AI61003 LINEAR ALGEBRA FOR AI AND ML
ASSIGNMENT 01 - PROBLEM 10

CASE 1 k = 4

The document-cluster association produced by the clustering algorithm is as follows.

S.No.	Article Title	Cluster ID
01	*Linear Algebra*	3
02	*Data Science*	1
03	*Artificial Intelligence*	1
04	*European Central Bank*	4
05	*Financial Technology*	4
06	*International Monetary Fund*	4
07	*Basketball*	2
08	*Swimming*	2
09	*Cricket*	2

Cluster ID	Assignments
01	$\{Data\ Science,\ Artificial\ Intelligence\}$
02	$\{Basketball,\ Swimming,\ Cricket\}$
03	$\{Linear\ Algebra\}$
04	$\{European\ Central\ Bank,\ Financial\ Technology,\ International\ Monetary\ Fund\}$

CASE 2 k = 8

The document-cluster association produced by the clustering algorithm is as follows.

S.No.	Article Title	Cluster ID
01	*Linear Algebra*	7
02	*Data Science*	1
03	*Artificial Intelligence*	2
04	*European Central Bank*	6
05	*Financial Technology*	3
06	*International Monetary Fund*	4
07	*Basketball*	5
08	*Swimming*	8
09	*Cricket*	5

Cluster ID	Assignments
01	$\{Data\ Science\}$

02	$\{Artificial\ Intelligence\}$
03	$\{Financial\ Technology\}$
04	$\{International\ Monetary\ Fund\}$
05	$\{Basketball,\ Cricket\}$
06	$\{European\ Central\ Bank\}$
07	$\{Linear\ Algebra\}$
08	$\{Swimming\}$

CASE 3 k = 12

9 objects cannot be clustered into 12 groups.

CHOOSING THE BEST CLUSTER SIZE

Let us first try to *manually* divide the nine articles into some compact clusters. Clearly, we can see three broad domains to which the articles belong.

- **Academia** : *Linear Algebra, Data Science, Artificial Intelligence*
- **Finance** : *Financial Technology, European Central Bank, International Monetary Fund*
- **Sports** : *Basketball, Swimming, Cricket*

Since we have to choose the best *k* value out of 4 and 8, we must select the one that more prominently expresses the underlying relationship among the articles in the form of the above three categories.

Choosing k = 8 value to cluster 9 objects is a very bad idea because like this every cluster (except one) would get a single object and hence it would not be possible to deduce any meaningful relationship among the objects. This behaviour is prominent when the *k-means algorithm* was run on the feature vectors of the nine articles for k = 8. As we can see the resulting classifications, the only information we can deduce is that articles on *Basketball* and *Cricket* are similar. Besides that, one can draw no other conclusion about the articles from the produced clustering.

On the other hand, choosing k = 4 gives excellent results. As we can see the resulting classifications, all articles on ***sports*** and ***finance*** are grouped under the same cluster. More importantly, the clusters corresponding to these two groups do not contain any other irrelevant articles (no contamination). This means that these two domains were perfectly identified by the algorithm. Now as far as the remaining three objects in the domain of ***academia*** are concerned, two of them fit inside the same cluster, that is better than all three of them being in different clusters. So it is clear that choosing k = 4 gives a lot more information about the semantic relatedness among the articles that are clustered.

Therefore for this data, k = 4 is the better choice.

In [1]:
```
import wikipedia
import numpy as np
from random import shuffle
from sklearn.feature_extraction.text import TfidfVectorizer
import sklearn.cluster as Clustering_Algorithm
```

In [2]:
```
def Get_Distance ( a , b ) :
    return np.linalg.norm(b-a)

def J_clust ( train , kmeans ) :
    s = 0
    for img in train :
        cluster_id = kmeans.predict(img.astype(np.float).reshape(1,-1))[0]
        s += Get_Distance(img, kmeans.cluster_centers_[cluster_id])
    return s / len(train)
```

In [3]:
```
articles = ['Linear Algebra', 'Data Science', 'Artificial intelligence', 'European Central Bank',
            'Financial technology', 'International Monetary Fund', 'Basketball', 'Swimming', 'Cricket']
```

In [4]:
```
wiki_content = []
for article in articles :
    print('Loading content from article :', article)
    wiki_content.append(wikipedia.page(article).content)
```

```
Loading content from article : Linear Algebra
Loading content from article : Data Science
Loading content from article : Artificial intelligence
Loading content from article : European Central Bank
Loading content from article : Financial technology
Loading content from article : International Monetary Fund
Loading content from article : Basketball
Loading content from article : Swimming
Loading content from article : Cricket
```

In [5]:
```
vectorizer = TfidfVectorizer(stop_words={'english'})
X = vectorizer.fit_transform(wiki_content).toarray()
```

In [6]:
```
def Do_Clustering ( train , k ) :
    kmeans = Clustering_Algorithm.KMeans(n_clusters=k, init='random', n_init=1)
    kmeans.fit(train)
    classes = list(kmeans.labels_)
    for i, cl in enumerate(classes) :
        print(articles[i], cl+1)
```

In [7]:
```
1 Do_Clustering(X, 4)
```

```
Linear Algebra 3
Data Science 1
Artificial intelligence 1
European Central Bank 4
Financial technology 4
International Monetary Fund 4
Basketball 2
Swimming 2
Cricket 2
```

In [8]:
```
1 Do_Clustering(X, 8)
```

```
Linear Algebra 7
Data Science 1
Artificial intelligence 2
European Central Bank 6
Financial technology 3
International Monetary Fund 4
Basketball 5
Swimming 8
Cricket 5
```