185.A83 Machine Learning for Health Informatics
2021S, VU, 2.0 h, 3.0 ECTS
Andreas Holzinger, Rudolf Freund
Marcus Bloice, Florian Endel, Anna Saranti

From Decision Making under Uncertainty to Probabilistic Graphical Models
Contact: andreas.holzinger@tuwien.ac.at


What will we learn today?
- 00 Reflection from last lecture
- 01 Decision Making under uncertainty
- 02 Some Basics of Graphs/Networks
- 03 Bayesian Networks (BN)
- 04 Markov Chain Monte Carlo (MCMC)
- 05 Metropolis Hastings Algorithm (MH)
- 06 Probabilistic Programming (PP)

Warm-up Quiz
On top-level - which machine learning approaches do we know?

- Symbolic ML
  - First order logic, inverse deduction, knowledge composition
  - Tom Mitchell, Steve Muggleton, Ross Quinlan, ...
- Bayesian ML
  - Statistical learning, probabilistic inference
  - Judea Pearl, Michael Jordan, David Heckermann, ...
- Cognitive ML
  - Analogisms from Psychology, Kernel machines
  - Vladimir Vapnik, Peter Hart, Douglas Hofstaedter, ...
- Connectionist ML
  - Neuroscience, Backpropagation
  - Geoffrey Hinton, Yoshua Bengio, Yann LeCun, ...
- Evolutionary ML
  - Nature-inspired concepts, genetic programming
  - John Holland (1929-2015), John Koza, Hod Lipson, ...

Why is this important for us?

- Decision node
- Chance node
- Outcome

Death from cancer
- Probability 2%
- utility 0%

Fertile survival
- Probability 98%
- utility 100%

Infertile death
- Probability 0.5%
- utility 0%

Infertile survival
- Probability 95%
- utility 95%


A short historical digression: whom do you see here?

01 Decision Making under uncertainty


What is medical action?

Decision Making: Learn good policy for selecting actions

Goal: Learn an optimal policy for selecting best actions within a given context

For $t = 1, \ldots, T$

1) The world produces a “context” $x_t \in X$

2) The learner selects an action $a_t \in \{1, \ldots, K\}$

3) The world reacts with a reward $r_t(a_t) \in [0,1]$

... permanent decision making under uncertainty!
What about the accuracy and uncertainty in medicine?

- Medical (clinical) data are defined and detected disturbingly “soft” ...
- ... having an obvious degree of variability and inaccuracy.
- Taking a medical history, the performance of a physical examination, the interpretation of laboratory tests, even the definition of diseases ... are surprisingly inexact.
- Data is defined, collected, and interpreted with a degree of variability and inaccuracy which falls far short of the standards which engineers do expect from most data.
- Moreover, standards might be interpreted variably by different medical doctors, different hospitals, different medical schools, different medical cultures, ...

Reasoning Foundations of Medical Diagnosis

Symbolic logic, probability, and value theory aid our understanding of how physicians reason.
Robert S. Leddy and Lee B. Levent

The purpose of this article is to analyze the complicated reasoning processes inherent in medical diagnosis. The impression of this problem has received recent emphasis by the increasing interest in the use of electronic computers as an aid to medical diagnostic processes assigned to a definite disease category, or that it may be one of several possible diagnoses, or else that in certain cases cannot be determined. This, obviously, is a greatly simplified explanation of the process of diagnosis, for the physician might also consider that after seeing a

Why is decision making so difficult?

- Computers are especially suited to help the physician collect and process clinical information and remind him of diagnoses which he may have overlooked. In many cases, computers may as well as a set of handwritten cards; scientists in other cases use sets of large-scale digital electronic computers.

- There are other ways in which computers may serve the physician, and some of these are suggested in this paper. For example, medical students might use the computer as an important aid in learning the methods of differential diagnosis. But to use the computer than we must understand how the physicians make a medical diagnosis. This, then, brings us to the subject of our investigation: the reasoning foundations of medical diagnosis and reasoning.

Medical diagnosis involves processes that can be computationally analyzed, as well as those characterized as “intuitive.” For instance, the reasoning founded on the use of medical diagnostic procedures

Why does a medical doctor make a decision?

3 July 1959, Volume 130, Number 5369

SCIENCE

02 Graphs = Networks

How can we learn and infer from data?

\[ p(h|d) = \frac{p(d|h) \cdot p(h)}{\sum_{h' \in \mathcal{H}} p(d|h') \cdot p(h')} \]

- Problem in \( \mathbb{R}^7 \) → complex

How do humans make decisions under uncertainty?

\[ p(\theta|D) = \frac{p(D|\theta) \cdot p(\theta)}{p(D)} \]

What are Probabilistic Graphical Models (PGM)?

