



*Instituto Politécnico Nacional.  
Escuela Superior de Cómputo.*

# Proyecto de clustering

Asignatura:  
Minería de datos.

Profesora: Ocampo Botello Fabiola.  
Grupo: 3CV6.

Fecha de entrega:  
11 de noviembre de 2019

Alumno:

- Lara Cázares Jaime Arturo
- Morales Flores Víctor Leonel
- Ramos Diaz Enrique.



# Proyecto de clustering

Tabla comparativa.

| Algoritmo         | Tipo                    | Características  |
|-------------------|-------------------------|--|
| Aglomerativo      | Jerárquico              | <ul style="list-style-type: none"><li>• Inicia con cada elemento en un cluster, luego fusiona todos en uno de forma iterativa.</li><li>• Utiliza una matriz de adyacencia con distancias entre puntos: <b>single link, average link, complete link</b>.</li><li>• Genera un dendograma.</li><li>• Alta complejidad: <math>O(n^2)</math></li><li>• No es incremental (dinámico).</li></ul>  |
| MST Single Link   | Jerárquico aglomerativo | <ul style="list-style-type: none"><li>• Encontrar el número máximo de elementos conectados en un grafo.</li><li>• Dos clusters se fusionan si la <b>distancia mínima</b> entre ellos es menor o igual que una distancia umbral dada.</li><li>• Complejidad en cada iteración: <math>O(n^2)</math></li><li>• Los clusters se fusionan en orden creciente de la distancia encontrada en un Minimum Spanning Tree.</li><li>• Efecto de cadena: clusters con elementos sin relación, sólo eran cercanos.</li></ul> |
| MST Average Link  | Jerárquico aglomerativo | <ul style="list-style-type: none"><li>• Dos clusters se fusionan si la <b>distancia promedio</b> entre ellos es menor o igual que una distancia umbral dada.</li><li>• Se debe examinar todo el grafo completo en cada iteración, y no solo el de umbral.</li></ul>  |
| MST Complete Link | Jerárquico aglomerativo | <ul style="list-style-type: none"><li>• Dos clusters se fusionan si la <b>distancia máxima</b> entre ellos es menor o igual que una distancia umbral dada.</li><li>• Encuentra cliques en vez de elementos conectados.</li><li>• Clique: es un grafo máximo en el que hay un borde entre dos vértices.</li><li>• Complejidad: <math>O(n^2)</math></li><li>• Clusters más compactos.</li><li>• Farthest Neighbor Algorithm.</li></ul>   |
| MST Divisivo      | Jerárquico divisivo     | <ul style="list-style-type: none"><li>• Todos los elementos se colocan inicialmente en un cluster y se dividen repetidamente en dos hasta que todos los elementos estén en su propio cluster.</li><li>• La distancia entre elementos de los clusters es el criterio de separación.</li><li>• Se quitan los bordes del más grande a los más pequeños.</li></ul>   |

|                    |             |  |
|--------------------|-------------|--|
| MST<br>Particional | Particional | <ul style="list-style-type: none"> <li>• Inicia con un cluster, y a partir de ese va generando el número de clusters ingresado; es inverso al aglomerativo.</li> <li>• Distancia promedio entre clusters como criterio de calidad en los resultados.</li> <li>• Alto número de combinaciones de posibles soluciones.</li> <li>• Identifica bordes inconsistentes utilizando el peso (distancia) de un borde en comparación con aquellos cercanos a él.</li> <li>• Complejidad: <math>O(n^2)</math></li> </ul>  |
| Error al cuadrado  | Particional | <ul style="list-style-type: none"> <li>• En este algoritmo se usa la minimización del error cuadrático para determinar a qué cluster pertenece el punto. Esta técnica la usa el algoritmo K-Medias.</li> <li>• Error cuadrático = <math>(\text{real} - \text{estimado})^2</math></li> </ul>  |
| K-Means            | Particional | <ul style="list-style-type: none"> <li>• K-medias es un método de agrupamiento, que tiene como objetivo la partición de un conjunto de <math>n</math> observaciones en <math>k</math> grupos en el que cada observación pertenece al grupo cuyo valor medio es más cercano.</li> <li>• <math>O(n^{dk+1} \log n)</math>, donde <math>n</math> es el número de entidades a particionar</li> <li>• Utiliza la medida entre el K centro y sus <math>n</math> puntos.</li> <li>• Utiliza la medida Euclidiana o Manhattan</li> </ul>  |
| Nearest Neighbor   | Particional | <ul style="list-style-type: none"> <li>• Supervisado: esto -brevemente- quiere decir que tenemos etiquetado nuestro conjunto de datos de entrenamiento, con la clase o resultado esperado dada «una fila» de datos.</li> <li>• Basado en Instancia: Esto quiere decir que nuestro algoritmo no aprende explícitamente un modelo (como por ejemplo en Regresión Logística o árboles de decisión). En cambio memoriza las instancias de entrenamiento que son usadas como «base de conocimiento» para la fase de predicción.</li> <li>• Se utiliza en la resolución de multitud de problemas, como en sistemas de recomendación, búsqueda semántica y detección de anomalías.</li> <li>• Complejidad <math>O(kN^{(1-1/k)})</math></li> </ul> |
| PAM                | Particional | <ul style="list-style-type: none"> <li>• La técnica de <b>clustering de partición entorno a centroides (PAM)</b> realiza una distribución de los elementos entre un número prefijado de clústeres o grupos.</li> <li>• Esta técnica recibe como dato de entrada el número de clusters a formar además de los elementos a clasificar y la matriz de similitudes.</li> <li>• Explorar todas las posibles particiones es computacionalmente intratable.</li> <li>• Complejidad <math>O(k(n-k)^2)</math>.</li> </ul>   |

