Marketing

Question 1

In [69]:
```
import pandas as pd
import numpy as np
import sklearn
from sklearn.cluster import KMeans
from sklearn.preprocessing import MinMaxScaler, StandardScaler, OneHotEncoder
from sklearn.decomposition import PCA
from sklearn.metrics import silhouette_score
import matplotlib.pyplot as plt
```

load data

df = Variables de segmentation

Variables d

df2 = Descripteurs

Descripteu

In [141]:
```
df = pd.read_csv("/Users/vincent/Desktop/school_2020/Book1.csv")
df2 = pd.read_csv("/Users/vincent/Desktop/school_2020/Book2.csv")
```

In [142]:
```
df.head()
```

	Id	Networking	Reputation of the school	Changing company	Academic knowledge	Cost-driven	Location	Ranking of the program	Increasing your salary	Launching your own company	Personal dev	Changing career
0	1	5	5	1	2	2	2	3	1	5	5	5
1	2	5	5	1	2	2	2	3	1	5	5	5
2	3	1	5	3	4	4	4	2	1	1	5	1
3	4	5	5	1	5	1	2	5	5	1	5	1
4	5	4	5	5	4	1	1	2	5	1	5	5

In [143]:
```
df2.head()
```

	Id	Gender	Country	Age	Experience	Financed
0	1	Male	International	44	13	Self
1	2	Male	France	37	11	Self
2	3	Male	International	43	15	Employer
3	4	Female	International	34	10	Self
4	5	Male	International	40	18	Self

In [144]:
```
#dropping this garbage column that starts counting at 1 ??? who counts from 1?!
df = df.drop('Id', axis=1)
df2 = df2.drop('Id', axis=1)
```

scale and PCA

we have to scale our data so that its normalized into a space that is legible by a computer

In [145]:
```
ss = StandardScaler()
```

In [146]:
```
df_scaled = ss.fit_transform(df)
df_scaled
```

```
array([[ 1.11433685,  0.81228226, -1.03397878, ...,  1.34420385,
        -0.58461089,  0.83360904],
       [ 1.11433685,  0.81228226, -1.03397878, ...,  1.34420385,
        -0.58461089,  0.83360904],
       [-2.09039832,  0.81228226,  0.16775467, ...,  1.34420385,
        -0.58461089,  0.83360904],
       ...,
       [ 0.31315306, -0.18162833,  0.76862139, ...,  0.65994504,
         1.79818245,  0.05160437],
       [-2.09039832, -3.16336012, -1.03397878, ...,  1.34420385,
        -0.58461089,  0.83360904],
       [ 0.31315306, -0.18162833,  0.76862139, ...,  0.65994504,
         1.79818245,  0.05160437]])
```

df2 had strings in it, which cant be read by a comp, so lets turn those into binaryyyy

In [147]:
```
cat_cols = ['Gender', 'Country', 'Financed']
#enc = OneHotEncoder(cat_cols, handle_unknown='ignore')
```

In [148]:
```
df2['bGender'] = np.where(df2['Gender'].str.contains('Male'), 1, 0)
df2['bCountry'] = np.where(df2['Country'].str.contains('France'), 1, 0)
df2['bFinanced'] = np.where(df2['Financed'].str.contains('Self'), 1, 0)
df2 = df2.drop(cat_cols, axis=1)
df2.head()
```

	Age	Experience	bGender	bCountry	bFinanced
0	44	13	1	0	1
1	37	11	1	1	1
2	43	15	1	0	0
3	34	10	0	0	1

