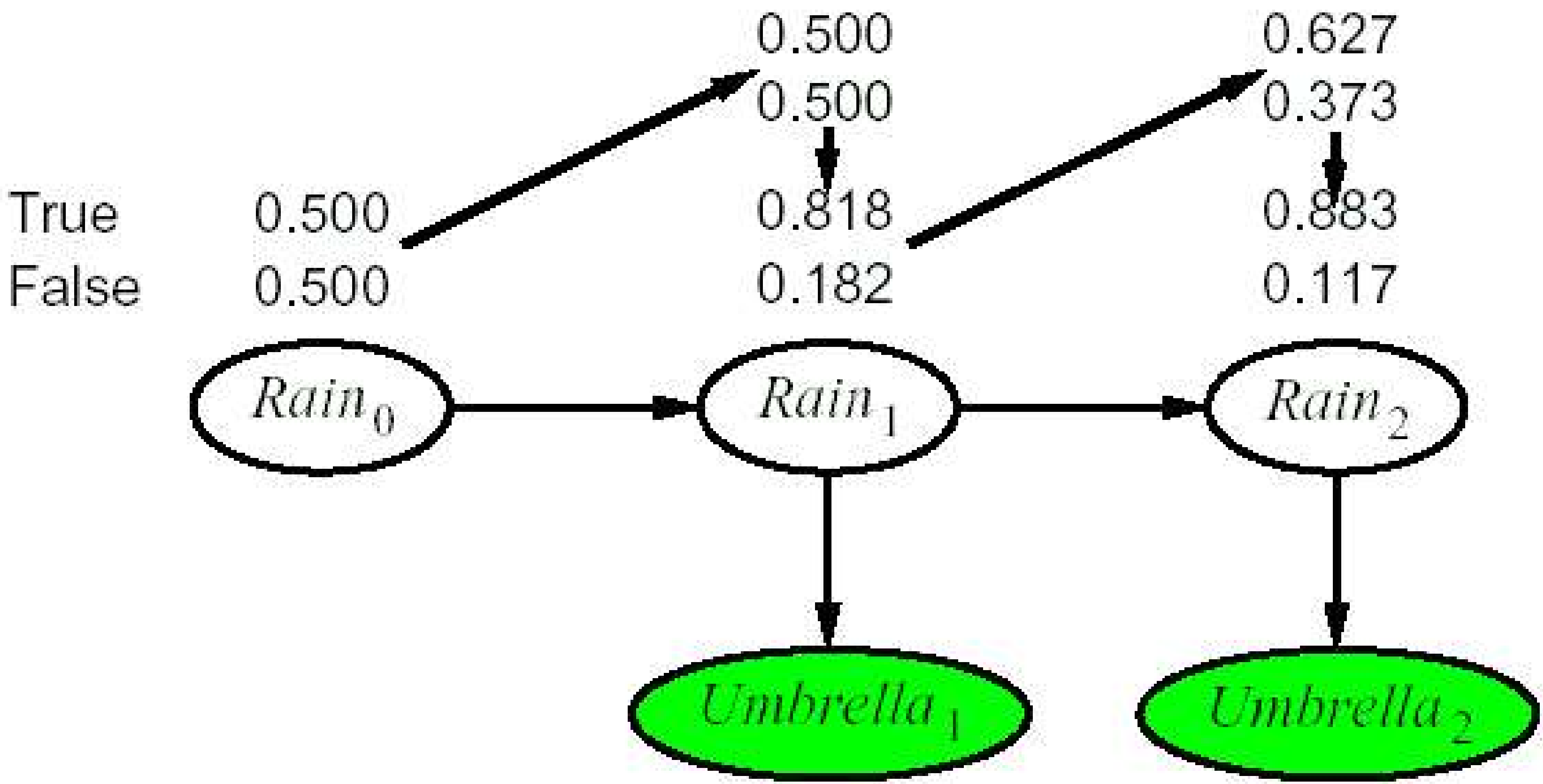
CSC421 Intro to Artificial Intelligence

UNIT 28: Learning from observations (+ leftovers from Prob. Reasoning over Time)

