

# Reconocimiento de patrones en curvas mediante el SOM

## Trabajo final de Redes Neuronales y Aprendizaje Estadístico

**Docente:** Pedro J. Zufiria.

**Autor:** Carlos Eduardo Meneses Marín.

**Fecha de entrega:** 20 - 03 - 2020

*Nota para la ejecución de este LiveScript:* por favor, asegúrese de que su *current folder* de Matlab sea la carpeta en la que se encuentre este fichero. Además, verifique que la carpeta *Data* también está en su *current folder* ya que los datos que contienen son llamados mediante el *path* relativo '`./Data/`'. Además, es necesario disponer de los siguientes *toolboxes*:

- *Statistics and Machine Learning Toolbox.*
- *Deep Learning Toolbox.*

## Introducción

Poder reconocer y establecer patrones a partir de movimientos es una capacidad tecnológica muy desarrollada en los últimos años. Es usada en el mundo de las videoconsolas como la *Wii*, la *PlayStation5* o el mismo teléfono móvil. Este trabajo tiene por objetivo reconocer y distinguir entre diferentes movimientos realizados con el teléfono móvil, como un círculo a derechas, un círculo a izquierdas o una línea recta. Los medios para alcanzar dicho propósito son un *dataset* de casi 400 movimientos de los tipos descritos, registrados mediante el acelerómetro del móvil. Como herramienta de resolución, uso un mapa auto-organizado de Kohonen.

La motivación de este trabajo se encuentra en las grandes posibilidades del reconocimiento de movimientos para el desarrollo de tecnologías para la accesibilidad y movilidad de personas con discapacidad, más allá de su actual expansión en el mundo de los videojuegos. Por ejemplo, en el desarrollo de dispositivos inalámbricos que una persona con parálisis motriz pueda usar en cualquier extremidad del cuerpo, que conserve su movilidad, y reconozca patrones que le permitan controlar un móvil, un ordenador, o quizás un exoesqueleto.

