



Campus Irapuato-Salamanca

División de
Ingenierías

Asunto: *Notificación de Dictamen de Modalidad de Titulación*

C. Leticia Damián Ramírez
Estudiante de la Maestría en Administración de Tecnologías
PRESENTE.

Una vez concluido el análisis y discusión sobre la propuesta del proyecto de tesis titulada: **“Optimización de rutas de mantenimiento. Método del ahorro y desarrollo de una red neuronal en base radial”** bajo la revisión del *Asesor(es): BAEZA SERRATO Roberto, Dr.*, en su tercera reunión extraordinaria del 12 de agosto del 2019, el comité académico de la Maestría en Administración de Tecnologías acordó la siguiente resolución:

Con fundamento en las fracciones primera a cuarta del Artículo 67 y el Artículo 68 del Estatuto Académico 2008 y el Artículo 78 del Reglamento Académico 2019, este comité acordó **POR UNANIMIDAD DE VOTOS ACEPTAR SU PROPUESTA DE TESIS** designando el siguiente jurado:

Presidente: CABAL YEPEZ Eduardo, Dr.
Secretario: BAEZA SERRATO Roberto, Dr.
Vocal: CARCAMO SOLIS María de Lourdes, Dra.

Se extiende la presente notificación de la resolución el 19 de agosto del 2019.

ATENTAMENTE
“LA VERDAD OS HARÁ LIBRES”
DIRECTORA SUPLENTE


Dra. ROCIO ALFONSINA LIZARRAGA MORALES

c.c.p. Secretaría Académica de la División de Ingenierías
BAEZA SERRATO Roberto Dr. – Asesor(es)

DIVISIÓN DE INGENIERÍAS – CAMPUS IRAPUATO-SALAMANCA
Domicilio Conocido, Comunidad de Palo Blanco, Carretera Salamanca-Valle de Santiago km 3.5 +1.8
Salamanca, Gto., C.P. 36885, Tel(464) 647-9940, FAX ext. 2311



Yuriria, Gto., a 11 de enero de 2021

Asunto: Solicitud cambio de sinodal e integración de codirector en tesis de la maestría en administración de tecnologías MAT.

DR. JOSÉ AMPARO ANDRADE LUCIO
DIRECTOR DE LA DIVISIÓN DE INGENIERÍAS
CAMPUS IRAPUATO-SALAMANCA

PRESENTE

El que suscribe Dr. Roberto Baeza Serrato director de la tesis titulada "OPTIMIZACIÓN DE RUTAS DE MANTENIMIENTO: MÉTODO DEL AHORRO Y DESARROLLO DE UNA RED NEURONAL EN BASE RADIAL", desarrollada por la alumna Leticia Damián Ramírez, me permito solicitarle de la manera más atenta su autorización para un cambio de sinodal, por un investigador más a fin al tema desarrollado en la tesis, donde se utiliza una herramienta de inteligencia artificial para la distribución de rutas.

Se propone al Dr. Luis Manuel Ledesma Carrillo como sinodal en sustitución de la Dra. María de Lourdes Cárcamo Solís.


También hago la solicitud de integrar al Dr. Everardo Vargas Rodríguez como codirector de la tesis.

De antemano agradezco el apoyo a la presente solicitud deseándole un excelente año 2021.

ATENTAMENTE


Dr. Roberto Baeza Serrato

Profesor investigador DEM Yuriria.


Vo. Bo. Dr. José Amparo Andrade Lucio
18 enero de 2021

Llenar en computadora con ayuda del oficio de modalidad.

La modalidad de tesis es única para los posgrados

Nivel:	
Licenciatura	
Maestría	X
Doctorado	

Modalidad:	Tesis
-------------------	-------

Año:	2021
-------------	------

Marcar con una X

Poner el número de año p.e. 2015

Información sobre Obtención de Grado Académico:

Nombre	Leticia Damián Ramírez
NUA	143479
Programa	Maestría Administración de Tecnologías




Para modalidades con Jurado completar la siguiente información:

Lugar, hora y fecha de la presentación


Lugar	Microsoft Teams	Firma y sello de autorización de reservación de lugar.
Hora	12 pm	
Fecha	29 de enero de 2021	

Título del trabajo	Optimización de rutas de mantenimiento: método del ahorro y desarrollo de una red neuronal en base radial.
---------------------------	--

Jurado

	Nombre con grado académico completo: p.e. Doctor en Informática Industrial Nombre Apellido Paterno Apellido Materno	Firma de autorización para realización de examen de grado o titulación
Presidente	Doctor of philosophy Eduardo Cabal Yépez	
Secretario	Doctor en Ciencia y Tecnología con Especialidad en Ingeniería Industrial y Manufactura. Roberto Baeza Serrato	
Vocal (1)	Doctor en ingeniería Eléctrica Luis Manuel Ledesma Carrillo	
Vocal 2 (Doctorado)		
Vocal 3 (Doctorado)		

Asesoría

Director del trabajo	Dr. Roberto Baeza Serrato	
Codirector	Dr. Everardo Vargas Rodríguez	

(No llenar para uso exclusivo de la Coordinación.)

Valida (nombre y firma): _____

Una vez terminado de llenar imprimir en dos tantos (uno para entregar al iniciar el trámite de autorización del examen de grado o titulación y otro para firma de recibido).


Salamanca, Gto, a 18 de Enero del 2021.


M. en I. HERIBERTO GUTIÉRREZ MARTÍN
JEFE DE LA UNIDAD DE ADMINISTRACIÓN ESCOLAR
PRESENTE.-

Por medio de la presente, se otorga autorización para proceder a los trámites de impresión, empastado de tesis y titulación al alumno(a) Leticia Damián Barrios del Programa de Maestría en Administración de Tecnologías y cuyo número de NUA es: 143479 del cual soy director. El título de la tesis es: Optimización de rutas de mantenimiento: método del ahorro y desarrollo de una red neuronal en base radial

Hago constar que he revisado dicho trabajo y he tenido comunicación con los sinodales asignados para la revisión de la tesis, por lo que no hay impedimento alguno para fijar la fecha de examen de titulación.

ATENTAMENTE


Dr. Roberto Boza Serrato
NOMBRE Y FIRMA
DIRECTOR DE TESIS
SECRETARIO


Dr. Everardo Vargas Rodríguez
NOMBRE Y FIRMA
DIRECTOR DE TESIS


Dr. Eduardo Cebal Yapez
NOMBRE Y FIRMA
PRESIDENTE


Dr. Luis Manuel Ledesma Carrillo
NOMBRE Y FIRMA
VOCAL



Universidad
de Guanajuato

UNIVERSIDAD DE GUANAJUATO
DIVISIÓN DE INGENIERÍAS
CAMPUS IRAPUATO-SALAMANCA

“OPTIMIZACIÓN DE RUTAS DE MANTENIMIENTO:
MÉTODO DEL AHORRO Y DESARROLLO DE UNA RED
NEURONAL EN BASE RADIAL”

TESIS PROFESIONAL

PARA OBTENER EL TÍTULO DE:
MAESTRÍA EN ADMINISTRACIÓN DE TECNOLOGÍAS

PRESENTA

LETICIA DAMIÁN RAMÍREZ

ASESOR

DR. ROBERTO BAEZA SERRATO

CO-ASESOR

DR. EVERARDO VARGAS RODRÍGUEZ

SINODALES

DR. EDUARDO CABAL YÉPEZ
DR. LUIS MANUEL LEDESMA CARRILLO

YURIRIA, GUANAJUATO

ENERO 2021

Dedicatoria

Dedico este trabajo de investigación a mis padres, hermanos y familia por la entrega total conmigo y estar ahí en cada peldaño de mi vida. Sin dejar atrás a las personas que creyeron en mí, me ayudaron y motivaron a seguir creciendo. Gracias

Agradecimientos

Agradezco a Dios por permitirme seguir formándome profesionalmente acompañada de mi familia y seres queridos.

Agradezco al CONACYT por la beca otorgada durante la realización de mi maestría.

Agradezco a la Universidad de Guanajuato, mi institución educativa que me ha respaldado en mi educación con una mención especial al Departamento de Estudios Multidisciplinarios sede Yuriria.

Agradezco a mi asesor de tesis, y profesores que hacen de sus alumnos formar grandes profesionistas.

Agradezco a mis entrañables amigos y compañeros de esta increíble experiencia.

ÍNDICE

CAPÍTULO 1: INTRODUCCIÓN	1
1.1 ANTECEDENTES DEL PROGRAMA DE REDUCCIÓN DE BRECHA DIGITAL	2
1.1.1 MODELO DE OPERACIÓN	5
1.1.2 ALCANCE DEL PROGRAMA	6
1.2 PLANTEAMIENTO DEL PROBLEMA	13
1.3 JUSTIFICACIÓN.....	13
1.4 OBJETIVO GENERAL	15
1.4.1 OBJETIVOS ESPECÍFICOS	15
1.5 HIPÓTESIS GENERAL.....	15
CAPÍTULO 2: MARCO TEÓRICO	16
2.1 REDES NEURONALES ARTIFICIALES (RNA).....	16
2.1.1 PROPIEDADES DE LAS REDES NEURONALES.....	17
2.2 EL CEREBRO HUMANO.....	18
2.2.1 LA NEURONA BIOLÓGICA.....	18
2.2.2 LA NEURONA ARTIFICIAL	20
2.3 ELEMENTOS BÁSICOS QUE COMPONEN UNA RED NEURONAL.....	26
2.4 ARQUITECTURA DE REDES	27
2.4.1 REDES DE UNA CAPA	27
2.4.2 REDES MULTICAPA	29
2.4.3 REDES RECURRENTEES	32
2.5 MECANISMO DE APRENDIZAJE	32
2.5.1 APRENDIZAJE SUPERVISADO	33
2.5.2 APRENDIZAJE NO SUPERVISADO.....	33
2.6 REDES DE NEURONAS DE BASE RADIAL	34
2.6.1 ARQUITECTURA DE LAS REDES DE BASE RADIAL.....	35
2.6.3 ENTRENAMIENTO DE LA RED.....	40
2.7 DIFERENCIAS ENTRE LAS REDES NEURONALES TIPO MLP Y RBF	44
2.8 MÉTODO DEL CODO (ELBOW METHOD).....	45
CAPÍTULO 3: ESTADO DEL ARTE	46
3.1 IMPLEMENTACIONES DE RED NEURONAL Y/O ALGORITMO K-MEDIAS	46
CAPÍTULO 4: METODOLOGÍA	66
CAPÍTULO 5: DESARROLLO DE LA METODOLOGÍA	67

5.1	OBTENCIÓN DE DATOS	67
5.2	NORMALIZACIÓN DE DATOS	68
5.3	ENTRENAMIENTO DE LA CAPA OCULTA DE LA RED NEURONAL DE BASE RADIAL	78
5.4	ENTRENAMIENTO DE LA CAPA DE SALIDA DE LA RED NEURONAL DE BASE RADIAL	132
5.5	RESULTADOS DEL COMPORTAMIENTO DE LA RED NEURONAL DE BASE RADIAL	143
5.5.1	PRUEBAS REALIZADAS CON N CENTROIDES	144
	CONCLUSIONES	173
	REFERENCIAS	174
	APÉNDICE	178

ÍNDICE DE TABLAS

Tabla 1. Ubicación geográfica de los nodos públicos del PRBD.....	7
Tabla 2. Ubicación geográfica de los nodos escolares del PRBD.....	10
Tabla 3. Artículos científicos	46
Tabla 4. Abscisas	69
Tabla 5. Ordenadas.....	73
Tabla 6. Núm. de centroides vs WCSS.....	79
Tabla 7 Datos de entrada	81
Tabla 8 Inicializar algoritmo k-medias con siete centros.....	86
Tabla 9 Segunda iteración con nuevos centroides	91
Tabla 10 Tercera iteración con nuevos centroides.....	96
Tabla 11 Cuarta iteración con nuevos centroides	100
Tabla 12 Quinta iteración con nuevos centroides	105
Tabla 13 Sexta iteración con nuevos centroides	109
Tabla 14 Séptima iteración con nuevos centroides	113
Tabla 15 Centroides finales	118
Tabla 16 Octava iteración con centroides finales	118
Tabla 17 Validación de distancias	123
Tabla 18 Clasificación de los parámetros	127
Tabla 19 Neuronas de la red de base radial	133
Tabla 20 Capa de salida de la red neuronal de base radial.....	139
Tabla 21 Clasificación general propuesta.....	162
Tabla 22 Subclasificación de grupos	170
Tabla 23. Porcentaje de rendimiento en Matlab	172

ÍNDICE DE FIGURAS

Figura 1 Línea del tiempo del Programa de Reducción de Brecha Digital	3
Figura 2 Ubicación de los nodos públicos y escolares.....	5
Figura 3 Modelo de operación del Programa de Reducción de Brecha Digital	6
Figura 4 Representación del diagrama de bloques del sistema nervioso	18
Figura 5 Elementos neuronales.....	19
Figura 6 Modelo de Neurona Artificial	21
Figura 7 Función de activación escalón.....	23
Figura 8 Función de actividad lineal y mixta	24
Figura 9 Función tangente hiperbólica	24
Figura 10 Función sigmoideal.....	25
Figura 11 Función de activación gaussiana	25
Figura 12 Ejemplo de una red neuronal totalmente conectada	26
Figura 13 Capa de S Neuronas	27
Figura 14 Capa de S Neuronas, Notación Abreviada	28
Figura 15 Red de tres capas.....	30
Figura 16 Notación abreviada, red de tres capas	31
Figura 17 Arquitectura de la red de neuronas de base radial	36
Figura 18 Funciones de base radial	38
Figura 19 Elemento de procesamiento de una RFB.....	39
Figura 20 Arquitectura de una red RBF	40
Figura 21 Algoritmo K-medias	42
Figura 22 Red Neuronal de Base Radial.....	44
Figura 23 Diagrama de la metodología	66
Figura 24 Distribución de datos	67
Figura 25 Diagrama de dispersión de los datos de entrada	78
Figura 26 Algoritmo del codo o elbow	79
Figura 27 Agrupación en siete centroides	80
Figura 28 Clasificación red neuronal de base radial	131
Figura 29 Diagrama de la red de base radial con siete centros	132
Figura 30 Entrenamiento de la red con siete centros	143
Figura 31 Entrenamiento RNA de base radial con 3 centros	144
Figura 32 Validación RNA de base radial con 3 centros	145
Figura 33 Pruebas RNA de base radial con 3 centros	145
Figura 34 Entrenamiento RNA de base radial con 4 centros	146
Figura 35 Validación RNA de base radial con 4 centros	147
Figura 36 Pruebas RNA de base radial con 4 centros	147
Figura 37 Entrenamiento RNA de base radial con 5 centros	148
Figura 38 Validación RNA de base radial con 5 centros	149
Figura 39 Pruebas RNA de base radial con 5 centros	149
Figura 40 Entrenamiento RNA de base radial con 6 centros	150
Figura 41 Validación RNA de base radial con 6 centros	151

Figura 42 Pruebas RNA de base radial con 6 centros.....	151
Figura 43 Entrenamiento RNA de base radial con 7 centros	152
Figura 44 Validación RNA de base radial con 7 centros	153
Figura 45 Pruebas RNA de base radial con 7 centros.....	153
Figura 46 Entrenamiento RNA de base radial con 8 centros	154
Figura 47 Validación RNA de base radial con 8 centros	155
Figura 48 Pruebas RNA de base radial con 8 centros.....	155
Figura 49 Entrenamiento RNA de base radial con 9 centros	156
Figura 50 Validación RNA de base radial con 9 centros	157
Figura 51 Pruebas RNA de base radial con 10 centros.....	157
Figura 52 Entrenamiento RNA de base radial con 10 centros	158
Figura 53 Validación RNA de base radial con 10 centros	159
Figura 54 Pruebas RNA de base radial con 10 centros.....	159
Figura 55 Entrenamiento RNA de base radial con 20 centros	160
Figura 56 Validación RNA de base radial con 20 centros.....	161
Figura 57 Pruebas RNA de base radial con 20 centros.....	161
Figura 58 Clasificación grafica general	168
Figura 59 Subclasificación de grupos.....	169

CAPÍTULO 1: INTRODUCCIÓN

A medida que la tecnología avanza, se utilizan aplicaciones de redes neuronales artificiales en la vida diaria, desde escuchar la música favorita de acuerdo con el historial guardado en tu dispositivo móvil hasta mostrar la ruta más corta para llegar a un lugar. Las redes neuronales artificiales (RNA) surgen como un intento para emular el funcionamiento de las neuronas de nuestro cerebro. (Caicedo Bravo y López Sotelo, 2009). Las redes neuronales son más que otra forma de emular ciertas características propias de los humanos, como la capacidad de memorizar y de asociar hechos. Si se examinan con atención aquellos problemas que no pueden expresarse a través de un algoritmo, se observará que todos ellos tienen una característica en común: la experiencia. El hombre es capaz de resolver estas situaciones acudiendo a la experiencia acumulada. Así, parece claro que una forma de aproximarse al problema consista en la construcción de sistemas que sean capaces de reproducir esta característica humana. En definitiva, las redes neuronales no son más que un modelo artificial y simplificado del cerebro humano, que es el ejemplo más perfecto del que disponemos para un sistema que es capaz de adquirir conocimiento a través de la experiencia. Una red neuronal es “un nuevo sistema para el tratamiento de la información, cuya unidad básica de procesamiento está inspirada en la célula fundamental del sistema nervioso humano: la neurona”. (Matich, 2001).

En la presente investigación se desarrolla una red neuronal de base radial para clasificar nodos públicos y escolares con acceso a internet, pertenecientes al Programa de Reducción de Brecha Digital (PBRD). Las redes de neuronas de base radial en inglés Radial Basic Functions (RBF) han sido aplicadas a una gran variedad de problemas, aunque es necesario señalar que su aplicación no ha sido tan extendida como el caso de perceptrón multicapa. Sin embargo, se han utilizado en diferentes campos, como análisis de series temporales (Moody and Darken, 1989), (Kadirkamanathan et al., 1991), procesamiento de imágenes (Saha et al., 1991), diagnósticos médicos (Lowe and Webb, 1990), reconocimiento automático del habla (Niranjan and Fallside, 1990), etc. (Isasi Viñuela y Galván León, 2008).

Buscando encontrar las rutas adecuadas para dar mantenimiento o supervisar cada uno de los 168 nodos de la red, distribuidos en la región I norte, la región III, la región V

este, la región VII centro sur, y región VIII Sureste del estado de Guanajuato. Se opta aplicar la función gaussiana, es una función simétrica y su forma es de campana, como tipo de función de base radial calcula la distancia euclidiana de un vector de entrada, es decir las coordenadas de un nodo respecto de un centro, siendo cada neurona de la capa oculta un grupo de nodos al que le corresponde una ruta. Para obtener mejores resultados se aplica el método del codo o elbow y así obtener el número de centros, es decir el número exacto de clústeres.

1.1 ANTECEDENTES DEL PROGRAMA DE REDUCCIÓN DE BRECHA DIGITAL

El Programa de Reducción de Brecha Digital tiene como objetivos generales: cerrar la brecha digital en el estado de Guanajuato mediante el desarrollo de infraestructura de banda ancha estatal con cobertura en regiones urbanas, sub-urbanas y rurales, así como también la promoción de acceso WiFi en espacios públicos y escolares mediante un modelo escalable, sustentable y gestión inteligente. Haciendo énfasis en llevar conectividad a zonas de difícil acceso y/o alto grado de marginación social y económica. (Domínguez, 2018).

Es un programa que promueve la Secretaría de Innovación, Ciencia y Educación Superior (SICES) que tiene como objetivos específicos: acercar el acceso a la información y conocimiento a la sociedad guanajuatense, fomentar el uso de las TIC para contar con una sociedad mejor informada, impulsar nuevos esquemas de divulgación de la ciencia y tecnología, habilitar las políticas de gobierno digital y datos abiertos y desarrollar una red de telecomunicaciones de impulso a la banda ancha. (Victoria-Rodríguez, M.T. y Vega-Corona, A., 2017). Este programa se crea para impulsar los esquemas de divulgación de la ciencia y la tecnología que promuevan una cultura científica desde edades tempranas mediante el uso innovador de las TIC, haciendo énfasis en comunidades de alta y muy alta marginación en el estado, así como aquellas que no cuentan con servicios comerciales de telefonía y/o internet. (Centro Mexicano de Energías Renovables, 2017). Se orienta a incorporar a comunidades rurales de alta y muy alta marginación a un sistema de conectividad y alfabetización digital, aplicando un modelo participativo de apropiación social de la ciencia y la tecnología, a fin de abatir la brecha digital. (Redacción AM, 2016).

En la figura 1 muestra el caso de estudio del programa de reducción de brecha digital en el estado de Guanajuato.

Caso de estudio del Programa de Reducción de Brecha Digital (PRBD)

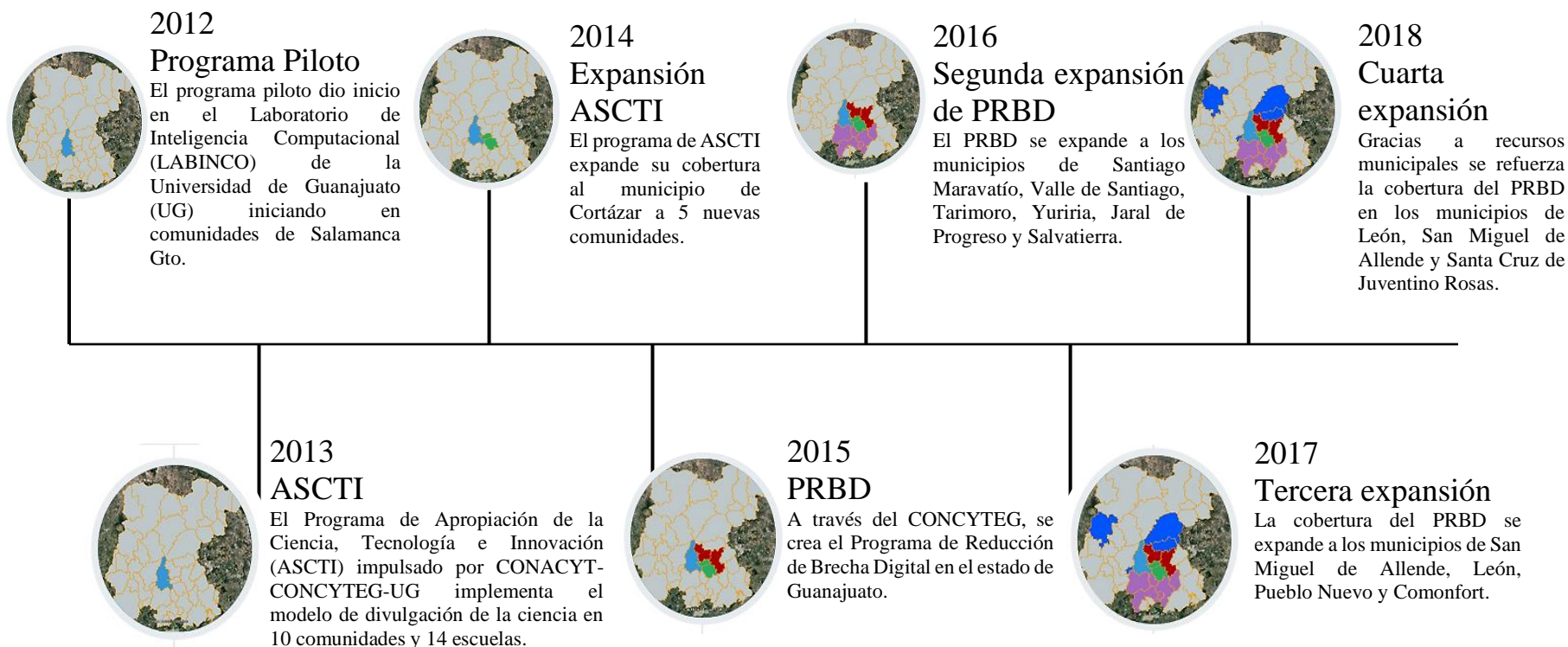


Figura 1 Línea del tiempo del Programa de Reducción de Brecha Digital Universidad de Guanajuato 2019. Recuperado de <http://www.guanajuatoconectado.org>

Nació en el 2012 como prueba piloto, dentro del Laboratorio de Inteligencia Computacional (LABINCO), perteneciente a la Universidad de Guanajuato (UG) en la División Ingenierías Campus Irapuato-Salamanca (DICIS). Implementando pruebas de concepto para la divulgación de la ciencia y la tecnología a través de un modelo de conectividad inalámbrica de banda ancha. En el 2013 obtuvo el nombre de Apropiación Social de la Ciencia, Tecnología e Innovación (ASCTI) impulsado por CONACYT-CONCYTEG-UG implementa el modelo de divulgación de la ciencia en 10 comunidades y 14 escuelas del municipio de Salamanca. La expansión ASCTI se logra en el 2014 con la colaboración del gobierno municipal, abarcando cinco comunidades de Cortázar. A través del CONCYTEG; ahora Secretaria de Innovación, Ciencia y Educación Superior (SICES) en el 2015, se crea el Programa de Reducción de Brecha Digital (PRBD) en el estado de Guanajuato y se amplía su cobertura a los municipios de Santa Cruz de Juventino Rosas, Celaya y Villagrán a través de 14 nodos de conectividad pública y 4 nodos escolares. Surge una segunda expansión de PRBD en el 2016, a los municipios de Santiago Maravatío, Valle de Santiago, Tarimoro, Yuriria, Jaral del Progreso y Salvatierra. Brindando servicio a través de 20 nodos de conectividad pública y 4 nodos de conectividad escolar. El programa sigue escalando y en el 2017 surge una tercera expansión, la cobertura del PRBD se expande a los municipios de San Miguel de Allende, León, Pueblo Nuevo y Comonfort. Con recursos municipales se logró construir 13 nodos de conectividad pública y a través de la SEG se implementan 50 nodos de conectividad escolar. Finalmente, en el 2018 se realizó la cuarta expansión del PRBD, se reforzó la cobertura de los municipios de León, San Miguel de Allende y Santa Cruz de Juventino Rosas a través de 29 nodos públicos. (Programa de Reducción de Brecha Digital, 2019).

Actualmente, maneja un sistema que genera registros de la cantidad de usuarios que se conectan en cada uno de los nodos y la cantidad de datos que se consumen en cada uno de ellos durante las últimas 48 horas, toda esta información se puede visualizar a través del portal de transparencia <http://www.guanajuatoconectado.org/views/index.php>. Para la distribución del servicio de internet en plazas y jardines se hace uso de los nodos de conectividad pública. Los nodos operan de manera autónoma los 365 días de año, las 24 horas del día. Cada uno de estos nodos soporta la conexión simultánea de 30 usuarios a 10 Mbps. por medio de un robusto sistema de ancho de banda, además este sistema permite

establecer una política de uso justo del servicio en la cuál a cada usuario se le asignan sesiones de una hora de conectividad con el fin de evitar una sobre explotación al servicio. El PRBD cuenta con 72 nodos de conectividad escolar y 96 nodos de conectividad pública distribuidos en 15 municipios beneficiados que se muestran en la figura 2. Beneficiando a 18,006 estudiantes beneficiados de todos los niveles educativos. Esfuerzo que se ha llevado a cabo mediante la coordinación de los niveles de gobierno estatal, municipal y federal.

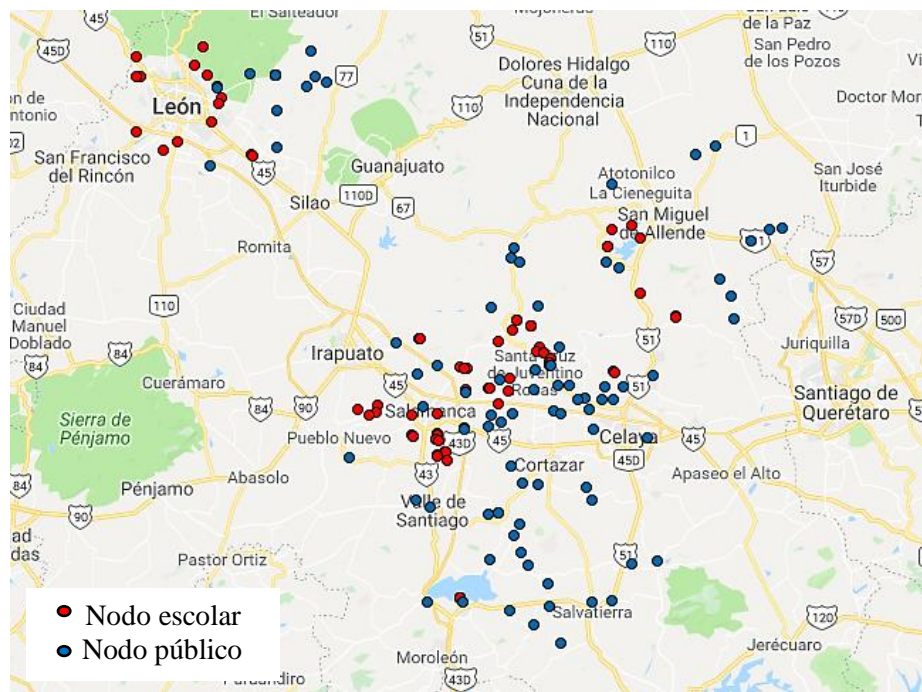


Figura 2 Ubicación de los nodos públicos y escolares

Nota. Recuperado de “Programa de Reducción de Brecha Digital”, Universidad de Guanajuato, 21 de enero, 2020. Recuperado de <http://www.guanajuatoconectado.org>. Copyright 2019 de Universidad de Guanajuato.

1.1.1 MODELO DE OPERACIÓN

El programa trabaja en base a un modelo con tecnología de telecomunicaciones de 4ta generación con células de banda ancha distribuidas estratégicamente en diferentes zonas del estado como se muestra en la figura 2, las cuales tienen áreas de cobertura de hasta 25 km de radio; a estas células son enlazados los nodos de conectividad pública y nodos de conectividad escolar encontrados en la figura 3, que se encuentran en las diferentes zonas de los municipios beneficiados por el programa.

Para la distribución del servicio de internet en plazas y jardines se hace uso de los nodos de conectividad pública, los cuales son un modelo completamente desarrollado por Ingenieros de la Universidad de Guanajuato. Los nodos de conectividad pública operan de manera autónoma gracias a su sistema de alimentación autónoma de energía solar el cual permite una operación completamente ecológica y libre de cables. El servicio de conectividad a las escuelas se brinda a través de los nodos escolares, estos nodos a diferencia de los públicos son mucho más sencillos en su composición e instalación debido al hecho que su función no es proveer internet al público en general sino solo a la institución educativa y además su alimentación es mediante energía eléctrica.

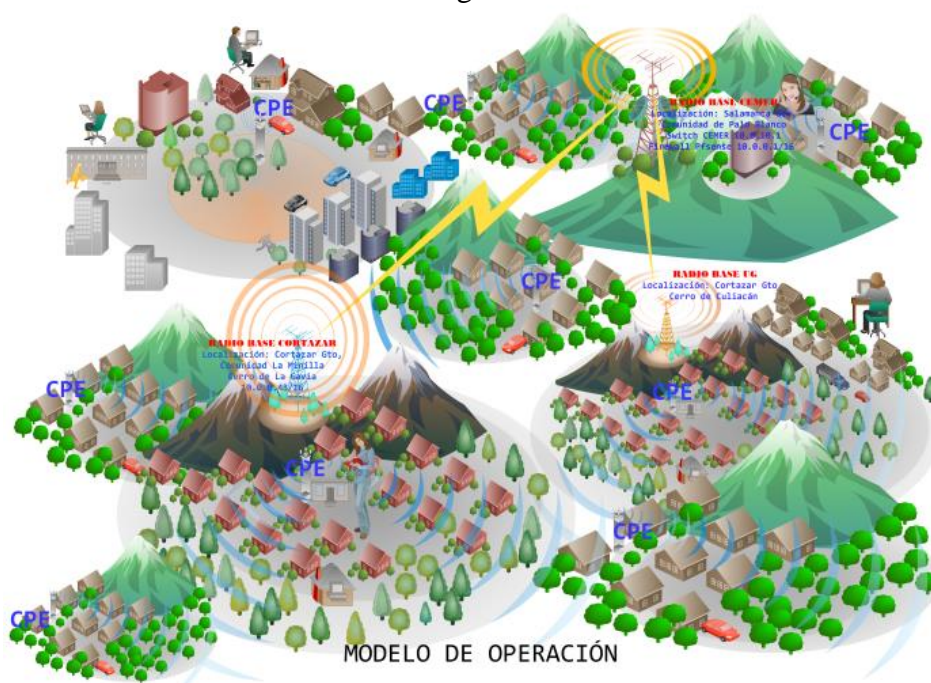


Figura 3 Modelo de operación del Programa de Reducción de Brecha Digital
 Nota. Recuperado de “Reducción de Brecha Digital”, de Victoria-Rodríguez, M.T. y Vega-Corona, A., 2017, Universidad de Guanajuato, 2-12. Copyright 2017 de Universidad de Guanajuato.

1.1.2 ALCANCE DEL PROGRAMA

La tabla 1 muestra la ubicación geográfica de los nodos públicos conectados a la fecha del 22 de enero 2020 de las comunidades pertenecientes a cada municipio del estado de Guanajuato con acceso a internet a través del PRBD.

Tabla 1. Ubicación geográfica de los nodos públicos del PRBD

Municipio	Comunidad	Latitud	Longitud
Celaya	Pedro María / Patria Nueva	20.52794	-100.878154
	Roque	20.581634	-100.838369
	San Elías	20.633491	-100.841508
	San Juan de la Vega	20.624287	-100.760489
	Parque Lineal	20.511533	-100.77846
	Gasca	20.60445	-100.856862
	Yustis	20.581375	-100.864164
	Ejido de Silva	20.566268	-100.836045
	San Cayetano	20.603806	-100.820861
	San isidro de Elguera	20.563705	-100.887103
Cortázar	La Minilla	20.422871	-100.892882
	El Diezmo	20.428459	-101.015925
	Tierra Fría	20.461384	-101.039803
	Galeras	20.426612	-100.98684
	Centro Cultural	20.488899	-100.962013
	La Gavia	20.397251	-100.883689
Jaral de Progreso	Santiago Capitiro	20.304513	-101.020675
	Providencia	20.35469	-101.021035
	Victoria de Cortázar	20.335613	-101.033919
	DIF Municipal	20.372157	-101.081279
	Jardín Municipal	20.37461	-101.063967
Juventino Rosas	Santiago de Cuenda	20.598988	-100.994854
	COMUDE SUR	20.637948	-100.992981
	Valencia	20.644062	-100.963131
	Dulces Nombres	20.650485	-100.964948
	San Diego de los Dolores	20.726128	-101.028434

	Franco Tavera	20.6063	-100.9288
	San Antonio de Romerillo	20.60756	-100.948
	San Antonio de Morales	20.58156	-100.9096
	El Naranjillo	20.714818	-101.000976
	Jaralillo	20.66503	-100.9768
	San Antonio de las Maravillas	20.58287	-100.89469
	Cerrito de Gasca	20.61717	-101.06116
Juventino Rosas	Romero	20.752	-100.98700
	Mesas de Acosta	20.74748	-101.07652
	La Peña	20.6755	-100.94534
	San Francisco del Duran	21.006077	-101.621692
	Vaquerías	21.15663	-101.39523
	Derramadero	21.16696	-101.41692
	San Rafael Cerro Verde	21.19078	-101.49721
	Nuevo Valle de Moreno	21.21237	-101.42442
	Mesas de Obispo	21.14766	-101.43512
León	Las Canelas	21.21906	-101.45616
	San José de Otates Norte	21.19078	-101.49721
	San José de Otates Sur	21.16791	-101.49336
	Sauz Seco	21.17207	-101.54277
	Cuesta Blanca	21.10638	-101.49065
	San José de los Romeros	21.03744	-101.49197
	Los Alisos	21.19096	-101.52930
	Alfaro	21.14697	-101.60730
	Congregación de Cárdenas	20.627029	-101.218901
	Cerro Gordo	20.593413	-101.126121
	Loma de Flores	20.683629	-101.262125
	El 4 de Altamira	20.485611	-101.165822

Salamanca	Valtierrilla	20.52851	-101.129726
	San José de Mendoza	20.690796	-101.215924
	Unión de Liebres	20.642489	-101.182539
	Recuerdo de Ancón	20.638631	-101.129593
	Santo Domingo	20.515328	-101.227903
	Valencia	20.602204	-101.082313
	Ecoparque	20.567796	-101.207279
Salvatierra	El Sabino	20.281045	-101.006595
	Urireo	20.216442	-100.842513
	Maravatío del Encinal	20.205091	-100.965222
	San Nicolás de los Agustinos	20.246335	-100.966111
	Centro Histórico	20.212867	-100.882702
San Miguel de Allende	Los Rodríguez	21.04082	-100.642947
	Cruz del Palmar	20.971912	-100.842675
	Don Francisco	20.905315	-100.918473
	Alonso Yáñez	20.829893	-100.855602
San Miguel de Allende	La Huerta Repetidor	20.81958	-100.83408
	La Huerta	20.821489	-100.829382
	Guadalupe de Tambula	20.86864	-100.57565
	Puerto de Nieto	20.89068	-100.53816
	Moral de Puerto de Nieto	20.89309	-100.51479
	Jalpa	20.76891	-100.61162
	Charco de Sierra	20.72778	-100.6075
	San José de Allendes	20.85651	-101.03344
	San Isidro de Ensaye (La Cueva)	20.83845	-101.03772
	El Nuevo Cimatarío	20.83181	-101.02277
	La Biznaga	21.02538	-100.69224
	Lagunillas	20.80138	-100.64883

Santiago Maravatío	Jardín Municipal	20.172432	-100.995011
Tarimoro	La Moncada	20.283137	-100.805517
	Centro Histórico	20.288112	-100.755605
Valle de Santiago	Santa Barbara	20.475553	-101.349059
	COMUDE	20.637948	-100.992981
Villagrán	Suchitlan	20.554805	-101.036322
	San Isidro Cerro Gordo	20.541822	-101.059277
	San Salvador Torrecillas	20.558234	-100.956586
	Col.18 de marzo	20.554805	-101.036322
	Los Ángeles	20.553862	-100.941347
Yuriria	Loma de Zempoala	20.291165	-101.080675
	Casacuaran	20.19831	-101.042313
	San Francisco de la Cruz	20.213835	-101.20112
	Centro Histórico	20.214227	-101.131645

Nota: Elaboración propia

La tabla 2 muestra la ubicación geográfica de los nodos escolares conectados a la fecha del 22 de enero 2020 de las comunidades pertenecientes a cada municipio del estado de Guanajuato con acceso a internet a través del PRBD.

Tabla 2. Ubicación geográfica de los nodos escolares del PRBD

Municipio	Clave	Comunidad	Escuela	Latitud	Longitud
Celaya	11EJN0861R	San Elías	SABES	20.631236	-100.838563
	11DPR1946W	Jalpilla	Enrique C Rébsamen	20.73049	-100.71975
Comonfort	11DPR1581F	Jalpilla	Benito Juárez	20.7328	-100.72132
	11DPR3873H1	San Isidro de San Jerónimo	Primaria Rafael Ramírez	20.763667	-100.765909
	11DPR1039V1	San Pedro	Mariano Escobedo	20.77533	-100.78924
	11DPR0314C1	El Naranjillo	Niños Héroe	20.715255	-101.000962
	11ETV0446S	El Naranjillo	Dani Ñañhu	20.713992	-101.000079
	11ETV1050P	Los Dulces Nombres	Telesecundaria No.1050	20.654297	-100.964101
	11DPR0322L	La Trinidad	General Lázaro Cárdenas	20.57278	-101.06333

Juventino Rosas	11DPR0724F	San Julián Tierra Blanca	Francisco Villa	20.59556	-101.04333
	11DPR2476B	Jaralillo	Juventino Rosas	20.66556	-100.97611
	11DPR0917U	Emiliano Zapata	Emiliano Zapata	20.61963	-101.04234
	11EPR0873M	Los Dulces Nombres	Gabriela Mistral	20.65555	-100.96188
	11DPR0722H	San Antonio Nuevo Valencia	Ignacio Allende	20.6697	-100.98833
	11DPR2077E	La Tinaja	Niños Héroes	20.70506	-101.03804
	11DPR2480O	San José del Sauz	Felipe Ángeles	20.67611	-100.98429
	11ETV0720H	Jaralillo	Frida Kahlo	20.66542	-100.97585
	11ETV0861G	La Tinaja	Nezahualcóyotl	20.70506	-101.03804
	11DPR3804L	El Carmen de Arriba	Octavio Paz	20.68799	-101.06271
	11ETV0610B	San Diego de los Dolores	Miguel Hidalgo y Costilla	20.726702	-101.031667
	11DPR0720J	San Diego de los Dolores	MTRO. Justo Sierra	20.726265	-101.028494
	11EPO0002C	Valencia	UPJR	20.641806	-100.961227
León	11DPR1893H1	San José del Potrero	Albino García	21.12927	-101.59938
	11DPR1893H	León Urbano	Miguel Hidalgo	21.08529	-101.61801
	11DPR1925J	Los Sauces	Emiliano Zapata	21.02413	-101.53926
	11DPR1900A	Los Naranjos	Tierra y Libertad	21.16928	-101.6267
	11DPR2326V	Alfaro	Insurgentes	21.14869	-101.6081
	11DPR3601Q	León Urbano	5 de mayo	21.11748	-101.60476
	11DPR2715L	Fraccionamiento Paseo de las Torres	Joel Cisneros Lara	21.16689	-101.75621
	11DPR3915Q	La Mora	Melchor Ocampo	21.06612	-101.76323
	11DPR1543C	Ibarrilla	Telesecundaria No. 581	21.18679	-101.64995
	11ETV0585T	Lagunillas	Benito Juárez	21.20174	-101.76388
	11EPR0395C	León Primaria	DR. PABLO DEL RIO	21.16634	-101.76312
	11EPR0193G	Mesa de Ibarrilla	Telesecundaria No. 1003	21.22037	-101.63424
	León	11ETV1003E	Los Sauces	Telesecundaria No. 123	21.0258
11ETV0124J		San Pedro del Monte	Telesecundaria No. 528	21.03328	-101.71161
11ETV0539H		Los Arcos	Telesecundaria No. 527	21.04922	-101.68482
Pueblo Nuevo	11DPR2381O	Congragación de Panales	Telesecundaria No.926	20.56314	-101.33472
	11ETV0926Z	Durazno de Fonseca	Juan Enríquez Pestalozzi	20.55292	-101.31231
		Los Miranda	General Santos Degollado	20.6401	-101.13816
	11EPR0504T	La Luz	Cinco de Febrero	20.55521	-101.18072
	11DPR0111H	Uruétaro	Cinco de Mayo	20.51961	-101.18027
	11DPR0128H	Valencia de Cerro Gordo	Benito Juárez	20.60014	-101.08022

Salamanca	11DPR0847P	El Xoconostle	Año de Juárez	20.63728	-101.12524	
	11DPR2184N	Uruétaro	Telesecundaria No. 149	20.5161	-101.18116	
	11ETV0143Y	La Capilla	Miguel Hidalgo	20.55888	-101.30033	
	11DPR0651D	Colonia el 4 de Altamira	Telesecundaria No. 927	20.4848	-101.16745	
	11ETV0927Z	Colonia el 4 de Altamira	Lázaro Cárdenas	20.48533	-101.16561	
	11DPR0105X	San Juan de Razos	Telesecundaria No. 380	20.55272	-101.2316	
	11ETV0393D	Loma Pelada	Telesecundaria No. 653	20.571437	-101.297963	
	11ETV0672O	San Rafael de Uruétaro	Andrés Delgado	20.50633	-101.17831	
	11DPR0935J	Santo Domingo	Leona Vicario	20.51569	-101.22903	
	11EPR0265J	Santo Domingo	Telesecundaria No. 236	20.51706	-101.23139	
	11ETV0228E	El Xoconostle	Telesecundaria No. 789	20.63725	-101.12582	
	11ETV0789N	Cuatro de Altamira	Preescolar Ignacio José Allende	20.47996	-101.18148	
	11DJN3106Z	Colonia el Cuatro de Altamira	Preescolar Carmen Ramos del Rio	20.48589	-101.16368	
	11DJN0156N	Uruétaro	Preescolar Manuel M. Ponce	20.51808	-101.18081	
	11DJN0037Z	El Recuerdo de Anáñ (Xoconoxtle de Arriba)	Preescolar Hermenegildo Galeana	20.63895	-101.12956	
	11DJN2450D	Cerro Gordo	SABES Cerro Gordo	20.59916	-101.12701	
	11MMS0156N	El Circuito	Margarita Maza de Juárez	20.50918	-101.18305	
	San Miguel de Allende	11DPR0252G	San José de Mendoza	Telesecundaria No. 43	20.690968	-101.215386
			El Circuito	TAYEN	20.50971	-101.1839
		11DPR2993N	Don Juan	Juan Aldama	20.89607	-100.80634
11DPR0624G		Corral de Piedras de Abajo	Miguel Hidalgo	20.84282	-100.82233	
11DPR1984Z		Agustín González	Justo Sierra	20.85922	-100.8531	
Valle de Santiago	11ETV0368E	Agustín González	Telesecundaria No. 546	20.85807	-100.85279	
	11DPR2231H	Pantoja	Simón Bolívar	20.87324	-100.78897	
	11DPR2456O	Tlaxcalilla	Melchor Ocampo	20.88913	-100.84462	
	11MMS0342I	Rancho Cuatro de Altamira	Guadalupe Victoria	20.47803	-101.18115	
	11DPR2279A		UTSOE	20.398449	-101.222193	
Villagrán		Alto de Altamira	General Emiliano Zapata	20.47034	-101.16312	
	11ETV0549O	Parque Agro-Industrial XONOTLI	CECYTEG	20.549669	-101.075751	

Nota: Elaboración propia

1.2 PLANTEAMIENTO DEL PROBLEMA

El programa de reducción de brecha digital permite llevar conectividad a las zonas de difícil acceso o alto grado de marginación social y económica. La cobertura del programa abarca 15 municipios del estado de Guanajuato, donde están distribuidos un total de 168 nodos. En el orden de seguir brindado un servicio de calidad y satisfacer el aumento en la demanda del servicio se debe planificar rutas de mantenimiento, implementando una red neuronal de base radial, que le permita ahorrar tiempo y combustible, eliminando la manera empírica de escoger las rutas.

La manera en la que explico el recorrido a los nodos públicos y escolares es como se realizaba en el tiempo que estuve colaborando en el proyecto, por ejemplo, en el municipio de Salamanca, partiendo de la División de Ingenierías Campus Irapuato-Salamanca ubicada en el cerro de palo blanco, saliendo una unidad con un equipo de cuatro personas, a las comunidades del norte y en los días siguientes a las comunidades del sur. Comenzando el mantenimiento por los nodos públicos, seguido por los nodos escolares cercanos al nodo público, en caso de no estar disponibles la mayoría de los nodos para la verificación o configuración, se regresa al día siguiente. Aproximadamente en dos semanas se deben haber visitado los nodos escolares y públicos, por día se visitan tres o cuatro nodos pertenecientes al municipio de Salamanca.

1.3 JUSTIFICACIÓN

Los programas públicos tienen la necesidad de ser atendidos periódicamente, utilizando herramientas que faciliten la asignación de rutas con la finalidad de ser aprovechados al máximo por los usuarios y seguir cumpliendo el propósito de su fundación. Esta investigación surge como propuesta para reducir costos económicos, logísticos, y operativos planteando trayectorias para la conservación de cada uno de los puntos de la red.

El sistema de transporte es el principal componente de las empresas, es el responsable de mover la materia prima, realizar mantenimiento o hacer una entrega, entre la organización y los clientes que se encuentran ubicados geográficamente en diversos lugares. Sin embargo, es uno de los costos logísticos más altos. Por lo tanto, para una administración efectiva del sistema de transporte es necesaria la aplicación de un sistema de asignación de rutas,

optimizando procesos de mantenimiento cuyo objetivo principal es minimizar tiempos y costos.

La huella es la señal que deja el transporte por donde pasa, esta señal que abandona emisiones de gases de efecto invernadero en el mundo. Los vehículos propiedad de las empresas tienen un alcance directo, teniendo en cuenta la cantidad de puntos que visita un transporte, los procesos que conforman la cadena logística tienen una fuerte incidencia en la huella de carbono. Así mismo las empresas deben modificar sus actividades de forma que sean amigables con el medio ambiente. El transporte es responsable de las emisiones de carbono, razón por la cual se debe pensar en las rutas que hagan menos daño al ecosistema.

Para lograr una optimización, se toma el modelo de datos, se trabaja con la latitud y longitud en una hoja de Excel, ubicando así las direcciones a donde partirán los vehículos, al final se visualizan los puntos agrupados por color de las rutas propuestas. Para armar y optimizar los recorridos se utiliza el algoritmo k-medias. Siendo fundamental el modelo de datos y la aplicación de la tecnología. La incorporación de la tecnología en la logística y transporte permite añadir valor agregado en el diseño y planificación, traduciéndose en la mejora sustancial en la calidad del servicio.

En la parte académica esta investigación puede tomarse como referencia para replicar, modificar y mejorar el modelo basado en una red neuronal de base radial, para cualquier aplicación de rutas de distribución. Esta investigación presenta un trabajo realizado en Excel y Matlab, pero puede trasladarse al área de programación y realizar todo en un mismo lenguaje, quizás utilizando software libre que reduzca aún más los gastos y tenga una interfaz amigable con el usuario.

Finalmente, utilizar los vehículos necesarios ayuda a optimizar los recursos, es necesario proveer la planificación de rutas mediante herramientas tecnológicas. Optimizar rutas es un factor importante para reducir gastos, en lugar de gastar combustible en dar doble vuelta por un mismo punto, escoger la trayectoria más larga, la optimización de rutas mostrara la ruta rentable para cada recorrido, permitiendo que el equipo de trabajo se concentre en otras tareas prioritarias, además de mejorar la productividad general y completando rápidamente el trabajo.

1.4 OBJETIVO GENERAL

Diseñar, desarrollar y proponer una red neuronal de base radial aplicada a la distribución de rutas de mantenimiento a los accesos públicos de internet pertenecientes al Programa de Reducción de Brecha Digital (PRBD) en el Estado de Guanajuato para ejecutar mejores trayectos.

1.4.1 OBJETIVOS ESPECÍFICOS

- Reprocesar cada dirección geográfica del nodo y obtener la ubicación en una coordenada específica, para ser las variables de entrada de la red neuronal de base radial.
- Determinar el número adecuado de clústeres por medio de un método matemático para implementar el algoritmo k-medias.
- Calcular la distancia euclidiana entre el nodo y el centroide de la red para determinar las coordenadas de los centroides finales al no haber variaciones en las últimas iteraciones.

1.5 HIPÓTESIS GENERAL

El modelo de optimización de rutas de mantenimiento aplicando una red neuronal de base radial y métodos matemáticos se pretende obtener un buen entrenamiento de la red.

1.5.1 HIPÓTESIS ESPECIFICAS

- El reprocesamiento de datos obtendrá la misma ubicación que en los datos originales representado en el mapa gráfico.
- Utilizando el método del codo arrojará el número de clústeres que llevara a una clasificación adecuada y obtener un buen entrenamiento sin experimentar con diferentes números aleatoriamente.
- Trabajar con seis decimales los cálculos de la distancia se obtendrá una buena precisión.

CAPÍTULO 2: MARCO TEÓRICO

2.1 REDES NEURONALES ARTIFICIALES (RNA)

El trabajo en redes neuronales artificiales, comúnmente conocidas como "redes neuronales", ha sido motivado desde su inicio por el reconocimiento de que el cerebro humano computa de una manera completamente diferente a la computadora digital convencional.

Un sistema nervioso "en desarrollo" es sinónimo de un cerebro plástico: la plasticidad permite que el sistema nervioso en desarrollo se adapte a su entorno. Así como la plasticidad parece ser esencial para el funcionamiento de las neuronas como unidades de procesamiento de información en el cerebro humano, también lo es con las redes neuronales formadas por neuronas artificiales.

Una red neuronal es un procesador formado por unidades de procesamiento simples que tiene una propensión natural a almacenar el conocimiento experimental y ponerlo a disposición para su uso. Se asemeja al cerebro en dos aspectos:

1. El conocimiento es adquirido por la red desde su entorno a través de un proceso de aprendizaje.
2. Las fuerzas de conexión interneurona, conocidas como pesos sinápticos, se utilizan para almacenar el conocimiento adquirido.

El procedimiento utilizado para realizar el proceso de aprendizaje se denomina algoritmo de aprendizaje, cuya función es modificar los pesos sinápticos de la red de manera ordenada para lograr el objetivo de diseño deseado.

La modificación de los pesos sinápticos proporciona el método tradicional para el diseño de redes neuronales. Tal enfoque es el más cercano a la teoría del filtro adaptativo lineal, que ya está bien establecido y aplicado con éxito en muchos campos diversos (Widrow y Stearns, 1985; Haykin, 2002). Sin embargo, también es posible que una red neuronal modifique su propia topología, que está motivada por el hecho de que las neuronas en el cerebro humano pueden morir y pueden crecer nuevas conexiones sinápticas.

2.1.1 PROPIEDADES DE LAS REDES NEURONALES

1. No linealidad. Una neurona artificial puede ser lineal o no lineal. Una red neuronal, formada por una interconexión de neuronas no lineales, es en sí misma no lineal.
2. Asignación de entrada-salida. Un paradigma popular de aprendizaje, llamado aprendizaje supervisado, implica la modificación de los pesos sinápticos de una red neuronal mediante la aplicación de un conjunto de ejemplos de entrenamiento etiquetados o ejemplos de tareas.
3. Adaptabilidad. Las redes neuronales tienen una capacidad incorporada para adaptar sus pesos sinápticos a los cambios en el entorno. En particular, una red neuronal capacitada para operar en un entorno específico se puede volver a entrenar fácilmente para hacer frente a cambios menores en las condiciones ambientales de funcionamiento.
4. Respuesta probatoria. En el contexto de la clasificación de patrones, una red neuronal puede diseñarse para proporcionar información no solo sobre qué patrón en particular seleccionar, sino también sobre la confianza en la decisión tomada.
5. Información contextual. El conocimiento está representado por la estructura misma y el estado de activación de una red neuronal. Cada neurona en la red se ve potencialmente afectada por la actividad global de todas las otras neuronas en la red.
6. Tolerancia a fallas. Una red neuronal, implementada en forma de hardware, tiene el potencial de ser inherentemente tolerante a fallas o capaz de computación robusta, en el sentido de que su rendimiento se degrada con gracia en condiciones de operación adversas.
7. VLSI Implementabilidad (very – large – scale - integrated). La naturaleza paralela masiva de una red neuronal lo hace potencialmente rápido para el cálculo de ciertas tareas.
8. Uniformidad de análisis y diseño. Básicamente, las redes neuronales disfrutan de la universalidad como procesadores de información.
9. Analogía neurobiológica. El diseño de una red neuronal está motivado por la analogía con el cerebro, que es una prueba viviente de que el procesamiento paralelo tolerante a fallas no solo es físicamente posible, sino también rápido y poderoso.

2.2 EL CEREBRO HUMANO

El sistema nervioso humano puede verse como un sistema de tres etapas, como se muestra en el diagrama de bloques de la figura 4 (Arbib, 1987). El centro del sistema es el cerebro, representado por la red neuronal (nervio), que recibe continuamente información, la percibe y toma las decisiones apropiadas. Dos conjuntos de flechas se muestran en la figura 4. Los que apuntan de izquierda a derecha indican la transmisión hacia adelante de señales portadoras de información a través del sistema. Las flechas que apuntan de derecha a izquierda (en azul) significan la presencia de retroalimentación en el sistema. Los receptores convierten los estímulos del cuerpo humano o del entorno externo en impulsos eléctricos que transmiten información a la red neuronal (cerebro). Impulsos eléctricos generados por la red neuronal en respuestas discernibles como salidas del sistema. (Haykin, 2009)

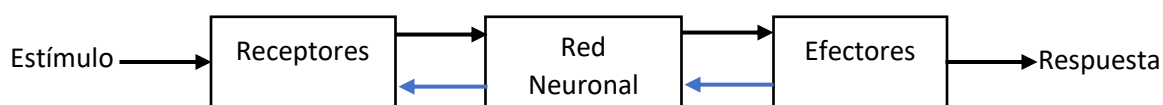


Figura 4 Representación del diagrama de bloques del sistema nervioso

Nota. Recuperado de Arbib, M.A., 1987. *Brains, Machines, and Mathematics*, 2d ed., New York: Springer-Verlag.

2.2.1 LA NEURONA BIOLÓGICA

Una neurona biológica es una célula especializada en procesar información. Está compuesta por el cuerpo de la célula (soma) y dos tipos de ramificaciones: el axón y las dendritas. La neurona recibe las señales (impulsos) de otras neuronas a través de sus dendritas y transmite señales generadas por el cuerpo de la célula a través del axón.

En general, una neurona consta de un cuerpo celular más o menos esférico, de 5 a 10 micras de diámetro, del que salen una rama principal, el axón y varias ramas más cortas que corresponden a las dendritas.

Una de las características de las neuronas es su capacidad de comunicarse. En forma concreta, las dendritas y el cuerpo celular reciben señales de entrada; el cuerpo celular las combina e integra y emite señales de salida. El axón transmite dichas señales a los terminales axónicos, los cuales distribuyen información.

Las señales que se utilizan son de dos tipos: eléctricas y químicas. La señal generada por la neurona y transportada a lo largo del axón es un impulso eléctrico, mientras que la señal que se transmite entre los terminales axónicos de una neurona y las dendritas de la otra es de origen químico.

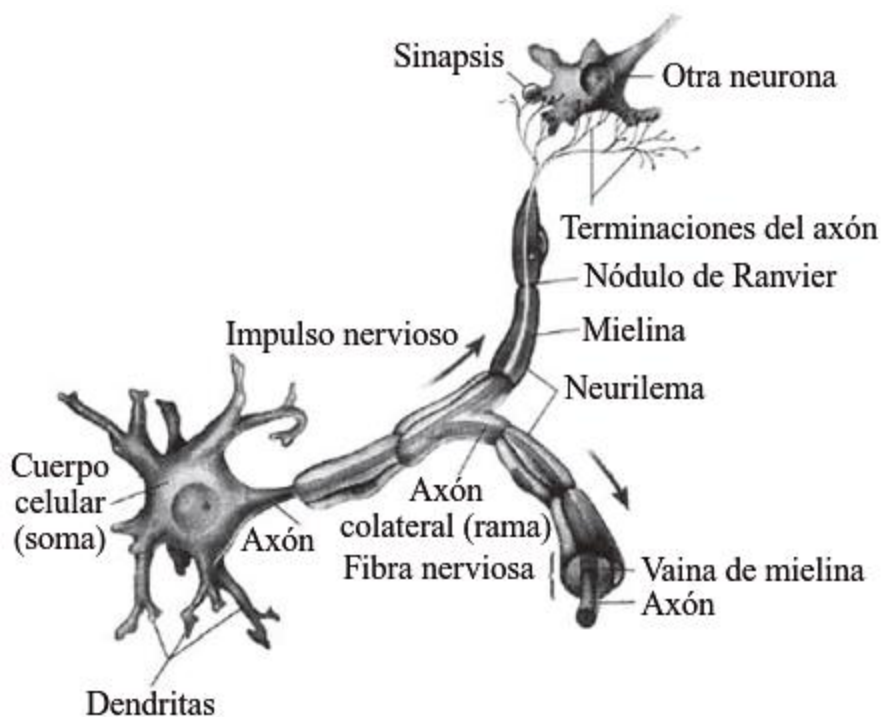


Figura 5 Elementos neuronales

Nota. Recuperado de Ponce Cruz, P. (2010). *Inteligencia artificial* (1st ed.). México, D.F.: Alfaomega.

Para establecer una similitud directa entre la actividad sináptica y la analogía con las redes neuronales artificiales podemos considerar que las señales que llegan a la sinapsis son las entradas a la neurona; éstas son ponderadas (atenuadas o simplificadas) a través de un parámetro denominado peso, asociado a la sinapsis correspondiente. Estas señales pueden excitar a la neurona (sinapsis con peso positivo) o inhibirla (peso negativo).

El efecto es la suma de las entradas ponderadas. Si la suma es igual o mayor que el umbral de la neurona, entonces la neurona se activa (da salida). Esta es una situación de todo o nada; cada neurona se activa o no se activa. Esta habilidad de ajustar señales es un mecanismo de aprendizaje. (Ponce Cruz, 2010)

2.2.2 LA NEURONA ARTIFICIAL

Como en el caso de la neurona biológica, la neurona artificial recibe unas entradas de estímulo que pueden provenir del sistema sensorial externo o de otras neuronas con las cuales posee conexión. Para el caso del modelo que proponemos en la figura 6, la información que recibe la neurona la definimos con el vector de entrada $\mathbf{X} = [x_1, x_2, \dots, x_n]$.

La información recibida por la neurona es modificada por un vector \mathbf{w} de pesos sinápticos cuyo papel es el de emular la sinapsis existente entre las neuronas biológicas. Estos valores se pueden asimilar a ganancias que pueden atenuar o amplificar los valores que se desean propagar hacia la neurona. El parámetro Θ_j se conoce como el bias o umbral de una neurona.

Los diferentes valores que recibe la neurona, modificados por los pesos sinápticos, los sumamos para producir lo que hemos denominado la entrada neta. Esta entrada neta es la que va a determinar si la neurona se activa o no.

La activación o no de la neurona depende de lo que llamaremos función de activación. La entrada neta la evaluamos en esta función y obtenemos la salida de la red. Si, por ejemplo, esta función la definimos como un escalón unitario, la salida será 1 si la entrada es mayor que cero, en caso contrario, la salida será 0.

Aunque no hay un comportamiento biológico que indique la presencia de algo parecido en las neuronas del cerebro, el uso de la función de activación es un artificio para poder aplicar las RNA (Redes Neuronales Artificiales) a una gran diversidad de problemas reales. De acuerdo con lo mencionado la salida y_j de la neurona se genera al evaluar la entrada neta en la función de activación.

Podemos propagar la salida de la neurona hacia otras neuronas o puede ser la salida de la red, que de acuerdo con la aplicación tendrá una interpretación para el usuario.

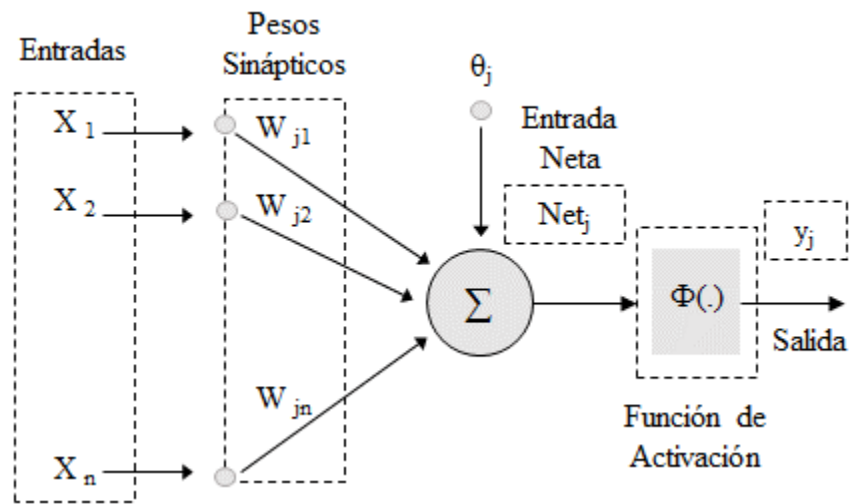


Figura 6 Modelo de Neurona Artificial

Nota. Recuperado de Caicedo Bravo, E., & López Sotelo, J. (2009). *Una aproximación práctica a las redes neuronales artificiales* (1st ed.). Cali: Programa Editorial Universidad del Valle.

2.2.2.1 PROCESO MATEMÁTICO EN LA NEURONA ARTIFICIAL

En una neurona artificial el cálculo de la entrada neta se puede representar con la ecuación 2.1, y en forma vectorial se representa con la ecuación 2.2

$$Net_j = \sum_{i=1}^N x_i w_{ji} + \theta_j \quad (2.1)$$

$$Net_j = w_1 x_{j1} + w_2 x_{j2} + \dots + w_i x_{ji} + \dots + w_N x_{jN} + \theta_j \quad (2.2)$$

$$Net_j = \mathbf{w}^T \mathbf{X}_j + \theta_j$$

La salida de la neurona artificial está determinada por una función de activación (*Fact*), tal como se aprecia en la ecuación 2.3.

$$y_j = Fact_j(Net_j) \quad (2.3)$$

La función de activación generalmente es de tipo escalón, lineal o sigmoideal. (Caicedo Bravo y López Sotelo, 2009).

2.2.2.2 FUNCIONES DE ACTIVACIÓN

Los modelos de neuronas utilizados en redes artificiales combinan sus entradas usando pesos que modelan sus conexiones sinápticas y, a continuación, le aplican a la entrada neta de la neurona una función de activación o transferencia. La entrada neta de la neurona recoge el nivel de estímulo que la neurona recibe de sus entradas y es la función de activación la que determina cuál es la salida de la neurona.

