10-701/15-781, Machine Learning: Homework 2

Eric Xing, Tom Mitchell, Aarti Singh
Carnegie Mellon University
Updated on January 20, 2010

1 Multiclass Classification[35pt, Ni Lao]

In this part, you are going to play with the 'The ORL Database of Faces'.

Figure 1: 6 sample images from two persons

Each image is 92 by 112 pixels. If we treat the luminance of each pixel as a feature, each sample has $92 * 112 = 10304$ real value features, which can be written as a random vector X. We will treat each person as a class Y ($Y = 1...K, K = 10$). We use X_i to refer the i-th feature. Given a set of training data $D = \{(y^l, x^l)\}$, we will train different classification models to classify images to their person id's. To simplify notation, we will use $P(y|x)$ in place of $P(Y = y|X = x)$.

We will select our models by 10-fold cross validation: partition the data for each face into 10 mutually exclusive sets (folds). In our case, exactly one image for each fold. Then, for k=1...10, leave out the data from fold k for all faces, train on the rest, and test on the left out data. Average the results of these 10 tests to estimate the training accuracy of your classifier.

Beware that we are actually not evaluating the generalization errors of the classifier here. When evaluating generalization error, we would need an independent test set that is not at all touched during the whole developing and tuning process.

For your convenience, a piece of code "loadFaces.m" is provided to help loading images as feature vectors.

1. **KNN [5 pt]** Implement the KNN algorithm we learnt from the class. Use L2-norm as the distance metric. Show your evaluation result here, and compare different values of K.

2. **Gaussian Classifier [5 pt]** For a Gaussian model we have

$$P(y|x) = \frac{P(x|y)P(y)}{P(x)}$$

where

$$P(x|y) = \frac{1}{(2\pi)^{d/2}|\Sigma_y|^{1/2}} \exp\{-(x-\mu_y)'\Sigma^{-1}(x-\mu_y)/2\},$$

$P(x) = \sum_y P(x|y)P(y)$, and $P(y) = \pi_y$. Please write down the MLE estimation of model parameters Σ_y, μ_y, and π_y. Here we do not assume that X_i are independent given Y

3. **Gaussian Naive Bayes Model [5 pt]** Gaussian NB is a form of Guassian model with assumption that X_i are independent given Y. Please implement the Gaussian NB model. Show your evaluation result here.

4. **Multinomial Logistic Regression [5 pt]** From the reading material (Tom's chapter draft) you will see a generalization of logistic regression, which allow Y to have more than two possible values. Write down the objective function, and the first order derivatives of the multinomial logistic regression model. Here we consider a L2-norm regularized objective function (with a term $\lambda|\theta|_2$).

5. **Gradient Ascent [5 pt]** Implement the logistic regression model with gradient ascent. Show your evaluation result here. Use regularization parameter $\lambda = 0$. **Hint:** The gradient ascent method (also known as "steepest ascent") is a first-order optimization algorithm. It optimizes a function $f(x)$ by

$$x_{t+1} = x_t + \alpha_t f'(x_t),$$

where α_t is called the *step size*, which is often picked by *line search*. For example, we can initialize $\alpha_t = 1.0$. Then set $\alpha_t = \alpha_t/2$ while $f(x_t + \alpha_t f'(x_t)) < f(x_t)$. The iteration stops when the change of x or $f(x)$ is smaller than a threshold (the optimization is converged). **Hint:** if the training time of your model is too long, you can consider use just a subset of the features (e.g. in Matlab X=X(:,1:100:d))

6. **Overfitting and Regularization [5 pt]** Now we test how regularization can help prevent overfitting. During cross validation, let's use m images from each person for training, and the reset for testing. Report your cross-validated result with varying $m = 1...9$ and varying regularization parameter λ.

7. **[5 pt]** Compare the four methods by training/testing time, and accuracy. Which method do you prefer?