HARSHINE VENKATARAMAN
1001569298
CSE5334
Homework-1
March 10,2019

CSE5334-HOMEWORK1

PROBLEM 1:

Using **np.random.multivariate_normal (Mu, Sigma, Number of samples)** We can generate 2 sets of 2-D Gaussian random data, each set containing 500 samples using given parameters below. $\mu 1 = [1, 0]$, $\mu 2 = [0, 1.5]$, $\Sigma 1 = [0.9\ 0.4\ 0.4\ 0.9]$, $\Sigma 2 = [0.9\ 0.4\ 0.4\ 0.9]$

The function **mykmeans(X, k, c). X** is the data that we have generated from above. **k** is the clusters and **c** is the initial centroid given.

Step1) The initial centroids are given
Step2) For each point in the data set X, we can determine the cluster by calculating the Euclidian distance to the centroids. Using **euclidean_dist(X, centroids, clusters)**

$$c_i^{t+1} = \frac{1}{|S_i^t|} \sum_{x_j \in S_i^t} x_j$$

Step3) Now, we update the centroids using formula
Step4) We continue step 2 and step 3 until the maximum iterations 10000 is reached(i.e mykmeans has converged).

Given Test Cases:
PROLEM 1 A)
k = 2 and initial centers c1 = (10, 10) and c2 = (−10, −10)

OUTPUT:
TOTAL NO OF DATA INSTANCES: 1000

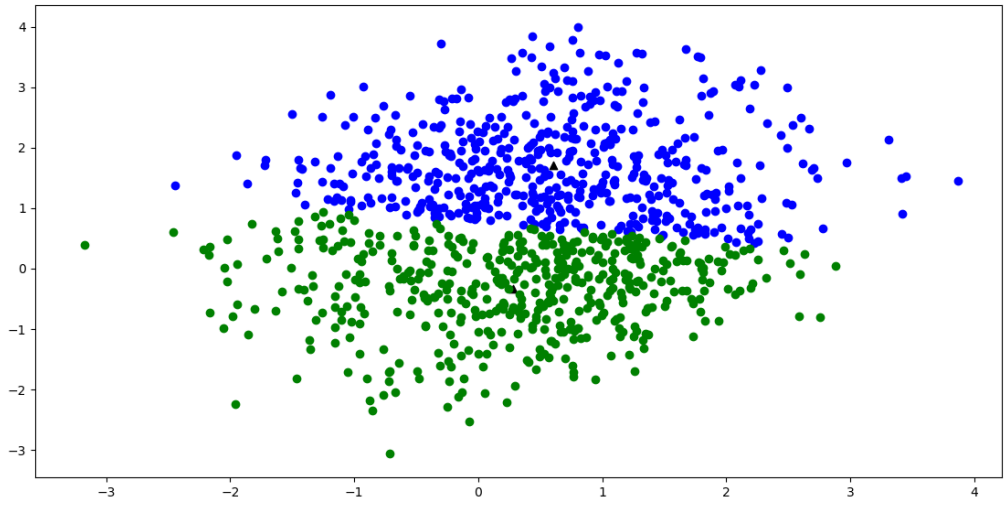
TOTAL NUMBER OF ITERATIONS: 9985

NEW CENTROIDS OF EACH CLUSTER: [[0.6046814368564711, 1.7089719420302798], [0.2783551076539984, -0.33899475578725874]]

Scatter Plot showing cluster and centroids:

The centroids are shown as black triangle in the figure below

HARSHINE VENKATARAMAN
1001569298
CSE5334
Homework-1
March 10,2019



PROBLEM1 B)

k = 4 and initial centers c1 = (10, 10) and c2 = (−10, −10), c3 = (10, −10) and c4 = (−10, 10)

OUTPUT:

TOTAL NO OF DATA INSTANCES: 1000

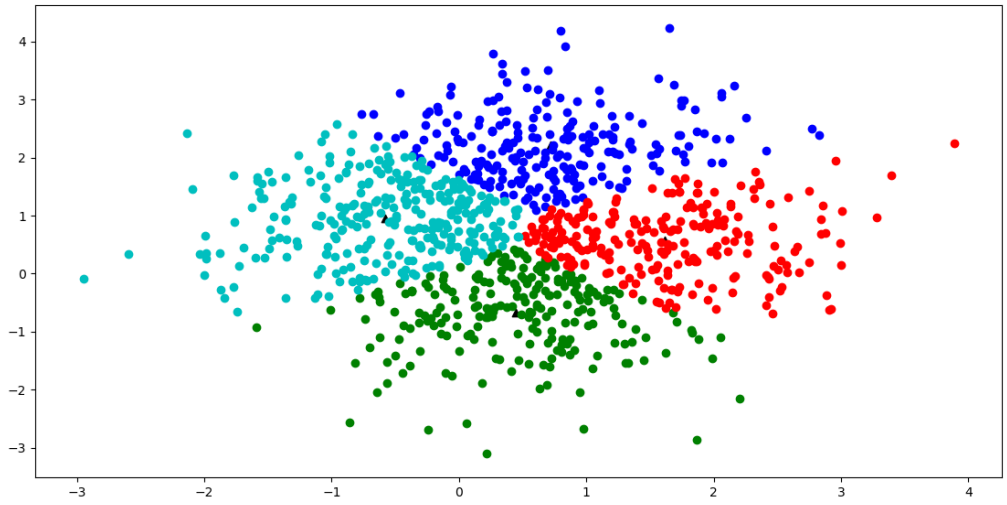
TOTAL NUMBER OF ITERATIONS: 9967

NEW CENTROIDS OF EACH CLUSTER: [[0.7190587137767331, 2.2145540442741583], [0.45004355972567184, -0.6757529425757193], [1.6180517915355, 0.5798487050531008], [-0.5806667916385254, 0.9487915947906757]]

Scatter Plot showing cluster and centroids:

The centroids are shown as black triangle in the figure below

HARSHINE VENKATARAMAN
1001569298
CSE5334
Homework-1
March 10,2019



REFERENCES:

#https://elearn.uta.edu/bbcswebdav/pid-8090434-dt-content-rid-139991592_2/courses/2192-DATA-MINING-25983-001/cse5334-s19-07_supervised_unsupervised_learning-2%282%29.pdf

 # stats.stackexchange.com

PROBLEM 2:

To generate random gaussian data we use **np.random.multivariate_normal(mu_vec, cov_mat, 500)** by passing the given mu and sigma.

Using the above function generate 2 sets of data and concatenate them.

To implement hypercube kernel for the Parzen-window estimation. Function used: **hypercubeKernel(h, x, x_i): where** h: window width; x: point x for density estimation,; x_i: point from training sample. The function returns array as input for a window function.

parzenWindowFn(x_vec, h=0.1) function implements the window function. Returns 1 if sample vector lies within a origin-centered hypercube, 0 otherwise.

$$k(u) = \begin{cases} 1 & |u_i| \leq 1/2, \quad i = 1, \cdots, d \\ 0 & o.w. \end{cases}$$

parzenEstimation(x_samples, point_x, h, d, window_func, kernel_func): function implements parzen-window estimation.

