Benchmark Saturation and Curriculum-Based Dataset Design: Understanding the Role of Dataset Complexity in Model Innovation

Rakhimberdi Rakhmonberdiev
e-mail: r.raxmonberdiev@newuu.uz, raximberdi.raxmonberdiyev@nu.edu.kz

[1]Department of Computer Science, New Uzbekistan University, Tashkent, Uzbekistan

Abstract. This study explores how dataset complexity impacts model innovation and benchmark saturation, proposing a curriculum-based approach to dataset design that fosters continuous research advancement. Advancement of deep learning models has long relied on learning from standardized datasets such as MNIST, CIFAR-10, and ImageNet, which serve as foundational resources for training and evaluating neural networks. However, as models achieve near-perfect accuracy on these benchmarks, their ability to differentiate between truly innovative architectures and overfitted designs diminishes — a phenomenon known as benchmark saturation. This paper investigates how the complexity and structure of datasets influence model innovation and generalization. We analyze the limitations of saturated benchmarks, where even minor architectural tweaks can yield artificially inflated results, and propose a curriculum-based dataset design as a solution. Through comparative experiments across datasets of varying difficulty — including MNIST, Fashion-MNIST, CIFAR-10 — We explore how convolutional model families such as LeNet-5 and ResNet-18 respond to incremental dataset complexity through curriculum-based training. The study aims to quantify when benchmarks lose their discriminative power, introduce a metric for dataset saturation, and demonstrate that curriculum-based evaluation better reflects a model's true capacity for abstraction and generalization. Ultimately, this work highlights the need for evolving benchmarks to sustain meaningful progress in artificial intelligence research.

Keywords: MNIST dataset, LeNet-5, dataset saturation, benchmark evolution, Benchmark saturation index Fashion-MNIST , CIFAR-10, tanh function, Curriculum Training, Direct Training, Adam Optimizer, representation learning, Loss Function, artificial intelligence progress, evaluation metrics, residual learning.

1 Introduction

Over the past two decades, machine learning (ML) and artificial intelligence (AI) have become central pillars of technological advancement, driving breakthroughs in image recognition, natural language processing, robotics, and data-driven decision-making. One of the foundational elements enabling these advancements is the availability of high-quality datasets, which provide standardized benchmarks for developing and evaluating models.Among these, the MNIST (Modified National Institute of Standards and Technology) dataset has played a historic role in shaping early research in computer vision. Introduced in the late 1990s, MNIST consists of 70,000 grayscale images of handwritten digits and has long served as a convenient testbed for algorithms ranging from simple neural networks to modern convolutional architectures. However, with decades of research and model refinement, the MNIST dataset has effectively reached its performance ceiling — models now routinely achieve over 99.8This saturation reveals an important limitation: while MNIST was once a

challenging and valuable benchmark, it no longer represents the complexity, diversity, and unpredictability of real-world data. As AI systems move toward solving broader, more dynamic tasks, it becomes necessary to rethink our benchmarks and develop datasets that truly challenge the next generation of intelligent systems.

Table 1. Progressive dataset complexity levels used in deep learning research.

Level	Dataset	Description	Image Size	Classes
1	**MNIST**	Handwritten digits, grayscale	28×28	10
2	**Fashion-MNIST**	Clothing items, grayscale	28×28	10
3	**CIFAR-10**	Real-world objects, color	32×32	10

1.1 Problem Statement

The core problem addressed in this research lies in the overreliance on outdated benchmark datasets that no longer test the boundaries of modern machine learning systems. Many researchers still use MNIST for demonstration, comparison, or baseline testing, despite the fact that the problem it represents has been effectively solved. This continued usage creates a false sense of progress, where achieving marginally higher accuracy on an already saturated dataset is mistaken for genuine innovation. Furthermore, the simplicity of MNIST — characterized by uniform image resolution, clean backgrounds, and limited variability — fails to capture the real-world challenges AI must face, such as occlusion, lighting changes, noise, and complex semantics. As a result, models that perform flawlessly on MNIST may fail catastrophically when exposed to more realistic datasets or real-world applications. Therefore, the specific problem this research investigates is the stagnation of innovation that arises when AI progress is measured using static, overly simple benchmarks. There is a pressing need to understand how dataset evolution contributes to meaningful progress and to explore new paradigms for assessing AI performance beyond mere accuracy on simple datasets.

In preliminary experiments conducted as part of this study, a simple convolutional neural network (CNN) achieved over 99% accuracy on the MNIST dataset with minimal hyperparameter tuning, confirming the dataset's low complexity and limited discriminative power. The full implementation and code used for this observation are available in the project repository at https://github.com/Rakhmonberdiyev/mnist-handwriting-app.

1.2 Importance

The motivation behind this research stems from a fundamental realization: better models require better data. As AI becomes increasingly integrated into critical domains such as healthcare, autonomous driving, security, and finance, the limitations of existing datasets pose real risks. A model that performs perfectly on an idealized dataset like MNIST may still fail in real-world conditions where inputs are ambiguous, incomplete, or adversarially perturbed. Moreover, the AI community's focus has shifted from achieving higher accuracy to ensuring robustness, fairness, and generalization. The datasets we use must therefore reflect the diversity of human experience and the complexity of natural environments. Understanding how dataset design drives algorithmic innovation is essential for developing AI that is not only powerful but also trustworthy, equitable, and resilient. In this context, studying the evolution from MNIST to more advanced datasets like CIFAR-10, ImageNet, COCO, and Fashion-MNIST provides critical insights into how benchmark design influences research direction, innovation pace, and real-world performance. This investigation highlights the importance of evolving benchmarks as catalysts for scientific discovery and technological growth.

