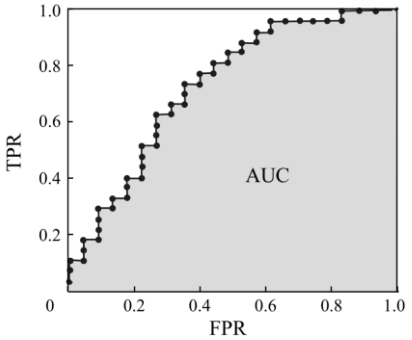
Lesson3: Linear Models

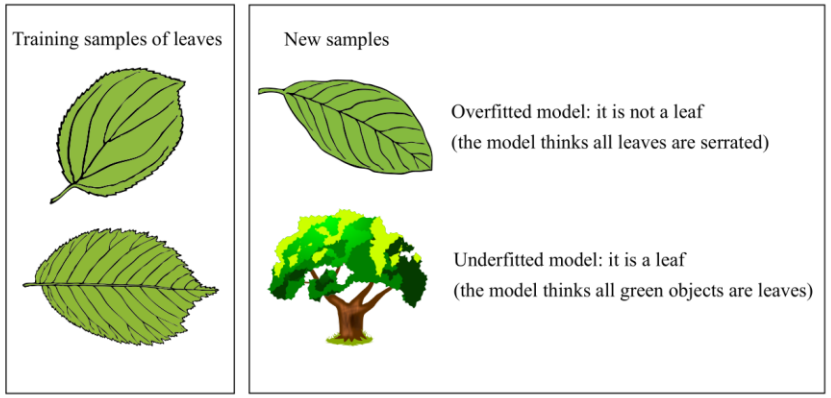
Instructor: Xiang Deng

2025.10.10

review



- Empirical Error and Overfitting: overfitting, underfitting



The confusion matrix of binary classification

Ground-truth class	Predicted class	
	Positive	Negative
Positive	TP	FN
Negative	FP	TN

Precision $P = \frac{TP}{TP + FP}$

Recall $R = \frac{TP}{TP + FN}$

$$F1 = \frac{2 \times P \times R}{P + R}$$

- Evaluation methods: generalization error, Hold-out, Cross-Validation, Bootstrapping
- Performance Measure: Mean Squared Error, Error rate, accuracy, Precision, recall, f1, P-R curve, break-even points, ROC curve and AUC, cost-sensitive error rate
- Performance Comparisons: Hypothesis testing
- Bias and Variance: the generalization error can be decomposed into the sum of bias, variance, and noise.

$$E(f; D) = bias^2(\boldsymbol{x}) + var(\boldsymbol{x}) + \varepsilon^2$$

Outline

- Linear Regression
  - Least Squares Method
- Binary Classification Problem
  - Logistic Regression
  - Linear Discriminant Analysis
- Multiclass Classification Problem
  - One vs. One (OvO)
  - One vs. Rest (OvR)
  - Many vs. Many (MvM)
- Class Imbalance Problem

The Basic Form

- The Basic form of the linear model

$$f(\boldsymbol{x}) = w_1 x_1 + w_2 x_2 + \ldots + w_d x_d + b$$

$\boldsymbol{x} = (x_1; x_2; \ldots; x_d)$ is a sample where $\boldsymbol{x}$ takes the value x_i on the i-th variable.

- The vector form

$$f(\boldsymbol{x}) = \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x} + b$$

where $\boldsymbol{w} = (w_1; w_2; \ldots; w_d)$

Advantages of the Linear Model

- Simple form and ease of modeling
- Comprehensibility
- Nonlinear models can be derived from linear models
  - Introducing multi-layer structures or high-dimensional mapping.

- An example
  - Determine the ripeness of a watermelon by considering its **color**, **root** and **sound** information.

$$f_{\text{ripe}}(\boldsymbol{x}) = 0.2 \cdot x_{\text{color}} + 0.5 \cdot x_{\text{root}} + 0.3 \cdot x_{\text{sound}} + 1$$

  - From the coefficients, we know that root is the most important variable, and sound is more important than color.

Linear Regression

- Given a data set $D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m)\}$

where $\boldsymbol{x}_i = (x_{i1}; x_{i2}; \ldots; x_{id})$ $y_i \in \mathbb{R}$

- The aim of linear regression
  - Learn a linear model that can accurately predict the real-valued output labels

- For discrete variables
  - An ordinal relationship exists between values
    - Convert the variables into real-valued variables
  - No ordinal relationship exists
    - Convert the discrete variable with k possible values into a k-dimensional vector

Linear Regression

- Linear regression aims to learn the function:

$$f(x) = wx + b, \text{ such that } f(x_i) \simeq y_i, i = 1, \ldots, m$$

- Parameter/model estimation: least square method

$$\begin{aligned}(w^*, b^*) &= \arg\min_{(w,b)} \sum_{i=1}^{m} \left(f\left(x_i\right) - y_i\right)^2 \\ &= \arg\min_{(w,b)} \sum_{i=1}^{m} \left(y_i - wx_i - b\right)^2\end{aligned}$$

Linear Regression - Least Square Method

- Minimize mean-square error (MSE)

$$E_{(w,b)} = \sum_{i=1}^{m} (y_i - wx_i - b)^2$$

- Calculate the derivatives of $E_{(w,b)}$ with respect to w and b respectively:

$$\frac{\partial E_{(w,b)}}{\partial w} = 2\left(w\sum_{i=1}^{m} x_i^2 - \sum_{i=1}^{m}(y_i - b)\,x_i\right)$$

$$\frac{\partial E_{(w,b)}}{\partial b} = 2\left(mb - \sum_{i=1}^{m}(y_i - wx_i)\right)$$

$$\begin{aligned}
\frac{\partial E_{(w,b)}}{\partial w} &= \frac{\partial}{\partial w}\left[\sum_{i=1}^{m}(y_i - wx_i - b)^2\right] \\
&= \sum_{i=1}^{m}\frac{\partial}{\partial w}\left[(y_i - wx_i - b)^2\right] \\
&= \sum_{i=1}^{m}\left[2\cdot(y_i - wx_i - b)\cdot(-x_i)\right] \\
&= \sum_{i=1}^{m}\left[2\cdot(wx_i^2 - y_i x_i + bx_i)\right] \\
&= 2\cdot\left(w\sum_{i=1}^{m}x_i^2 - \sum_{i=1}^{m}y_i x_i + b\sum_{i=1}^{m}x_i\right) \\
&= 2\left(w\sum_{i=1}^{m}x_i^2 - \sum_{i=1}^{m}(y_i - b)\,x_i\right)
\end{aligned}$$

$$\begin{aligned}
\frac{\partial E_{(w,b)}}{\partial b} &= \frac{\partial}{\partial b}\left[\sum_{i=1}^{m}(y_i - wx_i - b)^2\right] \\
&= \sum_{i=1}^{m}\frac{\partial}{\partial b}\left[(y_i - wx_i - b)^2\right] \\
&= \sum_{i=1}^{m}\left[2\cdot(y_i - wx_i - b)\cdot(-1)\right] \\
&= \sum_{i=1}^{m}\left[2\cdot(b - y_i + wx_i)\right] \\
&= 2\cdot\left[\sum_{i=1}^{m}b - \sum_{i=1}^{m}y_i + \sum_{i=1}^{m}wx_i\right] \\
&= 2\left(mb - \sum_{i=1}^{m}(y_i - wx_i)\right)
\end{aligned}$$

Linear Regression - Least Square Method

- We have the closed-form solutions (derivatives w.r.t. w and b equal to 0) :

convex function

$$E_{(w,b)} = \sum_{i=1}^{m} (y_i - wx_i - b)^2$$

$$w = \frac{\sum_{i=1}^{m} y_i (x_i - \bar{x})}{\sum_{i=1}^{m} x_i^2 - \frac{1}{m} \left(\sum_{i=1}^{m} x_i \right)^2}$$

$$b = \frac{1}{m} \sum_{i=1}^{m} (y_i - wx_i)$$

where

$$\bar{x} = \frac{1}{m} \sum_{i=1}^{m} x_i$$

How are they obtained?