HARSHINE VENKATARAMAN
1001569298
CSE5334
Homework-1
March 10,2019

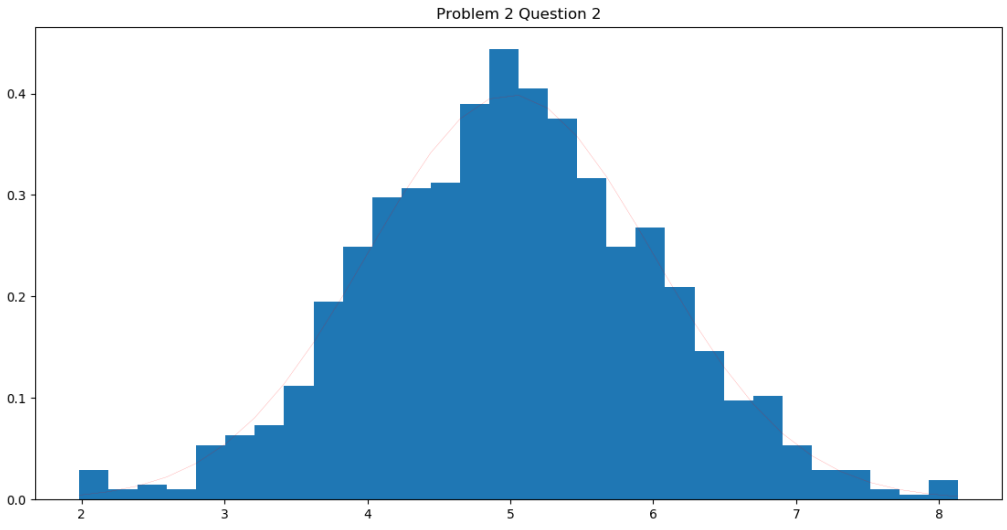
x_samples: training sample; point_x: point x for density estimation, h: window width d: dimensions; window_func: a Parzen window function (phi); kernel_function: A hypercube. The function returns the density estimate p(x).

mykde function calculates the densities.

$$p(x) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{h^d}k(\frac{x - x_i}{h})$$

Question2: Generate N = 1000 Gaussian random data with μ1 = 5 and σ1 = 1. Test your function mykde on this data with h = {.1, 1, 5, 10}. In your report, report the histogram of X along with the figures of estimated densities

