

Logística y distribución

Análisis de entrega de mercancía

David González





Objetivo

**Definir el número óptimo de
centros de abastecimiento
(clusters) de producto.**

David González

Antecedentes

Este análisis está basado en datos de una empresa que participa en el segmento de comida rápida. Los datos corresponden a la cantidad de cajas semanales que se envían a los diversos restaurantes localizados en la República Mexicana.





Análisis

Algoritmos no supervisados

Mediante el uso algoritmos no supervisados, se pretenden agrupar los diferentes destinos de entrega de producto, con la finalidad de encontrar grupos similares en el conjunto de datos y determinar la cantidad de centros de abastecimiento adecuados para atender los diversos puntos de consumo.

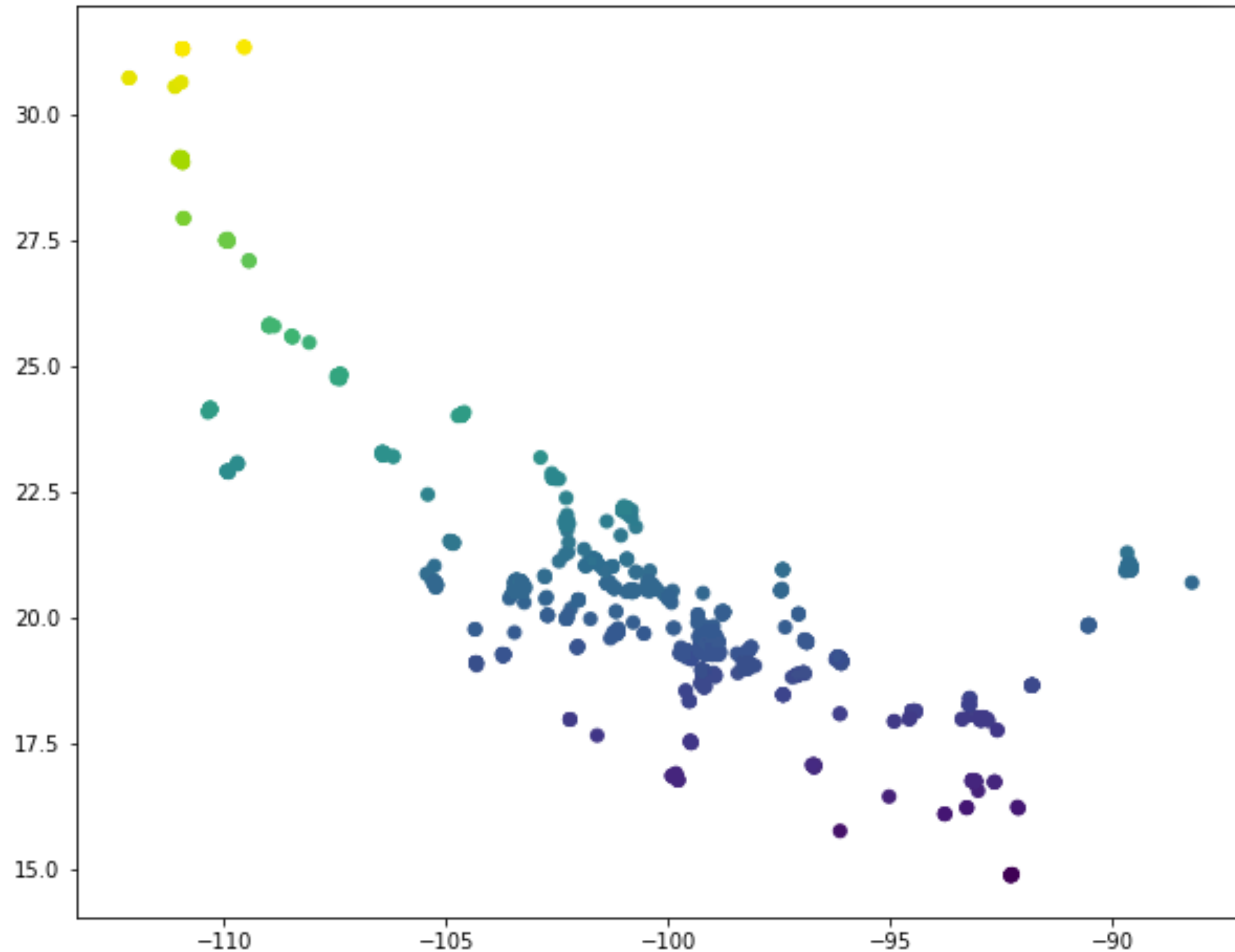
Geolocalización de sucursales

La finalidad es identificar en un mapa de la República Mexicana, los diversos puntos de consumo mediante su geolocalización, para observar de manera gráfica la agrupación de dichos puntos.

