Machine Learning to identify whether a vehicle is going to a correct path

Kindly take reference to the image given in the same folder

First to generate desired GPS trails I have used https://www.findlatitudeandlongitude.com/click-lat-lng-list/#.WaalPtMjEUs (https://www.findlatitudeandlongitude.com/click-lat-lng-list/#.WaalPtMjEUs)

As you can see in Raw_Data11.xlsx file the two successive columns are considered one GPS trail as you can see below

Thus the following Data have 10 GPS trail among which first 7 are for one path that the cab driver use to take and another 3 are for deviated path that it will take

The reason behind taking both the path during learning phase is that machine must have to know the desired path's that it can take during his journey

Reading the Data from the Excel file(Raw_Data11.xlsx) and Naming the columns as ["a","b","c","d","e","f","g","h","i","j","k","l","m","n","o

In [75]:
```
import pandas as pd
Raw_Data = pd.ExcelFile("Raw_Data11.xlsx")
print(Raw_Data.sheet_names)
Data_Fram = Raw_Data.parse("Sheet 1")
print(Data_Fram)
```

```
['Sheet 1', 'Sheet 2']
      Table 1  Unnamed: 1  Unnamed: 2  Unnamed: 3  Unnamed: 4  Unn
```

```
amed: 5  \
0         NaN         NaN         NaN         NaN         NaN
NaN
1   18.592725   73.819842   18.592765   73.819585   18.592752   73
.819571
2   18.592074   73.818984   18.592074   73.818727   18.592142   73
.818841
3   18.591586   73.817954   18.591260   73.817353   18.591735   73
.818111
4   18.591098   73.816838   18.590243   73.816752   18.591165   73
.817339
5   18.589755   73.816710   18.588860   73.816667   18.590108   73
.817081
6   18.588128   73.816581   18.587518   73.816366   18.588033   73
.816652
7   18.586176   73.816237   18.586013   73.816366   18.585959   73
.816223
8   18.584020   73.815894   18.584549   73.815894   18.584088   73
.815837
9   18.581864   73.815551   18.583084   73.815722   18.582176   73
.815622
10  18.579993   73.815894   18.581742   73.815594   18.580589   73
.815751
11  18.577877   73.815937   18.580237   73.815937   18.579043   73
.816009
12  18.575843   73.815637   18.578691   73.816109   18.577538   73
.815794
13  18.575437   73.814178   18.577348   73.815680   18.576033   73
.815494
14  18.575884   73.812718   18.576047   73.815508   18.575464   73
.814507
15  18.575681   73.811259   18.575396   73.814349   18.575748   73
.813262
16  18.575437   73.809800   18.576006   73.812633   18.576115   73
.811760
17  18.574908   73.808470   18.575843   73.811173   18.575504   73
.809958
18  18.574053   73.807440   18.575437   73.809543   18.574894   73
.808198
19  18.573484   73.806581   18.574786   73.808169   18.573714   73
.806782
20  18.572589   73.807096   18.573891   73.807182   18.572819   73
.806825
21  18.571694   73.807611   18.573077   73.806882   18.571925   73
.807640
22  18.570799   73.808126   18.572264   73.807783   18.570745   73
.808455
23  18.569904   73.808641   18.571124   73.808298   18.569484   73
.809571
24  18.569497   73.809714   18.569985   73.809028   18.568914   73
.810987
25  18.569131   73.810787   18.569131   73.810401   18.568019   73
.811417
```

```
26  18.568155   73.811517   18.568155   73.811173   18.566758   73
.811803
27  18.566690   73.811731   18.566934   73.811688   18.565416   73
.812361
28  18.565388   73.812160   18.565632   73.812203   18.564073   73
.812919
29  18.564168   73.812718   18.564331   73.812718   18.562730   73
.813477
30  18.562947   73.813276   18.563029   73.813233   18.561388   73
.814034
31  18.561727   73.813834   18.561727   73.813748   18.560208   73
.814850

    Unnamed: 6  Unnamed: 7  Unnamed: 8  Unnamed: 9  Unnamed: 10  U
nnamed: 11  \
0          NaN         NaN         NaN         NaN          NaN
NaN
1    18.592806   73.819499   18.592887   73.819542   18.592887
73.819585
2    18.591952   73.818512   18.592236   73.818855   18.592236
73.818812
3    18.591301   73.817482   18.591586   73.818083   18.591586
73.818040
4    18.590284   73.816838   18.590935   73.816967   18.591057
73.817225
5    18.589064   73.816710   18.589959   73.816795   18.589999
73.816795
6    18.587518   73.816452   18.588982   73.816624   18.588820
73.816667
7    18.585688   73.816195   18.587518   73.816495   18.587640
73.816538
8    18.583857   73.815594   18.585972   73.816152   18.586460
73.816409
9    18.582311   73.815336   18.584304   73.815808   18.585281
73.816066
10   18.580766   73.815551   18.582474   73.815551   18.584101
73.815765
11   18.579220   73.816066   18.580643   73.815637   18.582921
73.815637
12   18.577715   73.815765   18.579138   73.816023   18.581742
73.815508
13   18.576169   73.815465   18.577837   73.815765   18.580440
73.815722
14   18.575192   73.814650   18.576535   73.815508   18.579138
73.816152
15   18.575965   73.813148   18.575233   73.815250   18.577837
73.815808
16   18.576250   73.811603   18.575518   73.814092   18.576494
73.815551
17   18.575681   73.810101   18.576087   73.812933   18.575274
73.815379
18   18.575111   73.808599   18.575925   73.811603   18.575396
73.814220
```

```
19   18.574257   73.807611   18.575599   73.810229    18.575965
73.813019
20   18.573403   73.806667   18.575274   73.808856    18.576128
73.811731
21   18.572589   73.807182   18.574582   73.808084    18.575762
73.810358
22   18.571775   73.807697   18.573850   73.807397    18.575355
73.808899
23   18.570962   73.808212   18.573118   73.806624    18.574623
73.807912
24   18.570148   73.808727   18.572060   73.807569    18.573809
73.806925
25   18.569334   73.809242   18.571002   73.808341    18.572792
73.807011
26   18.569131   73.810616   18.569945   73.808942    18.571124
73.808298
27   18.568277   73.811388   18.569131   73.809843    18.569416
73.809714
28   18.567219   73.811560   18.568968   73.811216    18.567707
73.811646
29   18.566405   73.811774   18.567789   73.811474    18.565429
73.812375
30   18.565429   73.812160   18.566609   73.811731    18.562988
73.813319
31   18.561605   73.814092   18.565470   73.812160    18.560791
73.814220

    Unnamed: 12  Unnamed: 13  Unnamed: 14  Unnamed: 15  Unnamed: 1
6  \
0           NaN          NaN          NaN          NaN          Na
N
1     18.592806    73.819714    18.592725    73.819263    18.59278
6
2     18.592114    73.818898    18.592765    73.819778    18.59203
3
3     18.591626    73.818083    18.592236    73.819864    18.59085
3
4     18.591138    73.817225    18.591545    73.819714    18.58977
6
5     18.590243    73.816924    18.590548    73.819563    18.58871
8
6     18.589348    73.816795    18.589226    73.819864    18.58766
0
7     18.588454    73.816667    18.587721    73.820508    18.58660
3
8     18.587396    73.816409    18.586216    73.821151    18.58583
0
9     18.586338    73.816237    18.585138    73.822289    18.58499
6
10    18.585281    73.816066    18.583999    73.823576    18.58408
1
11    18.584223    73.815894    18.582921    73.824542    18.58316
6
```