	Age	Experience	bGender	bCountry	bFinanced
4	40	18	1	0	1

In [149]:

```
df2_scaled = ss.fit_transform(df2)
df2_scaled
```

```
array([[ 1.20115105, -0.40742227,  0.75894664, -1.22215829,  0.44857498],
       [-0.48731714, -0.96674065,  0.75894664,  0.81822462,  0.44857498],
       [ 0.95994131,  0.15189611,  0.75894664, -1.22215829, -2.22928172],
       [-1.21094637, -1.24639984, -1.31761569, -1.22215829,  0.44857498],
       [ 0.23631208,  0.99087367,  0.75894664, -1.22215829,  0.44857498],
       [ 1.20115105,  0.71121449,  0.75894664, -1.22215829,  0.44857498],
       [ 2.40719976,  0.15189611,  0.75894664, -1.22215829,  0.44857498],
       [-0.2461074 , -0.40742227,  0.75894664, -1.22215829,  0.44857498],
       [-0.96973663,  0.4315553 , -1.31761569,  0.81822462,  0.44857498],
       [-1.45215611,  0.4315553 ,  0.75894664,  0.81822462,  0.44857498],
       [ 0.47752183,  0.4315553 ,  0.75894664,  0.81822462,  0.44857498],
       [-0.48731714,  1.27053286,  0.75894664,  0.81822462, -2.22928172],
       [ 0.23631208, -0.12776308,  0.75894664,  0.81822462, -2.22928172],
       [ 0.47752183,  1.55019205,  0.75894664,  0.81822462,  0.44857498],
       [-0.72852689, -1.24639984,  0.75894664,  0.81822462,  0.44857498],
       [ 0.71873157, -0.96674065, -1.31761569, -1.22215829,  0.44857498],
       [ 0.71873157, -0.68708146,  0.75894664,  0.81822462,  0.44857498],
       [ 0.47752183, -0.12776308, -1.31761569, -1.22215829,  0.44857498],
       [ 1.20115105,  1.27053286,  0.75894664,  0.81822462,  0.44857498],
       [ 0.95994131,  1.27053286,  0.75894664,  0.81822462,  0.44857498],
       [-0.2461074 , -0.96674065, -1.31761569, -1.22215829,  0.44857498],
       [ 1.44236079,  0.71121449,  0.75894664,  0.81822462, -2.22928172],
       [ 1.20115105, -0.40742227, -1.31761569,  0.81822462,  0.44857498],
       [-0.00489766,  0.4315553 ,  0.75894664, -1.22215829,  0.44857498],
       [-1.9345756 , -2.0853774 , -1.31761569,  0.81822462,  0.44857498],
       [-0.48731714, -1.24639984,  0.75894664, -1.22215829,  0.44857498],
       [ 1.68357054, -2.0853774 ,  0.75894664, -1.22215829,  0.44857498],
       [-0.2461074 ,  1.55019205, -1.31761569, -1.22215829,  0.44857498],
       [ 0.71873157, -0.12776308, -1.31761569,  0.81822462,  0.44857498],
       [-0.2461074 , -0.96674065,  0.75894664,  0.81822462,  0.44857498],
       [-0.00489766,  0.99087367, -1.31761569, -1.22215829,  0.44857498],
```

```
[-1.9345756 , -0.40742227, -1.31761569, -1.22215829,  0.44857498],
[-0.72852689, -0.40742227,  0.75894664, -1.22215829,  0.44857498],
[-0.00489766, -0.12776308, -1.31761569,  0.81822462, -2.22928172],
[-1.9345756 , -0.40742227, -1.31761569, -1.22215829, -2.22928172],
[-0.00489766, -0.68708146,  0.75894664,  0.81822462,  0.44857498],
[-0.2461074 ,  2.10951043,  0.75894664,  0.81822462,  0.44857498],
[-0.96973663, -0.12776308, -1.31761569,  0.81822462, -2.22928172],
[ 0.71873157,  2.10951043,  0.75894664,  0.81822462,  0.44857498],
[ 0.95994131, -0.12776308, -1.31761569,  0.81822462,  0.44857498],
[-0.72852689,  1.55019205, -1.31761569,  0.81822462,  0.44857498],
[-0.00489766,  2.38916962,  0.75894664,  0.81822462,  0.44857498],
[-0.2461074 ,  2.38916962,  0.75894664,  0.81822462,  0.44857498],
[-0.96973663, -0.68708146,  0.75894664,  0.81822462,  0.44857498],
[ 0.23631208, -0.96674065,  0.75894664,  0.81822462,  0.44857498],
[ 0.47752183,  0.15189611, -1.31761569, -1.22215829,  0.44857498],
[ 1.68357054,  0.71121449, -1.31761569,  0.81822462, -2.22928172],
[ 0.95994131, -0.96674065,  0.75894664, -1.22215829,  0.44857498],
[ 0.71873157,  0.4315553 ,  0.75894664,  0.81822462,  0.44857498],
[-1.9345756 , -0.40742227,  0.75894664, -1.22215829,  0.44857498],
[-0.72852689, -0.40742227,  0.75894664,  0.81822462,  0.44857498],
[-0.96973663, -0.40742227, -1.31761569,  0.81822462, -2.22928172],
[ 0.71873157,  1.27053286,  0.75894664, -1.22215829,  0.44857498],
[ 2.16599002, -0.40742227,  0.75894664,  0.81822462,  0.44857498],
[ 0.71873157, -0.68708146,  0.75894664,  0.81822462,  0.44857498],
[-0.00489766, -0.40742227,  0.75894664, -1.22215829,  0.44857498],
[-0.48731714, -0.12776308,  0.75894664, -1.22215829,  0.44857498],
[ 0.47752183, -0.12776308,  0.75894664, -1.22215829,  0.44857498],
[ 1.20115105, -0.40742227,  0.75894664,  0.81822462,  0.44857498],
[ 0.47752183,  0.4315553 ,  0.75894664, -1.22215829,  0.44857498],
[-0.96973663, -0.40742227,  0.75894664,  0.81822462,  0.44857498],
[-0.72852689,  1.55019205,  0.75894664,  0.81822462,  0.44857498],
[ 0.47752183,  0.15189611, -1.31761569,  0.81822462,  0.44857498],
[-0.96973663, -0.40742227,  0.75894664, -1.22215829,  0.44857498],
[-0.2461074 ,  0.15189611,  0.75894664,  0.81822462,  0.44857498],
[-0.96973663, -0.68708146, -1.31761569, -1.22215829,  0.44857498],
[-0.2461074 , -0.40742227,  0.75894664, -1.22215829, -2.22928172],
[ 1.20115105, -0.12776308,  0.75894664,  0.81822462, -2.22928172],
[-0.72852689, -0.12776308,  0.75894664,  0.81822462,  0.44857498],
[ 0.71873157,  0.99087367,  0.75894664,  0.81822462,  0.44857498],
```

```
[-0.48731714, -0.68708146,  0.75894664,  0.81822462, -2.22928172],
[ 0.95994131, -0.96674065,  0.75894664, -1.22215829, -2.22928172],
[-0.96973663, -0.40742227,  0.75894664,  0.81822462, -2.22928172],
