Introduction of Machine Learning

Chul Min Yeum
Assistant Professor
Civil and Environmental Engineering
University of Waterloo, Canada

CIVE 497 – CIVE 700: Smart Structure Technology

UNIVERSITY OF WATERLOO
FACULTY OF ENGINEERING

Last updated: 2019-03-25

What is Machine Learning?

Learning is any process by which a system **improves performance** from **experience** (Herbert Simon)

model, coefficient

Definition by Tom Mitchell (1998):

Machine Learning is the study of algorithm that

- improve their performance ***P***
- at come task ***T***
- with experience ***E***

A well-defined learning task is given by <***P, T, E***>

Slide credit: Eric Eaton

Defining the Learning Task

Improve on task T, with respect to performance metric P, based on experience E

T: Playing checkers
P: Percentage of games won against an arbitrary opponent
E: Playing practice games against itself

T: Recognizing hand-written words
P: Percentage of words correctly classified
E: Database of human-labeled images of handwritten words

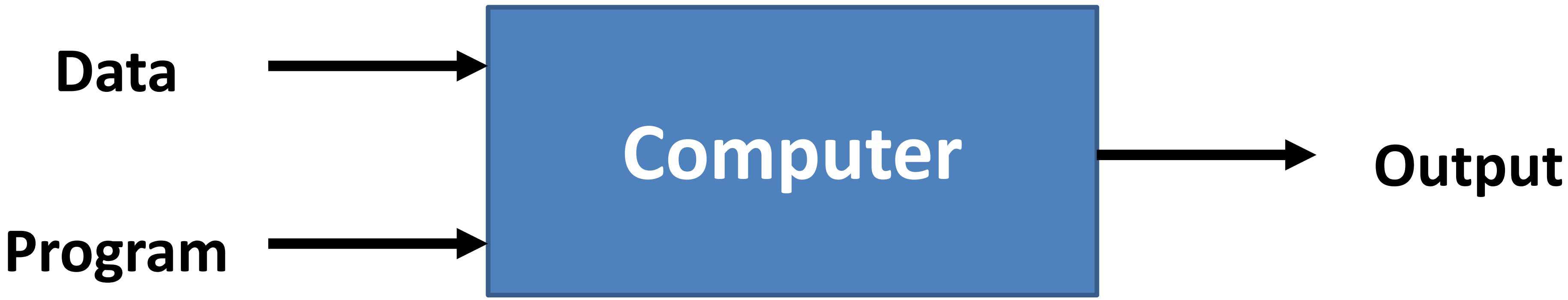
T: Driving on four-lane highways using vision sensors
P: Average distance traveled before a human-judged error
E: A sequence of images and steering commands recorded while observing a human driver.

T: Categorize email messages as spam or legitimate.
P: Percentage of email messages correctly classified.
E: Database of emails, some with human-given labels

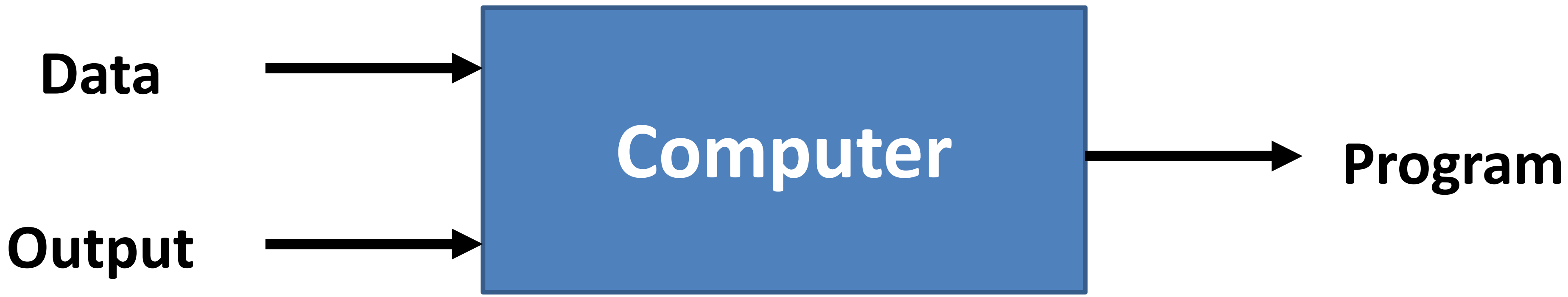
Slide credit: Ray Mooney

Why is Machine Learning Different from Traditional Programming?

Traditional Programming



Machine Learning



Slide credit: Pedro Domingos

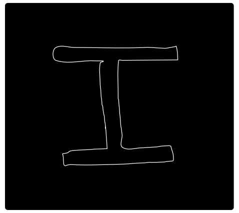
Example: H Beam Classification

Traditional Programming

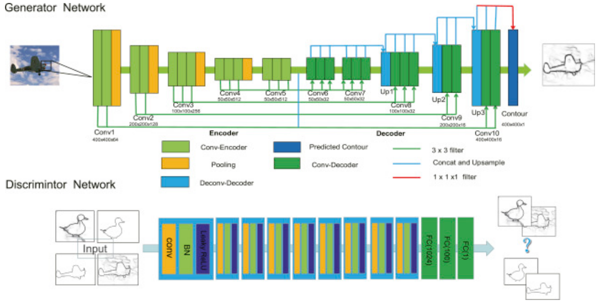
Data → Computer
Program → Computer
Computer → Output

Edge detection, Hough transform

Machine Learning



Data → Computer
Output → Computer
Computer → Program



Example: Image-to-Image Translation with Conditional Adversarial Nets

https://phillipi.github.io/pix2pix/

ML is used when:

- Human expertise does not exist (navigating on Mars)
- Humans can’t explain their expertise (speech recognition, image recognition)
- Models are based on huge amounts of data (genomics), which have complex patterns.

Slide credit: E. Alpaydin

Sample Applications

- Web search
- Computational biology
- Finance
- E-commerce
- Space exploration
- Robotics
- Information extraction
- Social network
- Debugging Software
- Inspection

What are your applications?

