In [0]:
```
from scipy import stats
import numpy as np
```

In [3]:
```
# Membangkitkan data normal random sebanyak 10
contoh = np.random.randn(10)
contoh
```

Out[3]:
```
array([ 1.30259508, -0.93182432,  1.23273377,  0.02925403, -0.66225749,
       -0.48258097,  1.399893  ,  0.97610898, -0.41725827,  2.09868991])
```

In [4]:
```
print (contoh.max(), contoh.min(), contoh.mean(), contoh.var())
```

```
2.098689908319387 -0.9318243229554479 0.45453537274856143 1.0182077131783405
```

In [5]:
```
stats.describe(contoh)
```

Out[5]:
```
DescribeResult(nobs=10, minmax=(-0.9318243229554479, 2.098689908319387), mean=0.45453537274856143, variance=1.1
313419035314896, skewness=0.0868739431858027, kurtosis=-1.4860889298610795)
```

In [6]:
```
stats.mode(contoh)
```

Out[6]:
```
ModeResult(mode=array([-0.93182432]), count=array([1]))
```

In [0]:
```
# membuat 2 contoh random

contoh1 = np.random.randint(45, size=(10))
contoh2 = np.random.randint(90, size=(10))
contoh3 = np.random.randint(120, size=(10))
```

In [8]:
```
# Menampilkan contoh 1

contoh1
```

Out[8]:
```
array([29, 14, 28, 12, 17,  7, 39,  5, 21, 15])
```

In [9]:
```
# Melakuakan ttest 1 populasi (apakah rata-rata contoh 1 sama dengan 15?)

stats.ttest_1samp(contoh1, 15)
```

Out[9]:
```
Ttest_1sampResult(statistic=1.1000888428374092, pvalue=0.2998460685995545)
```

In [10]:

```
# Melakukan ttest 2 populasi (apakah rata-rata contoh 1 dan 2 sama? Tanpa adanya asumsi variansi sama)

stats.ttest_ind(contoh1, contoh2, equal_var=False)
```

Out[10]:

```
Ttest_indResult(statistic=-3.402988379720548, pvalue=0.0057094693355698115)
```

In [11]:

```
# ANOVA digunakan untuk populasi lebih dari 2
# menguji ANOVA contoh 1, 2, dan 3
stats.f_oneway(contoh1, contoh2, contoh3)
```

Out[11]:

```
F_onewayResult(statistic=4.137457390969074, pvalue=0.02707822404718403)
```

In [0]:

```
# SCIPY
import pandas as pd
from scipy.cluster.vq import kmeans,vq,whiten
```

In [16]:

```
from google.colab import files
uploaded = files.upload()
```

Pilih File Tidak ada file yang dipilih

Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

```
Saving banjir.csv to banjir.csv
```

In [0]:

```
banjir = pd.read_csv("banjir.csv")
```

In [19]:
```
banjir.head()
```

Out[19]:

	Unnamed: 0	Meninggal & Hilang	Luka0luka	Menderita & mengungsi	Rumah Rusak Berat	Rumah Rusak Sedang	Rusak Ringan	Rumah Terendam	Fasilitas Kesehatan	Fasilitas Peribadatan	Fasilitas Pendidikan
0	11. ACEH	1	1	38476	24	51	50	3733	2	0	4
1	12. SUMATERA UTARA	5	10	1035	26	86	365	105	1	3	1
2	13. SUMATERA BARAT	1	3	516	6	7	24	128	0	2	4
3	14. RIAU	0	2	553	0	0	5	188	0	2	0
4	15. JAMBI	1	0	442	6	4	67	251	0	5	2

In [20]:
```
banjir.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 34 entries, 0 to 33
Data columns (total 11 columns):
Unnamed: 0               34 non-null object
Meninggal & Hilang       34 non-null int64
Luka0luka                34 non-null int64
Menderita & mengungsi    34 non-null int64
Rumah Rusak Berat        34 non-null int64
Rumah Rusak Sedang       34 non-null int64
Rusak Ringan             34 non-null int64
Rumah Terendam           34 non-null int64
Fasilitas Kesehatan      34 non-null int64
Fasilitas Peribadatan    34 non-null int64
Fasilitas Pendidikan     34 non-null int64
dtypes: int64(10), object(1)
memory usage: 3.0+ KB
```