12	18.583166	73.815722	18.582230	73.824112	18.582596
13	18.582108	73.815551	18.581823	73.822589	18.582230
14	18.581050	73.815594	18.581213	73.821473	18.581823
15	18.579993	73.815637	18.580481	73.820357	18.581132
16	18.578935	73.815894	18.579565	73.819585	18.580359
17	18.577755	73.815765	18.578264	73.819520	18.579301
18	18.576454	73.815465	18.576759	73.819692	18.577755
19	18.575233	73.815165	18.575843	73.818705	18.576209
20	18.575559	73.813620	18.575599	73.816988	18.575701
21	18.576047	73.812075	18.575477	73.814006	18.575498
22	18.575640	73.810616	18.575925	73.811302	18.575884
23	18.575274	73.808942	18.575355	73.809114	18.576006
24	18.574582	73.808084	18.574521	73.808084	18.575416
25	18.573891	73.807225	18.573789	73.806968	18.575009
26	18.572711	73.807096	18.573708	73.806496	18.574175
27	18.571247	73.807955	18.572874	73.806860	18.573240
28	18.569701	73.809543	18.571857	73.807676	18.571674
29	18.568033	73.811345	18.570840	73.808491	18.569782
30	18.565022	73.812418	18.569619	73.809478	18.568785
31	18.561239	73.814135	18.568928	73.810852	18.566243

