Machine Learning

3 – Linear & Logistic Regression

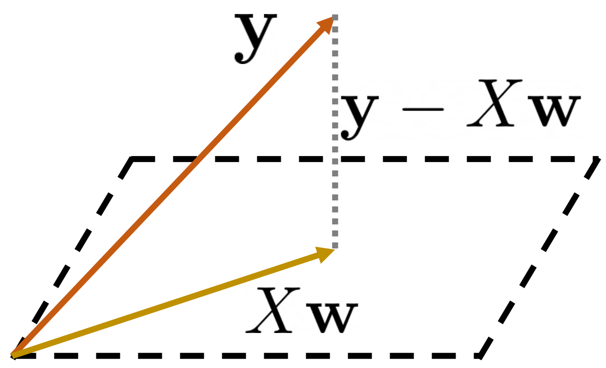
Avisek Gupta
Postdoctoral Fellow, IAI, TCG CREST
avisek003@gmail.com

August 19, 2022

Solving Multiple Linear Regression

$$\begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1d} \\ 1 & x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nd} \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_d \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

$$X\mathbf{w} = \mathbf{y}$$



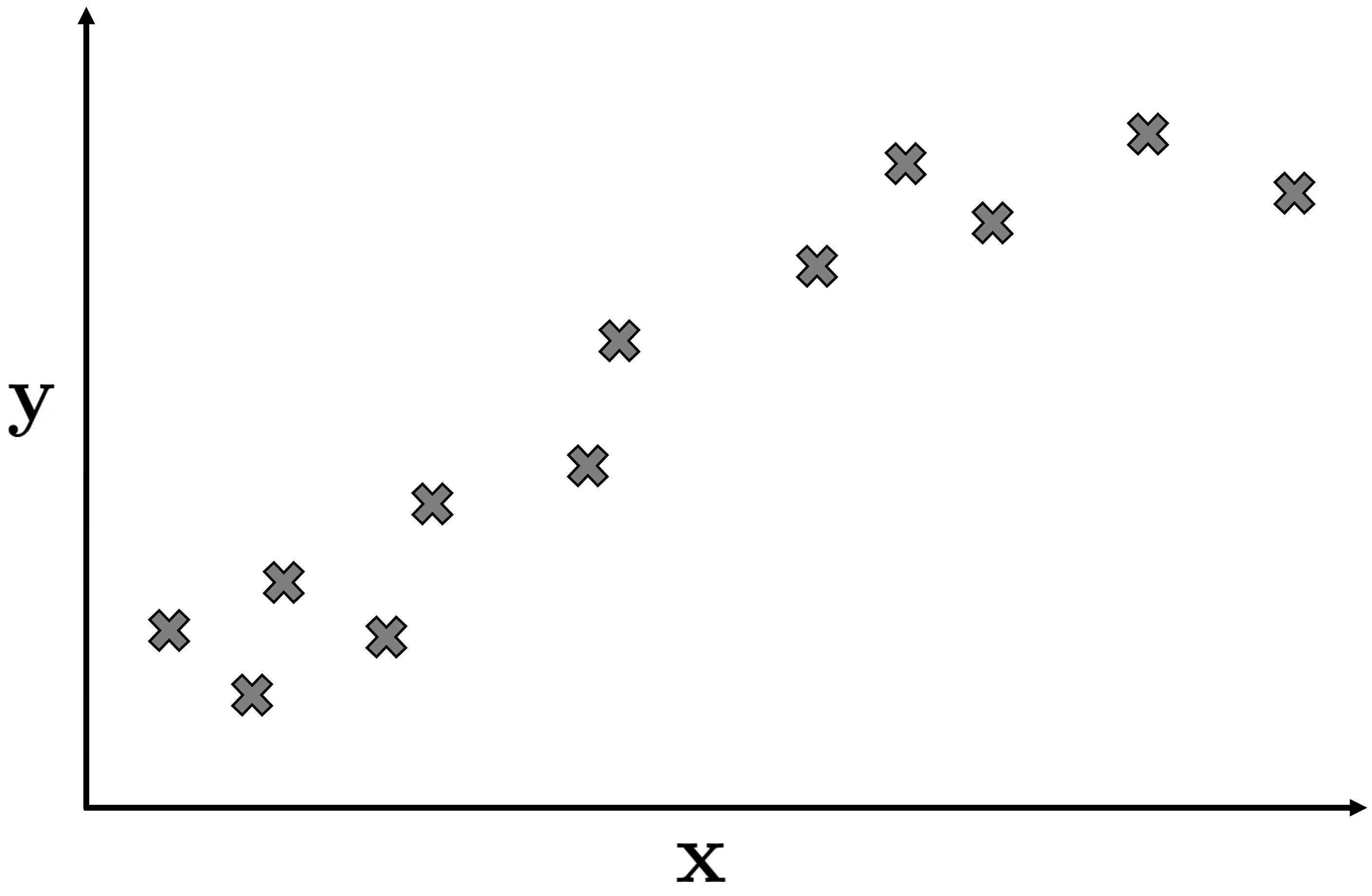
$$X^T(\mathbf{y} - X\mathbf{w}) = \mathbf{0}$$

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

Estimated w

Linear Regression

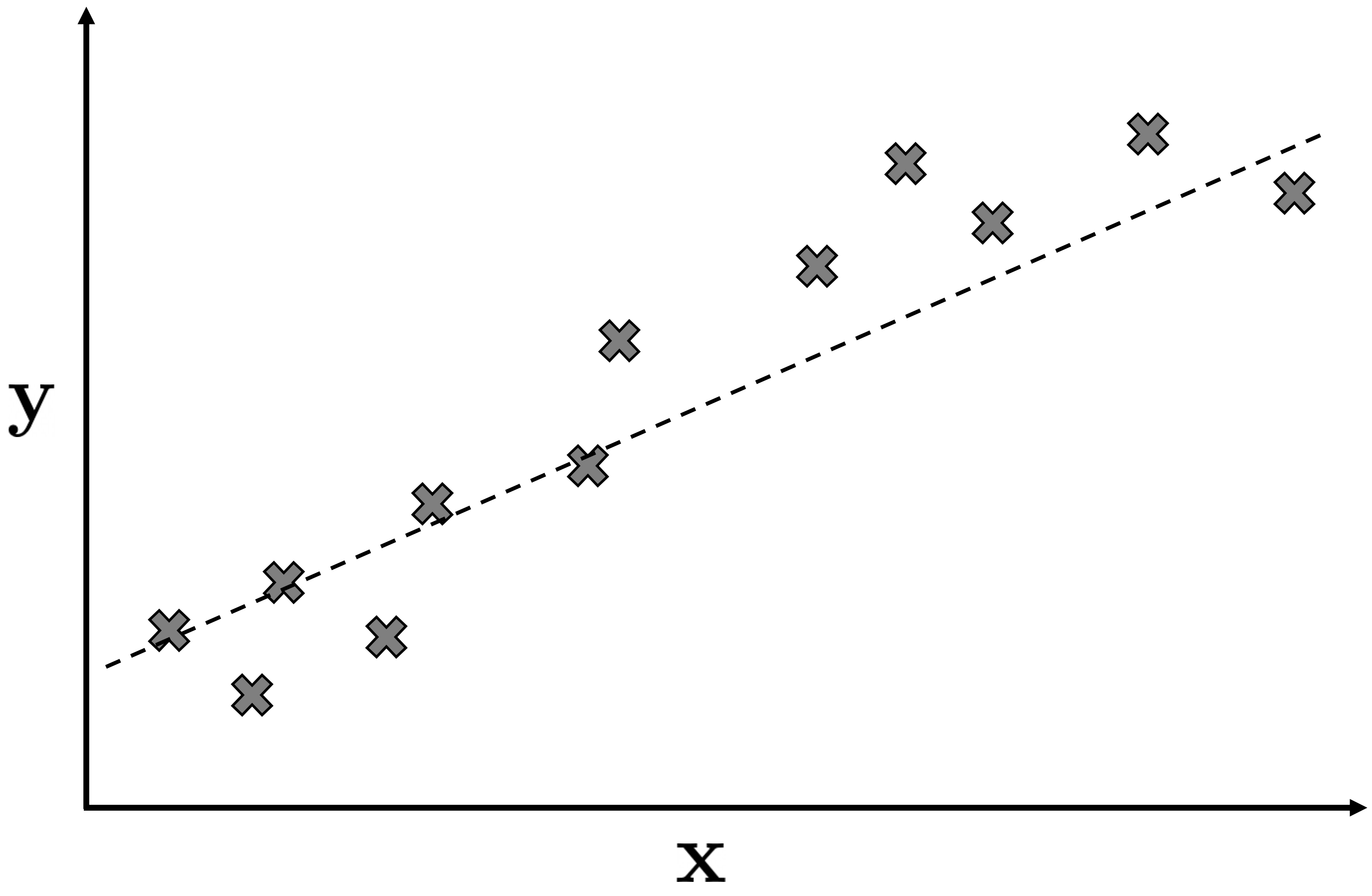
Given data $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}, \ x_i, y_i \in \mathbb{R},$



Linear Regression

Given data $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}, \; x_i, y_i \in \mathbb{R},$

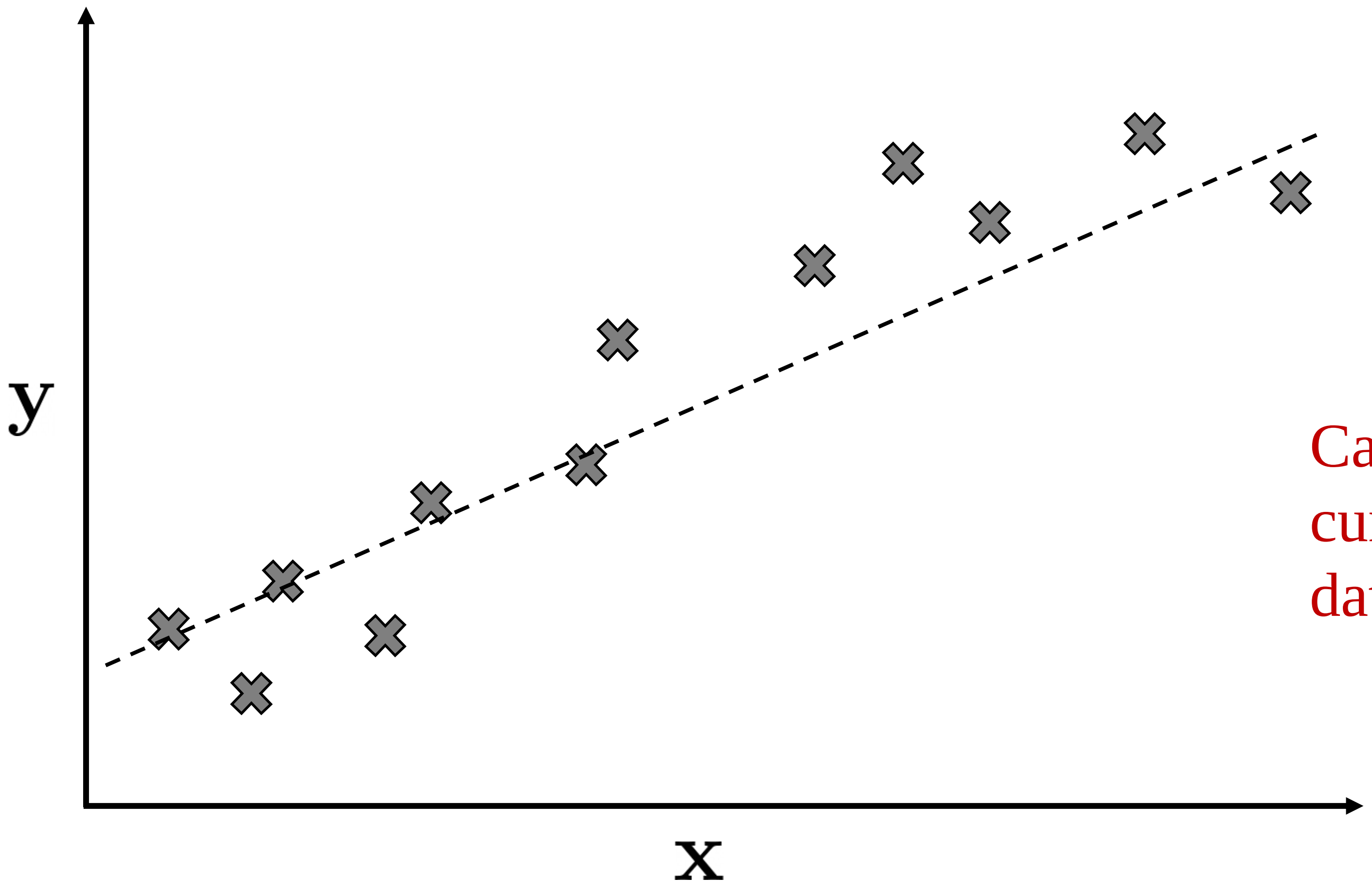
estimate $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$



Linear Regression

Given data $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}, \; x_i, y_i \in \mathbb{R},$

estimate $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$

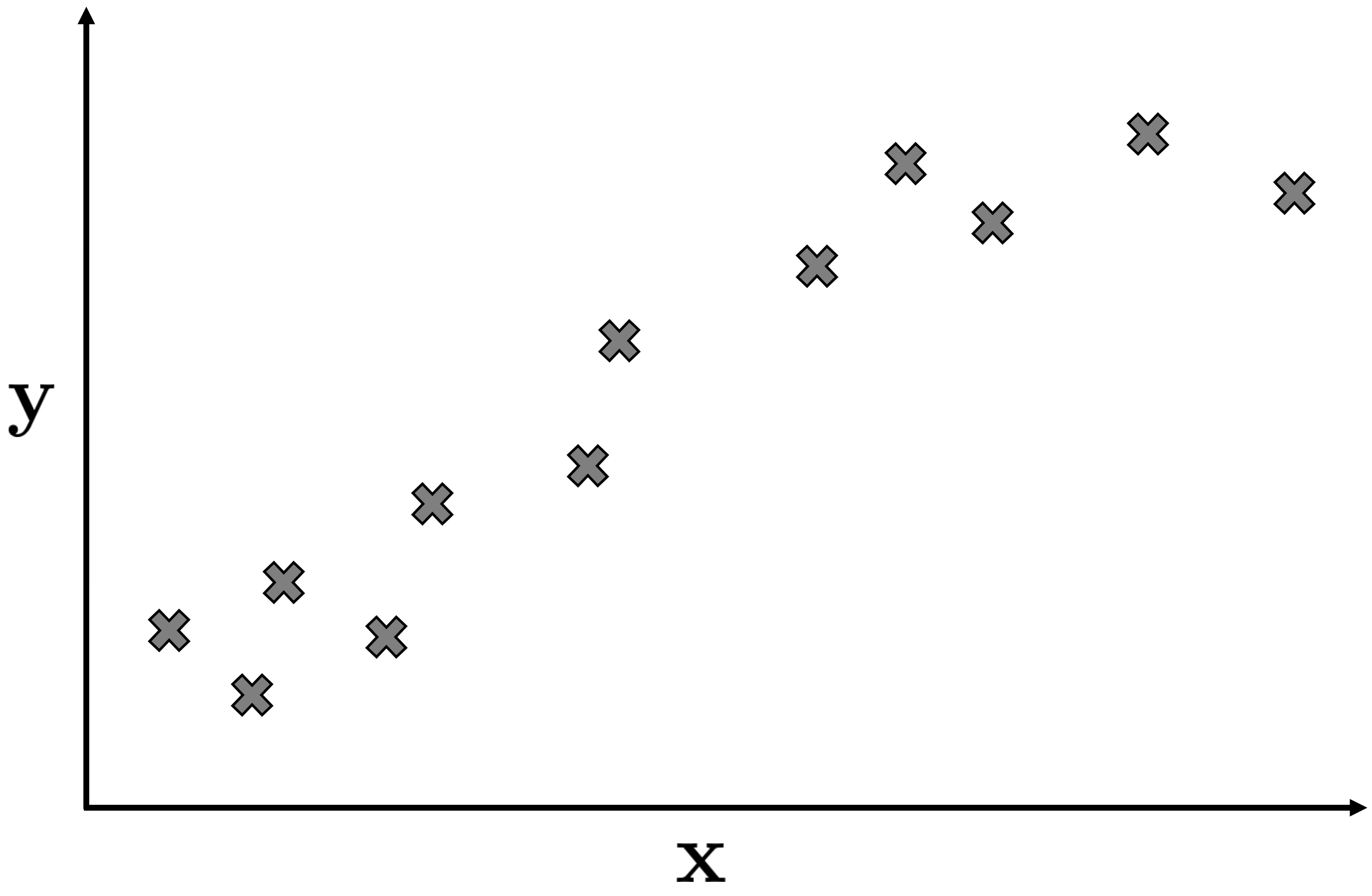


Can we estimate a curve that fits the data better?

