Logistic Regression

Yingming Li
yingming@zju.edu.cn

Data Science & Engineering Research Center, ZJU

16th April 2018

Adapted from slides provided by Prof. Jia Li.

Logistic Regression

Preserve linear classification boundaries.

- By the Bayes rule:

$$\hat{G}(x) = \arg\max_k Pr(G = k \mid X = x)$$

- Decision boundary between class k and l is determined by the equation:

$$Pr(G = k \mid X = x) = Pr(G = l \mid X = x).$$

- Divide both sides by $Pr(G = l \mid X = x)$ and take log. The above equation is equivalent to

$$\log \frac{Pr(G = k \mid X = x)}{Pr(G = l \mid X = x)} = 0.$$

Meaning of ξ_p

- Since we enforce linear boundary, we can assume

$$\log \frac{Pr(G = k \,|X = x)}{Pr(G = l \,|X = x)} = a_0^{(k,l)} + \sum_{j=1}^{P} a_j^{(k,l)} x_j.$$

- For logistic regression, there are restrictive relations between $a^{(k,l)}$ for different pairs of (k, l).

Assumptions

$$\log \frac{Pr(G=1\,|X=x)}{Pr(G=K\,|X=x)} = \beta_{10} + \beta_1^T x$$

$$\log \frac{Pr(G=2\,|X=x)}{Pr(G=K\,|X=x)} = \beta_{20} + \beta_2^T x$$

$$\vdots$$

$$\log \frac{Pr(G=K-1\,|X=x)}{Pr(G=K\,|X=x)} = \beta_{(K-1)0} + \beta_{K-1}^T x$$

- For any pair (k, l):

$$\log \frac{Pr(G = k \,|X = x)}{Pr(G = l \,|X = x)} = \beta_{k0} - \beta_{l0} + (\beta_k - \beta_l)^T x.$$

- Number of parameters: $(K - 1)(p + 1)$.
- Denote the entire parameter set by

$$\theta = \{\beta_{10}, \beta_1, \beta_{20}, \beta_2, \ldots, \beta_{(K-1)0}, \beta_{K-1}\}.$$

- The log ratio of posterior probabilities are called log-odds or logit transformations.

- Under the assumptions, the posterior probabilities are given by:

$$Pr(G = k \,|X = x) = \frac{\exp(\beta_{k0} + \beta_k^T x)}{1 + \sum_{l=1}^{K-1} \exp(\beta_{l0} + \beta_l^T x)}$$

for $k = 1, \ldots, K-1$

$$Pr(G = K \,|X = x) = \frac{1}{1 + \sum_{l=1}^{K-1} \exp(\beta_{l0} + \beta_l^T x)}.$$

- For $Pr(G = k \,|X = x)$ given above, obviously
  - Sum up to 1: $\sum_{k=1}^{K} Pr(G = k \,|X = x) = 1$.
  - A simple calculation shows that the assumptions are satisfied.

Comparison with LR on Indicators

- Similarities:
  - Both attempt to estimate $Pr(G = k \,|X = x)$.
  - Both have linear classification boundaries.
- Difference:
  - Linear regression on indicator matrix: approximate $Pr(G = k \,|X = x)$ by a linear function of x. $Pr(G = k \,|X = x)$ is not guaranteed to fall between 0 and 1 and to sum up to 1.
  - Logistic regression: $Pr(G = k \,|X = x)$ is a nonlinear function of x. It is guaranteed to range from 0 to 1 and to sum up to 1.

Fitting Logistic Regression Models

- Criteria: find parameters that maximize the conditional likelihood of G given X using the training data.
- Denote $p_k(x_i; \theta) = Pr(G = k \,|X = x_i; \theta)$.
- Given the first input x_1, the posterior probability of its class being g_1 is $Pr(G = g_1 \,|X = x_1)$.
- Since samples in the training data set are independent, the posterior probability for the N samples each having class g_i, $i = 1, 2, \ldots, N$, given their inputs $x_1, x_2, ..., x_N$ is:

$$\prod_{i=1}^{N} Pr(G = g_i \,|X = x_i).$$

- The conditional log-likelihood of the class labels in the training data set is

$$
\begin{aligned}
L(\theta) &= \sum_{i=1}^{N} \log Pr(G = g_i \,|X = x_i) \\
&= \sum_{i=1}^{N} \log p_{g_i}(x_i; \theta).
\end{aligned}
$$

