In [ ]:
```
Sprint 11. Aprenentatge No Supervisat - Agrupació
```

In [ ]:
```
#Nivell 1
```

In [ ]:
```
#Exercici 1
#Classifica els diferents vols utilitzant l'algorisme de K-means.
```

In [1]:
```
#Llibreries import
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

In [41]:
```
#Llibreries from
from kneed import KneeLocator
from sklearn.datasets import make_blobs
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score
from sklearn.preprocessing import StandardScaler
```

In [3]:
```
#Accedim a les dades del fitxer DelayedFlights
df_vols=pd.read_csv('Python/DelayedFlights.csv', engine="python", error_bad_lines=False, warn_bad_lines=False, sep=','
)
```

In [6]:
```
# Netegem el dataset d'atributs innecesaris.
df_vols.drop(['Year','Month','DayofMonth','DayOfWeek','TailNum','Cancelled', 'CancellationCode', 'Diverted', 'UniqueCa
rrier', 'Origin', 'Dest'], axis=1, inplace=True)
```

In [7]:
```
# Suprimim valors no numerics.
df_vols=df_vols.dropna()
```

In [8]:
```
# Utilitzarem només una petita part de les instàncies del dataframe.
df_vols=df_vols.head(10)
```

In [9]:
```
df_vols
```

Out[9]:

	Unnamed: 0	DepTime	CRSDepTime	ArrTime	CRSArrTime	FlightNum	ActualElapsedTime	CRSElapsedTime	AirTime	ArrDelay	DepDelay	Dis
3	4	1829.0	1755	1959.0	1925	3920	90.0	90.0	77.0	34.0	34.0	
5	6	1937.0	1830	2037.0	1940	509	240.0	250.0	230.0	57.0	67.0	
7	11	1644.0	1510	1845.0	1725	1333	121.0	135.0	107.0	80.0	94.0	
9	16	1452.0	1425	1640.0	1625	675	228.0	240.0	213.0	15.0	27.0	
11	18	1323.0	1255	1526.0	1510	4	123.0	135.0	110.0	16.0	28.0	
12	19	1416.0	1325	1512.0	1435	54	56.0	70.0	49.0	37.0	51.0	
13	21	1657.0	1625	1754.0	1735	623	57.0	70.0	47.0	19.0	32.0	
17	26	1422.0	1255	1657.0	1610	188	155.0	195.0	143.0	47.0	87.0	
19	30	2107.0	1945	2334.0	2230	362	147.0	165.0	134.0	64.0	82.0	
23	37	1812.0	1650	1927.0	1815	422	135.0	145.0	118.0	72.0	82.0	

In [99]:

```
#features
```

```
Out[99]: array([[ 9.77075874e+00,  3.27621022e+00],
               [-9.71349666e+00,  1.12745180e+01],
               [-6.91330582e+00, -9.34755911e+00],
               [-1.08618591e+01, -1.07506350e+01],
               [-8.50038027e+00, -4.54370383e+00],
               [-4.82456978e+00, -5.20159136e+00],
               [-3.03819028e+00,  9.84354132e+00],
               [-8.17498253e+00, -6.24197227e+00],
               [ 3.91207254e+00,  9.45363489e+00],
               [-4.64425636e+00, -5.14863028e+00],
               [ 2.09082004e+00,  1.80947495e+00],
               [ 5.26539366e+00,  5.56781226e+00],
               [ 7.61826975e+00,  4.87112533e+00],
               [ 3.30512908e+00,  2.19832357e+00],
               [-9.29263277e-01,  2.48591905e+00],
               [-9.15936347e+00, -8.42060745e+00],
               [-2.70722546e+00,  1.17740016e+01],
               [-9.14073328e+00, -6.20996976e+00],
               [ 6.52709436e+00, -2.46179896e+00],
               [-2.51808095e+00, -1.02841022e+01],
               [ 1.04758084e+01,  4.81244915e+00],
               [ 3.33377923e+00,  1.76514294e-01],
               [-5.99215115e+00, -9.15499469e+00],
               [-2.32349506e+00,  5.09622862e+00],
               [-5.00425652e+00, -7.73334317e+00],
               [-4.37073312e+00,  1.06963959e+01],
               [-5.67442996e+00,  1.00474557e+01],
               [-7.29703620e+00, -5.26223728e+00],
               [-1.93482274e+00,  3.62519329e+00],
               [ 1.23826438e+00, -1.65808600e+00],
               [ 4.61611430e-01,  6.41525984e-01],
               [ 1.10051899e+01, -3.16180960e+00],
               [-1.15907471e+01, -3.15696056e+00],
               [-9.14974448e+00, -7.76392066e+00],
               [-1.51535971e+00,  1.32438867e+01],
               [-2.23515637e+00,  7.62972808e+00],
               [-5.74406569e+00, -8.43035211e+00],
               [-8.95340615e-01,  1.50380391e+01],
               [-6.20596912e+00, -8.27420333e+00],
               [-2.25685549e+00,  3.54847161e+00],
               [ 1.33906372e+00,  1.05329129e+00],
```