- PGM can be seen as a combination between
  - Graph Theory + Probability Theory + Machine Learning
  - One of the most exciting advancements in AI in the last decades – with enormous future potential
  - Compact representation for exponentially-large probability distributions
    - Path \( (X, Y) := \text{edge}(X, Y) \)
    - Example Question: Is there a path connecting two proteins?
We start in 1736


252 years later: Belief propagation algorithm

Nobel Prize in Chemistry 2013

http://www.nobelprize.org/nobel_prizes/chemistry/laureates/2013

http://news.harvard.edu/gazette/story/2013/10/nobel_prize_awarded_2013
First Question: Where do graphs come from?

- Graphs as models for networks
- Given as direct input (point cloud data sets)
- Given as properties of a structure
- Given as a representation of information (e.g., Facebook data, viral marketing, etc., ...)

What do you see here?

NGC 5139 Omega Centauri by Edmund Hall in 1977, ESO, Atacama, Chile

Why is time and space important in health informatics?

Time

- e.g. Entropy

    Dali, S. (1931) The persistence of memory

Space

- e.g. Topology

    Bagula & Bourke (2012) Klein-Bottle

Why is the Complexity Problem: Time versus Space affecting us?

Exponential time versus polynomial space

P versus NP and the Computational Complexity Zoo, please have a look at
https://www.youtube.com/watch?v=9Y40lhMx3ts
Why are protein structures so important?


First yeast protein-protein interaction network


First human protein-protein interaction network


Getting Insight: Knowledge Discovery from Data


03 Bayesian Networks “Bayes’ Nets”
Bayesian reasoning and machine learning, Cambridge, Cambridge University Press.


http://www.cs.ucl.ac.uk/staff/d.barber/bml/

Probabilistic graphical models: principles and techniques, MIT press.


Chapter 8 Graphical Models is a sample chapter fully downloadable for free


http://bayes.cs.ucla.edu/BOOK-2K/

What are the rules of probability?

\[ P(x) = \sum_y P(x, y) \]

\[ P(x, y) = P(y|x)P(x) \]

\[ P(y|x) = \frac{P(x|y)P(y)}{P(x)} \]

\[ P(x) = \sum_y P(x|y)P(y) \]

Digression:
Markov Processes in Machine Learning
Why are Markov decision processes so important?

- Markov decision processes (MDP) are ... random processes in which the future, given the present, is independent of the past!
- one of the most important classes of random processes!

---

The Markov Process in Medical Prognosis

J. Robert Beck, M.D., and Stephen G. Pauker, M.D.

The physician's estimate of prognosis under alternative treatment plans is a principal factor in therapeutic decision making. Current methods of reporting prognosis, which include five-year survival, survival curves, and quality-adjusted life expectancy, are crude estimates of natural history. In this paper we describe a general-purpose model of medical prognosis based on the Markov process and show how this simple mathematical tool may be used to generate detailed and accurate assessments of life expectancy and health status. (Med Decis Making 3:49-459, 1983)

---

From where do we know such behaviour?

---

Standard RL-Agent Model goes back to Cybernetics 1950

Initialization T[i] arbitrarily
loop until policy good enough
  loop for n ∈ N
    Q(s,a) := R(s,a) + \gamma \sum_{s'} P(s'|s,a) T(s'|a)'
    T[i] := max_a Q(s,a)
  end loop
end loop

What has an RL-agent to do with MDP?

**Intelligent behavior** arises from the actions of an individual seeking to maximize its received reward signals in a complex and changing world.

- **Agent**
  - Representation
  - Learning algorithm
  - Action selection policy

- **Environment**

**Problem Formulation in a MDP**

- Markov decision processes specify setting and tasks
- Planning methods use knowledge of $P$ and $R$ to compute a good policy $\pi$
- Markov decision process model captures both sequential feedback and the more specific one-shot feedback (when $P(s'|s, a)$ is independent of both $s$ and $a$)

Supervised:
Learner told best $\alpha$

Exhaustive:
Learner shown every possible $x$

One-shot: Current $x$ independent of past $\alpha$

**Agent observes environmental state at each step $t$**

- 1) Overserves
- 2) Executes
- 3) Receives Reward

**Observation $O_t$**

- $O_t = sa_t = set$
- Agent state = environment state = information state
- Markov decision process (MDP)


