Reimplementation of Robust Classification via Regression for Learning with Noisy Labels

DD2412

Oskar Wallberg, Johannes Rosing
KTH Royal Institute of Technology
Industrial Engineering and Management, Machine Learning

December 20, 2024

Contents

1 Abstract

In this work, we attempt to reimplement and partially verify the findings of the paper "Robust Classification via Regression for Learning with Noisy Labels" by Englesson and Azizpour (ICLR 2024). The original paper combines compositional data analysis with a shifted Gaussian noise (SGN) model to simultaneously achieve loss reweighting and label correction. Due to limited computational resources, we focus on a simplified version of their experiments, training a standard cross-entropy (CE) baseline, a Generalized Cross-Entropy (GCE) model, an Early Learning Regularization (ELR) model, along with their main Shifted Gaussian Noise (SGN) approach for 100 epochs instead of the original 300–600 epochs. We investigate performance under symmetric and asymmetric label noise at various noise levels (0%, 20%, 40%) on the CIFAR-10 dataset. Our preliminary findings show that SGN generally provides better robustness than CE and ELR for all noisy scenarios (noise $> 0\%$), and better robustness than GCE for asymmetric noise scenarios. These results align qualitatively with the original paper, despite differences in training durations and computational resources. Our results demonstrate that SGN effectively balances Loss Reweighting (LR) and Label Correction (LC) to achieve stable and consistent robustness, particularly under challenging asymmetric noise conditions. While GCE outperformed SGN in symmetric scenarios, SGN uniquely avoided overfitting and preserved clean representations during later training stages. This reimplementation validates SGN as a promising approach for noise-robust learning, combining stability and resilience where traditional methods often falter.

2 Introduction

The significant progress in machine learning over the last decade has largely been fueled by supervised deep learning on expansive labeled datasets, such as ImageNet. However, as the reliance on annotated data grows, challenges like label noise, ambiguity, and the high costs of manual labeling have become increasingly prominent in industrial and real-world applications. These issues have motivated a shift toward developing robust learning methodologies that can address the imperfections in data annotation. By overcoming the limitations of traditional supervised approaches, robust algorithms aim to maintain high performance and generalization even in the presence of noisy or unreliable labels.

Training deep neural networks under label noise presents a persistent challenge, as these models are prone to overfitting to incorrect labels, leading to significant degradation in generalization. The pioneering work of Englesson and Azizpour (2024) introduces a novel approach that reformulates classification as a regression problem. Their method leverages a shifted Gaussian noise (SGN) model in a log-ratio transformed latent space, seamlessly integrating loss reweighting and label correction to enhance model robustness. This unified framework offers a promising solution to the problem of noisy labels, effectively addressing two key challenges in a principled and synergistic manner.

In this report, we focus on reimplementing a simplified version of their methodology. Given resource constraints, we limit our experiments to 100 training epochs and avoid extensive hyperparameter tuning. We compare the performance of the SGN approach to a standard cross-entropy (CE) baseline, a Generalized Cross-Entropy (GCE) model, and an Early Learning Regularization (ELR) model under conditions of both symmetric and asymmetric label noise. Our objective is to evaluate whether the robustness improvements demonstrated by SGN hold in a more resource-constrained setup, thereby assessing its practicality for broader applications.

Further, we changed the asymmetric noise calculation to make it even harder for the models to learn under asymmetric noise (The implementation can be found in section 4.2.2). This is an extended experiment to see if SGN can stay robust under harsher conditions, where it is harder to learn to "ignore" the noisy labels.

3 Background

The presence of noisy labels in large datasets is a well-known issue. Conventional training approaches like cross-entropy loss can lead to significant overfitting on incorrect labels. Two main strategies have emerged to combat this:

- **Loss Reweighting:** Assign lower weights to potentially noisy examples, slowing down the overfitting process.
- **Label Correction:** Attempt to estimate the "true" underlying label distribution, effectively correcting noise during training.

Englesson and Azizpour's approach leverages concepts from *compositional data analysis*, using a log-ratio transform (like the isometric log-ratio, ilr) to map categorical distributions into an unconstrained real space. By assuming a shifted Gaussian noise model in this transformed space, they achieve both loss reweighting (via predicted covariance) and label correction (via predicted shifts).

3.1 How SGN Works

The SGN method relies on transforming classification labels into a regression formulation using compositional data analysis techniques. This is achieved through the Isometric Log-Ratio (ilr) Transform and the Gaussian noise model.

Step 1: Transforming Classification Labels into Compositional Data

Classification labels, typically represented as a one-hot vector, are transformed into the probability simplex using label smoothing. This ensures all components are strictly positive (required for log-ratio transformations).

$$LS(y) = (1 - \gamma)\delta_y + \gamma u$$

Where

- δ_y is the one-hot vector representation of the label.
- $u = \frac{1}{K}$ (a uniform distribution over K classes).
- γ is a smoothing parameter (e.g. $\gamma = 0.1$).

Step 2: Mapping Compositional Data into Regression Space

The Isometric Log-Ratio (ilr) Transform maps the label-smoothed vector from the constrained probability simplex Δ^{K-1} to an unconstrained $\mathbb{R}^{K-1}$ space.

ilr Transform:

$$irl(LS(y)) = V^\top log(\frac{LS(y)}{g(LS(y))})$$

Where

- $g(LS(y)) = (\prod_{k=1}^{K} LS(y)_k)^{\frac{1}{K}}$ is the geometric mean of the label-smoothed components.
- V is the Helmert matrix (orthonormal basis).
- γ is a smoothing parameter (e.g. $\gamma = 0.1$).

Here, V is a $K \times (K-1)$ matrix ensuring that the transformed data lies in an unconstrained space.

Step 3: Regression with a Shifted Gaussian Noise Model

Once in regression space, the target $t = ilr(LS(y))$ is modeled with a shifted Gaussian noise distribution:

$$t(x) = \mu(x) + \epsilon(x) \ , \qquad \epsilon(x) \sim \mathcal{N}(\Delta(x), \Sigma(x))$$

Where

- $\mu(x)$ is the predicted mean of the Gaussian distribution.
- $\Sigma(x)$ is the predicted covariance matrix.
- $\Delta(x)$ is the predicted shift parameter for label correction.

The corresponding loss is given by the Negative Log-Likelihood (NLL):

$$\mathcal{L} = \frac{1}{2} \sum_{i=1}^{N} (\|S_\theta^{-1}(t_i - \mu_\theta(x_i))\|^2 + \log |S_\theta|)$$

where $S_\theta(x)$ is a parametrization of $\Sigma(x)$ ensuring efficient computation.

