FEDERAL UNIVERSITY OF SANTA CATARINA
TECHNOLOGICAL CENTER
GRADUATE PROGRAM IN ELECTRICAL ENGINEERING

Jorge Kysnney Santos Kamassury

Cyclic Co-Teaching and Optimized Baselines for Robust Deep Learning with Noisy Labels

Florianópolis
2025

Jorge Kysnney Santos Kamassury

Cyclic Co-Teaching and Optimized Baselines for Robust Deep Learning with Noisy Labels

Thesis submitted to the Graduate Program in Electrical Engineering at the Federal University of Santa Catarina as a partial requirement for the degree of doctor in Electrical Engineering.

Supervisor: Prof. Danilo Silva, Ph. D.

Florianópolis
2025

Jorge Kysnney Santos Kamassury

Cyclic Co-Teaching and Optimized Baselines for Robust Deep Learning with Noisy Labels

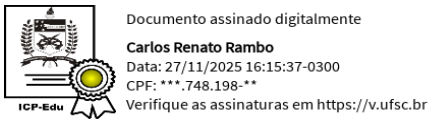
O presente trabalho em nível de doutorado foi avaliado e aprovado por banca examinadora composta pelos seguintes membros:

Prof. Cristiano Torezzan, Dr.
Universidade Estadual de Campinas

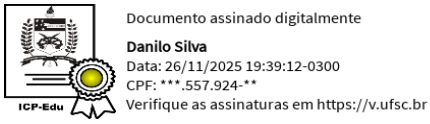
Prof. Eduardo Luiz Ortiz Batista, Dr.
Universidade Federal de Santa Catarina

Prof. Filipe Rolim Cordeiro, Dr.
Universidade Federal Rural de Pernambuco

Certificamos que esta é a **versão original e final** do trabalho de conclusão que foi julgado adequado para obtenção do título de Doutor em Engenharia Elétrica.



Prof. Carlos Renato Rambo, Dr. Sc.
Coordenador do Programa



Prof. Danilo Silva, Ph. D.
Orientador

Florianópolis, 2025.

Ficha catalográfica gerada por meio de sistema automatizado gerenciado pela BU/UFSC.
Dados inseridos pelo próprio autor.

Kamassury, Jorge Kysnney Santos
Cyclic co-teaching and optimized baselines for robust deep learning with noisy labels / Jorge Kysnney Santos Kamassury ; orientador, Danilo Silva, 2025.
117 p.

Tese (doutorado) - Universidade Federal de Santa Catarina, Centro Tecnológico, Programa de Pós-Graduação em Engenharia Elétrica, Florianópolis, 2025.

Inclui referências.

1. Engenharia Elétrica. 2. Learning with noisy labels. 3. Cyclic co-teaching. 4. Robust training . 5. Vanilla models. I. Silva, Danilo. II. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Engenharia Elétrica. III. Título.

À minha mãe, Francisca Glória Kamassury, por seu amor infinito,
por uma dedicação que sempre ultrapassou qualquer medida e
por ter sido meu maior amparo em cada etapa desta jornada
acadêmica.

ACKNOWLEDGEMENTS

Gostaria de registrar formalmente meu reconhecimento a todas as pessoas que, de modo direto ou indireto, contribuíram para que este trabalho se tornasse possível.

Ao meu orientador, Danilo Silva, que me acompanha desde o mestrado e cuja orientação dedicada e criteriosa foi decisiva em cada etapa desta trajetória. Agradeço pela paciência sempre renovável, pela dedicação incansável e pelas leituras críticas e sugestões que foram fundamentais para o amadurecimento deste trabalho. Sou grato pelas inúmeras reuniões e discussões, tanto sobre os temas centrais da tese quanto sobre os diferentes caminhos que avaliamos até definirmos sua direção. Foi um privilégio trilhar esta caminhada no universo da inteligência artificial e aprender com sua experiência e generosidade intelectual.

Aos professores Cristiano Torezzan, Eduardo Batista e Filipe Cordeiro, membros da banca examinadora, pelas leituras atentas e pelas valiosas contribuições e críticas que aprimoraram de maneira significativa esta tese.

Ao Grupo de Pesquisa em Aprendizado de Máquina e Aplicações (GAMA), por proporcionar um ambiente fértil para discussões, descobertas e aprendizagens. Em especial, ao Henrique Pickler, pela parceria nas etapas mais desafiadoras desta jornada.

À minha mãe, pelo amor incondicional, pela sabedoria transmitida e, sobretudo, pela compreensão e apoio ao permitir que eu seguisse minha jornada, mesmo diante das dificuldades inerentes a essa escolha; e aos meus irmãos, Kennedy Kamassury e Mateus Kamassury, pela hombridade, pelo caráter íntegro e por serem exemplos de boas pessoas, além do apoio e das risadas que sempre tornaram o caminho mais leve. Este trabalho é, inegavelmente, fruto do amparo de vocês.

À minha amada Ana Carolina (mineirinha), pelo amor, companheirismo e reciprocidade que tornam meus dias mais felizes, e pela compreensão generosa nos inúmeros momentos em que precisei me ausentar, emocional ou presencialmente, para dedicar-me à pesquisa, à escrita e ao desenvolvimento desta tese. Estendo ainda meus agradecimentos à sua família pelas risadas, pelos filmes de terror, pela convivência acolhedora com suas irmãs e pelos momentos alegres compartilhados com as crianças — cada uma com seus apelidos únicos — que trouxeram caos e ternura ao meu cotidiano ao longo desta jornada.