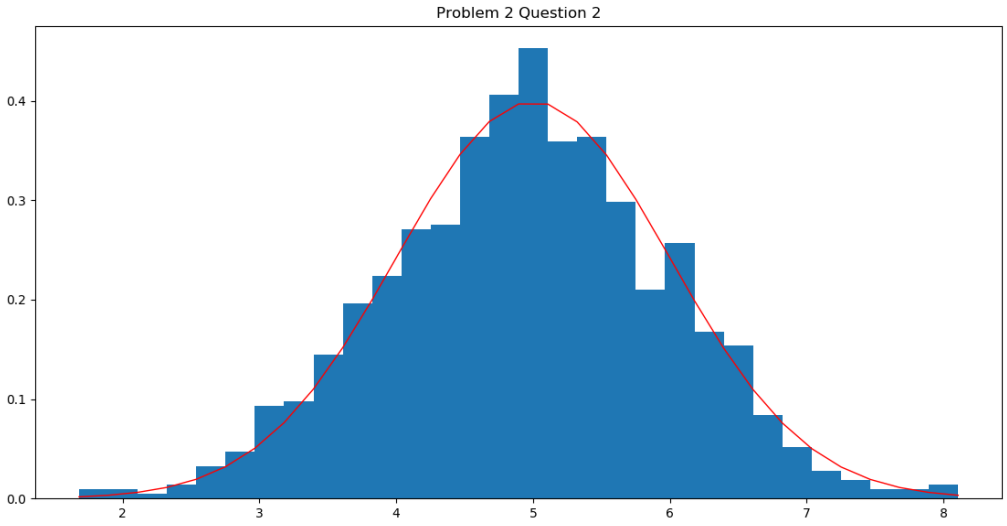
Outputs:
h =.1



---

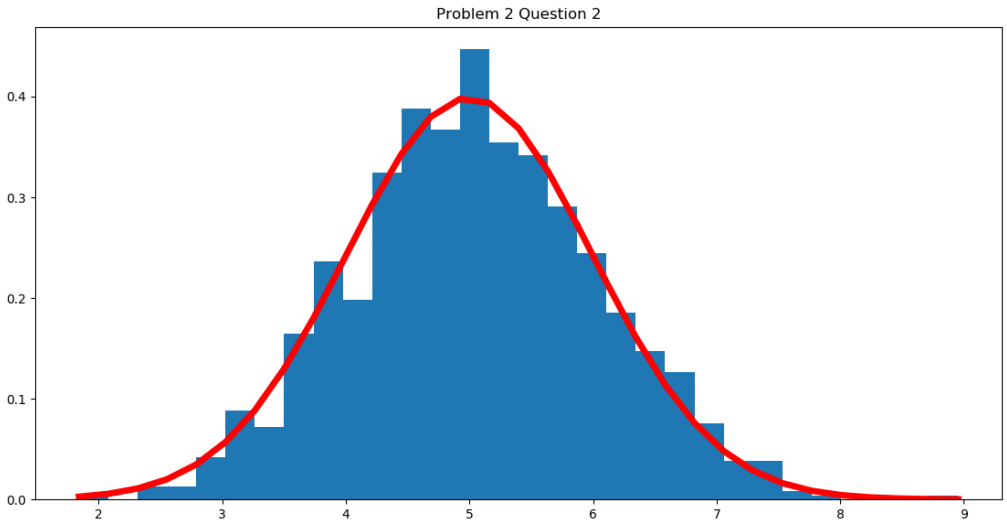
h = 1

HARSHINE VENKATARAMAN
1001569298
CSE5334
Homework-1
March 10,2019



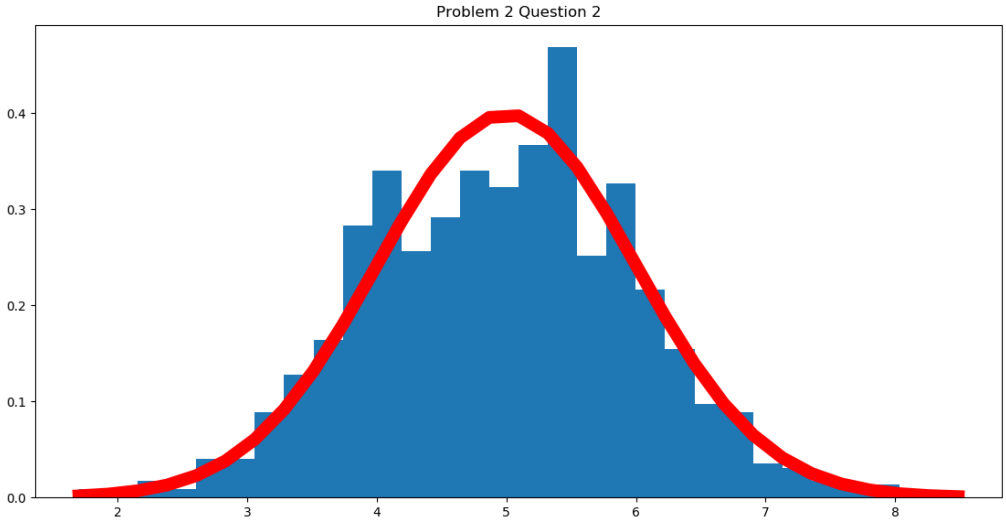
---

h = 5



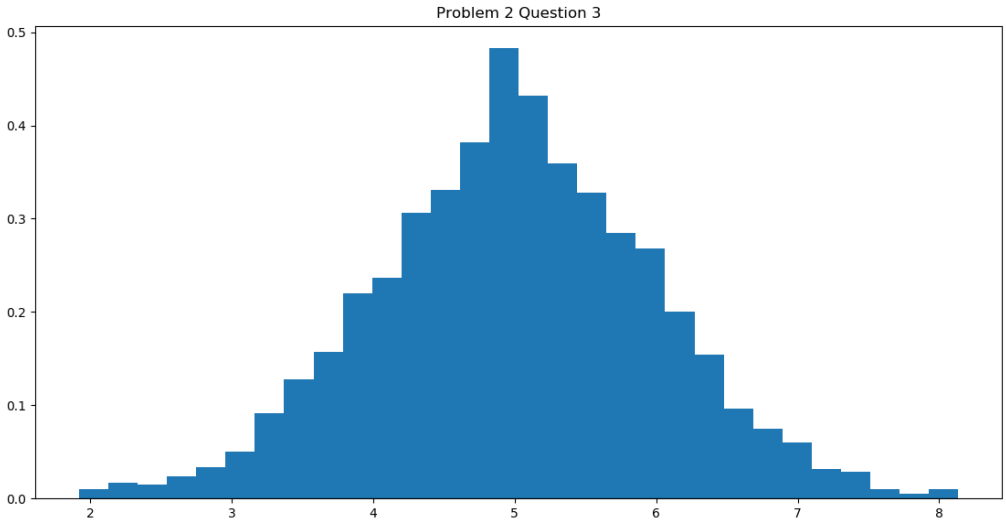
HARSHINE VENKATARAMAN
1001569298
CSE5334
Homework-1
March 10,2019