Step 4: Mapping Back to Classification Space

The regression predictions $\mu(x)$ in $\mathbb{R}^{K-1}$ are transformed back to the probability simplex Δ^{K-1} using the inverse ilr transform:

$$\pi = ilr^{-1}(\mu) = g(\mu)exp(V\mu)$$

Where:

- $g(\mu)$ rescales the components to sum to 1.

The final class prediction is:

$$y = \arg\max_k \pi_k$$

In summary the SGN process consists of (1) transforming labels to One-hot labels that are smoothed and mapped into regression space using the ilr transform. Then performing (2) gaussian regression to be able to incorporate label correction (Δ) and loss reweighting (Σ). To then (3) finally inverse map predicted regression means back to classification space using the inverse ilr transform. (Englesson and Azizpour 2024)

3.2 Model Implementations

The primary goal of comparing different models in this study is to evaluate the robustness of the SGN approach against baseline and alternative methods for handling noisy labels. Baseline models provide a point of reference for benchmarking, while alternative robust methods help contextualize the performance of SGN and its specific advantages.

Due to computational and time constraints, not all methods from the original paper could be reimplemented (see more under section *Limitations*). We focus on a selected subset of representative models, each offering unique strategies for combating label noise. These models were chosen based on their relevance to the original report, their prominence in prior research, and their complementary strategies to SGN. Below, we describe each method, including its strengths, weaknesses, and relation to SGN. A summary table is also provided for quick reference.

Cross-Entropy (CE)

Cross-Entropy (CE) is the most widely used loss function for classification tasks. It measures the difference between the true label distribution (typically one-hot encoded) and the predicted probability distribution. CE encourages the model to assign high probability to the correct class while penalizing incorrect predictions. It is simple and effective for clean datasets but struggles in the presence of noisy labels, often overfitting to incorrect labels. (Mao, Mohri, and Zhong 2023)

Relation to SGN: SGN extends beyond CE by introducing mechanisms like loss reweighting and label correction, which address noise-induced overfitting.

Generalized Cross-Entropy (GCE)

Generalized Cross-Entropy (GCE) extends the CE loss by introducing a robustness parameter (q) that controls the contribution of each sample to the loss. When $q = 1$, it reduces to CE, but for $q < 1$, it reduces the impact of hard-to-fit noisy labels. GCE balances CE and Mean Absolute Error (MAE), offering robustness to label noise. However, tuning the q parameter requires dataset-specific considerations. (Zhang and Sabuncu 2018)

Relation to SGN: Both GCE and SGN mitigate the influence of noisy labels, but SGN achieves this via probabilistic modeling rather than a parameterized trade-off.

Early Learning Regularization (ELR)

Early Learning Regularization (ELR) mitigates overfitting to noisy labels by penalizing predictions that deviate from the model's early, clean predictions. ELR assumes that the initial learning phase captures more reliable patterns and is effective at preventing overfitting during later stages. However, it requires monitoring and adjustment of the regularization term over time. (Liu et al. 2020)

Relation to SGN: SGN complements ELR's reliance on early learning by dynamically correcting noisy labels throughout training.

Shifted Gaussian Noise (SGN)

Shifted Gaussian Noise (SGN) introduces robustness by modeling noisy labels with a shifted Gaussian noise model in a transformed regression space. The shift parameter (Δ) allows for label correction, while the covariance matrix (Σ) facilitates loss reweighting. Together, these mechanisms address label noise in a unified, principled framework.

Key Mechanisms:

- **Loss Reweighting (LR):** Downweights noisy samples based on predicted covariance (Σ).
- **Label Correction (LC):** Estimates true labels via a shifted Gaussian distribution.

(Englesson and Azizpour 2024)

Summary Table of Strengths, Weaknesses, and Relation to SGN

Table 1: Comparison of Models: Strengths, Weaknesses, and Relation to SGN

Model	**Strengths**	**Weaknesses**	**Relation to SGN**
CE	Simple and efficient; optimal for clean datasets.	Prone to overfitting noisy labels; lacks noise robustness mechanisms.	SGN addresses CE's shortcomings with loss reweighting and label correction.
GCE	Flexible tuning of robustness through q; combines properties of CE and MAE.	Requires dataset-specific hyperparameter tuning.	SGN achieves robustness via probabilistic modeling rather than parameter tuning.
ELR	Effective against overfitting noisy labels; exploits temporal learning dynamics.	Requires careful monitoring of early training; sensitive to regularization parameters.	SGN complements ELR by dynamically correcting noisy labels throughout training.
SGN	Unified framework for loss reweighting and label correction; flexible noise modeling.	Requires careful tuning of noise parameters (Δ, Σ); computationally intensive.	–

3.3 Scope and Limitations

This study reimplements the SGN method proposed by Englesson and Azizpour (2024) with constraints on computational resources, training duration, and experimental scope. Unlike the original report, which trained models for 300–600 epochs, our implementation was limited to 100 epochs per run. This reduced training duration may have restricted the ability of SGN and other models to fully adapt to noisy labels, potentially underestimating their optimal performance. Additionally, while the original work likely involved extensive hyperparameter tuning, this study used pre-established, fixed settings for all models to ensure simplicity and fairness, limiting the optimization of SGN's noise-specific parameters.

The scope of our experiments was also narrower, focusing exclusively on CIFAR-10, a relatively simple dataset, whereas the original study evaluated SGN on more diverse datasets such as CIFAR-100 and possibly larger-scale benchmarks. The asymmetric noise implementation in this study introduced a more challenging label corruption process compared to the original report, which may have further stressed the models in ways not originally explored. Despite these limitations, the reimplementation retained the core elements of SGN, including the ilr transform and shifted Gaussian noise modeling, yielding results that qualitatively align with the original findings, albeit with some deviations. These constraints highlight the need for extended experiments to fully validate SGN's robustness across broader conditions.

4 Methodology

4.1 Dataset

We did our experiments on the CIFAR-10 dataset. The CIFAR-10 dataset contains a total of 60,000 images divided into two sets:

- **Training set**: 50 000 images.
- **Test set**: 10 000 images.

Each image is a 32x32 RGB image, and the dataset is evenly distributed across 10 classes, meaning each class has 5000 training examples per class, and 1000 test examples per class.

