

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
df = pd.read_excel('Corredores.xlsx')
df.head()
```

	Pais	Corredor	Lugar	Genero	Edad	Ventas
0	GBR	3475	3592.0	Male	52.0	4208
1	NY	13594	13853.0	Female	40.0	943
2	FRA	12012	12256.0	Male	31.0	9156
3	NaN	10236	10457.0	Female	33.0	4491
4	NY	9476	9686.0	Male	33.0	3008

```
df.info()
```

 <class 'pandas.core.frame.DataFrame'>
 RangeIndex: 1000 entries, 0 to 999
 Data columns (total 6 columns):
 # Column Non-Null Count Dtype
 -- - - - -
 0 Pais 994 non-null object
 1 Corredor 1000 non-null int64
 2 Lugar 999 non-null float64
 3 Genero 996 non-null object
 4 Edad 998 non-null float64
 5 Ventas 1000 non-null int64
 dtypes: float64(2), int64(2), object(2)
 memory usage: 47.0+ KB

```
df.describe()
```

	Corredor	Lugar	Edad	Ventas
count	1000.000000	999.000000	998.000000	1000.000000
mean	11745.475000	11975.069069	39.139279	4792.511000
std	6769.504775	6871.951154	12.772826	3008.418068
min	-2.000000	-111.000000	-63.000000	-494.000000
25%	5929.250000	6084.500000	31.000000	2238.000000
50%	11683.000000	11885.000000	38.000000	4920.000000
75%	17591.750000	17925.500000	46.000000	7345.000000
max	23290.000000	23662.000000	200.000000	9999.000000

```
df.isnull().sum()
```

```
Pais      6  
Corredor  0  
Lugar     1  
Genero    4  
Edad      2  
Ventas    0  
dtype: int64
```

```
df.dropna(subset=["Pais"], inplace=True)
```

```
df.dropna(subset=["Lugar"], inplace=True)
```

```
df.dropna(subset=["Genero"], inplace=True)
```

```
df.dropna(subset=["Edad"], inplace=True)
```

```
df.isnull().sum()
```

```
Pais      0  
Corredor  0  
Lugar     0  
Genero    0  
Edad      0  
Ventas    0  
dtype: int64
```

```
df.Pais.unique()
```

```
array(['GBR', 'NY', 'FRA', 'NJ', 'CA', 'NED', 'B.C', 'ITA', 'GER', 'SWE',  
       'DEN', 'FL', 'BRA', 'VA', 'MA', 'KY', 'IL', 'RSA', 'CAN', 'GA',  
       'MEX', 'CT', 'KOR', 'MO', 'NM', 'NZL', 'LAN', 'JPN', 'ESP', 'MI',  
       'PA', 'SUI', 'OR', 'TX', 'LA', 'VEN', 'PER', 'IRL', 'MN', 'DC',  
       'TN', 'NC', 'FIN', 'NOR', 'OH', 'ARG', 'CHI', 'AUS', 'PR', 'BEL',  
       'WA', 'MD', 'AZ', 'SD', 'AUT', 'SC', 'ID', 'VT', 'COL', 'ME', 'WI',  
       'PHI', 'CO', 'SUR', '-', 'LUX', 'AK', 'GAU', 'BER', 'HI', 'PAR',  
       'NH', 'IN'], dtype=object)
```

```
indexNames = df[ (df['Pais'] == "-") ].index  
df.drop(indexNames, inplace=True)
```

