Classification with Gaussians

Hiroshi Shimodaira*

27 February 2015

In the previous chapter we looked at probabilistic models of continuous variables; in particular we introduced the Gaussian (Normal) probability distribution, probably the most important probability distribution for continuous variables.

1 Classification

As before we use Bayes' theorem for classification, to relate the probability density function of the data given the class to the posterior probability of the class given the data.

First we consider the univariate case, with a continuous random variable x, whose pdf, given class C, is a Gaussian with mean μ_c and variance σ_c^2. Using Bayes' theorem we can write:

$$\begin{aligned} P(C\,|\,x) &\propto p(x|C)\,P(C) \\ &\propto N(x;\mu_c,\sigma_c^2)\,P(C) \\ &\propto \frac{1}{\sqrt{2\pi\sigma_c^2}}\exp\left(\frac{-(x-\mu_c)^2}{2\sigma_c^2}\right)P(C)\,, \end{aligned} \tag{1}$$

where $p(x|C)$ is the likelihood of class C given observation x[1].

Log likelihoods and log probabilities When dealing with Gaussians, it is often useful to take logs. We can define the log likelihood of the Gaussian pdf $LL(x|C) = LL(x|\mu_c,\sigma_c^2)$:

$$\begin{aligned} LL(x|\mu_c,\sigma_c^2) &= \ln p(x|\mu_c,\sigma_c^2) \\ &= \ln\left[\frac{1}{\sqrt{2\pi\sigma_c^2}}\exp\left(\frac{-(x-\mu_c)^2}{2\sigma_c^2}\right)\right] \\ &= -\ln\left(\sqrt{2\pi\sigma_c^2}\right) - \frac{(x-\mu_c)^2}{2\sigma_c^2} \\ &= \frac{1}{2}\left(-\ln(2\pi) - \ln\sigma_c^2 - \frac{(x-\mu_c)^2}{\sigma_c^2}\right) \end{aligned} \tag{2}$$

We can use Bayes' theorem to write log posterior probability $LP(C\,|\,x)$:

$$\begin{aligned} LP(C\,|\,x) &= LL(x|C) + LP(C) + \text{const.} \\ &= \frac{1}{2}\left(-\ln(2\pi) - \ln\sigma_c^2 - \frac{(x-\mu_c)^2}{\sigma_c^2}\right) + \ln P(C) + \text{const.}\,, \end{aligned} \tag{3}$$

*Heavily based on notes inherited from Steve Renals and Iain Murray.

[1] As was mentioned in Note 8, although $p(x|C)$ is generally called the class-conditional density function of x, it should be called the likelihood function of C when classification is concerned.

where $LP(C)$ is the log prior probability of class C, and 'const.' is the log of the constant of proportionality, $p(x)$, in equation (1), Bayes' theorem.

Log probability ratio If c_1 and c_2 are modelled by Gaussians with means μ_a and μ_b, and variances σ_a^2 and σ_b^2, then we can write the log odds (ratio of posterior probabilities) as follows:

$$\ln\frac{P(c_1\,|\,x)}{P(c_2\,|\,x)} = \ln P(c_1\,|\,x) - P(c_2\,|\,x)$$

$$\begin{aligned} \ln\frac{P(c_1\,|\,x)}{P(c_2\,|\,x)} &= \frac{1}{2}\left(-\ln(2\pi) - \ln\sigma_a^2 - \frac{(x-\mu_a)^2}{\sigma_a^2}\right) \\ &\quad - \frac{1}{2}\left(-\ln(2\pi) - \ln\sigma_b^2 - \frac{(x-\mu_b)^2}{\sigma_b^2}\right) + (\ln P(c_1) - \ln P(c_2)) \\ &= -\frac{1}{2}\left(\frac{(x-\mu_a)^2}{\sigma_a^2} - \frac{(x-\mu_b)^2}{\sigma_b^2} + \ln\sigma_a^2 - \ln\sigma_b^2\right) + \ln P(c_1) - \ln P(c_2)\,. \end{aligned} \tag{4}$$

Look at the right hand side of equation 4. The first two terms are the variance-weighted Euclidean distances from the means, the second two terms are log variances (arising from the normalisation terms) and the third two terms are the log prior probabilities. Think through why, intuitively, each of these terms should affect which class you believe best explains an observation.

