**Model Training and Evaluation Report**

## 1. Introduction

This report outlines the process and results of training a machine learning model for water quality classification using a neural network. The objective was to predict the potability of water based on various physicochemical properties. The key focus was on selecting suitable optimization techniques, and regularization methods to achieve an optimal balance between precision and recall.

## 2. Model Configuration

### 2.1 Optimization and Regularization Techniques Used

For this experiment, I implemented the following:

- **Optimizer:** Adam (Adaptive Moment Estimation)
- **Regularization:** L2 Regularization (Ridge Regression)
- **Early Stopping:** Implemented to halt training when validation loss stopped improving.
- **Dropout:** Applied to prevent overfitting.

### 2.2 Justification for Choices

#### Adam Optimizer

- Adam combines the advantages of **Momentum** and **RMSprop**, making it well-suited for handling noisy gradients and adaptive learning rates.
- It adjusts the learning rate dynamically, ensuring stable convergence.
- It is computationally efficient and typically performs well without requiring extensive hyperparameter tuning.

#### L2 Regularization

- L2 penalty discourages large weight values, preventing overfitting.
- Unlike L1 regularization, which forces some weights to zero, L2 regularization distributes penalties across all weights, retaining important features.
- It enhances the model’s generalization ability, especially when working with limited or noisy data.

## 3. Model Performance and Metrics

The model was evaluated using the following performance metrics:

| metric | Score |
|---|---|
| F1Score | 0.459 |
| precision | 0.71 |
| Recall | 0.33 |
| Accuracy | 0.689 |

### 3.1 Interpretation of Results

- The **F1 Score of 0.459** indicates a moderate balance between precision and recall, but there is room for improvement.
- The presence of **class imbalance** might have affected recall, leading to a lower F1 score.
- **Regularization strength:** If L2 regularization was too strong, it may have suppressed useful feature contributions, negatively impacting recall.

### 4. Comparison with Team Members' Models

To assess model effectiveness, a comparison was conducted with two team members' models, which used different optimization and regularization techniques.

| model | optimizer | Regularization | f1Score | precision | recall |
|---|---|---|---|---|---|
| My model | Adam | L2 | 0.459 | 0.68 | o.33 |
| Willy kalisa | SGD | L2 | 0.3 | 0.65 | 0.77 |
| Justice | RMSProp | L1 | 0.77 | 0.62 | 1.0 |

### 4.1 Key Observations

1. **Recall vs Precision Tradeoff:**
    - My model had **better precision but lower recall** than the other two members, meaning it made fewer false positives but missed some positive cases.
2. **Effect of Regularization and Optimization Choices:**

    - Adam's adaptive learning rate may have caused some fluctuations in training, while Willy Kalisa may have provided more stable convergence.
    - L2 regularization ensured my model generalized well but might have slightly penalized important features.

**5. Areas for Improvement**

To further improve the model performance of my model, I hope to implement the following:

- **Adjusting Regularization Strength:** Tuning the L2 penalty to find an optimal value that reduces overfitting without suppressing essential features.
- **Class Balancing Techniques:** Implementing oversampling or class weighting to ensure recall is not negatively affected by class imbalance.
- **Fine-Tuning the Learning Rate:** Experimenting with different learning rate schedules to improve convergence stability.
- **Threshold Adjustment:** Evaluating different classification thresholds to optimize precision-recall tradeoffs.

**6. Conclusion**

This report demonstrated the effectiveness of using Adam optimization and L2 regularization in training a water quality classification model. While the approach provided reasonable generalization, adjustments in regularization strength, class balancing, and threshold selection could further improve performance. The comparison with team members' models highlighted the impact of different optimization choices, guiding future enhancements.