Линейный Дискриминантный Анализ

- Учитывая помеченные данные, состоящие из d-мерных точек x_i вместе с их классами y_i , цель линейного дискриминантного анализа (LDA) состоит в том, чтобы найти вектор w, который максимизирует разделение между классами после проекции на w. Вспомним из Главы 7, что первым главным составляющим является вектор, который максимизирует прогнозируемую дисперсию точек. Ключевое различие между анализом главных компонент и LDA заключается в том, что первый имеет дело с неразмеченными данными и пытается максимизировать дисперсию, тогда как второй имеет дело с помеченными данными и пытается максимизировать различие между классами.

ОПТИМАЛЬНЫЙ ЛИНЕЙНЫЙ ДИСКРИМИНАНТ

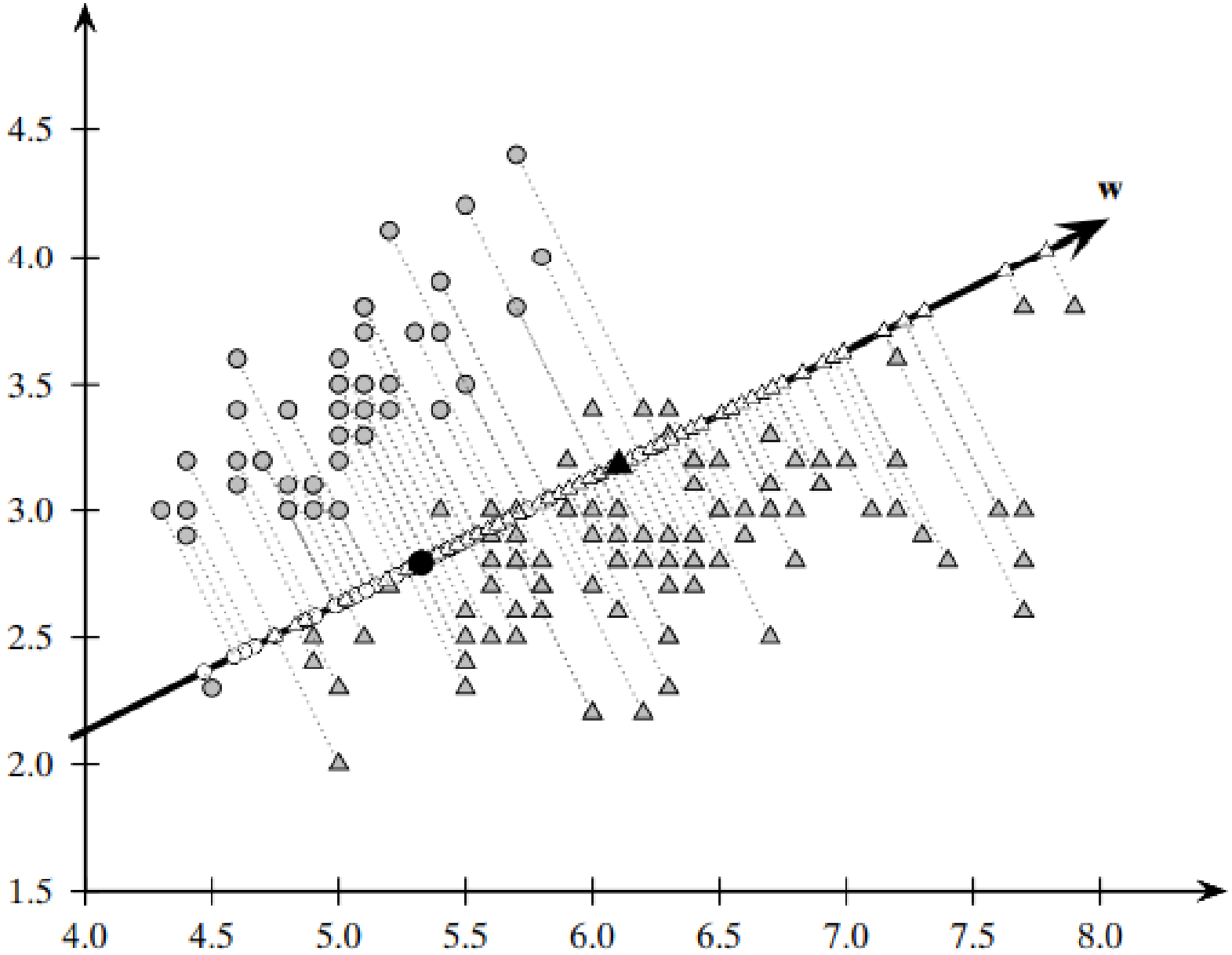
- Предположим, что набор данных **D** состоит из n помеченных точек $\{x_i, y_i\}$, где $\mathbf{x}_i \in R^d$ и $y_i \in \{c_1, c_2..., c_k\}$. Пусть $\mathbf{D}_i$ обозначает подмножество точек, помеченных классом c_i, т.е., $\mathbf{D}_i = \{\mathbf{x}_j \mid y_j = c_i\}$, и пусть $|\mathbf{D}_i| = n_i$ обозначает количество точек с классом ci. Мы предполагаем, что существует только k = 2 классов. Таким образом, набор данных **D** может быть разбит на $\mathbf{D}_1$ и $\mathbf{D}_2$.
- Пусть **w** единичный вектор, то есть, $\mathbf{w}^T\mathbf{w} = 1$. По уравнению (1.7), проекция любой d-мерной точки $\mathbf{x}_i$ на вектор **w** задаётся как

$$\mathbf{x}_i' = \left(\frac{\mathbf{w}^T\mathbf{x}_i}{\mathbf{w}^T\mathbf{w}}\right)\mathbf{w} = \left(\mathbf{w}^T\mathbf{x}_i\right)\mathbf{w} = a_i\mathbf{w}$$

