Introduction to Machine Learning

#05 Test MSE and Cross-Validation

HAYASHI, Toshiharu

Polynomial Regression

Polynomial: 多項式

- Consider the following regression model:

$$Y = f(X) + \varepsilon,$$

where f is a degree-d polynomial, or equivalently

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \cdots + \beta_d X^d + \varepsilon.$$

- In order to estimate the parameters, the training data are observed. The n observations of (X, Y) are denoted by

$$(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n).$$

- The estimates can be used for predicting Y: $\hat{Y} = \hat{f}(X)$.
- Our goal in today's class is to determine the degree d for predicting Y.
- In practice, you should apply the spline regressions instead of the polynomial regressions, which is left as an assignment.

Estimation of Parameters

$$Y = f(X) + \varepsilon$$

- The model:

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \cdots + \beta_d X^d + \varepsilon.$$

- Consider $X, X^2, \dots, X^d$ as d inputs $X_1, X_2, \cdots, X_d$. Then the above model becomes

$$\underline{Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_d X_d + \varepsilon}, \quad \text{where } X_i = X^i, i = 1, 2, \dots, d.$$

- The LSE $\widehat{\boldsymbol{\beta}} = \left(\hat{\beta}_0, \hat{\beta}_1, \cdots, \hat{\beta}_d\right)^T$ in the above multiple linear regression model is

$$\widehat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\boldsymbol{y}, \qquad \text{where } \mathbf{X} = \underbrace{\begin{pmatrix} 1 & x_1 & x_1^{\,2} & \cdots & x_1^{\,d} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_i & x_i^{\,2} & \cdots & x_i^{\,d} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & x_n^{\,2} & \cdots & x_n^{\,d} \end{pmatrix}}_{n \times (1+d)}, \; \boldsymbol{y} = \underbrace{\begin{pmatrix} y_1 \\ \vdots \\ y_i \\ \vdots \\ y_n \end{pmatrix}}_{n \times 1}.$$

Mean Squares Error (MSE)

$$Y = f(X) + \varepsilon$$

- Training mean squares error:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}(x_i)\right)^2 = \frac{1}{n}\sum_{i=1}^{n}\left(y_k - \hat{\beta}_0 - \hat{\beta}_1 x_i - \cdots - \hat{\beta}_d {x_i}^d\right)^2$$

where $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ are the training data.

- The LSE is the minimizer of the training MSE.

$$\widehat{\boldsymbol{\beta}} = \arg\min_{\widehat{\boldsymbol{\beta}}} \mathrm{MSE}, \qquad \widehat{\boldsymbol{\beta}} = \left(\hat{\beta}_0, \hat{\beta}_1, \cdots, \hat{\beta}_p\right)^T$$

- Hereafter, the training MSE means the minimum value of MSE, that is MSE for the LSE $\widehat{\boldsymbol{\beta}}$.
- As the degree d becomes higher, the training MSE becomes the lower, even if high degree terms have no effect on the prediction for Y.
- The training MSE canNOT be used to determine the degree d.

Prediction Error

- We would like to minimizes the following expectation:

$$E\left[\left(Y - \hat{f}(X)\right)^2\right],$$

the expectation of the squared error for prediction by using the fitted function. We call it the prediction error, shortly.

- We may determine the degree d which minimizes the prediction error.
- If various observations (x_0, y_0) will be obtained, the prediction error can be approximately evaluated by the test MSE, or

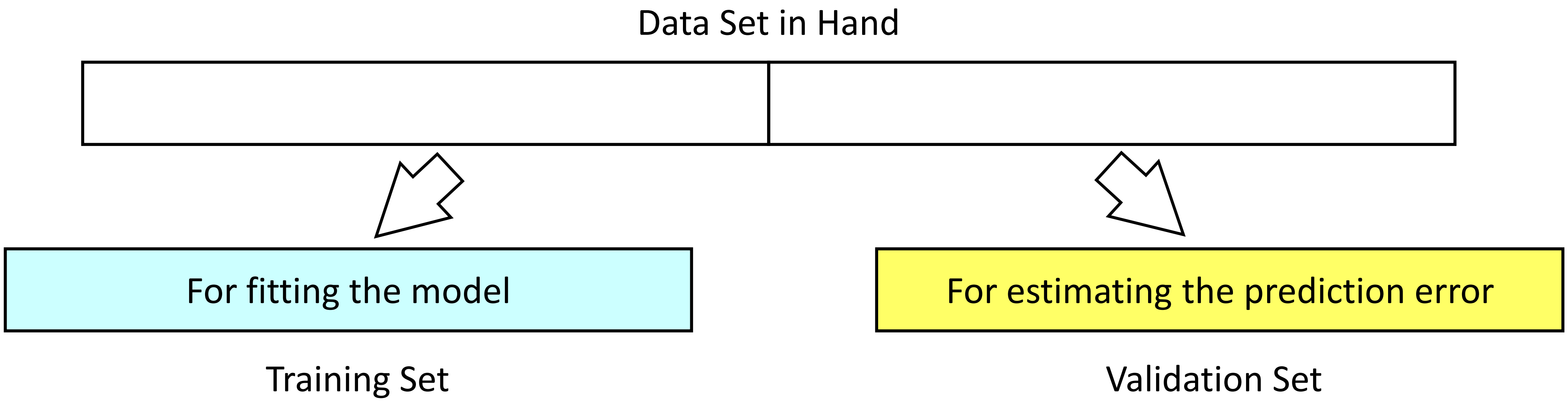
$$E\left[\left(Y - \hat{f}(X)\right)^2\right] \approx \text{Average of} \left(y_0 - \hat{f}(x_0)\right)^2,$$

where the average is taken over the various observations (x_0, y_0), the test data.

- The test data cannot be obtained in many practical situations.

