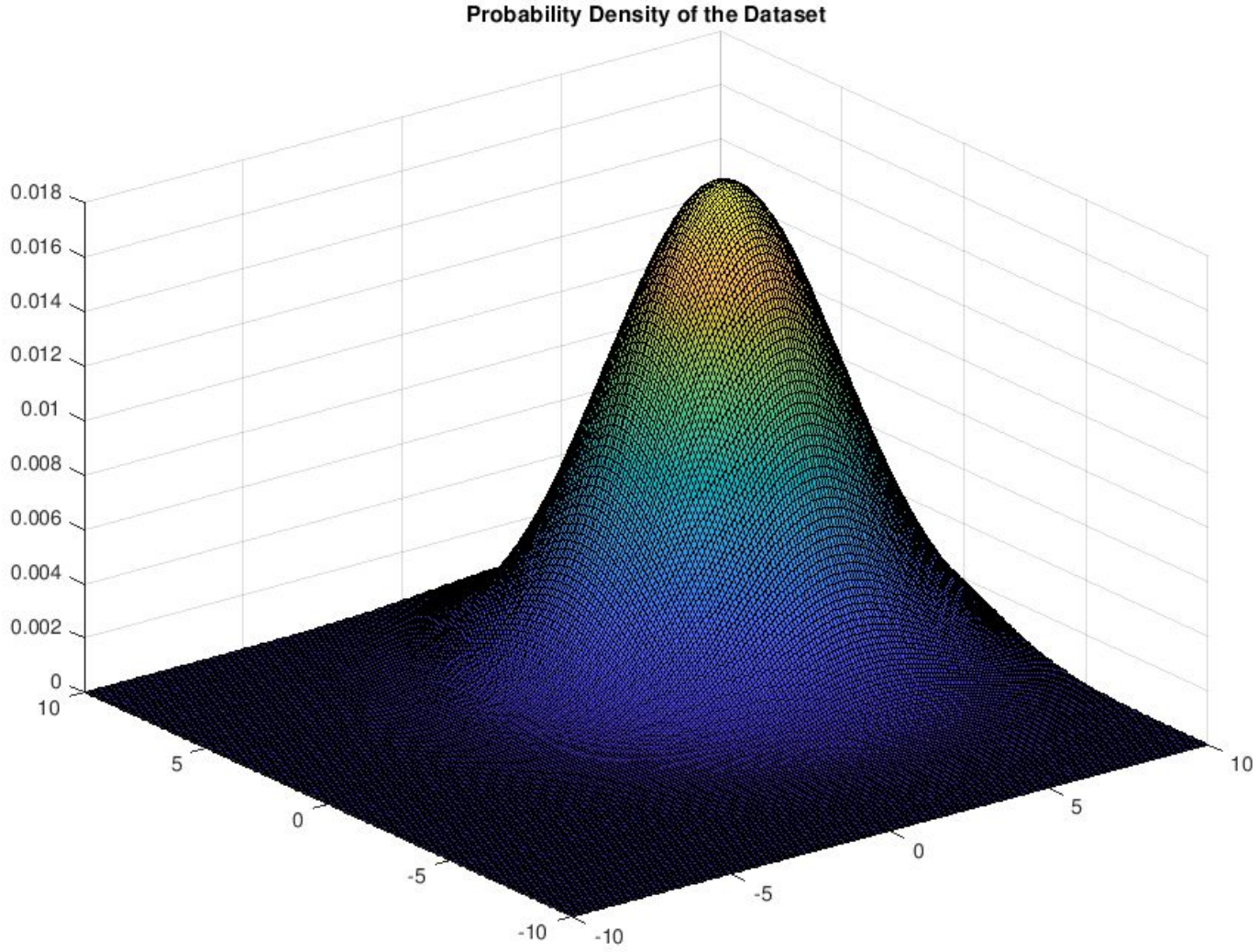
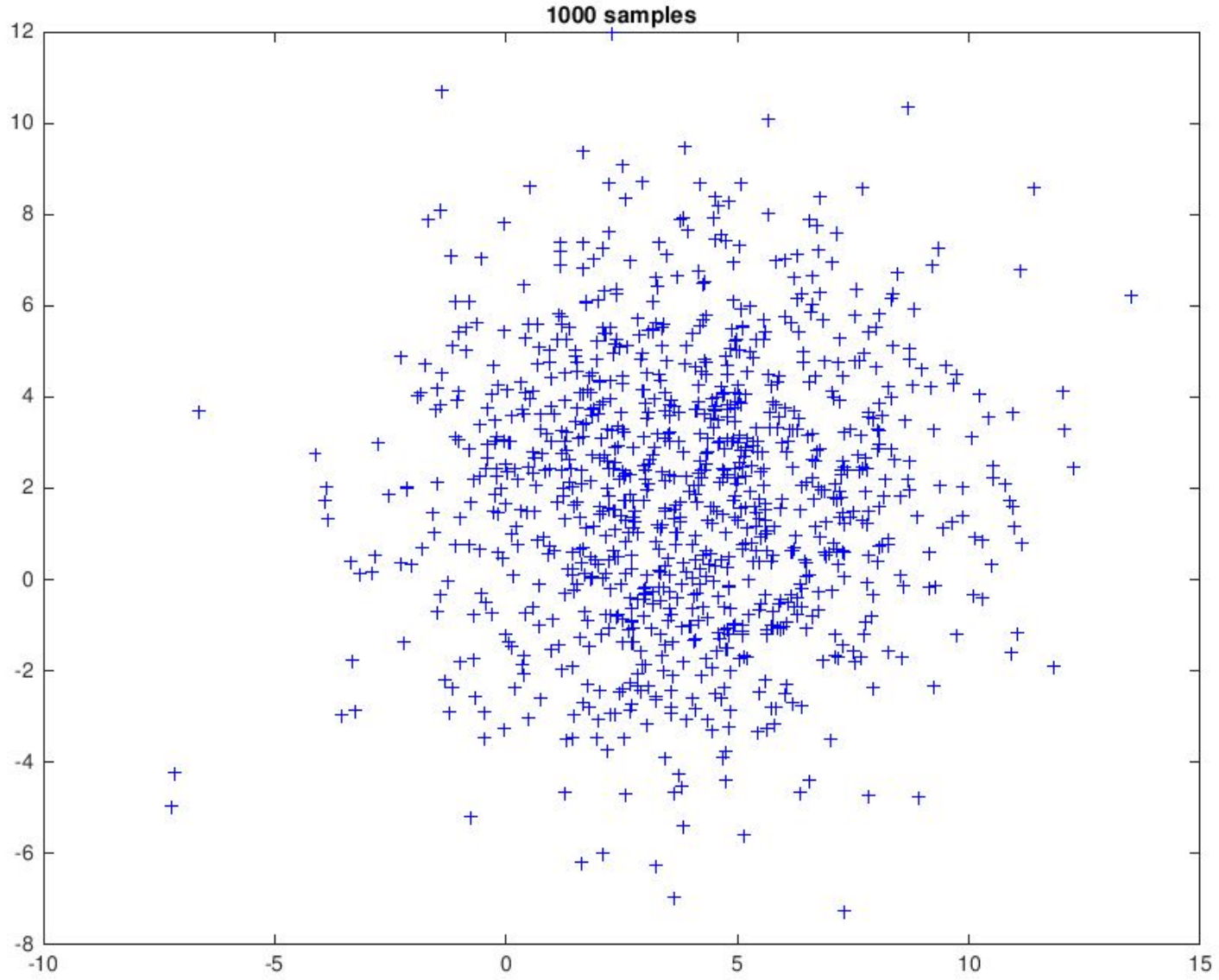
