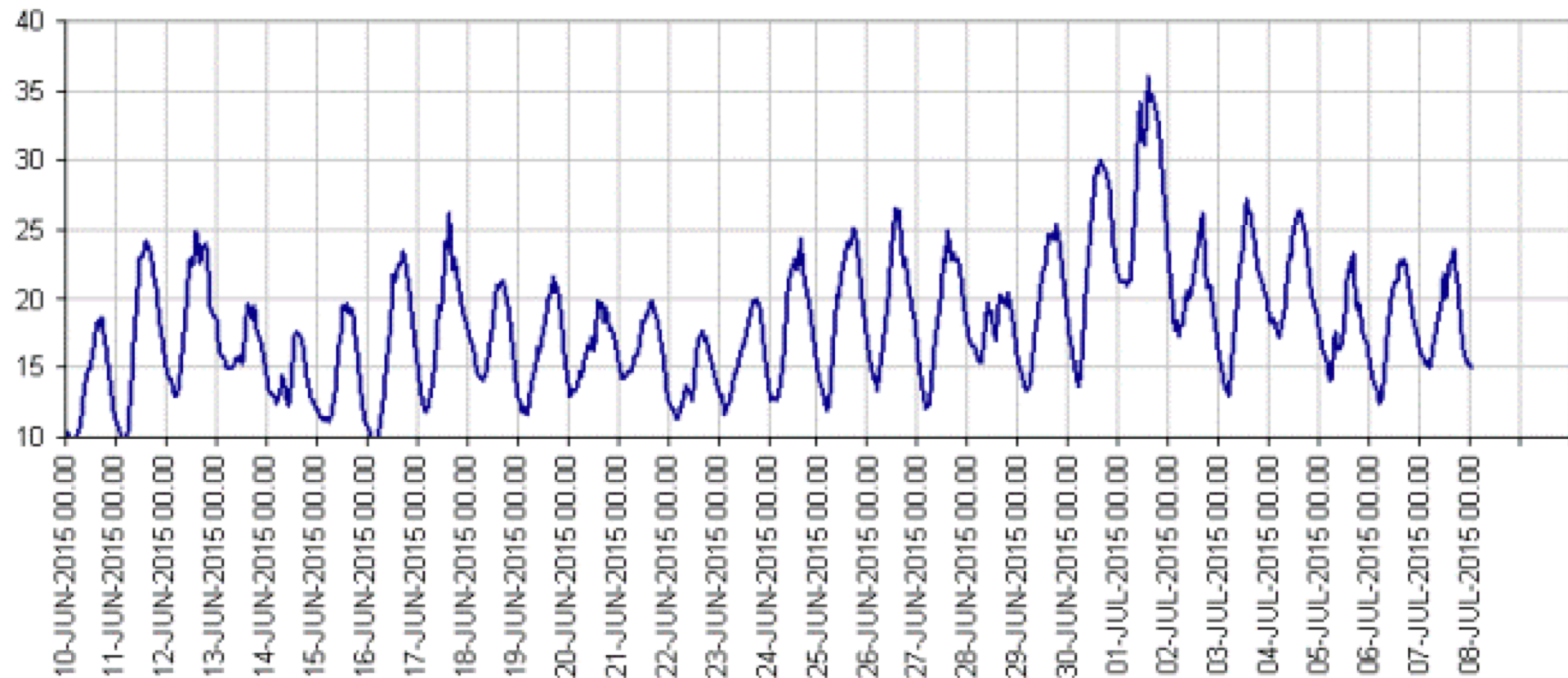


Linear regression

Example: weather prediction



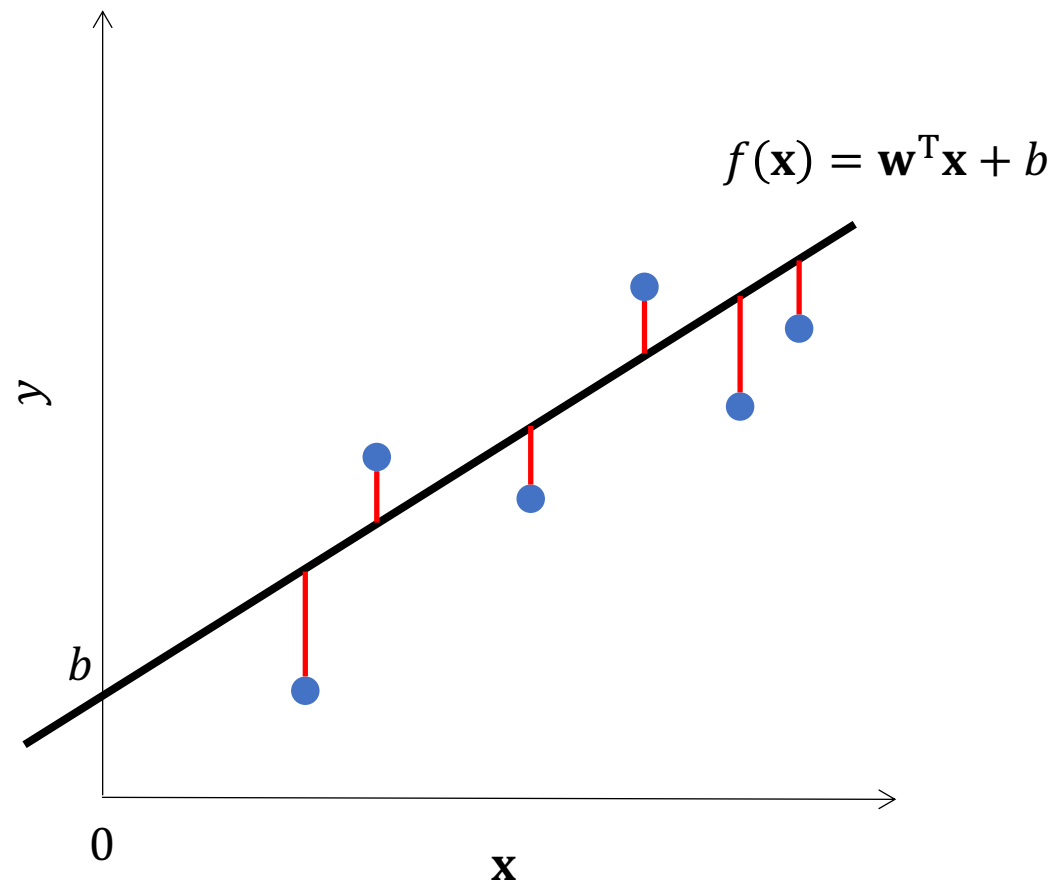
Forecast air temperature in London

Linear model

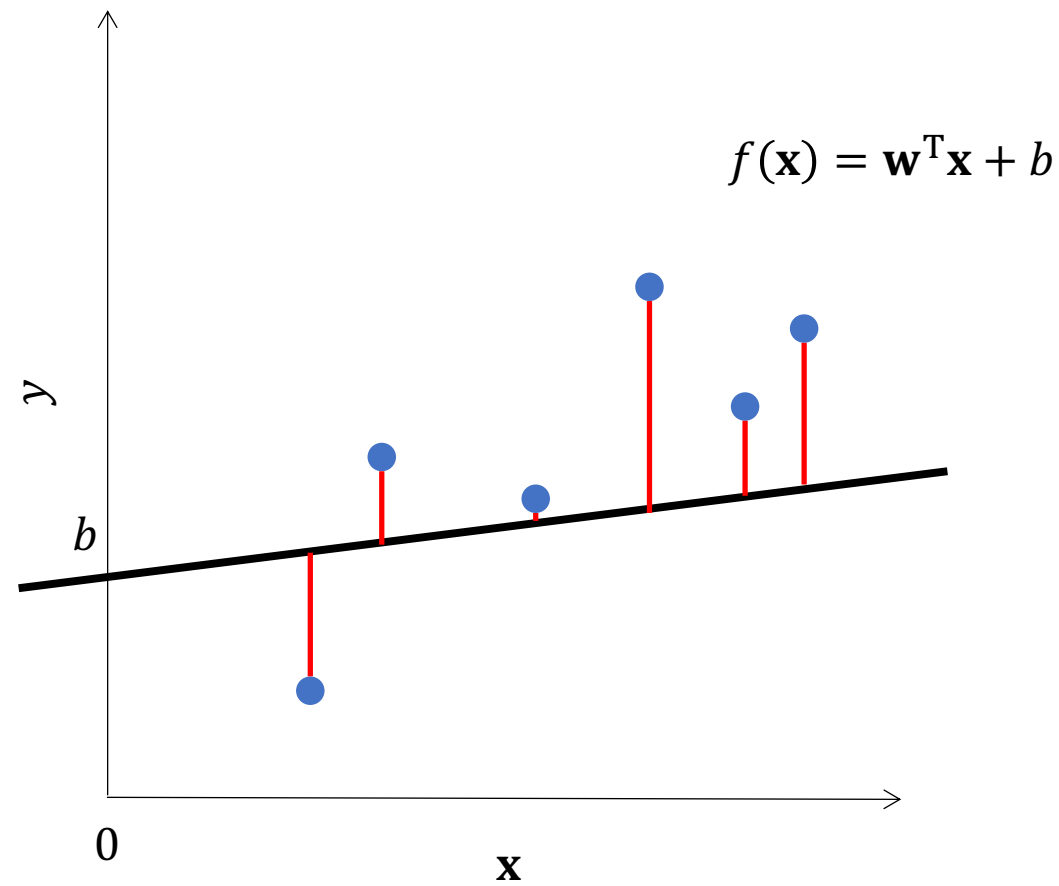
- Given training data $\{(\mathbf{x}_i, y_i) \sim p \text{ i.i.d.}\}_{i=1}^n$
- Assuming linear model $\hat{y} = f_{\mathbf{w},b}(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b$
- Find optimal parameters \mathbf{w}, b by minimizing empirical loss

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

Linear model



Linear model



Finding optimal parameters

- Matrix-vector notation $\mathbf{X}\mathbf{w} + b\mathbf{1} = \mathbf{y}$

$$\begin{matrix} & d \\ & \mathbf{x}_1 \\ & \mathbf{x}_2 \\ n & \mathbf{x}_3 \\ & \vdots \\ & \mathbf{x}_n \end{matrix} \quad \begin{matrix} 1 \\ \mathbf{w} \\ d \end{matrix} + \begin{matrix} 1 \\ b \\ b \\ b \\ \vdots \\ b \\ n \end{matrix} = \begin{matrix} 1 \\ \mathbf{y} \\ n \end{matrix}$$

Finding optimal parameters

- Matrix-vector notation: absorb the bias into the weight vector
 $\mathbf{X}\mathbf{w} = \mathbf{y}$

The diagram illustrates the matrix-vector equation $\mathbf{X}\mathbf{w} = \mathbf{y}$ with dimensions and structure:

