V9: Parameter Estimation for Bayesian Networks

Today, we assume that the network structure of a BN is given and that our data set D consists of fully observed instances of the network.

We want to „learn“ from the data set D the parameters in the probability distributions how the network variables affect eachother.

There exist 2 main approaches to deal with the parameter-estimation task:

- One is based on maximum-likelihood estimation (MLE)
- The other one uses Bayesian approaches.

Today we will focus on the MLE-approach.

Thumbtack example

Imagine that we have a **thumbtack** (dt. *Heftzwecke*) and
we conduct an experiment whereby we **flip** the thumbtack in the air.

It comes to land as either `head` or `tails`

By tossing the tumbtack several times, we obtain
a data set *x*[1] … *x*[n] of `head` or `tail` outcomes.

Based on this data set, we want to estimate the probability
with which the next flip will land heads or tails.

Thumbtack example

We assume implicitly that the thumbtack tosses are controlled by an (unknown) **parameter** θ, which describes the **frequency of heads**.

We also assume that the data instances are **independent** and **identically distributed**. This assumption is later abbreviated as IID.

One way of evaluating θ is by how well it **predicts the data**.

Suppose we observe the sequence of outcomes H, T, T, H, H.

The probability of the first toss is $P(X[1] = H) = \theta$

The probability of the second toss is $P(X[2] = T \mid X[1] = H)$

Since we assume that the coin tosses are independent, we can simplify this to

$$P(X[2] = T) = 1 - \theta$$

Thumbtack example

And so on… Thus, the probability of the full sequence is

$P(\langle H, T, T, H, H\rangle : \theta) = \theta\,(1-\theta)\,(1-\theta)\,\theta\,\theta = \theta^3\,(1-\theta)^2$

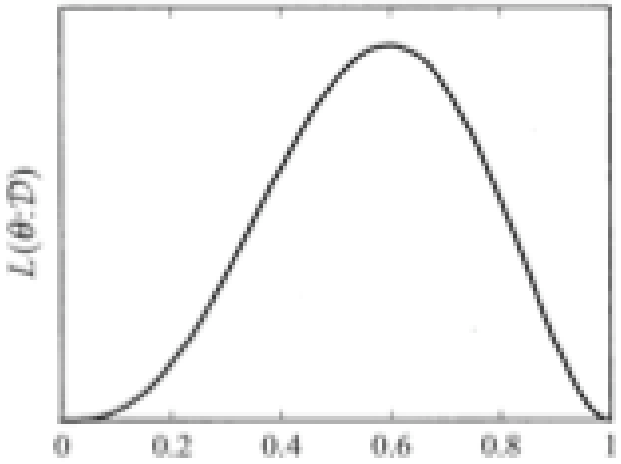
As expected, this probability depends on the particular value of θ.

As we consider different values of θ, we get different probabilities of the sequence.

Let us examine how the probability of the data changes as a function of θ.

We can define the **likelihood function** to be

$L(\theta : \langle H, T, T, H, H\rangle) =$
$P(\langle H, T, T, H, H\rangle : \theta) = \theta^3\,(1-\theta)^2$



Maximum likelihood estimator

Parameter values with higher likelihood
are more likely to generate the observed sequences.

Thus, we can use the likelihood function as our measure of quality
for different parameter values and select the parameter value
that maximizes the likelihood.

This value is called the **maximum likelihood estimator** (MLE).

From the figure, we see that $\theta = 0.6 = 3/5$ maximizes
the likelihood for the sequence H, T, T, H, H

Determining MLE

How can we find the MLE for the general case?

Assume that our data set D of observations contains M[1] heads and M[0] tails.

The likelihood function for this is $L(\theta : D) = \theta^{M[1]}(1-\theta)^{M[0]}$

It turns out that it is easier to maximize the **logarithm of the likelihood function**.

The log-likelihood function is:

$$l(\theta : D) = M[1]log\theta + M[0]log(1-\theta)$$

The log-likelihood is monotonically related to the likelihood.

Maximizing the one is equivalent to maximizing the other.

Determining MLE

In order to determine the value $\hat{\theta}$ that maximizes the log-likelihood, we take the derivative

$$\frac{dl}{d\theta} = M[1]\frac{1}{\theta} + M[0]\frac{1}{1-\theta}(-1)$$

and set this equal to zero, and solve for θ.

This gives

$$(1-\theta)M[1] = \theta M[0]$$
$$M[1] - \theta M[1] = \theta M[0]$$
$$M[1] = \theta(M[0] + M[1])$$
$$\hat{\theta} = \frac{M[1]}{M[1] + M[0]}$$

The ML principle

We now consider how to apply the ML principle to Bayesian networks.

Assume that we observe several IID samples
of a set of random variables X from an unknown distribution P*(X).

