# Modern Machine Learning
# Linear Methods for Classification — Logistic Regression

Kenneth E. Barner
Department of Electrical and Computer Engineering
University of Delaware

# Linear Methods for Classification

**Problem:** Classification is the prediction of qualitative responses

Consider the binary classification problem

**Examples:**

- Email: Spam/Not Spam
- Tumor: Benign/Malignant

Output $Y \in \{0,1\}$ $\begin{cases} 0 \coloneqq \text{Negative class (e.g. Not Spam)} \\ 1 \coloneqq \text{Positive class (e.g. Spam)} \end{cases}$

**Objective:** Divide the input space into a collection of regions labeled according to classification. Linear methods for classification result in linear decision boundaries

# Linear Methods for Classification

**Assumption:** Suppose there are $K$ classes and the fitted linear model for the $k^{\text{th}}$ indicator response variable is

$$\hat{f}_k(\boldsymbol{x}) = \hat{\beta}_{k0} + \widehat{\boldsymbol{\beta}}_k^T \boldsymbol{x} \quad k = 1,2,\dots,K$$

The decision boundary between classes $k$ and $l$ is the set of points for which $\hat{f}_k(\boldsymbol{x}) = \hat{f}_l(\boldsymbol{x})$

$$\{\boldsymbol{x}: (\hat{\beta}_{k0} - \hat{\beta}_{l0}) + (\widehat{\boldsymbol{\beta}}_k^T - \widehat{\boldsymbol{\beta}}_l^T)\boldsymbol{x} = 0\}$$

**Note:** The decision boundaries are a set of hyper planes $\Rightarrow$the input space is divided into regions with piecewise hyper plane decision boundaries

- This regression approach is within the class of methods that model discrimination functions for each class $\delta_k(\boldsymbol{x})$
  - $\boldsymbol{x}$ is classified to the class with the largest value for its discriminant function
  - Posterior probability models $\Pr(G = k|\boldsymbol{X} = \boldsymbol{x})$ are in this class
  - If $\delta_k(\boldsymbol{x})$ or $\Pr(G = k|\boldsymbol{X} = \boldsymbol{x})$ are linear in $\boldsymbol{x}$, then the decision boundaries are linear

**Notation:** $G(\boldsymbol{x})$ is the class predictor

**Logistic Regression:** Assume a Bernoulli class model:

$$\Pr(G = 1) = p \text{ and } \Pr(G = 0) = 1 - p, p \in [0,1]$$

Consider the ratio of the class probabilities transformed by the monotonic log function

*logit* transformation: $logit(p) = \log\left(\frac{p}{1-p}\right)$

Equate this to a linear function to establish linear boundaries

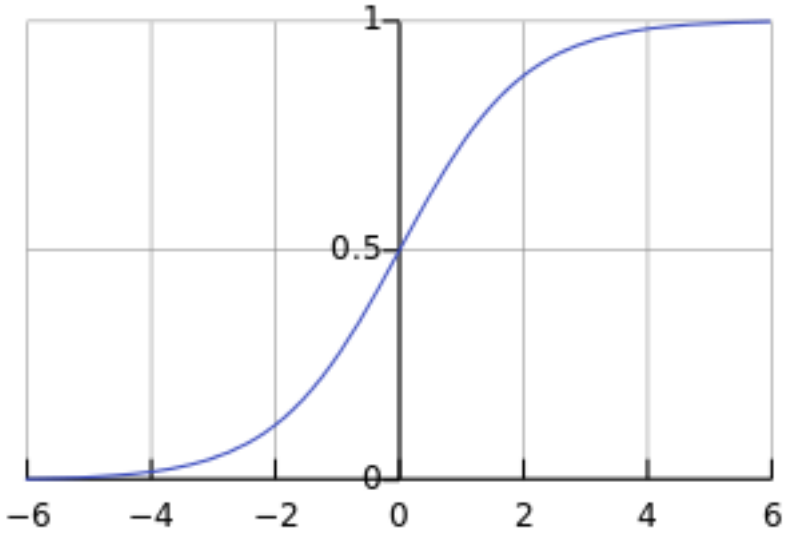
$$\text{logit}\big(\Pr(G = 1|\boldsymbol{X} = \boldsymbol{x})\big) = \log\left(\frac{\Pr(G = 1|\boldsymbol{X} = \boldsymbol{x})}{1 - \Pr(G = 1|\boldsymbol{X} = \boldsymbol{x})}\right) = \log\left(\frac{p}{1-p}\right) = \boldsymbol{\beta}^T\boldsymbol{x}$$

