IST 511 Information Management: Information and Technology

Machine Learning

Dr. C. Lee Giles

David Reese Professor, College of Information Sciences and Technology

The Pennsylvania State University

University Park, PA, USA

giles@ist.psu.edu

http://clgiles.ist.psu.edu

Special thanks to F. Hoffmann, T. Mitchell, D. Miller, H. Foundalis, R Mooney

Last time

- Web as a graph
- What is link analysis
  - Definitions
  - Why important
  - How are links used – ranking
- IR vs search engines
  - How are search engines related to information retrieval?
  - How is information gathered
- Impact and importance of search engines
- Impact on information science

Today

- Introduction to machine learning (ML)
  – Definitions/theory
  – Why important
  – How is ML used
- What is learning
  – Relation to animal/human learning
- Impact on information science

Tomorrow

- Topics used in IST
- Probabilistic reasoning
- Digital libraries
- Others?

Theories in Information Sciences

- Issues:
  - Unified theory? Maybe AI
  - Domain of applicability – interactions with the real world
  - Conflicts – ML versus human learning
- Theories here are
  - Mostly algorithmic
  - Some quantitative
- Quality of theories
  - Occam's razor – simplest ML method
  - Subsumption of other theories (AI vs ML)
  - ML very very popular in real world applications
    - ML can be used in nearly every topic involving data that we discuss
- Theories of ML
  - Cognitive vs computational
  - Mathematical

What is Machine Learning?

Aspect of AI: creates knowledge

Definition:

“changes in [a] system that ... enable [it] to do the same task or tasks drawn from the same population more efficiently and more effectively the next time.'' (Simon 1983)

There are two ways that a system can improve:

1. By acquiring new knowledge
   - acquiring new facts
   - acquiring new skills
2. By adapting its behavior
   - solving problems more accurately
   - solving problems more efficiently

What is Learning?

- Herbert Simon: “Learning is any process by which a system improves performance from experience.”
- What is the task?
  - Classification
  - Categorization/clustering
  - Problem solving / planning / control
  - Prediction
  - others

Why Study Machine Learning? Developing Better Computing Systems

- Develop systems that are too difficult/expensive to construct manually because they require specific detailed skills or knowledge tuned to a specific task (***knowledge engineering bottleneck***).
- Develop systems that can automatically adapt and customize themselves to individual users.
  - Personalized news or mail filter
  - Personalized tutoring
- Discover new knowledge from large databases (***data mining***).
  - Market basket analysis (e.g. diapers and beer)
  - Medical text mining (e.g. migraines to calcium channel blockers to magnesium)

Related Disciplines

- Artificial Intelligence
- Data Mining
- Probability and Statistics
- Information theory
- Numerical optimization
- Computational complexity theory
- Control theory (adaptive)
- Psychology (developmental, cognitive)
- Neurobiology
- Linguistics
- Philosophy

Human vs machine learning

- Cognitive science vs computational science
  - Animal learning vs machine learning
    - Don’t fly like birds
  - Many ML models are based on human types of learning
- Evolution vs machine learning
  - Adaptation vs learning

Adaptive vs machine learning

- An adaptive system is a set of interacting or interdependent entities, real or abstract, forming an integrated whole that together are able to respond to environmental changes or changes in the interacting parts. Feedback loops represent a key feature of adaptive systems, allowing the response to changes; examples of adaptive systems include: natural ecosystems, individual organisms, human communities, human organizations, and human families.
- Some artificial systems can be adaptive as well; for instance, robots employ control systems that utilize feedback loops to sense new conditions in their environment and adapt accordingly.

Types of Learning

- Induction vs deduction
- Rote learning (memorization)
- Advice or instructional learning
- *Learning by example or practice*
  - *Most popular; many applications*
- Learning by analogy; transfer learning
- Discovery learning
- Others?

Levels of Learning
Training

Many learning methods involve training

- Training is the acquisition of knowledge, skills, and competencies as a result of the teaching of vocational or practical skills and knowledge that relate to specific useful competencies (wikipedia).
- Training requires scenarios or examples (data)

Types of training experience

- Direct or indirect
- With a teacher or without a teacher
- An eternal problem:
  - Make the training experience representative of the performance goal

Types of training

- Supervised learning: uses a series of labelled examples with direct feedback
- Reinforcement learning: indirect feedback, after many examples
- Unsupervised/clustering learning: no feedback
- Semisupervised

Types of testing

- Evaluate performance by testing on data NOT used for testing (both should be randomly sampled)
- Cross validation methods for small data sets
- The more (relevant) data the better.

Testing

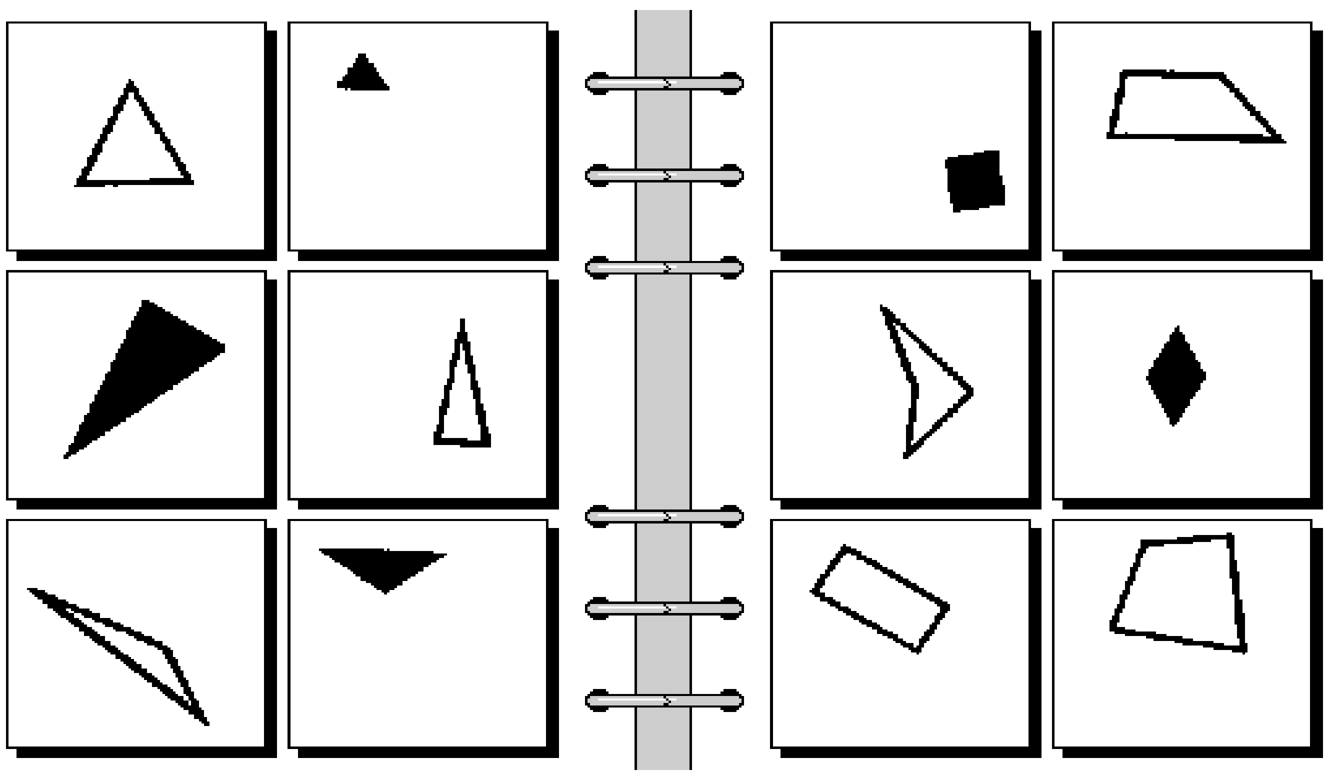
- How well the learned system work?
- Generalization
  - Performance on unseen or unknown scenarios or data
  - Brittle vs robust performance