The classes in the dataset and their corresponding labels are as follows:

Class Index	Class Name
0	Airplane
1	Automobile
2	Bird
3	Cat
4	Deer
5	Dog
6	Frog
7	Horse
8	Ship
9	Truck

Table 2: CIFAR-10 Class Index and Corresponding Class Names

Every epoch, we utilize all the training data and evaluate on the test set.

4.2 Noise Implementation

4.2.1 Symmetric noise

For symmetric noise, the noise process applies a uniform random corruption to the dataset labels. Specifically, each label is replaced with any other label in the dataset with equal probability, ensuring that all classes are equally likely to be selected. Mathematically, this can be expressed as:

$$P(y' = c \mid y = c') = \begin{cases} 1 - \eta & \text{if } c = c' \\ \frac{\eta}{K-1} & \text{if } c \neq c', \end{cases}$$

where y is the original label, y' is the corrupted label, K is the number of classes, and η is the noise level.

4.2.2 Asymmetric Noise

For asymmetric noise, the noise process is defined by a predefined probability transition matrix $\mathbf{P}$, where each entry P_{ij} represents the probability of the true label i being flipped to label j. This matrix introduces structured corruption that mimics domain-specific or real-world label noise patterns. The probability transition matricies for our noise rates at 20% ($\mathbf{P_{20\%}}$) and 40% ($\mathbf{P_{40\%}}$) respectively can be seen under section **Appendix**.

These matricies capture structured noise, where some labels are more likely to be confused with specific others, which is usually more representative of real-world label ambiguity, where label classes are more likely to be confused with some than others.

Asymmetric noise is introduced by randomly flipping a subset of labels according to a predefined severity. The process can be described mathematically as follows:

Transitional Probabilities: The following mapping describe the plausible confusions between label classes (i.e. likely mislabeling in a real world dataset) as we interpreted it in this study :

$$\begin{aligned}
&\textbf{Airplane } (0) \rightarrow \textbf{Ship } (8) \\
&\textbf{Automobile } (1) \rightarrow \textbf{Truck } (9) \\
&\textbf{Bird } (2) \rightarrow \textbf{Cat } (3),\ \textbf{Bird } (2) \rightarrow \textbf{Dog } (5) \\
&\textbf{Cat } (3) \rightarrow \textbf{Bird } (2),\ \textbf{Cat } (3) \rightarrow \textbf{Dog } (5) \\
&\textbf{Deer } (4) \rightarrow \textbf{Horse } (7) \\
&\textbf{Dog } (5) \rightarrow \textbf{Bird } (2),\ \textbf{Dog } (5) \rightarrow \textbf{Cat } (3) \\
&\textbf{Frog } (6) \rightarrow \textbf{Bird } (2) \\
&\textbf{Horse } (7) \rightarrow \textbf{Deer } (4) \\
&\textbf{Ship } (8) \rightarrow \textbf{Airplane } (0) \\
&\textbf{Truck } (9) \rightarrow \textbf{Automobile } (1)
\end{aligned} \tag{1}$$

Unlike prior approaches that often employ static and simplistic mappings for asymmetric noise, our method employs a tailored noise matrix to reflect realistic and complex noise distributions. By applying the noise through a probability matrix where all probabilities are bi-directional and not only mapped to one other class, we create scenarios where the model cannot easily learn patterns to "ignore" the noisy labels, as the noise could affect more than one class transition.

The asymmetric noise implementation in Englesson and Azizpour (2024). For the CIFAR-10 dataset, the label-swapping rules are defined as:

$$\begin{aligned} &\textbf{Truck } (9) \rightarrow \textbf{Automobile } (1) \\ &\textbf{Bird } (2) \rightarrow \textbf{Airplane } (0) \\ &\textbf{Cat } (3) \rightarrow \textbf{Dog } (5) \\ &\textbf{Dog } (5) \rightarrow \textbf{Cat } (3) \\ &\textbf{Deer } (4) \rightarrow \textbf{Horse } (7). \end{aligned} \tag{2}$$

The probability of flipping a label is determined by the severity parameter, which directly controls the fraction of labels affected. For example, a severity of 0.2 means 20% of labels are flipped according to the mapping rules.

4.3 Adaptation of the Original Approach

The original method proposed transforming labels using label smoothing and then applying an ilr transform to obtain a regression-style target. A neural network then predicts both a mean and a covariance matrix (and implicitly a shift) that model the data under Gaussian assumptions. The loss derived from this probabilistic view inherently downweights noisy examples and can gradually correct their labels.

In our reimplementation, we follow the same high-level steps:

1. Apply label smoothing to ensure all label components are strictly positive.
2. Transform labels to the unconstrained real space via the ilr transform.
3. Train a two-headed network predicting both mean and covariance, and incorporate a shift as proposed by SGN.
4. Convert predictions back to probabilities for evaluation.

4.4 Modifications

The main difference from the original study lies in the reduced number of training epochs (100 instead of 300–600) and the scale of hyperparameter tuning, which was limited due to computational constraints. While the original study explores a broader range of models and settings, our implementation focuses on four key methods: Cross-Entropy (CE), Generalized Cross-Entropy (GCE), Early Learning Regularization (ELR), and Shifted Gaussian Noise (SGN).

These models were selected to provide a comprehensive comparison between baseline methods (CE), robust loss functions (GCE and ELR), and the target approach (SGN). The choice of models aligns with those used in the original study, ensuring relevance and interpretability. However, we simplify certain aspects, such as reducing the number of epochs and omitting exhaustive grid searches for hyperparameter optimization. Instead, we rely on fixed hyperparameter configurations (detailed in Table 6) that are consistent across experiments for fairness.

Our primary objective is not to achieve state-of-the-art (SOTA) results but to evaluate whether the SGN approach demonstrates consistent robustness over CE, GCE, and ELR under noisy label conditions, particularly when computational resources are constrained. The core elements of the original SGN method—including the ilr transform, shifted Gaussian modeling, and inverse transform were however faithfully reproduced.

4.5 Implementation Details

The original codebase for the SGN method was implemented in TensorFlow, which we translated into PyTorch for our experiments. The exact model architecture used is a WideResNet-28-2, same as in the original paper by Englesson and Azizpour (2024). This smaller version of the otherwise common WideResNet-50-2 architecture, but achieves a balance between computational efficiency and performance.

For data preprocessing, we applied standard image augmentations, including random cropping and horizontal flipping, consistent with common practices for CIFAR-10. The input images were normalized to have zero mean and unit variance based on the dataset's channel statistics.