Filtering Example



Smoothing

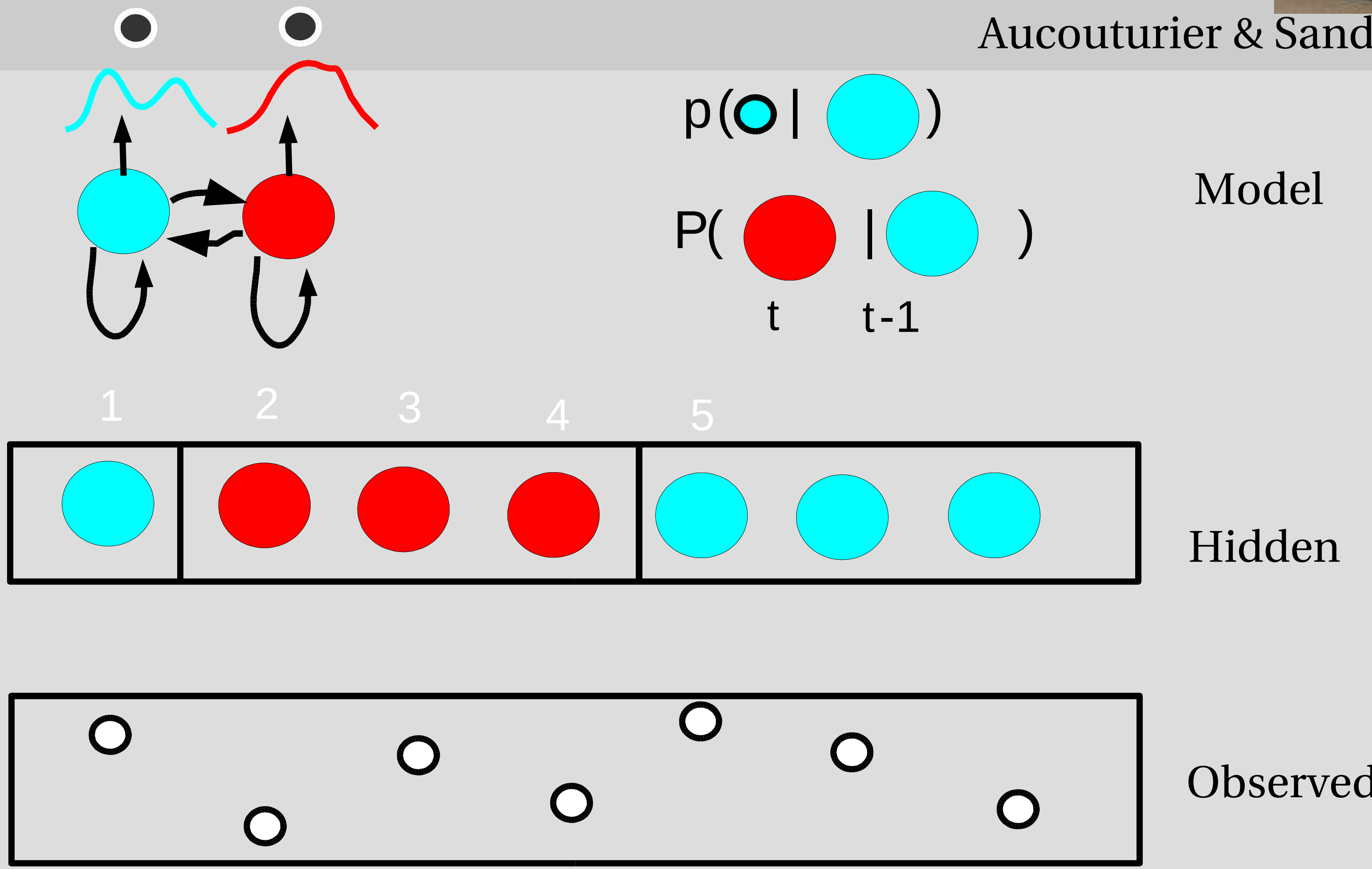
- Divide evidence $e_{1:t}$ to $e_{1:k}$, $e_{k+1,t}$
  - $P(X_k \mid e_{1:t}) = P(X_k \mid e_{1:k}, e_{k+1,t})$
    $= a\, P(X_k \mid e_{1:k})\, P(e_{k+1,t} \mid X_k, e_{k+1,t})$
    $= a\, P(X_k \mid e_{1:k})\, P(e_{k+1,t} \mid X_k)$
    $= a f_{1:k} b_{k+1:t}$
- Backward message computed by a backwards recursion

Hidden Markov Models

- Dominant method for automatic speech recognition
- Temporal probabilistic model in which the state of the process is described by a single discrete random variable
- Simple and elegant matrix implementation of all the basic algorithms