Aos insubstituíveis amigos Pablo Xavier e Jozinei Lopes, pelo apoio sempre presente, pelas discussões científico-filosóficas, pelas atualizações sobre o rock internacional e pelos dias compartilhados — das pedaladas que clarearam ideias aos domingos descendo o rio Tapajós — memórias que tornaram esta trajetória mais leve e inesquecível.

Aos amigos da graduação (Pablo dos Santos, Wandesson Duarte, Simone Carvalho e Aguinaldo Alves), pelos momentos de convivência que geraram reflexões divertidas e valiosas e que, de tantas maneiras, contribuíram para minha motivação ao longo dos anos.

Ao Instituto SENAI de Inovação em Sistemas Embarcados (ISI-SE), pelo tempo disponibilizado para a conclusão desta tese e pelas oportunidades de atuar com inteligência artificial em ambiente industrial, experiências que ampliaram profundamente minha compreensão aplicada

da área. Agradeço, em especial, a Giancarlo Marchesini, pelo apoio ao longo da minha trajetória no instituto e pelo compartilhamento de seus conhecimentos e vivências no contexto industrial; e a Deise Ferreira, cuja compreensão e flexibilidade na etapa final desta jornada foram essenciais para que eu dispusesse das condições necessárias para concluir este trabalho.

À Universidade Federal de Santa Catarina e ao Programa de Pós-Graduação em Engenharia Elétrica, pelo excelente suporte à pesquisa científica e pela reconhecida qualidade de seus servidores e docentes.

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), pelo financiamento (processo nº 141487/2020-8), que tornou possível a realização deste trabalho.

Por fim, enfatizo que concluir esta tese vai além de finalizar uma etapa acadêmica: simboliza encerrar um ciclo marcado por desafios, descobertas e amadurecimento, representando a materialização do sonho de um menino do interior da Amazônia, apaixonado pela ciência e guiado pelo desejo sincero de honrar aqueles que sempre estiveram ao seu lado. Levo comigo não apenas o conhecimento construído, mas também uma imensa gratidão por todos que fizeram parte desta caminhada.

"Test all things; hold fast what is good."
1 Thessalonians 5:21[a]

[a] An ancient teaching that reveals how wisdom arises from the ability to discern with clarity amid the noise that permeates human experience.

RESUMO

Redes neurais profundas exibem notável capacidade de aprendizado e generalização em tarefas complexas, mas são particularmente suscetíveis à memorização de rótulos incorretos, o que compromete seu desempenho. Conjuntos de dados em larga escala frequentemente contêm anotações imprecisas, devido à qualidade dos dados brutos, à complexidade da rotulação ou às variações no julgamento dos anotadores, tornando o aprendizado com rótulos ruidosos um desafio central em aprendizado profundo. Entre os métodos existentes para mitigar o efeito do ruído de rótulo, o *Co-Teaching* se destaca por treinar dois modelos em paralelo, permitindo a identificação mútua de amostras potencialmente ruidosas por meio da seleção cruzada. No entanto, o método ainda apresenta risco de acúmulo de erros, e certas estratégias de retenção e de decaimento da taxa de aprendizado podem reforçar esse problema, favorecendo o *overfitting*. Para abordar essas limitações, desenvolvemos o *Cyclic Co-Teaching* (CCT), que reduz o risco de *overfitting* por meio de modulações cíclicas na taxa de aprendizado e na retenção de amostras: variações periódicas que estabelecem uma dinâmica alternante entre fases de especialização (aprendizado intensivo) e consolidação (estabilização do conhecimento). Inicialmente, analisamos como diferentes valores da taxa de aprendizado (baixos, intermediários e altos) afetam a robustez de modelos *vanilla* (treinados sem mecanismos específicos para mitigação do ruído) frente a rótulos ruidosos, revelando que taxas intermediárias favorecem a discriminação entre amostras limpas e ruidosas, enquanto taxas muito baixas resultam em discriminação insuficiente e taxas muito altas geram instabilidade. Em seguida, usando o *framework Co-Teaching*, investigamos os efeitos das modulações cíclicas de aprendizado e retenção sobre métricas de generalização e consistência entre modelos, consolidando o *framework* CCT para aproveitar essas variações de robustez. Por fim, propomos uma abordagem de otimização univariada em duas etapas para o ajuste eficiente dos hiperparâmetros do método. Os resultados demonstram que o CCT supera consistentemente métodos do estado da arte em conjuntos sintéticos (CIFAR-10, CIFAR-100, Tiny-ImageNet) e reais (Animal-10N, Food-101N, Clothing1M), com ganhos expressivos, especialmente em cenários de alta taxa de ruído. Além disso, verificou-se que a otimização adequada dos modelos *vanilla* não apenas superou consistentemente os resultados previamente reportados na literatura para esses mesmos modelos, mas, em vários casos, alcançou desempenho superior até mesmo a métodos específicos de aprendizado com rótulos ruidosos. Esses achados reforçam a importância de estabelecer linhas de base robustas e sugerem que parte das melhorias frequentemente atribuídas a técnicas especializadas pode, na realidade, decorrer de diferenças metodológicas e de configurações de treinamento subótimas.

Palavras-chave: Aprendizado com Rótulos Ruidosos. Cyclic Co-Teaching. Treinamento Robusto. Modelos Vanilla. Taxa de Aprendizado.

RESUMO ESTENDIDO

Introdução

Redes neurais profundas demonstram notável capacidade de aprendizado e generalização em tarefas complexas, impulsionando avanços em diversos setores tecnológicos. Contudo, essa mesma capacidade as torna particularmente vulneráveis à memorização de rótulos incorretos, comprometendo significativamente seu desempenho. Essa vulnerabilidade é especialmente crítica, dado que conjuntos de dados em escala industrial frequentemente contêm anotações imprecisas, seja pelo próprio processo de anotação, por limitações na qualidade dos dados brutos, pela complexidade da tarefa de rotulação ou por variações no julgamento e na especialização dos anotadores.

