Bayesian Logic (BLOG) Probabilistic Modeling Language

karl@cs.tufts.edu

1 Modeling

- BLOG models specify the structure and contents of *possible worlds*.
- Define mutually recursive `random` and `fixed` functions.

2 Evidence

- Can make arbitrary propositions about the model - propositional conditioning:

```
obs (x :: Proposition)
```

3 Query

- Given any number of observations, can subsequently query the distribution of values for a particular function or expression in the model:

```
query (A (a1, ..., a_n))
```

4 Distribution Representation

- Java classes corresponding to specific kinds of distributions
- Logarithmic probabilities stored in real-valued data type / Java objects
- Users can '*easily*' add new Java distribution classes
- Output representation is distribution over arbitrary type, e.g.:

```
========  LW Trial Stats =========
Log of average likelihood weight (this trial): -3.4266215829983153
Average likelihood weight (this trial): 0.032496542400000314
Fraction of consistent worlds (this trial): 1.0
Fraction of consistent worlds (running avg, all trials): 1.0
======== Query Results =========
Number of samples: 10000
Distribution of values for Weather(@6)
  Dry 0.7168517226620216
  Rainy 0.2831482773379617
Distribution of values for RainyRegion
  true  0.7680060879338363
  false 0.23199391206616182
======== Done ========
```

5 Language Strengths

- Java EDSL means Java developers can easily interface arbitrary code into the engine
- Familiar object-oriented style (because of possible worlds), but with necessary functional aspects to make models more readable and concise.

6 Language Weaknesses

- Modularity
- Poorly documented & designed grammar - could have abstracted away a core set of probabilistic features and then called out to e.g. a Java grammar specification for the general-purpose language constructs

```
random Boolean Burglary ~ BooleanDistrib(0.001);

random Boolean Earthquake ~ BooleanDistrib(0.002);

random Boolean Alarm ~
  if Burglary then
    if Earthquake then BooleanDistrib(0.95)
    else  BooleanDistrib(0.94)
  else
    if Earthquake then BooleanDistrib(0.29)
    else BooleanDistrib(0.001);

random Boolean JohnCalls ~
  if Alarm then BooleanDistrib(0.9)
  else BooleanDistrib(0.05);

random Boolean MaryCalls ~
  if Alarm then BooleanDistrib(0.7)
  else BooleanDistrib(0.01);

obs JohnCalls = true;
obs MaryCalls = true;

query Burglary;
```

7 Memorable

- Metropolis-Hastings sampling
- Intuitive runtime system front-end

8 References

- BLOG syntax & semantics - https://bayesianlogic.github.io/download/blog-langref.pdf