# Homework 2 – Intro to Deep Learning

Daniel Jolin[1]
Student ID: 801282735
Homework: 2
github.com/RocketDan11/ml/real_time/homework/homework2.ipynb
Date: 2/18/25

**Abstract:** This document investigates deep learning models for CIFAR-10 and CIFAR-100 classification. We compare simplified versions of AlexNet, VGGNet, and two variants of ResNet (ResNet-11 and ResNet-18). Dropout techniques are employed to mitigate overfitting. Detailed comparisons include training loss, validation loss, validation accuracy, and model complexity.

# 1 Problem 1: Simplified AlexNet for CIFAR Datasets

The original AlexNet, designed for 227×227 images with over 60 million parameters, is too complex for low-resolution images such as those in CIFAR-10 (32×32×3). Therefore, we simplify the architecture to reduce training time while maintaining high accuracy.

## 1.1 Model Architecture

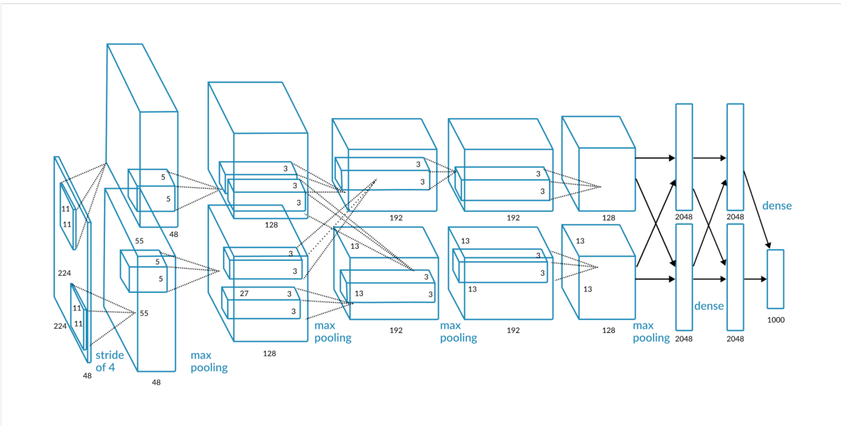


Figure 1: AlexNet - original architecture

Our customized AlexNet consists of five convolutional blocks, as outlined below:

```
# Conv1: input [3, 32, 32] -> output [64, 32, 32]
nn.Conv2d(3, 64, kernel_size=3, stride=1, padding=1),
```

[1]djolin1@charlotte.edu

```
nn.ReLU(inplace=True),
nn.MaxPool2d(kernel_size=2, stride=2),  # output: [64, 16, 16]

# Conv2: input [64, 16, 16] -> output [192, 16, 16]
nn.Conv2d(64, 192, kernel_size=3, padding=1),
nn.ReLU(inplace=True),
nn.MaxPool2d(kernel_size=2, stride=2),  # output: [192, 8, 8]

# Conv3: input [192, 8, 8] -> output [384, 8, 8]
nn.Conv2d(192, 384, kernel_size=3, padding=1),
nn.ReLU(inplace=True),

# Conv4: input [384, 8, 8] -> output [256, 8, 8]
nn.Conv2d(384, 256, kernel_size=3, padding=1),
nn.ReLU(inplace=True),

# Conv5: input [256, 8, 8] -> output [256, 8, 8],
# then pooling to [256, 4, 4]
nn.Conv2d(256, 256, kernel_size=3, padding=1),
nn.ReLU(inplace=True),
nn.MaxPool2d(kernel_size=2, stride=2)
```

## 1.2 Discussion and Results

- **Architecture Overview:** Our network reduces spatial dimensions via pooling while increasing the number of feature maps. This simplified AlexNet has approximately 35.9 million parameters, compared to about 60 million in the original architecture.

- **Dropout Integration:** We incorporated dropout layers (with a rate of 0.5) after the fully connected layers to improve generalization.

- **Performance:** On CIFAR-10, the model achieved a training loss of approximately 0.25 and a validation accuracy around 91%. For CIFAR-100, the validation accuracy was about 68%, reflecting the increased classification challenge.

Figure 2 shows the confusion matrix for CIFAR-10, and Figures 3 and 4 illustrate the training loss and accuracy curves, respectively.

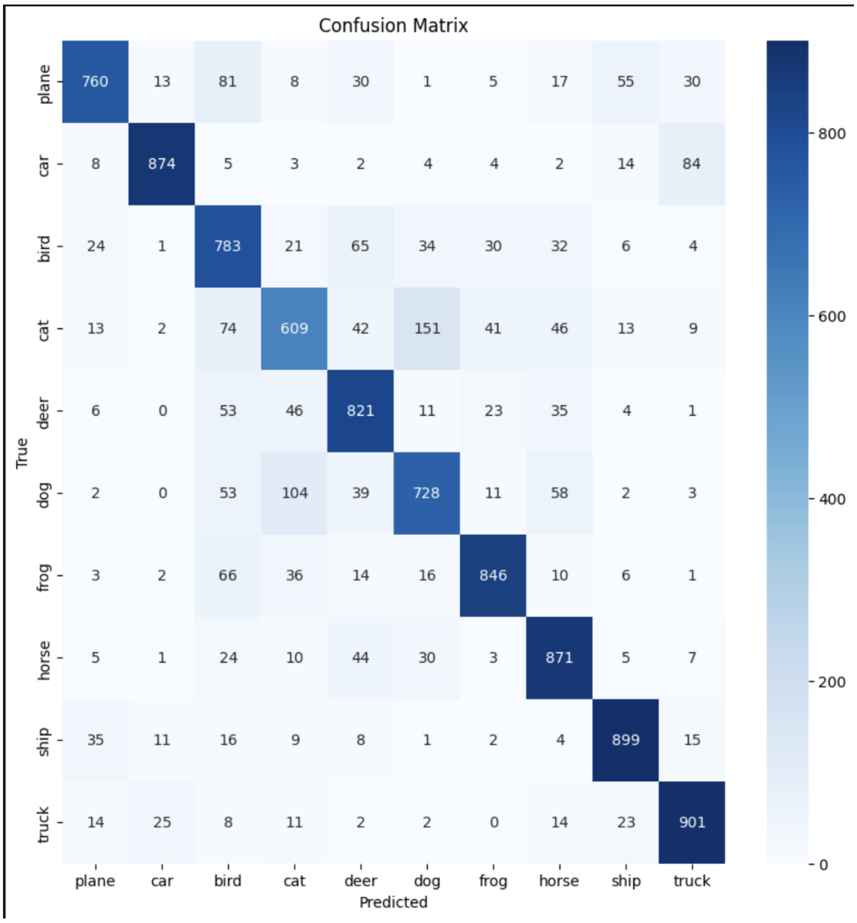


Figure 2: Confusion Matrix for Simplified AlexNet on CIFAR-10

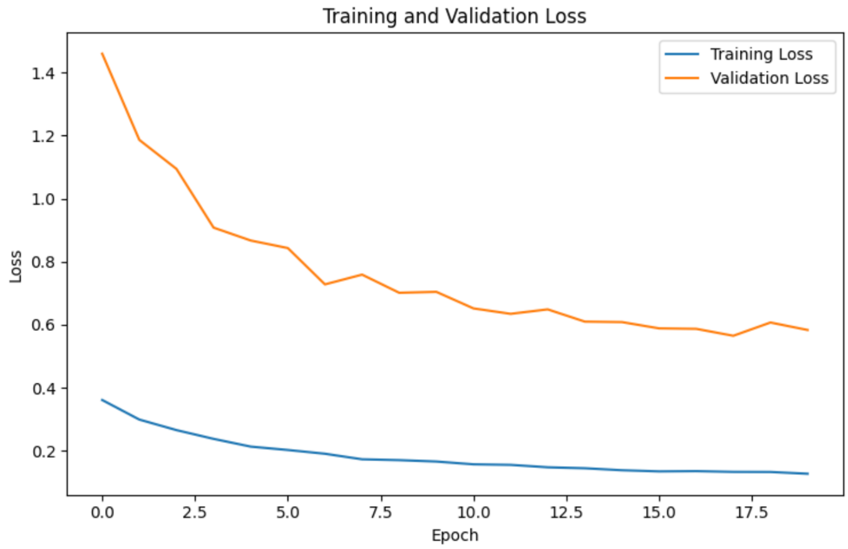


Figure 3: Training and Validation Loss for Simplified AlexNet

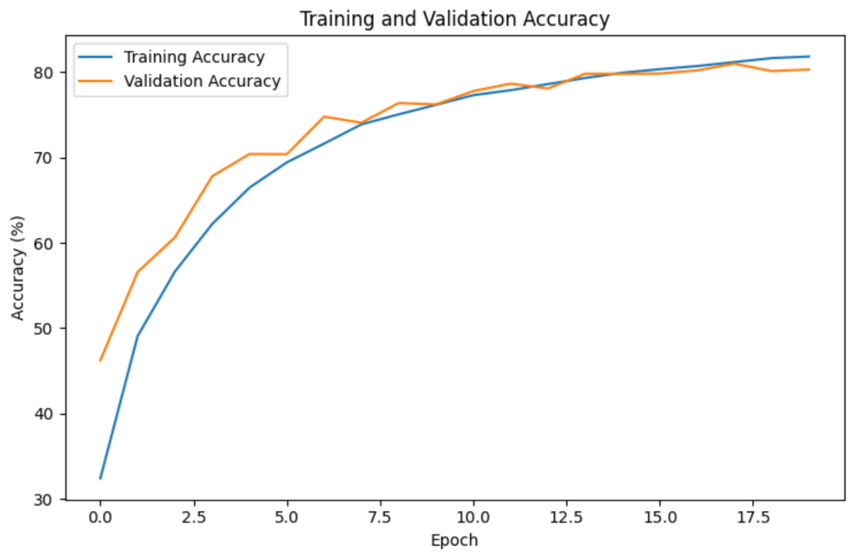


Figure 4: Validation Accuracy for Simplified AlexNet

# 2 Problem 2: Customized VGGNet for CIFAR Datasets

For VGGNet, we adapted the VGG-16 architecture to be more suitable for CIFAR images by reducing both the depth and the number of filters. Our aim was to obtain a model with a parameter count close to that of our simplified AlexNet while maintaining competitive performance.

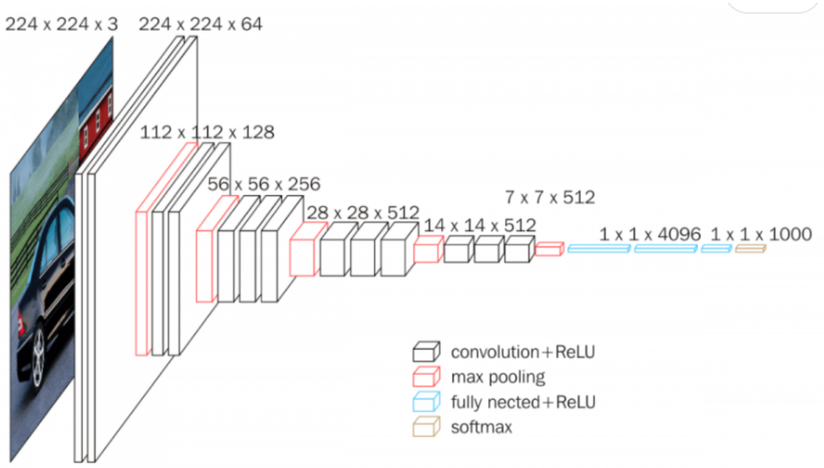


Figure 5: VGG - original architecture

### 2.1 Model Architecture

The modified VGGNet uses several convolutional blocks similar to the original VGG-16 but with fewer filters per layer. Key modifications include:

- Reduced number of filters in each convolutional block.

- Adjustment of the fully connected layers to suit the smaller image resolution.

### 2.2 Discussion and Comparative Results

- **CIFAR-10 Performance:** The customized VGGNet achieved a validation accuracy in the range of 91%–92%, comparable to the simplified AlexNet.

- **CIFAR-100 Performance:** Validation accuracy was around 68%, similar to AlexNet, despite the differences in model architecture.

- **Dropout Effects:** Incorporating dropout further improved generalization and slightly increased validation accuracy, reducing overfitting in both architectures.

- **Size:** The slimmed down VGG Net is similarly sized to AlexNet with 33,646,666 parameters for CIFAR10 and 34,015,396 for CIFAR100

Figures 6 and 7 show the training curves and accuracy for the customized VGGNet on CIFAR-10.

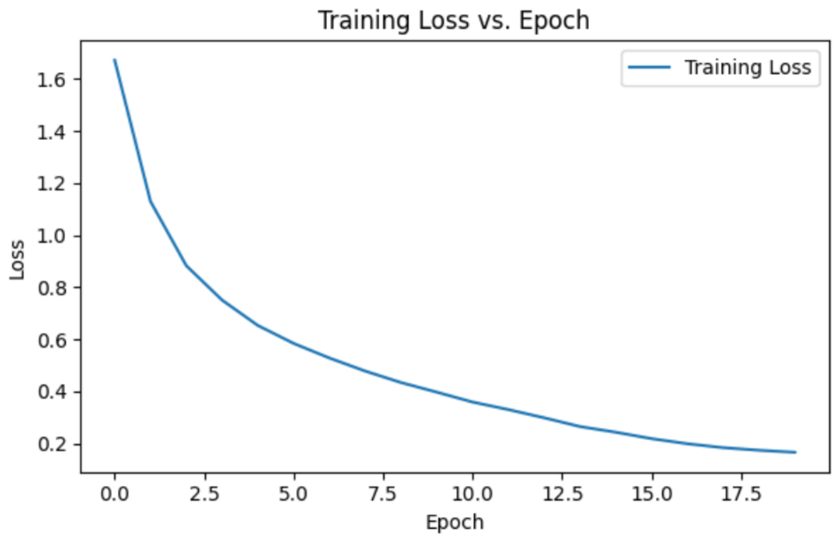


Figure 6: VGG - loss


Figure 7: VGG - accuracy

# 3 Problem 3: ResNet Architectures – ResNet-11 vs. ResNet-18

We also implemented a baseline ResNet-11 model and a deeper ResNet-18 model to compare the effect of increased depth on CIFAR-10 and CIFAR-100 classification.

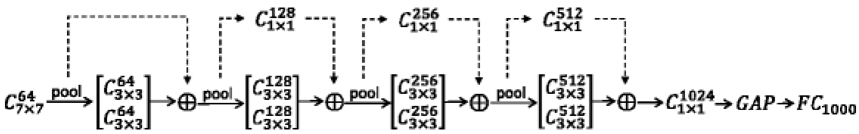


Figure 8: Resnet11

### 3.1 ResNet Architectures

- **ResNet-11:** Our baseline model with 11 layers, designed for CIFAR-10.

- **ResNet-18:** A deeper network with 18 layers, adapted for 32×32 input images. This model has roughly 18 million parameters, compared to about 11 million in ResNet-11.

### 3.2 Training and Evaluation

Both models were trained using the same hyperparameters on CIFAR-10 and CIFAR-100. Figures 9 and 10 compare the training/validation loss and validation accuracy on CIFAR-10.

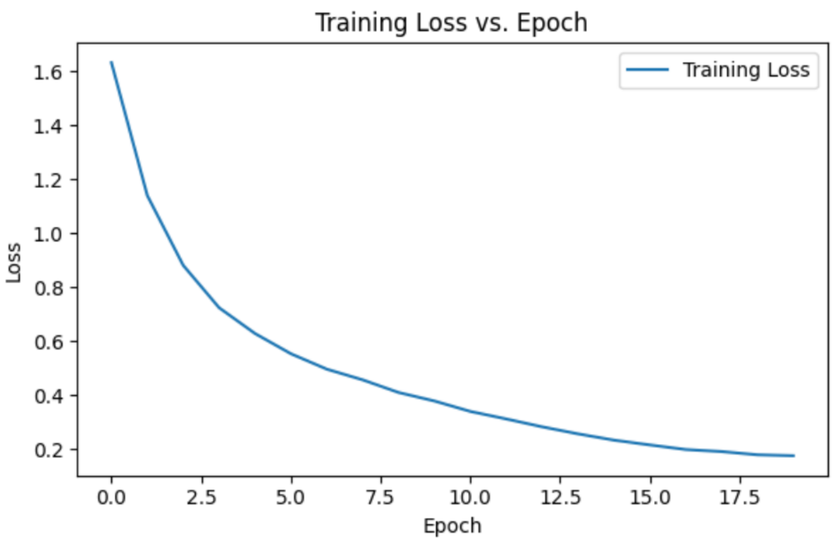


Figure 9: Resnet - loss

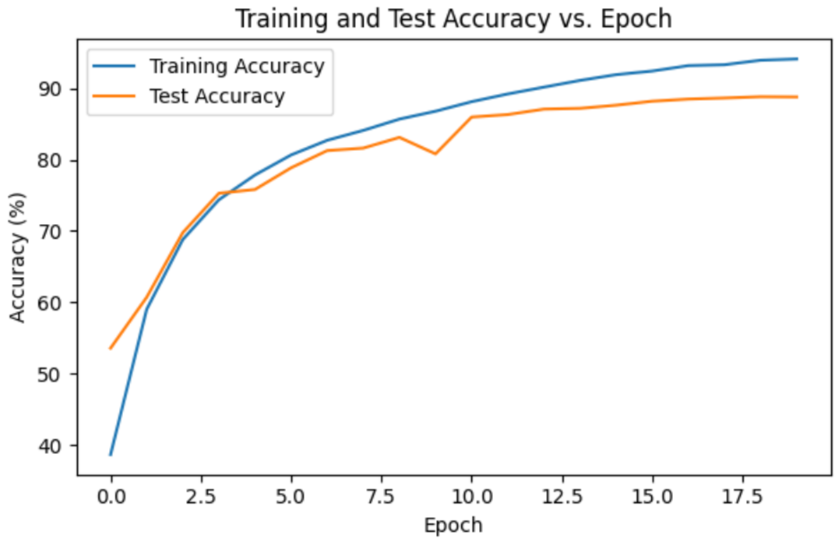


Figure 10: Resnet - accuracy

### 3.3 Results and Analysis

- **CIFAR-10:** ResNet-11 achieved around 91% validation accuracy, while ResNet-18 improved to approximately 93.5%.
- **CIFAR-100:** The accuracies were approximately 63% for ResNet-11 and 67% for ResNet-18.
- **Impact of Depth:** The deeper ResNet-18 demonstrates improved classification performance at the cost of increased computational complexity.
- **Dropout Experiments:** Adding dropout (e.g., with a rate of 0.3) yielded a modest improvement (1–2% increase in validation accuracy) in both models.
- **Model Size:** The baseline RESNET11 model has 4,903,242 total parameters, increasing to 11,173,962 with the implementation of RESNET18, more than doubling in size.

## 4 Conclusion

This homework investigated three deep learning architectures—AlexNet, VGGNet, and ResNet—adapted for CIFAR-10 and CIFAR-100 classification. Our key findings include:

- **Simplified AlexNet:** Reducing the complexity of the original AlexNet led to faster training with high accuracy, particularly when combined with dropout.
- **Customized VGGNet:** With modifications to suit CIFAR image sizes, the VGGNet achieved performance comparable to the simplified AlexNet.
- **ResNet Variants:** Increasing depth from ResNet-11 to ResNet-18 resulted in improved accuracy, demonstrating that deeper networks can better capture complex features, albeit with higher computational costs.

Overall, these experiments highlight the trade-offs between model complexity, training efficiency, and performance. Future work may focus on further architectural optimizations and additional regularization techniques to enhance performance on challenging datasets like CIFAR-100.