Throwing the Bayesian dice: Sampling (approximate inference) — Starting from the Monte Carlo method

 Jingming 2025-03-30 06:37:35 United States

Preface

I often have this experience: some things that are easy to explain are always given very high-level names to discourage outsiders. Typical examples are: Euclidean Distance and Manhattan Distance.

Taking a two-dimensional plane as an example, the former is actually the straight-line distance *d* = sqrt((*x* 1− *x2*)^2 + (y *1*− y2 *)*^ 2), and the latter is the sum of the absolute values of the differences in the directions of each coordinate axis d=| *x 1* −x *2*| +| *y1* −y2| *, which* is a calculation that *even junior high school students* can do.

If the former is named after Euclid, it is understandable; the latter actually means that when walking in Manhattan, you can only follow the straight and horizontal streets from point A to point B, so the distance you walk is the Manhattan distance. According to this logic, can it also be called Tokyo Distance? Chang'an City in the Tang Dynasty was like this, so where was Manhattan at that time... Why not call it Chang'an Distance?

I think that in order to spread knowledge with the least resistance, we should try our best to avoid presupposing a priori knowledge that is irrelevant to the problem. The best philosophy is the philosophy that children can understand. The same is true for the best physics/mathematics/... But since the world is already like this, let us at least understand it better.

This article will start with the Monte Carlo Method and then introduce several Monte Carlo method variants for sampling in Bayesian networks.

---

1. What is the Monte Carlo method?

Don’t let the name scare you! Even if you don’t recognize it, you’ve intuitively used it countless times.

The Monte Carlo method refers to: by generating a large number of random samples, using the statistical characteristics of the samples (such as frequency, mean) to approximate the target probability or expected value.

Although there are many variations, the most common Monte Carlo method is: **P(A|B) ≈ count(A,B) / count (B)** .

Take tossing a coin as an example: we don't know the probability of heads, but we can toss it **n** times, count the number of heads **n_positive** , and then estimate the probability of heads as **P(positive) = n_positive/n** . As long as n is large enough, this estimated probability can be close enough to the target probability.

---

2. Reasoning in Bayesian Networks: Computation (Exact Inference) — Is It Always Possible?

Let's take this Bayesian network as an example: **A->B->C, the value of each node is 0 or 1**

Given: **P(A), P(B|A), P(C|B)**

Question: **P(A=1|C=1)?**

calculate

P(A=1|C=1) = P(A=1,C=1) / P(C=1) ——This is the conditional probability formula, which can be derived based on the chain rule.

= ∑_B P(A=1,B,C=1) / ∑_A ∑_B P(A,B,C=1) —— The left side sums all possible values of B and eliminates the B variable; the right side eliminates the B and A variables.

= ∑_B P(A=1) * P(B|A=1) * P(C=1|A,B) / ∑_A ∑_B P(A) * P(B|A) * P(C=1|A,B) ——这也是链式法则。

= ∑_B P(A=1) * P(B|A=1) * P(C=1|B) / ∑_A ∑_B P(A) * P(B|A) * P(C=1|B) ——Why? Because given B (B as a condition), C is conditionally independent of A, that is, C⊥A|B.

The answer can be obtained by calculation, but let's look at **the time complexity** :

For this example, each time we sum, we need to calculate 3 multiplications for 2 possible values. The number of times we sum the numerator is 3-2=1, and the number of times we sum the denominator is 3-1=2:

3*2^1 + 3*2^2

For k values and n nodes, each sum needs to calculate n multiplications of k values, the number of sums of the numerator is n-2 times, and the number of sums of the denominator is n-1 times:

n*k^(n-1) + n*k^(n-2) = **O(k^n)**

The calculation method, as the number of nodes n increases, the time complexity increases exponentially - exponential explosion!

If the topology of the Bayesian network is more complex, with multiple paths and a ring structure, the calculation may be very difficult.

---

3. Reasoning of Bayesian Networks: Sampling (Approximate Inference) — Easier

Due to the exponential explosion of computational methods and the difficulty of calculation, it is much easier to approximate inference through sampling methods for Bayesian networks with complex structures. We will introduce four sampling methods: **Prior Sampling** , **Rejection Sampling** , **Likelihood Weighting Sampling** , and **Gibbs Sampling.** Because they all try to use the statistical properties of samples to approximate the target probability, they are all different variants of the Monte Carlo method.

For the same example:

A->B->C, the value of each node is 0 or 1

Given: **P(A), P(B|A), P(C|B)**

For a clearer explanation, the question is changed to: **P(A=1|B=1)?**

In the sampling method, the condition B=1 of the problem P(A=1|B=1) is called evidence. We will see its role later.

