

# Machine Learning

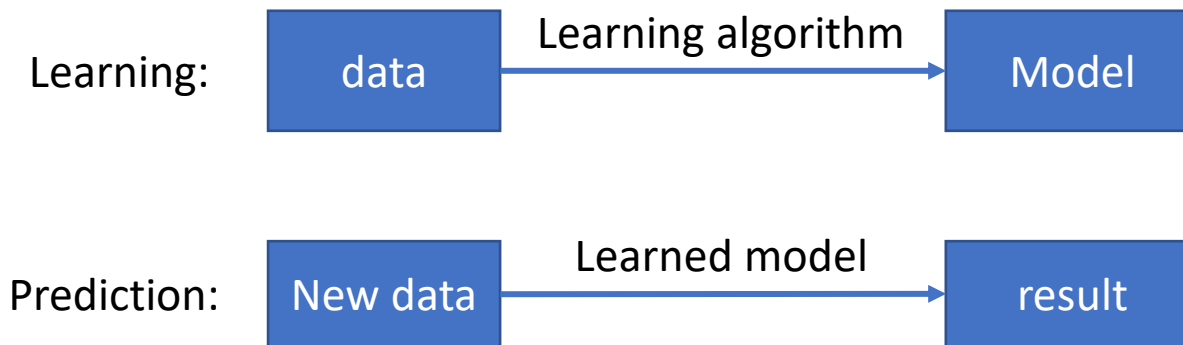
---

Model Selection

Dr. Shuang LIANG

# Recall: Machine Learning

- One possible definition
  - a set of methods that can automatically **detect patterns** in data, and then use the uncovered patterns to **predict future data**, or to perform other kinds of **decision making** under uncertainty



# Recall: Machine Learning $\approx$ Looking for Function (model)

- Speech Recognition

$$f\left(\text{[Waveform of 'How are you']}\right) = \text{"How are you"}$$

- Image Recognition

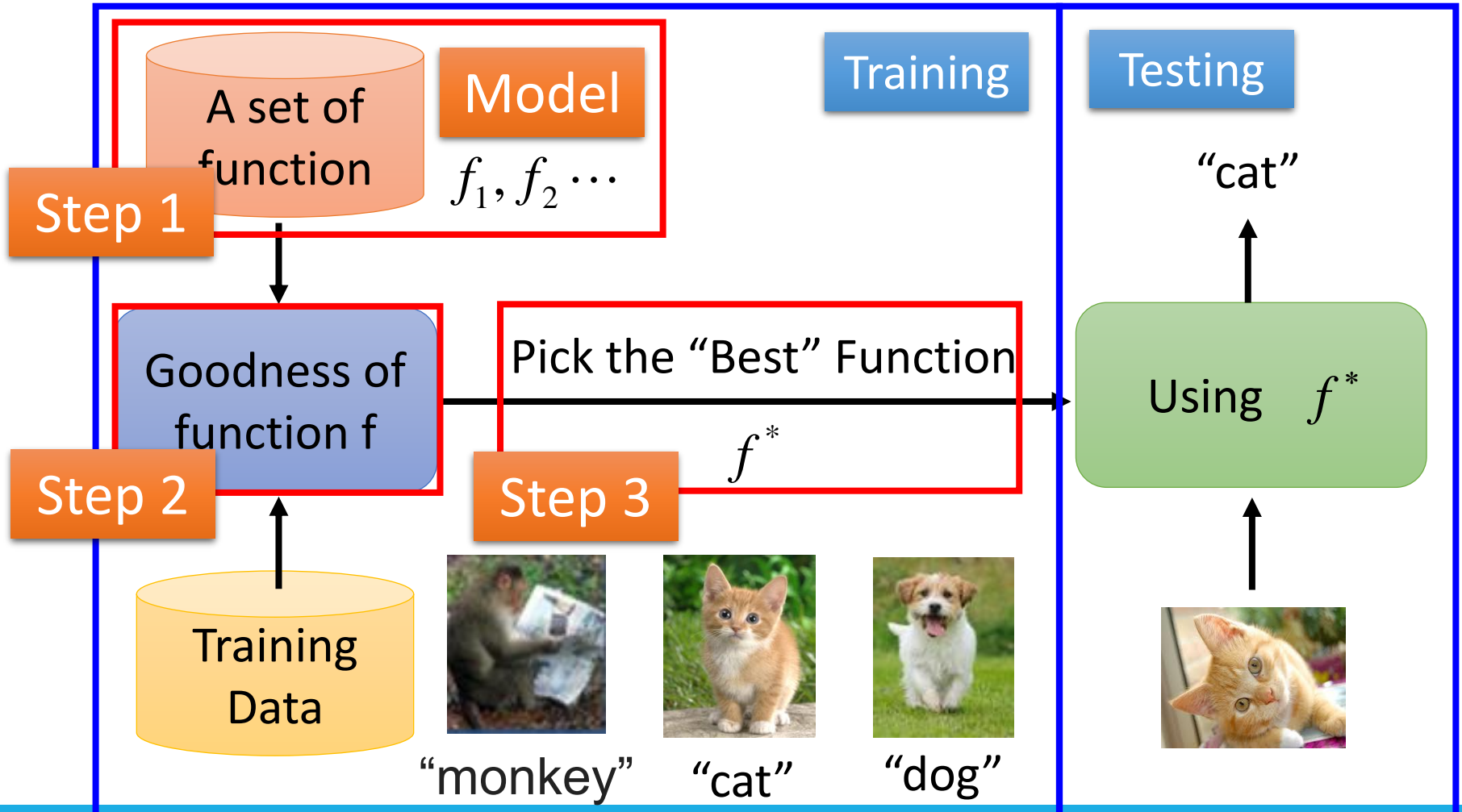
$$f\left(\text{[Image of a cat]}\right) = \text{"Cat"}$$

- Playing Go

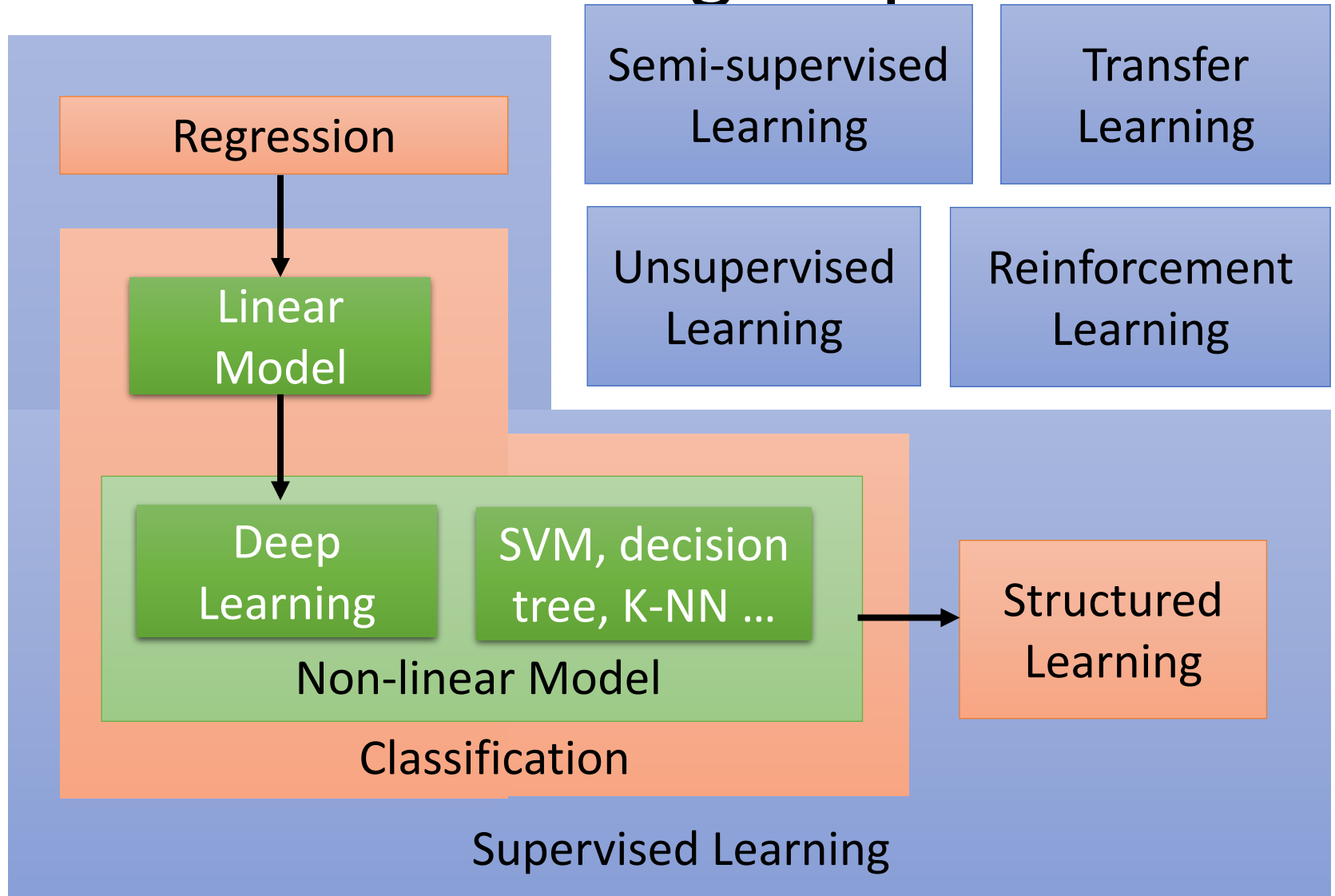
$$f\left(\text{[Go board state]}\right) = \text{"5-5"}_{\text{(next move)}}$$

# Recall: Framework

$$f\left(\text{Image of a cat}\right) = \text{"cat"}$$



# Recall: ML Learning Map



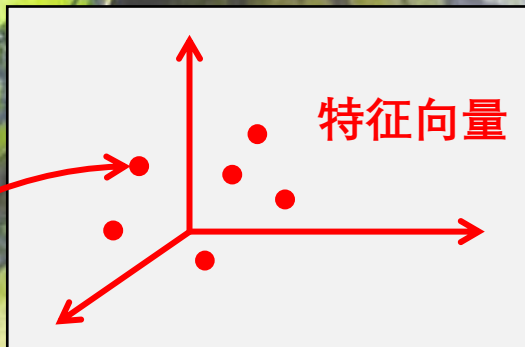
# Today's Topics

- Terminology
- Error and Overfitting
- Evaluation Methods
- Performance Measure
- Bias and Variance

# Today's Topics

- *Terminology*
- Error and Overfitting
- Evaluation Methods
- Performance Measure
- Bias and Variance

# Data



属性空间/样本空间/输入空间

属性/特征  
Attributes/Feature

属性值

标记/标签  
Label

示例/样本  
Instance/Sample

训练集/训练样本  
Training Set

测试集/测试样本  
Testing Set

训练

预测

模型/学习器  
Model/Learner

每个示例由3个特征描述，则其维数(dimensionality)为3

假设  
Hypothesis

逼近

真实/真相  
Ground-truth

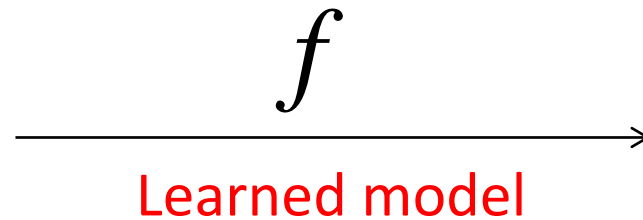
数据集  
Data Set

样例  
example

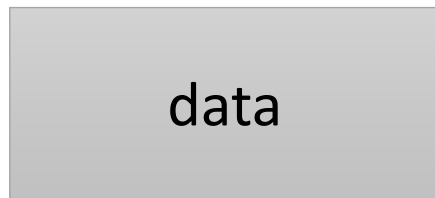
编号	色泽	根蒂	敲声	好瓜
1	青绿	蜷缩	浊响	是
2	乌黑	蜷缩	沉闷	是
3	青绿	硬挺	清脆	否
4	乌黑	稍蜷	沉闷	否
1	青绿	蜷缩	沉闷	?



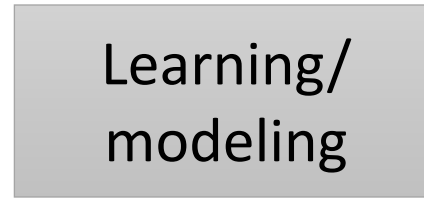
# Train & Test



cat



Collected from past observations



Learn to capture patterns in the data

Training



Apply the model to forecast what is going to happen for new data

Testing

# Task

- By prediction target
  - Classification: discrete value
    - Binary: Good melon, Bad melon
    - Multiclass: Cucumber; Pumpkin; Watermelon
  - Regression: continuous value
    - Ripeness of melon
  - Clustering: no label
- By label
  - Supervised Learning: Classification, Regression
  - Unsupervised Learning: Clustering
  - Semi-supervised Learning: Combing the two

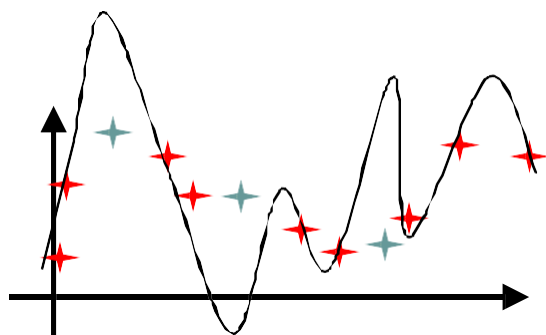
# Today's Topics

- Terminology
- *Error and Overfitting*
- Evaluation Methods
- Performance Measure
- Bias and Variance

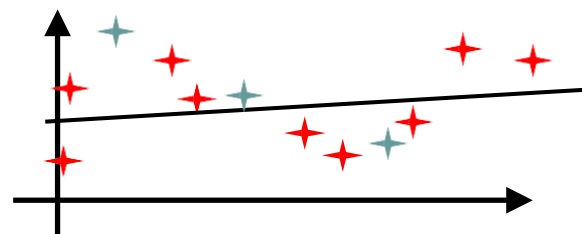
# Model Selection

- So many learning algorithms
- Even the same algorithm has many different parameter combinations
- Which one is the best?

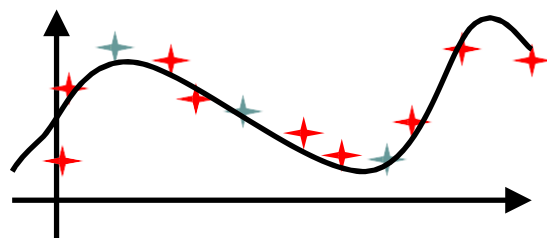
# Model Selection – What is a good model?



**Low Robustness**






**Low quality /High Robustness**



**Robust Model**

## LEGEND

-  Model built
-  Known Data
-  New Data

# Model Selection - Example

- When you build a neural network...

## Algorithms?

**SGD**  
**Adam**  
**Which step-size?**  
**Which batch-size?**  
**Which momentum?**

## Architectures?

**FullyConnected**  
**ConvNet**  
**ResNet**  
**Transformer**  
**Which width?**  
**Which depth?**  
**Batch normalization?**

## Regularizations?

**Weight decay?**  
**Early stopping?**  
**Data augmentations?**

# Accuracy & Error

- We need to evaluate our model

- Error rate

- $E = \frac{\text{the number of misclassified samples } (a)}{\text{the number of all samples } (m)} = \frac{a}{m}$

- How about accuracy?