Nesse contexto, o aprendizado com rótulos ruidosos (*Learning with Noisy Labels* – LNL) representa um desafio central no aprendizado profundo. A literatura recente identifica quatro estratégias principais para lidar com esse problema: modificação das funções de perda, algoritmos de treinamento robustos, processamento de dados e alterações nas arquiteturas dos modelos. Entre os métodos clássicos, destaca-se o *Co-Teaching*, que treina dois modelos simultaneamente para identificar mutuamente amostras potencialmente ruidosas por meio de seleção cruzada baseada no critério de pequena perda. Essa abordagem utiliza seleção cooperativa para reduzir o viés de aprendizado e frequentemente serve como base para métodos LNL mais avançados.

No entanto, mesmo com a seleção cooperativa, o acúmulo de erros pode persistir, favorecendo o *overfitting*. O *Co-Teaching* tradicional segue uma estratégia de treinamento estática, caracterizada por dois aspectos principais: a taxa de retenção de amostras decresce gradualmente até certa época e, a partir daí, permanece constante, enquanto a taxa de aprendizado é monotonicamente decrescente. Essa política fixa de retenção exige um balanceamento cuidadoso: valores altos expõem os modelos a muitas amostras potencialmente ruidosas, favorecendo a memorização; já valores baixos limitam a diversidade e restringem o aprendizado robusto. Além disso, uma política de decaimento monotônico da taxa de aprendizado pode restringir a exploração de regimes mais robustos, uma vez que a robustez frente a rótulos ruidosos mostra-se sensível à magnitude da taxa de aprendizado — sendo, em geral, maior em valores altos e menor em valores baixos. Consequentemente, essas políticas de treinamento podem tanto desperdiçar oportunidades de utilizar mais amostras em fases robustas quanto deixar de oferecer proteção adequada em fases sensíveis à memorização.

Paralelamente, sob outra perspectiva igualmente relevante do LNL, observam-se questões metodológicas frequentemente negligenciadas. Modelos básicos (*vanilla*) costumam ser subestimados devido a configurações subótimas de hiperparâmetros, como a taxa de aprendizado, enquanto avaliações incorretas, como o uso inadequado de conjuntos de teste em funções de validação, dificultam comparações justas entre métodos. Esses pontos reforçam a necessidade de critérios consistentes e avaliações robustas das técnicas LNL.

Objetivo

Desenvolver um método robusto e eficiente para o treinamento de redes neurais com rótulos ruidosos, capaz de mitigar o problema de acúmulo de erros presente no *Co-Teaching* tradicional por meio de estratégias cíclicas de taxa de aprendizado e retenção de amostras. Além disso, busca-se avaliar sistematicamente fatores críticos para a robustez em LNL e evidenciar os benefícios do ajuste adequado de hiperparâmetros, mesmo em modelos *vanilla*.

Metodologia

O método CCT, proposto nesta tese, estende o *framework Co-Teaching* mediante a incorporação de estratégias cíclicas destinadas a mitigar o acúmulo de erros durante o treinamento conjunto das redes. A metodologia foi estruturada em três etapas principais.

Primeiramente, investigou-se empiricamente o impacto da taxa de aprendizado na robustez de modelos *vanilla* face a rótulos ruidosos. Experimentos sistemáticos em CIFAR-10 e CIFAR-100 foram conduzidos sob diferentes tipos de ruído (simétrico e assimétrico) e níveis variados. Métricas como a perda em amostras limpas e ruidosas, bem como a razão entre elas, foram analisadas para quantificar a capacidade discriminativa dos modelos em diferentes configurações.

Em seguida, explorando as variações de robustez dos modelos ao longo do treinamento no *framework Co-Teaching*, observadas por métricas de generalização e consistência entre modelos, desenvolveu-se o *framework* CCT. Este substitui estratégias estáticas por abordagens cíclicas em dois eixos principais: a taxa de aprendizado, ajustada ciclicamente para escapar de mínimos locais e explorar diferentes regimes de robustez; e a taxa de retenção de amostras, sincronizada com essas variações para admitir mais exemplos em fases robustas (taxas de aprendizado altas) e aplicar filtragem mais seletiva em fases sensíveis (taxas de aprendizado baixas).

Por fim, projetou-se um *framework* de otimização univariada em duas etapas para ajuste eficiente dos hiperparâmetros. Na primeira etapa realiza-se a otimização univariada da taxa de aprendizado, obtendo-se um modelo *vanilla* calibrado; na segunda, esse modelo é usado para inicializar a otimização univariada dos hiperparâmetros da função de retenção cíclica — abordagem mais eficiente que uma busca em grade completa.

Os experimentos foram realizados com conjuntos de dados sintéticos (CIFAR-10, CIFAR-100, Tiny-ImageNet) e reais (Animal-10N, Food-101N, Clothing1M), e diversas arquiteturas neurais. As avaliações seguiram a metodologia convencional de aprendizado de máquina, utilizando conjuntos de treinamento, validação e teste distintos, com critério de parada antecipada e seleção do melhor modelo com base no desempenho de validação.