Environmental State is the current representation

- i.e. whatever data the environment uses to pick the next observation/reward
- The environment state is not usually visible to the agent
- Even if $S$ is visible, it may contain irrelevant information
- A State $S_t$ is Markov iff:

$$
\mathbb{P}[S_{t+1} | S_t] = \mathbb{P}[S_{t+1} | S_1, ..., S_t]
$$

Agent State is the agent’s internal representation

- i.e. whatever information the agent uses to pick the next action
- It is the information used by reinforcement learning algorithms
- It can be any function of history:

$$
S = f(H)
$$

Components of RL Agents and Policy of Agents

- RL agent components:
  - Policy: agent’s behaviour function
  - Value function: how good is each state and/or action
  - Model: agent’s representation of the environment
- Policy as the agent’s behaviour
  - is a map from state to action, e.g.
  - Deterministic policy: $a = (s)$
  - Stochastic policy: $(a | s) = \mathbb{P}[A_t = a | S_t = s$
- Value function is prediction of future reward:

$$
\nu_\pi(s) = \mathbb{E}_\pi \left[ R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \ldots \mid S_t = s \right]
$$

What if the environment is only partially observable?

- Partial observability: when agent only indirectly observes environment
- Formally this is a Partially Observable Markov Decision Process (POMDP):
  - Agent must construct its own state representation $S$, for example:

  - Complete history: $S_t^\pi = H_t$
  - Beliefs of environment state: $S_t^\pi = (\mathbb{P}[S_t^\pi = s^1], ..., \mathbb{P}[S_t^\pi = s^p])$
  - Recurrent neural network: $S_t^\pi = \sigma(S_{t-1}^\pi W_s + O_t W_o)$
Back to Bayesian Networks

So, what is a directed Bayesian Network (BN)?

- is a **probabilistic model**, consisting of two parts:
  - 1) a dependency structure and
  - 2) local probability models.

\[
p(x_1, \ldots, x_n) = \prod_{i=1}^{n} p(x_i \mid Pa(x_i))
\]

Where \(Pa(x_i)\) are the parents of \(x_i\)

BN inherently model the uncertainty in the data. They are a successful marriage between probability theory and graph theory; allow to model a multidimensional probability distribution in a sparse way by searching independence relations in the data. Furthermore this model allows different strategies to integrate two data sources.


Three types of Probabilistic Graphical Models

**Undirected:** Markov random fields, useful e.g. for computer vision (Details: Murphy 19)

\[
p(X) = \frac{1}{Z} \exp \left( \sum_{ij} W_{ij} x_i x_j + \sum_i x_i b_i \right)
\]

**Directed:** Bayes Nets, useful for designing models (Details: Murphy 10)

\[
p(x) = \prod_{k=1}^{K} p(x_k \mid Pa_k)
\]

**Factored:** useful for inference/learning

\[
p(x) = \prod_{s} f_s(x_s)
\]
How can one predict the future on past data and present status?


What is important in clinical decision making?

- = the prediction of the future course of a disease conditional on the patient’s history and a projected treatment strategy
- Danger: probable information!
- Therefore valid prognostic models can be of great benefit for clinical decision making and of great value to the patient, e.g., for notification and quality-of-life decisions


Example: Breast cancer - Probability Table

<table>
<thead>
<tr>
<th>Category</th>
<th>Node description</th>
<th>State description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diagnosis</td>
<td>Breast cancer</td>
<td>Present, absent.</td>
</tr>
<tr>
<td>Clinical history</td>
<td>Habit of drinking alcoholic beverages and smoking</td>
<td>Yes, no.</td>
</tr>
<tr>
<td></td>
<td>Taking female hormones</td>
<td>Yes, no.</td>
</tr>
<tr>
<td></td>
<td>Have gone through menopause</td>
<td>Yes, no.</td>
</tr>
<tr>
<td></td>
<td>Have ever been pregnant</td>
<td>Yes, no.</td>
</tr>
<tr>
<td></td>
<td>Family member has breast cancer</td>
<td>Yes, no.</td>
</tr>
<tr>
<td>Physical findings</td>
<td>Nipple discharge</td>
<td>Yes, no.</td>
</tr>
<tr>
<td></td>
<td>Skin thickening</td>
<td>Yes, no.</td>
</tr>
<tr>
<td></td>
<td>Breast pain</td>
<td>Yes, no.</td>
</tr>
<tr>
<td></td>
<td>Have a lump(s)</td>
<td>Yes, no.</td>
</tr>
<tr>
<td>Mammographic findings</td>
<td>Architectural distortion</td>
<td>Present, absent.</td>
</tr>
<tr>
<td>Mass</td>
<td>Score from one to three, score from four to five, absent</td>
<td></td>
</tr>
<tr>
<td>Microcalcification cluster</td>
<td>Score from one to three, score from four to five, absent</td>
<td></td>
</tr>
</tbody>
</table>

