# Validación de Clusters usando IEKA y SL-SOM

Alessandro Bokan Garay[2] Raquel Patiño Escarcina [12] Yván Túpac Valdivia [1]

[1] Sociedad Peruana de Computación

[2]Universidad Católica San Pablo

alessandro.bokan@ucsp.edu.pe, rpatino@ucsp.edu.pe, ytupac@ucsp.edu.pe

**Resumen**

*El análisis de cluster busca agrupar un conjunto de objetos de características similares en grupos denominados clusters. El problema de clustering consiste en calcular el número de clusters formados a partir de un conjunto de datos. Una solución a este problema es el método IEKA (Intelligent Evolutionary K-means Algorithm), el cual busca optimizar los centros de los clusters usando un Algoritmo Genético de Codificación Real (Real-Coded Genetic Algorithm), aplicando índices de validación de clusters como función de aptitud del Algoritmo Genético, de esta manera, es posible determinar la mejor clusterización. Los Mapas Auto-organizados Auto-etiquetados (SL-SOM) son un método de aprendizaje no-supervisado que también resuelven el problema de clustering, etiquetando una red previamente entrenada (SOM) con los datos del conjunto. En este paper se realiza una validación de clusters entre estos dos métodos, aplicados en distintas bases de datos seleccionadas; donde se demuestra que el método IEKA resultó ser un método de clusterización superior, respecto a la compactéz y separación entre clusters, a comparación del método SL-SOM.*

## 1. Introducción

El análisis de cluster consiste en dividir un conjunto de datos en grupos denominados clusters, los cuales se caracterizan por contener datos significativos, es decir, datos que guardan relación entre sí. La técnica de clustering tiene como objetivo capturar la estructura natural de los objetos, de esta manera, poder dividirla en grupos que poseean cierta información en común. Hoy en día, la técnica de clustering se aplica en distintos campos: en la economía, como la segmentación del mercado (Dolnicar and Leisch, 2003); en la biología, en el agrupamiento de genes significativos (Yeung et al., 2003); y en la medicina, como en la segmentación de imágenes cerebrales mediante resonancia magnética (Hall et al., 1992). Se aplica además en áreas representativas de la computación, como Recuperación de Información (*Information Retrieval*) (Manning et al., 2008) y Minería de Datos (*Data Mining*) (Han. J., 2006). Se puede hablar de cinco categorias de clustering en la actualidad: clustering jerárquico, clustering particional, clustering basado en densidad, clustering basado en cuadrillas y clustering basado en modelos (Tan P., 2006).

Los Algoritmos Genéticos (AG) tienen como objetivo resolver problemas complejos de optimización. Su finalidad es proveer las mejores soluciones en términos del objetivo de la función utilizada, hasta converger a una sola y más optima solución para el problema (Bandyopadhyay and Maulik, 2002). El enfoque de un AG se basa en la *Teoría de Darwin*, que habla sobre la selección natural y la supervivencia del más fuerte. Siguiendo esta teoría, se dice que las especies con mejor aptitud (*fitness*) sobrevivirán para dejar ancestros y continuar poblando la tierra (Goldberg and Richardson, 1987) (Coley, 1999). Un artículo reciente ha desarrollado un algoritmo para realizar la clusterización más optima de un conjunto de datos, al que denominaron

IEKA (*Intelligent Evolutionary K-means Algorithm*) (Tseng et al., 2010). El procedimiento del algoritmo IEKA es el de un AG con mejoras. El objetivo de este algoritmo es modificar directamente los centroides de los clusters en cada iteración mediante la técnica de clusterización *k-means*, y así, hallar la mejor clusterización. Para lograr este objetivo, este método plantea evaluar por medio del AG, los clusters obtenidos mediante una función de aptitud usando índices de validación.

Los Mapas Auto-organizados Auto-etiquetados o *Self-Labeling Self-Organizing Map* (SL-SOM) se encuentran dentro de los métodos de clustering no-supervisados. Este método se caracteriza por generar grupos sin información *a priori*, es decir, no se fija el número de clusters que se deben obtener. Esta técnica busca etiquetar una red neuronal previamente entrenada mediante el método de kohonen, denominada *Self-Organized Map* (SOM). Una vez obtenida la red neuronal entrenada, se halla la matriz de distancias entre las neuronas, denominada *U-matrix* (Costa, 1999), por la cual se empezará el proceso de etiquetado, cuya finalidad es determinar el número de clusters obtenidos en el proceso de clusterización.