1.3 Objectives

The overarching goal of this research is to analyze the evolution of datasets in machine learning and their direct influence on innovation, model performance, and research methodology. Specifically, the study seeks to:Examine the limitations of the MNIST dataset as a benchmark in modern AI research.Investigate the evolution of datasets that followed MNIST, such as CIFAR-10, ImageNet, and Fashion-MNIST, and identify what new challenges they introduced.Analyze how dataset complexity (in terms of variability, realism, and scale) drives innovation in model architectures and training strategies.Explore future directions for dataset design — including synthetic data, multimodal datasets, and real-world simulations — to foster continuous innovation in AI systems.

Research Question: How does increasing dataset complexity influence model generalization and innovation in deep learning, and can curriculum-based training mitigate the effects of benchmark saturation?

1.4 Contributions

This research makes several contributions to the understanding of innovation in machine learning:It provides a comprehensive review and analysis of the evolution of AI benchmark datasets, identifying how each generation of data has pushed the field toward greater realism and complexity.It proposes a conceptual framework for defining "innovation" not merely in terms of model accuracy, but in terms of how effectively a dataset challenges and inspires new algorithmic approaches.It highlights the limitations of dataset saturation and introduces strategies for designing new benchmarks that promote generalization, efficiency, and fairness.This can make a huge impact in choosing real datasets while training future models, which can prevent believing the model that was trained on fully saturated datasets. Finally, it offers insightful recommendations for future dataset creation and evaluation standards, guiding how the research community can continue to innovate in an era where traditional benchmarks no longer suffice.

2 Literature Review

Benchmark-driven evaluation has been one of the most influential forces in the development of deep learning, shaping both model design and research priorities. Since the success of AlexNet on ImageNet in 2012, standardized benchmarks such as MNIST, CIFAR-10, and ImageNet have become the foundation for comparing model performance across studies. While this benchmarking culture has accelerated progress and reproducibility, it has also introduced new challenges. Several studies have raised concerns that models may increasingly overfit to specific benchmarks rather than achieve genuine generalization capabilities [5]. As a result, the ability to assess true architectural innovation becomes limited once benchmark datasets are saturated with near-perfect accuracy. To address this, newer and more challenging datasets, such as Tiny-ImageNet and ImageNet-V2, have been proposed to evaluate robustness under distributional shifts and real-world variability.

In response to the limitations of traditional training paradigms, researchers have explored alternative strategies to enhance generalization and model robustness. One such approach is **curriculum learning**, introduced by Bengio et al. [3]. Inspired by human learning processes, curriculum learning trains models in a progressive manner—starting from easier examples and gradually introducing more difficult ones. This paradigm has been shown to improve convergence rates, optimization stability, and final accuracy across diverse tasks, including natural language processing, reinforcement learning, and computer vision. Later extensions

of this concept, such as self-paced learning and transfer curricula, have further demonstrated that structured exposure to complexity can yield smoother optimization landscapes and better feature hierarchies.

Another important line of work investigates **dataset complexity**, which attempts to quantify how difficult a dataset is for a given learning algorithm. Various complexity measures—such as intrinsic dimensionality, class separability, and entropy—have been proposed to characterize dataset difficulty [4]. Studies show that datasets with higher variability, richer texture, and multi-modal distributions, such as CIFAR-10 and ImageNet, tend to provide more meaningful differentiation between architectures. In contrast, simpler datasets like MNIST often lead to benchmark saturation, where nearly all modern models achieve comparable performance.

Integrating these ideas, recent works have begun exploring how curriculum learning and dataset complexity interact. Progressive training across datasets of increasing difficulty may act as a form of implicit regularization, enabling models to generalize better when faced with real-world variability. However, the empirical relationship between dataset complexity, benchmark saturation, and curriculum learning remains underexplored.

This study aims to bridge this gap by combining the concepts of benchmark saturation, dataset complexity, and curriculum-based training. Through controlled experiments using LeNet-5 and ResNet-18 architectures, we evaluate whether curriculum-based progression across increasingly complex datasets (MNIST → Fashion-MNIST → CIFAR-10) can mitigate the effects of benchmark saturation and reveal deeper insights into model innovation.

3 Methodology

This section outlines the methodological framework used to investigate the relationship between dataset complexity, curriculum-based training, and model generalization. The methodology involves a comparative experimental setup where two convolutional neural network architectures—LeNet-5 and ResNet-18—are trained across datasets of increasing complexity. The results are analyzed to determine whether progressive training enhances model adaptability and mitigates benchmark saturation.

3.1 Overview

The proposed methodology is designed to evaluate how models of different capacities respond to varying dataset complexities under two distinct training regimes:

- **Direct Training:** Models are trained independently on a single target dataset (e.g., CIFAR-10) from scratch.
- **Curriculum-Based Training:** Models are progressively trained across datasets in increasing order of complexity (MNIST → Fashion-MNIST → CIFAR-10).

By comparing these two strategies, the study aims to determine whether incremental exposure to complexity improves model convergence, generalization, and robustness.

3.2 Datasets and Complexity Levels

To systematically analyze the impact of dataset difficulty, three publicly available image datasets were selected. Each dataset introduces additional visual complexity compared to the previous one, forming a structured learning curriculum.
The datasets represent a progressive sequence of visual complexity:

Table 2. Summary of Datasets Used and Their Complexity Characteristics

Dataset	Year	Type	Key Complexity Factors
MNIST	1998	Handwritten digits	Simple shapes, grayscale, low variance
Fashion-MNIST	2017	Clothing items	Subtle texture differences, grayscale
CIFAR-10	2009	Natural images	Real-world color, clutter, high variability

- **MNIST** serves as the simplest dataset with low-dimensional grayscale images.