	Unnamed: 17	Unnamed: 18	Unnamed: 19
0	NaN	NaN	NaN
1	73.819349	18.567260	73.815680
2	73.819714	18.592887	73.819370
3	73.819563	18.591748	73.819799
4	73.819671	18.590528	73.819542
5	73.820014	18.589348	73.819971
6	73.820465	18.587965	73.820357
7	73.820980	18.586949	73.820958
8	73.821666	18.585932	73.821688
9	73.822546	18.584915	73.822546
10	73.823404	18.583898	73.823447

```
11    73.824263    18.582881    73.824348
12    73.823876    18.582149    73.823190
13    73.824112    18.581457    73.821902
14    73.822653    18.580643    73.820443
15    73.821473    18.579301    73.819499
16    73.820314    18.578081    73.819714
17    73.819542    18.576738    73.819799
18    73.819628    18.575925    73.818941
19    73.819735    18.575518    73.817225
20    73.817697    18.575437    73.815207
21    73.815207    18.575884    73.812976
22    73.812847    18.575681    73.810787
23    73.811388    18.575315    73.808770
24    73.809929    18.574338    73.807826
25    73.808470    18.573525    73.806453
26    73.807890    18.572548    73.807311
27    73.806946    18.571043    73.808041
28    73.807998    18.569823    73.809071
29    73.809371    18.569294    73.810530
30    73.811452    18.568562    73.811431
31    73.812075    18.566202    73.812032
```

