Knowledge Discovery & Data Mining

— Linear, Logistic Regression and Perceptron —

Instructor: Yong Zhuang

yong.zhuang@gvsu.edu

Outline

- Linear Regression
- Perceptron
- Logistic Regression

What is Linear Regression?

Linear regression is a statistical technique to predict a continuous target variable using one or more independent variables.

Examples:

- Predicting house prices based on the living area
- Estimating income based on education, major, and GPA, etc.

Linear regression

Problem Setup:

- Suppose we have n tuples, each represented by p attributes $x_i = (x_{i,1}, \ldots, x_{i,p})^T$ and a continuous output value y_i (for $i = 1, \ldots, n$).

In linear regression, we aim to learn a linear function that maps the p input attributes x_i to the output variable y_i:

$$\hat{y}_i = w^T x_i + b = \sum_{j=1}^{p} w_j x_{i,j} + b$$

where:

- $\hat{y}_i$: Predicted output for the i-th sample.
- $w = (w_1, \ldots, w_p)^T$: Weight vector representing the importance of each input attribute.
- b: Bias term, representing the baseline offset of the prediction.

Model Interpretation:

- Each weight w_j indicates the influence of the corresponding attribute $x_{i,j}$ on predicting $\hat{y}_i$.
- Linear regression assumes a linear relationship between inputs and output, with weights summing the contributions of each attribute, offset by b.

Determining the Optimal Weights and Bias

Objective: To learn the "best" weight vector w and bias scalar b from the training data. This allows the linear regression model to make the best possible predictions, where the predicted value $\hat{y}_i = w^T x_i + b$ is as close as possible to the observed value y_i (for $i = 1, \ldots, n$).

Method: Least Squares Regression

- **Goal:** Minimize the sum of squared differences between the predicted and observed values.
- **Loss Function:**

$$L(w, b) = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} (y_i - (w^T x_i + b))^2$$

The best weight vector w and bias scalar b are those that minimize $L(w, b)$, representing the total squared error.

Special Case: Single Input Attribute ($p = 1$)

- **Optimal Weight:**

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

- **Optimal Bias:**

$$b = \frac{1}{n} \sum_{i=1}^{n} (y_i - w x_i)$$

where $\bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_i$ is the average observed output value across all n training samples.