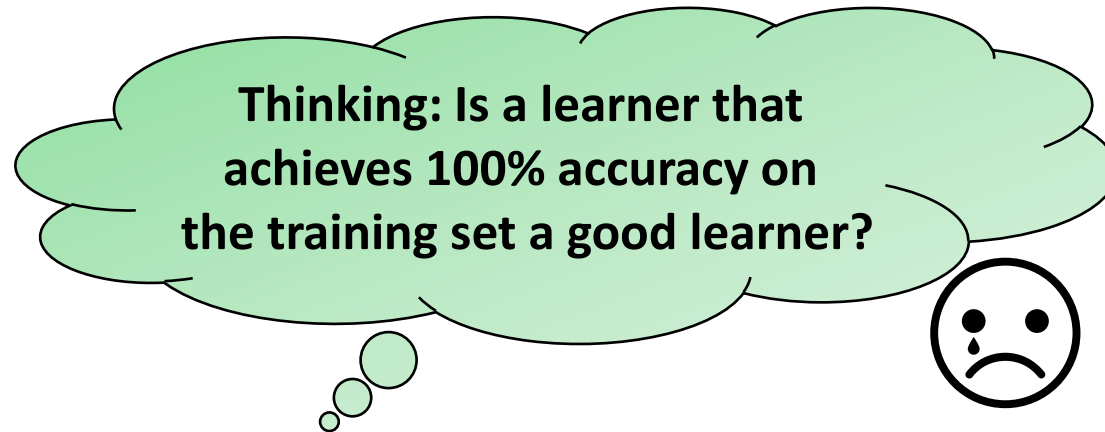
$$A = 1 - E = 1 - \frac{a}{m}$$

# Accuracy & Error

- Error 误差
  - Error is the difference between the output of a learner and the ground-truth of samples
- Training error/Empirical error 训练/经验误差
  - error on the training set
- Testing error 测试误差
  - error on the testing set
- Generalization error 泛化误差
  - error on all samples except training set



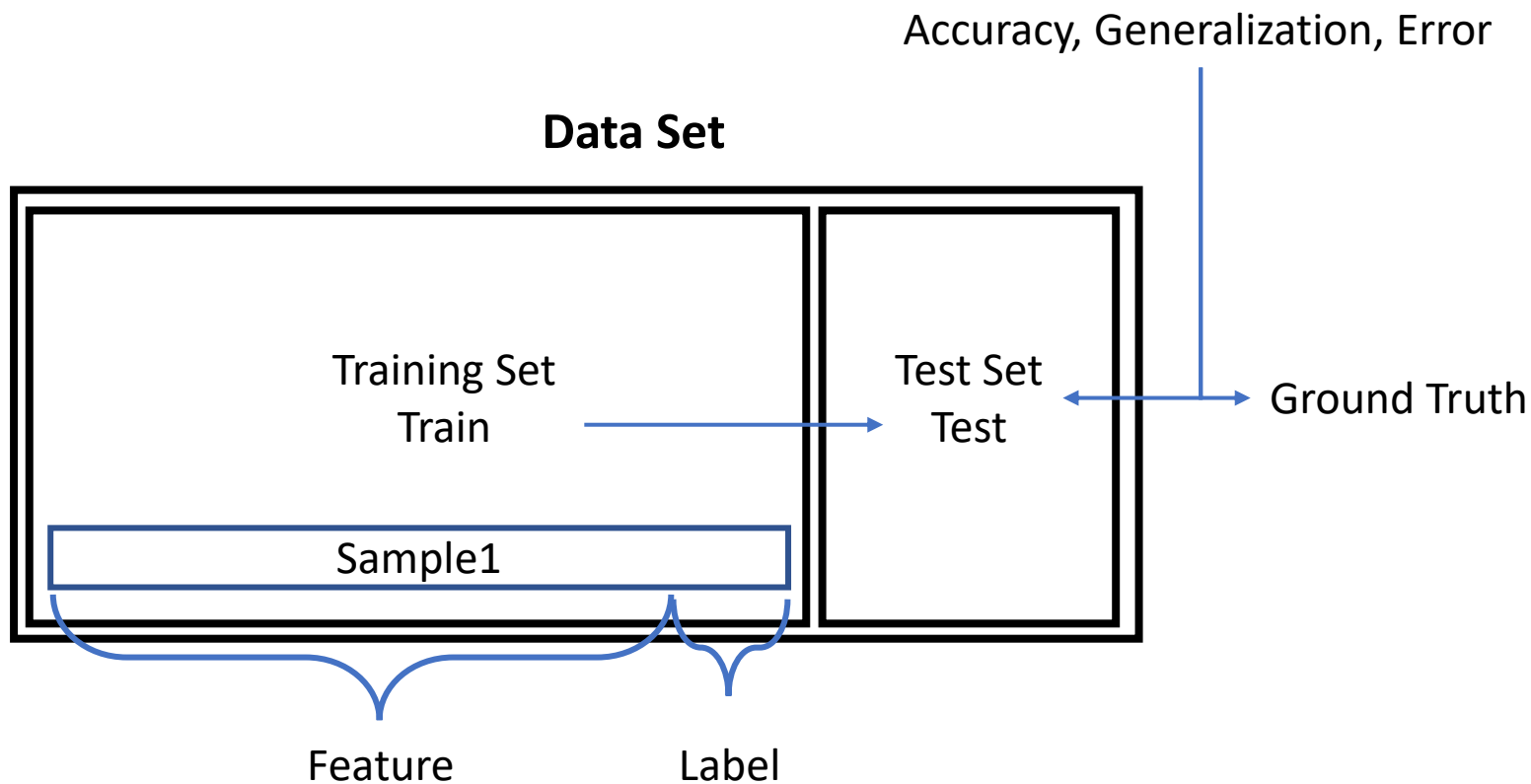
# Generalization



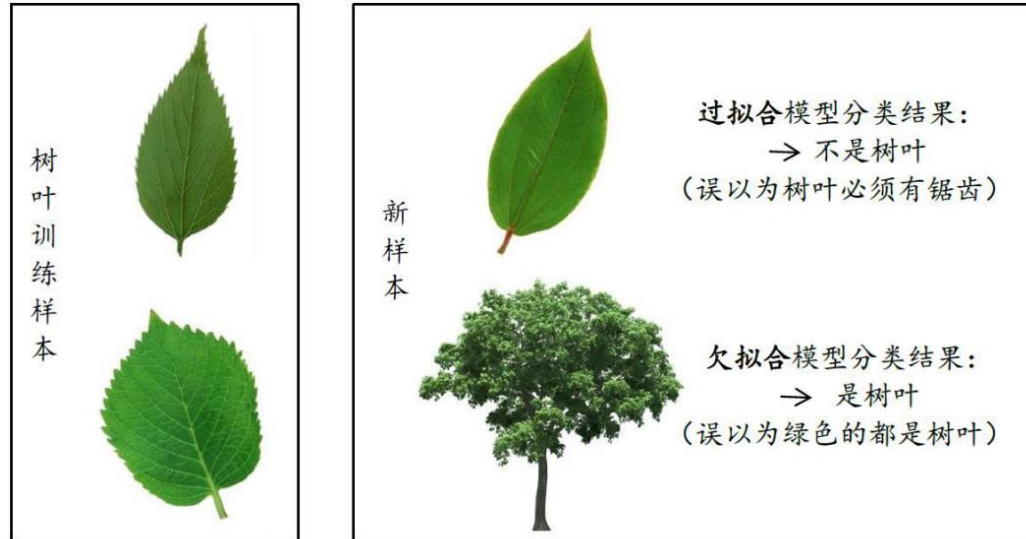
- The goal of machine learning is to make the learned model work well on "new samples" rather than just the training set.
- We call the ability of a model to adapt to new samples as **generalization**
- **We want to get a learner with a small generalization error!**

# Try

- Can you distinguish these terms ?



# Underfitting and Overfitting



过拟合、欠拟合的直观类比

**Overfitting:** The learner regards the features of the training sample itself as a **general property** that all potential samples will have.

**Underfitting:** The **general properties** of the training samples have **not** been learned by the learner.

**Overfitting is a key problem in machine learning**

# Overfitting

- Small loss on training data, large loss on testing data. Why?

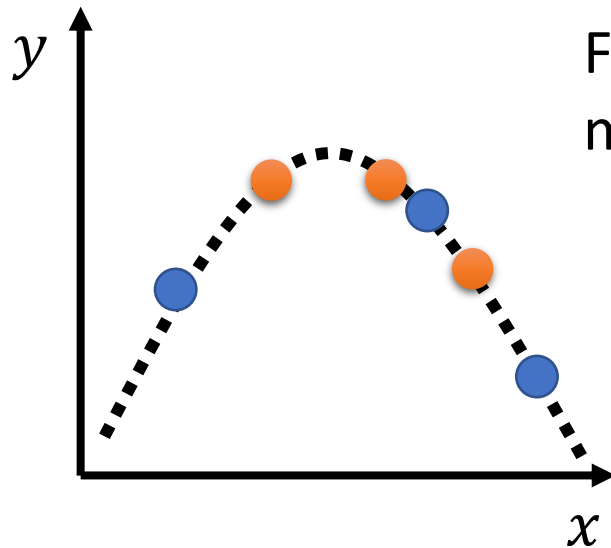
## An extreme example

Training data:  $\{(\mathbf{x}^1, \hat{y}^1), (\mathbf{x}^2, \hat{y}^2), \dots, (\mathbf{x}^N, \hat{y}^N)\}$

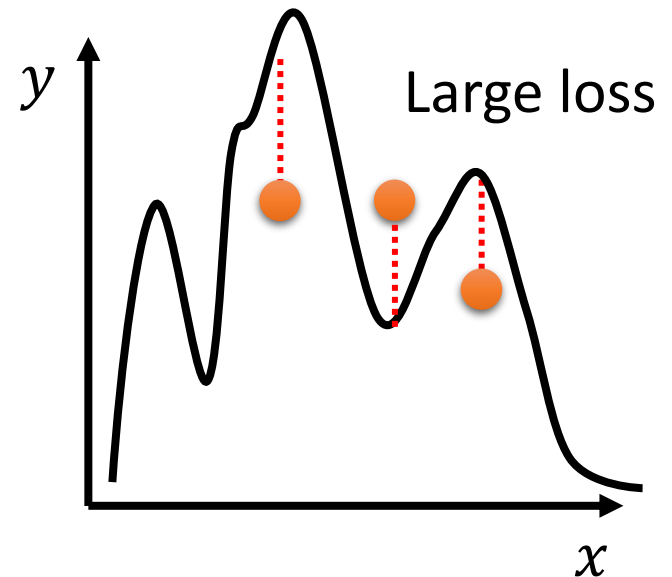
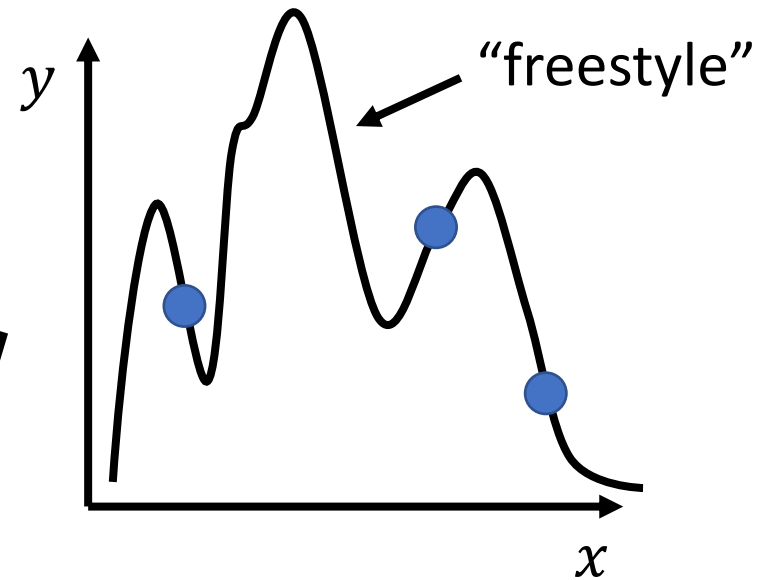
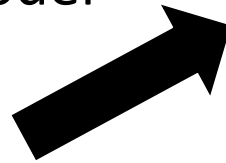
$$f(\mathbf{x}) = \begin{cases} \hat{y}^i & \exists \mathbf{x}^i = \mathbf{x} \\ random & otherwise \end{cases} \quad \text{Less than useless ...}$$

This function obtains **zero training loss**, but **large testing loss**.

# Overfitting



Flexible  
model

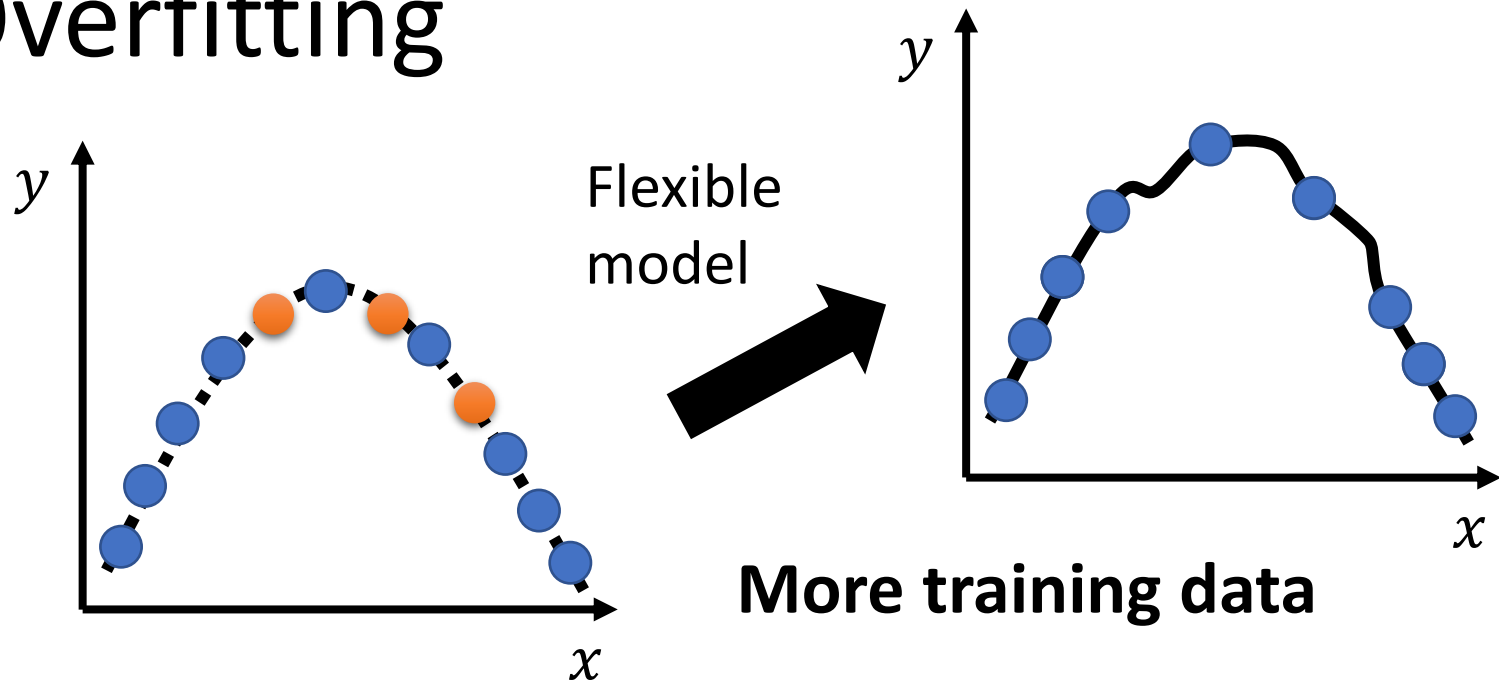


---- Real data distribution  
(not observable)

● Training data

● Testing data

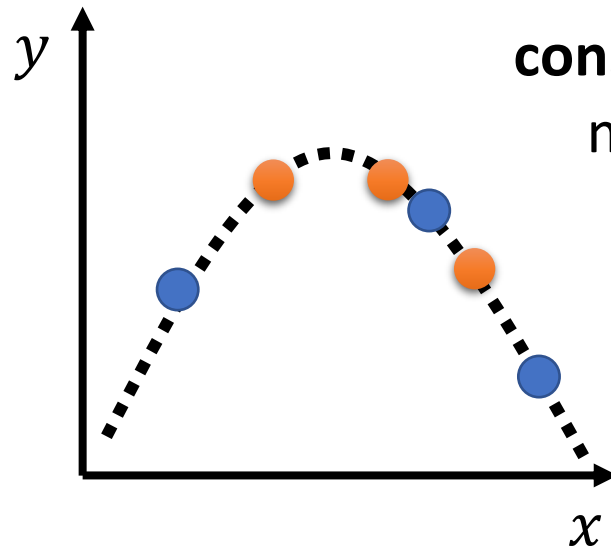
# Overfitting



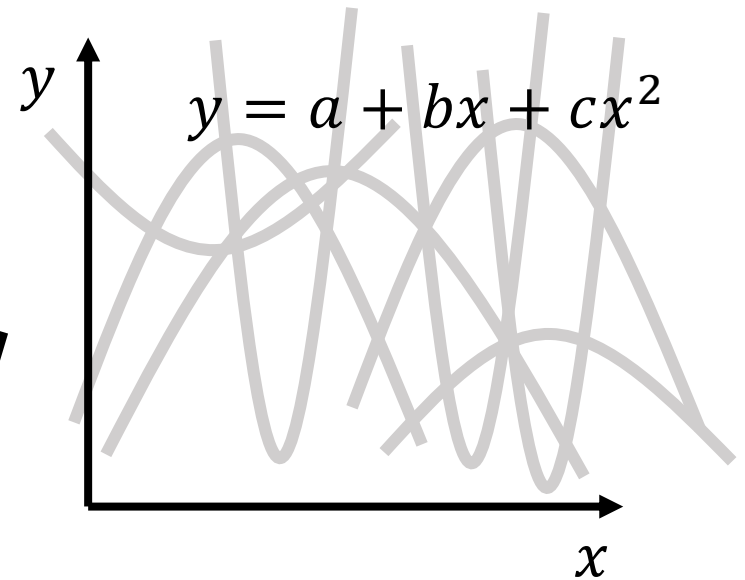
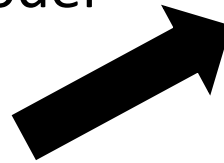
## Data augmentation



# Overfitting

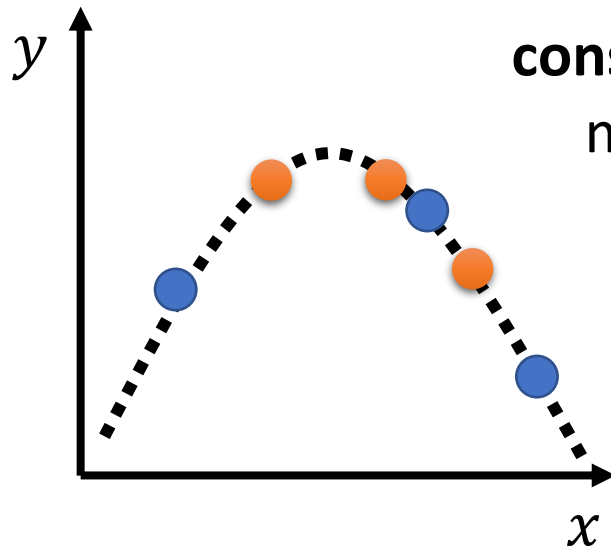


**constrained  
model**

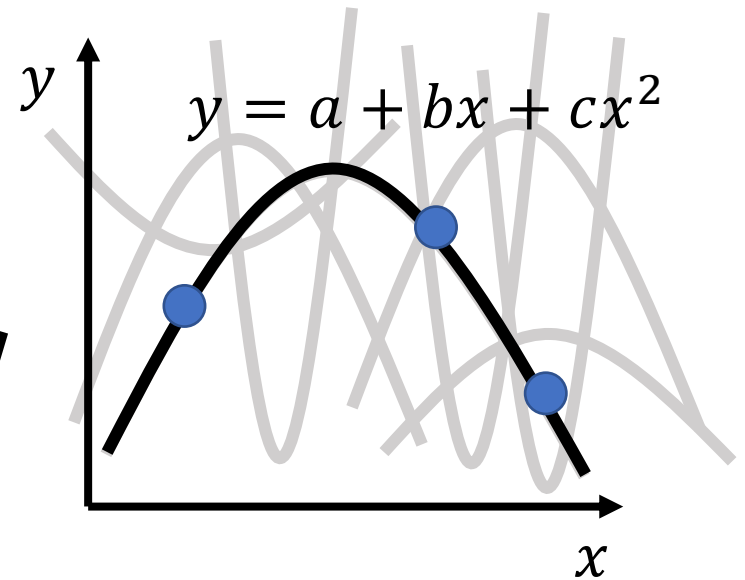
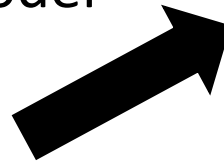


