Probabilistic & Approximate Computing

Sasa Misailovic
UIUC
In the previous episode...
But What About This?

\[
X := \text{Uniform}(0,1) \\
Y := \text{Uniform}(0,1) \\
Z := X + Y \\
\text{return } Z
\]

\[
\begin{align*}
2 - Z & \quad 1 \leq Z \leq 2 \\
Z & \quad 0 \leq Z < 1
\end{align*}
\]

$\psi$ sum_uniform.prb
This Episode: Probabilistic Programs

Extend Standard (Deterministic) Programs

Distribution  
X := Uniform(0, 1);

Assertion  
assert ( X >= 0 );

Observation  
observe ( X >= 0.5 );

Query  
return X;
Example Language:

WWW.WEBPPL.ORG
2.1. Basic definition.

We define a probability triple or (probability) measure space or probability space to be a triple $(\Omega, \mathcal{F}, P)$, where:

- the sample space $\Omega$ is any non-empty set (e.g. $\Omega = [0, 1]$ for the uniform distribution considered above);
- the $\sigma$-algebra (read “sigma-algebra”) or $\sigma$-field (read “sigma-field”) $\mathcal{F}$ is a collection of subsets of $\Omega$, containing $\Omega$ itself and the empty set $\emptyset$, and closed under the formation of complements* and countable unions and countable intersections (e.g. for the uniform distribution considered above, $\mathcal{F}$ would certainly contain all the intervals $[a, b]$, but would contain many more subsets besides);
- the probability measure $P$ is a mapping from $\mathcal{F}$ to $[0, 1]$, with $P(\emptyset) = 0$ and $P(\Omega) = 1$, such that $P$ is countably additive as in (1.2.3).
Probability Refresher

Probability Distribution

- Discrete Distributions
- Continuous Distributions
- Hybrid Joint Distributions
Distribution Function

Probability Distribution Function

Probability Mass Function

Probability Density Function
Example Codes

Assigning the probabilities to a fair dice

```javascript
var outcomes = [1, 2, 3, 4, 5, 6];
var probabilities =
    mapN( function (x) {return 1/outcomes.length} ,
          outcomes.length
    )

probabilities
```
Example Codes

**Normalizing the distribution of an unfair dice**

```javascript
var unnormalized = [3, 1, 1, 1, 1, 1];
var normalize = function(lst) {
    var Z = sum(unnormalized)
    map( function(x) {x / Z; }, unnormalized )
}

probability = normalize(unnormalized)
print(probability)
```
Example Codes

Normalizing the distribution of an unfair dice

```javascript
var outcomes = [1, 2, 3, 4, 5, 6]
var unnormalized = [1, 1, 3, 1, 1, 1];
var probability = normalize(unnormalized)

var event_probability = function(predicate) {
    var selected_elements = filter(predicate, outcomes)
    return sum(map(function(idx) {probability[idx-1]}, selected_elements))
}

event_probability(function(x) {return x <= 3})
```
Sampling and Enumerating

...// same as before
var dist = Categorical({ps: probability, vs: outcomes})

// direct sampling (change the number of elements to sample)
var res1 = repeat(100, function() { sample(dist) })
viz.auto(res1)

// enumeration
var res2 = Infer(
    {method: 'enumerate', maxExecutions: 100},
    function () {return sample(dist)})
    viz.auto(res2)
Probabilistic Programs and Graphical Models

\[
\begin{align*}
X & := \text{Uniform}(0,1) \\
Y & := \text{Uniform}(0,1) \\
Z & := X + Y \\
\text{return } Z
\end{align*}
\]
Bayes’ Rule

Belief Revision

Thomas Bayes
1701 – 1761
Bayes’ Rule

\[ \Pr(\theta \mid x) = \frac{\Pr(x \mid \theta) \cdot \Pr(\theta)}{\Pr(x)} \]
Bayes’ Rule

**Belief Revision**

Posterior Distribution

\[ \Pr(\theta \mid x) = \frac{\Pr(x \mid \theta) \cdot \Pr(\theta)}{\Pr(x)} \]

Prior Distribution

Likelihood

Normalization Constant
Is Our Brain Statistical?*

Probability of sickness is 1%

If a patient is sick, the probability that medical test returns positive is 80% (true positive)

If a patient is not sick, the probability that medical test returns positive is 9.6% (false positive)

For a given patient, the test returned positive.

What is the probability that the patient is ill?

* Kahneman and Tversky (1974)
Is Our Brain Statistical?

```javascript
var test_effective = function() {
  var PatientSick = flip(0.01);

  var PositiveTest =
    PatientSick? flip(0.8): flip(0.096);

  condition (PositiveTest == true);

  return PatientSick;
}

Infer ({method: 'enumerate'},
        test_effective)

Fallacy: Base rate neglect

For discussion: Goodman & Tenenbaum, Probabilistic Models of Cognition (Ch. 3)
Bayes’ Rule

Belief Revision

Posterior Distribution

Pr(θ | x) = \frac{Pr(x | θ) \cdot Pr(θ)}{Pr(x)}

Prior Distribution

Likelihood

Normalization Constant
Bayes’ Rule

Belief Revision

\[ \theta \equiv \text{sick} \]
\[ x \equiv \text{test} \]

\[
\Pr(\theta \mid x) = \frac{\Pr(x \mid \theta) \cdot \Pr(\theta)}{\Pr(x)}
\]
Bayes’ Rule

Belief Revision

\[ \Pr(\text{sick} \mid \text{test}) = \frac{\Pr(\text{test} \mid \text{sick}) \cdot \Pr(\text{sick})}{\Pr(\text{test})} \]

\[ \sum_{\text{sick} \in \{T,F\}} \Pr(\text{test}, \text{sick}) \]

(law of total probability)
Bayes’ Rule

Belief Revision

\[
\Pr(\text{sick} \mid \text{test}) = \frac{\Pr(\text{test} \mid \text{sick}) \cdot \Pr(\text{sick})}{\Pr(\text{test})} 
\]

\[
\sum_{\text{sick} \in \{T,F\}} \Pr(\text{test} \mid \text{sick}) \cdot \Pr(\text{sick})
\]
Bayes’ Rule

\( \Pr(sick \mid test) = \frac{\Pr(test \mid sick) \cdot \Pr(sick)}{\Pr(test)} \)

\( \Pr(test \mid sick = T) \cdot \Pr(sick = T) + \Pr(test \mid sick = F) \cdot \Pr(sick = F) \)
**Bayes’ Rule**

*Belief Revision*

\[
\Pr(sick \mid test) = \frac{\Pr(test \mid sick) \cdot 0.01}{\Pr(test)}
\]

\[
\Pr(test\mid sick = T) \cdot 0.01 + \Pr(test\mid sick = F) \cdot 0.99
\]
Bayes’ Rule

**Belief Revision**

\[
\operatorname{Pr}(sick \mid test) = \frac{0.80 \cdot 0.01}{\operatorname{Pr}(test)}
\]

\[
0.800 \cdot 0.01 + 0.096 \cdot 0.99
\]
Bayes’ Rule

\[ Pr(\theta | x) = \frac{Pr(x | \theta) \cdot Pr(\theta)}{Pr(x)} \]

Posterior Distribution

Prior Distribution

Likelihood

Normalization Constant
Bayes’ Rule

**Belief Revision**

Posterior Distribution

\[
Pr(\theta \mid x) \sim Pr(x \mid \theta) \cdot Pr(\theta)
\]

Likelihood

Prior Distribution

Enough to order different interpretations and select the most likely one.