Clustering sobre accidentes de tráfico

Sergio Carrasco Márquez

Correo: sergiocmarq@gmail.com

Contenido

1 Introducción

Se tiene un conjunto de datos de 89519 accidentes de tráfico sucedidos en España durante el año 2013. Se va a proceder a aplicar varias técnicas de clustering para poder extraer patrones de accidentes poder clasificarlos en función del tipo, descubriendo así factores en dichos accidentes que estén relacionados. En el conjunto de datos se ofrecen una serie de variables, algunas numéricas y otras continuas, que se usarán para aplicar clustering en diversos casos de investigación distintos. Las variables categóricas nos ayudan a definir el caso de investigación y las variables numéricas son las usadas para la búsqueda de clústeres. La variable de visibilidad aparece como una variable categórica, pero puede interpretarse como una variable numérica, ya que se puede ordenar de menor a mayor visibilidad, por lo que se ha usado en el conjunto de variables para la búsqueda de clústeres. Dicho conjunto de variables está definido por la cantidad de heridos graves, leves, cantidad de afectados, vehículos implicados, fallecidos y la visibilidad en el accidente. Para la correcta ejecución de los algoritmos todas las variables usadas deben estar normalizadas.

Los algoritmos utilizados serán KMeans, AgglomerativeClustering, MeanShift, SpectralClustering y Birch disponibles en la biblioteca de scikit-learn de python

2 Casos de estudio

2.1 Estudio de accidentes en autovías

En el primer caso de estudio se van a analizar solo los accidentes ocurridos en autovía, ya que es un filtro muy genérico y se pueden encontrar varios tipos de accidentes. Esto nos deja un conjunto de accidentes de 9371 accidentes. Como al usar esa cantidad de datos el computador en el que se están realizando los experimentos no es capaz de ejecutar todos los algoritmos se ha usado un muestreo aleatorio del conjunto de datos de 3500 entradas.

En una primera aproximación los parámetros de los algoritmos se han ajustado de la siguiente manera:

- KMeans: init='k-means++', n_clusters=4, n_init=5
- AgglomerativeClustering: n_clusters=4
- Birch: n_clusters=4, threshold=0.4
- SpectralClustering: n_clusters=4

Los resultados obtenidos quedan reflejados en la siguiente tabla.

Algoritmo	tiempo de ejecución	Número de clusters	Silhouette	Calinski-Harabasz
KMeans	0.673391819	4	0.475450723	2984.416676
AgglomerativeClustering	1.053525209	4	0.480153431	2497.540562
SpectralClustering	17.47801137	4	0.461050107	2687.467114
Birch	0.198131323	4	0.539748662	2097.093875
MeanShift	20.47558236	6	0.500153895	1003.073395

Tabla 1: Resultados primer experimento en el primer caso de estudio

De forma más visual la comparación entre los índices de puntuación Silhouette y Salinski-Harabasz quedan de la siguiente forma.

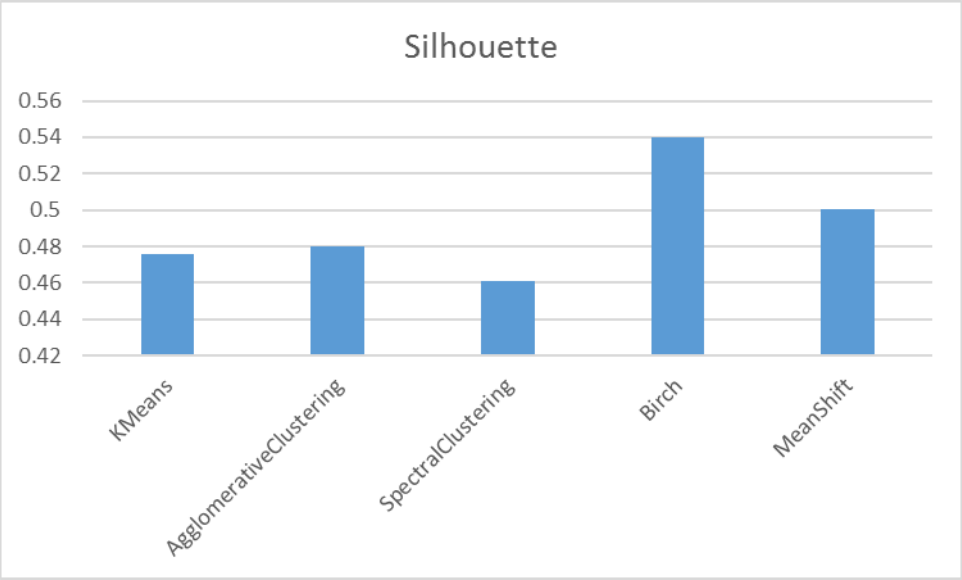

Ilustración 1: índice de Silhouette para el primer experimento del primer caso

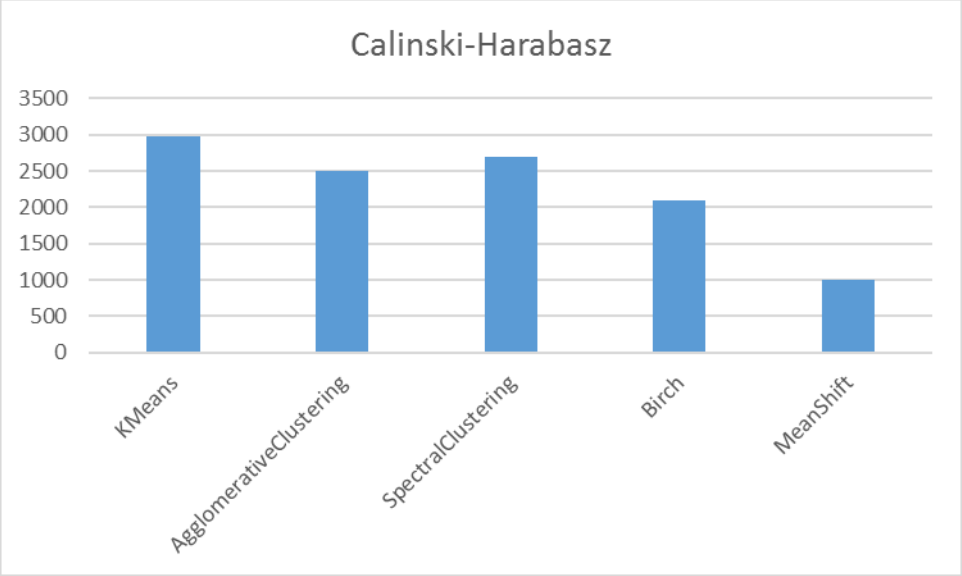

Ilustración 2: Índice de Calinski-Harabasz en el primer experimento del primer caso

En cuanto a la medida de silhouette, todos los algoritmos ofrecen medidas similares entre 0.46 y 0.54, siendo los algoritmos con mejor resultado Birch y MeanShift. En la gráfica del índice de Calinski-Harabasz, precisamente los algoritmos que obtenían un mejor valor en el índice de silhouette son los más perjudicados. Kmeans y Spectral Clustering ofrecen los mejores resultados en base a esta medida.

La elección de parámetros ha sido una primera aproximación, por lo que puede ser interesante ver la evolución de los algoritmos si se modifican los parámetros. En concreto la modificación del parámetro de número de clústeres que deben ser detectados es el que aparentemente puede influir más en los resultados. El algoritmo de Meanshift, que no necesita que se fije el

número de clústeres a priori, encuentra 6 clústeres y obtiene un buen resultado en la métrica de Silhouette, por lo que una aproximación podría ser elevar hasta 6 el número de clústeres del resto de algoritmos. Al ser 6 un número cercano al número de clústeres aproximado anteriormente, se va a aumentar hasta 10 el número de clústeres.

