
Bayesian Decision Theory and Machine Learning

Kathryn Blackmond Laskey
Department of Systems Engineering and Krasnow Institute
George Mason University

November 21, 1995

Decision Theory and Machine Learning

- **Decision theory provides a solid theoretical foundation for thinking about problems of action and inference under uncertainty**
 - framework for formulating problem
 - learning theory
 - framework for making tradeoffs between information and computational/data cost
 - **Machine learning is a problem of action/inference under uncertainty**
 - **Direct implementation of decision theory may not necessarily be best approach to practical learning**
 - Goal: approximate a decision theoretically sensible approach within resource constraints
 - The best way to do this may not be explicitly decision theoretic
-

Decision Theory

- **Decision problem**

- Possible actions $a \in A$
- Possible states of the world $s \in S$
- Consequences $c(s,a)$
- Goal: choose best action

- **Ingredients of a decision theoretic model**

- Utility function $u(c)$ expresses preferences for consequences
- Probability $P(s|a)$ expresses knowledge/uncertainty for states
- The best action maximizes expected utility

$$\begin{aligned} \mathbf{a}_{\text{optimal}} &= \mathbf{argmax}_a \{ \mathbf{E}[u|a] \} \\ &= \mathbf{argmax}_a \sum_s u(c(s,a))P(s | a) \end{aligned}$$



Bayesian Inference

- **Uncertainty about state of world is represented by a probability distribution over states**
 - Probability is a rational agent's *degree of belief* about uncertain states of the world
- **Beliefs are updated by conditioning on new information about the world**

$$\frac{P(s_i | x)}{P(s_j | x)} = \frac{P(x | s_i) P(s_i)}{P(x | s_j) P(s_j)}$$

Posterior odds ratio

Likelihood ratio

Prior odds ratio

- If there are “true” probabilities, any nondogmatic Bayesian who collects enough information will eventually learn them to within a close approximation
-

A Caricature of a Contrast

- **Statistical inference is about using large samples to draw inferences about a small number of parameters of an “objective” probability distribution**
 - Applies to inherently probabilistic phenomena
 - Don’t use statistics unless you have “enough data”
 - Don’t try to estimate too many things at once or test too many hypotheses at once
 - **Machine learning is about using small samples to learn the rules characterizing a phenomenon**
 - Applies to inherently deterministic phenomena
 - Not enough data to use statistics
 - Too many parameters (rules) to use statistics
-

Machine Learning as Bayesian Inference

- **The learning problem:**
 - **Given:** training set x of instances from some concept c
 - **Goal:** learn which concept c from family produced the training set
 - There may or may not be “noise” in the training data
 - **Bayesian inference applied to machine learning**
 - **Prior distribution $P(c)$ over**
 - **Likelihood function $P(x|c)$ for data given concept (may be deterministic)**
 - **Result of learning: posterior distribution $P(c|x)$ for concept given data**
-

Graphical Models for Probabilistic Reasoning

- **Bayesian networks**
 - Model for causal and/or correlational influences
 - Directed graph encodes dependency relationships
 - Local probability distributions encode strength of relationships
 - **Markov networks**
 - Model for correlational influences
 - Undirected graph encodes dependency relationships
 - Local probability distributions encode strength of relationships
 - **Hybrids and extensions**
 - **Other models that can be viewed as graphical probability models**
 - Neural networks
 - Networks of rules with certainty factors
-

Learning for High-Dimensional Parameter Spaces

- **In some classes of models there are exact Bayesian methods for computing the posterior distribution**
 - Decomposable models with complete data and conjugate prior distributions
 - **There are many approximate methods for cases in which exact methods are unavailable**
 - **Maximum likelihood or maximum a posteriori methods**
 - » EM algorithm
 - » Mean field approximation
 - » Backpropagation
 - **Monte Carlo**
 - » Gibbs sampling
 - » Metropolis-Hastings sampling
 - » Weighted Monte Carlo
-

Structural Uncertainty

- **Model can be decomposed as $M=(S, \theta)$**
 - **S - Structural assumptions (conditional independence, normality, connections in neural network, etc.)**
 - **θ is a structure-specific parameter (local probability distributions, mean and covariance of normal distribution, weights in neural network, etc.)**
 - **Traditional approach to statistical inference:**
 - **Pick “best” S**
 - **Estimate θ assuming S is the correct structure**
 - **Problems with traditional approach**
 - **Overfitting**
 - **Poor performance off training set**
 - **Underestimation of variance**
-

Approaches to Structural Uncertainty

- **Adjust significance levels for multiple hypothesis tests**
- **Sensitivity analysis to determine dependence of results on structural assumptions**
- **Holdout samples; cross-validation**
- **Formal Bayesian treatment of structural uncertainty**

Higher Order Uncertainty for Structures

- Structural and parameter uncertainty for concepts

$$\begin{aligned} P(c) &= \sum_{i=1} P(S_i) P(c|S_i) \\ &= \sum_{i=1} P(S_i) \sum_{s_i} P(c|s_i, S_i) f(s_i|S_i) \end{aligned}$$