|             |             |   |
|-------------|-------------|---|
| CLARANS     | Particional | <ul style="list-style-type: none"> <li>• Agrupación de aplicaciones grandes basadas en la búsqueda Aleatoria (Clustering Large Applications based upon RANdomized Search)</li> <li>• Selecciona aleatoriamente k objetos en el conjunto de datos como los medoides actuales.</li> <li>• Luego selecciona aleatoriamente un medoides actual xy un objeto y que no es uno de los medoides actuales..</li> <li>• Complejidad de <math>k(n-k)</math>.</li> </ul>  |
| Bond Energy | Particional | <ul style="list-style-type: none"> <li>• Usado para determinar cómo agrupar datos y cómo almacenarlos físicamente en disco en función de su uso.</li> <li>• La afinidad entre los atributos se basa en el uso de ellos en conjunto.</li> <li>• Usa medida de similitud</li> <li>• Los atributos que son usados juntos crean un cluster y son almacenados juntos.</li> </ul>   |
| BIRCH       | Particional | <ul style="list-style-type: none"> <li>• Reducción iterativa equilibrada y agrupamiento mediante jerarquía.</li> <li>• Hace uso del algoritmo CF: Función de Cluster(cluster feature)</li> <li>• Se define por la terna <math>CF=(N,LS,SS)</math>.</li> <li>• No funciona bien para datos de naturaleza no esférica, es decir, que los grupos no son agrupados de forma circular.</li> <li>• Complejidad de <math>O(n)</math>.</li> <li>• Clasificar una gran cantidad de datos.</li> <li>• Requiere escanear solo una vez la BD.</li> <li>• Asume que hay un espacio limitado de memoria.</li> <li>• Es incremental y jerárquico.</li> </ul> |
| DBSCAN      | Mezclado    | <ul style="list-style-type: none"> <li>• Clusters con un tamaño y densidad mínimos.</li> <li>• Densidad: Número mínimo de puntos dentro de una cierta distancia el uno del otro.</li> <li>• Se ingresa el número mínimo de puntos en cualquier cluster.</li> <li>• La distancia Eps como distancia máxima para la medida de densidad.</li> <li>• El vecindario Eps es el conjunto de puntos dentro de la distancia Eps.</li> <li>• El número deseado de clusters lo determina el algoritmo mismo.</li> </ul>  |
| CURE        | Mezclado    | <ul style="list-style-type: none"> <li>• Buen manejo de valores atípicos</li> <li>• Es un algoritmo híbrido entre los dos enfoques jerárquico y particional</li> <li>• Se toma un número c de puntos representativos del grupo</li> <li>• Selecciona los c puntos más dispersos del cluster y los</li> </ul>  |

|      |                            |  |
|------|----------------------------|--|
|      |                            | comprimen hacia el centroide por un factor de contracción $\alpha$   |
| ROCK | Aglomerativo<br>categórico | <ul style="list-style-type: none"> <li>• Usa alguna técnica aglomerativa</li> <li>• Una medida de similitud determina el par de puntos que serán unidos.</li> <li>• El enlace entre <math>(k_i, k_j)</math> es el número de enlaces entre dos clusters.</li> <li>• Un par de puntos se consideran vecinos si su similitud está por encima de un umbral.</li> </ul> |

## Nuestro conjunto de datos.

Nuestro conjunto de datos es referente a instancias de distintos tipos de vinos obtenidos de Italia. Contiene 13 atributos y un total de 178 instancias.

Las instancias representan características de los vinos generadas a partir de un análisis. A continuación se presenta el diccionario de datos para auxiliar al lector con la comprensión de los datos que se tienen, así como la naturaleza y dominio de los mismos.

Los datos fueron obtenidos del repositorio <https://archive.ics.uci.edu/ml/datasets/Wine>

## Diccionario de datos.

| Campo                           | Significado  | Tipo de datos y dominio                                    |
|---------------------------------|--|--|
| <b>Alcohol (%)</b>              | Porcentaje de alcohol.   | Es un tipo de dato numérico con un rango de [11.03, 14.83] |
| <b>Malic acid (g/L)</b>         | Cantidad de ácido málico. Se identifica por el peculiar olor del vino.   | Es un tipo de dato numérico con un rango de [0.74, 5.8]    |
| <b>Ash (mg/L)</b>               | Cantidad de ceniza que queda después de la evaporación e incineración.   | Es un tipo de dato numérico con un rango de [1.36, 3.23]   |
| <b>Alcalinity of ash (mg/L)</b> | Nivel de alcalinidad en la ceniza para evaluar la composición mineral del vino.  | Es un tipo de dato numérico con un rango de [10.6, 30]     |
| <b>Magnesium (mg/L)</b>         | El sabor ácido de los vinos parece estar relacionado con la concentración de magnesio en el mismo.   | Es un tipo de dato numérico con un rango de [70, 162]      |
| <b>Total Phenols (mg/mL)</b>    | Cantidad total de componentes fenólicos (fenol y polifenoles naturales). Afectan el sabor, el color y la sensación en la boca del vino.  | Es un tipo de dato numérico con un rango de [0.98, 3.88]   |
| <b>Flavanoids (mg/mL)</b>       | Contenido fenólico que se encuentran en la piel de la uva; son antioxidantes, anticarcinogénicas y antiinflamatorias.  | Es un tipo de dato numérico con un rango de [0.34, 5.08]   |
| <b>Nonflavanoid Phenols</b>     | A veces, también denominado fenol o flavonoide, abarca alrededor de 4000 compuestos bioactivos de origen vegetal y fúngico que tienen más de un anillo de fenol aromático dentro de la estructura en oposición a los no flavonoides o fenoles simples. | Es un tipo de dato numérico con un rango de [0.13,0.66]    |
| <b>Proanthocyanins</b>          | Las proantocianidinas juegan un papel importante en el vino, con la capacidad de unirse a las proteínas salivales, estos taninos condensados influyen fuertemente en la astringencia percibida del vino.   | Es un tipo de dato numérico con un rango de [0.41,3.58]    |
| <b>Color Intensity</b>          | La intensidad del color se puede observar con la opacidad del vino. Los vinos tintos profundamente opacos se han destacado por tener más pigmento y fenólicos que los vinos tintos más translúcidos.   | Es un tipo de dato numérico con un rango de [1.28,13]      |

|                                     |  |   |
|-------------------------------------|--|---|
| <b>Hue</b>                          | Matiz que tiene el vino                      | Es un tipo de dato numérico con un rango de [0.48,1.71] |
| <b>OD280/OD315 Of Diluted Wines</b> | Capacidad de dilución del vino.              | Es un tipo de dato numérico con un rango de [1.27, 4]   |
| <b>Proline (mg/L)</b>               | Aminoácido que forma parte de las proteínas. | Es un tipo de dato numérico con un rango de [278, 1680] |

## Tratamiento de datos.

|                             |         |   |               |               |                    |
|-----------------------------|---------|---|---------------|---------------|--------------------|
| ✓ Alcohol                   | Real    | 0 | Min<br>11.030 | Max<br>14.830 | Average<br>13.001  |
| ✓ MalicAcid                 | Real    | 0 | Min<br>0.740  | Max<br>5.800  | Average<br>2.336   |
| ✓ Ash                       | Real    | 0 | Min<br>1.360  | Max<br>3.230  | Average<br>2.367   |
| ✓ AlcalinityOfAsh           | Real    | 0 | Min<br>10.600 | Max<br>30     | Average<br>19.495  |
| ✓ Magnesium                 | Integer | 0 | Min<br>70     | Max<br>162    | Average<br>99.742  |
| ✓ TotalPhenols              | Real    | 0 | Min<br>0.980  | Max<br>3.880  | Average<br>2.295   |
| ✓ Flavanoids                | Real    | 0 | Min<br>0.340  | Max<br>5.080  | Average<br>2.029   |
| ✓ NonflavanoidPhenols       | Real    | 0 | Min<br>0.130  | Max<br>0.660  | Average<br>0.362   |
| ✓ Proanthocyanins           | Real    | 0 | Min<br>0.410  | Max<br>3.580  | Average<br>1.591   |
| ✓ ColorIntensity            | Real    | 0 | Min<br>1.280  | Max<br>13     | Average<br>5.058   |
| ✓ Hue                       | Real    | 0 | Min<br>0.480  | Max<br>1.710  | Average<br>0.957   |
| ✓ OD280/OD315OfDilutedWines | Real    | 0 | Min<br>1.270  | Max<br>4      | Average<br>2.612   |
| ✓ Proline                   | Integer | 0 | Min<br>278    | Max<br>1680   | Average<br>746.893 |

