Module 2, Part 1: Statistical Learning
TMA4268 Statistical Learning V2023

Daesoo Lee,

Department of Mathematical Sciences, NTNU

NTNU
Norwegian University of
Science and Technology

16/01/2023

Acknowledgements

Acknowledgements

- A lot of this material stems from Mette Langaas and her TAs (especiall Julia Debik). I would like to thank Mette for the permission to use her material!

Introduction

Introduction

Learning material for this module

- James et al (2013): An Introduction to Statistical Learning. Chapter 2 (except 2.2.3).
- Additional material (in this module page) on *random variables*, *covariance matrix* and the *multivariate normal distribution* (known for students who have taken TMA4267 Linear statistical models).

Introduction

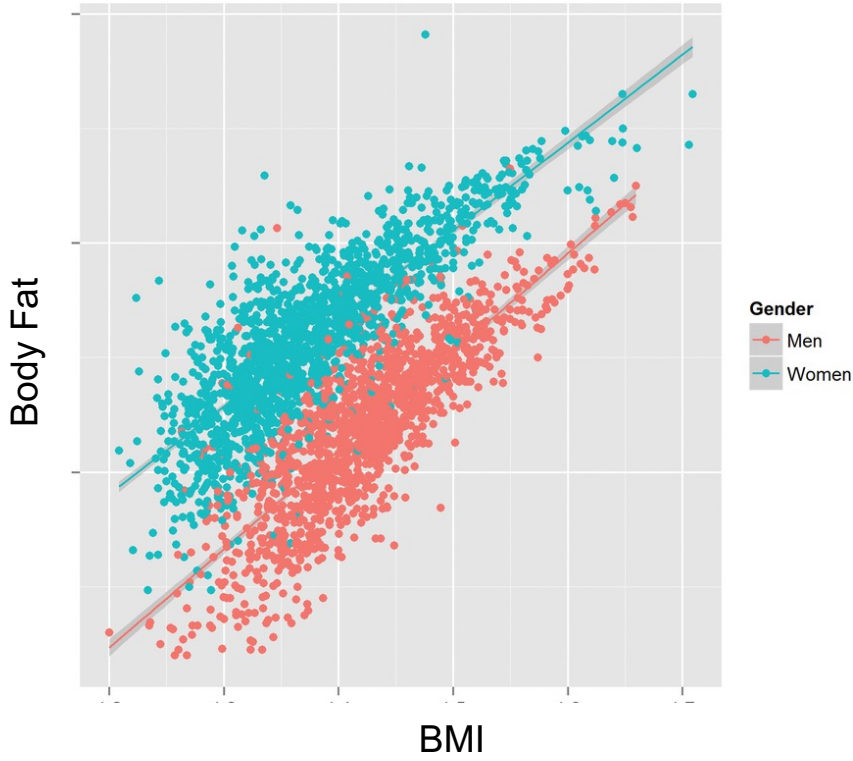
What will you learn?

- Statistical learning and examples
- Introduce relevant notation and terminology
- Prediction accuracy vs. model interpretability
- Bias-variance trade-off

What is statistical learning?

What is statistical learning?

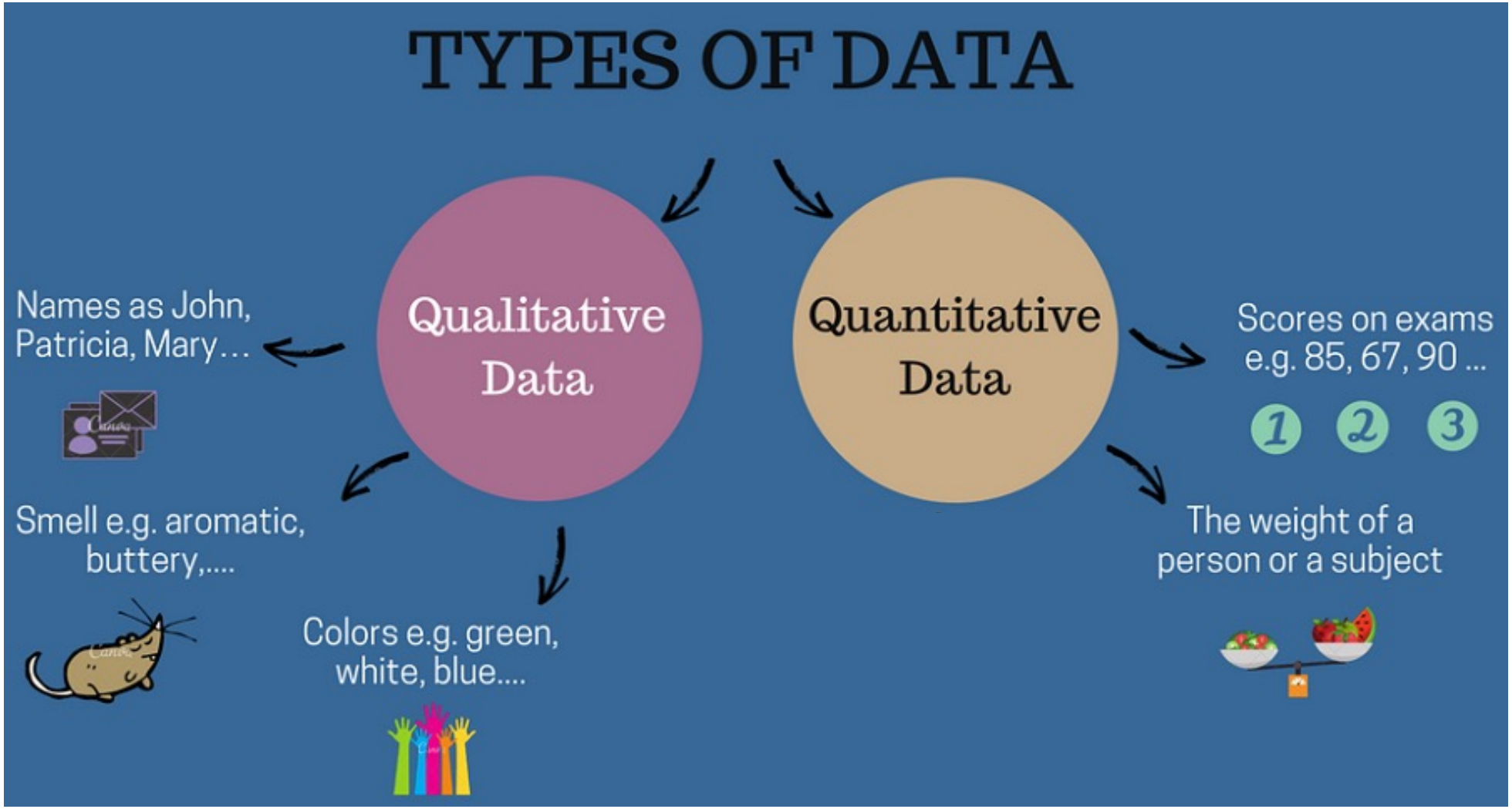
- *Statistical learning* is the process of learning from data. We would like to
  - draw conclusions about the relations between the variables (inference) or
  - find a predictive function for new observations (prediction).
- Want to find structures in the data
  that help us to learn something about the real world.
- Plays a key role in many areas of science, finance and industry.
- A fundamental ingredient in the training of a modern data scientist.



Fuster-Parra, Pilar, et al. "A comparison between multiple regression models and CUN-BAE equation to predict body fat in adults." PloS one 10.3 (2015): e0122291.

What is statistical learning?

Two variable types



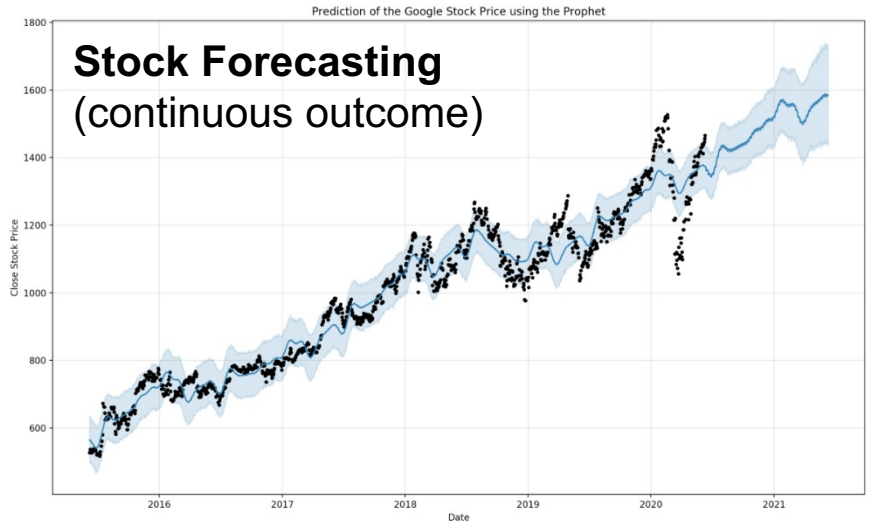
What is statistical learning?

Two variable types

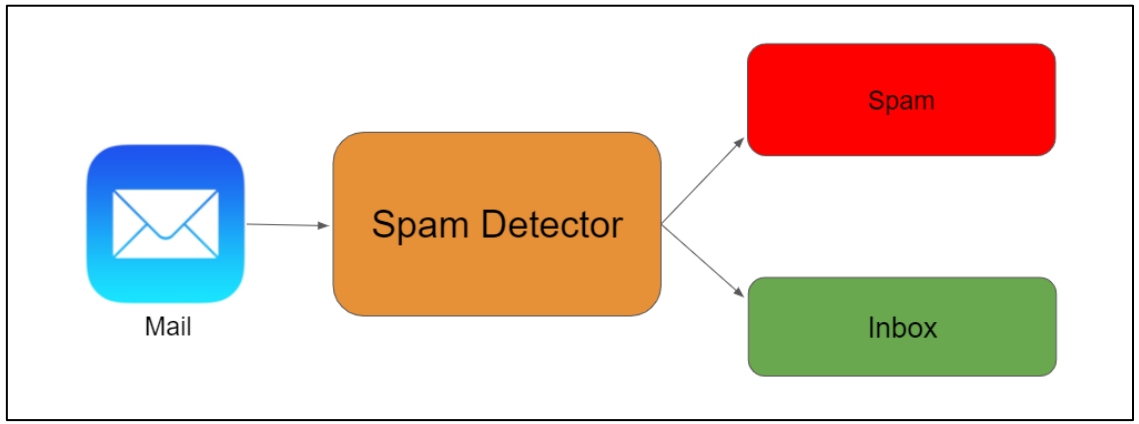
- ***Quantitative*** variables are variables from a continuous set, they have a numerical value.
  - Examples: a person's weight, a company's income, the age of a building, the temperature outside, the level of precipitation etc.
- ***Qualitative*** variables are variables from a discrete set with K different classes/labels/categories.
  - Examples: type of fruit {apples, oranges, bananas, …}, sex {male, female, other }, education level {none, low, medium, high}.
  - Qualitative variables which have only two classes are called binary variables and are usually coded by 0 (no) and 1 (yes).

Examples of learning problems

Examples of learning problems



Spam Detector
(binary outcome)



- Identification of risk factors for Prostate cancer (yes/no).
- Estimating the risk of heart disease, given knowledge about condition, behaviour, age, or demographic, diet and clinical measurements.

Examples of learning problems

Example 1: Handwritten digit recognition

- Classification problem with categorical response variable {0, 1, 2, …, 9}.

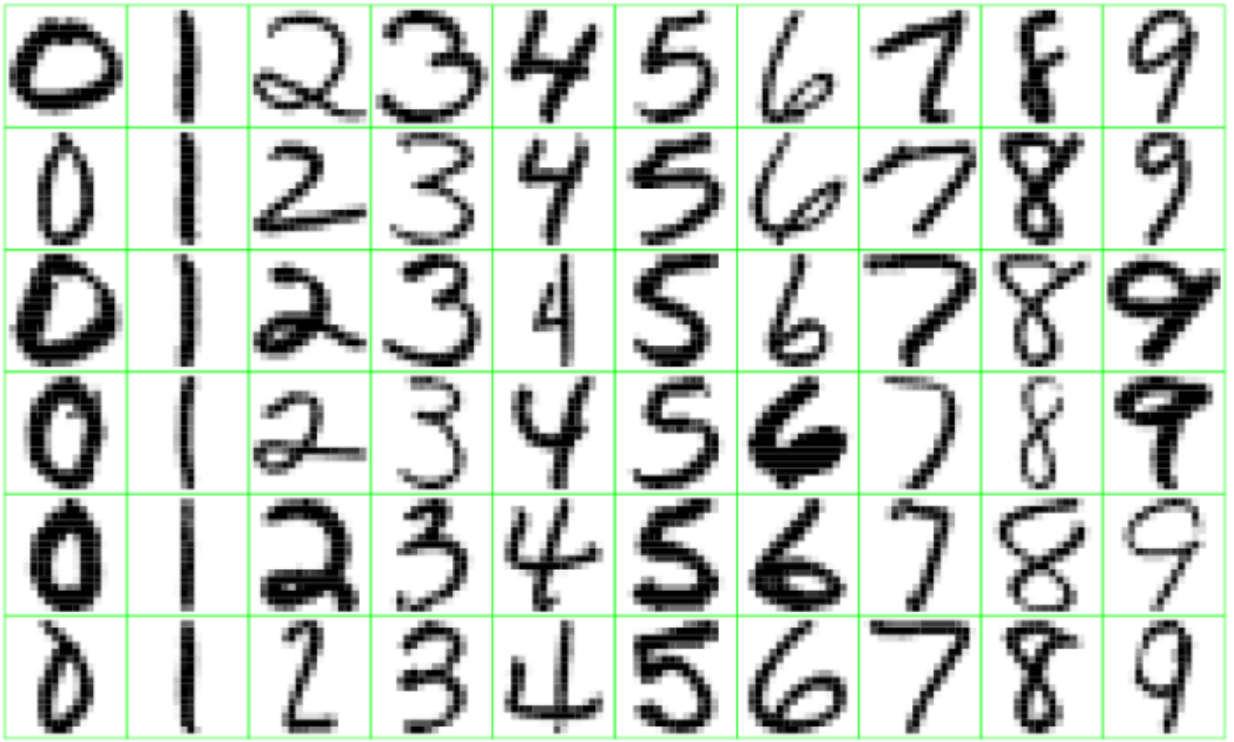


Image taken from https://web.stanford.edu/~hastie/ElemStatLearnII/

Examples of learning problems

Example 2: Email classification (spam detection)

- The table below shows the average percentage of words or characters in an email message, based on 4601 emails of which 1813 were classified as a spam.

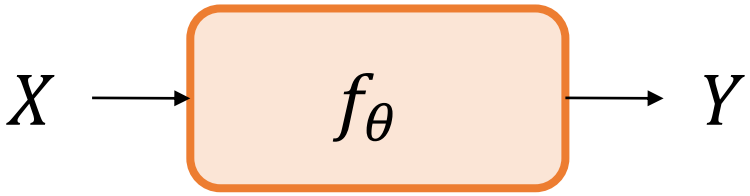
	you	free	george	!	$	edu
not spam	1.27	0.07	1.27	0.11	0.01	0.29
spam	2.26	0.52	0.00	0.51	0.17	0.01

Fig. Average percentage of words or characters in 4601 emails.

- The classification model can be based on the frequencies of words in emails.

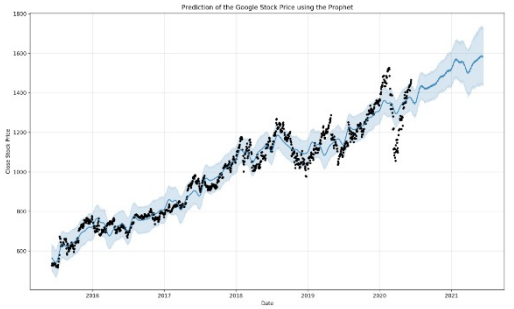
The Supervised Learning Problem

The Supervised Learning Problem

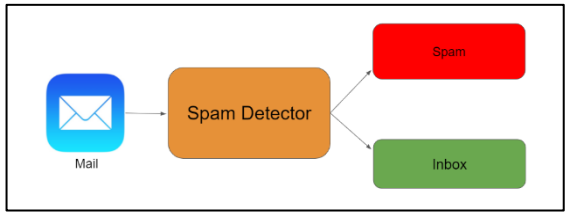


Some learnable function.
e.g., $Y = \beta_1 X$

Quantitative; Regression problem (*e.g.*, 1820)



Qualitative; Classification problem

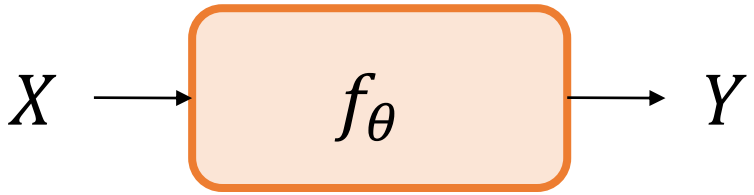


- X is called inputs, regressors, covariates, features, independent variables.
  - $X = \{X_1, \cdots, X_p\}$; *e.g.*, $\{age, \cdots, weight\}$
- Y is called dependent variables, response, target.
  - Y is ***quantitative*** in the **regression problem** (*e.g.*, stock price).
  - Y is ***qualitative*** in the **classification problem** (*e.g.*, survived/died, digit label 0-9).