The training was conducted with a fixed batch size and learning rate, as detailed in the hyperparameter table (Table 6). We did not engage in extensive hyperparameter tuning due to computational constraints. Instead, we opted for parameter settings commonly used in noise-robust learning tasks and established benchmarks for CIFAR-10, such as those described in Zagoruyko and Komodakis (2016), He et al. (2016), Zhang and Sabuncu (2018), Liu et al. (2020), and Englesson and Azizpour (2024). For example, Cross-Entropy (CE) uses standard settings, including learning rates around 0.1 and batch sizes of 128, as established in Zagoruyko and Komodakis (2016) and He et al. (2016). Generalized Cross-Entropy (GCE) with its robustness parameter (q) between 0.7 and 0.9 is widely adopted in Zhang and Sabuncu (2018). Early Learning Regularization (ELR) employs similar settings with a regularization weight ($\lambda_{\text{ELR}} = 3.0$) and EMA momentum ($\beta = 0.7$), as shown in Liu et al. (2020). For SGN, the hyperparameters ($\alpha = 0.995$ for label smoothing and EMA decay = 0.99) were based on findings in Englesson and Azizpour (2024). These choices ensure a fair comparison and align with established practices in noise-robust learning.

Additionally, the code was structured to allow reproducibility and scalability. Each implemented method (CE, GCE, ELR, and SGN) was modularized, ensuring that their respective loss functions and training loops could be independently evaluated. We also ensured that the training pipeline, including the optimizer and learning rate schedules, remained consistent across methods for fair comparisons.

All experiments were conducted using Google Colab Pro with a single NVIDIA T4 GPU. The training process for 100 epochs took approximately 66 minutes on average. For the main part of the reimplementation, we implemented four models (CE, GCE, ELR, and SGN) and evaluated them across five noise variants: 0%, 20% symmetric noise, 40% symmetric noise, 20% asymmetric noise, and 40% asymmetric noise. To ensure statistical robustness, each

noise variant was trained and evaluated five times per model, resulting in a total of $4 \times 5 \times 5 = 100$ runs for the main experiments.

Additionally, we conducted an ablation study exclusively for SGN to analyze the impact of its two core components: Loss Reweighting (LR) and Label Correction (LC). For this, we disabled LR and LC separately and tested under 40% symmetric and and asymmetric noise conditions, resulting in four ablation configurations. Each configuration was run five times, totaling $4 \times 5 = 20$ additional runs for the ablation study.

In total, we performed 120 full training and evaluation runs.

5 Extensions

This section highlights key extensions to the original work, showcasing how our approach broadens the scope of the original paper and introduces new perspectives for analyzing noise-robust learning techniques.

5.1 Performance Analysis Over the Training Process

The main extension consists of weighing in the performance trends over the entire training process (100 epochs) in the comparison of models. This differs from the original report which evaluates model performance at the final training epoch (presumably when the model has fully learned the training data). This longitudinal approach captures the temporal dynamics of learning under noisy conditions, revealing insights into:

- Early adaptation to general patterns in noisy data.
- Potential overfitting to noisy labels as training progresses.
- Differences in stability and volatility of performance across epochs.

Extension Potential: This temporal perspective enables more comprehensive evaluation of their strengths and weaknesses. It raises concern for optimal training strategies, such as early stopping or dynamic regularization, and offers insights into the bias-variance trade-off. Longitudinal analysis is particularly critical for evaluating noise-robust techniques, as their performance is often tied to how effectively they manage noise across different training phases.

5.2 Complex Asymmetric Noise Label Mapping

Another notable extension in our work is the introduction of a more complex asymmetric noise label mapping. Unlike simpler mappings used in the original study, we defined label flipping patterns informed by semantic or visual similarities between classes, creating a more complex and challenging noise structure to test the limits of SGN and other noise robust methods.

Extension Potential: This extension places SGN and other models under additional pressure to maintain stability and robustness, as the noise structure is not only complex but systematically biases certain classes. Analyzing model behavior under these conditions provides a deeper understanding of how well different techniques generalize when confronted with real-world label ambiguity.

5.3 Potential for Future Work

The extensions outlined here provide a strong foundation for further exploration and innovation in noise-robust learning. Promising directions for future research include:

- **Scalability to Complex Datasets:** Extending evaluations to larger, more diverse datasets like CIFAR-100 or ImageNet to test robustness in highly complex settings.
- **Dynamic Noise Modeling:** Investigating models that can adapt to evolving noise patterns or learn dynamic label corrections during training.
- **Incorporating Semi-Supervised Learning:** Exploring the integration of noisy labeled data with unlabeled data to enhance generalization and reduce reliance on clean labels.
- **Adopting Robustness Metrics:** Employing established metrics like the *Area Under the Noise Robustness Curve (AUNRC)* (Jiang et al. 2018), which evaluates model performance across a range of noise levels, and *Generalized Accuracy (GA)* (Patrini et al. 2017), which measures accuracy weighted by the clean and noisy label distributions. These metrics provide standardized ways to assess model stability and robustness under varying noise conditions.

Summary: The extensions proposed here emphasize the importance of longitudinal analysis in evaluating model performance. Observing models over time, rather than at a single endpoint, is key for comparing their overall behavior and robustness. This temporal analysis aligns closely with the proposed directions for future work, as it provides a richer framework for understanding the nuanced trade-offs involved in noise-robust learning.

6 Results

6.1 Main Findings

The result reported in the table below includes the **mean test accuracies and standard deviations** across all models and noise rates. Each experiment is run five times to establish these values. The hyperparameters are fixed across all models and runs, and each run consists of 100 epochs of training. The highest achieved mean per noise rate is highlighted in bold.

Table 3: Method Performance Comparison under Different Noise Rates

Method	**No Noise**	**Symmetric Noise Rate**		**Asymmetric Noise Rate**	
	0%	20%	40%	20%	40%
CE	93.45 ± 0.57	84.57 ± 0.77	74.92 ± 0.92	82.93 ± 0.92	61.33 ± 2.0
GCE	92.23 ± 0.45	**90.35** ± 0.63	**87.34** ± 0.75	**88.84** ± 0.72	69.07 ± 1.5
ELR	**93.51** ± 0.42	88.49 ± 0.60	81.35 ± 0.67	83.42 ± 0.65	64.12 ± 1.58
SGN (Main)	91.39 ± 0.25	90.02 ± 0.22	86.02 ± 0.37	88.01 ± 0.34	**77.58** ± 0.9

In the following, we report the **relative decrease** (%) in mean test accuracy compared to 0% noise for all models. The lowest decrease achieved per noise rate is highlighted in bold.

