



Universidad
Tecmilenio®



Aprendizaje Automático No Supervisado

Análisis de
componentes



A menudo los datos multivariados se ven como múltiples mediciones indirectas que surgen de una fuente subyacente que generalmente no se puede medir directamente. A continuación, se presentan algunos ejemplos:

- Las pruebas educativas y psicológicas utilizan las respuestas a los cuestionarios para medir la inteligencia subyacente y otras habilidades mentales de los sujetos.
- Los escáneres cerebrales EEG miden la actividad neuronal en varias partes del cerebro a través de señales electromagnéticas registradas en sensores que se colocan en varias posiciones de la cabeza.
- Los precios de negociación de las acciones cambian constantemente con el tiempo, por lo que reflejan varios factores no medidos, por ejemplo, la confianza del mercado, las influencias externas, así como otras fuerzas impulsoras que pueden ser difíciles de identificar o medir.



Para cada uno de los casos anteriores, el análisis de componentes independientes (ICA), el cual es un método computacional, resulta efectivo para dar una solución. En este tema aprenderás sus fundamentos y cómo implementarlo.



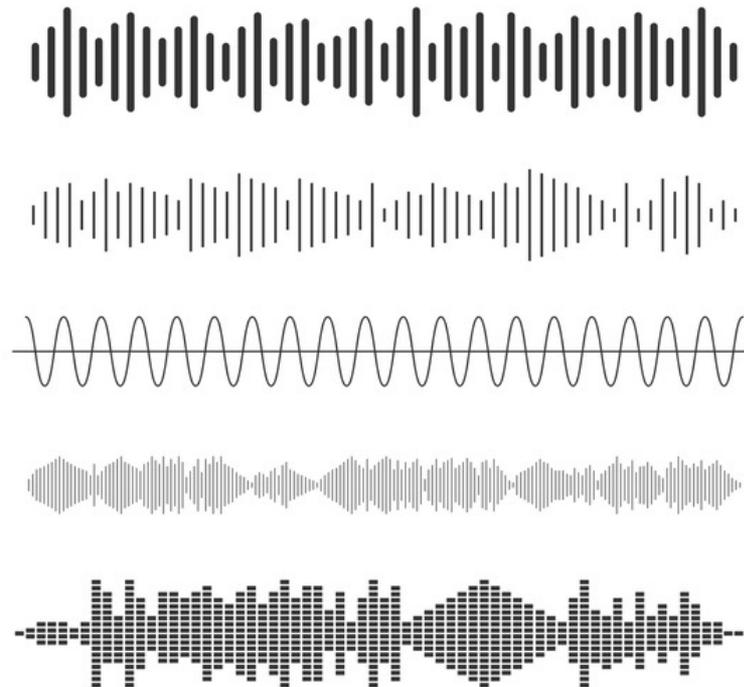


El análisis de componentes independientes (ICA) es un problema en el procesamiento de señales desde tiempo atrás.

Esta considera una situación en la que hay una serie de señales emitidas por algunos objetos o fuentes físicas.

Estas fuentes podrían ser diferentes áreas del cerebro que emiten señales eléctricas, como personas que hablan en la misma habitación emitiendo señales de voz o teléfonos móviles que emiten sus ondas de radio.

Además, pueden haber varios sensores o receptores que estén en diferentes posiciones, por lo que cada uno registra una mezcla de las señales de la fuente original con ligeras variaciones





A continuación, se consideran tres señales de fuentes implícitas y tres señales observadas. Sean $x_1(t)$, $x_2(t)$ y $x_3(t)$ las señales observadas, que son las amplitudes de las señales registradas en el momento t , y $s_1(t)$, $s_2(t)$ y $s_3(t)$ las señales originales. Entonces, las $x_i(t)$ son sumas ponderadas de $s_j(t)$, donde los coeficientes dependen de las distancias entre las fuentes y los sensores:

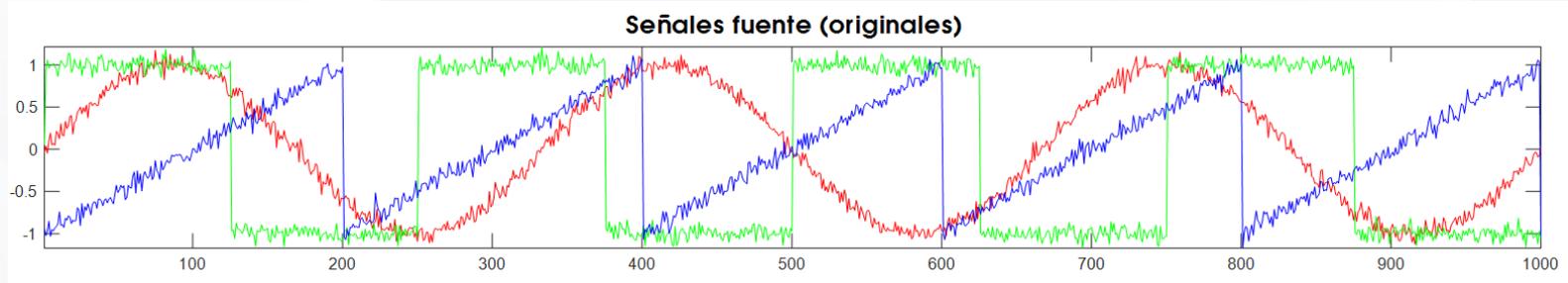
$$\begin{aligned}x_1(t) &= c_{11}s_1(t) + c_{12}s_2(t) + c_{13}s_3(t) \\x_2(t) &= c_{21}s_1(t) + c_{22}s_2(t) + c_{23}s_3(t) \\x_3(t) &= c_{31}s_1(t) + c_{32}s_2(t) + c_{33}s_3(t)\end{aligned}$$

Los c_{ij} son coeficientes constantes que corresponden a los pesos en la mezcla. Se asume que son desconocidos, ya que no es posible conocerlos sin saber todas las propiedades del sistema de mezcla físico, lo cual puede resultar muy difícil en general. Las señales fuente s_i también son desconocidas, ya que el problema consiste en que no es posible registrarlas de manera directa.

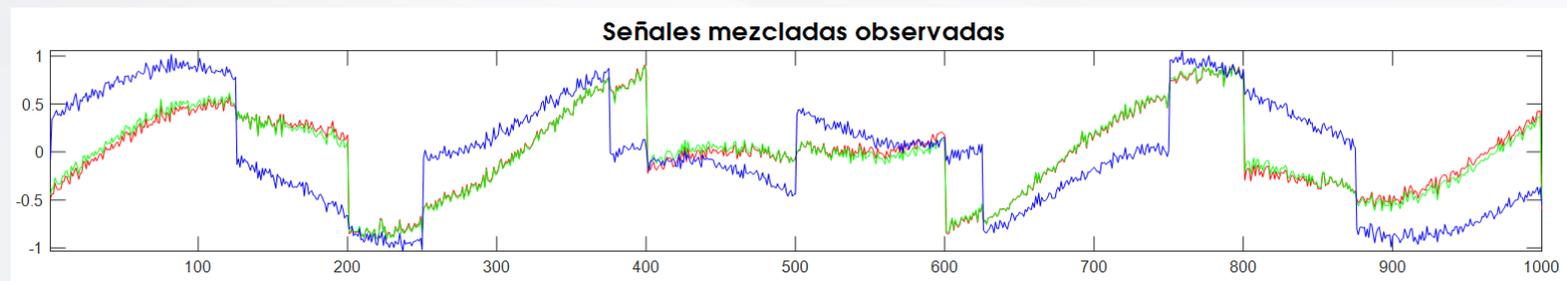




Con el propósito de ilustrar lo anterior, y empleando un proyecto en MATLAB, se generaron las tres formas de onda de la figura, las cuales corresponden a una señal sinusoidal, cuadrada y de diente de sierra con ruido agregado.



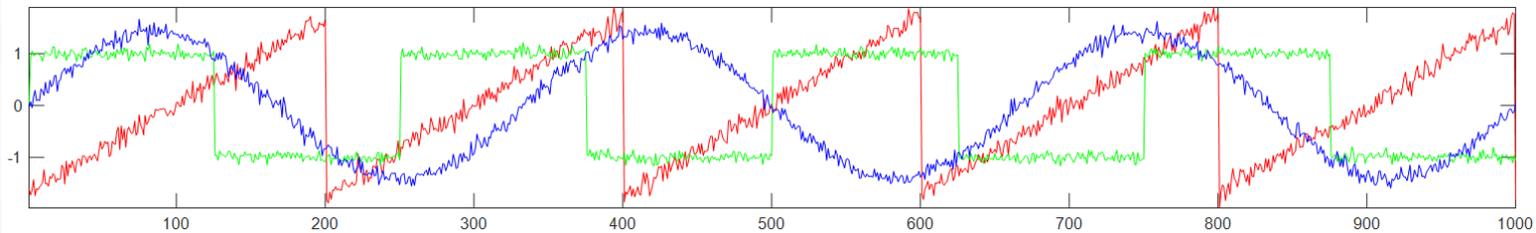
Después se realiza la mezcla de las señales originales que representarán a las señales mezcladas observadas x_i . Aunque la forma de cada una ha cambiado significativamente, las señales fuente están ocultas en estas.