```
[-4.73255503e+00,  7.63445426e+00],
[-9.57877598e+00, -5.60932504e+00],
[ 9.12900992e+00, -1.95971911e+00],
[-9.12973407e+00, -1.12604459e+00],
[ 6.36046404e+00, -3.84013596e-01],
[ 2.19371415e+00, -2.70308600e-01],
[-6.68767146e+00, -7.93972198e+00],
[ 8.51288074e-01, -6.05849176e-01],
[ 8.15820063e-03,  9.91835168e+00],
[-8.57669125e-01, -9.10292988e+00],
[-4.00625012e+00,  9.31932325e+00],
[-3.63616686e+00, -8.17034264e+00],
[-3.33859769e-01,  6.51347063e+00],
[-6.57513310e+00,  7.03471456e+00],
[-4.16389081e+00,  1.41080511e+01],
[-1.01378541e+00, -1.22483510e+01],
[ 3.94531642e+00, -1.45823407e+00],
[-4.43984363e+00,  8.11321523e+00],
[ 1.36379422e+00,  3.77869211e+00],
[-4.26596164e+00, -8.46659465e+00],
[-5.03761427e+00,  1.32766057e+01],
[-4.40572753e+00, -6.03503591e+00],
[-6.93192176e+00, -9.63706535e+00],
[-4.48074544e+00, -6.37591908e+00],
[-6.33221303e+00, -8.53070601e+00],
[-5.43053284e+00,  1.03166653e+01],
[-2.47199115e-01,  5.65696609e+00],
[-4.19804792e+00, -7.28451739e+00],
[-4.78414528e-01,  9.48554890e+00],
[-1.61796671e+00,  7.95530986e+00],
[-6.26337287e+00, -6.84548049e+00],
[-5.00626383e+00,  5.13045095e+00],
[ 2.81996606e+00,  4.31736135e+00],
[ 1.34848673e+00,  5.15919571e+00],
[ 6.05622582e+00,  3.38608105e+00],
[ 1.69492447e+00,  3.29996883e+00],
[-3.80025218e+00,  1.05063262e+01],
[ 6.05625997e+00,  1.25681813e+01],
[-3.82692678e+00,  8.50372394e+00],
[-2.54631499e+00,  6.10558107e+00],
[ 5.22863663e+00, -1.45261196e+00],
[-4.27360635e+00, -8.43896330e-01],
```

```
[-5.98816972e+00, -7.23800299e+00],
[-6.61288829e+00, -5.24342777e+00],
[-5.37510939e+00, -7.43613939e+00],
[-6.86520702e+00, -6.75091296e+00],
[-2.26981819e+00,  8.19201591e+00],
[ 4.79995281e+00, -1.16999864e+00],
[-7.47825050e+00, -3.85847325e+00],
[ 4.76520140e+00,  1.81268728e-01],
[-6.40146716e+00,  7.85751149e+00],
[-6.77352206e+00,  9.20283431e+00],
[-2.60465499e+00,  5.80042152e+00],
[ 4.67563404e+00,  5.97038840e+00],
[-8.85534880e+00, -7.46708926e+00],
[-4.81704581e+00,  8.16395209e+00],
[ 5.32873417e+00,  2.92590226e+00],
[ 5.86038226e+00,  4.10341333e+00],
[-3.58749504e+00,  4.98962002e+00],
[-3.28956047e+00, -6.82234903e+00],
[-8.11730724e+00, -5.16727228e+00],
[-3.45166254e+00,  6.80802364e+00],
[-7.45196338e+00, -8.23586216e+00],
[-1.69486686e+00,  9.73218813e+00],
[ 4.02535618e+00,  3.93667104e+00],
[-5.45142428e+00, -2.66507758e+00],
[ 3.77288841e+00,  4.06033504e+00],
[ 7.83857915e+00,  2.00131060e+00],
[-1.12643033e+01, -8.52839091e+00],
[ 5.35455652e+00,  4.12318258e+00],
[ 6.02859385e+00,  4.35399647e+00],
[-1.56424733e+00,  4.16592570e+00],
[-4.98173121e+00, -7.98346589e+00],
[-3.96603818e+00,  1.04257716e+01],
[-1.27203404e+01, -8.32593590e+00],
[-2.88898227e+00, -4.24211482e+00],
[-7.25272166e+00,  7.46799542e+00],
[-3.77595424e+00,  1.19213723e+01],
[-2.82723040e+00,  8.18625097e+00],
[-9.58375433e-01, -8.99031539e+00],
[ 9.49487800e+00,  3.08686939e+00],
[-5.23317252e+00,  7.45696737e+00],
[-6.67116466e+00, -8.74230430e+00],
[-5.93979826e+00, -1.72063979e+00],
```

