## **Tarea 3 AEM**

## **Equipo:**

## **Oscar Zambrano Aguirre**

## **Lilivette Cruz Aguirre**

## **Ricardo De León Flores**

1. **Descarga la base de datos Wholesale customers Data Set.**

<https://archive.ics.uci.edu/ml/datasets/Wholesale+customers>

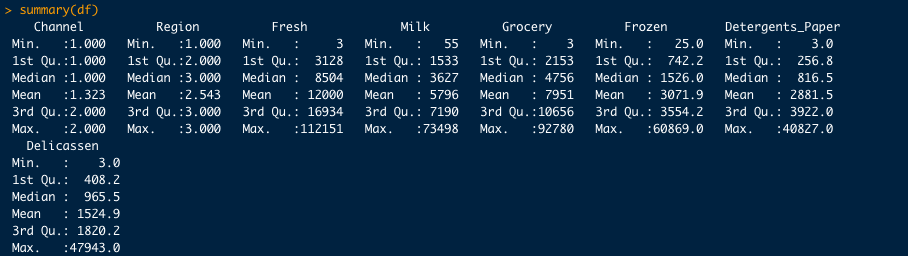


2. **Describe brevemente qué información contiene la base de datos.**

El conjunto de datos se refiere a clientes de un distribuidor mayorista. Incluye el gasto anual en unidades monetarias (m.u.) en diversas categorías de productos.

1) FRESH: gasto anual (m.u.) en productos frescos.  
2) MILK: gasto anual (m.u.) en productos lácteos.  
3) GROCERY: gasto anual (m.u.) en abarrotes

4) FROZEN: gasto anual (m.u.) en productos congelados.  
5) DETERGENTS\_PAPER: gasto anual (m.u.) en detergentes y papeles  
6) DELICATESSEN: gasto anual (m.u.) en productos selectos  
7) CHANNEL: canal de venta  
8) REGION: región.



3. **Selecciona solo las variables Fresh y Delicatessen.**



4. **Normaliza/escala el nuevo dataset (puedes usar scale(X)) 4.1. ¿Qué hace, operacionalmente, la función scale()?**

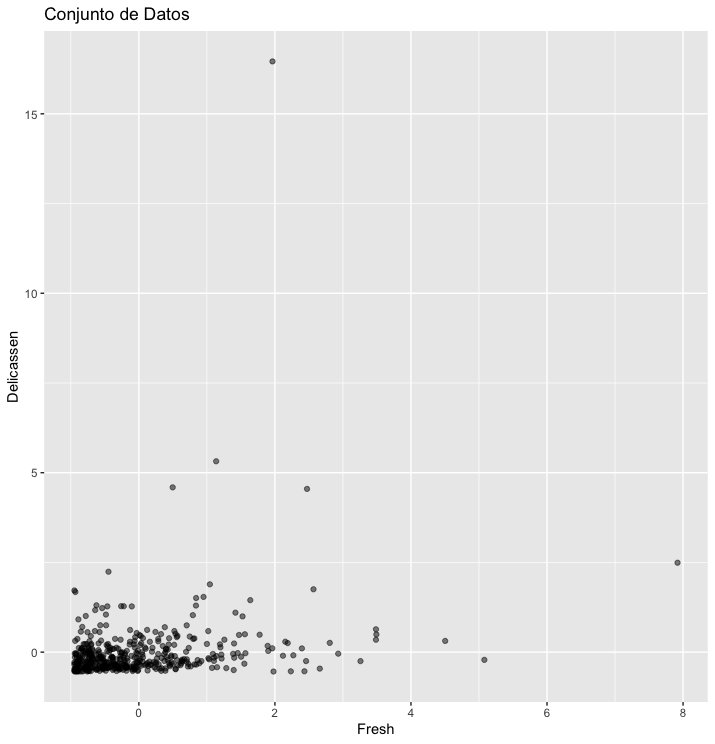
Para tipificar una tabla de datos multidimensional en R vamos a utilizar la función scale(X,center=...,scale=...), donde:

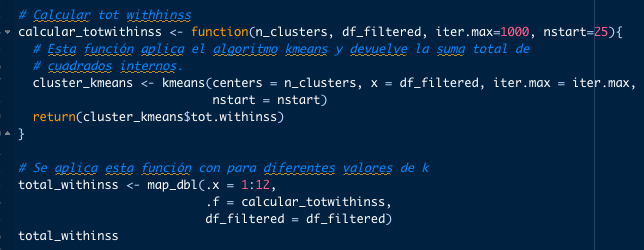
* **X**: Matriz o tabla de datos.
* **center**: Vector que restamos a la columnas. center=TRUE (defecto).
* **scale**: Vector por el que dividimos la matriz. scale=TRUE (defecto), dividimos por las desviaciones típicas muestrales. scale=FALSE, no dividimos por nada

