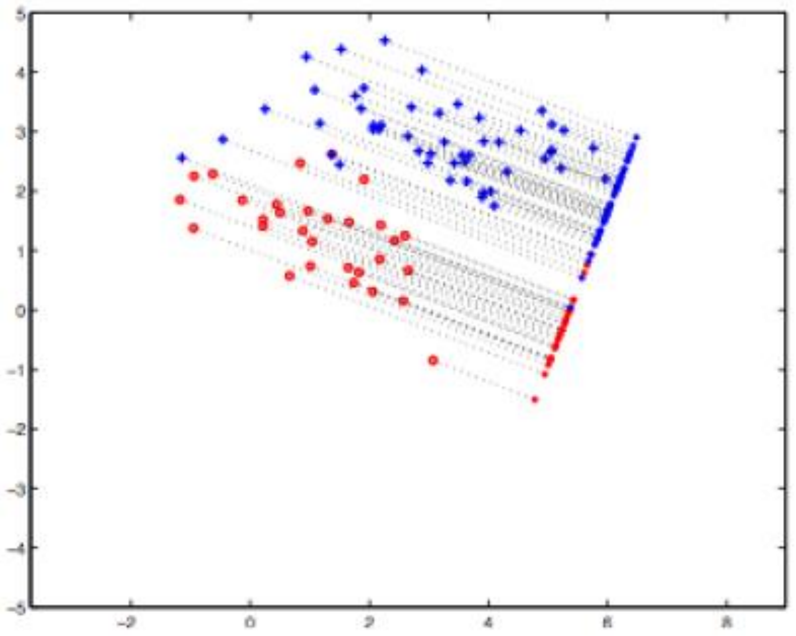
Machine Learning

Contents

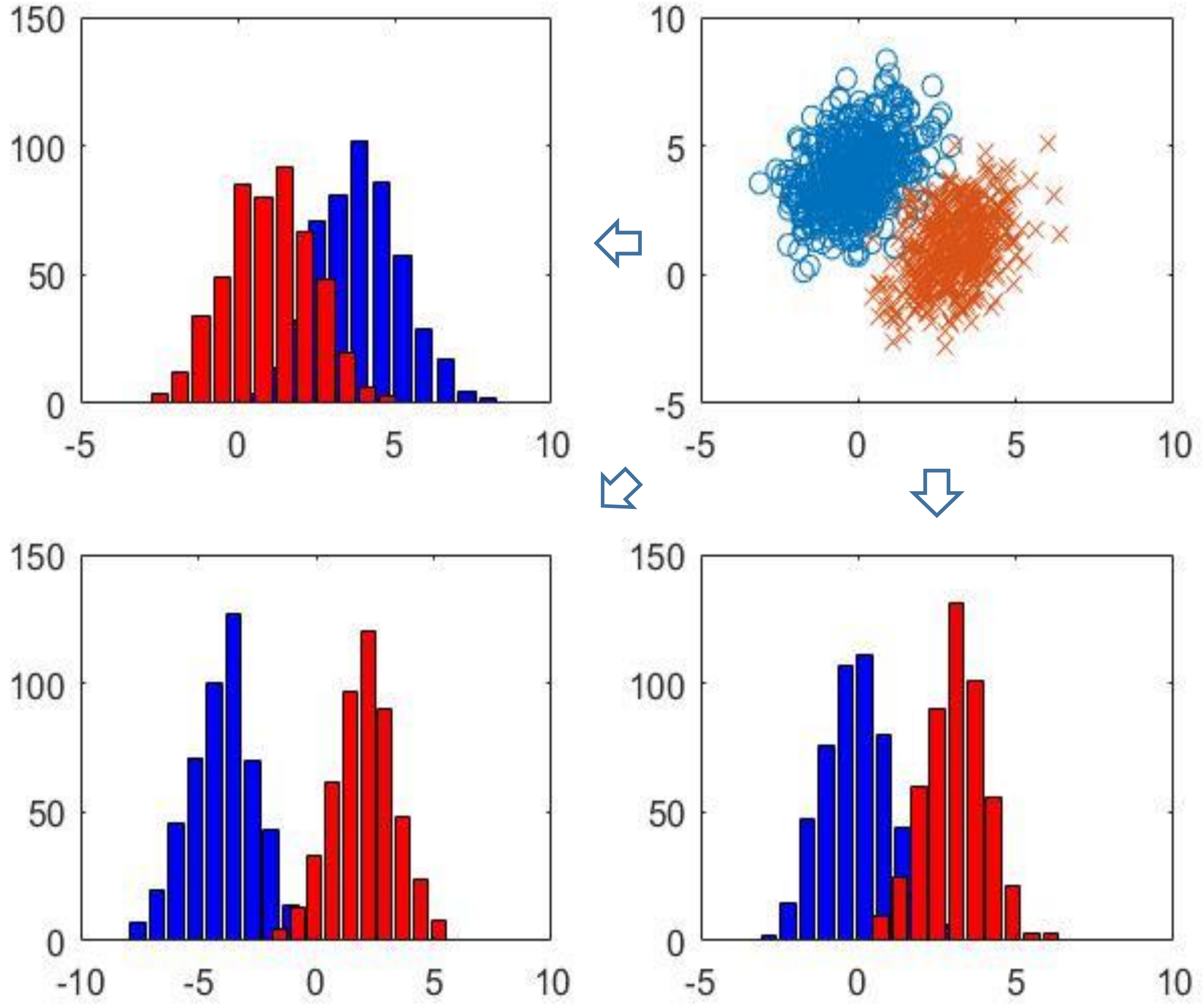
3.1. Linear Discriminant Analysis (LDA)