Figura 1: Dominio de datos antes de estandarización

En algunos de los algoritmos de segmentación el hecho de no eliminar las unidades puede afectar directamente en su funcionamiento. Como se puede visualizar a través de la Figura 1, el dominio de los datos entre cada uno de los atributos varía mucho, generando conflictos para algoritmos de clustering que se basen en la distancia debido a que habrá atributos que dominen el estudio.



|                                    |         |   |          |            |                   |
|------------------------------------|---------|---|----------|------------|-------------------|
| ✓ <b>Id</b>                        | Integer | 0 | Min<br>1 | Max<br>178 | Average<br>89.500 |
| ✓ <b>Alcohol</b>                   | Real    | 0 | Min<br>0 | Max<br>1   | Average<br>0.519  |
| ✓ <b>MalicAcid</b>                 | Real    | 0 | Min<br>0 | Max<br>1   | Average<br>0.315  |
| ✓ <b>Ash</b>                       | Real    | 0 | Min<br>0 | Max<br>1   | Average<br>0.538  |
| ✓ <b>AlcalinityOfAsh</b>           | Real    | 0 | Min<br>0 | Max<br>1   | Average<br>0.459  |
| ✓ <b>Magnesium</b>                 | Real    | 0 | Min<br>0 | Max<br>1   | Average<br>0.323  |
| ✓ <b>TotalPhenols</b>              | Real    | 0 | Min<br>0 | Max<br>1   | Average<br>0.453  |
| ✓ <b>Flavanoids</b>                | Real    | 0 | Min<br>0 | Max<br>1   | Average<br>0.356  |
| ✓ <b>NonflavanoidPhenols</b>       | Real    | 0 | Min<br>0 | Max<br>1   | Average<br>0.437  |
| ✓ <b>Proanthocyanins</b>           | Real    | 0 | Min<br>0 | Max<br>1   | Average<br>0.373  |
| ✓ <b>ColorIntensity</b>            | Real    | 0 | Min<br>0 | Max<br>1   | Average<br>0.322  |
| ✓ <b>Hue</b>                       | Real    | 0 | Min<br>0 | Max<br>1   | Average<br>0.388  |
| ✓ <b>OD280/OD315OfDilutedWines</b> | Real    | 0 | Min<br>0 | Max<br>1   | Average<br>0.491  |
| ✓ <b>Proline</b>                   | Real    | 0 | Min<br>0 | Max<br>1   | Average<br>0.334  |

Figura 2: Dominio de datos después de estandarización

Para la solución del problema previamente descrito se realiza una estandarización de los datos con un rango entre 0 y 1 como se puede apreciar en la Figura 2. Tras este preprocesamiento se asegura que ninguno de los datos dominará el estudio y que algoritmos de clustering como es el caso de K-Means funcionarán de forma adecuada.

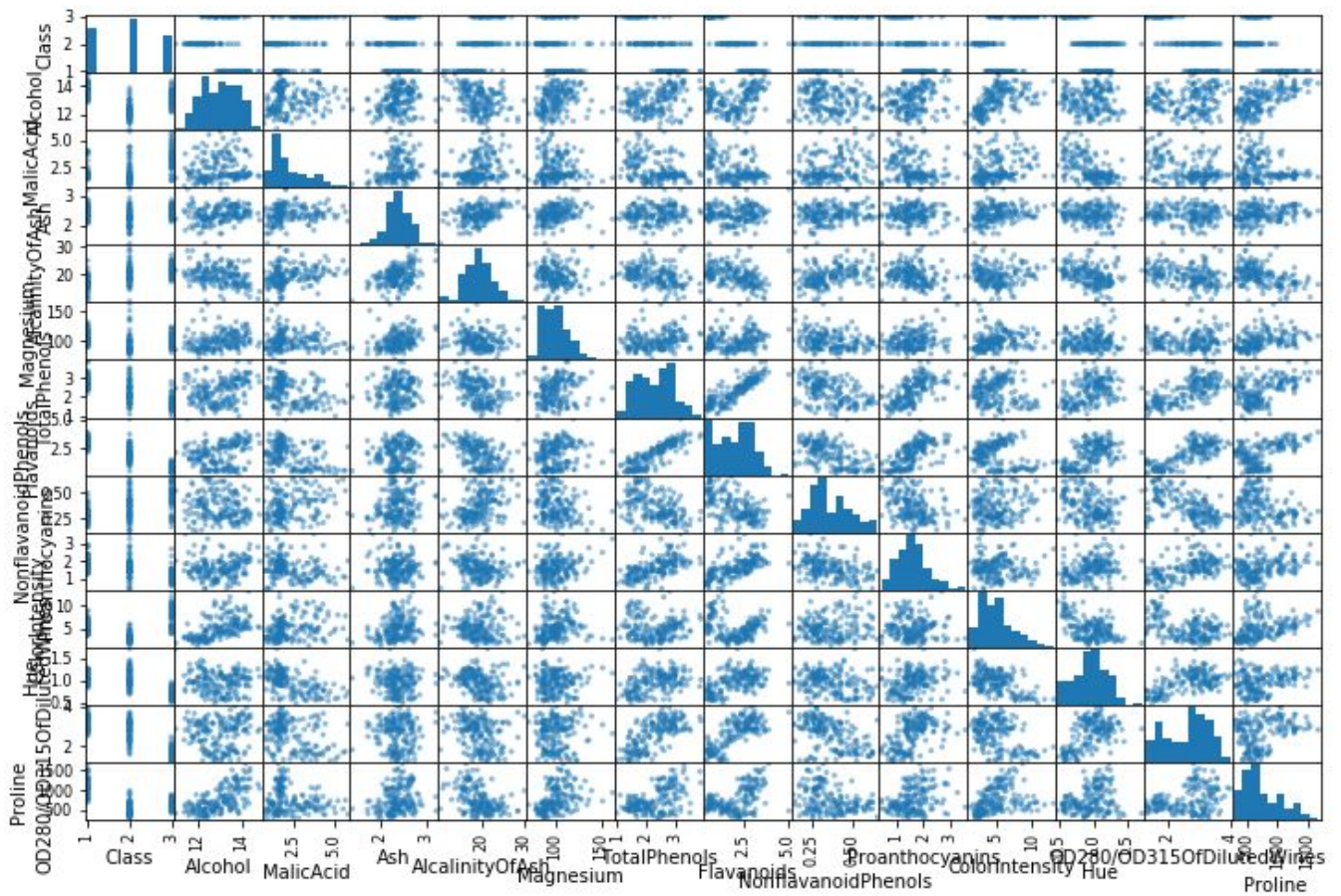
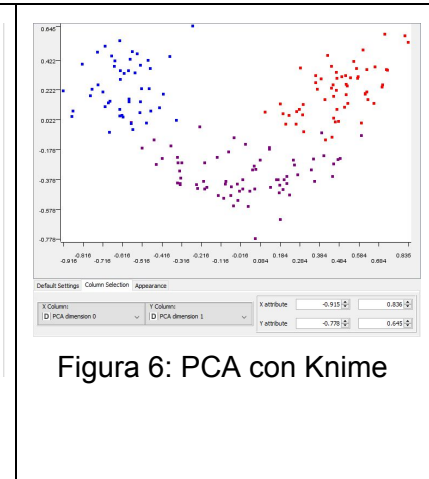
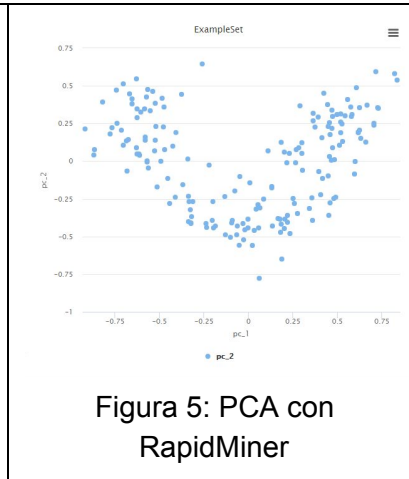
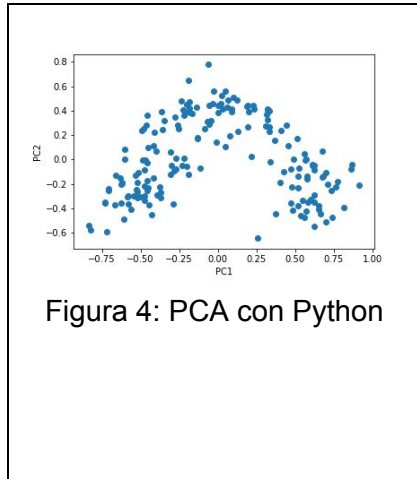


