UNIVERSIDAD CATÓLICA DE SANTA MARÍA

FACULTAD DE CIENCIAS E INGENIERÍAS FÍSICAS Y FORMALES

SEGUNDA ESPECIALIDAD EN SISTEMAS Y TECNOLOGÍAS DE LA INFORMACIÓN

ESTUDIO COMPARATIVO DE TÉCNICAS NO SUPERVISADAS DE MINERÍA DE DATOS PARA SEGMENTACIÓN DE ALUMNOS

Tesis presentada por:

LAURA OCHOA, LETICIA MARISOL

Para optar el Título de Segunda Especialidad en Sistemas y Tecnologías de la Información

Asesor: Ing. Robert Arisaca Mamani

Arequipa – Perú

2016

Agradecimientos

A Dios, por ser guía en mi vida,

A los docentes de la Segunda Especialidad,

Por sus enseñanzas y conocimientos compartidos

A mi familia y amigos

Por su confianza, apoyo y aprecio.

ÍNDICE

LISTA DE FIGURAS

LISTA DE TABLAS

RESUMEN

En este trabajo se realiza un estudio comparativo de técnicas no supervisadas de minería de datos para la segmentación de alumnos utilizando algoritmos de K-means y PAM dentro del clustering particional y métodos de ward, single, complete, average, mcquitty, median y centroid del clustering jerárquico aglomerativo, luego se elige el algoritmo de minería de datos con la que se obtiene mejor calidad de agrupamiento utilizando medidas internas como las distancias intra-cluster e inter-cluster, y el coeficiente de silueta, obteniendo mejores resultados con la técnica de clustering particional K-means para la segmentación académica en tres grupos que puede ser utilizado para reforzar el aprendizaje de los alumnos en los niveles básico, intermedio y avanzado.

ABSTRACT

In this work, a comparative study of unsupervised data mining techniques for student segmentation are performed using K-means and PAM algorithms within the partitioning clustering and methods of ward, single, complete, average, mcquitty, median and centroid of agglomerative hierarchical clustering. Then, the data mining algorithm with which it obtains better quality of clustering is chosen using internal measures such as intra-cluster and inter-cluster distances, and the silhouette coefficient, obtaining better results with the partitioning clustering technique K-means for academic segmentation in three groups that can be used to reinforce students' learning at the basic, intermediate and advanced levels.

INTRODUCCIÓN

La tendencia de las instituciones de educación superior es garantizar e incrementar la calidad de la educación haciendo uso de la tecnología de la información para optimizar la toma de decisiones, aumentar la tasa de graduación, reducir el abandono y la deserción.

Para ello se requiere el uso de herramientas de minería de datos educacional (MDE) que permitan descubrir conocimiento oportuno sobre el rendimiento de los alumnos a través de la segmentación académica, que sirva de soporte en la toma de decisiones de las autoridades y coordinadores de las instituciones de educación superior, para establecer estrategias que permitan reforzar el aprendizaje de los estudiantes de forma personalizada.

La minería de datos educacional (MDE) permite responder preguntas sobre qué sabe realmente un estudiante y cómo está aprendiendo. De esta manera, la MDE permite descubrir información útil que ayuda a los docentes y responsables de las instituciones educativas en determinar la manera más pertinente para guiar a sus estudiantes, maximizando su aprendizaje. (Romero y Ventura, 2007).

En el primer capítulo se define el planteamiento de la investigación, en el segundo capítulo se describen los conceptos de minería de datos, técnicas descriptivas, clustering, evaluación de clustering, metodologías y herramientas para el desarrollo de proyectos de minería de datos; en el tercer capítulo se desarrolla la propuesta de segmentación de alumnos utilizando técnicas no supervisadas de minería de datos, en el cuarto capítulo se evalúan las técnicas de minería de datos utilizados para el estudio comparativo, se presentan las pruebas realizadas y resultados obtenidos. Finalmente se mencionan las conclusiones y recomendaciones.

CAPÍTULO 1: PLANTEAMIENTO DE LA INVESTIGACIÓN

1.1. Planteamiento del Problema

Entre los problemas más complejos que enfrentan las instituciones de educación podemos mencionar: mejorar la calidad académica, disminuir la deserción y la reprobación, evitar el atraso estudiantil y los bajos índices de eficiencia relacionado con las tasas de graduación. Esto requiere gestionar estrategias y tomar medidas frente a estos acontecimientos; para ello es posible recurrir al proceso denominado Minería de Datos Educacional (MDE) (Eckert y Suénaga, 2013).

Los alumnos son la esencia de las universidades, los cuales son diferentes entre sí, tienen distintas necesidades y su rendimiento académico varía por lo que se hace necesario agrupar a los alumnos con características similares para reforzar su aprendizaje de forma personalizada.

En la mayoría de los casos las instituciones de educación superior no hacen uso de herramientas de minería de datos para realizar la segmentación académica, no pudiendo así personalizar la atención a los alumnos para mejorar su aprendizaje, pudiendo hacerlos más rentables en el tiempo evitando la deserción de los mismos.

Existen varios algoritmos de clustering que generan diferentes resultados de agrupamiento, por lo que existe incertidumbre en cual utilizar para la segmentación académica.

1.2. Objetivos de la Investigación

1.2.1. General

Realizar un estudio comparativo de técnicas no supervisadas de minería de datos para segmentación de alumnos.

1.2.2. Específicos

- Analizar diferentes técnicas de minería de datos para la segmentación académica.
- Utilizar un estudio de caso para aplicar técnicas de segmentación.
- Seleccionar la técnica de segmentación más adecuada para el estudio de caso.

1.3. Preguntas de Investigación

¿Es posible realizar un estudio comparativo de técnicas no supervisadas de minería de datos para segmentación de alumnos que permita obtener agrupaciones de calidad?

¿Qué técnicas de minería de datos se pueden utilizar para la segmentación académica?

1.4. Línea y Sub-Línea de Investigación

Línea: Sistemas de Información y Bases de Datos

Sub-Línea: Minería de Datos

1.5. Tipo y Nivel de Investigación

Tipo de investigación: Aplicada.

Nivel de investigación: Descriptiva, comparativa y experimental.

1.6. Palabras Claves

Minería de datos, segmentación académica, clustering

1.7. Solución Propuesta

1.7.1. Justificación e Importancia

Existen varios algoritmos de clustering que generan diferentes resultados de agrupamiento, por esta razón se requiere investigar las principales técnicas de minería de datos que se puedan utilizar para la segmentación académica y realizar un estudio comparativo que permita elegir el algoritmo con la que se obtiene mejor calidad de agrupamiento.

Las instituciones de educación superior con la segmentación académica utilizando técnicas y algoritmos de clustering de minería de datos podrán tener un mejor conocimiento de las características y rendimiento de sus alumnos, que les permitirá crear grupos de estudiantes y reforzar su aprendizaje mediante una enseñanza personalizada, pudiendo así disminuir la deserción y hacerlos más rentables en el tiempo.

1.7.2. Descripción de la Solución

Se investigó las dos principales metodologías para el desarrollo de proyectos de minería de datos, las cuales son CRIPS-DM (Cross- Industry Standard Process for Data Mining) y SEMMA (Sample, Explore, Modify, Model, Assess). Se eligió utilizar CRISP-DM ya que mantiene una perspectiva más amplia respecto a los objetivos empresariales del proyecto y por ser la metodología que se utiliza en mayor porcentaje para proyectos de minería de datos según encuestas en los últimos años por (KDnuggets,

2014), el cual es un sitio líder en análisis de negocio, big data, minería de datos.

Luego se investigó las características de herramientas de minería de datos que se utilizan para el desarrollo de proyectos de minería de datos como son R, RapidMiner, WEKA, SQL Server Analysis Services, SAS Enterprise Miner. Se eligió utilizar R ya que es el software más utilizado para minería de datos según encuestas realizadas por (KDnuggets, 2015) y por ser un lenguaje de programación y entorno de software de código abierto que permite crear gráficos estadísticos y realizar agrupación en clústeres, así como la aplicación de otras técnicas.

Se investigó y realizó un estudio comparativo de las principales técnicas de clustering como son K-means y PAM dentro del clustering particional y los métodos de ward, single, complete, average, mcquitty, median y centroid del clustering jerárquico aglomerativo. Luego se eligió el algoritmo de clustering particional K-means para la segmentación académica, ya que se obtuvo mejor calidad de agrupamiento utilizando medidas internas como las distancias intra-cluster e inter-cluster, y el coeficiente de silueta.

1.8. Delimitación de la investigación

Análisis de datos de los registros académicos de los alumnos del II Semestre de la Escuela Profesional de Ingeniería de Sistemas de la Universidad Católica de Santa María correspondiente al semestre Par 2014.

CAPÍTULO 2: FUNDAMENTOS TEÓRICOS

2.1. Estado del Arte

En el trabajo de (Chamba, 2015) realiza la segmentación de clientes tomando en cuenta el comportamiento de compra en la empresa tecnológica Master PC utilizando algoritmos de agrupamiento como k-means, k-medoids, y Self-Organizing Maps (SOM), se aplicó la metodología CRISP-DM para el proceso de minería de datos, el análisis se realizó en base al modelo RFM (Recencia, Frecuencia, Valor Monetario) y para validar el resultado de los algoritmos de agrupamiento y seleccionar el que proporcione grupos de mejor calidad se empleó la técnica de evaluación en cascada aplicando un algoritmo de clasificación, resultando el algoritmo k-medoids el de mayor precisión. Finalmente se utilizó el algoritmo Apriori para encontrar asociaciones entre productos, para cada grupo de clientes.

En su trabajo (Morelo, 2014) realiza la segmentación de perfiles de cliente de la empresa Zona T mediante el desarrollo de un software que implementa una nueva técnica de minería de datos basada en el algoritmo para el análisis de clústers K-Medias, el cual, amplía el alcance de este algoritmo y de los métodos utilizados por los software CRM analíticos en la actualidad para esta tarea. Para esto se recopilaron datos sociodemográficos y de comportamiento de un total de 180 clientes de la empresa Zona T durante los años 2012 y 2013. Y se definieron una serie de atributos que permitieran caracterizar los consumidores con base en las necesidades de la organización. Luego, con base en estos atributos se agruparon los clientes y finalmente se relacionaron los productos ofrecidos por la empresa basados en sus características y la descripción de cada segmento.

En el trabajo de (León y Muñoz, 2013) se desarrolla un proceso de extracción de las diferentes fuentes (bases de datos, hojas de cálculo, archivos planos, Data Warehouse, etc.), transformación y carga de los datos de los egresados, continuando con el filtrado, la segmentación y agrupación de la información con características y patrones similares que permiten tener conocimiento para hacer análisis y obtener una buena toma de decisiones para beneficiar al Programa Profesional de Ingeniería de Sistemas de la UCSM de acuerdo a las necesidades tanto en el ámbito académico, laboral y tecnológico.

En el trabajo de (Villazana et al., 2012) presenta un estudio comparativo entre métodos de agrupamiento basados en máquinas de soporte vectorial (SVM) y C-medios difuso para generar grupos de las señales de electrocardiogramas normales y patológicas en un espacio complejidad de Lempel-Ziv y entropía de Shannon. Una señal de electrocardiograma normal y ocho electrocardiogramas con arritmias fueron seleccionadas de la base de datos MIT-BIH Arrhythmia Database, y fueron pre-procesadas para remover el ruido. Cada señal de electrocardiograma fue dividida en 35 segmentos de cuatro segundos. Los resultados demostraron que tanto la máquina de soporte vectorial como C-medios difusos fueron capaces de detectar y agrupar los patrones normales y patológicos de las señales de electrocardiograma. No obstante, los índices de validación de agrupamiento revelaron que el agrupamiento obtenido con la C-medios difusos fue mejor que el obtenido con la máquina de soporte vectorial.

Margarita Gallardo (Gallardo, 2009) en su trabajo realiza un estudio comparativo de los algoritmos de agrupamiento basado en densidades como DBSCAN, Mean Shift y LPC para así poder comprender mejor las

características cualitativas de las agrupaciones basadas en densidades y medir cuantitativamente la relación existente entre estos tres algoritmos. DBSCAN y Mean Shift permitieron encontrar grupos de formas arbitrarias, mientras que LPC no siempre encuentra grupos en forma arbitraria.

En su trabajo (Sulla, 2015) aplica métodos de clasificación como árboles de decisión C4.5 (J48) sobre datos de evaluación original de los estudiantes para predecir la deserción estudiantil y poder determinar la proyección de apertura de grupos o secciones y otras acciones. El resultado de su árbol de decisión predijo el número de estudiantes que son propensos a abandonar la carrera. El análisis comparativo de los resultados indica que la predicción ha ayudado a determinar con mayor precisión el mejoramiento en el resultado. Para analizar la exactitud del algoritmo, se compara con el algoritmo RandomTree encontrando que es tan eficiente en términos de precisión de los resultados académicos del estudiante y el tiempo tomado para crear el árbol.

2.2. Bases Teóricas de la Investigación

2.2.1. Minería de Datos

Knowledge Discovery in Databases (KDD), Descubrimiento de conocimiento en base de datos, es el proceso de identificar patrones válidos, nuevos, útiles y comprensibles de grandes volúmenes de datos. La minería de datos es el núcleo matemático del proceso KDD, que comprende los algoritmos que exploran los datos, desarrollan modelos matemáticos y descubren patrones significativos (implícita o explícita), los cuales son la esencia del conocimiento útil (Maimon y Rokach, 2010).

La minería de datos es el proceso de detectar la información procesable de los conjuntos grandes de datos. Utiliza el análisis matemático para deducir los patrones y tendencias que existen en los datos. Normalmente, estos patrones no se pueden detectar mediante la exploración tradicional de los datos porque las relaciones son demasiado complejas o porque hay demasiado datos. (Microsoft Corporation, 2016).

Minería de datos es el proceso de extraer conocimiento útil y comprensible, previamente desconocida, desde grandes cantidades de datos almacenados en distintos formatos (Witten y Frank, 2005).

Esta tecnología emergente combina el análisis estadístico, el aprendizaje automático, y gestión de base de datos para extraer información de sistemas de bases de datos de gran tamaño. La minería de datos requiere la integración de varias tecnologías (Thuraisingham, 2000).

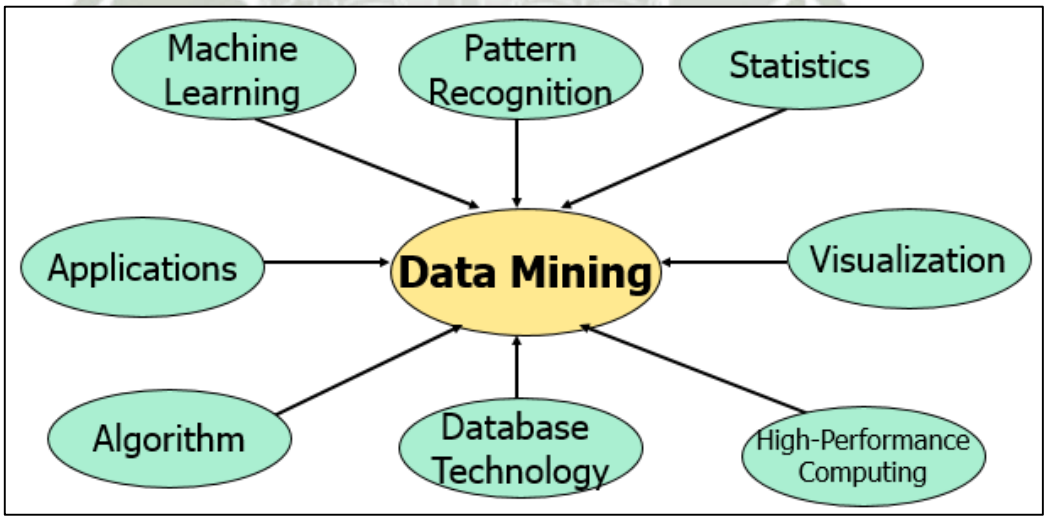


Figura 1: Minería de datos se integra con múltiples disciplinas (Han, Kamber y Pei, 2011)

La minería de datos se utiliza para explorar los datos y poder hallar patrones que sirvan para hacer interpretaciones, asociaciones y

clasificaciones. La minería de datos permite convertir datos en conocimiento, utiliza técnicas de aprendizaje automático, redes neuronales, visualización de datos y análisis estadístico.

Un proceso típico de minería de datos consta de los siguientes pasos generales (Hernández, Ramírez y Ferri, 2004):

- **Selección del conjunto de datos**, tanto en lo que se refiere a las variables objetivo (aquellas que se quiere predecir, calcular o inferir), como a las variables independientes (las que sirven para hacer el cálculo o proceso), como posiblemente al muestreo de los registros disponibles.
- **Análisis de las propiedades de los datos**, en especial los histogramas, diagramas de dispersión, presencia de valores atípicos y ausencia de datos (valores nulos).
- **Transformación del conjunto de datos de entrada**, se realizará de diversas formas en función del análisis previo, con el objetivo de prepararlo para aplicar la técnica de minería de datos que mejor se adapte a los datos y al problema, a este paso también se le conoce como preprocesamiento de los datos.
- **Seleccionar y aplicar la técnica de minería de datos**, se construye el modelo predictivo, de clasificación o segmentación.
- **Extracción de conocimiento**, mediante una técnica de minería de datos, se obtiene un modelo de conocimiento, que representa patrones de comportamiento observados en los valores de las variables del problema o relaciones de asociación entre dichas variables. También pueden usarse varias técnicas a la vez para generar distintos modelos,

aunque generalmente cada técnica obliga a un preprocesado diferente de los datos.

- **Interpretación y evaluación de datos**, una vez obtenido el modelo, se debe proceder a su validación comprobando que las conclusiones que arroja son válidas y suficientemente satisfactorias. En el caso de haber obtenido varios modelos mediante el uso de distintas técnicas, se deben comparar los modelos en busca de aquel que se ajuste mejor al problema. Si ninguno de los modelos alcanza los resultados esperados, debe alterarse alguno de los pasos anteriores para generar nuevos modelos.

2.2.2. Tareas de Minería de Datos

Las tareas de minería de datos se dividen en dos categorías, como se muestra en la Figura 2.

2.2.2.1. Predictiva

El objetivo de este tipo de minería, es predecir el valor particular de un atributo basado en otros atributos. El atributo a predecir es comúnmente llamado "clase" o variable dependiente, mientras que los atributos usados para hacer la predicción se llaman variables independientes (Tan, Steinbach y Kumar, 2006).

Permite predecir valores de variables desconocidas (variable dependiente o variable objetivo) a partir de otros atributos de la base de datos (variables independientes).

- **Clasificación:** El objetivo de esta tarea es la clasificación de un dato dentro de las clases definidas del dominio que se está modelando (Riquelme, Ruiz y Gilbert, 2006).

Permite la clasificación de los registros que tienen clase desconocida en categorías o clases ya definidas en la base de datos.

- **Regresión:** Predice un valor de una variable de valor continuo dado en base a los valores de las otras variables, suponiendo un modelo lineal o no lineal de dependencia (Tan, Steinbach y Kumar, 2006).

  El objetivo es predecir los valores de una variable continua a partir de la evolución sobre otra variable continua, generalmente el tiempo. Ejemplo, se intenta predecir el número de clientes o pacientes, los ingresos, llamadas, ganancias, costos, etc. a partir de los resultados de semanas, meses o años anteriores. (Hernández, 2006).

2.2.2.2. Descriptiva

El objetivo de este tipo de minería, es encontrar patrones (correlaciones, tendencias, grupos, trayectorias y anomalías) que resuman relaciones en los datos (Chen, Han y Yu, 1996).

Se encarga de identificar patrones para la descripción de los datos existentes.

- **Agrupamiento:** Permite obtener grupos o conjuntos en donde se incorpore elementos similares extraídos de las clases del dominio dado (Riquelme, Ruiz y Gilbert, 2006).

  Permite la segmentación en grupos excluyentes entre sí y cercanos dentro del grupo.

- **Reglas de Asociación:** Una asociación entre dos atributos ocurre cuando la frecuencia de que se den dos valores determinados de cada uno conjuntamente es relativamente alta. Ejemplo, en un supermercado se analiza si los pañales y la leche del bebe se compran conjuntamente (Hernández, 2006).

  Encuentra relaciones entre dos o más atributos que ocurren con mayor frecuencia.

- **Secuenciación:** Es un conjunto de objetos dado, con cada objeto asociado con su propia línea de tiempo de eventos, encuentra reglas que predicen fuertes dependencias secuenciales entre los diferentes eventos. Las reglas se forman descubriendo primero patrones. Las ocurrencias de eventos en los patrones se rigen por restricciones de temporización. (Tan, Steinbach y Kumar, 2006).

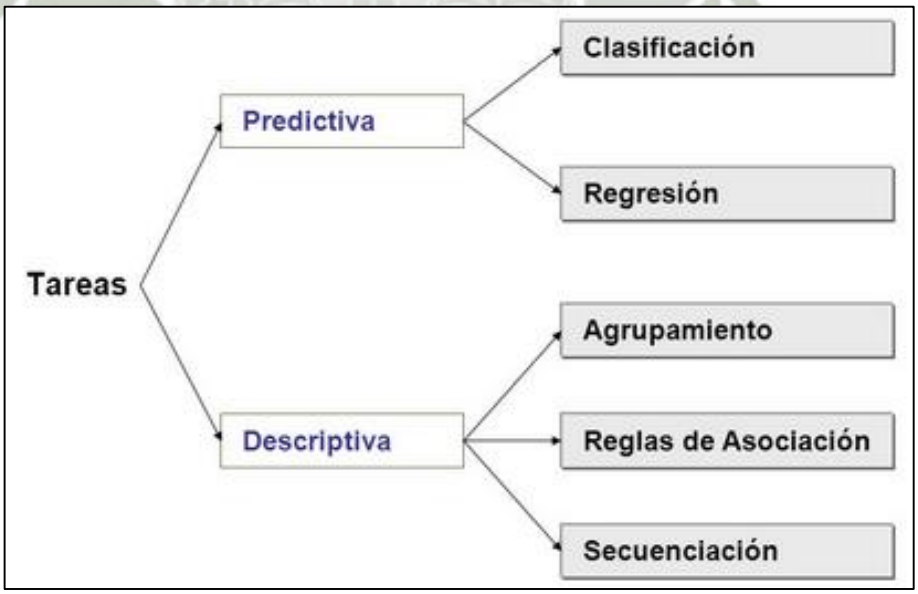


Figura 2: Tareas de la Minería de Datos (Oporto, 2014)

2.2.3. Técnicas de Minería de Datos

Las técnicas de minería de datos permiten llevar a cabo las tareas predictivas y descriptivas haciendo uso de algoritmos de minería de datos. Según el objetivo del análisis de los datos, los algoritmos utilizados se clasifican en supervisados y no supervisados (Weiss y Indurkhya, 1998):

- **Aprendizaje supervisado (o predictivo):** Predicen el valor de un atributo *(etiqueta)* de un conjunto de datos, desconocido a priori, a partir de otros atributos conocidos (atributos descriptivos). A partir de datos cuya etiqueta se conoce se induce una relación entre dicha etiqueta y otra serie de atributos. Esas relaciones sirven para realizar la predicción en datos cuya etiqueta es desconocida. Esta forma de trabajar se conoce como aprendizaje supervisado y se desarrolla en dos fases: Entrenamiento (construcción de un modelo usando un subconjunto de datos con etiqueta conocida) y prueba (prueba del modelo sobre el resto de los datos).

- **Aprendizaje no supervisado (o del descubrimiento del conocimiento):** Se descubren patrones y tendencias en los datos. El descubrimiento de esa información sirve para llevar a cabo acciones y obtener un beneficio (científico o de negocios) de ellas.

En la Tabla 1 se muestra algunas de las técnicas de minería de datos.

Tabla 1: Clasificación de las técnicas de minería de datos (Moreno et al., 2001)

Supervisados	No supervisados
Árboles de decisión	Detección de desviaciones
Inducción neuronal	Segmentación
Regresión	Agrupamiento (“clustering”)
Series temporales	Reglas de asociación
	Patrones secuenciales

2.2.4. Clustering

El análisis de clusters agrupa objetos basados solamente en la información encontrada en los datos que describen a los objetos y sus relaciones. El objetivo es que los objetos dentro de un grupo sean similares (o relacionados) entre sí y diferentes de (o no relacionados con) los objetos en otros grupos. A mayor similitud (u homogeneidad) dentro de un grupo y a mayor diferencia entre grupos, mejor o más distinto es el clustering. (Tan, Steinbach y Kumar, 2006), como se observa en la Figura 3.

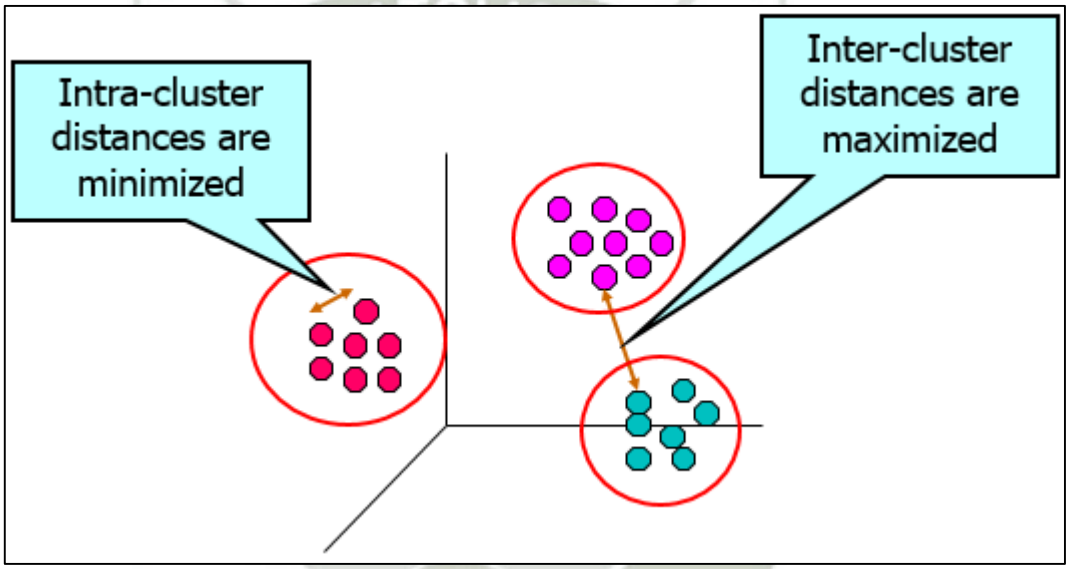


Figura 3: Análisis de Cluster (Tan, Steinbach y Kumar, 2006)

Clustering es una de las técnicas más útiles para descubrir conocimiento oculto en un conjunto de datos. En la actualidad el análisis de clustering en minería de datos ha jugado un rol muy importante en una amplia variedad de áreas tales como: reconocimiento de patrones, análisis de datos espaciales, procesamiento de imágenes, cómputo y multimedia, análisis médico, economía, bioinformática y biometría principalmente (Han y Kamber, 2006).

Clustering permite la formación de grupos cuyos registros u observaciones dentro del grupo sean lo más homogéneos o similares, y entre los grupos sean bastante distantes o diferentes.

La actividad de clustering implica los siguientes pasos (Jain et al., 1999) (Hernández, 2006):

a) Representación de patrones: Se refiere al número de clases, número de patrones disponibles, y el número, tipo y tamaño de las características disponibles para el algoritmo de clustering.
b) Definición de proximidad: La proximidad de los patrones es usualmente medida por una función distancia definida.
c) Clustering o agrupamiento: Puede ser realizado en un gran número de formas. Se pueden utilizar algoritmos de clustering jerárquicos, particionales y otras técnicas que abarcan métodos probabilísticos o de teoría de grafos.
d) Abstracción de datos: Es el proceso de extraer una representación simple y compacta del conjunto de datos.
e) Verificación de resultados: Consiste en validar el análisis de clustering realizado evaluando los resultados obtenidos.

En la Figura 4 se puede observar la secuencia típica de los tres primeros pasos, incluyendo una retroalimentación.

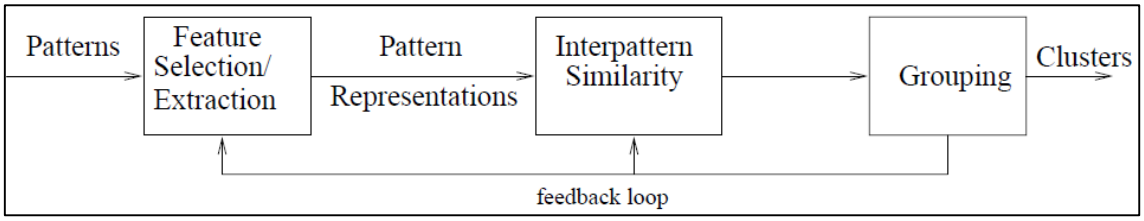


Figura 4: Etapas de Clustering (Jain et al., 1999)

Los requisitos y desafíos de los algoritmos de clustering son las siguientes (Han, Kamber y Pei, 2011):

- Escalabilidad: Para realizar agrupamiento o clustering en bases de datos con millones de registros u observaciones.
- Habilidad para trabajar con distintos tipos de atributos: Numérico, binario, categórico, ordinal, y una mezcla de estos.
- Descubrimiento de clusters con formas arbitrarias: Es importante diseñar algoritmos que puedan establecer clusters de formas arbitrarias, y no solamente con forma circular y densidad similares.
- Requerimientos mínimos en el conocimiento del dominio para determinar los parámetros de entrada: Existen parámetros que no son fáciles de determinar, y esto haría que sea difícil controlar la calidad del algoritmo.
- Habilidad para tratar con datos ruidosos: Algunos algoritmos de clustering son sensibles a tales datos y pueden derivarlos a clusters de baja calidad.
- Insensibilidad al orden de las observaciones de entrada: El algoritmo debe ser insensible al orden de las observaciones, y el resultado de clusters debe ser siempre el mismo.
- Alta dimensionalidad: Que pueda trabajar también en repositorios con un alto espacio dimensional, o gran cantidad de atributos.
- Clustering basado en restricciones: Agrupar los datos no sólo por el comportamiento, sino también que satisfagan ciertas restricciones.
- Interpretación y uso: Los resultados del clustering se espera que sean comprensibles, fáciles de interpretar y de utilizar.

2.2.5. Técnicas de Clustering

Los algoritmos de clustering varían entre sí por las reglas heurísticas que utilizan y el tipo de aplicación para el cual fueron diseñados. La mayoría de ellos se basa en el empleo sistemático de distancias entre vectores (objetos a agrupar) así como entre grupos que se van formando durante el clustering. Las características básicas por las que los algoritmos de clustering pueden ser clasificados son en función de (Hernández, 2006):

a) El tipo de dato que manejan (numérico, categórico y/o mixto).

b) El criterio utilizado para medir la similitud entre los puntos.

c) Los conceptos y técnicas de clustering empleadas (ej. lógica difusa, estadísticas).

En la literatura existen una gran cantidad de técnicas de clustering que varían de acuerdo a la arquitectura que utilizan (Jain et al., 1999). Una clasificación general divide los algoritmos en: clustering jerárquico, clustering particional y clustering basado en densidad, como se observa en la Figura 5.

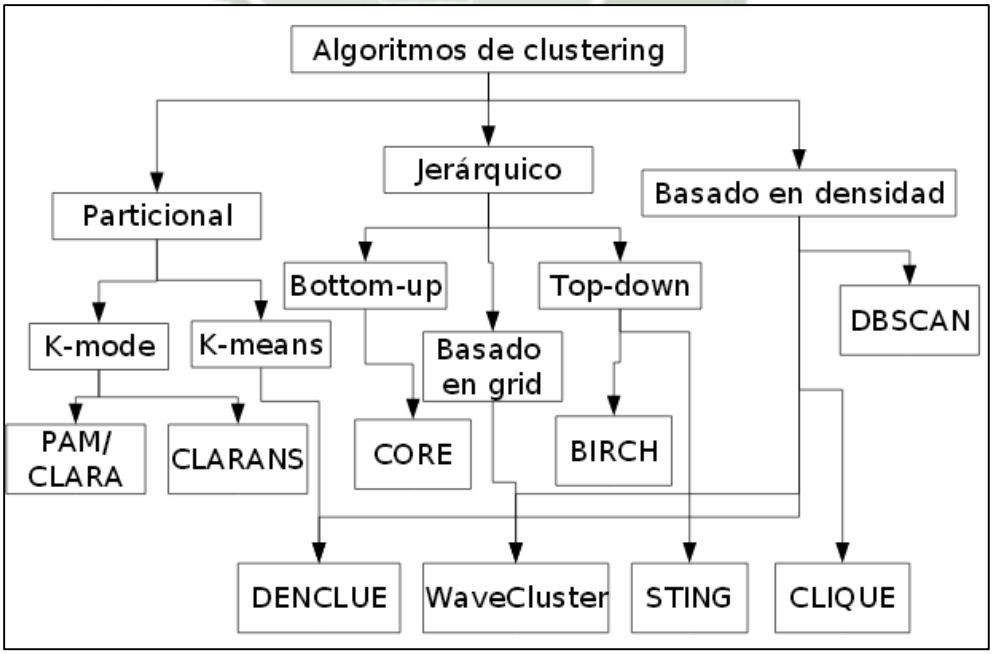


Figura 5: Algoritmos básicos de clustering (Hernández, 2006)

2.2.5.1. Clustering jerárquico

Un conjunto de clústeres anidados organizados como un árbol jerárquico (Tan, Steinbach y Kumar, 2006).

Un método jerárquico crea una descomposición jerárquica de un conjunto de datos, formando un dendrograma (árbol) que divide recursivamente el conjunto de datos en conjuntos cada vez más pequeños (Jain et al., 1999).

Un método jerárquico puede ser clasificado como aglomerativo o divisivo, basado en cómo se forma la descomposición jerárquica (Han, Kamber y Pei, 2011).

- Aglomerativo, llamado también *bottom-up*, comienza con cada objeto formando un grupo separado. Sucesivamente los objetos o grupos cercanos uno al otro se une, hasta que todos los grupos se combinan en uno (el nivel más alto de la jerarquía) o hasta que se cumpla alguna condición de terminación.

- Divisivo, llamado también *top-down*, comienza con todos los objetos del mismo cluster. En cada iteración sucesiva, un cluster se divide en grupos más pequeños, hasta que cada objeto este en un cluster, o se cumpla la condición de terminación.

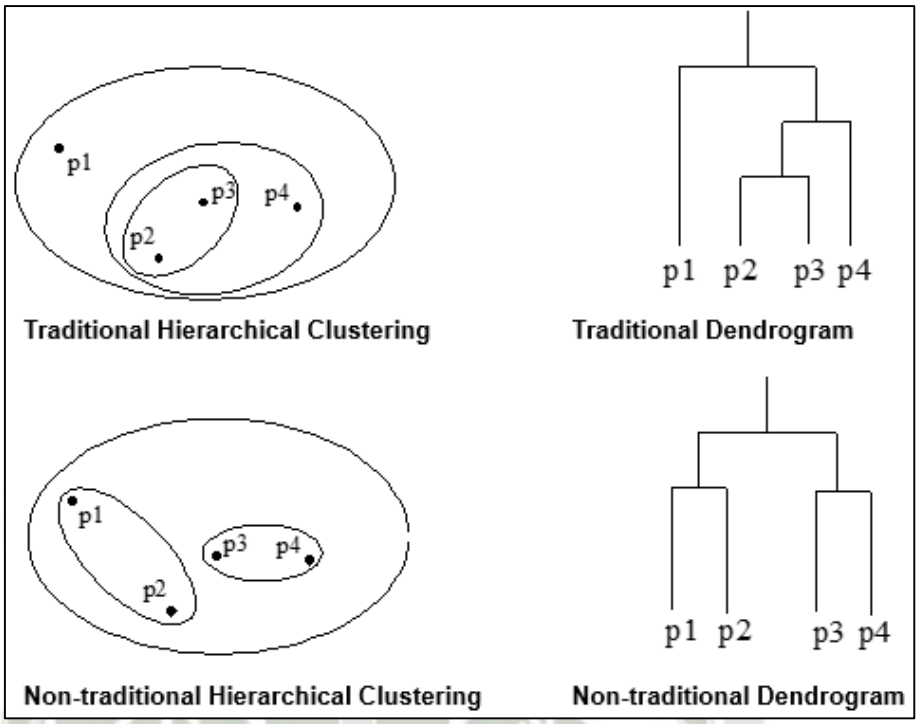


Figura 6: Clustering jerárquico (Tan, Steinbach y Kumar, 2006)

El clustering jerárquico no es recomendable para bases de datos grandes con millones de registros ya que la cantidad de distancias a calcular sería mayor, y la construcción del dendrograma sería compleja.

2.2.5.2. Clustering particional

Clustering particional es una división de objetos de datos en subconjuntos que no se superponen (clusters) de tal manera que cada objeto de datos está en exactamente un subconjunto (Tan, Steinbach y Kumar, 2006).

En la Figura 7 se muestra dos ejemplos de clustering particional.

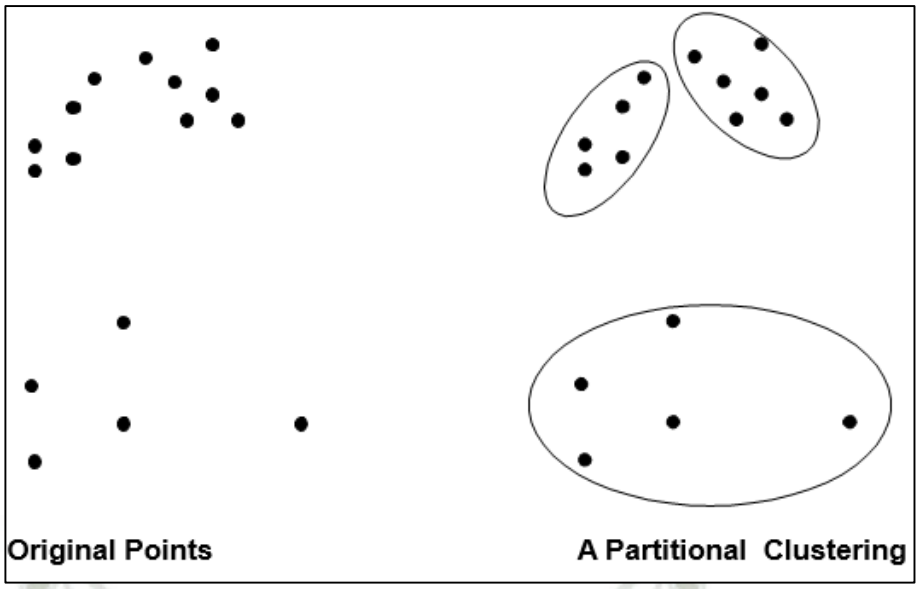


Figura 7: Clustering particional (Tan, Steinbach y Kumar, 2006)

Dado un conjunto de n objetos, un método de partición construye k particiones de los datos, donde cada partición representa un clúster y k≤ N; es decir, divide los datos en k grupos de manera que cada grupo debe contener al menos un objeto. En otras palabras, los métodos de partición realizan un nivel de partición en conjuntos de datos. Los métodos de partición básicos adoptan típicamente la separación de clúster exclusiva; es decir, cada objeto debe pertenecer exactamente a un grupo (Han, Kamber y Pei, 2011).