```
df.Pais.unique()
```

```
array(['GBR', 'NY', 'FRA', 'NJ', 'CA', 'NED', 'B.C', 'ITA', 'GER', 'SWE',
       'DEN', 'FL', 'BRA', 'VA', 'MA', 'KY', 'IL', 'RSA', 'CAN', 'GA',
       'MEX', 'CT', 'KOR', 'MO', 'NM', 'NZL', 'LAN', 'JPN', 'ESP', 'MI',
       'PA', 'SUI', 'OR', 'TX', 'VEN', 'PER', 'IRL', 'MN', 'TN', 'NC',
       'FIN', 'NOR', 'OH', 'ARG', 'CHI', 'AUS', 'PR', 'BEL', 'WA', 'MD',
       'AZ', 'SD', 'AUT', 'SC', 'ID', 'VT', 'COL', 'ME', 'WI', 'PHI',
       'CO', 'SUR', 'LUX', 'AK', 'GAU', 'BER', 'HI', 'PAR', 'NH', 'IN'],
      dtype=object)
```

df['Pais'].value_counts()

df['Pais'].replace(['NY'], ['USA'], inplace=True)

df['Pais'].replace(['CA'], ['USA'], inplace=True)

df['Pais'].replace(['LA'], ['USA'], inplace=True)

df['Pais'].replace(['DC'], ['USA'], inplace=True)

df.Pais.unique()



```
array(['GBR', 'USA', 'FRA', 'NJ', 'NED', 'B.C', 'ITA', 'GER', 'SWE',
       'DEN', 'FL', 'BRA', 'VA', 'MA', 'KY', 'IL', 'RSA', 'CAN', 'GA',
       'MEX', 'CT', 'KOR', 'MO', 'NM', 'NZL', 'LAN', 'JPN', 'ESP', 'MI',
       'PA', 'SUI', 'OR', 'TX', 'VEN', 'PER', 'IRL', 'MN', 'TN', 'NC',
       'FIN', 'NOR', 'OH', 'ARG', 'CHI', 'AUS', 'PR', 'BEL', 'WA', 'MD',
       'AZ', 'SD', 'AUT', 'SC', 'ID', 'VT', 'COL', 'ME', 'WI', 'PHI',
       'CO', 'SUR', 'LUX', 'AK', 'GAU', 'BER', 'HI', 'PAR', 'NH', 'IN'],
      dtype=object)
```

```
from sklearn import preprocessing as pp
le = pp.LabelEncoder()
le.fit(df.Pais.unique())
df['Pais'] = le.transform(df.Pais)
```

df.Pais.unique()



```
array([21, 63, 18, 44, 42, 5, 28, 22, 60, 14, 17, 8, 64, 34, 31, 25, 55,
       9, 19, 37, 13, 30, 40, 45, 47, 32, 29, 15, 38, 50, 58, 49, 62, 65,
       52, 27, 39, 61, 41, 16, 46, 48, 1, 10, 2, 54, 6, 67, 35, 4, 57,
       3, 56, 24, 66, 12, 36, 68, 53, 11, 59, 33, 0, 20, 7, 23, 51, 43,
       26])
```

df.Genero.unique()



```
array(['Male', 'Female'], dtype=object)
```

df['Genero'].replace(['Male', 'Female'], [0, 1], inplace=True)

df.Genero.unique()



```
array([0, 1])
```

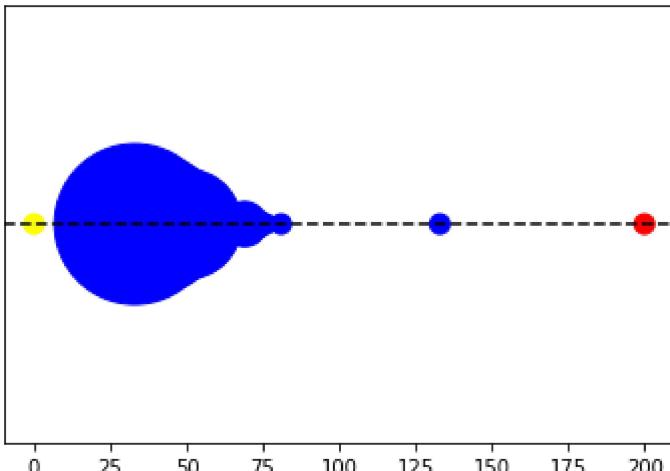
df.drop(df[df.Ventas < 0].index, inplace=True)

df.drop(df[df.Corredor < 0].index, inplace=True)

