Chapter *3* Generative Models

Siheng Zhang
zhangsiheng@cvte.com

September 11, 2020

This part corresponds to **Chapter 24, 31 in UML, Chapter ? in PRML, Chapter ? in PGM**, and mainly answers the following questions:

-

-

Contents

1 Naive Bayes

Recall that the Bayes optimal classifier (*in Chapter 1, Ex6*) is:

$$h_{\text{Bayes}}(\mathbf{x}) = \arg \max_{y \in \{0,1\}} p(Y = y | X = \mathbf{x})$$

To describe the posterior probability function we need 2^d parameters, this implies that the number of examples we need grows exponentially with the number of features. To avoid this problem, we assume that given the label, the features are independent of each other, i.e.,

$$\mathcal{P}(X = \mathbf{x} | Y = y) = \prod_{i=1}^{d} p(X_i = \mathbf{x}_i | Y = y)$$

Together with Bayes' rule, the Bayes optimal classifier can be simplified as:

$$h_{\text{Bayes}}(\mathbf{x}) = \arg \max_{y \in \{0,1\}} p(Y = y) \prod_{i=1}^{d} p(X_i = \mathbf{x}_i | Y = y) \tag{1}$$

Now the number of parameters we need to estimate is only $2d + 1$. When we also estimate the parameters using the maximum likelihood principle (see below), the resulting classifier is called the *Naive Bayes* classifier.

2 Density estimation

2.1 Parametric methods

2.2 Non-parametric methods

3 Bayesian Reasoning

4 PAC-Bayes

5 Generative models

5.1 GMM (Gaussian mixture models)

5.2 HMM (Hidden Markov models)

5.3 Naive Bayes to linear discriminant models

The usual assumption in Naive Bayes classifier is that each conditional probability $p(X = \mathbf{x} | Y = y)$ is a Gaussian distribution. Consider the binary classification task, denote the two conditional distribution as $\mathcal{N}(\mu_0, \Sigma_0)$, $\mathcal{N}(\mu_1, \Sigma_1)$, we will predict $h_{\text{Bayes}}(\mathbf{x}) = 1$ iff.

$$\frac{p(Y = 0)p(X = \mathbf{x} | Y = 0)}{p(Y = 1)p(X = \mathbf{x} | Y = 1)} > 1$$

$$\iff \log \frac{p(Y = 0)}{p(Y = 1)} + \log p(X = \mathbf{x} | Y = 0) - \log p(X = \mathbf{x} | Y = 1) > 0$$

$$\iff -\frac{1}{2}(\mathbf{x} - \mu_0)^\top \Sigma_0^{-1}(\mathbf{x} - \mu_0) + \frac{1}{2}(\mathbf{x} - \mu_1)^\top \Sigma_1^{-1}(\mathbf{x} - \mu_1) + \frac{1}{2}\log \frac{|\Sigma_1|}{|\Sigma_0|} + \log \frac{p(Y = 0)}{p(Y = 1)} > 0$$

$$\iff \frac{1}{2}\mathbf{x}^\top(\Sigma_1^{-1} - \Sigma_0^{-1})\mathbf{x} + (\mu_0^\top \Sigma_0^{-1} - \mu_1^\top \Sigma_1^{-1})\mathbf{x} + \underbrace{\frac{1}{2}(\mu_1^\top \Sigma_1^{-1}\mu_1 - \mu_0^\top \Sigma_0^{-1}\mu_0) + \frac{1}{2}\log \frac{|\Sigma_1|}{|\Sigma_0|} + \log \frac{p(Y = 0)}{p(Y = 1)}}_{b} > 0$$

which is a quadratic discriminant function.

Further, if we assume that $\Sigma_0 = \Sigma_1 = \Sigma$, the classifier can be simplified to be a linear discriminant function $\mathbf{w} \cdot \mathbf{x} + b$, with $\mathbf{w} = (\mu_0 - \mu_1)^\top \Sigma_1^{-1}$ and $b = \frac{1}{2}(\mu_1^\top \Sigma^{-1}\mu_1 - \mu_0^\top \Sigma^{-1}\mu_0) + \log \frac{p(Y=0)}{p(Y=1)}$. If the prior probability is equal, namely $p(Y = 0) = p(Y = 1)$, the bias term can be further simplified.

5.4 *v.s.* discriminant models

In generative approaches, it is assumed that the underlying distribution over the data has a specific parametric form and the goal is to estimate the parameters of the model. But in discriminative approaches, the goal is rather to learn an accurate predictor directly.

Of course, if we succeed in learning the underlying distribution accurately, prediction from the Bayes optimal classifier is reliable. The problem is that, it is usually more difficult to learn the underlying distribution than to learn an accurate predictor. This was phrased by Vladimir Vapnik:

> *"When solving a given problem, try to avoid a more general problem as an intermediate step."*

However, in some situations, it is reasonable to adopt the generative models. Sometimes it is easier (computationally) to estimate the parameters of the model than to learn a discriminative predictor. Additionally, in some cases we do not have a specific task at hand but rather would like to use the data at a later time.

Modern generative models have another big goal, that is to 'generate' (sample from the underlying distribution) data like that in the real world. The intuition behind this approach follows a famous quote from Richard Feynman:

> *"What I cannot create, I do not understand."*

6 Exercises and solutions

Chapter 4. Linear models, perceptron, MLP, deep learning, Generalization bounds on deep learning.