Figura 3: Matriz de dispersión obtenida con Python

El siguiente paso antes de aplicar los algoritmos consiste en determinar sobre qué valores es visible generar las agrupaciones. Como se aprecia en la figura 3, hay diversas opciones sobre las que se podrían hacer los agrupamientos considerando dos dimensiones. Sin embargo, si se aplica la segmentación únicamente sobre 2 atributos los demás atributos serán despreciados y se podría perder información importante de nuestro conjunto de datos. Para asegurar hacer el agrupamiento manteniendo la mayor cantidad de información se pueden aplicar técnicas de reducción dimensional como es el caso del análisis de componentes principales (PCA por sus siglas en inglés).



Para el caso que se está analizando se harán uso de los dos primeros componentes principales para trabajar de una forma más fácil con dos dimensiones. El resultado tras la aplicación de PCA es el mostrado en las figuras 4, 5 y 6 con las distintas herramientas.

Fácilmente se puede visualizar que la aplicación de algoritmos para agrupar es viable. No obstante, hasta el momento es desconocido el número de grupos o clases que se encuentran en el conjunto de datos en función de las características de cohesión y agrupamiento dentro de los clusters para asegurar grupos matemáticamente adecuados.

## Aplicación de clustering.

El siguiente paso a realizar consiste en la aplicación de los algoritmos de clustering para determinar cuál es la mejor opción. Para ello primero determinaremos el valor de K idóneo con el algoritmo de K-Means. Esto solo para tener un valor de referencia.

Para ello se hará uso del coeficiente de silueta que permite medir valores como la cohesión entre elementos de un cluster y separación entre los clusters. Es importante recordar que el coeficiente de silueta sólo funciona para algoritmos que funcionan con medida de distancia.

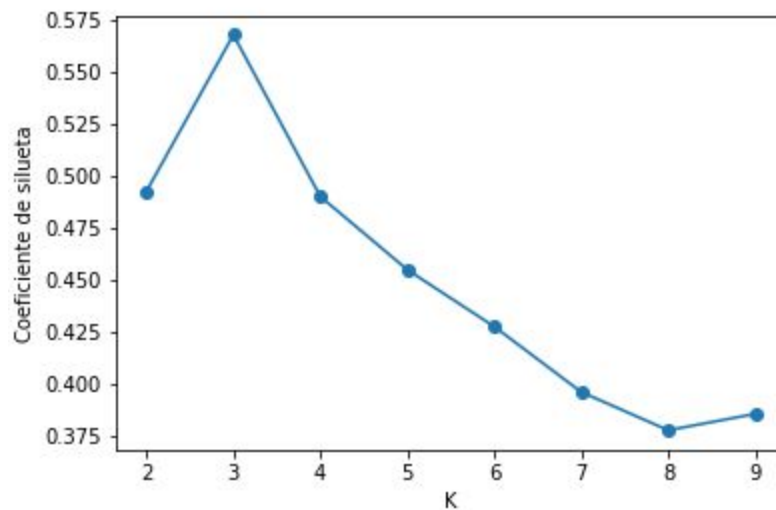


Figura 7: Coeficiente de silueta con K entre dos y nueve

El coeficiente de silueta es un valor entre menos uno y uno  $[-1,1]$ . El mejor valor de K es el que se aproxime más a uno. Auxiliándose de la figura 7 es fácil observar que el mejor valor de K para el algoritmo de K-Means es el tres. A partir de este punto, para los elementos que requieran un número de grupos esperados, se le asignará este valor(tres).

A continuación se muestran las pruebas realizadas con distintos algoritmos en las distintas herramientas.

|                             |  |
|-----------------------------|--|
| Herramienta                 | Python   |
| Algoritmo de clustering     | K-Means  |
| Parámetros de configuración | <p>Para utilizar el algoritmo de K-Means, se utiliza el método de la librería sklearn <b>KMeans()</b>, que recibe como parámetro el número de clusters <b>n_clusters</b> que se desean generar.</p> <pre> kmeans = KMeans(n_clusters=3).fit(X) data["KMEANSprediction"]=kmeans.fit_predict(X) plt.scatter(dfPCA["PC1"],             dfPCA["PC2"],             c=data["KMEANSprediction"],             cmap="rainbow",             alpha=0.8) graficarCentroides(kmeans.cluster_centers_) plt.ylabel("PC2") plt.xlabel("PC1") plt.savefig("KMeans.png") plt.show() </pre> |
| Explicación de resultados   | En la figura 8 se puede apreciar fácilmente el agrupamiento de los datos. No se muestra solapamiento entre los distintos grupos.   |