Therefore

$$\Pr(G = 1|\boldsymbol{X} = \boldsymbol{x}) = p = \frac{e^{\boldsymbol{\beta}^T\mathbf{x}}}{1 + e^{\boldsymbol{\beta}^T\mathbf{x}}} = \frac{1}{1 + e^{-\boldsymbol{\beta}^T\mathbf{x}}}$$

$$\Pr(G = 0|\boldsymbol{X} = \boldsymbol{x}) = 1 - p = \frac{1}{1 + e^{\boldsymbol{\beta}^T\mathbf{x}}}$$

The function $f(x) = \frac{e^x}{1+e^x} = \frac{1}{1+e^{-x}} \in [0,1]$ is called the logistic function




$f(\boldsymbol{\beta}^T\mathbf{x}) \to 0$ when $\boldsymbol{\beta}^T\mathbf{x} \to -\infty$
$f(\boldsymbol{\beta}^T\mathbf{x}) \to 1$ when $\boldsymbol{\beta}^T\mathbf{x} \to \infty$

To consider multiple observations, let:

$$y_i = 1 \text{ when } G_i = 1 \text{ and } y_i = 0 \text{ when } G_i = 0$$

Then in this Bernoulli case,

$$P(Y = y_i) = p^{y_i}(1-p)^{1-y_i}, \text{where } y_i \in \{0,1\}$$

If we have $n$ iid observations, the likelihood, as a function of $\boldsymbol{\beta}$, is

$$L(\boldsymbol{\beta}) = \prod_{i=1}^{n} p^{y_i}(1-p)^{1-y_i},$$

$$= \prod_{i=1}^{n} p(\mathbf{x}_i, \boldsymbol{\beta})^{y_i}\left(1 - p(\mathbf{x}_i, \boldsymbol{\beta})\right)^{1-y_i}$$

where $p = p(\mathbf{x}_i, \boldsymbol{\beta}) = \frac{e^{\boldsymbol{\beta}^T \mathbf{x}_i}}{1+e^{\boldsymbol{\beta}^T \mathbf{x}_i}}$

**Optimization Goal:** Determine $\boldsymbol{\beta}$ that maximizes the likelihood function $L(\boldsymbol{\beta})$

Equivalently, determine $\boldsymbol{\beta}$ that maximizes the negative log-likelihood function $NLL(\boldsymbol{\beta})$

$$NLL(\boldsymbol{\beta}) = -\sum_{i=1}^{n} \log\left(p(\mathbf{x}_i, \boldsymbol{\beta})^{y_i}\left(1 - p(\mathbf{x}_i, \boldsymbol{\beta})\right)^{1-y_i}\right)$$

$$= -\sum_{i=1}^{n} y_i \log\left(p(\mathbf{x}_i, \boldsymbol{\beta})\right) + (1 - y_i)\log\left(1 - p(\mathbf{x}_i, \boldsymbol{\beta})\right)$$

where, $p(\mathbf{x}_i, \boldsymbol{\beta}) = \frac{1}{1+e^{-\boldsymbol{\beta}^T \mathbf{x}_i}}$

To maximize $NLL(\boldsymbol{\beta})$, set its derivative to zero:

$$\frac{\partial NLL(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = -\sum_{i=1}^{n} \mathbf{x}_i \left(y_i - p(\mathbf{x}_i, \boldsymbol{\beta})\right) \quad \text{(Homework problem)}$$

**Observation:** the derivative is nonlinear in $\boldsymbol{\beta}$

**Solution:** utilize Gradient Descent or Newton-Raphson algorithm

**Gradient Descent:** Recall that gradient descent update:

$$\boldsymbol{\beta}^{k+1} = \boldsymbol{\beta}^{k} - \alpha \left.\frac{\partial J(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}}\right|_{\boldsymbol{\beta}^k}$$

# Linear Methods for Classification — Logistic Regression

Therefore the gradient descent updates for logistic regression take the following from

$$\boldsymbol{\beta}^{k+1} = \boldsymbol{\beta}^{k} - \alpha \left. \frac{\partial NLL(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} \right|_{\boldsymbol{\beta}^{k}}$$

$$= \boldsymbol{\beta}^{k} - \alpha \sum_{i=1}^{n} \mathbf{x}_i \left( p\left(\mathbf{x}_i, \boldsymbol{\beta}^{k}\right) - y_i \right)$$

