Machine Learning Project

Ethan Tran, Alyan Tharani, Akil Manivannan

Abstract

In this project, we investigate different approaches to training deep learning models for medical imaging recognition. Deep learning has demonstrated high potential in accurately screening various diseases through medical image analysis. Our task involves conducting experiments on two medical imaging datasets, specifically BreastMNIST and PneumoniaMNIST. We aim to explore modern techniques in the problem of image classification. Despite employing a supervised approach, imbalanced data can still pose challenges, where a dataset may have a disproportionate number of members in a specific class compared to another. LibAUC provides a deep learning library that directly optimizes commonly used performance metrics. In our case, we work on maximizing the ROC curve (Receiver Operating Characteristic curve), which illustrates the performance of a classification model and provides an aggregate measure of performance across all possible classification thresholds. Our focus must now shift to optimizing LibAUC's AUCM loss metric. In contrast to the standard Cross-Entropy loss, which measures differences between probability distributions and provides a probability score for each class, utilizing AUC aligns with our goal of benchmarking model performance on imbalanced datasets. AUC is designed to balance the representation of the true positive rate against the false positive rate and can measure a model's ability to distinguish between these classes. Thus, we experiment with various models, architectures, and learning techniques to demonstrate their effectiveness.

1 Introduction

Neural networks have revolutionized image classification by offering unprecedented accuracy and efficiency in analyzing complex visual data. In medical imaging, neural networks have enabled rapid advancements in disease detection and diagnosis. These networks excel at recognizing intricate patterns and variations in medical images, leading to earlier detection of diseases such as cancer, pneumonia, and neurological disorders. Moreover, neural networks can automate tedious and time-consuming tasks. As a result, the integration of neural networks into medical imaging workflows has significantly enhanced diagnostic accuracy, treatment planning, and has heavily pushed the development and research aimed to optimize these models.

We begin our project by utilizing ResNet18, a convolutional neural network that is 18 layers deep to train a model on the BreastMNIST and PneumoniaMNIST datasets. The BreastMNIST is based on a dataset of 780 breast ultrasound images. This is a binary-class data set with the classifications of malignant or benign and malignant. The PneumoniaMNIST is based on a prior dataset of 5,856 pediatric chest X-Ray images. The task is also binary-class, with classification of pneumonia against normal. We split the source training set with a ratio of 9:1 into training and validation set and use its source validation set as the test set. We crop the images of both these and resize them into $1\times28\times28$ greyscale images.

We must now address the problem of imbalanced datasets. Imbalance can adversely affect the performance of a classification model, as it tends to bias results towards the majority class. In medical contexts, where the minority class often represents positive samples, imbalanced data exacerbates the challenge, leading to inaccuracies in predicting the minority class. When the imbalance problem arises, AUC (Area Under ROC curve) presents itself as a more suitable measurement for assessing a model's performance in such a dataset, as AUC is defined as the probability that a positive sample has a higher score than a negative sample. Thus, in this paper, we experiment with different models to demonstrate how each performs in the task of optimizing LibAUC's AUCM loss metric. We employ different architecture and employ following techniques

- Naive Bayes
- Decision Trees
- K-Nearest Neighbors

- Logistic Regression
- Support Vector Machines
- ResNet50
- Ensemble Learning

to improve the performance based on the training of AUCM loss and present how each performed in their respective datasets. As we use a supervised learning model, we can first train our ResNet18 with Cross Entropy loss to deliver a baseline accuracy in our overall classification problem. With this standard, we can now begin to experiment with other approaches and various ways to optimize these models.

2 Cross Entropy and AUCM Loss

This section discusses out base models that we shall use in comparison to the various models we shall employ later in the paper.

2.1 Cross Entropy Loss

We may begin by training a basic ResNet18 model using Cross Entropy Loss as a criterion. Commonly used in classification problems Cross-entropy loss measures the difference between two probability distributions: the predicted probabilities and the actual (true) probabilities. Hence, the output is often representative of a probability score for each class. For multi-class classification, it's the negative log-likelihood of the true class. We trained the model over 30 epochs using the BreastMNIST and PneumoniaMNIST to achieve an accuracy score of 69.86% and 88.30% respectively.

2.2 AUCM Loss

Now, we can use a criterion to optimize a model directly under the ROC curve. We measure the ability of our most basic model which aims to distinguish positive and negative classes. As a baseline, our hyperparemeters are as follows:

$$\begin{aligned}
\text{BATCH_SIZE} &= 128 \\
\text{imratio} &= 0.1 \\
\text{total_epochs} &= 100 \text{ for BreastMNIST} \\
\text{total_epochs} &= 20 \text{ for PneumoniaMNIST} \\
\text{decay_epochs} &= [50, 75] \\
lr &= 0.1 \\
\text{margin} &= 1.0 \\
\text{epoch_decay} &= 0.003 \\
\text{weight_decay} &= 0.0001
\end{aligned}$$

After transforming our data into Tensors and normalizing, we proceed to split the data into their training, testing, and validation datasets, and ran our ResNet18 model with AUCMLoss as our loss function. In the end, we achieved the following train loss: 0.0011/0.0009, train auc: 0.9940/0.9983, test auc: 0.8889/0.9232 for BreastMNIST/PneumoniaMNIST respectively.

