ESCUELA POLITÉCNICA NACIONAL

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE CIENCIAS

ESTUDIO DE LA TÉCNICA *SPECTRAL CLUSTERING* EN UN PROBLEMA DE PARTICIONAMIENTO DE GRAFO TIPO *K-WAY* Y LA APLICACIÓN DE LA MISMA EN LA REALINEACIÓN DE EQUIPOS ECUATORIANOS DEPORTIVOS CON RESTRICCIÓN DE CARDINALIDAD

TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERA MATEMÁTICA

EMILY LISSETTE REDROBÁN CORRALES

emily.redroban@epn.edu.ec

DIRECTOR: DIEGO FERNANDO RECALDE CALAHORRANO

diego.recalde@epn.edu.ec

DMQ, AGOSTO 2023

CERTIFICACIONES

Yo, EMILY LISSETTE REDROBÁN CORRALES, declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

EMILY LISSETTE REDROBÁN CORRALES

Certifico que el presente trabajo de integración curricular fue desarrollado por EMILY LISSETTE REDROBÁN CORRALES, bajo mi supervisión.

DIEGO FERNANDO RECALDE CALAHORRANO
DIRECTOR

DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el(los) producto(s) resultante(s) del mismo, es(son) público(s) y estará(n) a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

EMILY LISSETTE REDROBÁN CORRALES

DIEGO FERNANDO RECALDE CALAHORRANO

DEDICATORIA

Quiero dedicar este trabajo principalmente,
a mi madre Nancy, quien ha sido el
pilar fundamental en mi vida y mi
mayor motivación para salir adelante.
A mis abuelitos Víctor y Rosa, quienes
me han apoyado y animado a sobrellevar
cada problemática en la vida.
TODO ESTO ES POR USTEDES Y PARA USTEDES.

AGRADECIMIENTO

Antes que todo, agradezco a Dios por haberme permitido llegar hasta donde me encuentro, por la fuerza y la sabiduría que me ha dado en todo este trayecto.

Al Dr. Diego Recalde, mi tutor, por la paciencia, el apoyo y la confianza que me ha brindado en está etapa crucial de mi vida.

A mi familia, por la compresión e impulso que me han dado cada día para poder culminar mis estudios.

A mis amigos: Anthony, Fabian, Joyce, Bryan, Danny, David por ese apoyo incondicional que han tenido conmigo en cualquier momento. En particular a Erick y Daniel, quienes sin su aliento no habría logrado alcanzar las metas y el lugar en el que me encuentro ahora.

A mis compañeros de carrera y universidad quienes entre todos hemos podido salir adelante en los obstáculos que se nos han presentado.

RESUMEN

El propósito principal de este estudio se enfoca en analizar la técnica de agrupamiento conocida como *Spectral Clustering*, la cual ha adquirido relevancia en los últimos tiempos por sus valiosos resultados. Este trabajo comienza con una revisión general de conceptos fundamentales y teoría básica, para luego adentrarse en el agrupamiento espectral y las cadenas de Markov. Se introduce el algoritmo propuesto por Marina Meila [6], el cual se convierte en el enfoque central de este estudio. Además, se detalla una pequeña modificación realizada para su implementación, la cual se describirá más adelante. Posteriormente, se llevan a cabo pruebas computacionales en diferentes instancias aleatorias, presentando resultados gráficos que evidencian resultados interesantes acerca del método junto a un comparativo con el método *k-means*. Adicionalmente, se incorpora una tabla resumen que detalla los tiempos de ejecución del *Spectral Clustering* y la técnica tradicional *k-means*, un estadístico que mide la calidad de la agrupación y la función objetivo de encontrar el corte máximo. Se destaca cómo el uso del espectro de una matriz impacta de manera significativa en el tiempo de ejecución del método *k-means*.

Además de las pruebas en instancias aleatorias, el algoritmo se pone en práctica en una instancia real que comprende ciudades del Ecuador, las cuales participan como sedes en campeonatos de fútbol interprovinciales. Los resultados de esta aplicación se contrastan con los obtenidos en un artículo realizado por Recalde et al. en 2018 [3], donde se abordó el mismo caso. Finalmente, se exponen las conclusiones y recomendaciones que se derivan de esta investigación.

Palabras clave: Agrupamiento espectral, k-medias, *clúster*, valores propios, vectores propios, puntuación de silueta.

ABSTRACT

The main objective of this study focuses on the analysis of the clustering technique known as Spectral Clustering, which has great relevance in recent times due to its valuable results. It begins with a general review of basic concepts and fundamental theory to a specific study of spectral clustering and Markov chains. The algorithm presented by Marina Meila [6], which will be the primary focus of this work, is introduced, with a minor adjustment made for its execution, which will be explained later.

Subsequently, computational tests are conducted on different random instances, presenting graphical results that reveal interesting findings about the method Spectral Clustering, and a comparison to the k-means method. Furthermore, a summary table is included detailing the execution times of Spectral Clustering and the traditional k-means technique, a statistic measuring clustering quality, and the objective function of finding the maximum cut, highlighting how the spectrum of a matrix significantly affects the k-means' execution time.

In addition to the tests on random instances, the algorithm is applied to a real instance composed of cities in Ecuador participating in interprovincial football championships. The results of this application are compared with those obtained in a study conducted by Recalde et al. in 2018 [3], where the same case was addressed.

Finally, the conclusions and recommendations derived from this research work are presented.

Keywords: Spectral Clustering, k-means, clúster, eigenvalues, eigenvectors, silhouette score.

Índice general

Índice de figuras

Capítulo 1

Introducción

1.1. Prólogo

Un desafío central tanto en la vida cotidiana como en el ámbito profesional es la necesidad de agrupar elementos en conjuntos de datos. Consideremos como ejemplo una empresa que debe incorporar 50 nuevos empleados en cuatro áreas de trabajo distintas, cada uno cuenta con una formación profesional diferente. El gerente se encuentra con el desafío de entrevistar a todos en el menor tiempo posible para seleccionar a aquellos que mejor se ajusten a las necesidades de la empresa. Naturalmente surge la interrogante: ¿Cuál es la estrategia más efectiva para agrupar a los aspirantes? Aunque podrían ser entrevistados por disponibilidad de horario, cargo a contratar o incluso por orden de llegada, en este caso, es esencial agruparlos en cuatro categorías relacionadas con las áreas de trabajo. En general, se busca que los miembros de un grupo sean lo más similares posible entre sí, mientras que los grupos en sí sean lo más diferente.

Sin embargo, lograr este objetivo no es tan simple como parece. Variables como: el campo de estudio, la experiencia, las habilidades técnicas y otros factores deben ser considerados al realizar estas agrupaciones. A medida que aumenta la cantidad de variables y datos disponibles, elegir una forma de agrupar puede llegar a ser un paradigma bastante complejo.

Este ejemplo sencillo ilustra las deficiencias de las técnicas de agrupamiento tradicionales. Actualmente, las personas buscan clasificar sus datos en grupos que proporcionen información valiosa. Esta necesidad ha impulsado el desarrollo de técnicas de agrupamiento de datos, ampliamente utilizadas en diversas áreas como Biología, Estadística, Informática, e incluso en Investigaciones Sociales como la Psicología y el Comercio.

A partir de las investigaciones efectuadas por Donath y Hoffman (1973) [4] y Fiedler (1973) [5], surgió la técnica de agrupación de datos conocida como *Spectral Clustering*. La cual hace uso del análisis espectral [1] para lograr agrupamientos más precisos y efectivos sin importar la dimensionalidad de los datos. Con avances tecnológicos, esta técnica ha ganado popularidad y ha demostrado ser poderosa e importante en el análisis de datos.

Varios algoritmos [2] han sido desarrollados para implementar la técnica de agrupamiento espectral y mejorar el proceso. Estas investigaciones suelen tener como objetivo abordar desafíos y limitaciones de técnicas de agrupamiento, así como reducir la cantidad de operaciones para optimizar el tiempo de ejecución, entre otras.

Este trabajo de titulación se enfoca en estudiar el método de agrupamiento espectral y cómo los resultados varían en diferentes estructuras. Para lograrlo, se procedió a implementar el «Algoritmo 1» desarrollado por Marina Meila en 2015 [6], utilizando el lenguaje de programación Python Posteriormente, este algoritmo se aplicó a conjuntos de datos reales que están vinculados al estudio titulado “An exact approach for the balanced *k-way* partitioning problem with weight constraints and its application to sports team realignment”, realizado por Recalde, D. y colaboradores en 2018 [3]. Finalmente, se demuestra que los resultados obtenidos por un método de programación lineal entera coinciden con los resultados obtenidos en esta investigación al emplear la técnica de *Spectral Clustering*.

[1] El análisis espectral es el término empleado para el estudio del espacio propio (valores y vectores propios) de una matriz cuadrada

[2] Un algoritmo es un conjunto finito de instrucciones que hay que seguir para alcanzar un determinado objetivo

La estructura de este trabajo se fundamenta en presentar los hallazgos y conclusiones obtenidas. En este sentido, el presente documento está organizado de la siguiente manera:

- En el «**Capítulo 1**», se establecen las bases del estudio. Se introduce la problemática que motiva la investigación y se lleva a cabo una revisión de estudios similares. Además, se abordan aspectos preliminares y se profundiza en el tema a desarrollar.
- El «**Capítulo 2**» aborda el problema principal que se pretende resolver y detalla el método de solución empleado, el *Spectral Clustering*. Se describe detalladamente cómo se implementó este método y se presentan las pruebas computacionales realizadas para validar su eficacia y rendimiento.
- El «**Capítulo 3**» presenta los resultados obtenidos de los experimentos ejecutados en el capítulo anterior. Además, se exponen los resultados específicos del problema planteado.
- Finalmente, en el «**Capítulo 4**», se presentan las conclusiones derivadas de los resultados obtenidos. Se resumen los principales hallazgos y se discute su implicación en el contexto más general. De la misma forma, se brindan recomendaciones para futuras investigaciones que puedan expandir y profundizar en los temas abordados en este estudio.

1.2. Marco teórico

1.2.1. Estado del arte

El agrupamiento espectral es particularmente valioso para analizar situaciones con estructuras complejas. Se han desarrollado diferentes enfoques para abordar este desafío. En [16] y [19], se resumen algoritmos desarrollados por investigadores como Jianbo Shi y Jitendra Malik (2000), Ng, Jordan y Weiss (2002), y Marina Meila y Jianbo Shi (2001). Morales A. (2018) presenta una estructura general de estos algoritmos en su estudio [11], como se muestra en la Figura 1.1.

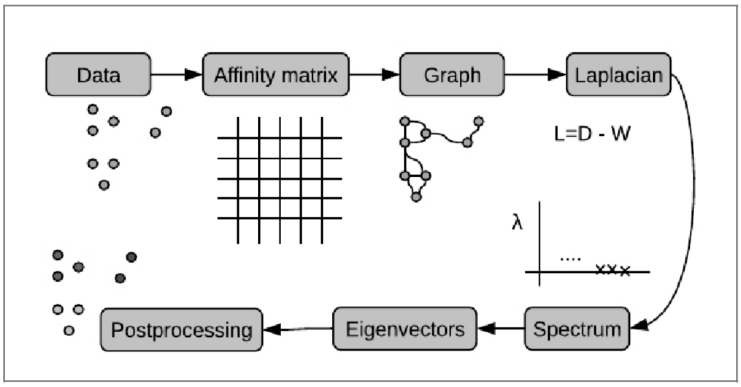

Figura 1.1: Proceso de agrupamiento mediante el análisis espectral. **Fuente:** [11]

La matriz Laplaciana [3] desempeña un papel crucial en el agrupamiento espectral al capturar la estructura de los datos y permitir la identificación de patrones de conectividad en el grafo. Adicionalmente, se calcula el análisis espectral para obtener los autovectores, que representan la estructura de los datos en un espacio de menor dimensión. Inicialmente, los algoritmos usaban la matriz laplaciana no normalizada, pero se descubrió que existe la posibilidad de que dicha matriz tenga entradas con valores altos de grados que podían dominar la matriz, afectando la visibilidad de los grupos. Para abordar esto, se desarrollaron algoritmos utilizando la matriz Laplaciana normalizada, como se describe en [19].

[3] La matriz Laplaciana es una matriz cuadrada denotada por $L = D - A$ donde D es una matriz diagonal que contiene los grados de los nodos del grafo y A es una matriz que representa las conexiones entre los mismos

La elección del número de grupos es un desafío en estos algoritmos. En [15], [20], y [17], se presentan métodos exitosos para este problema como el uso de la verosimilitud logarítmica de los datos, variedad de índices de selección de conglomerados y la heurística de la brecha propia [4]. Además, se menciona que [17] desarrolló un algoritmo que permite la selección automática de componentes enfocados en la creación de la matriz de similitud y la inclusión de parámetros para abordar este inconveniente.

Así mismo, se ha explorado la eficacia de las cadenas de Markov en la minería de datos [10], vinculándolas con el análisis espectral. Es relevante destacar que el *Spectral Clustering* en cadenas de Markov es el problema dual del *Spectral Clustering* tradicional, aunque se explorará más detalladamente en la siguiente sección.

En términos de aplicaciones prácticas, se han llevado a cabo estudios utilizando datos del mundo real. Por ejemplo, en el trabajo de Scot White y Padhraic Smyth [20], se aborda el caso de una red de coautoría de documentos de conferencias de NIPS (Sistemas de Procesamiento de Información Neuronal). El objetivo central era determinar el número óptimo de grupos a formar. Para lograrlo, se empleó una representación de la información a través de un grafo que ilustra las relaciones de coautoría entre pares de autores. Los resultados obtenidos fueron exitosos, ya que lograron clasificar a 1061 autores en 31 grupos representativos. Esta clasificación maximizó las conexiones entre los autores, lo que evidenció el éxito del enfoque.

En una línea similar, el mismo estudio de Scot White y Padhraic Smyth [20], se aplicó a la agrupación de equipos de fútbol americano universitario en el año 2000. Los resultados fueron eficientes en términos de la cantidad de grupos generados, aproximándose considerablemente a la solución óptima. A pesar de contar con pequeñas variaciones en los resultados, la metodología demostró su efectividad en este contexto.

Otra aplicación interesante se llevó a cabo en el sector de abastecimiento de agua, como se describe en el estudio de Herrera M., et al,[11]. En este caso, se abordó el desafío de la sectorización de una red de abastecimiento de agua. Mediante el uso del análisis espectral, se logró abordar de

[4] El objetivo de la heurística de la brecha propia es seleccionar un número entero k tal que k valores propios difieran significativamente del valor propio $k+1$.

manera efectiva el problema de sectorización que involucra 107 nodos de consumo y 134 tuberías. La metodología permitió dividir eficientemente el sistema en dos subsistemas hidráulicos, teniendo en cuenta la vulnerabilidad del mismo y la ubicación estratégica de los sensores. Este enfoque influyó significativamente en el desarrollo de metodologías multicriterio para problemas similares de sectorización.

En un estudio adicional de Herrera M. en [14], se replicó una investigación similar en el mismo contexto de abastecimiento de agua. Sin embargo, esta vez se abordó una instancia más amplia que involucraba la sectorización de 333 nodos de consumo y 479 tuberías en la ciudad de México. Los resultados obtenidos fueron sumamente alentadores y han posicionado esta técnica como una de las más ampliamente utilizadas en el campo del abastecimiento de agua. Además, este trabajo ha servido de base para investigaciones posteriores, como se puede ver en [7].

1.2.2. Preliminares

A continuación se presenta algunas definiciones que, en el caso de la Teoría de Grafos, difieren de las definiciones clásicas para adaptarse al contexto de este trabajo:

Definición 1: Un *grafo no dirigido* es un par ordenado denotado por $G = (V, E)$ cuyas componentes son: $V = \{v_1, \ldots, v_n\}$ el conjunto finito de nodos o vértices y $E = \{(v_i, v_j) : v_i, v_j \in V\}$ el conjunto de aristas que unen a los nodos.

Definición 2: Una *matriz de similitud* , denotada por $S = [S_{ij}]_{i,j=1}^{n}$, es una matriz cuadrada simétrica cuyas componentes representan una fuerza de vínculo o similitud existente entre el nodo i y el nodo j. Las similitudes son valores numéricos que satisfacen las condiciones de simetría $(S_{ij} = S_{ji})$ y no negatividad $(S_{ij} \geq 0)$

Definición 3: El *grado de un nodo* $i \in V$, notado por d_i, con V el conjunto de nodos; es un número real no negativo que corresponde a la suma de todas las aristas que inciden sobre el mismo, $d_i = \sum_{j \in V}^{n} S_{ij}$, donde S_{ij} refleja el peso de la similitud que existe entre el nodo i al nodo j.

Definición 4: Un *grafo de similitud* es la representación gráfica de las

relaciones de similitud entre los puntos de datos. En este tipo de representación gráfica, cada punto de datos es asignado a un nodo en el grafo, mientras que las similitudes entre los puntos se traducen en aristas que los conectan. Diversos tipos de grafos de similitud se pueden emplear, tales como el grafo de vecindad ϵ, el grafo de k-vecinos más cercanos y el grafo completamente conectado. Cada uno de estos grafos tiene su propia finalidad, que radica en plasmar las relaciones de cercanía local entre los distintos elementos que conforman el conjunto de datos.

Definición 5: Una *caminata aleatoria*, también conocida como *random walk*, es un proceso estocástico en el cual un sistema se desplaza de un nodo a otro de manera aleatoria. La evolución de los pasos en esta caminata es independiente de los eventos previos, lo que significa que cada movimiento no está influenciado por los pasos que lo precedieron.

Definición 6: Una *matriz de transición*, denotada por P, se define como una matriz cuadrada de dimensión n (siendo n el número de nodos en el grafo). En esta matriz, cada elemento representa la probabilidad de transición de moverse desde un nodo i hacia un nodo j. Esta probabilidad de transición se calcula, $p_{ij} = S_{ij}/d_i$.

Definición 7: Un *clúster* es un conjunto de datos que comparten similitudes entre sí (dentro del mismo grupo), pero se diferencian de los datos presentes en otros grupos.

Defincón 8: Una *partición de un conjunto de nodos* consiste en un número finito de subconjuntos del mismo, tal que son disjuntos dos a dos y su unión abarca la totalidad de los nodos.

Definición 9: El *aprendizaje no supervisado* se destaca como una técnica ideal para llevar a cabo análisis exploratorios de datos, debido a su enfoque particular. Esto se debe a que el conjunto de datos que se introduce carece de etiquetas o clasificaciones previas. En lugar de ello, los algoritmos extraen información relevante y patrones a partir de los datos proporcionados.