[-1.45215611,  0.4315553 ,  0.75894664, -1.22215829,  0.44857498],
[-1.21094637,  2.38916962, -1.31761569,  0.81822462,  0.44857498],
[-1.21094637, -0.12776308,  0.75894664, -1.22215829,  0.44857498],
[ 0.95994131,  2.948488  ,  0.75894664, -1.22215829,  0.44857498],
[-3.14062431,  0.71121449,  0.75894664,  0.81822462,  0.44857498],
[-0.00489766,  2.10951043, -1.31761569,  0.81822462, -2.22928172],
[-0.2461074 ,  0.99087367,  0.75894664,  0.81822462,  0.44857498],
[ 0.47752183, -0.12776308,  0.75894664, -1.22215829,  0.44857498],
[ 0.95994131, -0.68708146, -1.31761569,  0.81822462,  0.44857498],
[-0.2461074 ,  0.4315553 ,  0.75894664, -1.22215829,  0.44857498],
[ 1.20115105, -0.12776308, -1.31761569,  0.81822462, -2.22928172],
[-1.69336586, -0.68708146,  0.75894664,  0.81822462,  0.44857498],
[ 0.23631208,  1.55019205, -1.31761569, -1.22215829,  0.44857498],
[ 0.71873157, -0.12776308, -1.31761569, -1.22215829,  0.44857498],
[ 0.47752183, -0.96674065,  0.75894664,  0.81822462, -2.22928172],
[-0.48731714,  1.55019205,  0.75894664, -1.22215829,  0.44857498],
[-0.96973663,  0.15189611, -1.31761569, -1.22215829,  0.44857498],
[ 0.23631208,  0.15189611, -1.31761569,  0.81822462,  0.44857498],
[ 0.71873157, -1.24639984,  0.75894664, -1.22215829,  0.44857498],
[-0.2461074 , -0.40742227, -1.31761569,  0.81822462, -2.22928172],
[ 0.47752183,  0.71121449, -1.31761569,  0.81822462,  0.44857498],
[-1.9345756 ,  0.15189611,  0.75894664, -1.22215829, -2.22928172],
[ 1.20115105,  0.99087367, -1.31761569,  0.81822462,  0.44857498],
[-0.00489766, -0.40742227, -1.31761569,  0.81822462,  0.44857498],
[-0.96973663,  0.4315553 ,  0.75894664, -1.22215829,  0.44857498],
[ 1.20115105,  0.99087367, -1.31761569, -1.22215829,  0.44857498],
[-0.00489766,  0.99087367,  0.75894664,  0.81822462, -2.22928172],
[ 0.95994131, -0.68708146, -1.31761569, -1.22215829,  0.44857498],
[-1.21094637, -0.40742227,  0.75894664,  0.81822462,  0.44857498],
[ 0.47752183, -1.24639984, -1.31761569,  0.81822462,  0.44857498],
[ 1.20115105,  1.82985124, -1.31761569,  0.81822462, -2.22928172],
[-0.00489766, -0.68708146,  0.75894664, -1.22215829, -2.22928172],
[ 0.23631208,  0.71121449, -1.31761569,  0.81822462,  0.44857498],
[ 0.23631208, -0.96674065,  0.75894664,  0.81822462,  0.44857498],
[-1.69336586, -0.40742227, -1.31761569, -1.22215829,  0.44857498],
[-0.48731714,  0.15189611, -1.31761569, -1.22215829,  0.44857498],
```

```
       [-0.72852689, -0.40742227,  0.75894664,  0.81822462,  0.44857498],
       [-1.9345756 , -0.68708146,  0.75894664,  0.81822462,  0.44857498],
       [-0.00489766, -0.40742227, -1.31761569, -1.22215829,  0.44857498],
       [-0.48731714, -0.40742227,  0.75894664,  0.81822462,  0.44857498],
       [-1.21094637,  0.15189611,  0.75894664, -1.22215829,  0.44857498],
       [-0.96973663, -0.12776308,  0.75894664, -1.22215829,  0.44857498],
       [-0.00489766, -1.24639984, -1.31761569, -1.22215829,  0.44857498],
       [ 0.23631208, -0.96674065,  0.75894664, -1.22215829,  0.44857498],
       [ 0.95994131, -0.96674065,  0.75894664,  0.81822462,  0.44857498],
       [ 0.23631208,  0.15189611,  0.75894664,  0.81822462,  0.44857498],
       [-0.48731714, -0.12776308,  0.75894664,  0.81822462,  0.44857498],
       [ 0.95994131, -1.24639984,  0.75894664,  0.81822462,  0.44857498],
       [-1.21094637, -0.12776308,  0.75894664,  0.81822462,  0.44857498],
       [-0.00489766, -0.96674065,  0.75894664, -1.22215829,  0.44857498],
       [-0.2461074 ,  1.27053286, -1.31761569,  0.81822462,  0.44857498],
       [-0.48731714, -0.96674065, -1.31761569,  0.81822462,  0.44857498],
       [-1.45215611, -0.68708146,  0.75894664,  0.81822462,  0.44857498],
       [-0.96973663, -0.40742227, -1.31761569,  0.81822462,  0.44857498],
       [ 1.44236079, -0.40742227, -1.31761569,  0.81822462,  0.44857498],
       [-0.96973663, -0.40742227,  0.75894664, -1.22215829,  0.44857498],
       [ 1.44236079, -0.68708146, -1.31761569,  0.81822462,  0.44857498],
       [-0.2461074 , -0.68708146, -1.31761569,  0.81822462, -2.22928172],
       [ 0.23631208,  0.15189611,  0.75894664, -1.22215829,  0.44857498],
       [ 0.71873157,  0.15189611,  0.75894664,  0.81822462,  0.44857498],
       [ 4.09566796,  0.4315553 ,  0.75894664, -1.22215829,  0.44857498],
       [-1.21094637, -0.12776308, -1.31761569, -1.22215829, -2.22928172],
       [-0.00489766, -2.92435497, -1.31761569, -1.22215829,  0.44857498],
       [ 0.47752183,  0.15189611, -1.31761569,  0.81822462,  0.44857498],
       [ 0.95994131, -0.96674065, -1.31761569, -1.22215829,  0.44857498],
       [-0.96973663, -0.12776308,  0.75894664, -1.22215829,  0.44857498],
       [-0.2461074 ,  0.71121449,  0.75894664,  0.81822462, -2.22928172],
       [-0.72852689, -0.12776308, -1.31761569,  0.81822462, -2.22928172],
       [-0.00489766, -0.40742227,  0.75894664, -1.22215829,  0.44857498],
       [ 0.23631208, -0.96674065, -1.31761569,  0.81822462,  0.44857498],
       [ 0.95994131,  1.27053286, -1.31761569, -1.22215829,  0.44857498],
       [-0.48731714, -0.12776308,  0.75894664,  0.81822462,  0.44857498],
       [-0.2461074 , -0.96674065,  0.75894664,  0.81822462, -2.22928172],
       [ 0.95994131,  3.78746557, -1.31761569,  0.81822462,  0.44857498],
       [ 0.23631208,  1.27053286,  0.75894664,  0.81822462,  0.44857498],
```

