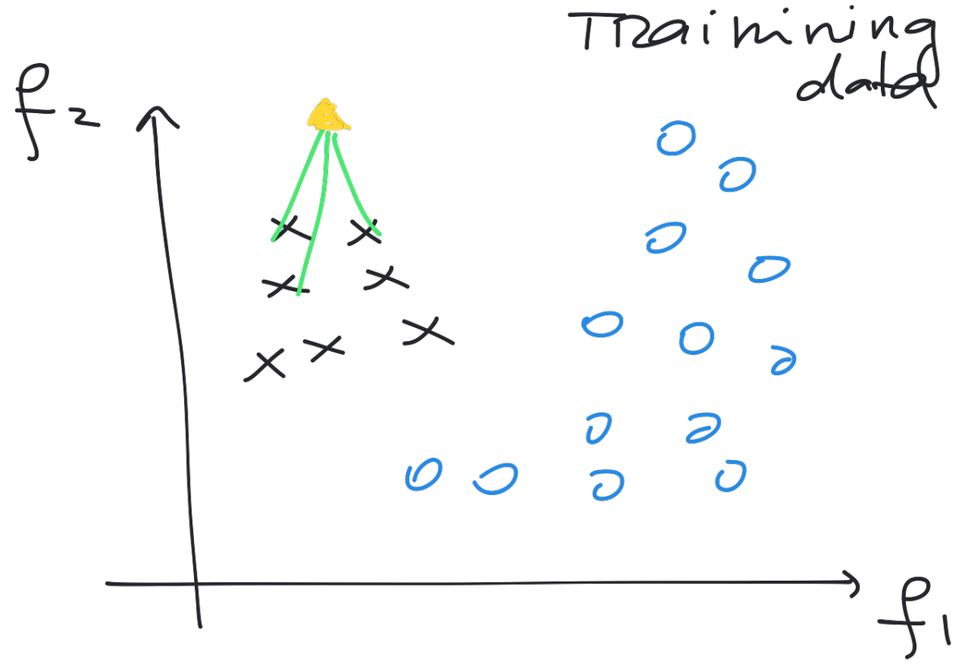
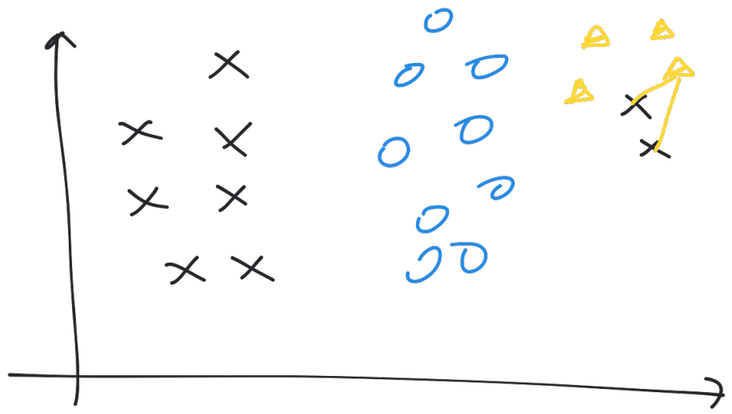


△ : test point

$k = 3$

- neighboring points are based on distances:
- Classification of △ is based:
  ① Majority vote
  when we have ties: ① flip coin — Randomly pick one.
  ② Choose class which neighbor is closest.

"IDEAL" (no ties): $K = \#\text{classes} + 1$.