Which of these things is NOT like the others?

Which of these things is like the others? And how?

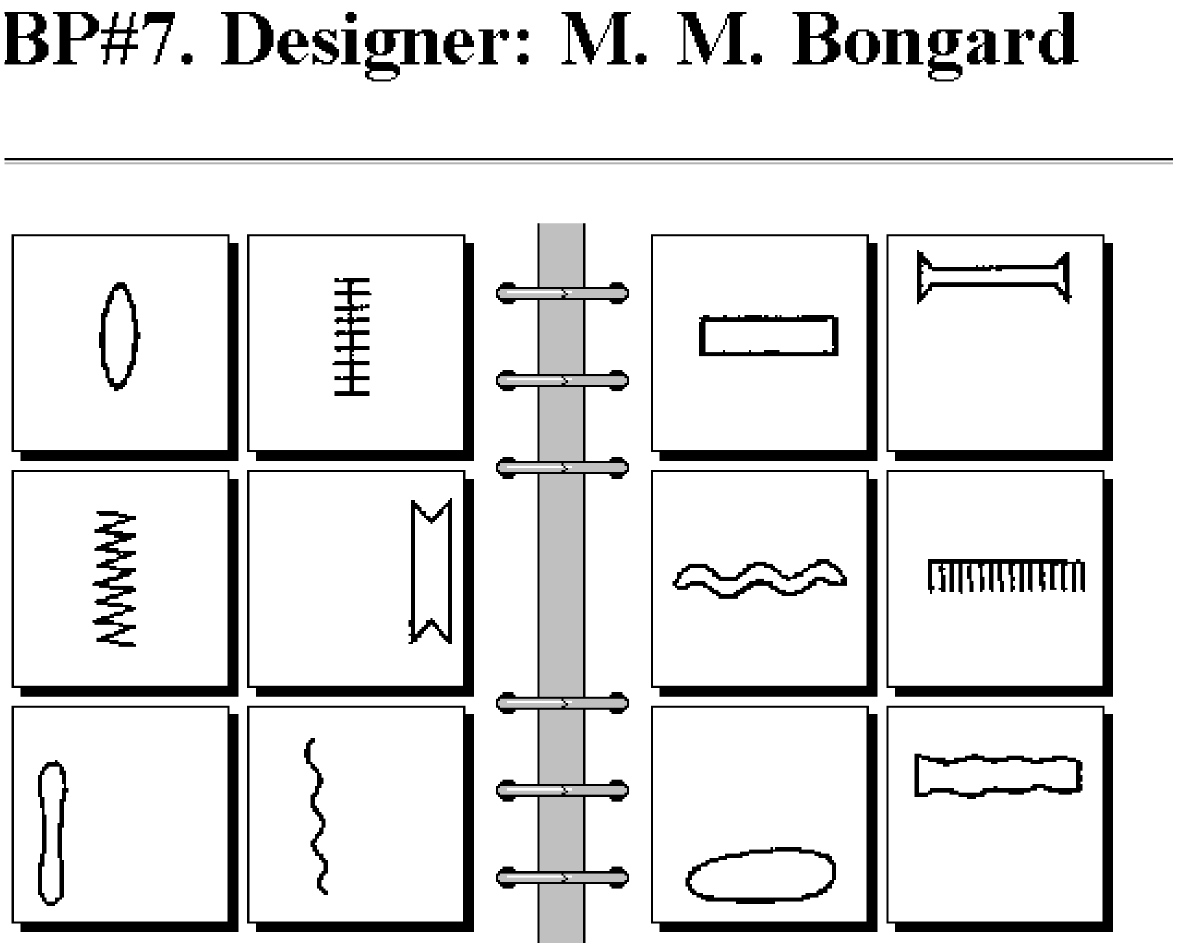
Harry Foundalis - Research on the Bongard Problems

The domain of my research in cognitive science is the *Bongard Problems*. These are problems on *visual pattern recognition* that appeared first in the appendix of a book published by the Russian scientist M. M. Bongard in 1967, in what was then the USSR. They became more widely known to the western world when D. R. Hofstadter mentioned them in his book, "*Goedel, Escher, Bach: an Eternal Golden Braid*", and speculated on how an automated system could be built to solve such problems. Rather than tiring the reader with words, I prefer to show what Bongard Problems (BP's) are by presenting a — rather trivial — BP, below.



Bongard problems

- visual pattern rule induction



Index of Bongard Problems

Usual ML stages

- Hypothesis, data
- Training or learning
- Testing or generalization

Why is machine learning necessary?

- *learning is a hallmark of intelligence*; many would argue that a system that cannot learn is not intelligent.
- without learning, everything is new; a system that cannot learn is not efficient because it rederives each solution and repeatedly makes the same mistakes.

Why is learning possible?

Because there are regularities in the world.

