Actividad Evaluable 4: Patrones con K-means

Equipo 3

Abiel Moisés Borja García A01654937

Aranza García Narvaéz A01654658

María Clarita Osorio Vergara A01654530

Gael Eduardo Pérez Gómez A01753336

Marco Uriel Pérez Gutiérrez A01660337

Mayo, 2022

Herramientas computacionales: el arte de la analítica

Grupo 222

Profesor:

Sergio Ruiz Loza

Instituto Tecnológico y de Estudios Superiores de Monterrey

1. Carga tus datos.

In [1]:
```
# Importamos las librerías que necesitamos
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sb
from sklearn.cluster import KMeans
from sklearn.metrics import pairwise_distances_argmin_min

%matplotlib inline
from mpl_toolkits.mplot3d import Axes3D

data = pd.read_csv("avocado.csv")
data.head(5)
```

Out[1]:

	Unnamed: 0	Date	AveragePrice	Total Volume	4046	4225	4770	Total Bags	Small Bags	Large Bags	XLarge Bags	type	year
0	0	2015-12-27	1.33	64236.62	1036.74	54454.85	48.16	8696.87	8603.62	93.25	0.0	conventional	2015
1	1	2015-12-20	1.35	54876.98	674.28	44638.81	58.33	9505.56	9408.07	97.49	0.0	conventional	2015
2	2	2015-12-13	0.93	118220.22	794.70	109149.67	130.50	8145.35	8042.21	103.14	0.0	conventional	2015
3	3	2015-12-06	1.08	78992.15	1132.00	71976.41	72.58	5811.16	5677.40	133.76	0.0	conventional	2015
4	4	2015-11-29	1.28	51039.60	941.48	43838.39	75.78	6183.95	5986.26	197.69	0.0	conventional	2015

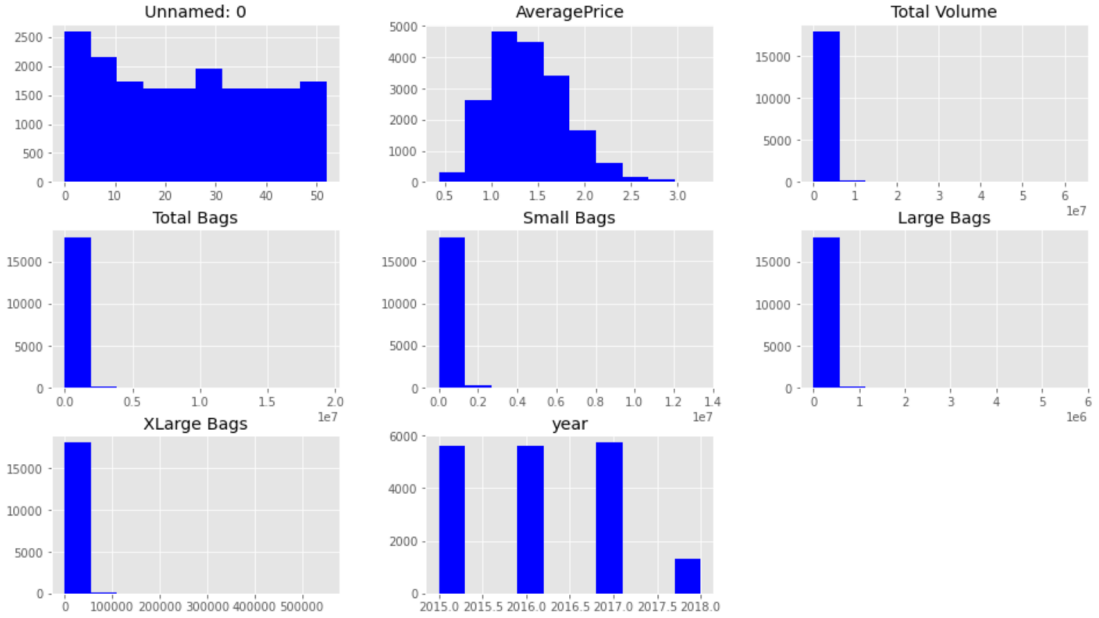
2. Si determinas que alguna variable no sirve basándose en la actividad pasada, elimínala y justifica por qué quitaste o no variables.
Eliminamos las columnas con los nombres '4046', '4225', '4770', ya que sus nombres no son relevantes o no podemos interpretar que es.