```
[ 3.41196272e+00,  4.32826637e+00],
[-1.59822319e+00,  1.16970352e+01],
[-2.72025275e-01,  5.62940926e+00],
[ 2.20927089e+00,  2.39591373e+00],
[ 3.26051064e-01,  1.15753065e+01],
[ 7.26338369e+00,  3.76449563e+00],
[-6.93710657e+00, -6.55745929e+00],
[ 1.52133649e+00,  8.39340130e+00],
[-6.81222421e+00, -5.51061429e+00],
[-9.18886226e+00, -8.52843937e+00],
[ 7.28916319e+00,  3.10831723e+00],
[ 4.81664889e+00, -9.90628455e-01],
[ 1.05357251e+01,  3.71644700e+00],
[-2.24223436e+00,  1.16780599e+01],
[ 8.12388450e+00,  2.70786535e-02],
[-7.78581847e+00,  8.94137297e+00],
[-5.29448320e+00,  9.87846629e+00],
[-1.05102685e+01, -1.84359799e+00],
[ 2.19299941e-01,  2.48091279e+00],
[ 4.42784914e+00,  2.91133761e+00],
[-6.16170926e+00,  9.55565453e+00],
[ 2.76981085e+00,  2.61186735e+00],
[-4.52106323e+00, -7.31994055e+00],
[-6.56745449e+00, -5.05925024e+00],
[ 3.42975650e+00,  2.33270627e+00],
[-5.55161880e+00,  5.72471791e+00],
[-6.93053832e+00, -7.67392085e+00],
[ 6.90054428e+00,  7.18935039e+00],
[-7.19461177e+00, -3.47611474e+00],
[ 2.51460950e+00,  1.32191852e+00],
[-3.69482229e+00, -4.70303718e+00],
[-9.64617499e+00, -1.02191283e+01],
[-1.80093405e+00,  8.80955986e+00],
[ 5.94128230e+00,  1.77289017e+00],
[ 7.82601668e+00, -2.83706692e-01],
[-2.74751611e-01,  1.27439462e+01],
[ 3.96506152e+00, -9.96047680e-02],
[ 5.62379408e+00,  3.51532713e+00],
[-7.17871760e+00, -5.77540236e+00],
[-3.11329531e+00,  9.99634570e+00],
[ 6.20982774e+00,  5.09597519e+00],
[ 5.44582814e+00,  8.70328437e-03],
```

```
[-4.66314418e+00,  8.12861696e+00],
[-6.10689955e+00, -8.59253327e+00],
[ 5.40077853e+00,  4.24792362e+00],
[-3.36604873e+00, -8.50693091e+00],
[-2.49513562e+00,  8.36917151e+00],
[ 2.67279364e+00,  3.84206349e+00],
[-5.89783460e+00, -8.78561098e+00],
[ 6.33565117e-01,  1.10821020e+01],
[ 8.60338038e+00, -1.97545123e+00],
[-2.34356455e+00,  1.57882019e+01],
[-4.16095402e+00,  8.21212832e+00],
[-2.95273333e+00,  1.01254260e+01],
[ 1.17244796e+00,  4.49729004e+00],
[-3.78359628e+00,  7.73352931e+00],
[ 2.46044681e+00,  1.65764447e+00],
[-6.61101792e+00, -9.00588650e+00],
[-5.16329769e+00, -5.15215944e+00],
[-4.60973223e+00, -4.64295809e+00],
[ 1.55501100e+00,  7.58904303e+00],
[ 6.07521814e+00,  2.78987754e+00],
[ 5.11612638e+00,  3.03279248e+00],
[ 2.31119611e+00, -2.19266018e+00],
[-2.60771923e+00,  1.33170562e+01],
[-1.51469855e+00,  7.24020680e+00],
[ 2.63137060e+00,  2.56843081e+00],
[-9.35026754e+00, -5.52733187e+00],
[-8.96724201e+00, -6.46652668e+00],
[-1.84380138e+00,  3.75276546e+00],
[-6.02463139e+00, -2.82288000e+00],
[ 2.67781361e+00,  9.49437511e+00],
[ 1.83363762e+00,  1.11247316e+01],
[ 3.34206621e+00,  4.96778383e+00],
[ 1.95950424e+00,  4.13765234e+00]])
```

In [98]:
```
#true_labels
```

Out[98]:
```
array([1, 0, 2, 2, 2, 2, 0, 2, 1, 2, 1, 1, 1, 1, 1, 2, 0, 2, 1, 2, 1, 1,
       2, 0, 2, 0, 0, 2, 0, 1, 1, 1, 2, 2, 0, 0, 2, 0, 2, 0, 1, 0, 2, 1,
       2, 1, 1, 2, 1, 0, 2, 0, 2, 0, 0, 0, 2, 1, 0, 1, 2, 0, 2, 2, 2, 2,
       0, 0, 2, 0, 0, 2, 0, 1, 0, 1, 1, 0, 1, 0, 0, 1, 1, 2, 2, 2, 2, 0,
       1, 2, 1, 0, 0, 0, 1, 2, 0, 1, 1, 0, 2, 2, 0, 2, 0, 1, 2, 1, 1, 2,
       1, 1, 0, 2, 0, 2, 2, 0, 0, 0, 2, 1, 0, 2, 2, 1, 0, 0, 1, 0, 1, 2,
       0, 2, 2, 1, 1, 1, 0, 1, 0, 0, 2, 1, 1, 0, 1, 2, 2, 1, 0, 2, 1, 2,
       1, 2, 2, 0, 1, 1, 0, 1, 1, 2, 0, 1, 1, 0, 2, 1, 2, 0, 1, 2, 0, 1,
       0, 0, 0, 1, 0, 1, 2, 2, 2, 0, 1, 1, 1, 0, 0, 1, 2, 2, 0, 2, 0, 0,
       1, 1])
```

In [105]:
```
scaler = StandardScaler()
scaled_df_vols = scaler.fit_transform(df_vols)
```

In [106]:
```
scaled_df_vols
```