Breast cancer – big picture – state of 1999


10 years later: Integration of microarray data

- Integrating microarray data from multiple studies to increase sample size;
- Approach to the development of more robust prognostic tests


Example: Bayes Net with four binary variables

<table>
<thead>
<tr>
<th>Gene</th>
<th>P(on)</th>
<th>P(off)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gene 1</td>
<td>0.8</td>
<td>0.2</td>
</tr>
</tbody>
</table>

Gene 2 | Gene 1 | Gene 1 |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>P(on)</td>
<td>0.3</td>
<td>0.6</td>
</tr>
<tr>
<td>P(off)</td>
<td>0.7</td>
<td>0.4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Gene 2</th>
<th>Gene 2 off</th>
<th>Gene 1 on</th>
<th>Gene 1 off</th>
</tr>
</thead>
<tbody>
<tr>
<td>P(on)</td>
<td>0.3</td>
<td>0.6</td>
<td></td>
</tr>
<tr>
<td>P(off)</td>
<td>0.7</td>
<td>0.4</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Prognosis</th>
<th>Gene 2 off off</th>
<th>Gene 2 off on</th>
<th>Gene 2 on off</th>
<th>Gene 2 on on</th>
</tr>
</thead>
<tbody>
<tr>
<td>P(good)</td>
<td>0.6</td>
<td>0.1</td>
<td>0.9</td>
<td>0.5</td>
</tr>
<tr>
<td>P(poor)</td>
<td>0.4</td>
<td>0.9</td>
<td>0.1</td>
<td>0.5</td>
</tr>
</tbody>
</table>


Concept Markov-Blanket

First, the structure is learned using a search strategy. Since the number of possible structures increases super exponentially with the number of variables, the well-known greedy search algorithm K2 can be used in combination with the Bayesian Dirichlet (BD) scoring metric:

\[
p(S|D) \propto p(S) \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{\Gamma(N'_{ij})}{\Gamma(N'_{ij} + N_{ijk})} \prod_{k=1}^{r_i} \frac{\Gamma(N'_{ijk} + N_{ijk})}{\Gamma(N'_{ijk})}
\]

\(N_{ijk}\) ... number of cases in the data set \(D\) having variable \(i\) in state \(k\) associated with the \(j\)-th instantiation of its parents in current structure \(S\).
\(n\) is the total number of variables.

Parameter learning -> second step

- Estimating the parameters of the local probability models corresponding with the dependency structure.
- CPTs are used to model these local probability models.
- For each variable and instantiation of its parents there exists a CPT that consists of a set of parameters.
- Each set of parameters was given a uniform Dirichlet prior:

\[
p(\theta_{ij}|S) = \text{Dir}(\theta_{ij}|N'_{ij1}, ..., N'_{ijk}, ..., N'_{ijr_i})
\]

Note: With \(\theta_{ij}\) a parameter set where \(i\) refers to the variable and \(j\) to the \(j\)-th instantiation of the parents in the current structure. \(\theta_{ij}\) contains a probability for every value of the variable \(x_i\) given the current instantiation of the parents. Dir corresponds to the Dirichlet distribution with \((N'_{ij1}, ..., N'_{ijk}, ..., N'_{ijr_i})\) as parameters of this Dirichlet distribution. Parameter learning then consists of updating these Dirichlet priors with data. This is straightforward because the multinomial distribution that is used to model the data, and the Dirichlet distribution that models the prior, are conjugate distributions. This results in a Dirichlet posterior over the parameter set:

\[
p(\theta_{ij}|D, S) = \text{Dir}(\theta_{ij}|N'_{ij1} + N_{ijk1}, ..., N'_{ijk} + N_{ijk}, ..., N'_{ijr_i} + N_{ijr_i})
\]

with \(N_{ijk}\) defined as before.

Next, \(N_{ij}\) is calculated by summing over all states of a variable:

\[
N_{ij} = \sum_{k=1}^{r_i} N_{ijk} \cdot N'_{ijk}
\]

and \(N'_{ij}\) have similar meanings but refer to prior knowledge for the parameters.

When no knowledge is available they are estimated using \(N_{ijk} = N/r_i q_i\)
with \(N\) the equivalent sample size,
\(r_i\) the number of states of variable \(i\) and
\(q_i\) the number of instantiations of the parents of variable \(i\).
\(\Gamma(\cdot)\) corresponds to the gamma distribution.