Algoritmo	tiempo de ejecución	Número de clusters	Silhouette	Calinski-Harabasz
KMeans	0.059039354	10	0.616190768	2910.689672
AgglomerativeClustering	0.522349596	10	0.629951543	2773.942441
SpectralClustering	3.598388195	10	0.572015035	2244.229399
Birch	0.242160559	10	0.469766301	1671.550335
MeanShift	11.99796271	6	0.500153895	1003.073395

Tabla 2: Resultados obtenidos con el número de clusters elevado hasta 10

El índice de Silhouette mejora notablemente al incrementar el número de clusters. La siguiente gráfica muestra las diferencias obtenidas entre ambas ejecuciones.

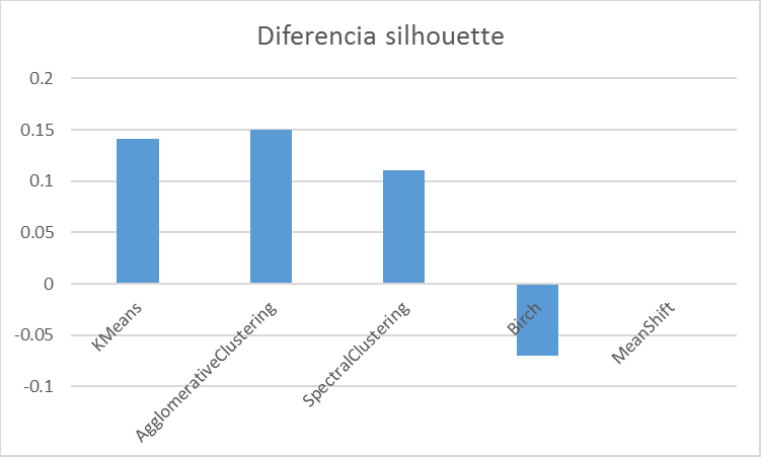

Ilustración 3: Diferencias entre los índices de Silhouette

Tanto Kmeans, Agglomerative y Spectral ganan en el índice de Silhouette, salvo Birch que experimenta una bajada, aunque no demasiado brusca. Si nos fijamos en las diferencias entre los resultados al índice de Calinski se aprecia lo siguiente.

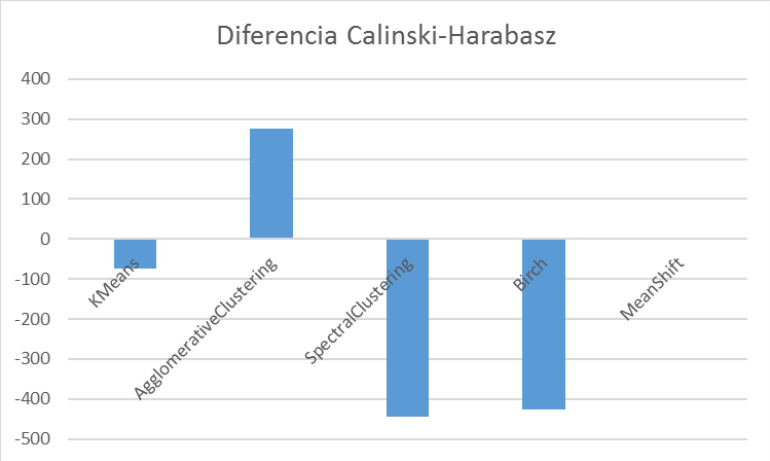

Ilustración 4: Diferencias entre los índices de Calinski-Harabasz

Todos los algoritmos parecen mostrar un peor resultado en este índice salvo el Agglomerative Clustering, que consigue un aumento considerable.

Otra manera de visualizar los resultados es fijarse directamente en el scatter matrix ofrecido por el modelo sobre el conjunto de datos. De esta forma se pueden apreciar que variables afectan conforman las características de cada clúster y se puede empezar a interpretar el resultado obtenido. Como las métricas obtenidas no fijan de forma unívoca a un algoritmo concreto como mejor que los demás, se va a mostrar el scatter matrix del algoritmo con mejor resultado en cada métrica para hacer una comparación más exhaustiva.

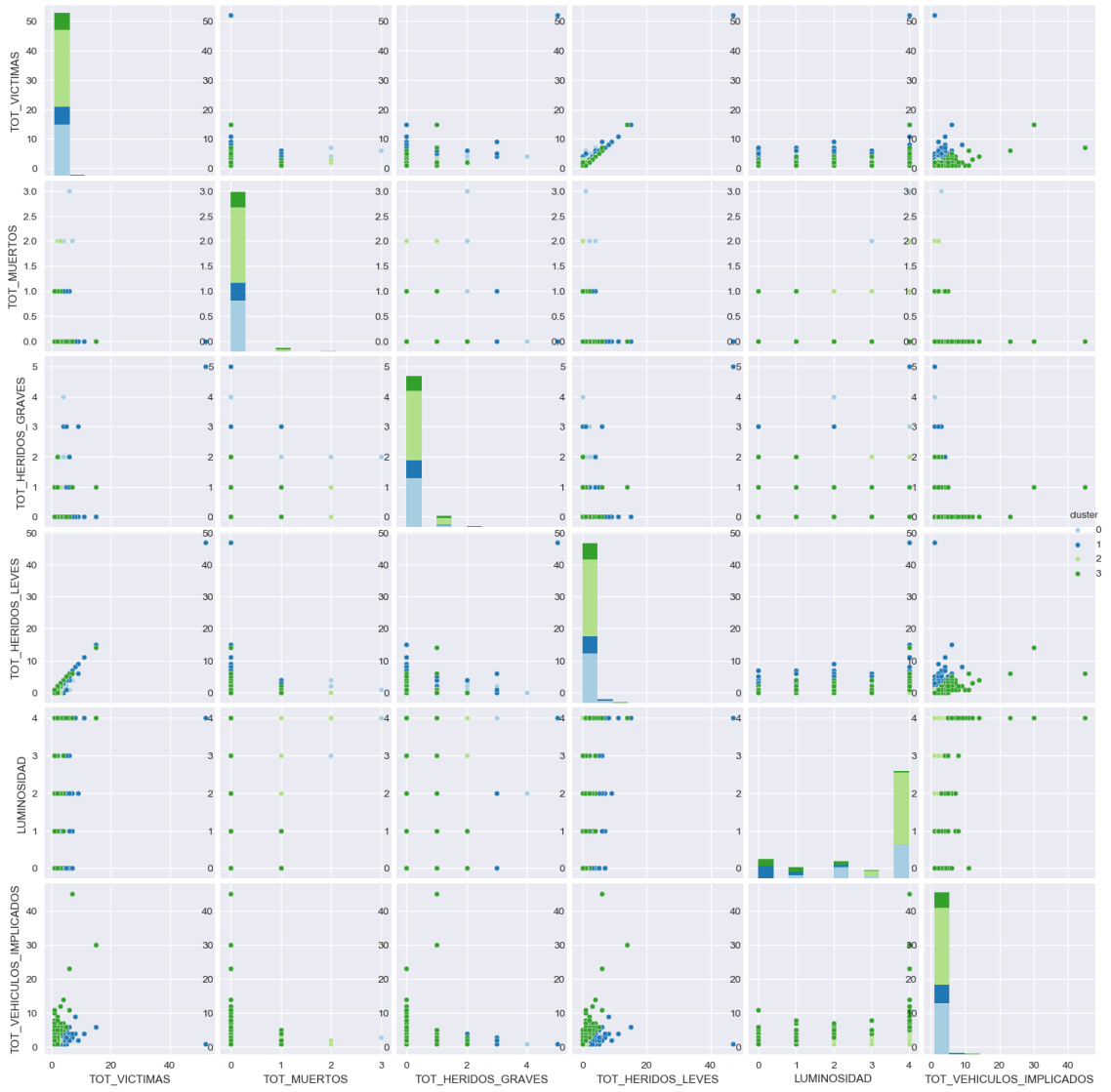

Ilustración 5: Scatter matrix Kmeans

En este caso no se puede extraer demasiada información, pero si se puede concluir que por ejemplo para el clúster 1, etiquetado de color verde claro, el número de vehículos implicado es bajo, al igual que el total de víctimas. Del mismo modo, el clúster 3, etiquetado de color verde oscuro se caracteriza por un número de víctimas bajo, pero el número de vehículos puede ser más alto. Cuando la visualización de los resultados no ofrece una respuesta clara, es posible sacar conclusiones de otra manera. Se puede extraer el valor medio de cada variable en cada uno de los clústeres.