Basándose en la independencia estadística es posible estimar la matriz de coeficientes W y obtener las señales fuente de la figura. Para esto se utilizó el algoritmo FastICA, el cual consiste en una versión popular y eficiente del método de ICA.

Estimación de las señales fuente





Al abordar computacionalmente el problema ICA conviene pensar en un conjunto de observaciones de variables aleatorias $(x_1(t), x_2(t), \dots, x_n(t))$, donde t es el tiempo o el índice de la muestra. Por ende, supón que se generan como una mezcla lineal de componentes independientes, en donde A es una matriz desconocida.

$$\begin{pmatrix} x_1(t) \\ x_2(t) \\ \cdot \\ \cdot \\ x_n(t) \end{pmatrix} = A \begin{pmatrix} s_1(t) \\ s_2(t) \\ \cdot \\ \cdot \\ s_n(t) \end{pmatrix}$$

El análisis de componentes independientes (ICA) consiste en estimar tanto la matriz A como el elemento $s_i(t)$, cuando solo se observa a $x_i(t)$. Aunque se está asumiendo que el número de componentes independientes es igual al número de variables observadas, solo es una suposición para simplificar el problema, por lo que no es completamente necesaria.

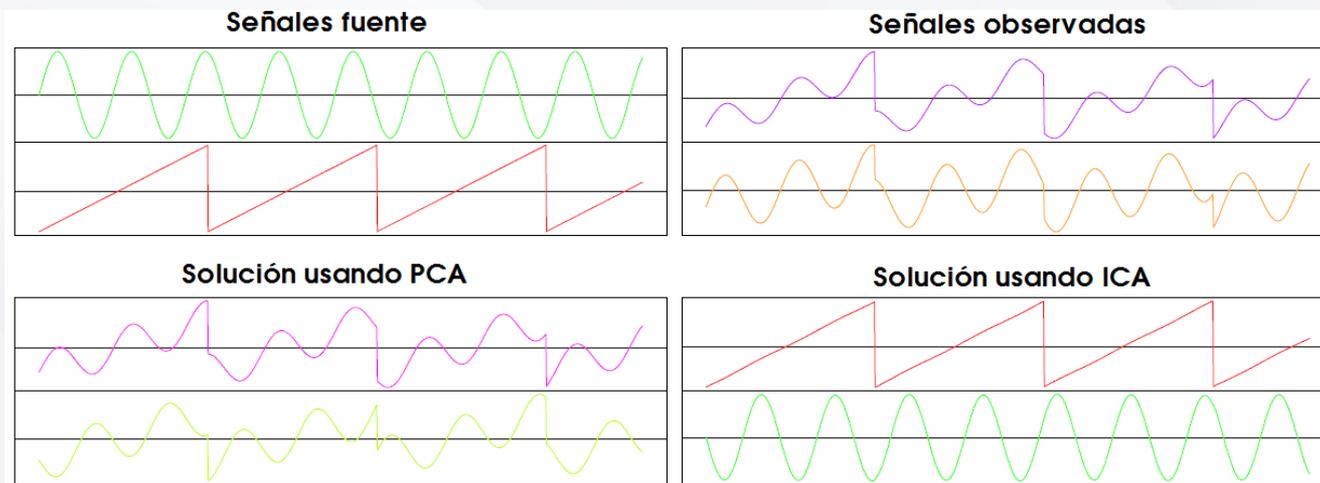
También podría definirse al ICA como una transformación lineal dada por una matriz W , de modo que las variables aleatorias $u_i, i = 1, \dots, n$ sean lo más independientes posible. Es importante tener en cuenta que el modelo puede estimarse solo si los componentes s_i no son gaussianos.





