Philipp Gabler, BSc

Automatic Graph Tracking in Dynamic Probabilistic Programs via Source Transformations

Master's Thesis

to achieve the university degree of
Master of Science

submitted to
Graz University of Technology

Supervisor
Univ.-Prof. Dipl.-Ing. Dr. mont. Franz Pernkopf

Co-supervisor
Dipl.-Ing. Martin Trapp, BSc

Institute of Signal Processing and Speech Communication

Faculty of Electrical and Information Engineering

Graz, XXXX 2020

Affidavit

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master‘s thesis.

Date

Signature

The LaTeX source of this document is available at
https://github.com/phipsgabler/master-thesis
or upon request from the author.[1]

[1] pgabler@student.tugraz.at

Abstract

Probabilistic programming is a means of describing stochastic models through the syntax of a programming language. These models describe probability distributions as stochastic function. Probabilistic programms distinguish themselves from normal programs by the possibility of being sampled from conditionally, with some of the internally variables fixed to observed values. While probabilistic programming systems are often implemented as separate, domain-specific languages, they can also be embedded into “host” programming languages with sufficient syntactic flexibility. The latter is advantageous if one wants to use regular general-purpose programming constructs or interact with other functionalities of the host language.

This thesis first presents a general system for extracting rich computation graphs in the Julia programming language. These graphs contain the whole recursive structure of a program in terms of executed instructions of the intermediate representation used by the compiler. The system is flexible enough to be used for multiple purposes that require dynamic program analysis or abstract interpretation, such as automatic differentiation or dependency analysis.

The second contribution is the application of the described graph tracking system to programs written for Turing, a probabilistic programming system implemented as an embedded language within Julia. The functions describing a Turing model can be analyzed, and from them the dependency structure of involved random variables be extracted. Given this structure, analytical Gibbs conditionals can be calculated and passed to Turing’s inference mechanism, where they are used in Markov-Chain Monte Carlo samplers approximating the modelled distribution.

Contents

1 Introduction

1.1 Problem Description

1.2 Related Work

2 Background

Some introduction here. sdf sd sld fslkdjf sldkj sldk sldkfs dfslkd flsdkjf lskdfj

2.1 Bayesian Inference and Probabilistic Programming

2.2 Computation Graphs and Automatic Differentiation

2.3 Metaprogramming and Compilation in Julia

3 Implementation of Dynamic Graph Tracking in Julia

3.1 Automatic Graph Tracking and Extended Wengert Lists

4 Graph Tracking in Probabilistic Models

4.1 Dependency Analysis in Dynamic Models

4.2 JAGS-Style Automatic Calculation of Gibbs Conditionals

4.3 Evaluation

5 Discussion

5.1 Future Work

Colophon

This document was typeset using the pdfLaTeX typesetting system, with the memoir document class. The body text is set in 11 pt Linux Libertine, enhanced by the microtype package. Other fonts include Biolinum and Inconsolata.

The document source has been written in Emacs with AUCTeX mode, using TeXworks as PDF viewer.