L. J. (2001). Cortical axon guidance by the glial wedge during the development of the corpus callosum. *Journal of Neuroscience*, 21(8), 2749–2758.
- Sigillito, V. G., Wing, S. P., Hutton, L. V., & Baker, K. B. (1989). Classification of radar returns from the ionosphere using neural networks. *Johns Hopkins APL Technical Digest*, 10(3), 262–266.
- Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., ... Lanctot, M. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587), 484–489.
- Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., ... Bolton, A. (2017). Mastering the game of go without human knowledge. *Nature*, 550(7676), 354.
- Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv Preprint arXiv:1409.1556*.
- Singer, A. C., & Frank, L. M. (2009). Rewarded outcomes enhance reactivation of experience in the hippocampus. *Neuron*, 64(6), 910–921.
- Smith, J. W., Everhart, J. E., Dickson, W. C., Knowler, W. C., & Johannes, R. S. (1988). Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. In *Proceedings of the Annual Symposium on Computer Application in Medical Care* (p. 261). American Medical Informatics Association.
- Sotero, R. C., & Martínez-Cancino, R. (2010). Dynamical mean field model of a neural-glia mass. *Neural Computation*, 22(4), 969–997.
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15, 1929–1958.
- Steel, R. G. D., & Torrie, J. (1960). *Principales And Pricedures Of Statistics*. Mcgraw-Hill Book Company, Inc.; New York; Toronto; London.
- Stevens, B. (2008). Neuron-astrocyte signaling in the development and plasticity of neural

- circuits. *NeuroSignals*.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1–9).
- Theodosis, D. T., Poulain, D. A., & Oliet, S. H. R. (2008). Activity-dependent structural and functional plasticity of astrocyte-neuron interactions. *Physiological Reviews*, *88*(3), 983–1008. <https://doi.org/10.1152/physrev.00036.2007>
- Tóth, G., Jones, A., & Montanarella, L. (2013). *LUCAS Topsoil Survey: methodology, data and results*. Publications Office.
- Turing, A. M. (1948). Intelligent Machinery. Report for National Physical Laboratory 1948. In B. Meltzer & D. Michie (Eds.), *Machine Intelligence* (pp. 3–23). Edinburgh, U.K 1969.
- Turing, A. M. (1950). Computing Machinery and Intelligence. *Mind*, *59*(236), 433–460. Retrieved from <http://www.loebner.net/Prizef/TuringArticle.html>
- Tzingounis, A. V, & Wadiche, J. I. (2007). Glutamate transporters: confining runaway excitation by shaping synaptic transmission. *Nat Rev Neurosci*, *8*(12), 935–947. Retrieved from <http://dx.doi.org/10.1038/nrn2274>
- Wade, J. J., McDaid, L. J., Harkin, J., Crunelli, V., & Kelso, J. A. (2011). Bidirectional coupling between astrocytes and neurons mediates learning and dynamic coordination in the brain: a multiple modeling approach. *PLoS One*, *6*(12), e29445.
- Wang, J. X., Kurth-Nelson, Z., Tirumala, D., Soyer, H., Leibo, J. Z., Munos, R., ... Botvinick, M. (2016). Learning to reinforcement learn. *arXiv Preprint arXiv:1611.05763*.
- Widrow, B., & Hoff, M. E. (1960). Adaptive switching circuits. In *IRE WESCON convention record* (Vol. 4, pp. 96–104). New York.
- Wiener, N. (1975). *Dios y golem, SA*. Siglo veintiuno editores.
- Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics Bulletin*, *1*(6), 80–83.
- Wu, Y., Schuster, M., Chen, Z., Le, Q. V, Norouzi, M., Macherey, W., ... Macherey, K. (2016). Google’s neural machine translation system: Bridging the gap between human and machine translation. *arXiv Preprint arXiv:1609.08144*.
- Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., ... Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning* (pp. 2048–2057).
- Yao, X., & Liu, Y. (1997). A new evolutionary system for evolving artificial neural networks. *IEEE Transactions on Neural Networks*, *8*(3), 694–713.

- Zahavy, T., Ben-Zrihem, N., & Mannor, S. (2016). Graying the black box: Understanding DQNs. In *International Conference on Machine Learning* (pp. 1899–1908).
- Zeiler, M. D. (2012). ADADELTA: an adaptive learning rate method. *arXiv Preprint arXiv:1212.5701*.
- Zeiler, M. D., & Fergus, R. (2014). Visualizing and understanding convolutional networks. In *European conference on computer vision* (pp. 818–833). Springer.