El clustering particional se puede utilizar para grandes cantidades de datos, encuentra clusters mutuamente exclusivos de forma circular y está basado en la distancia.

2.2.5.3. Clustering basado en densidad

Un grupo es una región densa de puntos, que está separada por regiones de baja densidad, de otras regiones de alta densidad. Se

utiliza cuando los grupos son irregulares o entrelazados, y cuando hay ruido presente (Tan, Steinbach y Kumar, 2006).

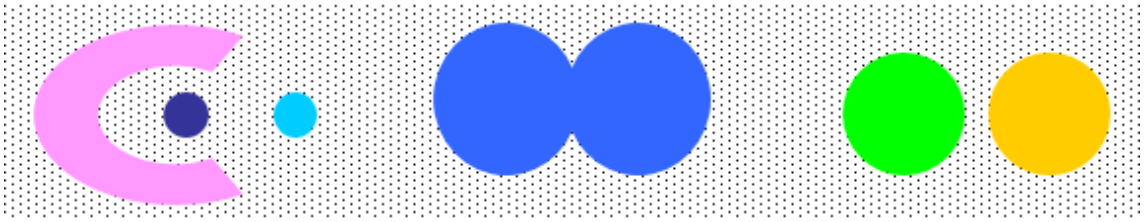


Figura 8: Clusters basados en densidad (Tan, Steinbach y Kumar, 2006)

La mayoría de los métodos de clustering agrupan objetos basados en la distancia entre objetos. Tales métodos pueden encontrar solamente clusters de forma esférica y encuentran dificultades para descubrir clusters de formas arbitrarias. Otros métodos de agrupamiento han sido desarrollados basados en la noción de densidad. Su idea general es continuar creciendo un cluster dado, siempre y cuando la densidad (número de objetos o puntos de datos) en el "vecindario" supere algún umbral. Por ejemplo, para cada punto de datos dentro de un grupo dado, la vecindad de un radio dado debe contener al menos un número mínimo de puntos. Este método puede usarse para filtrar el ruido u outliers y descubrir clusters de forma arbitraria (Han, Kamber y Pei, 2011).

2.2.6. Algoritmos de Clustering

2.2.6.1. Clustering jerárquico aglomerativo

El algoritmo de clustering aglomerativo es la técnica de clustering jerárquico más popular. Los algoritmos jerárquicos tradicionales utilizan una matriz de similitud o distancia (Tan, Steinbach y Kumar, 2006).

Este enfoque de clustering se refiere a una colección de técnicas de agrupamiento estrechamente relacionadas que producen un agrupamiento jerárquico comenzando con cada punto como un cluster singleton (con un solo elemento) e iterativamente lo agrupa con los dos clusters más cercanos hasta que un único cluster que abarca a los todos los demás permanece (Flores, 2014).

Hay algunos estudios que sugieren que estos algoritmos pueden producir mejor calidad de clusters. Sin embargo, los algoritmos de clustering jerárquicos no son recomendables para grandes cantidades de datos ya que son costosos en términos de sus requerimientos computacionales y de almacenamiento. El hecho de que todas las clusters terminen finalmente unidos también puede causar problemas para datos ruidosos o de alta dimensionalidad (Flores, 2014).

El algoritmo básico es sencillo, como se muestra en la Tabla 2.

Tabla 2: Algoritmo de Clustering Aglomerativo (Tan, Steinbach y Kumar, 2006)

Algoritmo de Clustering Aglomerativo
1. Calcular la matriz de proximidad 2. Dejar que cada punto de datos sea un cluster 3. **Repeat** 4. Combinar los dos clusters más cercanos 5. Actualizar la matriz de proximidad 6. **Until** solamente queda un solo cluster

Existen diferentes métodos de clustering jerárquico aglomerativo según el cálculo de distancias entre clusters que se utiliza, como por ejemplo:

- **Método single**, llamado también método de salto mínimo o distancia mínima, en este método la aglomeración de clusters está basada en la distancias de los dos casos más cercanos, como se observa en la Figura 9.

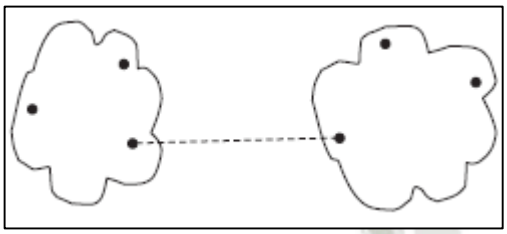


Figura 9: Método single (Tan, Steinbach y Kumar, 2006)

- **Método complete**, llamado también método de salto máximo o distancia máxima, en este método la distancia entre clusters es determinada por la distancia más alejada entre dos casos, como se observa en la Figura 10.

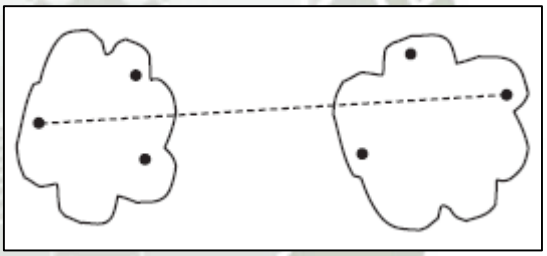


Figura 10: Método complete (Tan, Steinbach y Kumar, 2006)

- **Método average**, llamado también método de la media o promedio, en este método la distancia entre los grupos se determina por la distancia media entre todos los pares de casos en dos grupos diferentes. Su ventaja es tomar más de dos casos en consideración. Un tipo de método de enlace promedio minimiza la distancia dentro del grupo, que es la base de la distancia entre dos grupos.

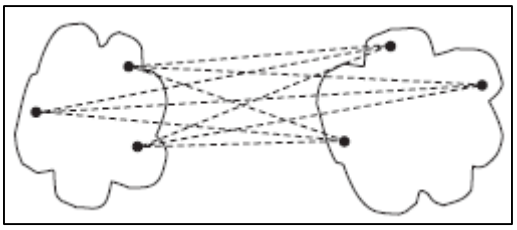


Figura 11: Método average (Tan, Steinbach y Kumar, 2006)

- **Método Ward**, este método en cada paso del análisis, considera la posibilidad de la unión de cada par de grupos y opta por la fusión de aquellos dos grupos que menos incrementen la suma de los cuadrados de las desviaciones al unirse. Tiende a crear grupos homogéneos de tamaño pequeño y similar.

- **Método median**, este método calcula la distancia entre los medoides de dos grupos.

- **Método centroid**, este método calcula la distancia entre los centroides de dos grupos.

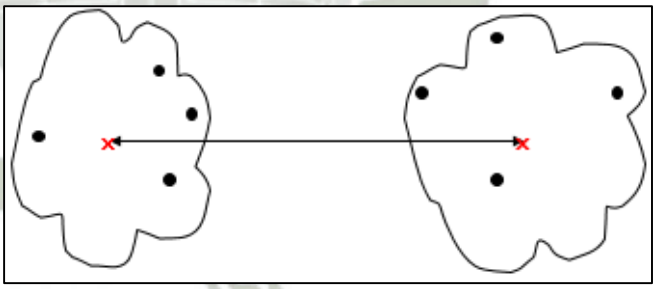


Figura 12: Método centroid (Tan, Steinbach y Kumar, 2006)

Según (Tan, Steinbach y Kumar, 2006), el clustering jerárquico tiene los siguientes problemas y limitaciones:

- Una vez que se toma la decisión de combinar dos grupos, no se puede deshacer.
- No se minimiza directamente ninguna función objetiva.

- Los diferentes esquemas tienen problemas con uno o más de los siguientes:
  - Sensibilidad al ruido y valores atípicos.
  - Dificultad para manejar clusters de diferentes tamaños y formas convexas.

2.2.6.2. K-means

Esta técnica está basada en el clustering particional que intenta encontrar un número de clusters (K) especificados por el usuario, los cuales son representados por sus centroides. El algoritmo básico se describe a continuación: Primero se eligen K centroides iniciales, donde K es un parámetro especificado por el usuario y corresponde al número de clusters deseados. Cada punto es asignado a su centroide más cercano y cada colección de puntos asignado a un centroide representa un cluster. El centroide de cada cluster se actualiza basado en la asignación de puntos al cluster. Se repiten los pasos de asignación y actualización hasta que los puntos dentro del cluster no cambien, o equivalentemente, hasta que los centroides dejen de cambiar (Flores, 2014).

El algoritmo básico de K-means es muy simple, como se muestra en la Tabla 3.

Tabla 3: Algoritmo básico K-means (Tan, Steinbach y Kumar, 2006)

Algoritmo básico K-means
1. Seleccionar K puntos iniciales como centroides 2. **Repeat** 3. Formar K cluster asignando cada punto a su centroide más cercano 4. Recalcular el centroide de cada cluster 5. **Until** los centroides no cambien

En la Figura 13 se muestra como a partir de tres centroides definidos inicialmente, los clusters finales se encuentran en cuatro iteraciones de asignación-actualización.

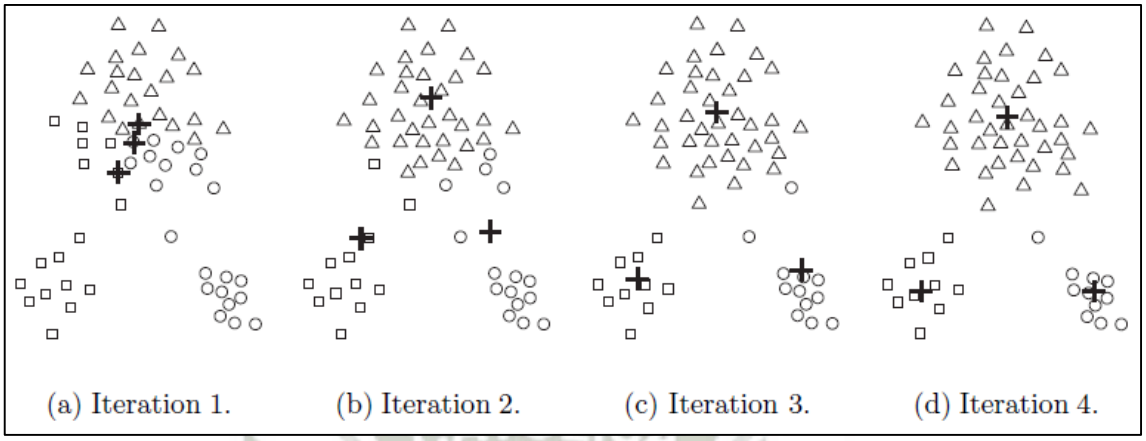


Figura 13: Algoritmo K-means para encontrar tres clusters en datos de prueba (Tan, Steinbach y Kumar, 2006)

Las limitaciones de K-means según (Tan, Steinbach y Kumar, 2006) son:

- K-means tiene problemas cuando los clusters son de diferente tamaño, densidades, que no tengan forma esférica.
- K-means tiene problemas cuando los datos contienen outliers.

La ventaja de K-means es ser un algoritmo simple, efectivo para pequeñas y medianas cantidades de datos. Utiliza el promedio para representar los centros de los clusters.

2.2.6.3. PAM

PAM (Partitioning Around Medoids) es un algoritmo típico de agrupación K-medoides. Aborda el problema de una manera iterativa. Al igual que el algoritmo k-means, los objetos representativos iniciales (llamados semillas) son elegidos arbitrariamente. Consideramos si la sustitución de un objeto representativo por un objeto representativo mejoraría la calidad de la agrupación. Todos los

posibles reemplazos son probados. El proceso iterativo de reemplazar objetos representativos por otros objetos continúa hasta que la calidad de la agrupación resultante no puede ser mejorada por ningún reemplazo. Esta calidad se mide mediante una función de coste de la disimilitud media entre un objeto y el objeto representativo de su agrupación (Han, Kamber y Pei, 2011).

K-medoides usa medianas en lugar de promedios para representar a los clusters.

Tabla 4: Algoritmo PAM. K-medoides (Han, Kamber y Pei, 2011)

Algoritmo PAM
1. Seleccionar K puntos iniciales como medoides o semillas 2. **Repeat** 3. Asigne cada objeto restante al cluster con el objeto representativo más cercano 4. Seleccionar al azar un objeto no representativo, o_{random} 5. Calcular el coste total, S, de intercambiar objeto representativo, o_j, con o_{random} 6. **if** S<0 **then** swap o_j con o_{random} para formar el nuevo conjunto de k objetos representativos 7. **Until** no cambien

PAM utiliza los medoides (mediana) para representar los centros de los clusters. PAM funciona de manera eficiente para pequeños conjuntos de datos, pero no escala bien para grandes conjuntos de datos.

2.2.6.4. DBSCAN:

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) encuentra objetos núcleo, es decir, objetos que tienen vecinos denso-alcanzables. Conecta objetos núcleos y sus vecinos para formar regiones densas como clusters (Han, Kamber y Pei, 2011).

DBSCAN es un algoritmo de clustering basado en densidad que produce un clustering particional, en el cual el número de cluster es determinado automáticamente por el algoritmo. Puntos con baja densidad son clasificados como ruido y son omitidos, por lo que DBSCAN no produce un clustering completo (Tan, Steinbach y Kumar, 2006).

DBSCAN opera en un enfoque de densidad en base a centros, la densidad es estimada para un punto en particular contando el número de puntos (MinPts) que se encuentra al interior de un radio especificado (Eps). Permite clasificar los puntos como aquellos que se encuentran al interior de una región densa (puntos de núcleo), aquellos en el borde de una región densa (puntos de borde) y aquellos que se encuentran dispersos (puntos de ruido o de fondo) (Tan, Steinbach y Kumar, 2006) (Flores, 2014):

- **Puntos de núcleo:** Puntos que tienen más de MinPts vecinos dentro de su vecindario de Radio Eps.
- **Puntos de borde:** Son los puntos que tienen menos de MinPts vecinos dentro de su vecindario de radio Eps, pero están en la vecindad de un punto de núcleo.
- **Puntos de ruido:** Son aquellos puntos que no caen en ninguna de las dos categorías anteriores.

En la Figura 14 se muestra un ejemplo de puntos de núcleo, de borde y de ruido considerando un Eps de valor 1 y un MinPts de valor 4.

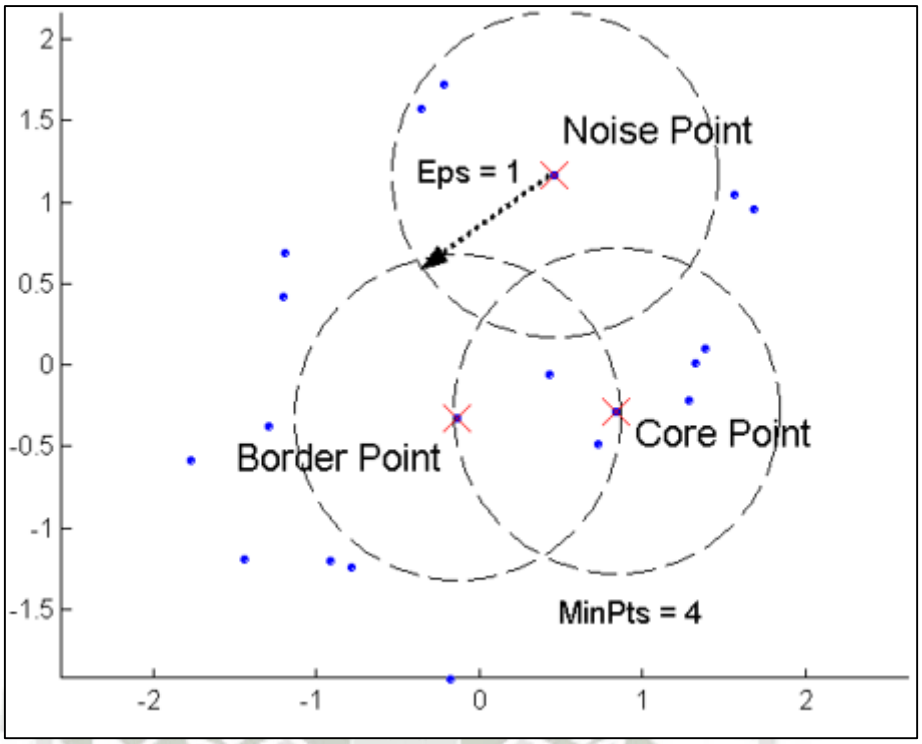


Figura 14: Puntos de núcleo, de borde y de ruido (Tan, Steinbach y Kumar, 2006)

El algoritmo comienza seleccionando un punto p arbitrario, si p es un punto central, se comienza a construir un grupo y se ubican en su grupo todos los objetos denso-alcanzables desde p. Si p no es un punto central se visita otro objeto del conjunto de datos. El proceso continúa hasta que todos los objetos han sido procesados. Los puntos que quedan fuera de los grupos formados se llaman puntos ruido, los puntos que no son ni ruido ni centrales se llaman puntos borde (Pascual, Pla y Sánchez, 2007).

El algoritmo DBSCAN se describe como sigue: Cualquier par de puntos de núcleo que estén lo suficientemente cerca (menor a una distancia Eps) son asignados a un mismo cluster. Así mismo, cualquier punto de borde que este lo suficientemente cerca de un punto de núcleo es puesto en el mismo cluster que el punto de núcleo. (Podría ser necesario resolver "empates" cuando un punto de borde se

encuentra a la misma distancia de dos puntos de núcleo de distintos clusters). Los puntos de ruido se descartan (Tan, Steinbach y Kumar, 2006).

Tabla 5: Algoritmo DBSCAN (Tan, Steinbach y Kumar, 2006)

Algoritmo DBSCAN
1. Etiquetar todos los puntos como núcleo, borde o ruido 2. Eliminar puntos de ruido 3. Poner un borde entre todos los puntos de núcleo que están dentro de Eps de cada uno de otros 4. Convierta cada grupo de puntos centrales conectados en un clúster separado 5. Asignar cada punto de borde a uno de los clusters de sus puntos de núcleo asociados

DBSCAN puede encontrar clusters de forma arbitraria; pero a veces no produce un clustering completo ya que los puntos con baja densidad son considerados como ruidos y omitidos.

2.2.7. Evaluación de Clustering

Un buen método de agrupamiento producirá clusters de alta calidad (Han, Kamber y Pei, 2011):

- Alta similitud intra-cluster: Cohesiva dentro de los clusters.
- Baja similitud inter-cluster: Distintivo entre clusters.

Algunas de las razones de evaluar los clusters según (Tan, Steinbach y Kumar, 2006) son:

- Establecer el número correcto de clusters.
- Para comparar los algoritmos de clustering.
- Comparar dos conjuntos de clusters para determinar cuál es el mejor.

2.2.7.1. Sum of Squares Errors (SSE)

SSE es un índice interno utilizado para medir que tan buena es una estructura de clustering sin tener en cuenta información externa. Se utiliza para comparar dos clustering o dos clusters. Puede ser utilizado también para estimar el número de clusters (Tan, Steinbach y Kumar, 2006).

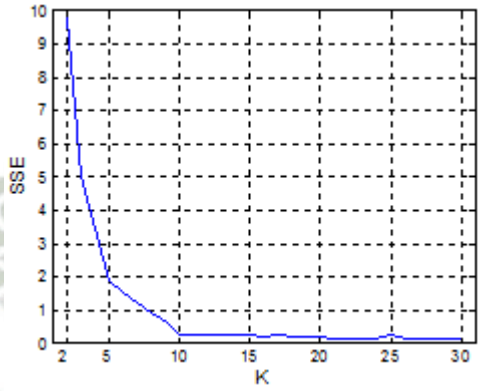


Figura 15: SSE para determinar el número de clusters (Tan, Steinbach y Kumar, 2006)

SSE es la suma de las diferencias cuadradas entre cada observación y la media de su grupo. Se puede utilizar como una medida de la variación dentro de un cluster. Si todos los casos dentro de un grupo son idénticos, entonces el SSE sería igual a 0. La fórmula para SSE es:

$$SSE = \sum_{i=1}^{n} (x_i - \bar{x})^2$$

2.2.7.2. Cohesión y separación de clusters

Las medidas no supervisadas de validación de clusters se dividen a menudo en dos clases: las medidas de cohesión del cluster, que determinan la similitud entre los objetos de un cluster y las medidas

de separación del cluster, que determina la diferencia o separación entre los clusters (Tan, Steinbach y Kumar, 2006).

La cohesión (WSS) se mide por la suma de los cuadrados intra-cluster:

$$WSS = \sum_{i} \sum_{x \in C_i} (x - m_i)^2$$

Donde:

- i es el identificador del cluster
- mi corresponde al promedio del cluster
- x es el punto de datos que pertenece al grupo Ci

La separación (BSS) se mide por la suma de los cuadrados inter-cluster:

$$BSS = \sum_{i} |C_i| (m - m_i)^2$$

Donde:

- |Ci| es el tamaño del cluster i
- mi corresponde al promedio del cluster
- m corresponde al promedio total

Según (Tan, Steinbach y Kumar, 2006) la cohesión del cluster puede ser definida como la suma de los pesos de los enlaces en el grafo de proximidad que conectan puntos dentro del cluster y la separación entre dos clusters puede ser medida como la suma de los pesos de los enlaces de puntos en un grupo a los puntos en el otro grupo. Esto está ilustrado en la Figura 16.

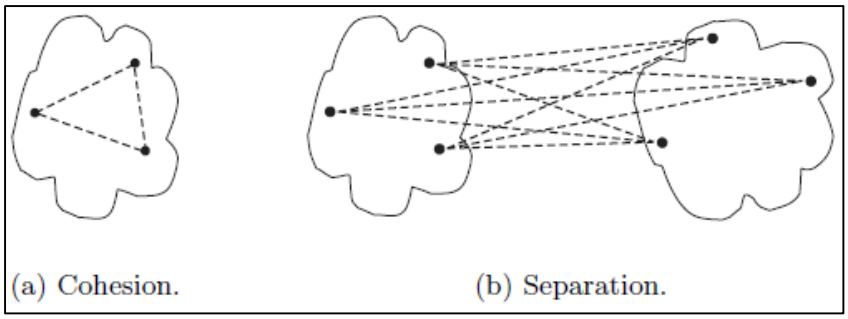


Figura 16: Cohesión y separación de clusters (Tan, Steinbach y Kumar, 2006)

2.2.7.3. Coeficiente de silueta

Según (Bioinformática, 2016) el coeficiente silueta (silhouette) mide cuan buena es la asignación de un elemento o dato a su cluster. Para esto compara las distancias de este elemento respecto a todos los demás elementos del cluster al que pertenece, contra las distancias respecto a los clusters vecinos. El coeficiente silueta del elemento i se denota s(i).

- Si $s(i) \approx -1$, el dato i está mal agrupado
- Si $s(i) \approx 0$, el dato i está entre dos clusters
- Si $s(i) \approx 1$, el dato i está bien agrupado

El promedio de los s de los elementos dentro un cluster, da una idea de la calidad de ese cluster. El promedio de los s de todos los elementos dan una idea de que tan bien están agrupados todos los datos.

Según (Ayala, 2014) para la observación i y el grupo C, la silueta se construye:

$$\bar{d}(i,C) = \frac{1}{|C|}\sum_{j \in C} d(i,j)$$

La disimilaridad media i con los elementos del grupo C. Para cada observación i, sea A el cluster al cual lo ha asignado el procedimiento cluster que empleamos y calculamos a(i) la disimilaridad media de i con todos los demás individuos del grupo A, a(i) = d(i;A). Obviamente estamos asumiendo que A contiene al menos otro objeto. Consideremos d(i;C) para todos los grupos C ≠ A y seleccionemos el que tiene el mínimo valor.

2.2.8. Metodologías para el desarrollo de proyectos de minería de datos

2.2.8.1. CRISP-DM

CRISP-DM (Cross-Industry Standard Process for Data Mining), es una de las metodologías más utilizadas para proyectos de minería de datos, ciencia de los datos según encuestas en los últimos años por (KDnuggets, 2014).

La metodología CRISP-DM estructura el ciclo de vida de un proyecto de minería de datos en seis fases (Chapman et al., 2000), como se muestra en la Figura 17.

a) **Comprensión del negocio:** Esta fase inicial se enfoca en la comprensión de los objetivos de proyecto y exigencias desde una perspectiva de negocio, luego convirtiendo este conocimiento de los datos en la definición de un problema de minería de datos y en un plan preliminar diseñado para alcanzar los objetivos.

b) **Comprensión de los datos:** La fase de entendimiento de datos comienza con la colección de datos inicial y continua con las actividades que le permiten familiarizar primero con los datos, identificar los problemas de calidad de datos, descubrir los primeros conocimientos en los datos, y/o descubrir subconjuntos interesantes para formar hipótesis en cuanto a la información oculta.

c) **Preparación de datos:** La fase de preparación de datos cubre todas las actividades necesarias para construir el conjunto de datos final [los datos que serán provistos en las herramientas de modelado] de los datos en brutos iniciales. Las tareas de preparación de datos probablemente van a ser realizadas muchas veces y no en cualquier orden prescripto. Las tareas incluyen la selección de tablas, registros, y atributos, así como la transformación y la limpieza de datos para las herramientas que modelan.

d) **Modelado:** En esta fase, varias técnicas de modelado son seleccionadas y aplicadas, y sus parámetros son calibrados a valores óptimos. Típicamente hay varias técnicas para el mismo tipo de problema de minería de datos. Algunas técnicas tienen requerimientos específicos sobre la forma de datos. Por lo tanto, volver a la fase de preparación de datos es a menudo necesario.

e) **Evaluación:** En esta etapa en el proyecto, usted ha construido un modelo (o modelos) que parece tener la alta calidad de una perspectiva de análisis de datos.

Antes del proceder al despliegue final del modelo, es importante evaluar a fondo ello y la revisión de los pasos ejecutados para crearlo, para comparar el modelo correctamente obtenido con los objetivos de negocio. Un objetivo clave es determinar si hay alguna cuestión importante de negocio que no ha sido suficientemente considerada. En el final de esta fase, una decisión en el uso de los resultados de minería de datos debería ser obtenida.

f) **Despliegue:** La creación del modelo no es generalmente el final del proyecto. Incluso si el objetivo del modelo es de aumentar el conocimiento de los datos, el conocimiento ganado tendrá que ser organizado y presentado en el modo en el que el cliente pueda usarlo. Ello a menudo implica la aplicación de modelos "vivos" dentro de un proceso de toma de decisiones de una organización, por ejemplo, en tiempo real la personalización de página Web o la repetida obtención de bases de datos de mercadeo. Dependiendo de los requerimientos, la fase de desarrollo puede ser tan simple como la generación de un informe o tan compleja como la realización repetida de un proceso cruzado de minería de datos a través de la empresa. En muchos casos, es el cliente, no el analista de datos, quien lleva el paso de desarrollo. Sin embargo, incluso si el analista realizara

el esfuerzo de despliegue, esto es importante para el cliente para entender de frente que acciones necesita para ser ejecutadas en orden para hacer uso de los modelos creados actualmente.

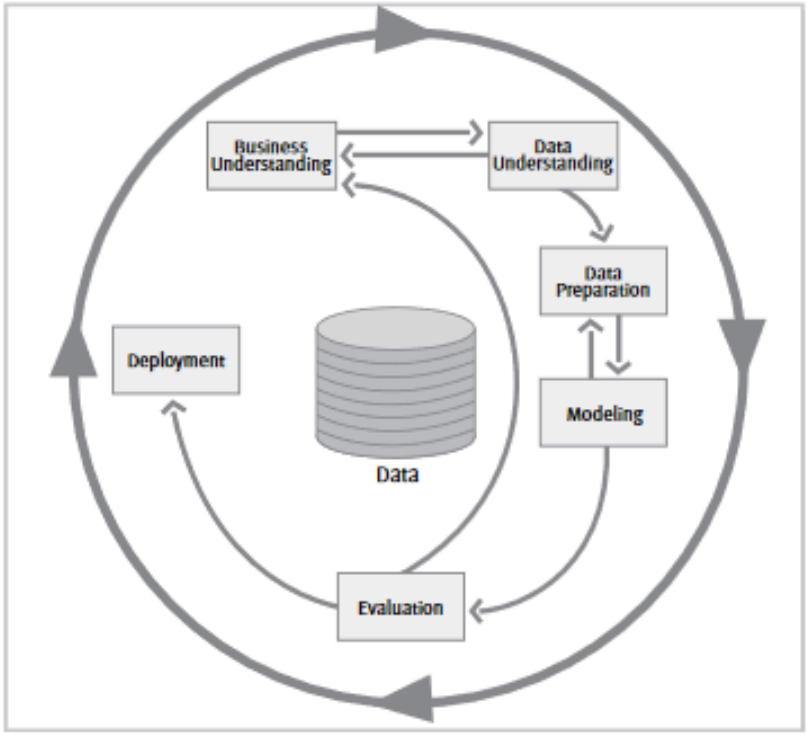


Figura 17: Ciclo de vida de CRISP-DM (Chapman et al., 2000)

2.2.8.2. SEMMA

El Instituto SAS define el concepto de minería de datos como el proceso de seleccionar (*selecting*), explorar (*exploring*), modificar (*modifying*), modelar (*modeling*) y valorar (*assessment*) grandes cantidades de datos con el objetivo de descubrir patrones desconocidos que puedan ser utilizados como una ventaja competitiva respecto a los competidores. Este proceso es resumido con la siglas SEMMA (Pérez y Santín, 2006).

Estas etapas permiten un fácil entendimiento del problema, tiene una estructura que permite la concepción, creación y evolución, ayudando

a presentar soluciones a los problemas planteados así como metas a encontrar mediante la minería de datos (Azevedo y Santos, 2008).

Considera cinco etapas en su proceso (SAS Institute, 1998) (Azevedo y Santos, 2008) (Albarrán y Salgado, 2013), el cual se muestra en la Figura 18:

a) **Muestrear (sample):** Consiste en obtener un extracto suficiente de los datos totales que pueda contener información significativa y pueda manipularse de forma rápida.

b) **Exploración (explore):** Es la búsqueda de tendencias no previstas y anomalías para entender el contenido y las ideas.

c) **Modificación (modify):** Consiste en la modificación de los datos mediante la creación, selección y transformación de variables que se adapten al proceso de selección del modelo.

d) **Modelado (model):** Modela los datos permitiendo que el software busque automáticamente la combinación de datos que permitan predecir de manera confiable la información requerida.

e) **Evaluación (assess):** Valora los datos mediante la evaluación de la utilidad y confiablidad de los hallazgos del proceso de minería de datos.

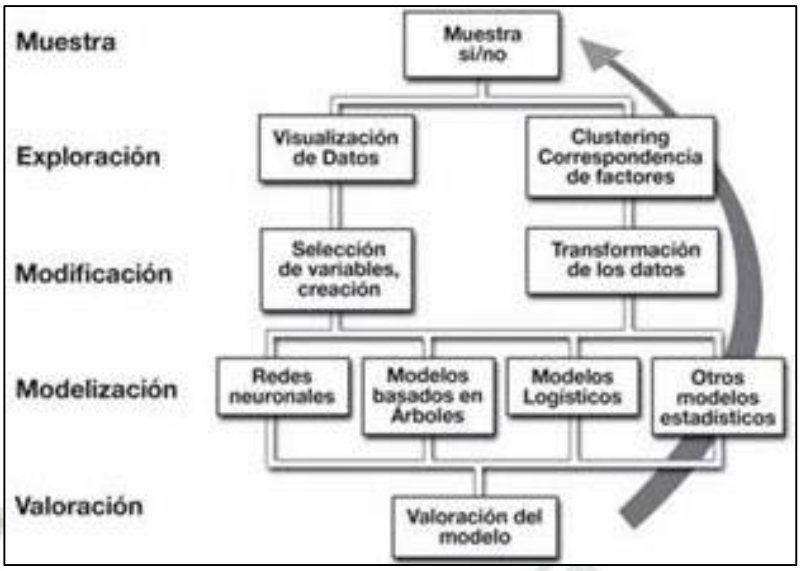


Figura 18: Ciclo de análisis SEMMA (Revista de Actuarios, 2006)

2.2.8.3. Comparación entre CRISP-DM y SEMMA

Las metodologías CRISP-DM y SEMMA son las más conocidas para el desarrollo de proyectos de minería de datos, ambas metodologías se organizan en fases, las cuales están interrelacionadas entre sí.

La metodología SEMMA se encuentra enfocada especialmente en aspectos técnicos, excluyendo actividades de análisis y comprensión del problema que se está abordando. Fue propuesta especialmente para trabajar con el software de minería de datos de la compañía SAS. Este producto organiza sus herramientas (llamadas "nodos") en base a las distintas fases que componen la metodología (Moine, Haedo y Gordillo, 2011).

La metodología SEMMA se centra más en las características técnicas del desarrollo del proceso, mientras que la metodología CRISP-DM, mantiene una perspectiva más amplia respecto a los objetivos empresariales del proyecto. La metodología SEMMA empieza

realizando un muestreo de datos, mientras que la metodología CRISP-DM empieza realizando un análisis del problema para su transformación en un problema técnico (Rodríguez et al., 2003).

Esta diferencia se puede apreciar claramente en la siguiente Figura 19.

SEMMA	CRISP
Orientado al desarrollo del proceso de MD	Orientado a los objetivos empresariales
Se inicia analizando los datos	Se inicia analizando los objetivos del negocio
Ligada a productos SAS	Metodología abierta y gratuita
	Orientado a una metodología de gestión de proyectos

Figura 19: Comparación entre SEMMA y CRISP-DM (Oporto, 2009)

En la Tabla 6 se muestra una comparación realizada entre CRISP-DM y SEMMA. CRISP-DM puede percibir para el proyecto de minería de datos los datos que debe buscar, cuáles son los objetivos que debe alcanzar y cuáles podrían ser los resultados esperados (Camargo y Silva, 2010).

Tabla 6: Comparación entre CRISP-DM y SEMMA (Camargo y Silva, 2010)

CRISP - DM	SEMMA
Abierta	Cerrada (Abierta en los aspectos generales únicamente)
Funciona en cualquier esquema que aplique minería de datos. Permite que cualquier sistema informático pueda seguir estos pasos	Funciona específicamente en SAS
Implica retroalimentación, es cíclica	Implica retroalimentación, es cíclica

CRISP - DM	SEMMA
Fases: Entendimiento del negocio, Entendimiento de los datos, Preparación de los datos, Modelado, Evaluado, Despliegue	Fases: Muestreo, Explorar, Modificar, Modelar, Evaluar
Metodología	Secuencia Lógica
Permite aplicar cualquier modelo estadístico	Está obligado a los modelos estadísticos que tenga incorporados la herramienta Enterprise Miner
Enfocada a resultados empresariales	Enfocada a resultados del proceso
Sigue el esquema propuesto en KDD	Sigue el esquema propuesto en KDD
Libre distribución	Distribución en clientes SAS

En la Figura 20, se observa que la metodología CRISP-DM se usa en mayor porcentaje para proyectos de minería de datos según encuestas realizadas por la comunidad KDnuggets, el cual es un sitio líder en análisis de negocio, big data, minería de datos.

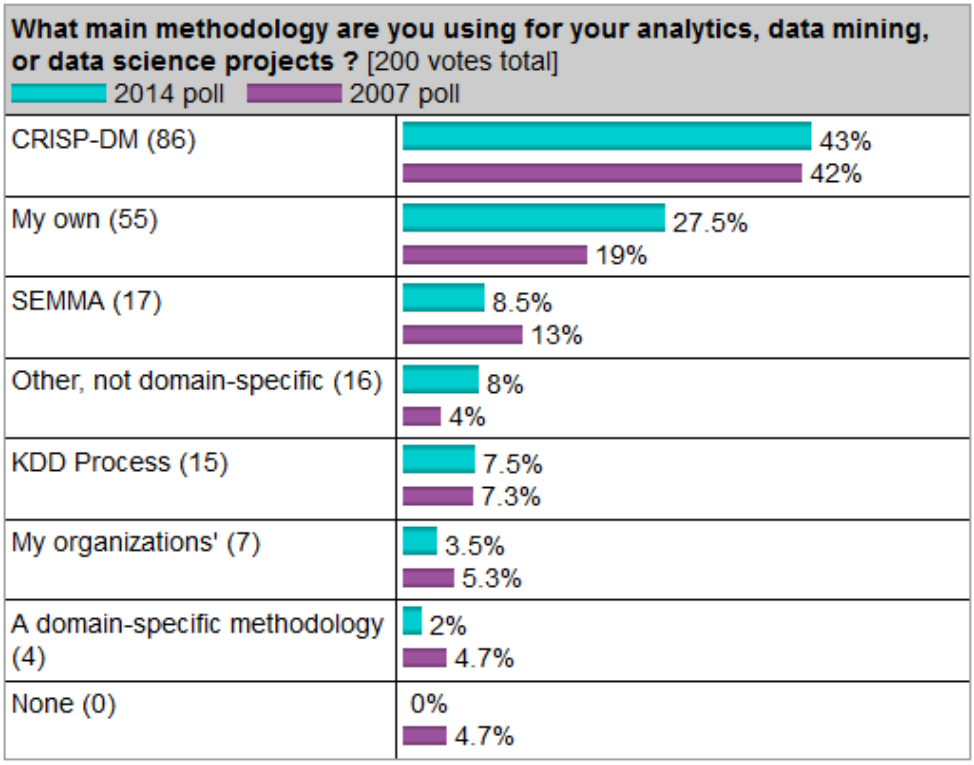


Figura 20: Metodologías más utilizadas en proyectos de minería de datos (KDnuggets, 2014)

2.2.9. Herramientas de Minería de Datos

Existen muchas herramientas libres y comerciales que se utilizan para el desarrollo de proyectos de minería de datos como son R, RapidMiner, WEKA, SQL Server Analysis Services, SAS Enterprise Miner, entre otros.

En la Figura 21 podemos observar que R es el software más utilizado para minería de datos con el 46.9% de votos según encuestas realizadas por la comunidad KDnuggets en Mayo del 2015.