HMM segmentation

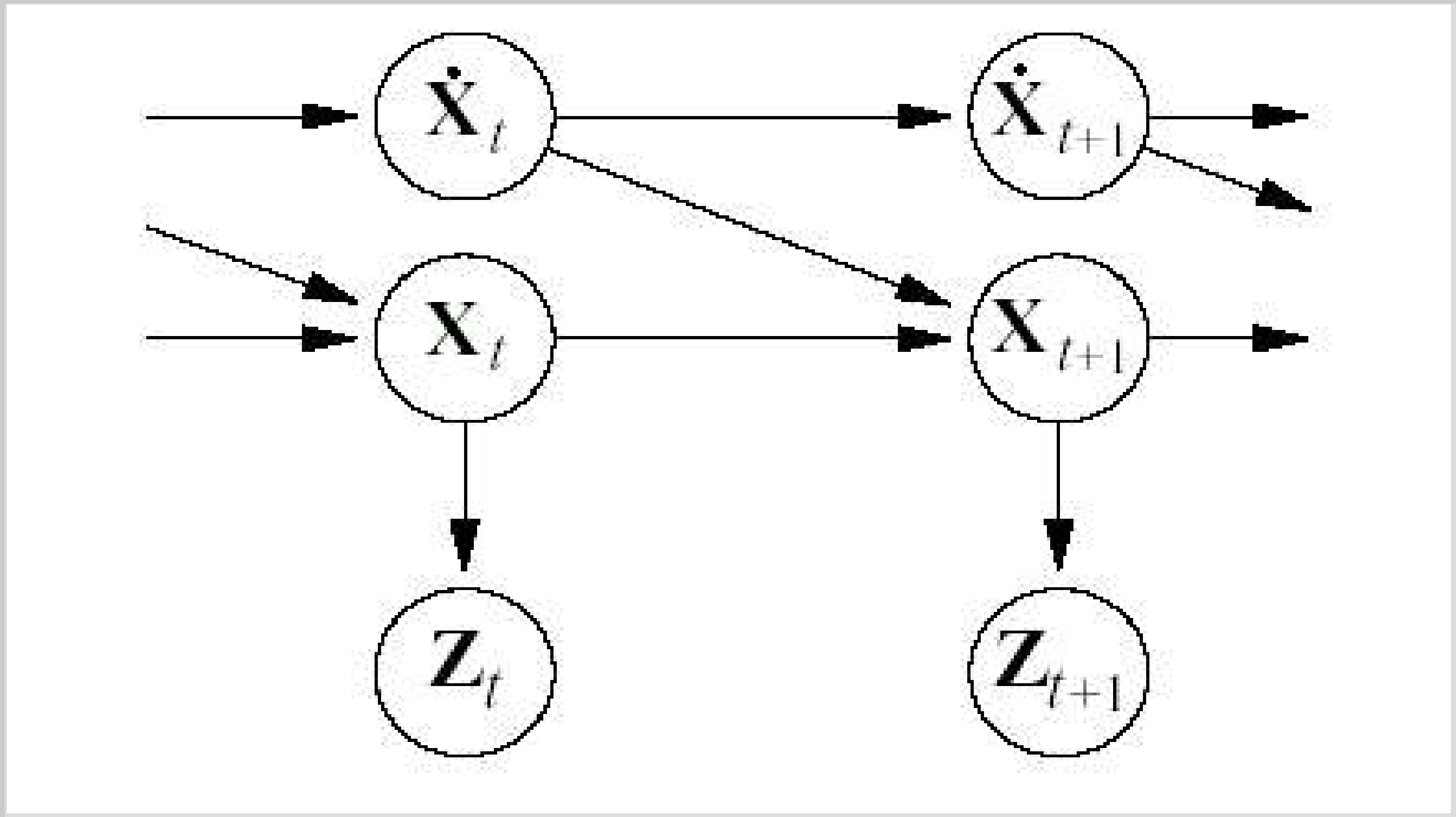
Aucouturier & Sandler, AES 01



Kalman Filter

- Modelling systems described by a set of continuous variables
  - Tracking a bird flying Xt = X,Y,Z, dX, dY, dZ
  - Airplanes, robots, ecosystems...
- Gaussian prior, linear Gaussian transition model and sensor model
  - i.e next state is a linear function of current state plus some Gaussian noise
  - Key property: linear Gaussian family remains closed under standard Baysian network operations