**Newton-Raphson algorithm:** The Newton-Raphson updates have the following form

$$\boldsymbol{\beta}^{k+1} = \boldsymbol{\beta}^{k} - \left( \left. \frac{\partial^2 J(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^{T}} \right|_{\boldsymbol{\beta}^{k}} \right)^{-1} \left. \frac{\partial J(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} \right|_{\boldsymbol{\beta}^{k}}$$

# Linear Methods for Classification — Logistic Regression

Solving for the second derivative yields

$$\frac{\partial^2 NLL(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^T} = \sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^T \left( p(\mathbf{x}_i, \boldsymbol{\beta})(1 - p(\mathbf{x}_i, \boldsymbol{\beta})) \right)$$

Therefore the Newton Raphson update for logistic regression is

$$\begin{aligned} \boldsymbol{\beta}^{k+1} &= \boldsymbol{\beta}^k - \left( \frac{\partial^2 J(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^T} \bigg|_{\boldsymbol{\beta}^k} \right)^{-1} \frac{\partial J(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} \bigg|_{\boldsymbol{\beta}^k} \\ &= \boldsymbol{\beta}^k - \left( \sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^T \left( p\left(\mathbf{x}_i, \boldsymbol{\beta}^k\right) \left(1 - p\left(\mathbf{x}_i, \boldsymbol{\beta}^k\right)\right) \right) \right)^{-1} \left( \sum_{i=1}^{n} \mathbf{x}_i \left( p\left(\mathbf{x}_i, \boldsymbol{\beta}^k\right) - y_i \right) \right) \end{aligned}$$

# Linear Methods for Classification — Logistic Regression

We can write the Newton-Raphson algorithm in matrix form

- Let $\mathbf{X} \in \mathbb{R}^{n \times (p+1)}$ be a matrix containing vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n$
- Let $\mathbf{p} \in \mathbb{R}^n$ be the a vector with the probabilities $p(\mathbf{x}_i, \boldsymbol{\beta}^k)$ for $i = 1, \ldots, n$
- Let $\mathbf{W} \in \mathbb{R}^{n \times n}$ be a diagonal matrix with diagonal entries $p(\mathbf{x}_i, \boldsymbol{\beta}^k)\left(1 - p(\mathbf{x}_i, \boldsymbol{\beta}^k)\right)$
- Let $\mathbf{y} \in \mathbb{R}^n$ denote the vector of $y_i$ values, for $i = 1, \ldots, n$

Then we have that

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

$$\frac{\partial^2 NLL(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^T} = \mathbf{X}^T \mathbf{W} \mathbf{X}$$

# Linear Methods for Classification — Logistic Regression

Thus, the Newton-Raphson equations in matrix form are given by

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

where $\mathbf{z} = \mathbf{X}\boldsymbol{\beta}^{k} + \mathbf{W}^{-1}(\mathbf{y}-\mathbf{p})$

- This process is repeated until convergence. The terms $\mathbf{p}$, $\mathbf{W}$, and $\mathbf{z}$ change at each iteration
- This algorithm is known as iteratively reweighted least squares (IRLS) since each iteration solves the weighted least squares problem

$$\boldsymbol{\beta}^{k+1} \leftarrow \underset{\boldsymbol{\beta}^{k}}{\operatorname{argmin}} \left(\mathbf{z}-\mathbf{X}\boldsymbol{\beta}^{k}\right)^T \mathbf{W}\left(\mathbf{z}-\mathbf{X}\boldsymbol{\beta}^{k}\right)$$

**Shrinkage in Classification:** Shrinkage can be applied to classification to reduce the number of nonzero coefficients.