```
df.drop(df[df.Lugar < 0].index, inplace=True)

df.drop(df[df.Edad < 0].index, inplace=True)

import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
edades = np.array(df['Edad'])
edad_unique, counts = np.unique(edades, return_counts=True)
sizes = counts*100
colors = ['blue']*len(edad_unique)
colors[-1] = 'red'
colors[0] = 'yellow'
plt.axhline(1, color='k', linestyle='--')
plt.scatter(edad_unique, np.ones(len(edad_unique)), s=sizes, color=colors)
plt.yticks([])
plt.show()
print(edad_unique)
print(counts)
```



```
[ 0.  18.  20.  21.  22.  23.  24.  25.  26.  27.  28.  29.  30.  31.
 32.  33.  34.  35.  36.  37.  38.  39.  40.  41.  42.  43.  44.  45.
 46.  47.  48.  49.  50.  51.  52.  53.  54.  55.  56.  57.  58.  59.
 60.  61.  62.  63.  64.  65.  66.  67.  69.  70.  71.  72.  73.  76.
 81. 133. 200.]
[ 1  1  6 11 16 16 11  7 11 15 15 21 47 51 50 64 34 25 25 23 24 17 47 42
 38 30 28 11 19 13 18 15 30 24 18 19 12  8  9  4  6  2  6  5  6  4  1  2
 2  4  5  3  1  2  2  1  1  1  1]
```

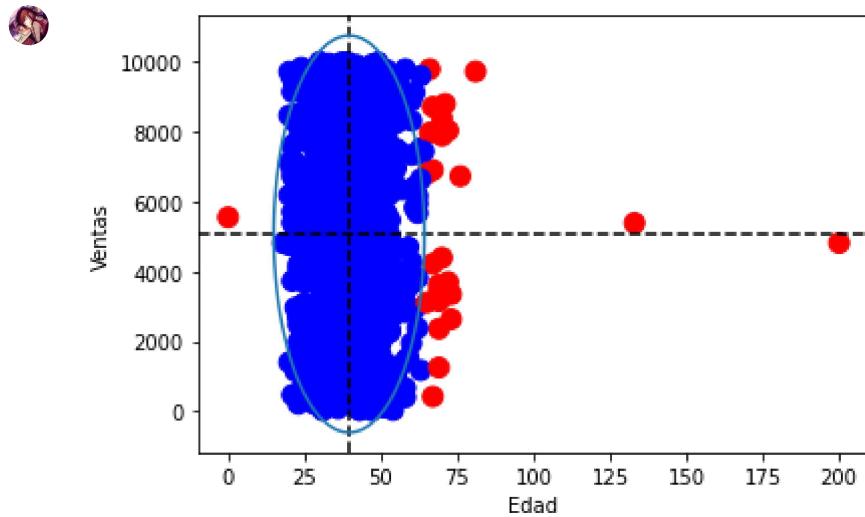
```
from math import pi
ventas = np.array(df['Ventas'])
media = (ventas).mean()
std_x = (ventas).std()*2
media_y = (edades).mean()
std_y = (edades).std()*2
colors = ['blue']*len(ventas)
for index, x in enumerate(ventas):
    ...
```