We assume that we know in advance the sample space we are dealing with
(i.e. which random variables and what values they can take).

However, we do not make any additional assumptions about P*.

We denote the training set of samples as D and assume
that it consists of M instances of X: $\xi[1] \ldots \xi[M]$.

Now we need to consider what exactly we want to learn.

The ML principle

We assume we are given a **parametric model**, defined by a function $P(\xi:\theta)$ for which we wish to estimate **parameters**.

Given a particular set of parameter values θ and an instance ξ of X, the model assigns a probability (or density) to ξ.

We require that for each choice of parameters θ , $P(\xi:\theta)$ is a legal distribution; that is, it is nonnegative and

$$\sum_{\xi} P(\xi:\theta) = 1$$

In general, for each model, not all parameter values are legal.

Thus, we need to define the parameter space θ, which is the set of allowable parameters.

The ML principle

As an example, the model we considered before has parameter space

$$\theta_{thumbtack}(x:\theta) = [0 \ \ldots \ 1]$$

and is defined as

$$P_{thumbtack}(x:\theta) = \begin{cases} \theta & \text{if } x = H \\ 1-\theta & \text{if } x = T \end{cases}$$

Another example

Suppose that X is a multinomial variable that can take values $x^1, \ldots x^K$.

The simplest representation of a multinomial distribution is a vector $\theta \in \Re^K$ such that

$$P_{multinomial}(x: \theta) = \theta_k \text{ if } x = x^k$$

The parameter space of this model is

$$\theta_{multinomial} = \left\{ \theta \in [0 \ \ldots \ 1]^K : \sum_i \theta_i = 1 \right\}$$

Another example

Suppose that X is a continuous variable that can take values in the real line.

A **Gaussian model** for X is

$$P_{Gaussian}(x: \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

where $\theta = \langle \mu, \sigma \rangle$

The parameter space for this model is $\theta_{Gaussian} = \Re \times \Re^+$
(we allow any real value of μ and any positive real value of σ).

Likelihood function

The next step in maximum likelihood estimation
is defining the **likelihood function**.

For a given choice of parameters θ the likelihood function is
the probability (or density) the model assigns the training data:

$$L(\theta : D) = \prod_m P(\xi[m] : \theta)$$

In the thumbtack example, we saw that we can write the likelihood function
using simpler terms with the counts M[1] and M[0].

The order of tosses was irrelevant.

The counts M[1] and M[0] were **sufficient statistics**
for the thumbtack learning problem.

Sufficient statistics

Definition:

A function $\tau(\xi)$ from instances of X to $\Re^l$ (for some l) is a **sufficient statistic** if, for any two data sets D and D‘ and any $\theta \in \Theta$, we have that

$$\sum_{\xi[m]\in D} \tau(\xi[m]) = \sum_{\xi'[m]\in D'} \tau(\xi[m]) \Rightarrow L(\theta: D) = L(\theta: D').$$

For the multinomial model, a sufficient statistic for the data is the tuple of counts $\langle M[1] \ldots M[K] \rangle$ such that M[k] is the number of times the value x^k appears in the training data.

To obtain these counts by summing instance-level statistics, we define $\tau(x)$ to be a tuple of dimension K, such that $\tau(x)$ has a 0 in every position, except at the position k for which $x = x^k$ where its value is 1.

Given the vector of counts, we can write the likelihood function as

$$L(\theta: D) = \prod_k {\theta_k}^{M[k]}$$

Likelihood function

The likelihood function measures the effect
of the choice of parameters on the training data.

If we have 2 sets of parameters θ and θ', so that $L(\theta : D) = L(\theta' : D)$, then we cannot, given only the data, distinguish between the 2 choices of parameters.

If $L(\theta : D) = L(\theta' : D)$ for all possible choices of D,
then the 2 parameters are **indistinguishable** for any outcome.

In such a situation, we can say in advance (i.e. before seeing the data)
that some distinctions cannot be resolved based on the data alone.

Likelihood function

Secondly, since we are maximizing the likelihood function,
we usually want it to be continuous (and preferably smooth) function of θ.

To ensure these properties, most of the theory of statistical estimation
requires that $P(\xi,\theta)$ is a continuous and differentiable function of θ,
and moreover that θ is a continuous set of points
(which is often assumed to be convex).

Likelihood function

Once we have defined the likelihood function,
we can use maximum likelihood estimation to choose the parameter values.

This can be stated formally as

Maximum Likelihood Estimation:
Given a data set D, choose parameters $\hat{\theta}$ that satisfy

$$L(\hat{\theta}: D) = \max_{\theta \in \Theta} L(\theta: D)$$

For the multinomial distribution, the maximum likelihood is attained when

$$\hat{\theta}_k = \frac{M[k]}{M}$$

i.e. the probability of each value of X
corresponds to its frequency in the training data.