The Supervised Learning Problem

Supervised learning and its objectives

Training Scheme



p covariates

n samples

x_{11}	x_{12}	$\cdots$	x_{1p}
x_{21}	x_{22}	$\cdots$	x_{2p}
$\vdots$	$\vdots$	$\vdots$	$\vdots$
x_{n1}	x_{n2}	$\cdots$	x_{np}

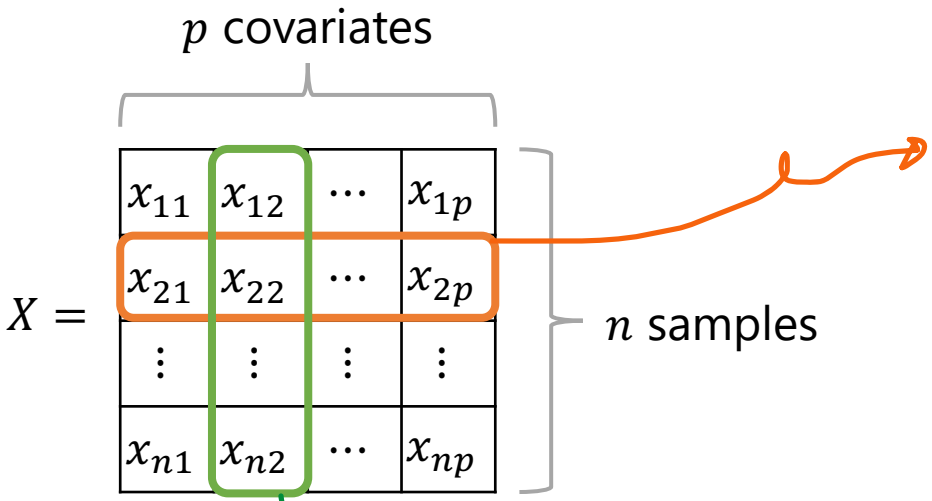
y_1
y_2
$\vdots$
y_n

After training, we'd like to:

- *Accurately predict* unseen test cases.
- *Understand* which covariate(s) affects the output.
- *Assess the quality* of our predictions and inference.

Notation and key statistical concepts

Notation and key statistical concepts



$$x_i = \begin{pmatrix} x_{i1} \\ x_{i2} \\ x_{i3} \\ \vdots \\ x_{ip} \end{pmatrix}$$

$$X_j = \begin{pmatrix} x_{1j} \\ x_{2j} \\ x_{3j} \\ \vdots \\ x_{nj} \end{pmatrix}$$

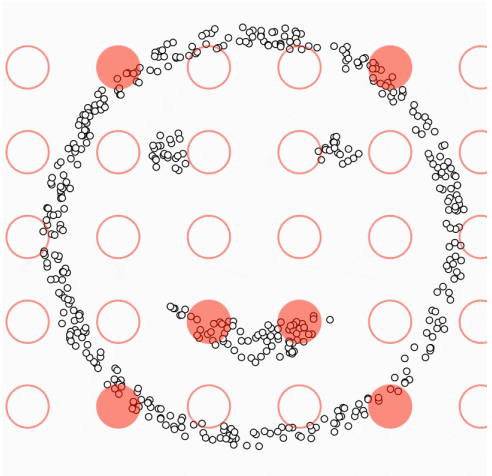
$$X = (X_1, X_2, \cdots, X_p)$$

$$Y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

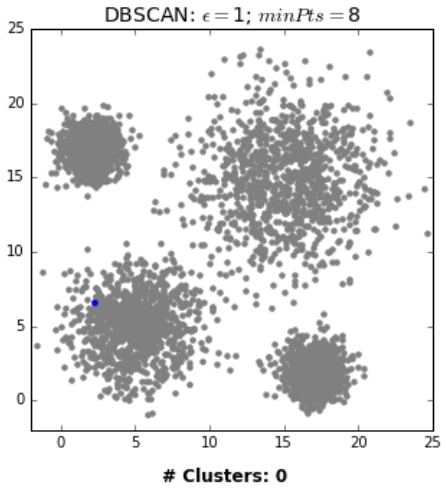
The Unsupervised Learning Problem

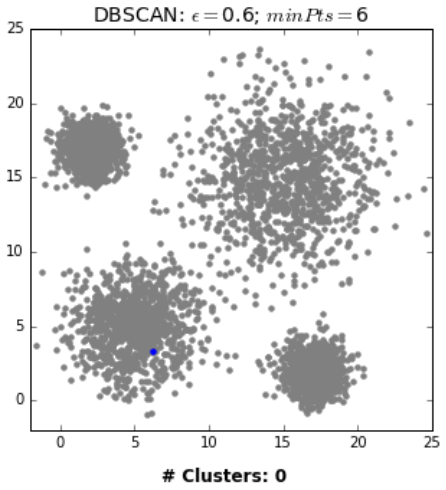
The Unsupervised Learning Problem

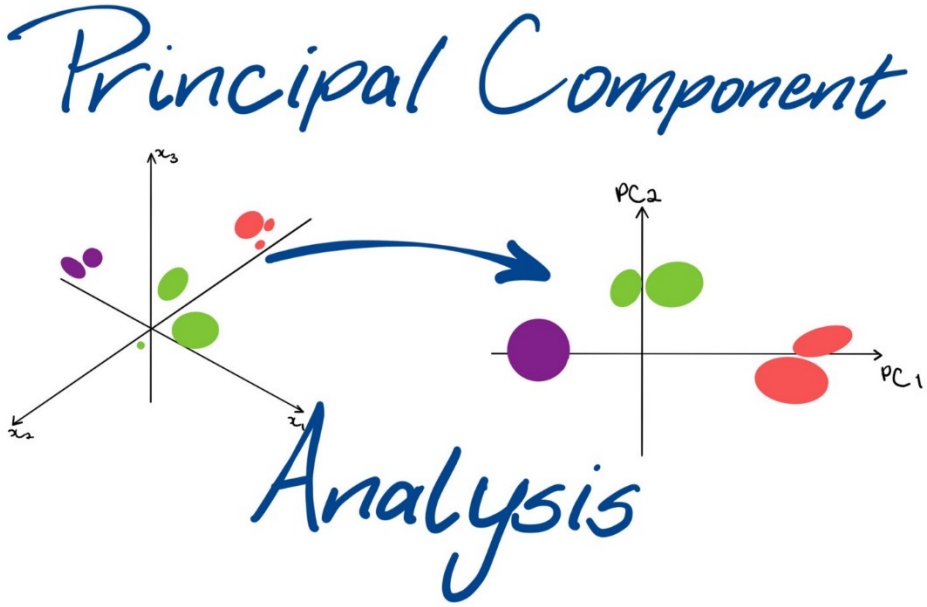
- There is **no outcome variable** y, just a set of predictors (features/covariates) x_i.
- Objective is more fuzzy
  - Find (hidden) patterns or grouping (*i.e.,* clustering) in the data – *to gain insight and understanding*.
  - There is no *correct* answer.
  - Difficult to know how well you are doing.
- Examples:



DBSCAN



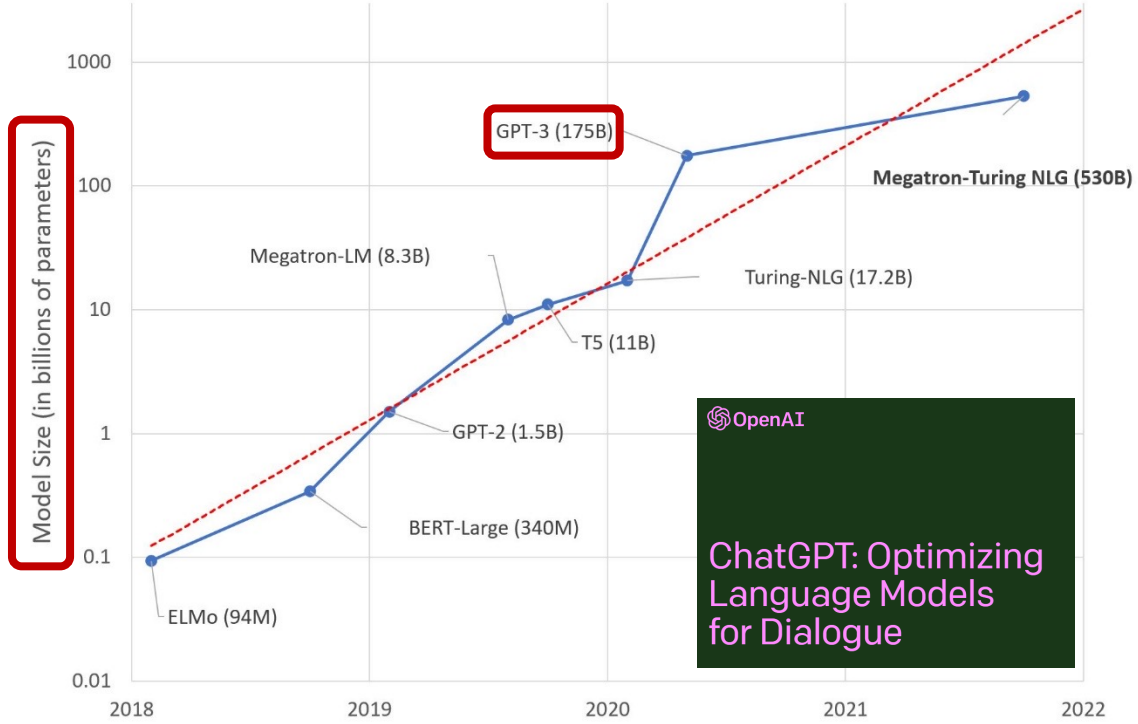




Overall philosophy

Overall philosophy

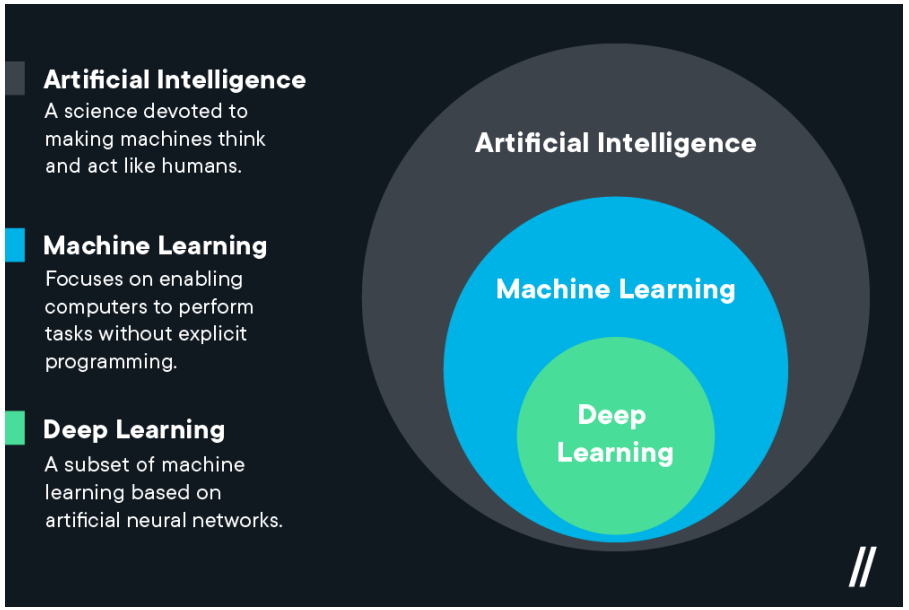
- Important to understand the simpler methods first, in order to grasp the more sophisticated ones.
- **Simpler methods often perform as well as fancier ones!**
- It is important to accurately *assess the performance of a method*, to know how well or how badly it is working.



Statistical Learning vs. Machine Learning

Statistical Learning vs. Machine Learning

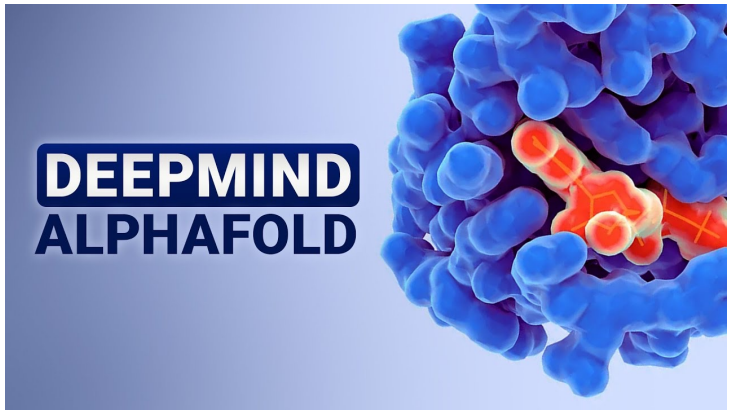
- Machine learning arose as a subfield of Artificial Intelligence.
- Statistical learning arose as a subfield of Statistics.

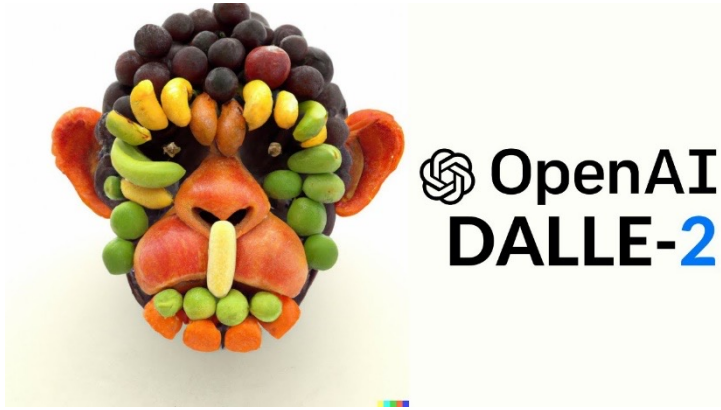


- There is much overlap -- both fields focus on supervised and unsupervised problems:
  - *Machine learning* has a greater emphasis on *large scale applications and prediction accuracy*.
    - NB! Explainable AI is a current hot topic though!
  - *Statistical learning* emphasizes models and their *interpretability, and precision and uncertainty*.
- The distinction has become more and more blurred, and there is a great deal of "cross-fertilization".
- Machine learning has the upper hand in Marketing!

Statistical Learning vs. Machine Learning

- There is a controversy and some skepticism against ``too fancy'' ML methods.
- Criticism:
  - ML often re-invents existing methods and names them differently, but often without awareness of existing methods in statistics.
  - Despite the record-breaking performances by Deep Learning, the models are often not fully understood.
- Yet, the progress has been remarkable:





Statistical Learning vs. Machine Learning

Deep Learning is like..

← Large ML method (i.e., deep learning) is like

wildly-untamed-yet-strong horse

Statistical Learning vs. Machine Learning

What is the aim in statistical learning?

