# Bit Blasting Probabilistic Programs

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

FPTalks - July 11 2024

# What are Probabilistic Programs?

Programs that represent **probability distributions**:

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

Primary analysis task is **probabilistic inference:**

$$
\begin{aligned}
pr(b) &= \sum_{a_i} pr(a = a_i) pr(b|a = a_i) \\
&= \frac{7}{10} pr(b|a = 1) + \frac{3}{10} pr(b|a = 0) \\
&= \frac{7}{10} e^{-\frac{1}{2}b^2} + \frac{3}{10} e^{-\frac{1}{2}(2-b)^2}
\end{aligned}
$$

# What are Probabilistic Programs?

Programs that represent **probability distributions**:

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

Primary analysis task is **probabilistic inference:**

$$pr(b) = \sum_{a_i} pr(a = a_i) pr(b|a = a_i)$$

$$= \frac{7}{10} pr(b|a = 1) + \frac{3}{10} pr(b|a = 0)$$

$$= \frac{7}{10} e^{-\frac{1}{2}b^2} + \frac{3}{10} e^{-\frac{1}{2}(2-b)^2}$$

discrete + continuous = **hybrid** probabilistic program

# Hybrid is Not Well Supported

| Method | Systems | Issue |
| --- | --- | --- |
| Hamiltonian Monte Carlo | Pyro, Stan | Limited support for discreteness |
| Sequential Monte Carlo | WebPPL | Scalability and accuracy issues |
| Algebraic Evaluation | PSI SOLVER | Closed form does not always exist |
| Knowledge Compilation | Dice, ProbLog | No support for continuous |

# Bit Blasting a Continuous Random Variable

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

- ✅ all random variables are discrete
- ✅ representation is exact
- ✅ exposes useful structure (e.g., arithmetic)
- ❌ infinite number of random bits

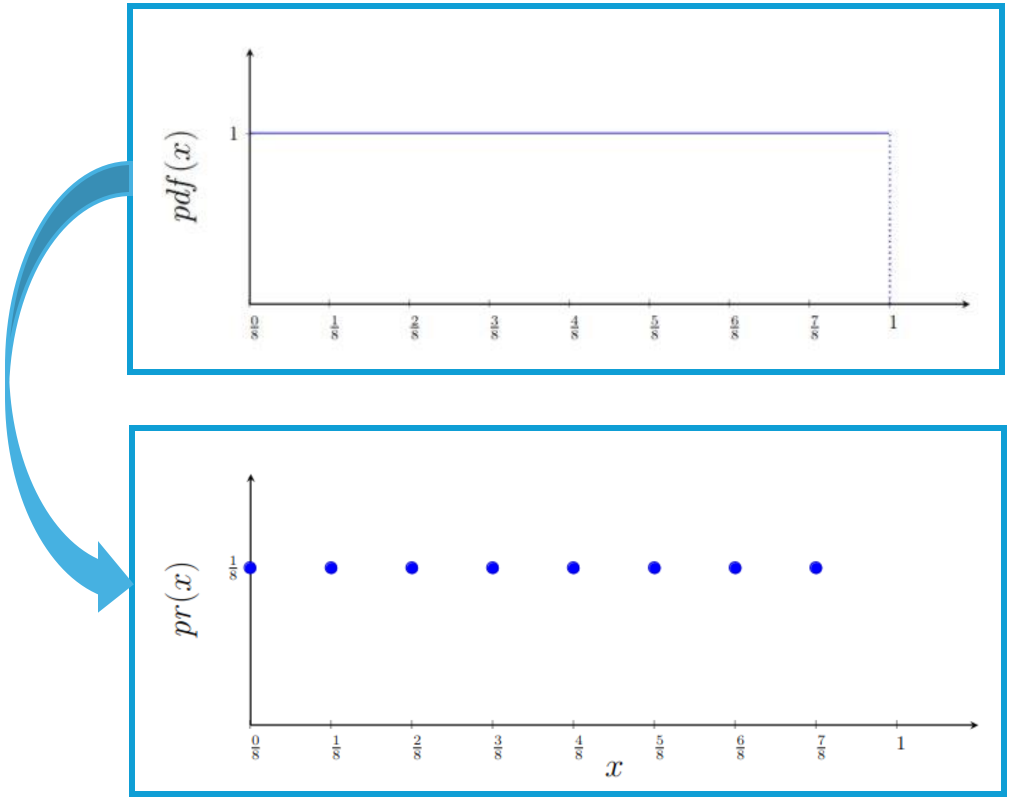
# Bit Blasting a Continuous Random Variable

