

**MODELADO POR REDES NEURONALES ARTIFICIALES
BACKPROPAGATION DEL COMPORTAMIENTO EXPLORATORIO DE RATAS
WISTAR EN LABERINTO EN CRUZ ELEVADO**

CÉSAR AUGUSTO CELIS CEBALLOS

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE CIENCIAS
ESCUELA DE FÍSICA
BUCARAMANGA
2008**

**MODELADO POR REDES NEURONALES ARTIFICIALES
BACKPROPAGATION DEL COMPORTAMIENTO EXPLORATORIO DE RATAS
WISTAR EN LABERINTO EN CRUZ ELEVADO**

Trabajo de Grado como requisito para optar al título de Físico

CÉSAR AUGUSTO CELIS CEBALLOS

Director

Físico & Ing. David Alejandro Miranda Mercado, MsC

Codirector

MD. Carlos Arturo Conde Cotes, Ph.D



**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE CIENCIAS
ESCUELA DE FÍSICA
BUCARAMANGA
2008**

Dedico este trabajo a aquellos seres humanos que durante la historia de nuestra civilización han sentido un imperioso deber hacia la humanidad de ayudarlo, y que por su voluntad de poder le han permitido avanzar en la conquista de la libertad. A quienes consideran como propósito en la vida el dedicar cada segundo posible en tratar de responder a los desafíos que la naturaleza nos presenta.

Es gracias a que tuve como base el trabajo de miles de hombres que han desentrañado una gran cantidad de misterios del Universo, que pude vislumbrar la manera de enfrentar el reto que me apremió. Espero que este sea el primer paso en el largo camino que me espera por recorrer. Gracias.

AGRADECIMIENTOS

Agradezco al MsC. David Miranda, mi director, por el apoyo que me brindó en el desarrollo de este proyecto, por creer en mis capacidades intelectuales y humanas y por sus consejos. Al Ph.D Carlos Conde, mi codirector, por aportarme tanto con su experiencia en Neurociencias y Estadística, por su paciencia y dedicación, por las charlas casuales tomando café y por mostrarme el camino gratificante de la ciencia.

A mis padres, Gustavo Celis y Gladys Ceballos, que siempre me han apoyado y han tenido la fe en que vale la pena brindarme su ayuda para así convertirme en alguien mejor cada día. A mis hermanos, Gustavo y Jessica, por convivir conmigo todo este tiempo.

A uno de mis mejores amigos, Byron Ríos, quien su pasión por las Matemáticas, me inspiró a apreciarlas y hacerlas fundamento en mi vida, y con quien obtuve la fortaleza necesaria para lograr lo que me he propuesto. A Juan Camilo Aljure, por mostrarme que la filosofía es un estilo de vida cuyo alto precio vale la pena pagar.

Y por último, a cada una de las personas que han compartido conmigo algunos momentos de mi existencia, y que han hecho posible que en medio del mar de posibilidades, todo haya ocurrido tal como ha sucedido hasta ahora. Gracias.

INDICE GENERAL

1. PRESENTACIÓN DEL PROYECTO	1
1.1. PLANTEAMIENTO DEL PROBLEMA	1
1.1. OBJETIVOS	2
1.1.1. OBJETIVO GENERAL.....	2
1.1.2. OBJETIVOS ESPECIFICOS	2
1.2. JUSTIFICACIÓN	2
1.3. METODOLOGIA.....	4
1.4. ENTIDADES INTERESADAS EN EL PROYECTO	7
2. MARCO TEÓRICO	8
2.1. LABERINTO EN CRUZ ELEVADO	8
2.1.1. COMPORTAMIENTO EXPLORATORIO DE RATAS EN LABERINTO EN CRUZ ELEVADO	10
2.1.2. ESTIMULOS RESPONSABLES DE LA AVERSION DE LAS RATAS POR LOS BRAZOS ABIERTOS DEL LABERINTO EN CRUZ ELEVADO	12
2.1.3. CRITERIOS DE CARACTERIZACIÓN DEL COMPORTAMIENTO EXPLORATORIO DE RATAS EN LABERINTO EN CRUZ ELEVADO	13
2.1.4. APLICACIONES DEL MODELADO DE COMPORTAMIENTO EXPLORATORIO DE RATAS EN LABERINTO EN CRUZ ELEVADO	15
2.2. REDES NEURONALES ARTIFICIALES	15
2.2.1. NEURONA ARTIFICIAL	16

2.2.2.	FUNCIONES DE TRANSFERENCIA	17
2.2.3.	APRENDIZAJE POR ERROR-CORRECCIÓN SUPERVISADO.....	18
2.2.4.	PERCEPTRÓN MULTICAPA	19
2.2.5.	PARÁMETROS PARA LA CONSTRUCCIÓN DE PMC	21
2.2.6.	ALGORITMOS DE APRENDIZAJE PARA PMC	22
2.2.7.	ALGORITMO BACKPROPAGATION	23
2.3.	NEUROANATOMÍA FUNCIONAL DE LA RATA	24
2.3.1.	AREAS SOMATOSENSITIVAS.....	25
2.3.2.	CORTEZA VISUAL PRIMARIA	27
2.3.3.	AREAS CORTICALES MOTORAS.....	27
2.3.4.	CORTEZA PRINCIPAL DE ASOCIACIÓN	28
2.3.5.	HIPOTÁLAMO Y SISTEMA LIMBICO	28
3.	MODELOS COMPUTACIONALES DE COMPORTAMIENTO DE RATAS EN LABERINTO EN CRUZ ELEVADO	30
3.1.	MODELO SALUM-MORATO-ROQUE (SMR)	30
3.2.	SEGUNDA VERSIÓN DEL MODELO SMR	33
3.3.	MODELO GIDDINGS-STEPHENSON-KALYNCHUK (GSK).....	34
3.4.	VENTAJAS, DESVENTAJAS Y POSIBLES MEJORAS A MODELOS EXISTENTES.....	37

4. NUESTRO MODELO: EL AUTOMATA NEUROFUNCIONAL-ANATOMICO (ANA).....	39
4.1. INTERACCIÓN DEL AUTOMATA CON EL ENTORNO.....	42
4.2. DESCRIPTORES DE ENTORNO Y VARIABLES NEUROFUNCIONALES – ANATÓMICAS DEL MODELO ANA.....	45
4.3. DESCRIPCIÓN DE LA RNA.....	49
4.4. FUNCIONAMIENTO DEL MODELO	50
5. IMPLEMENTACIÓN DEL MODELO ANA.....	59
5.1. PROTOCOLO DE EXPERIMENTACIÓN	59
5.2. BUSQUEDA DEL CRITERIO DE APRENDIZAJE.....	60
5.3. VALIDACIÓN DEL MODELO BAJO LOS CRITERIOS DE NÚMERO DE ENTRADAS Y TIEMPOS DE PERMANENCIA EN BRAZOS.....	63
5.3.1. VALIDACIÓN DEL MODELO PARA VALORES ALTOS DE MSE	71
5.4. CORRELACIONES ENTRE RATA REAL Y RATA VIRTUAL CON MSE 0.001.....	73
5.5. ANÁLISIS E INTERPRETACIÓN DE DENSIDADES DE ILUMINACIÓN, PAREDES Y ALTURAS	78
5.6. ANÁLISIS E INTERPRETACIÓN DE DENSIDADES DE NOVEDAD.....	86
5.7. DISCUSIÓN.....	88
5.8. CONCLUSIONES.....	93
BIBLIOGRAFÍA	95

ANEXO A	101
ANEXO B	141
ANEXO C	147
ANEXO D	148

INDICE DE FIGURAS

Fig 1. Laberinto en Cruz Elevado.....	9
Fig 2. Distribución de LCE de acuerdo a Conde	11
Fig 3. Esquema de una neurona artificial con su función de activación $\varphi()$	17
Fig 4. Esquema del aprendizaje por error-corrección supervisado	18
Fig 5. Arquitectura de un Perceptrón Multicapa	20
Fig 6. Cerebro de rata resaltando las cortezas somatosensitiva (SOM) y visual (VIS)	25
Fig 7. Arreglo espacial de vibrisas y cuerpo en la corteza somatosensitiva.....	26
Fig 8. Cortezas motoras M. Cortezas somatosensitivas S. Cortezas visuales V. .	28
Fig 9. Estructuras cerebrales incluyendo hipotálamo y sistema límbico	29
Fig 10. LCE utilizado en el modelo SMR	30
Fig 11. LCE utilizado en el modelo GSK. El punto verde representa a la rata.....	35
Fig 12. Esquema del modelo computacional ANA.....	39
Fig 13. Campo visual de la rata.....	42
Fig 14. División del campo de visión de acuerdo a la dirección de la cabeza.....	45
Fig 15. Rata Wistar.....	59
Fig 16. Media \pm error estándar de los tiempos de permanencia en los brazos del laberinto.....	65
Fig 17. Media \pm error estándar del porcentaje de los tiempos de permanencia en los brazos del laberinto.....	66
Fig 18. Media \pm error estándar de las entradas en los brazos del laberinto.	67
Fig 19. Media \pm error estándar del porcentaje de entradas en los brazos del laberinto.....	68
Fig 20. Media \pm error estándar del número de cruzamientos de la zona central del laberinto.....	69

Fig 21. Media \pm error estándar de parámetros comportamentales de tiempos de permanencia (T), porcentaje de tiempos de permanencia (%T), número de entradas (NE) y porcentaje de entradas a los brazos del laberinto (%E) para mse de 0.4.....	72
Fig 22. Correlaciones entre los datos de las variables de los animales reales y virtuales (mse 0.001) para parámetros comportamentales de tiempos de permanencia en BA (TBA) y BC (TBC), también para sus respectivos porcentajes (%TBA y %TBC).....	74
Fig 23. Correlaciones entre los datos de las variables de los animales reales y virtuales (mse 0.001) para parámetros comportamentales de número de entradas en BA (NEBA) y BC (NEBC), también para sus respectivos porcentajes (%EBA y %EBC).....	75
Fig 24. Correlaciones entre los datos de las variables de los animales reales para tiempos de decisión constantes y reales para parámetros comportamentales de tiempos de permanencia en BA (TBA) y BC (TBC), también para sus respectivos porcentajes (%TBA y %TBC).	77
Fig 25. Evolución temporal de la densidad de iluminación de acuerdo a la dirección de la cabeza.	79
Fig 26. Evolución temporal de la densidad de paredes de acuerdo a la dirección de la cabeza.....	80
Fig 27. Evolución temporal de la densidad de altura de acuerdo a la dirección de la cabeza.....	81
Fig 28. Evolución temporal de la densidad de novedad de acuerdo a la dirección de la cabeza	87
Fig 29. Evolución temporal del diagnóstico de aprendizaje (Dx) y del mse a través de los epochs.	89

INDICE DE TABLAS

Tabla 1. Secuencia de movimientos de una rata en nomenclatura Conde	12
Tabla 2. Funciones de transferencia	17
Tabla 3. Nomenclatura de las distribuciones de probabilidades.....	35
Tabla 4. Distribuciones de probabilidades.....	36
Tabla 5. Resultados de las curvas ROC y prueba de reproducibilidad para los datos de las variables de los animales virtuales para 100 epochs	61
Tabla 6. Resultados de las curvas ROC y prueba de reproducibilidad para los datos de las variables de los animales virtuales para 500 epochs.	62
Tabla 7. Resultados de las correlaciones de Pearson entre los datos de las variables de los animales reales y virtuales para cada uno de los parámetros comportamentales variando el valor de mse entre 0.1 – 0.05 – 0.01 – 0.005 – 0.001..	64
Tabla 8. Resultados de ANOVA de una vía de Kruskal Wallis seguida de método de Dunn para comparaciones múltiples entre los datos de las variables de los animales reales y virtuales para cada uno de los parámetros comportamentales variando el valor de mse entre 0.1 – 0.05 – 0.01- 0.005 – 0.001..	64

ABREVIACIONES

ANA	Autómata Neurofuncional Anatómico
ANOVA	Análisis de Varianza
BA	Brazos abiertos del laberinto
BC	Brazos cerrados del laberinto
CERLCE	Comportamiento exploratorio de ratas en LCE
CIMBIOS	Ciencia de Materiales Biológicos y Semiconductores
FFCLRP	Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto
GSK	Giddings-Stephenson-Kalynchuk
GUI	Guide User Interface
LCE	Laberinto en Cruz Elevado
MSE	Mean Square Error
NEBA	Número de entradas a los brazos abiertos
NEBC	Número de entradas a los brazos cerrados
NYC	Neurociencias y Comportamiento
PMC	Perceptrón Multicapa
ROC	Receiver Operating Characteristic
RNA	Red Neuronal Artificial

SMR	Salum-Morato-Roque
SRL	Sistema de Referencia de Laboratorio
SRR	Sistema de Referencia de la Rata
T	Tiempo total de permanencia en el laberinto
TBA	Tiempo de permanencia en los brazos abiertos del laberinto
TBC	Tiempo de permanencia en los brazos cerrados del laberinto
UIS	Universidad Industrial de Santander
UPB	Universidad Pontificia Bolivariana
%EBA	Porcentaje de entradas a los brazos abiertos
%EBC	Porcentaje de entradas a los brazos cerrados
%TBA	Porcentaje del tiempo de permanencia en los brazos abiertos
%TBC	Porcentaje del tiempo de permanencia en los brazos cerrados

RESUMEN

Titulo:

MODELADO POR REDES NEURONALES ARTIFICIALES BACKPROPAGATION DEL COMPORTAMIENTO EXPLORATORIO DE RATAS WISTAR EN LABERINTO EN CRUZ ELEVADO

Autor:

César Augusto Celis Ceballos

Palabras claves:

Laberinto en cruz elevado, modelado computacional, red neuronal artificial backpropagation, densidad.

Descripción:

El estudio del comportamiento de una rata de laboratorio en el laberinto en cruz elevado (LCE) es empleado en la comprensión de algunos estados anímicos y en pruebas de medicamentos ansiolíticos y ansiogénicos. Algunos autores sugieren que el modelado computacional de una rata de laboratorio en LCE permitiría un mejor entendimiento del funcionamiento del cerebro ante ciertos medicamentos y bajo ciertos estados anímicos. El modelado de una rata de laboratorio en LCE se puede realizar de diversas maneras, siendo el empleo de redes neuronales artificiales (RNA) una alternativa con un apreciable significado neurobiológico.

Se reporta un modelo computacional del comportamiento de ratas Wistar macho expuestas durante cinco minutos al LCE bajo condiciones de luz controlada, inyectadas con solución salina, basado en una RNA backpropagation. Los parámetros de entrenamiento de la red fueron extraídos directamente del ambiente en que se realizó el experimento y se introdujeron a la RNA. Para caracterizar los estímulos, se definieron parámetros visuales y táctiles tales como: densidad de iluminación, densidad de paredes, densidad de alturas y densidad de novedad.

Se midió el tiempo de permanencia y el número de entradas en los brazos abiertos y cerrados del laberinto para valores absolutos y porcentuales, también el número de cruzamientos en la zona central. Se fijó el nivel de significancia en el 5%. El modelo se evaluó usando curvas ROC, correlaciones de Pearson y ANOVA. Los resultados muestran la existencia de un criterio de aprendizaje de las RNA y la capacidad del modelo de reproducir el comportamiento exploratorio a partir de un mse de 0.01.

* Proyecto de Grado

** Facultad de Ciencias. Escuela de Física. Director: David Alejandro Miranda Mercado.

Codirector: Carlos Arturo Conde Cotes.

ABSTRACT

Title:

BACKPROPAGATION ARTIFICIAL NEURAL NETWORK MODELLING OF WISTAR RAT EXPLORATORY BEHAVIOR IN ELEVATED PLUS-MAZE

Author:

César Augusto Celis Ceballos

Key words:

Elevated plus-maze, computational modelling, backpropagation artificial neural network, density.

Description:

Study of laboratory rat behavior in elevated plus-maze (EMP) is used to understand mood states and to test anxiolytic and anxiogenic drugs. Some authors suggest that the computational modelling of a laboratory rat behavior in EMP would allow a better understanding of the brain operation under the effects of certain drugs and under certain mood states. The modelling of behavior of a laboratory rat in EMP can be conducted in different ways, being the artificial neural networks (ANN) an alternative with important neurobiological meaning.

This document reports a computational model of male Wistar rats injected with saline solution and exposed during five minutes to EMP under controlled conditions of light. The computational model is based in backpropagation ANN. The parameters of training were extracted directly from the experimental environment and introduced into the ANN. To characterize the stimulus, it defined visual and tactile parameters as illumination density, wall density, height density and novelty density.

The number of entries into and time spent in the open and closed arms, absolute and percentual values, and the number of crossings in the central zone were measured. The significance was placed in 5%. The model was evaluated through ROC curves, Pearson correlations and ANOVA. The results show the existence of an ANN learning criterion and the model's ability to reproduce the exploratory behavior since 0.01 mse.

* Undergraduate Thesis

** Faculty of science. Department of physics. Supervisor: David Alejandro Miranda Mercado.
Co-supervisor: Carlos Arturo Conde Cotes

1. PRESENTACIÓN DEL PROYECTO

1.1. PLANTEAMIENTO DEL PROBLEMA

La elaboración de modelos computacionales de sistemas neurales permite realizar hipótesis, pruebas y validaciones de procedimientos para el análisis de fármacos suministrados en humanos. Tal es el caso de los fármacos ansiolíticos, los cuales cumplen la función de disminuir a nivel cerebral los procesos que generan ansiedad en un ser humano [5,24,31,32].

Uno de los modelos animales utilizados para el estudio de trastornos de ansiedad, es el del Laberinto en Cruz Elevado (LCE), el cual consiste en un laberinto con dos “brazos abiertos” (BA) y dos “brazos cerrados” (BC) [16,26]. A grosso modo, en los experimentos en LCE se miden los siguientes parámetros: Tiempo de permanencia en los brazos y Número de entradas a los brazos (tanto absolutos como porcentuales), con los cuales se puede determinar si un fármaco produce efectos de tipo ansiolítico o no [30,31,32].

Actualmente es reducida la cantidad de modelos computacionales que permitan simular el comportamiento de una rata en el LCE [13,23,36,39]. Es por esto que se planteó el desarrollo de un modelo computacional que permita emular el comportamiento exploratorio de ratas en Laberinto en Cruz Elevado (LCE) a partir de la definición de parámetros ambientales.

1.1. OBJETIVOS

1.1.1. OBJETIVO GENERAL

- Proponer un modelo computacional basado en redes neuronales artificiales *backpropagation* para simular el comportamiento exploratorio de ratas en un Laberinto en Cruz Elevado

1.1.2. OBJETIVOS ESPECIFICOS

- Definir parámetros que permitan modelar los estímulos externos que percibe la rata *wistar* en laberinto en cruz elevado.
- Seleccionar las variables de entrada y salida de la red neuronal artificial *backpropagation* usada para modelar el comportamiento exploratorio de ratas *wistar* en laberinto en cruz elevado.
- Entrenar y validar el modelo computacional con conjuntos de entrenamiento y validación obtenidos a partir de los resultados experimentales registrados por el Grupo de Neurociencias y Comportamiento NYC UIS-UPB bajo los criterios del Número de entradas y Tiempo de permanencia en los brazos.

1.2. JUSTIFICACIÓN

Una de las problemáticas actuales en Neurociencias consiste en desarrollar modelos computacionales del cerebro de mamíferos, neuroanatómica y neurofuncionalmente análogos a este, que recreen los mecanismos y procesos

que ocurren en él, especialmente en ratas, debido a su gran uso por parte de los experimentalistas alrededor del mundo.

Tomando en consideración que la rata es un ente inteligente y que existe una gran variabilidad individual dadas sus condiciones genéticas y su experiencia ambiental, no es fácil realizar experimentos bajo un estricto control, lo que hace más complicado su estudio [7]. La comprensión del comportamiento exploratorio de la rata en el LCE (o en cualquier ambiente relativamente controlado), requiere de aproximaciones sucesivas que ayuden al entendimiento de los múltiples procesos involucrados e implica proponer hipótesis dirigidas a cada etapa del procesamiento de la información manteniendo un alto control de cada elemento participante. Una de las polémicas vigentes alrededor de los mecanismos involucrados en la motivación comportamental exploratoria de la rata en el LCE, se refiere a cuál sería la naturaleza de la información ambiental que induce a la rata evitar la exploración de los llamados BA del LCE [3,4,8,14,28,30,32,35,38,42,43].

Por otro lado, teniendo en cuenta que la relación de los parámetros de entrada y salida que gobiernan el movimiento del animal, se caracterizan por ser altamente no lineales, y considerando que las redes neuronales se han propuesto como algoritmos que podrían ajustarse a los procesamientos cerebrales [12,22], se propuso para el estudio del comportamiento exploratorio de ratas en LCE el diseño de una Red Neuronal Artificial (RNA), como una herramienta computacional para modelado y simulación de fenómenos no-lineales [26], tal como el comportamiento exploratorio de roedores, basada en definición de parámetros ambientales como aporte al estudio de la naturaleza de los estímulos principales en este modelo, esperando que permitiera simular los comportamientos básicos de una rata expuesta al LCE.

Es por esto que el grupo de investigación de la Escuela de Física de la UIS Ciencia de Materiales Biológicos y Semiconductores CIMBIOS y el Grupo de NYC UIS-UPB del Departamento de Ciencias Básicas, el cual realiza pruebas con ratas en el Laberinto en Cruz Elevado (LCE), concibieron la idea de crear un modelo computacional basado en Redes Neuronales Artificiales (RNA), capaz de reproducir el comportamiento exploratorio de una rata dentro del LCE, tratando a esta como un ente autómatas, por lo cual la toma de decisiones que esta ejecute serán inducidos por los estímulos externos que el medio le proporcione y procesos mentales internos propios de la naturaleza neuronal de la rata.

1.3. METODOLOGIA

Se realizó una investigación transversal en la cual los datos de entrenamiento y validación de la red neuronal artificial fueron tomados de experimentos realizados en el Grupo de NYC UIS-UPB. La investigación se dividió en cinco fases metodológicas descritas a continuación:

Primera Fase

Se hizo una revisión bibliográfica que incluyó la búsqueda y estudio de artículos publicados en revistas científicas, además de memorias de eventos, libros y exploración por la web. Para la búsqueda de información se utilizó *Google*, la base de datos de la IEEE, IoP, Science Direct, entre otras.

También se hizo la revisión de ciertos libros especializados, tal como *Neural Networks: A comprehensive foundation* de Simon Haykin, y algunas otras fuentes bibliográficas facilitadas por el Grupo de NYC UIS-UPB.

Segunda Fase

Con base en la información recolectada en la primera fase metodológica, se seleccionaron y definieron parámetros para modelar los estímulos externos que percibe la rata *wistar* en laberinto en cruz elevado (en adelante, variables de entorno).

Las variables de entorno fueron la base para definir las variables de entrada al modelo computacional. Las variables de salida del modelo indicaron la dirección y sentido en que debía moverse la rata *wistar* para unos ciertos estímulos de entrada.

Además de ello se utilizó el modelo neurofuncional de la rata [29] para plantear un algoritmo de pre-procesamiento de los valores numéricos asociados a las variables de entorno. Este pre-procesamiento sirvió para definir las variables de entrada a la red neuronal artificial *backpropagation*, además de indicar sus valores numéricos.

Tercera Fase

En esta fase se escribieron las rutinas en Matlab que permitieron la:

- Implementación del pre-procesamiento de los valores numéricos de las variables de entorno utilizando el modelo neurofuncional acondicionado en la fase anterior
- Implementación de la red neuronal artificial *backpropagation*
- Implementación de un algoritmo para la interpretación de la salida de la red neuronal artificial *backpropagation* y la obtención del vector de desplazamiento para que la rata se moviera en el laberinto en cruz elevado, modelando de esta manera el comportamiento exploratorio de la rata

Por último, se implementó el modelo computacional integrado por las diferentes rutinas programadas en Matlab.

Cuarta Fase

Se utilizaron resultados experimentales obtenidos en el Grupo de NYC UIS-UPB de comportamiento de ratas *wistar* en LCE, para la determinación de los valores numéricos de las variables de entorno y los datos de salida del modelo computacional, que corresponden a los desplazamientos de la rata en el laberinto.

El comportamiento exploratorio de ratas en LCE se estudió siguiendo un protocolo de experimentación explicado en mayor detalle en el capítulo 5 de implementación del modelo. Los videos analizados del comportamiento exploratorio de ratas *wistar* fueron tomados de los archivos de experimentos del Grupo de NYC UIS-UPB.

Posteriormente se dividieron los datos en dos grupos, uno fue utilizado para entrenar las redes neuronales artificiales y el otro para validar el modelo.

La validación se hizo simulando el comportamiento de la rata virtual en laberinto en cruz elevado medido en función de los movimientos de la rata, y comparándolos con los valores de los datos del grupo de validación bajo los criterios del Número de entradas y Tiempo de permanencia en los brazos.

Quinta fase

Por último se preparó el presente reporte en el cual se describirá el trabajo realizado y resultados obtenidos.

1.4. ENTIDADES INTERESADAS EN EL PROYECTO

- Grupo de Neurociencias y Comportamiento UIS-UPB
- Grupo de Investigación Ciencia de Materiales Biológicos y Semiconductores CIMBIOS.

2. MARCO TEÓRICO

El desarrollo de este proyecto requirió del manejo de tres ejes principales del conocimiento en neurociencias y comportamiento, inteligencia artificial y neuroanatomía. A continuación se resume en este capítulo los fundamentos teóricos más relevantes referentes al laberinto en cruz elevado, redes neuronales artificiales y la neuroanatomía de la rata.

El capítulo está organizado en tres partes. En la primera parte, que comprende desde la sección 2.1 hasta la 2.1.4 se explica lo referente al laberinto en cruz elevado y el comportamiento exploratorio de la rata en el LCE. La segunda parte del capítulo, que comprende desde la sección 2.2 hasta la 2.2.7 discute algunos fundamentos de redes neuronales artificiales, la arquitectura del Perceptrón Multicapa y el algoritmo *backpropagation*. Por último, desde la sección 2.3 hasta la 2.3.5 se encuentra un pequeño compendio de la neuroanatomía de la rata enfocada a las cortezas visuales, somatosensitivas, motoras y el sistema límbico.

2.1. LABERINTO EN CRUZ ELEVADO

Dentro de los llamados modelos etológicamente fundamentados, se destaca el Laberinto en Cruz Elevado (LCE) por su fácil implementación y por su consistencia como modelo del llamado trastorno de ansiedad generalizada [16,27,30,31,32]. Su fundamento se relaciona con la aversividad específica que presentan los roedores por los lugares abiertos, de manera que desde el punto de vista etológico, estos animales prefieren los lugares cerrados o cercanos a paredes para construir sus viviendas y para protegerse de predadores [2,15,30,42,43].

El LCE es un dispositivo hecho en madera con 4 brazos distribuidos en forma de cruz. Cada brazo mide 50x12 cm y están elevados 50 cm sobre el suelo. Dos de sus brazos, Brazos Cerrados (BC), colocados uno al frente del otro tienen paredes laterales de 40 cm de altura, permitiendo solamente como vía de acceso el centro del laberinto. Los otros dos brazos, Brazos Abiertos (BA), de iguales dimensiones y con igual orientación, tienen como paredes laterales un pequeño borde de acrílico de 2 cm de altura.

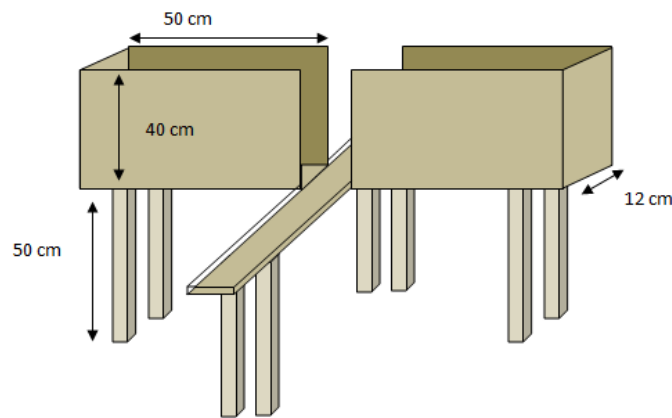


Fig 1. Laberinto en Cruz Elevado

Este modelo evalúa la exploración libre de la rata en dos ambientes, uno potencialmente aversivo (BA) y otro seguro (BC). Fármacos ansiolíticos clásicos como las benzodiazepinas, aumentan la exploración y el tiempo de permanencia en los BA mientras que los fármacos ansiogénicos como el pentilentetrazol invierten dicho efecto [5,31,32,42,43]. Este modelo es probablemente el más ampliamente utilizado en la actualidad como bioensayo para la búsqueda de nuevos fármacos ansiolíticos.

2.1.1. COMPORTAMIENTO EXPLORATORIO DE RATAS EN LABERINTO EN CRUZ ELEVADO

El movimiento del animal puede ser explicado en primera aproximación como el resultado de una ponderación entre la motivación de explorar y la aversión que experimenta en una determinada posición del laberinto [16,27,35]. En su estado natural la rata elige estar cerca de superficies verticales, preferiblemente rincones y lugares con poca iluminación, los campos abiertos y las alturas inducen comportamientos que permiten inferir estados aversivos, lo que genera que la rata frecuente y permanezca más en los BC que los BA [2,15,16,27,30,31,32,35,37,42,43]. Estos comportamientos pueden ser modulados por el efecto de fármacos ansiolíticos clásicos como las benzodiazepinas y son exacerbados por fármacos ansiogénicos como el pentilentetrazol [5,31,32,42,43].

Existen varios comportamientos específicos que conforman el comportamiento exploratorio de la rata en el LCE (CERLCE). Entre ellos se encuentra el *rearing*, que consiste en el levantamiento del cuerpo apoyado en sus patas traseras, el *head-dipping*, que ocurre cuando la rata agacha su cabeza y la saca por fuera de los brazos abiertos con el fin de explorar las alturas que se encuentran dispuestas a cada lado, y el *risk-assesment* o evaluación de riesgo, en el cual la rata encontrándose en los BC, estira su cuerpo y ubica su cabeza en la zona central o en los BA, con lo cual explora las zonas aversivas y así decide si se “arriesga” o no a ir hacia ellas. En el presente modelo, sólo se trató el comportamiento exploratorio básico que no incluye *rearing*, *head-dipping* ni *risk-assesment*, esto es, el comportamiento es evaluado por las entradas y los tiempos de permanencia en cada lugar.

El entorno en el que se encuentra la rata es el LCE dividido en 13 zonas: 3 zonas por brazo y una zona central. El movimiento se restringe por las paredes y alturas

que imposibilitan a la rata abandonar el LCE, por lo que siempre se encuentra en una zona definida. Las zonas 1-2-3 corresponden a los BC, la zona 4 es la zona central y las zonas 5-6-7 corresponden a los BA. La anterior distribución corresponde a la utilizada en el Grupo de NYC UIS-UPB, dirigido por el MD Carlos Conde (PhD), y por medio de la cual es posible caracterizar el CERLCE como una secuencia de posiciones y tiempos.

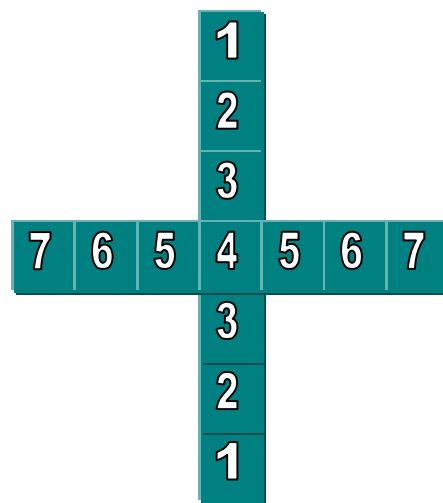


Fig 2. Distribución de LCE de acuerdo a Conde

Nótese en la figura 2 que esta distribución se basa en el principio de la no-lateralidad de la rata frente a la exploración libre del LCE, que indica que estadísticamente la rata no presenta una preferencia en la exploración de un brazo del mismo tipo respecto al otro ubicado al lado opuesto del primero [37].

En la Tabla 1 se presenta una secuencia de movimientos obtenida de un experimento con rata *wistar* en LCE realizado en el Grupo de NYC UIS-UPB siguiendo el protocolo de experimentación mencionado en el capítulo 5 de implementación, donde cada columna contiene el orden de la medición observada (Observación), la posición de la rata en el LCE (Posición) y el tiempo de permanencia en esta (Tiempo).

Tabla 1. Secuencia de movimientos de una rata en nomenclatura Conde

Observación	Posición	Tiempo (s)
1	4	0,38
2	3	0,27
3	2	0,38
4	1	29,05
5	2	3,07
6	3	6,37
7	4	0,49
8	5	1,64
9	6	0,38
10	7	4,88
11	6	0,6
12	5	0,82
13	4	2,36
14	3	0,54
15	2	2,25
16	3	0,43
17	4	0,49
18	3	0,32
19	2	0,27
20	1	245,01

2.1.2. ESTIMULOS RESPONSABLES DE LA AVERSION DE LAS RATAS POR LOS BRAZOS ABIERTOS DEL LABERINTO EN CRUZ ELEVADO

Los trabajos publicados de Montgomery y Mithani et al., permitieron que otros investigadores desarrollaran el LCE como un modelo etológico de ansiedad [16,27]. Posteriormente, Pellow et al. validaron el LCE y desde entonces ha sido amplio su uso en múltiples campos de investigación [30,31,32]. Sin embargo, es aún poco lo que se sabe sobre los estímulos responsables del comportamiento de ansiedad que experimenta la rata al ser expuesta por primera vez al LCE. Mucho se ha propuesto sobre cuales serían los estímulos más relevantes del entorno y que podrían influenciar el comportamiento exploratorio de la rata, de tal manera que esta presentara diferencias significativas (comúnmente utilizando un ANOVA) respecto al comportamiento de una rata control. Actualmente se manejan cuatro

líneas de investigación con el propósito de dilucidar esta incógnita: la primera propone que la tigmotaxia aporta al comportamiento en el LCE de manera significativa [15,42,43], la segunda indica que la altura a la que se encuentra el LCE influye [32,43], la tercera y una de las más estudiadas en los últimos años, propone como el estímulo más importante y que domina el fenómeno, incluso por encima de los demás, el nivel de iluminación de las distintas áreas del LCE [3,8,14,28,37,38], y por último la novedad que experimenta la rata al encontrarse en un entorno nuevo e inexplorado [43].

2.1.3. CRITERIOS DE CARACTERIZACIÓN DEL COMPORTAMIENTO EXPLORATORIO DE RATAS EN LABERINTO EN CRUZ ELEVADO

Dos de los criterios de caracterización del CERLCE más validados a nivel mundial por los científicos que trabajan en neurociencias y comportamiento son: el número de entradas en los brazos y el tiempo de permanencia en los mismos [30]. El Número de entradas hace referencia a la cantidad de veces en que la rata estando en la zona 4 entra a un BA o a un BC, definiéndose a la entrada solo cuando todas las 4 patas del animal se encuentran dentro del brazo [14]. El Tiempo de permanencia hace referencia al tiempo total en el que la rata permanece durante el experimento en los BA o en los BC.

En adelante se utilizará la siguiente notación para hacer referencia a cada parámetro descrito anteriormente:

- TBA** - Tiempo de permanencia en los brazos abiertos del laberinto
- TBC** - Tiempos de permanencia en los brazos cerrados del laberinto
- T** - Tiempo total de permanencia en el laberinto

NEBA - Número de entradas a los brazos abiertos

NEBC - Número de entradas a los brazos cerrados

%TBA - Porcentaje del tiempo de permanencia en los brazos abiertos

%TBC - Porcentaje del tiempo de permanencia en los brazos cerrados

%EBA - Porcentaje de entradas a los brazos abiertos

%EBC - Porcentaje de entradas a los brazos cerrados

donde matemáticamente los parámetros porcentuales se calculan de la siguiente forma;

$$\%TBA = \frac{TBA * 100}{T} \quad \%TBC = \frac{TBC * 100}{T}$$
$$\%EBA = \frac{NEBA * 100}{(NEBA + NEBC)} \quad \%EBC = \frac{NEBC * 100}{(NEBA + NEBC)}$$

Estos parámetros cumplen las desigualdades;

$$NEBC > NEBA \quad TBC > TBA \quad \%EBC > \%EBA \quad \%TBC > \%TBA$$

para la primera exposición de la rata al LCE a una libre exploración de este por 5 minutos [5,11,14], sin recibir ninguna clase de tratamiento antes o durante el experimento, ya sea farmacológico [5,24,31,32,42,43] (exceptuando la inyección de solución salina), o de manipulación [24], para un rango de niveles de iluminación desde 30 hasta 900 lx [4], medidos en la zona central del laberinto con un luxómetro estándar.