Likelihood function

For the Gaussian distribution, the maximum is attained when μ and σ correspond to the empirical **mean** and **variance** of the training data:

$$\hat{\mu} = \frac{1}{M}\sum_{m} x[m]$$

$$\hat{\sigma} = \sqrt{\frac{1}{M}\sum_{m}(x[m] - \hat{\sigma})^2}$$

MLE for Bayesian networks

The simplest example of a nontrivial network structure is a network consisting of 2 binary variables, say X and Y, with an arc X → Y.

This network is parametrized by a parameter vector θ
which defines the set of parameters for all the CPDs in the network.

In this case, θ_{x^1} and θ_{x^0} specify the probability of the 2 values of X:

$\theta_{y^1|x^1}$ and $\theta_{y^0|x^1}$ specify the probability of Y given that X = x^1 and
$\theta_{y^1|x^0}$ and $\theta_{y^0|x^0}$ specify the probability of Y given that X = x^0.

For brevity, we use the shorthand $\boldsymbol{\theta}_{Y|x^0}$ to refer to the set $\{\theta_{y^1|x^0}, \theta_{y^0|x^0}\}$
and $\boldsymbol{\theta}_{Y|X}$ to refer to $\boldsymbol{\theta}_{Y|x^1} \cup \boldsymbol{\theta}_{Y|x^0}$.

MLE for Bayesian networks

In this example, every training instance is a tuple ⟨x[m], y[m]⟩ that describes a particular assignment to X and Y.

Our likelihood function is:

$$L(\theta : D) = \prod_{m=1}^{M} P(x[m], y[m] : \theta)$$

Our network model X → Y specifies that P(X,Y:θ) has a product form.

Thus we can write

$$L(\theta : D) = \prod_{m=1}^{M} P(x[m] : \theta) P(y[m] | x[m] : \theta)$$

MLE for Bayesian networks

Exchanging the order of multiplication, we can equivalently write this term as

$$L(\theta : D) = \left(\prod_m P(x[m] : \theta) \right) \left(\prod_m P(y[m] | x[m] : \theta) \right)$$

That is, the likelihood decomposes into 2 separate terms, one for each variable.

Each of these terms is a **local likelihood function**
that measures how well the variable is predicted given its parents.

Each term depends only on the parameters for that variable‘s CPD.

The first term, $\prod_m P(x[m] : \theta_X)$ is identical to the
multinomial distribution we discussed earlier.

MLE for Bayesian networks

The second term can be decomposed further

$$\prod_m P(y[m]|x[m]:\theta) =$$

$$= \prod_{m:x[m]=x^0} P\left(y[m]|x[m]:\theta_{Y|X}\right) \cdot \prod_{m:x[m]=x^1} P\left(y[m]|x[m]:\theta_{Y|X}\right)$$

$$= \prod_{m:x[m]=x^0} P\left(y[m]|x[m]:\theta_{Y|x^0}\right) \cdot \prod_{m:x[m]=x^1} P\left(y[m]|x[m]:\theta_{Y|x^1}\right)$$

Thus, in this example, the likelihood function decomposes into a product of terms, one for each group of parameters in θ.

This property is called the **decomposability** of the likelihood function.

MLE for Bayesian networks

We can do one more simplification by using the notion of sufficient statistics.

Let us consider one term in this expression $\prod_{m:x[m]=x^0} P\left(y[m]|x[m]:\theta_{Y|x^0}\right)$

Each of the individual terms $P\left(y[m]|x[m]:\theta_{Y|X}\right)$
can take one of 2 values, depending on the value of y[m].

If y[m] = y^1 , it is equal to $\theta_{y^1|x^0}$.

If y[m] = y^0 , it is equal to $\theta_{y^0|x^0}$.

How many cases of each type do we get?

MLE for Bayesian networks

Let us restrict attention to those data cases where x[m] = x^0.

These again partition into 2 categories.

We get $\theta_{y^1|x^0}$ in those cases where x[m] = x^0 and y[m] = y^1.
We use M[x^0, y^1] to denote their number.

We get $\theta_{y^0|x^0}$ in those cases where x[m] = x^0 and y[m] = y^0.
We use M[x^0, y^0] to denote their number.

Thus the term in the eq. is equal to:

$$\prod_{m:x[m]=x^0} P\left(y[m]|x[m]:\theta_{Y|x^0}\right) = \left(\theta_{y^1|x^0}\right)^{M[x^0,y^1]} \cdot \left(\theta_{y^0|x^0}\right)^{M[x^0,y^0]}$$

MLE for Bayesian networks

Based on our discussion of the multinomial likelihood,

we know that we maximize $\theta_{Y|x^0}$ by setting

$$\theta_{y^1|x^0} = \frac{M[x^0, y^1]}{M[x^0, y^1] + M[x^0, y^0]} = \frac{M[x^0, y^1]}{M[x^0]}$$

and similarly for $\theta_{y^0|x^0}$.

