CS281A/Stat241A: Statistical Learning Theory

Lecture 12: Classification Continued (10/7/02)

Lecturer: Michael I. Jordan *Scribes: James Van Campen and Laura Parada*

1 Introduction

This lecture illustrated parameter estimation in generative and discriminative classification models using the maximum liklihood method.

2 Generative Case

2.1 Binary Classification

Given the data, $\mathcal{D} = \{(X_n, Y_n)\}_{n=1}^N$, we want to estimate the parameters, $\theta = (\pi, \mu_0, \mu_1, \Sigma)$. We start by taking logarithms.

$$\ell(\theta; \mathcal{D}) = \log\{\prod_{n=1}^N p(y_n|\pi)p(x_n|y_n, \mu_0, \mu_1, \Sigma)\} = \sum_{n=1}^N \log p(y_n|\pi) + \sum_{n=1}^N \log p(x_n|y_n, \mu_0, \mu_1, \Sigma)$$

This decouples π from μ_0, μ_1, and Σ, allowing us to find $\hat{\pi}_{ML}$. To do this we take the derivative w.r.t. π of the log of $p(y_n|\pi) = \pi^{y_n}(1-\pi)^{1-y_n}$ and set it to zero. Solving for π yields $\hat{\pi}_{ML} = \frac{1}{N}\sum_{n=1}^N y_n$ - which is the sample proportion.

Assumption: $x_n|y_n \sim$ Gaussian.

We have $p(x_n|y_n, \mu_o, \mu_1, \Sigma) = [p(x_n|Y_n = 1, \mu_1, \Sigma)]^{y_n}[p(x_n|Y_n = 0, \mu_0, \Sigma)]^{1-y_n}$. Taking logarithms yields, $y_n \log p(x_n|Y_n = 1, \mu_1, \Sigma) + (1 - y_n) \log p(x_n|Y_n = 0, \mu_0, \Sigma)$. Then we take derivatives w.r.t. to μ_0 and μ_1, and set them to zero. Solving for μ_0 and μ_1 yields

$$\hat{\mu}_{0_{ML}} = \frac{\sum_{n=1}^N (1-y_n)x_n}{\sum_{n=1}^N (1-y_n)} \qquad \text{and} \qquad \hat{\mu}_{1_{ML}} = \frac{\sum_{n=1}^N y_n x_n}{\sum_{n=1}^N y_n}.$$

Note that $\hat{\mu}_{0_{ML}}$ and $\hat{\mu}_{1_{ML}}$ are weighted means, i.e. convex combinations of the x_n's. Also $\hat{\Sigma}_{ML}$ is a pooled estimate of the covariance matrix.

2.2 Multiway Classification

$$\pi_k = p(Y_n^k = 1|\pi) \qquad \text{where} \qquad \sum_{k=1}^K \pi_k = 1$$

Assume $p(x_n|Y_n^k = 1, \mu_k, \Sigma_k) \sim N(\mu_k, \Sigma_k)$ and $\Sigma_k = \Sigma \quad \forall \quad k$.

For the predictive problem where we know x_{new} but not y_{new} and we have the following.

$$p(Y_{new}^k = 1|x_{new}, \theta) = \frac{\pi_k p(x_{new}|Y_{new}^k = 1)}{\sum_{j=1}^K \pi_k p(x_{new}|Y_{new}^j = 1, \theta)}$$

$$p(Y_{new}^k = 1|x_{new}, \theta) = \frac{e^{\beta_k^T x_{new}}}{\sum_{j=1}^K e^{\beta_j^T x_{new}}} \text{ where } \beta_k \equiv \begin{bmatrix} -\mu_k^T \Sigma^{-1} \mu_k + \log \pi_k \\ \Sigma^{-1} \mu_k \end{bmatrix}$$

This is the multinomial logit (a.k.a. softmax) function. We estimate π_k and μ_k as follows,

$$\hat{\pi}_{k_{ML}} = \frac{1}{N} \sum_{n=1}^N y_n^k \qquad \text{and} \qquad \hat{\mu}_{k_{ML}} = \frac{\sum_{n=1}^N y_n^k x_n}{\sum_{n=1}^N y_n^k}$$

Note that the $\hat{\mu}_k$'s are weighted sample sums, and $\hat{\Sigma}_{ML}$ is a pooled estimate of the covariance matrix.

3 Discriminative Case

3.1 Examples

• Logistic regression uses the logistic function.

$$p(Y_n = 1|x_n, \theta) = \frac{1}{1 + e^{-\theta^T x_n}}$$

• Probit regression is similar to logistic regression but it uses a cumulative Gaussian instead of a logistic function. It is used for noisy threshold models.

$$p(Y_n = 1|x_n, \theta) = \Phi(\theta^T, x_n) \text{ where } \Phi(z) = \int_{-\infty}^z e^{-\theta^T x_n}$$

$$Z \sim N(0, 1) \qquad \text{and} \qquad Y_n = \begin{cases} 1 & \text{if } \theta^T x_n > z \\ 0 & \text{otherwise} \end{cases}$$

• Noisy-OR model - see text figure 7.15. In this model $x_i \in \{0, 1\}$ and $y \in \{0, 1\}$ where

$$Y = \begin{cases} 1 & \text{if } (X_1 \wedge \neg Z_1) \vee (X_2 \wedge \neg Z_2) \vee \ldots \vee (X_n \wedge \neg Z_n) \\ 0 & \text{otherwise} \end{cases}$$

Define $\xi_i \equiv p(Z_i = 1)$. Then

$$p(Y = 0|x, \xi) = \prod_{i=1}^n p(Z_i = 1)^{x_i} = \prod_{i=1}^n \xi^{x_i} = \exp\left\{\sum_{i=1}^n x_i \log \xi_i\right\}$$

and

$$p(Y = 1|x, \xi) = 1 - \exp\left\{\sum_{i=1}^n x_i \log \xi_i\right\}$$

3.2 Parameter Estimation

The log liklihood for the discriminative classification problem is

$$\ell(\theta, \mathcal{D}) = \log \left\{ \prod_{n=1}^{N} p(y_n|x_n, \theta) \right\}.$$

$$\mu_n \equiv p(Y_n = 1|x_n, \theta) \qquad \text{and} \qquad 1 - \mu_n = p(Y_n = 0|x_n, \theta)$$

$$\ell(\theta, \mathcal{D}) = \log \left\{ \prod_{n=1}^{N} p(Y_n = 1|x_n, \theta)^{y_n} p(Y_n = 0|x_n, \theta)^{1-y_n} \right\} = \log \left\{ \prod_{n=1}^{N} \mu_n^{y_n} (1 - \mu_n)^{1-y_n} \right\}$$

$$\ell(\theta, \mathcal{D}) = \sum_{n=1}^{N} \{y_n \log \mu_n + (1 - y_n) \log(1 - \mu_n)\}$$

Then $\mu_n = logistic(\theta^T x_n) = \frac{1}{1+e^{\theta^T x_n}}$. We cannot solve for θ in closed form. Two methods for finding θ numerically are steepest descent and Newton-Raphson. Steepest descent is and online algorithm where $\hat{\theta}_{ML}^{(t+1)} = \hat{\theta}_{ML}^{(t)} + \rho(y_n - \mu_n^{(t)})x_n$ with $\mu_n^{(t)} \equiv \phi(\theta^{(t)T} x_n)$ and ρ is a step size. The Newton-Raphson algorithm used is iteratively reweighted least squares (IRLS). IRLS is a batch algorithm.