```
Out[106]: array([[-1.50613218,  0.68949749,  0.85513662,  0.57728312,  0.76486616,
                  2.83419411, -0.76502995, -0.97524604, -0.78676507, -0.44913737,
                 -0.95481654, -0.76660618, -1.125     ,  1.2579418 , -0.64061319,
                  0.        ,  0.        ,  0.        , -0.13150595],
                [-1.3026008 ,  1.12986253,  1.17987205,  0.89914219,  0.83235435,
                 -0.27330705,  1.77378626,  1.64726432,  1.84151126,  0.5736507 ,
                  0.3365337 ,  1.67311062, -1.125     , -0.31448545,  0.09151617,
                  0.        ,  0.        ,  0.        ,  0.50481316],
                [-0.79377236, -0.06483152, -0.20566577,  0.10687372, -0.13497638,
                  0.47737631, -0.24034127, -0.237665  , -0.27141677,  1.59643877,
                  1.39309299, -0.05691161,  0.75      ,  0.20965697, -0.09151617,
                  0.        ,  0.        ,  0.        ,  1.56534502],
                [-0.28494393, -0.8477027 , -0.57369925, -0.73903794, -0.58489765,
                 -0.12207715,  1.57068097,  1.48335743,  1.54948056, -1.29404925,
                 -1.22873932,  1.44183635,  1.375     ,  0.20965697, -0.54909702,
                  0.        ,  0.        ,  0.        , -0.97993143],
                [-0.08141255, -1.37369428, -1.30976622, -1.20944734, -1.10230711,
                 -0.73337392, -0.20649039, -0.237665  , -0.21988194, -1.24958021,
                 -1.1896075 , -0.03423766, -0.5       ,  0.73379939, -0.82364553,
                  0.        ,  0.        ,  0.        , -0.81024634],
                [ 0.02035314, -0.99449105, -1.00667982, -1.26721692, -1.43974807,
                 -0.68782274, -1.34049496, -1.30305984, -1.26775682, -0.31573023,
                 -0.28957551, -1.43548765, -1.75      , -1.36277029,  0.27454851,
                  0.        ,  0.        ,  0.        , -0.42845487],
                [ 0.22388451, -0.01182462,  0.29226188, -0.26862853, -0.08998425,
                 -0.16945037, -1.32356952, -1.30305984, -1.30211337, -1.11617307,
                 -1.03308019, -1.43548765,  0.125     , -1.36277029, -0.18303234,
                  0.        ,  0.        ,  0.        , -0.97993143],
                [ 0.73271295, -0.97002633, -1.30976622, -0.66888916, -0.65238584,
                 -0.56574559,  0.33512374,  0.74577639,  0.34700119,  0.12896023,
                  1.11917021,  0.54394801,  0.75      , -0.83862787,  2.83700127,
                  0.        ,  0.        ,  0.        , -1.19203781],
                [ 1.1397757 ,  1.82302972,  1.6777997 ,  2.12468248,  2.13712604,
                 -0.4072275 ,  0.19972021,  0.25405569,  0.1923967 ,  0.88493402,
                  0.92351108,  0.26959324,  0.75      , -0.31448545, -0.36606468,
                  0.        ,  0.        ,  0.        ,  1.01386845],
                [ 1.85213551,  0.62018077,  0.40050703,  0.44523838,  0.26995276,
                 -0.35256609, -0.00338509, -0.0737581 , -0.08245573,  1.2406864 ,
                  0.92351108, -0.19975748,  0.75      ,  1.78208422, -0.54909702,
                  0.        ,  0.        ,  0.        ,  1.4380812 ]])
```

In [107]:
```python
kmeans = KMeans(init="random", n_clusters=3, n_init=10, max_iter=300, random_state=42)
```

In [108]:
```python
kmeans.fit(scaled_df_vols)
```

Out[108]:
```
KMeans(init='random', n_clusters=3, random_state=42)
```

In [109]:
```python
print("Valor SSE més baix:", kmeans.inertia_)
```

```
Valor SSE més baix: 83.95196805310437
```

In [110]:
```python
kmeans.cluster_centers_
```

Out[110]:
```
array([[-0.33582677, -0.42262812, -0.29226188, -0.54200242, -0.46679332,
         0.31088677, -0.90889621, -0.95475768, -0.8941293 , -0.78265522,
        -0.86676993, -0.91795478, -0.8125    , -0.18344985, -0.34318564,
         0.        ,  0.        ,  0.        , -0.58753465],
       [-0.28494393, -0.22928883, -0.23453114, -0.16959497, -0.13497638,
        -0.3203766 ,  1.22653032,  1.29213271,  1.24599767, -0.19714611,
         0.07565486,  1.21963166,  0.33333333, -0.31448545,  0.79314014,
         0.        ,  0.        ,  0.        , -0.55571869],
       [ 0.73271295,  0.79279299,  0.62421365,  0.89226486,  0.75736747,
        -0.0941391 , -0.01466872, -0.01912247, -0.05382527,  1.2406864 ,
         1.08003838,  0.00430805,  0.75      ,  0.55908525, -0.33555929,
         0.        ,  0.        ,  0.        ,  1.33909822]])
```

In [111]:
```python
print("Iteracions: ", kmeans.n_iter_) #Nombre d'iteracions necessaries per la convergencia.
```