Table 4: Relative decrease in test accuracy (%) compared to 0% noise

Method	**Symmetric Noise Rate**		**Asymmetric Noise Rate**	
	20%	40%	20%	40%
CE	-9.5%	-19.8%	-11.3%	-34.4%
GCE	-2.0%	**-5.3%**	**-3.7%**	-25.1%
ELR	-5.4%	-13.0%	-10.8%	-31.4%
SGN (Main)	**-1.5%**	-5.9%	**-3.7%**	**-15%**

Further, we report the results of the ablation study, where Loss Reweighting (LR) and Label Correction (LC) were systematically included and excluded from the main SGN method. An ✗ marks the exclusion of a specific technique, while ✓ marks the inclusion. The noise rates for symmetric and asymmetric noise were in both cases 40%.

Table 5: Ablation Study on CIFAR-10, SGN 40% Noise

LR	**LC**	**CIFAR-10**	
		Symmetric	Asymmetric
✗	✓	83.03 ± 0.51	68.02 ± 1.18
✓	✗	84.72 ± 0.62	68.26 ± 1.2
✓	✓	**86.02** ± 0.37	**77.58** ± 0.9

6.2 Comparison Between SGN and CE

First, we compare the mean test accuracy per epoch for SGN and CE for **0% noise**. As expected, both models perform well, but CE comes out on top with a slight margin.

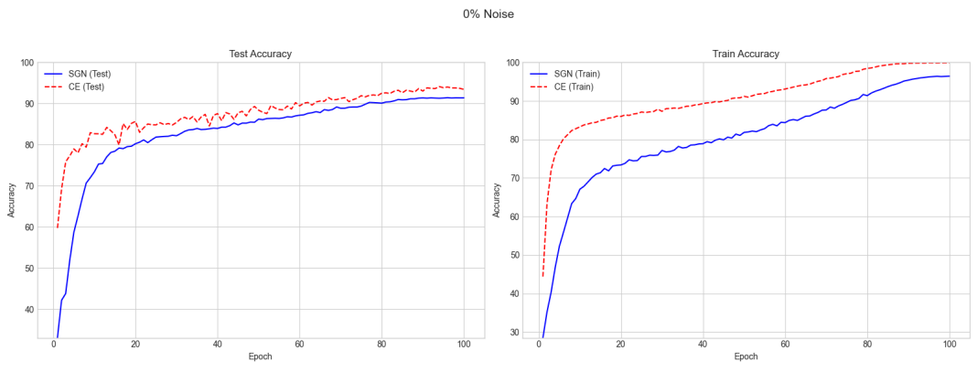


Figure 1: Mean test accuracy by epoch, 0% noise

6.2.1 Symmetric noise

Next, we compare SGN with CE for symmetric noise rates **20 and 40%**. In both cases, CE overfits to the noisy data during end of training while SGN remains robust.

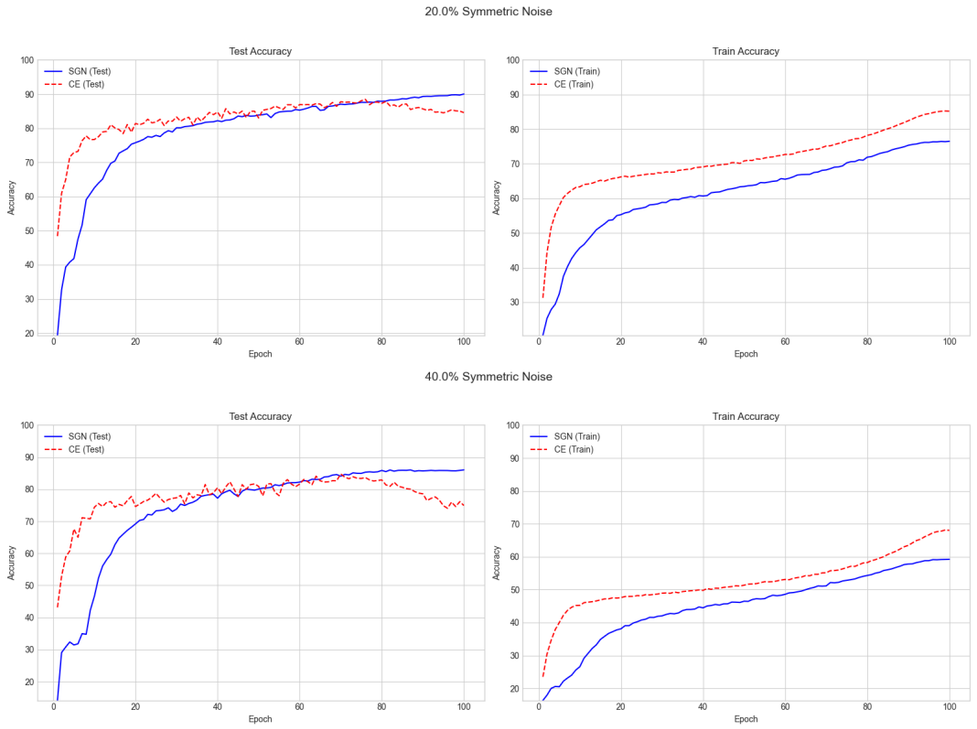


Figure 2: Mean test accuracy by epoch under symmetric noise. Top: 20% noise. Bottom: 40% noise.

6.2.2 Asymmetric noise

Lastly, we compare SGN with CE for asymmetric noise rates **20 and 40%**. Similarly, CE deteriorates during the end of training. SGN, on the other hand, does not, although it is not as robust at 40% asymmetric noise.