- Problem: Given a set of data points, each of which is labelled with a class, find the best set of basis vectors for projecting the data points such that classification is improved.

- Idea: Form the projection such that the variability across the different classes is maximized, while the variability within each class is minimized.

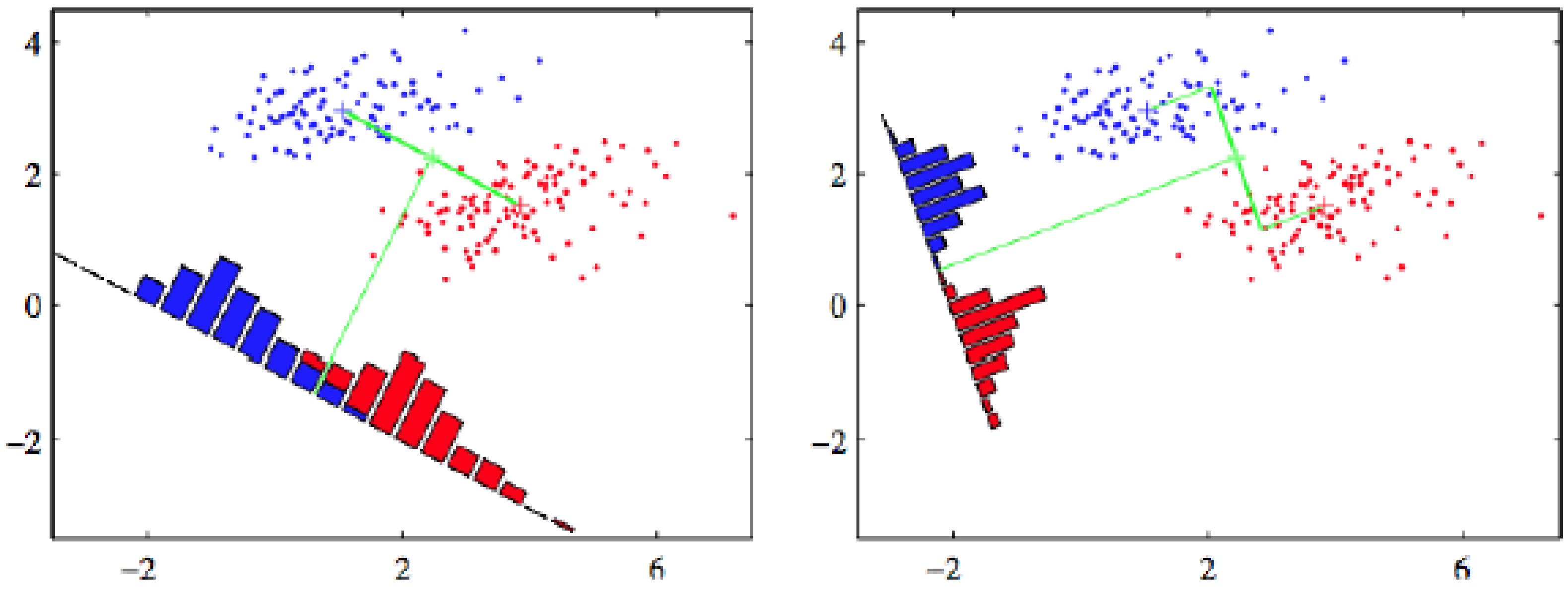


Which one is the best projection?



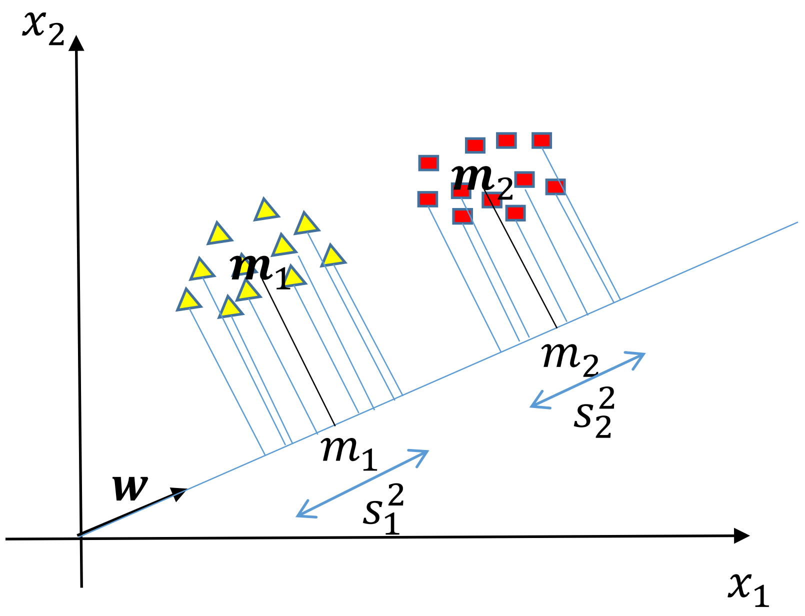
LDA: Maximize Difference of Means

- Considering the simple problem of projecting onto one dimension
- The two classes should be well separated in this single dimension
- A simple idea is to maximize the difference of the means in the projected space
- What is a problem with this solution?



LDA: A Better Idea

- Fisher's idea: Fisher's linear discriminant
  - Maximize a function that will give a large separation between the projected class means
  - While also giving a small variance within each class, thereby minimizing the class overlap



3.2. Projection with Basis Vectors

- Point A = $(a_1, a_2)^T$
- Basis Vector **w** = $(w_1, w_2)^T$
- Projection $\boldsymbol{w}^T A = w_1\ a_1 + w_2\ a_2$

참조:

T(transpose)는 행렬 혹은 벡터에서 행과 열의 위치를 교환하는 이항 연산자이다. 즉,

$$(a_1, a_2)^T = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} \tag{3.2.4}$$