Slide credit: Pedro Domingos

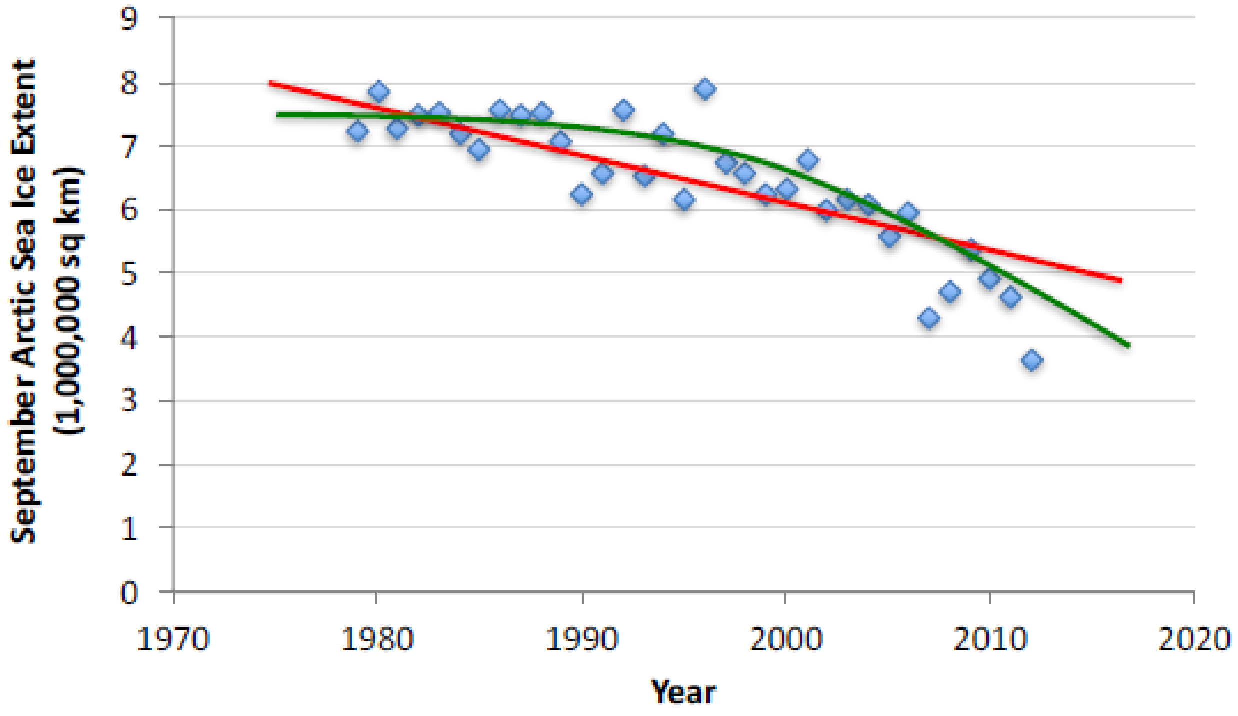
Types of Learning

- **Supervised (inductive) learning**
  - Given: training data + desired outputs (labels)
- **Unsupervised learning**
  - Given: training data (without desired outputs)
- **Semi-supervised learning**
  - Given: training data + a few desired outputs
- **Reinforcement learning**
  - Rewards from sequence of actions

Based on slide by Pedro Domingos

Supervised Learning: Regression

- Given $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$
- $\boxed{\text{Learn a function } f(x)}$ to predict y given x
  - y is real-valued == regression



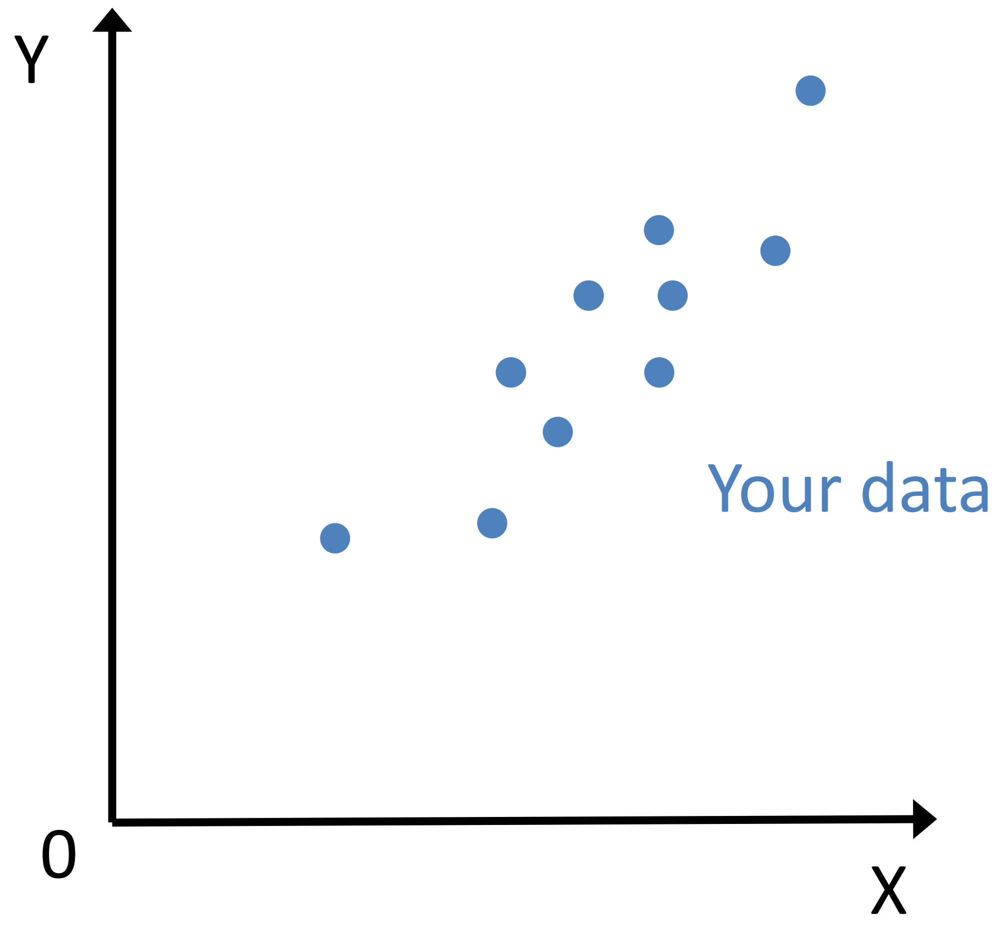
Data from G. Witt. Journal of Statistics Education, Volume 21, Number 1 (2013)

Revisit: Line Fitting

Data (measurement): $(x_1, y_1), \ldots, (x_n, y_n)$

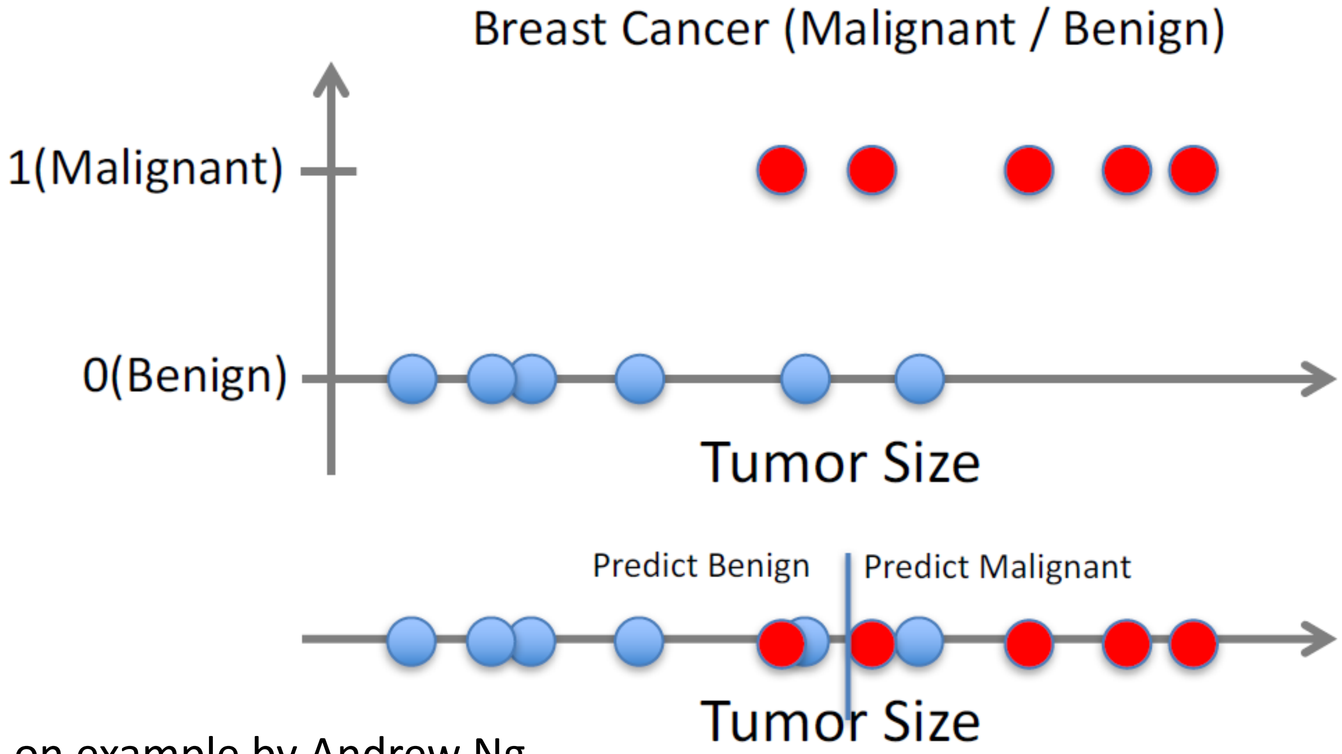
Known model: Line ($y_i = m\,x_i + b$)

