Machine Learning

OSU EECS, Fall 2017

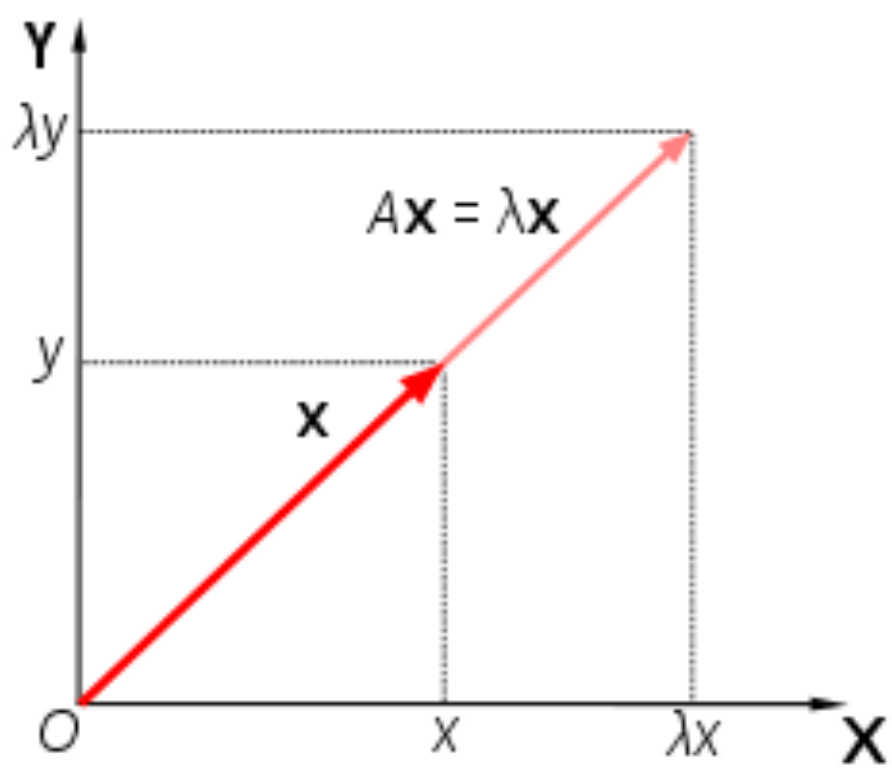
Prof. Liang Huang

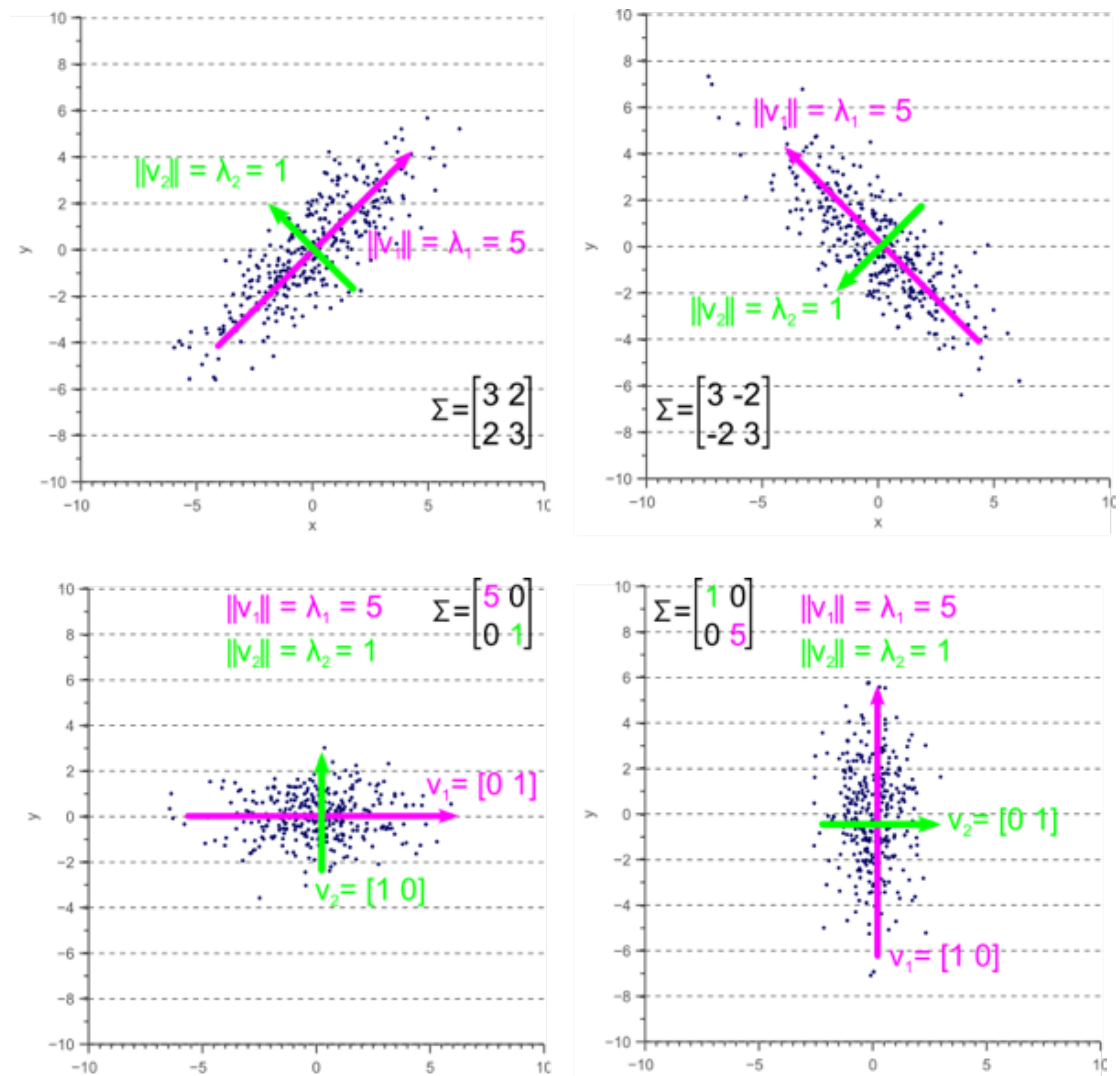
liang.huang@oregonstate.edu

http://classes.engr.oregonstate.edu/eecs/fall2017/cs534/

Warm-Up Questions

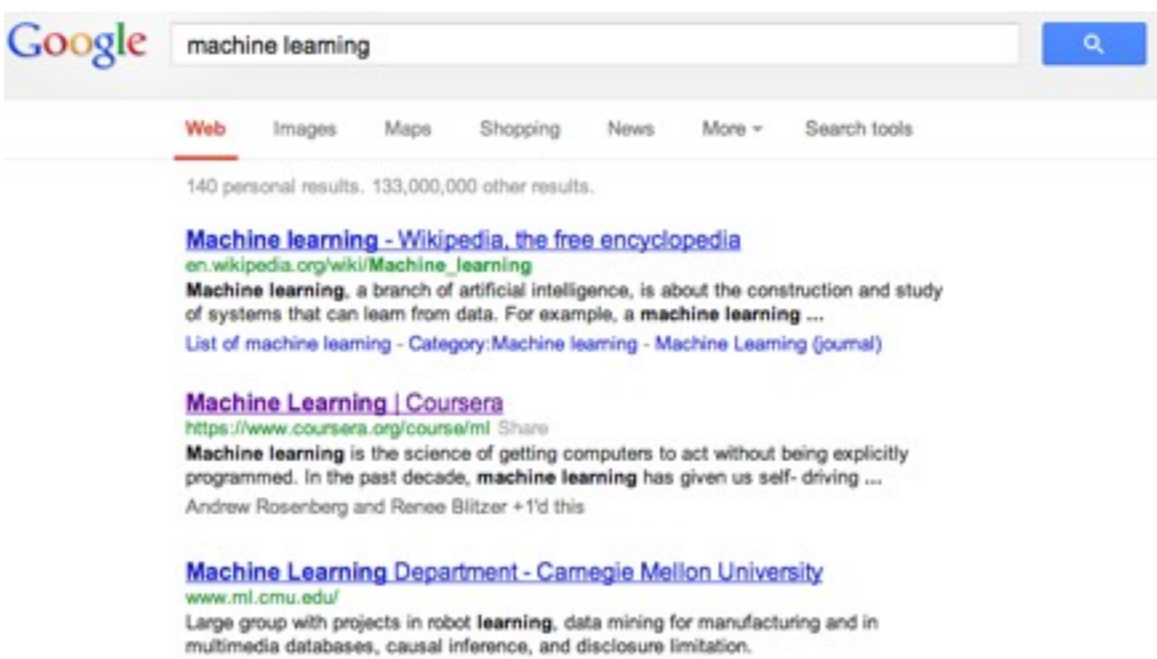
- what are the geometric interpretations of
  - eigenvector
  - covariance matrix
- 0% quiz on Tuesday

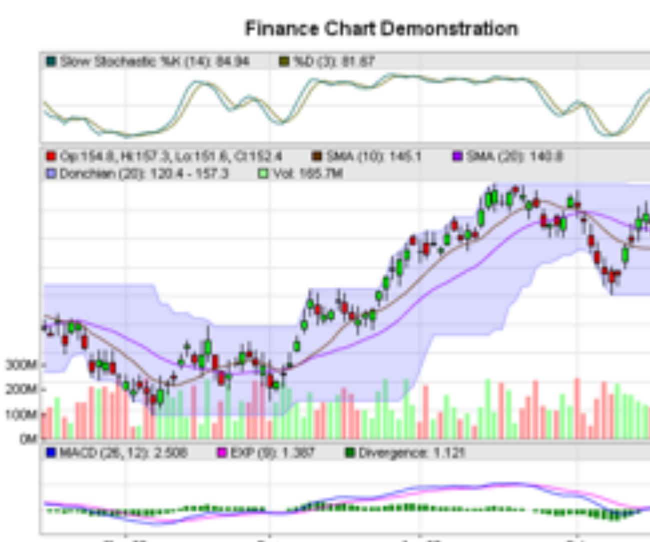


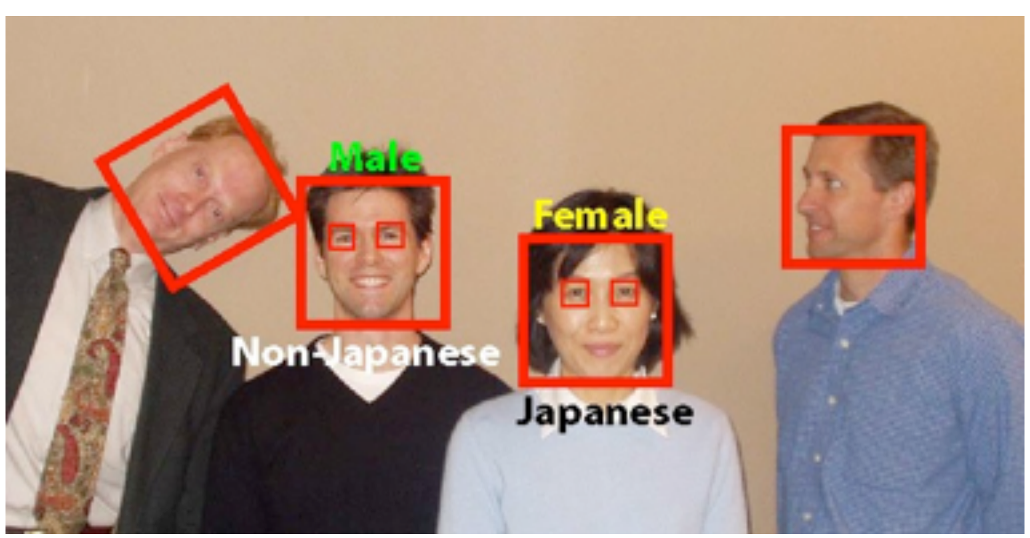


Machine Learning is Everywhere

- “A breakthrough in machine learning would be worth ten Microsofts” (Bill Gates)