K = 5

① when we have data with outliers,

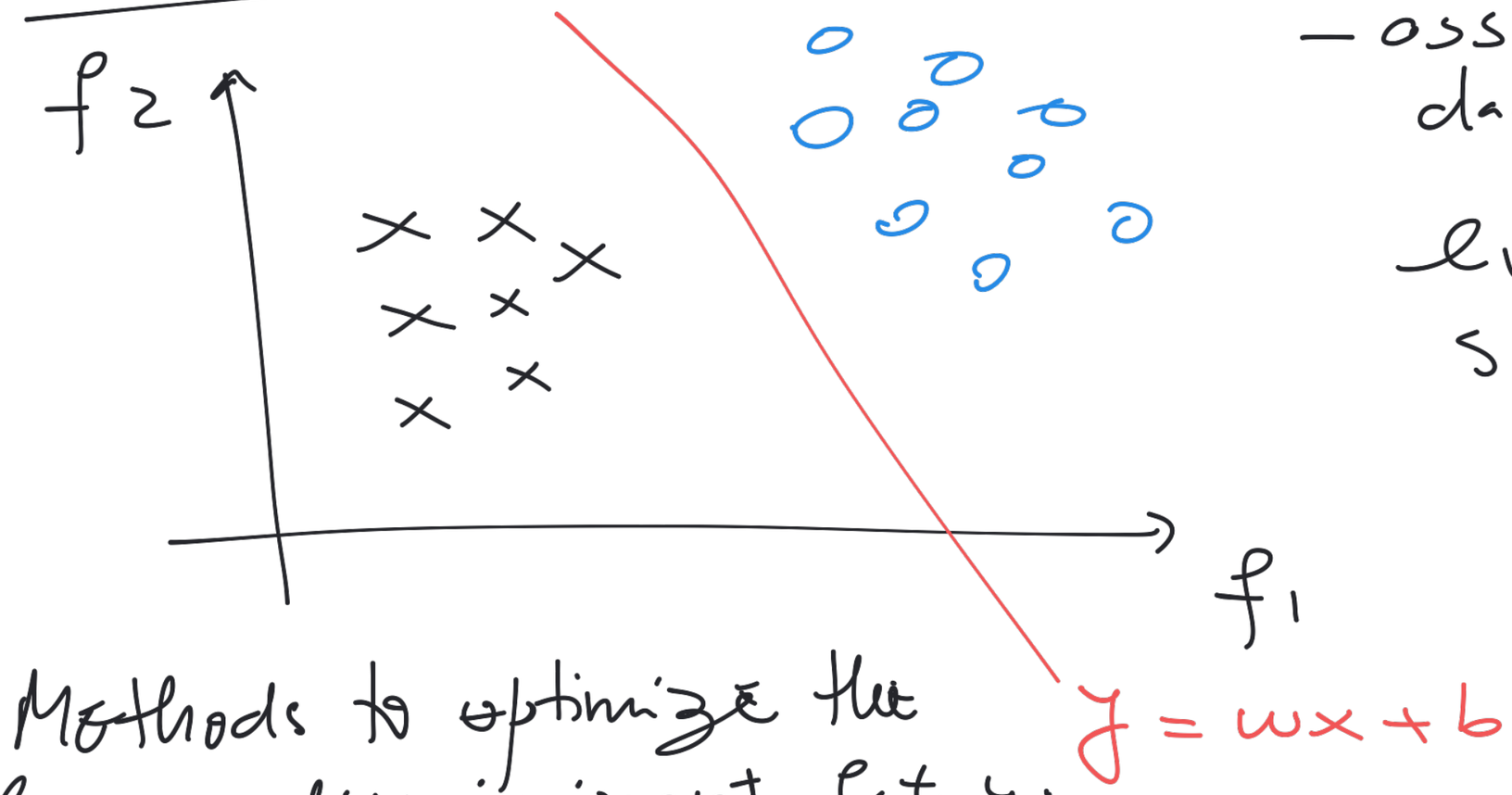
KNN will "fit" outliers with neighboring.

② Maybe we just don't enough data for class X in the farther-right region of ~~feature~~.

Weighted K-NN: each neighbor will be weight parameter $\frac{1}{d(N_i, \triangle)}$

⇒ neighbor points that are closest will have larger weight on the △ classification label.

