Machine Learning

Dadet Pramadihanto, Ph.D

Contents

- Background and Motivation
- What is learning
- What is machine learning
- How can we specify a learning problem?
- Examples of learning algorithms
- Representative applications in bioinformatics

Background - Computer Science

- Computer science is the science of information processing – theory and practice of representation, processing, and use of information.
- Computer Science offers a powerful paradigm for modeling complex phenomena such as cognition and life, and representing, processing, acquiring, and communicating knowledge that is new in the history of humanity.

Computer Science and Artificial Intelligence

- It is the human urge to understand what makes us human – our ability to perceive, think, reason, learn, and act – that led to the birth of Computer Science.
- The language of computation is the best language we have so far for describing how information is encoded, stored, manipulated and used by natural as well as synthetic systems
- Algorithmic or information processing models provide for biological, cognitive, and social sciences what calculus provided for classical physics
- Computation : Cognition : : Calculus : Physics (Artificial Intelligence, Cognitive Science)

Algorithmic explanations of mind

- Computation : Cognition : : Calculus : Physics (Artificial Intelligence, Cognitive Science)
  - What are the information requirements of learning?
  - What is the algorithmic basis of learning?
  - What is the algorithmic basis of rational decision making?
  - Can we automate scientific discovery?
  - Can we automate creativity?

Algorithmic explanations of mind

- Computation: Cognitive Science :: Calculus : Physics
- Computer science offers fundamentally new ways to understand cognitive processes –
  - Perception
  - Memory and learning
  - Reasoning and planning
  - Rational decisions and problem solving
  - Communication and Language
  - Behavior

Conceptual impact of computer science

- Pre-Turing
  - Focus on physical basis of the universe with the objective of explaining all natural phenomena in terms of physical processes
- Post-Turing
  - Focus on informational and algorithmic basis of the universe with the objective of explaining natural phenomena in terms of processes that acquire, store, process, manipulate, and use information
  - We understand a phenomenon when we can write a computer program that models it at the desired level of detail
  - Computer science offers fundamentally new ways to model and understand cognitive, biological, and social processes through computational or information processing or algorithmic models

Background-
Artificial Intelligence

- Computation : Cognition :: Calculus: Physics
  - Algorithms or computation or information processing provide for study of cognition what calculus provided for physics
- We have a theory of intelligent behavior when we have precise information processing models (computer programs) that produce such behavior
- We will have a theory of learning when we have precise information processing models of learning (computer programs that learn from experience)

Why should machine learn ?

- Some tasks are best specified by example (e.g., credit risk assessment, face recognition)
- Some tasks are best shown by demonstration (e.g., landing an airplane)
- Buried in large volume of data are useful predictive relationships (data mining)
- The operating environment of certain types of software (user characteristics, distribution of problem instances) may not be completely known at design time
- Environment changes over time – ability of software to adapt to changes would enhance usability

Why study machine learning ? (practical)

- Intelligent behavior requires knowledge. Explicitly specifying the knowledge needed for specific tasks is hard, and often infeasible
- Machine Learning is most useful when
  - the structure of the task is not well understood but a representative dataset is available
  - task (or its parameters) change dynamically
- If we can program computers to learn from experience, we can
  - Dramatically enhance the usability of software
  - Dramatically reduce the cost of software development
  - Automate discovery

Why study machine learning ? (Application)

- Spam Filtering, fraud detection (e.g. credit cards, phone calls)
- Search and recommendation (e.g. google, amazon)
- Automatic speech recognition & speaker verification
- Printed and handwritten text parsing
- Locating/tracking/identifying objects in images & video (e.g. faces)
- Financial prediction, pricing, volatility analysis
- Medical diagnosis/image analysis (e.g. pneumonia, pap smears)
- Driving computer players in games
- Scientific discovery

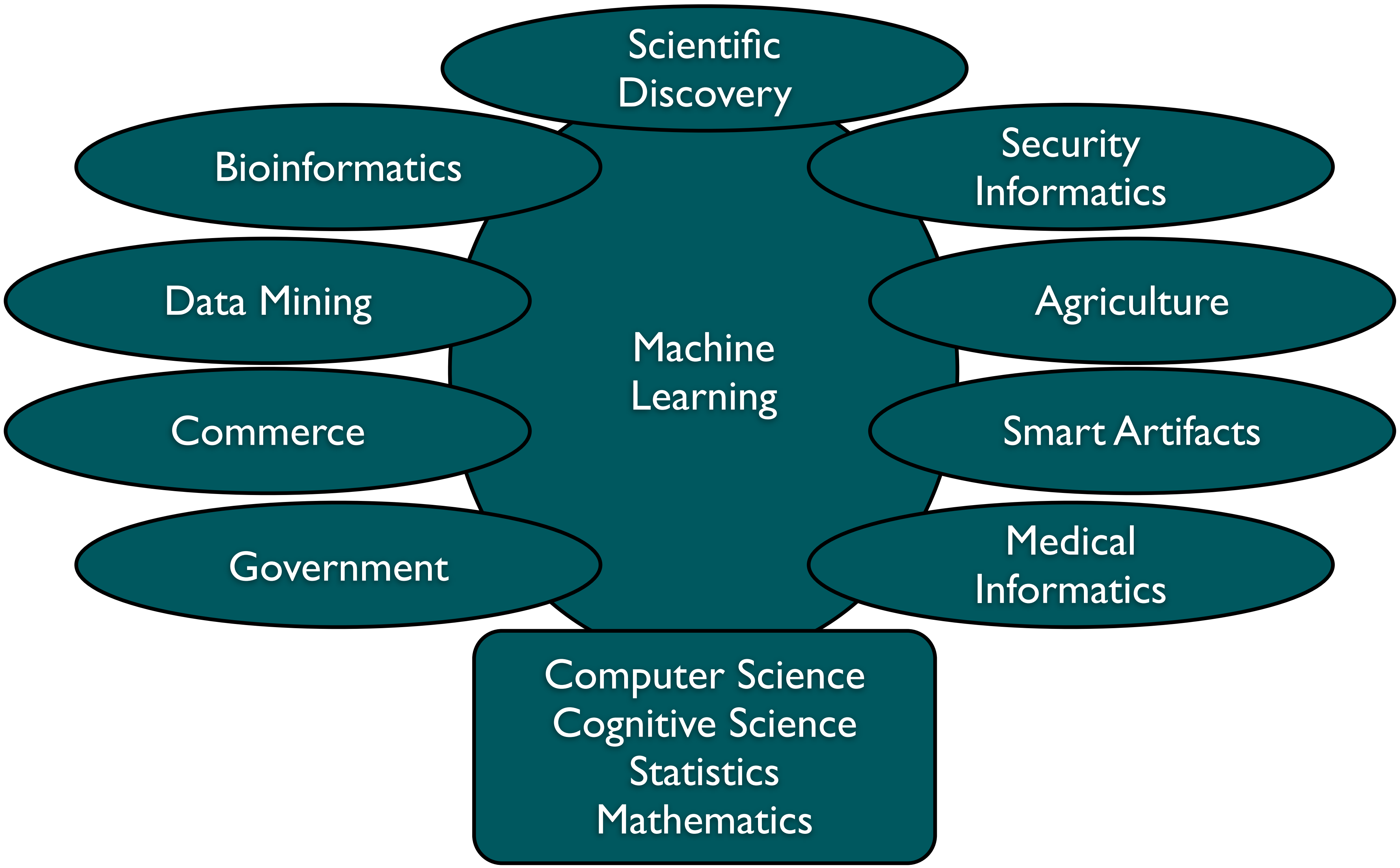
Examples of applications

- Using historical data to improve decisions
  - credit risk assessment, diagnosis, electric power usage prediction
- Using scientific data to acquire knowledge
  - in computational molecular biology
- Software applications that are hard to program
  - autonomous driving
  - face recognition,
  - speech recognition
- Self-customizing programs
  - newsreader that learns user interests