Binary Classification

- For binary classification, if $g_i = 1$, denote $y_i = 1$; if $g_i = 2$, denote $y_i = 0$.
- Let $p_1(x;\theta) = p(x;\theta)$, then

$$p_2(x;\theta) = 1 - p_1(x;\theta) = 1 - p(x;\theta).$$

- Since $K = 2$, the parameters $\theta = \{\beta_{10}, \beta_1\}$.
  We denote $\beta = (\beta_{10}, \beta_1)^T$.

- If $y_i = 1$, i.e., $g_i = 1$,

$$\begin{aligned}\log p_{g_i}(x;\beta) &= \log p_1(x;\beta) \\ &= 1 \cdot \log p(x;\beta) \\ &= y_i \log p(x;\beta).\end{aligned}$$

If $y_i = 0$, i.e., $g_i = 2$,

$$\begin{aligned}\log p_{g_i}(x;\beta) &= \log p_2(x;\beta) \\ &= 1 \cdot \log(1-p(x;\beta)) \\ &= (1-y_i)\log(1-p(x;\beta)).\end{aligned}$$

Since either $y_i = 0$ or $1-y_i = 0$, we have

$$\log p_{g_i}(x;\beta) = y_i \log p(x;\beta) + (1-y_i)\log(1-p(x;\beta)).$$

- The conditional likelihood

$$L(\beta) = \sum_{i=1}^{N} \log p_{g_i}(x_i; \beta)$$
$$= \sum_{i=1}^{N} [y_i \log p(x_i; \beta) + (1-y_i) \log(1-p(x_i; \beta))]$$

- There $p + 1$ parameters in $\beta = (\beta_{10}, \beta_1)^T$.
- Assume a column vector form for β:

$$\beta = \begin{pmatrix} \beta_{10} \\ \beta_{11} \\ \beta_{12} \\ \vdots \\ \beta_{1,p} \end{pmatrix}.$$

- Here we add the constant term 1 to x to accommodate the intercept.

$$x = \begin{pmatrix} 1 \\ x_{,1} \\ x_{,2} \\ \vdots \\ x_{,p} \end{pmatrix}$$

- By the assumption of logistic regression model:

$$p(x;\beta) = Pr(G = 1 \,|X = x) = \frac{\exp(\beta^T x)}{1 + \exp(\beta^T x)}$$
$$1 - p(x;\beta) = Pr(G = 2 \,|X = x) = \frac{1}{1 + \exp(\beta^T x)}$$

- Substitute the above in $L(\beta)$:

$$L(\beta) = \sum_{i=1}^{N} \left[y_i \beta^T x_i - \log\left(1 + e^{\beta^T x_i}\right)\right]$$

- To maximize $L(\beta)$, we set the first order partial derivatives of $L(\beta)$ to zero.

$$\begin{aligned}\frac{\partial L(\beta)}{\beta_1 j} &= \sum_{i=1}^{N} y_i x_{ij} - \sum_{i=1}^{N} \frac{x_{ij} e^{\beta^T x_i}}{1 + e^{\beta^T x_i}} \\ &= \sum_{i=1}^{N} y_i x_{ij} - \sum_{i=1}^{N} p(x; \beta) x_{ij} \\ &= \sum_{i=1}^{N} x_{ij} (y_i - p(x_i; \beta))\end{aligned}$$

for all $j = 0, 1, ..., p$.

- In matrix form, we write

$$\frac{\partial L(\beta)}{\partial \beta} = \sum_{i=1}^{N} x_i(y_i - p(x_i; \beta)).$$

- To solve the set of $p + 1$ nonlinear equations $\dfrac{\partial L(\beta)}{\beta_1 j}$, $j = 0, 1, ..., p$, use the Newton-Raphson algorithm.
- The Newton-Raphson algorithm requires the second-derivatives or Hessian matrix:

$$\frac{\partial^2 L(\beta)}{\partial \beta \partial \beta^T} = -\sum_{i=1}^{N} x_i x_i^T p(x_i; \beta)(1 - p(x_i; \beta)).$$