- Matrix \mathbf{X} :** A matrix with n rows and $d + 1$ columns. The first d columns contain feature vectors $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \dots, \mathbf{x}_n$. The last column contains ones, representing the bias term. The dimensions are labeled n (rows) and $d + 1$ (columns).
- Weight vector \mathbf{w} :** A vector with $d + 1$ elements. The first d elements are the weights \mathbf{w} , and the last element is the bias b . The dimensions are labeled 1 (width) and $d + 1$ (height).
- Output vector \mathbf{y} :** A vector with n elements, representing the target values. The dimensions are labeled 1 (width) and n (height).

The equation is shown as:

$$\begin{matrix} & d + 1 \\ & \mathbf{x}_1 \\ & \mathbf{x}_2 \\ n & \mathbf{x}_3 \\ & \vdots \\ & \mathbf{x}_n \\ & 1 \end{matrix} \begin{matrix} 1 \\ \mathbf{w} \\ b \end{matrix}^{d + 1} = \begin{matrix} 1 \\ \mathbf{y} \end{matrix}^n$$

Finding optimal parameters

- Loss $\hat{L}(\mathbf{w}) = \frac{1}{n} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$
- Set the gradient to zero to get the minimizer:

$$0 = \nabla_{\mathbf{w}} \hat{L}(\mathbf{w}) = \frac{1}{n} \nabla_{\mathbf{w}} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$$

$$0 = \nabla_{\mathbf{w}} [(\mathbf{X}\mathbf{w} - \mathbf{y})^T (\mathbf{X}\mathbf{w} - \mathbf{y})]$$

$$0 = \nabla_{\mathbf{w}} [\mathbf{w}^T \mathbf{X}^T \mathbf{X} \mathbf{w} - 2\mathbf{w}^T \mathbf{X}^T \mathbf{y} + \mathbf{y}^T \mathbf{y}]$$