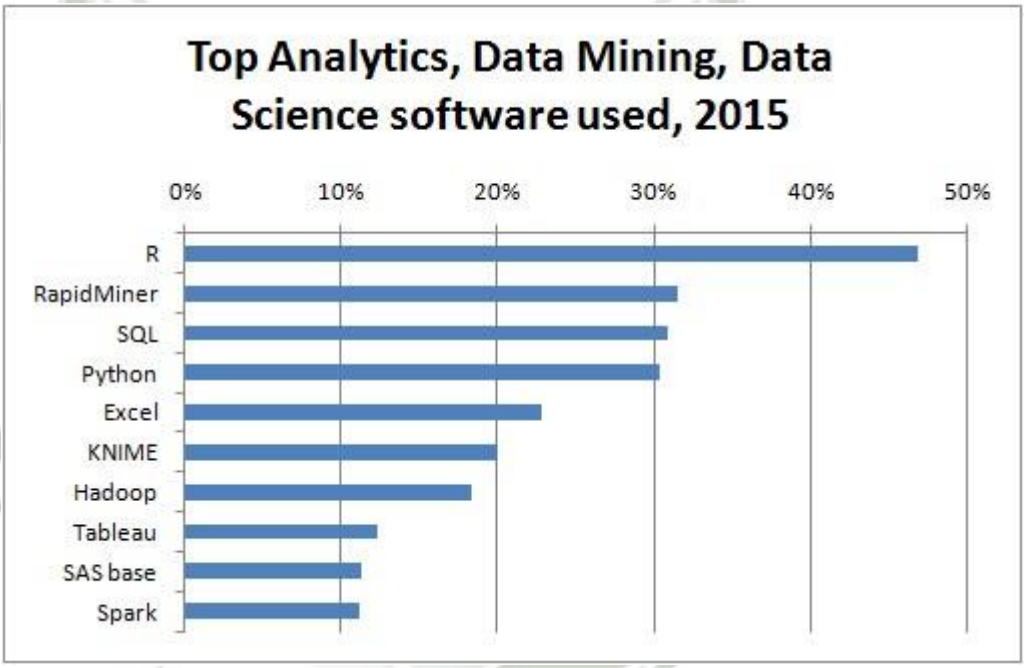


Figura 21: Software más utilizadas en Minería de Datos (KDnuggets, 2015)

2.2.9.1. R

R es un entorno y lenguaje de programación de software libre que se distribuye bajo la licencia de GNU GPL.

R es un lenguaje de programación y entorno de software de código abierto para computación y gráficos estadísticos. Proporciona múltiples técnicas para simulación, modelado lineal y no lineal,

análisis de series temporales, pruebas estadísticas clásicas, clasificación, agrupación en clústeres, etc. (García, 2013).

R es un lenguaje orientado a objetos: bajo este complejo término se esconde la simplicidad y flexibilidad de R. El hecho que R es un lenguaje de programación puede desaminar a muchos usuarios que piensan que no tienen "alma de programadores". Esto no es necesariamente cierto por dos razones. Primero R es un lenguaje interpretado (como Java) y no compilado (como C, C++, Fortran, Pascal), lo cual significa que los comandos escritos en el teclado son ejecutados directamente sin necesidad de construir ejecutables (Paradis, 2003).

Algunas de las características principales del lenguaje R son:

- Proporciona herramientas estadísticas y gráficas.
- Permite la aplicación de diferentes técnicas supervisadas y no supervisadas de minería de datos.
- Usuarios expertos pueden manipular los objetos de R directamente desde código desarrollado en C, también puede integrase a otros lenguajes de programación como Java, Python.
- Posee varios paquetes desarrollados por su comunidad de usuarios.
- Tiene su propio formato para la documentación basado en LaTeX.

- Puede utilizarse como herramienta de cálculo numérico, y resulta tan eficaz como otras herramientas específicas tales como GNU Octave y MATLAB.

2.2.9.2. RStudio

RStudio es un entorno de desarrollo integrado (IDE) para R (lenguaje de programación). Incluye una consola, editor de sintaxis que apoya la ejecución de código, así como herramientas para el trazado, la depuración y la gestión del espacio de trabajo. RStudio está disponible para Windows, Mac y Linux o para navegadores conectados a RStudio Server o RStudio Server Pro (RStudio, 2015).

RStudio permite el análisis de datos y desarrollo con R.

Características

RStudio es el primer entorno de desarrollo integrado para R. Está disponible en open source y ediciones comerciales. Tiene las siguientes características (RStudio, 2015):

- Un IDE que fue construido sólo para R
  - Resaltado de sintaxis, auto completado de código y sangría inteligente.
  - Ejecutar código R directamente desde el editor de código fuente.
  - Salto rápido a las definiciones de funciones.
- Colaboración
  - Ayuda y documentación integradas de R.

- Fácil administración de múltiples directorios de trabajo mediante proyectos.
- Navegación en espacios de trabajo y visor de datos.

- Potente autoría y depuración.
  - Depurador interactivo para diagnosticar y corregir los errores rápidamente.
  - Herramientas de desarrollo de paquetes extensas.
  - Autoría con Sweave y R Markdown

2.2.9.3. RapidMiner

RapidMiner es un programa informático para el análisis y minería de datos. Permite el desarrollo de procesos de análisis de datos mediante el encadenamiento de operadores a través de un entorno gráfico. Se usa en investigación, educación, capacitación, creación rápida de prototipos y en aplicaciones empresariales (Wikipedia, 2015).

RapidMiner es una herramienta de Minería de Datos ampliamente usada y probada a nivel internacional en aplicaciones empresariales, de gobierno y academia. Implementa más de 500 técnicas de pre-procesamiento de datos, modelación predictiva y descriptiva, métodos de prueba de modelos, visualización de datos, etc. (Microsystem, 2015)

Características

Las principales características de RapidMiner son (Wikipedia, 2015):

- Desarrollado en Java.

- Proporciona más de 500 operadores orientados al análisis de datos, incluyendo los necesarios para realizar operaciones de entrada y salida, pre procesamiento de datos y visualización.
- Permite utilizar los algoritmos incluidos en Weka.
- Multiplataforma.
- Representación interna de los procesos de análisis de datos en ficheros XML.
- Permite el desarrollo de programas a través de un lenguaje de script.
- Puede usarse de diversas maneras:
  - A través de un GUI.
  - En línea de comandos.
  - En batch (lotes).
  - Desde otros programas a través de llamadas a sus bibliotecas.
- Extensible.
- Incluye gráficos y herramientas de visualización de datos.
- Dispone de un módulo de integración con R.

2.2.9.4. WEKA

Weka (Waikato Environment for Knowledge Analysis - Entorno para Análisis del Conocimiento de la Universidad de Waikato) es una plataforma de software para aprendizaje automático y minería de datos escrito en Java. WEKA es un software libre distribuido bajo licencia GNU GPL. Contiene una colección de herramientas de visualización y algoritmos para análisis de datos y modelado

predictivo, unidos a una interfaz gráfica de usuario para acceder fácilmente a sus funcionalidades (Borao, 2013).

Weka es una colección de algoritmos de aprendizaje automático para tareas de minería de datos. Los algoritmos pueden ser aplicados directamente a un conjunto de datos o llamados desde su propio código Java. Weka contiene herramientas para pre-procesamiento de datos, clasificación, regresión, clustering, reglas de asociación y visualización. También es adecuado para desarrollar nuevos esquemas de aprendizaje de máquinas (WEKA, 2016).

Características

Las principales características de Weka son (Wikipedia, 2016):

- Disponible libremente bajo la licencia pública general de GNU.
- Portable ya que está implementado en JAVA.
- Puede ejecutarse en las plataformas de Windows, Mac y Linux.
- Contiene una extensa colección de técnicas para pre procesamiento de datos y modelado.
- Permite realizar pre-procesamiento de datos, clasificación, regresión, clustering, reglas de asociación y visualización.
- Interfaz gráfica de usuario amigable.

2.2.9.5. SQL Server Analysis Services (SSAS)

SQL Server Analysis Services (SSAS) ofrece funciones de procesamiento analítico en línea (OLAP) y minería de datos para

aplicaciones de Business Intelligence. En el caso de las aplicaciones de minería de datos, Analysis Services permite diseñar, crear y visualizar modelos de minería de datos que se construyen a partir de otros orígenes de datos mediante el uso de una gran variedad de algoritmos de minería de datos estándar del sector. (Microsoft, 2005)

SQL Server ha sido un líder en el análisis predictivo desde la versión en el año 2000 al proporcionar la minería de datos en Analysis Services. La combinación de la minería de datos de Integration Services, Reporting Services y SQL Server proporciona una plataforma integrada para el análisis predictivo en la que se incluye la limpieza de los datos, la preparación, el aprendizaje automático y la generación de informes. SQL Server En la minería de datos se incluyen varios algoritmos estándar, como los modelos de clústeres EM y K-means, redes neuronales, regresión logística y regresión lineal, árboles de decisión y clasificadores de naive Bayes. Todos los modelos tienen visualizaciones integradas para ayudarle a desarrollar, restringir y evaluar los modelos. Integrar la minería de datos en una solución de inteligencia empresarial le ayudará a tomar decisiones inteligentes sobre problemas complejos (Microsoft, 2016).

Características:

SQL Server proporciona las siguientes características para las soluciones integradas de minería de datos (Microsoft, 2016):

- **Varios orígenes de datos:** Puede usar cualquier origen de datos tabulares para la minería de datos, incluidas hojas de

cálculo y archivos de texto. También puede minar con facilidad cubos OLAP creados en Analysis Services.

- **Características integradas de limpieza de datos, administración de datos y generación de informes:** Integration Services proporciona herramientas para la generación de perfiles y la limpieza de los datos. Puede crear procesos ETL para limpiar datos al preparar el modelado.

- **Varios algoritmos personalizables:** Además de proporcionar algoritmos como la agrupación en clústeres, las redes neuronales y los árboles de decisión, la minería de datos de SQL Server le permite desarrollar sus propios complementos con algoritmos personalizados.

- **Infraestructura de prueba del modelo:** Pruebe los modelos y los conjuntos de datos usando herramientas estadísticas tan importantes como la validación cruzada, las matrices de clasificación, los gráficos de mejora respecto al modelo predictivo y los gráficos de dispersión. Cree y administre fácilmente conjuntos de prueba y entrenamiento.

- **Consultas y obtención de detalles:** La minería de datos de SQL Server proporciona el lenguaje DMX para integrar las consultas de predicción en las aplicaciones. También puede recuperar estadísticas detalladas y patrones de los modelos, y obtener detalles de datos de casos.

- **Herramientas de cliente:** Además de los estudios de desarrollo y diseño proporcionados por SQL Server, puede usar los Complementos de minería de datos para Excel para crear, consultar y examinar los modelos. O bien crear clientes personalizados, incluidos servicios web.

- **Compatibilidad con el lenguaje de scripting y API administrada:** Todos los objetos de minería de datos son completamente programables. El scripting es posible mediante MDX, XMLA o las extensiones de PowerShell para Analysis Services. Use el lenguaje DMX (Extensiones de minería de datos) para crear rápidamente consultas y scripts.

- **Seguridad e implementación:** Proporciona seguridad basada en roles a través de Analysis Services, incluidos permisos distintos para la obtención de detalles del modelo y los datos de la estructura. Fácil implementación de modelos en otros servidores, de forma que los usuarios puedan tener acceso a los patrones o realizar predicciones.

2.2.9.6. SAS Enterprise Miner

SAS Enterprise Miner es una herramienta comercial de minería de datos desarrollada por SAS Institute.

SAS Enterprise Miner es una solución para crear modelos predictivos, descriptivos y precisos sobre grandes volúmenes de datos por medio de diferentes fuentes, ofrece muchas características y funciones para

los analistas de negocio permitiendo que puedan modelar datos. Proporciona una visión que impulsa una mejor toma de decisiones, puede agilizar el proceso de minería de datos para desarrollar de manera rápida modelos, permitiendo comprender las relaciones claves y encontrar los patrones más importantes (DTyOC, 2015).

SAS Enterprise Miner es el módulo de software SAS para Data Mining. Permite crear modelos predictivos y descriptivos de minería de datos orientados a los negocios en un proceso transparente, permitiendo a todo el equipo colaborar más eficientemente. Incluye una interfaz de usuario intuitiva que incorpora principios de diseño comunes establecidos para el software SAS y herramientas de navegación adicionales para moverse fácilmente alrededor del espacio de trabajo (Qualex Consulting Services, 2013).

Las características de SAS Enterprise Miner según (DTyOC, 2015) son:

- Fácil de usar.
- Procesamiento por lotes.
- Preparación de datos, el resumen y la exploración.
- El modelo predictivo y descriptivo.
- La integración de código abierto.
- Prestaciones de alto rendimiento.

CAPÍTULO 3: DESARROLLO DE LA PROPUESTA

Se utiliza la metodología CRISP-DM.

3.1. Comprensión del negocio

Es la primera fase que se encarga de la comprensión de los objetivos del proyecto desde un punto de vista institucional para entender mejor el problema a resolver, lo cual permitirá obtener mayores beneficios de la minería de datos.

- **Determinación de objetivos de negocio**

  La Escuela Profesional de Ingeniería de Sistemas (EPIS) de la Universidad Católica de Santa María (UCSM), busca ser líder en el Sur del país en educación privada de Ingeniería de Sistemas así como lograr rentabilidad de la Escuela Profesional.

  La EPIS ha podido determinar los siguientes factores claves de éxito que le permitirán alcanzar sus metas planteadas en el plan estratégico 2014-2018:

  - Excelencia en la formación de los alumnos.
  - Mejorar el rendimiento académico de los alumnos.
  - Formar profesionales con estudios, grados y títulos de acuerdo a los estándares internacionales y del mercado local e internacional.

- **Evaluación de la situación**

  La EPIS de la UCSM busca mejorar la formación de sus alumnos así como su rendimiento académico y lograr incrementar la cantidad de alumnos que permanezcan en la carrera.

  Es por esta razón que se requiere un estudio comparativo de técnicas no supervisadas de minería de datos que permitan elegir el algoritmo de clustering con la que se obtiene mejor calidad de agrupamiento para

realizar la segmentación de alumnos que permita realizar cursos de reforzamiento personalizada.

Los recursos necesarios para la investigación son:

- Materiales:
  - R Versión 3.2.3 para utilizar el lenguaje de programación R en la aplicación de técnicas no supervisadas de minería de datos, así como la creación de gráficos.
  - RStudio Versión 0.99.491, IDE (Ambiente de Desarrollo Integrado) para facilitar el uso del lenguaje R. Se empleará para realizar la segmentación de alumnos.
  - Pentaho Data Integration Versión 5.4.0.1-130 para desarrollar el proceso ETL (Extracción, Transformación y Carga).
- Humanos: La autora de la investigación, la directora de la EPIS de la UCSM, el asesor de la investigación.
- Datos: Se cuenta con los registros académicos de los alumnos de la EPIS de la UCSM correspondientes al semestre par 2014.

- **Determinar los objetivos de la Minería de Datos**

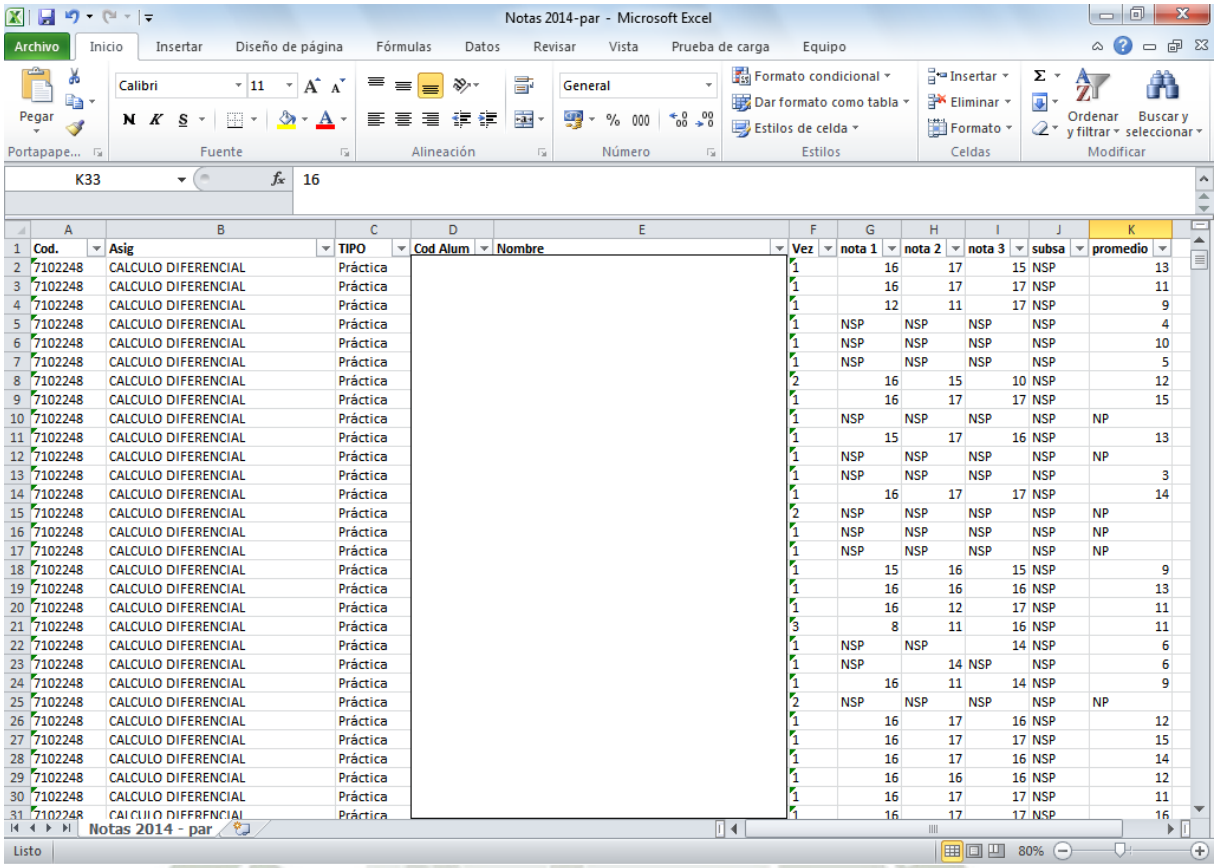
  Analizar y aplicar técnicas de minería de datos para realizar la segmentación de alumnos.

3.2. Comprensión de los datos

- **Recolección de datos iniciales**

  Los datos académicos iniciales fueron proporcionados por la EPIS de la UCSM, los cuales fueron extraídos del Sistema Académico de Ingreso de Notas de la UCSM.

Los datos académicos se encuentran en una hoja de cálculo de Excel llamada *Notas 2014 – par* como se muestra en la siguiente figura:



Notas 2014-par - Microsoft Excel

Cod.	Asig	TIPO	Cod Alum	Nombre	Vez	nota 1	nota 2	nota 3	subsa	promedio
7102248	CALCULO DIFERENCIAL	Práctica			1	16	17	15	NSP	13
7102248	CALCULO DIFERENCIAL	Práctica			1	16	17	17	NSP	11
7102248	CALCULO DIFERENCIAL	Práctica			1	12	11	17	NSP	9
7102248	CALCULO DIFERENCIAL	Práctica			1	NSP	NSP	NSP	NSP	4
7102248	CALCULO DIFERENCIAL	Práctica			1	NSP	NSP	NSP	NSP	10
7102248	CALCULO DIFERENCIAL	Práctica			1	NSP	NSP	NSP	NSP	5
7102248	CALCULO DIFERENCIAL	Práctica			2	16	15	10	NSP	12
7102248	CALCULO DIFERENCIAL	Práctica			1	16	17	17	NSP	15
7102248	CALCULO DIFERENCIAL	Práctica			1	NSP	NSP	NSP	NSP	NP
7102248	CALCULO DIFERENCIAL	Práctica			1	15	17	16	NSP	13
7102248	CALCULO DIFERENCIAL	Práctica			1	NSP	NSP	NSP	NSP	NP
7102248	CALCULO DIFERENCIAL	Práctica			1	NSP	NSP	NSP	NSP	3
7102248	CALCULO DIFERENCIAL	Práctica			1	16	17	17	NSP	14
7102248	CALCULO DIFERENCIAL	Práctica			2	NSP	NSP	NSP	NSP	NP
7102248	CALCULO DIFERENCIAL	Práctica			1	NSP	NSP	NSP	NSP	NP
7102248	CALCULO DIFERENCIAL	Práctica			1	NSP	NSP	NSP	NSP	NP
7102248	CALCULO DIFERENCIAL	Práctica			1	15	16	15	NSP	9
7102248	CALCULO DIFERENCIAL	Práctica			1	16	16	16	NSP	13
7102248	CALCULO DIFERENCIAL	Práctica			1	16	12	17	NSP	11
7102248	CALCULO DIFERENCIAL	Práctica			3	8	11	16	NSP	11
7102248	CALCULO DIFERENCIAL	Práctica			1	NSP	NSP	14	NSP	6
7102248	CALCULO DIFERENCIAL	Práctica			1	NSP	14	NSP	NSP	6
7102248	CALCULO DIFERENCIAL	Práctica			1	16	11	14	NSP	9
7102248	CALCULO DIFERENCIAL	Práctica			2	NSP	NSP	NSP	NSP	NP
7102248	CALCULO DIFERENCIAL	Práctica			1	16	17	16	NSP	12
7102248	CALCULO DIFERENCIAL	Práctica			1	16	17	17	NSP	15
7102248	CALCULO DIFERENCIAL	Práctica			1	16	17	16	NSP	14
7102248	CALCULO DIFERENCIAL	Práctica			1	16	16	16	NSP	12
7102248	CALCULO DIFERENCIAL	Práctica			1	16	17	17	NSP	11
7102248	CALCULO DIFERENCIAL	Práctica			1	16	17	17	NSP	16

Figura 22: Registro Académico de los alumnos del II Semestre de la EPIS correspondiente al semestre par 2014 (Fuente: Elaboración Propia)

Nota: Se ha ocultado el código y nombre de los alumnos para proteger sus datos personales.

- **Descripción de los datos**

Son en total 1142 registros para cursos correspondientes al II Semestre considerándose tanto los cursos prácticos como teóricos.

La hoja de cálculo de Excel llamada *Notas 2014 – par* cuenta con 11 campos de tipo TEXTO, los cuales son:

- Cod. (Código de curso)
- Asig (Nombre de la asignatura)

- TIPO (Práctica/Teórica)
- Cod Alum (Código del alumno)
- Nombre (Nombre del alumno)
- Vez (Cantidad de veces llevado el curso por el alumno)
- Nota 1 (Correspondiente a primera fase)
- Nota 2 (Correspondiente a segunda fase)
- Nota 3 (Correspondiente a tercera fase)
- Subsa (Nota de subsanación)
- Promedio (Promedio final del curso)

Son seis asignaturas que se dictan en el II Semestre:

- CALCULO DIFERENCIAL
- DESARROLLO HUMANO
- COMUNICACION ORAL Y ESCRITA
- FUNDAMENTOS DE PROGRAMACION
- PROGRAMACION I
- ESTRUCTURAS DISCRETAS I

- **Exploración de los datos**

  Explorando los datos se pudo observar que todos los campos son de tipo texto, se va a necesitar convertir los promedios a tipo numérico.

  Un problema que se pudo notar en los datos es que los registros están identificados por código del curso y código de alumno. Se requiere que los registros estén identificados únicamente por el código del alumno y se muestren sus notas de los diferentes cursos para cada uno de ellos en forma horizontal.

- **Verificación de la calidad de los datos**

  En esta tarea, se efectúan verificaciones sobre los datos, para determinar su consistencia.

  Examinando el registro académico de los alumnos se pudo observar lo siguiente:

  - Algunos alumnos llevaron solamente uno o dos cursos del II semestre, que no deberían ser considerados para la segmentación.
  - Se encontró como promedio final para algunos alumnos NP (No se presentó), los cuales deberían omitirse.

3.3. Preparación de los datos

En esta fase se procede a la preparación de los datos para adaptarlos a las técnicas de minerías de datos que posteriormente se utilizarán.

- **Seleccionar los datos**

  Los campos seleccionados que se utilizarán para la segmentación son:

  - Cod. (Código de curso)
  - Asig (Nombre de la asignatura)
  - TIPO (Práctica/Teórica)
  - Cod Alum (Código del alumno)
  - Nombre (Nombre del alumno)
  - Promedio (Promedio final del curso)

  Se considerarán solamente los cursos teóricos del II semestre, no se considerará los cursos de tipo práctica porque el promedio de la teoría ya está calculado con la nota práctica.

Se considerará únicamente a los alumnos que estén llevando todos los cursos del II semestre y cuyo promedio final sea diferente a NP (No se presentó).

- **Limpiar los datos**

  Se realizará las siguientes tareas:

  - **Des-normalización de fila:** Debido a que los cursos están distribuidos en las filas y estos deben convertirse en campos (columnas) mostrando como valores los promedios para cada alumno.
  - **Tratamiento de valores nulos:** Si el valor es nulo reemplazarlos con 0 (cero).
  - **Conversión de tipos:** El promedio de cada curso debe convertirse del tipo de dato texto a numérico (De String a Integer).
  - **Omitir código y nombre de los alumnos:** Para proteger sus datos personales, se utilizará en su lugar números generados de forma secuencial.

Proceso ETL

La herramienta empleada para los procesos ETL fue Pentaho.

Se utilizó el proceso ETL (Extracción, Transformación y Carga) para:

- Cargar los datos de los campos seleccionados de la hoja de cálculo del registro académico a una nueva base de datos que se creó en PostgreSQL.
- Limpiar los datos.

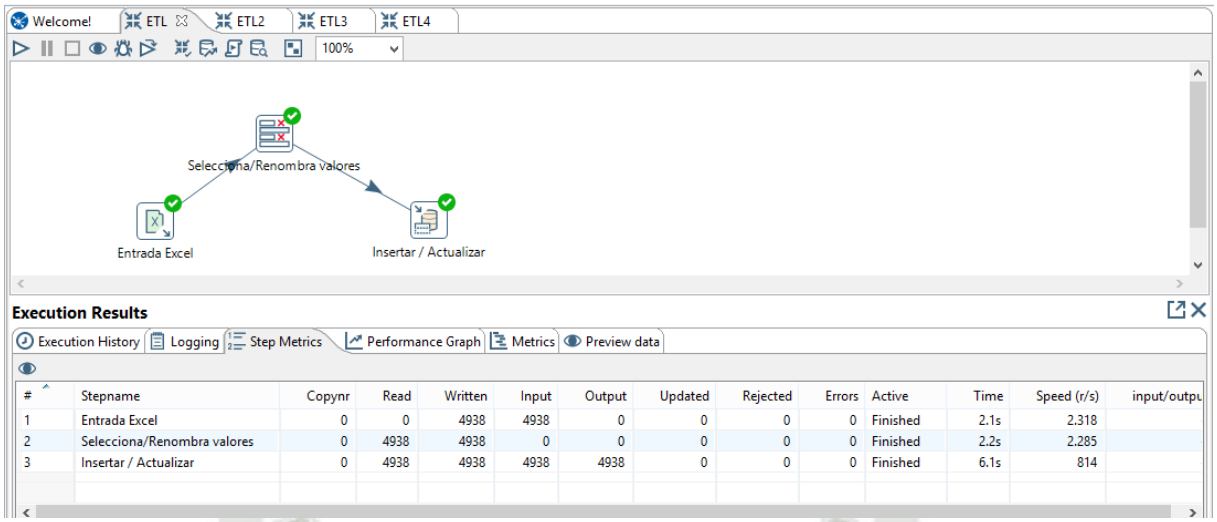


Figura 23: Proceso ETL para cargar los datos a una BD en PostgreSQL (Fuente: Elaboración Propia)

La extracción se realiza a partir del libro de Excel llamada *Notas 2014-par.xlsx*. Se utilizó la herramienta de transformación Select values para seleccionar los campos con los que se va a trabajar para la segmentación y renombrarlos como se muestra en la siguiente figura:

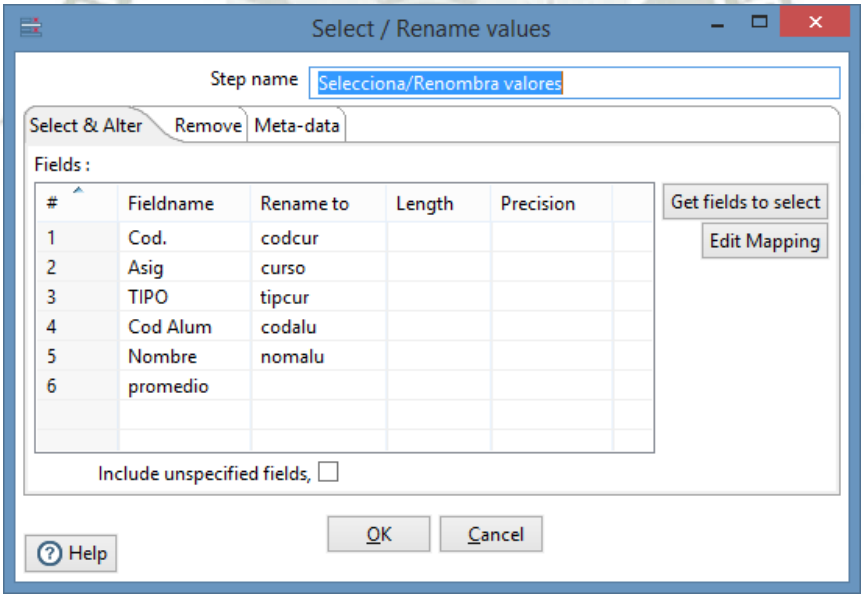
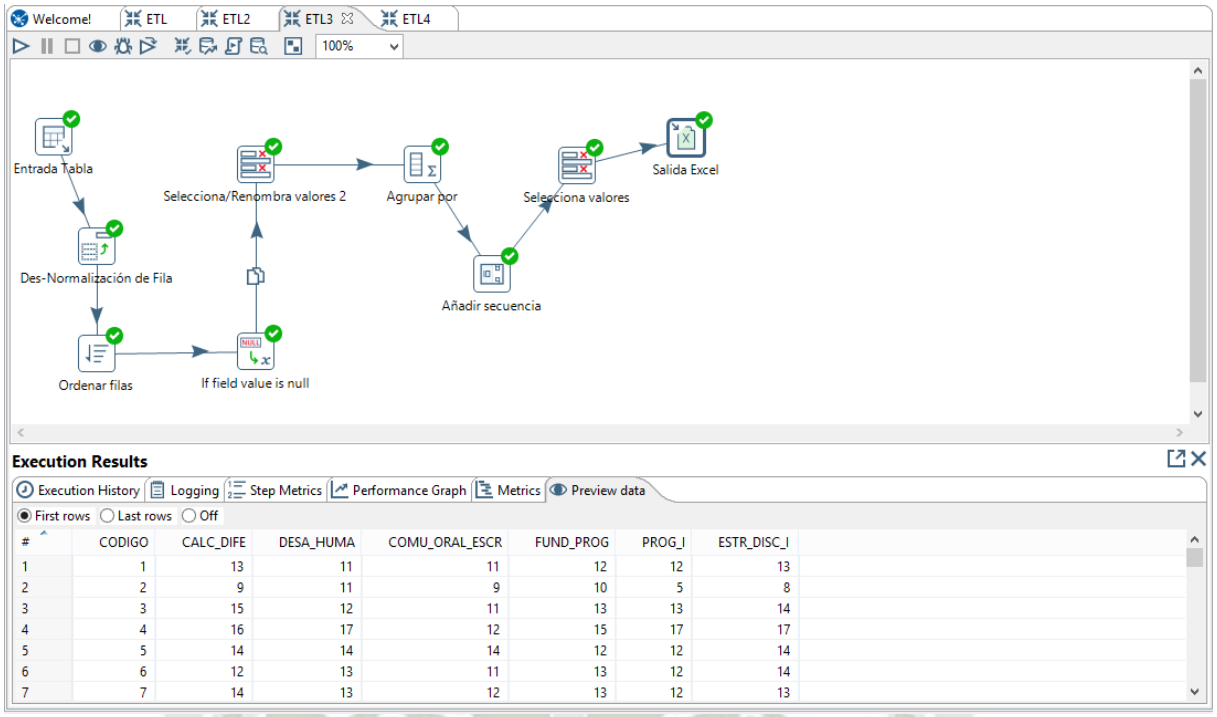


Figura 24: Seleccionar/Renombrar valores (Fuente: Elaboración Propia)

En el siguiente proceso ETL extraemos de la BD PostgreSQL los datos de los alumnos, realizamos transformaciones de limpieza de datos y lo cargamos a un libro de Excel llamado *Notas_IISem_2014.xls*



#	CODIGO	CALC_DIFE	DESA_HUMA	COMU_ORAL_ESCR	FUND_PROG	PROG_I	ESTR_DISC_I
1	1	13	11	11	12	12	13
2	2	9	11	9	10	5	8
3	3	15	12	11	13	13	14
4	4	16	17	12	15	17	17
5	5	14	14	14	12	12	14
6	6	12	13	11	13	12	14
7	7	14	13	12	13	12	13

Figura 25: Proceso ETL para reformatear y limpiar los datos (Fuente: Elaboración Propia)

Se utilizó una consulta SQL para extraer los datos de los alumnos que llevan todos los cursos teóricos del II semestre y cuyo promedio final sea diferente a NP como se muestra en la Figura 26.

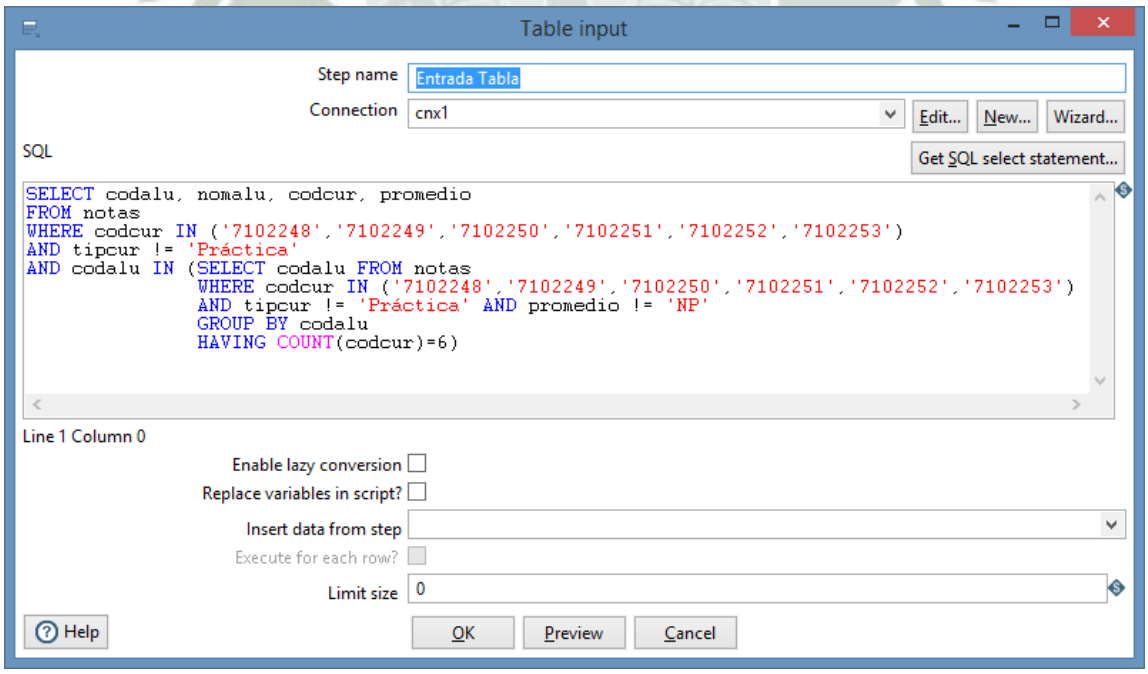


Figura 26: consulta SQL para extraer los datos de los alumnos (Fuente: Elaboración Propia)

Para convertir en campos (columnas) los cursos que están distribuidos en filas, mostrando como valores los promedios para cada alumno, se utilizó la herramienta de transformación *Des-normalización de filas* como se muestra en la Figura 27.

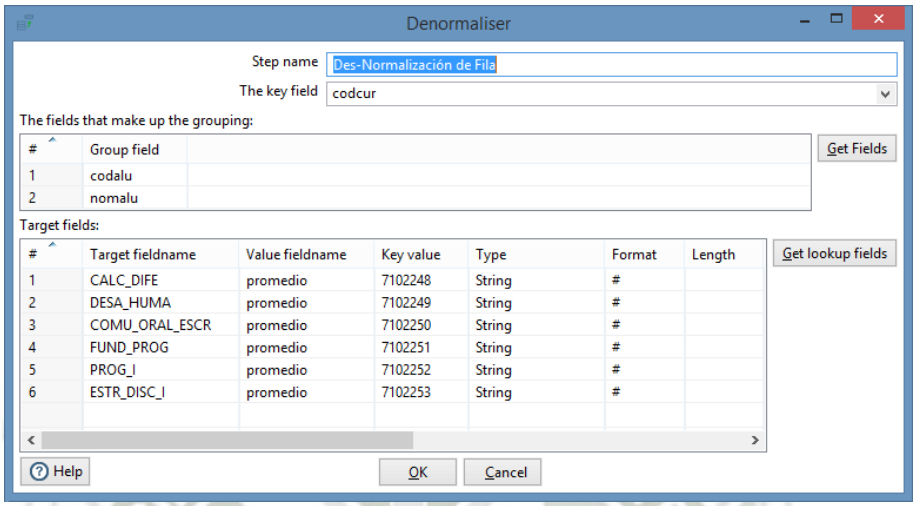


Figura 27: Des-normalización de filas (Fuente: Elaboración Propia)

Luego se ordenó las filas por el campo *nomalu* (Nombre del alumno).