For the sake of discussion, let's assume the known probabilities:

A	$P(A)$
0	0.6
1	0.4

A	B	$P(B \mid A)$
0	0	0.7
0	1	0.3
1	0	0.2
1	1	0.8

B	C	$P(C \mid B)$
0	0	0.9
0	1	0.1
1	0	0.4
1	1	0.6

Known Probability

Prior Sampling

1. Randomly generate a set of random numbers [0,1] to see which probability interval the node falls in and determine the sampling result of the node.

Suppose we generate a set of random numbers like this: 0.5, 0.6, 0.95, 0.8, 0.9, 0.98, ...

2. Multiple sampling

The first set of samples:

The sampling of A is determined by 0.5, which falls within the probability interval of P(A=0), so the sampling A=0

The sampling of B is determined by 0.6, which falls within the probability interval of P(B=0|A=0), so the sampling B=0

The sampling of C is determined by 0.95, which falls within the probability interval of P(C=1|B=0), so C=1

The first set of results: A=0, B=0, C=1

The second set of samples:

The sampling of A is determined by 0.8, which falls within the probability interval of P(A=1), so the sampling A=1

Use 0.9 to determine the sampling of B, which falls within the probability interval of P(B=1|A=1), so the sampling B=1

The sampling of C is determined by 0.98, which falls within the probability interval of P(C=1|B=1), so C=1

The second set of results: A=1, B=1, C=1

...repeat multiple times

3. Problem P(A=1|B=1) = count(A=1,B=1) / count(B=1), done.

Rejection Sampling

Just now in Prior Sampling, what we need is only the relevant sampling of **P(A=1|B=1)** , that is, the sampling with condition B=1, and then calculate the proportion of A=1. Therefore, for the first set of samples, it is obvious from B=0 that it is not what we need, so there is no need to sample the following nodes, and this set of samples is directly rejected. A new set of samples is started from the beginning. This is what Rejection means, saving a lot of work.

Likelihood Weighting Sampling

The above Rejection Sampling is very convenient, but the problem is that we will reject many sampling opportunities that are not relevant to the problem. Is there a way to avoid wasting them and make every sampling useful? This is Likelihood Weighting Sampling.

The difference is that each time the evidence is sampled, the value required by the evidence in the problem is directly taken, and a weight is calculated for this set of samples. **W= P(e_1 | parents(e_1)) * P(e_2 | parents(e_2)) * ....** at last **P = ratio of ∑weight** , rather than the ratio of count.

In this example, each time sampling, Evidence B=1 is directly set, and Wi =1 * P(B=1|A) is calculated.

Repeat multiple times, each group is useful (because Evidence B=1), and each has a corresponding Wi.

Finally, P(A=1|B=1) = ∑weight(A=1|B=1) / ∑weight(B=1) .

Why does this work?

Because although we use random values for sampling, the sampling results of Evience conform to its probability distribution. In this case, we directly set Evidence equal to the value we want and set the weight to the corresponding probability distribution. Finally, we use the ratio of weights instead of the ratio of samples to calculate P.

Gibbs Sampling

The Likelihood Weighting Sampling above is very good, but if there are many evidence variables (for example, B=1, C=1 are fixed at the same time), the weights of the likelihood weighted sampling may become very small or uneven (large weight variance), resulting in unstable results. And when the network structure is complex (for example, there are loops), likelihood weighted sampling is difficult to handle.

Gibbs Sampling , which is **a Markov Chain Monte Carlo (MCMC)** method, is suitable for dealing with multivariate and complex network structures. The idea is: start with an initial sample, keep the evidence constant, update the value of only one variable each time you sample, and keep the other variables constant. Through continuous iteration, a series of samples are generated, and finally the target distribution is approached.

Still using the example A->B->C, find P(A=1|B=1).

Initialize a set of samples randomly, such as [A=0, B=1, C=0] (note that B=1 is a fixed evidence). In each iteration, select a non-evidence variable (A or C) to update:

1. Update A: Sample new values according to P(A|B=1, C=0), assuming that A=1;

New sample: [A=1, B=1, C=0];

2. Update C: Sample new values according to P(C|A=1, B=1), assuming that C=1;

New sample: [A=1, B=1, C=1];

3. Repeat several times and record all samples;

4. The result P(A=1|B=1) ≈ the proportion of A=1 in the sample.

Note: Gibbs sampling requires dynamically computing new conditional probability distributions, iterating enough times, and the initial samples may affect early results, so the first few steps are usually discarded (called the "burn-in" phase).

Since Gibbs sampling requires dynamic calculation of new conditional probability distributions, it is more suitable for dealing with multiple variables and complex network structures. When there is a simple network and less evidence, Likelihood Weighting sampling can directly use the known conditional probability distribution, which makes calculation easier.

Summarize

If you can only remember one sentence from the entire article... Euclidean distance is the straight-line distance, Manhattan distance should be called Chang'an distance, Monte Carlo is to calculate the probability by statistics, and sampling is used when Bayesian network cannot calculate. Prior is to throw it randomly, Rejection is to use it selectively, Likelihood is to add weights, Gibbs is to iterate continuously...

完