5. **Con el dataset normalizado/escalado, obtén el número óptimo de parámetros 𝑘∗ para K-means: 5.1. Usando el método *elbow* variando de 1 a 12 clusters. Recuerda que para esto debes extraer el valor *tot.withinss* de cada valor de 𝑘 que pruebes. 5.2. Usando el estadístico Gap con la función *clusGap().* Usa el argumento B=150. Una vez que haya terminado el algoritmo, recuerda que puedes obtener el número óptimo 𝑘∗ con el *print(gap\_model, method = XX)*, sustituye en XX alguno de los métodos que mencioné en clas.**

**Método de Elbow**,  La función fviz\_nbclsut() del paquete factoextra automatiza todo el proceso, empleando como medida de varianza intra-cluster la suma de residuos cuadrados internos (wss).



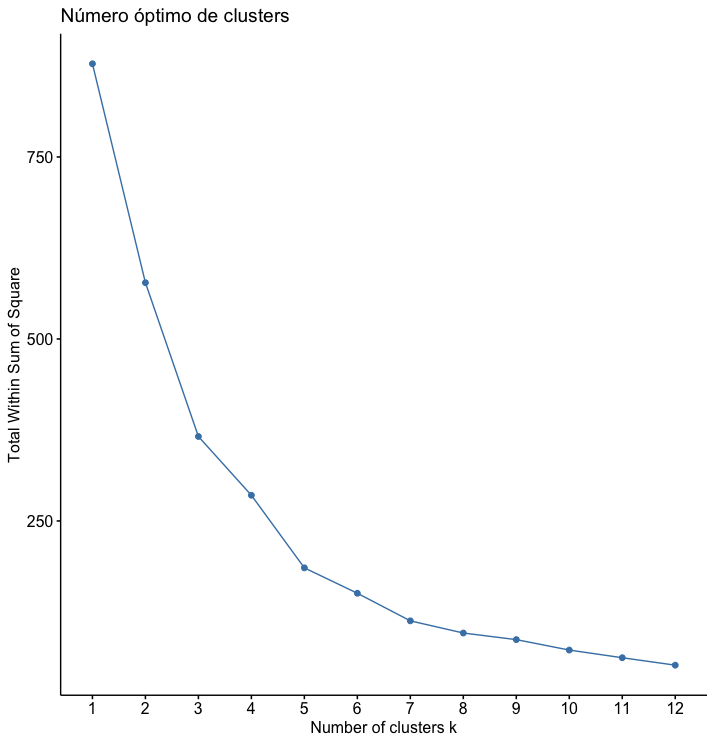






Graphical user interface, text, website

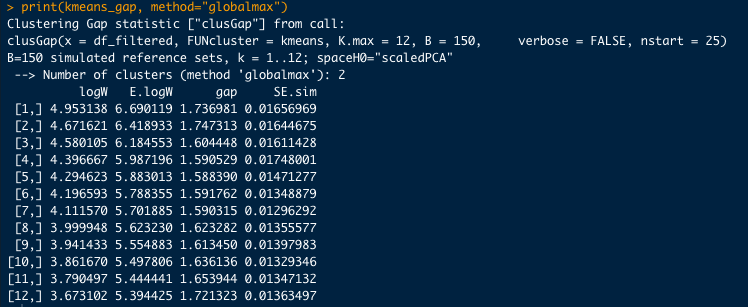
Description automatically generated



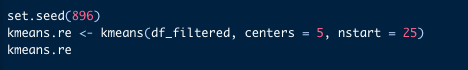
Se puede obtener el el estadístico *gap* con la función **fviz\_nbclust()** o con la función **clusGap()** del paquete cluster.

