



**UNIVERSIDAD NACIONAL AUTÓNOMA  
DE MÉXICO**

---

---

**FACULTAD DE CIENCIAS**

**MODELADO DE LA PLASTICIDAD Y  
HOMEOSTASIS EN EL HIPOCAMPO DE UN RATÓN**

**T E S I S**

**QUE PARA OBTENER EL TÍTULO DE:**

**LICENCIADA EN MATEMÁTICAS**

**P R E S E N T A :**

**LIUBOVE ORLOV SAVKO**



**DIRECTOR DE TESIS:  
DR. MARCO TULIO ANGULO BALLESTEROS  
Cd. Universitaria, CDMX., 2018**



Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.



# Agradecimientos

Le agradezco el apoyo, dedicación y tiempo a mi tutor, Marco Tulio Angulo, en el desarrollo de mi tesis. Gracias por su paciencia y los conocimientos brindados.

También le agradezco a Leonid Fridman por haberme recomendado con mi tutor. Alessio Franci, te agradezco el apoyo y ayuda en proponer y analizar ideas interesantes para mi tesis, y a todos mis sinodales les agradezco tomarse el tiempo en revisarla y darme sus comentarios.

Le doy las gracias a mi familia inmensamente por siempre apoyarme con mis decisiones y guiarme o corregirme cuando pudiera haber tomado la decisión equivocada. Mamá, gracias por siempre mostrar interés en mi trabajo a pesar de no tener tanto conocimiento en la materia.

Muchas gracias a mis amigos por siempre apoyarme; en especial a Sam Tortolero, Tania Duque, Daniel Bustillos y a Sebastián Michel, a quien conocí en mi estancia en Querétaro.

Todos ustedes formaron parte de mi formación y del desarrollo de mi tesis. Les agradezco todo su apoyo y guianzas.



# Datos del Jurado

1. Datos del Alumno  
Orlov  
Savko  
Liubove  
liuba1156@ciencias.unam.mx  
+5216461417555  
Universidad Nacional Autónoma de México  
Facultad de Ciencias  
Matemáticas  
414003428
2. Datos del tutor  
Dr.  
Angulo  
Ballesteros  
Marco Tulio
3. Datos del sinodal 1  
Dr.  
Ramírez  
Santiago  
Guillermo
4. Datos del sinodal 2  
Dr.  
Franci  
Alessio
5. Datos del sinodal 3  
Dra.  
Aparicio  
Martínez  
Andrea
6. Datos del sinodal 4  
Dr.  
Velasco  
Hernández  
Jorge Xicoténcatl
7. Datos del trabajo escrito  
Modelado de la plasticidad y homeostasis en el hipocampo de un ratón  
número de páginas: 80  
2018



# Índice general

<b>1. Introducción</b>	<b>3</b>
1.1. Historia de las Neurociencias . . . . .	3
1.2. Neurociencias y matemáticas . . . . .	8
1.3. El experimento de Slomowitz et al. sobre el hipocampo del ratón. . . . .	8
1.4. Objetivos . . . . .	9
<b>2. Construcción del modelo</b>	<b>11</b>
2.1. Antecedentes . . . . .	11
2.1.1. Modelo de Hodgkin-Huxley . . . . .	12
2.1.2. Modelo Integrate-and-Fire . . . . .	14
2.2. Modelo para red neuronal . . . . .	17
2.3. Construcción de algoritmos de aprendizaje . . . . .	20
2.3.1. Modelo biológico inspirado en la ley de aprendizaje de Hebb . . . . .	22
2.3.2. Modelo artificial de aprendizaje: método del descenso más rápido . . . . .	29
2.4. Algoritmo utilizando la tasa de disparo local . . . . .	30
<b>3. Resultados</b>	<b>33</b>
3.1. Resultados para redes simples . . . . .	33
3.2. Resultados: Red con Aprendizaje de Hebb . . . . .	45
3.3. Resultados: Gradiente . . . . .	51
3.4. Resultados: Tasa de Disparo Local . . . . .	55
<b>4. Conclusiones</b>	<b>61</b>
<b>5. Apéndice I</b>	<b>63</b>

8

*ÍNDICE GENERAL*

**Bibliografía**

**79**

# Índice de figuras

1.1.	Estructura de los centros nerviosos de las aves . . . . .	4
1.2.	Cerebro humano . . . . .	5
2.1.	Diagrama del modelo Hodgkin-Huxley. Del lado izquierdo se muestra la membrana neuronal y la concentración predominante de iones del exterior y el interior de la membrana. Del lado derecho demuestra el hecho que el modelo de Hodgkin-Huxley hace referencia a un circuito eléctrico . . . . .	13
2.2.	Comportamiento de una neurona utilizando el modelo Integrate-and-Fire . . . . .	16
2.3.	Grafo Erdős-Renyi . . . . .	17
2.4.	Matriz de adyacencia . . . . .	17
2.5.	Potencial postsináptico . . . . .	18
2.6.	Gráfica del potencial de membrana de neuronas en una red . .	21
2.7.	El campo de direcciones de la función de la actividad independiente de la sinapsis $a_0$ definida en 2.4 . . . . .	24
2.8.	Ventana de aprendizaje . . . . .	26
2.9.	Potencial de membrana sin aprendizaje Hebbiano . . . . .	27
2.10.	Potencial de membrana con aprendizaje Hebbiano . . . . .	27
2.11.	Estado de los pesos de la red de neuronas al finalizar esas 10 unidades de tiempo. . . . .	28
2.12.	Tiempos y tasas de disparo de red sin aprendizaje de Hebb . .	29
2.13.	Tiempos y tasas de disparo de red con aprendizaje de Hebb .	29
3.1.	Dependencia de la tasa de disparo con respecto a la entrada de corriente . . . . .	34
3.2.	Dependencia de la tasa de disparo $\langle f_2 \rangle_T$ de $I_1$ con $w = 1$ y $I_2 = 2$ . . . . .	39

3.3. Dependencia de las tasas individual (azul) y poblacional (rojo) del peso . . . . .	40
3.4. Relación entre la tasa de disparo de neurona 2 con la tasa de disparo de neurona 1 y el peso de la conexión. . Aquí $\langle f_1 \rangle$ representa la tasa de disparo de la neurona 1, $\langle f_2 \rangle$ representa la tasa de disparo de la neurona 2, y el eje z representa el peso entre las dos neuronas . . . . .	41
3.5. Potencial de membrana de dos neuronas con el peso $w = 1$ . . .	42
3.6. Potencial de membrana de dos neuronas con el peso $w = 0.5$ . .	42
3.7. Potencial de membrana de dos neuronas con el peso $w = -0.5$ . .	42
3.8. Potencial de membrana de dos neuronas con el peso $w = -1$ . .	42
3.9. El potencial de membrana se mantiene en reposo cuando no hay ningún estímulo que perturbe la red . . . . .	44
3.10. El potencial de membrana de las neuronas de la red con corriente de 5 unidades inyectada a la primer neurona) . . . . .	44
3.11. Las gráficas nos muestran la necesidad de inducir corrientes externas, pues sin ella no habrían tasas de disparos . . . . .	44
3.12. Potencial de membrana en red sin reglas de aprendizaje con $I_1 = 5, I_2 = 5, I_3 = 3$ . . . . .	46
3.13. Potencial de membrana en red con aprendizaje de Hebb con $I_1 = 5, I_2 = 5, I_3 = 3$ . . . . .	46
3.14. Potencial de membrana en red sin reglas de aprendizaje con $I_1 = 4, I_2 = 4, I_4 = 3$ . . . . .	47
3.15. Potencial de membrana en red con aprendizaje de Hebb con $I_1 = 4, I_2 = 4, I_3 = 3$ . . . . .	47
3.16. Potencial de membrana en red sin reglas de aprendizaje con $I_1 = 3, I_2 = 3, I_4 = 5$ . . . . .	48
3.17. Potencial de membrana en red con aprendizaje de Hebb con $I_1 = 3, I_2 = 3, I_3 = 5$ . . . . .	48
3.18. Potencial de membrana en red sin reglas de aprendizaje con $I_1 = 5, I_2 = 5, I_4 = 6$ . . . . .	49
3.19. Potencial de membrana en red con aprendizaje de Hebb con $I_1 = 5, I_2 = 5, I_3 = 6$ . . . . .	49
3.20. Función de costo para grafo de dos neuronas . . . . .	51
3.21. Función de costo para grafo de tres neuronas . . . . .	52
3.22. Evolución del potencial de membrana después de aplicar el algoritmo del gradiente más empinado en red de dos nodos . .	53

3.23. Evolución del potencial de membrana 2 después de aplicar el algoritmo del gradiente más empinado en red de tres nodos . .	54
3.24. Resultado del algoritmo de la tasa de disparo local en una red de dos nodos . . . . .	56
3.25. Resultado del algoritmo de la tasa de disparo local en una red de diez nodos . . . . .	57
3.26. Fallo del algoritmo de la tasa de disparo local . . . . .	58
3.27. Gráfica de densidad del error para redes entre 5 y 20 nodos con probabilidad de interconexión de 0.1 a 0.2. El color representa el error de la tasa de disparo después de aplicar el algoritmo. No se pudieron calcular valores para redes de 20 neuronas con probabilidad de interconexión mayor a 0.1, por lo que no se pudo hacer interpolación, resultando en una esquina blanca. .	59
5.1. Imagen de neurona clásica con sus tres partes bien distinguibles: axón, soma y dendritas . . . . .	64
5.2. Sinapsis -lugar donde se unen dos neuronas . . . . .	65
5.3. Gráfica del disparo de una neurona . . . . .	66
5.4. La forma de un PSP . . . . .	68



# Resumen Ejecutivo

En un artículo muy reciente de Slomowitz et al. [3], los autores reportan que la tasa de disparo poblacional de las neuronas en el hipocampo de un ratón es regulada a un valor constante a pesar de perturbaciones que disminuyen la actividad sináptica (i.e., antagonistas de los receptores de  $GABA_B$ ). Es decir, una vez aplicada una perturbación, la tasa de disparo poblacional regresa a un valor muy cercano al que tenía antes de la perturbación. Interesantemente, las tasas de disparo individual no regresan a su valor original. Esto sugiere la presencia de un mecanismo que confiere propiedades de “homeostasis” (*conjunto de eventos de autorregulación, conducentes al mantenimiento de una relativa constancia en composición y propiedades del medio interno de un organismo*) a la tasa de disparo poblacional en el hipocampo del ratón. Los mecanismos que confieren esta homeostasis no han sido completamente explicados.

En esta tesis, proponemos como hipótesis que la “plasticidad” de la fuerza en las conexiones sinápticas entre neuronas, en conjunto con mecanismos de retroalimentación, proveen un mecanismo para explicar la homeostasis observada por Slomowitz et al. [3]. Estudiamos esta hipótesis construyendo un modelo matemático de una red de neuronas cuyos pesos de conexión pueden ser ajustados a través de retroalimentación (i.e., “aprendizaje”). Utilizamos el modelo clásico “*Integrate-and-Fire*” para modelar la dinámica de cada neurona. Propusimos dos tipos de algoritmos para ajustar los pesos de interconexión entre neuronas: uno biológico inspirado en el aprendizaje Hebbiano, y el otro inspirado en inteligencia artificial. Construimos un software que nos permitió estudiar por simulación la dinámica de una red con un número arbitrario de neuronas regidos por estos algoritmos de ajuste. Encontramos que ambos algoritmos proveen homeostasis con respecto a la tasa de disparo poblacional, pero que la “calidad” de esta homeostasis (i.e., qué tan cerca del valor original puede llegar) depende de las propiedades de la red

de interconexión entre neuronas. Estos resultados sugieren que la hipótesis propuesta es un mecanismo plausible para explicar la homeostasis observada experimentalmente, y que en dicho caso la red de interconexión entre neuronas del hipocampo debería tener propiedades específicas que favorecen la homeostasis. En conjunto, nuestro trabajo ilustra una vez más como las matemáticas podrían ser valiosas para alcanzar el fin último de decodificar el “enigma del código neuronal”.

Esta tesis está organizada como sigue. En el Capítulo 1 se presenta una breve introducción de las neurociencias, desde sus inicios hasta ahora, desde una perspectiva biológica y matemática. En este capítulo también se presenta y discute a detalle el artículo de Slomowitz et al. [3] y se plantea la metodología que se siguió para afirmar la hipótesis propuesta. En el Capítulo 2 explicamos los modelos matemáticos de neuronas aisladas, extendemos estos modelos en una red de neuronas interconectadas, y explicamos los algoritmos de ajuste de pesos propuestos en esta tesis. En el Capítulo 3 se exponen los resultados obtenidos con el programa de Wolfram Mathematica desarrollado. En el Capítulo 4 se dan las conclusiones de estos resultados. Algunos términos básicos utilizados con frecuencia en neurociencias pueden encontrarse en el apéndice.

# Capítulo 1

## Introducción, objetivos y organización de la tesis

### 1.1. Sobre las Neurociencias: su origen, desarrollo y estado actual.

El cerebro humano contiene más de 80 mil millones de neuronas, interconectadas en una red de sinapsis [3]. El cerebro es responsable de la mayoría de nuestras acciones y sentimientos, nuestras alegrías, risas, arrepentimientos, dolor, lágrimas.

Iniciando con los egipcios en el siglo XVII A.C. [8], el estudio del cerebro ha permanecido como un tema de gran interés a largo de la historia. Esto, probablemente, motivado por la necesidad innata del ser humano de entender su existencia. En la actualidad, el estudio del cerebro ha tomado un papel central en las ciencias, motivado por la necesidad de entender los mecanismos que regulan nuestro comportamiento, de tratar y curar enfermedades neurológicas, e incluso desarrollar inteligencia artificial. Evidencia de esto son proyectos a escala global como “*The Human Brain Project*” auspiciado por la Unión Europea y con un presupuesto de 1.019 mil millones de euros, o la iniciativa “*BRAIN*” de los Estados Unidos con un presupuesto inicial de 110 millones de dólares anuales por nueve años.

Históricamente, la Neurociencia emergió con la obra del médico español Ramón y Cajal (1852-1934) [8]. Su obra fue desarrollada con la ayuda del surgimiento del microscopio y técnicas de fijación y tinción de los tejidos celulares. El gran descubrimiento de Cajal está resumido en su “doctrina

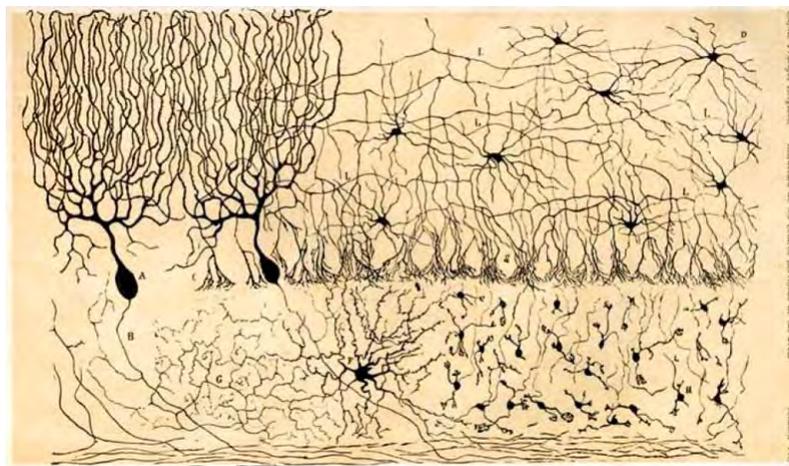


Figura 1.1: Células del cerebelo de Ramón y Cajal, mostrado en “Estructura de los centros nerviosos de las aves”, Madrid, 1905.

neuronal”: el sistema nervioso está formado por neuronas interconectadas (Fig. 1.1). Este descubrimiento transformó nuestro entendimiento del cerebro y del ser humano.

Otro descubrimiento fundacional fue hecho por Ross Harrison, quien descubrió que los conos de crecimiento de los axones guían el crecimiento de estos hacia sus lugares de destino [10], implicando la presencia de la estructura ordenada en las redes neuronales y arrojando luz a la función de las distintas partes de la neurona. Luigi Galvani fue el primero en notar que las células musculares producen electricidad, lo que motivó a Muller, Helmholtz, y Dubois-Reymond a desarrollar los fundamentos de la electrofisiología [10]. Helmholtz descubrió que la actividad eléctrica de las células nerviosas es la forma en que la “información” se transmite de un extremo a otro [10]. Por “información” aquí nos referimos a los impulsos neuronales, pues la cantidad de disparos, el promedio y los tiempos de disparo, proveen un código, el cual se traduce en nuestras acciones, pensamientos, sentimientos.

En los años sesenta del siglo XX, la bioquímica ofreció nuevas herramientas fundamentales a la neurología [10]. En particular, Hornykiewicz observó que los pacientes con enfermedad de Parkinson tienen déficit del neurotransmisor dopamina [10]. Esta fue la primera ocasión en que se documentó una correlación entre el déficit de un neurotransmisor y algún trastorno neurológico. La administración de L-Dopa — un precursor de la dopamina — es aún

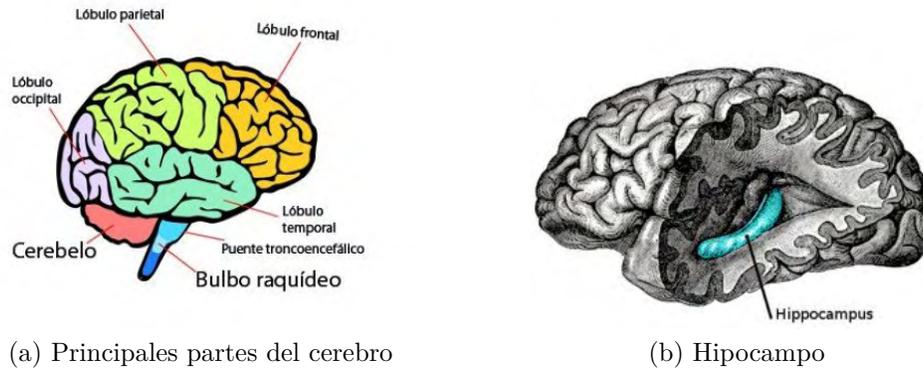


Figura 1.2: Cerebro humano

un tratamiento eficaz, aunque no definitivo, para los enfermos de Parkinson.

En el siglo XIX se planteó un problema fundamental en la historia de la Neurociencia: localizar anatómicamente las funciones en el cerebro [10]. Este problema aún no ha sido resuelto completamente. Sin embargo se ha llegado a un consenso, clasificando las estructuras como se muestra la Figura 1.2a.

En esta tesis nos centraremos en una estructura particular nombrada hipocampo. El hipocampo, es una de las partes del cerebro más importantes. Es parte del sistema límbico y se ubica en la parte interior del lóbulo temporal (Fig. 1.2b). Se ha mostrado que el hipocampo está relacionado con el almacenamiento y recuperación de recuerdos, el aprendizaje, emociones y la navegación espacial. Le pusieron el nombre hipocampo pues su estructura se asemeja a un caballito de mar [7]. El hipocampo es una de las regiones evolutivamente más antiguas de la corteza cerebral [8]. La historia del descubrimiento del hipocampo es muy interesante [11]. Fue gracias al paciente Henry Molaison, más conocido en la ciencia como HM, quien sufrió un fuerte golpe en la cabeza al ser atropellado por un ciclista a los 9 años de edad. Durante los 18 años posteriores tenía frecuentemente fuertes convulsiones y pérdida de conciencia debido a la epilepsia. Como no le era posible continuar viviendo así, decidió someterse a una intervención quirúrgica. Se le extirpó una parte del lóbulo temporal, perdiendo la mayor parte del hipocampo.

El tratamiento fue exitoso en el sentido que HM dejó de sufrir convulsiones constantes. Sin embargo, perdió la capacidad de mantener memoria por más de 20 segundos (su diagnóstico clínico fue “amnesia anterógrada y retrógrada”). Es decir, solo conservó su memoria a corto plazo, la que al-

macena cosas por no más de 20 segundos, y la memoria inconsciente, que es la memoria motora. Henry Molaison murió en el año 2008, pero gracias a él se hicieron grandes avances en el conocimiento del aprendizaje y la memoria. En particular, su caso permitió mostrar que el hipocampo tenía un rol fundamental en la memoria a largo plazo y el aprendizaje. De hecho, la región hipocampal es una de las primeras zonas en las que se hacen notorias enfermedades como demencia, amnesia y Alzheimer.

El camino para descubrir la función y localización de las funciones cerebrales ha sido muy largo y muy accidentado. Por ejemplo, inicialmente el médico y neuroanatomista austriaco Franz Joseph Gall (1757-1828) postuló que el cerebro no es un sólo órgano, sino que consiste en al menos 35 “centros”, cada uno de los cuales se relaciona con una función diferente. Además, Gall pensó que cada centro aumentaría de tamaño cuanto más funcionase, análogamente a como el tamaño de los músculos aumenta con el ejercicio. El crecimiento de los centros originaría una protrusión en el cráneo, y por lo tanto la forma del cráneo revelaría la personalidad del individuo [8]. Con base en la hipótesis de Gall, la forma del cráneo de una persona podría usarse para revelar si tenía tendencias criminales. Esta práctica se llama frenología, pero no tenía ningún fundamento científico y fue desprestigiada, pues se ha demostrado que las localizaciones del cerebro son borrosas (i.e., distribuidas), y las funciones no se realizan en lugares específicos, sino que varias secciones forman parte de estas funciones.

Otro ejemplo interesante es el caso del cirujano francés Pierre Paul Broca (1824-1880) quien describió en 1861 a un paciente que podía entender lo que le decían pero no podía hablar [10]. Su cerebro presentaba una lesión en la parte posterior del lóbulo frontal izquierdo, que hoy se conoce como área de Broca. En los años siguientes, Broca estudió a varios pacientes más, casi todos con lesiones en el hemisferio cerebral izquierdo, lo que le llevó a hacer una de las afirmaciones más conocidas sobre la función cerebral: “*Nous parlons avec l’hémisphère gauche!*” (“¡hablamos con el hemisferio izquierdo!”). En efecto, se ha verificado que la zona cerebral del habla se encuentra en el hemisferio izquierdo del cerebro.

En 1874, Karl Wernicke (1848-1904) estudió pacientes que, al contrario de los pacientes de Broca, podían hablar, pero no entender, y su lesión cerebral estaba en la parte posterior del lóbulo temporal izquierdo. Además de presentar sus descubrimientos, Wernicke propuso una nueva teoría de la función cerebral, llamada “conectivismo”[10]. Según esta teoría, sólo las funciones mentales más elementales, como las actividades motoras o perceptivas

sencillas. Se localizan en una única región cerebral. El conectivismo propone que las conexiones entre las diversas áreas hacen posible las funciones intelectuales complejas como la memoria o el aprendizaje. Esta noción de que una función involucra varias áreas del cerebro se conoce hoy como “procesamiento distribuido” y es posiblemente uno de los conceptos más prolíficos en la Neurociencia moderna. A excepción de los traumas, recientemente se ha mostrado que la mayoría de las enfermedades que afectan al sistema nervioso tiene algún componente heredado [8]. Este hecho sugiere de que a medida que se van haciendo no solo avances en Neurociencia, sino también en genética, nuestra manera de resolver problemas en neurología y psiquiatría será cada vez más exitosa.

La breve descripción anterior muestra cómo la Neurociencia se ha convertido en un área muy multidisciplinaria, resultado de la convergencia de varias ciencias: Anatomía, Embriología, Fisiología, etc. Las matemáticas, la computación y la ingeniería se han sumado al reto no hace mucho.