- We are talking about supervised methods now. Assume:
  - We observe one *quantitative* response Y and p different predictors $x_1, x_2, \dots, x_p$.
  - We assume that there is a function f that relates the response and the predictor variables:

$$Y = f(x) + \epsilon$$

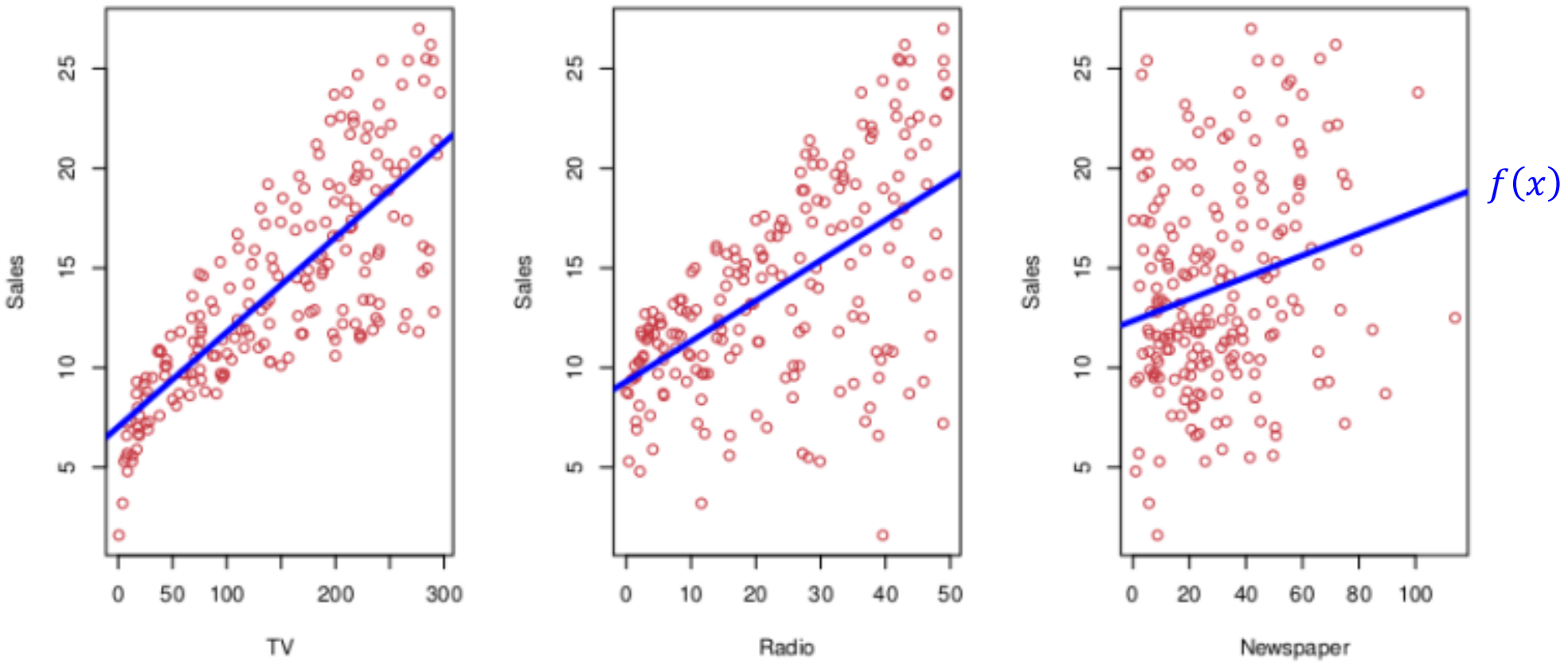
where ϵ is a random error term with mean 0 and independent of x.

- **The aim is to estimate f**

Examples of Statistical Learning

Examples of Statistical Learning

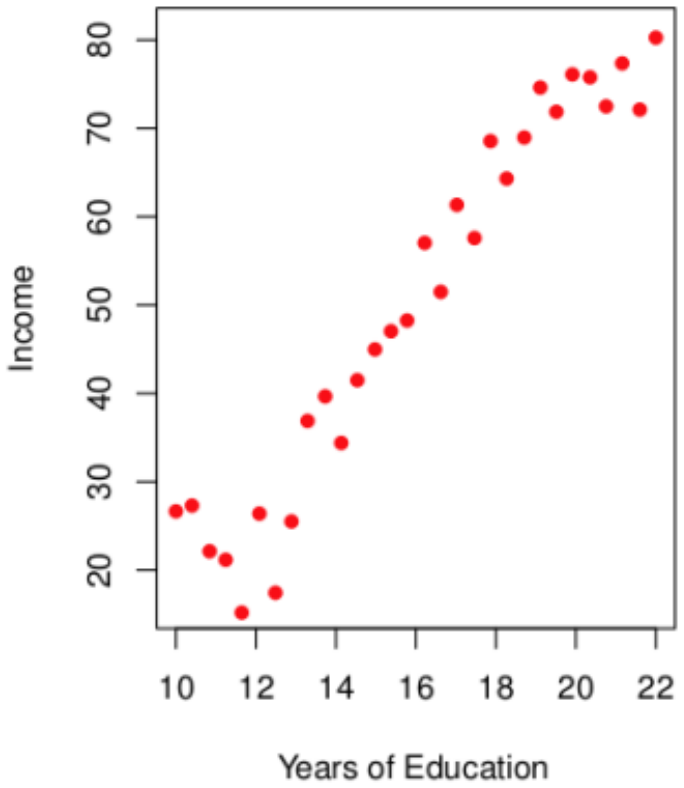
Example1: Sales of a product, given advertising budgets in different media

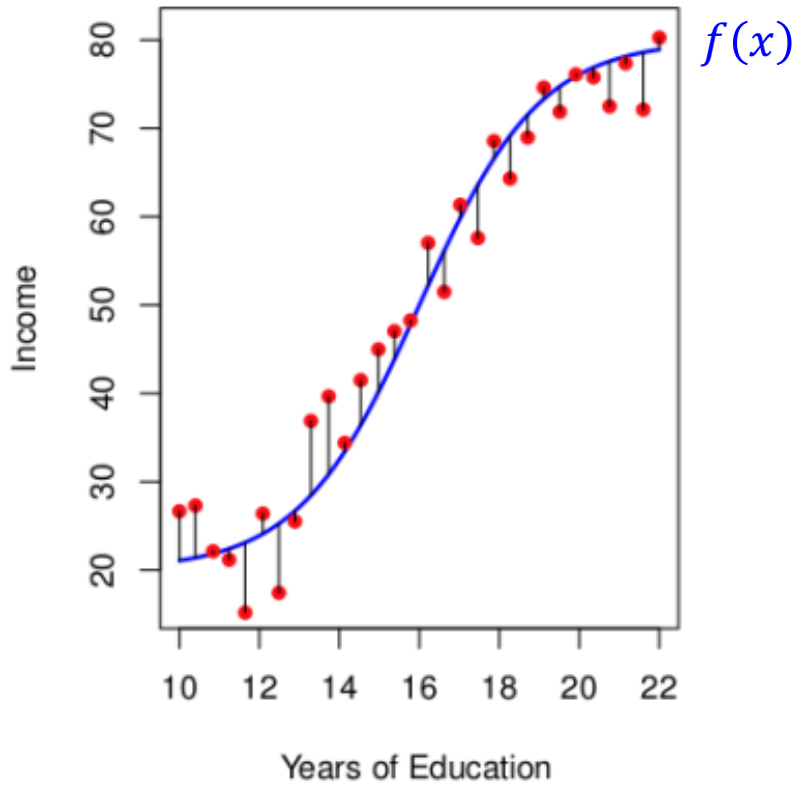


$f(x)$

Examples of Statistical Learning

Example 2: Income for given levels of education





Two Reasons for Estimating f

Two Reasons for Estimating f

- Reason1: Prediction
- Reason2: Inference

Two Reasons for Estimating f

Reason 1: Prediction

- **Aim:** predict a response Y given new observations x as accurately as possible.
- Notation:

$$\hat{Y} = \hat{f}(x)$$

- $\hat{f}$: estimated f
- $\hat{Y}$: prediction for Y given x.
- We *do not really care* about the shape of f ("black box").
  → No interpretation of regression parameters when the aim is purely prediction!

Two Reasons for Estimating f

Two quantities which influence accuracy of $\hat{Y}$ (= prediction of Y)

- The reducible error has to do with our estimate $\hat{f}$ of f.
  This error can be reduced by using the most *appropriate* statistical learning technique.
- The irreducible error comes from the error term ϵ and cannot be reduced by improving f.
  This is related to the unobserved quantities influencing the response and possibly the randomness of the situation.
- For a given $\hat{f}$ and a set of predictors X which gives $\hat{Y} = \hat{f}(x)$, we have

$$\mathrm{E}\left[\left(Y - \hat{Y}\right)^2\right] = \underbrace{\left(\mathrm{E}\left[f(X) - \hat{f}(X)\right]\right)^2}_{\text{model bias}} + \underbrace{\mathrm{Var}\left(\hat{f}(X)\right)}_{\text{model variance}} + \underbrace{\mathrm{Var}(\epsilon)}_{\text{variance of irreducible error}}$$

model bias
reducible error

model variance

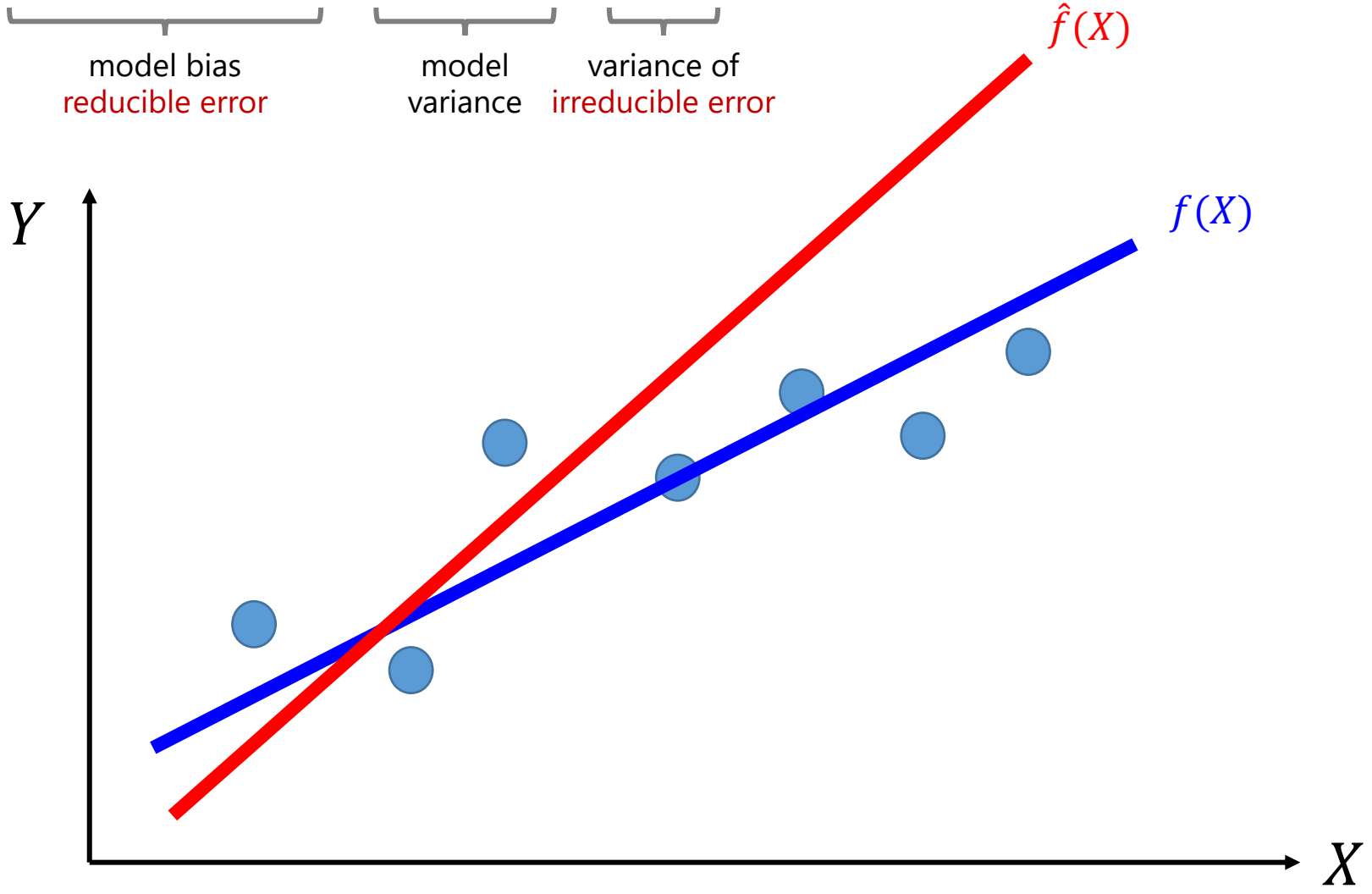
variance of
irreducible error

Two Reasons for Estimating f

$$\mathrm{E}\left[\left(Y-\hat{Y}\right)^2\right] = \underbrace{\left(\mathrm{E}\left[f(X)-\hat{f}(X)\right]\right)^2}_{\text{model bias}} + \underbrace{\mathrm{Var}\left(\hat{f}(X)\right)}_{\text{model variance}} + \underbrace{\mathrm{Var}(\epsilon)}_{\text{variance of}}$$

reducible error

irreducible error

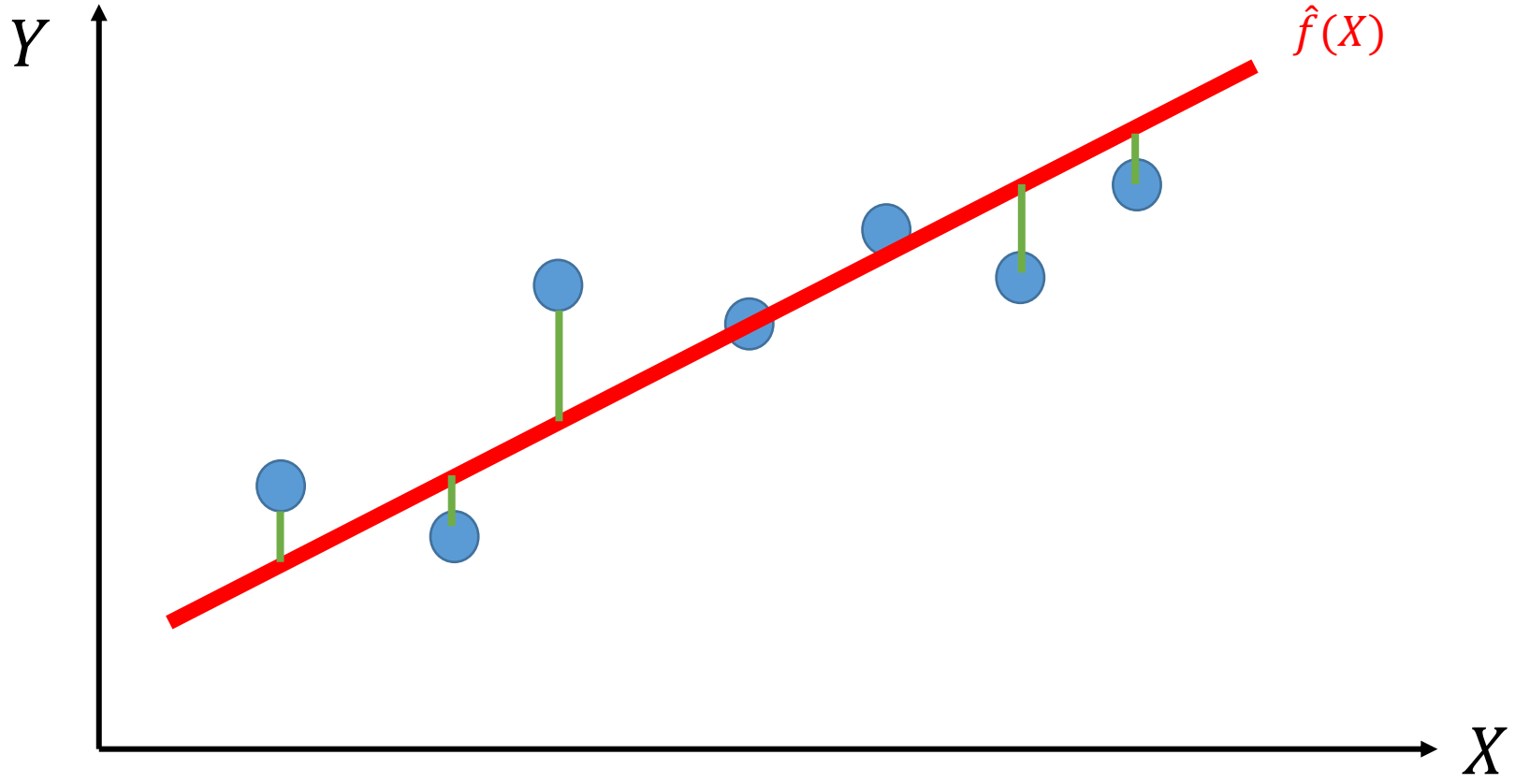


Two Reasons for Estimating f

$$\mathrm{E}\left[\left(Y-\hat{Y}\right)^2\right]=\underbrace{\left(\mathrm{E}\left[f(X)-\hat{f}(X)\right]\right)^2}_{\text{model bias}}+\underbrace{\mathrm{Var}\left(\hat{f}(X)\right)}_{\text{model variance}}+\underbrace{\mathrm{Var}(\epsilon)}_{\text{variance of}}$$