Las funciones de activación se pueden clasificar en:

1. Funciones de activación discretas

La salida de la neurona es discreta; esto es, sólo tomar un conjunto finito de valores. Normalmente, se utilizan dos valores, por lo que hablamos de neuronas binarias cuando la salida es 0 ó 1, mientras que hablamos de neuronas bipolares cuando su salida puede ser -1 ó +1.

2. Funciones de activación continuas

En este caso, la salida de la neurona puede tomar cualquier valor dentro de un intervalo. Generalmente, el rango de este intervalo está limitado, o bien al intervalo $[0, 1]$ o bien al intervalo $[-1, 1]$. (Berzal, 2018)

Algunas de las funciones de activación más usadas son las siguientes:

- Función escalón

La función de activación escalón en la figura 7 se asocia a neuronas binarias en las cuales, cuando la suma de las entradas es mayor o igual que el umbral de la neurona, la activación es 1; si es menor, la activación es 0 (o -1).

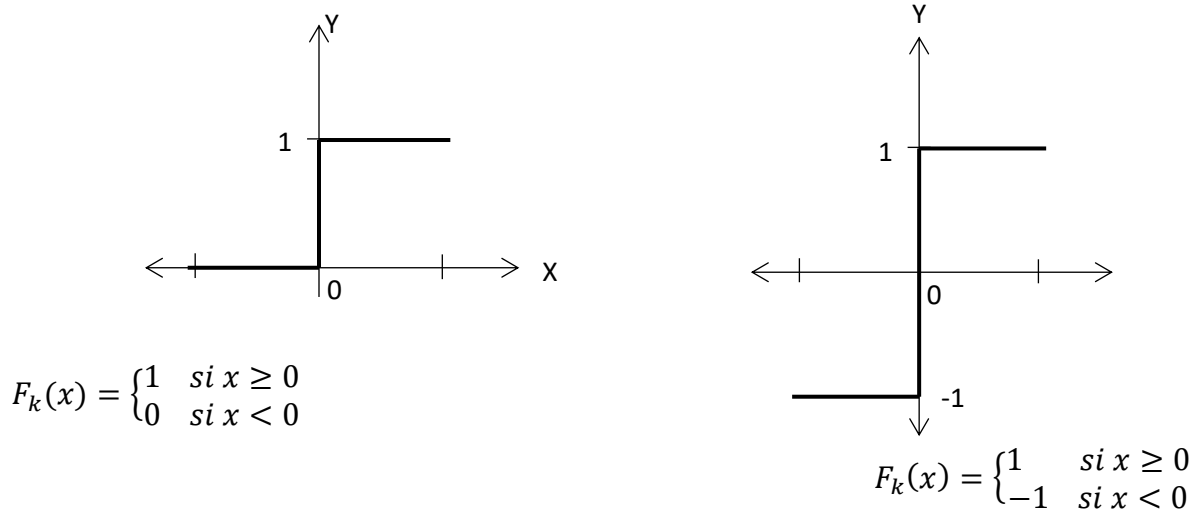
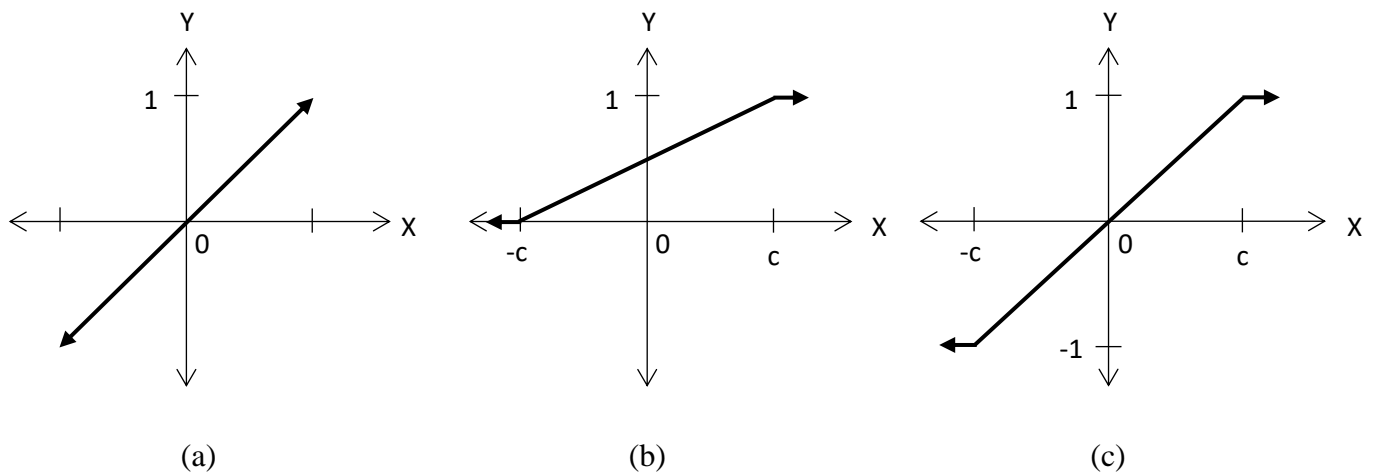


Figura 7 Función de activación escalón
 Nota. Recuperado de Ponce Cruz, P. (2010). *Inteligencia artificial* (1st ed.). México, D.F.: Alfaomega.

- Función lineal y mixta

La función lineal o identidad responde a la expresión $F_k(u) = u$. En las neuronas con función mixta, si la suma de las señales de entrada es menor que un límite inferior, la función se define como 0 (o -1). Si dicha suma es mayor o igual que el límite superior, entonces la activación es 1. Si la suma de entrada está comprendida entre los dos límites, entonces la activación es 1. Si la suma de entrada está comprendida entre ambos límites, superior e inferior, entonces la activación se define como una función lineal de la suma de las señales de entrada (véase la figura 8).



(a)

$$F_k(x) = x$$

(b)

$$F_k(x) = \begin{cases} 0 & \text{si } x < -c \\ 1 & \text{si } x > c \\ \frac{x}{2c} + \frac{1}{2} & \text{en otro caso} \end{cases}$$

(c)

$$F_k(x) = \begin{cases} -1 & \text{si } x < -c \\ 1 & \text{si } x > c \\ ax & \text{en otro caso} \end{cases}$$

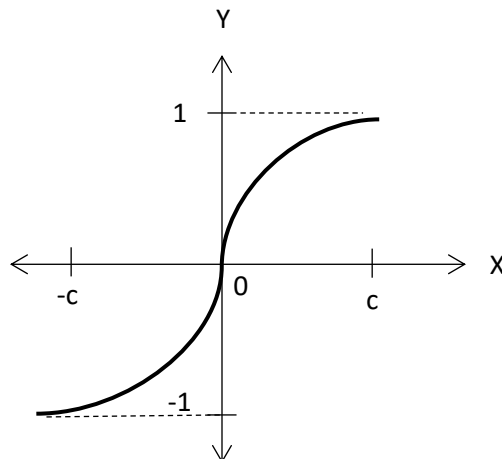
Figura 8 Función de actividad lineal y mixta

Nota. Recuperado de Ponce Cruz, P. (2010). *Inteligencia artificial* (1st ed.). México, D.F.: Alfaomega.

- Función tangente hiperbólica

La función de activación tangente hiperbólica se emplea en los casos que presentan variaciones suaves de valores positivos y negativos de la señal a clasificar. Como se puede ver en su descripción en la figura 9, es una de las funciones más empleadas en entrenamientos supervisados, como en el caso del entrenamiento de retropropagación del error.

Debe tenerse cuidado de emplear esta figura entre los umbrales positivos y negativos antes de la saturación, de otra forma la salida siempre generara valores saturados iguales a 1 y -1.



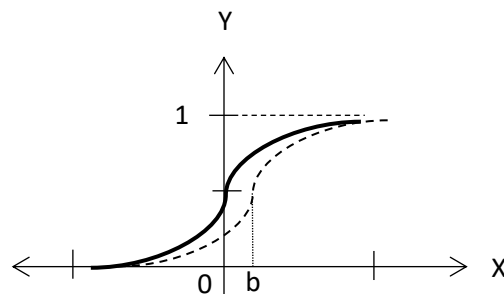
$$F_k(u) = \frac{e^u - e^{-u}}{e^u + e^{-u}}$$

Figura 9 Función tangente hiperbólica

Nota. Recuperado de Ponce Cruz, P. (2010). *Inteligencia artificial* (1st ed.). México, D.F.: Alfaomega.

- Función sigmoideal

Con la función sigmoideal el valor dado por la función es cercano a uno de los valores asintóticos. Esto hace que en la mayoría de los casos, el valor de salida esté comprendido en la zona alta o baja del sigmoide. De hecho, cuando la pendiente es elevada, esta función tiende a la función escalón. Sin embargo, la importancia de la función sigmoideal es que su derivada siempre es positiva y cercana a cero para los valores grandes positivos o negativos; además, toma su valor máximo cuando $x = 0$. Esto hace que se puedan utilizar reglas de aprendizaje definidas para las funciones escalón, con la ventaja, respecto a esta función, de que la derivada está definida en todo el intervalo (véase la figura 10).



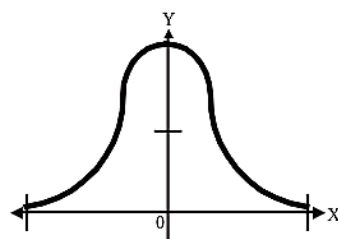
$$F_k(u) = \frac{1}{1 + e^{-u}}$$

Figura 10 Función sigmoideal

Nota. Recuperado de Ponce Cruz, P. (2010). *Inteligencia artificial* (1st ed.). México, D.F.: Alfaomega.

- Función de Gauss

Los mapeos ocultos algunas veces pueden realizarse con un solo nivel de neuronas mediante el uso de funciones de activación tipo Gauss, en lugar de funciones tipo sigmoideales (véase la figura 11). (Ponce Cruz, P. 2019)



$$F_k(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\left(\frac{x^2}{2\sigma^2}\right)}$$

Figura 11 Función de activación gaussiana

Nota. Recuperado de Ponce Cruz, P. (2010). *Inteligencia artificial* (1st ed.). México, D.F.: Alfaomega.

2.3 ELEMENTOS BÁSICOS QUE COMPONEN UNA RED NEURONAL

La distribución de las neuronas dentro de una red neuronal artificial se realiza formando niveles de un número de neuronas determinado. Si un conjunto de neuronas artificiales recibe simultáneamente el mismo tipo de información, lo denominaremos capa. En una red podemos diferenciar tres tipos de niveles:

- Entrada: Es el conjunto de neuronas que recibe directamente la información proveniente de las fuentes externas de la red.
- Oculto: Corresponde a un conjunto de neuronas internas a la red y no tiene contacto con el exterior. El número de niveles ocultos puede estar entre cero y un número elevado. En general las neuronas de cada nivel oculto comparten el mismo tipo de información, por lo que generalmente se denominan capas ocultas. Las neuronas de las capas ocultas pueden estar interconectadas de diferentes maneras, lo que determina, junto con su número, las distintas arquitecturas de redes neuronales.
- Salida: Es el conjunto de neuronas que transfieren la información que la red ha procesado hacia el exterior. (Caicedo Bravo y López Sotelo, 2009).

A continuación, se puede ver, en la figura 12, un esquema de una red neuronal:

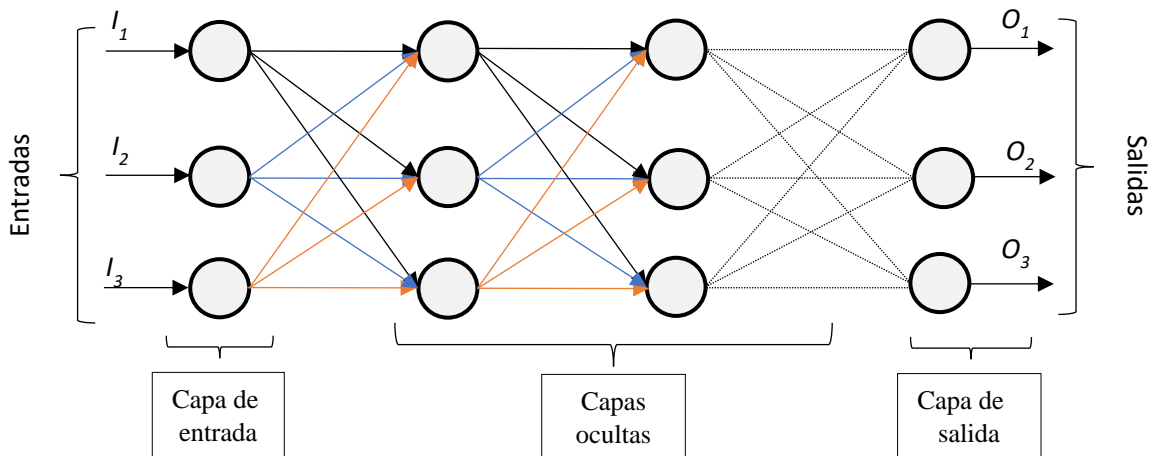


Figura 12 Ejemplo de una red neuronal totalmente conectada

Nota: Recuperado de Jorge Matich, D. (2001). *Ejemplo de una red neuronal totalmente conectada* [Image]. https://www.frro.utn.edu.ar/repositorio/catedras/quimica/5_anio/orientadora1/monogriais/matich-redesneuronales.pdf

2.4 ARQUITECTURA DE REDES

La elección de una arquitectura de red impacta en la capacidad de generalización de una red neuronal. Dependiendo en el tipo de arquitectura, una red neuronal es capaz de resolver tipos de problemas diferentes. Según Haykin, menciona tres tipos de arquitectura:

2.4.1 REDES DE UNA CAPA

Esta red sólo está compuesta por una capa de entrada y una salida (ver figura 13). Esta arquitectura es utilizada para resolver problemas lineales y no es capaz de resolver problemas no linealmente separables. (Ocampo Vega, 2014)

Una capa de neuronas

Una red de una sola capa de S neuronas se muestra en la figura 14. Tenga en cuenta que cada una de las R entradas está conectada a cada una de las neuronas y que la matriz de peso ahora tiene S filas. (Hagan, Demuth, Beale y De Jesús, 2016)

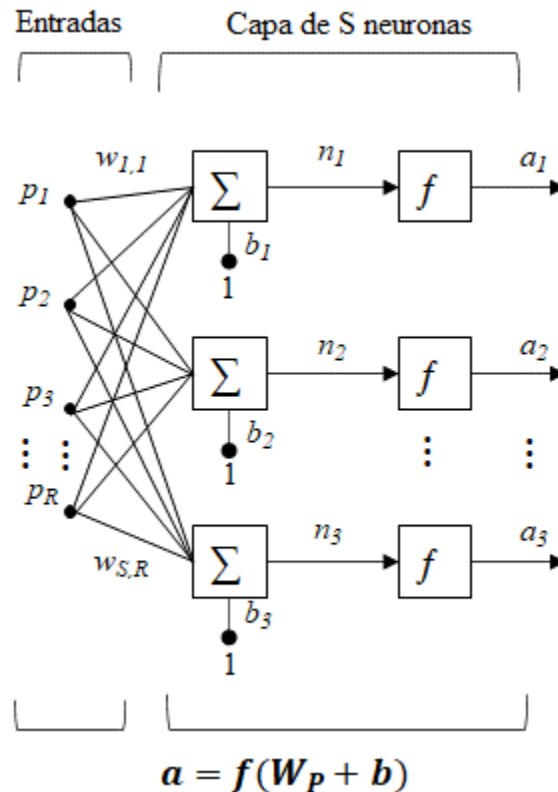


Figura 13 Capa de S Neuronas

Nota. Recuperado de Hagan, M., Demuth, H., Beale, M., & De Jesús, O. (2016). *Neural Network Design* (2nd ed.). [S. l.: s. n.].

La capa incluye la matriz de los pesos, las sumatorias, el vector bias **b**, los cuadros de función de transferencia y el vector de salida **a**.

Cada elemento del vector de entrada **p** está conectado a cada neurona a través de la matriz de pesos **W**. Cada neurona tiene un bias b_i , una sumatoria, una función de transferencia f y una salida a_i . En conjunto, las salidas forman el vector de salida **a**.

Es común que el número de entradas a una capa sea diferente del número de neuronas (es decir, $R \neq S$).

Los elementos del vector de entrada ingresan a la red a través de la matriz de pesos **W**:

$$\mathbf{W} = \begin{bmatrix} w_{1,1} & w_{1,2} & \dots & w_{1,R} \\ w_{2,1} & w_{2,2} & \dots & w_{2,R} \\ \vdots & \vdots & & \vdots \\ w_{S,1} & w_{S,2} & \dots & w_{S,R} \end{bmatrix}$$

Los índices de fila de los elementos de la matriz **W** indican la neurona de destino asociada con ese peso, mientras que los índices de columna indican la fuente de la entrada para ese peso. Por lo tanto, los índices en $w_{3,2}$ significa que este peso representa la conexión con la tercera neurona de la segunda fuente. (Hagan, Demuth, Beale y De Jesús, 2016)

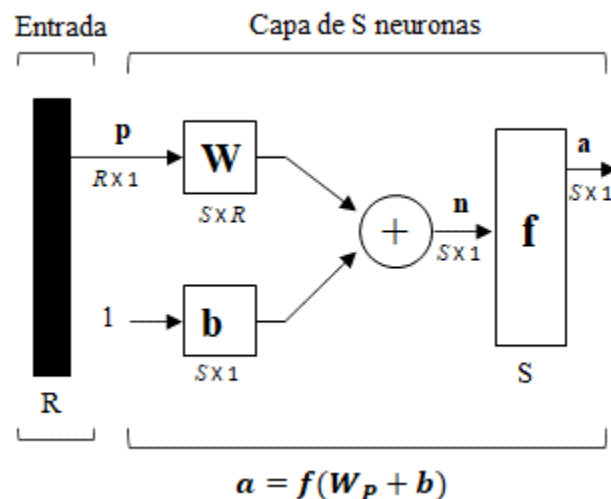


Figura 14 Capa de S Neuronas, Notación Abreviada

Nota. Recuperado de Hagan, M., Demuth, H., Beale, M., & De Jesús, O. (2016). *Neural Network Design* (2nd ed.). [S. l.: s. n.].

La red de una capa de S -neuronas, R -entradas, también se puede dibujar en notación abreviada, como se muestra en la Figura 14.

Los símbolos debajo de las variables significan que, para esta capa, \mathbf{p} es un vector de longitud R , \mathbf{W} es una matriz $S * R$, y \mathbf{a} y \mathbf{b} son vectores de longitud S . La capa incluye la matriz de pesos, las operaciones de sumatoria y multiplicación, el vector bias \mathbf{b} , los cuadros de función de transferencia y el vector de salida. (Hagan, Demuth, Beale y De Jesús, 2016)

2.4.2 REDES MULTICAPA

Las redes multicapa (ver figura 15) están compuestas por tres tipos de capas. La primera capa, la de entrada, recibe los estímulos del medio ambiente. El segundo tipo de capa, capa oculta, recibe las señales de los nodos de entrada, procesa la información y la envía a la capa de salida; la capa de salida procesa las señales enviadas por la capa oculta y realiza la clasificación. Cada capa se puede comunicar con capas adyacentes, pero solamente hacia adelante. Las redes multicapa son capaces de resolver problemas no linealmente separables. (Ocampo Vega, 2014)

Múltiples capas de neuronas

Ahora se considera una red con varias capas. Cada capa tiene su propia matriz de pesos \mathbf{W} , su propio vector bias \mathbf{b} , un vector de entrada neta \mathbf{n} y un vector de salida \mathbf{a} . Se inserta notación adicional para distinguir entre estas capas. Se utilizan superíndices para identificar las capas. Específicamente, se agrega el número de la capa como un superíndice a los nombres de cada una de estas variables. Por lo tanto, la matriz de pesos para la primera capa se escribe como \mathbf{W}^1 , y la matriz de pesos para la segunda capa se escribe como \mathbf{W}^2 . Esta notación se usa en la red de tres capas que se muestra en la Figura 15. (Hagan, Demuth, Beale y De Jesús, 2016)

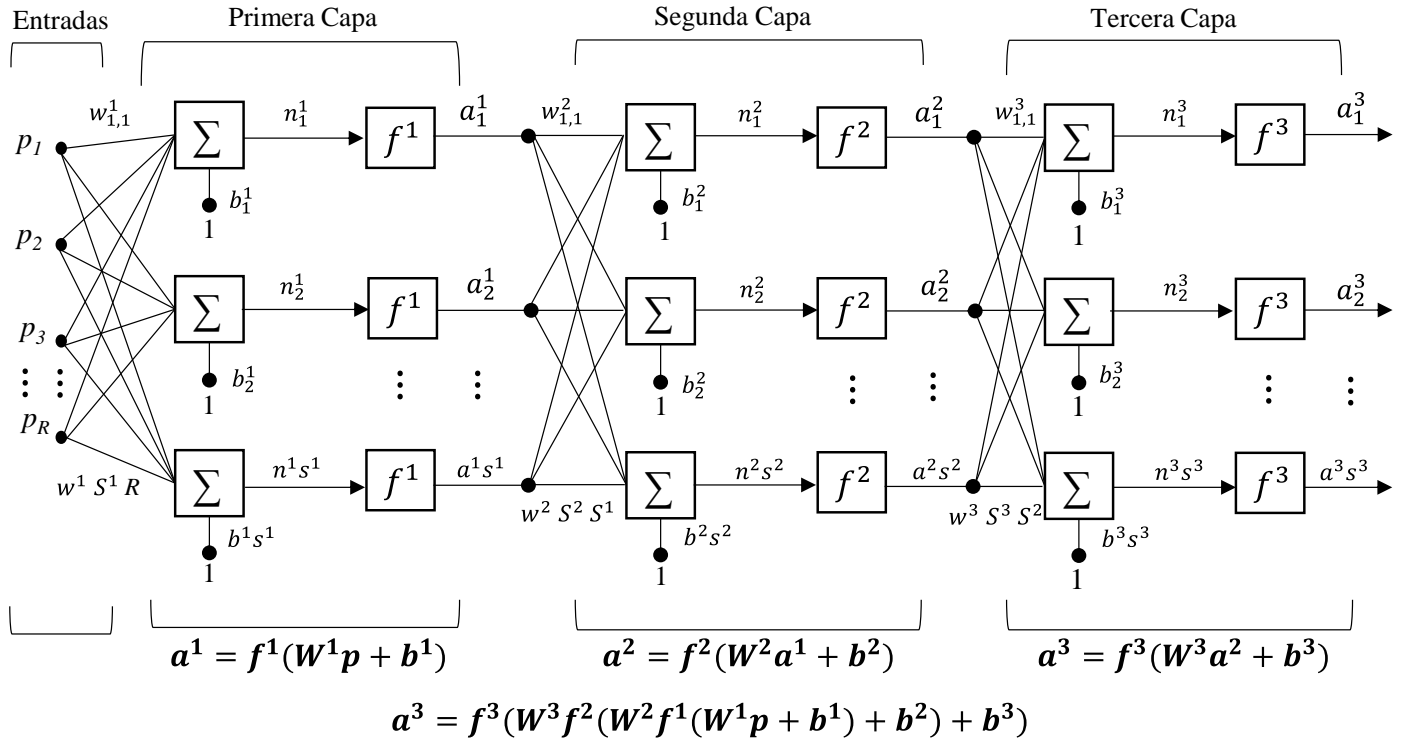


Figura 15 Red de tres capas

Nota. Recuperado de Hagan, M., Demuth, H., Beale, M., & De Jesús, O. (2016). *Neural Network Design* (2nd ed.). [S. l.: s. n.].

Como se muestra, hay entradas R , neuronas S^1 en la primera capa, neuronas S^2 en la segunda capa, etc. Las diferentes capas pueden tener diferentes números de neuronas.

Las salidas de las capas uno y dos son las entradas para las capas dos y tres. Así, la capa 2 puede verse como una red de una capa con $R = S^1$ entradas, $S = S^2$ neuronas y una $S^2 * S^1$ matriz de pesos \mathbf{W}^2 . La entrada a la capa 2 es \mathbf{a}^1 , y la salida es \mathbf{a}^2 .

Una capa cuya salida es la salida de la red se llama capa de salida. Las otras capas se llaman capas ocultas. La red que se muestra arriba tiene una capa de salida (capa 3) y dos capas ocultas (capas 1 y 2). La misma red de tres capas también se puede dibujar utilizando notación abreviada, como se muestra en la Figura 16. (Hagan, Demuth, Beale y De Jesús, 2016)

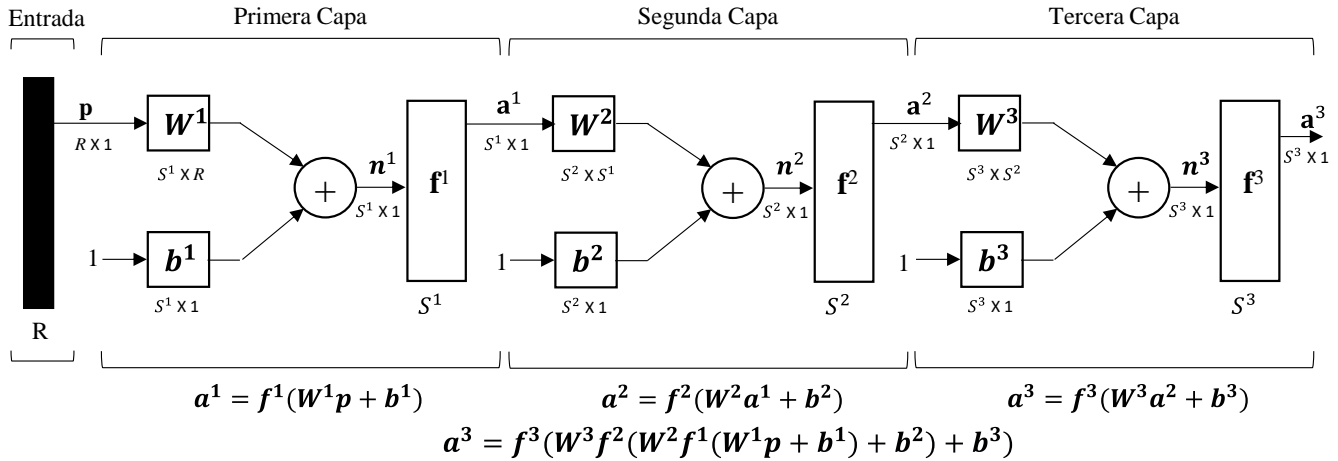


Figura 16 Notación abreviada, red de tres capas

Nota. Recuperado de Hagan, M., Demuth, H., Beale, M., & De Jesús, O. (2016). *Neural Network Design* (2nd ed.). [S. l.: s. n.].

El número de entradas a la red y el número de salidas de la red están definidos por especificaciones de problemas externos. Entonces, si hay cuatro variables externas para usar como entradas, hay cuatro entradas a la red. Del mismo modo, si debe haber siete salidas de la red, debe haber siete neuronas en la capa de salida. Finalmente, las características deseadas de la señal de salida también ayudan a seleccionar la función de transferencia para la capa de salida. Por lo tanto, la arquitectura de una red de una sola capa está determinada casi por completo por las especificaciones del problema, incluido el número específico de entradas y salidas y la característica particular de la señal de salida.

Ahora, ¿qué pasa si tenemos más de dos capas? Aquí el problema externo no le dice directamente el número de neuronas requeridas en las capas ocultas. De hecho, hay pocos problemas para los cuales uno puede predecir el número óptimo de neuronas necesarias en una capa oculta. Este problema es un área activa de investigación. (Hagan, Demuth, Beale y De Jesús, 2016)

2.4.3 REDES RECURRENTE

Esta arquitectura es similar a las redes de una capa y a la multicapa; la diferencia es que la salida de una neurona puede alimentar cualquier otra, inclusive, puede alimentarse a sí misma. Por otra parte, los elementos de retroalimentación implican el uso de conexiones compuestas por elementos de retardo de unidad de tiempo que resultan en un comportamiento dinámico no lineal. (Ocampo Vega, 2014)

2.5 MECANISMO DE APRENDIZAJE

Una red neuronal debe aprender a calcular la salida correcta para cada constelación neuronal (arreglo o vector) de entrada en el conjunto de ejemplos. Este proceso de aprendizaje se denomina: proceso de entrenamiento o acondicionamiento. El conjunto de datos (o conjunto de ejemplos) sobre el cual este proceso se basa es, por ende, llamado: conjunto de datos de entrenamiento.

El aprendizaje es el proceso por el cual una red neuronal modifica sus pesos en respuesta a una información de entrada. Los cambios que se producen durante el mismo se reducen a la destrucción, modificación y creación de conexiones entre las neuronas. En los sistemas biológicos existe una continua destrucción y creación de conexiones entre neuronas. En los modelos de redes neuronales artificiales, la creación de una nueva conexión implica que el peso de la misma pasa a tener un valor distinto de cero.

Durante el proceso de aprendizaje, los pesos de las conexiones de la red sufren modificaciones, por lo tanto, se puede afirmar que este proceso ha terminado (la red ha aprendido) cuando los valores de los pesos permanecen estables ($dw_{ij}/dt = 0$).

Un aspecto importante respecto al aprendizaje de las redes neuronales es el conocer cómo se modifican los valores de los pesos, es decir, cuáles son los criterios que se siguen para cambiar el valor asignado a las conexiones cuando se pretende que la red aprenda una nueva información.

Hay dos métodos de aprendizaje importantes que pueden distinguirse:

- a- Aprendizaje supervisado.
- b- Aprendizaje no supervisado

2.5.1 APRENDIZAJE SUPERVISADO

El aprendizaje supervisado se caracteriza porque el proceso de aprendizaje se realiza mediante un entrenamiento controlado por un agente externo (supervisor, maestro) que determina la respuesta que debería generar la red a partir de una entrada determinada. El supervisor controla la salida de la red y en caso de que ésta no coincida con la deseada, se procederá a modificar los pesos de las conexiones, con el fin de conseguir que la salida obtenida se aproxime a la deseada.

2.5.2 APRENDIZAJE NO SUPERVISADO

Las redes con aprendizaje no supervisado (también conocido como autosupervisado) no requieren influencia externa para ajustar los pesos de las conexiones entre sus neuronas. La red no recibe ninguna información por parte del entorno que le indique si la salida generada en respuesta a una determinada entrada es o no correcta.

Estas redes deben encontrar las características, regularidades, correlaciones o categorías que se puedan establecer entre los datos que se presenten en su entrada. Existen varias posibilidades en cuanto a la interpretación de la salida de estas redes, que dependen de su estructura y del algoritmo de aprendizaje empleado. (Matich, 2001).

En las técnicas de aprendizaje no supervisado, no existen clases predefinidas. Nuestro objetivo será encontrar patrones en los datos de entrada $x \in X$ que nos permitan construir un modelo de la distribución de probabilidad $p(X)$. ¿Qué utilidad puede tener algo así? El aprendizaje no supervisado nos puede servir como herramienta de análisis exploratorio de datos y para preprocesar los datos antes de utilizar una técnica supervisada. (Berzal, 2018)

- Exploración de datos
- Preprocesamiento de datos

2.5.2.1 CLUSTERING

La variante más conocida de las técnicas de aprendizaje no supervisado es la forma por los métodos de agrupamiento o clustering.

El objetivo de los algoritmos de clustering es encontrar agrupamiento de tal forma que los objetos de un grupo sean similares entre sí y diferentes de los objetos de otros grupos, a los que habitualmente llamaremos clusters.

Los resultados obtenidos por un método de agrupamiento dependen de varios factores: el algoritmo de agrupamiento empleado, el conjunto de datos disponible y la medida de similitud utilizada para comparar objetos. La medida de similitud, usualmente, se define como una medida de distancia.

2.5.2.1.1 MÉTODOS DE AGRUPAMIENTO POR PARTICIONES

En los métodos de agrupamiento de este tipo, generalmente se utiliza un parámetro k que indica el número de clusters que deseamos encontrar en los datos. Puede que se trate de un número de clusters ya conocido, fijado de antemano, o que tengamos que probar con distintos valores de k hasta encontrar el número más adecuado de clusters para nuestro conjunto de datos.

El algoritmo más conocido, el algoritmo de las k medias o k -means, asocia un centroide a cada cluster. Inicialmente, se pueden escoger k ejemplos aleatoriamente para inicializar los centroides (aunque también hay métodos más sofisticados de inicialización). Los puntos correspondientes a los distintos ejemplos se asignan al cluster cuyo centroide esté más cerca en cada momento. Iterativamente, el algoritmo recalcula los centroides en función de las asignaciones de puntos a clusters y reasigna los puntos a los clusters más cercanos, hasta que llega un momento en el que los centroides dejan de cambiar. (Berzal, 2018)

2.6 REDES DE NEURONAS DE BASE RADIAL

El trabajo original en funciones de base radial fue realizado por Powell y otros durante la década de 1980 (Powell, 1987). En este trabajo original, se utilizaron funciones de base radial para la interpolación exacta en un espacio multidimensional. En otras palabras, la función creada por la interpolación de base radial debía pasar exactamente a través de todos los objetivos en el conjunto de entrenamiento. El uso de funciones de base radial para la

interpolación exacta sigue siendo un área de aplicación importante, y también es un área activa de investigación. (Hagan, Demuth, Beale y De Jesús, 2016)

Un tipo de redes neuronales que fueron introducidas en 1985. Desde el punto de vista de su arquitectura es una red multicapa unidireccional con aprendizaje híbrido, pues en la capa oculta se sigue un algoritmo No-Supervisado y en la de salida el aprendizaje es Supervisado.

Debido a su simplicidad y velocidad en el aprendizaje, y el alto grado de generalización, esta red ha sido utilizada en diversas aplicaciones prácticas, sobre todo el reconocimiento y clasificación de patrones. Otro campo de aplicación que ha presentado resultados promisorios en la identificación y modelado de sistemas no lineales. (Caicedo Bravo y López Sotelo, 2009)

Las redes de base radial se caracterizan porque están formadas por una única capa oculta y cada neurona de esta capa posee un carácter local, en el sentido de que cada neurona oculta de la red se activa en una región diferente del espacio de patrones de entrada. Este carácter local viene dado por el uso de las llamadas funciones de base radial, generalmente la función gaussiana, como funciones de activación. Las neuronas de la capa de salida de las redes de base radial simplemente realizan una combinación lineal de las activaciones de las neuronas ocultas. (Isasi Viñuela y Galván León, 2008)

2.6.1 ARQUITECTURA DE LAS REDES DE BASE RADIAL

Las redes de neuronas de base radial están formadas por tres capas de neuronas: la capa de entrada, una única capa oculta y la capa de salida, como se muestra en la figura 17. La capa de entrada la componen un conjunto de neuronas que reciben las señales del exterior, transmitiéndolas a la siguiente capa sin realizar ningún proceso sobre dichas señales. Las neuronas de la capa oculta reciben las señales de la capa de entrada y realizan una transformación local y no lineal sobre dichas señales. Esta capa es la única que incluye componentes no lineales en las redes de base radial. Y, finalmente, la capa de salida que realiza una combinación lineal de las activaciones de las neuronas ocultas, que actúa además como salida de la red (Isasi Viñuela y Galván León, 2008)

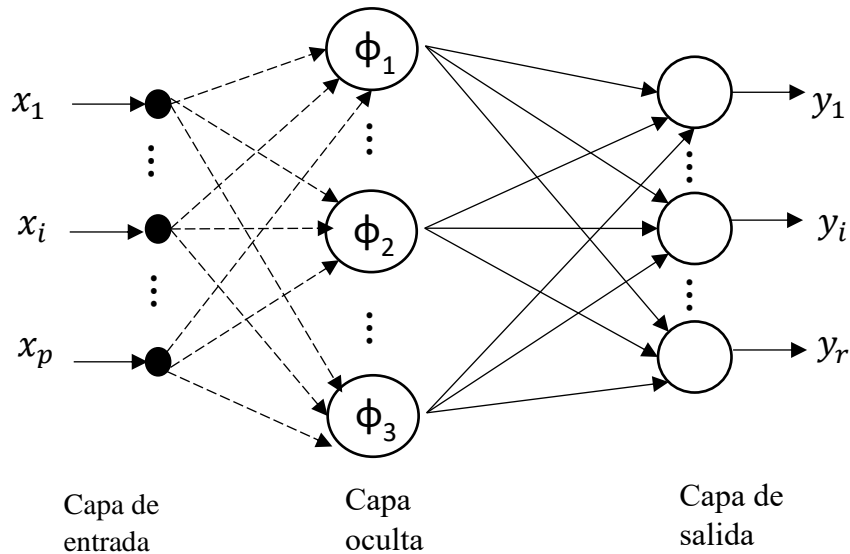


Figura 17 Arquitectura de la red de neuronas de base radial

Nota. Recuperado de Isasi Viñuela, P., y Galván León, I. (2008). *Redes de neuronas artificiales*. Madrid, [etc.]: Pearson-Prentice Hall.

Las redes de neuronas de base radial son redes con conexiones hacia adelante, como se observa en la figura 16, y estas conexiones se dirigen siempre de una capa a la siguiente capa. La red se caracteriza porque las conexiones de la capa de entrada a la capa oculta no llevan asociado ningún peso, mientras que, y como es habitual en el contexto de redes de neuronas, las conexiones de la capa oculta a la capa de salida sí llevan asociado un número real o peso de la conexión. En lo referente a los umbrales de las neuronas, en las redes de base radial únicamente las neuronas de salida poseen un umbral.

2.6.2 ACTIVACIONES DE LAS NEURONAS DE LA RED DE BASE RADIAL

Dada una red de neuronas de base radial con p neuronas en la capa de entrada, m neuronas en la capa de oculta y r neuronas en la capa de salida, las activaciones de las neuronas de salida para el patrón de entrada n , $X(n)=(x_1(n), x_2(n), \dots, x_p(n))$, denotadas como $y_k(n)$, vienen dadas por la siguiente expresión:

$$y_k(n) = \sum_{i=1}^m w_{ik} \phi_i(n) + u_k \text{ para } k = 1, 2, \dots, r \quad (2.4)$$

donde w_{ik} representa el peso de la conexión de la neurona oculta i a la neurona de salida k , u_k es el umbral de la neurona de salida k y $\phi_i(n)$ son las activaciones de las neuronas ocultas para el patrón de entrada $X(n)$. Se observa en la ecuación anterior que las neuronas de salida de la red utilizan la función de activación identidad, realizando una transformación lineal de las actividades de todas las neuronas ocultas.

Las funciones ϕ_i , también conocidas como funciones de base radial, determinan las actividades de las neuronas ocultas de la red en función del vector de entrada a la red $X(n)$ y vienen dadas por la siguiente expresión:

$$\phi_i(n) = \phi\left(\frac{\|X(n) - C_i\|}{d_i}\right) \text{ para } i = 1, 2, \dots, m \quad (2.5)$$

Donde ϕ es una función de base radial; $C_i = (c_{i1}, \dots, c_{ip})$ son vectores que representan los centros de la función de base radial; d_i son números reales que representan la desviación, anchura o dilatación de la función de base radial; y $\| \quad \|$ es la distancia euclídea del vector de entrada $X(n)$ al centro C_i , definida como:

$$\|X(n) - C_i\| = \left(\sum_{j=1}^p (x_j(n) - c_{ij})^2 \right)^{\frac{1}{2}} \quad (2.6)$$

Por lo tanto, la activación de una neurona oculta en las redes de base radial depende de la distancia del patrón de entrada $X(n)$ al centro C_i de la función de base radial. Estas funciones ϕ poseen un carácter local, pues son funciones que alcanzan un nivel cercano al máximo de su recorrido cuando el patrón de entrada $X(n)$ está próximo al centro de la neurona; a medida que el patrón se aleja del centro, el valor de la función va tendiendo al valor mínimo de su recorrido (ver figura 17). (Isasi Viñuela y Galván León, 2008)

La función de base radial ϕ puede adoptar diferentes formas y expresiones (figura 17), entre ellas:

- Función gaussiana:

$$\phi(r) = e^{\left(\frac{-r^2}{2}\right)} \quad (2.7)$$

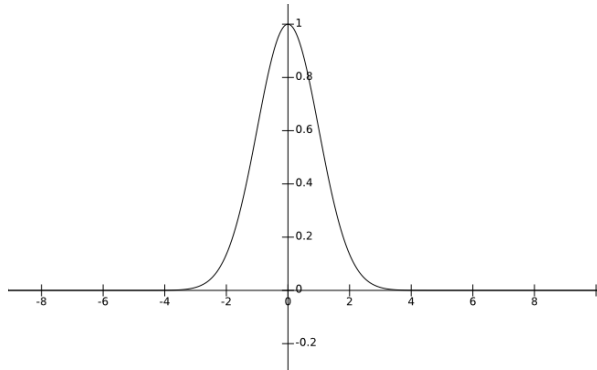
- Función inversa cuadrática:

$$\phi(r) = \frac{1}{1 + r^2} \quad (2.8)$$

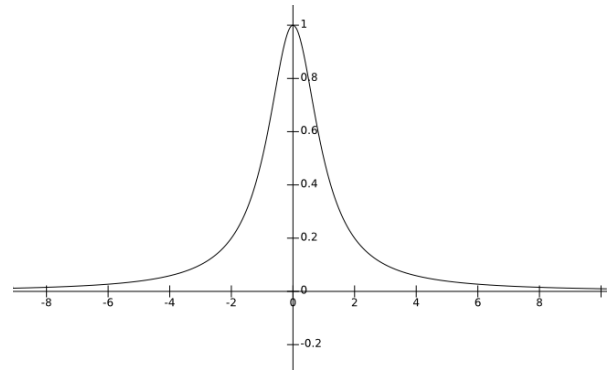
- Función inversa multicuadrática:

$$\phi(r) = \frac{1}{\sqrt{1 + r^2}} \quad (2.9)$$

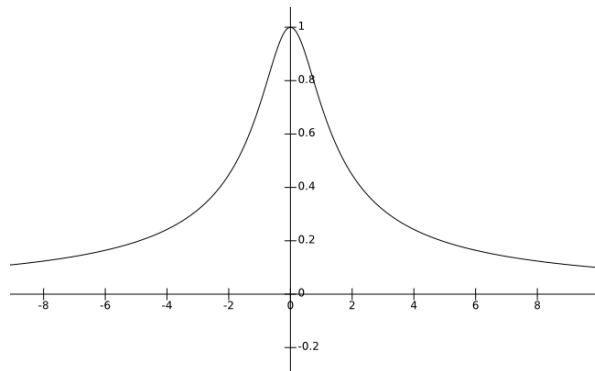
En el contexto de redes de neuronas de base radial, la más utilizada es la función gaussiana (Moody and Darken, 1989).



a) Función gaussiana



b) Función inversa cuadrática



a) Función inversa multicuadrática

Figura 18 Funciones de base radial

La función de activación ϕ es una función del tipo de base radial, normalmente la función de Gauss. Por esta razón, los parámetros que caracterizan a esta capa oculta son el centro de la función de activación (centroide) y su respectiva desviación estándar.

La capa de entrada corresponde simplemente a una interfaz entre el proceso a analizar o solucionar y la red neuronal. Toma los datos del entorno de trabajo de la red y los lleva a su interior para el procesamiento. La capa de salida procesa los datos provenientes de la capa oculta y nos entrega la salida de la red.

En la figura 19, apreciamos el modelo de un elemento de procesamiento de la capa oculta que recibe el vector de entrada \mathbf{x} de dimensión N , se calcula la norma respecto del vector centroide c_i , aplicamos la función de activación de base radial para generar la salida y_j de este elemento. (Caicedo Bravo y López Sotelo, 2009).

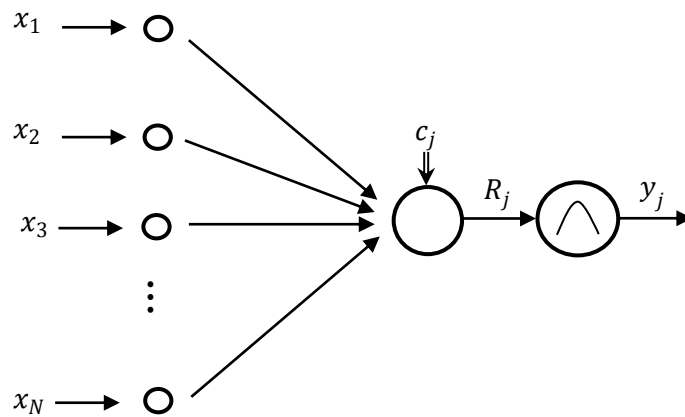


Figura 19 Elemento de procesamiento de una RFB

Nota. Recuperado de Caicedo Bravo, E., & López Sotelo, J. (2009). *Una aproximación práctica a las redes neuronales artificiales* (1st ed.). Cali: Programa Editorial Universidad del Valle.

En la figura 20, presentamos la arquitectura de una red RFB con N entradas, L elementos de procesamiento en la capa oculta con función de activación de base radial y un elemento de procesamiento en la salida con función de activación lineal.

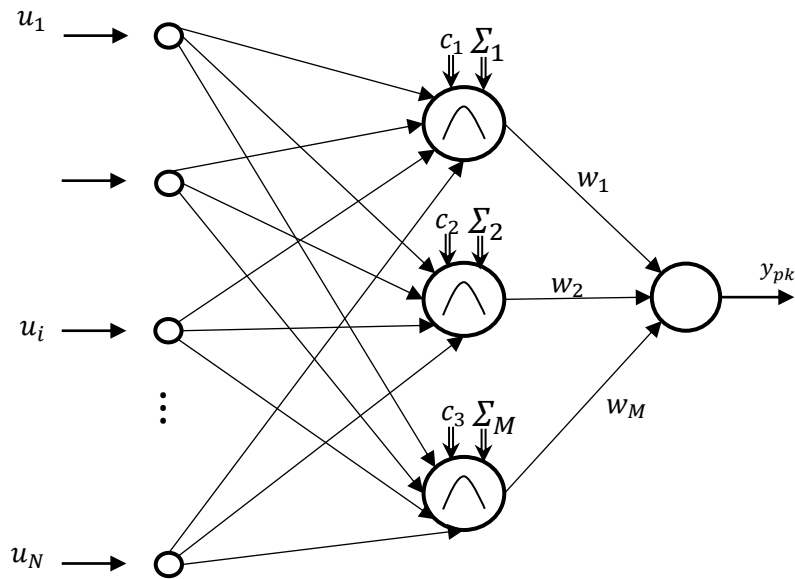


Figura 20 Arquitectura de una red RBF

Nota. Recuperado de Caicedo Bravo, E., & López Sotelo, J. (2009). *Una aproximación práctica a las redes neuronales artificiales* (1st ed.). Cali: Programa Editorial Universidad del Valle.

2.6.3 ENTRENAMIENTO DE LA RED

Con este tipo de red hemos planteado que el aprendizaje es híbrido por lo que incluye el aprendizaje supervisado en la capa de salida y el no-supervisado en la capa oculta; por esta razón el entrenamiento de las redes RBF lo dividiremos en dos etapas:

1. El entrenamiento de los elementos de procesamiento de la capa oculta lo realizaremos de forma no-supervisada. Presentaremos los datos de entrada y los nodos de procesamiento de esta capa, las unidades de procesamiento ocultas se encargarán de dividirlos por sectores y cada una de las neuronas asumirán la descripción de estos en el espacio de entrada a través del vector centroide.
2. El entrenamiento de los elementos de procesamiento de la capa de salida lo realizaremos de forma Supervisada con el algoritmo de corrección de error. (Caicedo Bravo y López Sotelo, 2009)

2.6.3.1 ENTRENAMIENTO DE LA CAPA OCULTA

El aprendizaje no-supervisado en estos elementos de procesamiento se cumple en las siguientes etapas:

1. Determinamos el valor de los centroides de los elementos ocultos y para ello utilizaremos el Algoritmo de las k-medias.

Como esta fase es No-Supervisada no conocemos a priori el número de clases o sectores en que se dividen los datos, por lo que se convierte en un parámetro de diseño que depende del tipo de problema y su adecuada selección se hace de manera heurística.

El paso siguiente es elegir los valores de los centroides C_j de los elementos de procesamiento de esta capa. La elección la podemos hacer de manera aleatoria, pero podemos tomar los primeros L patrones del conjunto de aprendizaje, donde L es el número de elementos de procesamiento ocultos escogido. El valor inicial de estos centroides es realmente irrelevante. (Caicedo Bravo y López Sotelo, 2009)

2. Luego de haber asignado el valor inicial de los centroides, en cada iteración t asignamos los patrones de aprendizaje x entre los L elementos de procesamiento ocultos. Cada patrón se asigna al elemento de cuyo centroide diste menos, con base en los siguientes pasos:
 - Tomamos el patrón x del conjunto de entrada y calculamos la distancia a cada uno de los centroides.
 - Verificamos cual es la menor de estas distancias.
 - El patrón de entrada se asigna al sector representado por el centroide cuya distancia haya sido mínima.
 - Repetimos este procedimiento para todos los patrones del conjunto de aprendizaje.
3. Calculamos los centroides de los nuevos sectores generados en el paso anterior, con base en la ecuación 2.10.

$$c_j = \frac{1}{p_j} \sum_{i=1}^{p_j} x_i \quad 2.10$$

p_j : Número de patrones que han correspondido a la neurona j-ésima.

4. Si hay variaciones en los centroides retornamos nuevamente al paso 2 en caso contrario, finalizamos la etapa de aprendizaje.
5. Calculamos con la ecuación 2.11 la desviación estándar σ_j por cada uno de los sectores encontrados, con el fin de determinar el radio de acción de cada uno de los elementos de procesamiento. (Caicedo Bravo y López Sotelo, 2009)

$$\sigma_j^2 = \frac{1}{p_j} \sum_{i=1}^{p_j} \|x_i - c_j\|^2 \quad 2.11$$

En la figura 21 se muestra el diagrama de flujo del algoritmo K- medias

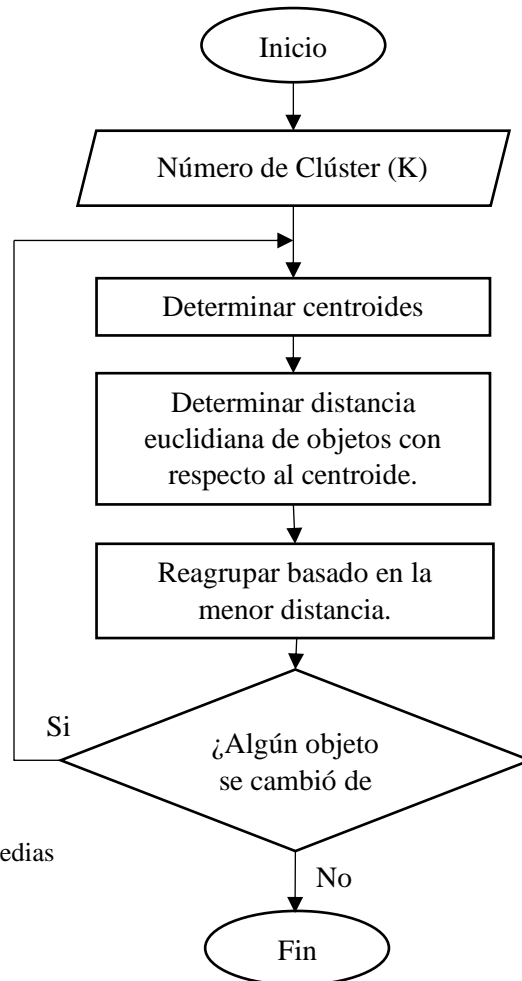


Figura 21 Algoritmo K-medias

2.6.3.2 ENTRENAMIENTO DE LA CAPA DE SALIDA

El entrenamiento de los elementos de procesamiento de la capa de salida lo realizaremos de forma supervisada con el algoritmo de corrección de error. Si conocemos la salida deseada d_{pk} y la salida que produce la red RBF y_{pk} podemos calcular el error global de la red con la ecuación 2.12, considerando a M como el número de elementos de procesamiento en la capa de salida y P la dimensión del conjunto de patrones de entrenamiento.

$$E_p = \frac{1}{2} \sum_{i=1}^P \sum_{k=1}^M \|d_{pk} - y_{pk}\|^2 \quad 2.12$$

Donde y_{pk} es la salida del k -ésimo elemento de procesamiento de la red, cuyo valor determinado por la ecuación 2.9, depende de las funciones de base radial; si reescribimos está incluyendo en la sumatoria el término de umbral, obtenemos la ecuación 2.13 y su representación matricial en la ecuación 2.14. Es bueno recordar, que el valor del umbral ϕ_0 es igual a 1.

$$y_{pk}(\mathbf{x}) = \sum_{j=0}^L \omega_{kj} \phi_j(r) \quad 2.13$$

$$\mathbf{y}(\mathbf{x}) = \mathbf{w}\Phi \quad 2.14$$

De la ecuación 2.14 vemos la dependencia lineal entre los pesos de la capa de salida y la salida de la red, por lo que la matriz de pesos que minimiza el error cuadrático se puede encontrar usando el método de los mínimos cuadrados. Otra forma de calcular esta matriz de pesos es utilizando métodos basados en gradiente. (Caicedo Bravo y López Sotelo, 2009).

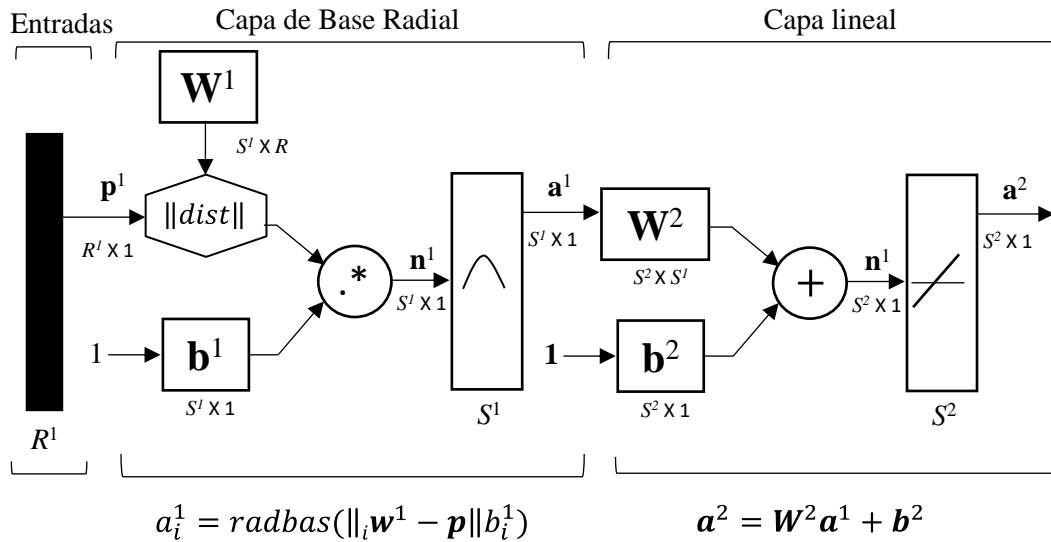


Figura 22 Red Neuronal de Base Radial

Nota. Recuperado de Hagan, M., Demuth, H., Beale, M., & De Jesús, O. (2016). *Neural Network Design* (2nd ed.). [S. l.: s. n.].

En la figura 22, la capa de entrada está formada por nodos fuente (unidades sensoriales) que conectan la red a su entorno. La segunda capa, que consta de unidades ocultas, aplica una transformación no lineal desde el espacio de entrada al espacio oculto (característica). Para la mayoría de las aplicaciones, la dimensionalidad de la única capa oculta de la red es alta; esta capa se entrena de manera no supervisada utilizando la etapa 1 del procedimiento de aprendizaje híbrido. La capa de salida es lineal, diseñada para suministrar la respuesta de la red al patrón de activación aplicado a la capa de entrada; esta capa se entrena de manera supervisada utilizando la etapa 2 del procedimiento híbrido. (Hagan, Demuth, Beale y De Jesús, 2016)

2.7 DIFERENCIAS ENTRE LAS REDES NEURONALES TIPO MLP Y RBF

- Una red tipo MLP (Multi Layer Perceptron) puede tener varias capas ocultas, mientras que en una RBF (Radial Basis Functions) solo podemos definir una capa oculta.
- El tipo de información de entrada a la capa oculta de una red RBF es diferente al que se procesa en una red MLP. La información que llega a la capa oculta de una red tipo

MLP es la suma ponderada de las entradas de la red, mientras que en la red tipo RBF la capa oculta procesa la diferencia entre el vector de entrada y el vector centroide de las neuronas.

- El procesamiento en la capa oculta en una red tipo MLP es global mientras en una red RBF es local. Esta afirmación es una consecuencia del tipo de funciones de activación que estas redes tienen en su capa oculta, en una red tipo RBF la capa oculta tiene función de Gauss cuya activación se encuentra localizada alrededor de su centro donde la salida es máxima, mientras que las redes tipo MLP tienen funciones de activación lineales o sigmoideas en donde la salida máxima se da para un rango mucho mayor de entradas. (Caicedo Bravo y López Sotelo, 2009).

2.8 MÉTODO DEL CODO (ELBOW METHOD)

Este método utiliza los valores de la inercia obtenidos tras aplicar el K-means a diferente número de Clusters (desde 1 a N Clusters), siendo la inercia la suma de las distancias al cuadrado de cada objeto del Cluster a su centroide:

$$Inercia = \sum_{i=0}^N \|x_i - \mu\|^2$$

Una vez obtenidos los valores de la inercia tras aplicar el K-means de 1 a N Clusters, representamos en una gráfica lineal la inercia respecto del número de Clusters. En esta gráfica se debería de apreciar un cambio brusco en la evolución de la inercia, teniendo la línea representada una forma similar a la de un brazo y su codo. El punto en el que se observa ese cambio brusco en la inercia nos dirá el número óptimo de Clusters a seleccionar; o, dicho de otra manera: el punto que representaría al codo del brazo será el número óptimo de Clusters. (Moya García, 2016)

CAPÍTULO 3: ESTADO DEL ARTE

3.1 IMPLEMENTACIONES DE RED NEURONAL Y/O ALGORITMO K-MEDIAS

La red neuronal artificial tiene una gran relevancia ya que a lo largo del tiempo ha tenido un campo de aplicación en diferentes áreas, así como el algoritmo k-medias, la tabla 3 muestra la descripción ascendente por año de publicación de algunos artículos científicos.

Tabla 3. Artículos científicos

Año de publicación	Título	Objetivo o problemática	Metodología	Resultado
1989	Phoneme Classification Experiments Using Radial Basis Functions	Se describe la aplicación de una red de funciones de base radial a un problema de clasificación de patrones de voz estática. Los resultados del reconocimiento se comparan con los obtenidos mediante la retropropagación y un modelo de Markov.	Se describe un método computacionalmente eficiente para resolver exactamente redes lineales de forma no iterativa. El método se aplicó a la clasificación de vocales en 20 clases utilizando 3 tipos diferentes de análisis de entrada y un número variable de funciones de base radial.	Resultados similares al modelado de Markov oculto cuantificado por vectores y la retropropagación convencional. La red de funciones de base radial es una alternativa atractiva a la propagación inversa. Renals, S., & Rohwer, R. (1989, June).
1992	Using Radial Basis Functions to Approximate	La red VI, derivada de redes de funciones de base radial, ajusta funciones y calcula	En el trabajo actual, determinamos si hay suficientes datos de entrenamiento	La activación de las unidades RBF está directamente relacionada con la proximidad

	Function and Its Error Bounds	intervalos de confianza para sus predicciones, indicando regiones locales de mal ajuste y extrapolación.	estimando la densidad de datos de entrenamiento local.	del punto de prueba a los datos de entrenamiento, propiedad que se explota en el diseño de la red VI. Leonard, J. A., Kramer, M. A., & Ungar, L. H. (1992).
1996	Median Radial Basis Function Neural Network	Un nuevo algoritmo para estimar los pesos de RBF basado en una estimación robusta y llamado RBF mediano.	Las funciones de base radial (RBF) consisten en una red neuronal de dos capas, donde cada unidad oculta implementa una función de núcleo. Cada núcleo está asociado con una región de activación desde el espacio de entrada y su salida se alimenta a una unidad de salida. Para encontrar los parámetros de una red neuronal que incorpora esta estructura, tenemos en cuenta dos enfoques estadísticos diferentes.	Se compararon los algoritmos RBF y MRBF en la estimación en PDF de datos generados artificialmente, así como en la segmentación de movimiento de una secuencia de imágenes de la vida real. En ambos casos, el MRBF dio una mejor estimación de los PDF implícitos y mostró mejores capacidades de clasificación. Bors, A. G., & Pitas, I. (1996).

1997	Growing Radial Basis Neural Networks: Merging Supervised and Unsupervised Learning with Network Growth Techniques.	Presenta un enfoque alternativo para construir y entrenar redes neuronales de función de base radial en crecimiento (GRBF). Este esquema se basa en la minimización de una medida de varianza condicional de clase localizada, que se calcula a partir de las salidas de las funciones de base radial y se relaciona con la estructura del espacio de características.	El esquema de aprendizaje híbrido propuesto proporciona un marco para incorporar algoritmos existentes en el entrenamiento de redes GRBF. Estos incluyen algoritmos no supervisados para la agrupación y la cuantización del vector de aprendizaje, así como algoritmos de aprendizaje para el entrenamiento de redes neuronales lineales de una sola capa.	Esta estrategia permite la representación del espacio de entrada de diferentes niveles de resolución creando pocos prototipos para representar regiones suaves en el espacio de entrada y utilizando los recursos disponibles donde realmente son necesarios. El esquema híbrido propuesto es compatible con una variedad de algoritmos de agrupamiento y LVQ (cuantización de vectores de aprendizaje). Karayiannis, N. B., & Mi, G. W. (1997).
1998	Normalized Gaussian Radial Basis Function networks	Se propone una nueva regla de aprendizaje que en gran medida reduce el número de nodos ocultos necesarios en las tareas de clasificación.	Los rendimientos de las redes RBF (NRBF) normalizadas y las redes RBF estándar se comparan en problemas simples de clasificación y mapeo. Los nodos ocultos realizan una función	Cuando se aplica a problemas de clasificación, el procedimiento de aprendizaje modificado da como resultado que los nodos se recluten principalmente a lo largo de los

			similar a una teselación de Voronoi del espacio de entrada, y los pesos de salida se convierten en la salida de la red sobre la partición definida por los nodos ocultos.	límites de la clase. Esto evita la necesidad de una cobertura densa de los dominios de clase, en contraste con las redes RBF estándar. Bugmann, G. (1998).
1998	Performance Evaluation of a Sequential Minimal Radial Basis Function (RBF) Neural Network Learning Algorithm	Este artículo presenta un análisis de rendimiento detallado del algoritmo de aprendizaje de la red de asignación de recursos mínimos (M-RAN) desarrollado recientemente.	El rendimiento de este algoritmo se compara con el de las redes de retroalimentación multicapa (MFN) entrenadas con 1) una variante del algoritmo de retropropagación estándar, conocido como RPROP y 2) el algoritmo de identificación de dependencia (DI) de Moody y Antsaklis en varios problemas de referencia en el Aproximación de funciones y áreas de clasificación de patrones.	Los resultados basados en problemas de referencia de la base de datos PROBEN1 muestran que M-RAN produce una red neuronal RBF con menor complejidad que la MFN que usa RPROP con una precisión de aproximación / clasificación similar. Yingwei, L., Sundararajan, N., & Saratchandran, P. (1998).
2001	Application of Minimal Radial	Presenta un nuevo enfoque para la protección de las	Presenta la clasificación de fallas y la ubicación en la protección a	Al usar esta nueva red, el número de patrones de

	Basis Function Neural Network to Distance Protection	líneas de transmisión de energía utilizando una red neuronal de función de base radial mínima (MRBFNN).	distancia utilizando una red neuronal RBF mínima (MRBFNN) donde el proceso de aprendizaje se hace secuencial y se elige el número óptimo de neuronas ocultas con una estrategia de poda.	entrenamiento y el tiempo de entrenamiento se reducen drásticamente y la clasificación de fallas y la precisión de la ubicación se mejoran significativamente. Dash, P. K., Pradhan, A. K., & Panda, G. (2001).
2002	Uso de redes neuronales para analizar modelos mecánicos de tejidos biológicos.	Se utilizaron redes neuronales para identificar la fuerza de indentación a partir de las propiedades elásticas y geométricas de una capa de cartílago articular modelada como un material transversalmente isótropo y para hallar el esfuerzo nominal de los ligamentos modelados como un material viscoelásticos sometido a deformaciones finitas.	Las soluciones analíticas de los dos problemas se utilizaron para crear el conjunto de los datos necesarios para el entrenamiento de redes neuronales tipo base radial, backpropagation y anfis. La verificación de su funcionamiento mostró que algunas de las redes aproximan la solución analítica con menos de 2%.	La backpropagation fue la más indicada para analizar estos problemas biomecánicos. Por una parte, el tiempo de entrenamiento fue significativamente menor que el tiempo requerido para una red anfis, y un poco mayor que el tiempo para una red radial. Cortes, D., Garcia, J., & Garcia, J. (2002).