Estas relaciones son válidas tanto para valores absolutos como porcentuales, y es comúnmente encontrada en casi toda publicación referente al LCE. Estas relaciones matemáticas de invariancia de los parámetros y sus desigualdades, conforman la línea base del CERLCE, cuya importancia dada por los investigadores se debe a que a través del grado de perturbación que distintos estímulos realicen en esta línea, se pueden proponer los factores que participarían en los estados de ansiedad en ratas.

2.1.4. APLICACIONES DEL MODELADO DE COMPORTAMIENTO EXPLORATORIO DE RATAS EN LABERINTO EN CRUZ ELEVADO

El LCE es una prueba muy aceptada y validada en Neurociencias para evaluar diferentes fenómenos neurológicos probados en ratas. Entre ellos se destacan el análisis del efecto de fármacos ansiolíticos y ansiogénicos [5,24,31,32,42,43], así como la búsqueda de explicaciones de los mecanismos neurobiológicos asociados a diferentes tipos de ansiedad y de memoria emocional. También ha sido utilizado para el estudio de aspectos etológicos dirigidos a comprender los efectos de luminosidad [3,8,14,28,37,38], la manipulación [24], las alturas [32,43] y la tigmotaxia [15,42,43].

2.2. REDES NEURONALES ARTIFICIALES

Una Red Neuronal Artificial (RNA) es un paradigma de aprendizaje y procesamiento automático inspirado en la forma en que funciona el sistema nervioso de los animales que posee una estructura de procesamiento distribuido masivamente en paralelo hecho de unidades simples o neuronas interconectadas

en una red que colaboran para producir un estímulo de salida o almacenar conocimiento de la experiencia y hacerlo disponible para su uso posterior [22,41].

Una de las aplicaciones de las RNA consiste en simular las propiedades observadas en los sistemas neuronales biológicos a través de modelos matemáticos recreados mediante mecanismos artificiales. El objetivo es conseguir que las máquinas den respuestas similares a las que es capaz de dar el cerebro que se caracterizan por su generalización y su robustez [20,22,33]. Principalmente las RNA se asemejan a los sistemas neuronales biológicos en 2 aspectos:

- El conocimiento es adquirido por la red neuronal de su entorno a través de un proceso de aprendizaje
- Las fuerzas de las conexiones interneuronales, conocidas como pesos sinápticos son usadas para almacenar el conocimiento adquirido

2.2.1. NEURONA ARTIFICIAL

El modelo de una neurona artificial es matemáticamente la sumatoria de los valores de entrada x_i previamente multiplicados por sus respectivos pesos sinápticos w_i , y posteriormente evaluados en la función de activación $\varphi()$, para dar la salida v .

Matemáticamente se puede expresar con la ecuación;

$$v = \varphi\left(\sum w_i x_i\right)$$

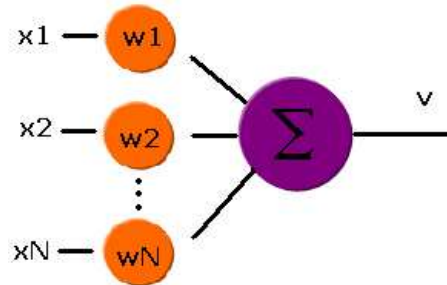





Fig 3. Esquema de una neurona artificial con su función de activación $\phi()$

2.2.2. FUNCIONES DE TRANSFERENCIA

Las funciones de transferencia son aquellas funciones matemáticas utilizadas para modelar el comportamiento de una neurona artificial. Estas toman la sumatoria de las entradas pesadas más el sesgo y retornan una salida correspondiente.

Tabla 2. Funciones de transferencia

Nombre	Relación Entrada/Salida	Icono	Función
Lineal	$a = n$		<i>purelin</i>
Sigmoidal Logarítmico	$a = \frac{1}{1 + e^{-n}}$		<i>logsig</i>
Tangente Sigmoidal Hiperbólica	$a = \frac{e^n - e^{-n}}{e^n + e^{-n}}$		<i>tansig</i>

Existe una gran cantidad de funciones de transferencia, más aquellas que se destacan por su amplio uso están contenidas en la tabla 2.

2.2.3. APRENDIZAJE POR ERROR-CORRECCIÓN SUPERVISADO

En términos conceptuales, el paradigma de aprendizaje supervisado se aplica básicamente a un sistema de aprendizaje cuya respuesta frente a la estimulación del ambiente (respuesta del sistema) es comparada con la de un profesor (respuesta deseada), definiéndose la diferencia entre estas dos respuestas como la señal de error que retroalimenta al sistema y cuya función es inducir correcciones en los parámetros libres (pesos sinápticos y sesgos) de la red de manera que su respuesta emule a la del profesor. Dicho de otra manera, el aprendizaje del sistema es supervisado por el profesor [22].

Este proceso se realiza iterativamente hasta obtener un valor de error determinado. Se considera que esta emulación es óptima desde un cierto sentido estadístico. El diagrama del proceso se muestra en la fig 4.

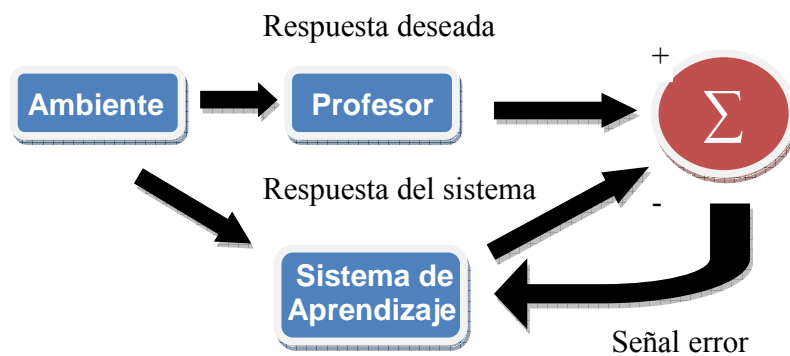


Fig 4. Esquema del aprendizaje por error-corrección supervisado

Es posible definir varias funciones de error, de acuerdo al criterio del programador. Una de las más utilizadas para este tipo de paradigma de aprendizaje es el mse (*mean-square-error*) o error cuadrático medio que se ve representada como

una superficie de error en el espacio de los parámetros libres y que se expresa por la ecuación;

$$mse = \frac{\sum_{k=1}^n e_k^2}{n}$$

donde e_k es el correspondiente error de la neurona de salida y n la cantidad de ejemplos de entrenamiento, definidos como el conjunto de parejas de vectores entrada y respuesta deseada administrados por el profesor para supervisar el entrenamiento. Cada estado actual del sistema durante el entrenamiento corresponde a un punto en la superficie de error y lo que se desea en cada iteración es mover este estado hacia el mínimo global, evitando de paso quedar atascado en un mínimo local, lo cual se logra a través de la actualización de los parámetros libres en función del gradiente del mse [22,40,41].

2.2.4. PERCEPTRÓN MULTICAPA

Es un tipo de arquitectura, entre tantas que existen en RNA, que tiene como características principales:

- Las neuronas se encuentran dispuestas en capas, que pueden ser capa de entrada, capas ocultas o capa de salida
- Las neuronas de una determinada capa se encuentran totalmente conectadas con la capa anterior y posterior a ella

Para disponer del uso de una RNA con aprendizaje supervisado, tal como el PMC, esta debe ser primero entrenada con un conjunto de ejemplos de entrenamiento a

través de sucesivas iteraciones, hasta que esta reproduzca lo enseñado con un valor de error por debajo de un límite establecido por el usuario.

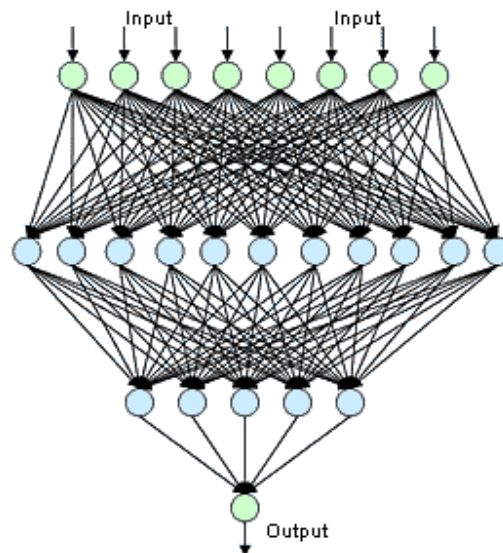


Fig 5. Arquitectura de un Perceptrón Multicapa¹

Los PMC tienen como propiedades principales:

- Un PMC es no-lineal si también lo son sus neuronas, siendo esto muy útil si la señal de entrada es también no-lineal
- Sirven para modelar procesos neurobiológicos permitiendo explicar el funcionamiento de las neuronas
- Ofrecen el potencial del reconocimiento de patrones complejos, discriminación no-lineal y generalización
- Aprenden eficientemente y son insensibles a datos incompletos o a ruido
- Mientras mayor sea el número de capas ocultas, mayor es su capacidad de discriminación de patrones complejos

¹ www.optimaltrader.net/neural_network.htm

La aplicación clásica y principal de los PMC consiste en el reconocimiento y procesamiento de patrones [20,21,33,40] o como aproximador de funciones [22,26,40,41].

2.2.5. PARÁMETROS PARA LA CONSTRUCCIÓN DE PMC

Algunas recomendaciones que muy probablemente son del tipo heurístico son:

- La capa de entrada debe tener igual cantidad de nodos que entradas
- La capa de salida debe tener tantas neuronas de salida como las salidas que se quieren
- Las capas ocultas deben tener más neuronas por capa que las de la capa de salida
- Usar solo una capa oculta, máximo dos
- Usar pocas neuronas por capa

Los últimos dos ítems se basan en que mientras más capas y neuronas tenga la red, más inestable será el gradiente, y por consecuencia el entrenamiento demorará más tiempo y disminuirá su capacidad de convergencia, incluso hasta no haber convergencia en absoluto. Además el número de mínimos locales aumenta drásticamente, por lo que hay una alta probabilidad de que el entrenamiento se estanque en un mínimo local y no logre converger al mínimo global [22,41].

La RNA debe mantenerse lo más pequeña posible, ya que intentos de incrementar el tamaño de un PMC para acomodar entradas complejas presentan dificultades. Incrementando el número de neuronas o agregando más capas se incrementa el

esfuerzo requerido para entrenar la red. Por lo general este esfuerzo extra falla al tratar de desarrollar una solución satisfactoria [20,21].

Para la mayoría de los casos en que una RNA trabaja en modo de reconocimiento de patrones, se aconsejan salidas que varíen entre 0 y 1, o que solo tomen valores binarios, es decir, 0 o 1. Por esto lo más conveniente es disponer neuronas *logsig* en la capa de salida, mientras que en las capas ocultas se aconsejan tanto *logsig* como *tansig*. También se sugiere que todos los valores de entrada sean normalizados para que estos se encuentren en el mismo orden de magnitud, y evitar así que uno sea despreciable respecto a otro, lo que complicaría el entrenamiento de la red debido al amplio dominio que esta debe manejar [22,40,41].

2.2.6. ALGORITMOS DE APRENDIZAJE PARA PMC

Es muy difícil saber de manera a priori para un determinado problema cual es el algoritmo más eficiente para el entrenamiento de un PMC. Depende de varios factores, tal como la complejidad del problema, el número de datos en el entrenamiento, el número de pesos y sesgos del PMC, el error de meta (valor de error de salida que se debe alcanzar para detener el entrenamiento) y si el PMC está siendo usado para realizar reconocimiento de patrones (análisis discriminativo) o no [22,40,41].

Sin embargo, uno de los más eficientes algoritmos de error-corrección para el entrenamiento de un PMC hasta ahora creado, es el *backpropagation*, el cual utiliza como función de desempeño el mse [22,40,41]. Los sistemas que usan el algoritmo de *backpropagation* [25,40,41,45] han sido usados para alcanzar altos niveles en el reconocimiento de patrones [20,21,33,40].

2.2.7. ALGORITMO BACKPROPAGATION

La función del algoritmo *backpropagation* es ajustar los pesos de manera que el gradiente de la función de desempeño (en este caso el mse) tienda a 0, lo cual se logra a través de iteraciones de la ecuación;

$$\vec{x}(k+1) = \vec{x}(k) - \alpha * \vec{g}(k)$$

Donde $\mathbf{x}(k+1)$ son los pesos y sesgos nuevos, $\mathbf{x}(k)$ los actuales, α la tasa de aprendizaje y $\mathbf{g}(k)$ el gradiente. Existen dos maneras de aplicar la ecuación anterior: modo incrementativo y modo de lote [22,40].

En el modo incrementativo el gradiente se computa y los pesos se actualizan después de cada entrada aplicada a la red. En el modo de lote todas las entradas son aplicadas a la red antes de que los pesos se actualicen.

Durante el entrenamiento del PMC con el algoritmo *backpropagation* dos señales recorren la red, una que va y otra que se devuelve. La señal de función que va hacia adelante se propaga por la red y posiciona los pesos sinápticos, y cuando esta llega a la capa de salida, la señal de salida y la respuesta deseada se restan para calcular la señal de error, tal como se muestra en la ecuación;

$$e_j(k) = d_j(k) - y_j(k)$$

donde $e_j(k)$ es la señal de error, $d_j(k)$ es la respuesta deseada y $y_j(k)$ la señal de función de la neurona j para la iteración k .

Esta señal de error se propaga de vuelta actualizando los valores de los pesos sinápticos de una capa en función de los valores de los pesos sinápticos de la capa posterior, presentándose así el fenómeno de retropropagación. El proceso finaliza cuando se obtiene una señal de salida muy cercana a la deseada en un sentido estadístico.

Es necesario tener en cuenta que cuando se usa este algoritmo para la solución de problemas de patrones de reconocimiento las salidas de las neuronas se saturan, o dicho de otra forma, no operan en la región lineal y cuando la cantidad de datos a entrenar es grande, algunos entrenamientos no convergen. Es por esto que se propone entonces el uso de una variante del algoritmo *backpropagation* llamada *resilient backpropagation*, el cual es un algoritmo muy eficiente, principalmente en problemas de reconocimiento ya que supera los inconvenientes que presenta trabajar con funciones sigmoideas y además requiere poca memoria para su cómputo [40].

2.3. NEUROANATOMÍA FUNCIONAL DE LA RATA

La corteza cerebral es la capa de sustancia gris que cubre el centro de sustancia blanca de los hemisferios cerebrales y representa la superficie del cerebro. Esta se divide en áreas citoarquitectónicas de acuerdo a la clasificación del histólogo alemán Korbinian Brodmann, pero más importante que la clasificación citoarquitectónica es la clasificación funcional de la corteza cerebral en varias áreas motoras y sensitivas [1,29].

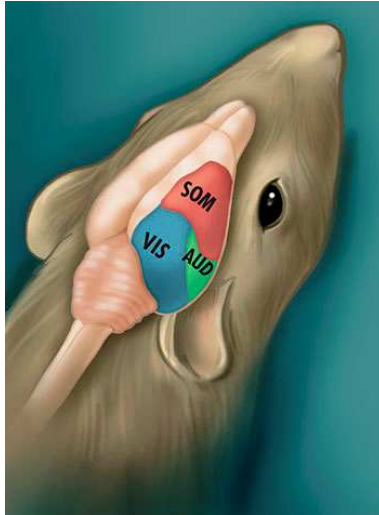


Fig 6. Cerebro de rata resaltando las cortezas somatosensitiva (SOM) y visual (VIS)²

Un área relativamente pequeña de la corteza cerebral está especializada en recibir información sensorial de los ojos y piel y después de un procesamiento por parte de las cortezas encargadas de ello (motora, premotora y somatosensitiva) se proyectan impulsos motores a través de la vía corticospinal hacia la medula espinal con el fin de generar el movimiento voluntario [1].

2.3.1. AREAS SOMATOSENSITIVAS

Área somatosensitiva primaria

Se relaciona básicamente con las modalidades sensitivas, pero también le es posible generar respuestas motoras después de su estimulación. Esta recibe fibras nerviosas que conducen sensibilidad general (tacto, dolor y temperatura). Las eferentes de la corteza somatosensitiva primaria se dirigen a la corteza motora, corteza de asociación somatosensitiva y medula espinal [1].

² www.impactlab.com/2006/08/25/scientists-erase-memories-in-brains-of-rats/

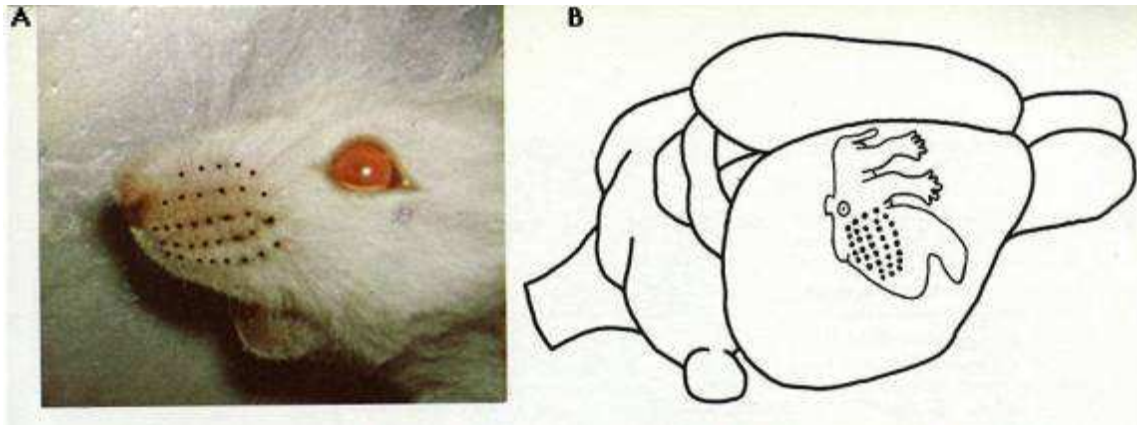


Fig 7. Arreglo espacial de vibrisas y cuerpo en la corteza somatosensitiva³

En el área de la corteza somatosensitiva se representa la sensibilidad del cuerpo en forma precisa, pero desproporcionada, donde la representación de las vibrisas es demasiado grande respecto de su tamaño relativo en el cuerpo. Este es un reflejo de la importancia funcional de esta región en la función sensitiva [1].

Área de asociación somatosensitiva

Recibe casi todos sus impulsos a partir del área somatosensitiva primaria. El procesamiento de aferencias somatosensitivas multisensoriales permite la percepción de forma, tamaño y textura, tanto la identificación de objetos mediante el contacto. Esta área se proyecta hacia áreas de asociación multimodales que reciben aferencias de más de una modalidad sensorial y sirven a la integración intermodal y las percepciones multisensoriales. Esta área desempeña un importante papel en la integración de estímulos visuales y somatosensitivos [1].

³ anackley.myweb.uga.edu/thesis2.htm

2.3.2. CORTEZA VISUAL PRIMARIA

El área visual primaria recibe fibras del núcleo geniculado lateral, que proceden de la retina. Las proyecciones de la retina en la corteza visual se organizan de una forma espacial determinada. La función de la corteza visual primaria es la de procesar la información visual y extraer de ella características tales como orientación, movimiento, etc [1].

2.3.3. AREAS CORTICALES MOTORAS

Área motora primaria

En esta área se representa el cuerpo en forma precisa pero desproporcionada. Esta recibe fibras de la corteza somatosensitiva primaria y la corteza motora suplementaria. Las conexiones entre las cortezas motora primaria y somatosensitiva son recíprocas. Esta funciona al inicio de movimientos finos de gran destreza. Esta no es la única área desde la cual puede generarse movimiento, es la que se caracteriza por iniciar el movimiento de gran destreza a un umbral de estimulación más bajo que el de otras áreas motoras [1].

Área motora suplementaria

Es crucial en la organización temporal del movimiento. La información anatómica y fisiológica disponible sugiere que el área motora suplementaria puede ser el sitio donde las aferencias y comandos externos concuerdan con las necesidades internas, lo cual conduce a facilitar la formulación (programación) de una estrategia de movimiento voluntario. El umbral de estimulación del área motora suplementaria es más alto que el de la corteza motora primaria [1].

Área premotora

Se relaciona con la función motora voluntaria dependiente de aferencias sensoriales (visuales y somatosensitivos). El umbral de estimulación que produce respuestas de esta área es más alto que el requerido para la corteza motora primaria. Esta se activa cuando se establece un nuevo programa motor o cuando este cambia conforme a la información sensorial recibida, por ejemplo cuando el animal explora el medio ambiente [1].

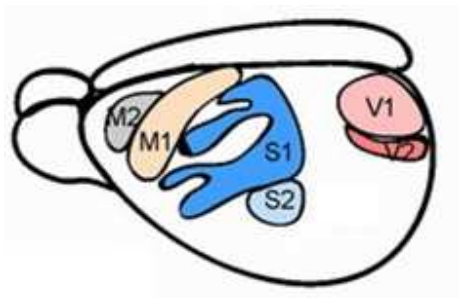


Fig 8. Cortezas motoras M. Cortezas somatosensitivas S. Cortezas visuales V. ⁴

2.3.4. CORTEZA PRINCIPAL DE ASOCIACIÓN

Esta se conecta con todas las áreas corticales sensitivas y de esta manera funciona en la percepción multisensorial compleja y de alto nivel [1].

2.3.5. HIPOTÁLAMO Y SISTEMA LIMBICO

El hipotálamo participa en la memoria a través de sus conexiones con la formación hipocámpica y núcleo talámico anterior, aunque los mecanismos subyacentes de

⁴ www.ciencia.cl/CienciaAlDia/volumen5/numero2/articulos/articulo4.html

esta son muy complejos y aún no se aclaran. El sistema límbico también participa en los procesos referentes a la memoria, con estructuras como el hipocampo y la amígdala. El hipocampo es importante para la memoria de acontecimientos y objetos. La amígdala interviene en la respuesta de orientación ante situaciones nuevas deteniendo la actividad en proceso y orientando el cuerpo a la situación nueva. Esta se manifiesta por la capacidad del animal de responder a cambios en el medio ambiente visual [1].

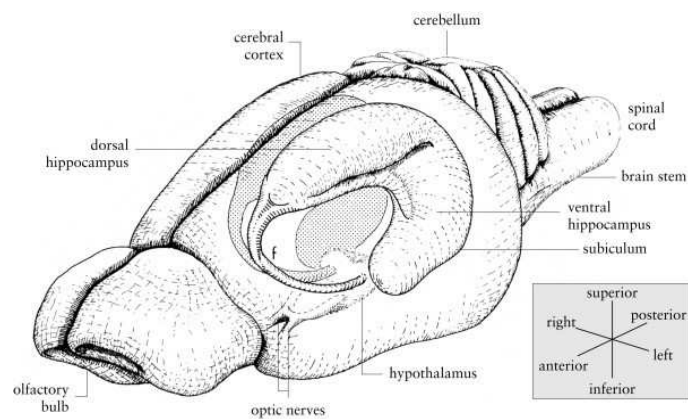


Fig 9. Estructuras cerebrales incluyendo hipotálamo y sistema límbico⁵

⁵ www.biomedcentral.com/1471-2202/6/36/figure/F3

3. MODELOS COMPUTACIONALES DE COMPORTAMIENTO DE RATAS EN LABERINTO EN CRUZ ELEVADO

3.1. MODELO SALUM-MORATO-ROQUE (SMR)

Este modelo pionero, publicado en 1999 por Cristiane Salum, Sílvia Morato y Carlos Antonio Roque-da-Silva, investigadores de la FFCLRP, Universidad de São Paulo, Campus de Ribeirão Preto, São Paulo, Brasil, describe el CERLCE como un conflicto de la rata entre permanecer en un lugar seguro (BC) y la exploración de un entorno potencialmente peligroso (BA), o dicho de otra forma, como una competencia entre la motivación y la aversión [35,36].

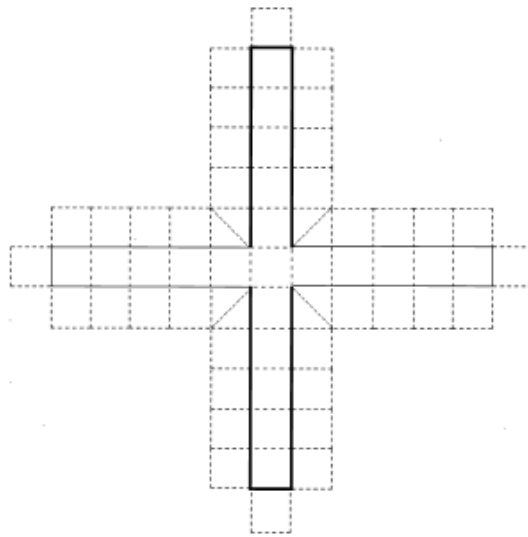


Fig 10. LCE utilizado en el modelo SMR

El modelo consiste en una RNA basada en la estructura del LCE dividido en cuadrados de igual tamaño, cinco en cada brazo y uno central, para un total de 21 unidades de movimiento. Cada unidad de la RNA corresponde a un cuadro

específico el cual se conecta sólo con sus vecinos más cercanos, hacia donde la rata tiene posibilidad de ir. También se agregaron zonas por fuera del laberinto, a las cuales la rata no puede moverse pero que si puede explorar, para un total de 65 unidades.

Las líneas gruesas y oscuras representan las paredes que rodean los brazos cerrados, y perpendicular a estos se encuentran los brazos abiertos. El comportamiento del animal se caracteriza por medio del NEB y TB. La nomenclatura usada corresponde a:

w_{ij} - elemento de matriz de pesos (conectando la unidad j con la i)

M_{ij} - componente de matriz de pesos asociada con la motivación exploratoria de la rata virtual

A_{ij} - componente de matriz de pesos asociada con la aversión de la rata virtual

M - constante que determina la motivación exploratoria inicial de la rata virtual

C - constante que determina la aversión inicial de la rata virtual

N_{ij} - número de veces que la rata virtual pasa de la unidad j a la i

r - parámetro de actividad espontánea

a_i - factor discriminativo de paredes de la motivación exploratoria

b_i - factor discriminativo de paredes del decaimiento de la motivación exploratoria

g_i - factor discriminativo de alturas del incremento y decremento de la aversión

μ - factor de variabilidad aleatoria para la motivación exploratoria

v - factor de variabilidad aleatoria para la aversión

El modelo está basado en el uso de una RNA competitiva, la cual tiene una matriz de pesos w_{ij} que representa la memoria emocional de la rata;

$$w_{ij} = M_{ij} - A_{ij}$$

donde M_{ij} corresponde a la motivación de explorar, y A_{ij} a la aversión de moverse de la zona j hacia la zona i , y se obtienen a partir de las siguientes ecuaciones;

$$M_{ij} = \frac{M}{(\alpha_i N_{ij} + \mu)^{b_i}} \quad A_{ij} = C(1 \pm v g_i N_{ij})$$

Las anteriores ecuaciones se basan en las suposiciones de que la tendencia exploratoria disminuye con el tiempo de exposición al entorno, y como consecuencia la rata evita retornar a la última posición visitada, de igual manera los valores de aversión también varían con N_{ij} . También se incluyen los parámetros aleatorios μ y v para expresar las diferencias en las secuencias de movimiento para diferentes ratas frente a iguales condiciones del entorno.

Una vez obtenido el valor de w_{ij} , este valor se modifica de acuerdo a la tendencia natural de las ratas de preferir como dirección de movimiento hacia delante con un 40% de probabilidad, un 20% por igual hacia ambos lados, 10 % hacia atrás y 10% de permanecer en la posición actual. La dirección de movimiento se define en referencia de la “dirección de la cabeza” la cual equivale a la dirección del último movimiento.

La velocidad de movimiento de la rata virtual permanecía constante a lo largo de la prueba y era de 2 s por movimiento, lo que no permitía comparar tiempos absolutos de permanencia en el laberinto, pero si el porcentaje de tiempo gastado en los brazos. El modelo fue probado por medio de análisis estadísticos de ANOVA, Bonferroni y pruebas t.

Los resultados arrojaron que las medidas de %TB y de NEB eran similares entre la rata real y virtual, lo que verificaba en primera aproximación del fenómeno la capacidad del modelo para reproducir el CERLCE.

3.2. SEGUNDA VERSIÓN DEL MODELO SMR

Posteriormente, en el 2007, se menciona en una presentación de poster una segunda versión del modelo SMR, realizada por los mismos tres investigadores de la primera versión más la colaboración de dos nuevos integrantes Julián Tejada y Rodrigo Oliveira. Los resultados de las pruebas mostraron similitud entre los datos de ratas virtuales y reales frente a los parámetros de TB y NEB [39].

La hipótesis de modelar el comportamiento exploratorio por medio de una matriz de pesos w_{ij} , donde cada elemento representa la tendencia de la rata por explorar la zona i estando en j , se mantiene. También lo hace la ecuación;

$$w_{ij} = M_{ij} - A_{ij}$$

donde M_{ij} corresponde a la motivación de explorar, y A_{ij} a la aversión de moverse de la zona j hacia la zona i . El cambio se produjo en la estructura matemática de M_{ij} y A_{ij} , siendo ahora;

$$M_{ij} = \frac{M}{(b_{ij}N_{ij} + \mu)} \quad A_{ij} = \frac{A}{(a_{ij}N_{ij} + \nu)}$$

donde, M y A son constantes, μ y ν parámetros que varían de rata en rata, y a_i y b_i son exponentes positivos. Los valores de los parámetros se determinaron a través de una exhaustiva búsqueda usando algoritmos genéticos.

Roque et al. concluyen que este modelo modificado es capaz de reproducir el CERLCE, gracias al desarrollo matemático en el que se basa, en este caso, una RNA competitiva. También consideran como su mayor debilidad el gran número de parámetros, que dificultan su interpretación biológica o cognitiva.

Tomando en consideración la forma en que se definen aversión y exploración como parámetros del modelo, es razonable creer que estos no aportan a la comprensión de la naturaleza de los estímulos “responsables” de la instalación biológica de los parámetros de aversión y exploración.

3.3. MODELO GIDDINGS-STEPHENSON-KALYNCHUK (GSK)

Este modelo fue desarrollado como Tesis de Grado para obtener el título de BSc en Matemáticas y Estadística en el 2002 de Joanna Marie Giddings en la Universidad de Acadia, Canadá, quien fue supervisada por el Dr. Paul Stephenson, y quien también contó con la ayuda para su desarrollo de la Dra. Lisa Kalynchuk de la Universidad de Dalhousie. El modelo consistió en el desarrollo de 7 distribuciones de probabilidades basadas en datos experimentales tomados de videos de experimentos con ratas en el LCE [13].

Las probabilidades que sirven para determinar el siguiente movimiento de la rata, difieren de acuerdo a la zona en el laberinto en la cual la rata se encuentre, y también de la dirección del movimiento. Este modelo está en capacidad de reproducir el comportamiento exploratorio de ratas en LCE que no han recibido ningún tratamiento que pudiera afectar su nivel de ansiedad.

El LCE utilizado en el modelo se compone de cuatro brazos divididos en 3 zonas más la zona central. El modelo cuenta con la exploración de las zonas dentro del LCE y también de zonas adyacentes, con lo cual se trata el comportamiento

exploratorio de las zonas del LCE, también el de las paredes (*rearing*), las alturas (*head-dipping*) y la evaluación de riesgo o *risk-assesment*.

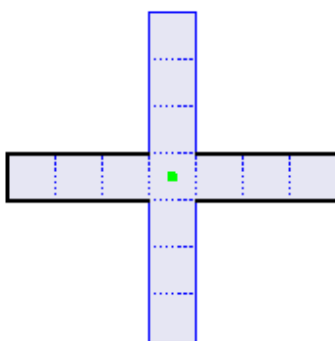


Fig 11. LCE utilizado en el modelo GSK. El punto verde representa a la rata

En cuanto a la simulación, cada iteración corresponde a 1 s en tiempo real. En cualquier instante de tiempo la rata tiene 5 posibles movimientos, Adelante, Atrás, Quieto, *Rear* y *Risk*. El movimiento de la rata es determinado por la salida de un generador de números aleatorios entre 0 y 1 y la distribución de probabilidades definida en la siguiente tabla, de manera que reproduzca las tendencias de las ratas de entrar más a los BC que a los BA, y de permanecer más tiempo en los BA que BA. *Risk-assesment* es sólo posible para zonas de los BC adyacentes al centro y para la zona central, como se observa en la tabla 2.

Tabla 3. Nomenclatura de las distribuciones de probabilidades

1	Brazo abierto	Movimiento hacia fuera
2		Movimiento hacia dentro
3	Brazo cerrado, no adyacente	Movimiento hacia fuera
4		Movimiento hacia dentro
5	Brazo cerrado, adyacente	Movimiento hacia fuera
6		Movimiento hacia dentro
7	Centro	

Tabla 4. Distribuciones de probabilidades

Distribución	1	2	3	4	5	6	7	
P(Adelante)	.08	.635	.11	.42	.86	.05	P(BA)	.02
P(Atrás)	.13	.00	.13	.02	.00	.03	P(BC)	.13
P(Quieto)	.34	.285	.59	.46	.10	.59	P(Quieto)	.51
P(Rear)	.45	.08	.17	.10	.03	.08	P(Rear)	.03
P(Risk)	---	---	---	---	.01	.25	P(Risk)	.31

La hipótesis del modelo plantea que el comportamiento exploratorio de la rata es gobernado por una inercia de movimiento, por la cual si la rata se está moviendo en una dirección particular, esta tenderá a continuar en esa dirección, luego cada decisión que toma la rata es afectada por el último movimiento. Lo anterior se soporta sobre la observación de la tendencia de la rata a moverse con poco retroceso cuando esta va saliendo del brazo abierto o entrando al cerrado, que correspondería a ejecutar la secuencia 7-6-5-4-3-2-1.

Este es un modelo estacionario, debido a que las distribuciones de probabilidades no varían con el tiempo, considerándose que la variación de estos valores durante los 5 minutos de simulación es muy poca y se puede despreciar como una primera aproximación del fenómeno.

Para verificar la capacidad del modelo, se obtuvieron resultados por medio de un análisis estadístico, después de haber simulado 100.000 iteraciones, de los cuales se usaron 5 medidas: %EBA, %TBA, NEBC, NEBA y el número total de entradas. Este modelo al igual que el SMR está en capacidad de reproducir el CERLCE, bajo las anteriores cinco medidas, sin embargo tiene limitaciones cualitativas semejantes a las mencionadas en el modelo anterior.

3.4. VENTAJAS, DESVENTAJAS Y POSIBLES MEJORAS A MODELOS EXISTENTES

Los modelos anteriormente referenciados, presentan ciertas ventajas, al igual que desventajas y posibilidad de mejoramiento. A continuación se enuncian ciertos ítems de estos aspectos.

VENTAJAS:

- Plantean el fenómeno por medio de una matemática sencilla que brinda a las simulaciones una gran estabilidad y capacidad de reproducción del fenómeno bajo los parámetros de TB y NEB
- Incorporan un análisis probabilístico consecuente con el orden aleatorio observado en las secuencias de movimiento de una gran cantidad de experimentos con ratas

DESVENTAJAS:

- No incluyen niveles de iluminación ni mecanismos de detección del entorno de la rata
- No realizan la toma de decisión de movimiento a través de la estructura mental, fisiológica o anatómica de la rata, excepto por la RNA del modelo SMR que es una analogía de Memoria Emocional

- Presentan dificultad para darle alguna interpretación biológica o cognitiva a los datos; son modelos de más carácter descriptivo que explicativo del fenómeno
- No separan la rata de la geometría del entorno, por lo cual, al ubicarla en otro tipo de entorno diferente al LCE, el modelo falla

POSIBLES MEJORAS:

- Representar a la rata como un ente autómatas capaz de tomar la decisión de movimiento basada en información relevante que esta adquiera del entorno que la rodea a través de los sentidos
- Los algoritmos utilizados para la toma de decisión deben emular anatómicamente y funcionalmente el cerebro de una rata real
- Expresar las ecuaciones que describen el movimiento de la rata virtual a través del laberinto en función de parámetros asociados al entorno, tal como el nivel de iluminación, de campo abierto o de proximidad a las alturas que presenten las zonas
- Generar algoritmos computacionales que a partir de la ubicación de la rata en el entorno, permitan obtener información relevante de este, emulando así los sentidos de la rata real, y que al ser procesada posteriormente permita obtener los parámetros asociados al entorno

4. NUESTRO MODELO: EL AUTOMATA NEUROFUNCIONAL-ANATOMICO (ANA)

El modelo se diseñó como un grupo de funciones de Matlab que emulan computacionalmente la fisiología de ciertas estructuras corporales y funcionalmente algunas estructuras cerebrales involucradas en la adquisición y tratamiento de señales biológicas. Los algoritmos funcionan como sistemas entrada/salida y cada uno procesa cierta parte de la información que posteriormente es introducida a la RNA, la cual modela las funciones de las cortezas motoras y determina una de las cuatro posibles direcciones de movimiento de la rata virtual en el laberinto: adelante, atrás, derecha e izquierda.