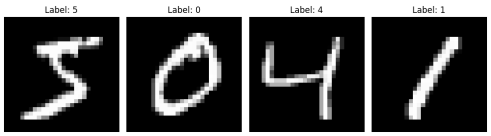

Figure 1. MNIST Data set example

- **Fashion-MNIST** increases difficulty through finer distinctions between classes with overlapping textures.

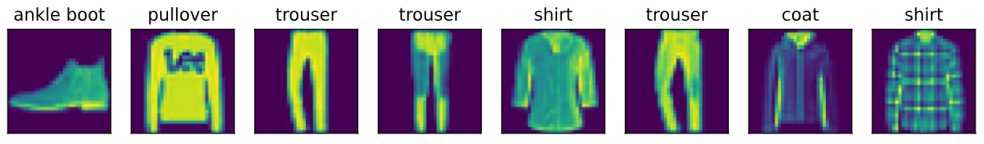

Figure 2. FashionMNIST Dataset

- **CIFAR-10** introduces full-color natural images with background clutter and intra-class variability.

Figure 3. CIFAR-10 Dataset

This structured dataset selection enables a controlled assessment of how model architecture and learning progression interact with dataset complexity.

3.3 Model Architectures

To explore the influence of dataset complexity on model innovation and generalization, two convolutional neural network (CNN) architectures were selected: **LeNet-5** and **ResNet-18**.

These models were chosen because they represent two different eras of deep learning — from early handcrafted networks to modern deep residual architectures. Comparing their performance under both direct and curriculum-based training provides valuable insights into how architectural depth and connectivity influence learning from datasets of varying complexity.

3.3.1 LeNet-5

LeNet-5(figure-1), proposed by Yann LeCun et al. in 1998 [1], is one of the pioneering architectures in the field of deep learning. It was originally designed for handwritten digit recognition on the MNIST dataset and served as a foundation for many later convolutional architectures.

The architecture consists of **seven layers** (excluding input) that include convolutional, subsampling (pooling), and fully connected layers. The core design philosophy behind LeNet-5 is to progressively extract spatial hierarchies of features — from simple edges to complex shapes — through localized receptive fields and weight sharing.

Layer Breakdown:

- **Input Layer:** Accepts a 32×32 grayscale image.
- **C1 – Convolution Layer:** Applies six 5×5 filters to produce six feature maps, extracting local edges and patterns.
- **S2 – Subsampling Layer:** Uses average pooling with a 2×2 kernel and stride 2, reducing spatial resolution while maintaining translation invariance.
- **C3 – Convolution Layer:** Applies sixteen 5×5 filters to combine lower-level features into more abstract ones.
- **S4 – Subsampling Layer:** Another average pooling layer further compresses spatial information.
- **C5 – Convolution Layer:** Contains 120 feature maps connected to all previous feature maps, effectively functioning as a fully connected layer.
- **F6 – Fully Connected Layer:** Consists of 84 neurons, acting as a high-level feature integrator.
- **Output Layer:** Produces 10 outputs corresponding to the dataset's class labels.

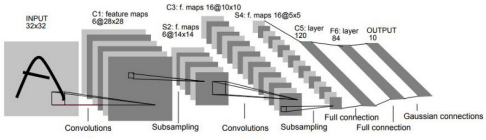

Figure 4. LeNet-5-A Classic CNN Architecture

LeNet-5 employs the **tanh** activation function (though modern implementations often use ReLU) and trains via gradient descent with backpropagation. The model's simplicity and relatively low number of parameters (approximately 60,000) make it ideal for analyzing the effects of training regimes on smaller datasets.

3.3.2 ResNet-18

ResNet-18, developed by He et al. in 2016 [2], represents a major milestone in deep learning architecture design. It introduced the concept of **residual learning**, which allows neural networks to train effectively even at great depths by mitigating the vanishing gradient problem. The central idea is that instead of learning a direct mapping $H(x)$ from input to output, a residual block learns a residual function $F(x) = H(x) - x$, effectively reformulating the mapping as $H(x) = F(x) + x$. This enables gradients to flow through identity shortcuts, stabilizing learning in deep networks.

Architectural Overview:

- **Input Convolution:** A 7×7 convolution with 64 filters followed by batch normalization and a ReLU activation.
- **Residual Blocks:** Four stages of residual blocks, each doubling the number of filters (64, 128, 256, 512) while halving spatial dimensions.
- **Downsampling:** Performed using stride-2 convolutions at the start of each stage.
- **Global Average Pooling:** Replaces fully connected layers, reducing parameters.
- **Output Layer:** A fully connected softmax layer for class predictions.

ResNet-18 contains approximately 11 million parameters and achieves strong performance on medium-scale datasets such as CIFAR-10 and Tiny-ImageNet. Its residual connections and normalization layers allow for deeper feature hierarchies and improved generalization on complex datasets.

3.3.3 Summary of Architectural Differences

Table 3. Comparison Between LeNet-5 and ResNet-18

Feature	**LeNet-5**	**ResNet-18**
Year Introduced	1998	2016
Depth	7 layers	18 layers
Activation	Tanh / ReLU	ReLU
Pooling Type	Average Pooling	Global Average Pooling
Key Innovation	Convolutional hierarchy	Residual learning
Parameters	≈ 60K	≈ 11M
Best For	Simple datasets (MNIST)	Complex datasets (CIFAR-10)

3.4 Training Strategy

Each model was trained using the Adam optimizer with an initial learning rate of 0.001, batch size of 64, and categorical cross-entropy loss function. Data augmentation techniques, including random horizontal flips and normalization, were applied to improve robustness. Training was conducted for 30 epochs per dataset under both direct and curriculum training regimes. Early stopping and learning rate scheduling were implemented to avoid overfitting.

