Summary

Concept Learning

- Hypotheses Space
- Relations between hypotheses
- Version Space
- Find-S
- Candidate Elimination
- Bias
  - Restriction Bias
  - Preference Bias
- Generalization

Concept Learning

- Hypotheses Space
- Relations between hypotheses
- Version Space
- Find-S
- Candidate Elimination
- Bias
  - Restriction Bias
  - Preference Bias
- Generalization

Concept Learning

- Hypotheses Space
- Relations between hypotheses
- Version Space
- Find-S
- Candidate Elimination
- Bias
  - Restriction Bias
  - Preference Bias
- Generalization

Concept Learning

- Hypotheses Space
- Relations between hypotheses
- Version Space
- Find-S
- Candidate Elimination
- Bias
  - Restriction Bias
  - Preference Bias
- Generalization

Concept Learning

- Hypotheses Space
- Relations between hypotheses
- Version Space
- Find-S
- Candidate Elimination
- Bias
  - Restriction Bias
  - Preference Bias
- Generalization

Concept Learning

- Hypotheses Space
- Relations between hypotheses
- Version Space
- Find-S
- Candidate Elimination
- Bias
  - Restriction Bias
  - Preference Bias
- Generalization

- Hypotheses Space
- Relations between hypotheses
- Version Space
- Find-S
- Candidate Elimination
- Bias
  - Restriction Bias
  - Preference Bias
- Generalization

Concept Learning

- Hypotheses Space
- Relations between hypotheses
- Version Space
- Find-S
- Candidate Elimination
- Bias
  - Restriction Bias
  - Preference Bias
- Generalization

Concept Learning

- Hypotheses Space
- Relations between hypotheses
- Version Space
- Find-S
- Candidate Elimination
- Bias
  - Restriction Bias
  - Preference Bias
- Generalization

Decision Trees

- Entropy — unpredictability
- Information Gain
- Overfitting
  Adaptation to non-generalizing details
- Pruning

Decision Trees

- Entropy — unpredictability
- Information Gain
- Overfitting
  Adaptation to non-generalizing details
- Pruning

Decision Trees

- Entropy — unpredictability
- Information Gain
- Overfitting
  Adaptation to non-generalizing details
- Pruning

Decision Trees

- Entropy — unpredictability
- Information Gain
- Overfitting
  Adaptation to non-generalizing details
- Pruning

Decision Trees

- Entropy — unpredictability
- Information Gain
- Overfitting
  Adaptation to non-generalizing details
- Pruning

Artificial Neuronal Networks

- Single Layer Networks
- Linear Separation
- Learning based on Error Minimization
- Multi Layer Networks
  - General Classifier
  - Function Approximator
- Differentiable Threshold Functions
- Error Back-Propagation
- Convergence Properties

Artificial Neuronal Networks

- Single Layer Networks
- Linear Separation
- Learning based on Error Minimization
- Multi Layer Networks
  - General Classifier
  - Function Approximator
- Differentiable Threshold Functions
- Error Back-Propagation
- Convergence Properties

Artificial Neuronal Networks

- Single Layer Networks
- Linear Separation
- Learning based on Error Minimization
- Multi Layer Networks
  - General Classifier
  - Function Approximator
- Differentiable Threshold Functions
- Error Back-Propagation
- Convergence Properties

Artificial Neuronal Networks

- Single Layer Networks
- Linear Separation
- Learning based on Error Minimization
- Multi Layer Networks
  - General Classifier
  - Function Approximator
- Differentiable Threshold Functions
- Error Back-Propagation
- Convergence Properties

Artificial Neuronal Networks

- Single Layer Networks
- Linear Separation
- Learning based on Error Minimization
- Multi Layer Networks
  - General Classifier
  - Function Approximator
- Differentiable Threshold Functions
- Error Back-Propagation
- Convergence Properties