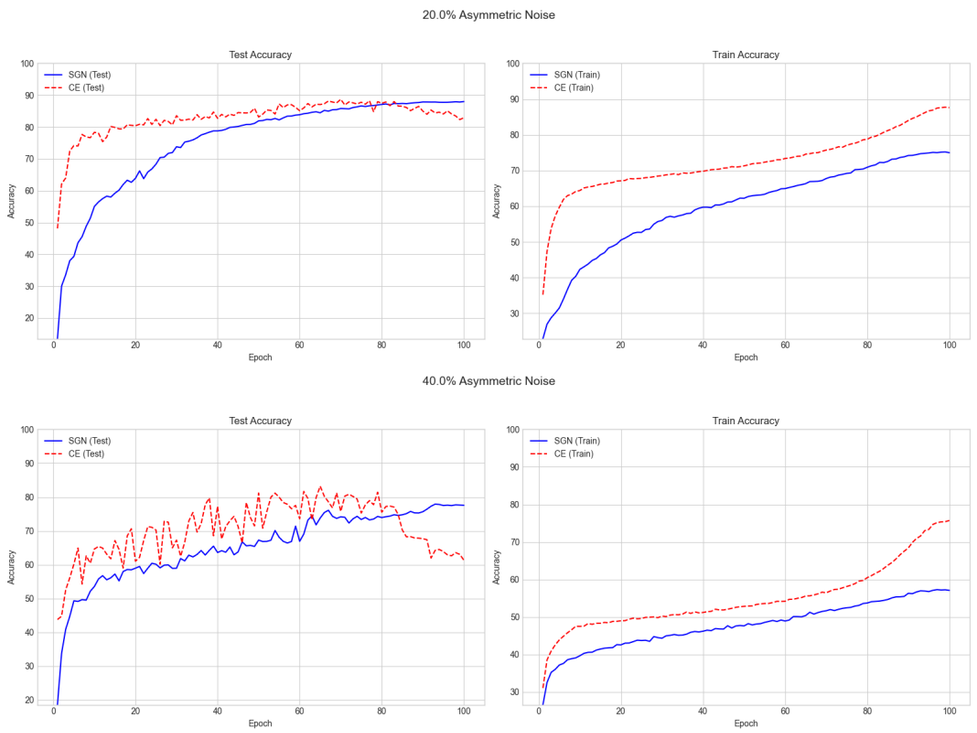
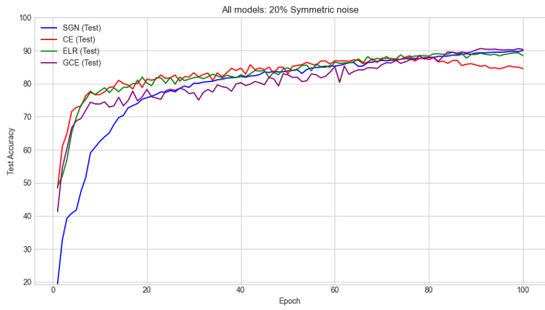


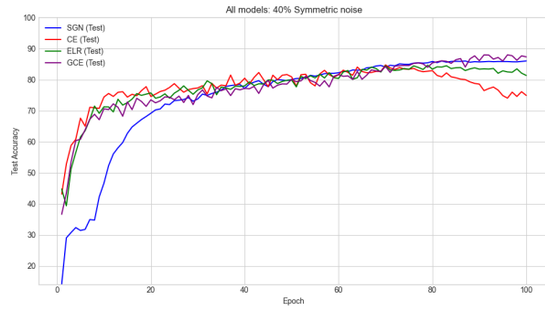
Figure 3: Mean test accuracy by epoch under asymmetric noise. Top: 20% noise. Bottom: 40% noise.

6.3 Comparison between all Models

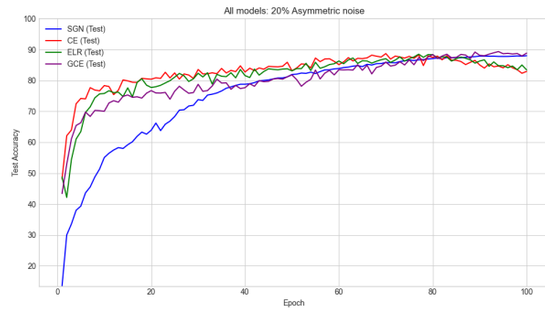
In the following, we present the mean test accuracy per epoch for all models implemented **(SGN, CE, ELR and GCE)**. Interestingly, although GCE slightly outperforms SGN in most cases, SGN is the only model that does not overfit during the end of training for 40% asymmetric noise.



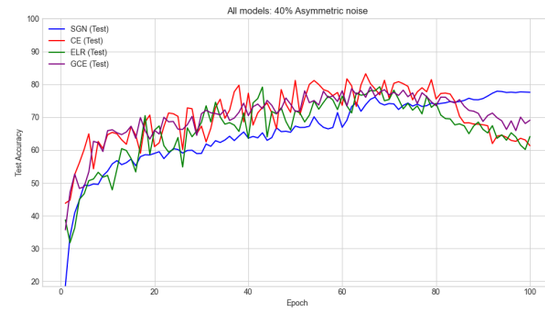
(a) Symmetric 20% Noise



(b) Symmetric 40% Noise



(c) Asymmetric 20% Noise

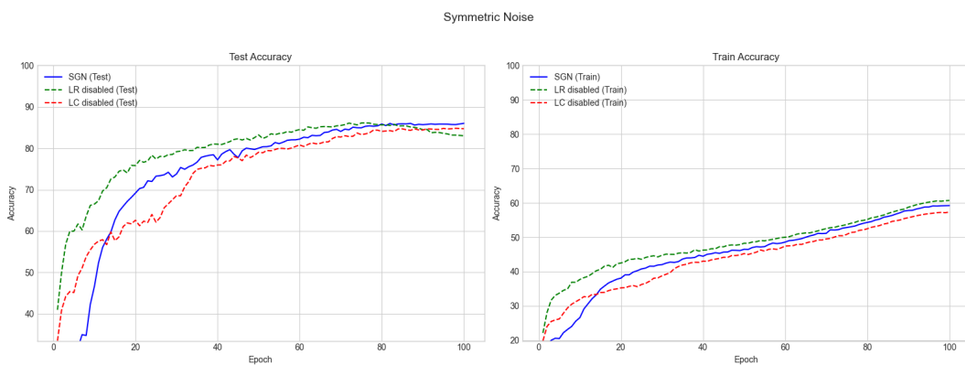


(d) Asymmetric 40% Noise

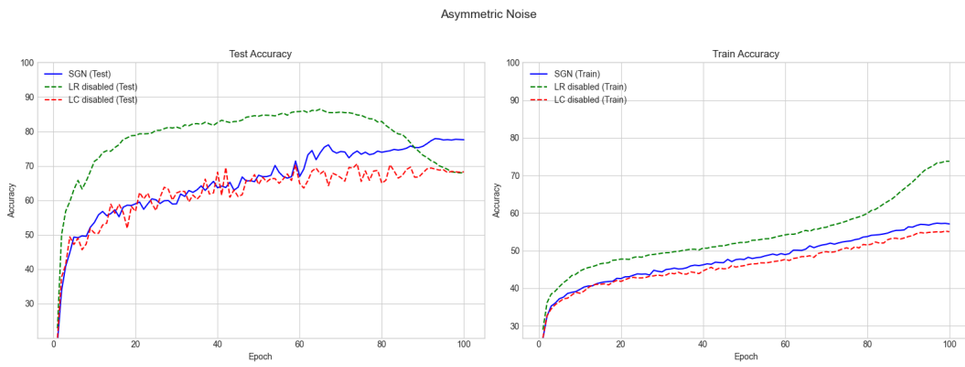
Figure 4: Mean test accuracy by epoch for all models.