- где a_i задает смещение или координату $\mathbf{x}'$i вдоль линии **w**:

$$a_i = \mathbf{w}^T\mathbf{x}_i$$

Пример



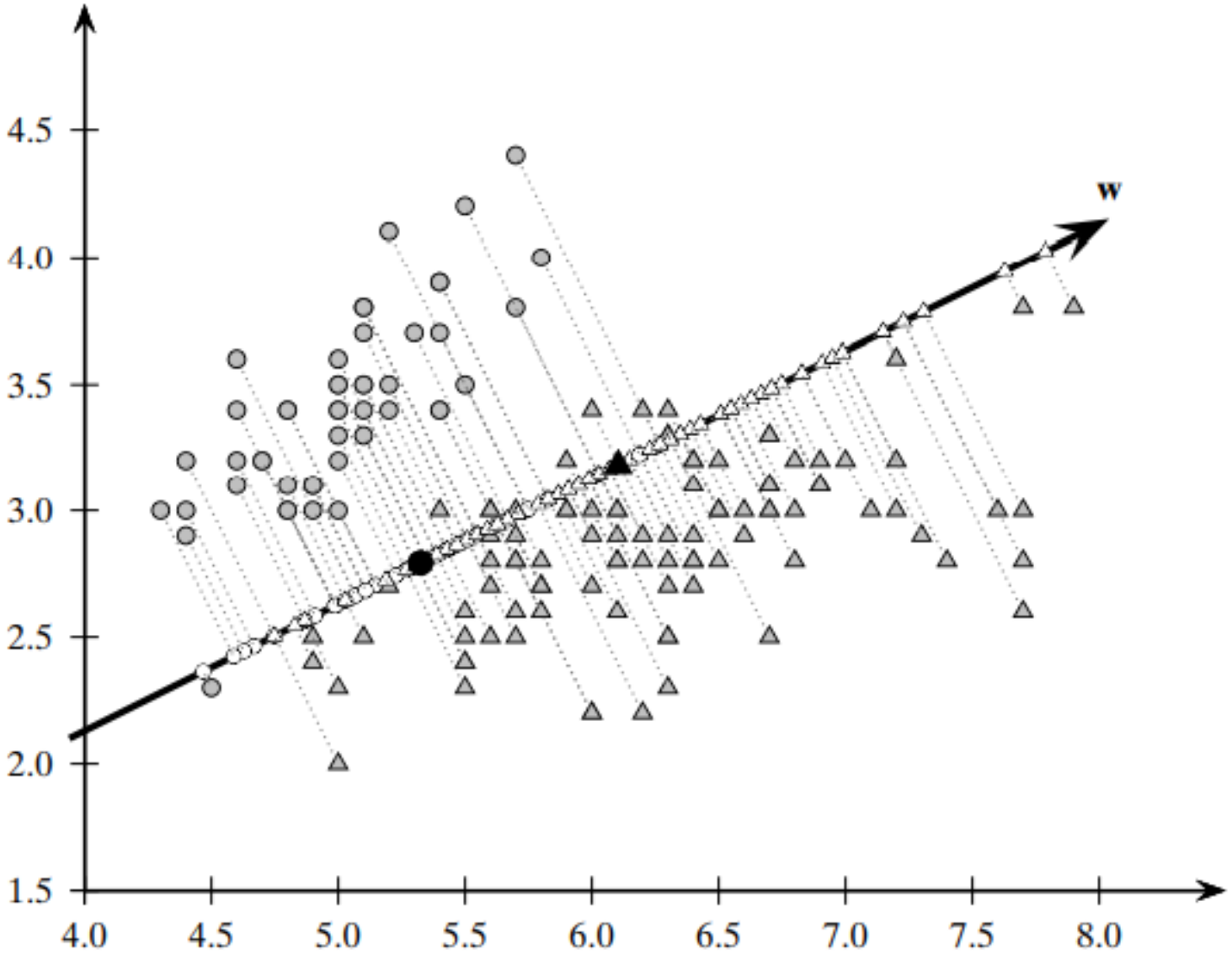
- Каждая координата точки a_i имеет связанную с ней метку исходного класса y_i, и таким образом мы можем вычислить для каждого из двух классов среднее значение проецируемых точек следующим образом:

$$
\begin{aligned}
m_1 &= \frac{1}{n_1} \sum_{\mathbf{x}_i \in \mathbf{D}_1} a_i \\
&= \frac{1}{n_1} \sum_{\mathbf{x}_i \in \mathbf{D}_1} \mathbf{w}^T \mathbf{x}_i \\
&= \mathbf{w}^T \left(\frac{1}{n_1} \sum_{\mathbf{x}_i \in \mathbf{D}_1} \mathbf{x}_i \right) \\
&= \mathbf{w}^T \boldsymbol{\mu}_1
\end{aligned}
$$

- где $\mu 1$ среднее значение всех точек в $\mathbf{D}_1$. Аналогично, мы можем получить