El objetivo de este trabajo es realizar una validación del número de clusters obtenidos mediante el método de IEKA y SL-SOM, para determinar la mejor clusterización de un conjunto de datos determinado. Para lograr esto, se ha realizado una variación al método IEKA, donde se usará un AG de Cromosomas de Longitud Variable (Zebulum, 2002) que se datalla a continuación.

El presente artículo está organizado de la siguiente manera: En la sección 2, se habla detalladamente sobre el algoritmo IEKA, donde se expone el algoritmo k-means junto con la representación del cromosma de longitud variable y la función de aptitud (índices) en el AG. En la sección 3, se explica detalladamente la SL-SOM. En la sección 4 se presentan los experimentos y resultados obtenidos en este estudio. En la sección 5, se exponen las conclusiones a las que se ha llegado mediante este estudio. Por último, se adjuntan las referencias en las que se basa este trabajo.

## 2. Intelligent Evolutionary K-means Algorithm (IEKA)

El método IEKA es un método de clusterización no-supervisado que se basa en la convergencia a una solución óptima que pueden generar los Algoritmos Genéticos (Tseng et al., 2010). Existen dos tipos AG de acuerdo a la representación del cromosoma según los valores de sus genes: AG de Codificación Binaria o *Binary-Coded Genetic Algorithm* (BGA) y los AG de Codificación Real o *Real-Coded Genetic Algorithm* (RGA) (Tang and Tseng, 2009b). Como el método IEKA modifica directamente los centros de clusters, entonces, se debe aplicar un RGA al problema.

Este método consiste en generar cromosomas que posean $k$ centros de $k$ clusters, que representarán a los genes de dichos cromosomas; los cuales serán recalculados mediante la técnica de clusterización *k-means* (Kumar et al., 2008). De esta manera, se genera una población inicial donde cada cromosoma poseerá $k$ centroides aleatorios. Luego, una vez calculados los clusters por cada cromosoma mediante k-means, se pasa a evaluar el grado de aceptación del cluster mediante los índices de validación; procediendo, luego, a realizar las operaciones elementales que forman la estructura de un AG. La clave de esta metodología, es la incorporación de tres

estrategias que atribuyen una mejora al AG: multi-elitismos (Tang and Tseng, 2009a), cruzamiento FFD (Ho et al., 2009) y mutación adaptativa (Srinivas and Patnaik, 1994). El objetivo de este algoritmo es modificar directamente los centroides de los clusters en cada iteración del AG, para luego evaluarlos (validarlos) y obtener, así, la clusterización más optima. Basándonos en esta metodología, usaremos *Cromosomas de Longitud Variable* (Zebulum, 2002), que representarán dichos centros de clusters, los cuales varían de tamaño conforme el AG vaya evolucionando. Además, de igual manera, se aplicará la técnica de clusterización k-means para tales cromosomas, que se evaluarán mediante índices de validación.

### 2.1. K-means

Este método busca particionar un conjunto de datos a partir de un número de clusters determinado (centros de clusters), denominado $k$. La clusterización mediante el método k-means puede ser descrita formalmente por el siguiente algoritmo (Tan P., 2006):

1. *Generar $k$ puntos aleatorios que representarán los centroides o centros de los $k$ primeros clusters.*
2. *Repetir*
3. *Cada dato del conjunto se agrupa al centro más cercano.*
4. *Re-calcular los centros de cada cluster.*
5. *Hasta que los centros no varíen.*

### 2.2. Representación del Cromosoma

La representación del cromosoma será de longitud variable, es decir, el tamaño del cromosoma será dinámico. Cabe resaltar, que como parte de la representación del cromosoma, también está el tipo de dato de cada gen, en este caso, los genes serán de tipo punto flotante, representando los centros de clusters. Por ende, el método IEKA se basa en un Algoritmo Genético de Codificación Real (RGA).