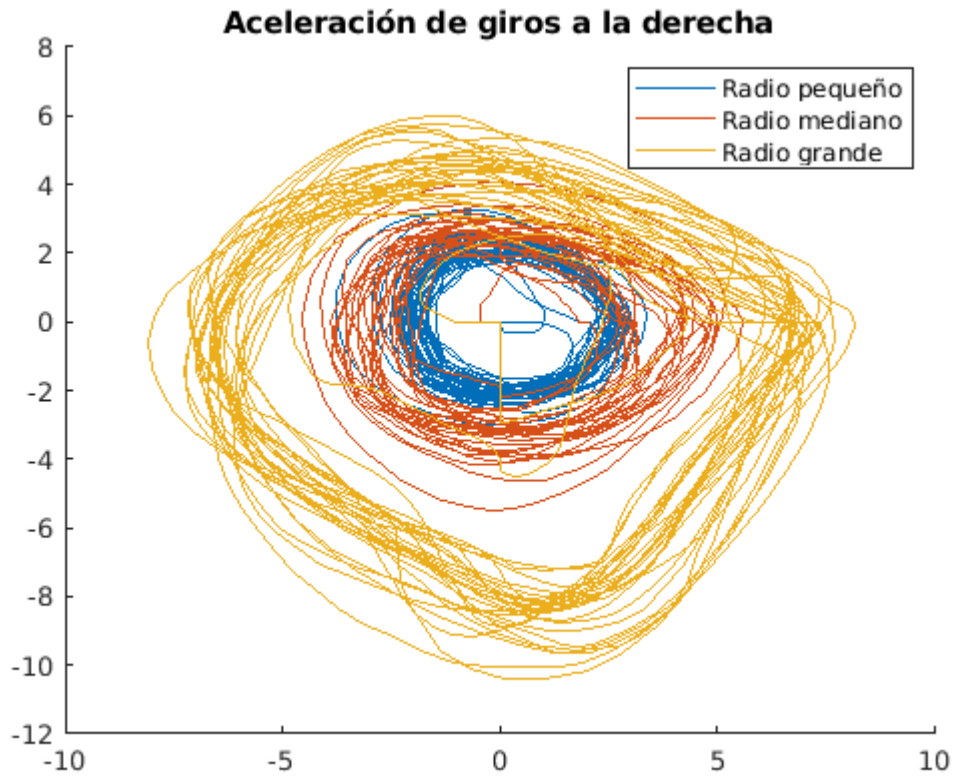
## La base de datos

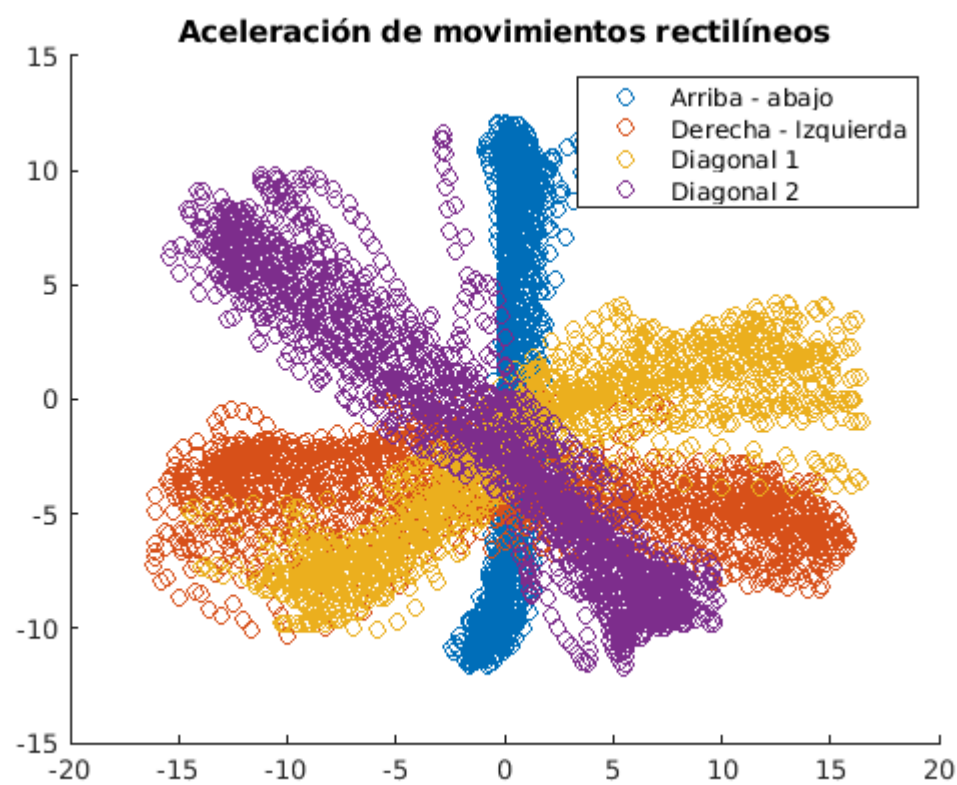
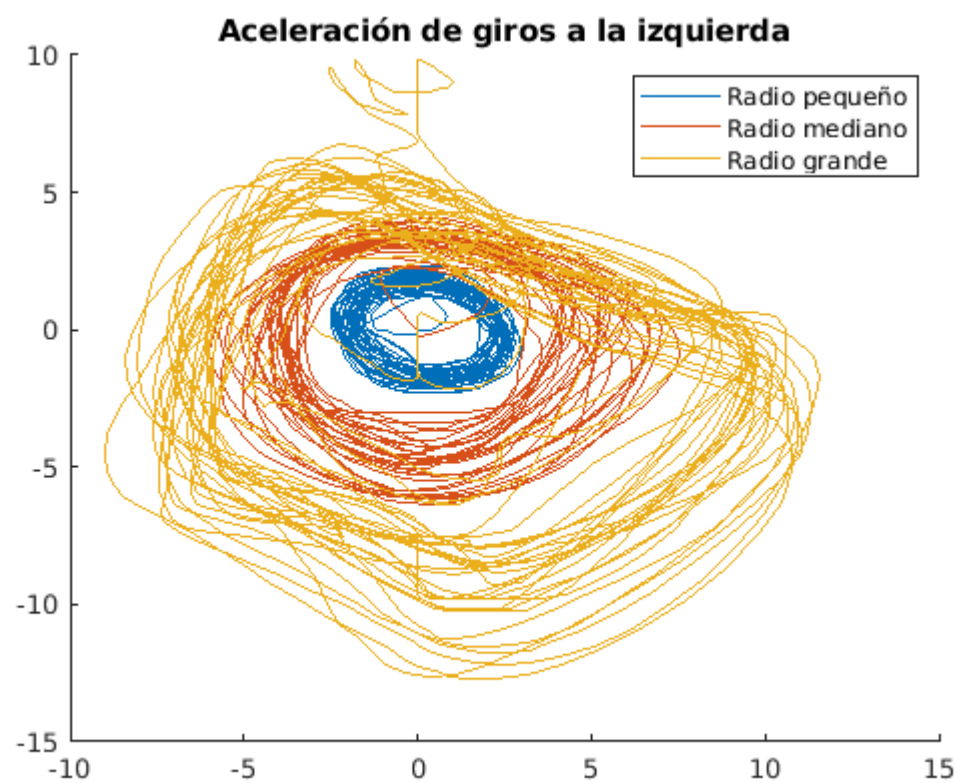
Los datos se recogieron mediante en el acelerómetro de mi móvil personal, en particular, limitándome al plano de una mesa y seleccionando sólo las componentes involucradas: la  $x$  y la  $y$ . Los movimientos fueron grabados en bloques, en los que procuraba que un movimiento completo durase en torno a 1 segundo. La tasa de muestreo es de, aproximadamente, 60 datos por segundo. De esta forma, para extraer las observaciones individuales de las curvas, sólo hay que dividir los datos de cada bloque en trozos iguales. En total, grabé cuatro tipos de movimientos, a saber:

- Círculos a derechas (radio pequeño, mediano y grande).

- Círculos a izquierdas (radio pequeño, mediano y grande).
- Movimientos rectilíneos (hasta 4 direcciones diferentes).
- Movimientos indefinidos (no serán tenidos en cuenta sino hasta el final de esta memoria).

A continuación, cargamos los datos y graficamos las aceleraciones de todos los bloques algunas de las las curvas:



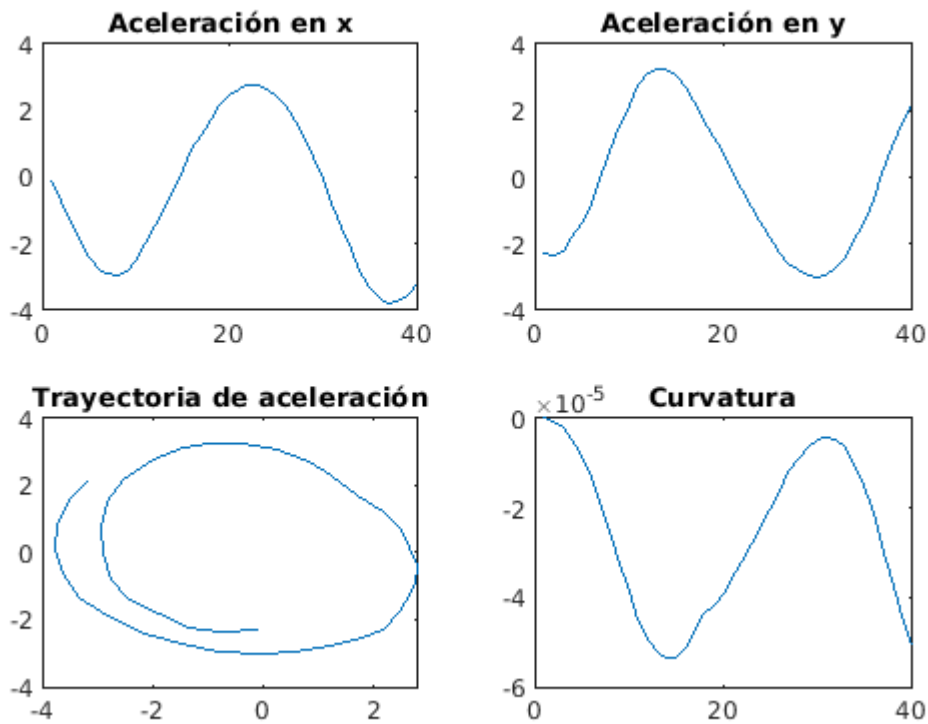


Ahora, graficamos tres ejemplos concretos de los círculos y las rectas. Nótese que se ha calculado la curvatura de cada movimiento, siguiendo la ecuación de *Curvas y Superficies. Definiciones, teoremas y resultados* de Juan de Burgos Román:

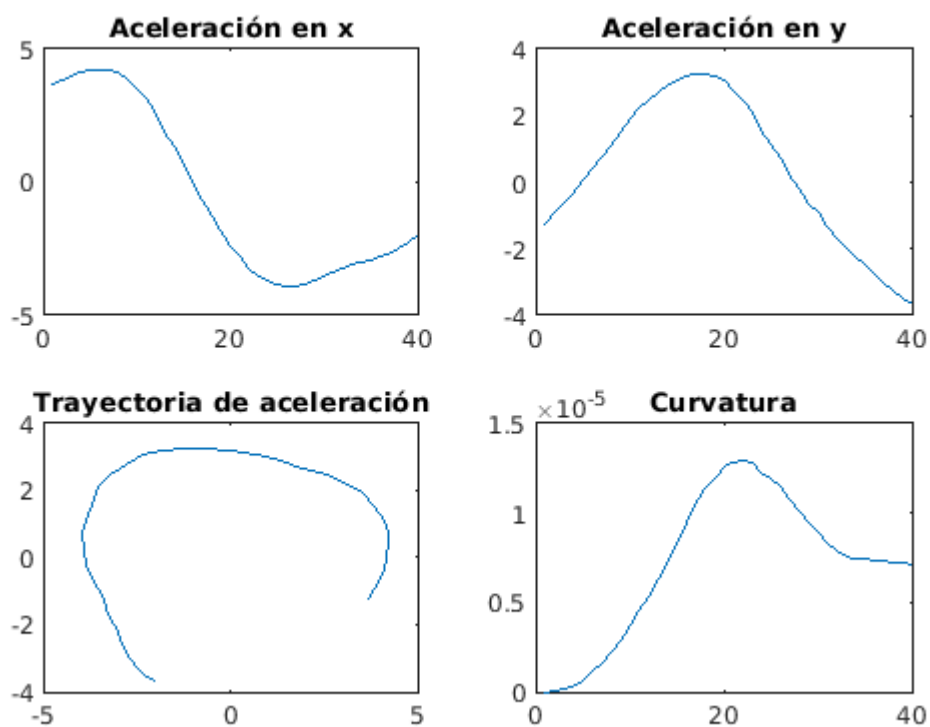
$$k = \frac{x'(\lambda)y''(\lambda) - x''(\lambda)y'(\lambda)}{[x'(\lambda)^2 + y'(\lambda)^2]^{3/2}},$$

donde la curva viene definida por  $x = x(\lambda)$ ,  $y = y(\lambda)$  y  $\lambda$  es el parámetro temporal.

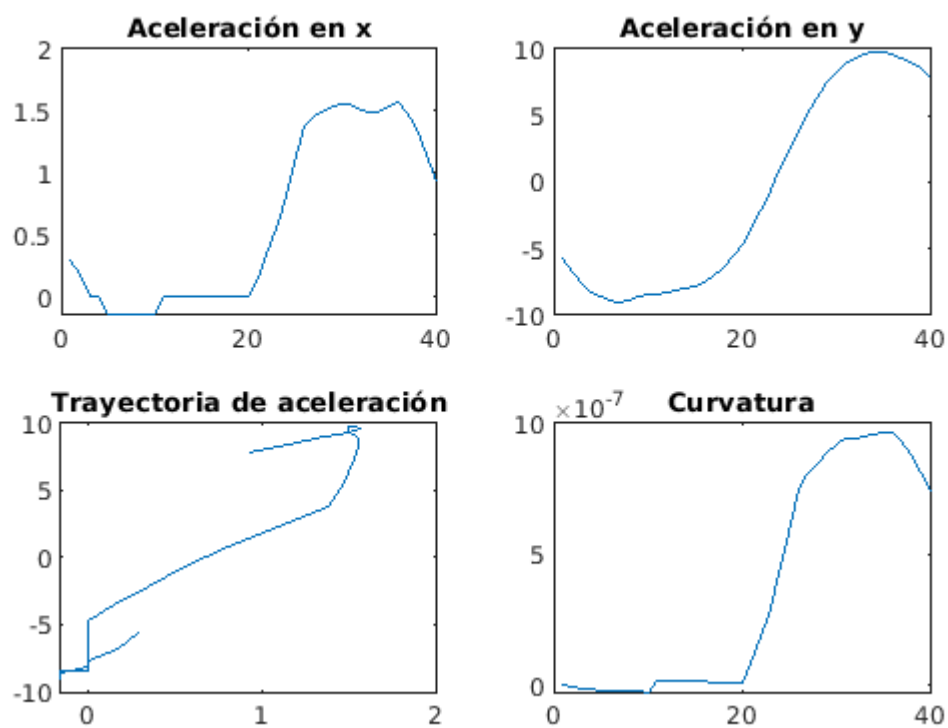
**Figuras para la observación nº1: Círculo a dchs.**



Figuras para la observación nº140:Círculo a izqs.



Figuras para la observación nº200:Mov. rectilíneo



Se puede observar que los ejemplos no son perfectos: hay círculos con algo más de una vuelta, otros sin llegar a terminar una vuelta y las rectas tienen imperfecciones en los extremos. Estos defectos están íntimamente ligados con la forma de separar cada observación de un bloque o apilamiento de observaciones y con la recolección de los datos, que puede verse en el siguiente [video de Youtube](#).

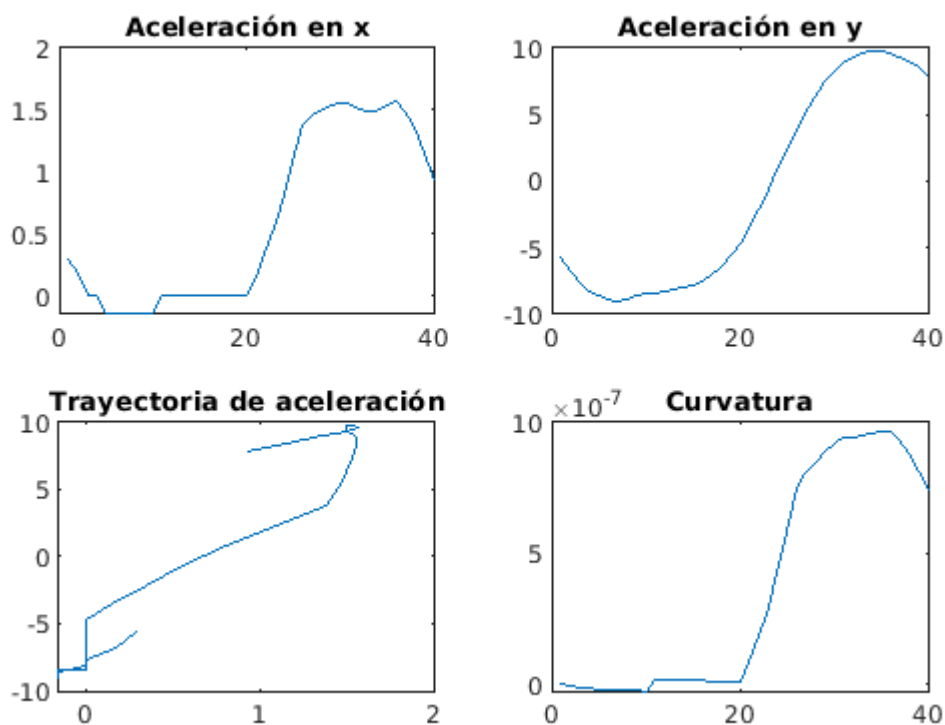
## Procedimiento de resolución

En primer lugar, hago uso de una separación inteligente de los datos: un 80% de aprendizaje y un 20% de test. A continuación, entreno un mapa auto-organizado de Kohonen (Self Organizing Map, SOM) de  $8 \times 8$  neuronas, evalúo la separación de las distintas categorías. Tras comprobar que los resultados son satisfactorios, asigno a cada neurona entrenada la clase o tipo de movimiento que más representa. De esta forma, he construido un clasificador que puede ser evaluado sobre el conjunto de test.