```
Iteracions:  2
```

In [112]:
```python
kmeans.labels_
```

Out[112]:
```
array([0, 1, 2, 1, 0, 0, 0, 1, 2, 2])
```

In [113]:
```python
kmeans_kwargs = {"init": "random", "n_init": 10, "max_iter": 300, "random_state": 42, }
```

In [114]:

```
sse = []
for k in range(1, 11):
    kmeans = KMeans(n_clusters=k, **kmeans_kwargs)
    kmeans.fit(scaled_df_vols)
    sse.append(kmeans.inertia_)
```

```
C:\Users\Xavier\anaconda3\lib\site-packages\sklearn\cluster\_kmeans.py:1039: UserWarning: KMeans is known to have a m
emory leak on Windows with MKL, when there are less chunks than available threads. You can avoid it by setting the en
vironment variable OMP_NUM_THREADS=1.
  warnings.warn(
```

In [125]:

```
scaled_df_vols
```

```
Out[125]: array([[-1.50613218,  0.68949749,  0.85513662,  0.57728312,  0.76486616,
                  2.83419411, -0.76502995, -0.97524604, -0.78676507, -0.44913737,
                 -0.95481654, -0.76660618, -1.125     ,  1.2579418 , -0.64061319,
                  0.        ,  0.        ,  0.        , -0.13150595],
                [-1.3026008 ,  1.12986253,  1.17987205,  0.89914219,  0.83235435,
                 -0.27330705,  1.77378626,  1.64726432,  1.84151126,  0.5736507 ,
                  0.3365337 ,  1.67311062, -1.125     , -0.31448545,  0.09151617,
                  0.        ,  0.        ,  0.        ,  0.50481316],
                [-0.79377236, -0.06483152, -0.20566577,  0.10687372, -0.13497638,
                  0.47737631, -0.24034127, -0.237665  , -0.27141677,  1.59643877,
                  1.39309299, -0.05691161,  0.75      ,  0.20965697, -0.09151617,
                  0.        ,  0.        ,  0.        ,  1.56534502],
                [-0.28494393, -0.8477027 , -0.57369925, -0.73903794, -0.58489765,
                 -0.12207715,  1.57068097,  1.48335743,  1.54948056, -1.29404925,
                 -1.22873932,  1.44183635,  1.375     ,  0.20965697, -0.54909702,
                  0.        ,  0.        ,  0.        , -0.97993143],
                [-0.08141255, -1.37369428, -1.30976622, -1.20944734, -1.10230711,
                 -0.73337392, -0.20649039, -0.237665  , -0.21988194, -1.24958021,
                 -1.1896075 , -0.03423766, -0.5       ,  0.73379939, -0.82364553,
                  0.        ,  0.        ,  0.        , -0.81024634],
                [ 0.02035314, -0.99449105, -1.00667982, -1.26721692, -1.43974807,
                 -0.68782274, -1.34049496, -1.30305984, -1.26775682, -0.31573023,
                 -0.28957551, -1.43548765, -1.75      , -1.36277029,  0.27454851,
                  0.        ,  0.        ,  0.        , -0.42845487],
                [ 0.22388451, -0.01182462,  0.29226188, -0.26862853, -0.08998425,
                 -0.16945037, -1.32356952, -1.30305984, -1.30211337, -1.11617307,
                 -1.03308019, -1.43548765,  0.125     , -1.36277029, -0.18303234,
                  0.        ,  0.        ,  0.        , -0.97993143],
                [ 0.73271295, -0.97002633, -1.30976622, -0.66888916, -0.65238584,
                 -0.56574559,  0.33512374,  0.74577639,  0.34700119,  0.12896023,
                  1.11917021,  0.54394801,  0.75      , -0.83862787,  2.83700127,
                  0.        ,  0.        ,  0.        , -1.19203781],
                [ 1.1397757 ,  1.82302972,  1.6777997 ,  2.12468248,  2.13712604,
                 -0.4072275 ,  0.19972021,  0.25405569,  0.1923967 ,  0.88493402,
                  0.92351108,  0.26959324,  0.75      , -0.31448545, -0.36606468,
                  0.        ,  0.        ,  0.        ,  1.01386845],
                [ 1.85213551,  0.62018077,  0.40050703,  0.44523838,  0.26995276,
                 -0.35256609, -0.00338509, -0.0737581 , -0.08245573,  1.2406864 ,
                  0.92351108, -0.19975748,  0.75      ,  1.78208422, -0.54909702,
                  0.        ,  0.        ,  0.        ,  1.4380812 ]])
```

In [129]:
```
x=scaled_df_vols[0]
y=scaled_df_vols[1]
kl = KneeLocator(x,y, S=1.0, curve="concave", direction="increasing")
```

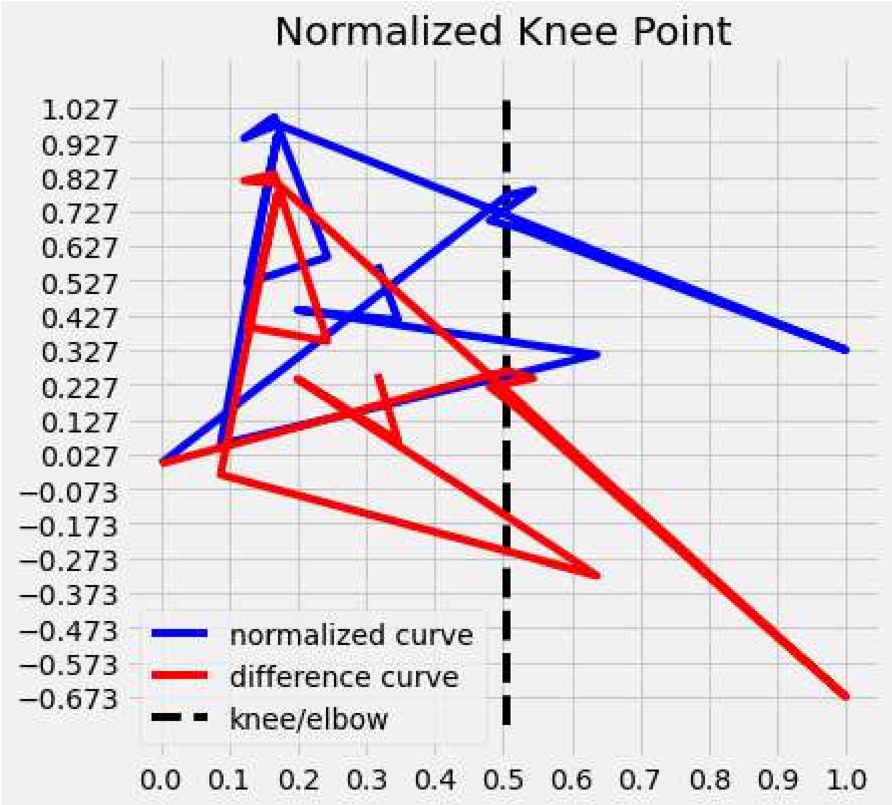
In [134]:
```
print("Valor knee:", round(kl.knee, 3))
```