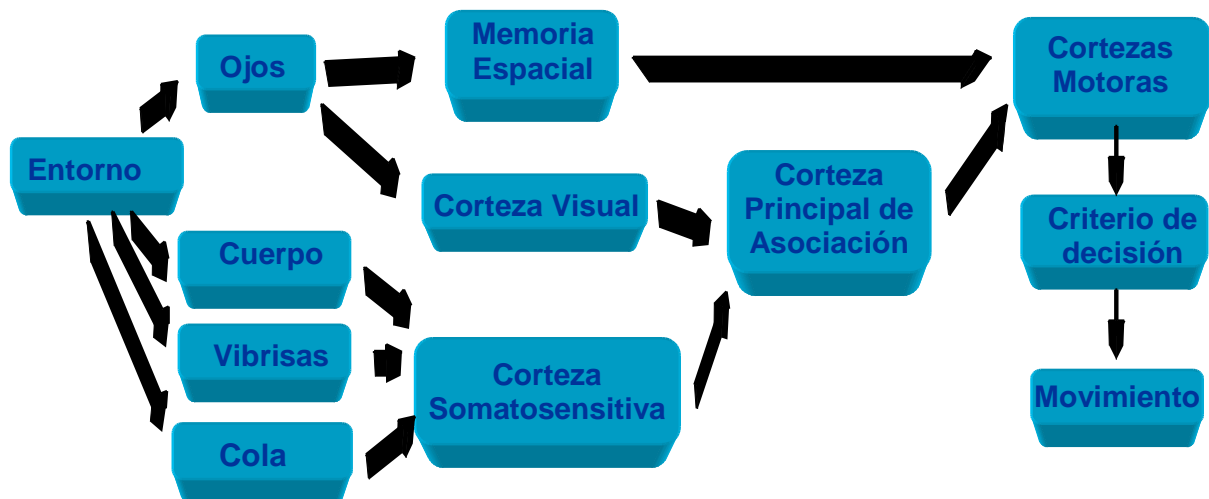


Fig 12. Esquema del modelo computacional ANA

La programación de los algoritmos se realizó en Matlab por la facilidad que presentaba este software en la creación, entrenamiento y simulación de RNA del

tipo PMC, conteniendo una gran cantidad de funciones ya programadas dentro de su *Neural Network Toolbox*, también por la rapidez en la programación de códigos, verificación de errores, etc.

Se empleó programación estructurada para generar los códigos que permitirían emular las funciones cerebrales de la rata, permitir la interacción de esta con el entorno por medio de los sentidos, realizar las simulaciones de comportamiento, y el desarrollo de funciones de diagnóstico y herramientas de anexo, propias de la matemática utilizada y del caso particular de trabajo (ver Anexo A).

También se empleó la programación orientada a objetos para generar interfaces gráficas por medio del GUI (*Guide User Interface*) de Matlab (ver Anexos A y B), permitiendo generar ventanas para la introducción y visualización de datos y gráficos, siendo esto muy útil tanto para el programador como para un usuario no experto en programación, pero capaz de utilizar el modelo computacional.

Todos los algoritmos fueron programados como funciones de Matlab (archivos *.m). Además de funciones, se crearon archivos temporales, almacenados en la carpeta temp~, útiles durante la ejecución, y también se utilizaron y generaron archivos en los formatos *.mat y *.xls para el almacenamiento de datos obtenidos de los procesos, para lo cual se crearon algoritmos para exportar los datos de Matlab a Excel, y viceversa (ver Anexo A).

Debido al gran número de procedimientos generados para el procesamiento de los datos, fue necesario el uso de múltiples nomenclaturas para estos. Es por esto que las posiciones en el LCE se disponen en el modelo en dos nomenclaturas:

- Nomenclatura por zonas, explicada anteriormente en la sección del LCE del capítulo 2, en la cual se representan las posiciones de la rata en el LCE como un número entre 1-2-3-4-5-6-7, donde las zonas 1-2-3 corresponden a los BC, la zona 4 es la zona central y las zonas 5-6-7 corresponden a los BA. La anterior distribución corresponde a la utilizada en el Grupo de NYC UIS-UPB, dirigido por el MD Carlos Conde (PhD)
- Nomenclatura vectorial, corresponde a vectores de 2 elementos donde el primero representa la ubicación de la rata en una determinada fila y el segundo elemento representa la columna dentro de la matriz de entorno *envm* definida en el Anexo A

También como consecuencia de la propiedad de la rata virtual de ser un ente autómatas que puede ubicarse en múltiples entornos, fue necesaria la definición de dos sistemas de referencia de movimiento:

- Sistema de Referencia de Laboratorio (SRL), en el cual la dirección de movimiento se representa analíticamente como Arriba, Abajo, Izquierda y Derecha, tomándose como referencia la matriz de entorno
- Sistema de Referencia de Rata (SRR), en el cual la dirección de movimiento se representa analíticamente como Adelante, Atrás, Izquierda y Derecha, tomándose como referencia la dirección de la cabeza de la rata

4.1. INTERACCIÓN DEL AUTOMATA CON EL ENTORNO

La rata virtual utiliza dos mecanismos para percibir el entorno. Por medio de la vía visual, a través de los ojos, adquiere información sobre el nivel de luminosidad de las zonas (medida en Lux), ubicación de paredes y alturas tanto para zonas cercanas como lejanas, y por medio del tacto, a través de las vibrisas, cuerpo y cola, adquiere información de la ubicación de paredes y alturas cercanas a la rata.

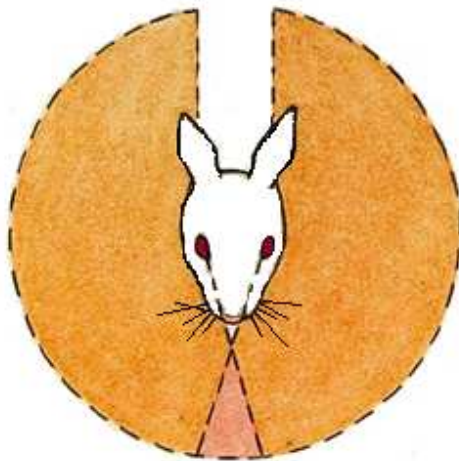


Fig 13. Campo visual de la rata

El campo visual de la rata como primera aproximación se puede cuantificar en 360° [44]. La rata prefiere permanecer en las zonas rodeadas de paredes, sin alturas y con baja luminosidad del LCE, mientras que siente aversión por zonas con alta luminosidad, al igual que por zonas con grandes campos abiertos y cercanas a alturas [2,15,16,27,30,32,37,42,43]. De igual manera, se siente muy atraída por aquellas zonas que esta ha visto, pero en las que no ha estado antes, posteriormente disminuye la motivación de verlas más seguido, o de pasar por ellas cada vez más [5,11,27,43]. La interpretación matemática de estas motivaciones en el modelo se hace a través de las densidades de paredes, alturas e iluminación para expresar la componente aversiva de los estímulos, y a través de la densidad

de novedad para expresar la componente motivacional de la exploración del entorno.

La definición del entorno se realizó por medio de una matriz *envm* de 7x9x6, donde los elementos de *envm(:, :, 1)* corresponden a las zonas de movilidad = 1, y alturas = 2. En más detalle, las zonas de movilidad se definen como aquellos lugares en los que la rata se puede encontrar y por lo tanto puede moverse sobre ellas. En el caso del LCE corresponden a superficies planas sin inclinación que conforman los llamados brazos del laberinto y la zona central. A la vez, se debe entender por alturas, como aquellos espacios vacíos ubicados alrededor de las zonas de movilidad situados por debajo de ellas respecto al plano horizontal.

Desde *envm(:, :, 2)* hasta *envm(:, :, 5)* se definieron las ubicaciones de las paredes, donde se requirieron de 4 matrices para asignar a cada zona del entorno sus respectivas paredes arriba, abajo, a la izquierda y derecha de esta. El elemento con valor 1 indica la presencia de una pared y 0 indica su ausencia.

En la matriz *envm(:, :, 6)* se definieron los niveles de iluminación de cada zona, los cuales se encuentran entre 0 y 500 Lux. A las zonas de movilidad de los BA y a las alturas se les asignó el mismo nivel de iluminación, mientras que la zona central está más iluminada que las anteriores y las zonas de los BC con el nivel de iluminación más bajo de todos.

A continuación se muestra una parte del código que contiene la descripción del entorno correspondiente al LCE en forma matricial, donde los BC van verticales y los BA horizontales.

```

envm = zeros(7,9,6);

envm(:,:,1) = [2 2 2 2 1 2 2 2 2   %Matriz de zonas
               2 2 2 2 1 2 2 2 2   %1 es zona de movilidad
               2 2 2 2 1 2 2 2 2   %2 es altura
               2 1 1 1 1 1 1 1 2
               2 2 2 2 1 2 2 2 2
               2 2 2 2 1 2 2 2 2
               2 2 2 2 1 2 2 2 2];

envm(:,:,2) = [0 0 0 0 1 0 0 0 0   %Matriz de pared arriba
               0 0 0 0 0 0 0 0 0
               0 0 0 0 0 0 0 0 0
               0 0 0 0 0 0 0 0 0
               0 0 0 0 0 0 0 0 0
               0 0 0 0 0 0 0 0 0
               0 0 0 0 0 0 0 0 0];

envm(:,:,3) = [0 0 0 0 0 0 0 0 0   %Matriz de pared abajo
               0 0 0 0 0 0 0 0 0
               0 0 0 0 0 0 0 0 0
               0 0 0 0 0 0 0 0 0
               0 0 0 0 0 0 0 0 0
               0 0 0 0 0 0 0 0 0
               0 0 0 0 1 0 0 0 0];

envm(:,:,4) = [0 0 0 0 1 1 0 0 0   %Matriz de pared a la izquierda
               0 0 0 0 1 1 0 0 0
               0 0 0 0 1 1 0 0 0
               0 0 0 0 0 0 0 0 0
               0 0 0 0 1 1 0 0 0
               0 0 0 0 1 1 0 0 0
               0 0 0 0 1 1 0 0 0];

envm(:,:,5) = [0 0 0 1 1 0 0 0 0   %Matriz de pared a la derecha
               0 0 0 1 1 0 0 0 0
               0 0 0 1 1 0 0 0 0
               0 0 0 0 0 0 0 0 0
               0 0 0 1 1 0 0 0 0
               0 0 0 1 1 0 0 0 0
               0 0 0 1 1 0 0 0 0];

Icen = 118;      %Luz en el centro
Ic = Icen-20;   %Luz en brazo cerrado
Io = Icen-10;   %Luz en brazo abierto
IC = Icen/500; IO = Ic/500; I = Io/500;

```

```

envm(:, :, 6) = [I I I I IO I I I I %Matriz de Iluminación
                I I I I IO I I I I
                I I I I IO I I I I
                I I I I IC I I I I
                I I I I IO I I I I
                I I I I IO I I I I
                I I I I IO I I I I];

```

4.2. DESCRIPTORES DE ENTORNO Y VARIABLES NEUROFUNCIONALES – ANATÓMICAS DEL MODELO ANA

Para describir cómo la rata interpreta la información que obtiene del entorno a través de los ojos, y del tacto de las vibrisas, cuerpo y cola, se definieron 16 parámetros, divididos en 4 grupos de 4 parámetros de acuerdo a la dirección en que se perciben y al estímulo al que corresponden de la siguiente manera:

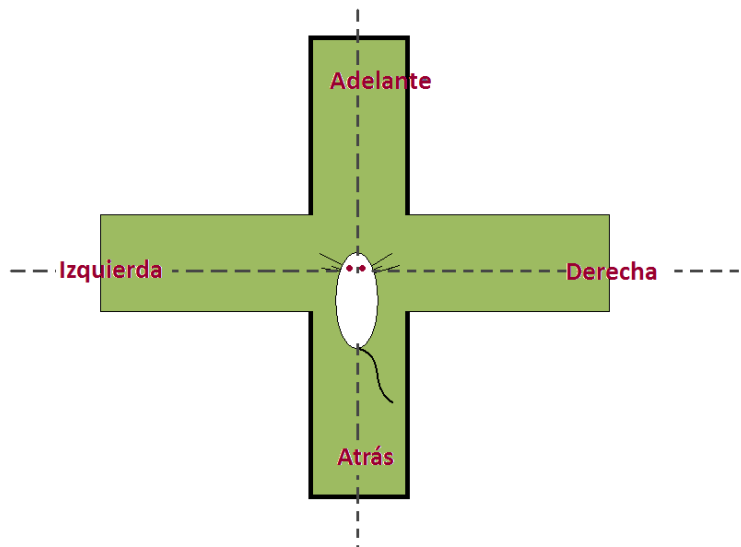


Fig 14. División del campo de visión de acuerdo a la dirección de la cabeza

ADELANTE:

1. Densidad de Iluminación Adelante (fid)
2. Densidad de Paredes Adelante (fwd)
3. Densidad de Alturas Adelante (fhd)
4. Densidad de Novedad Adelante (fnd)

ATRÁS:

5. Densidad de Iluminación Atrás (bid)
6. Densidad de Paredes Atrás (bwd)
7. Densidad de Alturas Atrás (bhd)
8. Densidad de Novedad Atrás (bnd)

IZQUIERDA:

9. Densidad de Iluminación Izquierda (lid)
10. Densidad de Paredes Izquierda (lwd)
11. Densidad de Alturas Izquierda (lhd)
12. Densidad de Novedad Izquierda (lnd)

DERECHA:

13. Densidad de Iluminación Derecha (rid)
14. Densidad de Paredes Derecha (rwd)
15. Densidad de Alturas Derecha (rhd)
16. Densidad de Novedad Derecha (rnd)

Donde la convención de las siglas está con palabras en inglés para facilitar así la distinción de términos:

F – Forward (Adelante)

B – Backward (Atrás)

L – Left (Izquierda)

R – Right (Derecha)

I – Illumination (Iluminación)

W – Wall (Pared)

H – Height (Altura)

N – Novelty (Novedad)

D – Density (Densidad)

La selección de estos parámetros se basó en el deseo de unificar en el modelo computacional los principales estímulos responsables del comportamiento aversivo que presenta la rata en la exploración del LCE reportados hasta la fecha por la literatura y que se encuentran explicados en mayor detalle en la sección 2.1.2 sobre los estímulos responsables de la aversión de la rata hacia los BA.

En el modelo ANA, las densidades son obtenidas como el resultado del procesamiento de la información que se proyecta de los sentidos hacia la corteza visual y somatosensitiva. Se considera una simplificación de las vías neurales por las cuales la información de carácter visual y sensitiva es decodificada por el cerebro y obtiene de esta las características relevantes. De esta manera, se comprimieron todas aquellas zonas encargadas del procesamiento de la información visual en un único algoritmo denominado Corteza Visual (esta compresión es notoria, partiendo del hecho de que el cerebro de la rata tiene al menos dos cortezas visuales, la primaria y la secundaria). De igual forma se comprimieron todas aquellas zonas encargadas del procesamiento de la

información sensitiva en un único algoritmo denominado Corteza Somatosensitiva (de nuevo se aclara que el cerebro de la rata tiene al menos dos cortezas somatosensitivas, la primaria y la secundaria). Si se desea ver lo anterior en más detalle ver la sección 2.3 sobre la neuroanatomía de la rata.

La definición matemática de estas densidades se planteó así;

$$d = \frac{\sum_{i=1}^N p_i}{N}$$

Siendo N la cantidad de zonas percibidas por la rata dentro de su campo de visión para una dirección específica o que se encuentran cercanas a esta para sentir las por tacto, y p que corresponde de acuerdo al tipo de densidad a:

- Magnitud de iluminación de la zona i, para densidad de iluminación
- Magnitud de la novedad de la zona i, para densidad de novedad
- 1 si hay pared o 0 si no hay en la zona i, para densidad de paredes
- 1 si hay altura o 0 si no hay en la zona i, para densidad de alturas

Con este planteamiento matemático se esperaría que cada una de las densidades dieran valores numéricos de modo que: altos valores de la densidad de iluminación (id) correspondieran a zonas muy iluminadas, y bajos valores a zonas poco iluminadas; altos valores de la densidad de paredes (wd) correspondieran a zonas rodeadas de paredes, y bajos valores a zonas con grandes campos abiertos; altos valores de la densidad de alturas (hd) correspondieran a zonas rodeadas de alturas, y bajos valores a zonas con pocas alturas o ninguna; altos valores de la densidad de novedad (nd) correspondieran a zonas muy visitadas por la rata, y bajos valores a zonas poco visitadas.

Además de lo anterior, las densidades se normalizaron para permitir un mejor entrenamiento de la RNA.

4.3. DESCRIPCIÓN DE LA RNA

Se creó un PMC con 16 nodos de entradas, uno por cada valor de densidad, interconectados con una capa oculta cuya cantidad de neuronas es equivalente a la mitad entera del número de ejemplos de entrenamiento. El anterior criterio se determinó de forma heurística. Por último se diseñó la capa de salida con 4 neuronas, cada una representando el nivel de motivación de la rata virtual por dirigirse hacia cualquiera de las 4 posibilidades de dirección de movimiento: Adelante, Atrás, Izquierda y Derecha. Las neuronas ocultas y de salida tienen como función de transferencia *logsig*, esto con la finalidad de que los valores numéricos de salida siempre estuvieran entre 0 y 1, y para brindar el suficiente nivel de no-linealidad que el problema requiere.

Para el entrenamiento de la RNA se utilizó el algoritmo *resilient backpropagation* por los criterios anteriormente mencionados en el marco teórico, además de su poco uso de memoria en su ejecución, alta rapidez y estadística de convergencia. Mediante la introducción de conjuntos de ejemplos de entrenamiento, cada uno compuesto de vectores de 16 elementos formados por las densidades, se le enseñó a las RNA a obtener 4 diferentes patrones de salida representadas por los vectores $[1, 0, 0, 0]$, $[0, 1, 0, 0]$, $[0, 0, 1, 0]$, $[0, 0, 0, 1]$, que en cuanto a la decisión de movimiento equivalen a Adelante, Atrás, Izquierda y Derecha.

No se le enseñó a la RNA a tomar la decisión de permanecer quieta, representada mediante el vector $[0, 0, 0, 0]$, con el fin de mantener al mínimo la cantidad de ejemplos de entrenamiento, y por consecuencia mantener al mínimo la cantidad de

neuronas ocultas. Con esto se especifica que la RNA cumple como función elegir hacia que dirección desea moverse la rata virtual, más no puede decidir si desea permanecer en una determinada zona o no.

Un determinado conjunto de entrenamiento se componía de dos matrices de datos. La primera que contenía vectores que representaban los estímulos que percibía la rata en una determinada posición dada por la secuencia obtenida de los datos crudos de los experimentos con ratas realizados por el Grupo de NYC UIS-UPB. Estos estímulos correspondían a los respectivos niveles de iluminación, la cantidad de paredes que rodeaban una zona, la cantidad de alturas que se hallaban en una zona y por último el nivel de novedad que representaba la zona para la memoria espacial de la rata. La segunda contenía las respectivas direcciones de movimiento que había ejecutado la rata real durante la exploración del LCE.

La selección del criterio de parada del entrenamiento de la RNA requirió la implementación de RNA ya entrenadas bajo el criterio de parada arbitrario de haber alcanzado un determinado número de epochs y posteriormente analizar su aprendizaje a través de curvas ROC a partir de la definición de un diagnóstico de aprendizaje que involucrara el cumplimiento de las desigualdades expresadas en la sección 2.1.3 y con esto encontrar el criterio de parada necesario. El anterior análisis se encuentra explicado en el capítulo 5 de implementación del modelo.

4.4. FUNCIONAMIENTO DEL MODELO

Para ilustrar mejor el funcionamiento computacional del modelo se mostrará un ejemplo numérico a la par de la explicación, donde se puede observar paso a paso como la información recolectada del entorno a través de los sentidos de la rata

virtual es procesada por su cerebro virtual para así decidir hacia donde se moverá dentro del LCE.

Lo primero que se define es el entorno en el cual se ubicará la rata, que por defecto es el LCE explicado en la sección 4.1 de interacción del autómata con el entorno. Posteriormente, el usuario sitúa la rata en una determinada posición a través del vector *rp* y define la dirección de la cabeza con el vector *hed*. (Para profundizar en las definiciones de los términos computacionales aquí utilizados, remitirse al Anexo A donde se explican de forma detallada los algoritmos).

rp =

4 5

hed =

-1 0

Una vez se ejecuta la simulación, el flujo de información ocurre de la siguiente manera: los ojos virtuales captan la información del entorno a través de la matriz *envm* y eliminan de esta todo elemento de matriz que se encuentre situado detrás de una pared, dejando solo los elementos que esta puede ver.

envm(:, :, 1) =

0	0	0	2	1	2	0	0	0
0	0	0	2	1	2	0	0	0
2	2	2	2	1	2	2	2	2
2	1	1	1	1	1	1	1	2
2	2	2	2	1	2	2	2	2
0	0	0	2	1	2	0	0	0
0	0	0	2	1	2	0	0	0

envm(:, :, 2) =

0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

envm(:, :, 3) =

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0

envm(:, :, 4) =

0	0	0	0	1	1	0	0	0
0	0	0	0	1	1	0	0	0
0	0	0	0	1	1	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	0	0	0
0	0	0	0	1	1	0	0	0
0	0	0	0	1	1	0	0	0

envm(:, :, 5) =

0	0	0	1	1	0	0	0	0
0	0	0	1	1	0	0	0	0
0	0	0	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	1	1	0	0	0	0
0	0	0	1	1	0	0	0	0
0	0	0	1	1	0	0	0	0

```
envm(:, :, 6) =
```

```
      0      0      0  0.216  0.196  0.216      0      0      0
      0      0      0  0.216  0.196  0.216      0      0      0
0.216  0.216  0.216  0.216  0.196  0.216  0.216  0.216  0.216
0.216  0.216  0.216  0.216  0.236  0.216  0.216  0.216  0.216
0.216  0.216  0.216  0.216  0.196  0.216  0.216  0.216  0.216
      0      0      0  0.216  0.1960 0.2160      0      0      0
      0      0      0  0.2160 0.1960 0.2160      0      0      0
```

De igual forma, las vibrisas, cuerpo y cola de la rata virtual, que forman el sentido del tacto de esta, obtienen de la matriz *envm* aquellos elementos de matriz que se encuentran cercanos a la posición de la rata, representando con esto aquello que la rata virtual puede tocar.

```
envm(:, :, 1) =
```

```
      3      1      3
      1      1      1
      3      1      3
```

```
envm(:, :, 2) =
```

```
      0      0      0
      0      0      0
      0      0      0
```

```
envm(:, :, 3) =
```

```
      0      0      0
      0      0      0
      0      0      0
```

```
envm(:, :, 4) =
```

```
      0      0      0
      0      0      0
      0      0      0
```

```
envm(:, :, 5) =
```

```
    0    0    0
    0    0    0
    0    0    0
```

Posteriormente, esta información cruda es enviada a sus respectivas cortezas para su procesamiento y extracción de características. A la corteza visual se dirige la información de los ojos, y a la corteza somatosensitiva la información del tacto. Cada una de estas cortezas procesa los datos y retornan dos tipos de vectores: el primero llamado *acc* (access) con 4 elementos que contiene los permisos de acceso a otras zonas, o dicho de otra forma, la posibilidad física de movimiento en una dirección; el segundo llamado *param* (parameters) con 12 elementos contiene los valores de las densidades de iluminación (que valen 0 para la corteza somatosensitiva), densidades de paredes y densidades de alturas.

```
vparam =
```

```
    0.2135    0.2135    0.2172    0.2172    0.7500    0.7500    0.7647    0.7647 ...
    0.4375    0.4375    0.1765    0.1765
```

```
vacc =
```

```
    1    1    1    1
```

```
sparam =
```

```
    0    0    0    0    0    0    0    0
```

```
sacc =
```

```
    1    1    1    1
```

La información que retornan las cortezas visual y somatosensitiva es enviada a la corteza asociativa multisensorial, que pondera el aporte de cada uno de los sentidos y retorna como salida nuevos vectores *param* y *acc*.

tparam =

```
0.2135  0.2135  0.2172  0.2172  0.7500  0.7500  0.7647  0.7647 ...
0.4375  0.4375  0.1765  0.1765
```

tacc =

```
1      1      1      1
```

Mientras tanto y de forma paralela, la memoria espacial de la rata virtual procesa la información que le llega de los ojos y obtiene tanto una nueva matriz *mem* de memoria espacial, como el vector *nd* que contiene las densidades de novedad.

mem =

```
0      0      0      0      0.5      0      0      0      0
0      0      0      0      0.5      0      0      0      0
0      0      0      0      0.5      0      0      0      0
0      0.5    0.5    0.5    1.0      0.5    0.5    0.5    0
0      0      0      0      0.5      0      0      0      0
0      0      0      0      0.5      0      0      0      0
0      0      0      0      0.5      0      0      0      0
```

nd =

```
0.0017  0.0017  0.0017  0.0017
```

Posteriormente, el vector *param* de la corteza asociativa y el vector *nd* de la memoria espacial se concatenan para formar el vector *input* que contiene los valores de entrada para la RNA.

```
input =  
  
    0.2135  0.2135  0.2172  0.2172  0.7500  0.7500  0.7647  0.7647 ...  
    0.4375  0.4375  0.1765  0.1765  0.0017  0.0017  0.0017  0.0017
```

Una vez simulada la RNA, esta retorna como salida un vector de 4 elementos con valores entre 0 y 1, que corresponden a las ganas de la rata virtual de moverse en una dirección (adelante, atrás, izquierda o derecha). Estos valores indican el grado de decisión que presenta el animal en la selección de una determinada dirección hacia la cual ejecutar su próximo movimiento, siendo mayor la intención a medida que aumenta el valor numérico.

```
outnet =  
  
    0.3432  0.7344  0.0000  0.0000
```

Pero para que una volición se convierta en acción, primero esta debe superar un umbral *minlim* que permite discriminar por debajo de él las voliciones inhibidas y por encima de él las aceptadas [1]. Es por esto que los elementos del vector de salida de la RNA que se encuentren por debajo del umbral se hacen 0, mientras que los que se encuentran por encima entran a competir, ganando el mayor de ellos, solo si este se encuentra un determinado valor *ind* por encima del otro, el anterior hecho se denominó “brecha de incertidumbre”, y representa la duda en la toma de dos decisiones que no tienen una diferencia significativa en su nivel de excitación. Una vez se tiene en cuenta lo anterior, si existe una volición con mayor grado de excitación que las demás esta gana y se cambia su valor a 1, mientras que las demás se cambian a 0. Ahora, si existe incertidumbre, la selección entre ellas para la toma de la decisión final se realiza de forma aleatoria.

```
minlim =  
  
    0.5
```

```
ind =  
    0.1
```

```
outnet =  
    0  1  0  0
```

El resultado obtenido corresponde a un movimiento hacia atrás, pero este debe ser transformado de SRR a SRL, con lo cual el movimiento es hacia abajo.

```
rmov =  
    1    0
```

Una vez se ha tomado la decisión final, se modifica el vector de posición de la rata *rp* por su nuevo valor, y esto retroalimenta el inicio del proceso, y así sucesivamente se ejecuta cada iteración hasta la cantidad de decisiones que el usuario haya asignado para la simulación.

```
rp =  
    5    5
```

```
hed =  
    1    0
```

El tiempo por iteración empleado en las simulaciones se definió constante a través de toda la simulación, de manera que cada zona explorada por la rata virtual tiene igual valor de tiempo de permanencia. La definición matemática corresponde a;

$$tis = \frac{300}{ns}$$

donde t_{is} es el tiempo por iteración de simulación y n_s el número de posiciones por secuencia de rata real, los cuales variaron entre 66 y 140 para las 17 ratas reales que se emplearán en la implementación del modelo, de lo cual se obtienen t_{is} que varían entre 2.1428 s y 4.5454 s al cambiar de rata virtual.

5. IMPLEMENTACIÓN DEL MODELO ANA

5.1. PROTOCOLO DE EXPERIMENTACIÓN

Se evaluaron 17 ratas *wistar* machos provenientes del bioterio de la Facultad de Salud de la UIS organizadas en tres grupos. Todos los animales 30 minutos antes de ser expuestos por primera vez al LCE, recibieron una inyección intraperitoneal de solución salina (0.9%), en un volumen de 10ml/Kg. Los experimentos fueron ejecutados bajo tres condiciones de luminosidad, alta (118 Lux), media (95 Lux) y baja (37 Lux), medida con un luxómetro estándar en el centro del LCE. Cada animal fue colocado en el centro del laberinto con la cabeza orientada hacia un BC permitiendo la exploración libre durante 5 minutos. Las sesiones fueron monitorizadas y filmadas en un sistema de TV-VHS para posterior registro y procesamiento de las secuencias de posiciones y de tiempos de permanencia utilizando un conjunto de programas computacionales (Prostcom) elaborados para tal fin [9].



Fig 15. Rata Wistar

5.2. BUSQUEDA DEL CRITERIO DE APRENDIZAJE

Para establecer si es posible argumentar un criterio de aprendizaje basado en los resultados de la RNA propuesta, y tomando como datos experimentales los obtenidos en la fase anterior, se entrenó un conjunto de 68 RNA con *Training*, 4 RNA por cada secuencia de datos (tanto posiciones como tiempos de permanencia en s) teniendo en cuenta que en el proceso de entrenamiento bajo el algoritmo de *backpropagation* se presenta una cierta variabilidad por ser este un método estocástico.

Se seleccionó como tipo de entrenamiento *Learning Analysis*, que permite obtener información del entrenamiento de las RNA por cada epoch, y como criterio de parada se debían alcanzar 100 epochs. Una vez obtenida la información, se generaron vectores de diagnósticos de aprendizaje con *LeAn* con los que se hicieron pruebas ROC [6,10,17,18,19] que permitieron validar la capacidad del mse para discriminar el aprendizaje de las RNA [34] del CERLCE frente a los parámetros comportamentales propuestos de TB y NEB tanto absolutos como porcentuales.

Para clasificar los resultados de las pruebas ROC entre válidos y no válidos, se establecieron los siguientes criterios de resultado fallido [6]:

- Si la casilla de valor de ROC está en blanco (sin valor alguno)
- Si el valor de ROC dio menor a 0.7
- Si el valor de la sensibilidad de la prueba dio menor a 0.8

Los resultados se muestran a continuación:

Tabla 5. Resultados de las curvas ROC y prueba de reproducibilidad para los datos de las variables de los animales virtuales para 100 epochs. Los criterios para resultados fallidos de ROC son: ROC en blanco (sin valor), ROC < 0.7 y Sensibilidad < 0.8. Para medir la reproducibilidad se tomó como criterio errores estándar inferiores a 0.1.

		REPETICIONES	Blanco	ROC<0.7	Sens<0.8	tot	Reprod (%SD>.1)
		100 epochs	n	E1	2	2	6
E2	0			2	7	9	
E3	0			3	5	7	
E4	1			3	7	7	
%	E1		12	12	35	35	82
	E2		0	12	41	53	
	E3		0	18	29	41	
	E4		6	18	41	41	

En la tabla anterior se condensan los resultados, donde cada repetición (E1, E2, E3 y E4) se hizo con 17 datos, y las columnas de Blanco, ROC < 0.7 y Sens < 0.8 indican la cantidad de datos que al ser analizados con las pruebas ROC no cumplieron con alguno de estos criterios. La columna tot contiene la cantidad de resultados fallidos sin importar la razón. Las filas superiores muestran los valores absolutos mientras que las inferiores los valores porcentuales. Como se observa, cerca de la mitad de los resultados fallaron por alguna de las tres razones, indicando que la prueba no es consistente.

También se realizó una prueba simple de reproducibilidad de los puntos de corte entre los cuatro resultados obtenidos de un mismo conjunto de datos de rata real. Se seleccionó como criterio de resultado fallido un porcentaje de desviación estándar (%SD) mayor que 0.1. En la tabla se muestra que el 82% de los resultados dieron fallidos, indicando una mala reproducibilidad.

Como consecuencia de haber obtenido un alto porcentaje de datos con diagnósticos fallidos en ROC y una mala reproducibilidad, se propuso aumentar la cantidad de epochs para realizar el diagnóstico, planteándose que la poca cantidad de estos sería posiblemente la culpable del hecho anterior. Es por esto que el análisis se repitió entrenándose 34 RNA, 2 RNA por cada secuencia de datos y como criterio de parada se debían alcanzar 500 epochs. Los resultados se muestran a continuación.

Tabla 6. Resultados de las curvas ROC y prueba de reproducibilidad para los datos de las variables de los animales virtuales para 500 epochs. Los criterios para resultados fallidos de ROC son: ROC en blanco (sin valor), ROC < 0.7 y Sensibilidad < 0.8. Para medir la reproducibilidad se tomo como criterio errores estándar inferiores a 0.1.

		REPETICIONES	Blanco	ROC<0.7	Sens<0.8	tot	Reprod (%SD>.1)
500 epochs	n	E1	0,00	1,00	1,00	2,00	13,00
		E2	0,00	0,00	0,00	0,00	
	%	E1	0	6	6	12	76
		E2	0	0	0	0	

En esta ocasión el 12% de los resultados dieron fallidos, por lo que se puede determinar que los resultados de reproducibilidad que se obtengan son conformes al comportamiento de los datos y no a la cantidad de epochs, luego no es necesario realizar pruebas ulteriores con una mayor cantidad de epochs. Con un 76% de resultados que fallaron esta vez es posible concluir que no hay reproducibilidad en los puntos de corte de las pruebas ROC.

Si la variable propuesta mse tuviera la capacidad de discriminar el aprendizaje de del CERLCE por las RNA frente a los parámetros comportamentales propuestos de TB y NEB tanto absolutos como porcentuales, y si la evolución temporal del aprendizaje a través del entrenamiento presentara un buen punto de corte, se esperarían altos valores de ROC (ROC > 0.7), sensibilidad (S > 0.8) [6] y una alta

reproducibilidad de los puntos de corte, pero como se observa en los resultados, esto no fué lo que ocurrió, verificándose con ello que la evaluación del aprendizaje de las RNA bajo los criterios de NEBC > NEBA y TPBC > TPBA tomando como referencia el mse del entrenamiento falla.

5.3. VALIDACIÓN DEL MODELO BAJO LOS CRITERIOS DE NÚMERO DE ENTRADAS Y TIEMPOS DE PERMANENCIA EN BRAZOS

Se entrenó un conjunto de 17 RNA, 1 RNA por cada secuencia de datos, bajo el criterio de parada de haber alcanzado un determinado valor de mse durante su entrenamiento para luego realizar una serie de pruebas que permitirían validar el desempeño del modelo frente a los parámetros comportamentales propuestos de TB y NEB tanto absolutos como porcentuales. Este proceso se realizó variando el mse en los siguientes valores: 0.1 - 0.05 – 0.01 – 0.005 – 0.001, para obtener así un total de 75 RNA.

VARIABLES MEDIDAS

TBA: Tiempo de permanencia en los brazos abiertos del laberinto

TBC: Tiempos de permanencia en los brazos cerrados del laberinto

NEBA: Número de entradas a los brazos abiertos

NEBC: Número de entradas a los brazos cerrados

%TBA: Porcentaje del tiempo de permanencia en los brazos abiertos

%TBC: Porcentaje del tiempo de permanencia en los brazos cerrados

%EBA: Porcentaje de entradas a los brazos abiertos

%EBC: Porcentaje de entradas a los brazos cerrados

p: Significancia estadística. Se fijó el nivel de significancia en el 5%

RESULTADOS

Tabla 7. Resultados de las correlaciones de Pearson entre los datos de las variables de los animales reales y virtuales para cada uno de los parámetros comportamentales variando el valor de mse entre 0.1 – 0.05 – 0.01 – 0.005 – 0.001. Estos indican buenos valores de correlación en todos los parámetros para mse de 0.005 y 0.001.

MSE	TBC	TBA	%TBC	%TBA	NEBC	NEBA	%EBC	%EBA	NC
0.1	,345	,370	,357	,373	-,063	,393	,263	,350	-,093
0.05	,385	,266	,400	,264	,081	,139	,516	,516	-,237
0.01	,746	,135	,464	,165	,595	,854	,750	,750	,782
0.005	,698	,749	,741	,750	,837	,795	,836	,836	,834
0.001	,847	,919	,935	,921	,994	,983	,988	,988	,998
	± .482	critical value .05 (two-tail)							
	± .606	critical value .01 (two-tail)							
17 sample size									

Tabla 8. Resultados de ANOVA de una vía de Kruskal Wallis seguida de método de Dunn para comparaciones múltiples entre los datos de las variables de los animales reales y virtuales para cada uno de los parámetros comportamentales variando el valor de mse entre 0.1 – 0.05 – 0.01- 0.005 – 0.001. Se encontraron diferencias significativas para mse de 0.1 y 0.05 respecto a la rata real para la gran mayoría de parámetros ($p < 0.001$).