Definición 10: El *corte mínimo* se refiere a la separación de un grafo en un número finito de conjuntos disjuntos de vértices, tal que la suma de los pesos de las aristas entre los conjuntos sea la más pequeña posible.

1.2.3. Particionamiento *k-way*

El problema de particionamiento *k-way* es un problema combinatorio que implica la división de un grafo en k subconjuntos disjuntos. Según Laszewski [18], se plantea este problema definiendo un grafo no dirigido $G = (V, E, w)$, donde V representa el conjunto de nodos, E el conjunto de aristas, y $w : E \to \mathbb{R}$, una función de los pesos sobre las aristas. El objetivo del particionamiento es crear k subconjuntos de nodos, denotados como $P_1, \cdots, P_k$, de manera que la suma de las aristas entre los conjuntos sea mínima y los tamaños de los subconjuntos sean aproximadamente iguales. Estos subconjuntos se denominan «particiones», y las aristas con nodos terminales en conjuntos diferentes se conocen como «corte». Además, con la finalidad de este trabajo se definirá el problema dual del particionamiento *k-way*, donde se maximice el corte y se minimice la distancia intra-grupal.

Los problemas de optimización como este tienen varias aplicaciones en situaciones del mundo real. Por ejemplo, en el ámbito de los circuitos, según Bruno Menegola [12], los diseños suelen involucrar miles de transistores, y el uso del particionamiento ayuda a mitigar la complejidad, al ocasionar sistemas más manejables y reducir la necesidad de comunicación entre procesadores. Esto no solo optimiza la eficiencia del producto, sino que también disminuye los costos de materiales.

El particionamiento *k-way* es un problema de optimización que cae dentro de la categoría de *NP-hard*, lo que significa que es computacionalmente desafiante de resolver. Para abordar problemas *NP-hard*, a menudo se recurre a modelos de optimización de programación entera que pueden proporcionar soluciones exactas para instancias pequeñas. Sin embargo, en la práctica, los problemas suelen ser más grandes y complejos, lo que hace necesario el empleo de enfoques heurísticos[5], los cuales son capaces de entregar resultados satisfactorios en un tiempo razonable.

[5]Una solución heurística es un método de resolución basada en reglas prácticas, conocimientos expertos o técnicas de aprendizaje

1.2.4. *K-means*

Los problemas de agrupación resultan ser interesantes porque pertenecen al campo del aprendizaje no supervisado. En este contexto, los algoritmos desempeñan un papel fundamental al asignar etiquetas a objetos o patrones dentro de una base de datos sin contar con información previa sobre los mismos. Uno de los problemas más destacados y a la vez accesibles en este campo es el algoritmo *k-means*, Alcántara, L. (2019) nos menciona en [22] que es una técnica numérica e iterativa, además que fue desarrollada por Stuart Lloyd en 1957, aunque no fue mencionada públicamente hasta 10 años después por James McQueen.

Por otro lado Helm, M. (2021) en [8], presenta que el algoritmo *k-means* persigue la agrupación de un conjunto de datos en *clústeres*, donde las observaciones dentro de cada *clúster* son similares entre sí, maximizando la similitud *intra-clúster* y minimizando la similitud entre *clústeres*. Su simplicidad y eficiencia se evidencian en su único parámetro, el número de *clústeres* a formar k, lo que lo convierte en una herramienta asequible y fácil de usar para problemas de agrupación en diversas áreas de estudio.

Matemáticamente, *k-means* se enfoca en minimizar la suma de cuadrados dentro de cada *clúster*. Además, Yadav y Sharma (2013) en [21] nos indican que el algoritmo *k-means* se compone de dos fases clave:

- **Fase de Inicialización:** Se eligen aleatoriamente k puntos, que actuarán como centroides iniciales para el algoritmo.
- **Fase de Asignación:** Se calcula la distancia entre cada punto y los centroides, asignando cada punto al centro más cercano. Posteriormente, se recalcula el centro del nuevo grupo formado.

Dado que es un proceso recursivo, se requieren múltiples iteraciones para que el algoritmo converja de acuerdo con el criterio establecido. En ocasiones, el criterio de parada se relaciona con un número predefinido de iteraciones. A pesar de sus ventajas, el algoritmo *k-means* presenta algunas desventajas, tal como señala Martin Helm (2021) en su trabajo [8] sobre *k-means*:

Ventajas del *k-means*:

- Su teoría es fácil de comprender, lo que facilita la comprensión de su funcionamiento.
- Dado su nivel de complejidad moderado, puede implementarse en diversos lenguajes de programación.
- Solo requiere un hiperparámetro: el número de *clústeres* k.

Desventajas del *k-means*:

- Es necesario conocer previamente el número de *clústeres* a generar, lo que puede llevar a una formación de grupos sin considerar la estructura de los datos. Una solución es probar varios números de *clústeres* y seleccionar el mejor.
- Al ser un método heurístico, sirve para abordar problemas *NP-hard*, buscando soluciones óptimas locales.
- Su solución no es determinista, ya que inicia con centroides aleatorios, lo que puede dar lugar a resultados distintos en ejecuciones diferentes.
- La influencia de valores atípicos es notable debido al uso de la distancia euclidiana, lo que podría afectar la formación de los *clústeres*.

Es importante destacar que esta técnica de agrupación se basa en la proximidad de los puntos, lo que puede dar lugar a *clústeres* de tamaños diferentes. Además, *k-means* es el punto de partida para el desarrollo de otros métodos de agrupación como *k-means++*, *k-medianas*, *k-medoides* y *fuzzy-c-means*. En [9], Beltrán, J (2016) hace uso del método *k-means* y las variaciones para crear y extraer información de datos geoespaciales con la finalidad de crear conjuntos con similitudes, además de probar con información real y sintética.

1.2.5. *Silhouette score*

Además, en su trabajo [8], Helm, M. (2021) hace referencia al concepto de la puntuación de la silueta (*Silhouette Score*), que sirve como una métrica

para evaluar la calidad de los grupos formados por métodos de agrupación. El propósito de esta métrica es determinar la similitud dentro de cada grupo y contrastarla con la similitud presente en otros grupos.

El coeficiente de silueta es una medida que varía en un rango de -1 a 1. Bhardwaj, A. (2020), en su estudio [1], proporciona una interpretación de los diferentes valores que puede asumir esta medida. El valor del *Silhouette Score* varía entre -1 y 1, donde:

- Un valor cercano a +1 indica que los puntos están bien asignados a sus respectivos clústeres y están lejos de los clústeres vecinos. Esto se considera una buena separación entre clústeres.
- Un valor cercano a 0 indica que los puntos están cerca del límite entre dos clústeres vecinos o que podrían haber sido asignados incorrectamente.
- Un valor cercano a -1 indica que los puntos han sido asignados a clústeres incorrectos y estarían mejor si se asignaran al clúster vecino.

Esta medida se calcula mediante la fórmula $(b - a)/\max(a, b)$, donde a es la distancia promedio dentro de cada grupo, y b es la distancia promedio entre todos los grupos. Por otro lado, el coeficiente de silueta se utiliza para determinar el número óptimo de grupos a identificar. Es decir, se elige el número de grupos que resulte en el valor más alto para el Puntaje de Silueta, lo que garantiza una mejor agrupación de los datos.

En resumen, un *Silhouette Score* más alto sugiere una mejor separación entre los clústeres y, por lo tanto, una agrupación más coherente y significativa. Sin embargo, es importante recordar que el *Silhouette Score* debe usarse en conjunto con otras evaluaciones y no como única métrica, ya que su interpretación puede variar según la estructura de los datos y los objetivos del análisis.

1.2.6. *Spectral Clustering*

El agrupamiento espectral es uno de los enfoques más modernos y populares en los últimos años. Se basa en información de valores y vectores

propios de una matriz generada de las similitudes entre puntos, por lo que se le conoce como un método de agrupamiento basado en similitud. Su objetivo principal es lograr un particionamiento en el que los pesos entre grupos sean muy bajos, es decir, diferentes entre sí, mientras que los pesos dentro de un mismo grupo deben ser altos, lo que indica similitud.

Venjajas y desventajas del *Spectral Clustering*

Como se observa en la Figura 1.1, el *Spectral Clustering* requiere del uso del algoritmo *k-means* para encontrar su solución. Dado que incluye una técnica de agrupamiento como *k-means* en su algoritmo, las desventajas resultan ser semejantes. Sin embargo, ¿por qué el agrupamiento espectral puede ser superior a otras técnicas de agrupamiento? A pesar de compartir algunas desventajas, presenta importantes ventajas que pueden llevar a mejores resultados en comparación con los métodos tradicionales. Algunas de estas ventajas nos señala Chatterjee M., [2]:

- Capacidad para agrupar elementos que no necesariamente son vectores, permitiendo la utilización de medidas de similitud en lugar de medidas de distancia.
- No tiene suposiciones acerca de la estructura de los datos.
- Permite una reducción de dimensionalidad y ayuda la agrupación precisa del conjunto de datos.
- Facilidad de implementación en diversos lenguajes y resultados eficientes debido al aprovechamiento de herramientas de álgebra lineal.

Una desventaja inherente a esta técnica radica en su alto costo computacional cuando se aplica a conjuntos de datos con una gran cantidad de información. Este costo se origina en el cálculo de los valores propios y los correspondientes vectores propios, ya que la agrupación se lleva a cabo utilizando estos vectores.

Cadenas de Markov

Las cadenas de Markov han demostrado su eficacia en diversas aplicaciones de minería de datos. En [10], Stewart W and Liu, N. señalan que en la década de 1970, Stewart propuso un método de agrupamiento de estados a través del análisis espectral en una cadena de Markov. Este enfoque implica organizar los estados en grupos significativos mediante la utilización de los vectores propios dominantes de la matriz de transición. A diferencia de otros enfoques de agrupamiento espectral, este método considera tanto la distancia en el corte mínimo del grafo como la distancia de los estados al estado estacionario, lo que conduce a agrupamientos más precisos y relevantes.

Si una cadena de Markov es tanto irreducible [6] como aperiódica [7], su matriz de transición tendrá un único valor propio igual a 1, el cual representa la conectividad total del grafo.

Utilizar el *Spectral Clustering* en cadenas de Markov ofrece varias ventajas en comparación con otros enfoques, ya que una matriz de similitud se puede interpretar como un grafo no dirigido. Sin embargo, al convertir la información de dicha matriz en una matriz de transición, el grafo se vuelve dirigido y pierde su simetría. No obstante, los métodos de agrupamiento espectral pueden aplicarse a grafos dirigidos si se consideran como cadenas de Markov. En relación con otros enfoques, la información asimétrica puede generar problemas en su ejecución, lo que no ocasiona en este caso.

Como se señaló anteriormente, existen diferencias significativas al trabajar con distintos enfoques de *Spectral Clustering*. Por ejemplo, al realizar un análisis con la matriz Laplaciana o la matriz de transición, las soluciones pueden variar mínimamente y, al mismo tiempo, ser relevantes. Esto se debe a que ambos problemas son duales, la diferencia radica en los algoritmos, ya que al utilizar la matriz Laplaciana se centran en los valores propios cercanos a 0, mientras que con la matriz de transición se usan los k valores propios más cercanos a 1 junto con sus respectivos

[6] Una cadena es irreducible siempre existe la posibilidad de pasar de un estado a otro con cierta probabilidad.

[7] Una cadena es aperiódica cuando la longitud del ciclo más corto de volver al mismo estado es de 1.

vectores propios.

Los agrupamientos coinciden si los valores propios de la matriz de transición cercanos al círculo unitario son positivos. Si hay valores propios negativos más cercanos al círculo unitario, se recomienda trabajar con su modulo debido a su capacidad informativa para clasificar los grupos deseados, como lo han demostrado investigaciones previas [10].

Existen muchos algoritmos para calcular el particionamiento espectral, ya sea utilizando la matriz Laplaciana o la matriz de transición, como se menciona en [13]. Sin embargo, este trabajo se centra especialmente en el análisis espectral en cadenas de Markov, por lo que se empleará la matriz de transición en el algoritmo propuesto en [6] por Marina Meila, Jianbo Shi (2001) y Ng, Jordan, y Weiss (2002).

Algorithm 1 Algorithm *Spectral Clustering*

Entrada: Matriz de similitud **S**, número de *clúster* **k**

1. ***Transformar S***
   Calcular $d_i \longleftarrow \sum_{j=1}^{n} S_{ij}$ los grados de los nodos, para $i = 1, \cdots, n$
   Formar la matriz de transición con $P_{ij} \longleftarrow \frac{S_{ij}}{d_i}$, para $i, j = 1, \cdots, n$

2. ***Descomposición propia***
   Calcular los k valores propios más grandes $\lambda_1 \geq \cdots \geq \lambda_k$
   y vectores propios $v_1, \ldots, v_k$ de P

3. ***Unir los datos en el k-th subespacio principal***
   Sea $x_i = [v_{i2}\, v_{i3} \ldots v_{ik}] \in \mathbb{R}^{nx(k-1)}$, para $i = 1, \ldots, n$

4. ***k-means***
   Ejecute el algoritmo k-MEANS en la "data" $x_{1:n}$

Salida: El agrupamiento C obtenido en el paso 4.

Capítulo 2

Definición del Problema y Método de Solución

2.1. Planteamiento del problema

Los torneos internos de diversos deportes como el fútbol, el baloncesto y el vóley, son frecuentes en varios países. Su objetivo es determinar el equipo sobresaliente en la disciplina seleccionada. En estos torneos, cada equipo suele representar a una provincia y se procura que todos los equipos disputen la misma cantidad de partidos, o como máximo tengan una diferencia de dos encuentros. Para lograr este equilibrio, se forman grupos entre provincias cercanas. Esto no solo minimiza la distancia entre las sedes de los equipos en cada grupo, sino que también reduce los costos de transporte relacionados con los traslados de una provincia a otra por carretera. Es así, como el enfoque principal de este estudio es analizar el problema de particionamiento *k-way* balanceado en el agrupamiento de equipos en el campeonato de segunda categoría de la liga profesional de fútbol en el Ecuador.

En Enero del año 2014, los directivos de la Federación Ecuatoriana de Fútbol (FEF) solicitaron la ayuda de docentes de la Escuela Politécnica Nacional para desarrollar una solución óptima para la conformacion de grupos de equipos del campeonato de la segunda categoría de la liga de fútbol. En este campeonato participaron 21 provincias, que son las siguientes:

Región Litoral:

- Esmeraldas, Santo Domingo de los Tsáchilas, Manabí, Santa Elena, Los Ríos, Guayas, El Oro.

Región Sierra:

- Imbabura, Pichincha, Cotopaxi, Tungurahua, Bolívar, Chimborazo, Cañar, Azuay, Loja

Región Amazónica:

- Sucumbíos, Napo, Orellana, Pastaza, Morona Santiago.

En el marco de dicho campeonato, cada provincia participó con sus dos equipos más destacados. Para lograr una distribución justa, las 21 provincias se dividieron en 4 grupos zonales equilibrados, de modo que cada grupo tuviera igual cantidad de provincias o, en su defecto, una diferencia máxima de una provincia. Los partidos en cada zona, se jugaron en el formato de Torneo Double Round Robin, también conocido como "todos contra todos' ' en dos rondas, en el que cada equipo se enfrentó dos veces a todos los demás equipos de su grupo zonal. Los detalles de este campeonato se pueden encontrar en [3]

Es importante mencionar que tanto el estudio previo [3] como este trabajo se centraron exclusivamente en la división óptima de las provincias para la conformación de los 4 grupos zonales.

Para lograr esto, se define un grafo no dirigido $G = (V, E)$, n el que la distancia entre la ciudad A y la ciudad B es igual en ambas direcciones. Este grafo está compuesto por:

- V representa los nodos del grafo que en este caso serán las capitales de cada provincia que participa en el campeonato.
- E son las aristas que conectan cada provincia con las demás, y en este caso, se considera la distancia en carretera existente entre las dos provincias medida en kilómetros.

Las distancias empleadas se extrajeron de Google Maps, considerando las vías disponibles.

La función objetivo tiene por finalidad calcular la suma total de las distancias por carretera en kilómetros, entre las capitalesprovinciales de cada grupo zonal, la cual queremos que sea minimizada. Es relevante señalar que, al disponer de las distancias entre las ciudades, se empleará una matriz de similitud. Dicha matriz está formada por las conexiones en el grafo, lo que da como resultado una matriz simétrica y positiva.

2.2. Método de solución

Para resolver el problema de agrupamiento en un grafo, existen varios métodos disponibles, como la modelización de un problema de programación entera, métodos tradicionales de agrupamiento como el algoritmo *k-means*, el algoritmo DBSCAN e incluso métodos de agrupamiento jerárquico como el algoritmo de Louvain. Sin embargo, uno de los métodos que ha ganado importancia recientemente es el *Spectral Clustering* que será el método usado para resolver el problema planteado.

El algoritmo empleado en este estudio ha sido previamente definido en la Sección 1; dicho algoritmo requiere únicamente la matriz de similitud como entrada, la cual, en nuestro contexto es la matriz de distancias entre las ciudades, y el número de *clústeres* a obtener. No obstante, un inconveniente que se presenta en el estudio es que, si seguimos el algoritmo conforme a su planteamiento original, la solución alcanzada no persigue el objetivo deseado, ya que estaríamos agrupando nodos con una mayor medida de similitud, que corresponden a ciudades con mayor distancias entre sí, por lo tanto, para este trabajo se realizó un pequeño cambio antes de ejecutar el algoritmo. Este cambio consistió en trabajar con una matriz en la que sus celdas están formadas por la inversa del valor de similitud, en otras palabras, se creó una nueva matriz de tal modo que su estructura esté dada por, $S_{nueva} = 1/S_{i,j}$ donde $S_{i,j}$ es la matriz de similitud original; de esta manera una menor distancia implicará un mayor valor de similitud y el algoritmo tenderá a agruparlos. Por otro lado, es importante aclarar que el método *Spectral Clustering* es un método de aproximación y no hay garantía de que se encuentre la solución óptima

del problema, antes de aplicar dicho algoritmo en nuestra instancia se realizará pruebas computacionales para ver el funcionamiento y eficiencia del *Spectral Clustering*.

2.2.1. Instancias a prueba

Para la experimentación computacional se considera gráficos de dispersión con diferente cantidad de puntos y gráficos con 1, 2 y 3 círculos, los cuales presentan estructuras complejas. A continuación, se detallará con más precisión cada instancia a prueba.