$$0 = 2\mathbf{X}^T \mathbf{X} \mathbf{w} - 2\mathbf{X}^T \mathbf{y}$$

$$\mathbf{w} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

Regularization

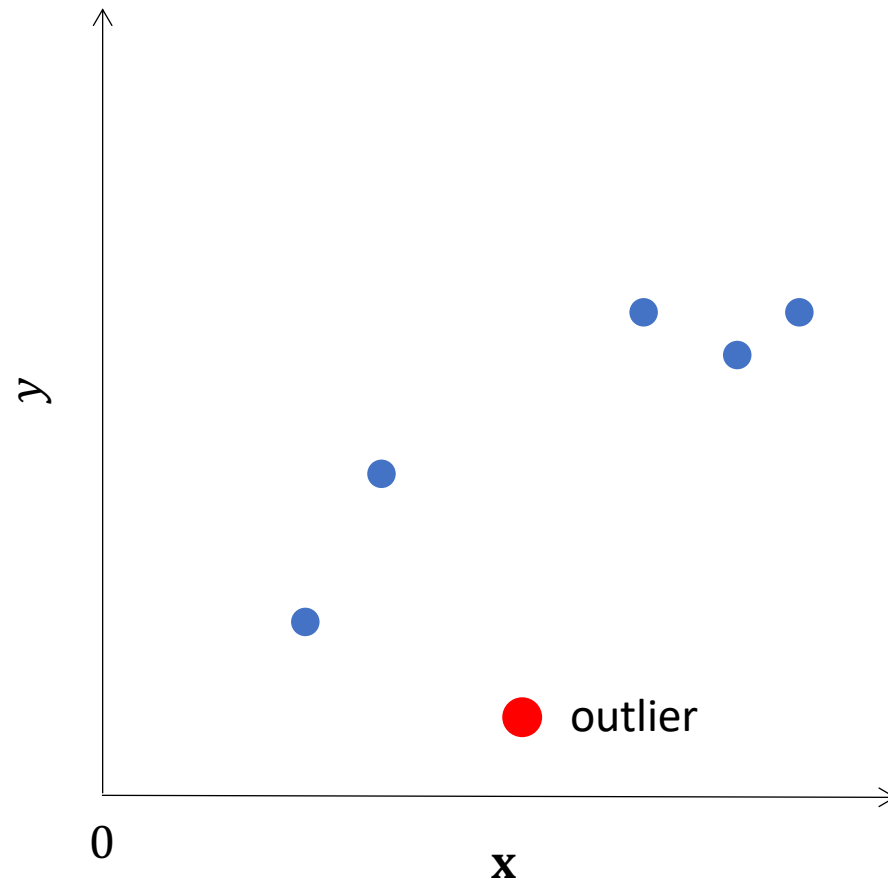
- Inverse well-defined only when $\text{rank}(\mathbf{X}^T\mathbf{X}) = d$ implying $n \geq d$
= (over-)determined system = more equations than variables
- Otherwise, add **regularization term (weight decay)**

$$\hat{L}(\mathbf{w}) = \frac{1}{n} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha \|\mathbf{w}\|^2$$

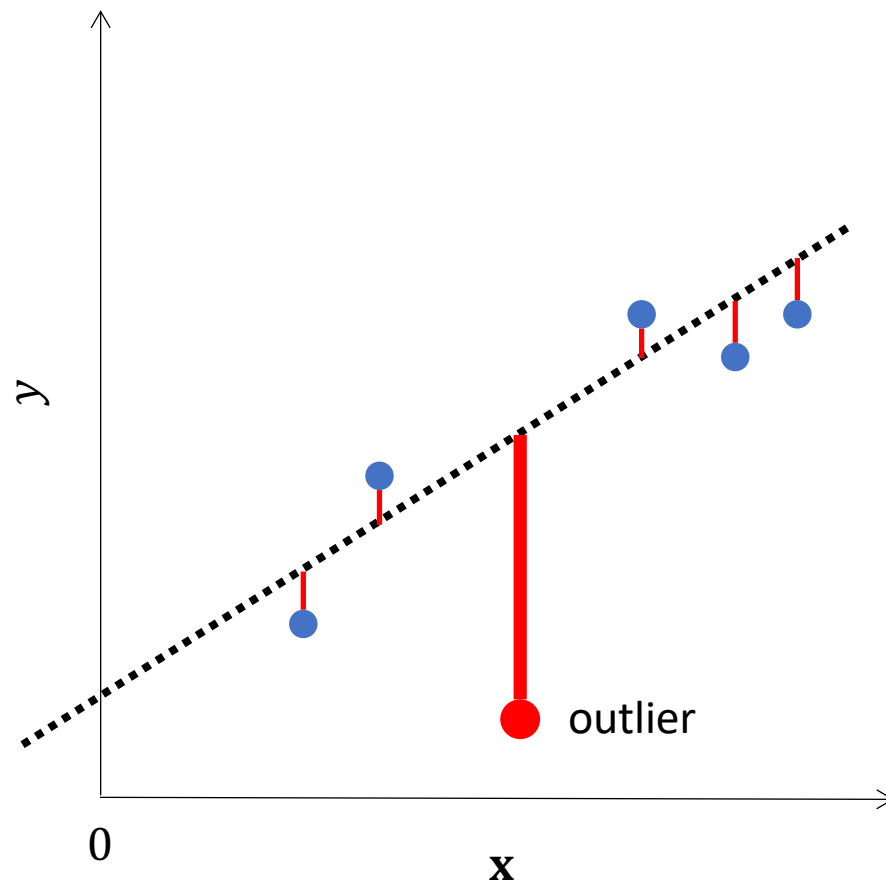
$$0 = \nabla_{\mathbf{w}} \hat{L}(\mathbf{w}) = \frac{2}{n} \mathbf{X}^T \mathbf{X} \mathbf{w} - \frac{2}{n} \mathbf{X}^T \mathbf{y} + 2\alpha \mathbf{w}$$

$$\mathbf{w} = \left(\frac{1}{n} \mathbf{X}^T \mathbf{X} + \alpha \mathbf{I} \right)^{-1} \mathbf{X}^T \mathbf{y}$$

