Project 2 FYS-STK4155 Autumn 2018

Jon Audun Baar

8. november 2018

1 Introduction

The concept of machine learning have gained a hughe popularity boost over the last couple of years. And this is no wonder, the techniques have a wide range of applications and can be a major asset. That is if you know when to use what.

When to use what is exactly what we're going to have a brief peek into in this project. Two widely used machine learning methods are regression analysis and neural networks. We aim to evaluate the performance of these two methods in two different cases. First we look at the problem of predicting a scalar, then we compare the two methods in the case of a classification problem. In both cases we use the much studied Ising model to generate our dataset.

2 Method

In order to present the methods used, we first need to establish some notation. In the following we assume that we are given a dataset consisting of $N \in \mathbb{N}$ samples, where each sample consists of p predictors and one response. We then denote by x_{ij} the j-th predictor of the i-th sample and by t_i the response of the i-th sample.

For all supervised learning methods the basic concept is the same: You want to fit a model to your trainingdata by minimizing some sort of error. This is of course also the case for regression analysis and neural networks.

2.1 Regression analysis

2.1.1 Linear regression

2.1.2 Logistic regression

2.2 Neural Networks

When dealing with neural networks, the overarching concept is still the same as when one does regression analysis: We have a model that we wish to fit to a dataset by minimizing some error. The difference is the model and the methods by which we minimze the error.

Again, as the scope of this text is not to fully explain neural networks rather than to analyse it's performance in some special cases, we will simply establish the needed notation and assumptions. For a thorough explanation of neural networks consult CITATION.

We will use a sequential multilayered network with L layers as our model. Let the number of nodes in layer l be denoted by N^l and let $N^0 = N$. We then denote by $W^l \in \mathbb{R}^{N^l \times N^{l-1}}$ and $\boldsymbol{b}^l \in \mathbb{R}^{N^l}$ the weight matrix and bias vector of layer l respectively. Finally we denote by $f^l : \mathbb{R}^{N^l} \to \mathbb{R}^{N^l}$ the activation function of layer l. Our model is then given by

$$f^L(\dots f^2(W^2 f^1(W^1 \boldsymbol{x} + \boldsymbol{b}^1) + \boldsymbol{b}^2) + \dots + \boldsymbol{b}^L) \tag{1}$$

2.3 Performance measures

2.3.1 Accuracy score

2.4 The Ising model

3 Implementation

Implementation is done in python.

4 Analysis of the methods

5 Conclusion