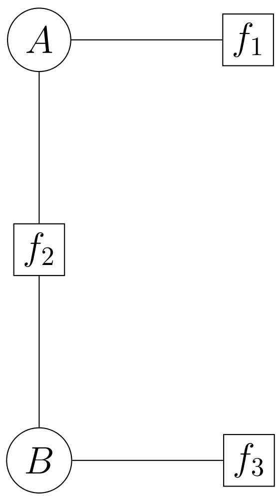
AIRR Handout on Max Product and Sampling

Nishanth Kumar

Last Revised: 02/20/2025

1 Max-Product Algorithm (with argmax tracking)

Let's study the max-product algorithm on an example problem:



The factor tables are:

A	$f_1(A)$
1	5
0	10

B	$f_3(B)$
1	10
0	1

A	B	$f_2(A, B)$
1	1	1
1	0	50
0	1	2
0	0	10

If we were to attempt to find the most-likely assignment to variables A and B simply by eyeballing these factor tables, we might pick $A = 0$ and $B = 1$ because those maximize the unary factors f_1 and f_3 connected to variables A and B respectively.

However, this naive eyeballing doesn't account for factor f_2. Let's walk through message passing for max product to see how to do this properly, and how the answer differs from our initial eyeballing.

1.1 Step 1: Compute Messages from Leaf Nodes

1.1.1 Compute Message from f_1 to A

Since $f_1(A)$ is a unary factor:

$$\mu_{f_1 \to A}(A) = f_1(A)$$

Thus, the messages are:

A	$\mu_{f_1 \to A}(A)$
1	5
0	10

1.1.2 Compute Message from f_3 to B

Since $f_3(B)$ is a unary factor:

$$\mu_{f_3 \to B}(B) = f_3(B)$$

Thus, the messages are:

B	$\mu_{f_3 \to B}(B)$
1	10
0	1

1.2 Step 2: Compute Message from Intermediate Node

1.2.1 Compute Message from B to f_2

Since B receives a message from f_3, it simply passes it along:

$$\mu_{B \to f_2}(B) = \mu_{f_3 \to B}(B)$$

Thus, the messages are:

B	$\mu_{B \to f_2}(B)$
1	10
0	1

1.3 Step 3: Compute Message from f_2 to A (Tracking $\arg\max$)

The max-product update rule is:

$$\mu_{f_2 \to A}(A) = \max_B f_2(A, B) \cdot \mu_{B \to f_2}(B)$$

Multiplying by the message from B:

A	B	$f_2(A, B) \cdot \mu_{B \to f_2}(B)$
1	1	$1 \times 10 = 10$
1	0	$50 \times 1 = 50$
0	1	$2 \times 10 = 20$
0	0	$10 \times 1 = 10$

Maximizing over B:

$$\mu_{f_2 \to A}(A = 1) = \max(10, 50) = 50, \quad \arg\max_B = 0$$

$$\mu_{f_2 \to A}(A = 0) = \max(20, 10) = 20, \quad \arg\max_B = 1$$

Thus, the messages are:

A	$\mu_{f_2 \to A}(A)$
1	50
0	20

Tracking $\arg\max$:
- If $A = 1$, the best B is **0**. - If $A = 0$, the best B is **1**.

1.4 Step 4: Compute Final Message at A

$$\mu_A(A) = \mu_{f_1 \to A}(A) \cdot \mu_{f_2 \to A}(A)$$

A	$\mu_{f_1 \to A}(A)$	$\mu_{f_2 \to A}(A)$	$\mu_A(A)$
1	5	50	$5 \times 50 = 250$
0	10	20	$10 \times 20 = 200$

Since $250 > 200$, we choose:

$$A = 1$$

1.5 Step 5: Compute B Using Tracked $\arg\max$

From **Step 3**, we tracked:

- If $A = 1$, the best B was **0**. - If $A = 0$, the best B was **1**.

Since we now know $A = 1$, we select:

$$B = 0$$

1.6 Final Answer

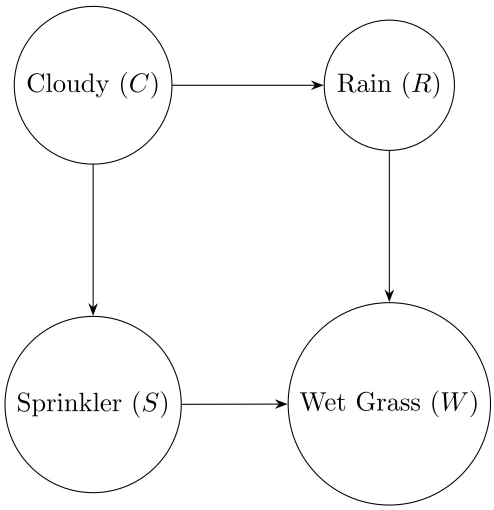
$$A = 1, \quad B = 0$$

Note that this is different than the answer we obtained by eyeballing the factors in the beginning (i.e. $A = 0, B = 1$). Intuitively, this should make sense by looking at factor f_2: it provides $A = 1, B = 0$ a much higher score than $A = 0, B = 1$.