Resultados e Discussão

Esta tese investigou a influência da taxa de aprendizado na capacidade de discriminar entre amostras limpas e ruidosas, especialmente em modelos *vanilla*. Diferentemente da literatura, que enfatiza principalmente taxas altas, realizamos uma análise sistemática de valores baixos, intermediários e altos, considerando também sua evolução temporal ao longo do treinamento. Os resultados mostraram que taxas intermediárias favorecem a separação entre perdas de exemplos limpos e ruidosos, enquanto taxas muito baixas levam à discriminação insuficiente e taxas muito altas podem gerar instabilidade. A análise temporal revelou que a robustez da discriminação depende não apenas do valor da taxa de aprendizado, mas também da duração de exposição, sugerindo que estratégias de decaimento — combinando valores iniciais altos e reduções subsequentes — podem explorar os benefícios de ambos os regimes. Esses resultados evidenciam que a simples otimização da taxa de aprendizado atua como um mecanismo eficaz de regularização implícita.

Os resultados da investigação sobre as políticas de variação cíclica da taxa de aprendizado e de retenção no *framework Co-Teaching* indicam que essas variações introduzem uma dinâmica oscilatória controlada entre as redes parceiras, em constraste com o padrão de divergência gradual e irreversível observado no *Co-Teaching* tradicional. Essa abordagem promove ciclos alternados de especialização e consolidação: durante os picos da taxa de aprendizado, observam-se aumentos na divergência das predições e redução na correlação de seleção de amostras, caracterizando fases de especialização em que as redes desenvolvem perspectivas complementares sobre diferentes subconjuntos dos dados de treinamento. Nos períodos de taxa de aprendizado reduzida, ambas as métricas reconvergem gradualmente, indicando fases

de consolidação com maior consenso na seleção de amostras confiáveis. Essa dinâmica rítmica gera "momentos de divergência" que favorecerem a exploração de diferentes regiões do espaço de dados, seguidos por fases de reconvergência que consolidam o conhecimento adquirido. A análise da correlação na seleção de amostras mostra que essa alternância entre especialização e consolidação torna-se mais pronunciada quando ambas as taxas variam ciclicamente e de forma sincronizada, sendo a configuração em fase (picos coincidentes) mais efetiva do que a inversa (picos desencontrados).
O método CCT demonstrou superar consistentemente os métodos estado da arte em diferentes conjuntos de dados, arquiteturas e cenários de ruído. Em conjuntos sintéticos, como CIFAR-10 e CIFAR-100, o CCT apresentou ganhos expressivos, particularmente em cenários com altas taxas de ruído, e resultados semelhantes foram observados em distintas arquiteturas neurais. Em conjuntos reais com ruído natural (Animal-10N, Food-101N, Clothing1M), o CCT manteve desempenho robusto, mostrando sua eficácia em situações práticas com padrões de ruído complexos.
O estudo de ablação confirmou a contribuição de cada componente do CCT, evidenciando degradação de desempenho quando o mecanismo de *checkpoint*, a otimização de hiperparâmetros ou o pré-treinamento *vanilla* eram removidos, sendo este último particularmente relevante em cenários com altas taxas de ruído. A análise do impacto da otimização da taxa de retenção cíclica revelou que o valor ótimo do hiperparâmetro ϵ frequentemente coincide com a taxa de ruído de rótulo τ, e que a escolha de ϵ baseada no conjunto de validação produz resultados iguais ou superiores aos obtidos ao definir $\epsilon = \tau$.
Um aspecto significativo revelado pelos experimentos foi que o ajuste adequado dos modelos *vanilla* permitiu que eles superassem consistentemente os resultados da literatura para modelos equivalentes e, em muitos casos, até mesmo métodos específicos de LNL. Essa observação destaca a importância de estabelecer *baselines* robustas e questiona a magnitude real das melhorias atribuídas a técnicas especializadas.

Considerações Finais

Esta tese propõe uma técnica para o treinamento robusto de redes neurais profundas na presença de rótulos ruidosos, culminando no desenvolvimento do método CCT. O CCT mitiga o acúmulo de erros observado no *Co-Teaching* tradicional por meio de modulações cíclicas na taxa de aprendizado e na taxa de retenção de amostras, estabelecendo uma dinâmica oscilatória controlada, que alterna entre fases de especialização e consolidação. Essa abordagem explora as variações naturais de robustez dos modelos durante o treinamento. Os resultados mostram que o CCT supera, de forma consistente, os métodos do estado da arte em diferentes conjuntos de dados e cenários de ruído, especialmente sob condições severas. Além disso, o *framework* de otimização univariada simplifica o ajuste de hiperparâmetros, tornando o método aplicável mesmo na ausência de informações precisas sobre o ruído.
Além da contribuição algorítmica, a tese destaca aspectos metodológicos relevantes em LNL, como a subestimação de modelos *vanilla* e o impacto de práticas de avaliação que podem comprometer a comparação entre métodos. Embora, em muitos cenários, modelos *vanilla* apresentem desempenho inferior ao de métodos específicos de LNL, observa-se que, quando bem otimizados, podem se aproximar — e, em alguns casos, até superar — tais métodos. Esses resultados reforçam a importância de estabelecer *baselines* sólidas e de distinguir ganhos metodológicos genuínos de efeitos decorrentes de configuração ou avaliação.

Palavras-chave: Aprendizado com Rótulos Ruidosos. Cyclic Co-Teaching. Treinamento Robusto. Modelos Vanilla. Taxa de Aprendizado.