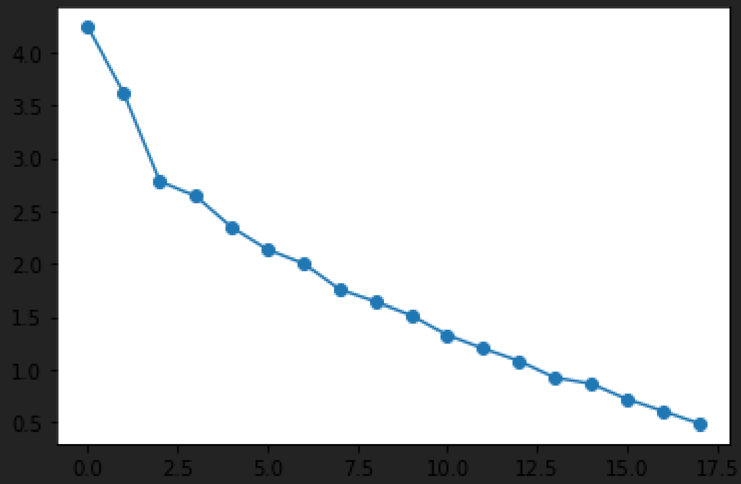
```
[-0.00489766, -0.12776308, -1.31761569,  0.81822462,  0.44857498],
[ 0.71873157,  0.4315553 , -1.31761569,  0.81822462,  0.44857498],
[ 0.23631208, -1.24639984,  0.75894664,  0.81822462,  0.44857498],
[ 1.20115105,  0.71121449, -1.31761569,  0.81822462,  0.44857498],
[-0.2461074 ,  1.82985124,  0.75894664,  0.81822462,  0.44857498],
[ 0.71873157, -0.68708146,  0.75894664,  0.81822462,  0.44857498],
[-0.2461074 , -0.12776308,  0.75894664,  0.81822462,  0.44857498],
[ 0.23631208,  1.27053286,  0.75894664,  0.81822462,  0.44857498],
[ 1.20115105,  0.4315553 ,  0.75894664,  0.81822462,  0.44857498],
[ 1.20115105, -1.24639984,  0.75894664,  0.81822462,  0.44857498],
[-0.00489766,  0.15189611, -1.31761569, -1.22215829,  0.44857498],
[-0.00489766,  0.71121449,  0.75894664,  0.81822462,  0.44857498],
[-0.96973663, -0.96674065, -1.31761569,  0.81822462, -2.22928172],
[ 0.47752183,  1.55019205,  0.75894664,  0.81822462,  0.44857498],
[ 1.92478028, -0.96674065, -1.31761569,  0.81822462,  0.44857498],
[-0.2461074 ,  0.15189611,  0.75894664, -1.22215829,  0.44857498],
[ 1.68357054,  0.15189611,  0.75894664,  0.81822462, -2.22928172],
[-1.45215611,  0.99087367, -1.31761569, -1.22215829,  0.44857498],
[-1.21094637,  0.4315553 ,  0.75894664, -1.22215829,  0.44857498],
[-0.48731714, -0.40742227,  0.75894664, -1.22215829,  0.44857498],
[-1.21094637,  0.71121449,  0.75894664,  0.81822462,  0.44857498],
[-0.72852689, -0.40742227, -1.31761569, -1.22215829, -2.22928172],
[ 0.71873157, -0.96674065, -1.31761569,  0.81822462,  0.44857498],
[-1.45215611,  0.4315553 ,  0.75894664,  0.81822462,  0.44857498],
[-0.00489766, -2.36503659,  0.75894664, -1.22215829, -2.22928172],
[-0.00489766, -2.0853774 ,  0.75894664, -1.22215829,  0.44857498],
[ 0.47752183, -1.52605903,  0.75894664,  0.81822462,  0.44857498],
[-1.45215611, -0.12776308,  0.75894664, -1.22215829, -2.22928172],
[-0.00489766,  0.99087367, -1.31761569,  0.81822462,  0.44857498],
[ 0.95994131,  0.15189611,  0.75894664, -1.22215829,  0.44857498],
[-1.21094637, -0.68708146,  0.75894664,  0.81822462,  0.44857498],
[-1.45215611, -0.96674065, -1.31761569,  0.81822462,  0.44857498],
[ 1.92478028,  0.4315553 , -1.31761569,  0.81822462,  0.44857498],
[-0.48731714,  1.82985124, -1.31761569,  0.81822462,  0.44857498],
[ 0.95994131,  0.71121449,  0.75894664, -1.22215829,  0.44857498],
[-0.00489766,  0.99087367,  0.75894664, -1.22215829,  0.44857498],
[-0.48731714, -0.68708146,  0.75894664, -1.22215829,  0.44857498],
[-1.69336586, -0.40742227, -1.31761569, -1.22215829,  0.44857498],
[-0.00489766, -0.40742227,  0.75894664,  0.81822462,  0.44857498],
```

```
       [ 0.95994131, -0.96674065,  0.75894664, -1.22215829,  0.44857498],
       [ 2.64840951, -0.12776308,  0.75894664, -1.22215829,  0.44857498],
       [ 0.71873157,  0.4315553 ,  0.75894664,  0.81822462,  0.44857498],
       [-0.48731714, -2.0853774 ,  0.75894664,  0.81822462,  0.44857498],
       [-0.48731714,  0.71121449, -1.31761569,  0.81822462,  0.44857498],
       [-0.2461074 ,  0.99087367,  0.75894664,  0.81822462,  0.44857498],
       [-0.00489766, -0.12776308, -1.31761569,  0.81822462,  0.44857498],
       [ 0.47752183,  0.15189611,  0.75894664,  0.81822462,  0.44857498],
       [-0.96973663, -0.12776308,  0.75894664, -1.22215829, -2.22928172],
       [-0.00489766, -1.52605903,  0.75894664,  0.81822462,  0.44857498]])
```