Discriminative Classifier

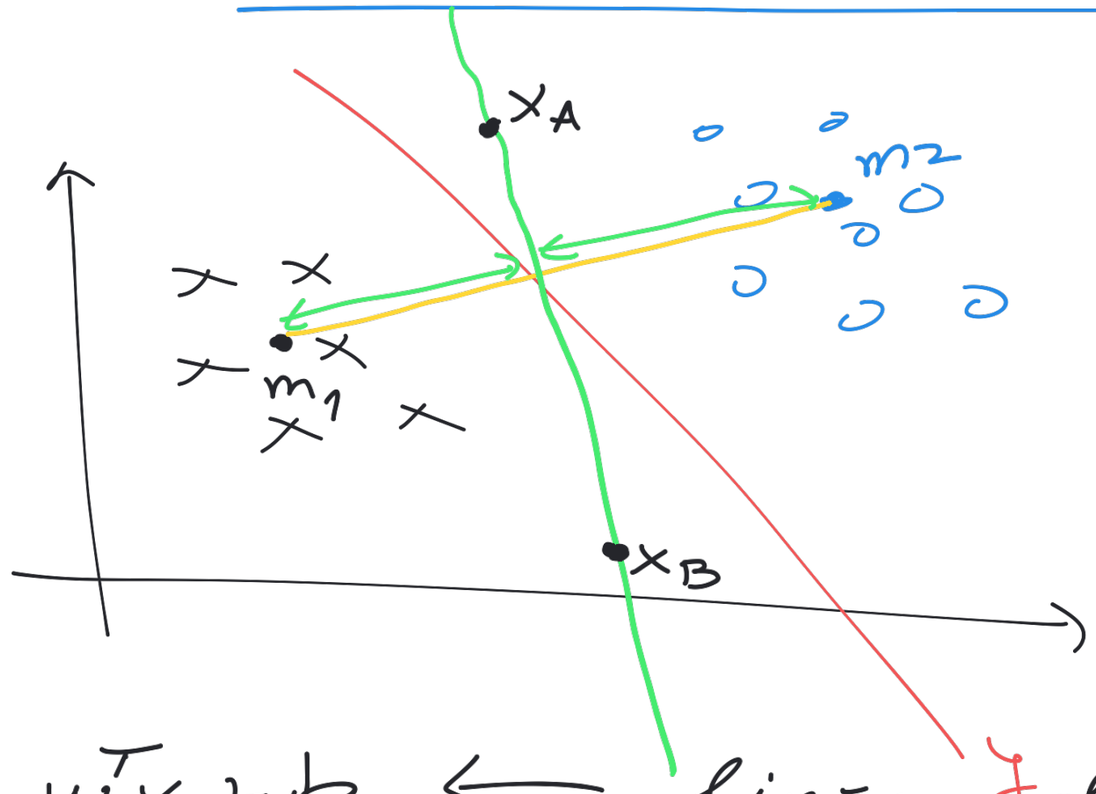


- assuming classes are linearly separable

Methods to optimize the linear discriminant fct y:

1. Least Squares classification ← has some disadvantages (more later)
2. Fisher's Linear Discriminant
3. Perceptron Algorithm.