reducible error

irreducible error



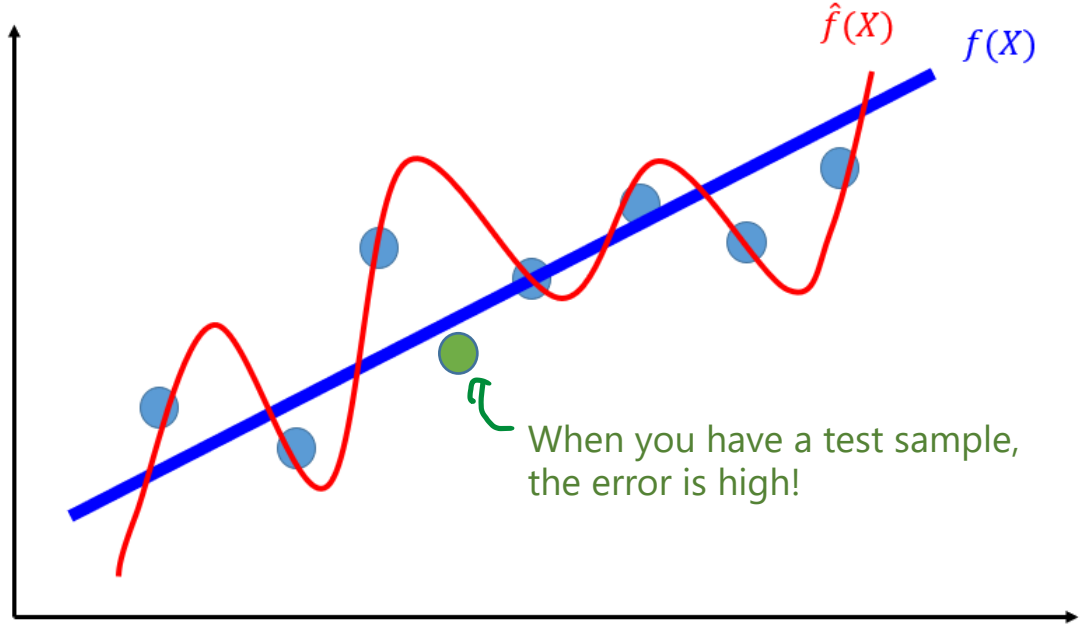
Two Reasons for Estimating f

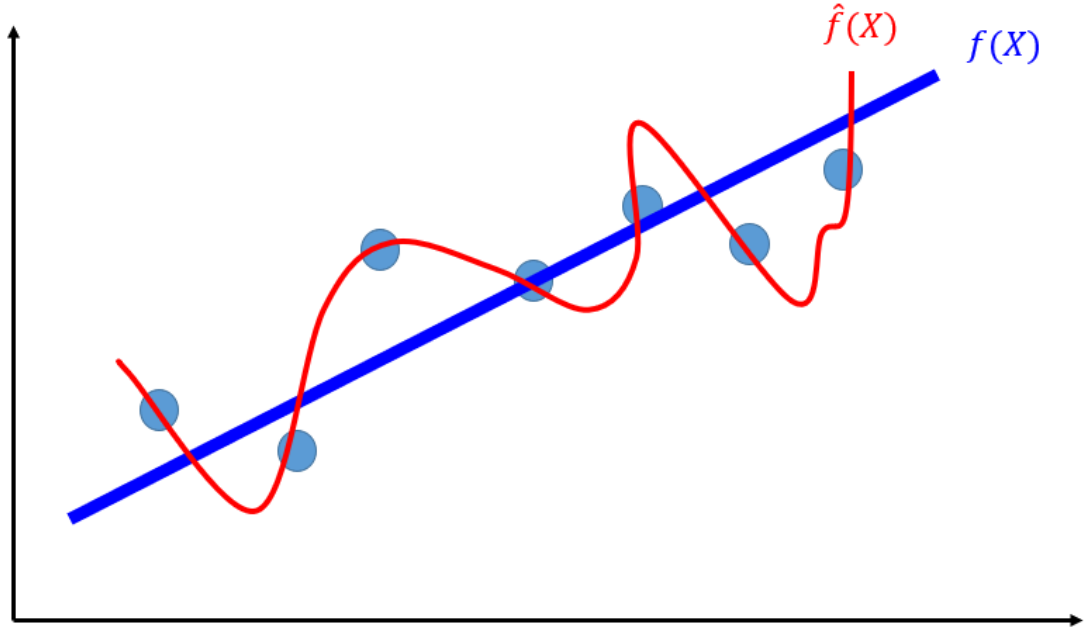
$$\mathrm{E}\left[\left(Y-\hat{Y}\right)^2\right] = \underbrace{\left(\mathrm{E}\left[f(X)-\hat{f}(X)\right]\right)^2}_{\text{model bias}} + \underbrace{\mathrm{Var}\left(\hat{f}(X)\right)}_{\text{model variance}} + \underbrace{\mathrm{Var}(\epsilon)}_{\text{variance of}}$$

reducible error

irreducible error

Why reducing it?





Two Reasons for Estimating f

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

$$= \left(\mathrm{E}\left[Y-\hat{Y}\right]\right)^2 + \mathrm{Var}\left(Y-\hat{Y}\right)$$

$\mathrm{Var}(X) = \mathrm{E}(X^2) - \left(\mathrm{E}(X)\right)^2$

$\rightarrow \mathrm{E}(X^2) = \left(\mathrm{E}(X)\right)^2 + \mathrm{Var}(X)$

$$= \left(\mathrm{E}\left[f(X)+\epsilon-\hat{f}(X)\right]\right)^2 + \mathrm{Var}\left(f(X)+\epsilon-\hat{f}(X)\right)$$

$Y = f(X) + \epsilon$

$\hat{Y} = \hat{f}(X)$

$$= \left(\mathrm{E}\left[f(X)-\hat{f}(X)\right] + \mathrm{E}[\epsilon]\right)^2 + \mathrm{Var}\left(f(X)-\hat{f}(X)\right) + \mathrm{Var}(\epsilon)$$

ϵ is a random error term with mean 0 and *independent* of X

$$= \left(\mathrm{E}\left[f(X)-\hat{f}(X)\right]\right)^2 + 2\mathrm{E}\left[f(X)-\hat{f}(X)\right]\mathrm{E}[\epsilon] + (\mathrm{E}[\epsilon])^2 + \mathrm{Var}\left(f(X)-\hat{f}(X)\right) + \mathrm{Var}(\epsilon)$$

$$= \left(\mathrm{E}\left[f(X)-\hat{f}(X)\right]\right)^2 + \mathrm{Var}\left(f(X)-\hat{f}(X)\right) + \mathrm{Var}(\epsilon)$$

$\mathrm{E}(\epsilon) = 0$

$$= \left(\mathrm{E}\left[f(X)-\hat{f}(X)\right]\right)^2 + \mathrm{Var}(f(X)) + \mathrm{Var}\left(\hat{f}(X)\right) + \mathrm{Var}(\epsilon)$$

$f(X)$ is fixed and doesn't vary.

$$= \underbrace{\left(\mathrm{E}\left[f(X)-\hat{f}(X)\right]\right)^2}_{\text{model bias}} + \underbrace{\mathrm{Var}\left(\hat{f}(X)\right)}_{\text{model variance}} + \underbrace{\mathrm{Var}(\epsilon)}_{\text{variance of}}$$

reducible error

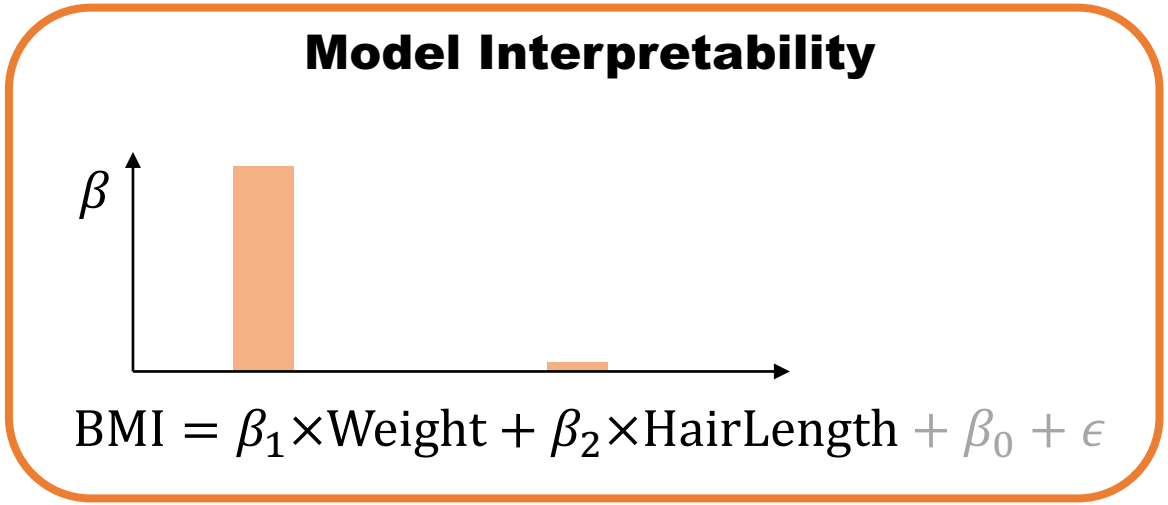
irreducible error

$f(X)$ is fixed and doesn't vary. $\mathrm{Var}\left(f(X)\right) = 0$

Two Reasons for Estimating f

Reason 2: Inference

- **Aim:** understand *how* the response variable is affected by the various predictors (covariates).

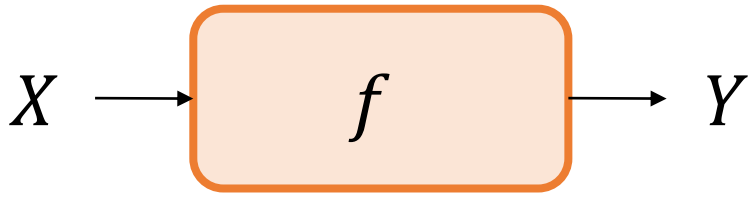


- The exact form of $\hat{f}$ is the *main interest*.
  - What's the relationship between the response and each predictor?
  - Can the relationship be linear? Or non-linearity is needed?

Estimating f

Estimating f

Overall Idea



p covariates

n samples

$$\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix} \quad \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

Two main approaches

- Parametric methods
- Non-parametric methods

Estimating f

Parametric Methods

- They are basically assumption about the shape of f.
- The multiple linear model is an example of a parametric method:

$$f(x) = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p + \epsilon$$

where $\epsilon \sim N(0, \sigma^2)$, β_0 is an intercept, $\beta_{i;i\in\{1,\cdots,p\}}$ is a covariate.

- The task simplifies to finding estimates of the $p + 1$ cofficients $\beta_0, \beta_1, \cdots, \beta_p$.
  To do this, we use the training data to fit the model, such that

$$Y \approx \hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots + \hat{\beta}_p x_p$$

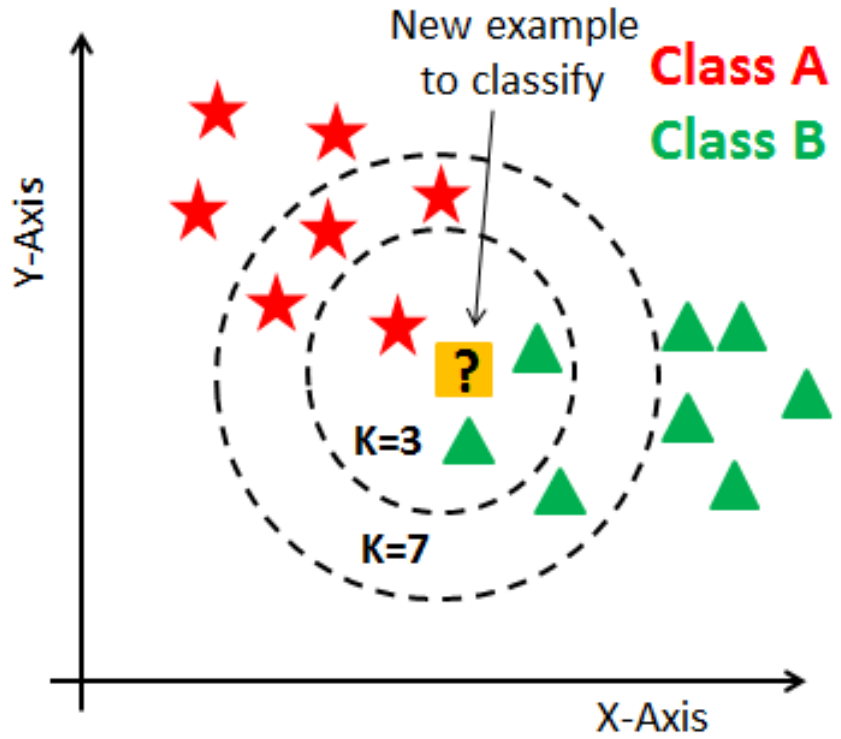
Estimating f

- Fitting a parametric model is thus done in two steps:
  - 1. Select a form for the function f.
  - 2. Estimate the unknown parameters in f using the training set.

Estimating f

Non-Parametric methods

- Non-parametric methods *seek an estimate of* f that gets close to the data points, but *without making explicit assumptions about the form of the function* f.
- Example: K-nearest neighbour (KNN) algorithm.

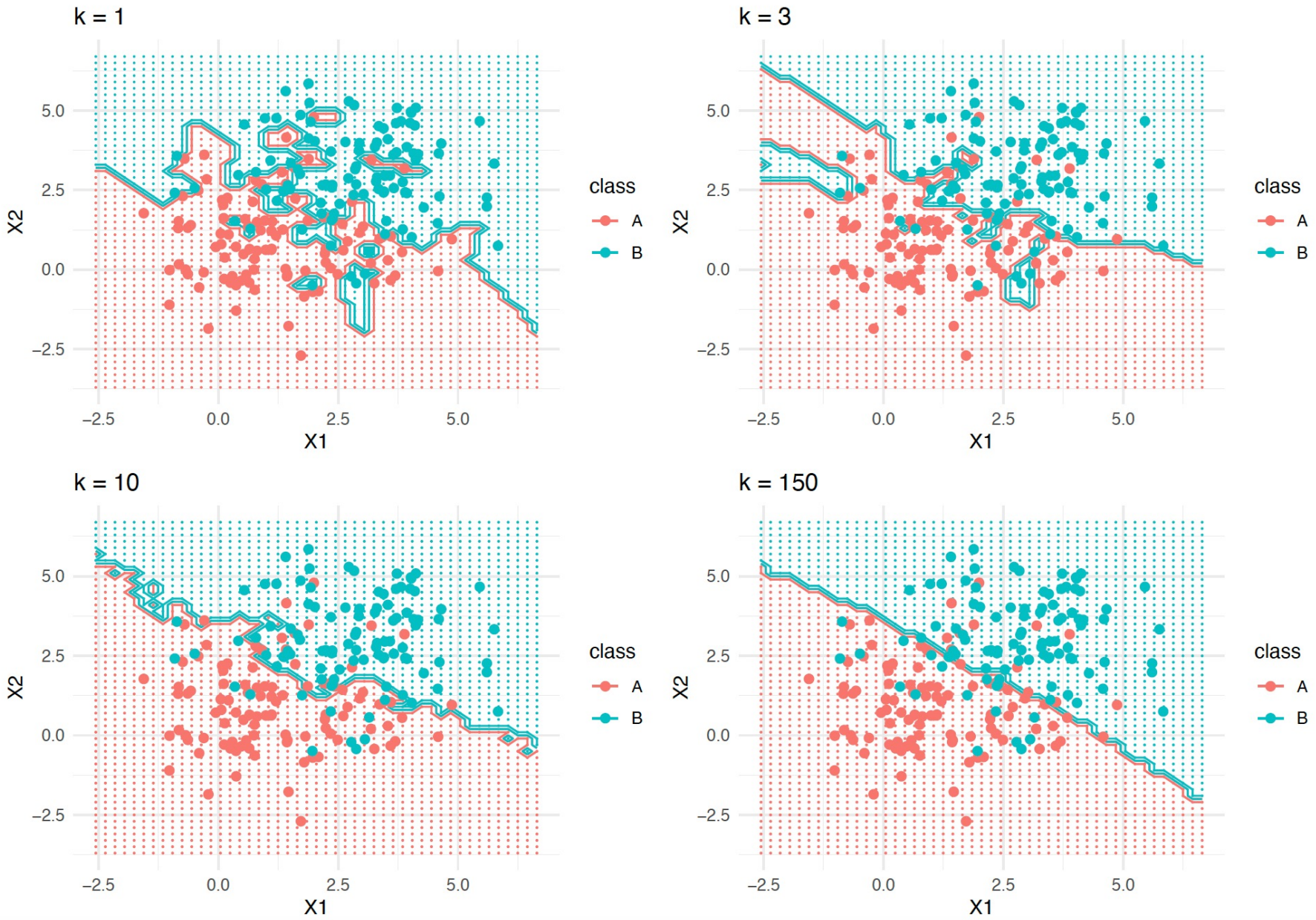


We'll learn the KNN more in detail in Module 4.