Validation Set Approach

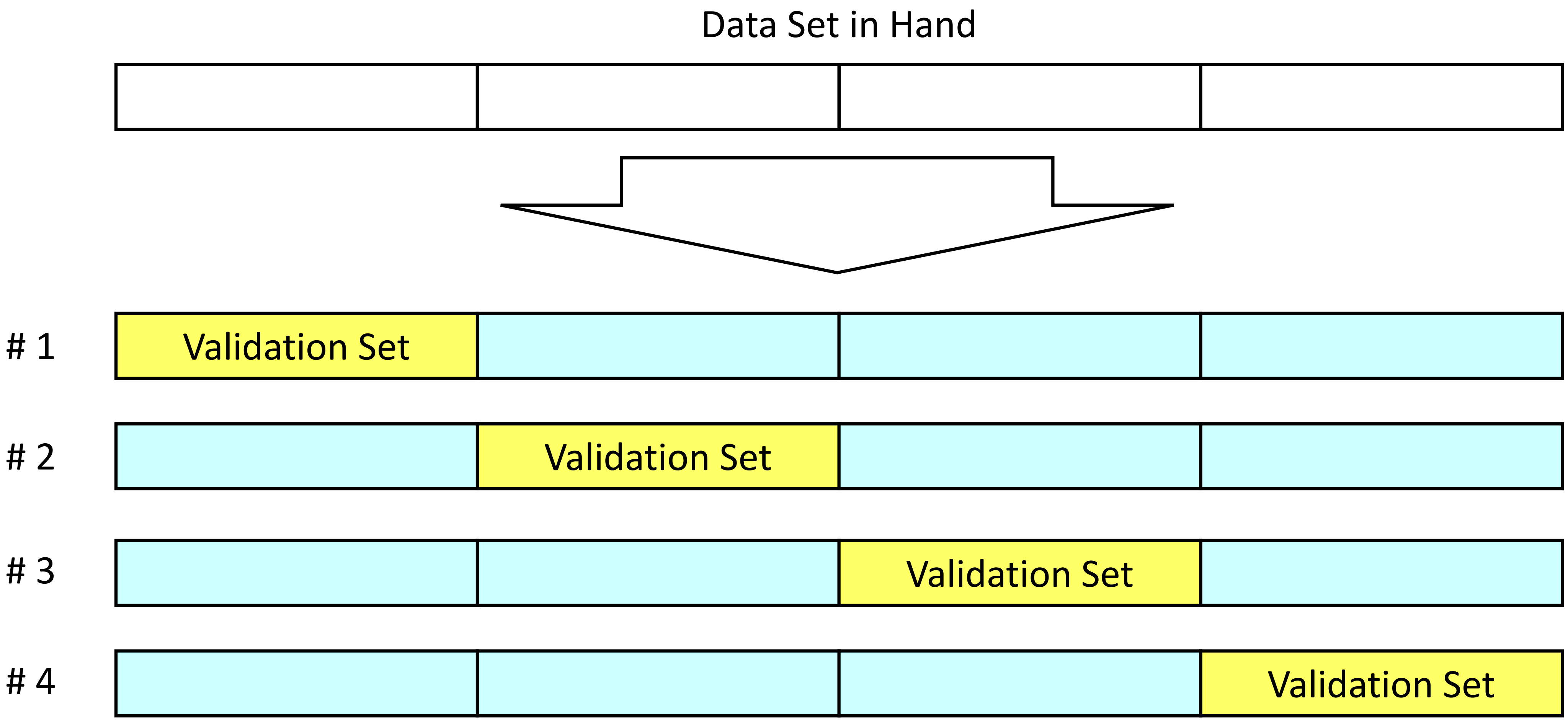
- The data set in hand is divided into two sets. One is used for fitting the model and the other is used for estimating the prediction error.



- This approach has two issues:
  - The model is fitted to only a half of the data set in hand.
  - The estimate of the prediction error is unstable if the size of the validation set is small.

Cross-Validation

- We split the data set into several parts.
- Treat each part as the validation set and the other parts as the training set.



K-Fold Cross-Validation

1. Split the data set into K parts whose sizes are roughly equal.
2. For the kth part, fit the model to the other $K - 1$ parts of the data. Denote the fitted function by $\hat{f}_{(-k)}(x)$.
3. Evaluate the squared error $SE_i = \left(y_i - \hat{f}_{(-k)}(x_i)\right)^2$ for every observation (x_i, y_i) in the kth part of the data.
4. Repeat the above steps 2 and 3 for $k = 1, 2, \dots, K$.
5. Finally, calculate the cross-validation estimate:

$$\mathrm{CV} = \mathrm{CV}_K = \frac{1}{n}\sum_{i=1}^{n} SE_i.$$

- Typically, $K = 5$ or 10.

Leave-One-Out Cross-Validation

1. Leave one observation, say (x_i, y_i), out of the data set.
2. Fit the model to the remaining data. Denote the fitted function by $\hat{f}_{-i}(x)$.
3. Evaluate the squared error $\left(y_i - \hat{f}_{-i}(x_i)\right)^2$.
4. Repeat the above steps for every observation in the original data set.
5. Finally, calculate the cross-validation estimate:

$$\mathrm{CV} = \mathrm{CV}_{\mathrm{LOO}} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}_{-i}(x_i)\right)^2.$$

- For the multiple regression including the polynomial regression, we have

the magic formula: $\mathrm{CV}_{\mathrm{LOO}} = \frac{1}{n}\sum_{i=1}^{n}\left\{\frac{y_i - \hat{f}(x_i)}{(1-h_{ii})}\right\}^2,$

where h_{ii} is the ith diagonal element of $\mathbf{H} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$.

Exercise 5.1

- Explain about the K-fold and leave-one-out cross-validations.

