Tarea:

Los datos para usar corresponden a el estado de los conocimientos de los estudiantes sobre las maquinas eléctricas de corriente continua, los datos constan de 6 columnas y cada línea corresponde a un estudiante, esta información ya está clasificada en la columna “UNS” dado que se van a aplicar 4 técnicas de Clustering desechamos esta columna para la implementación de los modelos.

Repo: https://github.com/dherrerambo/analisis-de-datos-avanzados/blob/master/Clase%205%20-%20Unsupervised/Tarea-Clase5.ipynb

Se tomo para el ejercicio los atributos LPR y PEG para el análisis de Cluster.

Desarrollo

Exploración de los datos:

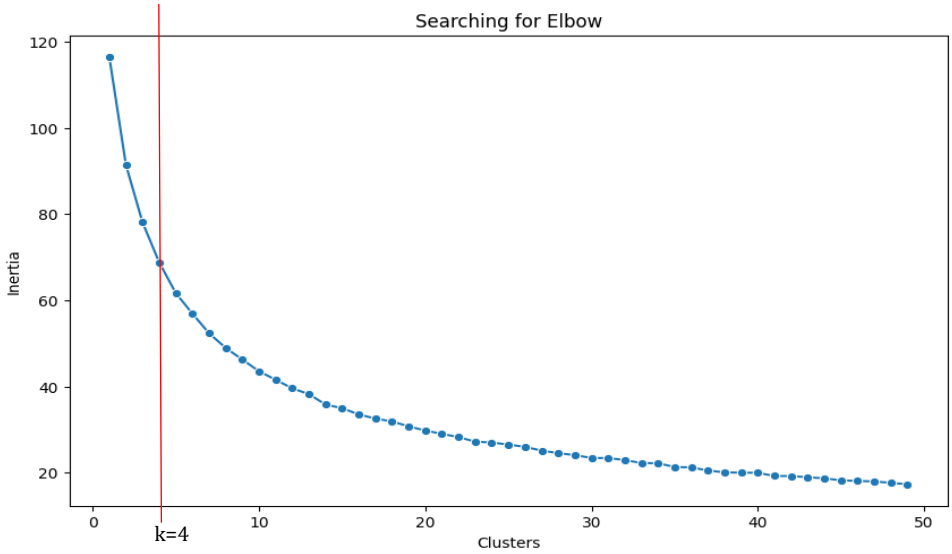
No se tienen valores nulos y los datos están estandarizados en escala de 0 a 1, para los atributos presentes.

Presentado por:
Diego Herrera Malambo

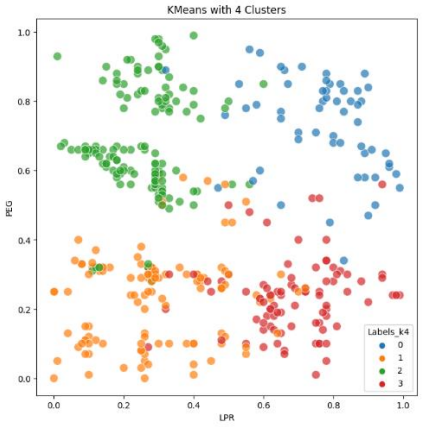
Implementaciones de Cluster

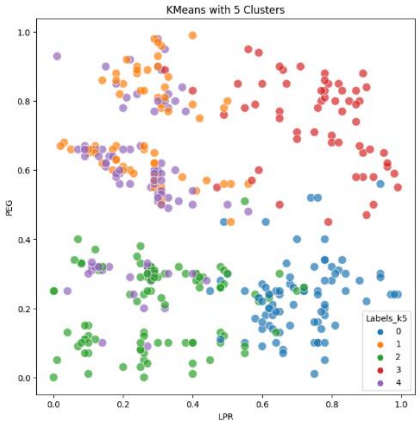
1. K-MEANS

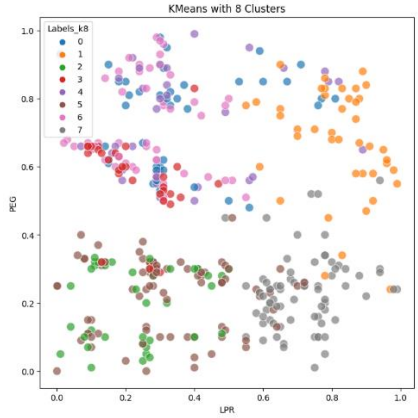
Este modelo de cluster permite la optimización del parámetro k a través de la gráfica Elbow:

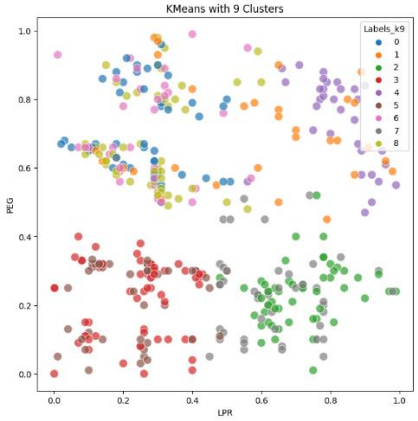


Al implementar diferentes valores de cluster para las variables seleccionadas podemos ver que visualmente la mas diferenciada es la que se genera con 4 clusters.