Why study machine learning ? (Scientific)

- Information processing models can provide useful insights into
  - How humans and animals learn
  - Information requirements of learning tasks
  - The precise conditions under which certain learning goals are achievable
  - Inherent difficulty of learning tasks
  - How to improve learning – e.g. value of active versus passive learning
  - Computational architectures for learning

Machine Learning in Context



Machine Learning : Applications

- Bioinformatics and Computational Biology
- Cognitive Science
- e-Commerce, e-Enterprises, e-Government, e-Science
- Environmental Informatics
- Human Computer Interaction
- Intelligent Information Infrastructure
- Medical Informatics
- Security Informatics
- Smart Artifacts
- Robotics
- Engineering

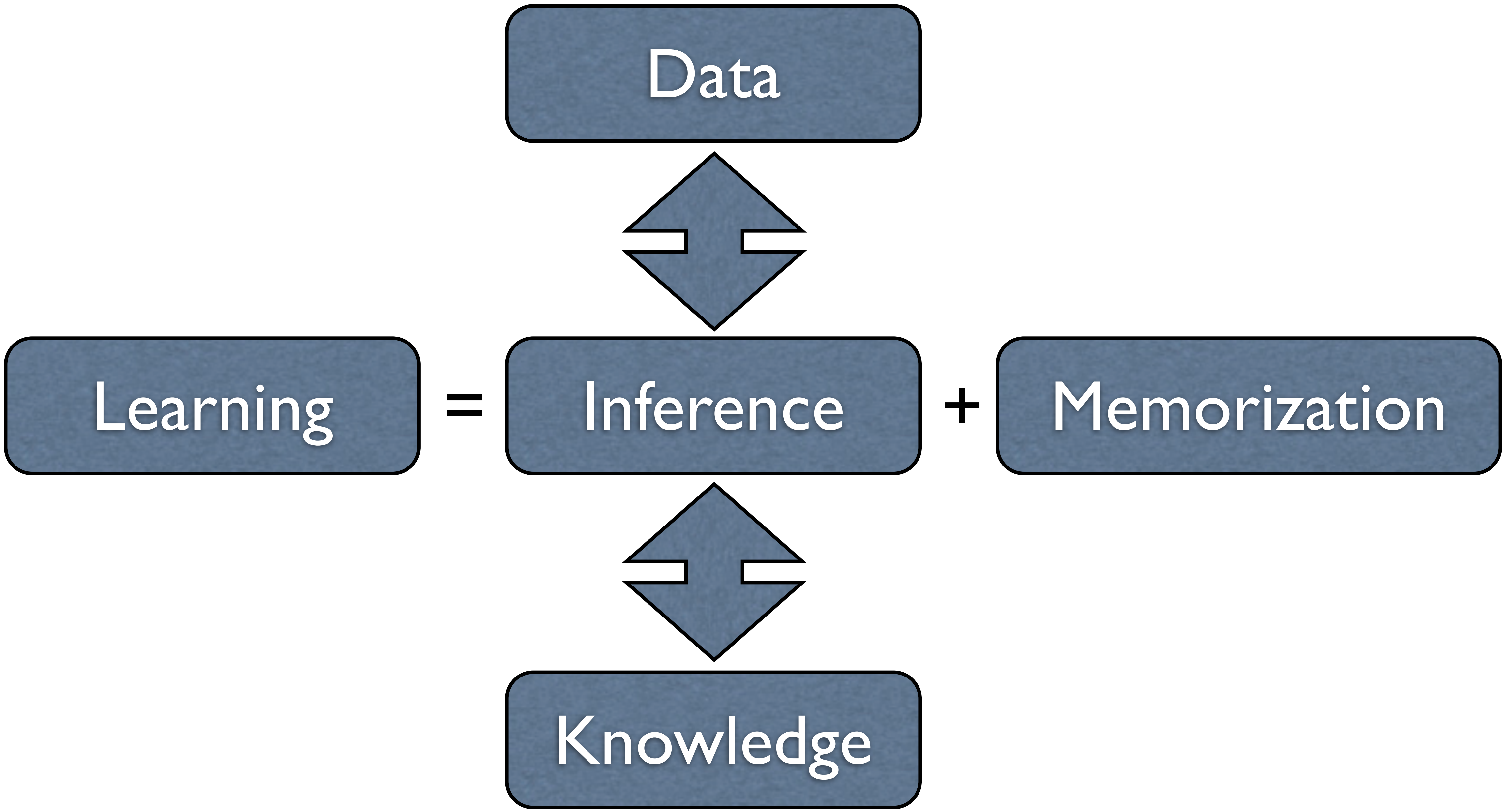
Machine Learning : Contributing Disciplines

- Computer Science – Artificial Intelligence, Algorithms and Complexity, Databases, Data Mining
- Statistics – Statistical Inference, Experiment Design, Exploratory Data Analysis
- Mathematics – Abstract Algebra, Logic, Information Theory, Probability Theory
- Psychology and Neuroscience – Behavior, Perception, Learning, Memory, Problem solving
- Philosophy – Ontology, Epistemology, Philosophy of Mind, Philosophy of Science

Machine Learning - Related disciplines

- Data mining – emphasis on large data sets, computational and memory considerations
- Applied Statistics – applied almost always to small data sets, manually by a statistician sometimes assisted by a computer
- Machine learning – emphasis on automating the discovery of regularities from data, characterizing what can be learned and under what conditions, obtaining guarantees regarding quality of learned models
- Machine Learning = (Statistical) Inference + Data Structures + Algorithms

What is learning ?



Inference

- Deduction

$$\frac{\begin{array}{l}\forall x\ At(Smoke,x) \Rightarrow At(Fire,x)\\ At(Smoke,Room1)\end{array}}{At(Fire,Room1)}$$

- Induction

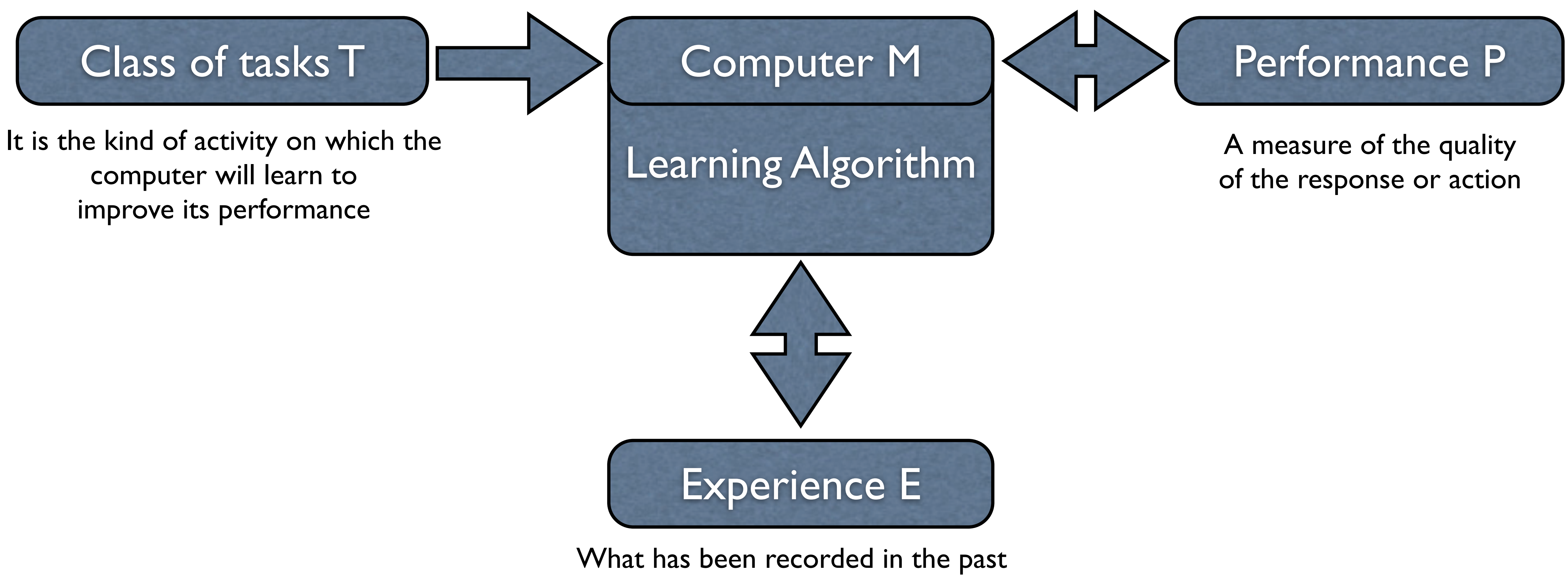
$$\frac{\begin{array}{l}At(Smoke,Room2) \wedge At(Fire,Room2)\\ At(Smoke,Room1) \wedge At(Fire,Room1)\\ At(Ice,Room3) \wedge \neg At(Fire,Room3)\end{array}}{\forall x\ At(Smoke,x) \Rightarrow At(Fire,x)?}$$