We will find m and b.



Supervised Learning: Classification

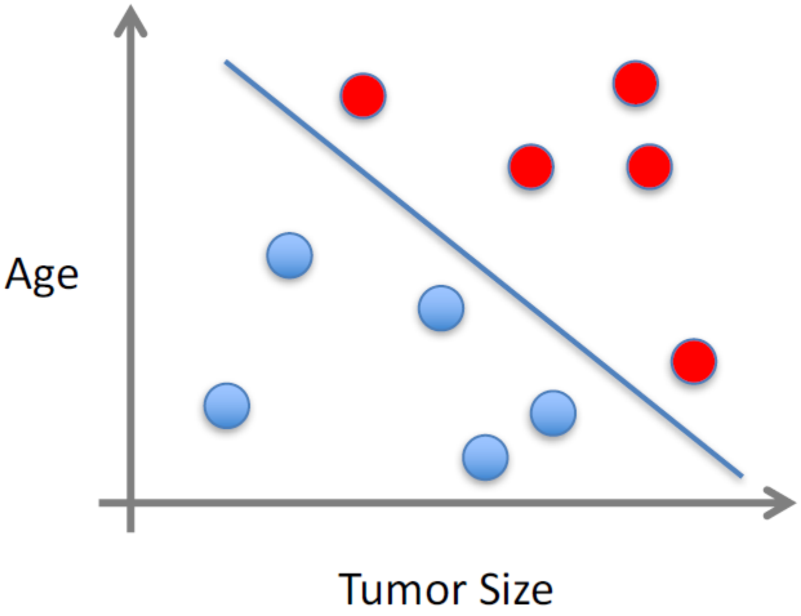
- Given $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$
- Learn a function $f(x)$ to predict y given x
  - y is categorical == classification



Based on example by Andrew Ng

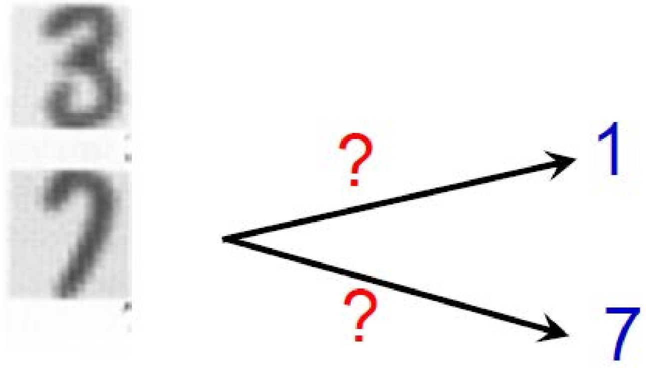
Supervised Learning: Classification

- x can be multi-dimensional
  - Each dimension corresponds to an attribute



- Clump Thickness
- Uniformity of Cell Size
- Uniformity of Cell Shape

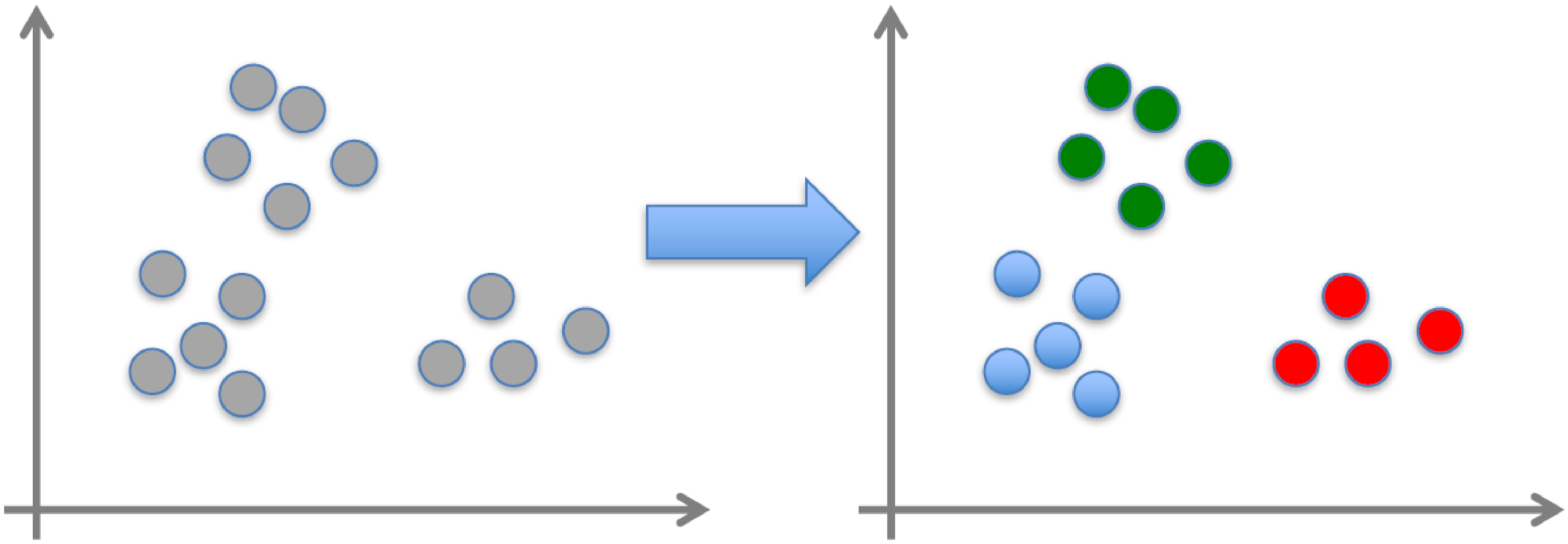
…



Based on example by Andrew Ng

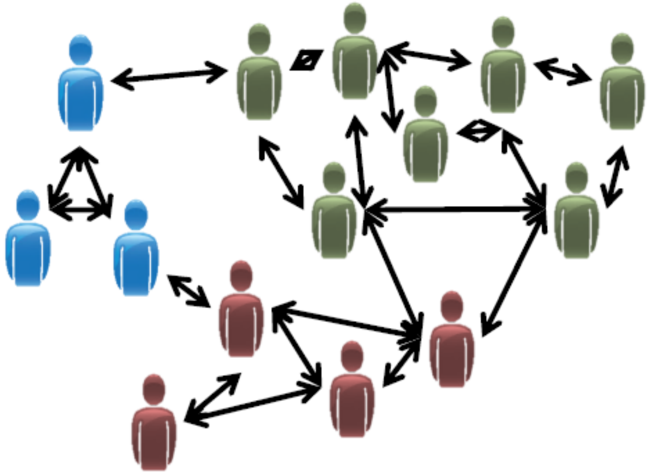
Unsupervised Learning: Clustering

- Given x_1, x_2, …, x_n (without labels)
- Output hidden structure behind the x's
  - E.g., clustering