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

- ✅ 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*

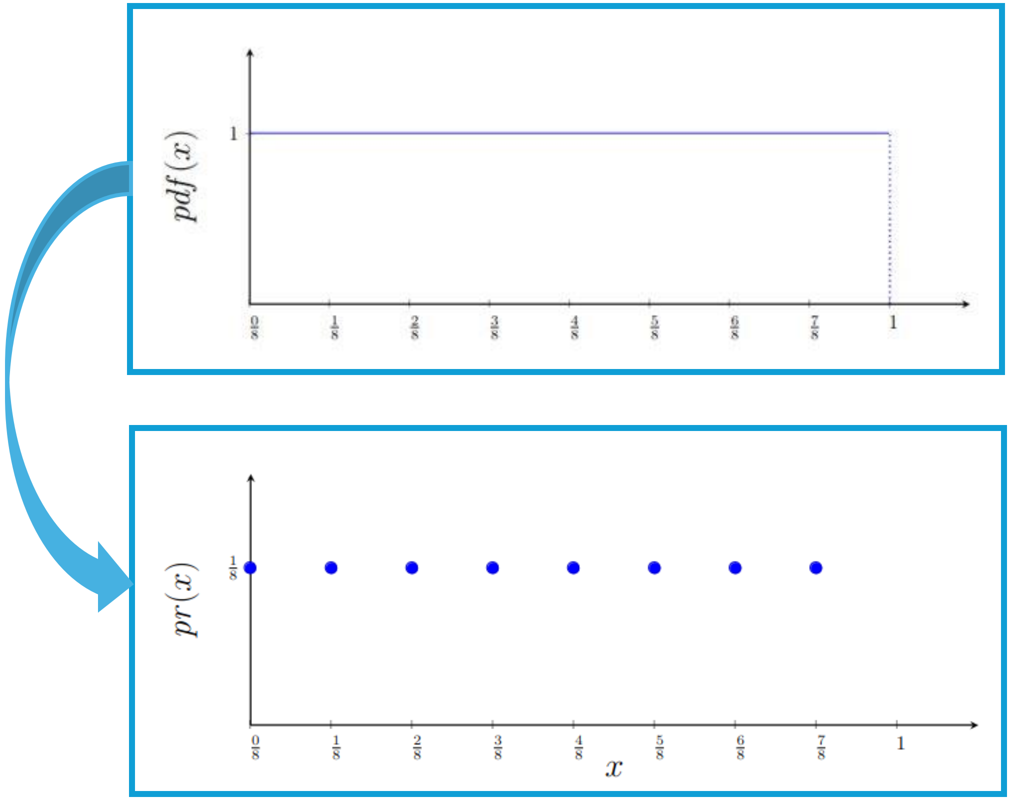
*a probabilistic program*

*of coin flips*

...

# Bit Blasting the Uniform

$X \sim 0.\,b_1 b_2 b_3$




**Naive discretization**

```
if flip(1/8) [0, 0, 0]
elif flip(1/7) [0, 0, 1]
elif flip(1/6) [0, 1, 0]
elif flip(1/5) [0, 1, 1]
elif flip(1/4) [1, 0, 0]
elif flip(1/3) [1, 0, 1]
elif flip(1/2) [1, 1, 0]
else [1, 1, 1] end
```