- Abduction

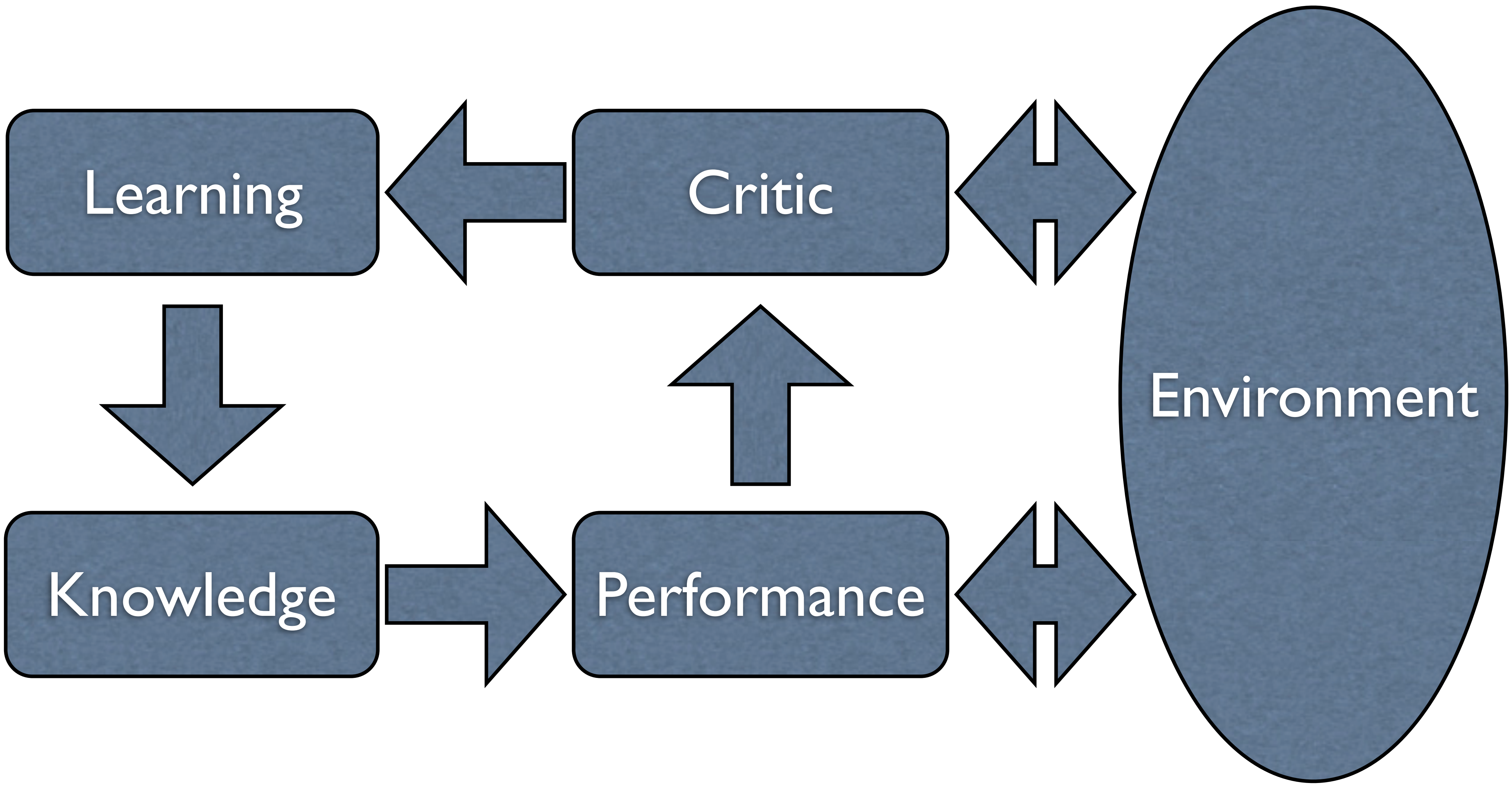
$$\frac{\begin{array}{l}\forall x\ At(Smoke,x) \Rightarrow At(Fire,x)\\ At(Fire,Room1)\end{array}}{At(Smoke,Room1)?}$$

What is machine learning ?

- A program M is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance as measured by P on tasks in T in an environment Z improves with experience E.



General framework for learning



Types of learning

- Rote Learning – useful when it is less expensive to store and retrieve some information than to compute it
- Learning from Instruction – transform instructions into operationally useful knowledge
- Learning form Examples (and counter-examples) – extract predictive or descriptive regularities from data
- Learning from Deduction (and explanation) – generalize instances of deductive problem-solving
- Learning from Exploration – learn to choose actions that maximize reward

Canonical learning problems

- Supervised Learning: given examples of inputs and corresponding desired outputs, predict outputs on future inputs.
  - Classification
  - Regression
  - Time series prediction
- Unsupervised Learning: given only inputs, automatically discover representations, features, structure, etc.
  - Clustering
  - Outlier detection
  - Compression
- Reinforcement Learning: given sequences of inputs, actions from a fixed set, and scalar rewards/punishments, learn to select actions in a way that maximizes expected reward.

Computational Model of Learning

- Model of the Learner: Computational capabilities, sensors, effectors, knowledge representation, inference mechanisms, prior knowledge, etc.
- Model of the Environment: Tasks to be learned, information sources (teacher, queries, experiments), performance measures
- Key questions: Can a learner with a certain structure learn a specified task in a particular environment? Can the learner do so efficiently? If so, how? If not, why not?

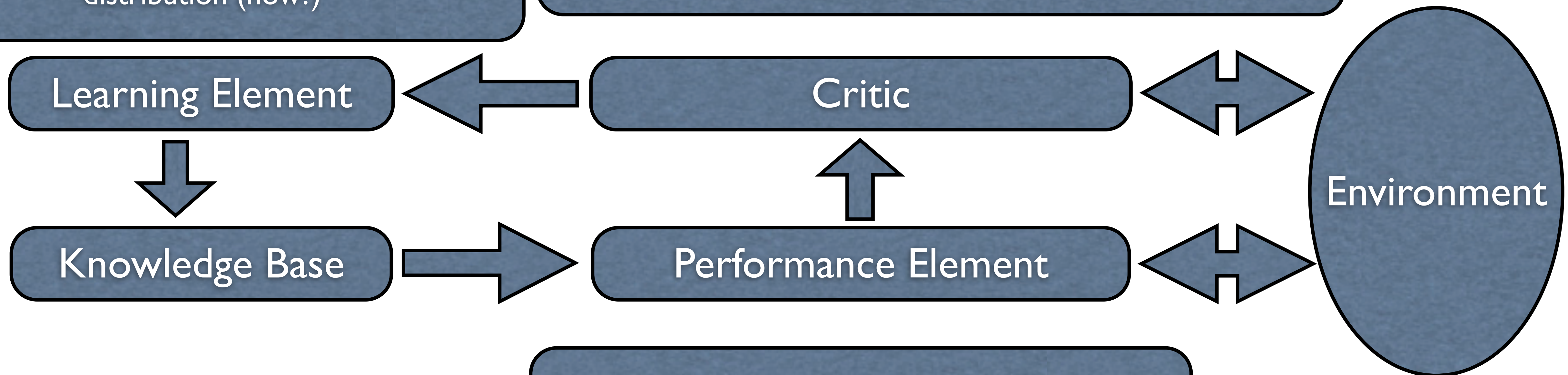
Models of Learning: What are they good for

- To make explicit relevant aspects of the learner and the environment
- To identify easy and hard learning problems (and the precise conditions under which they are easy or hard)
- To guide the design of learning systems
- To shed light on natural learning systems
- To help analyze the performance of learning systems

General framework for designing a learning program

Learning element
What is the learning algorithm?
search for a set of classification rules that are likely to perform well on novel cases (how?)
estimate a class conditional probability distribution (how?)