Aunque ICA separa dos señales mixtas, es comparado con otro método estadístico conocido como PCA. Esto es un ejemplo de un problema clásico en el procesamiento de señales conocido como “el efecto de fiesta de cóctel”, en donde diferentes micrófonos X_j captan mezclas de diferentes fuentes independientes S_i (música, voces de diferentes altavoces, etc.).

En la figura se muestran los resultados. El panel superior izquierdo muestra las dos señales fuente, mientras que el derecho muestra las señales mezcladas observadas. Los dos paneles inferiores muestran las soluciones al emplear cada algoritmo.

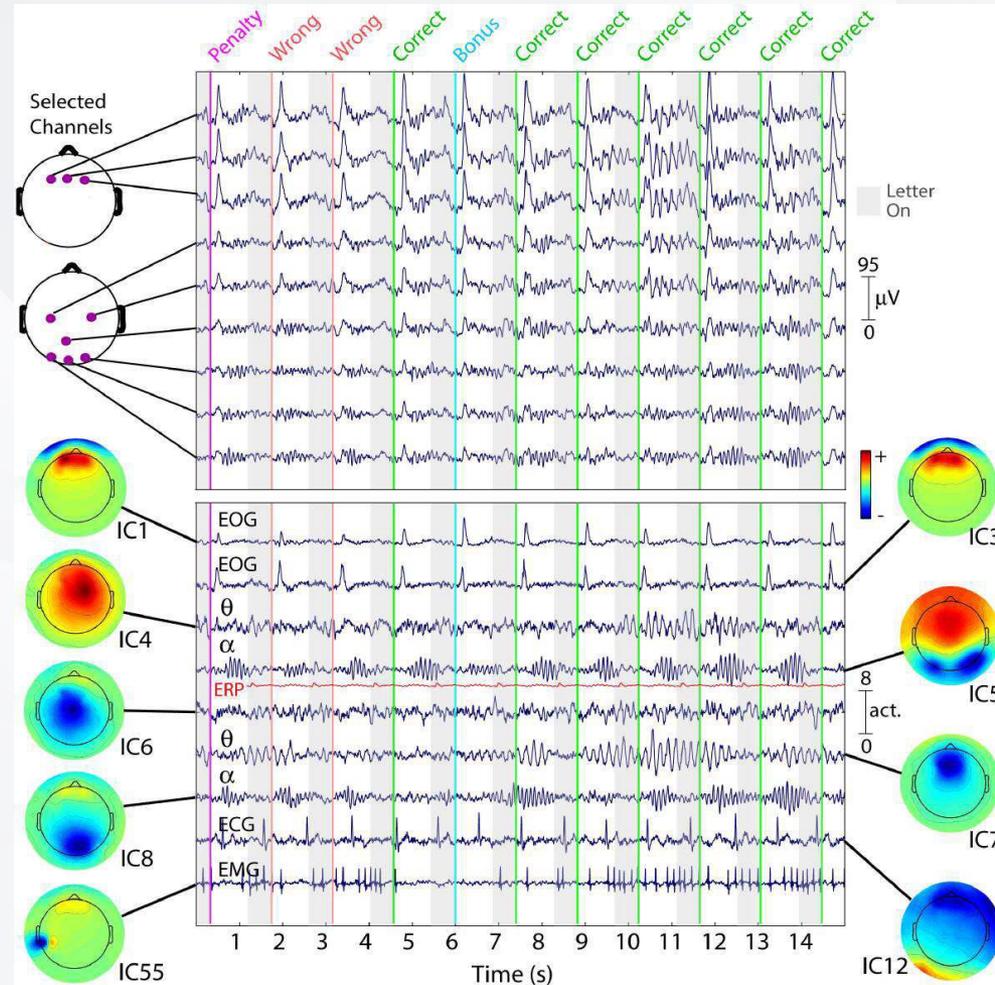




ICA se ha convertido en una herramienta bastante útil para entender el comportamiento del cerebro. El siguiente ejemplo consiste en utilizar ICA para desenredar los componentes de las señales en datos electroencefalográficos (EEG).

La parte inferior de la figura muestra una selección de componentes ICA.

Los segmentos de tiempo correspondientes muestran la actividad de los componentes ICA aprendidos.





Después de haber estudiado el tema puedes abordar las siguientes cuestiones:

- ¿Para qué tipo de problemas vale la pena implementar el algoritmo de análisis de componentes independientes?
- ¿Por qué considerarías que ICA corresponde a un método no supervisado?