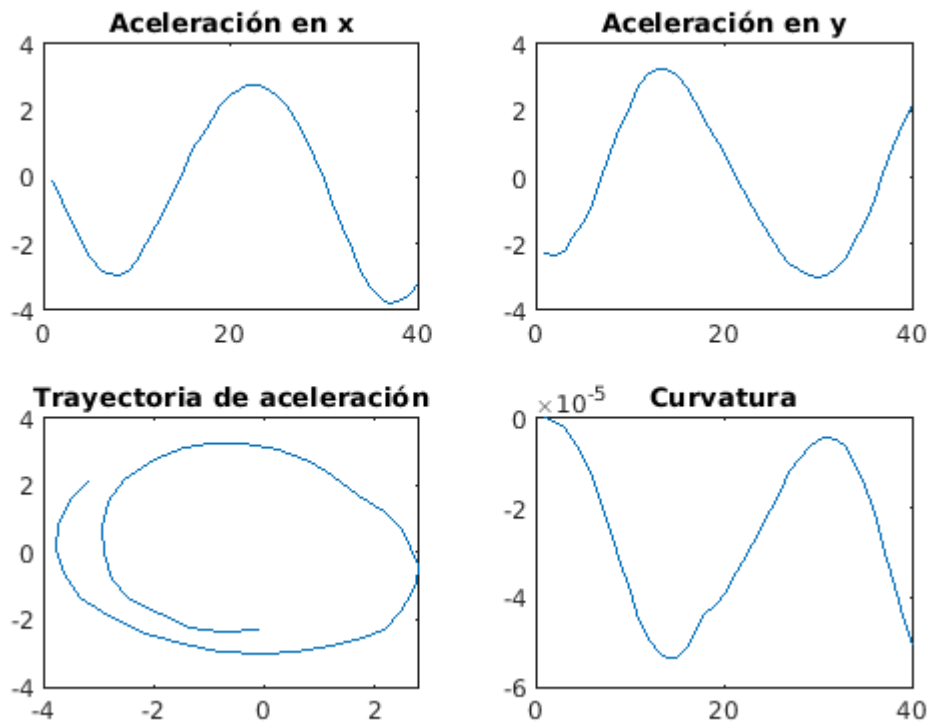
## Implementación del SOM

La siguiente caja de código implementa el SOM.

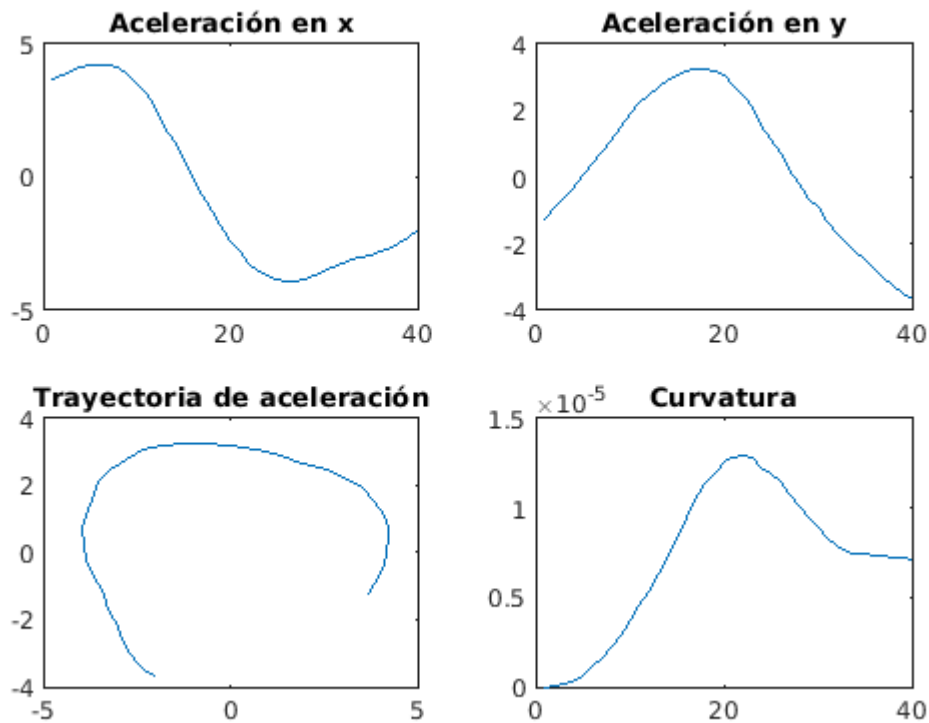
Figuras para la observación nº200: Mov. rectilíneo