David González

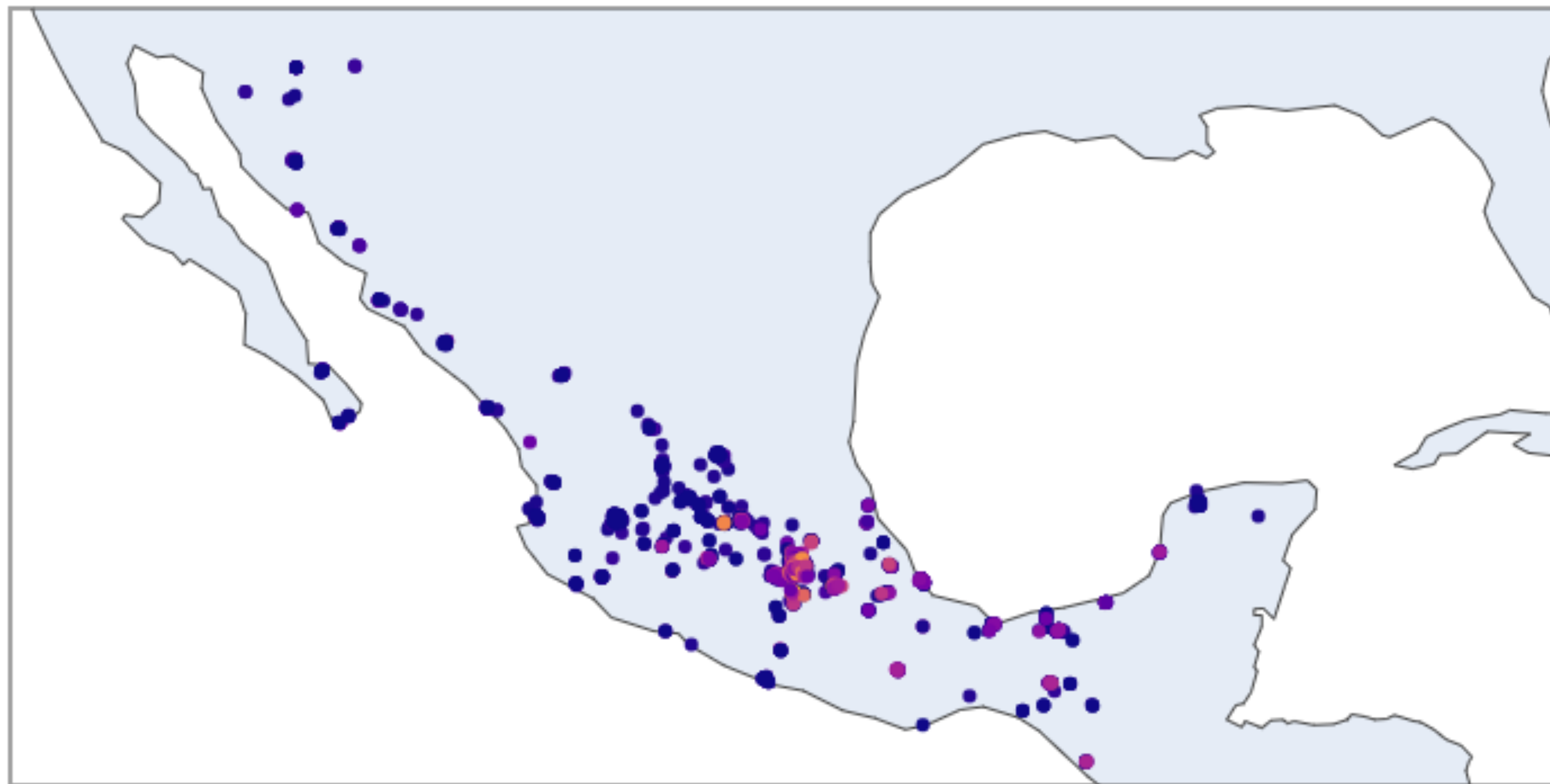
Geolocalización

Diagrama de dispersión para la ubicación geográfica de los restaurantes.



```
df2 = df[["Longitud", "Latitud"]]  
X = df2[["Longitud"]].to_numpy()  
y = df2[["Latitud"]].to_numpy()  
df2  
  
[21] fig = plt.figure(figsize=(10, 8))  
     plt.scatter(X[:, 0], y[:, 0], c=y);
```

```
[4] !pip install geopandas  
[5] import geopandas as gpd
```



Geolocalización

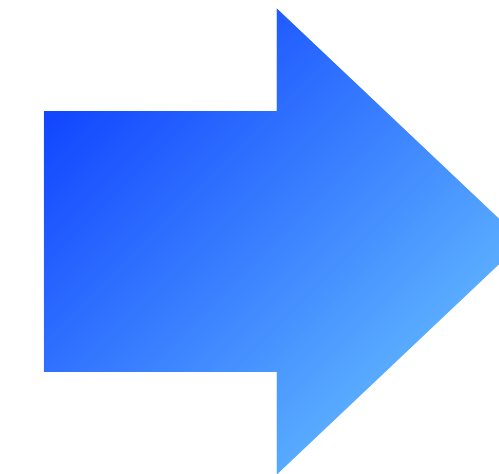
Mapa con geolocalización de puntos de consumo.

```
fig = go.Figure(data=go.Scattergeo(  
    lon = df['Longitud'],  
    lat = df['Latitud'],  
    text = df['Texto'],  
    mode = 'markers',  
    marker_color = df['Cajas'],  
))  
  
fig.update_layout(  
    title = 'Volumen de cajas distribuidas',  
    geo_scope='world',  
)  
fig.show()
```

Preparación de datos

```
X = df[["Cajas", "Costo_caja", "Latitud", "Longitud"]]  
X
```

Primeros intentos



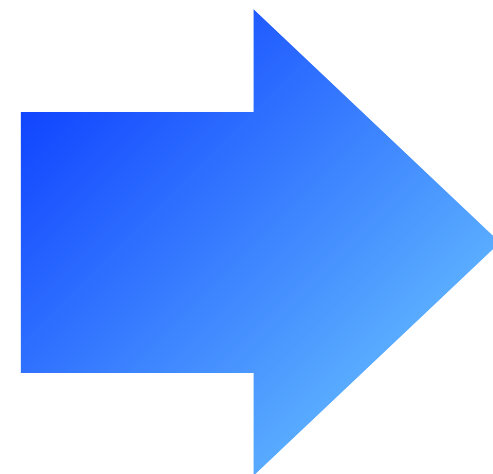
```
▶ group_0.shape  
(107, 5)  
[ ] group_1.shape  
(1378, 5)  
[ ] group_2.shape  
(288, 5)  
[ ] group_3.shape  
(331, 5)
```


