Inclass 10: Linear Classification and Perceptron Method

[SCS4049] Machine Learning and Data Science

Seongsik Park (s.park@dgu.edu)

Department of Artificial Intelligence, Dongguk University

Classification

MNIST dataset

MNIST dataset

- 70,000 images of 28×28 handwritten digits from 0 to 9
- “Hello, World!” of machine learning

Figure 3-1. A few digits from the MNIST dataset

Classification

The goal in classification

- Take an input vector $\mathbf{x}$
- Assign it to one-of-K discrete class $\mathcal{C}_k$ where $k = 1, 2, \cdots, K$
- Assign each input to one and only one class (disjoint)

The input space divided into decision region, whose boundaries are called **decision boundaries** or **decision surfaces**.

- (D-1)-dimensional hyperplanes within the D-dimensional input space
- Linearly separable dataset that can be separated exactly by linear decision surface

Classification

Binary representation $\mathbf{t}$

- Two-class problem, $t \in \{0, 1\}$ ($t = 1$ represents $\mathcal{C}_1$)
- $K > 2$ classes, $\mathbf{t} \in \{0, 1\}^K$ and $\sum_k t_k = 1$
- t_k is probability of $\mathcal{C}_k$

Approaches

- Direct assign (discriminant function)
- Model $p(\mathcal{C}_k|\mathbf{x})$ with class-conditional distribution $p(\mathbf{x}|\mathcal{C}_k)$ and prior distribution $p(\mathcal{C}_k)$ (generative model)
- Model $p(\mathcal{C}_k|\mathbf{x})$ directly (parametric model)

Classification

Generalized linear model

- $y(\mathbf{x}) = f(\mathbf{w}^T\mathbf{x} + w_0)$
- Activation function $f : \Re \to (0, 1)$
- Even if the function $f(\cdot)$ is nonlinear, the decision surfaces $\mathbf{w}^T\mathbf{x} + w_0 = const$ are linear functions of $\mathbf{x}$

Confusion matrix

		Predicted	
		Negative	Positive
Ground truth	Negative	**T**rue **N**egative	**F**alse **P**ositive (Type I Error)
	Positive	**F**alse **N**egative (Type II Error)	**T**rue **P**ositive

True positive rate (TPR): $\mathrm{TPR} = \dfrac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}$ (recall, sensitivity, hit rate)

Positive predictive value (PPV): $\mathrm{PPV} = \dfrac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}$ (precision)

Accuracy (ACC): $\mathrm{ACC} = \dfrac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FN} + \mathrm{TN} + \mathrm{FP}}$

True negative rate (TNR): $\mathrm{TNR} = \dfrac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}$ (specificity)

False negative rate (FNR): $\mathrm{FNR} = 1 - \mathrm{TPR}$

False positive rate (FPR): $\mathrm{FPR} = 1 - \mathrm{TNR}$

Discriminant functions

Linear discriminant function

Two classes

$$y(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0$$

$$\begin{aligned} \mathbf{x} \text{ is assigned to class } \mathcal{C}_1 \quad & \text{if } y(\mathbf{x}) \geq 0 \\ \text{to class } \mathcal{C}_2 \quad & \text{otherwise} \end{aligned}$$

Decision surface

- $\mathbf{w}$ determines the orientation
- w_0 determines the location
- $-w_0/\|\mathbf{w}\|$ is the normal distance from the origin