En este tema se abordaron los fundamentos para el análisis de componentes independientes, desde su formulación matemática hasta su implementación mediante software.

Como se comentó antes, debido a su generalidad, el método ICA tiene aplicaciones en muchas áreas diferentes, por ejemplo, las señales cerebrales. Sin embargo, en econometría, a menudo se tienen series de tiempo paralelas, por lo que el análisis con ICA podría separarlas en componentes independientes, lo que darían una idea de la estructura del conjunto de datos. Otra aplicación algo diferente es la extracción de características de imágenes, en donde el objetivo consiste en encontrar características que sean lo más independientes posible.

Finalmente, sería conveniente preguntarte lo siguiente: ¿qué otro tipo de aplicaciones podrías encontrar para el análisis con ICA?, ¿qué otro tipo de algoritmos se pueden relacionar con esta técnica?

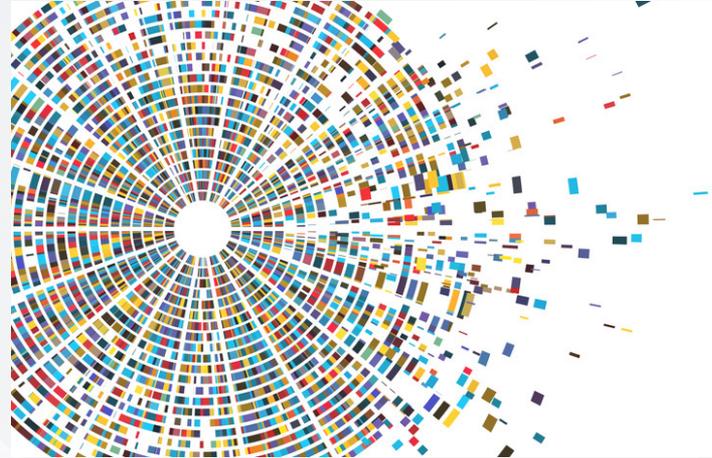


Aprendizaje Automático No Supervisado

Mapas autoorganizados
de características



Los mapas autoorganizados (SOM), también conocidos como mapas autoorganizados de características o mapas de Kohonen, constituyen un sistema de aprendizaje no supervisado a diferencia de otras variantes de redes neuronales, cuyos métodos de aprendizaje son inherentemente supervisados.



Los SOM también se destacan por utilizar un tipo diferente de función de costos, al cual se basa en la similitud de cada neurona con sus vecinas más cercanas, en vez de utilizar la propagación inversa como lo hacen la mayoría de las propuestas de aprendizaje profundo.

Este tipo de redes se llaman mapas porque las neuronas que las integran crean conexiones en forma de hojas que se activan mediante la presencia de varios patrones o clases de patrones en las señales de entrada. Los principales componentes de la autoorganización son la inicialización, la competencia, la cooperación y la adaptación.





Las redes neuronales que utilizan el aprendizaje automático no supervisado solo utilizan como información primaria los datos de entrada, por lo que, en función de estos, generan algún tipo de salida.

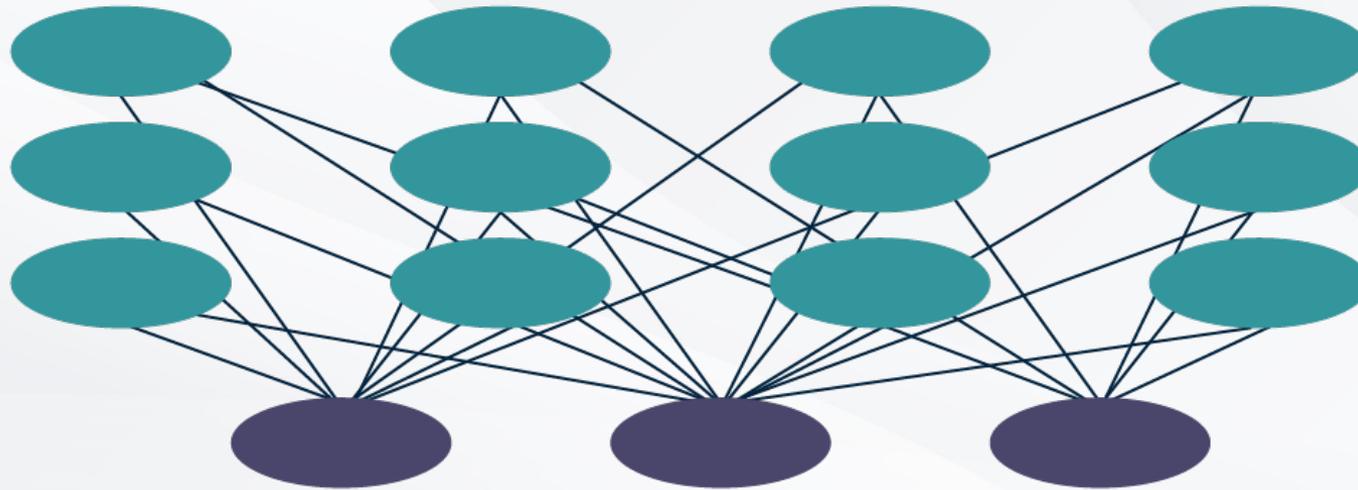
Las respuestas correctas no se conocen durante el proceso de aprendizaje, por que el trabajo de la red es intentar descubrir patrones en los datos de forma autodidacta y obtener algún tipo de agrupación o clasificación a partir de ellos.

