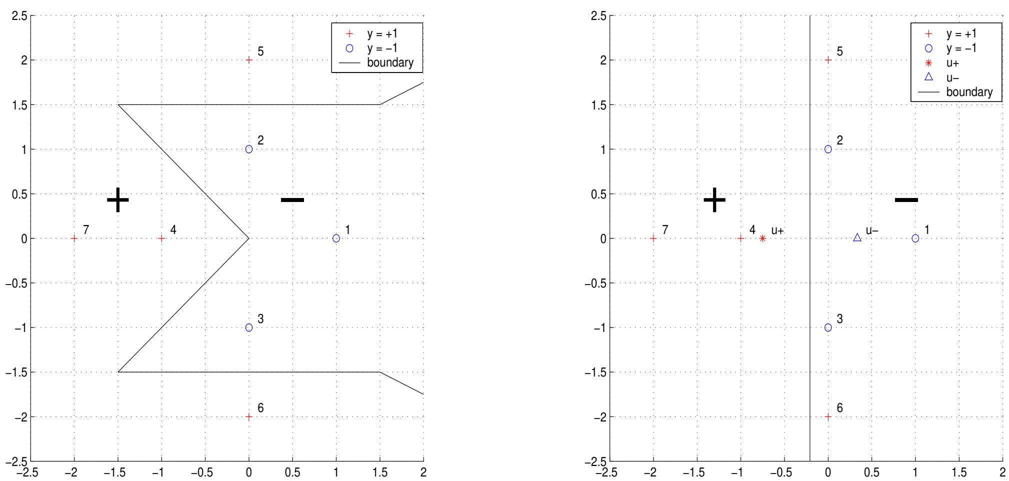
CS 156a Learning Systems - Homework #4

Ling Li, ling@cs.caltech.edu

November 17, 2000

4.1 Nearest-Neighbor Classifiers



(a) Boundary of 7 points.

(b) Boundary of 2 sample means.

Figure 1: The decision boundaries of the nearest-neighbor rule.

(i) Fig. 1(a) gives the decision boundary for those two classes.

(ii) The sample means of each class are: $\mu_{+1} = \left(-\frac{3}{4}, 0\right)$ and $\mu_{-1} = \left(\frac{1}{3}, 0\right)$. Thus the decision boundary is a line $(-\frac{5}{24}, t)$, shown in Fig. 1(b).

(iii) If the decision boundary in (ii) is used, two examples $\mathbf{x}_5 = (0, 2)$ and $\mathbf{x}_6 = (0, -2)$ would be misclassified.

4.2 Radial Basis Functions

$$g(\mathbf{x}) = \sum_{n=1}^{7} w_n e^{-0.5\|\mathbf{x}-\mathbf{x}_n\|^2}$$

(i) Use Matlab code

```
x = [1 0 0 -1 0 0 -2; 0 1 -1 0 2 -2 0];
y = [-1 -1 -1 1 1 1 1];
w = inv(exp(-0.5*dist2(x, x))) * y';
```

where `dist2(x,y)` is a Matlab program to calculate the distance between each column of `x` and each column of `y`.[1] We get

$$\mathbf{w} = \begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \\ w_6 \\ w_7 \end{pmatrix} = \begin{pmatrix} 0.99235287210292 \\ -4.10693164667434 \\ -4.10693164667434 \\ 3.84591206753906 \\ 3.15277397080197 \\ 3.15277397080197 \\ -0.78494281010475 \end{pmatrix}.$$

A little surprise to find w_1 and w_7 has opposite signs with their corresponding output y_1 and y_7. However, it really satisfies $y_i = g(x_i)$.

(ii) I designed a program to plot the decision boundary for these radial basis functions. It turns

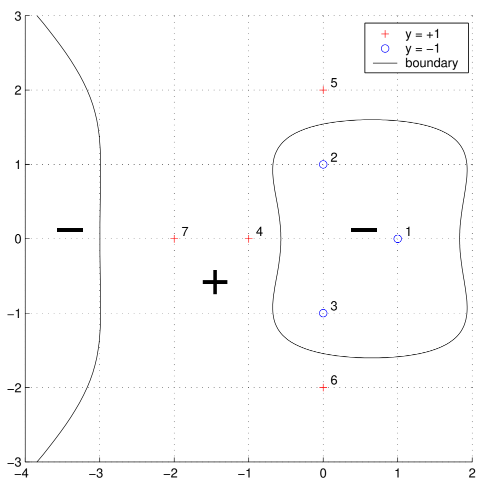


Figure 2: Decision boundary of radial basis functions.

out that the whole space is divided into 3 parts, two with -1 as output and the other one with $+1$ as output.