```

if abs(x-media_x) > std_x:
    colors[index] = 'green'

for index, x in enumerate(edades):
    if abs(x-media_y) > std_y:
        colors[index] = 'red'
plt.scatter(edades, ventas, s=100, color=colors)
plt.axhline(media, color='k', linestyle='--')
plt.axvline(media_y, color='k', linestyle='--')
v=media #y-position of the center
u=media_y #x-position of the center
b=std_x #radius on the y-axis
a=std_y #radius on the x-axis
t = np.linspace(0, 2*pi, 100)
plt.plot( u+a*np.cos(t) , v+b*np.sin(t) )
plt.xlabel('Edad')
plt.ylabel('Ventas')
plt.show()

```

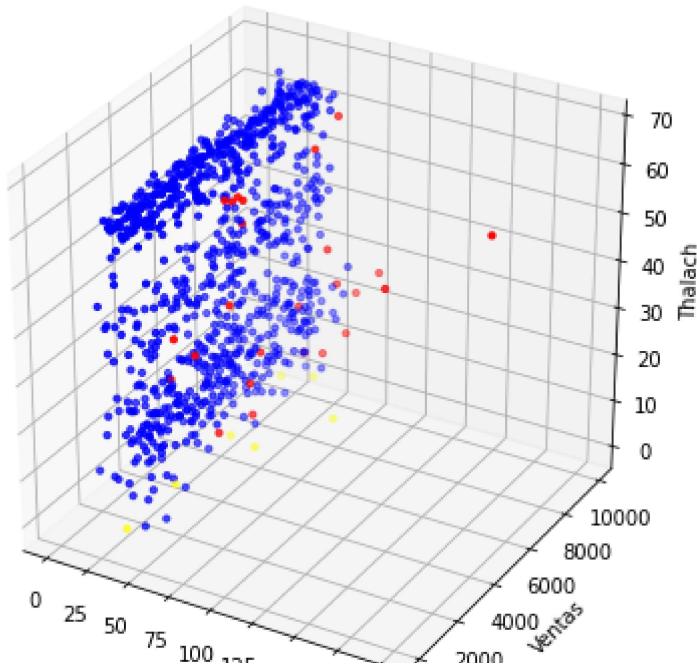


```

from mpl_toolkits.mplot3d import Axes3D
fig = plt.figure(figsize=(7,7))
ax = fig.gca(projection='3d')
pais = np.array(df['Pais'])
media_z = (pais).mean()
std_z = (pais).std()*2
for index, x in enumerate(pais):
    if abs(x-media_z) > std_z:
        colors[index] = 'yellow'
ax.scatter(edades, ventas, pais, s=10, c=colors)
plt.xlabel('Edad')
plt.ylabel('Ventas')
ax.set_zlabel('Thalach')
plt.show()

```





```
!pip install pyod
```

Requirement already satisfied: pyod in /usr/local/lib/python3.6/dist-packages (0.8.1)
 Requirement already satisfied: scikit-learn>=0.19.1 in /usr/local/lib/python3.6/dist-packages
 Requirement already satisfied: six in /usr/local/lib/python3.6/dist-packages (from pyod)
 Requirement already satisfied: numpy>=1.13 in /usr/local/lib/python3.6/dist-packages (from pyod)
 Requirement already satisfied: joblib in /usr/local/lib/python3.6/dist-packages (from pyod)
 Requirement already satisfied: matplotlib in /usr/local/lib/python3.6/dist-packages (from pyod)
 Requirement already satisfied: soum in /usr/local/lib/python3.6/dist-packages (from pyod)
 Requirement already satisfied: combo in /usr/local/lib/python3.6/dist-packages (from pyod)
 Requirement already satisfied: numba>=0.35 in /usr/local/lib/python3.6/dist-packages (from pyod)
 Requirement already satisfied: scipy>=0.19.1 in /usr/local/lib/python3.6/dist-packages (from pyod)
 Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.6/dist-packages
 Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.6/dist-packages (from pyod)
 Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /usr/local/lib/python3.6/dist-packages
 Requirement already satisfied: python-dateutil>=2.1 in /usr/local/lib/python3.6/dist-packages
 Requirement already satisfied: setuptools in /usr/local/lib/python3.6/dist-packages (from pyod)
 Requirement already satisfied: llvmlite<0.32.0,>=0.31.0dev0 in /usr/local/lib/python3.6/dist-packages

```
from pyod.models.knn import KNN
from pyod.models.pca import PCA
import pandas as pd
```

