Gaussian Discriminant Analysis

1 Generative Learning Algorithms

A discriminative learning algorithm learns $P(y \mid x)$ and relates x to y directly (like perceptron). In contrast, a generative learning algorithm learns $P(x \mid y)$ where x are features and y is our class. This asks the question "Given some sample in class y, what are the features?" A generative learning algorithm will also learn $P(y)$ which is called the class prior. When adding a new sample, using Bayes' rule we can then perform

$$P(y = 1 \mid x) = \frac{P(x \mid y = 1)P(y = 1)}{P(x)}$$

where

$$P(x) = P(x \mid y = 1)P(y = 1) + P(x \mid y = 0)P(y = 0)$$

for an example of binary classification (our classes being 0 and 1).

2 Multivariate Gaussian

A multivariate Gaussian is the generalization of the familiar bell-shaped curve over a 1-dimensional random variable to a vector value random variables.

If $z \sim \mathcal{N}(\vec{\mu}, \Sigma)$ where $\vec{\mu}$ is some mean vector and Σ is some covariance matrix and $z \in \mathbb{R}^n$, then $\vec{\mu} \in \mathbb{R}^n$ and $\Sigma \in \mathbb{R}^{n \times n}$. We can then say

$$\mathrm{E}[z] = \mu$$

$$\mathrm{Cov}(z) = \mathrm{E}\left[(z - \mu)(z - \mu)^T\right]$$

And the probability density function is

$$\mathcal{N}(x; \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}}} |\Sigma|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(x - \mu)^T \Sigma^{-1}(x - \mu)\right)$$

3 Discriminant Analysis

Suppose $x \in \mathbb{R}^n$, which means our features are any real numbers in n-dimensions. A key assumption in Gaussian discriminant analysis (GDA) is that $P(x \mid y)$ is Gaussian. As a result, we derive the probability densities as

$$P(x \mid y = 0) = \frac{1}{(2\pi)^{\frac{n}{2}}} |\Sigma|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(x - \mu_0)^T \Sigma^{-1}(x - \mu_0)\right)$$

$$P(x \mid y=1) = \frac{1}{(2\pi)^{\frac{n}{2}}} |\Sigma|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(x-\mu_1)^T \Sigma^{-1}(x-\mu_1)\right)$$

Typically, the positive and negative classes will have different means but the same covariance matrix. Next, we model $P(y)$. In our case, y is simply a Bernoulli random variable that takes on the values of 0 or 1. So

$$P(y) = \phi^y (1-\phi)^{1-y}$$

Our parameters are $\mu_0, \mu_1, \Sigma, \phi$, and we're given a training set $\left\{x^{(i)}, y^{(i)}\right\}_{i=1}^m$. In order to fit these parameters, we need to maximize the joint likelihood which is defined as

$$\mathcal{L}(\phi, \mu_0, \mu_1, \Sigma) = \prod_{i=1}^m P\left(x^{(i)}, y^{(i)}; \phi, \mu_0, \mu_1, \Sigma\right) = \prod_{i=1}^m P\left(x^{(i)} \mid y^{(i)}\right) P\left(y^{(i)}\right)$$

We can use maximum likelihood estimation then. We need to choose parameters such that the log likelihood is maximized. The value of ϕ that maximizes this is

$$\phi = \frac{1}{m}\sum_{i=1}^m y^{(i)} = \frac{1}{m}\sum_{i=1}^m \mathbb{1}\left(y^{(i)} = 1\right)$$

where $\mathbb{1}(Q)$ is the indicator function that returns 1 if Q is true and 0 if false. We derive this from the idea that the likelihood of some sample being classified as the positive class is the number of positive class samples divided by the total number of samples. With this we can determine that

$$\mu_0 = \frac{\sum_{i=1}^m \mathbb{1}\left(y^{(i)} = 0\right) x^{(i)}}{\sum_{i=1}^m \mathbb{1}\left(y^{(i)} = 0\right)}$$

and

$$\mu_1 = \frac{\sum_{i=1}^m \mathbb{1}\left(y^{(i)} = 1\right) x^{(i)}}{\sum_{i=1}^m \mathbb{1}\left(y^{(i)} = 1\right)}$$

since we compute the mean as the sum of the feature vectors for samples with a specified class divided by the total number of samples with that class. And then our covariance matrix is defined as

$$\Sigma = \frac{1}{m}\sum_{i=1}^m \left(x^{(i)} - \mu_{y^{(i)}}\right)\left(x^{(i)} - \mu_{y^{(i)}}\right)^T$$

per our definition of $\text{Cov}[z]$.

Based on our findings of how to maximize the parameters, how do we predict the classification of a new sample? We would choose

$$\arg\max_y P(y \mid x) = \arg\max_y \frac{P(x \mid y)P(y)}{P(x)}$$

which uses the arg max to determine the value of y that maximizes the function which would be either 0 or 1. Since the denominator $P(x)$ is a constant and independent of y, we can simplify this to

$$\arg\max_y P(x \mid y)P(y)$$