Finally \(p(S)\) is the prior probability of the structure.

\(p(S)\) is calculated by:

\[
p(S) = \prod_{i=1}^{n} \prod_{k=1}^{r_i} p(i \rightarrow x_i) \prod_{m=1}^{q_i} p(m|x_i)
\]

with \(p_i\) the number of parents of variable \(x_i\) and \(q_i\) all the variables that are not a parent of \(x_i\).

Next, \(p(a \rightarrow b)\) is the probability that there is an edge from \(a\) to \(b\) while \(p(ab)\) is the inverse, i.e., the probability that there is no edge from \(a\) to \(b\)

Predicting the prognosis of breast cancer (integrated a.)
For certain cases it is tractable if:
- Just one variable is unobserved
- We have singly connected graphs (no undirected loops -> belief propagation)
- Assigning probability to fully observed set of variables
- Possibility: Monte Carlo Methods (generate many samples according to the Bayes Net distribution and then count the results)
- Otherwise: approximate solutions ...

Often it is better to have a good solution within time – than an perfect solution too late ...

What Classes of Graphical Models do we know?

Remember

- Medicine is an extremely complex application domain – dealing most of the time with uncertainties -> **probable information**!
- When we have big data but little knowledge automatic ML can help to gain insight:
  **Structure learning and prediction in large-scale biomedical networks with probabilistic graphical models**
- If we have little data and deal with NP-hard problems we still need the human-in-the-loop

Protein Network Inference

- Hypothesis: most biological functions involve the interactions between many proteins, and the complexity of living systems arises as a result of such interactions.
- In this context, the problem of inferring a global protein network for a given organism,
- - using all (genomic) data of the organism,
- is one of the main challenges in computational biology


Problem: Is Graph Isomorphism NP-complete?

- Important for health informatics: Discovering relationships between biological components
- Unsolved problem in computer science:
- Can the graph isomorphism problem be solved in polynomial time?
  - So far, no polynomial time algorithm is known.
  - It is also not known if it is NP-complete
  - We know that subgraph-isomorphism is NP-complete

Finally a practical example

04 Markov Chain Monte Carlo (MCMC)

Monte Carlo Method (MC)
Monte Carlo Sampling
Markov Chains (MC)
MCMC
Metropolis-Hastings
What is the problem with observable data $\mathcal{D}$ in the real-world?

- Often we want to calculate characteristics of a high-dimensional probability distribution $p(\mathcal{D}|\theta)$.

$$p(h|d) \propto p(\mathcal{D}|\theta) \cdot p(h)$$

Posterior integration problem: (almost) all statistical inference can be deduced from the posterior distribution by calculating the appropriate sums, which involves an integration:

$$J = \int f(\theta) \cdot p(\theta|\mathcal{D}) d\theta$$

What is the problem of learning and inference?

- Statistical physics: computing the partition function – this is evaluating the posterior probability of a hypothesis and this requires summing over all hypotheses ... remember:

$$\mathcal{H} = \{H_1, H_2, ..., H_n\} \quad \forall (h, d)$$

$$P(h|d) = \frac{P(d|h) \cdot P(h)}{\sum_{h' \in \mathcal{H}} P(d|h') P(h')}$$

What was the origin of MCMC?

Summary: What are Monte Carlo methods?

- Class of algorithms that rely on repeated random sampling
- Basic idea: using randomness to solve problems with high uncertainty (Laplace, 1781)
- For solving multidimensional integrals which would otherwise intractable
- For simulation of systems with many dof
- e.g. fluids, gases, particle collectives, cellular structures - see our last tutorial on Tumor growth simulation!
for solving problems of probabilistic inference involved in developing computational models
as a source of hypotheses about how the human mind might solve problems of inference
For a function \( f(x) \) and distribution \( P(x) \), the expectation of \( f \) with respect to \( P \) is generally the average of \( f \), when \( x \) is drawn from the probability distribution \( P(x) \)

\[
\mathbb{E}_p(x)(f(x)) = \sum_x f(x)P(x)dx
\]

Solving intractable integrals
Bayesian statistics: normalizing constants, expectations, marginalization
Stochastic Optimization
Generalization of simulated annealing
Monte Carlo expectation maximization (EM)

Physical simulation via MC
Physical simulation
estimating neutron diffusion time
Computing expected utilities and best responses to Nash equilibria
Computing volumes in high-dimensions
Computing eigen-functions and values of operators (e.g., Schrödinger)
Statistical physics
Counting many things as fast as possible