2003	Radial basis function neural networks for nonlinear Fisher discrimination and Neyman-Pearson classification	Nueva técnica para el diseño de redes neuronales (NNs) de función de base radial (RBF).	Derivamos una nueva estrategia para implementar el análisis discriminante de Fisher no lineal utilizando un RBF NNs eligiendo las salidas de red deseadas.	Los datos sobre un problema de inspección de productos agrícolas y sobre datos sintéticos confirman la efectividad de estos métodos. Casasent, D., & Chen, X. W. (2003).
2003	On the Construction and Training of Reformulated Radial Basis Function Neural Networks.	Este artículo presentó un enfoque sistemático desarrollado para construir redes neuronales reformuladas con función de base radial adecuadas para el aprendizaje de gradiente descendente.	Este artículo también introdujo una nueva familia de redes neuronales reformuladas de función de base radial construidas por funciones de coseno de base radial e investigó su entrenamiento por descenso de gradiente.	Los experimentos indicaron que las redes neuronales de avance (FFNN) lograron una precisión considerablemente mayor que las redes neuronales de función de base radial convencionales. Karayiannis, N. B., & Randolph-Gips, M. M. (2003).
2003	Recurrent radial basis function network for time-series prediction.	Este artículo propone una red de función de base radial recurrente (RRBFN) que se puede aplicar a la	Basado en la arquitectura de las redes convencionales de función de base radial, el RRBFN tiene neuronas en bucle de entrada con funciones de activación	La red es capaz de predecir las dos series temporales y da buenos resultados para la identificación del sistema no lineal. La ventaja de la red

		monitorización dinámica y al pronóstico.	sigmoidea. Estas neuronas en bucle representan la memoria dinámica del RRBF, y las neuronas gaussianas representan la estática.	RRBF propuesta es combinar la flexibilidad de aprendizaje de la red RBF con los rendimientos dinámicos de la recurrencia local dada por las neuronas en bucle. Zemouri, R., Racoceanu, D., & Zerhouni, N. (2003).
2004	A Self-Adaptive Quantum Radial Basis Function Network for Classification for Classification Applications	Se propone una red de función de base radial cuántica autoadaptativa (QRBFN) para las aplicaciones de clasificación.	Se propone un algoritmo de aprendizaje autoadaptativo, que consiste en el algoritmo de autoagrupación (SCA) y el algoritmo de retropropagación. El método propuesto de SCA es un algoritmo rápido de un paso para una estimación dinámica del número de clústeres en un espacio de datos de entrada. El algoritmo de retropropagación para realizar el aprendizaje de parámetros.	1) Converge rápidamente; 2) Se construye automáticamente; 3) Tiene un error rms mucho menor; 4) tiene una tasa de clasificación de precisión más alta que otros modelos; y 5) puede representar y cuantificar la incertidumbre inherente a un conjunto de datos de entrenamiento. Lin, C. J., Chen, C. H., & Lee, C. Y. (2004)

2005	A Novel Radial Basis Function Network Classifier with Centers Set by Hierarchical Clustering.	Incorporar un algoritmo de agrupamiento jerárquico incremental para construir un RBFN optimizado para aplicaciones de clasificación de datos.	La contribución consta de dos partes. El primero es un algoritmo de agrupamiento jerárquico incremental para construir la capa oculta, y el segundo es mejorar el método de error cuadrático medio mínimo que calcula los pesos entre las capas ocultas y de salida de un RBFN.	Los resultados experimentales muestran que el clasificador de datos construido es capaz de ofrecer una precisión de clasificación comparable a la de la máquina de vectores de soporte (SVM) y el clasificador basado en la estimación de densidad de kernel. Ou, Y. Y., Oyang, Y. J., & Chen, C. Y. (2005)
2007	A fast method for implicit surface reconstruction based on radial basis functions network from 3D scattered points.	En este artículo se propone un método para la reconstrucción de superficies en 3D a partir de muchos puntos dispersos arbitrariamente.	se utiliza un algoritmo K-Means adaptado para reducir los centros de reconstrucción. Para que las características estén completas, se introducen dos métodos efectivos para extraer los puntos de característica antes del algoritmo K-Means adaptado.	El método de construcción se basa en funciones de base radial y el método de partición de unidad. Además, proponemos un algoritmo K-Means adaptado para para reducir los centros de interpolación en contraste al método convencional de

				reconstrucción RBF. Liu, H., Wang, X., & Qiang, W. (2007).
2007	La función de base radial de rango tipo M para la clasificación de imágenes de mamografía.	En este trabajo se propone la red neuronal función de base radial de rango tipo M (RMRBF) para la detección de anomalías que puedan producir cáncer de mama.	La red neuronal propuesta RMRBF se evaluó usando diferentes funciones de influencia, y su rendimiento se comparó con el de las redes neuronales RBF básica, mediana RBF y α -Trimmed Mean RBF.	Se propuso un nuevo tipo de red neuronal llamada función de base radial de rango tipo M (RMRBF) para la clasificación de imágenes de mamografía. Los resultados obtenidos con RMRBF propuesta son mejores que los resultados obtenidos mediante el uso de los algoritmos consultados. Gallegos-Funes, F. J., Ponomaryov, V., Michtchenko, A., & Moreno-Escobar, J. A. (2008).
2008	Redes neuronales de base radial aplicadas a la mejora de la calidad.	Construir una Red Neuronal Artificial RNA con Función de Base Radial, y que utiliza la distancia de Mahalanobis RND, para la mejora de la	Se desarrolló una RNA para aplicarla a la mejora de la calidad de procesos RND. Se evaluó la RND con relación a otras RNA y las técnicas estadísticas	Si se requiere calcular los valores para una calidad deseada, se deben calcular las RNA Inversa y Directa. Cevallos, J. (2008)

		calidad de diseño de procesos.	tradicionales de mejora de la calidad de procesos a fin de evaluar su desempeño a través del error de sus resultados.	
2009	Redes Neuronales Artificiales de Base Radial como Clasificador Difuso: Una Aplicación en Diagnóstico Médico.	Se presenta una alternativa para trabajar una Red Neuronal Artificial de Función de Base Radial (RBF) modificada para ser aplicada como Clasificador Difuso.	Para establecer una comparación se implementó el mismo sistema con redes neuronales tipo Perceptrón Multicapa MLP y, también se comparó con resultados de métodos de clasificación reportados en la literatura basados en Clustering Difuso como Fuzzy C-Means (FCM).	Los resultados logrados con la RBF como clasificador difuso permiten evidenciar que con esta alternativa se obtiene un buen clasificador con un bajo margen de error y con un buen desempeño en la generalización comparada con los otros métodos. Botia, J., Sarmiento, H., & Isaza, C. (2009).
2010	Evolutionary radial basis function network for classificatory problems	El algoritmo se utilizó para resolver el problema de detección de la diabetes india PIMA. Los datos se tomaron del depósito de aprendizaje automático de UCI.	El diagnóstico médico se ocupa de la detección automática de enfermedades basadas en los atributos de entrada. En este artículo, primero proponemos una naturaleza clasificatoria de la RBFN. Los diversos parámetros	El sistema fue capaz de resolver efectivamente el problema de detección de diabetes en los pacientes. Se desarrolló el algoritmo de entrenamiento del sistema. Utilizando la regla de aprendizaje del mapa de

			del sistema que incluyen el número de neuronas por clase, sus centros y extensiones correspondientes se optimizan mediante el uso de Algoritmo evolutivo personalizado.	organización de Kohonen. Kala, R., Vazirani, H., Khanwalkar, N., & Bhattacharya, M. (2010).
2010	Modelos de Redes Neuronales Perceptrón Multicapa y de base radial para la predicción del rendimiento académico de alumnos universitarios.	Diseñar e implementar modelos de RNA para predecir el rendimiento académico de los alumnos de primer año de la FACENA-UNNE, en base a los datos socioeducativos disponibles de los mismos.	Los datos de sus características socioeducativas se incorporan periódicamente en un almacén de datos con un diseño orientado a las decisiones. Este proceso incluye la integración, depuración y formateo de los datos, siguiendo las técnicas usuales de preprocesado, constituyentes de las etapas previas al modelado y análisis de los datos.	La arquitectura de las redes que presentan los modelos PM y FBR entrenados tienen la forma I:N-N-N:O, donde I es el número de variables de entrada, O es el número de variables de salida, y N es el número de unidades en cada capa. El modelo perceptrón multicapa (PM) predice adecuadamente tanto el Rendimiento Malo (0) como el Rendimiento Regular (1) y el Bueno (2), situación que no ocurre con el modelo de base radial (FBR). Longoni, M.

				G., Porcel, E., López, M. V., & Dapozo, G. N. (2010).
2012	Estimación de distancia con sensores ópticos reflexivos usando redes neuronales con funciones de base radial para aplicaciones embebidas.	El presente artículo tiene por objeto reducir el costo computacional, en términos de espacio de memoria y tiempo de procesamiento, en la linealización de la salida de los sensores para ser usados en sistemas embebidos de bajo costo.	Para el diseño de la red se estudió la eficiencia respecto al tamaño de la red y a la distribución de los centroides con resultados importantes. Para escoger el número de centroides se realizó de forma heurística.	Los resultados de dos diferentes sensores indican que la curva exponencial estimada se puede ajustar usando redes neuronales con funciones de base radial con una reducción de más del 50 % en el tiempo de procesamiento. Valencia, J. S. B., & Guzmán, L. J. M. (2013).
2013	Clustering Algorithms for radial basis function neural network	Las redes neuronales de función de base radial (RBFNN) forman una clase de redes neuronales artificiales, que tiene ciertas ventajas sobre otros tipos de redes neuronales, incluidas mejores capacidades de aproximación, estructura de	El entrenamiento RBFNN se basa principalmente en la estimación de varios parámetros: centros, anchos, pesos de conexión entre neuronas, etc. Los algoritmos de agrupamiento se utilizan en la estimación del parámetro central. Introducimos dos algoritmos, es decir, K-medias y algoritmos de	La función de base radial es una clase de red neuronal artificial que tiene muchas ventajas sobre otras redes. Para funcionar con precisión, hace uso de algunos parámetros y la selección precisa de estos parámetros aumenta su rendimiento. Munnoli, S. P., & Bapat, A. U. (2013).

		red simple y aprendizaje más rápido.	agrupamiento difuso para la selección de parámetros del centro RBFNN.	
2013	Review on determining number of Cluster in K-Means Clustering	Hay muchos métodos disponibles para estimar el número de conglomerados, como índices estadísticos, método basado en la varianza, teoría de la información, método de bondad de ajuste, etc. El artículo explora seis enfoques diferentes para determinar el número correcto de conglomerados en un conjunto de datos.	El algoritmo de agrupación en clústeres de k-medias es el más utilizado debido a su simplicidad. Nos centramos en seis enfoques diferentes: i) Por regla general; ii) método del codo; iii) Enfoque basado en criterios de información; iv) Un enfoque teórico de la información; v) Elegir k Usar la silueta y vi) Validación cruzada	Este documento aborda uno de los temas más controvertidos en la agrupación: el número correcto de agrupaciones, que algunos pueden considerar infundado porque, en muchos casos, "las agrupaciones no están en los datos sino en el ojo del espectador". Kodinariya, T. M., & Makwana, P. R. (2013).
2014	Propuesta de procedimiento para configurar una neuronal artificial de base radial con	Se presenta un procedimiento que permite la configuración de los parámetros de una arquitectura de red neuronal artificial de base radial para	El procedimiento garantiza la obtención del número de neuronas ocultas a partir de fijar el error de diagnóstico deseado para cada proceso, permitiendo	Se concluye que la elección de la función de distancia no influye sobre la selección del método para estimar los anchos. Se comprueba que esta

	aplicaciones en el diagnóstico de fallos.	tareas de diagnóstico de fallos, luego de establecer un orden lógico para la selección de estos.	seleccionar el método para estimar los anchos de las neuronas ocultas y la ecuación de distancia para la propagación del espacio de entradas.	arquitectura, con el entrenamiento adecuado, exhibe buenas propiedades para el diagnóstico de fallos. Barrero, E., Fernández, B., y Llanes, O. (2014).
2014	EBK-Means: A clustering Technique based on Elbow Method and K-Means in WSN.	Sensor de redes inalámbricas (WSN) consta de miles de nodos. Los nodos sensores se utilizan para detectar parámetros ambientales. El agrupamiento es el proceso de creación de subgrupos virtuales de los nodos sensores, ayuda a reducir los cálculos de enrutamiento.	En el método propuesto, encontrar un valor óptimo de 'k' se realiza mediante el método Elbow y la agrupación se realiza mediante el algoritmo k-means, de ahí el enrutamiento protocolo LEACH.	El proyecto de investigación ha sido implementado en MATLAB. Los resultados han demostrado que ha sido eficiente el esquema de enrutamiento. La agrupación mejora la vida útil de la red. Bholowalia, P., & Kumar, A. (2014).
2018	Integration K-Means Clustering Method and Elbow Method For Identification of	PYMES Indonesia tiene una variedad de clientes, pero las PYMES no tienen el mapeo de estos clientes, por lo que no saben qué clientes son	Los investigadores utilizarán una combinación del método K-Means y el algoritmo elbow para mejorar el rendimiento de k-medias y sea eficiente y eficaz en	Los datos de prueba utilizaron agrupaciones de k-medias y utilizan el método Elbow en 100 y 300 perfiles de clientes.

	The Best Customer Profile Cluster.	leales o no. El mapeo de clientes es una agrupación de perfiles de clientes para facilitar el análisis y la política de las pymes en la producción de bienes, especialmente las ventas de batik.	el procesamiento de grandes cantidades de datos. El resultado de determinar el mejor número de conglomerados con el método codo será el predeterminado para el proceso característico basado en el estudio de caso.	Los resultados de los cálculos de la Suma del Error Cuadrado de cada clúster que han experimentado la mayor disminución en $K = 3$. Syakur, M. A., Khotimah, B. K., Rochman, E. M. S., & Satoto, B. D. (2018, April).
2020	Abusive Text Examination Using Latent Dirichlet Allocation, Self Organizing Maps and K Means Clustering	El uso indebido generalizado de las redes sociales para difundir discursos de odio dirigidos a un individuo, comunidad, religión, raza, sexo o casta en particular ha motivado a investigadores de todo el mundo a formular estrategias y metodologías para contrarrestar esta amenaza.	Propone una metodología que emplea una combinación de la técnica de modelado de temas populares, es decir, la asignación de Dirichlet latente (LDA) y una técnica de aprendizaje automático no supervisada, es decir, mapas autoorganizados (SOM). Este método se compara con el agrupamiento de K-medias usado después de la aplicación de LDA.	Para ejecutar el objetivo anterior, se utiliza una combinación de una técnica de modelado de temas llamada asignación de dirichlet latente y mapas autoorganizados que deriva diez temas del corpus del conjunto de datos, asigna el tema dominante a cada valor de datos y luego a un mapa para agruparlos. También se representan varios clústeres

				mediante el uso de clústeres de K medias. Saini, Y., Bachchas, V., Kumar, Y., & Kumar, S. (2020, May).
2020	Optimización de las rutas para la intervención de pozos de petróleo	El objetivo es proponer un programa óptimo de intervención de pozos petroleros mediante la reasignación de rutas de los equipos SWAB en empresas dedicadas a estas operaciones.	La primera fase hace uso del método de agrupamiento utilizando k-medias con el propósito de obtener un número de agrupamientos. La segunda fase utiliza algoritmos genéticos (AG) para encontrar el recorrido óptimo en cada agrupamiento de la primera fase y optimizar el número de agrupamientos.	La implementación del algoritmo en dos fases (clustering y AG) es una solución para problemas de planificación de rutas en transporte con las características del proceso swab. Jiménez-Carrión, M., Sánchez-Candela, L., Keewong-Zapata, R., & Bazán, J. (2020).
2020	A Method for Establishing Tropospheric Atmospheric Refractivity Profile	Se propone un método de modelado que utiliza la función de base radial (RBF) multicuadrada (MQ) y la agrupación de k-medias para	Los TAR en 132 estaciones meteorológicas en China se obtienen utilizando los datos meteorológicos de diez años que se midieron mediante el método	En este trabajo se propone un método para establecer el modelo de perfil TAR basado en el algoritmo MQ-RBF, y los coeficientes para la corrección

	Model Based on Multiquadric RBF and k-means Clustering	establecer el modelo de perfil de refractividad atmosférica troposférica (TAR).	de sondeo con globos. (1) Generación del conjunto de datos de entrenamiento, (2) Predicción de los coeficientes TAR, (3) Optimización de regresión lineal, (4) Establecimiento del perfil del modelo TAR.	del error de refracción de onda se obtienen con el método de regresión lineal de mínimos cuadrados optimizado por el algoritmo de agrupamiento de k-medias. Ma, T., Liu, H., & Zhang, Y. (2020).
2020	Energy efficient building envelope using novel RBF neural network integrated affinity propagation	Este artículo presenta una nueva función de base radial (RBF) basada en la agrupación de propagación por afinidad (AP) para evaluar el rendimiento energético y ahorrar energía de los edificios.	El AP obtiene el número de nodos de la capa oculta del RBF. Luego, los principales factores que afectan el consumo de energía de los edificios se utilizan como entradas y salidas del RBF para construir el modelo de ahorro y rendimiento energético de los edificios.	Se puede predecir el consumo de energía de los edificios en diferentes condiciones y se puede mejorar el plan de construcción para construir edificios más eficientes energéticamente. Han, Y., Fan, C., Geng, Z., Ma, B., Cong, D., Chen, K., & Yu, B. (2020).
2020	Generalized Load Modeling Method Based on	Este artículo propone un método de modelado de carga generalizado utilizando	Las contribuciones son las siguientes: de acuerdo con las características variables en el tiempo de la carga generalizada,	Los resultados del caso de estudio han mostrado el proceso detallado y el análisis del modelo de carga

	Clustering and Neural Network	agrupación de K-medias y red neuronal RBF.	las curvas de carga se agrupan para usar y simplificar los datos de manera efectiva, lo que proporciona soporte de datos críticos para el siguiente paso del análisis	generalizada, así como las características que varían en el tiempo y los patrones de consumo con carga generalizada. Li, W., Xing, L., Huang, K., Ai, Q., & Wang, L. (2020, February).
2020	Modelo inteligente para monitoreo del desarrollo de plantas de maíz	Se construyó una red neuronal artificial de aprendizaje híbrido capaz de monitorear el desarrollo del cultivo e identificar posibles deficiencias de forma oportuna con una precisión del 98%.	La primera fase de aprendizaje de la RNA propuesta, se realizó empleando el algoritmo K medias, para la clasificación de patrones de entrada a la red en 6 grupos en la 2da fase se empleó el algoritmo de Windrow y Hoff para la obtención de la matriz de aprendizaje integrada por los pesos sinápticos.	La RNA es adecuada para el monitoreo del desarrollo de la planta de maíz, el modelo propuesto permite evaluar cualquiera de las etapas del cultivo bajo efectos de los factores climáticos expuestos como variables de entrada a la RNA. Gutiérrez, S. S. M., Álvarez, J. C., Olgún, E. A. G., Munguía, M. G., & Pérez, S. L. (2020).

<p>2020</p>	<p>Breast cancer clustering using modified spherical K-Means</p>	<p>En este artículo, utilizamos k-medias esféricas modificadas mediante el uso de la función de base radial del núcleo (RBF) mediante medidas de producto interno en k-medias esféricas para agrupar el conjunto de datos de Coimbra de cáncer de mama del aprendizaje automático UCI en grupos.</p>	<p>En este artículo, utilizamos la mejora del algoritmo de k-medias esféricas agregando un núcleo de función de base radial (RBF) y un algoritmo de k-medias esféricas para el algoritmo de comparación. La adición de la función del núcleo en k-medias esféricas tiene como objetivo obtener una mayor precisión y eficiencia. En este documento, usamos el kernel RBF con $\sigma = 0.001$.</p>	<p>El conjunto de datos de Coimbra de cáncer de mama tiene 116 instancias y 10 atributos. El tiempo de ejecución más rápido con Kernel Spherical K-Means es de 0,89 segundos en un entrenamiento de datos del 20% y 60% y produce la mayor precisión del 72,41%. Este método puede ayudar al personal médico a clasificar para predecir fácilmente los datos del cáncer de mama. Rustam, Z., & Fijri, A. L. (2020, March).</p>
<p>2020</p>	<p>Dynamic performance enhancement for nonlinear stochastic systems</p>	<p>Se propone un método de control híbrido novedoso para mejorar el desempeño de seguimiento del sistema de control basado en</p>	<p>La señal compensadora se forma utilizando los estados estimados del filtro de Kalman extendido (EKF) y una compensación no lineal realizada por la red</p>	<p>El compensador utiliza el estado estimado como entrada y es impulsado por una función no lineal genérica realizada por una red neuronal RBF.</p>

	<p>using RBF driven nonlinear compensation with extended Kalman filter</p>	<p>Proporcional-Integral (PI) para una clase de procesos dinámicos estocásticos no lineales y no gaussianos con estados no medibles</p>	<p>neuronal de función de base radial (RBF). Los pesos de RBF están entrenados para minimizar la entropía del error de seguimiento de circuito cerrado.</p>	<p>Utilizando el concepto de entropía como criterio, se han ajustado los pesos óptimos en RBF y se ha atenuado la aleatoriedad del error de seguimiento de bucle cerrado. Zhou, Y., Wang, A., Zhou, P., Wang, H., & Chai, T. (2020).</p>
--	--	---	---	--

CAPÍTULO 4: METODOLOGÍA

La figura 23 muestra el diagrama de la metodología

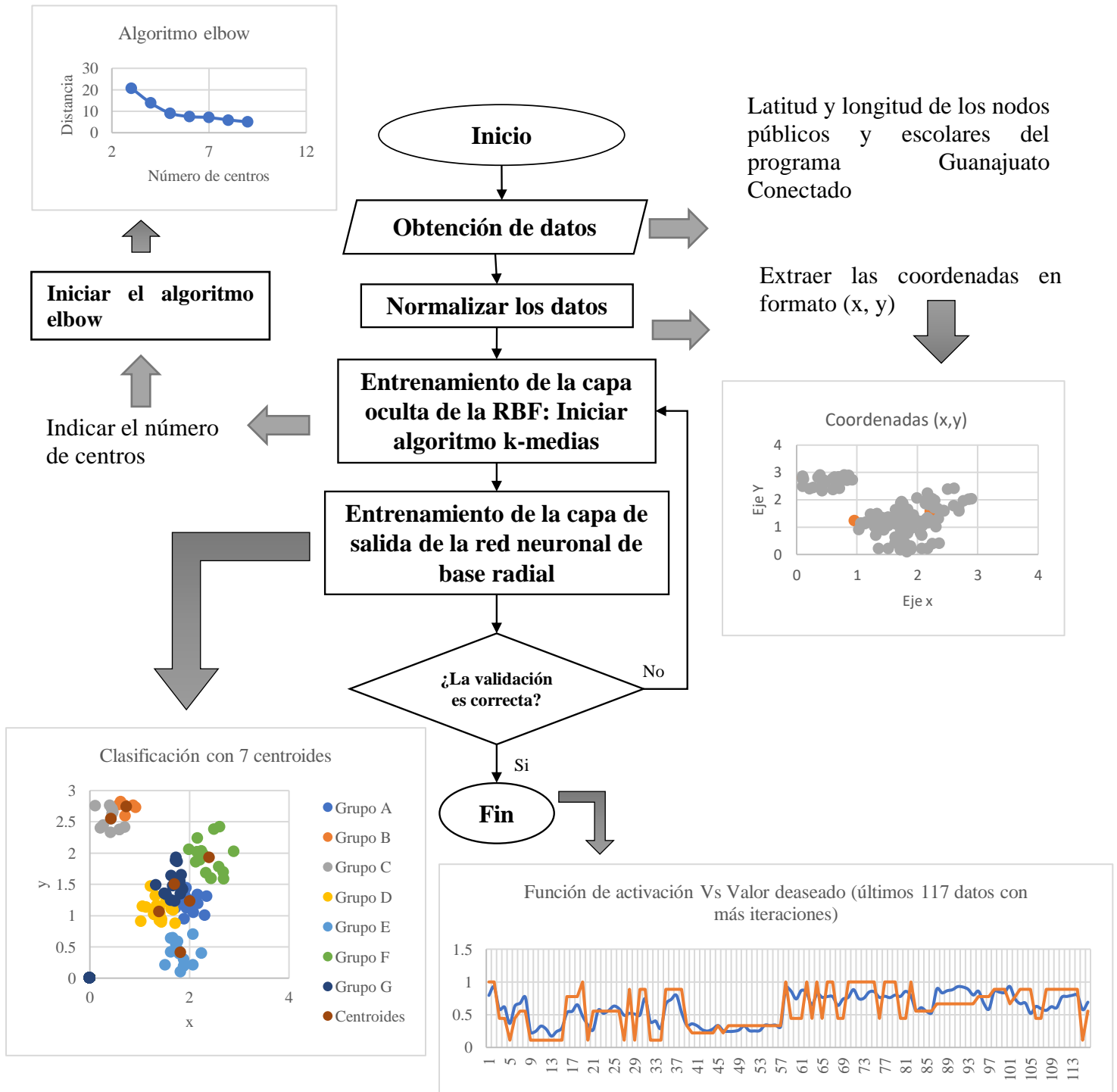


Figura 23 Diagrama de la metodología

CAPÍTULO 5: DESARROLLO DE LA METODOLOGÍA

5.1 OBTENCIÓN DE DATOS

El sistema de coordenadas geográficas de los puntos en esta investigación fue tomado del Programa de reducción de brecha digital (PRBD) del estado de Guanajuato. Para ello utilizamos dos coordenadas angulares, latitud (norte o sur) y longitud (este u oeste) de modo que cualquier nodo público o escolar de la superficie terrestre pueda ser referenciado (ver Tabla 1 y 2).

Los datos contenidos en la Tabla 1 y 2 no serían procesados directamente, para introducirlos a la red neuronal de base radial se normalizaron a valores entre el rango 0.1 y 2.9.

Se realizó una distribución de datos de la siguiente manera: 70% de los datos se utilizaron para el entrenamiento, 20 % para la validación y 10% para las pruebas (ver Figura 24).

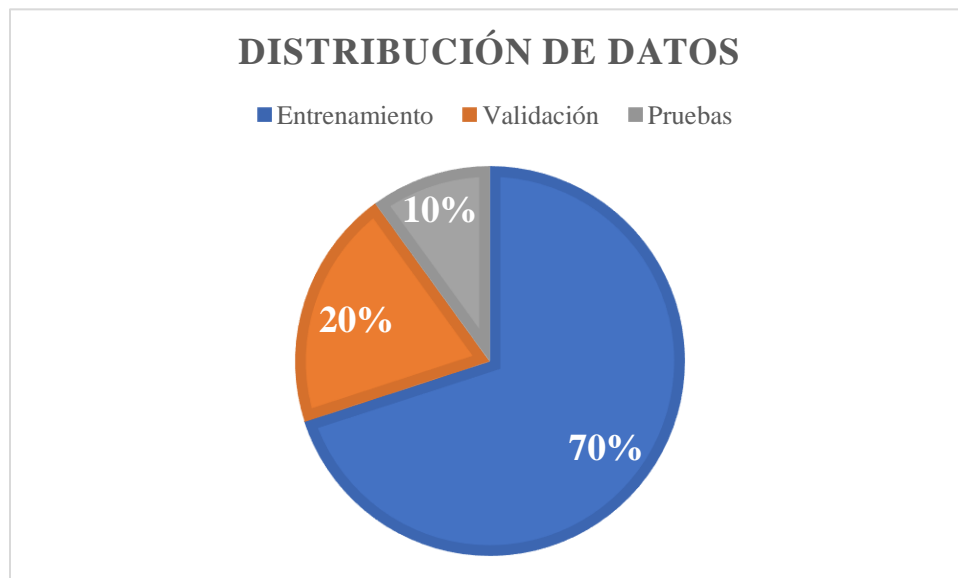


Figura 24 Distribución de datos

5.2 NORMALIZACIÓN DE DATOS

Para la normalización de datos, se trasladaron las coordenadas angulares latitud y longitud al plano en dos dimensiones abscisas y ordenadas, lo que permitirá graficarlos e iniciar con la entrada de datos a la red neuronal de base radial.

Ajuste de datos

Los puntos se colocan en dos columnas, de modo que en cada fila figuren la abscisa x y su ordenada y . Utilizamos un sistema de coordenadas cartesianas para representar la distribución bidimensional, obtendremos un conjunto de puntos conocido como diagrama de dispersión. Se realizó el siguiente procedimiento:

1. Determinación de la dependencia funcional entre las dos variables x e y en una distribución bidimensional. Se requiere la determinación de dos parámetros: la pendiente y la ordenada en el origen de la recta de regresión, $y = mx + b$.
2. Para la coordenada angular longitud. Encontrar la pendiente “ m ” y la ordenada “ b ” de acuerdo con la siguiente ecuación. Se utilizó el rango entre 0.1 y 2.9. Los resultados se muestran en la Tabla 4. Abscisas

$$m = \frac{y - y_1}{x - x_1} = \frac{2.9 - 0.1}{x_{max} - x_{min}} = \frac{2.9 - 0.1}{-100.51479 - (-101.76388)} = 2.2416$$

$$b = y - mx = 0.1 - (m * x_{min}) = 0.1 - (2.24163191 * (-101.76388)) = 228.217$$

$$y = mx + b = 2.2416(x) + 228.217$$

3. Para la coordenada angular latitud. Encontrar la pendiente “ m ” y la ordenada “ b ” de acuerdo con la siguiente ecuación. Se utilizó el rango entre 0.1 y 2.9. Los resultados se muestran en la Tabla 5. Ordenadas

$$m = \frac{y - y_1}{x - x_1} = \frac{2.9 - 0.1}{x_{max} - x_{min}} = \frac{2.9 - 0.1}{21.22037 - (20.172432)} = 2.6719138$$

$$b = y - mx = 0.1 - (m * x_{min}) = 0.1 - (2.6719138 * (20.172432)) = -53.798999$$

$$y = mx + b = 2.6719(x) - 53.7989$$

4. Calcular la variable dependiente y, dejando como constante el valor de la pendiente m y la ordenada b para cada valor x de la coordenada angular.

Tabla 4. Abscisas

Municipio	Nodo	(x)	y
	Comunidad	Longitud	Eje X
Celaya	Pedro María / Patria Nueva	-100.87815400	2.085471663
Celaya	Roque	-100.83836900	2.174654989
Celaya	San Elías	-100.84150800	2.167618506
Celaya	San Juan de la Vega	-100.76048900	2.349233282
Celaya	Parque Lineal	-100.77846000	2.308948915
Celaya	Gasca	-100.85686200	2.13320049
Celaya	Yustis	-100.86416400	2.116832094
Celaya	Ejido de Silva	-100.83604500	2.179864541
Celaya	San Cayetano	-100.82086100	2.21390148
Celaya	San isidro de Elguera	-100.88710300	2.065411299
Celaya	San Elías	-100.83856300	2.174220112
Comonfort	Jalpilla	-100.71975000	2.440555124
Comonfort	Jalpilla	-100.72132000	2.437035762
Comonfort	San Isidro de San Jeronimo	-100.76590900	2.337083637
Comonfort	San Pedro	-100.78924000	2.284784123
Cortazar	La Minilla	-100.89288200	2.052456909
Cortazar	El Diezmo	-101.01592500	1.776639794
Cortazar	Tierra Fría	-101.03980300	1.723114107
Cortazar	Galeras	-100.98684000	1.841837658
Cortazar	Centro Cultural	-100.96201300	1.897490653
Cortazar	La Gavia	-100.88368900	2.073064231
Jaral de Progreso	Santiago Capitiro	-101.02067500	1.765992042
Jaral de Progreso	Providencia	-101.02103500	1.765185055
Jaral de Progreso	Victoria de Cortázar	-101.03391900	1.736303869
Jaral de Progreso	DIF Municipal	-101.08127900	1.630140182
Jaral de Progreso	Jardín Municipal	-101.06396700	1.668947314
Juventino Rosas	Santiago de Cuenda	-100.99485400	1.82387322
Juventino Rosas	COMUDE SUR	-100.99298100	1.828071796

Juventino Rosas	Valencia	-100.96313100	1.894984509
Juventino Rosas	Dulces Nombres	-100.96494800	1.890911464
Juventino Rosas	San Diego de los Dolores	-101.02843400	1.74859922
Juventino Rosas	Franco Tavera	-100.92880000	1.971941974
Juventino Rosas	San Antonio de Romerillo	-100.94800000	1.928902641
Juventino Rosas	San Antonio de Morales	-100.90960000	2.014981306
Juventino Rosas	El Naranjillo	-101.00097600	1.810149949
Juventino Rosas	Jaralillo	-100.97680000	1.864343642
Juventino Rosas	San Antonio de las Maravillas	-100.89469000	2.048404038
Juventino Rosas	Cerrito de Gasca	-101.06116000	1.675239574
Juventino Rosas	Romero	-100.98700000	1.841478997
Juventino Rosas	Mesas de Acosta	-101.07652000	1.640808108
Juventino Rosas	La Peña	-100.94534000	1.934865382
León	San Francisco del Duran	-101.62169200	0.418733158
León	Vaquerías	-101.39523000	0.926377603
León	Derramadero	-101.41692000	0.877756607
León	San Rafael Cerro Verde	-101.49721000	0.697775981
León	Nuevo Valle de Moreno	-101.42442000	0.860944367
León	Mesas de Obispo	-101.43512000	0.836958906
León	Las Canelas	-101.45616000	0.789794971
León	San José de Otates Norte	-101.49721000	0.697775981
León	San José de Otates Sur	-101.49336000	0.706406264
León	Sauz Seco	-101.54277000	0.595647231
León	Cuesta Blanca	-101.49065000	0.712481086
León	San José de los Romeros	-101.49197000	0.709522132
León	Los Alisos	-101.52930000	0.625842013
León	Alfaro	-101.60730000	0.450994724
León	San José del Potrero	-101.59938000	0.468748449
León	León Urbano	-101.61801000	0.426986846
León	Los Sauces	-101.53926000	0.603515359
León	Los Naranjos	-101.62670000	0.407507065
León	Alfaro	-101.60810000	0.449201419
León	León Urbano	-101.60476000	0.456688469
León	Fraccionamiento Paseo de las Torres	-101.75621000	0.117193317
León	La Mora	-101.76323000	0.101457061
León	Ibarrilla	-101.64995000	0.355389123
León	Lagunillas	-101.76388000	0.1
León	León Primaria	-101.76312000	0.10170364
León	Mesa de Ibarrilla	-101.63424000	0.390605161
León	Los Sauces	-101.54116000	0.599256259
León	San Pedro del Monte	-101.71161000	0.2171701

León	Los Arcos	-101.68482000	0.277223419
Pueblo Nuevo	Congragación de Panales	-101.33472000	1.06201875
Pueblo Nuevo	Durazno de Fonseca	-101.31231000	1.112253721
Salamanca	Congregación de Cárdenas	-101.21890100	1.321642316
Salamanca	Cerro Gordo	-101.12612100	1.529620924
Salamanca	Loma de Flores	-101.26212500	1.224750018
Salamanca	El 4 de Altamira	-101.16582200	1.440625896
Salamanca	Valtierrilla	-101.12972600	1.521539841
Salamanca	San José de Mendoza	-101.21592400	1.328315654
Salamanca	Unión de Liebres	-101.18253900	1.403152535
Salamanca	Recuerdo de Ancón	-101.12959300	1.521837978
Salamanca	Santo Domingo	-101.22790300	1.301463145
Salamanca	Valencia	-101.08231300	1.627822335
Salamanca	Ecoparque	-101.20727900	1.347694562
Salamanca	Los Miranda	-101.13816000	1.502633917
Salamanca	La Luz	-101.18072000	1.407230063
Salamanca	Uruétaro	-101.18027000	1.408238798
Salamanca	Valencia de Cerro Gordo	-101.08022000	1.63251407
Salamanca	El Xoconostle	-101.12524000	1.531595802
Salamanca	Uruétaro	-101.18116000	1.406243745
Salamanca	La Capilla	-101.30033000	1.139108471
Salamanca	Colonia el 4 de Altamira	-101.16745000	1.436976519
Salamanca	Colonia el 4 de Altamira	-101.16561000	1.441101122
Salamanca	San Juan de Razos	-101.23160000	1.293175832
Salamanca	Loma Pelada	-101.29796300	1.144414414
Salamanca	San Rafael de Uruetaro	-101.17831000	1.412632396
Salamanca	Santo Domingo	-101.22903000	1.298936826
Salamanca	Santo Domingo	-101.23139000	1.293646575
Salamanca	El Xoconostle	-101.12582000	1.530295655
Salamanca	Cuatro de Altamira	-101.18148000	1.405526423
Salamanca	Colonia el Cuatro de Altamira	-101.16368000	1.445427471
Salamanca	Uruetaro	-101.18081000	1.407028317
Salamanca	El Recuerdo de Ancón (Xoconostle de Arriba)	-101.12956000	1.521911952
Salamanca	Cerro Gordo	-101.12701000	1.527628113
Salamanca	El Circuito	-101.18305000	1.402007061
Salamanca	San Jose de Mendoza	-101.21538600	1.329521652
Salamanca	El Circuito	-101.18390000	1.400101674
Salvatierra	El Sabino	-101.00659500	1.797554219
Salvatierra	Urireo	-100.84251300	2.165365666
Salvatierra	Maravatío del Encinal	-100.96522200	1.890297256
Salvatierra	San Nicolás de los Agustinos	-100.96611100	1.888304446

Salvatierra	Centro Histórico	-100.88270200	2.075276721
San Miguel de Allende	Don Juan	-100.80634000	2.246452217
San Miguel de Allende	Corral de Piedras de Abajo	-100.82233000	2.210608523
San Miguel de Allende	Agustín Gonzalez	-100.85310000	2.141633509
San Miguel de Allende	Agustín Gonzalez	-100.85279000	2.142328415
San Miguel de Allende	Pantoja	-100.78897000	2.285389363
San Miguel de Allende	Tlaxcalilla	-100.84462000	2.160642548
San Miguel de Allende	Los Rodríguez	-100.64294700	2.61271918
San Miguel de Allende	Cruz del Palmar	-100.84267500	2.165002522
San Miguel de Allende	Don Francisco	-100.91847300	1.995091306
San Miguel de Allende	Alonso Yáñez	-100.85560200	2.136024946
San Miguel de Allende	La Huerta Repetidor	-100.83408000	2.184269348
San Miguel de Allende	La Huerta	-100.82938200	2.194800535
San Miguel de Allende	Guadalupe de Tambula	-100.57565000	2.763574282
San Miguel de Allende	Puerto de Nieto	-100.53816000	2.847613062
San Miguel de Allende	Moral de Puerto de Nieto	-100.51479000	2.9
San Miguel de Allende	Jalpa	-100.61162000	2.682942782
San Miguel de Allende	Charco de Sierra	-100.60750000	2.692178306
San Miguel de Allende	San José de Allendes	-101.03344000	1.737377611
San Miguel de Allende	San Isidro de Ensaye (La Cuevita)	-101.03772000	1.727783426
San Miguel de Allende	El Nuevo Cimatarío	-101.02277000	1.761295823
San Miguel de Allende	La Biznaga	-100.69224000	2.502222418
San Miguel de Allende	Lagunillas	-100.64883000	2.599531659
Santa Cruz de Juventino Rosas	El Naranjillo	-101.00096200	1.810181332
Santa Cruz de Juventino Rosas	El Naranjillo	-101.00007900	1.812160693
Santa Cruz de Juventino Rosas	UPJR	-100.96410100	1.892810126
Santa Cruz de Juventino Rosas	La Trinidad	-101.06333000	1.670375233
Santa Cruz de Juventino Rosas	San Julian Tierra Blanca	-101.04333000	1.715207871
Santa Cruz de Juventino Rosas	Jaralillo	-100.97611000	1.865890368
Santa Cruz de Juventino Rosas	Emiliano Zapata	-101.04234000	1.717427087
Santa Cruz de Juventino Rosas	Los Dulces Nombres	-100.96188000	1.89778879
Santa Cruz de Juventino Rosas	San Antonio Nuevo Valencia	-100.98833000	1.838497626
Santa Cruz de Juventino Rosas	La Tinaja	-101.03804000	1.727066104
Santa Cruz de Juventino Rosas	San Jose del Sauz	-100.98429000	1.847553819
Santa Cruz de Juventino Rosas	Jaralillo	-100.97585000	1.866473192
Santa Cruz de Juventino Rosas	La Tinaja	-101.03804000	1.727066104
Santa Cruz de Juventino Rosas	El Carmen de Arriba	-101.06271000	1.671765045
Santa Cruz de Juventino Rosas	San Diego de los Dolores	-101.03166700	1.741352024
Santa Cruz de Juventino Rosas	San Diego de los Dolores	-101.02849400	1.748464722
Santa Cruz de Juventino Rosas	Los Dulces Nombres	-100.96122700	1.899252576
Santiago Maravatio	Jardín Municipal	-100.99501100	1.823521283

Tarimoro	La Moncada	-100.80551700	2.24829708
Tarimoro	Centro Histórico	-100.75560500	2.360181412
Valle de Santiago	Rancho Cuatro de Altamira	-101.18115000	1.406266162
Valle de Santiago	UTSOE	-101.22219300	1.314262863
Valle de Santiago	Alto de Altamira	-101.16312000	1.446682785
Valle de Santiago	Santa Barbara	-101.34905900	1.02987599
Valle de Santiago	COMUDE	-100.99298100	1.828071796
Villagran	Suchitlan	-101.03632200	1.730917228
Villagran	San Isidro Cerro Gordo	-101.05927700	1.679460567
Villagran	San Salvador Torrecillas	-100.95658600	1.90965599
Villagran	Col.18 de marzo	-101.03632200	1.730917228
Villagran	Los Ángeles	-100.94134700	1.943816218
Villagran	Parque Agro-Industrial XONOTLI	-101.07575100	1.642531923
Yuriria	Loma de Zempoala	-101.08067500	1.631494128
Yuriria	Casacuaran	-101.04231300	1.717487611
Yuriria	San Francisco de la Cruz	-101.20112000	1.361500773
Yuriria	Centro Histórico	-101.13164500	1.517238149

Tabla 5. Ordenadas

Municipio	Nodo	(x)	y
	Comunidad	Latitud	Eje Y
Celaya	Pedro María / Patria Nueva	20.52794	1.049887
Celaya	Roque	20.581634	1.193352
Celaya	San Elías	20.633491	1.331910
Celaya	San Juan de la Vega	20.624287	1.307318
Celaya	Parque Lineal	20.511533	1.006049
Celaya	Gasca	20.60445	1.254315
Celaya	Yustis	20.581375	1.192660
Celaya	Ejido de Silva	20.566268	1.152296
Celaya	San Cayetano	20.603806	1.252594
Celaya	San isidro de Elguera	20.563705	1.145448
Celaya	San Elías	20.631236	1.325885
Comonfort	Jalpilla	20.73049	1.591083
Comonfort	Jalpilla	20.7328	1.597255
Comonfort	San Isidro de San Jerónimo	20.763667	1.679729
Comonfort	San Pedro	20.77533	1.710891
Cortázar	La Minilla	20.422871	0.769151
Cortázar	El Diezmo	20.428459	0.784082
Cortázar	Tierra Fría	20.461384	0.872055
Cortázar	Galeras	20.426612	0.779147
Cortázar	Centro Cultural	20.488899	0.945573

Cortázar	La Gavia	20.397251	0.700697
Jaral de Progreso	Santiago Capitiro	20.304513	0.452909
Jaral de Progreso	Providencia	20.35469	0.586978
Jaral de Progreso	Victoria de Cortázar	20.335613	0.536006
Jaral de Progreso	DIF Municipal	20.372157	0.633648
Jaral de Progreso	Jardín Municipal	20.37461	0.640202
Juventino Rosas	Santiago de Cuenda	20.598988	1.239721
Juventino Rosas	COMUDE SUR	20.637948	1.343819
Juventino Rosas	Valencia	20.644062	1.360155
Juventino Rosas	Dulces Nombres	20.650485	1.377316
Juventino Rosas	San Diego de los Dolores	20.726128	1.579428
Juventino Rosas	Franco Tavera	20.6063	1.259258
Juventino Rosas	San Antonio de Romerillo	20.60756	1.262625
Juventino Rosas	San Antonio de Morales	20.58156	1.193155
Juventino Rosas	El Naranjillo	20.714818	1.549209
Juventino Rosas	Jaralillo	20.66503	1.416179
Juventino Rosas	San Antonio de las Maravillas	20.58287	1.196655
Juventino Rosas	Cerrito de Gasca	20.61717	1.288302
Juventino Rosas	Romero	20.752	1.648556
Juventino Rosas	Mesas de Acosta	20.74748	1.636479
Juventino Rosas	La Peña	20.6755	1.444154
León	San Francisco del Duran	21.006077	2.327428
León	Vaquerías	21.15663	2.729692
León	Derramadero	21.16696	2.757293
León	San Rafael Cerro Verde	21.19078	2.820938
León	Nuevo Valle de Moreno	21.21237	2.878625
León	Mesas de Obispo	21.14766	2.705725
León	Las Canelas	21.21906	2.896500
León	San José de Otates Norte	21.19078	2.820938
León	San José de Otates Sur	21.16791	2.759831
León	Sauz Seco	21.17207	2.770947
León	Cuesta Blanca	21.10638	2.595429
León	San José de los Romeros	21.03744	2.411227
León	Los Alisos	21.19096	2.821419
León	Alfaro	21.14697	2.703882
León	San José del Potrero	21.12927	2.656589
León	León Urbano	21.08529	2.539078
León	Los Sauces	21.02413	2.375664
León	Los Naranjos	21.16928	2.763492
León	Alfaro	21.14869	2.708477
León	León Urbano	21.11748	2.625087

León	Fraccionamiento Paseo de las Torres	21.16689	2.757106
León	La Mora	21.06612	2.487857
León	Ibarrilla	21.18679	2.810277
León	Lagunillas	21.20174	2.850222
León	León Primaria	21.16634	2.755636
León	Mesa de Ibarrilla	21.22037	2.900000
León	Los Sauces	21.0258	2.380126
León	San Pedro del Monte	21.03328	2.400112
León	Los Arcos	21.04922	2.442702
Pueblo Nuevo	Congragación de Panales	20.56314	1.143938
Pueblo Nuevo	Durazno de Fonseca	20.55292	1.116631
Salamanca	Congregación de Cárdenas	20.627029	1.314644
Salamanca	Cerro Gordo	20.593413	1.224825
Salamanca	Loma de Flores	20.683629	1.465874
Salamanca	El 4 de Altamira	20.485611	0.936787
Salamanca	Valtierrilla	20.52851	1.051410
Salamanca	San José de Mendoza	20.690796	1.485024
Salamanca	Unión de Liebres	20.642489	1.355952
Salamanca	Recuerdo de Ancón	20.638631	1.345644
Salamanca	Santo Domingo	20.515328	1.016189
Salamanca	Valencia	20.602204	1.248314
Salamanca	Ecoparque	20.567796	1.156379
Salamanca	Los Miranda	20.6401	1.349569
Salamanca	La Luz	20.55521	1.122750
Salamanca	Uruétaro	20.51961	1.027630
Salamanca	Valencia de Cerro Gordo	20.60014	1.242799
Salamanca	El Xoconostle	20.63728	1.342034
Salamanca	Uruétaro	20.5161	1.018251
Salamanca	La Capilla	20.55888	1.132556
Salamanca	Colonia el 4 de Altamira	20.4848	0.934620
Salamanca	Colonia el 4 de Altamira	20.48533	0.936036
Salamanca	San Juan de Razos	20.55272	1.116097
Salamanca	Loma Pelada	20.571437	1.166107
Salamanca	San Rafael de Uruetaro	20.50633	0.992147
Salamanca	Santo Domingo	20.51569	1.017156
Salamanca	Santo Domingo	20.51706	1.020816
Salamanca	El Xoconostle	20.63725	1.341954
Salamanca	Cuatro de Altamira	20.47996	0.921688
Salamanca	Colonia el Cuatro de Altamira	20.48589	0.937533
Salamanca	Uruetaro	20.51808	1.023542
Salamanca	El Recuerdo de Ancón (Xoconostle de Arriba)	20.63895	1.346496

Salamanca	Cerro Gordo	20.59916	1.240180
Salamanca	El Circuito	20.50918	0.999762
Salamanca	San José de Mendoza	20.690968	1.485483
Salamanca	El Circuito	20.50971	1.001178
Salvatierra	El Sabino	20.281045	0.390205
Salvatierra	Urireo	20.216442	0.217591
Salvatierra	Maravatío del Encinal	20.205091	0.187262
Salvatierra	San Nicolás de los Agustinos	20.246335	0.297462
Salvatierra	Centro Histórico	20.212867	0.208039
San Miguel de Allende	Don Juan	20.89607	2.033498
San Miguel de Allende	Corral de Piedras de Abajo	20.84282	1.891219
San Miguel de Allende	Agustín González	20.85922	1.935038
San Miguel de Allende	Agustín González	20.85807	1.931966
San Miguel de Allende	Pantoja	20.87324	1.972499
San Miguel de Allende	Tlaxcalilla	20.88913	2.014955
San Miguel de Allende	Los Rodríguez	21.04082	2.420258
San Miguel de Allende	Cruz del Palmar	20.971912	2.236142
San Miguel de Allende	Don Francisco	20.905315	2.058200
San Miguel de Allende	Alonso Yáñez	20.829893	1.856679
San Miguel de Allende	La Huerta Repetidor	20.81958	1.829124
San Miguel de Allende	La Huerta	20.821489	1.834224
San Miguel de Allende	Guadalupe de Tambula	20.86864	1.960208
San Miguel de Allende	Puerto de Nieto	20.89068	2.019097
San Miguel de Allende	Moral de Puerto de Nieto	20.89309	2.025536
San Miguel de Allende	Jalpa	20.76891	1.693738
San Miguel de Allende	Charco de Sierra	20.72778	1.583842
San Miguel de Allende	San José de Allendes	20.85651	1.927797
San Miguel de Allende	San Isidro de Ensaye (La Cueva)	20.83845	1.879543
San Miguel de Allende	El Nuevo Cimatarío	20.83181	1.861801
San Miguel de Allende	La Biznaga	21.02538	2.379004
San Miguel de Allende	Lagunillas	20.80138	1.780495
Santa Cruz de Juventino Rosas	El Naranjillo	20.715255	1.550376
Santa Cruz de Juventino Rosas	El Naranjillo	20.713992	1.547002
Santa Cruz de Juventino Rosas	UPJR	20.654297	1.387502
Santa Cruz de Juventino Rosas	La Trinidad	20.57278	1.169695
Santa Cruz de Juventino Rosas	San Julian Tierra Blanca	20.59556	1.230562
Santa Cruz de Juventino Rosas	Jaralillo	20.66556	1.417596
Santa Cruz de Juventino Rosas	Emiliano Zapata	20.61963	1.294875
Santa Cruz de Juventino Rosas	Los Dulces Nombres	20.65555	1.390850
Santa Cruz de Juventino Rosas	San Antonio Nuevo Valencia	20.6697	1.428657
Santa Cruz de Juventino Rosas	La Tinaja	20.70506	1.523136

Santa Cruz de Juventino Rosas	San José del Sauz	20.67611	1.445784
Santa Cruz de Juventino Rosas	Jaralillo	20.66542	1.417221
Santa Cruz de Juventino Rosas	La Tinaja	20.70506	1.523136
Santa Cruz de Juventino Rosas	El Carmen de Arriba	20.68799	1.477527
Santa Cruz de Juventino Rosas	San Diego de los Dolores	20.726702	1.580962
Santa Cruz de Juventino Rosas	San Diego de los Dolores	20.726265	1.579794
Santa Cruz de Juventino Rosas	Los Dulces Nombres	20.641806	1.354127
Santiago Maravatío	Jardín Municipal	20.172432	0.100000
Tarimoro	La Moncada	20.283137	0.395794
Tarimoro	Centro Histórico	20.288112	0.409087
Valle de Santiago	Rancho Cuatro de Altamira	20.47803	0.916532
Valle de Santiago	UTSOE	20.398449	0.703898
Valle de Santiago	Alto de Altamira	20.47034	0.895984
Valle de Santiago	Santa Barbara	20.475553	0.909913
Valle de Santiago	COMUDE	20.637948	1.343819
Villagran	Suchitlan	20.554805	1.121668
Villagran	San Isidro Cerro Gordo	20.541822	1.086978
Villagran	San Salvador Torrecillas	20.558234	1.130830
Villagran	Col.18 de marzo	20.554805	1.121668
Villagran	Los Ángeles	20.553862	1.119148
Villagran	Parque Agro-Industrial XONOTLI	20.549669	1.107945
Yuriria	Loma de Zempoala	20.291165	0.417244
Yuriria	Casacuaran	20.19831	0.169144
Yuriria	San Francisco de la Cruz	20.213835	0.210625
Yuriria	Centro Histórico	20.214227	0.211673

En la figura 25 podemos observar todos los puntos gráficamente en coordenadas (x, y).

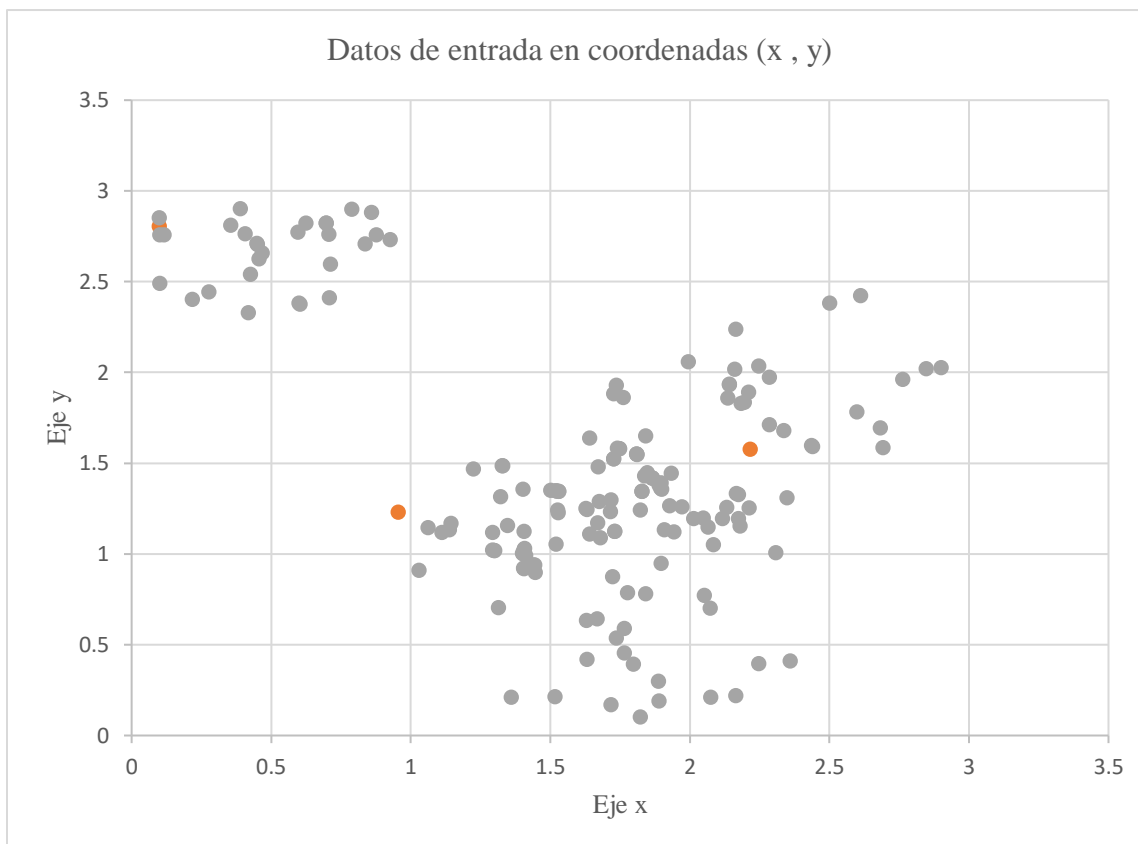


Figura 25 Diagrama de dispersión de los datos de entrada

5.3 ENTRENAMIENTO DE LA CAPA OCULTA DE LA RED NEURONAL DE BASE RADIAL

Antes de iniciar el algoritmo k-medias, se propone encontrar el número de centros aplicando el algoritmo elbow.

Algoritmo del codo o elbow

Se graficó la relación entre el número de centroides y la suma de cuadrados dentro del grupo (WCSS), seleccionando el número de grupos donde el cambio en WCSS empieza a estabilizarse, como se muestra en la figura 26. WCSS se define como la distancia entre cada integrante del grupo y su centroide.

$$WCSS = \sum_{I \in n} (X_i - Y_i)^2$$

Sé registró en la tabla 6 el número de centros vs WCSS, se observa que a partir del número de centroides 7 se estabiliza y en los centros 8 y 9 la variación numérica va siendo menor entre ellos; por lo tanto, se realizaron las pruebas con 7, 8, 9 y hasta 20 centros.

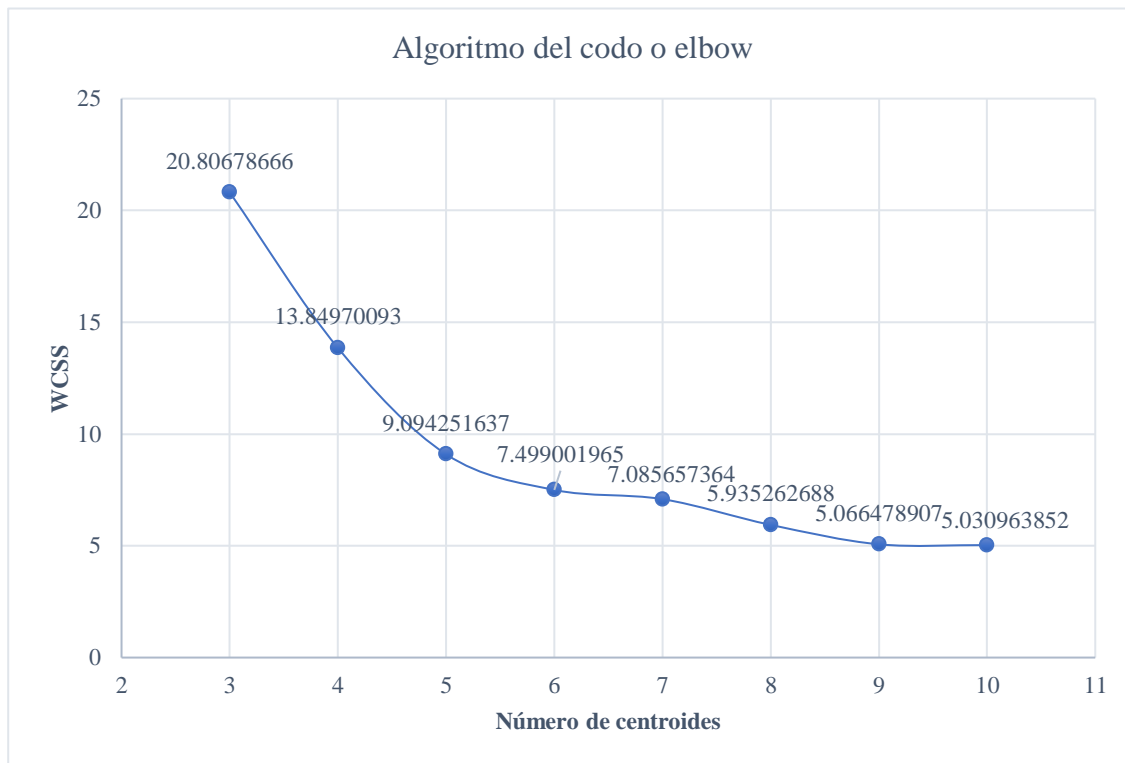


Figura 26 Algoritmo del codo o elbow

Tabla 6. Núm. de centroides vs WCSS

N. de centroides	WCSS
3	20.8067867
4	13.8497009
5	9.09425164
6	7.49900196
7	7.08565736
8	5.93526269
9	5.06647891
10	5.03096385

Se calculó el WCSS para 7 centroides como se explica a continuación, en la figura 27 se pueden observar los puntos que corresponden a cada centroide formado un total de siete grupos.

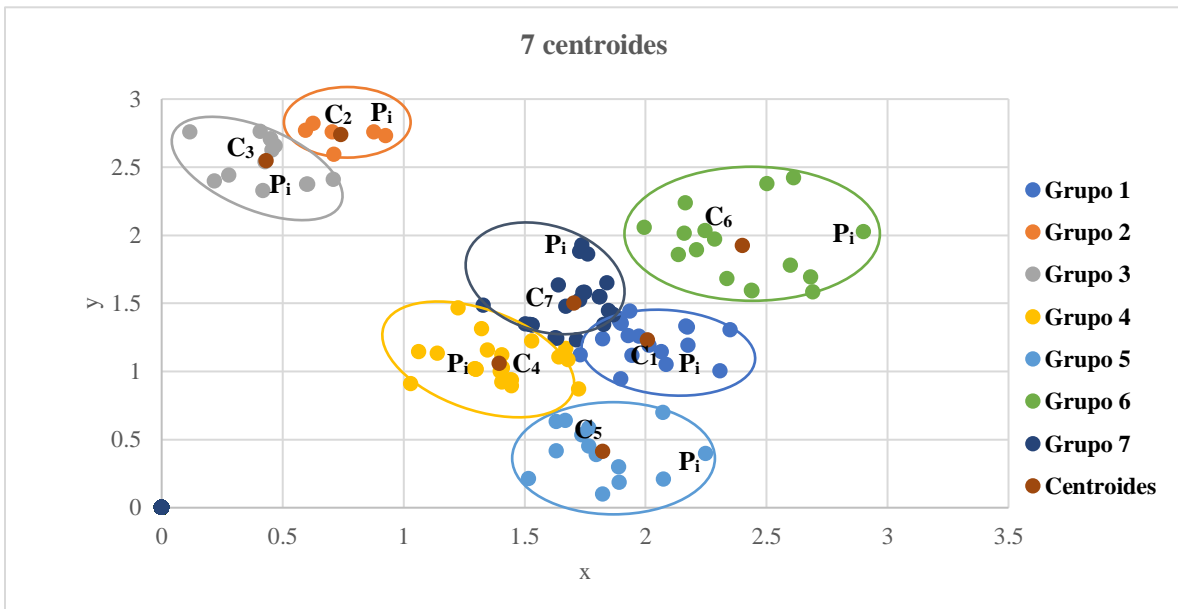


Figura 27 Agrupación en siete centroides

$$\begin{aligned}
 WCSS = & \sum_{P_i \text{ en Grupo 1}} distancia(P_i, C_1)^2 + \sum_{P_i \text{ en Grupo 2}} distancia(P_i, C_2)^2 + \sum_{P_i \text{ en Grupo 3}} distancia(P_i, C_3)^2 \\
 & + \sum_{P_i \text{ en Grupo 4}} distancia(P_i, C_4)^2 + \sum_{P_i \text{ en Grupo 5}} distancia(P_i, C_5)^2 \\
 & + \sum_{P_i \text{ en Grupo 6}} distancia(P_i, C_6)^2 + \sum_{P_i \text{ en Grupo 7}} distancia(P_i, C_7)^2
 \end{aligned}$$

$$\begin{aligned}
 WCSS = & 0.838234 + 0.118747 + 0.624663 + 1.188322 + 0.997365 + 2.0110445 \\
 & + 1.307882 = 7.085657
 \end{aligned}$$

El mismo procedimiento para encontrar la suma de cuadrados dentro del grupo (WCSS) se realizó para el número de centroides registrados en la tabla 6.

Iniciar Algoritmo k-medias con 7 centros

1. En la tabla 7, se muestran los datos de entrada, las primeras columnas llevan por nombre: Núm. de nodo, municipio, comunidad, tipo de nodo, latitud, longitud, abscisa (x) y ordenada (y).