La meta de las neurociencias es “resolver el cerebro”, es decir, entender cómo pensamos, cómo nos movemos, cómo recordamos. Las neuronas en el cerebro forman una enorme red de nodos (i.e. neuronas) interconectados. Esta red sirve para transmitir impulsos de una neurona a otra por medio de neurotransmisores y procesos químicos que aumentan o disminuyen la diferencia de potencial eléctrico del interior de las neuronas con el exterior. Esta diferencia de potencial genera impulsos eléctricos, comunmente llamados “acciones potenciales” [1, 8, 2]. Las acciones potenciales son eventos eléctricos generados cuando el potencial de membrana de la neurona sobrepasa un umbral (típicamente de -70 mV). Al pasar este umbral, se genera un cambio de potencial subsecuente muy marcado, y produciendo un impulso eléctrico. El momento de cruce del umbral se dice que la neurona “dispara” [1]. Las neuronas vecinas son sensibles a estos disparos pues su potencial de membrana es modificado. La cantidad de veces que una neurona dispara, el tiempo transcurrido entre cada disparo, el orden de los disparos entre múltiples neuronas, etc., conforman “el código neuronal” [1], que codifica nuestras acciones, pensamientos, controla funciones corporales tales como el latido de corazón o la producción de hormonas, y todo lo relacionado con las funciones del cerebro [1, 8]. Descifrar el código neuronal “resolvería” el cerebro, permitiéndonos entender como es que cada acción se genera y codifica en el cerebro. Por su dificultad, descifrar el código neuronal es una de las grandes y lejanas metas de la neurociencia moderna, sin embargo, se han hecho avances sorprendentes.

## 1.2. Neurociencias y matemáticas

Las neurociencias y las matemáticas han mantenido una relación muy estrecha. El primer modelo matemático reconocido en la neurociencia aparece con la investigación hecha por los fisiólogos Alan Lloyd Hodgkin (1914-1998) y Andrew Fielding Huxley (1917-2012) sobre el axón gigante del calamar [14]. Este modelo es considerado uno de los mayores hallazgos en biofísica del siglo XX, que les valió a ambos investigadores un premio Nobel. El modelo consiste en un sistema de ecuaciones diferenciales con cuatro variables de estado interconectadas de forma muy no lineal.

El modelo Hodgkin-Huxley fue el parteaguas para el estudio matemático de las neurociencias, siendo seguido por muchos otros modelos para estudiar el funcionamiento de neuronas. Esto dio a luz a modelos de menores dimensiones y mucho más fáciles de analizar, que son bastante populares en la actualidad. Ejemplos de estos modelos son el de Morris-Lecar, Fitzhugh-Nagumo y, uno de los más usados, el *Integrate-and-Fire model*, que es el que emplearemos en esta tesis. El estudio de modelos matemáticos en las neurociencias permanece siendo una de las herramientas fundamentales de esta ciencia. En la actualidad existe un énfasis [articular en la investigación de modelos de redes de neuronas [1]. Esta Tesis se basa en la construcción de un modelo de redes de neuronas, el cual busca explicar las observaciones experimentales recientemente reportadas que se describen a continuación.

## 1.3. El experimento de Slomowitz et al. sobre el hipocampo del ratón.

En el artículo “*Interplay between population firing stability and single neuron dynamics in hippocampal networks*” [3], Slomowitz et al. realizan un estudio experimental en el hipocampo de ratones para responder las siguientes dos preguntas fundamentales:

- ¿Cuáles son las propiedades básicas de las redes neuronales donde existe el proceso de homeostasis (es decir un conjunto de eventos de autorregulación conducentes al mantenimiento de una relativa constancia en la composición y propiedades del medio interno de un organismo)?
- La homeostásis en dichas redes, ¿actúa a nivel neuronal individual y/o poblacional?

Para responderlas, se hizo una serie de experimentos sobre ratones que se podían mover libremente. Se registraban datos de los impulsos surgidos en la parte CA1 del hipocampo en estado normal y durante “perturbaciones” inducidas por la aplicación de la droga baclofeno. Esta droga inhibe la actividad sináptica, pues es agonista de los receptores de  $GABA_B$ , que es el principal neurotransmisor inhibitorio. Los experimentos reportan un fenómeno inesperado y muy interesante: por un lado, como era de esperarse, la tasa de disparo poblacional (i.e., el promedio de las tasas individuales de disparo) disminuyó una vez aplicada la perturbación. Después de dos días con el estimulante inyectado constantemente, casi el 90 % de las neuronas tenían tasa de disparo individual distinta a su tasa de disparo original. Sin embargo, y sorprendentemente, la tasa de disparo poblacional se restauró a un nivel muy cercano al que tenía antes de que la perturbación fuera aplicada. Es decir, hubo un proceso de autorregulación u homeostasis, pero este fue a nivel neuronal poblacional, no individual. Mas específicamente, las neuronas cambiaron su tasa de disparo individual de tal forma que la tasa de disparo poblacional regresa al valor que tenía antes de que la perturbación fuera aplicada.

## 1.4. Objetivos

Para explicar los resultados experimentales de Slomowitz et al. [3], proponemos la siguiente hipótesis:

<p>“La tasa de disparo promedio de una red neuronal puede ser restaurada a un nivel pre-perturbación a través de un mecanismo de retroalimentación que ajusta la fuerza de las conexiones sinápticas”.</p>	ff
--	----

El objetivo general de esta tesis es construir un modelo matemático para estudiar la dinámica de una red de neuronas con nuestra hipótesis y analizar su plausibilidad biológica. Los objetivos específicos de la tesis son:

1. Construir un modelo matemático para un número arbitrario de neuronas conectadas a través de una red de interconexión arbitraria. Para esto utilizaremos el modelo *Integrate-and-Fire* para simular el método de disparo de neuronas, y modelos de redes aleatorias para sus interconexiones. Estos modelos se explicarán en la siguiente sección.
2. Construir un algoritmo de retroalimentación que permita ajustar los

pesos de interconexión entre neuronas para mantener la tasa de disparo poblacional constante. Para esto construiremos un algoritmo “biológico” basado en el modelo Hebbiano de aprendizaje, y otro “artificial”.

3. Analizar el comportamiento de los algoritmos de retroalimentación propuestos, y analizar su plausibilidad biológica para explicar las observaciones experimentales.
4. Programar un Notebook de Mathematica que implementa los puntos 1 y 2 arriba mencionados para redes arbitrarias. El objetivo se logró, y el código es abierto (ver Capítulo Código) y se puede descargar desde Github: <https://github.com/liubaoryol/neural-net-with-Hebbian-learning>

# Capítulo 2

## Construcción del modelo

### 2.1. Antecedentes: modelos matemáticos de neuronas individuales

Los modelos basados en la conductividad eléctrica de las neuronas pueden reproducir con muy alta precisión las medidas electrofisiológicas que han sido obtenidas experimentalmente [1], pero frecuentemente son muy complejos de analizar matemáticamente [1]. Por esta razón, los modelos fenomenológicos enfocados en describir únicamente los disparos de una neurona son muy populares para estudios del código neuronal, memoria y la dinámica de una red de neuronas interconectadas.

En este capítulo analizaremos dos modelos clásicos. El primero fue creado por Hodgkin y Huxley, y es esencial e importantísimo de ver, porque fue el punto de partida de la neurociencia computacional. Este modelo consiste en un sistema de ecuaciones diferenciales ordinarias no lineales acopladas que describe las corrientes eléctricas que fluyen a través de los canales iónicos de una neurona. El espacio de parámetros de estas ecuaciones es multidimensional. Todo esto mencionado hace extremadamente difícil de analizar el sistema, por lo que raramente se usa para análisis computacionales. Sin embargo, sería un gravísimo error no presentarlo, ya que de él parten muchos otros modelos matemáticos populares, como es el caso del de Morris-Lecar o el de Fitz-Naghumo[1, 2]. Estos dos últimos son similares al de Hodgkin-Huxley pero de menor dimensión.

Para eludir la complejidad del modelo de Hodgkin-Huxley se presenta el segundo modelo de esta tesis. Es un modelo fenomenológico simplificado muy

popular, conocido como *Integrate-and-Fire* que sólo se enfoca en describir los disparos neuronales, y no explica la conductividad de la neurona, como lo hacen los modelos anteriormente presentados. El modelo Integrate-and-Fire provee la base del modelo computacional que construimos en esta Tesis.

### 2.1.1. Modelo de Hodgkin-Huxley

Como mencionamos, los primeros que propusieron un modelo matemático de la neurona fueron Hodgkin y Huxley (HH), quienes construyeron este modelo partiendo de su investigación sobre el calamar en 1952 (aprovechando que este posee axones gigantes)[14]. Este modelo fue quizás el mayor descubrimiento del siglo XX en medicina, el cual ameritó un Premio Nobel en fisiología-medicina.

En las neuronas del calamar hay tres tipos de canales iónicos, y por ende, tres tipos iones. Estos iones son: sodio (Na), potasio (K) y los llamados “iones de fuga” (L), que es un conglomerado de varios tipos de iones (principalmente  $\text{Cl}^-$ ). Esencialmente, el modelo de HH consiste en modelar estas tres corrientes iónicas a través de resistencias o conductancias eléctricas de la membrana celular de una neurona.

Las neuronas del calamar son fáciles de analizar, pues son más grandes y utilizan una menor variedad de iones comparada con otras neuronas (e.g., humanas). Sin embargo, su análisis resulta muy esclarecedor pues el mecanismo de iones que tiene se preserva en organismos más complejos. No hay que olvidar que las neuronas en organismos mas complejos exhibirán muchas más propiedades por tener mayor variedad de iones.

El modelo HH se construye como sigue. La membrana se modela como un capacitor y el potencial de Nernst (ver Apéndice I), generado por la diferencia entre concentración iónica, es modelado como una batería (Fig 2.1). Con base en esta representación eléctrica, las ecuaciones diferenciales que describen la electrofisiología de la neurona toman la forma:

$$C \frac{du(t)}{dt} = - \sum_k I_k(t) + I(t),$$

donde  $C$  es la conductancia eléctrica,  $u(t)$  es el potencial de membrana,  $I(t)$  es la corriente suministrada del exterior, y  $\sum_k I_k(t)$  es la suma de las corrientes iónicas que pasan a través de la membrana.

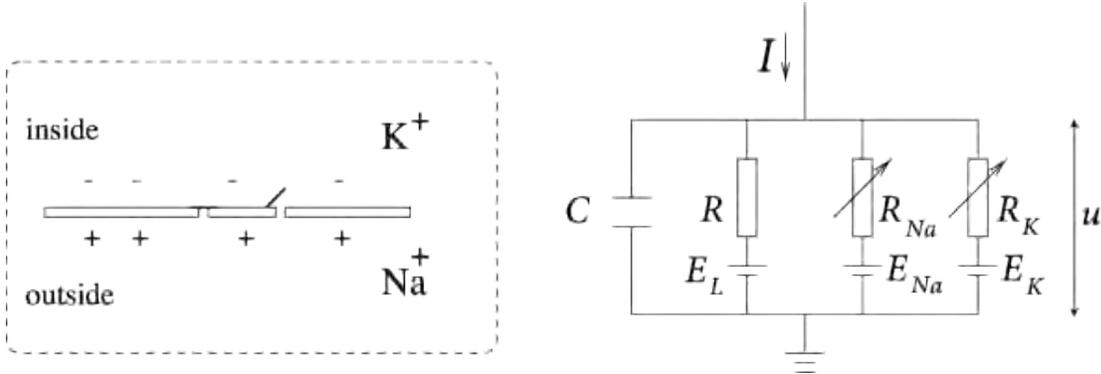


Figura 2.1: Diagrama del modelo Hodgkin-Huxley. Del lado izquierdo se muestra la membrana neuronal y la concentración predominante de iones del exterior y el interior de la membrana. Del lado derecho demuestra el hecho que el modelo de Hodgkin-Huxley hace referencia a un circuito eléctrico

La suma de corrientes iónicas satisface la siguiente igualdad [1]:

$$\sum_k I_k = g_{Na} m^3 h (u - E_{Na}) + g_K n^4 (u - E_K) + g_L (u - E_L), \quad (2.1)$$

donde

- $g_L = 1/R$ , con  $R$  la resistencia del “canal de fuga”.
- $g_x$  - conductividad de  $x \in \{K, Na\}$ , es decir, conductividad del potasio y sodio. Si todos los canales iónicos están abiertos, estos transmiten corrientes de máxima conductividad  $g_x$ . Su valor puede ser calculado empíricamente a través de mediciones (ver Tabla 2.1).
- $E_{Na}, E_K, E_L$ -potenciales de inversión (ver Apéndice I). Estos valores también son calculados empíricamente y se muestran en la Tabla 2.1. Son desplazados por +65 mV para tener el potencial de reposo de la membrana celular nula.

$x$	$E_x$	$g_x$
Na	$115mV$	$120mS/cm^2$
K	$-12mV$	$36mS/cm^2$
L	$10.6mV$	$0.3mS/cm^2$

Tabla 2.1: Conductividad y potenciales de inversión del sodio, potasio y los iones de fuga

Las variables  $m, n, h$  de la ecuación 2.1 son conocidas como “variables de activación”: son variables adimensionales entre 0 y 1 y describen la probabilidad de que el canal de sodio esté activo, el de potasio estáctivo, o el de sodio este inactivo, respectivamente. Las variables de activación evolucionan de acuerdo con las siguientes ecuaciones diferenciales [1]:

$$\begin{aligned}\dot{m} &= \alpha(u)(1 - m) - \beta_m(u)m, \\ \dot{n} &= \alpha(u)(1 - n) - \beta_n(u)n, \\ \dot{h} &= \alpha(u)(1 - h) - \beta_h(u)h.\end{aligned}$$

Las funciones  $\alpha$  y  $\beta$  son construidas empíricamente y fueron ajustadas por Hodgkin y Huxley con base en datos experimentales.

El modelo HH se comporta (intuitivamente) de la siguiente manera. Un estímulo proveniente del exterior provoca el aumento de  $u$ , causando que la conductividad de los canales de sodio aumente. Es decir, hay mayor flujo de sodio hacia dentro de la célula, mientras que la conductividad de los canales de potasio disminuyen, causando un menor flujo de potasio hacia el exterior. A su vez esto causa que  $u$  aumente aún más, lo que puede causar una acción potencial (ver Apéndice I). Una acción potencial es como una “fiesta de iones”. El modelo HH está entonces compuesto por ecuaciones diferenciales no lineales muy difíciles de analizar, pues están acopladas entre si.

Para subsanar esta limitante, un modelo fenomenológico simplificado muy popular es el modelo de Integrate-and-Fire que es, simplemente, un reinicio de la variable  $u$  en el modelo de Hodgkin-Huxley.

### 2.1.2. Modelo Integrate-and-Fire

Este es un modelo fenomenológico para los disparos de una neurona, el cual es muy popular para estudios del código neuronal, memoria, y la dinámica de una red de neuronas interconectadas [1]. De entre los modelos

existentes sobre disparos elegimos este en particular porque una de las funciones centrales del hipocamos es la memoria.

El modelo que elegimos y presentamos aquí es un modelo de umbral “duro” [1, 2], es decir, los disparos son generados cuando el potencial de membrana  $u(t)$  cruza cierto umbral fijo  $\gamma > 0$ . Así, la condición  $u(t) = \gamma$  genera un disparo. A  $\gamma$  se le llama umbral duro, pues define claramente el comienzo de un disparo, lo que es biológicamente poco realista. Experimentalmente se ha demostrado que un disparo inicia en distintos valores del potencial de membrana. El instante de tiempo en que el potencial de membrana cruza el umbral se define como el “tiempo de disparo”  $t^{(f)}$ :

$$t^{(f)} : u(t^{(f)}) = \gamma, \left. \frac{du(t)}{dt} \right|_{t=t^{(f)}} > 0.$$

El umbral biológico empíricamente calculado es de -55 mV, es decir 10mV más alto que el potencial de reposo. [1, 2, 8, 7]. Por esto, usualmente el potencial de membrana se desplaza por +65 mV, es decir, para que el potencial de reposo sea nulo, y el umbral a 10 mV. Así, una neurona dispara al llegar a 10 mV, en vez de -55 mV, y se reinicia al valor potencial 0, en vez de -65 mV. También es importante mencionar que, fisiológicamente, el potencial de membrana no puede ser menor a -80 mV aproximadamente. Desplazando por +65 mV, tenemos una restricción por abajo para el potencial de membrana: este no puede bajar a más de -15 mV.

El modelo Integrate-and-Fire describe a una neurona como un circuito eléctrico, que tiene un capacitor eléctrico  $C$  en paralelo con una resistencia  $R$ , excitado por una corriente externa  $I(t)$  dependiente del tiempo.  $I(t)$  se puede desglosar en dos componentes: la corriente resistiva  $I_R$  que atraviesa el resistor lineal  $R$  y la corriente  $I_C$  que fluye a través del condensador eléctrico:  $I(t) = I_R + I_C$ .

Por la ley de Ohm:

$$I_R = u/R,$$

y por la definición de capacitancia eléctrica

$$I_C = C \frac{du}{dt}.$$

Después de un poco de álgebra, obtenemos:

$$\tau_m \frac{du}{dt} = -u(t) + RI(t) \quad (2.2)$$

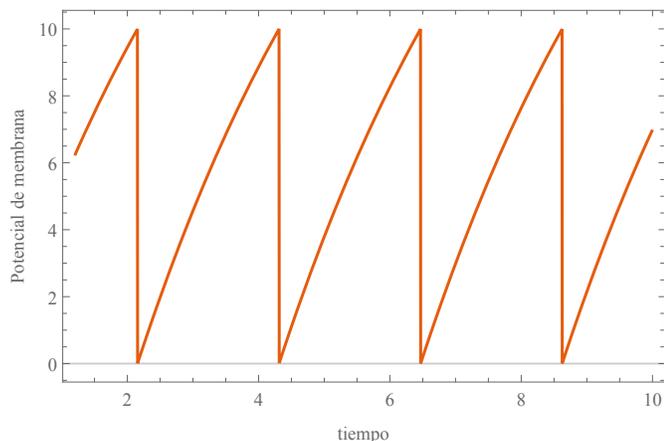


Figura 2.2: Comportamiento de una neurona utilizando el modelo Integrate-and-Fire de las ecuaciones. 2.1 y 2.2. Una corriente exterior constante de 6 unidades es aplicada. Aquí  $R = 4$  unidades. Las corrientes externas son aplicadas para inducir disparo neuronal, sin embargo las perturbaciones del baclofeno afectan los pesos (Nota: las unidades eléctricas no son reales, sino una medida de comparación y análisis)

donde  $\tau_m = RC$  se conoce como la constante de tiempo de la membrana. Sin pérdida de generalidad, supondremos que  $\tau_m = 1$  que equivale a un cambio de escala en el tiempo. Ya que el disparo es un proceso que dura muy poco tiempo y donde el voltaje sube y baja casi de manera instantánea, el suceso se define con una función delta de Dirac, que provoca que el potencial sea instantáneamente restaurado a su nivel de reposo. Esto se describe suplementando la ecuación (2.2) con la siguiente ecuación:

$$u(t^+) = 0 \text{ si } u(t) = \gamma.$$

Aquí  $u(t^+) = \lim_{x \rightarrow t^+} u(x)$ . El comportamiento típico de una neurona con corriente externa constante se ilustra en la Fig. 2.2.

La corriente externa  $I(t)$  es el estímulo externo a la neurona, la cual puede codificar la información recibida desde otras neuronas (i.e. las neuronas presinápticas). Las entradas sinápticas se discuten con mayor detalle en la siguiente sección, donde se define el comportamiento de las neuronas dentro de una red.

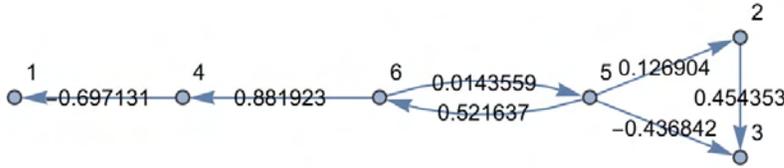


Figura 2.3: Grafo Erdős-Renyi con 6 nodos y cuya probabilidad de conexión  $p = 0.3$

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.454353 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -0.697131 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.126904 & -0.436842 & 0 & 0 & 0.521637 \\ 0 & 0 & 0 & 0.881923 & 0.0143559 & 0 \end{pmatrix}$$

Figura 2.4: Matriz de pesos del grafo mostrado en la Fig. 2.3

## 2.2. Construcción del modelo para una red de neuronas

Una neurona por sí sola no puede realizar muchas funciones, pues la información que procesa y guarda no es transmitida a ningún lado y por ende no será provechosa. Por esto analizaremos la dinámica de neuronas en una red, y no sólo la dinámica de una neurona independiente. La red a estudiar será una red de neuronas homogéneas, es decir que el valor del umbral, la resistencia y la constante de tiempo será el mismo para todas las neuronas dentro de dicha red.

Describiremos una red de  $N$  neuronas como un grafo dirigido  $\mathcal{G}$  [12, 1] con nodos  $\{1, \dots, N\}$ . Cada neurona es asociada a un nodo. La arista ( $j \rightarrow i$ ) denota que la  $j$ -ésima neurona está conectada (i.e., tiene una sinapsis) con la  $i$ -ésima neurona. Permitiremos que las aristas tengan signo y peso, de tal forma que ( $j \rightarrow i$ ) está asociada a un número real  $w_{ij} \in [-1, 1]$ . Una interacción corresponde a una sinapsis inhibitoria cuando  $w_{ij} < 0$ , y a una sinapsis excitatoria cuando  $w_{ij} > 0$ . La magnitud de la interconexión esta determinada por  $|w_{ij}|$ . Mientras más grande es el peso de una interconexión, más grande es el efecto de la  $j$ -ésima neurona en la  $i$ -ésima. De esta forma, la matriz de adyacencia ponderada  $W = (w_{ij}) \in \mathbb{R}^{N \times N}$  del grafo dirigido  $\mathcal{G}$ , describe las interacciones entre neuronas (Figura 2.4).

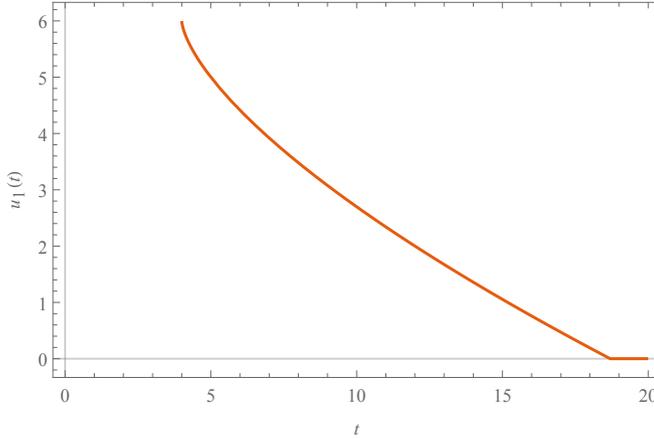


Figura 2.5: Gráfica del potencial postsináptico que obtiene la neurona que recibe un disparo al tiempo  $t = 4$

Cuando una neurona dispara, esta emite una corriente conocida como “corriente postsináptica”, que denotaremos como  $PSC(t^f, t)$ . Empíricamente [1], la corriente postsináptica se modela como una función decreciente en el tiempo (ver Fig. 2.5) y sigue la siguiente ecuación:

$$PSC(t^{(f)}, t) = \max \left[ (-t + t^{(f)})^{\frac{2}{3}} + 6, 0 \right] \mathcal{H}(t - t^{(f)}). \quad (2.3)$$

Aquí  $\mathcal{H}$  es la función escalón de Heaviside, también llamada función escalón unitario, que se define de esta forma:

$$\forall x \in \mathbb{R} : \quad \mathcal{H}(x) = \begin{cases} 0 & \text{si } x < 0 \\ 1 & \text{si } x \geq 0 \end{cases}$$

La corriente postsináptica de una neurona se propaga hacia todas sus neuronas vecinas, aumentando linealmente el potencial de membrana de cada una de esas neuronas vecinas. Eventualmente, estas neuronas pueden alcanzar su nivel de umbral y disparar, propagando su propia corriente postsináptica.