Different Varieties of Machine Learning

- Concept Learning
- Clustering Algorithms
- Connectionist Algorithms
- Genetic Algorithms
- Explanation-based Learning
- Transformation-based Learning
- Reinforcement Learning
- Case-based Learning
- Macro Learning
- Evaluation Functions
- Cognitive Learning Architectures
- Constructive Induction
- Discovery Systems
- Knowledge capture

Many online software packages & datasets

- Data sets
  - UC Irvine
  - http://www.kdnuggets.com/datasets/index.html
- Software (much related to data mining)
  - JMIR Open Source
  - Weka
  - Shogun
  - RapidMiner
  - ODM
  - Orange
  - CMU
  - Several researchers put their software online

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

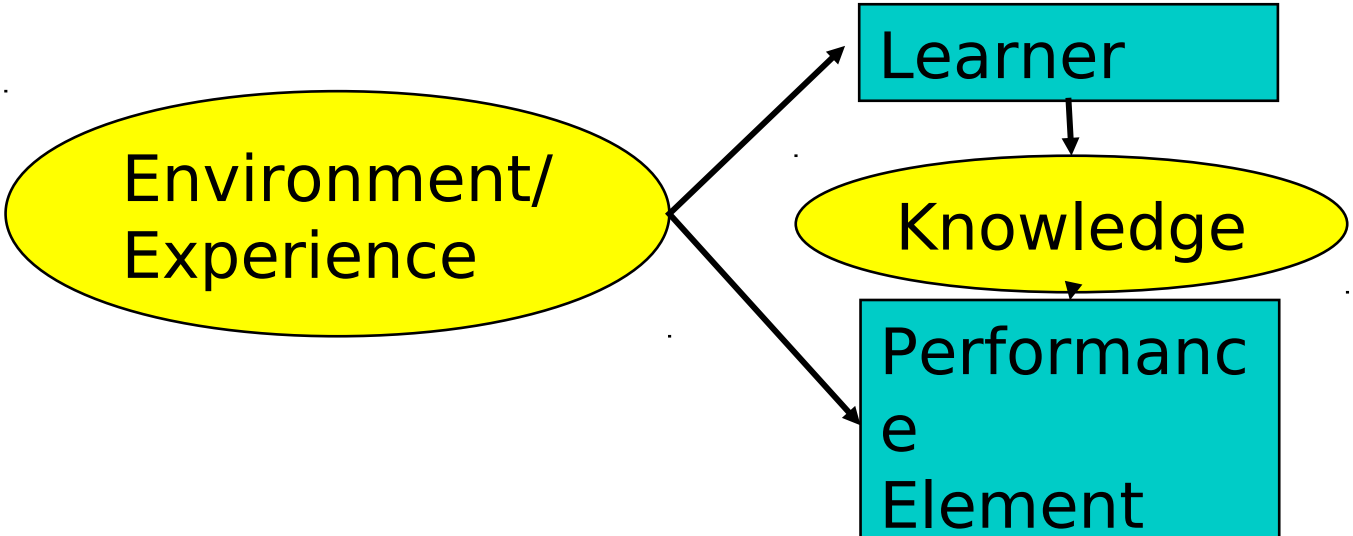
T: Recognizing hand-written words
P: Percentage of words correctly classified
E: Database of human-labeled images of handwritten words

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

Designing a Learning System

- Choose the training experience
- Choose exactly what is too 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.



Sample Learning Problem

- Learn to play checkers from self-play
- Develop an approach analogous to that used in the first machine learning system developed by Arthur Samuels at IBM in 1959.

Training Experience

- Direct experience: Given sample input and output pairs for a useful target function.
  - Checker boards labeled with the correct move, e.g. extracted from record of expert play
- Indirect experience: Given feedback which is ***not*** direct I/O pairs for a useful target function.
  - Potentially arbitrary sequences of game moves and their final game results.
- Credit/Blame Assignment Problem: How to assign credit blame to individual moves given only indirect feedback?

Source of Training Data

- Provided random examples outside of the learner's control.
  - Negative examples available or only positive?
- Good training examples selected by a "benevolent teacher."
  - "Near miss" examples
- Learner can query an oracle about class of an unlabeled example in the environment.
- Learner can construct an arbitrary example and query an oracle for its label.
- Learner can design and run experiments directly in the environment without any human guidance.

Training vs. Test Distribution

- Generally assume that the training and test examples are independently drawn from the same overall distribution of data.
  - IID: Independently and identically distributed
- If examples are not independent, requires ***collective classification***.
- If test distribution is different, requires ***transfer learning***.

Choosing a Target Function

- What function is to be learned and how will it be used by the performance system?
- For checkers, assume we are given a function for generating the legal moves for a given board position and want to decide the best move.
  - Could learn a function:

    ChooseMove(board, legal-moves) → best-move
  - Or could learn an ***evaluation function***, *V*(board) → R, that gives each board position a score for how favorable it is. *V* can be used to pick a move by applying each legal move, scoring the resulting board position, and choosing the move that results in the highest scoring board position.

Ideal Definition of $V(b)$

- If b is a final winning board, then $V(b) = 100$
- If b is a final losing board, then $V(b) = -100$
- If b is a final draw board, then $V(b) = 0$
- Otherwise, then $V(b) = V(b')$, where b' is the highest scoring final board position that is achieved starting from b and playing optimally until the end of the game (assuming the opponent plays optimally as well).
  - Can be computed using complete mini-max search of the finite game tree.

Approximating *V*(*b*)

- Computing *V*(*b*) is intractable since it involves searching the complete exponential game tree.
- Therefore, this definition is said to be ***non-operational***.
- An ***operational*** definition can be computed in reasonable (polynomial) time.
- Need to learn an operational *approximation* to the ideal evaluation function.

Representing the Target Function

- Target function can be represented in many ways: lookup table, symbolic rules, numerical function, neural network.
- There is a trade-off between the expressiveness of a representation and the ease of learning.
- The more expressive a representation, the better it will be at approximating an arbitrary function; however, the more examples will be needed to learn an accurate function.

Linear Function for Representing *V*(*b*)

- In checkers, use a linear approximation of the evaluation function.

$$\hat{V}(b) = w_0 + w_1 \cdot bp(b) + w_2 \cdot rp(b) + w_3 \cdot bk(b) + w_4 \cdot rk(b) + w_5 \cdot bt(b) + w_6 \cdot rt(b)$$

  - *bp*(*b*): number of black pieces on board b
  - *rp*(*b*): number of red pieces on board b
  - *bk*(*b*): number of black kings on board b
  - *rk*(*b*): number of red kings on board b
  - *bt*(*b*): number of black pieces threatened (i.e. which can be immediately taken by red on its next turn)
  - *rt*(*b*): number of red pieces threatened

Obtaining Training Values

- Direct supervision may be available for the target function.
  - < <*bp*=3,*rp*=0,*bk*=1,*rk*=0,*bt*=0,*rt*=0>, 100> (win for black)
- With indirect feedback, training values can be estimated using ***temporal difference learning*** (used in ***reinforcement learning*** where supervision is ***delayed reward***).

Temporal Difference Learning

- Estimate training values for intermediate (non-terminal) board positions by the estimated value of their successor in an actual game trace.

$$V_{train}(b) = \hat{V}(\text{successor}(b))$$

  where successor(b) is the next board position where it is the program's move in actual play.
- Values towards the end of the game are initially more accurate and continued training slowly "backs up" accurate values to earlier board positions.