Tabla 7 Datos de entrada

Núm. De Nodo	Municipio	Comunidad	Nodo	Latitud	Longitud	x	y
1	Celaya	Pedro María / Patria Nueva	Publico	20.52794	-100.878154	2.085471663	1.04988673
2	Celaya	Roque	Publico	20.581634	-100.838369	2.174654989	1.19335247
3	Celaya	San Elías	Publico	20.633491	-100.841508	2.167618506	1.3319099
4	Celaya	San Juan de la Vega	Publico	20.624287	-100.760489	2.349233282	1.30731761
5	Celaya	Parque Lineal	Publico	20.511533	-100.77846	2.308948915	1.00604864
6	Celaya	San isidro de Elguera	Publico	20.563705	-100.887103	2.065411299	1.14544773
7	Celaya	San Elías	Escolar	20.631236	-100.838563	2.174220112	1.32588474
8	Comonfort	Jalpilla	Escolar	20.73049	-100.71975	2.440555124	1.59108287
9	Comonfort	Jalpilla	Escolar	20.7328	-100.72132	2.437035762	1.59725499
10	Comonfort	San Isidro de San Jeronimo	Escolar	20.763667	-100.765909	2.337083637	1.67972895
11	Cortazar	Tierra Fría	Publico	20.461384	-101.039803	1.723114107	0.87205484
12	Cortazar	Centro Cultural	Publico	20.488899	-100.962013	1.897490653	0.94557254
13	Cortazar	La Gavia	Publico	20.397251	-100.883689	2.073064231	0.70069699
14	Jaral de Progreso	Santiago Capitiro	Publico	20.304513	-101.020675	1.765992042	0.45290905
15	Jaral de Progreso	Providencia	Publico	20.35469	-101.021035	1.765185055	0.58697766
16	Jaral de Progreso	Victoria de Cortázar	Publico	20.335613	-101.033919	1.736303869	0.53600557
17	Jaral de Progreso	DIF Municipal	Publico	20.372157	-101.081279	1.630140182	0.63364798
18	Jaral de Progreso	Jardín Municipal	Publico	20.37461	-101.063967	1.668947314	0.64020219
19	Juventino Rosas	Santiago de Cuenda	Publico	20.598988	-100.994854	1.82387322	1.23972086
20	Juventino Rosas	COMUDE SUR	Publico	20.637948	-100.992981	1.828071796	1.34381862

21	Juventino Rosas	Valencia	Publico	20.644062	-100.963131	1.894984509	1.3601547
22	Juventino Rosas	Dulces Nombres	Publico	20.650485	-100.964948	1.890911464	1.37731641
23	Juventino Rosas	San Diego de los Dolores	Publico	20.726128	-101.028434	1.74859922	1.57942798
24	Juventino Rosas	Franco Tavera	Publico	20.6063	-100.9288	1.971941974	1.2592579
25	Juventino Rosas	San Antonio de Romerillo	Publico	20.60756	-100.948	1.928902641	1.26262451
26	Juventino Rosas	San Antonio de Morales	Publico	20.58156	-100.9096	2.014981306	1.19315475
27	Juventino Rosas	El Naranjillo	Publico	20.714818	-101.000976	1.810149949	1.54920864
28	Juventino Rosas	Romero	Publico	20.752	-100.98700	1.841478997	1.64855574
29	Juventino Rosas	Mesas de Acosta	Publico	20.74748	- 101.07652	1.640808108	1.63647868
30	Juventino Rosas	La Peña	Publico	20.6755	-100.94534	1.934865382	1.44415433
31	León	San Francisco del Duran	Publico	21.006077	-101.621692	0.418733158	2.32742758
32	León	Vaquerías	Publico	21.15663	- 101.39523	0.926377603	2.72969221
33	León	Derramadero	Publico	21.16696	- 101.41692	0.877756607	2.75729308
34	León	San José de Otates Sur	Publico	21.16791	- 101.49336	0.706406264	2.7598314
35	León	Sauz Seco	Publico	21.17207	- 101.54277	0.595647231	2.77094656
36	León	Cuesta Blanca	Publico	21.10638	- 101.49065	0.712481086	2.59542855
37	León	San José de los Romeros	Publico	21.03744	- 101.49197	0.709522132	2.41122681
38	León	Los Alisos	Publico	21.19096	- 101.52930	0.625842013	2.82141902
39	León	Alfaro	Publico	21.14697	- 101.60730	0.450994724	2.70388153
40	León	San José del Potrero	Escolar	21.12927	-101.59938	0.468748449	2.65658865
41	León	León Urbano	Escolar	21.08529	-101.61801	0.426986846	2.53907788
42	León	Los Sauces	Escolar	21.02413	-101.53926	0.603515359	2.37566364
43	León	Los Naranjos	Escolar	21.16928	-101.6267	0.407507065	2.76349192
44	León	Alfaro	Escolar	21.14869	-101.6081	0.449201419	2.70847722
45	León	León Urbano	Escolar	21.11748	-101.60476	0.456688469	2.62508679
46	León	Fraccionamiento Paseo de las Torres	Escolar	21.16689	-101.75621	0.117193317	2.75710605
47	León	Los Sauces	Escolar	21.0258	-101.54116	0.599256259	2.37566364

48	León	San Pedro del Monte	Escolar	21.03328	-101.71161	0.2171701	2.40011165
49	León	Los Arcos	Escolar	21.04922	-101.68482	0.277223419	2.44270195
50	Pueblo Nuevo	Congragación de Panales	Escolar	20.56314	-101.33472	1.06201875	1.1439381
51	Salamanca	Congregación de Cárdenas	Publico	20.627029	-101.218901	1.321642316	1.314644
52	Salamanca	Cerro Gordo	Publico	20.593413	-101.126121	1.529620924	1.22482494
53	Salamanca	Loma de Flores	Publico	20.683629	-101.262125	1.224750018	1.46587432
54	Salamanca	Santo Domingo	Publico	20.515328	-101.227903	1.301463145	1.01618855
55	Salamanca	Valencia	Publico	20.602204	-101.082313	1.627822335	1.24831374
56	Salamanca	Ecoparque	Publico	20.567796	-101.207279	1.347694562	1.15637853
57	Salamanca	Los Miranda	Escolar	20.6401	-101.13816	1.502633917	1.34956858
58	Salamanca	La Luz	Escolar	20.55521	-101.18072	1.407230063	1.12274982
59	Salamanca	Uruétaro	Escolar	20.51961	-101.18027	1.408238798	1.02762969
60	Salamanca	Valencia de Cerro Gordo	Escolar	20.60014	-101.08022	1.63251407	1.24279891
61	Salamanca	El Xoconostle	Escolar	20.63728	-101.12524	1.531595802	1.34203378
62	Salamanca	Uruétaro	Escolar	20.5161	-101.18116	1.406243745	1.01825127
63	Salamanca	La Capilla	Escolar	20.55888	-101.30033	1.139108471	1.13255574
64	Salamanca	Colonia el 4 de Altamira	Escolar	20.4848	-101.16745	1.436976519	0.93462037
65	Salamanca	Colonia el 4 de Altamira	Escolar	20.48533	-101.16561	1.441101122	0.93603648
66	Salamanca	Santo Domingo	Escolar	20.51569	-101.22903	1.298936826	1.01715579
67	Salamanca	Santo Domingo	Escolar	20.51706	-101.23139	1.293646575	1.02081631
68	Salamanca	El Xoconostle	Escolar	20.63725	-101.12582	1.530295655	1.34195363
69	Salamanca	Cuatro de Altamira	Escolar	20.47996	-101.18148	1.405526423	0.92168831
70	Salamanca	Colonia el Cuatro de Altamira	Escolar	20.48589	-101.16368	1.445427471	0.937532755
71	Salamanca	Uruetaro	Escolar	20.51808	-101.18081	1.407028317	1.02354166
72	Salamanca	El Recuerdo de Anc ³ n (Xoconoxtle de Arriba)	Escolar	20.63895	-101.12956	1.521911952	1.34649588
73	Salamanca	San Jose de Mendoza	Escolar	20.690968	-101.215386	1.329521652	1.48548349

74	Salamanca	El Circuito	Escolar	20.50971	-101.1839	1.400101674	1.00117774
75	Salvatierra	El Sabino	Publico	20.281045	-101.006595	1.797554219	0.39020457
76	Salvatierra	Maravatío del Encinal	Publico	20.205091	-100.965222	1.890297256	0.18726203
77	Salvatierra	San Nicolás de los Agustinos	Publico	20.246335	-100.966111	1.888304446	0.29746245
78	Salvatierra	Centro Histórico	Publico	20.212867	-100.882702	2.075276721	0.20803883
79	San Miguel de Allende	Don Juan	Escolar	20.89607	-100.80634	2.246452217	2.03349836
80	San Miguel de Allende	Corral de Piedras de Abajo	Escolar	20.84282	-100.82233	2.210608523	1.89121895
81	San Miguel de Allende	Pantoja	Escolar	20.87324	-100.78897	2.285389363	1.97249856
82	San Miguel de Allende	Tlaxcalilla	Escolar	20.88913	-100.84462	2.160642548	2.01495527
83	San Miguel de Allende	Los Rodríguez	Publico	21.04082	-100.642947	2.61271918	2.42025788
84	San Miguel de Allende	Cruz del Palmar	Publico	20.971912	-100.842675	2.165002522	2.23614164
85	San Miguel de Allende	Don Francisco	Publico	20.905315	-100.918473	1.995091306	2.0582002
86	San Miguel de Allende	Alonso Yáñez	Publico	20.829893	-100.855602	2.136024946	1.85667912
87	San Miguel de Allende	Moral de Puerto de Nieto	Publico	20.89309	-100.51479	2.9	2.02553605
88	San Miguel de Allende	Jalpa	Publico	20.76891	-100.61162	2.682942782	1.6937378
89	San Miguel de Allende	Charco de Sierra	Publico	20.72778	-100.6075	2.692178306	1.58384198
90	San Miguel de Allende	San José de Allendes	Publico	20.85651	-101.03344	1.737377611	1.92779745
91	San Miguel de Allende	San Isidro de Ensaye (La Cuevita)	Publico	20.83845	-101.03772	1.727783426	1.87954268
92	San Miguel de Allende	El Nuevo Cimatario	Publico	20.83181	-101.02277	1.761295823	1.86180118
93	San Miguel de Allende	La Biznaga	Publico	21.02538	-100.69224	2.502222418	2.37900353
94	San Miguel de Allende	Lagunillas	Publico	20.80138	-100.64883	2.599531659	1.78049484
95	Santa Cruz de Juventino Rosas	El Naranjillo	Escolar	20.715255	-101.000962	1.810181332	1.55037626
96	Santa Cruz de Juventino Rosas	El Naranjillo	Escolar	20.713992	-101.000079	1.812160693	1.54700164
97	Santa Cruz de Juventino Rosas	UPJR	Escolar	20.654297	-100.964101	1.892810126	1.38750174

98	Santa Cruz de Juventino Rosas	La Trinidad	Escolar	20.57278	-101.06333	1.670375233	1.16969534
99	Santa Cruz de Juventino Rosas	San Julian Tierra Blanca	Escolar	20.59556	-101.04333	1.715207871	1.23056154
100	Santa Cruz de Juventino Rosas	Jaralillo	Escolar	20.66556	-100.97611	1.865890368	1.41759551
101	Santa Cruz de Juventino Rosas	San Jose del Sauz	Escolar	20.67611	-100.98429	1.847553819	1.4457842
102	Santa Cruz de Juventino Rosas	Jaralillo	Escolar	20.66542	-100.97585	1.866473192	1.41722144
103	Santa Cruz de Juventino Rosas	La Tinaja	Escolar	20.70506	-101.03804	1.727066104	1.5231361
104	Santa Cruz de Juventino Rosas	El Carmen de Arriba	Escolar	20.68799	-101.06271	1.671765045	1.47752653
105	Santa Cruz de Juventino Rosas	San Diego de los Dolores	Escolar	20.726702	-101.031667	1.741352024	1.58096166
106	Santa Cruz de Juventino Rosas	San Diego de los Dolores	Escolar	20.726265	-101.028494	1.748464722	1.57979403
107	Santa Cruz de Juventino Rosas	Los Dulces Nombres	Escolar	20.641806	-100.961227	1.899252576	1.35412687
108	Santiago Maravatio	Jardín Municipal	Publico	20.172432	-100.995011	1.823521283	0.1
109	Tarimoro	La Moncada	Publico	20.283137	-100.805517	2.24829708	0.39579422
110	Valle de Santiago	Alto de Altamira	Escolar	20.47034	-101.16312	1.446682785	0.8959845
111	Valle de Santiago	Santa Barbara	Publico	20.475553	-101.349059	1.02987599	0.90991318
112	Villagran	Suchitlan	Publico	20.554805	-101.036322	1.730917228	1.12166769
113	Villagran	San Isidro Cerro Gordo	Publico	20.541822	-101.059277	1.679460567	1.08697824
114	Villagran	Los Ángeles	Publico	20.553862	-100.941347	1.943816218	1.11914808
115	Villagran	Parque Agro-Industrial XONOTLI	Escolar	20.549669	-101.075751	1.642531923	1.107944745
116	Yuriria	Loma de Zempoala	Publico	20.291165	-101.080675	1.631494128	0.41724434
117	Yuriria	Centro Histórico	Publico	20.214227	-101.131645	1.517238149	0.21167264

2. En la tabla 8, se observan las coordenadas de los siete centros en la fila dos, en la columna centro 1, centro 2 al centro 7 se registra la distancia del nodo1 hasta el nodo 117 al centroide indicado.

Por ejemplo, la distancia del nodo 1 al primer centroide se calculó como sigue:

Coordenada del nodo 1: (2.085472, 1.049887)

Centro 1: (2.174655, 1.193352)

$$\text{Distancia euclidiana: } d_E(P_1, P_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} = \sqrt{(2.085472 - 2.174655)^2 + (1.049887 - 1.193352)^2}$$

$$d_E(P_1, P_2) = 0.168926$$

Las columnas binarias son el resultado de una comparación de distancias con respecto al centroide indicado, en cada celda queda registrado con un 1 si es la distancia más corta al centroide comparado.

Por ejemplo, en la fila tres de la tabla 8 se realizó la comparación de las primeras siete distancias con respecto al centro uno:

$$\text{Condición} = (Si(\text{Centro1} < \text{Centro2}, \text{Centro1} < \text{Centro3}, \text{Centro1} < \text{Centro4}, \text{Centro1} < \text{Centro5}, \text{Centro1} < \text{Centro6}, \text{Centro1} < \text{Centro7}), 1, 0)$$

$$\text{Condición} = SI(Y(0.168926 < 2.193352, 0.168926 < 2.228960, 0.168926 < 1.218901, 0.168926 < 0.719776, 0.168926 < 0.944023, 0.168926 < 0.461887), 1, 0) = 1$$

Tabla 8 Inicializar algoritmo k-medias con siete centros

Centro1		Centro2		Centro3		Centro4		Centro5		Centro6		Centro7	
2.174655	1.193352	0.706406	2.759831	0.426987	2.539078	1.321642	0.100000	1.797554	0.390205	2.285389	1.972499	1.847554	1.445784
0.168926	1	2.196755	0	2.228960	0	1.218901	0	0.719776	0	0.944023	0	0.461887	0
0.000000	1	2.147000	0	2.205747	0	1.386741	0	0.887272	0	0.786976	0	0.413179	0
0.138736	1	2.043062	0	2.118267	0	1.494416	0	1.011809	0	0.651325	0	0.339719	0
0.208484	1	2.192870	0	2.283038	0	1.585421	0	1.070255	0	0.668238	0	0.520438	0
0.230473	1	2.375689	0	2.427336	0	1.340037	0	0.800493	0	0.966737	0	0.637380	0

0.119286	1	2.110244	0	2.150963	0	1.283025	0	0.801336	0	0.855806	0	0.371031	0
0.132533	1	2.051994	0	2.127125	0	1.493212	0	1.008650	0	0.656101	0	0.347975	0
0.478427	0	2.091231	0	2.225568	0	1.864214	0	1.362189	0	0.411770	1	0.610543	0
0.481644	0	2.084865	0	2.219758	0	1.867050	0	1.365982	0	0.404728	1	0.608632	0
0.512782	0	1.955947	0	2.094505	0	1.877942	0	1.397843	0	0.297298	1	0.542558	0
0.554185	0	2.144154	0	2.111614	0	0.870200	0	0.487566	1	1.235771	0	0.587070	0
0.371773	1	2.170303	0	2.168327	0	1.023032	0	0.564288	0	1.097744	0	0.502698	0
0.503021	0	2.471394	0	2.467633	0	0.962014	0	0.415104	1	1.289403	0	0.778466	0
0.845732	0	2.538624	0	2.478918	0	0.567443	0	0.070200	1	1.605903	0	0.996220	0
0.731680	0	2.417086	0	2.366742	0	0.658694	0	0.199418	1	1.479960	0	0.862748	0
0.790099	0	2.450733	0	2.393034	0	0.601702	0	0.158144	1	1.537858	0	0.916555	0
0.780875	0	2.318176	0	2.253495	0	0.616402	0	0.295452	1	1.490595	0	0.840734	0
0.749477	0	2.327942	0	2.268963	0	0.642214	0	0.281138	1	1.467997	0	0.825144	0
0.353833	0	1.886655	0	1.907779	0	1.245472	0	0.849924	0	0.866002	0	0.207420	1
0.377836	0	1.806440	0	1.841652	0	1.342965	0	0.954102	0	0.777417	0	0.103810	1
0.325636	0	1.836250	0	1.882784	0	1.384453	0	0.974831	0	0.726210	0	0.097888	1
0.338161	0	1.820549	0	1.868894	0	1.398429	0	0.991517	0	0.714041	0	0.081041	1
0.574959	0	1.574649	0	1.633275	0	1.539805	0	1.190231	0	0.665318	0	0.166291	1
0.213157	1	1.962983	0	2.006197	0	1.329198	0	0.886377	0	0.779077	0	0.224197	0
0.255329	0	1.932906	0	1.971062	0	1.311663	0	0.882252	0	0.794358	0	0.200412	1
0.159674	1	2.041285	0	2.081642	0	1.294491	0	0.831867	0	0.824923	0	0.303074	0
0.509409	0	1.638248	0	1.700877	0	1.529328	0	1.159073	0	0.636417	0	0.109980	1
0.564107	0	1.588497	0	1.671472	0	1.633479	0	1.259118	0	0.549541	0	0.202863	1
0.693796	0	1.461174	0	1.512629	0	1.569278	0	1.256093	0	0.726907	0	0.281262	1
0.346988	0	1.800033	0	1.863479	0	1.477428	0	1.062857	0	0.634046	0	0.087327	1
2.090308	0	0.519354	0	0.211811	1	2.403472	0	2.377810	0	1.900100	0	1.678935	0
1.979529	0	0.222026	1	0.534532	0	2.659232	0	2.496428	0	1.555717	0	1.580185	0
2.031712	0	0.171369	1	0.500811	0	2.694112	0	2.539515	0	1.611624	0	1.631123	0

2.147000	0	0.000000	1	0.356100	0	2.730058	0	2.608780	0	1.764392	0	1.740384	0
2.232055	0	0.111315	1	0.286722	0	2.767856	0	2.666929	0	1.868889	0	1.822999	0
2.025776	0	0.164515	1	0.291002	0	2.568704	0	2.457722	0	1.691769	0	1.615572	0
1.905212	0	0.348619	0	0.310116	1	2.390912	0	2.295287	0	1.635799	0	1.492379	0
2.247092	0	0.101408	1	0.345340	0	2.808961	0	2.698836	0	1.864072	0	1.839824	0
2.291878	0	0.261468	0	0.166543	1	2.745583	0	2.676999	0	1.974823	0	1.879677	0
2.247482	0	0.259114	0	0.124711	1	2.695102	0	2.627208	0	1.941176	0	1.834980	0
2.205747	0	0.356100	0	0.000000	1	2.597982	0	2.548747	0	1.942852	0	1.792568	0
1.966301	0	0.397708	0	0.240555	1	2.386284	0	2.316846	0	1.729521	0	1.553160	0
2.363927	0	0.298922	0	0.225258	1	2.815996	0	2.750404	0	2.037673	0	1.951945	0
2.296256	0	0.262281	0	0.170850	1	2.750510	0	2.681873	0	1.978194	0	1.884087	0
2.236352	0	0.283752	0	0.090993	1	2.669121	0	2.606265	0	1.941654	0	1.823530	0
2.584274	0	0.589219	0	0.378825	1	2.917346	0	2.902729	0	2.305793	0	2.171109	0
1.969706	0	0.398831	0	0.237447	1	2.387569	0	2.319044	0	1.733663	0	1.556574	0
2.299568	0	0.607248	0	0.251664	1	2.551543	0	2.556822	0	2.111962	0	1.889151	0
2.271810	0	0.533638	0	0.178094	1	2.564968	0	2.554242	0	2.062480	0	1.860049	0
1.113733	0	1.654561	0	1.532867	0	1.075737	0	1.053151	0	1.477548	0	0.841532	1
0.861593	0	1.570695	0	1.516459	0	1.214644	0	1.039750	0	1.166868	0	0.542015	1
0.645801	0	1.741817	0	1.715536	0	1.143891	0	0.876573	0	1.063109	0	0.387175	1
0.988224	0	1.393917	0	1.337233	0	1.369307	0	1.218676	0	1.175425	0	0.623128	1
0.890983	0	1.842385	0	1.756104	0	0.916411	0	0.798725	0	1.372093	0	0.694815	1
0.549588	0	1.770224	0	1.762974	0	1.188432	0	0.874734	0	0.978181	0	0.295426	1
0.827787	0	1.726937	0	1.661192	0	1.056700	0	0.888480	0	1.243110	0	0.577594	1
0.689939	0	1.619512	0	1.603730	0	1.262608	0	1.003672	0	1.000374	0	0.358088	1
0.770666	0	1.780784	0	1.722458	0	1.026325	0	0.830045	0	1.221981	0	0.546110	1
0.784129	0	1.868981	0	1.802035	0	0.931663	0	0.746912	0	1.289252	0	0.606507	1
0.544391	0	1.777375	0	1.770208	0	1.184327	0	0.868421	0	0.979136	0	0.295711	1
0.660024	0	1.640454	0	1.628826	0	1.259654	0	0.988288	0	0.982696	0	0.332556	1

0.788109	0	1.876932	0	1.808828	0	0.922140	0	0.739977	0	1.297492	0	0.614442	1
1.037330	0	1.683822	0	1.576522	0	1.048566	0	0.992288	0	1.421078	0	0.774601	1
0.781736	0	1.965993	0	1.895881	0	0.842552	0	0.652997	1	1.340521	0	0.655639	0
0.777376	0	1.966216	0	1.896884	0	0.844528	0	0.651914	1	1.336816	0	0.651956	0
0.893268	0	1.840655	0	1.754008	0	0.917437	0	0.801054	0	1.373233	0	0.696206	1
0.897744	0	1.835490	0	1.748204	0	0.921242	0	0.807214	0	1.374501	0	0.698148	1
0.661272	0	1.639869	0	1.628004	0	1.259359	0	0.988561	0	0.983745	0	0.333817	1
0.815696	0	1.966606	0	1.890367	0	0.825959	0	0.660425	1	1.370533	0	0.685613	0
0.772798	0	1.966450	0	1.897938	0	0.846631	0	0.650816	0	1.332925	0	0.648093	1
0.786185	0	1.872318	0	1.804808	0	0.927480	0	0.744061	0	1.293073	0	0.610206	1
0.670467	0	1.631737	0	1.618985	0	1.262482	0	0.995224	0	0.987308	0	0.340442	1
0.894198	0	1.418533	0	1.387310	0	1.385506	0	1.191088	0	1.072785	0	0.519551	1
0.798038	0	1.890523	0	1.819915	0	0.904587	0	0.728874	0	1.314229	0	0.630784	1
0.887272	0	2.608780	0	2.548747	0	0.557414	0	0.000000	1	1.655789	0	1.056763	0
1.045503	0	2.831909	0	2.769894	0	0.575311	0	0.223130	1	1.828433	0	1.259248	0
0.940540	0	2.731326	0	2.675872	0	0.600081	0	0.129756	1	1.721459	0	1.149045	0
0.990313	0	2.895764	0	2.854926	0	0.761339	0	0.332136	1	1.776926	0	1.258519	0
0.843208	0	1.702733	0	1.888403	0	2.143289	0	1.703503	0	0.072368	1	0.710301	0
0.698792	0	1.736984	0	1.897637	0	1.999682	0	1.556810	0	0.110447	1	0.574648	0
0.786976	0	1.764392	0	1.942852	0	2.105958	0	1.655789	0	0.000000	1	0.684929	0
0.821722	0	1.633904	0	1.811151	0	2.090688	0	1.664827	0	0.131774	1	0.649600	0
1.302765	0	1.936321	0	2.188960	0	2.655273	0	2.187604	0	0.554647	1	1.238982	0
1.042834	0	1.549759	0	1.764219	0	2.296597	0	1.882154	0	0.289829	1	0.851727	0
0.883292	0	1.467309	0	1.640181	0	2.070768	0	1.679652	0	0.302684	1	0.629937	0
0.664451	0	1.691004	0	1.840239	0	1.936270	0	1.505028	0	0.189008	1	0.502046	0
1.103927	0	2.313232	0	2.525771	0	2.489759	0	1.972231	0	0.616895	1	1.201564	0
0.713261	0	2.245719	0	2.409136	0	2.095982	0	1.575789	0	0.485547	1	0.871410	0
0.648315	0	2.307865	0	2.458367	0	2.019940	0	1.491684	0	0.562611	1	0.855833	0

0.854764	0	1.324833	0	1.445956	0	1.874481	0	1.538770	0	0.549832	0	0.494445	1
0.818872	0	1.348377	0	1.458444	0	1.825301	0	1.490971	0	0.565301	0	0.449990	1
0.785932	0	1.385370	0	1.496357	0	1.815830	0	1.472043	0	0.535657	0	0.424865	1
1.230069	0	1.835752	0	2.081400	0	2.566637	0	2.109948	0	0.460720	1	1.139951	0
0.724746	0	2.131437	0	2.301173	0	2.111176	0	1.605016	0	0.368172	1	0.823105	0
0.510203	0	1.637407	0	1.700223	0	1.530445	0	1.160240	0	0.635618	0	0.111068	1
0.506429	0	1.641234	0	1.703796	0	1.527882	0	1.156889	0	0.636391	0	0.107227	1
0.342243	0	1.814068	0	1.864072	0	1.408508	0	1.001836	0	0.704514	0	0.073790	1
0.504834	0	1.859508	0	1.849655	0	1.125106	0	0.789798	0	1.011304	0	0.328051	1
0.460951	0	1.832034	0	1.836227	0	1.197106	0	0.844382	0	0.935723	0	0.252658	1
0.381603	0	1.773697	0	1.824326	0	1.425575	0	1.029661	0	0.695627	0	0.033628	1
0.413179	0	1.740384	0	1.792568	0	1.444894	0	1.056763	0	0.684929	0	0.000000	1
0.380911	0	1.774361	0	1.825016	0	1.425452	0	1.029327	0	0.695574	0	0.034260	1
0.555961	0	1.603484	0	1.649953	0	1.479758	0	1.135122	0	0.716695	0	0.143180	1
0.577627	0	1.605062	0	1.635960	0	1.421325	0	1.094574	0	0.788373	0	0.178632	1
0.581371	0	1.568709	0	1.626512	0	1.539287	0	1.192083	0	0.670282	0	0.171906	1
0.575304	0	1.574285	0	1.632951	0	1.540119	0	1.190602	0	0.665210	0	0.166665	1
0.318896	0	1.843607	0	1.889888	0	1.380749	0	0.969272	0	0.729030	0	0.105232	1
1.148353	0	2.884900	0	2.810589	0	0.501879	0	0.291364	1	1.928619	0	1.345999	0
0.800951	0	2.822428	0	2.812621	0	0.972720	0	0.450778	1	1.577141	0	1.123866	0
0.786366	0	2.005476	0	1.933788	0	0.805746	0	0.615568	1	1.364665	0	0.680424	0
1.179346	0	1.877986	0	1.737139	0	0.860864	1	0.927053	0	1.644810	0	0.977627	0
0.449491	0	1.932150	0	1.925951	0	1.100596	0	0.734492	0	1.015555	0	0.344464	1
0.506491	0	1.935271	0	1.917625	0	1.049838	0	0.706710	0	1.072985	0	0.396229	1
0.242472	1	2.055000	0	2.077732	0	1.194053	0	0.743472	0	0.919173	0	0.340525	0
0.538934	0	1.898700	0	1.877683	0	1.057791	0	0.734291	0	1.077367	0	0.395183	1
0.947295	0	2.518631	0	2.439880	0	0.443455	0	0.168247	1	1.687126	0	1.050988	0
1.181479	0	2.674053	0	2.570110	0	0.225230	1	0.332341	0	1.921084	0	1.277552	0

En resumen, cada patrón se asigna al elemento de cuyo centroide diste menos, con base en los siguientes pasos:

- Tomamos el patrón x del conjunto de entrada y calculamos la distancia a cada uno de los centroides.
- Verificamos cual es la menor de estas distancias.
- El patrón de entrada se asigna al sector representado por el centroide cuya distancia haya sido mínima.
- Repetimos este procedimiento para todos los patrones del conjunto de aprendizaje.

3. En la tabla 9, calcularon las coordenadas (x, y) de nuevos centroides con base al promedio de los elementos pertenecientes al centroide, por lo que se modifican también las distancias.

Los centroides de los nuevos sectores generados se calculan en la siguiente ecuación:

$$c_j = \frac{1}{p_j} \sum_{i=1}^{p_j} x_i \quad p_j: \text{Número de patrones que han correspondido a la neurona } j - \text{ésima.}$$

En Microsoft Excel

$$\text{Abscisa} = \text{PROMEDIO.SI.CONJUNTO}(\$G\$3:\$G\$119, J3:J119, 1) = 2.10489$$

$$\text{Ordenanda} = \text{PROMEDIO.SI.CONJUNTO}(\$H\$3:\$H\$119, J3:J119, 1) = 1.170635$$

Tabla 9 Segunda iteración con nuevos centroides

Iteración2_C1		Iteración2_C2		Iteración2_C3		Iteración2_C4		Iteración2_C5		Iteración2_C6		Iteración2_C7	
2.104890	1.170635	0.740752	2.739102	0.430980	2.545116	1.273557	0.560793	1.747099	0.561491	2.400218	1.925883	1.618824	1.335066
0.122299	1	2.159101	0	2.230034	0	0.947849	0	0.594161	0	0.930825	0	0.546889	0
0.073371	1	2.108416	0	2.206279	0	1.100958	0	0.762924	0	0.766472	0	0.573612	0
0.173045	1	2.004030	0	2.118438	0	1.180664	0	0.877715	0	0.637892	0	0.548804	0
0.279975	1	2.153420	0	2.282945	0	1.309343	0	0.958553	0	0.620663	0	0.730936	0
0.262161	1	2.337245	0	2.428064	0	1.127071	0	0.716455	0	0.924351	0	0.764543	0

0.046829	1	2.072307	0	2.151845	0	0.984304	0	0.665078	0	0.849220	0	0.485176	0
0.170027	1	2.012961	0	2.127301	0	1.181761	0	0.875632	0	0.641150	0	0.555472	0
0.538004	0	2.051165	0	2.224538	0	1.556722	0	1.241347	0	0.337222	1	0.860690	0
0.540672	0	2.044797	0	2.218716	0	1.558184	0	1.244516	0	0.330684	1	0.859194	0
0.559545	0	1.915867	0	2.093353	0	1.543731	0	1.264333	0	0.254122	1	0.796674	0
0.484668	0	2.109716	0	2.113940	0	0.546796	0	0.311489	1	1.252607	0	0.474612	0
0.306051	1	2.134196	0	2.170067	0	0.733041	0	0.412476	0	1.101700	0	0.478916	0
0.471014	0	2.435190	0	2.469478	0	0.811656	0	0.354446	1	1.268113	0	0.780230	0
0.793714	0	2.505553	0	2.481851	0	0.504114	0	0.110213	1	1.603713	0	0.894349	0
0.675318	0	2.383506	0	2.369474	0	0.492325	0	0.031252	1	1.481868	0	0.762272	0
0.733900	0	2.417593	0	2.395912	0	0.463410	0	0.027677	1	1.540306	0	0.807651	0
0.716758	0	2.285596	0	2.256478	0	0.363950	0	0.137426	1	1.504291	0	0.701510	0
0.686589	0	2.294979	0	2.271840	0	0.403286	0	0.110920	1	1.479098	0	0.696670	0
0.289384	0	1.849674	0	1.908981	0	0.873951	0	0.682562	0	0.896098	0	0.226133	1
0.326529	0	1.768921	0	1.842547	0	0.959487	0	0.786507	0	0.816180	0	0.209431	1
0.282804	0	1.798263	0	1.883465	0	1.012498	0	0.812240	0	0.758491	0	0.277298	1
0.297496	0	1.782506	0	1.869533	0	1.023639	0	0.828404	0	0.748544	0	0.275348	1
0.542268	0	1.536424	0	1.633607	0	1.123958	0	1.017939	0	0.737996	0	0.276684	1
0.159779	1	1.925037	0	2.006986	0	0.987722	0	0.733099	0	0.792344	0	0.361164	0
0.198579	1	1.895175	0	1.971942	0	0.960232	0	0.724321	0	0.813664	0	0.318429	0
0.092686	1	2.003400	0	2.082513	0	0.974470	0	0.686120	0	0.827827	0	0.420808	0
0.479781	0	1.599831	0	1.701159	0	1.124677	0	0.989728	0	0.700045	0	0.287163	1
0.545705	0	1.549481	0	1.671325	0	1.227095	0	1.091155	0	0.623778	0	0.384514	1
0.657558	0	1.423334	0	1.513045	0	1.136650	0	1.080230	0	0.812685	0	0.302213	1
0.322058	1	1.761476	0	1.863810	0	1.103475	0	0.902414	0	0.669788	0	0.334339	0
2.044821	0	0.522658	0	0.218032	1	1.962580	0	2.209771	0	2.021761	0	1.557241	0
1.954367	0	0.185864	1	0.528665	0	2.196510	0	2.318336	0	1.678783	0	1.557069	0
2.005827	0	0.138207	1	0.494599	0	2.231876	0	2.361632	0	1.734684	0	1.603717	0

2.116909	0	0.040116	1	0.349231	0	2.270998	0	2.432230	0	1.887979	0	1.691881	0
2.199730	0	0.148558	1	0.279490	0	2.311783	0	2.491493	0	1.992638	0	1.763134	0
1.992195	0	0.146428	1	0.285962	0	2.110580	0	2.281959	0	1.815694	0	1.552408	0
1.867115	0	0.329359	0	0.309050	1	1.934487	0	2.120870	0	1.758980	0	1.408883	0
2.216455	0	0.141352	1	0.338105	0	2.351588	0	2.522795	0	1.987560	0	1.787528	0
2.255264	0	0.291890	0	0.160022	1	2.295526	0	2.503942	0	2.098750	0	1.799300	0
2.210208	0	0.284243	0	0.117697	1	2.245012	0	2.454306	0	2.065067	0	1.751883	0
2.165178	0	0.372099	0	0.007239	1	2.151811	0	2.377719	0	2.066312	0	1.694143	0
1.925155	0	0.388486	0	0.241831	1	1.934609	0	2.144530	0	1.852145	0	1.453855	0
2.327725	0	0.334136	0	0.219634	1	2.366839	0	2.577463	0	2.161593	0	1.872882	0
2.259704	0	0.293154	0	0.164374	1	2.300459	0	2.508802	0	2.102122	0	1.803961	0
2.198181	0	0.306091	0	0.084002	1	2.220041	0	2.433842	0	2.065476	0	1.736292	0
2.543193	0	0.623818	0	0.378685	1	2.482130	0	2.734469	0	2.429636	0	2.068113	0
1.928478	0	0.390011	0	0.238811	1	1.936088	0	2.146804	0	1.856277	0	1.456832	0
2.252798	0	0.623740	0	0.258343	1	2.121096	0	2.391905	0	2.233963	0	1.760385	0
2.226773	0	0.550192	0	0.184742	1	2.129381	0	2.387360	0	2.184996	0	1.739755	0
1.043213	0	1.627194	0	1.536720	0	0.620328	0	0.899211	0	1.549908	0	0.588695	1
0.796376	0	1.538348	0	1.518993	0	0.755383	0	0.865016	0	1.239733	0	0.297882	1
0.577816	0	1.707439	0	1.717609	0	0.711693	0	0.698075	0	1.117775	0	0.141811	1
0.928339	0	1.362117	0	1.339714	0	0.906396	0	1.044394	0	1.262273	0	0.415217	1
0.818137	0	1.811857	0	1.759363	0	0.456250	0	0.636664	0	1.426466	0	0.449890	1
0.483350	0	1.734746	0	1.764689	0	0.773427	0	0.697103	0	1.027470	0	0.087218	1
0.757330	0	1.695108	0	1.664018	0	0.600182	0	0.716530	0	1.303818	0	0.324716	1
0.628275	0	1.584698	0	1.605545	0	0.821367	0	0.825124	0	1.066675	0	0.117091	1
0.699301	0	1.748367	0	1.725163	0	0.577637	0	0.656142	0	1.277125	0	0.299750	1
0.711177	0	1.837029	0	1.804937	0	0.485876	0	0.576291	0	1.338238	0	0.372644	1
0.477856	0	1.741885	0	1.771924	0	0.770702	0	0.690877	0	1.027605	0	0.093278	1
0.598368	0	1.605376	0	1.630571	0	0.822752	0	0.809746	0	1.046606	0	0.087506	1

0.715071	0	1.845049	0	1.811755	0	0.476313	0	0.569923	0	1.346024	0	0.381526	1
0.966532	0	1.655197	0	1.580118	0	0.587358	0	0.834127	0	1.489887	0	0.520708	1
0.708386	0	1.934136	0	1.898875	0	0.407986	1	0.485182	0	1.382185	0	0.439802	0
0.704025	0	1.934304	0	1.899863	0	0.410949	1	0.483652	0	1.378296	0	0.436818	0
0.820437	0	1.810157	0	1.757274	0	0.457068	0	0.639124	0	1.427797	0	0.450993	1
0.824961	0	1.805048	0	1.751480	0	0.460462	0	0.645445	0	1.429561	0	0.452209	1
0.599590	0	1.604806	0	1.629753	0	0.822269	0	0.810016	0	1.047730	0	0.088796	1
0.742350	0	1.935179	0	1.893478	0	0.384267	1	0.496401	0	1.413442	0	0.465164	0
0.699448	0	1.934482	0	1.900902	0	0.414092	1	0.482093	0	1.374213	0	0.433704	0
0.713195	0	1.840400	0	1.807722	0	0.481613	0	0.573707	0	1.341881	0	0.376703	1
0.608926	0	1.596735	0	1.620747	0	0.824020	0	0.816665	0	1.052193	0	0.097583	1
0.836855	0	1.384994	0	1.389316	0	0.926383	0	1.013969	0	1.157731	0	0.326069	1
0.724874	0	1.858796	0	1.822894	0	0.458206	0	0.560118	0	1.362098	0	0.399150	1
0.838765	0	2.575684	0	2.551699	0	0.551066	0	0.178563	1	1.649701	0	0.961618	0
1.006515	0	2.798811	0	2.772919	0	0.721037	0	0.400690	1	1.811856	0	1.179471	0
0.899633	0	2.697866	0	2.678757	0	0.668773	0	0.299416	1	1.706988	0	1.072027	0
0.963051	0	2.861335	0	2.857558	0	0.875894	0	0.482316	1	1.748307	0	1.215952	0
0.874399	0	1.662832	0	1.886184	0	1.765046	0	1.554400	0	0.187683	1	0.939002	0
0.728298	0	1.696875	0	1.895958	0	1.627298	0	1.408197	0	0.192752	1	0.812105	0
0.821928	0	1.724409	0	1.940805	0	1.736870	0	1.510199	0	0.123929	1	0.922296	0
0.846159	0	1.593888	0	1.809089	0	1.703382	0	1.511151	0	0.255597	1	0.869377	0
1.348869	0	1.898927	0	2.185309	0	2.291498	0	2.050443	0	0.538111	1	1.471553	0
1.067201	0	1.510450	0	1.761334	0	1.897754	0	1.726007	0	0.389341	1	1.053683	0
0.894331	0	1.427233	0	1.638149	0	1.662179	0	1.517116	0	0.426187	1	0.815169	0
0.686751	0	1.650896	0	1.838783	0	1.556654	0	1.352323	0	0.273106	1	0.734559	0
1.167500	0	2.274100	0	2.523098	0	2.188787	0	1.863494	0	0.509621	1	1.455390	0
0.779604	0	2.205650	0	2.407526	0	1.808295	0	1.468941	0	0.365821	1	1.122940	0
0.718086	0	2.267750	0	2.457044	0	1.749033	0	1.392256	0	0.449704	1	1.101807	0

0.841642	0	1.285098	0	1.444907	0	1.443548	0	1.366341	0	0.662843	0	0.604471	1
0.802969	0	1.308844	0	1.457630	0	1.394784	0	1.318194	0	0.674029	0	0.555272	1
0.771860	0	1.345796	0	1.495546	0	1.389429	0	1.300388	0	0.642127	0	0.545663	1
1.272017	0	1.797901	0	2.077893	0	2.194427	0	1.968137	0	0.464460	1	1.367552	0
0.785239	0	2.091409	0	2.299405	0	1.801633	0	1.487486	0	0.246706	1	1.077123	0
0.480684	0	1.598983	0	1.700501	0	1.125718	0	0.990896	0	0.699391	0	0.288056	1
0.476804	0	1.602816	0	1.704081	0	1.123700	0	0.987656	0	0.699544	0	0.286872	1
0.303330	0	1.775968	0	1.864676	0	1.032919	0	0.838765	0	0.739809	0	0.278959	1
0.434516	0	1.824071	0	1.851454	0	0.726792	0	0.613025	0	1.050947	0	0.173220	1
0.394263	0	1.795900	0	1.837742	0	0.802275	0	0.669831	0	0.976069	0	0.142166	1
0.343672	0	1.735602	0	1.824903	0	1.041619	0	0.864307	0	0.737470	0	0.260486	1
0.376735	0	1.702258	0	1.793101	0	1.054837	0	0.889981	0	0.732074	0	0.254118	1
0.342998	0	1.736265	0	1.825592	0	1.041643	0	0.864017	0	0.737306	0	0.260921	1
0.516728	0	1.565691	0	1.650540	0	1.063849	0	0.961854	0	0.784435	0	0.216995	1
0.530829	0	1.567915	0	1.636855	0	0.999485	0	0.919128	0	0.855375	0	0.151979	1
0.548204	0	1.530519	0	1.626858	0	1.122309	0	1.019487	0	0.743690	0	0.274732	1
0.542633	0	1.536060	0	1.633283	0	1.124233	0	1.018304	0	0.737943	0	0.276945	1
0.275601	1	1.805624	0	1.890576	0	1.010383	0	0.807108	0	0.760178	0	0.281076	0
1.106990	0	2.852586	0	2.813852	0	0.717489	0	0.467776	1	1.914792	0	1.251914	0
0.788000	0	2.786357	0	2.814645	0	0.988606	0	0.527878	1	1.537612	0	1.130694	0
0.713211	0	1.973682	0	1.936824	0	0.377261	1	0.449595	0	1.403538	0	0.471620	0
1.106178	0	1.851897	0	1.741426	0	0.425753	1	0.797375	0	1.705881	0	0.726371	0
0.377165	0	1.896449	0	1.927704	0	0.723712	0	0.560411	0	1.046291	0	0.241048	1
0.433576	0	1.900181	0	1.919601	0	0.664551	0	0.529823	0	1.106007	0	0.255391	1
0.169102	1	2.017824	0	2.078956	0	0.872358	0	0.591337	0	0.926889	0	0.390180	0
0.466589	0	1.863835	0	1.879712	0	0.659938	0	0.556369	0	1.114949	0	0.228356	1
0.889776	0	2.486854	0	2.443168	0	0.385649	0	0.184855	1	1.693200	0	0.917909	0
1.124697	0	2.644018	0	2.573891	0	0.425753	0	0.418579	1	1.928256	0	1.127977	0

4. En la tabla 10 se retomaron los cálculos de los centroides porque hay variaciones en las distancias al realizar la comparación entre la tabla 8 y la tabla 9, finalizamos la etapa cuando ya no hay cambios, como se puede comprobar en la tabla 16.

Tabla 10 Tercera iteración con nuevos centroides

Iteración3_C1		Iteración3_C2		Iteración3_C3		Iteración3_C4		Iteración3_C5		Iteración3_C6		Iteración3_C7	
2.065486	1.209849	0.740752	2.739102	0.430980	2.545116	1.367598	0.922629	1.815649	0.442012	2.400218	1.925883	1.603225	1.342339
0.161206	1	2.159101	0	2.230034	0	0.729066	0	0.665069	0	0.930825	0	0.563995	0
0.110408	1	2.108416	0	2.206279	0	0.851253	0	0.832706	0	0.766472	0	0.590533	0
0.159153	1	2.004030	0	2.118438	0	0.898634	0	0.956975	0	0.637892	0	0.564490	0
0.300021	1	2.153420	0	2.282945	0	1.054321	0	1.016596	0	0.620663	0	0.746830	0
0.317504	1	2.337245	0	2.428064	0	0.945039	0	0.749322	0	0.924351	0	0.781753	0
0.064401	1	2.072307	0	2.151845	0	0.732524	0	0.746461	0	0.849220	0	0.502377	0
0.159020	1	2.012961	0	2.127301	0	0.901806	0	0.953837	0	0.641150	0	0.571232	0
0.534804	0	2.051165	0	2.224538	0	1.264147	0	1.308003	0	0.337222	1	0.873496	0
0.536780	0	2.044797	0	2.218716	0	1.264443	0	1.311758	0	0.330684	1	0.871908	0
0.542727	0	1.915867	0	2.093353	0	1.230082	0	1.343071	0	0.254122	1	0.807701	0
0.480961	0	2.109716	0	2.113940	0	0.359095	1	0.439886	0	1.252607	0	0.485326	0
0.313153	1	2.134196	0	2.170067	0	0.530389	0	0.510168	0	1.101700	0	0.493980	0
0.509208	0	2.435190	0	2.469478	0	0.739551	0	0.364940	1	1.268113	0	0.795270	0
0.814036	0	2.505553	0	2.481851	0	0.615918	0	0.050838	1	1.603713	0	0.904201	0
0.691484	0	2.383506	0	2.369474	0	0.520324	0	0.153498	1	1.481868	0	0.772530	0
0.749951	0	2.417593	0	2.395912	0	0.534249	0	0.123006	1	1.540306	0	0.817242	0
0.722173	0	2.285596	0	2.256478	0	0.390434	0	0.266717	1	1.504291	0	0.709202	0
0.694075	0	2.294979	0	2.271840	0	0.413009	0	0.246578	1	1.479098	0	0.705206	0
0.243453	0	1.849674	0	1.908981	0	0.555638	0	0.797752	0	0.896098	0	0.243344	1
0.272605	0	1.768921	0	1.842547	0	0.624048	0	0.901893	0	0.816180	0	0.224852	1
0.227294	1	1.798263	0	1.883465	0	0.685248	0	0.921564	0	0.758491	0	0.292303	0
0.241913	1	1.782506	0	1.869533	0	0.693251	0	0.938328	0	0.748544	0	0.289805	0

0.486833	0	1.536424	0	1.633607	0	0.759306	0	1.139391	0	0.737996	0	0.278109	1
0.105791	1	1.925037	0	2.006986	0	0.691773	0	0.832057	0	0.792344	0	0.377961	0
0.146425	1	1.895175	0	1.971942	0	0.656246	0	0.828391	0	0.813664	0	0.335292	0
0.053193	1	2.003400	0	2.082513	0	0.701633	0	0.777142	0	0.827827	0	0.437949	0
0.424690	0	1.599831	0	1.701159	0	0.767107	0	1.107211	0	0.700045	0	0.292597	1
0.492588	0	1.549481	0	1.671325	0	0.866909	0	1.206821	0	0.623778	0	0.387987	1
0.601967	0	1.423334	0	1.513045	0	0.764346	0	1.207195	0	0.812685	0	0.296531	1
0.268255	1	1.761476	0	1.863810	0	0.770571	0	1.009209	0	0.669788	0	0.346918	0
1.990170	0	0.522658	0	0.218032	1	1.695230	0	2.346522	0	2.021761	0	1.540591	0
1.899340	0	0.185864	1	0.528665	0	1.860148	0	2.454442	0	1.678783	0	1.543655	0
1.950714	0	0.138207	1	0.494599	0	1.898930	0	2.498033	0	1.734684	0	1.590094	0
2.061442	0	0.040116	1	0.349231	0	1.952559	0	2.569573	0	1.887979	0	1.677369	0
2.144167	0	0.148558	1	0.279490	0	2.003044	0	2.629133	0	1.992638	0	1.748179	0
1.936609	0	0.146428	1	0.285962	0	1.796507	0	2.419542	0	1.815694	0	1.537419	0
1.811615	0	0.329359	0	0.309050	1	1.627571	0	2.258611	0	1.758980	0	1.393279	0
2.160957	0	0.141352	1	0.338105	0	2.038530	0	2.660304	0	1.987560	0	1.772838	0
2.199708	0	0.291890	0	0.160022	1	2.003253	0	2.641654	0	2.098750	0	1.783657	0
2.154676	0	0.284243	0	0.117697	1	1.953086	0	2.592005	0	2.065067	0	1.736171	0
2.109865	0	0.372099	0	0.007239	1	1.870202	0	2.515168	0	2.066312	0	1.678010	0
1.869888	0	0.388486	0	0.241831	1	1.641686	0	2.282165	0	1.852145	0	1.437768	0
2.272158	0	0.334136	0	0.219634	1	2.076187	0	2.715167	0	2.161593	0	1.857260	0
2.204147	0	0.293154	0	0.164374	1	2.008160	0	2.646516	0	2.102122	0	1.788324	0
2.142692	0	0.306091	0	0.084002	1	1.930834	0	2.571496	0	2.065476	0	1.720461	0
2.487941	0	0.623818	0	0.378685	1	2.220094	0	2.871309	0	2.429636	0	2.051793	0
1.873220	0	0.390011	0	0.238811	1	1.643672	0	2.284430	0	1.856277	0	1.440733	0
2.198408	0	0.623740	0	0.258343	1	1.872549	0	2.527704	0	2.233963	0	1.743568	0
2.172052	0	0.550192	0	0.184742	1	1.870705	0	2.523790	0	2.184996	0	1.723101	0
1.005630	0	1.627194	0	1.536720	0	0.377302	1	1.029883	0	1.549908	0	0.576426	0

0.751190	0	1.538348	0	1.518993	0	0.394699	0	1.002761	0	1.239733	0	0.282941	1
0.536075	0	1.707439	0	1.717609	0	0.342890	0	0.833432	0	1.117775	0	0.138662	1
0.878855	0	1.362117	0	1.339714	0	0.561712	0	1.182141	0	1.262273	0	0.398126	1
0.788185	0	1.811857	0	1.759363	0	0.114574	1	0.770757	0	1.426466	0	0.444336	0
0.439351	0	1.734746	0	1.764689	0	0.416878	0	0.827890	0	1.027470	0	0.097190	1
0.719781	0	1.695108	0	1.664018	0	0.234595	1	0.853991	0	1.303818	0	0.316033	0
0.579935	0	1.584698	0	1.605545	0	0.447785	0	0.960020	0	1.066675	0	0.100850	1
0.663994	0	1.748367	0	1.725163	0	0.204007	1	0.793858	0	1.277125	0	0.294336	0
0.682040	0	1.837029	0	1.804937	0	0.112591	1	0.713394	0	1.338238	0	0.370218	0
0.434224	0	1.741885	0	1.771924	0	0.415559	0	0.821461	0	1.027605	0	0.103760	1
0.550011	0	1.605376	0	1.630571	0	0.450328	0	0.943783	0	1.046606	0	0.071630	1
0.686521	0	1.845049	0	1.811755	0	0.103136	1	0.706870	0	1.346024	0	0.379255	0
0.929597	0	1.655197	0	1.580118	0	0.310285	1	0.966725	0	1.489887	0	0.509326	0
0.686131	0	1.934136	0	1.898875	0	0.070407	1	0.621334	0	1.382185	0	0.440310	0
0.681785	0	1.934304	0	1.899863	0	0.074715	1	0.619957	0	1.378296	0	0.437454	0
0.790398	0	1.810157	0	1.757274	0	0.116832	1	0.773164	0	1.427797	0	0.445349	0
0.794651	0	1.805048	0	1.751480	0	0.122921	1	0.779424	0	1.429561	0	0.446336	0
0.551254	0	1.604806	0	1.629753	0	0.449781	0	0.944099	0	1.047730	0	0.072930	1
0.720128	0	1.935179	0	1.893478	0	0.037940	1	0.631102	0	1.413442	0	0.464792	0
0.677222	0	1.934482	0	1.900902	0	0.079243	1	0.618551	0	1.374213	0	0.434475	0
0.684308	0	1.840400	0	1.807722	0	0.108342	1	0.710738	0	1.341881	0	0.374333	0
0.560487	0	1.596735	0	1.620747	0	0.451083	0	0.950985	0	1.052193	0	0.081419	1
0.785887	0	1.384994	0	1.389316	0	0.564141	0	1.151153	0	1.157731	0	0.308875	1
0.697338	0	1.858796	0	1.822894	0	0.085008	1	0.696668	0	1.362098	0	0.397052	0
0.862325	0	2.575684	0	2.551699	0	0.684352	0	0.054876	1	1.649701	0	0.971764	0
1.037485	0	2.798811	0	2.772919	0	0.902208	0	0.265461	1	1.811856	0	1.190216	0
0.929431	0	2.697866	0	2.678757	0	0.813614	0	0.161782	1	1.706988	0	1.083069	0
1.001858	0	2.861335	0	2.857558	0	1.005708	0	0.349500	1	1.748307	0	1.228605	0

0.843295	0	1.662832	0	1.886184	0	1.416480	0	1.648764	0	0.187683	1	0.944162	0
0.696653	0	1.696875	0	1.895958	0	1.284069	0	1.502064	0	0.192752	1	0.818648	0
0.793720	0	1.724409	0	1.940805	0	1.394477	0	1.600952	0	0.123929	1	0.928681	0
0.810710	0	1.593888	0	1.809089	0	1.349850	0	1.610333	0	0.255597	1	0.873571	0
1.328365	0	1.898927	0	2.185309	0	1.947618	0	2.132787	0	0.538111	1	1.476817	0
1.031106	0	1.510450	0	1.761334	0	1.536609	0	1.827827	0	0.389341	1	1.055688	0
0.851267	0	1.427233	0	1.638149	0	1.297408	0	1.626120	0	0.426187	1	0.816098	0
0.650665	0	1.650896	0	1.838783	0	1.209516	0	1.450491	0	0.273106	1	0.740555	0
1.166944	0	2.274100	0	2.523098	0	1.888030	0	1.919210	0	0.509621	1	1.465737	0
0.784475	0	2.205650	0	2.407526	0	1.524710	0	1.522832	0	0.365821	1	1.135461	0
0.729804	0	2.267750	0	2.457044	0	1.480444	0	1.439472	0	0.449704	1	1.115412	0
0.789370	0	1.285098	0	1.444907	0	1.071027	0	1.487846	0	0.662843	0	0.600632	1
0.750022	0	1.308844	0	1.457630	0	1.022456	0	1.440214	0	0.674029	0	0.551455	1
0.719426	0	1.345796	0	1.495546	0	1.018352	0	1.420830	0	0.642127	0	0.542980	1
1.248063	0	1.797901	0	2.077893	0	1.846185	0	2.055072	0	0.464460	1	1.372177	0
0.781563	0	2.091409	0	2.299405	0	1.501197	0	1.551132	0	0.246706	1	1.088397	0
0.425605	0	1.598983	0	1.700501	0	0.768079	0	1.108378	0	0.699391	0	0.293446	1
0.421718	0	1.602816	0	1.704081	0	0.766470	0	1.104996	0	0.699544	0	0.292474	1
0.247745	1	1.775968	0	1.864676	0	0.701394	0	0.948634	0	0.739809	0	0.293086	0
0.397146	0	1.824071	0	1.851454	0	0.390788	0	0.742043	0	1.050947	0	0.185243	1
0.350890	0	1.795900	0	1.837742	0	0.464386	0	0.794921	0	0.976069	0	0.158223	1
0.288092	0	1.735602	0	1.824903	0	0.702344	0	0.976877	0	0.737470	0	0.273234	1
0.321185	0	1.702258	0	1.793101	0	0.709964	0	1.004280	0	0.732074	0	0.265325	1
0.287419	0	1.736265	0	1.825592	0	0.702494	0	0.976533	0	0.737306	0	0.273692	1
0.461169	0	1.565691	0	1.650540	0	0.699875	0	1.084748	0	0.784435	0	0.219144	1
0.476096	0	1.567915	0	1.636855	0	0.632794	0	1.045463	0	0.855375	0	0.151570	1
0.492735	0	1.530519	0	1.626858	0	0.757029	0	1.141371	0	0.743690	0	0.275717	1
0.487198	0	1.536060	0	1.633283	0	0.759556	0	1.139764	0	0.737943	0	0.278351	1

0.220113	1	1.805624	0	1.890576	0	0.684724	0	0.915939	0	0.760178	0	0.296262	0
1.135919	0	2.852586	0	2.813852	0	0.940524	0	0.342102	1	1.914792	0	1.261720	0
0.834329	0	2.786357	0	2.814645	0	1.026248	0	0.435110	1	1.537612	0	1.145454	0
0.693851	0	1.973682	0	1.936824	0	0.083452	1	0.585002	0	1.403538	0	0.473009	0
1.078170	0	1.851897	0	1.741426	0	0.337962	1	0.914533	0	1.705881	0	0.718137	0
0.345995	0	1.896449	0	1.927704	0	0.414267	0	0.684917	0	1.046291	0	0.254954	1
0.405109	0	1.900181	0	1.919601	0	0.352518	0	0.659188	0	1.106007	0	0.266498	1
0.151757	1	2.017824	0	2.078956	0	0.608808	0	0.689159	0	0.926889	0	0.407206	0
0.435057	0	1.863835	0	1.879712	0	0.331557	0	0.688067	0	1.114949	0	0.237667	1
0.903643	0	2.486854	0	2.443168	0	0.570136	0	0.185813	1	1.693200	0	0.925527	0
1.138829	0	2.644018	0	2.573891	0	0.726534	0	0.376968	1	1.928256	0	1.133932	0

En la tabla 11 se vuelven a retomar el cálculo de centroides porque existe variación entre las distancias de la tabla 9 y 10.

Tabla 11 Cuarta iteración con nuevos centroides

Iteración4_C1	Iteración4_C2	Iteración4_C3	Iteración4_C4	Iteración4_C5	Iteración4_C6	Iteración4_C7							
2.035030	1.238992	0.740752	2.739102	0.430980	2.545116	1.355579	1.004901	1.822258	0.411294	2.400218	1.925883	1.672762	1.432866
0.195717	1	2.159101	0	2.230034	0	0.731278	0	0.690711	0	0.930825	0	0.563030	0
0.146894	1	2.108416	0	2.206279	0	0.840476	0	0.857787	0	0.766472	0	0.556115	0
0.161906	1	2.004030	0	2.118438	0	0.875411	0	0.983263	0	0.637892	0	0.505050	0
0.321546	1	2.153420	0	2.282945	0	1.038656	0	1.039500	0	0.620663	0	0.688023	0
0.359575	1	2.337245	0	2.428064	0	0.953371	0	0.768505	0	0.924351	0	0.766099	0
0.098354	1	2.072307	0	2.151845	0	0.723613	0	0.773372	0	0.849220	0	0.486603	0
0.164086	1	2.012961	0	2.127301	0	0.879321	0	0.979976	0	0.641150	0	0.512743	0
0.537046	0	2.051165	0	2.224538	0	1.233200	0	1.331988	0	0.337222	1	0.783925	0
0.538480	0	2.044797	0	2.218716	0	1.233058	0	1.335835	0	0.330684	1	0.781753	0
0.534308	0	1.915867	0	2.093353	0	1.191111	0	1.368931	0	0.254122	1	0.708706	0
0.481596	0	2.109716	0	2.113940	0	0.390807	1	0.471307	0	1.252607	0	0.563067	0

0.324056	1	2.134196	0	2.170067	0	0.545150	0	0.539549	0	1.101700	0	0.536617	0
0.539637	0	2.435190	0	2.469478	0	0.779311	0	0.382959	1	1.268113	0	0.834454	0
0.830848	0	2.505553	0	2.481851	0	0.687847	0	0.069983	1	1.603713	0	0.984382	0
0.705648	0	2.383506	0	2.369474	0	0.585181	0	0.184722	1	1.481868	0	0.850923	0
0.763824	0	2.417593	0	2.395912	0	0.603999	0	0.151463	1	1.540306	0	0.899109	0
0.728270	0	2.285596	0	2.256478	0	0.461750	0	0.293855	1	1.504291	0	0.800354	0
0.701830	0	2.294979	0	2.271840	0	0.480838	0	0.275505	1	1.479098	0	0.792673	0
0.211158	1	1.849674	0	1.908981	0	0.523871	0	0.828428	0	0.896098	0	0.245234	0
0.231992	0	1.768921	0	1.842547	0	0.581477	0	0.932543	0	0.816180	0	0.179027	1
0.185184	1	1.798263	0	1.883465	0	0.645882	0	0.951644	0	0.758491	0	0.233816	0
0.199760	1	1.782506	0	1.869533	0	0.652131	0	0.968459	0	0.748544	0	0.225111	0
0.444904	0	1.536424	0	1.633607	0	0.696094	0	1.170454	0	0.737996	0	0.165020	1
0.066264	1	1.925037	0	2.006986	0	0.666784	0	0.861074	0	0.792344	0	0.345902	0
0.108727	1	1.895175	0	1.971942	0	0.628587	0	0.857984	0	0.813664	0	0.307555	0
0.050030	1	2.003400	0	2.082513	0	0.685749	0	0.805263	0	0.827827	0	0.417823	0
0.383152	0	1.599831	0	1.701159	0	0.709159	0	1.137979	0	0.700045	0	0.180030	1
0.452995	0	1.549481	0	1.671325	0	0.806468	0	1.237411	0	0.623778	0	0.273838	1
0.559828	0	1.423334	0	1.513045	0	0.692998	0	1.238548	0	0.812685	0	0.206104	1
0.228308	1	1.761476	0	1.863810	0	0.726992	0	1.038980	0	0.669788	0	0.262346	0
1.948617	0	0.522658	0	0.218032	1	1.620727	0	2.375174	0	2.021761	0	1.540399	0
1.857767	0	0.185864	1	0.528665	0	1.777391	0	2.485472	0	1.678783	0	1.496278	0
1.909063	0	0.138207	1	0.494599	0	1.816368	0	2.528991	0	1.734684	0	1.544714	0
2.019454	0	0.040116	1	0.349231	0	1.871151	0	2.600145	0	1.887979	0	1.641548	0
2.102073	0	0.148558	1	0.279490	0	1.922606	0	2.659424	0	1.992638	0	1.717741	0
1.894481	0	0.146428	1	0.285962	0	1.715620	0	2.449908	0	1.815694	0	1.507876	0
1.769493	0	0.329359	0	0.309050	1	1.547625	0	2.288649	0	1.758980	0	1.372960	0
2.118936	0	0.141352	1	0.338105	0	1.957614	0	2.690746	0	1.987560	0	1.739000	0
2.157561	0	0.291890	0	0.160022	1	1.924788	0	2.671389	0	2.098750	0	1.763007	0

2.112539	0	0.284243	0	0.117697	1	1.874711	0	2.621705	0	2.065067	0	1.716725	0
2.067856	0	0.372099	0	0.007239	1	1.793316	0	2.544454	0	2.066312	0	1.666031	0
1.827911	0	0.388486	0	0.241831	1	1.563518	0	2.311727	0	1.852145	0	1.425537	0
2.230007	0	0.334136	0	0.219634	1	1.997870	0	2.744878	0	2.161593	0	1.836147	0
2.161999	0	0.293154	0	0.164374	1	1.929687	0	2.676254	0	2.102122	0	1.767564	0
2.100577	0	0.306091	0	0.084002	1	1.852837	0	2.601088	0	2.065476	0	1.703005	0
2.445970	0	0.623818	0	0.378685	1	2.145652	0	2.900014	0	2.429636	0	2.042891	0
1.831248	0	0.390011	0	0.238811	1	1.565572	0	2.313975	0	1.856277	0	1.428734	0
2.157039	0	0.623740	0	0.258343	1	1.800719	0	2.555720	0	2.233963	0	1.747659	0
2.130447	0	0.550192	0	0.184742	1	1.797254	0	2.552205	0	2.184996	0	1.722584	0
0.977643	0	1.627194	0	1.536720	0	0.324821	1	1.055808	0	1.549908	0	0.675638	0
0.717388	0	1.538348	0	1.518993	0	0.311597	1	1.032791	0	1.239733	0	0.370488	0
0.505608	0	1.707439	0	1.717609	0	0.280459	0	0.864563	0	1.117775	0	0.252528	1
0.841445	0	1.362117	0	1.339714	0	0.479179	0	1.212087	0	1.262273	0	0.449226	1
0.766656	0	1.811857	0	1.759363	0	0.055280	1	0.798201	0	1.426466	0	0.558107	0
0.407315	0	1.734746	0	1.764689	0	0.365194	0	0.859306	0	1.027470	0	0.189945	1
0.692283	0	1.695108	0	1.664018	0	0.151683	1	0.883381	0	1.303818	0	0.426749	0
0.543758	0	1.584698	0	1.605545	0	0.374728	0	0.991221	0	1.066675	0	0.189426	1
0.638471	0	1.748367	0	1.725163	0	0.128671	1	0.823661	0	1.277125	0	0.408264	0
0.661469	0	1.837029	0	1.804937	0	0.057356	1	0.742483	0	1.338238	0	0.483931	0
0.402534	0	1.741885	0	1.771924	0	0.365087	0	0.852879	0	1.027605	0	0.194282	1
0.513872	0	1.605376	0	1.630571	0	0.380317	0	0.975070	0	1.046606	0	0.167864	1
0.666407	0	1.845049	0	1.811755	0	0.052395	1	0.735843	0	1.346024	0	0.492887	0
0.902222	0	1.655197	0	1.580118	0	0.251307	1	0.993434	0	1.489887	0	0.612350	0
0.671052	0	1.934136	0	1.898875	0	0.107540	1	0.649856	0	1.382185	0	0.551220	0
0.666734	0	1.934304	0	1.899863	0	0.109802	1	0.648564	0	1.378296	0	0.548185	0
0.768794	0	1.810157	0	1.757274	0	0.057952	1	0.800583	0	1.427797	0	0.559071	0
0.772820	0	1.805048	0	1.751480	0	0.063944	1	0.806813	0	1.429561	0	0.559923	0

0.515129	0	1.604806	0	1.629753	0	0.379645	0	0.975382	0	1.047730	0	0.169002	1
0.704952	0	1.935179	0	1.893478	0	0.097052	1	0.658914	0	1.413442	0	0.576817	0
0.662200	0	1.934482	0	1.900902	0	0.112300	1	0.647247	0	1.374213	0	0.545010	0
0.663932	0	1.840400	0	1.807722	0	0.054723	1	0.739772	0	1.341881	0	0.488017	0
0.524259	0	1.596735	0	1.620747	0	0.379939	0	0.982247	0	1.052193	0	0.173826	1
0.747329	0	1.384994	0	1.389316	0	0.481289	0	1.181809	0	1.157731	0	0.347250	1
0.678004	0	1.858796	0	1.822894	0	0.044678	1	0.725382	0	1.362098	0	0.510587	0
0.881382	0	2.575684	0	2.551699	0	0.757096	0	0.032482	1	1.649701	0	1.050103	0
1.061642	0	2.798811	0	2.772919	0	0.976963	0	0.234136	1	1.811856	0	1.264457	0
0.952893	0	2.697866	0	2.678757	0	0.885588	0	0.131605	1	1.706988	0	1.155682	0
1.031738	0	2.861335	0	2.857558	0	1.073757	0	0.324547	1	1.748307	0	1.289271	0
0.822156	0	1.662832	0	1.886184	0	1.360760	0	1.676749	0	0.187683	1	0.830590	0
0.675447	0	1.696875	0	1.895958	0	1.231518	0	1.530030	0	0.192752	1	0.706658	0
0.775056	0	1.724409	0	1.940805	0	1.341936	0	1.628450	0	0.123929	1	0.816404	0
0.786065	0	1.593888	0	1.809089	0	1.291642	0	1.638973	0	0.255597	1	0.759510	0
1.314958	0	1.898927	0	2.185309	0	1.893050	0	2.158880	0	0.538111	1	1.363254	0
1.005585	0	1.510450	0	1.761334	0	1.473472	0	1.856756	0	0.389341	1	0.942100	0
0.820181	0	1.427233	0	1.638149	0	1.232240	0	1.655950	0	0.426187	1	0.703519	0
0.625889	0	1.650896	0	1.838783	0	1.155259	0	1.479049	0	0.273106	1	0.627877	0
1.169113	0	2.274100	0	2.523098	0	1.851198	0	1.940954	0	0.509621	1	1.362854	0
0.791571	0	2.205650	0	2.407526	0	1.495457	0	1.544487	0	0.365821	1	1.043321	0
0.742136	0	2.267750	0	2.457044	0	1.456596	0	1.460010	0	0.449704	1	1.030535	0
0.750367	0	1.285098	0	1.444907	0	0.998754	0	1.518877	0	0.662843	0	0.499131	1
0.710427	0	1.308844	0	1.457630	0	0.950545	0	1.471285	0	0.674029	0	0.450052	1
0.680310	0	1.345796	0	1.495546	0	0.948095	0	1.451788	0	0.642127	0	0.437976	1
1.232029	0	1.797901	0	2.077893	0	1.789679	0	2.081882	0	0.464460	1	1.258245	0
0.782232	0	2.091409	0	2.299405	0	1.465935	0	1.574441	0	0.246706	1	0.989822	0
0.384080	0	1.598983	0	1.700501	0	0.710075	0	1.139146	0	0.699391	0	0.180811	1

0.380185	0	1.602816	0	1.704081	0	0.708760	0	1.135752	0	0.699544	0	0.180163	1
0.205625	1	1.775968	0	1.864676	0	0.659546	0	0.978754	0	0.739809	0	0.224676	0
0.371181	0	1.824071	0	1.851454	0	0.355323	0	0.773460	0	1.050947	0	0.263182	1
0.319934	0	1.795900	0	1.837742	0	0.424566	0	0.826232	0	0.976069	0	0.206710	1
0.245983	0	1.735602	0	1.824903	0	0.656304	0	1.007247	0	0.737470	0	0.193731	1
0.279125	0	1.702258	0	1.793101	0	0.660619	0	1.034799	0	0.732074	0	0.175268	1
0.245311	0	1.736265	0	1.825592	0	0.656522	0	1.006899	0	0.737306	0	0.194342	1
0.419023	0	1.565691	0	1.650540	0	0.637629	0	1.115909	0	0.784435	0	0.105345	1
0.434581	0	1.567915	0	1.636855	0	0.568638	0	1.076801	0	0.855375	0	0.044671	1
0.450766	0	1.530519	0	1.626858	0	0.693302	0	1.172462	0	0.743690	0	0.163208	1
0.445271	0	1.536060	0	1.633283	0	0.696320	0	1.170828	0	0.737943	0	0.165284	1
0.178022	1	1.805624	0	1.890576	0	0.646174	0	0.945971	0	0.760178	0	0.239787	0
1.158464	0	2.852586	0	2.813852	0	1.018732	0	0.311297	1	1.914792	0	1.341365	0
0.869750	0	2.786357	0	2.814645	0	1.080721	0	0.426321	1	1.537612	0	1.186069	0
0.681034	0	1.973682	0	1.936824	0	0.141995	1	0.613173	0	1.403538	0	0.582541	0
1.057652	0	1.851897	0	1.741426	0	0.339271	1	0.936211	0	1.705881	0	0.828723	0
0.325960	0	1.896449	0	1.927704	0	0.393082	0	0.716222	0	1.046291	0	0.316586	1
0.386701	0	1.900181	0	1.919601	0	0.334120	1	0.690609	0	1.106007	0	0.345953	0
0.150607	1	2.017824	0	2.078956	0	0.599229	0	0.718215	0	0.926889	0	0.414596	0
0.413797	0	1.863835	0	1.879712	0	0.304894	1	0.719461	0	1.114949	0	0.326325	0
0.915484	0	2.486854	0	2.443168	0	0.649207	0	0.190857	1	1.693200	0	1.016460	0
1.150432	0	2.644018	0	2.573891	0	0.809534	0	0.364535	1	1.928256	0	1.231057	0

En la tabla 12 se vuelven a retomar el cálculo de centroides porque existe variación entre las distancias de la tabla 10 y 11.