Este mecanismo nos permite modelar la corriente externa de una neurona como la corriente eléctrica externa de la red, mas el  $PSC(t^{(f)}, t)$  multiplicado por el peso de la arista entre los dos nodos. Los pesos positivos incrementan el voltaje, excitando a la neurona, mientras que los negativos hiperpolarizan, las inhiben y es menos probable que dispare. Matemáticamente, esto se describe:

$$\tau_m \dot{u}_i = -u_i(t) + R(I_{ext}(t) + \sum_j w_{ij} PSC(t^{(f_j)}, t))$$

donde :

$u_i$  - potencial de membrana de la  $i$ -ésima neurona.

$\tau_m, R$  - constantes como en la ecuación 2.2.

$I_{ext}(t)$  - corriente suministrada del exterior.

$w_{ij}$  - el peso de la conexión de la neurona  $j$  a neurona  $i$ .

Para crear una red de interconexión entre  $N$  neuronas, utilizamos el modelo de Erdos-Renyi [9] para crear grafos aleatorios con  $N$  nodos (cada nodo corresponde a una neurona). Este modelo tiene como parámetro la probabilidad  $p \in [0, 1]$  de conexión entre dos nodos. Cuando  $p = 0$  se obtiene una totalmente desconexa, mientras que con  $p = 1$  se obtiene una gráfica completa. El peso  $w_{ij}$  de cada una de las aristas generadas por el modelo es después elegido como un número aleatorio en el intervalo  $[-1, 1]$ . En Wolfram Mathematica utilizamos el siguiente código:

```
In[1]:= weightedNet[n_,p_]:= Module[{ g, Wnet},
  g = RandomGraph[BernoulliGraphDistribution[n,p],
  DirectedEdges→ True,VertexLabels→ Name];
  Wnet =
  Graph[g,EdgeWeight→ RandomReal[{-1,1},EdgeCount[g]]];
  Return[Wnet]
]
```

En la Fig. 2.3 se presenta un ejemplo de una red aleatoria con  $N = 6$  nodos construida utilizando este método. Los números sobre las aristas indican los pesos de interconexión. En esta red, el nodo 4 manda información al nodo 1 y recibe información del nodo 6. El nodo 1 no manda información a ningún nodo, sus impulsos se quedan solo con él.

Vemos que el peso de la conexión de la neurona 6 a la 4 es de 0.882; esto es una conexión bastante intensa, pues el máximo peso que puede alcanzar una arista es uno. De la neurona 6 a la neurona 5 el peso es muy chico, de 0.014, se podría decir que casi imperceptible, pues esta muy cerca de ser un peso nulo. Es decir, los disparos de la neurona 6 tienen poca influencia sobre la neurona 5, pero mucha influencia sobre la neurona 4.

Ya que los pesos de las aristas entre las neuronas 4 y 6 y las neuronas 5 y 6 son positivos, las sinapsis entre estas neuronas son excitatorias. Cada vez que la neurona 6 dispare, se añade corriente externa positiva entrante a las neuronas 4 y 5, propiciando que disparen.

Note también que la neurona 5 manda señales a las neuronas 2, 3 y 6. Nótese que el peso de la conexión con la neurona 3 es negativo, tratándose de una sinapsis inhibitoria. Es decir, cuando la neurona 5 dispara, la neurona 3 tiene menos probabilidad de alcanzar el umbral y disparar, pues su potencial de membrana disminuyó. Esta red contiene un *ciclo* (i.e., un camino en la red que empieza y termina en el mismo nodo), formado por las neuronas 5 y 6. De la 6 a la 5 la señal es débil, y de la 5 a la 6 si es bastante fuerte.

Para facilitar la visualización de una red, normalmente no mostraremos los pesos de las aristas. Note también que el grafo ponderado puede ser descrito equivalentemente por su *matriz de adyacencia*  $(w_{ij}) \in \mathbb{R}^{n \times n}$ , donde  $w_{ij}$  es el peso de la interacción ( $j \rightarrow i$ ) (Fig. 2.4)

El procedimiento arriba descrito permite crear grafos aleatorios ponderados de cualquier dimensión que describen la red de interconexión entre neuronas.

Al tener las neuronas conectadas dentro de una red, sus disparos tienen influencia sobre el potencial de membrana de otras neuronas, cambiando su tasa de disparo. El comportamiento dinámico de una red de neuronas es mucho mas rico que el de una sola neurona, donde habíamos visto que tiene una tasa de disparo constante. Por ejemplo, la figura 2.6 muestra una red de tres neuronas que ocasiona que cada una de ellas tenga un comportamiento muy distinto.

### 2.3. Construcción de algoritmos de aprendizaje

Hasta ahora hemos vistos los modelos matemáticos de una neurona aislada, como se comporta dentro de una red, y como interactúa con otras neuronas. Vimos que a las conexiones entre neuronas, es decir, a los pesos de las aristas, se inicializan con números reales en el intervalo  $[-1,1]$ , y representan sinapsis inhibitorias o excitatorias y nos dicen qué tan dependientes son las neuronas entre ellas. Hasta ahora hemos considerado dichos pesos estáticos. Ahora le agregamos otro grado de libertad a la red: que los pesos sean dinámicos. Biológicamente, resulta plausible que la forma en que cambian los pesos no sea completamente aleatoria, sino tengan una dinámica que se adapta según los distintos estímulos externos, tratando de llevar la red a cierto estado deseado. Al cambiar los pesos de la red, se sigue una reacción

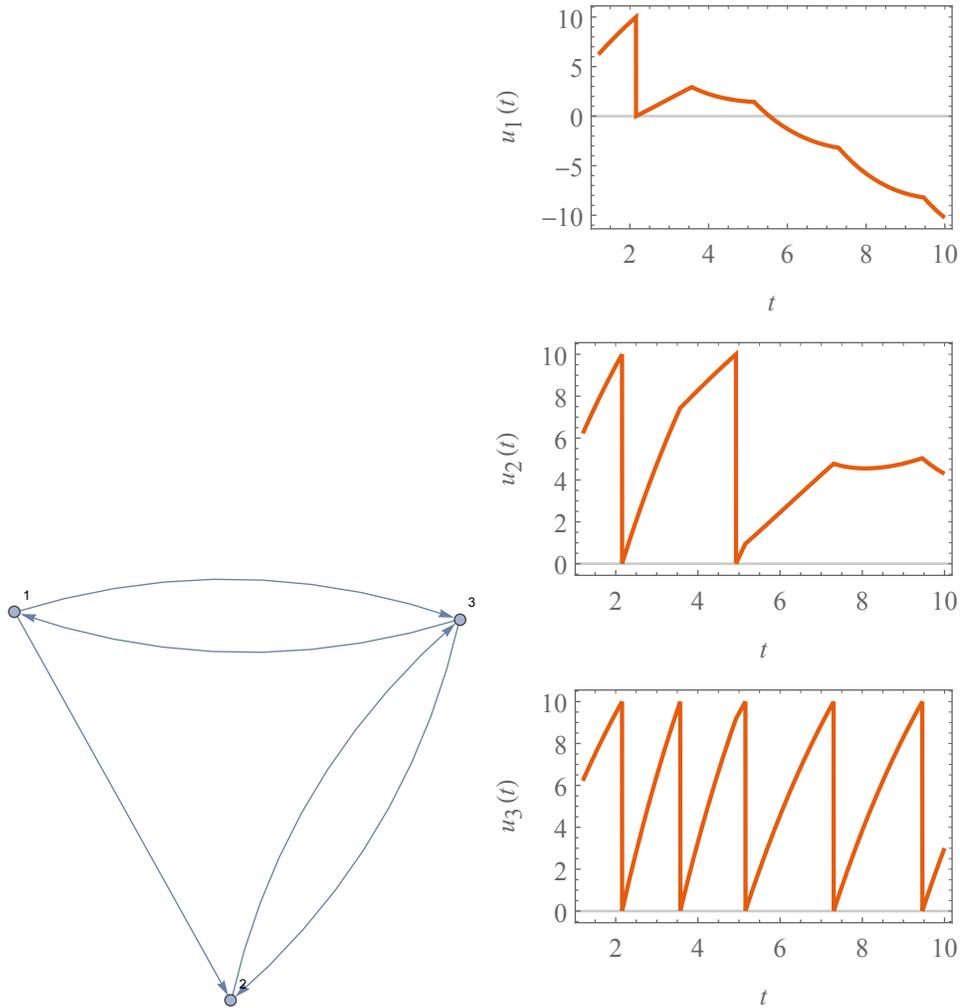


Figura 2.6: Red de tres neuronas y sus gráficas del potencial de membrana con corriente externa de 6 unidades en cada una de las neuronas y perturbación de -0.5 unidades sobre los pesos

en cadena, provocando que disminuya o aumente la frecuencia de disparos de otras neuronas.

Los pesos de las aristas forman una parte crucial del funcionamiento de una red. La posibilidad de que hayan cambios en las aristas se le llama “plasticidad neuronal” [1], mientras que al proceso mismo de cambio y al algoritmo que se usa para dicho proceso se le llama, genéricamente, “aprendizaje” [1].

Hay muchos modelos de aprendizaje reportados previamente (vea e.g., [1, 6, 5, 4]), En esta tesis exploraremos dos en particular: uno biológico y uno artificial (i.e. no biológicamente plausible, pero construido de tal forma que cumpla lo que requerimos).

### 2.3.1. Modelo biológico inspirado en la ley de aprendizaje de Hebb

*“Neurons that fire together, wire together”*

Donald Hebb (1904-1985) postuló que, cuando el axón de una neurona  $j$  contribuye persistentemente en disparar la neurona  $i$ , sucede un cambio metabólico en una o ambas neuronas de tal forma que la eficacia de  $j$  para disparar  $i$  se incrementa [1]. Es decir, si la neurona  $j$  dispara, y luego luego la neurona  $i$ , esto significaría que la neurona  $j$  al disparar mandó la señal decisiva para que la neurona  $i$  disparara.

Esta regla le pareció a Hebb lógica, pues pensó que sería un mecanismo que ayudaría a estabilizar algunos patrones de actividad neuronal en el cerebro [1]. Sin embargo, en 1949, cuando propuso ese modelo, no había tecnología suficiente para confirmar o refutar dicha teoría [1]. No fue hasta más de veinte años después que se pudo probar experimentalmente que Hebb tenía razón.

El modelo de aprendizaje de Hebb tiene dos características principales:

- Es dependiente del tiempo.
- Es altamente “selectivo”: solo cuando dos neuronas disparan **simultáneamente** es cuando la sinapsis conlleva proceso de modificación.

Es importante notar que la regla que propuso Hebb sigue un orden, la eficacia de la sinapsis sube si la neurona presináptica dispara justo antes que la neurona postsináptica. Pero, ¿qué pasa si es al revés, la neurona postsináptica dispara antes que la presináptica? Este caso fue añadido después, completando lo que ahora es conocido como “la regla de Hebb” [1]. En este

último caso, la neurona  $j$  no formaría parte en disparar la neurona  $i$ , por lo tanto la conexión no es tan necesaria, y así la eficacia de la sinapsis debería disminuir.

Las neuronas del hipocampo que estudiaremos en esta tesis, son un ejemplo de neuronas que cumplen la condición Hebbiana [1].

Unas ecuaciones sencillas que describen con detalle la teoría Hebbiana son las siguientes [1]. Para el peso  $w_{ij}$  de la neurona  $j$  a la neurona  $i$ , su dinámica es:

$$\dot{w}_{ij} = a_0 + \text{pre} + \text{post},$$

donde  $a_0$  modela la actividad independiente de la sinapsis. Las variables “pre” y “post” describen el primer y segundo postulado de Hebb. Estos tres componentes usualmente toman la forma [1]:

$$a_0 = -\gamma(1 + w_{ij})(1 - w_{ij})(w_\theta - w_{ij}), \quad (2.4)$$

$$\text{pre} = S_j(t) [a_1^{\text{pre}} + (a_2^{\text{pre,post}} * S_i)(t)];$$

$$\text{post} = S_i(t) [a_1^{\text{post}} + (a_2^{\text{post,pre}} * S_j)(t)].$$

$(f * g)(t)$  es la convolución de las funciones  $f$  y  $g$  sobre el dominio  $(0, \infty)$ . Se define como  $(f * g)(t) = \int_0^\infty f(s)g(t-s)ds$ .

En estas ecuaciones  $S_i(t) = \sum \delta(t - t_i^{(f)})$  y  $S_j(t) = \sum \delta(t - t_j^{(f)})$ , donde la función  $\delta$ , delta de Dirac, equivale a uno cuando el argumento es 0, y equivale a 0 en cualquier otro caso.

La variable  $a_0$  modela la actividad independiente de la sinapsis, es decir, muestra el comportamiento de los pesos cuando hay ausencia de disparos.  $a_0$  es estable cuando la condición inicial se encuentra en el intervalo  $[-1, 1]$ ;  $w_{ij} \in [-1, 1]$ . Por lo tanto siempre es estable, pues asumimos que los pesos en redes neuronales no pueden estar fuera del rango mencionado. El parámetro  $w_\theta$  es el umbral, el comportamiento de la dinámica de los pesos cambia de creciente a decreciente (o viceversa) al cruzar  $w_{ij} = w_\theta$  (Fig. 2.7).

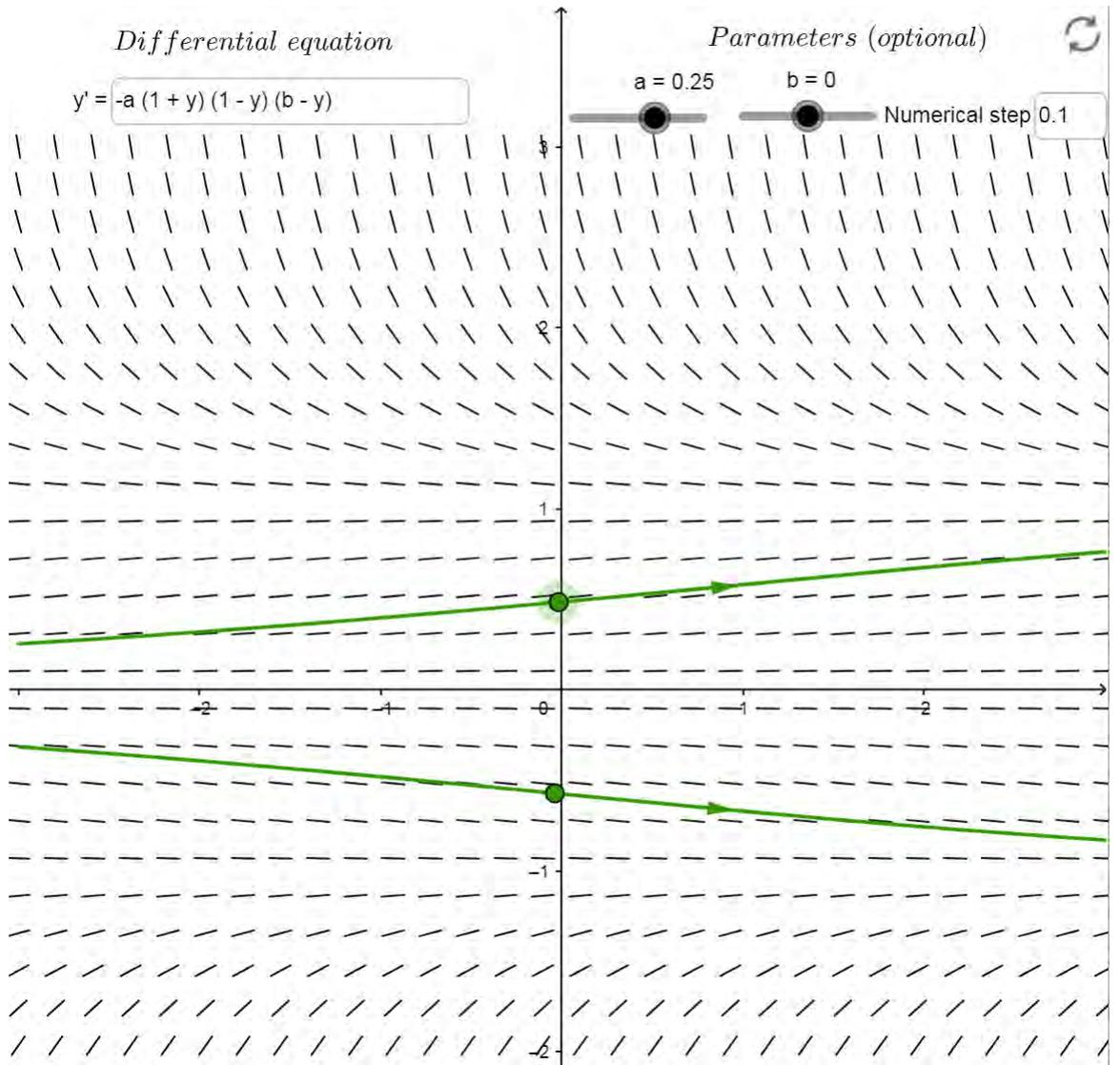


Figura 2.7: El campo de direcciones de la función de la actividad independiente de la sinapsis  $a_0$  definida en 2.4

La función  $a_2^{pre,post}(s)$  describe el cambio del peso de la conexión si una acción potencial postsináptica es seguida por un impulso presináptico en  $s$  unidades de tiempo. La función  $a_2^{post,pre}(s)$  describe el cambio del peso de la conexión si una acción potencial presináptica es seguida por un impulso postsináptico en  $s$  unidades de tiempo. Ambas funciones dependen de  $s$ , que

representa la diferencia de los tiempos de impulso post- y presinápticos. La función de plasticidad  $W(s)$  se llama ventana de aprendizaje y depende de la diferencia de los tiempos de disparo, donde el cambio en el aprendizaje es mayor mientras mas chica sea  $s$  (Fig. 2.8).

Una de las fórmulas matemáticas que describen la ventana de aprendizaje es la siguiente:

$$W(s) = \begin{cases} a_2^{post,pre}(-s) & \text{if } s < 0 \\ a_2^{pre,post}(s) & \text{if } s > 0 \end{cases} = \begin{cases} w_{ij}a_-e^{-\frac{s}{\tau_1}} & \text{if } s < 0 \\ (1 - w_{ij})a_+e^{-\frac{-s}{\tau_2}} & \text{if } s > 0 \end{cases} \quad (2.5)$$

donde  $s = t_j^{(f)} - t_i^{(f)}$  es la diferencia de los últimos tiempos de disparo entre neurona presináptica  $i$  y neurona postsináptica  $j$

Los componentes del aprendizaje Hebbiano pueden ser implementados en Wolfram Mathematica utilizando las siguientes funciones:

```
In[2]:=  $\gamma=1$ ;  $w_{thre}=0$ ;  $a_{plus}=1$ ;  $a_{minus}=-1$ ;
         $activind[w\_]:= -\gamma (1+w)(1-w)(w_{thre}-w)$ 
```

```
In[3]:=  $Hebbact1[w\_ ,preST\_ ,spiketimes\_List]:=$ 
         $\sum_{i=1}^{Length[spiketimes]} (1-w) a_{plus} Exp[preST-spiketimes[[i]]]$ 
```

```
In[4]:=  $Hebbact2[w\_ ,postST\_ ,spiketimes\_List]:=$ 
         $\sum_{i=1}^{Length[spiketimes]} w a_{minus} Exp[spiketimes[[i]]-postST]$ 
```

Con base en estas funciones, construimos en Mathematica un programa para simular una red de neuronas  $N$  cuyos pesos se ajustan de acuerdo a los postulados de Hebb arriba descritos. Las Figuras 2.9 y 2.10 muestran las redes neuronales y su potencial de membrana. La diferencia de estas dos figuras es que la primera muestra la interacción de las neuronas que tienen pesos estáticos, y la segunda son neuronas siguiendo el postulado de Hebb, es decir, tienen pesos dinámicos.