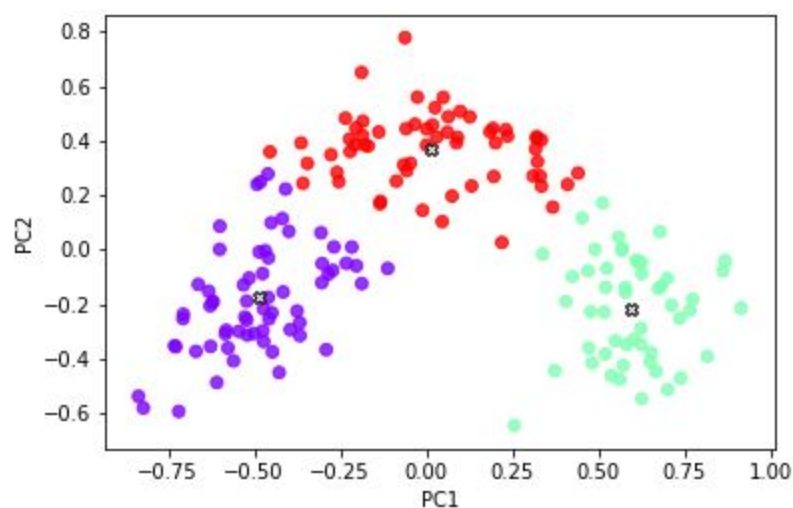


Figura 8: Clustering con K-means en Python

|   |  |
|---|--|
| Herramienta                               | Knime  |
| Algoritmo de clustering                   | K-Means  |
| Parámetros de configuración               |  |
| Explicación de resultados                 | En la figura 9 se observa la clasificación de los datos con un valor de $K = 3$ , el cual nos da como resultado una correcta división. |
|   |  |
| Figura 9: Clustering con K-means en Knime |  |



|                             |  |
|-----------------------------|--|
| Herramienta                 | RapidMiner   |
| Algoritmo de clustering     | K-Means  |
| Parámetros de configuración | K = 3  |
| Explicación de resultados   | En la figura 10 se observa la clasificación de los datos con un valor de K = 3, el cual nos da como resultado una correcta división. |

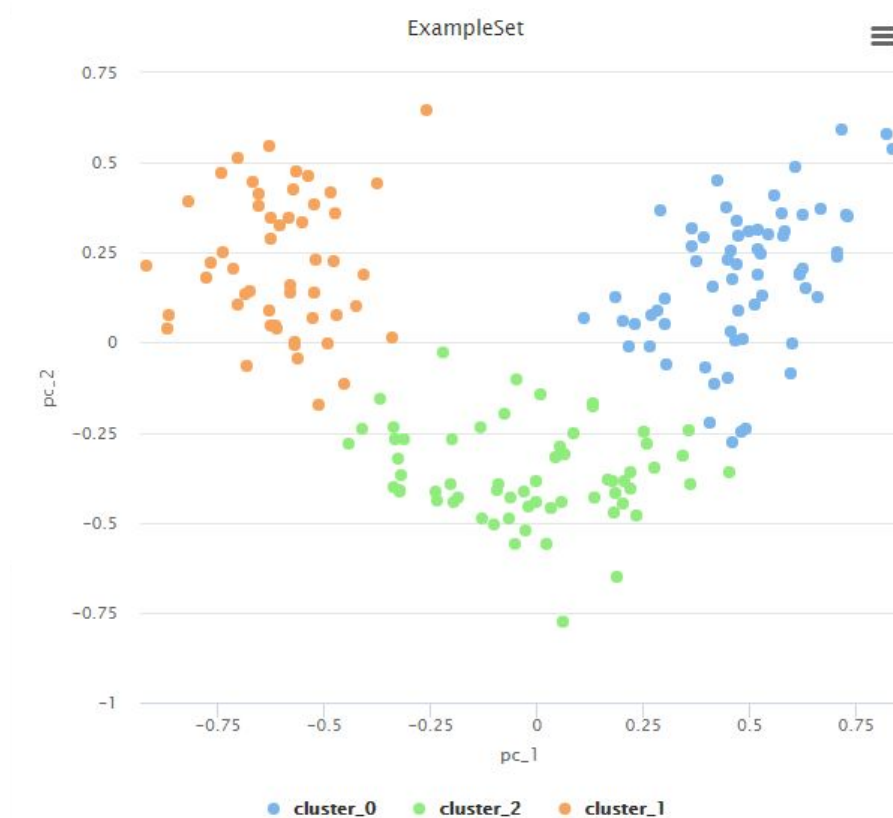


Figura 10: Clustering con K-means en RapidMiner

## Identificación de grupos creados

| Campo                             | Cluster 0        | Cluster 1           | Cluster 2           |
|-----------------------------------|------------------|---------------------|---------------------|
| <b>Alcohol (%)</b>                | 11.96-14.83      | 12.2-14.34          | 11.03-13.67         |
| <b>Malic acid (g/L)</b>           | 1.01-4.04        | 1.24-5.65           | 0.74-5.8            |
| <b>Ash (mg/L)</b>                 | 1.7-3.22         | 1.98-2.86           | 1.36 - 3.23         |
| <b>Alcalinity of ash (mg/L)</b>   | 11.2-30          | 16-27               | 10.6 - 28.500       |
| <b>Magnesium (mg/L)</b>           | 78-162           | 80-123              | 70-151              |
| <b>Total Phenols (mg/mL)</b>      | 2.2-3.88         | 0.98-2.8            | 1.1 - 3.5           |
| <b>Flavanoids (mg/mL)</b>         | 2.14-3.93        | 0.34-1.59           | 0.57 - 5.08         |
| <b>NonflavanoidPhenols</b>        | 0.13-0.5         | 0.17-0.63           | 0.14 - 0.66         |
| <b>Proanthocyanins</b>            | 1.25-3.28        | 0.55-2.7            | 0.41 - 3.58         |
| <b>ColorIntensity</b>             | 2.6-8.9          | 3.4-13              | 1.28 - 6            |
| <b>Hue</b>                        | 0.82-1.36        | 0.48-0.98           | 0.69 - 1.71         |
| <b>OD280/OD315OfDiluted Wines</b> | 2.51-4           | 1.27-2.47           | 1.67 - 3.69         |
| <b>Proline (mg/L)</b>             | 410-1680         | 372-880             | 278 - 870           |
| <b>PC1</b>                        | 0.112 - 0.836    | (-0.915) - (-0.257) | (-0.442) - 0.454    |
| <b>PC2</b>                        | (-0.277) - 0.594 | (-0.171) - 0.645    | (-0.778) - (-0.026) |

Esta clasificación se realizó de forma adecuada gracias a la aplicación de la reducción dimensional, de lo contrario agruparlos de forma correcta hubiera sido mucho más complicado. Esto puede ser asegurado debido a que el mismo dataset daba a conocer a priori que el número de clases era de tres. No obstante, sin el PCA el número de grupos máximos que se podían distinguir de forma adecuada eran dos ya que dos clases se intersectan en el mejor de los casos.

En la tabla superior se sombrearon las características que más auxilian a la segmentación adecuada de los grupos.