Cada individuo de la población tendrá el mismo tamaño o *longitud virtual* y se representará por medio de dos cadenas: la cadena genotipo (*genotype string*) y la cadena máscara de activación (*activation mask string*). En la primera, se encuentran los valores reales del cromosoma (o genes), en nuestro caso, los centros de clusters. La segunda, es una cadena compuesta de 1's y 0's, donde 1 representa los genes activos y 0, los inactivos (Zebulum, 2002). Cabe resaltar que las dos cadenas son de igual longitud. Por tanto, cada gen que se encuentre en estado de activación, activará también al gen de la cadena genotipo que se encuentra en la misma posición. Entonces, se puede decir, que la *longitud real* del cromosoma es la suma de todos los genes activos. A medida que el AG va evolucionando, los genes dentro de la máscara de activación irán variando, por ende, el número $k$ de centros también va a variar. Esta representación se puede ver claramente en la figura 1.

La máscara de activación es primordial para la convergencia prematura del AG. Los operadores de cruzamiento y mutación del AG desempeñan también una función importante en

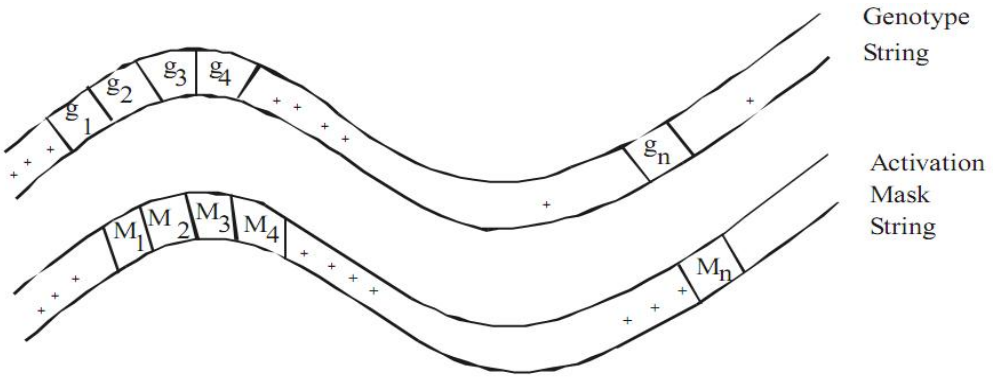


Figura 1: Representación del Cromosoma de Longitud Variable

la convergencia de este tipo de cromosomas. Dado que la mutación consiste en simplemente mutar o cambiar un gen al azar dentro del cromosoma seleccionado, entonces no es necesario mutar también la cadena adjunta (máscara). Sin embargo, para el cruzamiento, sí es necesario modificar también la máscara de activación, ya que, de esta manera se puede incrementar o decrementar la longitud del cromosoma, y así, determinar su convergencia de manera más rápida. Para este trabajo, se usará la técnica simple de cruzamiento denominada, cruzamiento de dos puntos (Lee and Antonsson, 2000).

### 2.3. Función de Aptitud

La función de aptitud dentro de un AG busca determinar los individuos que pasarán a la siguiente población o generación. Como el objetivo del método IEKA busca encontrar la mejor clusterización, entonces, esto se puede evaluar mediante los *índices de validación*, aplicados en el AG como función de aptitud. Por tanto, para determinar el número optimo de clusters en un conjunto de datos dado, se proponen dos criterios de validación de cluster, que evaluan cuantitativamente la exactitud y la calidad de una estructura de clustering. Tales índices son:

- **Índice de Dunn:** Esta técnica consiste en verificar que los conjuntos de clusters sean compactos y bien separados. Para cualquier partición de clusters, donde $X_i$ representa el i-ésimo cluster de tal partición. El Índice de Dunn se define matemáticamente como:

$$DI = \frac{\min\limits_{1\leq i\leq n_c}\left\{\min\limits_{1\leq j\leq n_c, i\neq j}\{dist(X_i, X_j)\}\right\}}{\max\limits_{1\leq k\leq n_c}\{diam(X_k)\}} \tag{1}$$

  donde $n_c =$ número de clusters, $dist(X_i, X_j) =$ distancia entre dos clusters, y $diam(X_k) =$ máxima distancia entre los elementos de un cluster $k$ (Dunn, 1974).