Learning Algorithm

- Uses training values for the target function to induce a hypothesized definition that fits these examples and hopefully generalizes to unseen examples.
- In statistics, learning to approximate a continuous function is called ***regression***.
- Attempts to minimize some measure of error (***loss function***) such as ***mean squared error***:

$$E = \frac{\sum_{b \in B} [V_{train}(b) - \hat{V}(b)]^2}{|B|}$$

Least Mean Squares (LMS) Algorithm

- A gradient descent algorithm that incrementally updates the weights of a linear function in an attempt to minimize the mean squared error

Until weights converge :

For each training example *b* do :

1) Compute the absolute error :

$$error(b) = V_{train}(b) - \hat{V}(b)$$

2) For each board feature, f_i, update its weight, w_i :

$$w_i = w_i + c \cdot f_i \cdot error(b)$$

for some small constant (learning rate) c

LMS Discussion

- Intuitively, LMS executes the following rules:
  - If the output for an example is correct, make no change.
  - If the output is too high, lower the weights proportional to the values of their corresponding features, so the overall output decreases
  - If the output is too low, increase the weights proportional to the values of their corresponding features, so the overall output increases.
- Under the proper weak assumptions, LMS can be proven to eventetually converge to a set of weights that minimizes the mean squared error.

Lessons Learned about Learning

- Learning can be viewed as using direct or indirect experience to approximate a chosen target function.
- Function approximation can be viewed as a search through a space of hypotheses (representations of functions) for one that best fits a set of training data.
- Different learning methods assume different hypothesis spaces (representation languages) and/or employ different search techniques.

Various Function Representations

- Numerical functions
  - Linear regression
  - Neural networks
  - Support vector machines
- Symbolic functions
  - Decision trees
  - Rules in propositional logic
  - Rules in first-order predicate logic
- Instance-based functions
  - Nearest-neighbor
  - Case-based
- Probabilistic Graphical Models
  - Naïve Bayes
  - Bayesian networks
  - Hidden-Markov Models (HMMs)
  - Probabilistic Context Free Grammars (PCFGs)
  - Markov networks

Various Search Algorithms

- Gradient descent
  - Perceptron
  - Backpropagation
- Dynamic Programming
  - HMM Learning
  - PCFG Learning
- Divide and Conquer
  - Decision tree induction
  - Rule learning
- Evolutionary Computation
  - Genetic Algorithms (GAs)
  - Genetic Programming (GP)
  - Neuro-evolution

Evaluation of Learning Systems

- Experimental
  - Conduct controlled cross-validation experiments to compare various methods on a variety of benchmark datasets.
  - Gather data on their performance, e.g. test accuracy, training-time, testing-time.
  - Analyze differences for statistical significance.
- Theoretical
  - Analyze algorithms mathematically and prove theorems about their:
    - Computational complexity
    - Ability to fit training data
    - Sample complexity (number of training examples needed to learn an accurate function)

History of Machine Learning

- 1950s
  - Samuel’s checker player
  - Selfridge’s Pandemonium
- 1960s:
  - Neural networks: Perceptron
  - Pattern recognition
  - Learning in the limit theory
  - Minsky and Papert prove limitations of Perceptron
- 1970s:
  - Symbolic concept induction
  - Winston’s arch learner
  - Expert systems and the knowledge acquisition bottleneck
  - Quinlan’s ID3
  - Michalski’s AQ and soybean diagnosis
  - Scientific discovery with BACON
  - Mathematical discovery with AM

History of Machine Learning (cont.)

- 1980s:
  - Advanced decision tree and rule learning
  - Explanation-based Learning (EBL)
  - Learning and planning and problem solving
  - Utility problem
  - Analogy
  - Cognitive architectures
  - Resurgence of neural networks (connectionism, backpropagation)
  - Valiant’s PAC Learning Theory
  - Focus on experimental methodology
- 1990s
  - Data mining
  - Adaptive software agents and web applications
  - Text learning
  - Reinforcement learning (RL)
  - Inductive Logic Programming (ILP)
  - Ensembles: Bagging, Boosting, and Stacking
  - Bayes Net learning

History of Machine Learning (cont.)

- 2000s
  - Support vector machines
  - Kernel methods
  - Graphical models
  - Statistical relational learning
  - Transfer learning
  - Sequence labeling
  - Collective classification and structured outputs
  - Computer Systems Applications
    - Compilers
    - Debugging
    - Graphics
    - Security (intrusion, virus, and worm detection)
  - Email management
  - Personalized assistants that learn
  - Learning in robotics and vision

http://www.kdnuggets.com/datasets/index.html

Datasets for Data Mining

- See first **UCI KDD Database Repository** -- the most popular site for datasets used for research in machine learning and knowledge discovery.
- Delve, Data for Evaluating Learning in Valid Experiments
- FEDSTATS, a comprehensive source of US statistics and more
- Financial Data Finder at OSU, a large catalog of financial data sets
- Grain Market Research, financial data including stocks, futures, etc.
- Investor Links, includes financial data
- Microsoft's TerraServer, aerial photographs and satellite images you can view and purchase.
- MIT Cancer Genomics gene expression datasets and publications, from MIT Whitehead Center for Genome Research
- MLnet (European Machine Learning Network) list of Datasets
- National Space Science Data Center (NSSDC), NASA data sets from planetary exploration, space and solar physics, life sciences, astrophysics, and more.
- Neural Networks Benchmarking homepage, from NIPS*95 workshop.
- PubGene(TM) Gene Database and Tools, genomic-related publications database
- SMD: Stanford Microarray Database, stores raw and normalized data from microarray experiments.
- STATLOG project datasets. This project did comparative studies of different machine learning, neural and statistical classification algorithms. About 20 different algorithms were evaluated on more than 20 different datasets.
- STATOO Datasets part 1 and part 2
- United States Census Bureau.

Supervised Learning Classification

- Example: Cancer diagnosis

Patient ID	# of Tumors	Avg Area	Avg Density	Diagnosis
1	5	20	118	Malignant
2	3	15	130	Benign
3	7	10	52	Benign
4	2	30	100	Malignant

Training Set

- Use this training set to learn how to classify patients where diagnosis is not known:

Patient ID	# of Tumors	Avg Area	Avg Density	Diagnosis
101	4	16	95	?
102	9	22	125	?
103	1	14	80	?