Linear Regression - Least Square Method

$$0 = w\sum_{i=1}^{m} x_i^2 - \sum_{i=1}^{m}(y_i - b)x_i \qquad w\sum_{i=1}^{m} x_i^2 = \sum_{i=1}^{m} y_i x_i - \sum_{i=1}^{m} b x_i$$

$$0 = 2\left(mb - \sum_{i=1}^{m}(y_i - wx_i)\right) \qquad b = \frac{1}{m}\sum_{i=1}^{m}(y_i - wx_i)$$

$$w\sum_{i=1}^{m} x_i^2 = \sum_{i=1}^{m} y_i x_i - \sum_{i=1}^{m}(\bar{y} - w\bar{x})x_i$$

$$w\sum_{i=1}^{m} x_i^2 = \sum_{i=1}^{m} y_i x_i - \bar{y}\sum_{i=1}^{m} x_i + w\bar{x}\sum_{i=1}^{m} x_i$$

$$w(\sum_{i=1}^{m} x_i^2 - \bar{x}\sum_{i=1}^{m} x_i) = \sum_{i=1}^{m} y_i x_i - \bar{y}\sum_{i=1}^{m} x_i$$

$$w = \frac{\sum_{i=1}^{m} y_i x_i - \bar{y}\sum_{i=1}^{m} x_i}{\sum_{i=1}^{m} x_i^2 - \bar{x}\sum_{i=1}^{m} x_i}$$

$$\frac{1}{m}\sum_{i=1}^{m} y_i = \bar{y},\ \frac{1}{m}\sum_{i=1}^{m} x_i = \bar{x}$$

$$\bar{y}\sum_{i=1}^{m} x_i = \frac{1}{m}\sum_{i=1}^{m} y_i \sum_{i=1}^{m} x_i = \bar{x}\sum_{i=1}^{m} y_i$$

$$\bar{x}\sum_{i=1}^{m} x_i = \frac{1}{m}\sum_{i=1}^{m} x_i \sum_{i=1}^{m} x_i = \frac{1}{m}(\sum_{i=1}^{m} x_i)^2$$

$$w = \frac{\sum_{i=1}^{m} y_i(x_i - \bar{x})}{\sum_{i=1}^{m} x_i^2 - \frac{1}{m}(\sum_{i=1}^{m} x_i)^2}$$

Multivariate Linear Regression

- Given a data set

$$D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m)\}$$

$$\boldsymbol{x}_i = (x_{i1}; x_{i2}; \ldots; x_{id}) \quad y_i \in \mathbb{R}$$

- The objective function of multivariate linear regression

$$f(\boldsymbol{x}_i) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b \text{ such that } f(\boldsymbol{x}_i) \simeq y_i$$

Multivariate Linear Regression

- Rewrite

$$\hat{\boldsymbol{w}} = (\boldsymbol{w}; b) = (w_1; \ldots; w_d; b) \in \mathbb{R}^{(d+1)\times 1}, \hat{\boldsymbol{x}}_i = (x_{i1}; \ldots; x_{id}; 1) \in \mathbb{R}^{(d+1)\times 1},$$

- Least square method

$$\begin{aligned}\hat{\boldsymbol{w}}^* &= \arg\min_{\hat{\boldsymbol{w}}} \sum_{i=1}^{m} \left(y_i - \hat{\boldsymbol{w}}^{\mathrm{T}} \hat{\boldsymbol{x}}_i\right)^2 \\ &= \arg\min_{\hat{\boldsymbol{w}}} \sum_{i=1}^{m} \left(y_i - \hat{\boldsymbol{x}}_i^{\mathrm{T}} \hat{\boldsymbol{w}}\right)^2\end{aligned}$$

$$\hat{\boldsymbol{w}}^* = \arg\min_{\hat{\boldsymbol{w}}} \begin{bmatrix} y_1 - \hat{\boldsymbol{x}}_1^{\mathrm{T}} \hat{\boldsymbol{w}} & \cdots & y_m - \hat{\boldsymbol{x}}_m^{\mathrm{T}} \hat{\boldsymbol{w}} \end{bmatrix} \begin{bmatrix} y_1 - \hat{\boldsymbol{x}}_1^{\mathrm{T}} \hat{\boldsymbol{w}} \\ \vdots \\ y_m - \hat{\boldsymbol{x}}_m^{\mathrm{T}} \hat{\boldsymbol{w}} \end{bmatrix}$$

Multivariate Linear Regression

$$\begin{bmatrix} y_1 - \hat{\boldsymbol{x}}_1^{\mathrm{T}}\hat{\boldsymbol{w}} \\ \vdots \\ y_m - \hat{\boldsymbol{x}}_m^{\mathrm{T}}\hat{\boldsymbol{w}} \end{bmatrix} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix} - \begin{bmatrix} \hat{\boldsymbol{x}}_1^{\mathrm{T}}\hat{\boldsymbol{w}} \\ \vdots \\ \hat{\boldsymbol{x}}_m^{\mathrm{T}}\hat{\boldsymbol{w}} \end{bmatrix}$$

$$= \boldsymbol{y} - \begin{bmatrix} \hat{\boldsymbol{x}}_1^{\mathrm{T}} \\ \vdots \\ \hat{\boldsymbol{x}}_m^{\mathrm{T}} \end{bmatrix} \cdot \hat{\boldsymbol{w}}$$

$$= \boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}}$$

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} & 1 \\ x_{21} & x_{22} & \cdots & x_{2d} & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{md} & 1 \end{pmatrix} = \begin{pmatrix} \boldsymbol{x}_1^{\mathrm{T}} & 1 \\ \boldsymbol{x}_2^{\mathrm{T}} & 1 \\ \vdots & \vdots \\ \boldsymbol{x}_m^{\mathrm{T}} & 1 \end{pmatrix}$$

$$\boldsymbol{y} = (y_1; y_2; \ldots; y_m)$$

$$\hat{\boldsymbol{w}}^* = \arg\min_{\hat{\boldsymbol{w}}} (\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}})^{\mathrm{T}} (\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}})$$

Let $E_{\hat{\boldsymbol{w}}} = (\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}})^{\mathrm{T}} (\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}})$ and find the derivative with respect to $\hat{\boldsymbol{w}}$

Multivariate Linear Regression - Least Square Method

- The derivative with respect to $\hat{\boldsymbol{w}}$

$$E_{\hat{\boldsymbol{w}}} = (\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}})^{\mathrm{T}}(\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}})$$

$$E_{\hat{\boldsymbol{w}}} = \boldsymbol{y}^{\mathrm{T}}\boldsymbol{y} - \boldsymbol{y}^{\mathrm{T}}\mathbf{X}\hat{\boldsymbol{w}} - \hat{\boldsymbol{w}}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\boldsymbol{y} + \hat{\boldsymbol{w}}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{X}\hat{\boldsymbol{w}}$$

$$\frac{\partial E_{\hat{\boldsymbol{w}}}}{\partial \hat{\boldsymbol{w}}} = 0 - \mathbf{X}^{\mathrm{T}}\boldsymbol{y} - \mathbf{X}^{\mathrm{T}}\boldsymbol{y} + (\mathbf{X}^{\mathrm{T}}\mathbf{X} + \mathbf{X}^{\mathrm{T}}\mathbf{X})\hat{\boldsymbol{w}}$$

$$= 2\mathbf{X}^{\mathrm{T}}(\mathbf{X}\hat{\boldsymbol{w}} - \boldsymbol{y})$$

$$\frac{\partial E_{\hat{\boldsymbol{w}}}}{\partial \hat{\boldsymbol{w}}} = 2\mathbf{X}^{\mathrm{T}}(\mathbf{X}\hat{\boldsymbol{w}} - \boldsymbol{y})$$

Recall Matrix transpose:

$$(AB)^T = B^T A^T$$

$$(A + B)^T = A^T + B^T$$

Recall Matrix Differentiation:

$$\frac{\partial \boldsymbol{a}^{\mathrm{T}}\boldsymbol{x}}{\partial \boldsymbol{x}} = \frac{\partial \boldsymbol{x}^{\mathrm{T}}\boldsymbol{a}}{\partial \boldsymbol{x}} = \boldsymbol{a}, \frac{\partial \boldsymbol{x}^{\mathrm{T}}\mathbf{A}\boldsymbol{x}}{\partial \boldsymbol{x}} = (\mathbf{A}+\mathbf{A}^{\mathrm{T}})\boldsymbol{x}$$

The closed-form solution of $\hat{\boldsymbol{w}}$ can be obtained by making the equation equal to 0.