$$m_2 = \mathbf{w}^T \boldsymbol{\mu}_2$$

Проекция на w



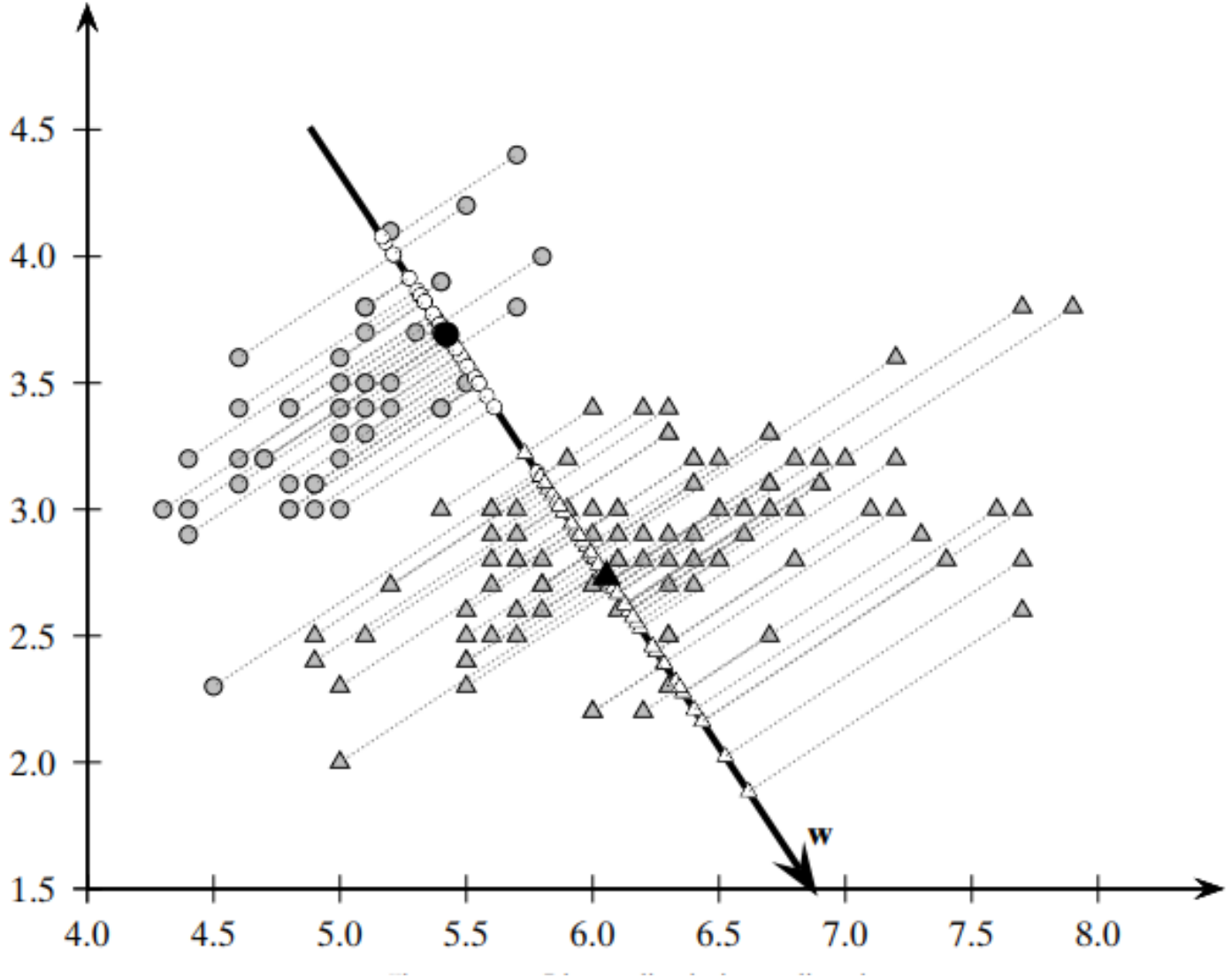
- Каждая координата точки a_i имеет связанную с ней метку исходного класса y_i, и таким образом мы можем вычислить для каждого из двух классов среднее значение проецируемых точек следующим образом:

$$
\begin{aligned}
m_1 &= \frac{1}{n_1} \sum_{\mathbf{x}_i \in \mathbf{D}_1} a_i \\
&= \frac{1}{n_1} \sum_{\mathbf{x}_i \in \mathbf{D}_1} \mathbf{w}^T \mathbf{x}_i \\
&= \mathbf{w}^T \left(\frac{1}{n_1} \sum_{\mathbf{x}_i \in \mathbf{D}_1} \mathbf{x}_i \right) \\
&= \mathbf{w}^T \boldsymbol{\mu}_1
\end{aligned}
$$

- где $\mu 1$ среднее значение всех точек в $\mathbf{D}_1$. Аналогично, мы можем получить

$$m_2 = \mathbf{w}^T \boldsymbol{\mu}_2$$

Линейное дискриминантное направление **w**



- Чтобы максимизировать разделение между классами, представляется разумным максимизировать разницу между прогнозируемыми средними, $|m_1 - m_2|$. Однако этого недостаточно. Для хорошего разделения дисперсия прогнозируемых точек для каждого класса также не должна быть слишком большой. Большая дисперсия привела бы к возможному совпадению точек двух классов из-за большого разброса точек, и поэтому мы можем не иметь хорошего разделения. LDA максимизирует разделение, гарантируя, что разброс s_i^2 для проецируемых точек внутри каждого класса является маленьким, где разброс определяется как $$s_i^2 = \sum_{\mathbf{x}_j \in \mathbf{D}_i} (a_j - m_i)^2$$

- Разброс – это общее квадратичное отклонение от среднего арифметического, в отличие от дисперсии, которая является средним отклонением от среднего арифметического. Другими словами $s_i^2 = n_i \sigma_i^2$ где $n_i = |\mathbf{D}_i|$ - размер, а σ_i^2 дисперсия для класса c_i.
-

- Мы можем включить два критерия линейного дискриминантного анализа, а именно максимизацию расстояния между прогнозируемыми средними и минимизацию суммы прогнозируемого разброса, в единый критерий максимизации, называемый задачей линейного дискриминантного анализа Фишера:

$$\max_{\mathbf{w}} \; J(\mathbf{w}) = \frac{(m_1 - m_2)^2}{s_1^2 + s_2^2} \qquad (20.1)$$

- Цель линейного дискриминантного анализа состоит в том, чтобы найти вектор **w,** который максимизирует $J(\mathbf{w})$, то есть направление, которое максимизирует разделение между двумя средними m_1 и m_2, и минимизирует общий разброс $s_1^2 + s_2^2$ двух классов. Вектор **w** также называется оптимальным линейным дискриминантом. Задача оптимизации [Ур. (20.1)] находится в проекционном пространстве. Чтобы решить ее, мы должны переписать ее в терминах входных данных

- Заметим, что мы можем переписать $(m_1 - m_2)^2$ следующим образом:

$$\begin{aligned}(m_1 - m_2)^2 &= \left(\mathbf{w}^T(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)\right)^2 \\ &= \mathbf{w}^T\left((\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T\right)\mathbf{w} \\ &= \mathbf{w}^T\mathbf{B}\mathbf{w} \end{aligned} \tag{20.2}$$

где **B** = $(\mu 1 - \mu 2)(\mu 1 - \mu 2)^T$ матрица ранга d ×d, называемая матрицей рассеяния между классами.

