**Optimizing Convolutional Neural Networks for Image Classification: A Comparative Study on Cat vs Dog**

**1. Introduction**

This report covers an in-depth analysis of image classification using Convolutional Neural Networks (CNNs), specifically targeting a binary classification task. The objective was to experiment with different CNN architectures, regularization methods, and hyperparameter settings to enhance model performance in classifying images. The study provides a comparative view on the impact of model complexity, regularization, and activation functions on training and validation accuracy.

**2. Experiment Design and Model Configurations**

**2.1 Model Architectures**

Various CNN architectures were explored to understand the effect of depth and complexity on performance:

* **Baseline Model**: A simple CNN with a few convolutional layers followed by fully connected layers.
* **Deep CNN Architecture**: An expanded model with additional convolutional and dense layers, aimed at capturing more complex patterns.

**2.2 Regularization Techniques**

To address overfitting and improve generalization, two primary regularization methods were applied:

* **Dropout Layers**: Introduced to randomly deactivate neurons during training, preventing dependency on specific neurons and reducing overfitting.
* **L2 Weight Regularization**: Penalties on large weights were applied to encourage simpler model structures that generalize better to unseen data.

**2.3 Activation Functions**

The choice of activation functions plays a critical role in CNN performance. Two functions were analyzed:

* **ReLU (Rectified Linear Unit)**: Frequently used in CNNs due to its simplicity and ability to reduce vanishing gradient issues.
* **Sigmoid and Tanh**: Tested in select configurations for comparison, though generally less common in CNNs for hidden layers.

**2.4 Hyperparameter Tuning**

Several hyperparameters were adjusted to optimize model performance:

* **Learning Rates**: Varied to optimize convergence speed.
* **Batch Sizes**: Tested to examine effects on training efficiency and stability.

**3. Training and Validation Results**

Each model configuration was trained across multiple epochs, and the training and validation accuracy and loss metrics were captured. The following tables summarize the performance across key configurations.

**3.1 Performance Metrics**

| **Model Configuration** | **Training Accuracy** | **Training Loss** | **Validation Accuracy** | **Validation Loss** |
| --- | --- | --- | --- | --- |
| Baseline CNN | 87.6% | 1.00 | 94.9% | 0.21 |
| Deep CNN with Dropout | 97.4% | 0.08 | 97.5% | 0.12 |
| Deep CNN with L2 Regularization | 98.0% | 0.07 | 97.3% | 0.15 |

**3.2 Epoch-wise Analysis**

The deep CNN models with dropout and L2 regularization outperformed the baseline, demonstrating both higher accuracy and lower loss in validation, which indicates better generalization and reduced overfitting.

**4. Comparative Analysis of Key Factors**

**4.1 Regularization Impact**

The addition of dropout layers resulted in a noticeable reduction in overfitting, as evidenced by lower validation loss and higher accuracy across the models. L2 regularization provided stability during training and validation, although dropout appeared slightly more effective in achieving the best validation metrics.

**4.2 Hyperparameter Effects**

Adjusting hyperparameters like learning rate and batch size impacted training speed and stability:

* **Lower Learning Rate**: Led to slower but more stable training, with fewer oscillations in loss.
* **Higher Batch Size**: Accelerated training but occasionally resulted in higher training loss due to fewer updates per epoch.

**4.3 Activation Function Analysis**

The **ReLU activation function** outperformed others, showing faster training convergence and better accuracy. Sigmoid and Tanh functions were less effective, as they led to slower convergence and higher loss in comparison to ReLU.

**5. Final Model Performance and Insights**

The highest-performing configuration was the deep CNN with dropout layers, which demonstrated strong generalization and high validation accuracy.

| **Best Model Configuration** | **Training Accuracy** | **Validation Accuracy** |
| --- | --- | --- |
| Deep CNN with Dropout | 98.4% | 98.1% |

**Key Insights**:

* **Model Depth**: Increasing the model’s depth by adding convolutional layers improved its ability to capture complex features, resulting in higher accuracy.
* **Regularization**: Dropout proved highly effective in reducing overfitting and enhancing validation accuracy.
* **Activation Functions**: ReLU was optimal for this task, providing faster training and superior performance compared to Sigmoid and Tanh.

**6. Conclusion**

This study demonstrates the effectiveness of CNNs in image classification tasks, with a focus on optimization through regularization and model architecture. Deeper architectures with dropout layers significantly improved performance, achieving high validation accuracy and minimized overfitting. This work underlines the importance of model complexity and regularization in achieving robust performance in neural networks.