Artificial Neuronal Networks

- Single Layer Networks
- Linear Separation
- Learning based on Error Minimization
- Multi Layer Networks
  - General Classifier
  - Function Approximator
- Differentiable Threshold Functions
- Error Back-Propagation
- Convergence Properties

Artificial Neuronal Networks

- Single Layer Networks
- Linear Separation
- Learning based on Error Minimization
- Multi Layer Networks
  - General Classifier
  - Function Approximator
- Differentiable Threshold Functions
- Error Back-Propagation
- Convergence Properties

Artificial Neuronal Networks

- Single Layer Networks
- Linear Separation
- Learning based on Error Minimization
- Multi Layer Networks
  - General Classifier
  - Function Approximator
- Differentiable Threshold Functions
- Error Back-Propagation
- Convergence Properties

Artificial Neuronal Networks

- Single Layer Networks
- Linear Separation
- Learning based on Error Minimization
- Multi Layer Networks
  - General Classifier
  - Function Approximator
- Differentiable Threshold Functions
- Error Back-Propagation
- Convergence Properties

Artificial Neuronal Networks

- Single Layer Networks
- Linear Separation
- Learning based on Error Minimization
- Multi Layer Networks
  - General Classifier
  - Function Approximator
- Differentiable Threshold Functions
- Error Back-Propagation
- Convergence Properties

Bayesian Learning

- Maximum a'Posteriori (MAP)
  The most likely hypothesis given the observed data
- Maximum Likelihood (ML)
  The hypothesis with highest likelihood of generating the data we have
- Bayes Optimal Classifier

$$\operatorname*{argmax}_{v} \sum_{h} P(v|h) \cdot P(h|D)$$

- Naive Bayes Classifier

$$\operatorname*{argmax}_{v} P(v) \prod_{i} P(a_i|v)$$

Bayesian Learning

- Maximum a'Posteriori (MAP)
  The most likely hypothesis given the observed data
- Maximum Likelihood (ML)
  The hypothesis with highest likelihood of generating the data we have
- Bayes Optimal Classifier

$$\underset{v}{\operatorname{argmax}} \sum_{h} P(v|h) \cdot P(h|D)$$

- Naive Bayes Classifier

$$\underset{v}{\operatorname{argmax}} P(v) \prod_{i} P(a_i|v)$$

Bayesian Learning

- Maximum a'Posteriori (MAP)
  The most likely hypothesis given the observed data
- Maximum Likelihood (ML)
  The hypothesis with highest likelihood of generating the data we have
- Bayes Optimal Classifier

$$\operatorname*{argmax}_{v} \sum_{h} P(v|h) \cdot P(h|D)$$

- Naive Bayes Classifier

$$\operatorname*{argmax}_{v} P(v) \prod_{i} P(a_i|v)$$

Bayesian Learning

- Maximum a'Posteriori (MAP)
  The most likely hypothesis given the observed data
- Maximum Likelihood (ML)
  The hypothesis with highest likelihood of generating the data we have
- Bayes Optimal Classifier

$$\operatorname*{argmax}_{v} \sum_{h} P(v|h) \cdot P(h|D)$$

- Naive Bayes Classifier

$$\operatorname*{argmax}_{v} P(v) \prod_{i} P(a_i|v)$$

Bayesian Learning

- Maximum a'Posteriori (MAP)
  The most likely hypothesis given the observed data
- Maximum Likelihood (ML)
  The hypothesis with highest likelihood of generating the data we have
- Bayes Optimal Classifier

$$\operatorname*{argmax}_{v} \sum_{h} P(v|h) \cdot P(h|D)$$

- Naive Bayes Classifier

$$\operatorname*{argmax}_{v} P(v) \prod_{i} P(a_i|v)$$

Boosting

- Generate several classifiers
- Use different training sample sets
- Voting can produce better results

Boosting

- Generate several classifiers
- Use different training sample sets
- Voting can produce better results

Boosting

- Generate several classifiers
- Use different training sample sets
- Voting can produce better results