- **Tipo de gráfico:** Gráfico de dispersión
  - **Instancia 1:** 25 puntos
  - **Instancia 2:** 100 puntos
  - **Instancia 3:** 200 puntos
- **Tipo de gráfico:** 1 círculo
  - **Instancia 4:** 1000 puntos
  - **Instancia 5:** 200 puntos
- **Tipo de gráfico:** 2 círculos
  - **Instancia 6:** 1400 puntos en total
    - *Círculo pequeño:* 300 puntos.
    - *Círculo mediano:* 800 puntos.
    - *Ruido:* 300 puntos.
  - **Instancia 7:** 800 puntos en total
    - *Círculo pequeño:* 200 puntos.
    - *Círculo mediano:* 400 puntos.
    - *Ruido:* 200 puntos.
  - **Tipo de gráfico:** 3 círculos
    - **Instancia 8:** 1400 puntos en total.
      - *Círculo pequeño:* 200 puntos.
      - *Círculo mediano:* 400 puntos.

        - ◇ *Círculo grande:* 600 puntos.
        - ◇ *Ruido:* 200 puntos.
    - **Instancia 9:** 900 puntos en total
        - ◇ *Círculo pequeño:* 200 puntos.
        - ◇ *Círculo mediano:* 200 puntos.
        - ◇ *Círculo grande:* 200 puntos.
        - ◇ *Ruido:* 300 puntos.

En cada instancia se clasificó en diferentes cantidades de *clústeres*.

2.2.2. Pruebas computacionales

Se llevaron a cabo pruebas computacionales con el propósito de medir el tiempo de ejecución tanto del algoritmo de *Spectral Clustering* como del algoritmo de *k-means*, adicionalmente se utiliza el *Silhouette Score* como medida de evaluación de la calidad de la agrupación, además, de hacer uso de la función objetivo, la cual está definida por la suma total de los valores de similitud entre cada par de puntos que forman parte del *clúster*. Los tiempos de generación de los métodos en segundos se expondrán en la siguiente sección, los cuales son el resultado de promediar 5 ejecuciones del experimento. Adicionalmente, se muestran los resultados gráficos de los *clústeres* obtenidos mediante ambos métodos, lo que permite analizar y comparar sus resultados.

Dado que algunas de las soluciones de las instancias comparten similitudes y se diferencian únicamente en la cantidad de puntos, se presentará los resultados gráficos pertinentes, mientras que las restantes se los podrá encontrar en la parte de anexos del estudio.

Gráfico de dispersión con puntos aleatorios

- **INSTANCIA 1**

  Esta instancia contiene 25 puntos aleatorios con el objetivo de evaluar los método de solución en en una instancia pequeña, similar a nuestra instancia principal del campeonato de fútbol; que consta de 21 nodos.

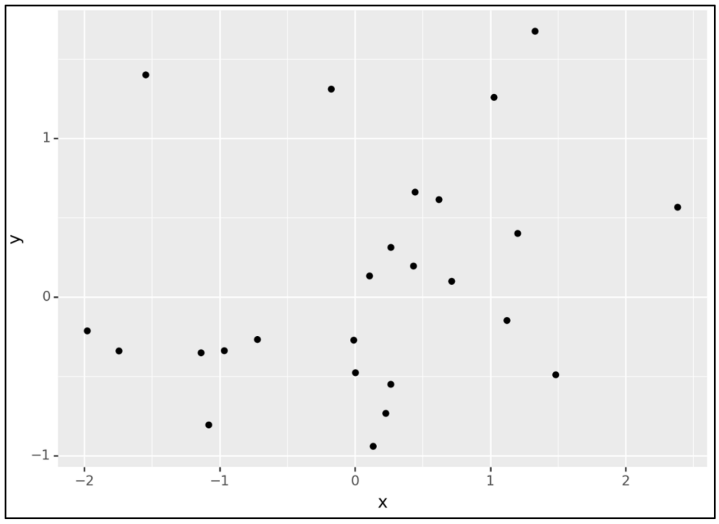

Figura 2.1: Instancia 1

- **Número de *clústeres*: 2**

  Silhouette Score: 0,40510 (SC), 0,41606 (KM).

  Función Objetivo: 494,67 (SC), 422,98 (KM).

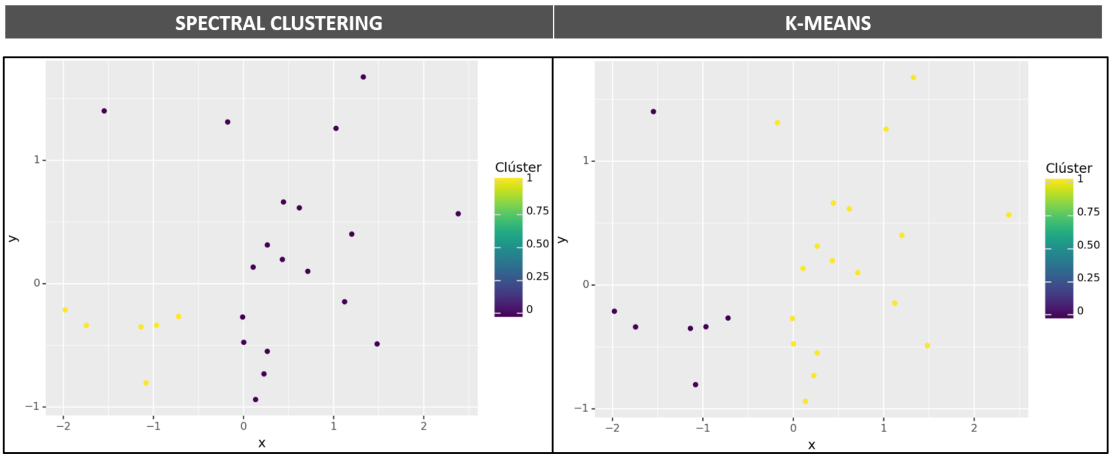

Figura 2.2: Instancia 1, 2 *clústers*

Notemos que la diferencia en la medida de la puntuación *Silhouette* entre el método de *Spectral Clustering* y el algoritmo *k-means* es relativamente pequeña y no es significativa. Los valores correspondientes no están ubicados en proximidad inmediata a 0, lo que nos lleva a concluir que, las agrupaciones están relativamente bien diferenciadas. En otras palabras, sus resultados presentan una partición eficiente en ambos casos, con la diferencia en un único punto, el cual está localizado en la región

superior izquierda. Es importante notar que al emplear la técnica de k se recorre una distancia total menor de 422.98, a comparación con el método espectral.

- **Número de *clústeres*: 4**

  Silhouette Score: 0,32942 (SC), 0,35243 (KM).

  Función Objetivo: 142,61 (SC), 146,45 (KM).

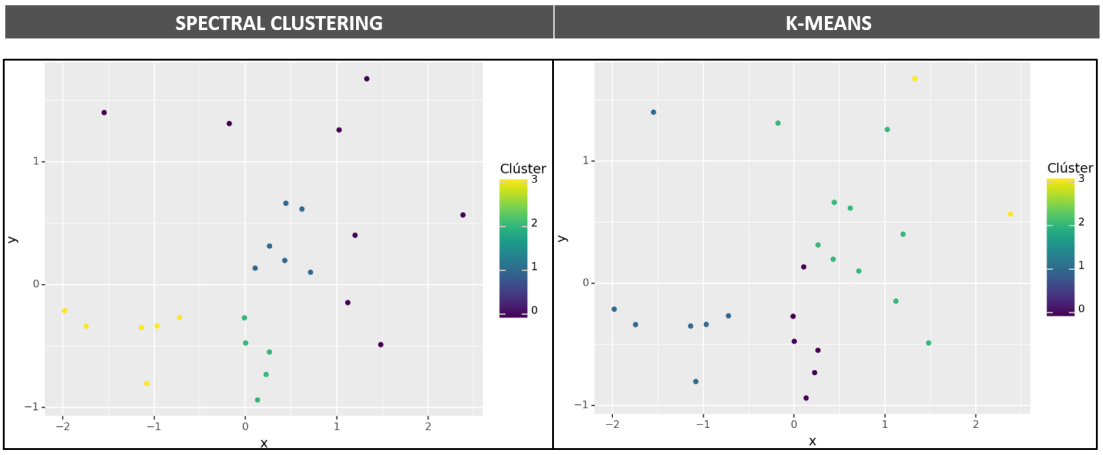

Figura 2.3: Instancia 1, 4 *clústers*

Por otro lado, al incrementar el número de *clústers* a crear, se hace evidente un aumento en la diferencia del coeficiente *Silhouette* en ambos métodos. Aunque los valores del *Silhouette Score* no se aproximan ni a 0 ni a 1, esto sugiere que los datos están relativamente bien agrupados, con una pequeña probabilidad de que algunos puntos sean asignados a *clústers* a los que no pertenecen. Ahora, se analizará estos resultados en conjunto con la función objetivo, la cual muestra una mejora en la técnica de agrupamiento espectral, reflejada en un valor de distancia menor a 142.61. Obteniendo una mejor solución para este caso.

- **Número de *clústeres*: 10**

  Silhouette Score: 0,34557 (SC), 0,40106 (KM).

  Función Objetivo: 21,20 (SC), 28,20 (KM).

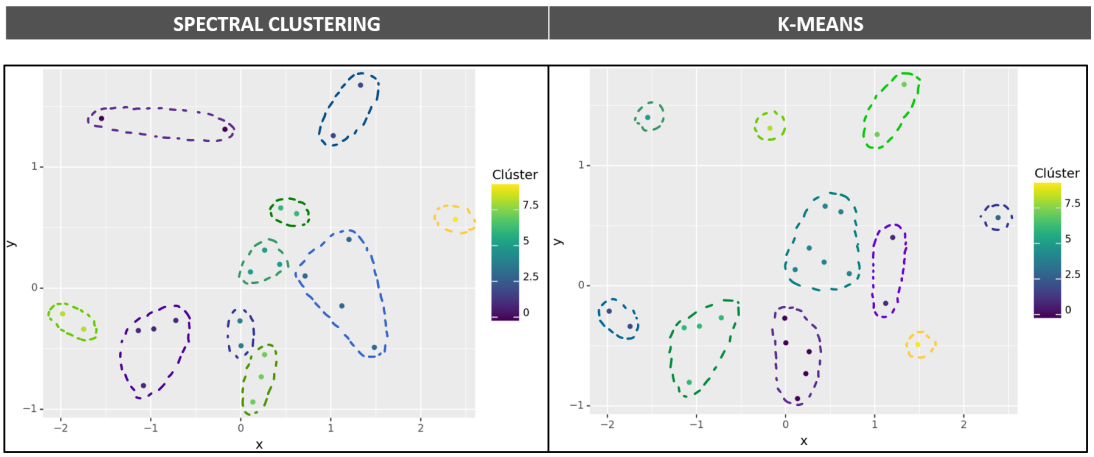

Figura 2.4: Instancia 1, 10 *clústers*

Se decidió fijar el número de *clústers* en 10 con el propósito de explorar cómo los algoritmos se desempeñan al trabajar con un número considerable de *clústers*. La medida *Silhouette* para el método de *Spectral Clustering* es de 0,3458, indicándonos una agrupación moderada, es decir no hay agrupaciones perfectamente definidas, sino más bien puntos que están cercanos a los límites de su*clúster*. Mientras que para el algoritmo *k-means* es más alta, con un valor de 0,4011, lo que indica que proporciona una solución superior en comparación con el otro método de agrupación., con grupos que se están mejor definidos. Sin embargo, al considerar la distancia total recorrida entre todos los puntos, el método de *Spectral Clustering* arroja un resultado mejor en la función objetivo.

Como se mencionó los resultados de los siguientes 2 experimentos se podrá encontrar en la parte de anexos y lo único que se presentará son los valores obtenido en distancia total y la medida de calidad.

- **INSTANCIA 2**

  Se realizó una ejecución con 100 puntos aleatorios.

  - **Número de *clústeres*: 2**

    Silhouette Score: 0,36902 (SC), 0,37236 (KM).

    Función Objetivo: 7827,80 (SC), 7871,26 (KM).

- **Número de *clústeres*: 3**

  Silhouette Score: 0,35302 (SC), 0,36050 (KM).

  Función Objetivo: 2572,95 (SC), 2471,86 (KM).

- **Número de *clústeres*: 4**

  Silhouette Score: 0,37370 (SC), 0,37593 (KM).

  Función Objetivo: 4138,77 (SC), 4105,36 (KM).

En estos casos, es notable que para 2, 3 y 4 grupos, los resultados presentan un comportamiento similar. Esto se debe a que los valores de *Silhouette Score* están bastante cercanos entre sí. Estos hallazgos indican que tanto *Spectral Clustering* como *k-means* están generando particionamientos coherentes. Los valores de *Silhouette score*, situados en un rango que no contiene valores ni muy altos, ni muy bajos de 0.35 a 0.37, revelan un equilibrio entre las agrupaciones que señala que los *clústeres* no están perfectamente separados. Algunos puntos parecen hallarse en los límites de grupos, lo que sugiere la posibilidad de que algunos de ellos estén mal ubicados en términos de asignación de agrupaciones. De manera similar, sucede con las distancias totales recorridas por los puntos: en todos los casos, la técnica de *k-means* resulta en distancias menores.

- **INSTANCIA 3**

  Se realizó una ejecución con 200 puntos aleatorios.

  - **Número de *clústeres*: 2**

    Silhouette Score: 0,32203 (SC), 0,34197 (KM).

    Función Objetivo: 8658,04 (SC), 7966,50 (KM).

  - **Número de *clústeres*: 3**

    Silhouette Score: 0,36107 (SC), 0,36346 (KM).

    Función Objetivo: 4204,46 (SC), 4331,83 (KM).

  - **Número de *clústeres*: 6**

    Silhouette Score: 0,33829 (SC), 0,38196 (KM).

    Función Objetivo: 1803,00 (SC), 1417,25 (KM).

  En el caso de 2 *clúster*, los valores de *Silhouette score* son relativamente bajos, especialmente para el método *Spectral Clustering*. Esto

sugiere que los grupos no están claramente definidos y podría existir la posibilidad de que algunos puntos en los límites estén asignados incorrectamente a grupos. Al comparar esta medida con nuestra función objetivo, notamos que el método tradicional *k-means* obtiene un mejor resultado en términos de calidad de agrupación.

En la instancia con 3 grupos, los valores de *Silhouette score* para ambos métodos se acercan a 0.36. Estos valores son superiores en comparación con la instancia de 2 grupos, lo que indica una asignación de grupos un poco más precisa. Sin embargo, aún presenta problemas en los límites entre los grupos. Al combinar esta medida con nuestra función objetivo, notamos que el *Spectral Clustering* presenta una solución mejor, ya que resulta en una función objetivo de 4204.46, que es menor que la obtenida por *k-means*.

En esta última instancia con 6 grupos, observamos una diferencia más pronunciada entre los valores de *Silhouette score*. El método tradicional *k-means* produce un valor más alto en comparación con el método estudiado en este trabajo. Concluyendo que *k-means* proporcionaría una solución con grupos más claramente separados, que además se puede corroborar con la función objetivo.

Gráfico de 1 círculo

- **INSTANCIA 4**

  Esta instancia contiene 1000 puntos aleatorios con la forma de un círculo.

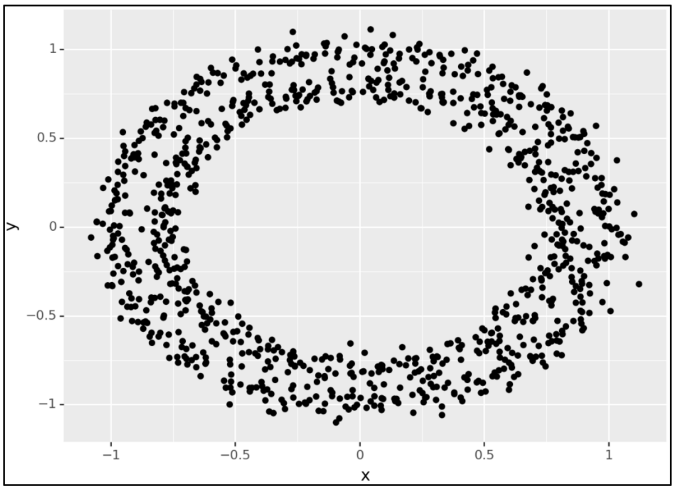

Figura 2.5: Instancia 4, Inicial

- **Número de *clústeres*: 2**

  Silhouette Score: 0,39868 (SC), 0,38233 (KM).

  Función Objetivo: 427359,09 (SC), 426426,61 (KM).

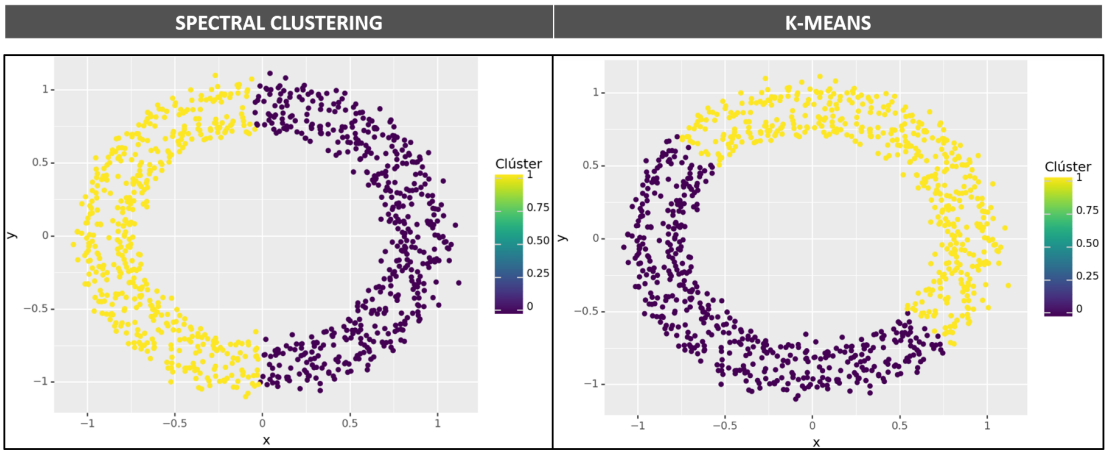

Figura 2.6: Instancia 4, 2 *clústers*

Es evidente que ambos valores de *Silhouette score* se sitúan en un rango considerado alto.Esto refleja una asignación correcta de puntos a sus grupos correspondientes, aunque en áreas limitadas pueden surgir problemas en los bordes de los *clústers*. Además, es importante observar que la media de la silueta del método *Spectral Clustering* es ligeramente superior al del *k-means*. Sin embargo, al considerar la función objetivo, resulta claro que el método convencional de *k-means* ofrece una solu-

ción más satisfactoria.

- **Número de *clústeres*: 3**

  Silhouette Score: 0,43873 (SC), 0,43932 (KM).

  Función Objetivo: 207007,68 (SC), 207376,57 (KM).