ABSTRACT

Deep neural networks exhibit remarkable learning and generalization capabilities in complex tasks but are particularly susceptible to memorizing incorrect labels, which compromises their performance. Large-scale datasets often contain inaccurate annotations due to raw data quality, labeling complexity, or variations in annotators' judgment, making learning with noisy labels a central challenge in deep learning. Among existing methods to mitigate the effect of label noise, Co-Teaching stands out by training two models in parallel, enabling the mutual identification of potentially noisy samples through cross-selection. However, the method still presents a risk of error accumulation, and certain retention strategies and learning rate decay schedules can reinforce this problem, favoring overfitting. To address these limitations, we developed Cyclic Co-Teaching (CCT), which reduces the risk of overfitting through cyclic modulations of the learning rate and sample retention — periodic variations that establish an alternating dynamic between specialization phases (intensive learning) and consolidation phases (knowledge stabilization). Initially, we analyzed how different learning rate values (low, intermediate, and high) affect the robustness of vanilla models (trained without specific noise-mitigation mechanisms) under noisy labels, revealing that intermediate rates favor discrimination between clean and noisy samples, while very low rates lead to insufficient discrimination and very high rates cause instability. Next, within the Co-Teaching framework, we investigated the effects of cyclic modulations of learning and retention on generalization and inter-model consistency metrics, consolidating the CCT framework to leverage these robustness variations. Finally, we propose a two-step univariate optimization approach for efficient hyperparameter tuning. The results demonstrate that CCT consistently outperforms state-of-the-art methods on synthetic datasets (CIFAR-10, CIFAR-100, Tiny-ImageNet) and real-world datasets (Animal-10N, Food-101N, Clothing1M), with significant gains, particularly in high-noise scenarios. Moreover, proper optimization of vanilla models not only consistently surpassed previously reported results for the same models in the literature but, in several cases, achieved superior performance even compared to specialized noisy-label learning methods. These findings reinforce the importance of establishing strong baselines and suggest that part of the improvements often attributed to specialized techniques may, in fact, stem from methodological differences and suboptimal training configurations.

Keywords: Learning with Noisy Labels. Cyclic Co-Teaching. Robust Training. Vanilla Models. Learning Rate.

LIST OF TABLES

LIST OF ALGORITHMS

LIST OF FIGURES

LIST OF SYMBOLS

$\mathcal{X}$	Data feature space
$\mathcal{Y}$	True label space
$\tilde{\mathcal{Y}}$	Noisy label space
$\mathcal{D}$	Training dataset
$\mathcal{D}_{\text{val}}$	Validation dataset
$\mathcal{D}_{\text{test}}$	Test dataset
$p(\mathbf{x}, \mathbf{y})$	Clean joint distribution
$p(\mathbf{x}, \tilde{\mathbf{y}})$	Observed joint distribution
$\mathbf{x}$	Data example (vector)
y_i	True label of instance i
$\tilde{y}_i$	Noisy label of instance i
$\mathbf{y}$	True label (one-hot vector)
$\tilde{\mathbf{y}}$	Noisy label (one-hot vector)
$f(\mathbf{x}; \theta)$	Neural network prediction
θ	Model parameters
Θ	Parameter space
ℓ	Loss function
$\mathbb{E}$	Expectation over $\mathcal{D}$
η	Learning rate
τ	True label noise rate
c	Number of classes
$\mathbf{T}$	True noise transition matrix
$\hat{\mathbf{T}}$	Estimated noise transition matrix

LIST OF ABBREVIATIONS AND ACRONYMS

BMM	Beta Mixture Model
BCE	Binary Cross-Entropy
CCT	Cyclic Co-Teaching
CE	Cross-Entropy
CNN	Convolutional Neural Network
DA	Data Augmentation
DL	Deep Learning
DNN	Deep Neural Network
GCE	Generalized Cross Entropy
GMM	Gaussian Mixture Model
KL	Kullback-Leibler
LNL	Learning with Noisy Labels
MAE	Mean Absolute Error
MSE	Mean Squared Error
ML	Machine Learning
RCE	Reverse Cross Entropy
SMA	Simple Moving Average
SCE	Symmetric Cross Entropy
SGD	Stochastic Gradient Descent
SGDR	Stochastic Gradient Descent with Warm Restarts
SSL	Semi-Supervised Learning

CONTENTS

1 INTRODUCTION

Deep learning (DL) has driven significant transformations across multiple sectors of contemporary technology. From recommendation systems that anticipate individual preferences, to computer vision algorithms enabling autonomous vehicle navigation, and medical tools capable of early tumor detection in magnetic resonance imaging scans and anomaly detection in electrocardiograms, deep neural networks (DNNs) have stood out for their learning capacity and generalization in complex tasks.

However, the success of these models is intrinsically linked to the availability of high-quality data. Historically, the effective development of machine learning (ML) systems has been based on the premise that carefully collected and labeled datasets are essential to ensure satisfactory model performance. Nevertheless, DNNs introduce a new layer of complexity that challenges this premise: their extraordinary capacity to memorize data.

This characteristic was highlighted by Zhang et al. (2017), who demonstrated that modern convolutional neural networks (CNNs) are capable of achieving high performance even when trained with completely random labels. Although this remarkable memorization ability enables the learning of subtle and complex patterns in legitimate data, it also makes the models particularly vulnerable to overfitting in the presence of noise, severely compromising their generalization capacity.

This inherent vulnerability of DNNs becomes even more critical when we consider that, in practice, perfectly labeled data is a rarely attainable idealization. In reality, almost every real-world dataset contains some degree of annotation imprecision, whether due to the low quality of raw data, the complexity of the labeling task (which may require high expertise), or variations in annotators' skill levels and judgment (XIAO et al., 2015).