Critic –Can the learner ask questions?
What type of questions?
How am I doing? –performance query
How would you diagnose X? – example based query
Why was I wrong? –explanation



Experience–What experiences are available?
Data –in medical diagnosis, expert diagnosed cases, feedback
How representative is the experience?

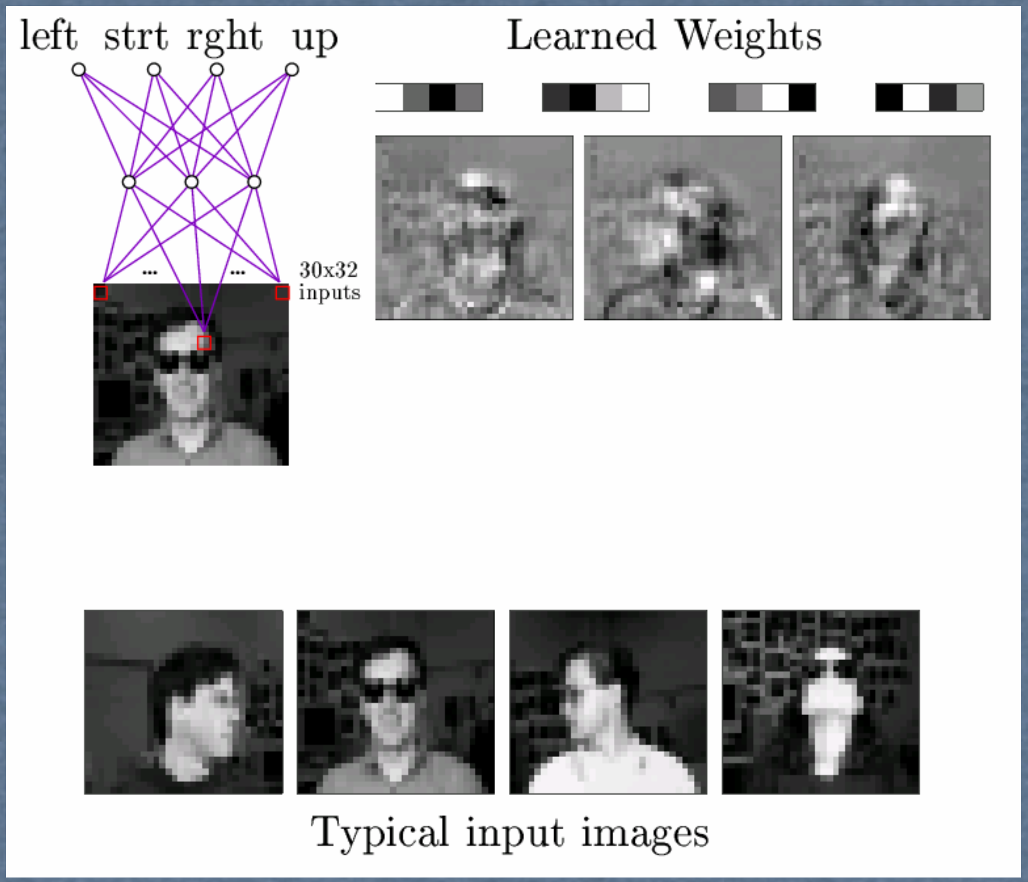
Performance element–
How is the learned knowledge encoded?
rules, probabilities, programs
How is the learned knowledge used?
e.g. matching rules
What is the performance measure?
How is performance measured?
online? batch?

Environment
Deterministic or stochastic?
Noisy or noise free?
...

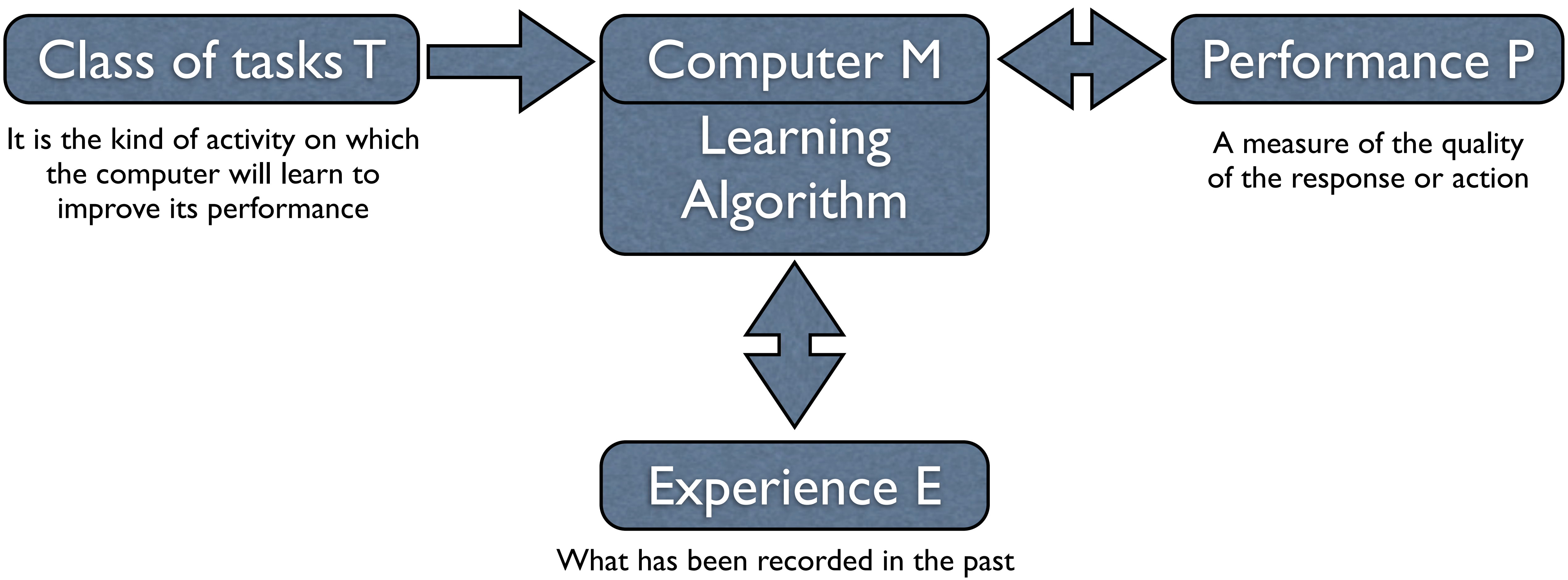
Designing a Learning Program

- Learning involves synthesis or adaptation of computational structures
  - Functions
  - Logic programs
  - Rules
  - Grammars
  - Probability distributions
  - Action policies
  - Behaviors
- Machine Learning = (Statistical) Inference + Data Structures + Algorithms