Multivariate Linear Regression - Discussion of full-rank

- $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ is a full-rank matrix or a positive definite matrix, then by making the above equation equal to 0:

$$\hat{\boldsymbol{w}}^{*} = \left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathrm{T}}\boldsymbol{y}$$

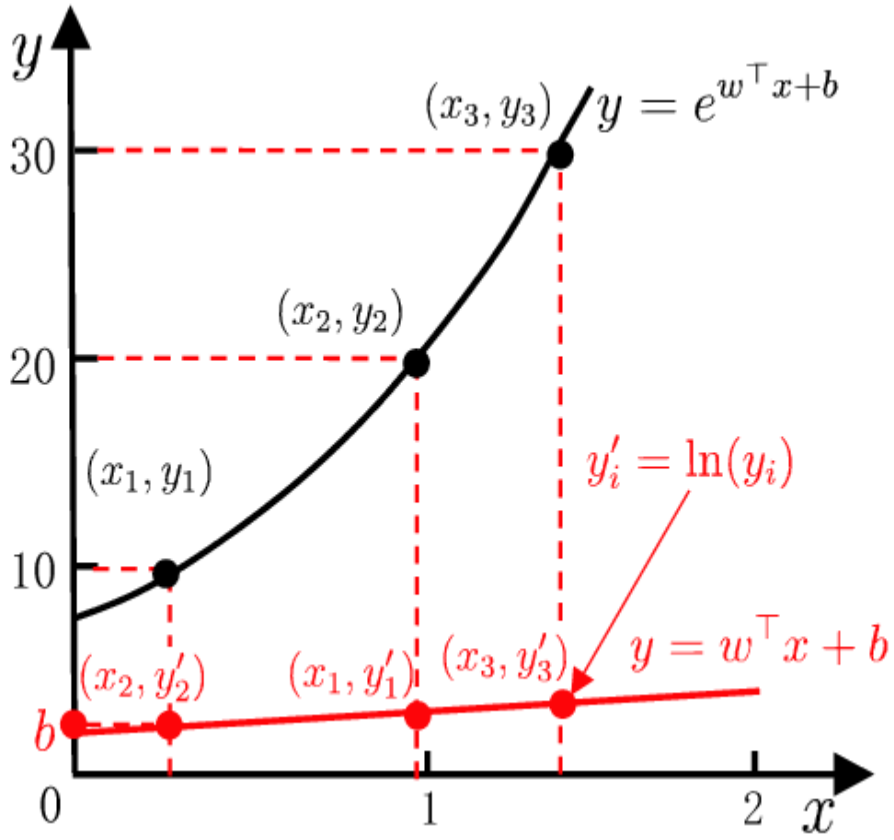
where $\left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}$ is the inverse of $\mathbf{X}^{\mathrm{T}}\mathbf{X}$, the learned multivariate linear regression model is

$$f\left(\hat{\boldsymbol{x}}_i\right) = \hat{\boldsymbol{x}}_i^{\mathrm{T}}\left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathrm{T}}\boldsymbol{y}$$

- $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ is often not full-rank
  - introducing regularization.

Log-Linear Regression

- The logarithm of the output label can be used for approximation



$$\ln y = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b$$

$$\Updownarrow$$

$$y = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b$$

Linear Regression - Generalized Linear Model

- The general form

$$y = g^{-1}\left(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b\right)$$

- Where the function $g(\cdot)$ is the link function.
  - a monotonic differentiable function

- Log-linear regression is a special case of generalized linear models when $g(\cdot) = \ln(\cdot)$

Binary Classification

- The predictions and the output labels

$$z = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b \qquad y \in \{0, 1\}$$

- The real-valued predictions of the linear regression model need to be converted into 0/1.

- Ideally, the unit-step function is desired

$$y = \begin{cases} 0, & z < 0; \\ 0.5, & z = 0; \\ 1, & z > 0, \end{cases}$$

  - which predicts positive for z greater than 0, negative for z smaller than 0, and an arbitrary output when z equals to 0.

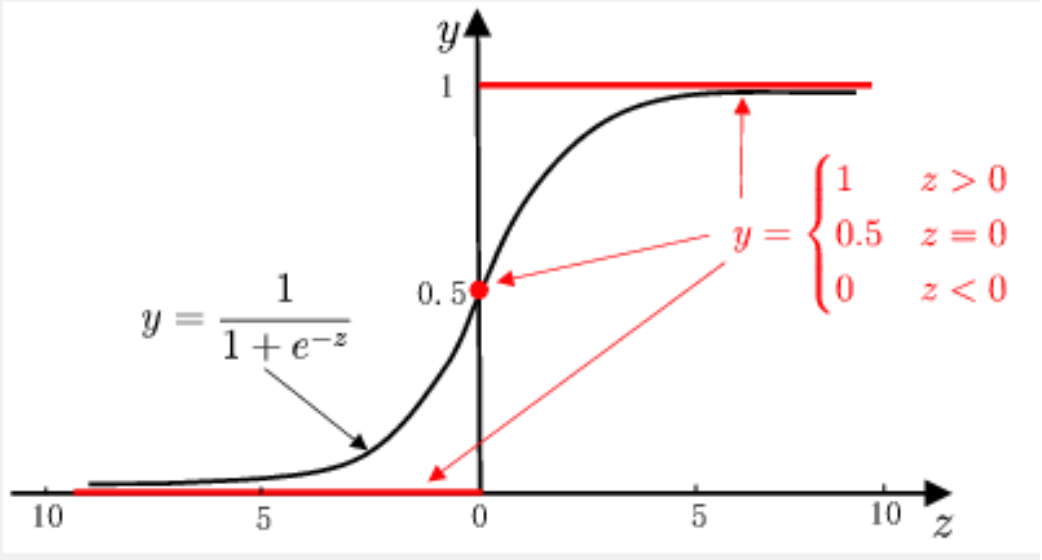
Binary Classification

- Disadvantages of unit-step function
  - not continuous

- Logistic function: a surrogate function to approximate the unit-step function
  - monotonic differentiable

$$y = \frac{1}{1 + e^{-z}}$$

Comparison between unit-step function and logistic function



Logistic regression

- Apply logistic function

$$y = \frac{1}{1 + e^{-z}} \quad \text{transform into} \quad y = \frac{1}{1 + e^{-(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b)}}$$

- Log odds
  - the logarithm of the relative likelihood of a sample being a positive sample

$$\ln \frac{y}{1 - y}$$

- Logistic regression has several nice properties

- without requiring any prior assumptions on the data distribution
- it predicts labels together with associated probabilities
- it is solvable with numerical optimization methods.

Logistic regression - maximum likelihood

- Log odds can be rewritten as

$$\ln \frac{p\left(y=1 \mid \boldsymbol{x}\right)}{p\left(y=0 \mid \boldsymbol{x}\right)} = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b$$

$$p\left(y=1 \mid \boldsymbol{x}\right) = \frac{e^{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b}}{1+e^{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b}}$$

$$p\left(y=0 \mid \boldsymbol{x}\right) = \frac{1}{1+e^{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b}}$$

Logistic regression - maximum likelihood

- Maximum likelihood
  - Given a data set

$$\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{m}$$

  - Maximizing the probability of each sample being predicted as the ground-truth label
    - the log-likelihood to be maximized is

$$\ell(\boldsymbol{w}, b) = \sum_{i=1}^{m} \ln p(y_i \mid \boldsymbol{x}_i; \boldsymbol{w}_i, b)$$

Logistic regression - maximum likelihood

- Transform into minimize negative log-likelihood
  - Let $\boldsymbol{\beta} = (\boldsymbol{w}; b), \hat{\boldsymbol{x}} = (\boldsymbol{x}; 1)$, $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b$ can be rewritten as $\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}$
  - Let

$$p_1\left(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta}\right) = p\left(y = 1 \mid \hat{\boldsymbol{x}}; \boldsymbol{\beta}\right)$$

$$p_0\left(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta}\right) = p\left(y = 0 \mid \hat{\boldsymbol{x}}; \boldsymbol{\beta}\right) = 1 - p_1\left(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta}\right)$$

the likelihood term in can be rewritten as

$$p\left(y_i \mid \boldsymbol{x}_i; \boldsymbol{w}_i, b\right) = y_i p_1\left(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta}\right) + \left(1 - y_i\right) p_0\left(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta}\right)$$