- The element on the jth row and nth column is (counting from 0):

$$
\begin{aligned}
&\frac{\partial L(\beta)}{\partial \beta_{1j} \partial \beta_{1n}} \\
&= -\sum_{i=1}^{N} \frac{(1+e^{\beta^T x_i}) e^{\beta^T x_i} x_{ij} x_{in} - (e^{\beta^T x_i})^2 x_{ij} x_{in}}{(1+e^{\beta^T x_i})^2} \\
&= -\sum_{i=1}^{N} x_{ij} x_{in} p(x_i;\beta) - x_{ij} x_{in} p(x_i;\beta)^2 \\
&= -\sum_{i=1}^{N} x_{ij} x_{in} p(x_i;\beta)(1-p(x_i;\beta)).
\end{aligned}
$$

- Starting with β^{old}, a single Newton-Raphson update is

$$\beta^{new} = \beta^{old} - \left(\frac{\partial^2 L(\beta)}{\partial \beta \partial \beta^T} \right)^{-1} \frac{\partial L(\beta)}{\partial \beta},$$

where the derivatives are evaluated at β^{old}.

- The iteration can be expressed compactly in matrix form.
  - Let $\boldsymbol{y}$ be the column vector of y_i.
  - Let $\mathbf{X}$ be the $N \times (p+1)$ input matrix.
  - Let $\mathbf{p}$ be the N-vector of fitted probabilities with ith element $p(x_i; \beta^{old})$.
  - Let $\mathbf{W}$ be an $N \times N$ diagonal matrix of weights with i th element $p(x_i; \beta^{old})(1-p(x_i; \beta^{old}))$.
  - Then

$$\frac{\partial L(\beta)}{\partial \beta} = \mathbf{X}^T(\boldsymbol{y} - \boldsymbol{p})$$

$$\frac{\partial^2 L(\beta)}{\partial \beta \partial \beta^T} = -\mathbf{X}^T\mathbf{W}\mathbf{X}.$$

- The Newton-Raphson step is

$$\begin{aligned}\beta^{new} =& \beta^{old} + (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T(\boldsymbol{y}-\boldsymbol{p}) \\ =& (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}(\mathbf{X}\beta^{old} + \mathbf{W}^{-1}(\boldsymbol{y}-\boldsymbol{p})) \\ =& (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}\boldsymbol{z},\end{aligned}$$

  where $z \stackrel{\triangle}{=} \mathbf{X}\beta^{old} + \mathbf{W}^{-1}(\boldsymbol{y}-\boldsymbol{p})$.
- If $\boldsymbol{z}$ is viewed as a response and $\mathbf{X}$ is the input matrix, β new is the solution to a weighted least square problem:

$$\beta^{new} \leftarrow \arg\min_{\beta}(\boldsymbol{z} - \mathbf{X}\beta)^T\mathbf{W}(\boldsymbol{z} - \mathbf{X}\beta).$$

- Recall that linear regression by least square is to solve

$$\arg\min_{\beta}(\boldsymbol{z}-\mathbf{X}\beta)^T(\boldsymbol{z}-\mathbf{X}\beta).$$

- $\boldsymbol{z}$ is referred to as the adjusted response.
- The algorithm is referred to as iteratively reweighted least squares or IRLS.

Pseudo Code

1. $\mathbf{0} \rightarrow \beta$
2. Compute $\boldsymbol{y}$ by setting its elements to

$$y_i = \begin{cases} 1 & \text{if } g_i = 1 \\ 0 & \text{if } g_i = 2 \end{cases}, \quad i = 1, 2, ..., N.$$

3. Compute $\boldsymbol{p}$ by setting its elements to

$$p(x_i; \beta) = \frac{e^{\beta\,T x_i}}{1 + e^{\beta\,T x_i}} \quad i = 1, 2, ..., N.$$

4. Compute the diagonal matrix $\mathbf{W}$. The ith diagonal element is $p(x_i; \beta)(1 - p(x_i; \beta))$, $i = 1, 2, ..., N$.
5. $\boldsymbol{z} \leftarrow \mathbf{X}\beta + \mathbf{W}^{-1}(\boldsymbol{y} - \boldsymbol{p})$.
6. $\beta \leftarrow (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}\boldsymbol{z}$.
7. If the stopping criteria is met, stop; otherwise go back to step 3.