Notations
Expectation of a function \( f(x, y) \) with respect to a random variable \( x \) is denoted by \( \mathbb{E}_x [f(x, y)] \)
In situations where there is no ambiguity as to which variable is being averaged over, this will be simplified by omitting the suffix, for instance \( \mathbb{E}_x \).
If the distribution of \( x \) is conditioned on another variable \( z \), then the corresponding conditional expectation will be written \( \mathbb{E}_x [f(x)|z] \)
Similarly, the variance is denoted \( \text{var}[f(x)] \), and for vector variables the covariance is written \( \text{cov}[x, y] \)
Global optimization: What is the main problem?

\[
\arg\max_x f(x)
\]

Normalization:
\[
p(x|y) = \frac{p(y|x) \cdot p(x)}{\int_X p(y|x) \cdot p(x) \, dx}
\]

Marginalization:
\[
p(x) = \int_Z p(x, z) \, dz
\]

Expectation:
\[
E_p(x)(f(x)) = \int_X f(x)p(x) \, dx
\]

Finally a practical example

05 Metropolis-Hastings Algorithm

5,233 citations as of 26.03.2017 (6,552 as of 22.04.2020)

JOURNAL OF THE AMERICAN STATISTICAL ASSOCIATION

Number 257

SEPTEMBER 1953

THE MONTE CARLO METHOD

NICHOLAS METROPOLIS AND S. ULAM

Los Alamos Laboratory

We shall present here the motivation and a general description of a method dealing with a class of problems in mathematical physics. The method is, essentially, a statistical approach to the study of differential equations, or more generally, of integro-differential equations that occur in various branches of the natural sciences.

A basic in the nineteenth century discovery that a deep distinction begins to appear between two different mathematical methods of treating physical phenomena. Problems involving only a few particles were studied by classical mechanics; through the study of systems of ordinary differential equations. For the description of systems with very many particles, an entirely different technique was used, namely, the method of statistical mechanics. In the latter approach, one does not concentrate on the individual particles but studies the properties of sets of particles. In pure mathematics an intensive study of the properties of sets of points was the subject of a new field. This is the so-called theory of sets, the lambda-theory of integration, and the twentieth century development of the theory of probability prepared the formal apparatus for the use of such models in theoretical physics, i.e., description of properties of aggregates of points rather than of individual points and their interactions.

Image Source: https://www.mahattanprojectcoi.org/xtal/histories/nicholas-metropolis-interview

34,140 cts (26.3.2017) - 37,202 (10.4.2018) - 41,751 (22.4.2020)

THE JOURNAL OF CHEMICAL PHYSICS

VOLUME 21, NUMBER 6

JUNE, 1953

Equation of State Calculations by Fast Computing Machines

NICHOLAS METROPOLIS, A. W. ROBINSON, MANDEL W. ROBINSON, AND ALAN H. TELLER

Los Alamos Scientific Laboratory, Los Alamos, N. Mex.

Edward Teller, Department of Physics, University of Chicago, Chicago, Illinois

A general method is described for bringing together the results of any sequence of computations involving a random number in order to estimate the true results of variations that are not included in the computer output. It provides a practical way to find the average of a function of a given variable, which must be calculated in a series of experiments. A special form of the method is described for the case where the sequence of experiments is characterized by a sequence of independent random variables.

I. INTRODUCTION

The purpose of the present paper is to describe a general method of evaluating the average of a function of a set of random variables, which may be constituted as a component of a general statistical system. The function is assumed to be the product of a function of a given variable and a function of the remaining variables. This function is integrable in the sense of the Lebesgue integral. The method is not suitable for numerical computation because it requires the evaluation of a large number of integrals of the function of the remaining variables. These integrals are usually evaluated by numerical integration. The method is not suitable for numerical computation because it requires the evaluation of a large number of integrals of the function of the remaining variables.

The method is not suitable for numerical computation because it requires the evaluation of a large number of integrals of the function of the remaining variables. The method is not suitable for numerical computation because it requires the evaluation of a large number of integrals of the function of the remaining variables. The method is not suitable for numerical computation because it requires the evaluation of a large number of integrals of the function of the remaining variables.

The method is not suitable for numerical computation because it requires the evaluation of a large number of integrals of the function of the remaining variables. The method is not suitable for numerical computation because it requires the evaluation of a large number of integrals of the function of the remaining variables. The method is not suitable for numerical computation because it requires the evaluation of a large number of integrals of the function of the remaining variables. The method is not suitable for numerical computation because it requires the evaluation of a large number of integrals of the function of the remaining variables. The method is not suitable for numerical computation because it requires the evaluation of a large number of integrals of the function of the remaining variables.
Monte Carlo sampling methods using Markov chains and their applications

By W. K. HASTINGS
University of Toronto

SUMMARY

A generalization of the sampling method introduced by Metropolis et al. (1953) is presented along with an exposition of the relevant theory, techniques of application and methods and difficulties of assessing the error in Monte Carlo estimates. Examples of the methods, including the generation of random orthogonal matrices and potential applications of the methods to numerical problems arising in statistics, are discussed.

