PRML ASSIGNMENT 1

April 2020

1 Generative modeling and Discriminant modeling

1.1 Generative modeling with the multivariate Gaussian

The two-class generative model learned from the class is to evaluate the posterior by the product of likelihood and prior. And the class that can maximize the posterior is the result. We assume that y obeys the Bernoulli distribution and x for every class obeys multivariate Gaussian distribution and shares the same con variance. By maximizing the log-likelihood, the parameters are gained.

$$p(y|x) \propto p(x|y)p(y) \tag{1}$$

$$\hat{y} = \arg \max_{y \in Q,1\}} P(y|x) = \arg \max_{y} P(y) \cdot p(x|y) \tag{2}$$

Accordingly, the three-class classification also uses the prior and likelihood to evaluate the posterior and deems the class that maximize the posterior as the final result. We still assume that every class's distribution share the same co variance. K classes weights π_j, and class-conditional densities are $P_j = N\left(\mu_j, \Sigma_j\right)$. Each class has an associated quadratic function,$f_j(x) = \log\left(\pi_j P_j(x)\right)$. To classify point x, pick $\arg\max_j f_j(x)$. Because $\Sigma_1 = \cdots = \Sigma_k$, the boundaries are linear. The evaluation functions for μ and Σ for every class's Gaussian distribution are as follows.

$$\mu = \frac{1}{m}\left(x^{(1)} + \cdots + x^{(m)}\right) \tag{3}$$

$$\Sigma = \frac{1}{m} \sum_{k=1}^{m} \left(x^{(k)} - \mu\right)\left(x^{(k)} - \mu\right)^{T} \tag{4}$$

1.2 Discriminative Modeling(Softmax Regression

As is taught on class, logistics regression, a.k.a discriminative model, assume that the postier obey sigmoid function. After defining the likelihood function and minimizing it with the gradient descent, the argmax of log-likelihood is gained.

Similarly, softmax regression for multi-classification problem is to assume that the postiers obey softmax function and minimize the log-likelihood by gradient descent.Concretely, our hypothesis h (x) takes the form:

$$h_\theta(x) = \begin{bmatrix} P(y=1|x;\theta) \\ P(y=2|x;\theta) \\ \vdots \\ P(y=K|x;\theta) \end{bmatrix} = \frac{1}{\sum_{j=1}^{K} \exp(\theta(j)\top x)} \begin{bmatrix} \exp\left(\theta^{(1)\top} x\right) \\ \exp\left(\theta^{(2)\top} x\right) \\ \vdots \\ \exp\left(\theta^{(K)\top} x\right) \end{bmatrix} \tag{5}$$

Here $\theta^{(1)}, \theta^{(2)}, \ldots, \theta^{(K)} \in \Re^n$ are the parameters of our model. Notice that the term $\frac{1}{\Sigma_{j-1}^{K} \exp\left(\theta^{(j)} z\right)}$ normalizes the distribution, so that it sums to one.

Therefore, our MLE function is as follows.

$$J(\theta) = -\left[\sum_{i=1}^{m} \sum_{k=1}^{K} 1\left\{y^{(i)} = k\right\} \log \frac{\exp\left(\theta^{(k)\top} x^{(i)}\right)}{\sum_{j=1}^{K} \exp\left(\theta^{(j)\top} x^{(i)}\right)}\right] \tag{6}$$

we cannot solve for the minimum of $J(\theta)$ analytically, and thus as usual we resort to an iterative optimization algorithm. Taking derivatives, one can show that the gradient is:

$$\nabla_\theta(k) J(\theta) = -\sum_{i=1}^{m} \left[x^{(i)} \left(1\left\{y^{(i)} = k\right\} - P\left(y^{(i)} = k | x^{(i)}; \theta\right)\right)\right] \tag{7}$$

Armed with this formula for the derivative, one can then plug it into a GD optimization algorithm and have it minimize $J(\theta)$.