2 Example: Univariate Gaussian classifier

We return to our previous pattern recognition example, a problem with two classes, S and T. Some observations are available for each class:

Class S	10	8	10	10	11	11
Class T	12	9	15	10	13	13

We assume that each class may be modelled by a Gaussian. The maximum likelihood estimators of the mean and variance of each pdf are:

$$\hat{\mu}(S) = 10 \qquad \hat{\sigma}^2(S) = 1$$
$$\hat{\mu}(T) = 12 \qquad \hat{\sigma}^2(T) = 4$$

The following unlabelled data points are available:

$$x_1 = 10 \qquad x_2 = 11 \qquad x_3 = 6$$

To which class should each of the data points be assigned? Assume the two classes have equal prior probabilities.

Since this is a two class problem, it is convenient to calculate the log posterior probability ratios for each case. (In a multiclass problem, we would calculate the log posterior probability for each class.)

$$\begin{aligned} \ln\frac{P(S\mid X=x)}{P(T\mid X=x)} &= -\frac{1}{2}\left(\frac{(x-\mu_S)^2}{\sigma_S^2} - \frac{(x-\mu_T)^2}{\sigma_T^2} + \ln\sigma_S^2 - \ln\sigma_T^2\right) + \ln P(S) - \ln P(T) \\ &= -\frac{1}{2}\left(\frac{(x-\mu_S)^2}{\sigma_S^2} - \frac{(x-\mu_T)^2}{\sigma_T^2} + \ln\sigma_S^2 - \ln\sigma_T^2\right) \\ &= -\frac{1}{2}\left((x-10)^2 - \frac{(x-12)^2}{4} - \ln 4\right) \end{aligned}$$

If the log ratio is less than 0, then assign to class T, otherwise assign to class S.

- $x_1 = 10$:

$$\ln \frac{P(S \mid X=x_1)}{P(T \mid X=x_1)} = -\frac{1}{2}\left((x_1-10)^2 - \frac{(x_1-12)^2}{4} - \ln 4\right)$$
$$= -\frac{1}{2}(0 - 1 - \ln 4)$$
$$= 1.19$$

- $x_2 = 11$:

$$\ln \frac{P(S \mid X=x_2)}{P(T \mid X=x_2)} = -\frac{1}{2}\left((x_2-10)^2 - \frac{(x_2-12)^2}{4} - \ln 4\right)$$
$$= -\frac{1}{2}(1 - 0.25 - \ln 4)$$
$$= 0.32$$

- $x_3 = 6$:

$$\ln \frac{P(S \mid X=x_3)}{P(T \mid X=x_3)} = -\frac{1}{2}\left((x_3-10)^2 - \frac{(x_3-12)^2}{4} - \ln 4\right)$$
$$= -\frac{1}{2}(16 - 9 - \ln 4)$$
$$= -2.81$$

We assign x_1 to S; x_2 to S; x_3 to T.

Now assume that the two classes do not have equal prior probabilities, in fact $P(S)=0.3$, $P(T)=0.7$. Including this prior information, to which class should each of the above test data points $\{x_1, x_2, x_3\}$ be assigned?

Again compute the log posterior probability ratios:

$$\ln \frac{P(S \mid X=x)}{P(T \mid X=x)} = -\frac{1}{2}\left(\frac{(x-\mu_S)^2}{\sigma_S^2} - \frac{(x-\mu_T)^2}{\sigma_T^2} + \ln \sigma_S^2 - \ln \sigma_T^2\right) + \ln P(S) - \ln P(T)$$
$$= -\frac{1}{2}\left((x-10)^2 - \frac{(x-12)^2}{4} - \ln 4\right) + \ln P(S) - \ln P(T)$$
$$= -\frac{1}{2}\left((x-10)^2 - \frac{(x-12)^2}{4} - \ln 4\right) + \ln(3/7)$$

Reclassifying use the prior probability information:

- $x_1 = 10$:

$$\ln \frac{P(S \mid X=x_1)}{P(T \mid X=x_1)} = -\frac{1}{2}\left((x_1-10)^2 - \frac{(x_1-12)^2}{4} - \ln 4\right) + \ln(3/7)$$
$$= -\frac{1}{2}(0 - 1 - \ln 4) + \ln(3/7)$$
$$= 0.34$$

- $x_2 = 11$:

$$\ln \frac{P(S \mid X=x_2)}{P(T \mid X=x_2)} = -\frac{1}{2}\left((x_2-10)^2 - \frac{(x_2-12)^2}{4} - \ln 4\right) + \ln(3/7)$$
$$= -\frac{1}{2}(1 - 0.25 - \ln 4) + \ln(3/7)$$
$$= -0.53$$

- $x_3 = 6$:

$$\ln \frac{P(S \mid X=x_3)}{P(T \mid X=x_3)} = -\frac{1}{2}\left((x_3-10)^2 - \frac{(x_3-12)^2}{4} - \ln 4\right) + \ln(3/7)$$
$$= -\frac{1}{2}(16 - 9 - \ln 4) + \ln(3/7)$$
$$= -3.66$$

We now assign x_1 to S; x_2 to T; x_3 to T

3 Multivariate Gaussian classifier

Now consider d-dimensional data $\mathbf{x}$ from class C modelled using a multivariate Gaussian:

$$p(\mathbf{x}|C) = p(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = N(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$$
$$= \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right). \quad (5)$$

The log likelihood is:

$$LL(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \ln p(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = -\frac{d}{2}\ln(2\pi) - \frac{1}{2}\ln|\boldsymbol{\Sigma}| - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}). \quad (6)$$

And we can write the log posterior probability:

$$\ln P(C|\mathbf{x}) = -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}) - \frac{1}{2}\ln|\boldsymbol{\Sigma}| + \ln P(C) + \text{const.} \quad (7)$$

4 Example: Multivariate Gaussian Classifier

Consider the following problem. We have two-dimensional data from three classes (A, B, C). The classes may be assumed to have equal prior probabilities. Our training data is in files `trainA.dat`, `trainB.dat`, and `trainC.dat`, with test data in files `testA.dat`, `testB.dat`,and `testC.dat`.

These files can be downloaded as `http://www.inf.ed.ac.uk/teaching/courses/inf2b/labs/MGC.zip`.

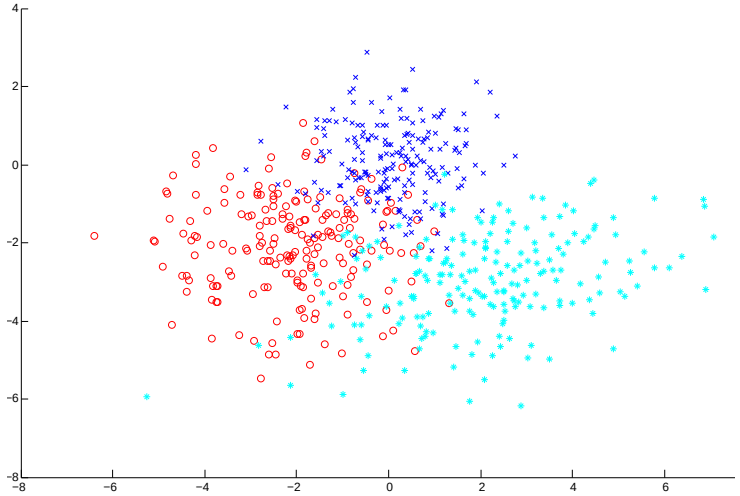

Figure 1: Training data from three classes: A (red circles), B (blue crosses) and C (cyan stars)

There are 200 points from each class for training, and a further 100 points from each class for testing. The data was generated from Gaussian distributions with the following parameters:

$$\mu_A = \begin{pmatrix} -2 \\ -2 \end{pmatrix} \quad \Sigma_A = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$$
$$\mu_B = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \Sigma_B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$
$$\mu_C = \begin{pmatrix} 2 \\ -3 \end{pmatrix} \quad \Sigma_C = \begin{pmatrix} 4.0 & 1.0 \\ 1.0 & 1.5 \end{pmatrix}$$