- Machine learning is the hot new thing” (John Hennessy, President, Stanford)
- “Web rankings today are mostly a matter of machine learning” (Prabhakar Raghavan, Dir. Research, Yahoo)







The Future of Software Engineering

- "See when AI comes, I'll be long gone (being replaced by autonomous cars) but the programmers in those companies will be too, by automatic program generators." --- an Uber driver to an ML prof

Machine Learning Failures

liang's rule: if you see "X carefully" in China, just don't do it.

Machine Learning Failures

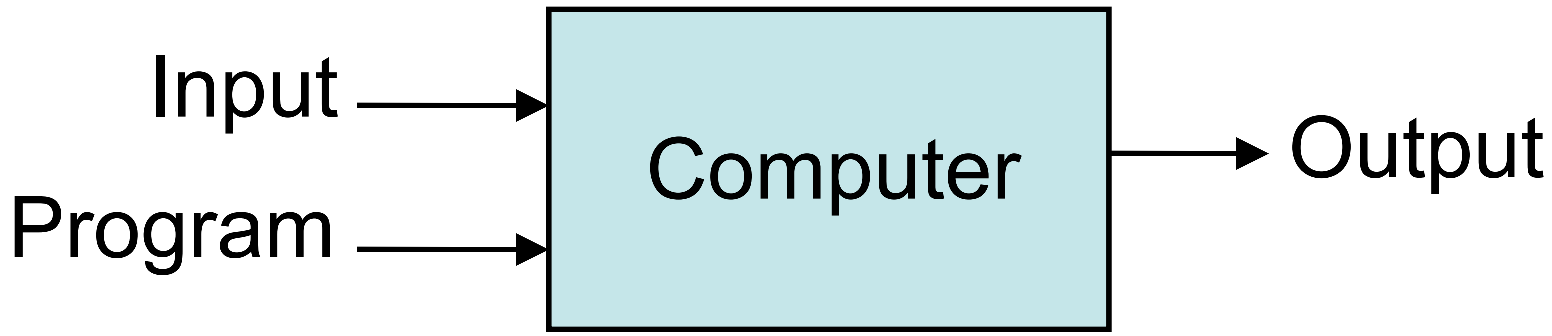
Machine Learning Failures

clear evidence that MT is used in real life.

