Solution

1. Generate dataset

1.1. Algorithm

A mixed Gaussian distribution has the probability density distribution as follow:

$\text{mixedGauss}(X) = \sum_{k=1}^{5} w_k N(\mu_k, \Sigma_k)$ with X is multi-dimension random variable $X = (X_1, X_2, \dots, X_p)$

where:

- Vector weights $w = [w_k]_{k=1,..,5}$ have: $w_k \geq 0$ and $\sum w_k = 1$
- Gauss distribution with mean μ_k, covariance matrix Σ_k:

$$N(\mu_k, \Sigma_k) = \frac{1}{\sqrt{(2\pi)^p |\Sigma|}} e^{-\frac{(X-\mu_k)^T \Sigma^{-1} (X-\mu_k)}{2}}$$

In the case X = (x, y) then N is a bivariate Gaussian distribution. With covariance matrix $\Sigma_k = \begin{bmatrix} \sigma_k^2 & 0 \\ 0 & \sigma_k^2 \end{bmatrix}$ (σ_k^2 is variance, assume that x and y have the same variance), we have:

$$N(\mu_k, \Sigma_k) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x-\mu_{k,x})^2 + (y-\mu_{k,y})^2}{2\sigma^2}}$$

Sampling algorithm:

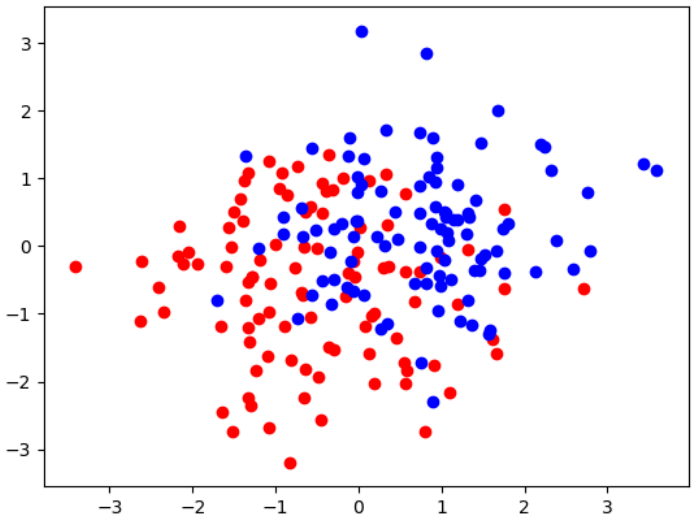
To sample from mixed Gaussian distribution, I use a sampling method as follow:

- Sample an indicator i from categorical distribution parametrized by vector weights w. Indicator i shows which Gaussian distribution should be sampled.
- Sample an sample (x, y) from Gaussian distribution $N(\mu_i, \Sigma_i)$

1.2. Result

I randomly initialize the mean values $\mu \in$ [-1, 0] for the first mixed distribution and $\mu \in$ [0, 1] for the second, the variance σ^2=1. The weight vector of each distribution is also initialized randomly.

The figure below visualize the two-class dataset



2. Draw the boundary decision for the sample dataset

2.1. Making decision without training from sampled dataset

First, I will discuss about how to draw the boundary decision without training a machine learning model from the generated data.

Because the dataset is generated from two mixed Gaussian distribution, we have already had the distribution of data in each class. It means that we know:

$$p(X \mid C_0) = \text{mixedGauss}_0(\text{X}) = \sum_{k=1}^{5} w_k^{(0)} N(\mu_k^{(0)}, \Sigma_k^{(0)})$$

$$p(X \mid C_1) = \text{mixedGauss}_1(\text{X}) = \sum_{k=1}^{5} w_k^{(1)} N(\mu_k^{(1)}, \Sigma_k^{(1)})$$

where $p(X \mid C_j)$ indicates that the distribution of data which belong to class C_j.

Now, given a data point X, we want to assign label for this data point. This work is to compute a posterior distribution which is the probability of class given the data point:

$$p(C_j \mid X) = \frac{p(X \mid C_j)p(C_j)}{p(X)} = \frac{p(X \mid C_j)p(C_j)}{\sum_{j \in \{0,1\}} p(X \mid C_j)p(C_j)} \quad \text{(Bayes's theorem)}$$

The decision theory for labeling a data point X will choose the class which the probability posterior is higher than the others (choose class j that $p(C_j \mid X) > p(C_t \mid X)$ with

$\forall t \neq j$). Note that we've already have $p(X \mid C_j)$ (j ∈ {0, 1}), the prior $p(C_j)$ can be estimated by the population of class C_j in dataset. Here, $p(C_0) = p(C_1) = \frac{100}{200} = 0.5$. Therefore, we can compute the posterior distribution.