- Что касается прогнозируемого разброса для класса c_1, то мы можем вычислить его следующим образом:

$$\begin{aligned} s_1^2 &= \sum_{\mathbf{x}_i \in \mathbf{D}_1} (a_i - m_1)^2 \\ &= \sum_{\mathbf{x}_i \in \mathbf{D}_1} (\mathbf{w}^T\mathbf{x}_i - \mathbf{w}^T\boldsymbol{\mu}_1)^2 \\ &= \sum_{\mathbf{x}_i \in \mathbf{D}_1} \left(\mathbf{w}^T(\mathbf{x}_i - \boldsymbol{\mu}_1)\right)^2 \\ &= \mathbf{w}^T\left(\sum_{\mathbf{x}_i \in \mathbf{D}_1} (\mathbf{x}_i - \boldsymbol{\mu}_1)(\mathbf{x}_i - \boldsymbol{\mu}_1)^T\right)\mathbf{w} \\ &= \mathbf{w}^T\mathbf{S}_1\mathbf{w} \end{aligned} \tag{20.3}$$

где $\mathbf{S}_1$ матрица рассеяния для $\mathbf{D}_1$.

- Аналогично, мы можем получить

$$s_2^2 = \mathbf{w}^T\mathbf{S}_2\mathbf{w} \tag{20.4}$$

- Обратите внимание еще раз, что матрица рассеяния, по существу, такая же, как и ковариационная матрица, но вместо того, чтобы записывать среднее отклонение от среднего арифметического, она записывает общее отклонение, то есть

$$\mathbf{S}_i = n_i \boldsymbol{\Sigma}_i \tag{20.5}$$

- Совмещая уравнения (20.3) и (20.4), знаменатель в уравнении (20.1) может быть переписан как:

$$s_1^2 + s_2^2 = \mathbf{w}^T \mathbf{S}_1 \mathbf{w} + \mathbf{w}^T \mathbf{S}_2 \mathbf{w} = \mathbf{w}^T (\mathbf{S}_1 + \mathbf{S}_2) \mathbf{w} = \mathbf{w}^T \mathbf{S} \mathbf{w} \tag{20.6}$$

- где $\mathbf{S} = \mathbf{S}_1 + \mathbf{S}_2$ обозначает матрицу рассеяния внутри класса для объединённых данных. Так как S_1 и $\mathbf{S}_2$ симметричны положительно определённым d × d матрицам, **S** имеет те же свойства.
- Используя уравнения (20.2) и (20.6), запишем целевую функцию линейного дискриминантного анализа [ур. (20.1)] следующим образом

$$\max_{\mathbf{w}} \; J(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{B} \mathbf{w}}{\mathbf{w}^T \mathbf{S} \mathbf{w}} \tag{20.7}$$

- Чтобы решить задачу для наилучшего направления **w**, мы дифференцируем целевую функцию относительно **w** и устанавливаем результат равным нулю. Нам явно не приходится иметь дело с ограничением, что $w^T w = 1$, потому что в ур. (20.7) члены, связанные с величиной **w**, отменяются в числителе и знаменателе. Напомним, что если f (x) и g(x) – две функции, то мы имеем

$$\frac{d}{dx}\left(\frac{f(x)}{g(x)}\right) = \frac{f'(x)g(x) - g'(x)f(x)}{g(x)^2}$$

- где f '(x) обозначает производную от f (x). Принимая производную от ур. (20.7) относительно вектора w и установка результата на нулевой вектор дает

$$\frac{d}{d\mathbf{w}}J(\mathbf{w}) = \frac{2\mathbf{B}\mathbf{w}(\mathbf{w}^T\mathbf{S}\mathbf{w}) - 2\mathbf{S}\mathbf{w}(\mathbf{w}^T\mathbf{B}\mathbf{w})}{(\mathbf{w}^T\mathbf{S}\mathbf{w})^2} = \mathbf{0}$$

$$\mathbf{B}\,\mathbf{w}(\mathbf{w}^T\mathbf{S}\mathbf{w}) = \mathbf{S}\,\mathbf{w}(\mathbf{w}^T\mathbf{B}\mathbf{w})$$

$$\mathbf{B}\,\mathbf{w} = \mathbf{S}\,\mathbf{w}\left(\frac{\mathbf{w}^T\mathbf{B}\mathbf{w}}{\mathbf{w}^T\mathbf{S}\mathbf{w}}\right)$$

$$\mathbf{B}\,\mathbf{w} = J(\mathbf{w})\mathbf{S}\mathbf{w}$$

$$\mathbf{B}\mathbf{w} = \lambda\mathbf{S}\mathbf{w}$$

- где λ = J(**w**). Ур. (20.8) представляет собой обобщённую задачу на собственные значения где λ обобщенного собственного значения **B** и **S**; собственному значению λ удовлетворяет уравнение det(**B** – λ**S**) = 0. Потому что цель к максимизации задачи [Ур. (20.7)], J(**w**) = λ должна быть выбрана по величине обобщенного собственного значения и w для соответствующего собственного вектора.
- Если S невырождено, то есть если S^{-1} существует, то ур. (20.8) приводит к стандартному уравнению собственных значений – собственных векторов, как

$$\mathbf{B}\mathbf{w} = \lambda\mathbf{S}\mathbf{w}$$

$$\mathbf{S}^{-1}\mathbf{B}\mathbf{w} = \lambda\mathbf{S}^{-1}\mathbf{S}\mathbf{w}$$

$$(\mathbf{S}^{-1}\mathbf{B})\mathbf{w} = \lambda\mathbf{w} \qquad (20.9)$$

Алгоритм

- Линейный дискриминант (D = $\{(x_i, y_i)\}_{i=1}^{n}$):