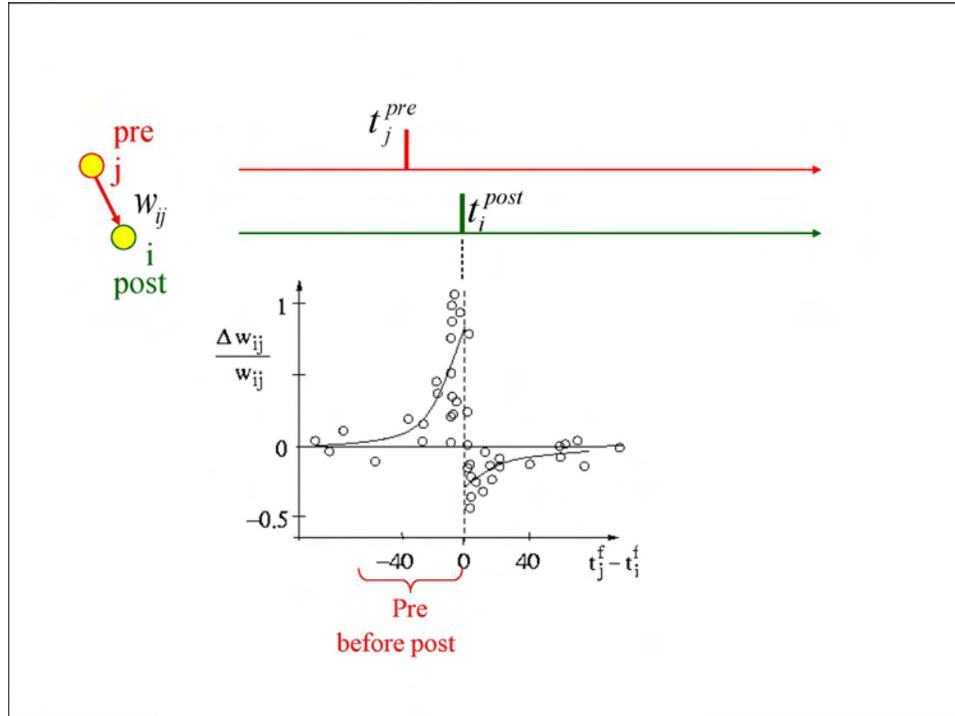


Figura 2.8: La función de la ventana de aprendizaje. Los cambios sinápticos  $\delta w_{ij}$  ocurren cuando el disparo presináptico  $t_j^{(f)}$  y la actividad postsináptica  $t_i^{(f)}$  ocurren lo suficientemente cerca una de otra. Los cambios de los pesos (círculos en la gráfica) son medidos experimentalmente en función de la diferencia  $t_j^{(f)} - t_i^{(f)}$ , y están superpuestos en la ventana de aprendizaje (línea continua). Se realiza un cambio de pesos positivo cuando el disparo presináptico sucede antes del disparo postsináptico; y si el tiempo es inverso, los cambios sinápticos son negativos. Los datos son tomados de los experimentos en el artículo de Bi y Poo [13]

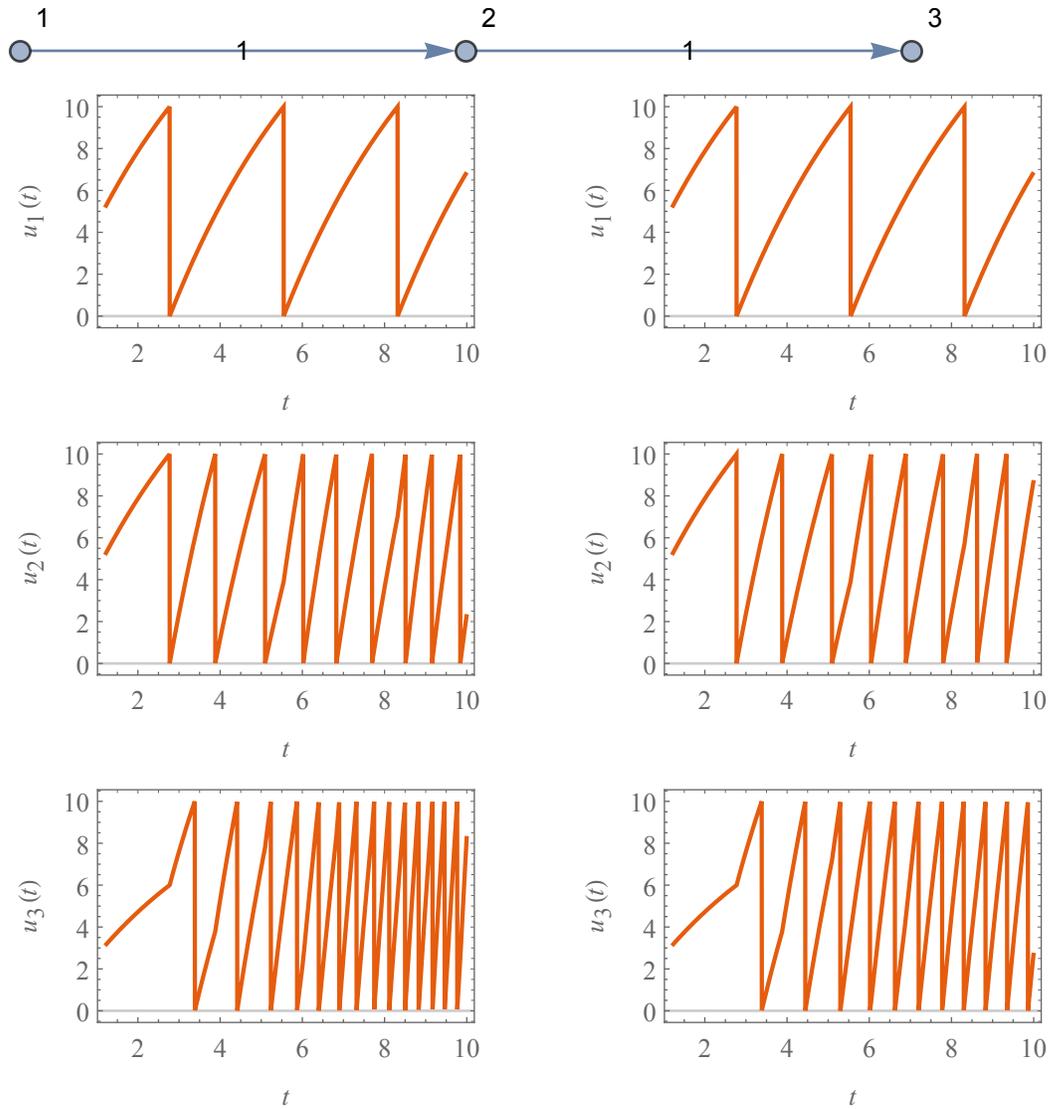


Figura 2.9: Potencial de membrana a través del tiempo de la red de tres neuronas

Figura 2.10: Comportamiento del potencial de membrana de la misma red de tres neuronas siguiendo el aprendizaje de Hebb en 10 unidades de tiempo

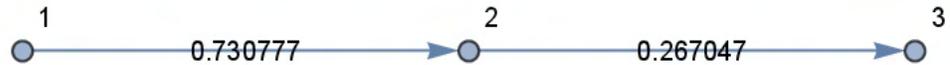


Figura 2.11: Estado de los pesos de la red de neuronas al finalizar esas 10 unidades de tiempo.

El efecto de la adaptación es disminuir los pesos de la interconexión de las neuronas 2 y 3, disminuyendo su tasa de disparo. Para que sea mas evidente, comparamos los tiempos de disparo de cada una de las neuronas (Fig 2.12 y 2.13). Debajo de cada gráfica, los tres números que están entre llaves, son las tasas de disparo de las neuronas 1, 2 y 3, respectivamente. Las tasas de disparo son distintas (*excepto la tasa de disparo de la neurona 1, pues esta recibe una corriente externa constante, sin tener ninguna influencia de la red, pues no tiene neuronas presinápticas conectadas a ella*). Como ya habrán notado, el aprendizaje Hebbiano no depende de la tasa de disparo deseada, pero de todos modos estudiaremos el algoritmo dada su importancia en la neurociencia, y veremos a que nos lleva. El programa de Wolfram nos facilitará el estudio de las tasas de disparo de la red con este aprendizaje. En el Capítulo siguiente estudiaremos la plausibilidad de que la regla de aprendizaje Hebbiana pueda ser la responsable de la homeostasis observada en los experimentos del hipocampo del ratón.

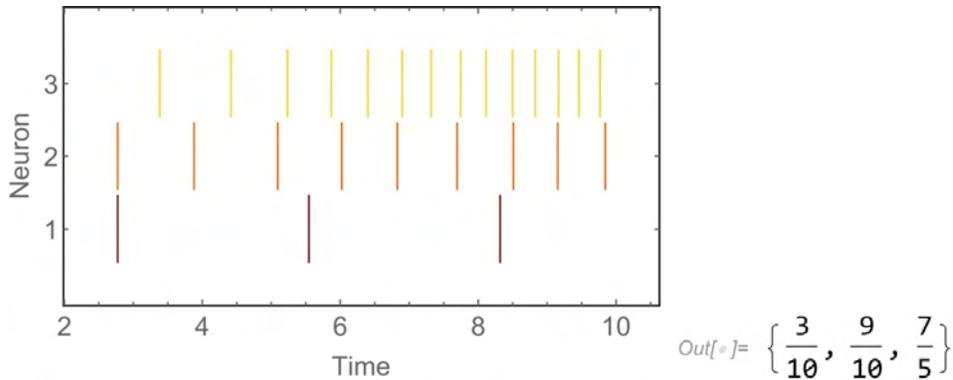


Figura 2.12: Tiempos y tasas de disparo de red sin aprendizaje de Hebb

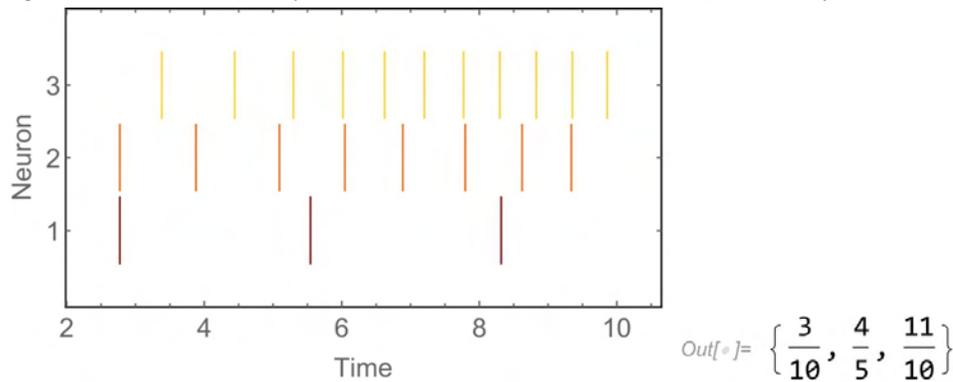


Figura 2.13: Tiempos y tasas de disparo de red con aprendizaje de Hebb

### 2.3.2. Modelo artificial de aprendizaje: método del descenso más rápido

En esta sección discutimos un algoritmo para ajustar los pesos entre neuronas utilizando un algoritmo comúnmente utilizado para entrenar redes neuronales artificiales. En el caso del aprendizaje de Hebbian el objetivo era facilitar la comunicación entre neuronas, sin saber exactamente cual serán las propiedades de la red resultante. Aquí exploraremos el uso de un algoritmo en donde los pesos son ajustados de tal forma que buscan explícitamente llegar a una tasa de disparo poblacional deseada (e.g., la tasa de disparo previa a la perturbación).

Para describir el algoritmo comenzamos definiendo la siguiente “función

de costo”:

$$c(W, \langle f \rangle_d) = \frac{1}{2}(\langle f \rangle_d - \langle f \rangle)^2,$$

donde  $\langle f \rangle$  es la tasa de disparo poblacional actual de las neuronas (*en el código de Mathematica usamos PFR por sus siglas en inglés para esta variable*). La variable  $\langle f \rangle_d$  es la tasa de disparo deseada: es la tasa que queremos alcanzar (*en el código se usó la variable desiredFR*),  $w = \text{vec}(W)$  es un vector que contiene todos los pesos actuales de la red.

Para alcanzar la tasa de disparo deseado, la meta es ajustar los pesos de la red  $W$  para minimizar la función de costo. Si el costo es 0, entonces la tasa de disparo poblacional es igual a la deseada, y habremos alcanzado la meta.

La función de costo, además de depender de  $\langle f \rangle_d$ , depende también, implícitamente, de los pesos de las conexiones, pues estos definen las tasas de disparo.

Para minimizar la función de costo, utilizaremos el método de “descender el gradiente” (gradient descent), el cual es muy popular para entrenar redes neuronales artificiales [6]. En este algoritmo, el valor siguiente de los pesos de la red  $w_{k+1}$  es calculado a partir de los pesos actuales  $w_k$  utilizando la siguiente fórmula.

$$w_{k+1} = w_k - \eta \nabla c(w_k, \langle f \rangle_d) \quad (2.6)$$

Aquí  $\nabla c$  es el gradiente de la función de costo con respecto a los pesos  $w$ , es decir, se calcula el costo  $c$  cuando se tienen los pesos  $w$ , y se calcula el costo  $c_t$  cuando se tienen los pesos  $w + t$ , y  $\nabla c := \frac{c - c_t}{t}$ . La constante  $\eta > 0$  es la “tasa de aprendizaje”. En la sección de Resultados podrá apreciar las gráficas.

## 2.4. Algoritmo utilizando la tasa de disparo local

Por último, analizaremos una nueva propuesta de adaptación de pesos de las sinapsis. Este algoritmo tiene base en la *tasa local de disparos* de las neuronas vecinas.

Como se ha descubierto en [3], la meta de la red neuronal del hipocampo es mantener constante su tasa de disparo poblacional; se sugiere que esto es

causado por algún tipo de autorregulación. Posiblemente, dicha autorregulación se alcanza a través de la manipulación adecuada de los pesos de las sinapsis. Afirmamos que una neurona no sabe cuál es la tasa de disparo global en la red, (la cual se mantiene constante), no obstante, para lograr la meta es condición suficiente mantener constante su tasa de disparo individual. Pero lograrlo sería una tarea muy compleja que requeriría demasiada energía y tiempo, pues la tasa individual de una neurona es inestable por ser parte de una red interconectada, donde el cambio de la tasa de disparo de una neurona afecta a todas las demás. Nosotros creemos que una neurona intenta mantener su tasa de disparo local. A continuación definimos este concepto.

Sea  $(G, E)$  un grafo de Erdos-Renyi, donde  $G$  es el conjunto de nodos y  $E$  es el conjunto de conexiones entre dichos nodos.

Definimos  $A_i = \{j \in G | j \rightarrow i \in E\}$  el conjunto de los nodos antecesores directos de  $i$ .

La tasa de disparo local de  $i$  se define como  $\langle f_{A_i} \rangle_T = \frac{\sum_{j \in A_i} \langle f_j \rangle_T + \langle f_i \rangle_T}{|A_i| + 1}$ , con  $T$  la ventana de tiempo observada. Esta es el promedio de las tasas de disparo de las neuronas antecesoras directas de  $i$  junto con la tasa de disparo de  $i$  misma. La motivación es la suposición que la neurona “siente” el número de disparos que recibe. Con esta información, la neurona  $i$  es capaz de inducir un proceso bioquímico adecuado que cambia la eficacia de la conexión con las neuronas presinápticas haciendo que la tasa de disparo local se mantenga casi constante. Así, guardando la información de la tasa de disparo local inicial, controla los pesos de tal forma para mantener la tasa de disparo local cerca de la deseada, es decir, la tasa de disparo local inicial, que denotaremos por  $\langle f_L \rangle_d$ .

Para mantener el modelo lo más simple posible, aumentamos el peso de la sinapsis  $w_{ij}$  proporcionalmente a la diferencia entre  $\langle f_L \rangle_d$  y  $\langle f_L \rangle$ ; cada peso lo aumentamos por  $\eta(\langle f_L \rangle_d - \langle f_L \rangle)$ . Claramente, si la tasa de disparo local actual es mayor que la deseada habría que disminuirla. Pero justo  $(\langle f_{A_i} \rangle_d - \langle f_{A_i} \rangle) < 0 \Rightarrow w_{ji}(n+1) < w_{ji}(n)$ , donde el argumento  $n$  es la cantidad de ciclos de tiempo de longitud  $T$  transcurridos. Por ende los impulsos recibidos por la neurona  $j$  son mas débiles, su tasa de disparo individual disminuye, junto con ella disminuyendo la tasa de disparo local. La disminución de la tasa de disparo de  $j$  se propaga, provocando cambios en la red entera hasta, posiblemente, alcanzar las neuronas antecesoras directas de  $j$ , disminuyendo la tasa de disparo de cada una de ellas. Estos cambios suceden, de forma iterativa, en cada una de las neuronas de la red. En caso que la tasa de disparo local sea menor que la deseada, entonces se intenta aumentarla. Pero,

análogamente,  $(\langle f_{A_i} \rangle_d - \langle f_{A_i} \rangle) > 0 \Rightarrow w_{ji}(n+1) > w_{ji}(n)$  y se genera el mismo proceso.

Matemáticamente, haciendo tender  $T \rightarrow 0$ , el algoritmo se describe como:

$$\forall i \in G, j \in A_i : \dot{w}_{ij} = \eta(\langle f_{A_i} \rangle_d - \langle f_{A_i} \rangle)$$

con  $\eta > 0$  suficientemente chica para tener precisión y evitar oscilaciones, pero suficientemente grande para que la adaptación del algoritmo sea rápido.

Es claro que si  $\langle f_{A_i} \rangle = \langle f_{A_i} \rangle_d \Rightarrow \langle \hat{f} \rangle = \langle \hat{f} \rangle_d$ , cumpliendo el propósito.

## Capítulo 3

# Resultados: aplicación de los modelos construidos para explicar las observaciones experimentales

En esta sección presentamos el uso del modelo y reglas de aprendizaje construidos en el Capítulo anterior para explicar las observaciones experimentales de Slomowitz et al. [3]. Comenzamos analizando el comportamiento de redes sencillas, para después incrementar su complejidad.

### 3.1. Resultados para redes simples

Para realizar un análisis riguroso empecemos por entender una red neuronal carente de cualquier regla de aprendizaje. Comenzaremos estudiando una red de dos nodos, como se ilustra en la Figura de abajo:



En esta red tenemos cinco variables.

- **Corriente externa** de la neurona 1 ( $I_1$ ). Esta es una variable independiente.

- **Corriente externa** de la neurona 2 ( $I_2$ ). Esta es la segunda variable independiente.
- **Peso de la conexión** de la neurona 1 a la neurona 2 ( $w_{21}$ ). Por tener una red de solo una conexión, denotaremos el peso por  $w$ . Esta es, también, una variable independiente.
- **Tasa de disparo durante  $T$  unidades de tiempo de la neurona 1**  $-\langle f_1 \rangle_T$ . Esta es una variable dependiente, que en la red presente solo depende de  $I_1$
- **Tasa de disparo en tiempo  $T$  de la neurona 2**  $-\langle f_2 \rangle_T$ . Esta es otra variable dependiente, y en la red presente depende de las tres variables:  $I_1$ ,  $I_2$  y  $w$ .

Para entender como se comporta esta red simple, la primera pregunta que nos planteamos fue: ¿Cómo depende la tasa de disparo de una neurona de la corriente externa? Encontramos que esta dependencia es casi lineal para valores suficientemente grandes de la corriente de entrada (Ver Fig. 3.1).

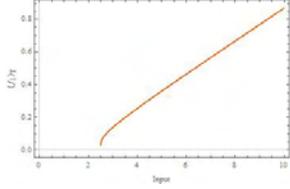


Figura 3.1: Dependencia de la tasa de disparo con respecto a la entrada de corriente

Esta observación puede probarse analíticamente con el modelo *Integrate-and-Fire*. Para esto, escribimos la ecuación de este modelo de nuevo, suponiendo una corriente de entrada constante:

$$\tau u_1'(t) = -u_1(t) + RI \quad (3.1)$$

El primer disparo sucede en  $\hat{t} = \min\{t : u(t) = \gamma\}$ . Ya que la corriente de entrada es constante, la tasa de disparo  $\langle f_1 \rangle$  se puede calcular teniendo solo el tiempo del primer disparo, y equivaldría a  $\langle f_1 \rangle = 1/\hat{t}$ .

Despejando 3.1, obtenemos

$$\tau u_1'(t) + u_1(t) = RI \quad (3.2)$$

Esta ecuación se puede resolver multiplicando por el factor integrante  $\mu = e^{t/\tau}$ . Así,

$$\begin{aligned} \tau u_1'(t) \cdot e^{t/\tau} + u_1(t) \cdot e^{t/\tau} &= RI \cdot e^{t/\tau} \\ \Leftrightarrow (\tau u_1 \cdot e^{t/\tau})' &= RI \cdot e^{t/\tau} \\ \Leftrightarrow \tau u_1 \cdot e^{t/\tau} &= \int RI \cdot e^{t/\tau} dt = RI\tau \cdot e^{t/\tau} + C \end{aligned}$$

donde  $C$  es una constante.

Entonces, la solución de la ecuación 3.1 es

$$u(t) = RI + Be^{-t/\tau} \quad (3.3)$$

donde  $B$  es una constante que depende de las condiciones iniciales. Ya que estamos partiendo del potencial de reposo, la condición inicial es  $u(0) = 0$ . Esto implica que  $B = -RI < 0$  pues  $R, I > 0$

Si la corriente externa es constante, los intervalos de tiempo entre cada disparo neuronal son iguales, entonces la tasa de disparo  $\langle f \rangle_T = \frac{1}{\hat{t}}$ , donde  $\hat{t}$  es el primer tiempo de disparo.

El primer disparo (así como cualquier otro disparo) puede calcularse a través de la siguiente ecuación

$$\begin{aligned} u(\hat{t}) &= RI(1 - e^{\hat{t}/\tau}) = \gamma \\ \Leftrightarrow 1 - \frac{\gamma}{RI} &= e^{-\hat{t}/\tau} \\ \Leftrightarrow \hat{t} &= -\tau \ln \left( 1 - \frac{\gamma}{RI} \right) \\ \langle f \rangle_T &= -\frac{1}{\tau \ln \left( 1 - \frac{\gamma}{RI} \right)} = g(I) \end{aligned} \quad (3.4)$$

donde  $\gamma > 0$  es el umbral de disparo. Si  $I > \frac{\gamma}{R}$ , la tasa de disparo es mayor que 0.

Ya que en nuestro caso estudiado, sustentado por observaciones empíricas, utilizamos  $\gamma = 10$  y  $R = \tau = 4$ , sustituyendo esto en (1.4) obtenemos que  $\langle f \rangle_T = -\frac{1}{4 \ln(1 - \frac{10}{4I})}$ .

Entre más grande la corriente externa, mayor es la tasa de disparo, como era de esperarse. Expandiendo la función 3.4 en su serie de Taylor, encontramos que los términos de orden superior son muy pequeños, explicando el comportamiento lineal de la Fig. 3.1.

Para ver esto, repetimos la ecuación 3.4 a continuación:

$$\langle f \rangle_T = -\frac{1}{\tau \ln \left(1 - \frac{\gamma}{RI}\right)} = g(I)$$

La serie de Taylor es de la forma

$$\sum_{n=0}^{\infty} \frac{f^{(n)}(I_0)}{n!} (I - I_0)^n$$

Ya que calcular las derivadas resulta ser complicado, hacemos un pequeño truco: sustituimos  $x = 1 - \frac{\gamma}{RI}$ , obteniendo como resultado

$$\langle f \rangle_T = -\frac{1}{\tau \ln x} = h(x).$$

Recordando la regla de la cadena tenemos:

$$\begin{aligned} \frac{dg}{dI} &= \frac{dh}{dI} = \frac{dh}{dx} \frac{dx}{dI} \\ \frac{d^2g}{dI^2} &= \frac{d}{dI} \frac{dh}{dI} = \frac{d}{dI} \left( \frac{dh}{dx} \frac{dx}{dI} \right) = \frac{d^2h}{dx dI} \frac{dx}{dI} + \frac{dh}{dx} \frac{d^2x}{dI^2} = \frac{d}{dx} \frac{dh}{dx} \left( \frac{dx}{dI} \right)^2 + \frac{dh}{dx} \frac{d^2x}{dI^2} \end{aligned}$$

Y así sucesivamente, la fórmula general de la regla de la cadena para derivadas de orden superior:

$$\frac{d^n h(x(I))}{dI^n} = \sum \frac{n!}{m_1! m_2! \cdots m_n!} \cdot h^{(m_1 + \cdots + m_n)} \cdot \prod_{j=1}^n \left( \frac{\frac{d^j x}{dI^j}}{j!} \right)^{m_j} \quad (3.5)$$

donde la suma es sobre los enteros no negativos  $(m_1, m_2, \dots, m_n)$  tal que  $m_1 + 2 \cdot m_2 + 3 \cdot m_3 + \cdots + n \cdot m_n = n$ .

Esta generalización de la regla de la cadena es llamada Fórmula de Faà di Bruno.

Ahora, sustituyendo en la serie de Taylor, obtenemos:

$$\begin{aligned} \sum_{n=0}^{\infty} \frac{d^n h(x_0)}{dI^n} (I - I_0)^n &= \\ \sum_{n=0}^{\infty} \left( \sum \frac{\cancel{n!}}{m_1! m_2! \cdots m_n!} \cdot h^{(m_1 + \cdots + m_n)}(x_0) \cdot \prod_{j=1}^n \left( \frac{\frac{d^j x}{dI^j}}{j!} \right)^{m_j} \right) \frac{(I - I_0)^n}{\cancel{n!}} & \quad (3.6) \end{aligned}$$

Para no tener notación tan engorrosa, denotaremos la suma  $m_1 + \dots + m_n = m$ . Ahora, veamos las derivadas de  $h(x)$ :

$$\begin{aligned} h(x) &= -\frac{1}{\tau} \ln^{-1}(x) \\ h'(x) &= \frac{1}{\tau} x^{-1} \ln^{-2}(x) \\ h''(x) &= -\frac{1}{\tau} (x^{-2} \ln^{-2}(x) + 2x^{-2} \ln^{-3}(x)) \end{aligned}$$

De hecho, en general:

$$\begin{aligned} (x^{-n} \ln^{-m}(x))' &= -nx^{-(n+1)} \ln^{-m}(x) - mx^{-(n+1)} \ln^{-(m+1)}(x) < x^{-n} \ln^{-m}(x) \\ &\Leftrightarrow -nx^{-1} - mx^{-1} \ln^{-1}(x) < 1 \\ &\Leftrightarrow \ln^{-1}(x) > \frac{-x-n}{m} \\ &\Leftrightarrow \ln(x) > -\frac{m}{x+n} \end{aligned}$$

Más aún, podemos notar que  $\exists x_0 > 0$  tal que  $\ln(x) > \frac{m}{x+n} \forall x > x_0$ , pues  $\ln(x)$  es creciente mientras que  $\frac{1}{x}$  es decreciente, por lo tanto,  $|(x^{-n} \ln^{-m})'| < |x^{-n} \ln^{-m}| \forall x > x_0$  y por lo tanto concluimos que si  $n < m$  entonces  $\exists x_0$  tal que  $\forall x > x_0$ ,  $(\ln^{-1}(x))^{(n)} > (\ln^{-1}(x))^{(m)}$ .