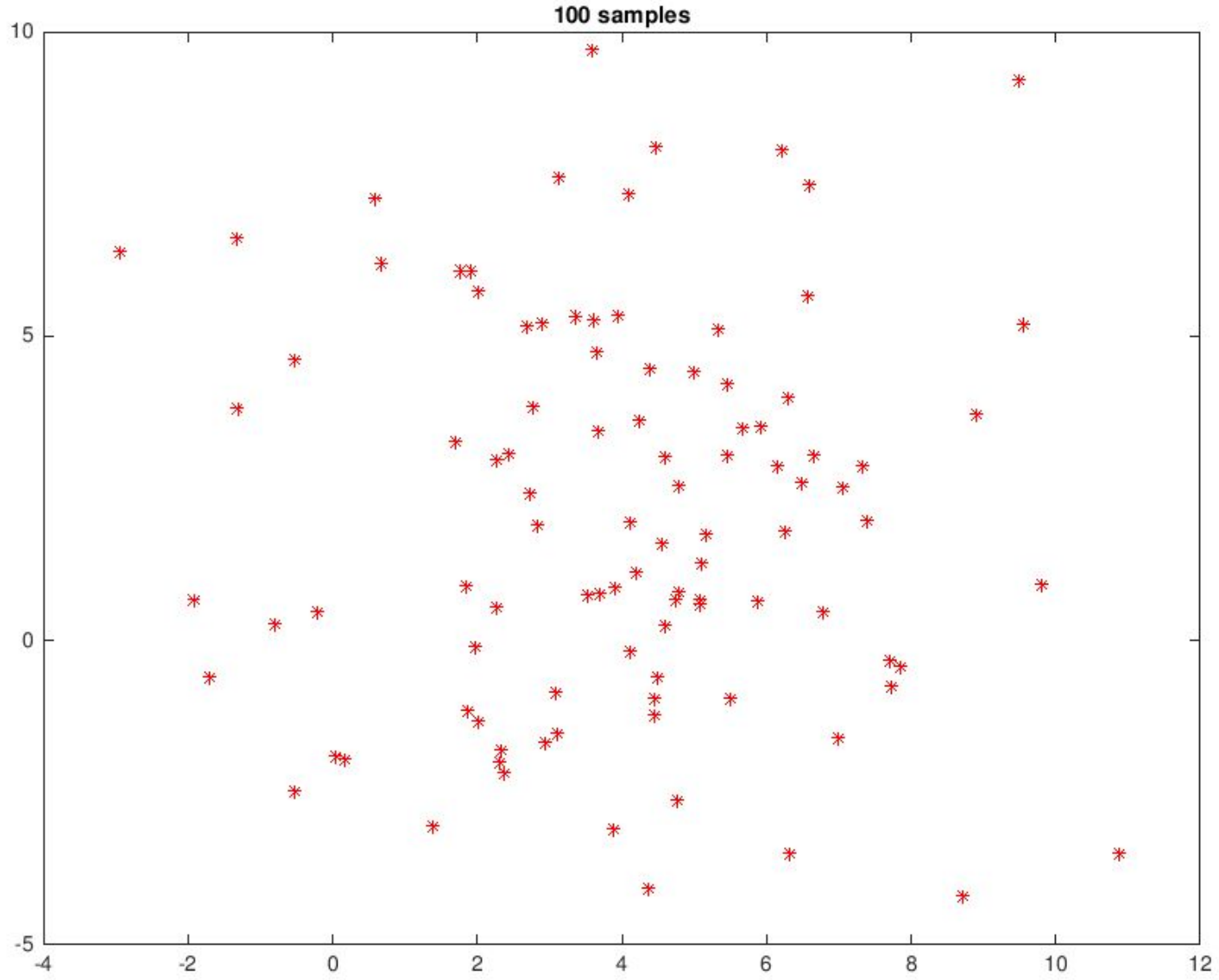
Hakkı Selim Öztürk
2013401027

