11849060 谭树杰

Lab 10 –k Nearest Neighbor

and Parzen Window

Class Project 8

sample	ω_1 x_1	ω_1 x_2	ω_1 x_3	ω_2 x_1	ω_2 x_2	ω_2 x_3	ω_3 x_1	ω_3 x_2	ω_3 x_3
1	0.28	1.31	-6.2	0.011	1.03	-0.21	1.36	2.17	0.14
2	0.07	0.58	-0.78	1.27	1.28	0.08	1.41	1.45	-0.38
3	1.54	2.01	-1.63	0.13	3.12	0.16	1.22	0.99	0.69
4	-0.44	1.18	-4.32	-0.21	1.23	-0.11	2.46	2.19	1.31
5	-0.81	0.21	5.73	-2.18	1.39	-0.19	0.68	0.79	0.87
6	1.52	3.16	2.77	0.34	1.96	-0.16	2.51	3.22	1.35
7	2.20	2.42	-0.19	-1.38	0.94	0.45	0.60	2.44	0.92
8	0.91	1.94	6.21	-0.12	0.82	0.17	0.64	0.13	0.97
9	0.65	1.93	4.38	-1.44	2.31	0.14	0.85	0.58	0.99
10	-0.26	0.82	-0.96	0.26	1.94	0.08	0.66	0.51	0.88

Use the right table as the training samples.

Let your window function be a spherical Gaussian, i.e.,

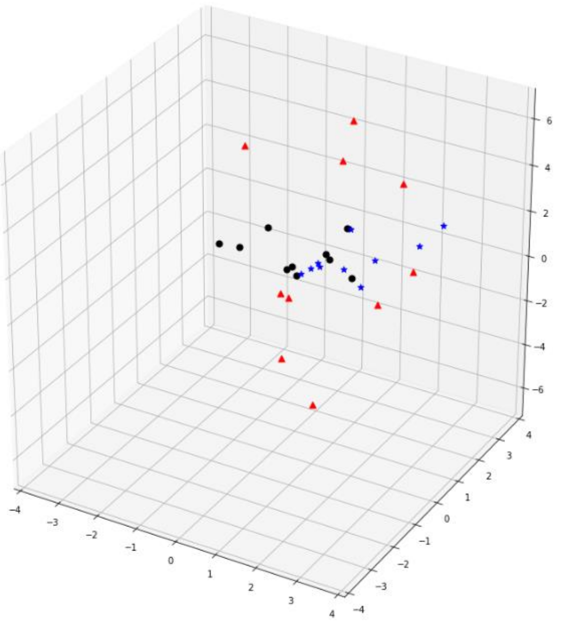
$$\varphi((\mathbf{x}-\mathbf{x}_i)/h) \propto \mathrm{Exp}[-(\mathbf{x}-\mathbf{x}_i)^t(\mathbf{x}-\mathbf{x}_i)/(2h^2)].$$

(a) Write a program to classify an arbitrary test point $\mathbf{x}$ based on the Parzen window estimates. Train your classifier using the three-dimensional data from your three categories in the table above. Set $h = 1$ and classify the following three points: $(0.50, 1.0, 0.0)^t$, $(0.31, 1.51, -0.50)^t$ and $(-0.3, 0.44, -0.1)^t$.

(b) Repeat with $h = 0.1$.

10

We first plot the datasets below, class ω_1 is marked red triangle, class ω_2 is marked read circle, class ω_3 is marked blue star:



For a particular number n (= number of total points), we use a fixed number k (number of points that fall inside the region or volume) and adjust the volume accordingly.

$$V_n = 1/\sqrt{n}$$

Define the kernel function as Guassian:

$$\delta_n(\mathbf{x}) = \frac{1}{V_n}\varphi\left(\frac{\mathbf{x}}{h_n}\right)$$

$$\varphi\left(\frac{x - x_i}{h}\right) \propto \exp(-(x - x_i)^t(x - x_i)/2h^2)$$

Then $p_n(x)$ is the average:

$$p_n(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} \delta_n(\mathbf{x} - \mathbf{x_i})$$

We implement it in python, the result is shown below:

```
h = 1
for i in range(3):
    r=parzen(X,h,test[i,:])
    result=np.where(r==np.max(r))
    print("The probabilities of point "
          + str(test[i,:])+" belong to three class is " + str(r))
#     print(r)
    print("thus, the point "+str(r)+" is class "+ str(result[0][0]+1))
    print(' ')
```

```
The probabilities of point [0.5 1.  0. ] belong to three class is [0.02139395 0.0634711  0.03613364]
thus, the point [0.02139395 0.0634711  0.03613364] is class 2

The probabilities of point [ 0.31  1.51 -0.5 ] belong to three class is [0.02074741 0.04829757 0.0316407 ]
thus, the point [0.02074741 0.04829757 0.0316407 ] is class 2

The probabilities of point [-0.3   0.44 -0.1 ] belong to three class is [0.03038367 0.03413827 0.0032046 ]
thus, the point [0.03038367 0.03413827 0.0032046 ] is class 2
```

For h = 0.1:

```python
h = 0.1
for i in range(3):
    r=parzen(X,h,test[i,:])
    result=np.where(r==np.max(r))
    print("The probabilities of point "
          + str(test[i,:])+" belong to three class is " + str(r))
#     print(r)
    print("thus, the point "+str(r)+" is class "+ str(result[0][0]+1))
    print(' ')
```

```
The probabilities of point [0.5 1.  0. ] belong to three class is [4.34399337e-43 2.69960946e-07 2.45366375e-47]
thus, the point [4.34399337e-43 2.69960946e-07 2.45366375e-47] is class 2

The probabilities of point [ 0.31  1.51 -0.5 ] belong to three class is [1.91818447e-44 6.76567745e-10 2.02834556e-43]
thus, the point [1.91818447e-44 6.76567745e-10 2.02834556e-43] is class 2

The probabilities of point [-0.3   0.44 -0.1 ] belong to three class is [7.46344916e-028 4.77525554e-011 3.33465184e-127]
thus, the point [7.46344916e-028 4.77525554e-011 3.33465184e-127] is class 2
```