Boosting

- Generate several classifiers
- Use different training sample sets
- Voting can produce better results

Instance based Learning

- Learning $\approx$ Storing
- Nearest Neighbor
- k-Nearest Neighbor
- Weighting (Kernel)
- Regression
- Radial Basis Functions

Instance based Learning

- Learning $\approx$ Storing
- Nearest Neighbor
- k-Nearest Neighbor
- Weighting (Kernel)
- Regression
- Radial Basis Functions

Instance based Learning

- Learning ≈ Storing
- Nearest Neighbor
- k-Nearest Neighbor
- Weighting (Kernel)
- Regression
- Radial Basis Functions

Instance based Learning

- Learning $\approx$ Storing
- Nearest Neighbor
- *k*-Nearest Neighbor
- Weighting (Kernel)
- Regression
- Radial Basis Functions

Instance based Learning

- Learning $\approx$ Storing
- Nearest Neighbor
- k-Nearest Neighbor
- Weighting (Kernel)
- Regression
- Radial Basis Functions

Instance based Learning

- Learning $\approx$ Storing
- Nearest Neighbor
- k-Nearest Neighbor
- Weighting (Kernel)
- Regression
- Radial Basis Functions

Instance based Learning

- Learning $\approx$ Storing
- Nearest Neighbor
- k-Nearest Neighbor
- Weighting (Kernel)
- Regression
- Radial Basis Functions

Reinforcement Learning

- Delayed Reward
- Temporal Credit Assignment
- Value Function
- Policy
- Temporal Difference techniques
- Q-Learning
- Sarsa-Learning

Reinforcement Learning

- Delayed Reward
- Temporal Credit Assignment
- Value Function
- Policy
- Temporal Difference techniques
- Q-Learning
- Sarsa-Learning

Reinforcement Learning

- Delayed Reward
- Temporal Credit Assignment
- Value Function
- Policy
- Temporal Difference techniques
- Q-Learning
- Sarsa-Learning

Reinforcement Learning

- Delayed Reward
- Temporal Credit Assignment
- Value Function
- Policy
- Temporal Difference techniques
- Q-Learning
- Sarsa-Learning

Reinforcement Learning

- Delayed Reward
- Temporal Credit Assignment
- Value Function
- Policy
- Temporal Difference techniques
- Q-Learning
- Sarsa-Learning

Reinforcement Learning

- Delayed Reward
- Temporal Credit Assignment
- Value Function
- Policy
- Temporal Difference techniques
- Q-Learning
- Sarsa-Learning

Reinforcement Learning

- Delayed Reward
- Temporal Credit Assignment
- Value Function
- Policy
- Temporal Difference techniques
- Q-Learning
- Sarsa-Learning

Reinforcement Learning

- Delayed Reward
- Temporal Credit Assignment
- Value Function
- Policy
- Temporal Difference techniques
- Q-Learning
- Sarsa-Learning

Genetic Algorithms

- Optimization
- Fitness Function
- Parallel Stochastic Gradient Following
- Crossover — Exchange of Partial Solutions
- Coding
- Genetic Programming

Genetic Algorithms

- Optimization
- Fitness Function
- Parallel Stochastic Gradient Following
- Crossover — Exchange of Partial Solutions
- Coding
- Genetic Programming

Genetic Algorithms

- Optimization
- Fitness Function
- Parallel Stochastic Gradient Following
- Crossover — Exchange of Partial Solutions
- Coding
- Genetic Programming

Genetic Algorithms

- Optimization
- Fitness Function
- Parallel Stochastic Gradient Following
- Crossover — Exchange of Partial Solutions
- Coding
- Genetic Programming

Genetic Algorithms

- Optimization
- Fitness Function
- Parallel Stochastic Gradient Following
- Crossover — Exchange of Partial Solutions
- Coding
- Genetic Programming

Genetic Algorithms

- Optimization
- Fitness Function
- Parallel Stochastic Gradient Following
- Crossover — Exchange of Partial Solutions
- Coding
- Genetic Programming