6.4 Ablation Study

As evident in the results, SGN (including LR and LC) achieves a higher final test accuracy for both 40% symmetric and asymmetric noise. We note that for asymmetric noise, disabling LR creates a higher maximum test accuracy during the earlier parts of training. However, this seems to lead to overfitting to noisy labels, as the training accuracy increases dramatically at the latter stages of training, while the test accuracy decreases.



(a) Symmetric 40% Noise



(b) Asymmetric 40% Noise

Figure 5: Ablation Study, mean test accuracy per epoch.

7 Discussion

Our re-implementation suggests that even with fewer epochs and less tuning, the core property of SGN—improved robustness against label noise—persists.

7.1 Performance Comparison Between Models

Table 3 compares the performance of several methods under different noise conditions. Notably, in our setup, GCE achieves the highest accuracy in most noisy scenarios, outperforming other methods including SGN. This stands in contrast to the original SGN results reported by Englesson and Azizpour (2024), where SGN surpassed GCE for all noise levels. A likely reason for this discrepancy is that the original SGN experiments involved longer training times (300–600 epochs) and potentially extensive hyperparameter tuning for SGN.

It is interesting to note that, despite the limited training duration of 100 epochs, our GCE results are even stronger than those reported in the original SGN reference experiments. This suggests that GCE is quite robust and effective under our chosen conditions. Additionally, all models were trained under the same standardized configuration, including identical data augmentations, input normalization, and use of L2 weight decay in the SGD optimizer. Keeping the setup uniform ensures a fair comparison among the methods.

One of the clearest benefits of using SGN, as revealed in the model comparison section, is its robustness to overfitting at late training stages. While CE, and ELR often exhibited promising performance early in training—often even surpassing SGN in the initial epochs—they tended to degrade as training progressed under noisy conditions, especially at higher noise rates and for asymmetric noise. CE, in particular, displayed a decline in test accuracy over time as it overfit the noisy labels, while ELR, though more robust than CE, still showed variance and drops later in training.

GCE, however, displayed high resistance to overfitting in comparison to CE and ELR. As mentioned, it often outperformed even SGN on high noise levels, with the only exception being at 40% asymmetric noise where SGN was the only model which did not deteriorate.

SGN, by contrast, maintained more stable and robust performance throughout the full training duration, and had less variance between results. This suggests that SGN's internal label correction and loss reweighting mechanisms helped it avoid the pitfall of progressively fitting to noisy labels, thus preserving cleaner representations and more faithful label estimates even as epochs advanced. Additionally, SGN had the overall least relative decrease (as seen in Table 4) in performance for higher noise ratios.

7.1.1 Early Stopping As Means to Improve Performance

As seen in the model comparison section, there were some points during the training process where ELR and CE achieved a higher test accuracy than SGN at end of training, even though they deteriorated later. This observation raises the question of whether early stopping might benefit the other models. For instance, if one were to halt training for CE, or ELR at an earlier point—before overfitting sets in—these models might reach final accuracies above SGN's final results. With this in mind, the danger here is that early stopping requires careful calibration and may reduce reproducibility or robustness across different noise scenarios. Moreover, relying on early stopping to salvage performance means the model's intrinsic training dynamics are more fragile. In contrast, SGN's strength lies in its stable training curve, which reduces the need for such interventionist techniques.

7.2 Robustness of SGN Under Asymmetric Noise

One key aspect we investigated was whether the SGN approach retained its noise-robustness even under the more challenging asymmetric noise scenario that we implemented. The original SGN paper by Englesson and Azizpour (2024) demonstrated impressive resilience under both symmetric and standard asymmetric conditions. In our replication, we introduced an even harsher asymmetric noise setup, allowing labels to be ambiguously flipped among multiple classes rather than following a simple one-to-one mapping. Despite this more difficult noise structure, SGN maintained consistently stronger performance than the baseline CE and ELR when measured at the end of training.

Interestingly, even though GCE very slightly outperformed SGN at 40% symmetric noise, SGN was the clear winner at 40% asymmetric noise. It was the only model that maintained its relative robustness under such high asymmetric noise levels. For example, while GCE experienced a 25.1% relative decrease in performance at this noise level, SGN only experienced a 15% relative decrease as shown in Table 4.

7.3 Ablation Study

The ablation study provides deeper insights into the roles of Loss Reweighting (LR) and Label Correction (LC) within SGN and how their combined use contributes to its robustness under noisy conditions.

7.3.1 Symmetric Noise

Under symmetric noise, disabling LR led to faster initial performance gains, as test accuracy grew rapidly during the early epochs. However, this advantage was short-lived, as the model eventually overfit the noisy labels, causing test accuracy to plateau and slightly decline towards the end of training (around epoch 90). Disabling LC, on the other hand, resulted in slower growth in accuracy during the early stages, as the model struggled to correct noisy labels efficiently. Nevertheless, the LC-disabled model still outperformed the LR-disabled variant in the later epochs, suggesting that while LC accelerates learning, LR plays a critical role in maintaining long-term stability.

In contrast, the full SGN model, which combines LR and LC, demonstrated steady and consistent growth without signs of overfitting. This balance highlights the complementary nature of LR and LC: LR mitigates overfitting by downweighting noisy samples, while LC accelerates convergence by correcting labels early. Together, they enable SGN to achieve higher final accuracy with stable training dynamics.

7.3.2 Asymmetric Noise

The impact of disabling LR and LC was even more pronounced under asymmetric noise. When LR was disabled, the test accuracy skyrocketed during the early epochs, outperforming the other variants initially. However, as training progressed, the model began to overfit the dominant noisy label transitions, causing a sharp decline in test accuracy after epoch 70, even as training accuracy continued to rise. This result underscores LR's importance in controlling overfitting, particularly in complex noise scenarios where certain label flips are more frequent.