Polynomial Regression

Given data $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\},\ x_i, y_i \in \mathbb{R},$

estimate $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i + \hat{\beta}_2 x_i^2 + \hat{\beta}_3 x_i^3 + ... + \hat{\beta}_d x_i^d$

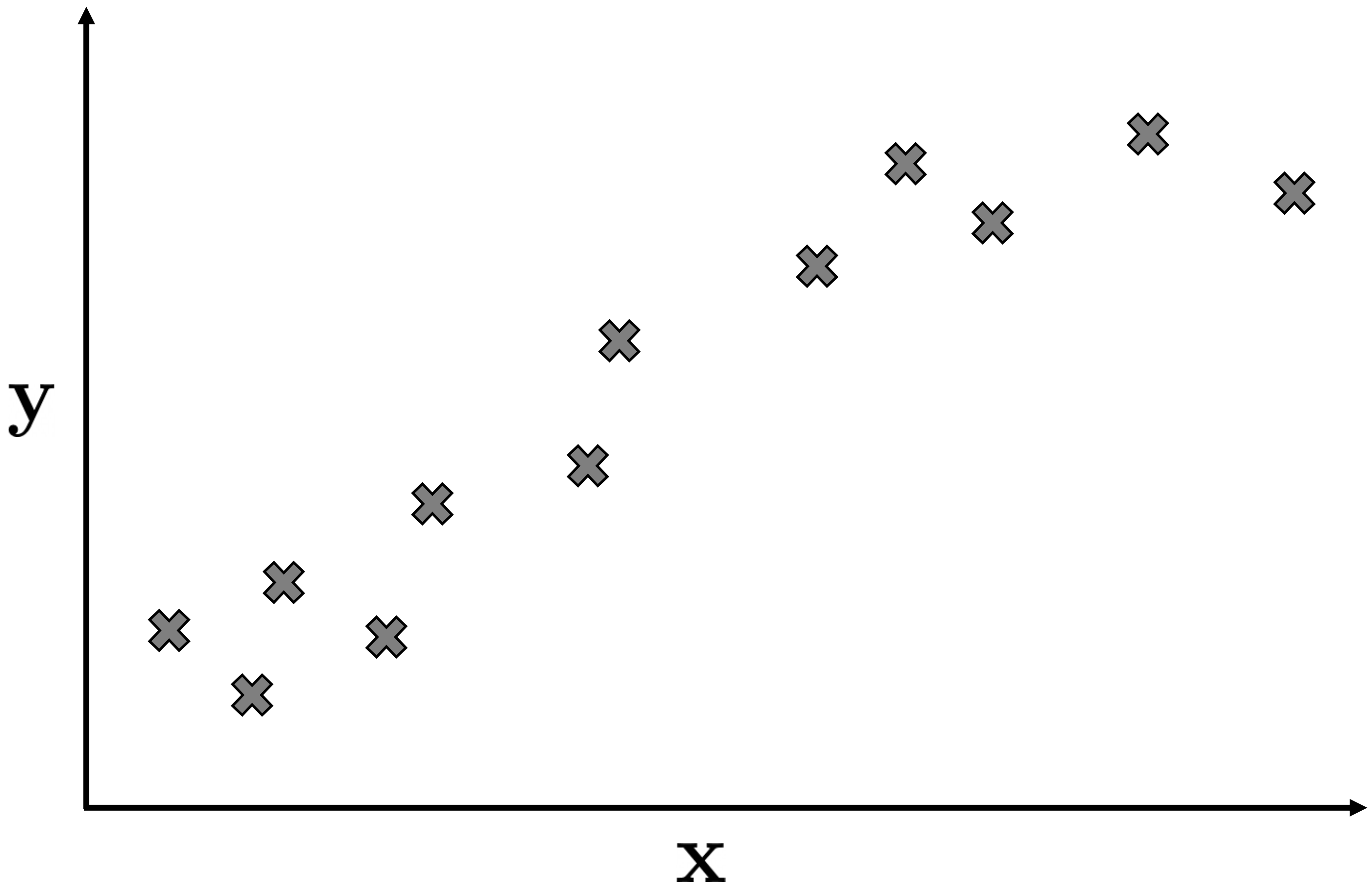


Polynomial Regression

Given data $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\},\ x_i, y_i \in \mathbb{R},$

estimate $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i + \hat{\beta}_2 x_i^2 + \hat{\beta}_3 x_i^3 + \ldots + \hat{\beta}_d x_i^d$

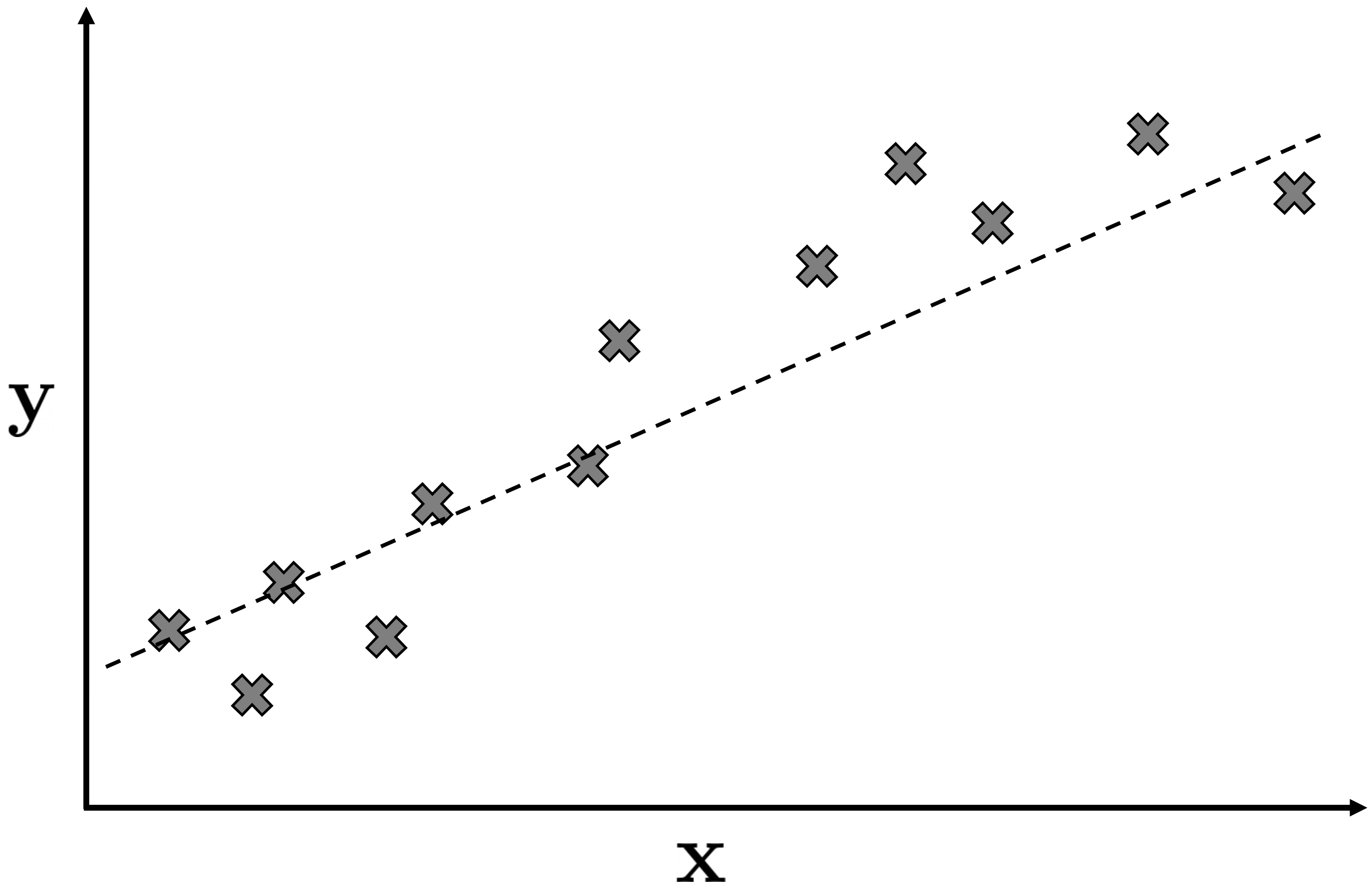
$d = ?$



Polynomial Regression

Given data $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}, \; x_i, y_i \in \mathbb{R},$

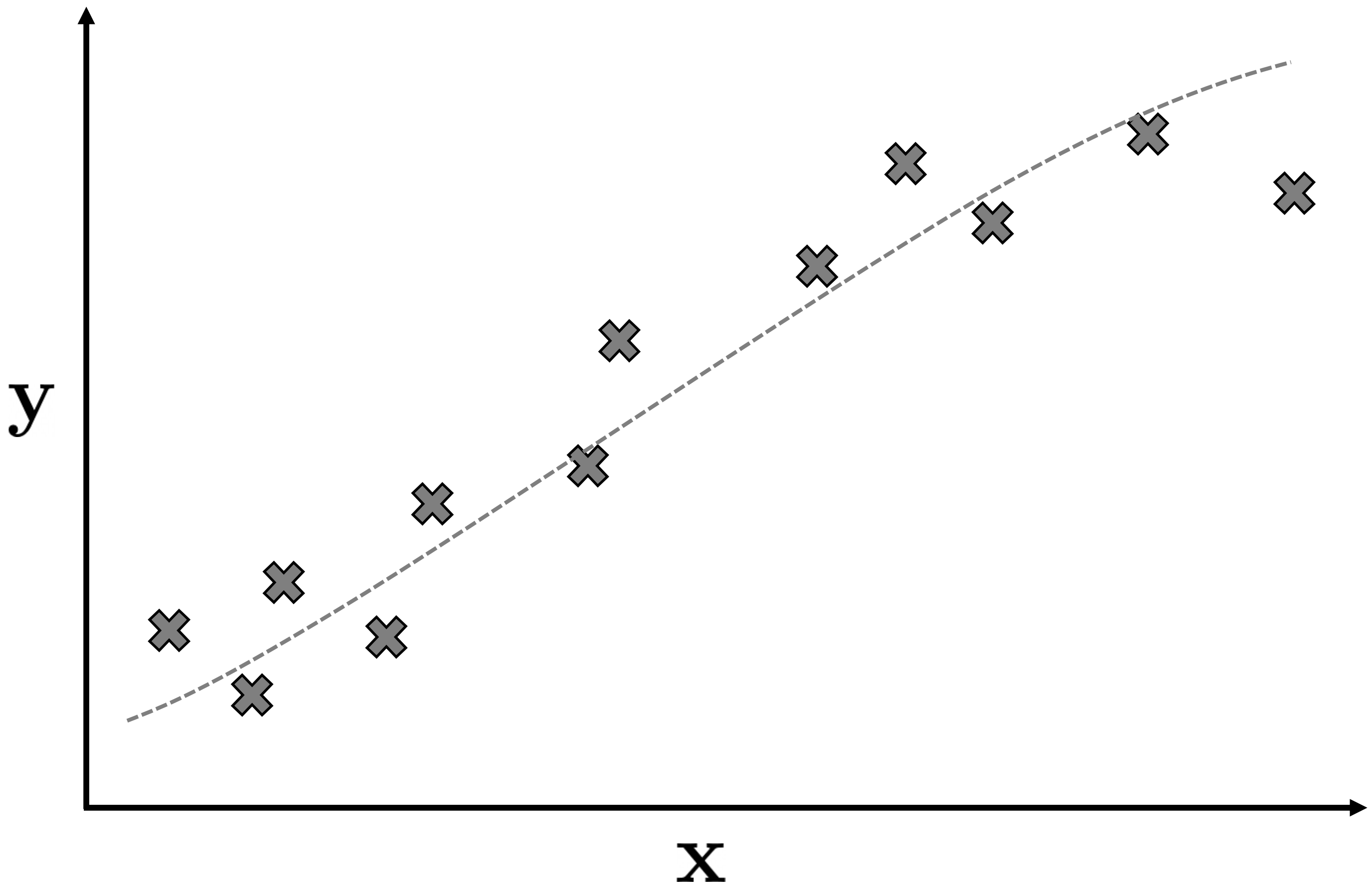
estimate $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i \qquad (d = 1)$



Polynomial Regression

Given data $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}, \; x_i, y_i \in \mathbb{R},$

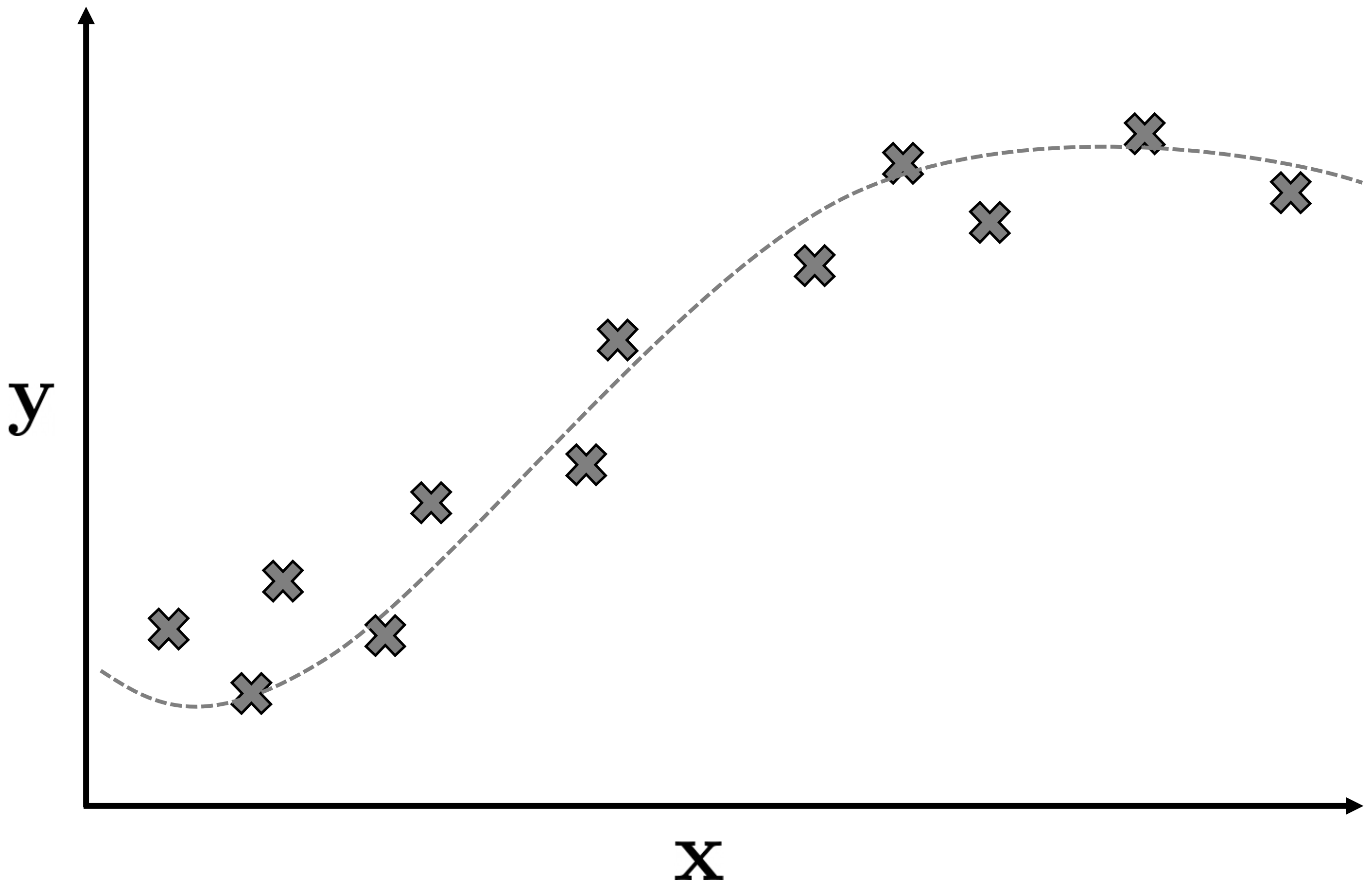
estimate $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i + \hat{\beta}_2 x_i^2 \qquad (d = 2)$



