Classification *

Logistic regression

- Logistic function $p(x) = \frac{e^{\beta_0 + \beta_1 x}}{1 + e^{\beta_0 + \beta_1 x}}$ and we fit with maximum likelihood

$\Rightarrow$ logit $\Rightarrow$ log odds $\log\left(\frac{p(x)}{1-p(x)}\right) = \beta_0 + \beta_1 x$ (this is linear in x).

↳ $l(\beta_0, \beta_1) = \prod_{i:y_i=1} p(x_i) \prod_{i':y_{i'}=0} (1 - p(x_{i'}))$

Like linear regression, we can measure the accuracy of the coefficient with the z-statistic of β_1 is $\hat{\beta}_1 / SE(\beta_1)$ and large value (absolute) against the null $H_0: \beta_1 = 0$ which implies $p(x) = \frac{e^{\beta_0}}{1+e^{\beta_0}}$

- and multiple logistic regression $\log\left(\frac{p(x)}{1-p(x)}\right) = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p$ and $p(x) = \frac{e^{\beta_0 + \beta_1 x_1 + \dots + \beta_p x_p}}{1 + e^{\beta_0 + \beta_1 x_1 + \dots}}$

when there is correlation among the predictors, the result obtained by one predictor will be different from using multiple predictors (confounding).

Logistic for classification

- When we wish to classify more than two classes $\Rightarrow$ multinomial logistic regression.

For k classes, we select the k^{th} as baseline and

$$P_k(Y=k \mid X=x) = \frac{e^{\beta_{k0} + \beta_{k1} x_1 + \dots \beta_{kp} x_p}}{1 + \sum_{l=1}^{k-1} e^{\beta_{l0} + \beta_{l1} x_1 + \dots \beta_{lp} x_p}} \quad \text{for } k = 1, \dots k-1 \text{ and } P_k(Y=K \mid X=x) = \frac{1}{1 + \sum_{l=1}^{k-1} e^{\beta_{l0} + \dots \beta_{lp} x_p}}$$

and $\log\left(\frac{P_k(Y=k \mid X=x)}{P_k(Y=K \mid X=x)}\right) = \beta_{k0} + \beta_{k1} x_1 + \dots + \beta_{kp} x_p$

$\Rightarrow$ but for multinomial we can use an alternative $\Rightarrow$ the softmax. rather than selecting a baseline each class is treated symmetrically

$$P_k(Y=k \mid X=x) = \frac{e^{\beta_{k0} + \beta_{k1} x_1 + \dots + \beta_{kp} x_p}}{\sum_{l=1}^{K} e^{\beta_{l0} + \beta_{l1} x_1 + \dots + \beta_{lp} x_p}} \quad \text{and} \quad \log\left(\frac{P_k(Y=k \mid X=x)}{P_k(Y=k' \mid X=x)}\right) = (\beta_{k0} - \beta_{k'0}) + (\beta_{k1} - \beta_{k'1}) x_1 + \dots$$

Generative model classification (Umbrella LDA, QDA, NB)

in logistic we model $P_k(Y=k \mid X=x)$ using logistic function $\rightarrow$ alternative $\rightarrow$ model the distribution of the predictors X separately in each of the response classes. Then use Bayes' theorem to flip these around into estimates for $P_k(Y=k \mid X=x)$ $\Rightarrow$ why do that $\Rightarrow$ when substantial separation between classes the parameters of logistic regression are unstable.

- Let π_k the probability that a random observation comes from the k^{th} class
- $f_k(x) = P_k(X \mid Y=k)$

and then $P_k(Y=k \mid X=x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^{K} \pi_l f_l(x)}$

to approximate $f_k(x)$ we can use linear discriminant, quadratic discriminant and naive bayes

Linear discriminant

$p = 1$

$f_k(x)$ is assume normal, with shared variance among classes $\sigma_1^2 = \dots = \sigma_K^2$

$$f_k(x) = \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{1}{2\sigma_k^2}(x-\mu_k)^2\right)$$

The Bayes classifier involves assigning an observation $X = x$ to the class for which p_k is largest. It amounts to assigning the observation, to the class for which $\delta_k(x) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k)$ is largest

- $\hat{\mu}_k = \frac{1}{n_k} \sum_{i : y_i = k} x_i$
- $\hat{\sigma}^2 = \frac{1}{n-K} \sum_{k=1}^{K} \sum_{i : y_i = k} (x_i - \hat{\mu}_k)^2$
- $\hat{\pi}_k = n_k / n$

$p > 1$

multivariate gaussian, $N(\mu_k, \Sigma)$ with common cov matrix Σ

$$f(x) = \frac{1}{(2\pi)^{p/2} |\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)\right)$$

$$\delta_k(x) = x^T \Sigma^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k + \log \pi_k$$

- LDA outperform logistic regression when the normality assumption hold.
- To test the model we use a confusion matrix. The Bayes classifier (and by extension LDA) we use a threshold of 50% for cutoff. We can change the threshold to get the best model. The ROC displays the two types of error for all possible threshold. To compare classifier use the AUC (best possible is 1, random classifier is 0.5).

quadratic discriminant

Unlike LDA, $x \sim N(\mu_k, \Sigma_k)$ where Σ_k is a covariance matrix for the kth class.

$$\delta_k(x) = -\frac{1}{2} x^T \Sigma_k^{-1} x + x^T \Sigma_k^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma_k^{-1} \mu_k - \frac{1}{2} \log |\Sigma_k| + \log \pi_k$$

LDA is less flexible classifier, so lower variance, but if the assumption of common variance is off, there will be a higher bias. → few training → LDA
→ many training or ≠ matrix → QDA

Naive Bayes

The naive Bayes takes a different tack for estimating $f_1(x), \dots f_K(x)$. Instead of assuming a distribution, we make the assumption, that within the k^{th} class, the p predictors are independent

$f_k(x) = f_{k1}(x_1) \times f_{k2}(x_2) \times \dots \times f_{kp}(x_p)$ where f_{kj} is the density function of the jth predictor among observations in the k^{th} class. It introduces some bias but reduces variance.

$$Pr(Y = k \mid X = x) = \frac{\pi_k \times f_{k1}(x_1) \times \dots \times f_{kp}(x_p)}{\sum_{l=1}^{K} \pi_l \times f_{l1}(x_1) \times \dots \times f_{lp}(x_p)} \quad \text{for } k = 1, \dots, K.$$

to estimate f_{kj} →
- x_j is quantitative → $x_j \mid Y = k \sim N(\mu_{jk}, \sigma_{jk}^2)$ → or histogram, or kernel density estimator
- x_j is qualitative → count per classes.

NB good when p is large or N is small and we need to reduce the variance.