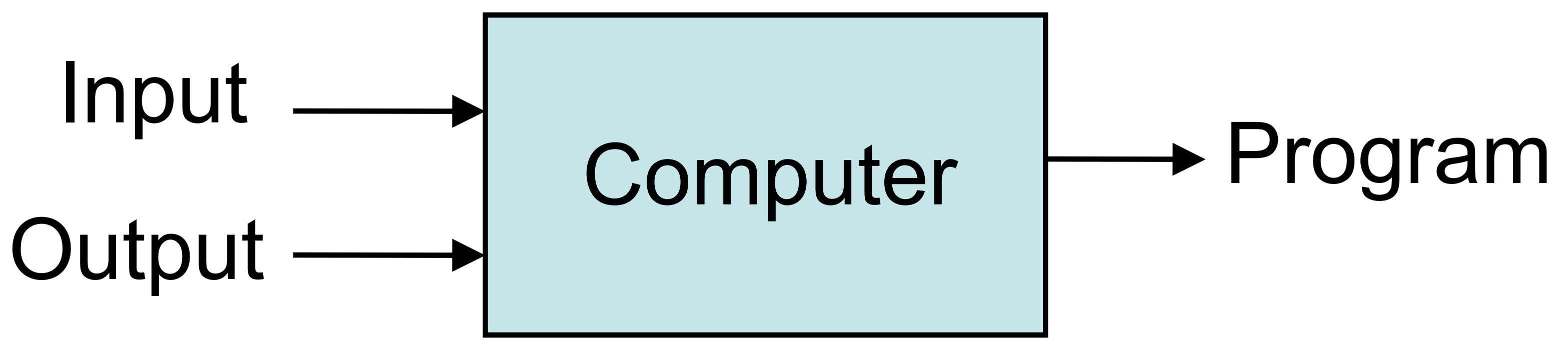
What is Machine Learning

- Machine Learning = Automating Automation
  - Getting computers to program themselves
  - Let the data do the work instead!

Traditional Programming



Machine Learning



Magic?

No, more like gardening

- **Seeds** = Algorithms
- **Nutrients** = Data
- **Gardener** = You
- **Plants** = Programs

ML in a Nutshell

- Tens of thousands of machine learning algorithms
- Hundreds new every year
- Every machine learning algorithm has three components:
  - **Representation**
  - **Evaluation**
  - **Optimization**

Representation

- Separating Hyperplanes
- Support vectors
- Decision trees
- Sets of rules / Logic programs
- Instances (Nearest Neighbor)
- Graphical models (Bayes/Markov nets)
- Neural networks
- Model ensembles
- Etc.

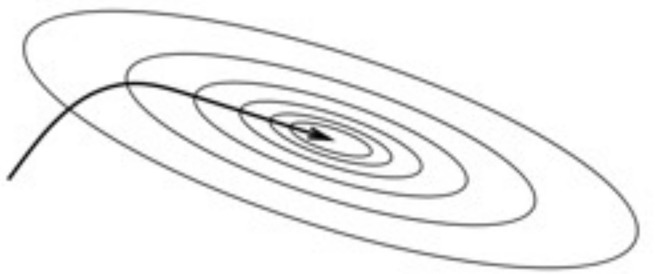
Evaluation

- Accuracy
- Precision and recall
- Squared error
- Likelihood
- Posterior probability
- Cost / Utility
- Margin
- Entropy
- K-L divergence
- Etc.

Optimization

- Combinatorial optimization
  - E.g.: Greedy search, Dynamic programming
- Convex optimization
  - E.g.: Gradient descent, Coordinate descent
- Constrained optimization
  - E.g.: Linear programming, Quadratic programming

Gradient Descent



- if learning rate is too big, it'll diverge
- if learning rate is too small, it'll converge very slowly