## Anexos

### Flujo de Knime

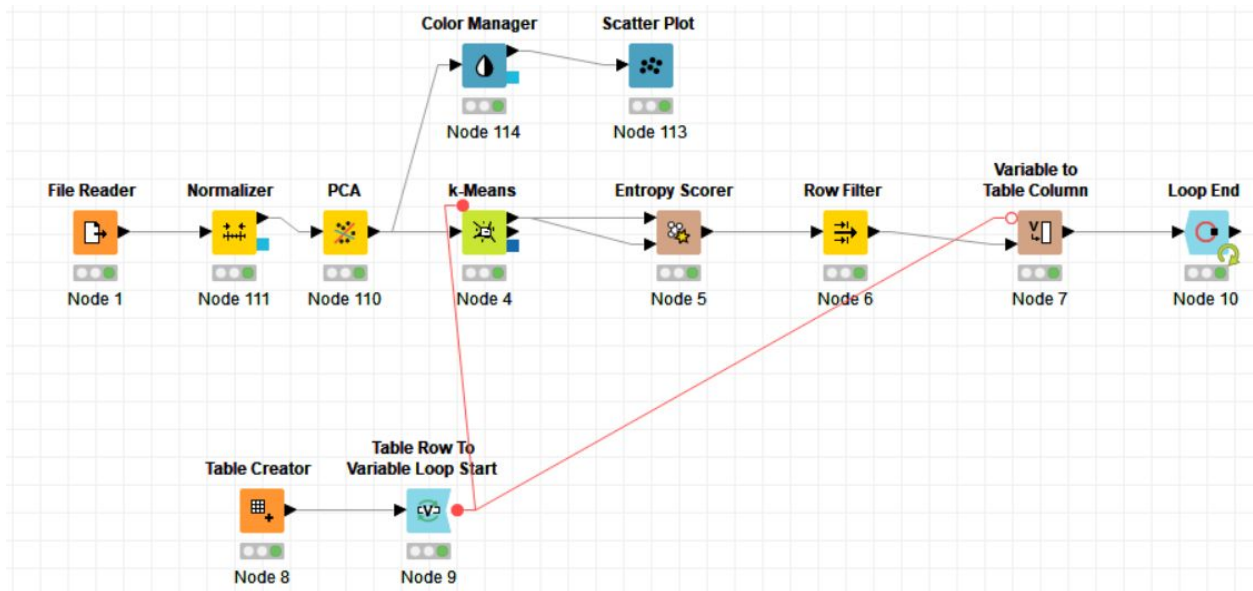


Figura 11: Flujo de Knime completo

### Flujo de RapidMiner

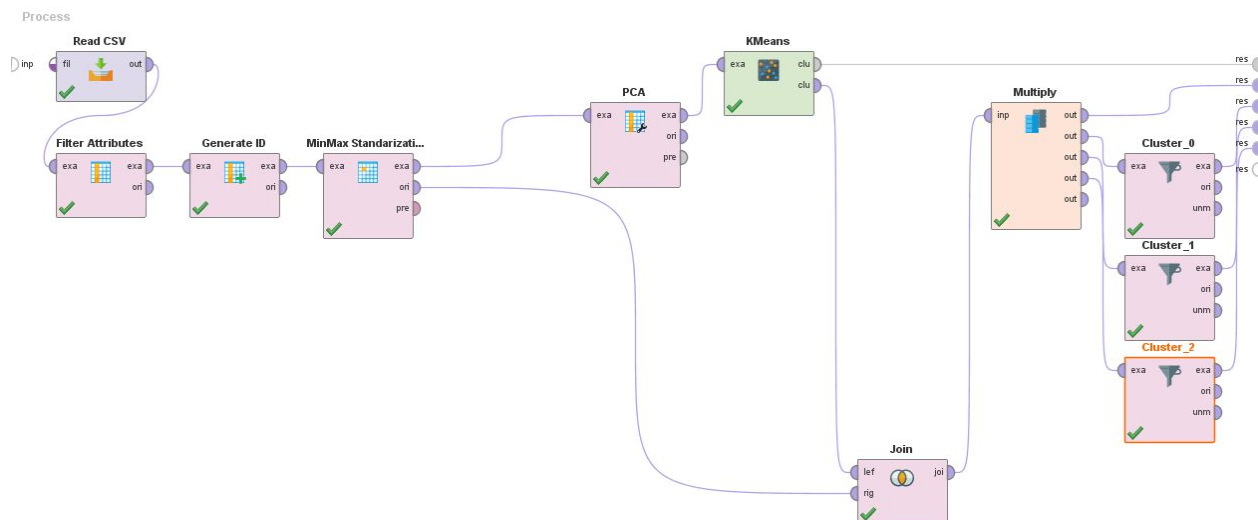


Figura 12: Flujo de RapidMiner completo

## Pruebas con otros algoritmos de Clustering

|                             |   |
|-----------------------------|---|
| Herramienta                 | Python  |
| Algoritmo de clustering     | Aglomerativo Complete-Link  |
| Parámetros de configuración | <p>Se utiliza el método de la librería sklearn <b>AgglomerativeClustering()</b>, que recibe dos parámetros: el número de clusters <b><i>n_clusters</i></b> que se desean generar, y el tipo de enlace <b><i>linkage</i></b> que va a usar como criterio de unión de clusters, en este caso <b><i>complete</i></b> o completo.</p> <pre>complete = AgglomerativeClustering(     n_clusters=3,     linkage = "complete").fit(X) data["CompletePrediction"]=complete.fit_predict(X) plt.scatter(dfPCA["PC1"],             dfPCA["PC2"],             c=data["CompletePrediction"],             cmap="rainbow",             alpha=0.8) plt.ylabel("PC2") plt.xlabel("PC1") plt.savefig("complete-link.png") plt.show()</pre> |

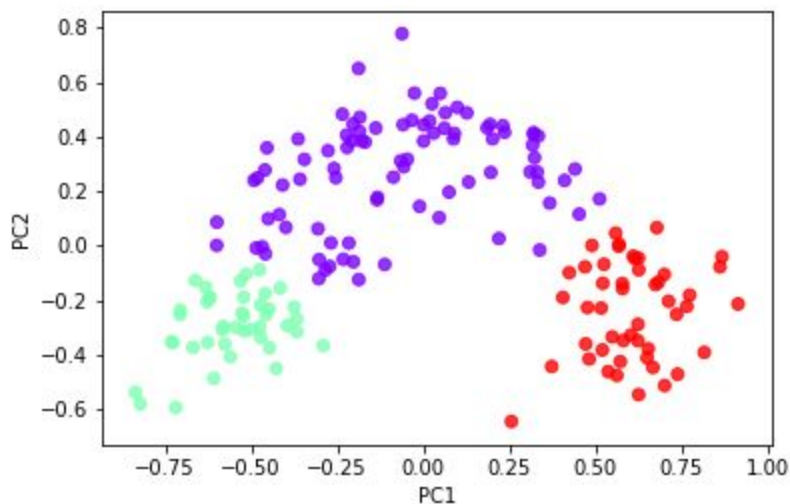


Figura 13: Clustering Aglomerativo con Complete-Link en Python

|                             |   |
|-----------------------------|---|
| Herramienta                 | Python  |
| Algoritmo de clustering     | Aglomerativo Average-Link   |
| Parámetros de configuración | <p>Se crea de forma casi idéntica al anterior, ahora cambiando el tipo de enlace en el atributo <b>linkage</b> por <b>average</b>.</p> <pre> average = AgglomerativeClustering(     n_clusters=3,     linkage = "average").fit(X)  data["AveragePrediction"] = average.fit_predict(X) plt.scatter(dfPCA["PC1"],             dfPCA["PC2"],             c=data["AveragePrediction"],             cmap="rainbow",             alpha=0.8) plt.ylabel("PC2") plt.xlabel("PC1") plt.savefig("average-link.png") plt.show() </pre> |