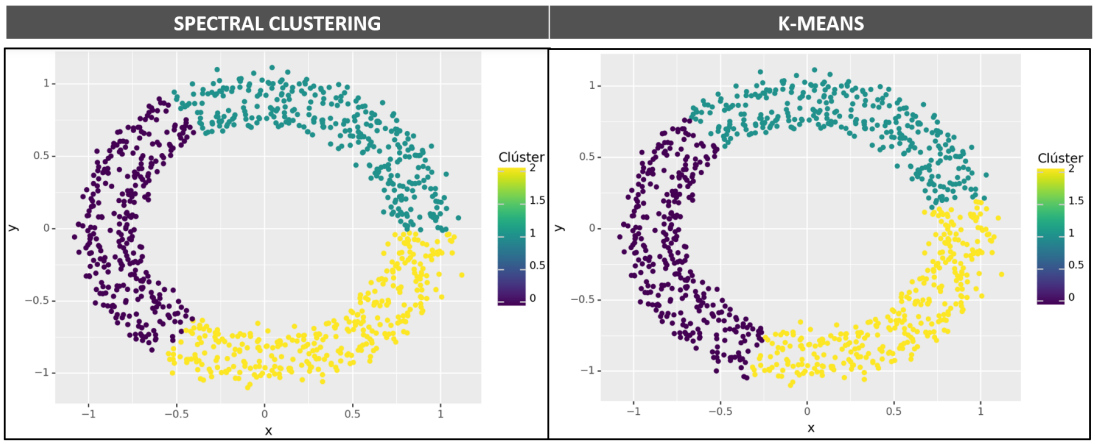

Figura 2.7: Instancia 4, 3 *clústers*

El *Silhouette score* muestra valores muy próximos entre los dos métodos, situándose alrededor de 0.43. Esto indica que los grupos están bien diferenciados entre sí. Aunque el *Silhouette score* es ligeramente superior en el método *k-means*, al considerar la función objetivo observamos que el enfoque de agrupamiento espectral ofrece una menor distancia total entre los grupos, lo que significa una mejora en la función objetivo.

- **Número de *clústeres*: 6**

  Silhouette Score: 0,43583 (SC), 0,44330 (KM).

  Función Objetivo: 59894,78 (SC), 58980,56 (KM).

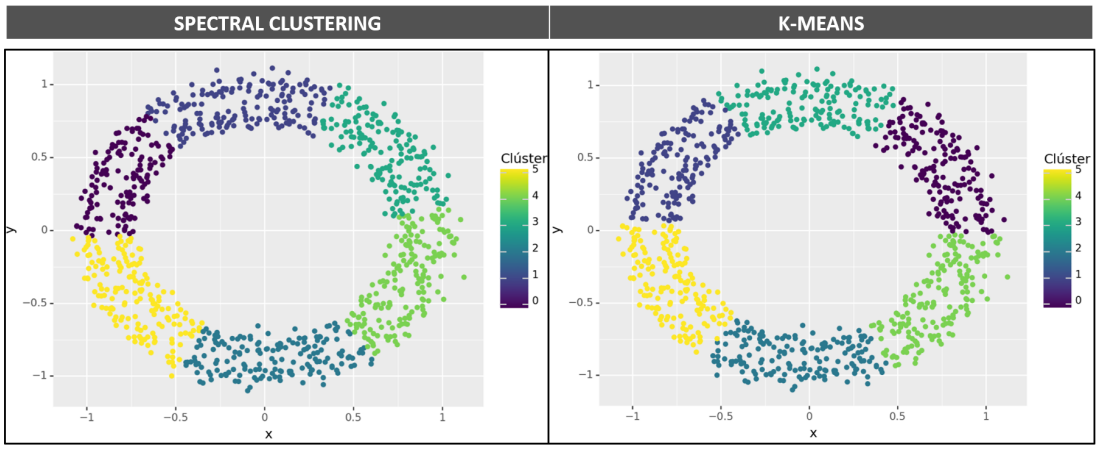

Figura 2.8: Instancia 4, 6 *clústers*

El patrón observado en el *Silhouette score* es análogo al caso previo: sus valores se mantienen en un rango considerablemente alto, indicando agrupaciones bien definidas. Sin embargo, en contraste con la función objetivo, en esta ocasión la técnica *k-means* demuestra ser más eficaz al ofrecer resultados más eficientes.

- **INSTANCIA 5**

  El experimento computacional de esta instancia se realizo con 200 puntos para obtener la estructura formada.

  Esta instancia presenta resultados similares a la instancia 4 con diferencia de tiempo y número de puntos. La representación gráfica se encuentra adjuntada en anexos.

  - **Número de *clústeres*: 2**

    Silhouette Score: 0,37580 (SC), 0,38041 (KM).

    Función Objetivo: 17182,68 (SC), 17124,07 (KM).

  - **Número de *clústeres*: 3**

    Silhouette Score: 0,42746 (SC), 0,43824 (KM).

    Función Objetivo: 8502,48 (SC), 8319,54 (KM).

  - **Número de *clústeres*: 6**

    Silhouette Score: 0,43481 (SC), 0,43568 (KM).

    Función Objetivo: 2385,04 (SC), 2381,76 (KM).

En este análisis, es evidente que el valor más bajo del coeficiente de *Silhouette score* se observa en el caso de 2 *clústeres*. Sin embargo, estos valores aún se consideran relativamente altos, lo que sugiere agrupaciones bien definidas con muy pocos puntos posiblemente mal asignados en los límites de los *clústeres*. Para los casos de 3 y 6 clústeres, los valores de *Silhouette score* tienen la misma interpretación debido a su similitud.

En cuanto a la función objetivo, es notable que en todos los casos, la solución más eficaz es proporcionada por *k-means* en relación con su coeficiente de *Silhouette score*. Esto se comprueba al considerar las distancias recorridas, que son considerablemente menores en comparación con las obtenidas mediante el *Spectral Clustering*

Gráfico de 2 círculo

- **INSTANCIA 6**

  Instancia con 2 círculos: un círculo pequeño con 300 puntos y un radio de 10, un círculo grande con 800 puntos y un radio de 50. Además, un ruido con 300 puntos. Trabajando con 1400 puntos en total.

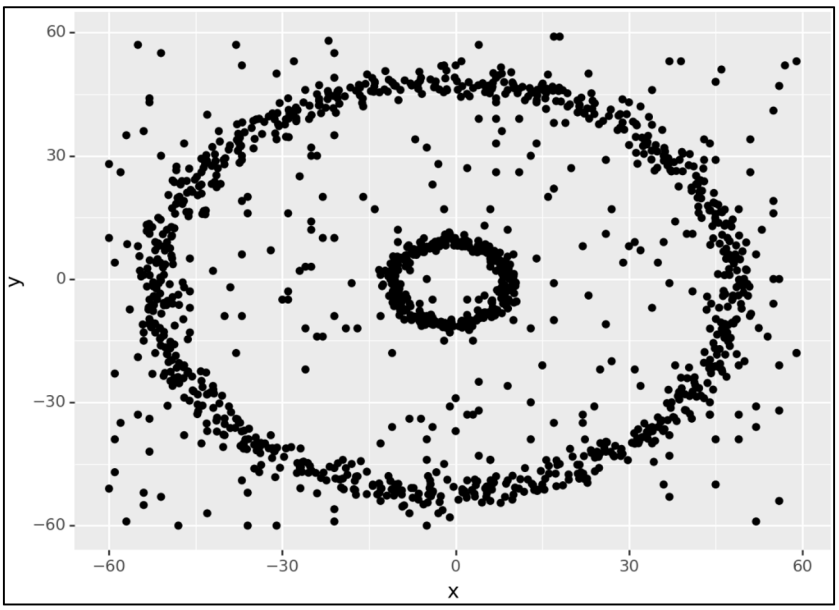

Figura 2.9: Instancia 6, Inicial

- **Número de *clústeres*: 2**

  Silhouette Score: 0,16607 (SC), 0,31818 (KM).

  Función Objetivo: 75025077,55 (SC), 47278691,62 (KM).

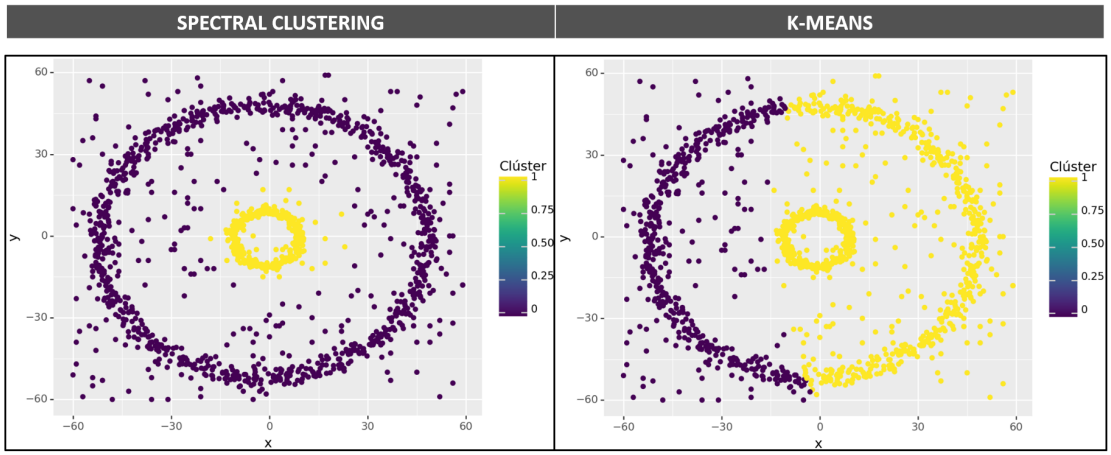

Figura 2.10: Instancia 6, 2 *clústers*

Al examinar el *Silhouette Score* de los dos grupos, se destaca una diferencia notable entre los resultados. Un valor cercano a 0, como el obtenido a través del método de *Spectral Clustering*, indica que las agrupaciones están mal definidas. Esta diferencia podría deberse a la posible asignación incorrecta de puntos que se encuentran en los límites, generando incertidumbre sobre a qué *clúster* pertenecen.

Mientras que, al mantener un enfoque de particionamiento más tradicional, se conserva un valor de *Silhouette Score* donde muy pocos puntos están incorrectamente asignados. La presencia de este inconveniente sugiere que la solución proporcionada por la función objetivo es más efectiva al optar por el método tradicional.

- **Número de *clústeres*: 3**

  Silhouette Score: 0,34468 (SC), 0,35557 (KM).

  Función Objetivo: 28381587,19 (SC), 25797541,37 (KM).

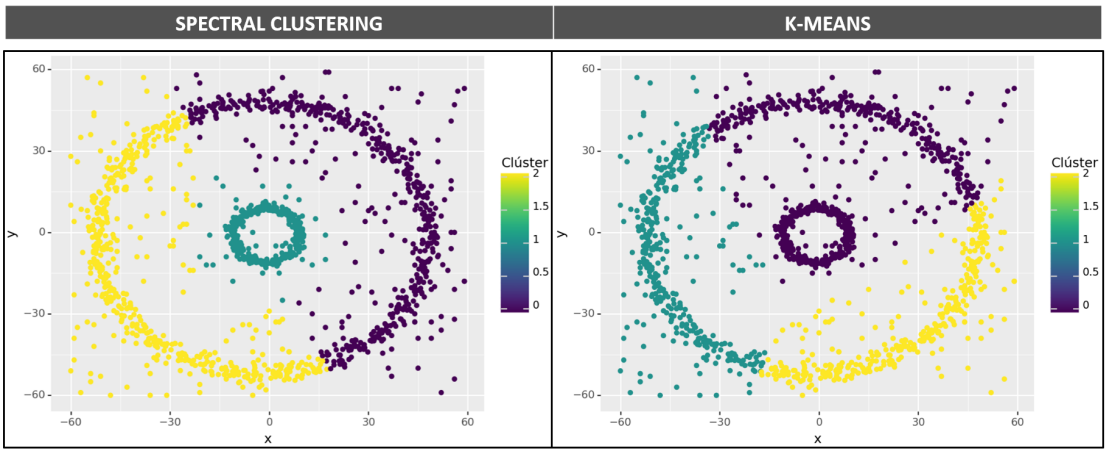

Figura 2.11: Instancia 6, 3 *clústers*

Al aumentar el número de grupos, se puede observar una mejora en la calidad de las agrupaciones. Los valores del *Silhouette Score* se acercan notablemente entre sí. Aunque estos valores no son muy altos, nos indica que son asignaciones que presenta coherencia en su clasificación. Sin embargo, en los puntos limites no se garantiza que se encuentren bien definidos. Nuevamente, al considerar la función objetivo, se destaca que el método de *k-means* proporciona valores menores, lo cual es lo que representa nuestra función objetivo.

- **Número de *clústeres*: 6**

  Silhouette Score: 0,40141 (SC), 0,49359 (KM).

  Función Objetivo: 8367594,09 (SC), 6756637,429 (KM).

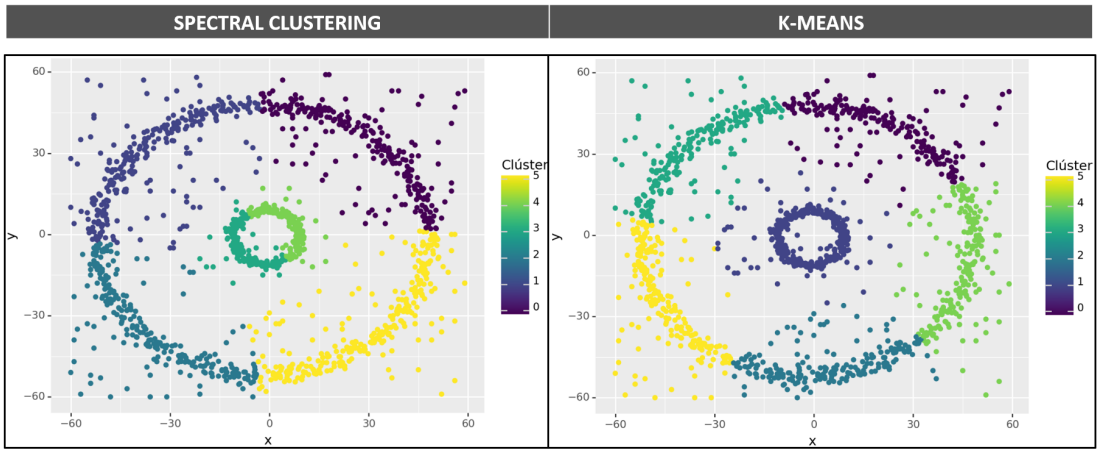

Figura 2.12: Instancia 6, 3 *clústers*

Se reafirma la tendencia de mejora en la calidad de las agrupaciones al aumentar el número de *clústeres*. Al analizar el caso de 6 *clústeres*, se observa una mayor eficiencia en los resultados. No obstante, es importante señalar que el valor del *Silhouette Score* obtenido mediante la técnica *k-means* es significativamente superior. Esto sugiere una separación más definida entre los grupos, lo cual se refleja en la función objetivo que presenta una distancia mucho menor en comparación con el resultado obtenido mediante el *Spectral Clustering*.

- **INSTANCIA 7**

  Instancia con 2 círculos: un círculo pequeño con 200 puntos y radio de 25, un círculo grande con 400 puntos y radio de 50 y un ruido con 200 puntos. Trabajando con 800 puntos en total.

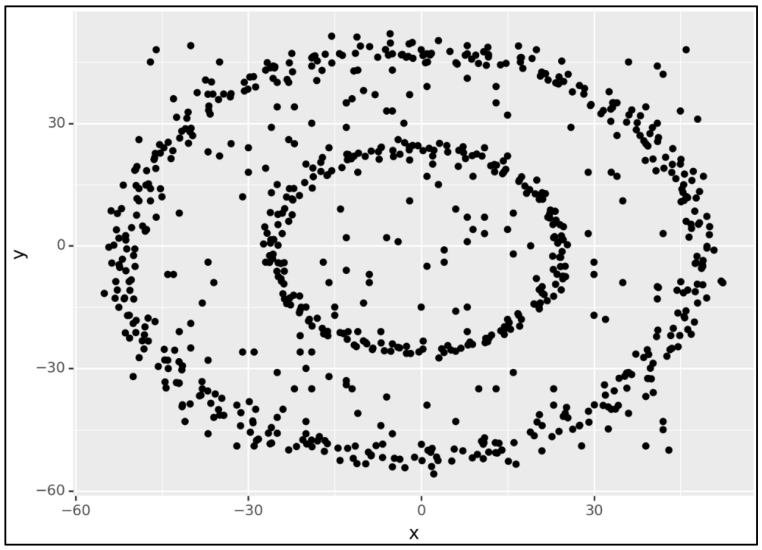

Figura 2.13: Instancia 7, 1 *clúster*

- **Número de *clústeres*: 2**

  Silhouette Score: 0,33606 (SC), 0,33598 (KM).

  Función Objetivo: 13355753,87 (SC), 13334219,30 (KM).

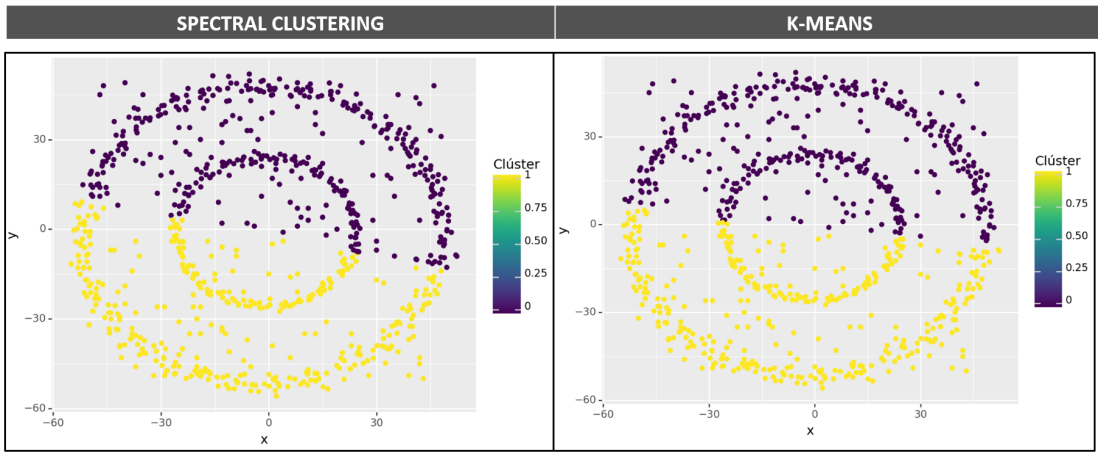

Figura 2.14: Instancia 7, 2 *clústers*

Cuando se reduce la cantidad de puntos y se modifica las dimensiones de los radios, los resultados tienden a cambiar de manera significativa en comparación con lo ocurrido en el caso anterior. Estos gráficos presentan una proximidad tanto en la representación gráfica como en los valores de *Silhouette Score* y la distancia recorrida. Al trabajar con un gran volumen de datos, incluso si se modifica un solo punto, se producen cambios notables en las distancias, como se puede apreciar en los resultados de distancia recorrida. Al presentar similitudes podemos notar que la medida de la silueta se encuentra próxima entre ambos métodos con una ligera variación en el *Spectral Clustering*; sin embargo, los dos métodos nos presentan el problema de que los puntos de los límites podrían haberse asignado incorrectamente. Con lo que respecta a la función objetivo, el *k-means* presenta resultados mejores.