In [76]:
```
Data = Data_Fram[1:]
Data.columns = ["a","b","c","d","e","f","g","h","i","j","k","l","m"
,"n","o","p","q","r","s","t"]
print(Data)
```

```
            a          b          c          d          e
f  \
1   18.592725  73.819842  18.592765  73.819585  18.592752  73.8195
71
2   18.592074  73.818984  18.592074  73.818727  18.592142  73.8188
41
3   18.591586  73.817954  18.591260  73.817353  18.591735  73.8181
11
4   18.591098  73.816838  18.590243  73.816752  18.591165  73.8173
39
5   18.589755  73.816710  18.588860  73.816667  18.590108  73.8170
81
6   18.588128  73.816581  18.587518  73.816366  18.588033  73.8166
52
7   18.586176  73.816237  18.586013  73.816366  18.585959  73.8162
23
8   18.584020  73.815894  18.584549  73.815894  18.584088  73.8158
37
9   18.581864  73.815551  18.583084  73.815722  18.582176  73.8156
22
10  18.579993  73.815894  18.581742  73.815594  18.580589  73.8157
51
11  18.577877  73.815937  18.580237  73.815937  18.579043  73.8160
09
12  18.575843  73.815637  18.578691  73.816109  18.577538  73.8157
94
```

```
13  18.575437  73.814178  18.577348  73.815680  18.576033  73.8154
94
14  18.575884  73.812718  18.576047  73.815508  18.575464  73.8145
07
15  18.575681  73.811259  18.575396  73.814349  18.575748  73.8132
62
16  18.575437  73.809800  18.576006  73.812633  18.576115  73.8117
60
17  18.574908  73.808470  18.575843  73.811173  18.575504  73.8099
58
18  18.574053  73.807440  18.575437  73.809543  18.574894  73.8081
98
19  18.573484  73.806581  18.574786  73.808169  18.573714  73.8067
82
20  18.572589  73.807096  18.573891  73.807182  18.572819  73.8068
25
21  18.571694  73.807611  18.573077  73.806882  18.571925  73.8076
40
22  18.570799  73.808126  18.572264  73.807783  18.570745  73.8084
55
23  18.569904  73.808641  18.571124  73.808298  18.569484  73.8095
71
24  18.569497  73.809714  18.569985  73.809028  18.568914  73.8109
87
25  18.569131  73.810787  18.569131  73.810401  18.568019  73.8114
17
26  18.568155  73.811517  18.568155  73.811173  18.566758  73.8118
03
27  18.566690  73.811731  18.566934  73.811688  18.565416  73.8123
61
28  18.565388  73.812160  18.565632  73.812203  18.564073  73.8129
19
29  18.564168  73.812718  18.564331  73.812718  18.562730  73.8134
77
30  18.562947  73.813276  18.563029  73.813233  18.561388  73.8140
34
31  18.561727  73.813834  18.561727  73.813748  18.560208  73.8148
50

           g          h          i          j          k
l  \
1   18.592806  73.819499  18.592887  73.819542  18.592887  73.8195
85
2   18.591952  73.818512  18.592236  73.818855  18.592236  73.8188
12
3   18.591301  73.817482  18.591586  73.818083  18.591586  73.8180
40
4   18.590284  73.816838  18.590935  73.816967  18.591057  73.8172
25
5   18.589064  73.816710  18.589959  73.816795  18.589999  73.8167
95
6   18.587518  73.816452  18.588982  73.816624  18.588820  73.8166
67
```

```
7   18.585688  73.816195  18.587518  73.816495  18.587640  73.8165
38
8   18.583857  73.815594  18.585972  73.816152  18.586460  73.8164
09
9   18.582311  73.815336  18.584304  73.815808  18.585281  73.8160
66
10  18.580766  73.815551  18.582474  73.815551  18.584101  73.8157
65
11  18.579220  73.816066  18.580643  73.815637  18.582921  73.8156
37
12  18.577715  73.815765  18.579138  73.816023  18.581742  73.8155
08
13  18.576169  73.815465  18.577837  73.815765  18.580440  73.8157
22
14  18.575192  73.814650  18.576535  73.815508  18.579138  73.8161
