

Sistema de enseñanza para la técnica de agrupamiento k-means Teaching system for the k-means clustering technique

A. Franco-Arcega ^{a,*}, V. I. Sobrevilla-Solís ^a, M. J. Gutiérrez-Sánchez ^a, L.H. García-Islas ^a, A. Suárez-Navarrete ^a,
E. Rueda-Soriano ^a

^a Universidad Autónoma del Estado de Hidalgo, 42184, Pachuca, Hidalgo, México.

Resumen

La tarea de agrupamiento es una de las técnicas de clasificación no supervisada más empleada en el procesamiento de datos, la cual tiene el objetivo de encontrar una partición idónea de un conjunto de datos. Uno de los algoritmos que más se ha aplicado en entornos reales y cotidianos es el conocido como algoritmo k-means. Existen diversas plataformas que permiten la aplicación de dicho algoritmo, pero el uso de estas plataformas es limitado, ya que el usuario no puede identificar los procesos que sigue el algoritmo para llegar al resultado final. Dado el uso del algoritmo k-means, con el fin de conocer su funcionamiento detallado y así saber en qué casos es conveniente usarlo, es de relevancia contar con una herramienta que muestre paso a paso cada una de las operaciones que realiza para cumplir su fin. Este trabajo presenta este sistema de enseñanza, el cual es una herramienta de apoyo que permite descubrir cómo se construye la partición de un conjunto de datos y de esta forma reforzar la enseñanza del algoritmo k-means.

Palabras Clave:

Agrupamiento, algoritmo k-means, sistema de enseñanza, herramienta computacional, reconocimiento de patrones.

Abstract

Clustering task is one of the most used unsupervised classification techniques in data processing, which has the aim of finding an ideal partition of a data set. One of the algorithms that has been applied the most in real and everyday fields is known as the k-means algorithm. There are several platforms that allow the application of this algorithm, but the use of these platforms is limited, since the user cannot identify the processes that the algorithm follows to reach the final result. Given the use of k-means, in order to show a detailed process and therefore known in which cases it is convenient to use it, it is important to have a tool that teaches step by step each one of the operations that it performs to reach its purpose. This work presents a teaching system, which works as a support tool that allows discovering how the partition of a data set is built in order to reinforce the learning of k-means algorithm.

Keywords:

Clustering, k-means algorithm, teaching system, computational system, pattern recognition.

1. Introducción

Actualmente, emplear algún software es muy común para solucionar diversos problemas cotidianos, ya que puede aplicarse en diferentes ámbitos tales como: la astronomía, en el aspecto financiero o en la medicina, por mencionar algunos. En particular, en el ámbito educativo, el uso de software ha proporcionado herramientas tecnológicas que, en conjunto con nóveles técnicas pedagógicas, han permitido reinventar la forma de enseñar (Marzocchi *et al.*, 2012) (Montserrat *et al.*, 2002) (Valencia & Charro, 2007). Anteriormente, los profesores empleaban estrategias de enseñanza simples, ya que la mayoría de ellos sólo exponía los temas usando lecciones magistrales, en donde el

alumno mantenía un estado pasivo, ya que sólo se limitaba a tomar notas en sus cuadernos. Ahora, con la tecnología al alcance de nuestras manos, este comportamiento ha cambiado; dado que los profesores se apoyan de diversas herramientas tecnológicas que refuerzan y se derivan en nuevos procesos enseñanza – aprendizaje. Por esta razón, ahora el alumno puede mantener un estado activo, ya que la interacción para poder adquirir cierto conocimiento en sus cursos es mayor.

La mayoría de las instituciones educativas, principalmente las de educación superior, capacitan a sus profesores para que puedan hacer un uso correcto de diversas herramientas tecnológicas, y de esta manera, logren emplearlas en la construcción de material educativo digital que les permita mejorar su enseñanza en las

*Autor para la correspondencia: afranco@uaeh.edu.mx

Correo electrónico: afranco@uaeh.edu.mx (Anilu Franco-Arcega), so314246@uaeh.edu.mx (Víctor Ignacio Sobrevilla-Solís), madejesus_gutierrez@uaeh.edu.mx (Ma. de Jesús Gutiérrez-Sánchez), luishg@uaeh.edu.mx (Luis Heriberto García-Islas), asuarez@uaeh.edu.mx (Alberto Suárez-Navarrete), estebanrs@uaeh.edu.mx (Esterban Rueda-Soriano).

diferentes asignaturas que imparten en sus programas educativos. En particular, aquellas asignaturas que presentan métodos computacionales para el procesamiento de datos, tales como Reconocimiento de Patrones o Minería de Datos, ya que en ellas es posible hacer uso de recursos digitales para que los alumnos puedan comprender algunos temas, especialmente aquellos relacionados a conocer el funcionamiento de algún método o algoritmo.

Estos algoritmos computacionales que se aprenden en dichas asignaturas ayudan al usuario a procesar conjuntos de datos para encontrar información valiosa (Han *et al.*, 2012). Cada algoritmo de procesamiento consiste en una serie de procesos que lo llevan a cumplir el objetivo de su aplicación (Murty and Devi, 2015). Este trabajo se centra en el algoritmo de agrupamiento k-means, el cual puede ser difícil de aprender si no se entienden las operaciones y el orden en que se ejecutan para llevar a cabo el proceso del algoritmo. Las operaciones matemáticas que aplica el algoritmo pueden mostrarse al alumno haciendo uso de un recurso didáctico que le permita visualizar en forma gráfica y detallada como se ejecutan.

Hoy en día existe software especializado que puede utilizarse para aplicar el algoritmo k-means (Frank *et al.*, 2016) (Matlab, 2018). Sin embargo, estos no muestran de manera detallada las operaciones necesarias para encontrar el resultado final, sólo se pueden utilizar para ejecutar el algoritmo y mostrar la partición final. Esto no puede considerarse un apoyo didáctico para los alumnos, por lo que en el presente trabajo se muestra el desarrollo de una herramienta computacional enfocada a reforzar la enseñanza del algoritmo k-means, mostrando paso a paso el procedimiento que el algoritmo hace para procesar conjuntos de datos que pueden graficarse en dos dimensiones; de esta manera, el usuario también puede ir visualizando este proceso.

2. Algoritmo k-means

El algoritmo k-means es un método de agrupamiento que trabaja conjunto de datos que son descritos por valores numéricos (Duda *et al.*, 2012) y es uno de los más populares algoritmos de agrupamiento en Minería de Datos (Wu and Kumar, 2009). Su objetivo es encontrar una partición de ese conjunto, la cual consiste en una serie de grupos que son representados por un centro. La Figura 1 muestra la descripción general del algoritmo. Para determinar el número de grupos a crear por el algoritmo, el usuario tiene que definir previamente el valor del parámetro k , el cual indica cuántos grupos se formarán en la partición resultante del conjunto de datos.

La idea de este algoritmo es ir afinando la posición de los centros representativos en el espacio de objetos, iteración a iteración hasta encontrar la mejor partición de los objetos, es decir, encontrar a los grupos que integren a los individuos más parecidos entre ellos.

2.1. Función de distancia

Con el fin de calcular la distancia que existe entre dos objetos distintos, y así identificar a los objetos más parecidos para crear los grupos, el algoritmo k-means puede emplear funciones de distancia. La distancia euclidiana es una de las más empleadas en la literatura y preferida en el proceso de este algoritmo (Yee *et al.*, 2019). La distancia euclidiana mide la distancia más corta entre dos puntos y se calcula usando (1).

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (1)$$

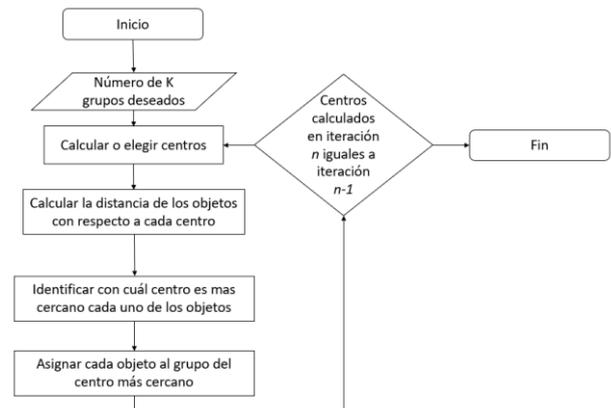


Figura 1: Descripción del algoritmo k-means.

La ecuación (1) indica que los objetos están descritos por i valores de atributos descriptores y que todos son considerados para poder obtener la distancia entre x y y . Entre más pequeño el valor de distancia más cercanos son los objetos comparados y por ende más similares.

2.2. Procedimiento del algoritmo k-means

Para llevar a cabo el proceso del algoritmo k-means se define a priori el número de grupos a formar. Considerando ese valor, se inicializan los centros que representarán a cada grupo. De forma inicial, los centros tienen que asegurar que al menos un objeto será asignado al grupo que representa y siguiendo un proceso iterativo, los objetos del conjunto de datos a procesar serán agrupados de la siguiente manera:

1. Calcular la distancia entre todos los objetos y los centros de cada grupo.
2. Identificar con cuál centro es más cercano cada uno de los objetos.
3. Asignar cada objeto al grupo del centro más cercano.
4. Recalcular los centros de los grupos considerando los objetos asignados a cada grupo. Para calcular el nuevo centro del grupo j , se obtiene la media de los objetos asignados a j .
5. Si los centros calculados en la iteración n son iguales a los centros de la iteración $n-1$, entonces la ejecución del algoritmo termina.

3. Metodología de desarrollo

Para el desarrollo del sistema de enseñanza propuesto en este trabajo se optó por utilizar una metodología de desarrollo ágil (Orjuela-Duarte and Rojas, 2008). La elegida fue la llamada XP, su nombre por sus siglas en inglés eXtreme Programming (Letelier and Penadés, 2006), la cual es una metodología que ofrece diversas ventajas sobre otros tipos de metodologías, como la adaptabilidad a los cambios, desarrollo exploratorio y menor burocracia (Pantaleo and Rinaudo, 2015). Su idea principal es ofrecer un ciclo de vida dinámico, en el cual se tienen iteraciones cortas que entregan resultados funcionales al final de cada una de ellas. En cada iteración de esta metodología se realiza un ciclo completo de análisis, diseño, desarrollo y pruebas, lo que lleva a que el desarrollo del proyecto sea vigilado en todo momento.

Cada etapa en XP cuenta con diferentes tareas que se tienen que realizar (Wells, 1999). Las etapas y sus respectivas actividades que se consideraron en este proyecto son:

- Análisis – requerimientos del proyecto, plan de entregas, plan de iteraciones, reuniones de seguimiento.
- Diseño – tarjetas CRC, funcionalidad, interfaz, soluciones spike.
- Desarrollo / codificación – uso de estándares, programación dirigida a pruebas, integraciones permanentes, ritmo sostenido.
- Pruebas – pruebas unitarias, detección y corrección de errores.

3.1. Análisis

En esta fase de planificación se llevó a cabo el análisis de requerimientos para conocer las particularidades que el sistema de enseñanza debía contener. Se estableció que para que este sistema sirviera como una herramienta de apoyo a la enseñanza de la técnica del algoritmo k-means, se debía crear un escenario didáctico que mostrara al usuario como el algoritmo se ejecuta, de acuerdo a su concepción.

Se planeó que este escenario didáctico mostrara la ejecución del algoritmo en dos formas: la primera es visualizando la descripción de los objetos contenidos en el conjunto de datos a procesar, así como las operaciones y procesos que lleva de manera secuencial para cumplir con su propósito. La segunda forma es representando de manera gráfica el comportamiento del algoritmo, mostrando tanto los objetos en su espacio dimensional como los centros que representan a los grupos formados por el algoritmo.

3.2. Diseño

Una vez que se obtuvieron los requerimientos del sistema y los planes de entregas e iteraciones se procedió a realizar el diseño. En esta etapa se crearon las tarjetas CRC donde se especificaron las actividades que realizará el sistema de enseñanza. La Figura 2 muestra las responsabilidades de cada clase propuesta y la interacción que existe entre estas clases. Este diseño permitió identificar y mapear las clases involucradas en el sistema.

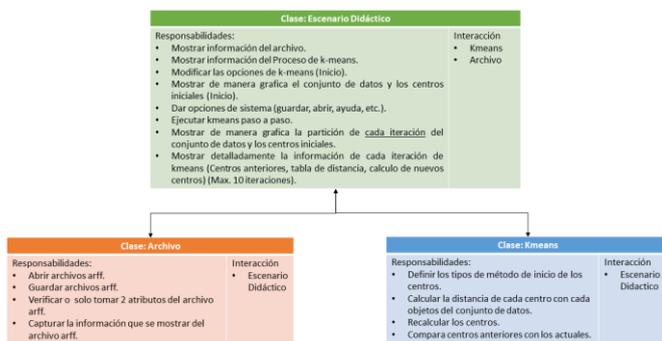


Figura 2: Tarjetas CRC de la etapa de diseño.

Posterior a la creación de las tarjetas CRC se llevó a cabo el diseño del funcionamiento interno del sistema, el cual está representado en el diagrama de flujo mostrado en la Figura 3.

En esta etapa también se diseñó la interfaz que permite al usuario identificar los elementos que interactúan en el proceso del algoritmo k-means. Se establece el uso de un panel interactivo que

permitirá mostrar las iteraciones que el algoritmo va ejecutando para crear la partición final del conjunto de datos usado. La Figura 4 presenta la forma general en cómo se visualiza esta interfaz.

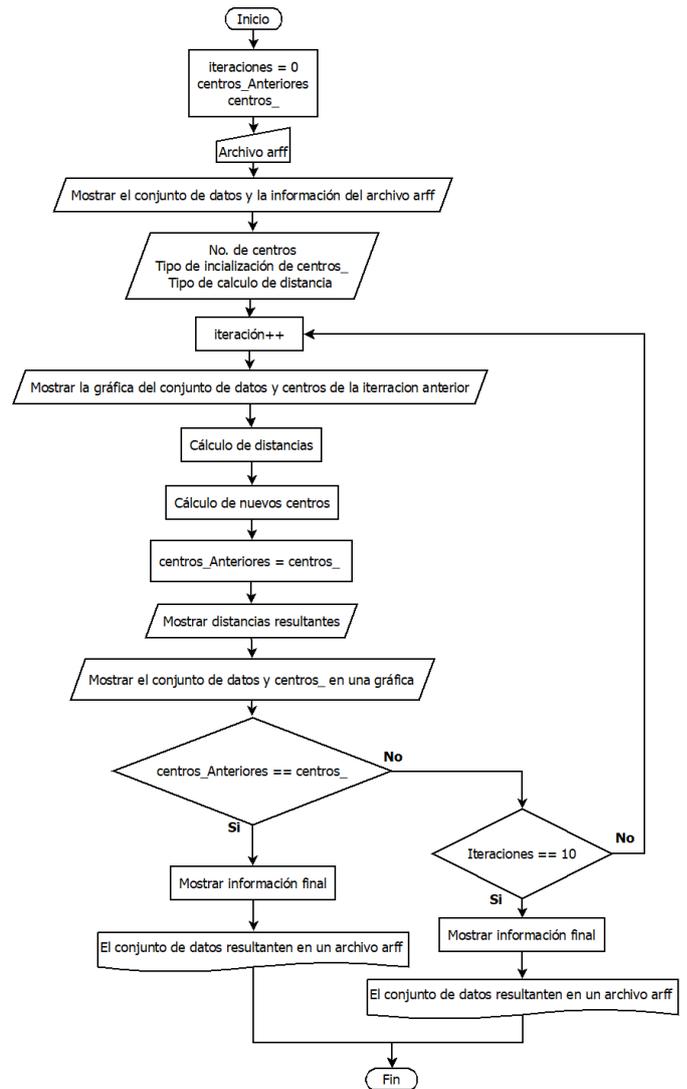


Figura 3: Diagrama del proceso del escenario didáctico.

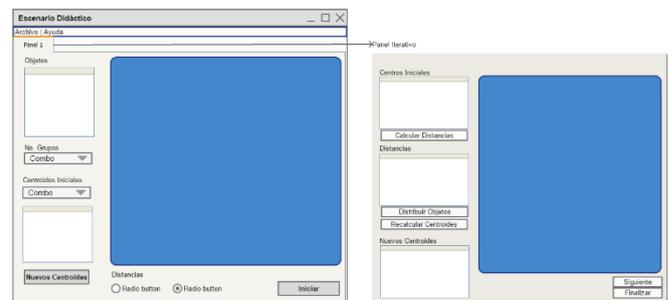


Figura 4: Diseño de la interfaz del sistema de enseñanza.

3.3. Desarrollo y codificación

Para el desarrollo del sistema se empleó el lenguaje de programación de Java 8, donde se programó el algoritmo k-means definiendo las clases especificadas en la etapa previa, las cuales permiten ejecutar el proceso del algoritmo k-means paso a paso,

es decir operación a operación. Además, se utilizaron las librerías Swing para la interfaz de usuario y la librería JFreeChart para mostrar los objetos en un plano cartesiano. Esta última librería permitió emplear funciones que permiten generar gráficas que muestran de manera visual el contenido del conjunto de datos que se está procesando, así como el comportamiento que tiene conforme las iteraciones del algoritmo avanzan.

3.4. Pruebas

Finalmente, se realizaron pruebas unitarias ejecutando el sistema haciendo uso de diferentes conjuntos de datos pequeños de dos dimensiones, con la finalidad de probar cada función y así detectar los errores, de tal manera que pudieron ser corregidos.

4. Sistema de enseñanza propuesto

El sistema de enseñanza propuesto inicia con una pantalla que le proporciona al usuario diversas opciones. La Figura 5 muestra esta pantalla, donde el botón *Real* le permite al usuario ejecutar el algoritmo k-means sobre conjuntos de datos grandes, mostrando únicamente el resultado final, no el proceso detallado. Para esta ejecución se empleó el Api de Weka para el procesamiento de conjuntos de gran volumen. Por su parte, el botón *Didáctico* le permite al usuario ejecutar el algoritmo para procesar, a nivel de iteración y con mayor grado de detalle, un conjunto de datos pequeño de dos dimensiones, con la finalidad de que pueda graficarse y por lo tanto se pueda visualizar el proceso paso a paso también de forma gráfica. El botón *Información* sólo contiene los datos generales de la versión del software y finalmente, el botón *Manual de Usuario* permite abrir un archivo en formato pdf que explica detalladamente el uso del sistema.

Cuando el usuario elige la opción *Didáctico*, el sistema pide abrir el archivo del conjunto de datos a procesar empleando un cuadro de diálogo de *Abrir Archivo*, donde el usuario podrá elegir el conjunto a usar desde alguna carpeta de origen, con formato ARFF. Una vez elegido el archivo, se abrirá una pantalla donde se muestra la información del conjunto de datos, es decir la descripción de los objetos que lo integran a través de sus dos dimensiones.



Figura 5: Pantalla inicial.

Posterior a la visualización de la información, el usuario elige el número de grupos a crear por el algoritmo. De acuerdo a esta selección, el sistema pide la opción de inicialización de los centros por grupo y los muestra para que el usuario los identifique. Como se puede observar en la Figura 6, del lado derecho de la pantalla se grafican tanto los objetos del conjunto de datos dentro de un

plano cartesiano, como los centros con los que iniciará la ejecución el algoritmo. Finalmente, para iniciar las iteraciones del algoritmo, el usuario tiene que elegir el tipo de distancia que usará k-means para medir la cercanía entre objetos – centros.

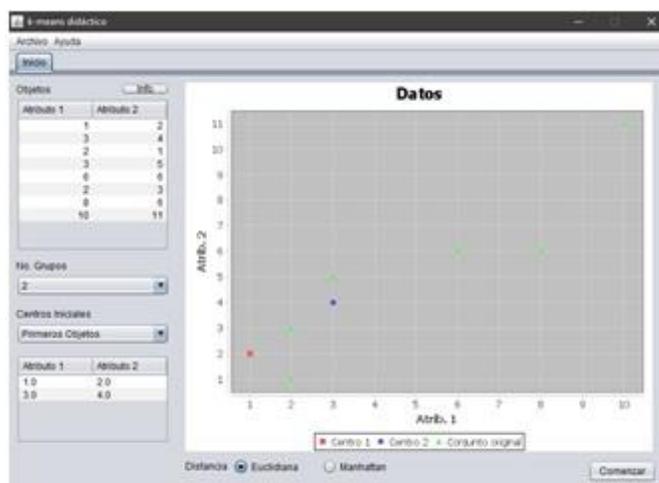


Figura 6: Configuración inicial del algoritmo.

A continuación, el algoritmo comienza a iterar, y en cada una de estas iteraciones, muestra de manera detallada al usuario el proceso que sigue el algoritmo k-means. En la Figura 7, se muestran los elementos que contiene, los cuales son:

- Área de visualización de los centros iniciales en la iteración.
- Botón de *Calcular Distancias*. Ejecuta el cálculo de distancias entre los centros y los objetos del conjunto de datos.
- Área de visualización de distancias: Muestra la distancias que obtuvo el proceso anterior.
- Botón de *Distribuir Objetos*: En el área del gráfico, muestra los grupos formados en esa iteración. A cada objeto le asigna un color de grupo, de acuerdo a su cercanía con los centros de los grupos.
- Botón de *Recalcular Centros*: Ejecuta la operación de recalcular el valor del centro de cada grupo.
- Área de visualización de los nuevos centros.
- Botón de *Siguiente Iteración*.
- Gráfico: Área donde se grafican los objetos del conjunto de datos y sus centros.

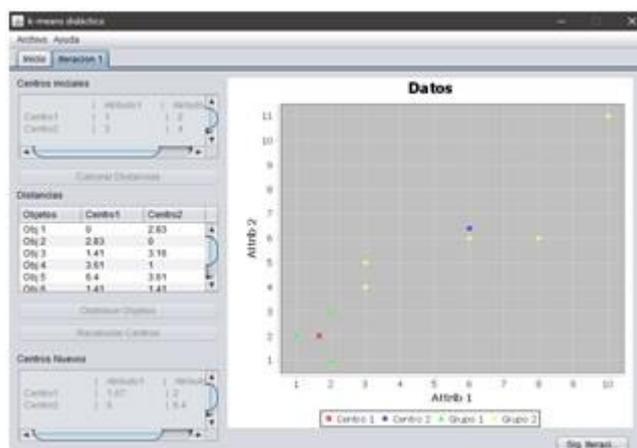


Figura 7: Pantalla de iteraciones del algoritmo.

El algoritmo va creando las iteraciones correspondientes y el usuario puede visualizar de cada una de ellas la información correspondiente, además de que puede navegar entre las pestañas de *Iteraciones* para que analice los cambios en la partición que se va generando en cada una.

Cuando la ejecución del algoritmo termina, porque los nuevos centros son iguales a los centros iniciales de la iteración, el sistema avisa al usuario a través de una ventana emergente que el proceso ha terminado e indica en qué iteración ocurrió tal hecho. Un ejemplo de esta acción se muestra en la Figura 8. Como actividad final, el usuario puede guardar el resultado del algoritmo, es decir, la partición creada por el algoritmo k-means.

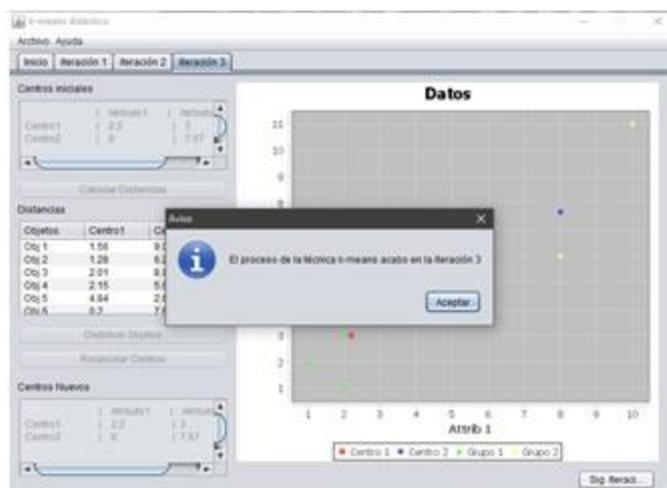


Figura 8: Fin de la ejecución del algoritmo.

5. Validación del sistema propuesto

El sistema de enseñanza de la técnica de agrupamiento k-means se le proporcionó a un grupo de 36 alumnos, los cuales estaban en el proceso de aprendizaje del algoritmo y lo conocían de manera teórica, fueron organizados en subgrupos de 5 o 6 personas para una mejor validación del sistema.

La sesión de pruebas consistió en dos etapas, en la primera se les explicó de manera general el propósito del sistema y la forma en cómo funciona. Posterior a esto, los alumnos interactuaron con el sistema procesando al menos dos conjuntos de datos en el entorno didáctico. Una vez terminada esta interacción, se les aplicó a los alumnos un cuestionario empleando la herramienta de formularios de Google. Los apartados a evaluar fueron *Aspectos técnicos del sistema*, *Aportes al aprendizaje* de los alumnos y *Opiniones del usuario*. Los primeros dos apartados de este cuestionario fueron diseñados con preguntas cuyas respuestas se validan a través de una escala de Likert (Joshi *et al.*, 2015), mientras que el tercer apartado fue con preguntas binarias.

Con respecto a los aspectos técnicos del sistema, se les pidió evaluar los siguientes rubros:

1. Diseño general de las pantallas.
2. Implementación de elementos como ventanas, gráficos, botones, tipos de letra, entre otros.
3. Interactividad en el sistema.
4. Secuenciación de las pantallas.
5. Navegación en la interfaz.
6. Implementación de teclas rápidas.

La Tabla 1 y la Figura 9 muestran el número de alumnos que eligieron una opción a cada una de las preguntas planteadas. Las opciones son (a) Muy de acuerdo, (b) De acuerdo, (c) Indiferente, (d) En desacuerdo, (e) Muy en desacuerdo. Como se puede apreciar, las opciones de “De acuerdo” y “Muy de acuerdo” obtuvieron los mayores porcentajes, lo cual indica que el sistema les pareció adecuado y amigable a los usuarios.

Tabla 1: Resultados de los aspectos técnicos del sistema

Pregunta	a	b	c	d	e
1	10	23	3	0	0
2	14	12	8	2	0
3	10	19	4	3	0
4	9	21	5	1	0
5	7	24	3	2	0
6	18	11	6	1	0
Promedio	11.33	18.33	4.83	1.5	0

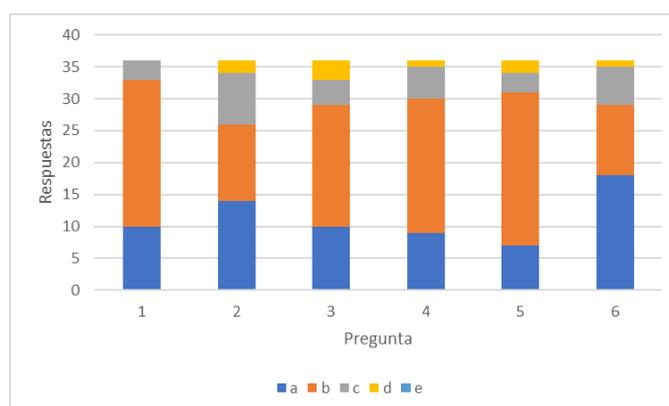


Figura 9: Resultados gráficos de los aspectos técnicos del sistema.

En cuanto a los aportes a su aprendizaje, los alumnos evaluaron los siguientes aspectos:

1. Mostrar paso a paso el proceso del algoritmo k-means refuerza tu aprendizaje del mismo.
2. Forma en que se presentó el proceso de k-means.
3. Presentación del resultado.
4. Elementos mostrados para entender el agrupamiento formado por el algoritmo.

En la Tabla 2 y en la Figura 10 se observa la percepción de los alumnos de acuerdo a las preguntas planteadas. Las opciones de respuesta son (a) Siempre, (b) Generalmente, (c) Ocasionalmente, (d) Nunca y (e) No observado. Para este aspecto, se observa un comportamiento similar al anterior, lo que muestra que la forma en cómo se presentó el algoritmo es aceptada por los usuarios.

Tabla 2: Resultados de aportes al aprendizaje

Pregunta	a	b	c	d	e
1	22	11	3	0	0
2	19	14	3	0	0
3	21	11	3	0	1
4	23	9	4	0	0
Promedio	21.25	11.25	3.25	0	0.25

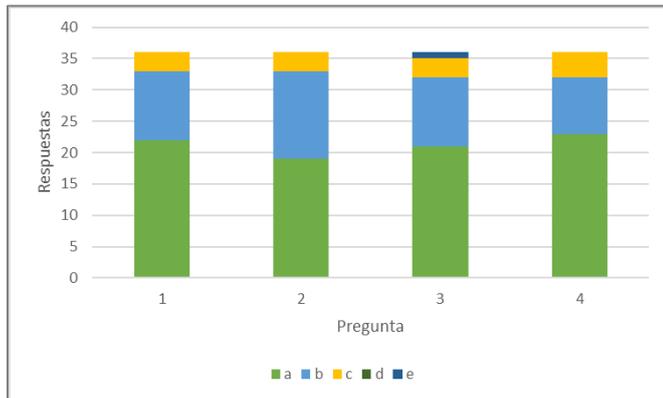


Figura 10: Resultados gráficos de los aportes al aprendizaje.

Finalmente, en las opiniones del usuario se incluyeron dos cuestionamientos y un apartado donde el evaluador puede poner en formato libre su opinión y sugerencias para el sistema. Las dos preguntas en este apartado refieren a lo siguiente:

1. El sistema ha despertado interés en usted para hacer uso de esta técnica en aplicaciones reales.
2. ¿Considera que el sistema es intuitivo?

Ambas preguntas de este apartado tuvieron respuestas de Si/No. Los resultados obtenidos se muestran en la Tabla 3.

Tabla 3: Resultados de opiniones del usuario

Pregunta	Si	No
1	27	9
2	35	1
Promedio	31	5

Considerando los aspectos positivos de cada apartado del cuestionario de validación del sistema, en el de aspectos técnicos del sistema hay una aceptación por parte de los alumnos del 82.38%. Con respecto al apartado de aporte al aprendizaje de los usuarios, se tiene un 90.27% de aceptación y del apartado de opiniones del usuario un 86.11%. De acuerdo a estos tres resultados, en general el sistema tiene un 86.25% de calificación por parte de los alumnos que interactuaron con el sistema.

6. Conclusiones

Dentro del proceso enseñanza – aprendizaje ha resultado eficiente hacer uso de recursos digitales con el fin de poder

mejorar la comprensión e interacción de los alumnos en los temas abordados en sus asignaturas. El sistema de enseñanza propuesto en este trabajo permite que sus usuarios refuercen el aprendizaje de la técnica de clasificación k-means, dado que les permite visualizar el proceso que se ejecuta para construir la partición resultante de un conjunto de datos determinado y de esta manera comprobar lo aprendido de manera teórica. Los resultados muestran que el sistema obtuvo una calificación general importante, por lo que se puede recomendar su uso dentro del aula.

Referencias

- Duda, R.O., Hart, P.E., Stork, D.G., (2012). *Pattern Classification*, John Wiley & Sons, Inc., USA.
- Frank, E., Hall, M.A., Witten I.H., (2016). *The WEKA Workbench*. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques", Morgan Kaufmann, Fourth Edition.
- Han, J., Kamber, M., Pei, J., (2012) *Data Mining, Concepts and Techniques*, Morgan Kaufmann Publishers, USA.
- Joshi, A., Kale, S., Chandel, S., Pal, D.K., (2015). Likert Scale: Explored and Explained. *Current Journal of Applied Science and Technology*, 7, 4, pp. 396-403.
- Letelier, P., Penadés, M., (2006). Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP). *Técnica administrativa*, 5, 26.
- Marzocchi, V. A., Vilchez, A., D'Amato, M., (2012). Incorporación de TICs de modelado molecular en la enseñanza universitaria de la Química. *Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología*, 8, 9-15.
- MATLAB (2018), The MathWorks, Inc., Natick, Massachusetts, United States.
- Monserrat, C., López, O., Meier, U., Juan, M.C., Grau, V., Gil, J.A., Lozano, J.A & Alcañiz, M., (2002). GeRTiSS: Simulador quirúrgico virtual para el entrenamiento en cirugías mínimamente invasivas. In: IX Congreso Nacional de Informática Médica. Informed, pp. 87-91.
- Murty, M.N., Devi, V.S., (2015). *Introduction to Pattern Recognition and Machine Learning*. World Scientific Publishing, Singapore.
- Orjuela-Duarte, A., Rojas C.M., (2008). Las Metodologías de Desarrollo Ágil como una Oportunidad para la Ingeniería del Software Educativo. *Revista Avances en Sistemas e Informática*, 5, pp. 159-171.
- Pantaleo, G., Rinaudo, L., (2015). *Ingeniería de software*. Alfaomega, Argentina.
- Valencia Armijos, W. V., Charro Arévalo, C., (2007). *Modelo tridimensional de la historia geológica del volcán Cotopaxi*. Tesis Sistemas Informáticos y de Computación. Escuela Politécnica Nacional.
- Wells, D., (1999). *Extreme Programming-Rules*. Retrieved from Extreme Programming: <http://www.extremeprogramming.org/rules.html>
- Wu, X., Kumar, V., (2009). *The Top Ten Algorithms in Data Mining*. CRC Press, Boca Raton, FL.
- Yee, L.Y., Belaidan, S.L., Rahman, A.A, Harun, K.S., (2019). Implementing K-Means Clustering Algorithm in Collaborative Trip Advisory and Planning System. *Periodicals of Engineering and Natural Sciences* 7,2, pp. 723-740.