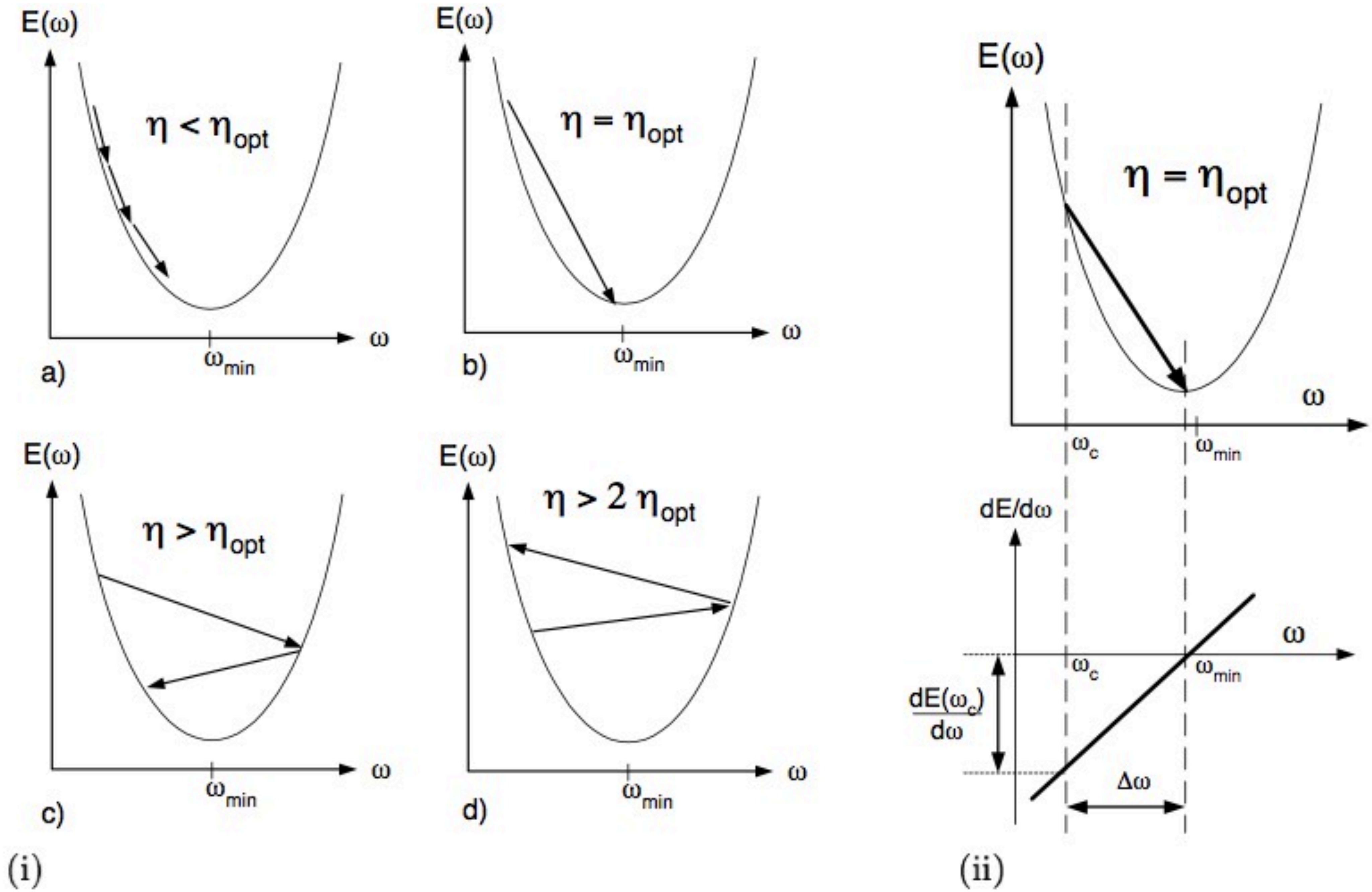


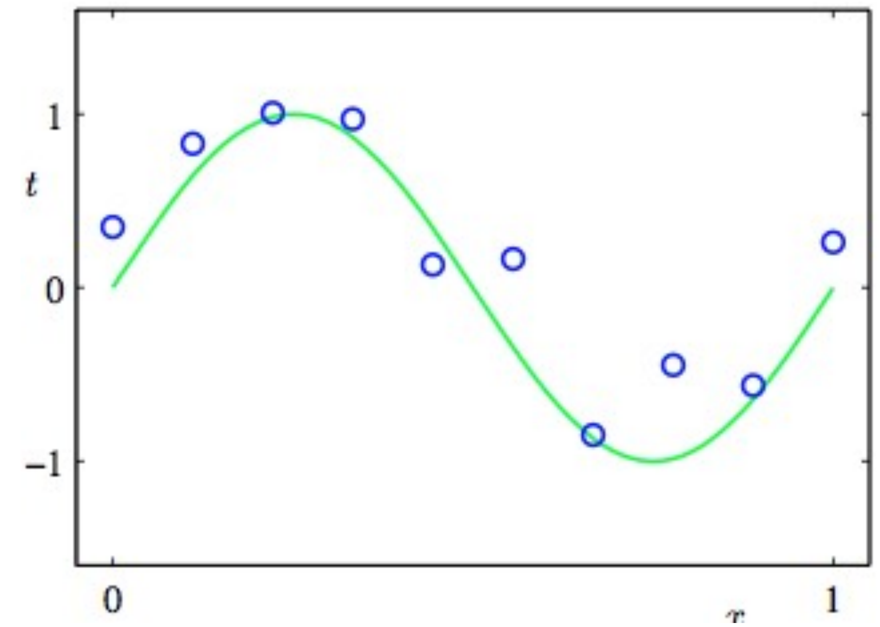
Fig. 6. Gradient descent for different learning rates.

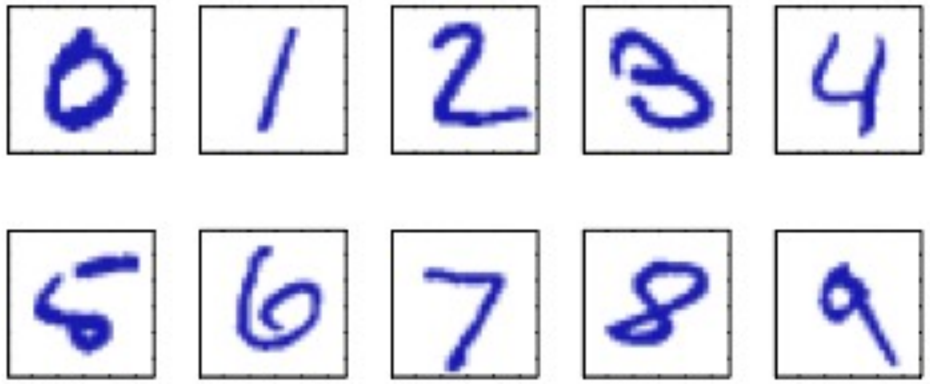
Types of Learning

- **Supervised (inductive) learning**
  - Training data includes desired outputs
- **Unsupervised learning**
  - Training data does not include desired outputs
- **Semi-supervised learning**
  - Training data includes a few desired outputs
- **Reinforcement learning**
  - Rewards from sequence of actions

Supervised Learning

- **Given** examples (X, $f(X)$) for an unknown function f
- **Find** a good approximation of function f
    - Discrete $f(X)$: Classification (binary, multiclass, structured)
    - Continuous $f(X)$: Regression