In the current context of rapidly expanding data availability, a striking contradiction emerges: advances in large-scale data collection techniques have been accompanied by increased exposure to the problem of noisy labeling. The construction of highly curated datasets, such as ImageNet (DENG et al., 2009), which required years of intensive human effort to ensure reliable annotations, has become increasingly unfeasible given the growing need for scalability and agility in developing DL-based solutions. To meet this demand, the industry has increasingly adopted datasets with limited curation, whose annotations are generated by automated or semi-automated means. In practice, although this strategy enables the construction of large-scale databases, it also entails significant compromises in data quality, with label noise being one of the most pervasive challenges (HAN et al., 2021; SONG et al., 2023).

Indeed, even datasets traditionally regarded as reliable (such as ImageNet, CalTech-256, MNIST, and ChestX-ray14) are not exempt from these issues. As evidenced by Northcutt, Jiang & Chuang (2021), these datasets also exhibit non-negligible levels of label noise, indicating that the problem of noisy labels is not an isolated exception but rather a recurring phenomenon in large-scale datasets used in DL.

In this context, the combination of the inherent vulnerability of DNNs to memorization

and the presence of noisy labels creates a problematic scenario where models tend to learn spurious patterns, severely compromising their generalization ability. This limitation not only degrades performance on unseen data but also directly impacts the reliability of systems in practical applications where high accuracy is crucial.

Consequently, learning with noisy labels (LNL) emerges as one of the most challenging problems in DL. Its relevance becomes even more evident in sensitive applications such as autonomous driving and medical diagnosis, which demand robust and reliable models for decision-making. Such a context underscores the importance of understanding the training dynamics under noisy data and developing strategies that reduce or eliminate the negative impact of corrupted labels while maximizing the use of available information.

1.1 VULNERABILITY OF DEEP NEURAL NETWORKS TO LABEL NOISE

Label noise is a recurring challenge in supervised tasks, compromising the performance of ML models. Classical algorithms, such as support vector machines (SVMs) and decision trees, tend to exhibit greater intrinsic robustness to certain types of noise, especially in scenarios with moderately corrupted data. Nevertheless, they remain sensitive to imperfect supervision and frequently require specific adaptations to address this problem. Moreover, their limited representational capabilities and scalability difficulties make them less suitable for complex domains, such as computer vision and natural language processing, areas where DNNs demonstrate superior performance.

This superiority, however, does not render DNNs immune to label noise; on the contrary, they exhibit significant vulnerability in this context. Zhang et al. (2017) demonstrated that DL models can memorize completely random data, showing that their high expressive capacity, while crucial for complex tasks, also makes them particularly susceptible to overfitting in the presence of incorrect labels. Unlike classical overfitting, which occurs when the model learns genuine but overly specific patterns, overfitting to label noise involves the assimilation of erroneous associations, systematically compromising generalization.

A particularly concerning aspect is that, even with the use of traditional regularization techniques such as data augmentation (DA), dropout, and batch normalization, this memorization behavior is not completely prevented (SONG et al., 2023). Modern architectures tend to memorize simple examples in the early epochs of training and gradually adapt to more difficult instances, including those that are incorrectly labeled. Thus, in the presence of noise, these models frequently internalize erroneous information, compromising their generalization capacity.

Faced with this challenging scenario, recent studies have focused on developing methods robust to label noise. The current landscape of LNL research reveals a rapidly expanding area, with relevant intersections with other ML subfields, such as uncertainty estimation (HUANG et al., 2022), semi-supervised learning (LI; SOCHER; HOI, 2020), self-supervised learning (ZHELTONOZHSKII et al., 2022), robust optimization, active learning (BERNHARDT

et al., 2022), confidence-based learning (NORTHCUTT; JIANG; CHUANG, 2021), and out-of-distribution data detection (ALBERT et al., 2022), among others. These connections have driven contributions ranging from theoretical formulations to practical solutions applied in scenarios sensitive to annotation errors, such as medical diagnosis and autonomous driving systems.

1.2 CURRENT CHALLENGES IN LNL

As a rapidly emerging research area, LNL faces challenges ranging from fundamental conceptual issues to practical methodological limitations. From a theoretical perspective, one of the open problems is the formal definition of label noise, with two predominant approaches in the literature.

The first is the class-conditional noise model, which assumes a latent distribution of clean labels corrupted through a transition matrix $\mathbf{T}$ (NATARAJAN et al., 2013; PATRINI et al., 2017). The second is the label-distribution model, which directly postulates a probability distribution over noisy labels (FRENAY; VERLEYSEN, 2014). While the first provides theoretical guarantees of statistical consistency, it depends on accurately estimating $\mathbf{T}$ and often ignores instance-dependent noise; the second offers greater flexibility but lacks a firm probabilistic foundation and may hinder the interpretation of labels as ground truth. This distinction directly impacts the development of LNL methods, as many algorithms explicitly rely on one of these assumptions.

From a methodological standpoint, one of the central challenges lies in the development and evaluation of techniques that enhance the robustness of ML models against incorrect labels (CARNEIRO, 2024). In this context, the literature has converged on four main lines of investigation, which are distinguished by their strategies, advantages, and limitations:

- Loss functions: Aim to reduce the model's sensitivity to noisy labels by modifying traditional loss functions. Strategies include applying softer penalties to hard examples, reweighting samples by estimated reliability, and using heuristics-inspired regularizations such as early learning. These methods are generally compatible with various architectures and easy to integrate. However, studies have shown that robust losses tend to perform better under low-to-moderate noise levels and small-class problems, while their effectiveness decreases in highly corrupted or large-scale datasets. Moreover, they often require carefully tuned hyperparameters that may hinder generalization across different noise regimes (SONG et al., 2023);