- Real data distribution  
(not observable)
- Training data
- Testing data

# Overfitting



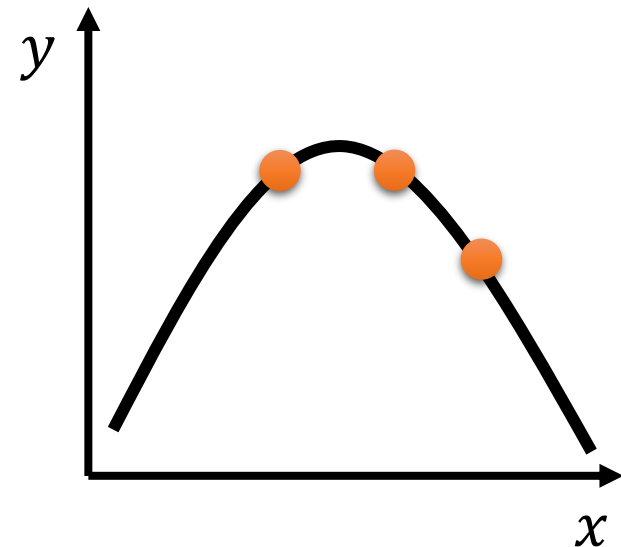
constrained  
model



---- Real data distribution  
(not observable)

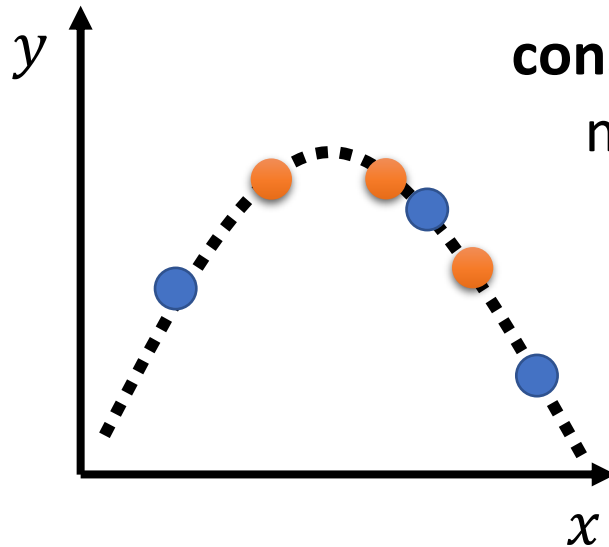
● Training data

● Testing data

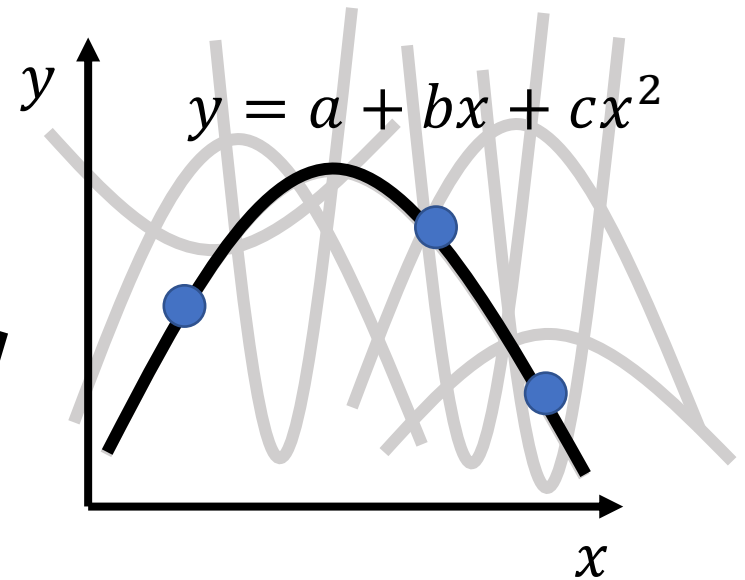
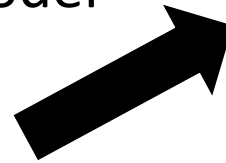




# Overfitting



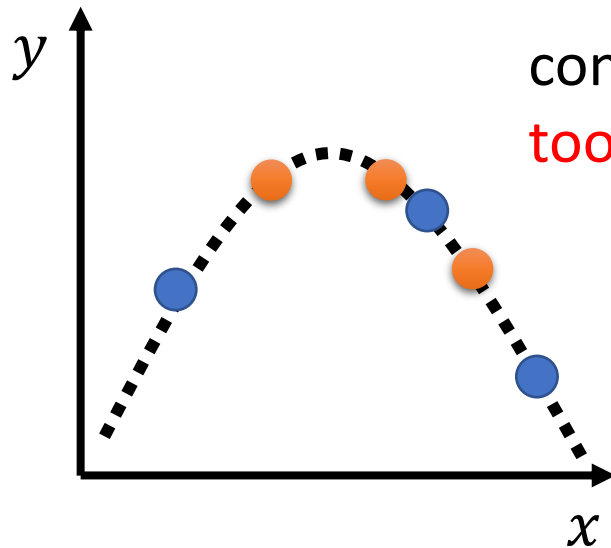
constrained  
model



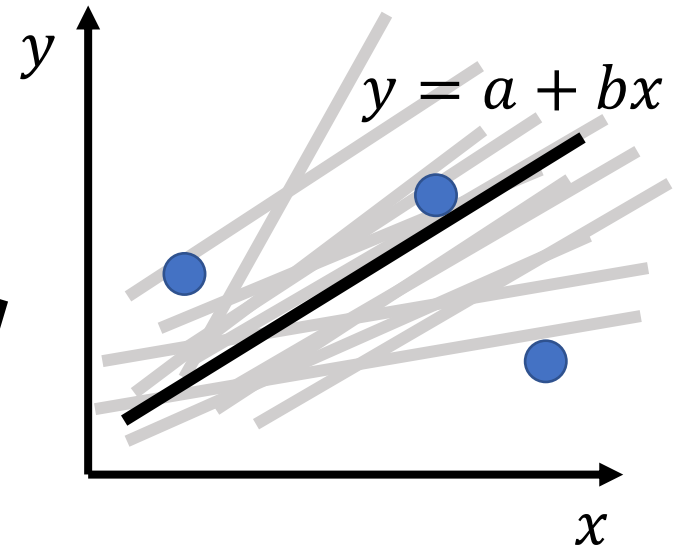
## Some Solutions

- Less parameters, sharing parameters
- Less features
- Early stopping
- Regularization
- Dropout

# Overfitting



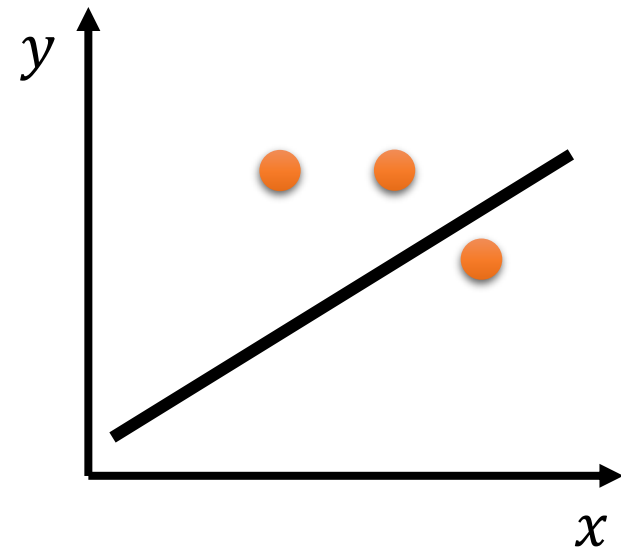
constrain  
too much



---- Real data distribution  
(not observable)

● Training data

● Testing data



# Overfitting

Sadly, overfitting can only be alleviated,  
but not completely avoided

# Today's Topics

- Terminology
- Error and Overfitting
- *Evaluation Methods*
- Performance Measure
- Bias and Variance

# Evaluate generalization performance

- We mentioned that we want to get a learner with a *small generalization error*
- How to evaluate the generalization performance of a learner?
  - Efficient and feasible experimental estimation method
  - Evaluation criteria for measuring model generalization ability

# Evaluate generalization performance

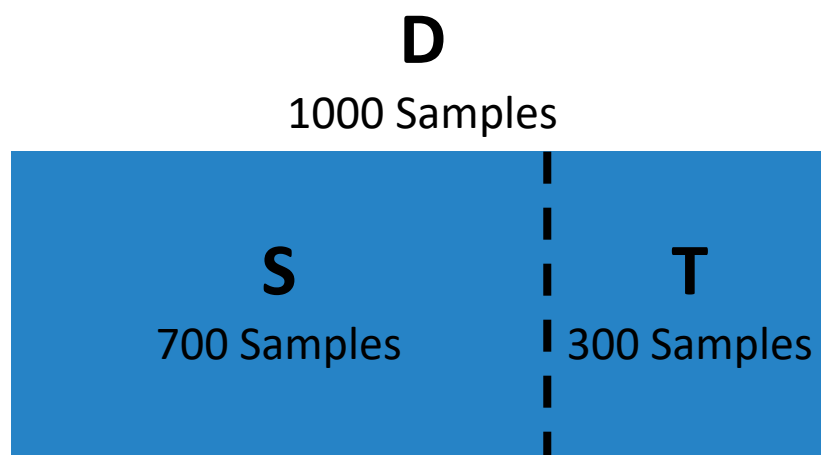
- We mentioned that we want to get a learner with a *small generalization error*
- How to evaluate the generalization performance of a learner?
  - *Efficient and feasible experimental estimation method*
  - Evaluation criteria for measuring model generalization ability

# Basic idea

- Use the test set to test the learner's ability to discriminate against new samples
- The test samples should not appear in the training samples as much as possible. **Why?**
- How to properly divide the data set to generate training set and test set?

# Hold-out 留出法

- Data set  $D$  is divided into two mutually exclusive sets
  - a training set  $S$ , and a testing set  $T$
- $D = S \cup T, S \cap T = \emptyset$
- Example: a binary classification problem



**If**

training on  $S$ , testing on  $T$ ,  
90 misclassification samples

**Then**

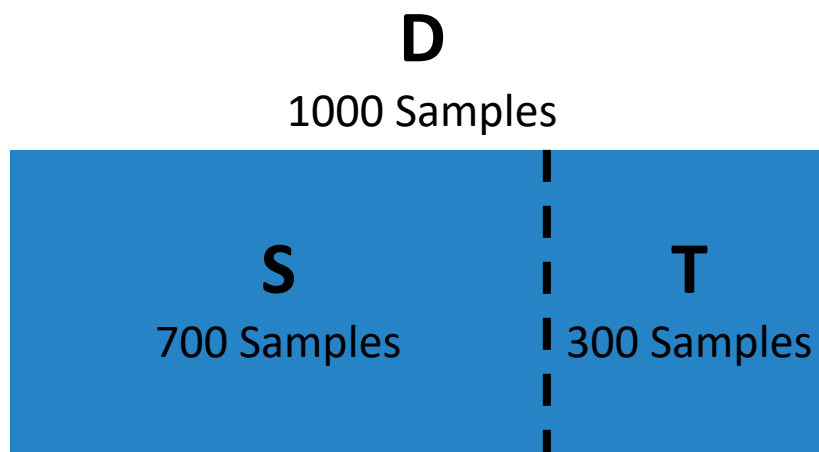
Error rate:  $(90/300) * 100\% = 30\%$

Accuracy:  $100\% - \text{Error rate} = 70\%$



# Hold-out 留出法

- When dividing the data set, try to ensure the **consistency of data distribution** as much as possible!
- 分层采样：保留类别比例



**If**

D contains 500 positive samples,  
500 negative samples

**Then**

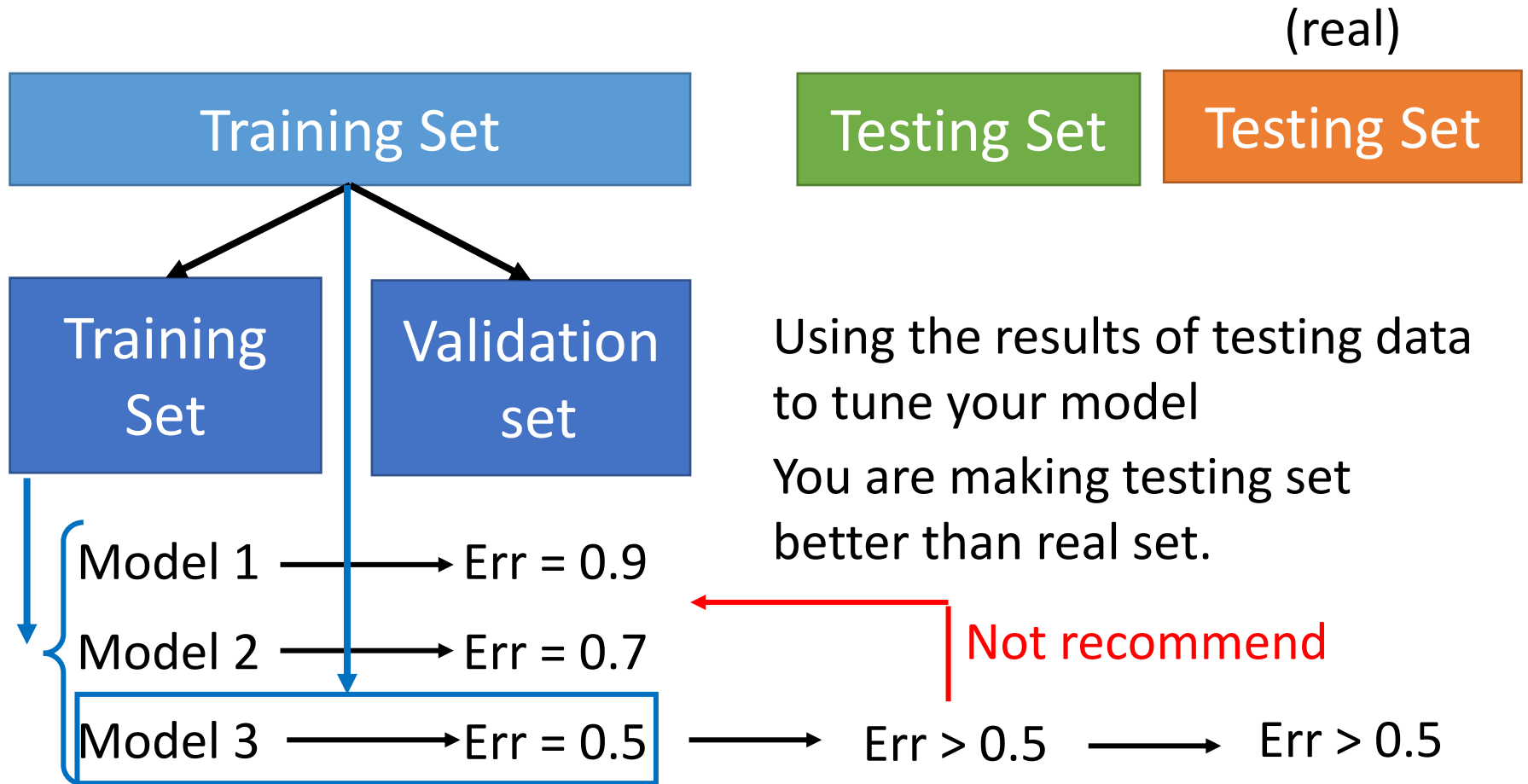
S: 350 positive samples,  
350 negative samples

T: 150 positive samples,  
150 negative samples

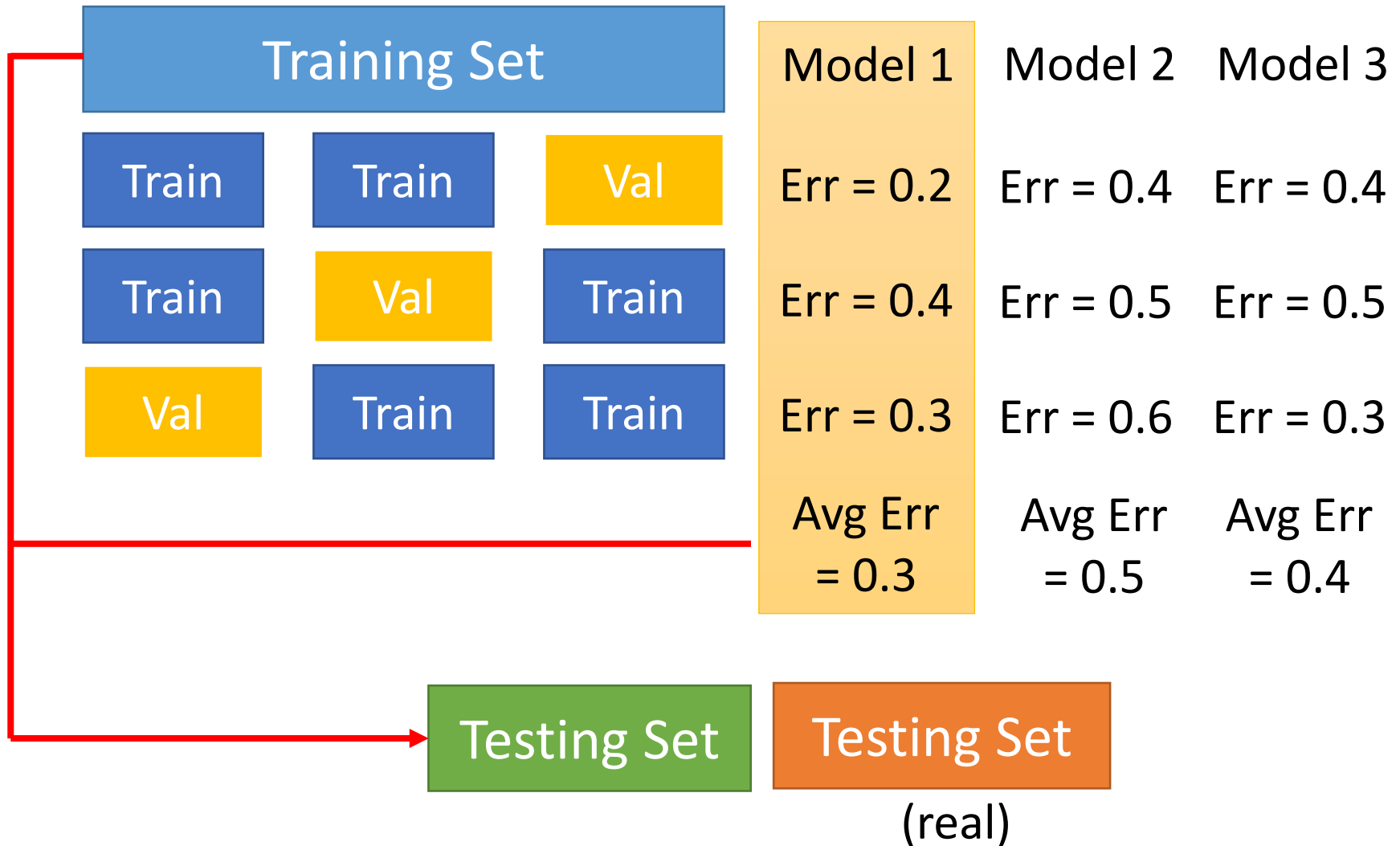
# Hold-out 留出法

- **Weakness:** We would like to get a model trained with  $D$ , but hold-out cannot do this due to the need to divide the dataset
  - $S \uparrow, T \downarrow$ , evaluation results may be unstable
  - $S \downarrow, T \uparrow$ , the model has a large deviation from the model trained by  $D$
- There is no perfect solution.
- Typically about **2/3 - 4/5** of the data is used for training.