In [2]:
```
# Eliminamos las columnas con esos nombres ya que sus nombres no son relevantes
data = data.drop(columns=['4046', '4225', '4770'])
data.head(5)
```

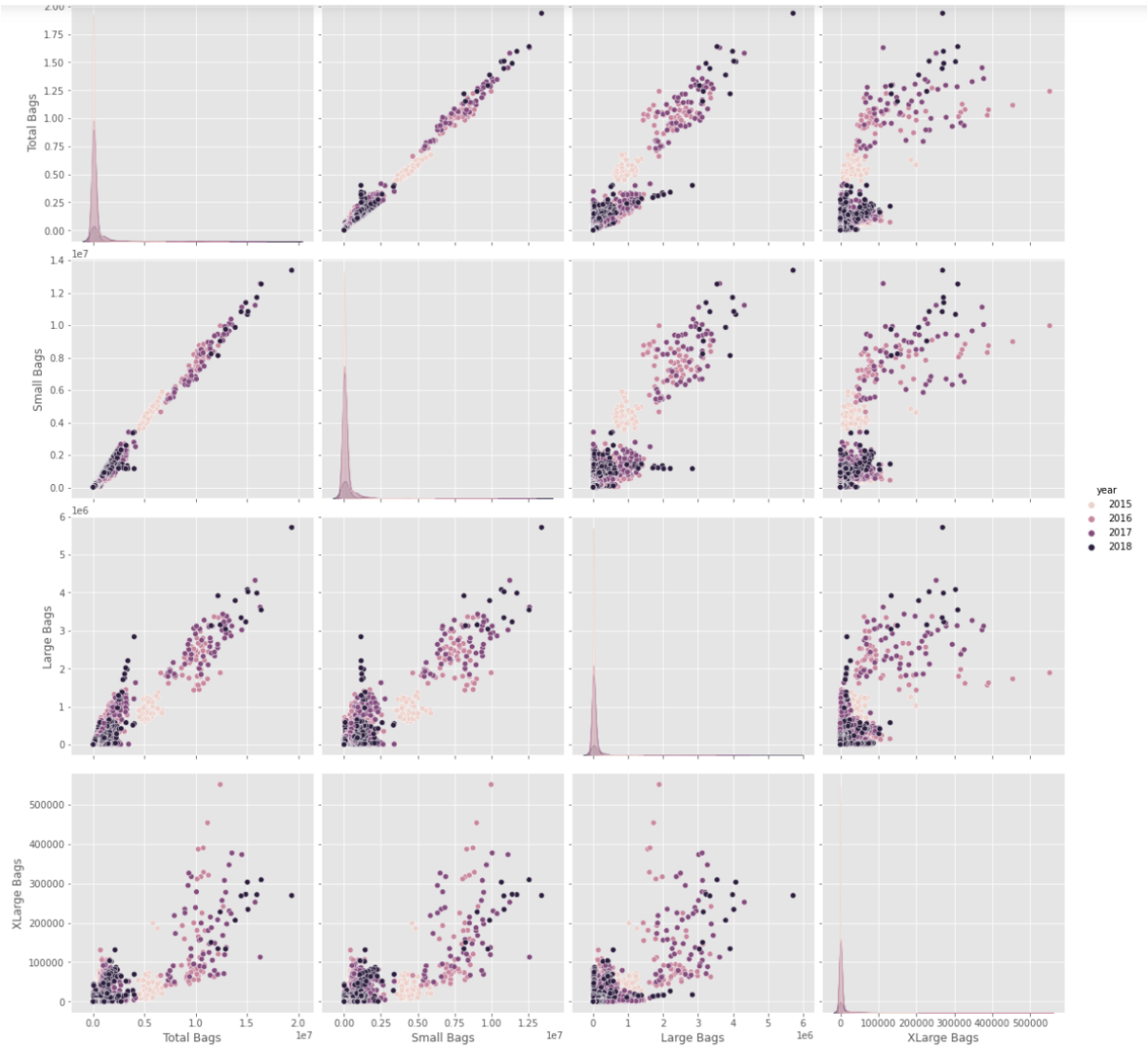
Out[2]:

	Unnamed: 0	Date	AveragePrice	Total Volume	Total Bags	Small Bags	Large Bags	XLarge Bags	type	year	region
0	0	2015-12-27	1.33	64236.62	8696.87	8603.62	93.25	0.0	conventional	2015	Albany
1	1	2015-12-20	1.35	54876.98	9505.56	9408.07	97.49	0.0	conventional	2015	Albany
2	2	2015-12-13	0.93	118220.22	8145.35	8042.21	103.14	0.0	conventional	2015	Albany
3	3	2015-12-06	1.08	78992.15	5811.16	5677.40	133.76	0.0	conventional	2015	Albany
4	4	2015-11-29	1.28	51039.60	6183.95	5986.26	197.69	0.0	conventional	2015	Albany

In [3]:
```
# Dispersión de datos con histograma
plt.rcParams['figure.figsize'] = (16, 9)
plt.style.use('ggplot')
data.hist(color = 'blue')
plt.show()
```

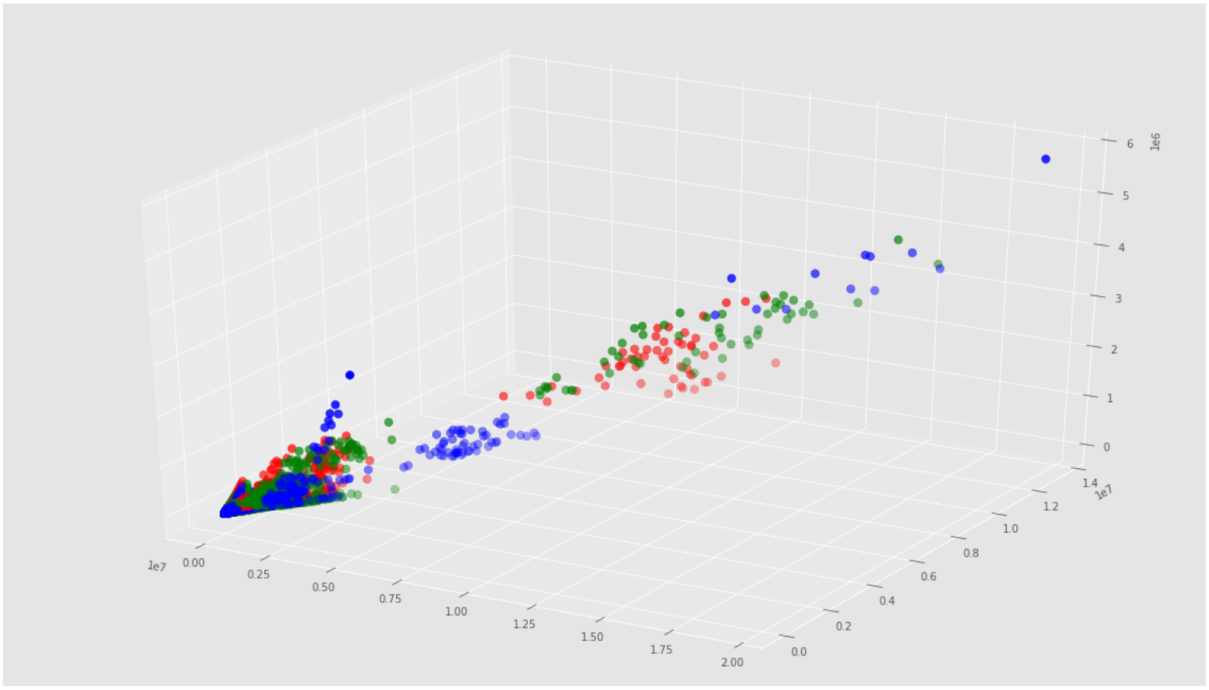


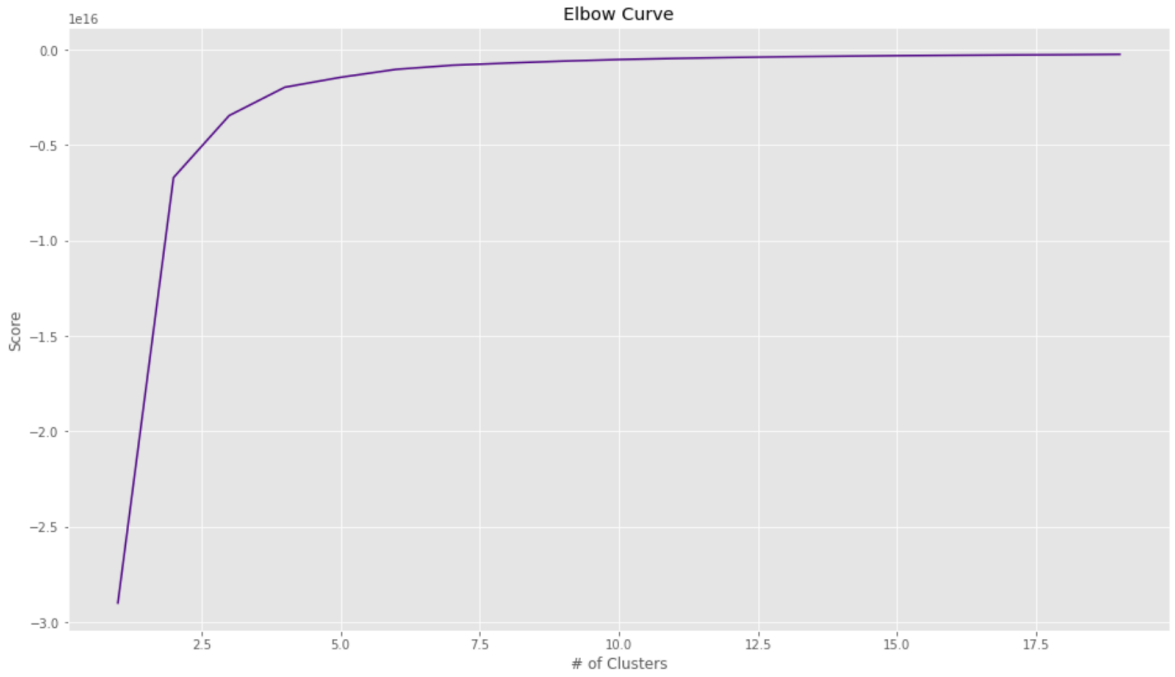
```
In [8]: # Escoger dimensiones y s_e cruzan para ver agrupación y relación con categorías
sb.pairplot(data.dropna(), hue='year',height=4,
            vars=["Total Bags","Small Bags","Large Bags", "XLarge Bags"],
            kind='scatter')
```