Recall that Logistic Regression can be formulated as:

$$\max_{\boldsymbol{\beta}} NLL(\boldsymbol{\beta}) = \max_{\boldsymbol{\beta}} \left\{ -\sum_{i=1}^{n} y_i \log\big(p(\mathbf{x}_i, \boldsymbol{\beta})\big) + (1 - y_i)\log\big(1 - p(\mathbf{x}_i, \boldsymbol{\beta})\big) \right\}$$

where $p(\mathbf{x}_i, \boldsymbol{\beta}) = \frac{1}{1+e^{-\boldsymbol{\beta}^T \mathbf{x}_i}}$. Rearranging,

$$\max_{\boldsymbol{\beta}} NLL(\boldsymbol{\beta}) = \max_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{n} y_i \boldsymbol{\beta}^T \mathbf{x}_i - \log\left(1 + e^{\boldsymbol{\beta}^T \mathbf{x}_i}\right) \right\}$$

$L_1$ regularization (shrinkage) is realized by adding a penalty term:

$$\max_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{n} \left[ y_i \boldsymbol{\beta}^T \mathbf{x}_i - \log\left(1 + e^{\boldsymbol{\beta}^T \mathbf{x}_i}\right) \right] - \lambda \sum_{j=1}^{p} |\beta_j| \right\}$$

$L_2$ shrinkage (regularization) can also be applied

In this case, the regularized negative log-likelihood function is given by

$$NLL(\boldsymbol{\beta}) = -\sum_{i=1}^{n} y_i \log\big(p(\mathbf{x}_i, \boldsymbol{\beta})\big) + (1 - y_i)\log\big(1 - p(\mathbf{x}_i, \boldsymbol{\beta})\big) - \frac{\lambda}{2}\sum_{j=1}^{p} \beta_j^2$$

and the derivative is given by

$$\frac{\partial NLL(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = -\sum_{i=1}^{n} \mathbf{x}_i\big(y_i - p(\mathbf{x}_i, \boldsymbol{\beta})\big) - \lambda\sum_{j=1}^{p} \beta_j$$

# Example: South African Heart Disease

**Objective:** Coronary Risk-Factor Study — establish the intensity of ischemic heart disease risk factors in the high-incident South African region.

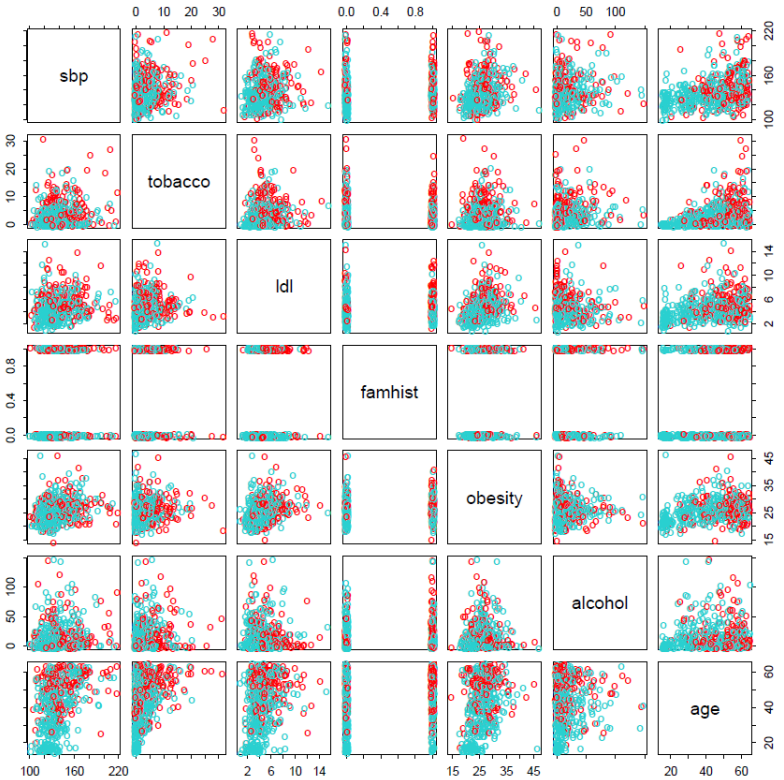
**Data** (by Rousseauw et al. 1983):

Given variables (clinical measures):

- `sbp:` systolic blood pressure
- `tobacco`
- `ldl:` Low-Density Lipoprotein (Cholesterol)
- `famhist:` family history
- `obesity`
- `alcohol`
- `age`

Data represents white males between 15 and 64; 160 cases in the data set

**Goal:** predict the presence or absence of myocardial infarction (`MI`) using logistic regression




A scatterplot matrix of the South African heart disease data. Each plot shows a pair of risk factors, and the cases and controls are color coded (red is a case). The variable family history of heart disease (famhist) is binary (yes or no).