Loss choice

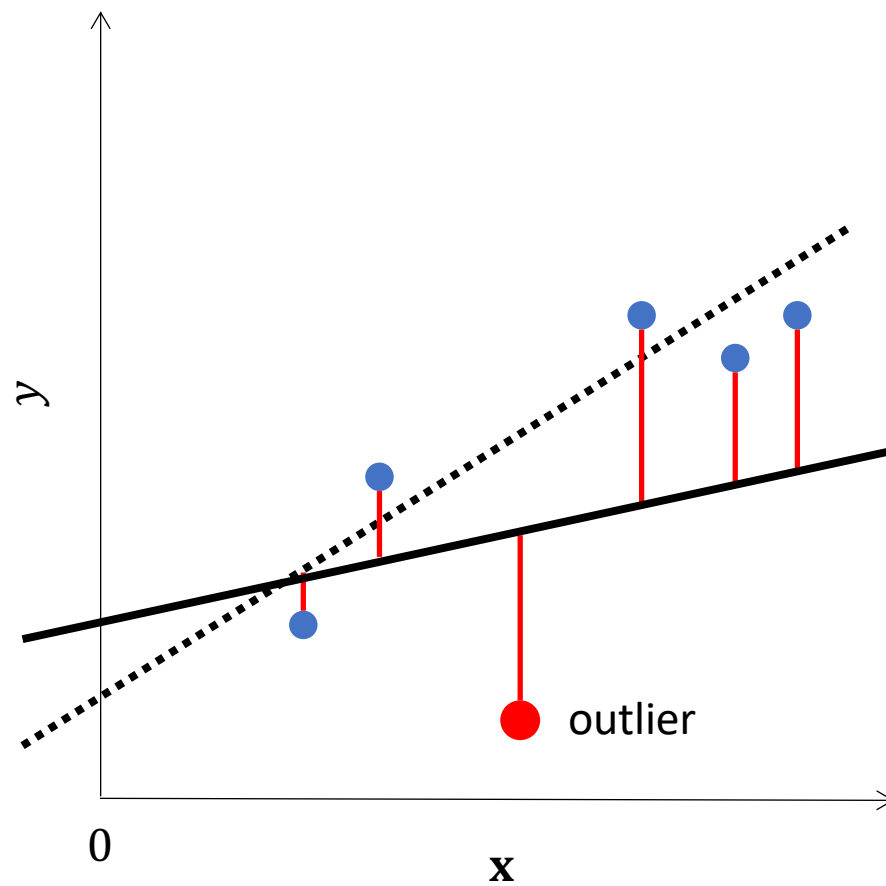


Loss choice



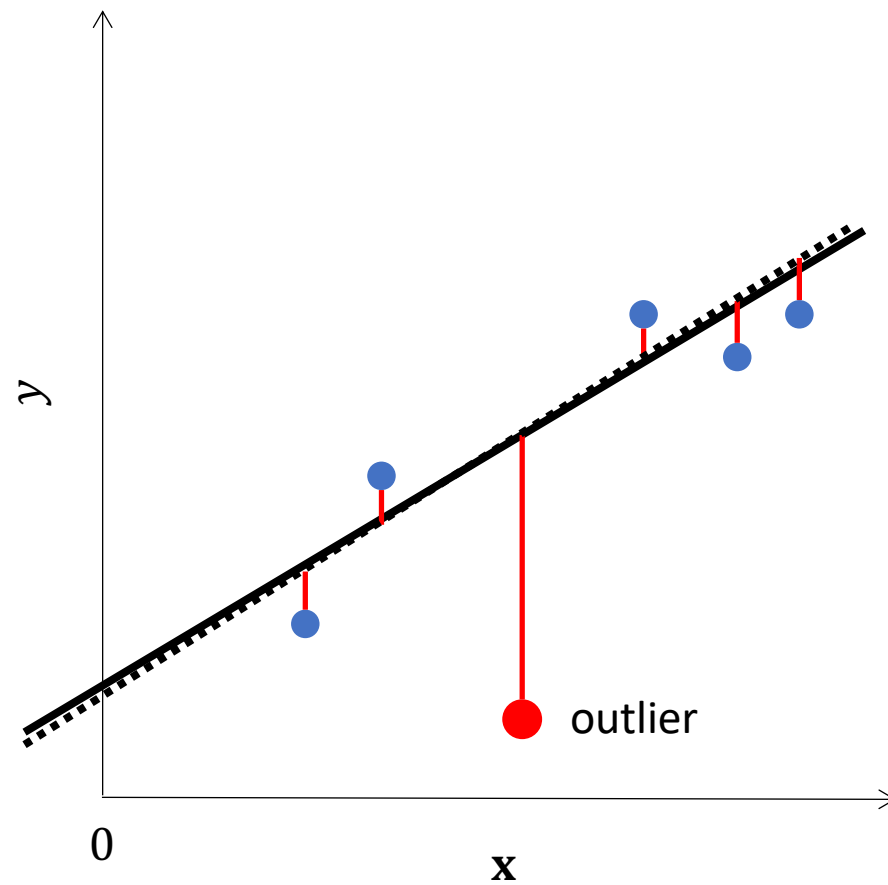
$$L_2 \text{ loss} \quad \|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2 = \frac{1}{n} \sum_{i=1}^n (\mathbf{w}^T \mathbf{x}_i + b - y_i)^2$$

Loss choice



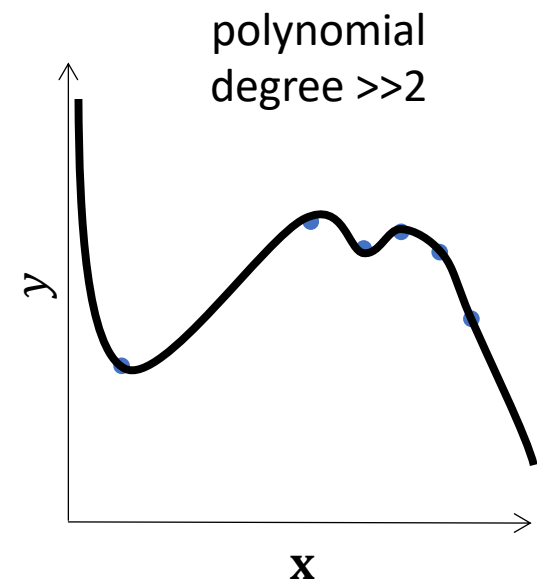
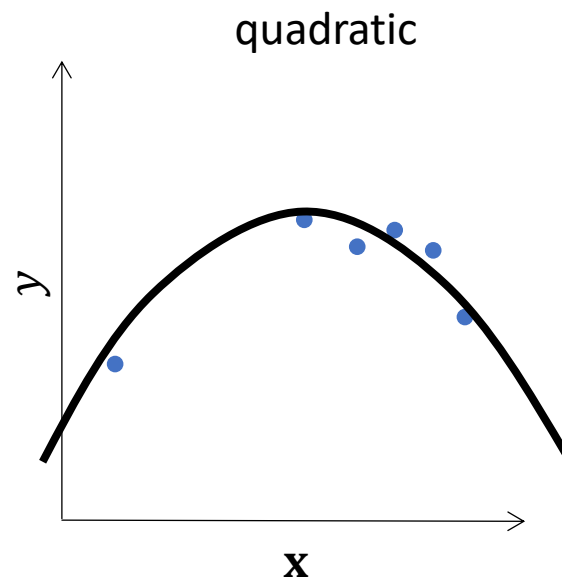
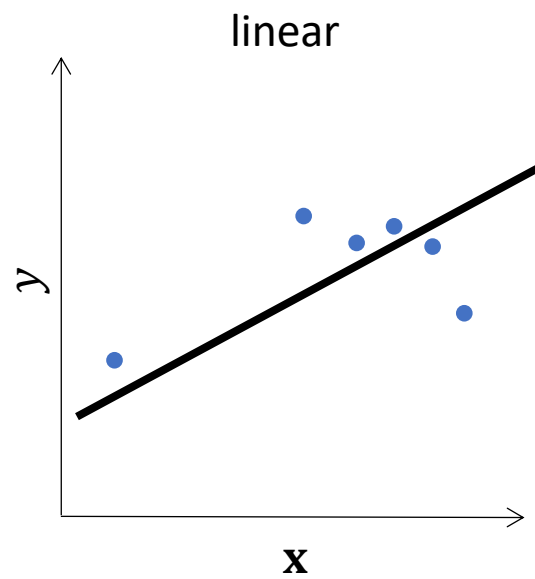
$$L_2 \text{ loss } \|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2 = \frac{1}{n} \sum_{i=1}^n (\mathbf{w}^T \mathbf{x}_i + b - y_i)^2$$