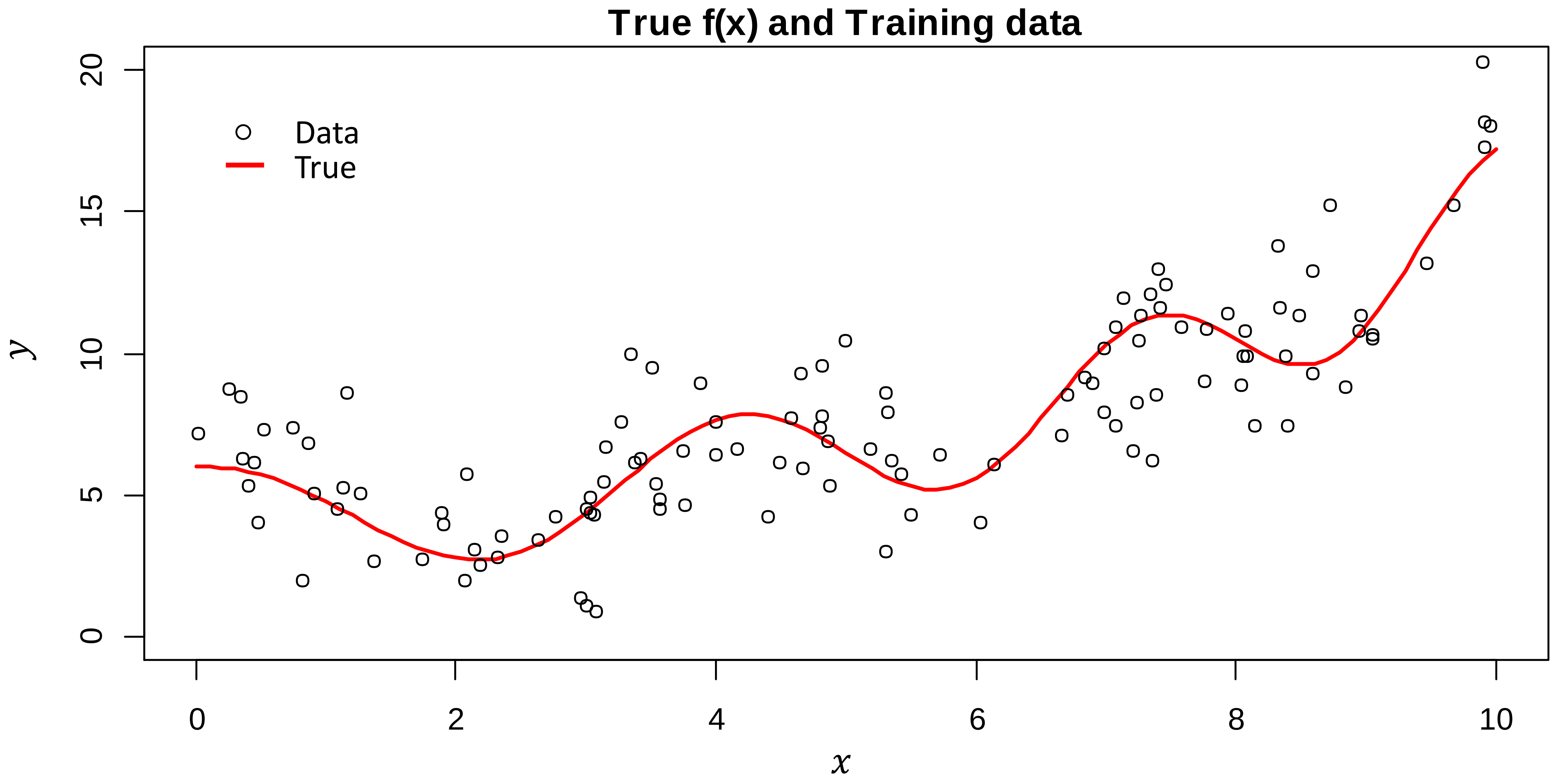
An Example

- True model:

$$Y = f(X) + \varepsilon, \qquad f(x) = 2\cos\left\{\frac{\pi}{5}\left(2x + \frac{1}{10}x^2\right)\right\} + e^{x/4} + 3,$$

  where X has the uniform distribution on the interval $[0, 10]$, $U(0, 10)$, and ε has the normal distribution $N(0, 4)$. Moreover, X and ε are independent.
- Models to be fitted: $f(x)$ is a degree-d polynomial for $d = 1, 2, \dots, 20$.
- Size of the training data: 120.
- Find the degree d which minimizes the leave-one-out cross-validation estimate $\mathrm{CV}_{\mathrm{LOO}}$ or the 10-fold cross-validation estimate CV_{10}.
- Test data are 5000 observations generated from the true model. Then, the test MSE is evaluated.

The True $f(x)$ and the Training Data



Estimation of Parameters

$$Y = f(X) + \varepsilon$$

- The model to be fitted:

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \cdots + \beta_d X^d + \varepsilon.$$

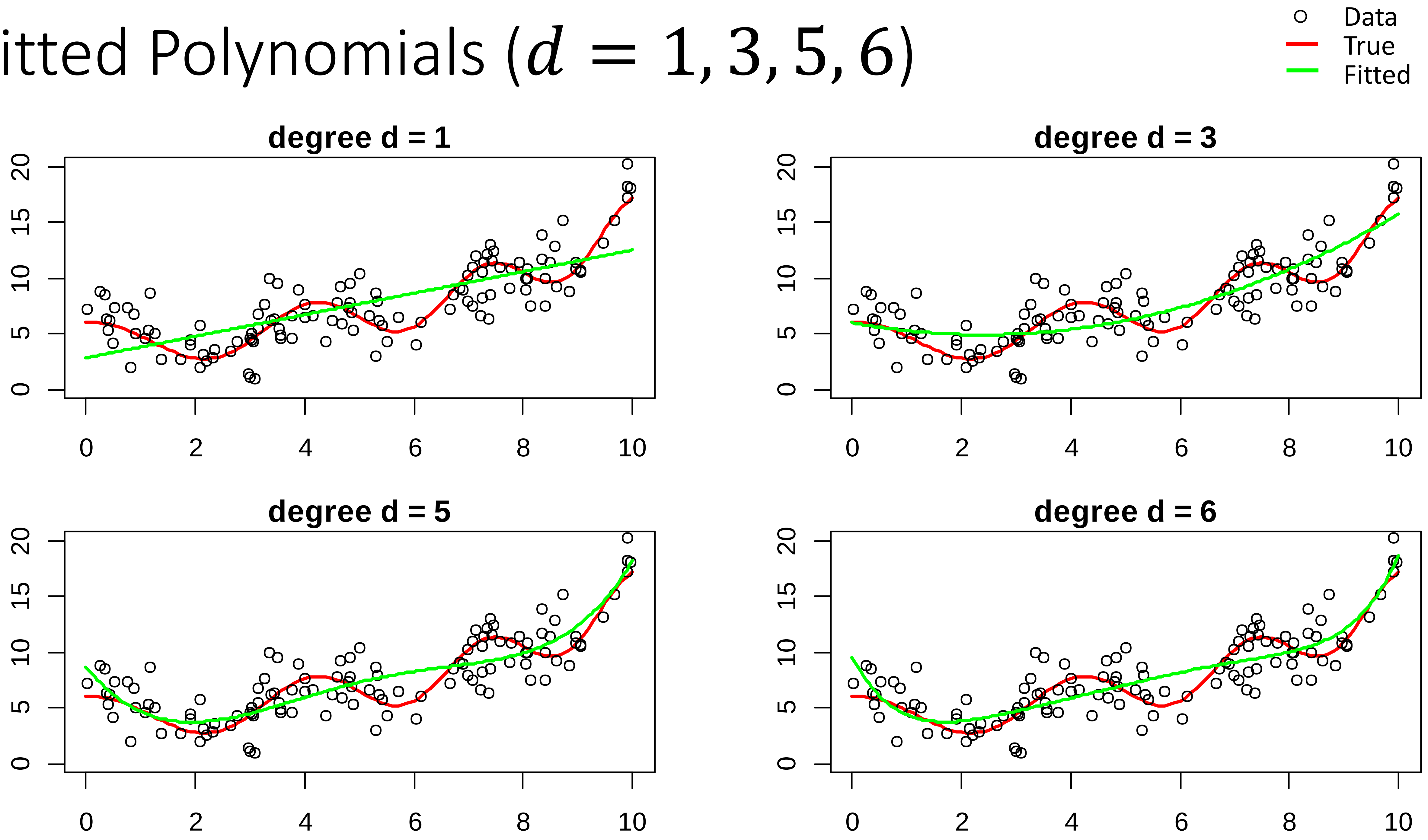
- Consider $X, X^2, \ldots, X^d$ as d inputs $X_1, X_2, \cdots, X_d$. Then the above model becomes

$$\underline{Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_d X_d + \varepsilon}, \quad \text{where } X_i = X^i, i = 1, 2, \ldots, d.$$