Polynomial Regression

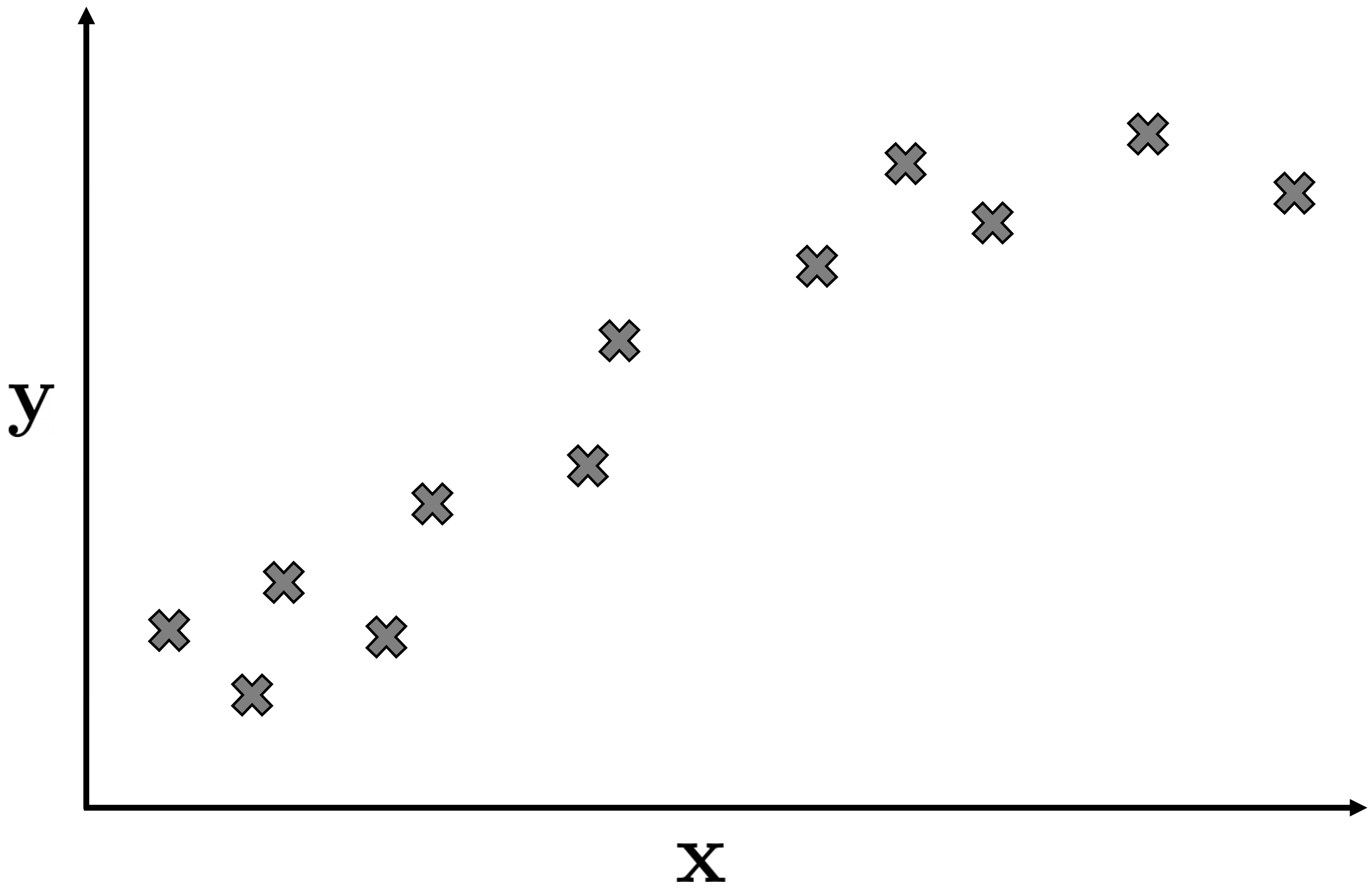
Given data $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\},\ x_i, y_i \in \mathbb{R},$

estimate $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i + \hat{\beta}_2 x_i^2 + \hat{\beta}_3 x_i^3 \qquad (d = 3)$



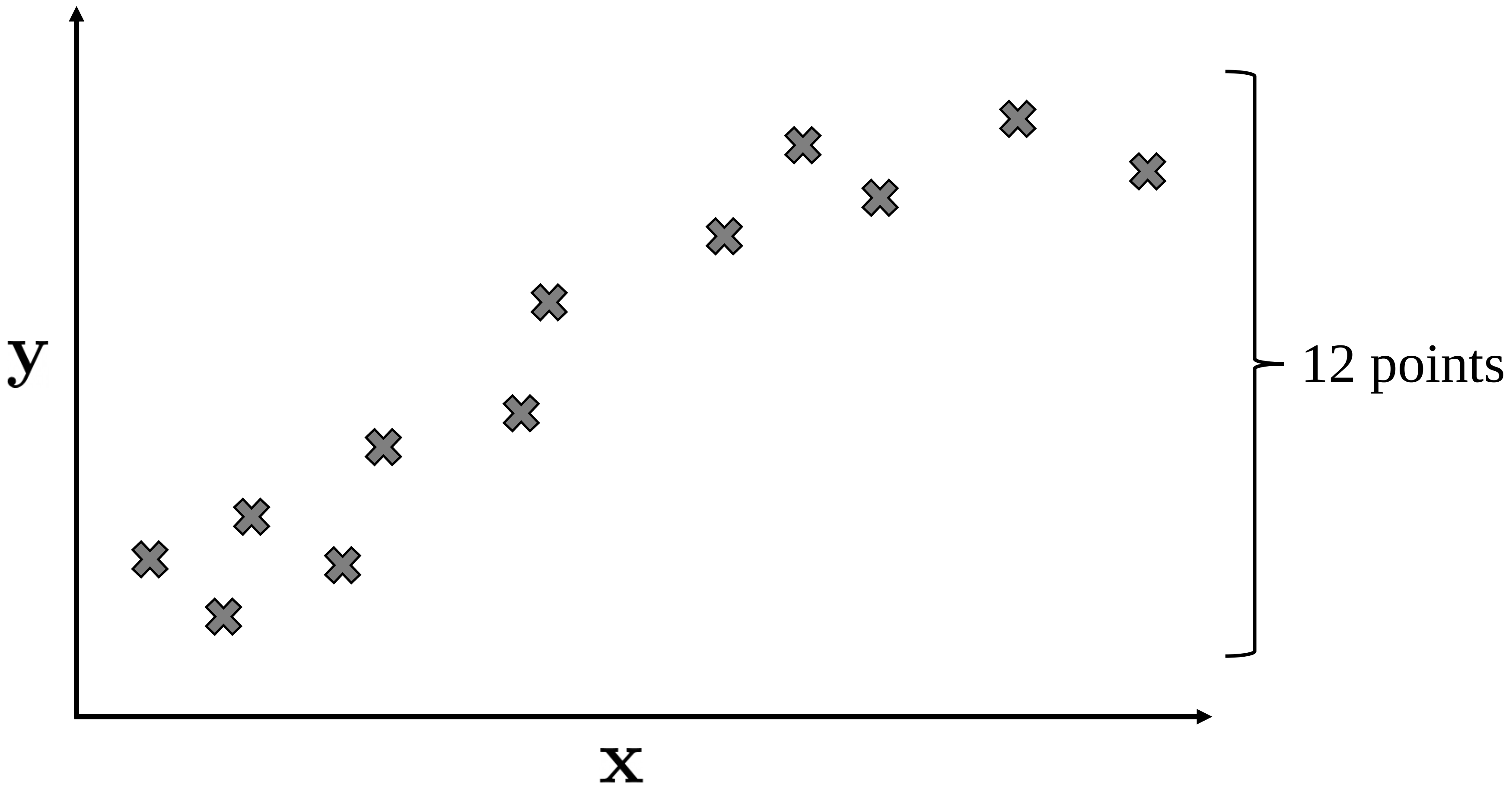
Polynomial Regression

Given data $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}, \ x_i, y_i \in \mathbb{R},$



Polynomial Regression

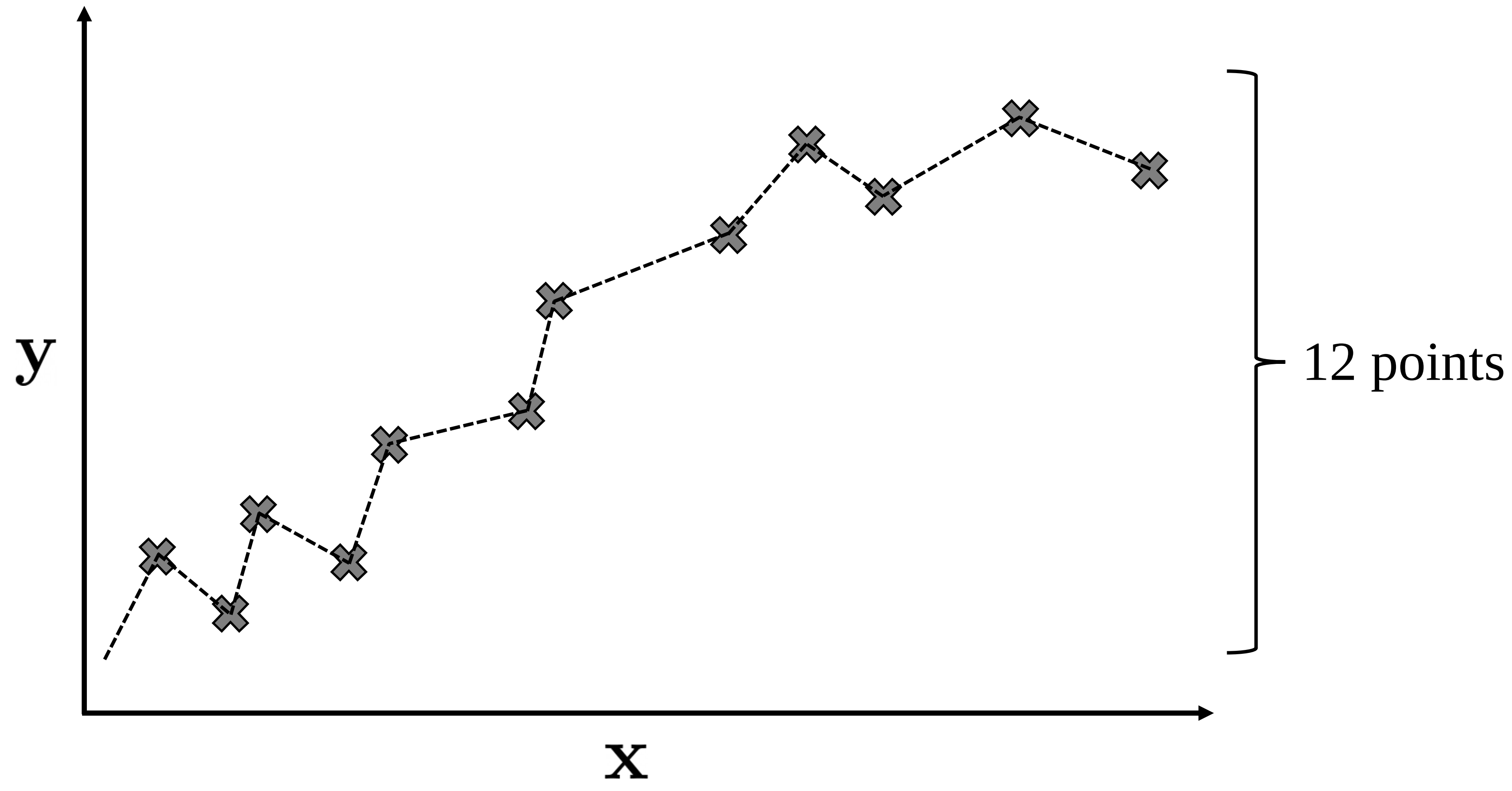
Given data $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\},\ x_i, y_i \in \mathbb{R},$



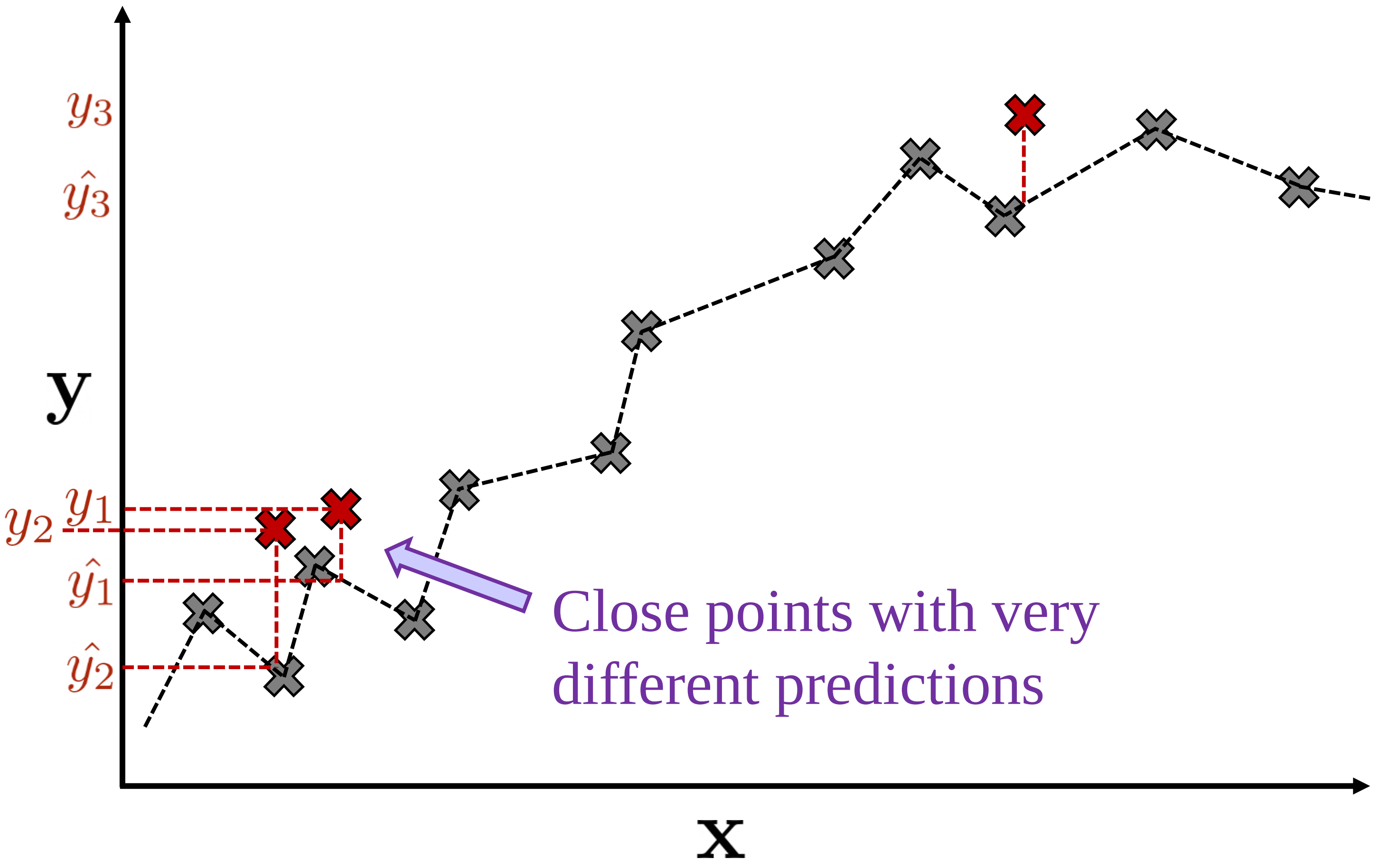
Polynomial Regression

Given data $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}, \; x_i, y_i \in \mathbb{R},$