- **Número de *clústeres*: 3**

  Silhouette Score: 0,36728 (SC), 0,36751 (KM).

  Función Objetivo: 6898351,13 (SC), 6892408,64 (KM).

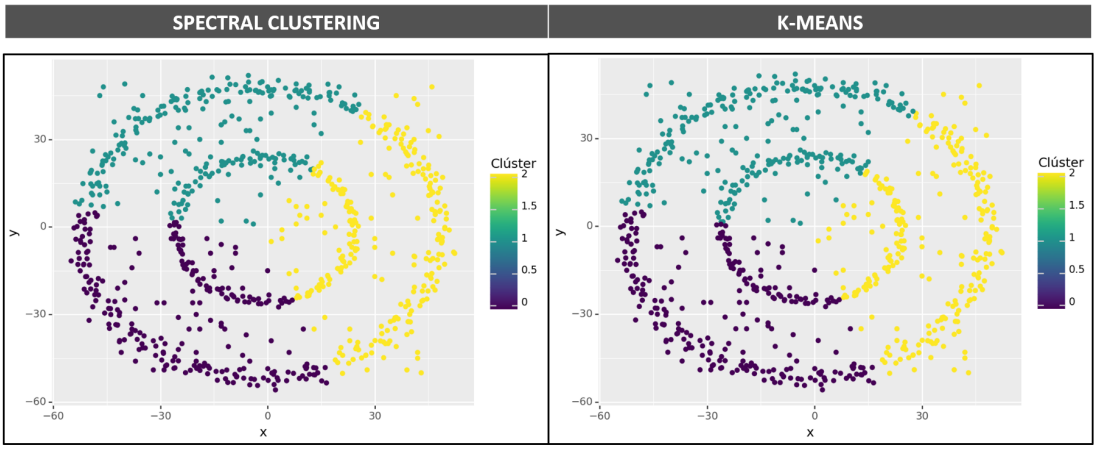

Figura 2.15: Instancia 7, 3 *clústers*

- **Número de *clústeres*: 6**

  Silhouette Score: 0,36225 (SC), 0,36223 (KM).

  Función Objetivo: 2533120,72 (SC), 2523062,67 (KM).

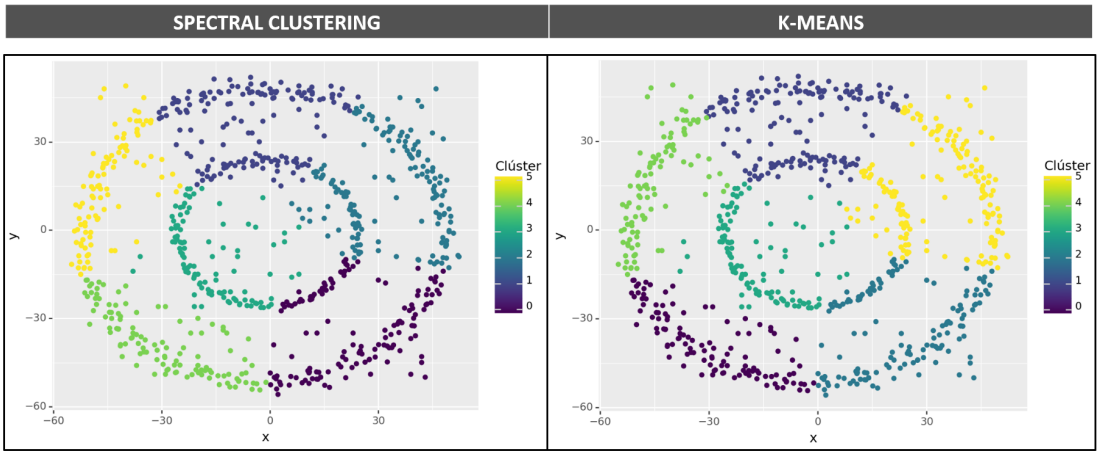

Figura 2.16: Instancia 7, 6 *clústers*

En los 2 últimos casos, como son de 2 y 3 grupos, se observa un patrón similar al caso anterior, con una cercanía en los valores medios de la puntuación de calidad (*Silhouette Score*) indicando que a pesar los grupos se encuentren bien separados entre sí, los problemas en los puntos que se encuentran en los límites sigue presente. En la función objetivo, la técnica tradicional nos brinda resultados más eficientes.

Gráfico de 3 círculo

La complejidad de los gráficos permiten ver la diferencias del algoritmo estudiado en este trabajo, por lo que se realizó experimentos con 3 círculos presentados a continuación.

- **INSTANCIA 8**

  Instancia con 3 círculos: un círculo pequeño con 200 puntos y un radio de 10, un círculo mediano con 400 puntos y un radio de 50 y un círculo grande con 600 puntos y un radio de 100 y un ruido con 300 puntos. Trabajando con 1500 puntos en total.

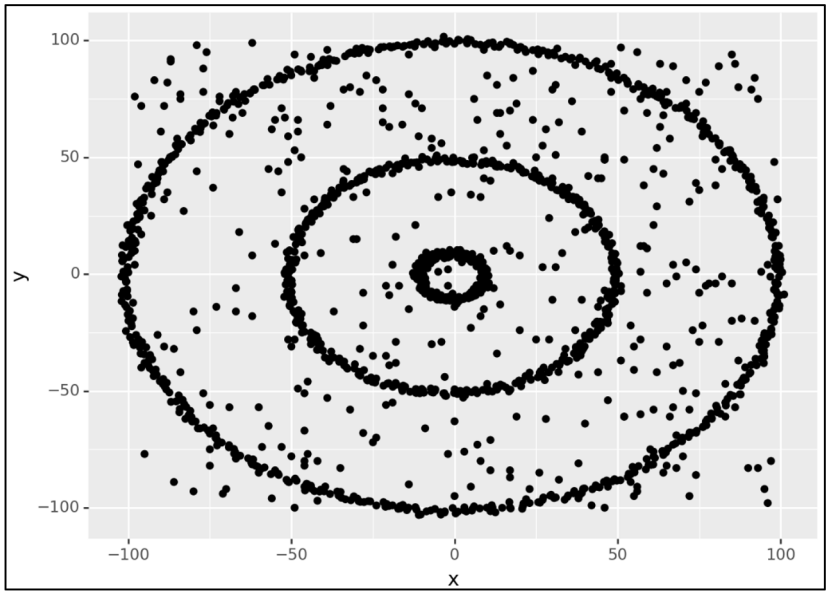

Figura 2.17: Instancia 8, Inicial

- **Número de *clústeres*: 2**

  Silhouette Score: 0,03901 (SC), 0,29215 (KM).

  Función Objetivo: 204699610,94 (SC), 104962804,05 (KM).

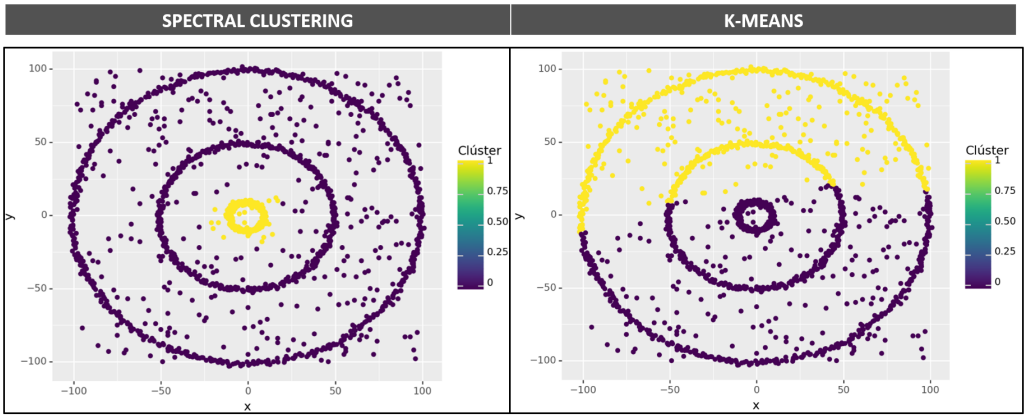

Figura 2.18: Instancia 8, 2 *clústers*

Los resultados presentados muestran similitudes con la instancia 6 analizada previamente. Sin embargo, las diferencias entre los dos métodos son evidentes desde una perspectiva estadística, como se observa en la medida del coeficiente de silueta. En particular, el *Spectral Clustering* muestra un valor considerablemente bajo en comparación con el *k-means*. La cercanía a 0 sugiere que la asignación de puntos en los límites de los grupos podría ser incorrecta. Por ejemplo, los puntos en el círculo intermedio podrían haber sido asignados al *clúster* del círculo más pequeño. Por otro lado, el método tradicional *k-means* conserva su estructura y problemas similares a los casos anteriores.

En términos de distancias, el enfoque de *Spectral Clustering* muestra distancias incluso el doble de largas en comparación con la técnica tradicional. Esto se debe a la inclusión de los círculos de mayor tamaño en un solo grupo, lo que resulta en una suma total de distancias mayor en este método.

- **Número de *clústeres*: 3**

  Silhouette Score: 0,19016 (SC), 0,31150 (KM).

  Función Objetivo: 78654322,75 (SC), 58124708,76 (KM).

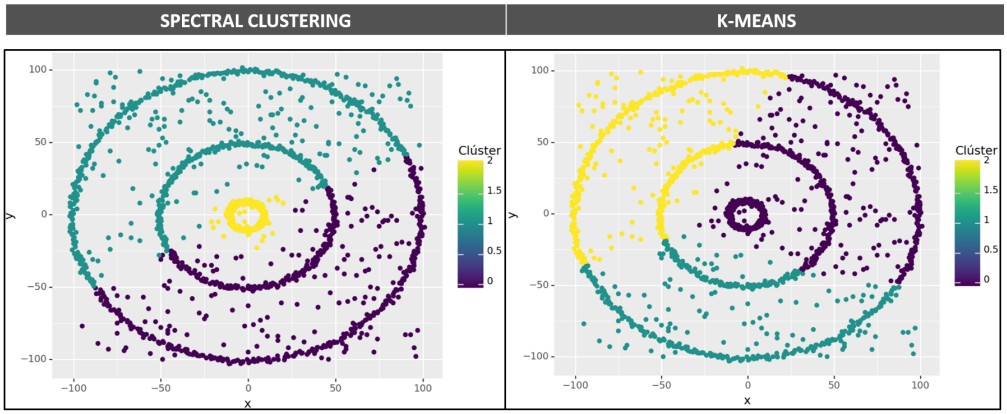

Figura 2.19: Instancia 8, 3 *clústers*

- **Número de *clústeres*: 6**

  Silhouette Score: 0,22263 (SC), 0,38020 (KM).

  Función Objetivo: 26894140,98 (SC), 21007917,18 (KM).

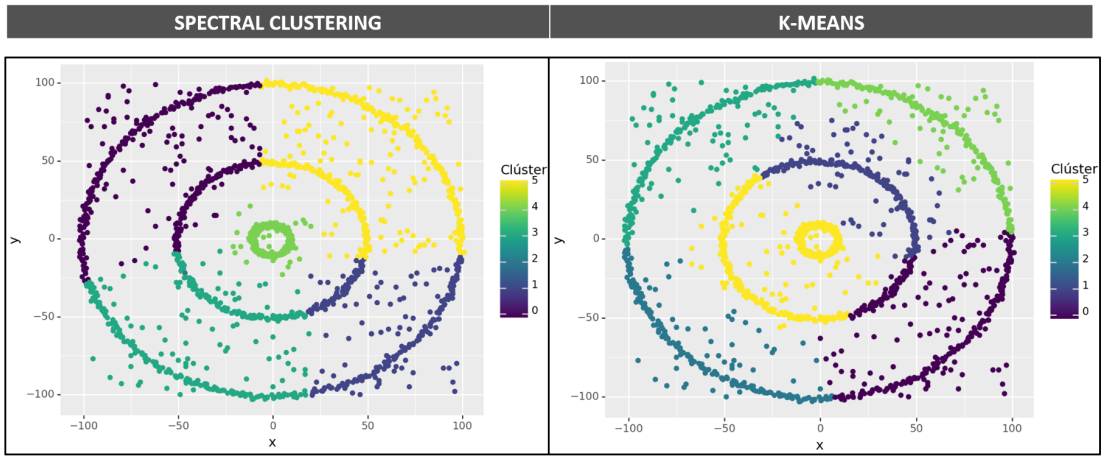

Figura 2.20: Instancia 8, 6 *clústers*

Por otro lado, al incrementar el número de grupos a formar como es en el caso para 3 y 6 grupos, se puede observar que la calidad de los *clústeres* aumenta. Sin embargo, aún persisten grandes diferencias entre ambos métodos, en particular con el método de agrupamiento espectral, que a pesar de subir sus coeficientes, estos siguen cercanos a 0 obteniendo una mala clasificación. En lo que respecta a la distancia total recorrida, la distancia generada por el agrupamiento espectral resulta considerablemente mayor; esto no es eficiente si nuestra intención es minimizar las distancias de trayectorias.

- **INSTANCIA 9**

  Instancia con 3 círculos: un círculo pequeño con 200 puntos y un radio de 30, un círculo mediano con 200 puntos y un radio de 60 y un círculo grande con 200 puntos y un radio de 130 y un ruido con 300 puntos. Trabajando con 900 puntos en total.

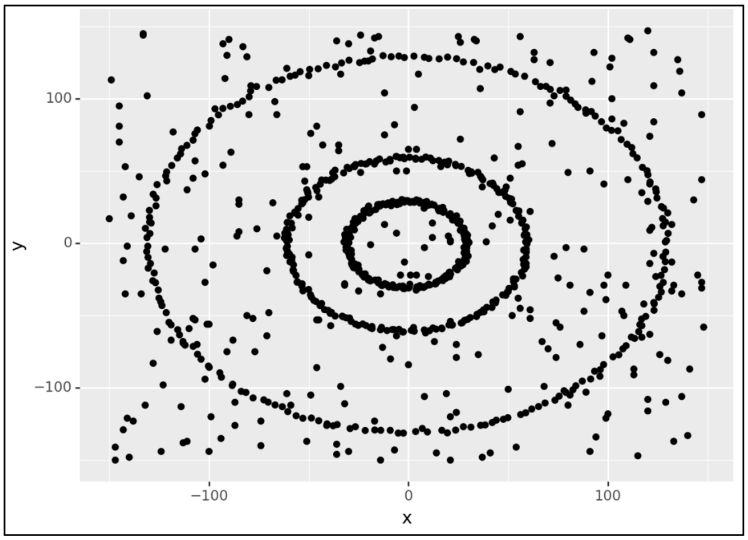

Figura 2.21: Instancia 9, Inicial

- **Número de *clústeres*: 2**

  Silhouette Score: 0,24053 (SC), 0,32990 (KM).

  Función Objetivo: 40973492,96 (SC), 44824896,54 (KM).

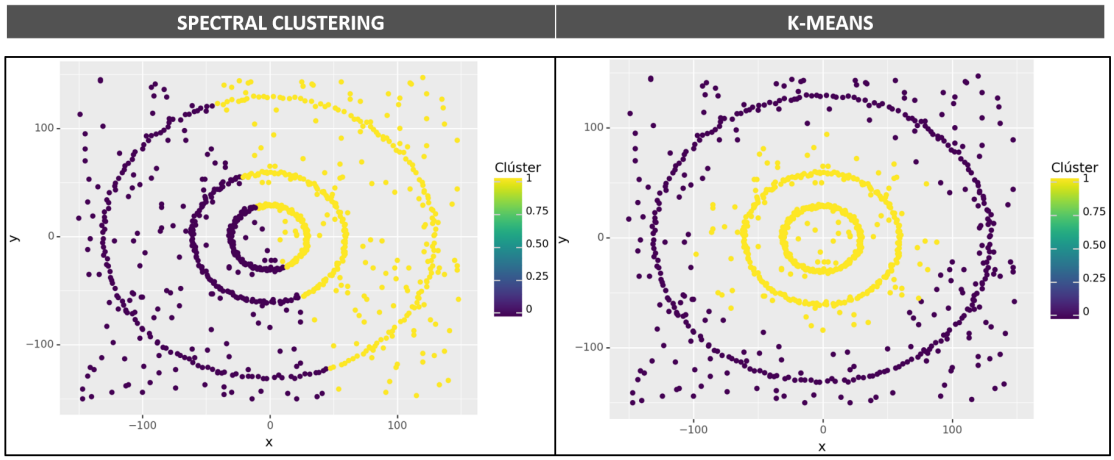

Figura 2.22: Instancia 9, 2 *clústers*

Es esencial destacar que al reducir la cantidad de puntos y, al mismo tiempo, variar los radios de los círculos en esta instancia

particular, los resultados difieren en comparación con los anteriores. Es llamativo observar que el *Spectral Clustering* tiende a agrupar dividiendo en la mitad de los casos, mientras que la técnica *k-means* forma grupos basados en los círculos, lo que no sucedía en instancias anteriores. En este contexto, a pesar de que la medida de *Silhouette* de la técnica de agrupamiento espectral es menor, los resultados de las distancias recorridas entre grupos es mejor a comparación con el método tradicional de agrupamiento. Vale recalcar que hay que tomar en cuenta, cuál es el objetivo que se propone al momento de querer encontrar un particionamiento.

- **Número de *clústeres*: 3**

  Silhouette Score: 0,24327 (SC), 0,39875 (KM).

  Función Objetivo: 23045388,54 (SC), 25686391,67 (KM).