1. INTRODUCTION

For numerical problems in a large number of dimensions, Monte Carlo methods are often more efficient than conventional numerical methods. However, implementation of the Monte Carlo methods requires sampling from high-dimensional probability distributions and this may be very difficult and expensive in analysis and computer time. General methods for sampling from, or estimating expectations with respect to, such distributions are as follows:

(i) If possible, factorize the distribution into the product of one-dimensional conditional distributions from which samples may be obtained.

(ii) Use importance sampling, which may also be used for variance reduction. That is, in order to evaluate the integral

\[ J = \int f(x) \, dx, \]

where \( f(x) \) is a probability density function, instead of obtaining independent samples \( x_1, \ldots, x_N \) from \( f(x) \) and using the estimate \( J = \frac{1}{N} \sum \delta f(x_i) \) we instead obtain the samples from

- Importance sampling is a technique to approximate averages with respect to an intractable distribution \( p(x) \).
- The term ‘sampling’ is arguably a misnomer since the method does not attempt to draw samples from \( p(x) \).
- Rather the method draws samples from a simpler importance distribution \( q(x) \) and then reweights them
- such that averages with respect to \( p(x) \) can be approximated using the samples from \( q(x) \).

So what is the MH-algorithm doing?

1. Choose a starting point \( x^1 \).
2. For \( i = 2 \) to \( N \) do
3. Draw a candidate sample \( x^{\text{end}} \) from the proposal \( q(x' | x^{i-1}) \).
4. Let \( u = \frac{q(x^{i-1} | x^*)}{q(x^* | x^{i-1})} \).
5. If \( u \leq 1 \) then \( x^i = x^{\text{end}} \).
6. else \( x^i = x^{i-1} \).
7. draw a random value \( w \) uniformly from the unit interval \([0, 1]\).
8. If \( w < u \) then \( x^i = x^{\text{end}} \).
9. else \( x^i = x^{i-1} \).
10. end if
11. end if
12. end if
13. end for

Why is Gibbs Sampling important?

The Gibbs Sampler is an interesting special case of MH:

Image Source: Peter Mueller, Anderson Cancer Center
How to learn modular structures from Network Data?


Algorithm 1 RJMCMC for sampling parameters

inputs:
Node Variables Data X
Network Data E
for iterations j = 1 to J do
    Sample \( \Theta^{(j)} \) given \( X^{(j)} \) using Alg 2 in Azizi et al., 2014
    Sample \( X^{(j+1)} \) given \( \Theta^{(j)} \) using Alg 3 in Azizi et al., 2014
    for modules i = 1 to \( K^{(j)} \) do
        Propose \( \omega^{(x)} \) \( \sim \) \( N(\omega^{(x)}, I) \)
        Accept with probability \( P_{\text{inc}} \) update \( Y^{(x)} \)
        for parents \( r = 1 \) to \( R_{k} \) do
            Propose \( \pi^{(x)} \) \( \sim \) \( N(\pi^{(x)}, I) \) accept with \( P_{\text{inc}} \)
            Propose \( \pi^{(x+1)} \) \( \sim \) \( N(\pi^{(x+1)}, I) \) accept with \( P_{\text{inc}} \)
        end for
    end for
end for
for condition \( c = 1 \) to \( C \) do
    Propose \( \mu^{(v)} \) \( \sim \) \( N(\mu^{(v)}, I) \) accept with \( P_{\text{inc}} \)
    Propose \( \mu^{(v+1)} \) \( \sim \) \( N(\mu^{(v+1)}, I) \) accept with \( P_{\text{inc}} \)
end for


Graphical Model


Myobacterium tuberculosis Gene Regulatory Network

An alternative approach

Discrete time modelling of disease incidence time series by MCMC

Discrete-time stochastic epidemic model of COVID-19

06 Probabilistic Programming
Book recommendations

- Practical Probabilistic Programming. Shuster Island (NY), Manning. (Avi Pfeffer 2016)
- Bayesian methods for hackers: probabilistic programming and Bayesian inference. Addison Wesely Professional. (Cameron Davidson-Pilon 2015)


Probabilistic-programming.org

- C → Probabilistic-C
- Scala → Figaro
- Scheme → Church
- Excel → Tabular
- Prolog → Prolog
- Javascript → webPP
- → Venture
- Python → PyMC

So, what is probabilistic programming?