2 All About Sampling

In lecture, we discussed the conceptual differences between rejection sampling and importance sampling. Let's study a concrete example to shed more light on these methods, as well as the main differences between them.

Consider the below example Bayes net:



The (conditional) probability tables for this network are as follows. Our goal will be to use sampling within this Bayes Net to estimate Our goal is to estimate $P(C = 1 \mid W = 1)$.

Cloudy (C)	P(C)
1	0.5
0	0.5

Table 1: Probability Distribution of Cloudy

Rain (R)	$C = 0$	$C = 1$
$P(R = 1 \mid C)$	0.2	0.8
$P(R = 0 \mid C)$	0.8	0.2

Table 2: Conditional Probability of Rain given Cloudy

2.1 Rejection Sampling

Let's try to solve this with rejection sampling. Recall that to perform this, we'll need to first perform ancestral sampling on the bayes net above to derive samples.

Sprinkler (S)	$C = 0$	$C = 1$
$P(S = 1 \mid C)$	0.5	0.1
$P(S = 0 \mid C)$	0.5	0.9

Table 3: Conditional Probability of Sprinkler given Cloudy

Wet Grass (W)	$R = 0, S = 0$ $R = 1, S = 0$	$R = 0, S = 1$ $R = 1, S = 1$
$P(W = 1 \mid R, S)$	0.1 0.9	0.9 0.99

Table 4: Conditional Probability of Wet Grass given Rain and Sprinkler

2.1.1 Step 1: Generating Samples Using Ancestral Sampling

Ancestral sampling essentially involves 'walking down' the bayes net and generating values for each of the variables in turn. In this case, we'd start with variable C. We sample according to the probability distribution $P(C)$ (which is uniform, and is thus equivalent to flipping a fair coin) and obtain 1 for instance. Given this, we sample R via $P(R|C = 1) : P(R = 1|C = 1) = 0.8, P(R = 0|C = 1) = 0.2$, let's say this gives us 1 as well. We move on to sampling S and ultimately W in a similar manner. Let's say we draw 8 samples in this fashion, illustrated in the table below.

Sample	C	R	S	Sampled W	Keep Sample?
1	1	1	1	1	Yes
2	0	1	0	1	Yes
3	1	0	1	1	Yes
4	0	0	1	1	Yes
5	1	1	0	1	Yes
6	0	0	0	0	No
7	1	0	0	0	No
8	0	0	0	0	No

Table 5: Generated Samples Using Ancestral Sampling

From these samples, we throw out any samples where $W = 0$ because we are ultimately interested in $P(C = 1 \mid W = 1)$.

2.1.2 Step 2: Estimating Posterior $P(C|W = 1)$

Using only the accepted samples:

- Accepted samples: {1, 2, 3, 4, 5}
- Number of samples where $C = 1$: 3
- Total accepted samples: 5

Thus, the rejection sampling estimate is:

$$P(C = 1|W = 1) \approx \frac{3}{5} = 0.6 \tag{1}$$

2.2 Importance Sampling

Now, let's study how importance sampling would work on the same problem.

We are interested in getting $P(C = 1 \mid W = 1)$. Our *evidence* here (denoted in lecture notes by E) is $W = 1$. Recall from lecture that importance sampling draws N samples $x_1, x_2, \ldots, x_N$ and computes an *importance weight* w_i for each sample x_i. The importance weight formula is $\Pi_j P(e_j|\text{parents}(E_j))$. Here j refers to each variable used as part of evidence. Since there is only one variable here, our importance weight formula is just $P(W = 1|\text{parents}(W))$ for every sample x_i.

So specifically, we will use ancestral sampling — just as we did in rejection sampling — to sample values for variables C, R, S. We will then look at our probability table $P(W|R, S)$ to get

$P(W = 1|R, S)$ and use this to get the *importance weight*. Finally, we will normalize these weights into a probability distribution.

Let's say we draw the exact same 8 samples as above in rejection sampling. The importance weights are computed in the table below:

Sample	C	R	S	$w_i = P(W\|R, S)$
1	1	1	1	0.99
2	0	1	0	0.9
3	1	0	1	0.9
4	0	0	1	0.9
5	1	1	0	0.9
6	0	0	0	0.1
7	1	0	0	0.1
8	0	0	0	0.1

Table 6: Generated Samples Using Ancestral Sampling along with importance weights.

Given these weights, we can compute our approximation for $P(C = 1|W = 1)$ by picking out the weights where $W = 1$ and $C = 1$ and dividing through by the sum of all weights.

$P(C = 1|W = 1) \approx \frac{0.99+0.9+0.9+0.1}{0.99+0.9+0.9+0.9+0.9+0.1+0.1+0.1} = 0.591$.

Notice here that *even the samples where W = 0* influenced the final value. That's the power of importance sampling; we're able to get a better estimate of the probability we care about (by leveraging knowledge of the conditional probability tables from the bayes net) using the same amount of samples used by rejection sampling.