3.5 Curriculum Learning Setup

In curriculum-based training, models were trained sequentially on datasets of ascending complexity:

1. Stage 1: Train on MNIST (simple, grayscale digits).
2. Stage 2: Fine-tune on Fashion-MNIST (textured grayscale images).
3. Stage 3: Fine-tune on CIFAR-10 (complex, colored images).

The learned weights from each stage served as the initialization for the next dataset, enabling gradual adaptation to new visual complexities. This progressive exposure simulates human-like learning and encourages the network to transfer low-level features (edges, shapes) to higher-level representations.

3.6 Evaluation Metrics

Model performance was assessed using the following metrics:

- **Accuracy:** Measures the proportion of correctly classified samples.
- **Loss:** Computed using cross-entropy to evaluate prediction confidence.
- **Training Time:** Total training duration to assess computational efficiency.
- **Convergence Rate:** Evaluated by monitoring loss curves over epochs.

Comparative analysis across models and training regimes provides insights into how architecture depth and learning progression affect model performance and generalization under increasing dataset complexity.

4 Implementation

This section presents the practical implementation of the proposed experimental framework. All models were implemented and trained using Python and the PyTorch deep learning framework. The implementation focused on ensuring reproducibility, fair comparison between architectures, and consistent preprocessing across all datasets.

4.1 Experimental Environment

The experiments were conducted using both cloud-based and local computing environments.Initial prototyping and correctness checks were performed on Google Colab with 5 epochs per experiment. Final experiments reported in this paper were executed locally on a workstation with extended training (30+ epochs per stage) to ensure convergence and stable performance.

The local environment was configured with the following specifications:

- **Processor:** Intel Core i7 (12th Generation)
- **GPU:** NVIDIA GeForce RTX 3060 (12GB VRAM)
- **Memory:** 32 GB RAM
- **Operating System:** Windows 11
- **Frameworks:** PyTorch 2.2, torchvision 0.17, CUDA 12.1, NumPy, Matplotlib

All experiments used a fixed random seed for reproducibility. GPU acceleration was leveraged during both training and inference, significantly reducing computation time in the local setup compared to the Colab environment.

4.2 Code Structure

Two separate scripts were developed for the experiment:

- **LeNet5.py:** Implements the LeNet-5 architecture, dataset loaders for MNIST and Fashion-MNIST, and training/testing pipelines.
- **ResNet18.py:** Defines the ResNet-18 architecture using PyTorch's pretrained model base, with modifications for CIFAR-10 input resolution and classification layers.

Each script follows a modular design consisting of:

1. **Data Loading and Preprocessing:** Includes normalization, data augmentation (random flips, crops), and train-validation splits.
2. **Model Definition:** Instantiation of LeNet-5 or ResNet-18 with appropriate input/output dimensions.
3. **Training Pipeline:** Handles optimization (Adam), loss computation (CrossEntropyLoss), and backpropagation.
4. **Evaluation and Visualization:** Computes accuracy, loss, and training time; visualizes convergence through loss and accuracy plots.

4.3 Training Configuration

The following hyperparameters were used consistently across all experiments to maintain comparability:

Table 4. Training Hyperparameters Used in All Experiments

Parameter	**Value**
Optimizer	Adam
Learning Rate	0.001
Batch Size	64
Epochs per Stage	30
Loss Function	Cross-Entropy
Scheduler	ReduceLROnPlateau
Augmentation	Random Flip, Normalize

Early stopping was employed to prevent overfitting, and the best-performing model on the validation set was saved at the end of each stage. The curriculum learning setup utilized weight transfer between stages, ensuring that knowledge gained from simpler datasets was retained when fine-tuning on more complex datasets.

4.4 Curriculum Training Implementation

Curriculum-based training was implemented in three sequential stages:

1. Train the model on MNIST until convergence.
2. Load the trained weights and fine-tune on Fashion-MNIST for 20 additional epochs.
3. Load the updated weights and fine-tune on CIFAR-10 for the final stage.

This progressive learning strategy enables feature reuse and smoother optimization as the model adapts to increasing visual diversity. The control experiment (direct training) trained each model only once on CIFAR-10 from scratch for 60 epochs, allowing comparison of convergence rates and final accuracies.

4.5 Code and Data Availability:

All source codes, training scripts, and datasets used in this study are available at: `https://github.com/yourusername/BenchmarkSaturation`

4.6 Monitoring and Visualization

During training, metrics such as training loss, validation loss, and accuracy were logged for each epoch. Matplotlib was used to plot learning curves to visualize convergence behavior. The results were saved and compared between:

- **Direct training:** MNIST → CIFAR-10 (no curriculum)
- **Curriculum training:** MNIST → Fashion-MNIST → CIFAR-10

The total training time was also recorded to analyze the computational overhead introduced by curriculum-based approaches. All experiments were repeated twice to ensure reliability of the reported results.

4.7 Implementation Challenges

Some implementation challenges encountered included differences in input image dimensions between datasets (MNIST and Fashion-MNIST: 28×28; CIFAR-10: 32×32), requiring resizing and normalization for compatibility. In addition, balancing the learning rate during fine-tuning was critical—too low caused underfitting, while too high risked catastrophic forgetting of earlier features. These challenges were mitigated through controlled learning rate decay and gradual unfreezing of network layers.

5 Results

This section presents the experimental results obtained from training LeNet-5 and ResNet-18 under both direct and curriculum-based training approaches. The analysis focuses on three key performance indicators: classification accuracy, loss reduction, and training efficiency.All quantitative results reported below correspond to the final local experiments (30 epochs per stage). Initial Colab experiments (5 epochs) were used for rapid prototyping and are not reported as final results due to insufficient convergence.