Test Set

Input Data Classification

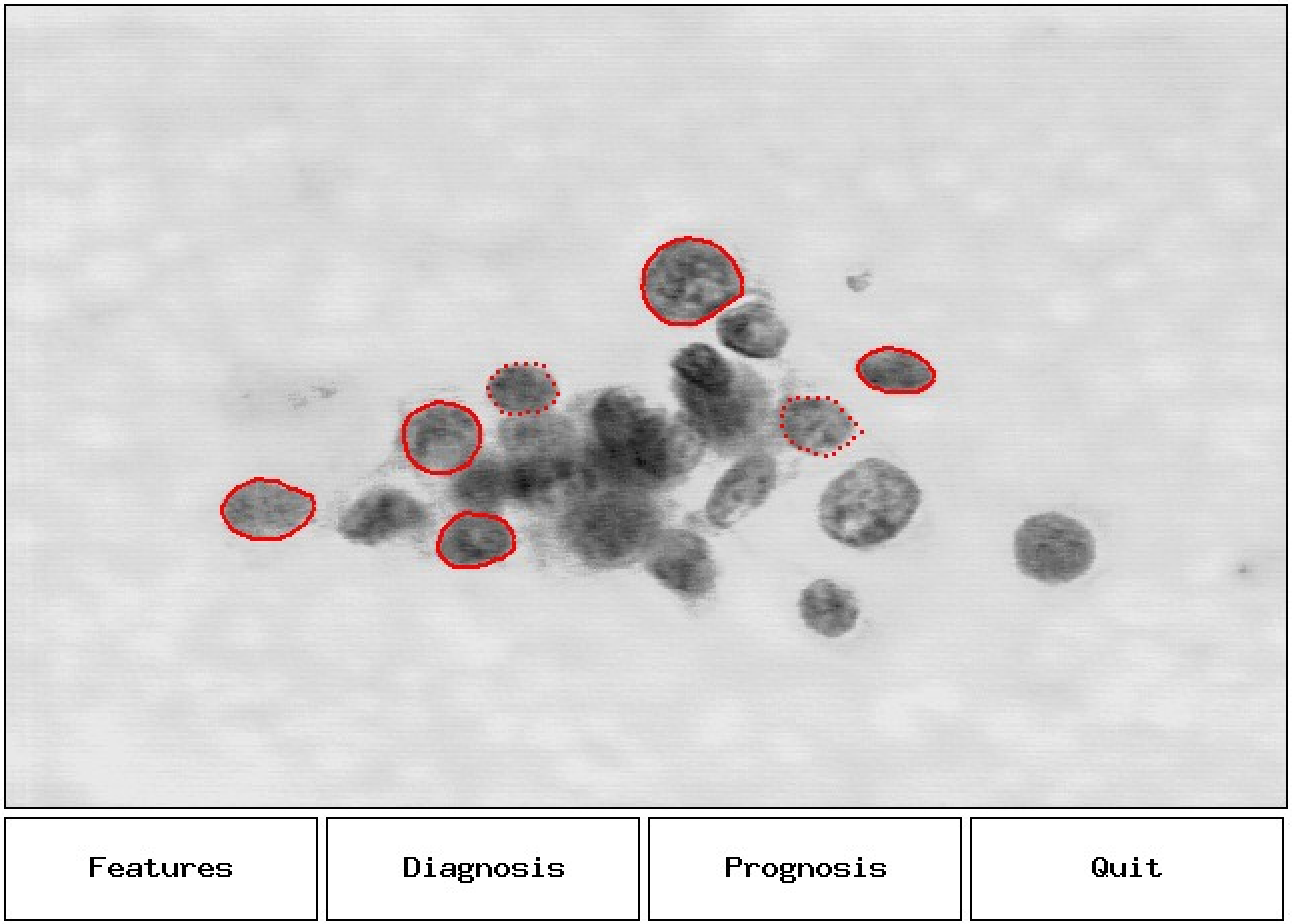
- The input data is often easily obtained, whereas the classification is not.

Classification Problem

- Goal: Use training set + some learning method to produce a predictive model.
- Use this predictive model to classify new data.
- Sample applications:

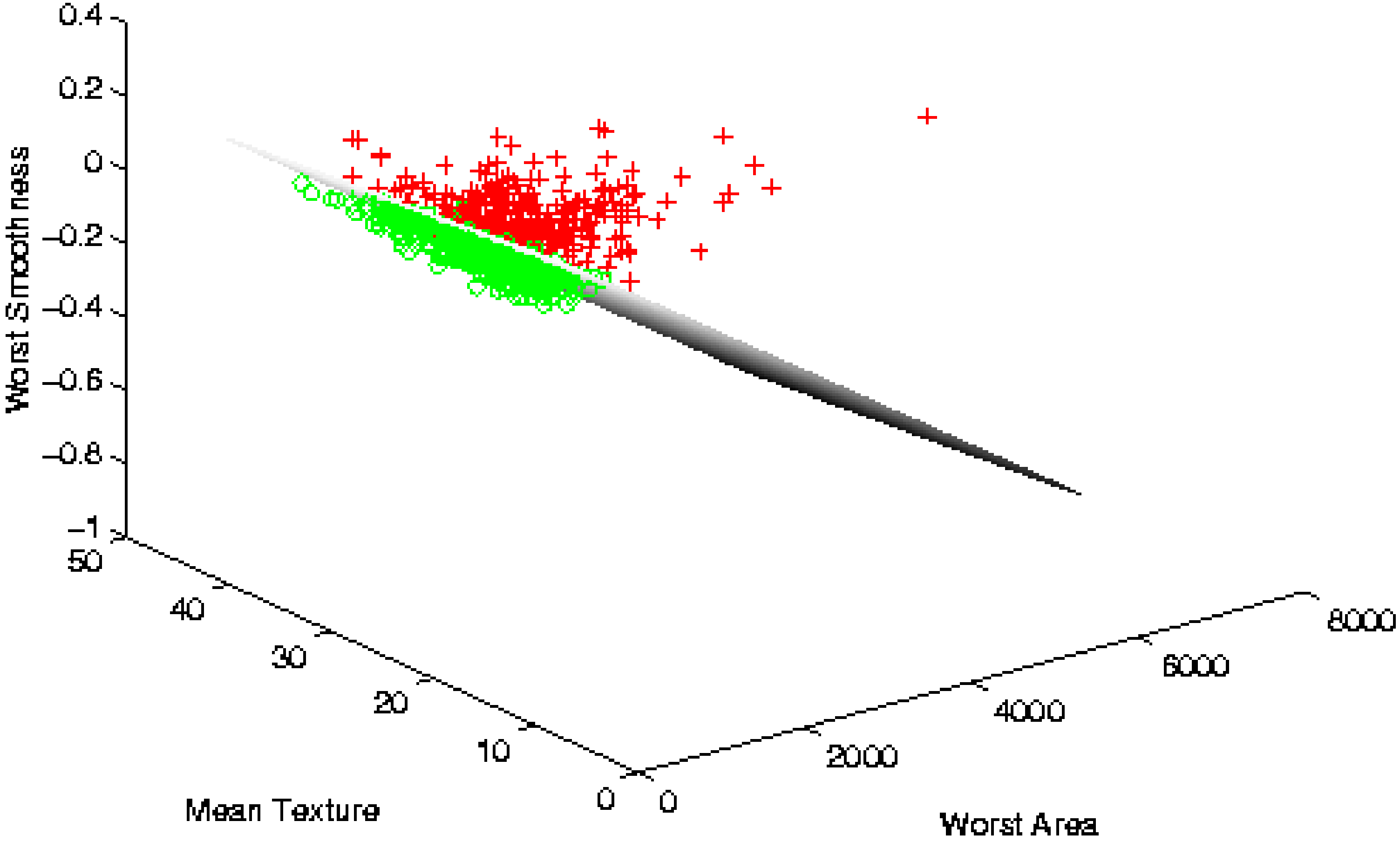
Application	Input Data	Classification
Medical Diagnosis	Noninvasive tests	Results from invasive measurements
Optical Character Recognition	Scanned bitmaps	Letter A-Z
Protein Folding	Amino acid construction	Protein shape (helices, loops, sheets)
Research Paper Acceptance	Words in paper title	Paper accepted or rejected