	H (5 GL)	p	Dunn's
TBC	10.150	0.071	
TBA	40.707	<0.001	0.1 - 0.05
%TBC	10.762	0.056	
%TBA	40.602	<0.001	0.1 - 0.05
NEBC	55.829	<0.001	0.1 - 0.05
NEBA	51.550	<0.001	0.1 - 0.05
%EBC	57.984	<0.001	0.1
%EBA	30.559	<0.001	0.1
NC	57.984	<0.001	0.1 - 0.05

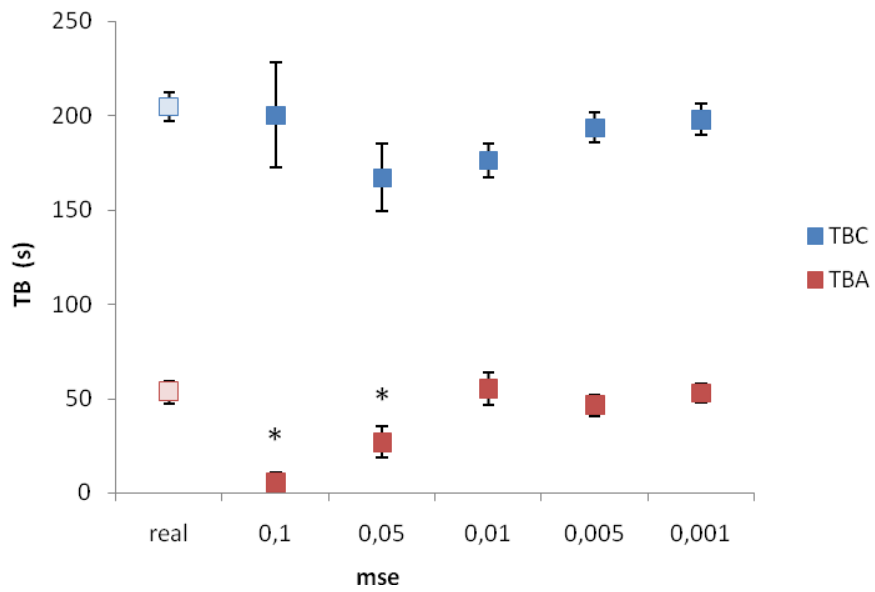


Fig 16. Media \pm error estándar de los tiempos de permanencia en los brazos del laberinto. (Rojo – Brazos Abiertos, Azul – Brazos Cerrados). *Valor significativamente diferente del TBA de las ratas reales (ANOVA de una vía de Kruskal Wallis seguida de método de Dunn para comparaciones múltiples, $p < 0.001$)

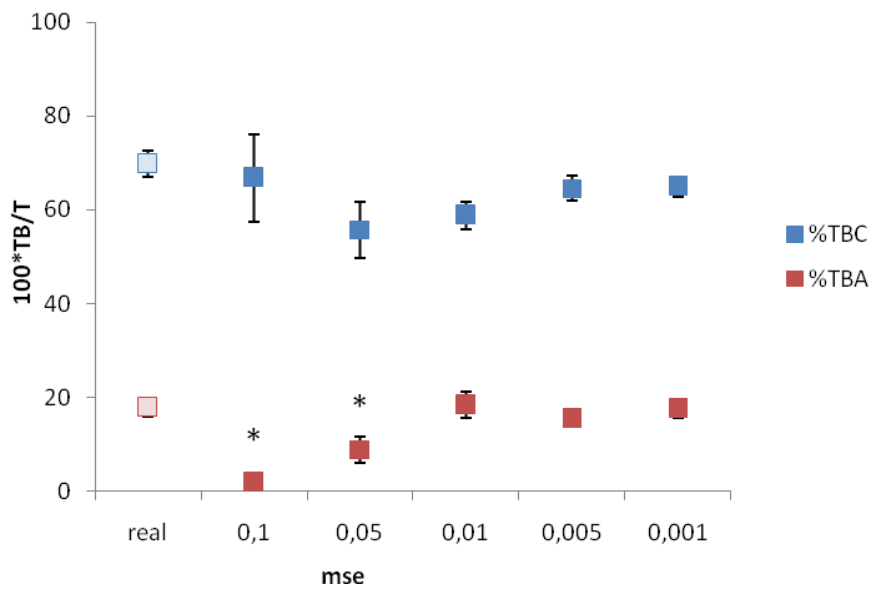


Fig 17. Media \pm error estándar del porcentaje de los tiempos de permanencia en los brazos del laberinto. (Rojo – Brazos Abiertos, Azul – Brazos Cerrados). *Valor significativamente diferente del %TBA de las ratas reales (ANOVA de una vía de Kruskal Wallis seguida de método de Dunn para comparaciones múltiples, $p < 0.001$)

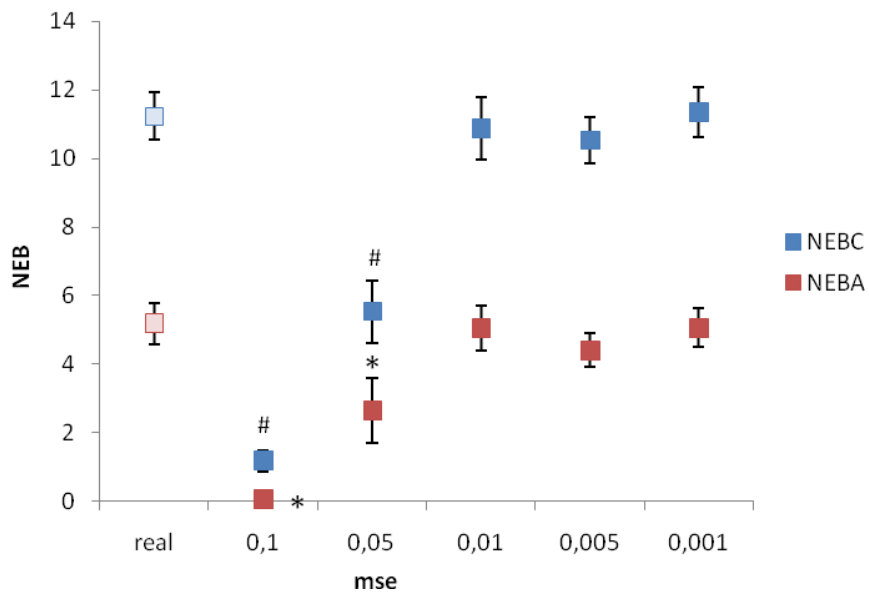


Fig 18. Media \pm error estándar de las entradas en los brazos del laberinto. (Rojo – Brazos Abiertos, Azul – Brazos Cerrados). *Valor significativamente diferente del NEBA de las ratas reales, #Valor significativamente diferente del NEBC de las ratas reales (ANOVA de una vía de Kruskal Wallis seguida de método de Dunn para comparaciones múltiples, $p < 0.001$)

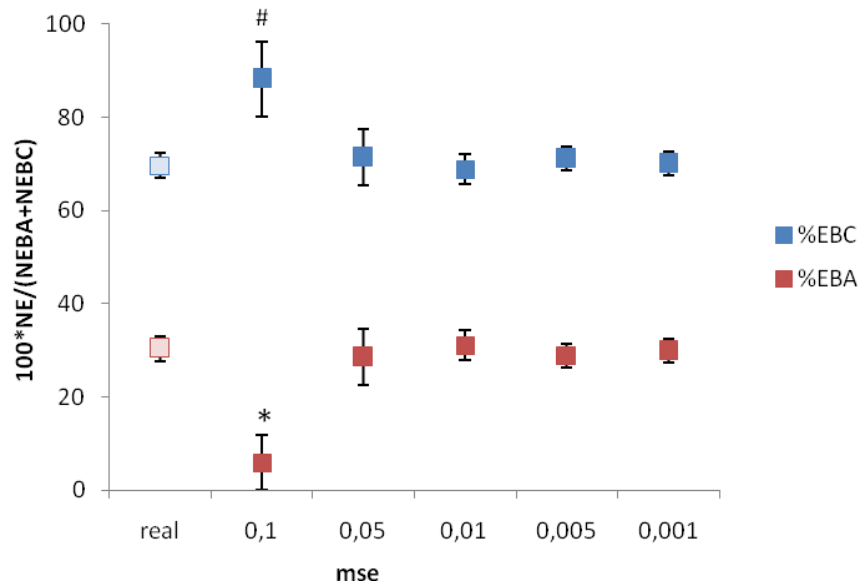


Fig 19. Media \pm error estándar del porcentaje de entradas en los brazos del laberinto. (Rojo – Brazos Abiertos, Azul – Brazos Cerrados). *Valor significativamente diferente del %EBA de las ratas reales, #Valor significativamente diferente del %EBC de las ratas reales (ANOVA de una vía de Kruskal Wallis seguida de método de Dunn para comparaciones múltiples, $p < 0.001$)

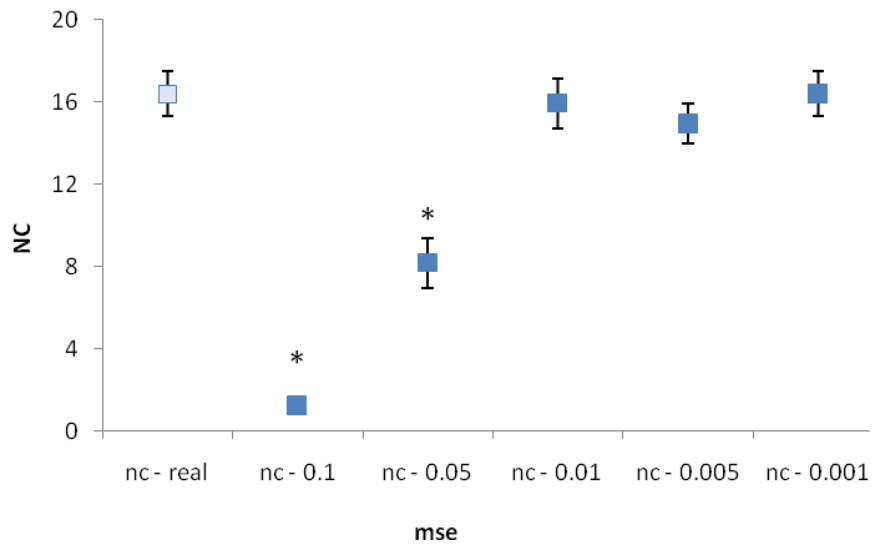


Fig 20. Media \pm error estándar del número de cruzamientos de la zona central del laberinto. *Valor significativamente diferente del NC de las ratas reales (ANOVA de una vía de Kruskal Wallis seguida de método de Dunn para comparaciones múltiples, $p < 0.001$)

La tabla 7 muestra las correlaciones de Pearson para los parámetros comportamentales entre los datos de las ratas reales y virtuales para valores de mse de 0.1 - 0.05 - 0.01 - 0.005 - 0.001. Se obtuvieron malas correlaciones en todos los parámetros para el mse de 0.1, de igual manera para el mse de 0.05 con excepción de los %EBC y %EBA, que obtuvieron buenas correlaciones. A partir de 0.01 se observan buenas y excelentes correlaciones para la gran mayoría de los parámetros. Para el mse de 0.01 se obtuvieron excelentes correlaciones para TBC, NEBA, %EBC, %EBA y NC, una buena correlación para NEBC y malas para TBA, %TBC y %TBA.

Para los mse de 0.005 y de 0.001 las correlaciones para todos los parámetros comportamentales fueron excelentes e incluso se destaca que para el mse de 0.001 todos los valores están por encima 0.91 excepto por TBC (0.847).

Los parámetros comportamentales en el laberinto (TBA, TBC, %TBA, %TBC, NEBA, NEBC, %EBA, %EBC y NC) obtenidos de la RNA (ratas virtuales) fueron comparados con los comportamientos medidos directamente de los animales reales utilizando ANOVA de una vía de Kruskal Wallis seguida de método de Dunn para comparaciones múltiples (Tabla 8). De esta comparación se obtuvieron diferencias significativas respecto a los valores obtenidos en ratas reales para todos los parámetros excepto para TBC y %TBC, con una significancia inferior a 0.001 para el mse de 0.1 y diferencias significativas para TBA, %TBA, NEBC, NEBA y NC con una significancia inferior a 0.001 para el mse de 0.05. Para todos los valores de mse no se obtuvieron diferencias significativas para los parámetros de TBC ($p = 0.071$) y %TBC ($p = 0.056$).

En las graficas 16, 17, 19 y 20 se muestran los parámetros comportamentales para cada uno de los brazos del laberinto, en las que se puede apreciar que para todo mse empleado se cumplen las desigualdades propuestas en la sección 2.1.3

relacionada con los criterios de caracterización del CERLCE, tanto para valores absolutos como porcentuales, validando con estos resultados la capacidad del modelo de reproducir el CERLCE bajo los parámetros de NEB y TB del laberinto.

5.3.1. VALIDACIÓN DEL MODELO PARA VALORES ALTOS DE MSE

Como se puede observar en las gráficas de la sección anterior, para todo el rango de mse desde 0.1 hasta 0.001 se cumple que;

$$NEBC > NEBA \quad TBC > TBA \quad \%EBC > \%EBA \quad \%TBC > \%TBA$$

Análisis que refuerza la conclusión obtenida con las pruebas ROC, en la que se considera que el cumplimiento de las anteriores desigualdades es independiente del mse al que haya alcanzado la RNA en el entrenamiento. Aun así, sería posible creer que esto se deba a que un mse de 0.1 ya es lo suficientemente bajo como para que este comportamiento se presente. Es por esto que se cuestionó si el hecho anterior también se presentaría para valores altos de mse. Para esto se entrenó un conjunto de 17 RNA, 1 RNA por cada secuencia de datos, para un único epoch, con lo cual se garantizó haber obtenido el mayor valor de mse posible para el determinado entrenamiento, que en promedio corresponde a 0.4.

Posteriormente se obtuvieron los parámetros comportamentales propuestos de TB y NEB tanto absolutos como porcentuales, para luego compararlos entre ellos y observar si cumplían o no con las desigualdades. Los resultados se resumen en el siguiente gráfico.

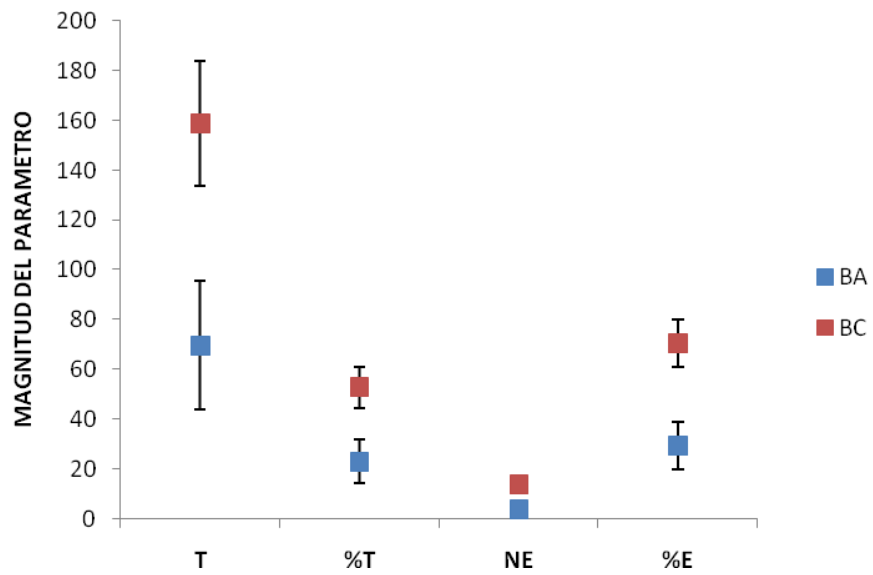


Fig 21. Media \pm error estándar de parámetros comportamentales de tiempos de permanencia (T), porcentaje de tiempos de permanencia (%T), número de entradas (NE) y porcentaje de entradas a los brazos del laberinto (%E) para mse de 0.4. (Rojo – Brazos Cerrados, Azul – Brazos Abiertos).

De este se concluye que el cumplimiento de las desigualdades ocurre para todo el dominio de mse, indicando que estas no sirven como criterio de aprendizaje de las RNA. Es por esto que retomando el análisis de la sección anterior, se puede proponer como buen criterio de aprendizaje aquel en el que la RNA obtenga buenos resultados de correlación y no presente diferencias significativas en ANOVA, hecho que se observa a partir de un mse de 0.01.

5.4. CORRELACIONES ENTRE RATA REAL Y RATA VIRTUAL CON MSE 0.001

Uno de los valores más bajos de mse con los cuales se entrenaron ratas virtuales fue de 0.001. La tasa de convergencia de las RNA que alcanzan este valor es muy inferior a la de valores de mse más altos, indicando con esto que el esfuerzo que realiza el entrenamiento para alcanzar este nivel de error es mucho más grande, y que niveles inferiores de mse podrían ser aún mucho más difíciles de obtener, es por esto que para el propósito de evaluación del modelo este fué el valor de mse más bajo con el cual se realizaron pruebas.

Se esperaría que entre los parámetros comportamentales de las ratas virtuales y las reales existieran muy buenas correlaciones, ya que como anteriormente se verificó, a partir de un mse de 0.01 se obtienen valores promedios similares sin diferencias significativas como lo indicaron los resultados con ANOVA y las correlaciones Pearson anteriores. A continuación se muestran gráficas de correlación con sus respectivas ecuaciones de ajuste lineal y valores de R^2 .

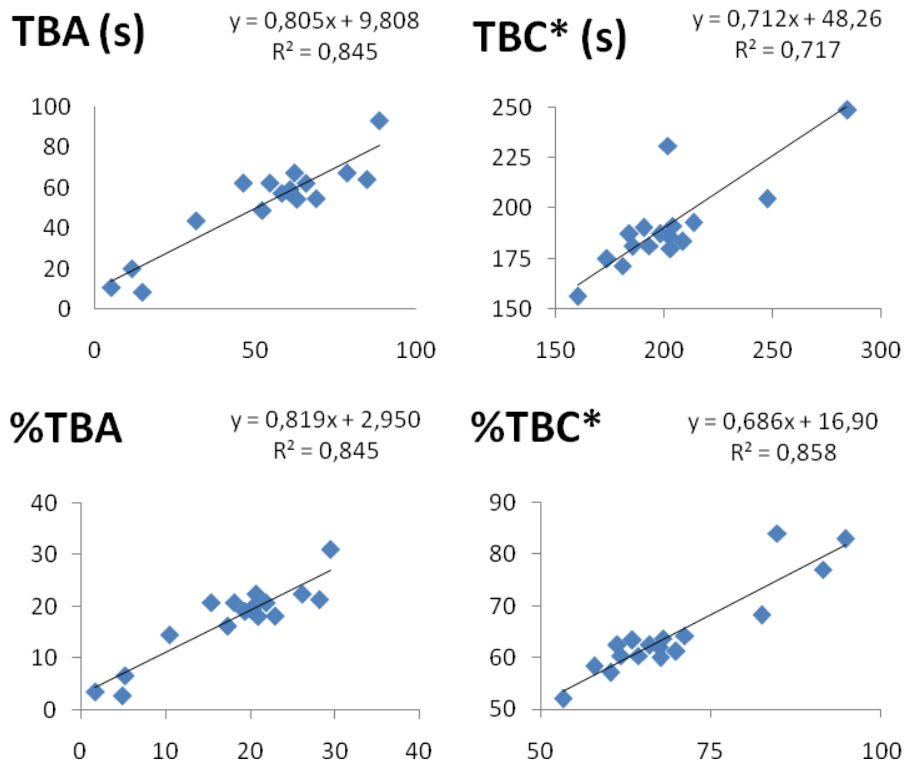


Fig 22. Correlaciones entre los datos de las variables de los animales reales y virtuales (mse 0.001) para parámetros comportamentales de tiempos de permanencia en BA (TBA) y BC (TBC), también para sus respectivos porcentajes (%TBA y %TBC). *No tan buenos ajustes lineales para TBC ($m = 0.7125$, $R^2 = 0.7176$) y %TBC ($m = 0.686$).

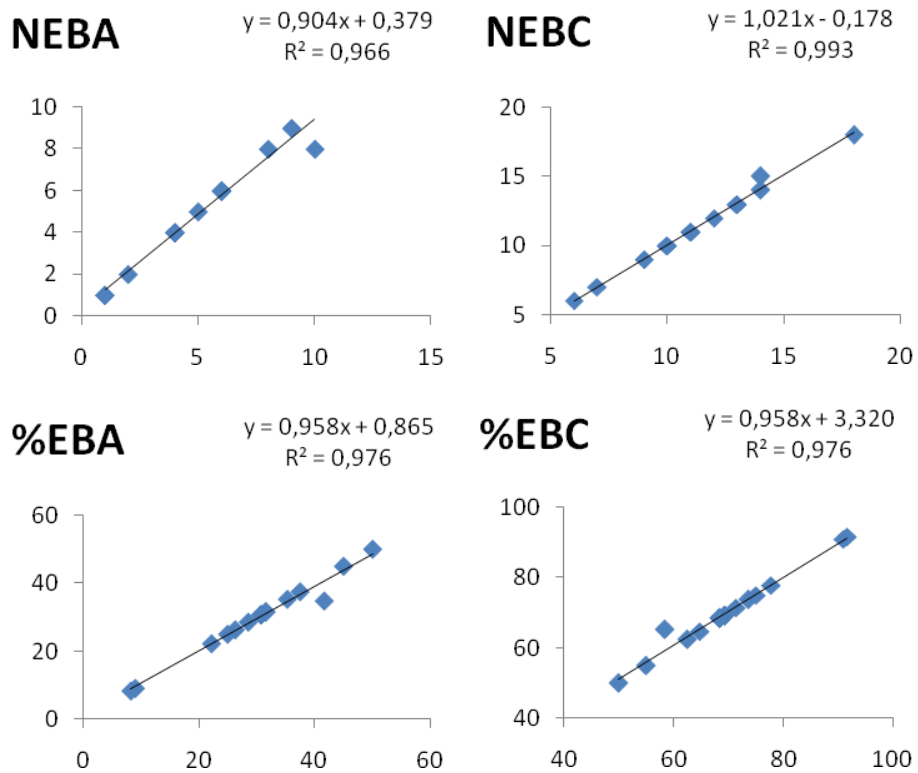


Fig 23. Correlaciones entre los datos de las variables de los animales reales y virtuales (mse 0.001) para parámetros comportamentales de número de entradas en BA (NEBA) y BC (NEBC), también para sus respectivos porcentajes (%EBA y %EBC). Se encontraron buenos ajustes lineales para todos los parámetros presentes.

En las figuras 22, 23 y tabla 7 se puede apreciar la alta correlación de los datos para BA y BC, sin embargo, como puede apreciarse en la figura 22, los ajustes para las variables TBC y %TBC no fueron tan buenos como los de las otras variables. Esta diferencia se atribuye a la asignación de grandes intervalos de tiempos por iteración de simulación, que tenían igual valor para cada zona explorada por la rata virtual en la simulación y que variaban entre 2.1428 s y 4.5454 s dependiendo de la rata virtual.

Para verificar la anterior hipótesis, se tomaron las secuencias de movimiento de los datos crudos de las ratas reales, y se reemplazaron sus tiempos de permanencia por zona por tiempos constantes según la ecuación del tis en la sección 4.4 sobre el funcionamiento del modelo. Los resultados obtenidos para TBA y TBC corresponden al mayor grado de precisión que se podría obtener con el entrenamiento de las RNA y que se obtendrían, hablando teóricamente, cuando el mse alcanzara un valor de cero.

Como se aprecia en la gráfica 24 y en las ecuaciones de los ajustes lineales, los valores de pendientes y de R^2 corresponden en muy buena proporción con los obtenidos para RNA entrenadas y que alcanzaron valores de mse de 0.001. Con esto se demuestra, primero que un valor de mse de 0.001 es equivalente en muy buena aproximación al máximo valor obtenido para el modelo, por lo que se considera innecesario ir más allá de este valor, ahorrando con ello costo computacional en tiempo. Como segunda conclusión, se puede decir que los valores de pendiente y de R^2 para las correlaciones anteriormente mostradas son los más precisos alcanzados por el modelo y que no son productos de otros factores, sino del modo de operación de este, por lo que si se deseara mejorar este aspecto, sería necesario modificar el modo de asignación de los tiempos de permanencia en las zonas.

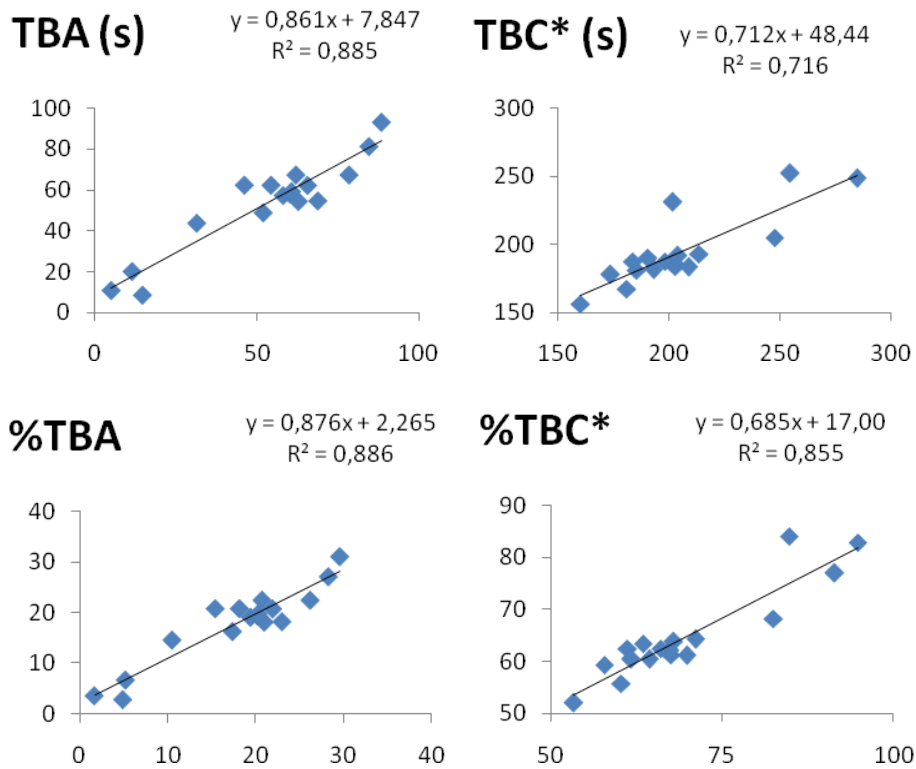


Fig 24. Correlaciones entre los datos de las variables de los animales reales para tiempos de decisión constantes y reales para parámetros comportamentales de tiempos de permanencia en BA (TBA) y BC (TBC), también para sus respectivos porcentajes (%TBA y %TBC). *No tan buenos ajustes lineales para TBC ($m = 0.7124$, $R^2 = 0.7166$) y %TBC ($m = 0.6854$).

5.5. ANÁLISIS E INTERPRETACIÓN DE DENSIDADES DE ILUMINACIÓN, PAREDES Y ALTURAS

Uno de los fundamentos del proyecto supone la posibilidad de describir la toma de decisión de movimiento de la rata durante la exploración libre de esta en el LCE a través de parámetros que representen los estímulos externos e internos del animal. A continuación se presentan gráficas del comportamiento de estos 16 descriptores del entorno propuestos (densidades, magnitud en el eje y) a través de la secuencia de 66 movimientos (eje x) obtenida de la experimentación con una rata bajo un nivel de iluminación alto de 118 Lux en el LCE, con el fin de visualizar en estas la capacidad de representación del fenómeno que las densidades puedan aportar.

La validez que aporta esta simulación a la capacidad del modelo ANA de reproducir el fenómeno se basa en que en esta simulación la rata virtual visitó todas las posiciones del LCE y estuvo con todas las diferentes posibilidades de dirección de la cabeza, con lo cual se abarcó todos los posibles valores que podían tomar las densidades de iluminación, paredes y alturas.

El eje x contiene la secuencia de posiciones avanzando temporalmente de izquierda a derecha. A su vez, el eje y muestra los valores numéricos de las densidades con su dominio en [0,1].

DENSIDADES DE ILUMINACION ID

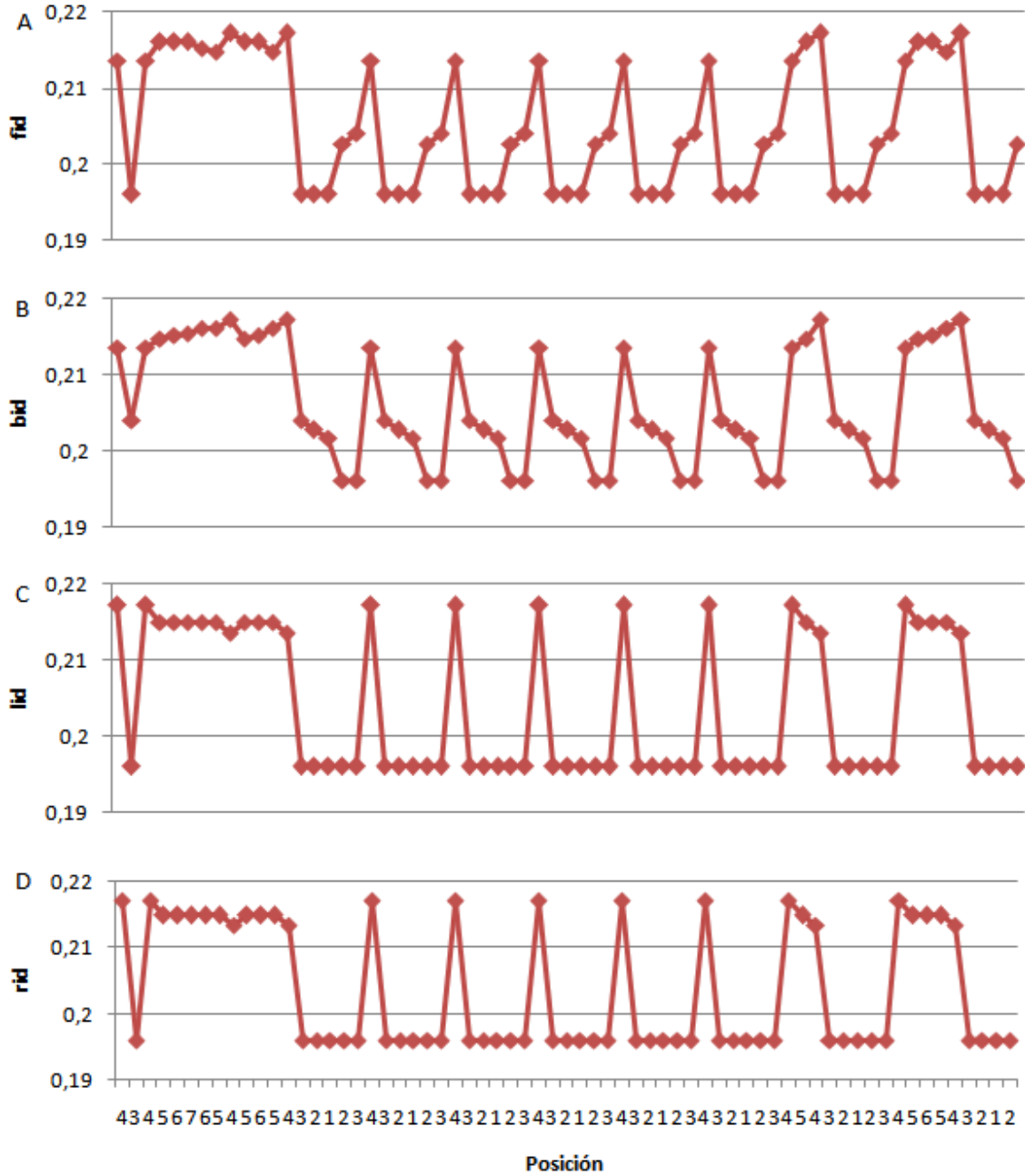


Fig 25. Evolución temporal de la densidad de iluminación de acuerdo a la dirección de la cabeza.
A. Adelante, B. Atrás, C. Izquierda, D. Derecha

DENSIDADES DE PAREDES WD

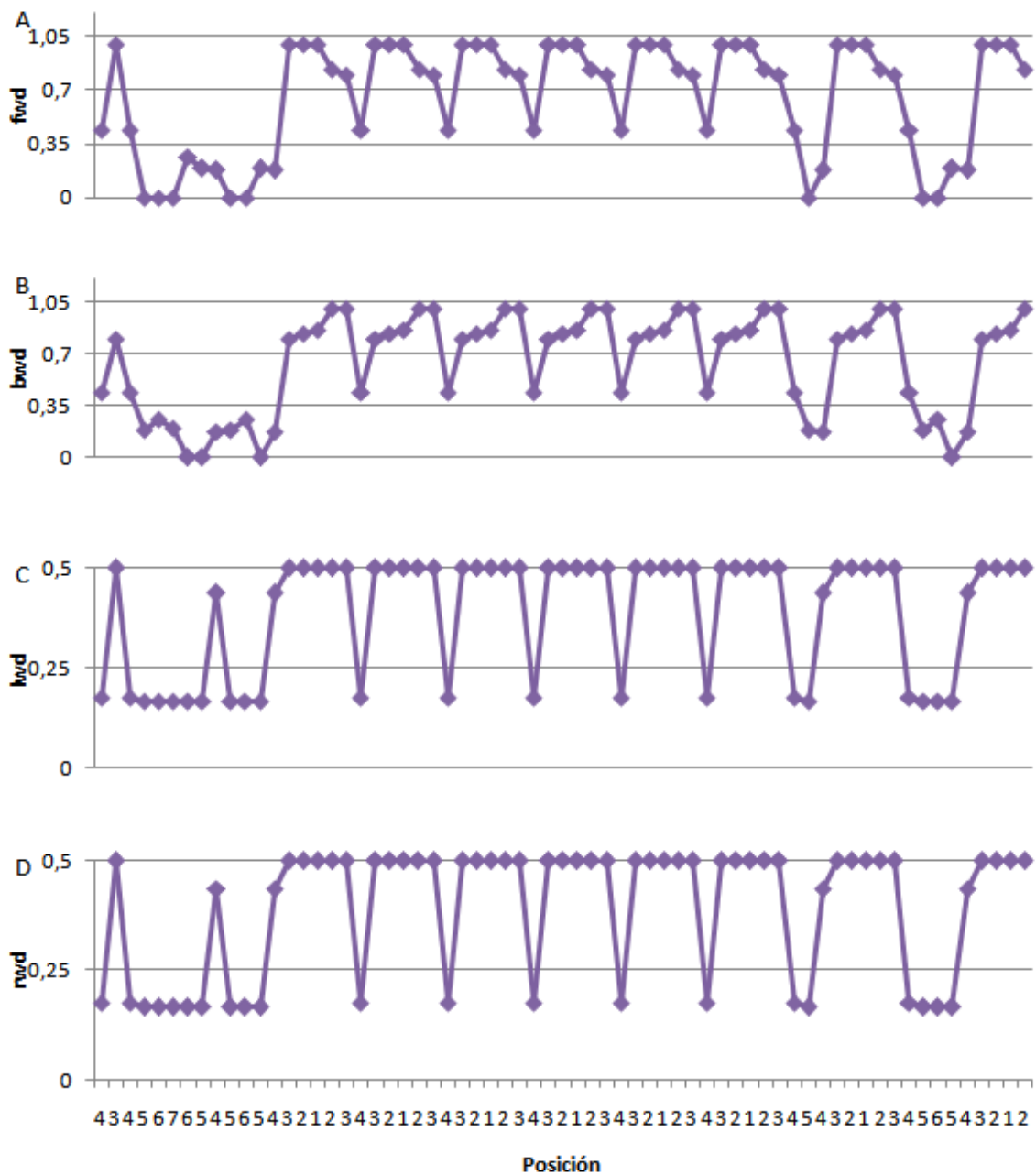


Fig 26. Evolución temporal de la densidad de paredes de acuerdo a la dirección de la cabeza.
A. Adelante, B. Atrás, C. Izquierda, D. Derecha

DENSIDADES DE ALTURA HD

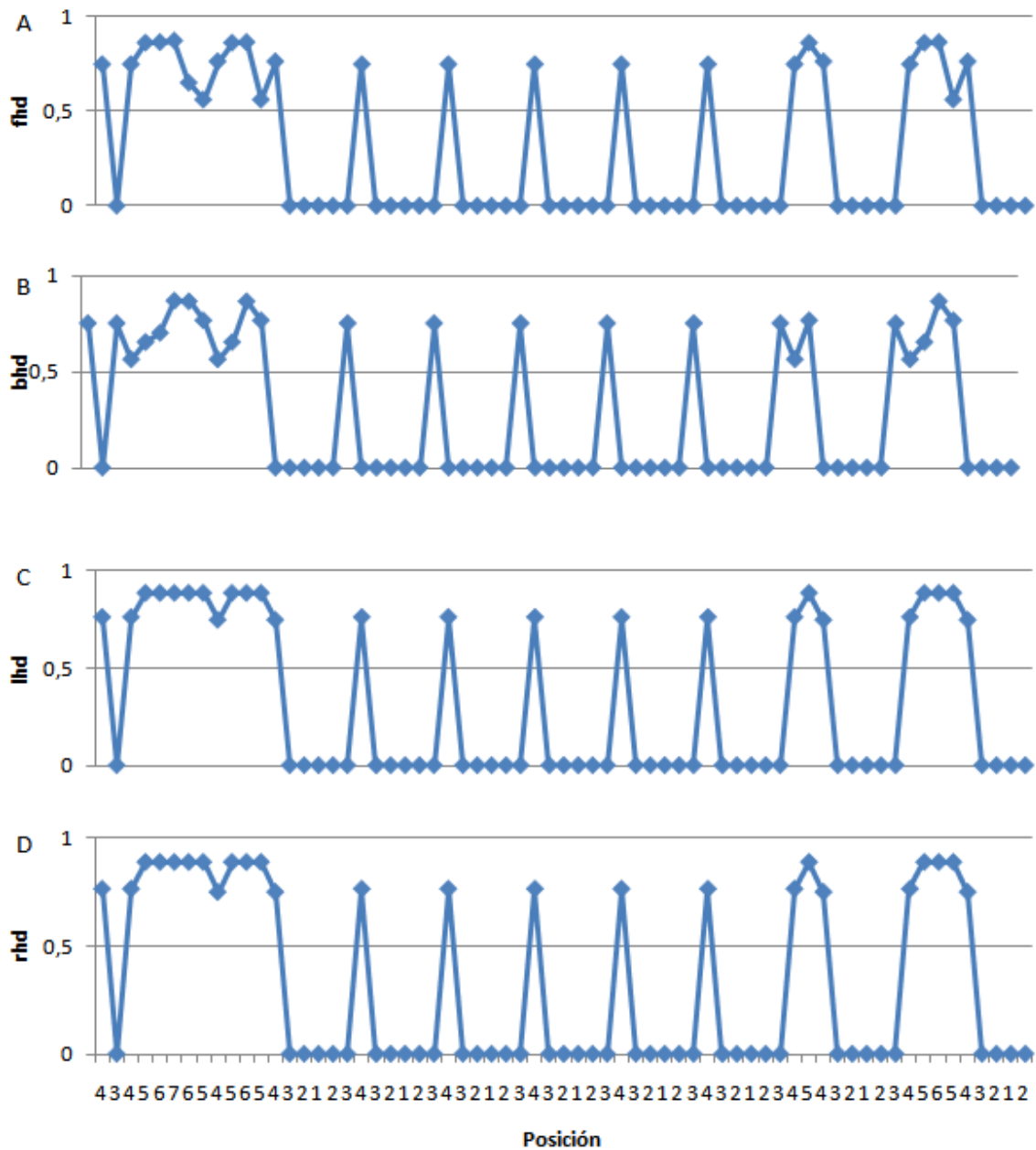


Fig 27. Evolución temporal de la densidad de altura de acuerdo a la dirección de la cabeza.
A. Adelante, B. Atrás, C. Izquierda, D. Derecha

Los patrones observados en las gráficas de fd y bd de estas densidades presentan una relación de inversión aproximada, como es de esperarse, que al invertir la dirección de la cabeza de la rata, esta perciba los mismos valores pero invertidos. Las gráficas de ld y rd son iguales, indicando la simetría que presenta el entorno respecto a las zonas de movilidad de la rata en el LCE.