Loss choice

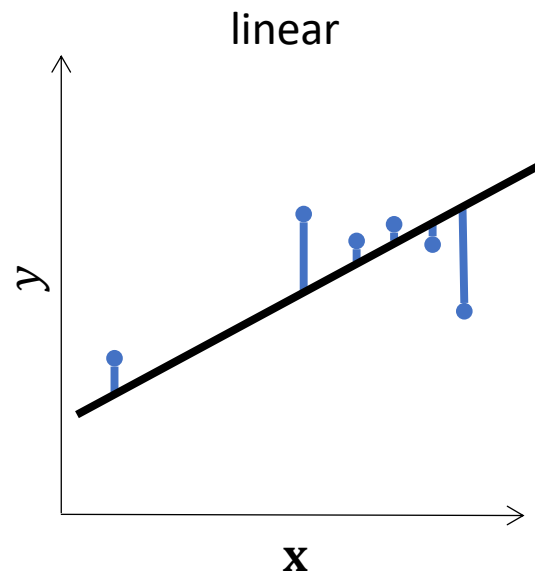


$$L_1 \text{ loss } \|\mathbf{X}\mathbf{w} - \mathbf{y}\|_1 = \frac{1}{n} \sum_{i=1}^n |\mathbf{w}^T \mathbf{x}_i + b - y_i|$$

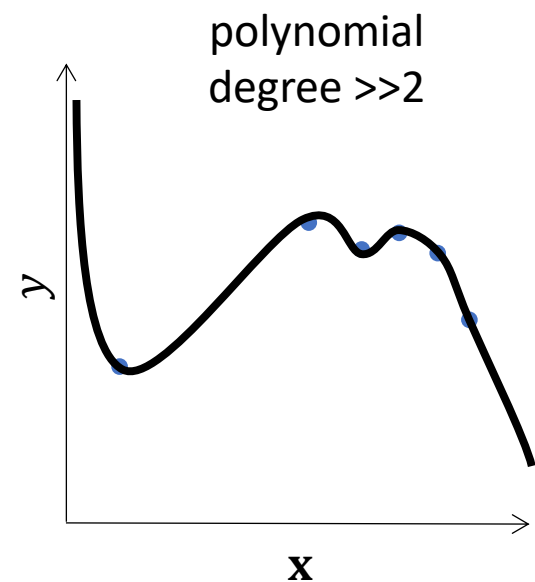
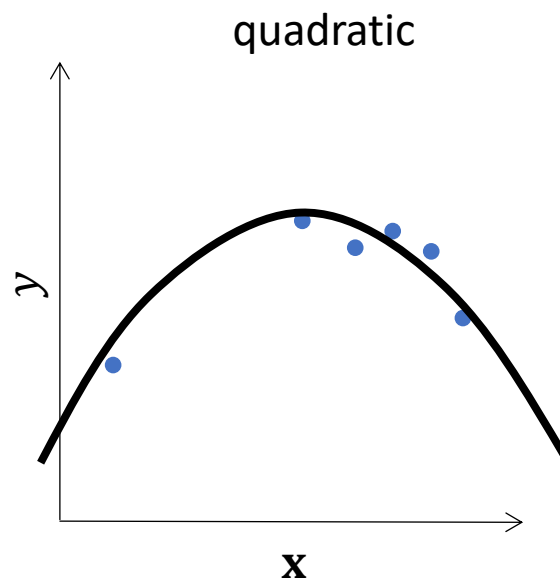
Model capacity