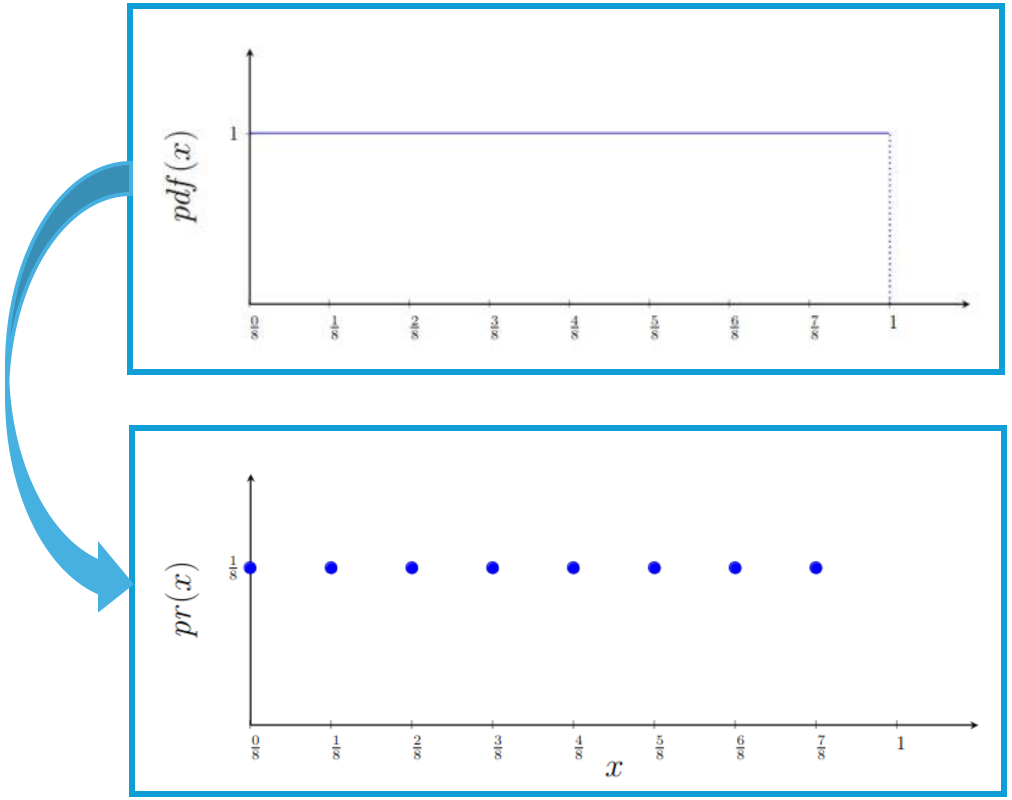
How many coin flips?

- 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

$X \sim 0.b_1 b_2 b_3$



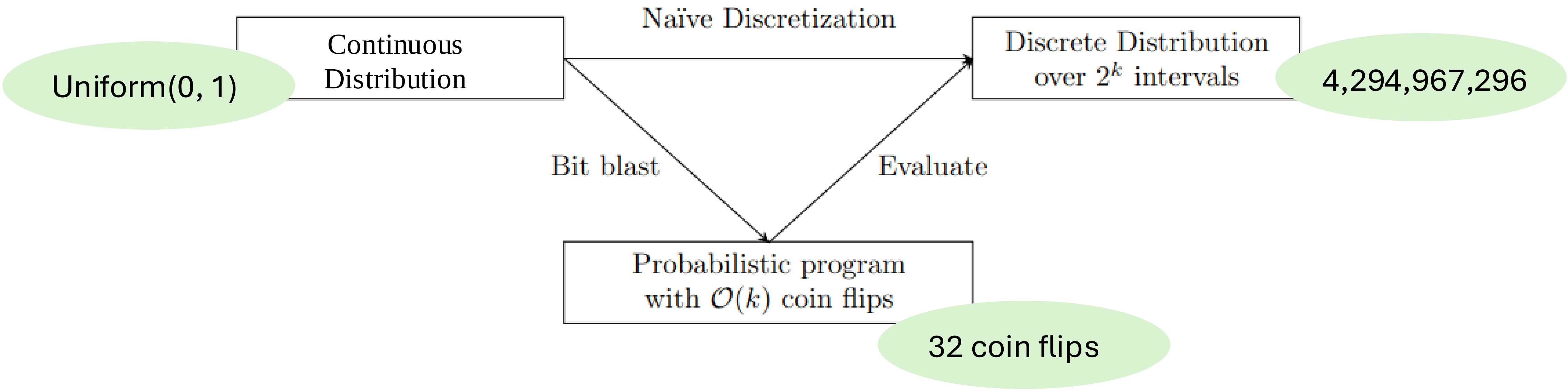

**Bit Blast**

```
a = flip(0.5)
b = flip(0.5)
c = flip(0.5)

[a, b, c]
```

How many coin flips?