Cluster	VICTIMAS	MUERTOS	HERIDOS_GRAVES	HERIDOS_LEVES	LUMINOSIDAD	VEHICULOS_IMPLICADOS
0	2.214421252	0.037950664	0.110056926	2.066413662	0.362428843	1.920303605
1	2.328236493	0.014271152	0.047910296	2.266055046	3.402650357	1.78695209
2	1.029925187	0.026184539	0.103491272	0.900249377	3.927680798	1.631546135
3	1.371134021	0.028350515	0.097938144	1.244845361	2.278350515	3.453608247

Tabla 3: Media de las variables para Kmeans

Con esta tabla se tiene una visión más global de los valores. Por ejemplo, la media de víctimas mortales es por lo general muy baja, por lo que se puede deducir que la dicha variable no ha influido a la hora de generar los clústeres. Sí que se observa que el clúster 3 tiene una media de vehículos implicados bastante más alta que la de los demás, lo que indica que esta variable puede ser importante para definir al clúster. Al fijarnos en la scatter matrix se observa que los accidentes con mayor número de vehículos implicados están bajos este clúster, pero también aparecen accidentes con un número de accidentes similar a los demás. Aparentemente el clúster 3 está formado por los accidentes con un número de vehículos relativamente alto, a la vez que el número de víctimas es bajo. El clúster 1 agrupa justamente al contrario, agrupa accidentes con muy pocos vehículos y con un valor de víctimas algo mayor.

Para el caso del algoritmo de Birch el scatter matrix es el siguiente.

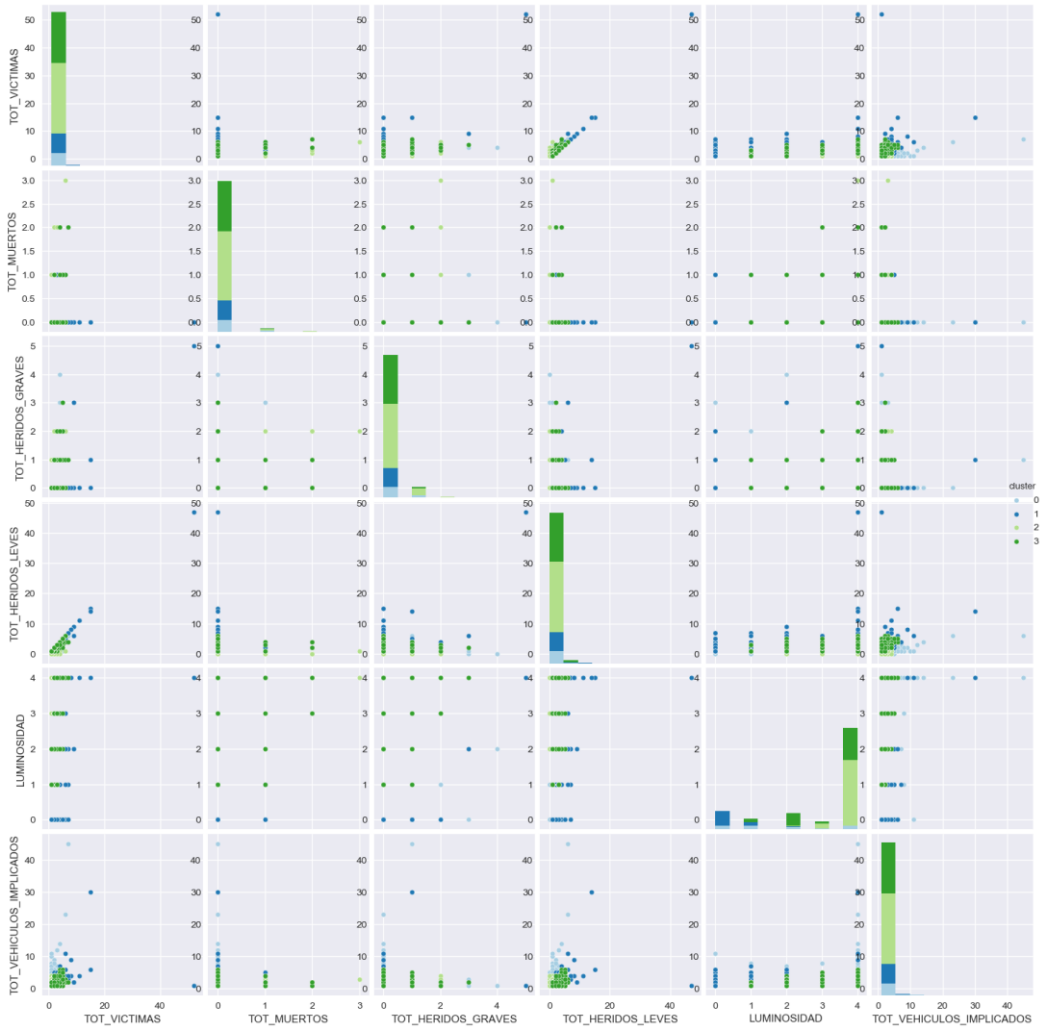

Ilustración 6: Scatter matrix de Birch

En cuanto al algoritmo de Birch, no hay mucha diferencia entre la visualización obtenida. Se aprecia que el clúster 3 se concentra cuando el número de víctimas es bajo y el de vehículos también. El clúster 0 tiende a aglomerar a los accidentes con pocas víctimas y un número mayor de accidentes. El clúster 1 parece tener en cuenta que la luminosidad sea baja y que el número de heridos leves aumente. El clúster 2 agrupa a los accidentes con un número de víctimas mortales mayor.

2.2 Estudio de accidentes sucedidos en carreteras con atropellos a animales sueltos

En este segundo caso de estudio se van a analizar los accidentes ocurridos en carreteras que impliquen atropellos a animales sueltos. El hecho de considerar estos accidentes, que tienen un carácter muy concreto, puede producir previsiblemente un número menor de clústeres, por lo que usando 4 clústeres como en el caso anterior se espera un mejor resultado en las métricas.

Los parámetros de los algoritmos se han ajustado de la misma forma que los del primer experimento del caso anterior. Tras la ejecución se obtienen los siguientes resultados.

Algoritmo	tiempo de ejecución	Número de clusters	Silhouette	Calinski-Harabasz
KMeans	0.014008522	4	0.674138497	992.4761422
AgglomerativeClustering	0.011007547	4	0.68091017	854.6193615
SpectralClustering	0.073048353	4	0.699571923	889.5339372
Birch	0.021011591	4	0.497134109	558.3380853
MeanShift	0.550366879	11	0.665740761	600.3061732

Tabla 4: Resultados del primer experimento del segundo caso de estudio

En cuanto al índice de Silhouette se ha producido la mejora esperada, dejando como ganador al spectral clustering como mejor algoritmo. Salvo el algoritmo de Birch, el resto no distan mucho de spectral clustering. En la siguiente gráfica se muestran los resultados de esta medida de forma visual.