1. $\mathbf{D}_i \leftarrow \{\mathbf{x}_j \mid y_j = c_i, j = 1, \ldots, n\}$, i = 1,2 // отдельный класс подмножеств
2. $\boldsymbol{\mu}_i \leftarrow \text{mean}(\mathbf{D}_i), i = 1,2$ // класс средних арифметических
3. $\mathbf{B} \leftarrow (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T$ // матрица рассеяния между классами
4. $\mathbf{Z}_i \leftarrow \mathbf{D}_i - \mathbf{1}_{n_i}\boldsymbol{\mu}_i^T, i = 1,2$ // центральный класс матриц
5. $\mathbf{S}_i \leftarrow \mathbf{Z}_i^T\mathbf{Z}_i, i = 1,2$ // класс матриц рассеяния
6. $\mathbf{S} \leftarrow \mathbf{S}_1 + \mathbf{S}_2$ // класс матрицы рассеяния внутри класса
7. $\lambda_1, \mathbf{w} \leftarrow \text{eigen}(\mathbf{S}^{-1}\mathbf{B})$// вычисление главного собственного вектора

Пример

$$\boldsymbol{\mu}_1 = \begin{pmatrix} 5.01 \\ 3.42 \end{pmatrix} \boldsymbol{\mu}_2 = \begin{pmatrix} 6.26 \\ 2.87 \end{pmatrix} \boldsymbol{\mu}_1 - \boldsymbol{\mu}_2 = \begin{pmatrix} -1.256 \\ 0.546 \end{pmatrix}$$

$$\mathbf{B} = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T = \begin{pmatrix} -1.256 \\ 0.546 \end{pmatrix} (-1.256\ 0.546) = \begin{pmatrix} 1.587 & -0.693 \\ -0.693 & 0.303 \end{pmatrix}$$

$$\mathbf{S}_1 = \begin{pmatrix} 6.09 & 4.91 \\ 4.91 & 7.11 \end{pmatrix} \mathbf{S}_2 = \begin{pmatrix} 43.5 & 12.09 \\ 12.09 & 10.96 \end{pmatrix} \mathbf{S} = \mathbf{S}_1 + \mathbf{S}_2 = \begin{pmatrix} 49.58 & 17.01 \\ 17.01 & 18.08 \end{pmatrix}$$

$$\mathbf{S}^{-1} = \begin{pmatrix} 0.0298 & -0.028 \\ -0.028 & 0.0817 \end{pmatrix}$$

$$\mathbf{S}^{-1}\mathbf{B} = \begin{pmatrix} 0.0298 & -0.028 \\ -0.028 & 0.0817 \end{pmatrix} \begin{pmatrix} 1.587 & -0.693 \\ -0.693 & 0.303 \end{pmatrix} = \begin{pmatrix} 0.066 & -0.029 \\ -0.100 & 0.044 \end{pmatrix}$$

$$J(\mathbf{w}) = \lambda_1 = 0.11$$

$$\mathbf{w} = \begin{pmatrix} 0.551 \\ -0.834 \end{pmatrix}$$

Оптимальный линейный дискриминант: линейная комбинация характерных точек

- Среднее арифметическое значение для класса c_i в пространстве признаков задается как

$$\boldsymbol{\mu}_i^{\phi} = \frac{1}{n_i} \sum_{\mathbf{x}_j \in \mathbf{D}_i} \phi(\mathbf{x}_j) \quad (20.12)$$

- а ковариационная матрица для класса c_i в пространстве признаков

$$\boldsymbol{\Sigma}_i^{\phi} = \frac{1}{n_i} \sum_{\mathbf{x}_j \in \mathbf{D}_i} \left(\phi(\mathbf{x}_j) - \boldsymbol{\mu}_i^{\phi}\right)\left(\phi(\mathbf{x}_j) - \boldsymbol{\mu}_i^{\phi}\right)^T$$

- Используя вывод, аналогичный уравнению (20.2) мы получаем выражение для матрицы рассеяния между классами в пространстве признаков

$$\mathbf{B}_{\phi} = \left(\boldsymbol{\mu}_1^{\phi} - \boldsymbol{\mu}_2^{\phi}\right)\left(\boldsymbol{\mu}_1^{\phi} - \boldsymbol{\mu}_2^{\phi}\right)^T = \mathbf{d}_{\phi}\mathbf{d}_{\phi}^T \quad (20.13)$$

- где $\mathbf{d}_{\phi} = \boldsymbol{\mu}_1^{\phi} - \boldsymbol{\mu}_2^{\phi}$ разность между средними векторами двух классов. Аналогично, используя уравнения (20.5) и (20.6) матрица рассеяния внутри класса в пространстве признаков задается следующим образом

$$\mathbf{S}_{\phi} = n_1\boldsymbol{\Sigma}_1^{\phi} + n_2\boldsymbol{\Sigma}_2^{\phi}$$

- S_ϕ -симметричная положительная полуопределенная матрица, размером d × d, где d-размерность пространства признаков. Из уравнения (20.9) мы заключаем, что наилучшим линейным дискриминантным вектором **w** в пространстве признаков является доминирующий собственный вектор, удовлетворяющий выражению

$$\left(\mathbf{S}_\phi^{-1}\mathbf{B}_\phi\right)\mathbf{w} = \lambda\mathbf{w} \text{ (20.14)}$$

- Где мы предполагаем, что S_ϕ не вырожден. Пусть δ_i обозначает i-е собственное значение, а u_i i-ый собственный вектор S_ϕ, при i = 1,...,d. Собственное разложение S_ϕ даёт S_ϕ = U$\mathbf{\Delta}U^T$, с обратным значением S_ϕ ,заданным как $\mathbf{S}_\phi^{-1} = \mathbf{U}\mathbf{\Delta}^{-1}\mathbf{U}^T$. Здесь U матрица, столбцы которой являются собственными векторами S_ϕ а $\mathbf{\Delta}$ диагональная матрица собственных значений S_ϕ. Таким образом, $\mathbf{S}_\phi^{-1}$ может быть выражена как спектральная сумма

$$\mathbf{S}_\phi^{-1} = \sum_{r-1}^{d} \frac{1}{\delta_r}\mathbf{u}_r\mathbf{u}_r^T$$