En las gráficas se puede observar el patrón que representa una de las secuencias más probables, la 4-3-2-1-2-3, apareciendo siete veces y con diferente forma de acuerdo a la densidad. De igual manera se observa el patrón de la secuencia 4-5-6-7-5-4. Las desigualdades de las curvas que no corresponden a ninguno de los dos anteriores patrones mencionados corresponden a regresiones de una posición a la anterior, por ejemplo, la secuencia 4-3-4 o 4-5-4 que también aparece en las curvas.

ANÁLISIS DE DENSIDADES DE ILUMINACIÓN

Con el fin de visualizar mejor la distribución de los datos en la gráfica 25, estos se dividieron en 3 regiones: la región superior contiene los valores altos de id, la región media los valores intermedios y la región inferior los valores más bajos de id.

Todos aquellos valores de fid en la región alta corresponden a valores percibidos por la rata virtual cuando se encontraba en alguna posición de los BA o en la zona central. Los valores en la región media se percibieron cuando se encontraba en las posiciones 2 y 3 de los BC y se dirigía hacia afuera del brazo. Por último los valores en la región inferior se percibieron cuando se encontraba en los BC y se dirigía hacia adentro de este. El menor valor se obtuvo cuando estaba en los BC

mirando hacia el rincón, mientras que el mayor se obtuvo cuando estaba en la zona central mirando hacia los BA.

Todos aquellos valores de bid en la región alta corresponden a valores percibidos por la rata virtual cuando se encontraba en alguna posición de los BA o en la zona central. Los valores en la región media se percibieron cuando se encontraba dentro de los BC y se dirigía hacia adentro del brazo. Por último los valores en la región inferior se percibieron cuando se encontraba en las posiciones 2 y 3 de los BC y se dirigía hacia afuera de este. El menor valor se obtuvo cuando estaba en las posiciones 2 y 3 de los BC mirando hacia afuera, mientras que el mayor se obtuvo cuando estaba en la zona central mirando hacia los BA.

Por último todos aquellos valores de lid y rid en la región alta corresponden a valores percibidos por la rata virtual cuando se encontraba en alguna posición de los BA o en la zona central, mientras que los valores en la región inferior se percibieron cuando se encontraba dentro de los BC. El menor valor se obtuvo cuando estaba en los BC, mientras que el mayor se obtuvo cuando estaba en la zona central mirando hacia los BC.

En las gráficas de fid y bid las partes descendentes de la curva se presentan cuando la rata se dirige a zonas de bajo nivel de luminosidad y las partes ascendentes se presentan cuando se dirige a zonas de alto nivel de luminosidad.

Resumiendo lo anterior se puede decir que la rata virtual del modelo ANA percibe altos niveles de iluminación en los BA, valores medios en la zona central y bajos niveles de iluminación en los BC. Esta percepción de los niveles de iluminación del entorno esta en concordancia con el análisis del fenómeno para el caso de la rata real, luego se puede concluir que la definición densidades de iluminación id

propuesta en el modelo ANA permite reproducir la percepción de los niveles de iluminación por parte de la rata real cuando esta explora el LCE.

ANÁLISIS DE DENSIDADES DE PAREDES

Con el fin de visualizar mejor la distribución de los datos en la gráfica 26, estos se dividieron en 3 regiones para fwd y bwd, y en 2 regiones para lwd y rwd de la misma forma hecha para las id.

Todos aquellos valores de fwd y bwd en la región alta corresponden a valores percibidos por la rata virtual cuando se encontraba dentro de los BC. Los valores en la región media se percibieron cuando se encontraba en la zona central y estaba mirando hacia los BC. Por último los valores en la región inferior se percibieron cuando se encontraba en los BA o en la zona central mirando hacia los BA. El menor valor de fwd se obtuvo cuando estaba en los BA mirando hacia afuera del LCE, mientras que el mayor se obtuvo cuando estaba en los BC mirando hacia el rincón. El menor valor de bwd se obtuvo cuando estaba en las posiciones 5 y 6 de los BA mirando hacia la zona central, mientras que el mayor se obtuvo cuando estaba en los BC mirando hacia la zona central.

Por último todos aquellos valores de lwd y rwd en la región alta corresponden a valores percibidos por la rata virtual cuando se encontraba en los BC o en la zona central mirando hacia los BA, mientras que los valores en la región inferior se percibieron cuando se encontraba en los BA o en la zona central mirando hacia los BC. El menor valor se obtuvo cuando estaba en los BA, mientras que el mayor se obtuvo cuando estaba en los BC.

En las gráficas de fwd y bwd las partes descendentes de la curva se presentan cuando la rata se dirige a zonas con gran campo abierto y las partes ascendentes se presentan cuando se dirige a zonas rodeadas de paredes.

Resumiendo lo anterior se puede decir que la rata virtual del modelo ANA percibe altos niveles de campo abierto en los BA, valores medios en la zona central y que bajos niveles en los BC. Esta percepción del campo abierto del entorno esta en concordancia con el análisis del fenómeno para el caso de la rata real, luego se puede concluir que la definición de densidades de paredes wd propuesta en el modelo ANA permite reproducir la percepción del campo abierto por parte de la rata real cuando esta explora el LCE.

ANÁLISIS DE DENSIDADES DE ALTURAS

Con el fin de visualizar mejor la distribución de los datos en la gráfica 27, estos se dividieron en 2 regiones: la región superior contiene los valores altos de hd y la región inferior contiene los valores más bajos.

Todos aquellos valores de hd en la región alta corresponden a valores percibidos por la rata virtual cuando se encontraba en alguna posición de los BA o en la zona central, mientras que los valores de hd en la región inferior se percibieron cuando se encontraba en los BC.

El menor valor de hd se obtuvo cuando la rata estaba en los BC y tiene valor de cero (0) indicando que no se perciben alturas en los BC. El mayor valor de fhd se obtuvo cuando estaba en la posición 7, el mayor valor de bhd se obtuvo cuando la rata esta en la posición 6 mirando hacia la zona central, y el mayor valor de lhd y rhd se obtuvo cuando estaba en los BA.

En las gráficas de fhd y bhd las partes descendentes de la curva se presentan cuando la rata se dirige a zonas con poca cantidad de alturas y las partes ascendentes se presentan cuando se dirige a zonas con gran cantidad de alturas.

Resumiendo lo anterior se puede decir que la rata virtual del modelo ANA percibe altos niveles de alturas en los BA y en la zona central y que no percibe alturas en los BC. Esta percepción de las alturas del entorno esta en concordancia con el análisis del fenómeno para el caso de la rata real, luego se puede concluir que la definición densidades de alturas h_d propuesta en el modelo ANA permite reproducir la percepción de las alturas por parte de la rata real cuando esta explora el LCE.

5.6. ANÁLISIS E INTERPRETACIÓN DE DENSIDADES DE NOVEDAD

En las gráficas de nd se aprecia un comportamiento ascendente durante toda la evolución temporal del movimiento de la rata en el LCE. Para el caso de fnd y bnd se presenta un comportamiento ascendente oscilatorio en la parte media de la secuencia exploratoria provocado por la secuencia 4-3-2-1-2-3-4.

En fnd se observa que los valores disminuyen para la transición de 4 a 3 y para la secuencia 1-2-3, mientras que la parte ascendente se debe a la transición de 3 a 4. En bnd la parte descendente se debe a la transición de 4 a 3, mientras que la parte ascendente se debe a la secuencia 3-2-1-2-3-4. Al final de la exploración en fnd y bnd se observa que los valores disminuyen para la transición de 4 a 5 y que aumenta para la transición de 3 a 4.

DENSIDADES DE NOVEDAD ND

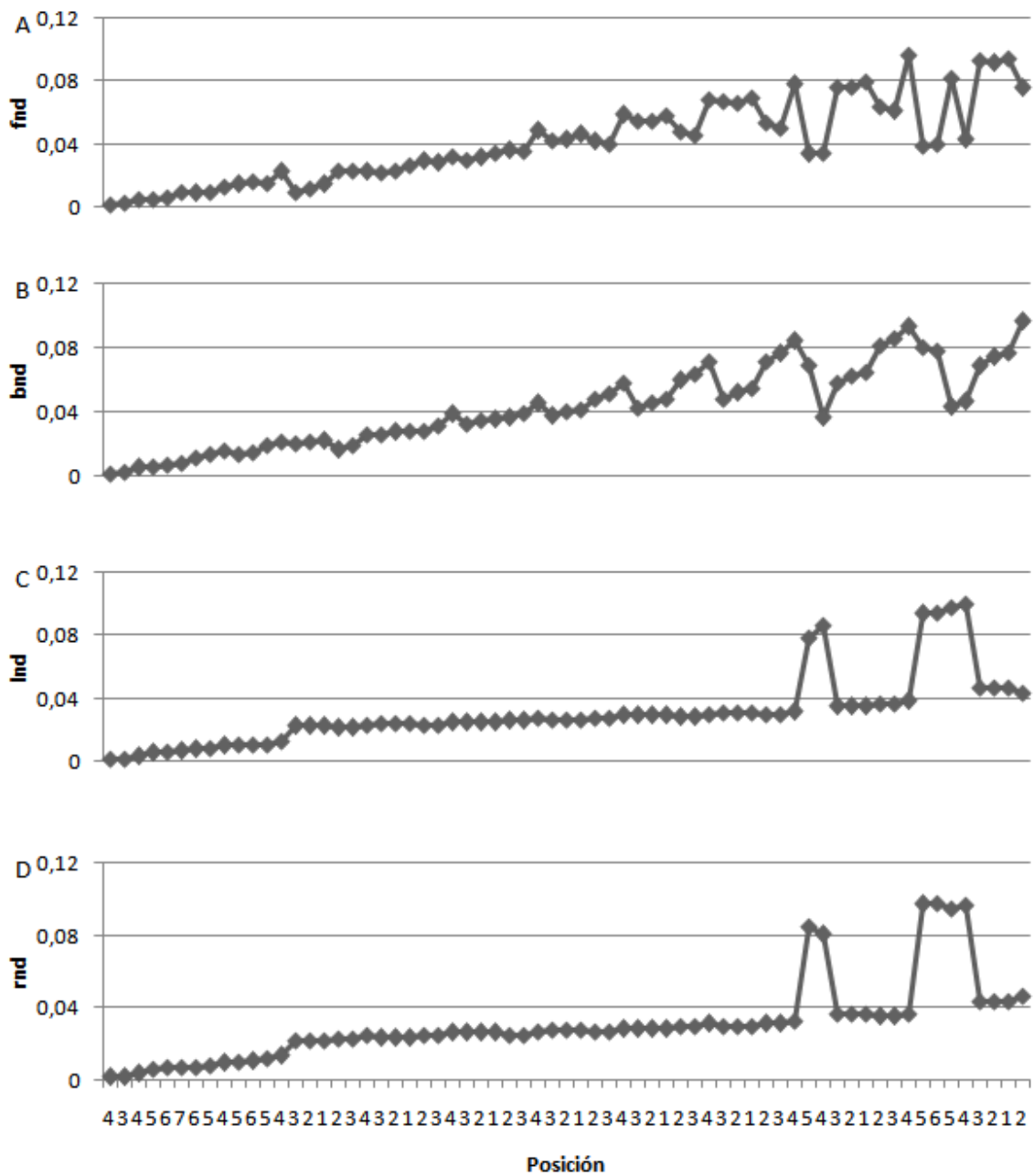


Fig 28. Evolución temporal de la densidad de novedad de acuerdo a la dirección de la cabeza.
A. Adelante, B. Atrás, C. Izquierda, D. Derecha

Las gráficas de lnd y rnd poseen valores similares más no son iguales, indicando que la rata ha explorado un poco más el brazo de un lado que el opuesto. Estas gráficas presentan cambios significativos en los valores de lnd y rnd al inicio y al final de la exploración, hecho que ocurre cuando la rata sale de un tipo de brazo, ya sea abierto o cerrado, y visita otro tipo de brazo después de haber permanecido gran tiempo explorando el anterior.

5.7. DISCUSIÓN

Se diseñó una rata virtual basada en RNA con características de ente autómeta, capaz de tomar decisiones de movimiento a partir de los estímulos externos encontrados en el entorno y de procesos mentales propios del animal. Esta característica posibilita estudios futuros del comportamiento de la rata en ambientes diferentes al LCE. La RNA entrenada se utilizó en el modo de asociación de patrones: patrones de comportamiento exploratorio.

Para establecer si era posible argumentar un criterio de aprendizaje basado en los resultados de la RNA propuesta se realizaron pruebas ROC. La prueba inicial ROC con 100 epochs fue inconsistente al presentar sus resultados, con una gran cantidad de datos fallidos, por el contrario los resultados obtenidos con 500 epochs mostraban pocos datos fallidos, indicando que la hipótesis propuesta para explicar estas inconsistencias, basada en la cantidad de epochs requeridos para obtener buenos resultados era correcta.

Esto se puede explicar si se tiene en cuenta que el proceso de entrenamiento de la RNA variaba la capacidad de reproducción del diagnóstico propuesto para determinar si esta había aprendido o no, presentandose dos estados: el primer estado era transitorio, en el cual se presentaba una variación del diagnóstico

entre estados 0 y 1, y posteriormente se alcanzaba un estado estacionario, en el cual se mantenía el diagnóstico de 1 desde un cierto valor de mse en adelante. Ver gráfica 29.

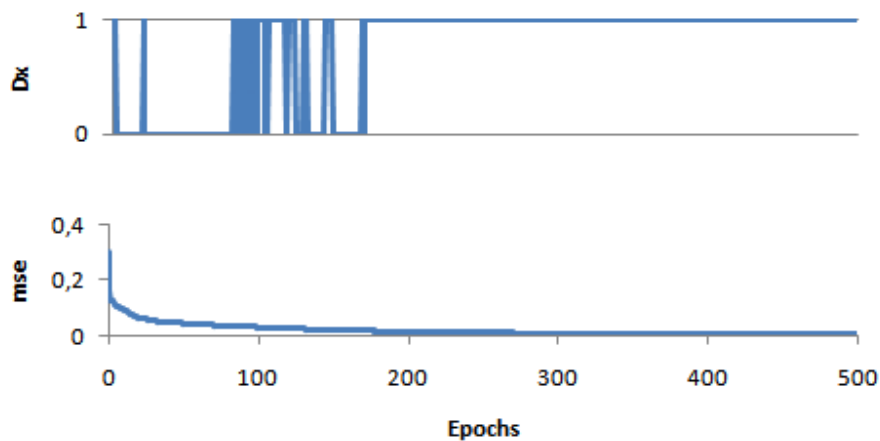


Fig 29. Evolución temporal del diagnóstico de aprendizaje (Dx) y del mse a través de los epochs. Se observa el estado transitorio (hasta 171 epochs) y posteriormente el estado estacionario.

La mayoría de los entrenamientos hasta 100 epochs presentan por completo el estado transitorio y como consecuencia, los valores de sensibilidad, especificidad y area bajo de la curva de la prueba ROC eran muy bajas, sin embargo para 500 epochs ya se observaba el estado estacionario, lo cual le daba una mayor estabilidad a la prueba ROC en la determinación del punto de corte.

A pesar de obtener buenos valores de área bajo la curva y sensibilidad, la reproducibilidad de los puntos de corte fue mala, dicho de otra forma, que para varios entrenamientos de RNA con un mismo conjunto de ejemplos de entrenamiento no se presentaron similares puntos de corte, sugiriéndose como responsable la estocasticidad del entrenamiento de las RNA usando el algoritmo *backpropagation*. De hecho en pruebas posteriores se demostró que sin importar el mse alcanzado, era posible que el diagnóstico se cumpliera, concluyéndose que el criterio de aprendizaje propuesto no era suficiente. Es por esto que se optó por

agregar al criterio de aprendizaje la condición de que los valores obtenidos por los parámetros comportamentales para las ratas virtuales no presentarán diferencias significativas respecto a las ratas reales usando ANOVA, y que además tuvieran buenas correlaciones respecto a los valores de las ratas reales.

En los resultados anteriores obtenidos se observó que el nuevo diagnóstico se cumple a partir de un mse de 0.01. Para un mse de 0.001 se mostró que las simulaciones eran capaces de reproducir casi a la perfección todos los parámetros comportamentales de entradas y en una buena medida más no comparable con la anterior los parámetros de tiempo. Se encontró que esta disminución en la exactitud radica en los tiempos de iteración de simulación. Es por esto que se considera que sería posible refinar el modelo corrigiendo el conflicto mencionado de los tiempos de permanencia para obtener mejores resultados, donde la solución propuesta consiste en asignar tiempos de decisión constantes y permitir que la rata virtual pueda decidir si desea quedarse quieta o no, mediante una quinta combinación de salida [0, 0, 0, 0], lo cual valdría la pena realizar en trabajos futuros.

Aún así la calidad de los datos obtenidos en las simulaciones sugiere una futura utilidad del modelo ANA en el estudio de: efecto de drogas ansiolíticas y ansiogénicas en el comportamiento de ratas, cambios de comportamiento del animal ante diferentes intensidades de iluminación, entre otros.

El modelo neurobiológico utilizado, fué tomado del trabajo de C. Salum, S. Morato y A.C. Roque-da-Silva sobre el desarrollo de modelos computacionales de la exploración de ratas en LCE. Para caracterizar la aversión y ganas de exploración de la rata, se definieron parámetros visuales y táctiles tales como: densidad de iluminación, densidad de paredes y densidad de alturas (para la aversión) y densidad de novedad (para la exploración).

Del análisis de las densidades se puede concluir que en las gráficas de fid - bid - fwd - bwd - fhd - bhd, las partes descendentes de la curva indican que la rata se dirige a zonas de bajo nivel de luminosidad, grandes campos abiertos y pocas alturas, y las partes ascendentes indican que la rata se dirige a zonas de alto nivel de luminosidad, rodeadas de paredes y gran cantidad de alturas.

La percepción de la rata de los niveles de iluminación, del campo abierto y de la cantidad de alturas del entorno esta en concordancia con el análisis del fenómeno para el caso de la rata real, luego se puede concluir que la definición de densidades de iluminación id, densidades de paredes wd y densidades de alturas hd propuesta en el modelo ANA permite reproducir la percepción de los niveles de iluminación, del campo abierto y de la cantidad de alturas por parte de la rata real cuando esta explora el LCE.

De la comparación de los valores absolutos de fid con bid, fwd con bwd y fhd con bhd en cada observación se evidencia la diferencia en los valores de aversión del entorno hacia delante y hacia atrás en las zonas del LCE. Esta diferencia provoca dos comportamientos clásicos en el LCE: el primero, cuando la rata se dirige hacia una zona de menor aversión, esta tiende a seguir en esa dirección, hacia adelante, ya que en la dirección opuesta, hacia atrás, se presentan dos estímulos inhibitorios, el de una mayor aversión y la falta de motivación para explorar una zona en la que la rata estaba. El segundo, cuando la rata va hacia una zona de mayor aversión, esta tiende a retrocederse varias veces, comportamiento que muestra la inseguridad de la rata al dirigirse a una zona que representa una posible amenaza, pero que aun así lo hace por una motivación de exploración mayor que el temor que le produce.

Las nd presentan un comportamiento ascendente durante toda la evolución temporal del movimiento de la rata en el LCE, lo cual es coherente con la hipótesis

de que a medida que transcurre el tiempo en la exploración del LCE por parte de la rata, esta va perdiendo el interés en explorar las zonas, debido al nivel de familiaridad con el entorno que esta va adquiriendo, dicho de otra forma, se pierde la novedad del entorno, que es lo que motiva a la rata a explorar zonas potencialmente aversivas como los BA.

De la comparación de los valores absolutos de fnd y bnd en cada observación se evidencia la diferencia en la motivación de exploración del entorno hacia delante y hacia atrás en las zonas del LCE. Esto provoca dos comportamientos clásicos en el LCE: el primero, cuando la rata se dirige hacia una zona de menor aversión, esta lo hace aún cuando en esa dirección la densidad de novedad es mayor. El segundo, cuando la rata va hacia una zona de mayor aversión, esta lo hace por una motivación de exploración mayor que el temor que le produce ir hacia una zona potencialmente aversiva, como lo es el BA.

En resumen se sugiere que la diferencia numérica entre los fd y bd para las cuatro densidades podría ser uno de los factores responsables de la toma de decisión de la rata para elegir hacia donde ejecutar el próximo movimiento. También del análisis del comportamiento cuando la rata se dirige hacia una zona de menor aversión aún cuando en esa dirección la densidad de novedad es mayor, se sugiere que una baja aversión es un estímulo mayor que un alto grado de familiaridad con el entorno en un experimento en el LCE con duración de 5 min.

Las lnd y rnd pueden tener valores diferentes en un determinado momento cuando la rata ha explorado un poco más el brazo de un lado que el opuesto. Posiblemente estas diferencias expliquen la selección de izquierda o derecha cuando la rata se encuentra en la posición 4. También se presentan cambios significativos en los valores de lnd y rnd cuando la rata sale de un tipo de brazo, ya sea abierto o cerrado, y visita otro tipo de brazo después de haber permanecido

gran tiempo explorando el anterior. Se sugiere que estas diferencias podrían aportar a la selección de BC o BA cuando la rata se encuentra en la posición 4.

En conclusión, todo el análisis anterior de las densidades sugiere que es posible caracterizar la aversión y exploración por medio de parámetros tales como densidad de paredes, densidad de huecos, densidad de iluminación, y densidad de novedad.

5.8. CONCLUSIONES

El criterio de aprendizaje propuesto de NEBC > NEBA y TPBC > TPBA tomando como referencia el mse del entrenamiento no es aplicable, sin embargo este mismo criterio más la condición de que los valores obtenidos por los parámetros comportamentales para las ratas virtuales no presenten diferencias significativas respecto a las ratas reales usando ANOVA, y que además tengan buenas correlaciones respecto a los valores de las ratas reales, si sirve y se cumple a partir de un mse de 0.01.

Para un mse de 0.001 el modelo ANA es capaz de reproducir casi a la perfección los parámetros comportamentales de entradas y en una buena medida los parámetros de tiempo.

Para las gráficas de fid - bid - fwd - bwd - fhd - bhd, las partes descendentes de las curvas indican que la rata se dirige a zonas de bajo nivel de luminosidad, grandes campos abiertos y pocas alturas, y las partes ascendentes indican que la rata se dirige a zonas de alto nivel de luminosidad, rodeadas de paredes y gran cantidad de alturas.

La definición de densidades de iluminación i_d , densidades de paredes w_d y densidades de alturas h_d propuesta en el modelo ANA permite reproducir la percepción de los niveles de iluminación, del campo abierto y de la cantidad de alturas por parte de la rata real cuando esta explora el LCE.

Es posible caracterizar la aversión y exploración por medio de parámetros tales como densidad de paredes, densidad de huecos, densidad de iluminación, y densidad de novedad.

BIBLIOGRAFÍA

[1] AFIFI, A. BERGMAN, R. Neuroanatomía funcional. McGraw Hill Interamericana. (2001) Capítulos 17-19-21.

[2] BARNETT, S. The rat: A study in behavior. London: Methuen. (1963).

[3] BECERRA, A. CARDENAS, F. MORATO, S. Effect of different illumination levels on rat behavior in the elevated plus-maze. *Physiol. Behav.* 85 (2005) 265-270.

[4] BECKER, A. GRECKSCH, G. Illumination has no effect on rat's behavior in the elevated plus-maze. *Physiol. Behav.* Vol 59 (1996) 1175-1177.

[5] BRANDÃO, M. ALBRECHET-SOUZA, L. OLIVEIRA, A. DE LUCA, M. C. TOMAZINI, F. SANTOS, N. A comparative study with two types of elevated plus-maze (transparent vs. opaque walls) on the anxiolytic effects of midazolam, one-trial tolerance and fear-induced analgesia. *Progress in Neuro-Psychopharmacol. Biol. Psych.* 29 (2005) 571-579.

[6] BURGUEÑO, M.J. GARCIA, J.L. GONZALEZ, J.M. Las curvas ROC en la evaluación de las pruebas diagnósticas. *Medicina clínica.* 104 (1995) 661-670.

[7] CONDE, C. AYALA, J. BAEZ, A. Evaluación comportamental comparativa por género y selección genética de ratas expuestas al laberinto en cruz elevado. *Salud UIS.* 33 (2001) 196-201.

[8] CONDE, C. AYALA, J. BOTELHO, S. HERRERA, A. VELASQUEZ, C. La vía visual puede ser el disparador de ansiogenicidad en el modelo del laberinto en cruz elevado. *Salud UIS*. 33 (2001) 190-195.

[9] CONDE, C. COSTA, V. TOMAZ, C. PROSTCOM. Un conjunto de programas para registro y procesamiento de datos comportamentales en investigaciones de fisiología y farmacología. *Biotemas*. 13 (2000) 145-159.

[10] ERDREICH, L. LEE, E. Use of relative operating characteristic analysis in epidemiology. A method for dealing with subjective judgement. *American J. of epidemiol.* 114 (1981) 649-662.

[11] FILE, S. ZANGROSSI, H. VIANA, M. GRAEFF, F. Trial 2 in the elevated plus-maze: a different form of fear?. *Psychopharmacol.* 111 (1993) 491-494.

[12] GHIRLANDA, S. ENQUIST, M. Artificial neural networks as models of stimulus control. *Animal Behaviour*. 56 (1998) 1383-1389.

[13] GIDDINGS, J. STEPHENSON, P. Modeling the behaviour of rats in an elevated plus maze (Bachelor's thesis). Acadia University. (2002).

[14] GRIEBEL, G. MOREAU, JL. JENCK, F. MARTIN, J. MISSLIN, R. Some critical determinants of the behavior of rats in the elevated plus-maze. *Behav. Processes*. 29 (1993) 37-48.

[15] GROSSEN, N. KELLEY, M. Species-specific behavior and acquisition of avoidance behavior in rats. *J. Comp. Physiol. Psychol.* 81 (1972) 307-310.

[16] HANDLEY, S. MITHANI, S. Effects of alpha2-adrenoceptor agonists and antagonists in a maze-exploration model of fear-motivated behavior. *Naunyn Schmiedeberg Arch. Pharmacol.* 327 (1984) 1-5.

[17] HANLEY, J. MCNEIL, B. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiol.* 143 (1982) 29-36.

[18] HANLEY, J. MCNEIL, B. A method of comparing the areas under receiver operating characteristic curves derived from the same cases. *Radiol.* 148 (1983) 839-843.

[19] HANLEY, J. Receiver operating characteristic (ROC) methodology: The state of the art. *Critical reviews in diagnostic imaging.* 29 (1989) 307-335.

[20] HARSTON, C. A neural network systems approach to image processing. *Comp. App. Systems.* (1991) 1539-1544.

[21] HARSTON, C. Pattern identification with a computerized neural network system. (1991) 935-939.

[22] HAYKIN, S. *Neural networks: A comprehensive foundation.* Prentice Hall. (1999) Capítulos 1-2-3-4.

[23] HENRIQUES, A. ARAÚJO, A. MORATO, S. Clustering exploratory activity in an elevated plus-maze with neural networks. *Proceedings of the IEEE-INNS-ENNS IJCNN 4.* (2000) 17-22.

[24] LAPIN, I.P. Only controls: effect of handling, sham injection, and intraperitoneal injection of saline on behavior of mice in an elevated plus-maze. *J. of Pharmacol. Toxicol. Methods.* 34 (1995) 73-77.

[25] LECUN, Y. Learning processes in an asymmetric threshold network, disordered systems and biological organization. Springer. (1986).

[26] MONTEROLA, C. SALOMA, C. Solving the nonlinear Schrödinger equation with an unsupervised neural network. *Optics Express.* 9 (2001) 72-84.

[27] MONTGOMERY, K. The relation between fear induced by novel stimulation and exploratory behavior. *J. Comp. Physiol. Psychol.* 48 (1955) 254:260.

[28] MORATO, S. CARDENAS, F. LAMPREA, M. Vibrissal sense is not the main sensory modality in rat exploratory behavior in the elevated plus-maze. *Behav. Brain Research.* 122 (2001) 169-174.

[29] PAXINOS, G. WATSON, C. The rat brain in stereotaxic coordinates. Elsevier Academic Press. (2004).

[30] PELLOW, S. CHOPIN, P. FILE, S. BRILEY, M. Validation of open: closed arm entries in an elevated plus-maze as a measure of anxiety in the rat. *J. Neurosci. Methods.* 14 (1985) 149-167.

[31] PELLOW, S. FILE, S. Anxiolytic and anxiogenic drug effects on exploratory activity in an elevated plus-maze: A novel test of anxiety in the rat. *Pharmacol. Biochem. Behav.* 24 (1986) 525-529.

[32] PELLOW, S. Anxiolytic and anxiogenic drug effects in a novel test of anxiety: are exploratory models of anxiety in rodents valid?. *Methods Find. Exp. Clin. Pharmacol.* 8 (1986) 557-565.

[33] RAJAVELU, A. MUSAVI, M.T. SHIRVAIKAR, M.V. A neural network approach to character recognition. *Neural Networks.* 2 (1988) 3877-3393.

[34] RODRIGUEZ, A. PÉRTEGA, S. GONZÁLEZ, A. PITA, S. SUÁREZ, G. CUERPO, M.A. Utilidad de las redes neuronales artificiales en la predicción de cáncer de próstata en la biopsia transrectal. *Actas. Urol. Esp.* 30 (2006) 18-24.

[35] SALUM, C. ROQUE-DA-SILVA, A.C. MORATO, S. Conflict as a determinant of rat behavior in three types of elevated plus-maze. *Behav. Processes.* 63 (2003) 87-93.

[36] SALUM, C. MORATO, S. ROQUE-DA-SILVA, A.C. Anxiety-like behavior in rats: a computational model. *Neural Networks.* 13 (2000) 21-29.

[37] SERRALVO, M. DE ARAÚJO, M. MENEGHINI, L. MARCIA, A. MESQUITA, F. YAE, D. MARINO, J. APARECIDA, M. MONTEIRO, T. An approach to evaluate the ability of rats discriminate different levels of illumination in the plus maze test: Effects of scopolamine. *Behav. Brain Research.* 180 (2007) 86-94.

[38] SERRALVO, M. ORLANDI, L. DA CUNHA, I. MARINO, J. APARECIDA, M. The gradient of luminosity between open/enclosed arms, and not the absolute level of lux, predicts the behavior of rats in the plus maze. *Behav. Brain Research.* 159 (2005) 55-61.

[39] TEJADA, J. OLIVEIRA, R. SALUM, C. MORATO, S. ROQUE-DA-SILVA, A.C. A model for the rat exploratory behaviour in the elevated plus maze. *BMC Neuroscience*. 8 (2007) (Suppl 2):P37.

[40] THE MATHWORKS INC. Neural networks toolbox. 4.0.6 version.

[41] TIMOTHY, M. Practical neural network recipes in c++. Academic Press. 1993.

[42] TREIT, D. FUNDYTUS, M. Thigmotaxis as a test for anxiolytic activity in rats. *Pharmacol. Biochem. Behav.* Vol 31 (1989) 959-962.

[43] TREIT, D. MENARD, J. ROYAN, C. Anxiogenic stimuli in the elevated plus-maze. *Pharmacol. Biochem. Behav.* 44 (1993) 463-469.

[44] TYLER, C. Foundations of Clinical Ophthalmology – Binocular vision. 2 (2004). Capítulo 24.

[45] WERBOS, P.J. Beyond regression: New tools for prediction and analysis in the behavioural sciences. (Ph.D. thesis). (1974).

ANEXO A

A continuación se describirán las leyendas contenidas dentro de los códigos en forma de comentario que explican el funcionamiento de las funciones, y a las cuales se pueden acceder por medio del comando *help* en el *Command Window* de Matlab. Se resalta con negrita la declaración de la función (primera línea), el nombre de la función, las variables involucradas, tanto de entrada como de salida, y propiedades de estas.