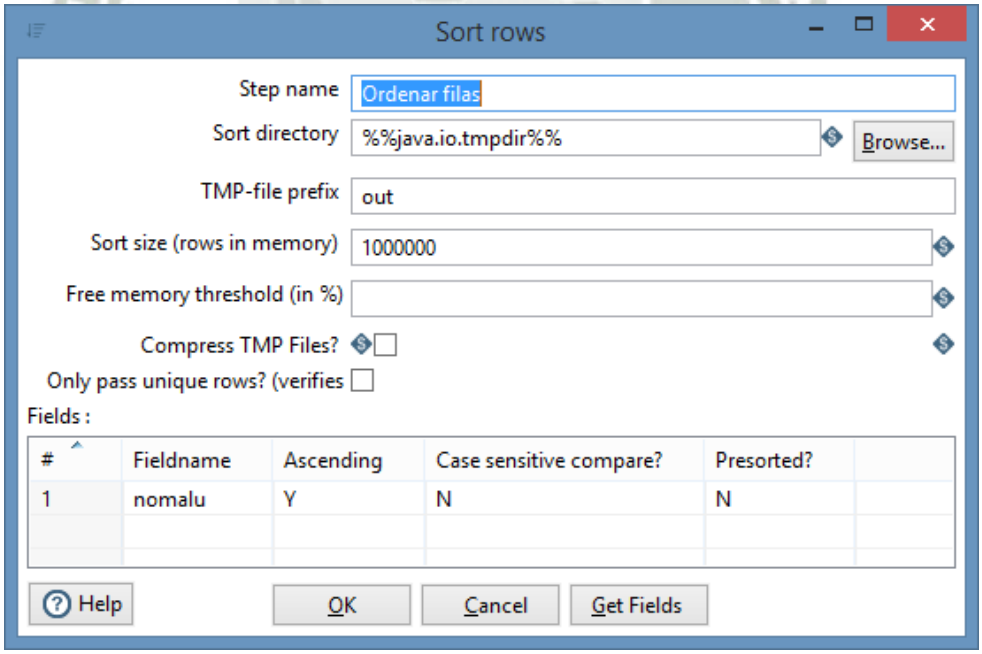


Figura 28: Ordenación de filas (Fuente: Elaboración Propia)

Se reemplazó los valores nulos con 0 (cero)

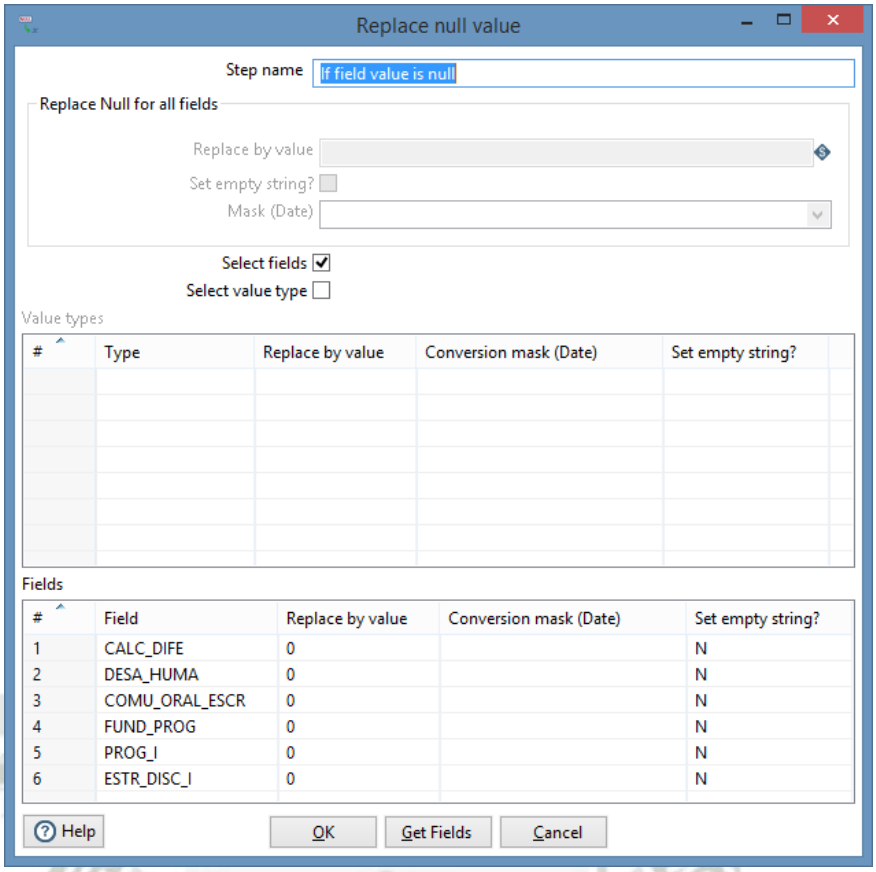


Figura 29: Tratamiento de valores nulos (Fuente: Elaboración Propia)

Se convirtió el promedio de cada curso del tipo de dato texto a numérico (De String a Integer).

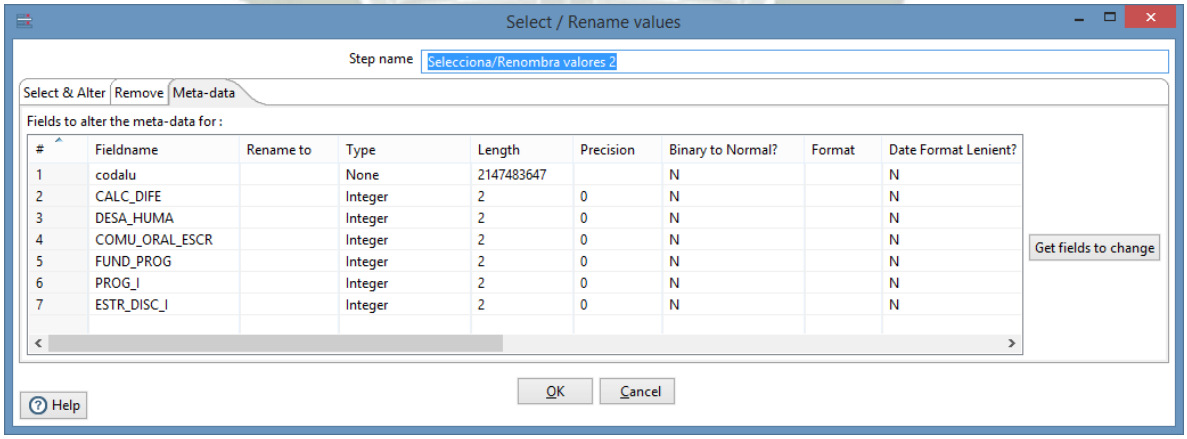


Figura 30: Conversión de tipo de datos (Fuente: Elaboración Propia)

Se generó código secuencial para identificar a los alumnos como se muestra en la Figura 31.

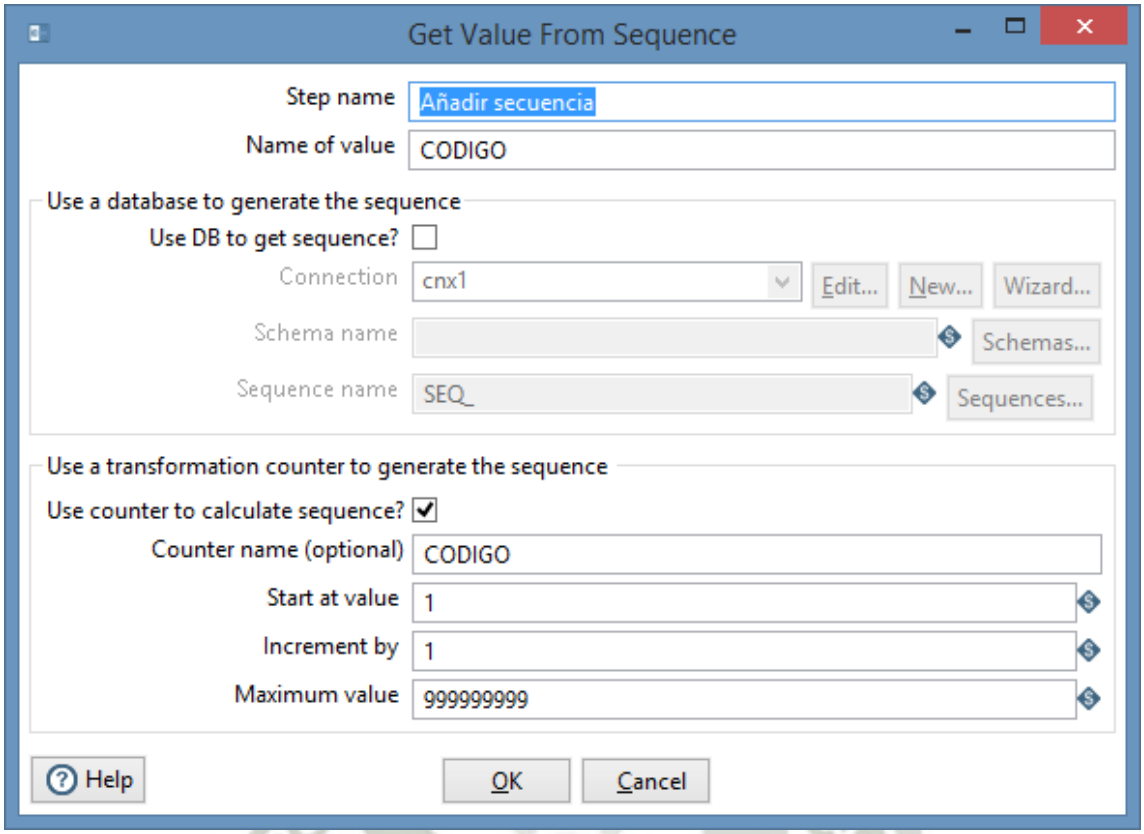


Figura 31: Añadir secuencia (Fuente: Elaboración Propia)

Se seleccionaron los campos en el orden respectivo para luego ser cargados a un libro de Excel llamado *Notas_IISem_2014.xls*.

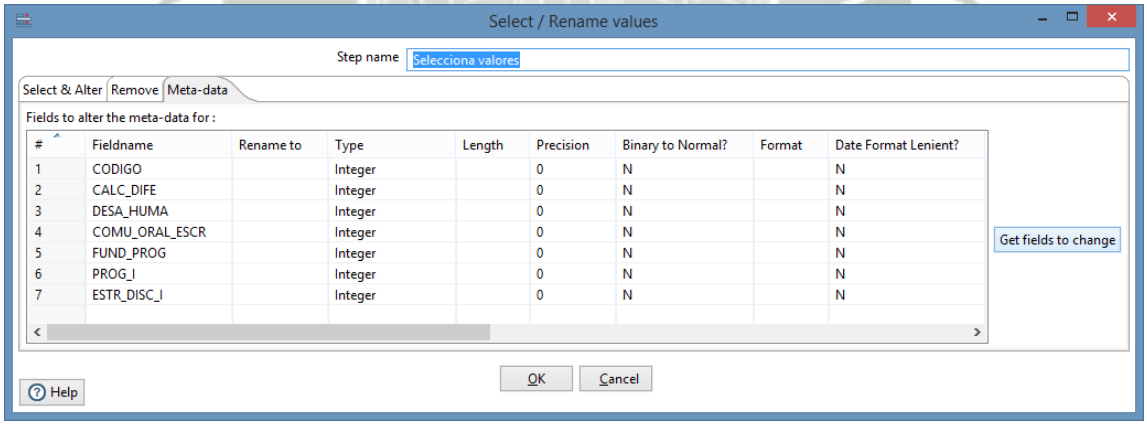


Figura 32: Seleccionar valores (Fuente: Elaboración Propia)

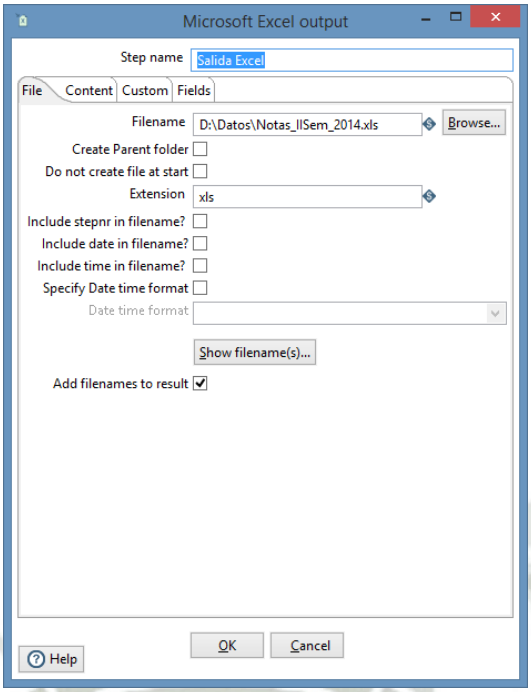


Figura 33: Cargar datos en Excel (Fuente: Elaboración Propia)

- **Construir datos**

  Los datos se guardaron como un archivo .CSV (Delimitado por comas)

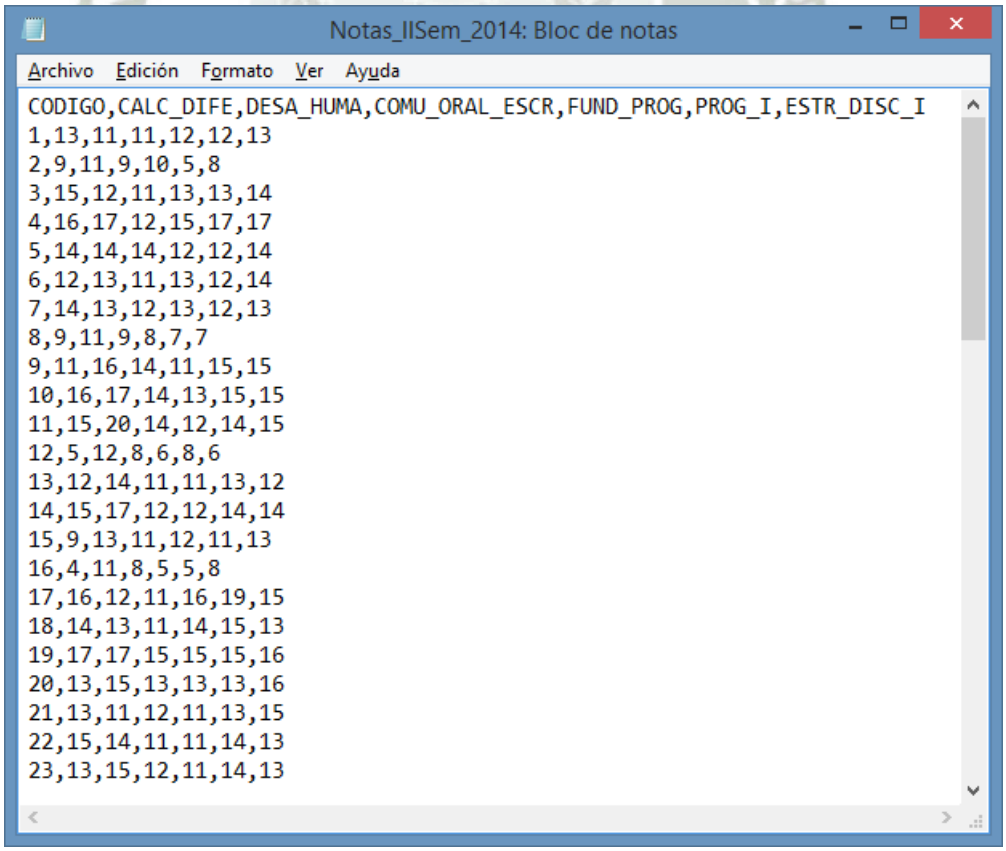


Figura 34: Parte del Archivo CSV (Fuente: Elaboración Propia)

El archivo *Notas_IISem_2014.csv* es el que se va a utilizar en la herramienta RStudio para realizar el estudio comparativo de técnicas no supervisadas de minería de datos para la segmentación de alumnos utilizando el lenguaje R. Los campos de los que consta son:

- CODIGO (Identificador del alumno)
- CALC_DIFE (Cálculo Diferencial)
- DESA_HUMA (Desarrollo Humano)
- COMU_ORAL_ESCR (Comunicación Oral y Escrita)
- FUND_PROG (Fundamentos de Programación)
- PROG_I (Programación I)
- ESTR_DISC_I (Estructuras Discretas I)

- **Formateo de los datos**

  El formateo de los datos se hizo reordenando los campos y dándole nombres más entendibles a dichos campos.

3.4. Modelado

En esta fase se eligen las técnicas de minería de datos para realizar la segmentación de alumnos.

- **Selección de la técnica de modelado**

  Para poder realizar la segmentación de alumnos se requiere utilizar técnicas de aprendizaje no supervisado que permitan realizar el clustering (agrupamiento).

  Las técnicas seleccionadas a utilizar son:

  - Clustering jerárquico aglomerativo
  - K-means
  - PAM

- **Construcción del Modelo**

  La construcción del modelo de las técnicas seleccionadas para la segmentación de alumnos se realizó aplicando técnicas de minería de datos utilizando el lenguaje R.

Primero se cargan los datos

```
setwd("D:/Datos")
notas<-read.table("Notas_IISem_2014.csv",header=TRUE,sep=",", row.names=1)
```

Verifico el paso anterior mostrando los primeros registros

```
> head(notas)
  CALC_DIFE DESA_HUMA COMU_ORAL_ESCR FUND_PROG PROG_I ESTR_DISC_I
1        13        11             11        12     12          13
2         9        11              9        10      5           8
3        15        12             11        13     13          14
4        16        17             12        15     17          17
5        14        14             14        12     12          14
6        12        13             11        13     12          14
```

Muestro un resumen de los campos

```
> summary(notas)
   CALC_DIFE       DESA_HUMA      COMU_ORAL_ESCR    FUND_PROG        PROG_I       ESTR_DISC_I
 Min.   : 1.00   Min.   : 7.00   Min.   : 3.00   Min.   : 4.00   Min.   : 1.00   Min.   : 2.00
 1st Qu.:10.00   1st Qu.:11.00   1st Qu.:11.00   1st Qu.:11.00   1st Qu.:11.00   1st Qu.:12.00
 Median :12.00   Median :13.00   Median :11.00   Median :12.00   Median :13.00   Median :13.00
 Mean   :11.86   Mean   :13.33   Mean   :10.91   Mean   :11.35   Mean   :12.09   Mean   :12.41
 3rd Qu.:14.00   3rd Qu.:15.00   3rd Qu.:12.00   3rd Qu.:13.00   3rd Qu.:14.00   3rd Qu.:15.00
 Max.   :17.00   Max.   :20.00   Max.   :15.00   Max.   :17.00   Max.   :19.00   Max.   :17.00
```

Muestro la estructura de la hoja de datos (data frame) notas

```
> str(notas)
'data.frame':   69 obs. of  6 variables:
 $ CALC_DIFE     : int  13 9 15 16 14 12 14 9 11 16 ...
 $ DESA_HUMA     : int  11 11 12 17 14 13 13 11 16 17 ...
 $ COMU_ORAL_ESCR: int  11 9 11 12 14 11 12 9 14 14 ...
 $ FUND_PROG     : int  12 10 13 15 12 13 13 8 11 13 ...
 $ PROG_I        : int  12 5 13 17 12 12 12 7 15 15 ...
 $ ESTR_DISC_I   : int  13 8 14 17 14 14 13 7 15 15 ...
```

Luego se aplica las técnicas no supervisadas seleccionadas de minería de datos para la segmentación de alumnos:

- **Clustering jerárquico aglomerativo**

Utilizando el método ward:

```
clust1 <- hclust(dist(notas), method= "ward.D2")
```

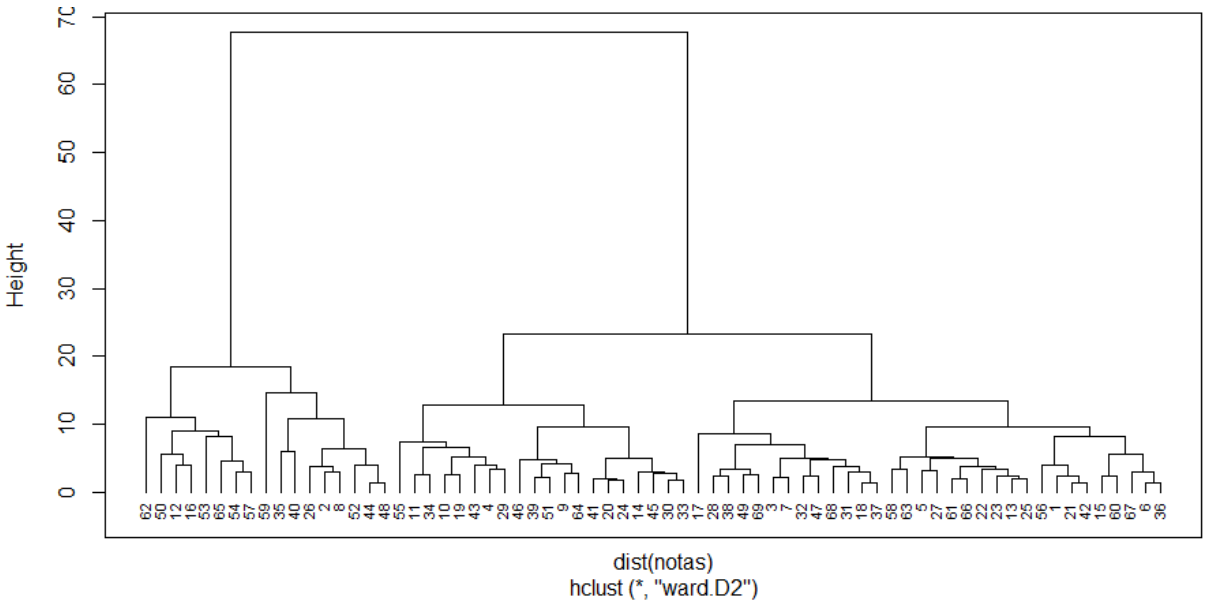


Figura 35: Dendrograma utilizando el método "ward" (Fuente: Elaboración Propia)

Se puede observar a partir del dendrograma generado por el método WARD varias agrupaciones de dos, tres o cuatro clusters como se muestra en las Figuras 36, 37 y 38 respectivamente:

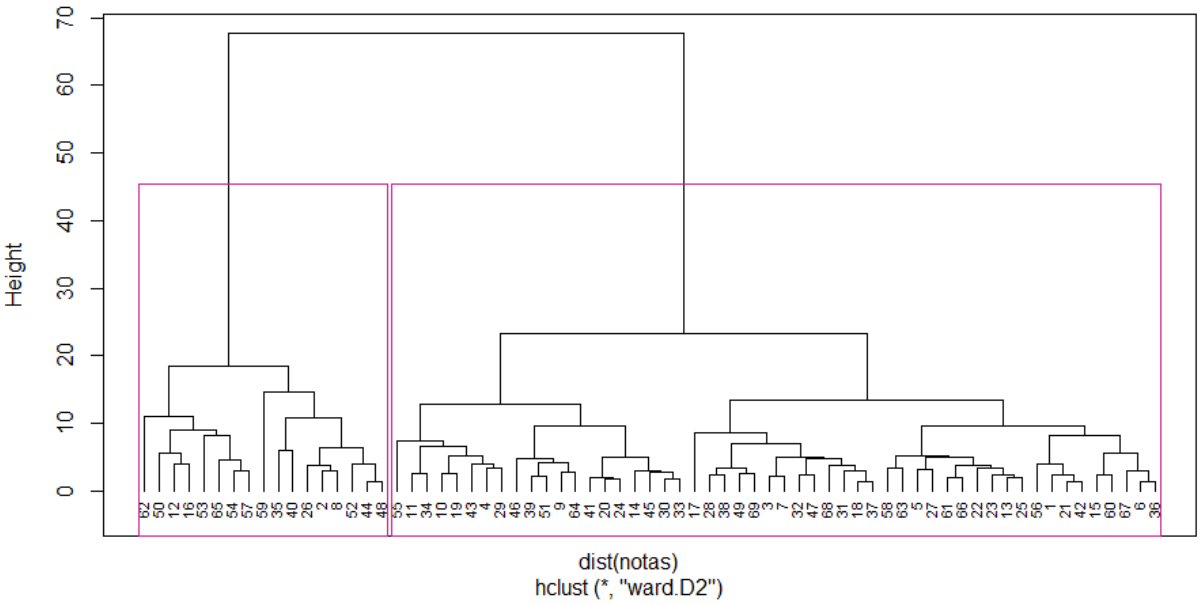


Figura 36: Agrupación jerárquica de dos grupos (Fuente: Elaboración Propia)

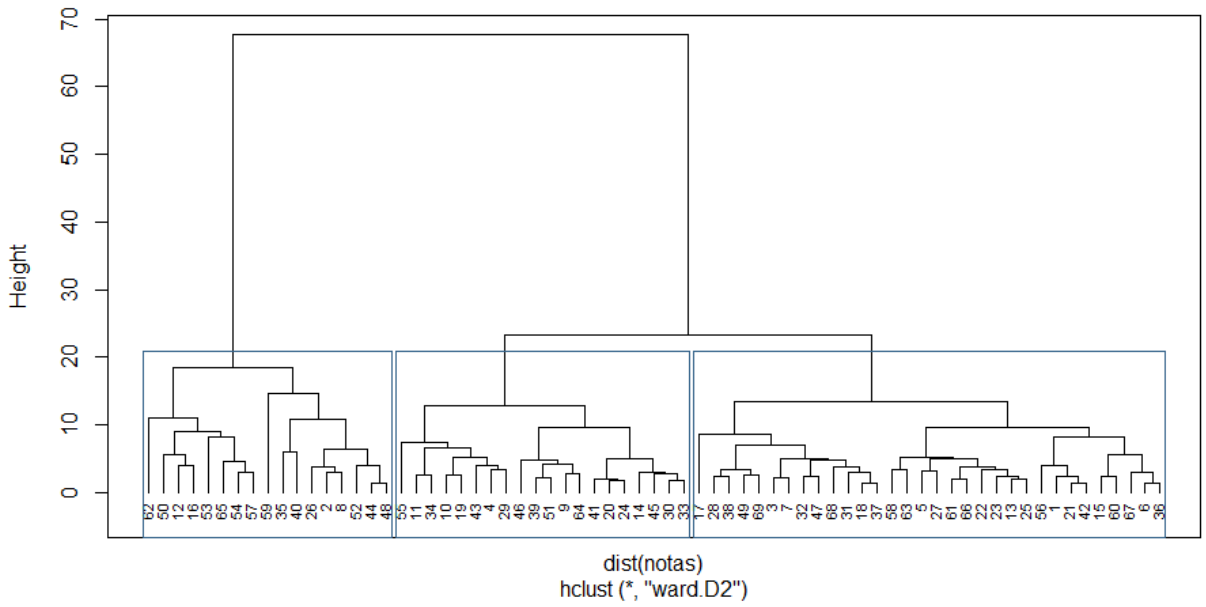


Figura 37: Agrupación jerárquica de tres grupos (Fuente: Elaboración Propia)

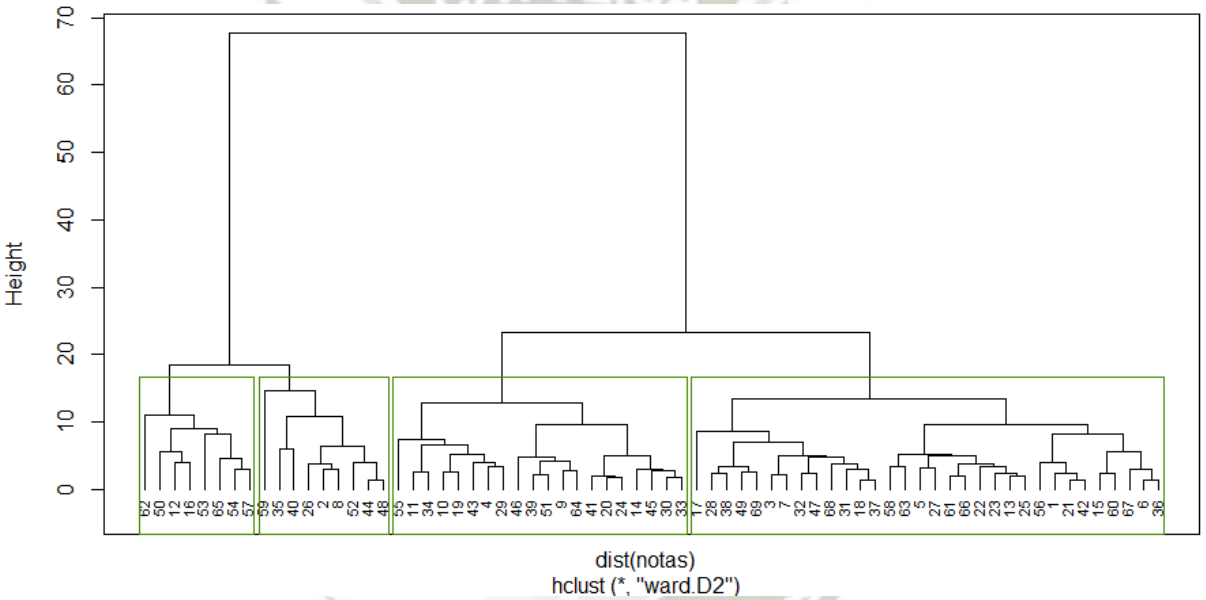


Figura 38: Agrupación jerárquica de cuatro grupos (Fuente: Elaboración Propia)

A continuación se muestra gráficos de barras construidos a partir de los centros de gravedad:

Se segmento en tres grupos ya que se quiere reforzar a los alumnos en los niveles básico, intermedio y avanzado.

```
> centros<-centers.hclust(notas,clust1,nclust=3)
> centros
          CALC_DIFE DESA_HUMA COMU_ORAL_ESCR FUND_PROG    PROG_I ESTR_DISC_I
Cluster 1 12.843750  12.84375      11.500000 12.093750 13.375000   13.343750
Cluster 2  7.294118  10.47059       7.470588  7.882353  7.117647    7.529412
Cluster 3 14.150000  16.55000      12.900000 13.100000 14.250000   15.050000
```

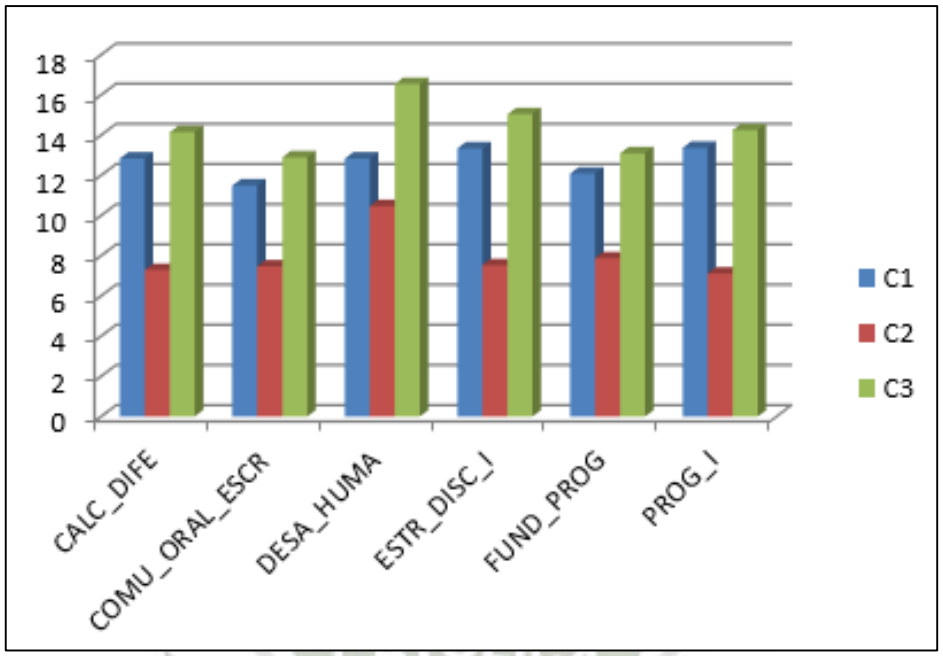


Figura 39: Promedio de notas en los clusters – Método Ward (Fuente: Elaboración Propia)

A continuación se muestra una comparación de los clusters

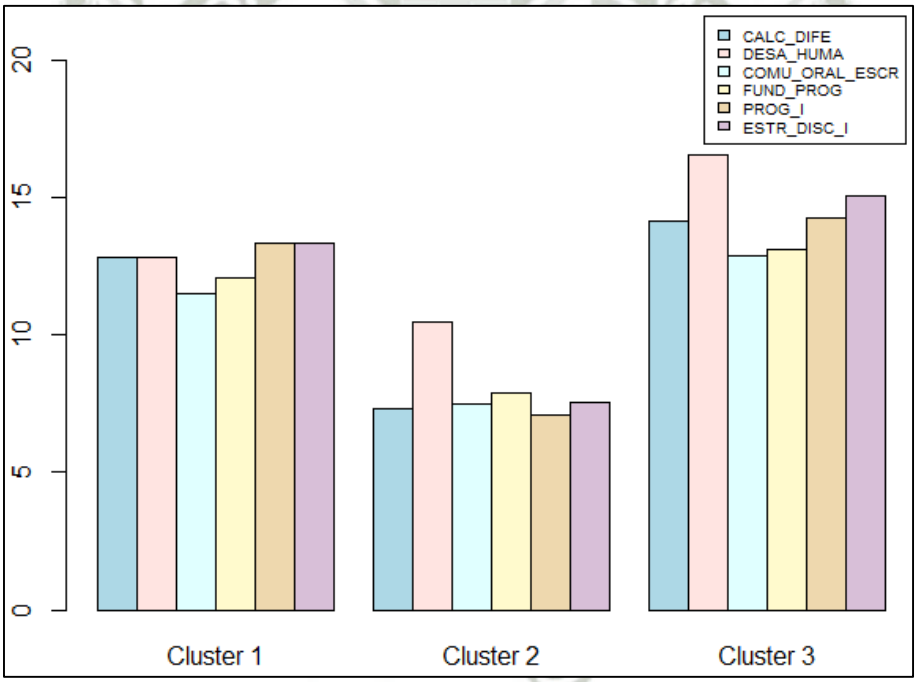


Figura 40: Comparación de clusters – Método ward (Fuente: Elaboración Propia)

Utilizando el método single (Agregación del salto mínimo):

```
clust1 <- hclust(dist(notas), method= "single")
```

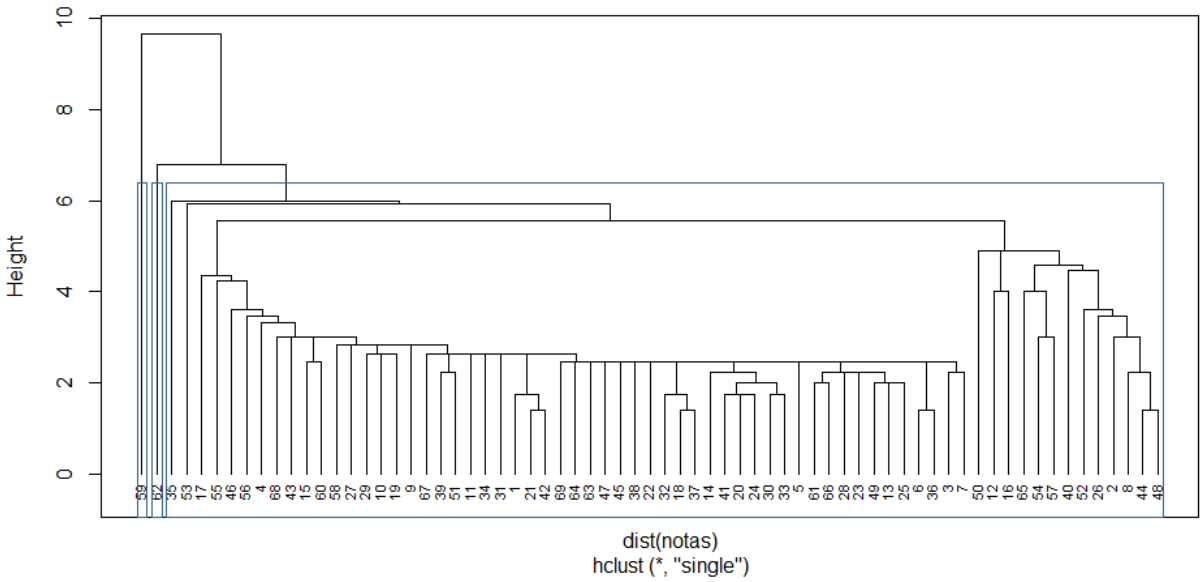


Figura 41: Dendrograma utilizando el método "single" (Fuente: Elaboración Propia)

Se puede observar a partir del dendrograma generado por el método single que no es muy recomendable para la agrupación de tres clusters.