Tabla 12 Quinta iteración con nuevos centroides

Iteración5_C1		Iteración5_C2		Iteración5_C3		Iteración5_C4		Iteración5_C5		Iteración5_C6		Iteración5_C7	
2.023299	1.239032	0.740752	2.739102	0.430980	2.545116	1.383050	1.028466	1.822258	0.411294	2.400218	1.925883	1.680470	1.466734
0.199102	1	2.159101	0	2.230034	0	0.702748	0	0.690711	0	0.930825	0	0.581195	0
0.158099	1	2.108416	0	2.206279	0	0.808595	0	0.857787	0	0.766472	0	0.564762	0
0.171622	1	2.004030	0	2.118438	0	0.841205	0	0.983263	0	0.637892	0	0.505461	0
0.333010	1	2.153420	0	2.282945	0	1.005618	0	1.039500	0	0.620663	0	0.687501	0
0.368615	1	2.337245	0	2.428064	0	0.926170	0	0.768505	0	0.924351	0	0.779241	0
0.102623	1	2.072307	0	2.151845	0	0.692316	0	0.773372	0	0.849220	0	0.501403	0
0.174128	1	2.012961	0	2.127301	0	0.845227	0	0.979976	0	0.641150	0	0.513447	0
0.545932	0	2.051165	0	2.224538	0	1.197854	0	1.331988	0	0.337222	1	0.770189	0
0.547267	0	2.044797	0	2.218716	0	1.197667	0	1.335835	0	0.330684	1	0.767741	0
0.540994	0	1.915867	0	2.093353	0	1.155129	0	1.368931	0	0.254122	1	0.690295	0
0.474114	0	2.109716	0	2.113940	0	0.374310	1	0.471307	0	1.252607	0	0.596207	0
0.319291	1	2.134196	0	2.170067	0	0.521076	0	0.539549	0	1.101700	0	0.564542	0
0.540631	0	2.435190	0	2.469478	0	0.763906	0	0.382959	1	1.268113	0	0.860781	0
0.827162	0	2.505553	0	2.481851	0	0.691310	0	0.069983	1	1.603713	0	1.017426	0
0.701283	0	2.383506	0	2.369474	0	0.583900	0	0.184722	1	1.481868	0	0.883826	0
0.759350	0	2.417593	0	2.395912	0	0.606057	0	0.151463	1	1.540306	0	0.932402	0
0.721848	0	2.285596	0	2.256478	0	0.465762	0	0.293855	1	1.504291	0	0.834605	0
0.695818	0	2.294979	0	2.271840	0	0.482168	0	0.275505	1	1.479098	0	0.826613	0
0.199427	1	1.849674	0	1.908981	0	0.488829	0	0.828428	0	0.896098	0	0.268514	0
0.221572	0	1.768921	0	1.842547	0	0.545428	0	0.932543	0	0.816180	0	0.192079	1
0.176452	1	1.798263	0	1.883465	0	0.609995	0	0.951644	0	0.758491	0	0.239532	0
0.191439	1	1.782506	0	1.869533	0	0.616133	0	0.968459	0	0.748544	0	0.228651	0
0.437412	0	1.536424	0	1.633607	0	0.661200	0	1.170454	0	0.737996	0	0.131687	1

0.055197	1	1.925037	0	2.006986	0	0.632502	0	0.861074	0	0.792344	0	0.357774	0
0.097300	1	1.895175	0	1.971942	0	0.593957	0	0.857984	0	0.813664	0	0.321527	0
0.046625	1	2.003400	0	2.082513	0	0.653039	0	0.805263	0	0.827827	0	0.432138	0
0.376354	0	1.599831	0	1.701159	0	0.673489	0	1.137979	0	0.700045	0	0.153684	1
0.448072	0	1.549481	0	1.671325	0	0.771148	0	1.237411	0	0.623778	0	0.242864	1
0.551601	0	1.423334	0	1.513045	0	0.660393	0	1.238548	0	0.812685	0	0.174316	1
0.223373	1	1.761476	0	1.863810	0	0.690867	0	1.038980	0	0.669788	0	0.255395	0
1.938875	0	0.522658	0	0.218032	1	1.617779	0	2.375174	0	2.021761	0	1.527342	0
1.850758	0	0.185864	1	0.528665	0	1.761454	0	2.485472	0	1.678783	0	1.470958	0
1.901942	0	0.138207	1	0.494599	0	1.801157	0	2.528991	0	1.734684	0	1.519832	0
2.011725	0	0.040116	1	0.349231	0	1.858890	0	2.600145	0	1.887979	0	1.618920	0
2.094028	0	0.148558	1	0.279490	0	1.912130	0	2.659424	0	1.992638	0	1.696411	0
1.886281	0	0.146428	1	0.285962	0	1.704416	0	2.449908	0	1.815694	0	1.486928	0
1.760696	0	0.329359	0	0.309050	1	1.538073	0	2.288649	0	1.758980	0	1.354550	0
2.111122	0	0.141352	1	0.338105	0	1.946290	0	2.690746	0	1.987560	0	1.716803	0
2.148936	0	0.291890	0	0.160022	1	1.917223	0	2.671389	0	2.098750	0	1.744174	0
2.103829	0	0.284243	0	0.117697	1	1.867279	0	2.621705	0	2.065067	0	1.698241	0
2.058721	0	0.372099	0	0.007239	1	1.787738	0	2.544454	0	2.066312	0	1.649588	0
1.818713	0	0.388486	0	0.241831	1	1.556476	0	2.311727	0	1.852145	0	1.409250	0
2.221432	0	0.334136	0	0.219634	1	1.990477	0	2.744878	0	2.161593	0	1.817145	0
2.153382	0	0.293154	0	0.164374	1	1.922111	0	2.676254	0	2.102122	0	1.748699	0
2.091750	0	0.306091	0	0.084002	1	1.845899	0	2.601088	0	2.065476	0	1.685058	0
2.436758	0	0.623818	0	0.378685	1	2.142566	0	2.900014	0	2.429636	0	2.027041	0
1.822040	0	0.390011	0	0.238811	1	1.558613	0	2.313975	0	1.856277	0	1.412507	0
2.147140	0	0.623740	0	0.258343	1	1.800191	0	2.555720	0	2.233963	0	1.735638	0
2.120755	0	0.550192	0	0.184742	1	1.795248	0	2.552205	0	2.184996	0	1.709273	0
0.965973	0	1.627194	0	1.536720	0	0.341167	1	1.055808	0	1.549908	0	0.697624	0
0.705719	0	1.538348	0	1.518993	0	0.292692	1	1.032791	0	1.239733	0	0.389729	0

0.493883	0	1.707439	0	1.717609	0	0.245031	1	0.864563	0	1.117775	0	0.285089	0
0.830144	0	1.362117	0	1.339714	0	0.465172	0	1.212087	0	1.262273	0	0.455721	1
0.755451	0	1.811857	0	1.759363	0	0.082505	1	0.798201	0	1.426466	0	0.588760	0
0.395586	0	1.734746	0	1.764689	0	0.329009	0	0.859306	0	1.027470	0	0.224676	1
0.680642	0	1.695108	0	1.664018	0	0.132709	1	0.883381	0	1.303818	0	0.455039	0
0.532269	0	1.584698	0	1.605545	0	0.342648	0	0.991221	0	1.066675	0	0.212964	1
0.626947	0	1.748367	0	1.725163	0	0.097335	1	0.823661	0	1.277125	0	0.439301	0
0.650377	0	1.837029	0	1.804937	0	0.025203	1	0.742483	0	1.338238	0	0.516646	0
0.390803	0	1.741885	0	1.771924	0	0.328894	0	0.852879	0	1.027605	0	0.229013	1
0.502376	0	1.605376	0	1.630571	0	0.346974	0	0.975070	0	1.046606	0	0.194201	1
0.655364	0	1.845049	0	1.811755	0	0.025343	1	0.735843	0	1.346024	0	0.525678	0
0.890579	0	1.655197	0	1.580118	0	0.265221	1	0.993434	0	1.489887	0	0.636198	0
0.660637	0	1.934136	0	1.898875	0	0.108236	1	0.649856	0	1.382185	0	0.585179	0
0.656324	0	1.934304	0	1.899863	0	0.109147	1	0.648564	0	1.378296	0	0.582184	0
0.757582	0	1.810157	0	1.757274	0	0.084870	1	0.800583	0	1.427797	0	0.589651	0
0.761585	0	1.805048	0	1.751480	0	0.089730	1	0.806813	0	1.429561	0	0.590318	0
0.503632	0	1.604806	0	1.629753	0	0.346347	0	0.975382	0	1.047730	0	0.195250	1
0.694515	0	1.935179	0	1.893478	0	0.109117	1	0.658914	0	1.413442	0	0.610467	0
0.651796	0	1.934482	0	1.900902	0	0.110271	1	0.647247	0	1.374213	0	0.579051	0
0.652860	0	1.840400	0	1.807722	0	0.024479	1	0.739772	0	1.341881	0	0.520759	0
0.512775	0	1.596735	0	1.620747	0	0.347024	0	0.982247	0	1.052193	0	0.198993	1
0.736251	0	1.384994	0	1.389316	0	0.460142	0	1.181809	0	1.157731	0	0.351449	1
0.667046	0	1.858796	0	1.822894	0	0.032178	1	0.725382	0	1.362098	0	0.543461	0
0.878333	0	2.575684	0	2.551699	0	0.761046	0	0.032482	1	1.649701	0	1.082878	0
1.060146	0	2.798811	0	2.772919	0	0.982305	0	0.234136	1	1.811856	0	1.296564	0
0.951198	0	2.697866	0	2.678757	0	0.888621	0	0.131605	1	1.706988	0	1.187599	0
1.032303	0	2.861335	0	2.857558	0	1.073442	0	0.324547	1	1.748307	0	1.319161	0
0.825211	0	1.662832	0	1.886184	0	1.324973	0	1.676749	0	0.187683	1	0.800972	0

0.678552	0	1.696875	0	1.895958	0	1.195490	0	1.530030	0	0.192752	1	0.679142	0
0.778886	0	1.724409	0	1.940805	0	1.305915	0	1.628450	0	0.123929	1	0.788495	0
0.787985	0	1.593888	0	1.809089	0	1.256110	0	1.638973	0	0.255597	1	0.728774	0
1.320117	0	1.898927	0	2.185309	0	1.857195	0	2.158880	0	0.538111	1	1.333527	0
1.007128	0	1.510450	0	1.761334	0	1.438725	0	1.856756	0	0.389341	1	0.909263	0
0.819653	0	1.427233	0	1.638149	0	1.197893	0	1.655950	0	0.426187	1	0.669939	0
0.627849	0	1.650896	0	1.838783	0	1.119334	0	1.479049	0	0.273106	1	0.599656	0
1.177791	0	2.274100	0	2.523098	0	1.815292	0	1.940954	0	0.509621	1	1.341459	0
0.801178	0	2.205650	0	2.407526	0	1.460242	0	1.544487	0	0.365821	1	1.027853	0
0.752524	0	2.267750	0	2.457044	0	1.422062	0	1.460010	0	0.449704	1	1.018463	0
0.745754	0	1.285098	0	1.444907	0	0.966616	0	1.518877	0	0.662843	0	0.464562	1
0.705396	0	1.308844	0	1.457630	0	0.918245	0	1.471285	0	0.674029	0	0.415511	1
0.675638	0	1.345796	0	1.495546	0	0.915160	0	1.451788	0	0.642127	0	0.403250	1
1.236488	0	1.797901	0	2.077893	0	1.753995	0	2.081882	0	0.464460	1	1.227808	0
0.790712	0	2.091409	0	2.299405	0	1.430166	0	1.574441	0	0.246706	1	0.971143	0
0.377299	0	1.598983	0	1.700501	0	0.674412	0	1.139146	0	0.699391	0	0.154340	1
0.373396	0	1.602816	0	1.704081	0	0.673064	0	1.135752	0	0.699544	0	0.154224	1
0.197663	1	1.775968	0	1.864676	0	0.623508	0	0.978754	0	0.739809	0	0.226641	0
0.359671	0	1.824071	0	1.851454	0	0.320159	0	0.773460	0	1.050947	0	0.297211	1
0.308208	0	1.795900	0	1.837742	0	0.388808	0	0.826232	0	0.976069	0	0.238714	1
0.238039	0	1.735602	0	1.824903	0	0.620126	0	1.007247	0	0.737470	0	0.191821	1
0.271354	0	1.702258	0	1.793101	0	0.624435	0	1.034799	0	0.732074	0	0.168392	1
0.237373	0	1.736265	0	1.825592	0	0.620346	0	1.006899	0	0.737306	0	0.192480	1
0.410450	0	1.565691	0	1.650540	0	0.602533	0	1.115909	0	0.784435	0	0.073159	1
0.424801	0	1.567915	0	1.636855	0	0.533865	0	1.076801	0	0.855375	0	0.013865	1
0.443182	0	1.530519	0	1.626858	0	0.658507	0	1.172462	0	0.743690	0	0.129439	1
0.437782	0	1.536060	0	1.633283	0	0.661431	0	1.170828	0	0.737943	0	0.131931	1
0.169217	1	1.805624	0	1.890576	0	0.610344	0	0.945971	0	0.760178	0	0.246061	0

1.156419	0	2.852586	0	2.813852	0	1.027650	0	0.311297	1	1.914792	0	1.374200	0
0.872740	0	2.786357	0	2.814645	0	1.071880	0	0.426321	1	1.537612	0	1.212163	0
0.670946	0	1.973682	0	1.936824	0	0.146971	1	0.613173	0	1.403538	0	0.616776	0
1.046523	0	1.851897	0	1.741426	0	0.372541	1	0.936211	0	1.705881	0	0.856343	0
0.315058	1	1.896449	0	1.927704	0	0.360136	0	0.716222	0	1.046291	0	0.348735	0
0.375960	0	1.900181	0	1.919601	0	0.302131	1	0.690609	0	1.106007	0	0.379758	0
0.143839	1	2.017824	0	2.078956	0	0.568051	0	0.718215	0	0.926889	0	0.436082	0
0.402701	0	1.863835	0	1.879712	0	0.271381	1	0.719461	0	1.114949	0	0.360790	0
0.910410	0	2.486854	0	2.443168	0	0.659785	0	0.190857	1	1.693200	0	1.050632	0
1.145236	0	2.644018	0	2.573891	0	0.827742	0	0.364535	1	1.928256	0	1.265632	0

En la tabla 13 se vuelven a retomar el cálculo de centroides porque existe variación entre las distancias de la tabla 11 y 12.

Tabla 13 Sexta iteración con nuevos centroides

Iteración6_C1		Iteración6_C2		Iteración6_C3		Iteración6_C4		Iteración6_C5		Iteración6_C6		Iteración6_C7	
2.007911	1.232855	0.740752	2.739102	0.430980	2.545116	1.389712	1.037391	1.822258	0.411294	2.400218	1.925883	1.684189	1.488474
0.198729	1	2.159101	0	2.230034	0	0.695872	0	0.690711	0	0.930825	0	0.594464	0
0.171359	1	2.108416	0	2.206279	0	0.800287	0	0.857787	0	0.766472	0	0.572411	0
0.187932	1	2.004030	0	2.118438	0	0.831793	0	0.983263	0	0.637892	0	0.508150	0
0.349350	1	2.153420	0	2.282945	0	0.996765	0	1.039500	0	0.620663	0	0.689276	0
0.376915	1	2.337245	0	2.428064	0	0.919771	0	0.768505	0	0.924351	0	0.789341	0
0.104625	1	2.072307	0	2.151845	0	0.684285	0	0.773372	0	0.849220	0	0.512833	0
0.190560	1	2.012961	0	2.127301	0	0.835871	0	0.979976	0	0.641150	0	0.516300	0
0.561701	0	2.051165	0	2.224538	0	1.187790	0	1.331988	0	0.337222	1	0.763294	0
0.562970	0	2.044797	0	2.218716	0	1.187575	0	1.335835	0	0.330684	1	0.760665	0
0.555023	0	1.915867	0	2.093353	0	1.144600	0	1.368931	0	0.254122	1	0.680331	0
0.459659	0	2.109716	0	2.113940	0	0.372146	1	0.471307	0	1.252607	0	0.617647	0
0.307773	1	2.134196	0	2.170067	0	0.516013	0	0.539549	0	1.101700	0	0.583301	0

0.536132	0	2.435190	0	2.469478	0	0.761796	0	0.382959	1	1.268113	0	0.878531	0
0.816603	0	2.505553	0	2.481851	0	0.695130	0	0.069983	1	1.603713	0	1.038791	0
0.689981	0	2.383506	0	2.369474	0	0.586389	0	0.184722	1	1.481868	0	0.905128	0
0.747910	0	2.417593	0	2.395912	0	0.609519	0	0.151463	1	1.540306	0	0.953893	0
0.708350	0	2.285596	0	2.256478	0	0.469909	0	0.293855	1	1.504291	0	0.856533	0
0.682740	0	2.294979	0	2.271840	0	0.485522	0	0.275505	1	1.479098	0	0.848409	0
0.184166	1	1.849674	0	1.908981	0	0.478992	0	0.828428	0	0.896098	0	0.285289	0
0.211317	0	1.768921	0	1.842547	0	0.534843	0	0.932543	0	0.816180	0	0.204028	1
0.170169	1	1.798263	0	1.883465	0	0.599563	0	0.951644	0	0.758491	0	0.246781	0
0.185898	1	1.782506	0	1.869533	0	0.605599	0	0.968459	0	0.748544	0	0.234713	0
0.432846	0	1.536424	0	1.633607	0	0.650080	0	1.170454	0	0.737996	0	0.111451	1
0.044619	1	1.925037	0	2.006986	0	0.623070	0	0.861074	0	0.792344	0	0.367889	0
0.084431	1	1.895175	0	1.971942	0	0.584343	0	0.857984	0	0.813664	0	0.333006	0
0.040325	1	2.003400	0	2.082513	0	0.644378	0	0.805263	0	0.827827	0	0.443438	0
0.373080	0	1.599831	0	1.701159	0	0.662363	0	1.137979	0	0.700045	0	0.139839	1
0.447780	0	1.549481	0	1.671325	0	0.760010	0	1.237411	0	0.623778	0	0.224424	1
0.545597	0	1.423334	0	1.513045	0	0.649581	0	1.238548	0	0.812685	0	0.154231	1
0.223569	1	1.761476	0	1.863810	0	0.680182	0	1.038980	0	0.669788	0	0.254564	0
1.929657	0	0.522658	0	0.218032	1	1.614619	0	2.375174	0	2.021761	0	1.518295	0
1.846682	0	0.185864	1	0.528665	0	1.754583	0	2.485472	0	1.678783	0	1.454269	0
1.897672	0	0.138207	1	0.494599	0	1.794481	0	2.528991	0	1.734684	0	1.503407	0
2.006382	0	0.040116	1	0.349231	0	1.853027	0	2.600145	0	1.887979	0	1.603873	0
2.088112	0	0.148558	1	0.279490	0	1.906765	0	2.659424	0	1.992638	0	1.682159	0
1.880092	0	0.146428	1	0.285962	0	1.698859	0	2.449908	0	1.815694	0	1.472944	0
1.753389	0	0.329359	0	0.309050	1	1.532998	0	2.288649	0	1.758980	0	1.342180	0
2.105623	0	0.141352	1	0.338105	0	1.940684	0	2.690746	0	1.987560	0	1.702010	0
2.141940	0	0.291890	0	0.160022	1	1.912689	0	2.671389	0	2.098750	0	1.731468	0
2.096673	0	0.284243	0	0.117697	1	1.862787	0	2.621705	0	2.065067	0	1.685760	0

2.050741	0	0.372099	0	0.007239	1	1.783789	0	2.544454	0	2.066312	0	1.638391	0
1.810618	0	0.388486	0	0.241831	1	1.552121	0	2.311727	0	1.852145	0	1.398199	0
2.214530	0	0.334136	0	0.219634	1	1.985989	0	2.744878	0	2.161593	0	1.804324	0
2.146401	0	0.293154	0	0.164374	1	1.917574	0	2.676254	0	2.102122	0	1.735973	0
2.084370	0	0.306091	0	0.084002	1	1.841551	0	2.601088	0	2.065476	0	1.672915	0
2.428611	0	0.623818	0	0.378685	1	2.139328	0	2.900014	0	2.429636	0	2.016160	0
1.813924	0	0.390011	0	0.238811	1	1.554282	0	2.313975	0	1.856277	0	1.401493	0
2.137578	0	0.623740	0	0.258343	1	1.797738	0	2.555720	0	2.233963	0	1.727202	0
2.111636	0	0.550192	0	0.184742	1	1.792353	0	2.552205	0	2.184996	0	1.700030	0
0.950062	0	1.627194	0	1.536720	0	0.344580	1	1.055808	0	1.549908	0	0.711197	0
0.691125	0	1.538348	0	1.518993	0	0.285487	1	1.032791	0	1.239733	0	0.402066	0
0.478357	0	1.707439	0	1.717609	0	0.233893	1	0.864563	0	1.117775	0	0.305618	0
0.817092	0	1.362117	0	1.339714	0	0.459141	1	1.212087	0	1.262273	0	0.459994	0
0.738927	0	1.811857	0	1.759363	0	0.090760	1	0.798201	0	1.426466	0	0.607892	0
0.380403	0	1.734746	0	1.764689	0	0.318095	0	0.859306	0	1.027470	0	0.246687	1
0.664631	0	1.695108	0	1.664018	0	0.126188	1	0.883381	0	1.303818	0	0.472775	0
0.518582	0	1.584698	0	1.605545	0	0.331973	0	0.991221	0	1.066675	0	0.228598	1
0.610689	0	1.748367	0	1.725163	0	0.087138	1	0.823661	0	1.277125	0	0.458760	0
0.633817	0	1.837029	0	1.804937	0	0.020941	1	0.742483	0	1.338238	0	0.537146	0
0.375528	0	1.741885	0	1.771924	0	0.318033	0	0.852879	0	1.027605	0	0.251051	1
0.488668	0	1.605376	0	1.630571	0	0.336063	0	0.975070	0	1.046606	0	0.211493	1
0.638794	0	1.845049	0	1.811755	0	0.025291	1	0.735843	0	1.346024	0	0.546226	0
0.874573	0	1.655197	0	1.580118	0	0.268065	1	0.993434	0	1.489887	0	0.650992	0
0.644135	0	1.934136	0	1.898875	0	0.113118	1	0.649856	0	1.382185	0	0.606521	0
0.639824	0	1.934304	0	1.899863	0	0.113638	1	0.648564	0	1.378296	0	0.603556	0
0.741060	0	1.810157	0	1.757274	0	0.093004	1	0.800583	0	1.427797	0	0.608737	0
0.745073	0	1.805048	0	1.751480	0	0.097485	1	0.806813	0	1.429561	0	0.609284	0
0.489917	0	1.604806	0	1.629753	0	0.335443	0	0.975382	0	1.047730	0	0.212489	1

0.678006	0	1.935179	0	1.893478	0	0.116779	1	0.658914	0	1.413442	0	0.631585	0
0.635297	0	1.934482	0	1.900902	0	0.114350	1	0.647247	0	1.374213	0	0.600453	0
0.636295	0	1.840400	0	1.807722	0	0.022173	1	0.739772	0	1.341881	0	0.541277	0
0.499108	0	1.596735	0	1.620747	0	0.336188	0	0.982247	0	1.052193	0	0.215619	1
0.723901	0	1.384994	0	1.389316	0	0.452117	0	1.181809	0	1.157731	0	0.354680	1
0.650466	0	1.858796	0	1.822894	0	0.037674	1	0.725382	0	1.362098	0	0.564060	0
0.868510	0	2.575684	0	2.551699	0	0.764974	0	0.032482	1	1.649701	0	1.104105	0
1.052187	0	2.798811	0	2.772919	0	0.986562	0	0.234136	1	1.811856	0	1.317435	0
0.943009	0	2.697866	0	2.678757	0	0.892238	0	0.131605	1	1.706988	0	1.208376	0
1.027028	0	2.861335	0	2.857558	0	1.076022	0	0.324547	1	1.748307	0	1.338830	0
0.835423	0	1.662832	0	1.886184	0	1.313862	0	1.676749	0	0.187683	1	0.783065	0
0.688861	0	1.696875	0	1.895958	0	1.184438	0	1.530030	0	0.192752	1	0.662813	0
0.789979	0	1.724409	0	1.940805	0	1.294860	0	1.628450	0	0.123929	1	0.771830	0
0.796874	0	1.593888	0	1.809089	0	1.244976	0	1.638973	0	0.255597	1	0.710064	0
1.332561	0	1.898927	0	2.185309	0	1.846095	0	2.158880	0	0.538111	1	1.315443	0
1.015510	0	1.510450	0	1.761334	0	1.427613	0	1.856756	0	0.389341	1	0.888925	0
0.825445	0	1.427233	0	1.638149	0	1.186817	0	1.655950	0	0.426187	1	0.649036	0
0.636843	0	1.650896	0	1.838783	0	1.108249	0	1.479049	0	0.273106	1	0.582864	0
1.193384	0	2.274100	0	2.523098	0	1.804827	0	1.940954	0	0.509621	1	1.329147	0
0.817362	0	2.205650	0	2.407526	0	1.450254	0	1.544487	0	0.365821	1	1.019629	0
0.769034	0	2.267750	0	2.457044	0	1.412454	0	1.460010	0	0.449704	1	1.012491	0
0.745743	0	1.285098	0	1.444907	0	0.955874	0	1.518877	0	0.662843	0	0.442531	1
0.704753	0	1.308844	0	1.457630	0	0.907475	0	1.471285	0	0.674029	0	0.393491	1
0.675568	0	1.345796	0	1.495546	0	0.904282	0	1.451788	0	0.642127	0	0.381206	1
1.248199	0	1.797901	0	2.077893	0	1.742872	0	2.081882	0	0.464460	1	1.209223	0
0.806179	0	2.091409	0	2.299405	0	1.419812	0	1.574441	0	0.246706	1	0.960796	0
0.374054	0	1.598983	0	1.700501	0	0.663286	0	1.139146	0	0.699391	0	0.140378	1
0.370143	0	1.602816	0	1.704081	0	0.661941	0	1.135752	0	0.699544	0	0.140720	1

0.192779	1	1.775968	0	1.864676	0	0.612931	0	0.978754	0	0.739809	0	0.231772	0
0.343394	0	1.824071	0	1.851454	0	0.310284	1	0.773460	0	1.050947	0	0.319078	0
0.292712	0	1.795900	0	1.837742	0	0.378500	0	0.826232	0	0.976069	0	0.259771	1
0.233021	0	1.735602	0	1.824903	0	0.609345	0	1.007247	0	0.737470	0	0.195036	1
0.266558	0	1.702258	0	1.793101	0	0.613517	0	1.034799	0	0.732074	0	0.168851	1
0.232369	0	1.736265	0	1.825592	0	0.609567	0	1.006899	0	0.737306	0	0.195715	1
0.403902	0	1.565691	0	1.650540	0	0.591401	0	1.115909	0	0.784435	0	0.055135	1
0.415762	0	1.567915	0	1.636855	0	0.522755	0	1.076801	0	0.855375	0	0.016559	1
0.438442	0	1.530519	0	1.626858	0	0.647394	0	1.172462	0	0.743690	0	0.108727	1
0.433219	0	1.536060	0	1.633283	0	0.650311	0	1.170828	0	0.737943	0	0.111672	1
0.162829	1	1.805624	0	1.890576	0	0.599961	0	0.945971	0	0.760178	0	0.253578	0
1.147763	0	2.852586	0	2.813852	0	1.032905	0	0.311297	1	1.914792	0	1.395448	0
0.870894	0	2.786357	0	2.814645	0	1.071828	0	0.426321	1	1.537612	0	1.229702	0
0.654568	0	1.973682	0	1.936824	0	0.152452	1	0.613173	0	1.403538	0	0.638321	0
1.029973	0	1.851897	0	1.741426	0	0.381750	1	0.936211	0	1.705881	0	0.873418	0
0.298476	1	1.896449	0	1.927704	0	0.351459	0	0.716222	0	1.046291	0	0.369771	0
0.359388	0	1.900181	0	1.919601	0	0.293961	1	0.690609	0	1.106007	0	0.401524	0
0.130527	1	2.017824	0	2.078956	0	0.560103	0	0.718215	0	0.926889	0	0.451451	0
0.386140	0	1.863835	0	1.879712	0	0.262480	1	0.719461	0	1.114949	0	0.382803	0
0.898282	0	2.486854	0	2.443168	0	0.665613	0	0.190857	1	1.693200	0	1.072525	0
1.132949	0	2.644018	0	2.573891	0	0.835508	0	0.364535	1	1.928256	0	1.287670	0

En la tabla 14 se vuelven a retomar el cálculo de centroides porque existe variación entre las distancias de la tabla 12 y 13.

Tabla 14 Séptima iteración con nuevos centroides

Iteración7_C1		Iteración7_C2		Iteración7_C3		Iteración7_C4		Iteración7_C5		Iteración7_C6		Iteración7_C7	
2.007911	1.232855	0.740752	2.739102	0.430980	2.545116	1.394533	1.060757	1.822258	0.411294	2.400218	1.925883	1.703119	1.502130
0.198729	1	2.159101	0	2.230034	0	0.691024	0	0.690711	0	0.930825	0	0.592214	0

0.171359	1	2.108416	0	2.206279	0	0.791310	0	0.857787	0	0.766472	0	0.563639	0
0.187932	1	2.004030	0	2.118438	0	0.819259	0	0.983263	0	0.637892	0	0.494706	0
0.349350	1	2.153420	0	2.282945	0	0.986024	0	1.039500	0	0.620663	0	0.674845	0
0.376915	1	2.337245	0	2.428064	0	0.916051	0	0.768505	0	0.924351	0	0.783024	0
0.104625	1	2.072307	0	2.151845	0	0.676203	0	0.773372	0	0.849220	0	0.508407	0
0.190560	1	2.012961	0	2.127301	0	0.823532	0	0.979976	0	0.641150	0	0.502990	0
0.561701	0	2.051165	0	2.224538	0	1.172778	0	1.331988	0	0.337222	1	0.742782	0
0.562970	0	2.044797	0	2.218716	0	1.172451	0	1.335835	0	0.330684	1	0.740056	0
0.555023	0	1.915867	0	2.093353	0	1.127620	0	1.368931	0	0.254122	1	0.658371	0
0.459659	0	2.109716	0	2.113940	0	0.378912	1	0.471307	0	1.252607	0	0.630392	0
0.307773	1	2.134196	0	2.170067	0	0.515978	0	0.539549	0	1.101700	0	0.589522	0
0.536132	0	2.435190	0	2.469478	0	0.768146	0	0.382959	1	1.268113	0	0.882697	0
0.816603	0	2.505553	0	2.481851	0	0.712363	0	0.069983	1	1.603713	0	1.051103	0
0.689981	0	2.383506	0	2.369474	0	0.601540	0	0.184722	1	1.481868	0	0.917254	0
0.747910	0	2.417593	0	2.395912	0	0.626236	0	0.151463	1	1.540306	0	0.966694	0
0.708350	0	2.285596	0	2.256478	0	0.487784	0	0.293855	1	1.504291	0	0.871542	0
0.682740	0	2.294979	0	2.271840	0	0.502165	0	0.275505	1	1.479098	0	0.862604	0
0.184166	1	1.849674	0	1.908981	0	0.465146	0	0.828428	0	0.896098	0	0.288860	0
0.211317	0	1.768921	0	1.842547	0	0.517764	0	0.932543	0	0.816180	0	0.201682	1
0.170169	1	1.798263	0	1.883465	0	0.583173	0	0.951644	0	0.758491	0	0.238682	0
0.185898	1	1.782506	0	1.869533	0	0.588728	0	0.968459	0	0.748544	0	0.225487	0
0.432846	0	1.536424	0	1.633607	0	0.627998	0	1.170454	0	0.737996	0	0.089686	1
0.044619	1	1.925037	0	2.006986	0	0.610576	0	0.861074	0	0.792344	0	0.362288	0
0.084431	1	1.895175	0	1.971942	0	0.571228	0	0.857984	0	0.813664	0	0.329152	0
0.040325	1	2.003400	0	2.082513	0	0.634417	0	0.805263	0	0.827827	0	0.439003	0
0.373080	0	1.599831	0	1.701159	0	0.641344	0	1.137979	0	0.700045	0	0.116928	1
0.447780	0	1.549481	0	1.671325	0	0.738422	0	1.237411	0	0.623778	0	0.201455	1
0.545597	0	1.423334	0	1.513045	0	0.626184	0	1.238548	0	0.812685	0	0.148096	1

0.223569	1	1.761476	0	1.863810	0	0.662535	0	1.038980	0	0.669788	0	0.238888	0
1.929657	0	0.522658	0	0.218032	1	1.598949	0	2.375174	0	2.021761	0	1.526684	0
1.846682	0	0.185864	1	0.528665	0	1.733353	0	2.485472	0	1.678783	0	1.452666	0
1.897672	0	0.138207	1	0.494599	0	1.773497	0	2.528991	0	1.734684	0	1.502218	0
2.006382	0	0.040116	1	0.349231	0	1.833132	0	2.600145	0	1.887979	0	1.604759	0
2.088112	0	0.148558	1	0.279490	0	1.887582	0	2.659424	0	1.992638	0	1.684159	0
1.880092	0	0.146428	1	0.285962	0	1.679408	0	2.449908	0	1.815694	0	1.475353	0
1.753389	0	0.329359	0	0.309050	1	1.514268	0	2.288649	0	1.758980	0	1.346734	0
2.105623	0	0.141352	1	0.338105	0	1.921150	0	2.690746	0	1.987560	0	1.703247	0
2.141940	0	0.291890	0	0.160022	1	1.894762	0	2.671389	0	2.098750	0	1.735518	0
2.096673	0	0.284243	0	0.117697	1	1.844927	0	2.621705	0	2.065067	0	1.690103	0
2.050741	0	0.372099	0	0.007239	1	1.766799	0	2.544454	0	2.066312	0	1.644316	0
1.810618	0	0.388486	0	0.241831	1	1.534499	0	2.311727	0	1.852145	0	1.404347	0
2.214530	0	0.334136	0	0.219634	1	1.968127	0	2.744878	0	2.161593	0	1.808216	0
2.146401	0	0.293154	0	0.164374	1	1.899640	0	2.676254	0	2.102122	0	1.739995	0
2.084370	0	0.306091	0	0.084002	1	1.823919	0	2.601088	0	2.065476	0	1.677684	0
2.428611	0	0.623818	0	0.378685	1	2.123487	0	2.900014	0	2.429636	0	2.022406	0
1.813924	0	0.390011	0	0.238811	1	1.536699	0	2.313975	0	1.856277	0	1.407684	0
2.137578	0	0.623740	0	0.258343	1	1.783271	0	2.555720	0	2.233963	0	1.736207	0
2.111636	0	0.550192	0	0.184742	1	1.777119	0	2.552205	0	2.184996	0	1.708173	0
0.950062	0	1.627194	0	1.536720	0	0.342761	1	1.055808	0	1.549908	0	0.734378	0
0.691125	0	1.538348	0	1.518993	0	0.264143	1	1.032791	0	1.239733	0	0.425059	0
0.478357	0	1.707439	0	1.717609	0	0.212525	1	0.864563	0	1.117775	0	0.327108	0
0.817092	0	1.362117	0	1.339714	0	0.439256	1	1.212087	0	1.262273	0	0.479741	0
0.738927	0	1.811857	0	1.759363	0	0.103191	1	0.798201	0	1.426466	0	0.630449	0
0.380403	0	1.734746	0	1.764689	0	0.299335	0	0.859306	0	1.027470	0	0.264749	1
0.664631	0	1.695108	0	1.664018	0	0.106477	1	0.883381	0	1.303818	0	0.495853	0
0.518582	0	1.584698	0	1.605545	0	0.308379	0	0.991221	0	1.066675	0	0.251931	1

0.610689	0	1.748367	0	1.725163	0	0.063279	1	0.823661	0	1.277125	0	0.481123	0
0.633817	0	1.837029	0	1.804937	0	0.035851	1	0.742483	0	1.338238	0	0.558663	0
0.375528	0	1.741885	0	1.771924	0	0.299623	0	0.852879	0	1.027605	0	0.268770	1
0.488668	0	1.605376	0	1.630571	0	0.312894	0	0.975070	0	1.046606	0	0.234629	1
0.638794	0	1.845049	0	1.811755	0	0.044090	1	0.735843	0	1.346024	0	0.567691	0
0.874573	0	1.655197	0	1.580118	0	0.265324	1	0.993434	0	1.489887	0	0.674309	0
0.644135	0	1.934136	0	1.898875	0	0.133086	1	0.649856	0	1.382185	0	0.626816	0
0.639824	0	1.934304	0	1.899863	0	0.133131	1	0.648564	0	1.378296	0	0.623791	0
0.741060	0	1.810157	0	1.757274	0	0.105070	1	0.800583	0	1.427797	0	0.631318	0
0.745073	0	1.805048	0	1.751480	0	0.108505	1	0.806813	0	1.429561	0	0.631926	0
0.489917	0	1.604806	0	1.629753	0	0.312254	0	0.975382	0	1.047730	0	0.235636	1
0.678006	0	1.935179	0	1.893478	0	0.139503	1	0.658914	0	1.413442	0	0.652283	0
0.635297	0	1.934482	0	1.900902	0	0.133321	1	0.647247	0	1.374213	0	0.620624	0
0.636295	0	1.840400	0	1.807722	0	0.039257	1	0.739772	0	1.341881	0	0.562775	0
0.499108	0	1.596735	0	1.620747	0	0.312845	0	0.982247	0	1.052193	0	0.238868	1
0.723901	0	1.384994	0	1.389316	0	0.429673	0	1.181809	0	1.157731	0	0.373968	1
0.650466	0	1.858796	0	1.822894	0	0.059839	1	0.725382	0	1.362098	0	0.585467	0
0.868510	0	2.575684	0	2.551699	0	0.782347	0	0.032482	1	1.649701	0	1.115928	0
1.052187	0	2.798811	0	2.772919	0	1.004378	0	0.234136	1	1.811856	0	1.328124	0
0.943009	0	2.697866	0	2.678757	0	0.909081	0	0.131605	1	1.706988	0	1.218818	0
1.027028	0	2.861335	0	2.857558	0	1.091119	0	0.324547	1	1.748307	0	1.346541	0
0.835423	0	1.662832	0	1.886184	0	1.293055	0	1.676749	0	0.187683	1	0.759976	0
0.688861	0	1.696875	0	1.895958	0	1.164322	0	1.530030	0	0.192752	1	0.639481	0
0.789979	0	1.724409	0	1.940805	0	1.274714	0	1.628450	0	0.123929	1	0.748522	0
0.796874	0	1.593888	0	1.809089	0	1.223690	0	1.638973	0	0.255597	1	0.687254	0
1.332561	0	1.898927	0	2.185309	0	1.825437	0	2.158880	0	0.538111	1	1.292413	0
1.015510	0	1.510450	0	1.761334	0	1.405401	0	1.856756	0	0.389341	1	0.867243	0
0.825445	0	1.427233	0	1.638149	0	1.164286	0	1.655950	0	0.426187	1	0.628062	0

0.636843	0	1.650896	0	1.838783	0	1.087797	0	1.479049	0	0.273106	1	0.559565	0
1.193384	0	2.274100	0	2.523098	0	1.788080	0	1.940954	0	0.509621	1	1.306323	0
0.817362	0	2.205650	0	2.407526	0	1.435501	0	1.544487	0	0.365821	1	0.998383	0
0.769034	0	2.267750	0	2.457044	0	1.399107	0	1.460010	0	0.449704	1	0.992429	0
0.745743	0	1.285098	0	1.444907	0	0.932363	0	1.518877	0	0.662843	0	0.427044	1
0.704753	0	1.308844	0	1.457630	0	0.884005	0	1.471285	0	0.674029	0	0.378218	1
0.675568	0	1.345796	0	1.495546	0	0.881014	0	1.451788	0	0.642127	0	0.364346	1
1.248199	0	1.797901	0	2.077893	0	1.721845	0	2.081882	0	0.464460	1	1.186370	0
0.806179	0	2.091409	0	2.299405	0	1.403582	0	1.574441	0	0.246706	1	0.938639	0
0.374054	0	1.598983	0	1.700501	0	0.642254	0	1.139146	0	0.699391	0	0.117431	1
0.370143	0	1.602816	0	1.704081	0	0.640973	0	1.135752	0	0.699544	0	0.117914	1
0.192779	1	1.775968	0	1.864676	0	0.595854	0	0.978754	0	0.739809	0	0.221635	0
0.343394	0	1.824071	0	1.851454	0	0.296574	1	0.773460	0	1.050947	0	0.334043	0
0.292712	0	1.795900	0	1.837742	0	0.362858	0	0.826232	0	0.976069	0	0.271837	1
0.233021	0	1.735602	0	1.824903	0	0.591195	0	1.007247	0	0.737470	0	0.183413	1
0.266558	0	1.702258	0	1.793101	0	0.594536	0	1.034799	0	0.732074	0	0.155036	1
0.232369	0	1.736265	0	1.825592	0	0.591434	0	1.006899	0	0.737306	0	0.184103	1
0.403902	0	1.565691	0	1.650540	0	0.569537	0	1.115909	0	0.784435	0	0.031855	1
0.415762	0	1.567915	0	1.636855	0	0.500554	0	1.076801	0	0.855375	0	0.039854	1
0.438442	0	1.530519	0	1.626858	0	0.625217	0	1.172462	0	0.743690	0	0.087614	1
0.433219	0	1.536060	0	1.633283	0	0.628225	0	1.170828	0	0.737943	0	0.089933	1
0.162829	1	1.805624	0	1.890576	0	0.583787	0	0.945971	0	0.760178	0	0.245709	0
1.147763	0	2.852586	0	2.813852	0	1.052181	0	0.311297	1	1.914792	0	1.407290	0
0.870894	0	2.786357	0	2.814645	0	1.082169	0	0.426321	1	1.537612	0	1.233368	0
0.654568	0	1.973682	0	1.936824	0	0.172828	1	0.613173	0	1.403538	0	0.658158	0
1.029973	0	1.851897	0	1.741426	0	0.394625	1	0.936211	0	1.705881	0	0.896647	0
0.298476	1	1.896449	0	1.927704	0	0.341854	0	0.716222	0	1.046291	0	0.381476	0
0.359388	0	1.900181	0	1.919601	0	0.286131	1	0.690609	0	1.106007	0	0.415825	0

0.130527	1	2.017824	0	2.078956	0	0.552378	0	0.718215	0	0.926889	0	0.452338	0
0.386140	0	1.863835	0	1.879712	0	0.252448	1	0.719461	0	1.114949	0	0.398814	0
0.898282	0	2.486854	0	2.443168	0	0.685755	0	0.190857	1	1.693200	0	1.087247	0
1.132949	0	2.644018	0	2.573891	0	0.857905	0	0.364535	1	1.928256	0	1.303776	0

5. En la tabla 16 no hay variaciones en las distancias al comparar la tabla 14 y 16 por lo tanto se obtienen los centroides finales (tabla 15), se demuestra al realizar la validación en la tabla 17.

Centroides finales

Tabla 15 Centroides finales

Núm. de centroide	Abscisa (x)	Ordenada (y)
Centroide 1	2.007911	0.232855
Centroide 2	0.740752	2.739102
Centroide 3	0.430980	2.545116
Centroide 4	1.394533	1.060757
Centroide 5	1.822258	0.411294
Centroide 6	2.400218	1.925883
Centroide 7	1.703119	1.502130

Tabla 16 Octava iteración con centroides finales

Iteración8_Centro1		Iteración8_Centro2		Iteración8_Centro3		Iteración8_Centro4		Iteración8_Centro5		Iteración8_Centro6		Iteración8_Centro7	
2.007911	1.232855	0.740752	2.739102	0.430980	2.545116	1.394533	1.060757	1.822258	0.411294	2.400218	1.925883	1.703119	1.502130
0.198729	1	2.159101	0	2.230034	0	0.691024	0	0.690711	0	0.930825	0	0.592214	0
0.171359	1	2.108416	0	2.206279	0	0.791310	0	0.857787	0	0.766472	0	0.563639	0
0.187932	1	2.004030	0	2.118438	0	0.819259	0	0.983263	0	0.637892	0	0.494706	0
0.349350	1	2.153420	0	2.282945	0	0.986024	0	1.039500	0	0.620663	0	0.674845	0
0.376915	1	2.337245	0	2.428064	0	0.916051	0	0.768505	0	0.924351	0	0.783024	0

0.104625	1	2.072307	0	2.151845	0	0.676203	0	0.773372	0	0.849220	0	0.508407	0
0.190560	1	2.012961	0	2.127301	0	0.823532	0	0.979976	0	0.641150	0	0.502990	0
0.561701	0	2.051165	0	2.224538	0	1.172778	0	1.331988	0	0.337222	1	0.742782	0
0.562970	0	2.044797	0	2.218716	0	1.172451	0	1.335835	0	0.330684	1	0.740056	0
0.555023	0	1.915867	0	2.093353	0	1.127620	0	1.368931	0	0.254122	1	0.658371	0
0.459659	0	2.109716	0	2.113940	0	0.378912	1	0.471307	0	1.252607	0	0.630392	0
0.307773	1	2.134196	0	2.170067	0	0.515978	0	0.539549	0	1.101700	0	0.589522	0
0.536132	0	2.435190	0	2.469478	0	0.768146	0	0.382959	1	1.268113	0	0.882697	0
0.816603	0	2.505553	0	2.481851	0	0.712363	0	0.069983	1	1.603713	0	1.051103	0
0.689981	0	2.383506	0	2.369474	0	0.601540	0	0.184722	1	1.481868	0	0.917254	0
0.747910	0	2.417593	0	2.395912	0	0.626236	0	0.151463	1	1.540306	0	0.966694	0
0.708350	0	2.285596	0	2.256478	0	0.487784	0	0.293855	1	1.504291	0	0.871542	0
0.682740	0	2.294979	0	2.271840	0	0.502165	0	0.275505	1	1.479098	0	0.862604	0
0.184166	1	1.849674	0	1.908981	0	0.465146	0	0.828428	0	0.896098	0	0.288860	0
0.211317	0	1.768921	0	1.842547	0	0.517764	0	0.932543	0	0.816180	0	0.201682	1
0.170169	1	1.798263	0	1.883465	0	0.583173	0	0.951644	0	0.758491	0	0.238682	0
0.185898	1	1.782506	0	1.869533	0	0.588728	0	0.968459	0	0.748544	0	0.225487	0
0.432846	0	1.536424	0	1.633607	0	0.627998	0	1.170454	0	0.737996	0	0.089686	1
0.044619	1	1.925037	0	2.006986	0	0.610576	0	0.861074	0	0.792344	0	0.362288	0
0.084431	1	1.895175	0	1.971942	0	0.571228	0	0.857984	0	0.813664	0	0.329152	0
0.040325	1	2.003400	0	2.082513	0	0.634417	0	0.805263	0	0.827827	0	0.439003	0
0.373080	0	1.599831	0	1.701159	0	0.641344	0	1.137979	0	0.700045	0	0.116928	1
0.447780	0	1.549481	0	1.671325	0	0.738422	0	1.237411	0	0.623778	0	0.201455	1
0.545597	0	1.423334	0	1.513045	0	0.626184	0	1.238548	0	0.812685	0	0.148096	1
0.223569	1	1.761476	0	1.863810	0	0.662535	0	1.038980	0	0.669788	0	0.238888	0
1.929657	0	0.522658	0	0.218032	1	1.598949	0	2.375174	0	2.021761	0	1.526684	0
1.846682	0	0.185864	1	0.528665	0	1.733353	0	2.485472	0	1.678783	0	1.452666	0
1.897672	0	0.138207	1	0.494599	0	1.773497	0	2.528991	0	1.734684	0	1.502218	0

2.006382	0	0.040116	1	0.349231	0	1.833132	0	2.600145	0	1.887979	0	1.604759	0
2.088112	0	0.148558	1	0.279490	0	1.887582	0	2.659424	0	1.992638	0	1.684159	0
1.880092	0	0.146428	1	0.285962	0	1.679408	0	2.449908	0	1.815694	0	1.475353	0
1.753389	0	0.329359	0	0.309050	1	1.514268	0	2.288649	0	1.758980	0	1.346734	0
2.105623	0	0.141352	1	0.338105	0	1.921150	0	2.690746	0	1.987560	0	1.703247	0
2.141940	0	0.291890	0	0.160022	1	1.894762	0	2.671389	0	2.098750	0	1.735518	0
2.096673	0	0.284243	0	0.117697	1	1.844927	0	2.621705	0	2.065067	0	1.690103	0
2.050741	0	0.372099	0	0.007239	1	1.766799	0	2.544454	0	2.066312	0	1.644316	0
1.810618	0	0.388486	0	0.241831	1	1.534499	0	2.311727	0	1.852145	0	1.404347	0
2.214530	0	0.334136	0	0.219634	1	1.968127	0	2.744878	0	2.161593	0	1.808216	0
2.146401	0	0.293154	0	0.164374	1	1.899640	0	2.676254	0	2.102122	0	1.739995	0
2.084370	0	0.306091	0	0.084002	1	1.823919	0	2.601088	0	2.065476	0	1.677684	0
2.428611	0	0.623818	0	0.378685	1	2.123487	0	2.900014	0	2.429636	0	2.022406	0
1.813924	0	0.390011	0	0.238811	1	1.536699	0	2.313975	0	1.856277	0	1.407684	0
2.137578	0	0.623740	0	0.258343	1	1.783271	0	2.555720	0	2.233963	0	1.736207	0
2.111636	0	0.550192	0	0.184742	1	1.777119	0	2.552205	0	2.184996	0	1.708173	0
0.950062	0	1.627194	0	1.536720	0	0.342761	1	1.055808	0	1.549908	0	0.734378	0
0.691125	0	1.538348	0	1.518993	0	0.264143	1	1.032791	0	1.239733	0	0.425059	0
0.478357	0	1.707439	0	1.717609	0	0.212525	1	0.864563	0	1.117775	0	0.327108	0
0.817092	0	1.362117	0	1.339714	0	0.439256	1	1.212087	0	1.262273	0	0.479741	0
0.738927	0	1.811857	0	1.759363	0	0.103191	1	0.798201	0	1.426466	0	0.630449	0
0.380403	0	1.734746	0	1.764689	0	0.299335	0	0.859306	0	1.027470	0	0.264749	1
0.664631	0	1.695108	0	1.664018	0	0.106477	1	0.883381	0	1.303818	0	0.495853	0
0.518582	0	1.584698	0	1.605545	0	0.308379	0	0.991221	0	1.066675	0	0.251931	1
0.610689	0	1.748367	0	1.725163	0	0.063279	1	0.823661	0	1.277125	0	0.481123	0
0.633817	0	1.837029	0	1.804937	0	0.035851	1	0.742483	0	1.338238	0	0.558663	0
0.375528	0	1.741885	0	1.771924	0	0.299623	0	0.852879	0	1.027605	0	0.268770	1
0.488668	0	1.605376	0	1.630571	0	0.312894	0	0.975070	0	1.046606	0	0.234629	1

0.638794	0	1.845049	0	1.811755	0	0.044090	1	0.735843	0	1.346024	0	0.567691	0
0.874573	0	1.655197	0	1.580118	0	0.265324	1	0.993434	0	1.489887	0	0.674309	0
0.644135	0	1.934136	0	1.898875	0	0.133086	1	0.649856	0	1.382185	0	0.626816	0
0.639824	0	1.934304	0	1.899863	0	0.133131	1	0.648564	0	1.378296	0	0.623791	0
0.741060	0	1.810157	0	1.757274	0	0.105070	1	0.800583	0	1.427797	0	0.631318	0
0.745073	0	1.805048	0	1.751480	0	0.108505	1	0.806813	0	1.429561	0	0.631926	0
0.489917	0	1.604806	0	1.629753	0	0.312254	0	0.975382	0	1.047730	0	0.235636	1
0.678006	0	1.935179	0	1.893478	0	0.139503	1	0.658914	0	1.413442	0	0.652283	0
0.635297	0	1.934482	0	1.900902	0	0.133321	1	0.647247	0	1.374213	0	0.620624	0
0.636295	0	1.840400	0	1.807722	0	0.039257	1	0.739772	0	1.341881	0	0.562775	0
0.499108	0	1.596735	0	1.620747	0	0.312845	0	0.982247	0	1.052193	0	0.238868	1
0.723901	0	1.384994	0	1.389316	0	0.429673	0	1.181809	0	1.157731	0	0.373968	1
0.650466	0	1.858796	0	1.822894	0	0.059839	1	0.725382	0	1.362098	0	0.585467	0
0.868510	0	2.575684	0	2.551699	0	0.782347	0	0.032482	1	1.649701	0	1.115928	0
1.052187	0	2.798811	0	2.772919	0	1.004378	0	0.234136	1	1.811856	0	1.328124	0
0.943009	0	2.697866	0	2.678757	0	0.909081	0	0.131605	1	1.706988	0	1.218818	0
1.027028	0	2.861335	0	2.857558	0	1.091119	0	0.324547	1	1.748307	0	1.346541	0
0.835423	0	1.662832	0	1.886184	0	1.293055	0	1.676749	0	0.187683	1	0.759976	0
0.688861	0	1.696875	0	1.895958	0	1.164322	0	1.530030	0	0.192752	1	0.639481	0
0.789979	0	1.724409	0	1.940805	0	1.274714	0	1.628450	0	0.123929	1	0.748522	0
0.796874	0	1.593888	0	1.809089	0	1.223690	0	1.638973	0	0.255597	1	0.687254	0
1.332561	0	1.898927	0	2.185309	0	1.825437	0	2.158880	0	0.538111	1	1.292413	0
1.015510	0	1.510450	0	1.761334	0	1.405401	0	1.856756	0	0.389341	1	0.867243	0
0.825445	0	1.427233	0	1.638149	0	1.164286	0	1.655950	0	0.426187	1	0.628062	0
0.636843	0	1.650896	0	1.838783	0	1.087797	0	1.479049	0	0.273106	1	0.559565	0
1.193384	0	2.274100	0	2.523098	0	1.788080	0	1.940954	0	0.509621	1	1.306323	0
0.817362	0	2.205650	0	2.407526	0	1.435501	0	1.544487	0	0.365821	1	0.998383	0
0.769034	0	2.267750	0	2.457044	0	1.399107	0	1.460010	0	0.449704	1	0.992429	0

0.745743	0	1.285098	0	1.444907	0	0.932363	0	1.518877	0	0.662843	0	0.427044	1
0.704753	0	1.308844	0	1.457630	0	0.884005	0	1.471285	0	0.674029	0	0.378218	1
0.675568	0	1.345796	0	1.495546	0	0.881014	0	1.451788	0	0.642127	0	0.364346	1
1.248199	0	1.797901	0	2.077893	0	1.721845	0	2.081882	0	0.464460	1	1.186370	0
0.806179	0	2.091409	0	2.299405	0	1.403582	0	1.574441	0	0.246706	1	0.938639	0
0.374054	0	1.598983	0	1.700501	0	0.642254	0	1.139146	0	0.699391	0	0.117431	1
0.370143	0	1.602816	0	1.704081	0	0.640973	0	1.135752	0	0.699544	0	0.117914	1
0.192779	1	1.775968	0	1.864676	0	0.595854	0	0.978754	0	0.739809	0	0.221635	0
0.343394	0	1.824071	0	1.851454	0	0.296574	1	0.773460	0	1.050947	0	0.334043	0
0.292712	0	1.795900	0	1.837742	0	0.362858	0	0.826232	0	0.976069	0	0.271837	1
0.233021	0	1.735602	0	1.824903	0	0.591195	0	1.007247	0	0.737470	0	0.183413	1
0.266558	0	1.702258	0	1.793101	0	0.594536	0	1.034799	0	0.732074	0	0.155036	1
0.232369	0	1.736265	0	1.825592	0	0.591434	0	1.006899	0	0.737306	0	0.184103	1
0.403902	0	1.565691	0	1.650540	0	0.569537	0	1.115909	0	0.784435	0	0.031855	1
0.415762	0	1.567915	0	1.636855	0	0.500554	0	1.076801	0	0.855375	0	0.039854	1
0.438442	0	1.530519	0	1.626858	0	0.625217	0	1.172462	0	0.743690	0	0.087614	1
0.433219	0	1.536060	0	1.633283	0	0.628225	0	1.170828	0	0.737943	0	0.089933	1
0.162829	1	1.805624	0	1.890576	0	0.583787	0	0.945971	0	0.760178	0	0.245709	0
1.147763	0	2.852586	0	2.813852	0	1.052181	0	0.311297	1	1.914792	0	1.407290	0
0.870894	0	2.786357	0	2.814645	0	1.082169	0	0.426321	1	1.537612	0	1.233368	0
0.654568	0	1.973682	0	1.936824	0	0.172828	1	0.613173	0	1.403538	0	0.658158	0
1.029973	0	1.851897	0	1.741426	0	0.394625	1	0.936211	0	1.705881	0	0.896647	0
0.298476	1	1.896449	0	1.927704	0	0.341854	0	0.716222	0	1.046291	0	0.381476	0
0.359388	0	1.900181	0	1.919601	0	0.286131	1	0.690609	0	1.106007	0	0.415825	0
0.130527	1	2.017824	0	2.078956	0	0.552378	0	0.718215	0	0.926889	0	0.452338	0
0.386140	0	1.863835	0	1.879712	0	0.252448	1	0.719461	0	1.114949	0	0.398814	0
0.898282	0	2.486854	0	2.443168	0	0.685755	0	0.190857	1	1.693200	0	1.087247	0
1.132949	0	2.644018	0	2.573891	0	0.857905	0	0.364535	1	1.928256	0	1.303776	0

En la tabla 17, se realizó la validación al comparar las columnas de la distancia de la tabla 14 y la tabla 16 de acuerdo con la siguiente prueba lógica, arrojando un uno si las distancias son iguales por lo tanto se comprueba que no hay variación entre las tablas:

$$\textit{Valido} = \textit{SI}(\textit{DatoTabla14} = \textit{DatoTabla15},1)$$

$$1 = \textit{SI}(1.198729 = 1.198729,1)$$

Tabla 17 Validación de distancias

d1	d2	Validación					d7
		d3	d4	d5	d6		
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	
1	1	1	1	1	1	1	

1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1

1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1

6. Se obtuvo la clasificación como se muestra en la tabla 18 primera columna, el procedimiento se realizó con base a la tabla 16; identificando los unos en la columna binaria del primer centroide y asignando a cada punto una A, el mismo procedimiento se emplea para el segundo centroide asignando a cada punto una B y así sucesivamente hasta llegar al séptimo centroide asignando una G a cada punto, a través de la siguiente prueba binaria:

Clasificación = SI(PrimerColumnaDato1a117 = 1, "A", SI(SegundaColumnaDato1a117 = 1, "B", SI(TercerColumnaDato1a117 = 1, "C", SI(CuartaColumnaDato1a117 = 1, "D", SI(QuintaColumnaDato1a117 = 1, "E", SI(SextaColumnaDato1a117 = 1, "F", SI(SéptimaColumnaDato1a117 = 1, "G"))))))))

Clasificación = SI(\$DE\$3:\$DE\$119 = 1, "A", SI(\$DH\$3:\$DH\$119 = 1, "B", SI(\$DK\$3:\$DK\$119 = 1, "C", SI(\$DN\$3:\$DN\$119 = 1, "D", SI(\$DQ\$3:\$DQ\$119 = 1, "E", SI(\$DT\$3:\$DT\$119 = 1, "F", SI(\$DW\$3:\$DW\$119 = 1, "G"))))))))

Se obtuvieron las coordenadas en la tabla 18, de los puntos que pertenecen a cada centroide a través de las pruebas lógicas. Posteriormente estos datos se utilizan para ser observados gráficamente.