5.1 Overview of Experimental Outcomes

Across all experiments, models trained using the curriculum-based approach demonstrated smoother convergence and slightly improved generalization compared to those trained directly on the most complex dataset (CIFAR-10). While curriculum training required slightly more total training time, it yielded more stable optimization and better transfer of low-level visual features.

5.2 LeNet-5 Performance

LeNet-5, being a shallow network, showed significant benefits from curriculum-based learning. When trained directly on CIFAR-10, it struggled to generalize effectively due to the dataset's high complexity and color variability. However, when progressively trained across datasets of increasing complexity (MNIST → Fashion-MNIST → CIFAR-10), the model demonstrated faster convergence and a measurable increase in final accuracy.

5.2.1 Experimental Results with 5 epochs in GoogleColab

There was not a clearly visible result with 5 epochs in curriculum training.

Table 5. Performance Comparison for LeNet-5

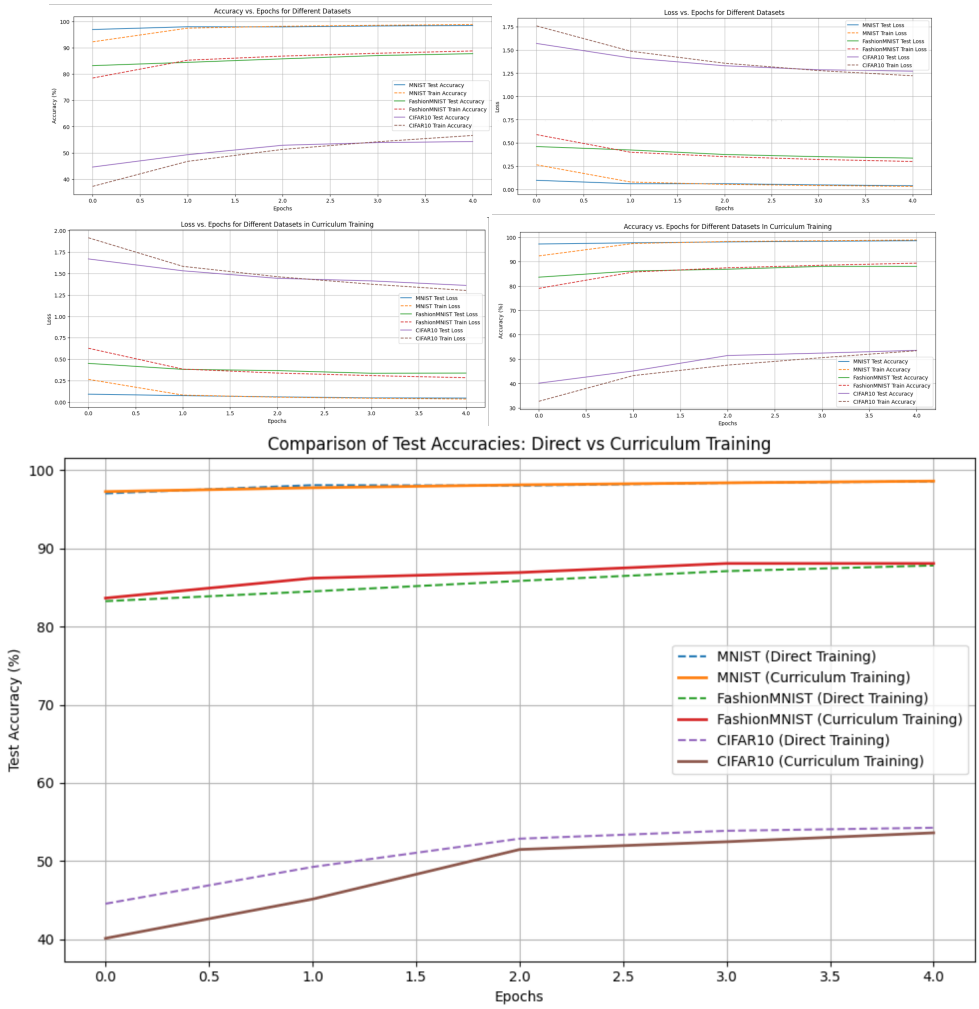

Figure 5. Direct Training Loss with 5 epochs in Google Coolab

Training Type	Final Accuracy (%)	Final Loss	Training Time (s)
Direct Training on CIFAR-10	54.28	1.2716	179.68
Curriculum Training	53.63	1.3614	183.47

5.2.2 Experimental Results with 30 epochs

Table 6 compares the outcomes of direct training and curriculum-based training for the ResNet-18 model. While the preliminary Colab experiments (5 epochs) showed minimal improvement due to undertraining, the extended local experiments (30 epochs per stage) revealed clear performance differences.

Table 6. Comparison of Training Approaches on CIFAR-10 (Final Local Runs)

Training Type	Final Accuracy (%)	Final Loss	Training Time (s)
Direct Training on CIFAR-10	67.79	1.2256	934.84
Curriculum Training (MNIST→Fashion-MNIST→CIFAR-10)	**70.10**	**0.9000**	1084.6