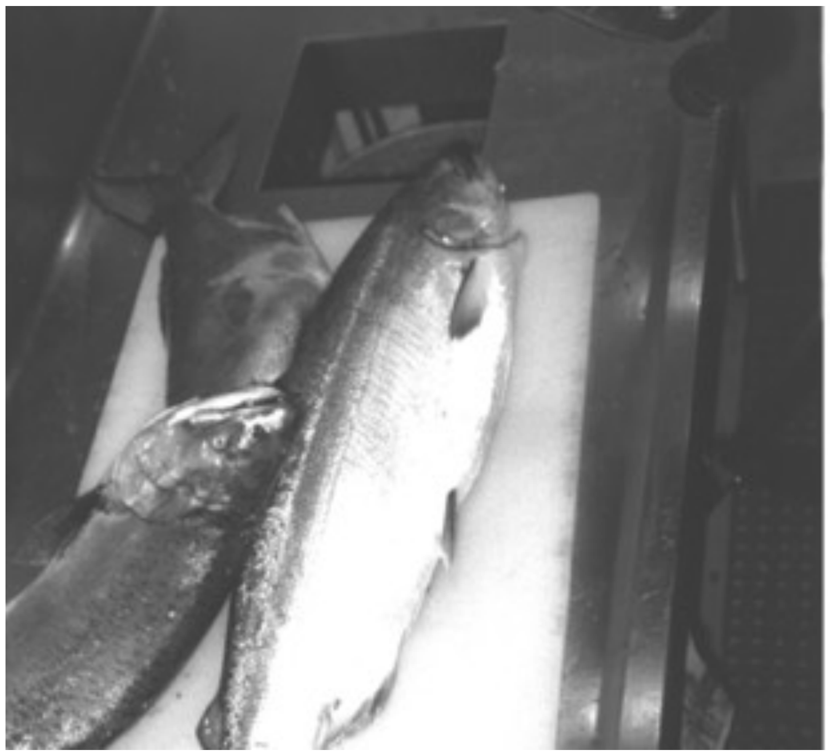


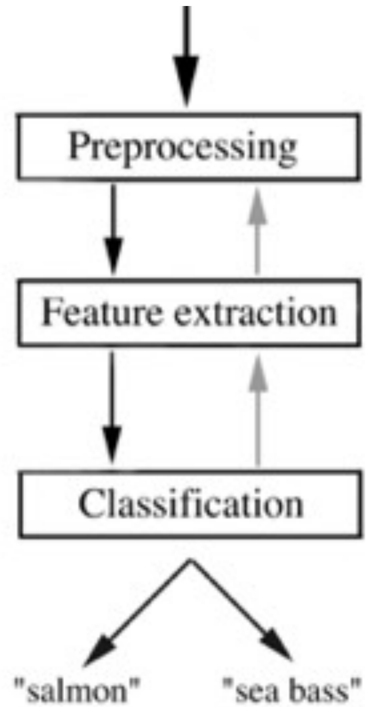
When is Supervised Learning useful

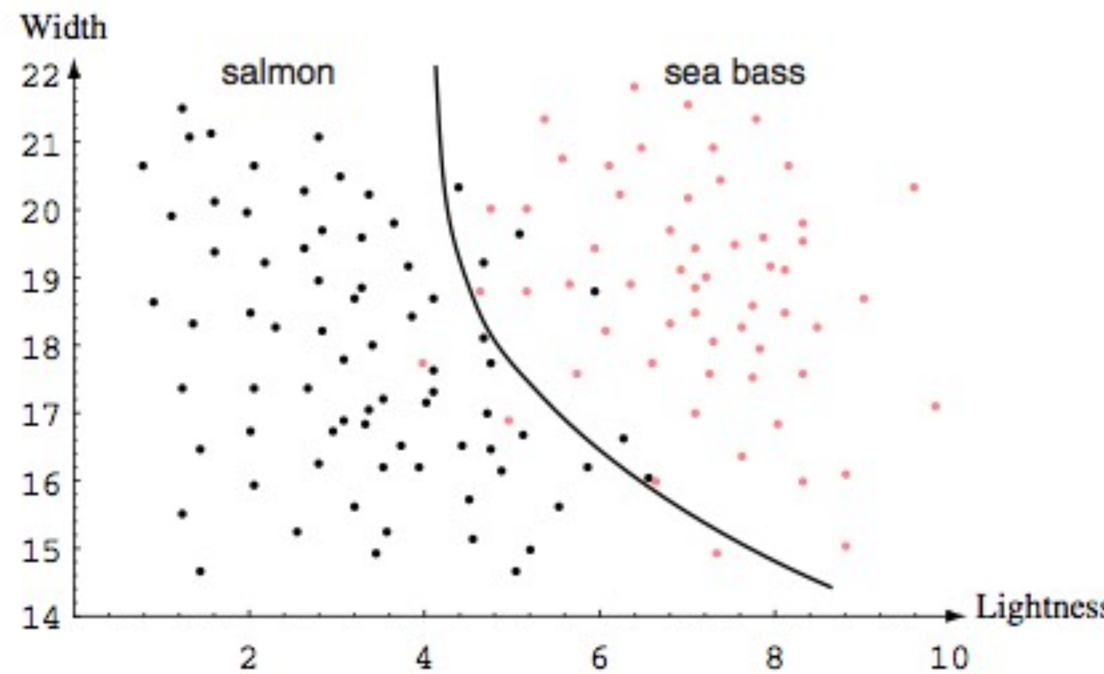
- when there is no human expert
  - input x: bond graph for a new molecule
  - output $f(x)$: predicted binding strength to AIDS protease
- when humans can perform the task but can't describe it
  - computer vision: face recognition, OCR
- where the desired function changes frequently
  - stock price prediction, spam filtering
- where each user needs a customized function
  - speech recognition, spam filtering

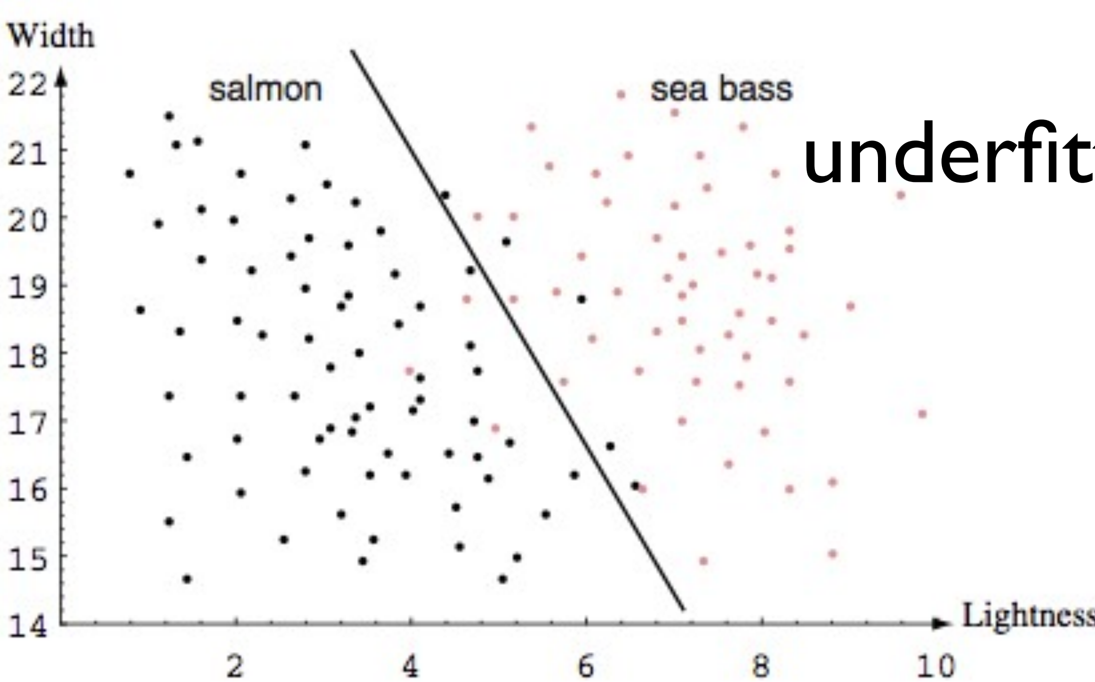
Classification

- input *X*: feature representation (“observation”)