Se segmento en tres grupos y se obtuvo los siguientes centros de gravedad

```
> centros<-centers.hclust(notas,clust1,nclust=3)
> centros
     CALC_DIFE DESA_HUMA COMU_ORAL_ESCR FUND_PROG   PROG_I ESTR_DISC_I
[1,]  12.04478  13.40299        11.1194  11.38806 12.26866    12.47761
[2,]   1.00000  11.00000         4.0000  12.00000 11.00000    12.00000
[3,]  10.00000  11.00000         4.0000   8.00000  1.00000     8.00000
```

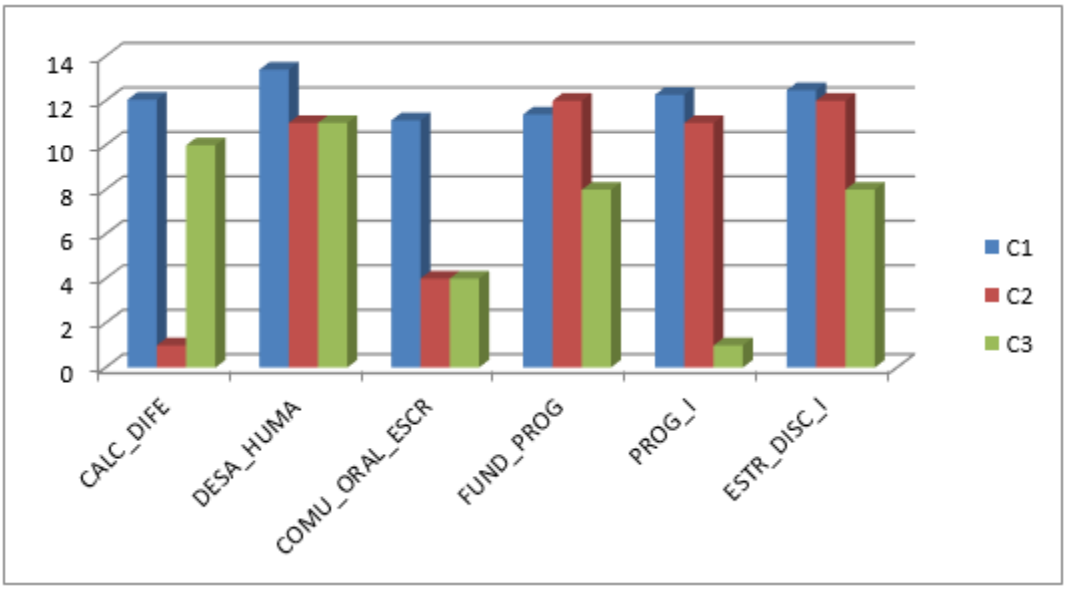


Figura 42: Promedio de notas en los clusters – Método single (Fuente: Elaboración Propia)

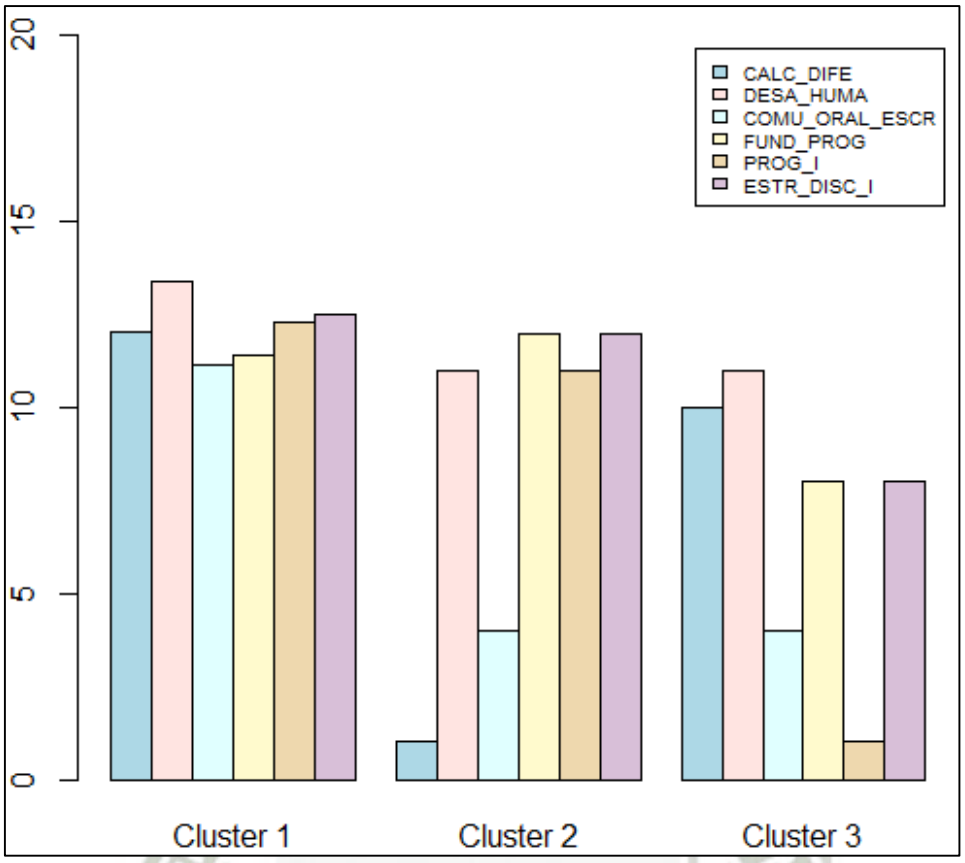


Figura 43: Comparación de clusters – Método single (Fuente: Elaboración Propia)

Utilizando el método complete (Agregación del salto máximo):

```
clust1 <- hclust(dist(notas), method= "complete")
```

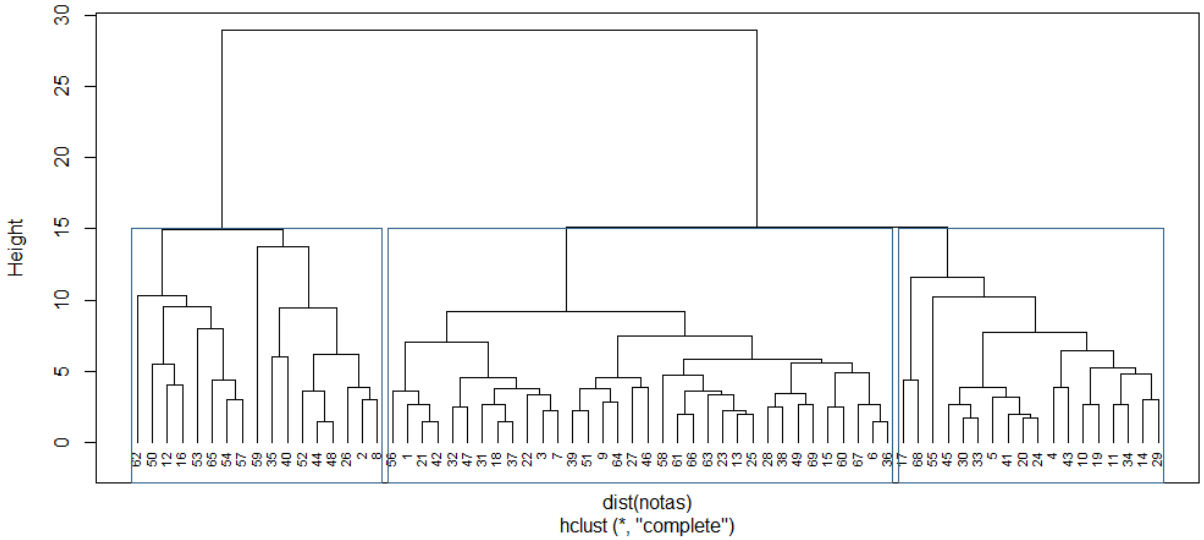


Figura 44: Dendrograma utilizando el método "complete" (Fuente: Elaboración Propia)

Se segmento en tres grupos y se obtuvo los siguientes centros de gravedad

```
> centros<-centers.hclust(notas,clust1,nclust=3)
> centros
     CALC_DIFE DESA_HUMA COMU_ORAL_ESCR FUND_PROG    PROG_I ESTR_DISC_I
[1,] 12.411765  13.41176       11.647059 11.941176 13.235294   13.382353
[2,]  7.294118  10.47059        7.470588  7.882353  7.117647    7.529412
[3,] 15.111111  15.88889       12.777778 13.500000 14.611111   15.166667
```

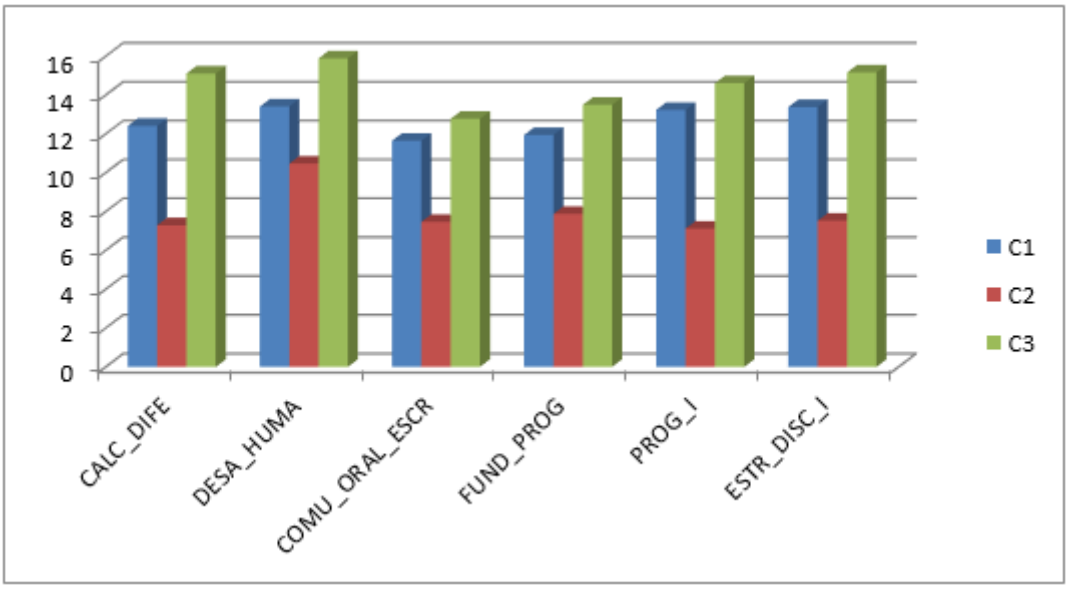


Figura 45: Promedio de notas en los clusters – Método complete (Fuente: Elaboración Propia)

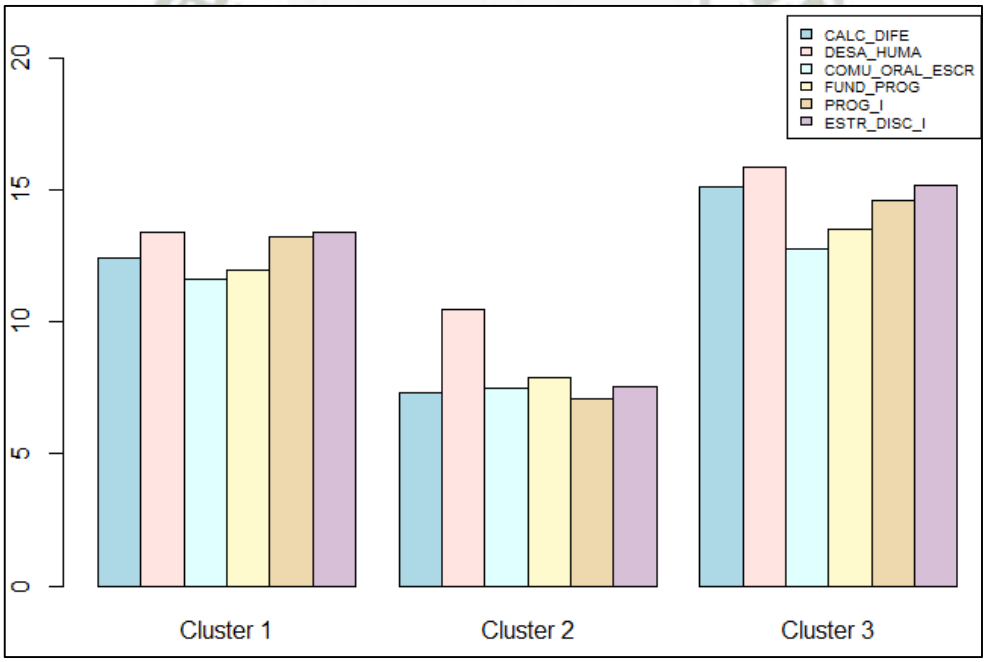


Figura 46: Comparación de clusters – Método complete (Fuente: Elaboración Propia)

Utilizando el método average (Agregación del Promedio):

```
clust1 <- hclust(dist(notas), method= "average")
```

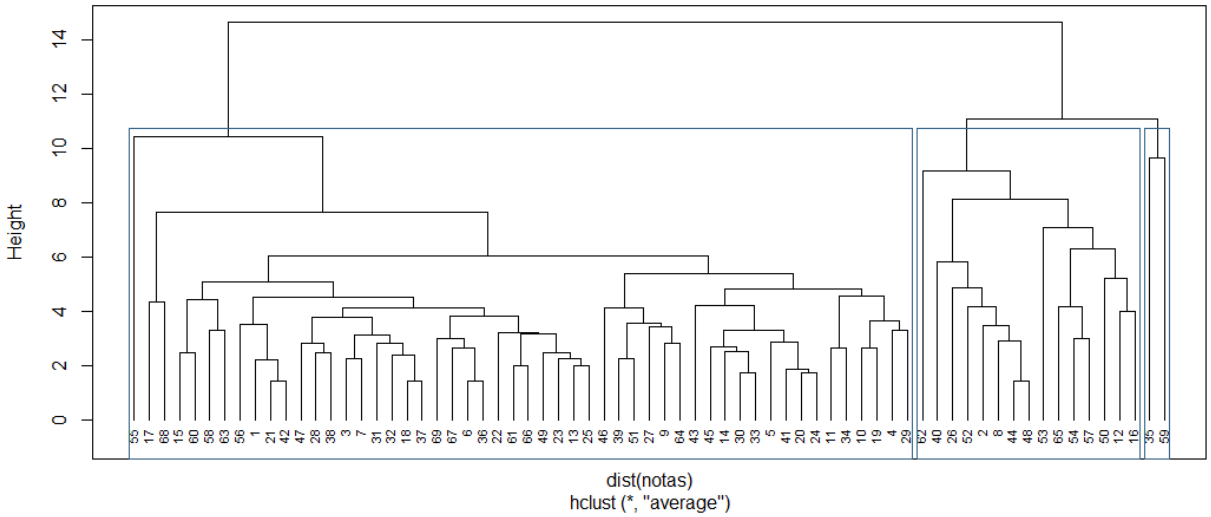


Figura 47: Dendrograma utilizando el método "average" (Fuente: Elaboración Propia)

Se segmento en tres grupos y se obtuvo los siguientes centros de gravedad

```
> centros<-centers.hclust(notas,clust1,nclust=3)
> centros
     CALC_DIFE DESA_HUMA COMU_ORAL_ESCR FUND_PROG     PROG_I ESTR_DISC_I
[1,] 13.346154  14.26923        12.03846  12.48077 13.711538    14.000000
[2,]  7.533333  10.40000         8.00000   7.40000  6.533333     7.133333
[3,]  5.500000  11.00000         3.50000  11.50000 11.500000    10.500000
```

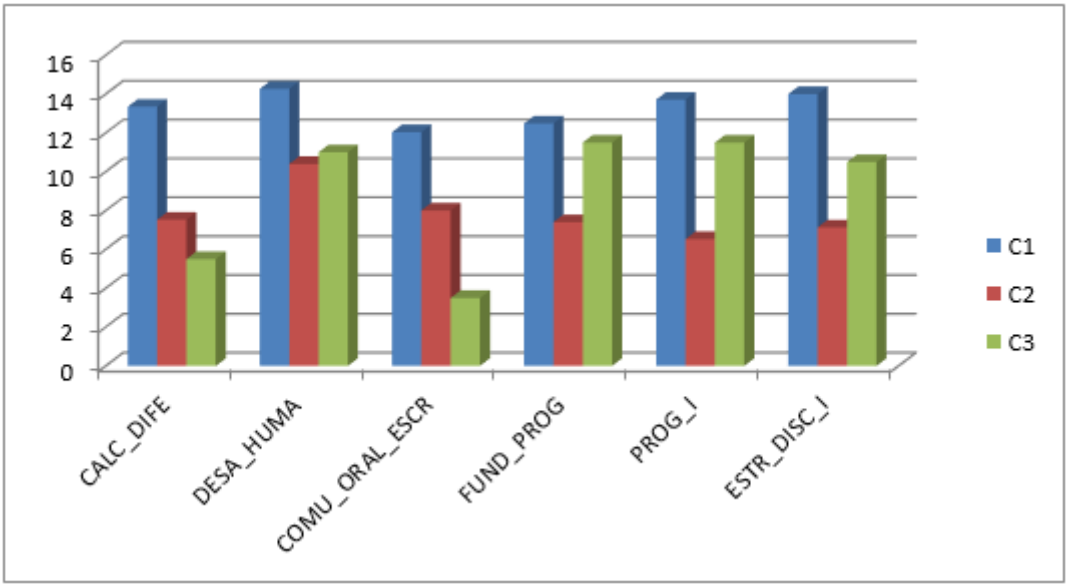


Figura 48: Promedio de notas en los clusters – Método average (Fuente: Elaboración Propia)

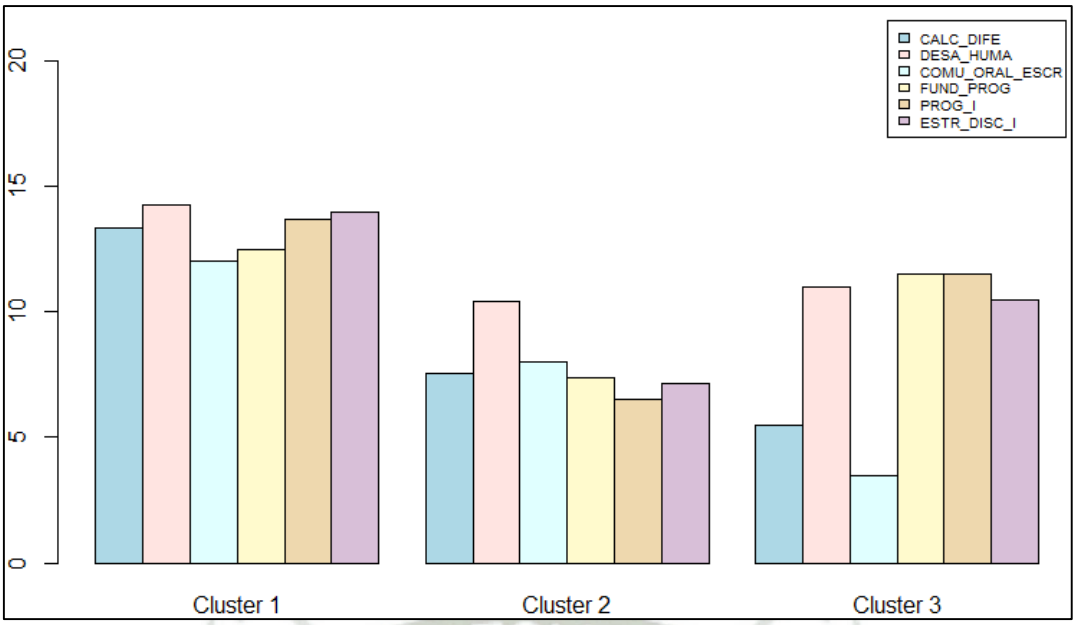


Figura 49: Comparación de clusters – Método average (Fuente: Elaboración Propia)

Utilizando el método mcquitty:

```
clust1 <- hclust(dist(notas), method= "mcquitty")
```

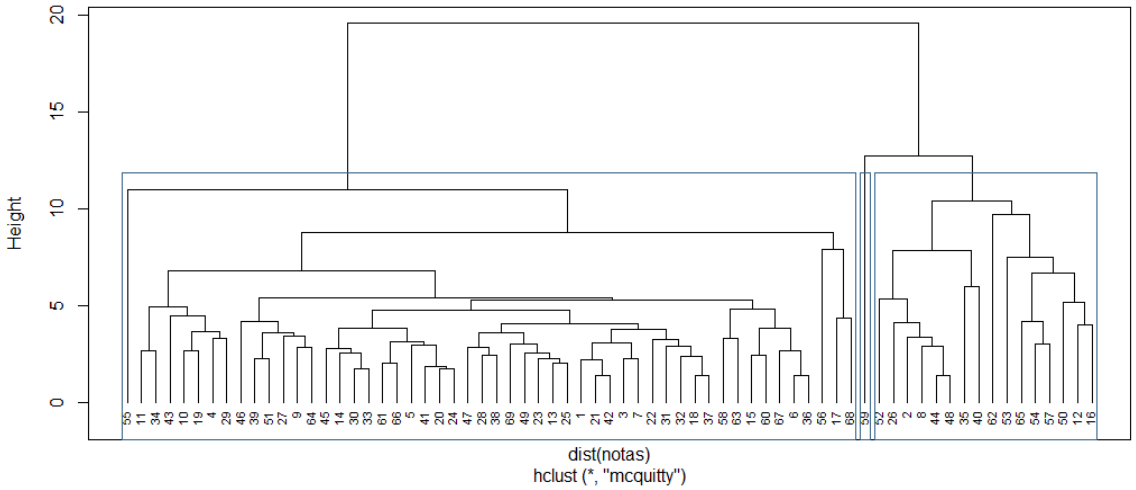


Figura 50: Dendrograma utilizando el método "mcquitty" (Fuente: Elaboración Propia)

Se segmento en tres grupos y se obtuvo los siguientes centros de gravedad

```
> centros<-centers.hclust(notas,clust1,nclust=3)
> centros
     CALC_DIFE DESA_HUMA COMU_ORAL_ESCR FUND_PROG   PROG_I ESTR_DISC_I
[1,]  13.34615  14.26923       12.03846  12.48077 13.71154       14.00
[2,]   7.68750  10.43750        7.68750   7.62500  6.87500        7.25
[3,]   1.00000  11.00000        4.00000  12.00000 11.00000       12.00
```

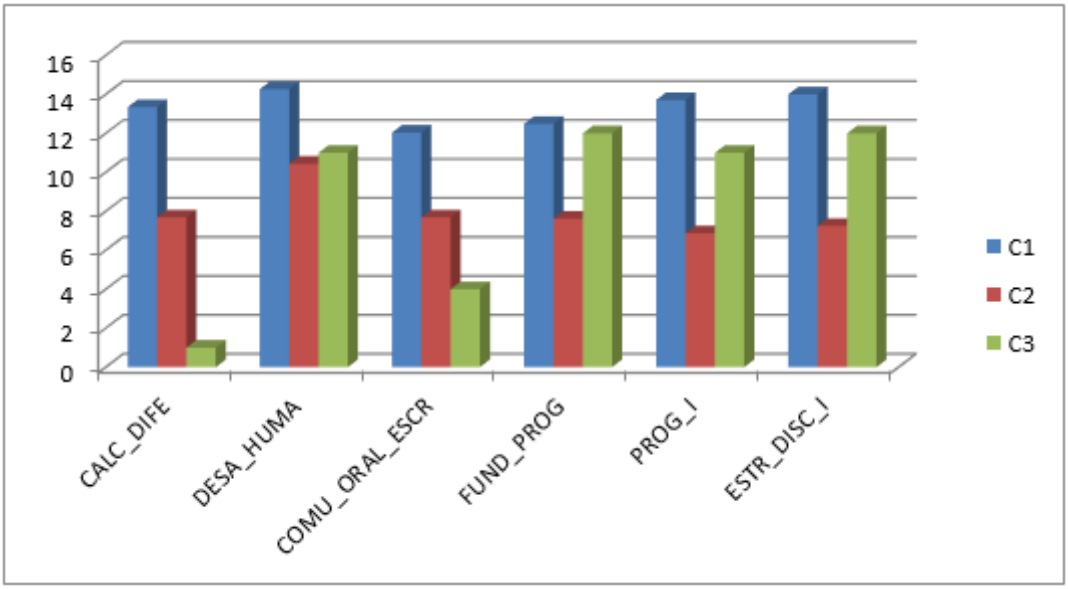


Figura 51: Promedio de notas en los clusters – Método mcquitty (Fuente: Elaboración Propia)

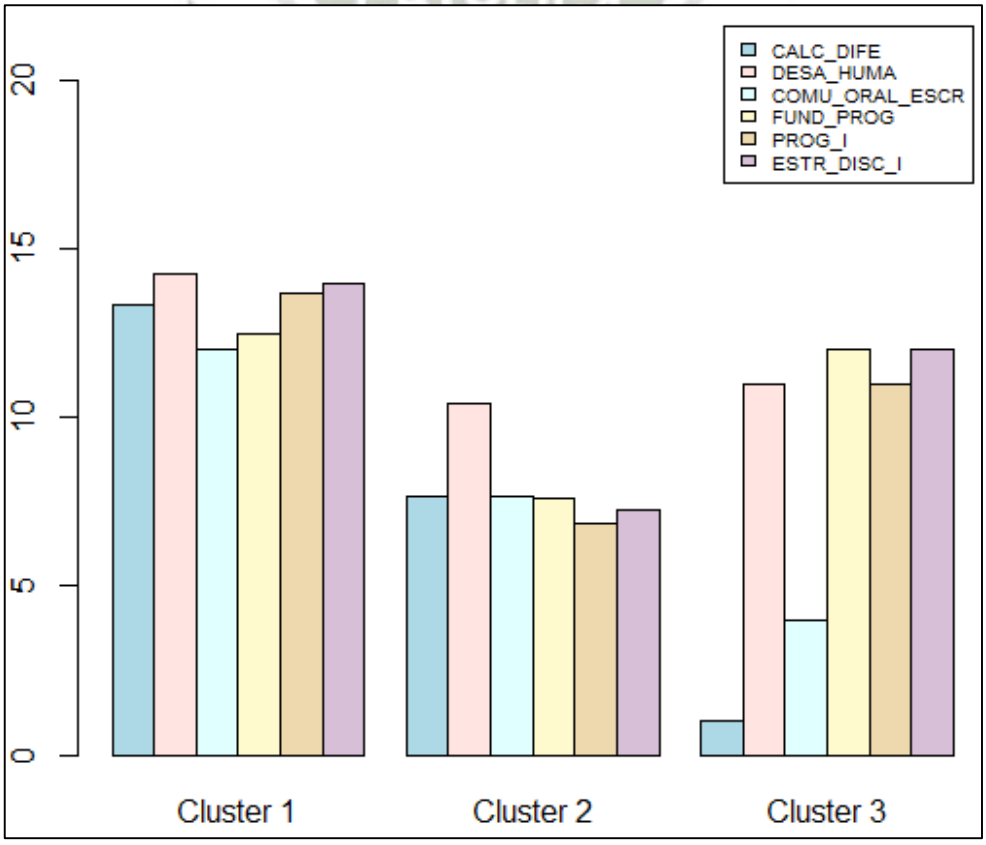


Figura 52: Comparación de clusters – Método mcquitty (Fuente: Elaboración Propia)

Utilizando el método median:

```
clust1 <- hclust(dist(notas), method= "median")
```

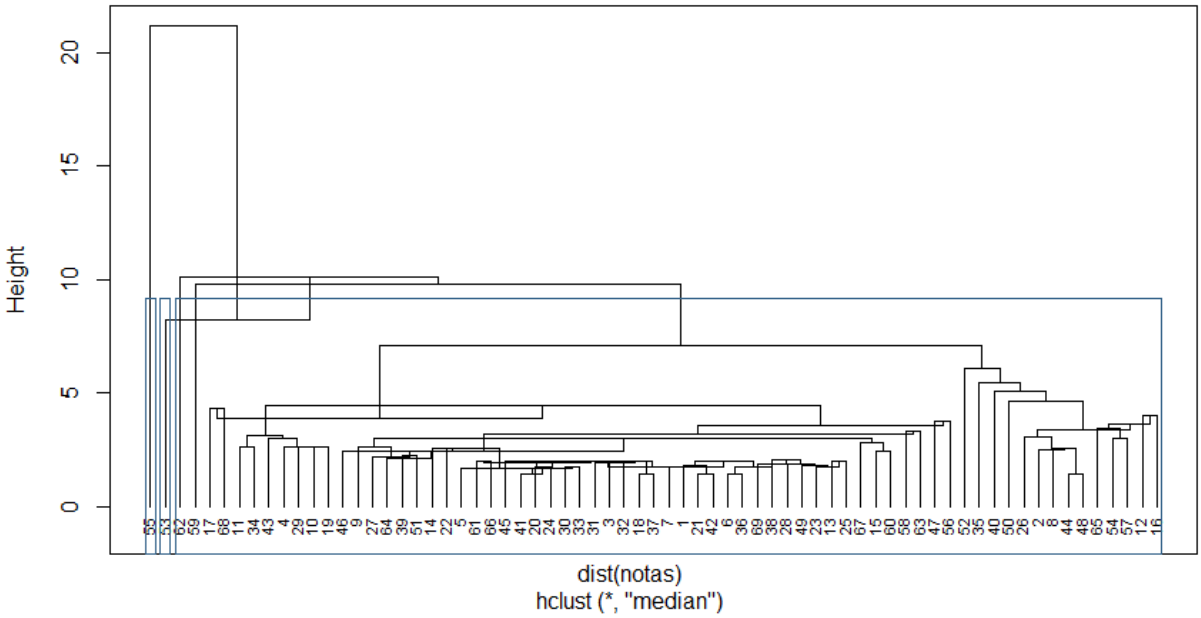


Figura 53: Dendrograma utilizando el método "median" (Fuente: Elaboración Propia)

Se segmento en tres grupos y se obtuvo los siguientes centros de gravedad

```
> centros<-centers.hclust(notas,clust1,nclust=3)
> centros
     CALC_DIFE DESA_HUMA COMU_ORAL_ESCR FUND_PROG   PROG_I ESTR_DISC_I
[1,]  11.89552  13.26866       10.89552  11.35821 12.13433    12.49254
[2,]   4.00000  11.00000        8.00000   5.00000  4.00000     2.00000
[3,]  17.00000  20.00000       15.00000  17.00000 17.00000    17.00000
```

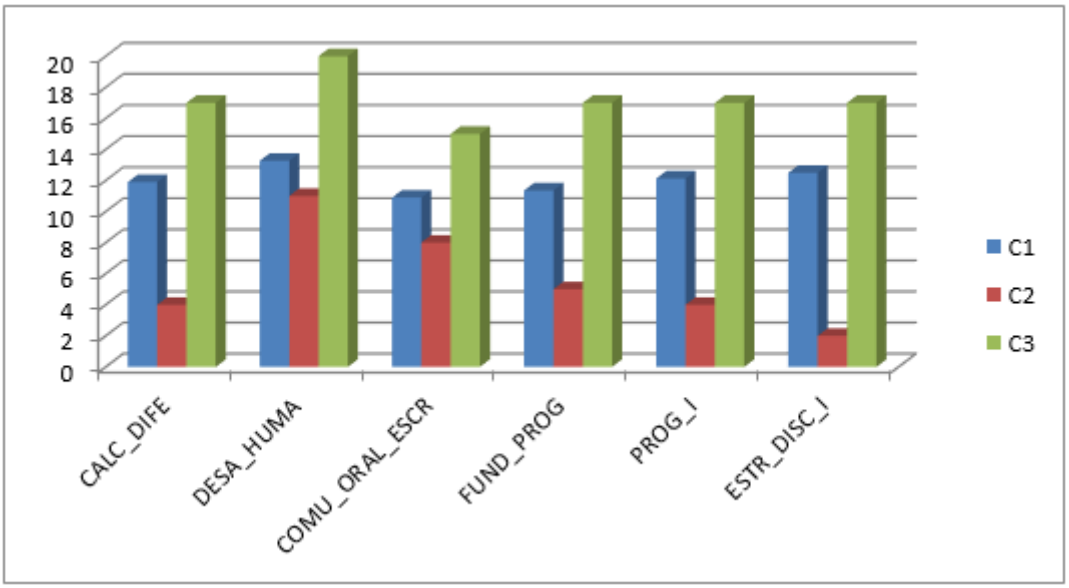


Figura 54: Promedio de notas en los clusters – Método median (Fuente: Elaboración Propia)

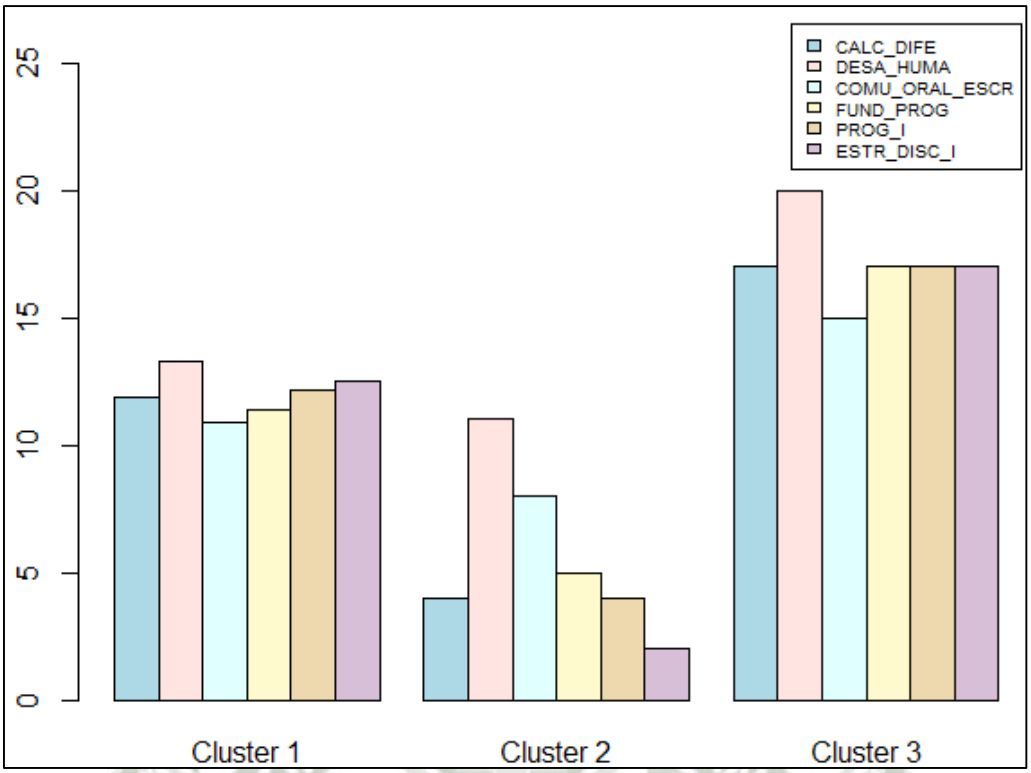


Figura 55: Comparación de clusters – método median (Fuente: Elaboración Propia)

Utilizando el método centroid:

```
clust1 <- hclust(dist(notas), method= "centroid")
```

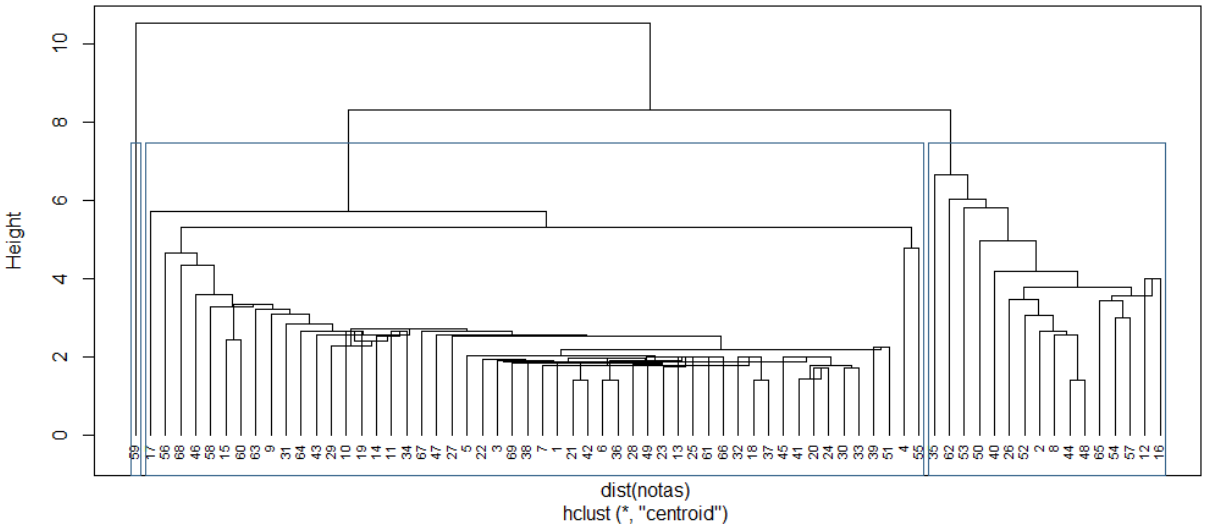


Figura 56: Dendrograma utilizando el método "centroid" (Fuente: Elaboración Propia)

Se segmento en tres grupos y se obtuvo los siguientes centros de gravedad

```
> centros<-centers.hclust(notas,clust1,nclust=3)
> centros
     CALC_DIFE DESA_HUMA COMU_ORAL_ESCR FUND_PROG   PROG_I ESTR_DISC_I
[1,]  13.34615  14.26923       12.03846  12.48077 13.71154       14.00
[2,]   7.68750  10.43750        7.68750   7.62500  6.87500        7.25
[3,]   1.00000  11.00000        4.00000  12.00000 11.00000       12.00
```

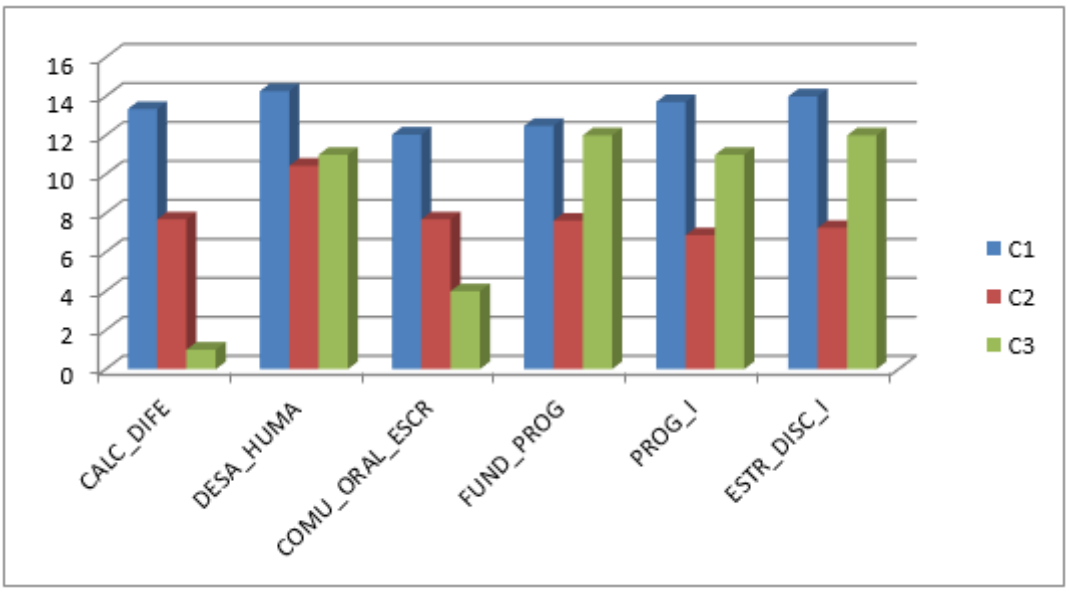


Figura 57: Promedio de notas en los clusters – Método centroid (Fuente: Elaboración Propia)

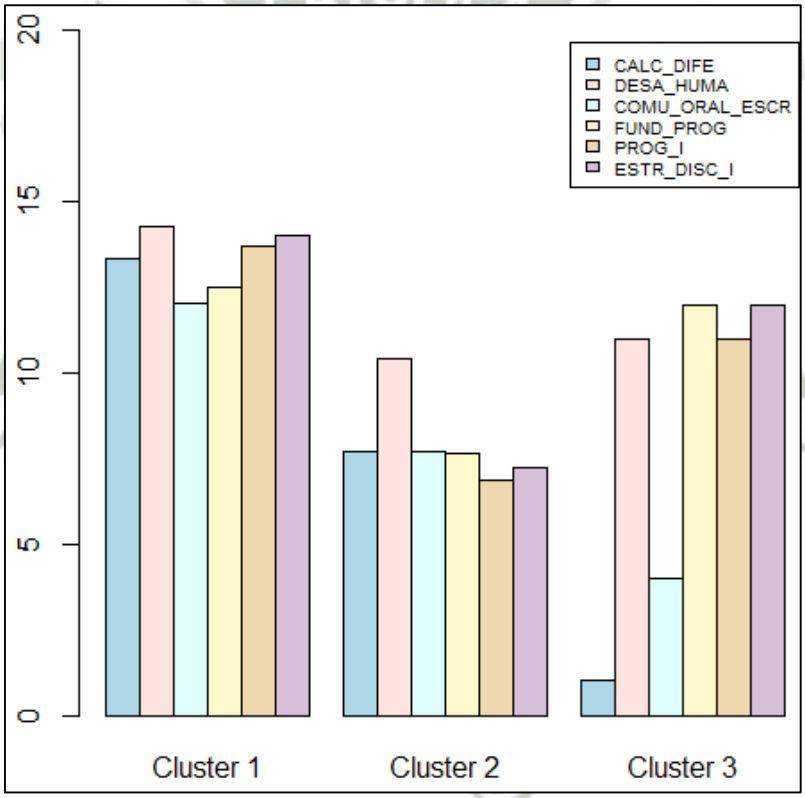


Figura 58: Comparación de clusters – Método centroid (Fuente: Elaboración Propia)

- **K-means**

Se segmento utilizando los diferentes algoritmos disponibles: Hartigan-Wong, Lloyd. Forgy y MacQueen, dando los mismos resultados para el estudio de caso, por lo que se ha elegido el algoritmo por defecto "Hartigan-Wong"

Se segmento en tres grupos:

```
grupos_Kmeans<-kmeans(notas, 3,iter.max=50, nstart=50)
grupos_Kmeans

K-means clustering with 3 clusters of sizes 17, 32, 20

Cluster means:
  CALC_DIFE DESA_HUMA COMU_ORAL_ESCR FUND_PROG    PROG_I ESTR_DISC_I
1  7.294118  10.47059       7.470588  7.882353  7.117647    7.529412
2 12.625000  12.96875      11.531250 11.968750 13.218750   13.250000
3 14.500000  16.35000      12.850000 13.300000 14.500000   15.200000

Clustering vector:
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
 2  1  2  3  2  2  2  1  3  3  3  1  2  3  2  1  3  2  3  3  2  2  2  3  2
26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50
 1  2  2  3  3  2  2  3  3  1  2  2  2  3  1  3  2  3  1  3  2  2  1  2  1
51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69
 3  1  1  1  3  2  1  2  1  2  2  1  2  3  1  2  2  2  2

Within cluster sum of squares by cluster:
[1] 571.7647 340.8750 276.5000
 (between_SS / total_SS =  68.5 %)
```