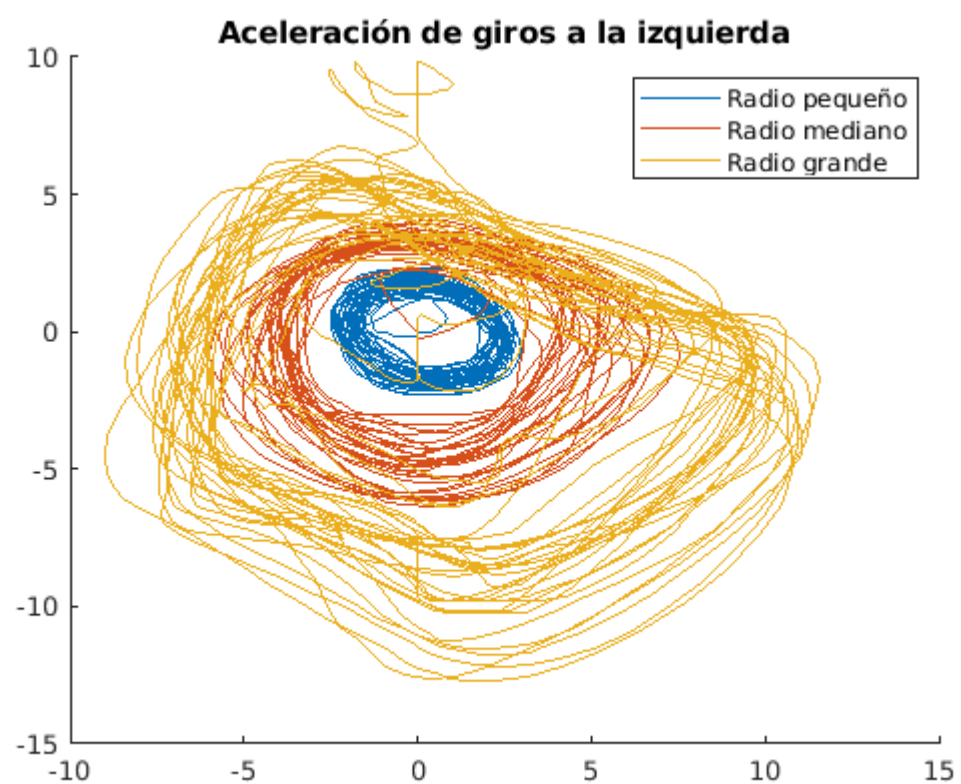
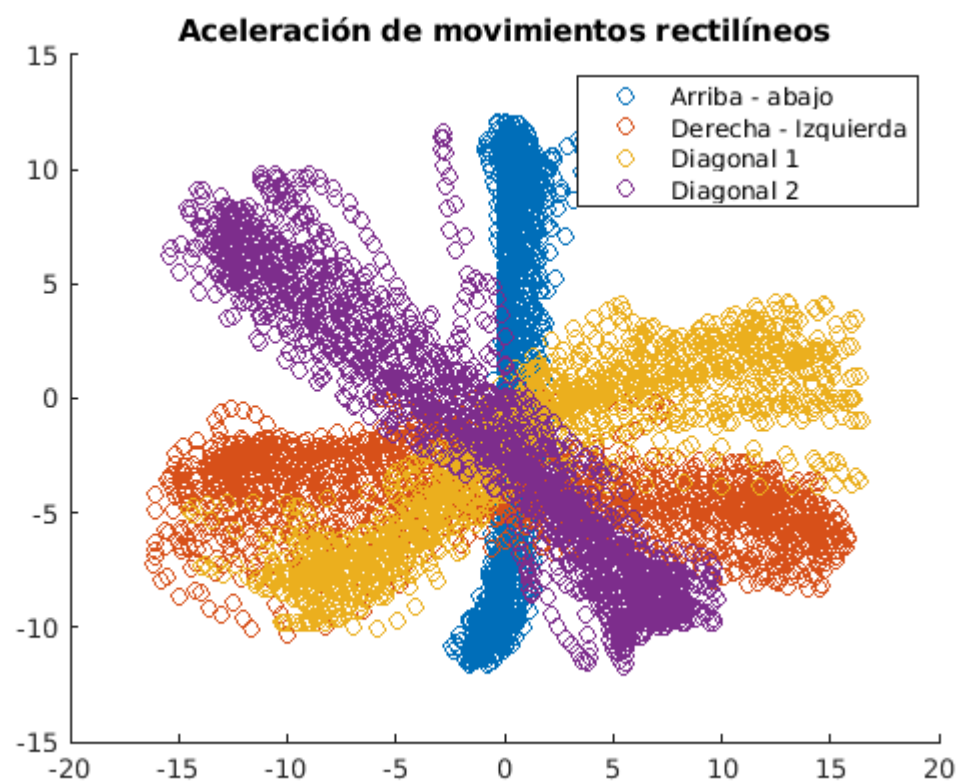


Figuras para la observación nº1:Círculo a dchs.



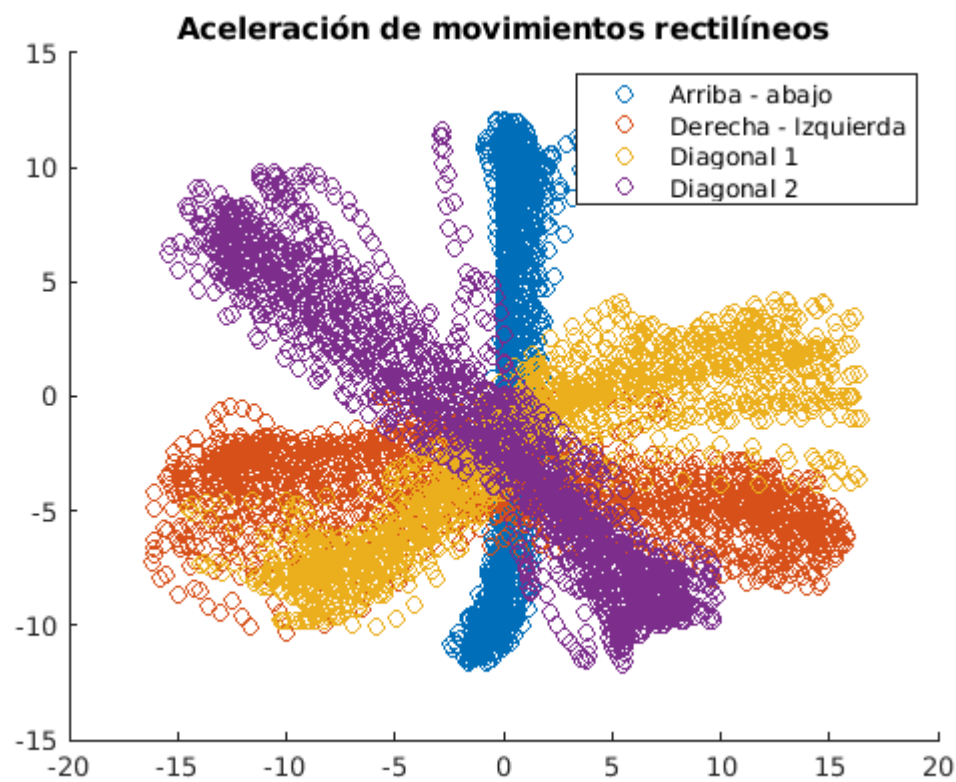
Figuras para la observación nº140:Círculo a izqs.