Example of Least Squares Regression

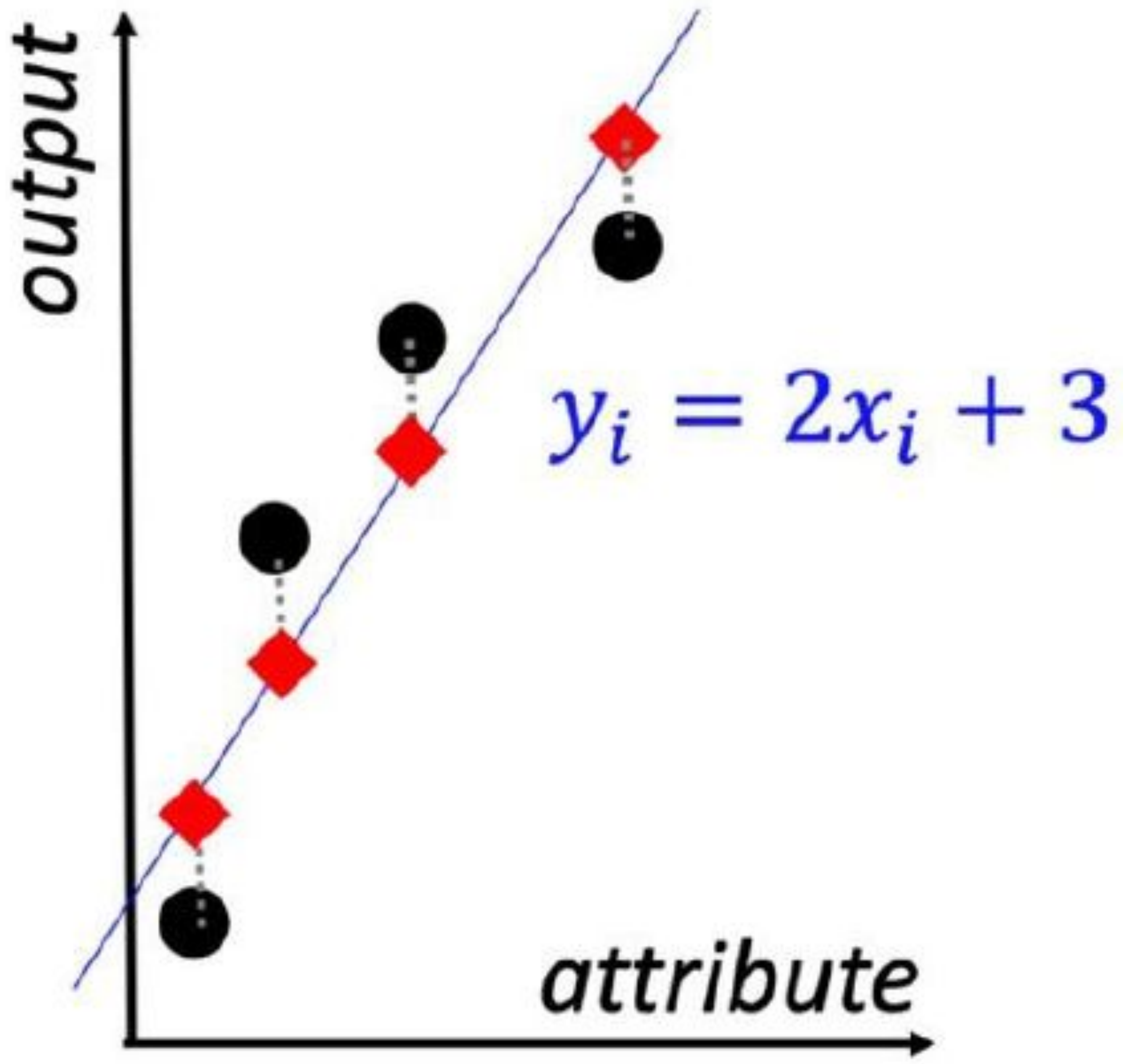
Example. we have four training tuples, each represented by a single attribute x_i and an output variable y_i. We want to find the least squares regression model $y = wx + b$ that best predicts the output y based on the input attribute x.

Index (i)	1	2	3	4
Attribute (x_i)	1	3	5	7
Output (y_i)	4	10	14	16

(a) Training tuples

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

$$b = \frac{1}{n} \sum_{i=1}^{n} (y_i - w x_i).$$



(b) Least square regression

$$\bar{y} = (y_1 + y_2 + y_3 + y_4)/4 = (4 + 10 + 14 + 16)/4 = 11$$

$$\sum_{i=1}^{4} x_i^2 = 1^2 + 3^2 + 5^2 + 7^2 = 84$$

$$\sum_{i=1}^{4} x_i (y_i - \bar{y}) = 1(4-11) + 3(10-11) + 5(14-11) + 7(16-11) = 40$$

$$w = \frac{\sum_{i=1}^{n} x_i (y_i - \bar{y})}{\sum_{i=1}^{n} x_i^2 - \frac{1}{n}(\sum_{i=1}^{n} x_i)^2} = \frac{40}{84-64} = 2$$

$$b = \frac{\sum_{i=1}^{4} (y_i - w x_i)}{4} = \frac{(4-2\times1)+(10-2\times3)+(14-2\times5)+(16-2\times7)}{4} = 3$$

Perceptron: Modifying Linear Regression for Classification

Suppose We have a binary classification task with:

- $y_i = +1$: Indicates a positive outcome (e.g., buy computer)
- $y_i = 0$: Indicates a negative outcome (e.g., not buy computer)

Modification Approach:

To modify linear regression for classification, we use the **sign** of the output from the linear regression model as the predicted class label.