- **Índice de Davies-Bouldin:** Es una función de la proporción de la suma de la dispersión dentro del cluster a la separación entre clusters. Matemáticamente, el

Índice de Davies-Bouldin se define de la siguiente manera:

$$DB = \frac{1}{n_c} \sum_{i=1}^{n} R_i \tag{2}$$

donde $R_i = \underset{1 \leq i \leq n_c, i \neq j}{max} R_{ij}$, $R_{ij} = (S_i + S_j)/d_{ij}$, $S_i =$ distancia máxima entre los centros de cluster $i$ y $j$ (Davies and Bouldin, 1979).

Este artículo busca presentar una dinámica en la estructura del Algoritmo Genético. El procedimiento del AG es el siguiente:

1. *Representación: Cromosoma de Longitud Variable.*
2. *Inicialización de la Población: Centros y genes activos aleatorios.*
3. *Clusterización mediante el algoritmo k-means.*
4. *Elitismo.*
5. *Selección por Ruleta.*
6. *Cruzamiento de dos puntos.*
7. *Mutación Uniforme: inicialización aleatoria del cromosoma seleccionado al azar.*
8. *Reemplazo.*
9. *Verificar convergencia.*

## 3. Self-Labeling Self-Organizing Map (SL-SOM)

SL-SOM es un método de clusterización no-supervisado basado en la segmentación de una imágen por medio de la gradiente, que busca determinar el número de clusters obtenidos por medio del proceso de clusterización de un conjunto de datos. Para realizar este cometido, en primer lugar se aplica un Mapa Auto-Organizado (SOM) o red de Kohonen al conjunto de datos que se desee clusterizar. Tales mapas son de tamaño $M$x$N$ y constan de 2 capas: entrada y salida. Las entradas del mapa se representan mediante un espacio *p-dimensional*, donde $R^p$, denominados *patrones*. Por tanto, cada neurona $i$, al igual que los patrones de entrada, se representa como $n_i = [n_{i1}, n_{i2}, n_{i3}, ..., n_{ip}]^T$. Cada neurona del mapa está conectada con las neuronas adyacentes (vecinas), de esta manera, cuando se realice el respectivo entrenamiento de la red neuronal, las neuronas activadas se actualizarán al igual que su vecindad. Al final del entrenamiento, las neuronas deben aparecer cercanas a los patrones de entrada de forma proporcional, formándose los respectivos clusters (Costa, 1999).

En segundo lugar, una vez entrenada nuestra red neuronal, se halla la matriz de distancias entre las neuronas vecinas denominada *U-matrix*. Dicha matriz es de tamaño $(2M-1, 2N-1)$. También se le conoce como un método de visualización de la SOM (Ultsch, 1993), con el objetivo de detectar visualmente las relaciones topológicas entre las neuronas. Esta matriz suele ser usada para la segmentación de imágenes, que consiste en subdividir la imágen en partes u objetos constituyentes. El gran problema que presenta la U-matrix son los agrupamientos de

neuronas que están demasiado cerca y pueden causar una degradación de bordes, lo cual ocaciona que la segmentación sea compleja. En definitiva, al igual que una imagen, la U-matrix busca segmentar la red neuronal aplicando 2 tipos de técnicas: crecimiento de regiones y extracción de contornos (Parker, 1997).

El método más simple de segmentar una imagen se da través de la *umbralización*, donde se escoge un valor entre la distancia máxima y mínima de la U-matrix, usado para binarizar la imágen. Sin embargo, este método conduce a resultados insatisfactórios, pues generálmente este tipo de imágenes poseen un histograma complejo y ruidoso. Un método eficiente para segmentar la U-matrix es la segmentación *watershed* (S. Beucher, 1979).

Para segmentar una imagen, o en este caso, una red neuronal, es necesario determinar *marcadores*, los cuales son conjuntos de neuronas a partir de las cuales se realiza el crecimiento de regiones o la extracción de contornos. Para esto, primero se debe realizar un previo filtrado de la imágen eliminando el posible ruido. Luego se determina el número de regiones conectadas mediante un proceso de umbralización denominado *dendograma*; donde se va probando una cantidad de umbrales que binarizarán la U-matrix, y a la vez se va detectando el número de regiones correspondientes para cada umbral. Al final, la mayor secuencia continua y constante del número de posibles regiones, representará los marcadores (Costa, 1999). Esto se puede apreciar en la figura 2.