Pruebas lógicas, utilizando la columna binaria del primer centroide tabla 16.

Abscisa = SI(Dato1a117 = 1, abscisa, 0)

Abscisa = SI(\$DE\$3:\$DE\$170 = 1, G3, 0)

Ordenada = SI(Dato1a117 = 1, ordenada, 0)

Ordenada = SI(\$DE\$3:\$DE\$170 = 1, H3, 0)

Lo mismo se realiza para obtener la abscisa y ordenada de los demás centroides, utilizando la columna correspondiente a cada centroide.

Tabla 18 Clasificación de los parámetros

Clasificación en 7 grupos	Grupo1 A		Grupo2 B		Grupo3 C		Grupo4 D		Grupo5 E		Grupo6 F		Grupo7 G	
	Abscisa (x)	Ordenada (y)	Abscisa (x)	Ordenada (y)	Abscisa (x)	Ordenada (y)	Abscisa (x)	Ordenada (y)	Abscisa (x)	Ordenada (y)	Abscisa (x)	Ordenada (y)	Abscisa (x)	Ordenada (y)
A	2.085472	1.049887	0	0	0	0	0	0	0	0	0	0	0	0
A	2.174655	1.193352	0	0	0	0	0	0	0	0	0	0	0	0
A	2.167619	1.331910	0	0	0	0	0	0	0	0	0	0	0	0
A	2.349233	1.307318	0	0	0	0	0	0	0	0	0	0	0	0
A	2.308949	1.006049	0	0	0	0	0	0	0	0	0	0	0	0
A	2.065411	1.145448	0	0	0	0	0	0	0	0	0	0	0	0
A	2.174220	1.325885	0	0	0	0	0	0	0	0	0	0	0	0
F	0	0	0	0	0	0	0	0	0	0	2.440555	1.591083	0	0
F	0	0	0	0	0	0	0	0	0	0	2.437036	1.597255	0	0
F	0	0	0	0	0	0	0	0	0	0	2.337084	1.679729	0	0
D	0	0	0	0	0	0	1.723114	0.872055	0	0	0	0	0	0
A	1.897491	0.945573	0	0	0	0	0	0	0	0	0	0	0	0
E	0	0	0	0	0	0	0	0	2.073064	0.700697	0	0	0	0
E	0	0	0	0	0	0	0	0	1.765992	0.452909	0	0	0	0
E	0	0	0	0	0	0	0	0	1.765185	0.586978	0	0	0	0
E	0	0	0	0	0	0	0	0	1.736304	0.536006	0	0	0	0
E	0	0	0	0	0	0	0	0	1.630140	0.633648	0	0	0	0
E	0	0	0	0	0	0	0	0	1.668947	0.640202	0	0	0	0
A	1.823873	1.239721	0	0	0	0	0	0	0	0	0	0	0	0
G	0	0	0	0	0	0	0	0	0	0	0	0	1.828072	1.343819
A	1.894985	1.360155	0	0	0	0	0	0	0	0	0	0	0	0
A	1.890911	1.377316	0	0	0	0	0	0	0	0	0	0	0	0
G	0	0	0	0	0	0	0	0	0	0	0	0	1.748599	1.579428
A	1.971942	1.259258	0	0	0	0	0	0	0	0	0	0	0	0
A	1.928903	1.262625	0	0	0	0	0	0	0	0	0	0	0	0

A	2.014981	1.193155	0	0	0	0	0	0	0	0	0	0	0	0
G	0	0	0	0	0	0	0	0	0	0	0	0	1.810150	1.549209
G	0	0	0	0	0	0	0	0	0	0	0	0	1.841479	1.648556
G	0	0	0	0	0	0	0	0	0	0	0	0	1.640808	1.636479
A	1.934865	1.444154	0	0	0	0	0	0	0	0	0	0	0	0
C	0	0	0	0	0.418733	2.327428	0	0	0	0	0	0	0	0
B	0	0	0.926378	2.729692	0	0	0	0	0	0	0	0	0	0
B	0	0	0.877757	2.757293	0	0	0	0	0	0	0	0	0	0
B	0	0	0.706406	2.759831	0	0	0	0	0	0	0	0	0	0
B	0	0	0.595647	2.770947	0	0	0	0	0	0	0	0	0	0
B	0	0	0.712481	2.595429	0	0	0	0	0	0	0	0	0	0
C	0	0	0	0	0.709522	2.411227	0	0	0	0	0	0	0	0
B	0	0	0.625842	2.821419	0	0	0	0	0	0	0	0	0	0
C	0	0	0	0	0.450995	2.703882	0	0	0	0	0	0	0	0
C	0	0	0	0	0.468748	2.656589	0	0	0	0	0	0	0	0
C	0	0	0	0	0.426987	2.539078	0	0	0	0	0	0	0	0
C	0	0	0	0	0.603515	2.375664	0	0	0	0	0	0	0	0
C	0	0	0	0	0.407507	2.763492	0	0	0	0	0	0	0	0
C	0	0	0	0	0.449201	2.708477	0	0	0	0	0	0	0	0
C	0	0	0	0	0.456688	2.625087	0	0	0	0	0	0	0	0
C	0	0	0	0	0.117193	2.757106	0	0	0	0	0	0	0	0
C	0	0	0	0	0.599256	2.375664	0	0	0	0	0	0	0	0
C	0	0	0	0	0.217170	2.400112	0	0	0	0	0	0	0	0
C	0	0	0	0	0.277223	2.442702	0	0	0	0	0	0	0	0
D	0	0	0	0	0	0	1.062019	1.143938	0	0	0	0	0	0
D	0	0	0	0	0	0	1.321642	1.314644	0	0	0	0	0	0
D	0	0	0	0	0	0	1.529621	1.224825	0	0	0	0	0	0
D	0	0	0	0	0	0	1.224750	1.465874	0	0	0	0	0	0
D	0	0	0	0	0	0	1.301463	1.016189	0	0	0	0	0	0

G	0	0	0	0	0	0	0	0	0	0	0	0	0	1.627822	1.248314
D	0	0	0	0	0	0	1.347695	1.156379	0	0	0	0	0	0	0
G	0	0	0	0	0	0	0	0	0	0	0	0	0	1.502634	1.349569
D	0	0	0	0	0	0	1.407230	1.122750	0	0	0	0	0	0	0
D	0	0	0	0	0	0	1.408239	1.027630	0	0	0	0	0	0	0
G	0	0	0	0	0	0	0	0	0	0	0	0	0	1.632514	1.242799
G	0	0	0	0	0	0	0	0	0	0	0	0	0	1.531596	1.342034
D	0	0	0	0	0	0	1.406244	1.018251	0	0	0	0	0	0	0
D	0	0	0	0	0	0	1.139108	1.132556	0	0	0	0	0	0	0
D	0	0	0	0	0	0	1.436977	0.934620	0	0	0	0	0	0	0
D	0	0	0	0	0	0	1.441101	0.936036	0	0	0	0	0	0	0
D	0	0	0	0	0	0	1.298937	1.017156	0	0	0	0	0	0	0
D	0	0	0	0	0	0	1.293647	1.020816	0	0	0	0	0	0	0
G	0	0	0	0	0	0	0	0	0	0	0	0	0	1.530296	1.341954
D	0	0	0	0	0	0	1.405526	0.921688	0	0	0	0	0	0	0
D	0	0	0	0	0	0	1.445427	0.937533	0	0	0	0	0	0	0
D	0	0	0	0	0	0	1.407028	1.023542	0	0	0	0	0	0	0
G	0	0	0	0	0	0	0	0	0	0	0	0	0	1.521912	1.346496
G	0	0	0	0	0	0	0	0	0	0	0	0	0	1.329522	1.485483
D	0	0	0	0	0	0	1.400102	1.001178	0	0	0	0	0	0	0
E	0	0	0	0	0	0	0	0	1.797554	0.390205	0	0	0	0	0
E	0	0	0	0	0	0	0	0	1.890297	0.187262	0	0	0	0	0
E	0	0	0	0	0	0	0	0	1.888304	0.297462	0	0	0	0	0
E	0	0	0	0	0	0	0	0	2.075277	0.208039	0	0	0	0	0
F	0	0	0	0	0	0	0	0	0	0	2.246452	2.033498	0	0	0
F	0	0	0	0	0	0	0	0	0	0	2.210609	1.891219	0	0	0
F	0	0	0	0	0	0	0	0	0	0	2.285389	1.972499	0	0	0
F	0	0	0	0	0	0	0	0	0	0	2.160643	2.014955	0	0	0
F	0	0	0	0	0	0	0	0	0	0	2.612719	2.420258	0	0	0

F	0	0	0	0	0	0	0	0	0	0	2.165003	2.236142	0	0
F	0	0	0	0	0	0	0	0	0	0	1.995091	2.058200	0	0
F	0	0	0	0	0	0	0	0	0	0	2.136025	1.856679	0	0
F	0	0	0	0	0	0	0	0	0	0	2.900000	2.025536	0	0
F	0	0	0	0	0	0	0	0	0	0	2.682943	1.693738	0	0
F	0	0	0	0	0	0	0	0	0	0	2.692178	1.583842	0	0
G	0	0	0	0	0	0	0	0	0	0	0	0	1.737378	1.927797
G	0	0	0	0	0	0	0	0	0	0	0	0	1.727783	1.879543
G	0	0	0	0	0	0	0	0	0	0	0	0	1.761296	1.861801
F	0	0	0	0	0	0	0	0	0	0	2.502222	2.379004	0	0
F	0	0	0	0	0	0	0	0	0	0	2.599532	1.780495	0	0
G	0	0	0	0	0	0	0	0	0	0	0	0	1.810181	1.550376
G	0	0	0	0	0	0	0	0	0	0	0	0	1.812161	1.547002
A	1.892810	1.387502	0	0	0	0	0	0	0	0	0	0	0	0
D	0	0	0	0	0	0	1.670375	1.169695	0	0	0	0	0	0
G	0	0	0	0	0	0	0	0	0	0	0	0	1.715208	1.230562
G	0	0	0	0	0	0	0	0	0	0	0	0	1.865890	1.417596
G	0	0	0	0	0	0	0	0	0	0	0	0	1.847554	1.445784
G	0	0	0	0	0	0	0	0	0	0	0	0	1.866473	1.417221
G	0	0	0	0	0	0	0	0	0	0	0	0	1.727066	1.523136
G	0	0	0	0	0	0	0	0	0	0	0	0	1.671765	1.477527
G	0	0	0	0	0	0	0	0	0	0	0	0	1.741352	1.580962
G	0	0	0	0	0	0	0	0	0	0	0	0	1.748465	1.579794
A	1.899253	1.354127	0	0	0	0	0	0	0	0	0	0	0	0
E	0	0	0	0	0	0	0	0	1.823521	0.100000	0	0	0	0
E	0	0	0	0	0	0	0	0	2.248297	0.395794	0	0	0	0
D	0	0	0	0	0	0	1.446683	0.895984	0	0	0	0	0	0
D	0	0	0	0	0	0	1.029876	0.909913	0	0	0	0	0	0
A	1.730917	1.121668	0	0	0	0	0	0	0	0	0	0	0	0

D	0	0	0	0	0	0	1.679461	1.086978	0	0	0	0	0	0
A	1.943816	1.119148	0	0	0	0	0	0	0	0	0	0	0	0
D	0	0	0	0	0	0	1.642532	1.107945	0	0	0	0	0	0
E	0	0	0	0	0	0	0	0	1.631494	0.417244	0	0	0	0
E	0	0	0	0	0	0	0	0	1.517238	0.211673	0	0	0	0

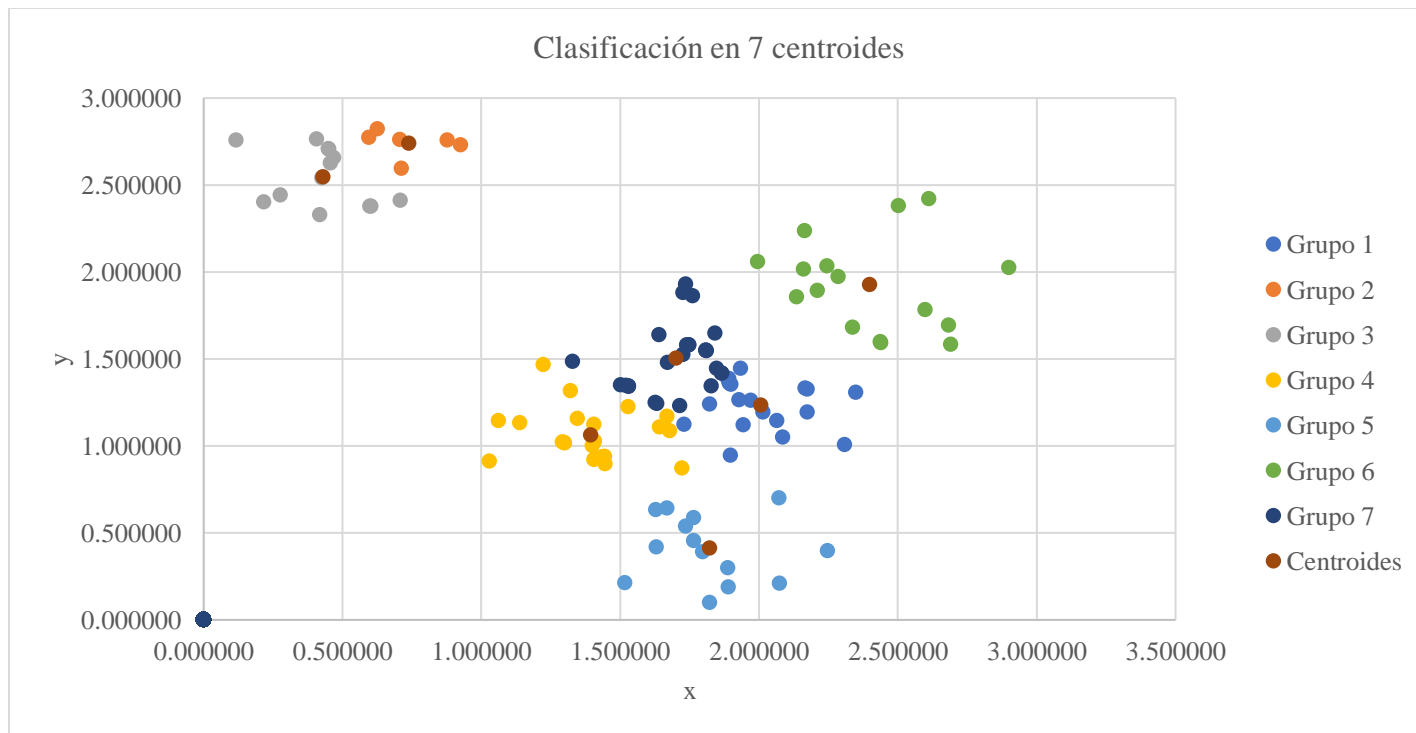


Figura 28 Clasificación red neuronal de base radial

5.4 ENTRENAMIENTO DE LA CAPA DE SALIDA DE LA RED NEURONAL DE BASE RADIAL

El entrenamiento de los elementos de procesamiento de la capa de salida se realizó de forma supervisada en donde los patrones se van propagando hasta generar una salida, esta se compara con los valores esperados y se calcula un error, este error es propagado hacia atrás para ajustar los pesos este proceso se repite hasta que se encuentra un resultado deseado. Se puede observar en la figura 29.

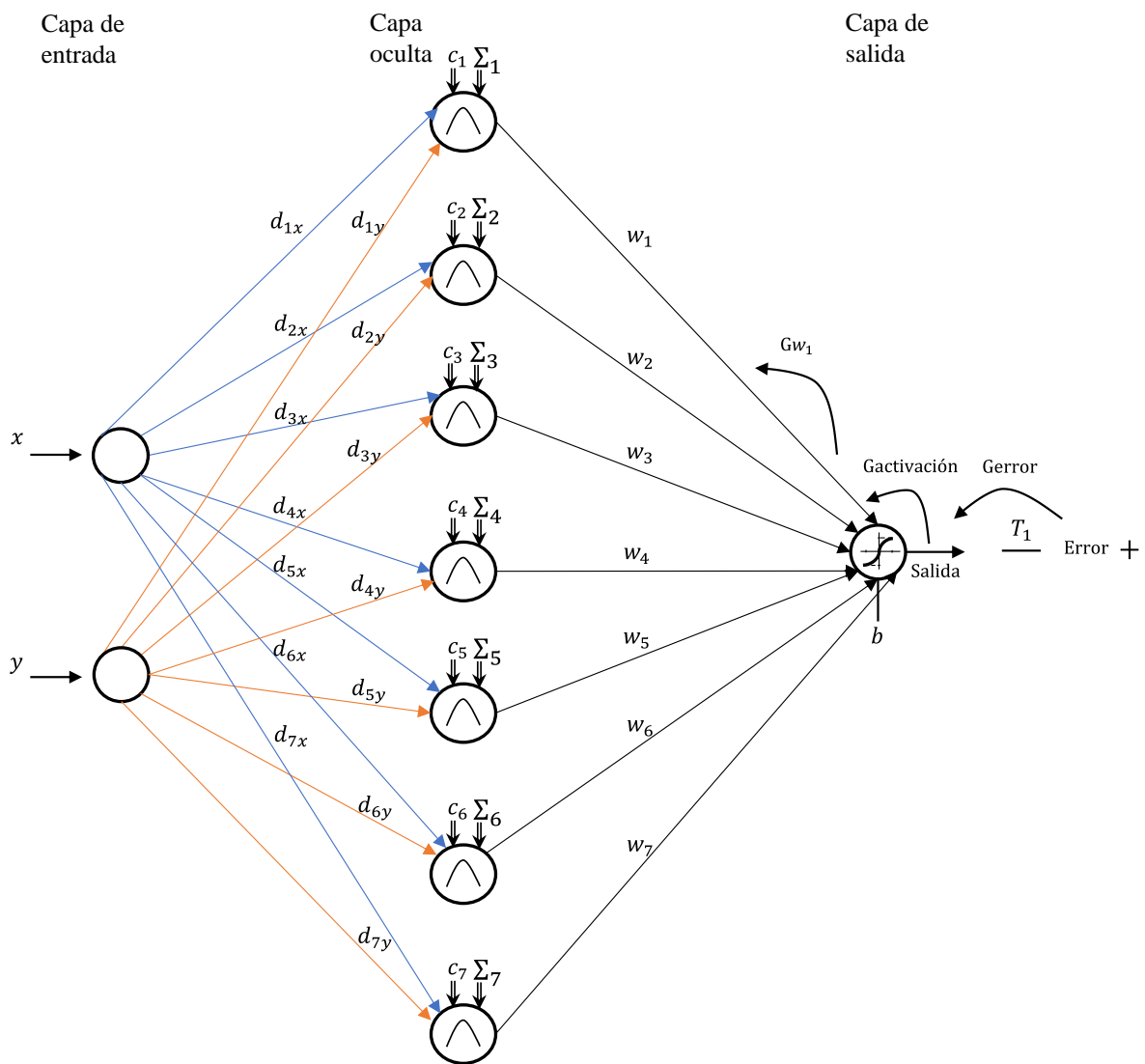


Figura 29 Diagrama de la red de base radial con siete centros

En la figura 29, la función de activación ϕ es una función del tipo base radial, la función de Gauss. Por lo tanto, los parámetros que caracterizan a esta capa oculta son el centroide de la función de activación.

Función de Gauss

$$\phi(x) = e^{-r^2}$$

$$r = \text{distancia euclidiana del nodo al centroide} = d_E(P_1, C_1) \\ = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Por cada centroide se introdujo una neurona por lo tanto en la tabla 19 se observan las siete columnas con el nombre de Neurona1, Neurona2 hasta Neurona7 de los primeros 117 datos de las coordenadas x e y.

Por ejemplo, para el primer dato de la neurona 1

Coordenadas (x, y) del centroide 1 = (2.007911,1.232855)

Coordenadas (x, y) del nodo 1 = (2.085472,1.049887)

$$\text{Dato 1} = \text{EXP}(-\text{POTENCIA}(\text{RAIZ}((2.007911 - X1) \wedge 2 + (1.232855 - Y1) \wedge 2), 2))$$

$$\text{Dato 1} = \text{EXP}(-\text{POTENCIA}(\text{RAIZ}((2.007911 - 2.085472) \wedge 2 + (1.232855 - 1.049887) \wedge 2), 2))$$

$$\text{Dato 1} = 0.961277$$

El procedimiento se repite para obtener los demás datos de acuerdo con el centroide indicado.

Tabla 19 Neuronas de la red de base radial

Núm. De Nodo	x	y	Neurona1	Neurona2	Neurona3	Neurona4	Neurona5	Neurona6	Neurona7
1	2.085472	1.049887	0.961277	0.009450	0.006922	0.620324	0.620592	0.420448	0.704183
2	2.174655	1.193352	0.971063	0.011732	0.007691	0.534635	0.479123	0.555726	0.727829
3	2.167619	1.331910	0.965298	0.018022	0.011246	0.511103	0.380295	0.665706	0.782912
4	2.349233	1.307318	0.885108	0.009685	0.005452	0.378233	0.339405	0.680299	0.634185
5	2.308949	1.006049	0.867565	0.004242	0.002752	0.432078	0.553994	0.425528	0.541655
6	2.065411	1.145448	0.989113	0.013644	0.009751	0.633022	0.549853	0.486181	0.772226
7	2.174220	1.325885	0.964338	0.017387	0.010830	0.507528	0.382758	0.662939	0.776469
8	2.440555	1.591083	0.729418	0.014887	0.007094	0.252737	0.169620	0.892509	0.575956
9	2.437036	1.597255	0.728378	0.015280	0.007279	0.252930	0.167889	0.896415	0.578288
10	2.337084	1.679729	0.734878	0.025463	0.012499	0.280403	0.153513	0.937463	0.648267
11	1.723114	0.872055	0.809542	0.011668	0.011462	0.866257	0.800811	0.208248	0.672069
12	1.897491	0.945573	0.909624	0.010517	0.009012	0.766260	0.747431	0.297083	0.706427
13	2.073064	0.700697	0.750184	0.002658	0.002247	0.554301	0.863590	0.200266	0.458794

14	1.765992	0.452909	0.513328	0.001878	0.002113	0.602022	0.995114	0.076391	0.331272
15	1.765185	0.586978	0.621218	0.003410	0.003645	0.696387	0.966454	0.111255	0.431126
16	1.736304	0.536006	0.571569	0.002895	0.003214	0.675588	0.977320	0.093243	0.392783
17	1.630140	0.633648	0.605464	0.005386	0.006148	0.788255	0.917273	0.104049	0.467860
18	1.668947	0.640202	0.627423	0.005159	0.005734	0.777113	0.926906	0.112171	0.475168
19	1.823873	1.239721	0.966652	0.032670	0.026142	0.805445	0.503439	0.447987	0.919946
20	1.828072	1.343819	0.956327	0.043758	0.033541	0.764847	0.419104	0.513683	0.960141
21	1.894985	1.360155	0.971458	0.039410	0.028798	0.711706	0.404289	0.562531	0.944623
22	1.890911	1.377316	0.966032	0.041697	0.030344	0.707087	0.391444	0.571027	0.950427
23	1.748599	1.579428	0.829149	0.094364	0.069344	0.674096	0.254117	0.580052	0.991989
24	1.971942	1.259258	0.998011	0.024581	0.017810	0.688801	0.476424	0.533760	0.876996
25	1.928903	1.262625	0.992897	0.027552	0.020475	0.721588	0.478961	0.515793	0.897322
26	2.014981	1.193155	0.998375	0.018068	0.013078	0.668656	0.522857	0.503940	0.824710
27	1.810150	1.549209	0.870064	0.077347	0.055358	0.662773	0.273899	0.612588	0.986421
28	1.841479	1.648556	0.818316	0.090637	0.061217	0.579687	0.216279	0.677667	0.960228
29	1.640808	1.636479	0.742542	0.131878	0.101337	0.675632	0.215671	0.516615	0.978306
30	1.934865	1.444154	0.951246	0.044923	0.030999	0.644712	0.339772	0.638512	0.944530
31	0.418733	2.327428	0.024147	0.760962	0.953574	0.077565	0.003548	0.016781	0.097221
32	0.926378	2.729692	0.033033	0.966044	0.756171	0.049563	0.002075	0.059707	0.121209
33	0.877757	2.757293	0.027292	0.981080	0.782995	0.043054	0.001669	0.049335	0.104700
34	0.706406	2.759831	0.017853	0.998392	0.885182	0.034722	0.001158	0.028312	0.076135
35	0.595647	2.770947	0.012776	0.978172	0.924858	0.028355	0.000848	0.018862	0.058637
36	0.712481	2.595429	0.029166	0.978787	0.921480	0.059581	0.002474	0.037003	0.113419
37	0.709522	2.411227	0.046219	0.897199	0.908908	0.100962	0.005311	0.045320	0.163051
38	0.625842	2.821419	0.011871	0.980218	0.891977	0.024952	0.000717	0.019247	0.054965
39	0.450995	2.703882	0.010174	0.918329	0.974718	0.027595	0.000796	0.012219	0.049192
40	0.468748	2.656589	0.012326	0.922383	0.986243	0.033248	0.001035	0.014059	0.057473
41	0.426987	2.539078	0.014913	0.870700	0.999948	0.044088	0.001543	0.013987	0.066952
42	0.603515	2.375664	0.037691	0.859916	0.943195	0.094923	0.004776	0.032373	0.139152
43	0.407507	2.763492	0.007416	0.894360	0.952906	0.020785	0.000534	0.009349	0.038020
44	0.449201	2.708477	0.009981	0.917650	0.973343	0.027089	0.000775	0.012047	0.048433
45	0.456688	2.625087	0.012977	0.910564	0.992969	0.035912	0.001153	0.014035	0.059927
46	0.117193	2.757106	0.002745	0.677633	0.866406	0.011007	0.000223	0.002731	0.016737
47	0.599256	2.375664	0.037242	0.858895	0.944565	0.094284	0.004727	0.031880	0.137852
48	0.217170	2.400112	0.010366	0.677699	0.935438	0.041583	0.001457	0.006802	0.049074
49	0.277223	2.442702	0.011574	0.738812	0.966446	0.042504	0.001483	0.008445	0.054050
50	1.062019	1.143938	0.405507	0.070810	0.094278	0.889154	0.328003	0.090517	0.583150
51	1.321642	1.314644	0.620237	0.093807	0.099526	0.932607	0.344157	0.215039	0.834706
52	1.529621	1.224825	0.795467	0.054185	0.052330	0.955838	0.473564	0.286671	0.898526
53	1.224750	1.465874	0.512918	0.156396	0.166155	0.824526	0.230120	0.203247	0.794413
54	1.301463	1.016189	0.579255	0.037522	0.045259	0.989408	0.528811	0.130706	0.672021

55	1.627822	1.248314	0.865276	0.049324	0.044417	0.914296	0.477875	0.347951	0.932308
56	1.347695	1.156379	0.642920	0.056507	0.062727	0.988727	0.458240	0.182694	0.782024
57	1.502634	1.349569	0.764199	0.081165	0.075943	0.909284	0.374367	0.320525	0.938503
58	1.407230	1.122750	0.688706	0.047039	0.050987	0.996004	0.507419	0.195724	0.793359
59	1.408239	1.027630	0.669165	0.034229	0.038473	0.998716	0.576211	0.166813	0.731905
60	1.632514	1.242799	0.868471	0.048115	0.043295	0.914138	0.483162	0.347854	0.930310
61	1.531596	1.342034	0.787576	0.075984	0.070035	0.906737	0.386447	0.334411	0.946437
62	1.406244	1.018251	0.664940	0.033233	0.037536	0.998058	0.581895	0.163363	0.724500
63	1.139108	1.132556	0.465391	0.064591	0.082350	0.932024	0.372726	0.108635	0.634643
64	1.436977	0.934620	0.660400	0.023733	0.027168	0.982444	0.655529	0.148016	0.675097
65	1.441101	0.936036	0.664065	0.023718	0.027066	0.982432	0.656630	0.149613	0.677657
66	1.298937	1.017156	0.577429	0.037754	0.045593	0.989021	0.526800	0.130210	0.671284
67	1.293647	1.020816	0.573995	0.038458	0.046529	0.988296	0.521551	0.129556	0.670769
68	1.530296	1.341954	0.786613	0.076123	0.070222	0.907100	0.386212	0.333625	0.945989
69	1.405526	0.921688	0.631478	0.023638	0.027730	0.980727	0.647804	0.135631	0.653460
70	1.445427	0.937533	0.667909	0.023701	0.026959	0.982383	0.657751	0.151305	0.680332
71	1.407028	1.023542	0.667062	0.033808	0.038088	0.998460	0.578531	0.165192	0.728538
72	1.521912	1.346496	0.779495	0.078116	0.072308	0.906765	0.381056	0.330513	0.944539
73	1.329522	1.485483	0.592128	0.146870	0.145120	0.831421	0.247419	0.261756	0.869487
74	1.400102	1.001178	0.655009	0.031583	0.036047	0.996426	0.590858	0.156404	0.709800
75	1.797554	0.390205	0.470335	0.001315	0.001487	0.542229	0.998945	0.065775	0.287856
76	1.890297	0.187262	0.330517	0.000396	0.000458	0.364665	0.946656	0.037522	0.171373
77	1.888304	0.297462	0.410958	0.000690	0.000765	0.437609	0.982829	0.054269	0.226385
78	2.075277	0.208039	0.348267	0.000278	0.000284	0.304057	0.900027	0.047049	0.163136
79	2.246452	2.033498	0.497614	0.062975	0.028505	0.187873	0.060116	0.965388	0.561264
80	2.210609	1.891219	0.622178	0.056169	0.027470	0.257781	0.096232	0.963529	0.664357
81	2.285389	1.972499	0.535761	0.051120	0.023128	0.196932	0.070521	0.984759	0.571046
82	2.160643	2.014955	0.529931	0.078829	0.037900	0.223707	0.068137	0.936758	0.623555
83	2.612719	2.420258	0.169362	0.027162	0.008433	0.035714	0.009459	0.748590	0.188185
84	2.165003	2.236142	0.356557	0.102135	0.044946	0.138740	0.031824	0.859344	0.471371
85	1.995091	2.058200	0.505929	0.130420	0.068321	0.257802	0.064430	0.833906	0.674042
86	2.136025	1.856679	0.666597	0.065516	0.034009	0.306266	0.112187	0.928127	0.731167
87	2.900000	2.025536	0.240709	0.005676	0.001719	0.040875	0.023114	0.771273	0.181504
88	2.682943	1.693738	0.512692	0.007713	0.003039	0.127369	0.092049	0.874743	0.369070
89	2.692178	1.583842	0.553544	0.005842	0.002389	0.141211	0.118644	0.816904	0.373471
90	1.737378	1.927797	0.573422	0.191766	0.123965	0.419245	0.099561	0.644448	0.833296
91	1.727783	1.879543	0.608550	0.180311	0.119470	0.457735	0.114787	0.634883	0.866712
92	1.761296	1.861801	0.633565	0.163463	0.106815	0.460158	0.121519	0.662107	0.875686
93	2.502222	2.379004	0.210557	0.039461	0.013331	0.051573	0.013112	0.805958	0.244761
94	2.599532	1.780495	0.522085	0.012601	0.005056	0.139451	0.083838	0.940951	0.414351
95	1.810181	1.550376	0.869431	0.077557	0.055482	0.662000	0.273172	0.613148	0.986305

96	1.812161	1.547002	0.871965	0.076611	0.054809	0.663089	0.275289	0.613017	0.986193
97	1.892810	1.387502	0.963518	0.042678	0.030899	0.701144	0.383675	0.578500	0.952065
98	1.670375	1.169695	0.888768	0.035892	0.032456	0.915801	0.549778	0.331380	0.894416
99	1.715208	1.230562	0.917888	0.039746	0.034140	0.876634	0.505272	0.385694	0.928769
100	1.865890	1.417596	0.947149	0.049178	0.035783	0.705033	0.362567	0.580502	0.966919
101	1.847554	1.445784	0.931412	0.055151	0.040147	0.702245	0.342731	0.585124	0.976250
102	1.866473	1.417221	0.947436	0.049065	0.035694	0.704833	0.362821	0.580643	0.966674
103	1.727066	1.523136	0.849475	0.086174	0.065593	0.722981	0.287868	0.540458	0.998986
104	1.671765	1.477527	0.841257	0.085575	0.068612	0.778369	0.313643	0.481106	0.998413
105	1.741352	1.580962	0.825116	0.096088	0.070887	0.676451	0.252924	0.575179	0.992353
106	1.748465	1.579794	0.828881	0.094469	0.069418	0.673905	0.253894	0.580097	0.991945
107	1.899253	1.354127	0.973835	0.038378	0.028036	0.711196	0.408664	0.561092	0.941413
108	1.823521	0.100000	0.267841	0.000292	0.000364	0.330521	0.907642	0.025568	0.138005
109	2.248297	0.395794	0.468389	0.000425	0.000363	0.310029	0.833810	0.094020	0.218450
110	1.446683	0.895984	0.651512	0.020335	0.023488	0.970572	0.686615	0.139468	0.648449
111	1.029876	0.909913	0.346164	0.032402	0.048192	0.855791	0.416241	0.054474	0.447546
112	1.730917	1.121668	0.914765	0.027419	0.024330	0.889706	0.598713	0.334631	0.864569
113	1.679461	1.086978	0.878834	0.027033	0.025100	0.921391	0.620680	0.294272	0.841213
114	1.943816	1.119148	0.983107	0.017050	0.013273	0.737034	0.597003	0.423533	0.814965
115	1.642532	1.107945	0.861479	0.030997	0.029208	0.938258	0.595935	0.288485	0.852952
116	1.631494	0.417244	0.446235	0.002061	0.002557	0.624840	0.964229	0.056873	0.306632
117	1.517238	0.211673	0.277046	0.000920	0.001327	0.479026	0.875565	0.024278	0.182714

La capa de salida está conectada a todas las neuronas de entrada, se le implementa un bias para que la superficie de separación no se quede parada en el origen del espacio n-dimensional. La función de activación que utiliza la neurona de salida es la sigmoideal.

La única neurona de salida realiza la suma ponderada de las entradas, se suma el bias y se pasa el resultado a una función de activación.

El siguiente procedimiento se realizó en la tabla 20 en Microsoft Excel

1. Se asignan los valores iniciales de forma aleatoria de -3 a 3 a los pesos w_{ji} , y al bias b_j .
2. Se presenta el vector de entrada x_n que es encontrado en la tabla 19 y se especifica el vector de salida T_{nj} deseado que es la clasificación obtenida en la tabla 18 en la primera columna.
3. Se calcula el valor de la función de agregación.

$$\text{Función de agregación}_{nj} = \sum_{i=1}^N w_{ji}x_{ni} + \theta_j$$

Función de agregación con siete centroides

$$\begin{aligned} \text{Agregación} = & (\text{Neurona1} * w1) + (\text{Neurona2} * w2) + (\text{Neurona3} * w3) \\ & + (\text{Neurona4} * w4) + (\text{Neurona5} * w5) + (\text{Neurona6} * w6) \\ & + (\text{Neurona7} * w7) + b \end{aligned}$$

4. Se calcula la salida de la red.

$$y_{nj} = f_j(\text{Función de agregación}_{nj})$$

$$y_{nj} = \frac{1}{1 + e^{-(\text{Función de agregación})}}$$

$$\text{Activación} = 1/(1 + \text{EXP}(-\text{Agregación}))$$

5. Se escribe el T como el valor deseado, datos obtenidos en la tabla 18.

$$\begin{aligned} T = & SI(\text{Clasificación} = "A", 1/7, SI(\text{Clasificación} = "B", 2/7, SI(\text{Clasificación} \\ & = "C", 3/7, SI(\text{Clasificación} = "D", 4/7, SI(\text{Clasificación} \\ & = "E", 5/7, SI(\text{Clasificación} = "F", 6/7, SI(\text{Clasificación} = "G", 1) \end{aligned}$$

6. Se calcula el gradiente de error

$$\text{Grad error} = -(T - \text{Activación})$$

Recordando que esta expresión surge de:

$$f = \frac{(\text{deseado} - \text{obtenido})^2}{2} \quad \text{utilizando } f = \frac{x^2}{2} \quad f' = \frac{2x}{2}$$

$$f' = \frac{1}{2}(2)(T - O)(-1) = -(T - O)$$

7. Se calcula el gradiente de activación

$$\text{Grad activación} = \text{Activación} * (1 - \text{Activación})$$

Recordando que esta expresión surge de:

$$\text{Función logística } y = \frac{1}{1 + e^{-z}}$$

$$\text{Usando la derivada } f = \frac{1}{x} \quad f' = \frac{-1}{x^2} dx$$

$$\text{Donde } x = 1 + e^{-z} \quad dx = -e^{-z}$$

$$\text{Por lo tanto } f' = \frac{e^{-z}}{(1 + e^{-z})^2} = \frac{1}{(1 + e^{-z})} \cdot \frac{e^{-z}}{(1 + e^{-z})}$$

$$\frac{e^{-z}}{(1+e^{-z})} = \frac{1+e^{-z}-1}{(1+e^{-z})^2} = \frac{1+e^{-z}}{1+e^{-z}} - \frac{1}{1+e^{-z}}$$

$$f' = \frac{e^{-z}}{(1+e^{-z})^2} = \frac{1}{(1+e^{-z})} \cdot \left(\frac{1+e^{-z}}{1+e^{-z}} - \frac{1}{1+e^{-z}} \right) = y(1-y)$$

8. Se calcula el error cuadrático medio

$$\text{Error cuadr medio} = \frac{(T - \text{Activación})^2}{2}$$

9. Se actualizan los pesos de la capa de salida.

$$w_{ji}(t+1) = w_{ji}(t) - (\alpha * x_{ni} * \text{Grad error} * \text{Grad activación})$$

$$w_{ji}(t+1) = w1_DatoAnterior - (0.25 * Neurona1_DatoAnterior * Gerror * Gact)$$

10. Si el error es diferente de cero para cada uno de los patrones aprendidos volver al paso 2.

En esta red neuronal de base radial el valor del parámetro de aprendizaje $\alpha=0.25$.

Propagación de error

En resumen:

- Se convierte la diferencia entre la salida de la red y la salida deseada en una derivada de error.
- Se calcula la derivada del error de la capa oculta a partir de la derivada del error de la capa superior.
- Se usa la derivada del error para obtener la derivada del peso.

El algoritmo de propagación de errores proporciona un método eficiente para calcular las derivadas del error para cada peso.

Tabla 20 Capa de salida de la red neuronal de base radial

w1	w2	w3	w4	w5	w6	w7	b	Agregación	Activación	T	Grad error	Grad activación	Error cuadr. medio
-1.000000	-3.000000	0.000000	-1.000000	1.000000	2.000000	3.000000	-1.000000	0.964086	0.723939	0.142857	0.581082	0.199851	0.168828
-1.027908	-3.000274	-0.000201	-1.018010	0.981983	1.987793	2.979556	-1.029032	1.137105	0.757148	0.142857	0.614291	0.183875	0.188676
-1.055329	-3.000606	-0.000418	-1.033107	0.968453	1.972101	2.959003	-1.057271	1.339694	0.792440	0.142857	0.649582	0.164479	0.210979
-1.081113	-3.001087	-0.000719	-1.046759	0.958295	1.954319	2.938091	-1.083981	1.052194	0.741196	0.142857	0.598339	0.191825	0.179005
-1.106510	-3.001365	-0.000875	-1.057612	0.948556	1.934799	2.919894	-1.112675	0.388030	0.595808	0.142857	0.452951	0.240821	0.102582
-1.130169	-3.001481	-0.000950	-1.069394	0.933449	1.923195	2.905123	-1.139945	0.715970	0.671719	0.142857	0.528862	0.220513	0.139847
-1.159007	-3.001878	-0.001234	-1.087850	0.917418	1.909020	2.882608	-1.169101	0.963871	0.723896	0.142857	0.581039	0.199870	0.168803
-1.187004	-3.002383	-0.001549	-1.102585	0.906305	1.889773	2.860065	-1.198134	1.100311	0.750318	0.857143	-0.106824	0.187341	0.005706
-1.183355	-3.002309	-0.001513	-1.101321	0.907154	1.894238	2.862947	-1.193131	1.126428	0.755179	0.857143	-0.101964	0.184884	0.005198
-1.179922	-3.002237	-0.001479	-1.100129	0.907945	1.898463	2.865672	-1.188418	1.336381	0.791894	0.857143	-0.065249	0.164798	0.002129
-1.177947	-3.002168	-0.001445	-1.099375	0.908358	1.900983	2.867415	-1.185729	-0.076313	0.480931	0.571429	-0.090498	0.249636	0.004095
-1.173375	-3.002102	-0.001381	-1.094483	0.912881	1.902159	2.871211	-1.180082	0.158060	0.539433	0.142857	0.396576	0.248445	0.078636
-1.195780	-3.002361	-0.001603	-1.113357	0.894470	1.894841	2.853810	-1.204713	-0.265649	0.433976	0.714286	-0.280310	0.245641	0.039287
-1.182867	-3.002316	-0.001564	-1.103815	0.909336	1.898289	2.861708	-1.187500	-0.466952	0.385338	0.714286	-0.328948	0.236853	0.054103
-1.172868	-3.002279	-0.001523	-1.092089	0.928719	1.899777	2.868160	-1.168021	-0.321927	0.420206	0.714286	-0.294079	0.243633	0.043241
-1.161741	-3.002218	-0.001457	-1.079615	0.946030	1.901769	2.875882	-1.150110	-0.320696	0.420506	0.714286	-0.293780	0.243681	0.043153
-1.151511	-3.002166	-0.001400	-1.067524	0.963521	1.903438	2.882912	-1.132213	-0.256410	0.436246	0.714286	-0.278039	0.245935	0.038653
-1.141161	-3.002074	-0.001295	-1.054049	0.979201	1.905217	2.890910	-1.115118	-0.170714	0.457425	0.714286	-0.256861	0.248187	0.032989
-1.131162	-3.001992	-0.001203	-1.041664	0.993974	1.907005	2.898483	-1.099180	0.891438	0.709187	0.142857	0.566330	0.206241	0.160365
-1.159388	-3.002946	-0.001967	-1.065183	0.979274	1.893923	2.871621	-1.128380	0.957147	0.722550	1.000000	-0.277450	0.200471	0.038489
-1.146090	-3.002337	-0.001500	-1.054548	0.985101	1.901066	2.884972	-1.114475	1.096140	0.749536	0.142857	0.606679	0.187732	0.184030
-1.173751	-3.003460	-0.002320	-1.074812	0.973590	1.885049	2.858075	-1.142948	1.011790	0.733370	0.142857	0.590513	0.195538	0.174353
-1.201637	-3.004663	-0.003196	-1.095224	0.962290	1.868565	2.830639	-1.171815	0.946170	0.720344	1.000000	-0.279656	0.201448	0.039104
-1.189959	-3.003334	-0.002220	-1.085730	0.965869	1.876735	2.844610	-1.157731	0.789561	0.687737	0.142857	0.544880	0.214755	0.148447
-1.219155	-3.004053	-0.002741	-1.105880	0.951932	1.861120	2.818955	-1.186985	0.667107	0.660855	0.142857	0.517998	0.224126	0.134161

-1.247973	-3.004853	-0.003335	-1.126823	0.938030	1.846150	2.792911	-1.216009	0.454398	0.611684	0.142857	0.468827	0.237527	0.109900
-1.275767	-3.005356	-0.003699	-1.145439	0.923474	1.832120	2.769951	-1.243849	0.761937	0.681774	1.000000	-0.318226	0.216958	0.050634
-1.260750	-3.004021	-0.002744	-1.133999	0.928202	1.842694	2.786977	-1.226589	0.937530	0.718600	1.000000	-0.281400	0.202214	0.039593
-1.249109	-3.002732	-0.001873	-1.125752	0.931278	1.852334	2.800637	-1.212363	0.601018	0.645889	1.000000	-0.354111	0.228716	0.062697
-1.234074	-3.000061	0.000179	-1.112072	0.935645	1.862794	2.820446	-1.192115	0.953562	0.721831	0.142857	0.578974	0.200791	0.167605
-1.261720	-3.001367	-0.000722	-1.130810	0.925770	1.844237	2.792994	-1.221178	-3.318201	0.034952	0.428571	-0.393619	0.033730	0.077468
-1.261640	-2.998841	0.002443	-1.130552	0.925782	1.844293	2.793317	-1.217859	-3.760121	0.022751	0.285714	-0.262963	0.022234	0.034575
-1.261592	-2.997429	0.003549	-1.130480	0.925785	1.844380	2.793494	-1.216397	-3.852426	0.020787	0.285714	-0.264927	0.020355	0.035093
-1.261555	-2.996106	0.004604	-1.130422	0.925788	1.844447	2.793635	-1.215049	-3.998051	0.018021	0.285714	-0.267694	0.017696	0.035830
-1.261534	-2.994924	0.005653	-1.130381	0.925789	1.844480	2.793726	-1.213865	-3.986966	0.018218	0.285714	-0.267496	0.017886	0.035777
-1.261518	-2.993754	0.006759	-1.130347	0.925790	1.844503	2.793796	-1.212669	-3.853417	0.020767	0.285714	-0.264948	0.020335	0.035099
-1.261479	-2.992436	0.008000	-1.130266	0.925793	1.844552	2.793949	-1.211322	-3.517212	0.028826	0.428571	-0.399745	0.027995	0.079898
-1.261350	-2.989926	0.010543	-1.129984	0.925808	1.844679	2.794405	-1.208524	-3.983303	0.018284	0.285714	-0.267431	0.017949	0.035760
-1.261336	-2.988749	0.011613	-1.129954	0.925809	1.844702	2.794471	-1.207324	-3.823930	0.021375	0.428571	-0.407197	0.020918	0.082904
-1.261314	-2.986794	0.013689	-1.129895	0.925811	1.844728	2.794575	-1.205195	-3.812273	0.021620	0.428571	-0.406951	0.021153	0.082805
-1.261287	-2.984809	0.015811	-1.129824	0.925813	1.844759	2.794699	-1.203043	-3.640383	0.025571	0.428571	-0.403000	0.024917	0.081205
-1.261250	-2.982623	0.018322	-1.129713	0.925817	1.844794	2.794867	-1.200532	-3.449775	0.030776	0.428571	-0.397796	0.029828	0.079121
-1.261138	-2.980072	0.021119	-1.129432	0.925831	1.844890	2.795280	-1.197566	-3.751507	0.022944	0.428571	-0.405628	0.022417	0.082267
-1.261121	-2.978039	0.023286	-1.129384	0.925832	1.844911	2.795366	-1.195293	-3.790275	0.022090	0.428571	-0.406481	0.021602	0.082613
-1.261099	-2.976024	0.025422	-1.129325	0.925834	1.844937	2.795473	-1.193097	-3.740149	0.023200	0.428571	-0.405372	0.022661	0.082163
-1.261070	-2.973933	0.027703	-1.129242	0.925836	1.844970	2.795610	-1.190801	-3.145890	0.041254	0.428571	-0.387318	0.039552	0.075008
-1.261059	-2.971338	0.031021	-1.129200	0.925837	1.844980	2.795674	-1.186971	-3.414583	0.031843	0.428571	-0.396729	0.030829	0.078697
-1.260945	-2.968712	0.033909	-1.128912	0.925852	1.845078	2.796096	-1.183913	-3.072986	0.044235	0.428571	-0.384336	0.042279	0.073857
-1.260903	-2.965959	0.037709	-1.128743	0.925858	1.845105	2.796295	-1.179851	-3.229171	0.038083	0.428571	-0.390489	0.036632	0.076241
-1.260862	-2.963317	0.041165	-1.128591	0.925863	1.845135	2.796489	-1.176275	-0.795529	0.310983	0.571429	-0.260446	0.214272	0.033916
-1.255204	-2.962329	0.042481	-1.116186	0.930439	1.846398	2.804624	-1.162323	-0.197168	0.450867	0.571429	-0.120561	0.247586	0.007268
-1.250576	-2.961629	0.043223	-1.109226	0.933007	1.848003	2.810853	-1.154861	0.129124	0.532236	0.571429	-0.039192	0.248961	0.000768
-1.248635	-2.961497	0.043351	-1.106895	0.934163	1.848702	2.813045	-1.152422	-0.336064	0.416766	0.571429	-0.154663	0.243072	0.011960

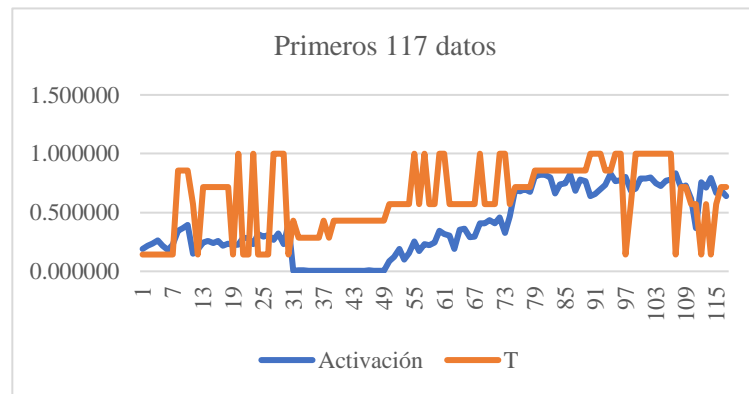
-1.243815	-2.960027	0.044913	-1.099145	0.936325	1.850613	2.820511	-1.143023	-0.427579	0.394705	0.571429	-0.176724	0.238913	0.015616
-1.237700	-2.959631	0.045390	-1.088702	0.941907	1.851992	2.827605	-1.132468	0.387933	0.595785	1.000000	-0.404215	0.240825	0.081695
-1.216643	-2.958430	0.046471	-1.066451	0.953537	1.860460	2.850294	-1.108131	-0.103180	0.474228	0.571429	-0.097201	0.249336	0.004724
-1.212747	-2.958088	0.046851	-1.060461	0.956313	1.861567	2.855032	-1.102073	0.404498	0.599768	1.000000	-0.400232	0.240046	0.080093
-1.194392	-2.956139	0.048675	-1.038621	0.965305	1.869265	2.877574	-1.078054	0.066944	0.516730	0.571429	-0.054699	0.249720	0.001496
-1.192041	-2.955978	0.048849	-1.035220	0.967038	1.869934	2.880283	-1.074639	-0.028262	0.492935	0.571429	-0.078494	0.249950	0.003081
-1.188758	-2.955810	0.049038	-1.030321	0.969864	1.870752	2.883873	-1.069734	0.618159	0.649800	1.000000	-0.350200	0.227560	0.061320
-1.171456	-2.954851	0.049901	-1.012109	0.979490	1.877682	2.902407	-1.049811	0.642220	0.655255	1.000000	-0.344745	0.225896	0.059425
-1.156123	-2.953372	0.051264	-0.994456	0.987014	1.884193	2.920833	-1.030342	0.110446	0.527584	0.571429	-0.043845	0.249239	0.000961
-1.154306	-2.953281	0.051367	-0.991729	0.988604	1.884639	2.922813	-1.027610	-0.247497	0.438440	0.571429	-0.132989	0.246210	0.008843
-1.150496	-2.952753	0.052041	-0.984100	0.991655	1.885529	2.928008	-1.019424	0.091138	0.522769	0.571429	-0.048660	0.249482	0.001184
-1.148492	-2.952681	0.052123	-0.981118	0.993644	1.885978	2.930057	-1.016390	0.108630	0.527131	0.571429	-0.044298	0.249264	0.000981
-1.146659	-2.952615	0.052198	-0.978406	0.995457	1.886391	2.931927	-1.013629	-0.014309	0.496423	0.571429	-0.075006	0.249987	0.002813
-1.143952	-2.952438	0.052412	-0.973770	0.997926	1.887001	2.935074	-1.008941	-0.005344	0.498664	0.571429	-0.072764	0.249998	0.002647
-1.141342	-2.952263	0.052623	-0.969275	1.000298	1.887590	2.938125	-1.004394	0.793050	0.688486	1.000000	-0.311514	0.214473	0.048521
-1.128203	-2.950992	0.053796	-0.954124	1.006749	1.893163	2.953925	-0.987691	0.135096	0.533723	0.571429	-0.037706	0.248863	0.000711
-1.126722	-2.950936	0.053861	-0.951823	1.008269	1.893481	2.955458	-0.985345	0.218938	0.554517	0.571429	-0.016912	0.247028	0.000143
-1.126024	-2.950912	0.053889	-0.950797	1.008956	1.893639	2.956169	-0.984301	0.267734	0.566537	0.571429	-0.004892	0.245573	0.000012
-1.125824	-2.950901	0.053901	-0.950498	1.009129	1.893689	2.956388	-0.984000	0.852780	0.701150	1.000000	-0.298850	0.209539	0.044656
-1.113621	-2.949678	0.055033	-0.936302	1.015095	1.898863	2.971174	-0.968345	0.500146	0.622494	1.000000	-0.377506	0.234995	0.071256
-1.100489	-2.946421	0.058251	-0.917863	1.020582	1.904668	2.990458	-0.946167	0.351007	0.586862	0.571429	0.015433	0.242455	0.000119
-1.101101	-2.946451	0.058218	-0.918795	1.020029	1.904522	2.989794	-0.947102	0.037881	0.509469	0.714286	-0.204817	0.249910	0.020975
-1.095083	-2.946434	0.058237	-0.911856	1.032812	1.905364	2.993478	-0.934306	-0.067700	0.483082	0.714286	-0.231204	0.249714	0.026728
-1.090312	-2.946428	0.058243	-0.906593	1.046476	1.905905	2.995951	-0.919872	0.043511	0.510876	0.714286	-0.203410	0.249882	0.020688
-1.085090	-2.946419	0.058253	-0.901032	1.058965	1.906595	2.998828	-0.907165	-0.027819	0.493046	0.714286	-0.221240	0.249952	0.024474
-1.080275	-2.946416	0.058257	-0.896828	1.071408	1.907245	3.001083	-0.893340	1.806760	0.858970	0.857143	0.001827	0.121141	0.000002
-1.080303	-2.946419	0.058255	-0.896839	1.071404	1.907192	3.001052	-0.893396	1.973884	0.878028	0.857143	0.020885	0.107095	0.000218
-1.080651	-2.946450	0.058240	-0.896983	1.071350	1.906653	3.000681	-0.893955	1.867827	0.866207	0.857143	0.009064	0.115893	0.000041

-1.080791	-2.946464	0.058234	-0.897035	1.071332	1.906394	3.000531	-0.894217	1.832130	0.862015	0.857143	0.004872	0.118945	0.000012
-1.080868	-2.946475	0.058228	-0.897067	1.071322	1.906259	3.000440	-0.894362	0.812777	0.692701	0.857143	-0.164442	0.212866	0.013521
-1.079386	-2.946238	0.058302	-0.896755	1.071405	1.912810	3.002087	-0.885611	1.399769	0.802147	0.857143	-0.054996	0.158707	0.001512
-1.078608	-2.946015	0.058400	-0.896452	1.071474	1.914685	3.003116	-0.883429	1.649463	0.838819	0.857143	-0.018324	0.135202	0.000168
-1.078295	-2.945934	0.058443	-0.896292	1.071514	1.915201	3.003533	-0.882810	2.026724	0.883575	0.857143	0.026432	0.102871	0.000349
-1.078748	-2.945978	0.058420	-0.896500	1.071438	1.914570	3.003036	-0.883490	0.850064	0.700581	0.857143	-0.156562	0.209767	0.012256
-1.076771	-2.945932	0.058434	-0.896165	1.071628	1.920903	3.004526	-0.875279	1.323801	0.789813	0.857143	-0.067329	0.166008	0.002267
-1.075339	-2.945910	0.058442	-0.895809	1.071885	1.923347	3.005558	-0.872485	1.209551	0.770219	0.857143	-0.086923	0.176981	0.003778
-1.073210	-2.945888	0.058451	-0.895266	1.072341	1.926489	3.006994	-0.868639	1.436948	0.807982	1.000000	-0.192018	0.155147	0.018436
-1.068939	-2.944460	0.059375	-0.892143	1.073083	1.931289	3.013200	-0.861191	1.517009	0.820098	1.000000	-0.179902	0.147537	0.016182
-1.064901	-2.943263	0.060167	-0.889106	1.073844	1.935501	3.018951	-0.854556	1.642596	0.837888	1.000000	-0.162112	0.135832	0.013140
-1.061413	-2.942363	0.060755	-0.886573	1.074513	1.939146	3.023772	-0.849051	1.083501	0.747156	0.857143	-0.109987	0.188914	0.006049
-1.060320	-2.942158	0.060825	-0.886305	1.074581	1.943333	3.025043	-0.843856	1.614306	0.834008	0.857143	-0.023135	0.138438	0.000268
-1.059902	-2.942148	0.060829	-0.886193	1.074649	1.944086	3.025375	-0.843055	1.893484	0.869152	1.000000	-0.130848	0.113727	0.008561
-1.056667	-2.941860	0.061035	-0.883730	1.075665	1.946367	3.029044	-0.839335	1.907761	0.870767	1.000000	-0.129233	0.112532	0.008351
-1.053497	-2.941581	0.061234	-0.881320	1.076666	1.948596	3.032630	-0.835700	1.835267	0.862388	0.142857	0.719531	0.118675	0.258862
-1.074066	-2.942492	0.060575	-0.896287	1.068475	1.936246	3.012306	-0.857047	1.187202	0.766240	0.571429	0.194812	0.179116	0.018976
-1.081819	-2.942805	0.060292	-0.904276	1.063679	1.933356	3.004503	-0.865771	1.307237	0.787050	1.000000	-0.212950	0.167602	0.022674
-1.073629	-2.942451	0.060596	-0.896454	1.068188	1.936797	3.012790	-0.856848	1.776430	0.855255	1.000000	-0.144745	0.123794	0.010475
-1.069386	-2.942230	0.060756	-0.893296	1.069812	1.939398	3.017122	-0.852368	1.811363	0.859527	1.000000	-0.140473	0.120741	0.009866
-1.065437	-2.941997	0.060927	-0.890319	1.071265	1.941879	3.021261	-0.848128	1.809529	0.859305	1.000000	-0.140695	0.120900	0.009898
-1.061408	-2.941788	0.061078	-0.887321	1.072808	1.944348	3.025372	-0.843876	1.745439	0.851377	1.000000	-0.148623	0.126534	0.011044
-1.057414	-2.941383	0.061387	-0.883922	1.074161	1.946889	3.030069	-0.839174	1.634577	0.836796	1.000000	-0.163204	0.136569	0.013318
-1.052726	-2.940906	0.061769	-0.879585	1.075909	1.949570	3.035632	-0.833602	1.830466	0.861817	1.000000	-0.138183	0.119088	0.009547
-1.049332	-2.940511	0.062061	-0.876802	1.076950	1.951936	3.039715	-0.829488	1.857353	0.864988	1.000000	-0.135012	0.116784	0.009114
-1.046064	-2.940138	0.062334	-0.874146	1.077950	1.954222	3.043625	-0.825546	1.825309	0.861202	0.142857	0.718345	0.119533	0.258010
-1.066969	-2.940962	0.061733	-0.889413	1.069178	1.942178	3.023416	-0.847013	0.009736	0.502434	0.714286	-0.211852	0.249994	0.022441
-1.063423	-2.940958	0.061737	-0.885036	1.081195	1.942516	3.025243	-0.833772	0.137530	0.534328	0.714286	-0.179957	0.248822	0.016192

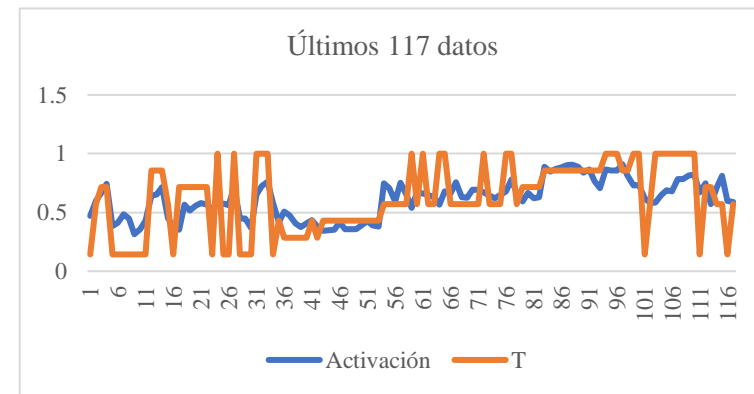
-1.058180	-2.940953	0.061742	-0.881566	1.090529	1.943569	3.027688	-0.822578	0.557170	0.635798	0.571429	0.064369	0.231559	0.002072
-1.060607	-2.941029	0.061654	-0.885182	1.087971	1.943049	3.025272	-0.826304	-0.130653	0.467383	0.571429	-0.104045	0.248936	0.005413
-1.058366	-2.940819	0.061966	-0.879641	1.090666	1.943402	3.028170	-0.819829	1.271648	0.781025	0.142857	0.638168	0.171025	0.203629
-1.083326	-2.941568	0.061302	-0.903917	1.074330	1.934271	3.004580	-0.847115	1.053487	0.741444	0.571429	0.170015	0.191705	0.014453
-1.090487	-2.941788	0.061098	-0.911425	1.069272	1.931873	2.997725	-0.855263	1.251187	0.777505	0.142857	0.634648	0.172991	0.201389
-1.117470	-2.942256	0.060733	-0.931654	1.052886	1.920249	2.975357	-0.882710	0.910306	0.713063	0.571429	0.141634	0.204604	0.010030
-1.123711	-2.942480	0.060522	-0.938452	1.048569	1.918159	2.969178	-0.889955	0.046912	0.511726	0.714286	-0.202560	0.249863	0.020515
-1.118065	-2.942454	0.060554	-0.930546	1.060769	1.918878	2.973057	-0.877302	-0.116859	0.470818	0.714286	-0.243467	0.249148	0.029638

5.5 RESULTADOS DEL COMPORTAMIENTO DE LA RED NEURONAL DE BASE RADIAL

Se realizaron un total 11636 iteraciones en Microsoft Excel, en la tabla 19 contiene sólo los primeros 117 datos. En la figura 30, se realizó una comparación gráfica que muestra el entrenamiento de la red neuronal de base radial con siete centros; a) comportamiento de los primeros 117 datos y b) los últimos 117 datos.



a)



b)

Figura 30 Entrenamiento de la red con siete centros

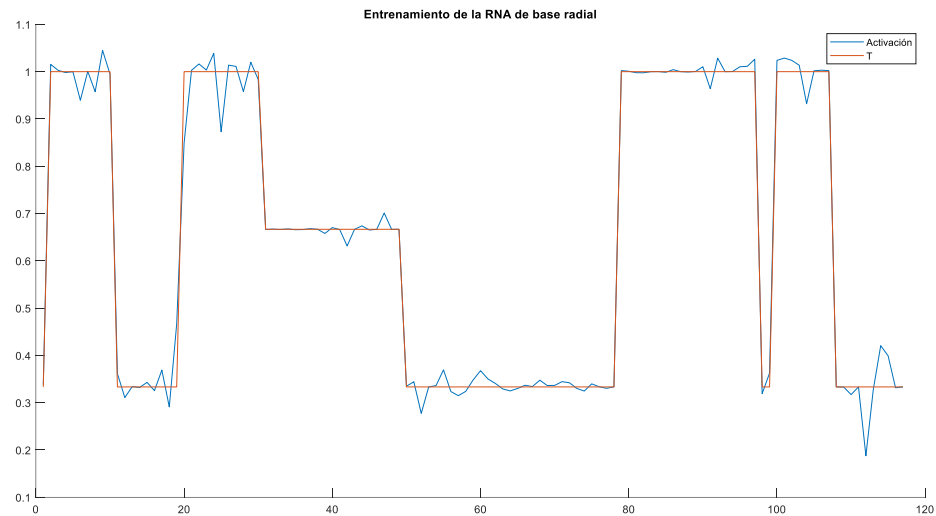
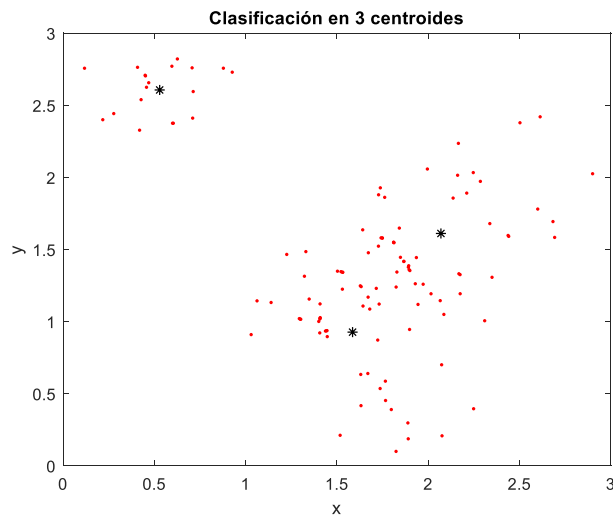
5.5.1 PRUEBAS REALIZADAS CON N CENTROIDES

Con los datos de entrada de la tabla 7, utilizando la metodología anterior y modificando el número de neuronas se produjeron los siguientes resultados en Matlab, se realizó un código de programación (ver Apéndice A) obteniendo la clasificación deseada T que será variable de entrada a la función newrbe, con la finalidad de mejorar el rendimiento de la red neuronal de base radial.