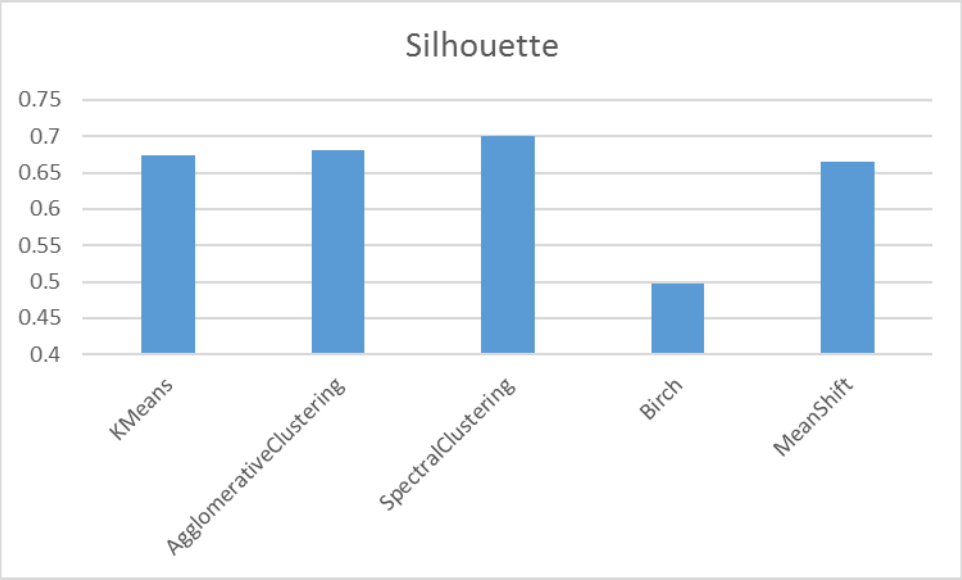

Ilustración 7: Índice de Silhouette para el primer experimento del segundo caso de estudio

La siguiente visualización corresponde al índice de Calinski-Harabasz obtenido en cada algoritmo.

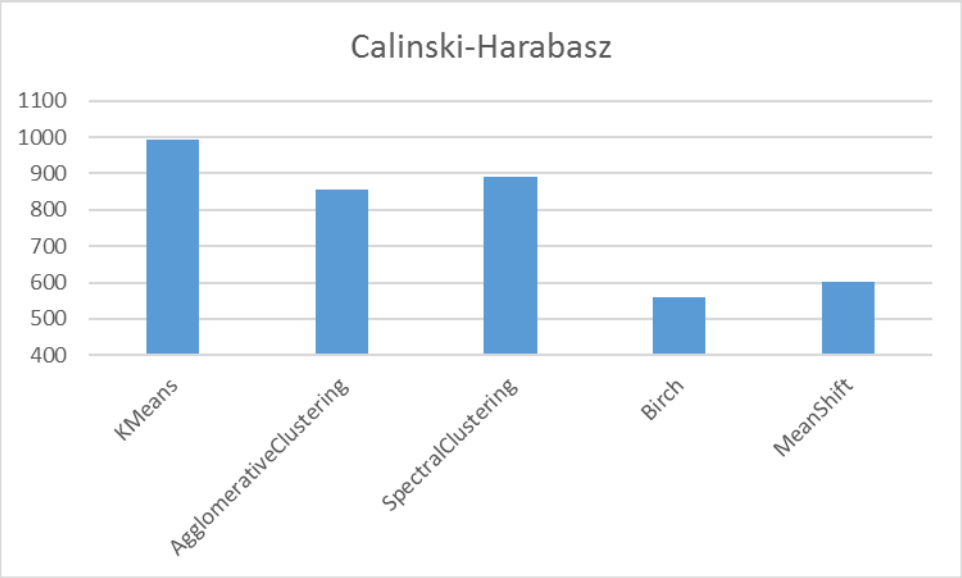

Ilustración 8: Índice de Calinski-Harabasz para el primer experimento del segundo caso de estudio

En cuanto a la métrica de Calinski, KMeans saca una clara venjata junto a spectral clustering. Birch vuelve a ser el más perjudicado junto a Meanshift. En general spectral y Kmeans son los algoritmos que ofrecen mejores resultados.

Al igual que antes se pueden ajustar los parámetros para conseguir mejores resultados. En este caso Meanshift da una clasificación de 11 clústeres. Si se fijan el número de clústeres a 11 para el resto de algoritmos los resultados son los siguientes.

Algoritmo	tiempo de ejecución	Número de clusters	Silhouette	Calinski-Harabasz
KMeans	0.027017832	11	0.804384399	1274.116702
AgglomerativeClustering	0.013008595	11	0.804789312	1220.286928
SpectralClustering	0.118078232	11	0.789618321	1066.579436
Birch	0.021015406	11	0.573432934	451.0273835
MeanShift	0.645545721	11	0.665740761	600.3061732

Tabla 5: Resultados del segundo caso de estudio con 11 clústeres

Al igualar el número de clústeres, se mejora bastante la calidad de los resultados de los algoritmos. Las siguientes gráficas muestran la evolución del comportamiento de los algoritmos al cambiar el número de clústeres.

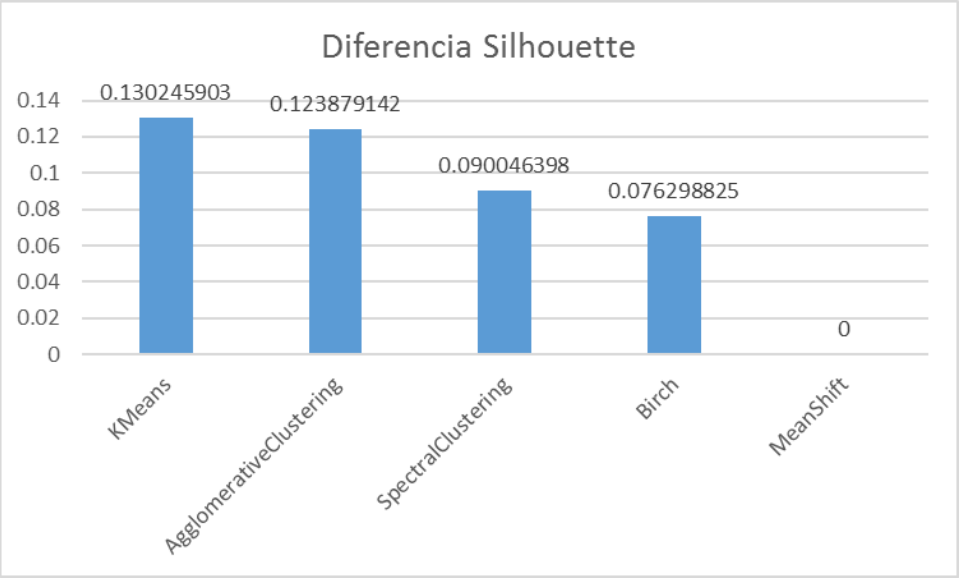

Ilustración 9: Diferencias entre los índices de Silhouette

Kmeans es el algoritmo que más se beneficia de aumentar el número de clústeres del conjunto de datos, seguido del agglomerative. En cuanto al índice de Calinski-Harabasz las diferencias son las siguientes.