Graphical user interface, text, application

Description automatically generated



6. **Corre el algoritmo de K-means con el valor que tú decidas del paso 5.**

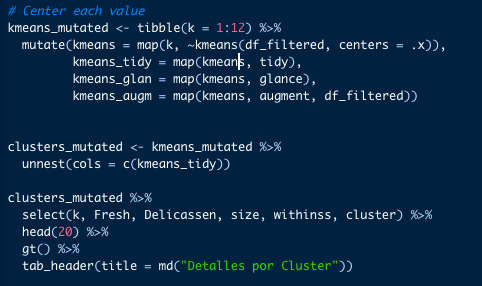


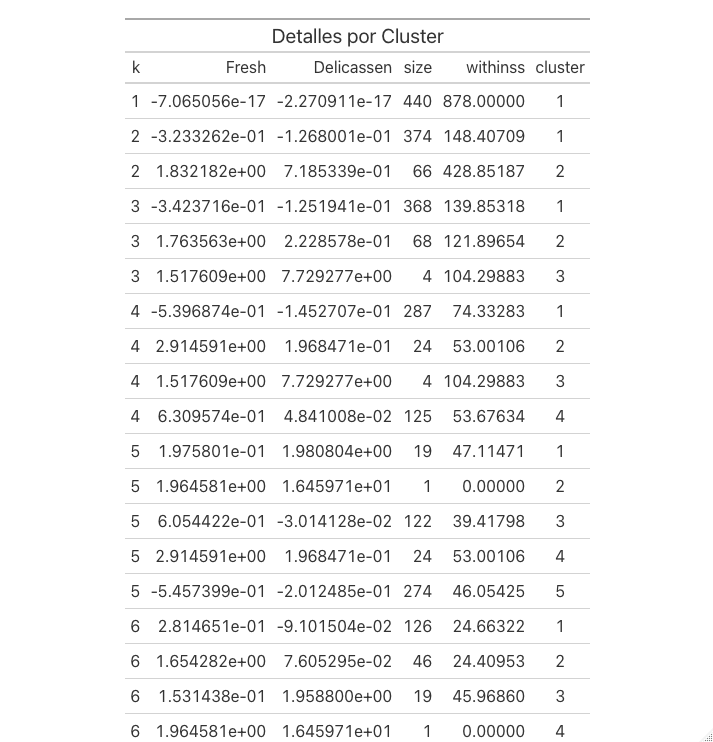
Esta centrado en 5 ya que vemos que es en donde los valores comienzan a estabilizarse.



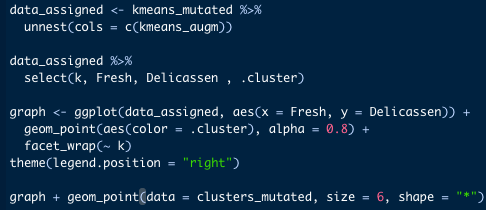
7. **Grafica los datos originales usando un color diferente por cluster. Recuerda que la variable cluster la puedes obtener de la lista que te arroja kmeans() en R.**

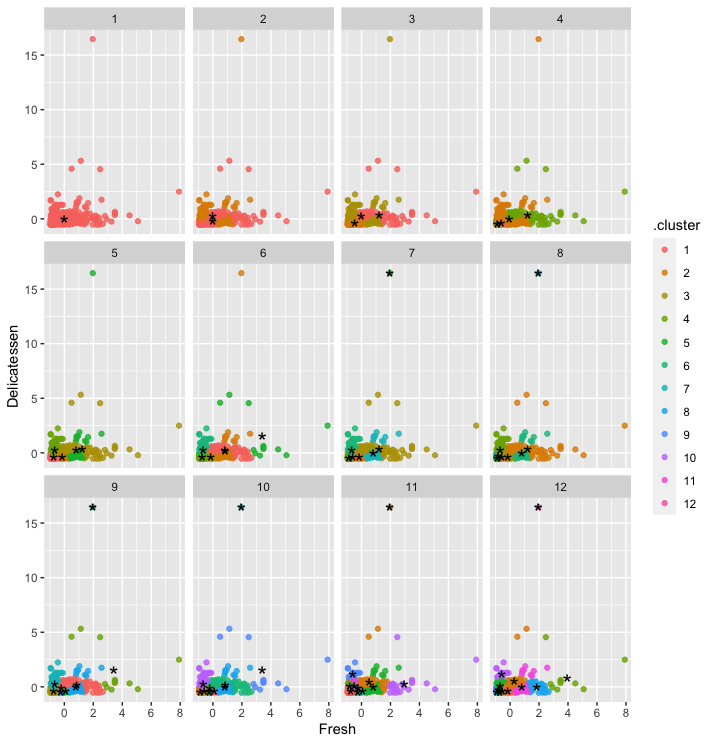
* Primero centramos cada valor y vemos su detalle por columna





* Obtenemos y graficamos cada clúster individualmente





* De igual manera se pueden observar los clústeres en una gráfica de diferente manera.



Chart, scatter chart

Description automatically generated