Our data has 3 dimensions

1. Year of the top phone release(2016,2017,2020)
2. Attributes of the phone(Display,Storage,Battery,Price,Camera(mp))
3. Company of a phone(Samsung,Apple,Google)

So order of the data is 3x3x5

Samsung:

	Display	Stroage	Battery	Price	Camera(mp)
2016	5.5	256	3600	823	12
2017	6.2	256	3500	825	12
2020	6.3	256	4300	1099	16

Apple:

	Display	Stroage	Battery	Price	Camera(mp)
2016	5.5	256	2900	1081	12
2017	5.5	256	2675	949	12
2020	6.5	256	3969	1499	12

Google:

	Display	Stroage	Battery	Price	Camera(mp)
2016	5.5	128	2700	963	12.3
2017	6.0	128	3300	949	12.2
2020	6.3	128	3700	930	12.2

In [1]:

```
# import libraries
import numpy as np
```

In [2]:

```
# Save the data in a matrix arr
arr=np.array([ [
                 [5.5,256,3600,823,12],
                 [6.2,256,3500,825,12],
                 [6.3,256,4300,1099,16] ],
               [
                 [5.5,256,2900,1081,12],
                 [5.5,256,2675,949,12],
                 [6.5,256,3696,1499,12] ],
               [
                 [5.5,128,2700,963,12.3],
                 [6.0,128,3300,949,12.2],
                 [6.3,128,3700,930,12.2] ] ] )
# Print the order of array
print(np.shape(arr))
# Print the array
print(arr)
```

```
(3, 3, 5)
[[[   5.5  256.  3600.   823.    12. ]
  [   6.2  256.  3500.   825.    12. ]
  [   6.3  256.  4300.  1099.    16. ]]

 [[   5.5  256.  2900.  1081.    12. ]
  [   5.5  256.  2675.   949.    12. ]
  [   6.5  256.  3696.  1499.    12. ]]
```

```
 [[   5.5  128.  2700.   963.    12.3]
  [   6.   128.  3300.   949.    12.2]
  [   6.3  128.  3700.   930.    12.2]]]
```

Let us try to collapse it to 2 dimension first

Now suppose we want see how the attributes of phones are changing across the years irrespective of company we can collapse the company dimension by taking average of attributes for a given year

In [3]:

```
# Take mean over for a particular year over different companies
Dim2_1=np.mean(arr,axis=0)
# Print the order of array
print(np.shape(Dim2_1))
# Print the array
print(Dim2_1)
```

```
(3, 5)
[[   5.5         213.33333333 3066.66666667  955.66666667   12.1       ]
 [   5.9         213.33333333 3158.33333333  907.66666667   12.06666667]
 [   6.36666667  213.33333333 3898.66666667 1176.           13.4       ]]
```

Looking at the data now we can see that year by the all the features and the price keep increasing

Now suppose we want to compare how the attributes of phones are varying with companies on a paricular year we can collapse the year dimension by chossing only a particular year or by taking sum for all companies

In [4]:

```
# Take mean over for a particular year over different companies
Dim2_2=np.mean(arr,axis=1)
# Print the order of array
print(np.shape(Dim2_2))
# Print the array
print(Dim2_2)
```

```
(3, 5)
[[   6.          256.         3800.          915.66666667   13.33333333]
 [   5.83333333  256.         3090.33333333 1176.33333333   12.        ]
 [   5.93333333  128.         3233.33333333  947.33333333   12.23333333]]
```

Looking at the data now we can see that features like display size or battery capacity are more for Samsung company while the price is more for Apple

Now only one 2 dimensional reduction is left. That is collapsing the attributes dimension, It doesn't make sense if we take average or max in this dimenson as different attribute cannot be added or compared. Insteasd we choose only one attribute to collapse the whole dimension

In [5]:

```
Dim2_3=arr[:,:,3]
# Print the order of array
print(np.shape(Dim2_3))
# Print the array
print(Dim2_3)
```

```
(3, 3)
[[ 823.  825. 1099.]
 [1081.  949. 1499.]
 [ 963.  949.  930.]]
```

Looking at the data now we can see that Samsung keeps increasing it's price year by year. While google tries to make them cheaper

As we are done with collapsing 1 dimension let's collapse 2 dimensions

There are many possible cases so let's decide what we want first and then collapse the dimensions

1. Suppose we want see how storages of phones are varying by years.We need to collapse the attribute dimension by choosing only 1 attribute and the company dimension by taking average over companies

In [6]:

```
# As we already have the array with average over companies now we need to choose the storage attri
bute
Dim1_1=Dim2_1[:,1]
# Print the order of array
print(np.shape(Dim1_1))
# Print the array
print(Dim1_1)
```

```
(3,)
[213.33333333 213.33333333 213.33333333]
```

As we can see the storage didn't change over the years. This indicates the we have already reached the maximum storage a phone requires.

2.Now let us see how the battery capacities are varying with different companies. For this we need to collapse the year dimension by taking average over year and the attribute dimension by selscting one attribute

In [7]:

```
# As we already have the array with average over years now we need to choose the battery attribute
Dim1_2=Dim2_2[:,2]
# Print the order of array
print(np.shape(Dim1_2))
# Print the array
print(Dim1_2)
```

```
(3,)
[3800.         3090.33333333 3233.33333333]
```

Looking at the array we can conclude that Samsung offers the most battery capacity while Apple offers the least battery capacity

In [ ]:

```

```