In [21]:

```
banjir_x = banjir.iloc[:, 1:11]
banjir_x
```

Out[21]:

	Meninggal & Hilang	Luka0luka	Menderita & mengungsi	Rumah Rusak Berat	Rumah Rusak Sedang	Rusak Ringan	Rumah Terendam	Fasilitas Kesehatan	Fasilitas Peribadatan	Fasilitas Pendidikan
0	1	1	38476	24	51	50	3733	2	0	4
1	5	10	1035	26	86	365	105	1	3	1
2	1	3	516	6	7	24	128	0	2	4
3	0	2	553	0	0	5	188	0	2	0
4	1	0	442	6	4	67	251	0	5	2
5	1	1	53110	34	2	227	4533	4	4	10
6	0	0	205	0	1	0	9	0	0	0
7	14	5	11204	23	30	177	2688	0	1	3
8	0	0	4258	0	0	0	169	0	0	0
9	2	0	0	3	0	12	0	0	0	0
10	1	1	12795	23	17	33	0	0	0	0
11	16	36	125810	1523	3482	6627	31946	9	108	61
12	38	144	121144	483	503	2113	26141	2	19	18
13	2	5	8915	1	2	204	0	0	2	4
14	18	37	107004	114	222	1129	23466	1	57	55
15	2	4	12784	160	171	818	2824	1	6	8
16	5	2	53457	39	5	86	462	0	1	2
17	1	68	21277	2343	5916	6751	2011	8	63	73
18	4	1	190	24	13	424	2000	0	0	0
19	1	0	300	1	1	0	333	0	0	1
20	0	0	33322	4	1	1	5470	2	9	18
21	0	4	9748	51	55	101	1084	0	2	4
22	0	0	80578	0	26	1	13800	0	1	1
23	0	0	0	0	0	0	0	0	0	0

