Shahid Beheshti University
Artificial Neural Networks
M.Sc - Fall 2024

Assignment 1

1 Theoretical Questions

1. **Exercise 1:** Examine the idea of sparse connectivity in multilayer perceptrons and how it affects the effectiveness of the model. Describe the effects of various pruning techniques on model size and performance, emphasizing the effects of optimal brain damage, sensitivity-based pruning, and magnitude-based pruning.

2. **Exercise 2:** Explain why training MLPs requires accurate weight initialization. Examine various weight initialization strategies, including He, Xavier, and random initialization, and describe when each might be suitable. Talk about and solve potential issues that could arise if the weights in our model are symmetric.

3. **Exercise 3:** Compare the Sigmoid, Tanh, and ReLU activation functions in terms of their mathematical properties and impact on training deep neural networks.

4. **Exercise 4:** Describe the Stochastic Gradient Descent (SGD) algorithm. How does it differ from Batch Gradient Descent? Discuss the impact of mini-batch size on the convergence of SGD.

5. **Exercise 5:** Provide the mathematical expressions for L1 and L2 regularization. How do they affect the weights during training?

6. **Exercise 6:** Define the vanishing and exploding gradient problems in deep learning. Explain how these problems affect the training of deep neural networks.

7. **Exercise 7:** Describe techniques to address vanishing and exploding gradient problems. Include mathematical justifications for how these techniques help.

8. **Exercise 8:** Explain why initializing all weights to zero in a neural network leads to poor learning. How does random weight initialization help, and what are some common strategies for initializing weights?

9. **Exercise 9:** Explain the purpose of activation functions in neural networks. Compare the properties of Sigmoid, Tanh, ReLU, and Leaky ReLU activation functions, and discuss scenarios where each is most appropriate.

10. **Exercise 10:** Define the Softmax function and explain how it is used in multi-class classification problems. How does it differ from the Sigmoid activation function?

11. **Exercise 11:** Explain how the use of Sigmoid and Tanh activation functions can lead to the vanishing gradient problem. What is the mathematical basis for this issue?

12. **Exercise 12:** Describe strategies to mitigate the learning slowdown caused by Sigmoid and Tanh activations. How do alternative activation functions like ReLU help address this problem?

13. **Exercise 13:** What are the signs of overfitting in a neural network model? How can you use validation data to detect overfitting during training?

14. **Exercise 14:** A neural network learns by adjusting its weights. The commonly used technique for updating effectively is called "backward propagation of errors," or backpropagation. Now consider the neural network below, where the cost function is "Mean Squared Error," and the activation function of the neurons is Sigmoid.

    (a) Calculate the network error after one step of feed-forward.

    (b) Determine the backpropagation step with a learning rate of 0.3.

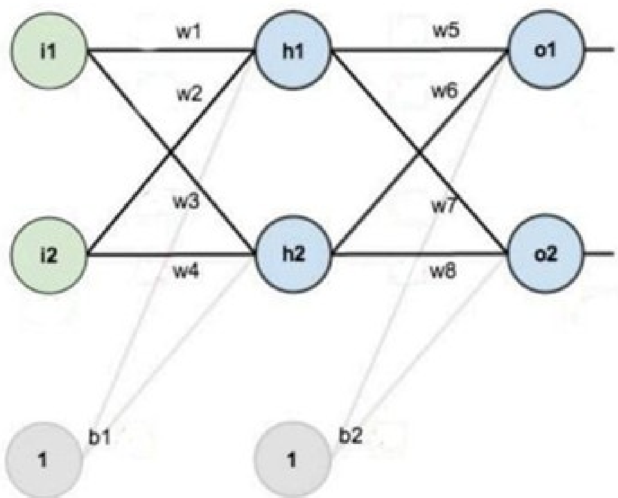

Figure 1:

- **Weights:**

$$\begin{aligned}
W_1 &= 0.15 && i_1 \to h_1 \\
W_2 &= 0.20 && i_1 \to h_2 \\
W_3 &= 0.25 && i_2 \to h_1 \\
W_4 &= 0.30 && i_2 \to h_2 \\
W_5 &= 0.40 && h_1 \to o_1 \\
W_6 &= 0.45 && h_1 \to o_2 \\
W_7 &= 0.50 && h_2 \to o_1 \\
W_8 &= 0.55 && h_2 \to o_2
\end{aligned}$$

- **Inputs:**

$$\begin{aligned}
i_1 &= 0.5 && \text{(input 1)} \\
i_2 &= 0.1 && \text{(input 2)}
\end{aligned}$$