- Entrenamiento de la red neuronal de base radial con 3 centroides utilizando Matlab

Núm. de centroides: 3

Entrenamientos: 100



Centroides : (1.5854, 0.927), (0.5288, 2.6064) y (2.0693, 1.6112)

Figura 31 Entrenamiento RNA de base radial con 3 centros

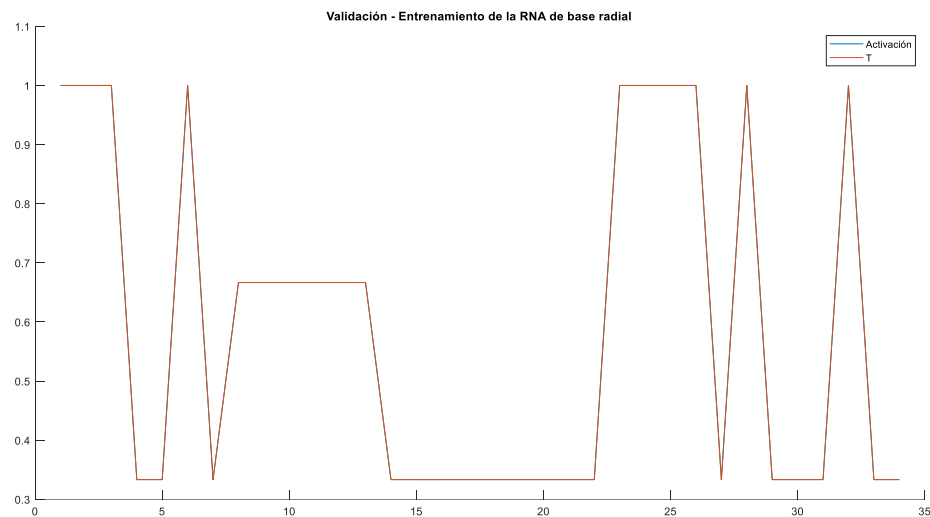
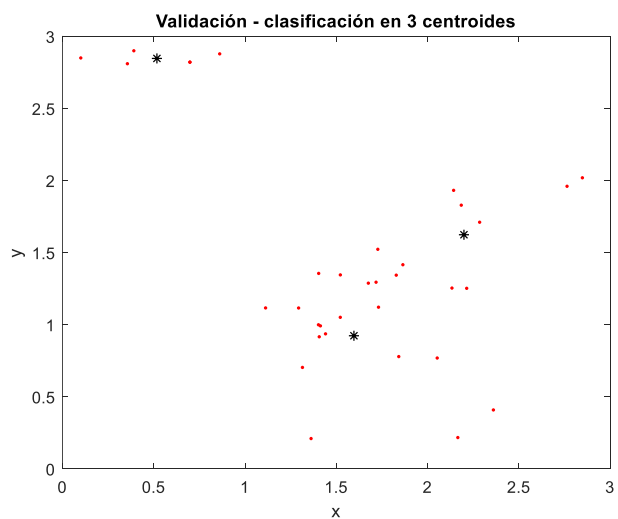


Figura 32 Validación RNA de base radial con 3 centros

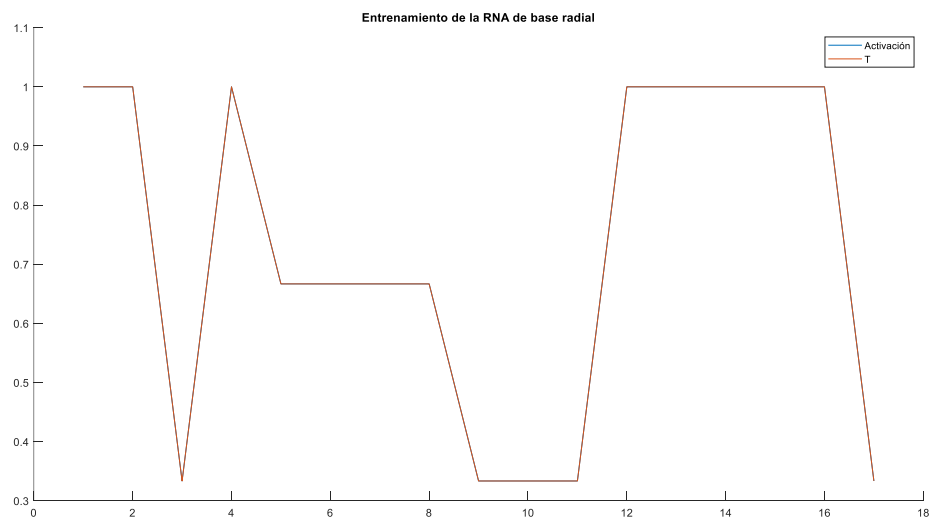
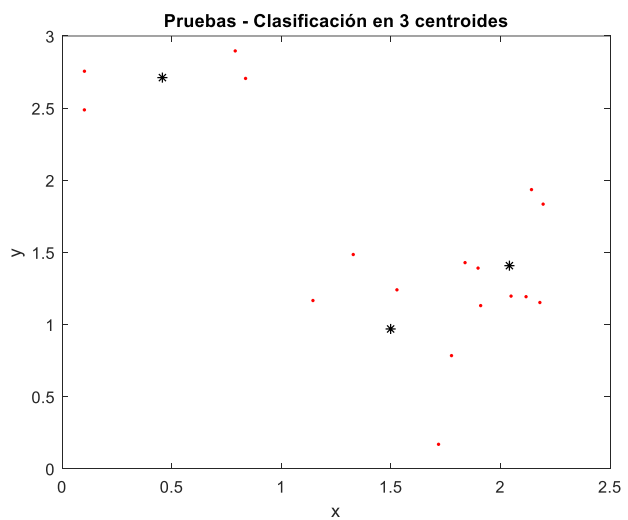
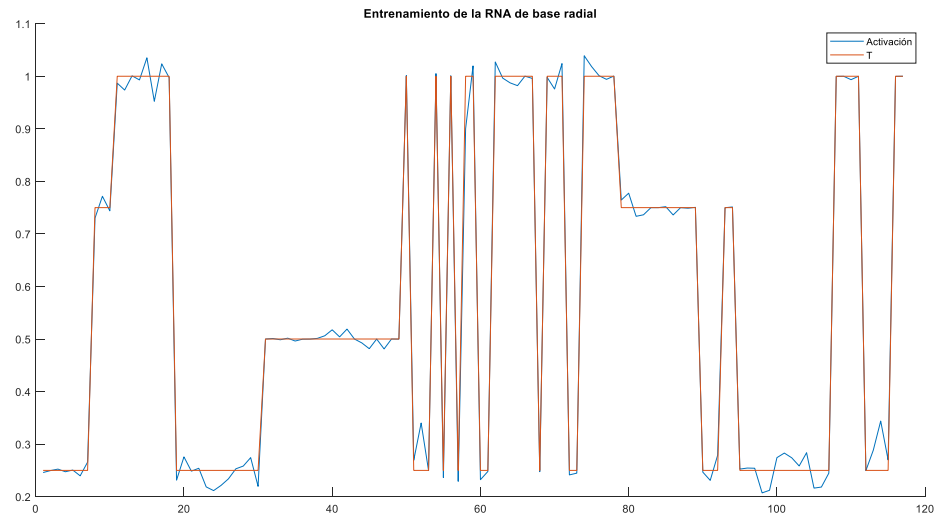
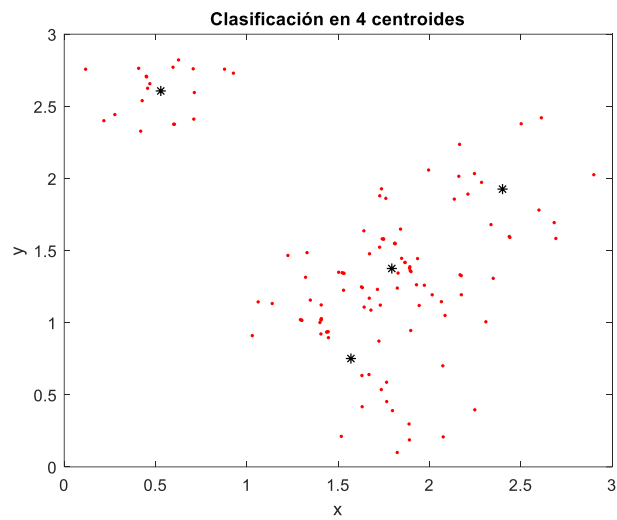


Figura 33 Pruebas RNA de base radial con 3 centros

- Entrenamiento de la red neuronal de base radial con 4 centroides utilizando Matlab

Núm. de centroides: 4

Entrenamientos: 100



Centroides: (1.7939, 1.3755), (0.5288, 2.6064) y (1.57, 0.7513)

Figura 34 Entrenamiento RNA de base radial con 4 centros

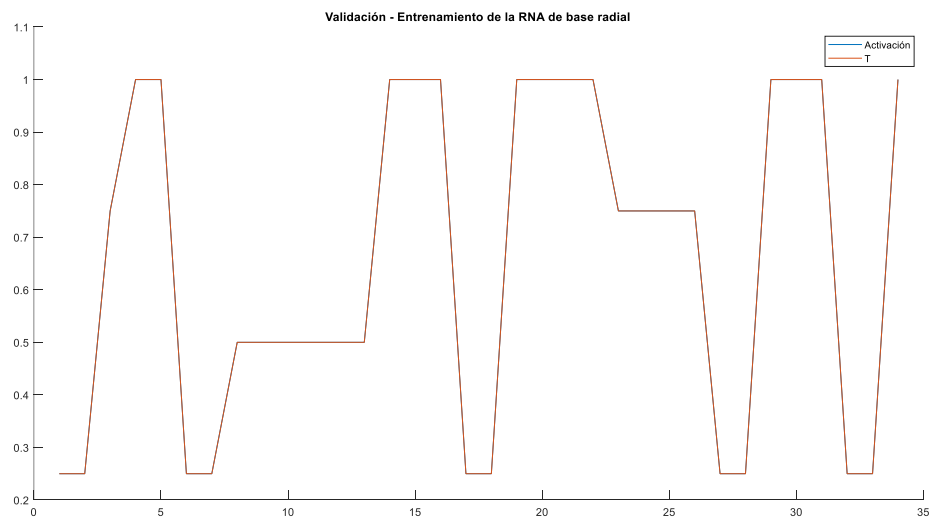
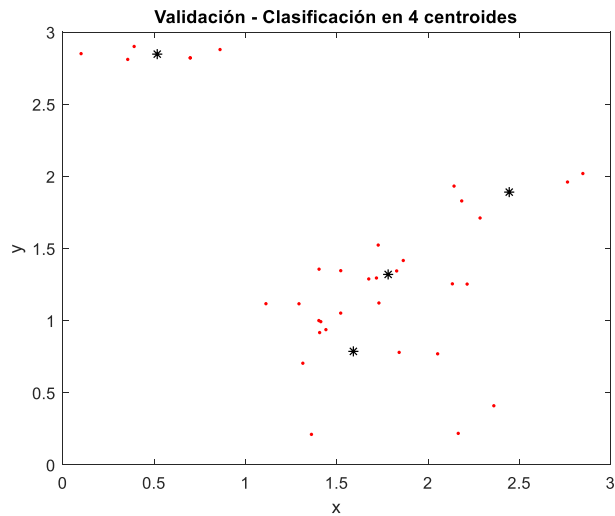


Figura 35 Validación RNA de base radial con 4 centros

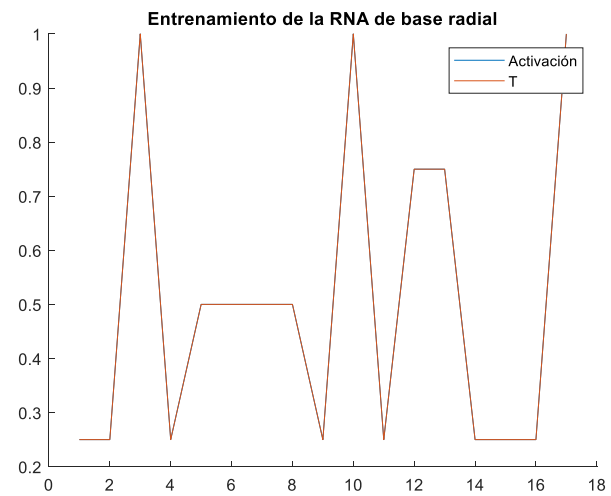
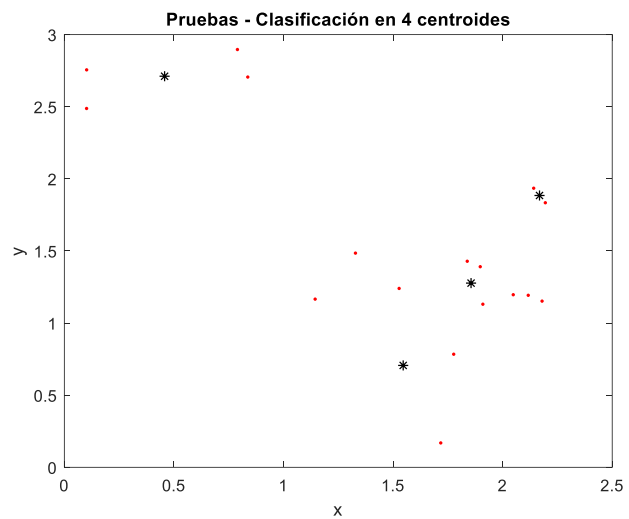
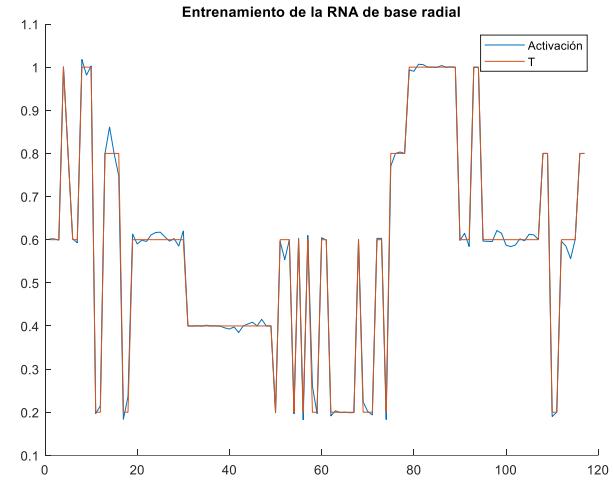
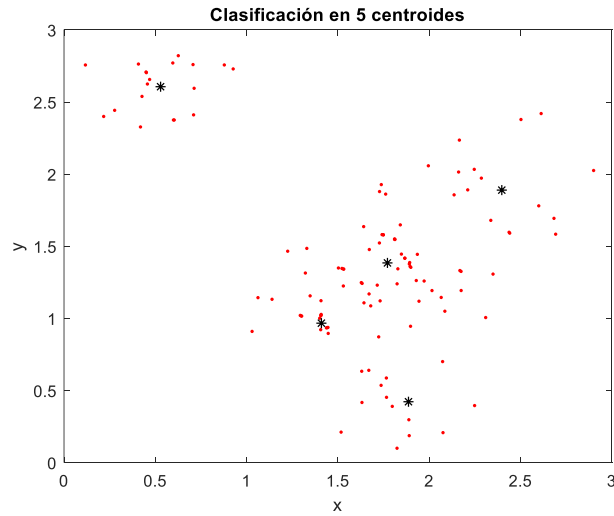


Figura 36 Pruebas RNA de base radial con 4 centros

- Entrenamiento de la red neuronal de base radial con 5 centroides utilizando Matlab

Núm. de centroides: 5

Entrenamientos: 100



Centroides: (1.4096, 0.967), (0.5288, 2.6064), (1.7711, 1.3849), (1.8863, 0.4223) y (2.3972, 1.8895)

Figura 37 Entrenamiento RNA de base radial con 5 centros

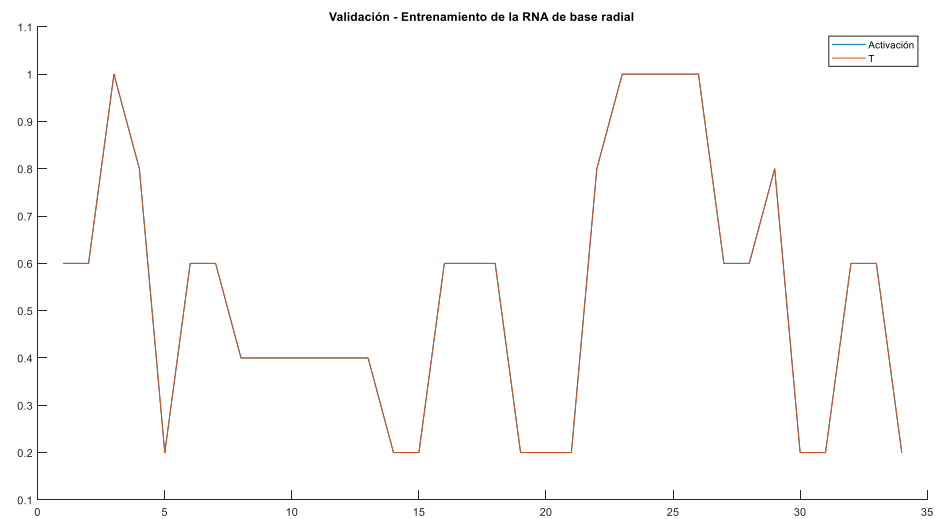
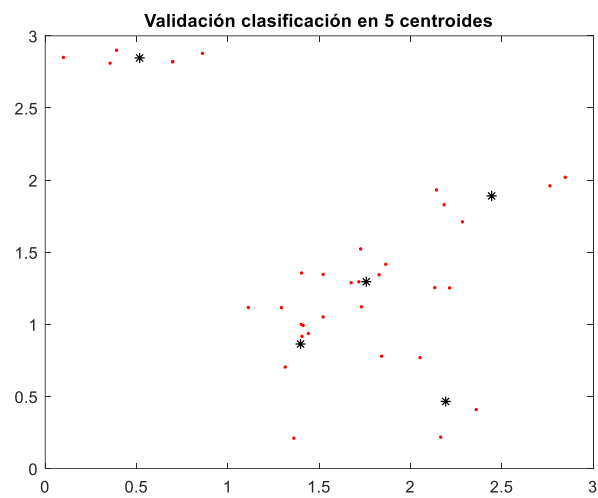


Figura 38 Validación RNA de base radial con 5 centros

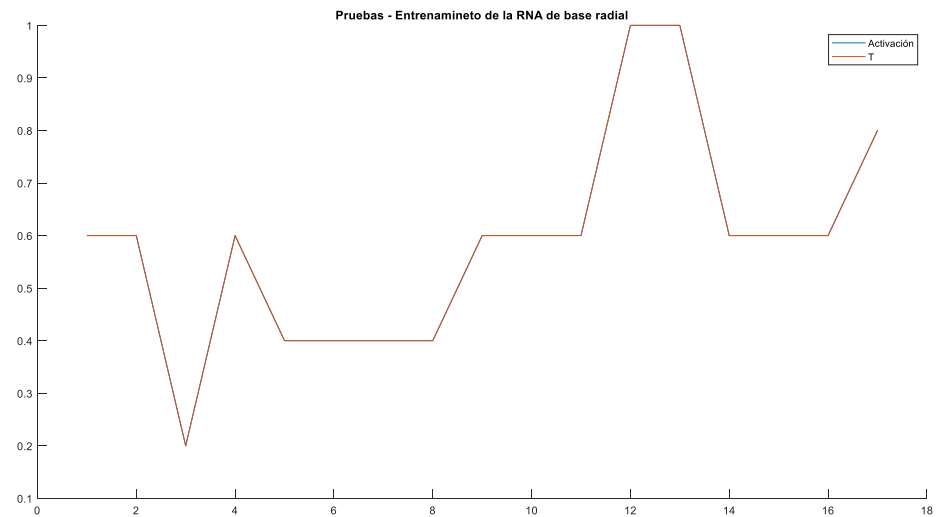
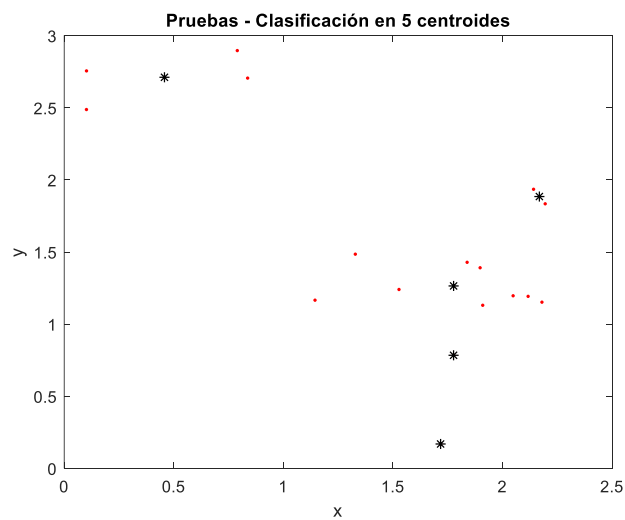
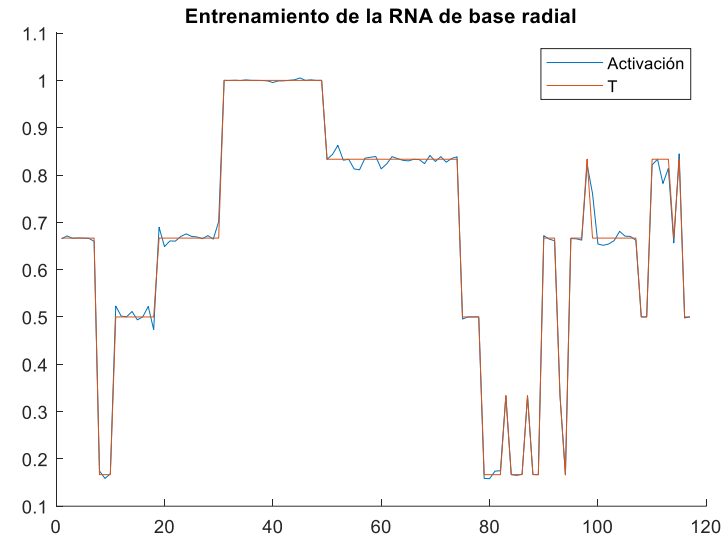
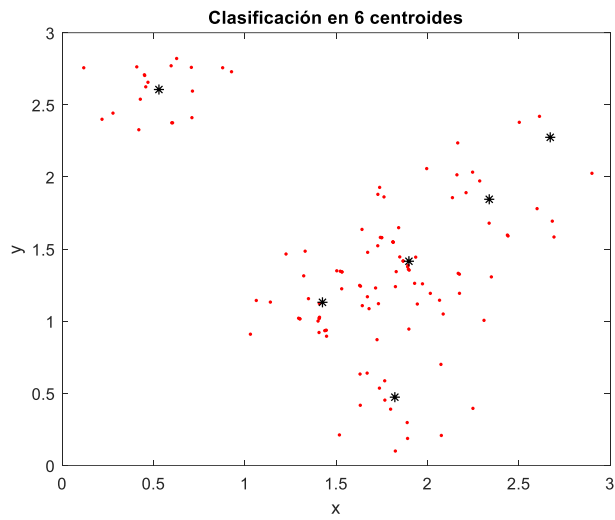


Figura 39 Pruebas RNA de base radial con 5 centros

- Entrenamiento de la red neuronal de base radial con 6 centroides utilizando Matlab

Núm. de centroides: 6

Entrenamientos: 100



Centroides: (2.3376, 1.8453), (2.6716, 2.2749), (1.8208, 0.4735), (1.8978, 1.4158), (1.4243, 1.1311) y (0.5288, 2.6064)

Figura 40 Entrenamiento RNA de base radial con 6 centros

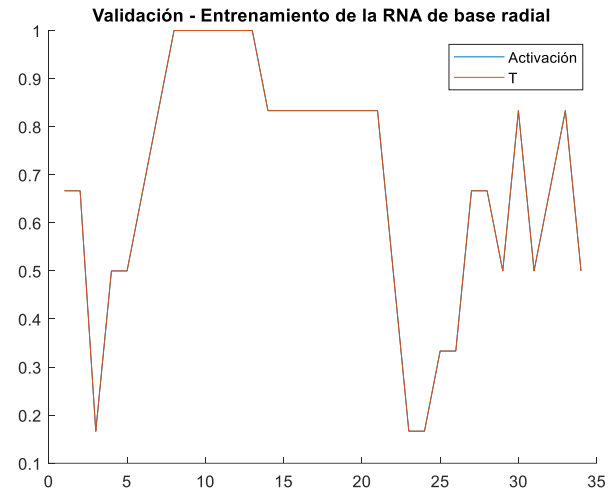
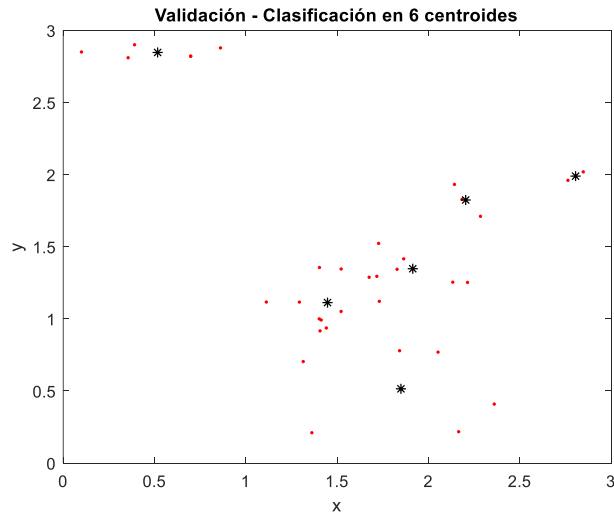


Figura 41 Validación RNA de base radial con 6 centros

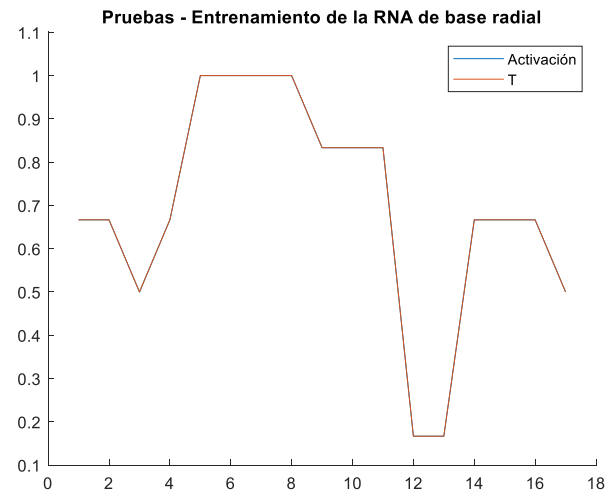
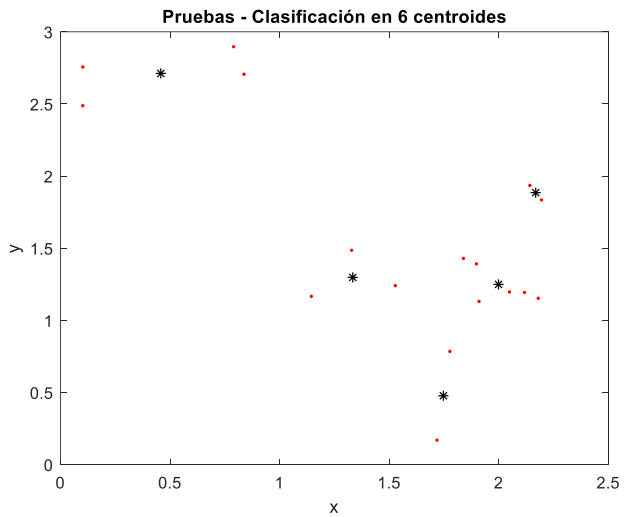
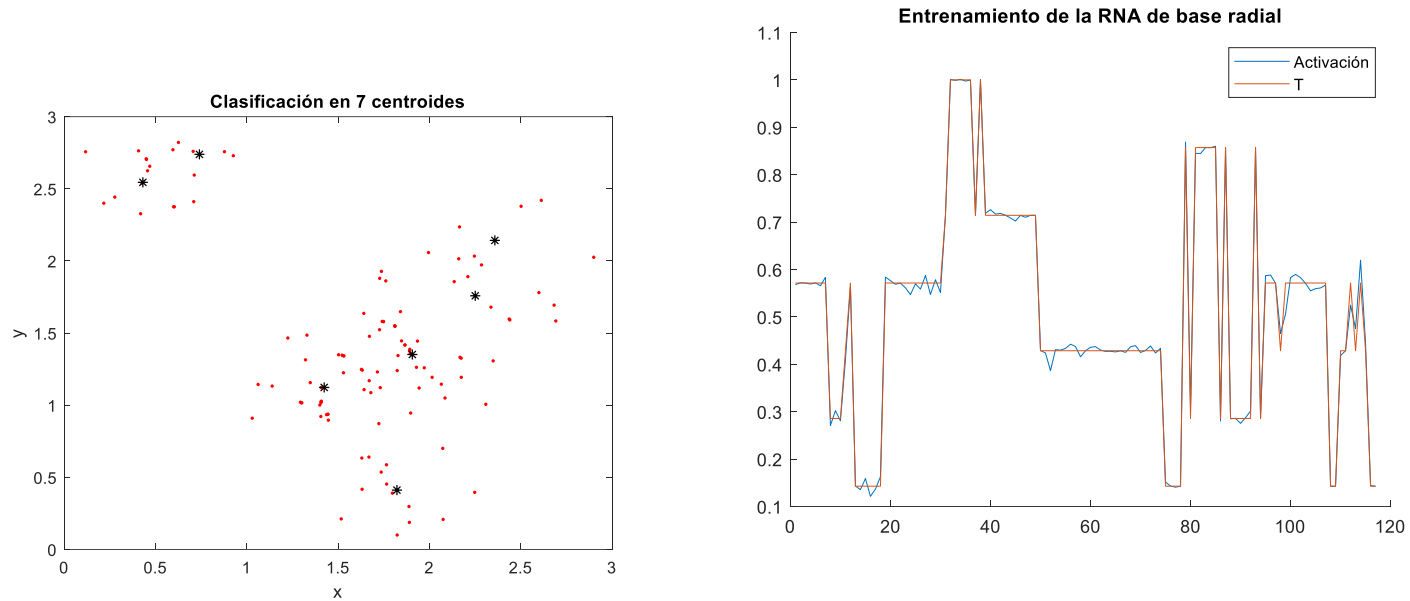


Figura 42 Pruebas RNA de base radial con 6 centros

- Entrenamiento de la red neuronal de base radial con 7 centroides utilizando Matlab

Núm. de centroides: 7

Entrenamientos: 100



Centroides: (1.8223, 0.4113), (2.2511, 1.7585), (1.424, 1.1231), (1.9066, 1.3515), (0.431, 2.5451), (2.3584, 2.1425) y (0.7408, 2.7391)

Figura 43 Entrenamiento RNA de base radial con 7 centros

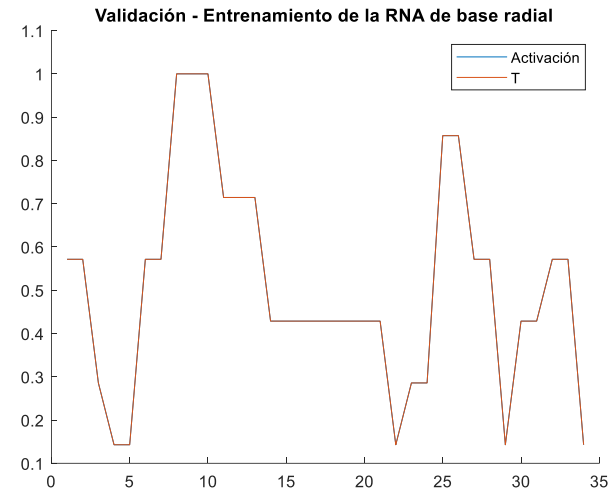
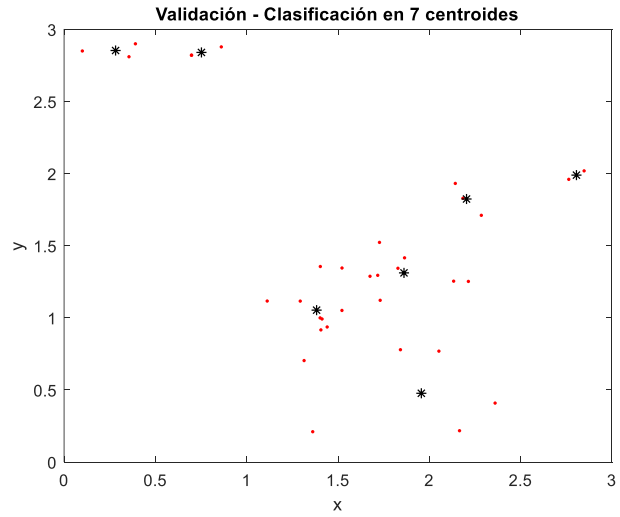


Figura 44 Validación RNA de base radial con 7 centros

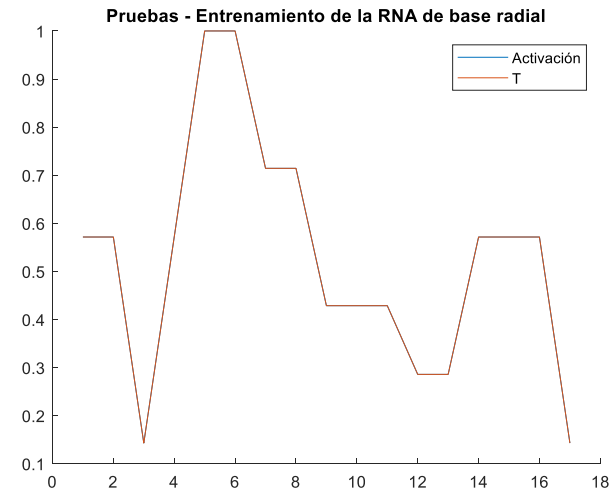
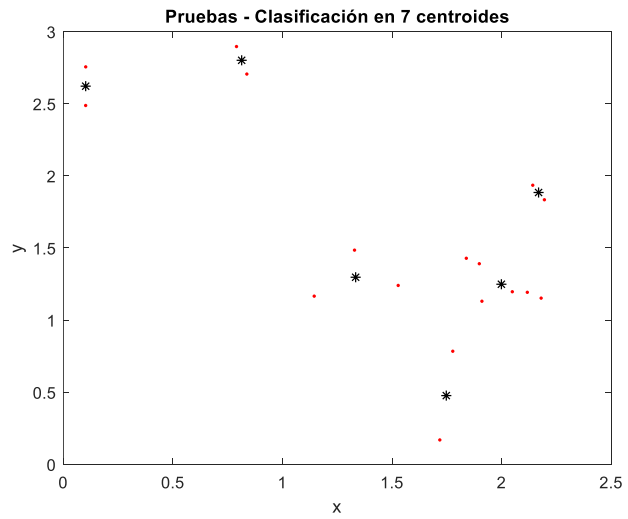
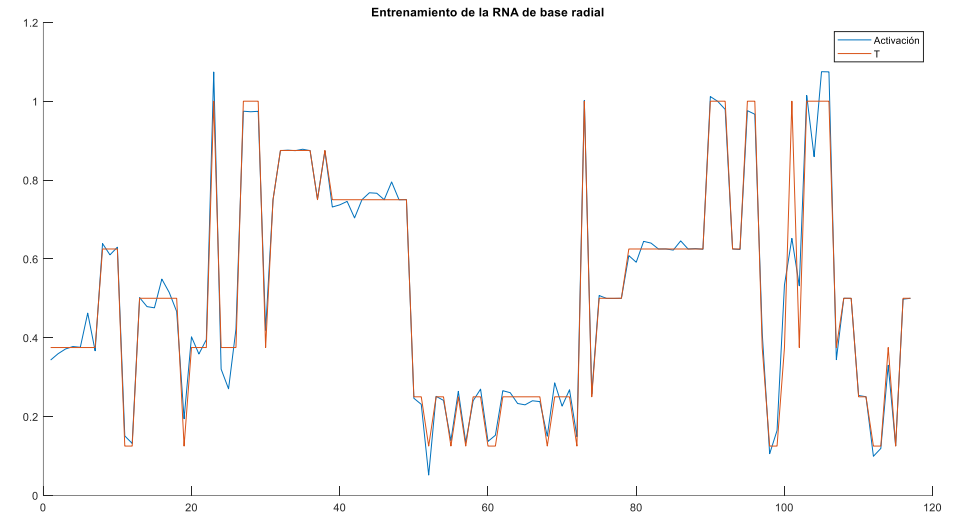
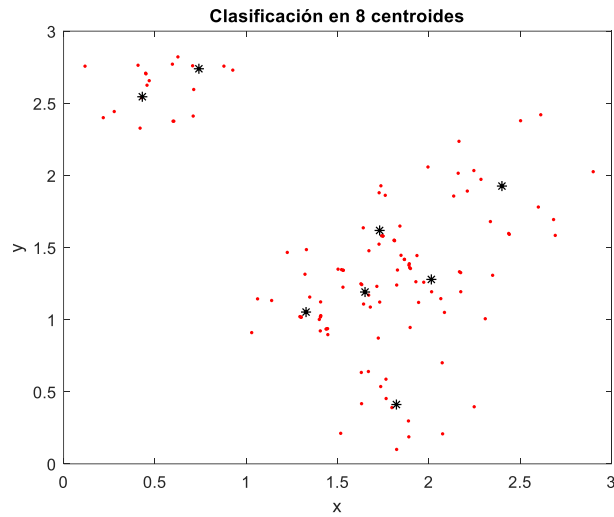


Figura 45 Pruebas RNA de base radial con 7 centros

- Entrenamiento de la red neuronal de base radial con 8 centroides utilizando Matlab

Núm. de centroides: 8

Entrenamientos: 100



Centroides: (1.6506, 1.1913), (1.3276, 1.0525), (2.0136, 1.2787), (1.8223, 0.4113), (2.4002, 1.9259), (0.431, 2.5451), (0.7408, 2.7391)
y (1.7304, 1.6182)

Figura 46 Entrenamiento RNA de base radial con 8 centros

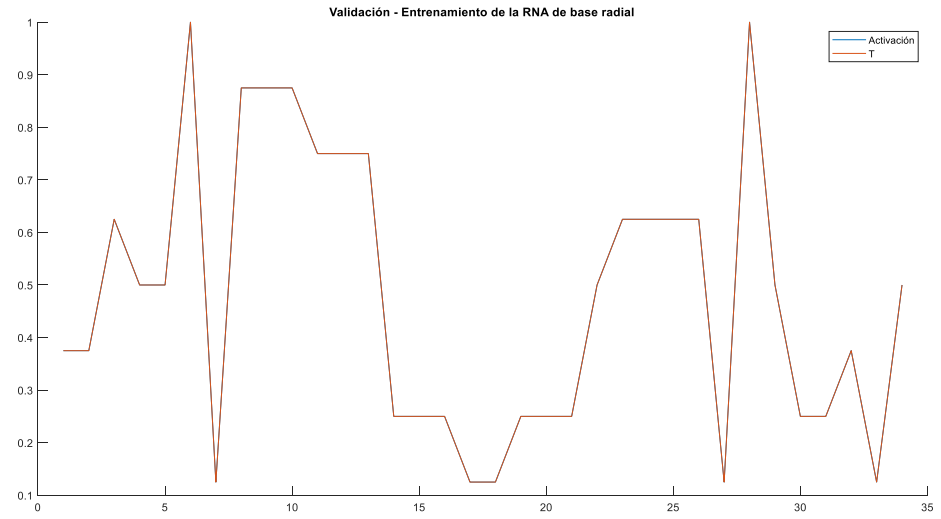
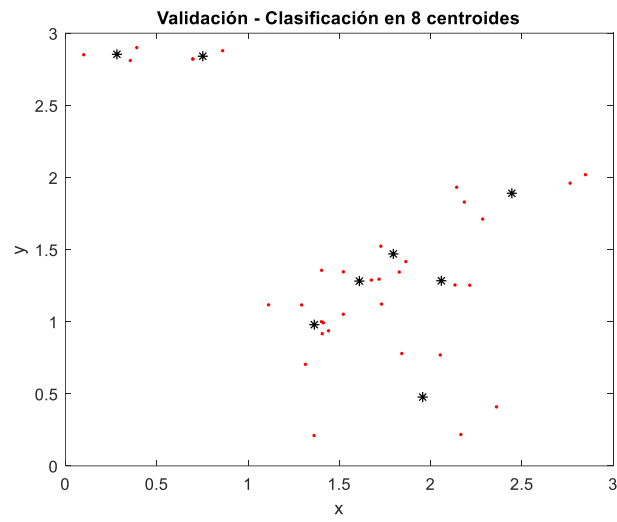


Figura 47 Validación RNA de base radial con 8 centros

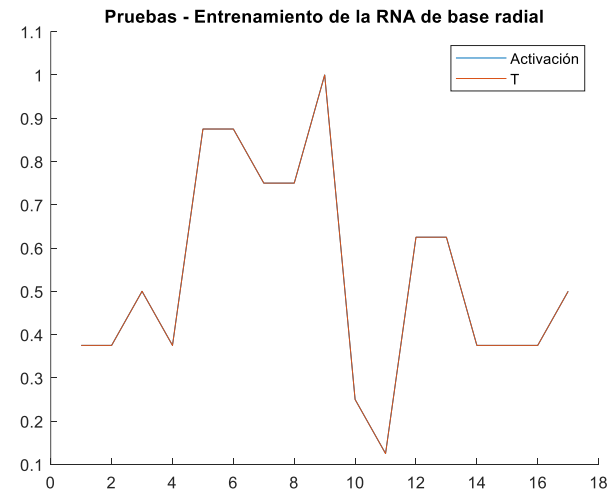
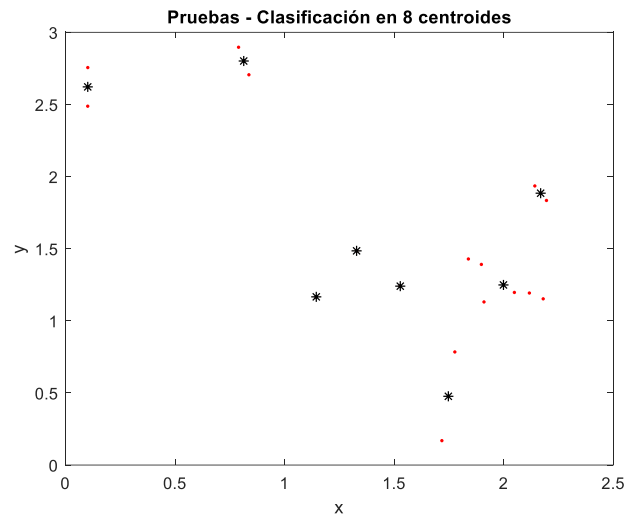
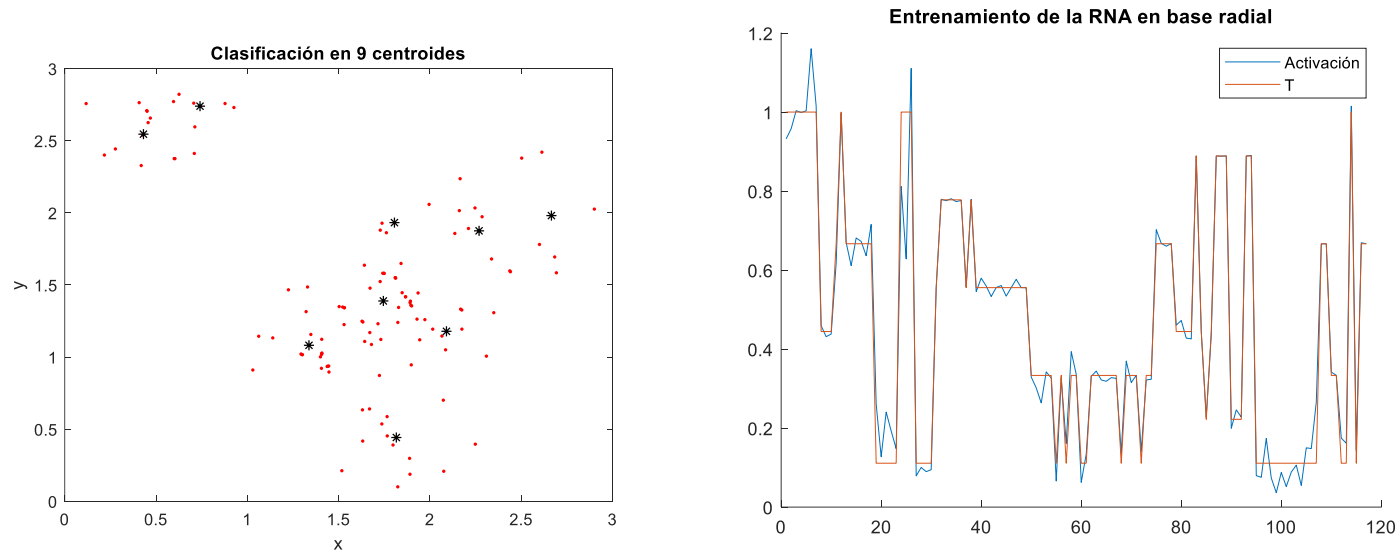


Figura 48 Pruebas RNA de base radial con 8 centros

- Entrenamiento de la red neuronal de base radial con 9 centroides utilizando Matlab

Núm. de centroides: 9

Entrenamientos: 100



Centroides: (1.7446,13886), (1.8054,1.9318), (1.3373, 1.0813), (2.2688, 1.8748), (0.431,2.5451), (1.8156, 0.442),
(0.7408, 2.7391), (2.6649, 1.9805) y (2.0902, 1.1783)

Figura 49 Entrenamiento RNA de base radial con 9 centros

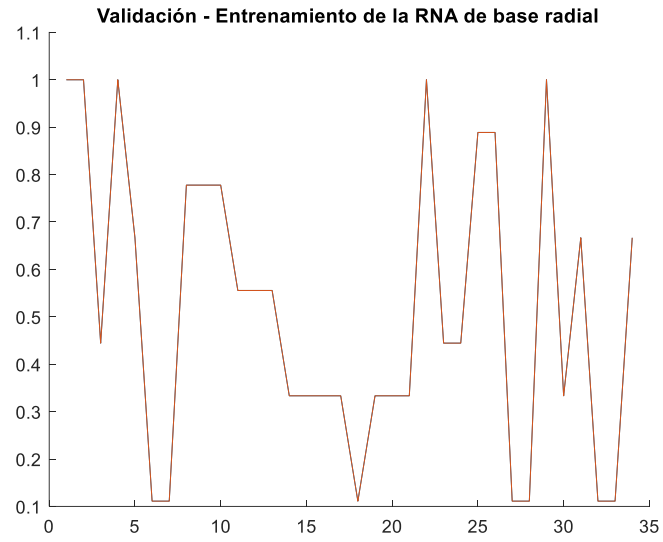
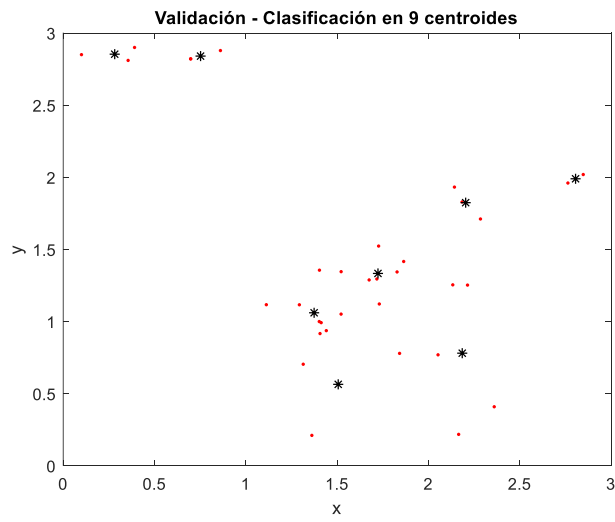


Figura 50 Validación RNA de base radial con 9 centros

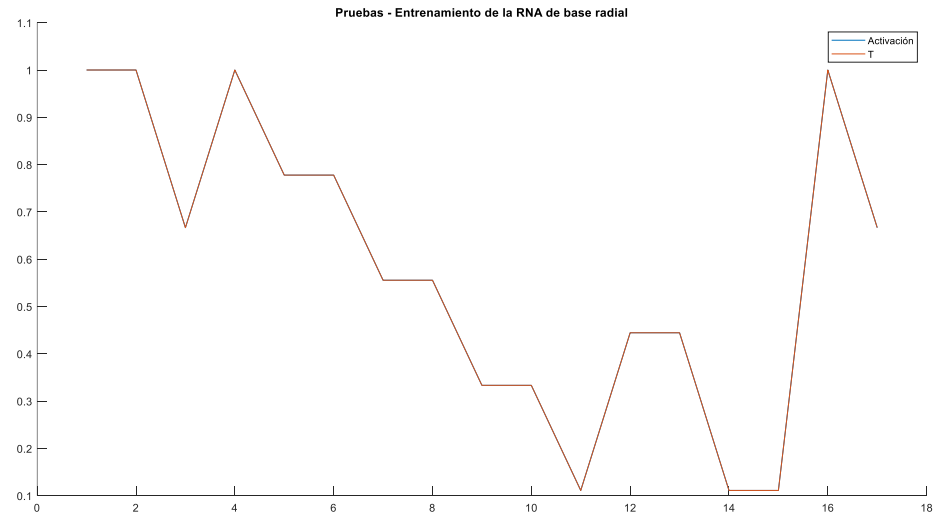
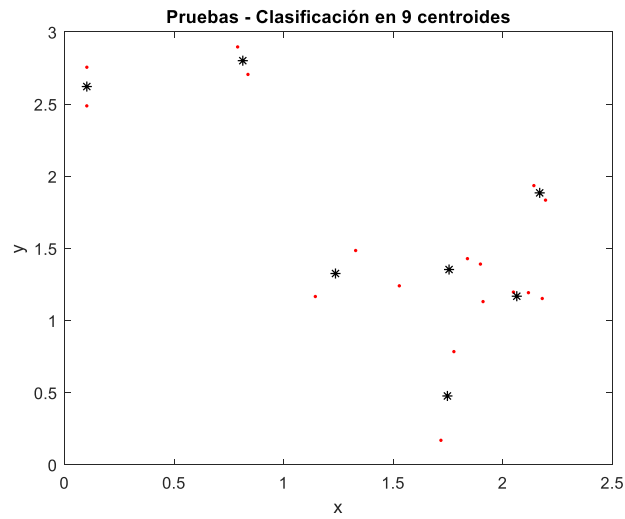
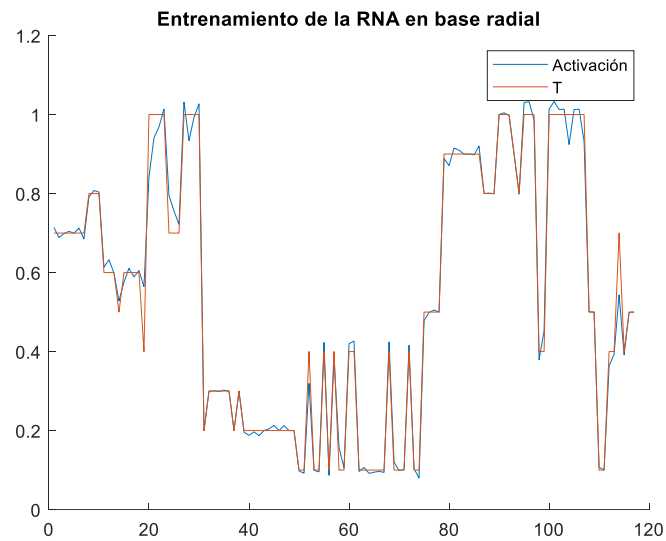
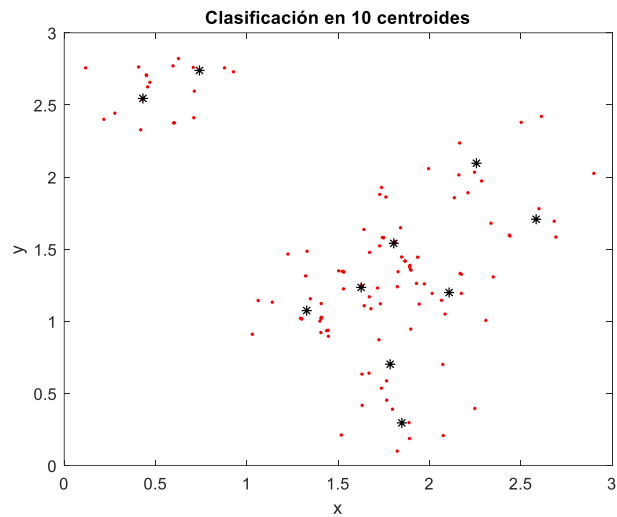


Figura 51 Pruebas RNA de base radial con 10 centros

- Entrenamiento de la red neuronal de base radial con 10 centroides utilizando Matlab

Núm. de centroides: 10

Entrenamientos: 100



Centroides: (1.3277, 1.0741), (0.431, 2.5451), (0.7408, 2.7391), (1.6261, 1.2348), (1.8487, 0.2956), (1.7849, 0.7022), (2.1077, 1.1995), (2.5842, 1.7074), (2.2571, 2.0958) y (1.8045, 1.5404)

Figura 52 Entrenamiento RNA de base radial con 10 centros

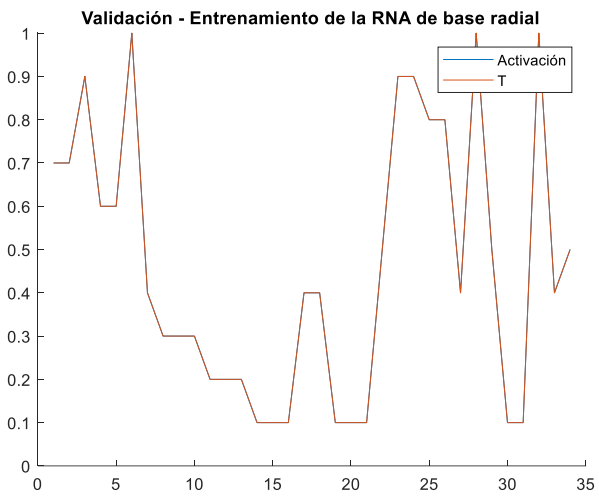
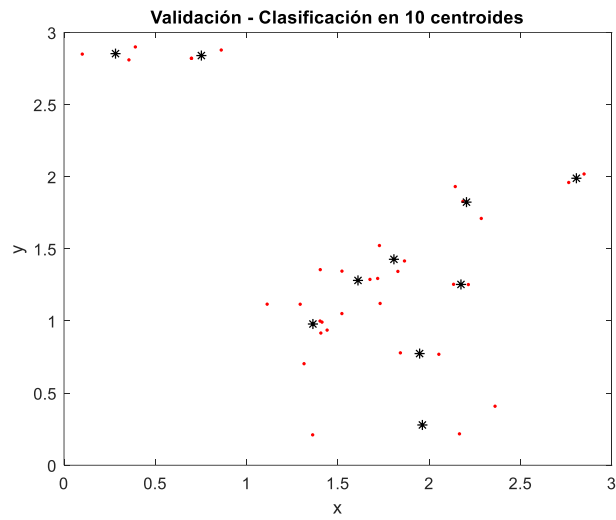


Figura 53 Validación RNA de base radial con 10 centros

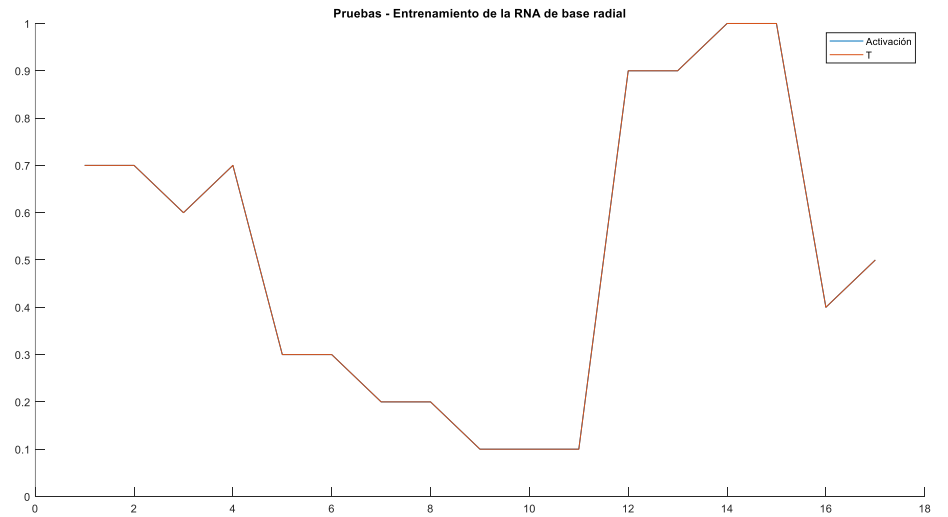
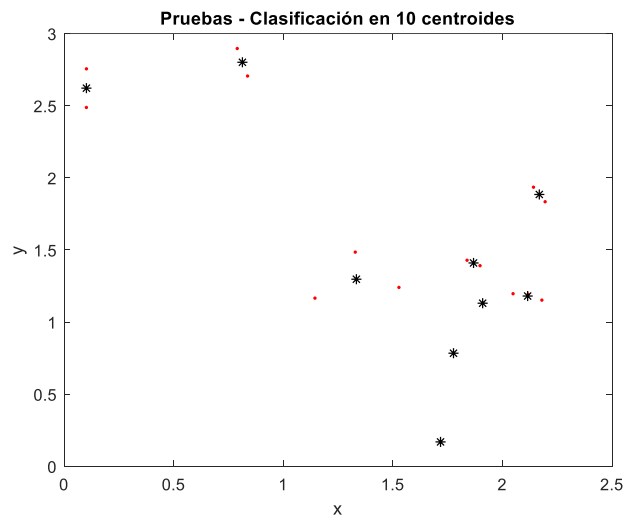
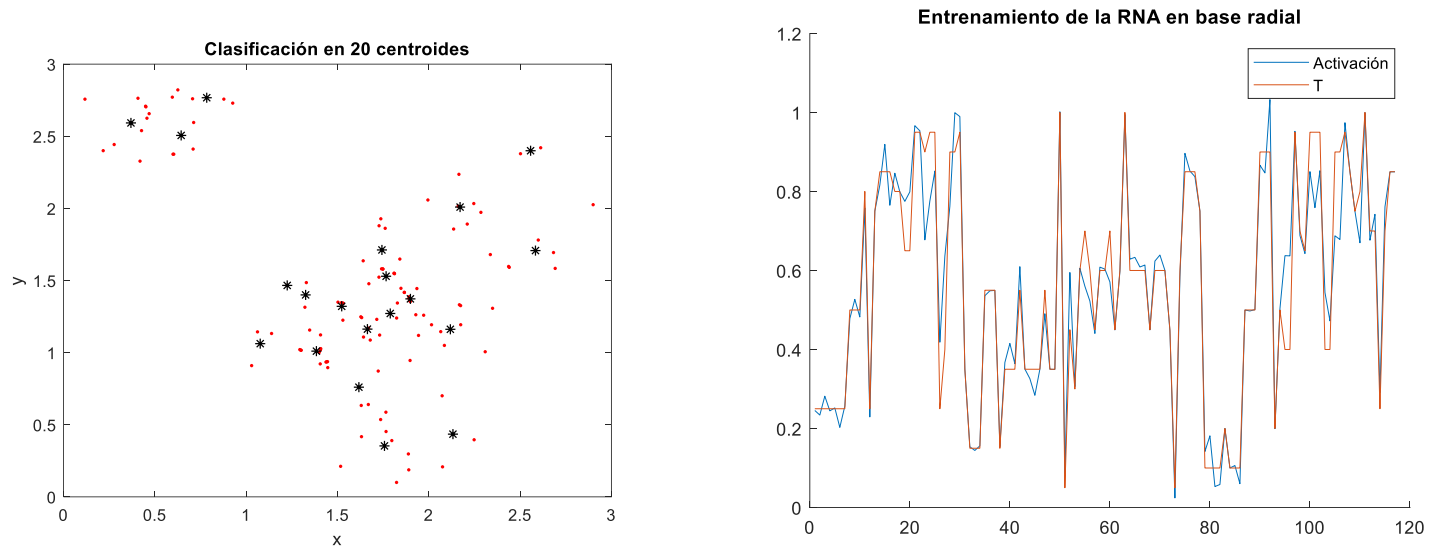


Figura 54 Pruebas RNA de base radial con 10 centros

- Entrenamiento de la red neuronal de base radial con 20 centroides utilizando Matlab

Núm. de centroides: 20

Entrenamientos: 100



Centroides: (1.3225, 1.4001), (2.1713, 2.009), (0.7841, 2.7671), (2.5575, 2.3996), (2.1182, 1.1618), (1.2248, 1.4659), (0.369, 2.5924), (1.7663, 1.5294), (1.5232, 1.321), (2.5842, 1.7074), (0.6441, 2.5058), (1.3846, 1.0103), (1.7891, 1.2714), (1.6639, 1.1629), (2.1322, 0.4348), (1.6172, 0.7605), (1.7573, 0.3533), (1.7434, 1.7118), (1.8994, 1.3726), (1.077, 1.0621).