52
15  18.575965  73.813148  18.575233  73.815250  18.577837  73.8158
08
16  18.576250  73.811603  18.575518  73.814092  18.576494  73.8155
51
17  18.575681  73.810101  18.576087  73.812933  18.575274  73.8153
79
18  18.575111  73.808599  18.575925  73.811603  18.575396  73.8142
20
19  18.574257  73.807611  18.575599  73.810229  18.575965  73.8130
19
20  18.573403  73.806667  18.575274  73.808856  18.576128  73.8117
31
21  18.572589  73.807182  18.574582  73.808084  18.575762  73.8103
58
22  18.571775  73.807697  18.573850  73.807397  18.575355  73.8088
99
23  18.570962  73.808212  18.573118  73.806624  18.574623  73.8079
12
24  18.570148  73.808727  18.572060  73.807569  18.573809  73.8069
25
25  18.569334  73.809242  18.571002  73.808341  18.572792  73.8070
11
26  18.569131  73.810616  18.569945  73.808942  18.571124  73.8082
98
27  18.568277  73.811388  18.569131  73.809843  18.569416  73.8097
14
28  18.567219  73.811560  18.568968  73.811216  18.567707  73.8116
46
29  18.566405  73.811774  18.567789  73.811474  18.565429  73.8123
75
30  18.565429  73.812160  18.566609  73.811731  18.562988  73.8133
19
31  18.561605  73.814092  18.565470  73.812160  18.560791  73.8142
20

            m          n          o          p          q
r  \
```

1	18.592806	73.819714	18.592725	73.819263	18.592786	73.8193
49						
2	18.592114	73.818898	18.592765	73.819778	18.592033	73.8197
14						
3	18.591626	73.818083	18.592236	73.819864	18.590853	73.8195
63						
4	18.591138	73.817225	18.591545	73.819714	18.589776	73.8196
71						
5	18.590243	73.816924	18.590548	73.819563	18.588718	73.8200
14						
6	18.589348	73.816795	18.589226	73.819864	18.587660	73.8204
65						
7	18.588454	73.816667	18.587721	73.820508	18.586603	73.8209
80						
8	18.587396	73.816409	18.586216	73.821151	18.585830	73.8216
66						
9	18.586338	73.816237	18.585138	73.822289	18.584996	73.8225
46						
10	18.585281	73.816066	18.583999	73.823576	18.584081	73.8234
04						
11	18.584223	73.815894	18.582921	73.824542	18.583166	73.8242
63						
12	18.583166	73.815722	18.582230	73.824112	18.582596	73.8238
76						
13	18.582108	73.815551	18.581823	73.822589	18.582230	73.8241
12						
14	18.581050	73.815594	18.581213	73.821473	18.581823	73.8226
53						
15	18.579993	73.815637	18.580481	73.820357	18.581132	73.8214
73						
16	18.578935	73.815894	18.579565	73.819585	18.580359	73.8203
14						
17	18.577755	73.815765	18.578264	73.819520	18.579301	73.8195
42						
18	18.576454	73.815465	18.576759	73.819692	18.577755	73.8196
28						
19	18.575233	73.815165	18.575843	73.818705	18.576209	73.8197
35						
20	18.575559	73.813620	18.575599	73.816988	18.575701	73.8176
97						
21	18.576047	73.812075	18.575477	73.814006	18.575498	73.8152
07						
22	18.575640	73.810616	18.575925	73.811302	18.575884	73.8128
47						
23	18.575274	73.808942	18.575355	73.809114	18.576006	73.8113
88						
24	18.574582	73.808084	18.574521	73.808084	18.575416	73.8099
29						
25	18.573891	73.807225	18.573789	73.806968	18.575009	73.8084
70						
26	18.572711	73.807096	18.573708	73.806496	18.574175	73.8078
90						
27	18.571247	73.807955	18.572874	73.806860	18.573240	73.8069

```
46
28  18.569701  73.809543  18.571857  73.807676  18.571674  73.8079
98
29  18.568033  73.811345  18.570840  73.808491  18.569782  73.8093
71
30  18.565022  73.812418  18.569619  73.809478  18.568785  73.8114
52
31  18.561239  73.814135  18.568928  73.810852  18.566243  73.8120
75
```