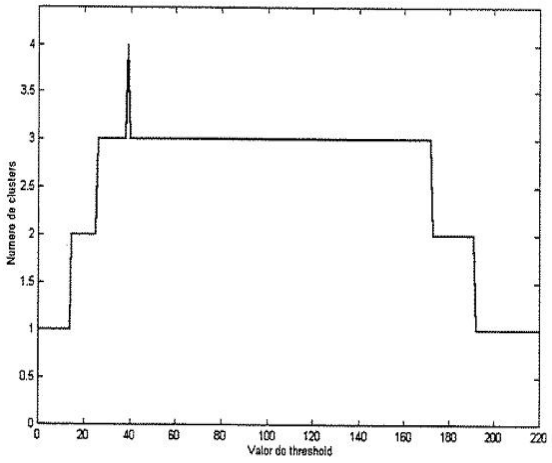


Figura 2: Número de posibles regiones (clusters) en función del umbral (threshold)

El algoritmo de la SL-SOM, a partir de una red neuronal entrenada, se describe a continuación:

1. *Obtención de la U-matrix.*
2. *Determinar los marcadores para la U-matrix*
3. *Aplicar el crecimiento de regiones sobre la U-matrix usando los marcadores obtenidos en el paso 2.*
4. *Etiquetado de regiones conectadas a la red neuronal segmentada en el paso 3.*
5. *Etiquetar los patrones de entrada de acuerdo a sus respectivas neuronas etiquetadas en el paso 4.*

## 4. Experimentos y Resultados

En este trabajo se realiza un análisis de clusters de 2 tipos de bases de datos: Iris y Sintética. *La BD Iris* está conformada por 150 instancias de plantas, donde cada una posee 4 atributos: largo del sépalo, ancho del sépalo, largo del pétalo y ancho del pétalo; consta de 3 clases, cada una de 50 instancias, tales clases son: Iris Setosa, Iris Versicolour e Iris Virginica. *La BD Sintética* ha sido creada automáticamente, la cual está conformada por 300 instancias, donde cada una posee 2 atributos; consta de 3 clases, de 100 instancias cada una . Acontinuación se presentan las gráficas de las bases de datos con las cuales se realizaron los experimentos. Para el conjunto de datos *Iris*, solamente se tomaron los 3 primeros atributos para generar una gráfica tridimensional.

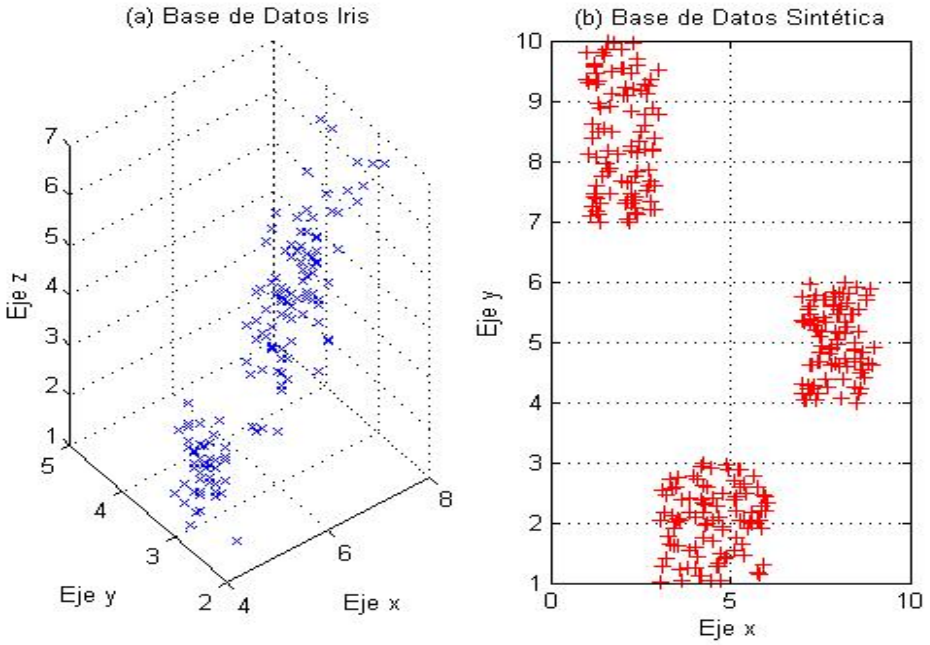


Figura 3: (a) Gráfica tridimensional BD IRIS (b) Gráfica bidimensional BD Sintética