h = 10



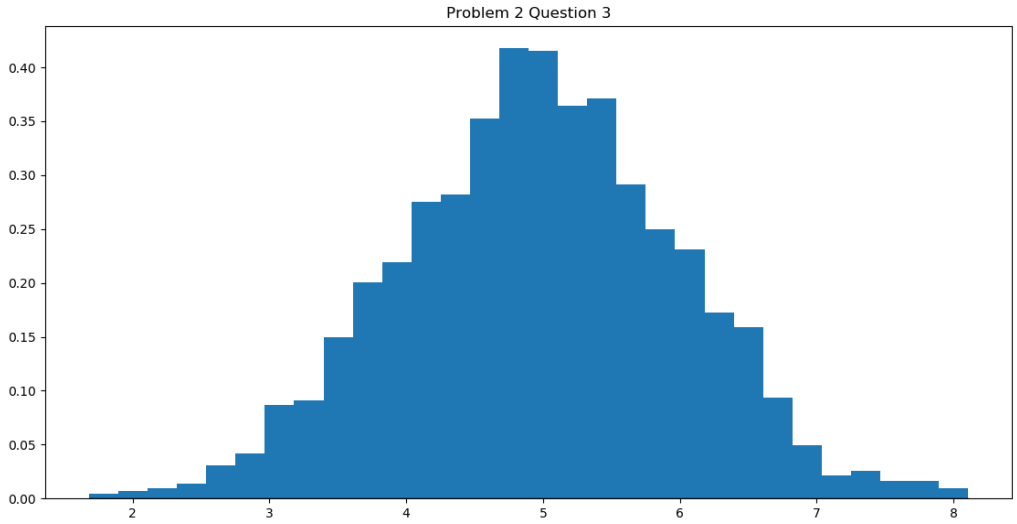
Question3): Generate N = 1000 Gaussian random data with μ1 = 5 and σ1 = 1 and another Gaussian random data with μ2 = 0 and σ2 = 0.2. Test your function mykde on this data with h = {.1, 1, 5, 10}. In your report, report the histogram of X along with the figures of estimated densities.