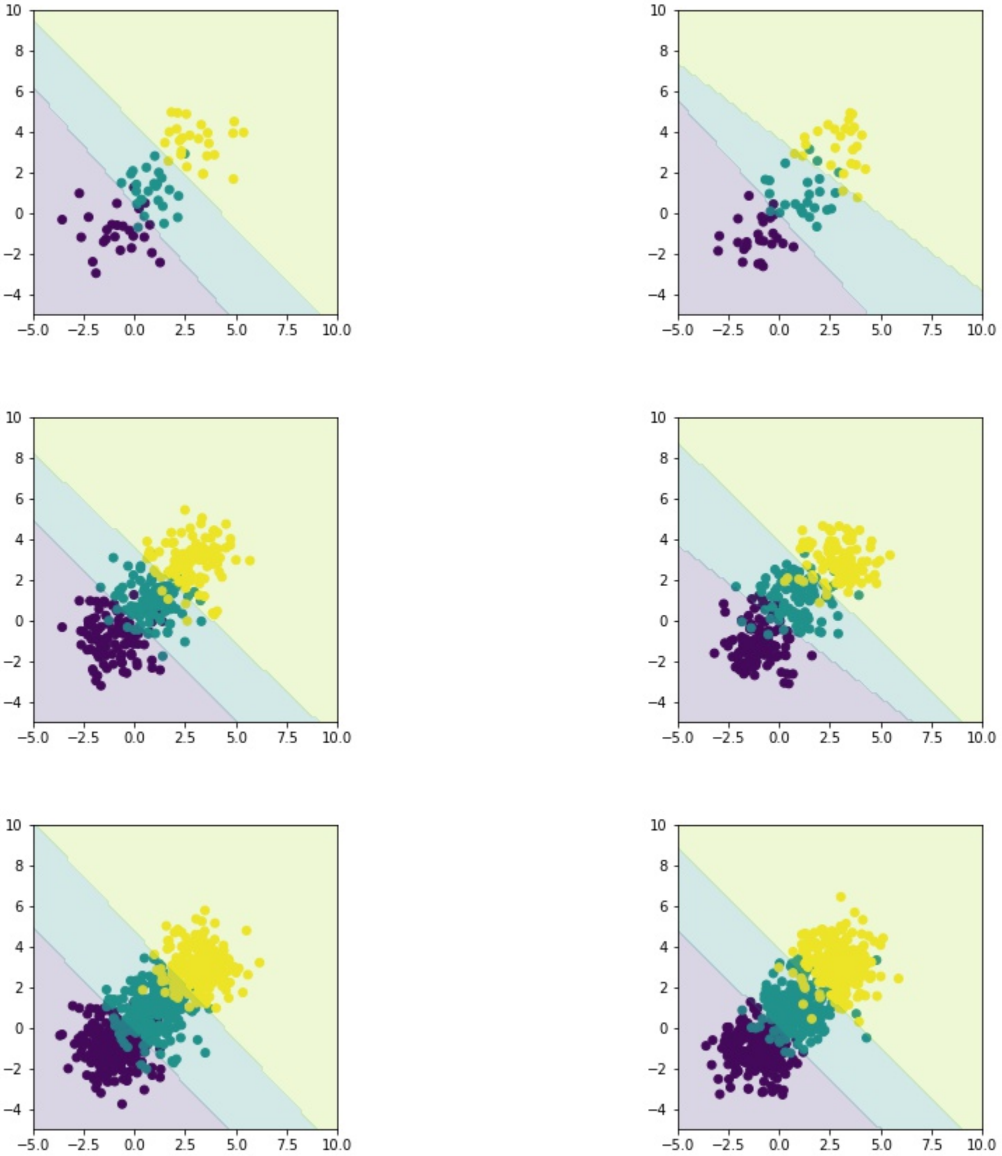
2 Exploration

2.1 Exploration:variation of the number of data used

In this section, the classification performances of discriminative and generative model are explored. I choose three Gaussian distributions and use different numbers of sample points for exploration.

As is shown in the pictures, with the sampling points growing, the classification boundaries do not show a huge discrepancy. The similarity between GDA and Softmax Regression can be attributed to the little overlapping level among different classes.

$$\boldsymbol{\mu 1} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \ \boldsymbol{\mu 2} = \begin{pmatrix} 3 \\ 3 \end{pmatrix}, \boldsymbol{\mu 3} = \begin{pmatrix} -1 \\ -1 \end{pmatrix} \quad \boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{8}$$



(a) GDA

(b) Softmax Regression

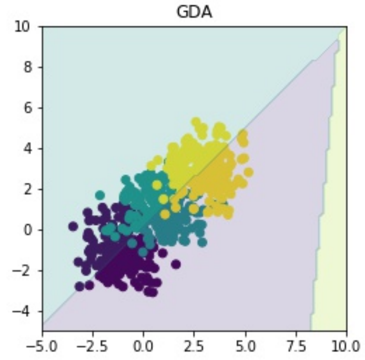
2.2 Exploration:variation of the overlapping levels

In this section, I will explore the influence caused by different gaussian overlapping levels and examine the boundary's change of the two methods. Here, I still use the Gaussian distributions mentioned before and only gradually change the variance to control the overlapping levels. The diagonal value of the covariance matrix is changed to 1.5, 2, 2.5 and the corrosponding classification results are as follows.

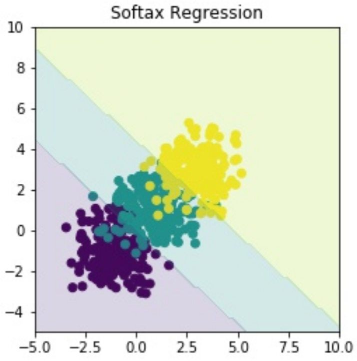
As is shown in the picture(c) and (d), the boundaries of GDA are quite stable comparing to the softmax regression. This is because we only use Gaussian distribution to generate the random points and GDA take the mean and variance of the all points to classify. The drastic overlapping level does not do damage to the fact that those points' mean and variance still obey the Gaussian distribution rule and therefore GDA perform better when the points only obey the Gasussian distribution.