Estimating f

KNN

- Big dots: training data
- Little dots: test data



Estimating f

Parametric methods vs Non-Parametric methods

Parametric methods

Advantages	Disadvantages
Simple to use and easy to understand	The function f is constrained to the specified form.
Requires little training data	The assumed function form of f will in general not match the true function, potentially giving a poor estimate.
Computationally cheap	Limited flexibility

NB!
We're talking about conventional statistical learning methods like linear models. It's not the case for deep learning methods.

Non-Parametric methods

Advantages	Disadvantages
Flexible: a large number of functional forms can be fitted	Can overfit the data
No strong assumptions about the underlying function are made	Computationally more expensive as more parameters need to be estimated
Can often give good predictions	Much data are required to estimate (the complex) f.

Think about KNN.

Prediction Accuracy vs Interpretability

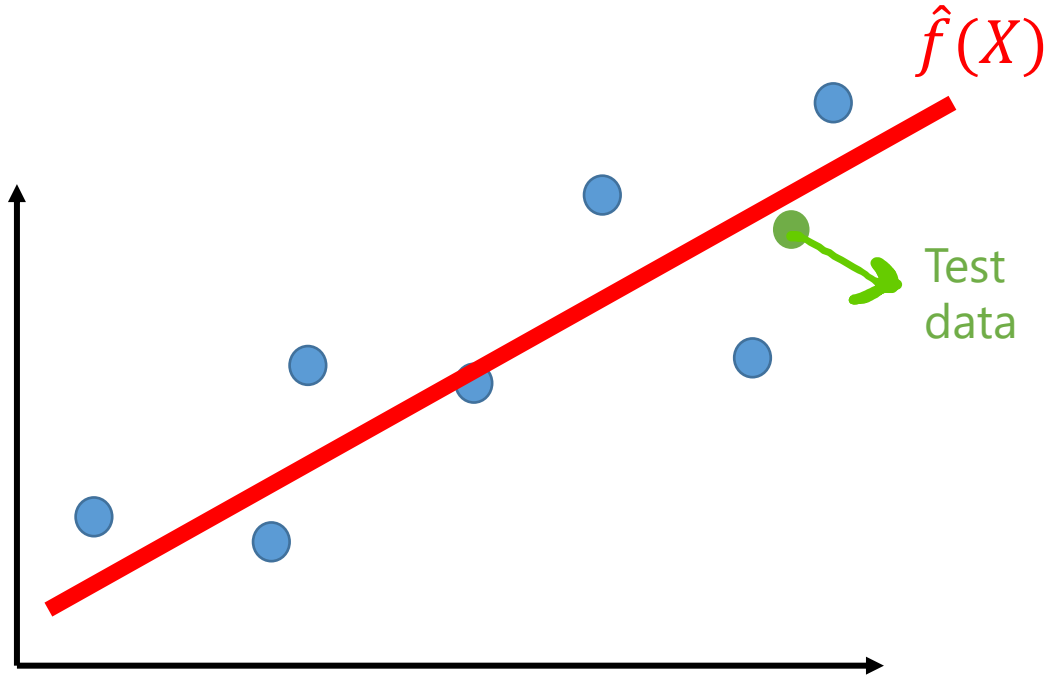
Prediction Accuracy vs Interpretability

- **Inflexible** methods:
  - Linear regression (M3)
  - Linear Discriminant Analysis (M4)
  - Subset Selection and Lasso (M6)

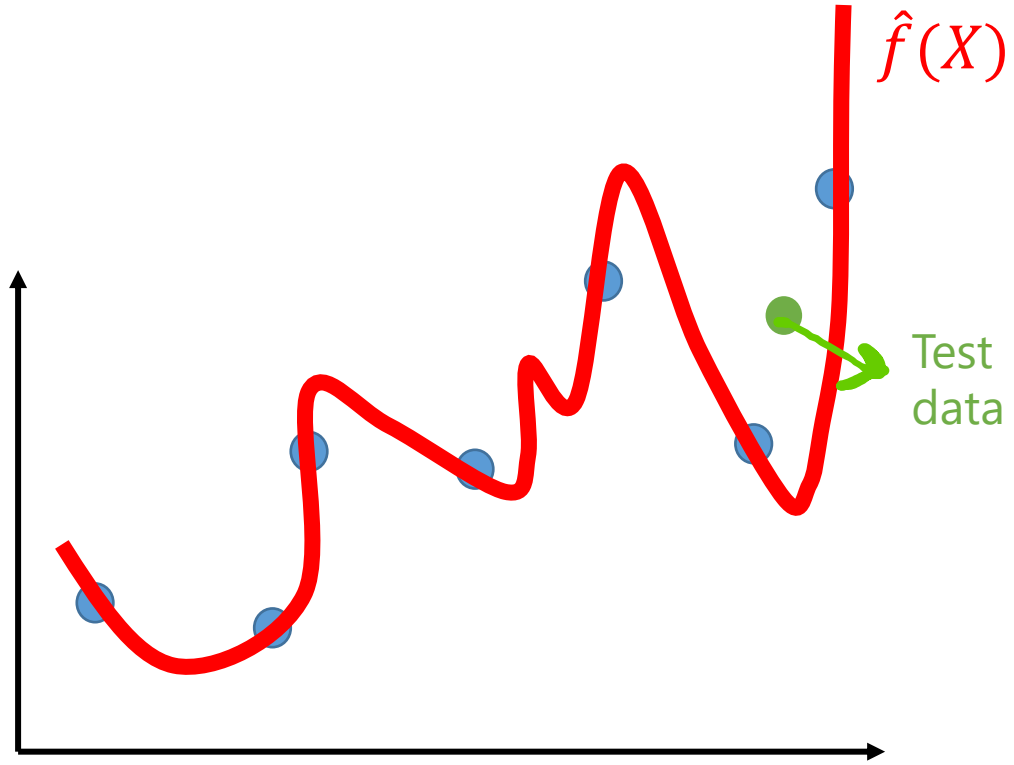
- **Flexible methods:**
  - KNN classification (M4), KNN regression, Smoothing splines (M7).
  - Bagging and Boosting (M8), Support Vector Machines (M9)
  - Neural Networks (M11)

Prediction Accuracy vs Interpretability

Why would I ever prefer an inflexible methods?



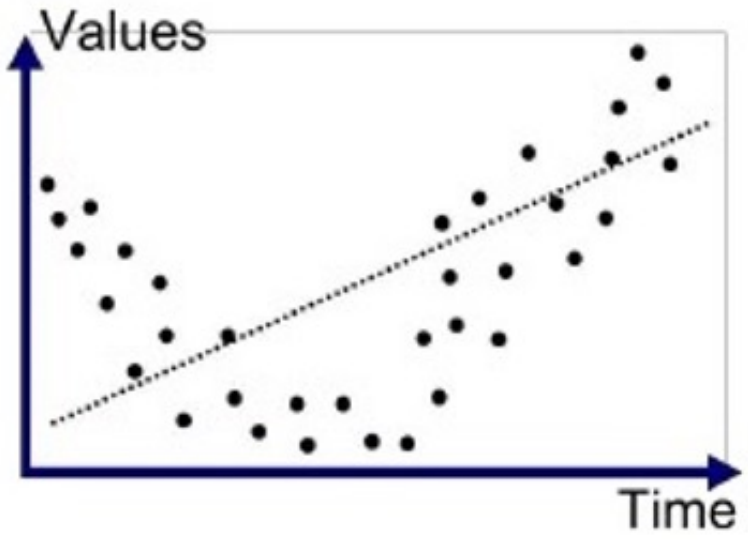
Inflexible



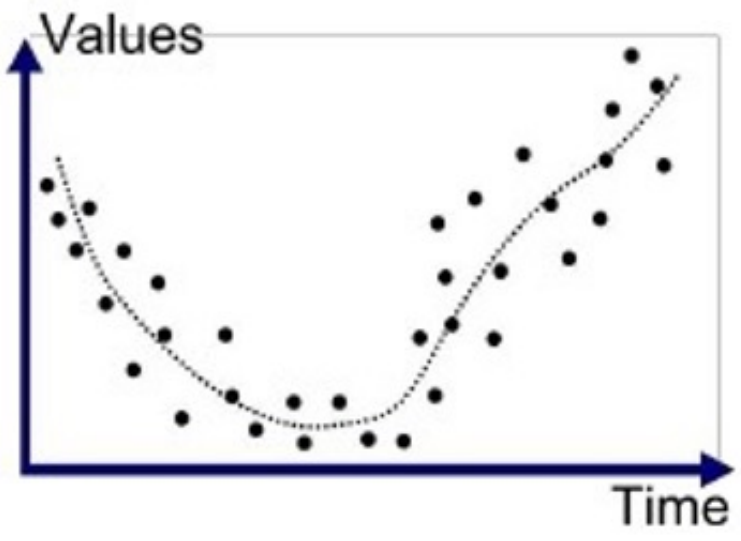
Flexible

Prediction Accuracy vs Interpretability

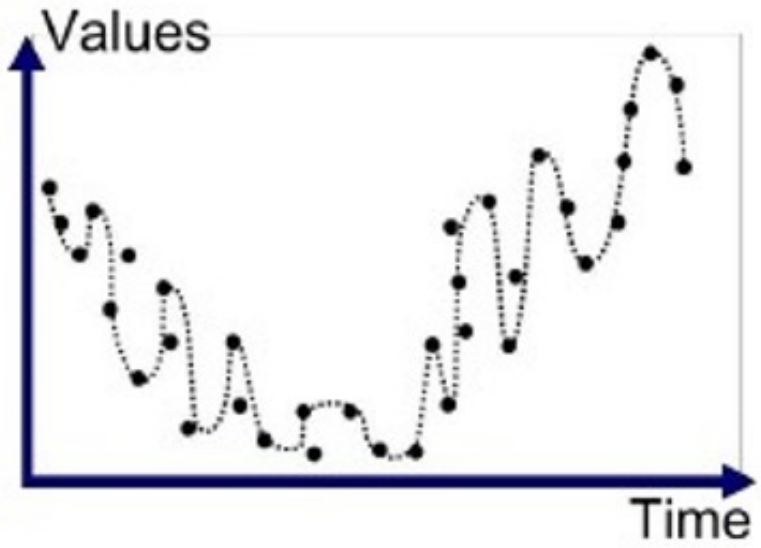
Potential Problems with High Flexibility



Underfitted



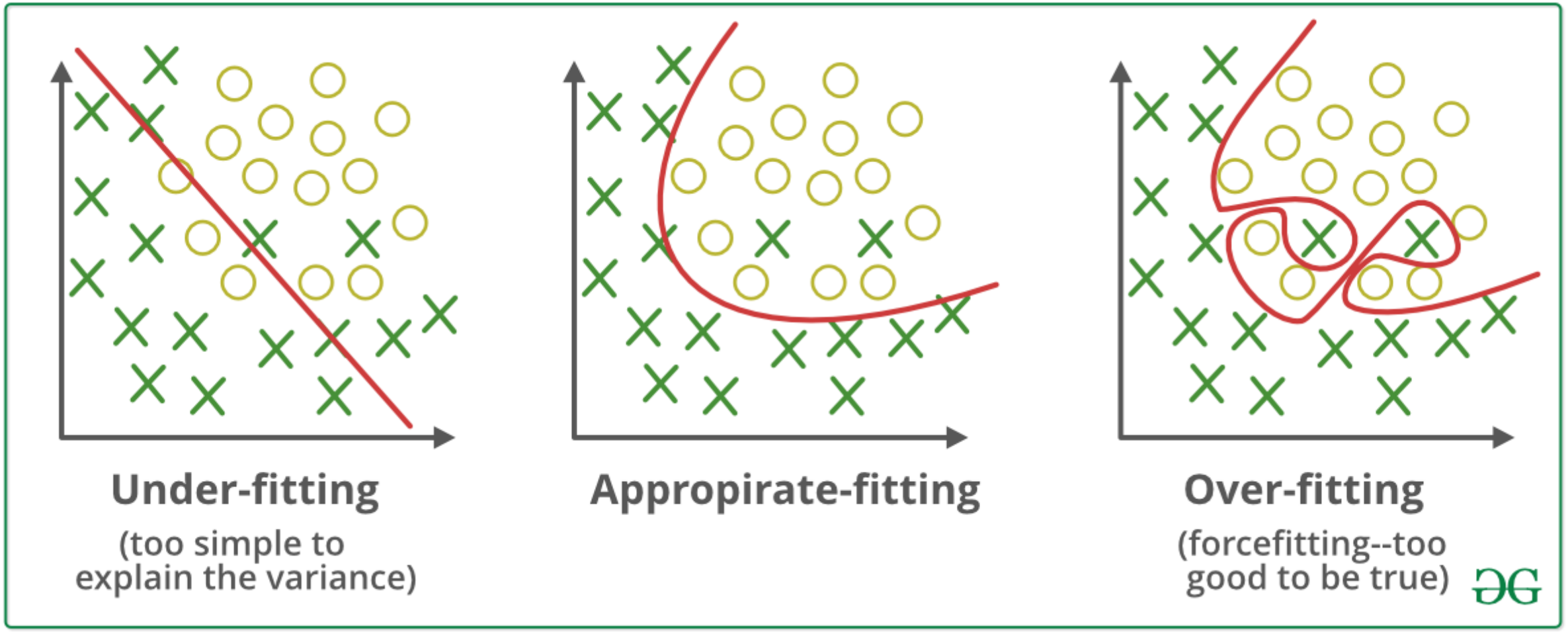
Good Fit/Robust



Overfitted

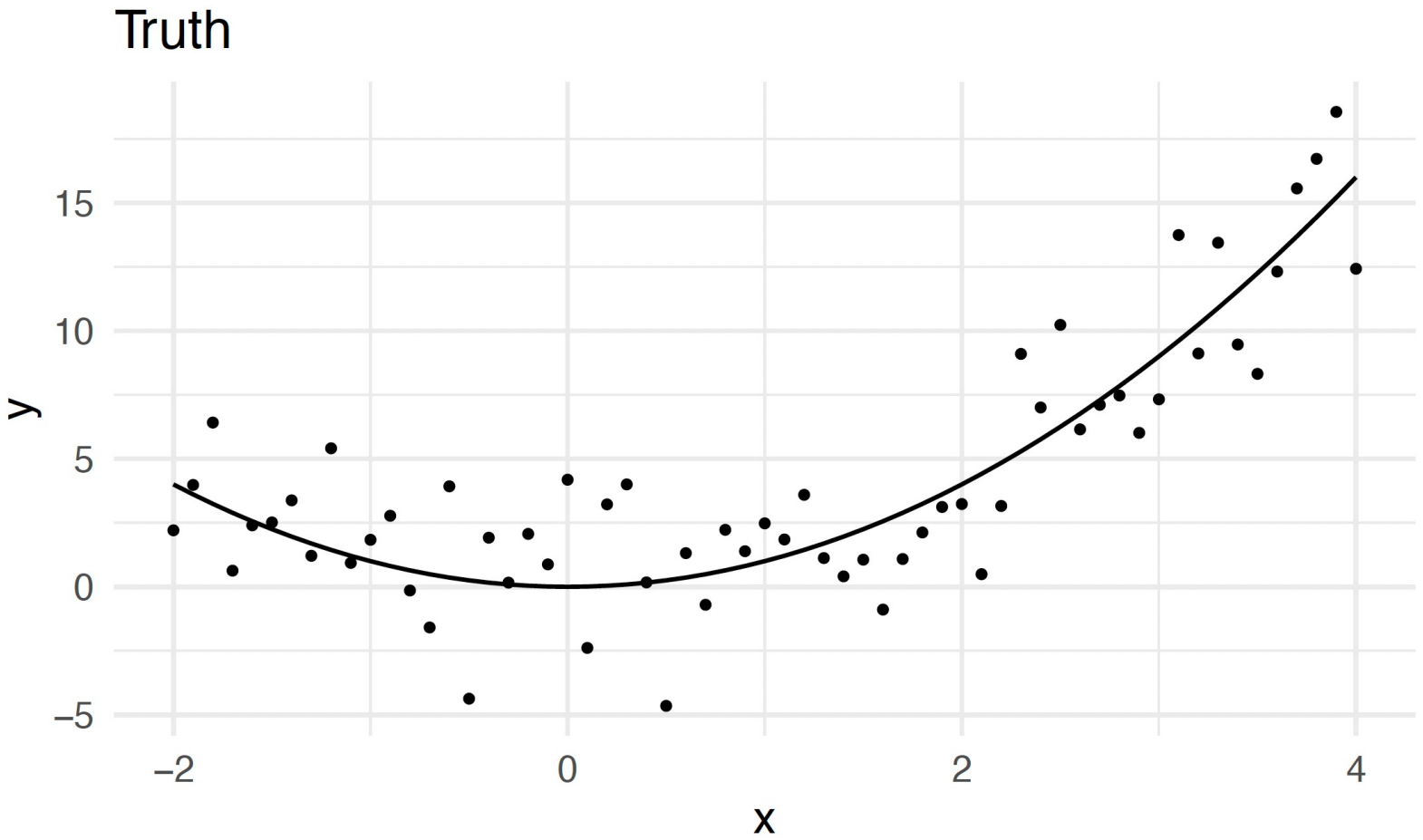
Prediction Accuracy vs Interpretability

Potential Problems with High Flexibility



Example for Potential Problems with High Flexibility

Example for Potential Problems with High Flexibility



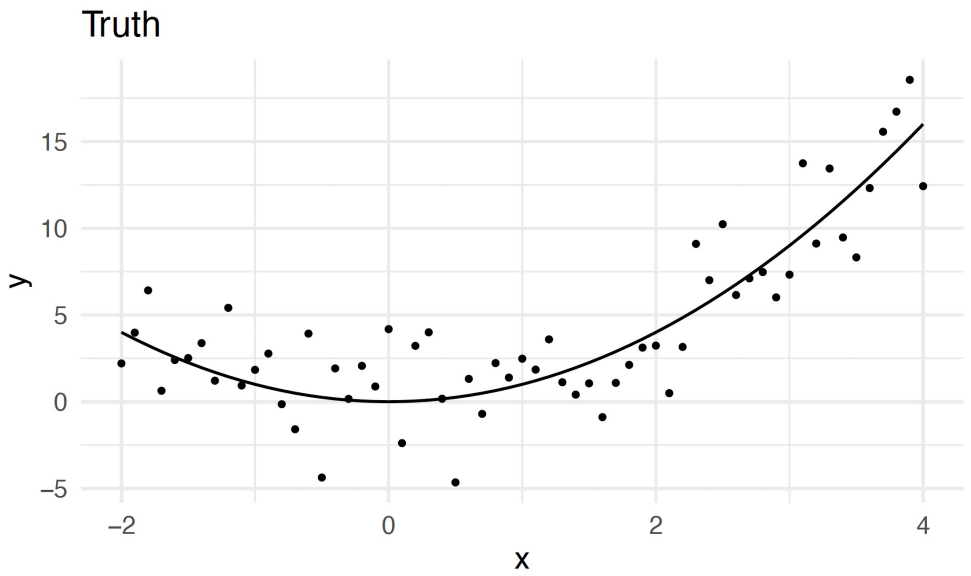
$$Y = x^2 + \epsilon$$
$$\epsilon \sim N(0, 2^2)$$