1. **Prediction Formula**:

$$\hat{y}_i = \text{sign}(w^T x_i)$$

*Introduce a dummy attribute with a constant value of 1 for all tuples to allow a bias term (**b**) in the model.*

   - Where $\hat{y}_i$ is the predicted class label for the i-th tuple.
   - $\text{sign}(z) = 1$ if $z > 0$ (positive class) and $\text{sign}(z) = 0$ otherwise (negative class).

2. **Linear Combination**:

   - We compute a weighted combination of the input attributes:
$$z = w^T x_i$$
   - If z is positive, the tuple is classified as a positive case; otherwise, it is classified as negative.

Perceptron: Modifying Linear Regression for Classification

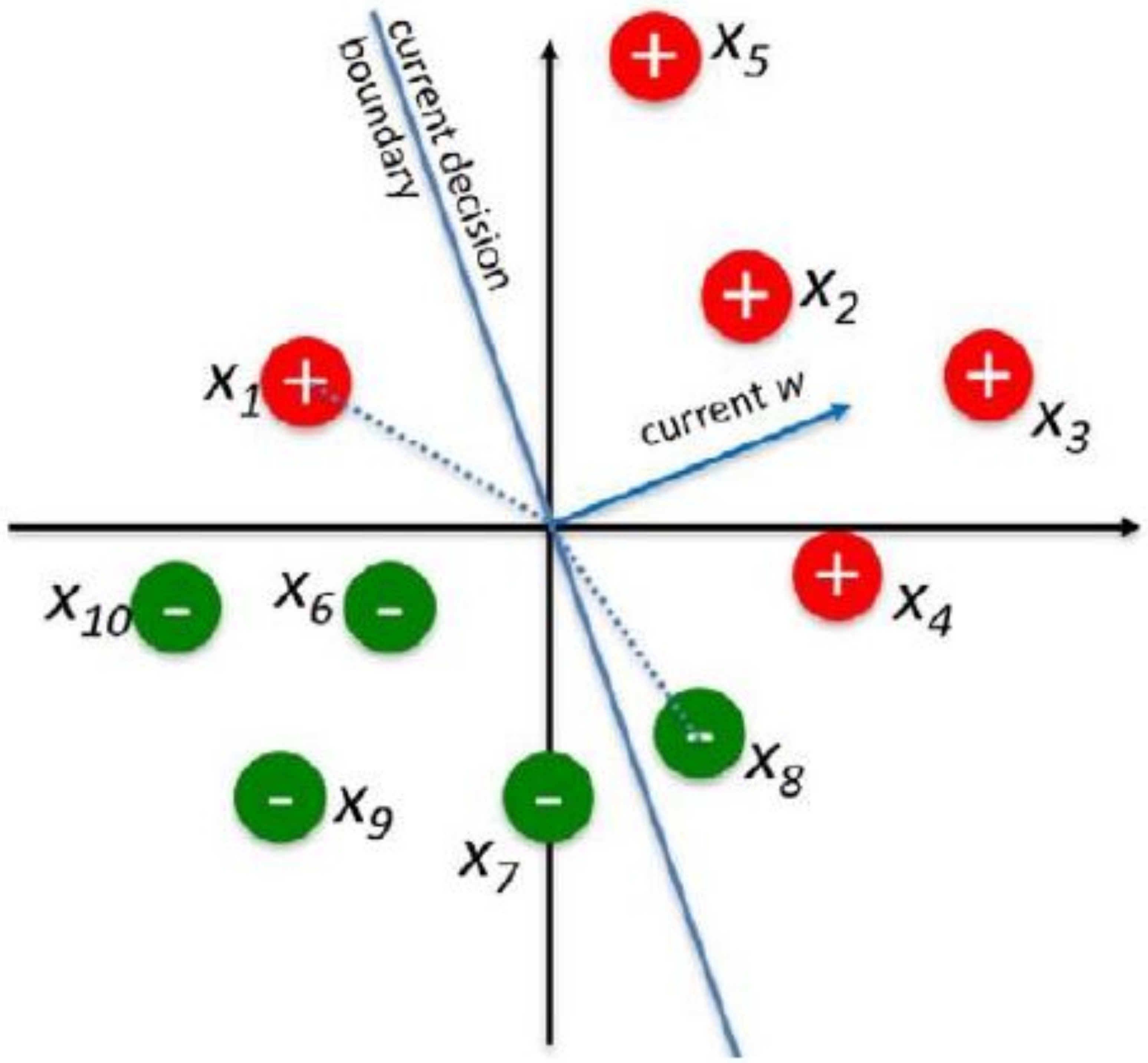
How can we find the optimal weight vector w from a set of training tuples?