Ahora, hagamos el mismo análisis con las derivadas de  $x$  con respecto a  $I$ :

$$\begin{aligned} x &= 1 - \frac{\gamma}{R} I^{-1} \\ x' &= \frac{\gamma}{R} I^{-2} \\ x'' &= -2 \frac{\gamma}{R} I^{-3} \\ x''' &= 3! \frac{\gamma}{R} I^{-4} \\ &\vdots \\ x^{(j)} &= j! \frac{\gamma}{R} (-I)^{-(j+1)} \end{aligned}$$

Sustituyendo dichos análisis en 3.6 obtenemos lo siguiente:

$$\begin{aligned}
& \sum_{n=0}^{\infty} \left( \sum \frac{1}{m_1!m_2! \cdots m_n!} \cdot (\ln^{-1}(x))^{(m)} \Big|_{x_0} \cdot \prod_{j=1}^n \left( \frac{\gamma}{R} (-I_0)^{-(j+1)} \right)^{m_j} \right) (I - I_0)^n = \\
& \sum_{n=0}^{\infty} \left( \sum \frac{1}{m_1!m_2! \cdots m_n!} \cdot (\ln^{-1}(x))^{(m)} \Big|_{x_0} \cdot \left( \frac{\gamma}{R} \right)^m \prod_{j=1}^n (-I_0)^{-jm_j} (-I_0)^{-m_j} \right) (I - I_0)^n = \\
& \sum_{n=0}^{\infty} \left( \sum \frac{1}{m_1!m_2! \cdots m_n!} \cdot (\ln^{-1}(x))^{(m)} \Big|_{x_0} \cdot \left( \frac{\gamma}{R} \right)^m (-I_0)^{-n} (-I_0)^{-m} \right) (I - I_0)^n = \\
& \sum_{n=0}^{\infty} \left( \sum \frac{1}{m_1!m_2! \cdots m_n!} \cdot (\ln^{-1}(x))^{(m)} \Big|_{x_0} \cdot \left( -\frac{\gamma}{I_0 R} \right)^m \right) \frac{(I - I_0)^n}{(-I_0)^n} < \\
& \sum_{n=0}^{\infty} \left( \sum \frac{1}{m_1!m_2! \cdots m_n!} \cdot \ln^{-1}(x) \Big|_{x_0} \cdot \left( -\frac{\gamma}{I_0 R} \right)^m \right) \frac{(I - I_0)^n}{(-I_0)^n} = \\
& \sum_{n=0}^{\infty} \left( \sum \frac{1}{m_1!m_2! \cdots m_n!} \cdot \frac{(-\gamma)^m}{\ln(1 - \frac{\gamma}{I_0 R}) \cdot (I_0 R)^m} \right) \frac{(I - I_0)^n}{(-I_0)^n}
\end{aligned}$$

Persiguiendo simplificar los cálculos al máximo, tomemos  $\gamma = R = 1$  y  $I \in (2, 10]$ , (también son estos valores los que hemos tomado en las simulaciones):

$$\sum_{n=0}^{\infty} \left( \sum \frac{1}{m_1!m_2! \cdots m_n!} \cdot \frac{(-1)^m}{\ln(1 - \frac{1}{I_0}) \cdot (I_0)^m} \right) \frac{(I - I_0)^n}{(-I_0)^n}$$

Primero observemos el comportamiento de  $\ln(1 - \frac{1}{I_0}) \cdot (I_0)^m = \ln((1 - \frac{1}{I_0})^{I_0^m})$ , por la restricción sobre  $I$ ,  $1 - \frac{1}{I_0} < 1 \Rightarrow (1 - \frac{1}{I_0})^{I_0^m} \rightarrow 0 \Rightarrow \ln((1 - \frac{1}{I_0})^{I_0^m}) \rightarrow -\infty$ , por lo que toda la fracción  $\frac{(-1)^m}{\ln((1 - \frac{1}{I_0})^{I_0^m})}$  tiende a 0. La fracción  $\frac{1}{m_1!m_2! \cdots m_n!}$  hace dicho término incluso más chico. Así, los sumandos para  $n \geq 2$  no influye mucho en la suma total.

También analizamos la dependencia entre la tasa de disparo de la neurona 2 y la corriente externa de la neurona 1. Es decir, la pregunta ¿Cómo depende  $\langle f_2 \rangle_T$  de  $I_1$ ?

¡Encontramos que esta relación también tiene una tendencia lineal! (Fig. 3.2) Subrayo la palabra tendencia dos veces, porque no es tal cual lineal, si no que la dependencia es escalonada, pero el comportamiento también es básicamente lineal.

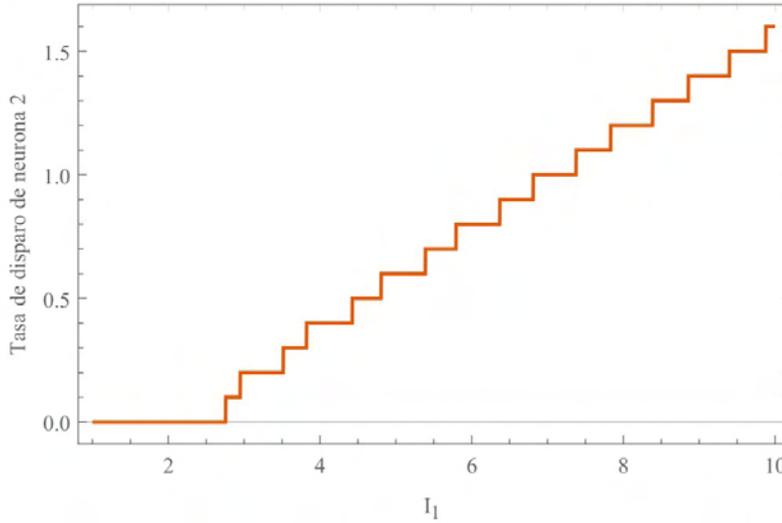


Figura 3.2: Dependencia de la tasa de disparo  $\langle f_2 \rangle_T$  de  $I_1$  con  $w = 1$  y  $I_2 = 2$

El comportamiento es escalonado porque la corriente que entra a la neurona 2 no es continua, pues cada vez que recibe un impulso de la neurona 1, la corriente que percibe da un salto en ese mismo tiempo. Esta relación es casi lineal debido a que la ecuación diferencial del modelo es de primer orden sumado de una constante. Es decir, si  $u_1(t)$  es solución a

$$\tau u_1'(t) = -u_1(t) + RI_1, \quad (3.7)$$

y  $u_2(t)$  es solución a

$$\tau u_2'(t) = -u_2(t) + RI_2, \quad (3.8)$$

entonces

$$\tau u_1'(t) + \tau u_2'(t) = -u_1(t) + RI_1 - u_2(t) + RI_2 = -u_1(t) - u_2(t) + RI_1 + RI_2. \quad (3.9)$$

Por lo tanto,  $u_3 = u_1 + u_2$  también es solución a la ecuación de Integrate-and-Fire.

Ahora veremos cómo cambia la tasa de disparo neuronal si cambiamos el peso entre las dos neuronas, dejando todo lo demás igual. Con ayuda de las simulaciones descubrimos que el comportamiento también es casi lineal (Fig. 3.3). La explicación de esto es de hecho la misma que en el caso anterior, pues recuerde que la corriente externa puede reescribirse como  $I(t) = I_{ext}(t) + \sum w_{ij}PSC(t - \hat{t}_j)$

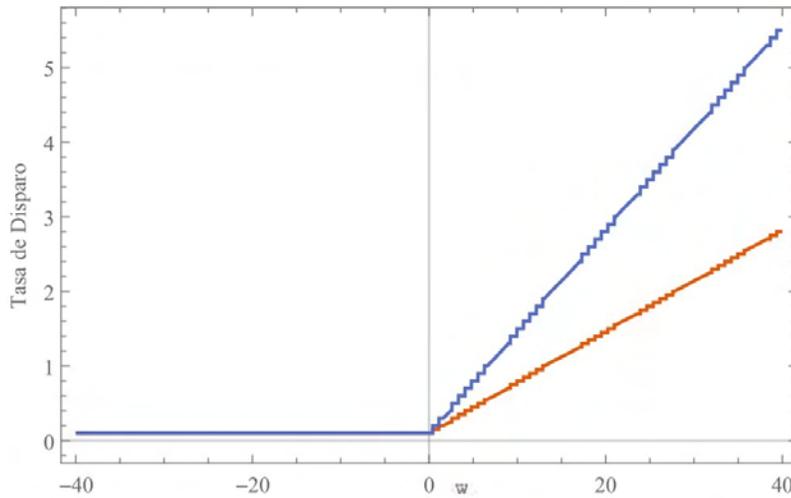


Figura 3.3: Dependencia de las tasas individual (azul) y poblacional (rojo) del peso

Ya que hemos visto que todas las dependencias son lineales, entonces la relación de la tasa de disparo de la neurona 2 con la tasa de disparo de la neurona 1 y el peso entre ellas es lineal (Ver Fig 3.4), y la tasa de disparo poblacional, que es el promedio entre las tasas de disparo de ambas neuronas, también será lineal.

Hemos descubierto dos cosas hasta ahora. Primero, que la relación entre las variables es casi lineal. Segundo, que la relación de la tasa de disparo con el peso de la conexión no es continua, entonces no podemos controlar la red a cualquier tasa de disparo que queramos. Por ende, la meta debe ser ajustar los pesos para acercarse lo mas posible a la tasa de disparo promedio deseada. Sin embargo, existen tasas de disparo promedio para las cuales no existe ningún peso  $w$  que puede alcanzarlos.

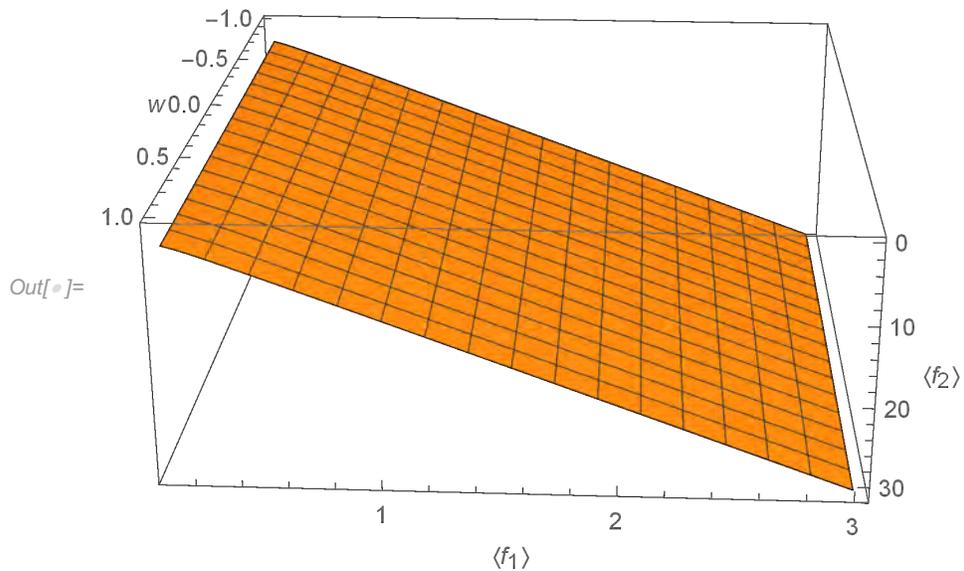


Figura 3.4: Relación entre la tasa de disparo de neurona 2 con la tasa de disparo de neurona 1 y el peso de la conexión. . Aquí  $\langle f_1 \rangle$  representa la tasa de disparo de la neurona 1,  $\langle f_2 \rangle$  representa la tasa de disparo de la neurona 2, y el eje z representa el peso entre las dos neuronas

En las figuras 3.5 a 3.8 se puede observar cómo el cambio del peso influye en el potencial de membrana. Las corrientes externas en las cuatro gráficas se mantienen igual:  $I_1 = 4$ ,  $I_2 = 3$ .

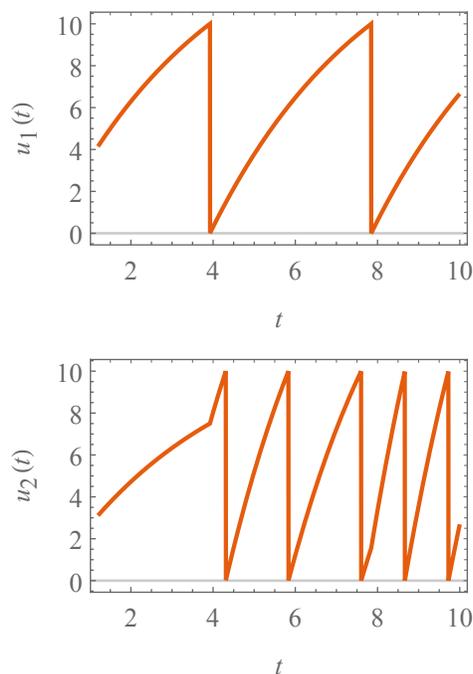


Figura 3.5: Potencial de membrana de dos neuronas con el peso  $w = 1$ .

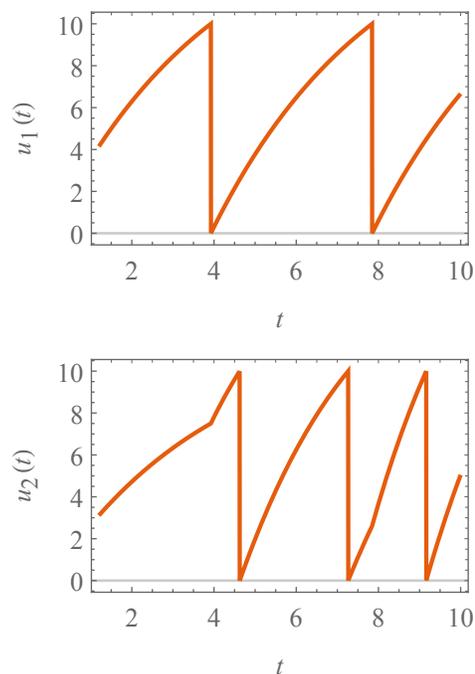


Figura 3.6: Potencial de membrana de dos neuronas con el peso  $w = 0.5$ .

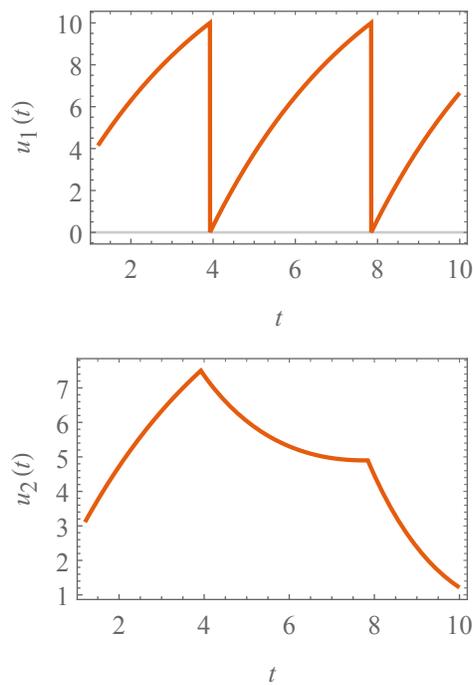


Figura 3.7: Potencial de membrana de dos neuronas con el peso  $w = -0.5$ .

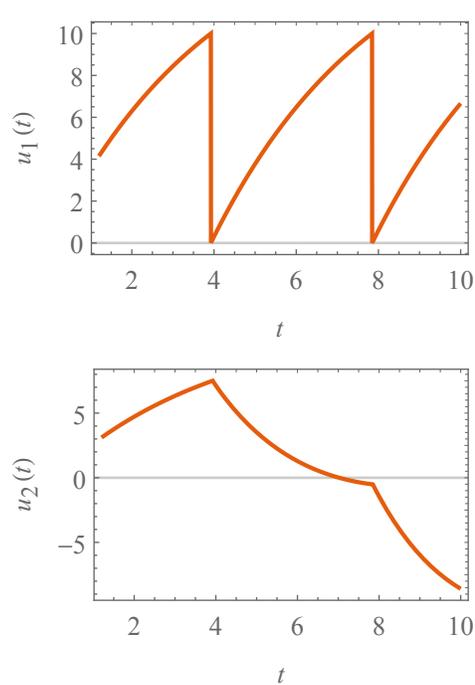


Figura 3.8: Potencial de membrana de dos neuronas con el peso  $w = -1$ .

Nótese que lo único que cambia entre las figuras 3.5-3.8 es el potencial de membrana de la neurona 2, pues ya que  $I_1$  se mantiene constante, el potencial de membrana de la neurona 1 seguirá siendo el mismo para todos los casos.

Ahora, veremos como  $I_1$  influye en toda la red.

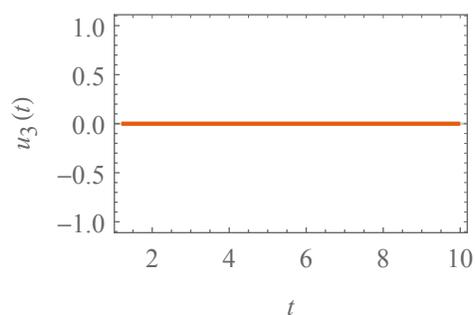
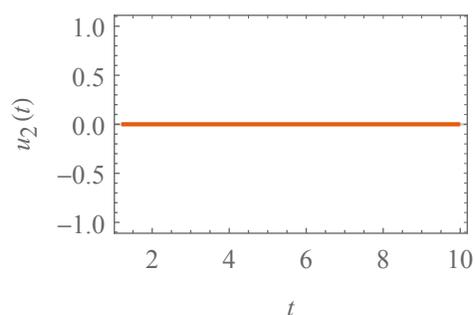
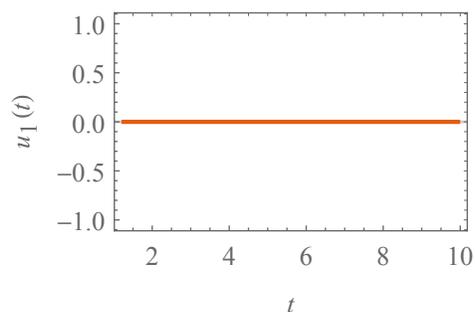
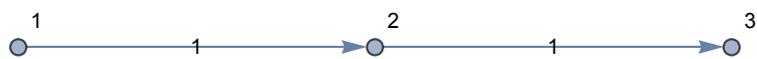


Figura 3.9: El potencial de membrana se mantiene en reposo cuando no hay ningún estímulo que perturbe la red

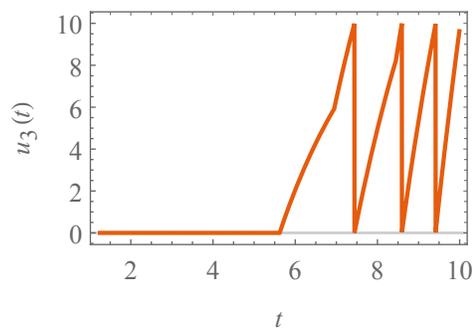
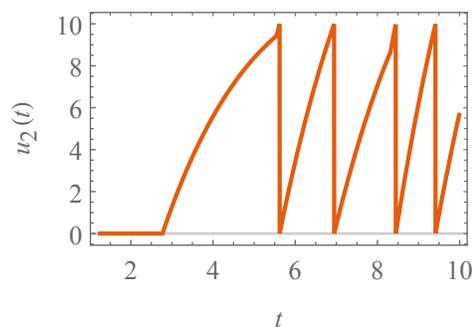
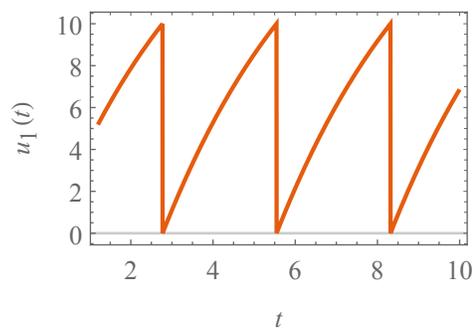


Figura 3.10: El potencial de membrana de las neuronas de la red con corriente de 5 unidades inyectada a la primer neurona)

Figura 3.11: Las gráficas nos muestran la necesidad de inducir corrientes externas, pues sin ella no habrían tasas de disparos

A pesar que las neuronas 2 y 3 no reciben ninguna corriente externa a

la red, sus disparos son mas frecuentes que los de la neurona que sí recibe el estímulo externo. Por ende, una neurona es mas sensible a la corriente externa que reciben las neuronas presinápticas que a la que recibe ella misma directamente.

## 3.2. Resultados para red con aprendizaje de Hebb

En esta sección analizaremos el comportamiento de una red de tres neuronas con aprendizaje de Hebb.

La red cuyos pesos estarán sujetos a cambios es la red de la Figura 3.9. Las gráficas serán analizadas en un lapso de 100 unidades de tiempo, para ver el comportamiento a largo plazo.