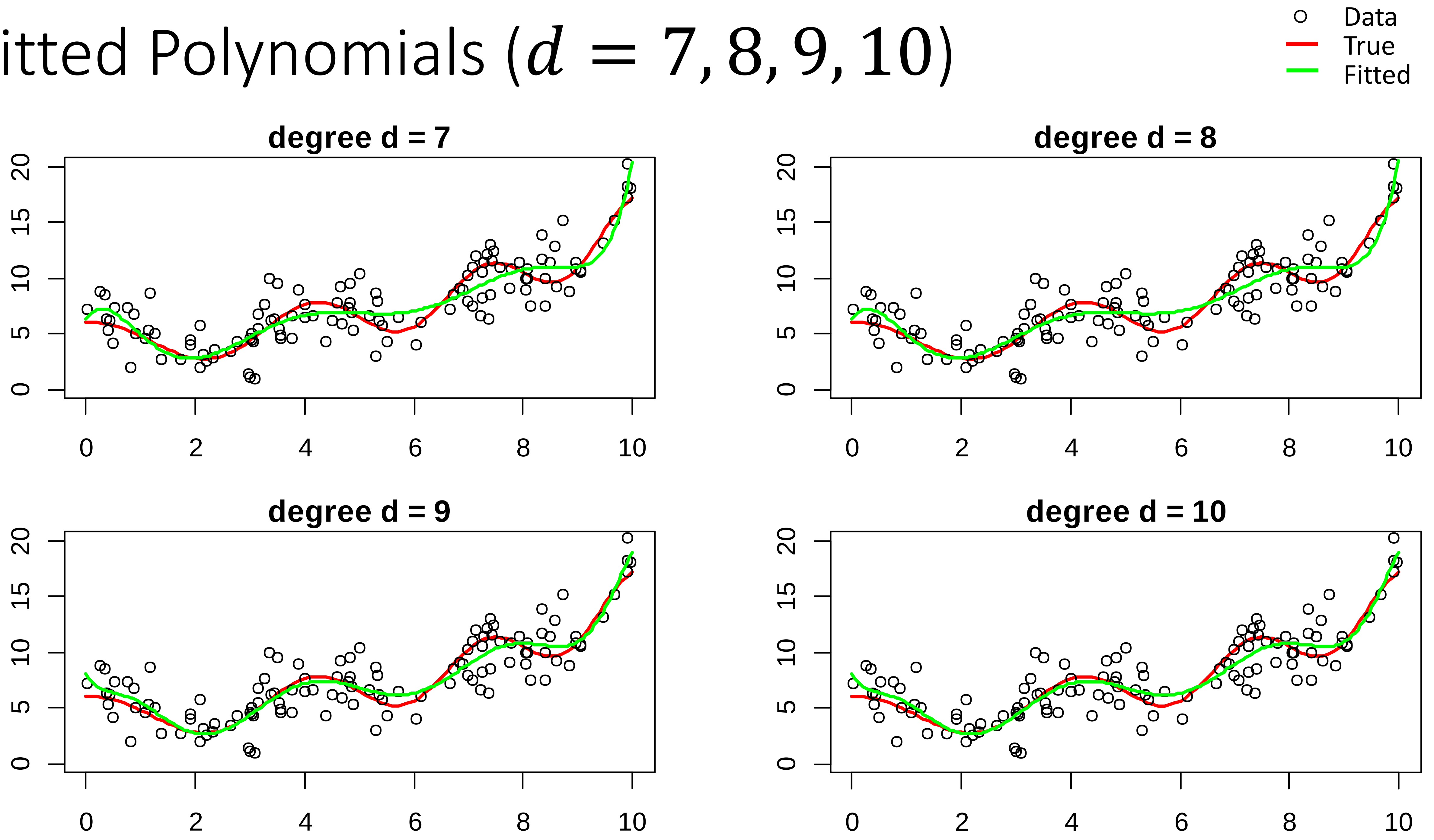
- The LSE $\widehat{\boldsymbol{\beta}} = \left(\hat{\beta}_0, \hat{\beta}_1, \cdots, \hat{\beta}_d\right)^T$ in the above multiple linear regression model is

$$\widehat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\boldsymbol{y}, \qquad \text{where } \mathbf{X} = \underset{n \times (1+d)}{\begin{pmatrix} 1 & x_1 & \cdots & x_1^{\,d} \\ \vdots & \vdots & & \vdots \\ 1 & x_i & \cdots & x_i^{\,d} \\ \vdots & \vdots & & \vdots \\ 1 & x_n & \cdots & x_n^{\,d} \end{pmatrix}}, \ \boldsymbol{y} = \underset{n \times 1}{\begin{pmatrix} y_1 \\ \vdots \\ y_i \\ \vdots \\ y_n \end{pmatrix}}.$$