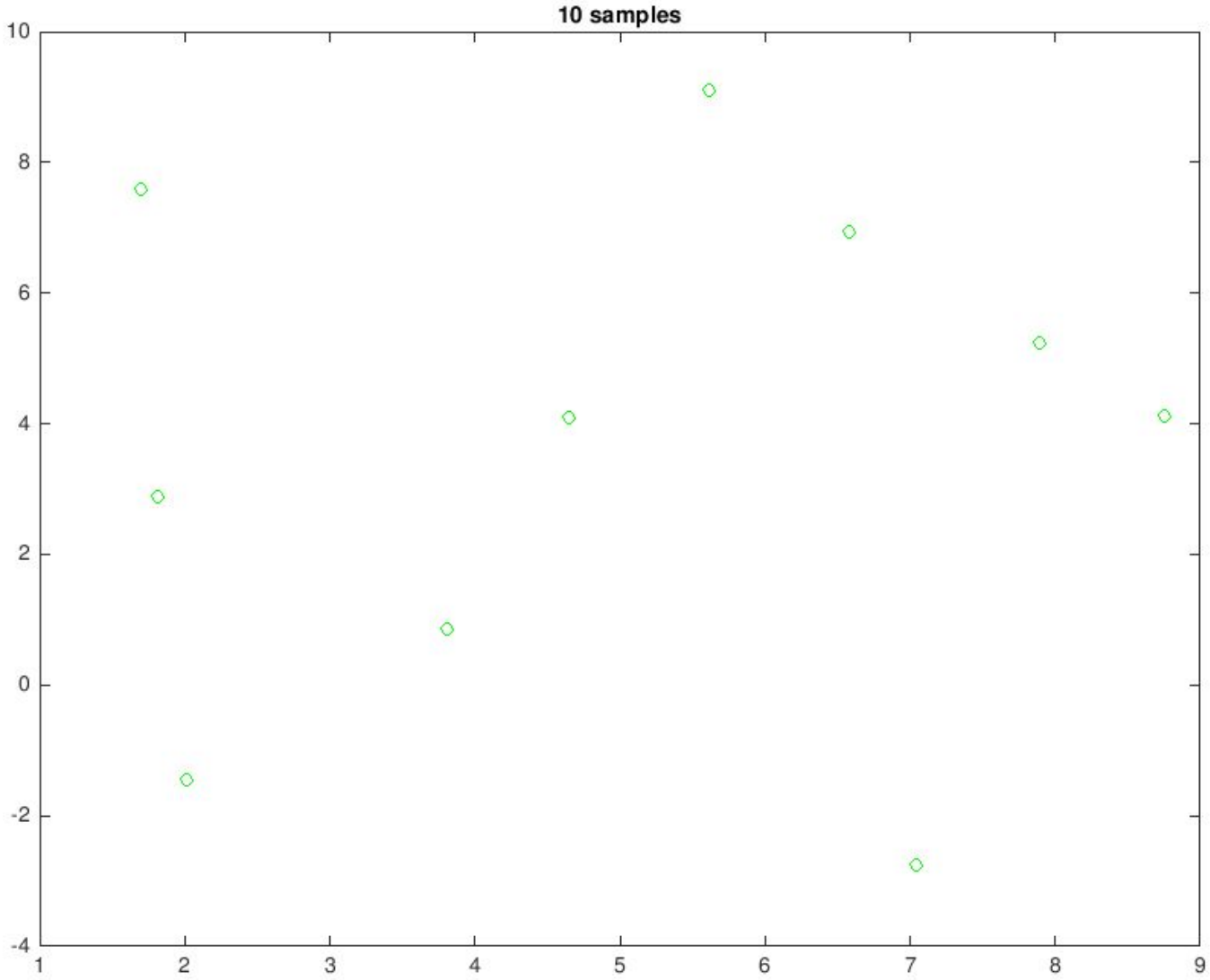
25.12.18

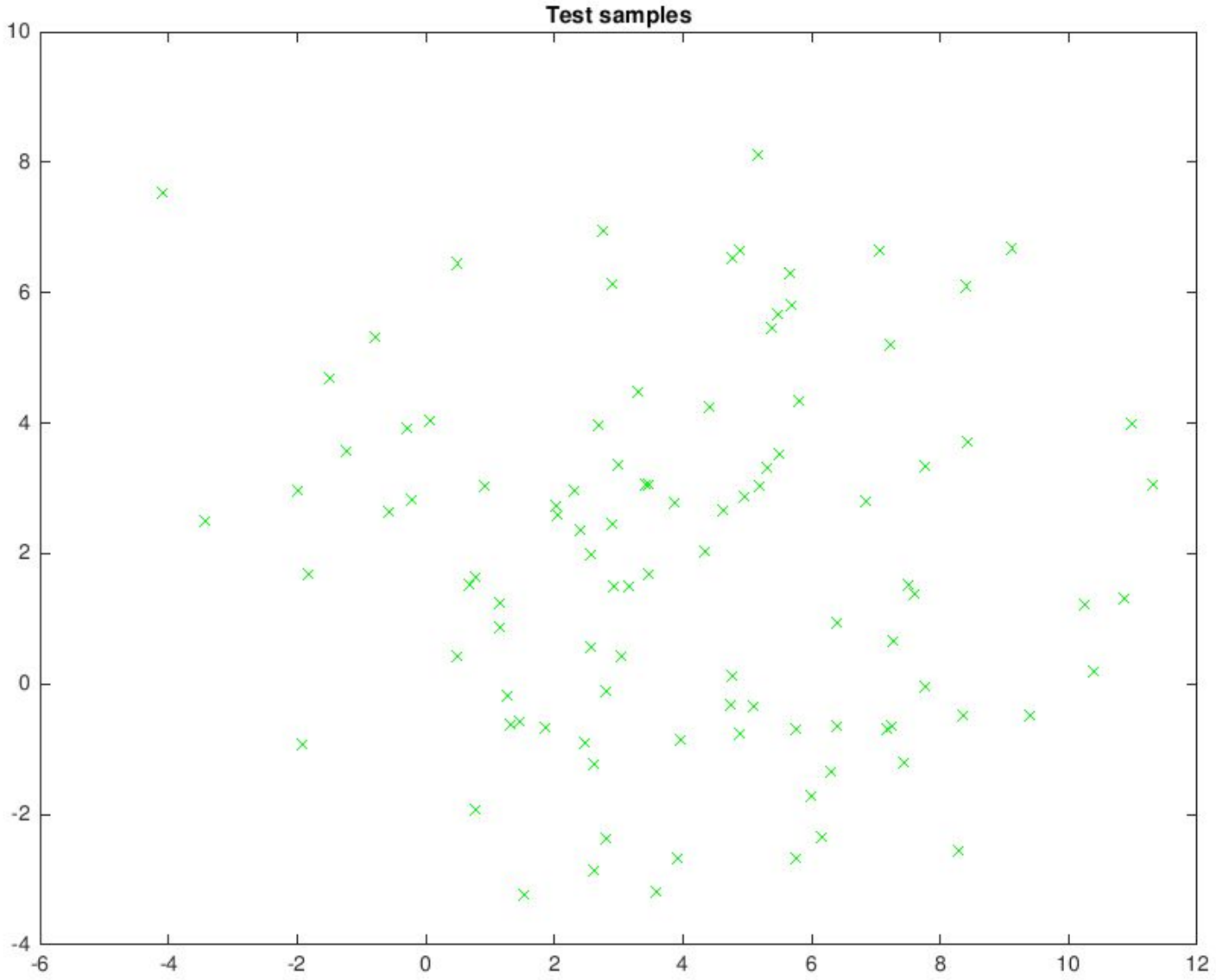
EE 573 Pattern Recognition – Project 3 - Density Estimation