  - the log-likelihood can be written:

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{m} \ln\left(y_i p_1(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta}) + (1 - y_i) p_0(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta})\right)$$

Logistic regression - maximum likelihood

- since $p_1(\hat{\boldsymbol{x}}_i;\boldsymbol{\beta}) = \frac{e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}}{1+e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}}, p_0(\hat{\boldsymbol{x}}_i;\boldsymbol{\beta}) = \frac{1}{1+e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}}$,

$$\begin{aligned}\ell(\boldsymbol{\beta}) &= \sum_{i=1}^{m} \ln\left(\frac{y_i e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i} + 1 - y_i}{1 + e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}}\right) \\ &= \sum_{i=1}^{m} \left(\ln(y_i e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i} + 1 - y_i) - \ln(1 + e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i})\right)\end{aligned}$$

- y_i can be 0 or 1:

$$\ell(\boldsymbol{\beta}) = \begin{cases} \sum_{i=1}^{m}(-\ln(1 + e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i})), & y_i = 0 \\ \sum_{i=1}^{m}(\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i - \ln(1 + e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i})), & y_i = 1 \end{cases}$$

$$\ell\left(\boldsymbol{\beta}\right) = \sum_{i=1}^{m}\left(-y_i\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i + \ln\left(1 + e^{\beta^{\mathrm{T}}\hat{\boldsymbol{x}}_i}\right)\right)$$

Logistic regression

- We have

$$\boldsymbol{\beta}^* = \arg\min_{\boldsymbol{\beta}} \ell(\boldsymbol{\beta})$$

- Taking Newton's method as an example, the update rule at the (t+1)th iteration is

$$\boldsymbol{\beta}^{t+1} = \boldsymbol{\beta}^t - \left(\frac{\partial^2 \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^{\mathrm{T}}} \right)^{-1} \frac{\partial \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}}$$

Need to calculate the first- and second-order derivatives with respect to β

higher order differentiable convex function, gradient descent method / Newton's method [Boyd and Vandenberghe, 2004]

Logistic regression

the first-order derivatives with respect to β are

$$
\begin{aligned}
\frac{\partial \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} &= \frac{\partial \sum_{i=1}^{m}\left(-y_i \boldsymbol{\beta}^{\mathrm{T}} \hat{\boldsymbol{x}}_i+\ln\left(1+e^{\boldsymbol{\beta}^{\mathrm{T}} \hat{\boldsymbol{x}}_i}\right)\right)}{\partial \boldsymbol{\beta}} \\
&= \sum_{i=1}^{m}\left(\frac{\partial\left(-y_i \boldsymbol{\beta}^{\mathrm{T}} \hat{\boldsymbol{x}}_i\right)}{\partial \boldsymbol{\beta}}+\frac{\partial \ln\left(1+e^{\boldsymbol{\beta}^{\mathrm{T}} \hat{\boldsymbol{x}}_i}\right)}{\partial \boldsymbol{\beta}}\right) \\
&= \sum_{i=1}^{m}\left(-y_i \hat{\boldsymbol{x}}_i+\frac{1}{1+e^{\boldsymbol{\beta}^{\mathrm{T}} \hat{\boldsymbol{x}}_i}} \cdot \hat{\boldsymbol{x}}_i e^{\boldsymbol{\beta}^{\mathrm{T}} \hat{\boldsymbol{x}}_i}\right) \\
&= -\sum_{i=1}^{m} \hat{\boldsymbol{x}}_i\left(y_i-\frac{e^{\boldsymbol{\beta}^{\mathrm{T}} \hat{\boldsymbol{x}}_i}}{1+e^{\boldsymbol{\beta}^{\mathrm{T}} \hat{\boldsymbol{x}}_i}}\right) \\
&= -\sum_{i=1}^{m} \hat{\boldsymbol{x}}_i\left(y_i-p_1\left(\hat{\boldsymbol{x}}_i ; \boldsymbol{\beta}\right)\right)
\end{aligned}
$$

Logistic regression

the second-order derivatives with respect β are

$$\begin{aligned}
\frac{\partial^2 \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^{\mathrm{T}}} &= -\frac{\partial \sum_{i=1}^{m} \hat{\boldsymbol{x}}_i \left(y_i - \frac{e^{\boldsymbol{\beta}^{\mathrm{T}} \hat{\boldsymbol{x}}_i}}{1+e^{\mathrm{T}} \hat{\boldsymbol{x}}_i} \right)}{\partial \boldsymbol{\beta}^{\mathrm{T}}} \\
&= -\sum_{i=1}^{m} \hat{\boldsymbol{x}}_i \frac{\partial \left(y_i - \frac{e^{\boldsymbol{\beta}^{\mathrm{T}} \hat{\boldsymbol{x}}_i}}{1+e^{\boldsymbol{\beta}^{\mathrm{T}} \hat{\boldsymbol{x}}_i}} \right)}{\partial \boldsymbol{\beta}^{\mathrm{T}}} \\
&= -\sum_{i=1}^{m} \hat{\boldsymbol{x}}_i \left(\frac{\partial y_i}{\partial \boldsymbol{\beta}^{\mathrm{T}}} - \frac{\partial \left(\frac{e^{\boldsymbol{\beta}^{\mathrm{T}} \hat{\boldsymbol{x}}_i}}{1+e^{\boldsymbol{\beta}^{\mathrm{T}} \hat{\boldsymbol{x}}_i}} \right)}{\partial \boldsymbol{\beta}^{\mathrm{T}}} \right) \\
&= \sum_{i=1}^{m} \hat{\boldsymbol{x}}_i \cdot \frac{\partial \left(\frac{e^{\boldsymbol{\beta}^{\mathrm{T}} \hat{\boldsymbol{x}}_i}}{1+e^{\boldsymbol{\beta}^{\mathrm{T}} \hat{\boldsymbol{x}}_i}} \right)}{\partial \boldsymbol{\beta}^{\mathrm{T}}}
\end{aligned}$$

Logistic regression

according to matrix differential formulas $\frac{\partial \boldsymbol{a}^{\mathrm{T}}\boldsymbol{x}}{\partial \boldsymbol{x}^{\mathrm{T}}} = \frac{\partial \boldsymbol{x}^{\mathrm{T}}\boldsymbol{a}}{\partial \boldsymbol{x}^{\mathrm{T}}} = \boldsymbol{a}^{\mathrm{T}}$,

numerator denominator numerator denominator

$$\frac{\partial\left(\frac{e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}}{1+e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}}\right)}{\partial \boldsymbol{\beta}^{\mathrm{T}}} = \frac{\frac{\partial e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}}{\partial \boldsymbol{\beta}^{\mathrm{T}}} \cdot \left(1+e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}\right) - e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i} \cdot \frac{\partial\left(1+e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}\right)}{\partial \boldsymbol{\beta}^{\mathrm{T}}}}{\left(1+e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}\right)^2}$$

natural constant

$$= \frac{\hat{\boldsymbol{x}}_i^{\mathrm{T}} e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i} \cdot \left(1+e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}\right) - e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i} \cdot \hat{\boldsymbol{x}}_i^{\mathrm{T}} e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}}{\left(1+e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}\right)^2}$$

$$= \hat{\boldsymbol{x}}_i^{\mathrm{T}} e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i} \cdot \frac{\left(1+e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}\right) - e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}}{\left(1+e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}\right)^2}$$

$$= \hat{\boldsymbol{x}}_i^{\mathrm{T}} e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i} \cdot \frac{1}{\left(1+e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}\right)^2}$$

$$= \hat{\boldsymbol{x}}_i^{\mathrm{T}} \cdot \frac{e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}}{1+e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}} \cdot \frac{1}{1+e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}}$$

from the previouse slide:

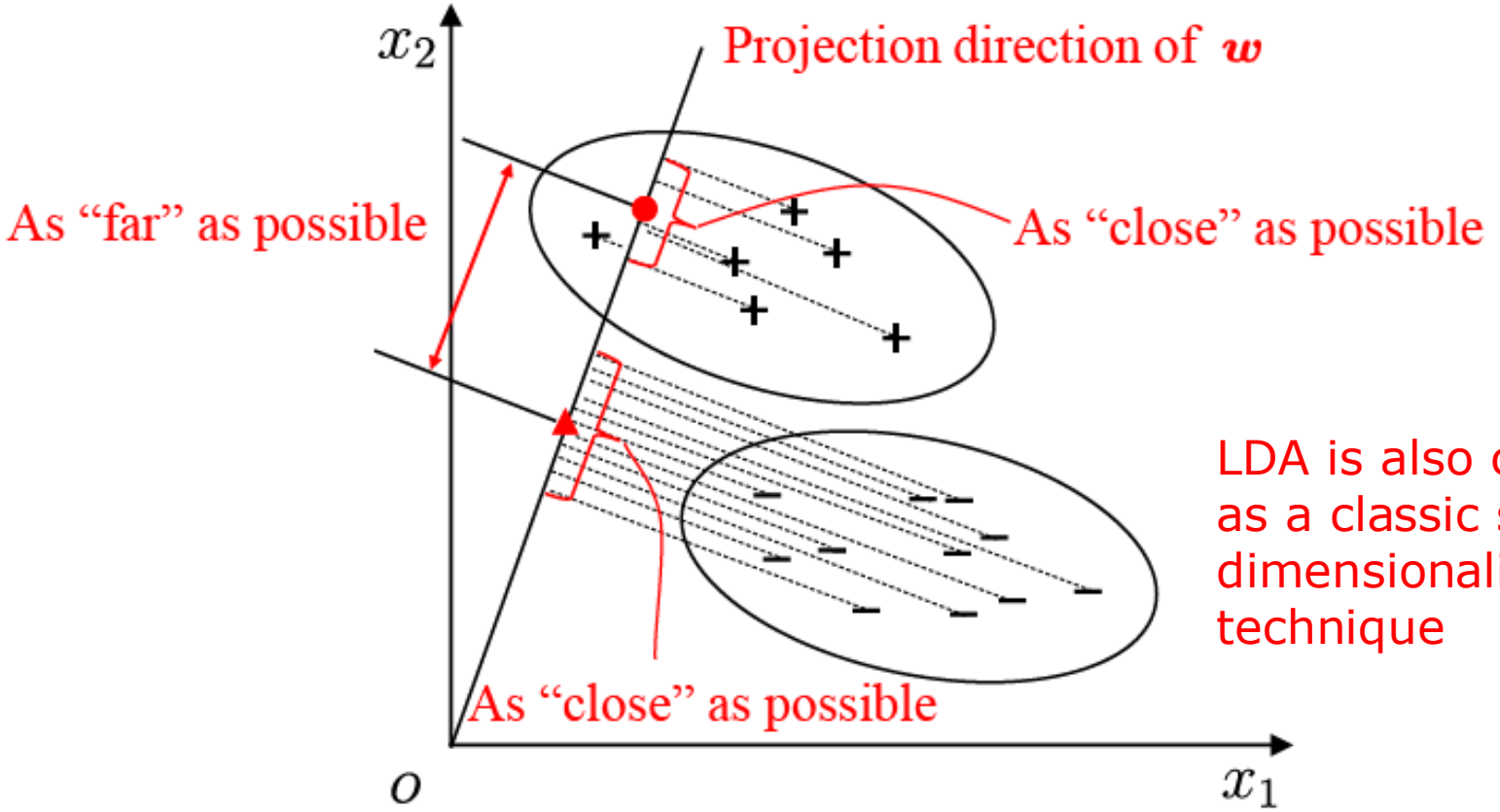
$$\frac{\partial^2 \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^{\mathrm{T}}} = \sum_{i=1}^{m} \hat{\boldsymbol{x}}_i \cdot \frac{\partial\left(\frac{e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}}{1+e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}}\right)}{\partial \boldsymbol{\beta}^{\mathrm{T}}}$$

$$\frac{\partial^2 \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^{\mathrm{T}}} = \sum_{i=1}^{m} \hat{\boldsymbol{x}}_i \cdot \hat{\boldsymbol{x}}_i^{\mathrm{T}} \cdot \frac{e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}}{1+e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}} \cdot \frac{1}{1+e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}}$$

$$= \sum_{i=1}^{m} \hat{\boldsymbol{x}}_i \hat{\boldsymbol{x}}_i^{\mathrm{T}} p_1\left(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta}\right)\left(1 - p_1\left(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta}\right)\right)$$

Binary Classification - Linear Discriminant Analysis

- Linear Discriminant Analysis [Fisher, 1936]



LDA is also considered as a classic supervised dimensionality reduction technique

Binary Classification - Linear Discriminant Analysis

- The idea of LDA: project the same class samples onto a line, while samples of different classes are far away from each other.
  - To make the projection points of similar samples as close as possible, we can make the covariance of the projection points of similar samples as small as possible
  - To make the projection points of examples from different classes as far away as possible, we can make the distance between the class centers as large as possible

- Some variables
  - the sample set of the i-th class X_i
  - the mean vector of the i-th class $\boldsymbol{\mu}_i$
  - the covariance matrix of the i-th class $\boldsymbol{\Sigma}_i$
  - the centers of those two classes samples $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\mu}_0$ and $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\mu}_1$
  - the covariances of the two classes samples $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Sigma}_0\boldsymbol{w}$ and $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Sigma}_1\boldsymbol{w}$

Binary Classification - Linear Discriminant Analysis

- We have the objective to be maximized

$$J = \frac{\left\| \boldsymbol{w}^{\mathrm{T}} \boldsymbol{\mu}_0 - \boldsymbol{w}^{\mathrm{T}} \boldsymbol{\mu}_1 \right\|_2^2}{\boldsymbol{w}^{\mathrm{T}} \boldsymbol{\Sigma}_0 \boldsymbol{w} + \boldsymbol{w}^{\mathrm{T}} \boldsymbol{\Sigma}_1 \boldsymbol{w}}$$
$$= \frac{\boldsymbol{w}^{\mathrm{T}} \left(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1 \right) \left(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1 \right)^{\mathrm{T}} \boldsymbol{w}}{\boldsymbol{w}^{\mathrm{T}} \left(\boldsymbol{\Sigma}_0 + \boldsymbol{\Sigma}_1 \right) \boldsymbol{w}}$$

To make the projection points of similar samples as close as possible, we can make the covariance of the projection points of similar samples as small as possible.
To make the projection points of examples from different classes as far away as possible, we can make the distance between the class centers as large as possible

- The within-class scatter matrix

$$\mathbf{S}_w = \boldsymbol{\Sigma}_0 + \boldsymbol{\Sigma}_1$$
$$= \sum_{\boldsymbol{x} \in X_0} \left(\boldsymbol{x} - \boldsymbol{\mu}_0 \right) \left(\boldsymbol{x} - \boldsymbol{\mu}_0 \right)^{\mathrm{T}} + \sum_{\boldsymbol{x} \in X_1} \left(\boldsymbol{x} - \boldsymbol{\mu}_1 \right) \left(\boldsymbol{x} - \boldsymbol{\mu}_1 \right)^{\mathrm{T}}$$

- The between-class scatter matrix

$$\mathbf{S}_b = \left(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1 \right) \left(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1 \right)^{\mathrm{T}}$$

Binary Classification - Linear Discriminant Analysis

- Generalized Rayleigh quotient

$$J = \frac{\boldsymbol{w}^{\mathrm{T}}\mathbf{S}_b\boldsymbol{w}}{\boldsymbol{w}^{\mathrm{T}}\mathbf{S}_w\boldsymbol{w}}$$

- Let $\boldsymbol{w}^{\mathrm{T}}\mathbf{S}_w\boldsymbol{w} = 1$, maximizing generalized Rayleigh quotient is equivalent to

$$\min_{\boldsymbol{w}} \; -\boldsymbol{w}^{\mathrm{T}}\mathbf{S}_b\boldsymbol{w}$$

$$\text{s.t. } \boldsymbol{w}^{\mathrm{T}}\mathbf{S}_w\boldsymbol{w} = 1$$

- Using the method of Lagrange multipliers

$$L(\boldsymbol{w}, \lambda) = -\boldsymbol{w}^{\mathrm{T}}\mathbf{S}_b\boldsymbol{w} + \lambda(\boldsymbol{w}^{\mathrm{T}}\mathbf{S}_w\boldsymbol{w} - 1)$$