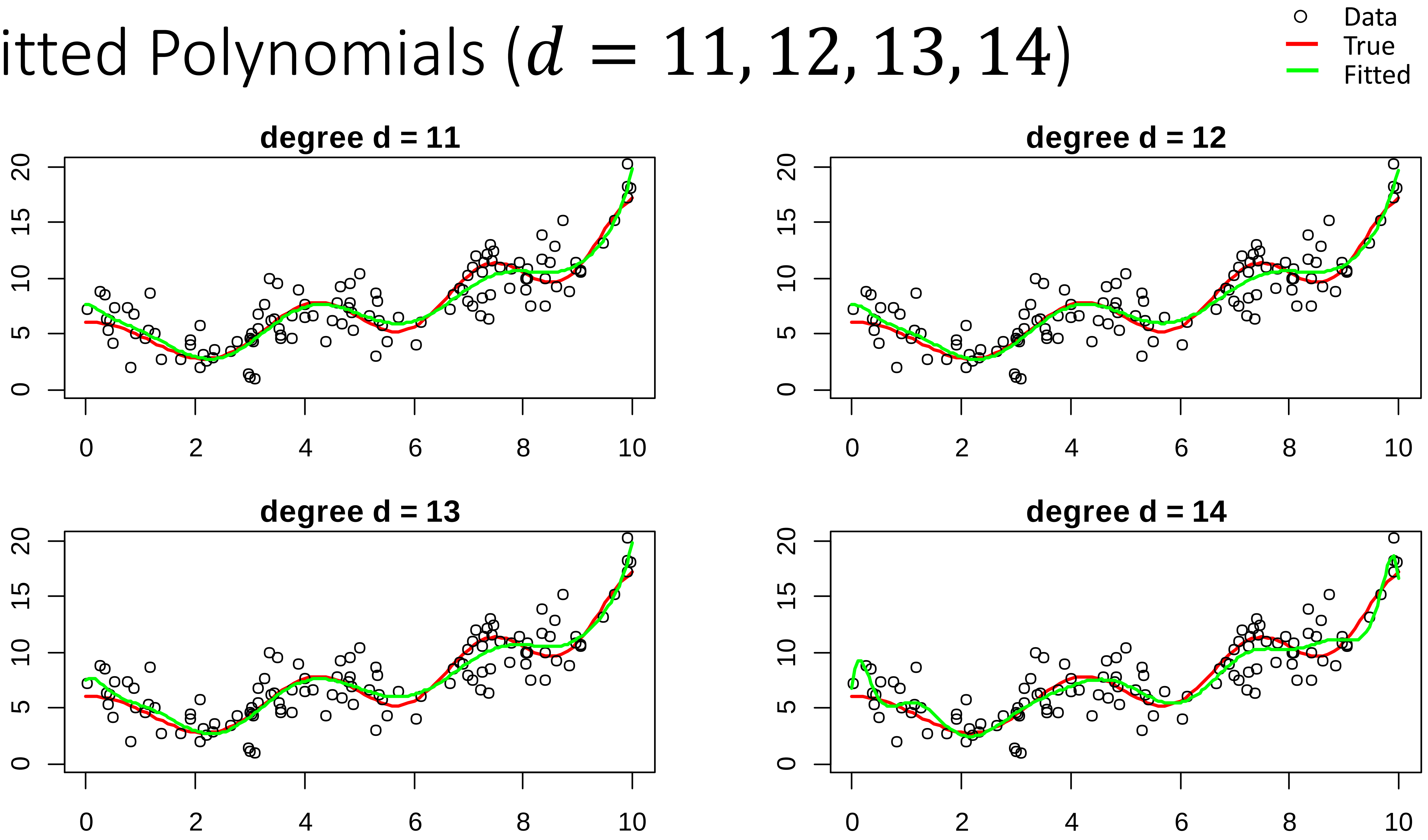
Fitted Polynomials ($d = 1, 3, 5, 6$)



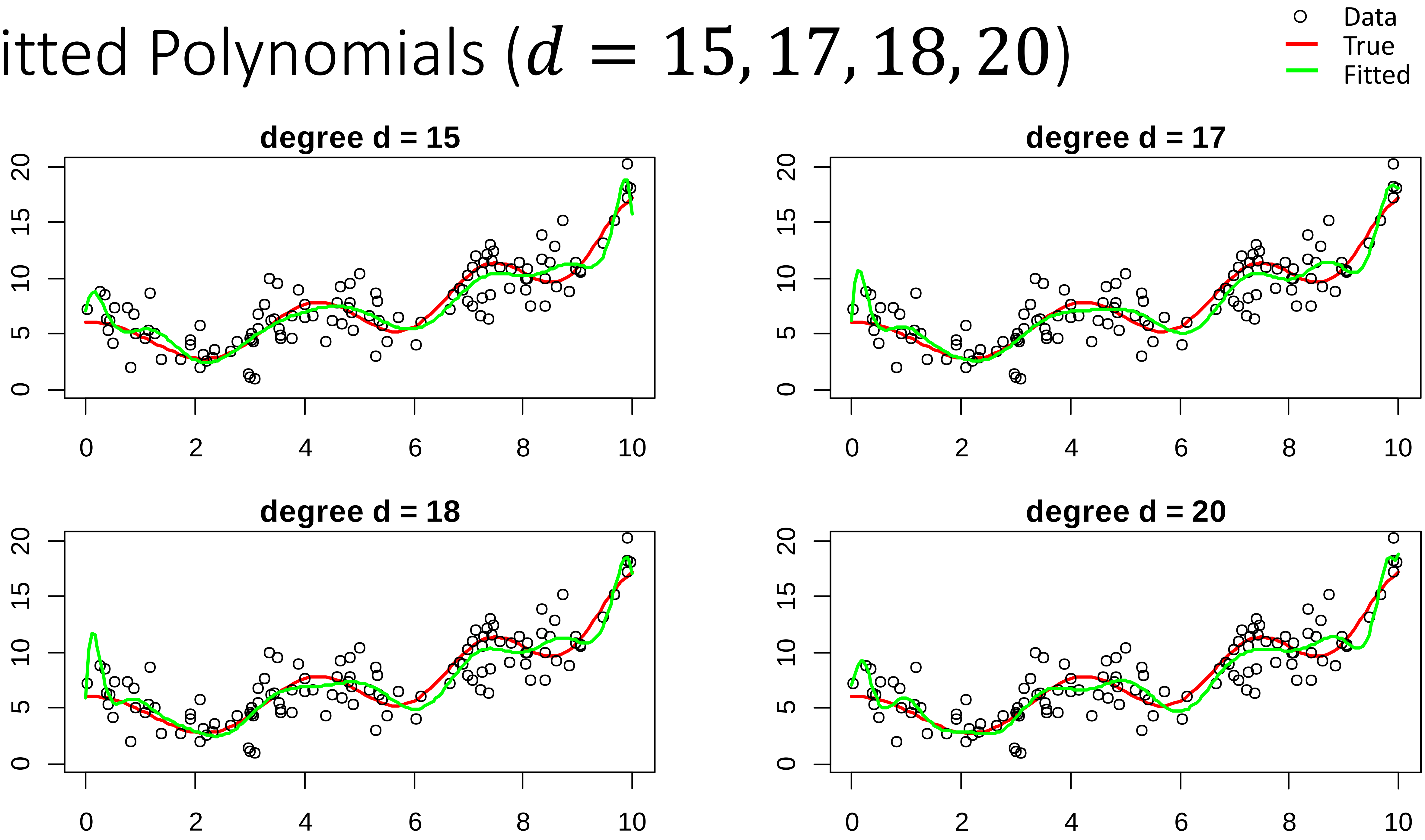
Fitted Polynomials ($d = 7, 8, 9, 10$)



Fitted Polynomials ($d = 11, 12, 13, 14$)



Fitted Polynomials ($d = 15, 17, 18, 20$)