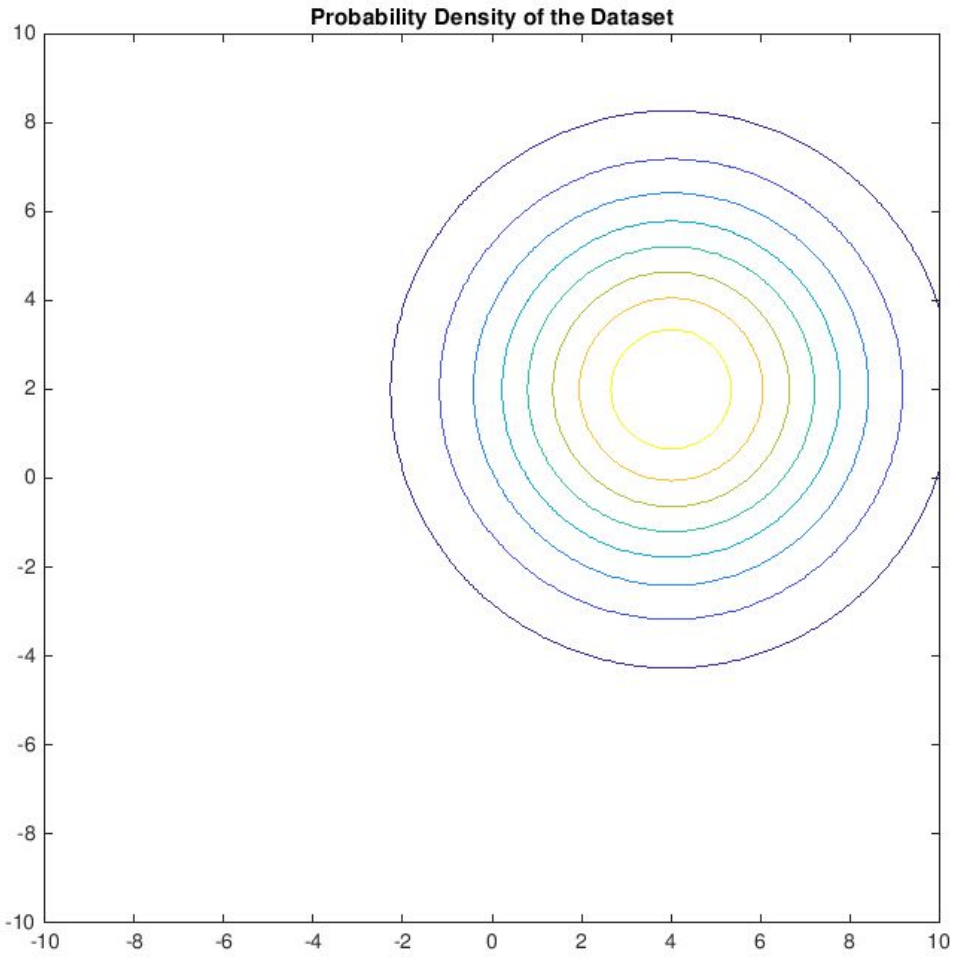

1000 samples
12
10
8
6
4
2
0
-2
-4
-6
-8
-10
-5
0
5
10
15


100 samples
10
5
0
-5
-4
-2
0
2
4
6
8
10
12

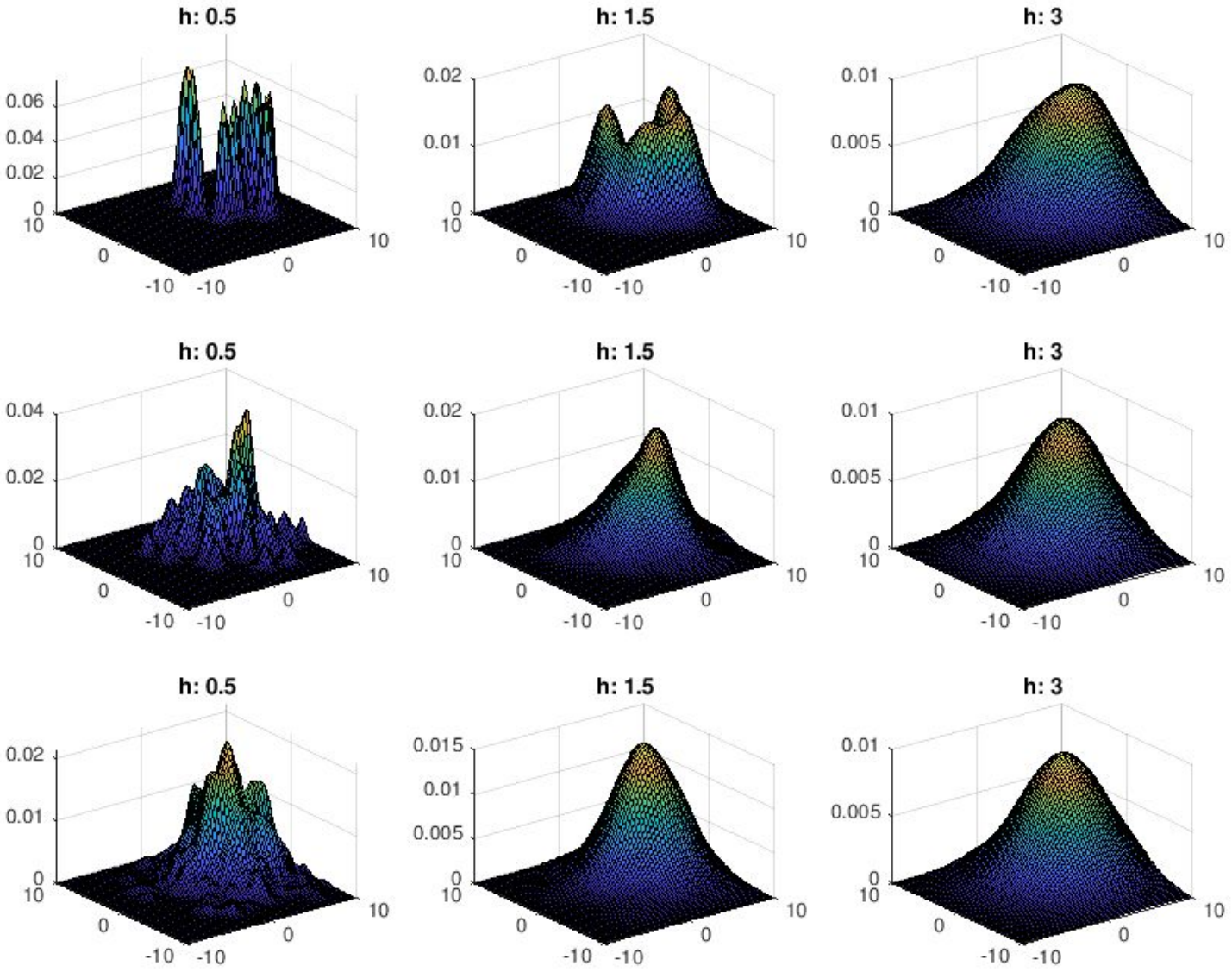

10 samples
10
8
6
4
2
0
-2
-4
1
2
3
4
5
6
7
8
9

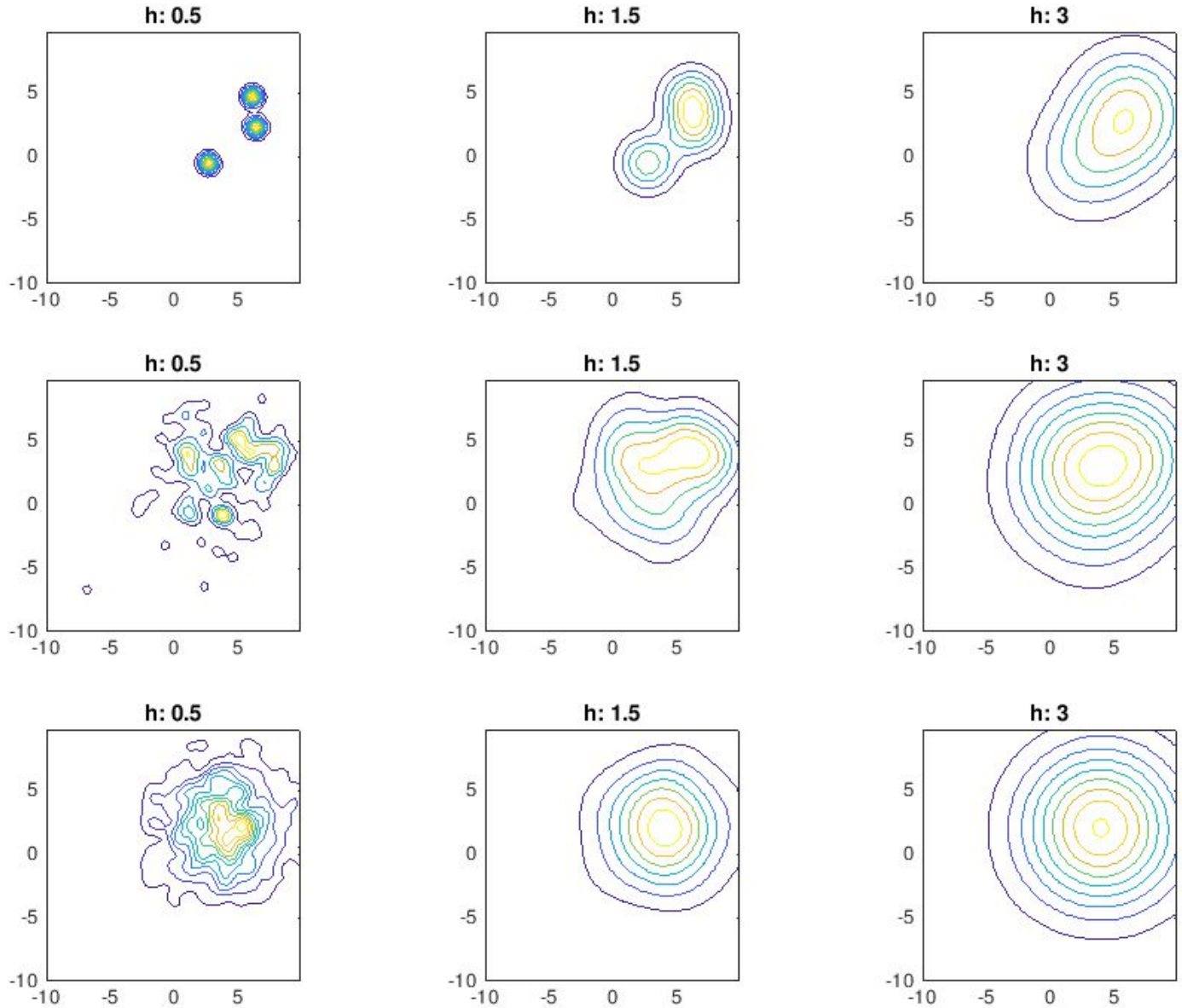

Test samples
10
8
6
4
2
0
-2
-4
-6
-4
-2
0
2
4
6
8
10
12

Probability Density for the given variables