Binary Classification - Linear Discriminant Analysis

the derivatives are:

$$\frac{\partial L(\boldsymbol{w},\lambda)}{\partial \boldsymbol{w}} = -\frac{\partial(\boldsymbol{w}^{\mathrm{T}}\mathbf{S}_b\boldsymbol{w})}{\partial \boldsymbol{w}} + \lambda\frac{\partial(\boldsymbol{w}^{\mathrm{T}}\mathbf{S}_w\boldsymbol{w}-1)}{\partial \boldsymbol{w}}$$

$$= -(\mathbf{S}_b + \mathbf{S}_b^{\mathrm{T}})\boldsymbol{w} + \lambda(\mathbf{S}_w + \mathbf{S}_w^{\mathrm{T}})\boldsymbol{w}$$

since $\mathbf{S}_b = \mathbf{S}_b^{\mathrm{T}}, \mathbf{S}_w = \mathbf{S}_w^{\mathrm{T}},$

$$\frac{\partial L(\boldsymbol{w},\lambda)}{\partial \boldsymbol{w}} = -2\mathbf{S}_b\boldsymbol{w} + 2\lambda\mathbf{S}_w\boldsymbol{w}$$

Let it be 0:

$$-2\mathbf{S}_b\boldsymbol{w} + 2\lambda\mathbf{S}_w\boldsymbol{w} = 0$$

$$\mathbf{S}_b\boldsymbol{w} = \lambda\mathbf{S}_w\boldsymbol{w}$$

$$(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)^{\mathrm{T}}\boldsymbol{w} = \lambda\mathbf{S}_w\boldsymbol{w}$$

Binary Classification - Linear Discriminant Analysis

$$(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)^{\mathrm{T}}\boldsymbol{w} = \lambda \mathbf{S}_w \boldsymbol{w}$$

Let $(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)^{\mathrm{T}}\boldsymbol{w} = \gamma$:

$$\gamma(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1) = \lambda \mathbf{S}_w \boldsymbol{w}$$

$$\boldsymbol{w} = \frac{\gamma}{\lambda}\mathbf{S}_w^{-1}(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)$$

Since the final solution of w only concerns its direction rather than its magnitude, its magnitude can be assigned arbitrarily. Moreover, as μ_0 and μ_1 have fixed magnitudes, γ is affected only by the magnitude of w. Therefore, by adjusting the magnitude of w, we can make $\gamma = \lambda$.

It finally gives: $\boldsymbol{w} = \mathbf{S}_w^{-1}(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)$

Extend LDA to multiclass classification problems

- The global scatter matrix

$$\mathbf{S}_t = \mathbf{S}_b + \mathbf{S}_w$$

$$= \sum_{i=1}^{m} (\boldsymbol{x}_i - \boldsymbol{\mu}) (\boldsymbol{x}_i - \boldsymbol{\mu})^T$$

- The within-class scatter matrix

$$\mathbf{S}_w = \sum_{i-1}^{N} \mathbf{S}_{w_i}$$

where $$\mathbf{S}_{w_i} = \sum_{\boldsymbol{x} \in X_i} (\boldsymbol{x} - \boldsymbol{\mu}_i) (\boldsymbol{x} - \boldsymbol{\mu}_i)^T$$

$$\mathbf{S}_b = \mathbf{S}_t - \mathbf{S}_w$$

- We have

$$= \sum_{i=1}^{N} m_i (\boldsymbol{\mu}_i - \boldsymbol{\mu}) (\boldsymbol{\mu}_i - \boldsymbol{\mu})^T$$

Extend LDA to multiclass classification problems

$$
\begin{aligned}
\mathbf{S}_b &= \mathbf{S}_t - \mathbf{S}_w \\
&= \sum_{i=1}^{m} (\boldsymbol{x}_i - \boldsymbol{\mu})(\boldsymbol{x}_i - \boldsymbol{\mu})^{\mathrm{T}} - \sum_{i=1}^{N} \sum_{\boldsymbol{x} \in X_i} (\boldsymbol{x} - \boldsymbol{\mu}_i)(\boldsymbol{x} - \boldsymbol{\mu}_i)^{\mathrm{T}} \\
&= \sum_{i=1}^{N} \left(\sum_{\boldsymbol{x} \in X_i} \left((\boldsymbol{x} - \boldsymbol{\mu})(\boldsymbol{x} - \boldsymbol{\mu})^{\mathrm{T}} - (\boldsymbol{x} - \boldsymbol{\mu}_i)(\boldsymbol{x} - \boldsymbol{\mu}_i)^{\mathrm{T}} \right) \right) \\
&= \sum_{i=1}^{N} \left(\sum_{\boldsymbol{x} \in X_i} \left((\boldsymbol{x} - \boldsymbol{\mu})(\boldsymbol{x}^{\mathrm{T}} - \boldsymbol{\mu}^{\mathrm{T}}) - (\boldsymbol{x} - \boldsymbol{\mu}_i)(\boldsymbol{x}^{\mathrm{T}} - \boldsymbol{\mu}_i^{\mathrm{T}}) \right) \right) \\
&= \sum_{i=1}^{N} \left(\sum_{\boldsymbol{x} \in X_i} \left(\boldsymbol{x}\boldsymbol{x}^{\mathrm{T}} - \boldsymbol{x}\boldsymbol{\mu}^{\mathrm{T}} - \boldsymbol{\mu}\boldsymbol{x}^{\mathrm{T}} + \boldsymbol{\mu}\boldsymbol{\mu}^{\mathrm{T}} - \boldsymbol{x}\boldsymbol{x}^{\mathrm{T}} + \boldsymbol{x}\boldsymbol{\mu}_i^{\mathrm{T}} + \boldsymbol{\mu}_i\boldsymbol{x}^{\mathrm{T}} - \boldsymbol{\mu}_i\boldsymbol{\mu}_i^{\mathrm{T}} \right) \right) \\
&= \sum_{i=1}^{N} \left(\sum_{\boldsymbol{x} \in X_i} \left(-\boldsymbol{x}\boldsymbol{\mu}^{\mathrm{T}} - \boldsymbol{\mu}\boldsymbol{x}^{\mathrm{T}} + \boldsymbol{\mu}\boldsymbol{\mu}^{\mathrm{T}} + \boldsymbol{x}\boldsymbol{\mu}_i^{\mathrm{T}} + \boldsymbol{\mu}_i\boldsymbol{x}^{\mathrm{T}} - \boldsymbol{\mu}_i\boldsymbol{\mu}_i^{\mathrm{T}} \right) \right) \\
&= \sum_{i=1}^{N} \left(- \sum_{\boldsymbol{x} \in X_i} \boldsymbol{x}\boldsymbol{\mu}^{\mathrm{T}} - \sum_{\boldsymbol{x} \in X_i} \boldsymbol{\mu}\boldsymbol{x}^{\mathrm{T}} + \sum_{\boldsymbol{x} \in X_i} \boldsymbol{\mu}\boldsymbol{\mu}^{\mathrm{T}} + \sum_{\boldsymbol{x} \in X_i} \boldsymbol{x}\boldsymbol{\mu}_i^{\mathrm{T}} + \sum_{\boldsymbol{x} \in X_i} \boldsymbol{\mu}_i\boldsymbol{x}^{\mathrm{T}} - \sum_{\boldsymbol{x} \in X_i} \boldsymbol{\mu}_i\boldsymbol{\mu}_i^{\mathrm{T}} \right) \\
&= \sum_{i=1}^{N} \left(-m_i\boldsymbol{\mu}_i\boldsymbol{\mu}^{\mathrm{T}} - m_i\boldsymbol{\mu}\boldsymbol{\mu}_i^{\mathrm{T}} + m_i\boldsymbol{\mu}\boldsymbol{\mu}^{\mathrm{T}} + m_i\boldsymbol{\mu}_i\boldsymbol{\mu}_i^{\mathrm{T}} + m_i\boldsymbol{\mu}_i\boldsymbol{\mu}_i^{\mathrm{T}} - m_i\boldsymbol{\mu}_i\boldsymbol{\mu}_i^{\mathrm{T}} \right) \\
&= \sum_{i=1}^{N} \left(-m_i\boldsymbol{\mu}_i\boldsymbol{\mu}^{\mathrm{T}} - m_i\boldsymbol{\mu}\boldsymbol{\mu}_i^{\mathrm{T}} + m_i\boldsymbol{\mu}\boldsymbol{\mu}^{\mathrm{T}} + m_i\boldsymbol{\mu}_i\boldsymbol{\mu}_i^{\mathrm{T}} \right) \\
&= \sum_{i=1}^{N} m_i \left(-\boldsymbol{\mu}_i\boldsymbol{\mu}^{\mathrm{T}} - \boldsymbol{\mu}\boldsymbol{\mu}_i^{\mathrm{T}} + \boldsymbol{\mu}\boldsymbol{\mu}^{\mathrm{T}} + \boldsymbol{\mu}_i\boldsymbol{\mu}_i^{\mathrm{T}} \right) \\
&= \sum_{i=1}^{N} m_i (\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^{\mathrm{T}}
\end{aligned}
$$