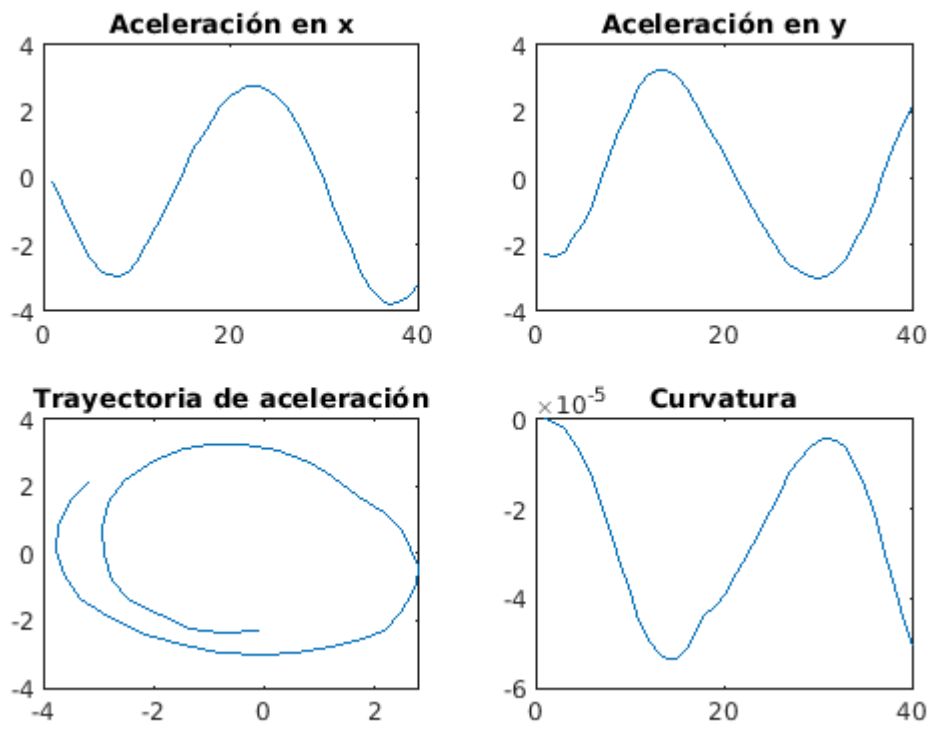




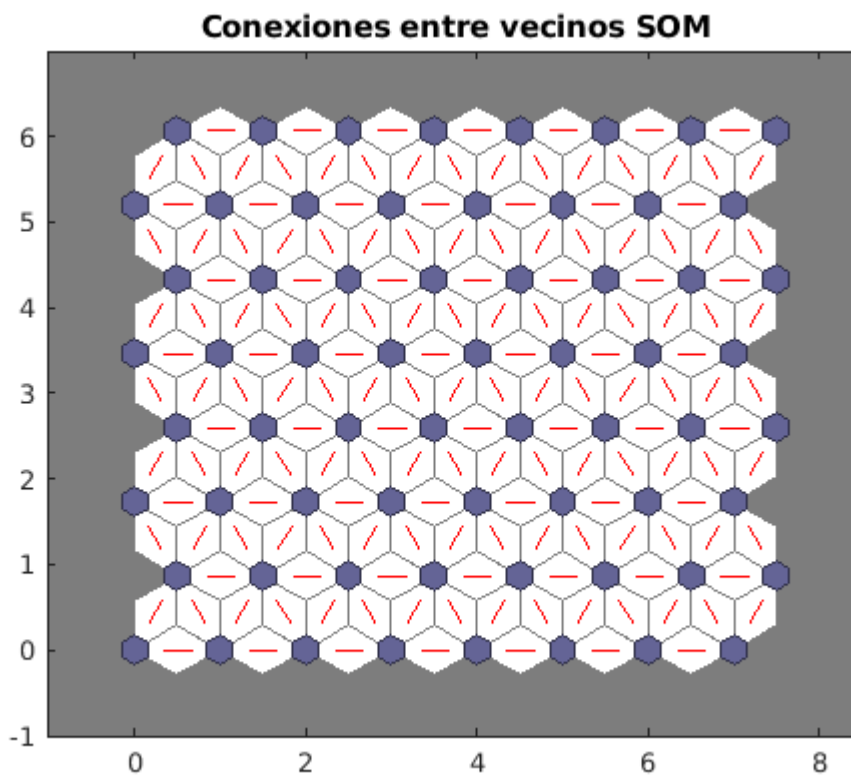
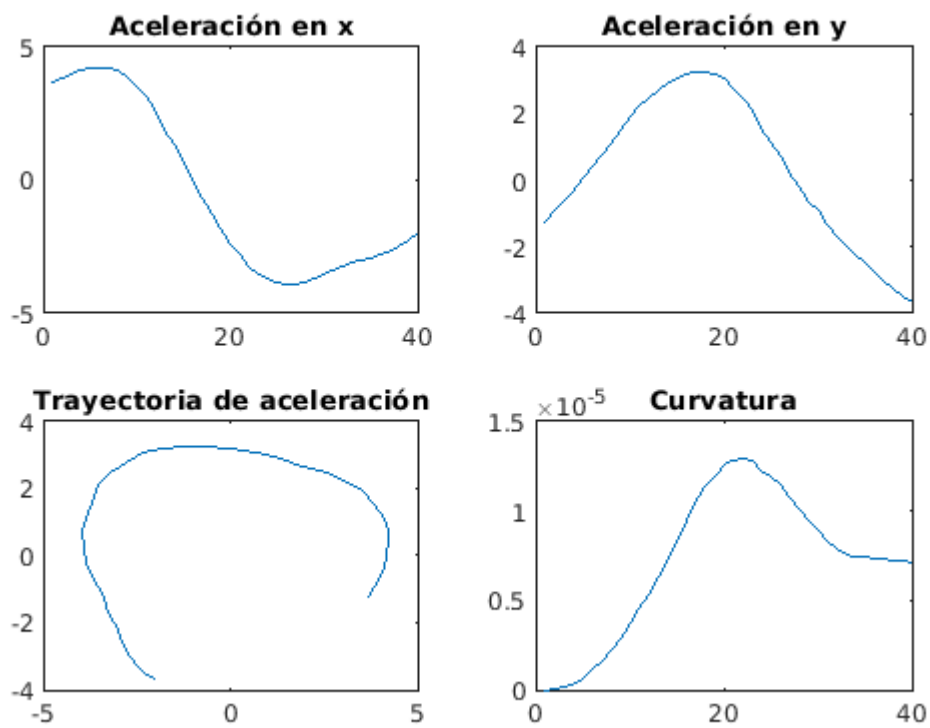
Las neuronas de la red se distribuyen según una topología hexagonal, como puede apreciarse en la figura a continuación, en la que las neuronas son los polígonos azules grisáceos y sus vecinos directos de muestran con líneas rojas.