- 3 bits: 3 flips
- 32 bits: 32 flips
- k bits: k flips ✅

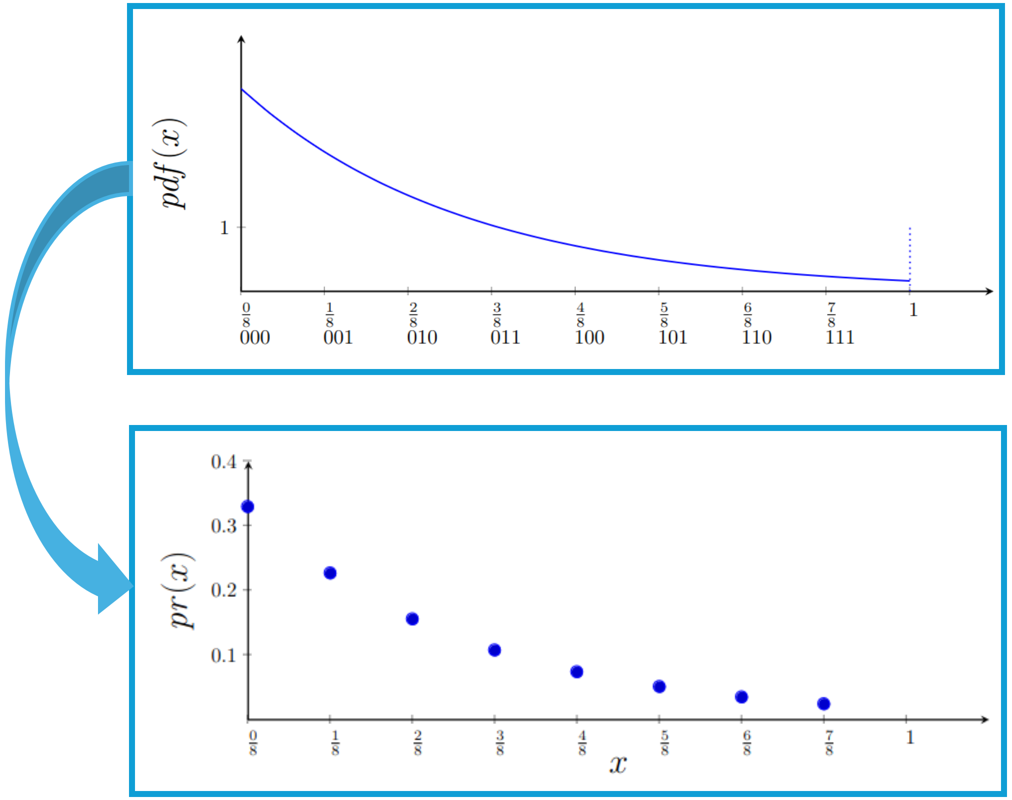
# Essence of Bit Blasting




*Which continuous distributions can be bit blasted?*

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

$X \sim 0.b_1 b_2 b_3$




…

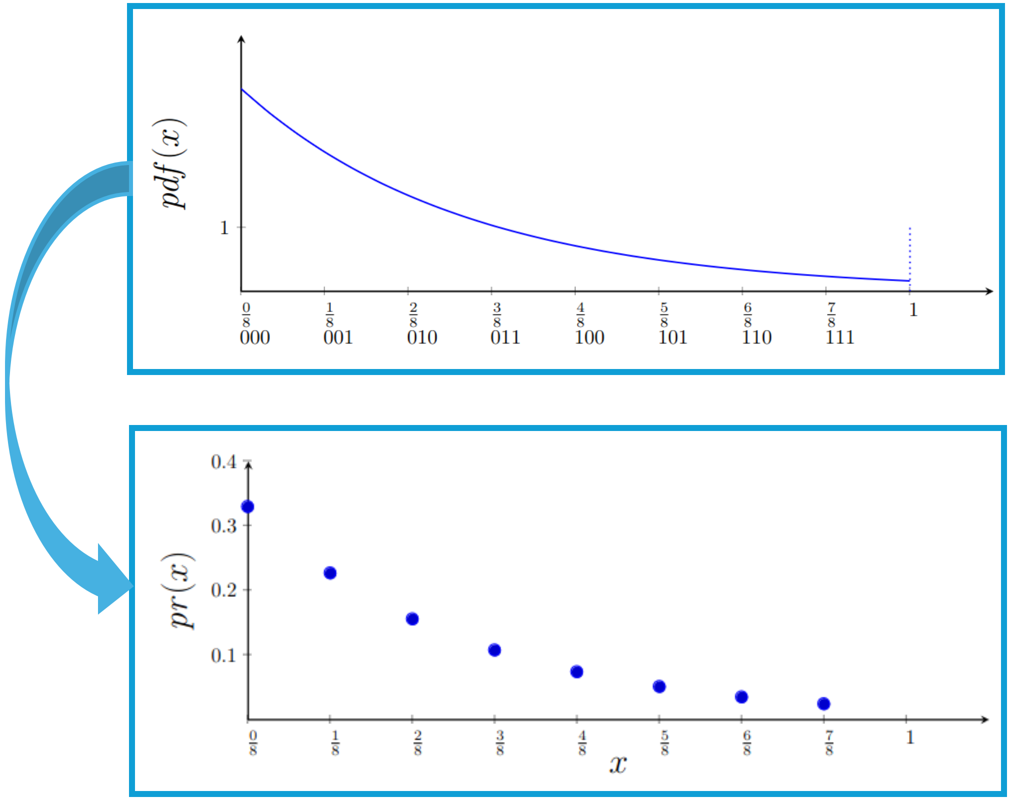
*represent bits using a probabilistic program of coin flips*