The perceptron algorithm iteratively adjusts the weight vector based on errors in predictions, gradually converging towards a solution.

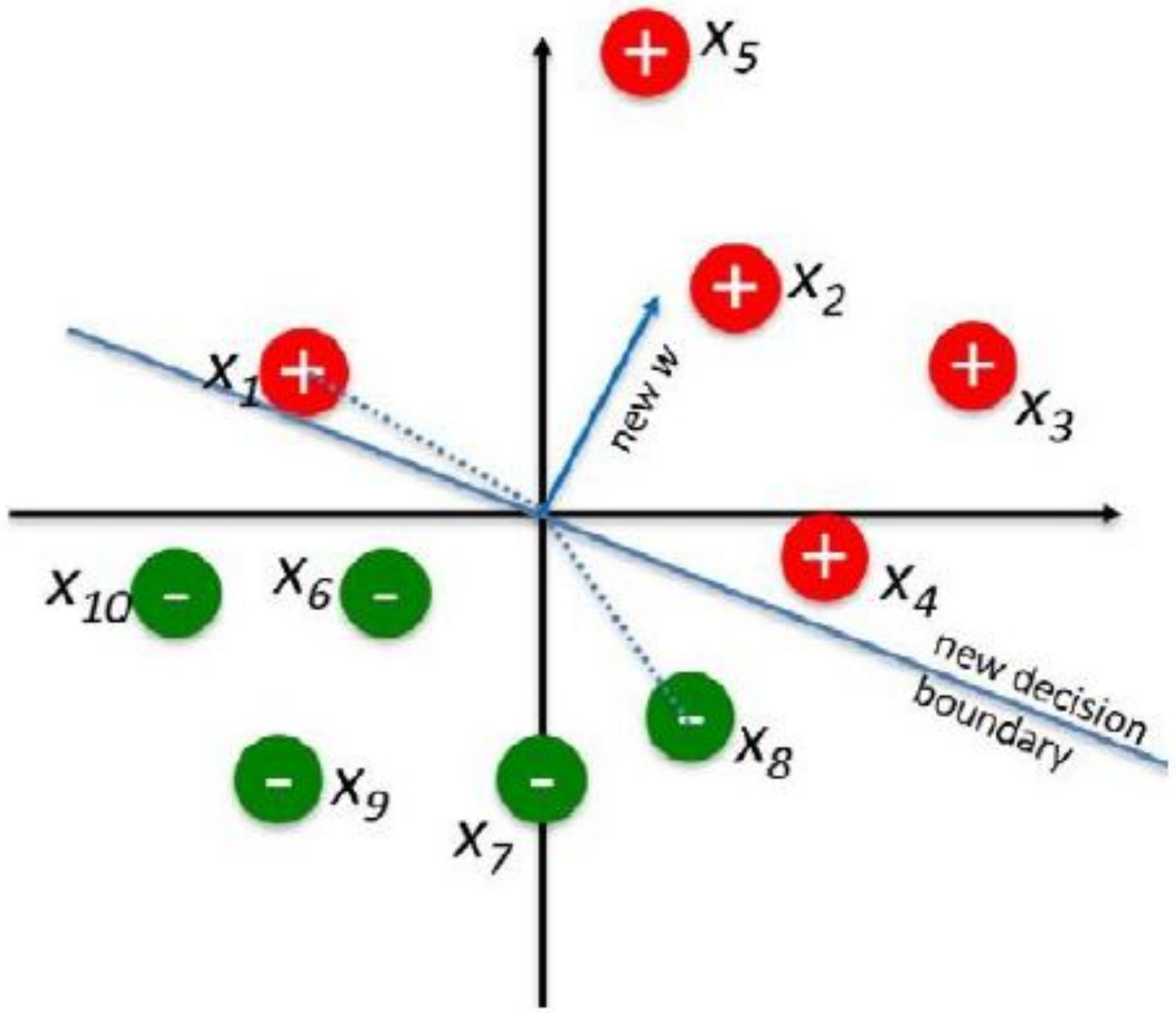
Training Algorithm:

1. **Initialize Weight Vector:**
   - Start with an initial guess for the weight vector w (e.g., set $w = 0$).
2. **Iterative Update Process:**
   - Continue updating until convergence or a maximum number of iterations is reached:
     - For each training tuple x_i:
       - **If Prediction is Correct:** No change to w.
       - **If Prediction is Incorrect:**
         - **Positive Tuple** $(+1)$:
           Update $w \leftarrow w + \eta x_i$
         - **Negative Tuple** (-1):
           Update $w \leftarrow w - \eta x_i$
       - Here, η is the learning rate, controlling the size of each update.

Perceptron: linear regression to classification



(a) current weight vector w



(b) updated weight vector w

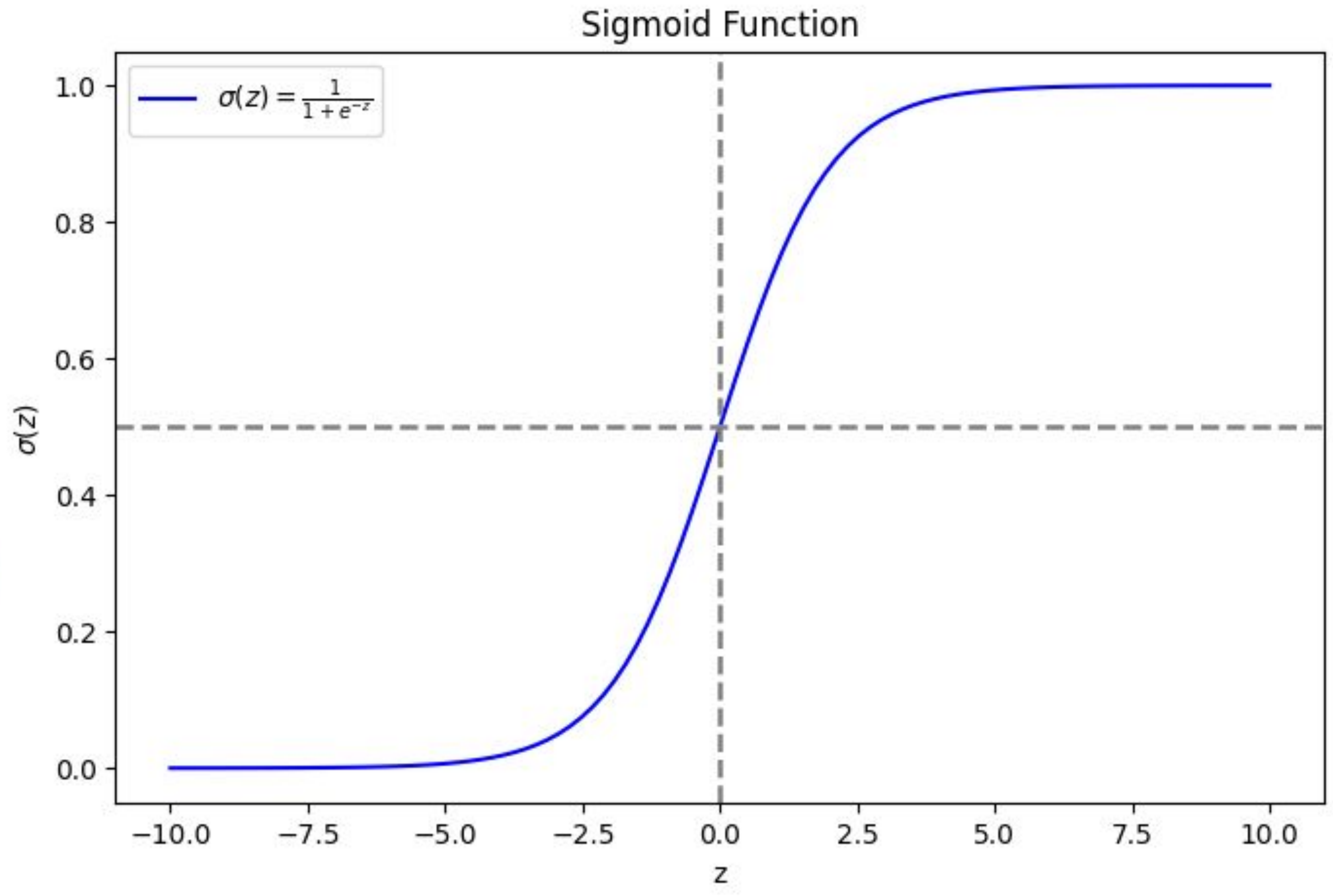
Logistic Regression: Sigmoid Function

How can we make a linear classifier not only predict which class label a tuple has but also tell how confident it is in making such a prediction?

Solution: Logistic regression provides a probabilistic framework for classification by leveraging the sigmoid function to produce confidence scores.

Sigmoid Function: The sigmoid function, denoted as $\sigma(z)$, maps any real-valued number into the interval (0,1), which can be interpreted as a probability.

$$\sigma(z) = \frac{1}{1+e^{-z}} = \frac{e^z}{1+e^z}$$



Logistic Regression Model:

In logistic regression, the probability of the class label being 1, given the input x_i and weight vector w, is computed as:

$$P(\hat{y}_i = 1 | x_i, w) = \sigma(w^T x_i) = \frac{1}{1+e^{-w^T x_i}}$$

Logistic regression: Determining the Optimal Weights

How can we find the optimal weight vector w from a set of training tuples?

maximum likelihood estimation (MLE).

Assume there are n training tuples (x_i, y_i) for $i = 1, \ldots, n$.

- Each true class label y_i for the i-th tuple is a binary variable, where $y_i \in \{0, 1\}$.
- The probability of correct classification for a tuple is given by:

$$P(\hat{y}_i = y_i) = p_i^{y_i}(1 - p_i)^{1-y_i}$$

where p_i is the probability of class 1 for the i-th tuple.

The likelihood function $L(w)$ is the product of probabilities for each training tuple:

$$L(w) = \prod_{i=1}^{n} p_i^{y_i}(1 - p_i)^{1-y_i}$$

Substituting $p_i = \frac{e^{w^T x_i}}{1+e^{w^T x_i}}$, we get:

$$L(w) = \prod_{i=1}^{n} \left(\frac{e^{w^T x_i}}{1 + e^{w^T x_i}}\right)^{y_i} \left(\frac{1}{1 + e^{w^T x_i}}\right)^{1-y_i}$$

Optimal Model Parameter

The optimal weight vector w^* maximizes the log-likelihood function $l(w) = \log L(w)$:

$$w^* = \arg\max_{w} l(w) = \arg\max_{w} \left(\sum_{i=1}^{n} y_i x_i^T w - \log(1 + e^{w^T x_i})\right)$$

Summary

- Linear Regression
- Perceptron
- Logistic Regression