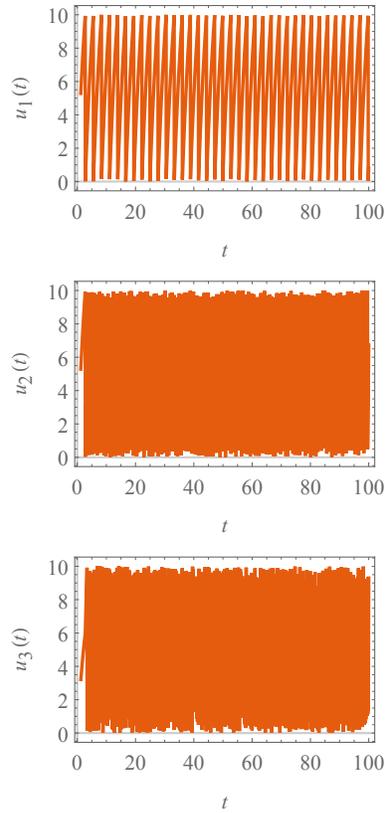


Figura 3.12: Potencial de membrana en red sin reglas de aprendizaje con  $I_1 = 5, I_2 = 5, I_3 = 3$

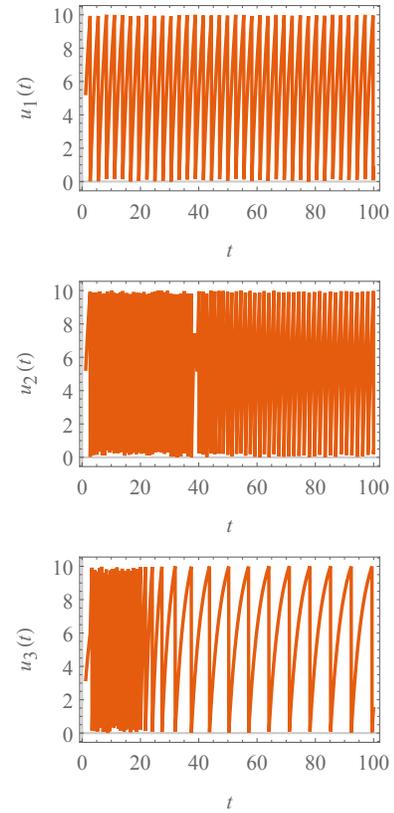


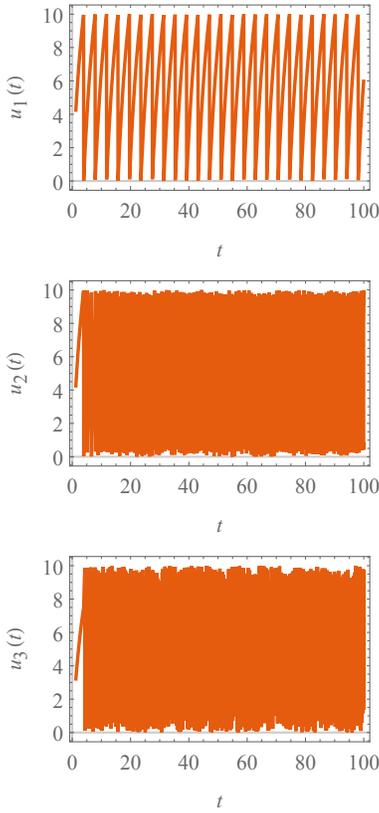
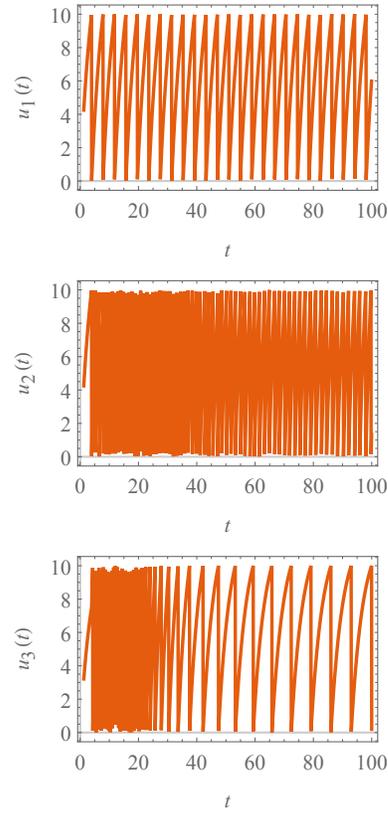
Figura 3.13: Potencial de membrana en red con aprendizaje de Hebb con  $I_1 = 5, I_2 = 5, I_3 = 3$

El disparo poblacional de la red sin aprendizaje de Hebb (con entradas constantes 5,5,3) fue de 2.443, con tasas de disparo individuales de 0.36,1.56, 5.41 correspondientes a las neuronas 1, 2 y 3, respectivamente, mientras que en la red sujeta a aprendizaje de Hebb la tasa de disparo poblacional fue de 0.503 con tasas individuales de 0.36,0.76,0.39.

Para mayor comodidad se visualizará en tabla 3.1.

	Red base	Red de Hebb	Red de Hebb perturbación=-0.5	Red de Hebb perturbación= -1	Red de Hebb perturbación=-2
$\frac{1}{3} \sum_i \langle f_i \rangle_T$	2.443	0.503	0.653	0.28	0.16
$\langle f_1 \rangle_T$	0.36	0.36	0.36	0.36	0.36
$\langle f_2 \rangle_T$	1.56	0.76	0.88	0.36	0.02
$\langle f_3 \rangle_T$	5.41	0.39	0.72	0.12	0.1

Tabla 3.1: Tasas de disparo poblacional y de cada una de las neuronas

Figura 3.14: Potencial de membrana en red sin reglas de aprendizaje con  $I_1 = 4, I_2 = 4, I_4 = 3$ Figura 3.15: Potencial de membrana en red con aprendizaje de Hebb con  $I_1 = 4, I_2 = 4, I_3 = 3$ 

En otra red, ahora con entradas constantes de 4, 4 y 3 unidades por neurona, la tasa de disparo poblacional fue de 1.757, mientras que la regulada

por el aprendizaje de Hebb fue de 0.45. Mas adelante se pueden observar otros dos ejemplos.

	Red base	Red de Hebb	Red de Hebb perturbación=-0.5	Red de Hebb perturbación=-1	Red de Hebb perturbación=-2
$\frac{1}{3} \sum_i \langle f_i \rangle_T$	1.757	0.45	0.56	0.2	0.113
$\langle f_1 \rangle_T$	0.24	0.24	0.24	0.24	0.24
$\langle f_2 \rangle_T$	1.11	0.68	0.68	0.24	0.02
$\langle f_3 \rangle_T$	3.88	0.42	0.78	0.12	0.08

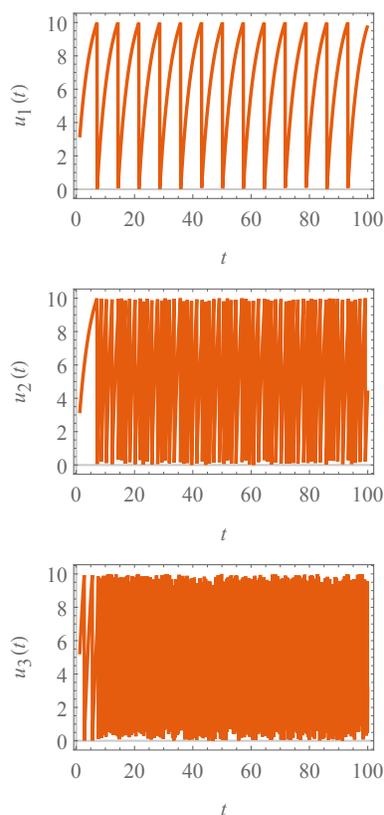


Figura 3.16: Potencial de membrana en red sin reglas de aprendizaje con  $I_1 = 3, I_2 = 3, I_4 = 5$

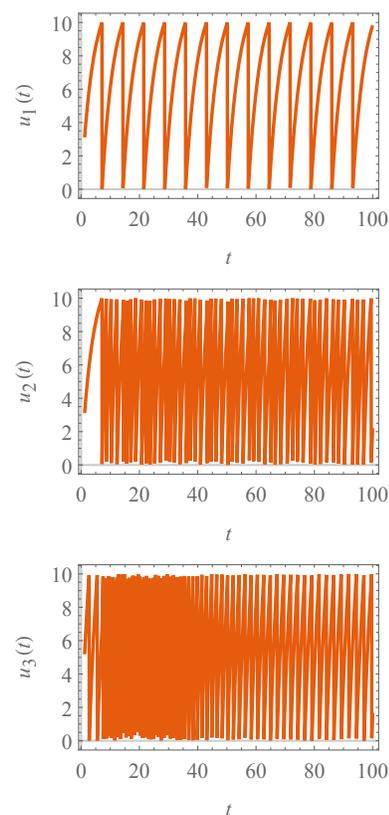


Figura 3.17: Potencial de membrana en red con aprendizaje de Hebb con  $I_1 = 3, I_2 = 3, I_3 = 5$

	Red base	Red de Hebb	Red de Hebb perturbación=-0.5	Red de Hebb perturbación= -1	Red de Hebb perturbación=-2
$\frac{1}{3} \sum_i \langle f_i \rangle_T$	1.05	0.44	0.433	0.2	0.14
$\langle f_1 \rangle_T$	0.12	0.12	0.12	0.12	0.12
$\langle f_2 \rangle_T$	0.61	0.51	0.36	0.12	0.02
$\langle f_3 \rangle_T$	2.42	0.68	0.82	0.36	0.28

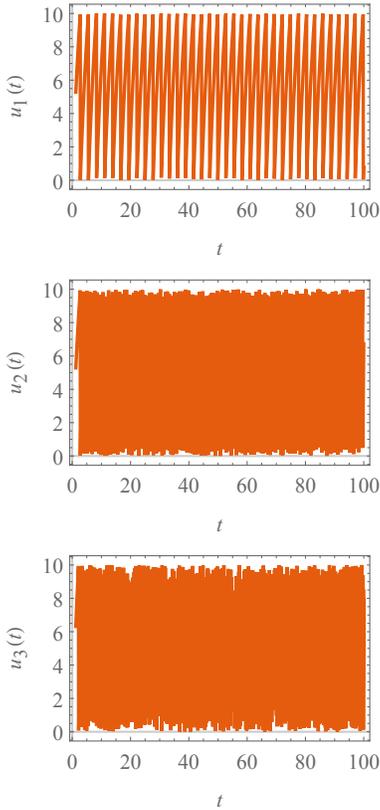


Figura 3.18: Potencial de membrana en red sin reglas de aprendizaje con  $I_1 = 5, I_2 = 5, I_4 = 6$

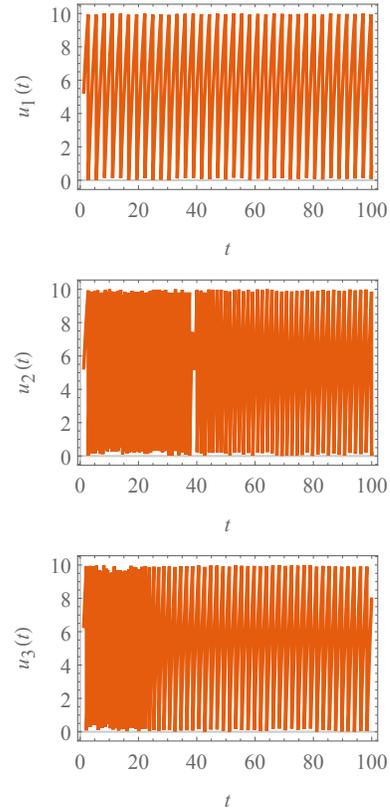


Figura 3.19: Potencial de membrana en red con aprendizaje de Hebb con  $I_1 = 5, I_2 = 5, I_3 = 6$

	Red base	Red de Hebb	Red de Hebb perturbación=-0.5	Red de Hebb perturbación= -1	Red de Hebb perturbación=-2
$\frac{1}{3} \sum_i \langle f_i \rangle_T$	2.543	0.607	0.753	0.393	0.253
$\langle f_1 \rangle_T$	0.36	0.36	0.36	0.36	0.36
$\langle f_2 \rangle_T$	1.56	0.76	0.88	0.36	0.02
$\langle f_3 \rangle_T$	5.71	0.7	1.02	0.46	0.38

A pesar de que la red no regresa a la misma tasa de disparo poblacional inicial al realizar cambios en las corrientes externas, vemos que su tasa de disparo poblacional oscila alrededor de 0.5, haciendo la red más robusta, a comparación de la red sin aprendizaje, donde la cantidad de disparos era excesivo, algunas neuronas disparando a más de una, incluso más de dos veces por unidad de tiempo. Desglosemos este análisis. La tasa de disparo poblacional de las redes con aprendizaje de Hebbian no se diferencian mucho entre cada uno de los cuatro ejemplos graficados, mientras que en las redes sin aprendizaje hay un comportamiento mucho más variado (Fig. 3.13-3.19). Es decir, la regla de aprendizaje permite estabilizar las tasas de disparo. Además, a diferencia de la red con pesos constantes, las redes reguladas por el postulado de Hebb no tienen una tasa de disparo poblacional mayor que uno. Este es un resultado más realista en individuos sanos, pues no queremos que haya saturación de disparos. Una saturación de disparos indicarán una anomalía en el cerebro. Podría representar convulsiones u otra enfermedad neurológica, en la cual el electroencefalograma muestra gran actividad neuronal. El análisis sugiere que, durante anomalías neurológicas, el aprendizaje en las redes neuronales no es hecho de forma correcta.

En conclusión, el aprendizaje Hebbiano por si solo no es capaz de explicar el comportamiento descrito en el experimento de Slomowitz et al., sin embargo, muestra una característica de interés: mantiene la tasa de disparo poblacional alrededor de 0.5, y las neuronas individuales no disparan a más de una vez por unidad de tiempo. Este resultado es muy interesante, pues la meta del modelo propuesto por Hebb era ilustrar como cambia la eficacia de la conexión, sin tomar en cuenta el fenómeno de estabilidad global de la red y, como ya lo demostramos a través de las simulaciones, la red preserva dicha propiedad.

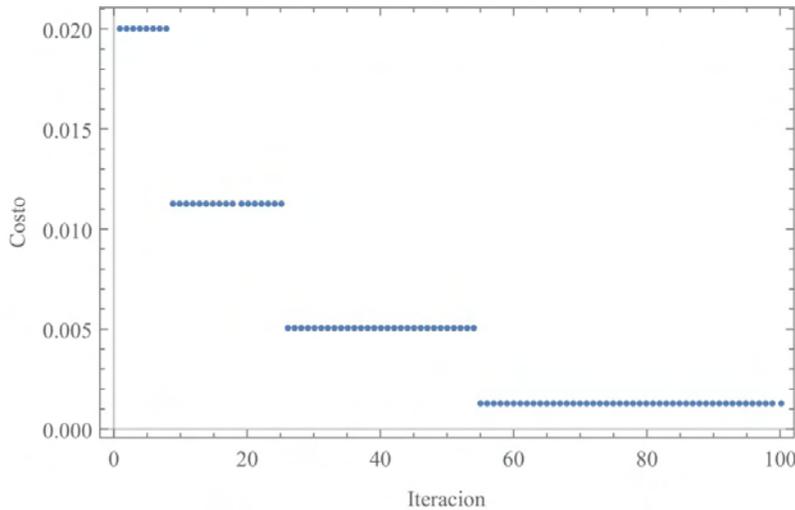


Figura 3.20: Función de costo de un grafo de dos neuronas con peso 1 entre ellas

### 3.3. Resultados usando el algoritmo del gradiente

Este algoritmo es artificial, y es diseñado para cumplir la propiedad descubierta en el artículo de Slomowitz et. al.

La base del algoritmo es definir una función de costo, lo cual hemos hecho en el capítulo 2.3.2.

Mostramos el valor de la función de costo al aplicar el algoritmo de aprendizaje de la ecuación 2.6 para dos redes: una red de dos neuronas (Fig. 3.20) y otra de tres neuronas (Fig. 3.21). En cada iteración el costo disminuye, es decir, la tasa de disparo poblacional cada vez se acerca a la tasa de disparo deseada. Esto implica que la tasa de disparo poblacional se acerca a la tasa de disparo poblacional deseada (Ver Fig. 3.22 y 3.23).

Aunque este algoritmo tiene éxito en regresar la tasa de disparo al nivel deseado, es muy poco plausible a un nivel biológico. En particular, el algoritmo requiere dos condiciones muy poco plausibles. Primero, afirmaríamos que cada neurona en la red puede calcular la tasa de disparo poblacional. Pero, ¿cómo podría una neurona saber la tasa de disparo de otra neurona a 5, 10 o 30 nodos de distancia? Lo segundo que estaríamos presumiendo es que la red calcula de alguna forma la función de costo para después cambiar el peso

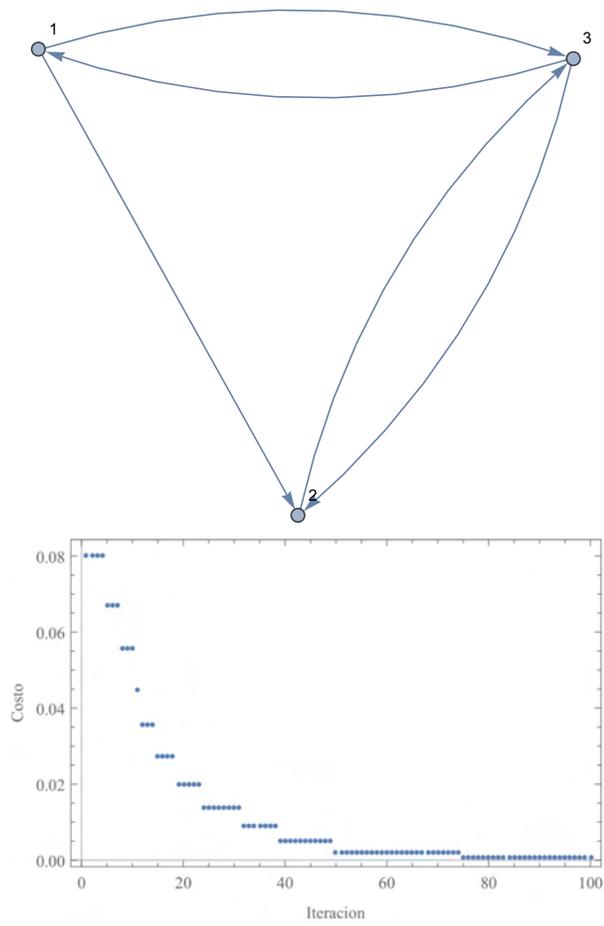


Figura 3.21: Función de costo para grafo de tres nodos de Erdos-Renyi.

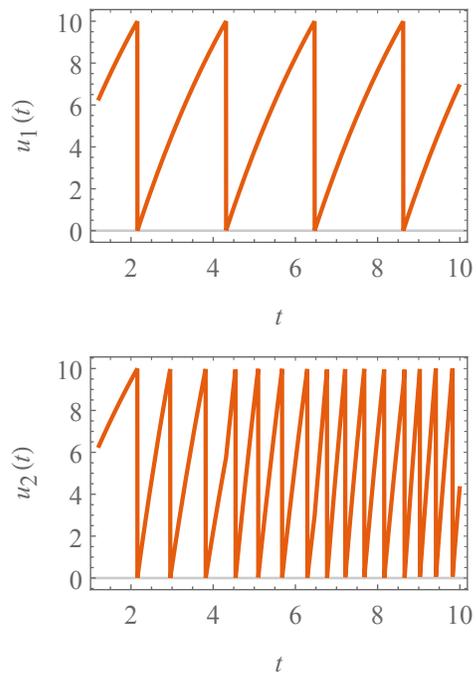


Figura 3.22: Potencial de membrana de ambas neuronas después de la aplicación del algoritmo del gradiente más empinado cuando la tasa de disparo deseada es 1. La tasa de disparo poblacional resultante fue de  $\frac{19}{20}$

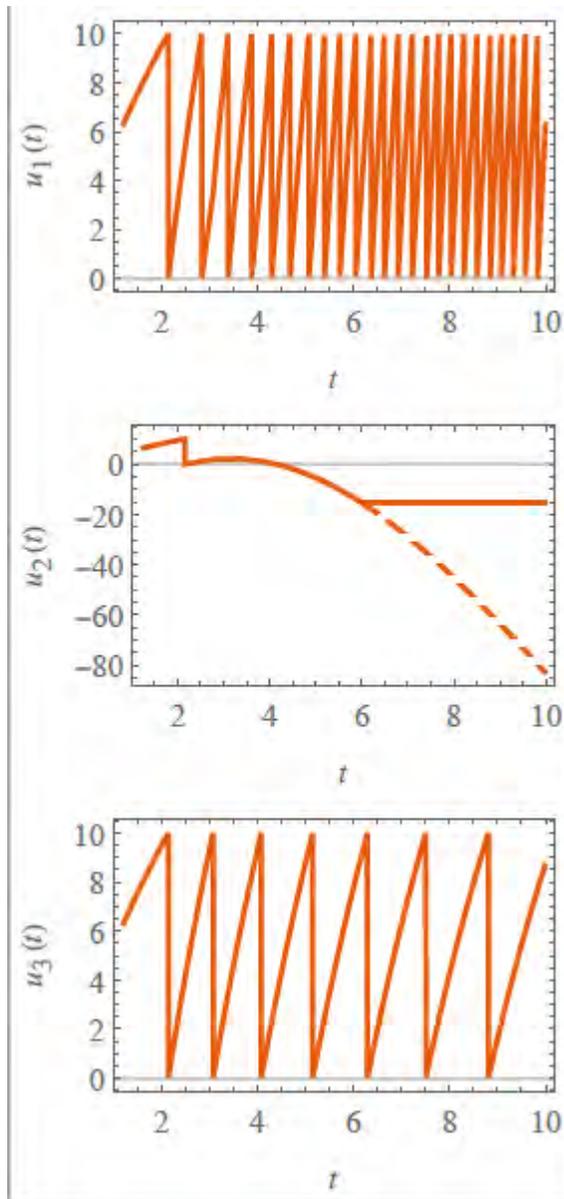


Figura 3.23: Potencial de membrana resultante después de la aplicación del algoritmo del gradiente más empinado en una red de tres neuronas, cuando la tasa de disparo deseada es  $\frac{19}{20}$ . El potencial de membrana de una neurona no puede ser menor a -15 mV por eso en la segunda gráfica vemos que el potencial de membrana se mantiene en -15 mV. Sin esa restricción el potencial de membrana bajaría indefinidamente

de las conexiones.

En conclusión, el segundo algoritmo es capaz de reproducir el comportamiento observado en los resultados experimentales de Slomowitz. Sin embargo, es muy poco plausible que pueda ser implementado en forma biológica.

### 3.4. Resultados utilizando la tasa de disparo local

Comenzaremos analizando el caso más simple de una red con dos neuronas. En la figura 3.24 vemos que la tasa de disparo poblacional se acerca exactamente a la deseada. Esto no es coincidencia, pues la tasa de disparo local equivale a la global, haciendo evidente la razón por la cual se alcanza la tasa global deseada.

Para generalizar, si tenemos una gráfica completamente conectada es claro que siempre se llegará a la tasa de disparo poblacional deseada, pues esta equivaldrá a la local. Esto sugiere que este algoritmo puede ser exitoso para redes densamente conectadas, como ocurre en redes neuronales. Por ejemplo, para una red de tres nodos este algoritmo también tiene éxito en restaurar la tasa de disparo.

El siguiente paso es analizar redes con una mayor cantidad de nodos. Creamos una red de Erdős-Renyi con 10 nodos. En la figura 3.25 vemos que el algoritmo acerca la tasa de disparo a la tasa deseada, con oscilaciones cada vez más pequeñas.

Sin embargo, encontramos que este comportamiento no ocurre para cualquier red, como por ejemplo en la red de la Figura 3.26. Esto pasa porque algunas tasas locales alcanzaron la deseable, mientras que otras se siguen adaptando. Si algunas neuronas tienen que disminuir su tasa individual de disparo para con eso disminuir la tasa global de disparo, estas neuronas necesitan disminuir su tasa individual actualizando sus pesos. Pero una neurona no puede tener una tasa de disparo menor que 0, y los pesos son inhibitorios a tal grado que ya no dispara, por lo que, por mas que disminuyamos los pesos, no puede continuar disminuyendo la tasa de disparo local. Este razonamiento implica que, en general, la tasa promedio de disparos de una red de neuronas no puede tomar cualquier valor. Por tanto, este último algoritmo de aprendizaje propuesto no siempre funciona, en particular cuando la tasa de disparo local es mayor a la deseada, pues la tasa de disparo individual es limitada por

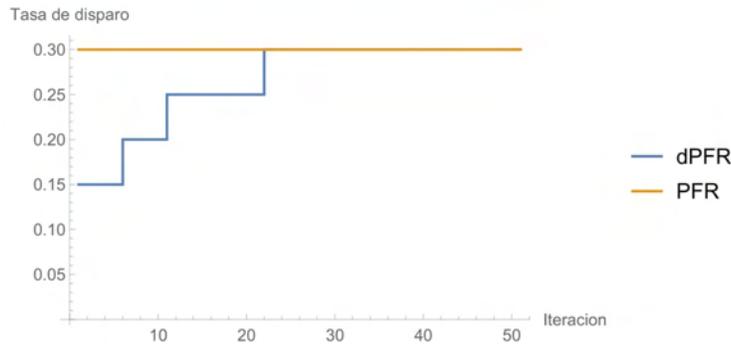


Figura 3.24: Vemos que el algoritmo que usa la tasa de disparo local hace que la tasa de disparo global actual se acerque a la deseada en una red de dos neuronas

abajo, evitando disminuir la local. Sin embargo siempre podemos aumentar los pesos de tal forma que aumente la tasa de disparo individual a cualquier cantidad propuesta, así aumentando la tasa de disparo local.