For K-Fold Cross-Validation

$$Y = f(X) + \varepsilon$$

- Remove the kth part from the data matrix $\mathbf{X}$ to make $\mathbf{X}_{(-k)}$ as follows:

kth part

$$\mathbf{X} = \begin{pmatrix} 1 & x_1 & \cdots & x_1^{\,d} \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ 1 & x_i & \cdots & x_i^{\,d} \\ \vdots & \vdots & & \vdots \\ 1 & x_n & \cdots & x_n^{\,d} \end{pmatrix} \Rightarrow \begin{pmatrix} 1 & x_1 & \cdots & x_1^{\,d} \\ \vdots & \vdots & & \vdots \\ 1 & x_i & \cdots & x_i^{\,d} \\ \vdots & \vdots & & \vdots \\ 1 & x_n & \cdots & x_n^{\,d} \end{pmatrix} = \mathbf{X}_{(-k)}.$$

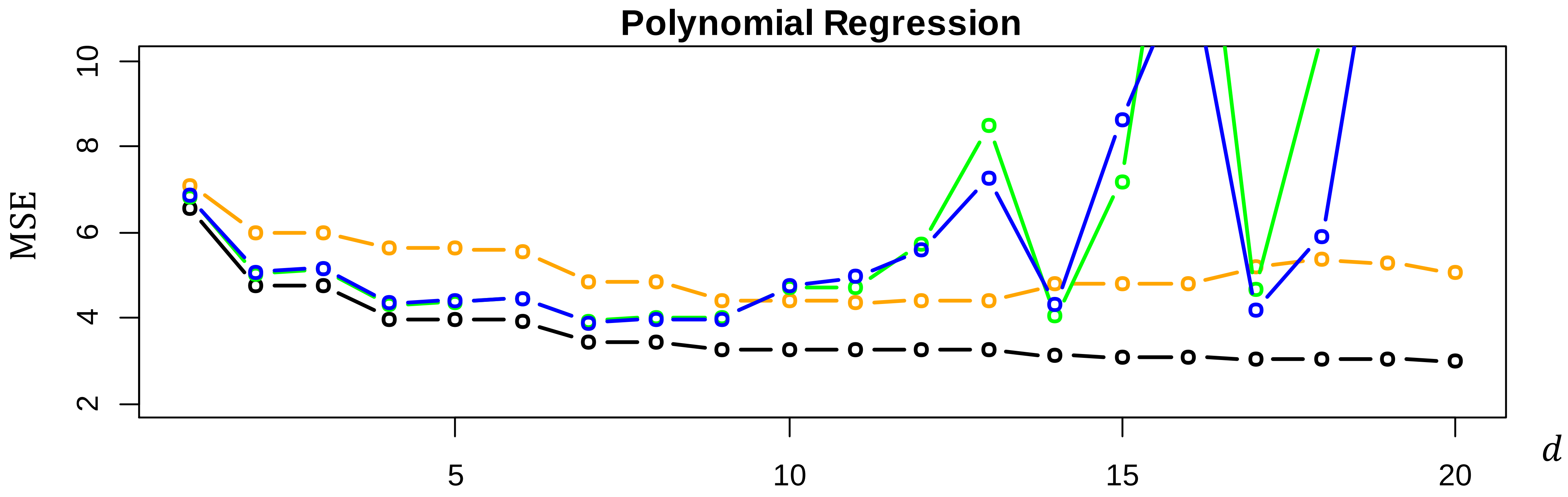
Similarly, remove the kth part from the vector $\boldsymbol{y}$ to make the vector $\boldsymbol{y}_{(-k)}$.

- Fit the model to the data without kth part to obtain the fitted function:

$$\hat{f}_{(-k)}(x) = \hat{\beta}_{(-k),0} + \hat{\beta}_{(-k),1}\, x + \cdots + \hat{\beta}_{(-k),d}\, x^d,$$

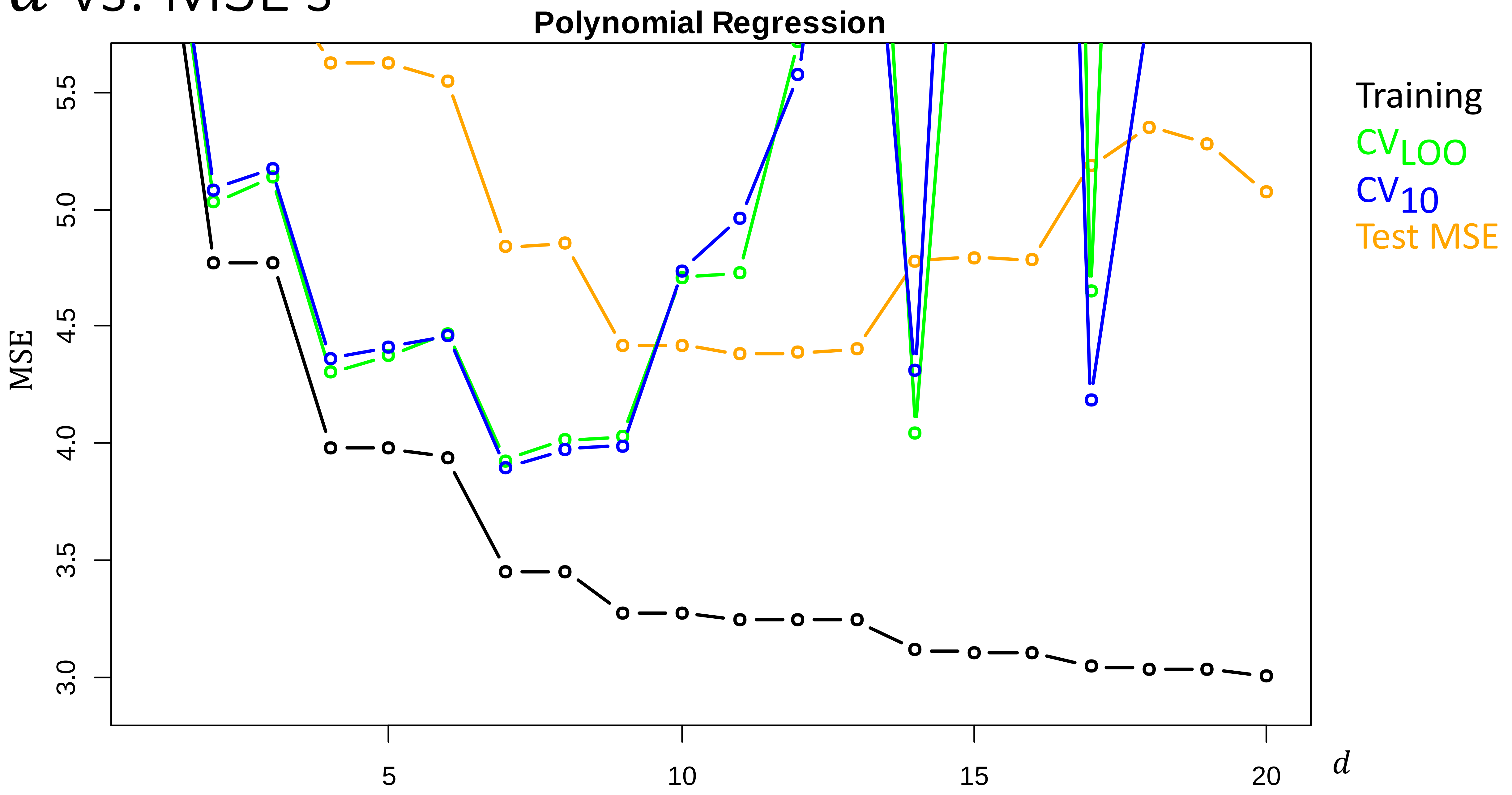
where $\widehat{\boldsymbol{\beta}}_{(-k)} = \left(\hat{\beta}_{(-k),0}, \hat{\beta}_{(-k),1}, \cdots, \hat{\beta}_{(-k),d}\right)^T = \left(\mathbf{X}_{(-k)}^T \mathbf{X}_{(-k)}\right)^{-1} \mathbf{X}_{(-k)}^T \boldsymbol{y}_{(-k)}$.