이고

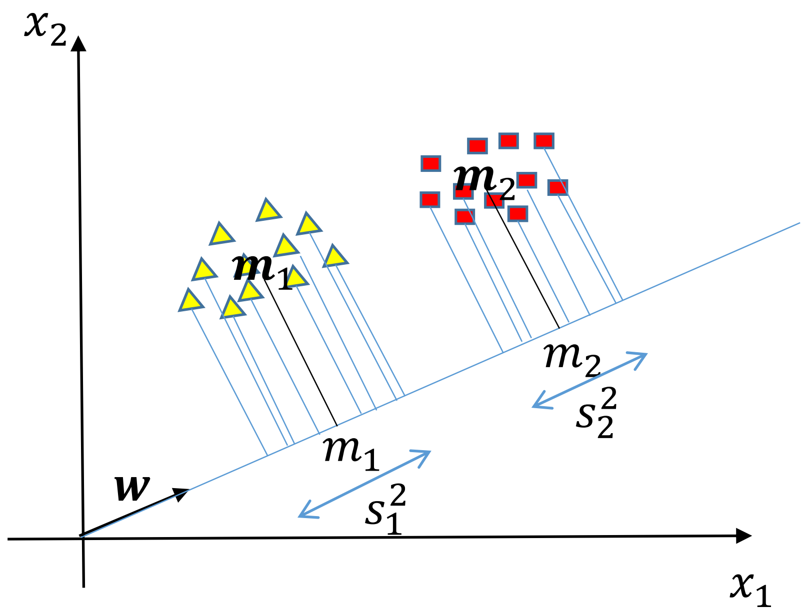
$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}^T = \begin{pmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \end{pmatrix} \tag{3.2.5}$$

이다. 또한 행렬 A와 B의 곱에 대하여 이항 연산자 T를 적용하면

$$(AB)^T = B^T A^T \tag{3.2.6}$$

이 된다.

3.3. LDA Formulation



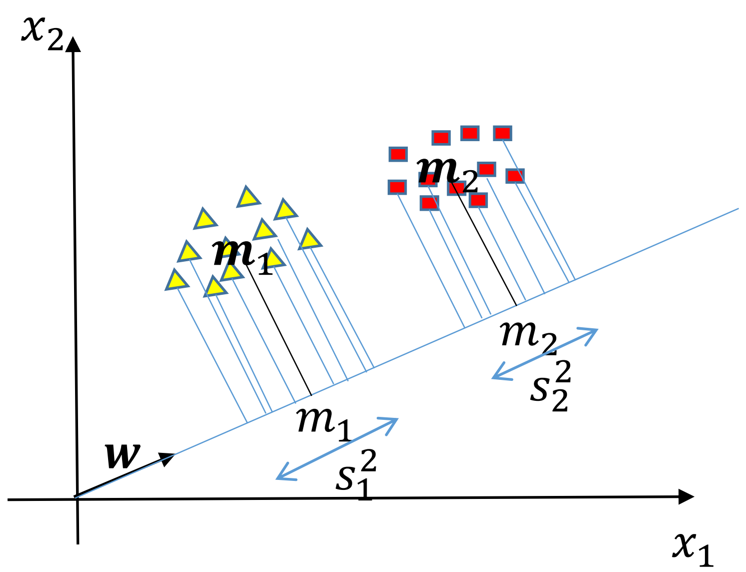
$$J(\boldsymbol{w}) = \frac{(m_1 - m_2)^2}{s_1^2 + s_2^2} \qquad (3.3.1)$$

$$m_1 = \frac{\sum_{i \in C_1} \boldsymbol{w}^T \boldsymbol{x}^i}{N_1} = \frac{\sum_t \boldsymbol{w}^T \boldsymbol{x}^t r^t}{\sum_t r^t} = \boldsymbol{w}^T \frac{\sum_t \boldsymbol{x}^t r^t}{\sum_t r^t} = \boldsymbol{w}^T \boldsymbol{m}_1 \qquad (3.3.2)$$