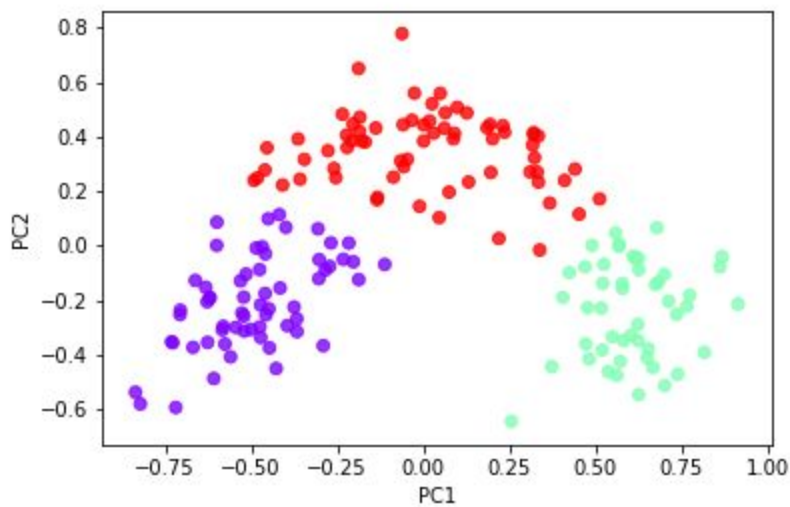


Figura 14: Clustering Aglomerativo con Average-Link en Python

|                             |  |
|-----------------------------|--|
| Herramienta                 | Python   |
| Algoritmo de clustering     | Birch  |
| Parámetros de configuración | <p>Para utilizar el algoritmo BIRCH, se utiliza el método de la librería sklearn <b>Birch()</b>, que recibe tres parámetros: el número máximo de subgrupos de CF en cada nodo <b>branching_factor</b>, el número de clusters que se desean generar <b>n_clusters</b> y un radio de umbral máximo entre el subcluster obtenido mediante la fusión de un nuevo punto y el subcluster más cercano <b>threshold</b>.</p> <pre> birch = Birch(branching_factor=30,               n_clusters=3,               threshold=0.17).fit(X) data["BIRCHPrediction"]=birch.fit_predict(X) plt.scatter(dfPCA["PC1"],             dfPCA["PC2"],             c=data["BIRCHPrediction"],             cmap="rainbow",             alpha=0.8) plt.ylabel("PC2") plt.xlabel("PC1") plt.savefig("birch.png") plt.show() </pre> |

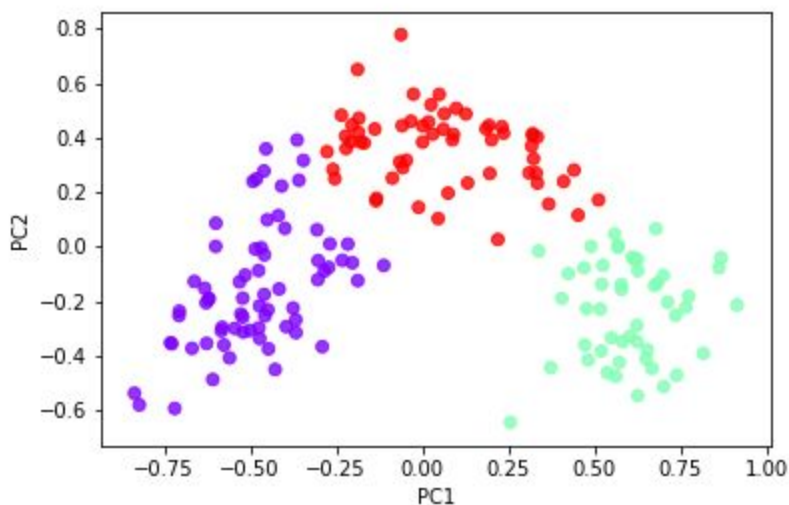


Figura 15: Clustering con Birch en Python

|                             |   |
|-----------------------------|---|
| Herramienta                 | Python  |
| Algoritmo de clustering     | DBSCAN  |
| Parámetros de configuración | <p>Para utilizar el algoritmo DBSCAN, se utiliza el método de la librería sklearn <b>DBSCAN()</b>, que recibe dos parámetros: la distancia de umbral máxima entre dos puntos para que uno se considere en la vecindad del otro (densidad) <b>eps</b>, y el número mínimo de puntos en cada cluster <b>min_samples</b>.</p> <pre> dbscan = DBSCAN(eps=0.14, min_samples=10).fit(X) data["DBSCANPrediction"]=dbscan.fit_predict(X) plt.scatter(dfPCA["PC1"],             dfPCA["PC2"],             c=data["DBSCANPrediction"],             cmap="rainbow",             alpha=0.8) plt.ylabel("PC2") plt.xlabel("PC1") plt.savefig("dbscan.png") plt.show() </pre> |
| Explicación de resultados   | <p>En la figura 16 se observa la clasificación de los datos haciendo uso de DBSCAN. Como se puede observar, con este algoritmo se encuentran cuatro grupos pero los grupos se intersectan careciendo de sentido el agrupamiento.</p>  |