Disabling LC led to a distinct behavior: while the test accuracy grew more gradually, it exhibited significant fluctuations across epochs, reflecting instability in label predictions. Without LC, the model was unable to correct noisy labels effectively, resulting in inconsistent performance. This effect was particularly detrimental under asymmetric noise, where systematic label corruption makes robust learning more challenging.

The full SGN model significantly outperformed both ablated versions, showing steady performance gains throughout training without overfitting or instability. By combining LR and LC, SGN successfully mitigates the early overfitting observed when LR is disabled and avoids the prediction instability seen when LC is removed. This highlights the importance of both components in addressing distinct challenges posed by noisy labels: LR ensures long-term robustness, while LC stabilizes and accelerates learning.

7.3.3 Insights and Implications

The ablation study reveals that LR and LC address complementary aspects of noise robustness. LR is particularly effective in preventing overfitting to noisy samples, ensuring stability during the later stages of training. LC, on the other hand, improves early convergence by dynamically correcting label distributions, leading to more reliable learning trajectories. The synergy between these components is critical to SGN's success, enabling it to outperform its individual variants under both symmetric and asymmetric noise conditions. This insight reinforces the value of integrating both strategies to develop noise-robust learning methods that remain stable, consistent, and generalizable across challenging noise scenarios.

8 Conclusion

This study aimed to reimplement and verify the robustness of the Shifted Gaussian Noise (SGN) method proposed by Englesson and Azizpour (2024) under resource-constrained settings. By comparing SGN against standard and robust baselines like Cross-Entropy (CE), Generalized Cross-Entropy (GCE), and Early Learning Regularization (ELR) on the CIFAR-10 dataset with symmetric and asymmetric noise, we demonstrated that SGN maintains its core property of improved robustness to noisy labels.

Our experiments showed that SGN outperforms CE and ELR consistently in both symmetric and asymmetric noise scenarios. Notably, while GCE achieved superior performance in most cases, SGN was the only method that retained stability and avoided overfitting at high asymmetric noise levels, confirming its resilience under more challenging noise structures. The ablation study revealed that SGN's robustness arises from the complementary interaction of Loss Reweighting (LR) and Label Correction (LC): LR mitigates overfitting in the later stages of training, while LC accelerates convergence and stabilizes learning in the presence of noise. These findings underscore the importance of integrating both components to achieve a balanced and effective noise-robust framework.

Although limited to 100 training epochs, our results qualitatively align with the original findings, highlighting SGN's ability to preserve clean representations over noisy datasets. Future work could explore longer training durations, scalability to larger datasets, and further refinements to optimize SGN's performance across diverse conditions. This reimplementation validates SGN as a promising solution for noise-robust learning, offering stability and consistency where traditional methods falter.

9 Appendix

$$\mathbf{P}_{\mathbf{20\%}} = \begin{bmatrix} 0.8 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.2 & 0.0 \\ 0.0 & 0.8 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.2 \\ 0.0 & 0.0 & 0.8 & 0.1 & 0.0 & 0.1 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.1 & 0.8 & 0.0 & 0.1 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.8 & 0.0 & 0.0 & 0.2 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.1 & 0.1 & 0.0 & 0.8 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.2 & 0.0 & 0.0 & 0.0 & 0.8 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.2 & 0.0 & 0.0 & 0.8 & 0.0 & 0.0 \\ 0.2 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.8 & 0.0 \\ 0.0 & 0.2 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.8 \end{bmatrix}. \tag{3}$$

$$\mathbf{P}_{\mathbf{40\%}} = \begin{bmatrix} 0.6 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.4 & 0.0 \\ 0.0 & 0.6 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.4 \\ 0.0 & 0.0 & 0.6 & 0.2 & 0.0 & 0.2 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.2 & 0.6 & 0.0 & 0.2 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.6 & 0.0 & 0.0 & 0.4 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.2 & 0.2 & 0.0 & 0.6 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.4 & 0.0 & 0.0 & 0.0 & 0.6 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.4 & 0.0 & 0.0 & 0.6 & 0.0 & 0.0 \\ 0.4 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.6 & 0.0 \\ 0.0 & 0.4 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.6 \end{bmatrix}. \tag{4}$$

Table 6: Hyperparameters for Model Training

Model	**Hyperparameter**	**Value**
Shifted Gaussian Noise (SGN)	Base Learning Rate	0.1
	Warmup Epochs	5
	Alpha	0.995
	EMA Decay	0.99
Cross-Entropy (CE)	Learning Rate	0.1
	Warmup Epochs	5
Generalized Cross-Entropy (GCE)	Learning Rate	0.1
	Warmup Epochs	5
	GCE q-parameter	0.7
Early Learning Regularization (ELR)	Base Learning Rate	0.1
	Warmup Epochs	5
	λ_{ELR} (Regularization Strength)	3.0
	β (EMA Momentum for p_t Updates)	0.7

References

Englesson, Erik and Hossein Azizpour (2024). "Robust Classification via Regression for Learning with Noisy Labels". In: *International Conference on Learning Representations (ICLR)*. Retrieved from `https://openreview.net/forum?id=wfgZc3IMqo`. URL: `https://openreview.net/pdf?id=wfgZc3IMqo`.

He, Kaiming et al. (2016). "Deep residual learning for image recognition". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778.

Jiang, Heinrich et al. (2018). "To trust or not to trust a classifier". In: *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 5541–5552.

Liu, Sheng et al. (2020). "Early-Learning Regularization Prevents Memorization of Noisy Labels". In: *Advances in Neural Information Processing Systems*, pp. 20331–20342. URL: `https://arxiv.org/abs/2007.00151`.

Mao, Anqi, Mehryar Mohri, and Yutao Zhong (2023). "Cross-Entropy Loss Functions: Theoretical Analysis and Applications". In: *arXiv preprint arXiv:2304.07288*. URL: `https://arxiv.org/abs/2304.07288`.

Patrini, Giorgio et al. (2017). "Making deep neural networks robust to label noise: A loss correction approach". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2233–2241.

Zagoruyko, Sergey and Nikos Komodakis (2016). "Wide residual networks". In: *arXiv preprint arXiv:1605.07146*. DOI: `10.48550/arXiv.1605.07146`.

Zhang, Zhilu and Mert R Sabuncu (2018). "Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels". In: *Advances in Neural Information Processing Systems*, pp. 8778–8788. URL: `https://arxiv.org/abs/1805.07836`.