…

George Marsaglia. "Random Variables with Independent Binary Digits." Ann. Math. Statist. 42 (6) 1922 - 1929, December, 1971. https://doi.org/10.1214/aoms/1177693058

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

$X \sim 0.b_1 b_2 b_3$




**Bit Blast**

```
a = flip(0.1824)
b = flip(0.3208)
c = flip(0.4073)

[a, b, c]
```

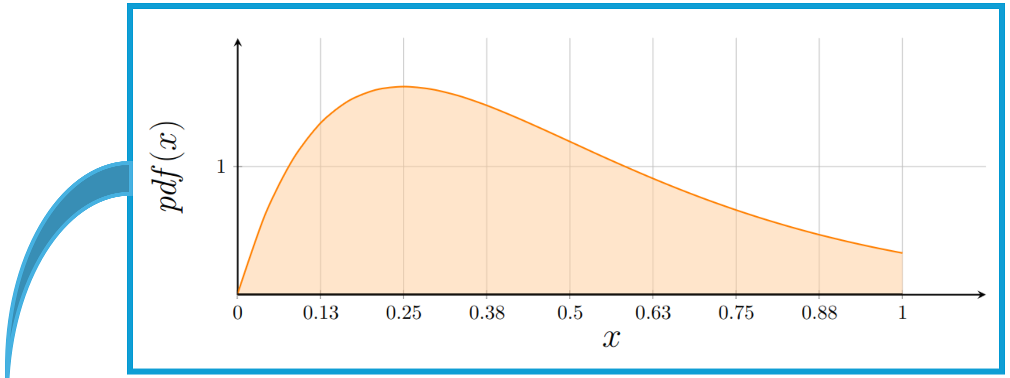
How many flips? k bits: k flips ✅

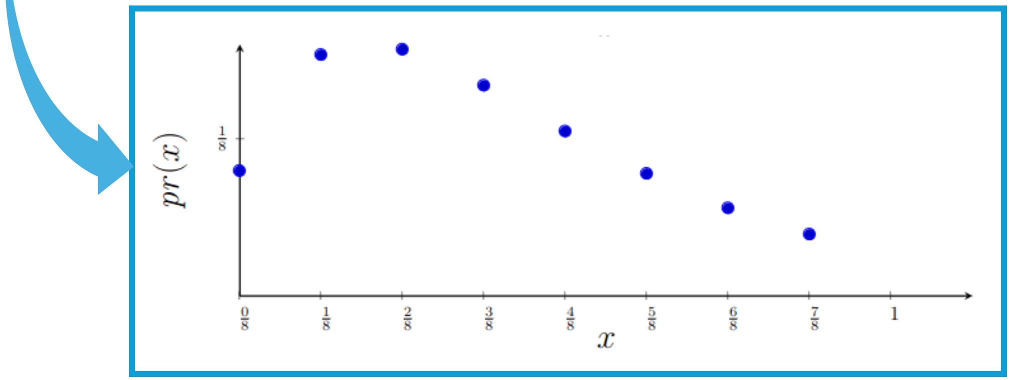
Cannot go beyond the exponential with just independent coins!

George Marsaglia. "Random Variables with Independent Binary Digits." Ann. Math. Statist. 42 (6) 1922 - 1929, December, 1971. https://doi.org/10.1214/aoms/1177693058

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

$X \sim 0.b_1 b_2 b_3$



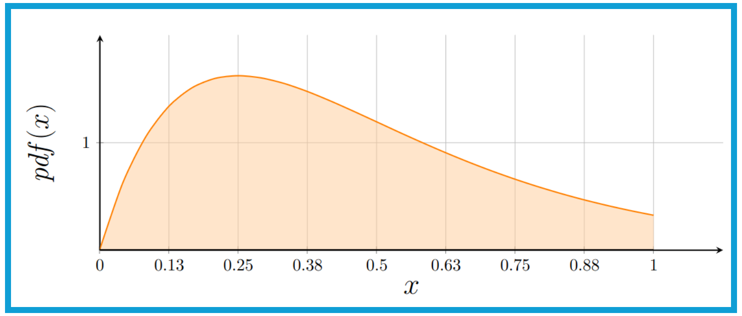




...