Model capacity



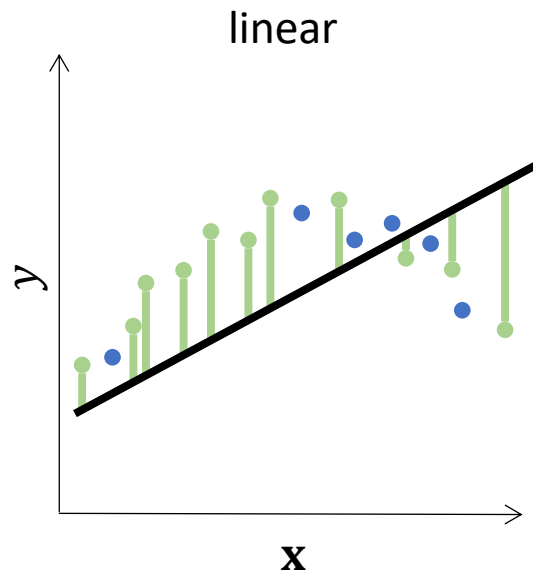
Large training error



Small training error

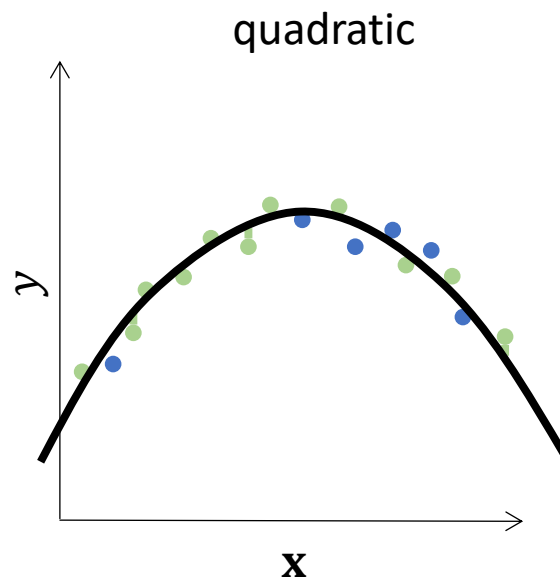
Model capacity

Underfitting

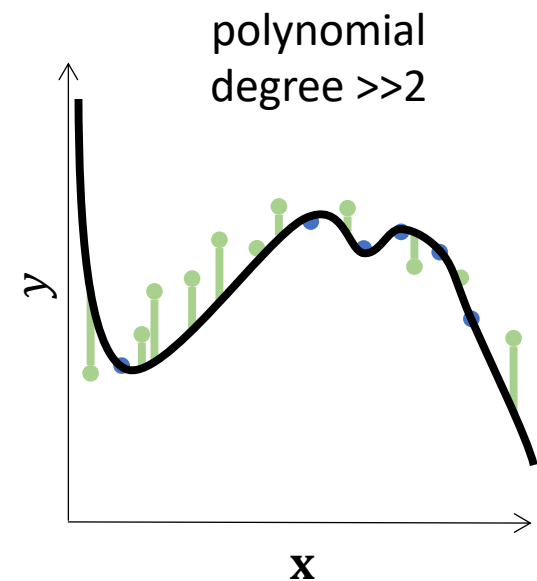


Large training error
Large generalization error

Appropriate capacity

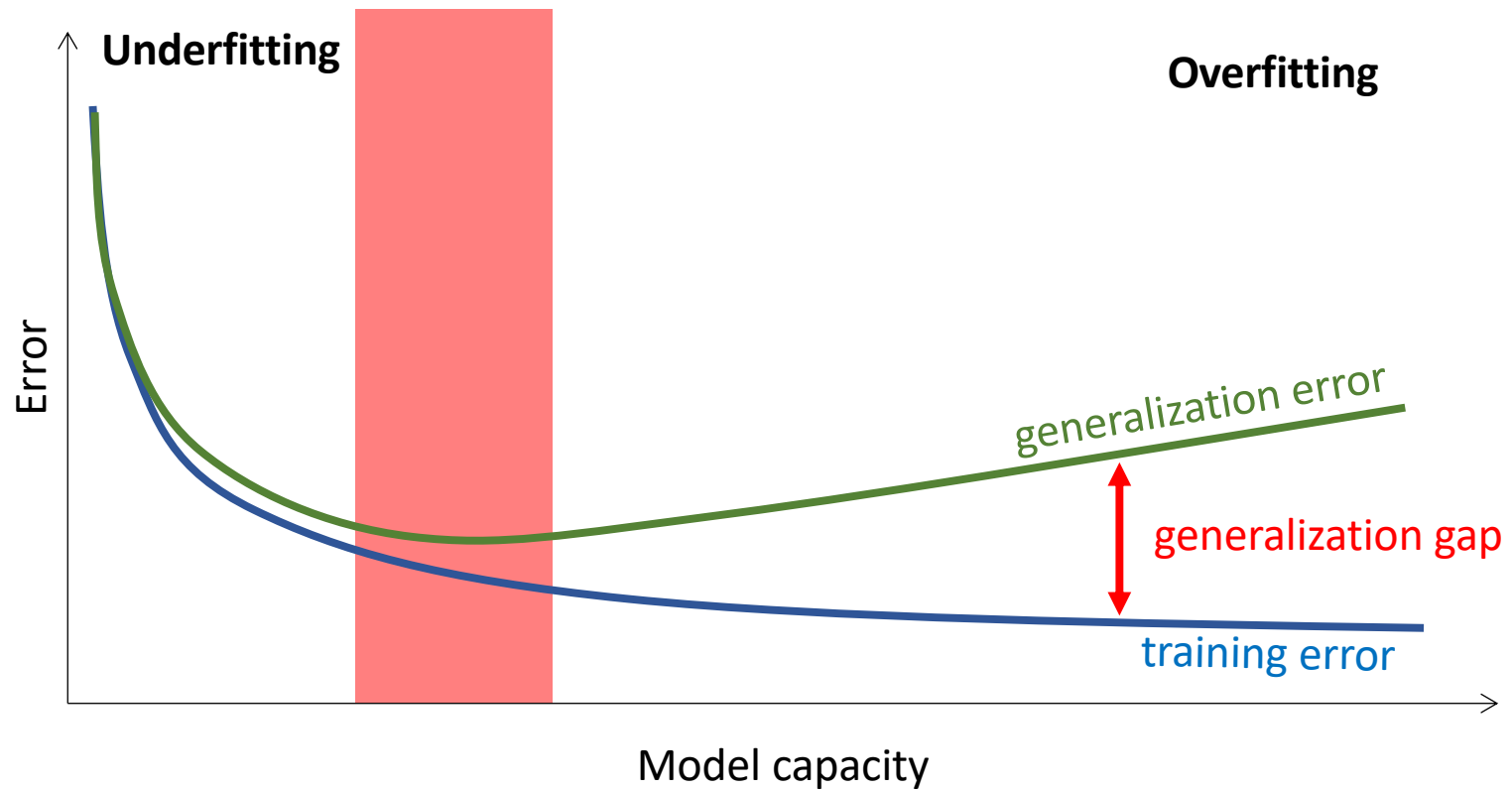


Overfitting



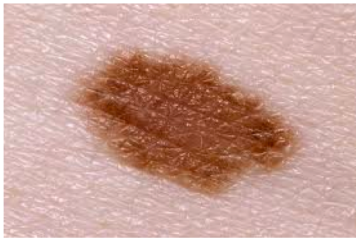
Small training error
Large generalization error