now applying PCA to reduce the dimensionality of our data

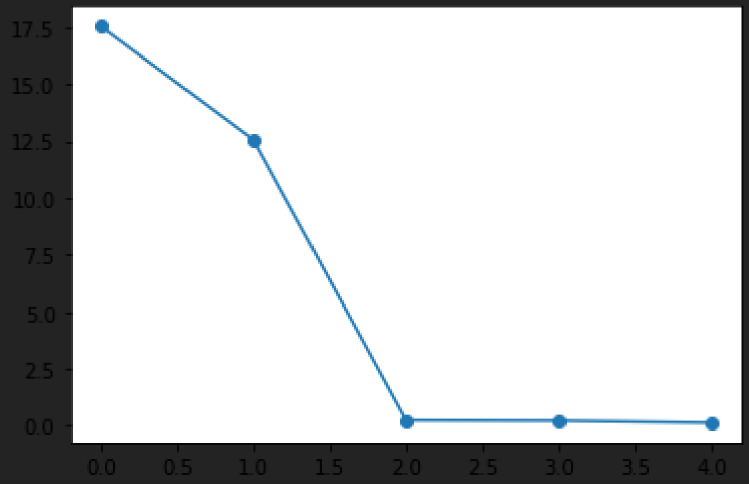
In [150]:
```
pca1 = PCA()
pca_data = pca1.fit_transform(df)
new_df = pd.DataFrame(pca_data)
new_df.columns = df.columns
plt.plot(pca1.explained_variance_, "-o")
```

```
[<matplotlib.lines.Line2D at 0x11e9f5c18>]
```



```
In [152]: pca2 = PCA()
          pca_data2 = pca2.fit_transform(df2)
          new_df2 = pd.DataFrame(pca_data2)
          new_df2.columns = df2.columns
          plt.plot(pca2.explained_variance_, "-o")
```

```
[<matplotlib.lines.Line2D at 0x11a97b4e0>]
```



export to csv so enginius can tell us what each principle component means bc im too tired to do the eigen math rn

```
In [153]: new_df.head()
```