Se muestra el tamaño de los clusters:

```
> grupos_Kmeans$size
[1] 17 32 20
```

Se muestran los centros de gravedad:

```
> grupos_Kmeans$centers
  CALC_DIFE DESA_HUMA COMU_ORAL_ESCR FUND_PROG    PROG_I ESTR_DISC_I
1  7.294118  10.47059       7.470588  7.882353  7.117647    7.529412
2 12.625000  12.96875      11.531250 11.968750 13.218750   13.250000
3 14.500000  16.35000      12.850000 13.300000 14.500000   15.200000
```

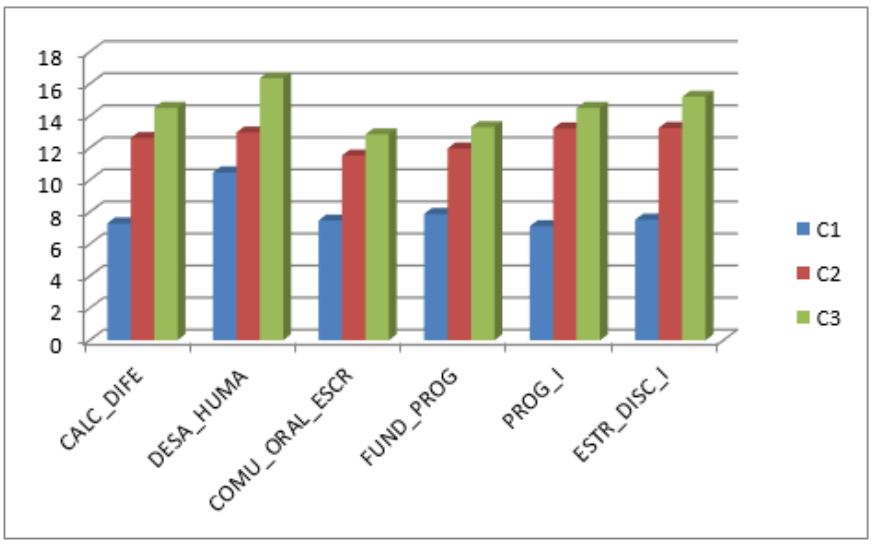


Figura 59: Promedio de notas en los clusters – Kmeans (Fuente: Elaboración Propia)

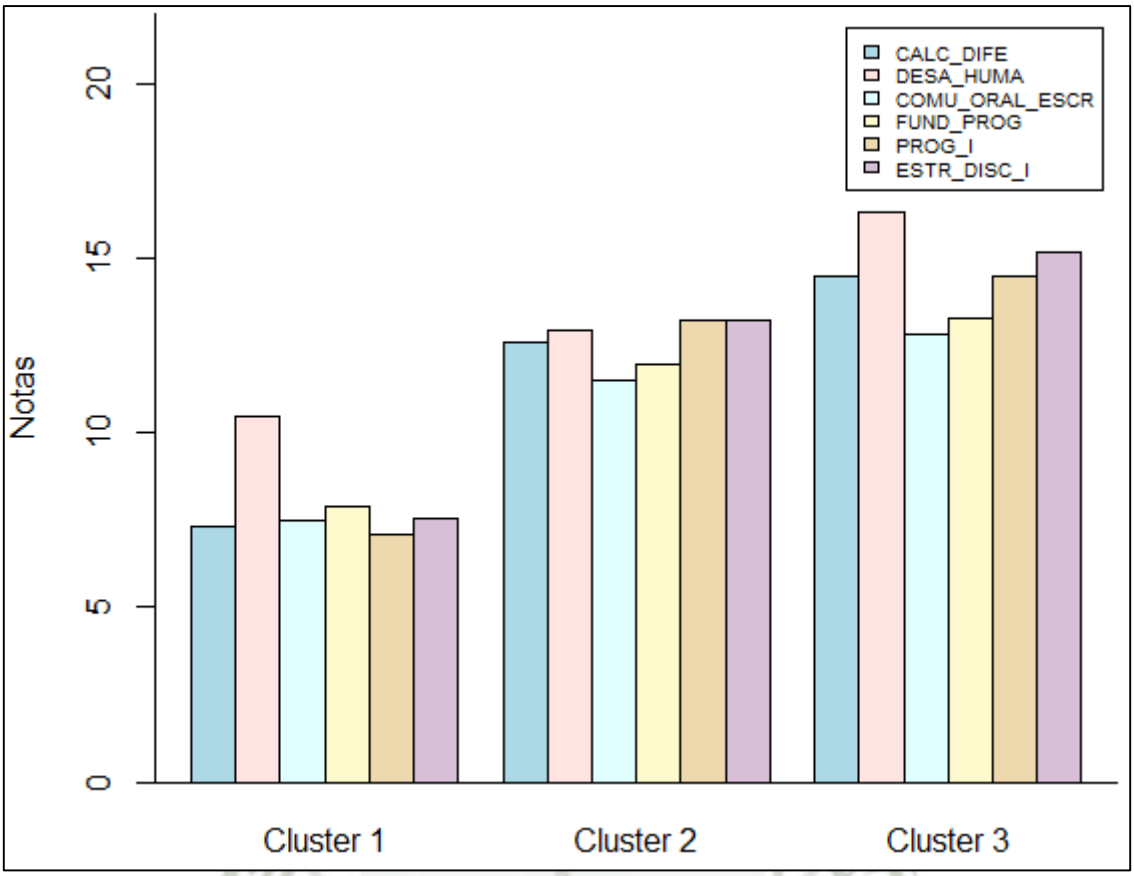


Figura 60: Comparación de clusters – Kmeans (Fuente: Elaboración Propia)

- **PAM**

Se segmento en tres grupos:

```
grupos_PAM <- pam(notas,3)
grupos_PAM
```

```
Medoids:
   ID CALC_DIFE DESA_HUMA COMU_ORAL_ESCR FUND_PROG PROG_I ESTR_DISC_I
25 25        12        14             11        11     13           14
8   8         9        11              9         8      7            7
45 45        15        15             13        14     14           14
Clustering vector:
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26
 1  2  3  3  3  1  1  2  1  3  3  2  1  3  1  2  3  3  3  3  1  1  1  3  1  2
27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52
 1  1  3  3  3  1  3  3  2  1  3  1  3  2  3  1  3  2  3  1  1  2  1  2  1  2
53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69
 2  2  3  1  2  1  2  1  1  2  1  1  2  1  1  3  1
Objective function:
   build     swap
4.061587 4.061587
```

Se muestra los medoides (medianas) que representan los clusters formados

```
> grupos_PAM$medoids
   CALC_DIFE DESA_HUMA COMU_ORAL_ESCR FUND_PROG PROG_I ESTR_DISC_I
25        12        14             11        11     13           14
8          9        11              9         8      7            7
45        15        15             13        14     14           14
```

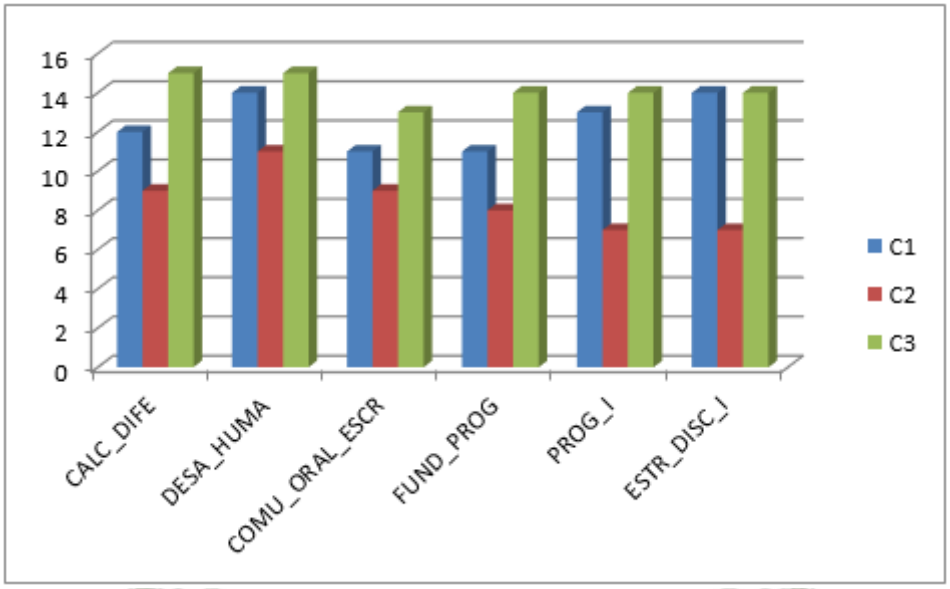


Figura 61: Mediana de notas en los clusters – PAM (Fuente: Elaboración Propia)

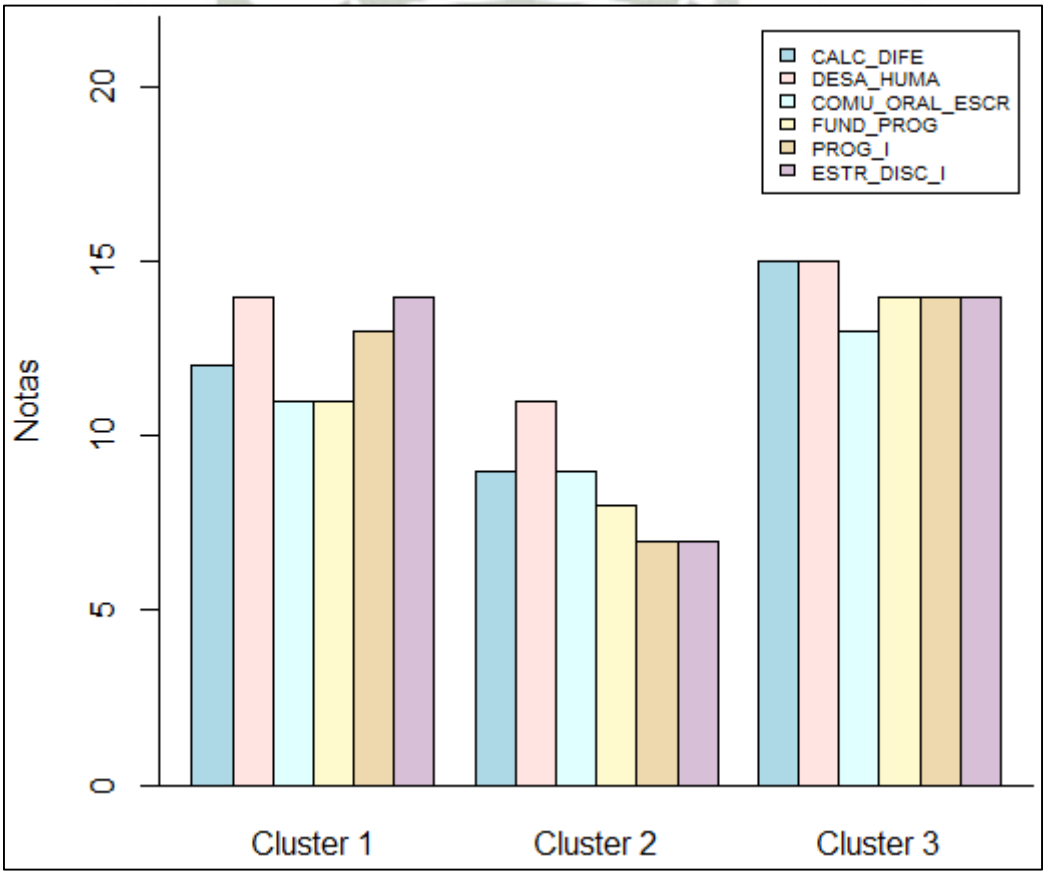


Figura 62: Comparación de clusters – PAM (Fuente: Elaboración Propia)

CAPÍTULO 4: EVALUACIÓN Y RESULTADOS

4.1. Evaluación de las técnicas no supervisadas de minería de datos

La evaluación de las técnicas no supervisadas de minería de datos aplicadas para la segmentación de alumnos se realizó teniendo en cuenta:

- Las distancias intra-cluster e inter-cluster
- El coeficiente de silueta.

4.1.1. Distancias intra-cluster e inter-cluster

Se debe seleccionar una técnica que minimice la distancia intra-cluster (cohesión) y maximice la distancia inter-cluster (separación).

La cohesión (WSS) se mide por la suma de los cuadrados intra-cluster:

$$WSS = \sum_{i} \sum_{x \in C_i} (x - m_i)^2$$

Donde:

- i es el identificador del cluster
- m_i corresponde al promedio del cluster
- x es el punto de datos que pertenece al grupo Ci

La separación (BSS) se mide por la suma de los cuadrados inter-cluster:

$$BSS = \sum_{i} |C_i| (m - m_i)^2$$

Donde:

- $|C_i|$ es el tamaño del cluster i
- m_i corresponde al promedio del cluster
- m corresponde al promedio total

4.1.1.1. Clustering jerárquico aglomerativo

En la Tabla 7 se muestra los resultados obtenidos de las distancias intra-cluster por cada método del clustering jerárquico aglomerativo.

Tabla 7: Distancias Intra-Cluster - Clustering Jerárquico (Fuente: Elaboración Propia)

Método	Cluster 1	Cluster 2	Cluster 3	Total WSS
Ward	377.88	571.76	261.80	1211.44
Single	3377.82	0	0	3377.82
Complete	374.26	571.76	275.94	1221.96
Average	911.58	388.40	46.50	1346.48
Mcquitty	911.58	461.31	0	1372.89
Median	3313.64	0	0	3313.64
Centroid	911.58	461.31	0	1372.89

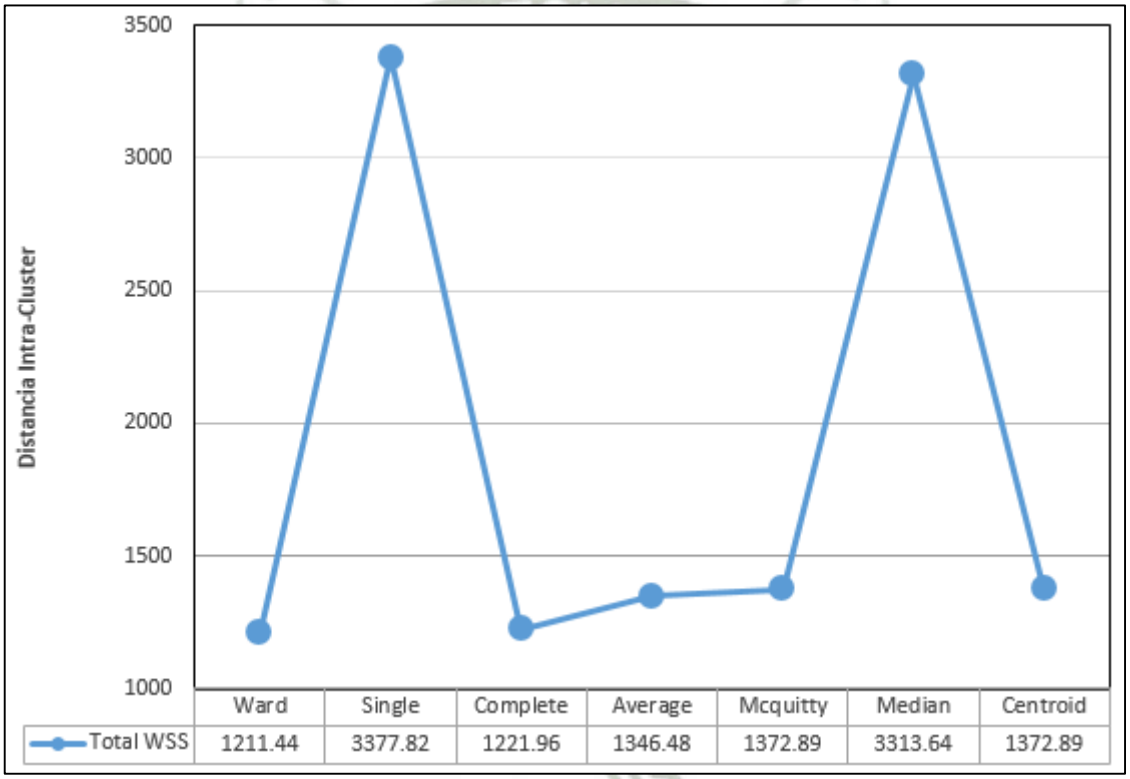


Figura 63: Distancias Intra-Cluster – Clustering Jerárquico (Fuente: Elaboración Propia)

En la Figura 63 se puede observar que la menor distancia intra-cluster, que representa mayor cohesión de la agrupación, se obtiene con el método de agregación de Ward.

En la Tabla 8 se muestra los resultados obtenidos de las distancias inter-cluster por cada método del clustering jerárquico aglomerativo.

Tabla 8: Distancias Inter-Cluster - Clustering Jerárquico (Fuente: Elaboración Propia)

Método	Cluster 1	Cluster 2	Cluster 3	Total BSS
Ward	4.66	101.33	34.30	2557.69
Single	0.12	172.84	210.22	391.31
Complete	3.48	101.33	39.24	2547.16
Average	10.83	109.99	104.78	2422.65
Mcquitty	10.83	103.76	172.84	2396.24
Median	0.02	289.61	164.81	455.49
Centroid	10.83	103.76	172.84	2396.24

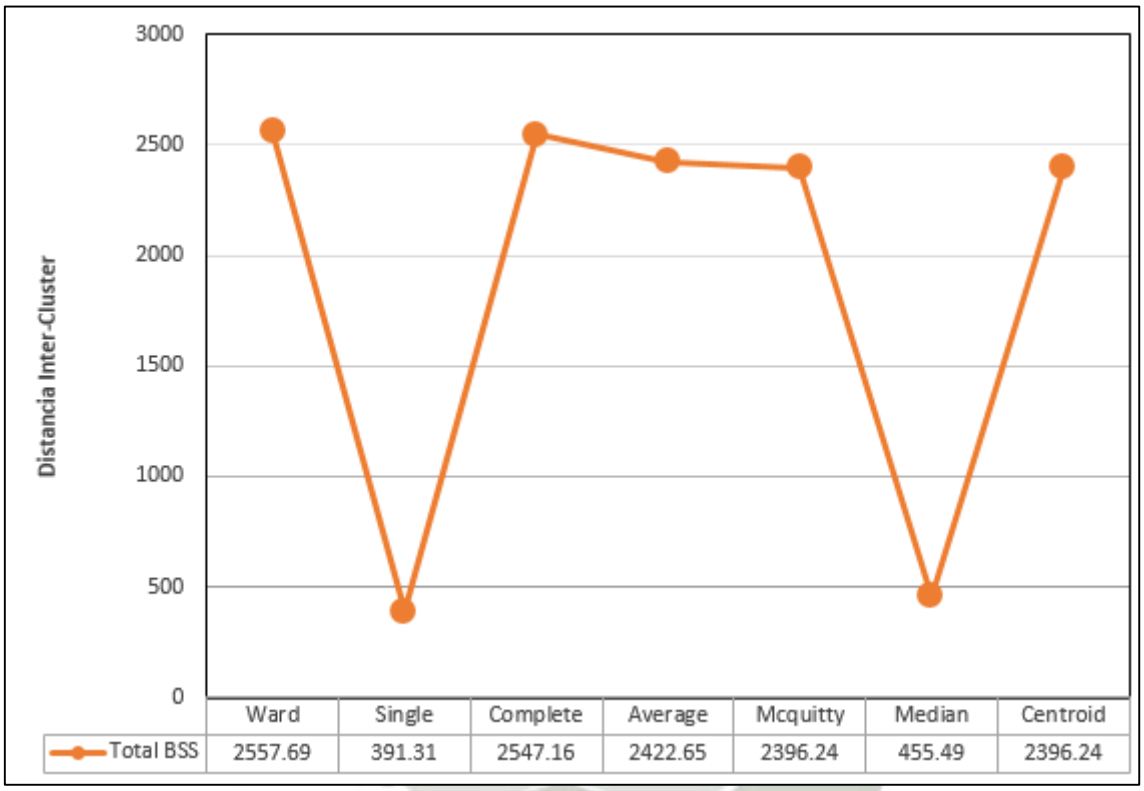


Figura 64: Distancias Inter-Cluster - Clustering Jerárquico (Fuente: Elaboración Propia)

En la Figura 64 se puede observar que la mayor distancia Inter-Cluster, que representa mayor separación entre los clusters formados, se obtiene con el método de agregación de Ward.

Dentro de los métodos de clustering jerárquico aglomerativo se elige la agregación de Ward, ya que los grupos formados según las distancias halladas tienen mayor similitud dentro del grupo y mayor diferencia entre los grupos.

4.1.1.2. K-means

En la Tabla 9 y Tabla 10 se muestran las distancias Intra-Cluster e Inter-Cluster respectivamente para el algoritmo de agrupación K-means.

Tabla 9: Distancias Intra-Cluster - Kmeans (Fuente: Elaboración Propia)

Cluster 1	Cluster 2	Cluster 3	Total WSS
571.76	340.88	276.50	1189.14

Tabla 10: Distancias Inter-Cluster - Kmeans (Fuente: Elaboración Propia)

Cluster 1	Cluster 2	Cluster 3	Total BSS
101.33	3.49	37.29	2579.99

4.1.1.3. PAM

En la Tabla 11 y Tabla 12 se muestran las distancias Intra-Cluster e Inter-Cluster respectivamente para el algoritmo PAM (Partitioning Around Medoids).

Tabla 11: Distancias Intra-Cluster -PAM (Fuente: Elaboración Propia)

Cluster 1	Cluster 2	Cluster 3	Total WSS
306.14	571.76	351.13	1229.03

Tabla 12: Distancias Inter-Cluster -PAM (Fuente: Elaboración Propia)

Cluster 1	Cluster 2	Cluster 3	Total BSS
2.52	101.33	32.36	2540.10

4.1.1.4. Comparación de las distancias intra-cluster e inter-cluster

A continuación se muestra el resumen de las distancias Intra-Cluster e Inter-Cluster del método jerárquico aglomerativo Ward, algoritmo Kmeans y PAM.

Tabla 13: Distancias Intra-Cluster e Inter-Cluster (Fuente: Elaboración Propia)

Algoritmos	Total WSS	Total BSS
Ward	1211.44	2557.69
Kmeans	1189.14	2579.99
PAM	1229.03	2540.10

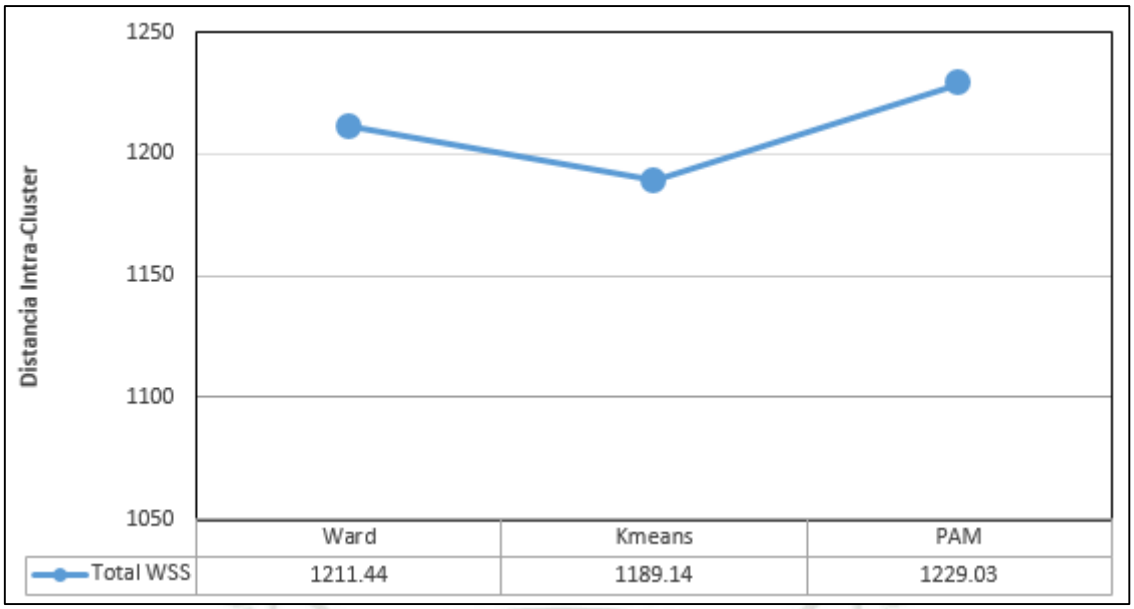


Figura 65: Comparación Distancia Intra-Cluster (Fuente: Elaboración Propia)

En la Figura 65 se puede observar que la menor distancia Intra-Cluster, que representa mayor cohesión y similitud de la agrupación, se obtiene con el algoritmo kmeans.

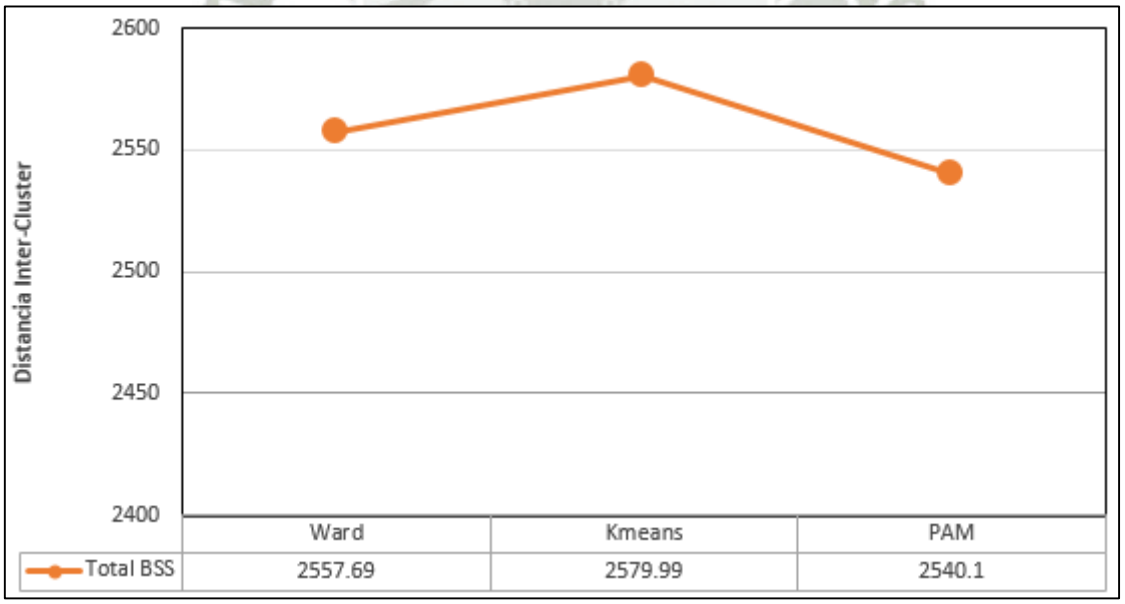


Figura 66: Comparación Distancia Inter-Cluster (Fuente: Elaboración Propia)

En la Figura 66 se puede observar que la mayor distancia Inter-Cluster, que representa mayor separación o diferencia entre los clusters formados, se obtiene con el algoritmo kmeans.

Para la segmentación de alumnos se elige el algoritmo kmeans por ser la técnica con menor distancia Intra-Cluster y mayor distancia Inter-Cluster, que representa mayor homogeneidad dentro del grupo y mayor diferencia entre los grupos.

4.1.2. Coeficiente silueta

Se utiliza el coeficiente silueta (silhouette) para medir la calidad de los clusters. Típicamente el coeficiente silueta está entre 0 y 1. Mientras más cercano a 1 es mejor. Si es negativo representa una mala agrupación.

En las Figuras 67, 68 y 69 se puede observar las siluetas obtenidas de dos, tres y cuatro clusters para el método jerárquico aglomerativo Ward, algoritmo Kmeans y PAM.

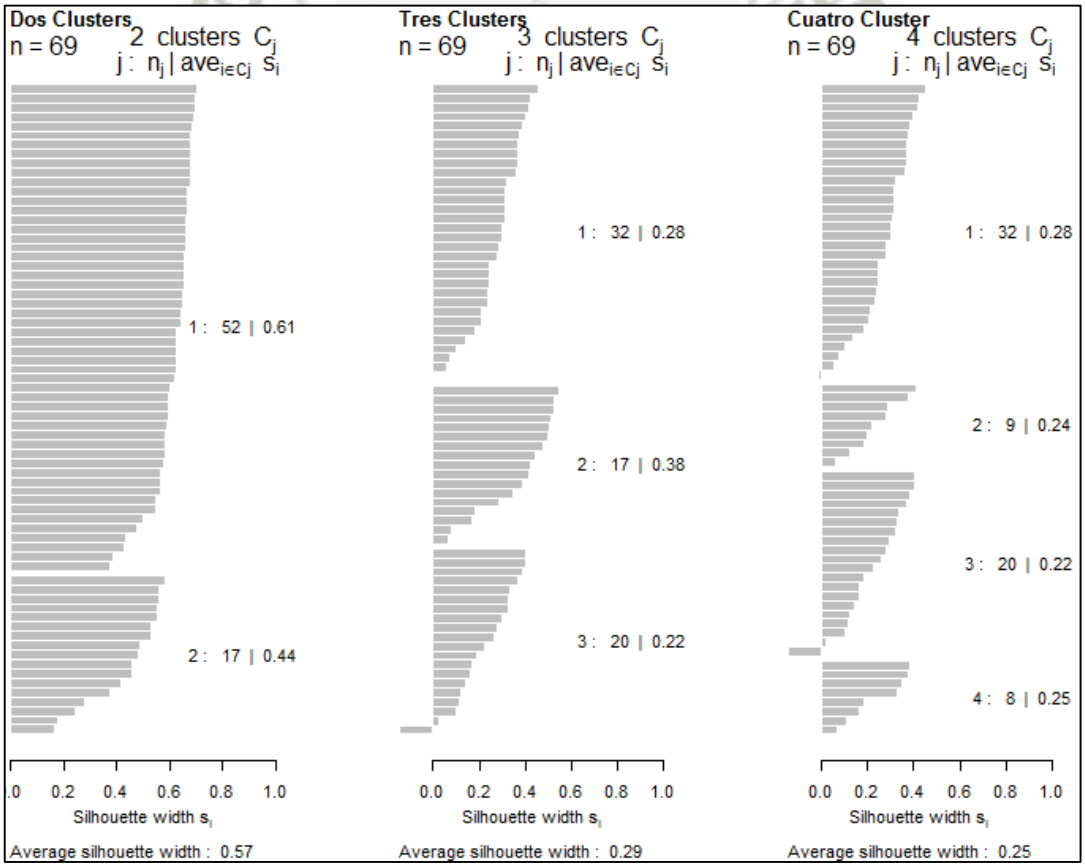


Figura 67: Silueta – Clustering Jerárquico (Fuente: Elaboración Propia)

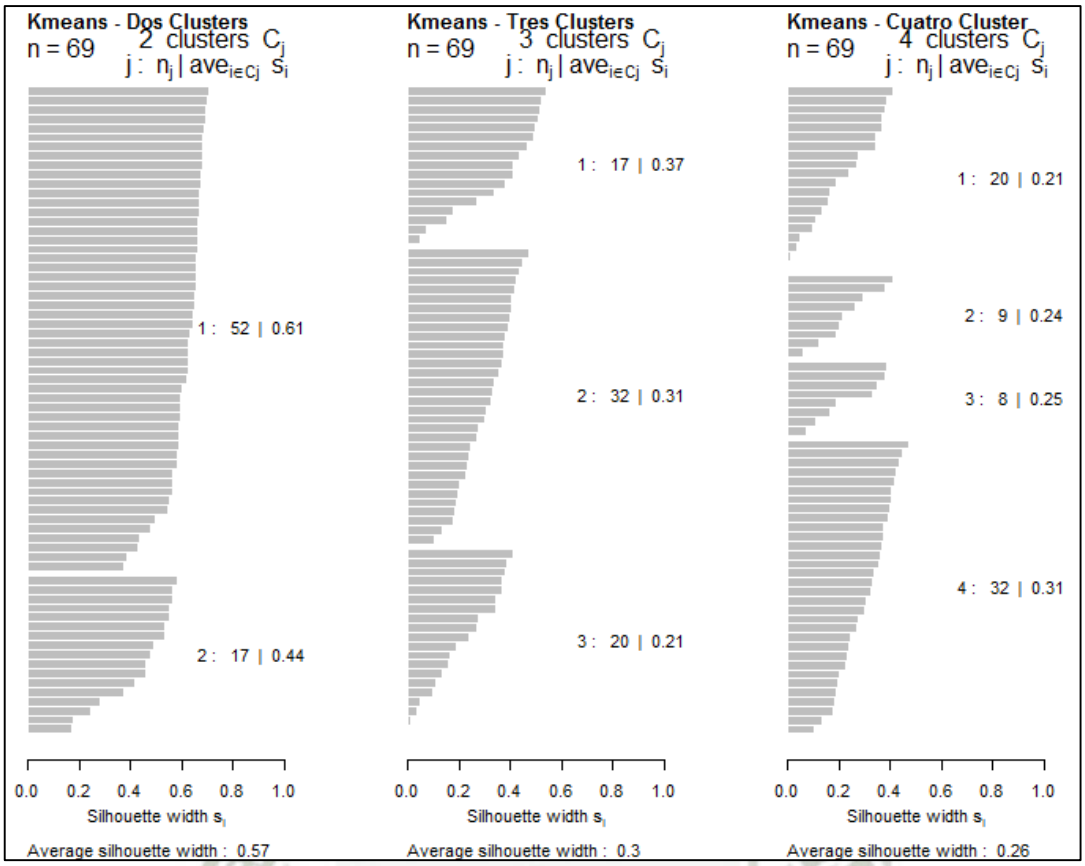


Figura 68: Silueta – Kmeans (Fuente: Elaboración Propia)

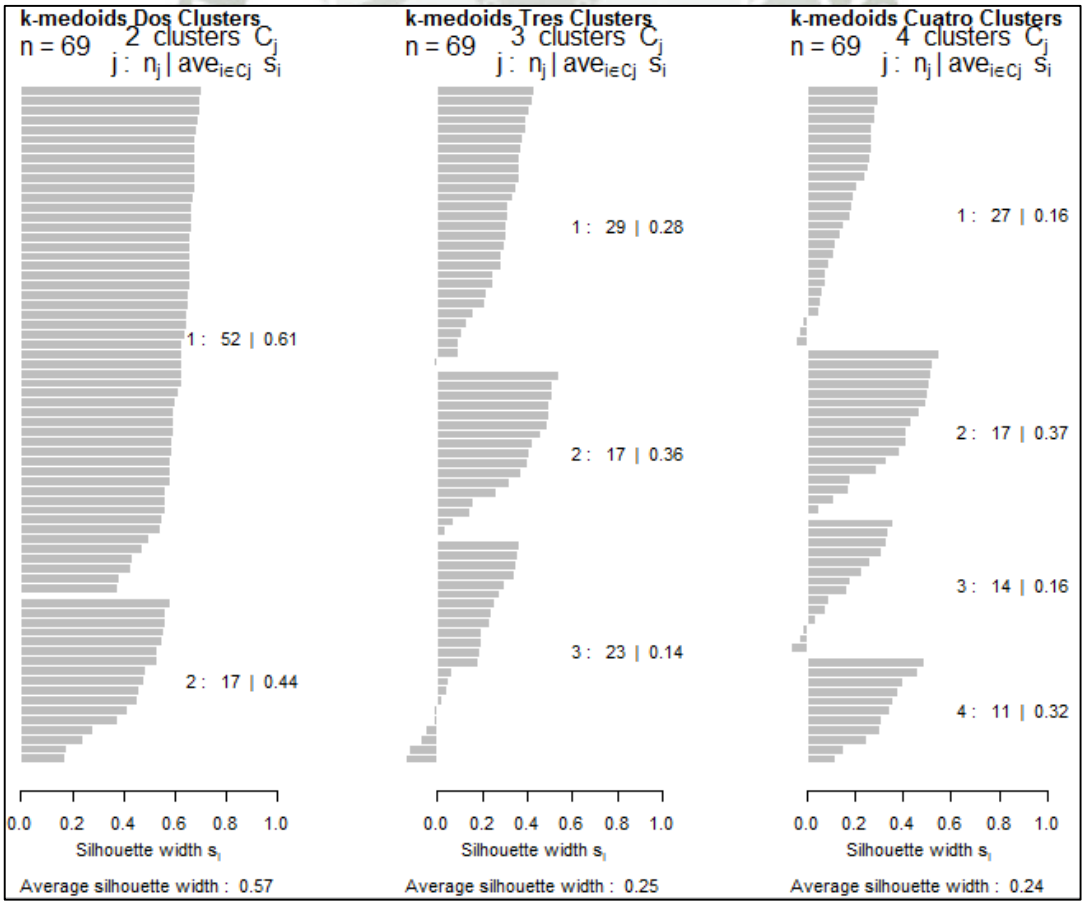


Figura 69: Silueta – PAM (Fuente: Elaboración Propia)

Los coeficientes de silueta obtenidos para dos, tres y cuatro clusters por el método jerárquico aglomerativo Ward, algoritmo Kmeans y PAM se muestran en la siguiente tabla:

Tabla 14: Coeficientes de Silueta (Fuente: Elaboración Propia)

Algoritmos	Dos Clusters	Tres Clusters	Cuatro Clusters
Ward	0.57	0.29	0.25
Kmeans	0.57	0.30	0.26
PAM	0.57	0.25	0.24

En la Tabla 14 se puede observar que con los tres algoritmos se obtiene un coeficiente de silueta de 0.57 para dos clusters, el cual representa una estructura razonable, clasificando bien los alumnos con promedio final aprobados y desaprobados.