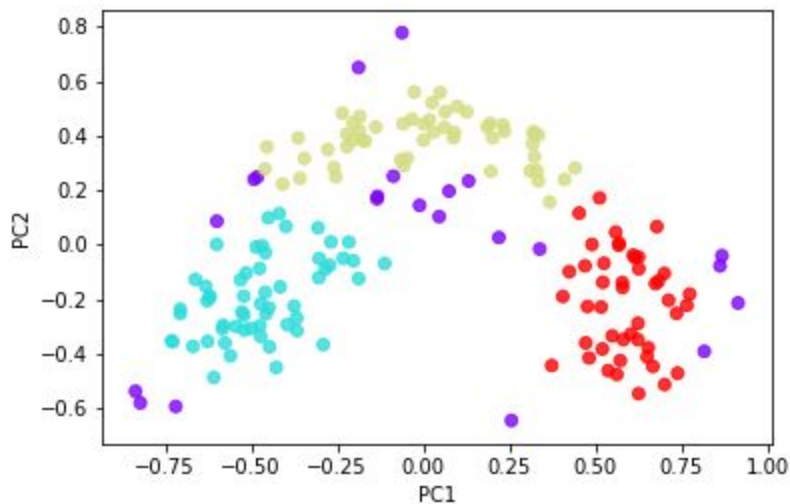


Figura 16: Clustering con DBSCAN en Python

|                             |  |
|-----------------------------|--|
| Herramienta                 | Knime  |
| Algoritmo de clustering     | K-Medoids  |
| Parámetros de configuración | <div><div><div>Dialog - 2:24 - k-Medoids</div><div><div>File</div><div><div>Default</div><div>Flow Variables</div><div>Memory Policy</div></div><div><div><div>Connected Distance Measure</div><div>Standard Distance (Euclidean/ Manhattan)<br/>for columns: "Class", "Alcohol", "MalicAcid", "Ash", ... 14 more&gt;</div></div><div><div>Partition count (k)</div><div>4</div></div><div><div>Chunk size</div><div>1,000</div></div><div><div><input checked="" type="checkbox"/> Constrain no. iterations</div><div>60</div></div><div><div><input type="checkbox"/> Use static seed</div><div><div></div><div>New Seed</div></div></div><div><div><input type="checkbox"/> Output relative distances to medoids</div></div><div><div><input checked="" type="checkbox"/> Choke on asymmetric distances</div></div></div><div><div><div><div></div><div>The "k" parameter is controlled by a variable.</div></div></div></div><div><div>OK</div><div>Apply</div><div>Cancel</div><div>?</div></div></div></div></div> |

0.645

0.422

0.222

0.022

-0.178

-0.378

-0.578

-0.778

-0.916

-0.816

-0.716

-0.616

-0.516

-0.416

-0.316

-0.216

-0.116

-0.016

0.084

0.184

0.284

0.384

0.484

0.584

0.684

0.836

Default Settings

Column Selection

Appearance

X Column:

D | PCA dimension 0

Y Column:

D | PCA dimension 1

X attribute

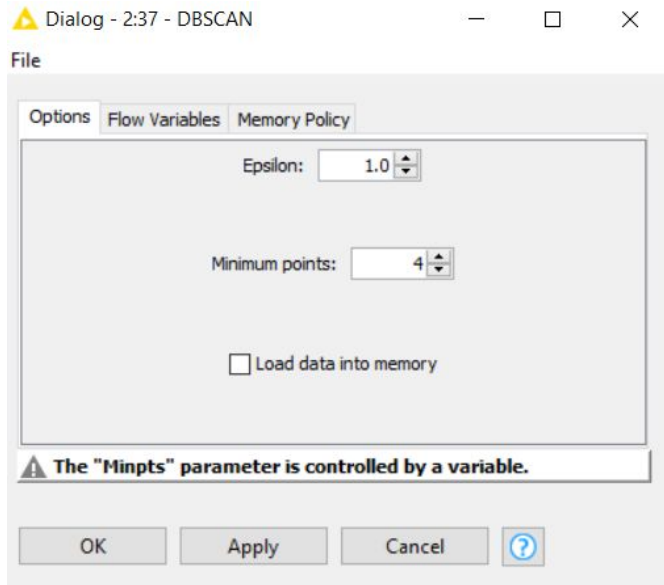
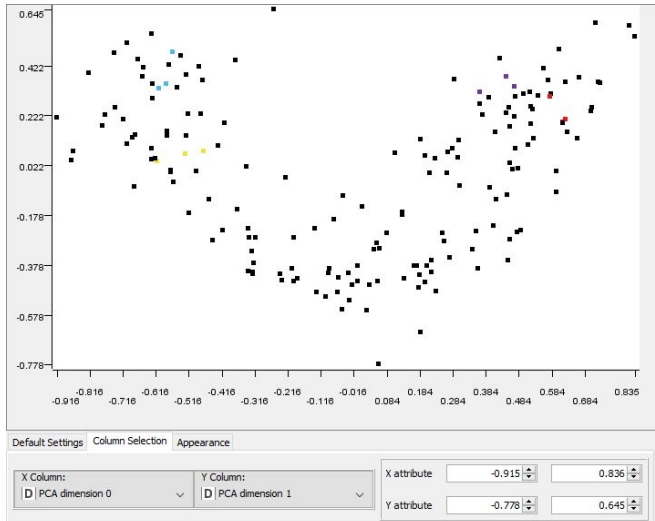
-0.915

0.836

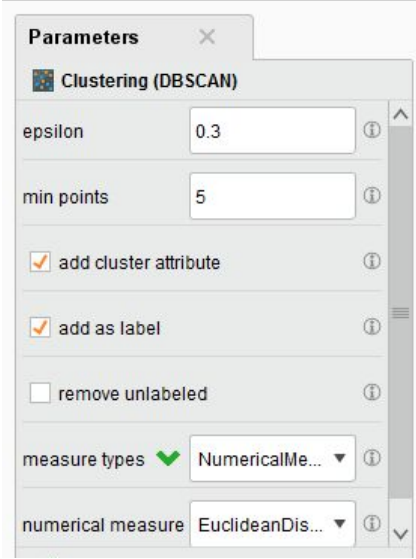
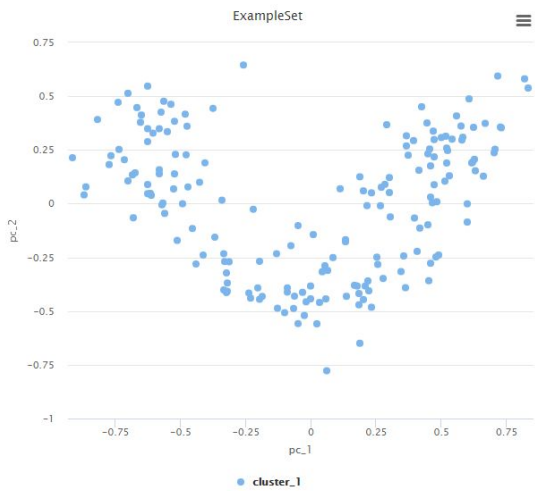
Y attribute

-0.778

0.645

|  |   |
|--|---|
| Herramienta  | Knime   |
| Algoritmo de clustering  | DBSCAN  |
| Parámetros de configuración  |   |
| Explicación de resultados  | <p>En la figura 18 se observa la clasificación de los datos con un valor de puntos mínimos=3, el cual nos da como resultado una división muy pobre y con demasiado ruido. Este valor escogido de puntos mínimos es el que mejor resultados tiene, aunque demuestra ser muy malo a comparación de otros algoritmos</p> |
|  |   |
| <p>Figura 18: Clustering con DBSCAN en Knime</p>                                     |   |



|  |   |
|--|---|
| Herramienta  | RapidMiner  |
| Algoritmo de clustering  | DBSCAN  |
| Parámetros de configuración  |   |
| Explicación de resultados  | <p>En la figura 19 se observa la clasificación de los datos haciendo uso de DBSCAN. Como se puede observar, con este algoritmo no se logró encontrar más de un cluster con esta herramienta por lo que fue descartado después de varios intentos.</p> |
|  |   |
| <p>Figura 19: Clustering con DBSCAN en RapidMiner</p>                                |   |