```
FullEDA = pd.DataFrame(data={'edad':edades,'ventas':ventas, 'pais':pais})
```

```
clf = KNN(contamination=0.18)
#clf = PCA(contamination=0.17)
clf.fit(FullEDA)
```

```
X=FullEDA
clf.fit(X)
scores_pred = clf.decision_function(X)
```

```
...  
y_pred = clf.predict(X)
```

```
scores_pred
```



```
array([ 35.22782991, 45.98912915, 30.16620626, 21.9317122 ,  
       23.36664289, 42.67317659, 32.66496594, 31. ,  
       46.59399103, 37.85498646, 19.26136028, 39.97499218,  
       42.23742416, 25.86503431, 52.92447449, 27.23967694,  
       46.27094121, 25.57342371, 42.57933771, 39.8246155 ,  
       40.37325848, 49.34571917, 38.44476557, 25.33771892,  
       27.44084547, 42.73172124, 62.49799997, 41.02438299,  
       47.10626285, 41.43669871, 37.4566416 , 29.22327839,  
       41.98809355, 27.60434748, 30.87069808, 31.63858404,  
       36.138622 , 27.03701167, 40.47221269, 42.43819035,  
       49.96999099, 37.10795063, 173.90802167, 22.22611077,  
       34.33656943, 40.62019202, 30.01666204, 58.14636704,  
       47.04253395, 50.019996 , 35.63705936, 44.56455991,  
       32.61901286, 36.35931793, 44.68780594, 34.71310992,  
       66.51315659, 20.24845673, 23.4520788 , 41.0487515 ,  
       32.89376841, 39.49683532, 18.11077028, 30.08321791,  
       31.79622619, 24.69817807, 49.92995093, 29.52964612,  
       18.70828693, 35.25620513, 37.16180835, 45.93473631,  
       40.31128874, 33.30165161, 35.36947837, 67.19374971,  
       25.01999201, 31.144823 , 43.01162634, 43.93176527,  
       34.20526275, 30.5122926 , 38.96151948, 36.29049462,  
       43.74928571, 35.35533906, 29.68164416, 39.57271787,  
       44.59820624, 55.3714728 , 35.52463934, 26.03843313,  
       36.59234893, 36.138622 , 48.35286961, 43.73785546,  
       47.138095 , 68.36665854, 52.50714237, 29.58039892,  
       35.44009029, 50.8920426 , 22.84731932, 46.33573135,  
       54.35071297, 68.02940541, 46.06517123, 29.89983278,  
       50.57667447, 54.01851534, 70.80960387, 26.43860813,  
       23.53720459, 31.40063694, 26.64582519, 52.95280918,  
       36.29049462, 41.30375286, 29.49576241, 42.20189569,  
       30.11644069, 17.72004515, 56.36488268, 25.07987241,  
       19.26136028, 32.64965543, 55.54277631, 21.30727575,  
       28.47806173, 36.05551275, 48.20788317, 32.31098884,  
       35.34119409, 39.05124838, 39.3192065 , 60.09159675,  
       41.30375286, 30.52867504, 49.92995093, 50.12983144,  
       21.67948339, 35.44009029, 39.92492956, 35.73513677,  
       54.08326913, 52.9905652 , 38.02630668, 42.95346319,  
       51.31276644, 30.23243292, 16.91153453, 25.19920634,  
       50.35871325, 20.1246118 , 34.98571137, 29.22327839,  
       57.92236183, 43.66921112, 28.17800561, 36.06937759,  
       41.54515616, 33.1662479 , 20.46948949, 35.28455753,  
       41.64132563, 36. , 56.39148872, 40.81666326,  
       28.86173938, 34.02939905, 24.04163056, 32.66496594,  
       48.02082881, 41.24318125, 34.71310992, 39.2173431 ,  
       42.29657197, 28.17800561, 24.04163056, 44.83302354,  
       12.76714533, 29.01723626, 30.08321791, 31.1608729 ,  
       25.51470164, 40.03748244, 41.49698784, 29.54657341,  
       42.42640687, 54.15717866, 34.2636834 , 48.55924217,  
       31. , 46.27094121, 35.52463934, 25.01999201,  
       42.53234064, 45.70557953, 49.12229636, 40.01249805,  