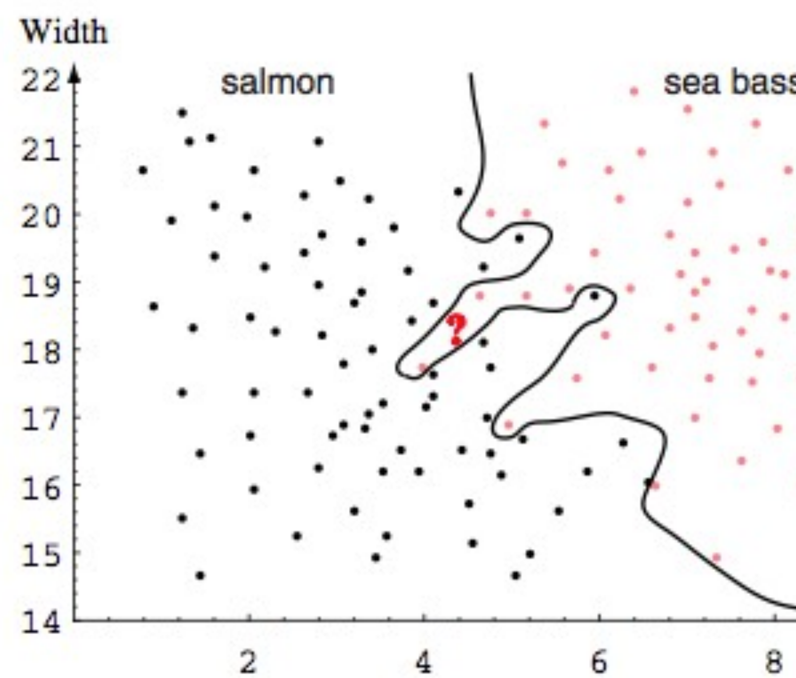








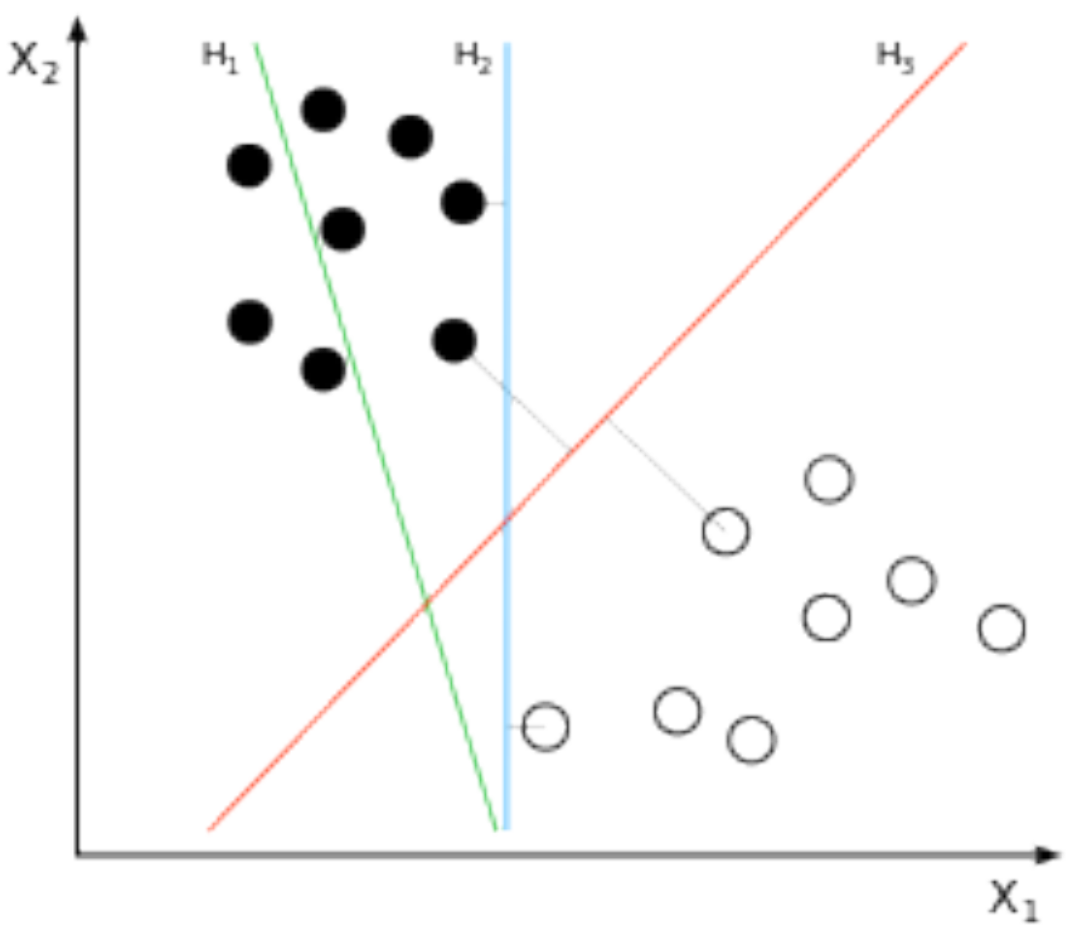
underfitting

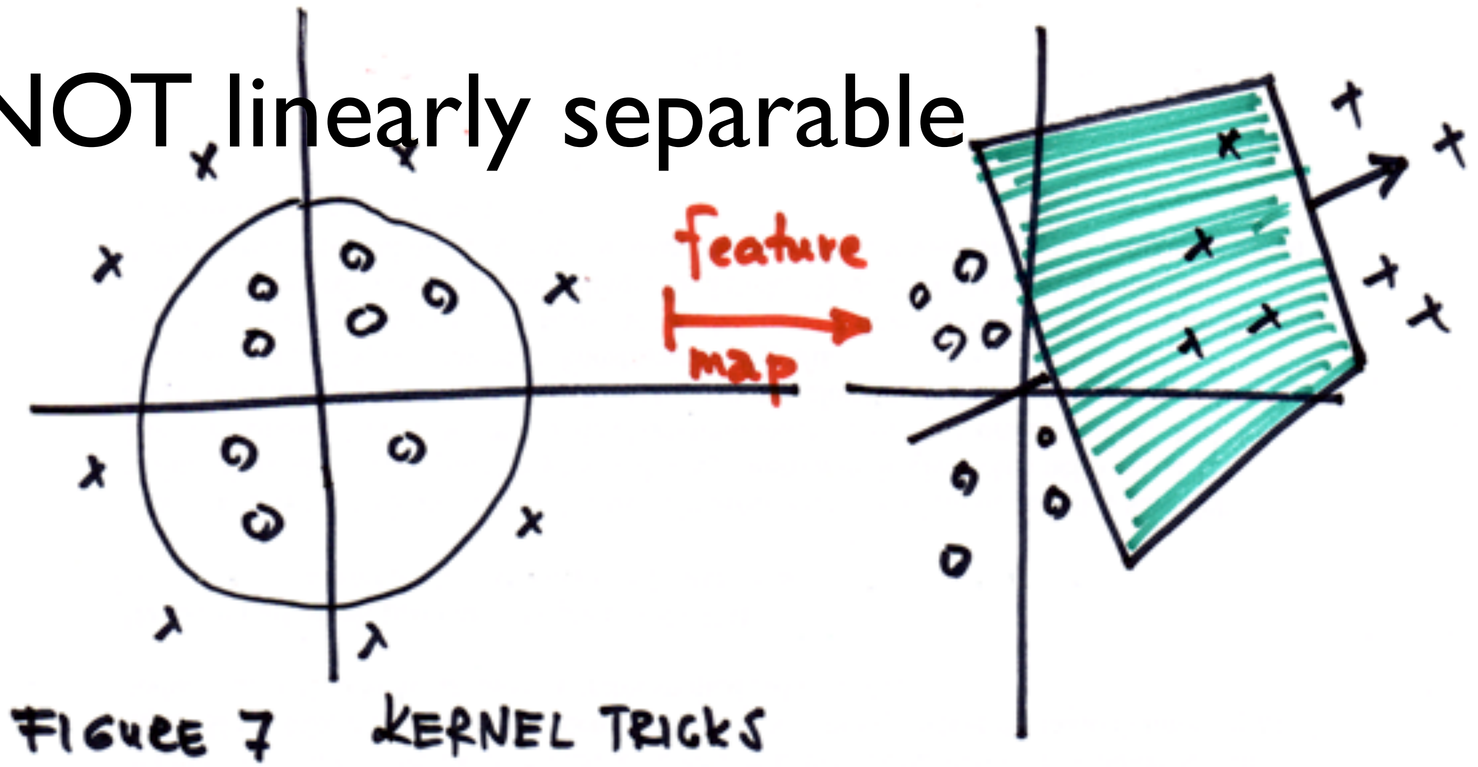


overfitting

Linear Classification

- Q1: how to learn a separating hyperplane
- Q2: how to learn the optimal separating hyperplane
- Q3: what if the data is NOT linearly separable

