

# Neural Net Classification Algorithm

Alexander Heckett

September 5, 2015

## Abstract

This paper provides a relatively in-depth look at the algorithm used behind my neural net classifier. It does not cover the background corresponding to, the progression of ideas that led to, or any results coming from the program implemented.

## Contents

<b>1</b>	<b>The pre-computation</b>	<b>1</b>
1.1	The Neural Net . . . . .	1
1.1.1	Simulated Data . . . . .	1
1.1.2	Training Scheme . . . . .	1
1.2	Boosting . . . . .	1
1.2.1	BrownBoost . . . . .	1
1.2.2	LogitBoost . . . . .	1
<b>2</b>	<b>Per-dataset computation</b>	<b>1</b>
2.1	Evaulation and Error . . . . .	1
2.2	The Decoder . . . . .	2
2.3	Finding Epsilon . . . . .	2

## 1 The pre-computation

### 1.1 The Neural Net

#### 1.1.1 Simulated Data

#### 1.1.2 Training Scheme

### 1.2 Boosting

#### 1.2.1 BrownBoost

#### 1.2.2 LogitBoost

## 2 Per-dataset computation

### 2.1 Evaulation and Error

Having constructed our monster classifier, we now brace to use it. We have already laid out the general principle behind how it is to be used. For some starting position, look forward some fixed number of points into the future, where each point is separated by a constant number of frames, find the error bar at these points, and pass them into the neural net. We will thus generate a sequence of classifications, starting at

every frame in the dataset except for the small slice of the set which would require looking at frames past the end.

## **2.2 The Decoder**

## **2.3 Finding Epsilon**

Up until now, I have assumed we somehow knew  $\epsilon$ . However, there is no obvious way to predict this from theory. We thus must resort to using the dataset to find the most probable value of it. This is similar in concept to finding  $\sigma$  in my Dynamic Binning algorithm, just harder to implement.

The program has been tasked with the problem of, given the classifications in an interval, find the best  $\epsilon$  to work with. Let's start by working this problem in reverse: if we knew  $\epsilon$ , what would be the distribution of classifications? At this point, we must define some more terminology before we can proceed.