Proposal for CS 698: A Residual Sampling Neural Network

Pierre McWhannel, Ren Yuan Xue, and Vincent Luong

March 2020

1 Introduction

The ability to model uncertainty in the neural network will help in combating overfitting since there is assumed to be random error in the data. Instead of having fixed numbers as weights, now weights are sampled from a family of distributions with unknown parameters. Given an input x, we are going to sample weights from the distributions n times and get n number of output y. This will form a distribution which will be split by a threshold into sections (see Figure 1). One side of this split will be used as a residual distribution. This residual distribution will be used to update the parameters of the weights and biases through backpropagation.

Formally, we define the residual distribution as the probability of our output to be within the range of $target\ value \pm threshold$ as follows:

$$Pr(|y - t| \leq threshold) = Residual\ Distribution$$

Where

$$y : model\ prediction,\ t : targets$$

More specifically, in a binary classification problem, we have:

$$y \in [0, 1]\ and\ t \in \{0, 1\}$$

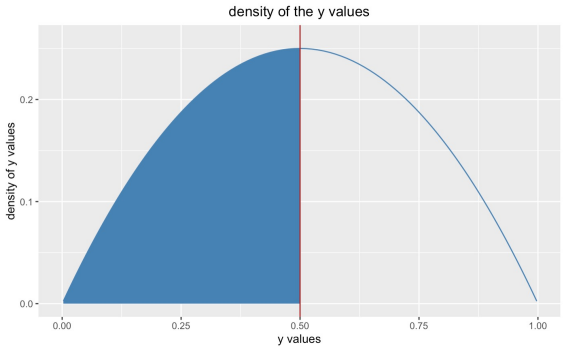

Figure 1: 0.5 threshold, when t = 1 the shaded area is residual distribution

We wish to minimize the probability introduced above. Targets are assumed to be generated in the following manner:

$$Target\ =\ t\ =\ f(x)\ +\ \epsilon$$

Where

$$f(x) : the\ model\ dependent\ on\ input\ covariates\ x$$

$$\epsilon\ : Error$$

2 Challenges

- **High computational cost**. If Monte Carlo methods are needed thereby requiring many samples from prior distributions.
- **Update weights**. Instead of updating a number for weight in traditional neural network, now we need to update the distribution of the weights, which is challenging to come up with a scheme.

3 Algorithm/Model Steps

Select a prior distribution for the weight and biases

1. Initialize the parameters of the priors
2. Run a sample through the network
    - The priors can be sampled at random
    - The operations between priors and inputs can be pushed through the network analytically
3. Formulate a residual distribution according to the threshold
4. Select samples from residual distribution and apply backpropagation
5. Collect a distribution of updates for each weight and bias
6. Fit the updated weights and biases according to these update distributions
7. Repeat steps 3 to 6

4 Methods of Analysis:

Generate simulated data to analyze and investigate how the model avoids overfitting and relation between threshold and irreducible error. Test model on a benchmark dataset such as MNIST and compare with L2 neural networks and feedforward neural network without decay.