Extend LDA to multiclass classification problems

- Objective

$$\max_{\mathbf{W}} \frac{\mathrm{tr}\left(\mathbf{W}^{\mathrm{T}}\mathbf{S}_b\mathbf{W}\right)}{\mathrm{tr}\left(\mathbf{W}^{\mathrm{T}}\mathbf{S}_w\mathbf{W}\right)}$$

$$\mathbf{W} = (\boldsymbol{w}_1, \boldsymbol{w}_2, ..., \boldsymbol{w}_i, ..., \boldsymbol{w}_{N-1}) \in \mathbb{R}^{d\times(N-1)}$$

$$\begin{cases} \mathrm{tr}(\mathbf{W}^{\mathrm{T}}\mathbf{S}_b\mathbf{W}) = \sum_{i=1}^{N-1} \boldsymbol{w}_i^{\mathrm{T}}\mathbf{S}_b\boldsymbol{w}_i \\ \mathrm{tr}(\mathbf{W}^{\mathrm{T}}\mathbf{S}_w\mathbf{W}) = \sum_{i=1}^{N-1} \boldsymbol{w}_i^{\mathrm{T}}\mathbf{S}_w\boldsymbol{w}_i \end{cases}$$

where $\mathbf{W} \in \mathbb{R}^{d\times(N-1)}$

$$\mathbf{S}_b\mathbf{W} = \lambda\mathbf{S}_w\mathbf{W}$$

Concatenating the eigenvectors corresponding to the d' largest non-zero eigenvalues of $\mathbf{S}_w^{-1}\mathbf{S}_b$ leads to the closed-form solution of $\mathbf{W}$, where $d' \leq N-1$

- Since the projection reduces the data while considering the class information, LDA is also considered as a classic supervised dimensionality reduction technique

Multiclass Classification

- Multiclass Classification learning methods
  - Some binary classification methods can be directly extended to accommodate multiclass cases
  - Apply some strategies to solve multiclass classification problems with any existing binary classification methods (more general)
    - Decompose the problem and then train a binary classifier for each divided binary classification problem
    - Ensemble the outputs collected from all binary classifiers into the final multiclass predictions

- Dividing strategies
  - One vs. One (OvO)
  - One vs. Rest (OvR)
  - Many vs. Many (MvM)

Multiclass Classification - OvO

- In the decomposing phase
  - puts the N classes into pairs
    - N(N-1)/2 binary classification tasks
  - trains a classifier for each task
    - N(N-1)/2 classifiers

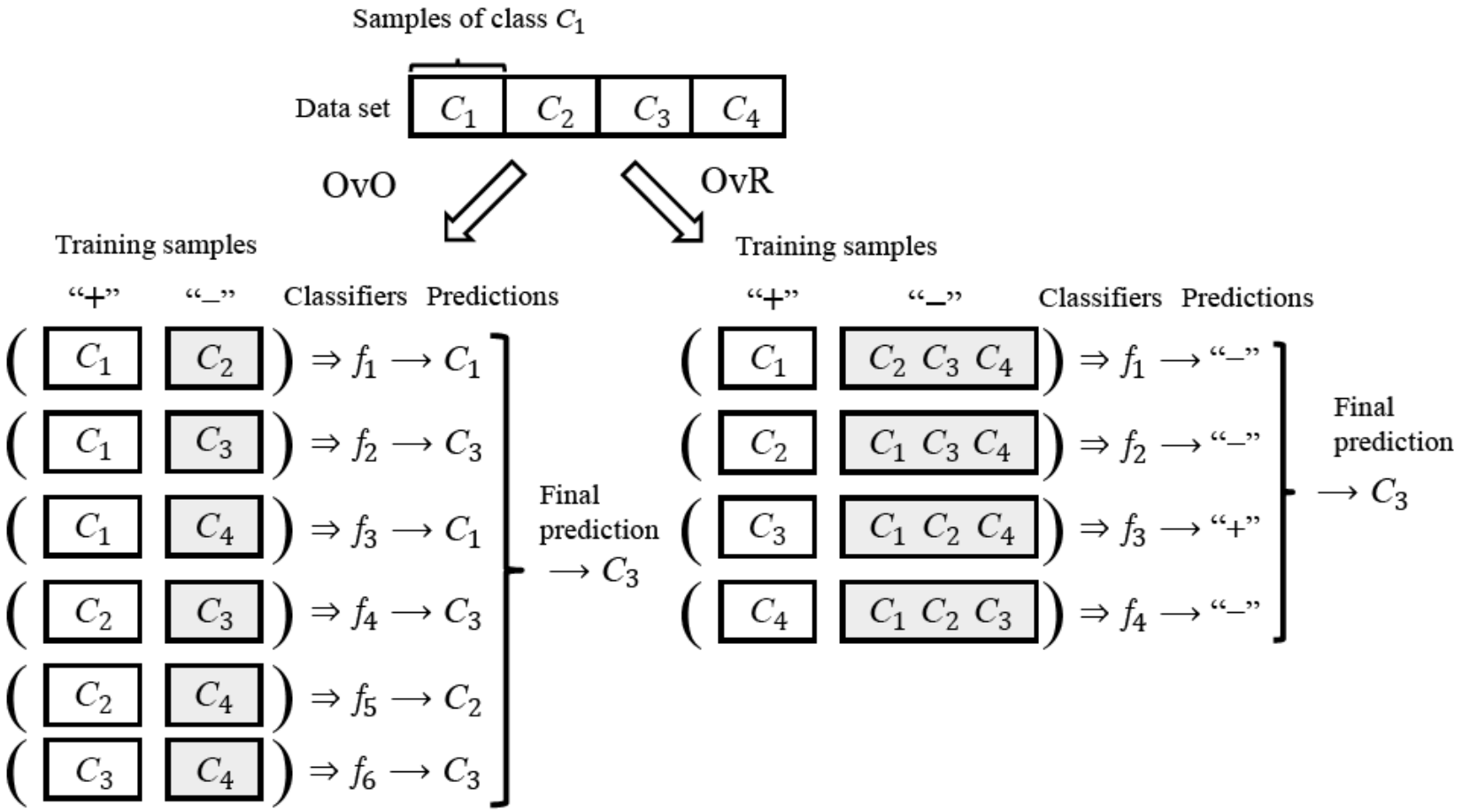
- In the testing phase
  - a new sample is classified by all classifiers
    - N(N-1)/2 classification outputs
  - the final prediction can be made via voting
    - the predicted class is the one received the most votes

Multiclass Classification - OvR

- In the decomposing phase
  - consider each class as positive in turn, and the rest classes are considered as negative
    - N binary classification tasks
  - trains a classifier for each task
    - N classifiers

- In the testing phase
  - a new sample is classified by all classifiers
    - N classification outputs
  - the prediction confidences are usually assessed
    - the class with the highest confidence is used as the classification result

Multiclass Classification – A comparison between OvO and OvR



Multiclass Classification – A comparison between OvO and OvR

OvO

- Train N(N-1)/2 classifiers, the memory and testing time costs are often higher
- Each classifier uses only samples of two classes. Hence, the computational cost of training OvO is lower