Es importante aclarar que el término “hueco” que aparece en la descripción de los códigos corresponde conceptualmente al término “altura”. También se dispone a continuación una porción de comentario en donde se explican algunas especificaciones computacionales de las variables *hed*, *rp* y *n* que por su gran uso por varios códigos se considera preferible mostrar aparte y de forma inicial. Debe asumirse que si una función hace mención de alguna de estas 3 variables dentro de su *help*, se está refiriendo a estas, de igual forma es necesario indicar que *nw* y *n* son equivalentes en significado (ver *help* de Training).

```
hed solo puede tomar los valores:
```

```
Arriba: [-1 0] Abajo: [1 0] Izquierda: [0 -1] Derecha: [0 1]
```

```
Los elementos del vector rp pueden tomar valores enteros entre [1 1] y las dos primeras componentes de size(env).
```

```
n es un número entero mayor o igual a 0 que permite especificar con cual worker ejecutar el algoritmo para trabajo en paralelo.
```

ALGORITMOS QUE EMULAN EL ENTORNO

ENV Contiene los distintos entornos

Sintaxis

```
fenvm = env(n)
```

Descripción

ENV contiene los distintos entornos

ENV(n), toma las entradas,
n - Escalar, número de worker
y retorna,
fenvm - QxRx6 matriz, entorno

Ejemplos

Aquí se tiene un entorno:

```
envm = zeros(5,5,6);

envm(:,:,1) = [1 1 1 1 1      %Matriz de zonas
               1 1 1 1 1      %1 es zona de movilidad
               1 1 1 1 1      %2 es hueco
               1 1 1 1 1
               1 1 1 1 1];

envm(:,:,2) = [0 0 0 0 0      %Matriz de pared arriba
               0 0 0 0 0
               0 0 1 0 0
               0 1 0 0 0
               0 0 0 0 0];

envm(:,:,3) = [0 0 0 0 0      %Matriz de pared abajo
               0 0 1 0 0
               0 1 0 0 0
               0 0 0 0 0
               0 0 0 0 0];

envm(:,:,4) = [0 0 0 0 0      %Matriz de pared a la izquierda
               0 0 0 0 0
               0 0 0 1 0
               0 0 0 0 0
               0 0 0 0 0];
```

```

envm(:, :, 5) = [0 0 0 0 0      %Matriz de pared a la derecha
                0 0 0 0 0
                0 0 1 0 0
                0 0 0 0 0
                0 0 0 0 0 ];

I = 0.5;          %Nivel de iluminación

envm(:, :, 6) = [I I I I I      %Matriz de iluminación
                I I I I I
                I I I I I
                I I I I I
                I I I I I];

```

Algoritmo

Los entornos se encuentran comprendidos en una matriz `envm(QxRx6)` que distingue los elementos así:

Zonas de movilidad = 1 y Huecos = 2 en `envm(:, :, 1)`
Paredes = 1 en `envm(:, :, 2:5)`, demás elementos = 0
Niveles de luminosidad en `envm(:, :, 6)`

Los elementos que componen la matriz de niveles de luminosidad deben ser valores reales entre 0 y 1.

PARENV Guarda el tipo de entorno y sus parámetros en un archivo

Sintaxis

```
parenv(Type,n)
```

Descripción

PARENV Guarda el tipo de entorno y sus parámetros en un archivo

PARENV(Type,n), toma las entradas,
Type - string, entorno seleccionado por el usuario
n - Escalar, número de worker
y salva Type junto con sus parámetros* requeridos en temp~\envtype.mat

* Para el caso del LCE corresponde a la Iluminación en lux.

Ejemplos

```
Type = 'LCE HI'; n = 1;  
parenv(Type);
```

Esta selección corresponde a param1 = 118, luego almacenará en el Archivo temp~\envtype1.mat Type y param1 para su posterior uso por **env**.

Algoritmo

Este código permite salvar en un archivo temporal la información relevante para la generación del entorno por parte de **env**.

Inicialmente se plantean 2 parámetros:

```
Type - Tipo de entorno ('LCE HI' - 'LCE MI' - 'LCE LI' -  
                        'Otro ...')  
param1 - Nivel de luminosidad de la zona central del LCE
```

Se impone una restricción a param1 de tomar valores entre 0 y 500, por razones de compatibilidad con códigos posteriores.

ALGORITMOS QUE COMPONEN LA RATA VIRTUAL

EYES Corresponde a los ojos de la rata virtual

Sintaxis

```
envm = eyes(rp,n)
```

Descripción

EYES(rp,n) Corresponde a los ojos de la rata virtual

EYES(rp,n), toma las entradas,

rp - 2 vector, posición de la rata

n - Escalar, número de worker

y retorna,

envm - QxRx6 matriz, entorno con elementos que la rata puede ver en su rango visual, que no fueron obstruidos por paredes

Ejemplos

Aquí se tiene un entorno:

```
envm = zeros(5,5,6);
```

```
envm(:,:,1) = [1 1 1 1 1      %Matriz de zonas
                1 1 1 1 1      %1 es zona de movilidad
                1 1 1 1 1      %2 es hueco
                1 1 1 1 1
                1 1 1 1 1];
```

```
envm(:,:,2) = [0 0 0 0 0      %Matriz de pared arriba
                0 0 0 0 0
                0 0 1 0 0
                0 1 0 0 0
                0 0 0 0 0];
```

```
envm(:,:,3) = [0 0 0 0 0      %Matriz de pared abajo
                0 0 1 0 0
                0 1 0 0 0
                0 0 0 0 0
                0 0 0 0 0];
```

```
envm(:,:,4) = [0 0 0 0 0      %Matriz de pared a la izquierda
                0 0 0 0 0
                0 0 0 1 0
                0 0 0 0 0
                0 0 0 0 0];
```

```

envm(:,:,5) = [0 0 0 0 0      %Matriz de pared a la derecha
               0 0 0 0 0
               0 0 1 0 0
               0 0 0 0 0
               0 0 0 0 0 ];

```

```

I = 0.5; %Nivel de iluminación

```

```

envm(:,:,6) = [I I I I I      %Matriz de iluminación
               I I I I I
               I I I I I
               I I I I I
               I I I I I];

```

Luego, el procedimiento modifica todas las matrices de acuerdo a los criterios explicados en la sección Algoritmo (a continuación), siendo $rp = [3 \ 3]$, dando como resultado:

```

envm(:,:,1) = [0 0 0 0 0      %Matriz de zonas
               0 0 0 0 0      %1 es zona de movilidad
               1 1 1 0 0      %2 es hueco
               1 1 1 0 0
               1 1 1 0 0];

```

```

envm(:,:,2) = [0 0 0 0 0      %Matriz de pared arriba
               0 0 0 0 0
               0 0 1 0 0
               0 1 0 0 0
               0 0 0 0 0];

```

```

envm(:,:,3) = [0 0 0 0 0      %Matriz de pared abajo
               0 0 0 0 0
               0 1 0 0 0
               0 0 0 0 0
               0 0 0 0 0];

```

```

envm(:,:,4) = [0 0 0 0 0      %Matriz de pared a la izquierda
               0 0 0 0 0
               0 0 0 0 0
               0 0 0 0 0
               0 0 0 0 0];

```

```

envm(:,:,5) = [0 0 0 0 0      %Matriz de pared a la derecha
               0 0 0 0 0
               0 0 1 0 0
               0 0 0 0 0
               0 0 0 0 0 ];

```

```

I = 0.5; %Nivel de iluminación

```

```

envm(:, :, 6) = [0 0 0 0 0      %Matriz de iluminación
                0 0 0 0 0
                I I I 0 0
                I I I 0 0
                I I I 0 0];

```

Algoritmo

El campo visual de una rata se puede considerar en primera aproximación de 360°, luego esta puede ver todo lo que le rodea, excepto aquellos elementos que se encuentran detrás de las paredes. Es por esto que **eyes** toma la matriz de entorno `envm` y:

- 1- Elimina todos los elementos que se encuentran detrás de una pared situada al frente de la posición de la rata. Las posibilidades son: Arriba, Abajo, Derecha e Izquierda
- 2- Elimina todos los elementos que se encuentran en el cuadrante detrás de una pared situada diagonal a la posición de la rata. Los cuadrantes son Arriba-derecho, Abajo-derecho, Arriba-izquierdo, Abajo-izquierdo

Los ojos de la rata le permiten captar información del entorno que se encuentre alejada de sus vibrisas, cuerpo o cola, tales como la ubicación de paredes y alturas, a la vez de percibir los niveles de iluminación de las zonas. Esta información es enviada a través del quiasma óptico y núcleo geniculado lateral a la corteza visual primaria para su posterior procesamiento.

TOUCH Corresponde al tacto de la rata virtual

Sintaxis

```
envm = touch(rp,n)
```

Descripción

TOUCH Corresponde al tacto de la rata virtual

TOUCH(rp,n), toma las entradas,
rp - 2 vector, posición de la rata
n - Escalar, número de worker
y retorna,
envm - 3x3x5 matriz, elementos que la rata puede tocar

Ejemplos

Aquí se tiene un entorno:

```
envm = zeros(5,5,6);

envm(:,:,1) = [2 2 1 2 2   %Matriz de zonas
               2 2 1 2 2   %1 es zona de movilidad
               1 1 1 1 1   %2 es hueco
               2 2 1 2 2
               2 2 1 2 2 ];

envm(:,:,2) = [0 0 1 0 0   %Matriz de pared arriba
               0 0 0 0 0
               0 0 0 0 0
               0 0 0 0 0
               0 0 0 0 0 ];

envm(:,:,3) = [0 0 0 0 0   %Matriz de pared abajo
               0 0 0 0 0
               0 0 0 0 0
               0 0 0 0 0
               0 0 1 0 0 ];

envm(:,:,4) = [0 0 1 1 0   %Matriz de pared a la izquierda
               0 0 1 1 0
               0 0 0 0 0
               0 0 1 1 0
               0 0 1 1 0 ];
```

```

envm(:,:,5) = [0 1 1 0 0      %Matriz de pared a la derecha
               0 1 1 0 0
               0 0 0 0 0
               0 1 1 0 0
               0 1 1 0 0 ];

I = 0.2;

envm(:,:,6) = [I I I I I      %Matriz de iluminación
               I I I I I
               I I I I I
               I I I I I
               I I I I I];

si rp = [2 3], entonces touch(rp) retorna;

envm(:,:,1) = [0 1 0
               0 1 0
               0 1 0];

envm(:,:,2) = [0 0 0
               0 0 0
               0 0 0];

envm(:,:,3) = [0 0 0
               0 0 0
               0 0 0];

envm(:,:,4) = [0 0 0
               0 1 0
               0 0 0];

envm(:,:,5) = [0 0 0
               0 1 0
               0 0 0];

```

Algoritmo

envm contiene los elementos que la rata puede tocar con sus vibrissas, cuerpo y cola, que equivalen a la percepción sensorial del entorno circundante o zonas cercanas a la rata. La información se almacena en una matriz 3x3x5, en la cual envm(:,:,1) contiene las zonas de movilidad y alturas cercanas a la posición actual de la rata. Los elementos de la posición (2,2) de envm(:,:,2:5) corresponden a las paredes que rodean a la rata, siendo estos = 1 si hay pared y 0 si no la hay. Los elementos ubicados detrás de paredes son eliminados. La posición central en la matriz corresponde con la posición actual de la rata.

Este algoritmo se basa en el comportamiento que presenta la rata que al encontrarse en una determinada zona de un entorno hace una exploración detallada de lo que le rodea usando su principal y más aguda vía sensorial, las vibrisas o bigotes, las cuales recogen información del exterior y la envían a través de los nervios por la medula espinal hacia la corteza somatosensitiva para su posterior procesamiento. Lo mismo pasa pero en menor medida con el cuerpo, y en mucha menor medida con la cola.


```

envm(:,:,4) = [0 0 0 0 0 0 0 0 0 0      %Matriz de pared a la
              0 0 0 0 0 0 0 0 0 0      izquierda
              0 0 0 0 1 1 0 0 0 0
              0 0 0 0 0 0 0 0 0 0
              0 0 0 0 1 1 0 0 0 0
              0 0 0 0 0 0 0 0 0 0
              0 0 0 0 0 0 0 0 0 0];

envm(:,:,5) = [0 0 0 1 0 0 0 0 0 0      %Matriz de pared a la
              0 0 0 1 0 0 0 0 0 0      derecha
              0 0 0 1 1 0 0 0 0 0
              0 0 0 0 0 0 0 0 0 0
              0 0 0 1 1 0 0 0 0 0
              0 0 0 1 0 0 0 0 0 0
              0 0 0 1 0 0 0 0 0 0];

I = 0.216;

envm(:,:,6) = [I I I I 0 0 0 0 0 0      %Matriz de iluminación
              I I I I 0 0 0 0 0 0
              I I I I I I I I I
              I I I I I I I I I
              I I I I I I I I I
              I I I I 0 0 0 0 0 0
              I I I I 0 0 0 0 0 0];

y para un hed = [0 1], se obtiene en viscortex;

param = [0.2153; 0.2160; 0.2149; 0.2149; 0.7000; 0.8750; ...
        0.8889; 0.8889; 0.2000; 0; 0.1667; 0.1667]
vacc = [1 0 0 0]

```

Algoritmo

A partir de la matriz suministrada por `eyes`, `viscortex` obtiene el vector `vacc` con las restricciones de movimiento debido a la proximidad de paredes o alturas a la posición actual de la rata, percibida por los ojos. Los elementos que componen `vacc` solo pueden ser valores de 0 y 1, donde 0 indica restricción y 1 indica permiso de movimiento en esa dirección.

Primero se halla el vector `tacc` usando `wacc` (wall access - restricciones de paredes) y `hacc` (height access - restricciones de alturas), el cual contiene las restricciones en SRL y después se transforma este a SRR, usando `hed`, para así obtener `vacc`.

La corteza visual del cerebro de la rata virtual (`viscortex`) recibe la información aferente de los ojos (`eyes`) y la procesa para obtener una interpretación significativa de los estímulos externos que esta

considera importantes, tales como el nivel de iluminación de las zonas, el nivel de campo abierto y la proximidad de alturas.

Para esto genera parámetros denominados "densidades", por ser estos un valor promedio del nivel de intensidad de los estímulos a lo largo de una zona en particular (por ej., la zona adelante de ella).

El vector param se compone de densidades de iluminación, de altura y de paredes, 4 de cada tipo correspondientes a cada una de las posibles direcciones: Adelante, Atrás, Izquierda y Derecha, para dar un total de 12 elementos. Estas densidades tienen sus valores normalizados, pudiendo entonces tomar valores reales entre 0 y 1, correspondiendo altos valores con zonas de alta iluminación, rodeadas de gran cantidad de paredes y alturas, y correspondiendo los bajos valores con zonas de baja iluminación, con gran campo abierto y en ausencia de alturas.

Las densidades se obtienen matemáticamente;

$$id = I/n \quad hd = H/n \quad wdu = W/(2*n)$$

donde;

- I - Sumatoria de los niveles de iluminación por cada zona en una determinada dirección
- H - Cantidad de zonas de altura en una determinada dirección
- W - Dos veces la cantidad de paredes en una determinada dirección
- n - Cantidad de zonas con nivel de iluminación distinto de 0 en una determinada dirección

El vector armado es;

```
param = [idf; idb; idl; idr; hdf; hdb; hdl; hdr; wdf; ...  
         wdb; wdl; wdr];
```

donde las letras corresponden a;

- i - illumination
- h - height
- w - wall
- d - density
- f - forward
- b - backward
- l - left
- r - right

Una vez el vector de densidades esta armado, esta información se envía a una corteza asociativa que tiene eferencias tanto de la corteza visual como de la corteza somatosensitiva, para su posterior ponderación.

SOMSENCORTEX Corresponde a la Corteza Somatosensitiva de la rata virtual

Sintaxis

```
[param,sacc] = somsencortex(rp,hed,n)
```

Descripción

SOMSENCORTEX Corresponde a la Corteza Somatosensitiva de la rata virtual

SOMSENCORTEX(rp,hed,n), toma las entradas,
rp - 2 vector, posición de la rata
hed - 2 vector fila, dirección de la cabeza
n - Escalar, número de worker
y retorna,
param - 8 vector columna, densidades de paredes y alturas
sacc - 4 vector fila, restricciones de movimiento

Ejemplos

Siendo la matriz obtenida por `touch`;

```
envm(:, :, 1) = [3 1 0  
                2 1 0  
                0 0 0];
```

```
envm(:, :, 2) = [0 0 0  
                0 0 0  
                0 0 0];
```

```
envm(:, :, 3) = [0 0 0  
                0 1 0  
                0 0 0];
```

```
envm(:, :, 4) = [0 0 0  
                0 0 0  
                0 0 0];
```

```
envm(:, :, 5) = [0 0 0  
                0 1 0  
                0 0 0];
```

y para un hed = [-1 0], se obtiene en `somsencortex`;

```
param = [0; 0; 1; 0; 0; 1; 0; 1]  
sacc = [1 0 0 0]
```

Algoritmo

A partir de la matriz suministrada por `touch`, `somsencortex` obtiene el vector `sacc` con las restricciones de movimiento debido a la proximidad de paredes o alturas a la posición actual de la rata, percibida por las vibrisas, cuerpo y cola. Los elementos que componen `sacc` solo pueden ser valores de 0 y 1, donde 0 indica restricción y 1 indica permiso de movimiento en esa dirección.

Primero se halla el vector `tacc` usando `wacc`(wall access - restricciones de paredes) y `hacc`(height access - restricciones de alturas), el cual contiene las restricciones en SRL y después se transforma este a SRR, usando `hed`, para así obtener `sacc`.

La corteza somatosensitiva del cerebro de la rata virtual (`somsencortex`) recibe la información aferente de las vibrisas, cuerpo y cola (`touch`) y la procesa para obtener una interpretación significativa de los estímulos externos que esta considera importantes, tales como el nivel de campo abierto y la proximidad de alturas. Para esto genera parámetros denominados "densidades", por ser estos un valor promedio del nivel de intensidad de los estímulos a lo largo de una zona en particular (por ej., la zona adelante de ella).

El vector `param` se compone de densidades de alturas y paredes, 4 de cada tipo correspondientes a cada una de las posibles direcciones: Adelante, Atrás, Izquierda y Derecha, para dar un total de 8 elementos. Estas densidades valen 1 cuando hay presencia de una pared o una altura en esa dirección, o vale 0 si no hay. El vector armado es;

```
param = [hdf; hdb; hdl; hdr; wdf; wdb; wdl; wdr]
```

donde las letras corresponden a;

```
h - height  
w - wall  
d - density  
f - forward  
b - backward  
l - left  
r - right
```

Una vez el vector de densidades esta armado, esta información se envía a una corteza asociativa que tiene eferencias tanto de la corteza visual como de la corteza somatosensitiva, para su posterior ponderación.

VSAC Corteza Asociativa de información aferente de las Cortezas Visual y Somatosensitiva

Sintaxis

```
[param,tacc] = vsac(rp,hed,n)
```

Descripción

VSAC Corteza Asociativa de información aferente de las Cortezas Visual y Somatosensitiva

VSAC(rp,hed,n), toma las entradas,
rp - 2 vector columna, posición de la rata
hed - 2 vector, dirección de la cabeza
n - Escalar, número de worker
y retorna,
param - 12 vector columna, densidades de paredes, alturas e iluminación
tacc - 4 vector fila, restricción de movimiento

Ejemplos

Siendo los vectores obtenidos por **viscortex**;

```
param = [0.2153; 0.2160; 0.2149; 0.2149; 0.7000; 0.8750; ...  
         0.8889; 0.8889; 0.2000; 0; 0.1667; 0.1667]  
vacc = [1 0 0 0]
```

y siendo los vectores obtenidos por **somsencortex**;

```
param = [0; 0; 1; 0; 0; 1; 0; 1]  
sacc = [1 0 0 0]
```

para el caso de completa oscuridad **vsac** retorna;

```
param = [0; 0; 0; 0; 0; 0; 1; 0; 0; 1; 0; 1]  
tacc = [1 0 0 0]
```

mientras que si hay luz **vsac** retorna;

```
param = [0.2153; 0.2160; 0.2149; 0.2149; 0.7000; 0.8750; ...  
         0.8889; 0.8889; 0.2000; 0.76; 0.1667; 0.85]  
tacc = [1 0 0 0]
```

Algoritmo

La comparación de la información suministrada por las cortezas visual y somatosensitiva en los vectores param se limita únicamente al caso de completa oscuridad,

```
sum(sum(envm(:, :, 6))) == 0
```

en el cual la detección del entorno solo es posible a través de las sensaciones táctiles, mientras que si hay luz, el aporte del tacto es redundante con el visual, reforzando esta vía de información y primando entonces la información visual, siendo relegada la táctil. En resumen, de acuerdo a los niveles de iluminación del entorno se excluye uno de los dos aportes sensoriales.

Mientras que el vector de restricción de movimiento tacc se obtiene como complementación de la información de los vectores de acceso visuales (vacc) y somatosensitivos (sacc).

El vector armado es;

```
param = [idf; idb; idl; idr; hdf; hdb; hdl; hdr; wdf; wdb; ...  
        wdl; wdr];
```

donde las letras corresponden a;

```
i - illumination  
h - height  
w - wall  
d - density  
f - forward  
b - backward  
l - left  
r - right
```

SPAMEM Memoria Espacial

Sintaxis

```
nd = spamem(rp,hed,n)
```

Descripción

SPAMEM Memoria espacial

SPAMEM(rp,hed,n), toma las entradas,
rp - 2 vector, posición de la rata
hed - 2 vector fila, dirección de la cabeza
n - Escalar, número de worker
y retorna,
nd - 4 vector columna, densidades de novedad

Ejemplos

Siendo

```
rp = [3 3]; hed = [-1 0]; n = 1;  
mem = [0 0 0 0 0  
        0 0 0 0 0  
        0 0 0 0 0  
        0 0 0 0 0  
        0 0 0 0 0 ];
```

y definiéndose el entorno como un campo abierto, o sea, no hay obstáculos que impidan observar.

```
nd = spamem(rp,hed,n) obtiene;
```

```
nd = [0.5; 0.5; 0.5; 0.5]
```

```
mem = [ 0.5 0.5 0.5 0.5 0.5  
        0.5 0.5 0.5 0.5 0.5  
        0.5 0.5 1 0.5 0.5  
        0.5 0.5 0.5 0.5 0.5  
        0.5 0.5 0.5 0.5 0.5 ]
```

Algoritmo

mem corresponde a la matriz QxR que almacena la información de la memoria espacial, y que se almacena en spamemn.mat, donde QxR es el mismo tamaño de la matriz de envm y n es el número de worker. Inicialmente mem se compone de elementos 0 los cuales se modifican a medida que evoluciona la simulación. Para generar la nueva matriz de

mem a partir de la anterior, se suma 0.5 a cada elemento correspondiente a una zona de movilidad que la rata ha visto pero que no ha estado, y se suma 1 a cada zona en la que esta ha estado.

Para obtener las densidades de novedad para cada dirección: ndf, ndb, ndl, ndr, se hace un promedio, sumando los elementos distintos de cero y dividiéndolos entre la cantidad de elementos para una determinada dirección.

```
ndx = Sum(elemento ~= 0)/(# elementos)
```

El vector obtenido es;

```
nd = [ndf; ndb; ndl; ndr];
```

donde las letras corresponden a;

```
n - novelty  
d - density  
f - forward  
b - backward  
l - left  
r - right
```

Estas densidades tienen sus valores normalizados, correspondiendo altos valores con zonas que han sido muy visitadas por la rata y correspondiendo los bajos valores con zonas poco visitadas. Para normalizar las densidades es necesario multiplicar estas por un factor de escala:

```
nd = nd/300
```

Este valor limita la cantidad de decisiones por secuencia a 300. El factor de escala se encontró suponiendo un experimento de 300 s, en el cual la rata virtual toma decisiones cada 1 s, por lo cual hay un máximo de 300 decisiones, y si esta permanece todo el experimento quieta en una posición límite, su densidad de novedad ndx aumentara 0.0033 cada vez, de modo que 300 decisiones después esta alcanzara una ndx = 1.

CRIT Selecciona la decisión de movimiento de la rata

Sintaxis

```
rmov = crit(outnet,tacc,minlim,ind)
```

Descripción

CRIT Selecciona la decisión de movimiento de la rata

```
CRIT(outnet,tacc,minlim,ind), toma las entradas,  
outnet - 4 vector columna, salida de la RNA  
tacc   - 4 vector columna, prohíbe direcciones de movimiento  
minlim - Escalar, valor mínimo para interpretar la decisión como  
         movimiento  
ind    - Escalar, brecha de indecisión  
y retorna,  
rmov   - 2 vector fila, movimiento de la rata en el Sistema de  
         Referencia de la Rata (SRR)
```

Ejemplos

```
ind    = 0.1; minlim = 0.1;  
tacc   = [1 0 1 1];  
outnet = [0 0.3 0.58 1];  
rmov   = crit(outnet,tacc,minlim,ind)  
  
rmov   = [0 1]
```

Algoritmo

outnet es un vector compuesto de 4 elementos que pueden tomar valores reales entre 0 y 1. Cada elemento corresponde a la decisión de movimiento en una determinada dirección, siendo las posibilidades:

- Adelante - Atrás - Izquierda - Derecha

y en el vector la distribución;

```
outnet = [Adelante; Atrás; Izquierda; Derecha]
```

Un alto valor en un elemento de outnet se interpreta como una disposición voluntaria de deseo de movimiento en esa dirección, un valor intermedio se interpreta como incertidumbre, y un valor bajo se interpreta como deseo en la inhibición de movimiento en esa dirección.

Los elementos que componen tacc solo pueden ser valores de 0 y 1, donde 0 indica restricción y 1 indica permiso de movimiento en esa

dirección, esto como consecuencia de una restricción autoimpuesta por la rata de no tratar de avanzar en una dirección en la que se encuentre algún obstáculo, ya sea una pared o un hueco.

`minlim` e `ind` son números reales entre 0 y 1. `minlim` corresponde al mínimo valor necesario para interpretar un elemento de `outnet` como deseo de movimiento, y se debe entender como el umbral entre la inhibición y la activación del movimiento, mientras que `ind` corresponde a la diferencia de valores necesaria entre dos elementos de `outnet` para que el mayor de los dos prime como una voluntad superior y esta corresponda al movimiento ejecutado, por el contrario si su diferencia es menor que `ind`, se presenta una indecisión sobre cual dirección de movimiento elegir y entonces se elige entre las dos de manera aleatoria.

De forma breve, `crit` discrimina las decisiones de movimiento de la siguiente manera:

- 1- Si algún valor está por debajo de `minlim`, entonces el movimiento en esa dirección se inhibe = 0
- 2- Los valores por encima de `minlim` compiten, si tienen diferencia mayor a `ind`, entonces gana el mayor, sino, la selección del movimiento se hace aleatoriamente

`Crit` devuelve como salida el vector `rmov`, que representa la dirección de movimiento elegida a ejecutar. De este modo `rmov` puede tomar los valores:

Adelante: [-1 0] Atrás: [1 0] Izquierda: [0 -1] Derecha: [0 1]
Quieto: [0 0]

VIRAT Simula el comportamiento exploratorio de una rata basada en RNA

Sintaxis

```
[rp,hed] = virat(rp,hed,Red,minlim,ind,n)
```

Descripción

VIRAT Simula el comportamiento exploratorio de una rata basada en RNA

VIRAT(rp,hed,Red,minlim,ind,n), toma las entradas,

- rp - 2 vector columna, posición de la rata
- hed - 2 vector, dirección de la cabeza
- Red - string, dirección y nombre de la RNA
- minlim - Escalar, valor mínimo para interpretar la decisión como movimiento
- ind - Escalar, brecha de indecisión
- n - Escalar, número de worker

y retorna,

- rp - 2 vector fila, nueva posición de la rata después de ejecutar el movimiento
- hed - 2 vector, nueva dirección de la cabeza

Algoritmo

Este se basa en un modelo neuroanatómico y neurofuncional del cerebro de una rata, donde esta adquiere la información del Entorno (**env**) en el que se encuentra por medio de los Ojos (**eyes**), Vibrisas, Cuerpo y Cola (**touch**), todos correspondientes a módulos computacionales programados.

La información obtenida por los sentidos es procesada por una Corteza Visual (**viscortex**), Somatosensitiva (**somsencortex**), y que comparan su información por medio de una Corteza Asociativa (**vsac**) dando así una salida y que junto con la salida de la Memoria Espacial (**spanem**) pasan luego a una RNA (Red) que toma la decisión de hacia dónde moverse (Adelante, Atrás, Izquierda o Derecha) en SSR.

Posteriormente, los datos de salida pasan por un Criterio de Selección de movimiento (**crit**), que elimina toda incertidumbre en la decisión, dejando como ultima solo una opción. El movimiento decidido entonces se expresa a través de los vectores rp (rat position) y hed (head direction).

minlim e ind son números reales entre 0 y 1. minlim corresponde al mínimo valor necesario para interpretar un elemento del vector salida de la RNA como deseo de movimiento, y se debe entender como el umbral entre la inhibición y la activación del movimiento, mientras que ind corresponde a la diferencia de valores necesaria entre dos elementos

de vector salida de la RNA para que el mayor de los dos prime como una voluntad superior y esta corresponda al movimiento ejecutado, por el contrario si su diferencia es menor que ind , se presenta una indecisión sobre cual dirección de movimiento elegir y entonces se elige entre las dos de manera aleatoria.

ALGORITMOS QUE ENTRENAN LAS RNA

CONDE2VR Convierte datos Conde a datos VR

Sintaxis

```
[Input,Target] = conde2vr(T,X,hed,stdis,n)
```

Descripción

CONDE2VR convierte datos de Conde a datos VR

CONDE2VR(T,X,hed,stdis), toma las entradas,

T - S vector, tiempos de permanencia
X - S vector, posición del LCE en nomenclatura Conde
hed - 2 vector fila, dirección de la cabeza
stdis - Lógico, deshabilita la opción de permanencia
n - Escalar, número de worker

y retorna

Input - SxR matriz de entrada de entrenamiento de la RNA para cerlce

Target - SxR matriz de meta de entrenamiento de la RNA para cerlce

Ejemplos

```
Si X = [4 3 2 1 2 3 4 5 6 7];  
T = [1 1 1 1 1 1 1 1 1 1];  
hed = [-1 0];  
stdis = 1; n = 1;
```

entonces los datos VR corresponden a;

Target =

```
[ 1  1  1  0  1  1  0  1  1  
  0  0  0  1  0  0  0  0  0  
  0  0  0  0  0  0  0  0  0  
  0  0  0  0  0  0  1  0  0 ]
```

Input =

```
[ 0.214 0.196 0.196 0.196 0.203 0.204 0.214 0.216 0.216
  0.214 0.196 0.196 0.196 0.203 0.204 0.214 0.216 0.216
  0.214 0.196 0.196 0.196 0.203 0.204 0.214 0.216 0.216
  0.217 0.196 0.196 0.196 0.196 0.196 0.217 0.215 0.215
  0.750 0 0 0 0 0 0.750 0.864 0.867
  0.750 0 0 0 0 0 0.750 0.563 0.652
  0.765 0 0 0 0 0 0.765 0.889 0.889
  0.765 0 0 0 0 0 0.765 0.889 0.889
  0.438 1.000 1.000 1.000 0.833 0.800 0.438 0 0
  0.438 0.800 0.833 0.857 1.000 1.000 0.438 0.188 0.261
  0.177 0.500 0.500 0.500 0.500 0.500 0.177 0.167 0.167
  0.177 0.500 0.500 0.500 0.500 0.500 0.177 0.167 0.167
  0.002 0.003 0.005 0.008 0.005 0.005 0.012 0.005 0.007
  0.002 0.003 0.004 0.005 0.010 0.013 0.014 0.011 0.012
  0.002 0.002 0.002 0.001 0.002 0.002 0.003 0.012 0.013
  0.002 0.002 0.002 0.002 0.002 0.002 0.003 0.015 0.016 ]
```

Algoritmo

stdis solo puede tomar valores de 1 para deshabilitar la permanencia y 0 para habilitarla, siendo la permanencia la capacidad de la rata virtual de decidir no moverse y permanecer un determinado tiempo en una misma zona.

T y X deben ser vectores de longitud mínima 2, donde los elementos de T corresponden a tiempos de permanencia de la rata en una determinada zona indicada en los elementos del vector X. Los tiempos deben estar en segundos y las posiciones en nomenclatura Conde.

Las matrices Input y Target corresponden a los ejemplos de entrenamiento de la RNA. Los elementos de Input equivalen a densidades de paredes, alturas, iluminación y novedad, las cuales están normalizadas, mientras que los elementos de Target equivalen a decisiones de movimiento en una determinada dirección, donde 1 indica movimiento y 0 lo contrario.

NNMOV Crea y entrena un PMC que decide hacia donde se moverá la rata

Sintaxis

```
net = nnmov(input,target,gap,goal,epochs,leal,ndir)
```

Descripción

NNMOV Crea y entrena un PMC que decide hacia donde se moverá la rata

```
NNMOV(input,target,gap,goal,epochs,leal), toma las entradas,  
input - 16xS matriz, conjunto de entrenamiento de la RNA  
target - 4xS matriz, salida esperada de entrenamiento de la RNA  
gap - Escalar, brecha deseada  
goal - Escalar, error deseado  
epochs - Escalar, epochs como criterio de parada  
leal - String, puede ser 'trainrp' o 'trainrpgap'  
ndir - String, nombre del directorio  
y retorna,  
net - Network
```

Algoritmo

input contiene los ejemplos de entrenamiento para la RNA, al igual que target contiene los valores de salida esperados para cada ejemplo de entrenamiento. S es el número de ejemplos de entrenamiento. input y target son matrices que se pueden obtener como resultado de ejecutar `conde2vr`.

Gap debe ser un número real entre 0 y 1, y corresponde a la altura de las zonas que representan los estados binarios '0' y '1' para las salidas de la RNA. El goal es un número real ≥ 0 y que en este caso representa el error que se espera alcanzar como criterio de parada del entrenamiento de la RNA, comparándose este con el mse (mean-square-error). Epochs corresponde al número de iteraciones para actualizar los pesos sinápticos y sesgos de la RNA durante el entrenamiento. ndir es el nombre del directorio en el cual se almacenan los archivos con información de la RNA durante el entrenamiento para el caso de 'Learning Analysis' seleccionado en la interfaz gráfica [Training](#).

El algoritmo de aprendizaje es el 'resilient backpropagation', para el cual se manejan dos diferentes criterios de parada. Si leal es 'trainrp2' los parámetros de parada son goal o epochs, mientras que si leal es 'trainrpgap' el parámetro de parada es el gap en equivalencia al funcionamiento del goal en el caso anterior. En mayor detalle, el gap como criterio de parada indica que el aprendizaje ha finalizado cuando todas las salidas de la RNA se encuentren dentro de alguno de los dos estados binarios '0' o '1' y no en la brecha de incertidumbre

comprendida entre estos dos. Este último comprende un mayor tiempo de ejecución respecto al primero.

Como resultado de este algoritmo, se obtiene una RNA entrenada, con una arquitectura de Perceptrón Multicapa (PMC), con una capa oculta con una cantidad de neuronas X , donde:

$$X = \text{fix}(S/2);$$

también con 16 nodos en la capa de entrada, que representan la cantidad de elementos que componen cada ejemplo de entrenamiento, y con 4 neuronas de salida que representan las posibilidades de movimiento de la rata en el LCE: Adelante, Atrás, Izquierda y Derecha. Se usó como función de transferencia en todas las neuronas 'logsig'.

TRAINING Entrena y salva RNA (Interfaz grafica)

Descripción

TRAINING, toma las entradas,

- type - string, tipo de entrenamiento, puede ser 'Standard' o 'Learning Analysis'
- Icen - Escalar, nivel de intensidad de la posición central (Lux)
- leal - string, algoritmo de aprendizaje, puede ser 'trainrp2' o 'trainrpgap'
- N - Escalar, cantidad de RNA a entrenar
- stilldis - Lógico, deshabilita la opción de permanencia
- stopcrit - String, criterio de parada del entrenamiento
- gap - Escalar, brecha deseada
- goal - Escalar, error deseado
- epochs - Escalar, cantidad máxima de iteraciones de entrenamiento
- nw - Escalar, numero de worker

y retorna RNA salvadas

Algoritmo

Training entrena y salva RNA basadas en datos crudos almacenados en archivos *.xls, los cuales deben estar contenidos en la carpeta 'Trainxls'. De acuerdo al nivel de iluminación existen 3 carpetas denominadas 'Alta', 'Media' y 'Baja'. N es la cantidad de RNA a entrenar para un mismo conjunto de ejemplos de entrenamiento.

stilldis solo puede tomar valores de 1 para deshabilitar la permanencia y 0 para habilitarla, siendo la permanencia la capacidad de la rata virtual de decidir no moverse y permanecer un determinado tiempo en una misma zona.

Icen puede ser 'HI' (high-118 Lux), 'MI' (medium-95 Lux), 'LI'(low-37 Lux). Xl vector fila con las posiciones, Tl con los tiempos que permanece la rata en cada posición, lo anterior en nomenclatura Conde y hed la dirección de la cabeza. Por cada sesión se salva una RNA.

stopcrit determina el criterio de parada del entrenamiento de la RNA, si se elige 'trainrp', los criterios pueden ser;

- Anyone (goal o epochs)
- Performance goal met (goal)
- Maximum epochs reached (epochs)

mientras que si se elige 'trainrpgap' el criterio es;

- All gaps created (gap)

Para dejar el entrenamiento estándar, escoger en type 'Standard', donde se guardan las redes en la carpeta 'Networks' en archivos con la forma name_x.mat, donde x es la x-esima red entrenada. Para utilizar el 'Análisis de Aprendizaje' que guarda la mayor cantidad de información obtenida durante el entrenamiento de la RNA, escoger 'Learning Analysis'.