Some Example Machine Learning Application on Face Detection



Designing a Learning System



Designing a Learning System

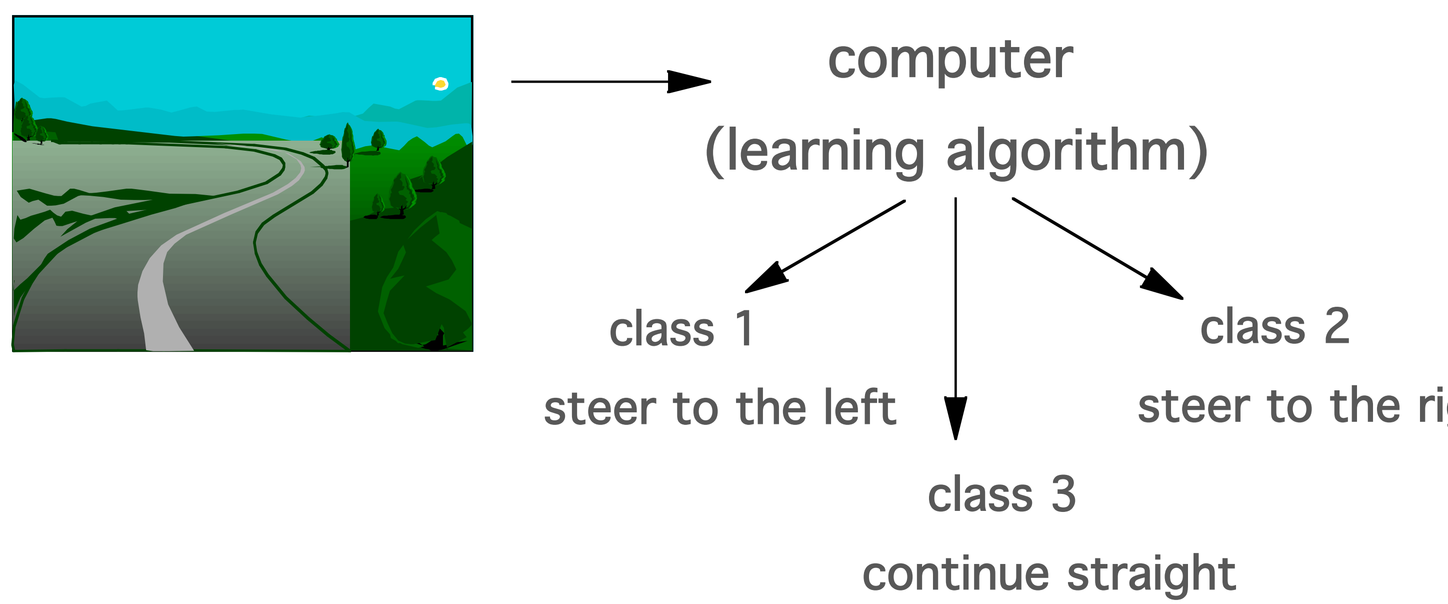
- Define the knowledge to learn
- Define the representation of the target knowledge
- Define the learning mechanism

- Handwritten recognition using Neural Networks
  - A function to classify handwritten images
  - A linear combination of handwritten features
  - A linear classifier

Automatic Car Driver

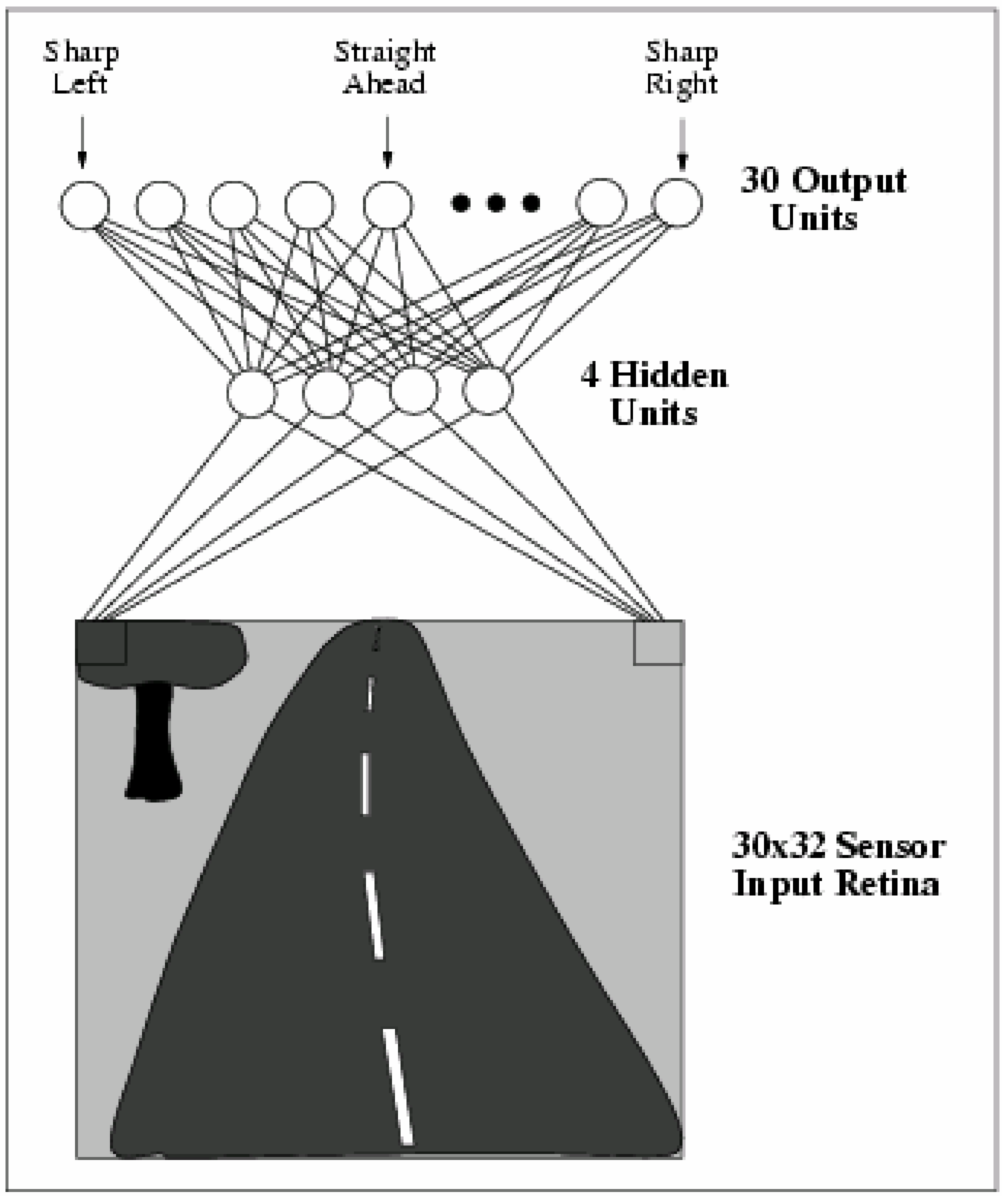
Automatic car drive (ALVINN 1989)

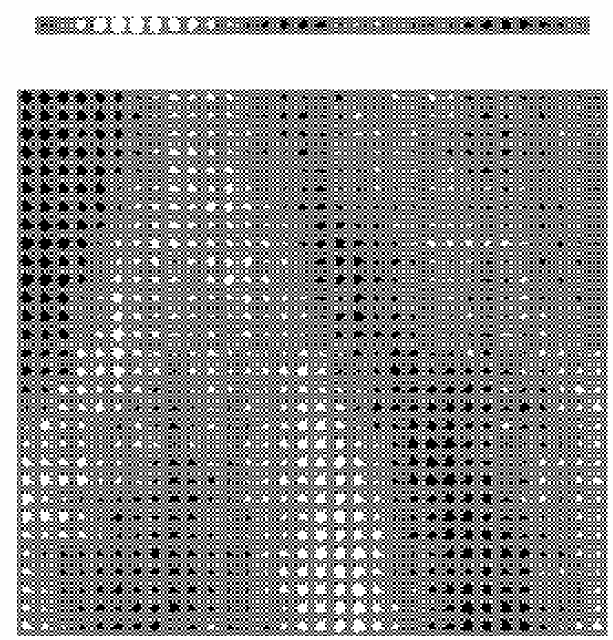
Train computer-controlled vehicle to steer correctly when driving on a variety of road types.



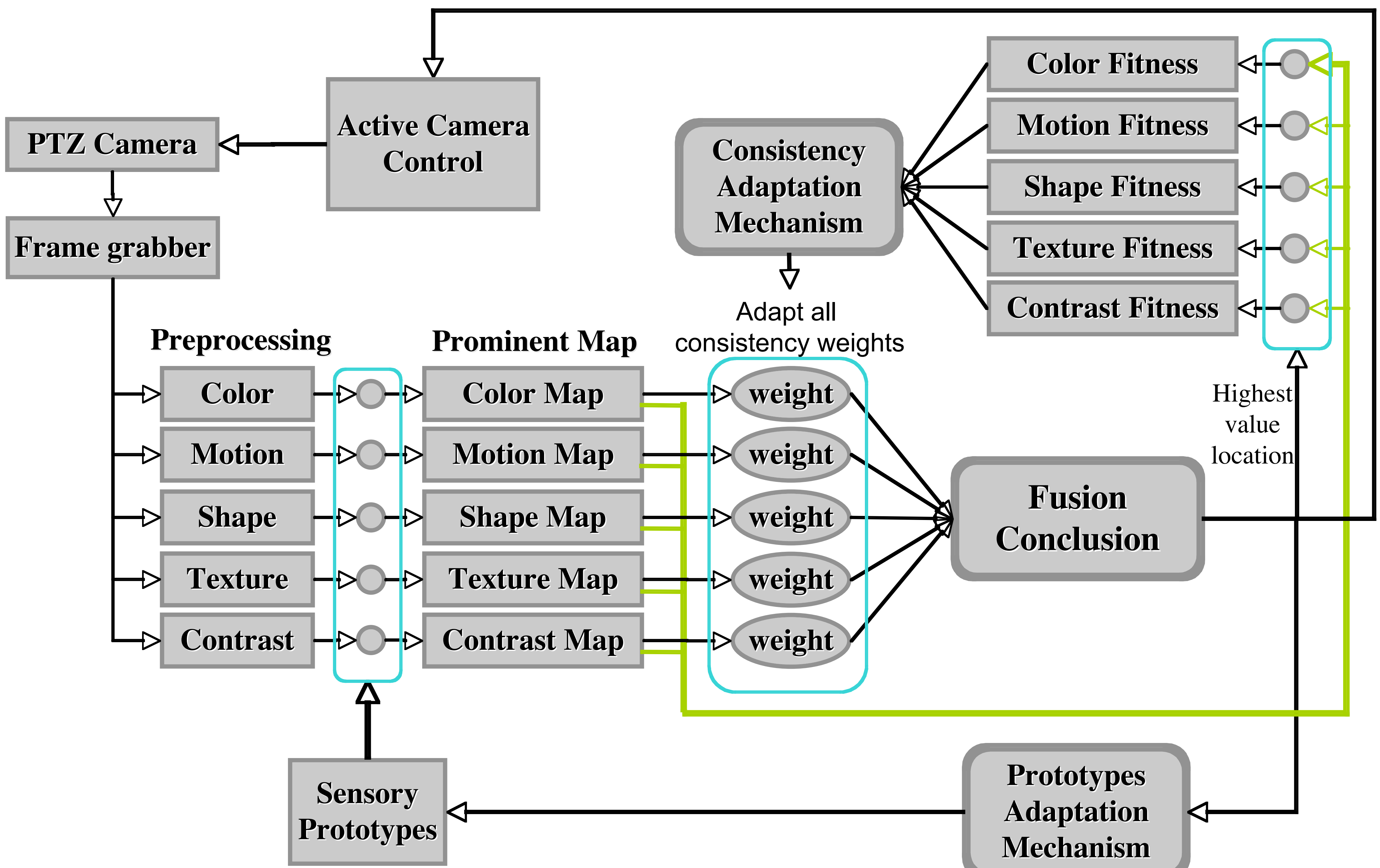
- Class of Tasks: Learning to drive on highways from vision stereos.
- Knowledge: Images and steering commands recorded while observing a human driver.
- Performance Module: Accuracy in classification

Automatic Car Driver

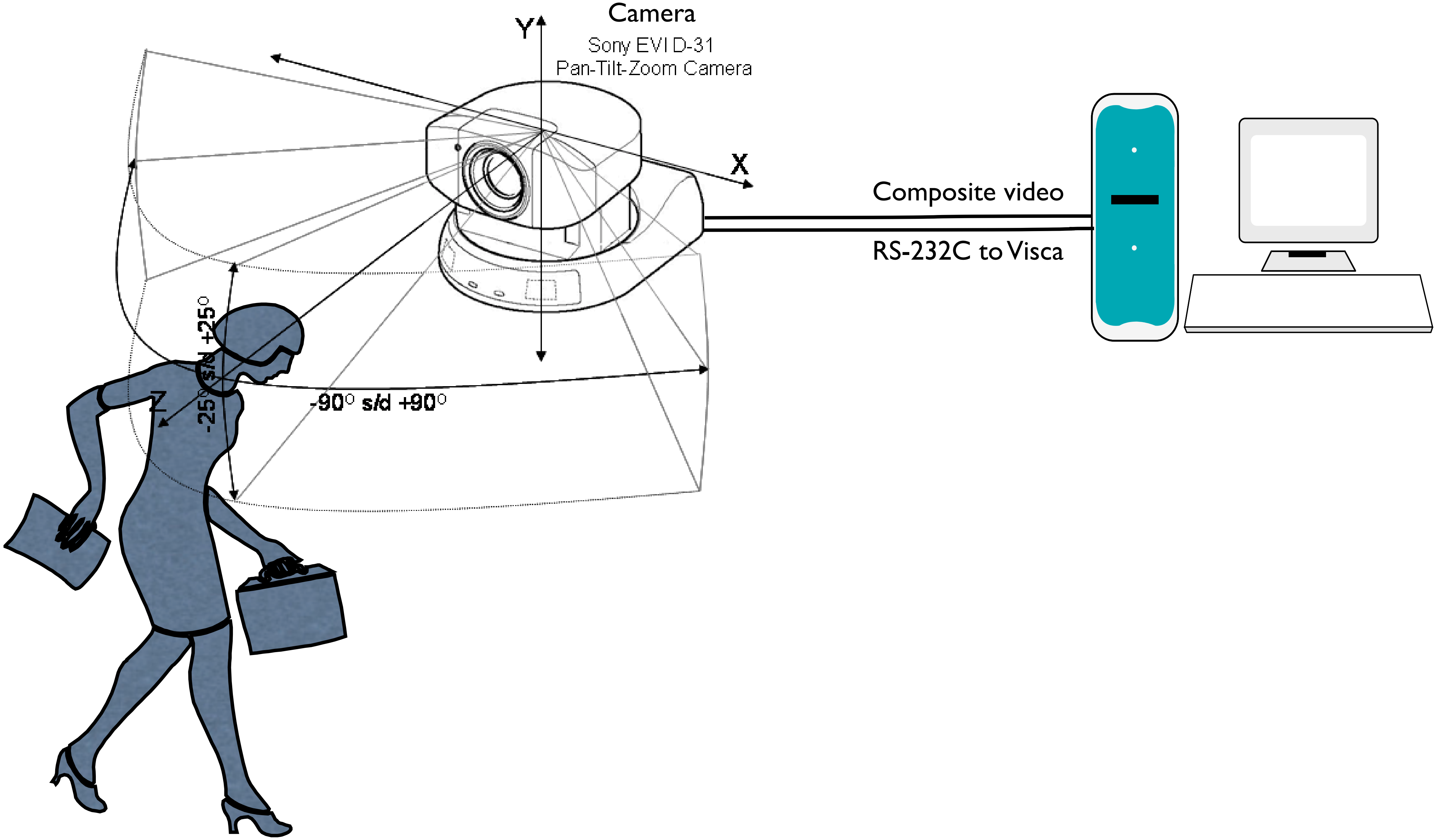




Learning Mechanism for Fusion of Image Information



Sensory Fusion applied to Face Detection and Tracking using Active Camera



A Result of Face Detection and Tracking

Summary

- Machine learning is the study of how to make computers learn.
- A learning algorithm needs the following elements: class of tasks, performance metric, and body of experience.
- The design of a learning algorithm requires to define the knowledge to learn, the representation of the target knowledge, and the learning mechanism.
- Machine learning counts with many successful applications and is becoming increasingly important in science and industry.

Unsupervised Clustering:
K-means and Mixtures of Gaussians

K-means Algorithm

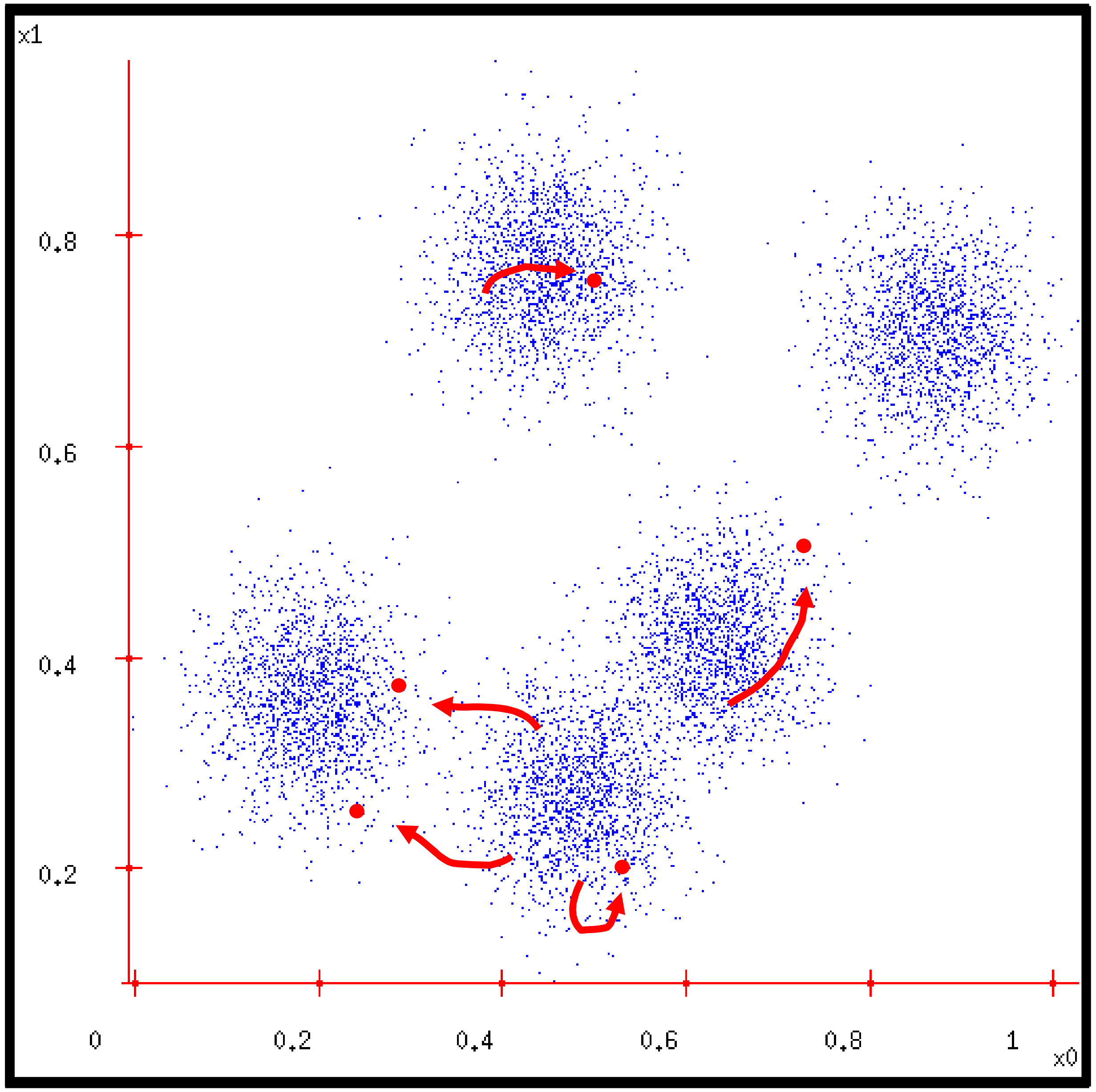
- Given data <x1 ... xn>, and K, assign each xi to one of K clusters, C1 ... Ck, minimizing

$$J = \sum_{j=1}^{K} \sum_{x_i \in C_j} ||x_i - \mu_j||^2$$

- where μj is mean over all points in cluster Cj
- K-Means Algorithm:
  - Initialize randomly
  - Repeat until convergence:
    - Assign each point xi to the cluster with the closest mean μj
    - Calculate the new mean for each cluster
    - $$\mu_j \leftarrow \frac{1}{|C_j|} \sum_{x_i \in C_j} x_i$$