Por tanto, este último algoritmo es capaz de reajustar la tasa de disparo a un valor deseado siempre y cuando este valor pueda ser alcanzado a través del ajuste de los pesos. Es decir, siempre y cuando sea biológicamente factible. El algoritmo utiliza solamente información local, por lo que hace plausible nuestra hipótesis de que el comportamiento experimental observado por Slowowitz et al. puede explicarse a través de un mecanismo de retroalimentación muy sencillo en el que cada neurona solo utiliza información local (i.e., de sus vecinos).

Para validar esta última observación, aplicamos el algoritmo propuesto a un conjunto de redes aleatorias con entre 5 y 20 neuronas, y con distintas probabilidades de interconexión (Fig. 3.27). Las probabilidades de conexión fueron elegidas de tal forma que cada neurona tiene aproximadamente  $n \leq 3$  conexiones. Los resultados obtenidos muestran que el algoritmo es capaz de conducir a la tasa de disparo promedio de la red a un error  $\leq 0.008$  en este conjunto de redes. Además, la efectividad del algoritmo es muy heterogénea: dado un número de nodos, existe cierta conectividad de la red (determinada por la probabilidad de conexión) para la cual el error es mínimo. Este último resultado soporta el hecho de que, para ciertas redes, es más fácil ajustar su tasa promedio de disparo.

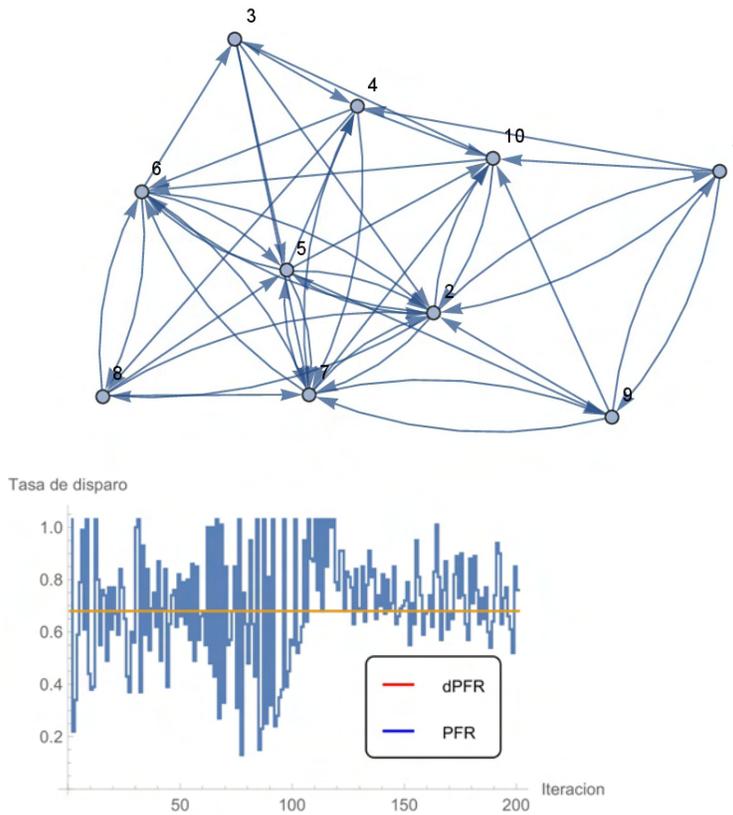


Figura 3.25: El algoritmo acerca, poco a poco, la tasa de disparo poblacional a la deseada en una red de diez neuronas como la que se muestra arriba. Las oscilaciones alrededor de la tasa de disparo deseada disminuyen conforme pasan las iteraciones

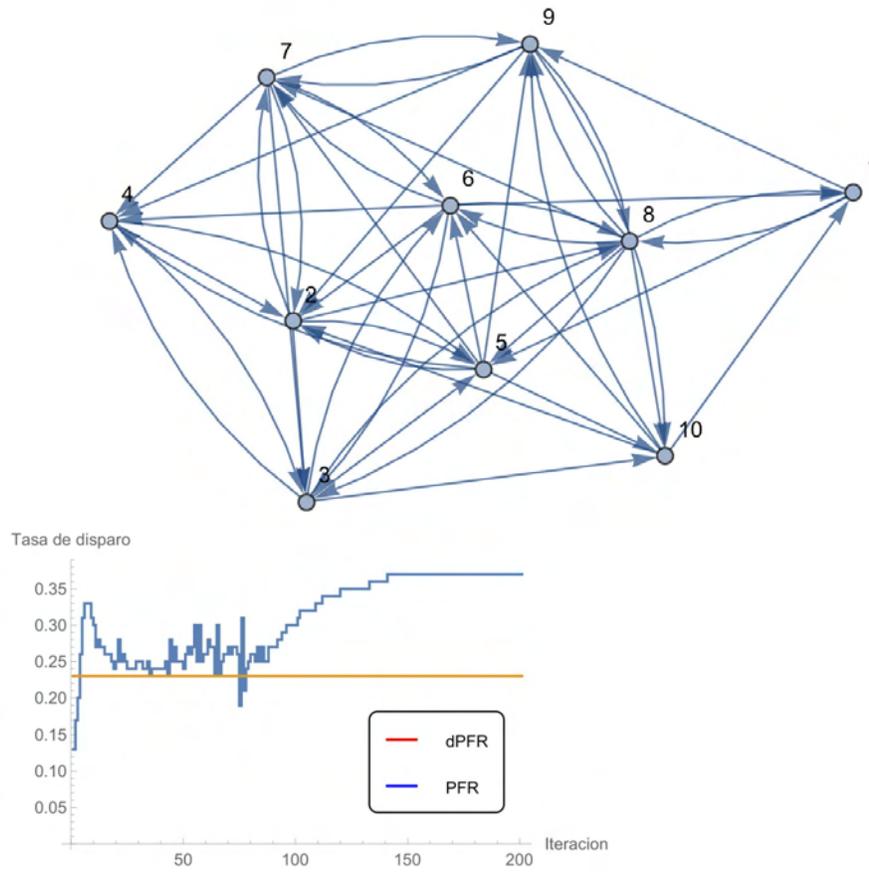


Figura 3.26: El algoritmo no funciona correctamente para todos los grafos aleatorios

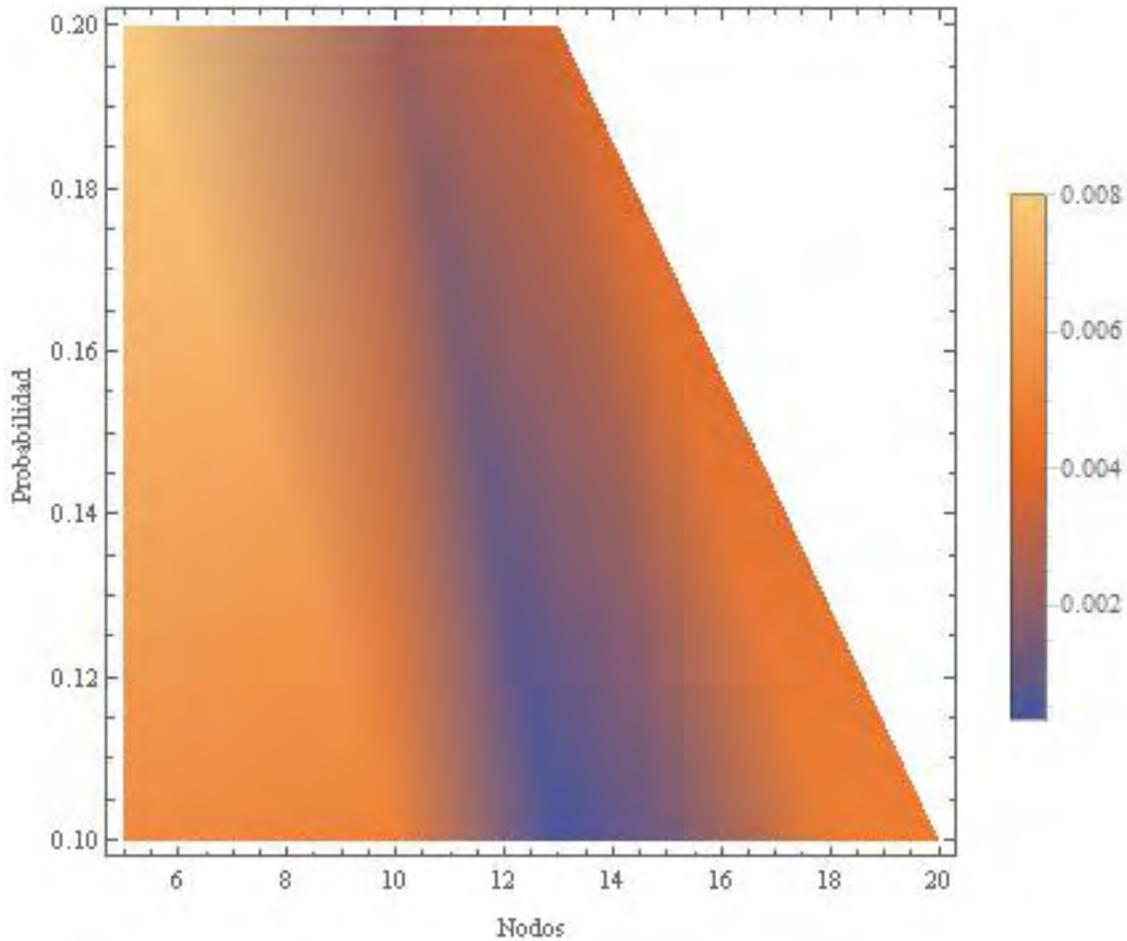


Figura 3.27: Gráfica de densidad del error para redes entre 5 y 20 nodos con probabilidad de interconexión de 0.1 a 0.2. El color representa el error de la tasa de disparo después de aplicar el algoritmo. No se pudieron calcular valores para redes de 20 neuronas con probabilidad de interconexión mayor a 0.1, por lo que no se pudo hacer interpolación, resultando en una esquina blanca.



# Capítulo 4

## Conclusiones

Las redes neuronales son tan complejas que no se puede proponer un modelo matemático simple que describa exactamente su comportamiento. Los modelos estudiados describen algunas propiedades a costa de otras. Por lo tanto, todos los modelos propuestos tienen sus ventajas y desventajas.

Se notó que el comportamiento de una red que sigue el modelo *Integrate-and-Fire* es sencillo en el sentido que las relaciones entre variables tienen dependencia casi lineal en ciertas regiones de operación. Sin embargo, a pesar de esta sencillez, este modelo de neurona dentro de una red es muy complejo por la cantidad de variables que tiene. Con cada neurona nueva aparecen al menos dos variables nuevas (tasa de disparo y los pesos formados con otras neuronas), haciendo el crecimiento de variables muy rápido.

Encontramos que el modelo de aprendizaje Hebbiano aporta estabilidad, y, a pesar de que quizá poblacionalmente no llega a la condición deseada (*al menos no en el plazo de tiempo observado*), sí se acerca bastante. Es decir, minimiza las variaciones con respecto a los estímulos recibidos.

El algoritmo del gradiente nos lleva exactamente a la tasa de disparo que deseamos, sin embargo la desventaja del algoritmo es que la red neuronal necesita hacer todos sus cálculos para cambiar sus parámetros de las sinapsis y esto no es biológicamente plausible, puesto que en el proceso biológico los cambios de la efectividad sináptica se tienen que efectuar al momento de la actividad neuronal. Además requiere que cada neurona tenga información de la tasa de disparo de todas las otras neuronas en la red, lo cual es muy poco factible desde una perspectiva biológica.

Por último, propusimos otro algoritmo de aprendizaje, biológicamente plausible, que cumple la propiedad encontrada de las redes neuronales descri-

ta en [3] (bajo ciertas condiciones especiales de la red subyacente). El modelo utiliza únicamente la tasa de disparo local y es biológicamente plausible pues cada neurona requiere solo información de las neuronas presinápticas, pues “siente” la cantidad de disparos que recibe desde ellas. El algoritmo presume que una neurona no puede distinguir desde dónde llegó exactamente el disparo, por lo que el cambio en el peso de la conexión se hace de la misma forma en todas las conexiones presinápticas. Este último algoritmo tiene la desventaja de que no funciona para cualquier red; el espacio alcanzable no es completo, es decir no se puede llegar a cualquier tasa poblacional deseable. El espacio alcanzable depende de las corrientes externas, pues una neurona no puede bajar su tasa de disparo local a números negativos, evitando que podamos disminuir la tasa de disparo local a un nivel arbitrariamente bajo. Esta limitante es de hecho bastante realista. Los cambios en el entorno (*corrientes externas*) son sutiles, por lo que la red, con el algoritmo propuesto, regresa a su tasa local deseable, y por lo tanto a su tasa global. Sin embargo, cuando hay cambios drásticos en el entorno, es normal que la red cambie también sus propiedades globales. Esta observación tendría que probarse por medio de experimentos.

Como trabajo futuro es analizar un modelo matemático llamado propagación hacia atrás, en donde cada capa representaría la red completa. Sin embargo tendríamos que hacer unas modificaciones a este algoritmo, pues si cada capa representa la red completa, se estaría suponiendo que la neurona está conectada consigo misma, lo cual no es biológicamente plausible. Por ende, al empezar la siguiente iteración, se tiene que actualizar el peso entre la neurona y ella misma a 0. Este cambio en cada iteración es chico, por lo que la actualización a 0 no cambiaría mucho el resultado y el algoritmo debería seguir funcionando correctamente.

Otro trabajo futuro es analizar modificaciones del algoritmo de aprendizaje de Hebb, haciendo que cumpla la propiedad analizada en esta tesis. Esto sería útil puesto que el algoritmo biológico de la tasa de disparo local no preserva el postulado de Hebb, el cual sabemos que se cumple en las redes neuronales del hipocampo por evidencia experimental.

# Capítulo 5

## Apéndice I

- **Neurona** - unidad de procesamiento elemental en el sistema central nervioso. La corteza consiste de neuronas y las celular de soporte, llamadas células gliales.

Estructura neuronal

- **Dendritas**-prolongaciones ramificadas que salen del cuerpo de una neurona y por las que recibe los impulsos nerviosos de otras neuronas a través de las sinapsis
- **Soma**- es el cuerpo celular de la neurona, el cual contiene el núcleo rodeado por el citoplasma, en el cual se hallan diferentes tipos de organelos. establece un paso de procesamiento no lineal que genera la acción potencial el cual se manda a través del axón.
- **Axón** - Prolongación filiforme que arranca del cuerpo de la neurona y termina en una ramificación que está en contacto con células musculares, glandulares, etc., o con otras células nerviosas, y por la cual circulan los impulsos nerviosos.
- **Potencial de membrana** - es la diferencia de potencial a ambos lados de una membrana neuronal que separa dos soluciones de diferente concentración de iones. Se denota por  $u(t)$ .
- **Potencial de reposo de la membrana celular** - es la diferencia de potencial que existe entre el interior y el exterior de una célula, cuando esta esta en reposo fisiológico, es decir, no esa excitada. El

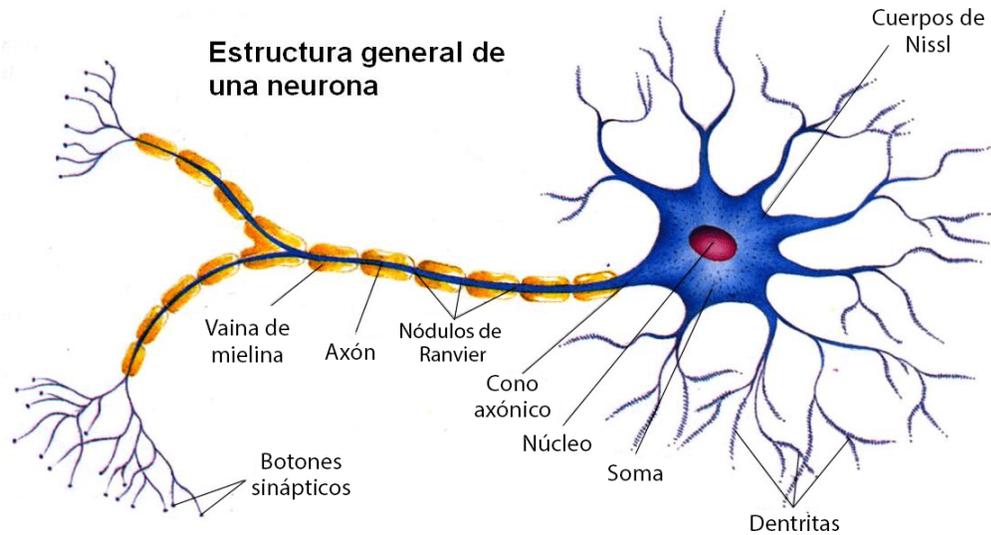


Figura 5.1: Imagen de neurona clásica con sus tres partes bien distinguibles: axón, soma y dendritas

potencial puede calcularse conociendo la concentración de los distintos iones dentro y fuera de la célula *principalmente sodio y potasio*.

El potencial de reposo se debe a que la membrana celular se comporta como una barrera semipermeable selectiva, es decir, permite el tránsito a través de ella de determinadas moléculas e impide el de otras. Este paso de sustancias es libre, no supone aporte energético adicional para que se pueda llevar a cabo.

El potencial de reposo de una neurona es aproximadamente  $-70$  mV.

- **Canales iónicos**- son un tipo de proteína en la membrana que permite el paso de iones específicos a través de la membrana celular. Su estructura semeja un poro o canal relleno de agua con un sistema de compuertas que regulan el paso de los iones. El intercambio de iones es la causa del cambio del potencial de membrana.
- **Sinapsis** - Región de comunicación entre el axón de una neurona y las dendritas o el soma de otra. El axón de la neurona **presináptica** conecta a las dendritas o al soma de la neurona **postsináptica** para llevar a cabo la transmisión del impulso nervioso. En dicha unión se

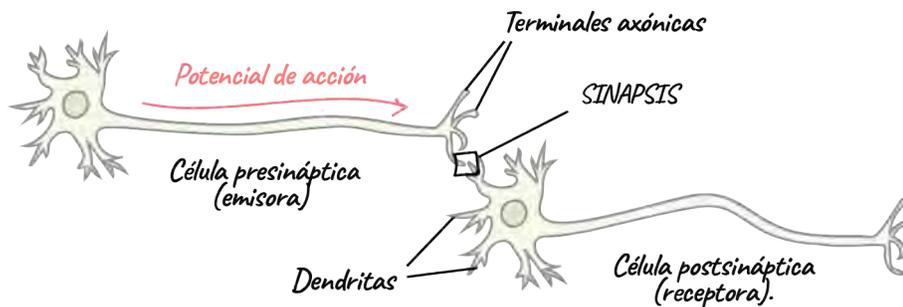


Figura 5.2: Sinapsis -lugar donde se unen dos neuronas

produce una cadena compleja de procesos bioquímicos donde se liberan **neurotransmisores**

La sinapsis puede ser:

- **inhibitoria** si el impulso de la neurona presináptica disminuye el potencial de membrana de la neurona postsináptica, es decir  $\Delta u < 0$ . Se dice que la neurona se hiperpolariza (nótese que después de un disparo la neurona siempre se hiperpolariza, teniendo como consecuencia el periodo refractario)
  - **excitatoria** si el impulso de la neurona presináptica aumenta el potencial de membrana de la neurona postsináptica, es decir  $\Delta u > 0$ . Se dice que la neurona se depolariza.
- **Neurotransmisores** - compuestos químicos que permiten la transmisión de información desde una neurona hacia otra neurona mediante la sinapsis que los separa. Son los encargados de inhibir o excitar la neurona postsináptica.
  - **Acción potencial (también llamado impulso o disparo)** - pulso eléctrico de 100 mV de duración de 1-2 ms. Este es el “lenguaje” entre neuronas, es la señal que manda una neurona a las neuronas a las que esta conectada. Puede ser observada por un electrodo. Fisiológicamente, un disparo es resultado del intercambio de de iones entre el interior y el exterior de la célula a través de los canales iónicos ubicados en la membrana. Los principales iones son sodio y potasio.
  - **Tasa de disparo** - Es definida como el número promedio de picos que aparecen durante un intervalo corto de tiempo, dividido entre la

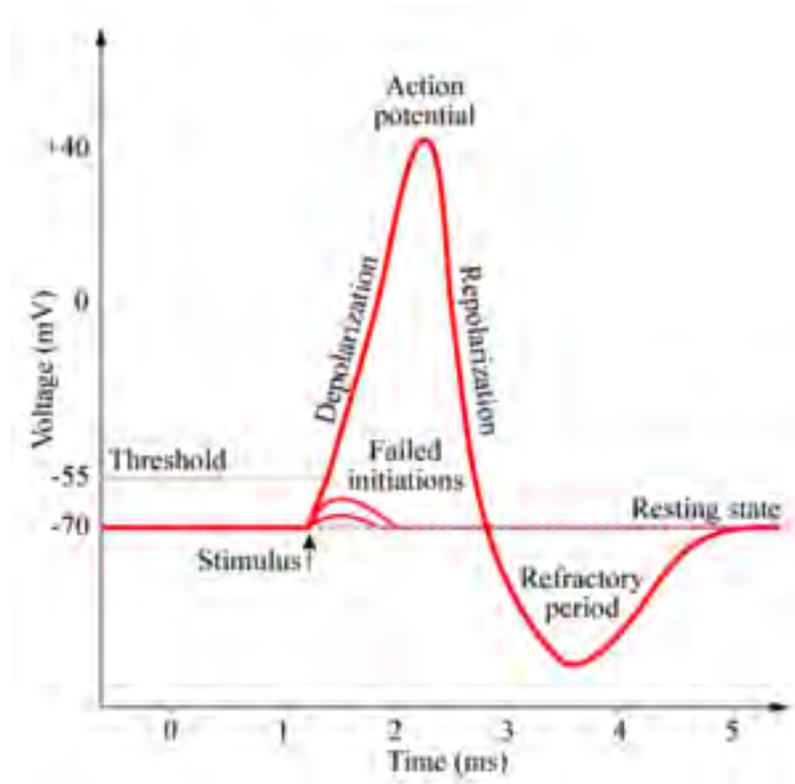


Figura 5.3: Gráfica del disparo de una neurona

duración del intervalo. Se considera que la mayor parte de información relevante la contiene la tasa de disparo neuronal, aunque el tiempo transcurrido entre cada una de ellas también es importante.