*represent bits using a probabilistic program of coin flips*

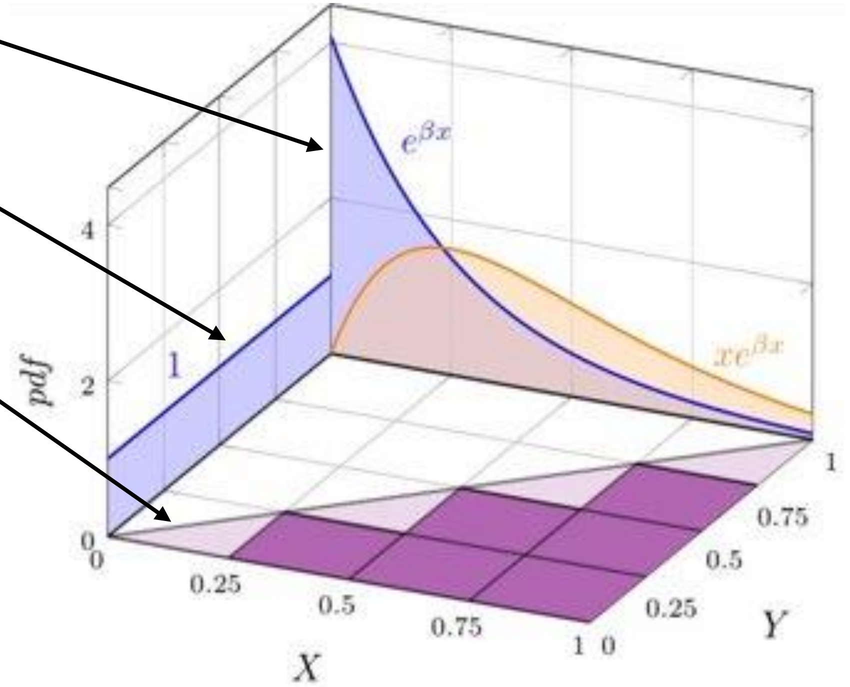
...