Thus, we can find the maximum likelihood parameters in this CPD by simply counting how many times each of the possible assignments of X and Y appears in the training data.

It turns out that these counts of the various assignments for some set of variables are generally useful.

MLE for Bayesian networks

Definition

Let Z be some set of random variables,

and z be some instantiation of these random variables.

Let D be a data set.

We define M[z] to be the number of entries in D that have Z[m] = z

$$M[z] = \sum_m \mathbf{1}\{Z[m] = z\}$$

This approach can be extended to general Bayesian networks.

Global Likelihood decomposition

We start by examining the likelihood function of a Bayesian network.

Suppose we want to learn the parameters for a BN
with structure G and parameters θ.

We are also given a data set D consisting of samples $\xi[1] \ldots \xi[M]$.

Writing the likelihood and repeating the previous steps gives

$$L[\theta : D] = \prod_m P_G(\xi[m] : \theta)$$

$$= \prod_m \prod_i P\left(x_i[m] \middle| pa_{X_i}[m] : \theta\right)$$

$$= \prod_i \left[\prod_m P\left(x_i[m] \middle| pa_{X_i}[m] : \theta\right) \right]$$

Global Likelihood decomposition

Each of the terms in the square brackets $\left[\prod_m P\left(x_i[m]\middle|pa_{X_i}[m]:\theta\right)\right]$ refers to the conditional likelihood of a particular variable given its parents in the network.

We use $\theta_{X_i\middle|Pa_{X_i}}$ to denote the subset of parameters that determines $P\left(X_i\middle|Pa_{X_i}\right)$ in our model. Then, we can write

$$L(\theta:D)=\prod_i L_i\left(\theta_{X_i\middle|Pa_{X_i}}:D\right)$$

where the **local likelihood function** for X_i is:

$$L_i\left(\theta_{X_i\middle|Pa_{X_i}}:D\right)=\prod_m P\left(x_i[m]\middle|pa_{X_i}[m]:\theta_{X_i\middle|Pa_{X_i}}\right)$$

This form is particularly useful when the parameters $\theta_{X_i\middle|Pa_{X_i}}$ are disjoint. That is, each CPD is parametrized by a separate set of parameters that do not overlap.

Global Likelihood decomposition

The previous analysis showed that the likelihood decomposes as a product of indepdendent terms, one for each CPD in the network.

This important property is called the **global decomposition** of the likelihood function.

Proposition

Let D be a complete data set for $X_1, \ldots X_n$, let G be a network structure over these variables and suppose that the parameters $\theta_{X_i|Pa_{X_i}}$ are disjoint from $\theta_{X_j|Pa_{X_j}}$ for all $j \neq i$. *Let* $\hat{\theta}_{X_i|Pa_{X_i}}$ be the parameters that maximize $L_i\left(\theta_{X_i|Pa_{X_i}} : D\right)$.

Then $\hat{\theta} = \langle \hat{\theta}_{X_1|Pa_1}, \ldots, \hat{\theta}_{X_n|Pa_n} \rangle$ maximizes L(θ : D).

In other words, we can maximize each local likelihood function *independently* of the rest of the network, and then combine the solutions to get an MLE solution.

Table-CPDs

We now consider the simplest parametrization of the CPDs: a table-CPD.

Suppose we have a variable X with parents U.

If we represent that CPD P(X | U) as a table, then we will have a parameter $\theta_{x|u}$ for each combination of $x \in Val(X)$ and $u \in Val(U)$.

In this case we can write the local likelihood function as:

$$L_X(\theta_{X|U} : D) = \prod_m \theta_{x[m]|u[m]}$$

$$= \prod_{u \in Val(U)} \left[\prod_{x \in Val(x)} (\theta_{x|u})^{M[u,x]} \right]$$

where M[u,x] is the number of times $\xi[m] = x$ and $u[m] = u$.

Table-CPDs

We need to maximize this term under the constraints that, for each choice of value for the parents **U** , the conditional probability is legal, that is

$$\sum \theta_{x|u} = 1 \quad \text{for all u.}$$

We can maximize each of the terms in square brackets in the previous eq. independently.

We can further decompose the local likelihood function for a tabular CPD into a product of simple likelihood functions that are each multinomial likelihoods.

The counts in the data for the different outcomes *x* are [M[u,x] : $x \in$ Val(X)].

The MLE parameters turn out to be $\hat{\theta}_{x|u} = \frac{M[u,x]}{M[u]}$ where $M[u] = \sum_x M[u,x]$