	s	t
1	18.567260	73.815680
2	18.592887	73.819370
3	18.591748	73.819799
4	18.590528	73.819542
5	18.589348	73.819971
6	18.587965	73.820357
7	18.586949	73.820958
8	18.585932	73.821688
9	18.584915	73.822546
10	18.583898	73.823447
11	18.582881	73.824348
12	18.582149	73.823190
13	18.581457	73.821902
14	18.580643	73.820443
15	18.579301	73.819499
16	18.578081	73.819714
17	18.576738	73.819799
18	18.575925	73.818941
19	18.575518	73.817225
20	18.575437	73.815207
21	18.575884	73.812976
22	18.575681	73.810787
23	18.575315	73.808770
24	18.574338	73.807826
25	18.573525	73.806453
26	18.572548	73.807311
27	18.571043	73.808041
28	18.569823	73.809071
29	18.569294	73.810530
30	18.568562	73.811431
31	18.566202	73.812032

Now the data-frame is converted to a list within list's and each list identify some coordinate on map

As the successive lists are generated horizontally among first row in our data frame ie there are 20 columns and each two column of first row gives us an coordinate on map that means we have 10 coordinates among which we have first 7 are from one path and other 3 are from deviated path, so that's how the below list is generated

In [77]:
```
from sklearn import svm
import numpy as np
x = Data.values.tolist()
initial_data = []
final_data = []
for row in x:
    initial_data.append([row[i:i+2] for i  in range(0, len(row), 2)
])
for y in initial_data:
    for z in y:
        final_data.append(z)
print(final_data)
```

```
[[18.592725, 73.819842], [18.592765, 73.819585], [18.592752, 73.81
9571], [18.592806, 73.819499], [18.592887, 73.819542], [18.592887,
73.819585], [18.592806, 73.819714], [18.592725, 73.819263], [18.59
2786, 73.819349], [18.56726, 73.81568], [18.592074, 73.818984], [1
8.592074, 73.818727], [18.592142, 73.818841], [18.591952, 73.81851
2], [18.592236, 73.818855], [18.592236, 73.818812], [18.592114, 73
.818898], [18.592765, 73.819778], [18.592033, 73.819714], [18.5928
87, 73.81937], [18.591586, 73.817954], [18.59126, 73.817353], [18.
591735, 73.818111], [18.591301, 73.817482], [18.591586, 73.818083]
, [18.591586, 73.81804], [18.591626, 73.818083], [18.592236, 73.81
9864], [18.590853, 73.819563], [18.591748, 73.819799], [18.591098,
73.816838], [18.590243, 73.816752], [18.591165, 73.817339], [18.59
0284, 73.816838], [18.590935, 73.816967], [18.591057, 73.817225],
[18.591138, 73.817225], [18.591545, 73.819714], [18.589776, 73.819
671], [18.590528, 73.819542], [18.589755, 73.81671], [18.58886, 73
.816667], [18.590108, 73.817081], [18.589064, 73.81671], [18.58995
9, 73.816795], [18.589999, 73.816795], [18.590243, 73.816924], [18
.590548, 73.819563], [18.588718, 73.820014], [18.589348, 73.819971
], [18.588128, 73.816581], [18.587518, 73.816366], [18.588033, 73.
816652], [18.587518, 73.816452], [18.588982, 73.816624], [18.58882
, 73.816667], [18.589348, 73.816795], [18.589226, 73.819864], [18.
```