Model capacity

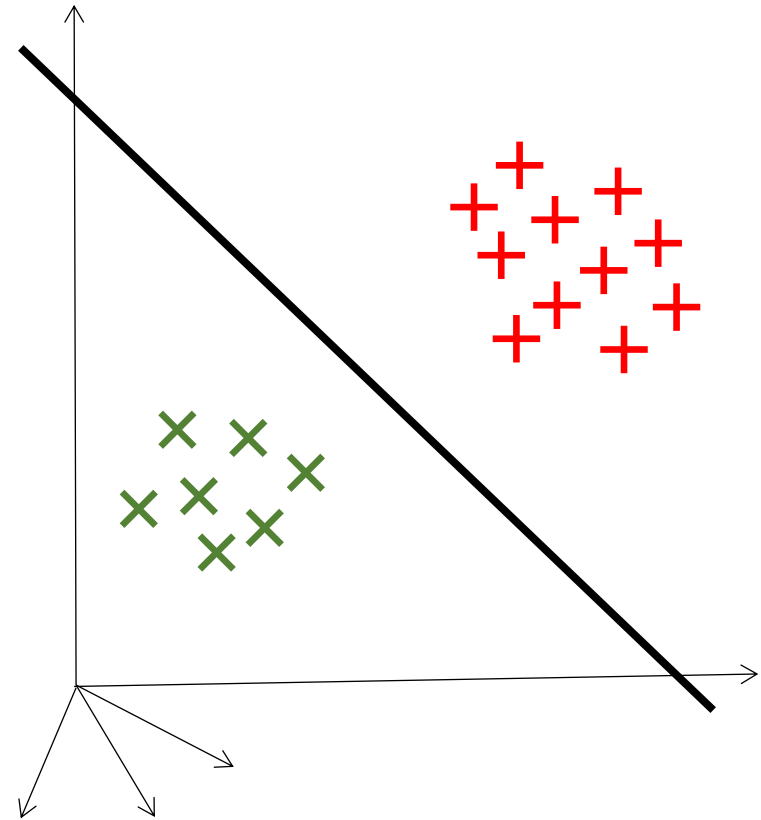


Linear classification

Linear classification



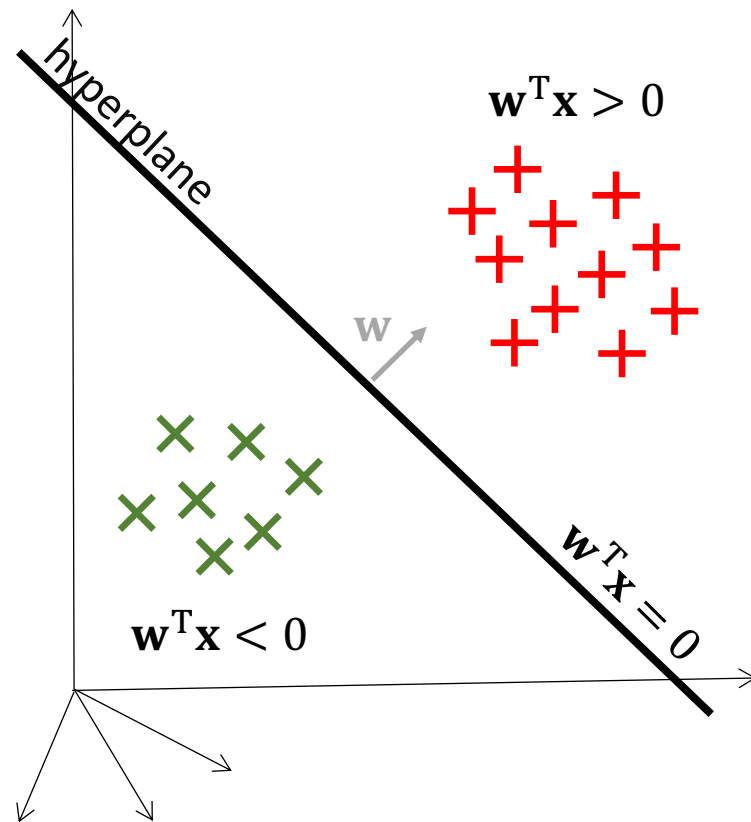
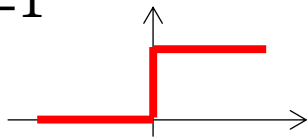
⋮



Linear classification

- Training data
 $\{(\mathbf{x}_i, y_i \in \{0,1\})\}_{i=1}^n$
- Linear model $f_{\mathbf{w}}(\mathbf{x}) = \mathbf{w}^T \mathbf{x}$
- 0-1 loss

$$\hat{L}(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^n \mathbb{I}_{\text{step}(\mathbf{w}^T \mathbf{x}_i) \neq y_i}$$



Hard to optimize!

* Often class labels in binary classification problems are denoted by +1 and -1

Linear classification

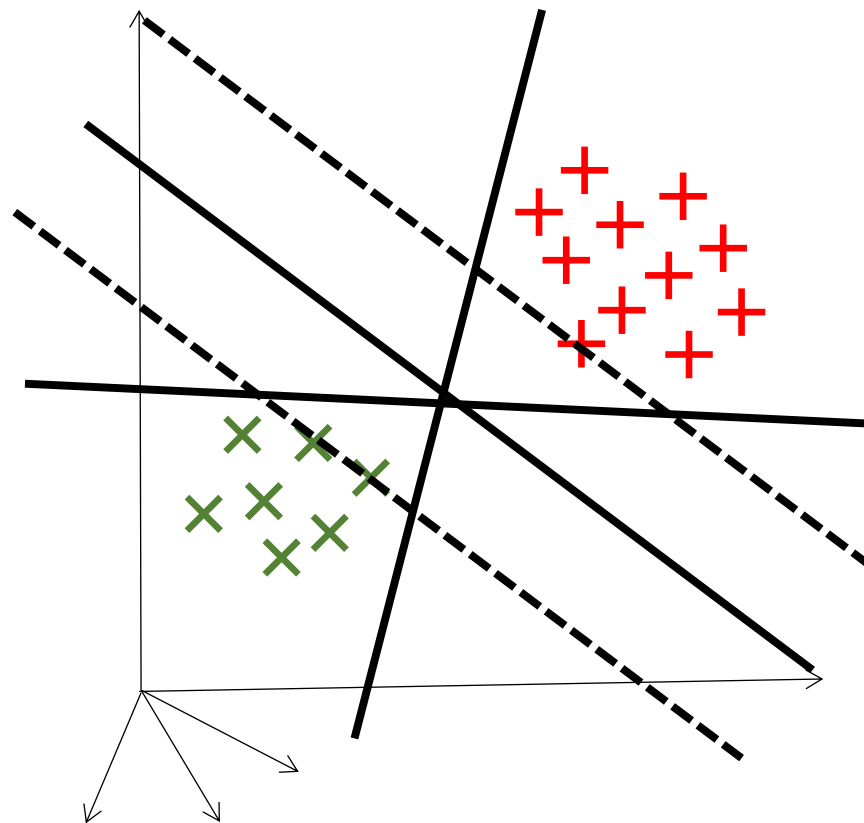
- Training data

$$\{(\mathbf{x}_i, y_i \in \{0,1\})\}_{i=1}^n$$

- Linear model $f_{\mathbf{w}}(\mathbf{x}) = \mathbf{w}^T \mathbf{x}$

- 0-1 loss

$$\hat{L}(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^n \mathbb{I}_{\text{step}(\mathbf{w}^T \mathbf{x}_i) \neq y_i}$$