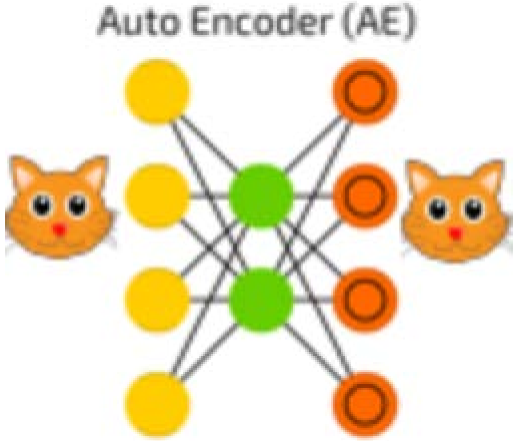


Example: Unsupervised Learning

Market segmentation

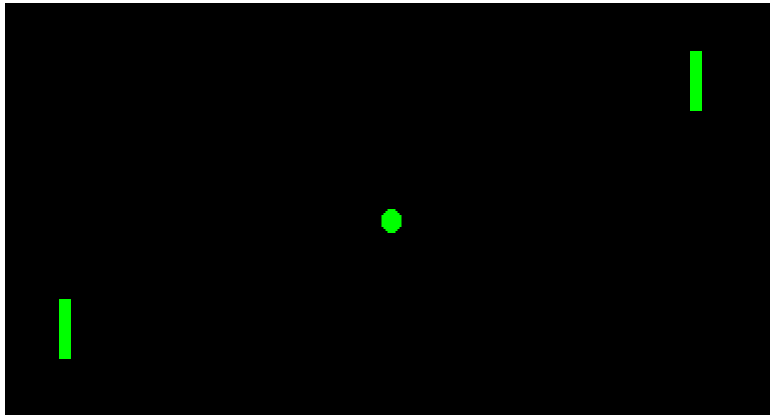


Social network analysis

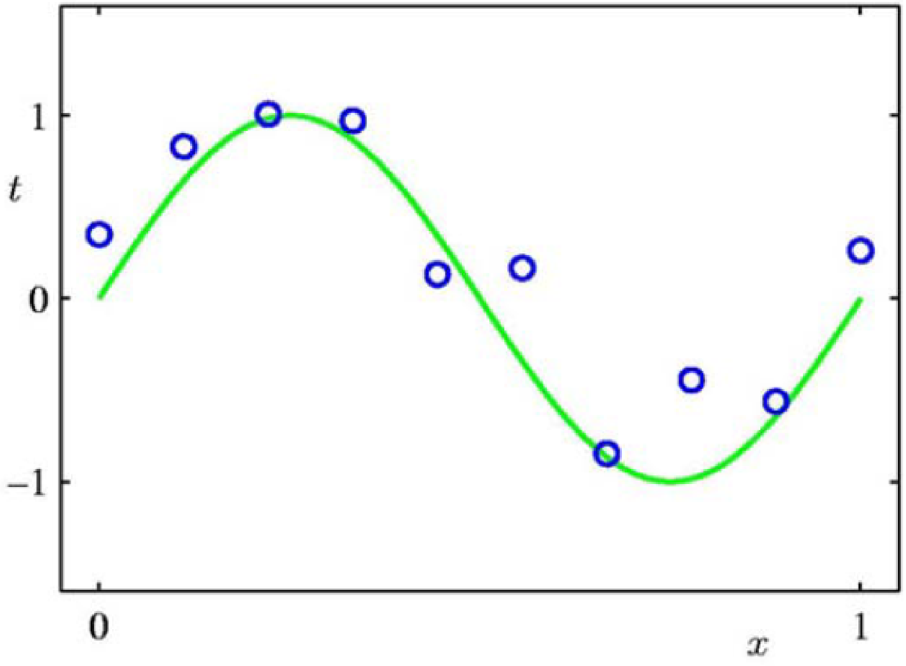


Autoencoder

- Given a sequence of states and actions with (delayed) rewards, output a policy
  - Policy is a mapping from states → actions that tells you what to do in a given state
- Examples:
  - Credit assignment problem
  - Game playing
  - Robot in a maze
  - Balance a pole on your hand



Example: Regression (Supervised Learning)



- Suppose we are given a training set of N observations

$$(x_1, \ldots, x_N) \text{ and } (y_1, \ldots, y_N), x_i, y_i \in \mathbb{R}$$

- Regression problem is to estimate y(x) from this data

Polynomial Curve Fitting

- The green curve is the true function (which is not a polynomial)

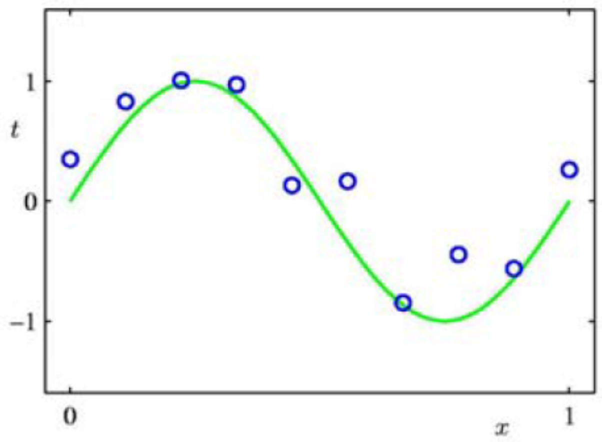
- The data points are uniform in x but have noise in y.

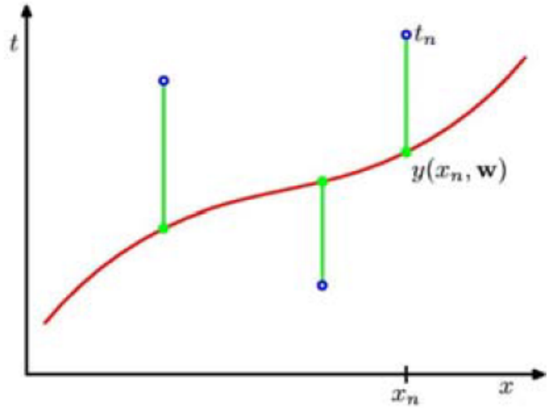
- We will use a loss function that measures the squared error in the prediction of y(x) from x. The loss for the red polynomial is the sum of the squared vertical errors.