Computational Efficiency

- Since $\mathbf{W}$ is an $N \times N$ diagonal matrix, direct matrix operations with it may be very inefficient.
- A modified pseudo code is provided next.

1. $0 \to \beta$
2. Compute $\boldsymbol{y}$ by setting its elements to

$$y_i = \begin{cases} 1 & \text{if } g_i = 1 \\ 0 & \text{if } g_i = 2 \end{cases}, \quad i = 1, 2, ..., N.$$

3. Compute $\boldsymbol{p}$ by setting its elements to

$$p(x_i; \beta) = \frac{e^{\beta T x_i}}{1 + e^{\beta T x_i}} \quad i = 1, 2, ..., N.$$

4. Compute the $N \times (p+1)$ matrix $\tilde{\mathbf{X}}$ by multiplying the ith row of matrix $\mathbf{X}$ by $p(x_i; \beta)(1 - p(x_i; \beta))$, $i = 1, 2, ..., \mathbf{N}$:

$$\mathbf{X} = \begin{pmatrix} x_1^T \\ x_2^T \\ \cdots \\ x_N^T \end{pmatrix} \quad \tilde{\mathbf{X}} = \begin{pmatrix} p(x_1; \beta)(1 - p(x_1; \beta))x_1^T \\ p(x_2; \beta)(1 - p(x_2; \beta))x_2^T \\ \cdots \\ p(x_N; \beta)(1 - p(x_N; \beta))x_N^T \end{pmatrix}$$

5. $\beta \leftarrow \beta + (\mathbf{X}^T \tilde{\mathbf{X}})^{-1} \mathbf{X}^T (\boldsymbol{y} - \boldsymbol{p})$.
6. If the stopping criteria is met, stop; otherwise go back to step 3.

Example

Diabetes data set

- Input X is two dimensional. X_1 and X_2 are the two principal components of the original 8 variables.
- Class 1: without diabetes; Class 2: with diabetes.
- Applying logistic regression, we obtain

$$\beta = (0.7679, -0.6816, -0.3664)^T.$$

- The posterior probabilities are:

$$Pr(G = 1 \mid X = x) = \frac{e^{0.7679 - 0.6816 X_1 - 0.3664 X_2}}{1 + e^{0.7679 - 0.6816 X_1 - 0.3664 X_2}}$$

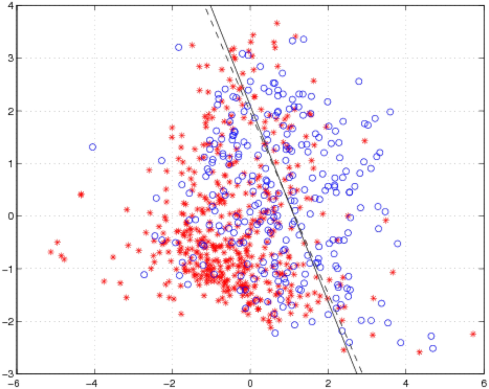
$$Pr(G = 2 \mid X = x) = \frac{1}{1 + e^{0.7679 - 0.6816 X_1 - 0.3664 X_2}}$$

- The classification rule is:

$$\hat{G}(x) = \begin{cases} 1 & 0.7679 - 0.6816 X_1 - 0.3664 X_2 \geqslant 0 \\ 2 & 0.7679 - 0.6816 X_1 - 0.3664 X_2 < 0 \end{cases}$$

Solid line: decision boundary obtained by logistic regression.
Dash line: decision boundary obtained by LDA.

- Within training data set classification error rate: 28.12%.
- Sensitivity: 45.9%.
- Specificity: 85.8%.