- Training algorithms: Encompass strategies that make the learning process more robust to noisy labels, including meta-learning methods, self-supervised pretraining, multi-model approaches, and techniques inspired by semi-supervised learning (SSL). In general, these solutions adjust the training regime to mitigate confirmation bias, leverage more stable representations, or adapt the model to the observed noise. Despite their demonstrated potential, they face challenges such as high computational cost, complexity in reliably

selecting samples, and scalability limitations (JIANG et al., 2018; LI; SOCHER; HOI, 2020; SONG et al., 2023);

- Data processing: Involves strategies that process or select training data to facilitate learning with noisy labels. It includes adversarial training, which applies small perturbations to samples to increase robustness, and data selection or cleaning methods, which prioritize clean and informative examples. It also covers approaches with multiple labels per instance, the use of prior knowledge to restructure data, and DA techniques combined with semi-supervised schemes. Despite robustness gains, these methods face challenges such as inadvertent exclusion of difficult examples, reliance on poorly generalizable heuristics, static thresholds, and high computational cost (SONG et al., 2023; CARNEIRO, 2024);
- Model architecture: Includes strategies that modify the network structure to directly handle noise in labels. The most common methods involve label transition modules, which learn mappings between clean and noisy labels, and graph-based models, which utilize graph neural networks to represent relationships between samples and support label selection or correction. While they can offer theoretical guarantees of consistency, these methods face challenges such as identifiability problems, strong assumptions about noise structure, and difficulties in constructing and training graph representations (SONG et al., 2023; CARNEIRO, 2024).

Despite promising results, the understanding of how different approaches suit distinct forms of noise remains open, as well as the role of interactions between components of the learning process.

Another methodological challenge is the establishment of adequate experimental configurations to evaluate different types of noise. Currently, there is a multiplicity of benchmarks and protocols, which vary in terms of noise type, corruption rate, dataset size, and class distribution. While this diversity reflects the complexity of real-world scenarios, it also hinders systematic comparison between methods, given the practical difficulty and, in many cases, the inherent intractability of evaluating all proposed models across all possible scenarios.

Recognizing the relevance of classical challenges in LNL, this thesis adopts a complementary direction within the *training-algorithm paradigm*, investigating the influence of training dynamics — particularly the learning rate and sample retention rate in the context of collaborative training — on model robustness against noisy labels. Furthermore, we demonstrate how careful optimization of traditional baselines can reveal the true potential of both the proposed strategy and established methods, a fundamental but frequently underestimated practice.

Unlike loss- or data-centered strategies, training-based approaches act directly on the learning process itself, intervening at the level of training dynamics to modulate the model's behavior without altering architectures or supervision signals. This orthogonality allows

methods that are computationally efficient, broadly compatible with existing frameworks, and capable of enhancing robustness through simple yet effective adjustments in the training schedule.

1.3 TRAINING METHODOLOGY AND LEARNING RATE

Among the motivations guiding this thesis, two interrelated observations stand out which, although they have received limited attention in the LNL literature, exert a fundamental impact on understanding the field.

The first observation concerns methodological shortcomings in the evaluation of LNL methods. A careful examination of the literature reveals practices that undermine the validity of empirical comparisons, most notably the inappropriate use of the test set to perform tasks that should be restricted to the validation stage. Such misuse leads to information leakage and artificially inflated performance estimates. Although querying the test set during hyperparameter tuning may help identify configurations that perform well on that specific dataset, this procedure fundamentally violates the independence assumption required for a reliable estimate of generalization (HASTIE; TIBSHIRANI; FRIEDMAN, 2009; GOODFELLOW; BENGIO; COURVILLE, 2016). By allowing test performance to influence model selection decisions, the test set ceases to function as an unbiased hold-out set and effectively becomes a validation set. This induces overfitting to the test data and results in overly optimistic estimates that fail to reflect the model's true generalization ability to unseen samples (RASCHKA, 2018).

The second observation, closely related to the first, concerns the role of optimization hyperparameters in robustness to label noise. While much of the research efforts in LNL focus on developing specialized architectures or complex algorithms, emerging evidence suggests that fundamental aspects of the training process, such as the optimization algorithm itself (FORET et al., 2021) and basic hyperparameters, such as batch size (ROLNICK et al., 2018) and learning rate (TANAKA et al., 2018), can influence the ability of models to resist overfitting caused by incorrect labels.

In particular, Tanaka et al. (2018) demonstrated that higher learning rates tend to promote greater robustness to noise, hindering the memorization of spurious patterns associated with erroneous labels. This result is especially significant, as it suggests that adequate optimization configurations can contribute to noise robustness, including in vanilla models, without relying on architectural modifications or prior knowledge about the type of corruption present in the data labels.

Despite this evidence, it is common to observe in the LNL literature a concerning methodological asymmetry: while proposed methods undergo hyperparameter fine-tuning stages, this same care is frequently not extended to vanilla models used as baselines. This disparity raises fundamental questions about the validity of comparisons: do the reported gains actually stem from the intrinsic superiority of the proposed method, or do they partially reflect an unfair comparison with suboptimized models?

By adopting default values or those inherited from other domains without critical reevaluation, one may significantly underestimate the true potential of traditional baselines, compromising both the relative evaluation of methods and a more precise understanding of the factors that actually contribute to noise robustness. This methodological gap not only affects the perception of progress in the field, but may also mask important discoveries about fundamental robustness mechanisms.

1.4 CYCLIC TRAINING

The Co-Teaching method (HAN et al., 2018), proposed as a robust solution to the problem of noisy labels, is based on the principle that two independently trained models can mutually identify examples with incorrect labels through cross-selection. This established strategy within noisy sample selection approaches is recognized for its effectiveness and flexibility, frequently serving as a foundation for advanced LNL methods. By combining co-training with low-loss sample selection, Co-Teaching helps reduce the typical overfitting of training with noisy data. Nevertheless, even with cooperative sample selection designed to reduce bias, error accumulation may still occur, compromising its effectiveness as training progresses.