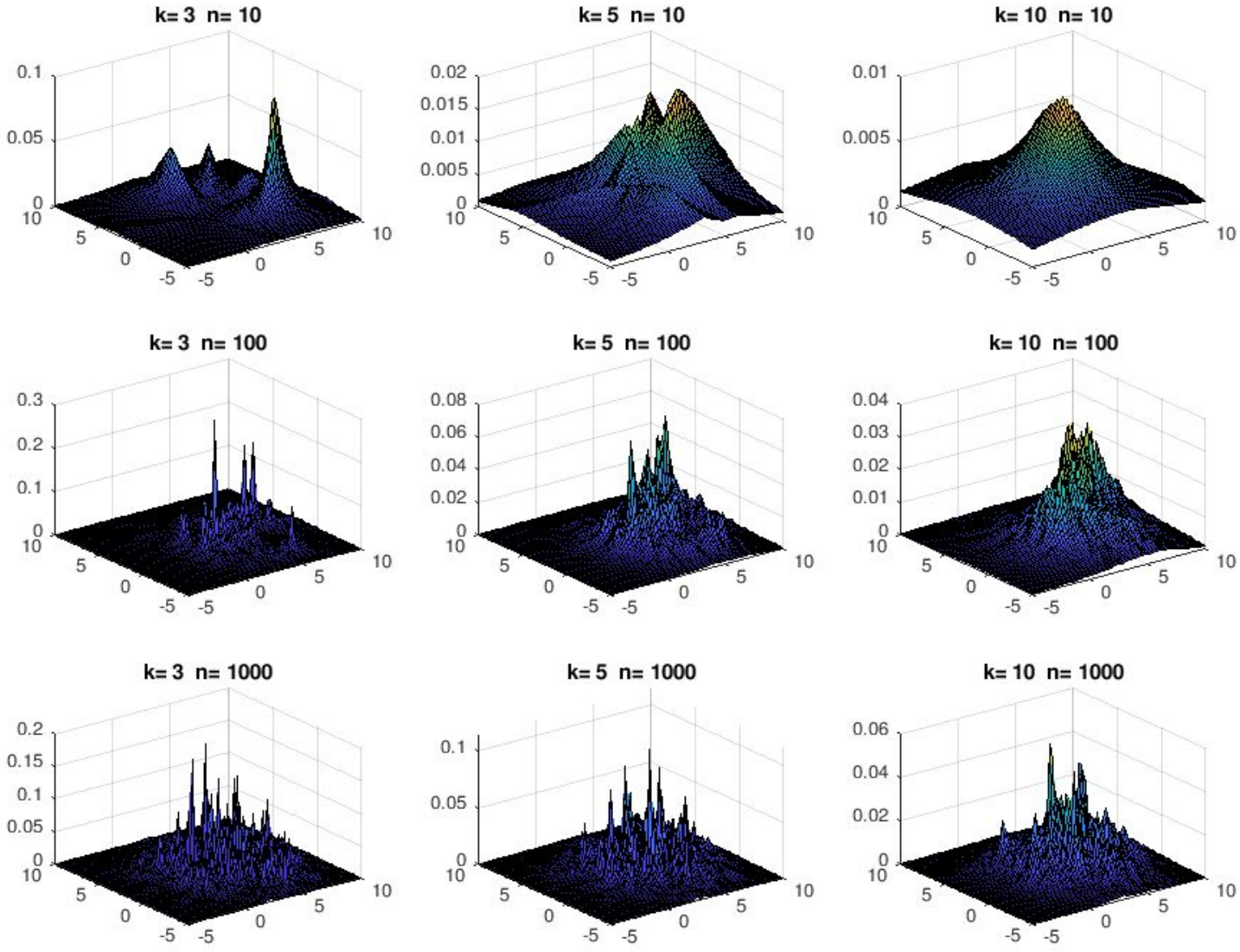


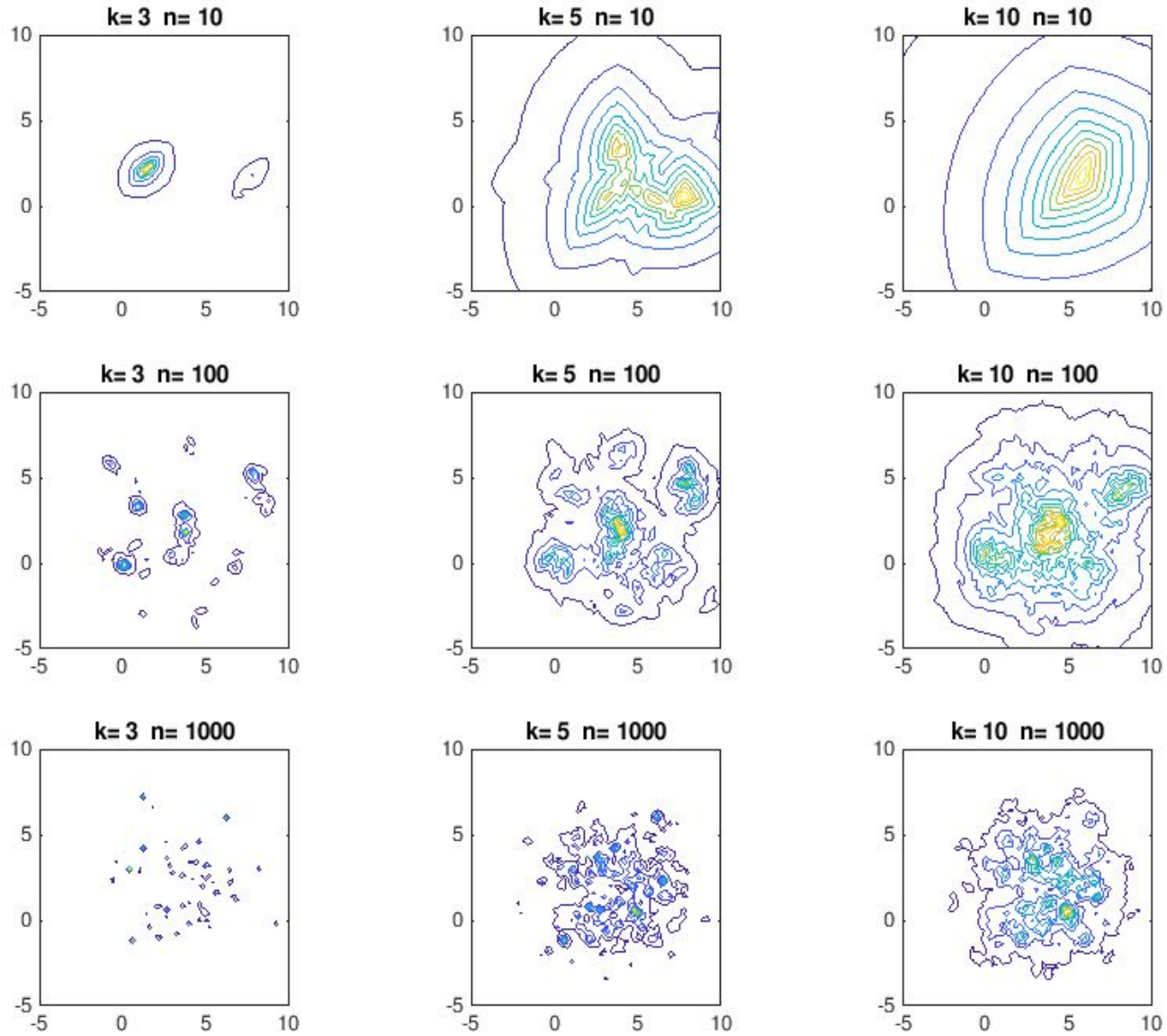
Density estimations with **Parzen**



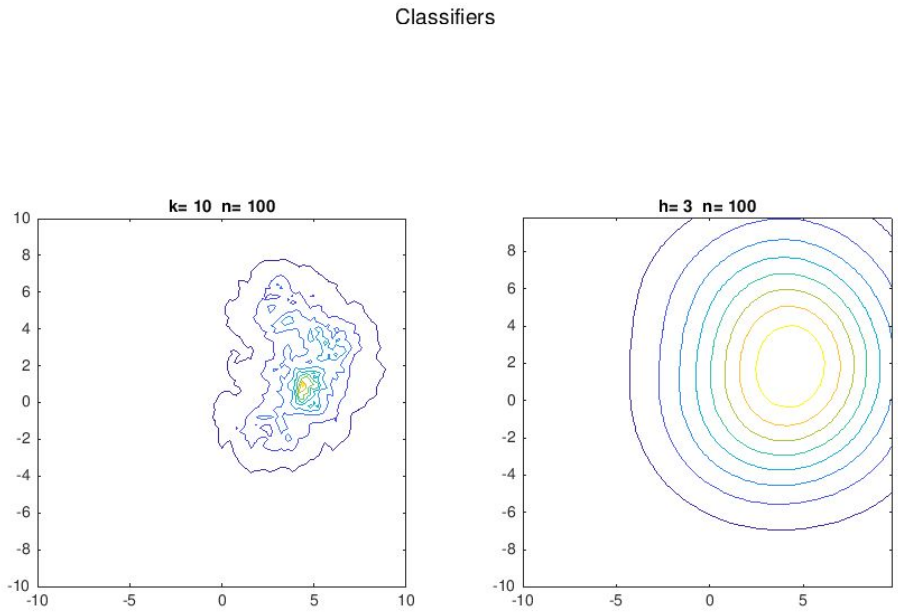

h: 0.5
h: 1.5
h: 3
h: 0.5
h: 1.5
h: 3
h: 0.5
h: 1.5
h: 3
5
0
-5
-10
-10
-5
0
5

density estimations with **kNN**




k= 3 n= 10
k= 5 n= 10
k= 10 n= 10
k= 3 n= 100
k= 5 n= 100
k= 10 n= 100
k= 3 n= 1000
k= 5 n= 1000
k= 10 n= 1000
10
5
0
-5
-5
0
5
10

Classification *n=100, k=10, h=3 is chosen.* The pdf estimates for this values plotted below.



For 100 of the test samples

Number of times parzen estimate is closer to actual probability	knn closer
58	**42**

This results seems natural because parzen pdf estimate is also closer to the actual pdf for this values.
kNN is very simple and intuitive but it needs large number of samples to perform well.
In parzen, choosing the appropriate window size h is difficult.
They can be applied to the data from any distribution, although they are both computationally heavy. To classify one point we have to compute a function which depends on all samples