Introduction to Neural Networks

Dr. Mehmet Kerem Turkcan, Columbia University
October 24, 2024

Contents

1 Introduction

In this lab, you will explore the concepts of neural networks using the TensorFlow Playground, an interactive web application that allows you to visualize and experiment with different neural network configurations.

2 Objectives

By the end of this lab, you should be able to:

- Understand the basic components of a neural network
- Experiment with different network architectures
- Analyze the effects of various hyperparameters on network performance
- Visualize decision boundaries for different classification problems

3 Materials

- Computer with internet access
- Web browser (Chrome recommended)
- Access to the TensorFlow Playground: https://playground.tensorflow.org/

4 Procedure

4.1 Part 1: Familiarization with TensorFlow Playground

1. Open the TensorFlow Playground in your web browser.
2. Observe the main components:
    - Input features (X_1, X_2)

- Hidden layers and neurons
- Output layer
- Data points (orange and blue)
- Training loss and test loss

4.2 Part 2: Simple Classification Problem

1. Set up a simple classification problem:
   - Choose "Circle" from the data type dropdown
   - Set noise to 0
   - Use 2 hidden layers with 4 neurons each
   - Set learning rate to 0.03
   - Use "tanh" as the activation function
2. Click "Run" and observe how the network learns to classify the points.
3. Record the final training loss and test loss after 1000 epochs.

4.3 Part 3: Experimenting with Hyperparameters

1. Modify the following parameters one at a time, keeping others constant:
   (a) Number of hidden layers (1, 2, 3)
   (b) Number of neurons per layer (2, 4, 8)
   (c) Learning rate (0.001, 0.01, 0.1)
   (d) Activation function (ReLU, tanh, sigmoid)
2. For each modification, run the network for 1000 epochs and record:
   - Final training loss
   - Final test loss
   - Observations on how quickly the network converges

4.4 Part 4: Complex Problems

1. Choose the "Spiral" data type from the left column.
2. Experiment with different network architectures to find a configuration that successfully classifies the spiral data.
3. Record your final network configuration and the resulting losses.

5 Questions for Analysis

1. How does the number of hidden layers affect the network's ability to learn complex patterns?
2. What role does the activation function play in the network's performance?
3. How does the learning rate impact the training process?
4. For the spiral data, which network configuration worked best? Why do you think this is the case?
5. How does adding noise to the data affect the network's performance? Experiment and explain your observations.