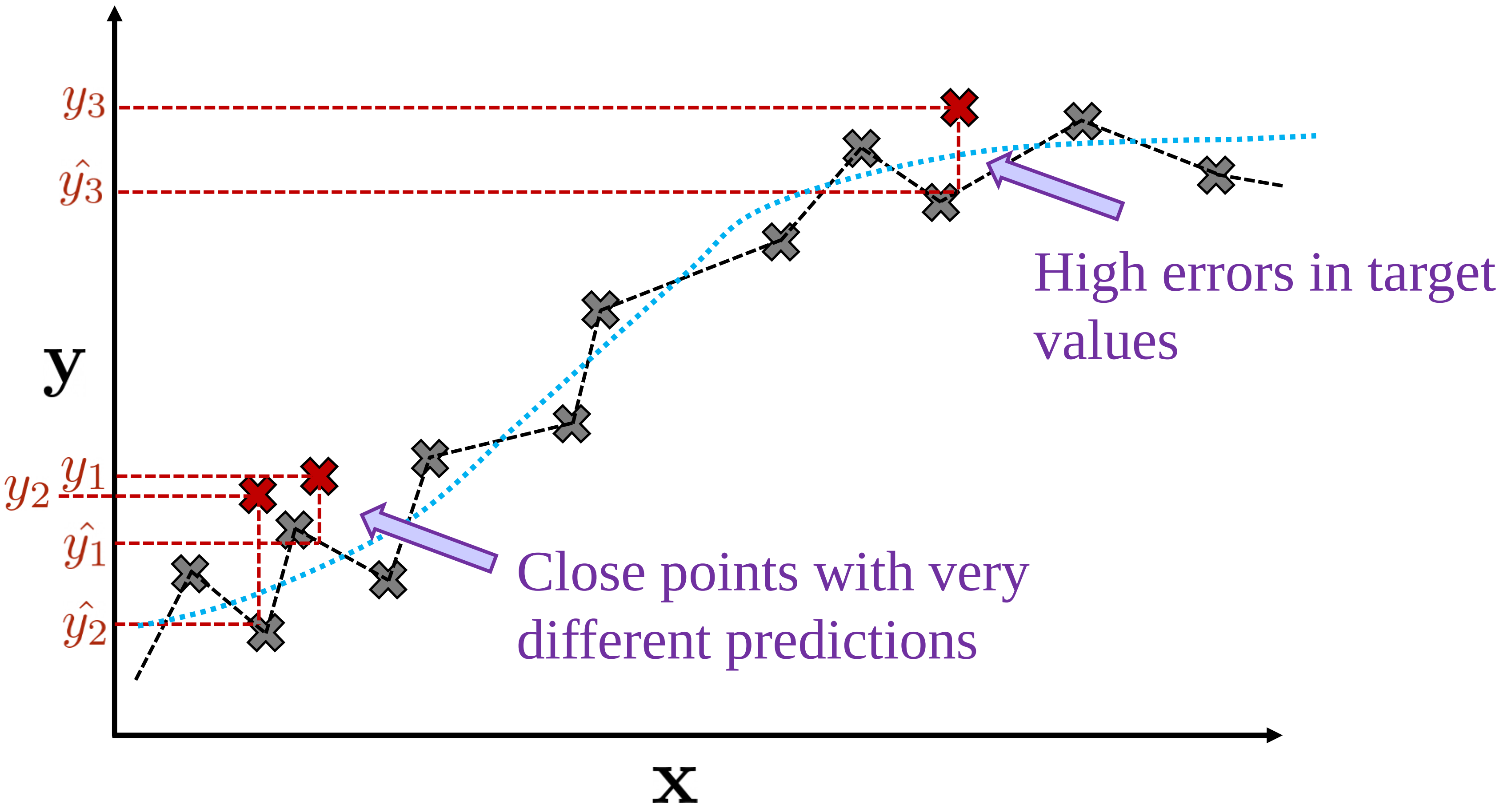
estimate $\hat{y_i} = \hat{\beta}_0 + \hat{\beta}_1 x_i + \hat{\beta}_2 x_i^2 + \hat{\beta}_3 x_i^3 + ... + \hat{\beta_{11}} x_i^{11}$



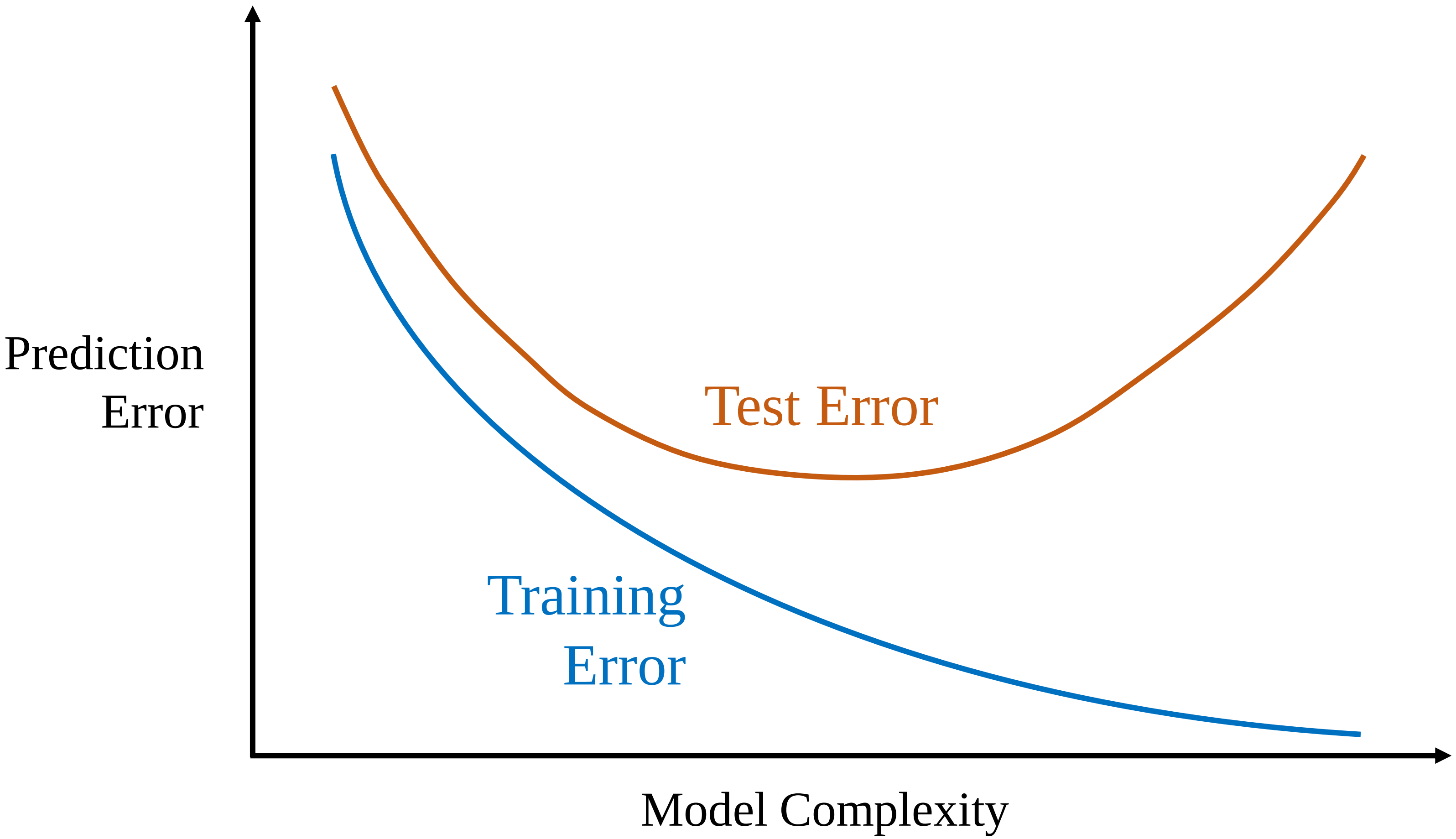
Complex Models on Unseen data



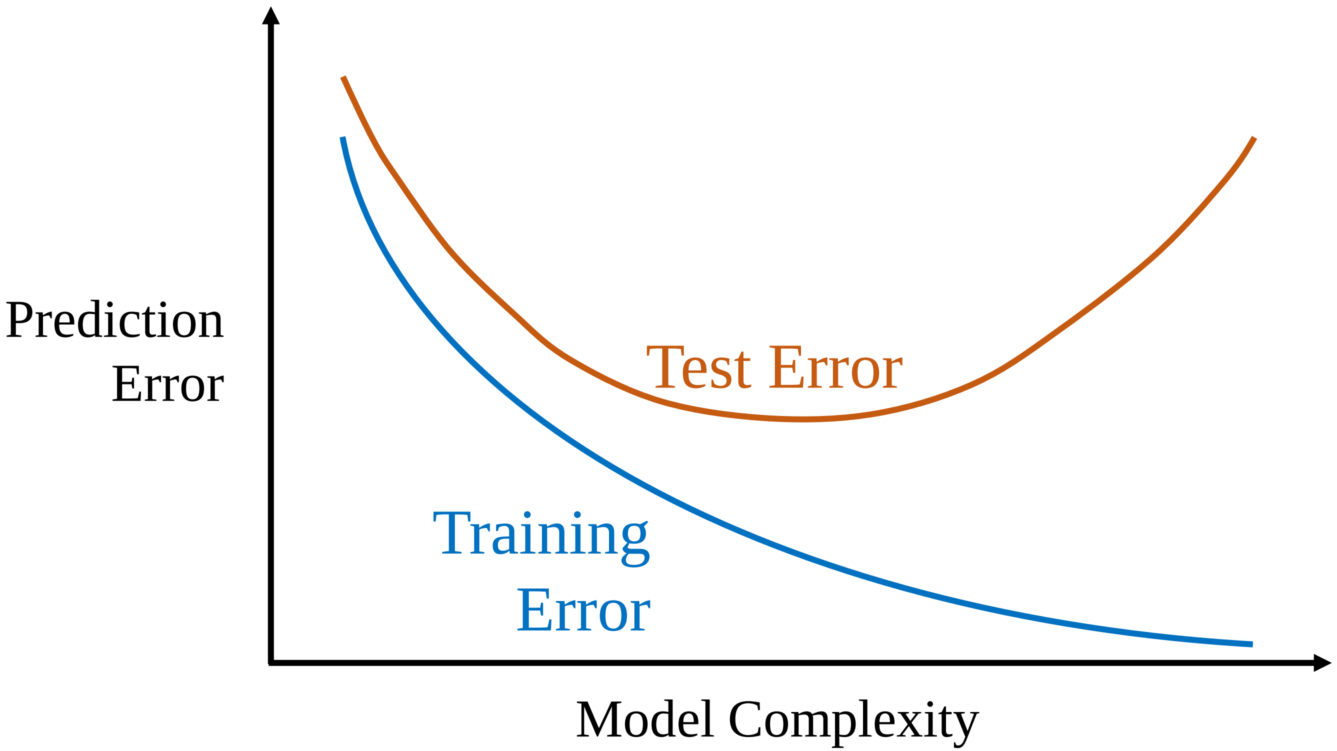
Complex Models on Unseen data

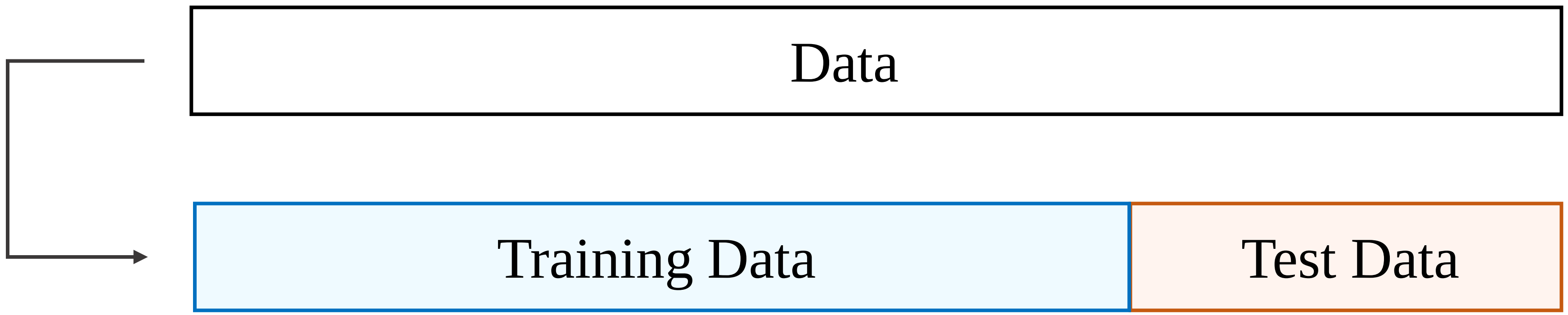


Trade-off: Model Complexity vs. Generalization Capability

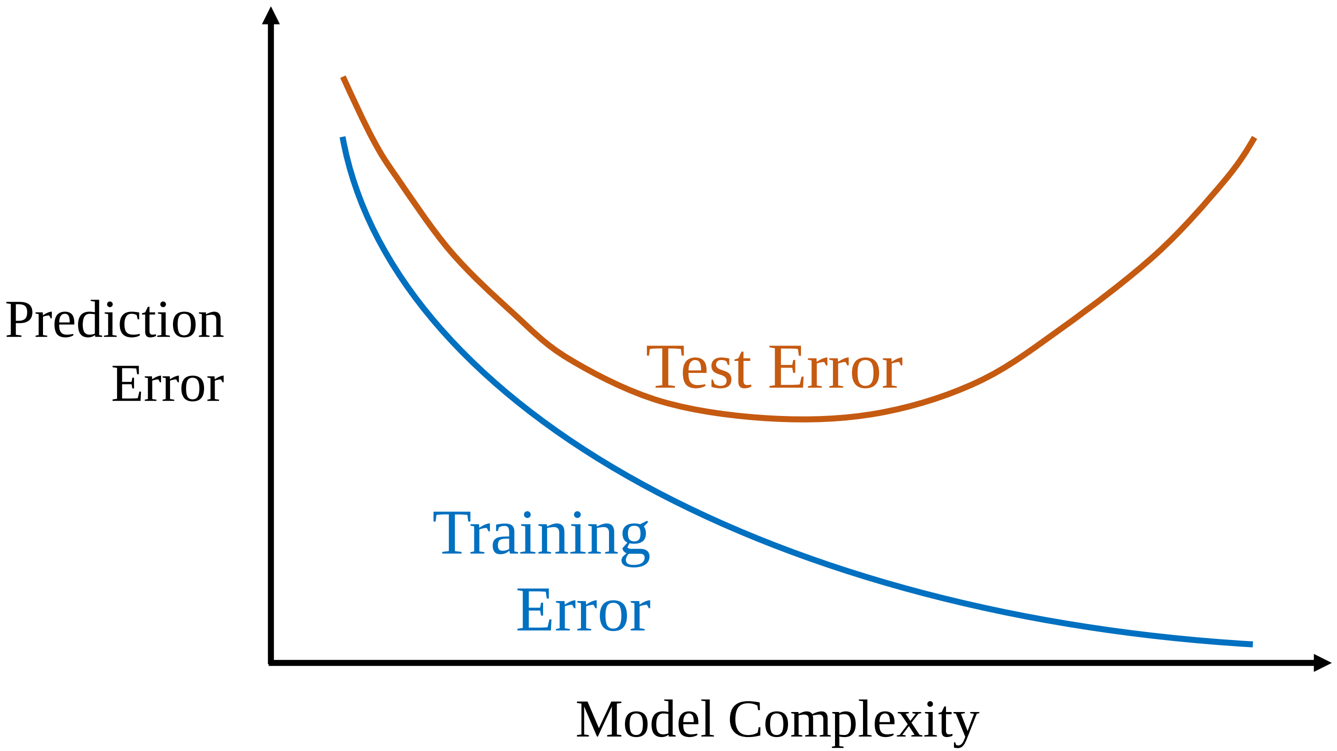


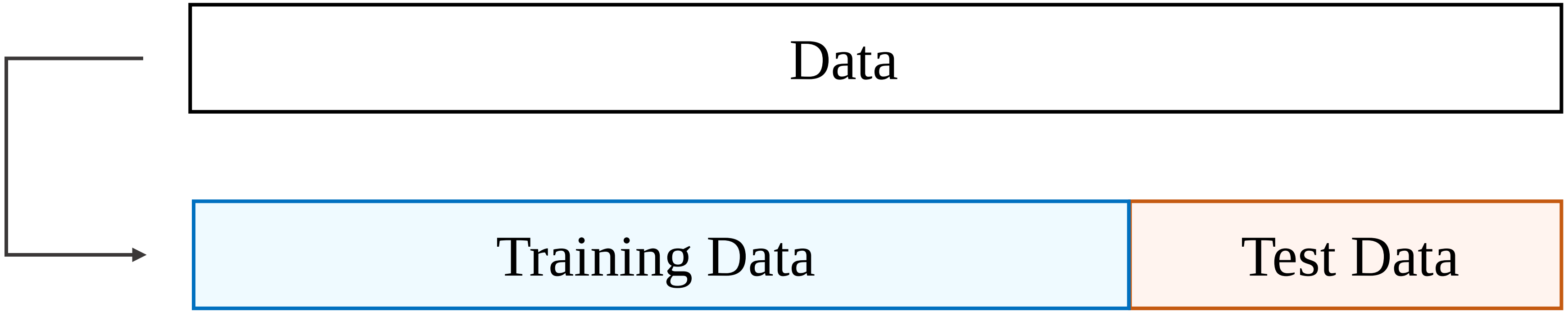
Trade-off: Model Complexity vs. Generalization Capability





