Deep Learning and Industrial Applications
Homework 2

1. Select 2 hyper-parameters of the artificial neural network used in Lab 2 and set 3 different values for each. Perform experiments to compare the effects of varying these hyper-parameters on the loss and accuracy metrics across the training, validation, and test datasets. Present your findings with appropriate tables.

   Hyperparameter: Learning Rate (0.01, 0.001, 0.0001) / Epoch (100, 300, 500)

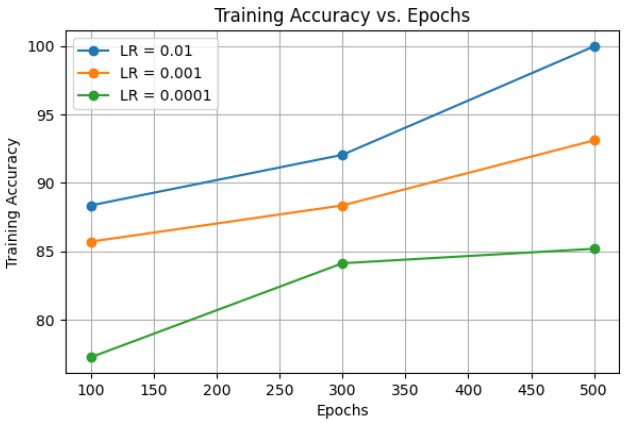
	Training Acc / Loss	Valid Acc / loss	Testing Acc / Loss
0.01 / 100	88.36% / 0.2317	79.01% / 0.6984	70.97% / 0.5203
0.001 / 100	85.71% / 0.3426	76.54% / 0.5131	83.87% / 0.5647
0.0001 / 100	77.25% / 0.4749	66.67% / 0.5705	61.29% / 0.6705

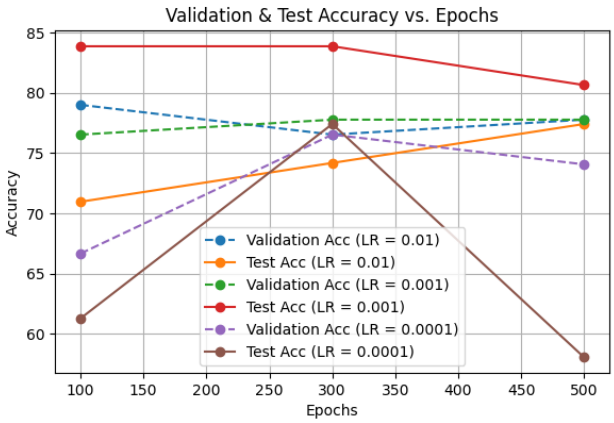
	Training Acc / Loss	Valid Acc / loss	Testing Acc / Loss
0.01 / 300	92.06% / 0.1713	76.54% / 0.8126	74.19% / 0.4601
0.001 / 300	88.36% / 0.2320	77.78% / 0.6230	83.87% / 0.6497
0.0001 / 300	84.13% / 0.3755	76.54% / 0.5352	77.42% / 0.6171

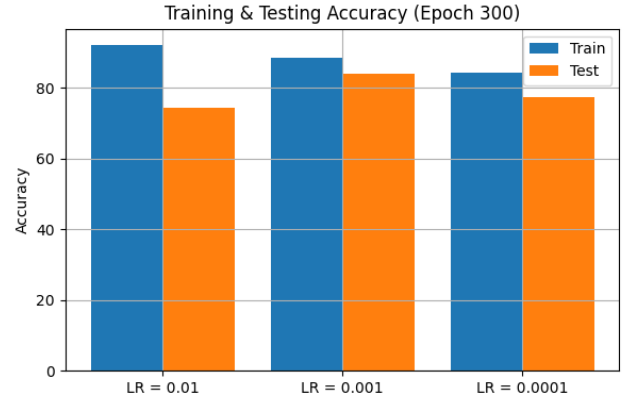
	Training Acc / Loss	Valid Acc / loss	Testing Acc / Loss
0.01 / 500	100.00% / 0.0297	77.78% / 2.2225	77.42% / 0.6281
0.001 / 500	93.12% / 0.1753	77.78% / 0.8035	80.65% / 0.6920
0.0001 / 500	85.19% / 0.3121	74.07% / 0.5283	58.06% / 0.5887

2. Based on your experiments in Question 1, analyze the outcomes. What differences do you observe with the changes in hyper-parameters? Discuss whether these adjustments contributed to improvements in model performance, you can use plots to support your points. (Approximately 100 words.)

- We can analyze the outcomes from settings of learning rate and epoch
  - Learning Rate
    - Learning Rate 0.01: training accuracy is higher but loss of validation and testing sets are higher. Model might be overfitting especially with higher epoch
    - Learning Rate 0.001: The performance of training / valid / testing sets is relatively stable and better among three epoch sets
    - Learning Rate 0.0001: training accuracy is lower among three epoch sets, which indicates model might be underfitting
  - Epoch
    - Epoch 100: Model is not converged, the accuracy is low
    - Epoch 300: The accuracy among three learning rates is relatively stable
    - Epoch 500: The model must be overfitting when learning rate is 0.01 or 0.001. When the learning rate is 0.0001, there is no significant improvement in Training Accuracy in the experiment with 500 Epochs compared to 300 Epochs. It may be necessary to modify the model architecture to achieve a breakthrough.
- From the accuracy of testing set and loss of training and valid sets, when learning rate is 0.001 and epoch is 300. The model can be relatively generalized.







3. In Lab 2, you may have noticed a discrepancy in accuracy between the training and test datasets. What do you think causes this occurrence? Discuss potential reasons for the gap in accuracy. (Approximately 100 words.)
    - Observe the above results, I think model overfitting causes this condition. The model learns patterns specific to the training data but cannot generalize to test data.
    - Another possible reason is an insufficient amount of training and validation data, which may lead to poor generalization. With limited data, the model may not capture diverse patterns, making it more sensitive to noise.
    - The third reason might be different data distribution, which can happen due to temporal shifts, sampling bias, or variations in feature distributions, causing the model to struggle with generalization.

4. Discuss methodologies for selecting relevant features in a tabular dataset for machine learning models. Highlight the importance of feature selection and how it can impact model performance. You are encouraged to consult external resources to support your arguments. Please cite any sources you refer to. (Approximately 100 words, excluding reference.)
    - Feature selection is crucial in machine learning, since it can remove redundant and irrelevant features, reducing model complexity and improve model's generalization ability
    - Common feature selection: filter method, wrapper method, embedded method
        — Filter method: The filter method selects features based on their statistical properties and relevance to the target variable, independent of any machine learning model (e.g., Correlation, Chi-Square, mutual information…)
        — Wrapper method: It selects features by training the model on different feature subsets and assessing the performance using a specific evaluation metric (e.g., RFE, SFS…)
        — Embedded method: It implements feature selection directly within the learning algorithm, with examples including LASSO (which applies L1 regularization) and tree-based algorithms. (e.g., Lasso, DecisionTree, RandomForest…)
    - Reference: 1.13. Feature selection — scikit-learn 1.6.1 documentation

5. While artificial neural networks (ANNs) are versatile, they may not always be the most efficient choice for handling tabular data. Identify and describe an alternative deep learning model that is better suited for tabular datasets. Explain the rationale behind its design specifically for tabular data, including its key features and advantages. Ensure you to reference any external sources you consult. (Approximately 150 words, excluding reference.)
   - TabNet
     - Algorithm Design: TabNet is a deep learning model specifically designed for tabular data, offering a unique architecture that focuses on efficiency and interpretability. It combines a feature transformer and an attentive transformer, where sequential attention is used to select the most relevant features at each decision step. This mechanism allows the model to focus on important features while ignoring irrelevant ones, improving computational efficiency and reducing the risk of overfitting.
     - Key Features & Advantages:
       - Sequential Attention Mechanism: Enables the model to prioritize key features while discarding unimportant ones, leading to enhanced efficiency and reduced overfitting.
       - Feature Masking: Ensures that only the most informative features contribute to the learning process, improving both model performance and interpretability.
       - Interpretability: Unlike traditional deep learning models, TabNet provides transparency by allowing users to understand which features drive decision-making.
       - Superior Performance: By effectively handling complex relationships in tabular data while maintaining computational efficiency, TabNet outperforms both deep learning and tree-based models in various data tasks.
   - Reference: TabNet: Attentive Interpretable Tabular Learning