$$E(\mathbf{w}) = \frac{1}{2}\sum_{i=1}^{N}\{y(x_i,\mathbf{w}) - t_i\}^2$$

target value

from Bishop



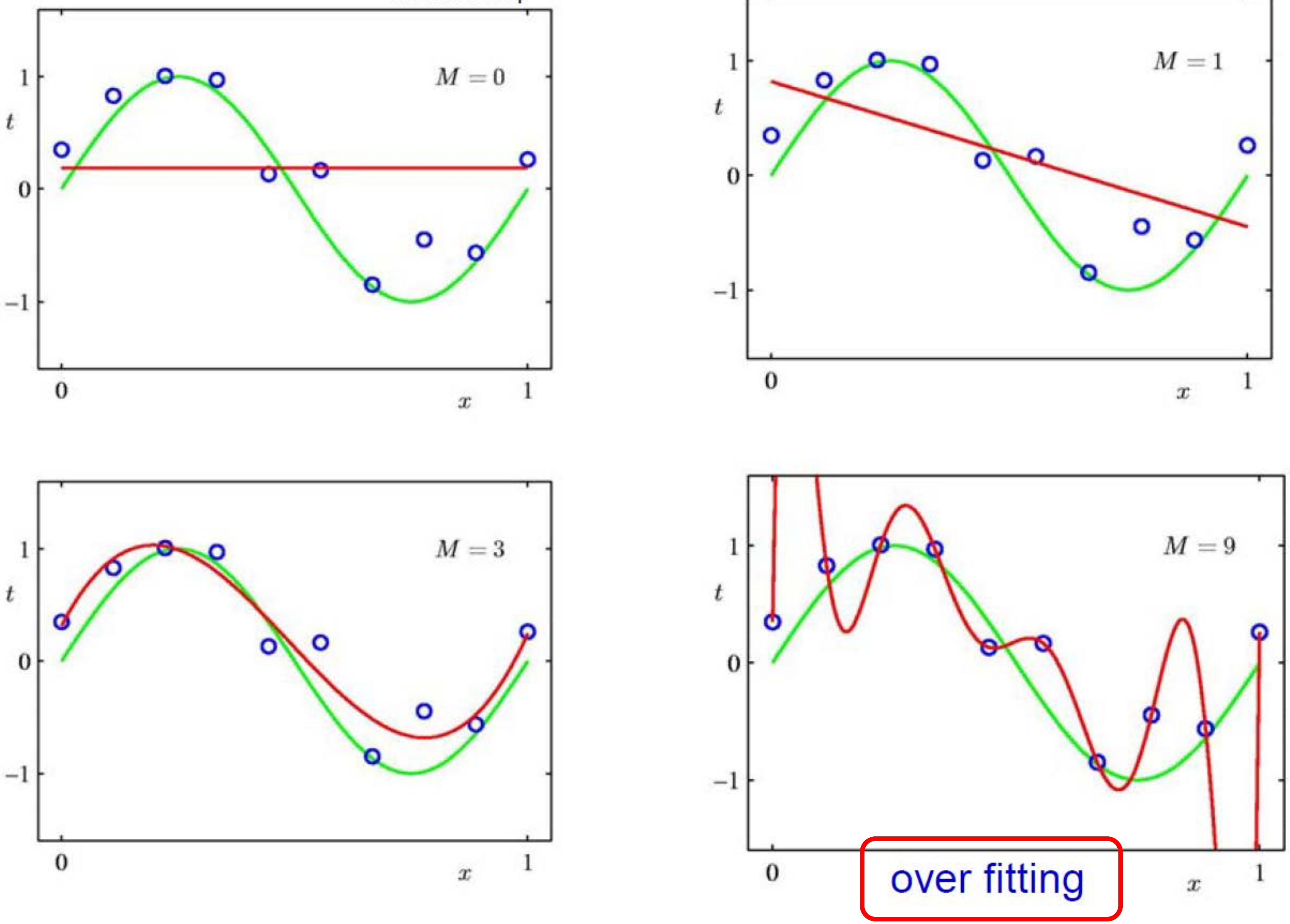


$$y(x,\mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M = \sum_{j=0}^{M} w_j x^j$$

polynomial regression

Some Fits to the Data: Which is Best?

from Bishop

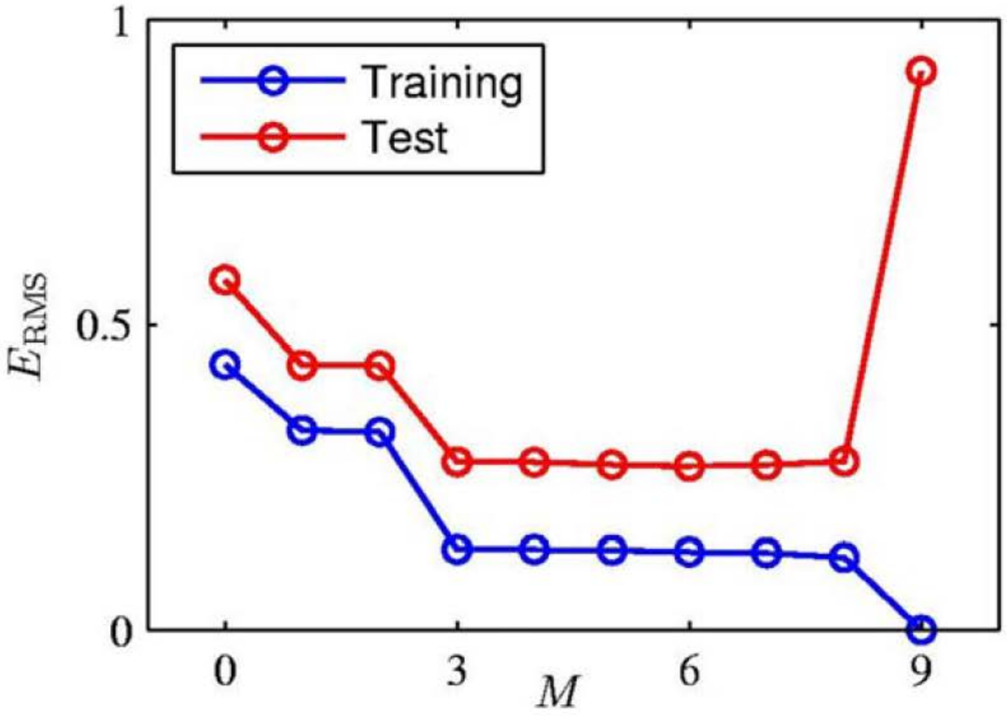


Polynomial Coefficients

	$M = 0$	$M = 1$	$M = 3$	$M = 9$
$w_0^\star$	0.19	0.82	0.31	0.35
$w_1^\star$		-1.27	7.99	232.37
$w_2^\star$			-25.43	-5321.83
$w_3^\star$			17.37	48568.31
$w_4^\star$				-231639.30
$w_5^\star$				640042.26
$w_6^\star$				-1061800.52
$w_7^\star$				1042400.18
$w_8^\star$				-557682.99
$w_9^\star$				125201.43

Overfitting

- test data: a different sample from the same true function



Root-Mean-Square (RMS) Error: $E_{\mathrm{RMS}} = \sqrt{2E(\mathbf{w}^\star)/N}$

- training error goes to zero, but test error increases with M

Trading off Goodness of Fit against Model Complexity

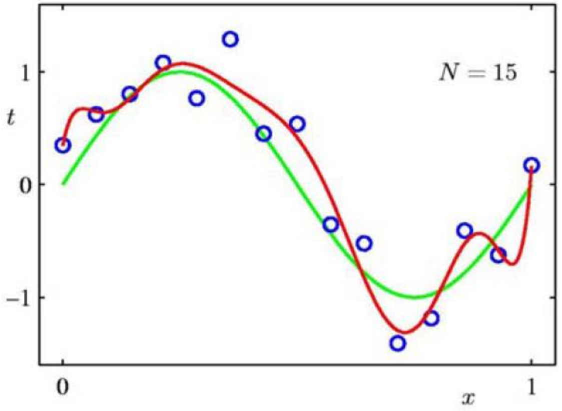
- If the model has as many degrees of freedom as the data, it can fit the training data perfectly

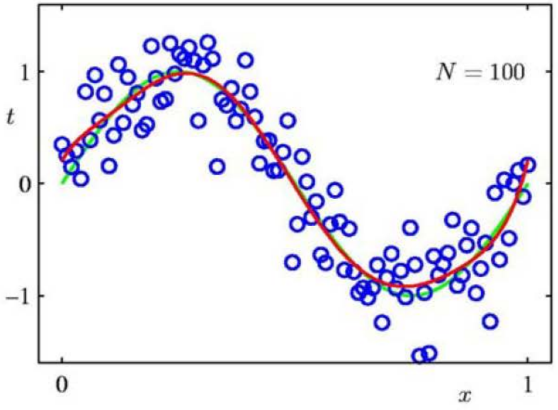
- But the objective in ML is generalization

- Can expect a model to generalize well if it explains the training data surprisingly well given the complexity of the model.