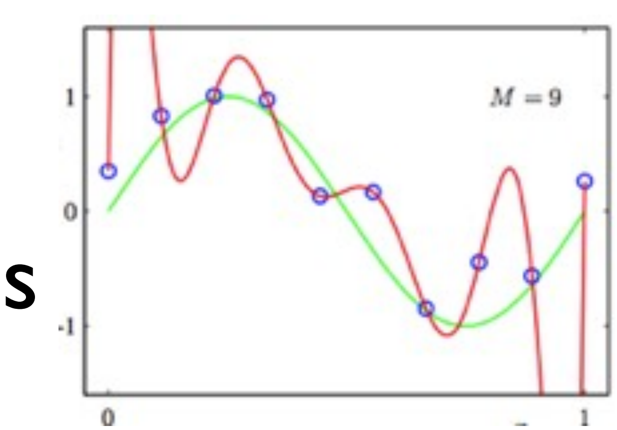




low-dimension

high-dimension

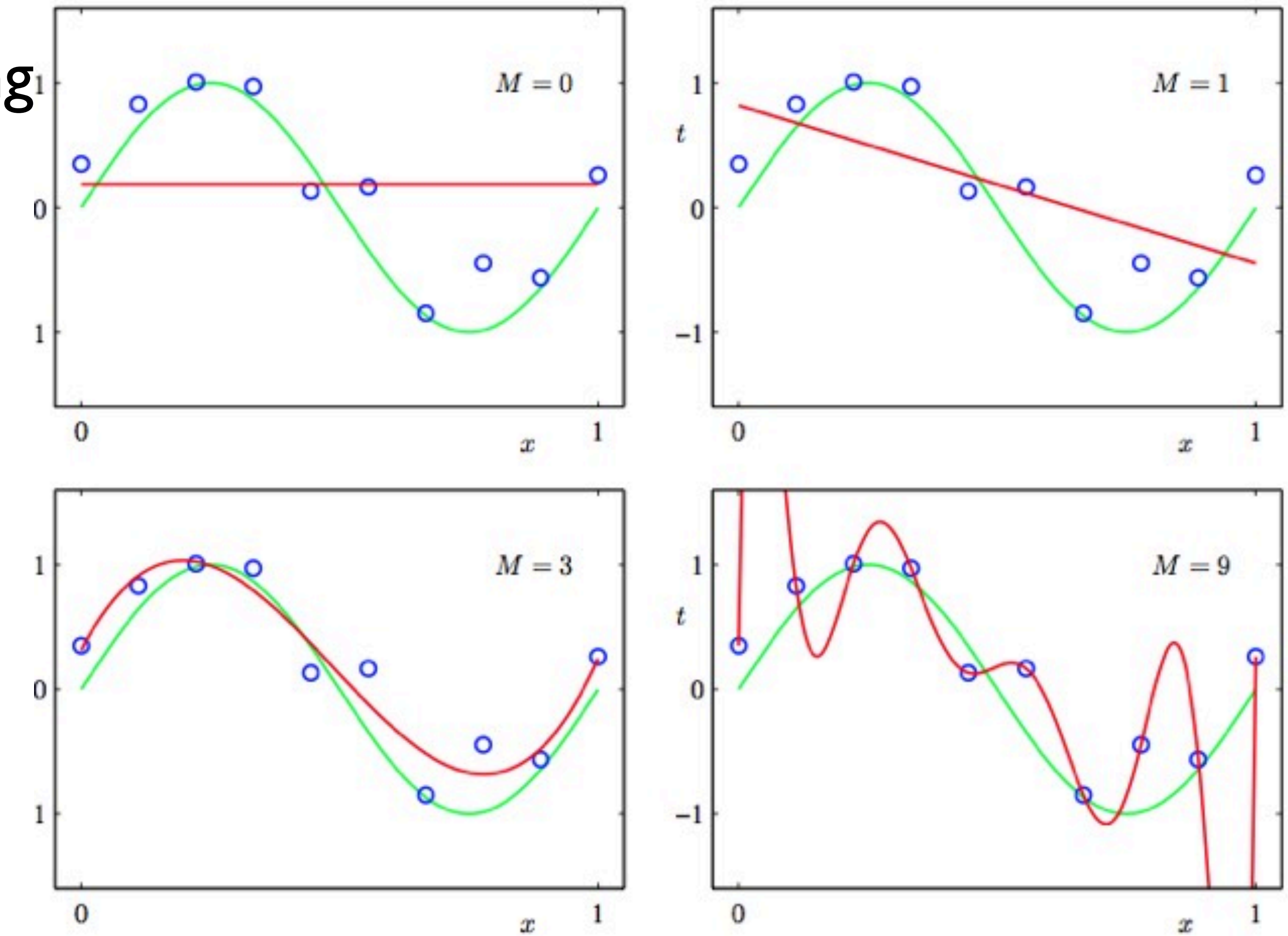
potential overfitting
in very high dimensions



Regression

- linear and non-linear regression
- overfitting and underfitting (same as in classification)
  - how to choose the optimal model complexity?