We can load it into Matlab as follows, then plot the three classes on the same scatter plot (Figure 1).

```
% load training and test data
xa = load('trainA.dat');
xb = load('trainB.dat');
xc = load('trainC.dat');
testa = load('testA.dat');
testb = load('testB.dat');
testc = load('testC.dat');
alltest = [testa; testb; testc];

% plot the training data
figure;
hold on;
scatter(xa(:, 1), xa(:,2), 'r', 'o');
scatter(xb(:, 1), xb(:,2), 'b', 'x');
scatter(xc(:, 1), xc(:,2), 'c', '*');
```


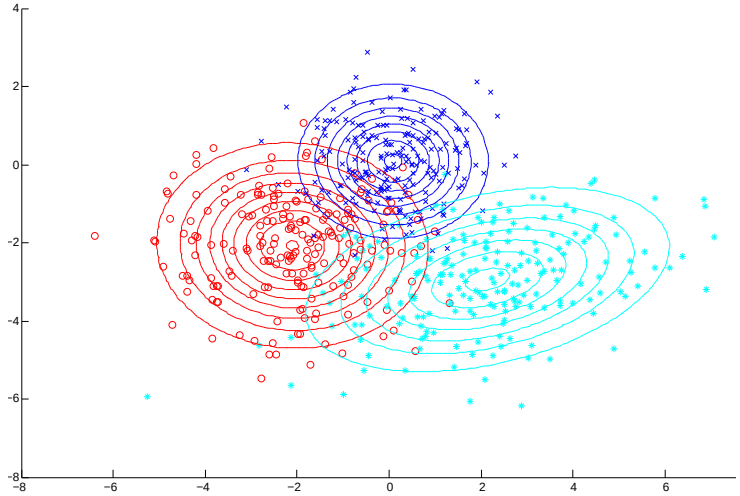

Figure 2: Gaussian distributions estimated from training data for each class

Each row of the matrices `xa`, `xb`, etc. corresponds to a 2-dimensional training data point. To model each class with a Gaussian density, we can immediately estimate the mean and covariance parameters with the sample mean and covariance (computed using the built-in Matlab functions `mean` and `cov`):

```
% train the gaussians
mua = mean(xa);
mub = mean(xb);
muc = mean(xc);

covara = cov(xa);
covarb = cov(xb);
covarc = cov(xc);
```

If there are n data points then `cov(x)` computes the covariance normalising by $1/(n-1)$; `cov(x,1)` normalises by $1/n$.

The resulting estimates for the mean and covariance of each class are as follows:

$$\hat{\mu}_A = \begin{pmatrix} -2.1 \\ -2.1 \end{pmatrix} \quad \hat{\Sigma}_A = \begin{pmatrix} 2.0 & -0.1 \\ -0.1 & 1.6 \end{pmatrix}$$
$$\hat{\mu}_B = \begin{pmatrix} 0.1 \\ 0.1 \end{pmatrix} \quad \hat{\Sigma}_B = \begin{pmatrix} 1.0 & 0.0 \\ 0.0 & 0.9 \end{pmatrix}$$
$$\hat{\mu}_C = \begin{pmatrix} 2.1 \\ -2.9 \end{pmatrix} \quad \hat{\Sigma}_C = \begin{pmatrix} 4.0 & 0.8 \\ 0.8 & 1.4 \end{pmatrix}$$

Compare these with the true values above.

We can plot the resulting Gaussians as contour plots over the training data points (Figure 2).

Figure 3 shows the testing points, labelled with their true classes, together with the Gaussians estimated from the training data.