# Example: South African Heart Disease

**Recall:** A Z-score greater than 2 in the absolute value $\Longrightarrow$ component is significant (at the 5% level)

**Results:** Variables `sbp`, `obesity`, and `alcohol` are not significant

- Systolic blood pressure (`sbp`) is not significant?!
- Obesity is not significant?!
- **Note:** On their own, they are significant. However, in the presence of other correlated variables they are no longer needed.

Results from a logistic regression fit to the South African heart disease data.

| Term | Coefficient | Std. Error | Z Score |
|---|---|---|---|
| `Intercept` | -4.130 | 0.964 | -4.285 |
| `sbp` | 0.006 | 0.006 | 1.023 |
| `tobacco` | 0.080 | 0.026 | 3.034 |
| `ldl` | 0.185 | 0.057 | 3.219 |
| `famhist` | 0.939 | 0.225 | 4.178 |
| `obesity` | -0.035 | 0.029 | -1.187 |
| `alcohol` | 0.001 | 0.004 | 0.136 |
| `age` | 0.043 | 0.010 | 4.184 |

# Example: South African Heart Disease

**Options to Improve Results:**

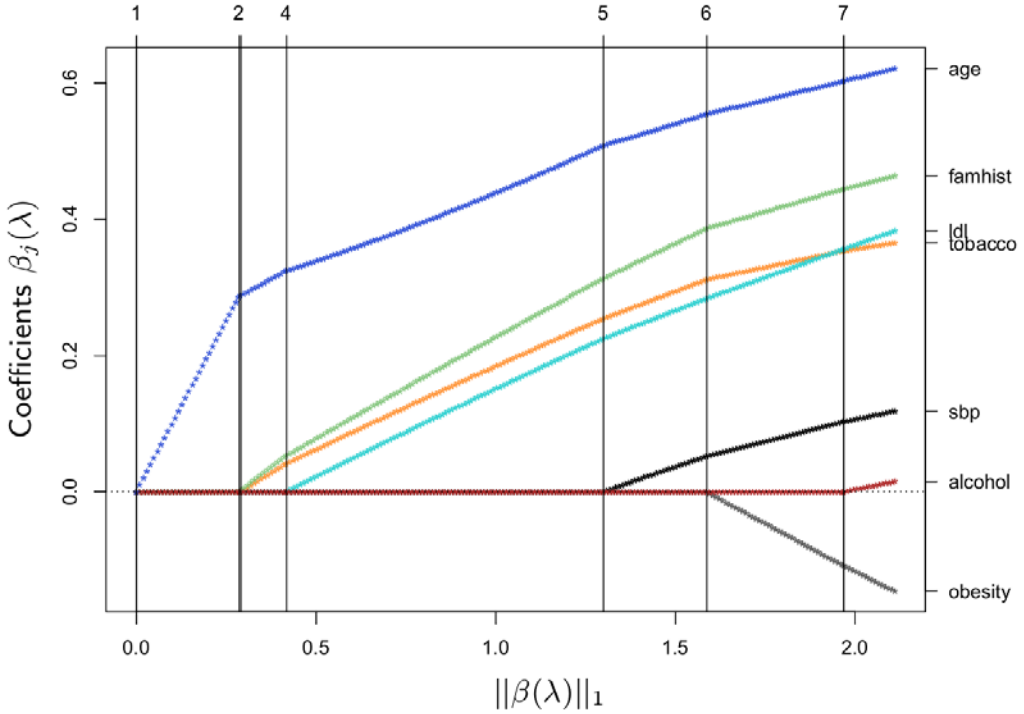
**Stepwise Logistic Regression Fit:** Drop the least significant coefficient and refit the model. Iterate, repeatedly dropping the least significant coefficient until no further terms can be dropped from the model. Results are show in the Table to the right.

**Analysis of Deviance**: Refit each of the models with one variable removed, then perform an *analysis of deviance* to decide which variable to exclude.