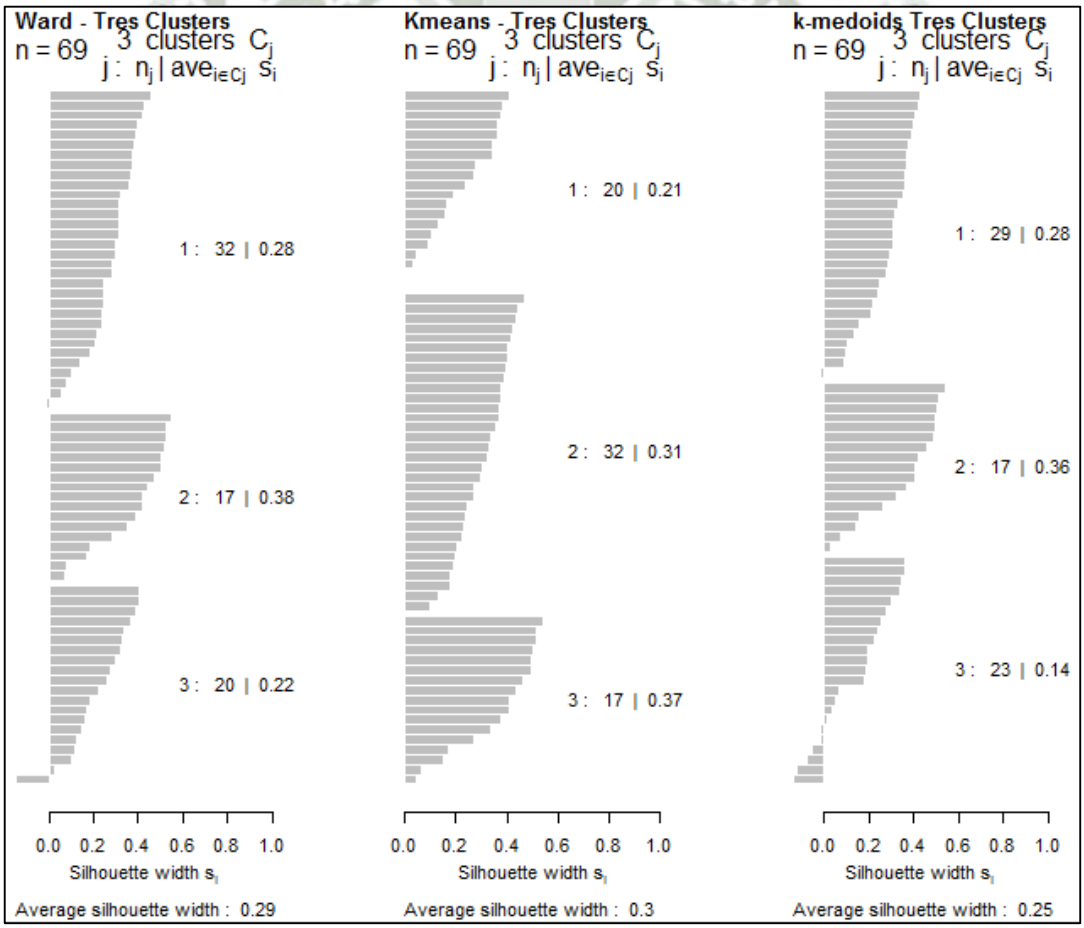


Figura 70: Comparación de Siluetas para tres clusters (Fuente: Elaboración Propia)

En la Figura 70 se puede observar que para agrupaciones de tres clusters presenta una mejor estructura de grupos el algoritmo de kmeans con 0.30 a

comparación de los algoritmos Ward y PAM que tienen 0.29 y 0.25 como coeficiente de siluetas respectivamente.

Se elige el algoritmo kmeans para la segmentación académica en tres grupos para el reforzamiento de los alumnos en los niveles básico, intermedio y avanzado, ya que con esta técnica de clustering se obtiene grupos de mejor calidad y con menor distancia Intra-Cluster y mayor distancia Inter-Cluster.

4.2. Pruebas y Resultados

Luego de elegir la técnica no supervisada con la que se obtiene grupos de mejor calidad para el problema, se realizó la segmentación de tres grupos con el algoritmo kmeans, como se muestra en la Figura 71, para representar a los alumnos de rendimiento BAJO, MEDIO y ALTO.

```
> grupos_Kmeans<-kmeans(notas, 3,iter.max=50, nstart=50)
> grupos_Kmeans
K-means clustering with 3 clusters of sizes 17, 32, 20

Cluster means:
  CALC_DIFE DESA_HUMA COMU_ORAL_ESCR FUND_PROG    PROG_I ESTR_DISC_I
1  7.294118  10.47059       7.470588  7.882353  7.117647    7.529412
2 12.625000  12.96875      11.531250 11.968750 13.218750   13.250000
3 14.500000  16.35000      12.850000 13.300000 14.500000   15.200000

Clustering vector:
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29
 2  1  2  3  2  2  2  1  3  3  3  1  2  3  2  1  3  2  3  3  2  2  2  3  2  1  2  2  3
30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58
 3  2  2  3  3  1  2  2  2  3  1  3  2  3  1  3  2  2  1  2  1  3  1  1  1  3  2  1  2
59 60 61 62 63 64 65 66 67 68 69
 1  2  2  1  2  3  1  2  2  2  2

Within cluster sum of squares by cluster:
[1] 571.7647 340.8750 276.5000
 (between_SS / total_SS =  68.5 %)
```

Figura 71: Resultados del Clustering Kmeans (Fuente: Elaboración Propia)

En la Figura 71 se puede observar los tamaños de cada cluster, los promedios de los clusters, el vector de clustering donde se especifica a que cluster corresponde cada alumno según su código así como la suma de los cuadrados Intra-Cluster.

En la Figura 72 se observa gráficamente que el 25% de la cantidad de alumnos pertenecen al Cluster 1 que son los alumnos con rendimiento bajo, el 46% corresponde al Cluster 2 que son los alumnos de rendimiento medio y el 29% representa al Cluster 3 que son los alumnos con rendimiento alto.

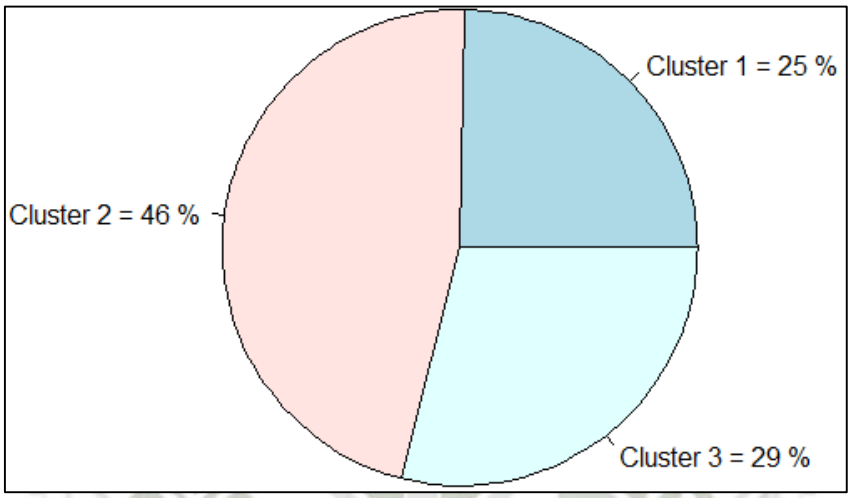


Figura 72: Porcentajes de cantidad de alumnos en cada cluster (Fuente: Elaboración Propia)

En la Tabla 15 se muestra los resultados de los promedios de los clusters obtenidos con la segmentación.

Tabla 15: Promedios de las notas de los clusters (Fuente: Elaboración Propia)

	CALC_DIFE	DESA_HUMA	COMU_ORAL	FUND_PROG	PROG_I	ESTR_DISC_I
C1	7.294118	10.47059	7.470588	7.882353	7.117647	7.529412
C2	12.625	12.96875	11.53125	11.96875	13.21875	13.25
C3	14.5	16.35	12.85	13.3	14.5	15.2

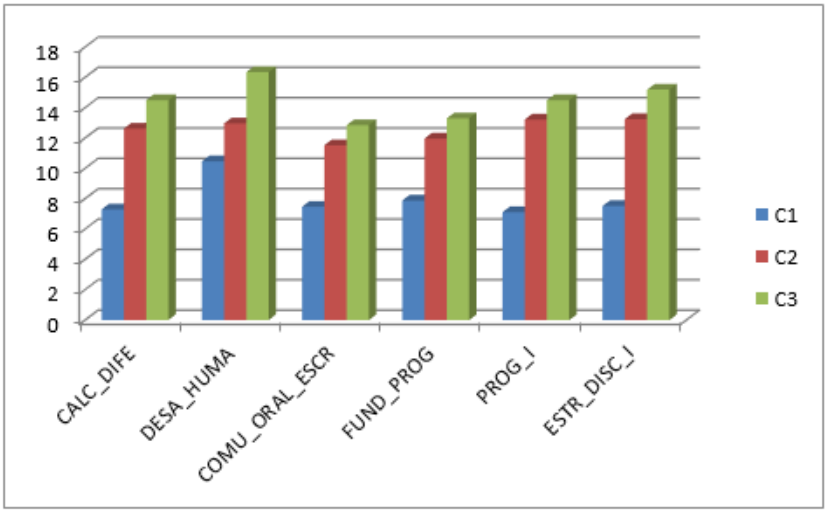


Figura 73: Promedios de las notas de los clusters (Fuente: Elaboración Propia)

C1 es el CLUSTER 1 que tiene los promedios de notas más bajos (Rendimiento Bajo)

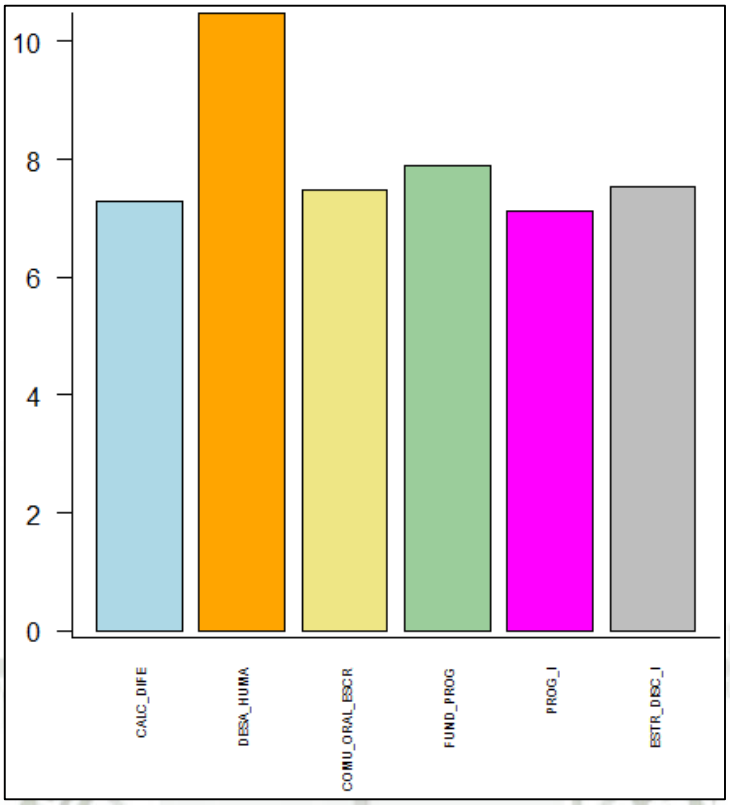


Figura 74: Cluster 1 – Rendimiento Bajo (Fuente: Elaboración Propia)

C2 es el CLUSTER 2 que tiene los promedios de notas medios (Rendimiento Medio)

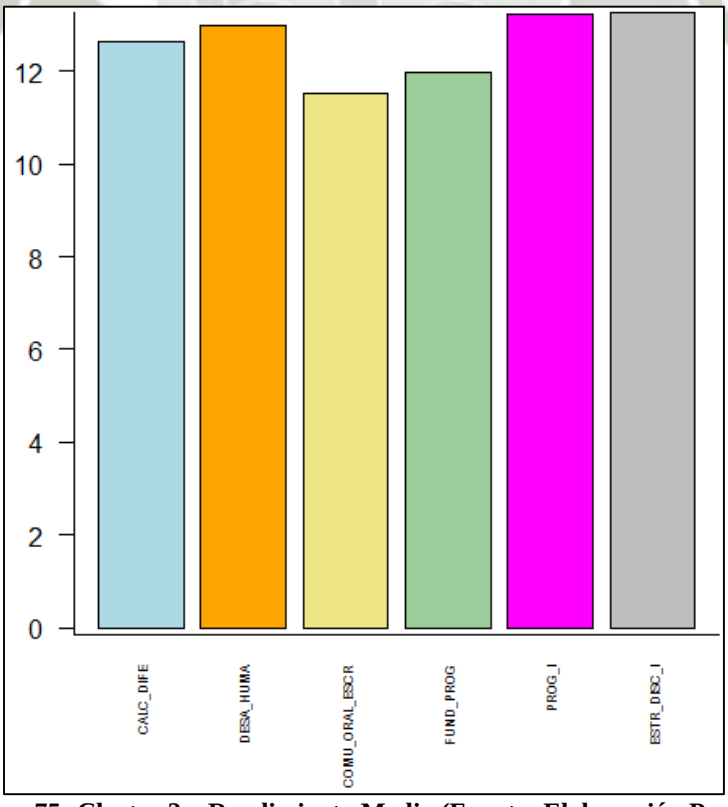


Figura 75: Cluster 2 – Rendimiento Medio (Fuente: Elaboración Propia)

C3 es el CLUSTER 3 que tiene los promedios de notas mayores (Rendimiento Alto)

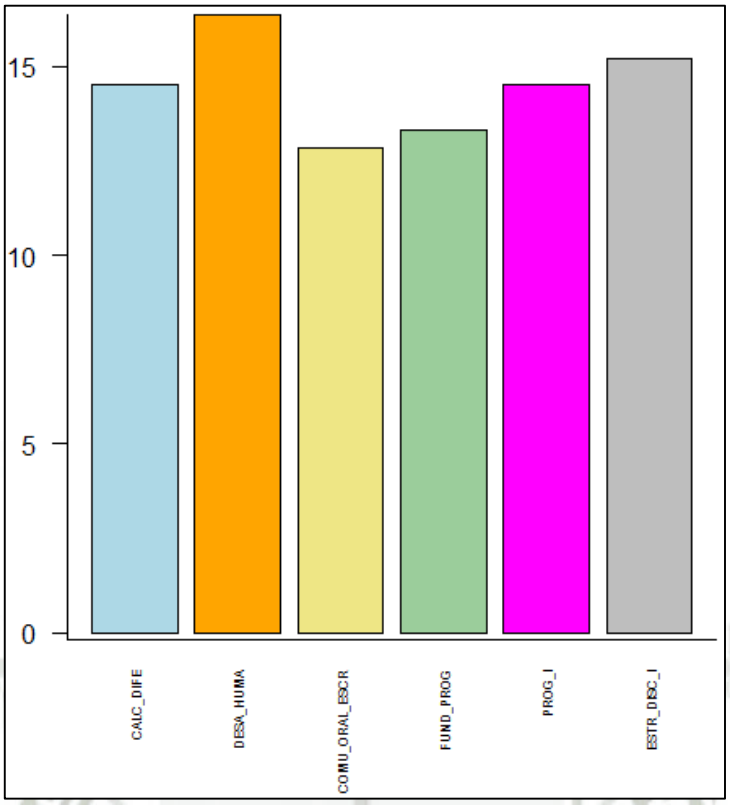


Figura 76: Cluster 3 – Rendimiento Alto (Fuente: Elaboración Propia)

A continuación se muestra una comparación de los clusters:

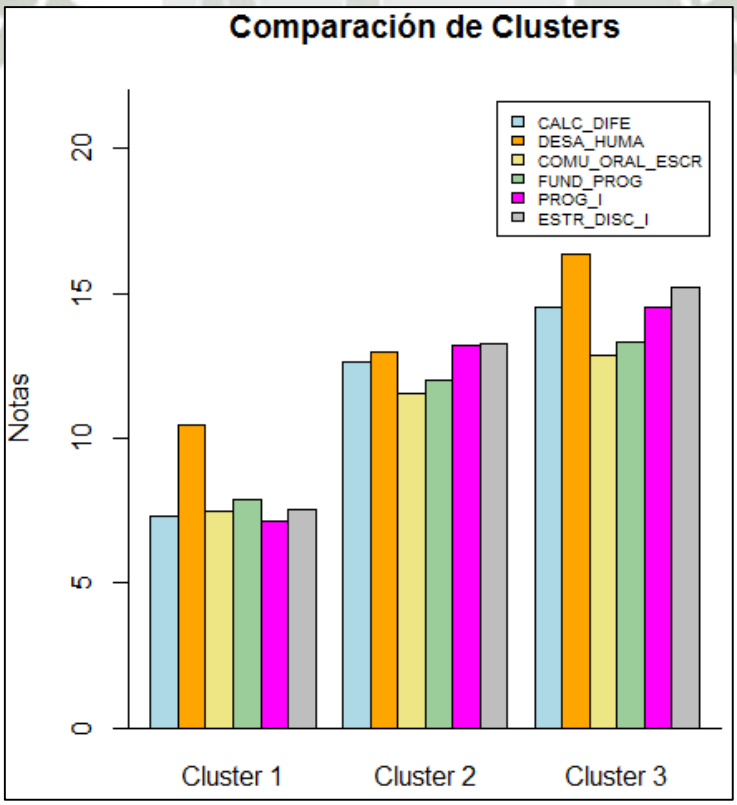


Figura 77: Comparación de Clusters (Fuente: Elaboración Propia)

En la Figura 78 se muestra un análisis de componentes principales, donde se aprecian los tres cluster, el cluster 1 (RENDIMIENTO BAJO) está más alejado de los cursos, así como el cluster 3 (RENDIMIENTO ALTO) está más cerca.

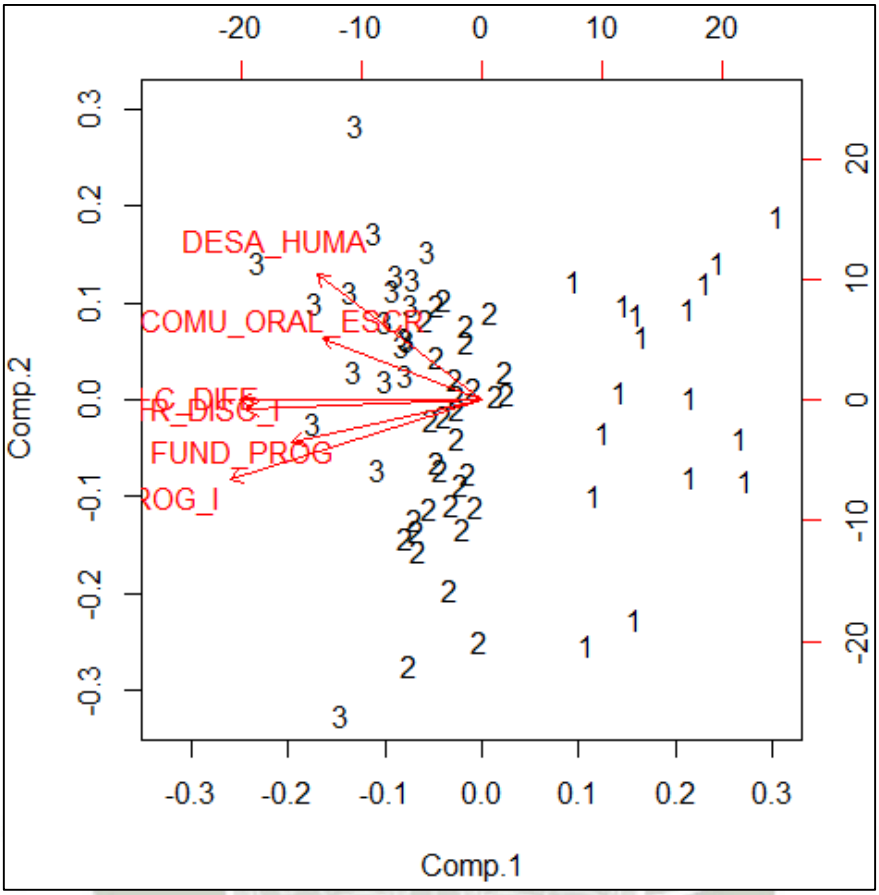


Figura 78: Componentes Principales (Fuente: Elaboración Propia)

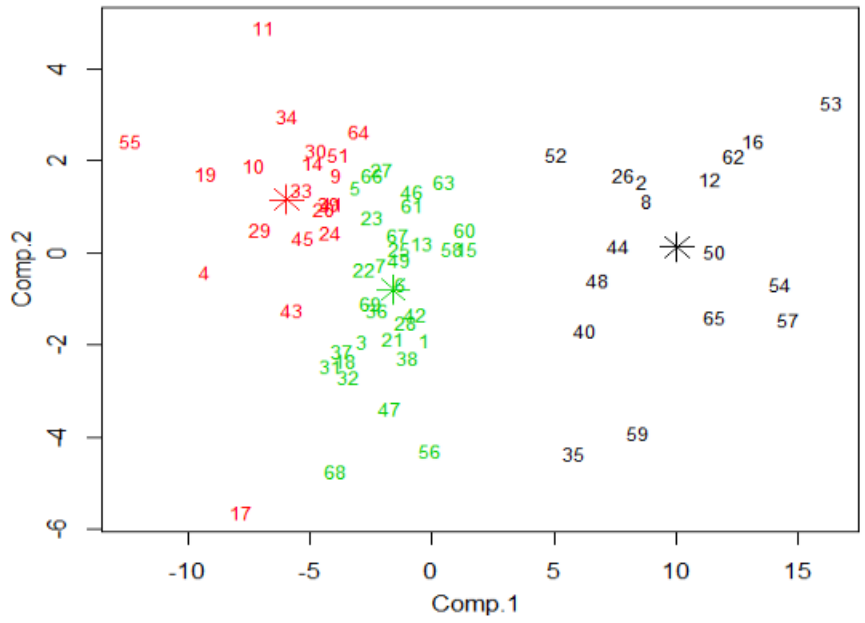


Figura 79: Clustering kmeans (Fuente: Elaboración Propia)

En la Figura 79 se puede observar los tres grupos obtenidos con el clustering kmeans, de color rojo están representados los alumnos de rendimiento alto, de color verde los alumnos de rendimiento medio y de color negro los alumnos de rendimiento bajo.

Luego se guardó en un archivo CSV llamado *notas_Kmeans* el resultado de la agrupación, como se observa en la Figura 80, donde:

- Los registros de color anaranjado representan a los alumnos de rendimiento bajo.
- Los registros de color verde representan a los alumnos de rendimiento medio.
- Los registros de color blanco representan a los alumnos de rendimiento alto.

ARCHIVO INICIO INSERTAR DISEÑO DE PÁGINA FÓRMULAS DATOS REVISAR VISTA Inicia

E21 | 13

	A	B	C	D	E	F	G	H
1	CODIGO	CALC_DIFE	DESA_HUMA	COMU_ORAL	FUND_PROG	PROG_I	ESTR_DISC_I	CLUSTER
2	1	13	11	11	12	12	13	2
3	2	9	11	9	10	5	8	1
4	3	15	12	11	13	13	14	2
5	4	16	17	12	15	17	17	3
6	5	14	14	14	12	12	14	2
7	6	12	13	11	13	12	14	2
8	7	14	13	12	13	12	13	2
9	8	9	11	9	8	7	7	1
10	9	11	16	14	11	15	15	3
11	10	16	17	14	13	15	15	3
12	11	15	20	14	12	14	15	3
13	12	5	12	8	6	8	6	1
14	13	12	14	11	11	13	12	2
15	14	15	17	12	12	14	14	3
16	15	9	13	11	12	11	13	2
17	16	4	11	8	5	5	8	1
18	17	16	12	11	16	19	15	3

notas_Kmeans

LISTO 100 %

Figura 80: Archivo generado notas_Kmeans.CSV (Fuente: Elaboración Propia)

Con la segmentación de alumnos se puede mejorar el proceso de enseñanza/aprendizaje a través de cursos de nivelación personalizada.

Para la validación se utilizó datos de los semestres posteriores correspondientes al tercer, cuarto y quinto semestre llevados en los años 2015 (Impar y Par) y 2016 (Impar). Con los datos del tercer semestre correspondiente al semestre Impar 2015 se realizó la segmentación de tres grupos con el algoritmo kmeans, como se muestra en la Figura 81, para representar a los alumnos de rendimiento BAJO, MEDIO y ALTO.

```
> grupos_Kmeans<-kmeans(notas, 3,iter.max=50, nstart=50)
> grupos_Kmeans
K-means clustering with 3 clusters of sizes 1, 34, 12

Cluster means:
  FISICA_I CALC_INTE TEOR_GRAL_SIST  PROG_II ALG_EST_DAT_I ESTR_DISC_II
1  2.00000      3.00        14.00000  3.00000       3.00000      3.00000
2 11.91176     11.50        14.76471 12.73529      12.08824     12.85294
3 13.58333     14.75        15.25000 16.16667      16.33333     15.66667

Clustering vector:
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26
 3  3  2  2  2  2  2  2  2  2  3  3  3  2  2  2  2  2  2  2  3  2  3  3  2  2
27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47
 2  2  3  3  2  2  1  3  2  2  2  2  2  3  2  2  2  2  2  2  2

Within cluster sum of squares by cluster:
[1]   0.0000 434.9706 132.4167
 (between_SS / total_SS =  63.1 %)
```

Figura 81: Clustering Kmeans – Tercer semestre (Fuente: Elaboración Propia)

En la Figura 82 se observa gráficamente que el 2.13% de la cantidad de alumnos pertenecen al Cluster 1 que son los alumnos con rendimiento bajo, el 72.34% corresponde al Cluster 2 que son los alumnos de rendimiento medio y el 25.53% representa al Cluster 3 que son los alumnos con rendimiento alto.

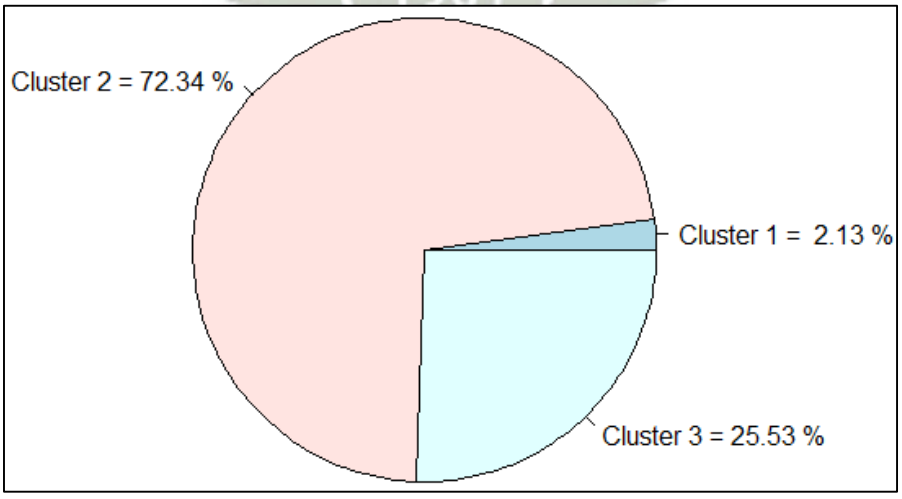


Figura 82: Porcentajes de alumnos en cada cluster – Tercer semestre (Fuente: Elaboración Propia)

Con los datos del cuarto semestre correspondiente al semestre Par 2015 se realizó la segmentación de tres grupos con el algoritmo kmeans, como se muestra en la Figura 83, para representar a los alumnos de rendimiento BAJO, MEDIO y ALTO.

```
> grupos_Kmeans<-kmeans(notas, 3,iter.max=50, nstart=50)
> grupos_Kmeans
K-means clustering with 3 clusters of sizes 1, 20, 13

Cluster means:
  CALC_VECT BASE_DATO DESA_APP_I ALG_EST_DAT_II GEST_FINA FISICA_II
1   8.00000   8.00000   13.00000        6.00000      1.00   2.00000
2  12.10000  13.00000   13.25000       12.20000     11.65  11.30000
3  14.84615  14.84615   14.46154       13.92308     13.00  12.53846

Clustering vector:
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
 3  2  2  2  2  2  3  3  3  1  2  3  2  2  3  2  3  2  2  3  3  2  2  3  3  2  2  2  2  3  3
32 33 34
 2  2  2

Within cluster sum of squares by cluster:
[1]   0.0000 219.5000 126.7692
 (between_SS / total_SS =  57.4 %)
```

Figura 83: Clustering Kmeans – Cuarto semestre (Fuente: Elaboración Propia)

En la Figura 84 se observa gráficamente que el 2.94% de la cantidad de alumnos pertenecen al Cluster 1 que son los alumnos con rendimiento bajo, el 58.82% corresponde al Cluster 2 que son los alumnos de rendimiento medio y el 38.24% representa al Cluster 3 que son los alumnos con rendimiento alto.

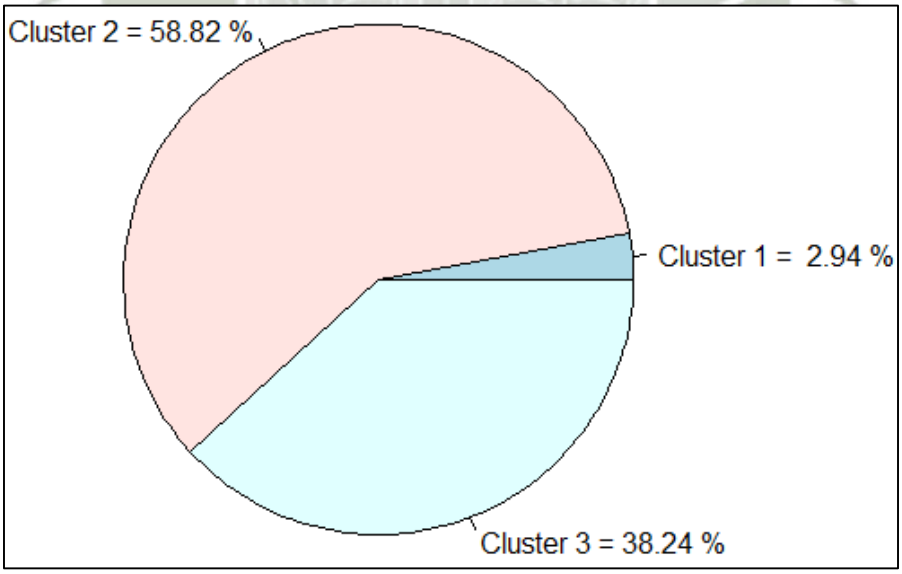


Figura 84: Porcentajes de alumnos en cada cluster – Cuarto semestre (Fuente: Elaboración Propia)

Con los datos del quinto semestre correspondiente al semestre Impar 2016 se realizó la segmentación de tres grupos con el algoritmo kmeans, como se muestra en la Figura 85, para representar a los alumnos de rendimiento BAJO, MEDIO y ALTO.

```
> grupos_Kmeans<-kmeans(notas, 3,iter.max=50, nstart=50)
> grupos_Kmeans
K-means clustering with 3 clusters of sizes 1, 21, 15

Cluster means:
  SIST_INF EST_PROB DINA_SIST DESA_APP_II INF_RED_I  ADM_EMP SIST_OPER
1  0.00000      0.0   0.00000     0.00000   0.00000  0.00000   0.00000
2 12.47619     13.0  12.28571    13.90476  13.04762 13.04762  11.61905
3 13.46667     16.4  13.26667    15.06667  14.26667 13.86667  12.93333

Clustering vector:
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28
 3  2  2  2  2  3  3  1  2  3  2  3  2  2  2  3  2  3  2  2  3  2  3  3  2  2  3  3
29 30 31 32 33 34 35 36 37
 2  2  2  3  3  2  2  2  3

Within cluster sum of squares by cluster:
[1]   0.0000 246.1905 250.8000
 (between_SS / total_SS =  73.6 %)
```

Figura 85: Clustering Kmeans – Quinto semestre (Fuente: Elaboración Propia)

En la Figura 86 se observa gráficamente que el 2.7% de la cantidad de alumnos pertenecen al Cluster 1 que son los alumnos con rendimiento bajo, el 56.8% corresponde al Cluster 2 que son los alumnos de rendimiento medio y el 40.5% representa al Cluster 3 que son los alumnos con rendimiento alto.

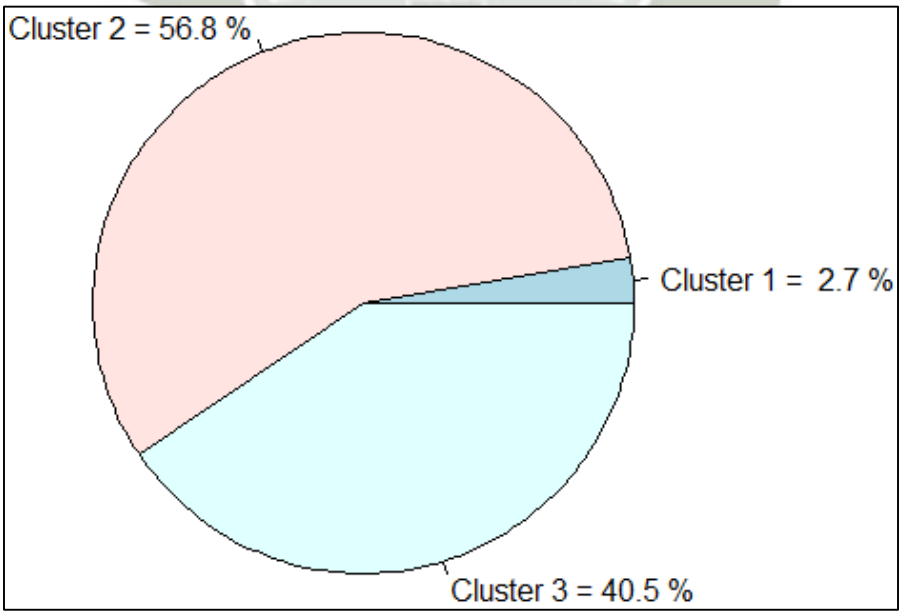


Figura 86: Porcentajes de alumnos en cada cluster – Quinto semestre (Fuente: Elaboración Propia)

En las pruebas y resultados se pudo observar que en el primer año es donde se produce con mayor frecuencia la deserción y el atraso estudiantil, reduciéndose la cantidad de alumnos regulares a partir del segundo año, los cuales se han mantenido dentro del rendimiento MEDIO y ALTO.

Luego se realizó la segmentación de tres grupos con el algoritmo kmeans para los alumnos del primer semestre correspondiente al semestre Impar 2016 en el curso de metodología de la programación correspondiente a las tres fases académicas de las que consta el semestre.

En la Figura 87 se muestra el resultado del clustering kmeans correspondiente a la primera fase del curso de metodología de la programación, para representar a los alumnos de rendimiento BAJO, MEDIO y ALTO, resultando los dos primeros grupos los alumnos que requieren un reforzamiento académico.

```
> grupos_Kmeans<-kmeans(notas, 3,iter.max=50, nstart=50)
> grupos_Kmeans
K-means clustering with 3 clusters of sizes 21, 58, 40

Cluster means:
   MET_PROG
1  6.047619
2 11.051724
3 15.950000

Clustering vector:
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25
  2   3   1   2   2   3   3   2   2   3   3   2   3   3   2   2   2   3   2   3   3   3   3   3   2
 26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50
  3   2   2   2   2   2   2   1   2   1   1   3   2   1   2   1   2   3   1   1   2   1   3   2   2
 51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75
  3   2   2   2   3   3   2   1   2   1   3   3   3   1   2   3   3   3   3   1   2   2   1   2   2
 76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99 100
  3   2   3   2   2   3   2   1   3   2   3   1   2   2   3   2   3   2   1   2   2   2   2   2   1
101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119
  1   3   2   2   2   1   2   3   3   3   3   2   2   2   2   3   1   2   2

Within cluster sum of squares by cluster:
[1] 54.95238 98.84483 97.90000
 (between_SS / total_SS =  84.9 %)
```

Figura 87: Clustering Kmeans – Primera fase (Fuente: Elaboración Propia)

En la Figura 88 se muestra un gráfico de barras construido a partir de los promedios de los clusters obtenidos con la segmentación, en donde se puede apreciar que el Cluster 1 corresponde a los alumnos de bajo rendimiento, el Cluster 2 a los alumnos de

rendimiento medio y el Cluster 3 a los alumnos de alto rendimiento en la primera fase del curso de metodología de la programación.

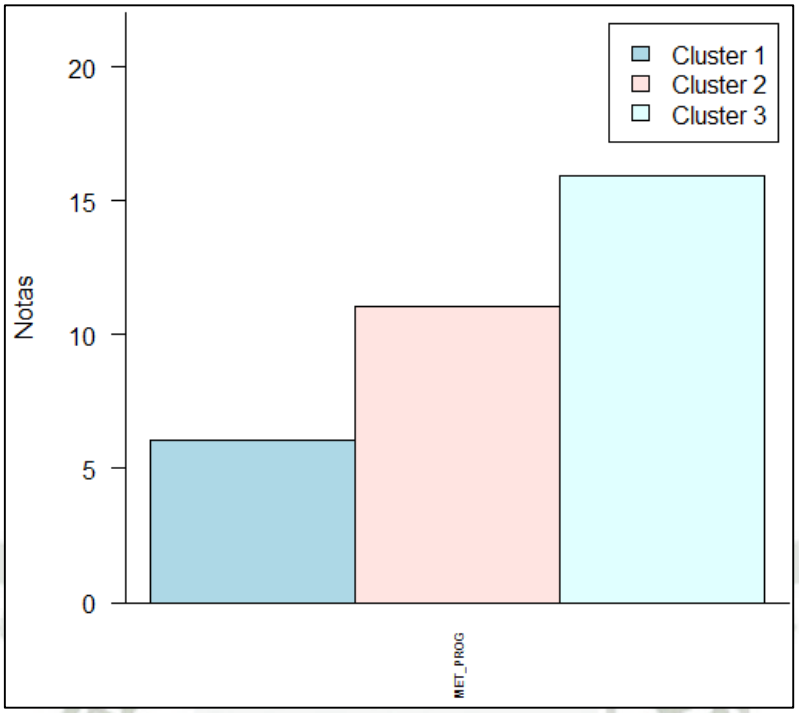


Figura 88: Promedios de notas de los clusters – Primera fase (Fuente: Elaboración Propia)

En la Figura 89 se muestra el resultado del clustering kmeans correspondiente a la segunda fase del curso de metodología de la programación, para representar a los alumnos de rendimiento BAJO, MEDIO y ALTO, resultando los dos primeros grupos los alumnos que requieren un reforzamiento académico.

```
> grupos_Kmeans<-kmeans(notas, 3,iter.max=50, nstart=50)
> grupos_Kmeans
K-means clustering with 3 clusters of sizes 23, 46, 45

Cluster means:
   MET_PROG
1  6.086957
2 11.260870
3 16.222222

Clustering vector:
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20  21  22  23
  2   3   3   2   3   3   2   2   3   2   3   3   2   3   2   3   2   2   3   1   2   3   2
 24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46
  3   3   3   2   2   3   3   1   2   1   1   2   1   2   2   1   3   3   1   2   2   1   3
 47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69
  3   1   3   3   2   2   2   3   2   2   2   1   3   3   2   2   3   3   3   3   1   2   3
 70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92
  1   2   3   2   2   3   1   2   3   3   1   3   2   2   1   2   3   3   1   3   2   1   2
 93  94  95  96  97  98  99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114
  2   1   3   1   1   2   1   1   2   1   2   3   2   3   3   2   3   3   2   3   2   2

Within cluster sum of squares by cluster:
[1]  73.82609  80.86957 105.77778
 (between_SS / total_SS =  86.2 %)
```

Figura 89: Clustering Kmeans – Segunda fase (Fuente: Elaboración Propia)

En la Figura 90 se muestra un gráfico de barras construido a partir de los promedios de los clusters obtenidos con la segmentación, en donde se puede apreciar que el Cluster 1 corresponde a los alumnos de bajo rendimiento, el Cluster 2 a los alumnos de rendimiento medio y el Cluster 3 a los alumnos de alto rendimiento en la segunda fase del curso de metodología de la programación.