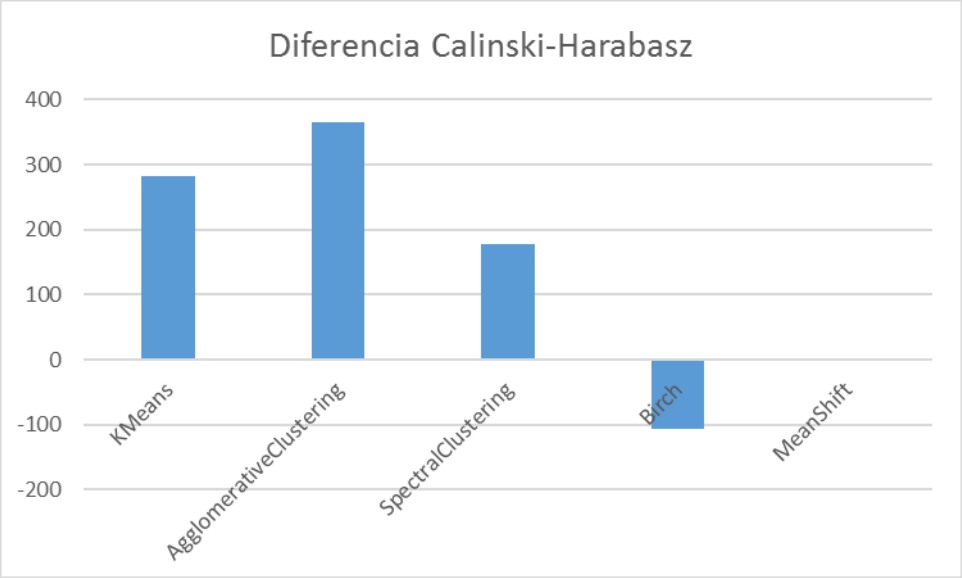

Ilustración 10: Diferencias entre los índices de Calinski-Harabasz

Agglomerative tiene una mejor ganancia en esta métrica, junto a Kmeans. Aumentar el número de clústeres hasta 11 ha resultado mejorar la calidad de los mismos en los algoritmos de Kmeans y agglomerative. Birch destaca por tener pérdida en la medida en logar de ganarla, por lo que entre la leve mejora en el índice de Silhouette y la pérdida en Calinski-Harabasz, aumentar el número de clústeres hasta 11 no parece ser una buena opción para este algoritmo.

La visualización de mapa de calor puede ser útil en este caso, ya que las métricas del agglomerative han sido bastante buenas. Dicha visualización se muestra a continuación.

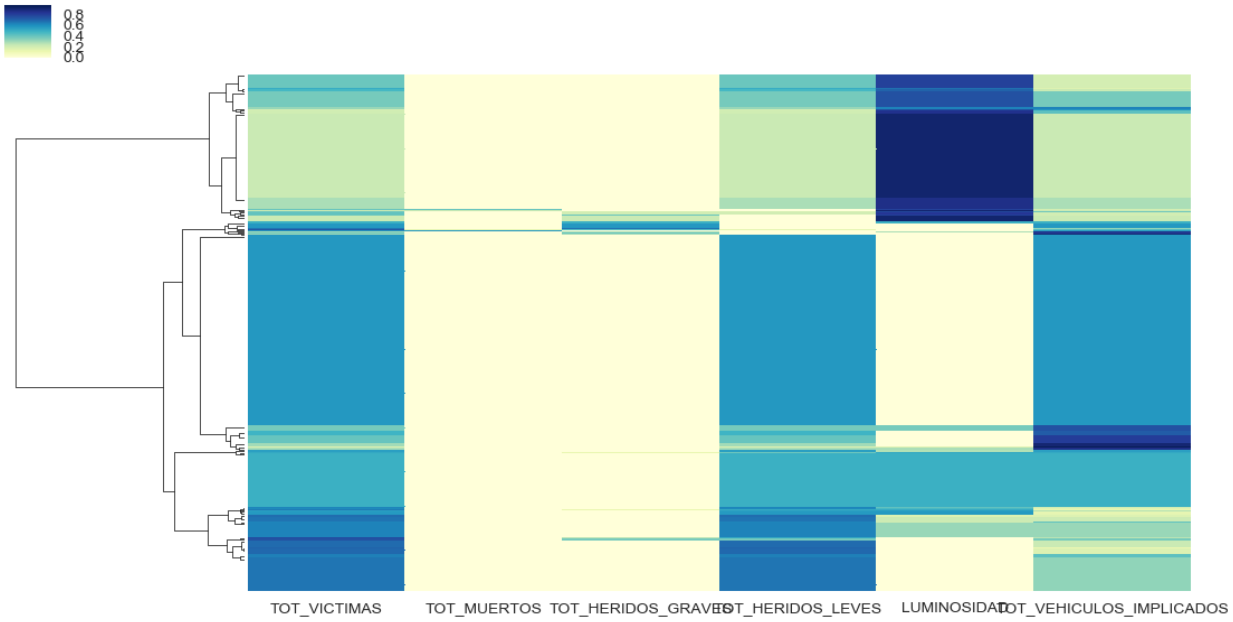

Ilustración 11: Visualización del mapa de calor

En la imagen superior se pueden ver los clústeres formados y el valor de las variables que lo caracterizan en función de color. A grandes rasgos se ve que el número de heridos graves en la mayoría de los clústeres es muy bajo, al igual que el total de muertos. Existe un clúster que abarca una gran cantidad de nodos que tiene un valor de víctimas algo elevado al igual que el valor de vehículos implicados. Se puede apreciar que la visibilidad es un factor a tener en cuenta, los clústeres con una mayor visibilidad tienen un número de víctimas menor.

En cuanto a la visualización del algoritmo de KMeans se obtiene la siguiente scatter matrix y tabla de valores medios de cada variable.

Cluster	VICTIMAS	MUERTOS	HERIDOS_GRAVES	HERIDOS_LEVES	LUMINOSIDAD	VEHICULOS_IMPLICADOS
0	1.147058824	0.058823529	0.441176471	0.647058824	0.352941176	2.117647059
1	1.595505618	0	0.02247191	1.573033708	1.337078652	1.08988764
2	1.401574803	0	0.011811024	1.38976378	0.015748031	1.070866142
3	1.145833333	0.027777778	0.076388889	1.041666667	3.715277778	1.048611111