How to Prevent Over-fitting?

- Add more data than the model "complexity"
- For 9th order polynomial:

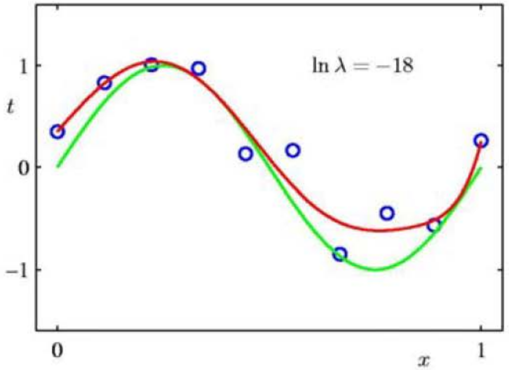


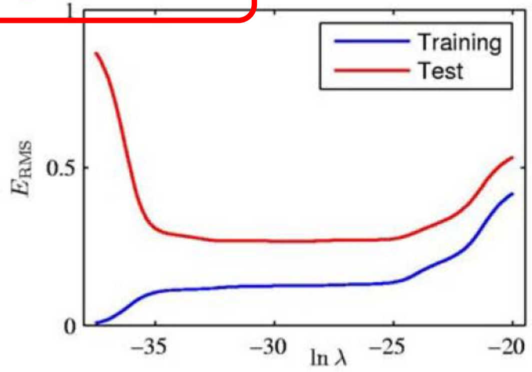


- Regularization: penalize large coefficient values

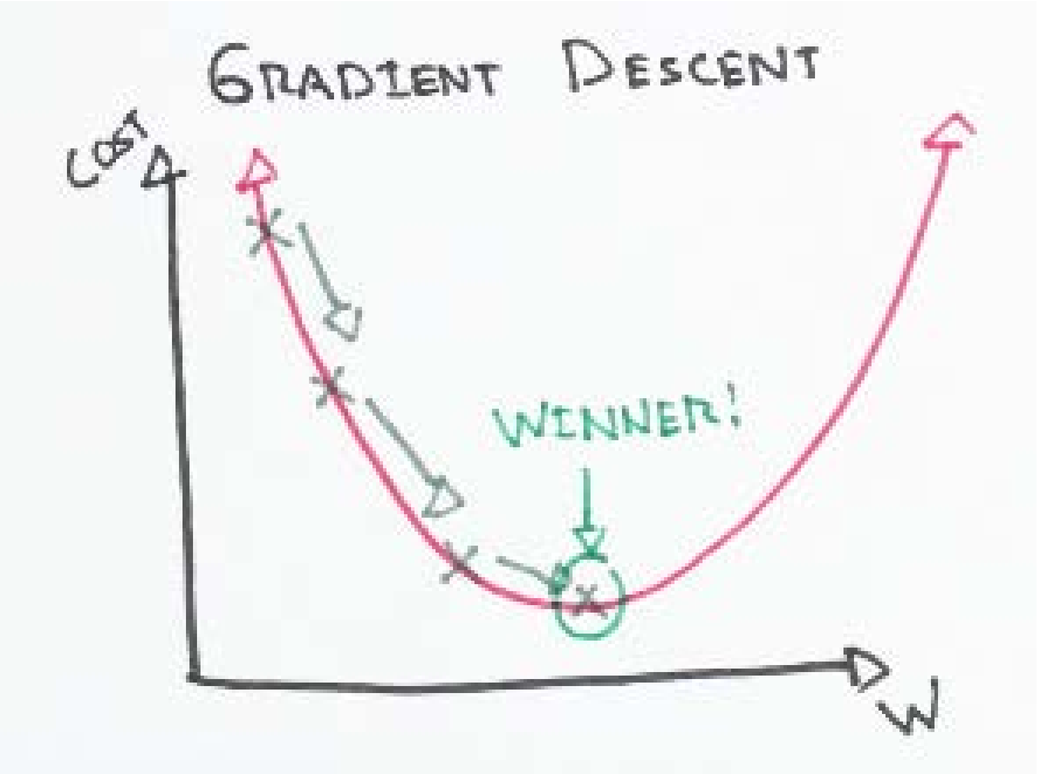
$$\widetilde{E}(\mathbf{w}) = \underbrace{\frac{1}{2}\sum_{i=1}^{N}\{y(x_i,\mathbf{w}) - t_i\}^2}_{\text{loss function}} + \underbrace{\frac{\lambda}{2}\|\mathbf{w}\|^2}_{\boxed{\text{regularization}}}$$