	Meninggal & Hilang	Luka0luka	Menderita & mengungsi	Rumah Rusak Berat	Rumah Rusak Sedang	Rusak Ringan	Rumah Terendam	Fasilitas Kesehatan	Fasilitas Peribadatan	Fasilitas Pendidikan
24	3	4	6539	8	4	19	3408	0	0	0
25	0	10	4442	5	0	4	859	0	2	2
26	0	20	39088	90	138	240	4596	7	15	30
27	3	12	423	25	0	5	185	0	0	0
28	4	0	18348	7	1	0	2082	1	1	1
29	0	7	5376	4	10	6	1469	0	0	0
30	1	0	5	0	1	0	0	0	0	0
31	0	0	1166	5	3	44	1485	0	0	0
32	0	0	1800	0	0	0	1451	0	0	0
33	0	132	2892	36	9	14	402	1	2	12

```
In [22]: x_array = np.array(banjir_x)

         x_array
```

```
Out[22]: array([[     1,      1,  38476,     24,     51,     50,   3733,      2,
                      0,      4],
                [     5,     10,   1035,     26,     86,    365,    105,      1,
                      3,      1],
                [     1,      3,    516,      6,      7,     24,    128,      0,
                      2,      4],
                [     0,      2,    553,      0,      0,      5,    188,      0,
                      2,      0],
                [     1,      0,    442,      6,      4,     67,    251,      0,
                      5,      2],
                [     1,      1,  53110,     34,      2,    227,   4533,      4,
                      4,     10],
                [     0,      0,    205,      0,      1,      0,      9,      0,
                      0,      0],
                [    14,      5,  11204,     23,     30,    177,   2688,      0,
                      1,      3],
                [     0,      0,   4258,      0,      0,      0,    169,      0,
                      0,      0],
                [     2,      0,      0,      3,      0,     12,      0,      0,
                      0,      0],
                [     1,      1,  12795,     23,     17,     33,      0,      0,
                      0,      0],
                [    16,     36, 125810,   1523,   3482,   6627,  31946,      9,
                    108,     61],
                [    38,    144, 121144,    483,    503,   2113,  26141,      2,
                     19,     18],
                [     2,      5,   8915,      1,      2,    204,      0,      0,
                      2,      4],
                [    18,     37, 107004,    114,    222,   1129,  23466,      1,
                     57,     55],
                [     2,      4,  12784,    160,    171,    818,   2824,      1,
                      6,      8],
                [     5,      2,  53457,     39,      5,     86,    462,      0,
                      1,      2],
                [     1,     68,  21277,   2343,   5916,   6751,   2011,      8,
                     63,     73],
                [     4,      1,    190,     24,     13,    424,   2000,      0,
                      0,      0],
                [     1,      0,    300,      1,      1,      0,    333,      0,
                      0,      1],
                [     0,      0,  33322,      4,      1,      1,   5470,      2,
                      9,     18],
                [     0,      4,   9748,     51,     55,    101,   1084,      0,
                      2,      4],
                [     0,      0,  80578,      0,     26,      1,  13800,      0,
                      1,      1],
                [     0,      0,      0,      0,      0,      0,      0,      0,
                      0,      0],
                [     3,      4,   6539,      8,      4,     19,   3408,      0,
                      0,      0],
                [     0,     10,   4442,      5,      0,      4,    859,      0,
                      2,      2],
                [     0,     20,  39088,     90,    138,    240,   4596,      7,
                     15,     30],
                [     3,     12,    423,     25,      0,      5,    185,      0,
                      0,      0],
                [     4,      0,  18348,      7,      1,      0,   2082,      1,
                      1,      1],