Multiclass Case $(K \geqslant 3)$

- When $K \geqslant 3$, β is a $(K-1)(p+1)$-vector:

$$\beta = \begin{pmatrix} \beta_{10} \\ \beta_1 \\ \beta_{20} \\ \beta_2 \\ \vdots \\ \beta_{(K-1)0} \\ \beta_{K-1} \end{pmatrix} = \begin{pmatrix} \beta_{10} \\ \beta_{11} \\ \vdots \\ \beta_{1p} \\ \beta_{20} \\ \vdots \\ \beta_{2p} \\ \vdots \\ \beta_{(K-1)0} \\ \vdots \\ \beta_{(K-1)p} \end{pmatrix}$$

- Let $\bar{\beta}_I = \begin{pmatrix} \beta_{I0} \\ \beta_I \end{pmatrix}$.
- The likelihood function becomes

$$
\begin{aligned}
L(\beta) &= \sum_{i=1}^{N} \log p_{g_i}(x_i; \beta) \\
&= \sum_{i=1}^{N} \log \left(\frac{e^{\bar{\beta}_{g_i}^T x_i}}{1 + \sum_{I=1}^{K-1} e^{\bar{\beta}_l^T x_i}} \right) \\
&= \sum_{i=1}^{N} \left[\bar{\beta}_{g_i}^T x_i - \log \left(1 + \sum_{I=1}^{K-1} e^{\bar{\beta}_l^T x_i} \right) \right]
\end{aligned}
$$

- Note: the indicator function $I(\cdot)$ equals 1 when the argument is true and 0 otherwise.
- First order derivatives:

$$\frac{\partial L(\beta)}{\partial \beta_{kj}} = \sum_{i=1}^{N} \left[I(g_i = k) x_{ij} - \frac{e^{\bar{\beta}_{g_i}^T x_i}}{1 + \sum_{I=1}^{K-1} e^{\bar{\beta}_l^T x_i}} \right]$$

$$= \sum_{i=1}^{N} x_{ij} (I(g_i = k) - p_k(x_i; \beta))$$

- Second order derivatives:

$$
\begin{aligned}
&\frac{\partial^2 L(\beta)}{\partial \beta_{kj} \partial \beta_{mn}} \\
&= \sum_{l=1}^{N} x_{ij} \cdot \frac{1}{(1 + \sum_{l=1}^{K-1} e^{\bar{\beta}_I^T x_i})^2} \cdot \\
&\quad \left[-e^{\bar{\beta}_k^T x_i} I(k = m) x_{in} \left(1 + \sum_{l=1}^{K-1} e^{\bar{\beta}_I^T x_i} \right) + e^{\bar{\beta}_k^T x_i} e^{\bar{\beta}_m^T x_i} x_{in} \right] \\
&= \sum_{i=1}^{N} x_{ij} x_{in} (-p_k(x_i; \beta) I(k = m) + p_k(x_i; \beta) p_m(x_i; \beta)) \\
&= -\sum_{i=1}^{N} x_{ij} x_{in} p_k(x_i; \beta) [I(k = m) - p_m(x_i; \beta)].
\end{aligned}
$$

- Matrix form.
  - $\boldsymbol{y}$ is the concatenated indicator vector of dimension $N \times (K{-}1)$.

$$\boldsymbol{y} = \begin{pmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \\ \vdots \\ \boldsymbol{y}_{K-1} \end{pmatrix} \quad \boldsymbol{y}_k = \begin{pmatrix} l(g_1 = k) \\ l(g_2 = k) \\ \vdots \\ l(g_N = k) \end{pmatrix}$$

$$1 \leqslant k \leqslant K - 1$$

  - $\boldsymbol{p}$ is the concatenated vector of fitted probabilities of dimension $N \times (K{-}1)$.

$$\boldsymbol{p} = \begin{pmatrix} \boldsymbol{p}_1 \\ \boldsymbol{p}_2 \\ \vdots \\ \boldsymbol{p}_{K-1} \end{pmatrix} \quad \boldsymbol{p}_k = \begin{pmatrix} p_k(x_1;\beta) \\ p_k(x_2;\beta) \\ \vdots \\ p_k(x_N;\beta) \end{pmatrix}$$

$$1 \leqslant k \leqslant K - 1$$

- $\tilde{\mathbf{X}}$ is an $N(K-1) \times (p+1)(K-1)$ matrix:

$$\tilde{\mathbf{X}} = \begin{pmatrix} \mathbf{X} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{X} & \cdots & \mathbf{0} \\ \cdots & \cdots & \cdots & \cdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{X} \end{pmatrix}$$