2.3 Exploration: the noise affect

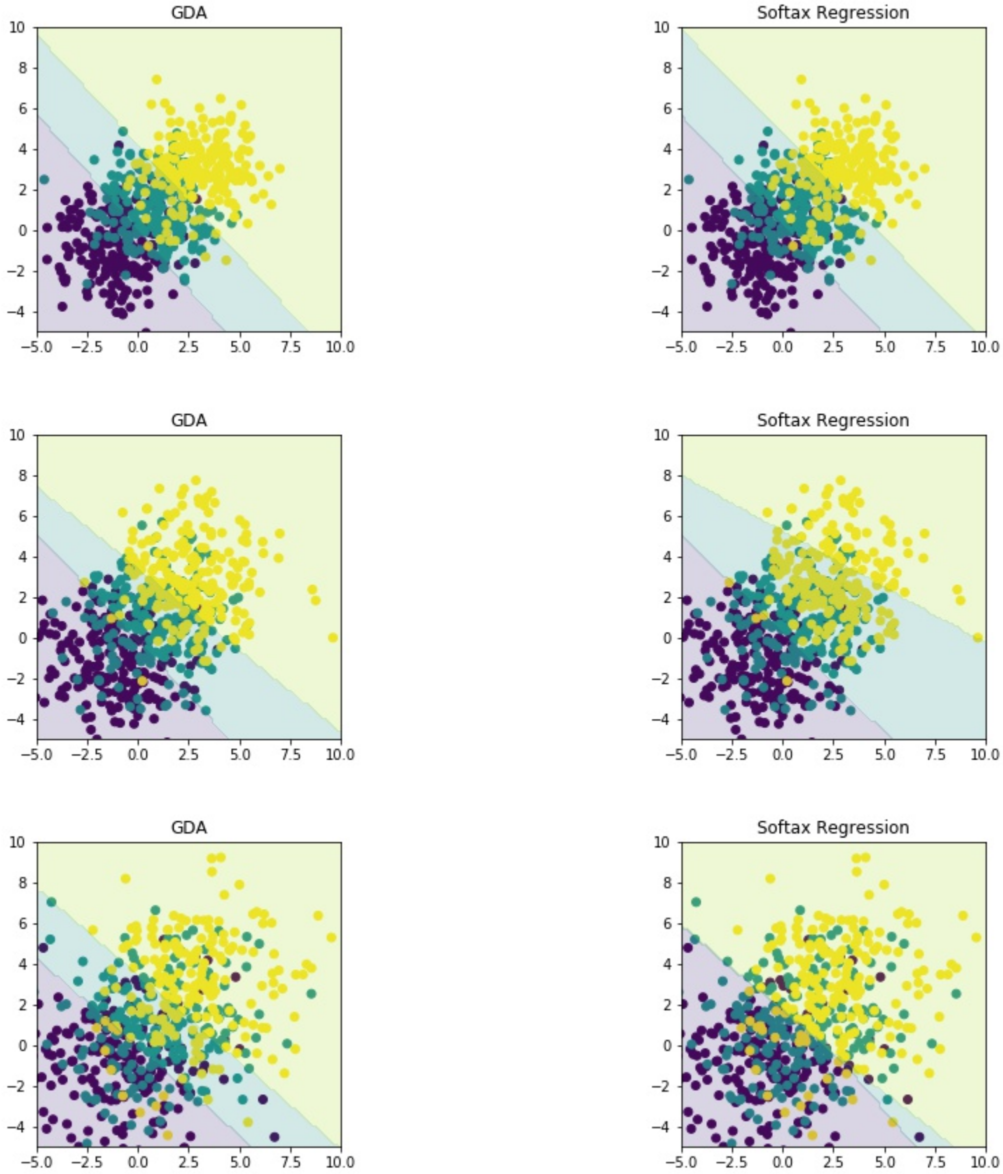
As is taught in class, though GDA perform better when applied to tasks obeying the Gaussian distribution, its effect is not as robust as the softmax regression is when applied to other kind of works. To prove that, I add some Yellow class pointes where located around (100,100) to the original database and explore the different response that GDA and softmax regression exhibit. And the GDA performance change hugly while softmax regression stays stable (pic.(e) and (f)).



(e) pic1.GDA



(f) pic2.Softmax Regression



(c) GDA

(d) Softmax Regression

3 Comparison

Based on exploration results, softmax regression is much more robust than GDA. When applied to tasks that obeys a not perfect Gaussion distribution, softmax regression can adjust to the data better and therefore has a better performance. Oppositely, GDA can perform better if the tasks totally obeying the Gaussian distribution; no matter how overlapping the classes are, GDA still does a good job. Besides, we can also use the GDA's model to generate more points while discriminative model can only classify.

4 Appendix

```
python source.py samples variance1 variance2 variance3
```

for example:

```
python source.py 200 1 1 1
```