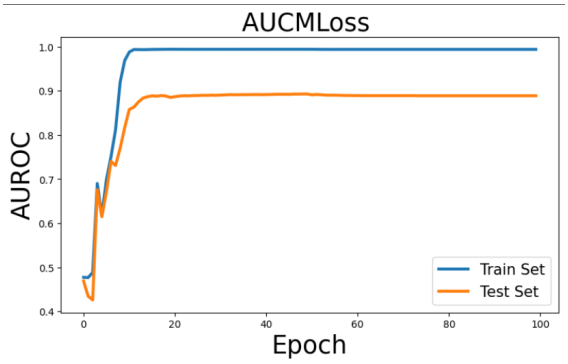


Figure 1: BreastMNIST

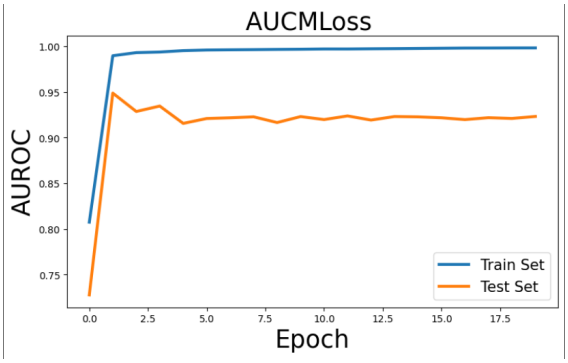


Figure 2: Pneumonia

which presents a baseline for our future models to improve upon.

3 Experiments

This section delves into exploring various techniques to enhance the performance of a deep learning model trained with the AUCM loss for imbalanced medical image classification tasks. While the previous section established the baseline performance with a ResNet18 architecture, we aim to further optimize the model's ability to maximize the Area Under the ROC Curve (AUC).

3.1 Other Non-Neural Models

The choice of models for this study was guided by the need to address the imbalanced nature of the datasets as well as its complexity due to the classification tasks. Each model offers distinct advantages and mechanisms for handling imbalanced data, which we exploit to enhance the overall performance.

3.1.1 Naive Bayes

Naive Bayes is chosen due to its simplicity and effectiveness in baseline performance benchmarks. It allows us to see how it partakes amongst this dataset and if it is beneficial for us to include its logic within our neural network solution. Further, it serves as a probabilistic model that assumes feature independence and offers a quick reference point for performance comparison against more complex models.

3.1.2 Decision Tree

Decision Trees are included for their ability to model non-linear decision boundaries. They are particularly useful in medical image classification (which we are looking at within this project) because they facilitate the understanding of which features are most influential in predicting the outcome, aiding interpretability. This model seems to do well with other datasets we have tested as well regarding medical image classification.

3.1.3 K Nearest Neighbors

The K Nearest Neighbors (KNN) algorithm is utilized for its simplicity and effectiveness in capturing the local structure of the data. KNN's performance is highly dependent on the choice of hyper parameters, which is why extensive tuning is conducted. This algorithm was difficult to facilitate due to the necessity of adjusting all parameters for minor changes. Its sensitivity to feature scaling and distance metric choices are some of the fine-tuning values we adjusted to adapt to the varied data distributions.

3.1.4 Logistic Regression

Logistic Regression is chosen for its ability to provide probabilistic outputs and to model the probability of class memberships. Its inclusion is pivotal for tasks requiring a balance between precision and recall, which was important within the dataset we used and it can help us decipher where have both false positives and false negatives which carry significant consequences if not detected.

3.1.5 Support Vector Machine

Support Vector Machines (SVM) are included for their robustness and effectiveness in high-dimensional spaces, common in image data. We can see that this model preformed very well amongst the other datasets and using the different kernels allowed us to be more felxable and further develop our model on non-linear boundaries effectively, making it suitable for complex image classification tasks as such.

3.1.6 Interaction Among Models

The selected models cover a spectrum from simple to complex and linear to non-linear classifiers. This diversity enables a comprehensive analysis of the dataset under different modeling assumptions. By comparing these models, we can identify which strategies are most effective in dealing with class imbalance and which configurations yield the best balance between sensitivity and specificity. The ensemble of these models could potentially lead to a robust prediction system such as the neural network which this project develops upon, further leveraging the strengths of each individual algorithm.

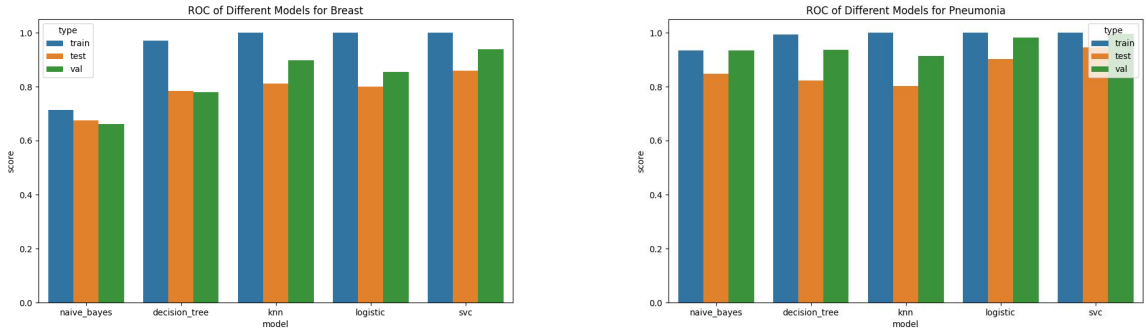
3.2 Optimization Techniques

For each of these algorithms, we conducted a hyper parameter tuning process. This involved isolating a set of relevant hyper parameters known to influence the respective algorithm's performance. Subsequently, we employed a 5-fold cross-validation strategy to evaluate the performance of each possible hyper parameter combination. Cross-validation helps mitigate over fitting and provides a more robust estimate of the model's generalization ability.

Table 1 shows the different hyperparameters used for each model type. Moreover, Figure 3 shows the accuracy of these optimized models on the dataset. Note that while the graph includes the ROC for the training, testing, and validation data, only the training data was used for optimization.

Model Type	**Parameter**	**Description**	**Values Tested**
Naive Bayes			
Decision Tree	criterion	Function to determine if ndoe should be split	"gini", "entropy", "loss"
	min_samples_split	Number of samples that warrant a leaf splitting	$[2, 50]$
	min_samples_leaf	Number of samples required to be a leaf node	$[1, 20]$
K Nearest Neighbors	n_neighbors	Number of neighbors to consider	$[1, \sqrt{x}]$
	weights	Should the distance of neighbors play a role	"uniform", "distance"
	metric	Distance metric	...
Logistic Regression	penalty	Regularization Metric	"l1", "l2", "elasticnet"
	C	Regularization Parameter	[10^-6, 10^6]
	class_weight	Should the number of classes matter	"balanced", "none"
Support Vector Machine	C	Regularization Parameter	[10^-6, 10^6]
	kernel	The kernel function to use	"poly", "rbf", "sigmoid"
	degree	Degree for polynomial kernel	[1, 8]
	gamma	How much a single example should influence	[1, 8]

Table 1: Different Hyper parameters for Each Model



(a) Breast Performance

(b) Pneumonia Performance

Figure 3: Other Model Performance

4 Final Method: Ensemble Learning

Building upon the exploration of individual machine learning algorithms, we investigated the potential benefits of an ensemble classifier to further improve the overall performance. Ensemble learning is a powerful technique that combines predictions from multiple models to achieve a more robust and accurate outcome.

In this context, we leveraged the optimized versions of the previously explored algorithms (Naive Bayes, Decision Tree, KNN, Logistic Regression, and SVM) alongside the optimized ResNet18 trained with AUCM loss. The core idea lies in exploiting the diversity of these models' decision-making processes. By combining their individual strengths and weaknesses, the ensemble classifier can potentially achieve superior performance compared to any single model.

Our approach to ensemble learning involved utilizing the predicted probabilities generated by each model. We adopted a simple yet effective strategy of averaging the probabilities from all models to create a final, consolidated probability. This approach assumes that the individual models capture complementary aspects of the data, and averaging their predictions leads to a more informed and robust classification decision.

Figure 4 shows the final model architecture, combining the optimized models discussed previosuly in the paper.

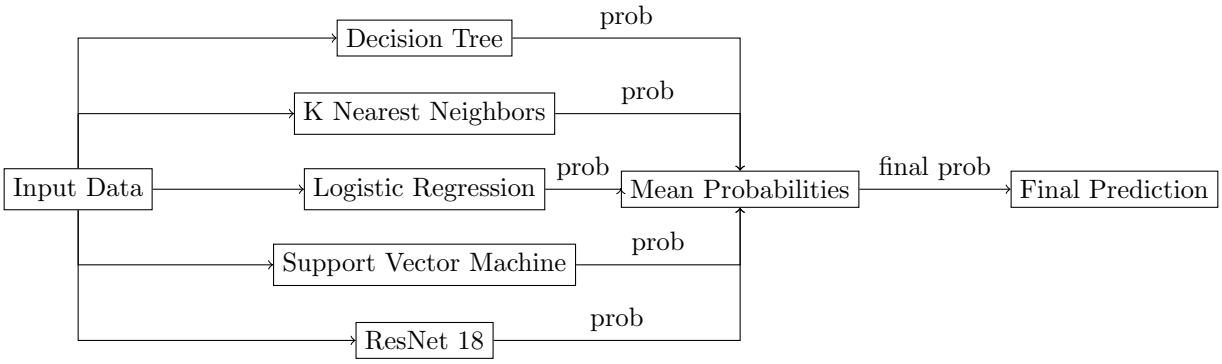


Figure 4: Final Ensemble Network

4.1 Final Accuracy

The final testing ROC accuracy was $\boxed{0.8999}$ for the breast cancer test set and $\boxed{0.9375}$ for the pneumonia test set. This is a 5 and 8 percent improvement over the base model, respectively.

We believe that further training of the ResNet, along with training a more detailed Ensemble Model such as soft voting can push this bound even further.