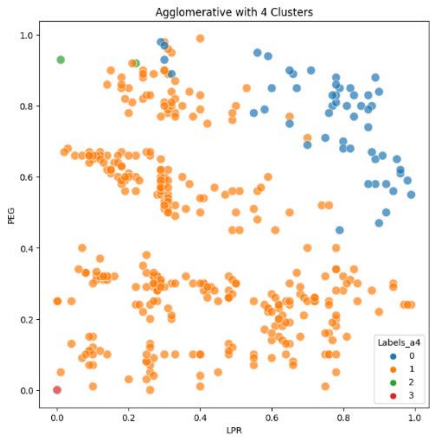


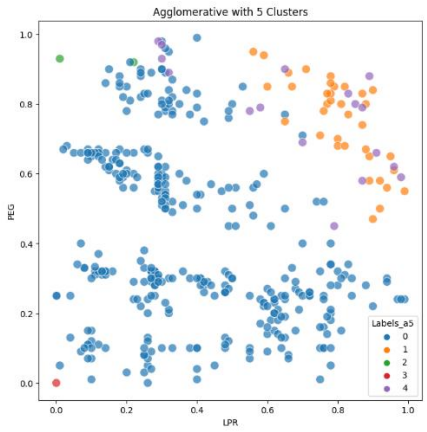


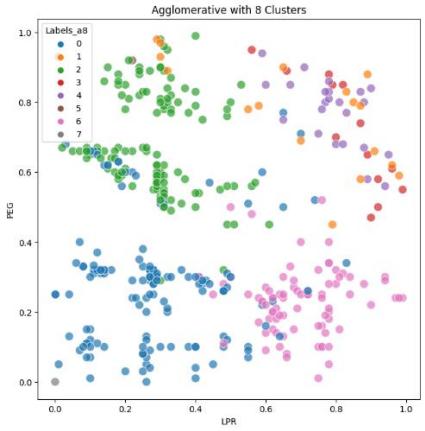
Presentado por:
Diego Herrera Malambo

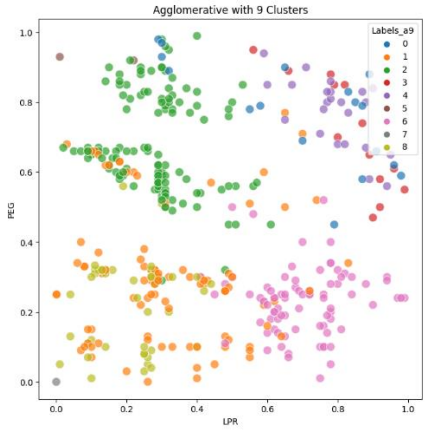
1. Cluster Jerárquico

Al implementar este método los cluster quedaron muy mezclados y sobrepuestos, y cuando son *k* bajos, las agrupaciones abarcan muchos elementos sin realizar bien la segmentación.









2. DBSCAN

Este método no sirvió con los datos del ejercicio, si validaron con varias combinaciones de parámetros pero no fue posible ver variaciones, para todas las combinaciones se marcaba un solo cluster.

3. MeanShift

Presento combinaciones muy marcadas y aleatorias de los cluster, estando estos mezclados, se realizaron varias pruebas con diferentes quantiles, este método tampoco arrojo resultados de clusterizacion favorables.

Presentado por:
Diego Herrera Malambo

Conclusiones

El método que mejor se acomodo a los datos fue el de ***K-Means con k=4***, dando una separación visualmente mas ajustada que los demás métodos, el que peor se desempeño fue el de DBSCAN, no aportando nada para su análisis.

Al comparar la clasificación real en **UMS** vs los resultados obtenido por ***K-Means con k=4*** (**Labels_k4**) no se puede hacer de forma directa ya que al implementar el modelo este asigna de forma aleatoria los cluster por ende no necesariamente el cluster 1 de **UMS** corresponda al cluster 1 del modelo, para lo cual se realiza una homologación según la distribución de **UMS** vs **Labels_k4** y se almaceno en un diccionario.

```
Labels_k4                              {2: 136, 1: 122, 3: 88, 0: 57}
Se van a llevar a la forma de UNS
UNS                                    {1: 129, 2: 122, 3: 102, 0: 50}
Homologación Labels_k4 a UMS:          {2: 1, 1: 2, 3: 3, 0: 0}
```

Obteniendo una nueva distribución de datos:

```
{1: 136, 2: 122, 3: 88, 0: 57}
```

Al implementar una tabla de contingencia/matriz de confusión, para ver la distribución de Labels_k4 vs UMS:

Frecuencia

N_Labels_k4	0	1	2	3
UNS				
0	0	0	44	6
1	0	7	68	54
2	6	79	10	27
3	51	50	0	1

Distribución

N_Labels_k4	0	1	2	3
UNS				
0	0.00	0.00	10.92	1.49
1	0.00	1.74	16.87	13.40
2	1.49	19.60	2.48	6.70
3	12.66	12.41	0.00	0.25

La diagonal representa los valores correctamente clasificados = 18 elementos, el resto han sido clasificados de manera incorrecta por el modelo.

A fin de ejercicio y validar el modelo se usa la métrica *accuracy_score* = 0.0446, lo que es consistente con lo visto en la matriz de confusión.