Preparación de datos

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 2104 entries, 0 to 2103  
Data columns (total 10 columns):  
#   Column      Non-Null Count  Dtype    
---  ---      -  
0   Origen      2104 non-null   object   
1   Cliente     2104 non-null   object   
2   Estado      2104 non-null   object   
3   Población   2104 non-null   object   
4   N_Cliente   2104 non-null   int64    
5   Cajas       2104 non-null   int64    
6   Ruta        2104 non-null   int64    
7   Costo_caja  2104 non-null   float64   
8   Longitud    2104 non-null   float64   
9   Latitud     2104 non-null   float64   
dtypes: float64(3), int64(3), object(4)  
memory usage: 164.5+ KB
```

Transformación de variables
categóricas a numéricas



```
[17] df.drop(["Origen", "N_Cliente"], axis="columns", inplace=True)
```

```
[18] categorical_col = []  
     for column in df.columns:  
         if df[column].dtype == object and len(df[column].unique()) > 5:  
             categorical_col.append(column)
```

```
[19] categorical_col  
     ['Cliente', 'Estado ', 'Población ']
```

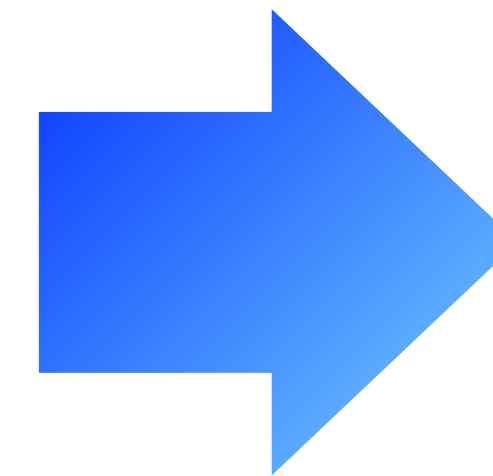
```
[20] from sklearn.preprocessing import LabelEncoder
```

```
[21] le = LabelEncoder()  
     for column in categorical_col:  
         df[column] = le.fit_transform(df[column])
```


Preparación de datos

```
[27] K = 4
```

```
▶ kmeans = KMeans(n_clusters=4, init="random", n_init=10, max_iter=300, algorithm="elkan")
```