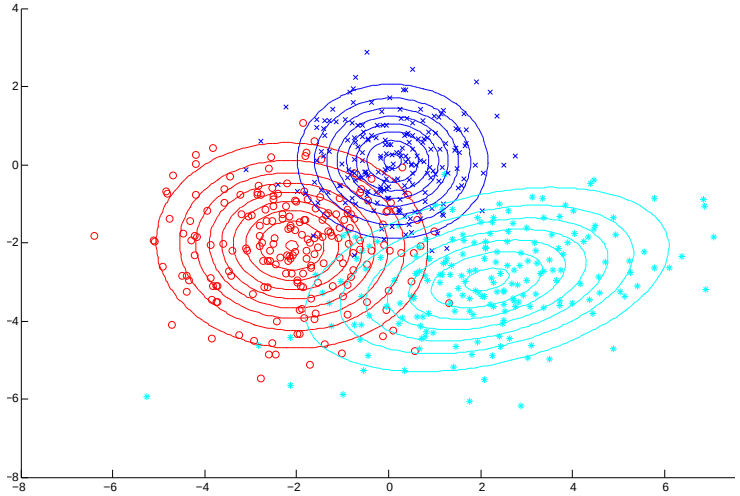

Figure 3: Test points, with true class labels, and distributions estimated from training data.

We can now go ahead and classify each testing point. For test points in class A, we can do the following:

```
testaOut = [gauss(mua,covara,testa) gauss(mub,covarb,testa)
            gauss(muc, covarc, testa)];
[maxaOut, classa] = max(testaOut,[],2);
```

The first line applies each of the three Gaussians to all the test points; the second line determines which class has the highest probability. We can do this for each set of training data. Figures 4, 5, and 6 shows how the test data from each class was classified.

We can look at the results using a *confusion matrix*. The column of a confusion matrix correspond to the predicted classes (i.e., classifier outputs). The rows correspond to the actual (true) class labels. The number at position (r, c) is the number of patterns from true class r that were classified as class c. The number of correctly classified patterns is obtained by summing the numbers on the leading diagonal:

		Predicted class		
Test Data		A	B	C
Actual	A	77	15	8
class	B	5	88	7
	C	9	2	89

From the confusion matrix we can see that the overall proportion of test patterns correctly classified is $(77 + 88 + 89)/300 = 254/300 = 0.85$.

5 Application case study

We discuss the use of a multidimensional Gaussian classifier applied in a medical task. The details are as follows:

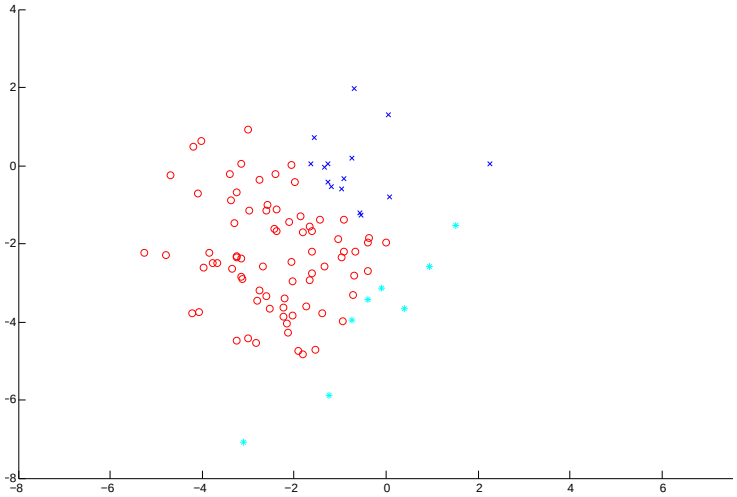

Figure 4: Classification of test points from class A

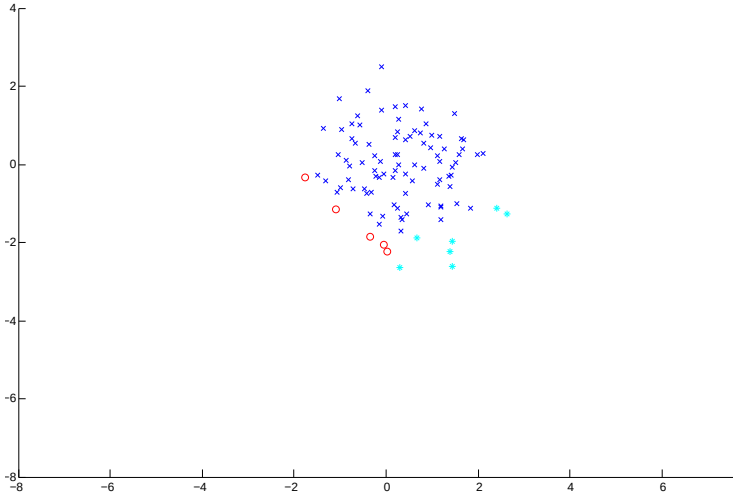

Figure 5: Classification of test points from class B

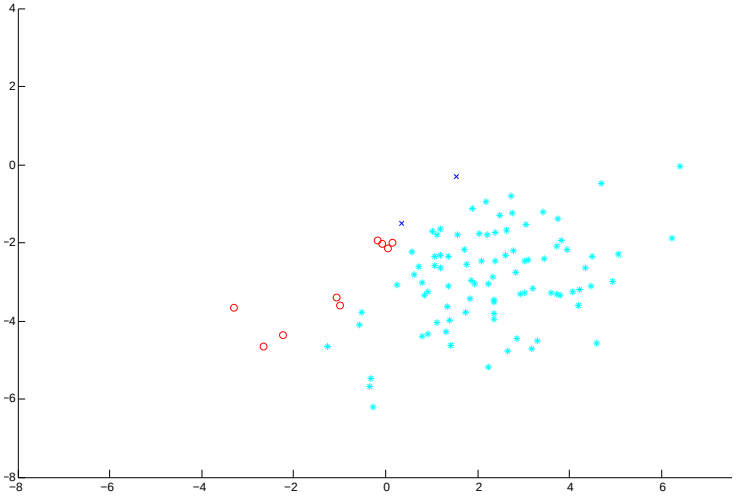

Figure 6: Classification of test points from class C

Task Predict recovery of patients entering hospital with severe head injuries.

Input features 6 categorical variables recorded for each patient: Age, plus 5 responses to stimulation (eye, motor, verbal), graded on scales (e.g., limb movement graded from 1 (nil) to 7 (normal)).

Output classes Three output classes: 1 (death); 2 (severe disability); 3 (good recovery).

Data set 500 patients in both the training and test sets.

Imbalanced classes 50% class 1, 10% class 2, 40% class 3.

Missing data Some feature values in some patterns were missing.

(D. M. Titterington et al (1981), "Comparison of discrimination techniques applied to a complex data set of head-injured patients", *J Roy Stat Soc Ser A*, 144(2), 145–175.)

This problem was tackled by modelling each class using a multivariate Gaussian. Training thus consisted of estimating the mean, covariance matrix and prior probability for each class. The mean vector and covariance matrix for each class was estimated using maximum likelihood (i.e., estimated using sample mean and sample covariance). The prior probabilities were estimated as the relative frequency for each class.

The missing values in the training set filled in with the class mean; since the class is unknown in the test set, the missing values in the test set were filled in with the overall mean.

At test time, each test data point was assigned to the class with the highest posterior probability.

First it is useful to see how well the system performs when tested on the training data. This is the confusion matrix that was obtained:

		Predicted class		
Training Data		1	2	3
Actual	1	209	0	50
class	2	22	1	29
	3	15	1	173

Overall 76.6% ((209 + 1 + 173)/500 = 383/500) of the training patterns were correctly classified.

The following confusion matrix was obtained on the test data:

		Predicted class		
Test Data		1	2	3
Actual	1	188	0	59
class	2	19	1	28
	3	29	2	171

Overall 72.0% ((188 + 1 + 171)/500 = 360/500) of the test patterns were correctly classified.

There are various points which may be noted from this experiment:

- The classification accuracy on the training set was better than the test set, but only a relative improvement of about 6.4%.
- Class two, which corresponded to only 10% of the training set, was rarely selected by the system.
- The baseline strategy, which could be obtained by choosing the class with the highest prior probability (class 1) for each example (i.e., ignoring the data) would achieve about 50% correct.
- The main confusion concerned items from class 1 being misclassified as class 3 (about 10% of all the data in both training and test sets).