                [     0,      7,   5376,      4,     10,      6,   1469,      0,
                      0,      0],
                [     1,      0,      5,      0,      1,      0,      0,      0,
                      0,      0],
                [     0,      0,   1166,      5,      3,     44,   1485,      0,
                      0,      0],
                [     0,      0,   1800,      0,      0,      0,   1451,      0,
                      0,      0],
                [     0,    132,   2892,     36,      9,     14,    402,      1,
                      2,     12]])
```

```
In [26]: x_scale = whiten(x_array)

         x_scale
```

```
Out[26]: array([[1.34204021e-01, 2.96247170e-02, 1.09447823e+00, 5.16394409e-02,
                 4.47890412e-02, 3.15878763e-02, 4.83769926e-01, 8.63809875e-01,
                 0.00000000e+00, 2.21179442e-01],
                [6.71020104e-01, 2.96247170e-01, 2.94413391e-02, 5.59427276e-02,
                 7.55266185e-02, 2.30591497e-01, 1.36072441e-02, 4.31904938e-01,
                 1.34453185e-01, 5.52948606e-02],
                [1.34204021e-01, 8.88741509e-02, 1.46780009e-02, 1.29098602e-02,
                 6.14751546e-03, 1.51621806e-02, 1.65878785e-02, 0.00000000e+00,
                 8.96354565e-02, 2.21179442e-01],
                [0.00000000e+00, 5.92494340e-02, 1.57304933e-02, 0.00000000e+00,
                 0.00000000e+00, 3.15878763e-03, 2.43634466e-02, 0.00000000e+00,
                 8.96354565e-02, 0.00000000e+00],
                [1.34204021e-01, 0.00000000e+00, 1.25730163e-02, 1.29098602e-02,
                 3.51286598e-03, 4.23277542e-02, 3.25277931e-02, 0.00000000e+00,
                 2.24088641e-01, 1.10589721e-01],
                [1.34204021e-01, 2.96247170e-02, 1.51075316e+00, 7.31558745e-02,
                 1.75643299e-03, 1.43408958e-01, 5.87444167e-01, 1.72761975e+00,
                 1.79270913e-01, 5.52948606e-01],
                [0.00000000e+00, 0.00000000e+00, 5.83137635e-03, 0.00000000e+00,
                 8.78216494e-04, 0.00000000e+00, 1.16633521e-03, 0.00000000e+00,
                 0.00000000e+00, 0.00000000e+00],
                [1.87885629e+00, 1.48123585e-01, 3.18706052e-01, 4.94877975e-02,
                 2.63464948e-02, 1.11821082e-01, 3.48345449e-01, 0.00000000e+00,
                 4.48177282e-02, 1.65884582e-01],
                [0.00000000e+00, 0.00000000e+00, 1.21121954e-01, 0.00000000e+00,
                 0.00000000e+00, 0.00000000e+00, 2.19011834e-02, 0.00000000e+00,
                 0.00000000e+00, 0.00000000e+00],
                [2.68408041e-01, 0.00000000e+00, 0.00000000e+00, 6.45493011e-03,
                 0.00000000e+00, 7.58109031e-03, 0.00000000e+00, 0.00000000e+00,
                 0.00000000e+00, 0.00000000e+00],
                [1.34204021e-01, 2.96247170e-02, 3.63963221e-01, 4.94877975e-02,
                 1.49296804e-02, 2.08479983e-02, 0.00000000e+00, 0.00000000e+00,
                 0.00000000e+00, 0.00000000e+00],
                [2.14726433e+00, 1.06648981e+00, 3.57875833e+00, 3.27695285e+00,
                 3.05794983e+00, 4.18665712e+00, 4.13997162e+00, 3.88714444e+00,
                 4.84031465e+00, 3.37298650e+00],
                [5.09975279e+00, 4.26595925e+00, 3.44603052e+00, 1.03924375e+00,
                 4.41742897e-01, 1.33490365e+00, 3.38768541e+00, 8.63809875e-01,
                 8.51536836e-01, 9.95307491e-01],
                [2.68408041e-01, 1.48123585e-01, 2.53593757e-01, 2.15164337e-03,
                 1.75643299e-03, 1.28878535e-01, 0.00000000e+00, 0.00000000e+00,
                 8.96354565e-02, 2.21179442e-01],
                [2.41567237e+00, 1.09611453e+00, 3.04380778e+00, 2.45287344e-01,
                 1.94964062e-01, 7.13254246e-01, 3.04102467e+00, 4.31904938e-01,
                 2.55461051e+00, 3.04121733e+00],
                [2.68408041e-01, 1.18498868e-01, 3.63650318e-01, 3.44262939e-01,