$$s_1^2 = \sum_t \left(\boldsymbol{w}^T \boldsymbol{x}^t - m_1\right)^2 r^t \qquad (3.3.3)$$

$$\begin{aligned} (m_1 - m_2)^2 &= (\boldsymbol{w}^T \boldsymbol{m}_1 - \boldsymbol{w}^T \boldsymbol{m}_2)^2 = (\boldsymbol{w}^T (\boldsymbol{m}_1 - \boldsymbol{m}_2))^2 \qquad (3.3.4) \\ &= \boldsymbol{w}^T (\boldsymbol{m}_1 - \boldsymbol{m}_2)(\boldsymbol{m}_1 - \boldsymbol{m}_2)^T \boldsymbol{w} = \boldsymbol{w}^T S_B \boldsymbol{w} \end{aligned}$$

$$S_B = (\boldsymbol{m}_1 - \boldsymbol{m}_2)(\boldsymbol{m}_1 - \boldsymbol{m}_2)^T \qquad (3.3.5)$$



$$J(\boldsymbol{w}) = \frac{(m_1 - m_2)^2}{s_1^2 + s_2^2} \qquad (3.3.1)$$

$$s_1^2 = \sum_t \left(\boldsymbol{w}^T \boldsymbol{x}^t - m_1\right)^2 r^t = \sum_t \left(\boldsymbol{w}^T \boldsymbol{x}^t - \boldsymbol{w}^T \boldsymbol{m}_1\right)^2 r^t = \sum_t \left(\boldsymbol{w}^T (\boldsymbol{x}^t - \boldsymbol{m}_1)\right)^2 r^t \qquad (3.3.6)$$
$$= \sum_t \boldsymbol{w}^T (\boldsymbol{x}^t - \boldsymbol{m}_1)(\boldsymbol{x}^t - \boldsymbol{m}_1)^T \boldsymbol{w} r^t = \boldsymbol{w}^T S_1 \boldsymbol{w}$$

$$S_1 = \sum_t (\boldsymbol{x}^t - \boldsymbol{m}_1)(\boldsymbol{x}^t - \boldsymbol{m}_1)^T r^t \qquad (3.3.7)$$

$$s_1^2 + s_2^2 = \boldsymbol{w}^T S_W \boldsymbol{w} \qquad (3.3.8)$$

$$S_W = S_1 + S_2 \qquad (3.3.9)$$

$$J(\boldsymbol{w}) = \frac{\boldsymbol{w}^T S_B \boldsymbol{w}}{\boldsymbol{w}^T S_W \boldsymbol{w}} \qquad (3.3.10)$$

예제 3.3-1

행렬 $\boldsymbol{A} = (a_{ij})$가 $d \times d$ 정방행렬이고, 다음 함수

$$g(\boldsymbol{w}) = \boldsymbol{w}^T \boldsymbol{A} \boldsymbol{w} = \sum_{i=1}^{d} \sum_{j=1}^{d} w_i w_j a_{ij}$$

에 대하여 $\dfrac{\partial g}{\partial \boldsymbol{w}}$를 구하여라.

풀이

문제로 주어진 함수를 더 자세히 적으면

$$g(\boldsymbol{w}) = \boldsymbol{w}^T \boldsymbol{A}\boldsymbol{w} = \sum_{i=1}^{d}\sum_{j=1}^{d} w_i w_j a_{ij} = w_k \sum_{j=1}^{d} w_j a_{kj} + \sum_{i \neq k} w_i \sum_{j=1}^{d} w_j a_{ij}$$

이 된다. 여기에, $\sum_{j=1}^{d} w_j a_{kj} = w_k a_{kk} + \sum_{j \neq k} w_j a_{kj}$와 $\sum_{j=1}^{d} w_j a_{ij} = w_k a_{ik} + \sum_{j \neq k} w_j a_{ij}$를 대입하면

$$g(\boldsymbol{w}) = w_k \sum_{j \neq k} w_j a_{kj} + w_k^2 a_{kk} + \sum_{i \neq k} w_i w_k a_{ik} + \sum_{i \neq k}\sum_{j \neq k} w_i w_j a_{ij}$$