```
Valor knee: 0.689
```

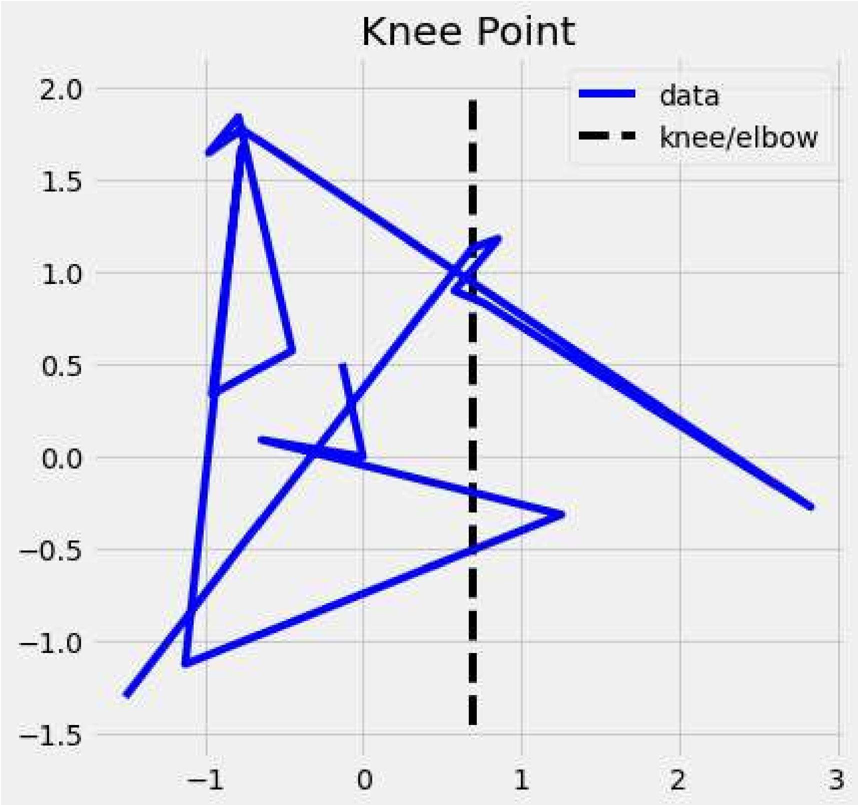
In [135]:
```
print("Valor elbow:", round(kl.elbow, 3))
```