Los mapas autoorganizados (SOM) se especializan en este tipo de aprendizaje, por lo que, aunque son una variante de red neuronal, los conceptos de neurona, conexión y pesos se interpretan de una manera sutilmente diferente.





En la figura se muestra una estructura básica de un mapa autoorganizado simple. En este ejemplo se tienen tres neuronas de entrada, que esencialmente representan las características del conjunto de datos (la entrada se puede representar mediante vectores bidimensionales, tridimensionales o de dimensión superior) y sobre estas se pueden observar las llamadas neuronas mapa.





Los mapas autoorganizados utilizan como base el aprendizaje no supervisado. Al proceso de entrenamiento se le denomina como **aprendizaje competitivo**.

El primer paso en el aprendizaje es la inicialización de los pesos en todas las conexiones, y como segunda etapa se utiliza como entrada de la red una muestra aleatoria del conjunto de datos y se calculan nuevamente los pesos de las neuronas que más se asemejan a los valores de las muestras recibidas (vector de entrada).

Para lo anterior, se utiliza una medida de similitud, que usualmente es el cuadrado de la distancia euclidiana, la cual se obtiene por la siguiente expresión:



$$distancia^2 = \sum_{i=0}^n (entrada_i - peso_i)^2$$

Donde n representa el número de conexiones.

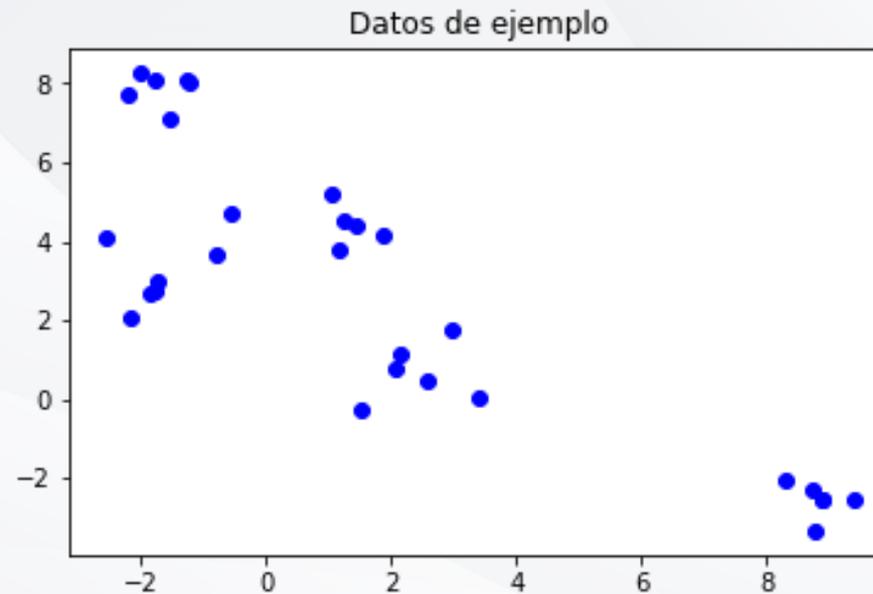




A continuación, se verá cómo se realiza el agrupamiento jerárquico utilizando el lenguaje de programación Python, así como varias librerías utilitarias que facilitan este proceso. Supongamos que se cuenta con un conjunto de datos como que se muestra en la figura.