이다. 따라서,

$$\frac{\partial g(\boldsymbol{w})}{\partial \boldsymbol{w}} = \sum_{j \neq k} w_j a_{kj} + 2 w_k a_{kk} + \sum_{i \neq k} w_i a_{ik} = \sum_{j=1}^{d} w_j a_{kj} + \sum_{i=1}^{d} w_i a_{ik} = A\boldsymbol{w} + A^T\boldsymbol{w}$$

이 된다.

Maximizing

$$J(\boldsymbol{w}) = \frac{\boldsymbol{w}^T S_B \boldsymbol{w}}{\boldsymbol{w}^T S_W \boldsymbol{w}} \quad (3.3.10)$$

From $\frac{\partial J(\boldsymbol{w})}{\partial \boldsymbol{w}} = 0$,

$$2S_B\boldsymbol{w}(\boldsymbol{w}^T S_W \boldsymbol{w}) - (\boldsymbol{w}^T S_B \boldsymbol{w}) 2 S_W \boldsymbol{w} = 0 \quad (3.3.11)$$

$$S_B\boldsymbol{w}(\boldsymbol{w}^T S_W \boldsymbol{w}) = (\boldsymbol{w}^T S_B \boldsymbol{w}) S_W \boldsymbol{w} \quad (3.3.12)$$

$$S_B\boldsymbol{w} = \frac{(\boldsymbol{w}^T S_W \boldsymbol{w})}{(\boldsymbol{w}^T S_B \boldsymbol{w})} S_W \boldsymbol{w} = \lambda S_W \boldsymbol{w} \quad (3.3.13)$$

$$S_B = (\boldsymbol{m}_1 - \boldsymbol{m}_2)(\boldsymbol{m}_1 - \boldsymbol{m}_2)^T \quad (3.3.5)$$

⟹

$$S_B\boldsymbol{w} = (\boldsymbol{m}_1 - \boldsymbol{m}_2)(\boldsymbol{m}_1 - \boldsymbol{m}_2)^T \boldsymbol{w} \quad (3.3.14)$$

scalar

$$\boldsymbol{w} = S_W^{-1}(\boldsymbol{m}_1 - \boldsymbol{m}_2) \quad (3.3.15)$$

참조: 벡터의 미분

d차원 벡터 $\boldsymbol{w} = (w_1, w_2, \ldots, w_d)^T$에 대한 미분 가능한 스칼라 함수 g가

$$g(w_1, w_2, \ldots, w_d) = g(\boldsymbol{w}) \tag{3.3.16}$$

로 주어졌다. 그러면, 함수 g의 벡터 $\boldsymbol{w}$에 대한 미분은

$$\frac{\partial g}{\partial \boldsymbol{w}} = \begin{pmatrix} \frac{\partial g}{\partial w_1} \\ \vdots \\ \frac{\partial g}{\partial w_d} \end{pmatrix} \tag{3.3.17}$$

이다. 또한 2차 미분은

$$\frac{\partial^2 g}{\partial \boldsymbol{w}^2} = \begin{pmatrix} \frac{\partial^2 g}{\partial w_1^2} & \cdots & \frac{\partial g}{\partial w_1 \partial w_d} \\ \vdots & & \vdots \\ \frac{\partial g}{\partial w_d \partial w_1} & \cdots & \frac{\partial^2 g}{\partial w_d^2} \end{pmatrix} \tag{3.3.18}$$

이 된다. 벡터 값을 지닌 함수 g가

이 된다. 벡터 값을 지닌 함수 g가

$$\boldsymbol{g}(\boldsymbol{w}) = \begin{pmatrix} \boldsymbol{g}_1(\boldsymbol{w}) \\ \vdots \\ \boldsymbol{g}_n(\boldsymbol{w}) \end{pmatrix} \tag{3.3.19}$$

