Bayesian neural network estimation of next-to-leading-order cross sections

by

René Alexander Ask

THESIS

for the degree of

MASTER OF SCIENCE

Faculty of Mathematics and Natural Sciences
University of Oslo

Autumn 2021

Bayesian neural network estimation of next-to-leading-order cross sections

René Alexander Ask

Bayesian neural network estimation of next-to-leading-order cross sections

http://www.duo.uio.no/

Printed: Reprosentralen, University of Oslo

Abstract

This is my abstract.

Acknowledgments

Acknowledgments yo

Contents

Introduction

Motivation, context and problem.

Outline of the Thesis

Give outline of thesis

Chapter 1

Machine Learning: Preliminaries

Machine learning is a field of study concerned with learning from known observations and of unseen ones. In this thesis, we'll focus on *supervised* machine learning, which is a subfield of machine learning that fits models on data points x with definite targets y. We'll confine ourselves even further and only study *regression* problems, which is a class of problems where the function we're trying to learn produces a continuous output, i.e a function $f : \mathbb{R}^p \to \mathbb{R}$.

1.1 Basic concepts in regression

The basic conceptual framework of a supervised machine learning problem is as follows. Assume a dataset $\mathcal{D}$ built up of n datapoints $(\boldsymbol{x}_i, y_i)$, where $\boldsymbol{x}_i \in \mathbb{R}^p$ is the set of *features* and $y_i \in \mathbb{R}$ is the *target*. I'll introduce a shorthand notation to represent the dataset as $\mathcal{D} = (X, \boldsymbol{y})$ where X is the set of features and $\boldsymbol{y}$ is the set of targets. The next ingredient is to assume the targets are of the form

$$y_i = f(\mathbf{x}_i) + \epsilon_i, \tag{1.1}$$

for some true function $f(\boldsymbol{x}_i)$ (also known as the ground truth), where ϵ_i is introduced to account for random noise. To approximate the outputs y_i, the standard approach is to choose a model class $\hat{f}(\boldsymbol{x}; \boldsymbol{\theta})$ combined with a procedure to choose parameters $\boldsymbol{\theta}$ such that the model is as close to $f(\boldsymbol{x}_i)$ as possible. This typically involves choosing a *metric* $\mathcal{C}$ to quantify the error, usually called a *cost*-function or a *loss*-function, and minimize it with respect to the parameters of the model.

1.1.1 Bias-variance trade-off

From eq. (1.1), we can deduce a general feature of machine learning problems that proves challenging. We cannot directly probe the true function $f(\mathbf{x}_i)$, because only y_i is observed. Because of this, choosing a model class is a delicate process. If the model class is too simple (i.e few parameters $\boldsymbol{\theta}$), it is likely to capture very general features of the ground truth whilst more nuances properties are missed entirely. Then we say that the model has a high bias and a low variance. Increasing the model complexity (i.e increasing number of parameters) allows the model to reproduce a growing number of nook-and-crannies of the data. A model that is too complex is said to have a low bias and a high variance.

Conclusion

Conclusion here.

Appendices

Appendix A

A.1 Appendix 1 title

Some appendix stuff.

Bibliography