Many possible solutions Support vector machine

Linear classification

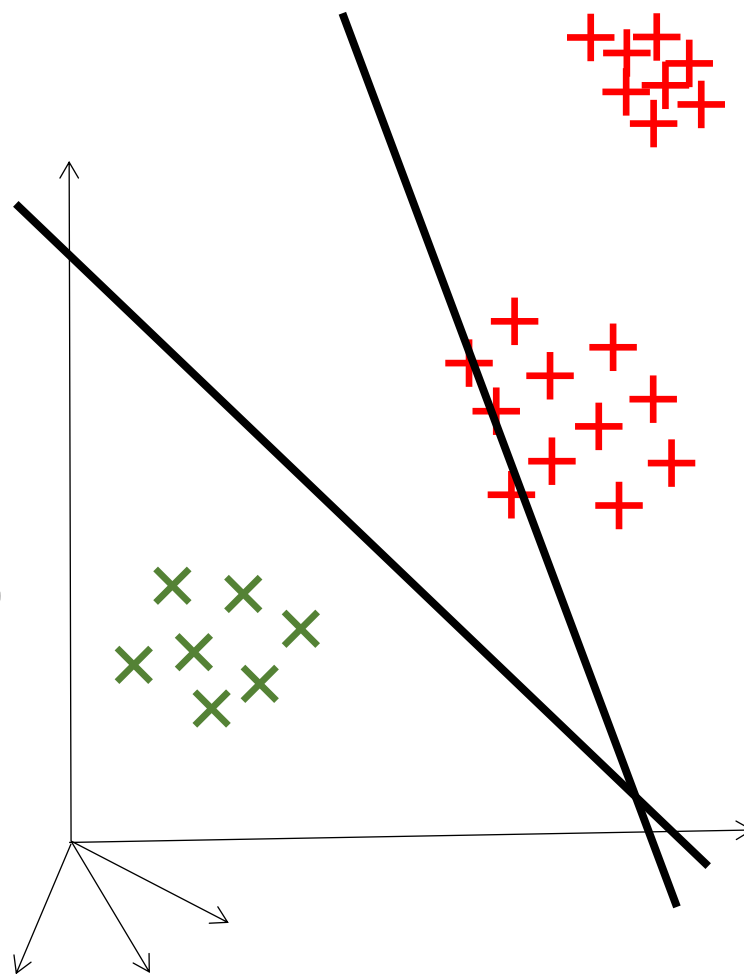
- Training data

$$\{(\mathbf{x}_i, y_i \in \{0,1\})\}_{i=1}^n$$

- Linear model $f_{\mathbf{w}}(\mathbf{x}) = \mathbf{w}^T \mathbf{x}$

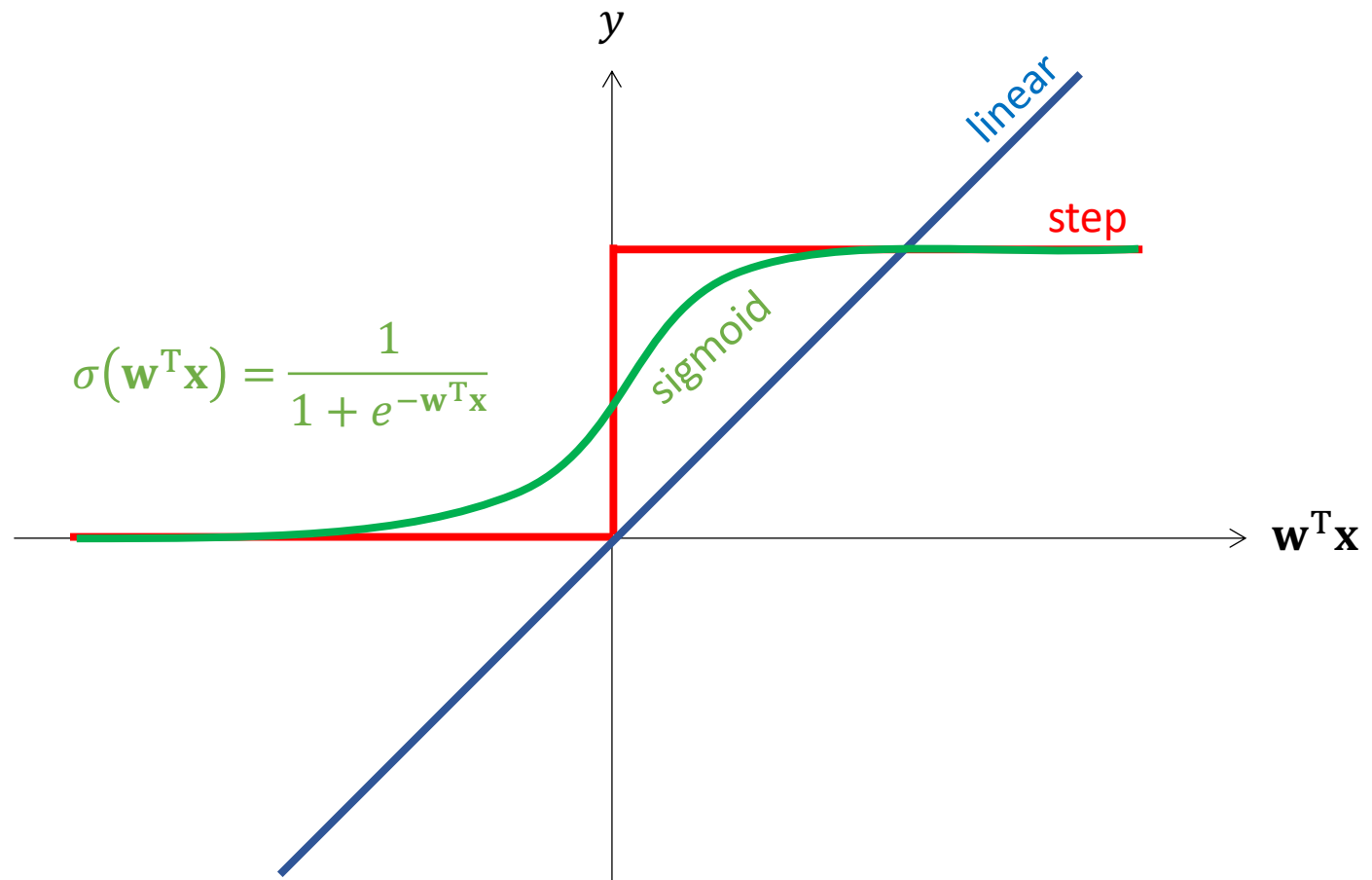
- L_2 -loss (ignore binary nature of y)

$$\hat{L}(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^n (\mathbf{w}^T \mathbf{x}_i - y_i)^2$$



Sensitive to outliers!

Logistic function a.k.a. sigmoid



MLE interpretation of logistic regression

- Assume $p_{\mathbf{w}}(y|\mathbf{x}) = \text{Bernoulli}(p)$, i.e. given \mathbf{x}
 $y = 1$ with probability p $y = 0$ with probability $1 - p$
- Linear model applied to **log-odds** (logit)

$$\log\left(\frac{p}{1-p}\right) = \mathbf{w}^T \mathbf{x}$$

$$\frac{p}{1-p} = e^{\mathbf{w}^T \mathbf{x}}$$

$$p = \frac{e^{\mathbf{w}^T \mathbf{x}}}{1 + e^{\mathbf{w}^T \mathbf{x}}} = \frac{1}{1 + e^{-\mathbf{w}^T \mathbf{x}}} = \sigma(\mathbf{w}^T \mathbf{x}) = p_{\mathbf{w}}(y = 1|\mathbf{x})$$

$$\text{and } p_{\mathbf{w}}(y = 0|\mathbf{x}) = 1 - \sigma(\mathbf{w}^T \mathbf{x})$$

MLE interpretation of logistic regression

- Assume $p_{\mathbf{w}}(y|\mathbf{x}) = \text{Bernoulli}(p)$, i.e. given \mathbf{x}
 $y = 1$ with probability p $y = 0$ with probability $1 - p$
- Log-likelihood

$$\begin{aligned}\hat{L}(\mathbf{w}) &= -\frac{1}{n} \sum_{i=1}^n \log p_{\mathbf{w}}(y|\mathbf{x}_i) \\ &= -\frac{1}{n} \sum_{i:y_i=1} \log \sigma(\mathbf{w}^T \mathbf{x}_i) - \frac{1}{n} \sum_{i:y_i=0} \log (1 - \sigma(\mathbf{w}^T \mathbf{x}_i)) \\ &= -\frac{1}{n} \sum_{i=1}^n y_i \log \sigma(\mathbf{w}^T \mathbf{x}_i) + (1 - y_i) \log (1 - \sigma(\mathbf{w}^T \mathbf{x}_i))\end{aligned}$$

MLE interpretation of logistic regression

- Assume $p_{\mathbf{w}}(y|\mathbf{x}) = \text{Bernoulli}(p)$, i.e. given \mathbf{x}
 $y = 1$ with probability p $y = 0$ with probability $1 - p$
- Vectorized version of log-likelihood

$$\hat{L}(\mathbf{w}) = -\frac{1}{n} [\mathbf{y}^T \log \sigma(\mathbf{Xw}) + (\mathbf{1} - \mathbf{y})^T \log(\mathbf{1} - \sigma(\mathbf{Xw}))]$$