로 주어지면 g의 $\boldsymbol{w}$에 대한 Jacobian 행렬은

$$\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{w}} = \begin{pmatrix} \frac{\partial g_1}{\partial w_1} & \cdots & \frac{\partial g_n}{\partial w_1} \\ \vdots & & \vdots \\ \frac{\partial g_1}{\partial w_d} & \cdots & \frac{\partial g_n}{\partial w_d} \end{pmatrix} \tag{3.3.20}$$

이다. 또한

$$\frac{\partial f(\boldsymbol{w}) g(\boldsymbol{w})}{\partial \boldsymbol{w}} = \frac{\partial f(\boldsymbol{w})}{\partial \boldsymbol{w}} g(\boldsymbol{w}) + f(\boldsymbol{w}) \frac{\partial g(\boldsymbol{w})}{\partial \boldsymbol{w}} \tag{3.3.21}$$

$$\frac{\partial f(\boldsymbol{w}) / g(\boldsymbol{w})}{\partial \boldsymbol{w}} = \left[\frac{\partial f(\boldsymbol{w})}{\partial \boldsymbol{w}} g(\boldsymbol{w}) - f(\boldsymbol{w}) \frac{\partial g(\boldsymbol{w})}{\partial \boldsymbol{w}} \right] / g^2(\boldsymbol{w}) \tag{3.3.22}$$

$$\frac{\partial f(g(\boldsymbol{w}))}{\partial \boldsymbol{w}} = f'(g(\boldsymbol{w})) \frac{\partial g(\boldsymbol{w})}{\partial \boldsymbol{w}} \tag{3.3.23}$$

이다.

More than 2 Classes (K > 2)

- Projection from d-dim space to (c-1)-dim space:
  - $y_i = \mathbf{w}_i^T \mathbf{x}, \quad i = 1, \ldots, c-1$
  - $\mathbf{y} = W^T \mathbf{x}$ (W = d×(c-1) matrix, w_i is the i-th column vector)
  - Within-class scatter matrix for $\mathcal{C}_i$ is

$$\mathrm{S_i} = \sum_{\mathrm{t}} \mathrm{r_i^t}(\mathrm{x^t} - \mathrm{m_i})(\mathrm{x^t} - \mathrm{m_i})^{\mathrm{T}}$$

    where $r_i^t = 1$ if $x^t \in C_i$ and 0 otherwise.
  - The total within-class scatter is

$$\mathrm{S_W} = \sum_{\mathrm{i=1}}^{\mathrm{K}} \mathrm{S_i}$$

  - The between-class scatter matrix is

$$\mathrm{S_B} = \sum_{\mathrm{i=1}}^{\mathrm{K}} \mathrm{N_i}(\mathrm{m_i} - \mathrm{m})(\mathrm{m_i} - \mathrm{m})^{\mathrm{T}}$$

    where $N_i = \sum_t r_i^t$

More than 2 Classes (K > 2)

- The between-class scatter matrix after projection is:
- The within-class scatter matrix after projection is:
- Thus we need to find the matrix $\mathbf{W}$ that maximizes