# A Purely Continuous Gamma $xe^{-3x}$



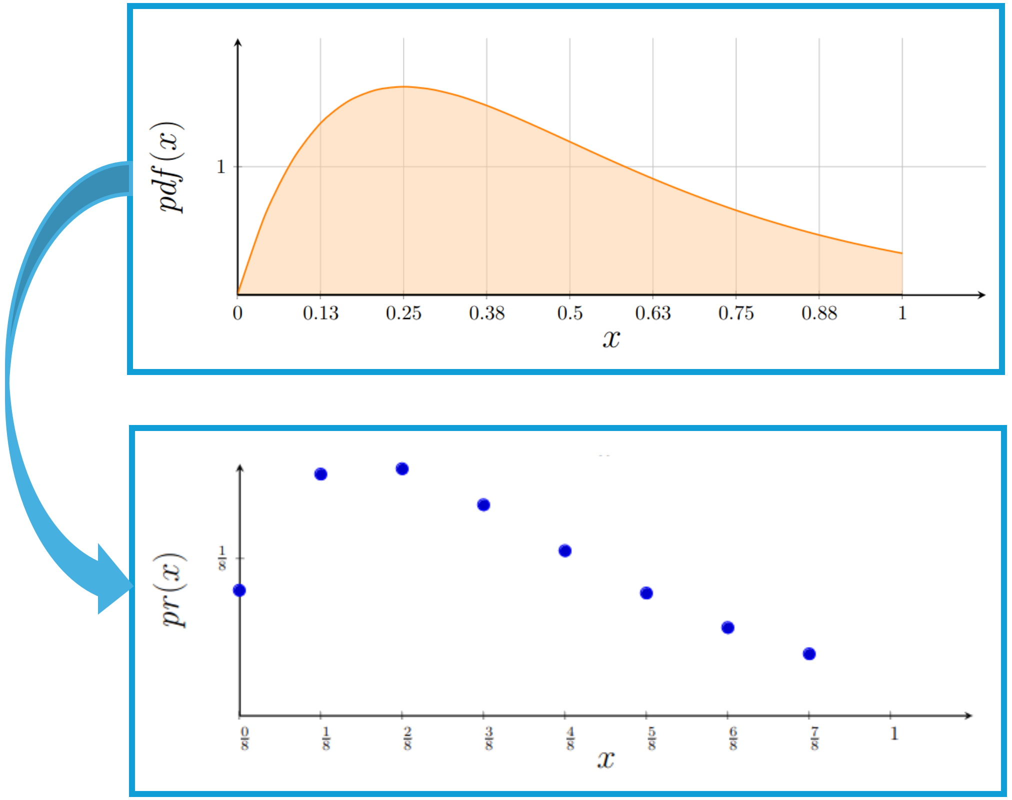

**Continuous Program**

```
X = exponential(-3)
Y = uniform(0, 1)
observe(Y < X)
return X
```



# Bit Blasting the Gamma

$X \sim 0.b_1 b_2 b_3$

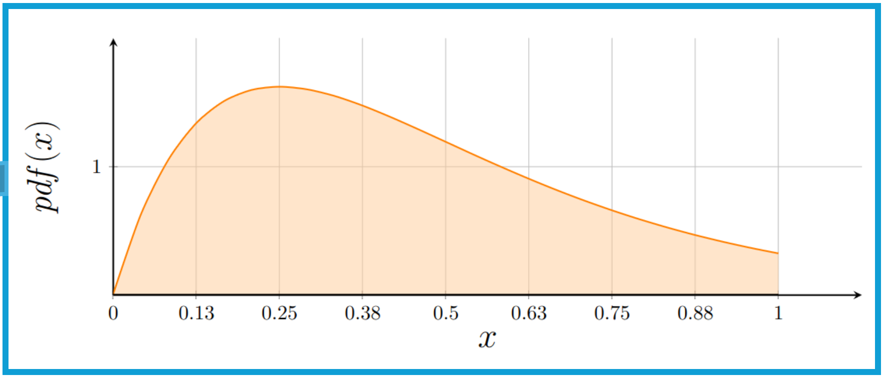


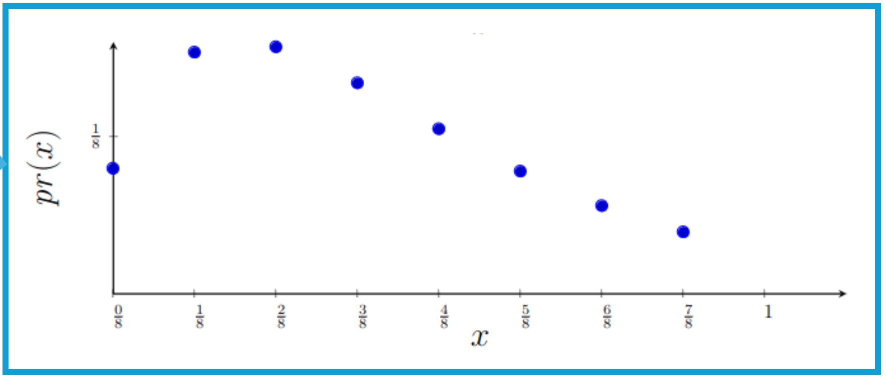

**Bit Blast**

```
X = bitblast(exponential(-3))
Y = bitblast(uniform(0, 1))
observe(Y < X)

return X
```

# Bit Blasting the Gamma

$X \sim 0.b_1b_2b_3$

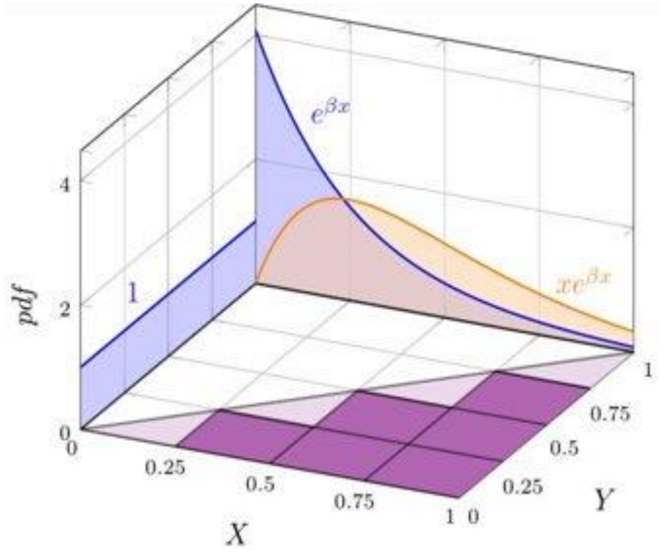



**Bit Blast**

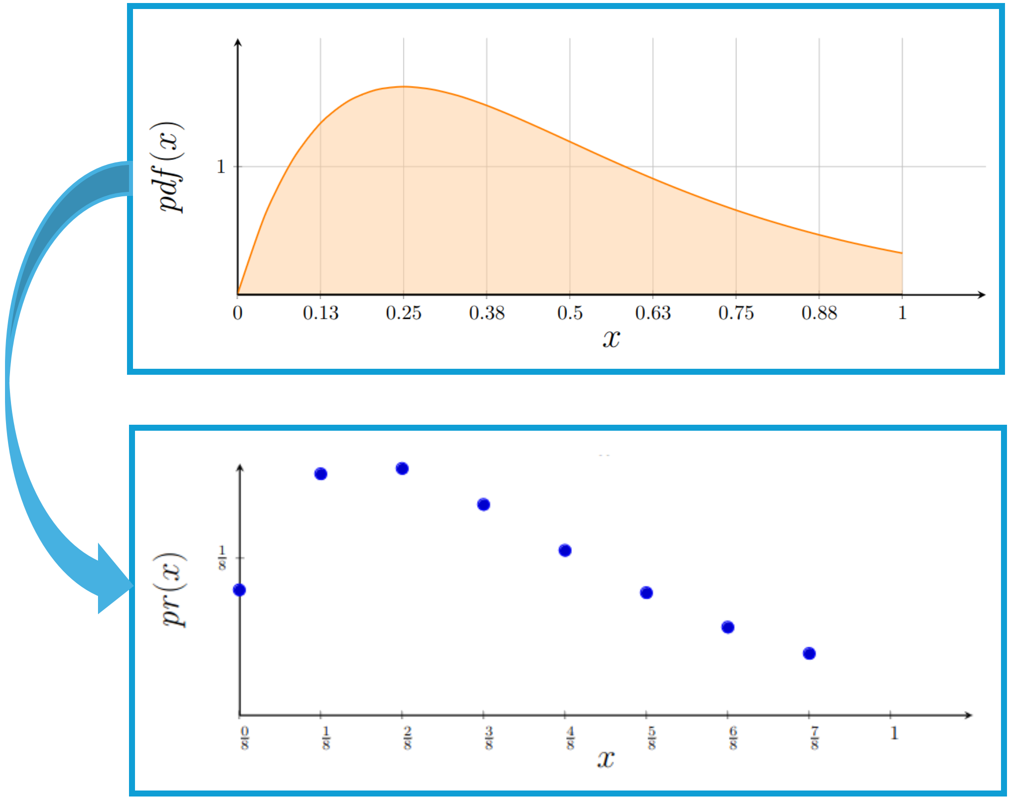
```
X = bitblast(exponential(-3))
Y = bitblast(uniform(0, 1))
observe(Y < X)

return X
```

**Wrong**



# Bit Blasting the Gamma

$X \sim 0.b_1 b_2 b_3$




**Bit Blast**

```
X = [flip(.182), flip(.320), flip(.407)]
Y = [flip(.5), flip(.5), flip(.5)]
Z = [flip(.182), flip(.320), flip(.407)]

observe(Y < X)