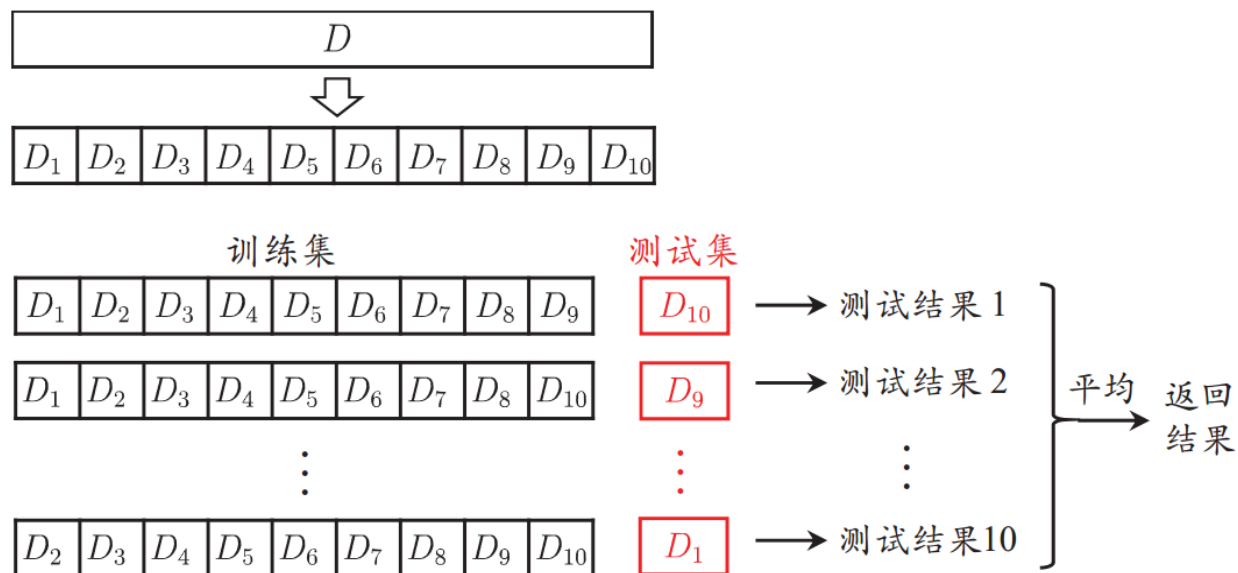
# Cross Validation 交叉验证法



# N-fold Cross Validation



# N-fold Cross Validation



10 折交叉验证示意图

In order to partition the dataset without bias, N-fold cross validation are usually performed out multiple times, e.g. 5次10折交叉验证

In this case, how many training do we need to carry out?

# N-fold Cross Validation

- We have used all data for training, and all data for testing, and used each data point the same number of times
- Cross-validation returns an unbiased estimate of the generalization error and its variance
- The value of  $N$  is important!
  - What if  $N$ =the number of samples ( $m$ )?
- **Leave-One-Out** 留一法
  - $m-1$  samples for training
  - 1 sample for testing
  - Accurate, but time-consuming

# Bootstrapping 自助法

- Training set  $D \rightarrow D'$  (pick a sample from  $D$   $m$  times)
- The probability of a sample  $a$  not picked is  $(1-1/m)^m$ 
  - $\lim_{m \rightarrow \infty} \left(1 - \frac{1}{m}\right)^m = \frac{1}{e} \approx 0.368$
  - $D'$ : training set
  - $D \setminus D'$ : test set
- Advantages
  - It's useful when dataset is small and training and test set are hard to construct
- Disadvantages
  - The training set  $D'$  has different distribution of  $D$ , which may introduce bias in evaluation

# Parameter Tuning

- It is *impossible* to exhaust all parameters. We need to select a range and change step for each parameter.
- Example
  - range- $\rightarrow$ [0,0.2] step- $\rightarrow$ 0.05.
  - Then we need to evaluate 5 parameters.
- The final model should be trained on dataset D (using all samples) before it can be submitted to users.



# Today's Topics

- Terminology
- Error and Overfitting
- Evaluation Methods
- *Performance Measure*
- Bias and Variance

# Evaluate generalization performance

- We mentioned that we want to get a learner with a *small generalization error*
- How to evaluate the generalization performance of a learner?
  - Efficient and feasible experimental estimation method
  - Evaluation criteria for measuring model generalization ability

# Evaluate generalization performance

- We mentioned that we want to get a learner with a *small generalization error*
- How to evaluate the generalization performance of a learner?
  - Efficient and feasible experimental estimation method
  - *Evaluation criteria for measuring model generalization ability*

# Performance Measure

- **Definition**

- Performance metric is an evaluation criterion to measure the generalization ability of the model, **reflecting the task requirements**
- Using different performance metrics often leads to different judgments.

In the prediction task, if a sample set

$$D = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$$

is given, then we evaluate the performance of the learner  $f$  by comparing the predicted result  $f(x)$  with the ground truth.

# Measure for Regression

- Mean Square Error, MSE
- 均方误差

$$E(f; D) = \frac{1}{m} \sum_{i=1}^m (f(\mathbf{x}_i) - y_i)^2$$

# Measure for Classification


- Error rate & Accuracy

Error rate

$$E(f; D) = \frac{1}{m} \sum_{i=1}^m \mathbb{I}(f(\mathbf{x}_i) \neq y_i)$$

Accuracy

$$\begin{aligned} \text{acc}(f; D) &= \frac{1}{m} \sum_{i=1}^m \mathbb{I}(f(\mathbf{x}_i) = y_i) \\ &= 1 - E(f; D) . \end{aligned}$$



Are the enough  
for all cases?

# Thinking

- **Can all of these cases in the “picking watermelon problem” be measured using Accuracy/Error rate?**
- 有多少比例的西瓜被判别错误？
- 挑出的西瓜中有多少比例是好瓜？
- 所有好瓜中有多少比例被挑了出来？



# Precision and Recall

- Suitable for information retrieval, web search scenarios

A "**confusion matrix**" can be obtained by combining the statistics of the true labels and the predicted results

Ground truth	Predicted Result	
	True	False
True	<i>TP</i>	<i>FN</i>
False	<i>FP</i>	<i>TN</i>

Precision  $P = \frac{TP}{TP + FP}$

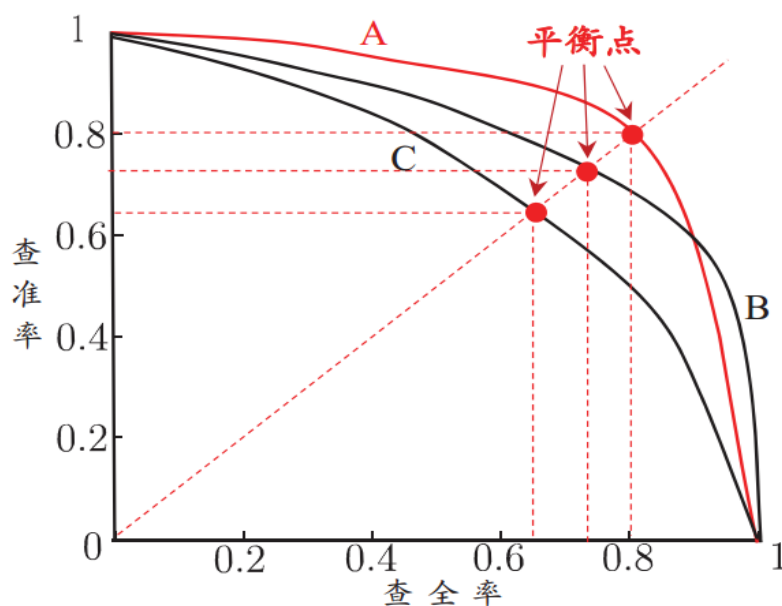
Recall  $R = \frac{TP}{TP + FN}$

Recall and precision are contradictory measures. **Why?**



# Precision and Recall

- According to the prediction results of the learner, the samples are sorted according to the probability of positive examples. If all samples are predicted as positive examples one by one, then the precision-recall curve, referred to as "**P-R curve**", can be obtained.



P-R曲线与平衡点示意图

## Break-even point

The value of "**precision = recall**" on the curve  
Used to measure the performance of  
classifiers with crossed P-R curves.

Comprehensive  
consideration of recall  
and precision

# F1 Score

- BEP is too simplistic
- More commonly used than P-R curve break-even point

$$F1 = \frac{2 * P * R}{P + R} = \frac{2 * TP}{\text{the number of samples} + TP - TN}$$

- A more general form  $F_\beta$

$$F_\beta = \frac{(1 + \beta^2) * P * R}{(\beta^2 * P) + R}$$

$\beta > 0$  measures the relative importance of recall to precision

$\beta = 1$ : F1 Score

$\beta > 1$ : Recall is more important (Information retrieval)

$\beta < 1$ : Precision is more important (Recommender system)

# F1 Score is essentially a harmonic mean!

- F1 Score is the harmonic mean(调和平均) of  $P$  and  $R$

$$\frac{1}{F1} = \frac{2 * P * R}{P + R} = \frac{1}{2} \left( \frac{1}{P} + \frac{1}{R} \right)$$

- $F_\beta$  is a weighted harmonic mean (加权调和平均)

$$\frac{1}{F_\beta} = \frac{(1+\beta^2)*P*R}{(\beta^2*P)+R} = \frac{1}{1+\beta^2} \left( \frac{1}{P} + \frac{\beta^2}{R} \right)$$

# ROC

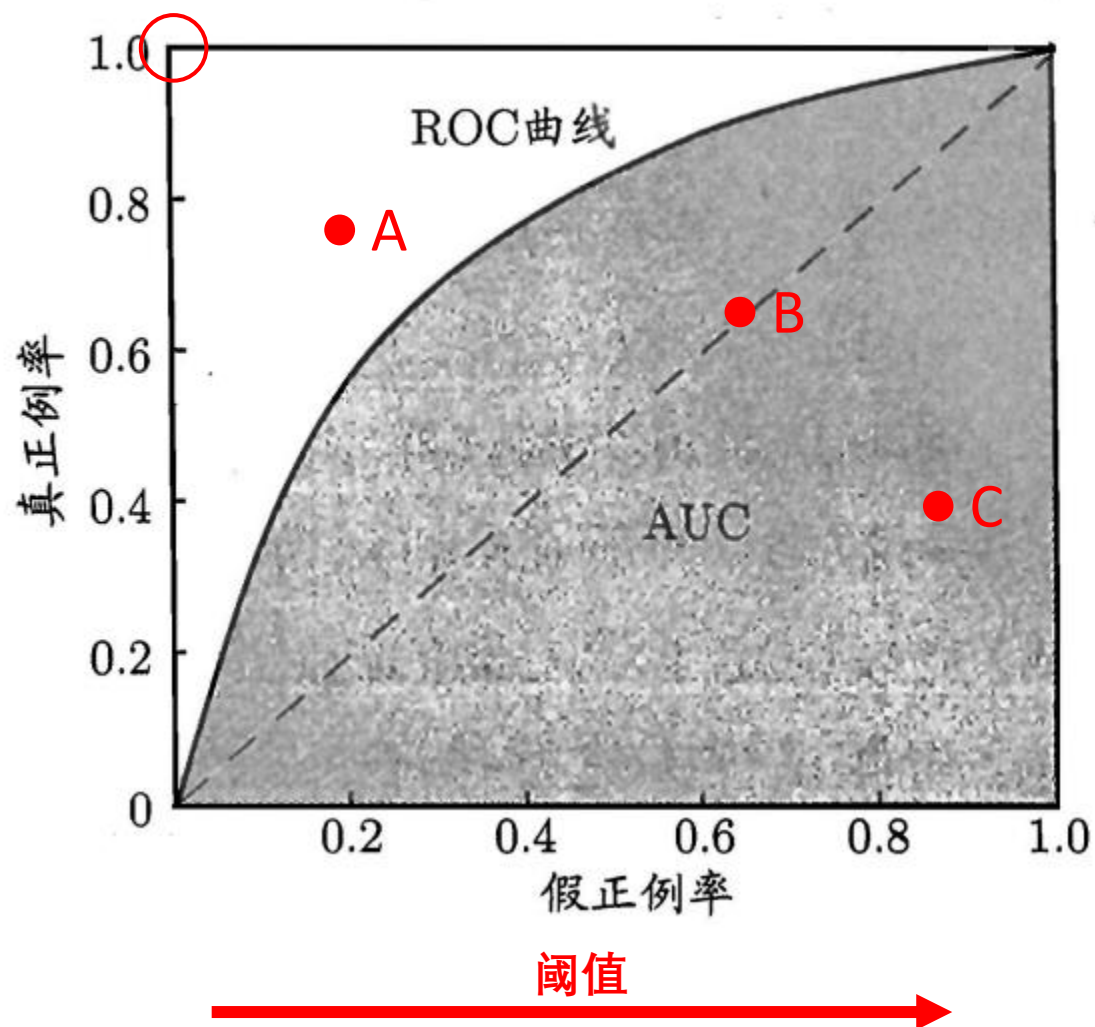
- Receiver Operating Characteristic(受试者工作特征)
- The steps of sorting the samples and predicting the samples as positive examples are consistent with the PR curve
- But the axes change

	Name	Expression	
Vertical axis 纵轴	True Positive Rate/TPR 真正例率	$\frac{TP}{TP + FN}$	=Recall
Horizontal axis 横轴	False Positive Rate/FPR 假正例率	$\frac{FP}{TN + FP}$	

Ground truth	Predicted Result	
	True	False
True	$TP$	$FN$
False	$FP$	$TN$

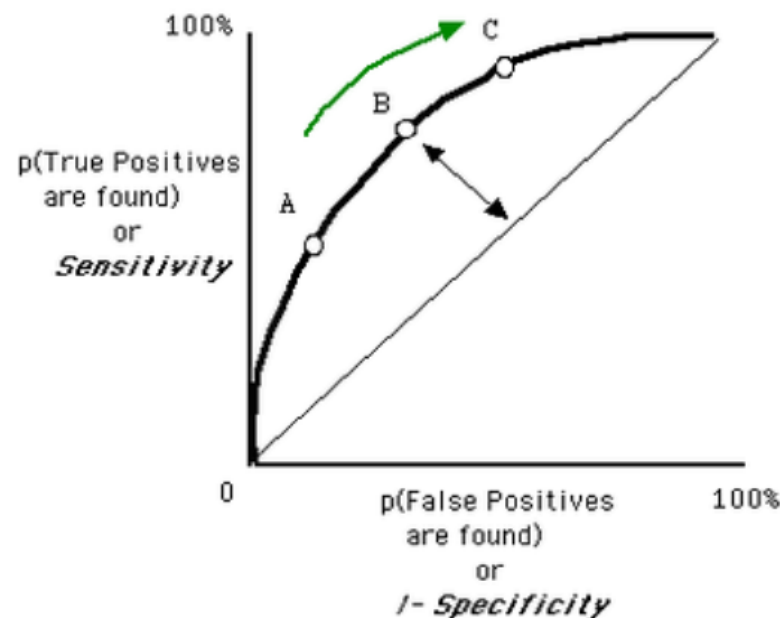
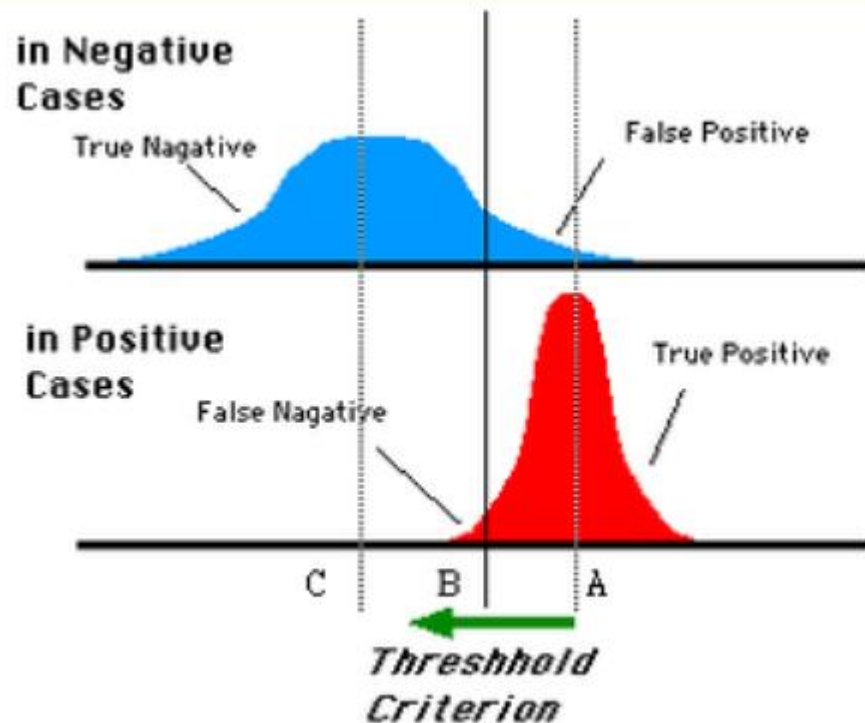
Eg. 医疗诊断中  
真正利率=确诊率  
假正利率=误诊率