Linear discriminant function

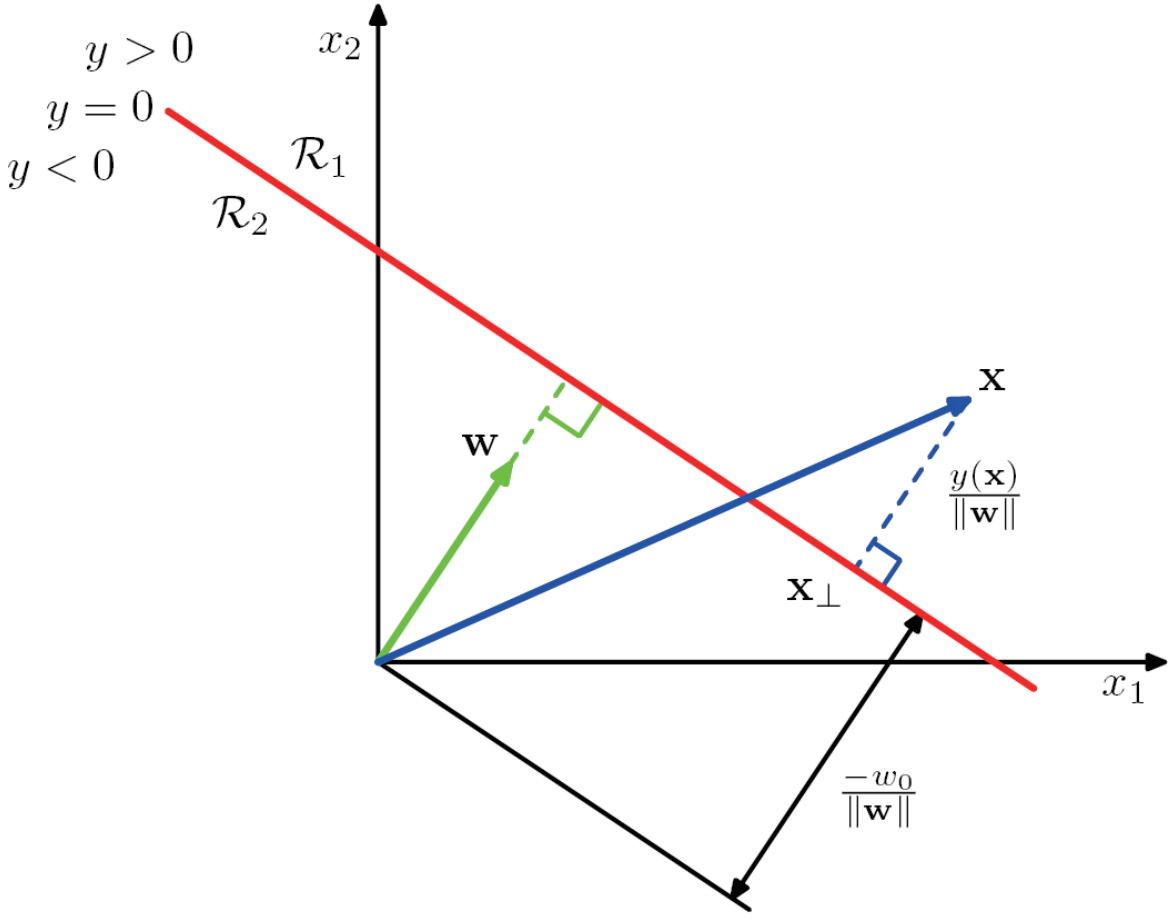


Figure 1: Decision surface of linear discriminant function

Linear discriminant function

Multiple classes

- (K-1) classifier: one-against-the rest
- $K(K\text{-}1)/2$ classifier: one-against-one
- K-class discriminant