In fact, we only need to compute $p(C_1 \mid \mathrm{X})$ and assign X to class 1 if $p(C_1 \mid \mathrm{X}) > 0.5$; assign to class 0 if otherwise. So the decision boundary can be drawn from the function $p(C_1 \mid \mathrm{X}) = 0.5$

The family of this model is called "Generative Model". It means that the model is possible to generate synthetic data points in the input space.

2.2.Training a classifier from sampled dataset

When we only observe the dataset, we can build a classifier to classify data points. Here, I will test with some state-of-the-art classification model including:

- SVM with the linear kernel (Linear SVM)
- SVM with the gauss kernel (RBF SVM)
- Neural Network
- Random Forest

For SVM, the threshold for making decision will be 0 and will be 0.5 with Neurel Network and Random Forest.

2.3. Result

To draw the decision boundary, I divide the rectangle [x_min, x_max] x [y_min, y_max] into a grid with step size is 0.02. After I compute the probability for each point in grid and assign this point to color blue if the probability > 0.5 and assign to color read if otherwise.

This figure 2 shows the decision boundary of five models. The accuracy on the sampled dataset is on the lower right of figure. One can see that Random Forest model is very fit to data. The decision boundary of generative model is approximately a linear line.

2.4. Discussion

- The complexity model like Random Forest sometimes is overfitted when the training data is small. Therefore, it can not generalize for the new data. If the new data is sampled from the mixed Gaussian distribution, I think the generative model will have a good generalization on the overall data.
- In this problem, we have a knowledge about the distribution of data, so the generative model doesn't need to learn a distribution of data. That why . In fact, we only observe the data, then we often assume that $p(X \mid C_j)$ follows a specific distribution and the traning process is to estimate the parameters of this distribution from dataset.

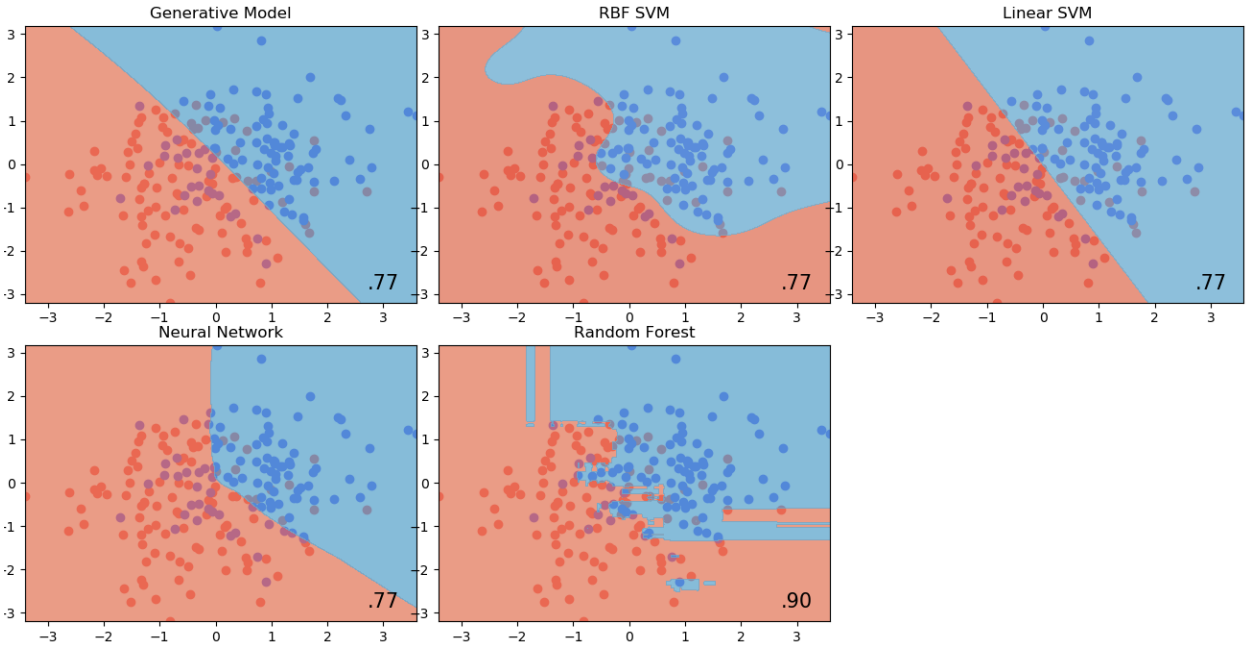


Figure 2: Decision boundary