	Networking	Reputation of the school	Changing company	Academic knowledge	Cost-driven	Location	Ranking of the program	Increasing your salary	Launching your own company	Personal dev	Changing career	C pr
0	-0.864621	3.246970	2.977510	2.885319	0.320777	-1.566546	-0.236069	-1.886132	-0.385450	1.147241	-0.256647	1.
1	-0.864621	3.246970	2.977510	2.885319	0.320777	-1.566546	-0.236069	-1.886132	-0.385450	1.147241	-0.256647	1.
2	-2.801328	2.951599	0.075888	-1.454817	-0.736540	-1.036595	-3.259385	1.839996	-1.176350	-1.134459	0.692407	0.
3	-1.382418	-2.867785	2.453917	-1.144465	4.101465	2.211512	-0.672203	-0.318327	0.853783	-0.987209	-1.866509	-1
4	4.150873	-1.447914	-1.971437	-0.262210	0.202581	-1.875030	0.538392	0.754126	0.708139	0.591011	-0.785617	-0

In [154]:

```python
new_df2.head()
```

	Age	Experience	bGender	bCountry	bFinanced
0	4.483296	-2.612615	-0.711419	0.080674	-0.043393
1	-2.782444	-2.869428	0.124746	-0.564251	-0.210609
2	3.983657	-0.434787	-0.586350	0.094664	0.952393
3	-5.941320	-3.135740	-0.023798	0.824460	-0.209773
4	1.791333	3.196785	-0.770411	0.105705	-0.033203

In [155]:

```python
new_df.to_csv('scaled_df1.csv', index=False)
new_df2.to_csv('scaled_df2.csv', index=False)
```

kmeans clustering aka segmentation

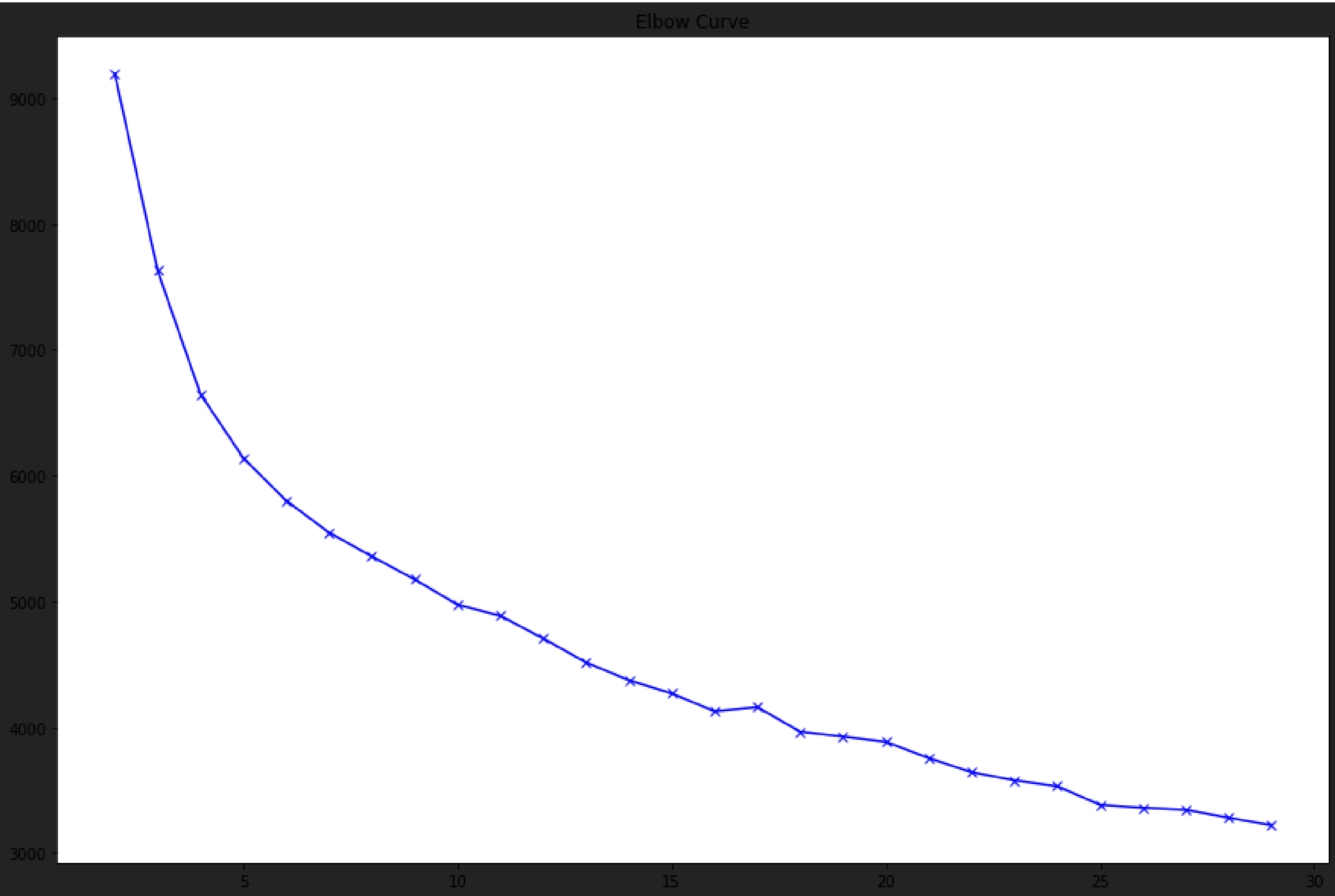
In [78]:

```python
X1 = new_df
distortions = []
for k in range(2,30):
    k_means = KMeans(n_clusters = k)
    k_means.fit(X1)
```

```
    distortions.append(k_means.inertia_)

fig = plt.figure(figsize=(15,10))
plt.plot(range(2,30), distortions, 'bx-')
plt.title("Elbow Curve")
```

```
Text(0.5, 1.0, 'Elbow Curve')
```