# ROC



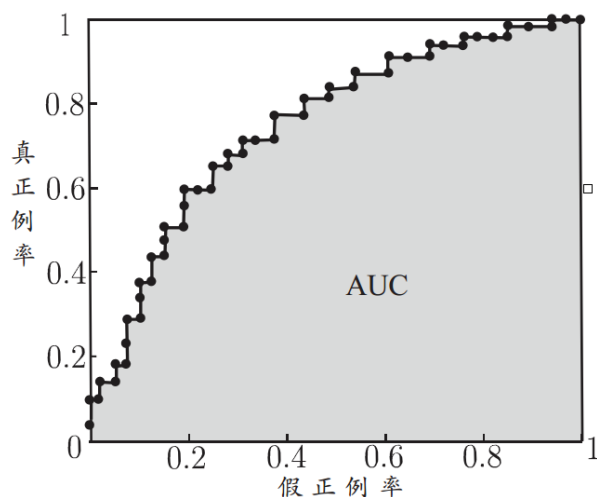
# ROC

- ROC represents the performance of a classifier under different thresholds



# ROC

- Can such a smooth ROC curve be generated in real tasks?
- In real-world tasks, an approximate ROC curve is generally drawn with limited test examples.



**Drawing steps** ( $m$  positive samples,  $n$  negative samples)

- ① Sample sorting
- ② Set the classification threshold to the predicted value
- ③ Let the coordinates of the previous marker point be  $(x, y)$ , then for the current sample:
  - True Positive  $\Rightarrow \left(x, y + \frac{1}{m}\right)$
  - False Positive  $\Rightarrow \left(x + \frac{1}{n}, y\right)$
- ④ Connect adjacent points

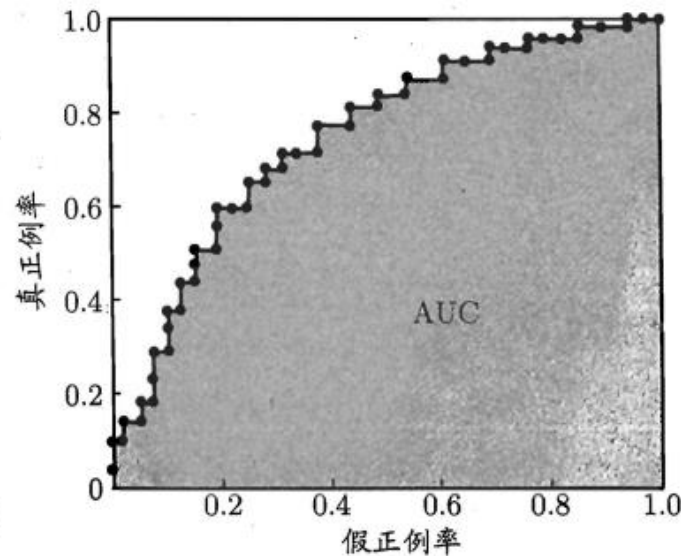
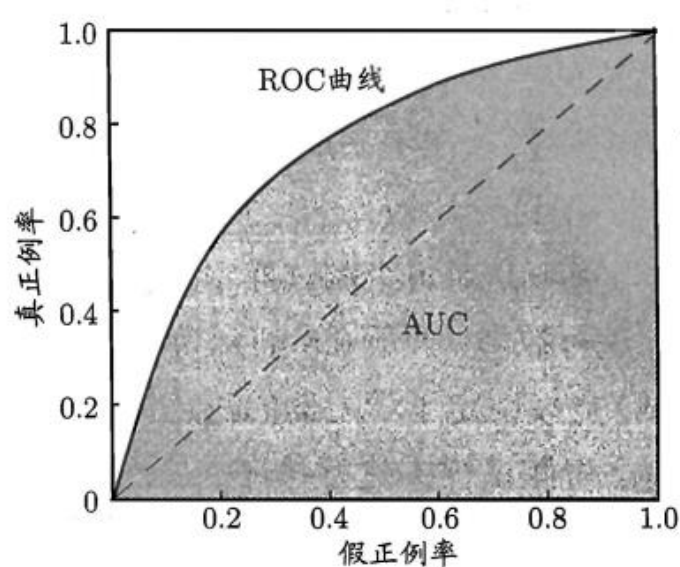
# ROC

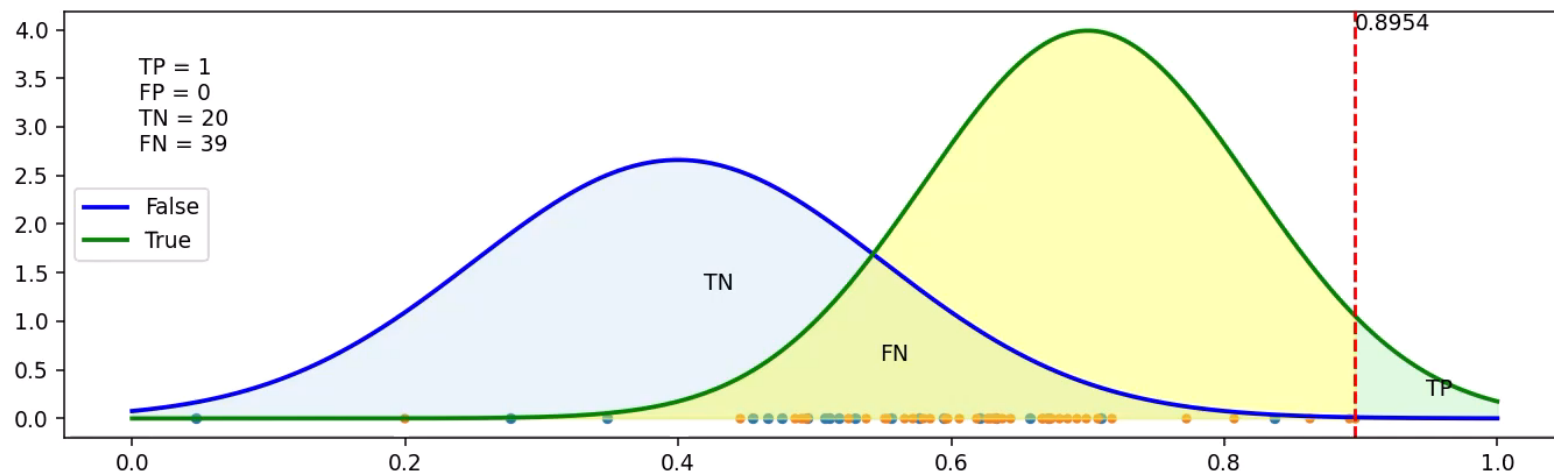
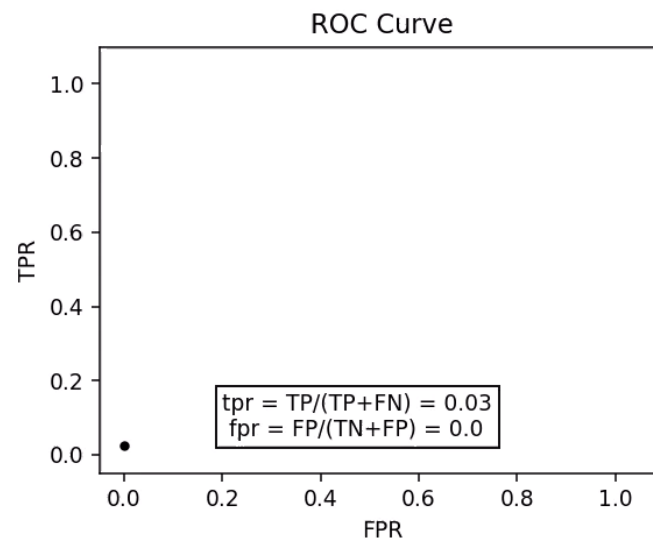
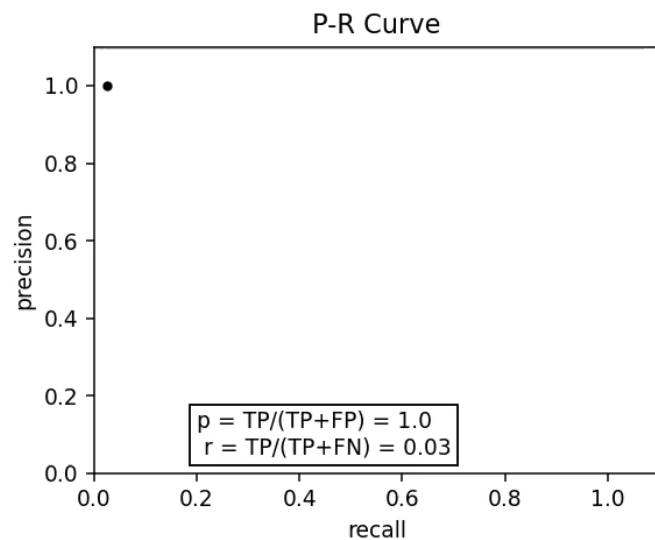
- If the ROC curve of learner A is completely "wrapped" by the curve of learner B, it can be asserted that the performance of B is better than A.
- But how to judge when two ROC curves cross?



# AUC: Area Under ROC Curve

- AUC measures the *ranking quality* of sample predictions
- The quality of the ranking itself can reflect the generalization performance of the learner in the "general case".





# Thinking

- Does diagnosing a patient as healthy have the same consequences as diagnosing a healthy person as a patient?
- Does an access control system have the same consequences for keeping passable people out as it does for strangers in?



# Cost-sensitive Error Rate

## 代价敏感错误率

- In order to weigh the different losses caused by different types of errors, **unequal costs** can be assigned to errors.
- Cost Matrix:  $cost_{ij}$  represents the cost of predicting the  $i$ -th class sample as the  $j$ -th class sample

真实类别	预测类别	
	第 0 类	第 1 类
第 0 类	0	$cost_{01}$
第 1 类	$cost_{10}$	0

- Cost-sensitive Error Rate: Minimize the total cost

$$E(f; D; cost) = \frac{1}{m} \left( \sum_{\mathbf{x}_i \in D^+} \mathbb{I}(f(\mathbf{x}_i) \neq y_i) \times cost_{01} + \sum_{\mathbf{x}_i \in D^-} \mathbb{I}(f(\mathbf{x}_i) \neq y_i) \times cost_{10} \right).$$

# Performance Measure

- **Summary**

MSE for Regression  $E(f; D) = \frac{1}{m} \sum_{i=1}^m (f(\mathbf{x}_i) - y_i)^2$

Accuracy for Classification  $\text{acc}(f; D) = \frac{1}{m} \sum_{i=1}^m \mathbb{I}(f(\mathbf{x}_i) = y_i)$   
 $= 1 - E(f; D) .$

Precision  $P = \frac{TP}{TP + FP}$

Recall  $R = \frac{TP}{TP + FN}$

$$F1 = \frac{2 * P * R}{P + R} = \frac{2 * TP}{\text{the number of samples} + TP - TN}$$

# Performance Measure

- Have a try!

Label	0	0	0	0	1	1	1	1	2	2	2	2
Prediction	0	1	0	2	1	0	1	2	0	0	1	2

## Calculate

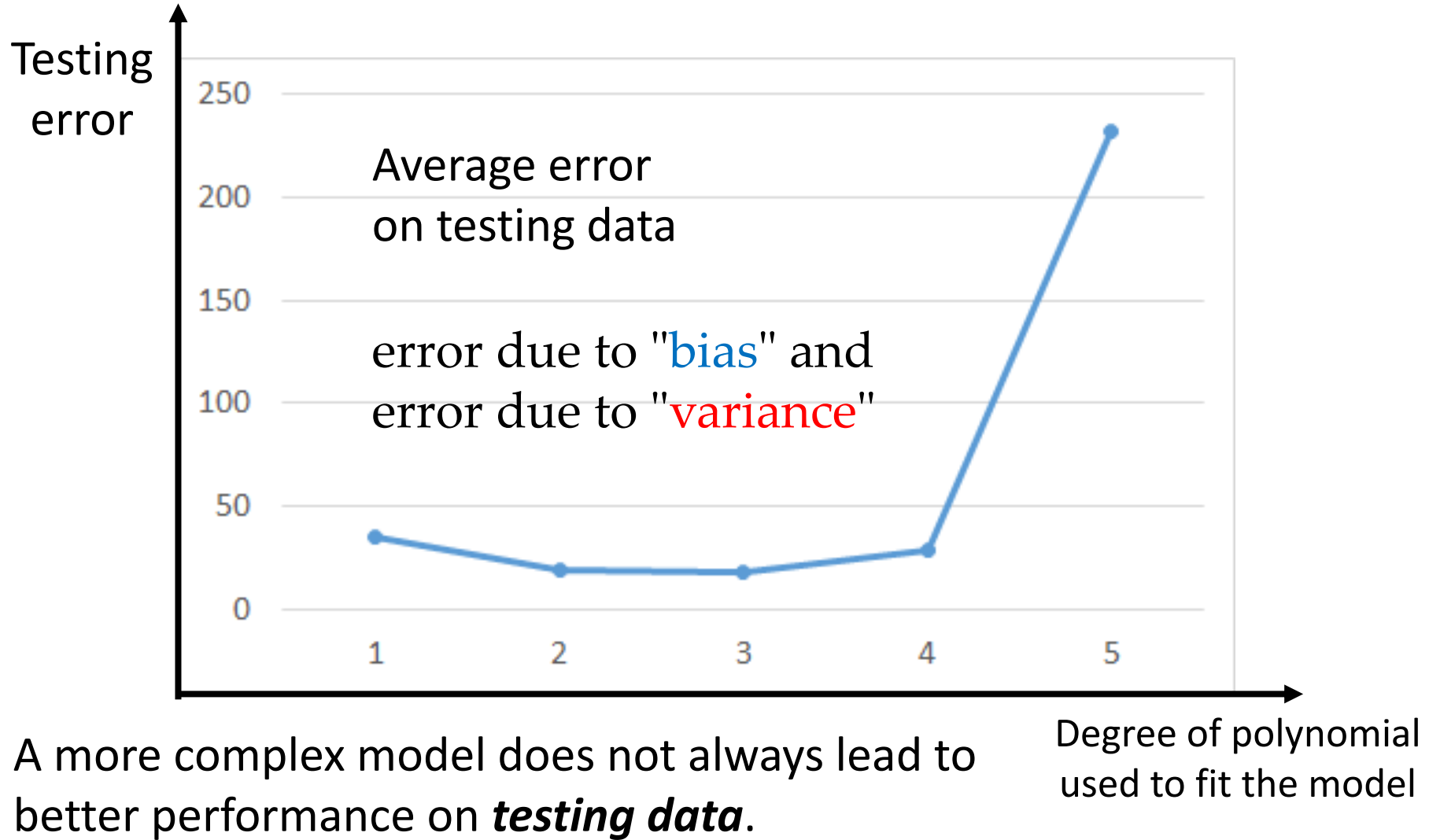
- Accuracy
- Precision(Class 1)
- Recall(Class2)
- F1 Score(Class0)

5/12, 1/2, 1/4, 4/9

# Today's Topics

- Terminology
- Error and Overfitting
- Evaluation Methods
- Performance Measure
- *Bias and Variance*