58766, 73.820465], [18.587965, 73.820357], [18.586176, 73.816237],
[18.586013, 73.816366], [18.585959, 73.816223], [18.585688, 73.816
195], [18.587518, 73.816495], [18.58764, 73.816538], [18.588454, 7
3.816667], [18.587721, 73.820508], [18.586603, 73.82098], [18.5869
49, 73.820958], [18.58402, 73.815894], [18.584549, 73.815894], [18
.584088, 73.815837], [18.583857, 73.815594], [18.585972, 73.816152
], [18.58646, 73.816409], [18.587396, 73.816409], [18.586216, 73.8
21151], [18.58583, 73.821666], [18.585932, 73.821688], [18.581864,
73.815551], [18.583084, 73.815722], [18.582176, 73.815622], [18.58
2311, 73.815336], [18.584304, 73.815808], [18.585281, 73.816066],
[18.586338, 73.816237], [18.585138, 73.822289], [18.584996, 73.822
546], [18.584915, 73.822546], [18.579993, 73.815894], [18.581742,
73.815594], [18.580589, 73.815751], [18.580766, 73.815551], [18.58
2474, 73.815551], [18.584101, 73.815765], [18.585281, 73.816066],
[18.583999, 73.823576], [18.584081, 73.823404], [18.583898, 73.823
447], [18.577877, 73.815937], [18.580237, 73.815937], [18.579043,
73.816009], [18.57922, 73.816066], [18.580643, 73.815637], [18.582
921, 73.815637], [18.584223, 73.815894], [18.582921, 73.824542], [
18.583166, 73.824263], [18.582881, 73.824348], [18.575843, 73.8156
37], [18.578691, 73.816109], [18.577538, 73.815794], [18.577715, 7
3.815765], [18.579138, 73.816023], [18.581742, 73.815508], [18.583
166, 73.815722], [18.58223, 73.824112], [18.582596, 73.823876], [1
8.582149, 73.82319], [18.575437, 73.814178], [18.577348, 73.81568]
, [18.576033, 73.815494], [18.576169, 73.815465], [18.577837, 73.8
15765], [18.58044, 73.815722], [18.582108, 73.815551], [18.581823,
73.822589], [18.58223, 73.824112], [18.581457, 73.821902], [18.575
884, 73.812718], [18.576047, 73.815508], [18.575464, 73.814507], [
18.575192, 73.81465], [18.576535, 73.815508], [18.579138, 73.81615
2], [18.58105, 73.815594], [18.581213, 73.821473], [18.581823, 73.
822653], [18.580643, 73.820443], [18.575681, 73.811259], [18.57539
6, 73.814349], [18.575748, 73.813262], [18.575965, 73.813148], [18
.575233, 73.81525], [18.577837, 73.815808], [18.579993, 73.815637]
, [18.580481, 73.820357], [18.581132, 73.821473], [18.579301, 73.8
19499], [18.575437, 73.8098], [18.576006, 73.812633], [18.576115,
73.81176], [18.57625, 73.811603], [18.575518, 73.814092], [18.5764
94, 73.815551], [18.578935, 73.815894], [18.579565, 73.819585], [1
8.580359, 73.820314], [18.578081, 73.819714], [18.574908, 73.80847
], [18.575843, 73.811173], [18.575504, 73.809958], [18.575681, 73.
810101], [18.576087, 73.812933], [18.575274, 73.815379], [18.57775
5, 73.815765], [18.578264, 73.81952], [18.579301, 73.819542], [18.
576738, 73.819799], [18.574053, 73.80744], [18.575437, 73.809543],
[18.574894, 73.808198], [18.575111, 73.808599], [18.575925, 73.811
603], [18.575396, 73.81422], [18.576454, 73.815465], [18.576759, 7
3.819692], [18.577755, 73.819628], [18.575925, 73.818941], [18.573
484, 73.806581], [18.574786, 73.808169], [18.573714, 73.806782], [
18.574257, 73.807611], [18.575599, 73.810229], [18.575965, 73.8130
19], [18.575233, 73.815165], [18.575843, 73.818705], [18.576209, 7
3.819735], [18.575518, 73.817225], [18.572589, 73.807096], [18.573
891, 73.807182], [18.572819, 73.806825], [18.573403, 73.806667], [
18.575274, 73.808856], [18.576128, 73.811731], [18.575559, 73.8136
2], [18.575599, 73.816988], [18.575701, 73.817697], [18.575437, 73
.815207], [18.571694, 73.807611], [18.573077, 73.806882], [18.5719
25, 73.80764], [18.572589, 73.807182], [18.574582, 73.808084], [18

```
.575762, 73.810358], [18.576047, 73.812075], [18.575477, 73.814006
], [18.575498, 73.815207], [18.575884, 73.812976], [18.570799, 73.
808126], [18.572264, 73.807783], [18.570745, 73.808455], [18.57177
5, 73.807697], [18.57385, 73.807397], [18.575355, 73.808899], [18.
57564, 73.810616], [18.575925, 73.811302], [18.575884, 73.812847],
[18.575681, 73.810787], [18.569904, 73.808641], [18.571124, 73.808
298], [18.569484, 73.809571], [18.570962, 73.808212], [18.573118,
73.806624], [18.574623, 73.807912], [18.575274, 73.808942], [18.57
5355, 73.809114], [18.576006, 73.811388], [18.575315, 73.80877], [
18.569497, 73.809714], [18.569985, 73.809028], [18.568914, 73.8109
87], [18.570148, 73.808727], [18.57206, 73.807569], [18.573809, 73
.806925], [18.574582, 73.808084], [18.574521, 73.808084], [18.5754
16, 73.809929], [18.574338, 73.807826], [18.569131, 73.810787], [1
8.569131, 73.810401], [18.568019, 73.811417], [18.569334, 73.80924
2], [18.571002, 73.808341], [18.572792, 73.807011], [18.573891, 73
.807225], [18.573789, 73.806968], [18.575009, 73.80847], [18.57352
5, 73.806453], [18.568155, 73.811517], [18.568155, 73.811173], [18
.566758, 73.811803], [18.569131, 73.810616], [18.569945, 73.808942
], [18.571124, 73.808298], [18.572711, 73.807096], [18.573708, 73.
806496], [18.574175, 73.80789], [18.572548, 73.807311], [18.56669,
73.811731], [18.566934, 73.811688], [18.565416, 73.812361], [18.56
8277, 73.811388], [18.569131, 73.809843], [18.569416, 73.809714],
[18.571247, 73.807955], [18.572874, 73.80686], [18.57324, 73.80694
6], [18.571043, 73.808041], [18.565388, 73.81216], [18.565632, 73.
812203], [18.564073, 73.812919], [18.567219, 73.81156], [18.568968
, 73.811216], [18.567707, 73.811646], [18.569701, 73.809543], [18.
571857, 73.807676], [18.571674, 73.807998], [18.569823, 73.809071]
, [18.564168, 73.812718], [18.564331, 73.812718], [18.56273, 73.81
3477], [18.566405, 73.811774], [18.567789, 73.811474], [18.565429,
73.812375], [18.568033, 73.811345], [18.57084, 73.808491], [18.569
782, 73.809371], [18.569294, 73.81053], [18.562947, 73.813276], [1
8.563029, 73.813233], [18.561388, 73.814034], [18.565429, 73.81216
], [18.566609, 73.811731], [18.562988, 73.813319], [18.565022, 73.
812418], [18.569619, 73.809478], [18.568785, 73.811452], [18.56856
2, 73.811431], [18.561727, 73.813834], [18.561727, 73.813748], [18
.560208, 73.81485], [18.561605, 73.814092], [18.56547, 73.81216],
[18.560791, 73.81422], [18.561239, 73.814135], [18.568928, 73.8108
52], [18.566243, 73.812075], [18.566202, 73.812032]]
```

