**Applications of Machine learning**

* Learning to recognize spoken words.

**All** of the most successful speech recognition systems employ machine learning in some form.

For example, the SPHINXsy stem (e.g., Lee 1989) learns speaker-specific strategies for recognizing the primitive sounds (phonemes) and words from the observed speech signal. Neural network learning methods (e.g., Waibel et al. 1989) and methods for learning hidden Markov models (e.g., Lee 1989) are effective for automatically customizing to,individual speakers, vocabularies, microphone characteristics, background noise, etc. Similar techniques have potential applications in many signal-interpretation problems.

* Learning to drive an autonomous vehicle.

Machine learning methods have been used to train computer-controlled vehicles to steer correctly when driving on a variety of road types. For example, the **ALVINN** system (Pomerleau 1989) has used its learned strategies to drive unassisted at 70 miles per hour for 90 miles on public highways among other cars. Similar techniques have possible applications in many sensor-based control problems.

* Learning to classify new astronomical structures.

Machine learning methods have been applied to a variety of large databases to learn general regularities implicit in the data. For example, decision tree learning algorithms have been used by NASA to learn how to classify celestial objects from the second Palomar Observatory Sky Survey (Fayyad et al. 1995). This system is now used to automatically classify **all** objects in the Sky Survey, which consists of three terrabytes of image data.

* Learning to play world-class backgammon.

The most successful computer programs for playing games such as backgammon are based on machiie learning algorithms. For example, the world's top computer program for backgammon, TD-GAMMON(T esauro 1992, 1995). learned its strategy by playing over one million practice games against itself. It now plays at a level competitive with the human world champion. Similar techniques have applications in many practical problems where very large search spaces must be examined efficiently..

**Some disciplines and examples of their influence on machine learning.**

* Artificial intelligence

Learning symbolic representations of concepts. Machine learning as a search problem. Learning as an approach to improving problem solving. Using prior knowledge together with training data to guide learning.

* Bayesian methods

Bayes' theorem as the basis for calculating probabilities of hypotheses. The naive Bayes classifier.Algorithms for estimating values of unobserved variables.

* Computational complexity theory

Theoretical bounds on the inherent complexity of different learning tasks, measured in terms of the computational effort, number of training examples, number of mistakes, etc. required in order to learn.

* Control theory

Procedures that learn to control processes in order to optimize predefined objectives and that learn to predict the next state of the process they are controlling.

* Information theory

Measures of entropy and information content. Minimum description length approaches to learning.

Optimal codes and their relationship to optimal training sequences for encoding a hypothesis.

* Philosophy

Occam's razor, suggesting that the simplest hypothesis is the best. Analysis of the justification for generalizing beyond observed data.

* Psychology and neurobiology

The power law of practice, which states that over a very broad range of learning problems, people's response time improves with practice according to a power law. Neurobiological studies motivating artificial neural network models of learning.

* Statistics

Characterization of errors (e.g., bias and variance) that occur when estimating the accuracy of a hypothesis based on a limited sample of data. Confidence intervals, statistical tests.

**Perspectives and Issues in Machine Learning**

* Perspective:
  + It involves searching a very large space of possible hypothesis to determine the one that best fits the observed data.
* Issues:
  + What algorithms exist for learning general target functions from specific training examples?
  + Which algorithms perform best for which types of problems and representations?
  + How much training data is sufficient? What general bounds can be found to relate the confidence in learned hypotheses to the amount of training experience and the character of the learner's hypothesis space?
  + When and how can prior knowledge held by the learner guide the process of generalizing from examples? Can prior knowledge be helpful even when it is only approximately correct?
  + What is the best strategy for choosing a useful next training experience, and how does the choice of this strategy alter the complexity of the learning problem?
  + What is the best way to reduce the learning task to one or more function approximation problems? Put another way, what specific functions should the system attempt to learn? Can this process itself be automated?
  + How can the learner automatically alter **its** representation to improve its ability to represent and learn the target function?