Tabla 6: Medias de las variables de KMeans en el segundo caso de estudio

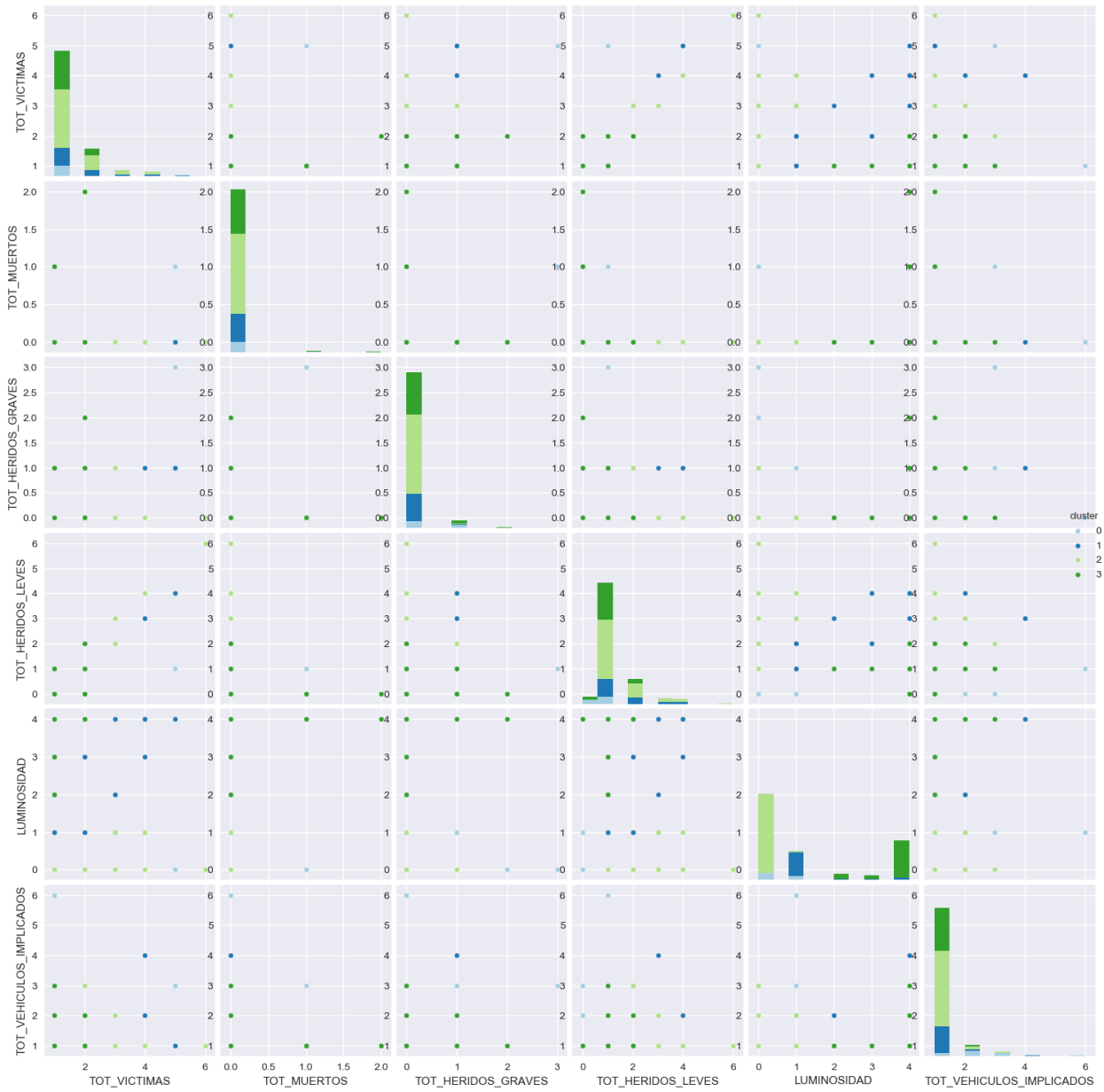

Ilustración 12: Scatter matrix de KMeans en el segundo caso de estudio

La luminosidad juega un papel importante en la clasificación de accidentes. Los accidentes del clúster 3 tienen mayor luminosidad, mientras que los del clúster 2 un valor bajo. El total de víctimas distingue al clúster 1 del 3, que tiene un mayor valor. El clúster 0 es el correspondiente a los accidentes más graves, con muertos ocurridos cuando la visibilidad es baja. Los accidentes del clúster 3 son también de mayor gravedad, pero la iluminación es alta.

2.3 Estudio de accidentes con lluvia

En los casos anteriores se ha tenido en cuenta para su elección el número de clúster previsibles, primero accidentes en autovías y posteriormente accidentes en carreteras con atropellos a animales sueltos. En este caso se va a seleccionar un caso basándose en la peligrosidad del mismo. La lluvia es un factor de riesgo y realizar un análisis sobre este tipo de accidentes es una cuestión interesante.

Se van a usar los mismos algoritmos que en los casos anteriores con una primera aproximación en cuanto a parámetros igual. Tras la ejecución de los mismos se obtiene la siguiente tabla de resultados.

Algoritmo	tiempo de ejecución	Número de clusters	Silhouette	Calinski-Harabasz
KMeans	0.041027784	4	0.470795891	1380.190953
AgglomerativeClustering	0.195130587	4	0.494921777	1269.158309
SpectralClustering	0.994660139	4	0.457678012	1070.711182
Birch	0.18012023	4	0.470883793	907.341791
MeanShift	5.684771776	9	0.45055718	460.6466127

Tabla 7: Resultados del primer experimento del tercer caso de estudio

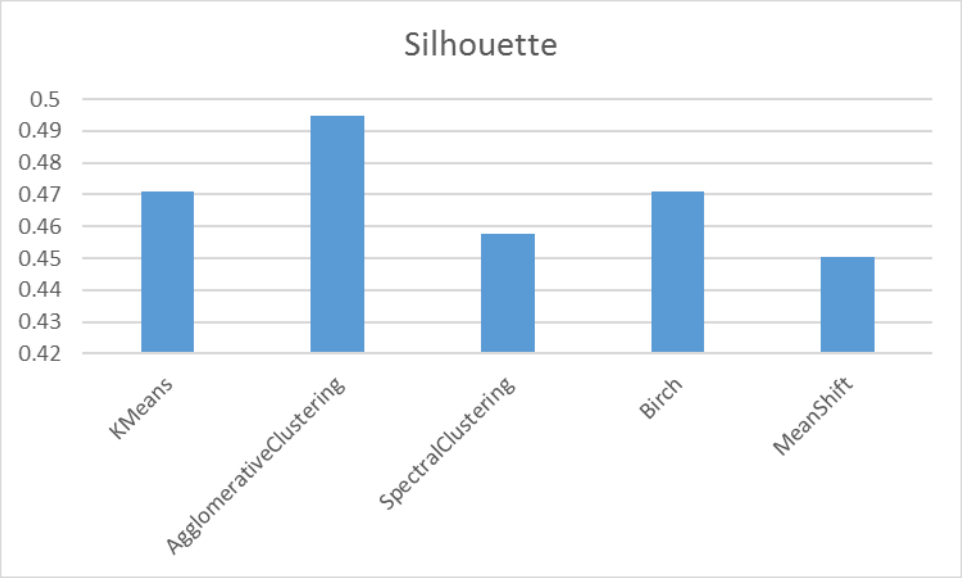

Ilustración 13: índice de Silhouette para el primer experimento del tercer caso de estudio

El índice de Silhouette da al agglomerative como mejor solución, mientras que el spectral clustering y meanshift quedan a la cola.

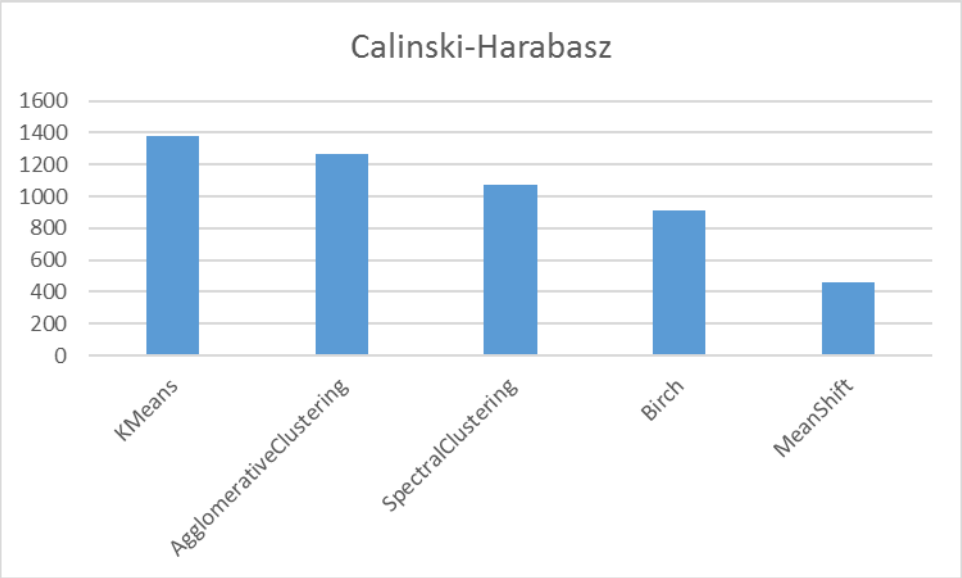

Ilustración 14: índice de Calinski-Harabasz para el primer experimento del tercer caso de estudio

En cuanto al índice de Calinski-Hrabasz KMeans y agglomerative están por encima de los demás. En una segunda aproximación del ajuste de los parámetros se va a seguir el procedimiento anterior e igualar el número de clústeres de los algoritmos al ofrecido por meanshift. Los resultados se exponen a continuación.