En el modo 'Standard' solo se salvan las RNA que han parado su entrenamiento al alcanzar el criterio de parada (stopcrit) seleccionado por el usuario. En el modo 'Learning Analysis' se salva una RNA por cada epoch de entrenamiento ejecutado.

Gap debe ser un número real entre 0 y 1, y corresponde a la altura de las zonas que representan los estados binarios '0' y '1' para las salidas de la RNA. El goal es un numero real ≥ 0 y que en este caso representa el error que se espera alcanzar como criterio de parada del entrenamiento de la RNA, comparándose este con el mse (mean-square-error). Epochs corresponde al número de iteraciones para actualizar los pesos sinápticos y sesgos de la RNA durante el entrenamiento.

ALGORITMOS QUE SIMULAN LAS RNA

VISCOSEQ Visualiza secuencia de movimiento en coordenadas

Sintaxis

```
viscoseq(coseq,n)
```

Descripción

VISCOSEQ Visualiza secuencia de movimiento en coordenadas

VISCOSEQ(coseq,n), toma las entradas,
coseq - 2xM matriz, coordenadas vectoriales
n - Escalar, número de worker
y retorna, secuencia de M gráficas de posiciones de rata

SPEEDY_GUIDE Simula RNA (Interfaz grafica)

Descripción

SPEEDY_GUIDE, toma las entradas,
Red - string, nombre y directorio de la RNA
M - escalar, # de decisiones
a - escalar, tipo de ambiente
ohd - escalar, dirección de la cabeza
ind - escalar, brecha de indecisión
so - string, handles de radiobutton seleccionado
minlim - escalar, valor mínimo para interpretar la decisión como movimiento
y visualiza en ventana la secuencia de movimiento de la rata virtual

Algoritmo

Speedy_GUIDE simula RNA. Red debe contener el directorio y nombre de la RNA a simular. M es un número entero positivo.

Actualmente se encuentran definidos 3 tipos de ambientes para simular en el LCE, y cada uno de ellos corresponde a una misma topografía pero con diferentes niveles de iluminación, definidos por el nivel de iluminación en la zona central (4) y que pueden ser:

'HI' (high-118 Lux) 'MI' (medium-95 Lux) 'LI' (low-37 Lux)

También existe una limitación respecto al tamaño del ambiente, y es que este debe representarse en una matriz de 7x9x6. La posición

inicial de la rata para simular puede ser seleccionada a través de un panel de radiobuttons. Si la zona seleccionada corresponde en el ambiente a una zona prohibida por ser un hueco, entonces por defecto se escogerá la posición [4 5] que equivale a la zona central del LCE.

minlim e ind son números reales entre 0 y 1. minlim corresponde al mínimo valor necesario para interpretar un elemento del vector salida de la RNA como deseo de movimiento, y se debe entender como el umbral entre la inhibición y la activación del movimiento, mientras que ind corresponde a la diferencia de valores necesaria entre dos elementos de vector salida de la RNA para que el mayor de los dos prime como una voluntad superior y esta corresponda al movimiento ejecutado, por el contrario si su diferencia es menor que ind, se presenta una indecisión sobre cual dirección de movimiento elegir y entonces se elige entre las dos de manera aleatoria.

La opción "Verificar" permite desplegar una ventana que contiene los porcentajes de entradas y de tiempos de permanencia de los Brazos Abiertos y de los Brazos Cerrados y que se encuentran relacionados por medio de desigualdades.

La opción "Guardar" permite salvar la secuencia de movimientos en Nomenclatura Conde de la rata virtual simulada en un archivo *.xls.

ALGORITMOS QUE DIAGNOSTICAN LAS RNA

LEAN Diagnostica el aprendizaje de una RNA a lo largo del Entrenamiento (Interfaz grafica)

Descripción

LEAN, toma las entradas,
folder - string, carpeta que contiene las RNA
epoch - escalar, máximo epoch de entrenamiento
M - escalar, # de decisiones
a - escalar, tipo de ambiente
ohd - escalar, dirección de la cabeza
ind - escalar, brecha de indecisión
so - string, handles de radiobutton seleccionado
gap - escalar, valor de la brecha
minlim - escalar, valor mínimo para interpretar la decisión como movimiento
y salva en un archivo *.xls un vector columna de 0s y 1s

Algoritmo

LEAN diagnostica el aprendizaje de una RNA a lo largo del entrenamiento y representa el resultado a través de un vector columna de 0s y 1s. folder debe contener las RNA por cada epoch producto de un entrenamiento, y se obtiene a través de la opción "Cargar Carpeta". M es un número entero positivo.

Actualmente se encuentran definidos 3 tipos de ambientes para simular en el LCE, y cada uno de ellos corresponde a una misma topografía pero con diferentes niveles de iluminación, definidos por el nivel de iluminación en la zona central (4) y que pueden ser:

'HI' (high-118 Lux) 'MI' (medium-95 Lux) 'LI' (low-37 Lux)

También existe una limitación respecto al tamaño del ambiente, y es que este debe representarse en una matriz de 7x9x6. La posición inicial de la rata para simular puede ser seleccionada a través de un panel de radiobuttons. Si la zona seleccionada corresponde en el ambiente a una zona prohibida por ser un hueco, entonces por defecto se escogerá la posición [4 5] que equivale a la zona central del LCE.

minlim e ind son números reales entre 0 y 1. minlim corresponde al mínimo valor necesario para interpretar un elemento del vector salida de la RNA como deseo de movimiento, y se debe entender como el umbral entre la inhibición y la activación del movimiento, mientras que ind corresponde a la diferencia de valores necesaria entre dos elementos de vector salida de la RNA para que el mayor de los dos prime como una voluntad superior y esta corresponda al movimiento ejecutado, por el

contrario si su diferencia es menor que ind, se presenta una indecisión sobre cual dirección de movimiento elegir y entonces se elige entre las dos de manera aleatoria.

epoch es un numero entero positivo y debe ser menor que la cantidad de RNA contenidas en la carpeta. Gap debe ser un número real entre 0 y 1, y corresponde a la altura de las zonas que representan los estados binarios '0' y '1'.

LEAN salva en un archivo *.xls una matriz de 3 vectores columna, donde el primero corresponde a un vector con 0s y 1s, donde 0 indica que la RNA correspondiente a ese epoch de entrenamiento no aprendió y 1 que si aprendió para un determinado minlim e ind de simulación, y bajo los criterios;

ptba < ptbc feba < febc

los cuales conforman el diagnostico (Dx), donde;

ptba - Porcentaje de tiempo gastado en brazo abierto
ptbc - Porcentaje de tiempo gastado en brazo cerrado
feba - Frecuencia de entrada a los brazos abiertos
febc - Frecuencia de entrada a los brazos cerrados

El segundo vector corresponde a los valores del error mse por epoch y el tercero corresponde a los valores del error respecto al gap (ergap). El ergap corresponde a una medida del error que presentan los datos al encontrarse estos fuera de la zona binaria ('0' o '1') a la que pertenecen de acuerdo al Target en el entrenamiento.

REBP Genera parámetros para validar el comportamiento de ratas en LCE

Sintaxis

```
param = rebp(J,T)
```

Descripción

REBP Genera parámetros para validar el comportamiento de ratas en LCE

REBP(J,T), toma las entradas,

J - S vector columna, secuencia de posiciones en nomenclatura Conde

T - S vector columna, respectivos tiempos de permanencia por posición

y retorna,

param - 9 vector fila, parámetros de comportamiento exploratorio de ratas en LCE

Ejemplos

Dada la secuencia de movimientos con sus respectivos tiempos

```
J = [4 3 2 1 2 3 4 5 6 5 4 3 2 1];
```

```
T = [1 4 1.3 6.7 1.5 3.1 1.6 2.5 0.8 1.4 2 3.3 0.8 7.5];
```

el vector param corresponde a;

```
param = [4.7 28.2 12.53 75.2 1 2 33.33 66.66 3]
```

Algoritmo

rebp (rat exploratory behavior parameters) retorna el vector param con los parámetros para analizar el comportamiento exploratorio de ratas en LCE, los cuales son:

- Porcentaje de tiempo gastado en los brazos
- Tiempos de permanencia
- Frecuencia o Porcentaje de entrada a los brazos
- Numero de entrada a los brazos
- Número total de cruzamientos

El vector de parámetros es de la forma;

```
param = [tba tbc ptba ptbc neba nebc feba febc nc]
```

donde;

tba - Tiempo de permanencia en brazo abierto
tbc - Tiempo de permanencia en brazo cerrado
ptba - Porcentaje de tiempo gastado en brazo abierto
ptbc - Porcentaje de tiempo gastado en brazo cerrado
neba - Numero de entradas a los brazos abiertos
nebc - Numero de entradas a los brazos cerrados
feba - Frecuencia de entrada a los brazos abiertos
febc - Frecuencia de entrada a los brazos cerrados
nc - Número total de cruzamientos

Matemáticamente se pueden obtener los parámetros a través de las ecuaciones;

T_n - Tiempo de permanencia total en la posición n
 $T = T_1 + T_2 + T_3 + T_4 + T_5 + T_6 + T_7$
 $tba = T_5 + T_6 + T_7$
 $tbc = T_1 + T_2 + T_3$
 $ptba = tba \cdot 100 / T$
 $ptbc = tbc \cdot 100 / T$
neba = # de veces que después de 4 sigue 5
nebc = # de veces que después de 4 sigue 3
 $nc = neba + nebc$
 $feba = neba \cdot 100 / nc$
 $febc = nebc \cdot 100 / nc$

S es la cantidad de movimientos en la secuencia. Tiempo en s.

ALGORITMOS DESARROLLADOS COMO SOPORTE PARA LOS ALGORITMOS ANTERIORES

FIADGUI Organiza información referente al entrenamiento de redes

Descripción

FIADGUI, toma las entradas,
type - string, tipo de organización
fs - vector o string, selección de carpetas
nxfile - string, carpeta con archivos crudos
sheetype - string, tipo de datos
y retorna archivos con información del entrenamiento de RNA

Algoritmo

FIADGUI organiza los datos obtenidos de una RNA en el proceso de entrenamiento de acuerdo al type.

Existen tres type:

- xls (Archiva la información en archivos *.xls)
- mat (Archiva la información en archivos *.mat)
- stopmsg (Almacena criterios de parada)

stopmsg permite salvar en un archivo *.xls los criterios de parada de los entrenamientos de las RNA. **FIADGUI** carga archivos salvados en la carpeta 'TrainFiles'. fs puede ser un vector con los números de las carpetas o un string 'all'.

Para los type de 'xls' y 'mat' es posible además discriminar los datos a organizar (sheetype), los cuales pueden ser:

- InTa (Input y Target)
- p&s (pesos sinápticos y sesgos)
- s&e&m&g (salidas, errores, mse y gradiente)
- Todas (Permite organiza todos los tipos anteriores)

Debido a la gran cantidad de veces que se usa el comando xlswrite en el código es necesario detener su ejecución cuando este comando ha sido utilizado más de 26500 veces. Esto como consecuencia de la forma en que opera Matlab.

A la vez, debido a la gran cantidad de matrices de datos que se pueden obtener del entrenamiento de una RNA, al almacenar estas en hojas de calculo, es necesario tener en cuenta que estas tienen una capacidad máxima vertical de 65536, por lo cual tan pronto se llena una hoja es necesario pasar a la otra.

FIXCONDATA Arregla los datos de Conde eliminando repeticiones y secuencias

Sintaxis

`fixcondata(seq)`

Descripción

FIXCONDATA Arregla los datos de Conde eliminando repeticiones y secuencias

FIXCONDATA(seq), toma la entrada,
seq - escalar, habilita la eliminación de secuencias y carga datos contenidos en archivos *.xls que tengan como directorio la carpeta GenFixData y salva la información tratada en los mismos archivos

Ejemplos

Si se tiene una secuencia = 4 3 2 1 2 3 2 3 4 4 3 4 5 6 6 5 6 7 6 7

Al ejecutar `fixcondata` con la habilitación de eliminación de secuencias, la secuencia queda:

4 3 2 1 2 3 4 5 6 7

Algoritmo

`fixcondata` arregla los datos en nomenclatura Conde, eliminando posibles errores al repetir varias posiciones, y también elimina secuencias que no corresponden al comportamiento general exploratorio de la rata en LCE.

Para deshabilitar la eliminación de secuencias, se hace `seq = 0`, sino, `seq = 1`. Las secuencias a eliminar corresponden a:

`seq1 = [7 6 7]` `seq2 = [6 5 6]` `seq3 = [4 3 4]` `seq4 = [3 2 3]`

Lo anterior hace parte de una decisión bayesiana en la cual se considera que la gran mayoría de secuencias de exploración de ratas en el LCE que no se encuentran bajo los efectos de fármacos y con un nivel de iluminación en la zona central entre 30 y 500 Lux no contienen ninguna de las 4 secuencias mencionadas anteriormente.

CERLCE Administra interfaces graficas

Descripción

CERLCE a través de paneles y pushbuttons permite abrir 4 interfaces graficas

Entrenar: Corresponde a la función **Training**. Entrena RNA de acuerdo al criterio del usuario

Simular: Corresponde a la función **Speedy_GUIDE**. Visualiza la secuencia de movimiento de RNA entrenadas

Organizar: Corresponde a la función **Fiadgui**. Organiza datos en archivos *.xls y *.mat

Diagnosticar: Corresponde a la función **LeAn**. Diagnostica el aprendizaje

COOR2CONDE Convierte las coordenadas vectoriales a Nomenclatura Conde

Sintaxis

```
seq = coor2conde(coseq)
```

Descripción

COOR2CONDE Convierte las coordenadas vectoriales a Nomenclatura Conde

COOR2CONDE(coseq), toma las entradas,
coseq - Sx2 matriz, coordenadas vectoriales de la rata
y retorna,
seq - S vector fila, posiciones en nomenclatura Conde

Ejemplos

Si coseq == [1 5; 4 2], entonces seq será [1 7]

Algoritmo

Para convertir coordenadas de una nomenclatura a otra se utilizan los vectores:

```
Conde1 = [1 5;2 5;3 5;4 5;4 4;4 3;4 2];  
Conde2 = [7 5;6 5;5 5;4 5;4 6;4 7;4 8];
```

luego, si alguna coincide con una determinada posición de los vectores, esa es la correspondiente coordenada en nomenclatura Conde

CONDE2COOR Convierte las coordenadas en Nomenclatura Conde a vectoriales

Sintaxis

```
seq = conde2coor(coseq)
```

Descripción

CONDE2COOR Convierte las coordenadas en Nomenclatura Conde a vectoriales

CONDE2COOR(coseq), toma las entradas,
coseq - S vector fila, posiciones en nomenclatura Conde
y retorna,
seq - Sx2 matriz, coordenadas vectoriales de la rata

Ejemplos

Si coseq == [4 3 2], entonces seq será [4 5;3 5;2 5]

Algoritmo

Para convertir coordenadas de una nomenclatura a otra se utilizan los vectores:

```
pcondel = [1 5;2 5;3 5;4 5;4 4;4 3;4 2];  
pconde2 = [7 5;6 5;5 5;4 5;4 6;4 7;4 8];
```

luego, se alternan las coordenadas aleatoriamente entre los dos cada vez que en la secuencia se encuentre un 4, indicando que la rata se encuentra en el centro del LCE y basada en el principio de no-lateralidad, esta puede elegir ir hacia cualquiera de los dos brazos cerrados o abiertos indistintamente.

ANEXO B

INTERFACES GRÁFICAS

Con el fin de facilitarle al usuario del modelo ANA la implementación de los distintos protocolos diseñados, compuestos de etapas de entrenamiento de RNA, de simulación de RNA, de organización de la información en archivos *.mat y *.xls y de diagnóstico del aprendizaje de las RNA; se desarrollaron 5 interfaces gráficas, cada una de las cuales se explicarán detalladamente a continuación.

CERLCE

Corresponde a la ventana introductoria para el uso de las distintas aplicaciones, por medio de la cual se puede acceder a las otras interfaces dándole clic en el pushbutton deseado.

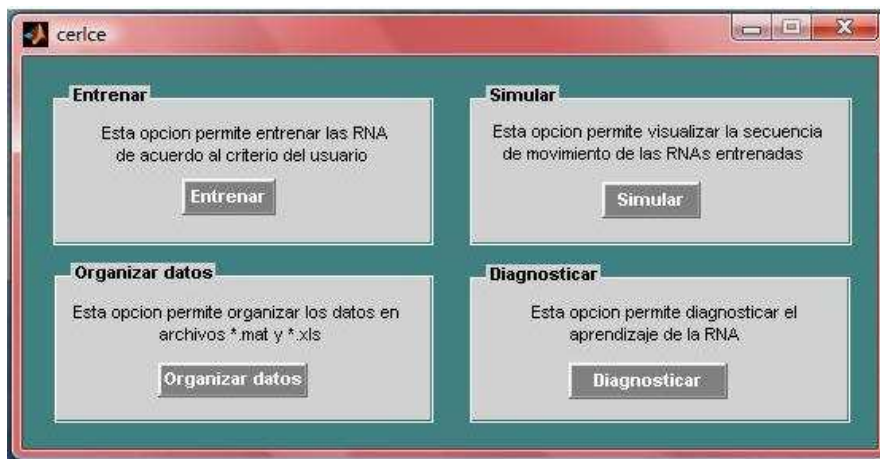


Fig 1. Interfaz cerlce

Su nombre CERLCE corresponde a las siglas de “Comportamiento exploratorio de la rata en el laberinto en cruz elevado”. A su vez, las otras ventanas también poseen nombres particulares, la ventana para entrenar se llama Training, para simular Speedy_GUIDE, para organizar datos Fiadgui y para diagnosticar LeAn.

TRAINING

A través de la introducción de algunos datos y seleccionando algunas opciones en esta ventana, es posible entrenar RNA. Para ello, en el panel de “Parámetros de entrenamiento” se debe escoger entre 3 niveles de iluminación del entorno “Alta”, “Media” y “Baja”; en este mismo panel se deben introducir el número de redes a entrenar, la cantidad máxima de epochs por entrenamiento y el goal deseado.

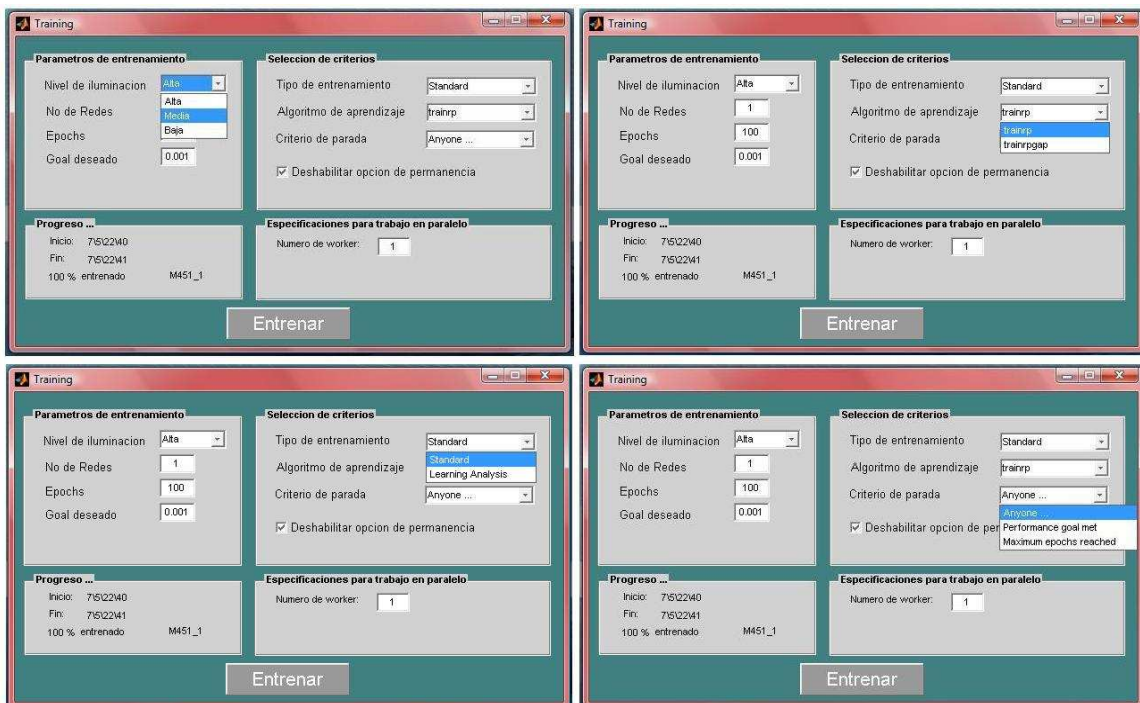


Fig 2. Interfaz Training

En el panel de “Selección de criterios” se puede escoger el tipo de entrenamiento entre “Standard” o “Learning Analysis”, también el algoritmo de aprendizaje entre “trainrp” o “trainrpgap” (la opción “trainrpgap” no se implementó en el actual proyecto pero se proyecta utilizar en posteriores desarrollos), también el criterio de parada entre “Anyone”, “Performance goal met” o “Maximum epochs reached” y por último es posible deshabilitar la opción de permanencia.

A través del panel de “Progreso...” se puede visualizar el día, mes, hora y minutos en que se inició y finalizó el entrenamiento, también el % de RNA entrenadas y el nombre del archivo *.xls que contiene la secuencia de movimientos en nomenclatura Conde con la cual se entrenará la RNA. Y por último, el panel de “Especificaciones para trabajo en paralelo” permite ejecutar múltiples procesos en simultáneo, aprovechando con esto las ventajas de los equipos multiprocesadores y multinúcleos. Para esto es necesario introducir en el número de worker un número entero positivo y diferente por cada ventana.

SPEEDY_GUIDE

Esta ventana permite visualizar la secuencia de movimientos obtenida de la simulación de la rata virtual para una determinada RNA entrenada. En la gráfica se muestra el entorno, donde las zonas negras corresponden a alturas, y las zonas amarillas corresponden a zonas de movilidad, las cuales además varían su tonalidad de acuerdo al nivel de iluminación de estas. Para el caso del LCE, este se muestra con los BC en dirección vertical y los BA en dirección horizontal. También se muestra la rata como un cuadrado rosado.

Para ejecutar la simulación se debe cargar la red que se encuentra almacenada en un archivo *.mat. Posteriormente se introducen los valores del mínimo valor e

indecisión (explicados en el Anexo A como las variables *minlim* e *ind*), y la cantidad de decisiones que la rata virtual puede tomar. Se debe seleccionar el tipo de ambiente entre “LCE HI”, “LCE MI”, “LCE LI” y “Otro...”, y seleccionar para el inicio de la simulación la dirección de la cabeza de la rata entre “Arriba”, “Abajo”, “Derecha” e “Izquierda”.

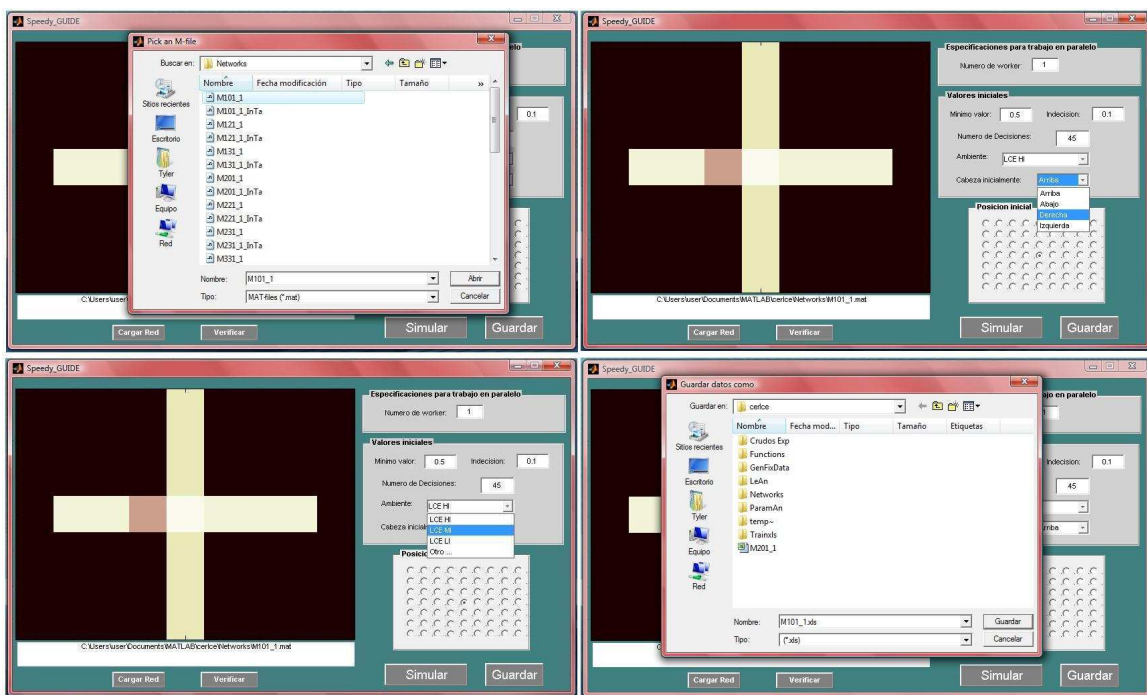


Fig 3. Interfaz Speedy_GUIDE

En la parte inferior derecha se sitúa un panel de radiobuttons, por medio del cual se indica la posición inicial de la rata. Para visualizar la simulación, se da clic en “Simular”, y una vez terminada, es posible salvar en un archivo *.xls la secuencia de movimientos expresados en nomenclatura Conde. Al igual que Training, Speedy_GUIDE también permite el trabajo en paralelo de procesos.

FIADGUI

A través de esta ventana es posible organizar la información del aprendizaje de las RNA en archivos *.mat y *.xls para su posterior procesamiento y análisis usando Matlab o Excel. En el panel de “Tipo de organización” se puede seleccionar entre tres tipos de archivos “xls”, “mat” y “stopmsg”, y también el tipo de datos a organizar, que pueden ser “InTa”, “p&s”, “s&e&m&g” y “Todas”. En el panel de “Directorio de datos” se introduce el nombre de la carpeta que contiene la información. Al igual que las otras interfaces, esta permite monitorear el progreso y trabajar en paralelo.

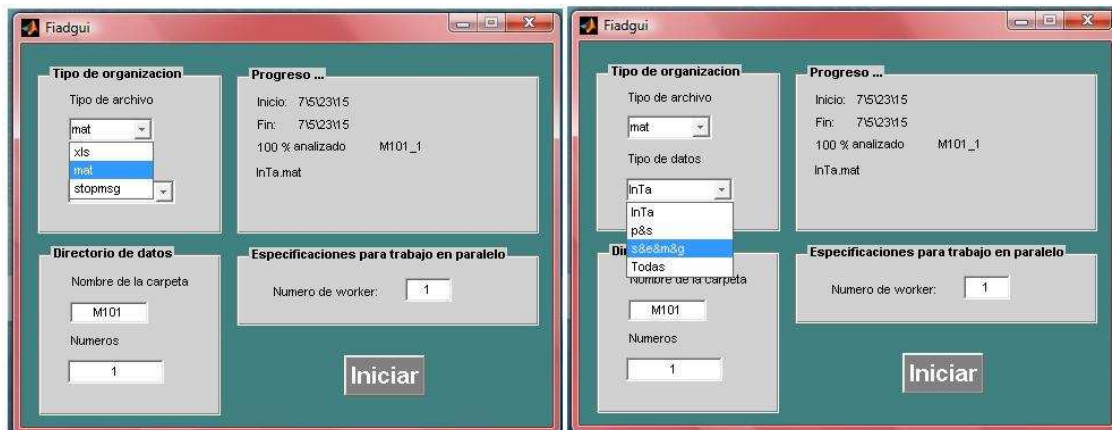


Fig 4. Interfaz Fiadgui

LEAN

Esta ventana permite realizar diagnósticos del aprendizaje de las RNA. Para ello es necesario introducir en el panel de “Valores iniciales” la cantidad de epochs de entrenamiento a diagnosticar, junto con el valor mínimo e indecisión. A la vez hay que escoger el ambiente, la dirección de la cabeza de la rata, el número de

decisiones y la posición inicial de la misma forma que se hace en “Speedy_GUIDE”. Esta interfaz también permite el trabajo en paralelo.

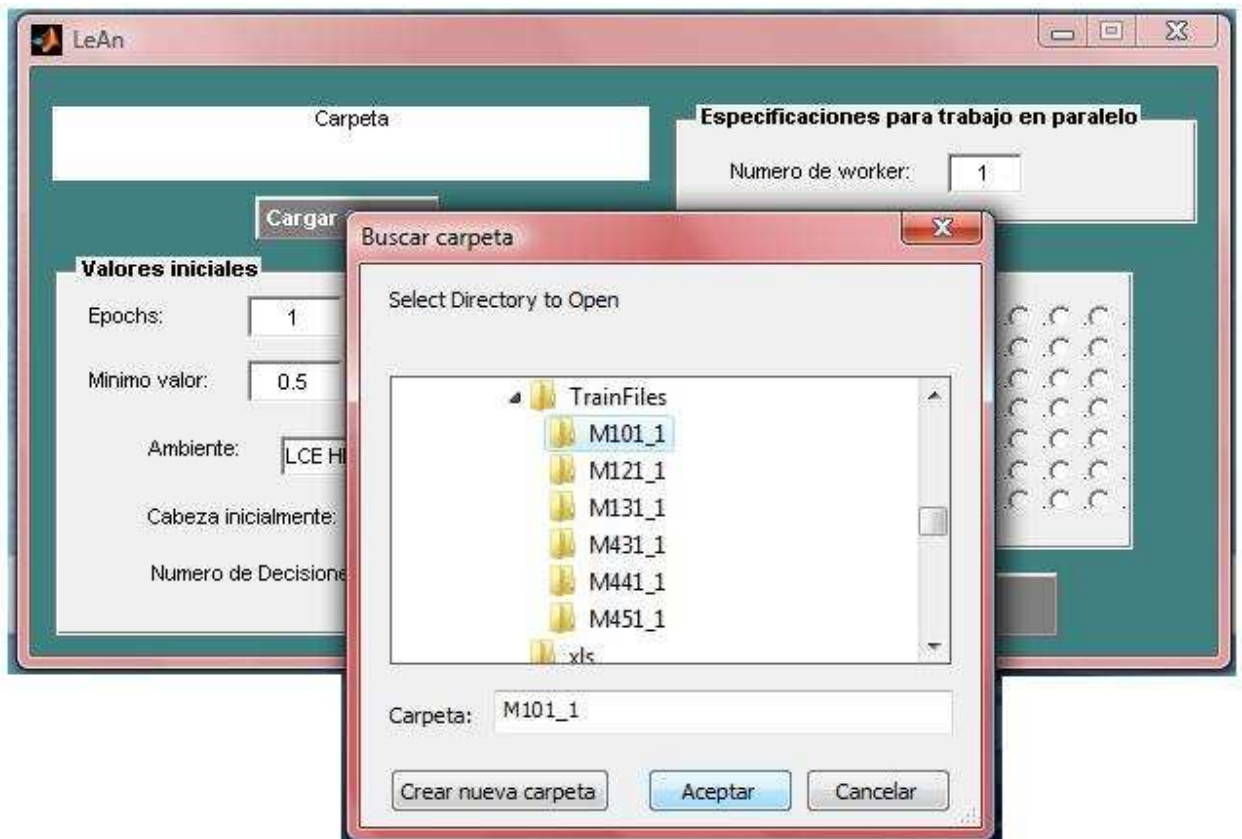


Fig 5. Interfaz LeAn

ANEXO C

Este trabajo fue presentado como exposición oral en el XXII Congreso Nacional de Física, que se realizó del 22 al 26 de Octubre del 2007 en Ibagué, Tolima, y fue una buena referencia para la participación en el II Latin American School of Computational Neuroscience LASCON 2008 que se realizó del 13 de Julio al 1 de Agosto del 2008 en Ribeirão Preto, São Paulo, Brasil.

Para mayor información, revisar los links:

<http://calima.univalle.edu.co/scf/eventos/2007/www/index.htm>

<http://neuron.ffclrp.usp.br/page.php?15>

http://funding.ibro.info/Pub/Pub_Main_Display.asp?LC_Docs_ID=3438

ANEXO D

Este anexo contiene las tablas de los valores numéricos con los cuales se realizaron los análisis numéricos y graficas presentadas en el capítulo 5 de implementación.

1. Pruebas ROC

Análisis ROC para 100 epochs

GRUPO	VARIABLE	ROC	VC1	sc	ec	blanco	ROC<0.7	sen<0.8	tot
M101_1	DxInv-mse	1,00	0,10	1,00	1,00				
M121_1	mse	0,92	0,33	0,25	1,00			1,00	1,00
M131_1	mse	0,87	0,35	0,00	0,99			1,00	1,00
M201_1	DxInv-mse	0,98	0,09	0,93	0,96				
M221_1	DxInv-mse	0,96	0,08	0,93	1,00				
M231_1	DxInv-mse	0,73	0,07	0,40	1,00			1,00	1,00
M331_1	DxInv-mse	0,79	0,22	1,00	0,08				
M341_1	DxInv-mse	0,97	0,08	0,97	1,00				
M431_1	DxInv-mse	0,97	0,08	0,96	1,00				
M441_1	mse	0,92	0,10	1,00	0,91				
M451_1	DxInv-mse	1,00	0,08	1,00	1,00				
M501_1	mse	0,98	0,12	1,00	0,98				
M541_1	mse	1,00	0,26	1,00	1,00				
M551_1	mse		0,39	0,00	0,99	1,00	1,00	1,00	1,00
M601_1	DxInv-mse	0,96	0,08	0,94	1,00				
M611_1	mse		0,36	0,00	0,99	1,00	1,00	1,00	1,00
M641_1	mse	0,78	0,17	0,50	0,98			1,00	1,00

GRUPO	VARIABLE	ROC	VC2	sc	ec	blanco	ROC <0.7	sen<0.8	tot
M101_2	DxInv-mse	0,96	0,10	0,96	0,97				
M121_2	mse	0,96	0,40	0,00	0,99			1,00	1,00
M131_2	mse	0,98	0,16	0,50	0,99			1,00	1,00
M201_2	mse	0,98	0,11	1,00	0,97				
M221_2	DxInv-mse	0,92	0,08	1,00	0,86				
M231_2	DxInv-mse	0,97	0,06	0,62	1,00			1,00	1,00
M331_2	DxInv-mse	0,84	0,42	1,00	0,11				
M341_2	DxInv-mse	0,64	0,10	0,96	0,33		1,00		1,00
M431_2	DxInv-mse	0,99	0,08	0,96	0,98				
M441_2	mse	0,78	0,20	0,43	0,99			1,00	1,00
M451_2	DxInv-mse	0,91	0,08	0,96	0,73				
M501_2	DxInv-mse	0,92	0,09	0,86	0,97				
M541_2	mse	0,68	0,08	0,91	0,69		1,00		1,00
M551_2	mse	0,96	0,16	0,50	0,98			1,00	1,00
M601_2	DxInv-mse	0,79	0,08	0,83	0,85				
M611_2	mse	0,96	0,47	0,00	0,99			1,00	1,00
M641_2	DxInv-mse	0,76	0,08	0,00	1,00			1,00	1,00

GRUPO	VARIABLE	ROC	VC3	sc	ec	blanco	ROC <0.7	sen<0.8	tot
M101_3	DxInv-mse	0,85	0,26	0,99	0,13				
M121_3	DxInv-mse	0,85	0,09	0,74	0,98			1,00	1,00
M131_3	mse	0,64	0,09	0,84	0,75		1,00		1,00
M201_3	mse	0,63	0,11	0,60	0,97		1,00	1,00	1,00
M221_3	mse	0,83	0,09	1,00	0,76				
M231_3	DxInv-mse	0,96	0,08	0,93	1,00				
M331_3	DxInv-mse	0,81	0,47	0,99	0,00				
M341_3	DxInv-mse	0,92	0,09	0,96	0,82				
M431_3	DxInv-mse	0,68	0,09	0,94	0,50		1,00		1,00
M441_3	mse	0,87	0,09	0,88	0,84				
M451_3	DxInv-mse	0,82	0,08	0,89	0,81				
M501_3	mse	0,86	0,46	0,00	0,99			1,00	1,00
M541_3	mse	0,98	0,14	1,00	0,98				
M551_3	DxInv-mse	0,88	0,09	0,86	1,00				
M601_3	DxInv-mse	0,86	0,08	0,76	0,92			1,00	1,00
M611_3	DxInv-mse	0,88	0,05	0,00	1,00			1,00	1,00
M641_3	mse	0,99	0,17	1,00	0,99				

GRUPO	VARIABLE	ROC	VC4	sc	ec	blanco	ROC <0.7	sen<0.8	tot
M101_4	DxInv-mse	0,97	0,12	0,97	1,00				
M121_4	DxInv-mse	0,66	0,09	0,70	0,82		1,00	1,00	1,00
M131_4	mse	0,83	0,10	0,86	0,92				
M201_4	DxInv-mse	0,86	0,07	0,82	1,00				
M221_4	DxInv-mse	0,97	0,08	0,97	1,00				
M231_4	DxInv-mse	0,82	0,08	0,92	0,83				
M331_4	DxInv-mse	0,86	0,11	0,81	1,00				
M341_4	DxInv-mse	1,00	0,07	0,97	1,00				
M431_4	DxInv-mse	0,78	0,09	0,90	0,73				
M441_4	mse		0,44	0,00	0,99	1,00	1,00	1,00	1,00
M451_4	DxInv-mse	0,93	0,08	0,96	0,78				
M501_4	mse	0,97	0,14	0,75	0,98			1,00	1,00
M541_4	DxInv-mse	0,89	0,09	0,86	0,99				
M551_4	mse	0,85	0,63	0,00	0,99			1,00	1,00
M601_4	mse	0,95	0,11	0,57	0,99			1,00	1,00
M611_4	DxInv-mse	0,69	0,06	0,00	1,00		1,00	1,00	1,00
M641_4	mse	0,71	0,12	0,40	0,98			1,00	1,00

Análisis ROC para 500 epochs

GRUPO	VARIABLE	ROC	VC1	sc	ec	blanco	ROC <0.7	sen<0.8	tot
M101_1	DxInv-mse	1,00	0,20	1,00					
M121_1	mse	0,50	0,03	1,00	0,00		1,00		1,00
M131_1	DxInv-mse	0,92	0,65	1,00	0,03				
M201_1	DxInv-mse	0,87	0,19	1,00	0,00				
M221_1	DxInv-mse	0,94	0,05	0,93	0,97				
M231_1	DxInv-mse	0,87	0,03	0,95	0,48				
M331_1	DxInv-mse	0,86	0,23	1,00	0,00				
M341_1	DxInv-mse	0,95	0,36	1,00	0,12				
M431_1	DxInv-mse	0,83	0,11	1,00	0,14				
M441_1	DxInv-mse	0,86	0,03	0,72	0,99			1,00	1,00
M451_1	DxInv-mse	0,85	0,02	0,82	0,91				
M501_1	DxInv-mse	0,90	0,03	0,79	0,98			1,00	1,00
M541_1	DxInv-mse	0,97	0,05	0,97	1,00				
M551_1	DxInv-mse	0,98	0,04	0,97	0,99				
M601_1	DxInv-mse	0,94	0,05	0,94	1,00				
M611_1	DxInv-mse	0,71	0,28	1,00	0,00				
M641_1	DxInv-mse	0,51	0,04	0,90	0,44		1,00		1,00

GRUPO	VARIABLE	ROC	VC2	sc	ec	blanco	ROC <0.7	sen<0.8	tot
M101_2	DxInv-mse	0,93	0,06	0,96	0,91				
M121_2	DxInv-mse	0,53	0,04	0,61	0,73		1,00	1,00	1,00
M131_2	DxInv-mse	0,96	0,40	1,00	0,10				
M201_2	DxInv-mse	0,96	0,02	0,89	1,00				
M221_2	DxInv-mse	0,92	0,04	0,96	0,70				
M231_2	DxInv-mse	0,92	0,05	0,98	0,47				
M331_2	DxInv-mse	0,91	0,42	1,00	0,10				
M341_2	mse	0,70	0,03	0,97	0,69				
M431_2	DxInv-mse	0,94	0,09	0,99	0,40				
M441_2	DxInv-mse	0,71	0,03	0,68	0,97			1,00	1,00
M451_2	DxInv-mse	0,93	0,54	1,00	0,20				
M501_2	DxInv-mse	0,73	0,04	0,87	0,78				
M541_2	DxInv-mse	0,79	0,17	1,00	0,01				
M551_2	DxInv-mse	0,96	0,05	0,96	0,99				
M601_2	DxInv-mse	0,90	0,45	1,00	0,03				
M611_2	DxInv-mse	0,95	0,03	0,91	1,00				
M641_2	DxInv-mse	0,98	0,04	0,98	1,00				

Datos procesados para 100 epochs

VC1	VC2	VC3	VC4	SD	PROM	%SD
0,10	0,10	0,26	0,12	0,08	0,15	0,54
0,33	0,40	0,09	0,09	0,162	0,22	0,72
0,35	0,16	0,09	0,10	0,12	0,17	0,68
0,09	0,11	0,11	0,07	0,02	0,09	0,21
0,08	0,08	0,09	0,08	0,008	0,08	0,09
0,07	0,06	0,08	0,08	0,009	0,07	0,12
0,22	0,42	0,47	0,11	0,17	0,30	0,56
0,08	0,10	0,09	0,07	0,01	0,09	0,12
0,08	0,08	0,09	0,09	0,006	0,08	0,07
0,10	0,20	0,09	0,44	0,165	0,21	0,80
0,08	0,08	0,08	0,08	0,001	0,08	0,01
0,12	0,09	0,46	0,14	0,173	0,20	0,86
0,26	0,08	0,14	0,09	0,085	0,14	0,60
0,39	0,16	0,09	0,63	0,245	0,32	0,77
0,08	0,08	0,08	0,11	0,016	0,09	0,18
0,36	0,47	0,05	0,06	0,21	0,24	0,89
0,17	0,08	0,17	0,12	0,046	0,14	0,34

Datos procesados para 500 epochs

VC1	VC2	SD	PROM	%SD
0,20	0,06	0,10	0,13	0,80
0,03	0,04	0,01	0,03	0,24
0,65	0,40	0,18	0,53	0,33
0,19	0,02	0,12	0,10	1,17
0,05	0,04	0,00	0,04	0,03
0,03	0,05	0,01	0,04	0,24
0,23	0,42	0,14	0,32	0,43
0,36	0,03	0,23	0,19	1,18
0,11	0,09	0,01	0,10	0,14
0,03	0,03	0,00	0,03	0,02
0,02	0,54	0,36	0,28	1,30
0,03	0,04	0,01	0,04	0,21
0,05	0,17	0,08	0,11	0,75
0,04	0,05	0,00	0,04	0,09
0,05	0,45	0,29	0,25	1,15
0,28	0,03	0,18	0,15	1,15
0,04	0,04	0,00	0,04	0,00

2. Correlaciones de Pearson y ANOVA

Parámetros comportamentales para rata real y mse 0.4 – 0.1 – 0.05 – 0.01 – 0.005 – 0.001 con los cuales se obtuvieron las correlaciones y ANOVA.

Rata - real	tba	tbc	ptba	ptbc	neba	nebc	feba	febc	nc
M101	31,50	247,38	10,50	82,46	4,00	10,00	28,57	71,43	14,00
M121	54,49	173,53	18,16	57,84	6,00	13,00	31,58	68,42	19,00
M131	5,03	284,33	1,68	94,76	1,00	10,00	9,09	90,91	11,00
M431	62,93	202,12	20,97	67,37	4,00	9,00	30,77	69,23	13,00
M441	60,82	192,96	20,28	64,33	6,00	11,00	35,29	64,71	17,00
M451	11,49	201,47	5,21	91,43	1,00	11,00	8,33	91,67	12,00
M331	14,71	254,26	4,90	84,75	2,00	7,00	22,22	77,78	9,00
M341	60,39	208,57	20,23	69,85	8,00	18,00	30,77	69,23	26,00
M601	68,96	190,29	22,99	63,43	5,00	14,00	26,32	73,68	19,00
M611	84,77	180,73	28,26	60,24	10,00	14,00	41,67	58,33	24,00
M641	88,64	159,87	29,55	53,29	9,00	11,00	45,00	55,00	20,00
M201	62,18	202,81	20,73	67,60	4,00	9,00	30,77	69,23	13,00
M221	52,08	213,39	17,36	71,14	6,00	13,00	31,58	68,42	19,00
M231	58,23	203,88	19,41	67,96	4,00	12,00	25,00	75,00	16,00
M501	78,60	197,97	26,20	65,98	6,00	6,00	50,00	50,00	12,00
M541	46,23	183,68	15,41	61,23	6,00	10,00	37,50	62,50	16,00
M551	65,78	185,30	21,93	61,76	6,00	13,00	31,58	68,42	19,00

Rata - 0.4	tba	tbc	ptba	ptbc	neba	nebc	feba	febc	nc
simHI_M101	0,00	296,34	0,00	98,78	0,00	1,00	0,00	100,00	1,00
simHI_M121	0,00	148,45	0,00	49,48	0,00	48,00	0,00	100,00	48,00
simHI_M131	0,00	237,80	0,00	79,27	0,00	17,00	0,00	100,00	17,00
simHI_M431	0,00	296,30	0,00	98,77	0,00	1,00	0,00	100,00	1,00
simHI_M441	61,22	214,29	20,41	71,43	7,00	1,00	87,50	12,50	8,00
simHI_M451	41,49	156,38	13,83	52,13	13,00	19,00	40,63	59,38	32,00
simLI_M331	295,65	0,00	98,55	0,00	1,00	0,00	100,00	0,00	1,00
simLI_M341	297,76	0,00	99,25	0,00	1,00	0,00	100,00	0,00	1,00
simLI_M601	113,64	90,91	37,88	30,30	1,00	20,00	4,76	95,24	21,00
simLI_M611	0,00	296,70	0,00	98,90	0,00	1,00	0,00	100,00	1,00
simLI_M641	0,00	109,46	0,00	36,49	0,00	27,00	0,00	100,00	27,00
simMI_M201	131,25	97,50	43,75	32,50	15,00	4,00	78,95	21,05	19,00
simMI_M221	53,13	84,38	17,71	28,13	17,00	27,00	38,64	61,36	44,00
simMI_M231	0,00	255,45	0,00	85,15	0,00	15,00	0,00	100,00	15,00
simMI_M501	193,27	34,62	64,42	11,54	13,00	12,00	52,00	48,00	25,00
simMI_M541	0,00	81,43	0,00	27,14	0,00	38,00	0,00	100,00	38,00
simMI_M551	0,00	297,48	0,00	99,16	0,00	1,00	0,00	100,00	1,00

Rata - 0.1	tba	tbc	ptba	ptbc	neba	nebc	feba	febc	nc
simHI_M101	0,00	193,90	0,00	64,63	0,00	6,00	0,00	100,00	6,00
simHI_M121	0,00	296,88	0,00	98,96	0,00	1,00	0,00	100,00	1,00
simHI_M131	0,00	296,34	0,00	98,78	0,00	1,00	0,00	100,00	1,00
simHI_M431	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
simHI_M441	0,00	39,56	0,00	13,19	0,00	1,00	0,00	100,00	1,00
simHI_M451	0,00	247,30	0,00	82,43	0,00	1,00	0,00	100,00	1,00
simLI_M331	0,00	269,57	0,00	89,86	0,00	1,00	0,00	100,00	1,00
simLI_M341	0,00	297,76	0,00	99,25	0,00	1,00	0,00	100,00	1,00
simLI_M601	0,00	297,12	0,00	99,04	0,00	1,00	0,00	100,00	1,00
simLI_M611	0,00	8,57	0,00	2,86	0,00	1,00	0,00	100,00	1,00
simLI_M641	90,76	0,00	30,25	0,00	1,00	0,00	100,00	0,00	1,00
simMI_M201	0,00	180,00	0,00	60,00	0,00	1,00	0,00	100,00	1,00
simMI_M221	0,00	183,67	0,00	61,22	0,00	1,00	0,00	100,00	1,00
simMI_M231	0,00	296,81	0,00	98,94	0,00	1,00	0,00	100,00	1,00
simMI_M501	0,00	296,25	0,00	98,75	0,00	1,00	0,00	100,00	1,00
simMI_M541	0,00	237,50	0,00	79,17	0,00	1,00	0,00	100,00	1,00
simMI_M551	0,00	264,36	0,00	88,12	0,00	1,00	0,00	100,00	1,00

Rata - 0.05	tba	tbc	ptba	ptbc	neba	nebc	feba	febc	nc
simHI_M101	0,00	201,22	0,00	67,07	0,00	3,00	0,00	100,00	3,00
simHI_M121	25,00	121,88	8,33	40,63	2,00	3,00	40,00	60,00	5,00
simHI_M131	10,98	245,12	3,66	81,71	1,00	11,00	8,33	91,67	12,00
simHI_M431	0,00	263,64	0,00	87,88	0,00	2,00	0,00	100,00	2,00
simHI_M441	49,45	191,21	16,48	63,74	5,00	10,00	33,33	66,67	15,00
simHI_M451	0,00	202,70	0,00	67,57	0,00	10,00	0,00	100,00	10,00
simLI_M331	8,70	247,83	2,90	82,61	2,00	8,00	20,00	80,00	10,00
simLI_M341	125,37	107,46	41,79	35,82	4,00	9,00	30,77	69,23	13,00
simLI_M601	17,31	164,42	5,77	54,81	2,00	2,00	50,00	50,00	4,00
simLI_M611	2,14	278,57	0,71	92,86	1,00	1,00	50,00	50,00	2,00
simLI_M641	0,00	73,11	0,00	24,37	0,00	6,00	0,00	100,00	6,00
simMI_M201	56,25	150,00	18,75	50,00	3,00	4,00	42,86	57,14	7,00
simMI_M221	6,12	64,29	2,04	21,43	2,00	2,00	50,00	50,00	4,00
simMI_M231	47,87	181,91	15,96	60,64	3,00	12,00	20,00	80,00	15,00
simMI_M501	78,75	18,75	26,25	6,25	17,00	1,00	94,44	5,56	18,00
simMI_M541	9,38	178,13	3,13	59,38	1,00	5,00	16,67	83,33	6,00
simMI_M551	17,82	151,49	5,94	50,50	2,00	5,00	28,57	71,43	7,00

Rata - 0.01	tba	tbc	ptba	ptbc	neba	nebc	feba	febc	nc
simHI_M101	14,63	193,90	4,88	64,63	4,00	9,00	30,77	69,23	13,00
simHI_M121	65,63	159,38	21,88	53,13	8,00	14,00	36,36	63,64	22,00
simHI_M131	10,98	248,78	3,66	82,93	1,00	10,00	9,09	90,91	11,00
simHI_M431	36,36	154,55	12,12	51,52	3,00	18,00	14,29	85,71	21,00
simHI_M441	59,34	181,32	19,78	60,44	6,00	11,00	35,29	64,71	17,00
simHI_M451	145,95	121,62	48,65	40,54	2,00	6,00	25,00	75,00	8,00
simLI_M331	8,70	252,17	2,90	84,06	2,00	7,00	22,22	77,78	9,00
simLI_M341	64,93	170,15	21,64	56,72	6,00	14,00	30,00	70,00	20,00
simLI_M601	54,81	190,38	18,27	63,46	5,00	14,00	26,32	73,68	19,00
simLI_M611	66,43	156,43	22,14	52,14	9,00	15,00	37,50	62,50	24,00
simLI_M641	83,19	151,26	27,73	50,42	9,00	12,00	42,86	57,14	21,00
simMI_M201	78,75	165,00	26,25	55,00	5,00	9,00	35,71	64,29	14,00
simMI_M221	52,04	189,80	17,35	63,27	5,00	10,00	33,33	66,67	15,00
simMI_M231	3,19	175,53	1,06	58,51	1,00	11,00	8,33	91,67	12,00
simMI_M501	48,75	195,00	16,25	65,00	5,00	4,00	55,56	44,44	9,00
simMI_M541	87,50	109,38	29,17	36,46	9,00	7,00	56,25	43,75	16,00
simMI_M551	56,44	184,16	18,81	61,39	6,00	14,00	30,00	70,00	20,00

Rata - 0.005	tba	tbc	ptba	ptbc	neba	nebc	feba	febc	nc
simHI_M101	43,90	204,88	14,63	68,29	4,00	10,00	28,57	71,43	14,00
simHI_M121	25,00	162,50	8,33	54,17	2,00	14,00	12,50	87,50	16,00
simHI_M131	10,98	248,78	3,66	82,93	1,00	10,00	9,09	90,91	11,00
simHI_M431	54,55	163,64	18,18	54,55	4,00	8,00	33,33	66,67	12,00
simHI_M441	59,34	181,32	19,78	60,44	6,00	12,00	33,33	66,67	18,00
simHI_M451	20,27	227,03	6,76	75,68	1,00	11,00	8,33	91,67	12,00
simLI_M331	8,70	252,17	2,90	84,06	2,00	6,00	25,00	75,00	8,00
simLI_M341	58,21	179,10	19,40	59,70	8,00	17,00	32,00	68,00	25,00
simLI_M601	54,81	193,27	18,27	64,42	5,00	13,00	27,78	72,22	18,00
simLI_M611	51,43	214,29	17,14	71,43	6,00	9,00	40,00	60,00	15,00
simLI_M641	78,15	138,66	26,05	46,22	7,00	11,00	38,89	61,11	18,00
simMI_M201	67,50	180,00	22,50	60,00	4,00	10,00	28,57	71,43	14,00
simMI_M221	9,18	241,83	3,06	80,61	3,00	10,00	23,08	76,92	13,00
simMI_M231	63,83	150,00	21,28	50,00	5,00	9,00	35,71	64,29	14,00
simMI_M501	56,25	187,50	18,75	62,50	5,00	6,00	45,45	54,55	11,00
simMI_M541	62,50	187,50	20,83	62,50	6,00	10,00	37,50	62,50	16,00
simMI_M551	62,38	181,19	20,79	60,40	6,00	13,00	31,58	68,42	19,00

Rata - 0.001	tba	tbc	ptba	ptbc	neba	nebc	feba	febc	nc
simHI_M101	43,90	204,88	14,63	68,29	4,00	10,00	28,57	71,43	14,00
simHI_M121	62,50	175,00	20,83	58,33	6,00	13,00	31,58	68,42	19,00
simHI_M131	10,98	248,78	3,66	82,93	1,00	10,00	9,09	90,91	11,00
simHI_M431	54,55	186,36	18,18	62,12	4,00	9,00	30,77	69,23	13,00
simHI_M441	59,34	181,32	19,78	60,44	6,00	11,00	35,29	64,71	17,00
simHI_M451	20,27	295,95	5,49	80,22	1,00	12,00	7,69	92,31	13,00
simLI_M331	8,70	252,17	2,90	84,06	2,00	7,00	22,22	77,78	9,00
simLI_M341	58,21	183,58	19,40	61,19	8,00	18,00	30,77	69,23	26,00
simLI_M601	54,81	190,38	18,27	63,46	5,00	14,00	26,32	73,68	19,00
simLI_M611	64,29	171,43	21,43	57,14	8,00	15,00	34,78	65,22	23,00
simLI_M641	93,28	156,30	31,09	52,10	9,00	11,00	45,00	55,00	20,00
simMI_M201	67,50	180,00	22,50	60,00	4,00	9,00	30,77	69,23	13,00
simMI_M221	48,98	192,86	16,33	64,29	6,00	13,00	31,58	68,42	19,00
simMI_M231	57,45	191,49	19,15	63,83	4,00	12,00	25,00	75,00	16,00
simMI_M501	67,50	187,50	22,50	62,50	6,00	6,00	50,00	50,00	12,00
simMI_M541	62,50	187,50	20,83	62,50	6,00	10,00	37,50	62,50	16,00
simMI_M551	62,38	181,19	20,79	60,40	6,00	13,00	31,58	68,42	19,00

Valores promedios y errores estándar (ee) de los parámetros comportamentales para rata real y mse 0.1 – 0.05 – 0.01 – 0.005 – 0.001

	real	0,1	0,05	0,01	0,005	0,001
TBC	204,855294	200,328134	167,159859	176,400105	193,744165	198,040764
ee	7,54763243	27,918148	17,7048145	8,90172108	8,142504	8,51108792
TBA	53,3429412	5,33860603	26,7727686	55,1531478	46,2921259	52,7713255
ee	5,96510143	5,33860603	8,4025783	8,54051586	5,43889583	5,22631754

	real	0,1	0,05	0,01	0,005	0,001
%TBC	69,7304034	66,7760447	55,7199529	58,8000351	64,5814561	64,9295398
ee	2,85724092	9,30604934	5,90160484	2,96724036	2,71419531	2,19127393
%TBA	17,8677042	1,77953534	8,92425621	18,3843826	15,430711	17,5161891
ee	1,95439039	1,77953534	2,80085943	2,84683862	1,81296417	1,77228498

	real	0,1	0,05	0,01	0,005	0,001
NEBC	11,2352941	1,17647059	5,52941176	10,8823529	10,5294118	11,3529412
ee	0,69414257	0,31195887	0,90796171	0,89476563	0,6703689	0,71198344
NEBA	5,17647059	0,05882353	2,64705882	5,05882353	4,41176471	5,05882353
ee	0,60776407	0,05882353	0,96230871	0,65005989	0,50043234	0,55882353

	real	0,1	0,05	0,01	0,005	0,001
%EBC	69,6447646	88,2352941	71,4720247	68,8891017	71,1338156	70,0874166
ee	2,61517101	8,05474349	6,07626828	3,26987441	2,578512	2,55613935
%EBA	30,3552354	5,88235294	28,5279753	31,1108983	28,8661844	29,9125834
ee	2,61517101	5,88235294	6,07626828	3,26987441	2,578512	2,55613935

	real	0,1	0,05	0,01	0,005	0,001
NC	16,4117647	1,23529412	8,17647059	15,9411765	14,9411765	16,4117647
ee	1,12151035	0,30352599	1,20426721	1,22615668	0,96050916	1,08485166

Valores promedios y errores estándar de los parámetros comportamentales para mse 0.4

	BA	BC	ee BA	ee BC
T	69,8475185	158,67531	24,92041	25,9403275
%T	23,2825062	52,8917702	8,30680335	8,64677585
NE	4	13,6470588	1,52431277	3,57682612
%E	29,5570963	70,4429037	9,56925203	9,56925203

Parámetros comportamentales para ratas reales con tiempos constantes 300/n

Rata	tba	tbc	%tba	%tbc	neba	nebc	%eba	%ebc	nc
M101	43,902439	204,878049	14,6341463	68,2926829	4	10	28,5714286	71,4285714	14
M121	62,5	178,125	20,8333333	59,375	6	13	31,5789474	68,4210526	19
M131	10,9756098	248,780488	3,65853659	82,9268293	1	10	9,09090909	90,9090909	11
M201	67,5	183,75	22,5	61,25	4	9	30,7692308	69,2307692	13
M221	48,9795918	192,857143	16,3265306	64,2857143	6	13	31,5789474	68,4210526	19
M231	57,4468085	191,489362	19,1489362	63,8297872	4	12	25	75	16
M331	8,69565217	252,173913	2,89855072	84,057971	2	7	22,2222222	77,7777778	9
M341	58,2089552	183,58209	19,4029851	61,1940299	8	18	30,7692308	69,2307692	26
M431	54,5454545	186,363636	18,1818182	62,1212121	4	9	30,7692308	69,2307692	13
M441	59,3406593	181,318681	19,7802198	60,4395604	6	11	35,2941176	64,7058824	17
M451	20,2702703	231,081081	6,75675676	77,027027	1	11	8,33333333	91,6666667	12
M501	67,5	187,5	22,5	62,5	6	6	50	50	12
M541	62,5	187,5	20,8333333	62,5	6	10	37,5	62,5	16
M551	62,3762376	181,188119	20,7920792	60,3960396	6	13	31,5789474	68,4210526	19
M601	54,8076923	190,384615	18,2692308	63,4615385	5	14	26,3157895	73,6842105	19
M611	81,4285714	167,142857	27,1428571	55,7142857	10	14	41,6666667	58,3333333	24
M641	93,2773109	156,302521	31,092437	52,1008403	9	11	45	55	20

Número de movimientos ejecutados por las ratas reales y tiempos de decisión tis

Rata	No de mov	tis
M101	82	3,658536585
M121	96	3,125
M131	82	3,658536585
M201	80	3,75
M221	98	3,06122449
M231	94	3,191489362
M331	69	4,347826087
M341	134	2,23880597
M431	66	4,545454545
M441	91	3,296703297
M451	74	4,054054054
M501	80	3,75
M541	96	3,125
M551	101	2,97029703
M601	104	2,884615385
M611	140	2,142857143
M641	119	2,521008403

3. Análisis de densidades

Posiciones	fid	bid	lid	rid	fhd	bhd
4	0,2135	0,2135	0,21717647	0,21717647	0,75	0,75
3	0,196	0,204	0,196	0,196	0	0
4	0,2135	0,2135	0,21717647	0,21717647	0,75	0,75
5	0,216	0,21475	0,21488889	0,21488889	0,86363636	0,5625
6	0,216	0,21513043	0,21488889	0,21488889	0,86666667	0,65217391
7	0,216	0,21533333	0,21488889	0,21488889	0,875	0,7
6	0,21513043	0,216	0,21488889	0,21488889	0,65217391	0,86666667
5	0,21475	0,216	0,21488889	0,21488889	0,5625	0,86363636
4	0,21717647	0,21717647	0,2135	0,2135	0,76470588	0,76470588
5	0,216	0,21475	0,21488889	0,21488889	0,86363636	0,5625
6	0,216	0,21513043	0,21488889	0,21488889	0,86666667	0,65217391
5	0,21475	0,216	0,21488889	0,21488889	0,5625	0,86363636
4	0,21717647	0,21717647	0,2135	0,2135	0,76470588	0,76470588
3	0,196	0,204	0,196	0,196	0	0
2	0,196	0,20266667	0,196	0,196	0	0
1	0,196	0,20171429	0,196	0,196	0	0
2	0,20266667	0,196	0,196	0,196	0	0
3	0,204	0,196	0,196	0,196	0	0
4	0,2135	0,2135	0,21717647	0,21717647	0,75	0,75
3	0,196	0,204	0,196	0,196	0	0
2	0,196	0,20266667	0,196	0,196	0	0
1	0,196	0,20171429	0,196	0,196	0	0
2	0,20266667	0,196	0,196	0,196	0	0
3	0,204	0,196	0,196	0,196	0	0
4	0,2135	0,2135	0,21717647	0,21717647	0,75	0,75
3	0,196	0,204	0,196	0,196	0	0
2	0,196	0,20266667	0,196	0,196	0	0
1	0,196	0,20171429	0,196	0,196	0	0
2	0,20266667	0,196	0,196	0,196	0	0
3	0,204	0,196	0,196	0,196	0	0
4	0,2135	0,2135	0,21717647	0,21717647	0,75	0,75
3	0,196	0,204	0,196	0,196	0	0
2	0,196	0,20266667	0,196	0,196	0	0
1	0,196	0,20171429	0,196	0,196	0	0
2	0,20266667	0,196	0,196	0,196	0	0
3	0,204	0,196	0,196	0,196	0	0
4	0,2135	0,2135	0,21717647	0,21717647	0,75	0,75
3	0,196	0,204	0,196	0,196	0	0
2	0,196	0,20266667	0,196	0,196	0	0
1	0,196	0,20171429	0,196	0,196	0	0

2	0,20266667	0,196	0,196	0,196	0	0
3	0,204	0,196	0,196	0,196	0	0
4	0,2135	0,2135	0,21717647	0,21717647	0,75	0,75
3	0,196	0,204	0,196	0,196	0	0
2	0,196	0,20266667	0,196	0,196	0	0
1	0,196	0,20171429	0,196	0,196	0	0
2	0,20266667	0,196	0,196	0,196	0	0
3	0,204	0,196	0,196	0,196	0	0
4	0,2135	0,2135	0,21717647	0,21717647	0,75	0,75
3	0,196	0,204	0,196	0,196	0	0
2	0,196	0,20266667	0,196	0,196	0	0
1	0,196	0,20171429	0,196	0,196	0	0
2	0,20266667	0,196	0,196	0,196	0	0
3	0,204	0,196	0,196	0,196	0	0
4	0,2135	0,2135	0,21717647	0,21717647	0,75	0,75
5	0,216	0,21475	0,21488889	0,21488889	0,86363636	0,5625
4	0,21717647	0,21717647	0,2135	0,2135	0,76470588	0,76470588
3	0,196	0,204	0,196	0,196	0	0
2	0,196	0,20266667	0,196	0,196	0	0
1	0,196	0,20171429	0,196	0,196	0	0
2	0,20266667	0,196	0,196	0,196	0	0
3	0,204	0,196	0,196	0,196	0	0
4	0,2135	0,2135	0,21717647	0,21717647	0,75	0,75
5	0,216	0,21475	0,21488889	0,21488889	0,86363636	0,5625
6	0,216	0,21513043	0,21488889	0,21488889	0,86666667	0,65217391
5	0,21475	0,216	0,21488889	0,21488889	0,5625	0,86363636
4	0,21717647	0,21717647	0,2135	0,2135	0,76470588	0,76470588
3	0,196	0,204	0,196	0,196	0	0
2	0,196	0,20266667	0,196	0,196	0	0
1	0,196	0,20171429	0,196	0,196	0	0
2	0,20266667	0,196	0,196	0,196	0	0

lhd	rhd	fwd	bwd	lwd	rwd	fnd
0,76470588	0,76470588	0,4375	0,4375	0,17647059	0,17647059	0,00166667
0	0	1	0,8	0,5	0,5	0,00333333
0,76470588	0,76470588	0,4375	0,4375	0,17647059	0,17647059	0,005
0,88888889	0,88888889	0	0,1875	0,16666667	0,16666667	0,005
0,88888889	0,88888889	0	0,26086957	0,16666667	0,16666667	0,00666667
0,88888889	0,88888889	0	0,2	0,16666667	0,16666667	0,01
0,88888889	0,88888889	0,26086957	0	0,16666667	0,16666667	0,00924242
0,88888889	0,88888889	0,1875	0	0,16666667	0,16666667	0,01
0,75	0,75	0,17647059	0,17647059	0,4375	0,4375	0,01333333
0,88888889	0,88888889	0	0,1875	0,16666667	0,16666667	0,015
0,88888889	0,88888889	0	0,26086957	0,16666667	0,16666667	0,01666667
0,88888889	0,88888889	0,1875	0	0,16666667	0,16666667	0,01566667
0,75	0,75	0,17647059	0,17647059	0,4375	0,4375	0,02277778
0	0	1	0,8	0,5	0,5	0,01
0	0	1	0,83333333	0,5	0,5	0,01166667
0	0	1	0,85714286	0,5	0,5	0,015
0	0	0,83333333	1	0,5	0,5	0,02348485
0	0	0,8	1	0,5	0,5	0,02333333
0,76470588	0,76470588	0,4375	0,4375	0,17647059	0,17647059	0,02277778
0	0	1	0,8	0,5	0,5	0,0225
0	0	1	0,83333333	0,5	0,5	0,02333333
0	0	1	0,85714286	0,5	0,5	0,02666667
0	0	0,83333333	1	0,5	0,5	0,02939394
0	0	0,8	1	0,5	0,5	0,0285
0,76470588	0,76470588	0,4375	0,4375	0,17647059	0,17647059	0,03277778
0	0	1	0,8	0,5	0,5	0,03
0	0	1	0,83333333	0,5	0,5	0,03166667
0	0	1	0,85714286	0,5	0,5	0,035
0	0	0,83333333	1	0,5	0,5	0,03621212
0	0	0,8	1	0,5	0,5	0,0355
0,76470588	0,76470588	0,4375	0,4375	0,17647059	0,17647059	0,04888889
0	0	1	0,8	0,5	0,5	0,0425
0	0	1	0,83333333	0,5	0,5	0,04333333
0	0	1	0,85714286	0,5	0,5	0,04666667
0	0	0,83333333	1	0,5	0,5	0,04212121
0	0	0,8	1	0,5	0,5	0,04066667
0,76470588	0,76470588	0,4375	0,4375	0,17647059	0,17647059	0,05888889

0	0	1	0,8	0,5	0,5	0,055
0	0	1	0,83333333	0,5	0,5	0,055
0	0	1	0,85714286	0,5	0,5	0,05833333
0	0	0,83333333	1	0,5	0,5	0,0480303
0	0	0,8	1	0,5	0,5	0,04566667
0,76470588	0,76470588	0,4375	0,4375	0,17647059	0,17647059	0,06833333
0	0	1	0,8	0,5	0,5	0,0675
0	0	1	0,83333333	0,5	0,5	0,06666667
0	0	1	0,85714286	0,5	0,5	0,07
0	0	0,83333333	1	0,5	0,5	0,05393939
0	0	0,8	1	0,5	0,5	0,05083333
0,76470588	0,76470588	0,4375	0,4375	0,17647059	0,17647059	0,07833333
0,88888889	0,88888889	0	0,1875	0,16666667	0,16666667	0,03416667
0,75	0,75	0,17647059	0,17647059	0,4375	0,4375	0,035
0	0	1	0,8	0,5	0,5	0,07666667
0	0	1	0,83333333	0,5	0,5	0,07666667
0	0	1	0,85714286	0,5	0,5	0,08
0	0	0,83333333	1	0,5	0,5	0,06409091
0	0	0,8	1	0,5	0,5	0,06116667
0,76470588	0,76470588	0,4375	0,4375	0,17647059	0,17647059	0,09666667
0,88888889	0,88888889	0	0,1875	0,16666667	0,16666667	0,03916667
0,88888889	0,88888889	0	0,26086957	0,16666667	0,16666667	0,04
0,88888889	0,88888889	0,1875	0	0,16666667	0,16666667	0,0825
0,75	0,75	0,17647059	0,17647059	0,4375	0,4375	0,04333333
0	0	1	0,8	0,5	0,5	0,09333333
0	0	1	0,83333333	0,5	0,5	0,09166667
0	0	1	0,85714286	0,5	0,5	0,095
0	0	0,83333333	1	0,5	0,5	0,07590909

bnd	lnd	rnd
0,00166667	0,00166667	0,00166667
0,0025	0,00166667	0,00166667
0,00555556	0,00333333	0,00333333
0,006	0,00555556	0,00611111
0,00712121	0,00611111	0,00666667
0,00833333	0,00666667	0,00722222
0,01166667	0,00777778	0,00722222
0,01416667	0,00833333	0,00777778
0,01611111	0,01	0,00944444
0,01366667	0,01055556	0,01
0,015	0,01111111	0,01055556
0,01916667	0,01111111	0,01166667
0,02166667	0,01277778	0,01333333
0,02066667	0,02277778	0,02166667
0,0219697	0,02277778	0,02166667
0,02208333	0,02277778	0,02166667
0,01666667	0,02166667	0,02277778
0,01916667	0,02166667	0,02277778
0,02611111	0,02333333	0,02444444
0,02583333	0,02444444	0,02333333
0,02787879	0,02444444	0,02333333
0,02861111	0,02444444	0,02333333
0,02833333	0,02333333	0,02444444
0,03166667	0,02333333	0,02444444
0,03888889	0,025	0,02611111
0,03283333	0,025	0,02611111
0,03469697	0,025	0,02611111
0,03541667	0,025	0,02611111
0,03666667	0,02611111	0,025
0,03916667	0,02611111	0,025
0,04555556	0,02777778	0,02666667
0,038	0,02666667	0,02777778
0,04060606	0,02666667	0,02777778
0,04194444	0,02666667	0,02777778
0,04833333	0,02777778	0,02666667
0,05166667	0,02777778	0,02666667
0,05833333	0,02944444	0,02833333

0,043	0,02944444	0,02833333
0,04651515	0,02944444	0,02833333
0,04847222	0,02944444	0,02833333
0,06	0,02833333	0,02944444
0,06416667	0,02833333	0,02944444
0,07166667	0,03	0,03111111
0,04816667	0,03111111	0,03
0,05242424	0,03111111	0,03
0,055	0,03111111	0,03
0,07166667	0,03	0,03111111
0,07666667	0,03	0,03111111
0,08444444	0,03166667	0,03277778
0,06916667	0,07888889	0,085
0,03666667	0,08666667	0,08055556
0,0585	0,035	0,03666667
0,06257576	0,035	0,03666667
0,065	0,035	0,03666667
0,08166667	0,03666667	0,035
0,08583333	0,03666667	0,035
0,09333333	0,03833333	0,03666667
0,0805	0,09388889	0,09722222
0,07818182	0,09444444	0,09777778
0,04333333	0,09833333	0,095
0,04666667	0,1	0,09666667
0,069	0,04666667	0,04333333
0,07439394	0,04666667	0,04333333
0,07736111	0,04666667	0,04333333
0,09666667	0,04333333	0,04666667