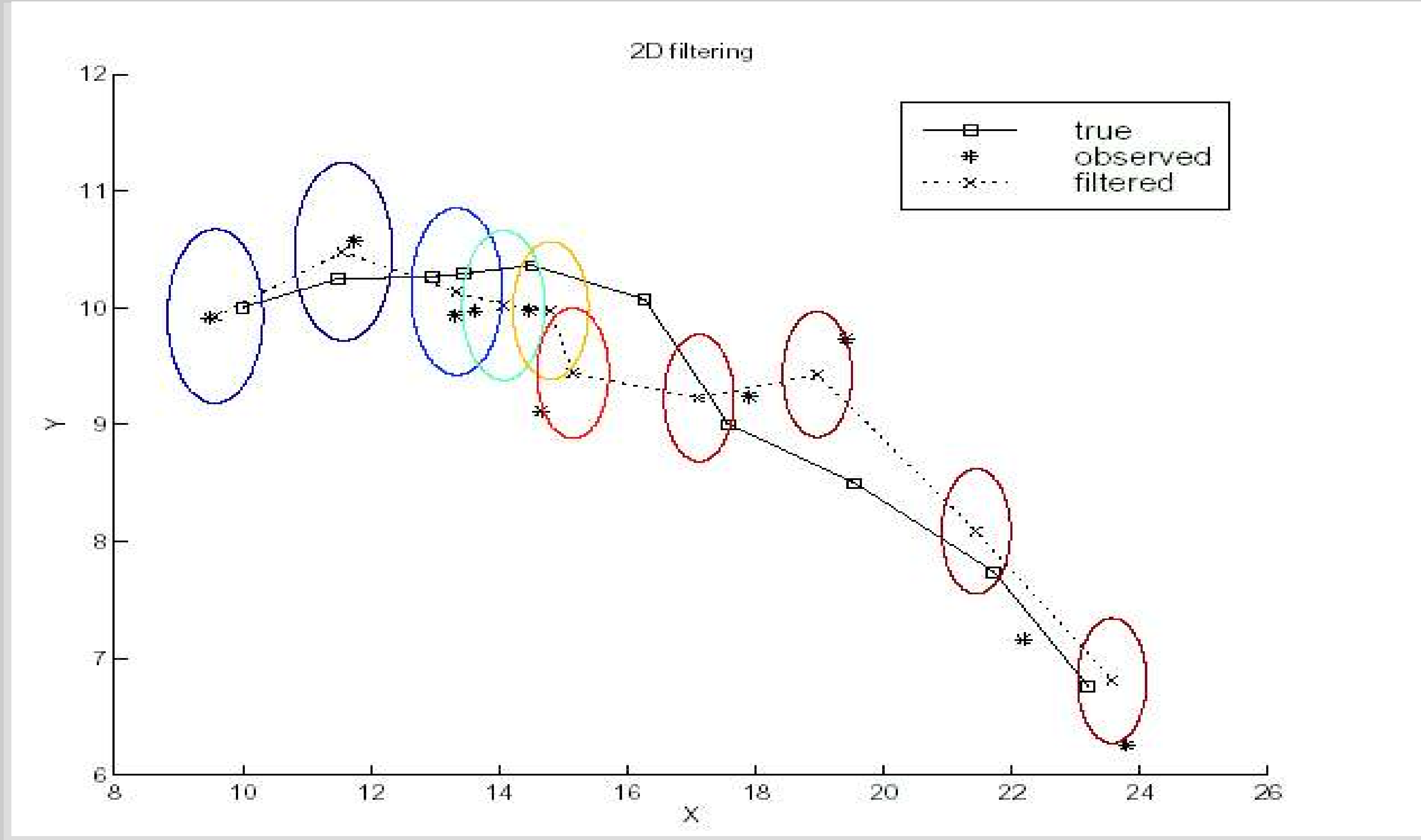
Kalman Filter



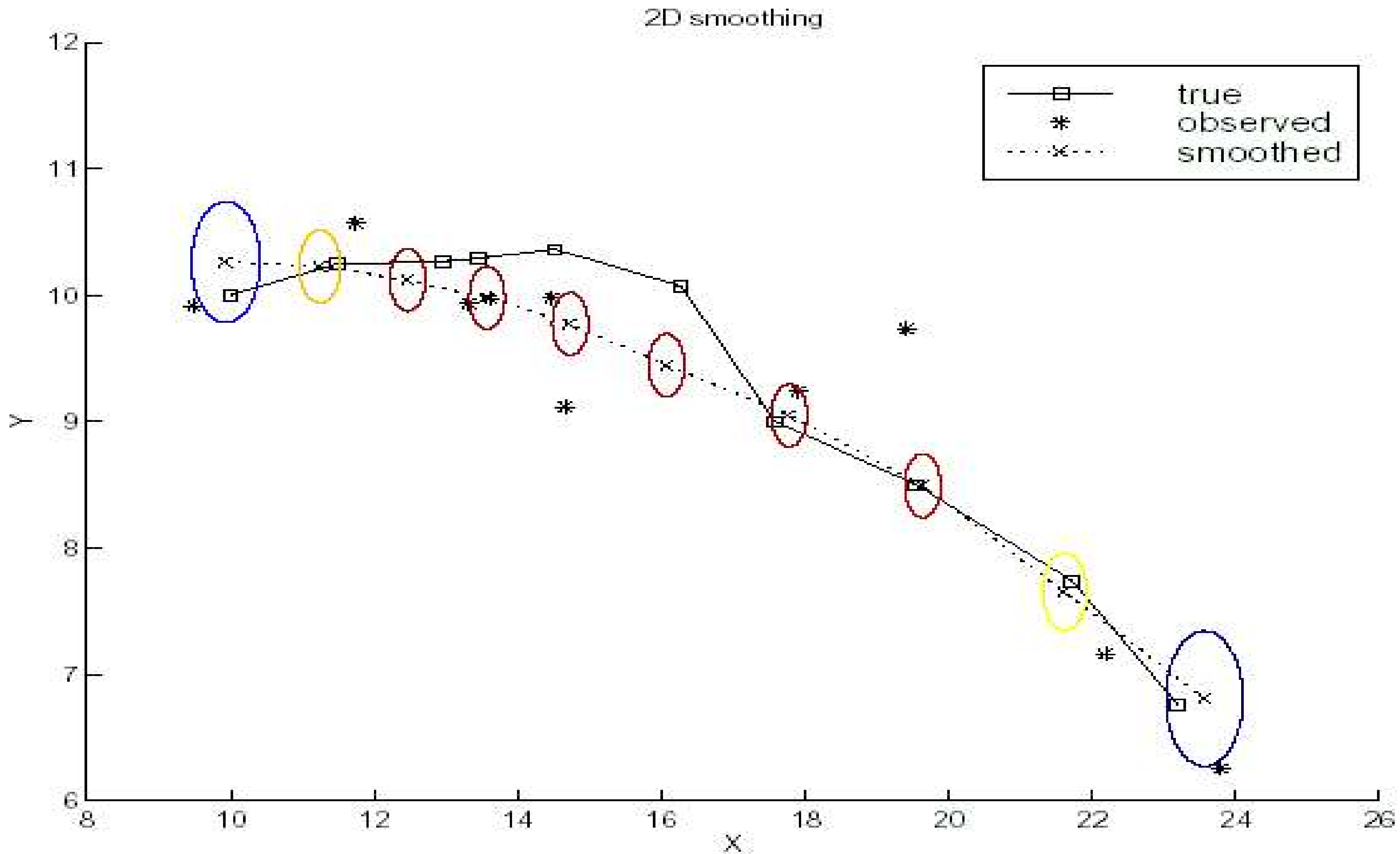
Position, velocity and position measurement

Basically we forward messages (means + covariance matrix) to produce new message (means + covariance matrix)