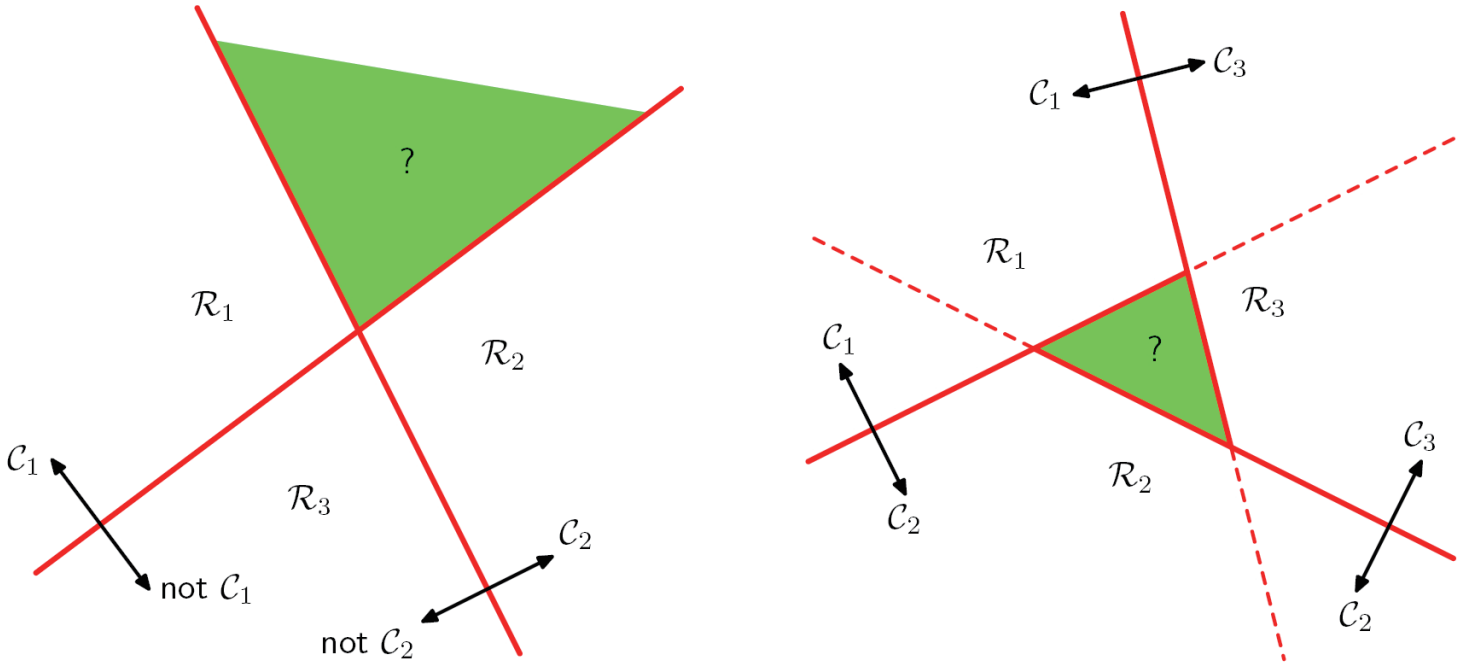


Figure 2: (K-1) classifier and $K(K\text{-}1)/2$ classifier

Linear discriminant function

K-class discriminant

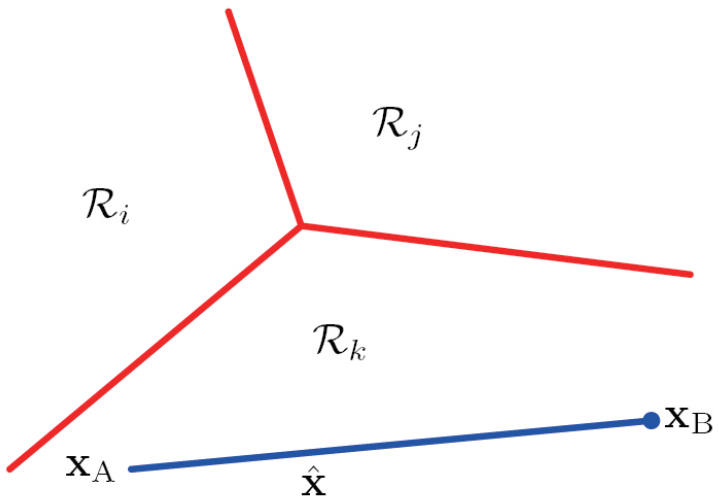
$$y_k(\mathbf{x}) = \mathbf{w}_k^T \mathbf{x} + w_{k0}$$

$$\text{assign } \mathcal{C}_k \text{ if } y_k(\mathbf{x}) > y_j(\mathbf{x}) \text{ for all } j \neq k$$

- Decision surface

$$(\mathbf{w}_k - \mathbf{w}_j)^T \mathbf{x} - (w_{k0} - w_{j0}) = 0$$

- Singly connected and convex



Least squares for classification

Min of sum-of-squares error $\rightarrow$ closed form solution

$$y_k(\mathbf{x}) = \mathbf{w}_k^T \mathbf{x} + w_{k0}$$

$$\mathbf{y}(\mathbf{x}) = \widetilde{\mathbf{W}}^T \widetilde{\mathbf{x}} \quad \text{where } \widetilde{\mathbf{w}}_k = (w_{k0}, \mathbf{w}_k^T)^T \quad \widetilde{\mathbf{x}} = (1, \mathbf{x}^T)^T$$

$$\text{assign} \ \ \arg\max_k y_k$$

Given a training dataset $\{\mathbf{x}_n, \mathbf{t}_n\}$ $(n = 1, 2, \cdots, N)$

$$E_D(\widetilde{\mathbf{W}}) = \frac{1}{2} \mathrm{Tr} \left\{ (\widetilde{\mathbf{X}} \widetilde{\mathbf{W}} - \mathbf{T})^T (\widetilde{\mathbf{X}} \widetilde{\mathbf{W}} - \mathbf{T}) \right\}$$

$$\widetilde{\mathbf{W}} = (\widetilde{\mathbf{X}}^T \widetilde{\mathbf{X}})^{-1} \widetilde{\mathbf{X}}^T \mathbf{T}$$

$$\mathbf{T} \in \{0, 1\}^{N \times K} \quad \widetilde{\mathbf{X}} \in \Re^{N \times (D+1)}$$