- Matrix $\mathbf{W}$ is an $N(K-1) \times N(K-1)$ square matrix:

$$\mathbf{W} = \begin{pmatrix} \mathbf{W}_{11} & \mathbf{W}_{12} & \cdots & \mathbf{W}_{1(K-1)} \\ \mathbf{W}_{21} & \mathbf{W}_{22} & \cdots & \mathbf{W}_{2(K-1)} \\ \cdots & \cdots & \cdots & \cdots \\ \mathbf{W}_{(K-1),1} & \mathbf{W}_{(K-1),2} & \cdots & \mathbf{W}_{(K-1),(K-1)} \end{pmatrix}$$

- Each submatrix $\mathbf{W}_{km}$, $1 \leqslant k, \ m \leqslant K-1$, is an $N \times N$ diagonal matrix.
- When $k = m$, the i th diagonal element in $\mathbf{W}_{kk}$ is $p_k(x_i; \beta^{old})(1-p_k(x_i; \beta^{old}))$.
- When $k \neq m$, the i th diagonal element in $\mathbf{W}_{km}$ is $-p_k(x_i; \beta^{old})p_m(x_i; \beta^{old})$.

- Similarly as with binary classification

$$\frac{\partial L(\beta)}{\partial \beta} = \tilde{\mathbf{X}}^T(\boldsymbol{y} - \boldsymbol{p})$$

$$\frac{\partial^2 L(\beta)}{\partial \beta \partial \beta^T} = -\tilde{\mathbf{X}}^T \mathbf{W} \tilde{\mathbf{X}}.$$

- The formula for updating β^{new} in the binary classification case holds for multiclass.

$$\beta^{new} = (\tilde{\mathbf{X}}^T \mathbf{W} \tilde{\mathbf{X}})^{-1} \tilde{\mathbf{X}}^T \mathbf{W} \boldsymbol{z},$$

where $\boldsymbol{z} \stackrel{\triangle}{=} \tilde{\mathbf{X}} \beta^{old} + \mathbf{W}^{-1}(\boldsymbol{y} - \boldsymbol{p})$. Or simply:

$$\beta^{new} = \beta^{old} + (\tilde{\mathbf{X}}^T \mathbf{W} \tilde{\mathbf{X}})^{-1} \tilde{\mathbf{X}}^T (\boldsymbol{y} - \boldsymbol{p}).$$

Computation Issues

- Initialization: one option is to use $\beta = 0$.
- Convergence is not guaranteed, but usually is the case.
- Usually, the log-likelihood increases after each iteration, but overshooting can occur.
- In the rare cases that the log-likelihood decreases, cut step size by half.

Connection with LDA

- Under the model of LDA:

$$\begin{aligned}
&\log \frac{Pr(G=k\,|X=x)}{Pr(G=K\,|X=x)} \\
=&\log \frac{\pi_k}{\pi_k} - \frac{1}{2}(\mu_k+\mu_K)^T \sum^{-1}(\mu_k-\mu_K) \\
&+ x^T \sum^{-1}(\mu_k-\mu_K) \\
=&a_{k0} + a_k^T x.
\end{aligned}$$

- The model of LDA satisfies the assumption of the linear logistic model.
- The linear logistic model only specifies the conditional distribution $Pr(G=k\,|X=x)$. No assumption is made about $Pr(X)$.

- The LDA model specifies the joint distribution of X and G. $Pr(X)$ is a mixture of Gaussians:

$$Pr(x) = \sum_{k=1}^{K} \pi_k \phi \left(X; \mu_k, \sum \right).$$

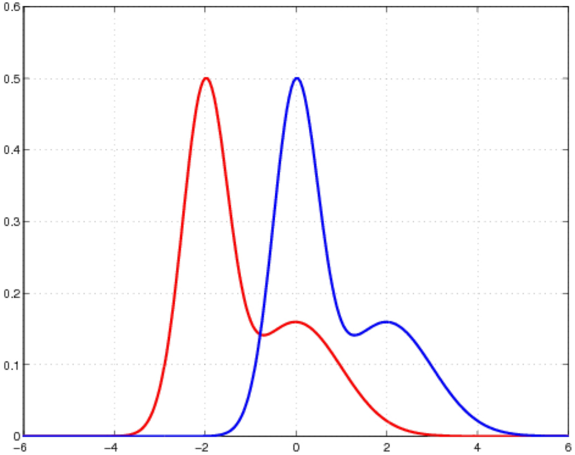
where ϕ is the Gaussian density function.

