National Tsing Hua University

1130IEEM 513600

Deep Learning and Industrial Applications

Homework 2

Name: 董少霖　　　　　　Student ID: 113034567

Due on 2025.03.27

1. (20 pts) 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.

   Learning Rate: 0.1, 0.01, 0.001

   Batch Size: 16,32,64

Learning Rate	**Batch Size**	Training Loss	Training Acc	Validation Loss	Validation Acc	Test Loss	Test Acc
0.001	16	0.3383	84.13%	0.5881	77.78%	0.5030	70.97%
0.001	32	0.3668	84.66%	0.4633	79.01%	0.5804	80.65%
0.001	64	0.4343	79.89%	0.8915	74.07%	0.6884	74.19%
0.01	16	0.3355	85.18%	0.2967	81.48%	0.4151	80.65%
0.01	32	0.2682	88.89%	0.4678	76.54%	0.4397	83.87%
0.01	64	0.2720	89.95%	0.4506	77.78%	0.5043	74.19%
0.1	16	0.6864	56.08%	0.7184	50.62%	0.7658	48.39%
0.1	32	0.6904	53.97%	0.6843	55.56%	0.8677	48.39%
0.1	64	0.6892	54.50%	0.7020	54.32%	3.4853	48.39%

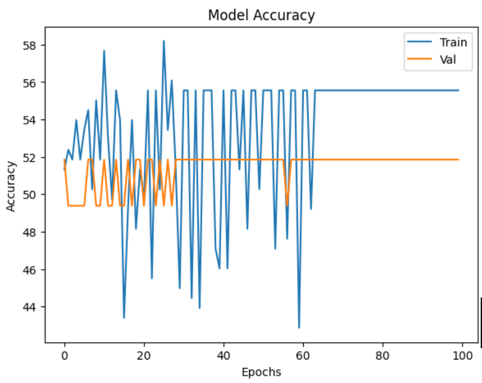
2. (20 pts) Based on your experiments in Question 1, analyze the outcomes. What differences do you observe with the changes in hyper-parameters? Discuss whether these adjustments contributed to improvements in model performance, you can use plots to support your points. (Approximately 100 words.)

   Based on the results from Question 1, both the learning rate and batch size had a notable impact on the model's performance. The highest test accuracy of **83.87%** was achieved with a learning rate of **0.01** and a batch size of **32**.

Increasing the learning rate to **0.1** resulted in lower accuracy, likely due to the model struggling to converge. Conversely, while a smaller learning rate of **0.001** improved training stability, it reduced overall performance, possibly due to slower convergence.

Among the batch sizes tested, a moderate batch size of **32** consistently outperformed batch sizes of **16** and **64** when paired with the same learning rate (not significant in learning=0.1). This suggests that batch size **32** strikes a better balance between stability and efficiency.

In conclusion, the optimal combination of learning rate (**0.01**) and batch size (**32**) provided the best trade-off, demonstrating that moderate values for both parameters can enhance model performance.



Learning Rate=0.1

3. (20 pts) 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.)

- Data Imbalance:
  The training data has 54.8% positive cases, 45.2% negative cases. The training data has a slight bias towards positive cases, which may cause the model to overfit on that pattern, reducing its performance on the test set.
- Sample Size:
  The training dataset has **273 samples**, while the test dataset has only **31 samples**. A small sample size may result in the model not learning effectively. Also, this limited test sample size may lead to high variance in test accuracy, making it sensitive to small prediction errors.
- Missing Value and Data Quality:
  The training data contains missing values in multiple features. If these missing values were imputed improperly or the test data has different missing value patterns, this could cause discrepancies in model performance.
- Hyperparameter Tuning:

The experiment in the Q1 suggests the need for a more optimal learning rate combined with improved batch size selection.

4. (20 pts) 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.)

In real-world machine learning tasks, not every feature in the dataset plays an equal role in enhancing model performance. Some features may be redundant, irrelevant, or introduce noise. By selecting only the most important features, feature selection improves model accuracy, reduces the risk of relying on unimportant data, and enhances the model's interpretability. These are three main categories:

- Filter Methods:
  Use statistical metrics to evaluate each feature with target variable. For example: chi-square test, Pearson's correlation… and so on.
- Wrapper Methods:
  These iteratively select feature subsets by evaluating model performance. Techniques like Recursive Feature Elimination (RFE) and forward/backward selection are commonly used.
- Embedded Methods:
  Embedded methods select features during model training by combining filter and wrapper method advantages. They dynamically identify important features as the model trains. For example: Lasso regression and decision trees.

Effective feature selection enhances model efficiency, reduces computation costs, and prevents the inclusion of noisy or redundant features, improving generalization on unseen data.

Reference:

- https://www.geeksforgeeks.org/feature-selection-techniques-in-machine-learning/
- Chandrashekar, G., & Sahin, F. (2014). A survey on feature selection methods. *Computers & Electrical Engineering*. https://doi.org/10.1016/j.compeleceng.2013.11.024.

5. (20 pts) 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, introduced by Arik and Pfister from Google Research, is a deep learning model designed for tabular data. Here are the key features and advantages of TabNet:

- TabNet directly processes raw tabular data without preprocessing and is trained using gradient descent, allowing seamless integration into end-to-end learning frameworks.
- TabNet uses sequential attention to focus on key features at each step, improving interpretability and learning efficiency. Its instance-specific feature selection adapts to each input, achieved through a single deep learning architecture for both feature selection and decision-making.
- TabNet matches or exceeds the performance of other tabular models in classification and regression tasks across various domains. It also offers two types of interpretability: local (showing feature importance and combinations) and global (measuring each feature's contribution to the model).
- For the first time in tabular data, unsupervised pre-training with masked feature prediction has shown notable performance gains.

These characteristics make TabNet particularly suitable for structured data in applications like finance, healthcare, and marketing.

Reference:

- https://medium.com/@turkishtechnology/deep-learning-with-tabnet-b881236e28c1
- Arik, S. Ö., & Pfister, T. (2021, May). Tabnet: Attentive interpretable tabular learning. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 35, No. 8, pp. 6679-6687). https://arxiv.org/abs/1908.07442