A tales BD's se aplicaron los métodos previstos anteriormente, obteniendo resultados que se visualizarán en adelante. Para la *BD Iris*, el método IEKA con Representación de Longitud Variable determinó 2 clusters: uno de 100 instancias, y el otro de 50 instancias. El método SL-SOM determinó, también, 2 clusters: uno de 114 instancias, y el otro de 36 instancias. Esto se puede apreciar en la figura 4. Para la *BD Sintética*, el método IEKA determinó 3 clusters, cada uno con 100 instancias. Por otro lado, el método SL-SOM determinó solamente 2 clusters: uno de 267 instancias, y el otro de 33 instancias. Esto se puede ver claramente en la figura 5. Para obtener los resultados se ha ejecutado 200 veces los 2 algoritmos para cada una de las bases de datos correspondientes. Los cálculos realizados en el método IEKA se han fijado en el cuadro 1.

Respecto a la primera BD analizada (Iris), se puede decir que los dos métodos aplicados dan como resultado un número igual de clusters encontrados. Sin embargo, como se puede apreciar en la figura 4, el método IEKA determina dos clusters que están bien separados, a comparación

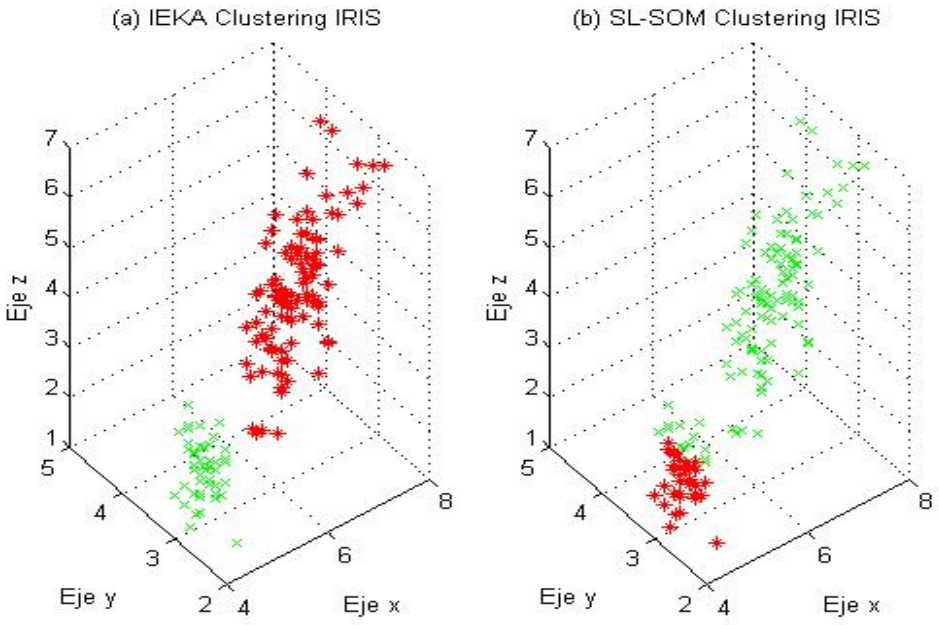


Figura 4: (a) Gráfica 3D IEKA Clustering Iris (b) Gráfica 3D SL-SOM Clustering Iris

del método SL-SOM, donde los clusters se encuentran muy unidos. Cabe resaltar, que los dos métodos de validación de cluster no determinaron un total de 3 clusters, que representan las 3 clases pertenecientes a la BD Iris, mencionadas anteriormente.