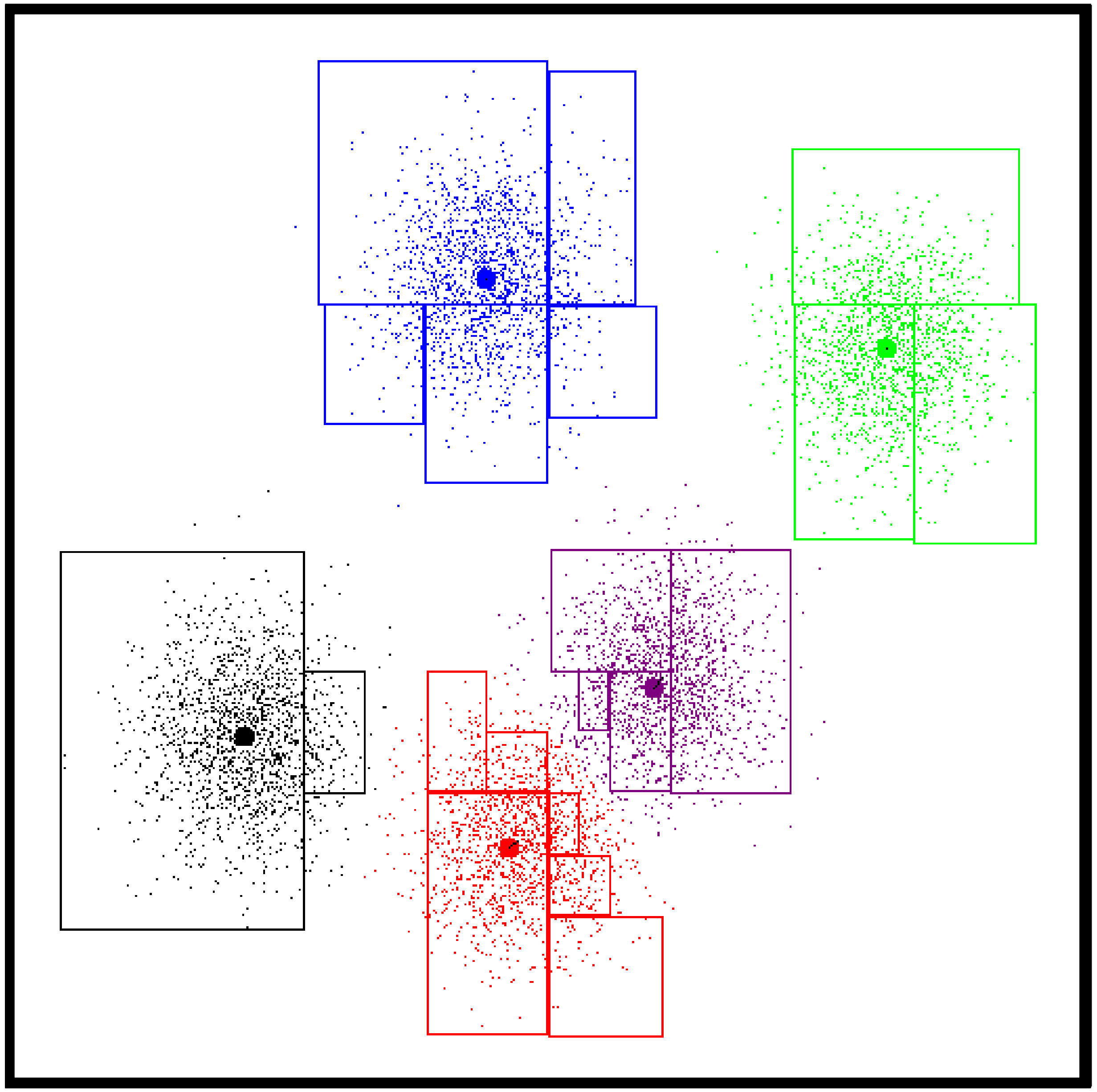
Example of K-means Clustering

- Ask user how many clusters they'd like. (e.g. k=5)
- Randomly guess k cluster Center locations
- Each datapoint finds out which Center it's closest to.
- Each Center finds the centroid of the points it owns...
- ...and jumps there
- ...Repeat until terminated!



Example of K-Means Clustering

- Advance apologies: in Black and White this example will deteriorate
- Example generated by Dan Pelleg's super-duper fast K-means system:
  - Dan Pelleg and Andrew Moore. Accelerating Exact k-means Algorithms with Geometric Reasoning.
  - Proc. Conference on Knowledge Discovery in Databases 1999,
- K-means continues … (9x)
- K-means terminates



Mixture of Gaussians

- Given observed variables X, unobserved Z
- Define $Q(\theta'|\theta) = E_{Z|X,\theta}[\log P(X, Z|\theta')]$
- where $\theta = \langle \pi, \mu_{ji} \rangle$

- Iterate until convergence:
  - E Step: For each observed example X(n), calculate P(Z(n)|X(n),θ)

$$P(z(n) = k \mid x(n), \theta) = \frac{[\prod_i N(x_i(n)|\mu_{k,i}, \sigma)] \quad (\pi^k (1-\pi)^{(1-k)})}{\sum_{j=0}^{1} [\prod_i N(x_i(n)|\mu_{j,i}, \sigma)] \quad (\pi^j (1-\pi)^{(1-j)})}$$

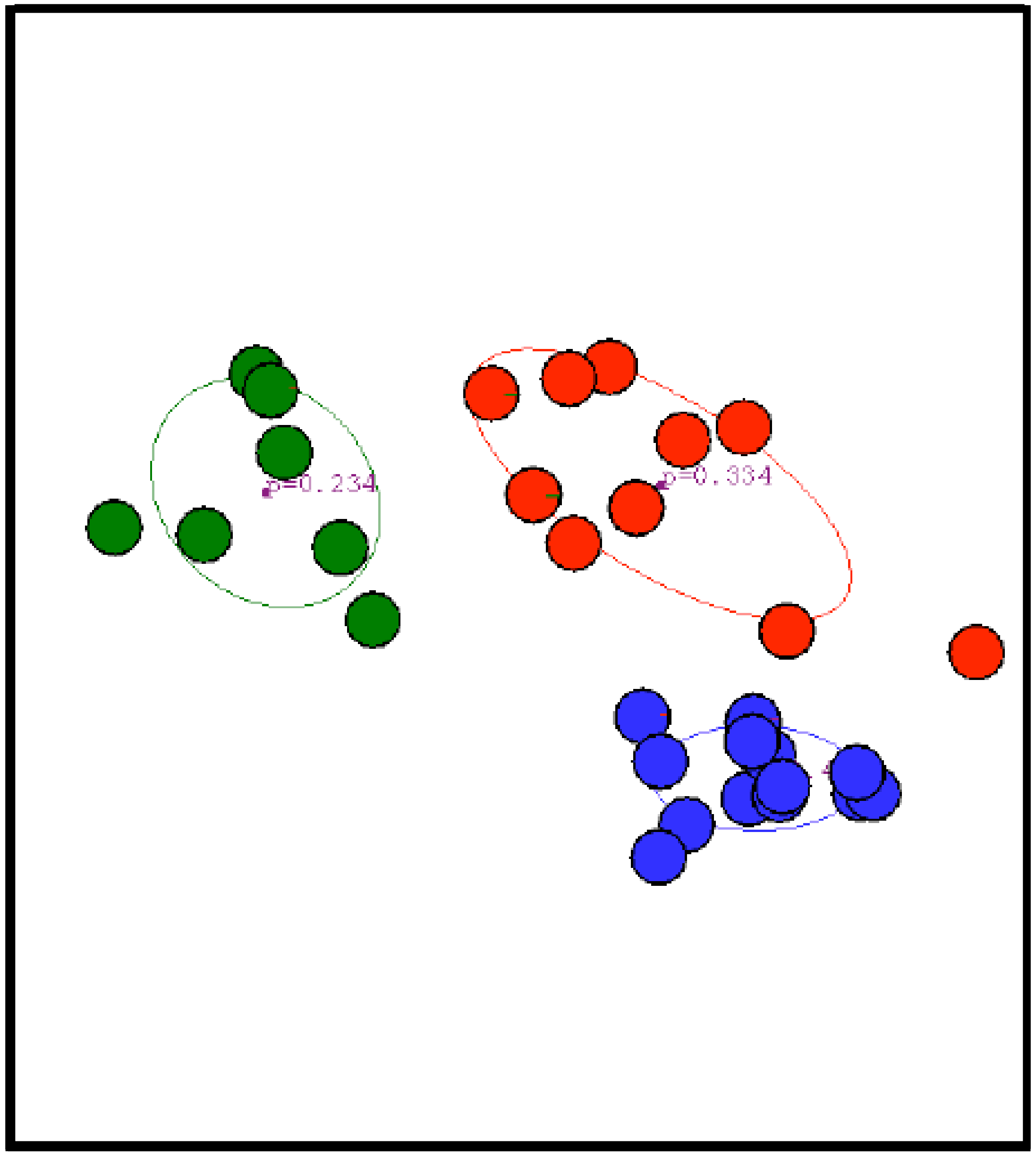
  - M Step: Update $\theta \leftarrow \arg\max_{\theta'} Q(\theta'|\theta)$

$$\pi \leftarrow \frac{1}{N} \sum_{n=1}^{N} E[z(n)] \qquad \mu_{ji} \leftarrow \frac{\sum_{n=1}^{N} P(z(n) = j|x(n), \theta) \quad x_i(n)}{\sum_{n=1}^{N} P(z(n) = j|x(n), \theta)}$$

Gaussian Mixture Example

- Gaussian Mixture Example: Start
- After first iteration
- After 2nd iteration
- After 3th iteration
- After 4th iteration
- After 5th iteration
- After 6th iteration

- After 20th iteration



K-Means vs Mixture of Gaussians

- Both are iterative algorithms to assign points to clusters
- Objective function
  - K Means: minimize $J = \sum_{j=1}^{K} \sum_{x_i \in C_j} ||x_i - \mu_j||^2$
  - MixGaussians: maximize $P(X|\theta)$
- Mixture of Gaussians is the more general formulation
  - Equivalent to K Means when $\Sigma k = \sigma$ I, and $\sigma \to 0$