Genetic Algorithms

- Optimization
- Fitness Function
- Parallel Stochastic Gradient Following
- Crossover — Exchange of Partial Solutions
- Coding
- Genetic Programming

Learning Theory

- PAC-Learnable
- Probably (δ) Approximately (ϵ) Correct
- Complexity measured in $\frac{1}{\delta}$, $\frac{1}{\epsilon}$ och n
- Number of Training Examples

$$m \geq \frac{1}{\epsilon}\left[\ln|H| + \ln\frac{1}{\delta}\right]$$

- VC-dimension

$$\mathrm{VC}(H) \leq \log_2|H| \qquad m \geq \frac{1}{\epsilon}\left[4\log_2\frac{2}{\delta} + 8\mathrm{VC}(H)\cdot\log_2\frac{13}{\epsilon}\right]$$

- Errors During Learning

Learning Theory

- PAC-Learnable
- Probably (δ) Approximately (ϵ) Correct
- Complexity measured in $\frac{1}{\delta}$, $\frac{1}{\epsilon}$ och n
- Number of Training Examples

$$m \geq \frac{1}{\epsilon}\left[\ln|H| + \ln\frac{1}{\delta}\right]$$

- VC-dimension

$$\mathrm{VC}(H) \leq \log_2|H| \qquad m \geq \frac{1}{\epsilon}\left[4\log_2\frac{2}{\delta} + 8\mathrm{VC}(H)\cdot\log_2\frac{13}{\epsilon}\right]$$

- Errors During Learning

Learning Theory

- PAC-Learnable
- Probably (δ) Approximately (ϵ) Correct
- Complexity measured in $\frac{1}{\delta}$, $\frac{1}{\epsilon}$ och n
- Number of Training Examples

$$m \geq \frac{1}{\epsilon}\left[\ln |H| + \ln \frac{1}{\delta}\right]$$

- VC-dimension

$$\mathrm{VC}(H) \leq \log_2 |H| \qquad m \geq \frac{1}{\epsilon}\left[4\log_2 \frac{2}{\delta} + 8\mathrm{VC}(H) \cdot \log_2 \frac{13}{\epsilon}\right]$$

- Errors During Learning

Learning Theory

- PAC-Learnable
- Probably (δ) Approximately (ϵ) Correct
- Complexity measured in $\frac{1}{\delta}$, $\frac{1}{\epsilon}$ och n
- Number of Training Examples

$$m \geq \frac{1}{\epsilon}\left[\ln|H| + \ln\frac{1}{\delta}\right]$$

- VC-dimension

$$\mathrm{VC}(H) \leq \log_2|H| \qquad m \geq \frac{1}{\epsilon}\left[4\log_2\frac{2}{\delta} + 8\mathrm{VC}(H)\cdot\log_2\frac{13}{\epsilon}\right]$$

- Errors During Learning

Learning Theory

- PAC-Learnable
- Probably (δ) Approximately (ϵ) Correct
- Complexity measured in $\frac{1}{\delta}$, $\frac{1}{\epsilon}$ och n
- Number of Training Examples

$$m \geq \frac{1}{\epsilon}\left[\ln|H| + \ln\frac{1}{\delta}\right]$$

- VC-dimension

$$\mathrm{VC}(H) \leq \log_2|H| \qquad m \geq \frac{1}{\epsilon}\left[4\log_2\frac{2}{\delta} + 8\mathrm{VC}(H)\cdot\log_2\frac{13}{\epsilon}\right]$$

- Errors During Learning

- PAC-Learnable
- Probably (δ) Approximately (ϵ) Correct
- Complexity measured in $\frac{1}{\delta}$, $\frac{1}{\epsilon}$ och n
- Number of Training Examples

$$m \geq \frac{1}{\epsilon}\left[\ln|H| + \ln\frac{1}{\delta}\right]$$

- VC-dimension

$$\mathrm{VC}(H) \leq \log_2|H| \qquad m \geq \frac{1}{\epsilon}\left[4\log_2\frac{2}{\delta} + 8\mathrm{VC}(H)\cdot\log_2\frac{13}{\epsilon}\right]$$