Respecto a la segunda BD analizada, el método SL-SOM presenta el problema de solapamiento entre clusters, determinando solamente 2 clusters. En cambio, el método IEKA encontró 3 clusters de 100 instancias cada uno; ver figura 5.

| **Cálculos en IEKA** | | | | |
|---|---|---|---|---|
| Bases de Datos | Iteraciones | Ciclos AG | Índice Dunn | Clusters |
| IRIS | 200 | 20 | 0.8947 | 2 |
| SINTÉTICA | 200 | 40 | 1.2347 | 3 |

Cuadro 1: Tabla de cálculos realizados con el Método IEKA

## 5. Conclusiones

Después de analizar los resultados de los experimentos realizados, se puede concluir que el método IEKA resultó ser más eficaz que el método SL-SOM, ya que, al realizar el proceso de clustering de las dos bases de datos propuestas (BD Iris y Sintética), IEKA generó una clusterización sin solapamientos, cuyos clusters encontrados se encuentran bien compactos y bien separados. Para demostrar esta afirmación, en la figura 3, se vizualizan claramente las regiones insertadas en el espacio, tanto para el conjunto de datos del Iris, como para el Sintético. En cambio, el método SL-SOM no resultó ser muy exacto a la hora de determinar la cantidad de

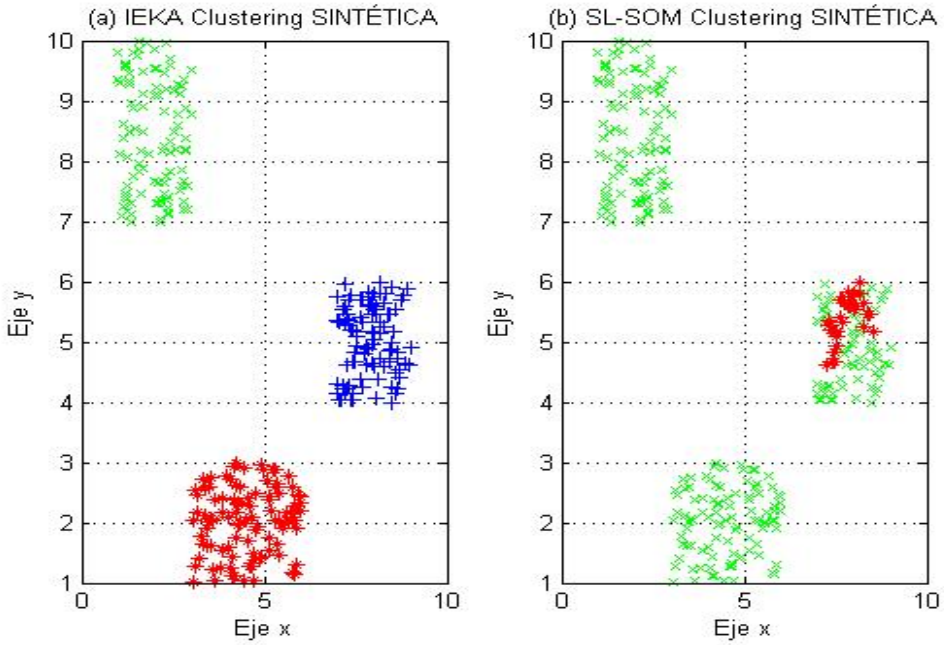


Figura 5: (a) Gráfica 2D IEKA Clustering Sintética (b) Gráfica 2D SL-SOM Clustering Sintética

clusters correspondientes, además de, presentar un caso de solapamiento entre clusters.

El método IEKA con Representación del Cromosoma de Longitud Variable resultó ser un método rápido y eficaz, al realizar una clusterización compacta, donde los clusters encontrados se encuentran bien separados. Esto se debe, a que el algoritmo llego a converger a un resultado óptimo en pocos ciclos, ver cuadro 1.

El método SL-SOM presentó un problema en la U-matrix al momento de determinar el umbral indicado mediante el dendograma, esta es la principal razón de un posible solapamiento entre clusters. Por tanto, para solucionar tal inconveniente debe existir una métrica que regule los umbrales del dendograma, para así determinar el número correcto de marcadores en la red neuronal. También es importante el algoritmo que realice el crecimiento de regiones una vez encontrado los marcadores. Esto evita que los clusters encontrados esten demasiado juntos.