h = .1



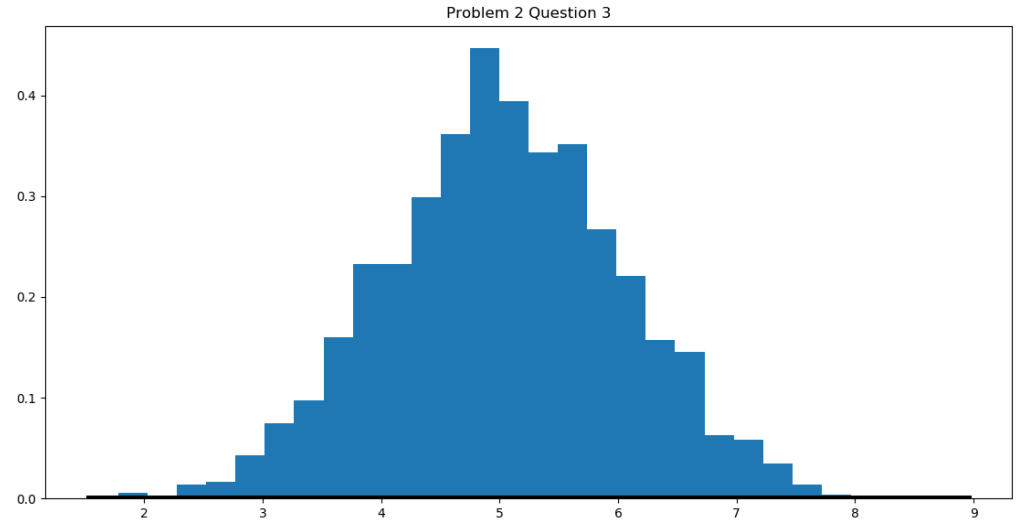
HARSHINE VENKATARAMAN
1001569298
CSE5334
Homework-1
March 10,2019

h = 1



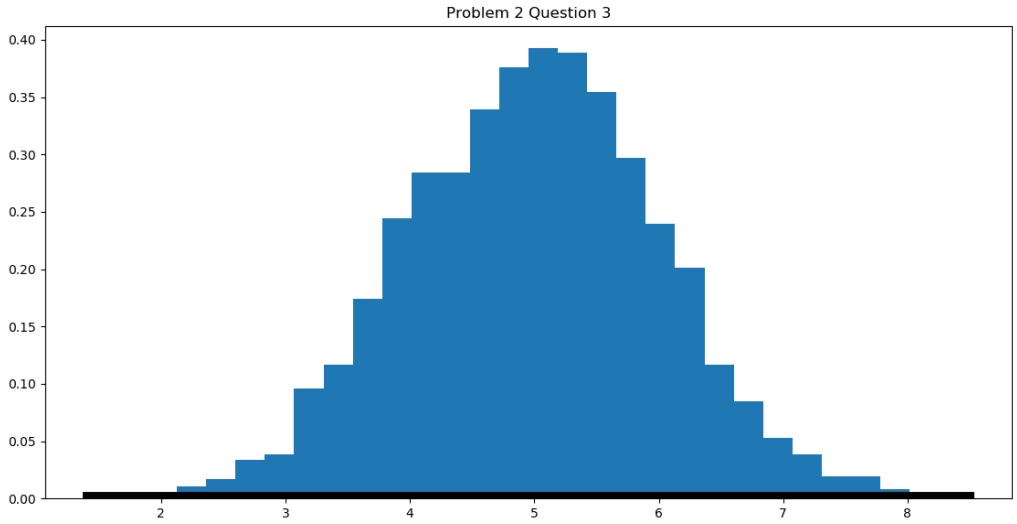
---

h = 5



HARSHINE VENKATARAMAN
1001569298
CSE5334
Homework-1
March 10,2019

h = 10



Question4):

	p(x) actual
p([0,0]^t	0.159154943092
p([0.5,0.5]^t	0.0965323526301
p([0.3,0.2]^t	0.139753228337

Drawback or failure cases:

Size of the training data set

The probability densities are estimated based on the training dataset in Parzen-window technique.

Need a reasonable size dataset for a good estimate.

As the training samples increases in dataset the accuracy increases.(Central limit theorem) since we reduce the likelihood of encountering a sparsity of points for local regions - assuming that our training samples are *i.i.d* (**i**ndependently drawn and **i**dentically **d**istributed).

However, As training samples increases it will lead to a degrade the computational performance.

HARSHINE VENKATARAMAN
1001569298
CSE5334
Homework-1
March 10,2019

References:

```
# References: https://sebastianraschka.com/blog/index.html
# https://scikit-
learn.org/stable/modules/generated/sklearn.neighbors.KernelDensity.html#sklearn.neighb
ors.KernelDensity
```