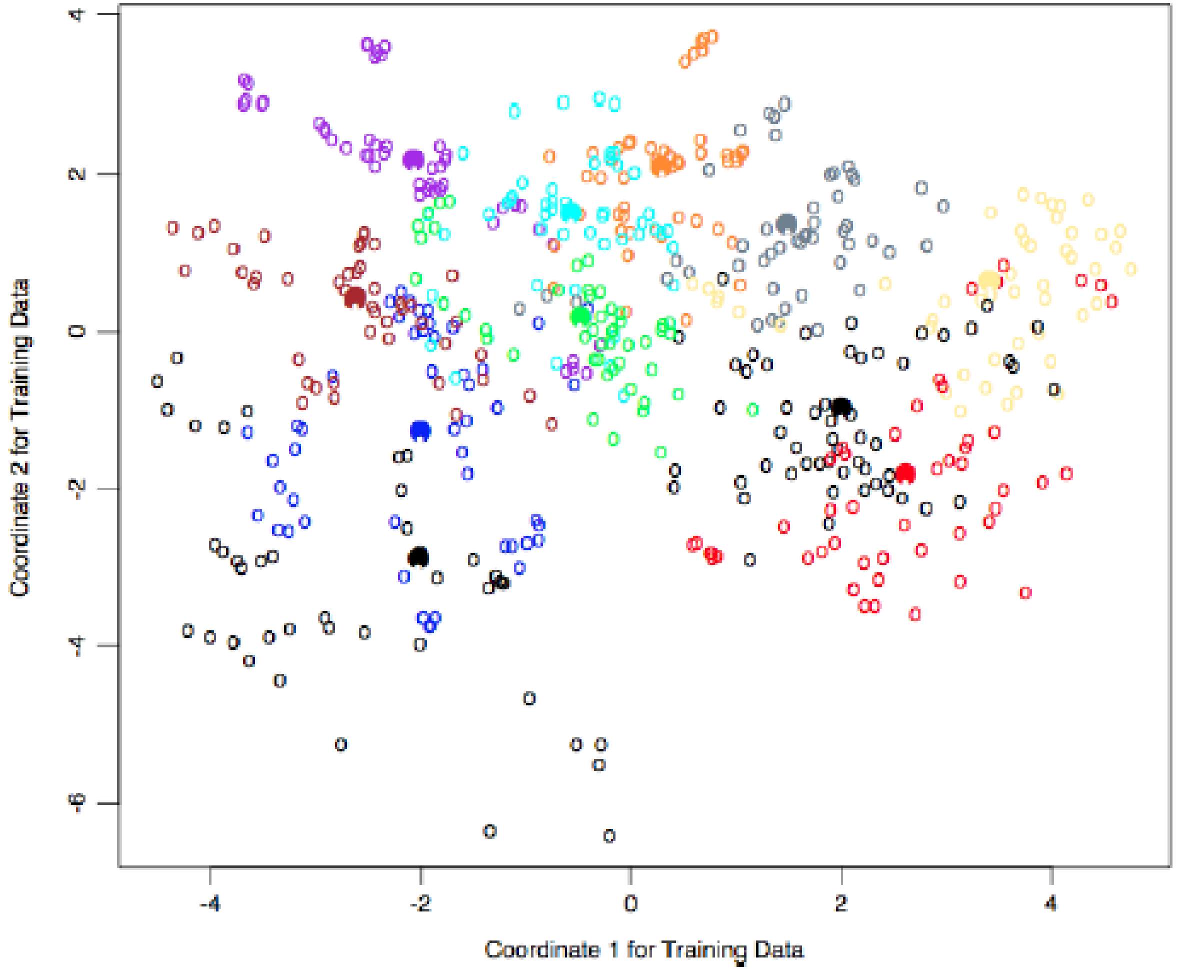
$$J(\mathbf{W}) = \frac{|\mathbf{W}^{\mathrm{T}}\mathrm{S}_{\mathrm{B}}\mathbf{W}|}{|\mathbf{W}^{\mathrm{T}}\mathrm{S}_{\mathrm{W}}\mathbf{W}|}$$

- The largest eigenvectors of $\mathrm{S}_{\mathrm{W}}^{-1}\mathrm{S}_{\mathrm{B}}$ are the solution.

LDA on Vowels Data

- 11 vowels from words spoken by fteen speakers. (Source: David Deterding, Mahesan Niranjan, Tony Robinson, see Hastie book website for data)

vowel	word	vowel	word
i	heed	O	hod
I	hid	C:	hoard
E	head	U	hood
A	had	u:	who'd
a:	hard	3:	heard
Y	hud		



LDA on Vowels Data: Decision Boundaries