Para implementar el agrupamiento jerárquico y construir el dendrograma, Python cuenta con un método específico denominado `cluster.hierarchy()`, que se encuentra en la librería especializada Scipy.

Por consiguiente, el código correspondiente a la construcción del modelo y a la construcción del dendrograma se ha implementado empleando tales herramientas.





La neurona del mapa con el mejor resultado se denomina como **unidad de mejor coincidencia** o **BMU**. En sentido general, esto significa que todo el vector de entrada se puede representar solamente con esta neurona de mapeo.



Durante el proceso de aprendizaje, los mapas autoorganizados no solamente calculan el valor de la BMU, sino que también intentan acercar este valor a los datos de entrada recibidos. Esto significa que los pesos de la conexión se actualizan, de tal manera que la nueva distancia calculada sea cada vez menor. Los pesos de los vecinos de la BMU también se modifican para que se acerquen al vector de entrada, con el fin de que todo el mapa se vaya acomodando en dirección a este punto (figura).





El siguiente paso en el entrenamiento del mapa es calcular el valor correspondiente al radio de los vecinos. Para obtener este valor, el algoritmo aplica la siguiente relación:

$$\sigma(t) = \sigma_0 e^{-\frac{t}{\lambda}}$$

Donde t es la iteración actual y σ_0 es el radio del mapa. El valor correspondiente a λ se define por:



$$\lambda = \frac{k}{\sigma_0}$$

Donde k indica el número de iteraciones.





La fórmula anterior utiliza un decrecimiento exponencial, haciendo que el radio se reduzca a medida que avanza el proceso de entrenamiento, que era el objetivo inicial. Esto significa que cada iteración acercará más los puntos relevantes a los datos de entrada.

La operación de ajuste iterativo que se realiza durante el aprendizaje se muestra gráficamente en la figura:





El algoritmo de aprendizaje de los mapas autoorganizados se puede resumir en los siguientes puntos:

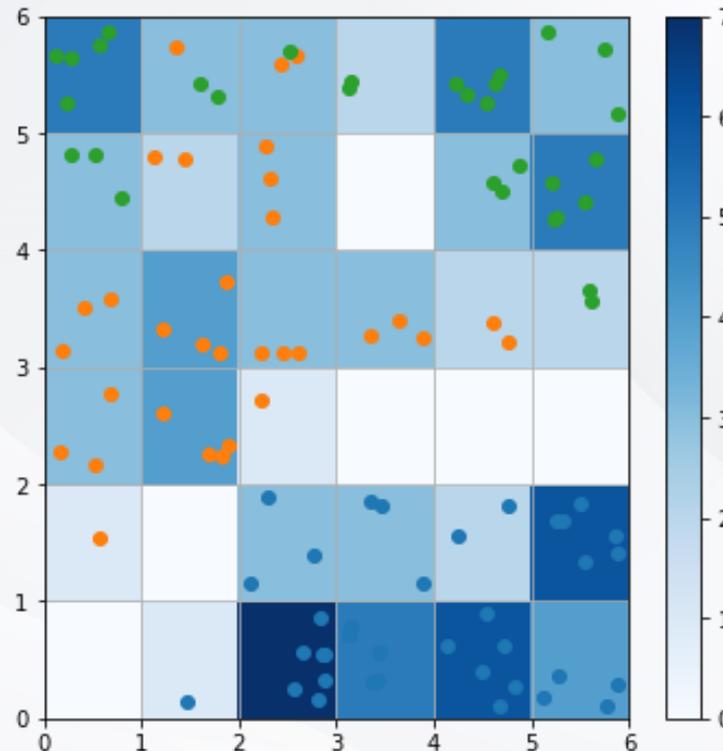
1. Inicialización de pesos.
2. Selección del conjunto de datos que constituyen el vector de entrada y utilizarlos para alimentar la red.
3. Calcular la BMU.
4. Calcular el radio del vecindario que se actualizará progresivamente.
5. Ajustar el peso de las neuronas dentro del radio para que se acerquen al valor del vector de entrada.
6. Repetir los pasos del 2 al 5 para el resto de los vectores de entrada que se puedan conformar con el conjunto de datos.