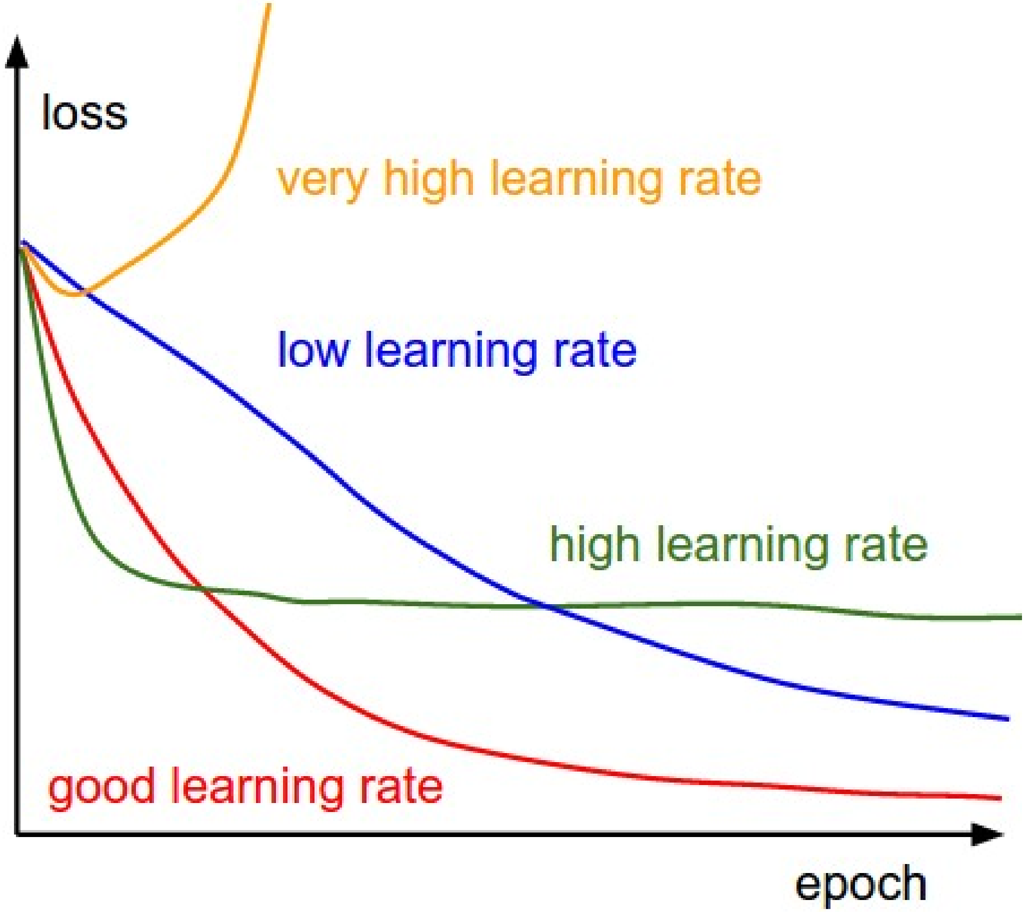
"ridge" regression



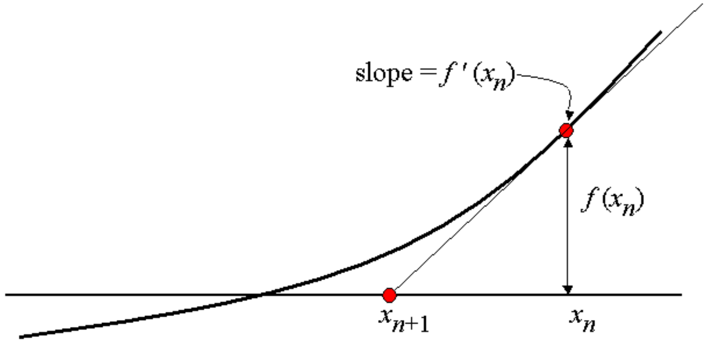


Learning Rate





$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$



Use a validation set:

Divide the total dataset into three subsets:

- Training data is used for learning the parameters of the model.
- Validation data is not used for learning but is used for deciding what type of model and what amount of regularization works best.
- Test data is used to get a final, unbiased estimate of how well the learning machine works. We expect this estimate to be worse than on the validation data.

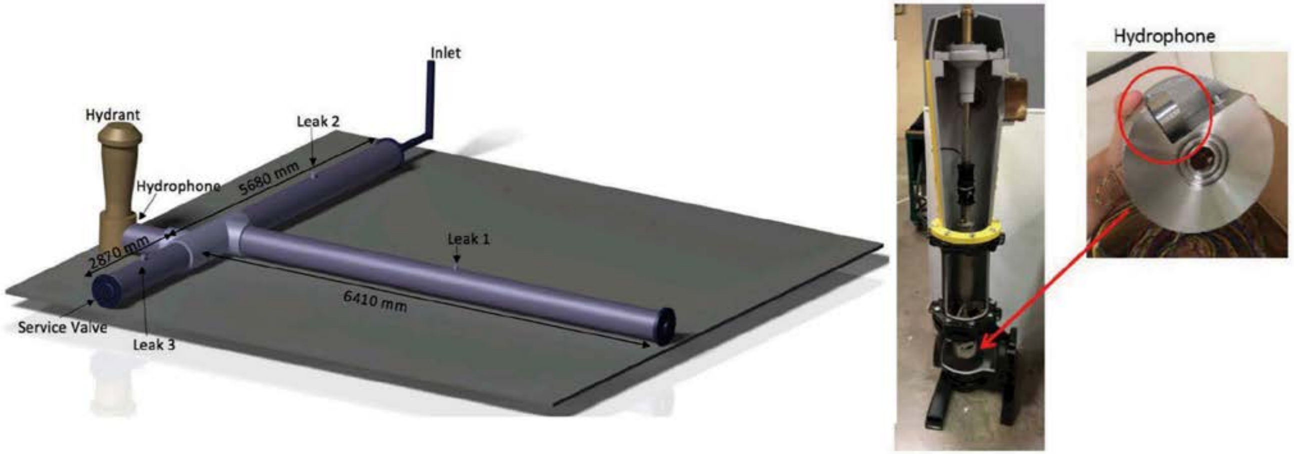
We could then re-divide the total dataset to get another unbiased estimate of the true error rate.

Example: Damage Detection

Collapse classification

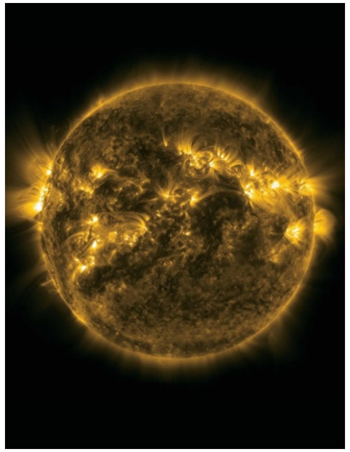
- Supervised
- Unsupervised
- Semi-supervised

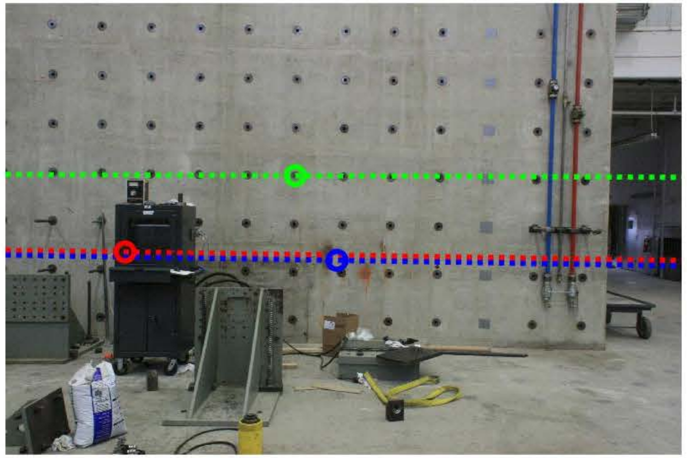
Pipeline inspection

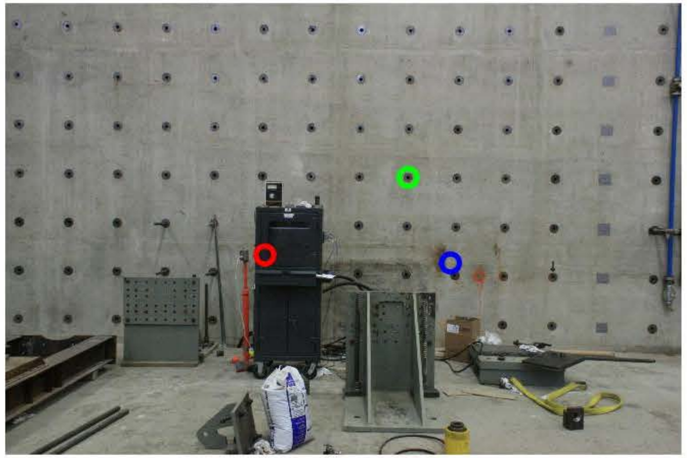


Leak detection (Cody et al, 2018)