Application: Breast Cancer Diagnosis



Research by Mangasarian,Street, Wolberg

Breast Cancer Diagnosis Separation

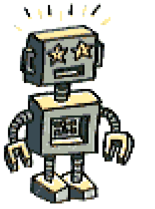


Research by Mangasarian,Street, Wolberg

The revolution in robotics

- Cheap robots!!!
- Cheap sensors
- Moore’s law

Robotics and ML



Areas that robots are used:

- Industrial robots
- Military, government and space robots
- Service robots for home, healthcare, laboratory

Why are robots used?

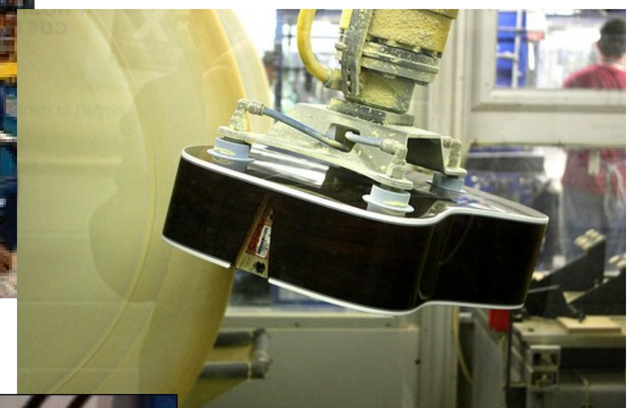
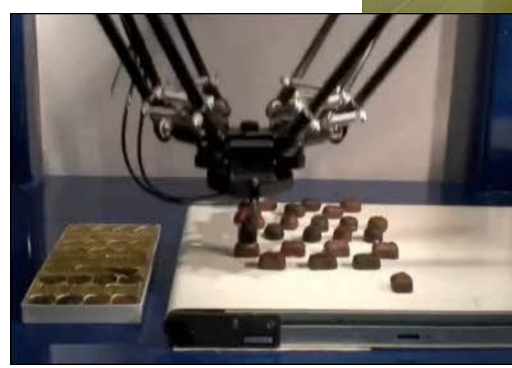
- Dangerous tasks or in hazardous environments
- Repetitive tasks
- High precision tasks or those requiring high quality
- Labor savings

Control technologies:

- Autonomous (self-controlled), tele-operated (remote control)

Industrial Robots

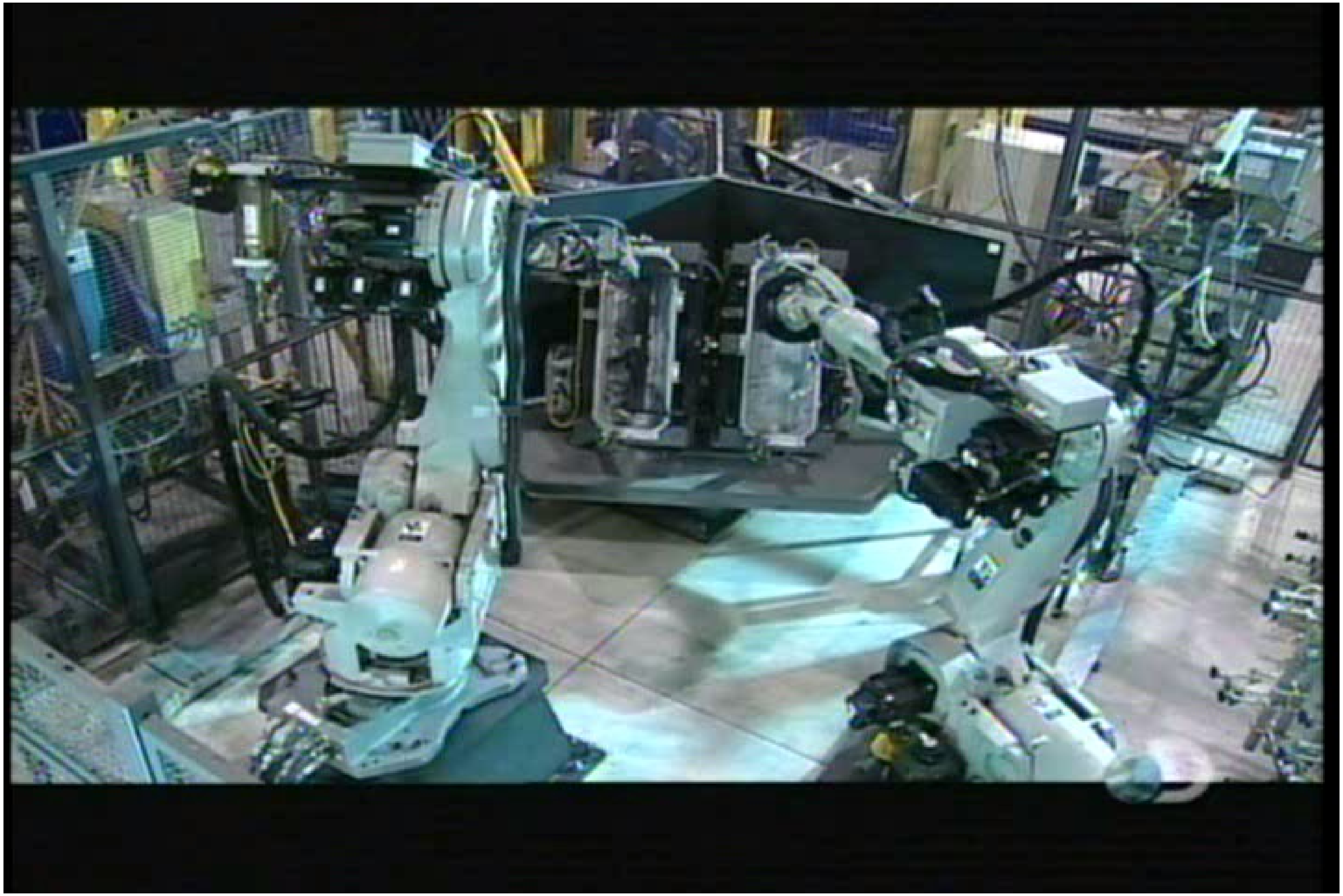
- Uses for robots in manufacturing:
  - Welding
  - Painting
  - Cutting
  - Dispensing
  - Assembly
  - Polishing/Finishing
  - Material Handling
    - Packaging, Palletizing
    - Machine loading


Industrial Robots

- Uses for robots in Industry/Manufacturing
  - Automotive:
    - Video - Welding and handling of fuel tanks from TV show “How It’s Made” on Discovery Channel. This is a system I worked on in 2003.
  - Packaging:
    - Video - Robots in food manufacturing.