Mediante la biblioteca Scikit-Learn de Python, así como importando un conjunto de datos, se ha construido un mapa de Kohonen. A saber, es necesario declarar la cantidad de neuronas que va a tener el mapa resultante (arreglo bidimensional), además del número de características que tienen los datos de entrada, la razón de aprendizaje y el valor de σ_0 (radio inicial del mapa).

Aunque se emplearon 3000 iteraciones en el modelo (épocas), es importante destacar que este valor es excesivamente alto. No obstante, a los efectos prácticos garantizará una máxima convergencia de modelo a los datos de entrenamiento.



En la figura se puede observar cómo la red se ajustó a los datos de entrenamiento, por lo que, con esta información como base, es posible utilizar el modelo generado para agrupar/clasificar nuevos valores de entrada. Al construir una función de clasificación, con el apoyo de la herramienta de reporte importada en las primeras secciones de este ejemplo, se puede comprobar la precisión del mapa de Kohonen.





Aunque el uso de los mapas de Kohonen se puede extrapolar a prácticamente cualquier situación, se mencionarán las más interesantes y populares.

- Aplicaciones marítimas.
- Robótica.
- Clasificación de imágenes satelitales.
- Estudios psicolingüísticos.
- Aplicaciones sociales y de negocios.

Por otro lado, en aplicaciones sociales como la clasificación de la pobreza mundial o la evaluación de factores de calidad de vida (el estado de salud, la nutrición y los servicios educativos), le ha permitido a diversas organizaciones identificar y tomar decisiones acertadas para distribuir los recursos de apoyo o efectuar adecuadamente sus proyectos de desarrollo social.





Piensa en los conceptos vistos durante el tema y responde las siguientes preguntas:

- ¿Cuáles diferencias notas entre las redes neuronales y los mapas autoorganizados?
- ¿Qué ventajas tienen los mapas autoorganizados sobre otros modelos no supervisados?
- Menciona dos aplicaciones prácticas de los mapas de Kohonen y su impacto en la sociedad.





En este tema se abordaron los mapas autoorganizados de características, los cuales forman parte de la familia de las redes neuronales, pero con una definición relativamente diferente.

Este tipo de red utiliza el aprendizaje competitivo, en donde, ante un estímulo en la entrada, solamente una neurona es capaz de activarse. Los mapas de Kohonen tienen solamente dos capas, por lo que todas las neuronas que conforman la capa de salida están interconectadas a las entradas, pero independientes entre sí.

Los usos de los SOM son muy diversos, por tanto, en la actualidad los podemos encontrar en aplicaciones como la compresión de datos, el reconocimiento del lenguaje, la separación de fuentes de sonido o la detección de fraudes en instituciones bancarias. Aunque su popularidad ha disminuido por la introducción de otros tipos de redes neuronales más complejas, su impacto y alcance todavía es muy amplio y relevante.





Tecmilenio no guarda relación alguna con las marcas mencionadas como ejemplo. Las marcas son propiedad de sus titulares conforme a la legislación aplicable, estas se utilizan con fines académicos y didácticos, por lo que no existen fines de lucro, relación publicitaria o de patrocinio.

Todos los derechos reservados @ Universidad Tecmilenio

La obra presentada es propiedad de ENSEÑANZA E INVESTIGACIÓN SUPERIOR A.C. (UNIVERSIDAD TECMILENIO), protegida por la Ley Federal de Derecho de Autor; la alteración o deformación de una obra, así como su reproducción, exhibición o ejecución pública sin el consentimiento de su autor y titular de los derechos correspondientes es constitutivo de un delito tipificado en la Ley Federal de Derechos de Autor, así como en las Leyes Internacionales de Derecho de Autor. El uso de imágenes, fragmentos de videos, fragmentos de eventos culturales, programas y demás material que sea objeto de protección de los derechos de autor, es exclusivamente para fines educativos e informativos, y cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por UNIVERSIDAD TECMILENIO. Queda prohibido copiar, reproducir, distribuir, publicar, transmitir, difundir, o en cualquier modo explotar cualquier parte de esta obra sin la autorización previa por escrito de UNIVERSIDAD TECMILENIO. Sin embargo, usted podrá bajar material a su computadora personal para uso exclusivamente personal o educacional y no comercial limitado a una copia por página. No se podrá remover o alterar de la copia ninguna leyenda de Derechos de Autor o la que manifieste la autoría del material.