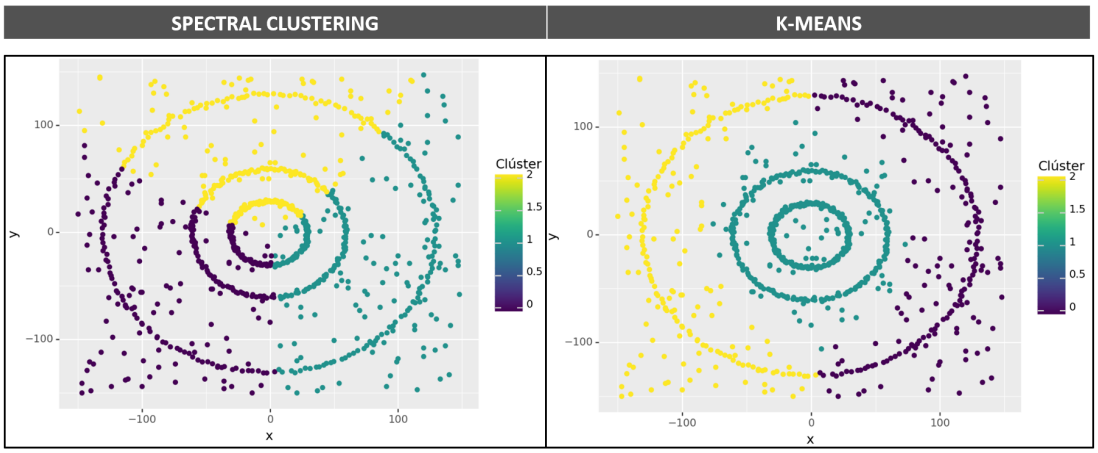

Figura 2.23: Instancia 9, 3 *clústers*

En el caso de 3 grupos, se observa un comportamiento similar a la del escenario con 2 grupos. El *Spectral Clustering* clasifica de manera equitativa a los grupos, dividiéndolos en tres partes iguales, mientras que la técnica de *k-means* los agrupa formando círculos. En estos resultados, el *Silhouette Score* de la técnica *k-means* es considerablemente mejor que el obtenido mediante el método estudiado, es considerable que en los dos casos los puntos que ocasionan problemas son los de las fronteras. No obstante, en términos de las distancias recorridas entre los gru-

pos, el *Spectral Clustering* presenta un resultado con un valor menor con una diferencia significativa ante el otro método.

- **Número de *clústeres*: 6**

  Silhouette Score: 0,37134 (SC), 0,37148 (KM).

  Función Objetivo: 8851811,44 (SC), 8795179,81 (KM).

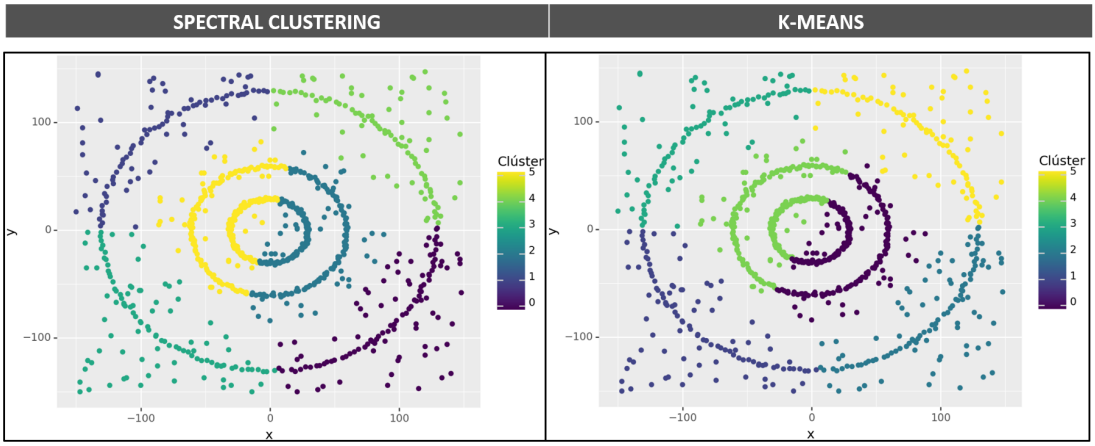

Figura 2.24: Instancia 9, 6 *clústers*

En el último de los casos al comparar los resultados en la clasificación en 6 grupos, se puede observar una notable similitud, e incluso sus valores en términos del *Silhouette score*, los cuales resultan cercanos; indicándonos que cada vez que aumenta la cantidad de grupos, los *clústers* presentan mejor calidad y se encuentran bien separados. Sin embargo, en este escenario, el resultado con menor valor en cuanto a las distancias fue logrado por la técnica tradicional denominada *Spectral Clustering*.

A través del análisis de los experimentos, se puede constatar que los resultados no siguen un patrón predecible. No obstante, en las instancias aleatorias, el comportamiento del *Spectral Clustering*, a pesar de no alcanzar siempre la mejor puntuación en el *Silhouette Score*, demostró ser mejor que la técnica tradicional, ya que contaba con distancias menores a las brindadas por el *k-means*, al igual que en algunos casos con estructura circular. Además, al trabajar con estructuras circulares que presentan un mayor grado de complejidad, al momento de clasificar no sé tuvo el resultado esperado y presento problemas al separar los mismos.

Capítulo 3

Resultados

3.1. Resultados pruebas computacionales

Una vez realizados experimentos computacionales con diversas estructuras y números de puntos, se observó diferencias entre la aplicación del método tradicional de agrupamiento (*k-means*) y el método del *Spectral Clustering*. Los resultados gráficos mostraron desde pequeñas variaciones hasta cambios significativos en los agrupamientos obtenidos, como fue el caso de instancias con puntos dispersos y con 1 estructura circular.

En escenarios de estructuras complejas, se observó que los resultados fueron bastante similares al trabajar con 2 círculos. En algunos casos, el agrupamiento espectral logró resultados superiores en comparación de la técnica tradicional, tomando en cuenta la función objetivo la cual es maximizar el corte en el grafo asociado. Sin embargo, al lidiar con 3 círculos y estructuras más complejas, los resultados tomaron direcciones opuestas en las representaciones gráficas; a pesar de ello, en las distancias totales entre los grupos, los resultados proporcionados por el *Spectral Clustering* resultaron mejores en comparación con la técnica *k-means*. Es importante destacar que debido a que se trabajó con instancias aleatorias y se variaron parámetros, los resultados pueden manifestar tanto diferencias significativas como diferencias mínimas en respuesta a estos cambios.

Hasta este punto del trabajo, se han presentado detalles sobre la formación de clústeres, los objetivos de cada método, comparaciones de la medida de *Silhouette Score* y la función objetivo, la cual consiste en la suma de todos los puntos dentro de los grupos e incluso la representación gráfica de los resultados de los experimentos. Sin embargo, hasta ahora, no se ha realizado un análisis exhaustivo de los tiempos de ejecución de los algoritmos utilizados.

Evidentemente, el *Spectral Clustering* tiene un mayor tiempo de ejecución en comparación con el método *k-means* estándar, ya que uno de los pasos del algoritmo (representado en el Algoritmo 1) involucra la aplicación del *k-means*, es decir, el tiempo de ejecución de la técnica de agrupamiento espectral contiene tanto el cálculo del espacio propio de la matriz como el de la técnica *k-means*. Por lo tanto, a continuación se presenta una tabla con un resumen de los tiempos de los experimentos realizados previamente, recordando que estos tiempos son promedios obtenidos a partir de 5 experimentos.

En la tabla se incluyen la siguientes información:

- **Tiempo de ejecución:**
  - El tiempo total del «***Spectral Clustering***» se muestra en la tabla de color rojo y se divide en dos fases: el tiempo dedicado al cálculo de los valores y vectores propios («**Fase 1**»); y el tiempo de ejecución del *k-means* en la matriz de vectores propios («***k-means***»). Cada fase muestra el porcentaje que representa con respecto al tiempo total de ejecución.
  - La columna «***k-means***» nos presenta en color rojo los tiempos en que se ejecutó la técnica de *k-means* teniendo como dato de entrada la matriz de similitud.
- La columna «**Representación KM-SC**» muestra el porcentaje de la representación del tiempo de ejecución del *k-means* en el tiempo de ejecución del *Spectral Clustering*, tomando como si el tiempo del agrupamiento espectral es el 100 %.
- La columna de «**Medida de calidad (*Silhouette Score*)**» representa los valores que tiene cada agrupación en ambas técnicas.

- La columna de «**Función Objetivo**» presenta la suma total de las distancias de los puntos de todos los grupos formados en ambas técnicas.

Esta tabla nos ayuda a evaluar el rendimiento tanto del algoritmo *Spectral Clustering* como de la técnica tradicional del *k-means*.

3.1.1. Tiempos Instancias randoms

Forma de la figura				Tiempo de ejecución (segundos)					Representación	Medida de calidad		Función Objetivo	
Diseño	Puntos	Total de puntos	*Clústeres*	*Spectral Clustering*				*K-means*	KM-SC	*Silhouette Score*		Distancia Total	
				Fase 1		K-means				*Spectral Clustering*	*K-means*	*Spectral Clustering*	*K-means*
Instancias Dispersas	**25**	**25**	**2**	**0,01022**				**0,01082**	106 %	0,405097587	0,416061215	494,6647738	422,9843293
				0,00101	**9,90 %**	0,00921	**90,10 %**						
			4	**0,01335**				**0,01345**	101 %	0,329421229	0,352433502	142,6091684	146,4543677
				0,00017	**1,30 %**	0,01317	**98,70 %**						
			10	**0,02741**				**0,02782**	101 %	0,345565159	0,401056824	21,15942435	28,19923175
				0,00080	**2,93 %**	0,02661	**97,07 %**						
	100	**100**	**2**	**0,02467**				**0,01483**	60 %	0,369021702	0,372357776	7827,799972	7871,254981
				0,01385	**56,17 %**	0,01081	**43,83 %**						
			3	**0,03045**				**0,01763**	58 %	0,353023838	0,360504526	2572,951943	2471,859475
				0,01534	**50,38 %**	0,01511	**49,62 %**						
			4	**0,03198**				**0,01995**	62 %	0,373704947	0,375930664	4138,767476	4105,355258
				0,01275	**39,86 %**	0,01923	**60,14 %**						
	200	**200**	**2**	**0,07055**				**0,03222**	46 %	0,322029679	0,341974885	8658,034868	7966,5006
				0,05370	**76,11 %**	0,01685	**23,89 %**						
			4	**0,07623**				**0,03402**	45 %	0,361068265	0,363463686	4204,46412	4331,830759
				0,05328	**69,90 %**	0,02295	**30,10 %**						
			6	**0,07638**				**0,03783**	50 %	0,338294251	0,381955407	1803,000612	1417,246367
				0,05431	**71,10 %**	0,02207	**28,90 %**						
1 círculo	c_1**=200**	**200**	**2**	**0,07112**				**0,02312**	33 %	0,375801859	0,380408885	17182,67998	17124,07232
				0,05885	**82,74 %**	0,01227	**17,26 %**						
			3	**0,07651**				**0,02477**	32 %	0,427459992	0,438241785	8502,482754	8319,537891
				0,05753	**75,19 %**	0,01898	**24,81 %**						
			6	**0,07955**				**0,03756**	47 %	0,434811109	0,435679142	2385,036588	2381,764239
				0,05663	**71,19 %**	0,02291	**28,81 %**						
	c_1**=1000**	**1000**	**2**	**1,30057**				**0,11776**	9 %	0,398681295	0,382326055	427359,0886	426426,613
				1,28783	**99,02 %**	0,01274	**0,98 %**						
			3	**1,33052**				**0,17249**	13 %	0,438725191	0,439315066	207007,6857	207376,5718
				1,30581	**98,14 %**	0,02471	**1,86 %**						
			6	**1,44553**				**0,31756**	22 %	0,435825425	0,443303033	59894,78149	58980,55701
				1,37071	**94,82 %**	0,07482	**5,18 %**						
2 círculos	c_1**=200** c_2**=400** r_1**=25** r_2**=50 ruido=200**	**800**	**2**	**0,83844**				**0,13911**	17 %	0,336062262	0,335984482	13355753,87	13334219,3
				0,82503	**98,40 %**	0,01342	**1,60 %**						
			3	**0,89710**				**0,14894**	17 %	0,367282773	0,367505496	6898351,127	6892408,643
				0,87656	**97,71 %**	0,02054	**2,29 %**						
			6	**0,92174**				**0,16663**	18 %	0,362251161	0,362227626	2533120,719	2523062,665
				0,86667	**94,03 %**	0,05506	**5,97 %**						
	c_1**=300** c_2**=800** r_1**=10** r_2**=50 ruido=200**	**1300**	**2**	**2,77107**				**0,34618**	12 %	0,166073	0,318175797	75025077,55	47278691,62
				2,75503	**99,42 %**	0,01604	**0,58 %**						
			3	**3,20501**				**0,41630**	13 %	0,344684544	0,355565669	28381587,19	25797541,37
				3,17774	**99,15 %**	0,02726	**0,85 %**						
			6	**3,31235**				**0,54043**	16 %	0,40140539	0,49359249	8367594,09	6756637,42
				3,25203	**98,18 %**	0,06032	**1,82 %**						

Forma de la figura				**Tiempo de ejecución**						**Medida de calidad**		**Función Objetivo**	
Diseño	**Puntos**	**Total de puntos**	***Clústeres***	***Spectral Clustering***				***K-means***	**Representación KM-SC**	***Silhouette Score***		**Distancia Total**	
				Fase 1		***K-means***				***Spectral Clustering***	***K-means***	***Spectral Clustering***	***K-means***
3 círculos	c_1**=200** r_1**=10** c_2**=400** r_2**=50** c_3**=600** r_3**=100 ruido=300**	**1500**	**2**	**4,09710**				**0,48990**	12 %	0,039006349	0,292152993	204699610,9	104962804
				4,07990	**99,58 %**	0,01720	**0,42 %**						
			3	**4,32493**				**0,65475**	15 %	0,190160994	0,3115009	78654322,75	58124708,76
				4,29709	**99,36 %**	0,02784	**0,64 %**						
			6	**4,35304**				**0,81156**	19 %	0,222630124	0,380202553	26894140,97	21007917,18
				4,28789	**98,50 %**	0,06515	**1,50 %**						
	c_1**=200** r_1**=30** c_2**=200** r_2**=60** c_3**=200** r_3**=130 ruido=300**	**900**	**2**	**1,02045**				**0,12440**	12 %	0,240531117	0,329904399	40973492,96	44824896,54
				1,00796	**98,78 %**	0,01249	**1,22 %**						
			3	**1,10624**				**0,14807**	13 %	0,24327149	0,398750226	23045388,54	25686391,67
				1,07386	**97,07 %**	0,03238	**2,93 %**						
			6	**1,14894**				**0,24220**	21 %	0,371338653	0,371481849	8851811,439	8795179,806
				1,08580	**94,50 %**	0,06314	**5,50 %**						

En la instancia menor (25 puntos), se pudo notar que el tiempo empleado por el algoritmo *Spectral Clustering* fue menor en todos los grupos en comparación con el tiempo empleado por el *k-means*. Esto se debe a que, al tratarse de una cantidad reducida de puntos, los cálculos de valores y vectores propios no presentan un alto costo computacional, permitiendo que la técnica *Spectral Clustering* requiera realizar la menor cantidad de iteraciones posibles al encontrar los *clústeres*. Además, se pudo apreciar que la mayor parte del tiempo se empleó en la ejecución de la fase 1, representan alrededor del 90 %, 94 % y 97 % del tiempo total, respectivamente. En este caso en particular el tiempo ejecutado por la técnica *k-means* es mayor con un 6 % y 1 % que el tiempo del agrupamiento espectral.

Por otro lado, se llevaron a cabo experimentos incrementando el número de puntos con el propósito de observar cómo afecta esto a los tiempos de ejecución, al momento de disponer de una mayor cantidad de información, los tiempos tienden a incrementarse. Sin embargo, es importante destacar que la mayor parte del tiempo se dedica al cálculo del espacio propio en lugar del agrupamiento en sí. Notando que el algoritmo más rápido es el de la técnica de agrupamiento tradicional.

Una vez realizados los experimentos con puntos dispersos en el plano, se procedió a cambiar de estructura a un círculo con 200 puntos. Se esperaba que los resultados fueran similares a las soluciones anteriores (sin estructura circular) con 200 puntos. Sin embargo, se observó que no resultó ser así.

Por ejemplo, al crear dos grupos en instancias dispersas, el 76,11 % del tiempo se destinó a la primera fase y el 23,89 % restante fue para el algoritmo *k-means*. En cambio, al trabajar con la estructura del círculo, se encontró que el 82,74 % del tiempo se empleó en la primera fase y solo el 17,26 % se empleó para la segunda fase (*k-means*). La diferencia en la unidades de los tiempos no es significativa, sin embargo presenta variaciones y esto se debe a que son instancias con puntos aleatorios y los valores cambian. A pesar de que el cambio en el tiempo por fases fue del 6 %, a la escala en la que se realizaban los cálculos (0,005 seg.), esta variación se consideró relativamente mínima. Los resultados obtenidos fueron similares en la clasificación de 3 y 6 grupos.

Por otro lado, cuando se aumentó el número de puntos a 1000, se observó un incremento en el tiempo total de aproximadamente 1.3 segundos en comparación con el experimento de 200 puntos. La mayor parte del tiempo fue dirigida a la primera fase, y solo el 0,98 % del tiempo total se utilizó para el algoritmo *k-means*. Confirmando que el algoritmo del *Spectral Clustering* tiene un alto costo computacional.

En otro aspecto, los cambios más notables en la formación de grupos se observaron en los casos con 2 y 3 círculos con un radio determinado.

Analicemos el caso de la instancia con 800 puntos en total, donde se distribuyeron de manera que había un círculo pequeño con 200 puntos, un círculo más grande con 400 puntos y un conjunto de puntos dispersos (ruido) con 200 puntos. Aunque se esperaba que esta estructura compleja llevara a un aumento en los tiempos de ejecución, se encontró que los mismos no variaron significativamente. Esto se debe a que el algoritmo da mayor importancia a la cantidad de puntos que a la complejidad de la estructura en sí. Los tiempos encontrados fueron de 0.8384, 0.89710 y 0.92174 para 2, 3 y 6 grupos respectivamente. El porcentaje de tiempo dedicado al *k-means* (segunda fase) fueron del 1.60 %, 2.29 % y 5.97 % respectivamente, es decir, a medida que se aumenta el número de grupos, el porcentaje de tiempo en la fase 2 va incrementándose. Sin embargo, al comparar el tiempo del *k-means* en su versión estándar con el tiempo total del *Spectral Clustering*, representan aproximadamente solo el 17 %, y 18 % del tiempo utilizado en la misma, obteniendo un tiempo considerablemente mayor en el algoritmo de *Spectral Clustering* en comparación con el tiempo requerido por el método de *k-means*

Ahora bien, al considerar el caso donde se añade un círculo adicional y se incrementa la cantidad de puntos en el ruido del gráfico, la estructura de trabajo sigue siendo similar, pero los tiempos tienden a aumentar, especialmente en la fase 1 en comparación con la fase 2 (*k-means*). Es importante destacar que, a medida que el gráfico se vuelve más complejo y la cantidad de puntos aumenta, el rendimiento del *k-means* en el *Spectral Clustering* mejora significativamente.