Fisher's Discriminant Function

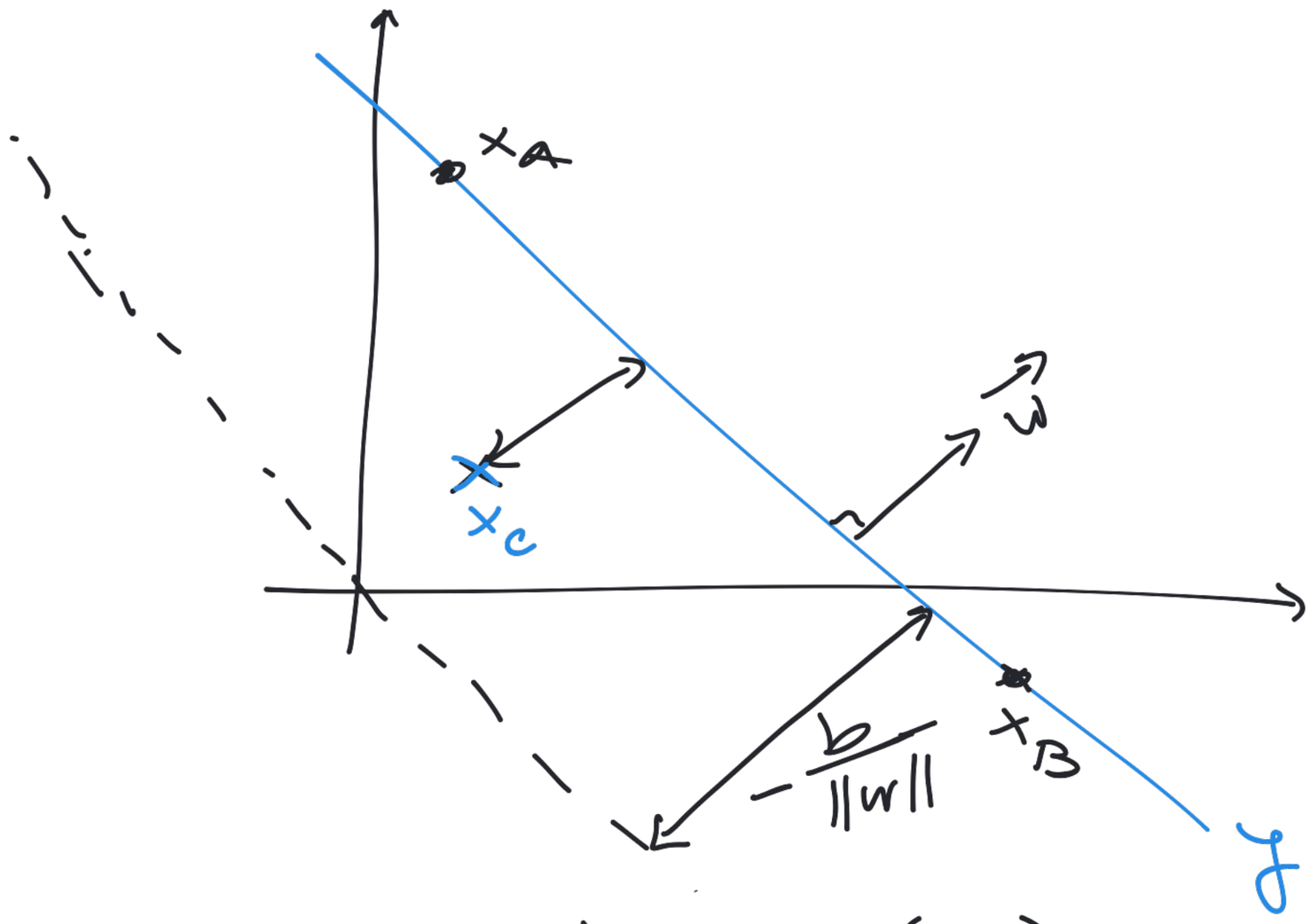


$y = w^T x + b$ ← linear discriminant fct for high-dimensions.

$y(x_A) = 0 = y(x_B)$

$\Rightarrow w^T x_A + b = w^T x_B + b \Leftrightarrow \underline{w^T (x_A - x_B) = 0}$