4.3 Experiment with Nearest Neighbors

(i) For $K = 1$, 19 out of 250 test points are wrongly classified based on the training data. If the mean squared error (MSE) in Homework #1 is used, the error here is

$$\frac{19 \times \|1 - (-1)\|^2}{250} = 0.304,$$

which is smaller than what we got in Homework #1 with 1 hidden neuron, and larger than that of 2–5 hidden neurons.

[1] All the codes for this homework can be found at `http://www.cs.caltech.edu/~ling/course/cs156a/hw4/`.

(ii) For different K, the number of mis-classified test points (denoted by N_E) and the MSE's are

K	1	3	5	7	9	11	13	15
N_E	19	24	28	22	20	41	79	111
MSE	0.304	0.384	0.448	0.352	0.320	0.656	1.264	1.776

The curve of the percentage of mis-classified test points can be found in Fig. 3.

(iii) We can estimate the out-of-sample error by in-sample leave-one-out error. That is, we can take one sample out of the training set and use the other samples to classify that taken-out sample. The average error in classifying those taken-out samples (denoted as E_{est}) can be taken as the estimate of the out-of-sample error. Here for a training set of 25 points, the standard variation of the out-of-sample is also calculated as

$$\sigma_{\text{est}} = \sqrt{\frac{E_{\text{est}}(1 - E_{\text{est}})}{25}}.$$

K	1	3	5	7	9	11	13	15
E_{est}	0.12	0.20	0.16	0.12	0.16	0.32	0.48	0.84
σ_{est}	0.0650	0.0800	0.0733	0.0650	0.0733	0.0933	0.0999	0.0733

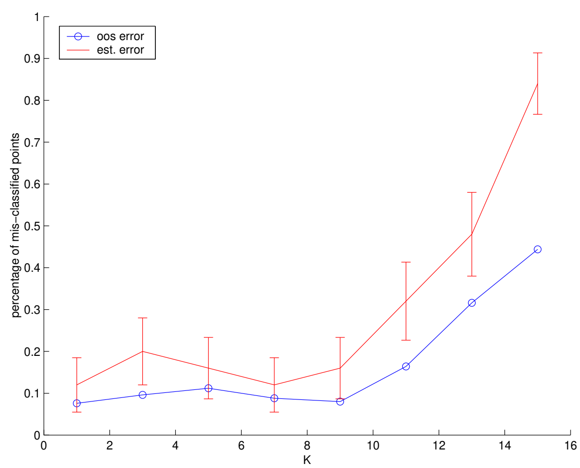
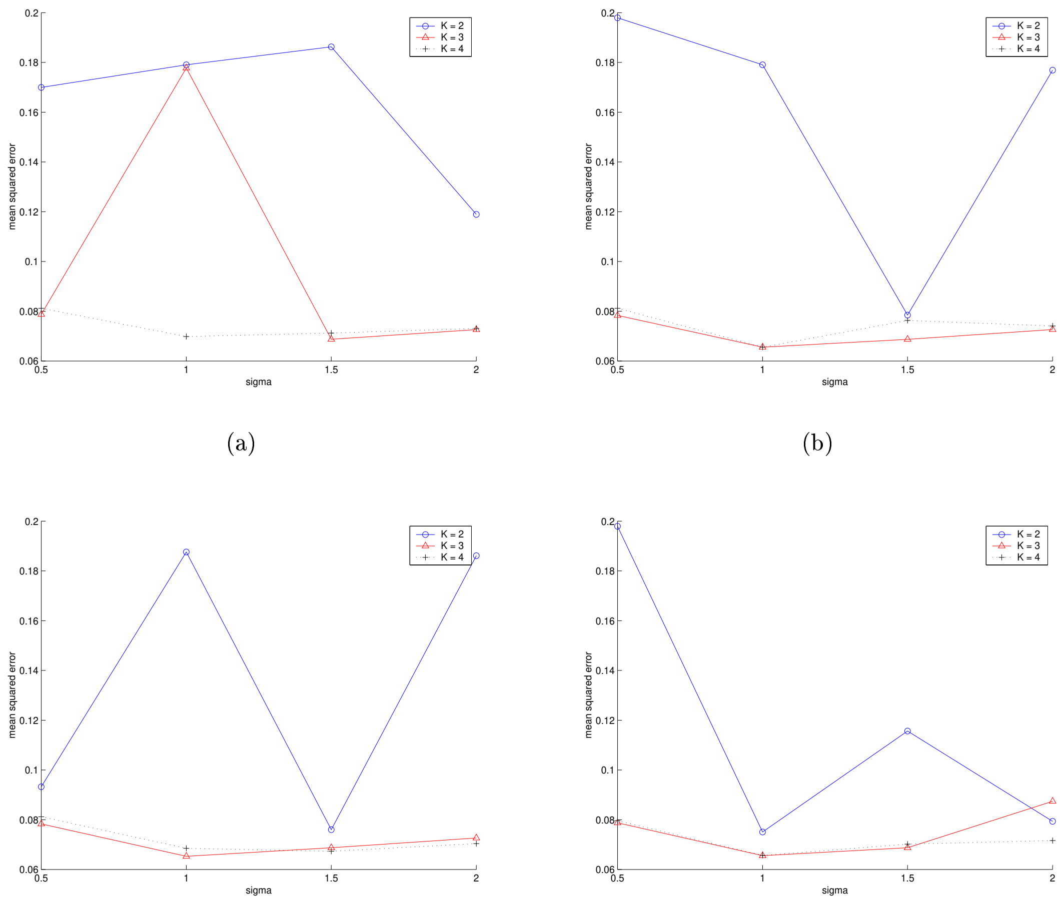