d vs. MSE's



d	6	7	8	9	10	11	12	13
Training MSE	3.937	3.453	3.453	3.273	3.273	3.246	3.244	3.243
CV_{LOO}	4.470	3.925	4.014	4.029	4.710	4.728	5.720	8.476
CV_{10}	4.461	3.898	3.974	3.988	4.738	4.961	5.574	7.280
Test MSE	5.549	4.839	4.853	4.418	4.417	4.379	4.387	4.404

d vs. MSE's


Polynomial Regression
MSE
3.0
3.5
4.0
4.5
5.0
5.5
5
10
15
20
d
Training
CV_{LOO}
CV_{10}
Test MSE

Report 5

We consider the polynomial regression with degree-d, and denote the training mean squares error by $\mathrm{MSE}(d)$. Then

$$\mathrm{MSE}(d) = \min_{\hat{\beta}_0, \hat{\beta}_1, \dots, \hat{\beta}_d} \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i - \hat{\beta}_2 {x_i}^2 - \cdots - \hat{\beta}_d {x_i}^d \right)^2,$$

where $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ are the training data.

Show that the training MSE monotonically decreases as d increases, that is,

$$\mathrm{MSE}(d) \geq \mathrm{MSE}(d+1)$$

for any $d \geq 1$.

(Hint) A degree-$(d+1)$ polynomial $\hat{\beta}_0 + \hat{\beta}_1 x_i + \cdots + \hat{\beta}_d {x_i}^d + \hat{\beta}_{d+1} {x_i}^{d+1}$ is equal to the degree-d polynomial $\hat{\beta}_0 + \hat{\beta}_1 x_i + \cdots + \hat{\beta}_d {x_i}^d$ if $\hat{\beta}_{d+1} = 0$.

- Submit the report via “Moodle System” by 3:00pm on 23th May.

Maximum

- Consider a real-valued function $f(x, y)$, and let $g(x) = f(x, c)$, where c is a constant. Then we have

$$\max_{x,y} f(x, y) \geq \max_{x} g(x) \ (= \max_{x \in \mathbb{R},\, y = \mathrm{c}} f(x, y)).$$

$\because$ Let $x^* = \arg\max_{x} g(x)$. Then $g(x^*) = \max_{x} g(x)$.

On the other hand, for any $\tilde{x}$ and $\tilde{y}$, $\max_{x,y} f(x, y) \geq f(\tilde{x}, \tilde{y})$.

In particular, for $\tilde{x} = x^*$ and $\tilde{y} = c$, we have

$$\max_{x,y} f(x, y) \geq f(x^*, c) = g(x^*) = \max_{x} g(x).$$

Review of the Regression Spline

$$Y = f(X) + \varepsilon$$

- Suppose f is a cubic spline function with K knots:

$$f(X) = \beta_0 h_0(X) + \beta_1 h_1(X) + \cdots + \beta_{K+3} h_{K+3}(X),$$

or equivalently

$$Y = \beta_0 h_0(X) + \beta_1 h_1(X) + \cdots + \beta_{K+3} h_{K+3}(X) + \varepsilon,$$

where $\{h_0, h_1, \dots, h_{K+3}\}$ is the truncated-power basis, or any other basis.

- Consider $h_0(X), h_1(X), \dots, h_{K+3}(X)$ as $K + 4$ inputs $X_0, X_1, X_2, \cdots, X_{K+3}$, or $X_i = h_i(X), \;\; i = 0, 1, 2, \dots, K + 3.$

  Then the above model becomes

$$Y = \beta_0 X_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_{K+3} X_{K+3} + \varepsilon.$$

Review of the Regression Spline

$$Y = \beta_0 h_0(X) + \beta_1 h_1(X) + \cdots + \beta_{K+3} h_{K+3}(X) + \varepsilon$$

- $Y = \beta_0 X_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_{K+3} X_{K+3} + \varepsilon,\ X_i = h_i(X),\ i = 0, 1, 2, \dots, K+3.$
- The LSE $\widehat{\boldsymbol{\beta}} = \left(\hat{\beta}_0, \hat{\beta}_1, \cdots, \hat{\beta}_{K+3}\right)^T$ in the above linear regression model is

$$\widehat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\boldsymbol{y},$$

where $\mathbf{X} = \underset{n \times (K+4)}{\begin{pmatrix} h_0(x_1) & h_1(x_1) & \cdots & h_{K+3}(x_1) \\ \vdots & \vdots & & \vdots \\ h_0(x_k) & h_1(x_k) & \cdots & h_{K+3}(x_k) \\ \vdots & \vdots & & \vdots \\ h_0(x_n) & h_1(x_n) & \cdots & h_{K+3}(x_n) \end{pmatrix}}$, $\boldsymbol{y} = \underset{n \times 1}{\begin{pmatrix} y_1 \\ \vdots \\ y_k \\ \vdots \\ y_n \end{pmatrix}}$,

and $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ are the training data.

Assignment 5

自由度4よりK+4から8個の関数がいる

Suppose that $f(x)$ is a cubic spline function with K knots $\frac{10i}{K+1}$, $i = 1, 2, \ldots, K$. We consider the regression problem with the training data given in the “Moodle System.”

