Bit Blasting Probabilistic Programs

Poorva Garg, Steven Holtzen, Guy Van den Broeck, Todd Millstein

UCLA

FPTalks - July 11 2024
What are Probabilistic Programs?

Programs that represent probability distributions:

```plaintext
a ~ flip(0.7)
b ~ if a
    then normal(0, 1)
    else normal(2, 1)
return b
```

Primary analysis task is probabilistic inference:

\[
pr(b) = \sum_{a_i} pr(a = a_i) pr(b|a = a_i)
\]
\[
= \frac{7}{10} \cdot pr(b|a = 1) + \frac{3}{10} \cdot pr(b|a = 0)
\]
\[
= \frac{7}{10} e^{-\frac{1}{2}b^2} + \frac{3}{10} e^{-\frac{1}{2}(2-b)^2}
\]
What are Probabilistic Programs?

Programs that represent probability distributions:

\[ a \sim \text{flip}(0.7) \]
\[ b \sim \begin{cases} \text{normal}(0, 1) & \text{if } a \\ \text{normal}(2, 1) & \text{else} \end{cases} \]

return \( b \)

Primary analysis task is probabilistic inference:

\[
pr(b) = \sum_{a_i} pr(a = a_i) pr(b|a = a_i) \\
= \frac{7}{10} pr(b|a = 1) + \frac{3}{10} pr(b|a = 0) \\
= \frac{7}{10} e^{-\frac{1}{2}b^2} + \frac{3}{10} e^{-\frac{1}{2}(2-b)^2}
\]

**discrete** + **continuous** = **hybrid** probabilistic program
Hybrid is Not Well Supported

- Hamiltonian Monte Carlo
  - Pyro
  - Stan

- Sequential Monte Carlo
  - WebPPL

- Algebraic Evaluation
  - PSI

- Knowledge Compilation
  - Dice
  - ProbLog

- Limited support for discreteness
- Scalability and accuracy issues
- Closed form does not always exist
- No support for continuous
Bit Blasting a Continuous Random Variable

Infinite binary representation in $[0,1)$: $X \sim 0.b_1b_2b_3...$

- All random variables are discrete
- Representation is exact
- Exposes useful structure (e.g., arithmetic)
- Infinite number of random bits
Bit Blasting a Continuous Random Variable

**Finite** binary representation in $[0,1)$: $X \sim 0.b_1 b_2 b_3 \ldots b_k$

- all random variables are discrete
- representation is exact **up to k bits**
- exposes useful structure (e.g., arithmetic)

? does the distribution over k bits have a program using a few independent coin flips?
Bit Blasting the Uniform

\[ X \sim 0. b_1 b_2 b_3 \]

... represent bits using a probabilistic program of coin flips ...

![Graph showing probability distribution]

![Graph showing probability density function]
Bit Blasting the Uniform

\[ X \sim 0.b_1b_2b_3 \]

Naive discretization

\[
\begin{align*}
\text{if flip}(1/8) & \ [0, 0, 0] \\
\text{elif flip}(1/7) & \ [0, 0, 1] \\
\text{elif flip}(1/6) & \ [0, 1, 0] \\
\text{elif flip}(1/5) & \ [0, 1, 1] \\
\text{elif flip}(1/4) & \ [1, 0, 0] \\
\text{elif flip}(1/3) & \ [1, 0, 1] \\
\text{elif flip}(1/2) & \ [1, 1, 0] \\
\text{else} & \ [1, 1, 1] \end{align*}
\]

How many coin flips?
- 3 bits: 7 flips
- 32 bits: 4,294,967,295 flips
- k bits: \(2^k-1\) flips ✗

see GuBPI, AQUA, etc.
Bit Blasting the Uniform

\[ X \sim 0. b_1 b_2 b_3 \]

**Bit Blast**

\[
\begin{align*}
a &= \text{flip}(0.5) \\
b &= \text{flip}(0.5) \\
c &= \text{flip}(0.5) \\
[a, b, c]
\end{align*}
\]

How many coin flips?
- 3 bits: 3 flips
- 32 bits: 32 flips
- \( k \) bits: \( k \) flips
Essence of Bit Blasting

Which continuous distributions can be bit blasted?
Bit Blasting the Exponential $\lambda e^{-\lambda x}$

$X \sim 0. b_1 b_2 b_3$

... represent bits using a probabilistic program of coin flips ...

Bit Blasting the Exponential $\lambda e^{-\lambda x}$

$X \sim 0. b_1 b_2 b_3$

<table>
<thead>
<tr>
<th>Bit Blast</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a = \text{flip}(0.1824)$</td>
</tr>
<tr>
<td>$b = \text{flip}(0.3208)$</td>
</tr>
<tr>
<td>$c = \text{flip}(0.4073)$</td>
</tr>
</tbody>
</table>

$[a, b, c]$

How many flips? k bits: k flips

Cannot go beyond the exponential with just independent coins!

Bit Blasting the Gamma \[ \frac{1}{\Gamma(k)\theta^k} x^{k-1} e^{-x/\theta} \]

\[ X \sim 0. b_1 b_2 b_3 \]

... represent bits using a probabilistic program of coin flips ...
A Purely Continuous Gamma $xe^{-3x}$

Continuous Program

$X = \text{exponential}(-3)$

$Y = \text{uniform}(0, 1)$

observe($Y < X$)

return $X$
Bit Blasting the Gamma

\[ X \sim 0.b_1b_2b_3 \]

**Bit Blast**

\[
X = \text{bitblast} \left( \text{exponential}(-3) \right) \\
Y = \text{bitblast} \left( \text{uniform}(0, 1) \right) \\
\text{observe}(Y < X) \\
\text{return } X
\]
Bit Blasting the Gamma

\[ X \sim 0. b_1 b_2 b_3 \]

Bit Blast

\[ X = \text{bitblast}(\text{exponential}(-3)) \]
\[ Y = \text{bitblast}(\text{uniform}(0, 1)) \]
\[ \text{observe}(Y < X) \]

return X

Wrong
Bit Blasting the Gamma

\[ X \sim 0. b_1 b_2 b_3 \]

**Bit Blast**

\[ X = [\text{flip}(0.182), \text{flip}(0.320), \text{flip}(0.407)] \]
\[ Y = [\text{flip}(0.5), \text{flip}(0.5), \text{flip}(0.5)] \]
\[ Z = [\text{flip}(0.182), \text{flip}(0.320), \text{flip}(0.407)] \]

\( \text{observe}(Y < X) \)
\( \text{return } (\text{if } \text{flip}(0.208) \text{ then } Z \text{ else } X) \)

How many coin flips?
- 3 bits: 10 flips
- 32 bits: 97 flips
- \( k \) bits: \( 3k + 1 \) flips

\[ \text{pdf}(x) \]

\[ \text{pr}(x) \]
Paper shows more:

- Efficient bit blasting for other common continuous distributions
- **HyBit** system for hybrid probabilistic programming [https://github.com/Tractables/Dice.jl/tree/hybit](https://github.com/Tractables/Dice.jl/tree/hybit)
- Supports scalable probabilistic inference in Dice (core language guarantees BDDs of size $O(poly(k))$)
- Comprehensive evaluation on suite of hybrid programs:
  - HyBit supports all benchmarks
  - Gets the best accuracy on 11 out of 19 of them
- Check out our paper: [https://dl.acm.org/doi/10.1145/3656412](https://dl.acm.org/doi/10.1145/3656412)