

Probabilistic Paths

by Sven Nilsen, 2017

A probabilistic path is similar to an existential path, but for reasoning about uncertainty. For example, the following two functions f_0 and f_1 have the same existential path:

$$\begin{aligned} f_0 &:= \lambda(x : \text{nat}) = \text{if } (x \% 2) == 0 \{ 1 \} \text{ else } \{ 0 \} \\ f_1 &:= \lambda(x : \text{nat}) = \text{if } (x \% 10) == 0 \{ 1 \} \text{ else } \{ 0 \} \end{aligned}$$

$$\exists f_0 \iff \exists f_1 \iff \lambda(x : \text{nat}) = x == 0 \vee x == 1$$

Yet, the function f_0 returns 1 five times as often compared to f_1 :

$$\begin{aligned} f_0 &: 1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ \dots \\ f_1 &: 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ \dots \end{aligned}$$

So, their probabilistic paths are different:

$$\begin{aligned} \exists_p f_0 &:= \lambda(_ : \text{nat}) = \text{if } x == 0 \vee x == 1 \{ 1/2 \} \text{ else } \{ 0 \} \\ \exists_p f_1 &:= \lambda(x : \text{nat}) = \text{if } x == 0 \{ 9/10 \} \text{ else if } x == 1 \{ 1/10 \} \text{ else } \{ 0 \} \end{aligned}$$

All boolean functions have an assigned probability of returning true , by using a higher order if function:

$$P(\text{if}(A, B)(x)) = P(x) \cdot P(A) + (1 - P(x)) \cdot P(B)$$

$$\text{if} := \lambda(a, b) = \lambda(c) = \text{if } c \{ a \} \text{ else } \{ b \}$$

Just like the existential path can not inherit knowledge about the input, the probability path uses a uniform random distribution for the input. This means the probability of input $P(x)$ gets replaced by $1/2$ after deriving the probability formula for a boolean function. The probabilistic path of every boolean function can be constructed by the probx function:

$$\text{probx} := \lambda(k : \text{real} \wedge [\text{prob}] \text{ true}) = \lambda(x : \text{bool}) = \text{if } x \{ k \} \text{ else } \{ 1 - k \}$$

$$\text{prob} := \lambda(x : \text{real}) = x \geq 0 \wedge x \leq 1$$

For example, for the function and :

$$\begin{aligned} P(\text{and}(x_0, x_1)) &= P(x_0) \cdot P(x_1) = 1/2 \cdot 1/2 = 1/4 \\ \exists_p \text{and} &\iff \text{probx}(1/4) \end{aligned}$$

In probability theory there are conditional probabilities. The equivalent concept for probabilistic paths is constructed by adding sub-type constraints to the input before taking the probabilistic path:

$$\exists_{p\text{and}}\{ (= \text{true}), _ \} \Leftrightarrow \exists_{p\text{id}} \Leftrightarrow \text{probx}(1/2)$$

Translating back and forth between the two notations is straight forward:

$$\begin{aligned} P(\text{and}(a, b) \mid a) &= (\exists_{p\text{and}}\{ (= \text{true}), _ \})(\text{true}) \\ P(\neg\text{and}(a, b) \mid a) &= (\exists_{p\text{and}}\{ (= \text{true}), _ \})(\text{false}) \end{aligned}$$

In proofs, inject the sub-type constraints so you get only free inputs and then take the probabilistic path:

$$\underline{P(\text{and}(a, b) \mid a) = (\exists_{p\text{id}})(\text{true})}$$

$$P(\text{and}(a, b) \mid a)$$

$$P(\text{and}(a : (= \text{true}), b))$$

$$P(\text{and}\{ (= \text{true}), _ \}(a, b))$$

$$(\exists_{p\text{and}}\{ (= \text{true}), _ \})(\text{true})$$

$$\text{and}\{ (= \text{true}), _ \} \Leftrightarrow \text{id}$$

$$(\exists_{p\text{id}})(\text{true})$$

$$\underline{P(\neg\text{and}(a, b) \mid a) = (\exists_{p\text{id}})(\text{false})}$$

$$P(\neg\text{and}(a, b) \mid a)$$

$$P(\neg\text{and}(a : (= \text{true}), b))$$

$$P(\neg\text{and}\{ (= \text{true}), _ \}(a, b))$$

$$\text{not} \cdot \text{and} \Leftrightarrow \text{nand}$$

$$P(\text{nand}\{ (= \text{true}), _ \}(a, b))$$

$$(\exists_{p\text{nand}}\{ (= \text{true}), _ \})(\text{true})$$

Here I do a trick by inverting the output.

$$(\exists_{p\text{and}}\{ (= \text{true}), _ \})(\text{false})$$

$$\text{and}\{ (= \text{true}), _ \} \Leftrightarrow \text{id}$$

$$(\exists_{p\text{id}})(\text{false})$$