Example for Potential Problems with High Flexibility

Fitting a Polynomial Function

- **Poly1**: $\hat{y} = \beta_0 + \beta_1 x$
- **Poly2**: $\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2$
- **Poly10**: $\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \cdots + \beta_{10} x^{10}$
- **Poly20**: $\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \cdots + \beta_{20} x^{20}$

Flexibility increases

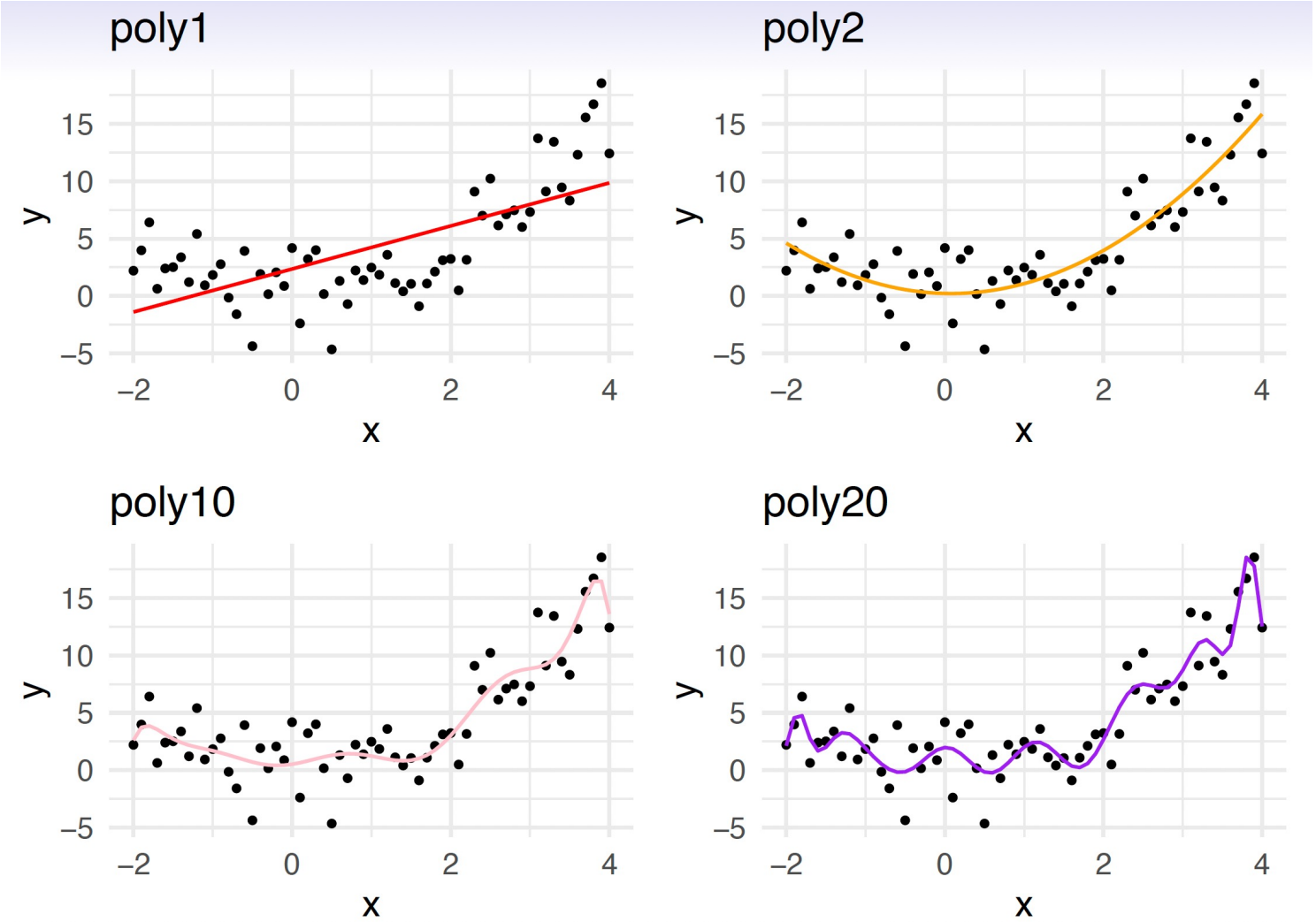


$$Y = x^2 + \epsilon$$
$$\epsilon \sim N(0, 2^2)$$

We know that the best fit is **Poly2**.

Example for Potential Problems with High Flexibility

Problems with High Flexibility



$\mathrm{E}\left[\left(Y-\hat{Y}\right)^2\right]$ decreases with the higher polynomial order.

But we know that **poly2** is the best fit.

How do we assess the best fit?

Assessing the Model Accuracy

Assessing the Model Accuracy

Measuring the Quality of Fit

You study for an exam, and often with the previous years' exams.

X_{train} (training dataset)

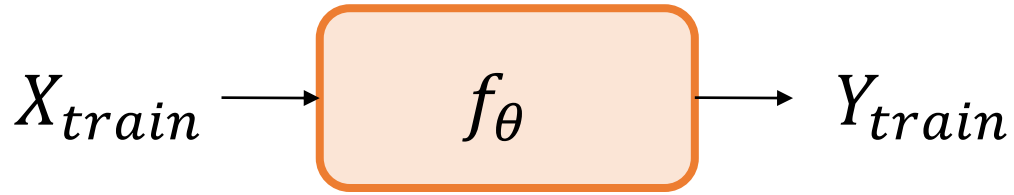
You take the exam.

X_{test} (test dataset)

Assessing the Model Accuracy

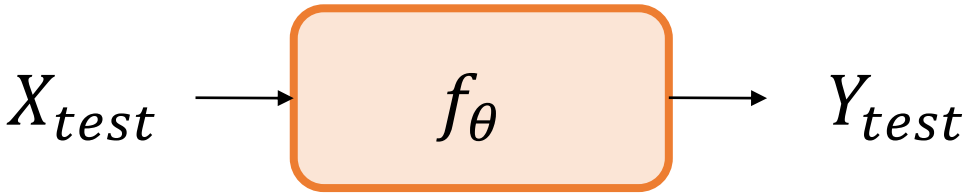
Measuring the Quality of Fit

During Training



$$\text{MSE}_{train} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y})^2$$

During Testing



$$\text{MSE}_{test} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y})^2$$

Assessing the Model Accuracy

Measuring the Quality of Fit

$$\text{MSE}_{train} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y})^2$$

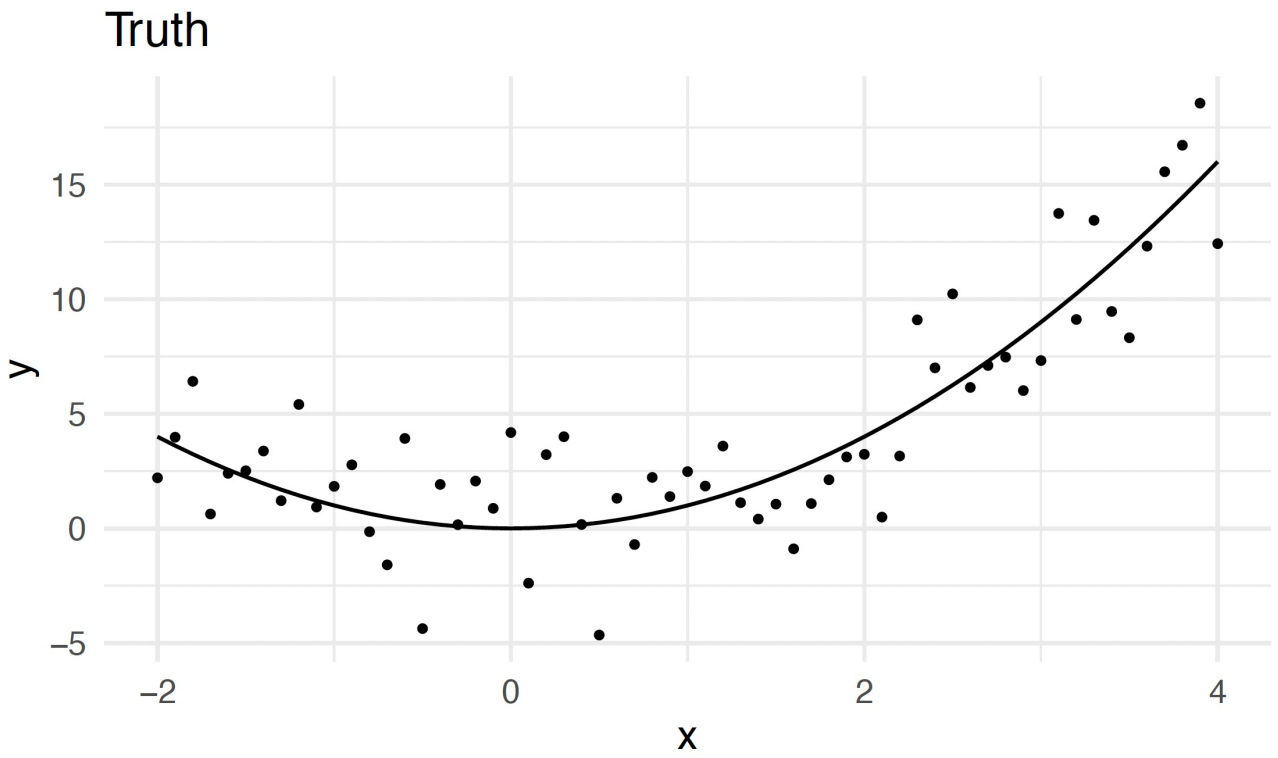
Your score on your mock exam (training dataset)

$$\text{MSE}_{test} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y})^2$$

Your score on the actual exam (test dataset)

Example for Potential Problems with High Flexibility

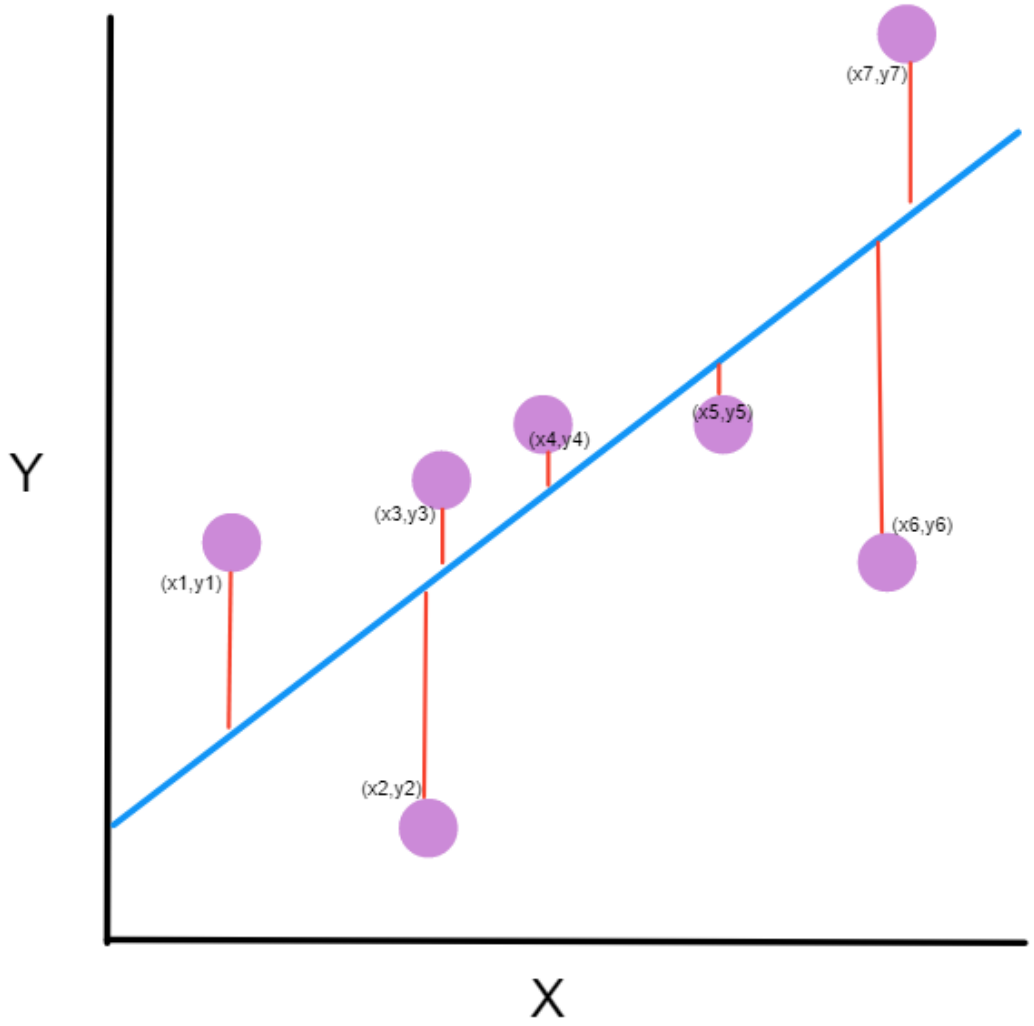
Then, we can create X_{train} and X_{test} from X



- We have X.
- We randomly pick 80% of X as X_{train} and the rest is X_{test}.
- We train $\hat{f}$ on X_{train} and test on X_{test}

Assessing the Model Accuracy

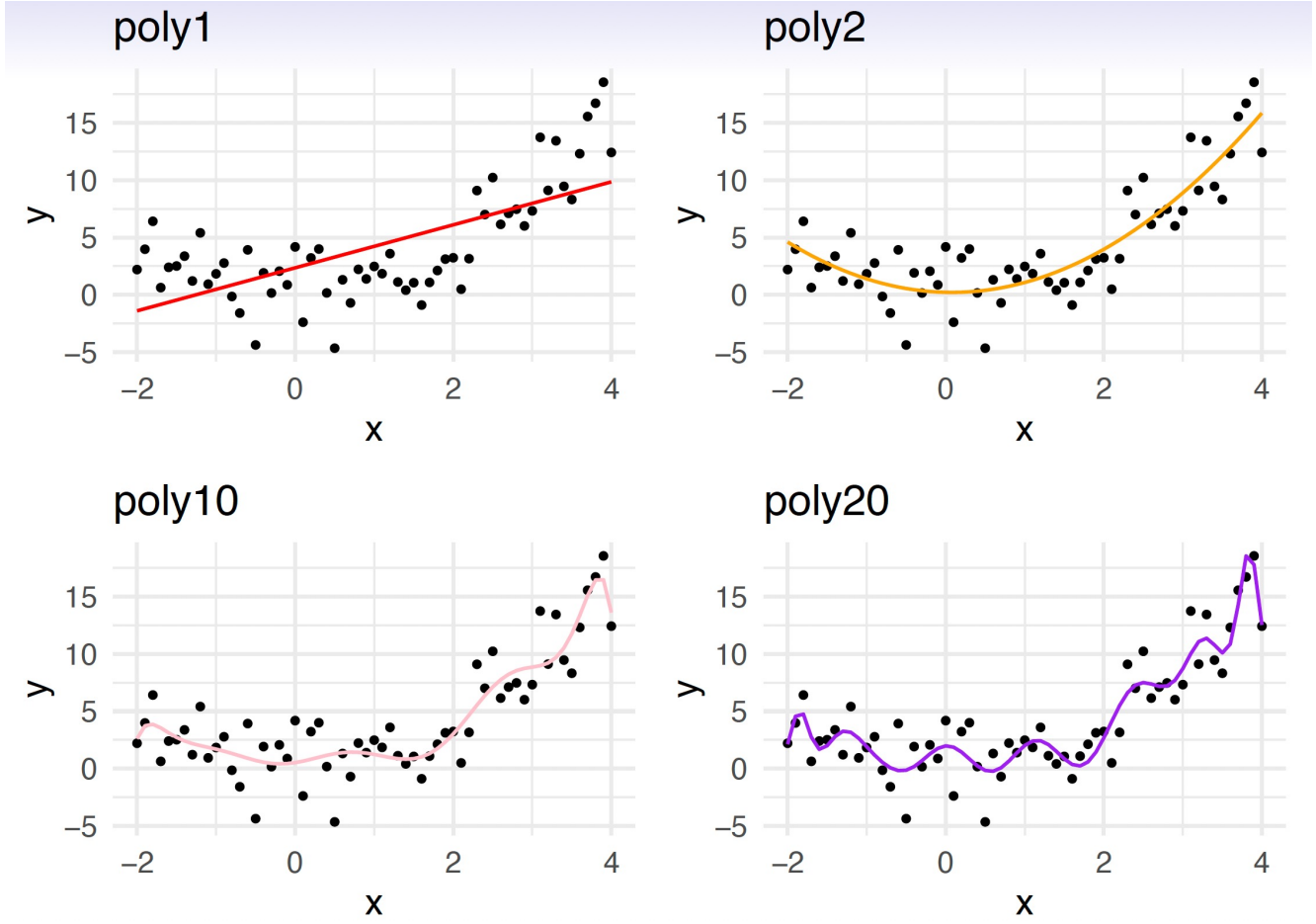
MSE (Mean Squared Error) Illustration

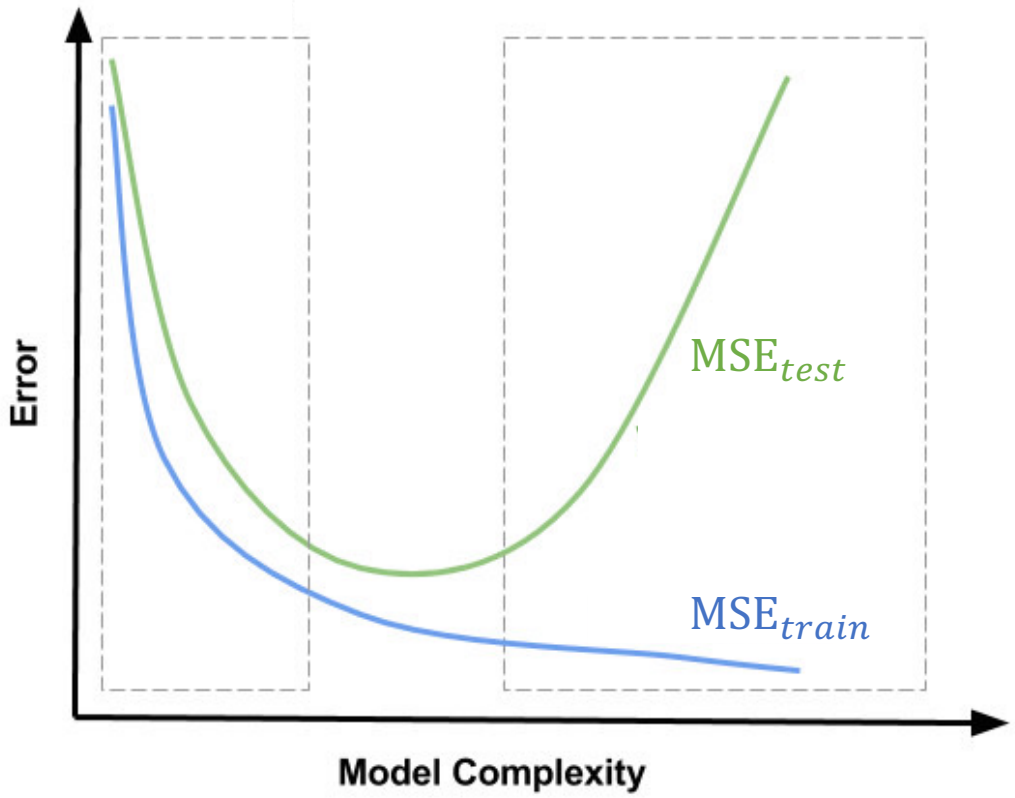


$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y})^2$$

Assessing the Model Accuracy

Example of MSE_{train} and MSE_{test}

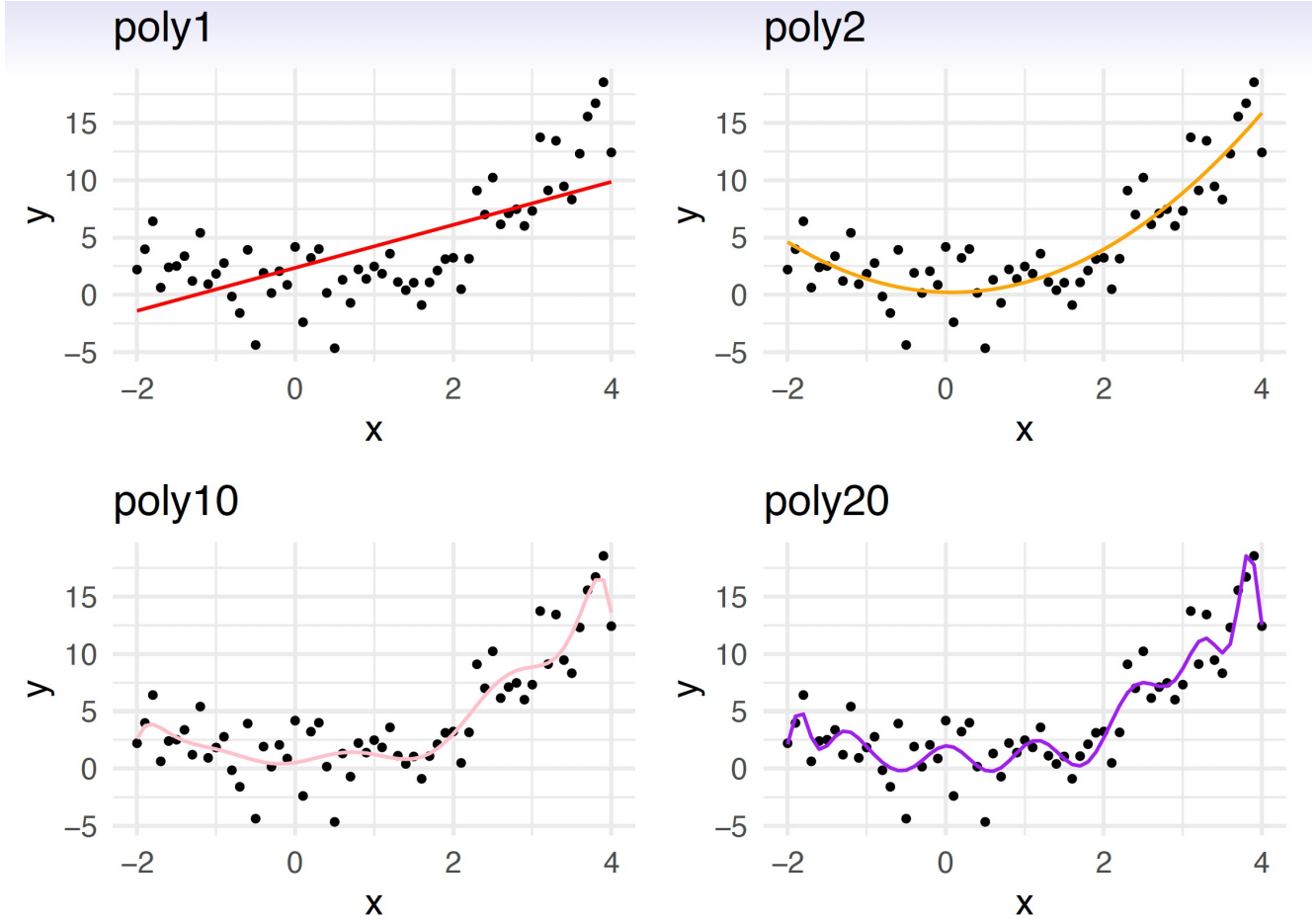


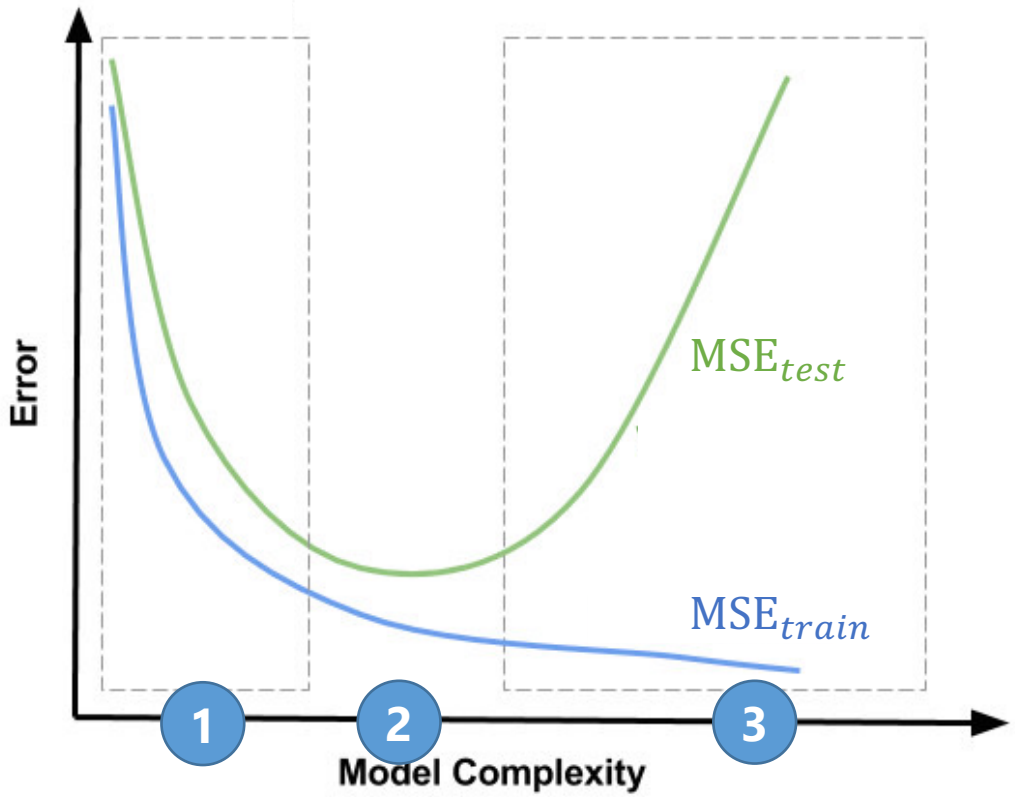


e.g. polynomial order

Assessing the Model Accuracy

Quiz: How would you select the best model?



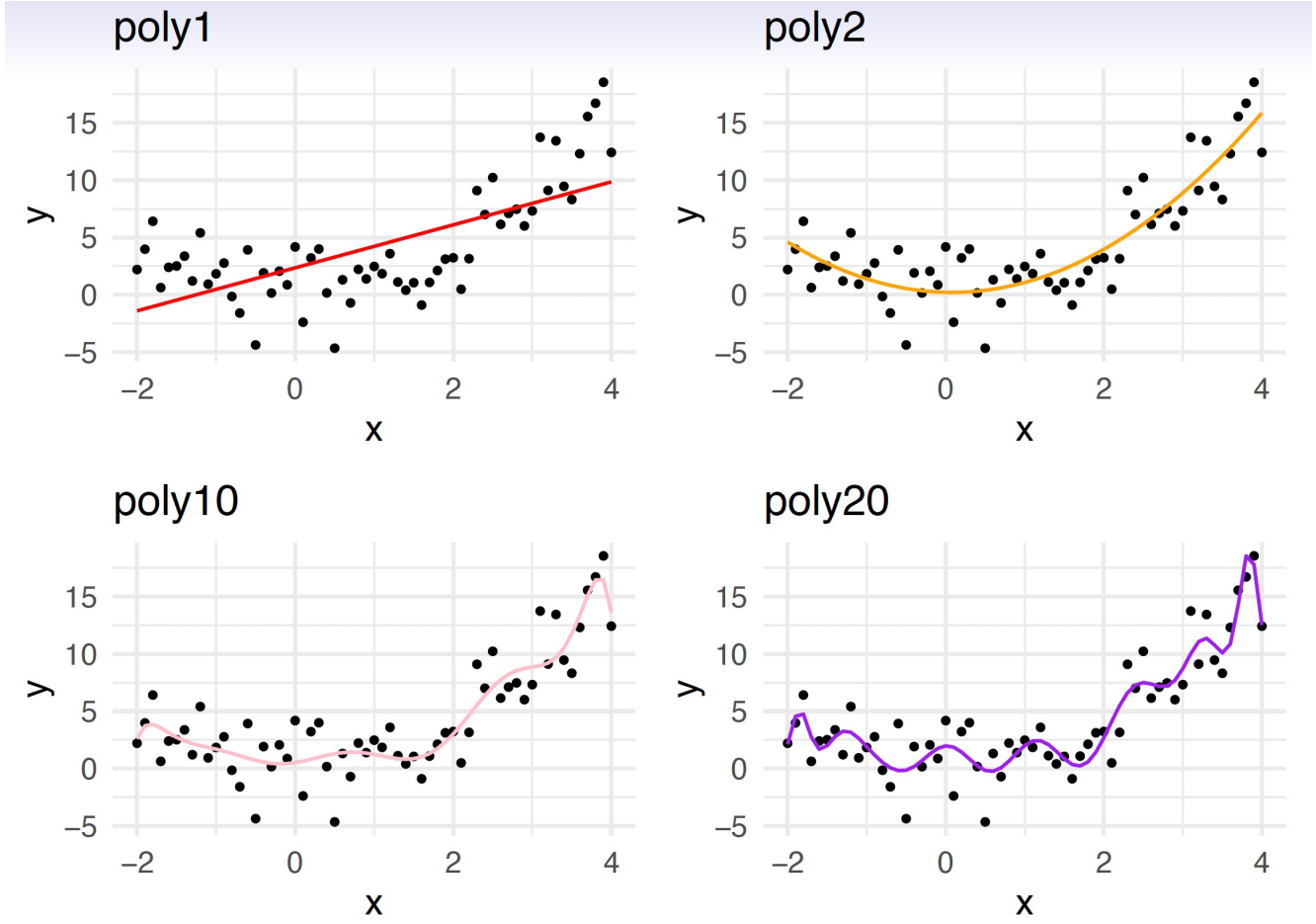


e.g. polynomial order

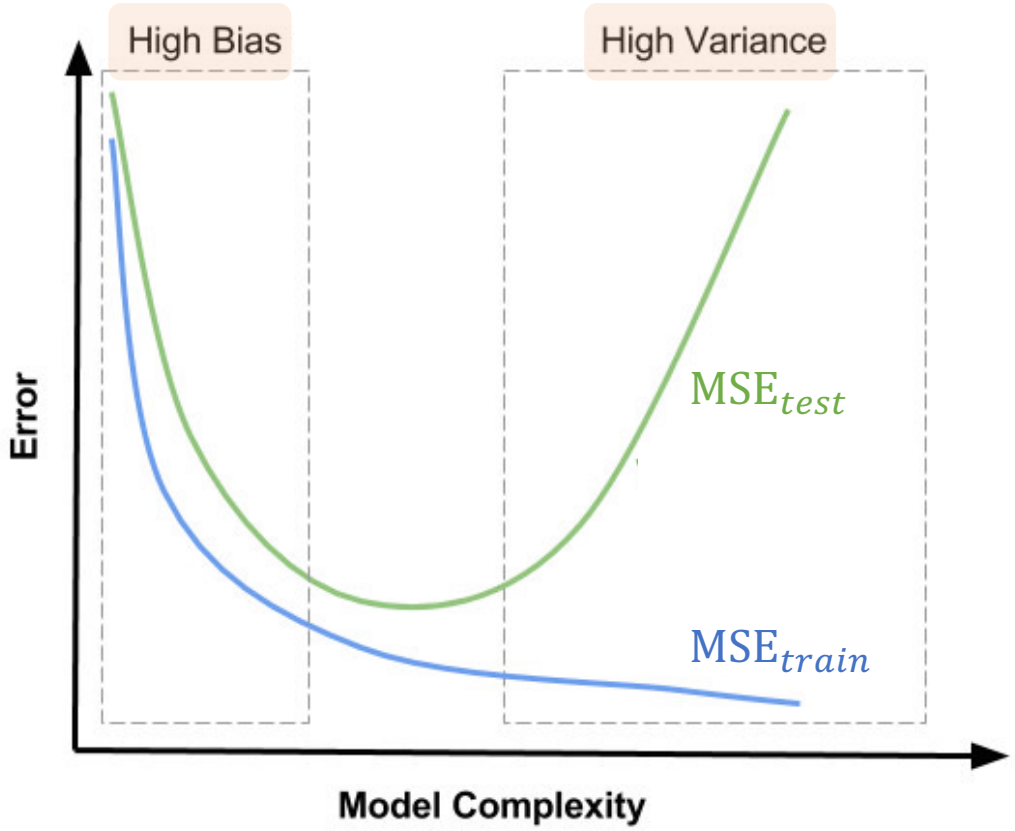
Bias-Variance Trade-off

Bias-Variance Trade-off

Bias and Variance?