```
[36] group_0.shape  
(525, 9)
```

```
[37] group_1.shape  
(532, 9)
```

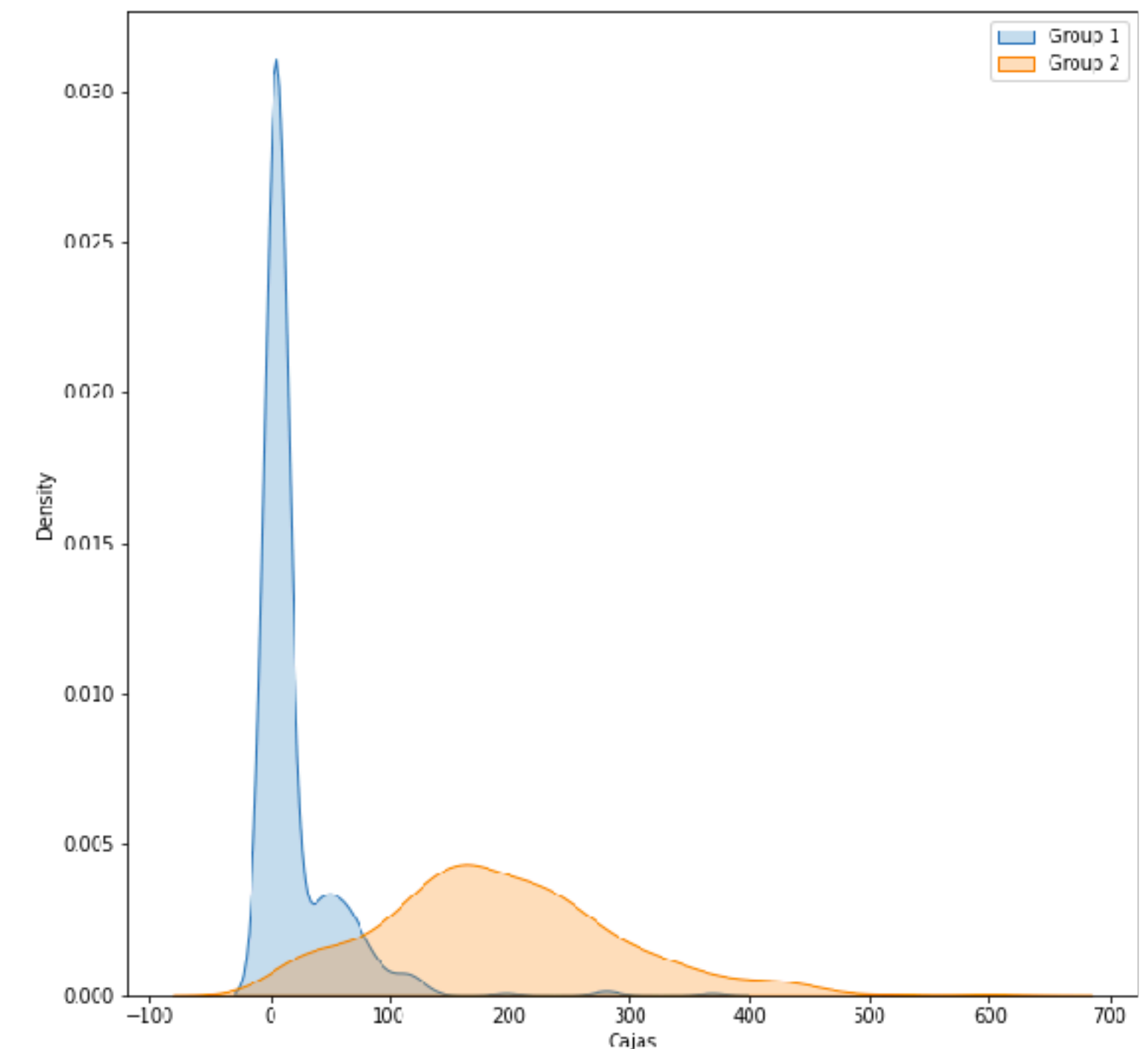
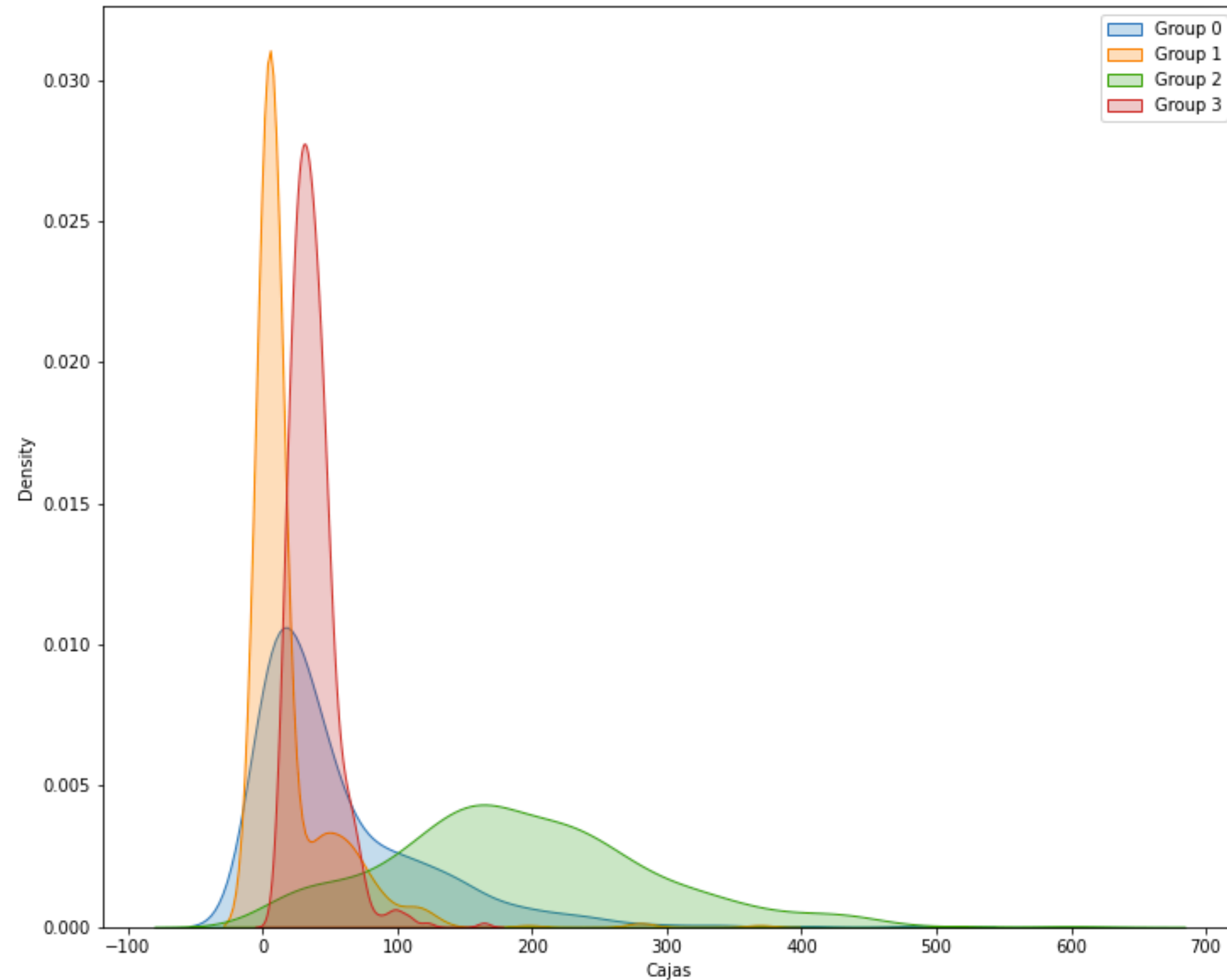
```
[38] group_2.shape  
(522, 9)
```

```
[39] group_3.shape  
(525, 9)
```

Resultados

Datos agrupados en base al volumen de cajas distribuidas.

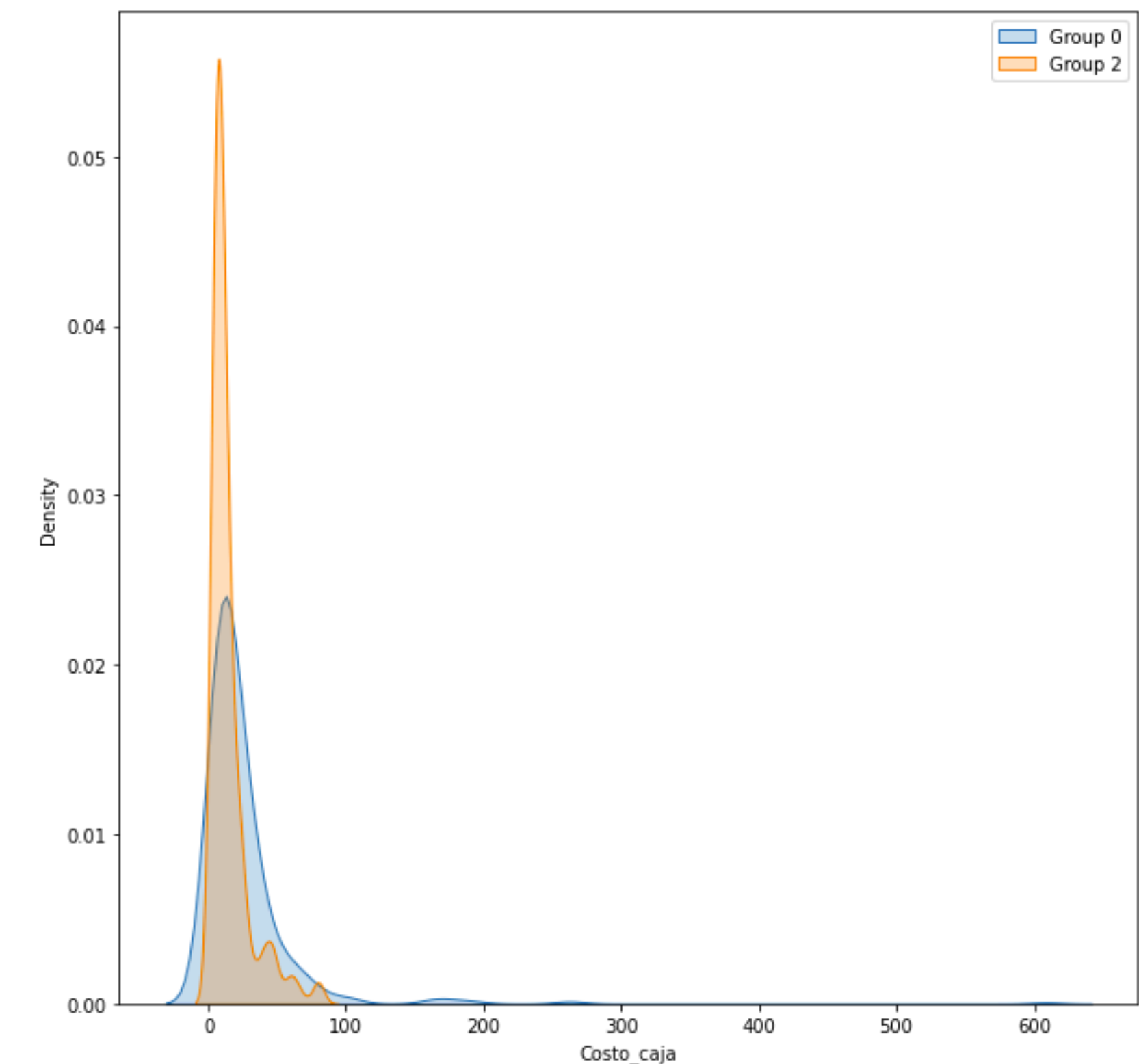
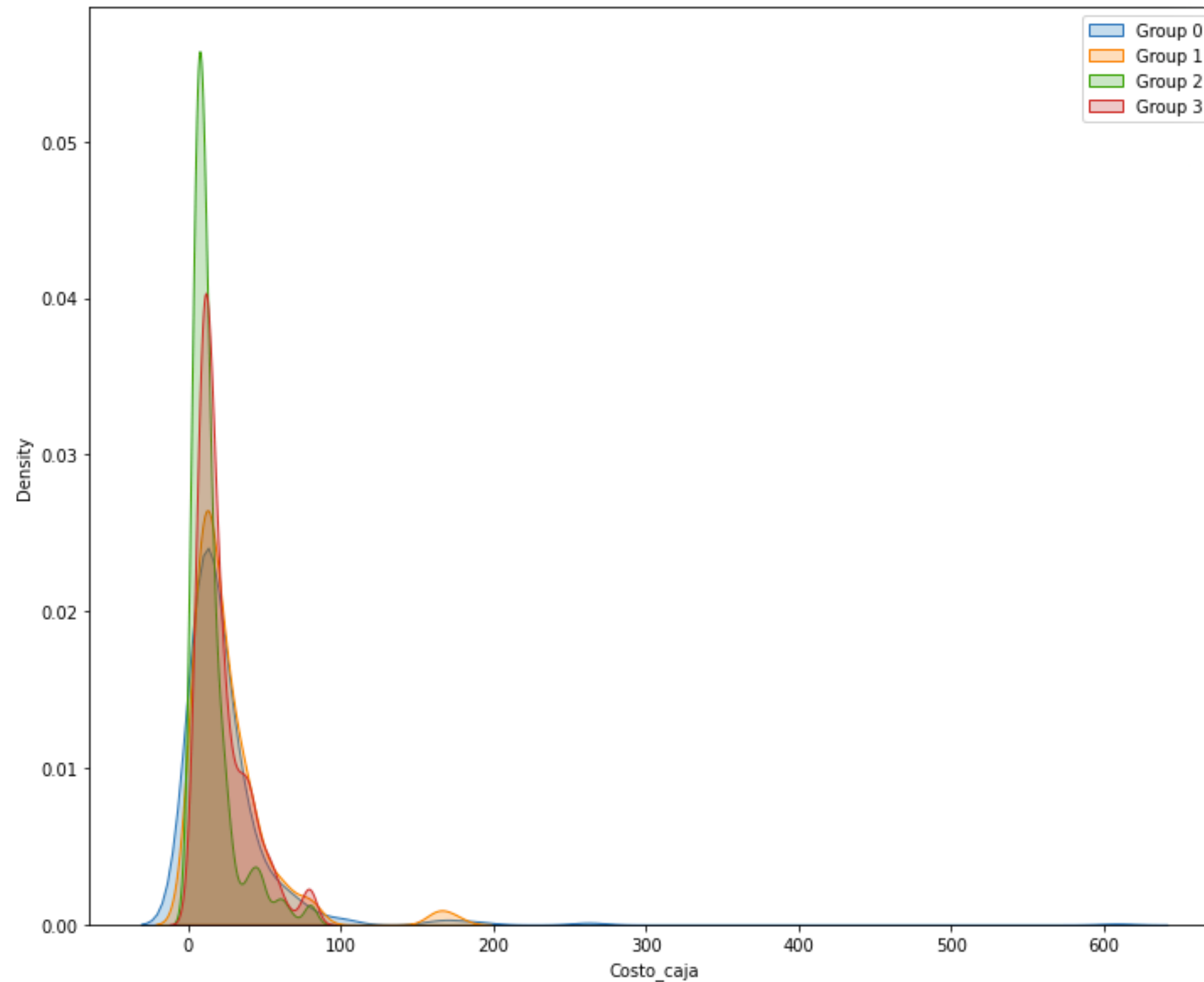
Entre el grupo 1 y grupo 2, está la mayor variación de volumen de cajas entregadas, esto es especialmente importante ya que determina la capacidad de los vehículos para hacer las entregas, a mayor cantidad de cajas mayor capacidad requerida.



Resultados

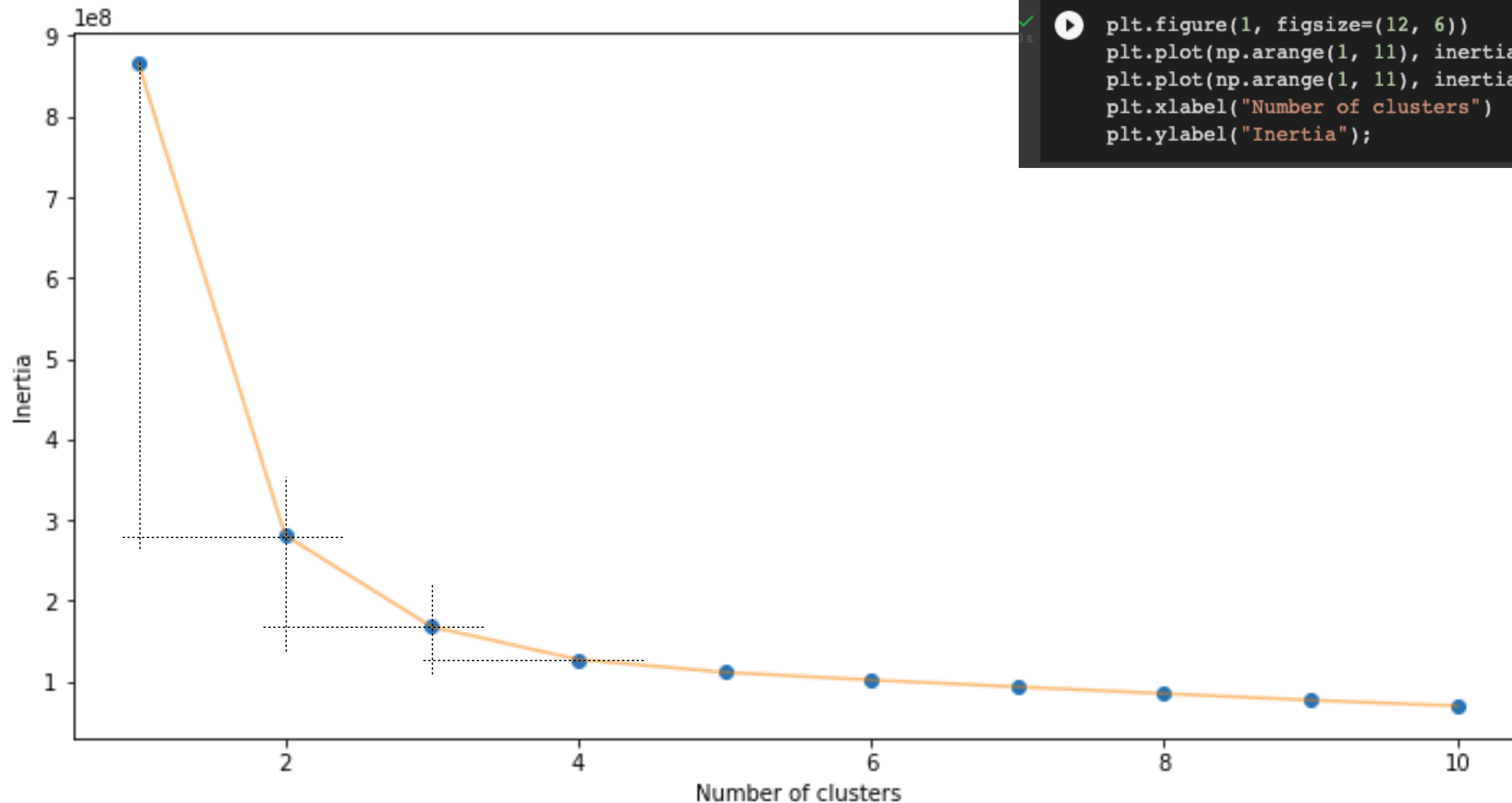
Datos agrupados en base al costo de caja entregada.

El costo por caja, es una variable que está en un rango bastante uniforme en los 4 grupos, por lo que en foco de la decisión estaría en función del volumen desplazado.



Centros de abasto

Método de CODO



```
[54] inertia = []
      for n in range(1, 11):
          model = KMeans(n_clusters=n, init="random", n_init=10, max_iter=300, algorithm="full")
          model.fit(X)
          inertia.append(model.inertia_)

plt.figure(1, figsize=(12, 6))
plt.plot(np.arange(1, 11), inertia, 'o')
plt.plot(np.arange(1, 11), inertia, '-', alpha=0.5)
plt.xlabel("Number of clusters")
plt.ylabel("Inertia");
```

CEDIS

3



Centros de abasto

```
[14] K = 4

[15] kmeans = KMeans(n_clusters=K, init="random", n_init=10, max_it

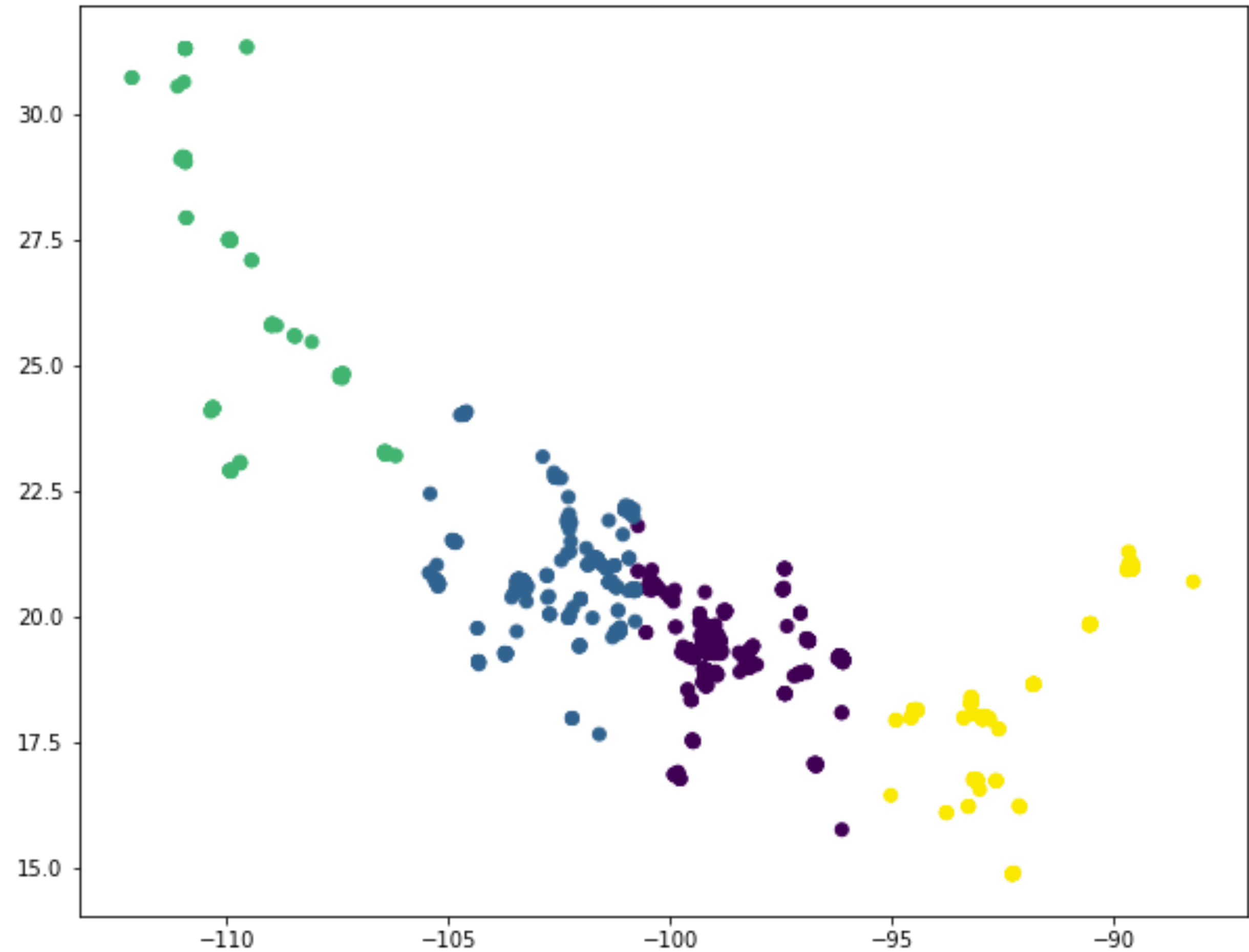
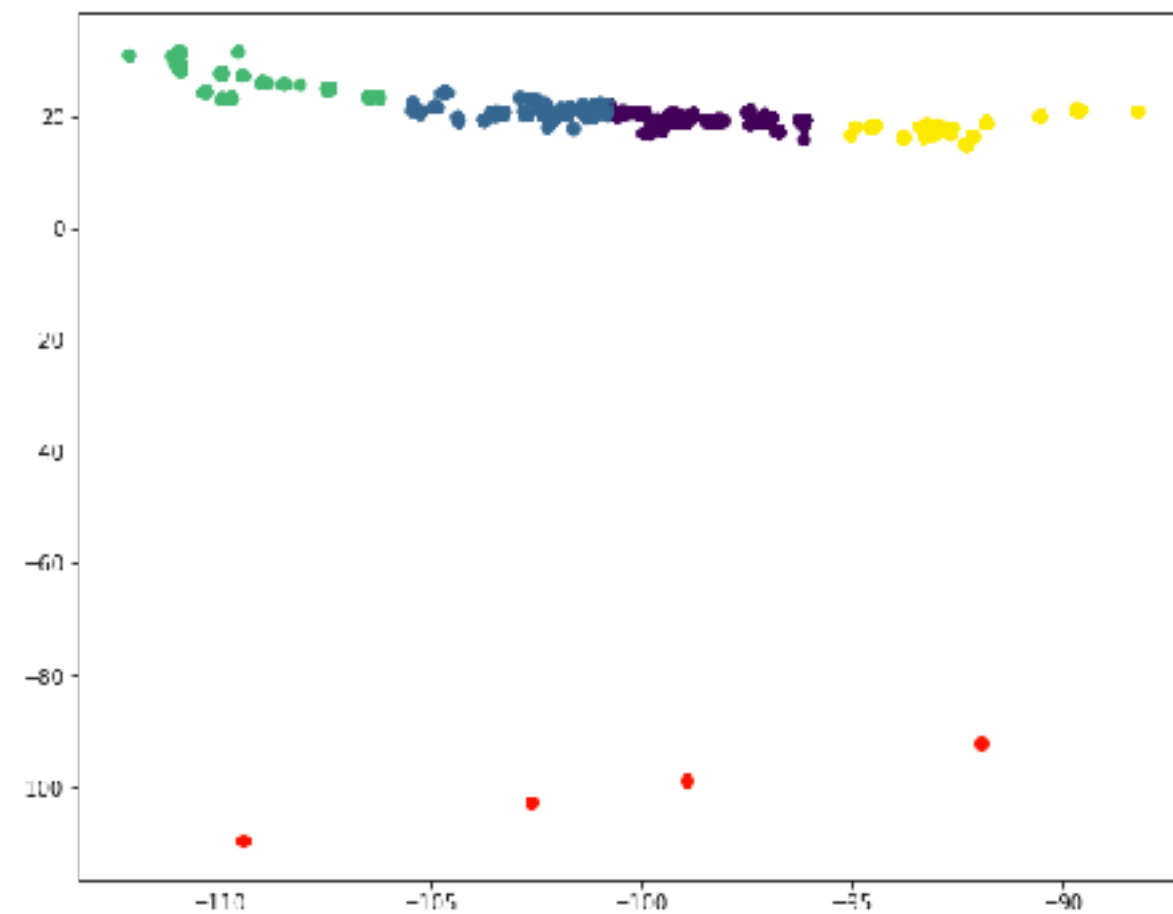
[16] kmeans.fit(X)

      KMeans(algorithm='elkan', init='random', n_clusters=4)

[17] centers = kmeans.cluster_centers_

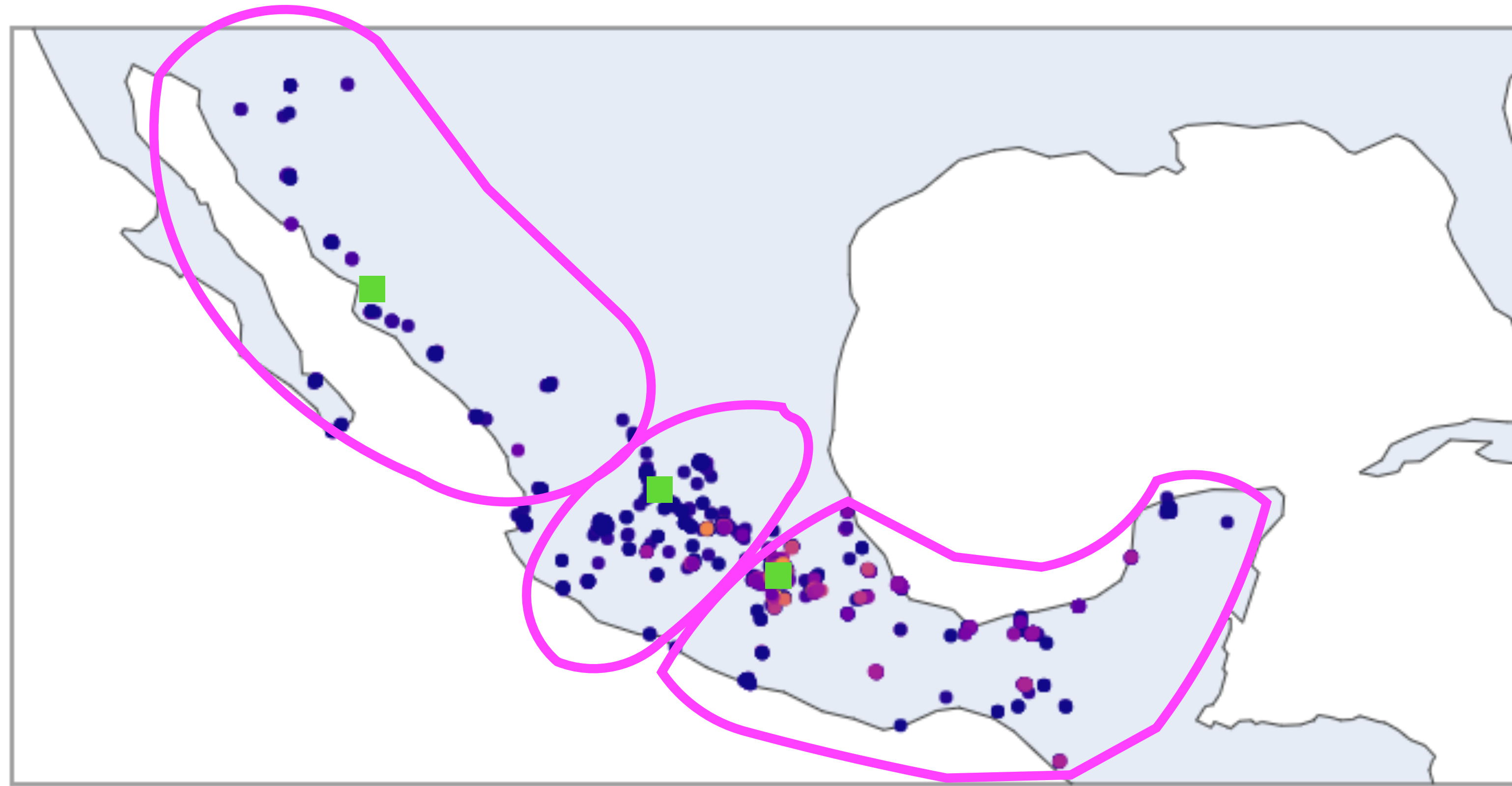
▶ y_pred = kmeans.predict(X)

▶ fig = plt.figure(figsize=(10, 8))
  plt.scatter(X[:, 0], y[:, 0], c=y_pred)
  plt.scatter(centers[:, 0], centers[:, 0], c='r');
```



Centros de abasto

Volumen de cajas distribuidas



David González

**¡Gracias por
su atención!**

David González