       33.2565783 , 36.72873534, 43.92038251, 28.72281323,  
       26.4952826 , 38.20994635, 26.57066051, 43.15089802,  
       46.10856753, 38.92300091, 31.25699922, 39.45883931,  
       31.01612484, 51.26402247, 36.40054945, 26.32489316,  
       34.40930107, 44.59820624, 46.17358552, 25.47547841,  
       33.2565783 , 33.67491648, 26.94438717, 33.43650699,  
       32.984845 , 35.22782991, 37.21558813, 42.98837052,
```

41.30375286,	37.22902094,	30.7408523 ,	49.13247399,
42.52058325,	43.74928571,	30.43024811,	51.97114584,
21.54065923,	32.20248438,	26.43860813,	45.63989483,
35.90264614,	40.93897898,	47.24404724,	15.5241747 ,
34.2636834 ,	50.94114251,	31.591138 ,	26. ,
33.15116891,	29.08607914,	46.49731175,	30.14962686,
36. ,	27.29468813,	43.87482194,	38.79432948,
21.9544984 ,	31.44837039,	36.06937759,	25.86503431,
18.46618531,	49.37610758,	36.05551275,	41.19465985,
28.46049894,	19.74841766,	36.61966685,	30.33150178,
45.32107677,	50.14977567,	34.55430509,	43.65775991,
32.38826948,	32.58834147,	26.57066051,	33.58571125,
38.58756276,	36.138622 ,	54.49770637,	33.2565783 ,
45.49725266,	30.47950131,	69. ,	29.27456234,
28.35489376,	30.3644529 ,	54.81788029,	46.43274706,
19.67231557,	27.09243437,	37.12142239,	36.29049462,
46.69047012,	38.37968212,	43.92038251,	30.56141358,
31.17691454,	23.04343724,	19.6468827 ,	30.8058436 ,
47.77028365,	30.82207001,	31.78049716,	48.38388161,
44.50842617,	41.30375286,	30.9354166 ,	41.15823125,
49.25444142,	25.3179778 ,	31.19294792,	60.09159675,
46.10856753,	41.92851059,	36.79673899,	25.94224354,
40.47221269,	28.39013913,	38.02630668,	19.74841766,
34.78505426,	52.44044241,	38.83297568,	42.16633728,
29.42787794,	36.18010503,	18.70828693,	31.57530681,
48.96937819,	20.78460969,	22.09072203,	34.525353 ,
41.09744518,	41.53311931,	46.71188286,	36. ,
55.19057891,	51.34199061,	27.58622845,	28.80972058,
48.27007354,	54.7813837 ,	24.35159132,	28.87905816,
39.37003937,	30.757113 ,	43.18564576,	50.15974482,
44.29446918,	27.58622845,	32.61901286,	26.70205985,
45.475268 ,	26.34387974,	44.87761134,	20.46948949,
39.11521443,	32.26453161,	45.86937976,	24.61706725,
35.58089375,	36.138622 ,	32.01562119,	57.92236183,
40.64480287,	63.69458376,	21.47091055,	45.12205669,
47.85394446,	38.07886553,	39.11521443,	37.33630941,
41.30375286,	29.68164416,	54.46099522,	40.60788101,
42.72001873,	46.78675026,	36.06937759,	39.63584237,
26.19160171,	40.80441153,	37.65634077,	43.11612227,
48.76474136,	28.47806173,	46.36809248,	45.01110974,
45.45327271,	29.42787794,	30.11644069,	50.6359556 ,
46.8508271 ,	39. ,	42.67317659,	47.39198244,
27.45906044,	48.09365863,	45.39823785,	47.58150901,
46.23851209,	42.95346319,	68.36665854,	34.84250278,
33. ,	33.9705755 ,	51.8748494 ,	28.23118843,
41.0487515 ,	40.31128874,	43.46262762,	50.67543784,
37.4566416 ,	46.05431576,	45.88027899,	30.3644529 ,
37.96050579,	52.35456045,	25.07987241,	46.19523785,
28.7923601 ,	34.92849839,	23.02172887,	46.19523785,
21. ,	40.43513324,	33.2565783 ,	38.58756276,
51.8748494 ,	31.0322413 ,	41.82104733,	36.81032464,
29.9833287 ,	96.00520819,	29.9833287 ,	30.52867504,
40.57092555,	53.57238094,	28.98275349,	31.20897307,
31.46426545,	54.85435261,	28.4956137 ,	38.48376281,
34.0293905,	47.85394446,	43.24349662,	28.01785145,
26.64582519,	42.04759208,	50.08991915,	48.35286961,
34.72751071,	21.09502311,	23.38803113,	44.37341546,
98.23950326,	33. ,	27.73084925,	35.44009029,

25.33771892,	46.8294779 ,	34. ,	26.87005769,
25.17935662,	37.12142239,	39.87480407,	34.36568055,
45.89117562,	29.27456234,	50.3189825 ,	57.41950888,
33.24154028,	34.49637662,	31.63858404,	42.57933771,
35.07135583,	40.70626487,	35.39774004,	48.3011387 ,
34.0881211 ,	33.18132005,	38.18376618,	44.14748011,
38.13135193,	34.59768778,	36.59234893,	40.6571027 ,
50.7838557 ,	23.34523506,	29.84962311,	44.19275959,
31.70173497,	34.94281042,	36.08323711,	32.984845 ,
28.87905816,	29.06888371,	36.35931793,	47.81213235,
44.04543109,	37.70941527,	15.93737745,	28.7923601 ,
24.16609195,	56.9473441 ,	21.1896201 ,	28.47806173,
47.80167361,	49.2036584 ,	34.525353 ,	29.08607914,
44.51965858,	39.96248241,	30.14962686,	33.9705755 ,
24. ,	38.91015292,	23.34523506,	38.62641583,
46.34652091,	40.43513324,	44.87761134,	38.92300091,
40.62019202,	47.59201614,	32.26453161,	40.21193853,
23.02172887,	40.5092582 ,	30.59411708,	53.00943312,
21.37755833,	27.62245463,	39.87480407,	40.07492982,
47.29693436,	53.46026562,	56.10704056,	32.984845 ,
34.42382896,	27.67670501,	36.5650106 ,	46.88283268,
18.78829423,	49.96999099,	36.81032464,	36.41428291,
29.42787794,	46.41120554,	46.19523785,	31.84336666,
35.22782991,	30.61045573,	42.63801121,	34.92849839,
28.86173938,	44.04543109,	25.49509757,	39.61060464,
42.72001873,	34.525353 ,	54.95452666,	29.20616373,
44.50842617,	36.93237063,	33.98529094,	28.51315486,
29.54657341,	31. ,	21.02379604,	44.19275959,
40.91454509,	41.01219331,	40.26164428,	36.61966685,
45.31004304,	30.23243292,	47.85394446,	37.54996671,
23.43074903,	41.40048309,	48.09365863,	38.02630668,
45.67274899,	38.65229618,	41.48493703,	47.68647607,
44.04543109,	45.86937976,	47.94788838,	51.59457336,
53.76802024,	33.07567082,	39.20459157,	25.35744467,
26.88865932,	14.35270009,	48.57983121,	41.48493703,
65.9241989 ,	39. ,	27.09243437,	50.15974482,
29.44486373,	34.40930107,	21. ,	29.22327839,
46.95742753,	42.95346319,	31.19294792,	34.84250278,
35.44009029,	41.59326869,	47.89572006,	36.27671429,
43.94314509,	40.62019202,	29.15475947,	51.92301994,
43.39354791,	40.32369031,	39.08964057,	36.63331817,
46.35730795,	25.01999201,	56.90342696,	39.56008089,
35.05709629,	34.94281042,	50.81338406,	44.14748011,
57.28874235,	42.55584566,	22.04540769,	23.85372088,
31.20897307,	20.09975124,	38.92300091,	46.09772229,
23.53720459,	42.93017587,	44.83302354,	33.3166625 ,
37.44329045,	47.94788838,	27.58622845,	48.51803788,
17.72004515,	23.4520788 ,	42.48529157,	41.10960958,
42.42640687,	48.74423043,	37.69615365,	44.01136217,
55.76737397,	23.62202362,	52.15361924,	48.96937819,
32.15587038,	45.08880127,	35.86084215,	47.50789408,
46.57252409,	24.91987159,	38.91015292,	33.74907406,
30.61045573,	45.48626166,	41.70131892,	45.27692569,
35.86084215,	45.32107677,	36.90528417,	56.4092191 ,
35.58089375,	50.81338406,	25.01999201,	47.39198244,
47.86439178,	44.45222154,	43.41658669,	50.28916384,
16.1245155 ,	36.46916506,	27.07397274,	35.90264614,

31.01612484,	42.23742416,	14.14213562,	25.57342371,
29.06888371,	33.67491648,	12.08304597,	51.15662225,
50.51732376,	39.11521443,	29.88310559,	35.56683849,
30.56141358,	38.50973903,	29.47880595,	40.22437072,
54.53439282,	42.43819035,	28.58321186,	31.70173497,
40.91454509,	24.43358345,	66.46803743,	24.59674775,

```
FullEDA [y_pred == 1]
```



	edad	ventas	pais
14	33.0	9578	63
26	41.0	6857	8
42	200.0	4817	63
47	67.0	6904	19
56	58.0	1448	63
...
839	44.0	7717	63
892	35.0	9722	3
903	30.0	7238	9
917	39.0	3647	18
930	55.0	3871	22

69 rows × 3 columns

38.65229618. 24.8394847 . 39.06404997. 41.42463035.

```
FullEDA.to_excel("entrenada.xlsx")
```

38.05259518,	36.78314832,	51.31276644,	72.56031973,
27.03701167,	26.92582404,	26.70205985,	43.05810028,
49.01020302,	42.63801121,	43.94314509,	21.84032967,
23.4520788 ,	40.31128874,	22.71563338,	45.01110974,
36.74234614,	30.87069808,	28.23118843,	26.92582404,
45.97825573,	34.79942528,	38.91015292,	41.70131892,
42.15447782,	19.41648784,	26.43860813,	42.53234064,
37.94733192,	48.09365863,	47.63402146,	42.72001873,
45.72745346,	48.27007354,	28.08914381,	39.44616585,
50.44799302,	43.27817002,	51.50728104,	19.23538406,
31.65438358,	46.86149806,	31.32091953,	40.75536774,
30.61045573,	40.70626487,	29.42787794,	34.59768778,
24.59674775,	41.06093034,	22.75961335,	31.17691454,
34.6121366 ,	48.96937819,	30.9354166 ,	49.81967483,
55.45268253,	21.26029163,	32.38826948,	42.01190308,
46.57252409,	36.37306696,	44.38468204,	31.40063694,
30.6757233 ,	24.8997992 ,	37.77565354,	51.66236541,
48.27007354,	38.02630668,	44.10215414,	42.77849927,
43.24349662,	46.8508271 ,	51.28352562,	36.19392214,
34.20526275,	27.29468813,	32.64965543,	23.79075451,
19.6977156 ,	59.38013136,	40.32369031,	32.38826948,