# Where does the error come from



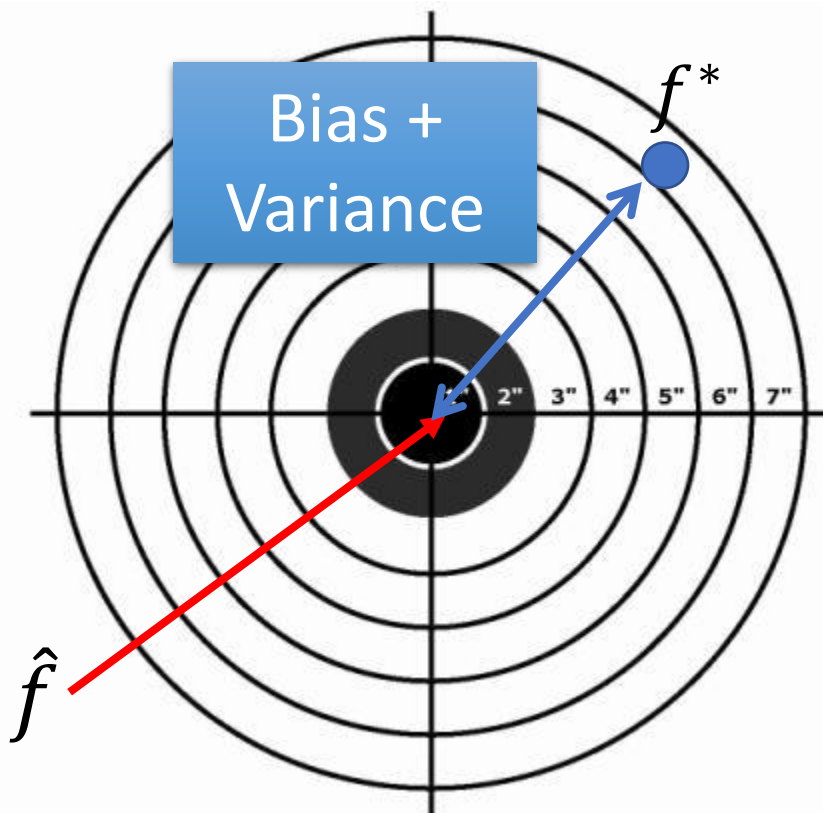


# Estimator

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

From training data,  
we find  $\hat{f}$

$\hat{f}^*$  is an estimator of  $\hat{f}$



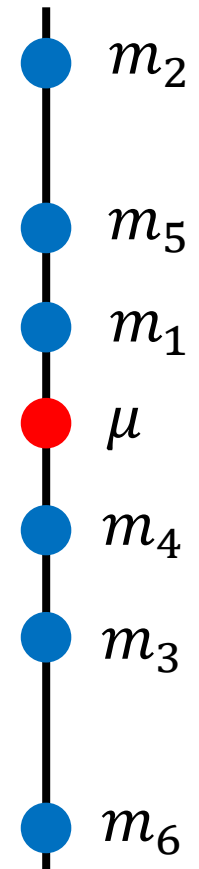
# Bias and Variance of Estimator

- Estimate the mean of a variable  $x$ 
  - assume the mean of  $x$  is  $\mu$
  - assume the variance of  $x$  is  $\sigma^2$
- Estimator of mean  $\mu$ 
  - Sample  $N$  points:  $\{x^1, x^2, \dots, x^N\}$

$$m = \frac{1}{N} \sum_n x^n \neq \mu$$

$$E[m] = E\left[\frac{1}{N} \sum_n x^n\right] = \frac{1}{N} \sum_n E[x^n] = \mu$$

unbiased



# Bias and Variance of Estimator

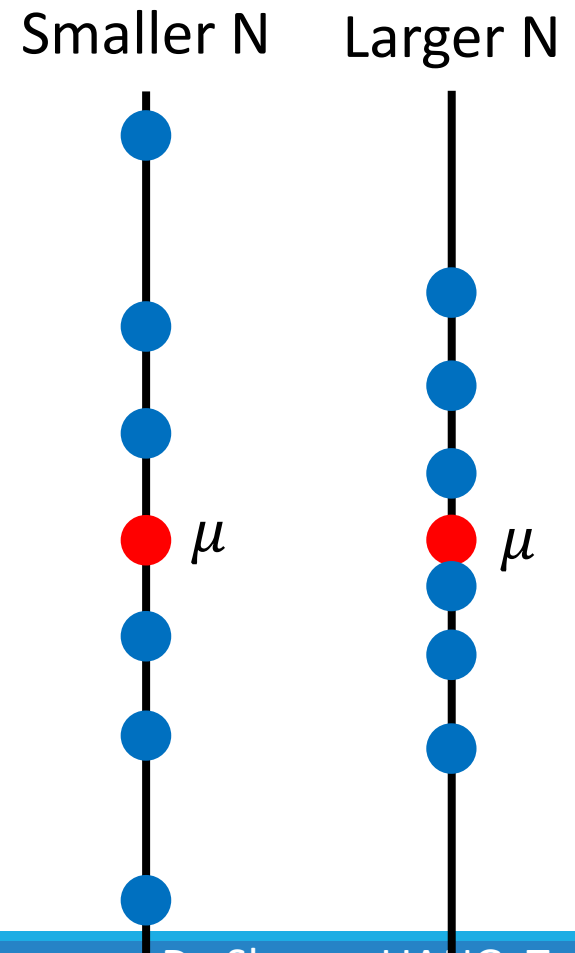
- Estimate the mean of a variable  $x$ 
  - assume the mean of  $x$  is  $\mu$
  - assume the variance of  $x$  is  $\sigma^2$
- Estimator of mean  $\mu$ 
  - Sample  $N$  points:  $\{x^1, x^2, \dots, x^N\}$

$$m = \frac{1}{N} \sum_n x^n \neq \mu$$

$$\text{Var}[m] = \frac{\sigma^2}{N}$$

Variance depends  
on the number of  
samples

unbiased



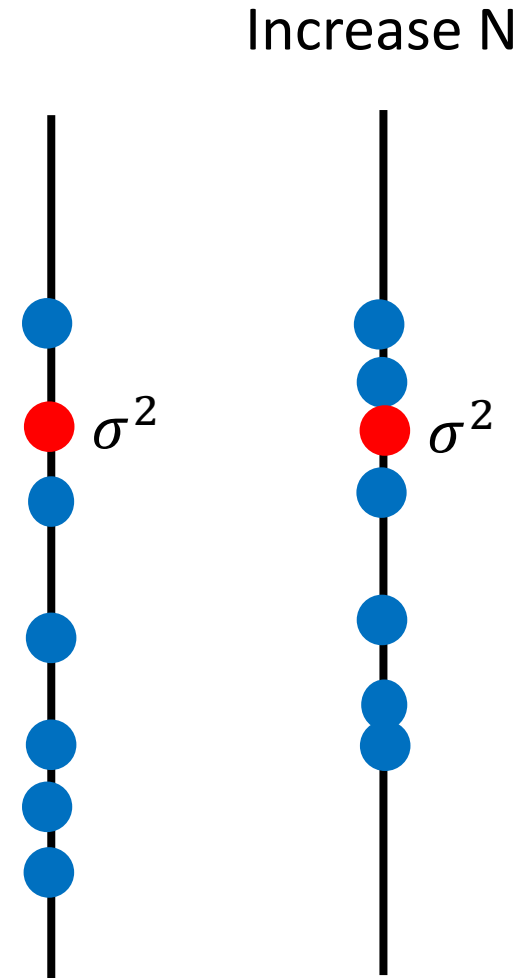
# Bias and Variance of Estimator

- Estimate the mean of a variable  $x$ 
  - assume the mean of  $x$  is  $\mu$
  - assume the variance of  $x$  is  $\sigma^2$
- Estimator of variance  $\sigma^2$ 
  - Sample  $N$  points:  $\{x^1, x^2, \dots, x^N\}$

$$m = \frac{1}{N} \sum_n x^n \quad s^2 = \frac{1}{N} \sum_n (x^n - m)^2$$

Biased estimator

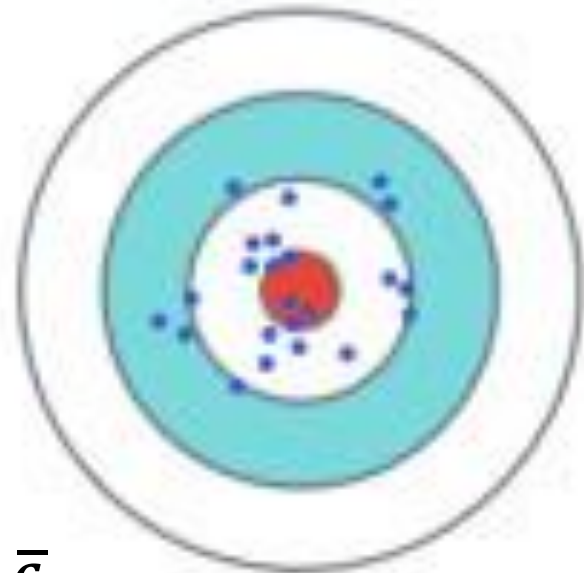
$$E[s^2] = \frac{N-1}{N} \sigma^2 \neq \sigma^2$$



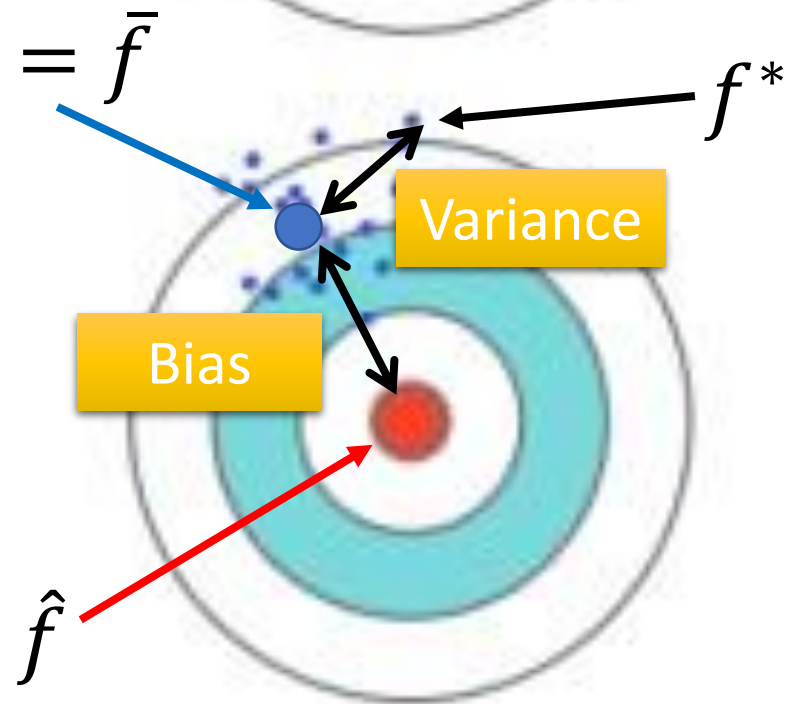
Low Variance

High Variance

Low Bias



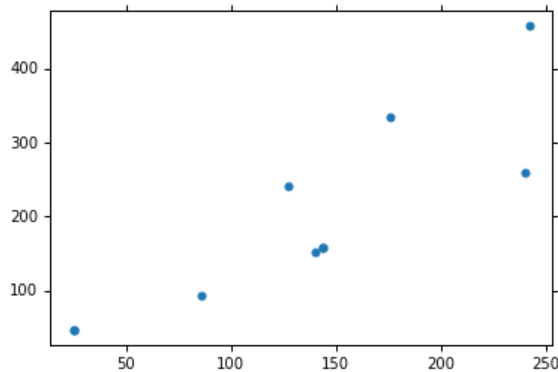
High Bias



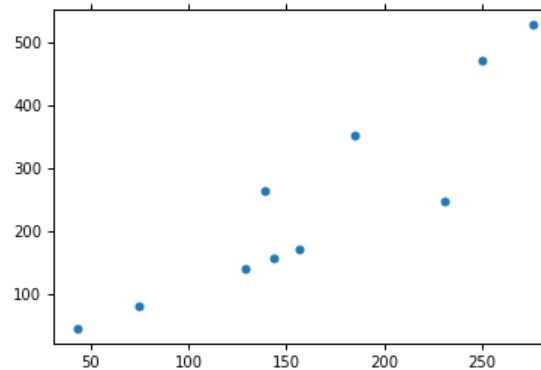
# Collecting Data

- We are collecting training data to find  $f^*$

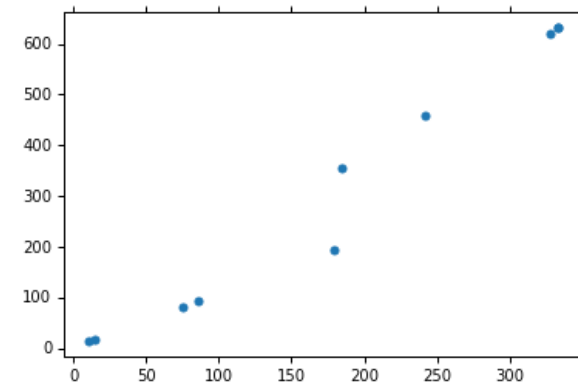
Scene 1



Scene 2



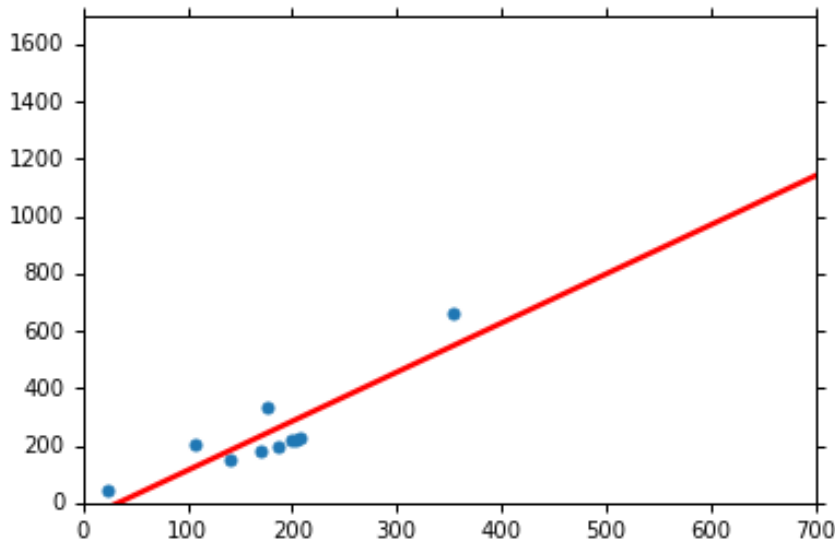
Scene 3



# Collecting Data

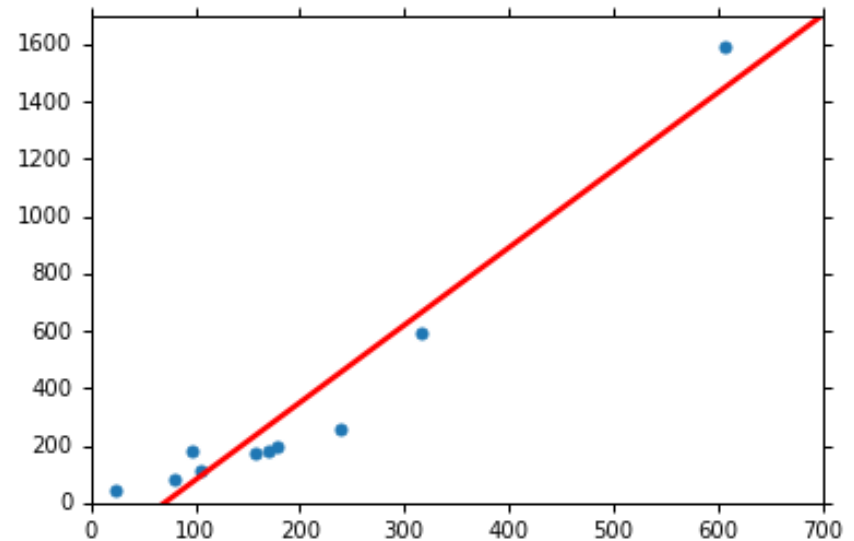
- In different scenes, we use the same model, but obtain different  $f^*$

Scene 123



$$y = b + w \cdot x$$

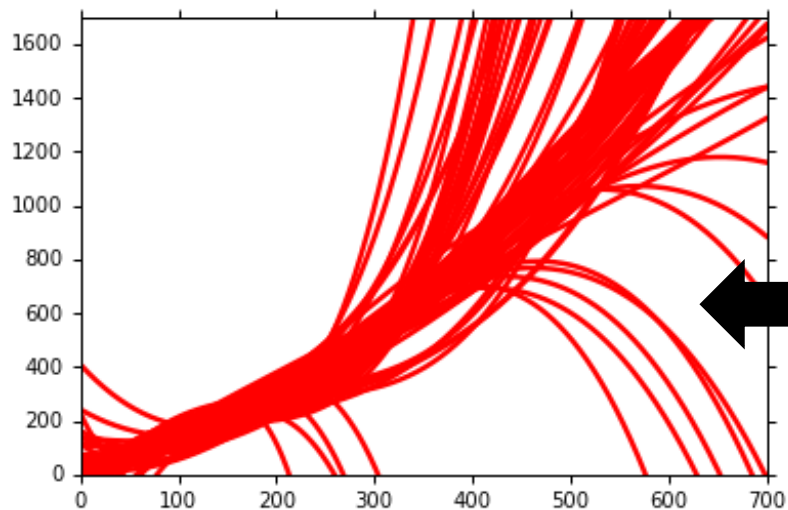
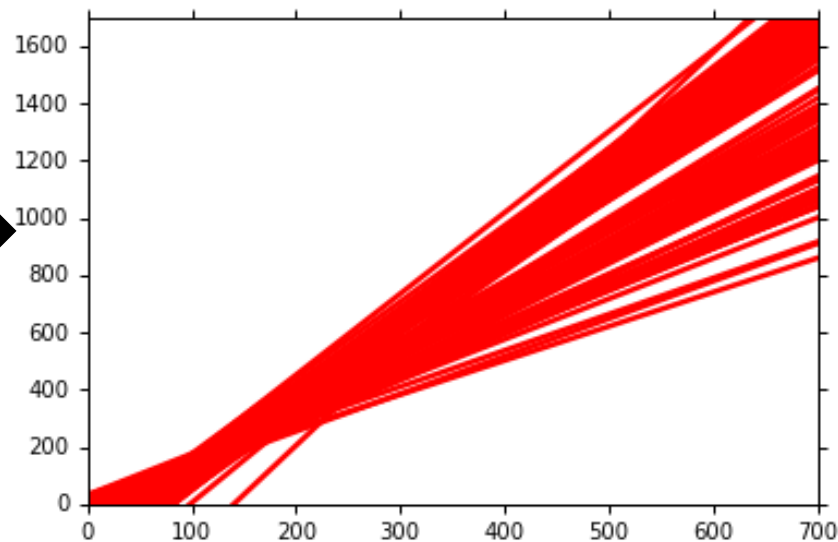
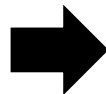
Scene 345



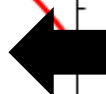
$$y = b + w \cdot x$$

$f^*$  in 100 scenes

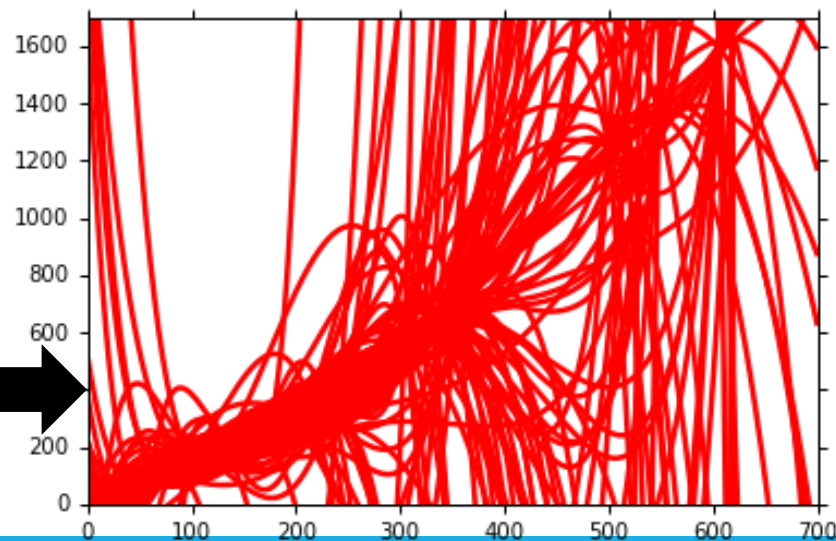
$$y = b + w \cdot x$$



$$y = b + w_1 \cdot x + w_2 \cdot (x)^2 + w_3 \cdot (x)^3$$



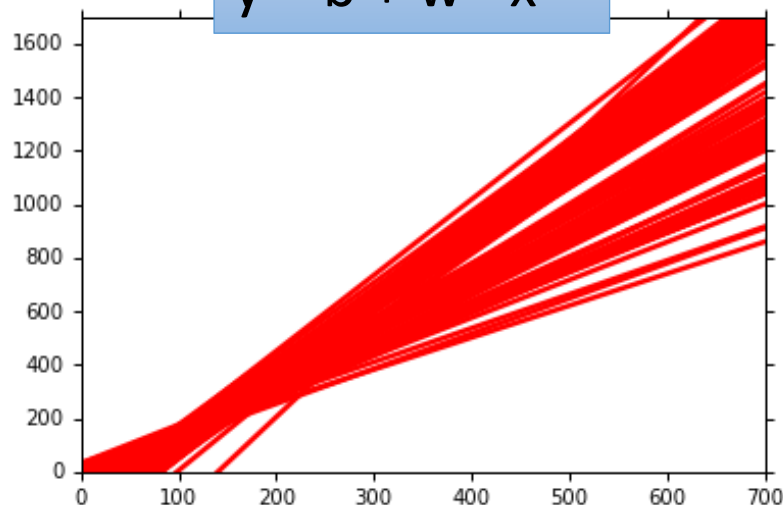
$$y = b + w_1 \cdot x + w_2 \cdot (x)^2 + w_3 \cdot (x)^3 + w_4 \cdot (x)^4 + w_5 \cdot (x)^5$$



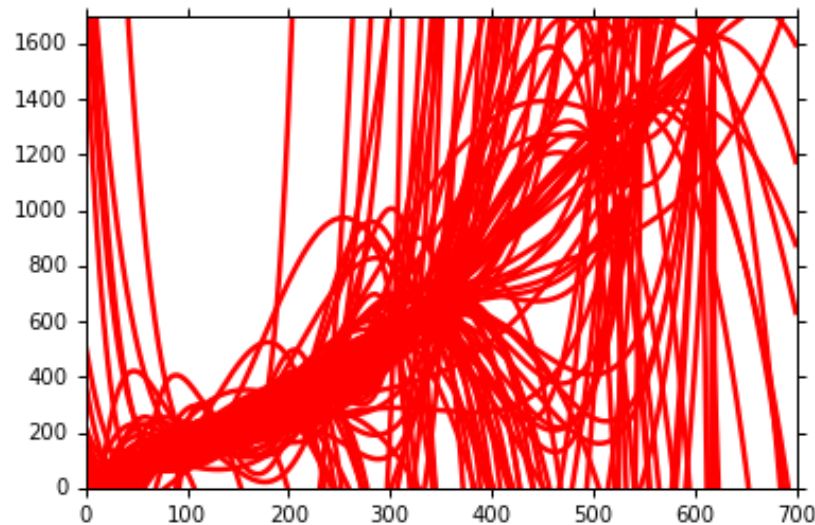


# Variance

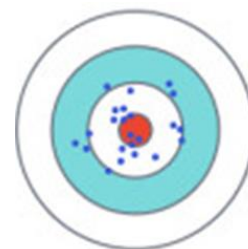
$$y = b + w \cdot x$$



$$y = b + w_1 \cdot x + w_2 \cdot (x)^2 + w_3 \cdot (x)^3 + w_4 \cdot (x)^4 + w_5 \cdot (x)^5$$



Small  
Variance



Large  
Variance

Simpler model is less influenced by the sampled data

Consider the extreme case  $f(x) = c$

# Bias

$$E[f^*] = \bar{f}$$

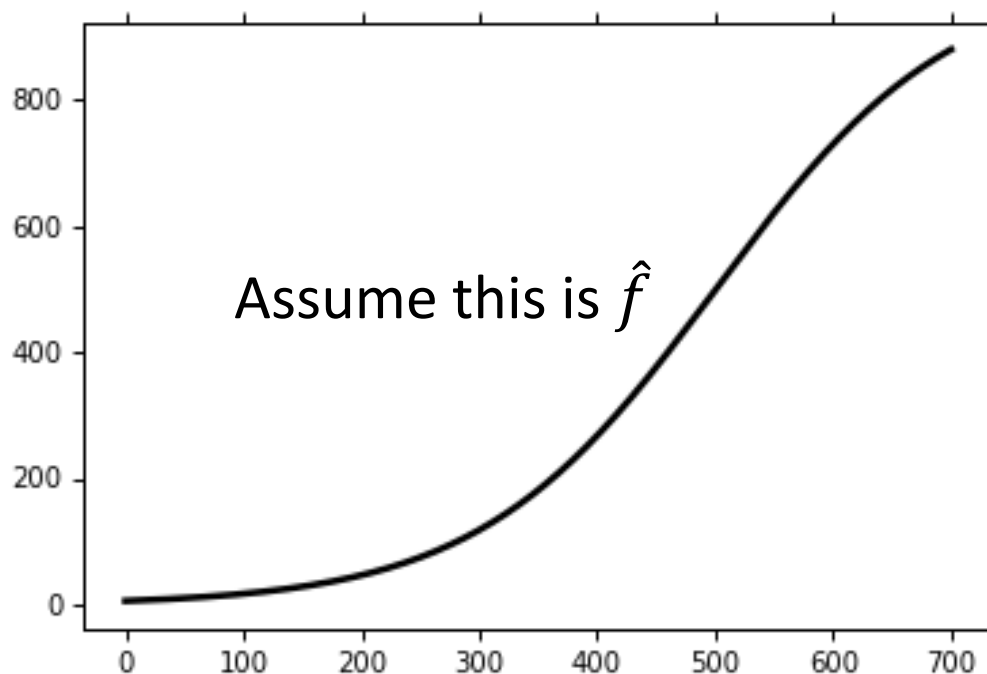
- Bias: If we average all the  $f^*$ , is it close to  $\hat{f}$



Large  
Bias



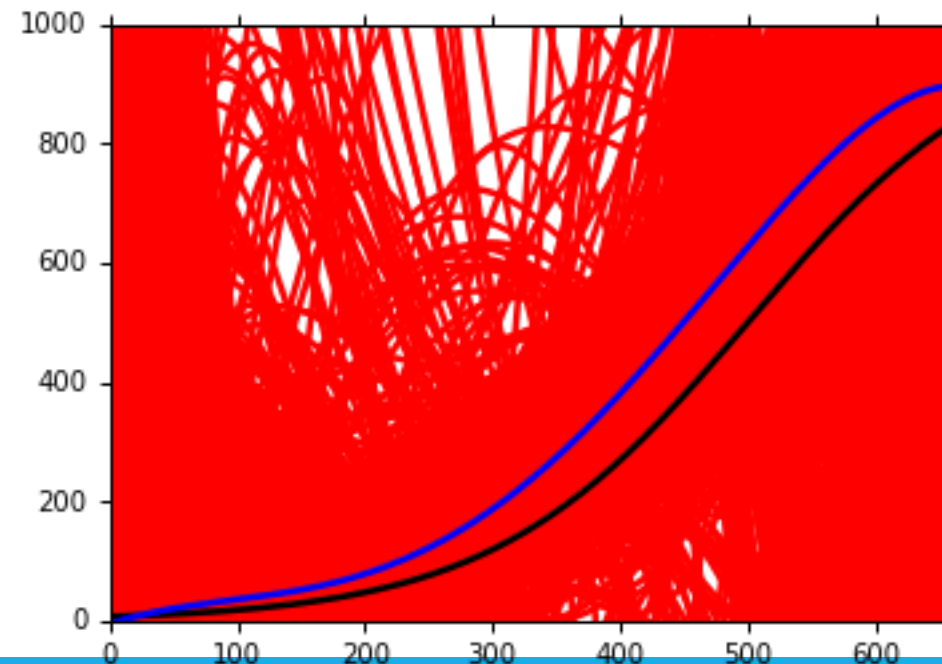
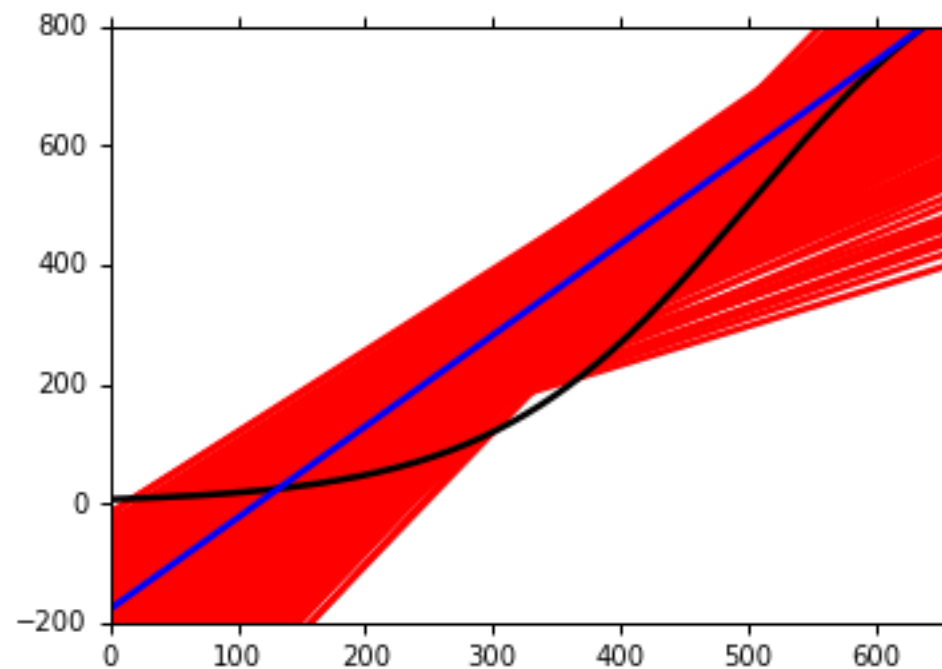
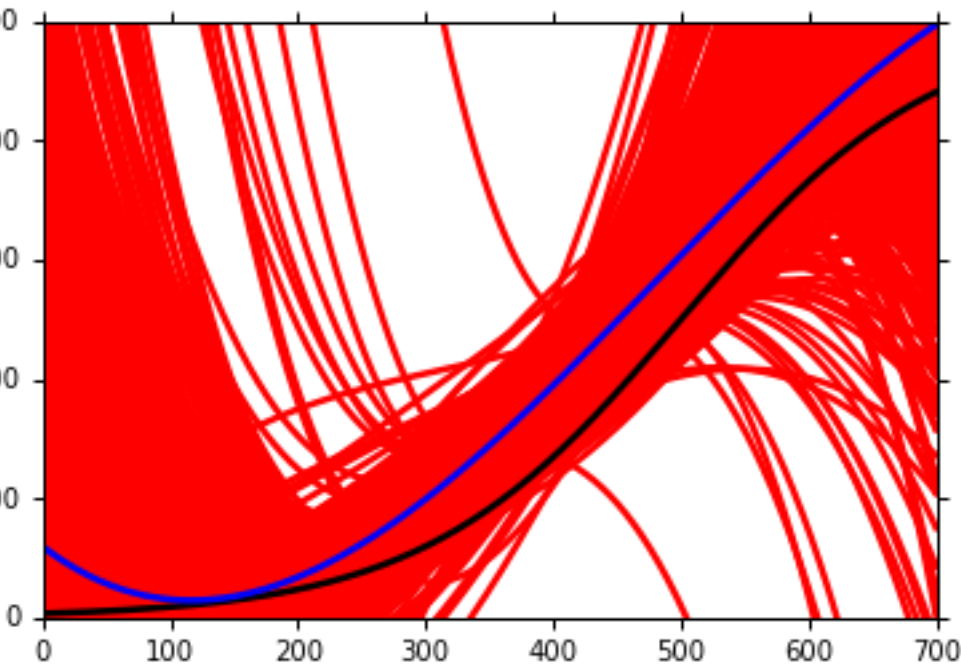
Small  
Bias



Black curve: the true function  $\hat{f}$

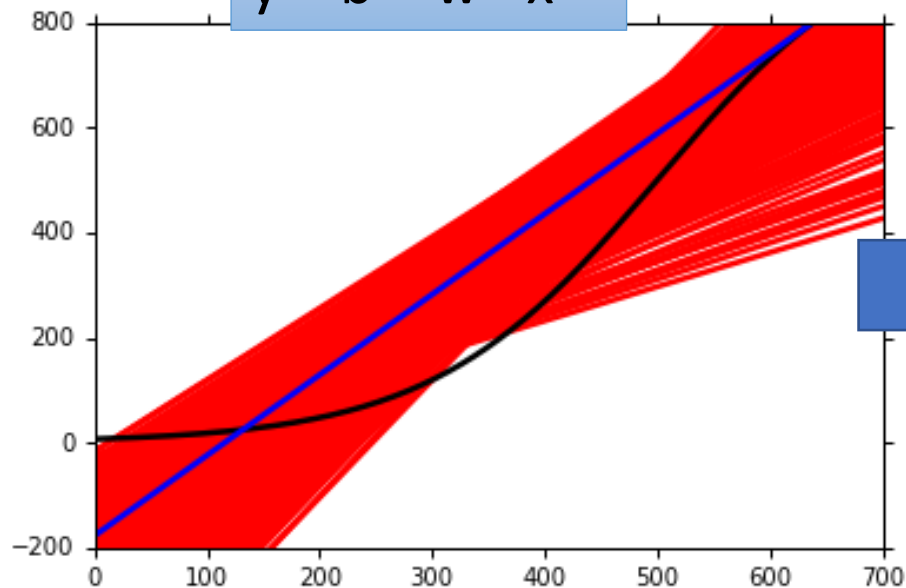
Red curves: 5000  $f^*$

Blue curve: the average of 5000  $f^*$   
 $= \bar{f}$

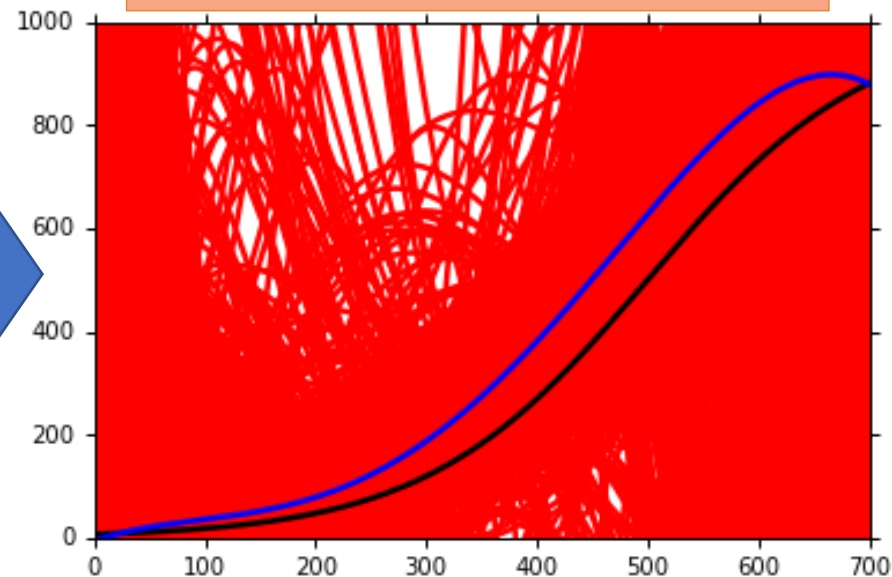


# Bias

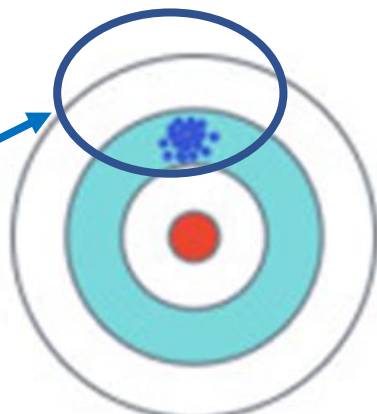
$$y = b + w \cdot x$$



$$y = b + w_1 \cdot x + w_2 \cdot (x)^2 + w_3 \cdot (x)^3 + w_4 \cdot (x)^4 + w_5 \cdot (x)^5$$