3. Determina un valor de k.

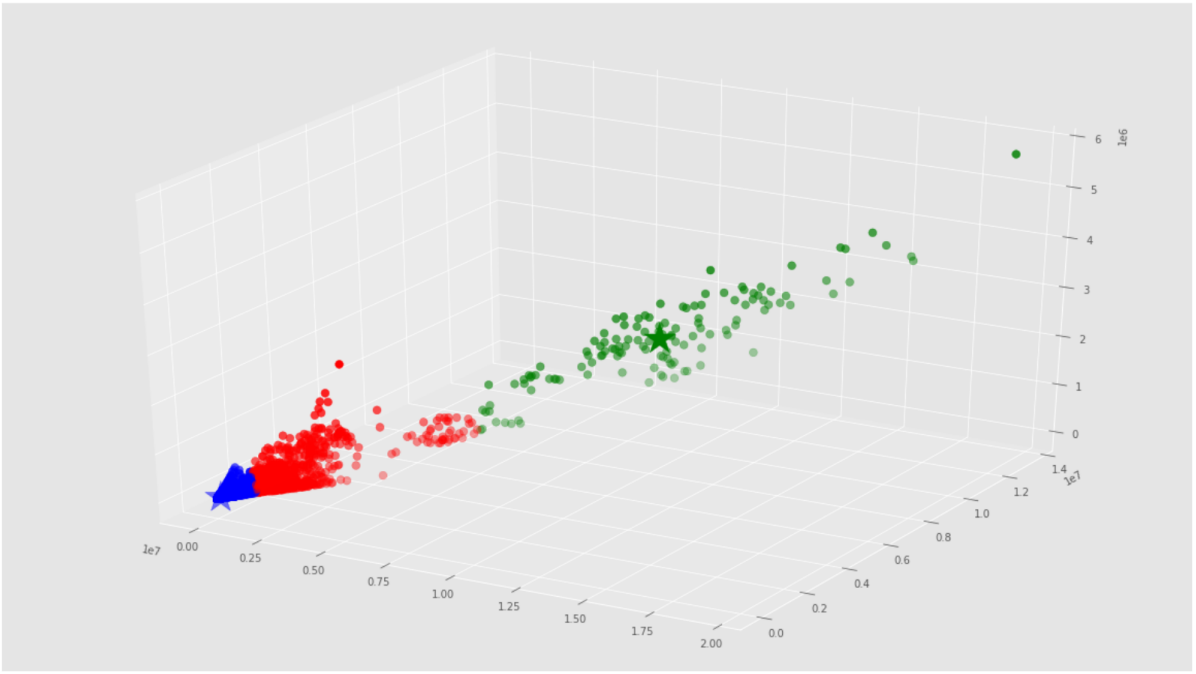
Realizamos la Elbow Curv y vemos el punto de inflexión para determinar K, el número de grupos a crear.





Analizamos la gráfica y determinamos que K es 3, por lo tanto crearemos 3 grupos en el algoritmo de K-means.

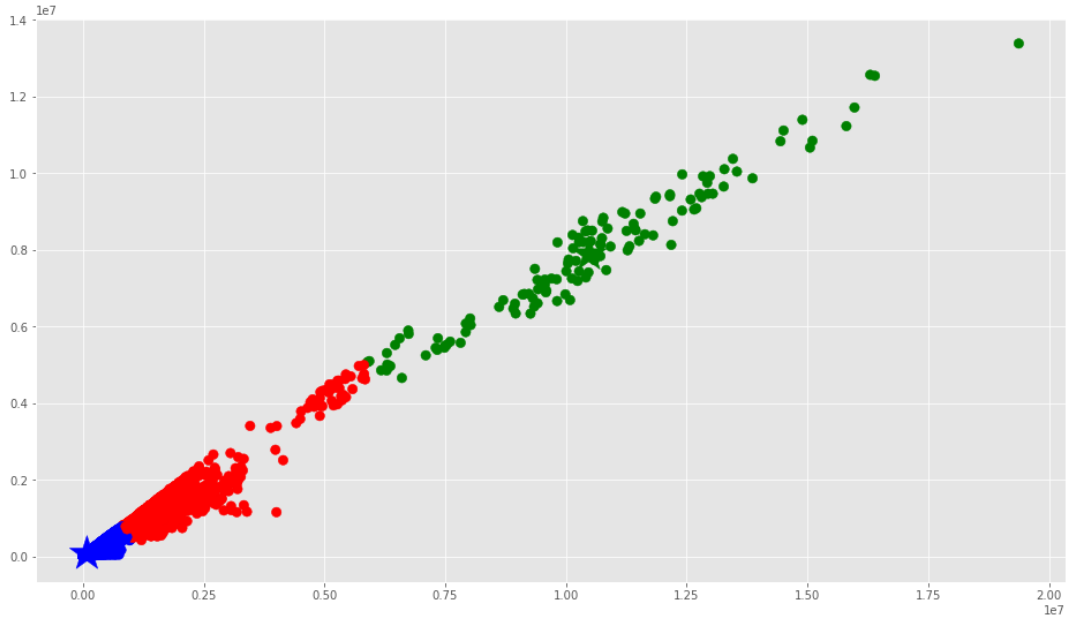
4. Utilizando scikitlearn calcula los centros del algoritmo k-means.