- Включая уравнения (20.13) и (20.15) в уравнении (20.14) получаем:

$$\lambda \mathbf{w} = \left(\sum_{r=1}^{d} \frac{1}{\delta_r} \mathbf{u}_r \mathbf{u}_r^T \right) \mathbf{d}_\phi \mathbf{d}_\phi^T \mathbf{w} = \sum_{r=1}^{d} \frac{1}{\delta_r} \left(\mathbf{u}_r \left(\mathbf{u}_r^T \mathbf{d}_\phi \right) \left(\mathbf{d}_\phi^T \mathbf{w} \right) \right) = \sum_{r=1}^{d} b_r \mathbf{u}_r$$

- где $b_r = \frac{1}{\delta_r} \left(\mathbf{u}_r^T \mathbf{d}_\phi \right) \left(\mathbf{d}_\phi^T \mathbf{w} \right)$ скалярное значение. Используя деривацию подобную той, что в уравнении (7.32), r-й собственный вектор S_ϕ может быть выражен как линейная комбинация характерных точек

$$\mathbf{u}_r = \sum_{j=1}^{n} c_{rj} \phi \left(\mathbf{x}_j \right)$$

- где c_{rj} скалярный коэффициент.

- Таким образом мы можем переписать w

$$\mathbf{w} = \frac{1}{\lambda}\sum_{r=1}^{d} b_r \left(\sum_{j=1}^{n} c_{rj}\phi(\mathbf{x}_j) \right)$$

$$= \sum_{j=1}^{n} \phi(\mathbf{x}_j) \left(\sum_{r=1}^{d} \frac{b_r c_{rj}}{\lambda} \right)$$

$$= \sum_{j=1}^{n} a_j \phi(\mathbf{x}_j)$$

- где $a_j = \sum_{r=1}^{d} b_r c_{rj}/\lambda$ скалярное значение для характерной точки $\phi(\mathbf{x}_j)$. Поэтому вектор направления w может быть выражен как линейная комбинация точек в пространстве признаков.

Задача линейного дискриминантного анализа через матрицу ядра

- Теперь мы перепишем задачу ядра линейного дискриминантного анализа [Ур. (20.11)] в терминах матрицы ядра. Проецируя среднее значение для класса c_i приведенное в уравнении (20.12) на направление линейного дискриминанта **w**, мы имеем

$$m_i = \mathbf{w}^T \boldsymbol{\mu}_i^{\phi} = \left(\sum_{j=1}^{n} a_j \phi(\mathbf{x}_j) \right)^T \left(\frac{1}{n_i} \sum_{\mathbf{x}_k \in \mathbf{D}_i} \phi(\mathbf{x}_k) \right)$$

$$= \frac{1}{n_i} \sum_{j=1}^{n} \sum_{\mathbf{x}_k \in \mathbf{D}_i} a_j \phi(\mathbf{x}_j)^T \phi(\mathbf{x}_k)$$

$$= \frac{1}{n_i} \sum_{j=1}^{n} \sum_{\mathbf{x}_k \in \mathbf{D}_i} a_j K(\mathbf{x}_j, \mathbf{x}_k)$$

$$= \mathbf{a}^T \mathbf{m}_i \qquad (20.16)$$

- Где $\mathbf{a} = (a_1, a_2, \ldots, a_n)^T$ весовой вектор, и

$$\mathbf{m}_i = \frac{1}{n_i}\begin{pmatrix} \sum_{\mathbf{x}_k \in \mathbf{D}_i} K(\mathbf{x}_1, \mathbf{x}_k) \\ \sum_{\mathbf{x}_k \in \mathbf{D}_i} K(\mathbf{x}_2, \mathbf{x}_k) \\ \vdots \\ \sum_{\mathbf{x}_k \in \mathbf{D}_i} K(\mathbf{x}_n, \mathbf{x}_k) \end{pmatrix} = \frac{1}{n_i}\mathbf{K}^{c_i}\mathbf{1}_{n_i} \qquad (20.17)$$

- Где $\mathbf{K}^{c_i}$ является n × n_i подмножеством ядра матрицы, ограниченного столбцами, принадлежащими точкам только в $\mathbf{D}_i$, и $\mathbf{1}_{n_i}$ является n_i-мерным вектором вес записи которого равны единице. Таким образом, вектор n-длины m_i сохраняет для каждой точки D свое среднее значение ядра по отношению к указанным точкам $\mathbf{D}_i$.

- Мы можем переписать разделение между проецируемыми средними в пространстве признаков следующим образом:

$$
\begin{aligned}
(m_1 - m_2)^2 &= \left(\mathbf{w}^T \boldsymbol{\mu}_1^{\phi} - \mathbf{w}^T \boldsymbol{\mu}_2^{\phi}\right)^2 \\
&= \left(\mathbf{a}^T \mathbf{m}_1 - \mathbf{a}^T \mathbf{m}_2\right)^2 \\
&= \mathbf{a}^T (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T \mathbf{a} \\
&= \mathbf{a}^T \mathbf{M} \mathbf{a} \qquad (20.18)
\end{aligned}
$$

- где $\mathbf{M} = (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T$ матрица рассеяния между классами.