## Referencias

Bandyopadhyay, S. and Maulik, U. (2002). Genetic clustering for automatic evolution of clusters and application to image classification. *Pattern Recognition*, 35(6):1197–1208.

Coley, D. A. (1999). *An introduction to Genetic Algorithms for Scientists and Engineers*. World Scientific, London.

Costa, J. F. (1999). *Clasifiçao Automática e Análise de Dados por Redes Neurais Auto-Organizávies*. PhD thesis, Departamento de Engenharia de Computaçao e Automaçao Industrial, Univ. Estadual de Campinas.

Davies, D. and Bouldin, W. (1979). A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(4):224–227.

Dolnicar, S. and Leisch, F. (2003). Winter tourist segments in austria: Identifying stable vacation styles using bagged clustering techniques. *Journal of Travel Research*, 41(3):281–292.

Dunn, J. C. (1974). Well separated clusters and optimal fuzzy partitions. *J. Cybern*, 4:95–104.

Goldberg, D. and Richardson, J. (1987). Genetic algorithms with sharing for multimodal function optimization. *Proceedings of the Second International Conference on Genetic Algorithms on Genetic algorithms and their application*, pages 41–49.

Hall, L., Bensaid, A., Clarke, L., Velthuizen, R., Silbiger, M., and Bezdek, J. (1992). A comparison of neural network and fuzzy clustering techniques in segmenting magnetic resonance images of the brain. *Neural Networks, IEEE Transactions On*, 3(5):672–682.

Han. J., K. M. (2006). *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers, Munich, Germany.

Ho, S. Y., Shu, L. S., and Chen, J. H. (2009). Intelligent evolutionary algorithms for large parameter optimization problems. *IEEE Trans. Evolutionary Computation*, 8:522–541.

Kumar, V., Hu, X., Quinlan, J., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G., Liu, A., Yu, P., Zhou, Z., Steinbach, M., Hand, D., and Steinberg, D. (2008). Top 10 algorithms in data mining. *Knowledge Information Systems*, 14:1–37.

Lee, C. L. and Antonsson, E. K. (2000). Variable length genome for evolutionary algorithms. *GECCO 2000 Springer*, pages 1–7.

Manning, D., Raghavan, P., and Schutze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press, Massachusetts, U.S.A.

Parker, J. R. (1997). *Algorithms for Image Processing and Computer Vision*. Wiley Computer Publisher, Canada.

S. Beucher, C. L. (1979). Use of watersheds in contour detection. *International Workshop on Image Processing: Real-time Edge and Motion Detection/Estimation, Rennes, France*, pages 1–12.

Srinivas, M. and Patnaik, L. M. (1994). Adaptive probabilities of crossover and mutation in genetic algorithms. *IEEE Transactions on Systems, Man and Cybernetics*, 24:656–667.

Tan P., Steinbach M., K. V. (2006). *Introduction to Data Mining*. Addison-Wesley Companion, Minnesota, U.S.A.

Tang, P. H. and Tseng, M. H. (2009a). Intelligent evolutionary algorithms for large parameter optimization problems. *National Computer Symposium, Taiwan*.

Tang, P. H. and Tseng, M. H. (2009b). Medical data mining using bga and rga for weighting of features in fuzzy k-nn classification. *Proceeding of the Eighth International Conference on Machine Learning and Cybernetics, Baoding*, 5:3070–3075.

Tseng, M., Chiang, C., Tang, P., and Wu, H. (2010). A study on cluster validity using intellignet evolutionary k-means approach. *Proceedings of the Ninth International Conference on Machine Learning and Cybernetics, Qingdao*, pages 2510–2515.

Ultsch, A. (1993). Self-organizing neural netoworks for visualization and classification. *Information and Classification, Springer, Berlin*, pages 307–313.

Yeung, K., Medvedovic, M., and Bumgarner, R. (2003). Clustering gene-expression data with repeated measurements. *Genome Biology*, 4(17).

Zebulum, R. S. (2002). *Evolutionary Electronics*. PhD thesis, The CRC Press International Series on Computational Intelligence.