Industrial Robots - Automotive



Military/Government Robots

- iRobot

Remotec Andros

Military/Government Robots

Soldiers in Afghanistan being trained how to defuse a landmine using a PackBot.

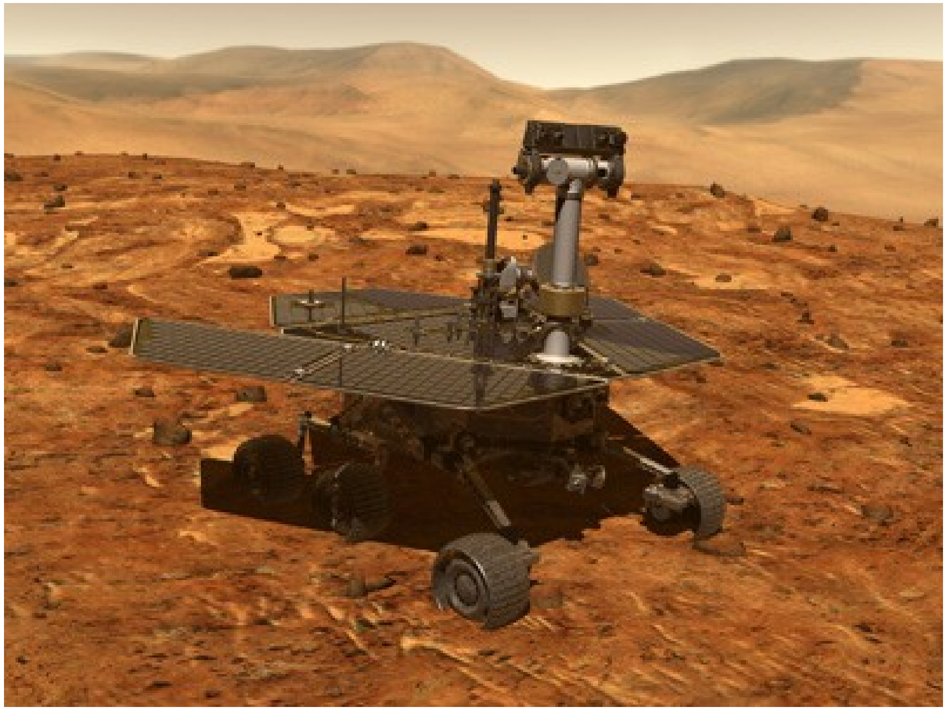
Military Robots

- Aerial drones (UAV)

Military suit

Space Robots

- Mars Rovers – Spirit and Opportunity
  - Autonomous navigation features with human remote control and oversight



Service Robots

- Many uses…
  - Cleaning & Housekeeping
  - Humanitarian Demining
  - Rehabilitation
  - Inspection
  - Agriculture & Harvesting
  - Lawn Mowers
  - Surveillance
  - Mining Applications
  - Construction
  - Automatic Refilling
  - Fire Fighters
  - Search & Rescue

iRobot Roomba vacuum cleaner robot

Medical/Healthcare Applications

DaVinci surgical robot by Intuitive Surgical.

St. Elizabeth Hospital is one of the local hospitals using this robot. You can see this robot in person during an open house (website).

Japanese health care assistant suit (HAL - Hybrid Assistive Limb)

Also… Mind-controlled wheelchair using NI LabVIEW

Laboratory Applications

Drug discovery

Test tube sorting

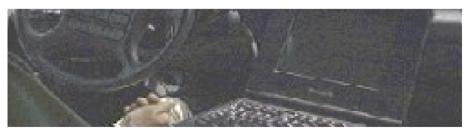
ALVINN

[redacted]