Kalman filtering



Kalman Smoothing



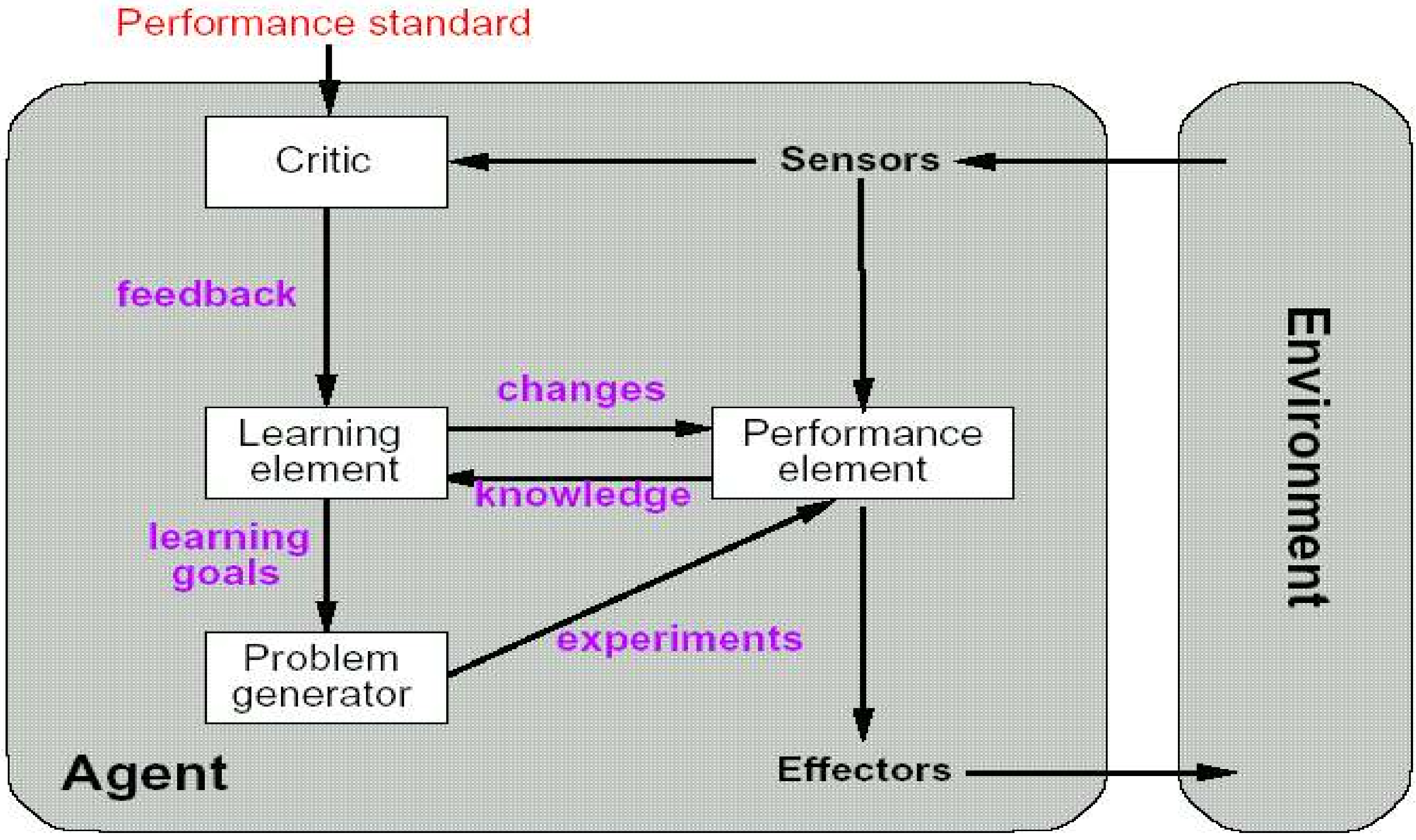
Outline

- Learning agents
- Inductive Learning
- Decision tree learning
- Measuring learning performance

Learning

- Learning is essential for unknown environs
  - Designer lacks omniscience
- Learning is useful as a system construction method
  - Expose the agent to reality rather than trying to write it down
- Learning modifies agents decision making mechanisms to improve performance

Learning Agents



Learning Element

- Design is dictated by:
  - Type of performance element
  - Functional component to be learned
  - How functional component is represented
  - Type of feedback

Example scenarios

Performance element	Component	Representation	Feedback
Alpha−beta search	Eval. fn.	Weighted linear function	Win/loss
Logical agent	Transition model	Successor−state axioms	Outcome
Utility−based agent	Transition model	Dynamic Bayes net	Outcome
Simple reflex agent	Percept−action fn	Neural net	Correct action

Supervised learning: correct answers for each training instance
Unsupervised learning: no feedback
Reinforcement learning: occasional rewards

Other terms: Classification, Regression, Clustering, Online, Offline

Inductive Learning

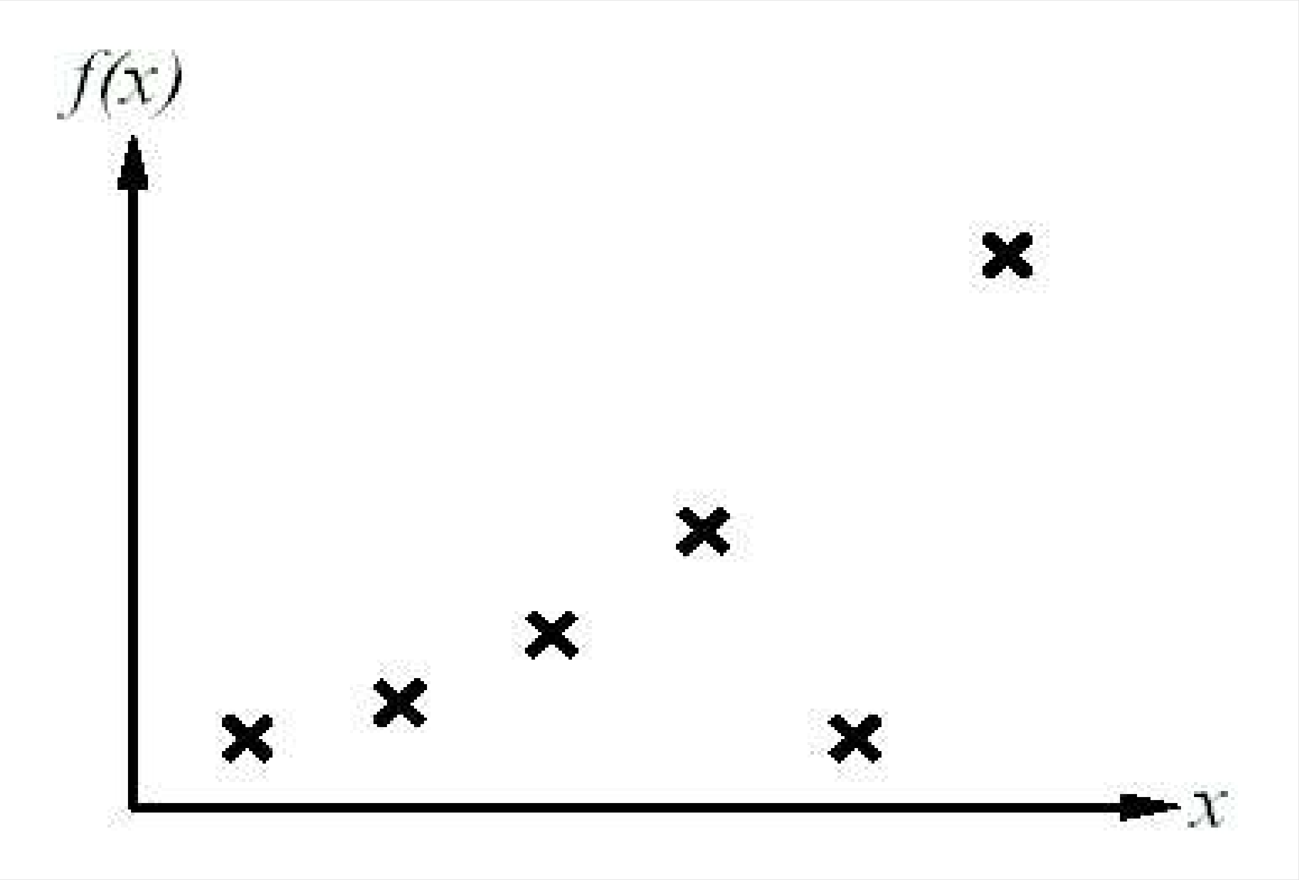
- Simplest form: “learn” a function from examples (training set)
  - An example is a pair x, f(x)
  - Find hypothesis h(x) such that h(x) ≈ f(x) given a training set of example
- Highly simplified. Why ?

Inductive Learning

- Learning a function is simplified:
  - Ignores prior knowledge
  - Assumes deterministic observable environment
  - Assumes examples are given
  - Assumes that the agent wants to learn f – why ?

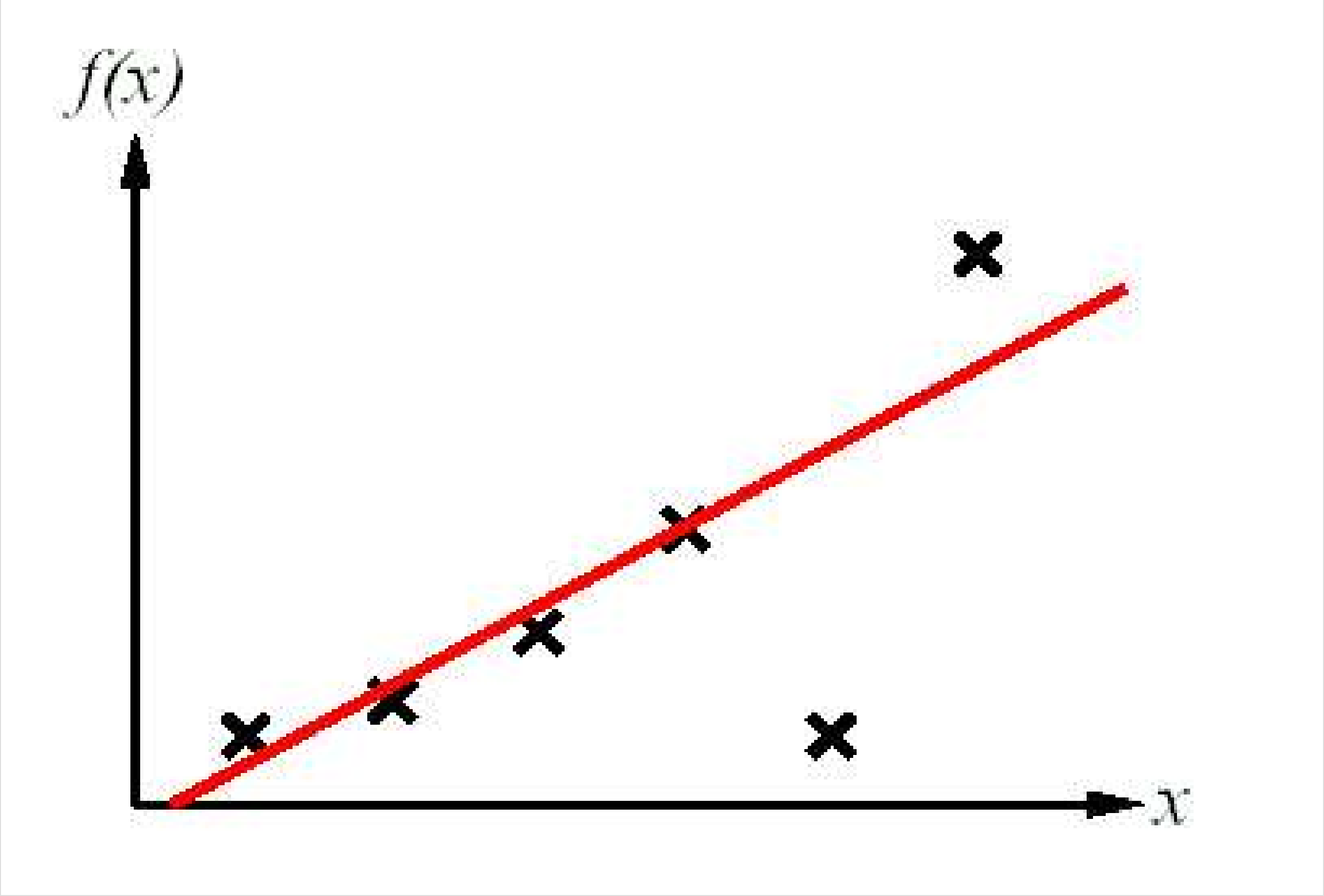
Inductive Learning Method

- Construct/adjust h to agree with f on training set e.g curve fitting



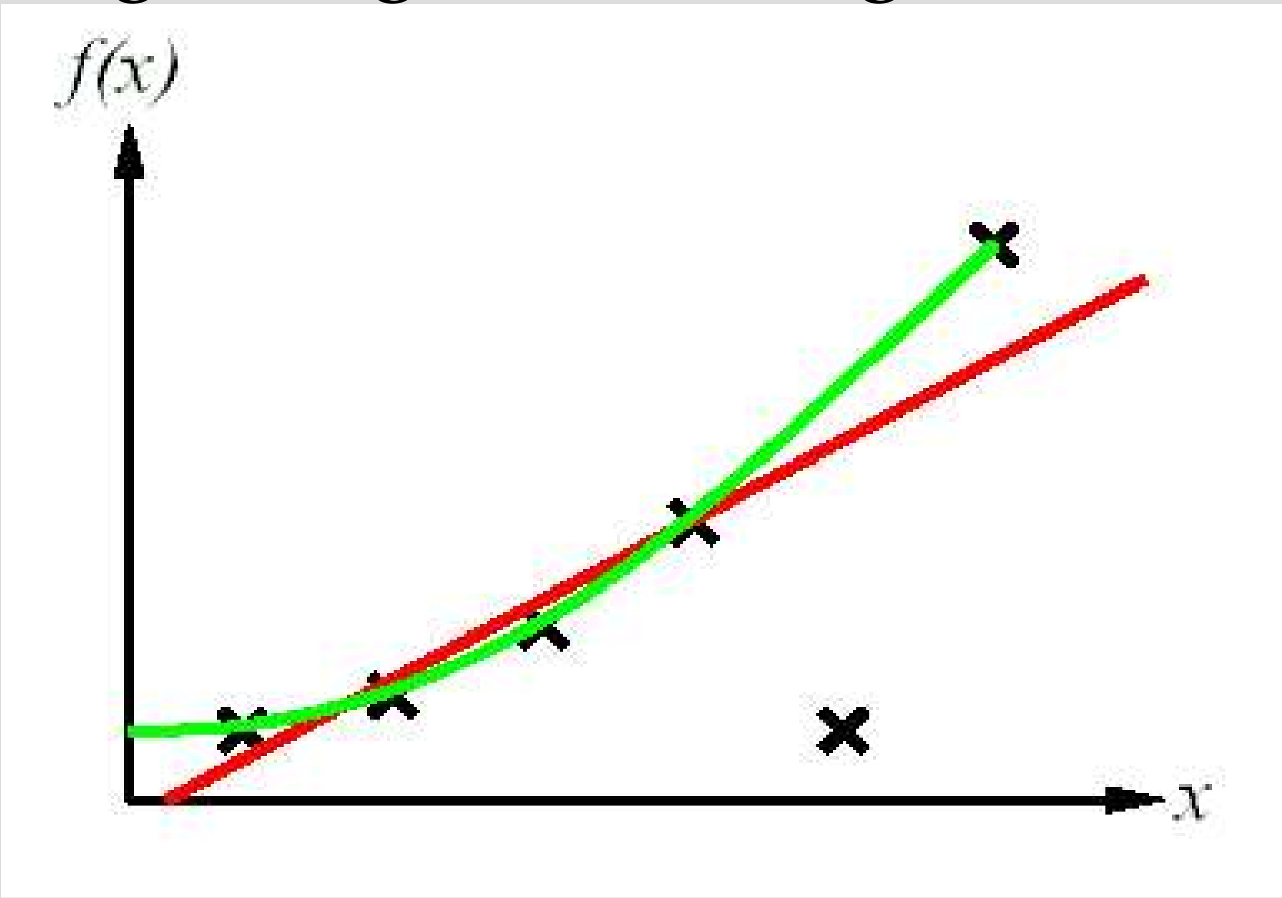
Inductive Learning Method

- Construct/adjust h to agree with f on training set e.g curve fitting



Inductive Learning Method

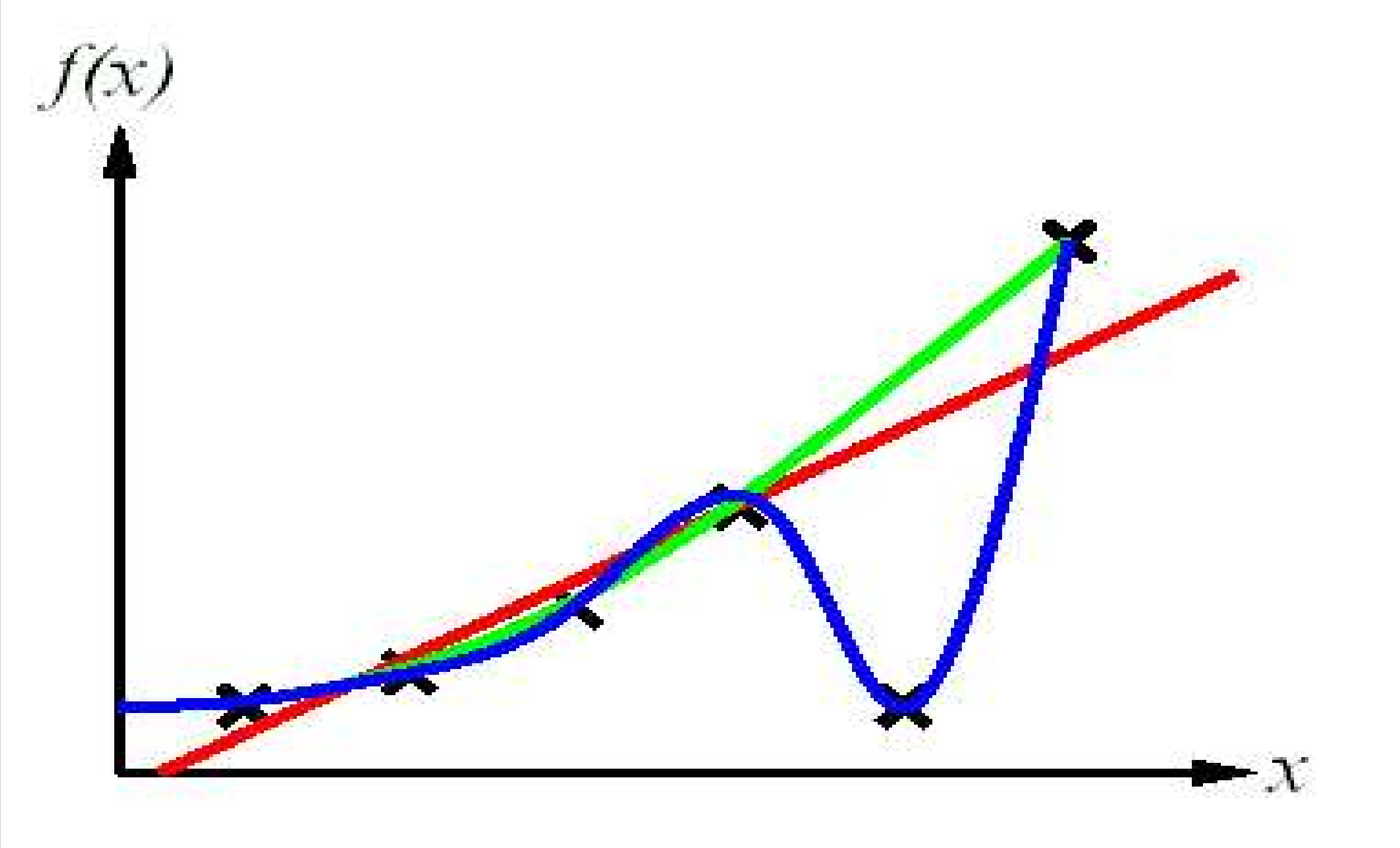
- Construct/adjust h to agree with f on training set e.g curve fitting



CONSISTENT HYPOTHESIS

Inductive Learning Method

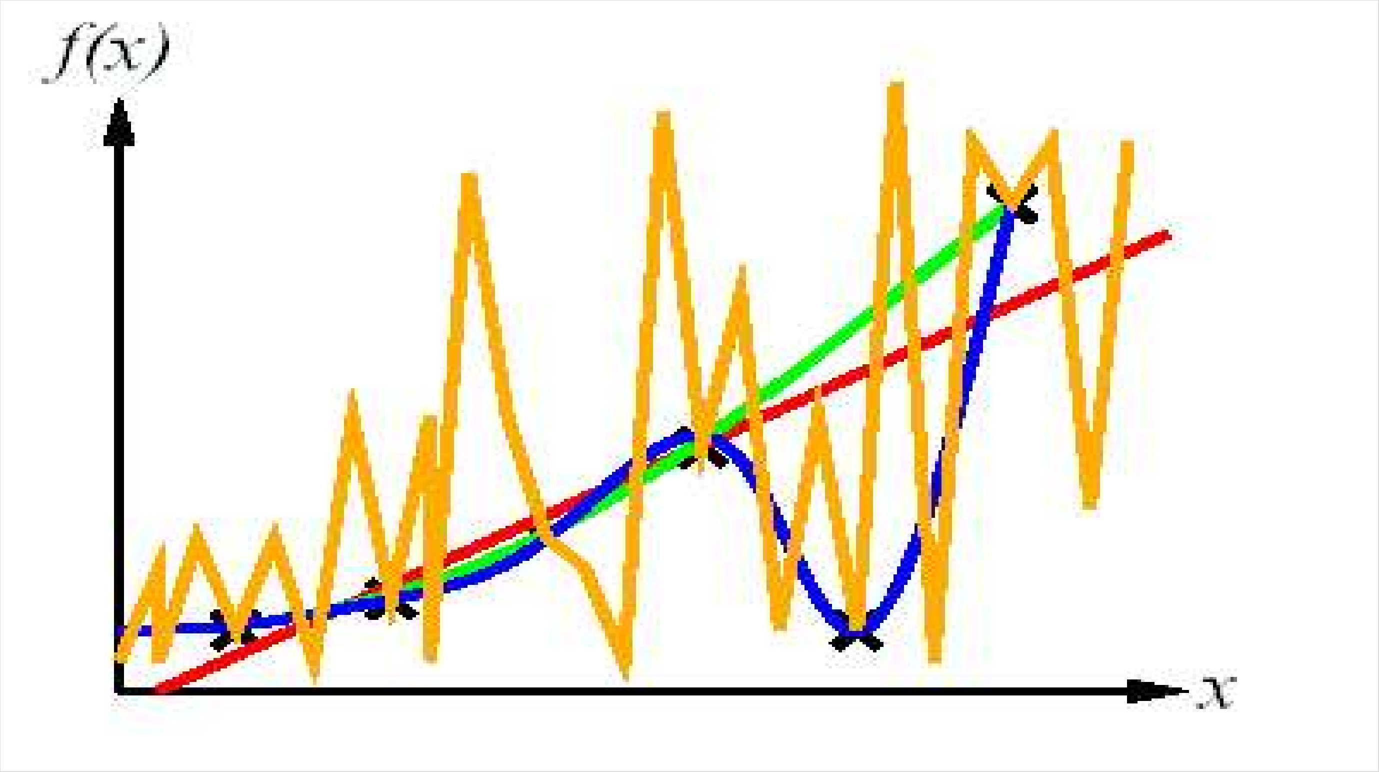
- Construct/adjust h to agree with f on training set e.g curve fitting



CONSISTENT
HYPOTHESIS

Inductive Learning Method

- Construct/adjust h to agree with f on training set e.g curve fitting



CONSISTENT HYPOTHESIS

OCCAM'S RAZOR
Maximize a combination of consistency and simplicity