Example: Homography Estimation and Fundamental Matrix

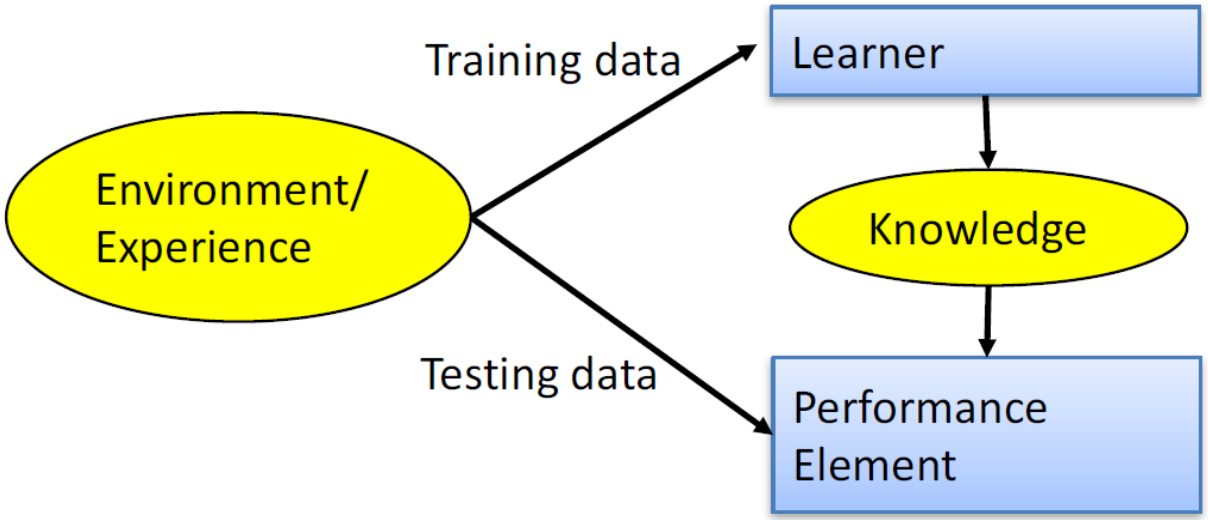






Designing a Learning System

- Choose the training experience
- Choose exactly what is to be learned
  - i.e. the ***target function***
- Choose how to represent the target function
- Choose a learning algorithm to infer the target function from the experience



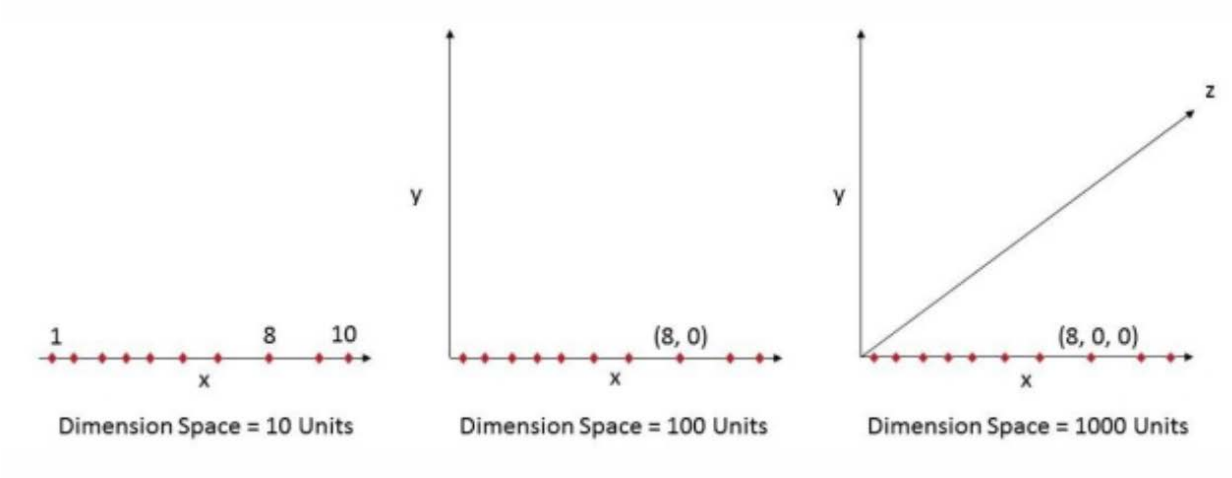
Slide credit: Ray Mooney

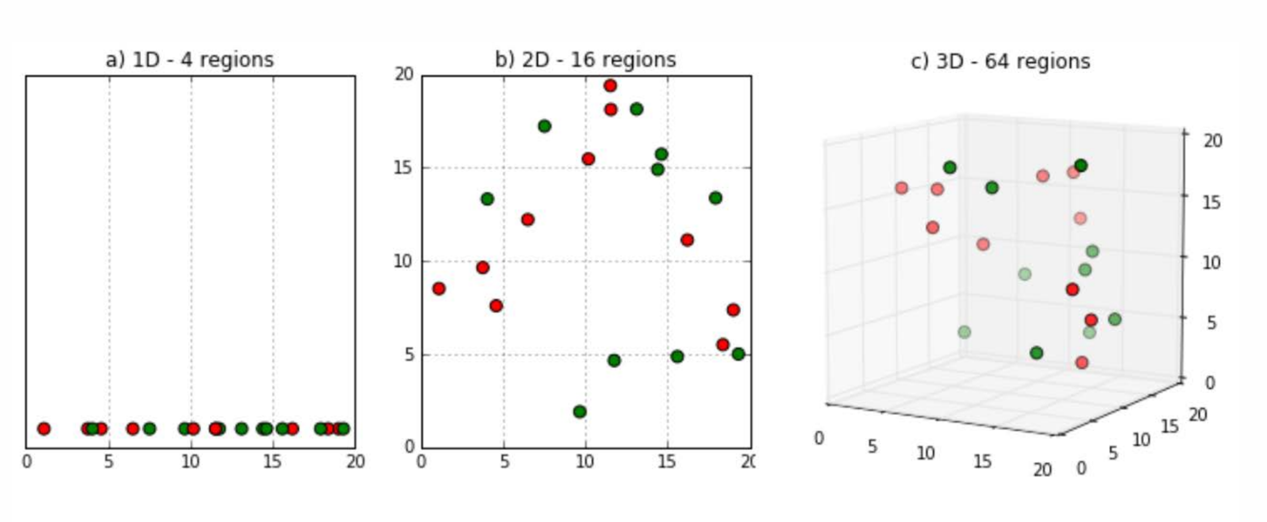
Loop

- Understand domain, prior knowledge, and goals
- Data integration, selection, cleaning, pre-processing, etc.
- Learn models
- Interpret results
- Consolidate and deploy discovered knowledge

Slide credit: Pedro Domingos

Curse of Dimensionality





- "As the number of features or dimensions grows, the amount of data we need to generalize accurately grows exponentially," Charles Isbell
- Think of image recognition problem of high resolution images 1280 × 720 = 921,600 pixels i.e. 921600 dimensions.
- That's why it's called **Curse of Dimensionality**. Value added by additional dimension is much smaller compared to overhead it adds to the algorithm.

SIFT Descriptor

https://www.kdnuggets.com/2017/04/must-know-curse-dimensionality.html

Example: Summation of N Numbers from a to a+N

$$\sum_{i=1}^{N} i = \frac{N(N+1)}{2}$$

$$\sum_{i=a}^{N+a} i = \frac{N(a+N+1)}{2}$$

Given $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$

How to design your problem?

Q and A

Slide Credits and References

- Lecture notes: Eric Eaton
- Lecture notes: A. Zisserman
- Lecture notes: David Sontag
- Bishop (2006) Pattern Recognition and Machine Learning