- This sum cannot be computed explicitly
 - Approximate by searching over small part of space
 - Heuristic or Monte Carlo search
-

Learning about Structure

- Use Bayes rule to update prior distribution
- Posterior distribution for (S, s) given training sample

$$\begin{aligned} P(\phi | \mathbf{x}) &= \prod_{i=1} P(S_i | \mathbf{x}) P(\phi | \mathbf{x}, S) \\ &= \prod_{i=1} P(S_i | \mathbf{x}) \prod_{s_i} P(c | \mathbf{x}, s_i, S_i) f(s_i) \end{aligned}$$

- Learning algorithm for structure/parameter learning
 - Search heuristic for searching over structures
 - Method for computing posterior probabilities of structures (exact or approximate)
 - Method for approximating posterior out-of-sample predictions
-

Some Examples

- **Learning Bayesian networks**
 - Cooper & Herskovits, Heckerman, Shachter
 - **Bayesian learning of neural networks**
 - Neal, McKay, Hinton
 - **Bayesian learning of graphical models**
 - York, Madigan, Raftery, Buntine
-

Advantages to Model Averaging

- **Classical approach breaks down on high-dimensional parameter spaces**
 - Significance tests not valid when many models are considered
 - No good basis for deciding among competing model choice heuristics
 - No way to account for hidden variability due to model exploration
 - **Bayesian approach provides unified framework for:**
 - **Combining into a single analysis**
 - » exploration
 - » model choice
 - » parameter estimation
 - **Suggesting and evaluating competing heuristic approaches**
-

More Advantages

- **Ease of interpretation**
 - Bayesian: “The probability of a direct link between A and B is greater than .95.”
 - Classical: “The chance of getting a test statistic this extreme if there is no link between A and B is less than 5%.”
 - **Theory applies to any problem**
 - Discrete or continuous variables
 - Deterministic or stochastic phenomenon
 - Nested or non-nested models
 - iid or correlated variables
 - **General purpose algorithms for computing posterior distributions are becoming available**
 - **No need to choose arbitrary null hypothesis**
 - All hypotheses are compared simultaneously against each other
-

Criticisms

- **Theory is far ahead of ability to compute**
 - First figure out what you really want to compute, then try to approximate it
 - Decision theory provides a unified framework for thinking about what you want to compute
 - **Where do the priors come from?**
 - Bayesians are explicit about their assumptions. Assumptions are often buried in classical methods.
 - When you have knowledge, you should include it in the analysis
 - NFL theorems: there are no assumption-free methods
-

Issues

- **Identifiability**
 - Several concepts may be “observationally equivalent” (not distinguishable even with infinite data)
 - We usually care about good performance off training set and not about the “correct” concept
 - **The “correct” model may not be “good for purpose”**
 - Computational complexity
 - Storage requirements
 - Unintelligibility
 - **Many machine learning algorithms include “bias”**
 - Bias pushes system toward “good” models
 - Priors in a Bayesian analysis act as bias
 - Priors should be about belief not utility!
-

Decision Theory in Machine Learning

- **Goal: Acquire a high-utility problem representation**
 - **Utility includes:**
 - **Utility for base problem**
 - » Don't use information-theoretic distance when you care about correct treatment of patient
 - **Computational cost**
 - **Ease of explanation**
 - **Best model may not be explicitly decision theoretic**
 - **Approximate solution to the right problem is better than exact solution to the wrong problem**
-

Occam's Razor

- **Occam's razor says "prefer simplicity"**
- **As a heuristic it has stood the test of time**
- **It has been argued that Bayes justifies Occam's razor. More precisely, if:**
 - you put a positive prior probability on a sharp null hypothesis
 - the data are generated by a model "near" the null model
 - the sample size is not too large

Then (usually) the posterior probability of the null hypothesis is larger than its prior probability

Occam's Razor (cont.)

- Of course we don't really believe the null hypothesis!
 - We don't believe the alternative hypothesis either!
 - When predictive consequences of H_0 and H_A are similar:
 - H_0 is robust to plausible departures from H_0
 - When H_A has many parameters in relation to the amount of data available we may do much worse by using H_A
 - H_0 is robust to (likely) misspecification of parameters θ_A of H_A
 - But Occam's razor only works if we're willing to abandon simple hypotheses when they conflict with observations
-

Decision Theory and Occam's Razor

- **Occam's razor is really about utility and not probability**
 - Choose the simplest model that will give you good performance on problems you haven't seen
 - **Decision theoretic justification**
 - The simple model is not "correct"
 - Adding more parameters to fit the data is often not the way to make it correct
 - Too-complex models give false sense of precision and are difficult to apply
 - Occam's razor is a heuristic for finding high-utility models
-

Summary

- **Decision theory is a unified framework for**
 - Thinking about problems of machine learning
 - Designing machine learning approaches that can be expected to work well
 - Analyzing behavior of machine learning algorithms
 - **Machine learning problems are often formulated as inference problems**
 - **It may be fruitful to think about machine learning as an attempt to find a high-utility representation**
-