Algoritmo	tiempo de ejecución	Número de clusters	Silhouette	Calinski-Harabasz
KMeans	0.037023067	9	0.655921356	1525.271459
AgglomerativeClustering	0.134088993	9	0.669814285	1436.157806
SpectralClustering	0.812484503	9	0.560920881	1089.193545
Birch	0.140093565	9	0.441214895	689.5499065
MeanShift	3.963652134	9	0.45055718	460.6466127

Tabla 8: Resultados del segundo experimento en el tercer caso de estudio

Agglomerative supera a KMeans en el índice de Silhouette por muy poco, aunque KMeans le supera en el índice de Calinski-Harabasz. La efectividad de aumentar el número de clústeres se muestra en las siguientes gráficas.

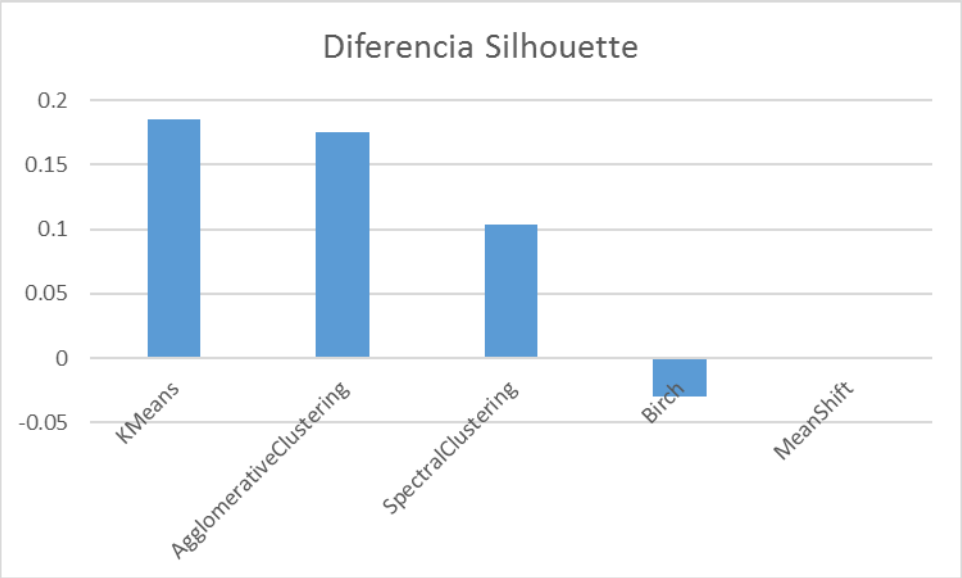

Ilustración 15: Diferencias entre los índices de Silhouette

Tanto KMeans como Agglomerative son los que más se benefician de de aumentar el número de clústeres. El algoritmo de Birch pierde calidad al aumentar los clústeres.

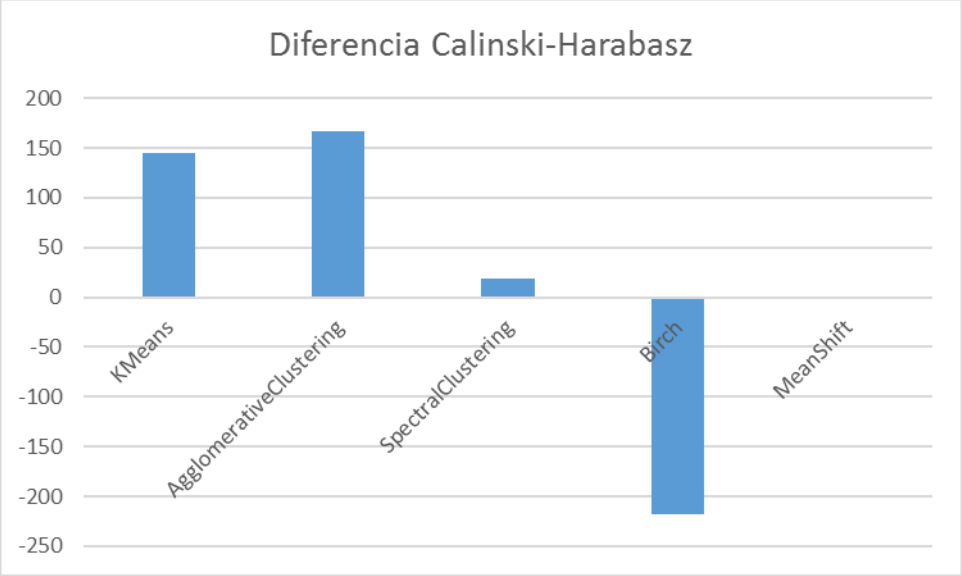

Ilustración 16: Diferencias entre los índices de Calinski-Harabasz

De nuevo KMeans y Agglomerative son los que más se aprovechan de aumentar los clústeres, pero Birch experimenta una caída muy brusca.

En los anteriores casos de estudio se han usado visualizaciones de los resultados de los algoritmos para comparar los resultados de distintos algoritmos con la configuración inicial. En este caso se va a usar la visualización del mismo algoritmo, pero utilizando los distintos resultados obtenidos tanto en la configuración inicial, como en la aproximada. En primer lugar se muestran los clústeres obtenidos por la segunda aproximación de KMeans.

Clúster	VICTIMAS	MUERTOS	HERIDOS_GRAVES	HERIDOS_LEVES	LUMINOSIDAD	VEHICULOS_IMPLICADOS
0	1.002079002	0.01039501	0	0.991683992	3.906444906	1
1	1.985507246	0.014492754	0.065217391	1.905797101	0.166666667	1.34057971
2	1.003952569	0.023715415	0	0.980237154	3.873517787	2.09486166
3	1.714788732	0.017605634	0.031690141	1.665492958	3.313380282	1.373239437
4	2.703252033	0	0.085365854	2.617886179	2.662601626	1.634146341
5	1.35483871	0.032258065	1.225806452	0.096774194	0.806451613	1.258064516
6	1.26	0.02	0.013333333	1.226666667	2.326666667	2.606666667
7	1.262626263	0.050505051	1.141414141	0.070707071	3.535353535	1.282828283
8	1.214285714	0.023809524	0	1.19047619	0.214285714	2.642857143

Tabla 9: Medias de los valores de KMeans en el segundo experimento del tercer caso de estudio