Drives 70 mph on a public highway

Predecessor of the Google car

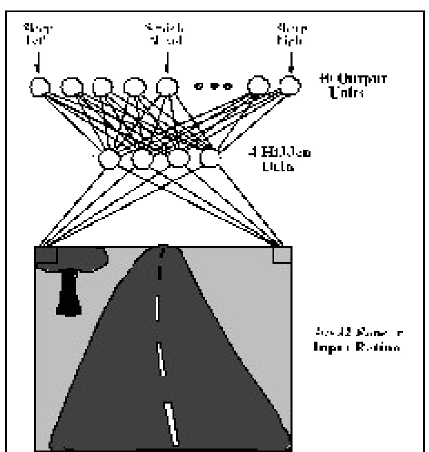
Camera image

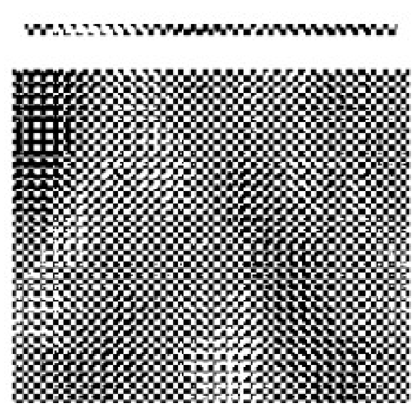


30 outputs for steering

4 hidden units

30x32 pixels as inputs



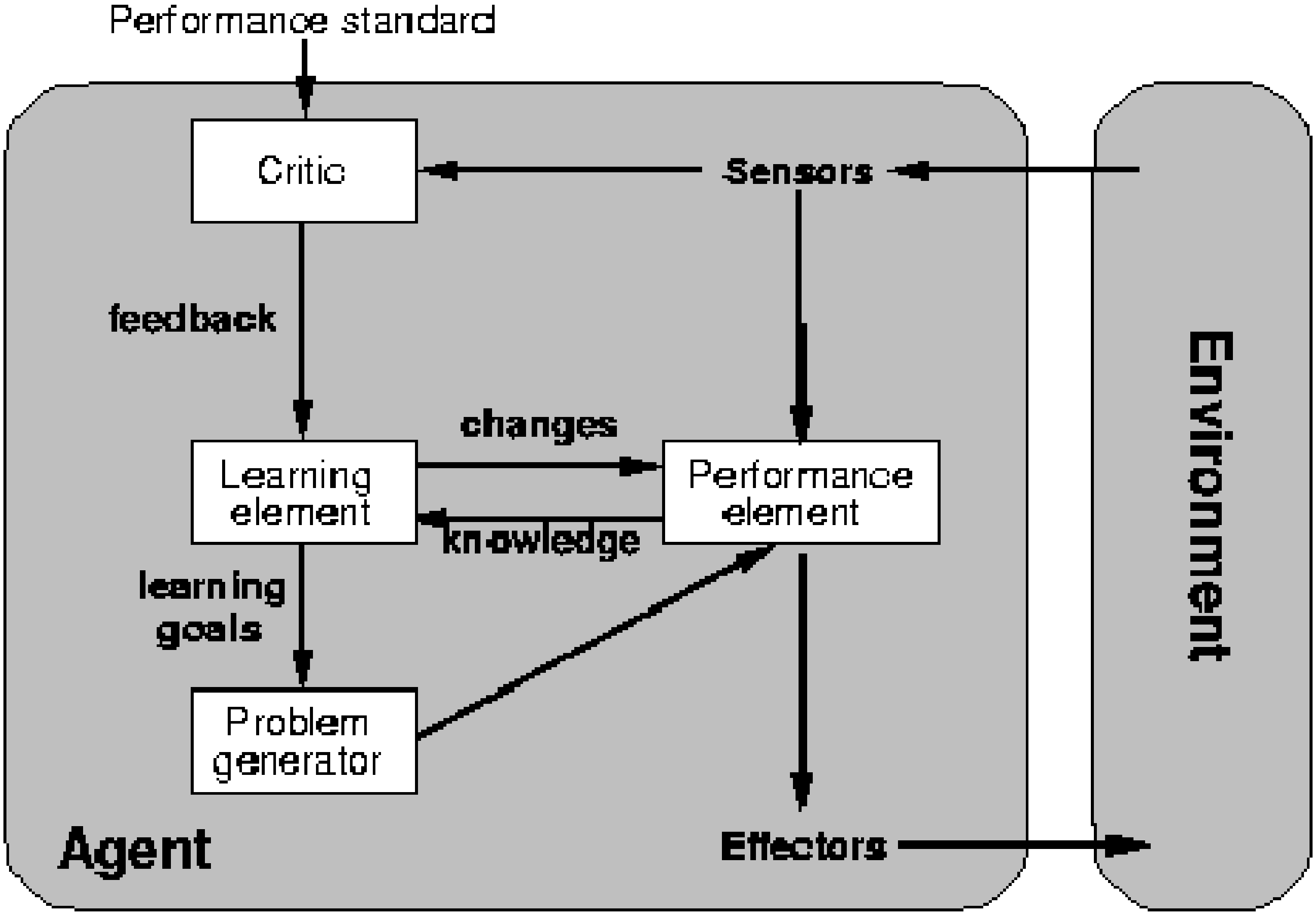


30x32 weights into one out of four hidden unit

Learning vs Adaptation

- ”Modification of a behavioral tendency by expertise.” (Webster 1984)
- ”A learning machine, broadly defined is any device whose actions are influenced by past experiences.” (Nilsson 1965)
- ”Any change in a system that allows it to perform better the second time on repetition of the same task or on another task drawn from the same population.” (Simon 1983)
- ”An improvement in information processing ability that results

A general model of learning agents



Disciplines relevant to ML

- Artificial intelligence
- Bayesian methods
- Control theory
- Information theory
- Computational complexity theory
- Philosophy
- Psychology and neurobiology
- Statistics
- Many practical problems in engineering and business

Machine Learning as

- Function approximation (mapping)
  - Regression
- Classification
- Categorization (clustering)
- Prediction
- Pattern recognition

ML in the real world

- **Real World Applications Panel: Machine Learning and Decision Support**

- Google
- Orbitz
- Astronomy

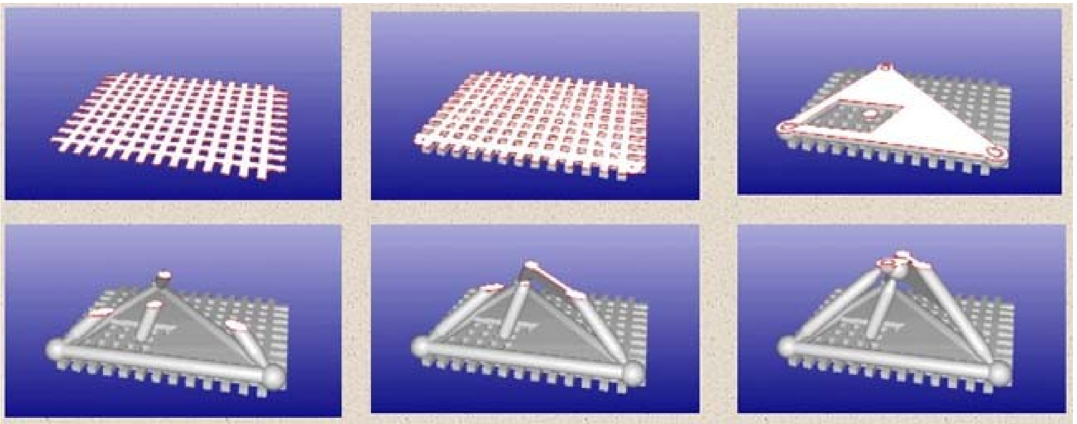
Working Applications of ML

- Classification of mortgages
- Predicting portfolio performance
- Electrical power control
- Chemical process control
- Character recognition
- Face recognition
- DNA classification
- Credit card fraud detection
- Cancer cell detection

Artificial Life

- GOLEM Project (Nature: Lipson, Pollack 2000)

  http://www.demo.cs.brandeis.edu/golem/
- Evolve simple electromechanical locomotion machines from basic building blocks (bars, acuators, artificial neurons) in a simulation of the physical world (gravity, friction).
- The individuals that demonstrate the best locomotion ability are fabricated through rapid prototyping technology.



Issues in Machine Learning

- What algorithms can approximate functions well and when
  - How does the number of training examples influence accuracy
- Problem representation / feature extraction
- Intention/independent learning
- Integrating learning with systems
- What are the theoretical limits of learnability
- Transfer learning
- Continuous learning

Measuring Performance

- Generalization accuracy
- Solution correctness
- Solution quality (length, efficiency)
- Speed of performance

Scaling issues in ML

- Number of
  - Inputs
  - Outputs
  - Batch vs realtime
  - Training vs testing

Machine Learning versus Human Learning

- Some ML behavior can challenge the performance of human experts (e.g., playing chess)
- Although ML sometimes matches human learning capabilities, it is not able to learn as well as humans or in the same way that humans do
- There is no claim that machine learning can be applied in a truly creative way
- Formal theories of ML systems exist but are often lacking (why a method succeeds or fails is not clear)
- ML success is often attributed to manipulation of symbols (rather than mere numeric information)

Observations

- ML has many practical applications and is probably the most used method in AI.
- ML is also an active research area
- Role of cognitive science
  - Computational model of cognition
    - ACT-R
- Role of neuroscience
  - Computational model of the brain
    - Neural networks
- Brain vs mind; hardware vs software
- Nearly all ML is still dependent on human “guidance”

Questions

- How does ML affect information science?
- Natural vs artificial learning – which is better?
- Is ML needed in all problems?
- What are the future directions of ML?