A probabilistic programming language in OCaml

Anik Roy, Christ's College, ar899@cam.ac.uk

January 21, 2020

Project Supervisor: Dr R. Mortier

Director of Studies: Dr R. Mortier

Project Overseers: Dr Jon A. Crowcroft & Dr Thomas Sauerwald

Work Completed

So far, I have managed to complete the bulk of my core implementation. This includes designing a simple DSL, which is shallowly embedded in OCaml. I have designed the main data structures for representing distributions, and defined the ways in which they can be manipulated and composed. This allows for generative models to be written using my ppl, which can represent both continuous or discrete distributions.

I have implemented a set of primitive distributions (e.g. categorical, normal, uniform, beta, etc.) which can be used to build up more complex models.

I have also implemented 5 main inference algorithms (more than single one required by my proposal) and checked they all function correctly and produce sensible posterior distributions. I have also written a set of example models to show how programs using the language will look. These models will also be used for evaluation. There have been no major unexpected difficulties, and I am on schedule as in my proposal.

Next Steps

The main next steps for this project will be to perform more evaluation, comparing against writing inference procedures by hand (e.g. in python), as well as against other ppls (such as webppl or anglican). I'll also work to optimise inference and sampling procedures to make sure they are comparable to similar systems

I will also consider more extensions, for example around the visualisation of programs. Since they are represented in a tree structure internally, it should be possible to visualise the different intermediate variables in a graph.