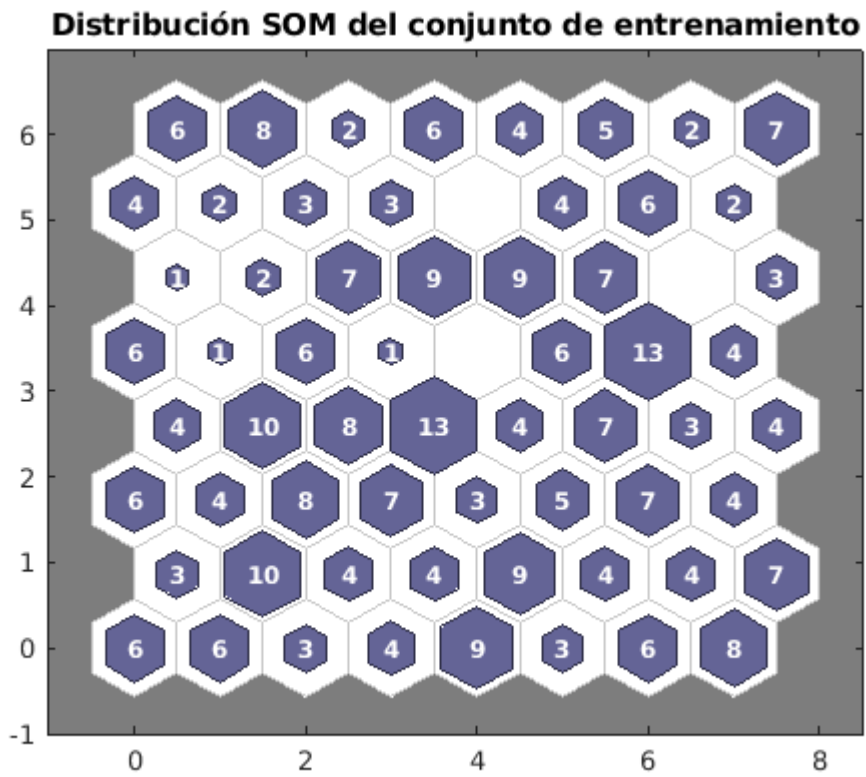
Figuras para la observación nº1: Círculo a dchs.



Figuras para la observación nº140: Círculo a izqs.

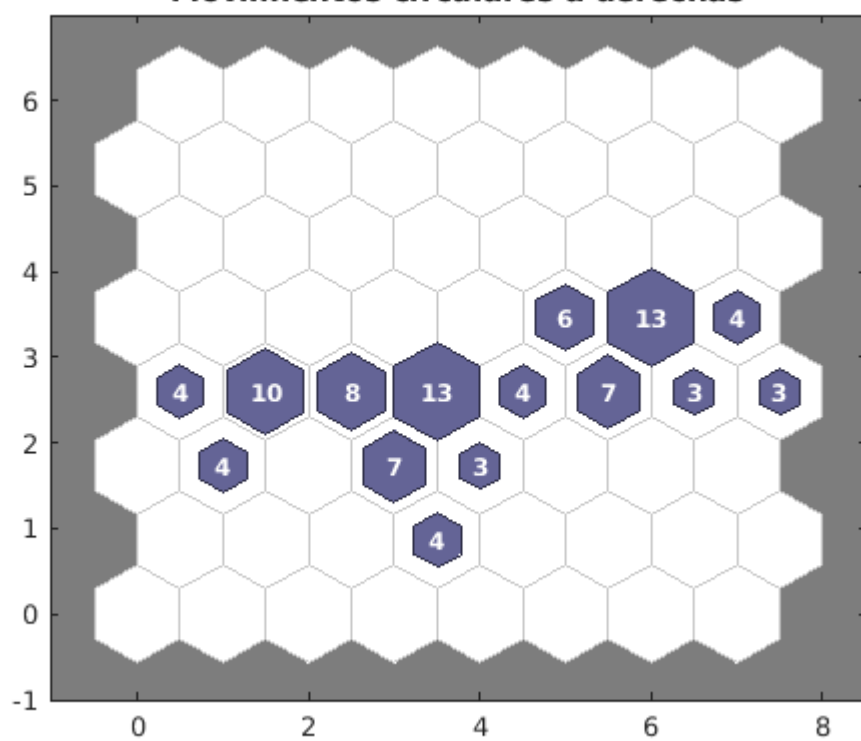


El número de observaciones de entrenamiento que se asocia a cada neurona puede leerse en este gráfico:

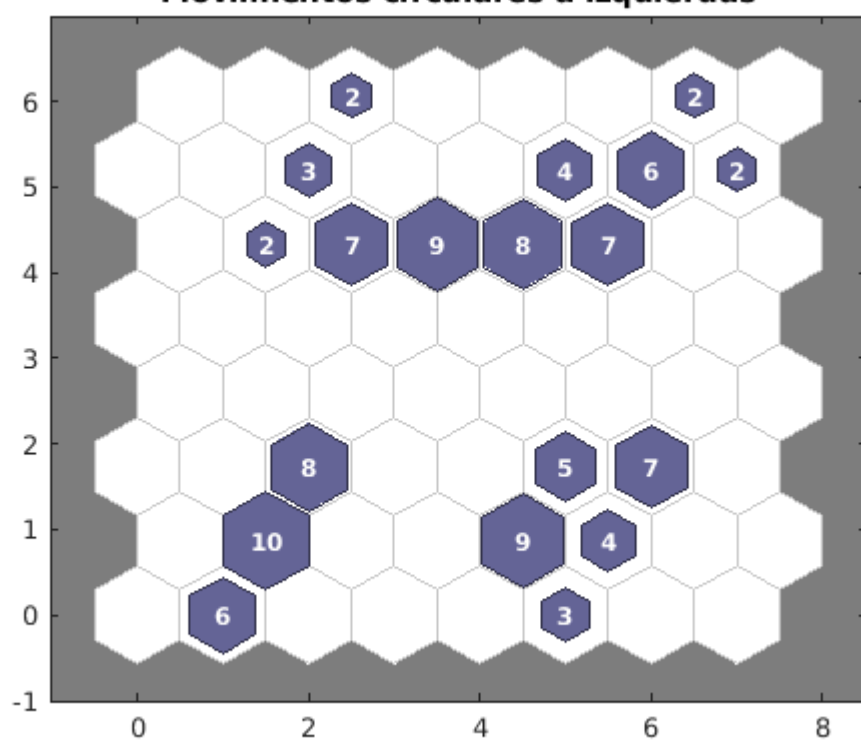


Es imposible determinar con la figura anterior si el mapa auto-organizado logra separar los grupos de los diversos tipos de movimientos considerados en esta fase: movimientos circulares hacia la derecha, circulares hacia la izquierda y movimientos rectilíneos; por lo tanto, los grafico ahora:

**Movimientos circulares a derechas**



**Movimientos circulares a izquierdas**



Se puede discernir una buena separación: movimientos circulares a derechas por el centro, circulares a izquierdas colindantes a aquellos y los movimientos rectilíneos en el borde del mapa. Ahora bien, lo ideal es elaborar una matriz de confusión y calcular un porcentaje de error, como hacemos a continuación.

		Datos de entrenamiento		
Clase verdadera	Círculo a dchs.	93		
	Círculo a izqs.		104	
	Mov. rectilíneo	1	1	127
		Círculo a dchs.	Círculo a izqs.	Mov. rectilíneo
		Clase predicha		

El porcentaje de error sobre el conjunto de entrenamiento es: 0.6135%.