OvR

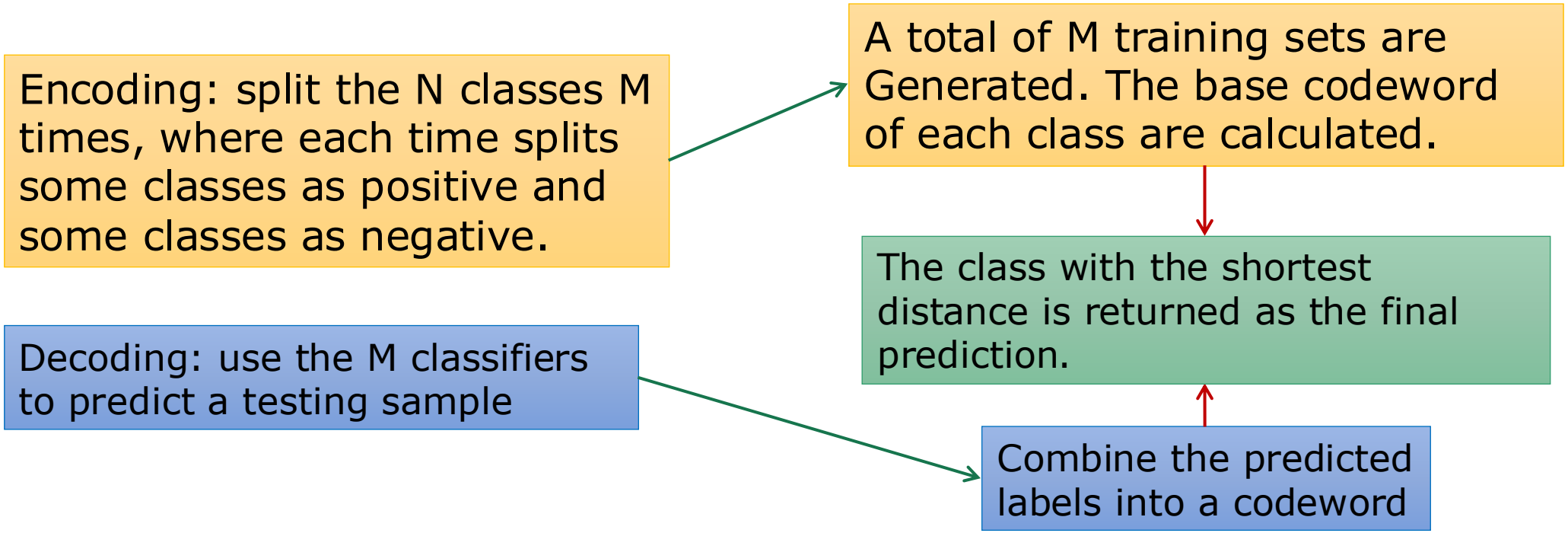
- Train N classifiers, the memory and testing time costs are often lower
- Each classifier uses all training samples. Hence, the computational cost of training OvO is higher

As for the prediction performance, it depends on the specific data distribution, and in most cases, the two methods have similar performance.

Multiclass Classification - MvM

- Many vs Many(MvM)
  - MvM conducts multiple trials, and each trial puts several classes as positive and several classes as negative.

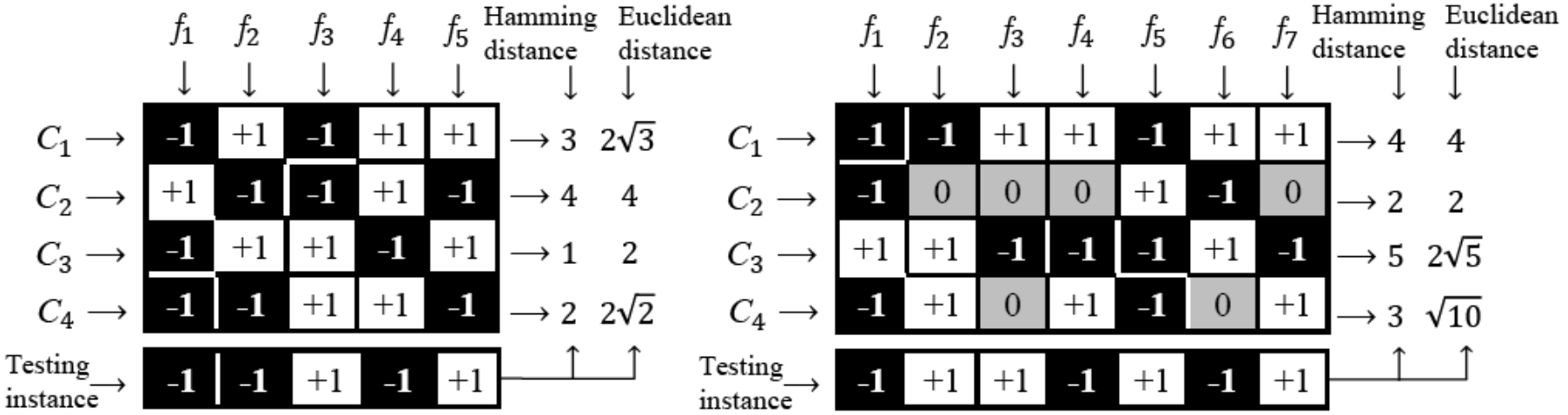
- Error Correcting Output Code, ECOC



Multiclass Classification - MvM

- Error Correcting Output Code, ECOC

Deactivated class, Hamming distance is 0.5 not 1



	f_1	f_2	f_3	f_4	f_5	Hamming distance	Euclidean distance
C_1	-1	+1	-1	+1	+1	3	$2\sqrt{3}$
C_2	+1	-1	-1	+1	-1	4	4
C_3	-1	+1	+1	-1	+1	1	2
C_4	-1	-1	+1	+1	-1	2	$2\sqrt{2}$
Testing instance	-1	-1	+1	-1	+1		

(a) Binary ECOC coding.

[Dietterich and Bakiri,1995]

	f_1	f_2	f_3	f_4	f_5	f_6	f_7	Hamming distance	Euclidean distance
C_1	-1	-1	+1	+1	-1	+1	+1	4	4
C_2	-1	0	0	0	+1	-1	0	2	2
C_3	+1	+1	-1	-1	-1	+1	-1	5	$2\sqrt{5}$
C_4	-1	+1	0	+1	-1	0	+1	3	$\sqrt{10}$
Testing instance	-1	+1	+1	-1	+1	-1	+1		

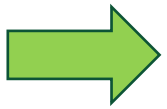
(b) Ternary ECOC coding.

[Allwein et al. 2000]

- The ECOC codeword has the error tolerance and correction ability. a longer ECOC codeword produces better correction ability
- In theory, the correction ability of a fixed length codeword increases as the distances between classes increase.

Class Imbalance Problem

- Class imbalance
  - a significantly different number of samples for each class. (the positive class is the minority)

the classes are balanced $\frac{y}{1-y} > 1$  $\frac{y}{1-y} > \frac{m^+}{m^-}$ the observed class ratio

- Rescaling
  - undersampling
    - some negative samples are selectively dropped so that the classes are balanced (EasyEnsemble [Liu et al.,2009])
  - oversampling
    - increase the number of positive samples so that the classes are balanced (SMOTE [Chawla et al.2002])
  - threshold-moving

$$\frac{y}{1-y} > \frac{m^+}{m^-}$$

Summary

- The objective of optimization of each model
  - Least squares method: Minimize the mean-square error
  - Logistic regression: Maximizing the likelihood of sample distribution
  - Linear discriminant analysis: minimize the within-class scatter matrix and maximize the between-classes scatter matrix
- Optimization method of parameters
  - Least squares method: Linear Algebra
  - Logistic regression: convex optimization, Newton's method
  - Linear discriminant analysis: matrix theory, generalized Rayleigh quotient

Summary

- Linear Regression
  - Least Squares Method (minimize the mean-square error)
- Binary Classification Problem
  - Logistic Regression
    - Unit-step function, logistic function, maximum likelihood method
  - Linear Discriminant Analysis
    - Maximizing generalized Rayleigh quotient
- Multiclass Classification Problem
  - One vs. One (OvO)
  - One vs. Rest (OvR)
  - Many vs. Many (MvM)
    - Error Correcting Output Code
- Class Imbalance Problem
  - Strategy: rescaling

Thanks!