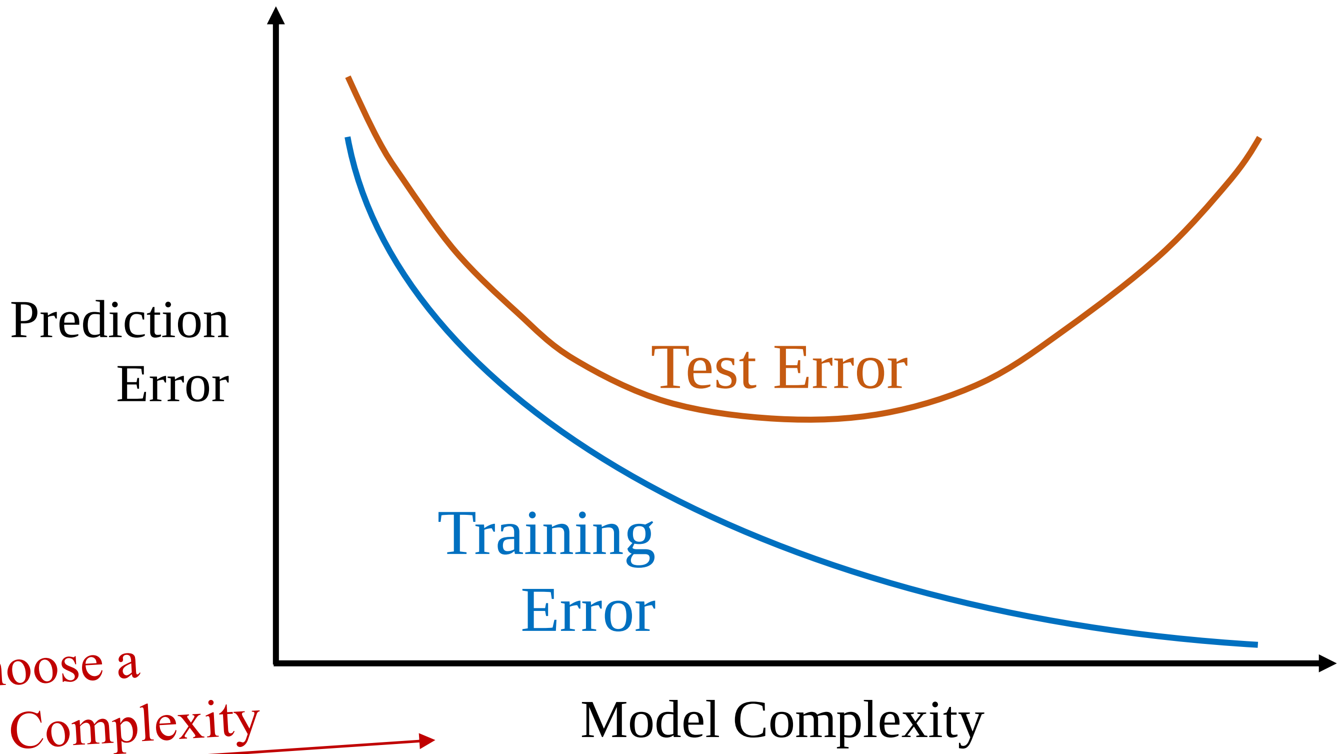
Trade-off: Model Complexity vs. Generalization Capability



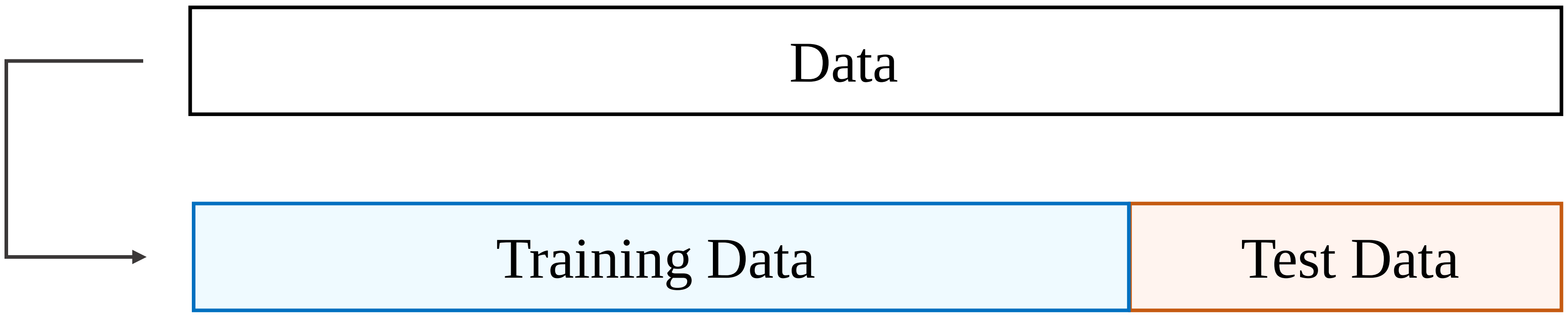


Accuracy on Test Data is an estimate of the general accuracy

Trade-off: Model Complexity vs. Generalization Capability

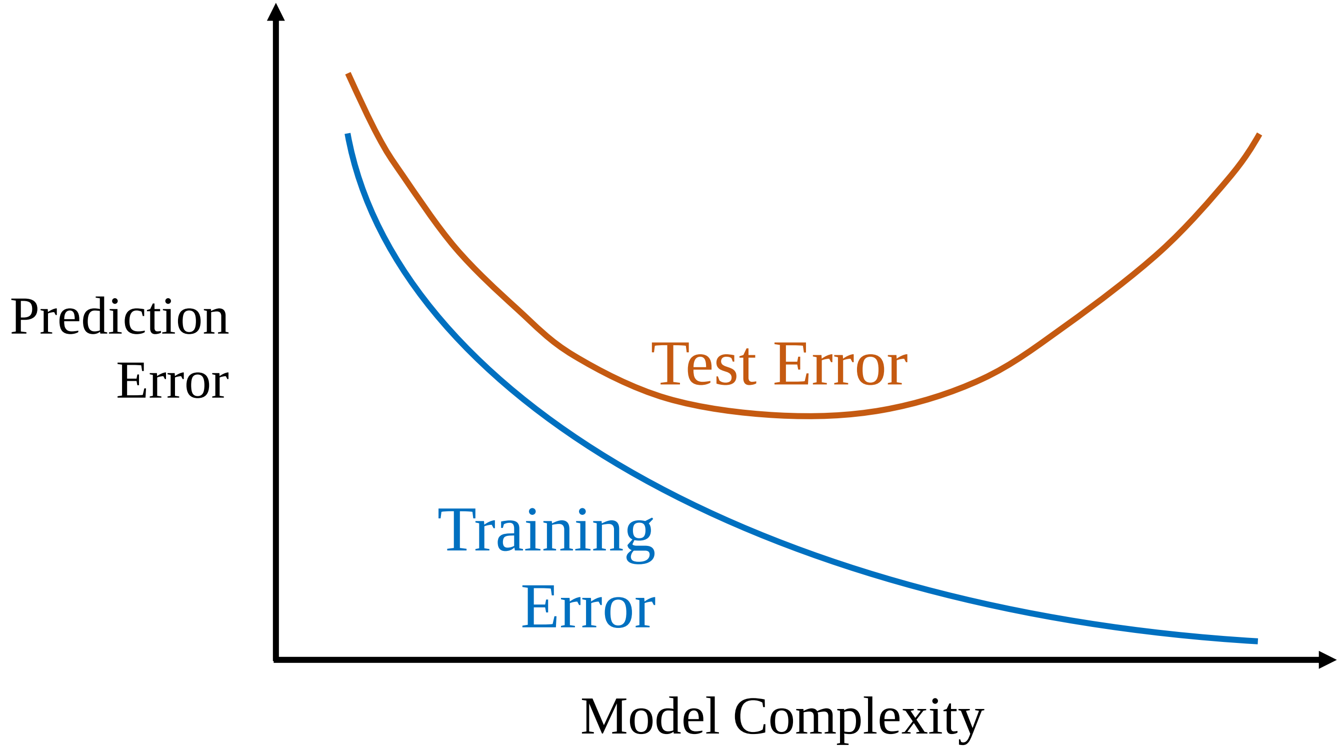


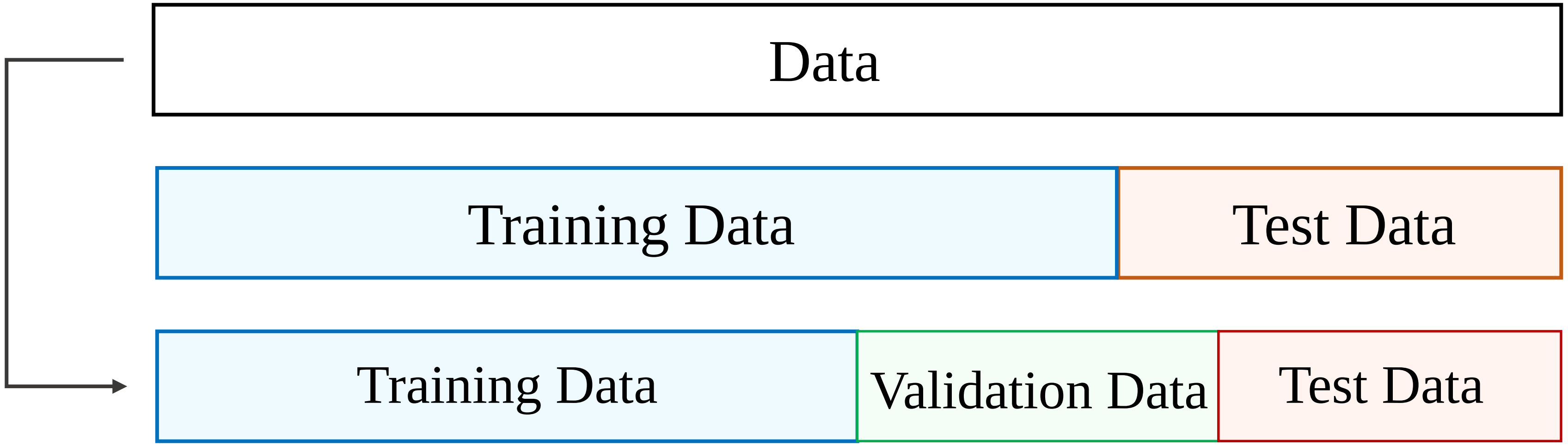
How can we choose a suitable Model Complexity



Accuracy on Test Data is an estimate of the general accuracy

Trade-off: Model Complexity vs. Generalization Capability





For all models –

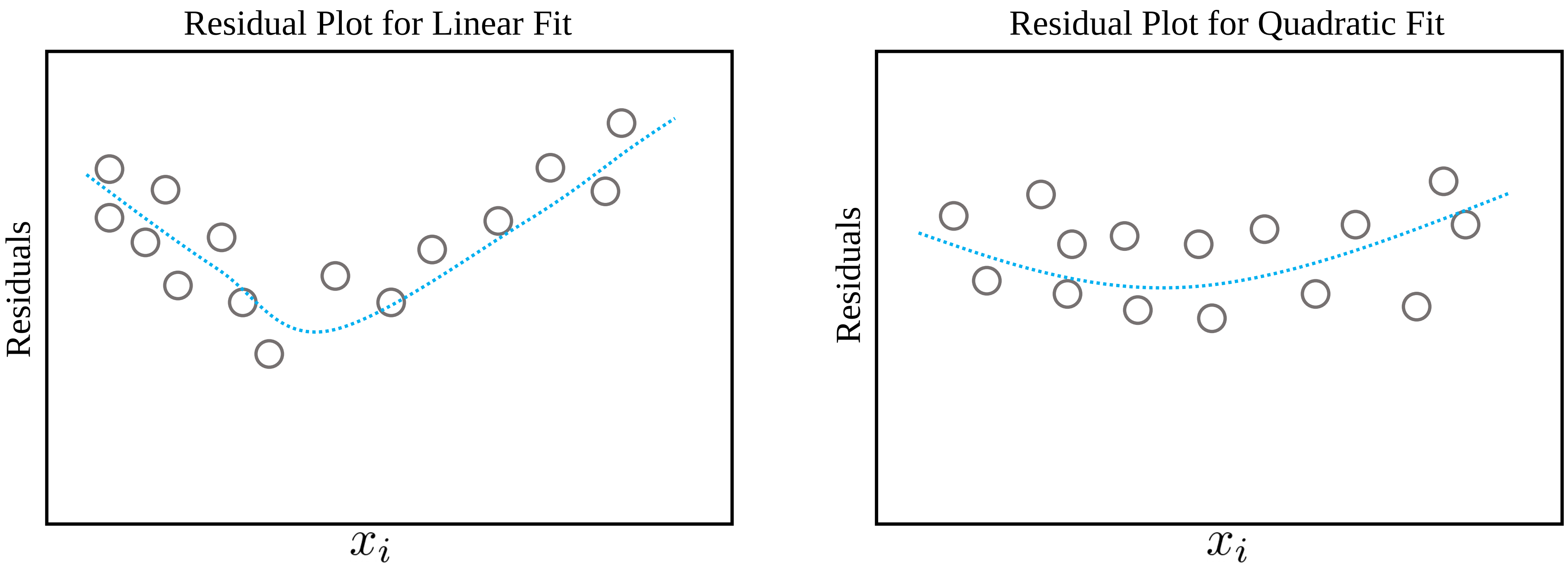
1. Train on the Training Data
2. Compare on the Validation Data
3. Obtain estimate of general accuracy on Test data

Residual Plots

Plots the relationship between the residuals $e_i = y_i - \hat{y}_i$ and a variable x_i .