- The residual deviance of a fitted model is minus twice its log-likelihood, and the deviance between two models is the difference of their individual residual deviances (in analogy to sums-of-squares).
- Yields same result as Stepwise Logistic Regression, in this case
- In general, analysis of deviance is a superior approach, but more computationally intensive

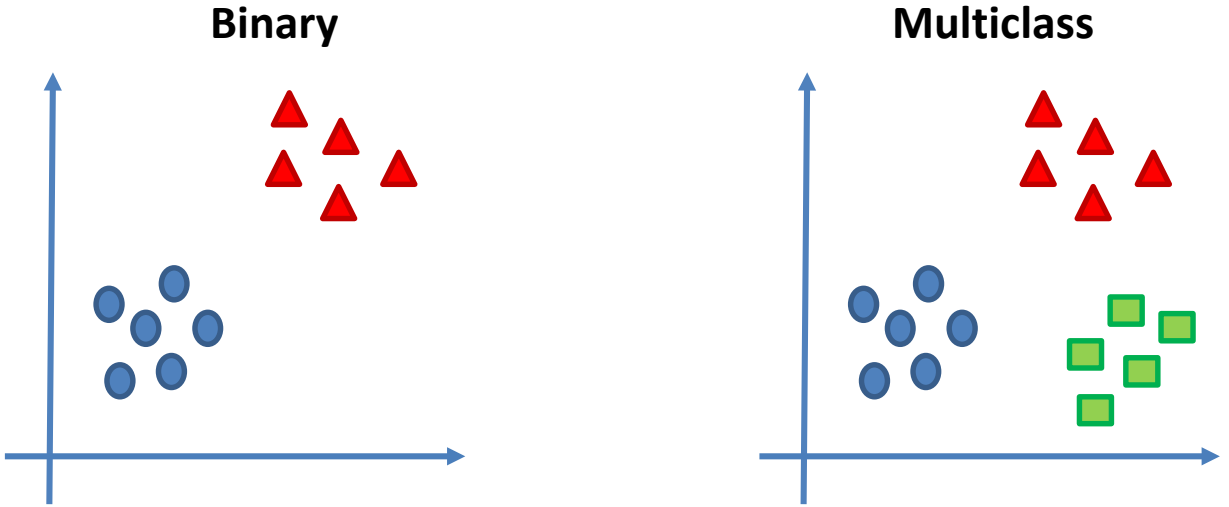
Results from stepwise logistic regression fit to the South African heart disease data.

| Term | Coefficient | Std. Error | Z Score |
|---|---|---|---|
| Intercept | -4.204 | 0.498 | -8.45 |
| tobacco | 0.081 | 0.026 | 3.16 |
| ldl | 0.168 | 0.054 | 3.09 |
| famhist | 0.924 | 0.223 | 4.14 |
| age | 0.044 | 0.010 | 4.52 |