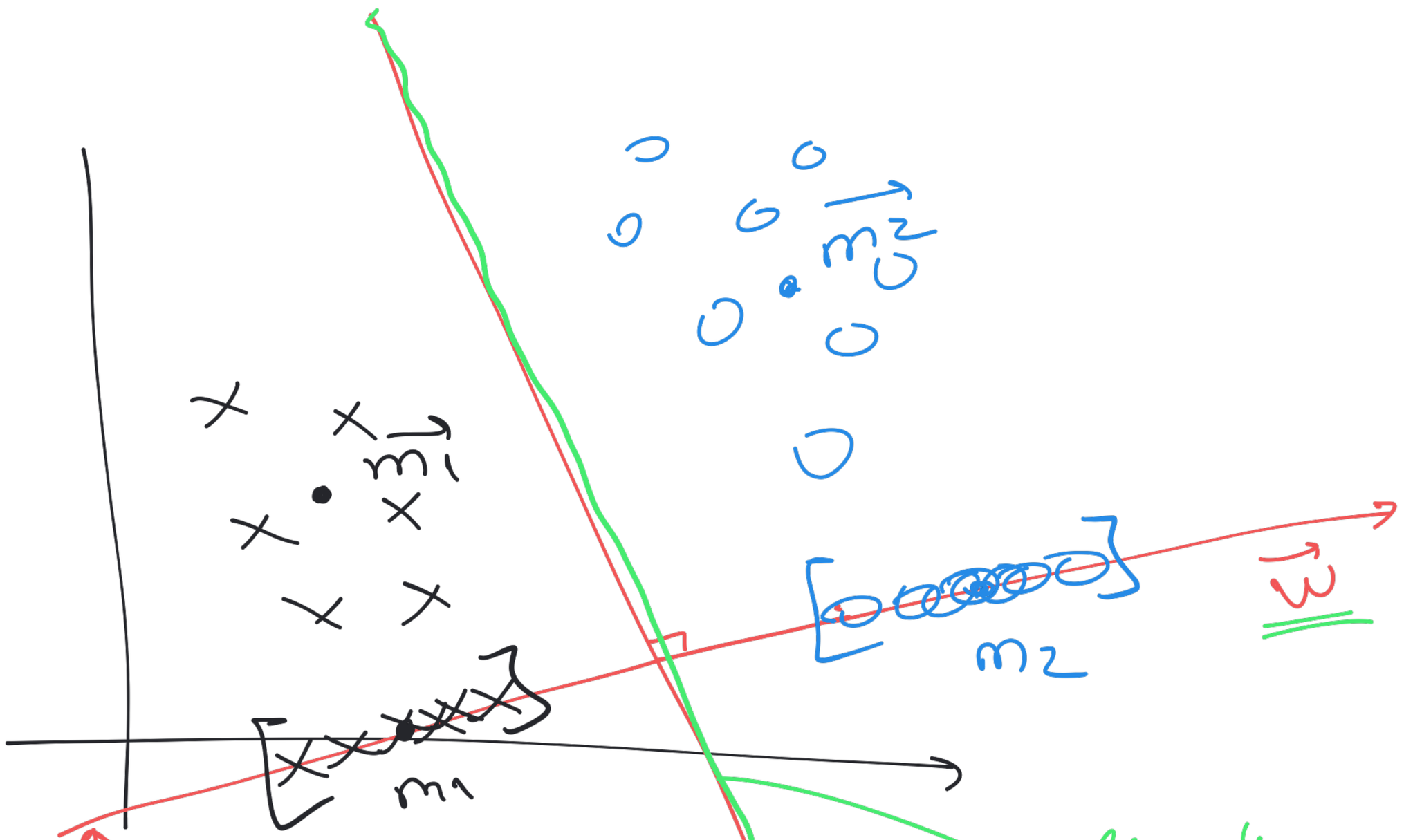
$x_A \neq x_B \Rightarrow w^T \perp x_A - x_B$



$$y = w^T x + b$$

$$\frac{y}{\|w\|} = \frac{w^T x}{\|w\|} + \frac{b}{\|w\|}$$

$$\mathcal{D}(x_c, y) = \frac{y(x_c)}{\|\vec{w}\|}$$



direction of projection that maximizes class separability

direction where we define discriminant function.

- Mean of each class :

$$\vec{m_1} = \frac{1}{N_1} \sum_{n \in C_1} \vec{x_n} \qquad (2\text{-D})$$

$$\vec{m_2} = \frac{1}{N_2} \sum_{n \in C_2} \vec{x_n} \qquad (2\text{-D})$$

$$m_2 - m_1 = \vec{w}^T (\vec{m_2} - \vec{m_1})$$

↳ we want to maximize this differences in means.

within class variance:

$$s_k^2 = \sum_{n \in C_k} (y_n - m_k)^2$$

$$= \sum_{n \in C_k} \left(\vec{w}^T \vec{x}_n - \underline{m_k} \right)$$

$$= \vec{w}^T \sum_{n \in C_k} (\underline{\vec{x}_n} - \underline{\vec{m}_k})(\vec{x}_n - \vec{m}_k)^T \cdot \vec{w}$$

Fisher's Discriminant function:

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

$S_B \equiv$ between class distance

$$= \frac{\vec{w}^T \overbrace{(\vec{m}_2 - \vec{m}_1)(\vec{m}_2 - \vec{m}_1)^T}^{S_B} \vec{w}}{\vec{w}^T \left(\underbrace{\sum_{n \in C_1} (\vec{x}_n - \vec{m}_1)(\vec{x}_n - \vec{m}_1)^T + \sum_{n \in C_2} (\vec{x}_n - \vec{m}_2)(\vec{x}_n - \vec{m}_2)^T}_{= S_w} \right) \vec{w}}$$

$= S_w$ = within class distance

$$= \frac{\vec{w}^T \cdot S_B \cdot \vec{w}}{\vec{w}^T \cdot S_w \cdot \vec{w}}$$

$$\arg\max_w J(w)$$