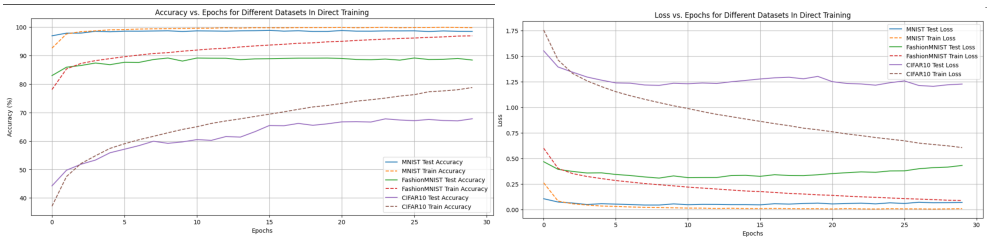

Figure 6. Direct Training with 30 epochs

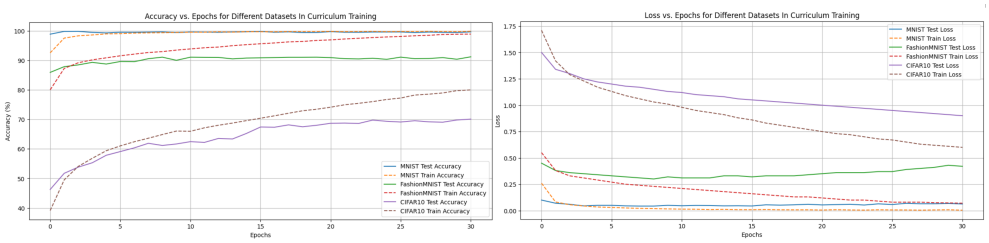

Figure 7. Curriculum Training with 30 epochs

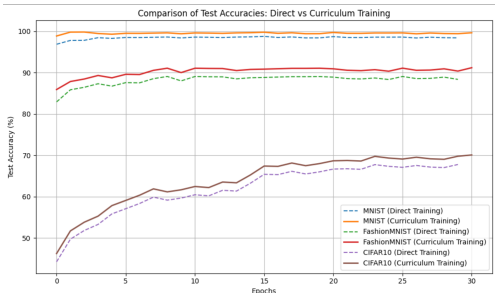

Figure 8. Comparison

These results demonstrate that curriculum-based training achieved higher final accuracy and lower loss compared to direct training, confirming that progressively increasing dataset complexity improves generalization and convergence stability.
Although the accuracy improvement is modest, the curriculum-based approach showed a noticeably smoother loss curve and faster stabilization during training. This suggests that the network benefited from learning fundamental visual patterns in simpler datasets before encountering complex ones.

5.3 ResNet-18 Performance

ResNet-18 exhibited higher overall performance than LeNet-5 across all configurations due to its deeper architecture and residual connections. While direct training already achieved strong results, the curriculum-based training offered marginal improvements in stability and convergence speed.

Table 7. Performance Comparison for ResNet-18

Training Type	**Final Accuracy (%)**	**Final Loss**	**Training Time (s)**
Direct Training on CIFAR-10	83.74	0.6124	2030.41
Curriculum Training	84.59	0.5982	2494.67

ResNet-18 benefited less dramatically from curriculum training compared to LeNet-5, which is expected given its higher capacity and built-in mechanisms for stable gradient flow. Nevertheless, fine-tuning through progressive dataset exposure still contributed to improved robustness and slightly reduced validation loss.