- Linear logistic regression maximizes the conditional likelihood of G given X: $Pr(G = k \,|X = x)$.
- LDA maximizes the joint likelihood of G and X: $Pr(X = x, G = k)$.

- If the additional assumption made by LDA is appropriate, LDA tends to estimate the parameters more efficiently by using more information about the data.
- Samples without class labels can be used under the model of LDA.
- LDA is not robust to gross outliers.
- As logistic regression relies on fewer assumptions, it seems to be more robust.
- In practice, logistic regression and LDA often give similar results.

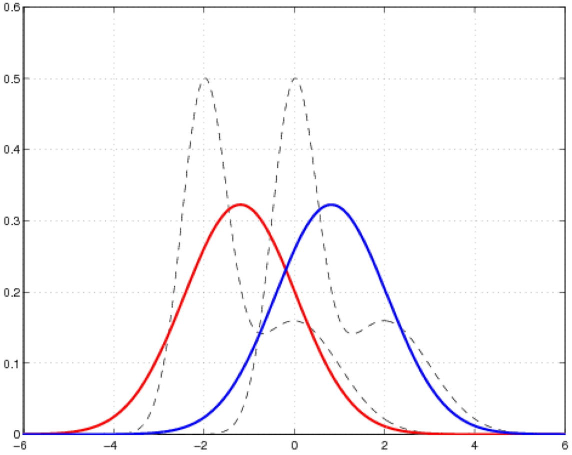
Simulation

- Assume input X is 1-D.
- Two classes have equal priors and the class-conditional densities of X are shifted versions of each other.
- Each conditional density is a mixture of two normals:
  - Class 1 (red): $0.6N\left(-2, \frac{1}{4}\right) + 0.4N(0, 1)$.
  - Class 2 (blue): $0.6N\left(0, \frac{1}{4}\right) + 0.4N(2, 1)$.
- The class-conditional densities are shown below.


0.6
0.5
0.4
0.3
0.2
0.1
0
−6
−4
−2
0
2
4
6

LDA Result

- Training data set: 2000 samples for each class.
- Test data set: 1000 samples for each class.
- The estimation by LDA: $\hat{\mu}_1 = -1.1948$, $\hat{\mu}_2 = 0.8224$, $\hat{\sigma}_2 = 1.5268$. Boundary value between the two classes is $(\hat{\mu}_1 + \hat{\mu}_2)/2 = -0.1862$.
- The classification error rate on the test data is 0.2315.
- Based on the true distribution, the Bayes (optimal) boundary value between the two classes is -0.7750 and the error rate is 0.1765.


0.6
0.5
0.4
0.3
0.2
0.1
0
−6
−4
−2
0
2
4
6

Logistic Regression Result

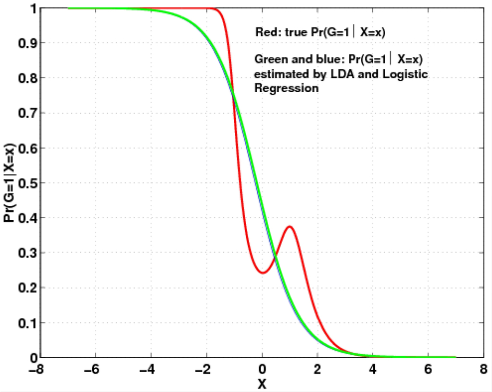
- Linear logistic regression obtains

$$\beta = (-0.3288, -1.3275)^T.$$

  The boundary value satisfies $-0.3288-1.3275X = 0$, hence equals -0.2477.
- The error rate on the test data set is 0.2205.
- The estimated posterior probability is:

$$Pr(G = 1 \mid X = x) = \frac{e^{-0.3288-13275x}}{1 + e^{-0.3288-1.3275x}}.$$

The estimated posterior probability $Pr(G = 1 \,|X = x)$ and its true value based on the true distribution are compared in the graph below.



Thank you!