Least squares for classification

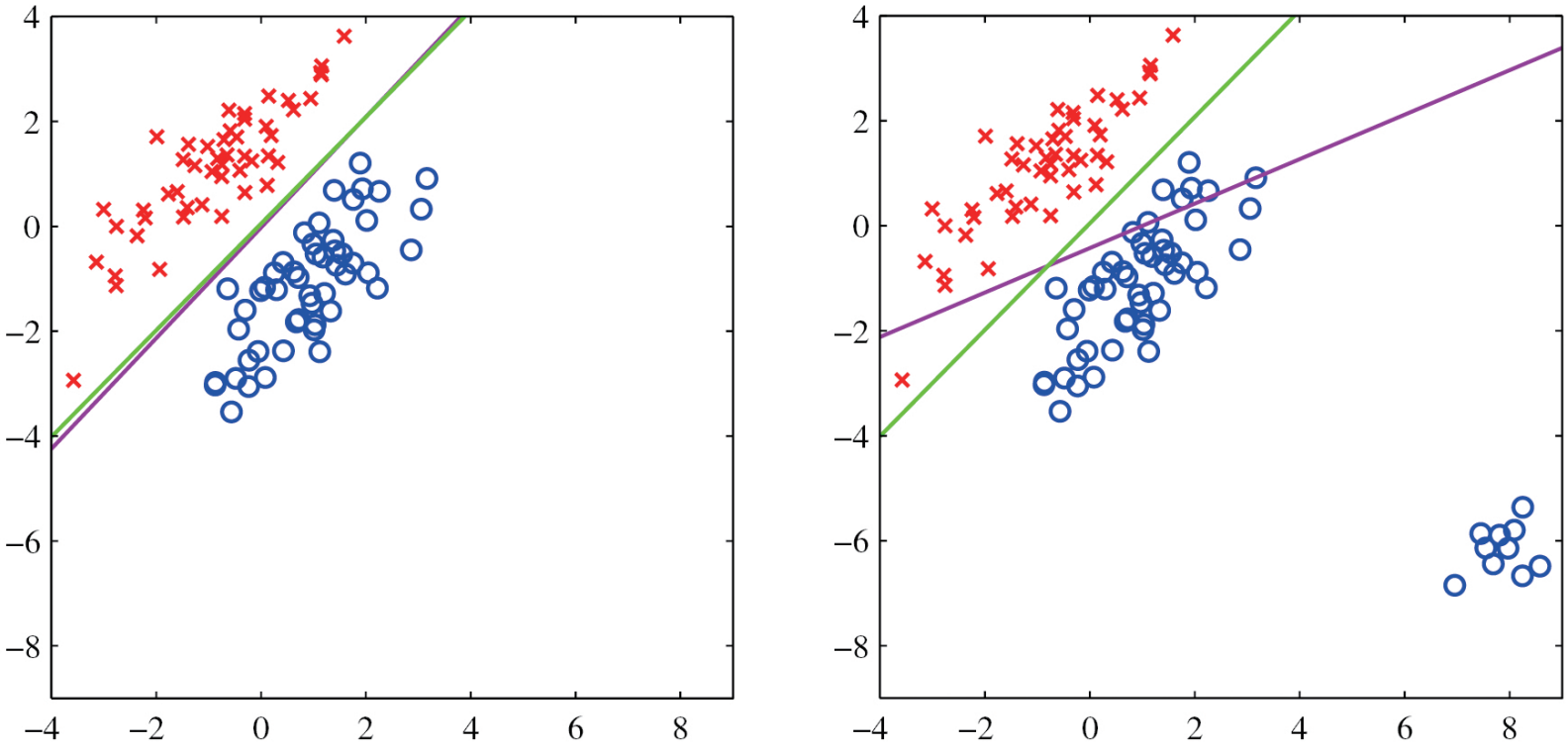


Figure 3: Two classes: least square and logistic regression

Least squares for classification

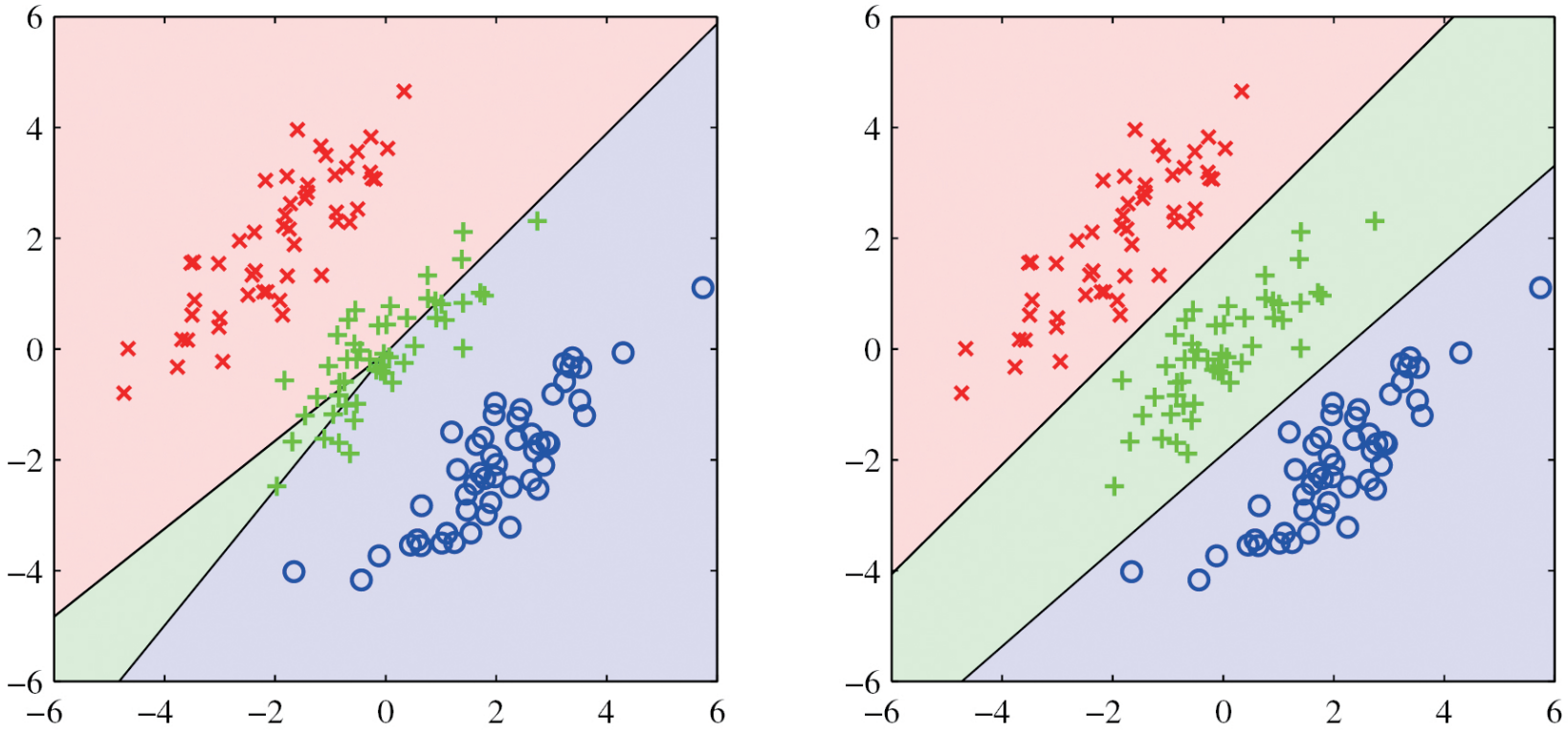


Figure 4: Multiple classes: least square and logistic regression

Perceptron algorithm

The perceptron algorithm

Generalized linear model

$$y(\mathbf{x}) = f(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}))$$

Nonlinear activation function given by a step function

$$f(a) = \begin{cases} +1, & a \geq 0 \\ -1, & a < 0 \end{cases}$$

Target values $t = +1$ for class $\mathcal{C}_1$ and $t = -1$ for class $\mathcal{C}_2$.

The perceptron algorithm

Perceptron criterion that we want to minimize

$$E_P(\mathbf{w}) = -\sum_{n \in \mathcal{M}} \mathbf{w}^T \boldsymbol{\phi}_n t_n$$

Misclassified pattern $\mathcal{M} = \{\mathbf{x}_n : \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n) t_n < 0\}$

The stochastic gradient descent algorithm

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta \nabla E_P(\mathbf{w}) = \mathbf{w}^{(\tau)} + \eta \boldsymbol{\phi}_n t_n$$

The perceptron algorithm