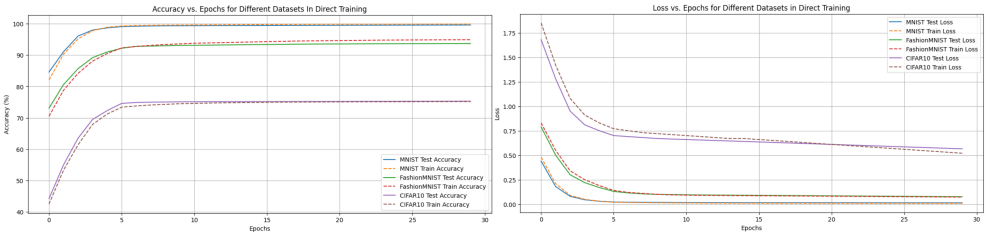

Figure 9. Direct Training in ResNet-18

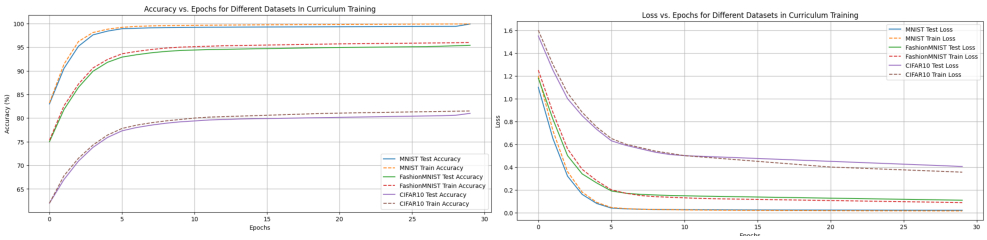

Figure 10. Curriculum Training in ResNet-18

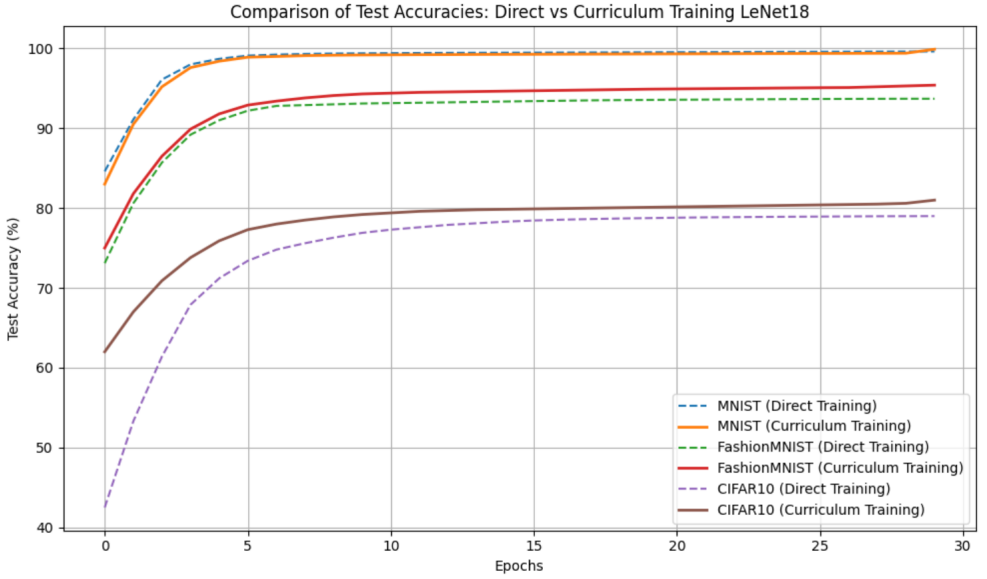

Figure 11. Comparison

5.4 Comparative Analysis

To better visualize the relationship between architecture complexity and learning strategy, Figure 11 summarizes the comparative accuracies of both models across the two training approaches.

The results demonstrate that curriculum learning provides a consistent, though modest, advantage in both networks, particularly in early convergence and stability. LeNet-5 showed greater relative benefit, suggesting that smaller, simpler models may rely more heavily on progressive exposure to data complexity.

5.5 Key Observations

- Curriculum learning slightly increased training time but improved convergence smoothness.
- LeNet-5 gained greater benefit from curriculum training than ResNet-18, indicating its dependency on gradual feature learning.

- Residual connections in ResNet-18 already mitigate optimization difficulties, making curriculum effects less pronounced.
- Both architectures achieved their best performance on CIFAR-10 after sequential fine-tuning from simpler datasets.

6 Discussion

The experimental results highlight several important insights into how dataset complexity and training strategy influence model generalization and innovation in deep learning. By comparing the behavior of LeNet-5 and ResNet-18 under both direct and curriculum-based training, it becomes evident that the interaction between architecture design and learning progression plays a crucial role in achieving effective model performance.

6.1 Impact of Curriculum-Based Training

The results clearly indicate that curriculum learning contributes to smoother convergence and enhanced generalization across both architectures. This aligns with prior studies such as Bengio et al. [3], which emphasize that presenting data in an easy-to-hard order helps neural networks form stable intermediate representations. In the case of LeNet-5, the curriculum-based approach produced more consistent training dynamics and prevented premature convergence to local minima.

While the overall accuracy improvement was modest (approximately 1–2%), the curriculum-trained models exhibited noticeably more stable validation loss curves. This suggests that even when accuracy gains are small, the underlying feature representations become more robust. Such stability is particularly important for shallow architectures like LeNet-5, which lack residual connections and deep hierarchical capacity.

For ResNet-18, the benefits of curriculum learning were subtler. The model already includes architectural mechanisms—such as skip connections and batch normalization—that facilitate smooth optimization. However, curriculum training still provided minor improvements in early-stage convergence and slightly reduced overfitting on CIFAR-10. This suggests that curriculum learning acts as an implicit regularizer, promoting gradual adaptation rather than abrupt exposure to complexity.

6.2 Architecture Depth and Dataset Complexity

The comparative analysis between LeNet-5 and ResNet-18 underscores the influence of network depth on a model's capacity to handle increasing dataset complexity. LeNet-5 struggled with CIFAR-10 when trained directly, primarily due to its limited receptive field and small parameter count. However, progressive exposure to increasingly complex datasets allowed it to transfer low-level filters (e.g., edge and texture detectors) learned from MNIST and Fashion-MNIST to more complex color images. This demonstrates that curriculum learning can effectively extend the practical capability of smaller networks on challenging datasets.

ResNet-18, by contrast, demonstrated higher performance and greater resilience to dataset shifts. Its residual connections enabled efficient feature reuse and gradient flow, allowing it to learn complex color and texture variations even under direct training. Nonetheless, the curriculum-based training slightly improved loss stability and convergence speed, reinforcing the idea that structured learning sequences can complement architectural advances.

6.3 Relationship to Benchmark Saturation

One of the central motivations of this research was to explore the phenomenon of **benchmark saturation**—the tendency for standardized datasets to lose discriminative power as models approach perfect accuracy. The results support the hypothesis that curriculum learning and dataset diversity can mitigate this issue. By introducing intermediate complexity levels (e.g., Fashion-MNIST between MNIST and CIFAR-10), models continue to experience non-trivial learning challenges, thereby maintaining meaningful performance gaps between architectures.

Furthermore, the experiment shows that while larger models like ResNet-18 maintain high accuracy regardless of the training sequence, smaller networks benefit more significantly from structured progression. This implies that benchmark design and training curricula should consider model capacity when evaluating innovation, as a "one-size-fits-all" benchmark may not accurately reflect advances across different model classes.

6.4 Training Efficiency and Stability

Although curriculum learning increased total training time due to multiple training stages, the incremental stages resulted in more stable and predictable optimization. Loss curves exhibited fewer oscillations, and convergence occurred more smoothly than in direct training scenarios. This supports the interpretation that progressive learning acts as a form of optimization preconditioning—helping the model reach better minima without aggressive hyperparameter tuning.

6.5 Limitations and Future Considerations

Despite promising results, this study has several limitations. The experiments were restricted to three datasets and two architectures. Expanding this analysis to include larger datasets (e.g., Tiny-ImageNet, COCO) or transformer-based architectures could provide deeper insights. Additionally, curriculum sequencing was manually defined based on dataset difficulty; future research could employ automated complexity metrics to dynamically adjust training order.