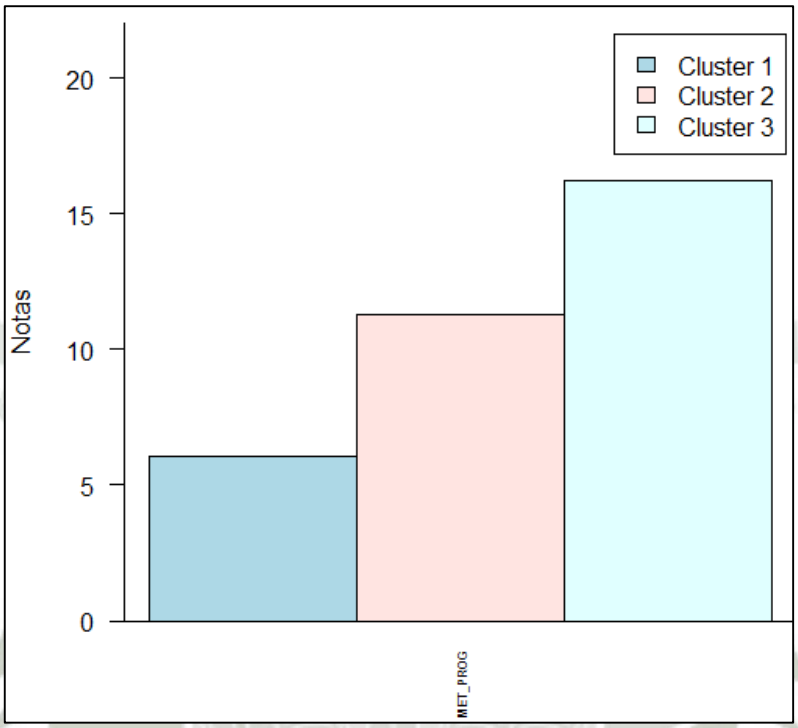


Figura 90: Promedios de notas de los clusters – Segunda fase (Fuente: Elaboración Propia)

En la Figura 91 se muestra el resultado del clustering kmeans correspondiente a la tercera fase del curso de metodología de la programación, para representar a los alumnos de rendimiento BAJO, MEDIO y ALTO, resultando los dos primeros grupos los alumnos que requieren un reforzamiento académico.

```
> grupos_Kmeans<-kmeans(notas, 3,iter.max=50, nstart=50)
> grupos_Kmeans
K-means clustering with 3 clusters of sizes 16, 58, 34

Cluster means:
  MET_PROG
1  4.93750
2 12.06897
3 16.44118

Clustering vector:
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20  21  22  23
  2   3   2   2   2   3   2   2   2   3   3   3   2   2   2   3   2   2   3   2   2   3   2
 24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46
  3   3   2   2   2   2   3   2   3   1   1   2   1   2   1   3   3   1   2   2   3   1   1
 47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69
  3   2   2   2   2   3   2   1   2   1   3   3   2   2   3   3   3   3   2   2   1   3   2
 70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92
  2   2   3   2   3   3   2   2   3   2   1   3   3   3   1   3   2   2   2   1   1   2   2
 93  94  95  96  97  98  99 100 101 102 103 104 105 106 107 108
  1   1   2   2   2   3   2   3   2   2   2   2   2   2   2   2

Within cluster sum of squares by cluster:
[1]  64.93750 113.72414  58.38235
 (between_SS / total_SS =  86.0 %)
```

Figura 91: Clustering Kmeans – Tercera fase (Fuente: Elaboración Propia)

En la Figura 92 se muestra un gráfico de barras construido a partir de los promedios de los clusters obtenidos con la segmentación, en donde se puede apreciar que el Cluster 1 corresponde a los alumnos de bajo rendimiento, el Cluster 2 a los alumnos de rendimiento medio y el Cluster 3 a los alumnos de alto rendimiento en la tercera fase del curso de metodología de la programación.

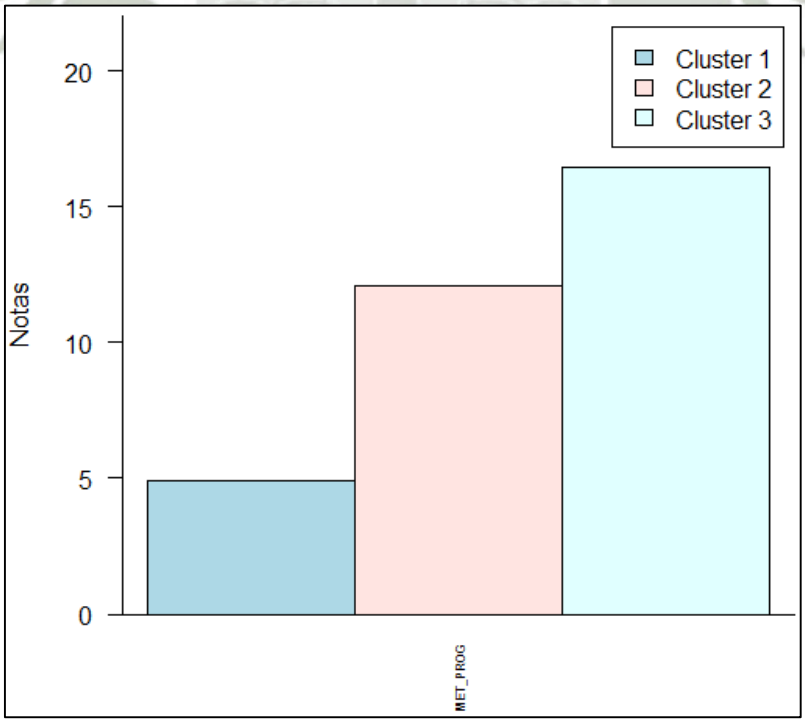


Figura 92: Promedios de notas de los clusters – Tercera fase (Fuente: Elaboración Propia)

La EPIS de la UCSM estableció estrategias de nivelación académica impartiendo cursos de reforzamiento los fines de semana, a través de los cuales se pudo mejorar el rendimiento académico de los alumnos e incrementar la cantidad de aprobados en cada fase como se muestra en los siguientes resultados del curso de metodología de la programación:

En la Tabla 16 se muestra las cantidades de alumnos aprobados, desaprobados y NSP obtenidos de los promedios de la primera fase, en la Figura 93 se muestra de forma gráfica dichos resultados representados por porcentajes calculados con respecto al total.

Tabla 16: Resultados de la primera fase (Fuente: Elaboración Propia)

Fase 1	
Aprobados	65
Desaprobados	54
NSP	12
Total	131

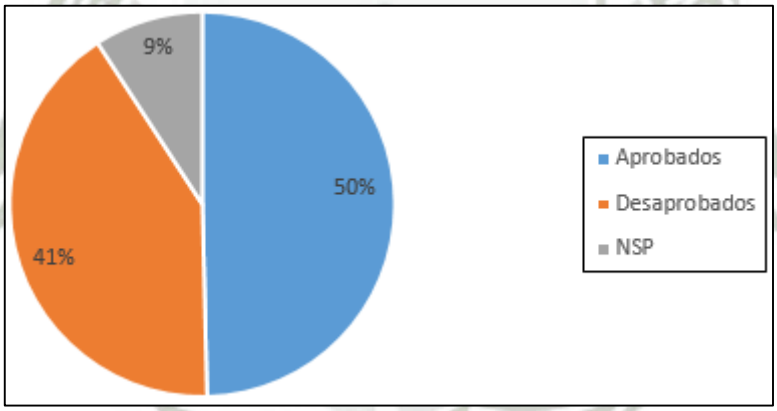


Figura 93: Porcentajes de aprobados, desaprobados y NSP – Fase 1 (Fuente: Elaboración Propia)

En la Tabla 17 se muestra las cantidades de alumnos aprobados, desaprobados y NSP obtenidos de los promedios de la segunda fase, en la Figura 94 se observa un incremento de 3% en la cantidad de aprobados con respecto a la primera fase.

Tabla 17: Resultados de la segunda fase (Fuente: Elaboración Propia)

Fase 2	
Aprobados	69
Desaprobados	45
NSP	17
Total	131

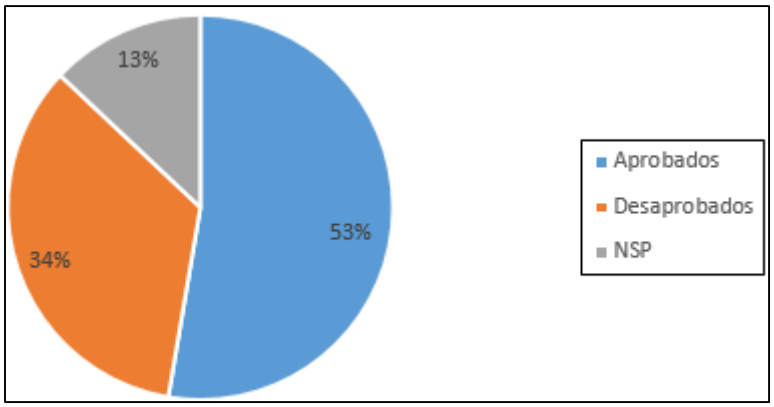


Figura 94: Porcentajes de aprobados, desaprobados y NSP – Fase 2 (Fuente: Elaboración Propia)

En la Tabla 18 se muestra las cantidades de alumnos aprobados, desaprobados y NSP obtenidos de los promedios de la tercera fase, en la Figura 95 se observa un incremento de 5% en la cantidad de aprobados con respecto a la segunda fase.

Tabla 18: Resultados de la tercera fase (Fuente: Elaboración Propia)

Fase 3	
Aprobados	76
Desaprobados	32
NSP	23
Total	131

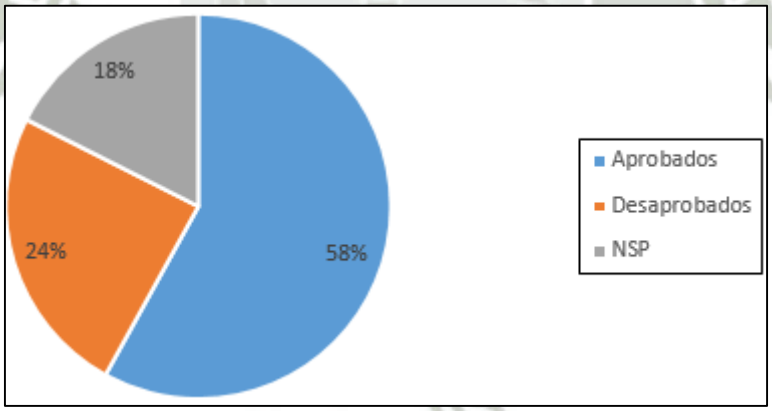


Figura 95: Porcentajes de aprobados, desaprobados y NSP – Fase 3 (Fuente: Elaboración Propia)

En la Tabla 19 se muestra las cantidades de alumnos aprobados, desaprobados y NSP obtenidos de los promedios finales del curso de metodología de programación, y en la Figura 96 se muestra de forma gráfica dichos resultados representados por porcentajes calculados con respecto al total, se observa un incremento de 13%, 10% y 5% con respecto a la primera, segunda y tercera fase respectivamente.

Tabla 19: Resultados del promedio final (Fuente: Elaboración Propia)

Promedio Final	
Aprobados	82
Desaprobados	37
NSP	12
Total	131

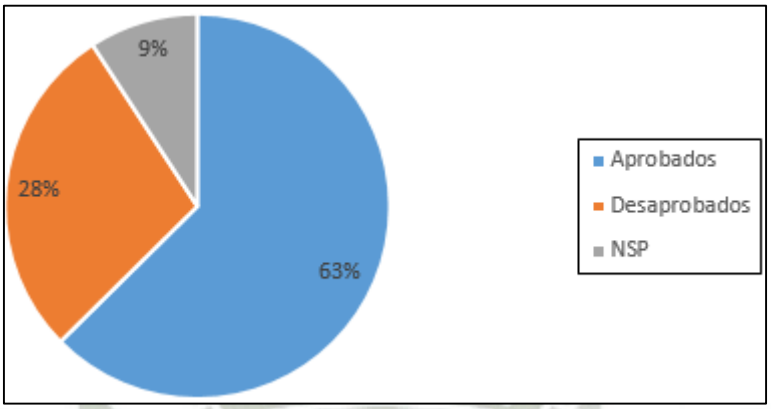


Figura 96: Porcentajes de aprobados, desaprobados y NSP – Promedio Final (Fuente: Elaboración Propia)

Se recomienda realizar la segmentación de alumnos para el reforzamiento académico de los cursos con mayor cantidad de promedios bajos desde el primer año, los cuales pueden ser determinados luego de una exploración de datos académicos, para detectar alumnos sobre los cuales se debe prestar atención con el objeto de mejorar su rendimiento a través de cursos de nivelación personalizada, y de este modo reducir la deserción y reincidencia de matrícula.

CONCLUSIONES

PRIMERA.- Se logró realizar un estudio comparativo de técnicas no supervisadas de minería de datos para segmentación de alumnos en tres agrupaciones correspondientes a rendimiento BAJO, MEDIO y ALTO, obteniendo grupos de mejor calidad, con menor distancia intra-cluster y mayor distancia inter-cluster, que representa mayor homogeneidad dentro del grupo y mayor diferencia entre los grupos.

SEGUNDA.- Se analizó diferentes técnicas no supervisadas de minería de datos, como son Kmeans y PAM dentro del clustering particional y los métodos del clustering jerárquico aglomerativo para realizar la segmentación académica.

TERCERA.- Se utilizó como estudio de caso los registros académicos de los alumnos del II Semestre de la Escuela Profesional de Ingeniería de Sistemas de la Universidad Católica de Santa María correspondiente al semestre Par 2014, a partir del cual se pudo realizar el estudio comparativo para la segmentación académica.

CUARTA.- Se utilizó medidas como las distancias intra-cluster e inter-cluster obtenidas por las diferentes técnicas de agrupación aplicadas, resultando la técnica de clustering particional Kmeans con mayor similitud dentro de la agrupación y mayor separación entre los grupos formados. Finalmente se utilizó el coeficiente de silueta para medir la calidad de las agrupaciones formadas por las técnicas utilizadas que permitieron seleccionar el método Kmeans por ser la técnica de clustering con la que se obtiene grupos de mejor calidad para la segmentación académica en tres grupos para el reforzamiento de los alumnos en los niveles básico, intermedio y avanzado.

RECOMENDACIONES

PRIMERA.- Se recomienda investigar y comparar otras técnicas no supervisadas de minería de datos para la segmentación como son agrupaciones basadas en densidad, algoritmos basados en redes neuronales, lógica difusa, algoritmos híbridos.

SEGUNDA.- Se recomienda investigar funciones para calcular la matriz de distancias en tablas de datos cuyas variables están mezcladas entre variables cualitativas y cuantitativas.

TERCERA.- Se recomienda utilizar el coeficiente de silueta y la suma de cuadrados de error (SSE) para determinar la cantidad de grupos en la que se segmentará los datos para las diferentes técnicas de clustering.

CUARTA.- Se recomienda investigar y realizar la integración de los algoritmos implementados de las herramientas de minería de datos en los diferentes lenguajes de programación para construir sistemas de información a través de los cuales se pueda realizar consultas relacionadas a minería de datos y generar conocimiento.

BIBLIOGRAFÍA

Albarrán, S. y Salgado, M. (2013). La Inteligencia Analítica y la Competitividad en las Empresas. *Revista de Estudios en Contaduría, Administración e Informática (RECAI), 2(3)*, 24-47.

Ayala, G. (2014). *Análisis de Datos con R*. Recuperado de http://www.uv.es/ayala/docencia/ad/ad13.pdf

Azevedo, A. y Santos, M. (2008). KDD, SEMMA and CRISP-DM: A parallel overview. *IADIS European Conference Data Mining*, 182-185.

Bioinformática (2016). *Introducción al clustering en bioinformática*, Recuperado de http://genoma.unsam.edu.ar/trac/docencia/wiki/Bioinformatica/Guias/DataMining.

Borao, D. (2013). *Incidencia del ruido en los datos de test sobre la precisión de modelos de clasificación y regresión*. Tesis de Máster Universitario en Ingeniería del Software, Métodos Formales y Sistemas de Información. Universidad Politécnica de Valencia, España.

Camargo. H. y Silva, M. (2010). Dos caminos en la búsqueda de patrones por medio de Minería de Datos: SEMMA y CRISP. *Revista de Tecnología - Journal of Technology*, *9(1)*, 11-18.

Chamba, S. (2015). *Minería de datos para segmentación de clientes en la empresa tecnológica Master PC*. Tesis para optar el título de Ingeniera de Sistemas. Universidad Nacional de Loja, Ecuador.

Chapman, P., Clinton, J., Kerbe.r R., Khabaza T., Reinartz, T., Shearer C. y Wirth R. (2000). *CRISP-DM 1.0*. Washington D. C., EEUU: SPSS.

Chen, M., Han, J. y Yu, P. (1996). Data Mining: An Overview from a Database Perspective. *IEEE Transactions on Knowledge and Data Engineering*, *8(6)*, 866-883.

DTyOC De Tecnología y Otras Cosas (2015). *SAS Analytics y Enterprise*. Recuperado de https://dtyoc.com/2015/05/09/sas-analytics-y-enterprise/

Eckert, K. y Suénaga, R. (2013). *Aplicación de técnicas de minería de datos al análisis de situación y comportamiento académico de alumnos de la UGD*. Recuperado de http://sedici.unlp.edu.ar/handle/10915/27103.

Flores, C. (2014). *Exigencias de calidad de suministro en base a densidad de consumo mediante técnicas de minería de datos*. Memoria para optar al título de ingeniero civil electricista, Universidad de Chile, Santiago, Chile.

Gallardo, M. (2009). *Aplicación de técnicas de clustering para la mejora del aprendizaje*. Proyecto de fin de carrera en Ingeniería de Telecomunicación, Universidad Carlos III de Madrid, España.

García, F. (2013). *Aplicación de técnicas de Minería de Datos a datos obtenidos por el Centro Andaluz de Medio Ambiente (CEAMA)*. Trabajo Fin de Máster Universitario en Estadística Aplicada. Universidad de Granada, España.

Han, J. y Kamber, M. (2006). *Data Mining. Concepts and Techniques*. San Francisco, CA, EEUU: Morgan Kaufmann.

Han, J., Kamber, M. y Pei, J. (2011). *Data Mining: Concepts and Techniques*, *3rd ed*. San Francisco, CA, EEUU: Morgan Kaufmann.

Hernández, E. (2006). *Algoritmo de clustering basado en entropía para descubrir grupos en atributos de tipo mixto*. Tesis para obtener el grado de Maestro en Ciencias en la especialidad de Ingeniería Eléctrica Opción Computación. Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, Departamento de Ingeniería Eléctrica y Computación, México.

Hernández, J., Ramírez, M. y Ferri, C. (2004). *Introducción a la Minería de Datos*. Madrid, España: Pearson.

Hernández, J. (2006). *Minería de Datos. El proceso de KDD*. Recuperado de http://users.dsic.upv.es/~jorallo/master/dm2.pdf

Jain, A., Murty, M. y Flynn, P. (1999) Data clustering: a review. *ACM Computing Surveys, 31(3)*, 264–323.

KDnuggets (2014). *What main methodology are you using for your analytics, data mining, or data science projects?* Recuperado de http://www.kdnuggets.com/polls/2014/analytics-data-mining-data-science-methodology.html.

KDnuggets (2015). *Analytics, Data Mining, Data Science software/tools used in the past 12 months* Recuperado de http://www.kdnuggets.com/polls/2015/analytics-data-mining-data-science-software-used.html.

León, N. y Muñoz, C. (2013). *Propuesta de gestión de información de agrupamiento (clustering), utilizando técnicas de minería de datos para egresados del programa profesional de Ingeniería de Sistemas de la Universidad Católica de Santa María*. Tesis para optar el título profesional de Ingeniero de Sistemas. Universidad Católica de Santa María, Arequipa, Perú.

Maimon, O. y Rokach, L. (2010). *Data Mining and Knowledge Discovery Handbook, Second Edition*. New York, EEUU: Springer.

Microsoft Corporation (2005). *SQL Server Analysis Services*. Recuperado de https://technet.microsoft.com/es-es/library/ms175609(v=sql.90).aspx.

Microsoft Corporation (2015). *Minería de datos (SSAS)*. Recuperado de https://msdn.microsoft.com/es-es/library/bb510516%28v=sql.120%29.aspx.

Microsoft Corporation (2016). *Conceptos de Minería de Datos. SQL Server 2016*. Recuperado de https://msdn.microsoft.com/es-es/library/ms174949.aspx.

Microsoft Corporation (2016). *Minería de datos (SSAS)*. Recuperado de https://msdn.microsoft.com/es-es/library/bb510516(v=sql.130).aspx.

Moine, J., Haedo, A. y Gordillo, S. (2011). Estudio comparativo de metodologías para minería de datos. *XIII Worshop de Investigadores en Ciencias de la computación*, 278-281, ISBN:978-950-673-892-1

Morelo, K. (2014). *Sistema para caracterización de perfiles de clientes de la empresa Zona T*. Proyecto de grado para optar el título de Ingeniero de Sistemas. Universidad de Cartagena, Cartagena de Indias, Colombia.

Moreno, M., Miguel, L., García, F. y Polo, M. (2001). Aplicación de Técnicas de Minería de Datos en la Construcción y Validación de Modelos Predictivos y Asociativos a Partir de Especificaciones de Requisitos De Software. *CEUR Workshop Proceedings - Decision Support in Software Engineering, 84*.

Oporto. S. (2009). *El Proceso de la Minería de Datos*. Recuperado de http://es.slideshare.net/bemaguali/mineria-de-datos-1867890.

Oporto, S. (2014). *Introducción a la Minería de Datos. Introducción a Técnicas de Minería de Datos.* Recuperado de http://slideplayer.es/slide/92234/

Paradis. E. (2003). *R para Principiantes*. Recuperado de https://cran.r-project.org/doc/contrib/rdebuts_es.pdf

Pascual, D., Pla, F. y Sánchez, J. (2007). Algoritmos de agrupamiento. *Métodos Informáticos Avanzados, E-treballs d'informàtica i tecnologia*. R. Quirós, F. Pla, J. M. Badía y M. Chover eds. 2007. pp. 163-175

Pérez, C. y Santín, D. (2006). *Data Mining - Soluciones con Enterprise Miner con 1 CD.* México: Alfaomega.

Qualex Consulting Services Inc. (2013). *SAS Enterprise Miner*. Recuperado de http://www.qlx.com/Software_Sales/enterprise_miner.html.

Revista de Actuarios (2006). *Minería de Datos (Data Mining). Análisis avanzados de grandes volúmenes de datos en el sector seguros (2ª parte)*. Recuperado de http://www.actuarios.org/espa/web-nueva/publicaciones/revista/revista25/datamining.htm

Riquelme, J., Ruiz, R. y Gilbert, K. (2006). Minería de datos: Conceptos y Tendencias. *Revista Iberoamericana de Inteligencia Artificial, 10(29),* 11-18.

Rodríguez, M., Álvarez, J., Mesa, J. y Gonzáles, A. (2003). *Metodologías para la realización de proyectos de Data Mining*. Recuperado de http://www.aeipro.com/files/congresos/2003pamplona/ciip03_0257_0265.2134.pdf

Romero, C. y Ventura, S. (2007). Educational data mining: A survey from 1995 to 2005. *Expert Systems with Applications,* 33(1), 135-146

RStudio (2015). *Take control of your R code*. Recuperado de https://www.rstudio.com/products/rstudio/

SAS Institute (1998). *Data Mining and the Case for Sampling*. Cary, NC, EEUU: SAS Institute Inc.

Sulla, J. (2015). *Aplicación de técnicas supervisadas de minería de datos para determinar la predicción de deserción académica*. Tesis de segunda especialidad en Ingeniería de Software. Universidad Católica de Santa María, Arequipa, Perú.

Tan, P., Steinbach, M. y Kumar, V. (2006). *Introduction to Data Mining*. New York, EEUU: Pearson Education.

Thuraisingham, B. (2000) A Primer for Understanding and Applying Data Mining. *IT Professional*, *2(1)*, 28-31.

Villazana, S., Arteaga, F., Seijas, C. y Rodriguez, O. (2012). Estudio Comparativo entre Algoritmos de Agrupamiento Basado en SVM y C-medios Difuso Aplicados a Señales Electrocardiográficas Arrítmicas. *Revista Ingeniería UC, 19(1)*, 16-24.

Weiss, S. y Indurkhya, N. (1998). *Predictive Data Mining: A Practical Guide*. San Francisco, CA, EEUU: Morgan Kaufmann.

WEKA (2016). *Weka 3: Data Mining Software in Java*. Recuperado de http://www.cs.waikato.ac.nz/ml/weka/index.html.

Wikipedia La enciclopedia libre (2015). *RapidMiner*. Recuperado de https://es.wikipedia.org/wiki/RapidMiner.

Wikipedia La enciclopedia libre (2016). *Weka (aprendizaje automático)*. Recuperado de https://es.wikipedia.org/wiki/Weka_(aprendizaje_autom%C3%A1tico).

Witten, I. y Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques, Second Edition*. San Francisco, CA, EEUU: Morgan Kaufmann.

ANEXOS

Anexo A: Distancias intra-cluster e inter-cluster

Clustering jerárquico aglomerativo

- **Utilizando el método ward:**

Tabla 20: Distancia Intra-Cluster – Método ward (Fuente: Elaboración Propia)

Cluster 1	Cluster 2	Cluster 3
5.69140625	14.71972318	19.41
7.00390625	5.837370242	16.41
11.25390625	14.5432526	5.41
4.12890625	24.42560554	15.11
4.44140625	18.07266436	4.11
5.44140625	63.30795848	17.81
20.81640625	27.5432526	6.21
60.56640625	11.07266436	10.21
8.00390625	19.07266436	9.41
7.75390625	33.60207612	6.11
7.94140625	43.36678201	6.11
6.62890625	60.01384083	4.31
4.06640625	32.66089965	6.91
13.56640625	26.24913495	9.21
6.62890625	103.9550173	10.01
14.50390625	57.30795848	5.11
7.69140625	16.01384083	38.21
4.69140625	**571.7647059**	7.21
5.50390625		51.01
4.69140625		13.51
7.19140625		**261.8**
7.44140625		
9.81640625		
25.00390625		
13.31640625		
21.44140625		
8.87890625		
15.56640625		
9.75390625		
15.12890625		
26.87890625		
6.44140625		
377.875		

Total distancia Intra-Cluster (WSS) = 377.875 + 571.7647059 + 261.8
Total distancia Intra-Cluster (WSS) = 1211.44

Tabla 21: Distancia Inter-Cluster – Método ward (Fuente: Elaboración Propia)

Cluster 1	4.656907
Cluster 2	101.3308
Cluster 3	34.30231
Total (BSS) =	**2557.691**

- **Utilizando el método single (Agregación del salto mínimo):**

Tabla 22: Distancia Intra-Cluster – Método single (Fuente: Elaboración Propia)

Cluster 1	Cluster 2	Cluster 3
7.420584	0	0
94.34596	**0**	**0**
16.16685		
85.24148		
15.24148		
5.166852		
7.704166		
88.77879		
30.10715		
53.30118		
70.28626		
150.54		
1.286255		
28.137		
11.70417		
193.9131		
90.56984		
18.55491		
85.46536		
22.53999		
14.51014		
12.52506		
7.65939		
21.03252		
3.375808		
79.137		
11.98775		
8.017599		
52.42058		
28.04745		
28.71909		
20.00267		
32.65939		
44.27133		
88.1967		

9.674315
16.37581
7.808643
22.42058
56.37581
22.22655
10.00267
35.31611
69.68924
26.9579
20.39073
15.94297
60.37581
7.405658
171.7937
23.21163
49.64446
299.1818
223.1072
157.4654
33.1967
227.5698
7.599688
12.48029
5.360882
7.137002
19.8982
151.7191
9.68924
14.34596
43.9579
10.46536
3377.821

Total distancia Intra-Cluster (WSS) = 3377.821 + 0 + 0
Total distancia Intra-Cluster (WSS) = 3377.82

Tabla 23: Distancia Inter-Cluster – Método single (Fuente: Elaboración Propia)

Cluster 1	0.123214
Cluster 2	172.8387
Cluster 3	210.2155
Total (BSS) =	**391.3095**

- **Utilizando el método complete (Agregación del salto máximo):**

Tabla 24: Distancia Intra-Cluster – Método complete (Fuente: Elaboración Propia)

Cluster 1	Cluster 2	Cluster 3
8.25692	14.71972	13.94753
10.66869	5.83737	16.72531
3.786332	14.54325	3.947531
5.609862	24.42561	21.05864
20.84516	18.07266	5.83642
3.786332	63.30796	44.6142
17.37457	27.54325	12.83642
10.60986	11.07266	8.83642
9.845156	19.07266	10.39198
9.08045	33.60208	6.169753
4.609862	43.36678	7.169753
2.25692	60.01384	4.280864
10.1981	32.6609	9.058642
6.668685	26.24913	10.39198
19.37457	103.955	8.058642
10.66869	57.30796	2.83642
4.551038	16.01384	46.72531
7.727509	**571.7647**	43.05864
6.198097		**275.9444**
13.90398		
9.08045		
20.9628		
9.903979		
7.903979		
16.9628		
30.02163		
14.02163		
17.37457		
7.374567		
12.72751		
16.49221		
7.962803		
11.55104		
5.903979		
374.2647		

Total distancia Intra-Cluster (WSS) = 374.2647 + 571.7647 + 275.9444
Total distancia Intra-Cluster (WSS) = 1221.96

Tabla 25: Distancia Inter-Cluster – Método complete (Fuente: Elaboración Propia)

Cluster 1	3.479241
Cluster 2	101.3308
Cluster 3	39.2355
Total (BSS) =	**2547.157**

- **Utilizando el método average (Agregación del Promedio):**

Tabla 26: Distancia Intra-Cluster – Método average (Fuente: Elaboración Propia)

Cluster 1	Cluster 2	Cluster 2
16.0466	13.37333	23.25
9.738905	4.106667	23.25
40.66198	14.37333	**46.5**
7.508136	21.70667	
7.700444	16.10667	
6.238905	33.50667	
17.20044	12.44	
21.27737	22.57333	
40.73891	37.70667	
9.661982	42.90667	
10.50814	51.37333	
30.16198	27.04	
54.62352	23.50667	
8.085059	54.17333	
41.58506	13.50667	
6.35429	**388.4**	
14.50814		
7.161982		
3.931213		
6.161982		
5.661982		
10.93121		
12.35429		
22.0466		
10.85429		
15.12352		
9.546598		
11.81583		
19.58506		
6.277367		
5.046598		
12.39275		
8.123521		
8.161982		

15.93121
11.5466
6.585059
26.20044
15.46967
10.81583
11.0466
95.20044
41.85429
23.81583
30.58506
10.81583
20.93121
13.46967
6.085059
15.46967
31.62352
6.35429
911.5769

Total distancia Intra-Cluster (WSS) = 911.5769 + 388.4 + 46.5
Total distancia Intra-Cluster (WSS) = 1346.48

Tabla 27: Distancia Inter-Cluster – Método average (Fuente: Elaboración Propia)

Cluster 1	10.8301
Cluster 2	109.9946
Cluster 3	104.7843
Total (BSS) =	**2422.654**

- **Utilizando el método mcquitty:**

Tabla 28: Distancia Intra-Cluster – Método mcquitty (Fuente: Elaboración Propia)

Cluster 1	Cluster 2	Cluster 3
16.0466	13.48047	0
9.738905	3.980469	**0**
40.66198	15.23047	
7.508136	24.98047	
7.700444	16.10547	
6.238905	68.35547	
17.20044	29.10547	
21.27737	10.73047	
40.73891	19.73047	
9.661982	36.73047	
10.50814	41.23047	
30.16198	56.73047	

54.62352	30.23047
8.085059	25.23047
41.58506	54.48047
6.35429	14.98047
14.50814	**461.3125**
7.161982	
3.931213	
6.161982	
5.661982	
10.93121	
12.35429	
22.0466	
10.85429	
15.12352	
9.546598	
11.81583	
19.58506	
6.277367	
5.046598	
12.39275	
8.123521	
8.161982	
15.93121	
11.5466	
6.585059	
26.20044	
15.46967	
10.81583	
11.0466	
95.20044	
41.85429	
23.81583	
30.58506	
10.81583	
20.93121	
13.46967	
6.085059	
15.46967	
31.62352	
6.35429	
911.5769	

Total distancia Intra-Cluster (WSS) = 911.5769 + 461.3125 + 0
Total distancia Intra-Cluster (WSS) = 1372.89

Tabla 29: Distancia Inter-Cluster – Método mcquitty (Fuente: Elaboración Propia)

Cluster 1	10.8301
Cluster 2	103.7648
Cluster 3	172.8387
Total (BSS) =	**2396.241**

- **Utilizando el método median:**

Tabla 30: Distancia Intra-Cluster – Método median (Fuente: Elaboración Propia)

Cluster 1	Cluster 2	Cluster 3
7.065048	0	0
90.05012	**0**	**0**
16.9755		
89.24415		
17.30385		
5.079973		
8.691914		
84.93072		
32.52773		
57.60236		
74.76654		
145.4979		
1.676988		
30.94564		
10.42326		
187.3785		
93.43818		
19.96057		
90.60236		
24.39341		
14.75162		
14.05012		
9.303854		
22.85609		
3.706839		
75.16953		
14.05012		
8.676988		
56.18445		
30.52773		
30.84117		
20.96057		
35.25908		
47.43818		
83.42326		

9.826242
17.69191
8.050123
24.63221
53.39341
23.66206
10.00535
37.37848
66.22923
29.79639
21.19938
16.06505
57.24415
7.945645
164.9009
24.88594
48.274
216.3636
32.42326
219.7815
7.691914
173.3486
11.79639
5.453108
211.7218
7.587436
22.22923
145.9307
11.06505
14.34863
45.69191
11.274
3313.642

Total distancia Intra-Cluster (WSS) = 3313.642 + 0 + 0
Total distancia Intra-Cluster (WSS) = 3313.642

Tabla 31: Distancia Inter-Cluster – Método median (Fuente: Elaboración Propia)

Cluster 1	0.016002
Cluster 2	289.6068
Cluster 3	164.8097
Total (BSS) =	**455.4886**

- **Utilizando el método centroid:**

Tabla 32: Distancia Intra-Cluster – Método centroid (Fuente: Elaboración Propia)

Cluster 1	Cluster 2	Cluster 3
16.0466	13.48047	0
9.738905	3.980469	**0**
40.66198	15.23047	
7.508136	24.98047	
7.700444	16.10547	
6.238905	68.35547	
17.20044	29.10547	
21.27737	10.73047	
40.73891	19.73047	
9.661982	36.73047	
10.50814	41.23047	
30.16198	56.73047	
54.62352	30.23047	
8.085059	25.23047	
41.58506	54.48047	
6.35429	14.98047	
14.50814	**461.3125**	
7.161982		
3.931213		
6.161982		
5.661982		
10.93121		
12.35429		
22.0466		
10.85429		
15.12352		
9.546598		
11.81583		
19.58506		
6.277367		
5.046598		
12.39275		
8.123521		
8.161982		
15.93121		
11.5466		
6.585059		
26.20044		
15.46967		
10.81583		
11.0466		

95.20044
41.85429
23.81583
30.58506
10.81583
20.93121
13.46967
6.085059
15.46967
31.62352
6.35429
911.5769

Total distancia Intra-Cluster (WSS) = 911.5769 + 461.3125 + 0
Total distancia Intra-Cluster (WSS) = 1372.889

Tabla 33: Distancia Inter-Cluster – Método centroid (Fuente: Elaboración Propia)

Cluster 1	10.8301
Cluster 2	103.7648
Cluster 3	172.8387
Total (BSS) =	**2396.241**

Kmeans

Tabla 34: Distancia Intra-Cluster – Kmeans (Fuente: Elaboración Propia)

Cluster 1	Cluster 2	Cluster 3
14.71972	5.847656	15.775
5.83737	8.535156	19.275
14.54325	11.09766	4.375
24.42561	3.785156	16.875
18.07266	4.722656	4.775
63.30796	4.285156	52.175
27.54325	18.41016	15.075
11.07266	9.535156	7.075
19.07266	8.285156	10.375
33.60208	8.597656	7.575
43.36678	6.097656	6.775
60.01384	3.285156	5.475
32.6609	12.22266	5.575
26.24913	6.347656	8.575
103.955	16.84766	9.775
57.30796	9.347656	8.475
16.01384	4.847656	4.275
571.7647	6.972656	9.775
	4.847656	47.375
	7.222656	17.075
	24.72266	**276.5**
	8.285156	

9.035156
25.91016
12.22266
18.78516
7.910156
13.53516
9.222656
13.78516
29.78516
6.535156
340.875

Total distancia Intra-Cluster (WSS) = 571.7647 + 340.875 + 276.5
Total distancia Intra-Cluster (WSS) = 1189.14

Tabla 35: Distancia Inter-Cluster – Kmeans (Fuente: Elaboración Propia)

Cluster 1	101.3308
Cluster 2	3.48707
Cluster 3	37.28905
Total (BSS) =	**2579.991**

PAM

Tabla 36: Distancia Intra-Cluster – PAM (Fuente: Elaboración Propia)

Cluster 1	Cluster 2	Cluster 3
8.512485	14.71972	16.60302
3.857313	5.83737	17.29868
6.82283	14.54325	14.16824
20.30559	24.42561	6.689981
3.029727	18.07266	26.77694
14.61593	63.30796	6.342155
10.13317	27.54325	40.47259
10.65042	11.07266	12.2552
4.857313	19.07266	17.2552
1.650416	33.60208	8.037807
9.82283	43.36678	8.776938
6.788347	60.01384	7.342155
12.82283	32.6609	7.559546
5.098692	26.24913	16.12476
6.995244	103.955	4.646503
8.891795	57.30796	12.21172
19.09869	16.01384	9.559546
10.82283	**571.7647**	10.47259
7.995244		9.385633
17.20214		6.907372

30.34007
13.13317
13.99524
6.581451
10.54697
15.96076
8.098692
10.37455
7.133175
306.1379

1.603025
55.60302
35.03781
351.1304

Total distancia Intra-Cluster (WSS) = 306.1379 + 571.7647 + 351.1304
Total distancia Intra-Cluster (WSS) = 1229.033

Tabla 37: Distancia Inter-Cluster – PAM (Fuente: Elaboración Propia)

Cluster 1	2.522089
Cluster 2	101.3308
Cluster 3	32.36232
Total (BSS) =	**2540.097**