- **Periodos refractarios** - es el intervalo durante el cual es imposible desencadenar un segundo potencial de acción en la célula excitable. La neurona no puede disparar dos veces consecutivas, necesita tiempo para estabilizarse y regresar a su estado normal, es decir, a su potencial de reposo, que esta en aproximadamente -70 mV.
- **Potencial Postsináptico** - llamado PSP por sus siglas en inglés, es un cambio temporal en el potencial de membrana de la célula postsináptica causado por el flujo de iones cargados variando la probabilidad de que se produzca un potencial de acción en la neurona. Un PSP es llamado potencial excitatorio postsináptico si despolariza la membrana y aumenta la probabilidad de que se de un disparo y es llamado potencial inhibitorio postsináptico si hiperpolarizan la neurona, disminuyendo la probabilidad de disparo. Ver Figura 5.4.

Los potenciales sinápticos, a diferencia de los potenciales de acción, tienen una duración relativamente larga y no son fenómenos de «todo o nada». Estos se generan en una neurona postsináptica cuando recibe un impulso proveniente de una neurona presináptica. Tiene carácter lineal, es decir, se suman los PSP's por cada señal eléctrica recibida de una neurona presináptica. Cada sumando se pondera, dependiendo de la importancia de la neurona presináptica.

- **Potencial de Nernst** -

$$E_x = \frac{kT}{q} \ln \frac{n_2}{n_1} \quad (5.1)$$

- $x$  - algún ion en específico
- $k$  - constante de Boltzmann
- $T$  - temperatura
- $q$  - carga de los iones.
- $n_1$ - concentración de iones en un punto (para nuestra investigación es dentro de la célula)

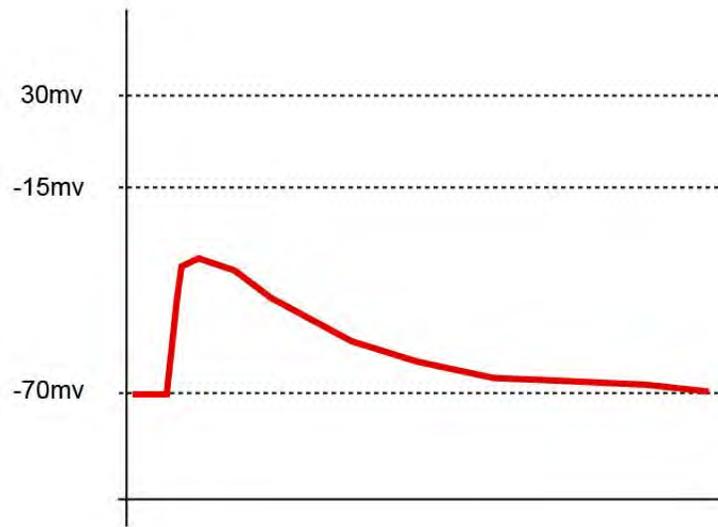


Figura 5.4: La forma de un PSP

- $n_2$ - concentración de iones en otro punto (para nuestra investigación es fuera de la célula)

La ecuación (1.1) se obtiene de la teoría termodinámica y sencillas operaciones aritméticas.

- **Potencial de inversión**  $E_x$  - así le llaman al potencial de Nernst cuando la célula esta en equilibrio, pues cuando  $\nabla u < E_x$  los iones de  $x$  fluyen adentro de la célula para disminuir la diferencia de concentración, y cuando  $\nabla u > E_x$  - fluyen hacia fuera. Por eso se le llama potencial de inversión.

## Código en Mathematica

```
In[*]:= timeconstant=R=4; threshold=10; time=10;
n=3;
timeConstant=4;
p=0.5; externalinput=Table[6,n]; T=10;
```

```
In[*]:= weightedNet[n_,p_]:= Module[{ g, Wnet},
g = RandomGraph[BernoulliGraphDistribution[n,p],DirectedEdges→True,VertexLabels→"Name"];
Wnet = Graph[g,EdgeWeight→RandomReal[{-1,1},EdgeCount[g]]];
Return[Wnet]
]
```

```
In[*]:= PSC[spikingtime_,t_]:= (- (t-spikingtime)2/3+6)HeavisideTheta[t-spikingtime]
```

```
In[*]:= PSC2[spikingtime_,t_]:=Ramp[PSC[spikingtime,t]]
```

```
In[*]:= inInit[externalInput_,n_] := Table[externalInput,n]
```

```
In[*]:= firingRate[tableST_,t0_,t1_]:=Module[{FR},
FR=Table[Length[Select[tableST[[i],t0<#<t1&]]]/(t1-t0),{i,1,Length[tableST]}];
Return[FR]
]
```

```
In[*]:= firingRate[tableST_,T_]:=Module[{FR},
FR=Table[Length[Select[tableST[[i],0<#<T&]]]/T,{i,1,Length[tableST]}];
Return[FR]
]
```

```
In[*]:= PoblationalFR[tableST_,t0_,t1_]:=Mean[firingRate[tableST,t0,t1]]
```

```
In[*]:= PoblationalFR[tableST_,T_]:=Mean[firingRate[tableST,T]]
```

```

In[*]:= inGen[graph_,nodo_,spikingtime_,in_]:=
Module[{WM, input = in, k, adj={}},
WM=Normal[WeightedAdjacencyMatrix[graph]];
For[k=1, k<=Length[nodo],k++,
Table[
If[MemberQ[Select[VertexOutComponent[graph,nodo[[k]],1],#&#nodo[[k]]&], i],
input[[i]]+=WM[[nodo[[k]],i]]*PSC2[spikingtime,t]
],
{i,1,Length[in]}];
];
Return[input]
];

```

```

In[*]:= varsGen[n_]:= Table[ToExpression["u[t]"],{i,n}];

```

```

In[*]:= ciGen[index_,sols_,st_,ic_,n_]:=
Module[{ci = ic},
ci=Table[
If[MemberQ[index,i],
ci[[i]]=0,
ci[[i]]=u[t]/.sols[[i]]/.t->st],
{i,1,n}];
Return[ci]
];

```

```

In[*]:= sisGen[vars_,input_]:=Table[timeConstant D[vars[[i]],t]==-vars[[i]]+ R input[[i]],{i,1,Length[vars]}];

```

```

In[*]:= funGen[functions_, times_]:=
Table[
Piecewise[Table[
{functions[[j,i]], If[j==1,0<t<=times[[j]],times[[j-1]] < t <= times[[j]]}],
{j,1,Length[times]}]],
{i,1,Length[functions[[1;;]]]}
];

```

```

In[*]:= (*SPIKING TRAIN*)
spktrnF[data_,opts:OptionsPattern[]]:=Module[{
options={ChartBaseStyle->EdgeForm[None],ChartElementFunction->"LineDensity",BarOrigin->Left,BarSpace->
DistributionChart[data,If[opts==={},options,PrependTo[options,{opts}]]]}

SeedRandom@0;
dat=Sort /@ RandomInteger[{200,700},{3,15}];

spktrnF[TableST,ImageSize->400,AspectRatio->1/2,ChartStyle->"SolarColors",FrameStyle->16,FrameLabel->

```

```

In[*]:= irregularidad[tableST_]:=Module[{irr,i,j},

irr=Table[If[Length[tableST[[i]]]>=2,Abs[tableST[[i,2]]-2tableST[[i,1]]],0],{i,1,Length[tableST

For[i=1,i<Length[tableST],i++;
For[j=2,j<Length[tableST[[i]],j++;
irr[[i]]+=Abs[tableST[[i,j]]-2tableST[[i,j-1]]+tableST[[i,j-2]]]];

Return[irr];
]

```

```

In[*]:= netSim[n_, p_, treshold_, externalinput_List]:=
Module[{net, sis, sols, newST=0, ic= Table[0,n], Δt=10-5, U, i0,
TableInputs, TableST=Table[{},n], TableVoltageFunctions={},
listST={1}, orderedST={}, indexST={}, index},
net = weightedNet[n,p];
(*TableInputs[t_] = inInit[externalinput,n];*)
TableInputs[t_]=externalinput;
U[t_] = varsGen[n];
While[ listST ≠ {},
listST = {};
sis = sisGen[ U[t], TableInputs[t] ];
sols = Flatten@Table[ NDSolve[Flatten@{
sis[[i]], u[ newST+Δt ] == ic[[i]],
WhenEvent[{u[t]=treshold},{AppendTo[listST,{i,t}], "StopIntegration"}],
U[t][[ i ]], {t, newST+Δt, 100}],
{i,1,Length[U[t]]}
];
index = listST[[ Flatten[ Position[ listST[;;,2], Min[listST[;;,2]] ],1],1 ];
newST = Min[listST[;;,2]];
TableInputs[t_] = inGen[net,index,newST,TableInputs[t]];
ic = ciGen[index,sols,newST,ic,n];
AppendTo[indexST,index];
AppendTo[orderedST,newST];
AppendTo[TableVoltageFunctions,sols];
Table[If[MemberQ[index,i],AppendTo[TableST[[i],newST] ],{i,1,n}];

];
Return[{net,indexST, orderedST, TableVoltageFunctions, TableST}]
]

```

```

In[*]:= netSim[G_, treshold_, externalinput_List]:=
Module[{sis, sols, newST=0, ic, dt=10-5, U, i0,
  TableInputs, TableST, TableVoltageFunctions={},
  listST={1}, orderedST={}, indexST={}, index},
  n=VertexCount[G];
  ic= Table[0,n];
  TableST=Table[{},n];
  (*TableInputs[t_] = inInit[externalinput,n];*)
  TableInputs[t_]=externalinput;
  U[t_] = varsGen[n];
  While[ listST ≠ {},
    listST = {};
    sis = sisGen[ U[t], TableInputs[t] ];
    sols = Flatten@Table[ NDSolve[Flatten@{
      sis[[i]], u[ newST+dt ] == ic[[i]],
      WhenEvent[{u[t]==treshold},{AppendTo[listST,{i,t}], "StopIntegration"}
      U[t][[ i ]], {t, newST+dt, 100}],
      {i,1,Length[U[t]]}
    ];
    index = listST[[ Flatten[ Position[ listST[;;,2], Min[listST[;;,2]] ],1],1 ];
    newST = Min[listST[;;,2]];
    TableInputs[t_] = inGen[G,index,newST,TableInputs[t]];
    ic = ciGen[index,sols,newST,ic,n];
    AppendTo[indexST,index];
    AppendTo[orderedST,newST];
    AppendTo[TableVoltageFunctions,sols];
    Table[If[MemberQ[index,i],AppendTo[TableST[[i],newST]],{i,1,n}];
  ];

  Return[{indexST, orderedST, TableVoltageFunctions, TableST}]
]

```

```

In[*]:= ImplicitFR[in_]:= (* una gran diferencia de entre los dos valores implica que los disparos no son
Module[{firingrate,firingrate2,spikingtimes},
  spikingtimes={};
  neurindividual=NDSolve[{timeconstant u'[t]==-u[t]+ R in, u[0]==0, WhenEvent[u[t]==treshold,{AppendTo[spikingtimes,t]}];
  If[Length[spikingtimes]==0,firingrate=0,firingrate =1/spikingtimes[[1]];
  If[Length[spikingtimes]==0,firingrate2=0,firingrate2 =Length[spikingtimes]/time];
  Return[{firingrate,firingrate2}];
]

```

```

In[*]:= AnalyticFR[in_]=-1/(timeconstant*Log[1-treshold/(R in)]); (*Este firing rate solo es cierto cua

```

```

In[*]:= FR2[FR1_, a12_] := Module[{new},
  newin=in/.FindRoot[AnalyticFR[in]==FR1,{in,2.6}];
  new=relFR[a12,newin][[2]];
  Return[new];
] (*NOTA, it gives only the first input value, and FR2 does change when input changes*)

```

```

In[*]:= relFR[a12_,externalinput_List]:=
Module[{TableST},
m={{∞,a12},{∞,∞}};
G=WeightedAdjacencyGraph[m,DirectedEdges→True,VertexLabels→"Name"];
{indexST1,orderedST1,TableVoltageFunctions1,TableST}=netSim[G,10,externalinput];
Return[firingRate[TableST,10]]
]

```

```

In[*]:= rangoFR[input1_List]:=Module[{a,b}, (*rango del firing rate de la neurona 2*)
a=relFR[-1,input1];
b=relFR[1,input1];
Return[{a[[2]],b[[2]]}]
]

```

# Red Neuronal con Aprendizaje de Hebbian

```
In[8]:=  $\gamma=1$ ; wthre=0.5; aplus=1; aminus=-1;  
activind[w_]:= $\gamma$  w (1-w) (wthre-w) (*The activity independent function*)
```

```
In[10]:= Hebbact1[w_,preST_,spiketimes_List]:=
```

$$\sum_{i=1}^{\text{Length}[\text{spiketimes}]} (1-w) \text{ aplus Exp}[\text{preST}-\text{spiketimes}[[i]]]$$

```
In[4]:= (*second hebbs postulate*)  
Hebbact2[w_,postST_,spiketimes_List]:=
```

$$\sum_{i=1}^{\text{Length}[\text{spiketimes}]} w \text{ aminus Exp}[\text{spiketimes}[[i]]-\text{postST}]$$

```

In[5]:= hebbNet[n_, p_, treshold_, externalinput_List]:=
Module[{net, sis, sols, newST=0, ic= Table[0,n], Δt=10-5, U, i0, Δw=0.1,
  TableInputs, TableST=Table[{},n], TableVoltageFunctions={},
  listST={1}, orderedST={}, indexST={}, index, netmatrix, vectorIN, vectorOUT, l={}},
  net = weightedNet[n,p];
  netmatrix=Normal@WeightedAdjacencyMatrix[net];
  netmatrix=netmatrix/.x_/;x==0→Infinity;
  TableInputs[t_]=externalinput;
  U[t_] = varsGen[n];
  While[ listST ≠ {},
    listST = {};
    sis = sisGen[ U[t], TableInputs[t] ];
    sols = Flatten@Table[ NDSolve[Flatten@{
      sis[[i]], u[ newST+Δt ] == ic[[i]],
      WhenEvent[{u[t]=treshold}, {AppendTo[listST, {i,t}], "StopIntegration"}
      U[t][[ i ]], {t, newST+Δt, 100} ],
      {i,1,Length[U[t]]}
    ];
    index = listST[[ Flatten[ Position[ listST[;;,2], Min[listST[;;,2]] ],1],1,1 ];
    newST = Min[listST[;;,2]];
    TableInputs[t_] = inGen[net,index,newST,TableInputs[t] ];
    ic = ciGen[index,sols,newST,ic,n];
    AppendTo[indexST,index];
    AppendTo[orderedST,newST];
    AppendTo[TableVoltageFunctions,sols];
    Table[If[MemberQ[index,i],AppendTo[TableST[[i],newST]],{i,1,n}];];
    For[i=1,i≤Length[index],i++,{
      vectorOUT=VertexOutComponent[G,index[[i],1];
      vectorIN=VertexInComponent[G,index[[i],1];
      For[j=2,j≤Length[vectorOUT],j++,If[netmatrix[[index[[i],vectorOUT[[j]]]==Infinity|n
        netmatrix[[index[[i],vectorOUT[[j]]]+=Δw(activind[netmatrix[[index[[i],vectorO
          ];
          For[j=2,j≤ Length[vectorIN],j++,If[netmatrix[[index[[i],vectorIN[[j]]]==Infinity
            netmatrix[[index[[i],vectorIN[[j]]]+=Δw(activind[netmatrix[[index[[i],vectorIN
              ];
            newmatrix=netmatrix/.x_/;x==0→Infinity;
            AppendTo[1,netmatrix/.x_/;x==Infinity→0];
            net=WeightedAdjacencyGraph[newmatrix,DirectedEdges→True,VertexLabels→"Name"];};]
          ];
    Return[{net,indexST, orderedST, TableVoltageFunctions, TableST,1}]
  ]

```

```

In[6]:= hebbNet[G_, treshold_, externalinput_List]:=
Module[{sis, sols, newST=0, ic, Δt=10-5, U, i0,

```

```

TableInputs, TableST, TableVoltageFunctions={},
listST={1}, orderedST={}, indexST={}, index,n,net=G,netmatrix,l,vectorIN,vectorOUT,Δt
n=VertexCount[net];
ic= Table[0,n];
TableST=Table[{},n];
TableInputs[t_]=externalinput;
netmatrix=Normal@WeightedAdjacencyMatrix[net];
l={netmatrix};
netmatrix=netmatrix/.x_/;x==0→Infinity;
U[t_] = varsGen[n];
While[ listST ≠ {},
  listST = {};
  sis = sisGen[ U[t], TableInputs[t] ];
  sols = Flatten@Table[ NDSolve[Flatten@{
    sis[[i]], u[ newST+Δt ] == ic[[i]],
    WhenEvent[{u[t]==treshold},{AppendTo[listST,{i,t}], "StopIntegration"}],
    U[t][[ i ]],{t,newST+Δt,100}],
    {i,1,Length[U[t]]}
  ];
  index = listST[[ Flatten[ Position[ listST[;;,2], Min[listST[;;,2]] ],1],1 ];
  newST = Min[listST[;;,2]];
  TableInputs[t_] = inGen[net,index,newST,TableInputs[t] ];
  ic = ciGen[index,sols,newST,ic,n];
  AppendTo[indexST,index];
  AppendTo[orderedST,newST];
  AppendTo[TableVoltageFunctions,sols];
  Table[If[MemberQ[index,i],AppendTo[TableST[[i],newST] ],{i,1,n}];

  For[i=1,i≤Length[index],i++,{
    vectorOUT=VertexOutComponent[G,index[[i],1];
    vectorIN=VertexInComponent[G,index[[i],1];

    For[j=2,j≤Length[vectorOUT],j++,If[netmatrix[[index[[i],vectorOUT[[j]]]==Infinity|In
      netmatrix[[index[[i],vectorOUT[[j]]]+=Δw(activind[netmatrix[[index[[i],vectorO
    ];
    For[j=2,j≤ Length[vectorIN],j++,If[netmatrix[[index[[i],vectorIN[[j]]]==Infinity
      netmatrix[[index[[i],vectorIN[[j]]]+=Δw(activind[netmatrix[[index[[i],vectorIN
    ];
    newmatrix=netmatrix/.x_/;x==0→Infinity;
    AppendTo[l,netmatrix/.x_/;x==Infinity→0];
    net=WeightedAdjacencyGraph[newmatrix,DirectedEdges→True,VertexLabels→"Name"];}]
];

Return[{indexST, orderedST, TableVoltageFunctions, TableST,l}
]

```

# Metodo del gradiente mas empinado

```
In[ ]:= quadraticCostFunction[w_, desiredFR_, input_] := Module[{wmatrix, PFR, cost, TableST},  
  wmatrix=w/.x_/;x==0→Infinity;  
  G=WeightedAdjacencyGraph[wmatrix,DirectedEdges→True, VertexLabels→"Name"];  
  TableST=netSim[G,threshold,input][[4]];  
  PFR=PoblationalFR[TableST,10];  
  cost=1/2(PFR-desiredFR)^2;  
  Return[cost];  
]
```

```
In[ ]:= updateweightGRD[currentweight_, desiredFR_, input_] := Module[{neww, h=0.1, eta, theta, K=0.01},  
  eta=RandomReal[{-0.2,0.2}];  
  theta=RandomReal[{-0.1,0.1}];  
  neww=currentweight+h(K eta quadraticCostFunction[currentweight+theta,desiredFR,input]);  
  Return[neww]  
]
```

```
In[ ]:= IterativeUpdateGRD[n_, w_, desiredFR_, input_] := Module[{neww, tablew={}},  
  neww=w;  
  For[i=1,i≤n,i++,  
    neww=updateweightGRD[neww,desiredFR,input,i];  
    tablew=AppendTo[tablew,neww];  
  Return[tablew]  
]
```

# Actualizar utilizando las tasas de disparo locales.

```
In[*]:= localFR[G_,TableFR_]:=Module[{FRvector={},i,vertexIN,mean}, (*Ve el firing rate promedio de las r
For [i=1,i≤VertexCount[G],i++,{
vertexIN=VertexInComponent[G,i,1];
mean=Mean[TableFR[[vertexIN]];
FRvector=AppendTo[FRvector,mean]}
];
Return[FRvector];
]
```

```
In[*]:= UPDATE[w_,desiredTableFR_,input_,numberiterations_]:=Module[{G,j,localfr,i,neww,weightsIN,TableST
neww=w;
For [j=1,j≤numberiterations,j++,
G=WeightedAdjacencyGraph[neww,DirectedEdges→True,VertexLabels→"Name"];
TableST=netSim[G,10,input][[4]];
localfr=localFR[G,firingRate[TableST,10]];
TablePFR=AppendTo[TablePFR,PoblationalFR[TableST,10]];
For [i=1,i≤Length[w],i++,{
neww[[;;,i]]+=0.5(desiredTableFR[[i]]-localfr[[i]]);
}];
];
Return[{TablePFR,neww,localfr}];
]
```

```
In[*]:= UPDATEwperturbation[w_,desiredTableFR_,input_,numberiterations_]:=Module[{G,j,localfr,i,neww,weig
neww=w;
For [j=1,j≤numberiterations,j++,
G=WeightedAdjacencyGraph[neww,DirectedEdges→True,VertexLabels→"Name"];
TableST=netSim[G,10,input][[4]];
localfr=localFR[G,firingRate[TableST,10]]+RandomReal[{-0.05,0.05},Length[w]];
TablePFR=AppendTo[TablePFR,PoblationalFR[TableST,10]];
For [i=1,i≤Length[w],i++,{
neww[[;;,i]]+=0.5(desiredTableFR[[i]]-localfr[[i]]);
}];
];
Return[{TablePFR,neww,localfr}];
]
```

# Bibliografía

- [1] WULFRAM GERSTNER y WERNER M. KISTLER, *Spiking Neuron Models*, Cambridge University Press, 2002.
- [2] EUGENE M. IZHIKEVICH, *Dynamical Systems in Neuroscience*, Massachusetts Institute of Technology, 2007.
- [3] EDDEN SLOMOWITZ et al., *Interplay between population firing stability and single neuron dynamics in hippocampal networks*, eLIFE, 2015.
- [4] ALEXANDER S. POZNYAK, EDGAR N. SANCHEZ y WEN YU, *Differential Neural Networks for Robust Nonlinear Control*, World Scientific Publishing Co., 2001.
- [5] SIMON HAYKIN, *Neural Networks: a comprehensive foundation*, segunda edición, Pearson Education Inc., 1999
- [6] MICHAEL NIELSEN, *Neural Networks and Deep Learning*, 2017.
- [7] PER ANDERSEN, RICHARD MORRIS, DAVID AMARAL, TIM BLISS y JOHN O'KEEFE, *The Hippocampus Book*, Oxford University Press, 2007.
- [8] ERIC R. KANDEL, JAMES H. SCHWARTZ y THOMAS M. JESSELL, *Principles of Neural Science*, cuarta edición, McGraw-Hill Companies, 2000.
- [9] ALBERT-LÁSZLÓ BARABÁSI, *Network Science*, 2016.
- [10] CARMEN CAVADA, *Historia de la Neurociencia*, Catedrática de Anatomía Humana y Neurociencia, Universidad Autónoma de Madrid, 2012.
- [11] SAM KEAN, *What happens when you remove the hippocampus?*, video en TED-Ed, 2014.

- [12] OLAF SPORNS, *The future of network neuroscience*, Network Neuroscience, Volumen 1, 2017.
- [13] BI, G. AND POO, M, *Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type*, J. Neurosci., 1998
- [14] A. L. HODGKIN AND A. F. HUXLEY, *A quantitative description of membrane current and its application to conduction and excitation in nerve*, Journal of Physiology, University of Cambridge, 1952.