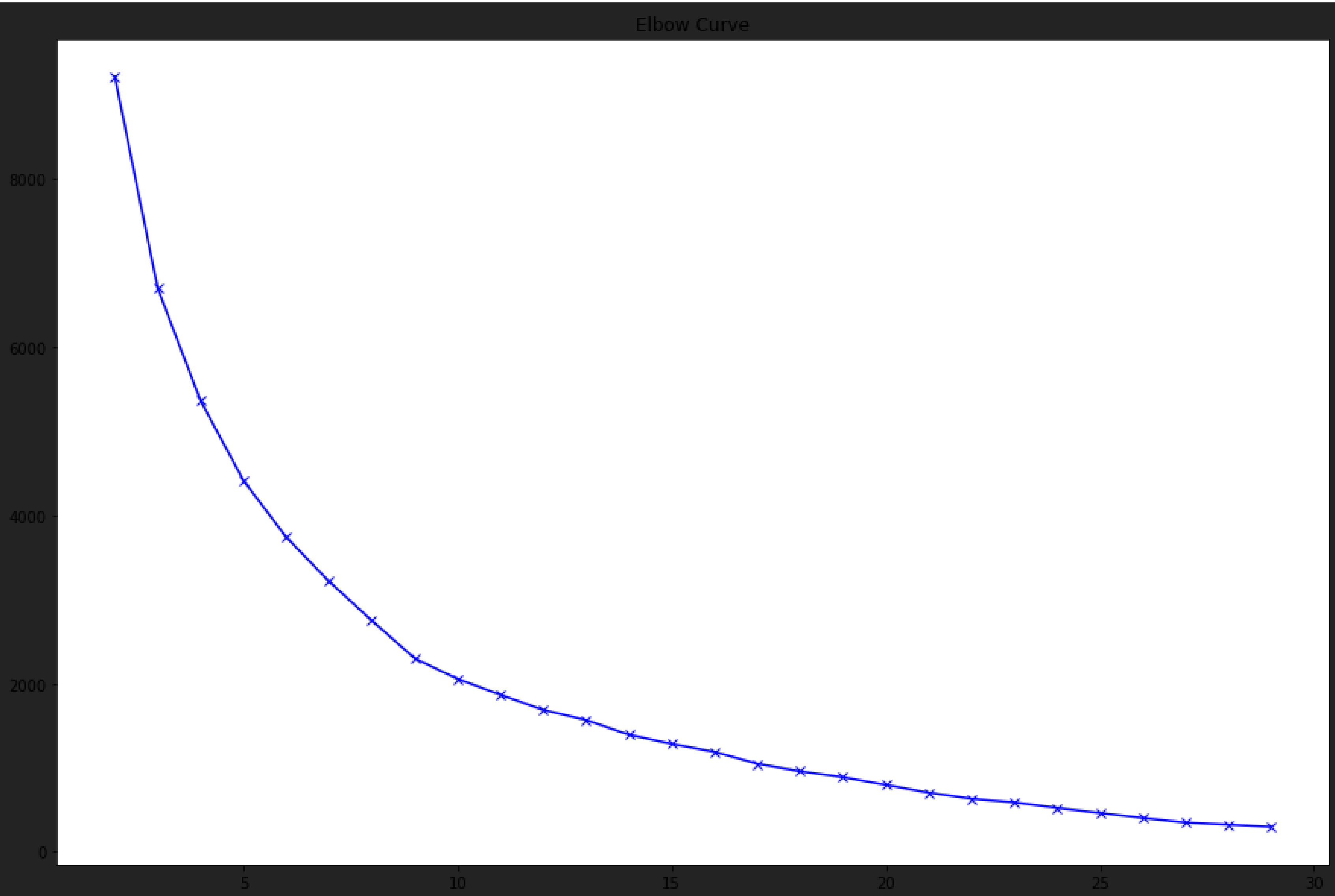


In [79]:
```
X2 = new_df2
distortions = []
for k in range(2,30):
```

```
    k_means = KMeans(n_clusters = k)
    k_means.fit(X2)
    distortions.append(k_means.inertia_)

fig = plt.figure(figsize=(15,10))
plt.plot(range(2,30), distortions, 'bx-')
plt.title("Elbow Curve")
```

```
Text(0.5, 1.0, 'Elbow Curve')
```




```
In [76]: CENTROIDS_df = 7
k_means_df = KMeans(n_clusters = CENTROIDS_df)
#kmeans_df = pd.DataFrame(k_means.fit_transform(new_df))
```

```
X1['idx'] = k_means.fit_predict(new_df)
X1
```