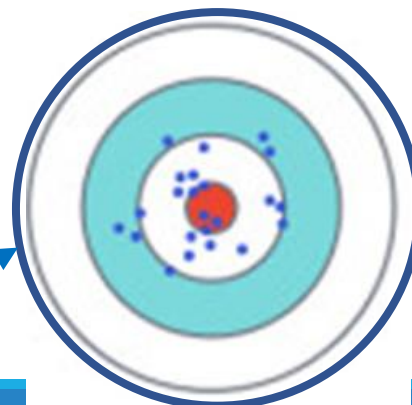


model



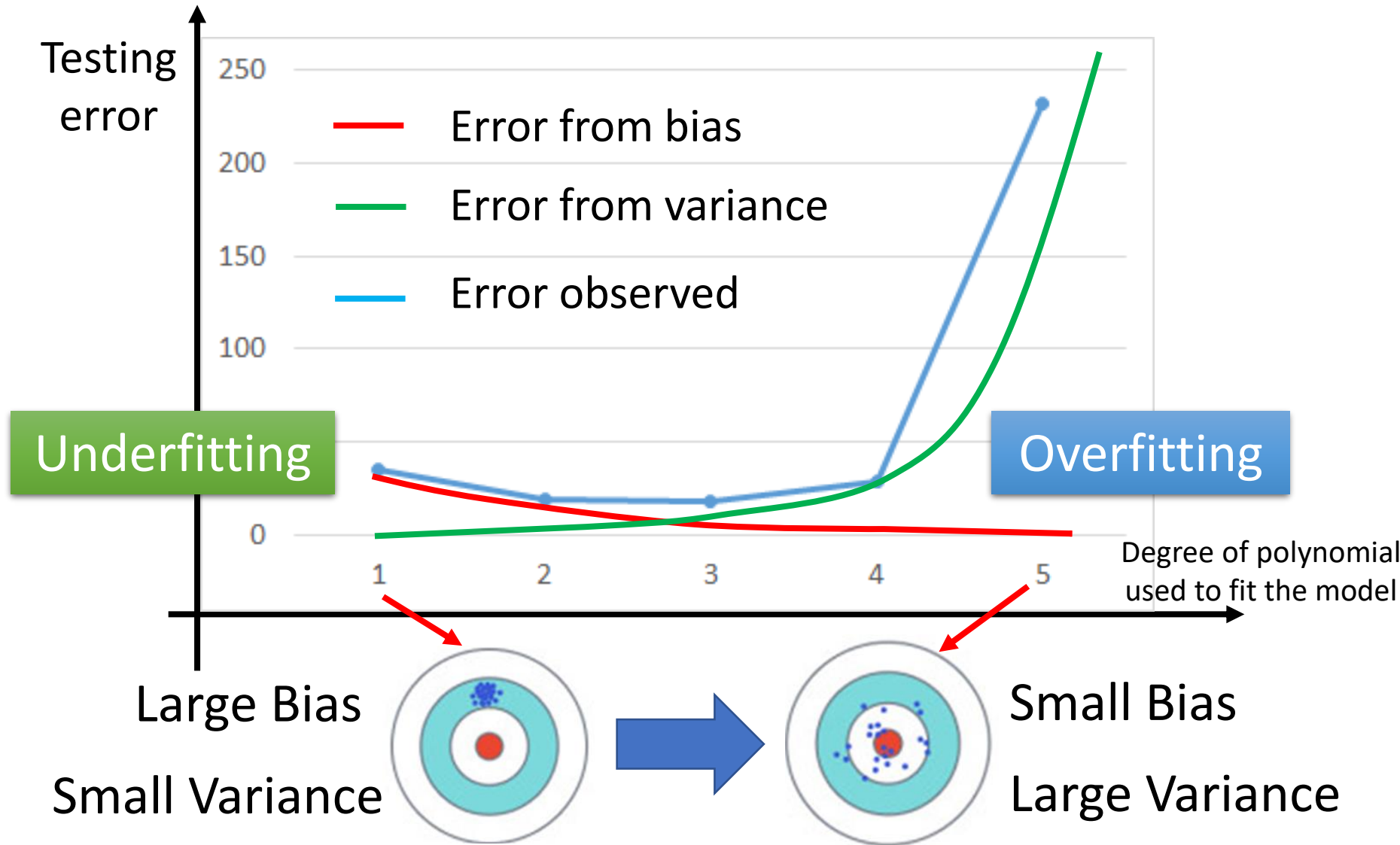
Large  
Bias

model



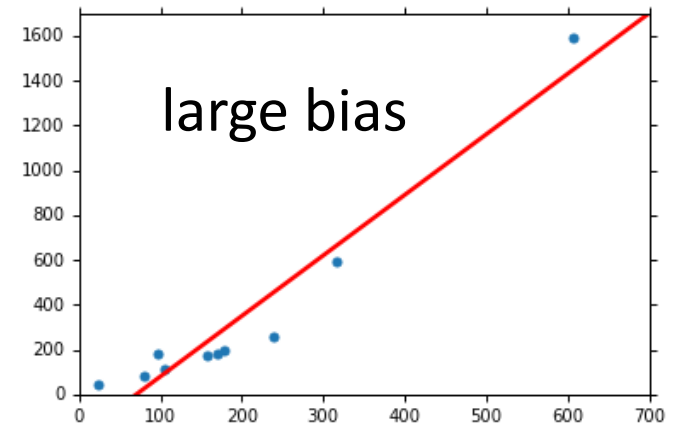
Small  
Bias

# Bias v.s. Variance



# What to do with large bias?

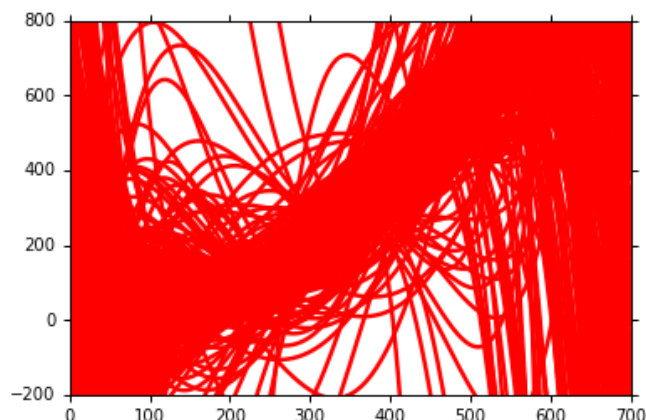
- Diagnosis:
  - If your model cannot even fit the training examples, then you have large bias **Underfitting**
  - If you can fit the training data, but large error on testing data, then you probably have large variance **Overfitting**
- For bias, redesign your model:
  - Add more features as input
  - A more complex model



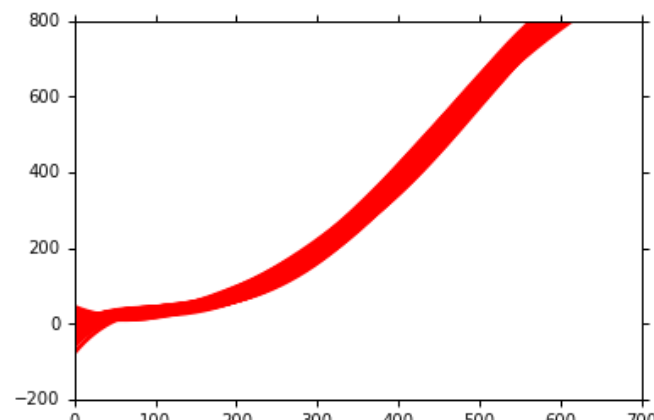
# What to do with large variance?

- More data

Very effective,  
but not always  
practical

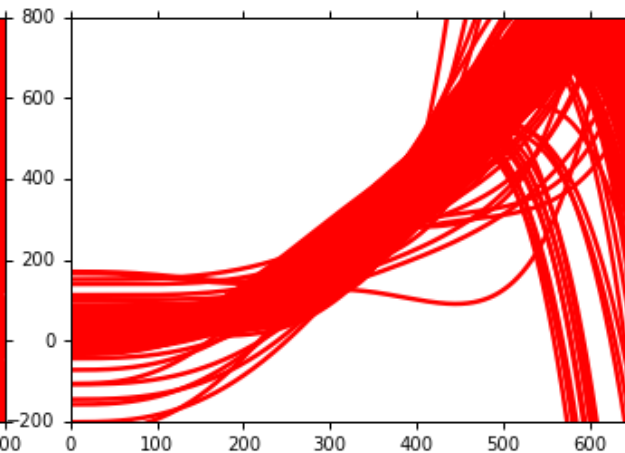
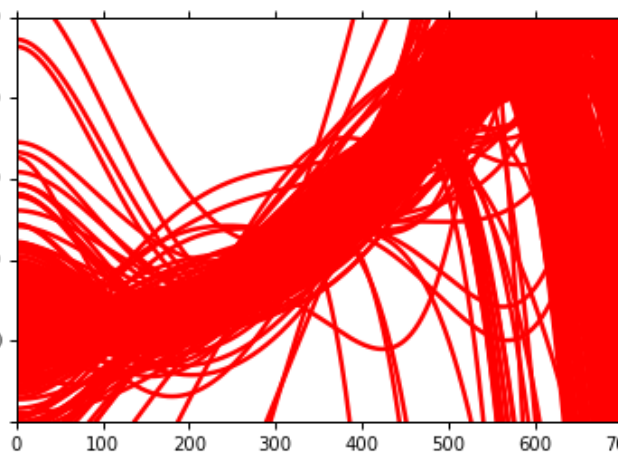
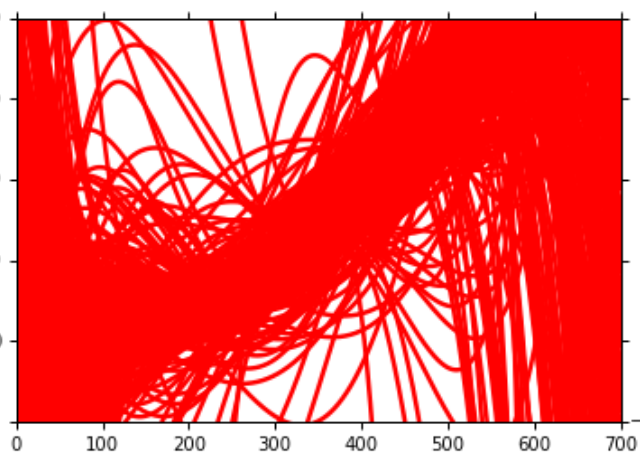


10 examples

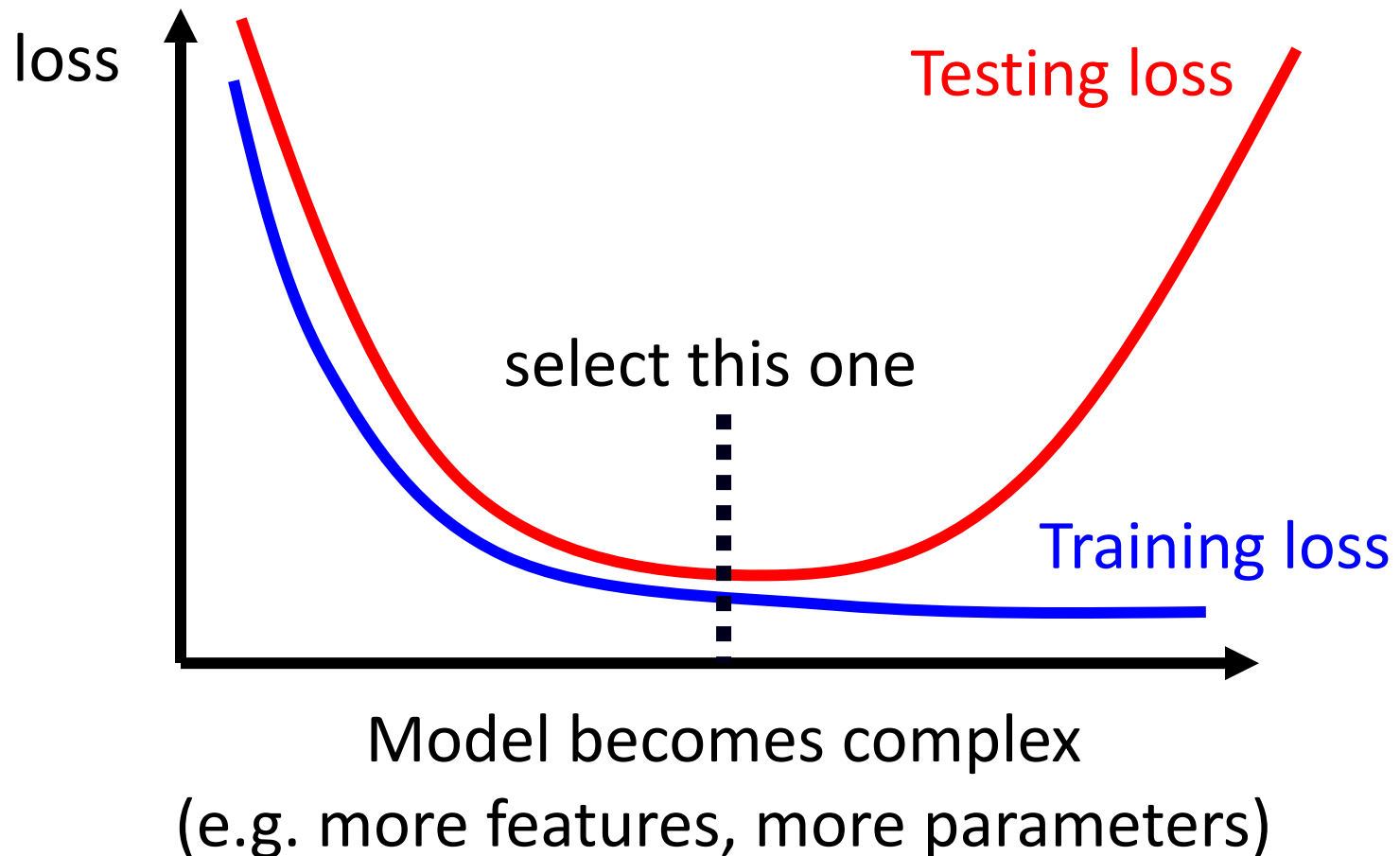


100 examples

- Regularization



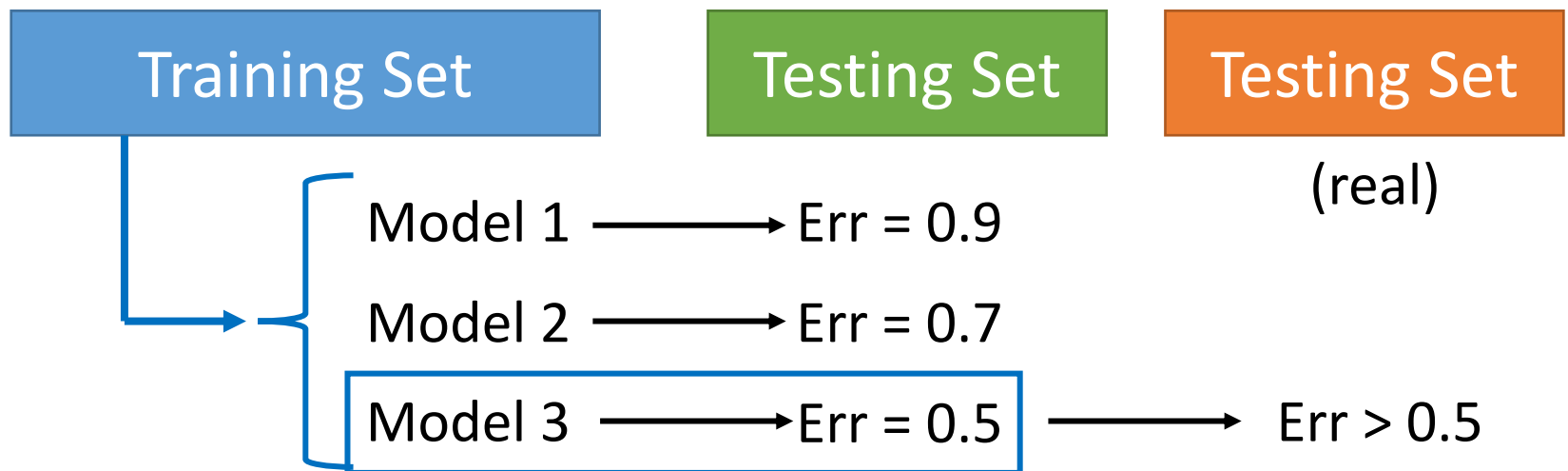
# Bias-Complexity Trade-off





# Model Selection

- There is usually a trade-off between bias and variance.
- Select a model that balances two kinds of error to minimize total error
- What you should NOT do:



# Bias and Variance

- Summary
- Bias describes *the fitting ability* of the learning algorithm.
- Variance captures the impact of *data perturbation*.
- In order to achieve good generalization performance, *both bias and variance need to be small*.

# Bias and Variance

- Summary

What to do when:

Large bias	Large variance
Add more features as input	More data
A more complex model	Regularization

# Summary

- **Terminology**
  - Data (Data set/Feature/Sample...)
  - Task (Train/Test...)
- **Error and overfitting**
  - Accuracy, Error and Generalization
  - How to deal with over fitting?
- **Evaluation Methods**
  - Hold-out/Cross Validation/Bootstrapping
- **Performance Measure**
  - MSE/Accuracy/Precision/Recall/F1
- **Bias and Variance**
  - What to do with large bias/variance?

# Exercise and Thinking

- The data set contains 100 samples, of which half are positive and half negative. Assume that the model generated by the learning algorithm predicts the new sample as the label with a larger number of training samples (random guessing when the number of training samples is the same). Try to evaluate the algorithm with error rate, using 10-fold cross validation.

**50%**

- What if  $N=5$ ?

**50%**

- What if  $N=100$ ?

**100%**