School of Electronic Engineering and Computer Science
Queen Mary University of London

MSc PROJECT DEFINITION
2023-24

This project definition must be undertaken in consultation with your supervisor. The feasibility of the project should have been assessed and the project aims should be clearly defined.
Submission of this document implies that you have discussed the specification with your supervisor.

Project Title: Developing a suite of test problems for evaluating black box probabilistic programming languages.

Supervisor: Dr. Fredrik Dahlqvist

Student name: Patrick Pagni

Student email: ec23035@qmul.ac.uk

PROJECT AIMS:

Libraries and languages following the probabilistic programming paradigm offer a black box solution for developing inference models. By providing prior distributions and a model for the likelihood, these programmes estimate the posterior distribution. In simple cases, it is possible to enumerate all the possible outcomes with their associated probabilities; in cases where the prior distributions are continuous, or multivariate, enumerating all possible outcomes is not feasible. To overcome this, these programming languages implement different sampling techniques, such as Monte Carlo Markov Chains, to estimate the posterior distribution.

Given the sampling methods employed and the black box nature of these implementations, it is important to provide some way to verify the quality of the programming language's outputs. Especially since these programming languages are increasingly being used to implement machine learning models for automated decision-making. This project aims to develop a test suite where a user can submit their probabilistic programming language and are provided a metric or set of metrics evaluating the language's performance.

PROJECT OBJECTIVES:

This project consists of two work streams: one is a theoretical stream, and the other is practical. The theoretical workstream consists of two objectives: the first is to find a suitable set of test problems which provide sufficient coverage over the infinite set of possible problems; the second is to identify a suitable metric for evaluating the quality of the programming language's output.

The practical workstream consists of the following objectives: design a system where programmers can assess a programming language's performance against the test cases defined during the theoretical work stream. This design should include a distribution strategy. This system will initially be implemented for one probabilistic programming language for a prototype. Other languages will be considered for potential future development, but they are not considered as within scope for this project.

State how your project will be aligned with the learning outcomes of your programme of study.

The theoretical workstream will require further study of statistics and set theory. Both of these mathematical fields are foundational for AI development. This project will require extending my studies of applied statistics, taken in semester 1 of the 2023/24 year, by researching different probability distributions and more concepts within probability theory.

The practical workstream will require research into software engineering methods to ensure the programme is usable and fast. Additionally, this project will require the development of a distribution strategy, whether this be via API or a package that users can download.

METHODOLOGY:

Finding a suitable set of test problems which provide sufficient coverage over the infinite set of problems will be investigated by analysing the infinite set and dividing subsets such that one test problem for each subset can act as proxy for all the problems within the set. The initial batch of test cases will be instances where the posterior distribution is the same as that of the prior distribution, since this batch of cases have exact closed-form solutions. Once this has been implemented, this project will seek to identify further test cases, although a complete implementation of all the total set of problems is considered outside of the scope of this project.

The output for a probabilistic programming language is an observation, whereas the input for the test suite will need to be a probability distribution to allow for comparison. One strategy for obtaining the probability distribution used by the test language would be to sample from this found distribution, reconstruct the distribution from the samples and compare this to the expected distribution. Alternatively, it may be possible to extract the probability of the observation using the test language itself. In this case, the test suite can calculate the probability of the observation under the exact posterior distribution and compare this to the probability of the observation found by the test language. The metric used for comparison may be cross entropy loss in the instance where two probability distributions for an observation are being compared.

Development of the software component will begin once the first set of test problems have been calculated. This will serve as the prototype where the approaches for the metrics above will be tested, and the ultimate approach decided upon. Then we will consider further test cases to be added to the test suite.

PROJECT MILESTONES

The deliverables for this project include:

- Paper documenting the problems used for the test set including an analysis of the coverage of the total set;
- System design specification outlining the components of the software, performance assessment and an analysis of where the system can be improved. This design specification will include the scope for the prototype as well;
- System prototype supporting one programming language initially.

REQUIRED KNOWLEDGE/ SKILLS/TOOLS/RESOURCES:

This project will require developing my understanding of statistics, including different methods for calculating exact posterior distributions. This will be required for defining the test problem set. It will also be important to understand the sampling techniques used by probabilistic programming languages, particularly Monte Carlo Markov Chains.

This project will also require me to develop my software engineering abilities, as a key aspect for assessing the performance of the test suite will be speed.

This project should not require any specific hardware accelerators because the system will not be processing much data. Nor should this require an ethics board for the same reason.

TIMEPLAN

Project Phase	Deliverable	Week commencing								
		2024-05-03	2024-05-17	2024-05-31	2024-06-14	2024-06-28	2024-07-12	2024-07-26	2024-08-09	2024-08-23
Research	Paper on the inital test problem set									
	Additions to the initial problem set									
Design	System design specification									
Development	Implementation of system									
Evaluation	Dissertation Paper									
	Reflective Essay									
	Video presentation									