## Clasificando el conjunto de test

Para verificar realmente lo bueno que es el clasificador, debe estudiarse el error de clasificación en el conjunto de test, como se hace en este trozo de código.

		Datos de test		
Clase verdadera	Círculo a dchs.	25		
	Círculo a izqs.	2	22	1
	Mov. rectilíneo	3	2	20
		Círculo a dchs.	Círculo a izqs.	Mov. rectilíneo
		Clase predicha		

El porcentaje de error sobre el conjunto de test es: 10.6667%.

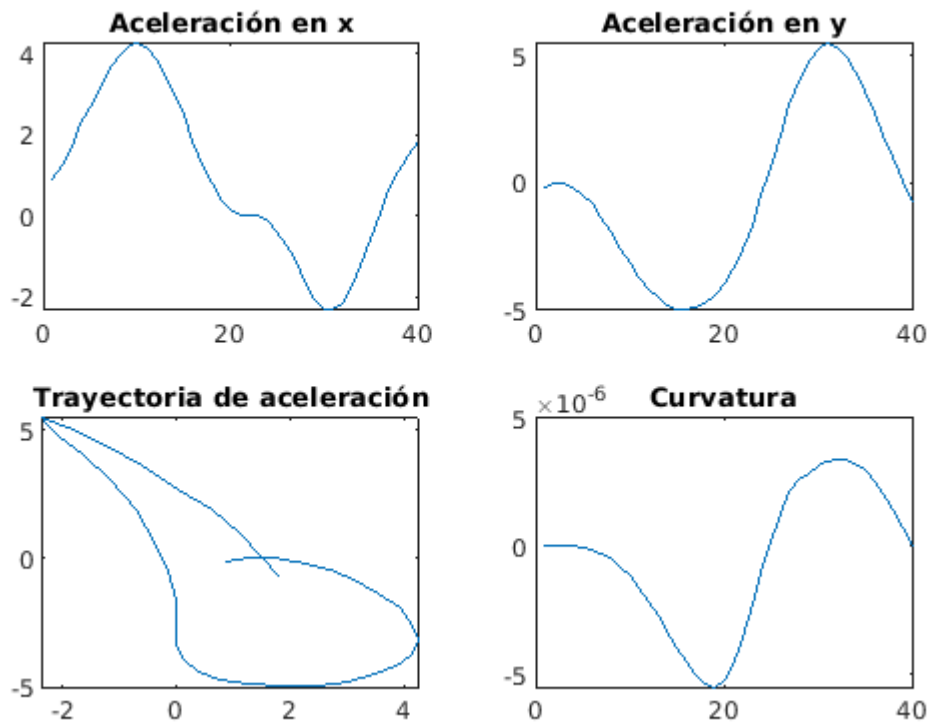
## Discusión

El bajo error de clasificación cometido en entrenamiento muestra que el mapa auto-organizado de Kohonen efectivamente es capaz de separar los tres tipos de movimientos, aunque incurriendo, posiblemente, en algo de sobre-aprendizaje, como lo sugiere el hecho de tener dos órdenes de magnitud más en el error de clasificación sobre el conjunto de test. Ahora bien, el error de clasificación en test está en torno al 10%, lo cual es un resultado prometedor, teniendo en cuenta que no he realizado ningún tipo de ajuste de hiperparámetros en el SOM, sino que he aplicado todos los valores por defecto.

Es, quizás, censurable la simplicidad y relativamente alta dimensionalidad de las características usadas en la clasificación de los diferentes tipos de movimiento. En efecto, las características consisten en la concatenación de 40 medidas consecutivas de aceleración en el eje  $x$ , otras 40 de aceleración en el eje  $y$ , antes de otras 40 medidas de curvatura. Luego, estoy clasificando vectores de valores reales, de dimensión 120. No obstante, precisamente, esta metodología seguida se estableció así para resaltar la capacidad del SOM en separar patrones de alta dimensionalidad usando simplemente una proyección en dos dimensiones.

Además, con el fin de simular una situación más realista, también registré un nuevo tipo de curva, el "movimiento indefinido". Lo elaboré realizando movimientos ondulantes sobre el plano que no terminaban de cerrarse en un círculo. Repito ahora, escuetamente, el proceso de esta memoria y muestro los resultados en test. Adelanto que el error sobre el conjunto de test se mantiene en torno al 10%.

Figuras para la observación nº390:Mov. indefinido



Datos de entrenamiento				
Clase verdadera	Círculo a dchs.			
	Círculo a izqs.			
	Mov. indefinido			
	Mov. rectilíneo			
	88		5	
		104		
	8	7	49	
	1	1		127
Clase predicha				
Círculo a dchs.      Círculo a izqs.      Mov. indefinido      Mov. rectilíneo				

El porcentaje de error sobre el conjunto de entrenamiento es: 5.6410%.

Datos de test				
Clase verdadera	Círculo a dchs.			
	Círculo a izqs.			
	Mov. indefinido			
	Mov. rectilíneo			
	25			
	2	22		1
	6	1	18	
		1		24
Clase predicha				
Círculo a dchs.      Círculo a izqs.      Mov. indefinido      Mov. rectilíneo				

El porcentaje de error sobre el conjunto de test es: 11.0000%.



## Conclusión

He implementado exitosamente un mapa auto-organizado de Kohonen sobre la concatenación de medidas cinemáticas de curvas. También, se ha mostrado la capacidad del mapa auto-organizado de segmentar en el plano un conjunto de datos de muchas dimensiones. Dicha capacidad a quedado cuantificada con los relativamente bajos errores de clasificación en los conjuntos de test de sendos experimentos propuestos (con y sin "movimientos indefinidos").

Para finalizar, quiero hacer énfasis en los aspectos éticos del problema abordado, en lo que concierne al uso de la solución aquí desarrollada para abordar soluciones de accesibilidad para personas con discapacidad, como mencionaba en la introducción. El punto más importante es la fiabilidad de la solución. En efecto, un error en test de entorno al 10% puede ser demasiado alta para usarla cómodamente en el manejo de un teléfono móvil o un ordenador, más aún, su uso sería, seguramente, inaceptable como software crítico en el control por parte de una persona con discapacidad de un exoesqueleto. Ahora bien, queda como trabajo futuro un análisis y estudio detallado con el fin de disminuir el error, y desarrollar, por lo tanto, una solución aceptable para aplicaciones de control del teléfono o una computadora.

## Anexo: Funciones

En esta sección se recogen todas las funciones elaboradas para la confección de este trabajo, a saber:

- `load_data`
- `plotAcceleration`
- `plotExample`
- `countsTable`