Figure 3: The actual out-of-sample errors (testing errors) and the in-sample leave-one-out errors (estimated errors).

Fig. 3 gives the actual out-of-sample errors and their in-sample leave-one-out estimates. The estimated error has a similar trend as the out-of-sample error, and is always larger. However, these two properties in Fig. 3 are not true for all situations. I used different combinations of the given 275 points as training/testing set, and found out anything can happen. Sometimes the estimate remains very small and the out-of-sample error goes very high; sometimes the out-of-sample error seems to be one half of the estimate (such as in Fig. 3). Averagely, the in-sample leave-one-out estimate is close to the out-of-sample error.

4.4 Experiment with Radial Basis Functions

The clustering (Lloyd's algorithm) may produce different results since the initial setting of the centers can vary much. So when plotting the mean squared errors of the radial basis functions, I got many different combinations, 4 of them shown in Fig. 4.

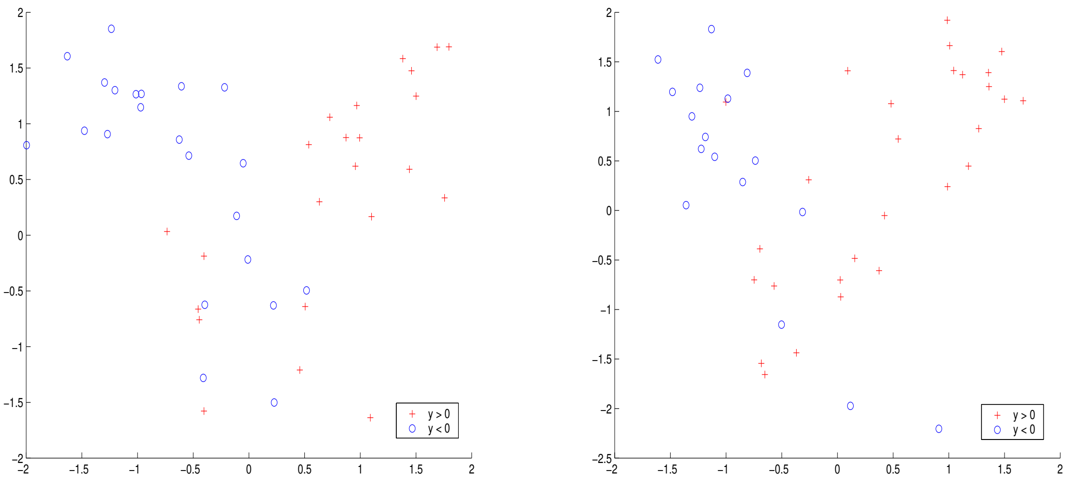


(a) (b) (c) (d)

Figure 4: Testing errors of radial basis functions.

From the plots we can find that

- Usually the error of $K = 2$ is larger than those of $K = 3$ and $K = 4$. However, situations under which the error of $K = 2$ is smaller do exist (Fig. 4(d)).
- The errors of $K = 3$ and $K = 4$ usually are very close, and small. However, during about 100 runs, the error of $K = 3$ sometimes was large (≥ 0.15) while the error of $K = 4$ was always small.
- The value of σ has no significant impact on the error. However, from the error curve of $K = 3$ and $K = 4$, we may deduce that usually $\sigma = 1$ does best.



(a) training data (b) testing data

Figure 5: Training and testing data for problem 4.4.

When we look at the distributions of the training and testing data (Fig. 5), the impression is that the data naturally form 3 clusters. This may explain why $K = 2$ usually does worse than $K = 3$ or $K = 4$, and the latter two usually have similar performances. And the size of one cluster for $K = 3, 4$ is approximately less than 2×2, which means $\sigma = 1 \sim 1.5$ is most suitable for a radial basis function, since with such value the function has great effect on points within the same cluster while having small effect on points in other clusters.