Figura 55 Entrenamiento RNA de base radial con 20 centros

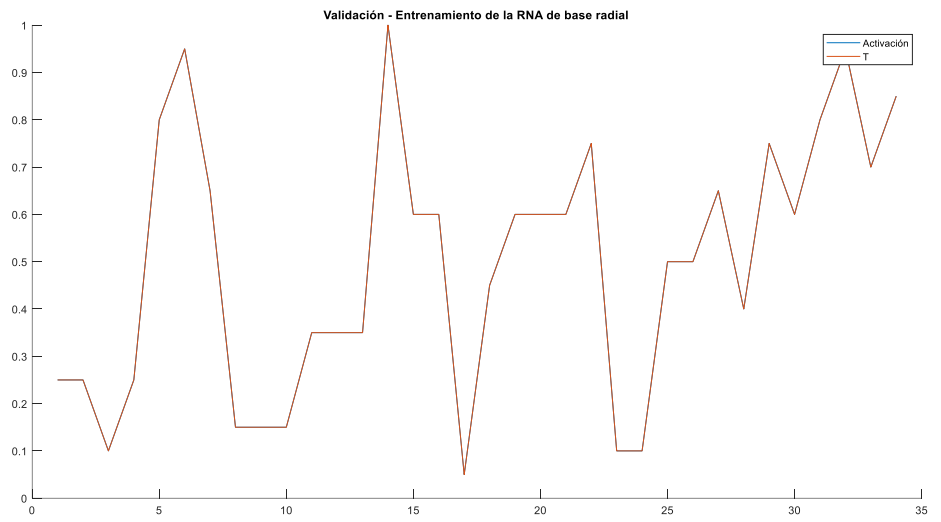
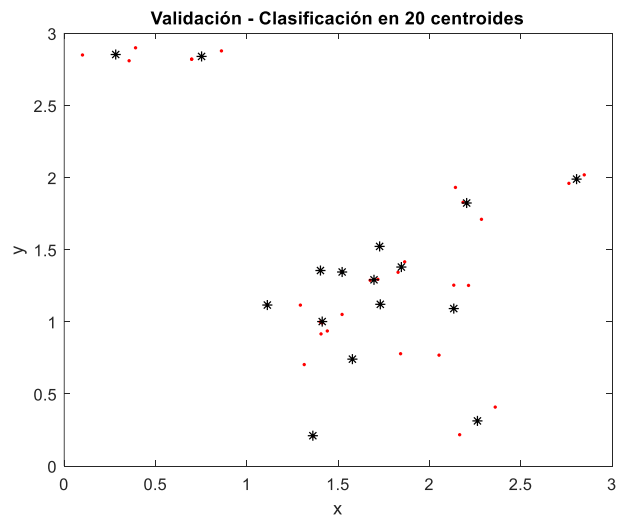


Figura 56 Validación RNA de base radial con 20 centros

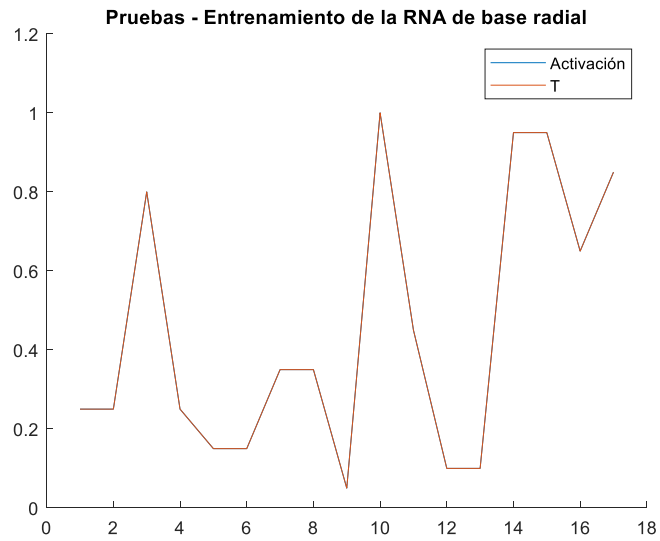
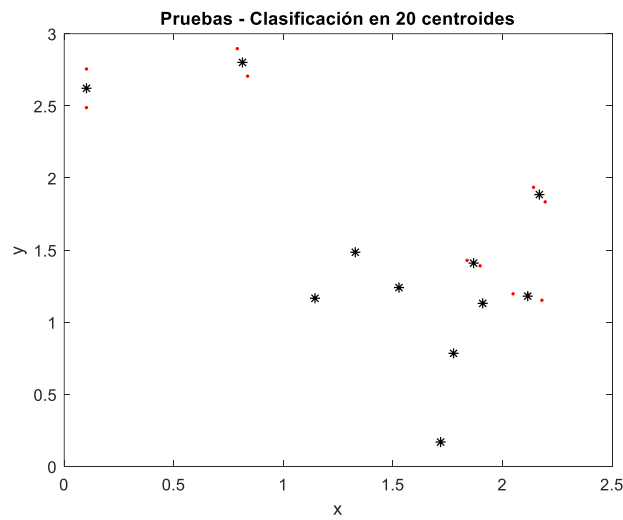


Figura 57 Pruebas RNA de base radial con 20 centros

Clasificación general propuesta.

Tabla 21 Clasificación general propuesta

	Municipio	Comunidad	Nodo	Latitud	Longitud	x	y	T	Salida de la red
1	Cortazar	La Gavia	Publico	20.3973	-100.8837	2.0731	0.7007	0.1429	0.143
2	Jaral de Progreso	Santiago Capitiro	Publico	20.3045	-101.0207	1.7660	0.4529	0.1429	0.1357
3	Jaral de Progreso	Providencia	Publico	20.3547	-101.0210	1.7652	0.5870	0.1429	0.159
4	Jaral de Progreso	Victoria de Cortázar	Publico	20.3356	-101.0339	1.7363	0.5360	0.1429	0.1215
5	Jaral de Progreso	DIF Municipal	Publico	20.3722	-101.0813	1.6301	0.6336	0.1429	0.1367
6	Jaral de Progreso	Jardín Municipal	Publico	20.3746	-101.0640	1.6689	0.6402	0.1429	0.1628
7	Salvatierra	El Sabino	Publico	20.2810	-101.0066	1.7976	0.3902	0.1429	0.1516
8	Salvatierra	Maravatío del Encinal	Publico	20.2051	-100.9652	1.8903	0.1873	0.1429	0.1432
9	Salvatierra	San Nicolás de los Agustinos	Publico	20.2463	-100.9661	1.8883	0.2975	0.1429	0.1408
10	Salvatierra	Centro Histórico	Publico	20.2129	-100.8827	2.0753	0.2080	0.1429	0.1429
11	Santiago Maravatio	Jardín Municipal	Publico	20.1724	-100.9950	1.8235	0.1000	0.1429	0.1427
12	Tarimoro	La Moncada	Publico	20.2831	-100.8055	2.2483	0.3958	0.1429	0.1428
13	Yuriria	Loma de Zempoala	Publico	20.2912	-101.0807	1.6315	0.4172	0.1429	0.144
14	Yuriria	Centro Histórico	Publico	20.2142	-101.1316	1.5172	0.2117	0.1429	0.1428
15	Cortazar	La Minilla	Publico	20.4229	-100.8929	2.0525	0.7692	0.1429	0.1429
16	Cortazar	Galeras	Publico	20.4266	-100.9868	1.8418	0.7791	0.1429	0.1429
17	Salvatierra	Urireo	Publico	20.2164	-100.8425	2.1654	0.2176	0.1429	0.1429
18	Tarimoro	Centro Histórico	Publico	20.2881	-100.7556	2.3602	0.4091	0.1429	0.1429
19	Yuriria	San Francisco de la Cruz	Publico	20.2138	-101.2011	1.3615	0.2106	0.1429	0.1429
20	Cortazar	El Diezmo	Publico	20.4285	-101.0159	1.7766	0.7841	0.1429	0.1429
21	Yuriria	Casacuaran	Publico	20.1983	-101.0423	1.7175	0.1691	0.1429	0.1429
1	Comonfort	Jalpilla	Escolar	20.7305	-100.7198	2.4406	1.5911	0.2857	0.271
2	Comonfort	Jalpilla	Escolar	20.7328	-100.7213	2.4370	1.5973	0.2857	0.3021
3	Comonfort	San Isidro de San Jeronimo	Escolar	20.7637	-100.7659	2.3371	1.6797	0.2857	0.2808
4	San Miguel de Allende	Corral de Piedras de Abajo	Escolar	20.8428	-100.8223	2.2106	1.8912	0.2857	0.3038

5	San Miguel de Allende	Alonso Yáñez	Publico	20.8299	-100.8556	2.1360	1.8567	0.2857	0.2803
6	San Miguel de Allende	Jalpa	Publico	20.7689	-100.6116	2.6829	1.6937	0.2857	0.2849
7	San Miguel de Allende	Charco de Sierra	Publico	20.7278	-100.6075	2.6922	1.5838	0.2857	0.2861
8	San Miguel de Allende	San José de Allendes	Publico	20.8565	-101.0334	1.7374	1.9278	0.2857	0.2754
9	San Miguel de Allende	San Isidro de Ensaye (La Cuevita)	Publico	20.8385	-101.0377	1.7278	1.8795	0.2857	0.2874
10	San Miguel de Allende	El Nuevo Cimatario	Publico	20.8318	-101.0228	1.7613	1.8618	0.2857	0.3014
11	San Miguel de Allende	Lagunillas	Publico	20.8014	-100.64883	2.5995	1.7805	0.2857	0.2866
12	Comonfort	San Pedro	Escolar	20.7753	-100.7892	2.2848	1.7109	0.2857	0.2857
13	San Miguel de Allende	Agustín Gonzalez	Escolar	20.8581	-100.8528	2.1423	1.9320	0.2857	0.2857
14	San Miguel de Allende	La Huerta Repetidor	Publico	20.8196	-100.8341	2.1843	1.8291	0.2857	0.2857
15	San Miguel de Allende	Agustín Gonzalez	Escolar	20.8592	-100.8531	2.1416	1.9350	0.2857	0.2857
16	San Miguel de Allende	La Huerta	Publico	20.8215	-100.8294	2.1948	1.8342	0.2857	0.2857
1	Cortazar	Tierra Fría	Publico	20.4614	-101.0398	1.7231	0.8721	0.4286	0.4089
11	Salamanca	Uruétaro	Escolar	20.5196	-101.1803	1.4082	1.0276	0.4286	0.4285
14	Salamanca	Uruétaro	Escolar	20.5161	-101.1812	1.4062	1.0183	0.4286	0.4301
16	Salamanca	Colonia el 4 de Altamira	Escolar	20.4848	-101.1675	1.4370	0.9346	0.4286	0.4274
17	Salamanca	Colonia el 4 de Altamira	Escolar	20.4853	-101.1656	1.4411	0.9360	0.4286	0.4261
21	Salamanca	Cuatro de Altamira	Escolar	20.4800	-101.1815	1.4055	0.9217	0.4286	0.4395
22	Salamanca	Colonia el Cuatro de Altamira	Escolar	20.4859	-101.1637	1.4454	0.9375	0.4286	0.4249
23	Salamanca	Uruetaro	Escolar	20.5181	-101.1808	1.4070	1.0235	0.4286	0.4292
26	Salamanca	El Circuito	Escolar	20.5097	-101.1839	1.4001	1.0012	0.4286	0.4337
28	Valle de Santiago	Alto de Altamira	Escolar	20.4703	-101.1631	1.4467	0.8960	0.4286	0.4185
33	Salamanca	El 4 de Altamira	Publico	20.4856	-101.1658	1.4406	0.9368	0.4286	0.4286
38	Salamanca	El Circuito	Escolar	20.5092	-101.1831	1.4020	0.9998	0.4286	0.4286
39	Salamanca	San Rafael de Uruetaro	Escolar	20.5063	-101.1783	1.4126	0.9921	0.4286	0.4286
40	Valle de Santiago	Rancho Cuatro de Altamira	Escolar	20.4780	-101.1812	1.4063	0.9165	0.4286	0.4286
41	Valle de Santiago	UTSOE	Escolar	20.3984	-101.2222	1.3143	0.7039	0.4286	0.4286
3	Salamanca	Congregación de Cárdenas	Publico	20.6270	-101.2189	1.3216	1.3146	0.4286	0.424
4	Salamanca	Cerro Gordo	Publico	20.5934	-101.1261	1.5296	1.2248	0.4286	0.3865

5	Salamanca	Loma de Flores	Publico	20.6836	-101.2621	1.2248	1.4659	0.4286	0.4309
7	Salamanca	Valencia	Publico	20.6022	-101.0823	1.6278	1.2483	0.4286	0.4334
9	Salamanca	Los Miranda	Escolar	20.6401	-101.1382	1.5026	1.3496	0.4286	0.4376
12	Salamanca	Valencia de Cerro Gordo	Escolar	20.6001	-101.0802	1.6325	1.2428	0.4286	0.4359
13	Salamanca	El Xoconostle	Escolar	20.6373	-101.1252	1.5316	1.3420	0.4286	0.4372
20	Salamanca	El Xoconostle	Escolar	20.6373	-101.1258	1.5303	1.3420	0.4286	0.4368
24	Salamanca	El Recuerdo de Anc�n (Xoconostle de Arriba)	Escolar	20.6390	-101.1296	1.5219	1.3465	0.4286	0.4387
25	Salamanca	San Jose de Mendoza	Escolar	20.6910	-101.2154	1.3295	1.4855	0.4286	0.4239
35	Salamanca	Uni�n de Liebres	Publico	20.6425	-101.1825	1.4032	1.3560	0.4286	0.4286
36	Salamanca	Recuerdo de Anc�n	Publico	20.6386	-101.1296	1.5218	1.3456	0.4286	0.4286
42	Salamanca	San Jos� de Mendoza	Publico	20.6908	-101.2159	1.3283	1.4850	0.4286	0.4286
44	Salamanca	Cerro Gordo	Escolar	20.5992	-101.1270	1.5276	1.2402	0.4286	0.4286
27	Juventino Rosas	La Trinidad	Escolar	20.5728	-101.0633	1.6704	1.1697	0.4286	0.465
30	Villagran	San Isidro Cerro Gordo	Publico	20.5418	-101.0593	1.6795	1.0870	0.4286	0.4755
31	Villagran	Parque Agro-Industrial XONOTLI	Escolar	20.5497	-101.0758	1.6425	1.1079	0.4286	0.4442
34	Salamanca	Valtierrilla	Publico	20.5285	-101.1297	1.5215	1.0514	0.4286	0.4286
2	Pueblo Nuevo	Congragaci�n de Panales	Escolar	20.5631	-101.3347	1.0620	1.1439	0.4286	0.4289
15	Salamanca	La Capilla	Escolar	20.5589	-101.3003	1.1391	1.1326	0.4286	0.4272
29	Valle de Santiago	Santa Barbara	Publico	20.4756	-101.3491	1.0299	0.9099	0.4286	0.4285
32	Pueblo Nuevo	Durazno de Fonseca	Escolar	20.5529	-101.3123	1.1123	1.1166	0.4286	0.4286
43	Salamanca	Loma Pelada	Escolar	20.5714	-101.2980	1.1444	1.1661	0.4286	0.4286
6	Salamanca	Santo Domingo	Publico	20.5153	-101.2279	1.3015	1.0162	0.4286	0.4298
8	Salamanca	Ecoparque	Publico	20.5678	-101.2073	1.3477	1.1564	0.4286	0.4424
10	Salamanca	La Luz	Escolar	20.5552	-101.1807	1.4072	1.1227	0.4286	0.4158
18	Salamanca	Santo Domingo	Escolar	20.5157	-101.2290	1.2989	1.0172	0.4286	0.4284
19	Salamanca	Santo Domingo	Escolar	20.5171	-101.2314	1.2936	1.0208	0.4286	0.4252
37	Salamanca	San Juan de Razos	Escolar	20.5527	-101.2316	1.2932	1.1161	0.4286	0.4286
1	Celaya	Pedro Mar�a / Patria Nueva	Publico	20.5279	-100.8782	2.0855	1.0499	0.5714	0.568
2	Celaya	Roque	Publico	20.5816	-100.8384	2.1747	1.1934	0.5714	0.5719

3	Celaya	San Elías	Publico	20.6335	-100.8415	2.1676	1.3319	0.5714	0.5714
4	Celaya	San Juan de la Vega	Publico	20.6243	-100.7605	2.3492	1.3073	0.5714	0.5693
5	Celaya	Parque Lineal	Publico	20.5115	-100.7785	2.3089	1.0060	0.5714	0.5718
6	Celaya	San isidro de Elguera	Publico	20.5637	-100.8871	2.0654	1.1454	0.5714	0.5656
7	Celaya	San Elías	Escolar	20.6312	-100.8386	2.1742	1.3259	0.5714	0.5833
8	Cortazar	Centro Cultural	Publico	20.4889	-100.9620	1.8975	0.9456	0.5714	0.5639
9	Juventino Rosas	Santiago de Cuenda	Publico	20.5990	-100.9949	1.8239	1.2397	0.5714	0.584
10	Juventino Rosas	COMUDE SUR	Publico	20.6379	-100.9930	1.8281	1.3438	0.5714	0.5764
11	Juventino Rosas	Valencia	Publico	20.6441	-100.9631	1.8950	1.3602	0.5714	0.5689
12	Juventino Rosas	Dulces Nombres	Publico	20.6505	-100.9649	1.8909	1.3773	0.5714	0.5711
13	Juventino Rosas	San Diego de los Dolores	Publico	20.7261	-101.0284	1.7486	1.5794	0.5714	0.5614
14	Juventino Rosas	Franco Tavera	Publico	20.6063	-100.9288	1.9719	1.2593	0.5714	0.5471
15	Juventino Rosas	San Antonio de Romerillo	Publico	20.6076	-100.9480	1.9289	1.2626	0.5714	0.5698
16	Juventino Rosas	San Antonio de Morales	Publico	20.5816	-100.9096	2.0150	1.1932	0.5714	0.5586
17	Juventino Rosas	El Naranjillo	Publico	20.7148	-101.0010	1.8101	1.5492	0.5714	0.5873
18	Juventino Rosas	Romero	Publico	20.7520	-100.98700	1.8415	1.6486	0.5714	0.5473
19	Juventino Rosas	Mesas de Acosta	Publico	20.7475	-101.07652	1.6408	1.6365	0.5714	0.5783
20	Juventino Rosas	La Peña	Publico	20.6755	-100.94534	1.9349	1.4442	0.5714	0.5515
21	Juventino Rosas	El Naranjillo	Escolar	20.7153	-101.0010	1.8102	1.5504	0.5714	0.5871
22	Juventino Rosas	El Naranjillo	Escolar	20.7140	-101.0001	1.8122	1.5470	0.5714	0.5883
23	Juventino Rosas	UPJR	Escolar	20.6543	-100.9641	1.8928	1.3875	0.5714	0.571
24	Juventino Rosas	San Julian Tierra Blanca	Escolar	20.5956	-101.0433	1.7152	1.2306	0.5714	0.5062
25	Juventino Rosas	Jaralillo	Escolar	20.6656	-100.9761	1.8659	1.4176	0.5714	0.5829
26	Juventino Rosas	San Jose del Sauz	Escolar	20.6761	-100.9843	1.8476	1.4458	0.5714	0.5896
27	Juventino Rosas	Jaralillo	Escolar	20.6654	-100.9759	1.8665	1.4172	0.5714	0.5828
28	Juventino Rosas	La Tinaja	Escolar	20.7051	-101.0380	1.7271	1.5231	0.5714	0.5706
29	Juventino Rosas	El Carmen de Arriba	Escolar	20.6880	-101.0627	1.6718	1.4775	0.5714	0.5551
30	Juventino Rosas	San Diego de los Dolores	Escolar	20.7267	-101.0317	1.7414	1.5810	0.5714	0.5595
31	Juventino Rosas	San Diego de los Dolores	Escolar	20.7263	-101.0285	1.7485	1.5798	0.5714	0.5611

32	Juventino Rosas	Los Dulces Nombres	Escolar	20.6418	-100.9612	1.8993	1.3541	0.5714	0.5671
33	Villagran	Suchitlan	Publico	20.5548	-101.0363	1.7309	1.1217	0.5714	0.5249
34	Villagran	Los Ángeles	Publico	20.5539	-100.9413	1.9438	1.1191	0.5714	0.6196
35	Celaya	Gasca	Publico	20.6045	-100.8569	2.1332	1.2543	0.5714	0.5714
36	Celaya	San Cayetano	Publico	20.6038	-100.8209	2.2139	1.2526	0.5714	0.5714
37	Juventino Rosas	Jaralillo	Publico	20.6650	-100.9768	1.8643	1.4162	0.5714	0.5714
38	Juventino Rosas	Cerrito de Gasca	Publico	20.6172	-101.06116	1.6752	1.2883	0.5714	0.5714
39	Juventino Rosas	Emiliano Zapata	Escolar	20.6196	-101.0423	1.7174	1.2949	0.5714	0.5714
40	Juventino Rosas	La Tinaja	Escolar	20.7051	-101.0380	1.7271	1.5231	0.5714	0.5714
41	Valle de Santiago	COMUDE	Publico	20.6379	-100.9930	1.8281	1.3438	0.5714	0.5714
42	Villagran	Col.18 de marzo	Publico	20.5548	-101.0363	1.7309	1.1217	0.5714	0.5714
43	Celaya	Yustis	Publico	20.5814	-100.8642	2.1168	1.1927	0.5714	0.5714
44	Celaya	Ejido de Silva	Publico	20.5663	-100.8360	2.1799	1.1523	0.5714	0.5714
45	Juventino Rosas	San Antonio de las Maravillas	Publico	20.5829	-100.8947	2.0484	1.1967	0.5714	0.5714
46	Juventino Rosas	Los Dulces Nombres	Escolar	20.6556	-100.9619	1.8978	1.3908	0.5714	0.5714
47	Juventino Rosas	San Antonio Nuevo Valencia	Escolar	20.6697	-100.9883	1.8385	1.4287	0.5714	0.5714
48	Villagran	San Salvador Torrecillas	Publico	20.5582	-100.9566	1.9097	1.1308	0.5714	0.5714
1	León	San Francisco del Duran	Publico	21.0061	-101.6217	0.4187	2.3274	0.7143	0.7142
2	León	San José de los Romeros	Publico	21.0374	-101.49197	0.7095	2.4112	0.7143	0.7145
3	León	Alfaro	Publico	21.1470	-101.60730	0.4510	2.7039	0.7143	0.7187
4	León	San José del Potrero	Escolar	21.1293	-101.5994	0.4687	2.6566	0.7143	0.7262
5	León	León Urbano	Escolar	21.0853	-101.6180	0.4270	2.5391	0.7143	0.7168
6	León	Los Sauces	Escolar	21.0241	-101.5393	0.6035	2.3757	0.7143	0.7184
7	León	Los Naranjos	Escolar	21.1693	-101.6267	0.4075	2.7635	0.7143	0.7144
8	León	Alfaro	Escolar	21.1487	-101.6081	0.4492	2.7085	0.7143	0.7087
9	León	León Urbano	Escolar	21.1175	-101.6048	0.4567	2.6251	0.7143	0.7022
10	León	Fraccionamiento Paseo de las Torres	Escolar	21.1669	-101.7562	0.1172	2.7571	0.7143	0.7143
11	León	Los Sauces	Escolar	21.0258	-101.5412	0.5993	2.3757	0.7143	0.7101
12	León	San Pedro del Monte	Escolar	21.0333	-101.7116	0.2172	2.4001	0.7143	0.7143

13	León	Los Arcos	Escolar	21.0492	-101.6848	0.2772	2.4427	0.7143	0.7142
14	León	Lagunillas	Escolar	21.2017	-101.7639	0.1000	2.8502	0.7143	0.7143
15	León	Ibarrilla	Escolar	21.1868	-101.6500	0.3554	2.8103	0.7143	0.7143
16	León	Mesa de Ibarrilla	Escolar	21.2204	-101.6342	0.3906	2.9000	0.7143	0.7143
17	León	León Primaria	Escolar	21.1663	-101.7631	0.1017	2.7556	0.7143	0.7143
18	León	La Mora	Escolar	21.0661	-101.7632	0.1015	2.4879	0.7143	0.7143
1	San Miguel de Allende	Don Juan	Escolar	20.8961	-100.8063	2.2465	2.0335	0.8571	0.8689
2	San Miguel de Allende	Pantoja	Escolar	20.8732	-100.7890	2.2854	1.9725	0.8571	0.8449
3	San Miguel de Allende	Tlaxcalilla	Escolar	20.8891	-100.8446	2.1606	2.0150	0.8571	0.8447
4	San Miguel de Allende	Los Rodríguez	Publico	21.0408	-100.6429	2.6127	2.4203	0.8571	0.8571
5	San Miguel de Allende	Cruz del Palmar	Publico	20.9719	-100.8427	2.1650	2.2361	0.8571	0.857
6	San Miguel de Allende	Don Francisco	Publico	20.9053	-100.9185	1.9951	2.0582	0.8571	0.8599
7	San Miguel de Allende	Moral de Puerto de Nieto	Publico	20.8931	-100.5148	2.9000	2.0255	0.8571	0.8571
8	San Miguel de Allende	La Biznaga	Publico	21.0254	-100.6922	2.5022	2.3790	0.8571	0.8571
9	San Miguel de Allende	Guadalupe de Tambula	Publico	20.8686	-100.5757	2.7636	1.9602	0.8571	0.8571
10	San Miguel de Allende	Puerto de Nieto	Publico	20.8907	-100.5382	2.8476	2.0191	0.8571	0.8571
1	León	Vaquerías	Publico	21.1566	-101.39523	0.9264	2.7297	1	1.0007
2	León	Derramadero	Publico	21.1670	-101.41692	0.8778	2.7573	1	0.999
3	León	San José de Otates Sur	Publico	21.1679	-101.49336	0.7064	2.7598	1	1.0011
4	León	Sauz Seco	Publico	21.1721	-101.54277	0.5956	2.7709	1	0.9975
5	León	Cuesta Blanca	Publico	21.1064	-101.49065	0.7125	2.5954	1	0.9997
6	León	Los Alisos	Publico	21.1910	-101.52930	0.6258	2.8214	1	1.0008
7	León	Nuevo Valle de Moreno	Publico	21.2124	-101.42442	0.8609	2.8786	1	1
8	León	San Rafael Cerro Verde	Publico	21.1908	-101.49721	0.6978	2.8209	1	1
9	León	San José de Otates Norte	Publico	21.1908	-101.49721	0.6978	2.8209	1	1
10	León	Mesas de Obispo	Publico	21.1477	-101.43512	0.8370	2.7057	1	1
11	León	Las Canelas	Publico	21.2191	-101.45616	0.7898	2.8965	1	1

La figura 28, muestra la clasificación general en siete grupos, cada grupo tiene un centroide que puede ser el punto de partida a cada punto de su grupo.

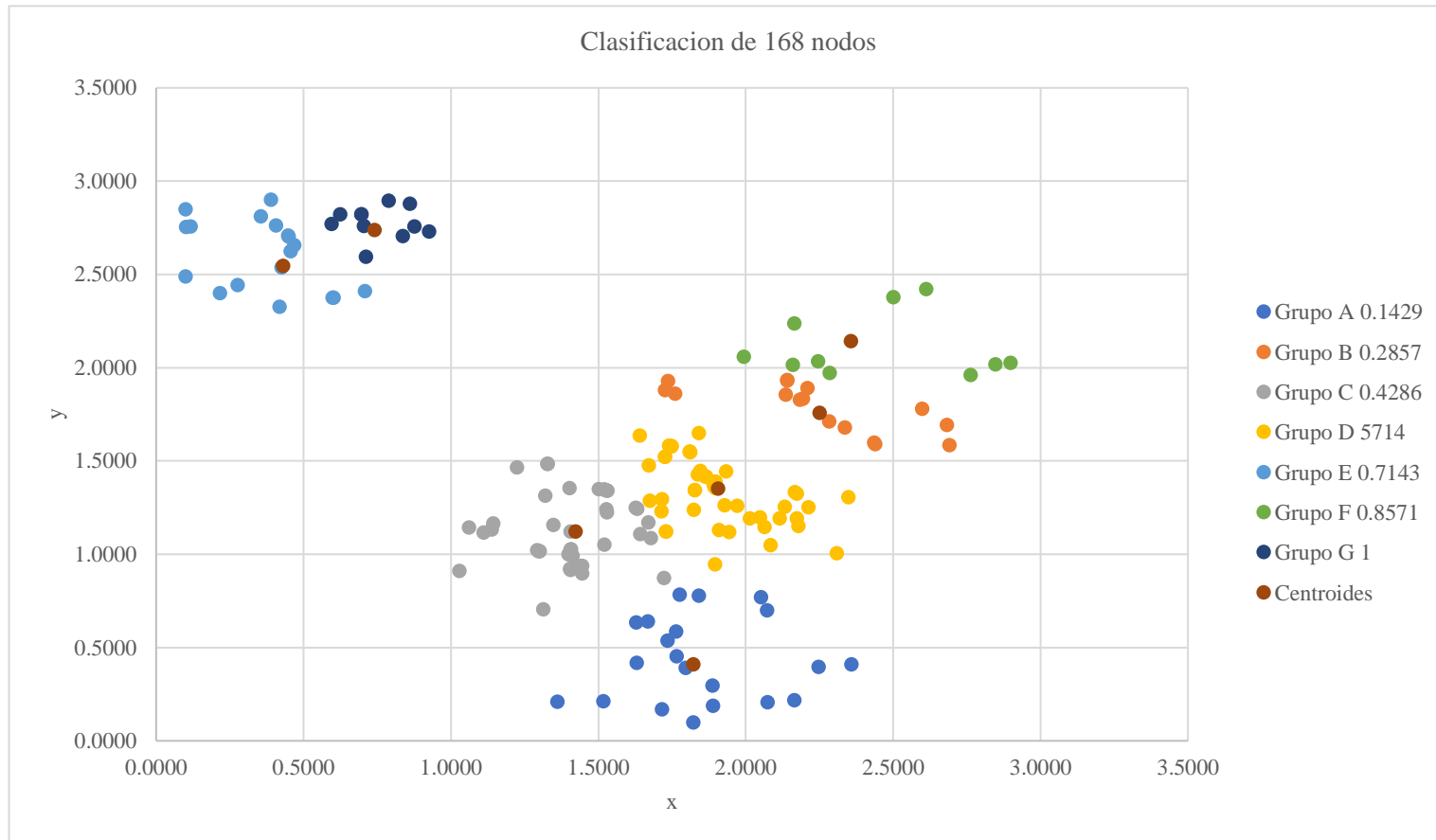


Figura 58 Clasificación grafica general

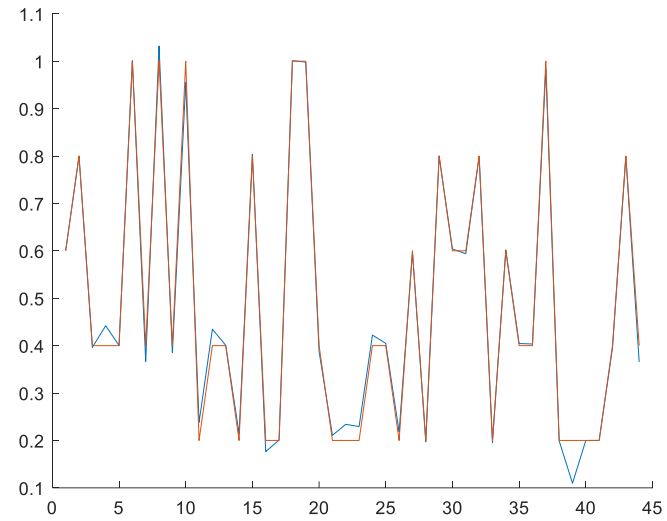
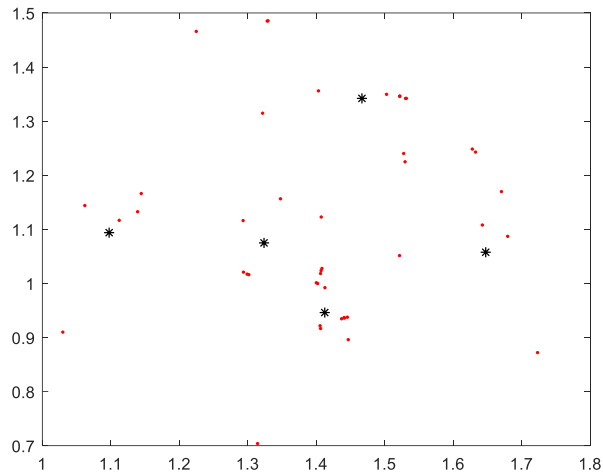
Centroides: (1.8223, 0.4113), (2.2511, 1.7585), (1.424, 1.1231), (1.9066, 1.3535), (0.431, 2.5451), (2.3584, 2.1425) y (0.7408, 2.7391)

A su vez cada uno de los grupos se puede volver a clasificar.

Por ejemplo, el grupo C, se vuelve a clasificar, devolviendo los siguientes resultados:

Núm. de centroides: 5

Entrenamientos: 100



Centroides: (1.4124, 0.9461), (1.4667, 1.3421), (1.6474, 1.0576), (1.0975, 1.0938), (1.3237, 1.0749)

Figura 59 Subclasificación de grupos

Tabla 22 Subclasificación de grupos

1	Cortazar	Tierra Fría	Publico	20.4614	-101.0398	1.7231	0.8721	0.4286	0.4089	1	0.6	0.6001
11	Salamanca	Uruétaro	Escolar	20.5196	-101.1803	1.4082	1.0276	0.4286	0.4285	1	0.2	0.2383
14	Salamanca	Uruétaro	Escolar	20.5161	-101.1812	1.4062	1.0183	0.4286	0.4301	1	0.2	0.2144
16	Salamanca	Colonia el 4 de Altamira	Escolar	20.4848	-101.1675	1.4370	0.9346	0.4286	0.4274	1	0.2	0.1764
17	Salamanca	Colonia el 4 de Altamira	Escolar	20.4853	-101.1656	1.4411	0.9360	0.4286	0.4261	1	0.2	0.2008
21	Salamanca	Cuatro de Altamira	Escolar	20.4800	-101.1815	1.4055	0.9217	0.4286	0.4395	1	0.2	0.2108
22	Salamanca	Colonia el Cuatro de Altamira	Escolar	20.4859	-101.1637	1.4454	0.9375	0.4286	0.4249	1	0.2	0.234
23	Salamanca	Uruetaro	Escolar	20.5181	-101.1808	1.4070	1.0235	0.4286	0.4292	1	0.2	0.2293
26	Salamanca	El Circuito	Escolar	20.5097	-101.1839	1.4001	1.0012	0.4286	0.4337	1	0.2	0.2181
28	Valle de Santiago	Alto de Altamira	Escolar	20.4703	-101.1631	1.4467	0.8960	0.4286	0.4185	1	0.2	0.1972
33	Salamanca	El 4 de Altamira	Publico	20.4856	-101.1658	1.4406	0.9368	0.4286	0.4286	1	0.2	0.1953
38	Salamanca	El Circuito	Escolar	20.5092	-101.1831	1.4020	0.9998	0.4286	0.4286	1	0.2	0.1993
39	Salamanca	San Rafael de Uruetaro	Escolar	20.5063	-101.1783	1.4126	0.9921	0.4286	0.4286	1	0.2	0.1097
40	Valle de Santiago	Rancho Cuatro de Altamira	Escolar	20.4780	-101.1812	1.4063	0.9165	0.4286	0.4286	1	0.2	0.2
41	Valle de Santiago	UTSOE	Escolar	20.3984	-101.2222	1.3143	0.7039	0.4286	0.4286	1	0.2	0.2001
3	Salamanca	Congregación de Cárdenas	Publico	20.6270	-101.2189	1.3216	1.3146	0.4286	0.424	1	0.4	0.3964
4	Salamanca	Cerro Gordo	Publico	20.5934	-101.1261	1.5296	1.2248	0.4286	0.3865	1	0.4	0.4421
5	Salamanca	Loma de Flores	Publico	20.6836	-101.2621	1.2248	1.4659	0.4286	0.4309	1	0.4	0.4003
7	Salamanca	Valencia	Publico	20.6022	-101.0823	1.6278	1.2483	0.4286	0.4334	1	0.4	0.3667
9	Salamanca	Los Miranda	Escolar	20.6401	-101.1382	1.5026	1.3496	0.4286	0.4376	1	0.4	0.385
12	Salamanca	Valencia de Cerro Gordo	Escolar	20.6001	-101.0802	1.6325	1.2428	0.4286	0.4359	1	0.4	0.4346
13	Salamanca	El Xoconostle	Escolar	20.6373	-101.1252	1.5316	1.3420	0.4286	0.4372	1	0.4	0.4008
20	Salamanca	El Xoconostle	Escolar	20.6373	-101.1258	1.5303	1.3420	0.4286	0.4368	1	0.4	0.3876
24	Salamanca	El Recuerdo de Ancón (Xoconostle de Arriba)	Escolar	20.6390	-101.1296	1.5219	1.3465	0.4286	0.4387	1	0.4	0.4223
25	Salamanca	San Jose de Mendoza	Escolar	20.6910	-101.2154	1.3295	1.4855	0.4286	0.4239	1	0.4	0.4047
35	Salamanca	Unión de Liebres	Publico	20.6425	-101.1825	1.4032	1.3560	0.4286	0.4286	1	0.4	0.4045
36	Salamanca	Recuerdo de Ancón	Publico	20.6386	-101.1296	1.5218	1.3456	0.4286	0.4286	1	0.4	0.4035
42	Salamanca	San José de Mendoza	Publico	20.6908	-101.2159	1.3283	1.4850	0.4286	0.4286	1	0.4	0.3951

44	Salamanca	Cerro Gordo	Escolar	20.5992	-101.1270	1.5276	1.2402	0.4286	0.4286	1	0.4	0.3656
27	Juventino Rosas	La Trinidad	Escolar	20.5728	-101.0633	1.6704	1.1697	0.4286	0.465	1	0.6	0.597
30	Villagran	San Isidro Cerro Gordo	Publico	20.5418	-101.0593	1.6795	1.0870	0.4286	0.4755	1	0.6	0.604
31	Villagran	Parque Agro-Industrial XONOTLI	Escolar	20.5497	-101.0758	1.6425	1.1079	0.4286	0.4442	1	0.6	0.5939
34	Salamanca	Valtierrilla	Publico	20.5285	-101.1297	1.5215	1.0514	0.4286	0.4286	1	0.6	0.602
2	Pueblo Nuevo	Congragación de Panales	Escolar	20.5631	-101.3347	1.0620	1.1439	0.4286	0.4289	1	0.8	0.7998
15	Salamanca	La Capilla	Escolar	20.5589	-101.3003	1.1391	1.1326	0.4286	0.4272	1	0.8	0.8037
29	Valle de Santiago	Santa Barbara	Publico	20.4756	-101.3491	1.0299	0.9099	0.4286	0.4285	1	0.8	0.8001
32	Pueblo Nuevo	Durazno de Fonseca	Escolar	20.5529	-101.3123	1.1123	1.1166	0.4286	0.4286	1	0.8	0.7991
43	Salamanca	Loma Pelada	Escolar	20.5714	-101.2980	1.1444	1.1661	0.4286	0.4286	1	0.8	0.799
6	Salamanca	Santo Domingo	Publico	20.5153	-101.2279	1.3015	1.0162	0.4286	0.4298	1	1	1.0011
8	Salamanca	Ecoparque	Publico	20.5678	-101.2073	1.3477	1.1564	0.4286	0.4424	1	1	1.032
10	Salamanca	La Luz	Escolar	20.5552	-101.1807	1.4072	1.1227	0.4286	0.4158	1	1	0.9549
18	Salamanca	Santo Domingo	Escolar	20.5157	-101.2290	1.2989	1.0172	0.4286	0.4284	1	1	1.0011
19	Salamanca	Santo Domingo	Escolar	20.5171	-101.2314	1.2936	1.0208	0.4286	0.4252	1	1	0.9982
37	Salamanca	San Juan de Razos	Escolar	20.5527	-101.2316	1.2932	1.1161	0.4286	0.4286	1	1	0.9836

Tabla 23. Porcentaje de rendimiento en Matlab

Número de centroides	% Rendimiento de aprendizaje	% Rendimiento de validación	% Rendimiento de prueba
3	100	100	100
4	100	100	100
5	100	100	100
6	99.145299	100	100
7	100	100	100
8	94.017094	100	100
9	90.598290	100	100
10	89.743590	100	100
20	55.555556	100	100

El rendimiento de las tres etapas se mejoró al utilizar la función newrbe de Matlab, como se observó en las gráficas anteriores en donde se utilizó la función deseada arrojada por el algoritmo de k-medias. Utilizando Microsoft Excel se hace con la finalidad de conocer los procesos que se siguen en la capa de entrada, en la oculta y en la capa de salida. Adquiriendo conocimiento sobre el comportamiento de las redes neuronales. Esta es una propuesta para clasificar por grupos, de tal forma que cada grupo es una ruta, es de suma importancia formar grupos y un punto central en común a todos ellos cuando se desconoce el lugar y el número de nodos aumenta, se observó también que da mejores resultados al subclasificar el grupo después de una primera clasificación, teniendo así rutas con menos puntos a visitar.

CONCLUSIONES

Este proyecto de investigación muestra una aplicación del algoritmo elbow y una red neuronal de base radial, se pudo entender el proceso que se lleva a cabo en la capa de entrada usando la metodología explicada en Microsoft Excel. Es interesante aplicar estos procedimientos con datos reales porque se obtienen resultados importantes que pueden ser llevados a la vida real y mejorar el rendimiento.

Como se observó en los resultados del método elbow, el número adecuado de centroides es 7, ya puesto en práctica en Matlab, se observaron pocas diferencias entre la salida deseada de la red y la salida de la función de activación, también, se experimentó agregando más centroides y los resultados fueron menos rendimiento en la etapa de aprendizaje. En base al código en Matlab donde el usuario elige el número de centroides y el número de entrenamientos deseados, se obtiene la clasificación, y en conjunto con Excel se pueden interpretar los datos arrojados por la red neuronal de base radial.

Utilizando la red neuronal de base radial es una forma innovadora de clasificar los nodos públicos y escolares del proyecto de brecha Digital en Guanajuato, agrupándolos en grupos y asignando una ruta para cada uno de los grupos es como se propone hacer el recorrido, evitando así la manera empírica de realizarlo cuando el número de puntos va creciendo considerablemente.

Es necesario comprender desde un enfoque básico el funcionamiento de las redes neuronales, debido a su amplio campo de aplicación y complejidad que puede llegar a ser. Los estudiantes pueden entender este enfoque, con la finalidad de ampliar su interés, adentrándose en las redes neuronales y el campo de la programación, aprovechando las herramientas que ofrece; clasificación y predicción de patrones.

Esta propuesta deja una gran área de oportunidad, porque puede trasladarse a desarrollar la metodología presentada en esta investigación en un software libre con técnicas de programación, modificarse para una distribución de puntos de otra índole o una propuesta para aprender el funcionamiento de una red neuronal de base radial.

REFERENCIAS

- Caicedo Bravo, E., y López Sotelo, J. (2009). *Una aproximación práctica a las redes neuronales artificiales* (1st ed., p. 13). Cali: Programa Editorial Universidad del Valle.
- Matich, D. J.: *Redes neuronales, Conceptos básicos y aplicaciones*. Ed. Universidad Tecnológica Nacional, México, marzo, 2001.
- Domínguez, C. (2018, 23 enero). Busca Cemer reducir brecha digital en 12 municipios. *La Vanguardia*. Recuperado de <https://periodicocorreo.com.mx/busca-cemer-reducir-brecha-digital-12-municipios/>
- Redacción AM. (2016, 25 octubre). Dan acceso digital a 2 mil guanajuatenses. *am*. Recuperado de <https://www.am.com.mx/noticias/Dan-acceso-digital-a-2-mil-guanajuatenses-20161025-0178.html>
- Programa de Reducción de Brecha Digital [Internet]. Salamanca: Universidad de Guanajuato, Centro Interinstitucional de Energía; c2019 [citado 21 enero 2020]. Disponible en: <http://www.guanajuatoconectado.org/>
- Centro Mexicano de Energías Renovables S.C. [Internet]. Salamanca Gto.: Centro Mexicano de Energías Renovables S.C.; c2017 [citado 21 de enero 2020]. Disponible en: <https://www.cemersc.org/brecha-digital>.
- Victoria-Rodríguez, M.T. y Vega-Corona, A., (2017). *Reducción de Brecha Digital*. Universidad de Guanajuato, 2-12.
- Widrow, B., and S.D. Stearns, 1985. *Adaptive Signal Processing*, Englewood Cliffs, NJ: Prentice-Hall
- Haykin, S., 2002. *Adaptive Filter Theory*, 4th ed., Englewood Cliffs, NJ: Prentice Hall.
- Arbib, M.A., 1987. *Brains, Machines, and Mathematics*, 2d ed., New York: Springer-Verlag.
- Haykin, S. (2009). *Neural networks and learning machines* (3rd ed., pp. 1-11). New York [etc.]: Pearson education.
- Ponce Cruz, P. (2010). *Inteligencia artificial* (1st ed.). México, D.F.: Alfaomega.
- Berzal, F. (2018). *Redes Neuronales & Deep Learning* [Ebook] (p. 194). Madrid. Retrieved from <https://deep-learning.ikor.org/>
- Ocampo Vega, R. (2014). *Red de Base Radial: Nueva Aproximación con Menos Parámetros* (Maestría). Instituto Tecnológico y de Estudios Superiores de Monterrey.
- Hagan, M., Demuth, H., Beale, M., & De Jesús, O. (2016). *Neural Network Design* (2nd ed.). [S. l.: s. n.].
- Isasi Viñuela, P., y Galván León, I. (2008). *Redes de neuronas artificiales*. Madrid, [etc.]: Pearson-Prentice Hall.

- Moody, J. and Darken, C. (1989). Fast learning in networks of locally-tuned processing units. *Neural Computation*, 1: 281-294.
- Niranjan, M. and Fallside, F. (1990). Neural networks and radial basis functions in classifying static speech patterns. *Computer Speech and Language*, 4: 275-289.
- Kadirkamanathan, V., Niranjan, M., and Fallside, F. (1991). Sequential adaptation of radial basis function neural network. In *Advances in Neural Information Processing Systems*, pages 721-727. Morgan Kaufmann.
- Lowe, D. and Webb, A. (1990). Exploiting prior knowledge in network optimization: an illustration from medical prognosis. *Network*, 1:299-323.
- Moya García, R., 2016. *Machine Learning*. [ebook] pp.60-63. Available at: <<https://docplayer.es/150852042-Machine-learning-en-python-con-ejemplos-ricardo-moya-garcia-phd-version-septiembre-2016.html>> [Accessed 30 May 2020].
- M.J.D. Powell, (1987). Radial basis functions for multivariable interpolation: a review. *Algorithms for Approximation*, Oxford, pp. 143-167.
- Barrero Vicedo, E., Fernández Díaz, B., & Llanes Santiago, O. (2014). Propuesta de procedimiento para configurar una red neuronal artificial de Base Radial con aplicaciones en el diagnóstico de fallos. *Ingeniería Electrónica, Automática y Comunicaciones*, 35(3), 60-75.
- Bholowalia, P., & Kumar, A. (2014). EBK-means: A clustering technique based on elbow method and k-means in WSN. *International Journal of Computer Applications*, 105(9).
- Cevallos Ampuero, Juan (2008). Redes Neuronales de Base Radial aplicadas a la mejora de la calidad. *Industrial Data*, 11(2),63-72.[fecha de Consulta 8 de Junio de 2020]. ISSN: 1560-9146. Disponible en: <https://www.redalyc.org/articulo.oa?id=816/81619829009>
- Valencia, J. S. B., & Guzmán, L. J. M. (2013). Estimación de distancia con sensores ópticos reflexivos usando redes neuronales con funciones de base radial para aplicaciones embebidas. *Ingeniería y Universidad*, 17(1), 27-40
- Ou, Y. Y., Oyang, Y. J., & Chen, C. Y. (2005, July). A novel radial basis function network classifier with centers set by hierarchical clustering. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005*. (Vol. 3, pp. 1383-1388). IEEE.
- Lin, C. J., Chen, C. H., & Lee, C. Y. (2004, July). A self-adaptive quantum radial basis function network for classification applications. In *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No. 04CH37541)* (Vol. 4, pp. 3263-3268). IEEE.
- Bors, A. G., & Pitas, I. (1996). Median radial basis function neural network. *IEEE transactions on Neural Networks*, 7(6), 1351-1364.
- Bugmann, G. (1998). Normalized Gaussian radial basis function networks. *Neurocomputing*, 20(1-3), 97-110.
- Casent, D., & Chen, X. W. (2003). Radial basis function neural networks for nonlinear Fisher discrimination and Neyman–Pearson classification. *Neural networks*, 16(5-6), 529-535.

- Dash, P. K., Pradhan, A. K., & Panda, G. (2001). Application of minimal radial basis function neural network to distance protection. *IEEE Transactions on Power Delivery*, 16(1), 68-74.
- Kala, R., Vazirani, H., Khanwalkar, N., & Bhattacharya, M. (2010). Evolutionary Radial Basis Function Network for Classificatory Problems. *IJCSA*, 7(4), 34-49.
- Karayiannis, N. B., & Mi, G. W. (1997). Growing radial basis neural networks: Merging supervised and unsupervised learning with network growth techniques. *IEEE Transactions on Neural networks*, 8(6), 1492-1506.
- Longoni, M. G., Porcel, E., López, M. V., & Dapozo, G. N. (2010). Modelos de Redes Neuronales Perceptrón Multicapa y de Base Radial para la predicción del rendimiento académico de alumnos universitarios. In *XVI Congreso Argentino de Ciencias de la Computación*.
- Karayannis, N. B., & Randolph-Gips, M. M. (2003). On the construction and training of reformulated radial basis function neural networks. *IEEE Transactions on Neural Networks*, 14(4), 835-846.
- Leonard, J. A., Kramer, M. A., & Ungar, L. H. (1992). Using radial basis functions to approximate a function and its error bounds. *IEEE transactions on neural networks*, 3(4), 624-627.
- Liu, H., Wang, X., & Qiang, W. (2007). A fast method for implicit surface reconstruction based on radial basis functions network from 3D scattered points. *International Journal of Neural Systems*, 17(06), 459-465.
- Yingwei, L., Sundararajan, N., & Saratchandran, P. (1998). Performance evaluation of a sequential minimal radial basis function (RBF) neural network learning algorithm. *IEEE Transactions on neural networks*, 9(2), 308-318.
- Botia, J., Sarmiento, H., & Isaza, C. (2009). Redes neuronales artificiales de base radial como clasificador difuso: Una aplicación en diagnóstico médico. *Universidad de Antioquia*.
- Renals, S., & Rohwer, R. (1989, June). Phoneme classification experiments using radial basis functions. In *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN'89)* (Vol. 1, pp. 461-467).
- Cortes, D., Garcia, J., & Garcia, J. (2002). Uso de Redes Neuronales para Analizar Modelos Mecánicos de Tejidos Biológicos. In *VI Congreso Colombiano de Elementos Finitos y Modelamiento Numérico*. Bogotá, Colombia mayo.
- Gallegos-Funes, F. J., Ponomaryov, V., Michtchenko, A., & Moreno-Escobar, J. A. (2008). La función de base radial de rango tipo M para la clasificación de imágenes de mamografía. *Científica*, 12(2), 55-65.
- Munnoli, S. P., & Bapat, A. U. (2013). Clustering algorithms for radial basis function neural network. *ITSI Transactions on Electrical and Electronics Engineering (ITSI-TEEE)*, 1, 113-116.
- Zemouri, R., Racoceanu, D., & Zerhouni, N. (2003). Recurrent radial basis function network for time-series prediction. *Engineering Applications of Artificial Intelligence*, 16(5-6), 453-463.
- Syakur, M. A., Khotimah, B. K., Rochman, E. M. S., & Satoto, B. D. (2018, April). Integration k-means clustering method and elbow method for identification of the best customer profile cluster. In

IOP Conference Series: Materials Science and Engineering (Vol. 336, No. 1, p. 012017). IOP Publishing.

Kodinariya, T. M., & Makwana, P. R. (2013). Review on determining number of Cluster in K-Means Clustering. *International Journal*, 1(6), 90-95.

Saini, Y., Bachchas, V., Kumar, Y., & Kumar, S. (2020, May). Abusive Text Examination Using Latent Dirichlet Allocation, Self Organizing Maps and K Means Clustering. In *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)* (pp. 1233-1238). IEEE.

Jiménez-Carrión, M., Sánchez-Candela, L., Keewong-Zapata, R., & Bazán, J. (2020). Optimización de las rutas para la intervención de pozos de petróleo. *Información tecnológica*, 31(4), 71-84.

Ma, T., Liu, H., & Zhang, Y. (2020). A Method for Establishing Tropospheric Atmospheric Refractivity Profile Model Based on Multiquadric RBF and k-means Clustering. *Engineering Letters*, 28(3).

Han, Y., Fan, C., Geng, Z., Ma, B., Cong, D., Chen, K., & Yu, B. (2020). Energy efficient building envelope using novel RBF neural network integrated affinity propagation. *Energy*, 209, 118414.

Li, W., Xing, L., Huang, K., Ai, Q., & Wang, L. (2020, February). Generalized Load Modeling Method Based on Clustering and Neural Network. In *2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC)* (pp. 619-622). IEEE.

Gutiérrez, S. S. M., Álvarez, J. C., Olguín, E. A. G., Munguía, M. G., & Pérez, S. L. (2020). Modelo inteligente para monitoreo del desarrollo de plantas de maíz. *Tecnología Educativa Revista CONAIC*, 7(1), 13-18.

Rustam, Z., & Fijri, A. L. (2020, March). Breast cancer clustering using modified spherical K-Means. In *Journal of Physics: Conference Series* (Vol. 1490, No. 1, p. 012028). IOP Publishing.

Zhou, Y., Wang, A., Zhou, P., Wang, H., & Chai, T. (2020). Dynamic performance enhancement for nonlinear stochastic systems using RBF driven nonlinear compensation with extended Kalman filter. *Automatica*, 112, 108693.

APÉNDICE

Apéndice A. Código en Matlab

Para clasificar los 117 puntos en n centroides utilizando el algoritmo k-medias

```
clear all
close
tic;
Num_centrosT=input('Elige el numero de centros (valor entre 3 y 117): ');
Entrenamientos=input('Elige el numero de entrenamientos: ');

Rengs=117;%Numero de datos por columna
Finales=zeros(2,Num_centrosT-2);
Finales(1,:)=3:Num_centrosT;
Ti=1.0/Num_centrosT;

X      1.934865382      1.63251407      1.737377611
=[2.085471663      0.418733158      1.531595802      1.727783426
2.174654989      0.926377603      1.406243745      1.761295823
2.167618506      0.877756607      1.139108471      2.502222418
2.349233282      0.706406264      1.436976519      2.599531659
2.308948915      0.595647231      1.441101122      1.810181332
2.065411299      0.712481086      1.298936826      1.812160693
2.174220112      0.709522132      1.293646575      1.892810126
2.440555124      0.625842013      1.530295655      1.670375233
2.437035762      0.450994724      1.405526423      1.715207871
2.337083637      0.468748449      1.445427471      1.865890368
1.723114107      0.426986846      1.407028317      1.847553819
1.897490653      0.603515359      1.521911952      1.866473192
2.073064231      0.407507065      1.329521652      1.727066104
1.765992042      0.449201419      1.400101674      1.671765045
1.765185055      0.456688469      1.797554219      1.741352024
1.736303869      0.117193317      1.890297256      1.748464722
1.630140182      0.599256259      1.888304446      1.899252576
1.668947314      0.2171701      2.075276721      1.823521283
1.82387322      0.277223419      2.246452217      2.24829708
1.828071796      1.06201875      2.210608523      1.446682785
1.894984509      1.321642316      2.285389363      1.02987599
1.890911464      1.529620924      2.160642548      1.730917228
1.74859922      1.224750018      2.61271918      1.679460567
1.971941974      1.301463145      2.165002522      1.943816218
1.928902641      1.627822335      1.995091306      1.642531923
2.014981306      1.347694562      2.136024946      1.631494128
1.810149949      1.502633917      2.9      1.517238149];
1.841478997      1.407230063      2.682942782
1.640808108      1.408238798      2.692178306
```

Y=[1.04988673	2.327427577	1.342033784	1.879542683
1.193352469	2.729692215	1.018251271	1.861801175
1.331909903	2.757293084	1.132555743	2.379003529
1.307317608	2.759831402	0.934620369	1.780494838
1.00604864	2.770946564	0.936036483	1.550376263
1.145447727	2.595428546	1.017155786	1.547001636
1.325884737	2.411226809	1.020816308	1.387501742
1.591082869	2.821419015	1.341953627	1.169695345
1.59725499	2.703881527	0.921688306	1.230561541
1.679728953	2.656588653	0.937532755	1.417595507
0.872054835	2.539077884	1.02354166	1.445784197
0.945572543	2.375663637	1.34649588	1.417221439
0.700696988	2.763491924	1.485483492	1.523136102
0.452909046	2.708477219	1.001177741	1.477526533
0.586977665	2.625086789	0.390204573	1.58096166
0.536005565	2.75710605	0.187262033	1.579794034
0.633647983	2.375663637	0.297462445	1.354126866
0.640202188	2.400111648	0.208038834	0.1
1.239720861	2.442701954	2.033498356	0.395794217
1.343818623	1.143938096	1.891218946	0.895984495
1.360154704	1.314643996	1.972498564	0.909913182
1.377316406	1.224824942	2.014955274	1.121667694
1.579427981	1.465874317	2.420257878	1.086978237
1.259257895	1.016188553	2.236141642	1.119148079
1.262624506	1.248313736	2.058200199	1.107944745
1.193154748	1.156378526	1.856679117	0.417244341
1.549208636	1.349568581	2.025536053	0.211672637];
1.648555735	1.122749819	1.693737797	
1.636478685	1.027629688	1.583841983	
1.44415433	1.242798906	1.927797446	

```

for Num_centros=3:1:Num_centrosT

j=randperm(Rengs);
centros=zeros(2,Num_centros);
for i=1:1:Num_centros
    centros(1,i)=X(j(i),1);
    centros(2,i)=Y(j(i),1);
end
rrr=Num_centrosT*Num_centrosT;
Distancias=zeros(Rengs,rrr);%Matriz con 117 renglones y 60 columnas para
llenar con iteraciones
NuevoCentro=zeros(2, Num_centros);

cuenta=1;%contador que indica que inicia con la primera iteración
j=0;%variable auxiliar de posicionamiento

while centros ~= NuevoCentro%realizar iteraciones hasta que los vectores
solucion se repitan
%%while centros(1,1) ~= NuevoCentro(1,1)%realizar iteraciones hasta que
los vectores solucion se repitan

    if cuenta==1%condicion de si es la primera iteración, tomar valores
propuestos
        a=1;

```

```

else%condicion de si es la segunda iteración o mayor, tomar los
valores previos como pivote
for i=1:1:Num_centros
    centros(1,i)=NuevoCentro(1,i);
    centros(2,i)=NuevoCentro(2,i);
end
end

for i=1:1:Num_centros
    Distancias(:,i+Num_centros*2*j)=sqrt(((centros(1,i)-
X).^2)+((centros(2,i)-Y).^2));%Generamos los vectores con las distancias
resultantes
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%*****
for i=1:1:Rengs%encontramos las x mas cercanas para cada renglón
aux=min(Distancias(i,1+Num_centros*2*j:Num_centros+Num_centros*2*j));%Enc
uentra la distancia mas corta de las tres y lo guarda en la variable
"aux"
    k=1;
    while(aux~=Distancias(i,k+Num_centros*2*j))
        k=k+1;
    end
    Distancias(i,Num_centros+k+Num_centros*2*j)=1;
end

Resultados=zeros(3,Num_centros);%R1=elementos con 1 en la
col...,R2=sumade las X en la col,r3=suma de las y en la col 1, mismo para
C2 y C3

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%*****
for i=1:1:Rengs%aquí calculo las sumas de las X's, Y's y el numero
de 1's de cada columna
    k=1;
    while(Distancias(i,Num_centros+k+Num_centros*2*j)~=1)
        if k==Num_centros
            break;
        else
            k=k+1;
        end
    end
    end

    T(i,1)=Ti*k

    Resultados(1,k) = Resultados(1,k) + 1;%incremento la cuenta de 1's
    Resultados(2,k) = Resultados(2,k) + X(i,1);%sumo el valor de la X en
dicho renglón para auxiliar de primera columna
    Resultados(3,k) = Resultados(3,k) + Y(i,1);%sumo el valor de la Y en
dicho renglón para auxiliar de primera columna
    end

for i=1:1:Num_centros
    NuevoCentro(1,i)=Resultados(2,i)/Resultados(1,i);%Calculo la X
promedio de cada columna

```

```

        NuevoCentro(2,i)=Resultados(3,i)/Resultados(1,i);%Y media, cada
columna
    end

```

```

    cuenta=cuenta+1;%incremento el contador
    j=j+1;%incremento variable posicionamiento

```

```

end

```

```

soluciones=zeros(Num_centros, Rengs,2);

```

```

%%%%%%Ajuste manual*****
for i=1:1:Rengs%Agrupamos los distintos grupos en tres vectores
auxiliares

```

```

    k=1;
    while(Distancias(i,(cuenta-1)*Num_centros*2-k)~=1)
        if k==Num_centros
            break;
        else
            k=k+1;
        end
    end
end

```

```

    if k~=Num_centros
        soluciones(Num_centros-k,i,1)=X(i,1);
        soluciones(Num_centros-k,i,2)=Y(i,1);
    else
        soluciones(8,i,1)=X(i,1);
        soluciones(8,i,2)=Y(i,1);
    end
end

```

```

end

```

```

if Num_centros == Num_centrosT
    plot(X,Y,'r. ');
    hold on
    for i=1:1:Num_centros
        plot(NuevoCentro(1,i),NuevoCentro(2,i),'k*');%primer centro en
asterisco azul
        hold on
    end
else
    i=1;
end

```

```

Nuevas_Distancias=zeros(1,Num_centros);
for j=1:1:Num_centros
    for i=1:1:Rengs
        if (soluciones(j,i,1)+soluciones(j,i,2))~=0

```

```

Nuevas_Distancias(1,j)=Nuevas_Distancias(1,j)+sqrt(((soluciones(j,i,1)-
NuevoCentro(1,j)).^2)+((soluciones(j,i,2)-
NuevoCentro(2,j)).^2));%Calculamos distancias totales centros
        else

```



```

        a=1;
    end
end
end

SumaTotal=0;
for i=1:1:Num_centros
    SumaTotal=SumaTotal+Nuevas_Distancias(1,i);
end
Finales(2,Num_centros-2)=SumaTotal;
end

```

El siguiente código se utilizó para ver gráficamente el entrenamiento de la red neuronal de base radial utilizando la función newrbe, recibiendo como parámetros de entrada los 117 puntos de la tabla 7 y la función deseada T del código anterior perteneciente a la clasificación arrojada.

```

figure(2)
hold on

```

```

X2=[2.085471663 1.04988673 0.625842013 2.821419015
2.174654989 1.193352469 0.450994724 2.703881527
2.167618506 1.331909903 0.468748449 2.656588653
2.349233282 1.307317608 0.426986846 2.539077884
2.308948915 1.00604864 0.603515359 2.375663637
2.065411299 1.145447727 0.407507065 2.763491924
2.174220112 1.325884737 0.449201419 2.708477219
2.440555124 1.591082869 0.456688469 2.625086789
2.437035762 1.59725499 0.117193317 2.75710605
2.337083637 1.679728953 0.599256259 2.375663637
1.723114107 0.872054835 0.2171701 2.400111648
1.897490653 0.945572543 0.277223419 2.442701954
2.073064231 0.700696988 1.06201875 1.143938096
1.765992042 0.452909046 1.321642316 1.314643996
1.765185055 0.586977665 1.529620924 1.224824942
1.736303869 0.536005565 1.224750018 1.465874317
1.630140182 0.633647983 1.301463145 1.016188553
1.668947314 0.640202188 1.627822335 1.248313736
1.82387322 1.239720861 1.347694562 1.156378526
1.828071796 1.343818623 1.502633917 1.349568581
1.894984509 1.360154704 1.407230063 1.122749819
1.890911464 1.377316406 1.408238798 1.027629688
1.74859922 1.579427981 1.63251407 1.242798906
1.971941974 1.259257895 1.531595802 1.342033784
1.928902641 1.262624506 1.406243745 1.018251271
2.014981306 1.193154748 1.139108471 1.132555743
1.810149949 1.549208636 1.436976519 0.934620369
1.841478997 1.648555735 1.441101122 0.936036483
1.640808108 1.636478685 1.298936826 1.017155786
1.934865382 1.44415433 1.293646575 1.020816308
0.418733158 2.327427577 1.530295655 1.341953627
0.926377603 2.729692215 1.405526423 0.921688306
0.877756607 2.757293084 1.445427471 0.937532755
0.706406264 2.759831402 1.407028317 1.02354166
0.595647231 2.770946564 1.521911952 1.34649588
0.712481086 2.595428546 1.329521652 1.485483492
0.709522132 2.411226809 1.400101674 1.001177741

```

```

1.797554219 0.390204573
1.890297256 0.187262033
1.888304446 0.297462445
2.075276721 0.208038834
2.246452217 2.033498356
2.210608523 1.891218946
2.285389363 1.972498564
2.160642548 2.014955274
2.61271918 2.420257878
2.165002522 2.236141642
1.995091306 2.058200199
2.136024946 1.856679117
2.9 2.025536053
2.682942782 1.693737797
2.692178306 1.583841983
1.737377611 1.927797446
1.727783426 1.879542683
1.761295823 1.861801175
2.502222418 2.379003529
2.599531659 1.780494838
1.810181332 1.550376263
1.812160693 1.547001636
1.892810126 1.387501742
1.670375233 1.169695345
1.715207871 1.230561541
1.865890368 1.417595507
1.847553819 1.445784197
1.866473192 1.417221439
1.727066104 1.523136102
1.671765045 1.477526533
1.741352024 1.58096166
1.748464722 1.579794034
1.899252576 1.354126866
1.823521283 0.1
2.24829708 0.395794217
1.446682785 0.895984495
1.02987599 0.909913182
1.730917228 1.121667694
1.679460567 1.086978237
1.943816218 1.119148079
1.642531923 1.107944745
1.631494128 0.417244341
1.517238149 0.211672637
]
X1=X2'

```

```

x=1:1:117;
net=newrbe(X1,T')
outputs = net(X1)

```

```

plot(x,outputs)
hold on
plot(x,T')
hold off

```