- Мы также можем вычислить прогнозируемый разброс для каждого класса, s_1^2 и s_2^2 , исключительно в терминах функции ядра, как

$$s_1^2 = \sum_{\mathbf{x}_i \in \mathbf{D}_1} \left\| \mathbf{w}^T \phi(\mathbf{x}_i) - \mathbf{w}^T \boldsymbol{\mu}_1^{\phi} \right\|^2$$

$$= \sum_{\mathbf{x}_i \in \mathbf{D}_1} \left\| \mathbf{w}^T \phi(\mathbf{x}_i) \right\|^2 - 2 \sum_{\mathbf{x}_i \in \mathbf{D}_1} \mathbf{w}^T \phi(\mathbf{x}_i) \cdot \mathbf{w}^T \boldsymbol{\mu}_1^{\phi} + \sum_{\mathbf{x}_i \in \mathbf{D}_1} \left\| \mathbf{w}^T \boldsymbol{\mu}_1^{\phi} \right\|^2$$

$$= \left(\sum_{\mathbf{x}_i \in \mathbf{D}_1} \left\| \sum_{j=1}^{n} a_j \phi(\mathbf{x}_j)^T \phi(\mathbf{x}_i) \right\|^2 \right) - 2 \cdot n_1 \cdot \left\| \mathbf{w}^T \boldsymbol{\mu}_1^{\phi} \right\|^2 + n_1 \cdot \left\| \mathbf{w}^T \boldsymbol{\mu}_1^{\phi} \right\|^2$$

-

$$= \left(\sum_{\mathbf{x}_i \in \mathbf{D}_1} \mathbf{a}^T \mathbf{K}_i \mathbf{K}_i^T \mathbf{a} \right) - n_1 \cdot \mathbf{a}^T \mathbf{m}_1 \mathbf{m}_1^T \mathbf{a}$$

используя ур.(20.16)

$$= \mathbf{a}^T \left(\left(\sum_{\mathbf{x}_i \in \mathbf{D}_1} \mathbf{K}_i \mathbf{K}_i^T \right) - n_1 \mathbf{m}_1 \mathbf{m}_1^T \right) \mathbf{a}$$

$$= \mathbf{a}^T \mathbf{N}_1 \mathbf{a}$$

- где $\mathbf{K}_i$ i-ый столбец ядра матрицы, а $\mathbf{N}_1$ матрица рассеяния класса для c_1. Пусть K(x_i,x_j) = K_{ij} .

- Мы можем выразить $\mathbf{N}_1$ более компактно в матричных обозначениях следующим образом:

$$\mathbf{N}_1 = \left(\sum_{\mathbf{x}_i \in \mathbf{D}_1} \mathbf{K}_i \mathbf{K}_i^T \right) - n_1 \mathbf{m}_1 \mathbf{m}_1^T$$

$$= (\mathbf{K}^{c_1}) \left(\mathbf{I}_{n_1} - \frac{1}{n_1} \mathbf{1}_{n_1 \times n_1} \right) (\mathbf{K}^{c_1})^T \qquad (20.19)$$

- где $\mathbf{I}_{n_1}$ это тождественная матрица $n_1 \times n_1$, а $\mathbf{1}_{n_1 \times n_1}$ матрица $n_1 \times n_1$, все записи которой равны 1.
- Аналогично получаем $s_2^2 = \mathbf{a}^T \mathbf{N}_2 \mathbf{a}$, где

$$\mathbf{N}_2 = (\mathbf{K}^{c_2}) \left(\mathbf{I}_{n_2} - \frac{1}{n_2} \mathbf{1}_{n_2 \times n_2} \right) (\mathbf{K}^{c_2})^T \ (20.20)$$

- где N - матрица рассеяния n × n внутри класса

- При замене уравнений (20.18) и (20.20) в уравнении (20.11) получим условие максимизации ядра линейного дискриминантного анализа

$$\max_{\mathbf{w}} J(\mathbf{w}) = \max_{\mathbf{a}} J(\mathbf{a}) = \frac{\mathbf{a}^T \mathbf{M} \mathbf{a}}{\mathbf{a}^T \mathbf{N} \mathbf{a}}$$

- Обратите внимание, что все термины в приведенном выше выражении включают только функции ядра. Весовой вектор **a** - это собственный вектор, соответствующий наибольшему собственному значению обобщенной задачи на собственные значения:

$$\mathbf{M}\mathbf{a} = \lambda_1 \mathbf{N}\mathbf{a} \qquad (20.21)$$

- Если N невырожденный, то a-доминирующий собственный вектор, соответствующий наибольшему собственному значению для системы

$$\left(\mathbf{N}^{-1}\mathbf{M}\right)\mathbf{a} = \lambda_1 \mathbf{a}$$

- Как и в случае линейного дискриминантного анализа [ур. (20.10)], когда есть только два класса, нам не нужно решать для собственного вектора, потому что а может быть получено непосредственно:

$$\mathbf{a} = \mathbf{N}^{-1}(\mathbf{m}_1 - \mathbf{m}_2)$$

- Как только **a** получено, мы можем нормализовать **w** как единичный вектор, гарантируя, что $\mathbf{w}_n^T \mathbf{w}_n = 1$, из чего следует, что

$$\sum_{i=1}^{n} \sum_{j=1}^{n} a_i a_j \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) = 1, \text{или } \mathbf{a}^T \mathbf{K} \mathbf{a} = 1$$

- Иными словами, мы можем гарантировать, что **w** является единичным вектором, если масштабируем **a** на $\frac{1}{\sqrt{\mathbf{a}^T \mathbf{K} \mathbf{a}}}$.

- В конечном итоге, мы можем спроецировать любую точку x на дискриминантное направление следующим образом:

$$\mathbf{w}^T\phi(\mathbf{x}) = \sum_{j=1}^{n} a_j \phi(\mathbf{x}_j)^T \phi(\mathbf{x}) = \sum_{j=1}^{n} a_j K(\mathbf{x}_j, \mathbf{x}) \qquad (20.22)$$

Алгоритм

- В алгоритме 20.2 приводится псевдокод для дискриминантного анализа ядра. Суть метода заключается в вычислении ядерной матрицы **K** размера $n \times n$ и ядерных матриц $\mathbf{K}^{ci}$ размером $n \times n_i$, для каждого класса c_i. После вычисления матриц межклассовой и внутриклассовой инвариантности **M** и **N**, вектор весов **a** получается как доминирующий собственный вектор $\mathbf{N}^{-1}\mathbf{M}$. Последний шаг масштабирует **a** так, чтобы **w** был нормализован до единичной длины. Сложность дискриминантного анализа ядра $O(n^3)$, при этом основными шагами являются вычисление **N** и нахождение доминирующего собственного вектора $\mathbf{N}^{-1}\mathbf{M}$, оба шага занимают время $O(n^3)$.