- Errors During Learning

Learning Theory

- PAC-Learnable
- Probably (δ) Approximately (ϵ) Correct
- Complexity measured in $\frac{1}{\delta}$, $\frac{1}{\epsilon}$ och n
- Number of Training Examples

$$m \geq \frac{1}{\epsilon}\left[\ln|H| + \ln\frac{1}{\delta}\right]$$

- VC-dimension

$$\mathrm{VC}(H) \leq \log_2|H| \qquad m \geq \frac{1}{\epsilon}\left[4\log_2\frac{2}{\delta} + 8\mathrm{VC}(H)\cdot\log_2\frac{13}{\epsilon}\right]$$

- Errors During Learning

Support Vector Machines

- Linear separation in high-dimensional space
- Maximization of margins
- Minimizing the VC-dimension $\Rightarrow$ Optimal Generalization
- Kernels for efficient computation

Support Vector Machines

- Linear separation in high-dimensional space
- Maximization of margins
- Minimizing the VC-dimension $\Rightarrow$ Optimal Generalization
- Kernels for efficient computation

Support Vector Machines

- Linear separation in high-dimensional space
- Maximization of margins
- Minimizing the VC-dimension $\Rightarrow$ Optimal Generalization
- Kernels for efficient computation

Support Vector Machines

- Linear separation in high-dimensional space
- Maximization of margins
- Minimizing the VC-dimension $\Rightarrow$ Optimal Generalization
- Kernels for efficient computation

Support Vector Machines

- Linear separation in high-dimensional space
- Maximization of margins
- Minimizing the VC-dimension $\Rightarrow$ Optimal Generalization
- Kernels for efficient computation

Rule Based Learning

- Sequential Covering
- Successive Specialization
- Greedy or Beam-search
  Heuristic Search
- Inverse Resolution
  Bottom-up: explain the examples

Rule Based Learning

- Sequential Covering
- Successive Specialization
- Greedy or Beam-search
  Heuristic Search
- Inverse Resolution
  Bottom-up: explain the examples

Rule Based Learning

- Sequential Covering
- Successive Specialization
- Greedy or Beam-search
  Heuristic Search
- Inverse Resolution
  Bottom-up: explain the examples

Rule Based Learning

- Sequential Covering
- Successive Specialization
- Greedy or Beam-search
  Heuristic Search
- Inverse Resolution
  Bottom-up: explain the examples

Rule Based Learning

- Sequential Covering
- Successive Specialization
- Greedy or Beam-search
  Heuristic Search
- Inverse Resolution
  Bottom-up: explain the examples

Methods we have not talked about

- Learning of sequences/time-series
  Hidden Markov Models
  Kalman Filters
- Unsupervised Learning
  Self Organizing Maps
- Recurrent Neural Networks
- Hybrid Methods

Methods we have not talked about

- Learning of sequences/time-series
  Hidden Markov Models
  Kalman Filters
- Unsupervised Learning
  Self Organizing Maps
- Recurrent Neural Networks
- Hybrid Methods

Methods we have not talked about

- Learning of sequences/time-series
  Hidden Markov Models
  Kalman Filters
- Unsupervised Learning
  Self Organizing Maps
- Recurrent Neural Networks
- Hybrid Methods

Methods we have not talked about

- Learning of sequences/time-series
  Hidden Markov Models
  Kalman Filters
- Unsupervised Learning
  Self Organizing Maps
- Recurrent Neural Networks
- Hybrid Methods

Methods we have not talked about

- Learning of sequences/time-series
  Hidden Markov Models
  Kalman Filters
- Unsupervised Learning
  Self Organizing Maps
- Recurrent Neural Networks
- Hybrid Methods