                 1.50175021e-01, 5.16777656e-01, 3.65970070e-01, 4.31904938e-01,
                 2.68906369e-01, 4.42358885e-01],
                [6.71020104e-01, 5.92494340e-02, 1.52062383e+00, 8.39140914e-02,
                 4.39108247e-03, 5.43311472e-02, 5.98718741e-02, 0.00000000e+00,
                 4.48177282e-02, 1.10589721e-01],
                [1.34204021e-01, 2.01448075e+00, 6.05239973e-01, 5.04130041e+00,
                 5.19552878e+00, 4.26499505e+00, 2.60611123e-01, 3.45523950e+00,
                 2.82351688e+00, 4.03652482e+00],
                [5.36816083e-01, 2.96247170e-02, 5.40469027e-03, 5.16394409e-02,
                 1.14168144e-02, 2.67865191e-01, 2.59185602e-01, 0.00000000e+00,
                 0.00000000e+00, 0.00000000e+00],
                [1.34204021e-01, 0.00000000e+00, 8.53372148e-03, 2.15164337e-03,
                 8.78216494e-04, 0.00000000e+00, 4.31544027e-02, 0.00000000e+00,
                 0.00000000e+00, 5.52948606e-02],
                [0.00000000e+00, 0.00000000e+00, 9.47868891e-01, 8.60657348e-03,
                 8.78216494e-04, 6.31757526e-04, 7.08872622e-01, 8.63809875e-01,
                 4.03359554e-01, 9.95307491e-01],
                [0.00000000e+00, 1.18498868e-01, 2.77289057e-01, 1.09733812e-01,
                 4.83019072e-02, 6.38075101e-02, 1.40478596e-01, 0.00000000e+00,
                 8.96354565e-02, 2.21179442e-01],
                [0.00000000e+00, 0.00000000e+00, 2.29210070e+00, 0.00000000e+00,
                 2.28336289e-02, 6.31757526e-04, 1.78838065e+00, 0.00000000e+00,
                 4.48177282e-02, 5.52948606e-02],
                [0.00000000e+00, 0.00000000e+00, 0.00000000e+00, 0.00000000e+00,
                 0.00000000e+00, 0.00000000e+00, 0.00000000e+00, 0.00000000e+00,
                 0.00000000e+00, 0.00000000e+00],
                [4.02612062e-01, 1.18498868e-01, 1.86006683e-01, 1.72131470e-02,
                 3.51286598e-03, 1.20033930e-02, 4.41652266e-01, 0.00000000e+00,
                 0.00000000e+00, 0.00000000e+00],
                [0.00000000e+00, 2.96247170e-01, 1.26355969e-01, 1.07582168e-02,
                 0.00000000e+00, 2.52703010e-03, 1.11320216e-01, 0.00000000e+00,
                 8.96354565e-02, 1.10589721e-01],
                [0.00000000e+00, 5.92494340e-01, 1.11188702e+00, 1.93647903e-01,
                 1.21193876e-01, 1.51621806e-01, 5.95608514e-01, 3.02333456e+00,
                 6.72265923e-01, 1.65884582e+00],
                [4.02612062e-01, 3.55496604e-01, 1.20325473e-02, 5.37910842e-02,
                 0.00000000e+00, 3.15878763e-03, 2.39746682e-02, 0.00000000e+00,
                 0.00000000e+00, 0.00000000e+00],
                [5.36816083e-01, 0.00000000e+00, 5.21922406e-01, 1.50615036e-02,
                 8.78216494e-04, 0.00000000e+00, 2.69812212e-01, 4.31904938e-01,
                 4.48177282e-02, 5.52948606e-02],
                [0.00000000e+00, 2.07373019e-01, 1.52924289e-01, 8.60657348e-03,
                 8.78216494e-03, 3.79054515e-03, 1.90371825e-01, 0.00000000e+00,
                 0.00000000e+00, 0.00000000e+00],
                [1.34204021e-01, 0.00000000e+00, 1.42228691e-04, 0.00000000e+00,
                 8.78216494e-04, 0.00000000e+00, 0.00000000e+00, 0.00000000e+00,
                 0.00000000e+00, 0.00000000e+00],
                [0.00000000e+00, 0.00000000e+00, 3.31677308e-02, 1.07582168e-02,
                 2.63464948e-03, 2.77973311e-02, 1.92445310e-01, 0.00000000e+00,
                 0.00000000e+00, 0.00000000e+00],
                [0.00000000e+00, 0.00000000e+00, 5.12023289e-02, 0.00000000e+00,
                 0.00000000e+00, 0.00000000e+00, 1.88039154e-01, 0.00000000e+00,
                 0.00000000e+00, 0.00000000e+00],
                [0.00000000e+00, 3.91046264e+00, 8.22650751e-02, 7.74591613e-02,
                 7.90394845e-03, 8.84460536e-03, 5.20963060e-02, 4.31904938e-01,
                 8.96354565e-02, 6.63538327e-01]])
```