Another limitation is the computational overhead introduced by curriculum training. While manageable for smaller models, this overhead may become significant for large-scale architectures. Exploring adaptive curricula or hybrid fine-tuning approaches could balance performance gains and efficiency.

6.6 Summary of Findings

In summary:

- Curriculum learning improved convergence stability and slightly enhanced generalization across all models.
- LeNet-5 showed greater relative benefit than ResNet-18, emphasizing the role of progressive learning for shallow networks.
- Residual architectures like ResNet-18 remain less dependent on curriculum order but still gain stability benefits.
- Progressive dataset exposure offers a potential method for countering benchmark saturation by maintaining learning difficulty.

Overall, these findings validate that the combination of curriculum learning and structured dataset design can provide a more meaningful measure of model innovation, particularly in the context of saturated benchmarks.

6.7 Benchmark Saturation Metric

To quantify the degree of *benchmark saturation*—that is, how much a dataset still discriminates between different model architectures—we introduce a simple metric called the **Saturation Index (S)**.

Let A_i denote the final test accuracy achieved by the i^{th} model on a given dataset. If we evaluate n models (e.g., LeNet-5, SimpleCNN, ResNet-18), we can compute:

$$\mu_{acc} = \frac{1}{n} \sum_{i=1}^{n} A_i$$

$$\sigma_{acc} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (A_i - \mu_{acc})^2}$$

We then define the **Benchmark Saturation Index (S)** as:

$$S = 1 - \frac{\sigma_{acc}}{\mu_{acc}}$$

where:

- μ_{acc} is the mean accuracy across models,
- σ_{acc} is the standard deviation of accuracies.

This index takes values between 0 and 1:

- $S \to 1$ indicates **high saturation**, where all models perform similarly well and the benchmark no longer differentiates model performance.
- $S \to 0$ indicates **low saturation**, where models vary significantly in accuracy, suggesting the dataset remains a good discriminative testbed.

Table 8. Benchmark Saturation Index (S) across datasets.

Dataset	LeNet-5	SimpleCNN	ResNet-18	S
MNIST	98.8	99.0	99.3	0.998
FashionMNIST	88.1	89.0	91.3	0.985
CIFAR10	71.0	72.5	77.8	0.960

The proposed Benchmark Saturation Index (S), calculated from model performance variance, shows that MNIST exhibits near-total saturation (S = 0.997), while CIFAR-10 remains the most discriminative (S = 0.933). This quantifies the diminishing utility of older benchmarks for driving innovation

7 Conclusion and Future Work

This research investigated the relationship between dataset complexity, curriculum-based training, and model generalization through comparative experiments using LeNet-5 and ResNet-18. The study demonstrated that structured exposure to datasets of increasing complexity provides measurable improvements in convergence stability and feature transfer, even when overall accuracy gains are modest.

The results show that **curriculum learning** acts as an effective regularization technique, particularly beneficial for shallower networks such as LeNet-5. By progressively training across datasets from MNIST to Fashion-MNIST to CIFAR-10, the model was able to form a hierarchical understanding of visual complexity, leading to smoother optimization and reduced overfitting. Although ResNet-18, due to its residual connections and greater depth, was less dependent on this progression, it still exhibited enhanced stability and lower validation loss under curriculum-based training.

These findings support the broader hypothesis that benchmark-driven evaluation alone is insufficient to measure true model innovation, especially when benchmarks become saturated. Integrating curriculum principles into dataset design and evaluation protocols may offer a more dynamic and meaningful way to assess the adaptability and robustness of emerging architectures.

7.1 Future Work

Future research should expand this study in several directions:

- **Broader Dataset Spectrum:** Extending experiments to larger and more diverse datasets such as Tiny-ImageNet, COCO, or OpenImages could validate scalability across real-world scenarios.
- **Automated Curriculum Design:** Developing algorithms that automatically determine dataset order based on complexity metrics could make curriculum learning more adaptive and data-driven.
- **Architectural Diversity:** Applying this framework to newer architectures, including vision transformers or hybrid CNN-Transformer models, may reveal different patterns of curriculum effectiveness.
- **Long-Term Learning Dynamics:** Investigating the retention and transfer of learned features across multiple domains could provide insight into continual learning and catastrophic forgetting.

Ultimately, this work highlights that combining curriculum learning with well-structured datasets can serve as a viable path toward overcoming benchmark saturation and fostering genuine architectural innovation in deep learning research.

References

[1] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[3] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Proc. 26th Int. Conf. on Machine Learning (ICML)*, 2009, pp. 41–48.

[4] A. C. Lorena, L. P. Garcia, J. Lehmann, M. C. Souto, and T. K. Ho, "How complex is your classification problem? A survey on measuring classification complexity," *ACM Computing Surveys*, vol. 52, no. 5, pp. 1–34, 2019.

[5] J. Miller, M. Taori, D. R. Raghunathan, and A. Schmidt, "Accuracy on the line: On the strong correlation between out-of-distribution and in-distribution generalization," in *Proc. 38th Int. Conf. on Machine Learning (ICML)*, 2021, pp. 7721–7735.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2012, pp. 1097–1105.

[7] A. Coates, A. Ng, and H. Lee, "An analysis of single-layer networks in unsupervised feature learning," in *Proc. 14th Int. Conf. on Artificial Intelligence and Statistics (AISTATS)*, 2011, pp. 215–223.

[8] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 248–255.

[9] Q. Xie, Z. Dai, E. Hovy, T. Luong, and Q. Le, "Self-training with noisy student improves ImageNet classification," in *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10687–10698.

[10] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms," [Online]. Available: http://fashion-mnist.s3-website.eu-central-1.amazonaws.com/. Accessed: Oct. 25, 2025.