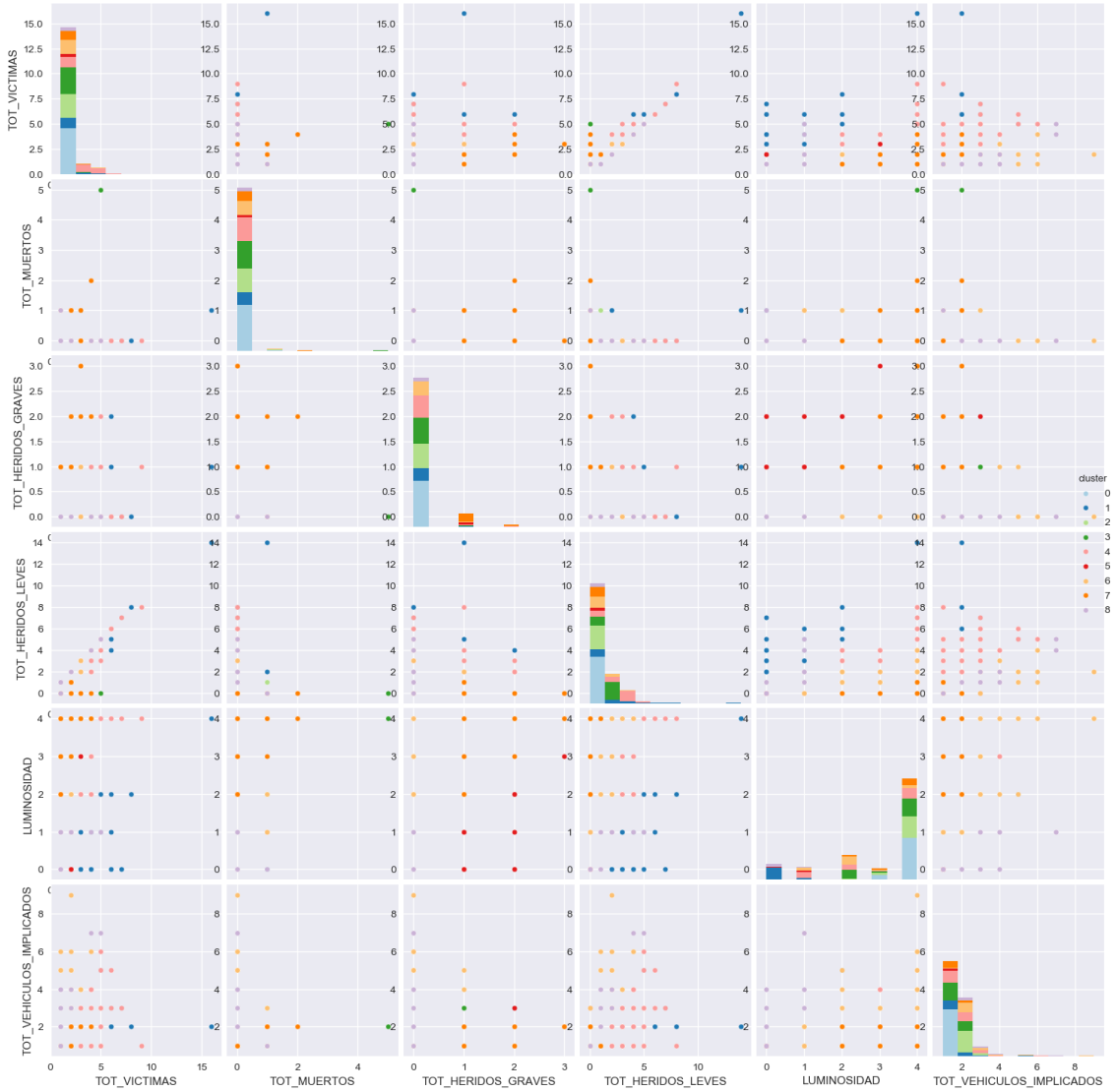

Ilustración 17: Scatter matrix de KMeans aproximando clústeres en el tercer caso de estudio

Aunque el agrupamiento en clústeres de la aproximación con 9 clústeres, de forma visual es mucho más complicado poder extraer conclusiones. Una posible solución a este problema es

eliminar los clústeres minoritarios para poder ofrecer información de forma más clara, aunque parte de la misma se pierde, por lo que se debe elegir bien qué clústeres se eliminan y bajo qué criterio se poda un clúster. Aún asi, se pueden extraer conclusiones como por ejemplo que el clúster 4 se caracteriza por tener un número bajo de muertos, y el número de vehículos implicados, como el de heridos leves es bastante variable, pero tienen un valor medio de 1.6 y 2.6 respectivamente, por lo que no es especialmente alto. El clúster 1 tiene un número de victimas alto. Con la ayuda de la tabla superior se puede añadir información, como por ejemplo que los clústeres 0, 2 y 8 no tienen heridos graves, al igual que el clúster 4 no tiene muertos.

En cuanto al algoritmo con los parámetros sin aproximar se obtiene la siguiente visualización.

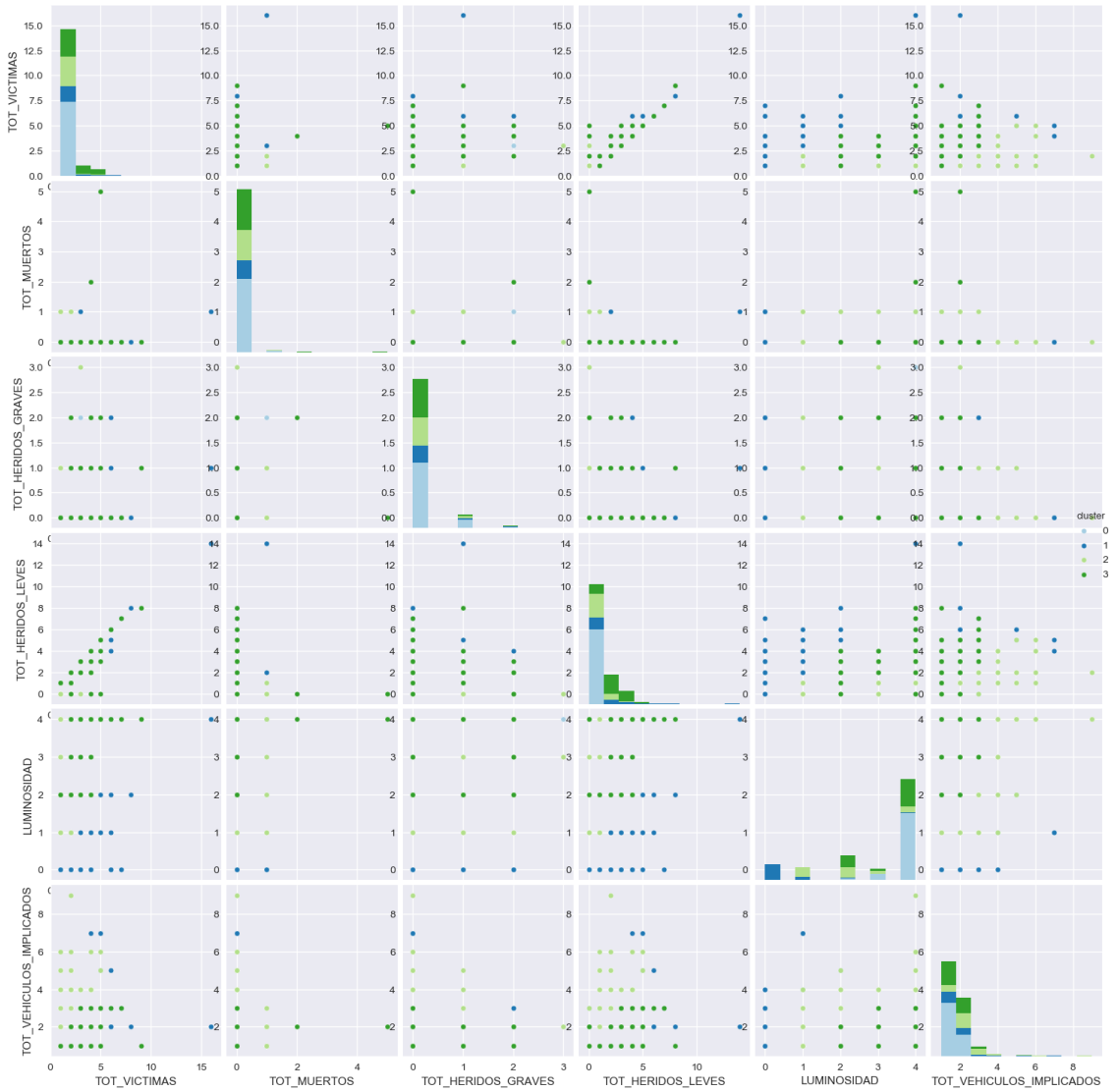

Ilustración 18: Scatter matris de KMeans sin aproximar clústeres en el tercer caso de estudio

En este caso la visualización ofrece una información más clara, aunque los clústeres formados sean de peor calidad. El clúster 0 tiene un total de vehículos implicados bajo, el clúster 1 tiende a tener una baja luminosidad, mientras que el clúster 3, tiene una visibilidad alta. El total de víctimas del clúster 2 tiende a ser bajo, mientras que el del clúster 1 es más alto.