Uses of Residual Plots:

1. Identify non-linearity of variable-target relationships

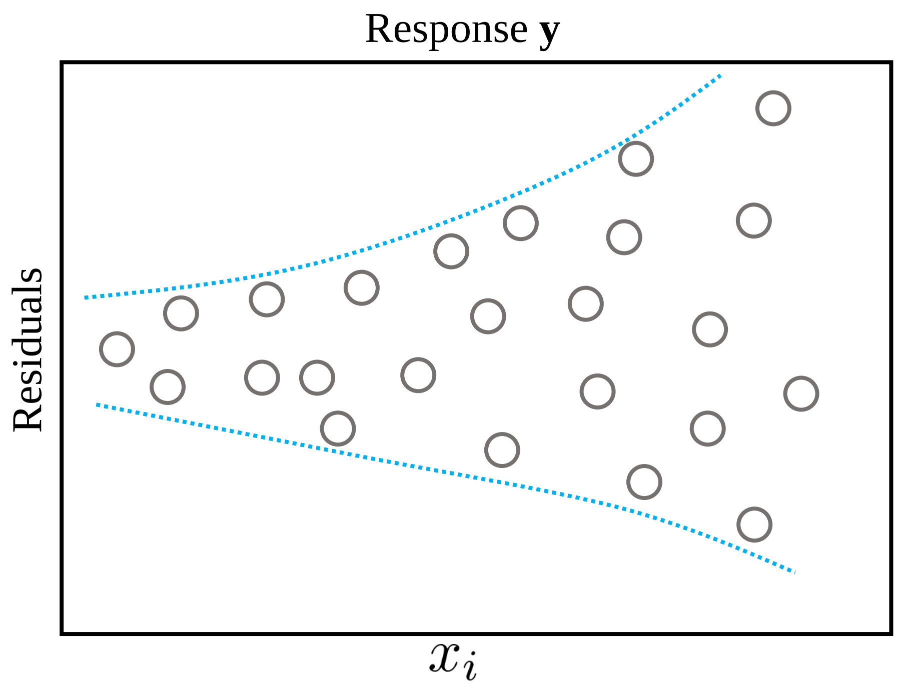


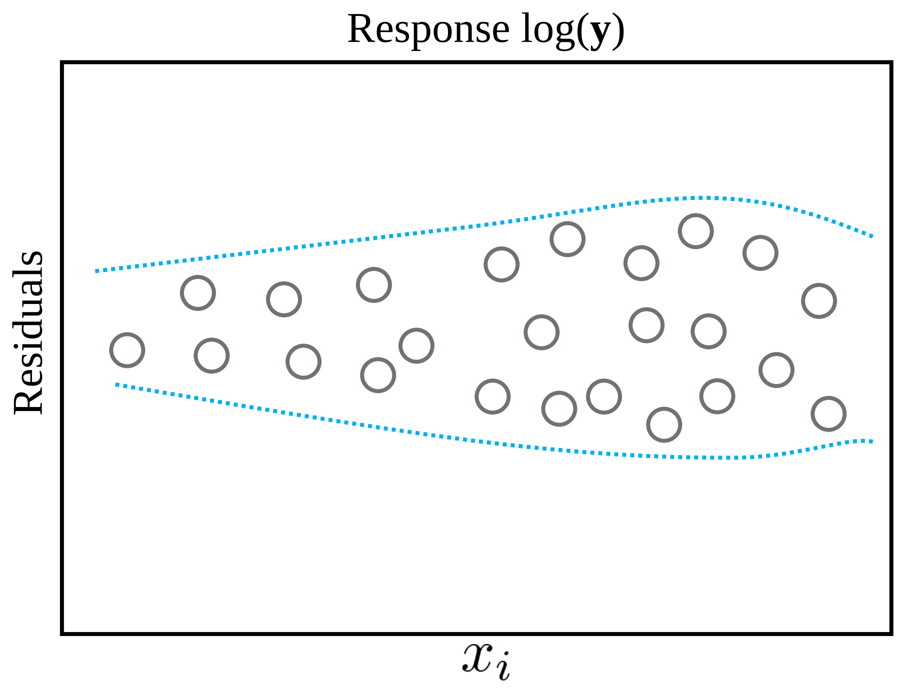
Residual Plots

Plots the relationship between the residuals $e_i = y_i - \hat{y}_i$ and a variable x_i .

Uses of Residual Plots:

2. Non-constant variance of error terms (heteroscedasticity)





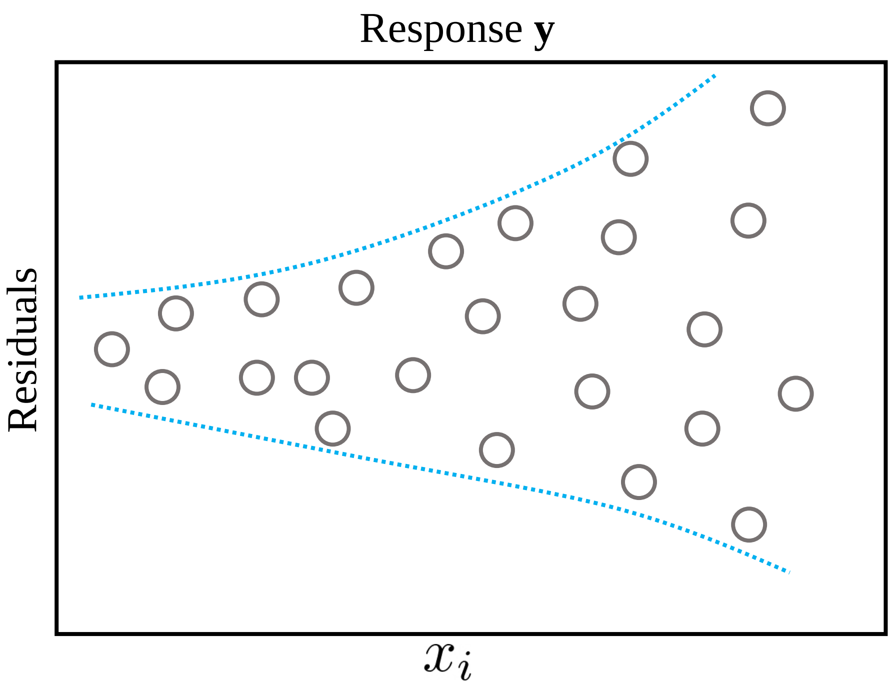
Residual Plots

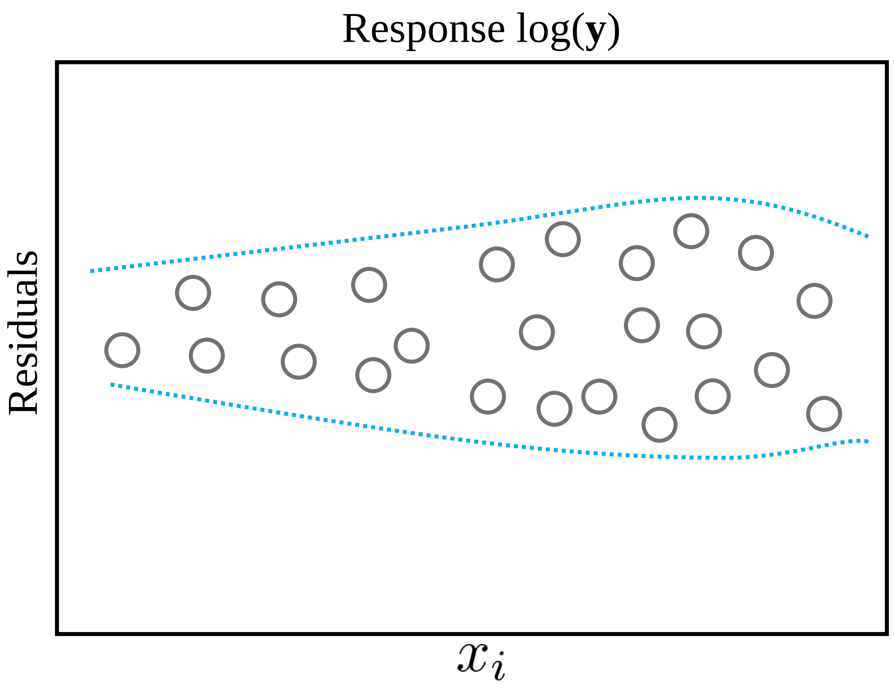
Plots the relationship between the residuals $e_i = y_i - \hat{y}_i$ and a variable x_i .

Uses of Residual Plots:

2. Non-constant variance of error terms (heteroscedasticity)

1. Transform the response using a concave function
2. Weight the responses



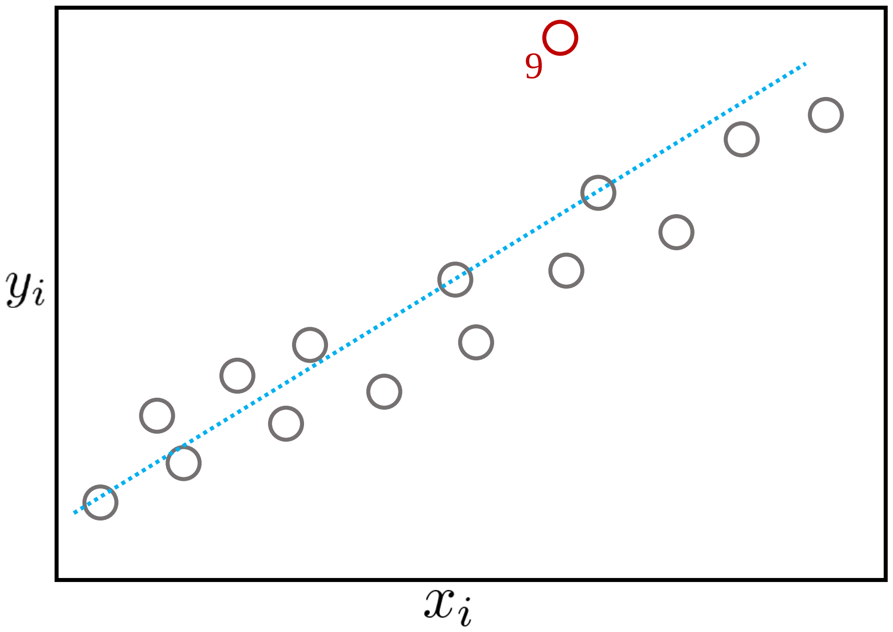


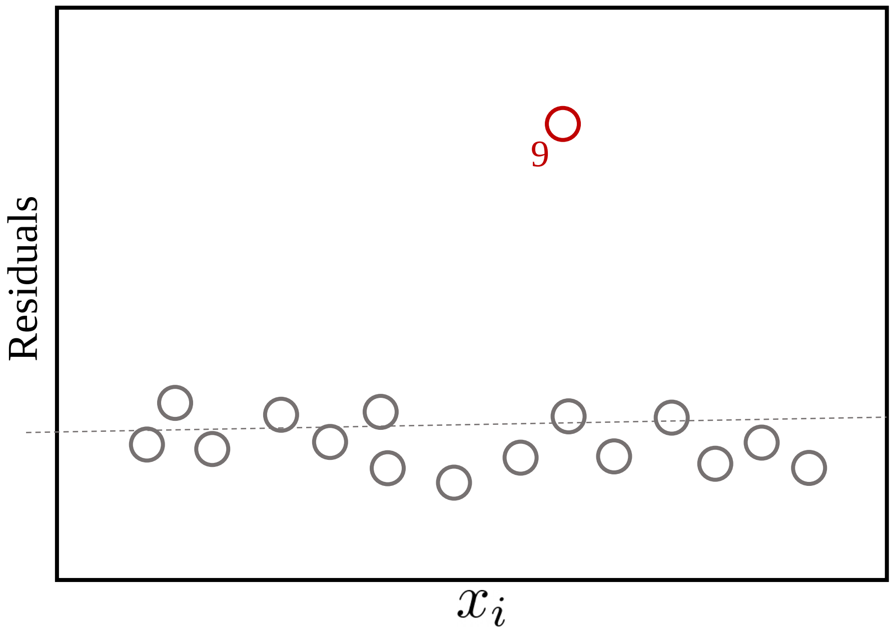
Residual Plots

Plots the relationship between the residuals $e_i = y_i - \hat{y}_i$ and a variable x_i .

Uses of Residual Plots:

3. Identifying outliers





Classification: Logistic Regression

Given: $X = \{\mathbf{x_1}, \mathbf{x_2}, ..., \mathbf{x_n}\}, \mathbf{x_i} \in \mathbb{R}^d$, and $Y = \{y_1, y_2, ..., y_n\}, y_i \in \{0, 1, ...K\}$,

for the problem of classification, we wish to estimate a function $\hat{f}(X) = \hat{Y}$, that correctly classifies the data instances.

Binary Classification: $y_i \in \{0, 1\}$

Classification: Logistic Regression

Given: $X = \{\mathbf{x_1}, \mathbf{x_2}, ..., \mathbf{x_n}\}, \mathbf{x_i} \in \mathbb{R}^d$, and $\mathbf{y} = \{y_1, y_2, ..., y_n\}, y_i \in \{0, 1, ..., K\}$,

for the problem of classification, we wish to estimate a function $\hat{f}(X) = \hat{\mathbf{y}}$, that correctly classifies the data instances.

Binary Classification: $y_i \in \{0, 1\}$

Logistic Regression

We estimate the function:

$$\hat{y}_i = g(w_0 + w_1 x_{i1} + w_2 x_{i2} + ... + w_d x_{id})$$

where,

$$g(t) = \frac{1}{1 + exp(-t)}$$

Logistic Regression

We estimate the function:

$$\hat{y}_i = g\left(w_0 + w_1 x_{i1} + w_2 x_{i2} + \ldots + w_d x_{id}\right)$$

where,

$$g(t) = \frac{1}{1 + exp(-t)}$$

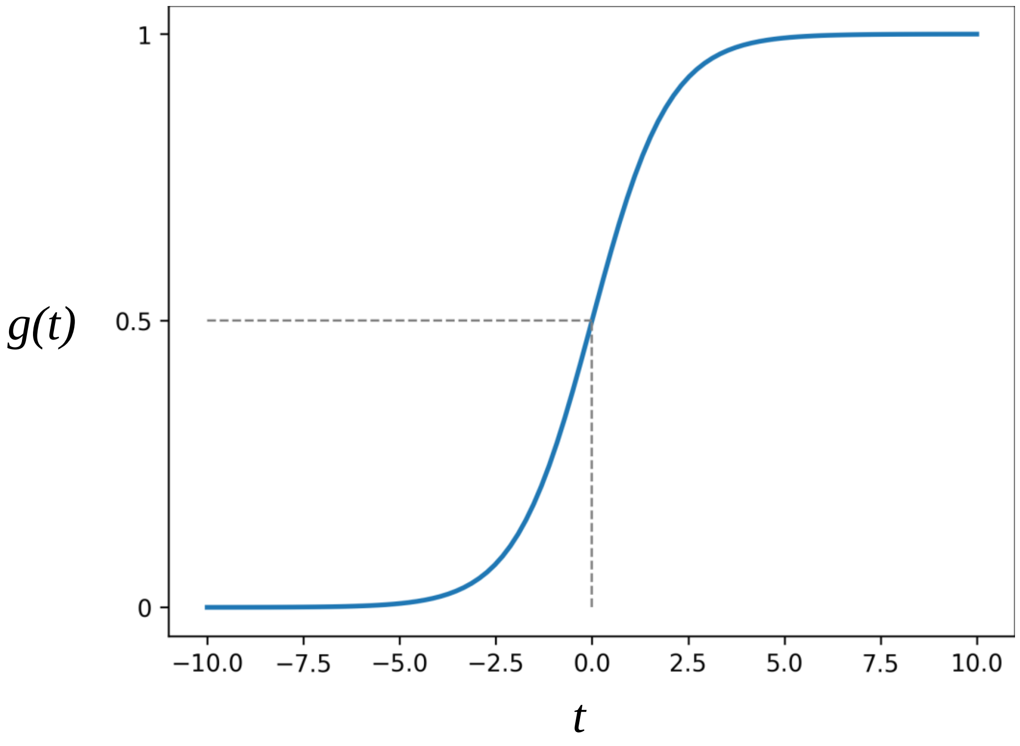
Logistic Regression

We estimate the function:

$$\hat{y}_i = g(w_0 + w_1 x_{i1} + w_2 x_{i2} + \dots + w_d x_{id})$$

where,

$$g(t) = \frac{1}{1 + exp(-t)}$$



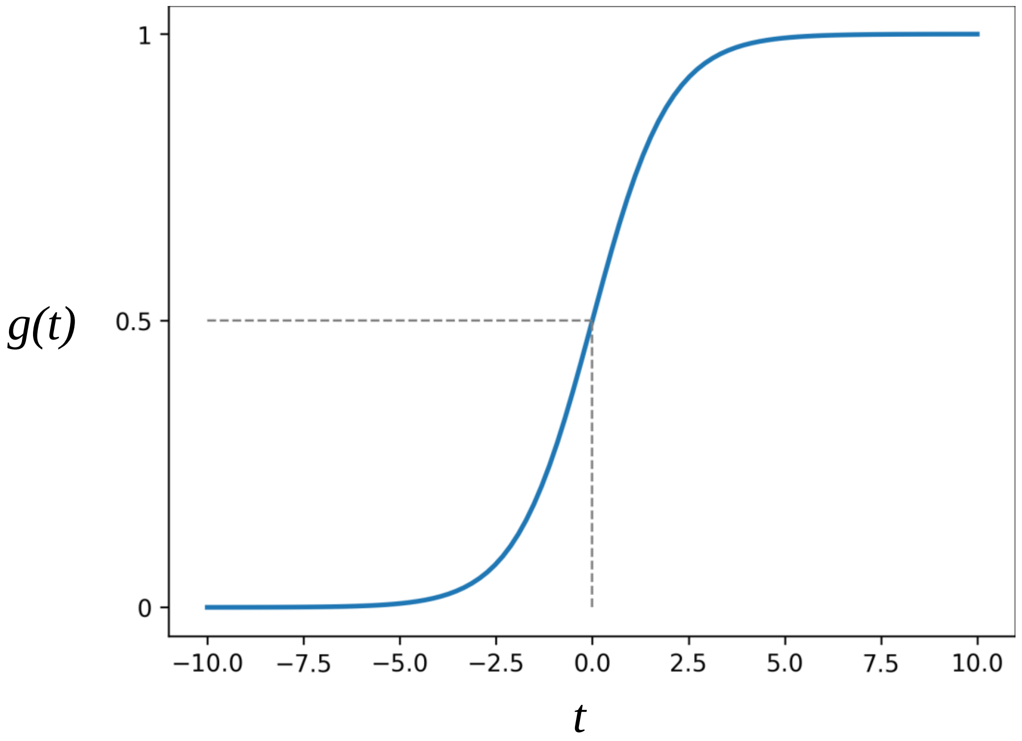
Logistic Regression

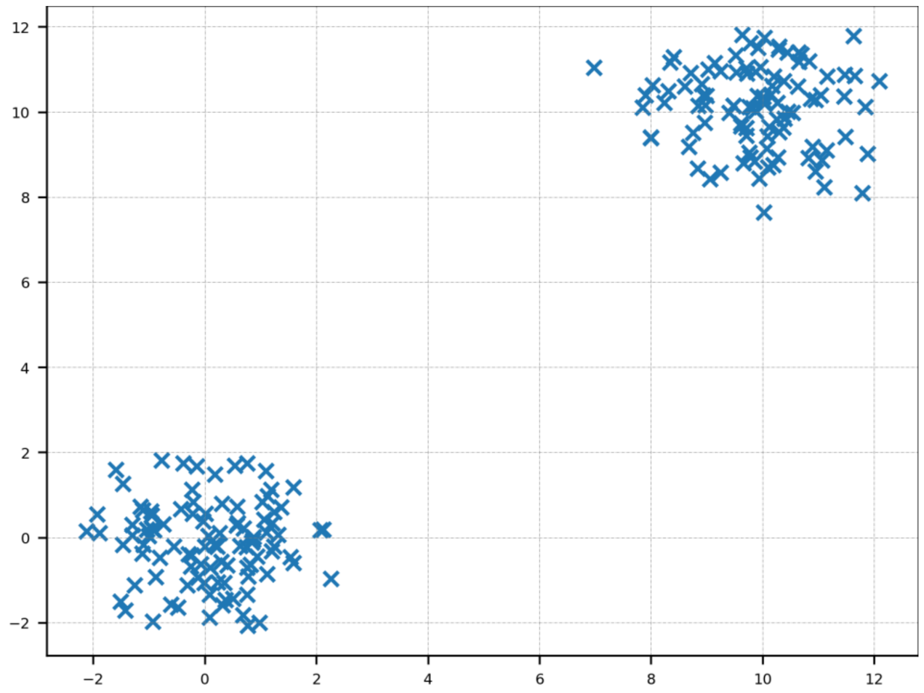
We estimate the function:

$$\hat{y}_i = g(w_0 + w_1 x_{i1} + w_2 x_{i2} + \ldots + w_d x_{id})$$

where,

$$g(t) = \frac{1}{1 + exp(-t)}$$





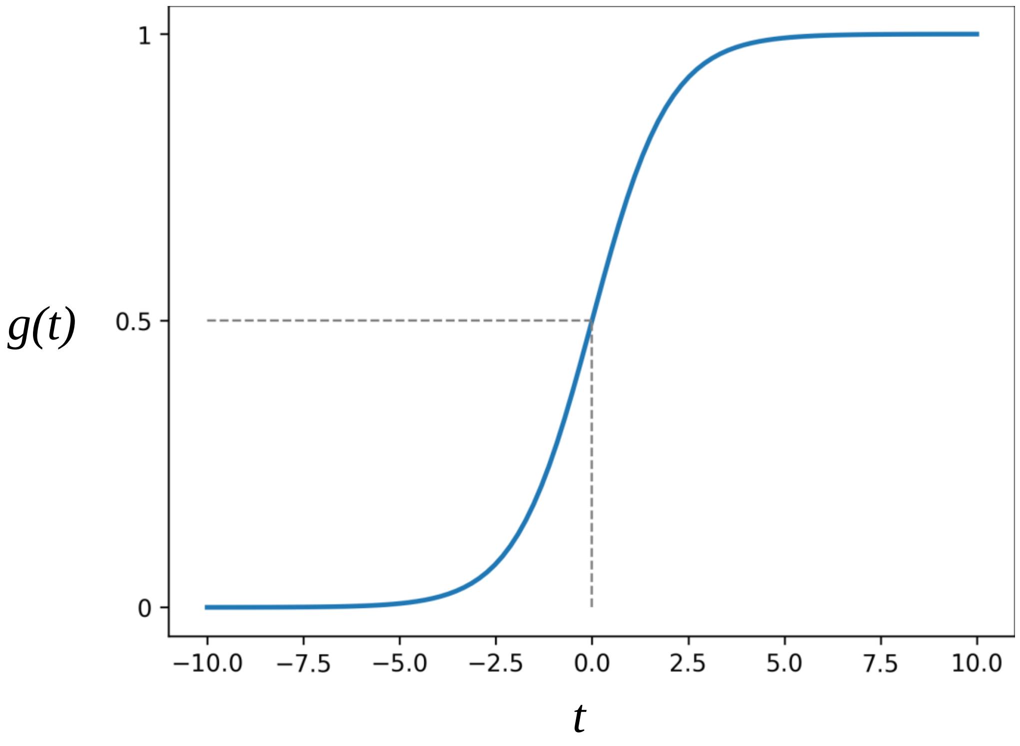
Logistic Regression

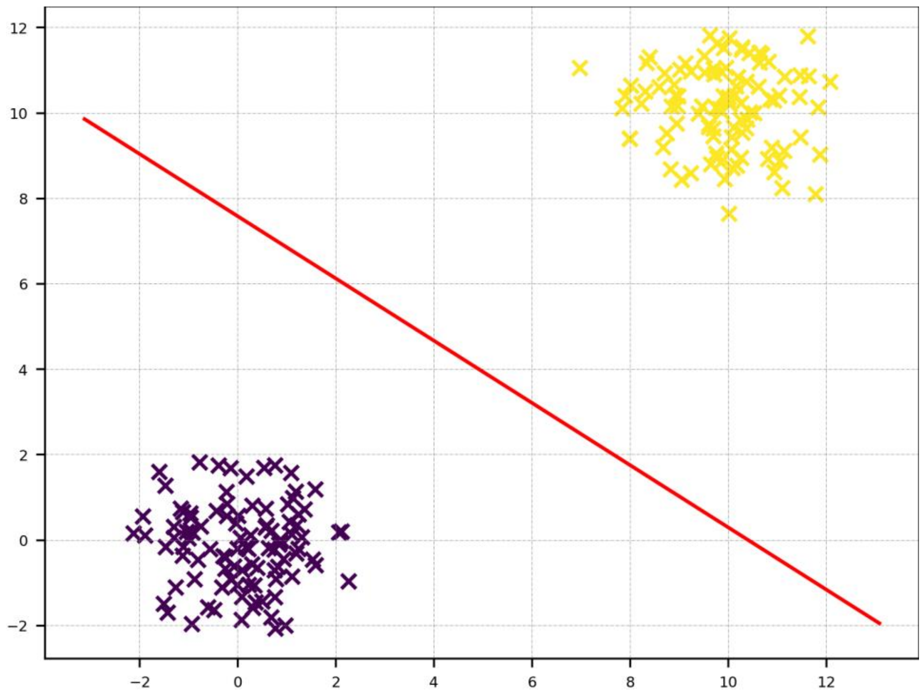
We estimate the function:

$$\hat{y}_i = g(w_0 + w_1 x_{i1} + w_2 x_{i2} + \ldots + w_d x_{id})$$

where,

$$g(t) = \frac{1}{1 + exp(-t)}$$





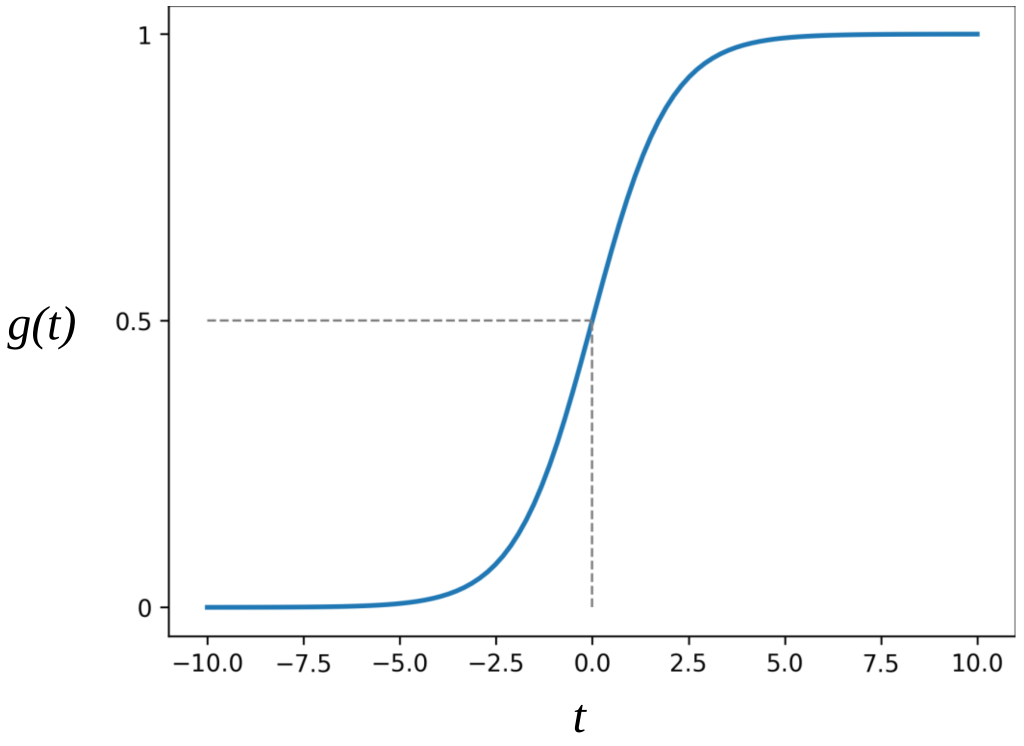
Logistic Regression

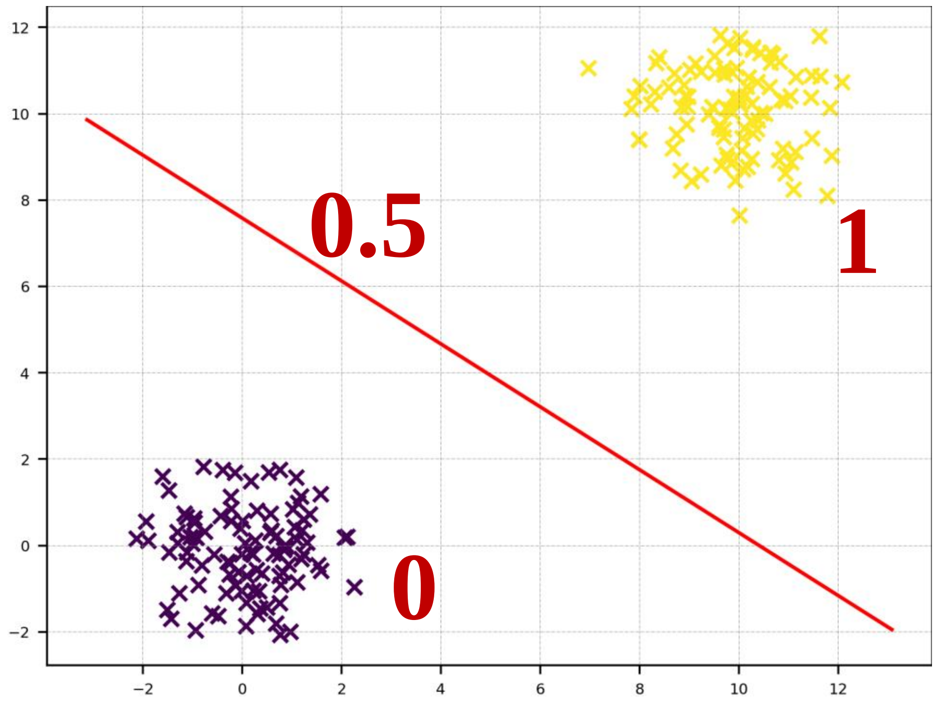
We estimate the function:

$$\hat{y}_i = g(w_0 + w_1 x_{i1} + w_2 x_{i2} + \ldots + w_d x_{id})$$

where,

$$g(t) = \frac{1}{1 + exp(-t)}$$





Logistic Regression

We estimate the function:

$$\hat{y}_i = g(w_0 + w_1 x_{i1} + w_2 x_{i2} + ... + w_d x_{id})$$

where,

$$g(t) = \frac{1}{1 + exp(-t)}$$

Choice of Loss Functions:

- Mean Square Error:

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

- Binary Cross Entropy Loss:

$$-\frac{1}{n}\sum_{i=1}^{n}\{y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)\}$$

Logistic Regression

Mean Square Error loss function:

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \frac{1}{n}\sum_{i=1}^{n}\ell_i$$

- For correctly classified points:
  - When $y_i = 1$ and $\hat{y}_i = 1, \ell_i = 0$.
  - When $y_i = 0$ and $\hat{y}_i = 0, \ell_i = 0$.
- For misclassified points:
  - When $y_i = 1$ and $\hat{y}_i = 0, \ell_i = 1$.
  - When $y_i = 0$ and $\hat{y}_i = 1, \ell_i = 1$.

Logistic Regression

Binary Cross-Entropy loss function:

$$-\frac{1}{n}\sum_{i=1}^{n}\{y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)\} = \frac{1}{n}\sum_{i=1}^{n}\ell_i$$

- For correctly classified points:
  - When $y_i \to 1$ and $\hat{y}_i \to 1, \ell_i \to 0$.
  - When $y_i \to 0$ and $\hat{y}_i \to 0, \ell_i \to 0$.
- For misclassified points:
  - When $y_i \to 1$ and $\hat{y}_i \to 0, \ell_i \to +\infty$.
  - When $y_i \to 0$ and $\hat{y}_i \to 1, \ell_i \to +\infty$.