- **Biases:**

$$\begin{aligned}
b_{1-1} = b_{1-2} && = 0.35\text{(hidden layer bias)} \\
b_{2-1} = b_{2-2} && = 0.60\text{(output layer bias)}
\end{aligned}$$

- **Target outputs:**

$$\begin{aligned}
o_1 &= 0.01 && \text{(target output 1)} \\
o_2 &= 0.99 && \text{(target output 2)}
\end{aligned}$$

2 Practical Exercise

Health Insurance Charges Prediction Using MLP

2.1 Introduction

The purpose of this assignment is to predict individual medical costs based on several personal factors. A dataset is given to you that includes details such as age, gender, body mass index (BMI), smoking status, number of dependents, and residential region in the US. Using this data, we are going to build a model that can predict health insurance charges.

The challenge here is to design and train a custom Multi-Layer Perceptron (MLP) model for this regression task. This model should be built using no pretrained models and optimized through careful tuning of its architecture and hyperparameters.

2.2 Dataset Description

The dataset includes the following features:

- **age**: The age of the primary beneficiary.
- **sex**: The gender of the insurance holder, either male or female.
- **bmi**: The body mass index (BMI) of the individual, a measure that correlates weight with height. The ideal range is 18.5 to 24.9.
- **children**: The number of dependents covered by the health insurance policy.
- **smoker**: Whether the individual is a smoker (yes/no).
- **region**: The geographical region of the policyholder in the US (northeast, southeast, southwest, northwest).
- **charges**: The actual medical costs billed by health insurance. This is the target variable we aim to predict.

2.3 Project Requirements

Students are required to implement the following steps as part of their assignment:

2.3.1 Data Encoding and Preprocessing

The dataset contains both categorical features (like *sex*, *smoker*, *region*) and continuous features (like *age*, *bmi*, *children*). Before building the model, students must preprocess the data:

- Categorical variables should be encoded into numerical form using appropriate techniques (e.g., one-hot encoding).
- Continuous variables should be scaled or normalized to ensure that no single feature disproportionately affects the model's learning.

The goal is to maximize the use of the data and ensure that the model can effectively learn from the given features.

2.3.2 Model Design and Implementation

Students will design a custom Multi-Layer Perceptron (MLP) model to predict the medical costs. An MLP is a type of neural network with multiple layers of neurons, which makes it capable of learning complex patterns in the data.

The architecture of the MLP (e.g., number of layers, number of neurons per layer, activation functions) is a key factor in model performance. Students must experiment with different architectures and configurations to optimize the model's performance. The training process should include:

- Choosing the right number of hidden layers and neurons.
- Applying regularization techniques (such as dropout or weight decay) to prevent overfitting.
- Using an appropriate loss function for regression tasks (e.g., Mean Squared Error).
- Tuning hyperparameters such as learning rate, batch size, and number of epochs.

2.3.3 Model Evaluation and Analysis

To assess model performance, students should split the data into training and test sets (or use cross-validation). The model will be evaluated primarily using the Mean Squared Error (MSE), which measures the average squared difference between the predicted and actual medical costs. Students should also perform error analysis to identify where the model is underperforming and suggest potential improvements.

2.3.4 Exploratory Data Analysis (EDA)

Before building the model, students should conduct comprehensive Exploratory Data Analysis (EDA) to gain insights into the dataset. This will help uncover relationships between features and the target variable (medical charges). Key tasks include:

- Visualizing distributions of features (e.g., histograms, boxplots).
- Examining correlations between features using heatmaps.
- Identifying any outliers, missing values, or potential data imbalances.

This step ensures that students have a thorough understanding of the data, which can inform model design decisions.

2.4 Report Structure

The final submission must include a detailed report that outlines the following:

- **Introduction**: A brief overview of the task, the dataset, and the objective of the project.
- **Data Analysis**: A summary of the dataset, including insights gained from EDA and any preprocessing steps taken.

- **Model Explanation**: A detailed explanation of the MLP architecture, including the number of layers, neurons, activation functions, and the rationale behind these design choices.
- **Hyperparameter Tuning**: A discussion of the tuning process, with explanations of how different hyperparameters were optimized to improve the model's performance.
- **Results**: An evaluation of the model's performance on the test set, including MSE or other relevant metrics. A comparison of different model architectures and their results should also be included.
- **Conclusion**: A summary of the project, including key findings, challenges encountered, and potential areas for improvement.

2.5 Submission Files

Students must submit the following files:

- **Code (.ipynb format)**: The complete implementation of the model in a Jupyter Notebook.
- **Practical Task Report**: A detailed scientific-style report following the structure outlined above.
- **Theoretical Questions**: Answers to any theoretical questions included in the assignment.