Overview

- We investigate the statistical implications of nesting probabilistic programs
- Nesting programs allows definition of models which could not otherwise be expressed, e.g. experimental design, reasoning about reasoning [1]
- Changes are required to existing systems to ensure consistent estimation
- We delineate possible nesting methods and assert their respective correctness using recent results from nested Monte Carlo estimation [2, 3]

Take Home

- Observing the output of another query (i.e. program) is statistically sound, though semantically challenging
- Sampling from the conditional distribution of one query inside another is semantically straightforward but statistically problematic;
- Convergence is possible but requires additional assumptions/precautions
- Number of samples used in each call of a nested query must \( \to \infty \)
- Convergence rate is very slow:
  \( \frac{\text{number of samples used in } k_{\text{th}} \text{ layer}}{\text{number of samples used in } 0_{\text{th}} \text{ layer}} \to 1 \)
  
  Consistency of sampling from nested queries follows from (4), but only if
  
  \( k_{\text{th}} \text{ layer} \) giving asymptotic bias

- Consistency of sampling from nested queries follows from (4), but only if
  
  \( k_{\text{th}} \text{ layer} \) giving asymptotic bias

- Estimation of sampling from nested queries follows from (4), but only if
  
  \( k_{\text{th}} \text{ layer} \) giving asymptotic bias

- As conditional returns unweighted samples, Anglican is unable to catch the special cases

Empirical Demonstration of Convergence of NMC

- Simple analytic model from [2]
- (Red) If \( N_k = 5 \) is fixed then doesn’t converge
- (Blue) Setting \( N_0 \approx N_1 \approx N_2 \) gives slow convergence
- (Green) Setting \( N_0 \approx N_2 \approx N_2 \) gives faster convergence

Special Cases

- Discrete expectations can be collapsed through enumeration
- Linear \( f_1 \) can be collapsed, e.g. for linear \( f_1 \) and \( D = 1 \)
  
  \( E \left[ f_0 \left( y^{(0)} \right) E \left[ f_1 \left( y^{(1)} \right) \mid y^{(0)} \right] \right] = E \left[ f_0 \left( y^{(0)} \right) f_1 \left( y^{(1)} \right) \right] \) \hspace{2cm} (5)

- Products of expectations can be combined to a single expectation over a distribution defined by the product of the marginal distributions

- These latter two demonstrate consistency of nested sampling when inputs to the inner query are deterministic and proper weighting is used

Nested Sampling

Consider the following nested Anglican [4] model

\[
{\text{(defm prior [I [normal 0 1])}} \quad \text{(defquery outer-q [d M]})
\]

\[
\text{(let [z (sample (gamma y 1))]} \quad \text{(let [y (sample (beta 2 3))]}
\]

\[
\text{(observe (normal y z) D) \quad \text{z (inner y D))}}
\]

\[
\text{(z) \quad (+ y z))}
\]

for which outer defines the distribution

\[
\tilde{\pi}(z, y, D) = \frac{p(y)p(z)p(D|y, z)}{p(y)p(z)p(D|y, z) \neq \tilde{\pi}(z, y, D)}
\]

What went wrong?

- Consistency of sampling from nested queries follows from (4), but only if every \( N_k \to \infty \)
- Conditional implicitly uses a fixed \( N_k \) giving asymptotic bias
- As conditional returns unweighted samples, Anglican is unable to catch the special cases

Nested Observation

- Instead of sampling from another query we might want to condition on it giving a certain output
- Statistical correctness follows directly from the linearity and products of expectations special cases given in [2]
- Can first of such cases as defining pseudo-marginal approaches
- There are still substantial semantical difficulties – in general it is not possible to evaluate the density on program outputs

Estimates as Variables

- One might wish to use estimates as variables in another program
- This allows arbitrary nested estimation problems to be encoded, but correctness must be assessed on a case-by-case basis

\[
{\text{(defm prior [I [normal 0 1])}} \quad \text{(defquery outer-q [d M]})
\]

\[
\text{(let [theta (sample (prior))] \quad y (sample (lik theta d))}
\]

\[
\text{log-lik (observe* log-marg (inner theta d y) \quad log-marg (inner theta d y))}
\]

\[
\text{(dquery :importance inner-q [y d]) \quad (take M) \quad log-marginal())}
\]

\[
\text{(dquery :importance outer-q [y d]) \quad (take M) \quad collect-results empirical-mean())}
\]

References