underfitting

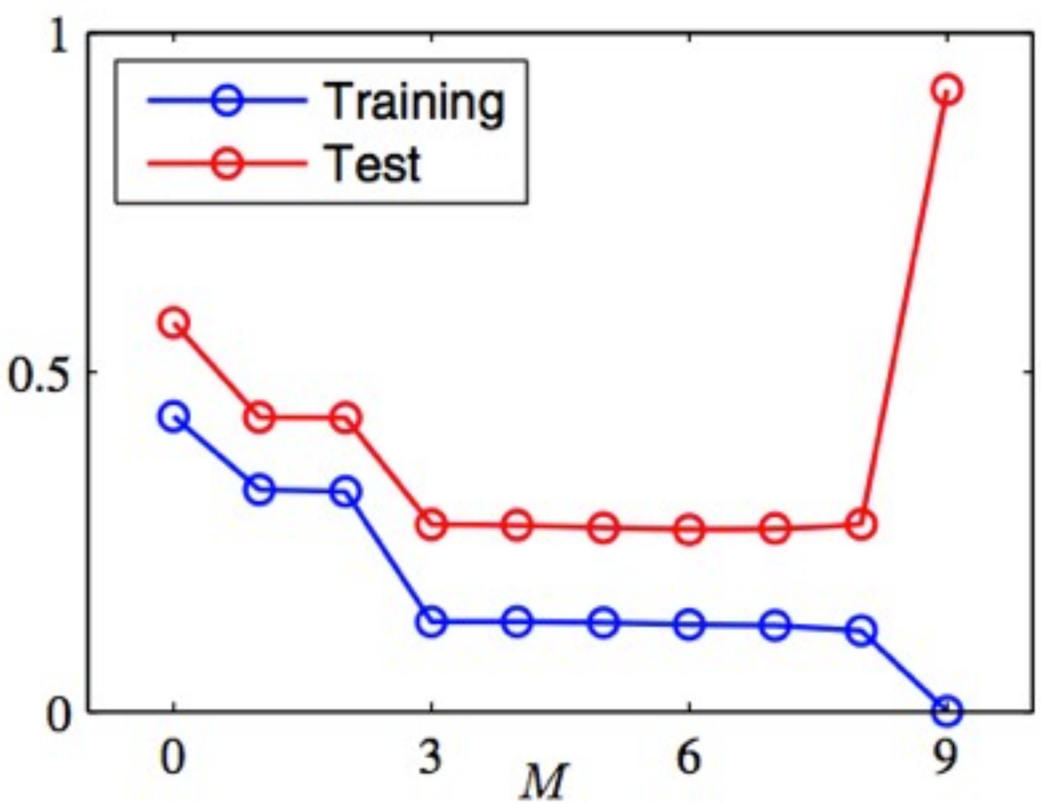


underfitting

overfitting

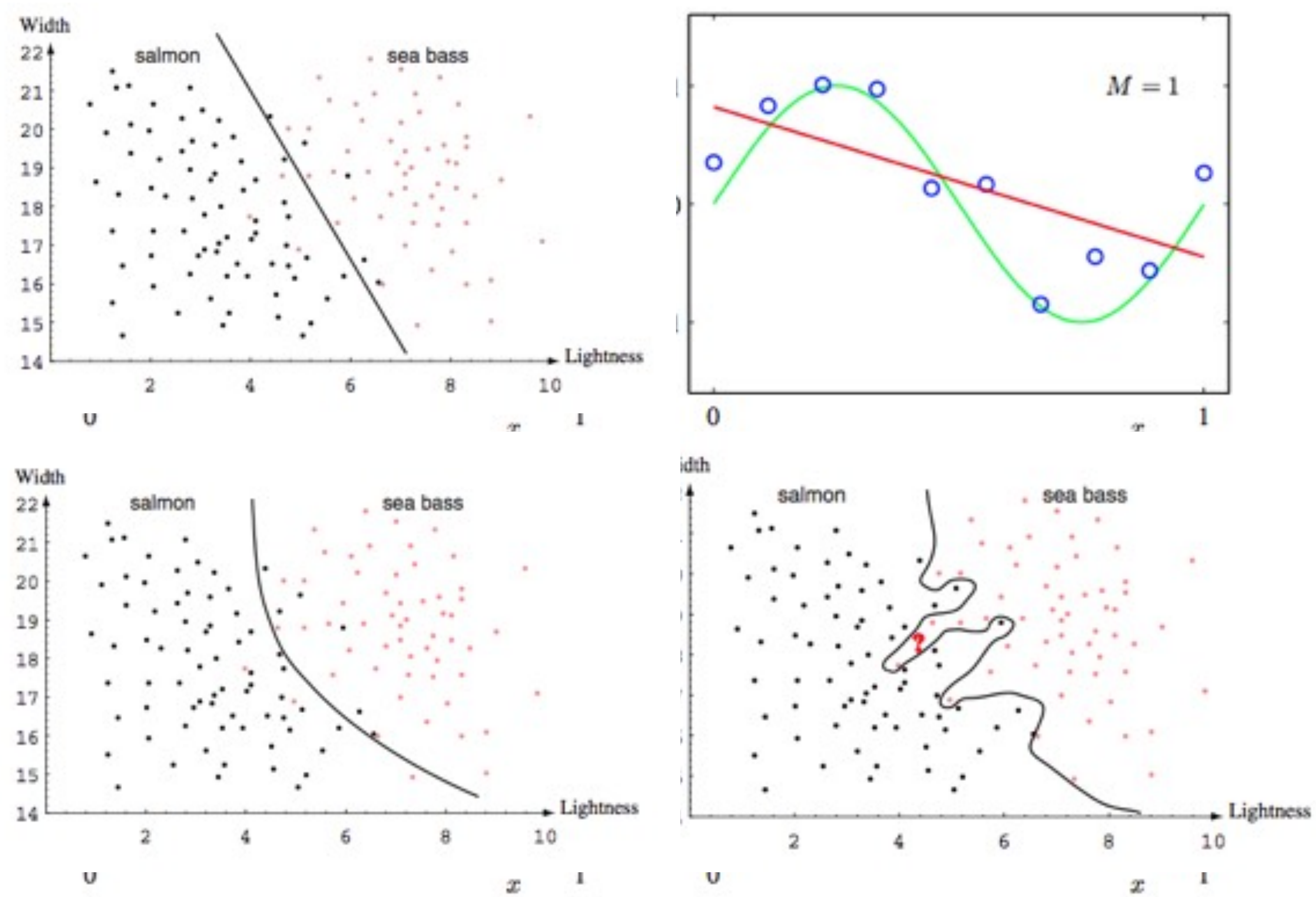
Training, Test, & Generalization Error

- but you don't know test data a priori
  - generalization error: prob. of error on possible test data
- use held-out training data to "simulate" test-data



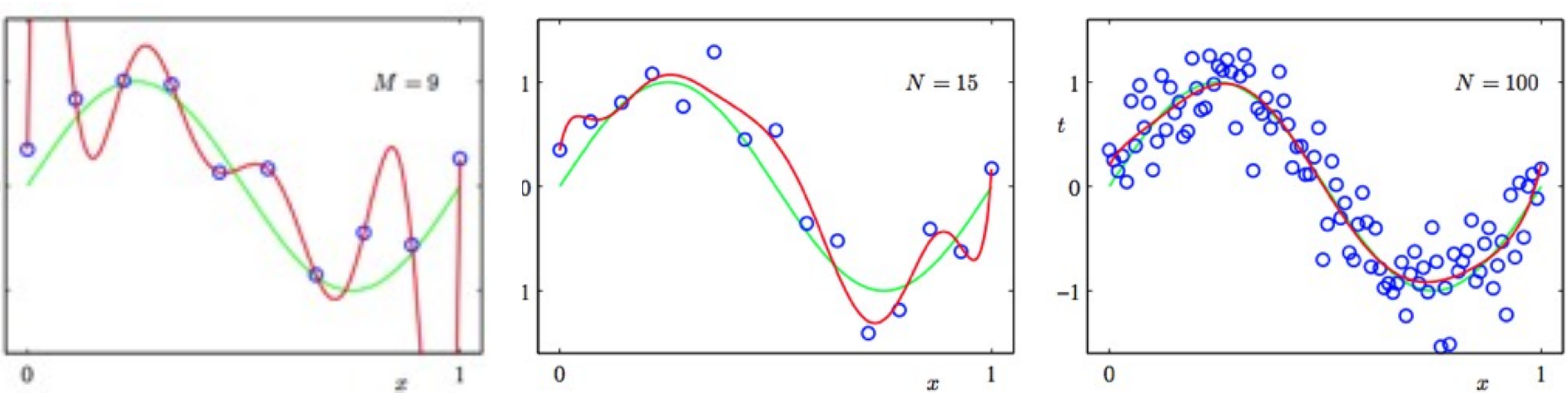
underfitting

overfittin



Ways to Prevent Overfitting

- held-out data to simulate generalization error
- more data points (overfitting is more likely on small data)
    - assuming same model complexity
- regularization (explicit control of model complexity)



polynomials of degree 9

What We'll Cover

- **Supervised learning**
  - Linear Classification (Perceptron)
  - Linear Regression
  - Logistic Regression
  - Support Vector Machines
  - Instance-based learning (e.g. Nearest Neighbors)
  - Structured Prediction
- **Unsupervised learning**
  - Clustering (k-means, EM)
  - Dimensionality reduction (PCA etc.)