- Probabilistic thinking is a valuable tool for decision making
- Overcoming uncertainties is the huge success currently in machine learning (and for AI ;-)
- Probabilistic reasoning is a versatile tool
- PPLs are domain specific languages that use probabilistic models and the methods to make inferences in those models
- The “magic” is in combining “probability methods” with “representational power”

Medical Example

1. Simple example: Nucleotide “A” may follow nucleotide “T” in the sequence more frequently than for outcome Y,
\[ P(A|T) > P(A|T, Y) \]

Image Source: Dan Williams, Life Technologies, Austin TX
Digression on Concept Learning

Recursive reasoning: a case for probabilistic programming

Why is decision making so hard for machines?

You are talking to your colleague and want to refer to the middle object – which wording would you prefer: circle or blue?


Why do we need concepts?
Concepts can be defined as a category membership

- can be relational and abstract
- category = set of objects that have commonalities
- concept = mental representation of categories
- concepts can be defined, e.g. triangle = a polygon with three sides, a gland = group of cells

Example: How do human pathologists make diagnoses?

What is ground truth? Where is the ground truth?

- := information provided by direct observation (empirical evidence) in contrast to information provided by inference
  - Empirical evidence = information acquired by observation or by experimentation in order to verify the truth (fit to reality) or falsify (non-fit to reality).
  - Empirical inference = drawing conclusions from empirical data (observations, measurements)
  - Causal inference = drawing conclusions about a causal connection based on the conditions of the occurrence of an effect
  - Causal machine learning is key to ethical AI in health to model explainability for bias avoidance and algorithmic fairness for decision making
When is a cup a cup? (When is a cat a cat?)

- Bruner, Goodnow, and Austin (1956) published “A Study of Thinking”, which became a landmark in cognitive science and has much influence on machine learning.
  - Rule-Based Categories
  - A concept specifies conditions for membership

Concept learning

- which is highly relevant for ML research, concerns the factors that determine the subjective difficulty of concepts:
  - Why are some concepts psychologically extremely simple and easy to learn,
  - while others seem to be extremely difficult, complex, or even incoherent?
  - These questions have been studied since the 1960s but are still unanswered ...


Bruder, Goodnow, and Austin (1956) published “A Study of Thinking”, which became a landmark in cognitive science and has much influence on machine learning.

- Rule-Based Categories
- A concept specifies conditions for membership

How can we model basic cognitive capacities as intuitive Bayes?

- Similarity
- Representativeness and evidential support
- Causal judgement
- Coincidences and causal discovery
- Diagnostic inference
- Predicting the future

\[ P(h|x, T) = \frac{P(x|h, T)P(h|T)}{\sum_{h' \in H_T} P(x|h', T)P(h'|T)} \]


How does our mind get so much out of it?


How can we learn words for objects – concepts from examples


How do we understand our world?

**What is probabilistic program induction?**

- **Deductive Reasoning** = Hypothesis > Observations > Logical Conclusions (general → specific – proven correctness)
  - DANGER: Hypothesis must be correct! DR defines whether the truth of a conclusion can be determined for that rule, based on the truth of premises: A=B, B=C, therefore A=C
- **Inductive reasoning** = makes broad generalizations from specific observations (specific → general – not proven correctness)
  - DANGER: allows a conclusion to be false if the premises are true
  - generate hypotheses and use DR for answering specific questions
- **Abductive reasoning** = inference = to get the best explanation from an incomplete set of preconditions.
  - Given a true conclusion and a rule, it attempts to select some possible premises that, if true also, may support the conclusion, though not uniquely.
  - Example: "When it rains, the grass gets wet. The grass is wet. Therefore, it might have rained." This kind of reasoning can be used to develop a hypothesis, which in turn can be tested by additional reasoning or data.

**What is the difference between deduction, induction, abduction?**

**Drawn by Human or Machine Learning Algorithm?**

**What can a Bayesian program learning (BPL) framework do?**

A Bayesian program learning (BPL) framework, capable of learning and on people.
What can we do with graphical models?

- Cognition as probabilistic inference
  - Visual perception, language acquisition, motor learning, associative learning, memory, attention, categorization, reasoning, causal inference, decision making, theory of mind
- Learning concepts from examples
- Learning causation from correlation
- Learning and applying intuitive theories (balancing complexity vs. fit)

Appendix

Thank you!