3.2. Resultados para la Instancia del Agrupamiento en el Campeonato Ecuatoriano de Fútbol

Los resultados anteriores han demostrado que el *Spectral Clustering* forma grupos de manera más eficiente en instancias con un diseño de dispersión sin ninguna estructura; sin embargo, no garantiza que los grupos sean necesariamente balanceados en términos de cardinalidad. El objetivo principal de este trabajo de titulación es aplicar el *Spectral Clustering* en la instancia que cuenta con 21 ciudades del Ecuador donde se juega la liga de fútbol de segunda categoría. Para cumplir con las condiciones establecidas en este campeonato, se desarrolló un nuevo algoritmo *k-means* que asegura la formación de grupos con la misma cantidad de ciudades o con una diferencia máxima de 1 entre ellos.

Para este campeonato la FEF contaba con un solución empírica la cual fue presentada por Recalde (2018) en [3] y al adaptarse a los datos presentes en este trabajo se cuenta con un total de **39936.4 km** recorridos.

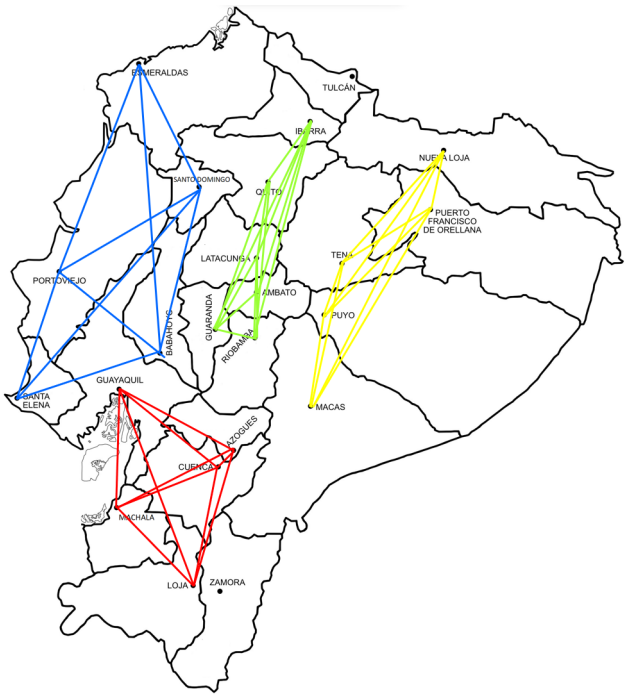

Figura 3.1: Solución Empírica al campeonato de la segunda liga de la FEF
Fuente: Imagen adaptada de [3]

Sin embargo, es posible mejorar esta solución para obtener resultados mejores. Al aplicar el *Spectral Clustering* con el *k-means* se obtuvo varias soluciones; algunas de estas superaban considerablemente a la solución empírica, o se aproximaban a ella. Sin embargo, al considerar todas las soluciones disponibles y evaluar la función objetivo, la solución de menor valor se destaca.

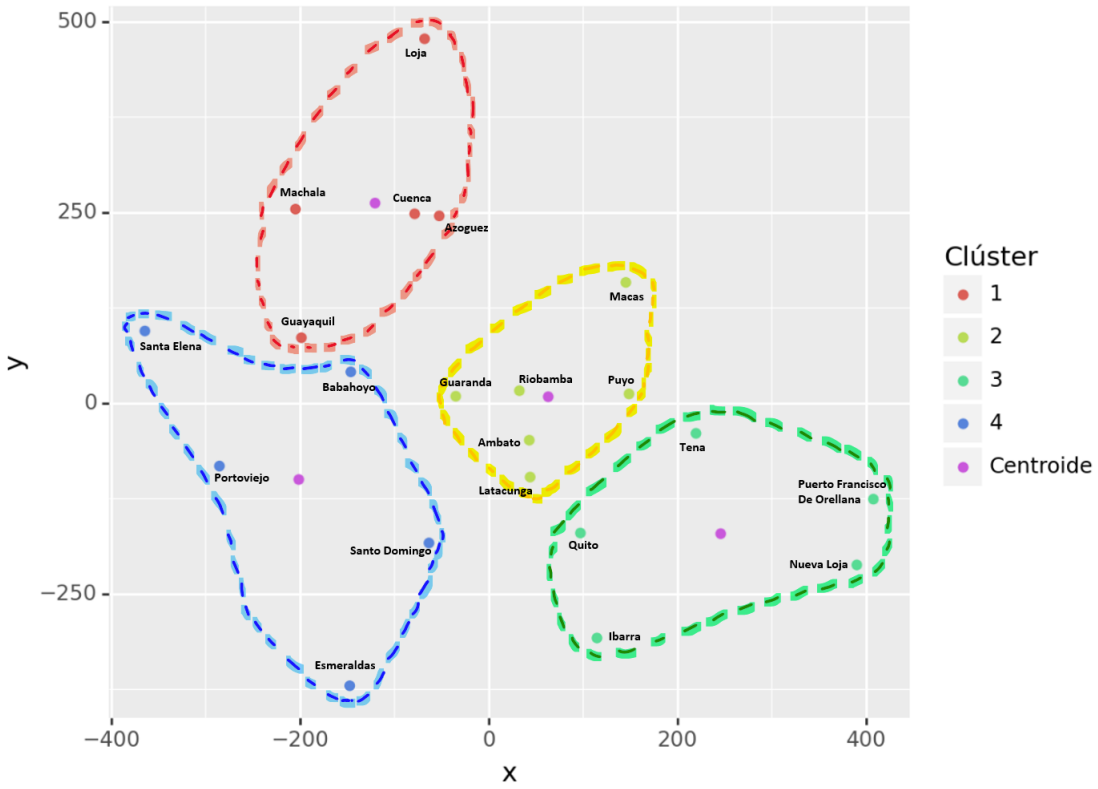

Figura 3.2: Solución de la instancia de ciudades

El algoritmo *k-means*, implementado con la restricción de cardinalidad, fue diseñado para ejecutar iteraciones con diversos puntos de inicialización y comparar los resultados obtenidos e ir variando de acuerdo a los grupos que ya completen la cardinalidad esperada. Es por esta razón que se generaron múltiples soluciones divergentes.

La solución nos presentó grupos conformados por:

- **Grupo 1:** Cañar, Azuay, Guayas, Loja, El Oro.
- **Grupo 2:** Cotopaxi, Tungurahua, Chimborazo, Bolivar, Pastaza, Morona Santiago.

- **Grupo 3:** Pichincha, Imbabura, Napo, Sucumbios, Orellana.
- **Grupo 4:** Santa Elena, Los Ríos, Manabí, Esmeraldas, Santo Domingo.

Con estos grupos formados, la distancia recorrida total es de **38747.6 km**, y se obtuvieron en un tiempo de 0.434747 segundos. Es importante mencionar que este tiempo es mayor que el del experimento de la instancia 1, que contaba con solo 25 puntos. La razón detrás de esta diferencia en tiempo es que los experimentos anteriores utilizaron el algoritmo *k-means* implementado en Python, mientras que para esta instancia se programó un nuevo algoritmo.

Es sorprendente ver que los grupos formados con este método coinciden con la solución óptima expuesta en [3], ya que se utilizó un método de aproximación donde se realizo varias ejecuciones para escoger la de mejor rendimiento. No obstante, los resultados difieren debido a que no se utilizaron los mismos datos del artículo. La diferencia entre la solución empírica y la solución óptima es de $1188{,}8km$, lo que representa la cantidad de kilómetros en carretera que se lograron ahorrar en el campeonato. A continuación, se muestra una representación gráfica en el mapa de Ecuador de la solución óptima.

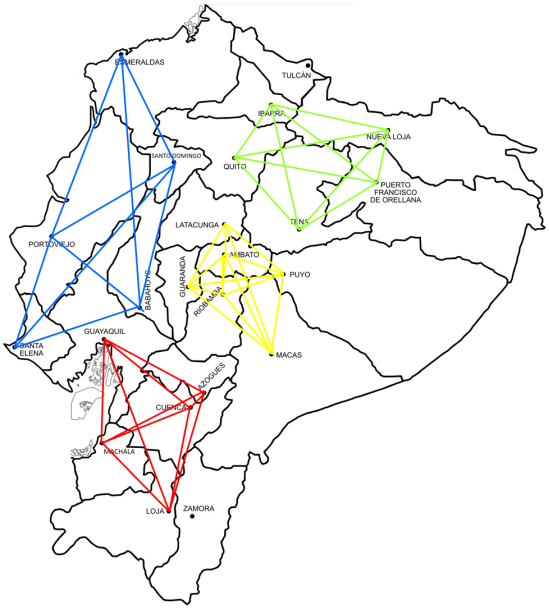

Figura 3.3: Solución Óptima al campeonato de la segunda liga de la FEF
Fuente: Imagen adaptada de [3]

Capítulo 4

Conclusiones y Recomendaciones

4.1. Conclusiones

Luego de un estudio del método *Spectral Clustering* tanto en pruebas computacionales como en una instancia real de distancias entre provincias del Ecuador se puede concluir que,

- Al utilizar el agrupamiento espectral, es esencial considerar el objetivo principal del estudio, ya que ciertos indicadores estadísticos como el *Silhouette Score* nos proporcionan información sobre la calidad de la agrupación. Sin embargo, la misma debe ir ligada a otra prueba estadística que verifique el resultado, como se presento en este trabajo al tener un enfoque principal el cual es encontrar el corte máximo de un grafo.
- El *Spectral Clustering* es una técnica de agrupamiento que ha demostrado ser eficiente en muchas de las aplicaciones como en redes sociales al buscar similitudes entre personas. Sin embargo, para problemas donde se maximiza el corte en un grafo, la técnica de agrupamiento espectral tiene un comportamiento similar a la técnica del *k-means* tradicional, en algunos casos el resultado es más eficiente y en otros no se halla la solución óptima sino una aproximación.

- Se tuvo en cuenta un aspecto importante al analizar los resultados, que fue el comportamiento de los signos de los valores propios de la matriz estocástica, ya que los mismos pueden influir en los resultados. Esto se debe a que el *Spectral Clustering* en cadenas de Markov es un problema dual del *Spectral Clustering* clásico, y se seleccionan los valores propios más cercanos al círculo unitario. Cuando estos valores son positivos, el algoritmo funciona de manera habitual; sin embargo, si un valor propio negativo está más cercano al círculo unitario, se debe trabajar con dicho valor en su módulo, ya que proporciona información relevante para la clasificación de datos en *clústeres*.

- El uso del *Spectral Clustering* como una cadena de Markov es una poderosa herramienta para clasificar datos, especialmente en situaciones donde las matrices no son simétricas, ya que el algoritmo es aplicable tanto en grafos no dirigidos como dirigidos, lo que amplía su potencial en diversas áreas de aplicación.

- En instancias con una cantidad reducida de datos, el *Spectral Clustering* presenta un comportamiento similar al del *k-means*, con resultados bastante parecidos o incluso idénticos. Sin embargo, el tiempo de ejecución del agrupamiento espectral se destaca al ser menor debido a que al trabajar con un número reducido de vectores propios, el costo computacional es bajo; provocando que la técnica del *k-means* realice la menor cantidad de iteraciones posible, lo que también contribuye a una mayor eficiencia en términos de tiempo de ejecución.

- Con lo que respecta al costo computacional del cálculo espectral, este aumenta a medida que aumenta la cantidad de puntos, lo que se traduce en una mayor proporción del tiempo de ejecución destinado a la misma. A pesar de esto, los tiempos de ejecución no son excesivamente altos, incluso en casos con un gran número de puntos, como en nuestra instancia con 1500 puntos que tomó alrededor de 4.3 segundos para completar la agrupación al calcular 1500 valores y autovectores.

- En cuanto a la aplicación en una instancia real con datos de ciuda-

des del Ecuador, los resultados coincidieron con los obtenidos en un estudio del mismo caso, después de realizar varias ejecuciones del mismo. Esto indica que los problemas de particionamiento *k-way* pueden abordarse tanto mediante un modelo de programación entera como a través de técnicas de agrupamiento, como el *Spectral Clustering*, pero sin tener la garantía de corroborar que nos da la solución óptima.

4.2. Recomendaciones

- Dentro de los métodos de *Spectral Clustering*, existen dos formas de calcular: utilizando la matriz Laplaciana o la matriz de transición. Este trabajo fue enfocado en trabajar con la matriz de transición y cadenas de Markov. Sin embargo, es recomendable realizar un estudio más profundo para visualizar las diferencias entre ambos métodos del *Spectral Clustering*, más no comparar un método tradicional con uno haciendo uso de espectros.
- Dado que el *Spectral Clustering* tiende a tener un costo computacional elevado, sería beneficioso explorar posibles optimizaciones en el algoritmo para mejorar su eficiencia y reducir el tiempo de cálculo de las soluciones.
- Existen varios algoritmos enfocados en las técnicas de agrupamiento, y en este trabajo se ha utilizado el *k-means* clásico. Sin embargo, sería recomendable realizar una comparación con otras variantes más desarrolladas dentro del *k-means* como: *k-means++*, *k-means* paralelo, K-medoids, fuzzy *k-means*, entre otros. Estas variantes podrían aportar mejoras significativas en términos de rendimiento y eficiencia durante la ejecución del algoritmo.

Capítulo A

Anexo I: Instancias a prueba

A.1. Instancia 2

Se presentan los resultados gráficos de la instancia 2 con 100 puntos aleatorios.

Gráfico inicial

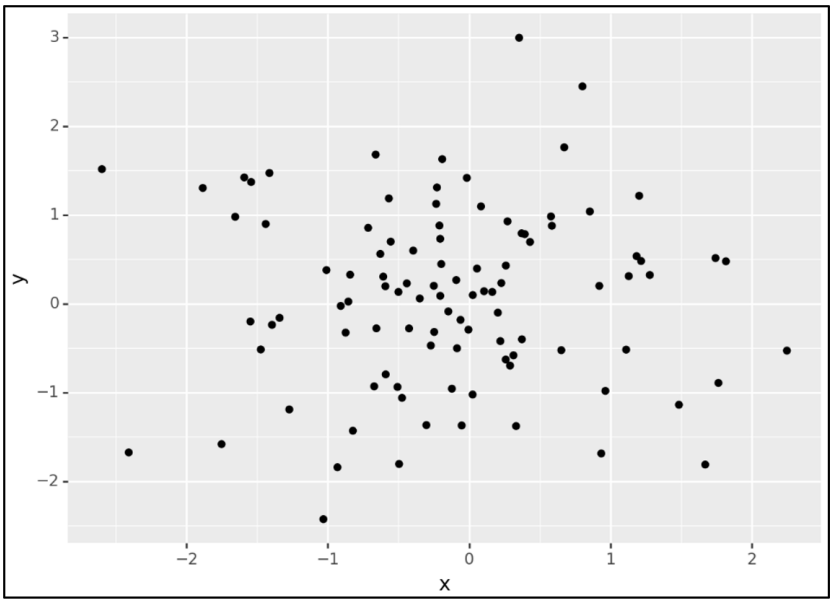

Figura A.1: Instancia 2

- **Número de clúster: 2**

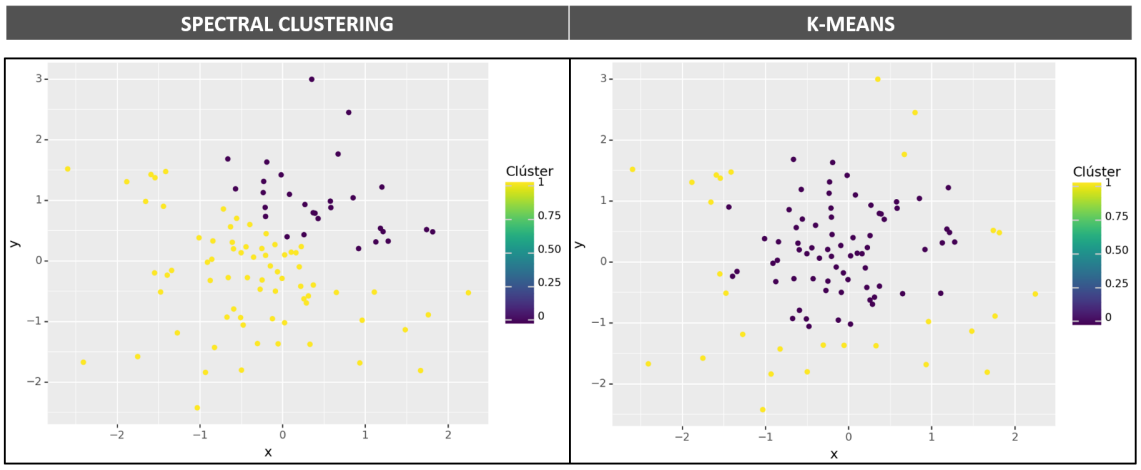

Figura A.2: Instancia 2, 2 clústers

- **Número de clúster: 3**

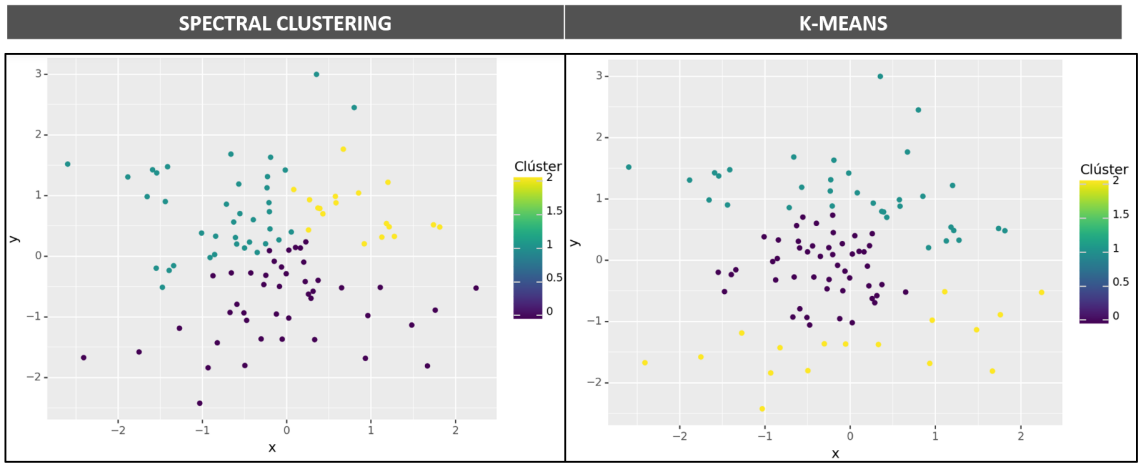

Figura A.3: Instancia 2, 3 clústers

- **Número de clúster: 6**

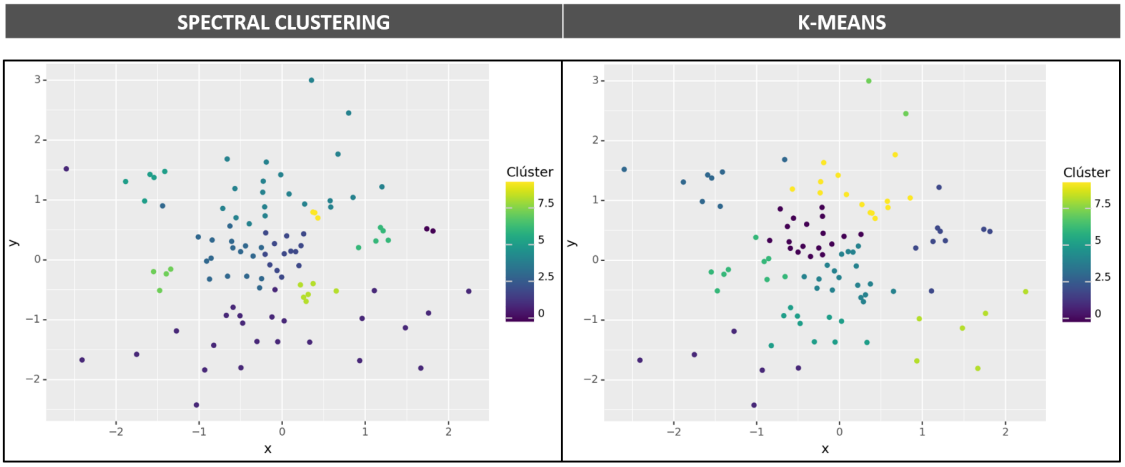

Figura A.4: Instancia 2, 6 clústers

A.2. Instancia 3

Se presentan los resultados gráficos de la instancia 3 con 200 puntos aleatorios.