NB! The U-shape of MSE_{test}



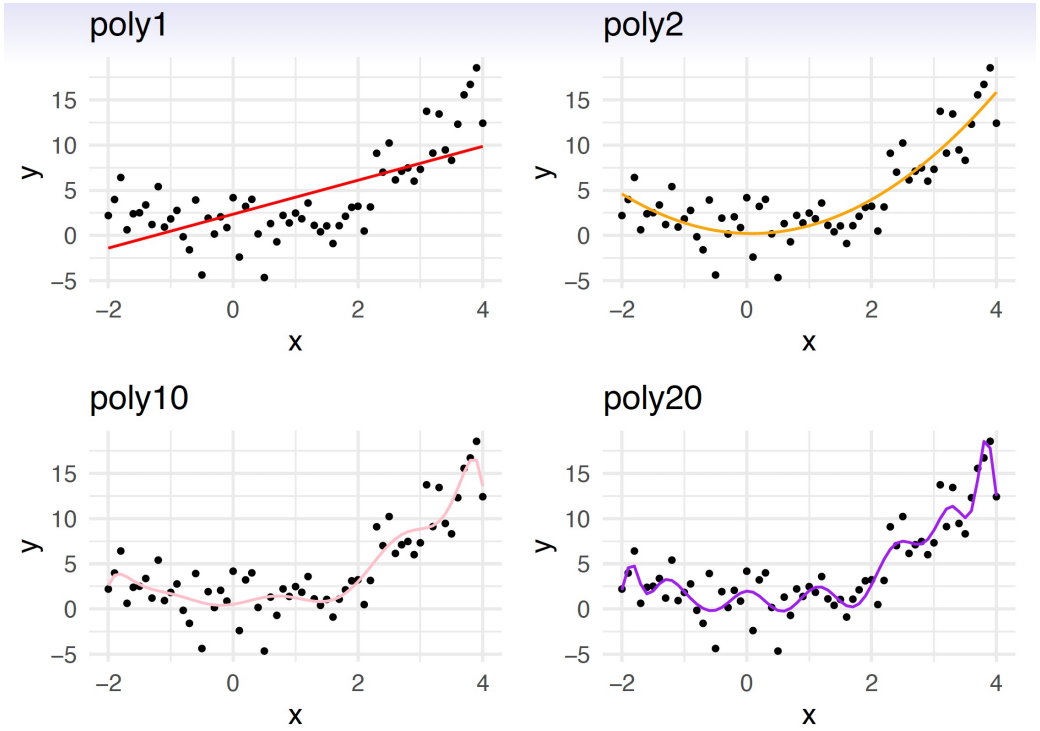
e.g. polynomial order

Bias-Variance Trade-off

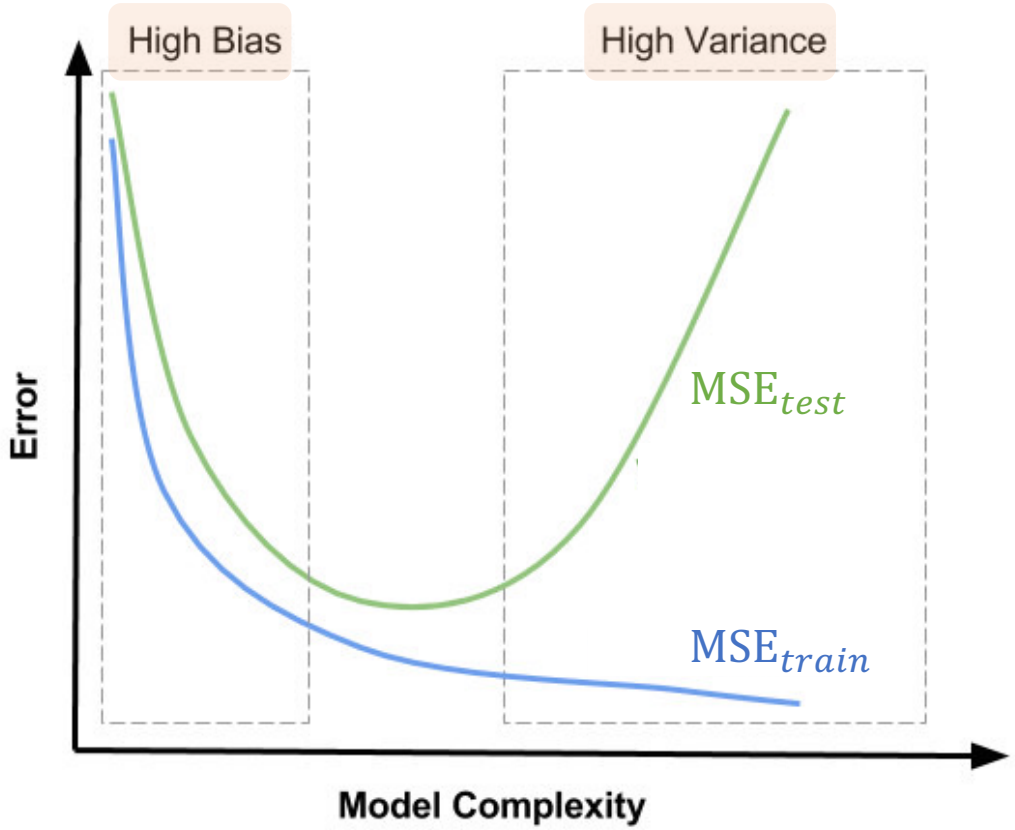
$$\mathrm{E}\left[\left(Y-\hat{Y}\right)^2\right] = \underbrace{\left(\mathrm{E}\left[f(X)-\hat{f}(X)\right]\right)^2}_{\text{model bias}} + \underbrace{\mathrm{Var}\left(\hat{f}(X)\right)}_{\text{model variance}} + \underbrace{\mathrm{Var}(\epsilon)}_{\text{variance of}}$$

reducible error

irreducible error



NB! The U-shape of MSE_{test}

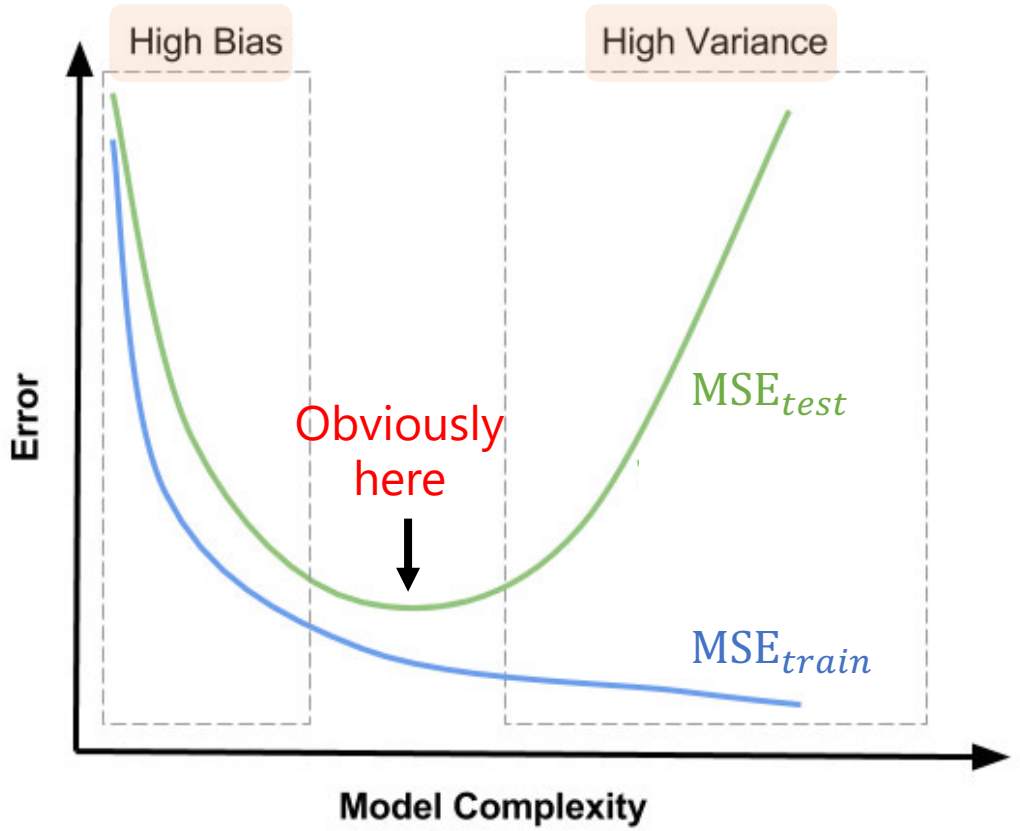


e.g. polynomial order

Bias-Variance Trade-off

How to Choose the Best Model

NB! The U-shape of MSE_{test}



e.g. polynomial order

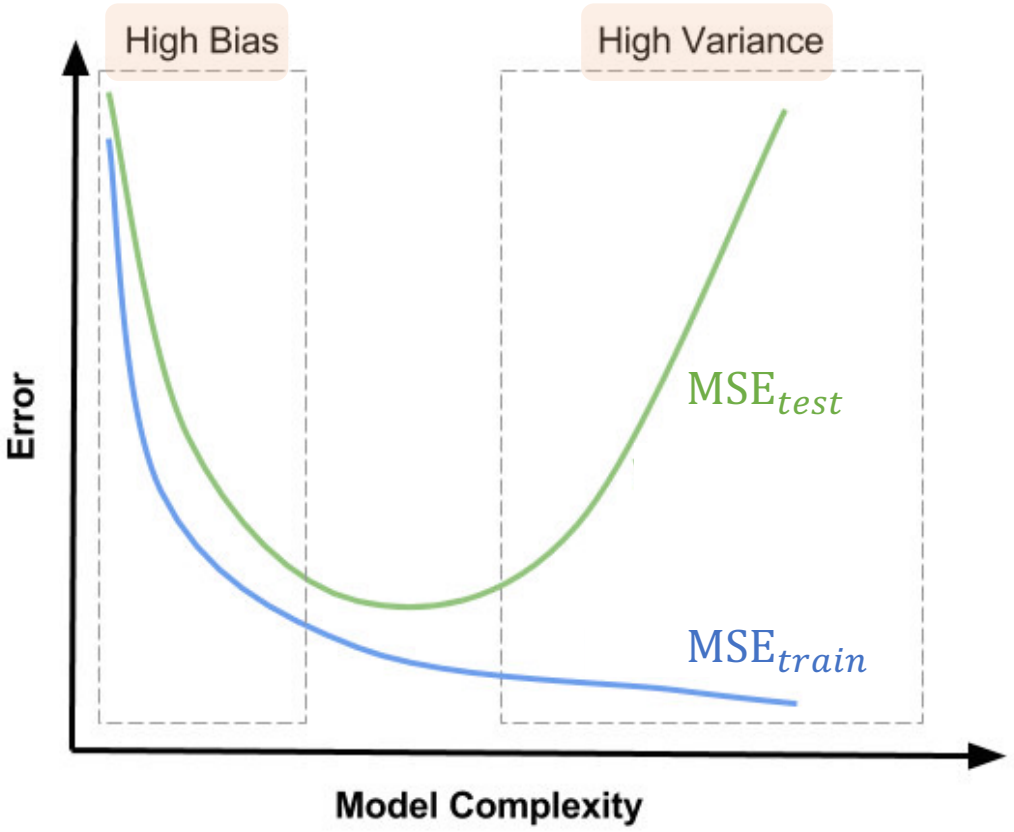
Bias-Variance Trade-off

$$\mathrm{E}\left[\left(Y-\hat{Y}\right)^2\right]=\underbrace{\left(\mathrm{E}\left[f(X)-\hat{f}(X)\right]\right)^2}_{\text{model bias}}+\underbrace{\mathrm{Var}\left(\hat{f}(X)\right)}_{\text{model variance}}+\underbrace{\mathrm{Var}(\epsilon)}_{\text{variance of}}$$

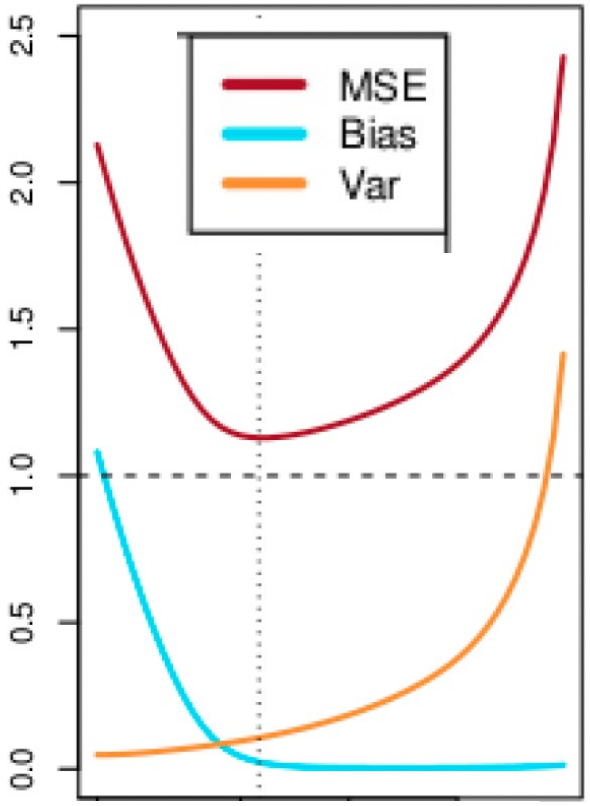
reducible error

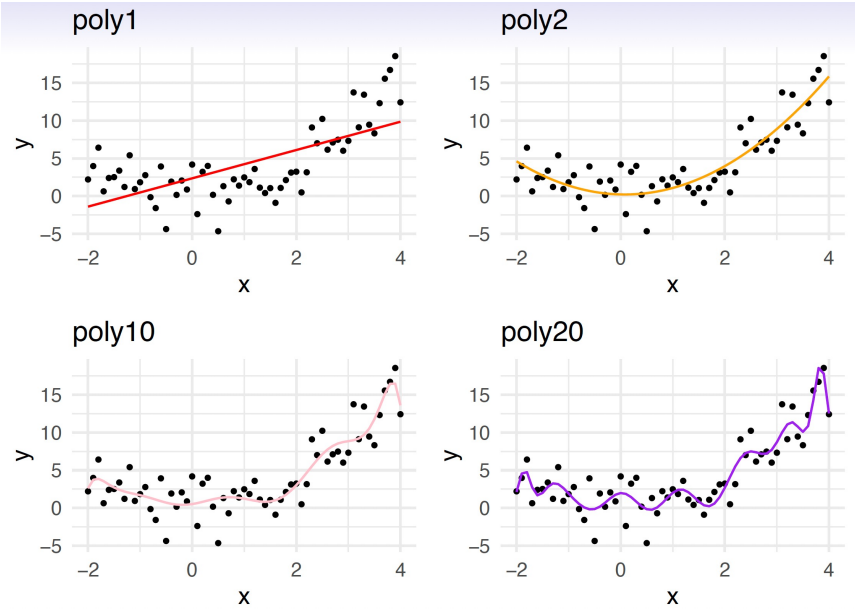
irreducible error

Decomposition of MSE



e.g. polynomial order






The End!