Finite Gaussian Neurons - A Defense Against Adversarial Attacks?

by

Felix Grezes

A dissertation proposal submitted to the Graduate Faculty in Computer Science in partial fulfillment of the requirements for the degree of Doctor of Philosophy, The City University of New York.

2020

This manuscript has been read and accepted for the Graduate Faculty in Computer Science in satisfaction of the dissertation requirements for the degree of Doctor of Philosophy.

(required signature)

Date

Chair of Examining Committee

(required signature)

Date

Executive Officer

1st Committee member

2d Committee member

3d Committee member

4th Committee member

Supervisory Committee

THE CITY UNIVERSITY OF NEW YORK

Abstract

Finite Gaussian Neurons - A Defense Against Adversarial Attacks?

by

Felix Grezes

Advisor: Pr. Michael I. Mandel

Text of Abstract, up to 350 words

Acknowledgements

Contents

Chapter 1

Introduction

Chapter 2

Background

2.1 Neural Networks

2.1.1 Particulars of Neural Networks for Vision

2.1.2 Particulars of Neural Networks for Audio

2.2 Adversarial Attacks on Neural Networks

2.2.1 Visual Adversarial Attacks

2.2.2 Audio Adversarial Attacks

2.3 Proposed Work

2.3.1 Main Idea: Finite Gaussian Neuron Activity

A typical artificial neuron's output y is defined by its inputs x_i and associated weights w_i as:

$$y = \varphi(\sum_i w_i x_i)$$

with φ being the non-linear activation function required by the universal approximator theorem [?, ?]. Usually a bias term is included, but it can be written as an extra input with value 1.

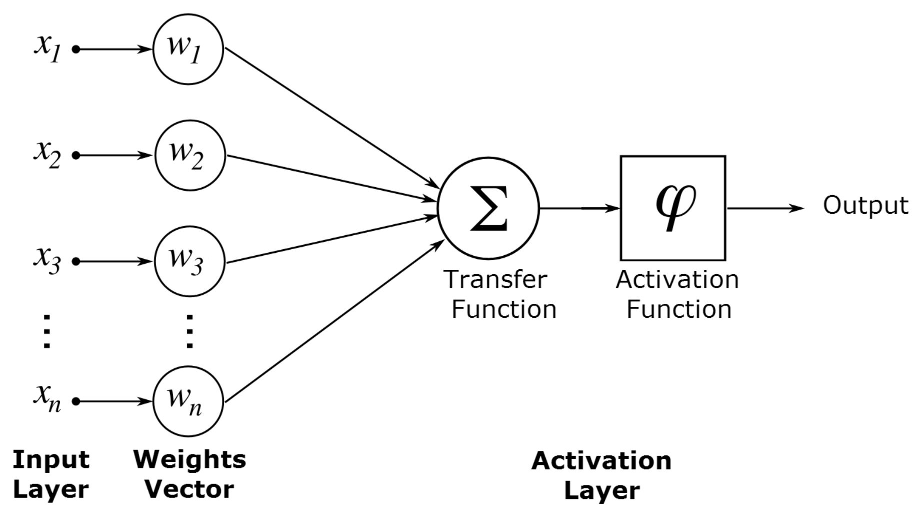

Figure 2.1: Model of an Artificial Neuron

The x_i inputs times w_i weights product defines an underlying linear activity gradient over the input space, which is theorized to be a reason adversarial attacks on neural networks are effective [?]. An visual example of the linear activity over a 2D input space is given by 2.2.

To counter the locally linear decision boundaries of neural networks, I propose a modified neuron architecture, the Finite Gaussian Neuron (FGN). The output y of a FGN is the x_i inputs times w_i weights product, multiplied

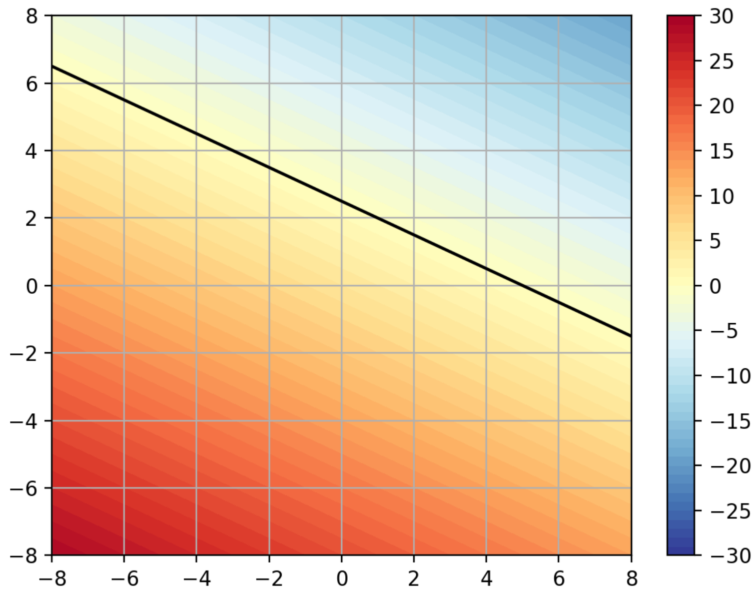

Figure 2.2: Underlying Linear Activity Map for a 2D Neuron

by a circular Gaussian with learned mean and variance parameters:

$$y = (\sum_i w_i x_i) * e^{(-1/\sigma^2)*(\sum_i (x_i - c_i)^2)}$$

with σ the learned mean and c_i the learned center per input dimension. An visual example of the localized activity over a 2D input space is given by 2.3. Note that the circular Gaussian provides both the non-linearity and the bias term of classical artificial neurons. Both of which are needed for the universal approximator theorem for neural networks [].

$\varphi(\cdot)$ needs to be a nonconstant, bounded, and continuous function.

We can write the activity of a LGN as:

$$y = (\sum_i w_i x_i) * e^{(-1/\sigma^2)*(\sum_i (x_i - c_i)^2)}$$

$$y = w_T x*$$

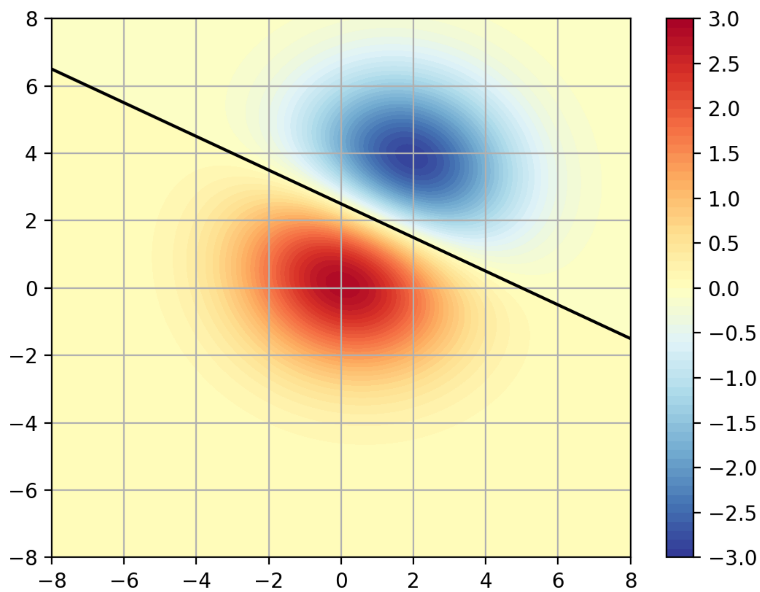

Figure 2.3: Localized Activity Map for a Proposed 2D Neuron

2.3.2 Task 1: Visual

2.3.3 Task 2: Audio

Bibliography