Obtenemos los centros de los 3 grupos y realizamos una gráfica para verlos.

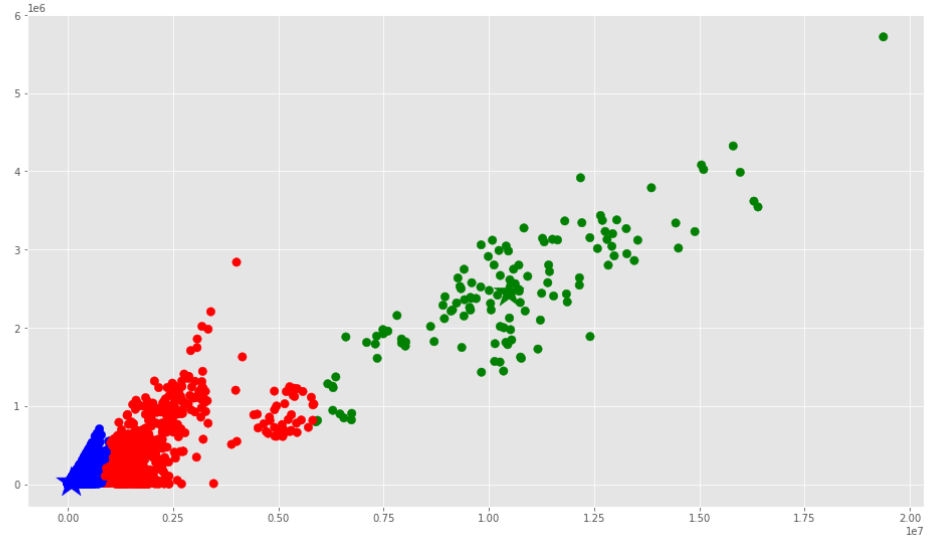
In [11]:
```
# Getting the values and plotting it
f1 = data["Total Bags"].values
f2 = data["Small Bags"].values

plt.scatter(f1, f2, c=asignar, s=70)
plt.scatter(C[:, 0], C[:, 1], marker='*', c=colores, s=1000)
plt.show()
```



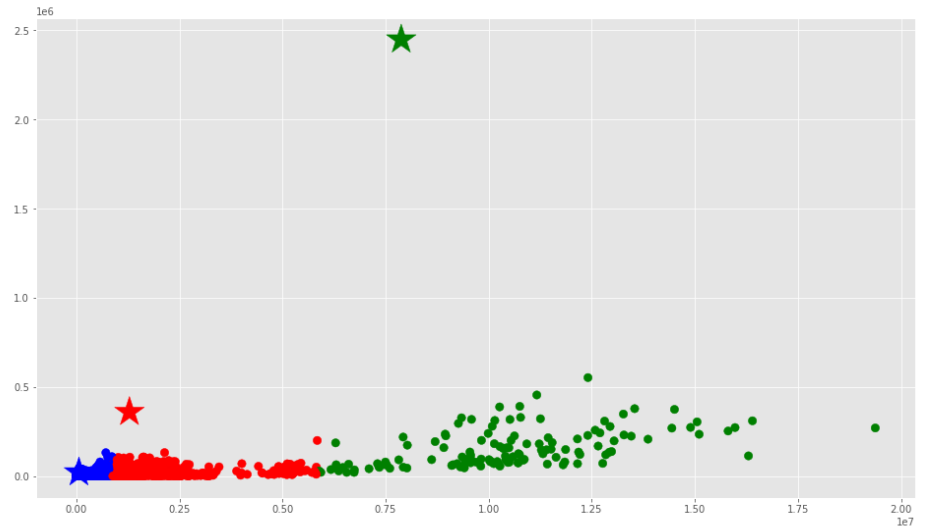
In [12]:
```
# Getting the values and plotting it
f1 = data['Total Bags'].values
f2 = data['Large Bags'].values

plt.scatter(f1, f2, c=asignar, s=70)
plt.scatter(C[:, 0], C[:, 2], marker='*', c=colores, s=1000)
plt.show()
```



In [13]:
```
f1 = data['Total Bags'].values
f2 = data['XLarge Bags'].values

plt.scatter(f1, f2, c=asignar, s=70)
plt.scatter(C[:, 1], C[:, 2], marker='*', c=colores, s=1000)
plt.show()
```



5. Contamos la cantidad de datos de acuerdo a los colores que separamos.

```
In [14]: copy =  pd.DataFrame()
         copy['AveragePrice']=data['AveragePrice'].values
         copy['year']=data['year'].values
         copy['label'] = labels;
         qtygroup =  pd.DataFrame()
         qtygroup['color']=colores
         qtygroup['cantidad']=copy.groupby('label').size()
         qtygroup
```

Out[14]:

	color	cantidad
0	blue	17188
1	green	128
2	red	933

6. Mostramos por año la cantidad de precio promedio.

```
In [15]: grp_ref_indx = copy['label'] ==0
         grp_ref = copy[grp_ref_indx]

         divgroup =  pd.DataFrame()
         divgroup['year']=[2015,2016,2017,2018]
         divgroup['cantidad']=grp_ref.groupby('AveragePrice').size()
         divgroup
```

Out[15]:

	year	cantidad
0	2015	NaN
1	2016	151.0
2	2017	59.0
3	2018	2.0

Basado en los centros responde las siguientes preguntas:

- ¿Crees que estos centros puedan ser representativos de los datos? ¿Por qué?
  Los centros de los grupos nos pueden ayudar para confirmar o comprobar alguna teoría o hipótesis que teníamos sobre nuestra muestra de datos. A la vez, nos puede ayudar a descubrir una nueva relación en los datos que no habíamos visto. Al final, con la ayuda de los centros podremos obtener las etiquetas de los grupos y cada vez que tengamos nuevos datos poder clasificarlos en nuestros grupos obtenidos.

- ¿Cómo obtuviste el valor de k a usar?
  El algoritmo de k means nos indica obtener la elbow curl, la cual tendremos que analizar y encontrar su punto de inflexión. Ese punto o el valor del punto es el valor de k, el cual es el número de grupos que crearemos.

- ¿Los centros serían más representativos si usaras un valor más alto? ¿Más bajo?
  Mientras el valor de K sea más alto, los grupos que formemos serán más precisos, por lo tanto, los valores del grupo se acercarán más al centro del grupo. Mientras más bajo que, significa que no se encuentran más similitudes.

- ¿Qué distancia tienen los centros entre sí? ¿Hay alguno que esté muy cercano a otros?
  En nuestro caso tenemos 3 grupos, rojo, azul y verde. Los grupos rojo y azul tienen centros cercanos, mientras que el verde está más alejado de ellos. Esto significa que los grupos rojo y azul tienen más similitudes juntos que con el grupo verde.

- ¿Qué pasaría con los centros si tuviéramos muchos outliers en el análisis de cajas y bigotes?
  Si tuviéramos varios valores atípicos, nuestra gráfica haría que las cajas sean más extensas y se acerquen más a los extremos de minimo y maximo, mientras que los bigotes que harían más cortos.

- ¿Qué puedes decir de los datos basándose en los centros?
  Un análisis rápido que podemos dar es que los grupos rojo y azul cuentan con una gran similitud, ya que sus centros son muy cercanos. Mientras que el grupo verde, es más único y no cuenta con tantas similitudes con ellos, ya que su centro está alejados del de ellos.