Logistic Regression

We estimate the function:

$$\hat{y}_i = g(w_0 + w_1 x_{i1} + w_2 x_{i2} + \dots + w_d x_{id})$$

where,

$$g(t) = \frac{1}{1 + exp(-t)}$$

Choice of Loss Functions:

- Mean Square Error:

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

- Binary Cross Entropy Loss:

$$-\frac{1}{n}\sum_{i=1}^{n}\{y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)\}$$

How can we optimize these loss functions?
- Reweighted Least Squares
- Gradient Descent

Logistic Regression

We estimate the function:

$$\hat{y}_i = g(w_0 + w_1 x_{i1} + w_2 x_{i2} + \ldots + w_d x_{id})$$

where,

$$g(t) = \frac{1}{1 + exp(-t)}$$

Choice of Loss Functions:

- Mean Square Error:

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

- Binary Cross Entropy Loss:

$$-\frac{1}{n}\sum_{i=1}^{n}\{y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)\}$$

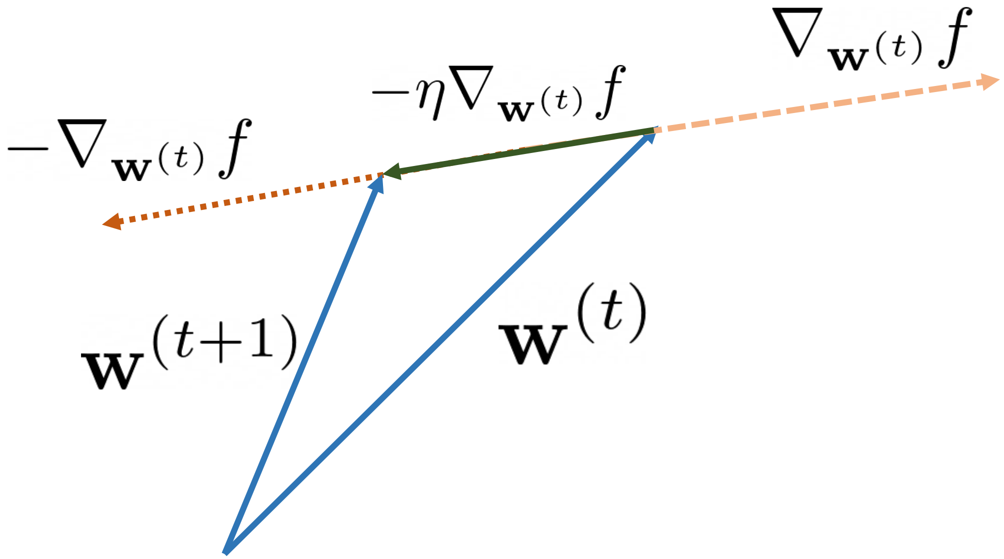
How can we optimize these loss functions?

- Reweighted Least Squares
- Gradient Descent

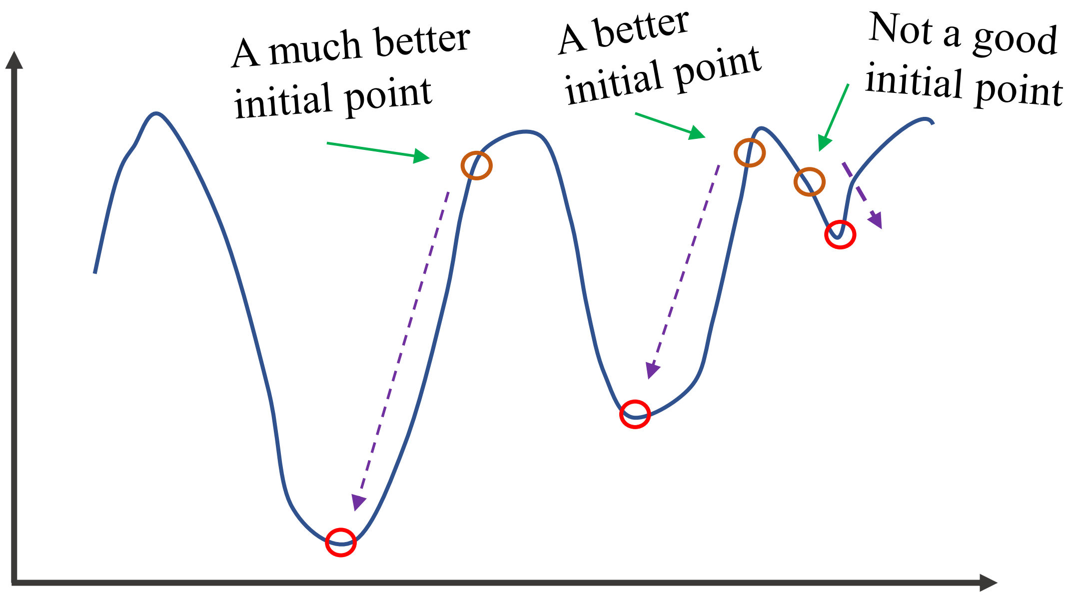
Gradient Descent

We wish to optimize a differentiable function $f_{\mathbf{w}} : X \rightarrow \mathbf{y}$ by the following procedure:

1. Initialize $\mathbf{w}^{(0)}$
2. Update $\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \eta \nabla_{\mathbf{w}^{(t)}} f$



$\mathbf{w}^{(t+1)}$ is updated by a small amount in the negative direction of the gradient

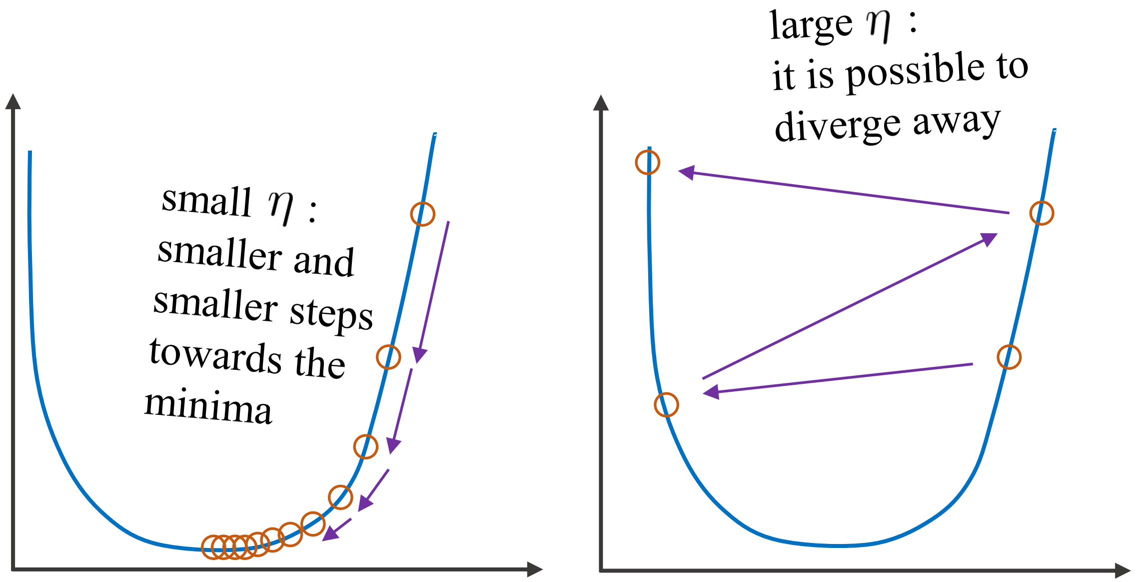


The Gradient Descent procedure is usually run multiple times from different initializations to obtain the best local minima

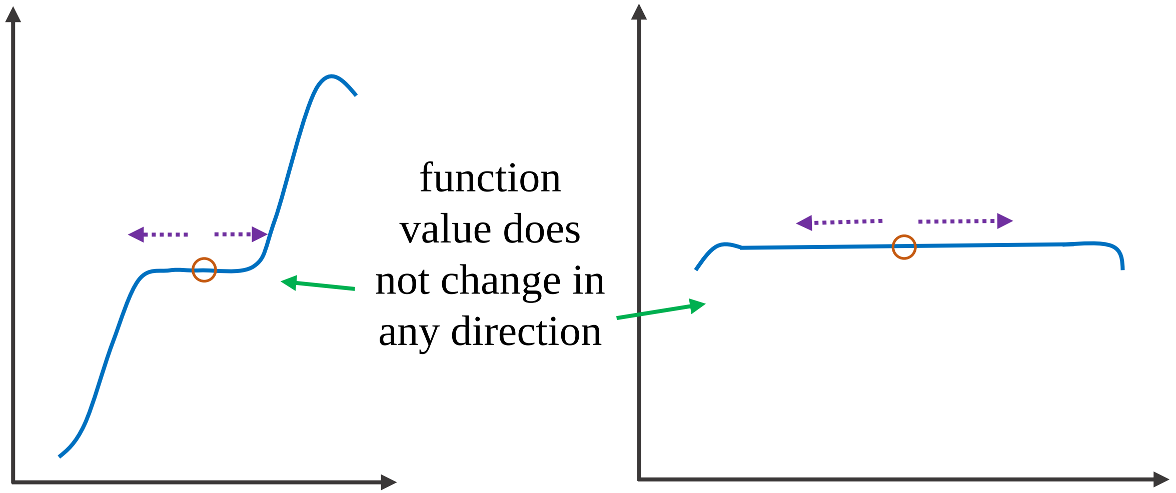
Gradient Descent

We wish to optimize a differentiable function $f_{\mathbf{w}} : X \rightarrow \mathbf{y}$ by the following procedure:

1. Initialize $\mathbf{w}^{(0)}$
2. Update $\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \eta \nabla_{\mathbf{w}^{(t)}} f$



The 'step size' η needs to be small enough, large η can cause divergence



Gradient Descent can get stuck at points of inflections and plateau regions

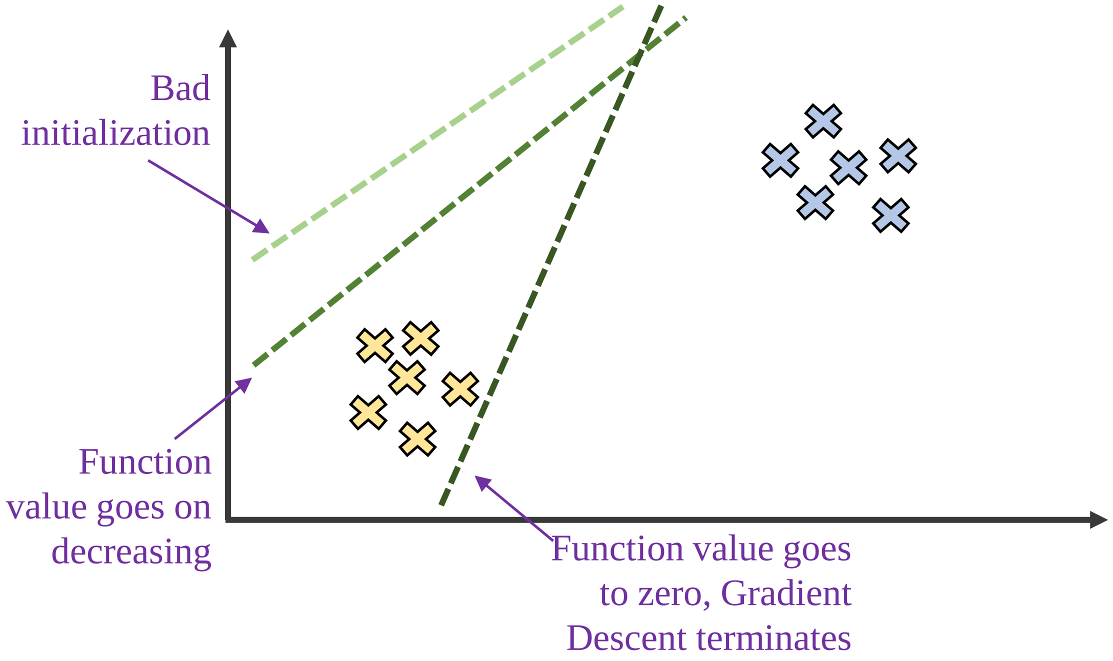
Using Gradient Descent for Logistic Regression: BCE vs MSE

Binary Cross-Entropy

$$-\frac{1}{n}\sum_{i=1}^{n}\{y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)\} = \frac{1}{n}\sum_{i=1}^{n}\ell_i$$

- For misclassified points:
  - When $y_i \to 1$ and $\hat{y}_i \to 0, \ell_i \to +\infty$
  - When $y_i \to 0$ and $\hat{y}_i \to 1, \ell_i \to +\infty$

Large gradient magnitudes

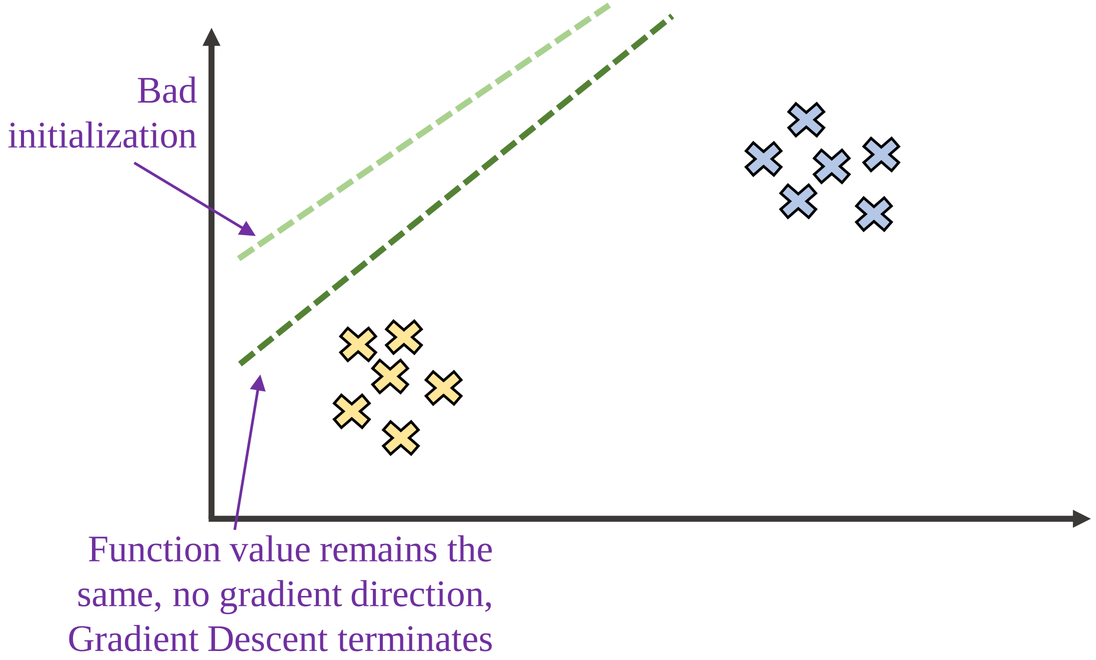


Mean Square Error

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \frac{1}{n}\sum_{i=1}^{n}\ell_i$$

- For misclassified points:
  - When $y_i = 1$ and $\hat{y}_i = 0, \ell_i = 1$
  - When $y_i = 0$ and $\hat{y}_i = 1, \ell_i = 1$

Small gradient magnitudes



Reading Material on Linear & Logistic Regression:

- Chapters 2, 3, 4, in *An Introduction to Statistical Learning with Applications in R*, by James G., Witten D., Hastie T., Tibshirani R. (https://www.statlearning.com/)

- Chapter 2 'Generative and Discriminative Classifiers: Naive Bayes and Logistic Regression', in *Machine Learning*, by Tom Mitchell. (http://www.cs.cmu.edu/~tom/mlbook/NBayesLogReg.pdf)

Readings for the Next Class:

- Chapter 2 'Optimal Classification', in *Fundamentals of Pattern Recognition and Machine Learning*, by Ulisses Braga-Neto.