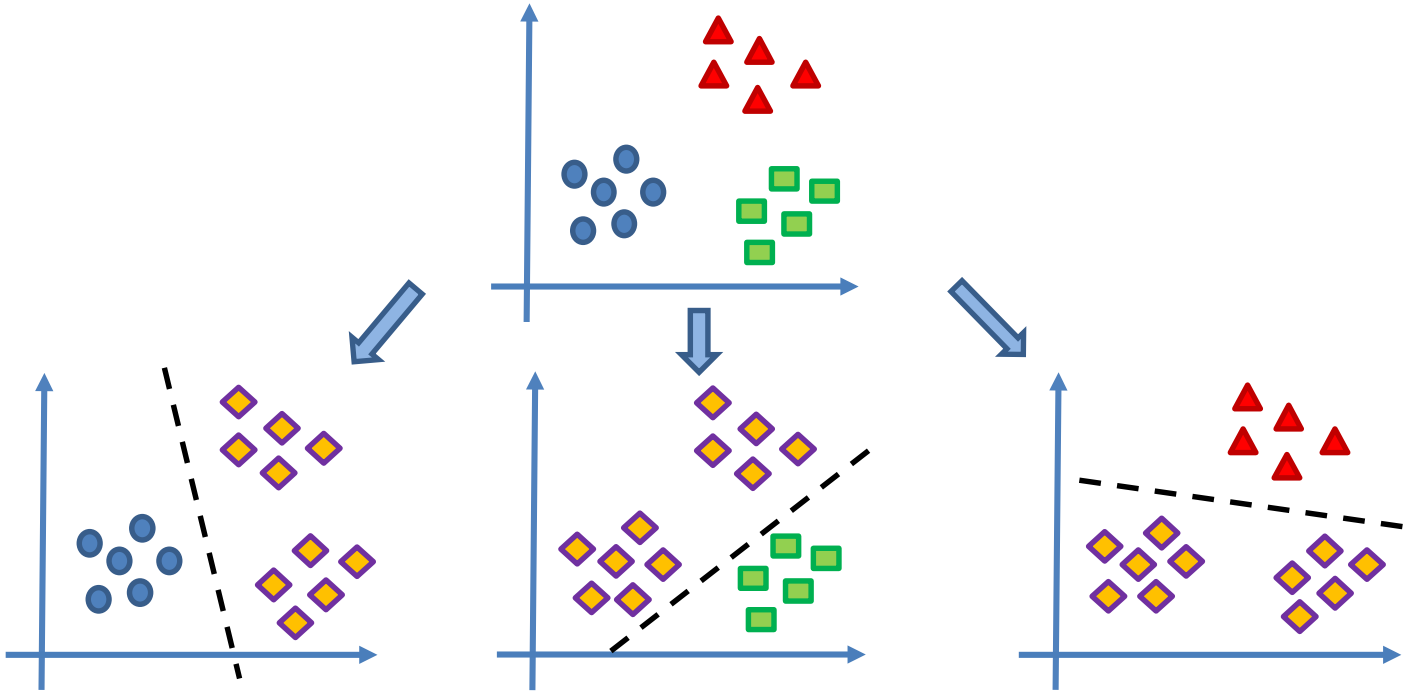



Effect of the penalty $\lambda \sum_{j=1}^{p} |\beta_j|$ on the estimated coefficients $\widehat{\boldsymbol{\beta}}$ as a function of the $L_1$ norm

**Binary Classification for the Multiclass Case:** Suppose we have $K$ classes, where $K > 2$, and we want to use (binary) logistic regression. What are the options for handling the multiclass case?



**One-vs-All Prediction**: Train $K$ separate binary classifiers



**Methodology:**

- Train $K$ separate binary classifiers producing a set of coefficients $\boldsymbol{\beta}_j$, for $j = 1, \ldots, K$
- For every new example $\mathbf{x}$, obtain the predicted label $\hat{\jmath}$ as

$$\hat{\jmath} = \underset{j}{\operatorname{argmax}}\ p\left(\mathbf{x}, \boldsymbol{\beta}_j\right)$$

Recall that $p\left(\mathbf{x}, \boldsymbol{\beta}_j\right) = \frac{1}{1+e^{-\boldsymbol{\beta}_j^T \mathbf{x}}}$

# Linear Methods for Classification — Logistic Regression

**Summary:**

- Classification is the prediction of qualitative responses
- Linear methods for classification result in linear decision boundaries
- Logistic Regression assumes a Bernoulli class model: $\Pr(G = 1) = p$ and $\Pr(G = 0) = 1 - p, p \in [0,1]$ and the log ratio of the class probabilities $logit(p) = \log\left(\frac{p}{1-p}\right)$

$$\Pr(G = 1|\boldsymbol{X} = \boldsymbol{x}) = p = \frac{1}{1 + e^{-\boldsymbol{\beta}^T\mathbf{x}}}$$

$$\Pr(G = 0|\boldsymbol{X} = \boldsymbol{x}) = 1 - p = \frac{1}{1 + e^{\boldsymbol{\beta}^T\mathbf{x}}}$$

- $\boldsymbol{\beta}^T\mathbf{x} > 0 \Longrightarrow \Pr(G = 1|\boldsymbol{X} = \boldsymbol{x}) > \Pr(G = 0|\boldsymbol{X} = \boldsymbol{x})$ while $\boldsymbol{\beta}^T\mathbf{x} < 0 \Longrightarrow \Pr(G = 0|\boldsymbol{X} = \boldsymbol{x}) > \Pr(G = 1|\boldsymbol{X} = \boldsymbol{x})$
  - $\boldsymbol{\beta}^T\mathbf{x} > 0 \Longrightarrow \hat{G}(\mathbf{x}) = 1$ and $\boldsymbol{\beta}^T\mathbf{x} < 0 \Longrightarrow \hat{G}(\mathbf{x}) = 0$
- Results were extended to the multiple iid observation case
- Optimal $\boldsymbol{\beta}$ determine through Gradient Descent or Newton-Raphson algorithm
- $L_1$and $L_2$shrinkage (regularization) can be applied
- Binary classifiers can be applied to the multiclass case through a one-versus-all strategy