1. For $K = 4$, solve the following problems:
   A) Draw the graphs of the all functions in a spline basis which you will use below. If you will use the B-spline functions, the above knots are treated as the interior knots, and the boundary knots are 0 and 10.
   B) Find the cubic spline function $f(x)$ which gives the best fitting to the training data.
   C) Evaluate $\mathrm{CV_{LOO}}$ by using the magic formula $\mathrm{CV_{LOO}} = \frac{1}{n}\sum_{i=1}^{n}\left\{\frac{y_i - \hat{f}(x_i)}{(1-h_{ii})}\right\}^2$.
   D) Evaluate $\mathrm{CV_{LOO}} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}_{-i}(x_i)\right)^2$ without using the magic formula.

(Continued to the next slide)

Assignment 5 (continued)

2. Use $\mathrm{CV}_{\mathrm{LOO}}$ or CV_{10} to determine the number of knots, K, among $1, 2, \dots, 15$.
3. Unlike the polynomial regression, the training MSE does NOT monotonically decrease as the degrees of freedom increase or equivalently as the number of knots, K, increases. What difference between the polynomial and spline regressions induces the phenomenon just mentioned?

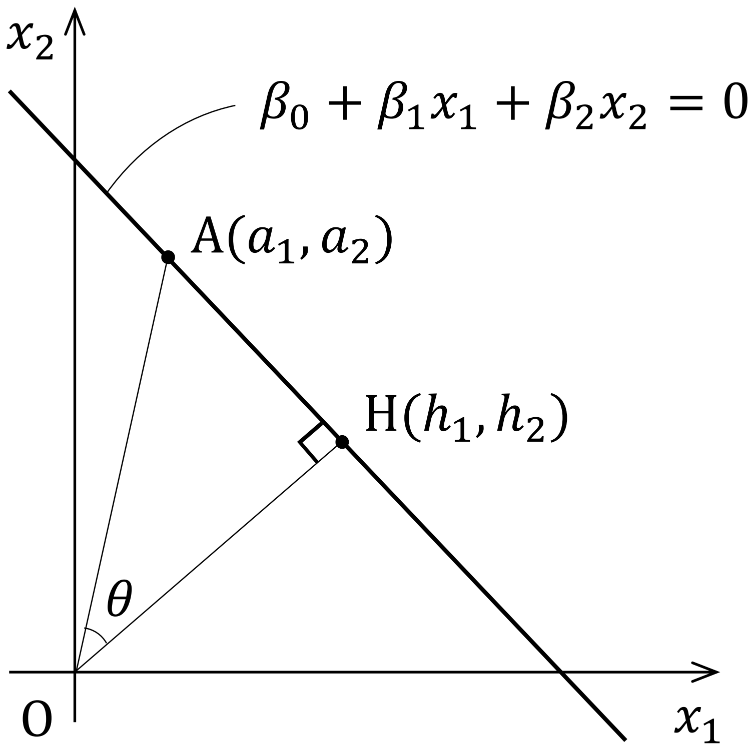
- Submit your answer to the assignment via “Moodle System” by 3:00pm on 6th June.
    - You may use any computer language for the problems 1 and 2.
    - Attach a key part of your source code to your answer, and give detailed explanations about your source code.

An extra note: The training data and the true model are the same ones in the example for the polynomial regression given in today’s class.

Homework

We consider two points A and H on a line $\beta_0 + \beta_1 x_1 + \beta_2 x_2 = 0$, and suppose the segment OH is perpendicular to the line, where O is the origin. We denote the size of the angle $\angle\mathrm{AOH}$ by θ. Solve the following problems.

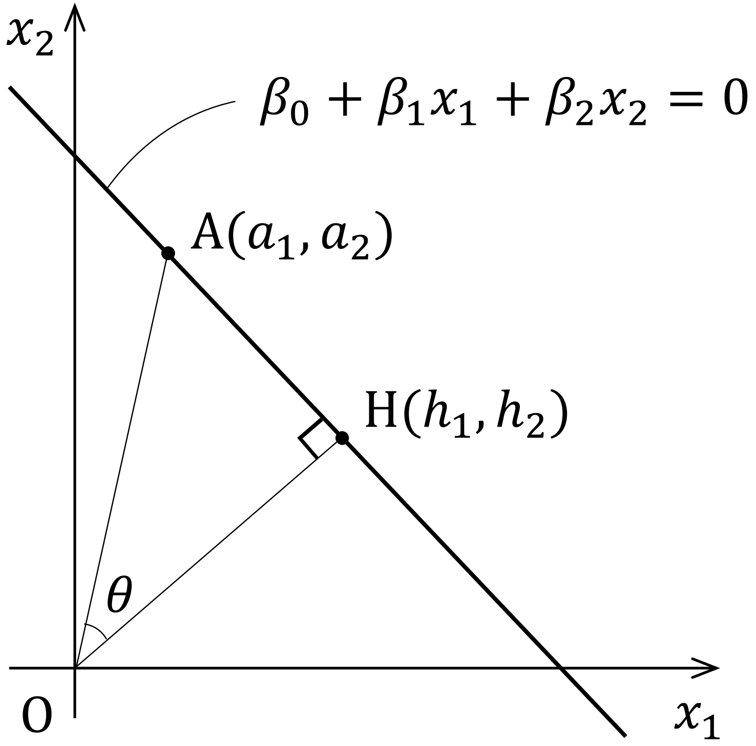
1. Find the value of the inner product $\boldsymbol{\beta} \cdot \overrightarrow{\mathrm{OA}}$, where $\boldsymbol{\beta} = \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix}$ and $\overrightarrow{\mathrm{OA}} = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}$. Note that the value of $\boldsymbol{\beta} \cdot \overrightarrow{\mathrm{OA}}$ is independent of a_1 and a_2.
2. Find the value of the inner product $\boldsymbol{\beta} \cdot \overrightarrow{\mathrm{OH}}$.
3. Find the value of $\boldsymbol{\beta} \cdot \overrightarrow{\mathrm{AH}}$. Answer a geometrical relation between $\boldsymbol{\beta}$ and $\overrightarrow{\mathrm{AH}}$.



(Continued to the next slide)

Homework (continued)

4. Express $|\mathrm{OH}|$ in terms of $|\mathrm{OA}|$ and θ, where $|\mathrm{OH}|$ is the length of the segment OH.
5. Express $\boldsymbol{\beta} \cdot \overrightarrow{\mathrm{OA}}$ in terms of $\|\boldsymbol{\beta}\|$, $|\mathrm{OA}|$ and θ, where $\|\boldsymbol{\beta}\|$ is the norm of $\boldsymbol{\beta}$.
6. Express $|\mathrm{OH}|$ in terms of β_0, β_1 and β_2.



You need not submit the homework but the above problems give the key idea to understand the classification described in the next class.