	Networking	Reputation of the school	Changing company	Academic knowledge	Cost-driven	Location	Ranking of the program	Increasing your salary	Launching your own company	Personal dev	Changing career
0	-0.864621	3.246970	2.977510	2.885319	0.320777	-1.566546	-0.236069	-1.886132	-0.385450	1.147241	-0.256647
1	-0.864621	3.246970	2.977510	2.885319	0.320777	-1.566546	-0.236069	-1.886132	-0.385450	1.147241	-0.256647
2	-2.801328	2.951599	0.075888	-1.454817	-0.736540	-1.036595	-3.259385	1.839996	-1.176350	-1.134459	0.692407
3	-1.382418	-2.867785	2.453917	-1.144465	4.101465	2.211512	-0.672203	-0.318327	0.853783	-0.987209	-1.866509
4	4.150873	-1.447914	-1.971437	-0.262210	0.202581	-1.875030	0.538392	0.754126	0.708139	0.591011	-0.785617
...	...	...	...	...	...	...	...	...	...	...	...
192	0.764847	1.709765	-0.495760	-1.685259	-1.900965	2.936704	-1.385588	-0.111770	-2.161464	3.175317	-0.258228
193	-1.923550	5.451500	-0.827410	0.397593	-3.259149	2.147808	2.857874	1.227871	0.567607	-0.766280	-2.594120
194	0.830345	-0.157907	-0.960533	-0.753141	-0.485773	1.545426	-1.755626	-0.236817	-1.950126	-0.347470	-1.824623
195	-2.438497	-3.152476	-0.217916	1.794988	-1.921662	-2.327751	-1.206735	1.884174	-1.762781	-1.360205	-0.363577
196	-0.235236	1.476507	-3.009335	0.419355	1.852891	1.077462	-0.150166	-0.547856	-0.133756	-0.353544	-1.613892

197 rows × 19 columns

In [77]:

```
CENTROIDS_df2 = 9
k_means_df2 = KMeans(n_clusters = CENTROIDS_df2)
#kmeans_df = pd.DataFrame(k_means.fit_transform(new_df))
X2['idx'] = k_means.fit_predict(new_df2)
X2
```

	Age	Experience	bGender	bCountry	bFinanced	idx
0	4.482537	-2.612852	0.696923	0.079458	-0.0	10

	Age	Experience	bGender	bCountry	bFinanced	idx
1	-2.783300	-2.870050	-0.156407	-0.563423	0.0	18
2	3.988315	-0.432278	0.723575	0.090893	-0.0	3
3	-5.942214	-3.136531	-0.006289	0.824998	-0.0	5
4	1.790312	3.196426	0.756841	0.104350	0.0	28
...	...	...	...	...	...	...
192	-0.143063	3.688431	-0.042984	-0.512313	-0.0	11
193	-0.124007	-0.430435	-0.712530	0.253382	0.0	16
194	2.054940	0.059727	-0.076249	-0.525769	-0.0	16
195	-4.019042	0.505211	0.672207	0.064232	-0.0	21
196	-1.317964	-5.289186	-0.178584	-0.572394	0.0	4

197 rows × 6 columns

calc silhoutte score

In [82]:

```
silhouette_scores = []
for clusters in range (2, 15):
    silhouette_scores.append(silhouette_score(new_df, KMeans(n_clusters=clusters).fit_pr
edict(new_df)))
silhouette_scores
```

```
[0.42892582330701395,
 0.28652223843784785,
 0.20709437182422633,
 0.1918260811007373,
 0.15779337054515635,
 0.15750220565281872,
 0.14695322505321767,
 0.15289063195803013,
 0.1534116372670955,
```

```
 0.1574182336056877,
 0.16146102624303493,
 0.1637853694254049,
 0.1638382889015344]
```

In [84]:
```
silhouette_scores_df2 = []
for clusters in range (2, 30):
    silhouette_scores_df2.append(silhouette_score(new_df2, KMeans(n_clusters=clusters).f
it_predict(new_df2)))
silhouette_scores_df2
```

```
[0.4399142714648029,
 0.3746435170940155,
 0.3992416318825048,
 0.40374757284722085,
 0.41214236947561456,
 0.429673620140503,
 0.4517054957124213,
 0.4666596549473109,
 0.476580665873192,
 0.4645099324809888,
 0.5021643866381706,
 0.4877915947730886,
 0.500457913538215,
 0.5297794132586645,
 0.5184083108072808,
 0.5411611485321279,
 0.5384357670475921,
 0.5475316440592597,
 0.5595342774274862,
 0.557465685179288,
 0.566994635582538,
 0.5640793604589242,
 0.5801305210593586,
 0.5795186130729701,
 0.593376673023028,
 0.5890539124089897,
```

```
0.5929027876984826,
0.5841578871756462]
```

final decision

we should use 3 or 4 clusters since 'the standard' is not to trust anything under ~0.20

In [ ]: