

**UNIVERSIDAD TECNOLÓGICA DE LOS ANDES
FACULTAD DE INGENIERÍA**

**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS
INFORMÁTICA**



**SISTEMA DE INTELIGENCIA ARTIFICIAL PARA EL CONTROL DE
ANDROIDES AUTÓNOMOS**

TESIS PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO DE SISTEMAS
E INFORMÁTICA

LÍNEA 07: INTELIGENCIA ARTIFICIAL

PRESENTADO POR:

BACH. SERGIO MONDRAGÓN CONTRERAS

ASESOR

Mg. MARLENY PERALTA ASCUE

ABANCAY-PERÚ
2018

**UNIVERSIDAD TECNOLÓGICA DE LOS ANDES
FACULTAD DE INGENIERÍA**

**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS E
INFORMÁTICA**



**SISTEMA DE INTELIGENCIA ARTIFICIAL PARA EL CONTROL DE
ANDROIDES AUTÓNOMOS**

LINEA 07: INTELIGENCIA ARTIFICIAL

Presentado por el bachiller SERGIO MONDRAGON CONTRERAS, a la Escuela Profesional de Ingeniería de Sistemas e Informática, sustentado y aprobado el 24 de agosto de 2018 para obtener el título profesional de:

INGENIERO DE SISTEMAS E INFORMÁTICA

Ante al jurado integrado por:

Presidente: _____
Mg. Edison Chiclla Carrasco

Primer Miembro: _____
Dr. Toribio Tapia Molina

Segundo Miembro: _____
Ing. Eduardo Chavez Vasquez

Asesor: _____
Mg. Marleny Peralta Ascue

DEDICATORIA:

Dedicado a mis padres y hermanos que siempre están, acompañando, reafirmando su fidelidad, confianza, aconsejando, en cada etapa de la vida.

Sergio Mondragón Contreras

AGRADECIMIENTOS:

Este contenido de investigación científica, escenifica el proceso de despliegue de someros y polifacéticas conjeturas de inteligencia artificial automática. Representa el ahincó y tolerancia de años de esfuerzo para alcanzar resultados que someramente gratifica las conjeturas. Reconozco el afán de cooperar para concluir esta etapa de investigación a mis padres y hermanos.

Agradecer a la Ing. Marleny Peralta Ascue, mi asesor, por su apoyo durante el proceso de la investigación y presentación de la tesis.

ÍNDICE DEL CONTENIDO

Resumen	8
Abstract	9
Introducción	10
Capítulo I.....	12
Planteamiento del problema.	12
1.1. Descripción del problema.....	12
1.1.2.1. Problema general	13
1.1.2.2. Problemas específicos	13
1.2. Objetivos de la investigación.....	13
1.2.1. Objetivo General.....	13
1.2.2. Objetivos específicos.	13
1.3. Justificación del problema.....	13
1.4. Delimitación.....	14
1.4.1 Delimitación Espacial	14
1.4.2 Delimitación Temporal.....	14
1.4.3. Delimitación Conceptual.....	14
Capítulo II	15
Marco Teórico.....	15
2.1. Antecedentes de la investigación.....	15
2.2. Bases teóricas	16
2.2.1. Mel-Frequency Cepstral Coefficients (MFCCs)	16
2.2.2. Backpropagation.....	17
2.2.3. Algoritmo LMS (del inglés, Least-Mean-Square algorithm)	19
2.2.4. Modelo de almacenamiento y extracción de datos.....	19
2.2.5. Sistema de emulación de Inteligencia Artificial	20
2.2.6. Biomecánica articular.....	21
2.2.6.1. Fundamentos de biomecánica articular humana.	22
2.2.7. Sistema de biomecánica articular androide.....	26
2.3. Marco conceptual.....	29
2.3.1. Inteligencia artificial.....	29
2.3.2. Robot.....	29
2.3.3. Robótica.	29

2.3.4. Backpropagation.....	30
2.4.1. Formulación de hipótesis.....	30
2.4.1.1. Hipótesis general.....	30
2.4.1.2. Hipótesis específicas.....	30
2.4.1.3. Operacionalización de variables.....	31
Capítulo III.....	32
Diseño Metodológico.....	32
3.1. Tipo y nivel de investigación.....	32
3.2. Diseño de la investigación.....	32
3.2.1.1. Java.....	33
3.2.1.2. C++.....	34
3.2.1.3. Pic18f4550.....	35
3.3. Población y muestra.....	35
3.4. Técnicas de recolección de datos.....	36
3.5. Técnicas de procesamiento de datos.....	36
Capítulo IV.....	37
Resultados de la Investigación.....	37
4.1. Captura he identificación de patrones de sonidos.....	37
4.2. Emulación del razonamiento y autonomía basada en hechos.....	42
4.2.1 Algoritmo puesta en producción de red neuronal backpropagation.....	47
4.2.2 Algoritmo puesta en producción de correlación de Pearson.....	49
4.3 Toma de decisión autónoma.....	50
4.3.1 Algoritmo de toma de decisiones AFD.....	51
Conclusiones.....	58
Recomendaciones.....	60
Bibliografía.....	61

ÍNDICE DE FIGURAS

Figura 1. Diagrama de bloques de extracción de MFCC	16
Figura 2. Modelo de red de tres capas.	18
Figura 3. Diagrama de sistema de emulación de inteligencia artificial.....	21
Figura 4. Movimiento de Flexión, Extensión.....	23
Figura 5. Movimiento de Abducción.....	23
Figura 6. Movimiento de Aducción	23
Figura 7. Movimiento de Rotación Interna, Rotación Externa.	24
Figura 8. Movimiento de Antepulsión-retropulsión.....	24
Figura 9. Movimiento de Flexión, Extensión.....	24
Figura 10. Movimiento de pronación y supinación.....	25
Figura 11. Movimiento de la cadera de Flexión	25
Figura 12. Movimiento de la cadera de Extensión, Abducción.....	25
Figura 13. Movimiento de piernas Aducción, Rotación Interna, Rotación Externa.	26
Figura 14. Placa madre de control de driver´s de motores CC	26
Figura 15. Sensores de control de giro de motor CC.....	27
Figura 16. Arreglo de driver`s para 8 Motores CC	27
Figura 17. Motor de CC	27
Figura 18. Esquema de la mecánica de movimientos del androide	28
Figura 19. Algoritmo de captura de sonido.....	37
Figura 20. Algoritmo MFCC	38
Figura 21. Patrón de sonido generado por el algoritmo MFCC.....	38
Figura 22. Algoritmo de inyección de patrones y ejecución de redes neuronales.	39
Figura 23. Diagrama de flujo del algoritmo de captura he identificación de patrones de sonidos.....	40
Figura 24. Aciertos, intentos fallidos en la captura de patrones de sonido.....	41
Figura 25. Jerarquía de directorios raíz de configuración de parámetros y de almacenamiento de redes neuronales y ficheros WAV.....	42
Figura 26. Network_humano fichero de parametrización de eventos.....	43
Figura 27. Event_* ficheros de parametrización de sistemas.	43
Figura 28. System_* ficheros de parametrización de redes neuronales.....	43

Figura 29. Ficheros de redes neuronales de parametrización de entradas, salidas, pesos, umbrales.	45
Figura 30. Raíz de directorios de ficheros WAV.....	46
Figura 31. Ficheros de audio en formato WAV	46
Figura 32. Algoritmo para reproducir ficheros de audio en formato WAV.	47
Figura 33. Grafo AFD de toma de decisiones.....	51
Figura 34. Algoritmo para mover las piernas del androide.....	53
Figura 35. Algoritmo del micro controlador recibiendo los parámetros de la PC.	53
Figura 36. Algoritmo del micro controlador para apagar/prender motores.	54
Figura 37. Algoritmo del micro controlador para lectura de sensores.	54
Figura 38. Diagrama del funcionamiento de emulación del razonamiento basada en conocimiento de hechos.	55
Figura 39. Acierto, intentos fallidos en la emulación de razonamiento y autonomía.	57
Figura 40. Androide.....	63
Figura 41. Interfaz de usuario para grabar archivos planos de parámetros de sonido.	64
Figura 42. Entrenador de redes neuronales.	65

ÍNDICE DE TABLAS

Tabla 1. Tabla de valores de Movimiento, Plano, Eje del cuerpo humano...	22
Tabla 2. Definición de Operación de Variable	31
Tabla 3. Resultados esperados y Herramientas a usarse.....	33
Tabla 4. Resultados de la pruebas de captura he identificación de patrones.	41
Tabla 5. Tabla de transiciones del AFD	51
Tabla 6. Resultados de la pruebas razonamiento y autonomía.....	56

RESUMEN

La inteligencia artificial es una sabiduría para elaborar sistemas autómatas con capacidades para emular y superar la inteligencia y talento humano mediante a un conjunto modelos lógicos, físicos, matemáticos intercomunicados. Las redes neuronales backpropagation son procesadores de cálculos matemáticos que pueden ser representados para emular el razonamiento humano. Una configuración de aproximación para absolver una incógnita, responde a la edificación de redes neuronales interconectadas con aforo para desarrollar singularidades humanas u otro tipo de inteligencias superiores. El modelo neuronal backpropagation, abstrae una aproximación de la función de organización y resolución de eventualidades como cerebro humano. Este modelo matemático es capaz de aprender a través de un aprendizaje supervisado de patrones de conocimiento. Se investigó un sistema que involucran tres disciplinas, ingeniería de sistemas, electrónica y mecánica que sintetizan en un androide semi-autónomo que emula el comportamiento humano como; razonar, hablar, escuchar, movimiento mecánico. Se desarrolló y probó tres sistemas integrados que interactúan sincronizados. Estos sistemas son; un sistema de control (software de control del computador y la placa madre del robot), sistema electrónico (electrónica de control de controladores y fuentes de energía de los motores CC), y un sistema mecánico (mecánica de movimientos).

Palabras claves: Inteligencia artificial, red neuronal, androide, backpropagation

ABSTRACT

Artificial intelligence is a wisdom to develop outdated systems with capabilities to emulate and overcome intelligence and human talent through a set of interconnected logical, physical, and mathematical models. Backpropagation neural networks are processors of mathematical calculations that can be represented to emulate human reasoning. An approximation configuration to answer an unknown, responds to the building of interconnected neural networks with capacity to develop human singularities or other superior intelligences. The neuronal model backpropagation, abstracts an approximation of the function of organization and resolution of eventualities as a human brain. This mathematical model is capable of learning through a supervised learning of knowledge patterns. We investigated a system involving three disciplines, systems engineering, electronics and mechanics that synthesize in a semi-autonomous droid that emulates human behavior as; reason, talk, listen, mechanical movement. Three integrated systems that interact synchronized were developed and tested. These systems are; a control system (computer control software and robot's motherboard), electronic system (control electronics for controllers and power sources of DC motors), and a mechanical system (movement mechanics).

Keywords: Artificial intelligence, neuronal network, android, backpropagation.

INTRODUCCIÓN

La Inteligencia artificial (IA), es la asociación de una pluralidad de investigación evolutiva, en las ciencias computacionales, en convergencia de áreas como; matemática, física, lógica, filosofía, biología, etc. que investigan nuevas soluciones capaces de absolver conjeturas y automatización de procesos repetitivos y complejidad lógica-matemática.

La inteligencia artificial cada día busca igualar y superar las capacidades humanas a través de sistemas lógicos-mecánicos-físico-mecánico, como una entidad android. Un androide es un autómatas antropomorfo que, procura asemejar una figura humanizada, procura emular características de hábito humano.

El aprendizaje automático y aprendizaje profundo evolucionan en una carrera de la perfección y tolerancia ante dilemas difusos, la corporaciones como; SpaceX, Tesla Motor, Google, Amazon, Facebook, Intel, Microsoft, Netflix, Spotify, Apple, etc. Invierten millones de dólares de su presupuesto para adquirir y embarcarse nuevas tecnologías basadas en inteligencia artificial. Las grandes corporaciones son conscientes que la IA, es una necesidad competitiva en los negocios, siendo el eslabón del éxito, todo su corte de negocio fluye a través de la IA. La investigación busca desplegar un sistema que interactúe con componentes mecánicos a través de consolas lógico-electrónicas, para emular singularidades humanas.

En la investigación se contempló la edificación de una figura física humana. Las extremidades son remolcadas por motores mecánicos.

La unidad principal de procesos es interpretado, por el sistema de redes neuronales backpropagation, delegado para recibir, procesar y enviar datos al controlador principal de motores.

La unidad de salida de audio del computador y audios grabados representan las voces generadas del androide.

La unidad de entrada de audio es delegado para capturar los sonidos para ser procesados por redes neuronales, representado el audio del androide.

La tesis está organizada en cuatro capítulos; primer capítulo denominado planteamiento del problema de la investigación, donde se describe el planteamiento y enunciado del problema, justificación, delimitaciones y objetivos de la investigación. En el segundo capítulo está referido el marco teórico, en el tercer capítulo se describe el diseño metodológico, se define el tipo y nivel de la investigación, población y muestra. En el cuarto capítulo denominado resultados, se describe el desarrollo del proyecto, así como la presentación de resultados. Finalmente conclusiones, recomendaciones, bibliografía y anexos.

CAPÍTULO I

PLANTEAMIENTO DEL PROBLEMA

1.1. Descripción del problema

La ausencia de tecnología con Inteligencia Artificial (IA), en las corporaciones nacionales, limita la competitividad respecto a corporaciones extranjeras, que exploran nuevas soluciones de negocio, optimizando ganancias y reduciendo pérdidas.

En Perú las corporaciones bancarias; Banco de Crédito BCP, BBVA Continental, Interbank, son las que lideran la demanda de tecnología con IA, para segmentar y explorar nuevos clientes (BCP, 2017)(Gestión, 2016).

Ricardo Márquez: “La industria que más crece en el mundo es la robótica”. Presidente de la Sociedad Nacional de Industrias indica que el Perú debe prepararse para el nuevo y cambiante escenario mundial, el mundo de hoy cada vez van apareciendo nuevos tipos de industrias y que el país debe adaptarse y prepararse para afrontar ese ritmo (Comercio, 2018).

La Sociedad Americana del Cáncer usa la IA para revisa mamografías 30 veces más rápido que el proceso tradicional y con un 99% de precisión. La Inteligencia Artificial ha logrado preestablecer funciones en robots que alivian al personal médico para realizar tareas (Robótica Inteligencia Artificial industria médica (Gestión, 2018).

1.1.2 Formulación del problema

1.1.2.1. Problema general

¿De qué manera el Sistema de Inteligencia Artificial emula el control de Androides autónomos?

1.1.2.2. Problemas específicos

- a) ¿De qué manera el Sistema de Inteligencia Artificial captura e identifica patrones de sonido?
- b) ¿De qué manera el Sistema de Inteligencia Artificial emula el razonamiento basado en conocimiento de hechos?

1.2. Objetivos de la investigación

1.2.1. Objetivo General

Emular mediante el Sistema de Inteligencia Artificial el control de Androides autónomos.

1.2.2. Objetivos específicos

- a) Capturar e identificar patrones de sonido mediante el Sistema de Inteligencia Artificial.
- b) Emular el razonamiento basado en conocimiento de hechos mediante el Sistema de Inteligencia Artificial.

1.3. Justificación del problema

Esta tesis es motivada porque en Perú, aun no contamos con empresas dedicadas a la investigación y comercialización de sistemas con Inteligencia Artificial (IA), que cubran la demanda de las grandes corporaciones bancarias, siendo una oportunidad de

negocio. Pero la IA desde la perspectiva nacional es otra, porque Perú es un país agrícola, minera y pesquera, no tenemos tecnología que aporte un valor agregado a las materias primas, siendo una oportunidad de negocio y generación de trabajo.

La investigación surge por la necesidad de contar con sistemas con IA que se ejecuten en diferentes sectores de producción; manufactura, agricultura, industria, pesca, minería, salud, banca y den un valor agregado maximizando las ganancias, minimizando costos.

La IA ofrece una gama de automatización de actividades, mejorando la velocidad, calidad de producción, de materias primas con valor agregado, durante las 24 horas continuas durante los 365 días al año. La implementación de la IA diferentes sectores, desencadenará un crecimiento económico saludable sostenido, aumentando el producto bruto interno (PBI).

1.4. Delimitaciones

1.4.1 Delimitación Espacial

La investigación se desarrolló en la ciudad de Lima en coordinación con la Universidad Tecnológica de los Andes - Abancay - Apurímac.

1.4.1 Delimitación Temporal

La investigación se desarrolló en siete años y medio aproximadamente, se inició en abril de 2011 hasta junio de 2018.

1.4.2 Delimitación Conceptual

La investigación se realiza en función a la teoría de la Inteligencia Artificial, que abarca el reconocimiento de voz con *Mel Frequency Cepstral Coefficient* (MFCC) y redes neuronales *backpropagation* basada en conocimiento de hechos.

CAPÍTULO II

MARCO TEÓRICO

2.1. Antecedentes de la investigación.

- a) En 1962, H.A. Ernst publicó el desarrollo de una mano mecánica controlada por una computadora. La MH-1 era capaz de "sentir" bloques y apilarlos sin la ayuda de ningún operario. Por la misma fecha, Tomovic y Boni diseñaron otra mano capaz de detectar el tamaño y peso de un objeto. Como diseñar manos parecía poco, al año siguiente se lanzó al mercado el primer prototipo comercial de brazo articulado. Lo creó la American Machina y Foundry Company, y se llamó **VESATRAN**.
- b) Algunos años después, en 1968, el Laboratorio de Inteligencia Artificial de Standford decide poner a una computadora ojos y oídos (cámaras y micrófonos) y añadir algo de inteligencia aunque fuese artificial. Los ingenieros de robots ven en este avance la posibilidad de sacar a sus máquinas de la oscuridad y el silencio y no se lo piensan dos veces.
- c) La compañía Kawasaki Heavy Industries negocia una licencia con Unimation y en 1970 desarrollan el primero robot articulado provisto de una cámara y controlado por una computadora. Lo llamaron el **brazo Stanford** y era capaz de apilar bloques según una estrategia trazada y de forma inteligente. El éxito obtenido por este proyecto llevó a considerar el control por la computadora y la sensorización como partes fundamentales de un sistema robotizado, y cualquier robot que se preciase, debía incluir ambos.
- d) A partir de la segunda mitad de los setenta todo evolucionó mucho más deprime. Ingenieros de la IBM desarrollaron en 1975 un manipulador inteligente con sensores de contacto y fuerza. En el MIT (Instituto Tecnológico de Massachussets) se

trabajaba en aspectos de inteligencia artificial. Y el Jet Propulsión Laboratory desarrollaba técnicas de control sobre el brazo Stanford para proyectos espaciales.

2.2. Bases teóricas

2.2.1. Mel-Frequency Cepstral Coefficients (MFCCs).

Los Coeficientes Cepstrales Mel-Frequency (MFCC) son las características dominantes ampliamente utilizadas para aplicaciones de procesamiento de voz como identificación, clasificación y reconocimiento. Mel-Frequency Cepstral (MFC) es el espectro de potencia a corto plazo de un sonido basado en la transformada de coseno lineal del espectro de potencia logarítmica en la frecuencia Mel-Scale. Los Coeficientes Cepstrales Mel-Frequency (MFCC) forman colectivamente un MFC y se derivan del representación cepstral de los archivos de sonido. La diferencia entre el cepstrum y Mel-Frequency Cepstrum es que la respuesta del sistema auditivo humano es más suave que la frecuencia linealmente especificada bandas utilizadas en el cepstrum normal. El diagrama de bloques que forma la extracción de MFCC se muestra en la figura 1 a continuación (Ajith, Mauri, Buford, Suzuki y Thampi, 2011, p.498).



Figura 1. Diagrama de bloques de extracción de MFCC.

Los MFCC en los marcos de corta duración se extraen como se describe a continuación.

1. 20 marcos cortos de tiempo son extraídos con un 50% de superposición para cada una de las palabras (archivos .wav). Como resultado de la superposición del número total de cuadros es 39.
2. Filtro de preénfasis de paso alto aplicado con el denominador de coeficientes $a = 1$ y el numerador $b = [1, -0.97]$.
3. Multiplica por la ventana de Hamming en el marco de caché.
4. La Transformación Rápida de Fourier (FFT) se aplica a cada cuadro con un tamaño de FFT de 512.
5. Las amplitudes del banco de filtros de escala de Mel se calculan con el número de filtros como 24.
6. Se calcula el espectro de inicio de sesión de cada filtro.
7. La transformación discreta de la transformación del coseno (DCT) se aplica al procedimiento Mel-filtrado coeficientes cepstrales.
8. Los Coeficientes Cepstrales Mel-Frequency (MFCC) se producen en 24 dimensiones en cada cuadro.

2.2.2. Backpropagation

Backpropagation es un modelo de red neuronal de aprendizaje supervisado, con retroalimentación, que desarrolla un ciclo de propagación y adaptación en dos fases. La primera vez inyectado un patrón de entrada, que estimula la red que se propaga desde la primera capa, hacia las capas más profundas de la red, hasta generar un resultado, este resultado se coteja con la salida deseada, en caso no satisfaga el resultado deseado, se desplegará hacia atrás una señal de error para cada una de las salidas de las capas internas de cada neurona de la red. Tener en cuenta que las capas profundas de la red

solo admiten una fracción de la señal total del error, basándose aproximadamente en la participación que haya aportado en el resultado de la salida original, este proceso se repite, capa por capa, hasta que todas las neuronas de la red hayan recibido una señal de error de participación en el error total. Teniendo la señal de error, se procede a actualizar los pesos de conexión de cada neurona, para hacer que la red converja hasta un estado que permita clasificar todos los patrones de entrada sin errores (Universidad Tecnológica de Pereira, 2000).

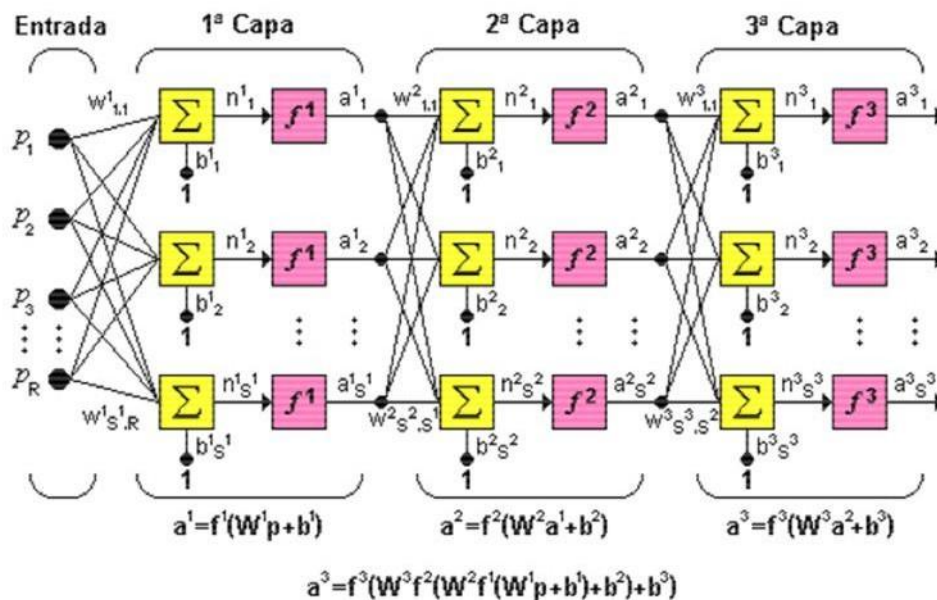


Figura 2. Modelo de red de tres capas.

W¹: Equivale a la matriz de pesos de la primera capa.

W²: Equivale a la matriz de pesos de la segunda capa.

W³: Equivale a la matriz de pesos de la tercera capa.

R:S¹:S²:S³ : Para definir la configuración de una red multicapa, se asigna una notación abreviada, donde el número de entradas seguido del número de neuronas en cada capa.

S¹: Es el equivalente de neuronas y el exponente equivale a la capa que corresponde.

2.2.3. Algoritmo LMS (del inglés, *Least-Mean-Square algorithm*).

Es un algoritmo de gradiente estocástico, tiene como función el reajuste de los pesos en base al error cuadrático. El modelo *backpropagation* depende de un set de entrada y salida esperado como se muestra a continuación: (Universidad Tecnológica de Pereira, 2000).

$$\{P1,T1\},\{P2,T2\},\dots\{PQ,TQ\}$$

Donde PQ es la entrada y TQ es la salida esperada para el patrón q-ésimo.

2.2.4. Modelo de almacenamiento y extracción de datos.

Hay muchas técnicas para el almacenamiento de datos, entre las más conocidas y comerciales son los gestores de base de datos, ya que nos ofrecen una arquitectura diseñada para soportar enormes cantidades de datos debidamente ordenadas y segmentados.

Como técnica de almacenamiento de datos, no se toma ningún gestor de base de datos, por las siguientes razones.

1. Son poco los gestores multiplataforma que se ejecuten en todos los sistemas operativos, no siendo un punto considerable, pero se tiene en cuenta este tipo característica.
2. Se tiene que tener un modelo y estructura de datos capaz de soportar el todas las eventualidades que pueda generar el sistema de inteligencia artificial variable, impredecible. Se busca una modelo de datos autoajustable.

La metodología propuesta es contar con una plataforma de almacenamiento auto gestionada, que esté dispuesta a soportar cualquier tipo de almacenamiento de datos en diferentes eventualidades que puedan surgir. El segundo punto es crítico para nuestro sistema de gestión de datos, puesto que el sistema de inteligencia artificial almacenara todo tipo de datos en tiempo real que ejecuten diferentes eventualidades.

Las redes neuronales son algoritmos de procesamiento de datos capaces de retener patrones de información, por ejemplo podemos guardar una oración, como respuesta de una pregunta, movimientos mecánicos en respuesta a preguntas o cambios de factores físicos, químicos capturados por sensores. Para el sistema de inteligencia artificial se utilizó el algoritmo de red neuronal *backpropagation* por ser un algoritmo que permite trabajar con números reales, esto permite desarrollar una enorme gama de combinaciones en un rango muy corto.

2.2.5. Sistema de Emulación de Inteligencia Artificial.

El sistema está compuesto por: 1) un coaching (entrenador de redes neuronales), que ayuda a entrenar las redes neuronales, y ponerla a punto para su uso; 2) reconocimiento de señales de sonido (MFCC) permite capturar sonidos y obtener un resumen de patrones de un sonido capturado; 3) motor de inferencia, se encarga de captura los patrones de sonido e identificar su significado y su respuesta como movimiento o sonido; 4) mapa de redes neuronales, es la agrupación de redes que se cargan en la memoria del computador; 5) diccionario de palabras (texto) en un contenedor de archivos tipo texto con un identificador que la redes neuronales obtienen como resultado de una ejecución; 6) Diccionario de palabras (sonidos) es un contenedor de archivos tipo audio con un identificador que la redes neuronales obtiene como resultado de una ejecución; 7) configuración y captura de palabras, es una panel de control donde se

vincula un texto con un sonido en particular, ya sea para patrón de reconocimiento o síntesis; 8) Síntesis de voz, permite unir varios sonidos de palabras y reproducirlo en una oración, 9) control hardware, permite enviar y recibir una matriz de patrones para controlar los motores del androide y recibir datos de los sensores del androide.

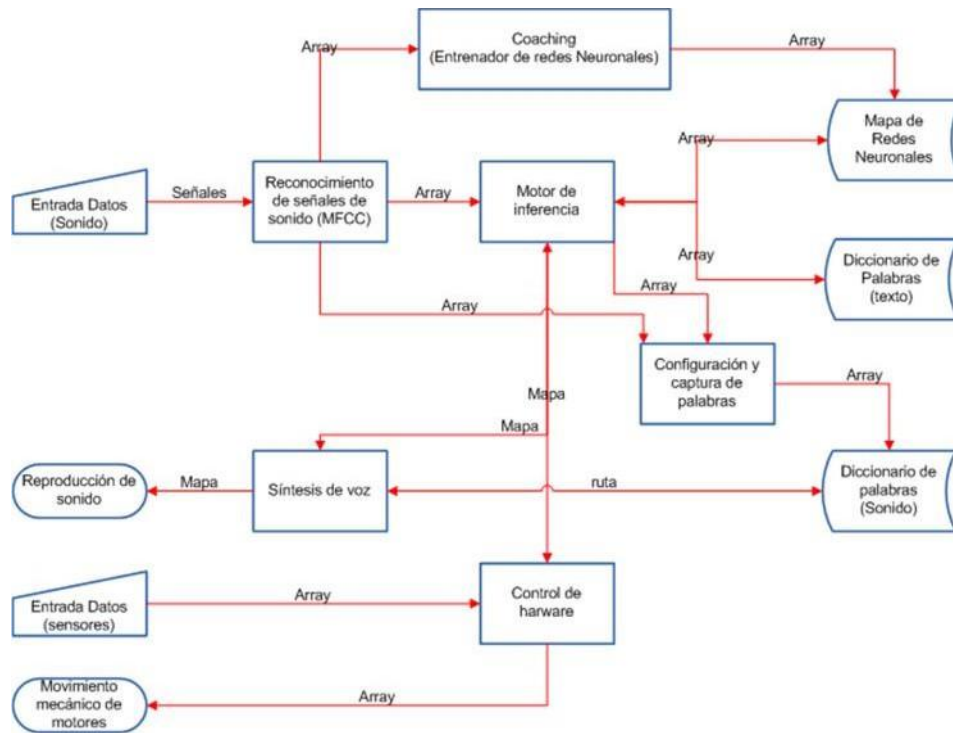


Figura 3. Diagrama de sistema de emulación de inteligencia artificial.

2.2.6. Biomecánica articular

Un androide la compone articulaciones mecánicas inspiradas en las articulaciones humanas que permite emular un movimiento humano. Las articulaciones mecánicas toman la funcionalidad articular inspirada en los fundamentos de la biomecánica articular humana. (Instituto Medico Leloir, p. 2-5).

2.2.6.1. Fundamentos de biomecánica articular humana.

La biomecánica articular es la disciplina práctica y funcional que estudia los movimientos realizados por las articulaciones. De la morfología de las superficies articulares (en el preparado) se deduce la función de cada articulación (en el vivo). (Instituto Medico Leloir, p. 2-5)

Al analizar la biomecánica de una articulación, hay que seguir una serie de pasos en su análisis:

1. Clasificación (tipo y género articular).
2. Movimiento que realizas.
3. Plano y eje de cada movimiento.
4. Posición de referencia.
5. Recorrido de grados.
6. Tiempos del recorrido (si corresponde)
7. Músculos que intervienen.

Para simplificar el análisis hay que tener en cuenta ciertas reglas:

— Todos los movimientos (Flexión-extensión, Abducción-Aducción, Rotación interna-externa) se realizan bajo un plano (Sagital, Frontal, Axial) de ejes (laterolateral, anteroposterior, cefalocaudal).

Tabla 1. Tabla de valores de Movimiento, Plano, Eje del cuerpo humano.

Movimiento	Plano	Eje
Flexión-extensión	Sagital	Laterolateral
Abducción- Aducción	Frontal	Anteroposterior
Rotación interna- externa	Axial	Cefalocaudal

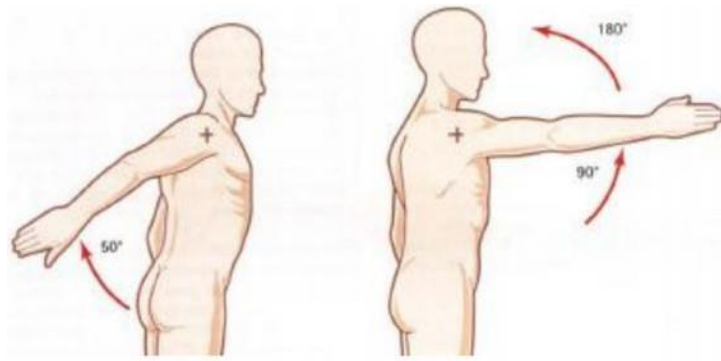


Figura 4. Movimiento de Flexión, Extensión.

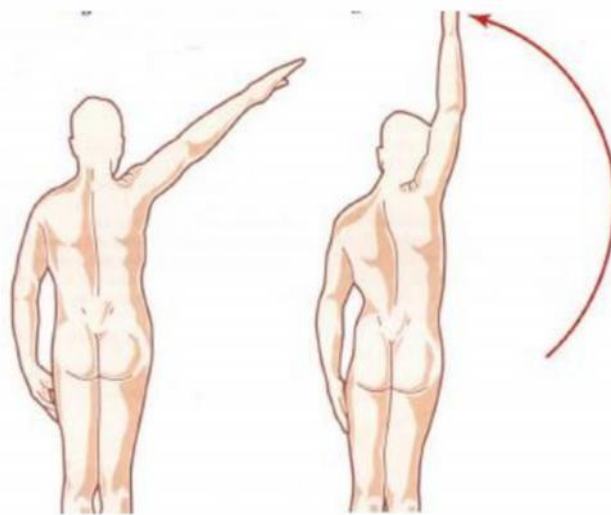


Figura 5. Movimiento de Abducción.

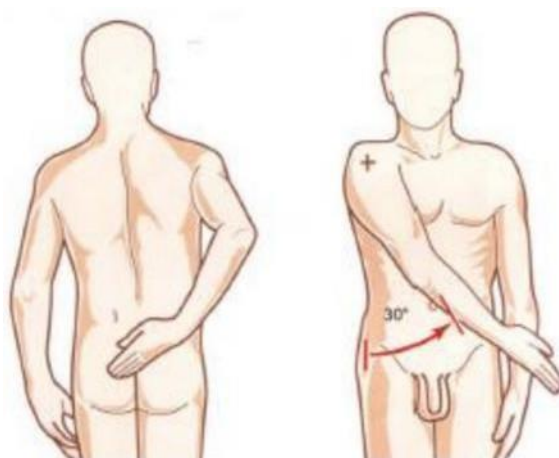


Figura 6. Movimiento de Aducción.

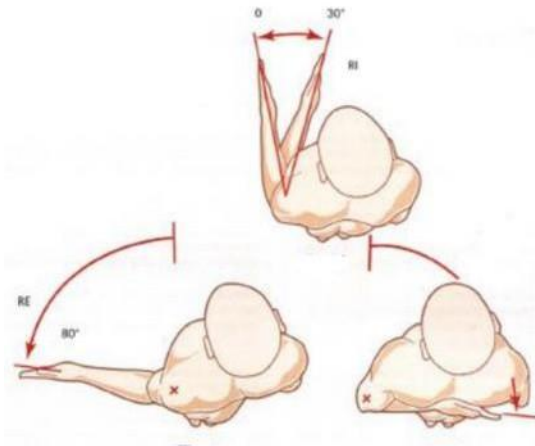


Figura 7. Movimiento de Rotación Interna, Rotación Externa.

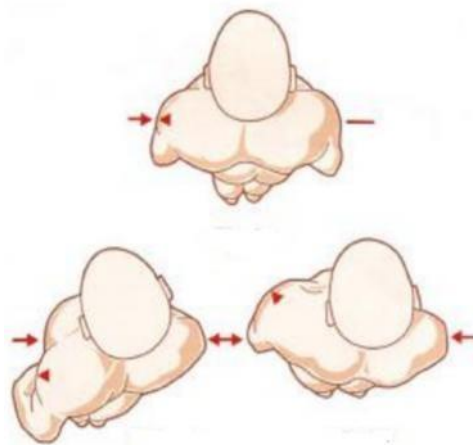


Figura 8. Movimiento de Antepulsión-retropulsión.

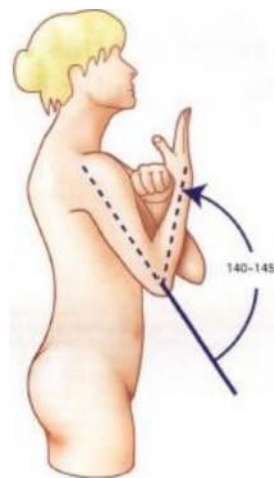


Figura 9. Movimiento de Flexión, Extensión.

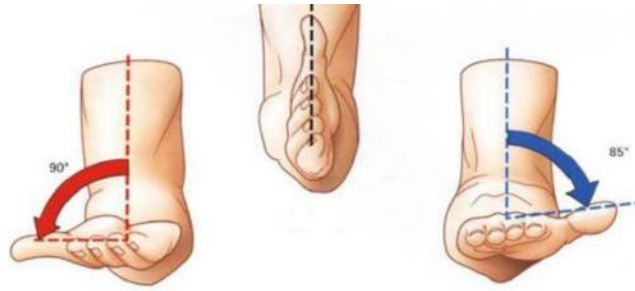


Figura 10. Movimiento de pronación y supinación.



Figura 11. Movimiento de la cadera de Flexión.

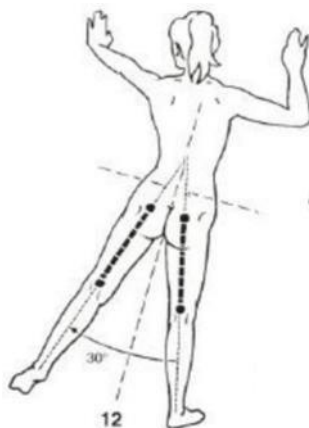


Figura 12. Movimiento de la cadera de Extensión, Abducción.

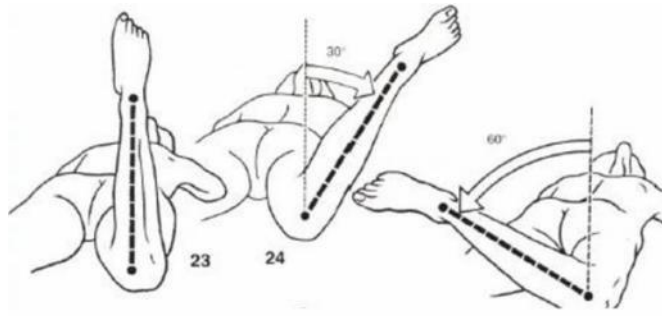


Figura 13. Movimiento de piernas Aducción, Rotación Interna, Rotación Externa.

2.2.7. Sistema de biomecánica articular androide.

La electrónica androide tiene como plataforma principal, una placa madre, que esta a su vez contiene zócalos, para conectar el arreglo de driver's de los motores CC. Esta placa madre está gobernada por un microcontrolador PIC 18f4550 que recibe órdenes del computador a través del puerto USB.

Los driver tienen la funcionalidad de encender, apagar, cambiar de polaridad a los motores CC, esa gama de funcionalidades proporcionan al androide la animación necesaria para emular el comportamiento mecánico de un ser humano.

La placa madre contiene una serie de zócalos para recibir datos de los diferentes sensores que se puedan a incluir.

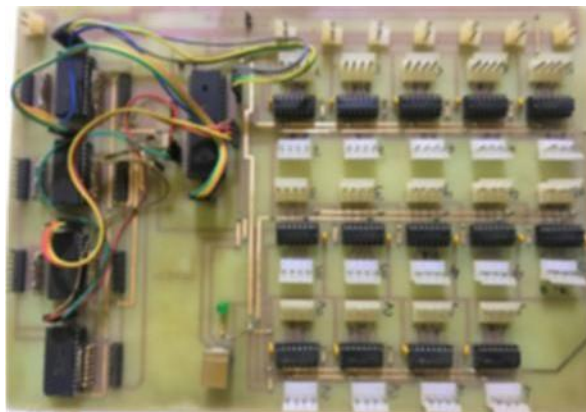


Figura 14. Placa madre de control de driver's de motores CC.

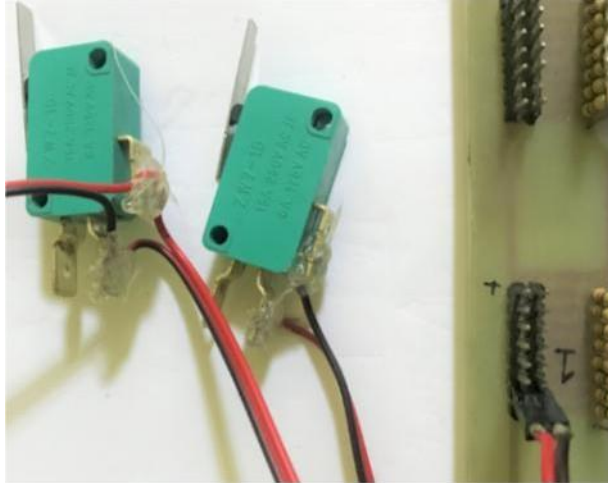


Figura 15. Sensores de control de giro de motor CC.

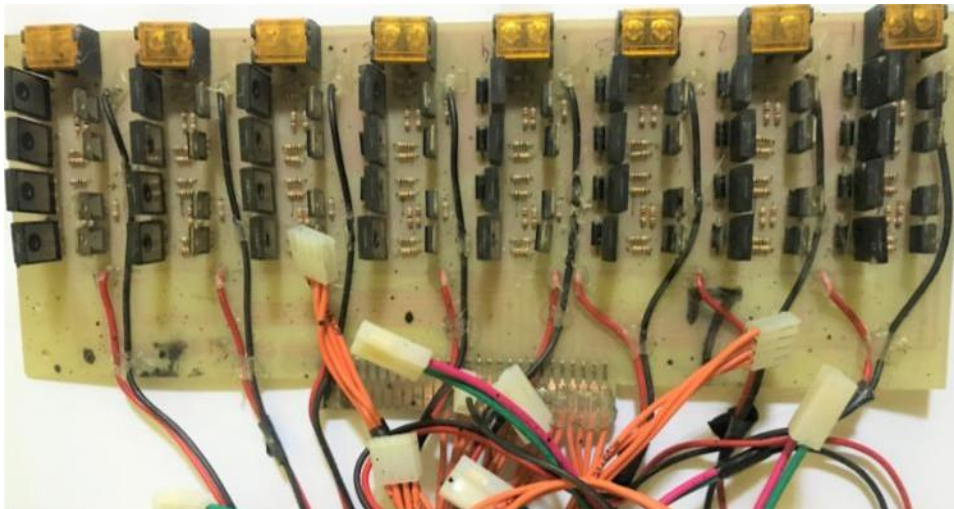


Figura 16. Arreglo de driver`s para 8 Motores CC



Figura 17. Motor de CC.

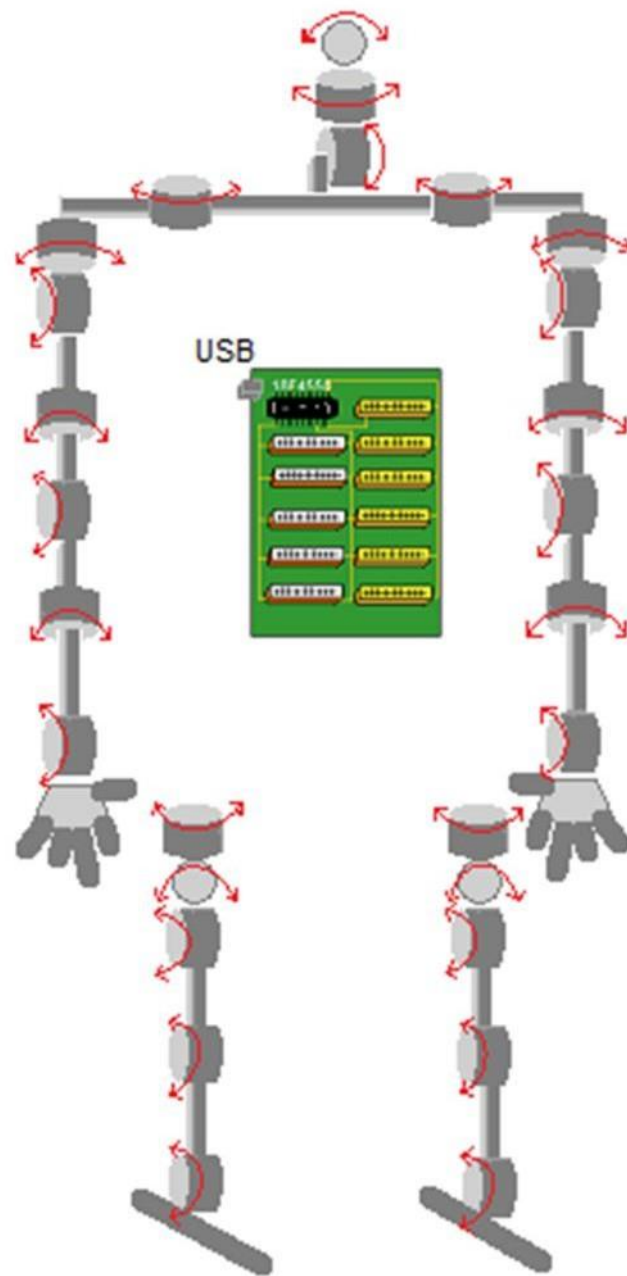


Figura 18. Esquema de la mecánica de movimientos del androide.

2.3. Marco conceptual.

2.3.1. Inteligencia artificial (IA).

Es la rama de la ciencia que se encarga del estudio de la inteligencia en elementos artificiales y, desde el punto de vista de la ingeniería, propone la creación de elementos que posean un comportamiento inteligente. La IA pretende construir sistemas y máquinas que presenten un comportamiento que si fuera llevado a cabo por una persona, se diría que es inteligente. Es muy interdisciplinar, ya que en ella intervienen disciplinas tan variadas como la Neurociencia, la Psicología, las Tecnologías de la Información, la Ciencia Cognitiva, la Física, las Matemáticas, etc. (Romero, Dafonte, Gómez y Penousal, 2007, p.10).

2.3.2. Robot

Una definición de robot usada por *The Robot Institute of America* es: “Un robot es un manipulador multifuncional y reprogramable diseñado para mover materiales, partes, herramientas, o dispositivos especiales, a través de movimientos variables programados para la ejecución de una variedad de tareas”.

2.3.3. Robótica

La robótica es la ciencia y la tecnología de los robots. Se ocupa del diseño, manufactura y aplicaciones de los robots. La robótica combina diversas disciplinas como son: la mecánica, la electrónica, la informática, la inteligencia artificial y la ingeniería de control. Otras áreas importantes en robótica son el álgebra, los autómatas programables y las máquinas de estados (conocimientosweb.net, 2014).

2.3.4. Backpropagation.

Existen varios modelos de algoritmos que permiten ir corrigiendo el error de predicción; uno de los más utilizados es el "*backpropagation*", que consiste básicamente en replegar los patrones de entrada entre las neuronas de entrada y capas profundas y propagar el error hacia atrás, desde la capa de salida hasta la de entrada, permitiendo así la adaptación de los pesos con el fin de reducir dicho error. (Hilera y Martínez, 2000).

2.4. Hipótesis

2.4.1. Formulación de hipótesis

2.4.1.1. Hipótesis general

Si se aplica el Sistema de Inteligencia Artificial entonces permitirá emular el control de Androides autónomos.

2.4.1.2. tesis específicas.

- a) Si se aplica el Sistema de Inteligencia Artificial entonces permitirá capturar e identificar patrones de sonido.
- b) Si se aplica el Sistema de Inteligencia Artificial entonces permitirá emular el razonamiento basado en conocimiento de hechos.

2.4.1.3. Operacionalización de variables

Tabla 2. Operacionalización de Variables

VARIABLE	DIMENSIÓN	INDICADOR	Unidad de Medida
INDEPENDIENTE Sistema de Inteligencia Artificial	<ul style="list-style-type: none"> - Algoritmos de reconocimiento de voz. - Algoritmo de Aprendizaje supervisado de entrenamiento de redes neuronales. 	<ul style="list-style-type: none"> - Algoritmo de parametrización (MFCC). - Red Neuronal Backpropagation. 	<ul style="list-style-type: none"> Número de coeficiente resultante Propagación de error
DEPENDIENTE Control de Androides Autónomos.	<ul style="list-style-type: none"> - Captura e identifica patrones de sonido - Emular el razonamiento basado en conocimiento de hechos. 	<ul style="list-style-type: none"> - Porcentaje de reconocimiento de patrones - Porcentaje de razonamiento de movimientos 	<ul style="list-style-type: none"> Número de intentos satisfactorio Número de intentos satisfactorios

CAPITULO III

DISEÑO METODOLÓGICO

3.1. Tipo y nivel de investigación

Tipo : Investigación Tecnológica

Nivel : Explicativo

Es investigación tecnológica porque busca descubrir nuevos conocimientos (ámbito científico) y transformar (ámbito tecnológico) en la mejora de procesos. La investigación tecnológica abarca con mayor énfasis la transformación, cuyo fin es obtener conocimiento para lograr modificar la realidad en estudio, persiguiendo un conocimiento práctico (García, 2007).

El nivel de la investigación es explicativo ya que se explica los efectos producidos del prototipo del Sistema de Inteligencia Artificial para emular el razonamiento humano.

3.2. Diseño y Método de la investigación.

El diseño de la investigación es Experimental, debido a que se recaban datos, para obtener información acerca del fenómeno estudiado (Androide autónomo) con información clara y suficiente, sin error o con un error para poder manejar o controlar. Reproduce el fenómeno en una situación controlada llamada experimento. (Gómez, 2012).

El método usado en este trabajo de investigación es el método deductivo-inductivo. Este método permite describir, detallar, especificar todos los aspectos del desarrollo del Sistema de Inteligencia Artificial para el control de Androides autónomos.

En esta sección se describe la forma de alcanzar la solución del problema planteado, se describe las herramientas, metodologías y métodos utilizados.

En la Tabla 3, se muestra un resumen del mapeo de herramientas y métodos usados en la investigación alineados a cada resultado esperado.

Tabla 3. Resultados esperados y Herramientas a usarse.

Resultados esperados	Herramientas a usarse
<p>Captura he identificación de patrones de sonidos: Desarrollar un modelo de algoritmos, de Coefi cientes Cepstrales en las Frecuencias de Mel (MFCC) para obtener patrones de sonido.</p>	<p>Java. C++. Mel Frecuent Cepstral Coeficients (MFCC).</p>
<p>Emulación del razonamiento basado en hechos: Desarrollar un motor de inferencia, basada en redes neuronales, que almacenan información como preposiciones.</p>	<p>Java. Red neuronal Backpropagation. Pic 18f4550.</p>

3.2.1. Herramientas

3.2.1.1. Java

Java es un lenguaje de programación completo, orientado a objetos, distribuido, interpretado, robusto, seguro, arquitectura neutra, portátil, de alto rendimiento, multiproceso, lenguaje dinámico (Gosling, 1995).

Java está diseñado para satisfacer los desafíos de desarrollo de aplicaciones en el contexto de entornos heterogéneos, en toda la red distribuida. Supremo Entre estos retos

es la entrega segura de aplicaciones que consumen mínimo de recursos del sistema, puede ejecutarse en cualquier hardware y software plataforma, y se puede ampliar de forma dinámica.

Java se originó como parte de un proyecto de investigación para desarrollar un software avanzado para una amplia variedad de dispositivos conectados en red y sistemas embebidos. El objetivo era desarrollar un portátil de pequeño tamaño, su distribución, funcionamiento, fiable y en tiempo real ambiente. Cuando se inició el proyecto, C ++ fue el idioma de su elección. Pero en el tiempo de las dificultades encontradas con C ++ crecieron hasta el punto en el problemas podrían solventarse mediante la creación de un lenguaje completamente nuevo ambiente. Diseño y arquitectura decisiones extrajeron de una variedad de lenguajes como Eiffel, Smalltalk, Objective C, y Cedar / Mesa. El resultado es un entorno de idioma que ha demostrado ser ideal para el desarrollo seguro, distribuidos, aplicaciones de usuario final basados en red en entornos que van desde dispositivos conectados a la Red embebido en el World-Wide Web y el escritorio (Sun Microsystems, 1995).

3.2.1.2. C++.

C++ es un propósito general lenguaje de programación con un sesgo hacia la programación de sistemas que soporta eficiente cómputo de bajo nivel, la abstracción de datos, programación orientada a objetos y la programación genérica.

El lenguaje de programación C ++ proporciona un modelo de la memoria y de cálculo que coincida exactamente con la de la mayoría de las computadoras. Además, proporciona mecanismos potentes y flexibles para la abstracción; es decir, el idioma construcciones que permiten al programador para introducir y utilizar nuevos tipos de objetos que coinciden con los conceptos de una aplicación. Por lo tanto, C ++ soporta estilos de programación que se basan en la manipulación bastante directa de los recursos

de hardware para ofrecer un alto grado de eficiencia más alto nivel estilos de programación que depender de los tipos definidos por el usuario para proporcionar un modelo de datos y de cálculo que está más cerca a la visión de un ser humano de la tarea a ejecutar por un ordenador. Estos estilos de más alto nivel de programación a menudo se llaman los datos abstracción, la programación orientada a objetos, y programación genérica (Stroustrup, 2013).

3.2.1.3. Pic18f4550.

Es un microcontrolador USB, tiene un conector USB para conectar e interactuar con la PC u otro host USB. Microchip proporcionar firmware libre para USB HID (dispositivo de interfaz humano), este dispositivo que se puede utilizar para hacer de mouse, joystick, CDC (dispositivo de clase de la comunicación), dispositivo de clase de audio (hacer su propio micrófono, altavoz, etc. dispositivos USB), es un dispositivo de almacenamiento masivo (Añadir sus propios discos USB), USB a RS232, etc., etc. Su código de gestor de arranque se puede programar y cambiar el firmware (Microchip, 2009).

3.3. Población y muestra.

La población está contenida por todas las capacidades humanas. El método de muestreo es no probabilístico a selección por criterio del investigador, se considera ciertas capacidades humanas para realizar las diferentes emulaciones, como es el reconocimiento de voz (8 palabras) y el razonamiento basado en hechos para emular los movimientos (cabeza, brazos y piernas).

3.4. Técnicas de recolección de datos.

La información del presente estudio se obtuvo a través de:

- a) Recopilación de material bibliográfico.
- b) Libros y apuntes bibliográficos a los cuales el autor tiene acceso.
- c) Evaluación de las herramientas software.

3.5. Técnicas de procesamiento de datos.

Para el análisis y procesamiento y presentación de resultados de las variables de la presente investigación, se efectuó una serie de análisis:

Análisis descriptivos a partir de los cuales se puede observar todo el proceso de desarrollo del Sistema de Inteligencia Artificial para el control de androides autónomos y describir algunas situaciones particulares que se encontraron en el proyecto.

Los datos obtenidos de las pruebas del reconocimiento de sonido y razonamiento basado en conocimiento de hechos, serán analizados y procesados debidamente obteniendo tablas, gráficos estadísticos, para luego interpretarlos.

CAPITULO IV

RESULTADOS DE LA INVESTIGACIÓN

4.1. Captura he identificación de patrones de sonidos

Para capturar el sonido se utilizó el micrófono de la computadora personal y para generar las ventanas de patrones de sonidos se desarrolló un algoritmo que captura 512 patrones de sonido, para luego ser procesados por el algoritmo MFCC.

```
try {
    DataLine.Info info = new DataLine.Info(TargetDataLine.class, format);
    line = (TargetDataLine) AudioSystem.getLine(info);
    formato=line.getFormat().isBigEndian();
    line.open(aF);
    line.start();

    byte buffer[] = new byte[512];
    while (line.read(buffer, 0, buffer.length)>0) {
        double[] fspeechFrame=convertirByteADouble(buffer);
        double maxAmpl = 0;
        double maxIndex = 0;
        for (int i = 0; i < fspeechFrame.length; i++) {
            if (Math.abs(fspeechFrame[i]) > maxAmpl) {
                maxAmpl = Math.abs(fspeechFrame[i]);
                maxIndex = i;
            }
        }
    }
}
```

Figura 19. Algoritmo de captura de sonido.

Luego de capturar los 512 patrones por el algoritmo anterior, se procede a analizar los patrones de sonidos utilizando el algoritmo de MFCC el cual genera un arreglo de 12 patrones para luego ser identificado por las redes neuronales.

```

int nnumberOfFilters = 24;
int nlifteringCoefficient = 22;
boolean oisLifteringEnabled = true;
boolean oisZeroThCepstralCoefficientCalculated = false;
int nnumberOfMFCCParameters = 12; //without considering 0-th
double dsamplingFrequency = 8000.0;
int nFFTLenght = 512;
if (oisZeroThCepstralCoefficientCalculated) {
    nnumberOfMFCCParameters = nnumberOfMFCCParameters + 1;
}

MFCC mfcc = new MFCC(nnumberOfMFCCParameters,
                    dsamplingFrequency,
                    nnumberOfFilters,
                    nFFTLenght,
                    oisLifteringEnabled,
                    nlifteringCoefficient,
                    oisZeroThCepstralCoefficientCalculated);

double[] dparameters = mfcc.getParameters(fspeechFrame);

```

Figura 20. Algoritmo MFCC.

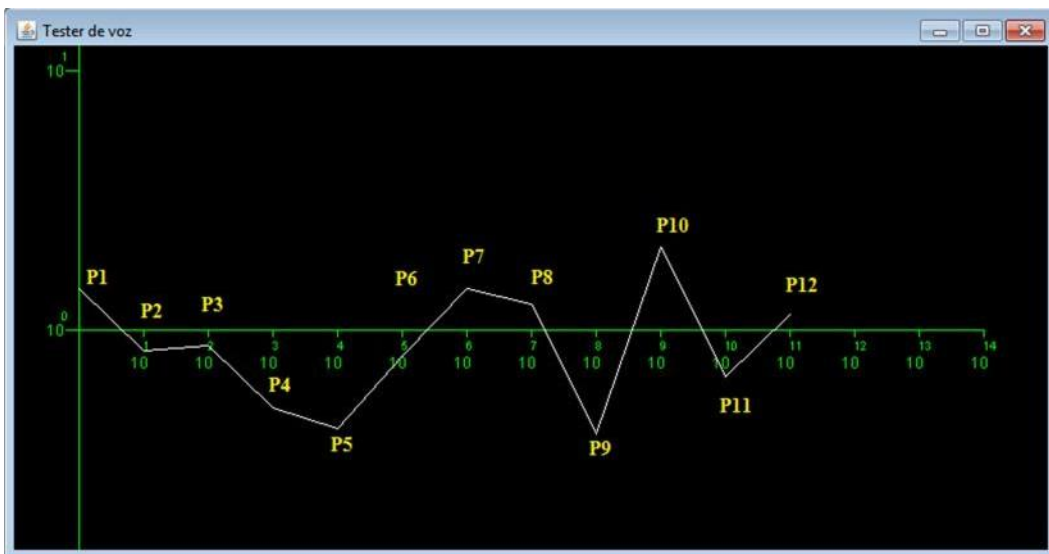


Figura 21. Patrón de sonido generado por el algoritmo MFCC.

Después de obtener los 12 patrones, se inyectan a variables globales, las cuales son consultadas por la redes diferentes redes neuronales, el código más abajo muestra la inyección de los 12 patrones en variables globales llamadas puertos receptores, líneas

más abajo se muestra la ejecución de todas la redes neuronales de identificación de patrones.

```
try{
    if(maxAmpl >= 7000){

        for (String key : mapNet.keySet()) {
            Net net = mapNet.get(key);
            net.getNetinput().setInputProvabel(dparameters);
            Core core = new Core(net);
            core.start();
            lenNet++;
        }

        for (String key : mapNet.keySet()) {
            Net net = mapNet.get(key);
            nombre = net.getKey();
            nombre = nombre.substring(0,nombre.indexOf("v"));
            if(resultadoRed.get(nombre)!=null){
                resultado = resultadoRed.get(nombre);
            }
            resultado+=resultado+Math.abs(net.getResultOutput());
            resultadoPromedio=resultadoPromedio+Math.abs(net.getResultOutput());
            resultadoRed.put(nombre, resultado);
            resultado=0;
        }
    }
}
```

Figura 22. Algoritmo de inyección de patrones y ejecución de redes neuronales.

En la Figura 23, se muestra un diagrama de flujo de interacción entre el algoritmo de parametrización de habla MFCC y los algoritmos de redes neuronales recibiendo los patrones de sonido, para la identificación del patrón de sonido. Existen varias docenas de redes neuronales que tiene como objetivo identificar un determinado patrón, las salidas de las redes neuronales deben mostrar un margen de error no superior al límite (0.1), si es que se cumple el caso se abra identificado satisfactoriamente el patrón de sonido.

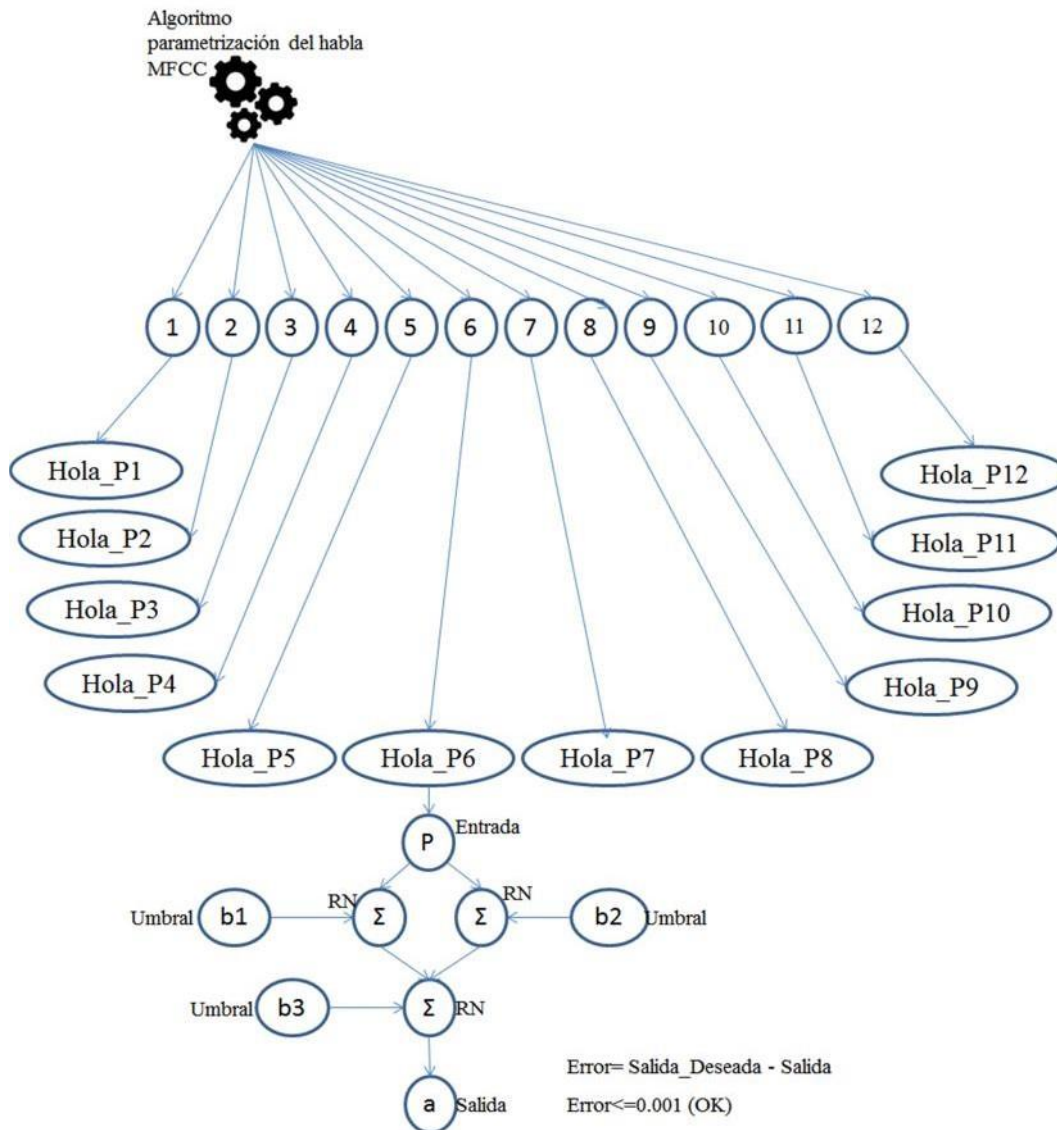


Figura 23. Diagrama de flujo del algoritmo de captura he identificación de patrones de sonidos.

Se realizó una serie de pruebas para verificar la reacción del prototipo computacional en la captura he identificación patrones de sonidos. Se tomaron para la prueba una lista de 8 palabras, de las cuales se realizan 10 intentos por cada palabra teniendo como resultados la siguiente tabla.

Tabla 4. Resultados de la pruebas de captura he identificación de patrones.

Palabras	Intentos satisfactorios	Intentos fallidos	Total Intentos
Hola	7	3	10
Como	8	2	10
Estas	6	4	10
Tú	8	2	10
Nombre	6	4	10
Avanza	7	3	10
Retrocede	7	3	10
Saluda	6	4	10
TOTAL	55	25	80

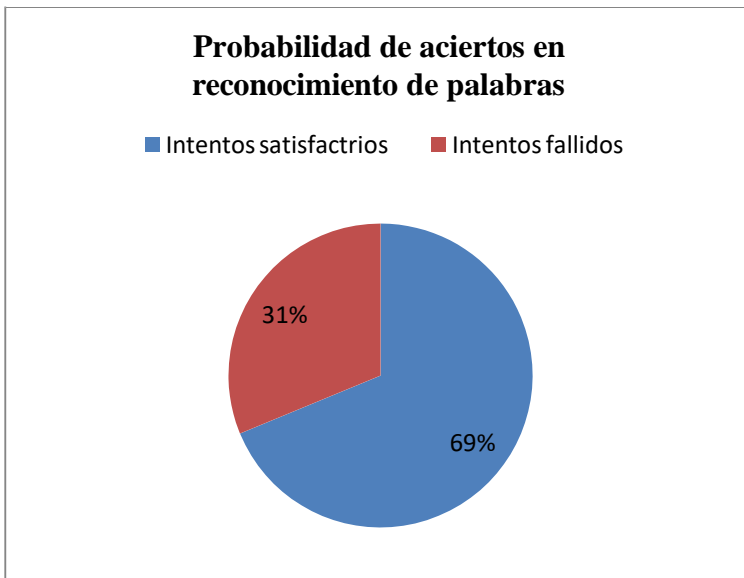


Figura 24. Aciertos, intentos fallidos en la captura de patrones de sonido.

En la figura 24, se muestra que del 100 % de pruebas se tiene que el 31% de las pruebas son fallidas, equivalentes a 25 pruebas y un 69 % de la pruebas son satisfactorias equivalentes a un 55 pruebas de 8 palabras donde:

El Error promedio: **3.125**.

La desviación estándar: **0.78062475**

El error estándar de la prueba: **0.275992527**

4.2. Emulación del razonamiento basado en conocimiento de hechos.

Para la emulación del razonamiento basado en conocimiento de hechos se cuenta con algoritmos que almacenen la información en una jerarquía que se pueda asemejar al sistema de organización de tareas de un ser humano, a continuación se mostrará cómo se organiza la información para su posterior uso por el sistema de emulación de inteligencia artificial.

En la figura 25 se muestra tres directorios. **Config** almacena los parámetros de configuración del sistema. **Db sound** almacena los ficheros WAV los sonidos grabados para su posterior concatenación y reproducción. Los ficheros WAV están organizados en orden alfabético. **Network** este directorio raíz que almacena los eventos, sistemas, y redes neuronales y sus parámetros.

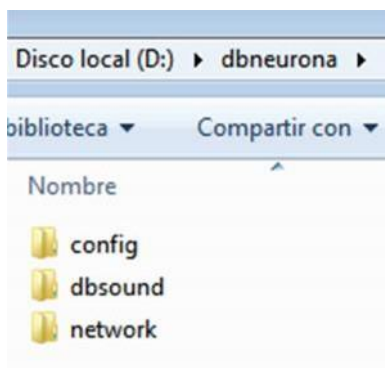


Figura 25. Jerarquía de directorios raíz de configuración de parámetros y de almacenamiento de redes neuronales y ficheros WAV.

En la figura 26 se muestra el fichero **network_humano** donde se parametrizan los eventos que se ejecutaran en androide.

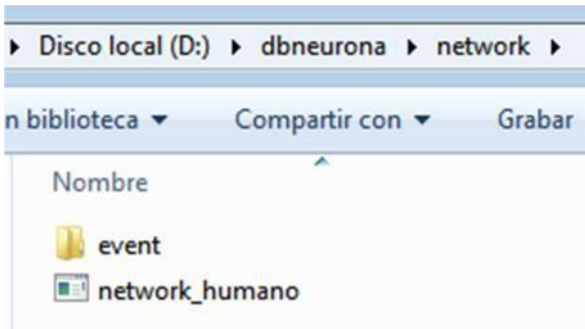


Figura 26. Network_humano fichero de parametrización de eventos.

En la figura 27 se muestra los ficheros **event_mover_brazo**, **event_mover_pierna**, **event_oir** donde se parametrizan los sistemas que se ejecutaran en android.

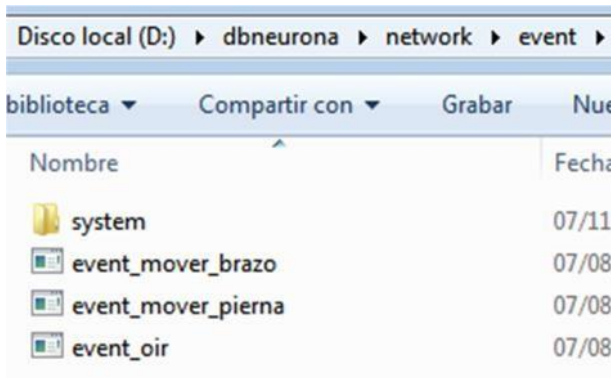


Figura 27. Event_* ficheros de parametrización de sistemas.

En la figura 28 se muestra los ficheros **system_brazo**, **system_oido**, **system_pierna** donde se parametrizan las redes neuronales que se ejecutaran en android.

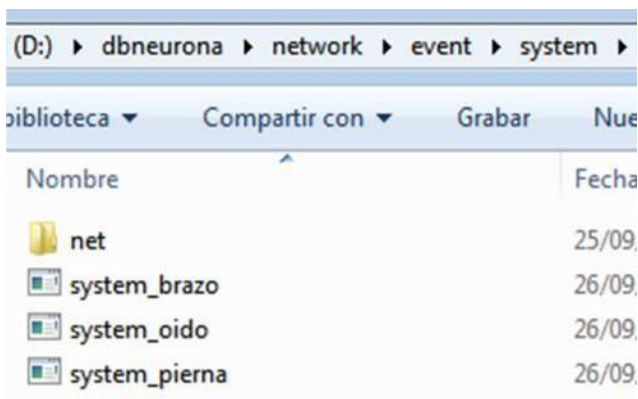


Figura 28. System_* ficheros de parametrización de redes neuronales.

En la figura 29 se muestra los directorios raíz de las entradas (input) de las redes neuronales para su entrenamiento (12 patrones procesados por el algoritmo MFCC). Directorio raíz de las salidas (output). Directorio raíz de los umbrales (threshold). Directorio raíz de los pesos (weight) que se generan después del entrenamiento de las redes neuronales. **hola02016-35-04-02-35-46-463v0** es un fichero ejemplo donde se parametriza la arquitectura ([1,2,1]), el puerto de entrada (1), puerto de salida (13), pesos (w_*v0), umbrales (t_*v0), entradas (i_*v0) y salida (o_*v0). La arquitectura ([1,2,1]) indica que tiene una entrada, dos neurona intermedias y una salida. Los puertos ([1:13]) indican los identificadores de la variables globales donde recolecta la información (puerto 1) y el puerto de depósito (13) donde almacena la información procesada.

Ejemplo de la parametrización interna de una red neuronal que identifica el primer punto (P0) del patrón de sonido **hola** “**hola02016-35-04-02-35-46-463v0.com**”.

```
hola02016-35-04-02-35-46-463v0:[1,2,1][1:13] w_hola02016-35-04-02-35-46-463v0 t_hola02016-35-04-02-35-46-463v0 i_hola02016-35-04-02-35-46-463v0 o_hola02016-35-04-02-35-46-463v0.
```

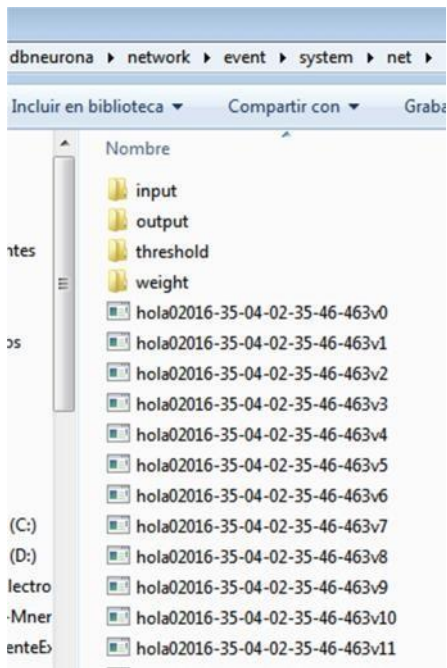


Figura 29. Ficheros de redes neuronales de parametrización de entradas, salidas, pesos, umbrales.

En la figura 30 se muestra los directorios raíz de los ficheros de audio que serán utilizados para reproducir el sonido para emular el habla del androide. En cada directorio clasificado alfabéticamente existen sonidos grabados para su uso.

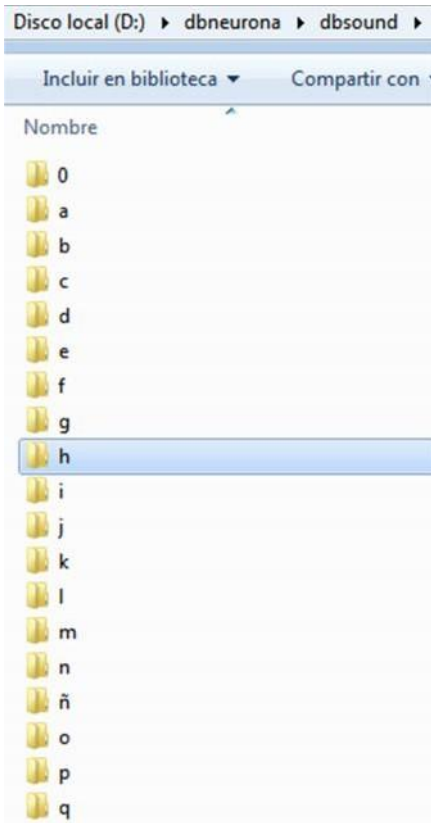


Figura 30. Raíz de directorios de ficheros WAV.

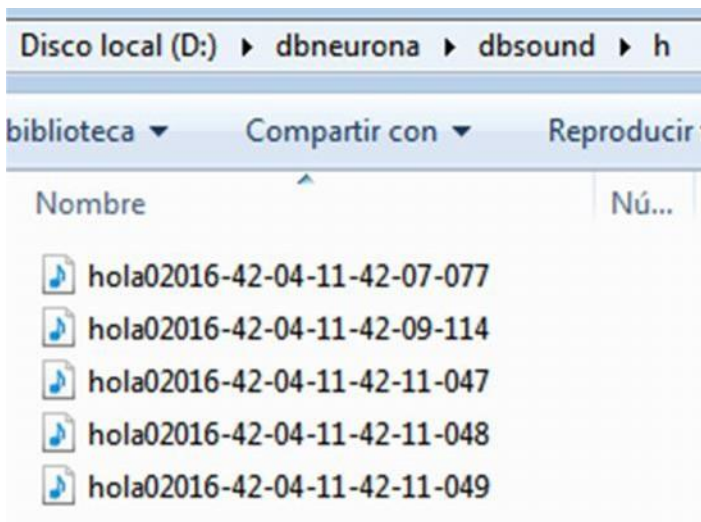


Figura 31. Ficheros de audio en formato WAV.

```

format = Formato.formatoStereo;
DataLine.Info info = new DataLine.Info(SourceDataLine.class, format);
bufferSize = (int) format.getSampleRate() * format.getFrameSize();

try {
    sline = (SourceDataLine) AudioSystem.getLine(info);
    sline.open(format, bufferSize);
} catch (LineUnavailableException ex) { }

ByteArrayOutputStream outWav = new ByteArrayOutputStream();

try {
    AudioInputStream audioInputStream = AudioSystem.getAudioInputStream(file);
    AudioInputStream playbackInputStream = AudioSystem.getAudioInputStream(audioI:
int frameSizeInBytes = audioInputStream.getFormat().getFrameSize();
int bufferLengthInFrames = sline.getBufferSize() / 8;
int bufferLengthInBytes = bufferLengthInFrames * frameSizeInBytes;
byte[] data = new byte[bufferLengthInBytes];
int count =0;
while ((count = playbackInputStream.read(data, 0, data.length))!=-1) {
    outWav.write(data, 0, count);
}
SoundPlay play = new SoundPlay(outWav, audioInputStream.getFormat());
play.start();
outWav.close();

public SoundPlay(ByteArrayOutputStream out, AudioFormat format){
    this.format = format;
    try{
        audio = out.toByteArray();
        InputStream input = new ByteArrayInputStream(audio);
        DataLine.Info info = new DataLine.Info(SourceDataLine.class, format);
        line = (SourceDataLine) AudioSystem.getLine(info);
        ais = new AudioInputStream(input, format, audio.length / format.getFrameSize());
        line.open(format);
        line.start();
        input.close();
        out.close();
    }catch(Exception e){
        System.err.println(e);
    }
}
}

```

Figura 32. Algoritmo para reproducir ficheros de audio en formato WAV.

4.2.1 Algoritmo puesta en producción de red neuronal backpropagation.

```

public class Core extends Thread {

static Logger log = Logger.getLogger(Core.class.getName());

private Net net=null;
private CommunicationObject comObjects=null;
private Integer presicione[][] = null;
private Integer arquitectura[] = null;
private double[][][] weight = null;
private double[][][] threshold = null;

```

```

public Core(Net net){
this.net=net;
comObjects=CommunicationObject.getInstance();
presicione = net.getNetoutput().getPresicione();
arquitecture = net.getNetarquitecture();
weight = net.getNetWeight();
threshold = net.getNetThreshold();
}

public void run(){
double a[][]=new double [arquitecture.length][1];

FunctionTransfer ft = new FunctionTransfer();
HashMap<Integer,Double> port=comObjects.getMapports();

int capa=0;

NetInput netInput = net.getNetinput();
a[capa] = netInput.getInputProvavel()[0];

NetOutput outPort = net.getNetoutput();
Integer outPorts[] = outPort.getOutputport();

do{
a[capa+1] =new double [weight[capa].length];
log.debug("====CAPA===="+(capa+1));
for(int f=0;f<weight[capa].length;f++){
for(int c=0;c<weight[capa][f].length;c++){
a[capa+1][f] = a[capa+1][f] + a[capa][c]*weight[capa][f][c];

log.debug("columna="+c+" fila="+f);
log.debug("entrada:"+a[capa][c]+ " * peso:"+weight[capa][f][c]+ " =
salida:"+a[capa+1][f]);
}

if(capa!=arquitecture.length-2){
//log.debug("Red="+net.getKey()+"; Previo="+a[capa+1][f]);
a[capa+1][f] = ft.geRound2Decim(ft.getTanSig(a[capa+1][f]+threshold[capa][f][0]));
log.debug("columna="+0+" fila="+f);
log.debug("entrada:"+a[capa+1][f]+ " * umbral:"+threshold[capa][f][0]+ " =
salida:"+a[capa+1][f]+threshold[capa][f][0]);
log.debug("aplica funcion de activacion TanSig:"+ a[capa+1][f]);
}else{
a[capa+1][f] = ft.geRound3Decim(a[capa+1][f]+threshold[capa][f][0]);
//log.debug("Red="+net.getKey()+"; Resultado="+a[capa+1][f]);
log.debug("columna="+0+" fila="+f);
log.debug("entrada:"+a[capa+1][f]+ " * umbral:"+threshold[capa][f][0]+ " =
salida:"+a[capa+1][f]);
net.setResultOutput(a[capa+1][f]);
}
}
}

```

```

}

capa++;
}while(capa<arquitecture.length-1);

comObjects.setNet(net.getKey(), net);

for(Integer ip:outPorts){
port.put(ip, net.getResultOutput());
}

}
}

```

4.2.2 Algoritmo puesta en producción de correlación de Pearson

```

public class CorelacionPearson {

public void correlacionPearson(String buqueda [],String label []){
String subMatrizX []=null;
String subMatrizY []=null;
double ordenar []=null;
String result [][]=null;
int itera=0;
String lblItera=null;
String fileName="";
String newFileName="";
String url="C:\\Unidad D\\dbneurona\\network\\event\\system\\net\\input\\";
String selectedPattern;

double x,y,zX,zY,xx,yy,xy,prom,sumX,sumY=0,mediaX,mediaY;

x=y=zX=zY=xx=yy=xy=prom=sumX=0;

selectedPattern = recuperarContenidoArchivo(url+patronRed+".com");

subMatrizX = selectedPattern.split("\n");

ordenar = new double [buqueda.length-1];

for(int z=1;z<buqueda.length;z++){

subMatrizY = buqueda[z].split("\n");
prom = subMatrizY.length;

for(int j=0;j<subMatrizX.length;j++){
y = Double.parseDouble(subMatrizY[j]);

```

```

x = Double.parseDouble(subMatrizX[j]);
sumX = sumX + x;
sumY = sumY + y;
log.debug("X:"+x+" Y:"+y);
}

mediaX = sumX/prom;
mediaY = sumY/prom;
log.debug("sumaX:"+sumX+" sumaY:"+sumY);
log.debug("promedioX:"+mediaX+" promedioY:"+mediaY);
for(int j=0;j<subMatrizX.length;j++){
y = Double.parseDouble(subMatrizY[j]);
x = Double.parseDouble(subMatrizX[j]);
zX = x - mediaX;
zY = y - mediaY;
xx = xx + zX * zX;
yy = yy + zY * zY;
xy = xy + zY * zX;
log.debug("x:"+x+" y:"+y+" x-promedioX(zX):"+zX+" y-promedioY(zY):"+zY+"
zX*zX:"+xx+" zY*zY:"+yy+" zY*zX="+xy);
}

ordenar[itera] = xy/Math.pow((xx*yy), 1.0/2.0);
itera++;
sumX = 0;
sumY = 0;
zX = 0;
zY = 0;
xx = 0;
yy = 0;
xy = 0;
}
result = getOrdenar(ordenar,label);

}

```

4.3 Toma de decisión autónoma.

Se desarrolla un algoritmo Autómata Finito Determinista (AFD) que analizar el mejor resultado del modelo de red neuronal, obtiene el título de la etiqueta, he identificar el estado del nodo del resultado. En la tabla 5 se muestra el estado de las transmisiones de los nodos.

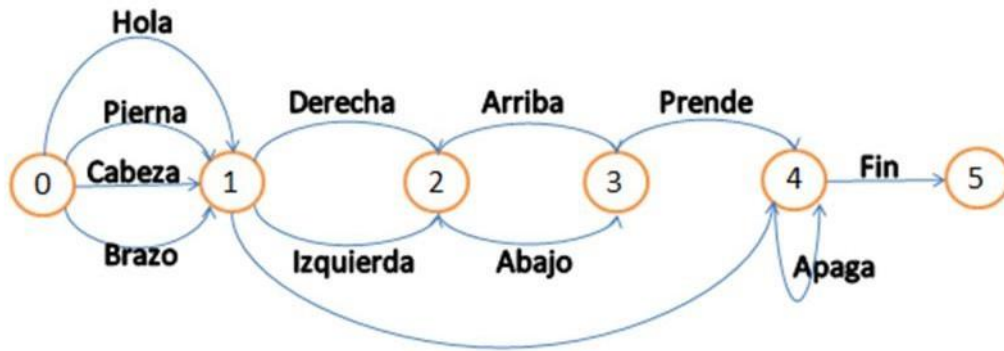


Figura 33. Grafo AFD de toma de decisiones.

Tabla 5. Tabla de transiciones del AFD de toma de decisiones

	Nodo 0	Nodo 1	Nodo 2	Nodo 3	Nodo 4	Nodo 5
hola	error	0	error	error	error	error
brazo	error	0	error	error	error	error
cabeza	error	0	error	error	error	error
pierna	error	0	error	error	error	error
izquierda	error	error	1	error	error	error
derecha	error	error	1	error	error	error
arriba	error	error	error	2	error	error
abajo	error	error	error	2	error	error
prende	error	error	error	error	3	error
apaga	error	error	error	error	4	error
fin	error	error	error	error	error	4

4.3.1 Algoritmo de toma de decisiones AFD.

```

public class Automata {
int cont;
String afd [];
boolean aceptado;
static Logger log = Logger.getLogger(Automata.class.getName());
private static final String BRAZO="brazo";
private static final String CABEZA="cabeza";
private static final String PIERNA="pierna";
private static final String IZQUIERDA="izquierda";
private static final String DERECHA="derecha";
private static final String ARRIBA="arriba";
private static final String ABAJO="abajo";
private static final String PRENDE="prende";
private static final String APAGA="apaga";

```

```

private static final String FIN="fin";

public Automata(String afd []){
cont =0; node0(); this.afd=afd;
}
public void node0(){
log.debug("Nodo 00");
aceptado=false;
if(cont<afd.length){
if(afd[cont].equals(BRAZO) || afd[cont].equals(CABEZA) ||
afd[cont].equals(PIERNA)){
cont=3;
node1();
}
}
}
public void node1(){
log.debug("Nodo 01");
if(cont<afd.length){
if(afd[cont].equals(IZQUIERDA) || afd[cont].equals(DERECHA)){
cont=5; node2();
}
}
}
public void node2(){
log.debug("Nodo 02");
if(cont<afd.length){
if(afd[cont].equals(ARRIBA) || afd[cont].equals(ABAJO)){
cont=7; node3();
}
}
}
public void node3(){
log.debug("Nodo 03 ");
if(cont<afd.length){
if(afd[cont].equals(PRENDE)){
cont=8; node4();
}
}
}
public void node4(){
log.debug("Nodo 04");
if(cont<afd.length){
if(afd[cont].equals(APAGA)){
cont=9; node5();
}
}
}
public void node5(){
log.debug("Nodo 05 ");

```

```

aceptado=true;
}
}

```

```

byte[] respuesta = iface.QWriteRead(out, out.length, 11, 1);
String response = new String(respuesta, "utf-8");
for(int h=1;h<response.length();h++){
    capturaSensorMusloIzq = Integer.parseInt(response.substring(1,2));
    capturaSensorMusloIzq = Integer.parseInt(response.substring(3,4));
    capturaSensorRodillaIzq = 1;
    if(incr==14 || (incr+1)==15){
        if(out[incr]==1 && capturaSensorMusloIzq==0 && actualizaMusloIzq!=capturaSensorMusloIzq){out[incr]
        } else if(out[incr+1]==1 && capturaSensorMusloIzq==0 && actualizaMusloIzq!=capturaSensorMusloIzq){out
        }
    }
    if(incr==12 || (incr+1)==13){
        if(out[incr]==1 && capturaSensorMusloDer==0 && actualizaMusloDer!=capturaSensorMusloDer){out[incr]
        } else if(out[incr+1]==1 && capturaSensorMusloDer==0 && actualizaMusloDer!=capturaSensorMusloDer){out
        }
    }

    if(capturaSensorMusloIzq==1){
        actualizaMusloIzq=1;
    }

    if(capturaSensorMusloDer==1){
        actualizaMusloDer=1;
    }
}

```

Figura 34. Algoritmo para mover las piernas del androide.

```

    setup_port_A(ANO_TO_AN3_ANALOG);
    setup_adc(adc_clock_internal);
    APAGADO();
    usb_init();
    usb_task();
    usb_wait_for_enumeration();
    enable_interrupts(global);
    while (TRUE)
    {
        ;

        if(usb_enumerated()){
            if (usb_kbhit(1)){
                usb_get_packet(1, recbuf, Lenbuf);

                sendmsn[1]='0';
                sendmsn[2]='0';
                sendmsn[3]='0';
                sendmsn[4]='0';
                sendmsn[5]='0';
                sendmsn[6]='0';
                sendmsn[7]='0';
                sendmsn[8]='0';

                if(recbuf[0]==1){
                    param=0;
                    grupo=1;
                    sendmsn[0]='1';
                    PRENDE();
                }else if (recbuf[0]==2){
                    param=16;
                }
            }
        }
    }

```

Figura 35. Algoritmo del micro controlador recibiendo los parámetros de la PC.

```

VOID PRENDE() {
    acum=1;
    pin=0;

    while(acum<=112) {

        if(recbuf[1]==1 && acum==(param+1)) {
            output_high(pin_b1);
            output_high(pin_b0);
            output_low(pin_b0);
        } else if(recbuf[2]==1 && acum==(param+2)) {
            output_high(pin_b1);
            output_high(pin_b0);
            output_low(pin_b0);
        } else if(recbuf[3]==1 && acum==(param+3)) {
            output_high(pin_b1);
            output_high(pin_b0);
            output_low(pin_b0);
        } else if(recbuf[4]==1 && acum==(param+4)) {
            output_high(pin_b1);
            output_high(pin_b0);
            output_low(pin_b0);
        } else if(recbuf[5]==1 && acum==(param+5)) {
            output_high(pin_b1);
            output_high(pin_b0);
            output_low(pin_b0);
        } else if(recbuf[6]==1 && acum==(param+6)) {
            output_high(pin_b1);
            output_high(pin_b0);
        }
    }
}

```

Figura 36. Algoritmo del micro controlador para apagar/prender motores. .

```

float lectura_sensor(int grupot,int pint) {

    if(grupot==1) {
        if(pint==0) {PORTB7=0;PORTB6=0;PORTB5=0;PORTB4=0;}
        else if(pint==1) {PORTB7=1;PORTB6=0;PORTB5=0;PORTB4=0;}
        else if(pint==2) {PORTB7=0;PORTB6=1;PORTB5=0;PORTB4=0;}
        else if(pint==3) {PORTB7=1;PORTB6=1;PORTB5=0;PORTB4=0;}
        else if(pint==4) {PORTB7=0;PORTB6=0;PORTB5=1;PORTB4=0;}
        else if(pint==5) {PORTB7=1;PORTB6=0;PORTB5=1;PORTB4=0;}
        else if(pint==6) {PORTB7=0;PORTB6=1;PORTB5=1;PORTB4=0;}
        else if(pint==7) {PORTB7=1;PORTB6=1;PORTB5=1;PORTB4=0;}
        else if(pint==8) {PORTB7=0;PORTB6=0;PORTB5=0;PORTB4=1;}
        else if(pint==9) {PORTB7=1;PORTB6=0;PORTB5=0;PORTB4=1;}
        else if(pint==10) {PORTB7=0;PORTB6=1;PORTB5=0;PORTB4=1;}
        else if(pint==11) {PORTB7=1;PORTB6=1;PORTB5=0;PORTB4=1;}
        else if(pint==12) {PORTB7=0;PORTB6=0;PORTB5=1;PORTB4=1;}
        else if(pint==13) {PORTB7=1;PORTB6=0;PORTB5=1;PORTB4=1;}
        else if(pint==14) {PORTB7=0;PORTB6=1;PORTB5=1;PORTB4=1;}
        else if(pint==15) {PORTB7=1;PORTB6=1;PORTB5=1;PORTB4=1;}
    } else if(grupot==2) {
        if(pint==0) {PORTD3=0;PORTD2=0;PORTD1=0;PORTD0=0;}
        else if(pint==1) {PORTD3=1;PORTD2=0;PORTD1=0;PORTD0=0;}
        else if(pint==2) {PORTD3=0;PORTD2=1;PORTD1=0;PORTD0=0;}
        else if(pint==3) {PORTD3=1;PORTD2=1;PORTD1=0;PORTD0=0;}
        else if(pint==4) {PORTD3=0;PORTD2=0;PORTD1=1;PORTD0=0;}
        else if(pint==5) {PORTD3=1;PORTD2=0;PORTD1=1;PORTD0=0;}
        else if(pint==6) {PORTD3=0;PORTD2=1;PORTD1=1;PORTD0=0;}
        else if(pint==7) {PORTD3=1;PORTD2=1;PORTD1=1;PORTD0=0;}
        else if(pint==8) {PORTD3=0;PORTD2=0;PORTD1=0;PORTD0=1;}
    }
}

```

Figura 37. Algoritmo del micro controlador para lectura de sensores. .

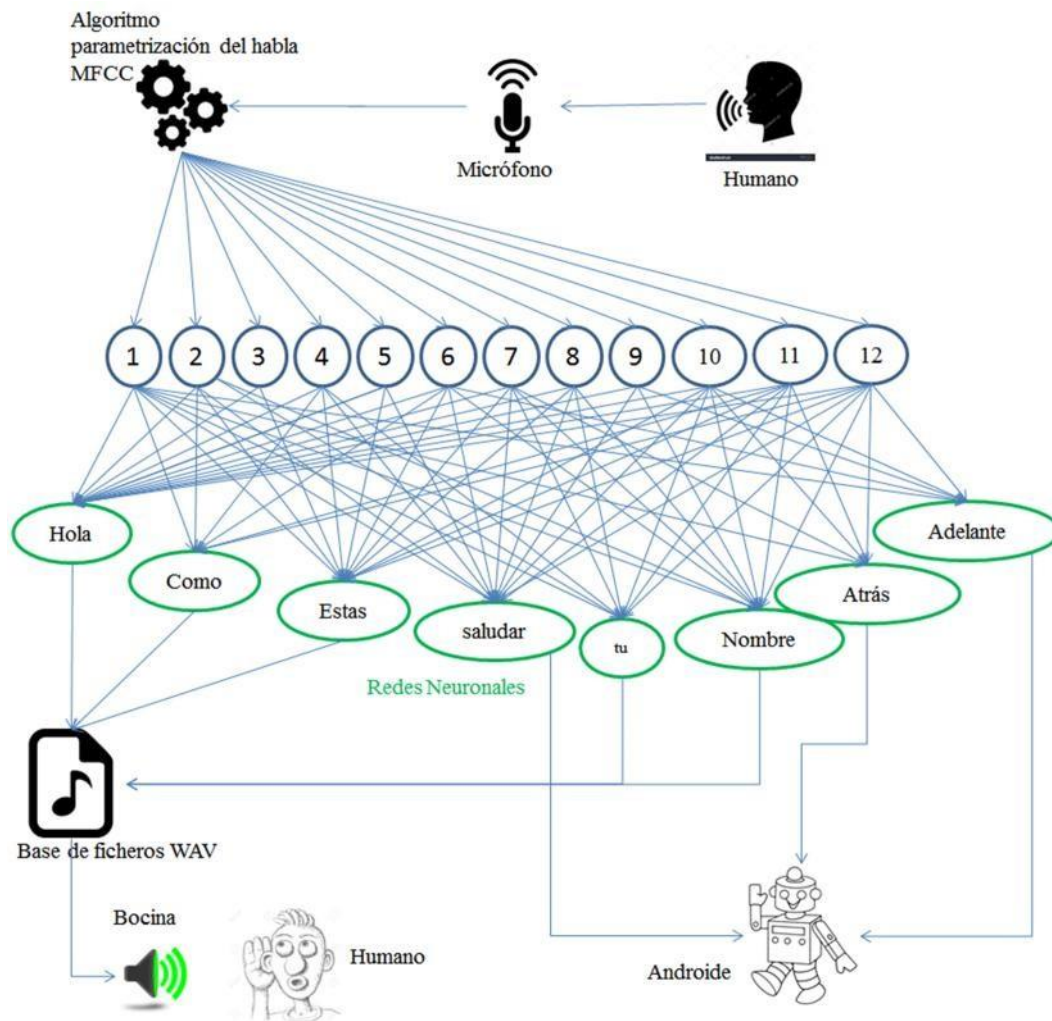


Figura 38. Diagrama del funcionamiento de emulación del razonamiento basada en conocimiento de hechos.

Se realiza una serie de pruebas para verificar la reacción de prototipo computacional respecto a las preguntas realizadas por el interlocutor humano. Se toma para la prueba una lista de 6 palabras / preguntas, de las cuales se realizan 10 intentos por cada palabra / pregunta, teniendo como resultados la siguiente tabla.

Tabla 6. Resultados de la pruebas razonamiento y autonomía.

Interlocución		Intentos satisfactorios	Intentos fallidos	Total Intentos
Humano	Androide			
Hola	Hola como estas	10	0	10
Hola como estas	bien	10	0	10
Tu nombre	Grievous	10	0	10
Avanzar	Movimiento de motores de piernas hacia adelante	10	0	10
Retroceder	Movimiento de motores de piernas hacia atrás	10	0	10
Saludar	Movimiento de brazos	10	0	10
	TOTAL	60	0	60

Cuando el interlocutor humano habla una orden o realiza una pregunta el androide procede a revisar su base de conocimientos e identificar sus respuestas, estas respuestas puede ser palabras (sonidos) o movimientos.

Se entrenó redes neuronales para almacenar algunas respuestas (Identificador de respuestas) respecto a diferentes entradas (Identificador de preguntas / identificador de palabras). En la Tabla 6, se muestra las diferentes reacciones del androide respecto al interlocutor humano, teniendo en cuenta que los patrones de sonido han sido identificados correctamente, en esta prueba solo se evalúa la reacción del androide a órdenes y preguntas correctamente identificadas.

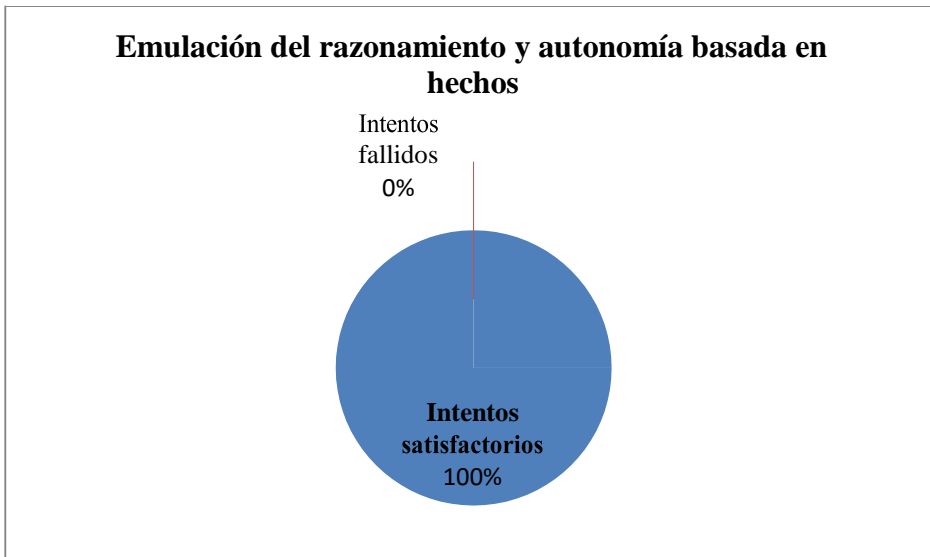


Figura 39. Acierto, intentos fallidos en la emulación de razonamiento y autonomía.

La gráfica expresa el número de intentos satisfactorios e intentos fallidos que están inscritos en el la gráfica de la figura 39, en la gráfica permiten hacer comparaciones entre el porcentaje de 0% de intentos fallidos respecto al 100% de intentos satisfactorios partiendo de que los patrones son correctamente identificados.

En las pruebas se verifican que no se tienen errores las razón es que para cada pregunta u orden recibida por el interlocutor humano existe una red neuronal asignada asegurando el resultado correcto.

Se debe tener en cuenta, que para una toma de decisiones no solo de debe tener en cuenta las pablas u órdenes, sino también la entonación de cada una de ellas. Por ejemplo si una persona grita colérica pidiendo algo, probablemente uno no reaccione de la mejor manera, siendo una variable a tomar en cuenta que el prototipo androide no está preparado para identificar ese tipo de características, limitando su rango de inferencia solo a la identificación de palabras, mas no los estados emocionales de un humano, otro ejemplo es la interfaz gráfica con la que no cuenta el prototipo androide, siendo una información vital para saber más del interlocutor y tomar una decisión más acertada.

CONCLUSIONES

La investigación se ha centrado en crear un prototipo de un Sistema de Inteligencia Artificial para el control de androides autónomos. La emulación del comportamiento racional, mecánico, humano es un desafío que muchas empresas, agencias, centros de investigación han emprendido, en estos últimos años, porque han identificado la gran importancia que tendrá en un futuro no muy lejano. Durante estas ultima décadas se han construidos androides con capacidades de autonomía aceptables como el androide Atlas de Dynamic Boston.

Las redes neuronales son una técnica viable para poder procesar información probabilística, muy habitual en sistemas con inteligencia artificial, por la enorme variedad de información imprecisa, ambigua. Si se acompaña de algoritmos que gestionen su correcto funcionamiento existen una enorme probabilidad de generar un agente con inteligencia artificial.

Respecto a la hipótesis general; se ha demostrado que se puede desarrollar un sistema autónomo que permite emular algunos comportamiento básicos de un humano, como escuchar, tomar decisiones en base a experiencia de hechos, y hablar reproduciendo sonidos previamente recordados, dado que el ser humano es casi la perfección en cuanto a evolución se trata, no es fácil traducir todo el raciocinio de un ser humano en fórmulas matemáticas dada que el ser humano equilibra sus decisiones en base a recuerdos, datos obtenidos del entorno y trivialidades del momento que no necesariamente son racionales. Uno de los métodos que nos aproxima en plasmar el comportamiento de un ser humano son las redes neuronales ya que trabajan con datos atípicos y retorna decisiones esperadas o similares al de un humano.

Respecto a la hipótesis específica 1; Se ha logrado capturar e identificar el reconocimiento de patrones mediante el desarrollo del algoritmo de Coeficientes Cepstrales en las Frecuencias de Mel (MFCC), se realizó las pruebas con 8 palabras de reconocimiento, de las cuales se obtuvo como resultado un 69% de intentos satisfactorios.

Respecto a la hipótesis específica 2; Se ha logrado el razonamiento de hechos mediante el algoritmo de red neuronal *backpropagation*, se entrenó la red neuronal de tal modo que se realizaron las pruebas de razonamiento, ordenando o preguntando al androide y este procede a revisar su base de conocimientos e identifica sus respuestas, estas respuestas se dan a través de órdenes de movimientos de cabeza, brazos y piernas. Como resultado se obtuvo un 100% de intentos satisfactorios, se tiene un razonamiento correctamente identificado.

RECOMENDACIONES

Desarrollar proyectos de investigación que utilicen como método el análisis de sonido el algoritmo MFCC, por su facilidad de uso y comprensión. Siendo la base para futuras investigaciones y mejoras y evoluciones de algoritmos más complejos y más sofisticados.

Desarrollar proyectos con redes neuronales, por ser la base para desarrollar algoritmos complejos autónomos y por su enorme gama de uso y variedad de aplicaciones y variantes que se puede ajustar a la necesidad del investigador.

Replicar el modelo planteado en el grado universitario, por ser de suma importancia para futuras investigaciones que pueden cultivar la imaginación del investigador.

Utilizar el modelo en células de investigación en la universidad.

BIBLIOGRAFÍA

- Ajith, A., Mauri, J.L., Buford, J.F., Suzuki, J. y Thampi, S.M. (2011) *Advances in Computing and Communications, Part III*. Obtenido de: https://books.google.com.pe/books?id=t41iF1dp9CEC&printsec=frontcover&hl=es&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false
- Basogain, X. “Redes Neuronales Artificiales y sus aplicaciones”.
- BCP (2017) Arturito BCP. Términos y condiciones. Recuperado de <https://www.viabcp.com/terminos-y-condiciones/terminos-condiciones-arturito>
- Bonnie Chantarotwong (2006). IMS-256 Final Project, Fall.
- Comercio. (2018), la industria que más crece en el mundo de la robótica, escrito por Ricardo Marquez. Recuperado de <https://elcomercio.pe/economia/negocios/sector-financiero-son-tendencias-robotizacion-noticia-525340>
- Conocimientosweb.net. (2014) Principios de la robótica. Recuperado de <https://www.conocimientosweb.net/portal/article2506.html>
- Cybenko, G. Cybenko. (1989). Approximations by superpositions of a sigmoidal function *Math. Control Signals Systems*, pp. 303–314. Sitio web: https://www.dartmouth.edu/~gvc/Cybenko_MCSS.pdf.
- C. R. Jankowski, H. D. H. Vo, and R. P. Lippmann. (July 1995). “A comparison of signal processing front ends for automatic word recognition,” *IEEE Trans. Speech Audio Process.*, vol. 3, no. 4.
- Chu, S.; Narayanan, S.; Jay Kuo, C.-C. (2009), Environmental sound recognition with time-frequency audio features. *IEEE Trans. Audio, Speech and Lang. Processing*. Vol. 17(6), pp. 1142-1158.
- García Córdoba, Fernando. (2009). La investigación tecnológica. Investigar, idear e innovar en Ingenierías y Ciencias Sociales. México: Editorial Limusa.
- Gestión (2016). Banco de Crédito lanza “Arturito Chat” Recuperado de <https://gestion.pe/economia/empresas/banco-credito-lanza-arturito-bcp-chat-cognitivo-facebook-messenger-149468>
- Gomez, S. (2012). Metodología de la investigación. Editorial: Red Tercer Milenio. Mexico. ISBN 978-607-733-149-0
- Hilera, J.R., y V.J. Martínez. 2000. Redes neuronales artificiales. Fundamentos, modelos y aplicaciones. 390 p. Editorial Alfaomega Ra-Ma, Madrid, España.
- Instituto Medico Leloir. Fundamentos de biomecánica articular. Recuperado de http://imedleloir.com.ar/documentos/Biomecanica_articular.pdf.

- Isaac Asimov. (1975). Yo Robot Barcelona: Impreso por Romany /Valls.
- James Gosling. (February 1995). Java: an Overview. Recuperado de <http://www.cs.dartmouth.edu/~mckeeman/cs118/references/OriginalJavaWhitepaper.pdf>
- Kostic, D. (2009). "Introduction Robotics".
- Koval, S. (2006). Androides y Posthumanos. La integración hombre-máquina, Vol 1. Pag. 1. Obtenido de: http://www.diegolevis.com.ar/secciones/Articulos/santiago_koval1.pdf
- Microchip. (2009) PIC18F2455/2550/4455/4550 Data Sheet. Sitio web: <http://ww1.microchip.com/downloads/en/DeviceDoc/39632e.pdf>.
- Negnevitsky, M. (2005). Artificial Intelligence, segunda edición. Pag 1.
- Sun Microsystems. (octubre 1995). The Java Language Environment A White Paper. Sitio web: <http://tech-insider.org/java/research/acrobat/9505.pdf>.
- Stroustrup, B. (2013) C++ Programming Language. USA. Addison-Wesley. Pearson Education
- S. B. Davis and P. Mermelstein. (1980). "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," IEEE Trans. Acoust., Speech, Signal Processing, vol. 28(4), pp. 357–366.
- Rabiner, L.; Juang, 1993, B. Fundamentals of speech recognition, Prentice Hall. Sitio Web: <http://www.stillhq.com/diary/asr-short.pdf>
- Romero, J.J., Dafonte, C., Gomez, A. y Penousal, F.J. (2007). Inteligencia Artificial y Computación Avanzada. Publicaciones de la Fundación Alfredo Brañas. ISBN.: 84-934497-0-9.
- R. Sproat, (1998). "Multilingual Text-to-Speech Synthesis", *KLUWER academic publishers*.
- Valero, X.; Farré, P.; Alías, F. (2011) Comparison of Machine Learning Techniques for the Automatic Recognition of Soundscapes. Forum Acusticum 2011, Aalborg (Dinamarca).
- Universidad Tecnológica de Pereira. Facultad de Ingeniería Eléctrica (2000). Tutorial redes neuronales. Pereira, Colombia. Disponible en <http://medicinaycomplejidad.org/pdf/redes/Backpropagation.pdf>

ANEXO 1:



Figura 40. Androide construido.

ANEXO 2: INTERFACE DE ENTRENADOR DE REDES NEURONALES

La siguiente pantalla permite la capturar de sonidos en formato WAV, y obtener una patrón de 12 valores que luego serán la entraras de las redes neuronales. En esta pantalla se puede configurar el evento, sistema, grupo de red neuronal donde se agruparan las redes neuronales clasificadas por patrón (palabras, movimientos) de acuerdo al criterio del administrador. Esta pantalla incluye la funcionalidad de entrenar manualmente las redes neuronales.

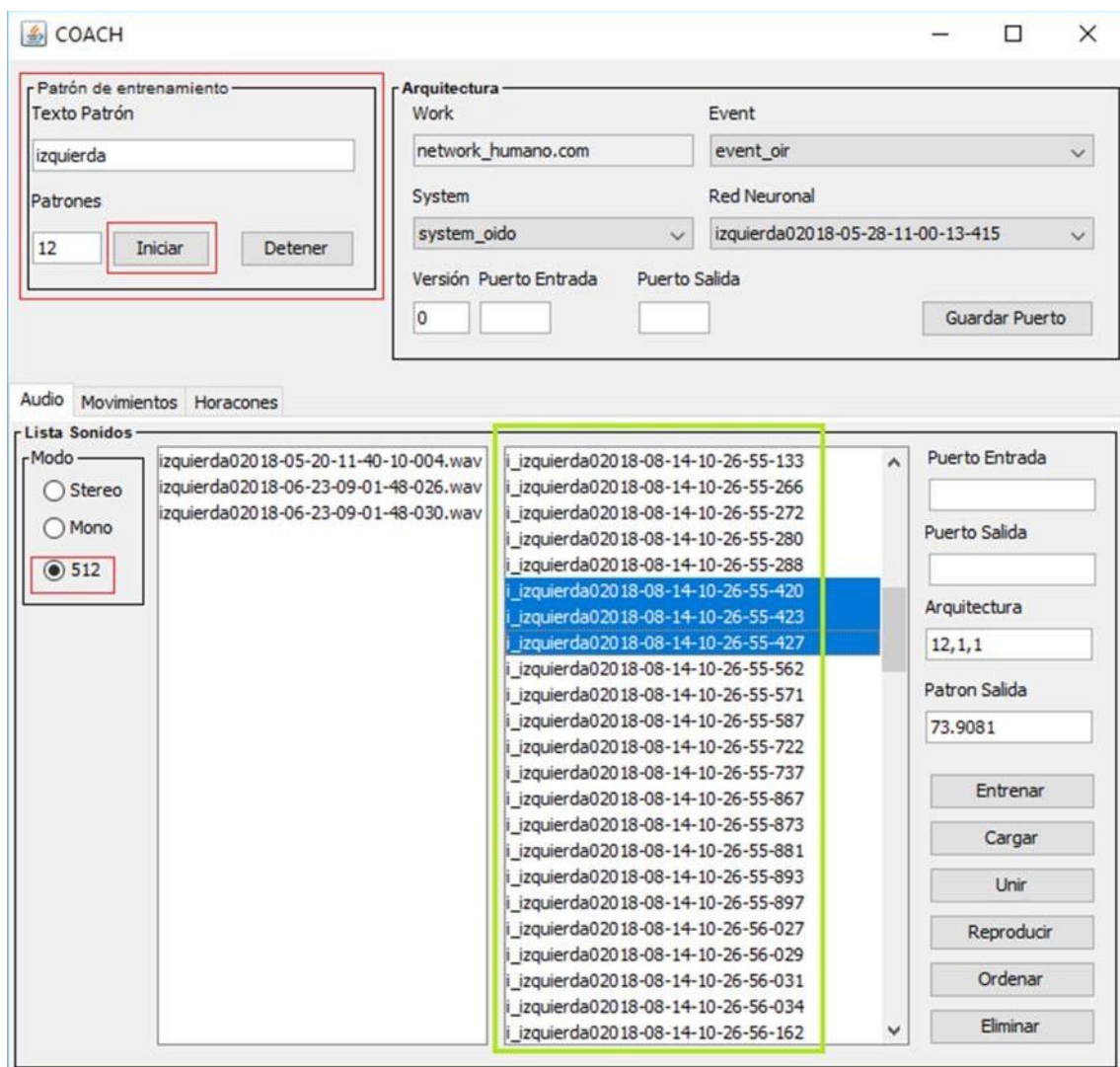


Figura 41. Interfaz de usuario para grabar archivos planos de parámetros de sonido.

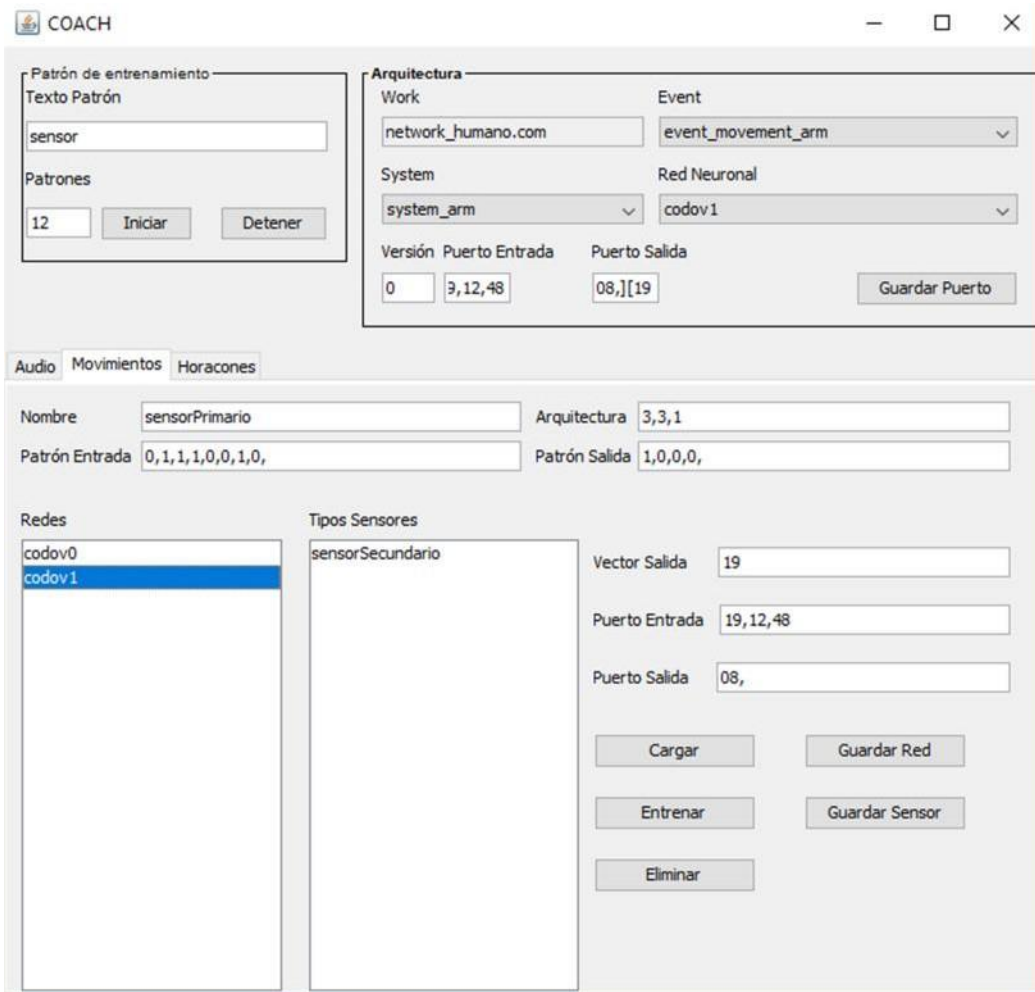


Figura 42. Entrenador de redes neuronales.

ANEXO 3: ENTRENAMIENTO DE UNA RED NEURONAL

Doce entradas de un patrón de sonido

```
double Entradas[][] =  
{ {2.933293311231783,0.7910251009150709,9.97121845170498,7.569186021720711,  
6.787793361491363,5.117052212897864,1.6473752977875957,1.4291999504154138,  
11.39146170066378,7.323455465651588,9.156415284213669,8.234094668376793,} };
```

Una salida de un patrón de sonido

```
double Salida[][] = { {1.0} };
```

Arquitectura de la red neuronal; 12 entradas, 1 neurona capa intermedia, 1 neurona en la salida.

```
int Arquitectura[]={ 12,1,1 };
```

Peso de la capa de entrada (Peso[capa][salidas][entradas]), se tiene una dos capas (0,1), una capa intermedia (0), doce capas de entrada (1,2,3,4,5,6,7,8,9,10,11,12).

```
Peso [0][0][0]=5.552763261691283;  
Peso [0][0][1]=-4.8383752690038255;  
Peso [0][0][2]=6.24661148008914;  
Peso [0][0][3]=-13.373427893977771;  
Peso [0][0][4]=-19.042367866320895;  
Peso [0][0][5]=-3.3936202802120246;  
Peso [0][0][6]=-6.629396011881022;  
Peso [0][0][7]=-7.903751196279768;  
Peso [0][0][8]=0.3234287191854117;  
Peso [0][0][9]=16.788153948640883;  
Peso [0][0][10]=-17.33236438462546;  
Peso [0][0][11]=2.2996011841137474;  
Peso [1][0][0]=0.001;
```

Peso de umbrales (Umbral [capa][salidas][entradas]), un umbral en la capa intermedia y en la capa de salida.

```
Umbral [0][0][0]=-1340.301306273662;  
Umbral [1][0][0]=68.6982;
```

Tasa de aprendizaje: 0.05

Resultado.

La red neuronal aprende en 23 épocas.

Log de los procesos que se están ejecutando cuando se entrena una red neuronal.

```
2018-05-08 21:44:50 DEBUG CoreAlfa5: Capa 1; filas:1 columnas:12  
2018-05-08 21:44:50 DEBUG CoreAlfa5: Capa 2; filas:1 columnas:1  
2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -  
====MUESTRA=====0  
2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - ====CAPA=====1  
2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0  
2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - ====CAPA=====2
```

2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=68.697
 2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=0, ERROR===== > -
 67.697=1.0-68.697; MUESTRA=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
 Y LIMITES=====
 2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.552763261691283-
 0.1*0.0*2.933293311231783
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8383752690038255-
 0.1*0.0*0.7910251009150709
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.24661148008914-
 0.1*0.0*9.97121845170498
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.373427893977771-
 0.1*0.0*7.569186021720711
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.042367866320895-
 0.1*0.0*6.787793361491363
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936202802120246-
 0.1*0.0*5.117052212897864
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.629396011881022-
 0.1*0.0*1.6473752977875957
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.903751196279768-
 0.1*0.0*1.4291999504154138
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234287191854117-
 0.1*0.0*11.39146170066378
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.788153948640883-
 0.1*0.0*7.323455465651588
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.33236438462546-
 0.1*0.0*9.156415284213669
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996011841137474-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.301306273662-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.001-0.1*135.394*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=13.5404
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=68.6982-0.1*135.394

2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=55.1588 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -

===== 2018-05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====0=====>

2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -

====MUESTRA=====0

2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - ====CAPA=====1

2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0

2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - ====CAPA=====2

2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=41.618

2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=1, ERROR=====> - 40.618=1.0-41.618; MUESTRA=0

2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS Y LIMITES=====

2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-0.1*0.0*2.933293311231783

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-0.1*0.0*0.7910251009150709

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-0.1*0.0*9.97121845170498

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-0.1*0.0*7.569186021720711

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-0.1*0.0*6.787793361491363

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-0.1*0.0*5.117052212897864

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-0.1*0.0*1.6473752977875957

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-0.1*0.0*1.4291999504154138

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-0.1*0.0*11.39146170066378

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-0.1*0.0*7.323455465651588

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-0.1*0.0*9.156415284213669

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=13.5404-0.1*81.236*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=21.664
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=55.1588-0.1*81.236
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=47.0352 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====1=====>
 2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
 =====MUESTRA=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====1
 2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
 2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=25.371
 2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=2, ERROR=====> -
 24.371=1.0-25.371; MUESTRA=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
 Y LIMITES=====
 2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
 0.1*0.0*2.933293311231783
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
 0.1*0.0*0.7910251009150709
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
 0.1*0.0*9.97121845170498
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
 0.1*0.0*7.569186021720711
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
 0.1*0.0*6.787793361491363
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
 0.1*0.0*5.117052212897864
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
 0.1*0.0*1.6473752977875957
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
 0.1*0.0*1.4291999504154138

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
 0.1*0.0*11.39146170066378
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
 0.1*0.0*7.323455465651588
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
 0.1*0.0*9.156415284213669
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=21.664-0.1*48.742*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=26.5382
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=47.0352-0.1*48.742
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=42.161 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====2=====>
 2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
 =====MUESTRA=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====1
 2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
 2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=15.623
 2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=3, ERROR=====> -
 14.623=1.0-15.623; MUESTRA=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
 Y LIMITES=====
 2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
 0.1*0.0*2.933293311231783
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
 0.1*0.0*0.7910251009150709
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
 0.1*0.0*9.97121845170498
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
 0.1*0.0*7.569186021720711
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
 0.1*0.0*6.787793361491363
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
 0.1*0.0*5.117052212897864
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
 0.1*0.0*1.6473752977875957
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
 0.1*0.0*1.4291999504154138
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
 0.1*0.0*11.39146170066378
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
 0.1*0.0*7.323455465651588
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
 0.1*0.0*9.156415284213669
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=26.5382-0.1*29.246*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=29.4628
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=42.161-0.1*29.246
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=39.2364 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====3=====>
 2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
 =====MUESTRA=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====1
 2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
 2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=9.774
 2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=4, ERROR=====> -
 8.774=1.0-9.774; MUESTRA=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
 Y LIMITES=====
 2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
 0.1*0.0*2.933293311231783

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
 0.1*0.0*0.7910251009150709
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
 0.1*0.0*9.97121845170498
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
 0.1*0.0*7.569186021720711
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
 0.1*0.0*6.787793361491363
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
 0.1*0.0*5.117052212897864
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
 0.1*0.0*1.6473752977875957
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
 0.1*0.0*1.4291999504154138
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
 0.1*0.0*11.39146170066378
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
 0.1*0.0*7.323455465651588
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
 0.1*0.0*9.156415284213669
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=29.4628-0.1*17.548*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=31.2176
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=39.2364-0.1*17.548
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=37.4816 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====4=====>
 2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
 =====MUESTRA=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====1

2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
 2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=6.264
 2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=5, ERROR=====> -
 5.264=1.0-6.264; MUESTRA=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
 Y LIMITES=====
 2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
 0.1*0.0*2.933293311231783
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
 0.1*0.0*0.7910251009150709
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
 0.1*0.0*9.97121845170498
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
 0.1*0.0*7.569186021720711
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
 0.1*0.0*6.787793361491363
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
 0.1*0.0*5.117052212897864
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
 0.1*0.0*1.6473752977875957
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
 0.1*0.0*1.4291999504154138
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
 0.1*0.0*11.39146170066378
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
 0.1*0.0*7.323455465651588
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
 0.1*0.0*9.156415284213669
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=31.2176-0.1*10.528*-1.0

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=32.2704
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=37.4816-0.1*10.528
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=36.4288 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====5=====>
 2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
 =====MUESTRA=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====1
 2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
 2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=4.158
 2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=6, ERROR=====> -
 3.1580000000000004=1.0-4.158; MUESTRA=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
 Y LIMITES=====
 2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
 0.1*0.0*2.933293311231783
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
 0.1*0.0*0.7910251009150709
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
 0.1*0.0*9.97121845170498
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
 0.1*0.0*7.569186021720711
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
 0.1*0.0*6.787793361491363
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
 0.1*0.0*5.117052212897864
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
 0.1*0.0*1.6473752977875957
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
 0.1*0.0*1.4291999504154138
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
 0.1*0.0*11.39146170066378
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
 0.1*0.0*7.323455465651588
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
 0.1*0.0*9.156415284213669

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=32.2704-0.1*6.316*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=32.902
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=36.4288-0.1*6.316
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=35.7972 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====6=====>
 2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
 =====MUESTRA=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====1
 2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
 2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=2.895
 2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=7, ERROR=====> -
 1.895=1.0-2.895; MUESTRA=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
 Y LIMITES=====
 2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
 0.1*0.0*2.933293311231783
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
 0.1*0.0*0.7910251009150709
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
 0.1*0.0*9.97121845170498
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
 0.1*0.0*7.569186021720711
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
 0.1*0.0*6.787793361491363
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
 0.1*0.0*5.117052212897864
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
 0.1*0.0*1.6473752977875957
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
 0.1*0.0*1.4291999504154138
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
 0.1*0.0*11.39146170066378
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
 0.1*0.0*7.323455465651588
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
 0.1*0.0*9.156415284213669
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=32.902-0.1*3.79*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=33.281
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=35.7972-0.1*3.79
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=35.4182 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====7=====>
 2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
 =====MUESTRA=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====1
 2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
 2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=2.137
 2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=8, ERROR=====> -
 1.137=1.0-2.137; MUESTRA=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
 Y LIMITES=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
 0.1*0.0*2.933293311231783
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
 0.1*0.0*0.7910251009150709
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
 0.1*0.0*9.97121845170498
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
 0.1*0.0*7.569186021720711

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
 0.1*0.0*6.787793361491363
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
 0.1*0.0*5.117052212897864
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
 0.1*0.0*1.6473752977875957
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
 0.1*0.0*1.4291999504154138
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
 0.1*0.0*11.39146170066378
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
 0.1*0.0*7.323455465651588
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
 0.1*0.0*9.156415284213669
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=33.281-0.1*2.274*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=33.5084
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=35.4182-0.1*2.274
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=35.1908 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====8=====>
 2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
 =====MUESTRA=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====1
 2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
 2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=1.682
 2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=9, ERROR=====> -
 0.6819999999999999=1.0-1.682; MUESTRA=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
 Y LIMITES=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
 0.1*0.0*2.933293311231783
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
 0.1*0.0*0.7910251009150709
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
 0.1*0.0*9.97121845170498
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
 0.1*0.0*7.569186021720711
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
 0.1*0.0*6.787793361491363
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
 0.1*0.0*5.117052212897864
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
 0.1*0.0*1.6473752977875957
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
 0.1*0.0*1.4291999504154138
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
 0.1*0.0*11.39146170066378
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
 0.1*0.0*7.323455465651588
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
 0.1*0.0*9.156415284213669
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=33.5084-0.1*1.364*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=33.6448
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=35.1908-0.1*1.364
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=35.0544 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====9=====>

2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
=====MUESTRA=====0
2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====1
2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=1.41
2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=10, ERROR=====> -
0.4099999999999999=1.0-1.41; MUESTRA=0
2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
Y LIMITES=====0
2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
0.1*0.0*2.933293311231783
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
0.1*0.0*0.7910251009150709
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
0.1*0.0*9.97121845170498
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
0.1*0.0*7.569186021720711
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
0.1*0.0*6.787793361491363
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
0.1*0.0*5.117052212897864
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
0.1*0.0*1.6473752977875957
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
0.1*0.0*1.4291999504154138
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
0.1*0.0*11.39146170066378
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
0.1*0.0*7.323455465651588
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
0.1*0.0*9.156415284213669
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
0.1*0.0*8.234094668376793
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013

```

2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
===== 2018-
05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=33.6448-0.1*0.82*-1.0
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=33.7268
2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=35.0544-0.1*0.82
2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=34.9724 2018-
05-08 21:44:50 DEBUG CoreAlfa5:208 -
===== 2018-
05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====10=====>
2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
====MUESTRA=====0
2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - ====CAPA=====1
2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - ====CAPA=====2
2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=1.246
2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=11, ERROR=====> -
0.246=1.0-1.246; MUESTRA=0
2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
Y LIMITES=====
2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
0.1*0.0*2.933293311231783
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
0.1*0.0*0.7910251009150709
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
0.1*0.0*9.97121845170498
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
0.1*0.0*7.569186021720711
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
0.1*0.0*6.787793361491363
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
0.1*0.0*5.117052212897864
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
0.1*0.0*1.6473752977875957
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
0.1*0.0*1.4291999504154138
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
0.1*0.0*11.39146170066378
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234

```

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
 0.1*0.0*7.323455465651588
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
 0.1*0.0*9.156415284213669
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=33.7268-0.1*0.492*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=33.776
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=34.9724-0.1*0.492
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=34.9232 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====11=====>
 2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
 =====MUESTRA=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====1
 2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
 2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=1.147
 2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=12, ERROR=====> -
 0.14700000000000002=1.0-1.147; MUESTRA=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
 Y LIMITES=====
 2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
 0.1*0.0*2.933293311231783
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
 0.1*0.0*0.7910251009150709
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
 0.1*0.0*9.97121845170498
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
 0.1*0.0*7.569186021720711
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
 0.1*0.0*6.787793361491363
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
 0.1*0.0*5.117052212897864

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
 0.1*0.0*1.6473752977875957
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
 0.1*0.0*1.4291999504154138
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
 0.1*0.0*11.39146170066378
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
 0.1*0.0*7.323455465651588
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
 0.1*0.0*9.156415284213669
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=33.776-0.1*0.294*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=33.8054
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=34.9232-0.1*0.294
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=34.8938 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====12=====>
 2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
 =====MUESTRA=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====1
 2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
 2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=1.088
 2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=13, ERROR=====> -
 0.08800000000000008=1.0-1.088; MUESTRA=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
 Y LIMITES=====
 2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
 0.1*0.0*2.933293311231783
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
 0.1*0.0*0.7910251009150709
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384

```

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
0.1*0.0*9.97121845170498
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
0.1*0.0*7.569186021720711
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
0.1*0.0*6.787793361491363
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
0.1*0.0*5.117052212897864
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
0.1*0.0*1.6473752977875957
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
0.1*0.0*1.4291999504154138
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
0.1*0.0*11.39146170066378
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
0.1*0.0*7.323455465651588
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
0.1*0.0*9.156415284213669
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
0.1*0.0*8.234094668376793
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
===== 2018-
05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=33.8054-0.1*0.176*-1.0
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=33.823
2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=34.8938-0.1*0.176
2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=34.8762 2018-
05-08 21:44:50 DEBUG CoreAlfa5:208 -
===== 2018-
05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====13=====>
2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
====MUESTRA=====0
2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - ====CAPA=====1
2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - ====CAPA=====2
2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=1.053

```

2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=14, ERROR=====> -
 0.052999999999999936=1.0-1.053; MUESTRA=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
 Y LIMITES=====
 2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
 0.1*0.0*2.933293311231783
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
 0.1*0.0*0.7910251009150709
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
 0.1*0.0*9.97121845170498
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
 0.1*0.0*7.569186021720711
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
 0.1*0.0*6.787793361491363
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
 0.1*0.0*5.117052212897864
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
 0.1*0.0*1.6473752977875957
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
 0.1*0.0*1.4291999504154138
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
 0.1*0.0*11.39146170066378
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
 0.1*0.0*7.323455465651588
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
 0.1*0.0*9.156415284213669
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=33.823-0.1*0.106*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=33.8336
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=34.8762-0.1*0.106
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=34.8656

```

2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
===== 2018-
05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====14=====>
2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
=====MUESTRA=====0
2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====1
2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=1.032
2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=15, ERROR=====> -
0.032000000000000003=1.0-1.032; MUESTRA=0
2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
Y LIMITES=====
2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
0.1*0.0*2.933293311231783
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
0.1*0.0*0.7910251009150709
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
0.1*0.0*9.97121845170498
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
0.1*0.0*7.569186021720711
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
0.1*0.0*6.787793361491363
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
0.1*0.0*5.117052212897864
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
0.1*0.0*1.6473752977875957
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
0.1*0.0*1.4291999504154138
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
0.1*0.0*11.39146170066378
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
0.1*0.0*7.323455465651588
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
0.1*0.0*9.156415284213669
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
0.1*0.0*8.234094668376793

```

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=33.8336-0.1*0.064*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=33.84
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=34.8656-0.1*0.064
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=34.8592 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====15=====>
 2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
 =====MUESTRA=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====1
 2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
 2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=1.019
 2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=16, ERROR=====> -
 0.018999999999999906=1.0-1.019; MUESTRA=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
 Y LIMITES=====
 2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
 0.1*0.0*2.933293311231783
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
 0.1*0.0*0.7910251009150709
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
 0.1*0.0*9.97121845170498
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
 0.1*0.0*7.569186021720711
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
 0.1*0.0*6.787793361491363
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
 0.1*0.0*5.117052212897864
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
 0.1*0.0*1.6473752977875957
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
 0.1*0.0*1.4291999504154138
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
 0.1*0.0*11.39146170066378
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
 0.1*0.0*7.323455465651588
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
 0.1*0.0*9.156415284213669
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=33.84-0.1*0.038*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=33.8438
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=34.8592-0.1*0.038
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=34.8554 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====16=====>
 2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
 =====MUESTRA=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====1
 2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
 2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=1.012
 2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=17, ERROR=====> -
 0.012000000000000001=1.0-1.012; MUESTRA=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
 Y LIMITES=====
 2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
 0.1*0.0*2.933293311231783
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
 0.1*0.0*0.7910251009150709
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
 0.1*0.0*9.97121845170498
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
 0.1*0.0*7.569186021720711
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
 0.1*0.0*6.787793361491363

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
 0.1*0.0*5.117052212897864
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
 0.1*0.0*1.6473752977875957
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
 0.1*0.0*1.4291999504154138
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
 0.1*0.0*11.39146170066378
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
 0.1*0.0*7.323455465651588
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
 0.1*0.0*9.156415284213669
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=33.8438-0.1*0.024*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=33.8462
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=34.8554-0.1*0.024
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=34.853 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====17=====>
 2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
 =====MUESTRA=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====1
 2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
 2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=1.007
 2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=18, ERROR=====> -
 0.0069999999999999895=1.0-1.007; MUESTRA=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
 Y LIMITES=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
 0.1*0.0*2.933293311231783
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528

```

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
0.1*0.0*0.7910251009150709
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
0.1*0.0*9.97121845170498
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
0.1*0.0*7.569186021720711
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
0.1*0.0*6.787793361491363
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
0.1*0.0*5.117052212897864
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
0.1*0.0*1.6473752977875957
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
0.1*0.0*1.4291999504154138
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
0.1*0.0*11.39146170066378
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
0.1*0.0*7.323455465651588
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
0.1*0.0*9.156415284213669
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
0.1*0.0*8.234094668376793
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
===== 2018-
05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=33.8462-0.1*0.014*-1.0
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=33.8476
2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=34.853-0.1*0.014
2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=34.8516 2018-
05-08 21:44:50 DEBUG CoreAlfa5:208 -
===== 2018-
05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====18=====>
2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
====MUESTRA=====0
2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - ====CAPA=====1
2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0

```

2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
 2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=1.004
 2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=19, ERROR=====> -
 0.00400000000000000036=1.0-1.004; MUESTRA=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
 Y LIMITES=====
 2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
 0.1*0.0*2.933293311231783
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
 0.1*0.0*0.7910251009150709
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
 0.1*0.0*9.97121845170498
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
 0.1*0.0*7.569186021720711
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
 0.1*0.0*6.787793361491363
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
 0.1*0.0*5.117052212897864
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
 0.1*0.0*1.6473752977875957
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
 0.1*0.0*1.4291999504154138
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
 0.1*0.0*11.39146170066378
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
 0.1*0.0*7.323455465651588
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
 0.1*0.0*9.156415284213669
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=33.8476-0.1*0.008*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=33.8484

```

2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=34.8516-0.1*0.008
2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=34.8508 2018-
05-08 21:44:50 DEBUG CoreAlfa5:208 -
===== 2018-
05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====19=====>
2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
=====MUESTRA=====0
2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====1
2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=1.002
2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=20, ERROR=====> -
0.0020000000000000018=1.0-1.002; MUESTRA=0
2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
Y LIMITES=====
2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
0.1*0.0*2.933293311231783
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
0.1*0.0*0.7910251009150709
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
0.1*0.0*9.97121845170498
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
0.1*0.0*7.569186021720711
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
0.1*0.0*6.787793361491363
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
0.1*0.0*5.117052212897864
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
0.1*0.0*1.6473752977875957
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
0.1*0.0*1.4291999504154138
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
0.1*0.0*11.39146170066378
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
0.1*0.0*7.323455465651588
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
0.1*0.0*9.156415284213669
2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324

```

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=33.8484-0.1*0.004*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=33.8488
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=34.8508-0.1*0.004
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=34.8504 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====20=====>
 2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
 =====MUESTRA=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====1
 2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
 2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=1.002
 2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=21, ERROR=====> -
 0.00200000000000000018=1.0-1.002; MUESTRA=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
 Y LIMITES=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
 0.1*0.0*2.933293311231783
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
 0.1*0.0*0.7910251009150709
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
 0.1*0.0*9.97121845170498
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
 0.1*0.0*7.569186021720711
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
 0.1*0.0*6.787793361491363
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
 0.1*0.0*5.117052212897864
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
 0.1*0.0*1.6473752977875957
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
 0.1*0.0*1.4291999504154138

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
 0.1*0.0*11.39146170066378
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
 0.1*0.0*7.323455465651588
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
 0.1*0.0*9.156415284213669
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=33.8488-0.1*0.004*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=33.8492
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=34.8504-0.1*0.004
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=34.85 2018-05-
 08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====21=====>
 2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
 =====MUESTRA=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====1
 2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
 2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=1.001
 2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=22, ERROR=====> -
 9.99999999998899E-4=1.0-1.001; MUESTRA=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
 Y LIMITES=====
 2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
 0.1*0.0*2.933293311231783
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
 0.1*0.0*0.7910251009150709
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
 0.1*0.0*9.97121845170498
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
 0.1*0.0*7.569186021720711
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734

2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
 0.1*0.0*6.787793361491363
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
 0.1*0.0*5.117052212897864
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
 0.1*0.0*1.6473752977875957
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
 0.1*0.0*1.4291999504154138
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
 0.1*0.0*11.39146170066378
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
 0.1*0.0*7.323455465651588
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
 0.1*0.0*9.156415284213669
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=33.8492-0.1*0.002*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=33.8494
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=34.85-0.1*0.002
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=34.8498 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====22=====>
 2018-05-08 21:44:50 DEBUG CoreAlfa5:139 -
 =====MUESTRA=====0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====1
 2018-05-08 21:44:50 DEBUG CoreAlfa5:155 - Capa=1 fila=0 salida=-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:146 - =====CAPA=====2
 2018-05-08 21:44:50 DEBUG CoreAlfa5:158 - 2 fila=0 salida=1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:164 - EPOCA=23, ERROR=====>
 0.0=1.0-1.0; MUESTRA=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:190 - =====ACTUALIZAR PESOS
 Y LIMITES=====
 2018-05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=0 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=5.5528-
 0.1*0.0*2.933293311231783

2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=5.5528
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-4.8384-
 0.1*0.0*0.7910251009150709
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-4.8384
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=6.2466-
 0.1*0.0*9.97121845170498
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=6.2466
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-13.3734-
 0.1*0.0*7.569186021720711
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-13.3734
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-19.0424-
 0.1*0.0*6.787793361491363
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-19.0424
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-3.3936-
 0.1*0.0*5.117052212897864
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-3.3936
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-6.6294-
 0.1*0.0*1.6473752977875957
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-6.6294
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-7.9038-
 0.1*0.0*1.4291999504154138
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-7.9038
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=0.3234-
 0.1*0.0*11.39146170066378
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=0.3234
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=16.7882-
 0.1*0.0*7.323455465651588
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=16.7882
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=-17.3324-
 0.1*0.0*9.156415284213669
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=-17.3324
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=2.2996-
 0.1*0.0*8.234094668376793
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=2.2996
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=-1340.3013-0.1*0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=-1340.3013
 2018-05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:196 - Capa=1 fila=0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:199 - peso=33.8494-0.1*-0.0*-1.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:202 - nuevo peso=33.8494
 2018-05-08 21:44:50 DEBUG CoreAlfa5:205 - umbral=34.8498-0.1*-0.0
 2018-05-08 21:44:50 DEBUG CoreAlfa5:207 - nuevo umbral=34.8498 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:208 -
 ===== 2018-
 05-08 21:44:50 DEBUG CoreAlfa5:224 - EPOCA=====23=====>

ANEXO 3

2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>event_listen.com;
url=C:\UnidadD\dbneurona\network\event\event_listen.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 -
Fichero==>event_movement_arm.com;
url=C:\UnidadD\dbneurona\network\event\event_movement_arm.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>event_movement_leg.com;
url=C:\UnidadD\dbneurona\network\event\event_movement_leg.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>codov0c.com;
url=C:\UnidadD\dbneurona\network\event\system\net\codov0c.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>codov1c.com;
url=C:\UnidadD\dbneurona\network\event\system\net\codov1c.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>derecha02018-05-28-11-
00-40-856.com; url=C:\UnidadD\dbneurona\network\event\system\net\derecha02018-
05-28-11-00-40-856.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_abajo02018-08-14-10-
27-47-878.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_abajo02018-08-14-10-27-
47-878.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_abajo02018-08-14-10-
27-47-881.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_abajo02018-08-14-10-27-
47-881.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_abajo02018-08-14-10-
27-47-885.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_abajo02018-08-14-10-27-
47-885.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_abajo02018-08-14-10-
27-47-890.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_abajo02018-08-14-10-27-
47-890.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_abajo02018-08-14-10-
27-47-894.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_abajo02018-08-14-10-27-
47-894.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_arriba02018-08-14-10-
27-33-941.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_arriba02018-08-14-10-
27-33-941.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_arriba02018-08-14-10-
27-33-945.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_arriba02018-08-14-10-
27-33-945.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_arriba02018-08-14-10-
27-33-950.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_arriba02018-08-14-10-
27-33-950.com

2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_arriba02018-08-14-10-27-34-080.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_arriba02018-08-14-10-27-34-080.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_arriba02018-08-14-10-27-34-084.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_arriba02018-08-14-10-27-34-084.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_brazo02018-08-14-10-25-41-833.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_brazo02018-08-14-10-25-41-833.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_brazo02018-08-14-10-25-41-900.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_brazo02018-08-14-10-25-41-900.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_brazo02018-08-14-10-25-41-909.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_brazo02018-08-14-10-25-41-909.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_brazo02018-08-14-10-25-42-040.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_brazo02018-08-14-10-25-42-040.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_brazo02018-08-14-10-25-42-043.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_brazo02018-08-14-10-25-42-043.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_cabeza02018-08-14-10-26-10-671.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_cabeza02018-08-14-10-26-10-671.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_cabeza02018-08-14-10-26-10-675.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_cabeza02018-08-14-10-26-10-675.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_cabeza02018-08-14-10-26-10-679.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_cabeza02018-08-14-10-26-10-679.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_cabeza02018-08-14-10-26-10-943.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_cabeza02018-08-14-10-26-10-943.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_cabeza02018-08-14-10-26-11-410.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_cabeza02018-08-14-10-26-11-410.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_derecha02018-08-14-10-27-15-583.com;

url=C:\UnidadD\dbneurona\network\event\system\net\input\i_derecha02018-08-14-10-27-15-583.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_derecha02018-08-14-10-27-15-591.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_derecha02018-08-14-10-27-15-591.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_derecha02018-08-14-10-27-15-724.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_derecha02018-08-14-10-27-15-724.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_derecha02018-08-14-10-27-15-730.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_derecha02018-08-14-10-27-15-730.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_derecha02018-08-14-10-27-15-738.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_derecha02018-08-14-10-27-15-738.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_fin02018-08-14-10-28-34-817.com; url=C:\UnidadD\dbneurona\network\event\system\net\input\i_fin02018-08-14-10-28-34-817.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_fin02018-08-14-10-28-34-823.com; url=C:\UnidadD\dbneurona\network\event\system\net\input\i_fin02018-08-14-10-28-34-823.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_fin02018-08-14-10-28-34-831.com; url=C:\UnidadD\dbneurona\network\event\system\net\input\i_fin02018-08-14-10-28-34-831.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_fin02018-08-14-10-28-34-961.com; url=C:\UnidadD\dbneurona\network\event\system\net\input\i_fin02018-08-14-10-28-34-961.com
2018-10-10 22:48:36 DEBUG ManagerFile:69 - Fichero==>i_fin02018-08-14-10-28-34-967.com; url=C:\UnidadD\dbneurona\network\event\system\net\input\i_fin02018-08-14-10-28-34-967.com
2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>i_izquierda02018-08-14-10-26-55-110.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_izquierda02018-08-14-10-26-55-110.com
2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>i_izquierda02018-08-14-10-26-55-118.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_izquierda02018-08-14-10-26-55-118.com
2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>i_izquierda02018-08-14-10-26-55-125.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_izquierda02018-08-14-10-26-55-125.com
2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>i_izquierda02018-08-14-10-26-55-133.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_izquierda02018-08-14-10-26-55-133.com

2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>i_izquierda02018-08-14-10-26-55-266.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_izquierda02018-08-14-10-26-55-266.com

2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>i_pierna02018-08-14-10-26-29-933.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_pierna02018-08-14-10-26-29-933.com

2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>i_pierna02018-08-14-10-26-29-938.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_pierna02018-08-14-10-26-29-938.com

2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>i_pierna02018-08-14-10-26-29-943.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_pierna02018-08-14-10-26-29-943.com

2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>i_pierna02018-08-14-10-26-30-076.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_pierna02018-08-14-10-26-30-076.com

2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>i_pierna02018-08-14-10-26-30-081.com;
url=C:\UnidadD\dbneurona\network\event\system\net\input\i_pierna02018-08-14-10-26-30-081.com

2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>izquierda02018-05-28-11-00-13-415.com; url=C:\UnidadD\dbneurona\network\event\system\net\izquierda02018-05-28-11-00-13-415.com

2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>o_derecha02018-05-28-11-00-40-000.com;
url=C:\UnidadD\dbneurona\network\event\system\net\output\o_derecha02018-05-28-11-00-40-000.com

2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>o_derecha02018-05-28-11-00-40-856.com;
url=C:\UnidadD\dbneurona\network\event\system\net\output\o_derecha02018-05-28-11-00-40-856.com

2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>o_izquierda02018-05-28-11-00-13-415.com;
url=C:\UnidadD\dbneurona\network\event\system\net\output\o_izquierda02018-05-28-11-00-13-415.com

2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>o_sensorPrimario.com;
url=C:\UnidadD\dbneurona\network\event\system\net\output\o_sensorPrimario.com

2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>o_sensorSecundario.com;
url=C:\UnidadD\dbneurona\network\event\system\net\output\o_sensorSecundario.com

2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>t_derecha02018-05-28-11-00-40-856.com;
url=C:\UnidadD\dbneurona\network\event\system\net\threshold\t_derecha02018-05-28-11-00-40-856.com

2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>t_izquierda02018-05-28-11-00-13-415.com;

url=C:\UnidadD\dbneurona\network\event\system\net\threshold\t_izquierda02018-05- 28-11-00-13-415.com
 2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>t_sensorPrimario.com;
 url=C:\UnidadD\dbneurona\network\event\system\net\threshold\t_sensorPrimario.com
 2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>t_sensorSecundario.com;
 url=C:\UnidadD\dbneurona\network\event\system\net\threshold\t_sensorSecundario.com
 m
 2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>>w_derecha02018-05-28-11-00-40-856.com;
 url=C:\UnidadD\dbneurona\network\event\system\net\weight\w_derecha02018-05-28-11-00-40-856.com
 2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>>w_izquierda02018-05-28-11-00-13-415.com;
 url=C:\UnidadD\dbneurona\network\event\system\net\weight\w_izquierda02018-05-28-11-00-13-415.com
 2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>>w_sensorPrimario.com;
 url=C:\UnidadD\dbneurona\network\event\system\net\weight\w_sensorPrimario.com
 2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>>w_sensorSecundario.com;
 url=C:\UnidadD\dbneurona\network\event\system\net\weight\w_sensorSecundario.com
 2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>system_arm.com;
 url=C:\UnidadD\dbneurona\network\event\system\system_arm.com
 2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>system_leg.com;
 url=C:\UnidadD\dbneurona\network\event\system\system_leg.com
 2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>system_listen.com;
 url=C:\UnidadD\dbneurona\network\event\system\system_listen.com
 2018-10-10 22:48:37 DEBUG ManagerFile:69 - Fichero==>network_humano.com;
 url=C:\UnidadD\dbneurona\network\network_humano.com
 2018-10-10 22:48:37 DEBUG Arbol:101 -
 Url=C:\UnidadD\dbneurona\network\network_humano.com 2018-10-10 22:48:37 DEBUG Arbol:101 -
 Url=C:\UnidadD\dbneurona\network\event\event_movement_arm.com
 2018-10-10 22:48:37 DEBUG Arbol:132 - ID====>movement_arm
 2018-10-10 22:48:37 DEBUG Arbol:101 -
 Url=C:\UnidadD\dbneurona\network\event\system\system_arm.com 2018-10-10 22:48:37 DEBUG Arbol:132 - ID====>system_arm

ANEXO 4

Ejecución en producción de red neuronal de un patrón de sonido de 12 entradas. 2018-05-

08 21:26:54 DEBUG Core:53 - =====CAPA=====1
 2018-05-08 21:26:54 DEBUG Core:58 - columna=0 fila=0
 2018-05-08 21:26:54 DEBUG Core:59 - entrada:2.933293311231783 * peso:5.5528 = salida:16.287991098607847
 2018-05-08 21:26:54 DEBUG Core:58 - columna=1 fila=0
 2018-05-08 21:26:54 DEBUG Core:59 - entrada:0.7910251009150709 * peso:-4.8384 = salida:12.460695250340368

2018-05-08 21:26:54 DEBUG Core:58 - columna=2 fila=0
 2018-05-08 21:26:54 DEBUG Core:59 - entrada:9.97121845170498 * peso:6.2466 = salida:74.7469084307607
 2018-05-08 21:26:54 DEBUG Core:58 - columna=3 fila=0
 2018-05-08 21:26:54 DEBUG Core:59 - entrada:7.569186021720711 * peso:-13.3734 = salida:-26.47884391211906
 2018-05-08 21:26:54 DEBUG Core:58 - columna=4 fila=0
 2018-05-08 21:26:54 DEBUG Core:59 - entrada:6.787793361491363 * peso:-19.0424 = salida:-155.7347202189822
 2018-05-08 21:26:54 DEBUG Core:58 - columna=5 fila=0
 2018-05-08 21:26:54 DEBUG Core:59 - entrada:5.117052212897864 * peso:-3.3936 = salida:-173.0999486086724
 2018-05-08 21:26:54 DEBUG Core:58 - columna=6 fila=0
 2018-05-08 21:26:54 DEBUG Core:59 - entrada:1.6473752977875957 * peso:-6.6294 = salida:-184.0210584078255
 2018-05-08 21:26:54 DEBUG Core:58 - columna=7 fila=0
 2018-05-08 21:26:54 DEBUG Core:59 - entrada:1.4291999504154138 * peso:-7.9038 = salida:-195.31716897591886
 2018-05-08 21:26:54 DEBUG Core:58 - columna=8 fila=0
 2018-05-08 21:26:54 DEBUG Core:59 - entrada:11.39146170066378 * peso:0.3234 = salida:-191.6331702619242
 2018-05-08 21:26:54 DEBUG Core:58 - columna=9 fila=0
 2018-05-08 21:26:54 DEBUG Core:59 - entrada:7.323455465651588 * peso:16.7882 = salida:-68.6855352134722
 2018-05-08 21:26:54 DEBUG Core:58 - columna=10 fila=0
 2018-05-08 21:26:54 DEBUG Core:59 - entrada:9.156415284213669 * peso:-17.3324 = salida:-227.3881874855772
 2018-05-08 21:26:54 DEBUG Core:58 - columna=11 fila=0
 2018-05-08 21:26:54 DEBUG Core:59 - entrada:8.234094668376793 * peso:2.2996 = salida:-208.45306338617792
 2018-05-08 21:26:54 DEBUG Core:65 - columna=0 fila=0
 2018-05-08 21:26:54 DEBUG Core:66 - entrada:-1.0 * umbral:-1340.3013 = salida:-1341.3013
 2018-05-08 21:26:54 DEBUG Core:67 - aplica funcion de activacion TanSig:-1.0 2018-05-08 21:26:54 DEBUG Core:53 - =====CAPA=====2
 2018-05-08 21:26:54 DEBUG Core:58 - columna=0 fila=0
 2018-05-08 21:26:54 DEBUG Core:59 - entrada:-1.0 * peso:33.8494 = salida:-33.8494
 2018-05-08 21:26:54 DEBUG Core:71 - columna=0 fila=0
 2018-05-08 21:26:54 DEBUG Core:72 - entrada:1.0 * umbral:34.8498 = salida:1.0
2018-05-08 21:26:54 DEBUG Core:74 - RESULTADO FINAL=1.0 (OK)

Correlación de Pearson, verificación de correlación en producción de un patrón capturado con un patrón muestra.

2018-05-08 22:20:41 DEBUG CorelacionPearson:152 - X:14.425406636956714
 Y:11.887153833545915
 2018-05-08 22:20:41 DEBUG CorelacionPearson:152 - X:7.297709342709723
 Y:8.715420008198919

2018-05-08 22:20:41 DEBUG CorelacionPearson:152 - X:5.11997137526848
Y:2.875286933354046

2018-05-08 22:20:41 DEBUG CorelacionPearson:152 - X:1.8311072895030545
Y:0.40814814609608874

2018-05-08 22:20:41 DEBUG CorelacionPearson:152 - X:-5.773104974806091
Y:0.21093969514879776

2018-05-08 22:20:41 DEBUG CorelacionPearson:152 - X:0.596029142020279 Y:-
3.042813575601854

2018-05-08 22:20:41 DEBUG CorelacionPearson:152 - X:-5.661522892178421 Y:-
8.825686325818223

2018-05-08 22:20:41 DEBUG CorelacionPearson:152 - X:-14.903901099294565 Y:-
8.242063712265585

2018-05-08 22:20:41 DEBUG CorelacionPearson:152 - X:-8.006366267294045 Y:-
3.9332974668544645

2018-05-08 22:20:41 DEBUG CorelacionPearson:152 - X:-1.5423956799249932
Y:2.464277812026172

2018-05-08 22:20:41 DEBUG CorelacionPearson:152 - X:-3.1969237484506907
Y:1.4422590224157967

2018-05-08 22:20:41 DEBUG CorelacionPearson:152 - X:-1.5238251721814609 Y:-
2.081713526899553

2018-05-08 22:20:41 DEBUG CorelacionPearson:157 - sumaX:-11.337816047672014
sumaY:1.8779108433460583

2018-05-08 22:20:41 DEBUG CorelacionPearson:158 - promedioX:-
0.9448180039726678 promedioY:0.1564925702788382

2018-05-08 22:20:41 DEBUG CorelacionPearson:167 - x:14.425406636956714
y:11.887153833545915 x-promedoX(zX):15.370224640929381 y-
promedioY(zY):11.730661263267077 zX*zX:236.24380551263272
zY*zY:137.60841367351475 zY*zX=180.3028988030634

2018-05-08 22:20:41 DEBUG CorelacionPearson:167 - x:7.297709342709723
y:8.715420008198919 x-promedoX(zX):8.24252734668239 y-
promedioY(zY):8.558927437920081 zX*zX:304.18306257343977
zY*zY:210.86365256109593 zY*zX=250.85009226838991

2018-05-08 22:20:41 DEBUG CorelacionPearson:167 - x:5.11997137526848
y:2.875286933354046 x-promedoX(zX):6.064789379241148 y-
promedioY(zY):2.718794363075208 zX*zX:340.964732787996
zY*zY:218.25549534978546 zY*zX=267.33900744590915

2018-05-08 22:20:41 DEBUG CorelacionPearson:167 - x:1.8311072895030545
y:0.40814814609608874 x-promedoX(zX):2.7759252934757224 y-
promedioY(zY):0.2516555758172505 zX*zX:348.6704940229543
zY*zY:218.31882587862538 zY*zX=268.03758452406447

2018-05-08 22:20:41 DEBUG CorelacionPearson:167 - x:-5.773104974806091
y:0.21093969514879776 x-promedoX(zX):-4.828286970833423 y-
promedioY(zY):0.05444712486995956 zX*zX:371.9828490956741
zY*zY:218.32179036803197 zY*zX=267.7746981804555

2018-05-08 22:20:41 DEBUG CorelacionPearson:167 - x:0.596029142020279 y:-
3.042813575601854 x-promedoX(zX):1.540847145992947 y-promedioY(zY):-
3.199306145880692 zX*zX:374.3570590229887 zY*zY:228.55735018310193
zY*zX=262.84505643641756

2018-05-08 22:20:41 DEBUG CorelacionPearson:167 - x:-5.661522892178421 y:-
8.825686325818223 x-promedoX(zX):-4.7167048882057525 y-promedioY(zY):-

8.98217889609706 zX*zX:396.60436402541274 zY*zY:309.23688790459335
zY*zX=305.2113435423771
2018-05-08 22:20:41 DEBUG CorelacionPearson:167 - x:-14.903901099294565 y:-
8.242063712265585 x-promedoX(zX):-13.959083095321898 y-promedioY(zY):-
8.398556282544423 zX*zX:591.4603648875143 zY*zY:379.77263553565973
zY*zX=422.44748857115246
2018-05-08 22:20:41 DEBUG CorelacionPearson:167 - x:-8.006366267294045 y:-
3.9332974668544645 x-promedoX(zX):-7.061548263321376 y-promedioY(zY):-
4.0897900371333025 zX*zX:641.3258287627315 zY*zY:396.49901808349455
zY*zX=451.3277383052202
2018-05-08 22:20:41 DEBUG CorelacionPearson:167 - x:-1.5423956799249932
y:2.464277812026172 x-promedoX(zX):-0.5975776759523254 y-
promedioY(zY):2.307785241747334 zX*zX:641.6829278415281
zY*zY:401.82489080552136 zY*zX=449.94865736385975
2018-05-08 22:20:41 DEBUG CorelacionPearson:167 - x:-3.1969237484506907
y:1.4422590224157967 x-promedoX(zX):-2.252105744478023 y-
promedioY(zY):1.2857664521369585 zX*zX:646.754908125839
zY*zY:403.47808617496224 zY*zX=447.05297535094496
2018-05-08 22:20:41 DEBUG CorelacionPearson:167 - x:-1.5238251721814609 y:-
2.081713526899553 x-promedoX(zX):-0.579007168208793 y-promedioY(zY):-
2.238206097178391 zX*zX:647.0901574266762 zY*zY:408.48765270840875
zY*zX=448.3489127251399
**2018-05-08 22:20:41 DEBUG CorelacionPearson:182 - Resultado correlacion de
Pearson:0.8720557641071979**

MATRIZ DE CONSISTENCIA

TÍTULO: Sistema de Inteligencia Artificial para el Control de Androides Autónomos						
TIPO DE INVESTIGACIÓN: Investigación Tecnológica		NIVEL DE INVESTIGACIÓN: Explicativo		DISEÑO DE LA INVESTIGACIÓN: Experimental		
PROBLEMA	OBJETIVOS	HIPÓTESIS	VARIABLES E INDICADORES			
<p>GENERAL</p> <p>¿De qué manera el Sistema de Inteligencia Artificial emula el control de Androides autónomos?</p> <p>ESPECIFICOS</p> <ul style="list-style-type: none"> • ¿De qué manera el Sistema de Inteligencia Artificial captura e identifica patrones de sonido? • ¿De qué manera el Sistema de Inteligencia Artificial emula el razonamiento basado en conocimiento de hechos? 	<p>GENERAL</p> <p>Emular mediante el Sistema de Inteligencia Artificial el control de Androides autónomos.</p> <p>ESPECIFICOS</p> <ul style="list-style-type: none"> • Capturar e identificar patrones de sonido mediante el Sistema de Inteligencia Artificial. • Emular el razonamiento basado en conocimiento de hechos mediante el Sistema de Inteligencia Artificial. 	<p>GENERAL</p> <p>Si se aplica el Sistema de Inteligencia Artificial entonces permitirá emular el control de Androides autónomos.</p> <p>ESPECIFICOS</p> <ul style="list-style-type: none"> • Si se aplica el Sistema de Inteligencia Artificial entonces permitirá capturar e identificar patrones de sonido. • Si se aplica el Sistema de Inteligencia Artificial entonces permitirá emular el razonamiento basado en conocimiento de hechos. 	VARIABLE	DIMENSIÓN	INDICADOR	Unidad de Medida
			INDEPENDIENTE	Algoritmos de reconocimiento de voz.	- Algoritmo de parametrización (MFCC).	Número de coeficiente resultante
			Sistema de Inteligencia Artificial	Algoritmo de Aprendizaje supervisado de entrenamiento de redes neuronales.	- Red Neuronal Backpropagación.	Propagación de error
			DEPENDIENTE	Captura e identificación de patrones de sonido	- Porcentaje de reconocimiento de patrones	Número de intentos satisfactorio
Control de Androides Autónomos.	Emular el razonamiento basado en conocimiento de hechos.	- Porcentaje de razonamiento de movimientos	Número de intentos satisfactorios			
			TIPO Y NIVEL DE INVESTIGACIÓN		DISEÑO Y MÉTODO DE LA INVESTIGACIÓN	
			Tipo : Investigación Tecnológica Nivel : Explicativo		Diseño: Experimental Método: Deductivo – Inductivo	