In [32]:

```
# computing K-Means with K = 3 (3 clusters)

centroids,_ = kmeans(x_scale,3)
print(centroids)
```

```
[[1.14073418 1.54048528 2.09199915 4.15912663 4.12673931 4.22582609
  2.20029137 3.67119197 3.83191576 3.70475566]
 [0.22814684 0.22119789 0.3811416  0.04339147 0.01867674 0.06163848
  0.23203159 0.27353979 0.08664761 0.19721834]
 [3.75771258 2.68103689 3.24491915 0.64226555 0.31835348 1.02407895
  3.21435504 0.64785741 1.70307367 2.01826241]]
```

In [0]:

```
# assign each sample to a cluster
clx,_ = vq(x_scale,centroids)
```

In [34]:

```
print(clx)
```

```
[1 1 1 1 1 1 1 1 1 1 1 0 2 1 2 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1]
```

In [0]:

```
kluster = pd.DataFrame({"provinsi": banjir.iloc[:,0],
"kluster": np.array(clx)})
```

In [36]:
```
print(kluster)
```

```
    kluster                       provinsi
0         1                       11. ACEH
1         1             12. SUMATERA UTARA
2         1             13. SUMATERA BARAT
3         1                       14. RIAU
4         1                      15. JAMBI
5         1           16. SUMATERA SELATAN
6         1                   17. BENGKULU
7         1                    18. LAMPUNG
8         1  19. KEPULAUAN BANGKA BELITUNG
9         1             21. KEPULAUAN RIAU
10        1                31. DKI JAKARTA
11        0                 32. JAWA BARAT
12        2                33. JAWA TENGAH
13        1              34. DI YOGYAKARTA
14        2                 35. JAWA TIMUR
15        1                     36. BANTEN
16        1                       51. BALI
17        0        52. NUSA TENGGARA BARAT
18        1        53. NUSA TENGGARA TIMUR
19        1           61. KALIMANTAN BARAT
20        1          62. KALIMANTAN TENGAH
21        1         63. KALIMANTAN SELATAN
22        1           64. KALIMANTAN TIMUR
23        1           65. KALIMANTAN UTARA
24        1             71. SULAWESI UTARA
25        1            72. SULAWESI TENGAH
26        1           73. SULAWESI SELATAN
27        1          74. SULAWESI TENGGARA
28        1                  75. GORONTALO
29        1             76. SULAWESI BARAT
30        1                     81. MALUKU
31        1               82. MALUKU UTARA
32        1                91. PAPUA BARAT
33        1                      94. PAPUA
```

In [0]:
```
banjir["clx"]=clx
```

In [38]:
```
banjir.head()
```

Out[38]:

	Unnamed: 0	Meninggal & Hilang	Luka0luka	Menderita & mengungsi	Rumah Rusak Berat	Rumah Rusak Sedang	Rusak Ringan	Rumah Terendam	Fasilitas Kesehatan	Fasilitas Peribadatan	Fasilitas Pendidikan	clx
0	11. ACEH	1	1	38476	24	51	50	3733	2	0	4	1
1	12. SUMATERA UTARA	5	10	1035	26	86	365	105	1	3	1	1
2	13. SUMATERA BARAT	1	3	516	6	7	24	128	0	2	4	1
3	14. RIAU	0	2	553	0	0	5	188	0	2	0	1
4	15. JAMBI	1	0	442	6	4	67	251	0	5	2	1

In [40]:
```
kluster1=banjir[banjir.clx==1]
print(kluster1)
```

```
                       Unnamed: 0  Meninggal & Hilang  Luka0luka  \
0                        11. ACEH                   1          1
1               12. SUMATERA UTARA                  5         10
2               13. SUMATERA BARAT                  1          3
3                        14. RIAU                   0          2
4                       15. JAMBI                   1          0
5             16. SUMATERA SELATAN                  1          1
6                    17. BENGKULU                   0          0
7                     18. LAMPUNG                  14          5
8    19. KEPULAUAN BANGKA BELITUNG                  0          0
9               21. KEPULAUAN RIAU                  2          0
10                 31. DKI JAKARTA                  1          1
13              34. DI YOGYAKARTA                   2          5
15                      36. BANTEN                  2          4
16                        51. BALI                  5          2
18         53. NUSA TENGGARA TIMUR                  4          1
19            61. KALIMANTAN BARAT                  1          0
20           62. KALIMANTAN TENGAH                  0          0
21          63. KALIMANTAN SELATAN                  0          4
22            64. KALIMANTAN TIMUR                  0          0
```

In [0]:
```
karakteristik=banjir.groupby("clx")
```

In [47]:
```
karakteristik.mean()
```

Out[47]:

	Meninggal & Hilang	Luka0luka	Menderita & mengungsi	Rumah Rusak Berat	Rumah Rusak Sedang	Rusak Ringan	Rumah Terendam	Fasilitas Kesehatan	Fasilitas Peribadatan	Fasilitas Pendidikan
clx										
0	8.5	52.000000	73543.5	1933.000000	4699.000000	6689.000000	16978.500000	8.500000	85.500000	67.000000
1	1.7	7.466667	13398.9	20.166667	21.266667	97.566667	1790.466667	0.633333	1.933333	3.566667
2	28.0	90.500000	114074.0	298.500000	362.500000	1621.000000	24803.500000	1.500000	38.000000	36.500000

In [0]:
```
```