```
Valor elbow: 0.689
```

In [132]:
```
kl.plot_knee_normalized()
```



In [133]:
```
kl.plot_knee()
```



In [ ]:
```
#Nivell 2
```

In [ ]:
```
#Exercici 2
#Classifica els diferents vols utilitzant l'algorisme de clustering jeràrquic.
```

In [ ]:
```
#Aglomerative Hierachical clustering
```

In [31]:
```
df_vols #dataframe de 10 instàncies de les dades dels vols.
```

Out[31]:

	Unnamed: 0	DepTime	CRSDepTime	ArrTime	CRSArrTime	FlightNum	ActualElapsedTime	CRSElapsedTime	AirTime	ArrDelay	DepDelay	Dis
3	4	1829.0	1755	1959.0	1925	3920	90.0	90.0	77.0	34.0	34.0	
5	6	1937.0	1830	2037.0	1940	509	240.0	250.0	230.0	57.0	67.0	
7	11	1644.0	1510	1845.0	1725	1333	121.0	135.0	107.0	80.0	94.0	
9	16	1452.0	1425	1640.0	1625	675	228.0	240.0	213.0	15.0	27.0	
11	18	1323.0	1255	1526.0	1510	4	123.0	135.0	110.0	16.0	28.0	
12	19	1416.0	1325	1512.0	1435	54	56.0	70.0	49.0	37.0	51.0	
13	21	1657.0	1625	1754.0	1735	623	57.0	70.0	47.0	19.0	32.0	
17	26	1422.0	1255	1657.0	1610	188	155.0	195.0	143.0	47.0	87.0	
19	30	2107.0	1945	2334.0	2230	362	147.0	165.0	134.0	64.0	82.0	
23	37	1812.0	1650	1927.0	1815	422	135.0	145.0	118.0	72.0	82.0	

In [38]:
```
#Llibreries import
import scipy.cluster.hierarchy as shc #Hierachical clustering
```

In [ ]:
```
#Llibreries from
from sklearn.preprocessing import normalize
from sklearn.cluster import AgglomerativeClustering #Aglomerative Hierachical clustering
```

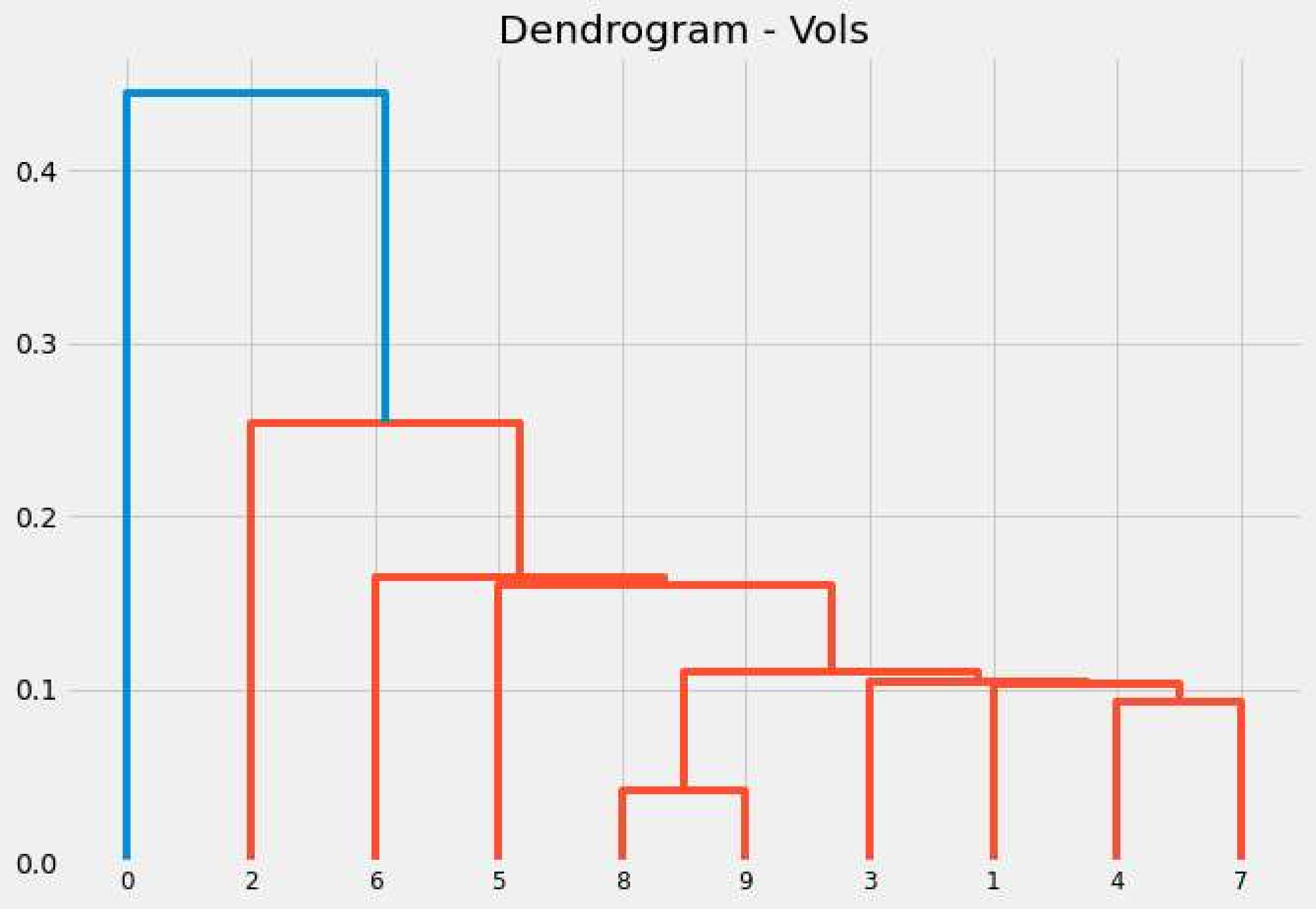
In [34]:

```
#Normalització dels vols
df_vols_scaled = normalize(df_vols)
df_vols_scaled = pd.DataFrame(df_vols_scaled, columns=df_vols.columns)
df_vols_scaled.head()
```

Out[34]:

	Unnamed: 0	DepTime	CRSDepTime	ArrTime	CRSArrTime	FlightNum	ActualElapsedTime	CRSElapsedTime	AirTime	ArrDelay	DepDelay	Dis
0	0.000735	0.336031	0.322435	0.359915	0.353668	0.720197	0.016535	0.016535	0.014147	0.006247	0.006247	0.0
1	0.001415	0.456716	0.431487	0.480294	0.457423	0.120015	0.056588	0.058946	0.054231	0.013440	0.015798	0.3
2	0.002951	0.441114	0.405160	0.495046	0.462848	0.357667	0.032466	0.036223	0.028710	0.021465	0.025222	0.2
3	0.004562	0.414041	0.406342	0.467650	0.463373	0.192478	0.065015	0.068437	0.060737	0.004277	0.007699	0.4
4	0.006108	0.448963	0.425887	0.517852	0.512422	0.001357	0.041740	0.045813	0.037329	0.005430	0.009502	0.2

In [54]:
```
plt.figure(figsize=(10, 7))
plt.title("Dendrogram - Vols") #Generem un dendrograma dels vols
vols = shc.dendrogram(shc.linkage(df_vols_scaled, method='single'))
```



In [51]:
```
cluster = AgglomerativeClustering(n_clusters=2, affinity='euclidean', linkage='ward')
cluster.fit_predict(df_vols_scaled)
```

Out[51]:
```
array([1, 0, 0, 0, 0, 0, 0, 0, 0, 0], dtype=int64)
```

In [ ]:
```
#Nivell 3
```

In [ ]:

```
#Exercici 3
#Calcula el rendiment del clustering mitjançant un paràmetre com pot ser silhouette.
#El parametre silhouette té un interval [-1,1]
```

In [55]:

```
#Llibreries import
import pandas as pd
import numpy as np
import seaborn as sns
```

In [56]:

```
#Llibreries from
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score
%matplotlib inline
```

In [66]:

```
X= df_vols_scaled['ArrDelay']
Y= df_vols_scaled['Distance']
Z= np.concatenate((X,Y))
Z=pd.DataFrame(Z)
```

In [68]:

```
KMean= KMeans(n_clusters=2)
KMean.fit(Z)
label=KMean.predict(Z)
```

In [69]:

```
print(f'Silhouette Score(n=2): {silhouette_score(Z, label)}')
```

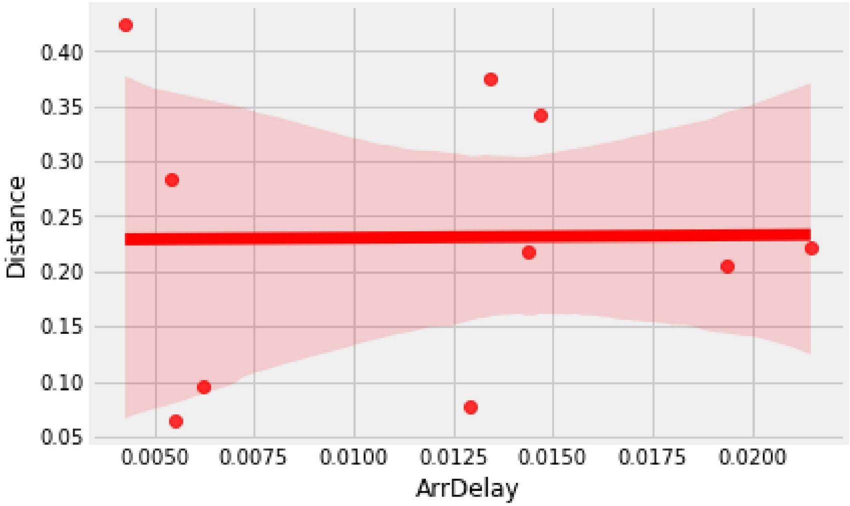
```
Silhouette Score(n=2): 0.7766823273238923
```

In [ ]:

```
#El gran de separació dels clusters és més gran si el parametre silhouette s'apropa a 1.
#En el cas de l'exercici els clusters es troben més separats que junts.
```

In [88]:
```
sns.regplot(x=X, y=Y, scatter=True, color="r")
```

Out[88]: <matplotlib.axes._subplots.AxesSubplot at 0x2a2a123a4f0>



In [ ]: