CSCI 4922/5922 Lab Assignment Guide

Instructions and Examples

Course Staff

Spring 2026

1 Introduction

The lab assignments for this course are intended to help you develop skills in designing, evaluating, and analyzing deep learning models, while also clearly and rigorously communicating your experimental process and findings.

Your report is a **scientific artifact**. Its purpose is not to narrate what you did in a casual way, nor to reproduce your code, but to allow a knowledgeable reader to:

- Understand your experimental design,
- Reproduce your results,
- Interpret what your results imply.

Accordingly, every report must contain three core sections: **Methods**, **Results**, and **Analysis**. Each serves a distinct purpose and is evaluated independently.

2 Methods Section

The goal of the Methods section is to make your experiments **reproducible**.

Bad Example (Methods)

We used the CIFAR-10 dataset and trained a neural network. We tried a few different learning rates and activation functions and picked the best one. The model worked pretty well overall.

Why this is bad: This description omits dataset splits, model architecture, optimizer details, exact hyperparameters, and reproducibility controls. As a result, the experiment cannot be reproduced.

Good Example (Methods)

We used the CIFAR-10 dataset containing 60,000 color images across 10 classes. The dataset was split into 45,000 training images, 5,000 validation images, and 10,000 test images. All images were normalized to zero mean and unit variance per channel.

We trained a convolutional neural network with three convolutional layers (32, 64, and 128 filters, 3×3 kernels), each followed by ReLU activations and 2×2 max-pooling, followed by two fully connected layers of sizes 256 and 10 with a softmax output.

Models were trained using the Adam optimizer with learning rate 10^{-3}, batch size 128, and for 50 epochs. We fixed the random seed to 42 for reproducibility. We varied the learning rate in {1e-2, 1e-3, 1e-4} and activation function in {ReLU, LeakyReLU}, resulting in six experimental configurations.

Why this is good: This description fully specifies the dataset, preprocessing, model architecture, training procedure, and experimental variables. A reader could reproduce the experiment and understand exactly what was tested.

3 Results Section

The goal of the Results section is to **present empirical findings clearly and accurately**, without interpretation. The Results section may describe patterns and comparisons in the data, but must not attribute causes, mechanisms, or implications to those patterns. Words such as "because", "suggests", "indicates", or "implies" generally belong in the Analysis section.

In addition to tables, the Results section often benefits from:

- **Learning curves** (e.g., training/validation loss vs. epoch) to summarize training behavior,
- **Bar charts** to compare performance across configurations,
- **Confusion matrices** to show which classes are being confused.

Bad Example (Results)

Our model did really well and got around 85% accuracy. ReLU worked better than tanh. The loss also went down nicely during training.

Why this is bad: This uses vague language, provides no exact values, includes interpretation, and presents no figures or tables.

Subtle Bad Example (Results)

Table 1 shows that ReLU performs better than LeakyReLU, particularly at a learning rate of 10^{-3}, because it generalizes better.

Why this is bad: Although this references a table and uses precise language, it introduces a causal explanation ("because it generalizes better") that belongs in the Analysis section.

Good Example (Results)

Table 1 shows the classification accuracy for each activation function and learning rate.

Activation	Learning Rate	Accuracy (%)
ReLU	10^{-2}	82.1
ReLU	10^{-3}	85.7
ReLU	10^{-4}	84.9
LeakyReLU	10^{-2}	81.4
LeakyReLU	10^{-3}	84.6
LeakyReLU	10^{-4}	84.1

Table 1: Test accuracy for each configuration on CIFAR-10.

Across both activation functions, accuracy is highest at a learning rate of 10^{-3} and lower at both 10^{-2} and 10^{-4}. For each learning rate, ReLU achieves higher accuracy than LeakyReLU. The difference between activation functions is smallest at 10^{-2} and largest at 10^{-3}. The overall range of observed accuracies spans from 81.4% to 85.7%. The top-performing configuration is ReLU with learning rate 10^{-3}, while the lowest-performing configuration is LeakyReLU with learning rate 10^{-2}.

Why this is good: This description directs the reader's attention to key patterns in the data, references the table explicitly, and summarizes observable trends without offering explanations or causal claims.

Additional Figure Examples (Results)

Figure 1 shows training and validation loss over epochs for a representative configuration. Figure 2 shows test accuracy for each configuration as a bar chart. Figure 3 shows a confusion matrix for a 5-class example (for brevity).

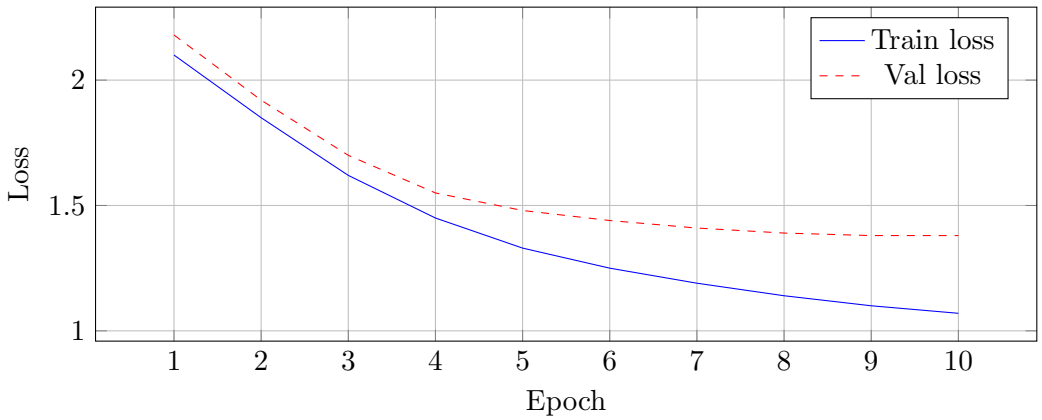

Figure 1: Example learning curves (training and validation loss) over 10 epochs.

Why these figures fit in Results: They present empirical outputs (learning curves, configuration comparisons, and class-level errors) in a labeled, readable form. The accompanying text should

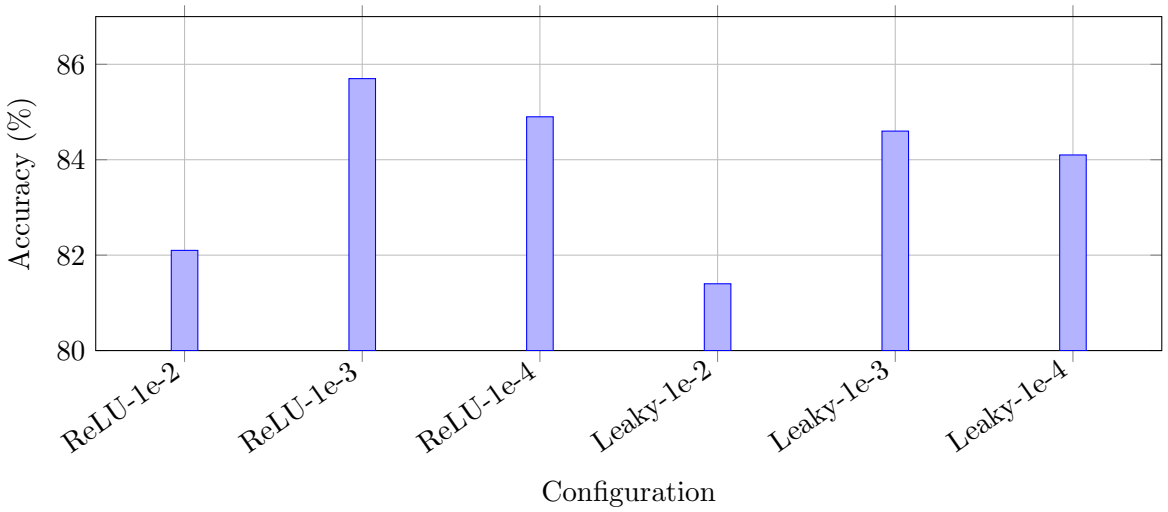

Figure 2: Example bar chart comparing test accuracy across configurations.

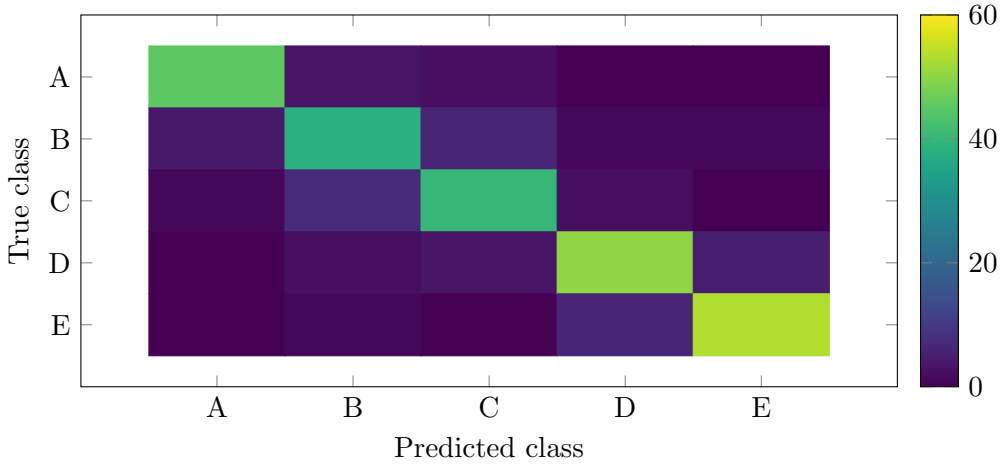

Figure 3: Example confusion matrix (5-class illustration). Larger diagonal values indicate more correct predictions.

describe what is visible in the figure (e.g., which curve is lower, which bar is highest, which off-diagonal cells are large) without providing explanations.

4 Analysis Section

The goal of the Analysis section is to **interpret the results**. Furthermore, a strong analysis connects observed results to mechanisms, theory, or known properties of models and data.
A good analysis typically includes:

- Identification of a trend or pattern,
- A hypothesis about why the trend occurs,
- Evidence from the results that supports the hypothesis,
- A discussion of limitations or alternative explanations,
- Suggestions for follow-up experiments.

A good analysis avoids:

- Restating numbers without interpretation,
- Vague statements like "this worked better" without explanation,
- Overgeneralizing from a single dataset or task.

Bad Example (Analysis)

ReLU worked better because it is a better activation function. A lower learning rate also helped because it trained better. The model might be overfitting a bit.

Why this is bad: These explanations are circular, vague, and not grounded in theory or data. They do not meaningfully explain the observed trends.

Good Example (Analysis)

Models using ReLU outperformed LeakyReLU slightly across all learning rates, with the largest difference at learning rate 10^{-3}. This is consistent with prior work suggesting that ReLU's sparsity promotes better generalization in moderate-depth networks.

Performance peaked at learning rate 10^{-3} and decreased slightly at 10^{-2} and 10^{-4}. This pattern aligns with known optimization behavior in which excessively large learning rates lead to unstable updates, while very small learning rates slow convergence within a fixed training budget.

The confusion matrix in Figure 3 indicates that most errors arise from confusion between specific pairs of classes (off-diagonal cells with relatively large values). This motivates class-specific analysis, such as inspecting representative misclassified examples for those pairs and evaluating whether the classes share visual similarity or dataset imbalance.

Why this is good: This analysis identifies trends, connects them to established principles, and proposes concrete follow-up analysis grounded in the observed data.

5 How to Think About Experimental Design

Each lab is an experiment, not just a programming task. The goal should be viewed as a means to learn why a model behaviors the way it does and not simply to maximize a metric. A well designed experiment isolates a small number of variables and measures their effects while holding others fixed. Changing too many things at once may produce good results, but it does not produce useful scientific insight.

A strong experiment begins with a clear question, such as whether a deeper network improves generalization, whether data augmentation increases robustness, or whether a different optimizer leads to more stable training. That question should determine what is changed and what remains fixed.

Good experimental practice

- Change one factor at a time when possible, such as learning rate, depth, or regularization.
- Keep all other factors fixed and documented.
- Use consistent datasets, splits, and evaluation metrics across runs.
- Run multiple trials when randomness is significant and report averages or variability.

Careful record keeping of hyperparameters, random seeds, and training conditions is also essential for reproducibility. You may consider tools for this such as Weights and Biases and/or finding a suitable workflow that is intuitive for you.

Common experimental pitfalls

- Changing multiple hyperparameters simultaneously and attributing the outcome to only one.
- Reporting only the best performing run.
- Modifying or inspecting the test set during development.
- Drawing conclusions from small performance differences without accounting for variance.

Avoiding these pitfalls is what makes an experiment scientifically meaningful rather than a search for the largest number.

6 Modality-Specific Considerations

Different data modalities impose different statistical structure and failure modes on neural networks. A strong lab report should show awareness of how these properties interact with preprocessing, model design, and evaluation. Below are some examples of various considerations to make in different modalities.

Vision

Image resolution determines how much spatial detail is available to the model, with higher resolutions providing more information but increasing computation and overfitting risk. Data augmentation, such as cropping, flipping, and color jitter, changes the effective training distribution

and strongly affects generalization. Class imbalance is common in vision datasets and can lead to misleading accuracy if not handled properly. Visual similarity between classes also matters, since fine grained classification tasks are more sensitive to resolution, augmentation, and model capacity.

Text

Text data is discrete and sequential. Vocabulary size and tokenization determine what the model can represent and how it handles rare words. Sequence length affects both memory usage and optimization, especially for attention based models. When using pre trained embeddings or language models, it is important to report whether they are frozen or fine tuned, since this choice affects both performance and overfitting.

Audio

Audio data is continuous and temporal. The sampling rate controls how much frequency detail is preserved, while windowing and feature extraction determine how temporal structure is presented to the model. Temporal alignment between inputs and labels is critical. Noise and silence can introduce spurious cues, so their presence and any augmentation strategies should be discussed.

These modality specific choices play a key role in shaping what the model can learn and how it fails, so they should be explicitly reflected in the lab report.

7 Scientific Rigor and Integrity

You are expected to report results honestly and transparently. This includes:

- Reporting negative or null results,
- Not removing outliers without clear justification,
- Not tuning hyperparameters or model choices on the test set,
- Clearly stating limitations and sources of uncertainty,
- Ensuring that all data processing is performed correctly prior to model training.

8 Notes on Data Hygiene

Scientific and deep learning progress depends on understanding why a method works, not merely that it produces a result. A model may appear to perform extremely well even when its success is driven by simple errors that occur upstream of training rather than by the modeling approach itself. A few examples of some common pitfalls are below but this is by no means an exhaustive list.

When combining multiple datasets, it is essential to track the data source and ensure that dataset splits are performed in a source aware manner or deduplicating data prior to splitting the data. Likewise, when observations are not independent, such as repeated visits from the same patient, data must be split in a subject aware way. Failing to do so introduces information leakage between the training and test sets, which can artificially inflate apparent model performance.

Data leakage can also arise during preprocessing, particularly through imputation. If an imputation model is fit on the full dataset prior to splitting, information from the test set is inadvertently incorporated into the training data. As with any form of leakage, this leads to overly optimistic performance estimates that do not reflect true generalization.

If you have questions about these or any other data hygiene best practices you're always welcome to ask in Piazza, Office Hours, or after lecture.

9 Templates

You may use any standard academic report or conference template (e.g., article or proceedings formats). Avoid informal, blog-style, or presentation-based templates. A convenient place to find suitable LaTeX templates is the Overleaf template gallery: Overleaf LaTeX template gallery. Choose a template that emphasizes clarity and readability rather than visual styling.

10 Questions and Support

If you have questions about the lab assignments, report expectations, or course content, please post them on **Piazza**. This allows the entire class to benefit from the discussion and helps the course staff provide consistent answers.

You are also strongly encouraged to take advantage of the **office hours** offered throughout the week. Office hours are a good opportunity to:

- Ask clarifying questions about the material,
- Get feedback on your experimental design or analysis approach,
- Discuss any difficulties you are encountering with the labs.

Reaching out early when something is unclear will save you time and help you get more out of the course.