return (if flip(0.208) then Z else X)
```

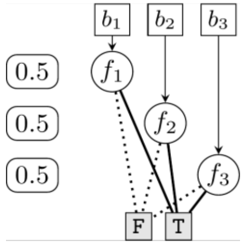
How many coin flips?

- 3 bits: 10 flips
- 32 bits: 97 flips
- k bits: 3k+1 flips ✅

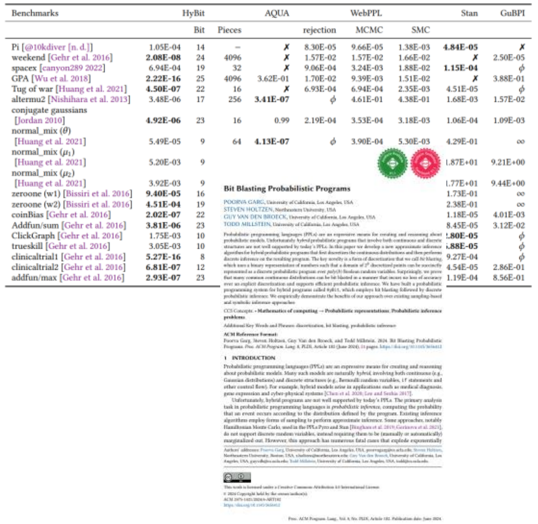
# Paper shows more:

| Distribution | Density | Distribution | Density |
|---|---|---|---|
| Uniform | $1$ | Gamma | $\frac{\beta^\alpha x^{\alpha-1} e^{-\beta x}}{\Gamma(\alpha)}$ |
| Linear | $x$ | Laplace | $\frac{1}{2b} e^{\frac{-\lvert x-\mu \rvert}{b}}$ |
| Polynomial | $x^n$ | Chi-squared | $\frac{1}{2^{\frac{k}{2}} \Gamma(\frac{k}{2})} x^{\frac{k}{2}-1} e^{\frac{-x}{2}}$ |
| Exponential | $\lambda e^{-\lambda x}$ | Student-T | $c(1+\frac{x^2}{\nu})^{-\frac{\nu+1}{2}}$ |

- Efficient bit blasting for other common continuous distributions
- ***HyBit*** system for hybrid probabilistic programming 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