Clustering of Data is to be done for learning process

As I mentioned above the first 7 coordinate belong to one path which is clustered to "0" cluster and next 3 belongs to cluster on "1"

So the desired output is generated using nested for loop

In [86]:
```
y = []
for val in range(int(len(final_data)/10)):
    for x in range(10):
        if(x>6):
            y.append(1)
        else:
            y.append(0)
print(y)
```

```
[0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0,
0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0,
0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0,
0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1,
1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1,
0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0,
0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0,
0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0,
0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1,
1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1,
0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0,
0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0,
0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0,
0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1,
1, 1]
```

K nearest neighbor Algorithm is used for learning and prediction

In [79]:
```
from sklearn.neighbors import KNeighborsClassifier
knn = KNeighborsClassifier()
knn.fit(final_data,y)
KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkow
ski',
           metric_params=None, n_jobs=1, n_neighbors=5, p=2,
           weights='uniform')
```

Out[79]:
```
KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minko
wski',
           metric_params=None, n_jobs=1, n_neighbors=5, p=2,
           weights='uniform')
```

As the learning is completed now we can run the test to get the desired result

In [80]:
```
import pandas as pd
Raw_Data = pd.ExcelFile("Raw_Data11.xlsx")
print(Raw_Data.sheet_names)
Data_Fram = Raw_Data.parse("Sheet 2")
print(Data_Fram)
```

```
['Sheet 1', 'Sheet 2']
      Table 1  Unnamed: 1
0   18.592752   73.819313
1   18.591491   73.819699
2   18.589986   73.819613
3   18.588806   73.820043
4   18.587626   73.820601
5   18.586447   73.821158
6   18.585552   73.821974
7   18.584494   73.822961
8   18.583437   73.824120
9   18.581850   73.822789
10  18.580020   73.820171
11  18.577986   73.819699
12  18.576318   73.819656
13  18.575789   73.818026
14  18.575382   73.816094
15  18.575504   73.813648
16  18.575952   73.811159
17  18.575423   73.809056
18  18.574203   73.807726
19  18.572494   73.807082
20  18.570826   73.808413
21  18.569443   73.810430
22  18.567612   73.811932
23  18.565375   73.812575
24  18.563137   73.813219
25  18.560900   73.814163
26  18.558744   73.815966
27  18.556587   73.817768
28  18.554431   73.819571
29  18.552275   73.821373
```

In [81]:
```
test_Data = Data_Fram[1:]
test_Data.columns = ["a","b"]
print(test_Data)
```

```
            a          b
1   18.591491  73.819699
2   18.589986  73.819613
3   18.588806  73.820043
4   18.587626  73.820601
5   18.586447  73.821158
6   18.585552  73.821974
7   18.584494  73.822961
8   18.583437  73.824120
9   18.581850  73.822789
10  18.580020  73.820171
11  18.577986  73.819699
12  18.576318  73.819656
13  18.575789  73.818026
14  18.575382  73.816094
15  18.575504  73.813648
16  18.575952  73.811159
17  18.575423  73.809056
18  18.574203  73.807726
19  18.572494  73.807082
20  18.570826  73.808413
21  18.569443  73.810430
22  18.567612  73.811932
23  18.565375  73.812575
24  18.563137  73.813219
25  18.560900  73.814163
26  18.558744  73.815966
27  18.556587  73.817768
28  18.554431  73.819571
29  18.552275  73.821373
```

In [82]:
```
from sklearn import svm
import numpy as np
x = test_Data.values.tolist()
initial_data = []
final_test_data = []
for row in x:
    initial_data.append([row[i:i+2] for i  in range(0, len(row), 2)
])
for y in initial_data:
    for z in y:
        final_test_data.append(z)
print(final_test_data)
```

```
[[18.591491, 73.819699], [18.589986, 73.819613], [18.588806, 73.82
0043], [18.587626, 73.820601], [18.586447, 73.821158], [18.585552,
73.821974], [18.584494, 73.822961], [18.583437, 73.82412], [18.581
85, 73.822789], [18.58002, 73.820171], [18.577986, 73.819699], [18
.576318, 73.819656], [18.575789, 73.818026], [18.575382, 73.816094
], [18.575504, 73.813648], [18.575952, 73.811159], [18.575423, 73.
809056], [18.574203, 73.807726], [18.572494, 73.807082], [18.57082
6, 73.808413], [18.569443, 73.81043], [18.567612, 73.811932], [18.
565375, 73.812575], [18.563137, 73.813219], [18.5609, 73.814163],
[18.558744, 73.815966], [18.556587, 73.817768], [18.554431, 73.819
571], [18.552275, 73.821373]]
```

Now the output is an array of 0 and 1 means in the above list or coordinates first 13 coordinates belong to 1 cluster ie it is traveling from the deviated path and after some point, it will come back to the first path

In [85]:
```
knn.predict(final_test_data)
```

Out[85]:
```
array([1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0,
       0, 0, 0, 0, 0, 0])
```