Gráfico inicial

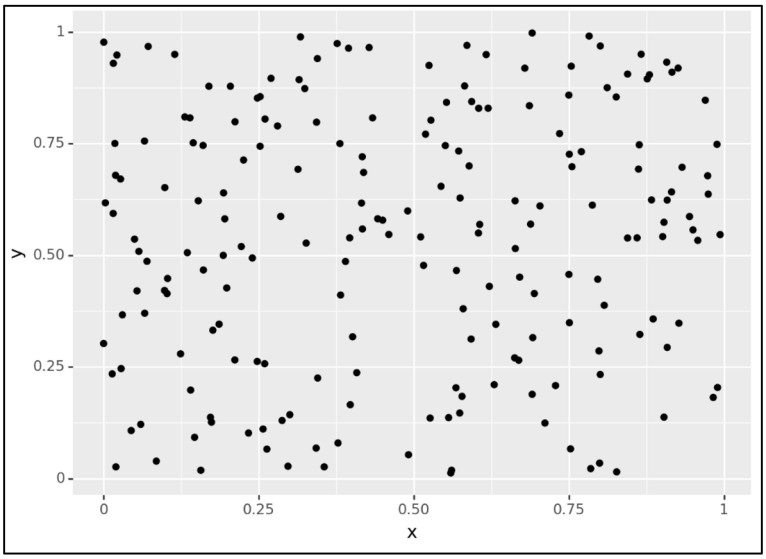

Figura A.5: Instancia 3

- **Número de clúster: 2**

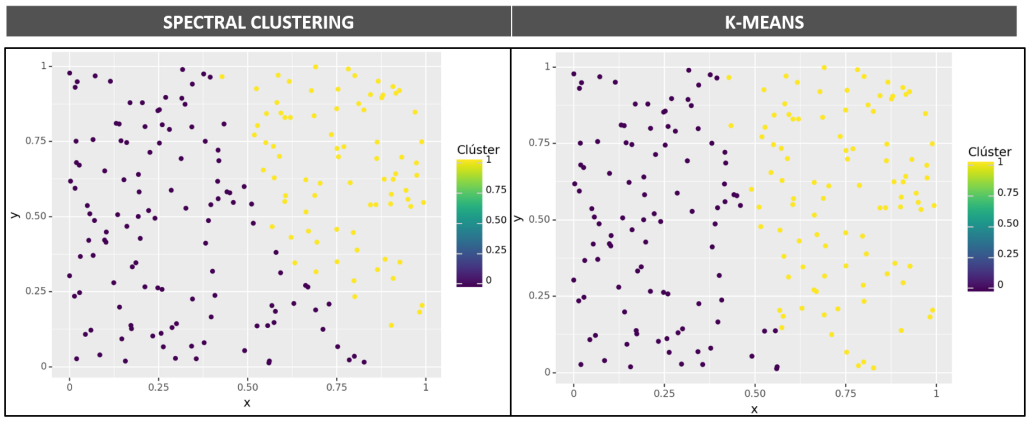

Figura A.6: Instancia 3, 2 clústers

- **Número de clúster: 3**

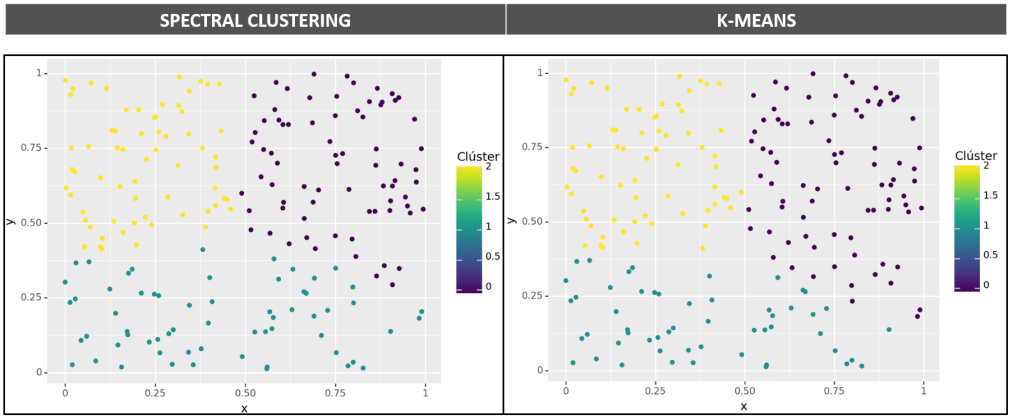

Figura A.7: Instancia 3, 3 clústers

- **Número de clúster: 6**

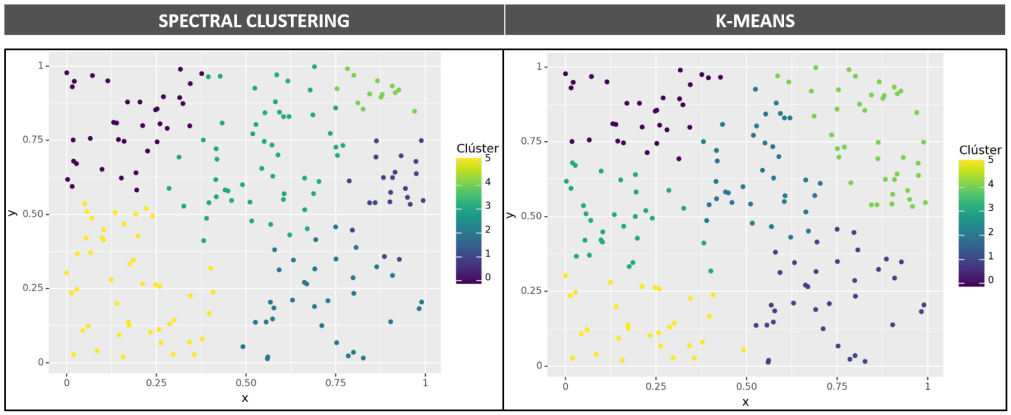

Figura A.8: Instancia 3, 6 clústers

A.3. Instancia 5

Se presentan los resultados gráficos de la instancia 5 con 200 puntos aleatorios y estructura circular.

Gráfico inicial

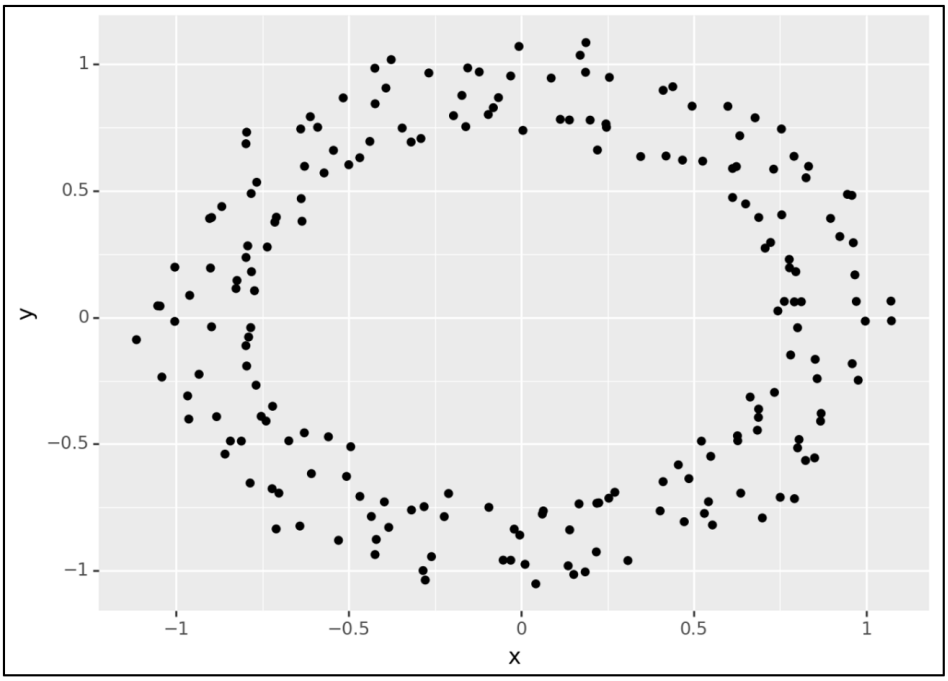

Figura A.9: Instancia 5

- **Número de clúster: 2**

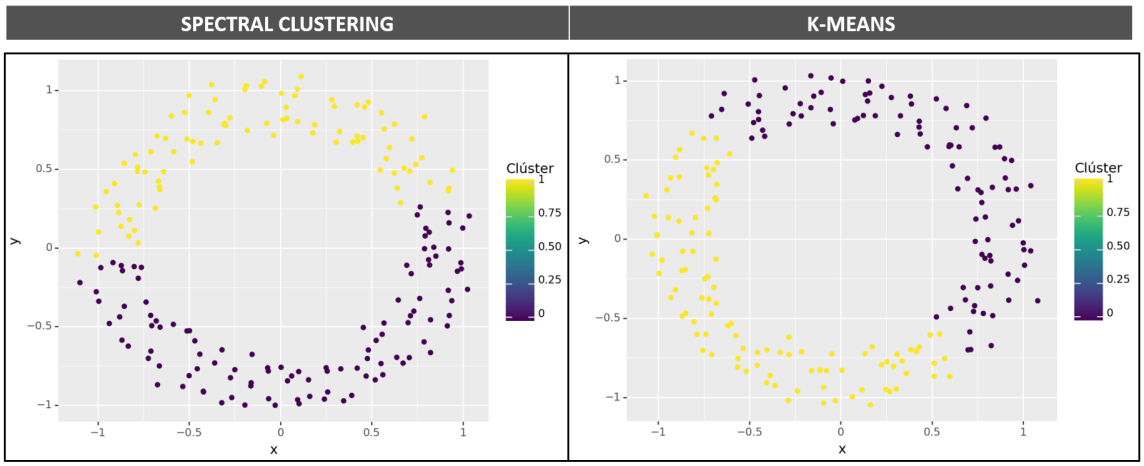

Figura A.10: Instancia 5, 2 clústers

- **Número de clúster: 3**

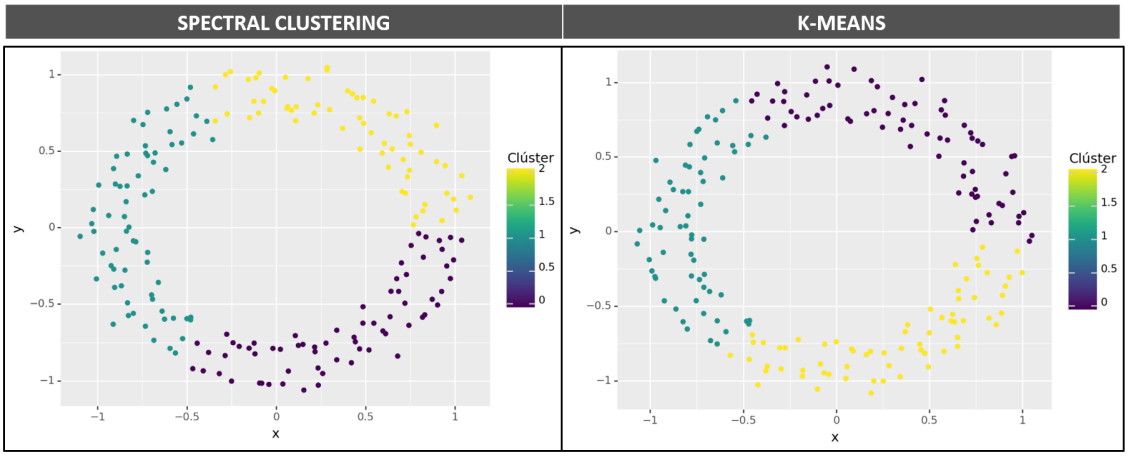

Figura A.11: Instancia 5, 3 clústers

- **Número de clúster: 6**

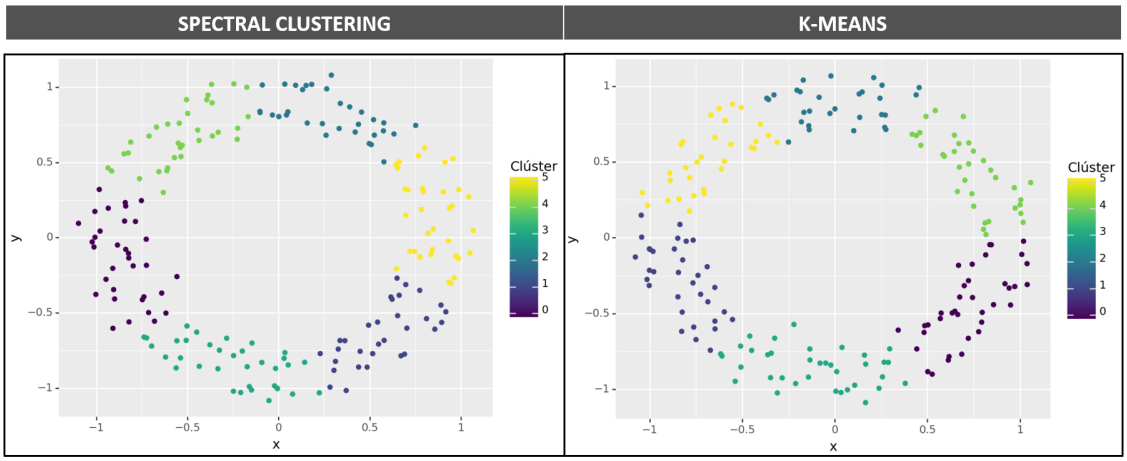

Figura A.12: Instancia 5, 6 clústers

Referencias bibliográficas

[1] Ashutosh Bhardwaj. Silhouette coefficient, may 2020.

[2] Marina Chatterjee. Introduction to spectral clustering, oct 2022.

[3] Ramiro Torres Diego Recalde, Daniel Severín and Polo Vaca. An exact approach for the balanced k-way partitioning problem with weight constraints and its application to sports team realignment. *Journal of Combinatorial Optimization*, page 916–936, 2018.

[4] Donath W. E. and Hoffman A. J. Lower bounds for the partitioning of graphs. *IBM Journal of Research and Development*, 17(5):420–425, sep 1973.

[5] Miroslav Fiedler. Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal*, 23(2):298–305, 1973.

[6] Garret Fitzmaurice. *Handbook of Cluster Analysis*. Christian Hennig, Marina Meila, Fionn Murtagh, Roberto Rocci, New York, first edition, 2015.

[7] Joanna Gutiérrez. *Monitorización, detección y estimación de estados de fallo en la calidad del agua de redes de distribución urbanas*. PhD thesis, Universidad Polit[ecnica de Valencia, Valencia, España, apr 2021.

[8] Martin Helm. A deep dive into k-means. jun 2021.

[9] Javier Beltrán Jorba. Geocluster: Librería de algoritmos de agrupamiento para objetos geoespaciale. Master's thesis, Universidad Zaragoza, jun 2016.

[10] Ning Liu and William J. Stewart. *Markov Chains and Spectral Clustering*, volume 6821, pages 87–98. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[11] J. Izquierdo M. Herrera, J.A. Gutiérrez-Pérez and R. Pérez-García. Métodos de análisis espectral para la sectorización de la red de abastecimiento y su posterior gestión. sep 2012.

[12] Bruno Menegola. *A Study of the k-way Graph Partitioning Problem*. PhD thesis, Universidad Federal de Rio Grande Do Sul, Porto Alegre, 2012.

[13] Natalia Pignataro and Guillermo Figueredo. Introducción al reconocimiento de patrones, spectral clustering. 2008.

[14] Manuel Herrera; Joaquín Izquierdo; Rafael Pérez-García and David Ayala-Cabrera. La regularización del grafo de la red de abastecimiento de agua para la propuesta de su sectorización. oct 2011.

[15] Chris Fraley; Adrian E Raftery. Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*, 97(458):611–631, jun 2002.

[16] Deepak Verma and Marina Meila. Comparison of spectral clustering methods.

[17] Erica Vidal. *Algoritmo divisivo de clustering con determinaci´on autom´atica de componentes*. PhD thesis, Universidad Nacional de Rosario, Rosario - Santa Fe - Argentina, 2014.

[18] Gregor von Laszewski. Intelligent structural operators for the k-way graph partitioning problem. Technical report, Syracuse University, New York, 1991.

[19] Ulrike von Luxburg. A tutorial on spectral clustering. *Stat Comput*, pages 395–416, 2007.

[20] Scott White and Padhraic Smyth. A spectral clustering approach to finding communities in graphs. pages 1–12.

[21] Jyoti Yadav and Monika Sharma. Monika sharma. *International Journal of Engineering Trends and Technology (IJETT)*, Volume 4(7):2972–2076, jul 2013.

[22] Luis Ángel Alcántara Rosas. K means: Programando el algoritmo desde cero en python, nov 2019.