Алгоритм

ALGORITHM 20.2. Kernel Discriminant Analysis

KERNELDISCRIMINANT $(\mathbf{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}, K)$:

1 $\mathbf{K} \leftarrow \left\{K(\mathbf{x}_i, \mathbf{x}_j)\right\}_{i,j=1,\ldots,n}$ // compute $n \times n$ kernel matrix

2 $\mathbf{K}^{c_i} \leftarrow \left\{\mathbf{K}(j,k) \mid y_k = c_i, 1 \le j,k \le n\right\}, i = 1,2$ // class kernel matrix

3 $\mathbf{m}_i \leftarrow \frac{1}{n_i}\mathbf{K}^{c_i}\mathbf{1}_{n_i}, i = 1,2$ // class means

4 $\mathbf{M} \leftarrow (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T$ // between-class scatter matrix

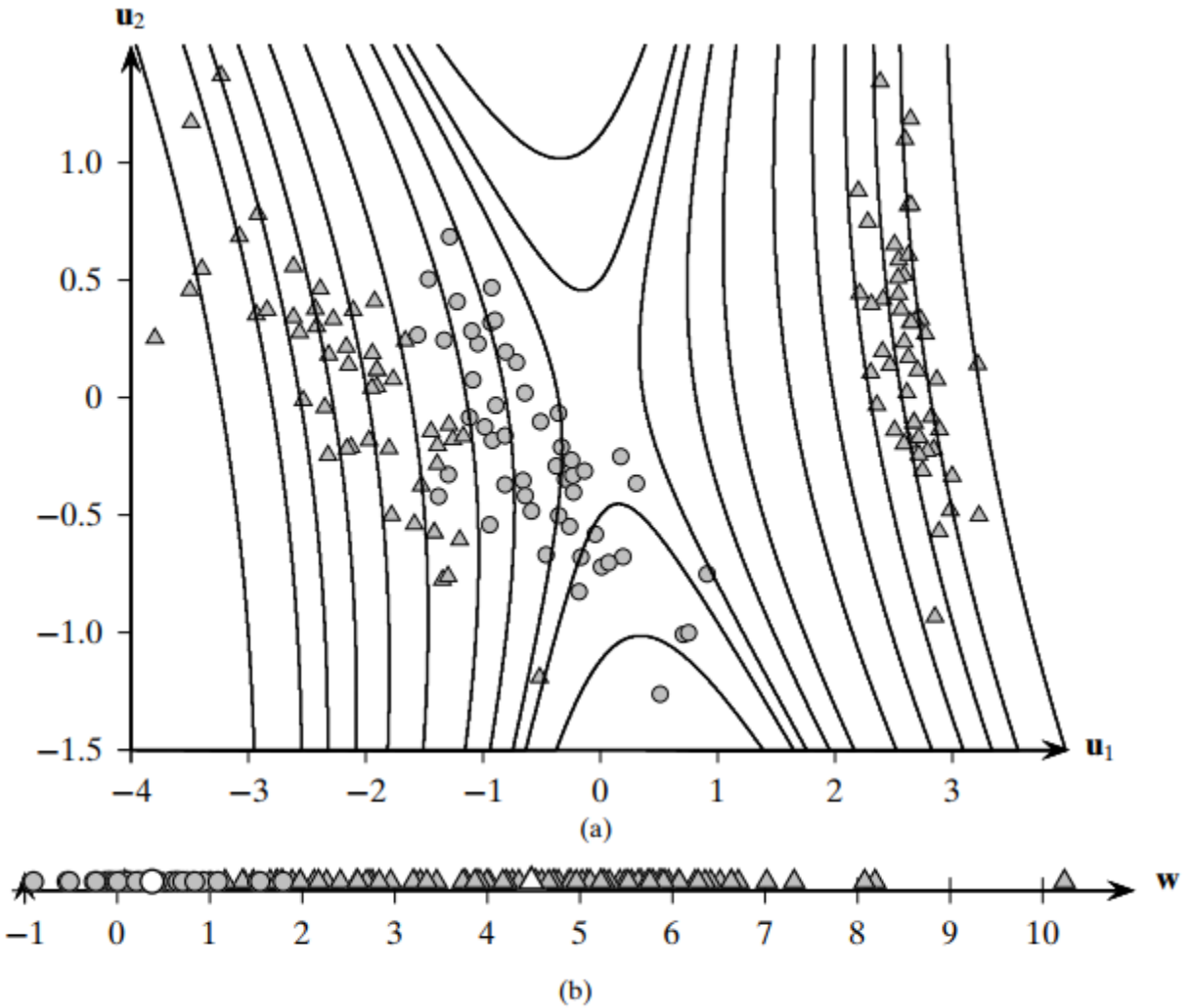
5 $\mathbf{N}_i \leftarrow \mathbf{K}^{c_i}(\mathbf{I}_{n_i} - \frac{1}{n_i}\mathbf{1}_{n_i \times n_i})(\mathbf{K}^{c_i})^T, i = 1,2$ // class scatter matrices

6 $\mathbf{N} \leftarrow \mathbf{N}_1 + \mathbf{N}_2$ // within-class scatter matrix

7 $\lambda_1, \mathbf{a} \leftarrow \text{eigen}(\mathbf{N}^{-1}\mathbf{M})$ // compute weight vector

8 $\mathbf{a} \leftarrow \frac{\mathbf{a}}{\sqrt{\mathbf{a}^T\mathbf{K}\mathbf{a}}}$ // normalize $\mathbf{w}$ to be unit vector

Пример



Пример