A fundamental reason for this to occur is the gradual convergence of the two models throughout training. As both are optimized based on the same filtered data and similar loss functions, their internal representations tend to become progressively more similar (YU et al., 2019). This convergence compromises the diversity necessary for effective cross-selection: the more similar the models become, the lower their ability to mutually detect noisy examples, which degrades filtering quality and makes training increasingly susceptible to label noise. To mitigate this problem, different strategies have been explored, such as model disagreement (YU et al., 2019), joint loss mechanisms (WEI et al., 2020), and self-supervised learning (TAN et al., 2021).

A particularly distinct approach is CJC-Net (ZHANG et al., 2021), which combines the O2U-Net framework (HUANG et al., 2019) with a two-network architecture trained in parallel. Inspired by O2U-Net, the method adopts a cyclic training phase, where the learning rate is periodically modulated to force the network to alternate between underfitting and overfitting. During this process, the loss values of each sample are monitored, and those with inconsistent behavior or high loss are progressively removed, allowing subsequent training to occur on a cleaner subset.

Although it employs two networks, CJC-Net does not follow the typical Co-Teaching dynamics, since the models are trained independently in the pre-training and fine-tuning phases. Collaboration between networks occurs only in the intermediate noise elimination phase, through a joint loss function used to select samples with the lowest combined loss. Thus, the method extends O2U-Net at the ensemble level, maintaining progressive noise removal throughout training as its central principle.

This integration between cyclic training and parallel networks represents a promising hybrid approach. In contrast, the application of cyclic learning rate schedules in the original Co-Teaching framework, maintaining its cross-selection logic, is still underexplored. This strategy may favor divergence between models throughout training, reinforcing their mutual noise detection capability. The effectiveness of this type of oscillation is supported by established methods in the literature, such as SGDR (LOSHCHILOV; HUTTER, 2017). Its incorporation into Co-Teaching constitutes a direction to be explored, with potential to mitigate premature convergence between models and reinforce the robustness of cross-filtering.

Another aspect closely related to premature convergence concerns how the sample retention rate is controlled throughout training. Although Co-Teaching employs a progressive reduction of this rate in the early epochs, it usually stabilizes at a fixed value from a certain point onwards. In practice, this means retaining a constant fraction of samples until the end of training, regardless of the learning phase. This rigidity can be inefficient given the dynamic nature of the optimization process, especially in the final stages, when more aggressive filtering could help mitigate overfitting to remaining noisy labels. By maintaining a fixed rate for a long period, the method tends to repeatedly select the same examples, including shared errors, which reduces diversity between models and weakens the benefit of cross-selection.

This limitation suggests the need for a more flexible sample retention policy, capable of considering model evolution throughout training, especially in the most sensitive stages. Generally, a natural direction is to seek to align the degree of filtering with different learning stages, making the process more sensitive to memorization risk and learning progression.

1.5 GOALS AND CONTRIBUTIONS

The main objective of this thesis is to investigate how controlled variations in learning rate and sample retention rate can be systematically exploited within the Co-Teaching framework to increase the robustness of DL models to label noise. Situated within the broader class of training-algorithm approaches, this work explores how the modulation of training dynamics can enhance model resilience in a simple and cost-effective manner, complementing prior research focused on loss design and data processing. This investigation culminates in the development of the Cyclic Co-Teaching (CCT) method, which integrates coordinated cyclic variations of these components to promote a dynamic balance between learning and robustness to label noise.

The thesis presents the following contributions:

- Empirical investigation of the role of learning rate in robustness to noisy labels, with systematic experiments that confirm findings from the literature and deepen understanding of its relationship with noise memorization;
- Experimental evidence that vanilla models, when subjected to simple learning rate optimization, frequently neglected in benchmarks, can outperform established LNL

methods, highlighting the importance of calibration even in simple configurations and the need for more critical comparisons;

- Development of the CCT method as a practical and effective solution for LNL that combines coordinated cyclic variations of learning rate and sample retention rate, and a checkpoint mechanism for efficient continuity between cycles;
- Proposition of a simple univariate optimization framework for tuning CCT hyperparameters, compatible with clean or noisy validation sets, expanding its applicability in real-world scenarios;

It is worth emphasizing that the choice of the Co-Teaching framework is deliberate and theoretically grounded. Co-Teaching remains the canonical reference within multi-network training methods for LNL, providing a simple yet effective collaborative paradigm that has inspired numerous extensions such as Co-Teaching+ (YU et al., 2019) and JoCoR (WEI et al., 2020). Building upon this well-established foundation, the present work advances the understanding of collaborative training dynamics under noisy supervision and demonstrates how cyclic coordination of learning and retention rates can further enhance robustness beyond existing variants.

1.6 DOCUMENT ORGANIZATION

This thesis is structured to provide a logical progression from theoretical foundations to the proposal and experimental validation of the CCT method, and is organized as follows:

- In Chapter 2, the fundamentals and state of the art of LNL are presented, including a review of label noise sources, noise type taxonomies, problem formulation, and a systematic analysis of the main methods. In this chapter, the necessary context is established to understand subsequent contributions and the proposed investigations are positioned within the current LNL landscape;
- In Chapter 3, frequently overlooked issues in the LNL literature are discussed, emphasizing rigorous evaluation practices and critical optimization of traditional baselines. Key problems are analyzed, including information leakage, evaluation practices that can obscure peak performance, and underestimation of vanilla models. Additionally, a comprehensive empirical investigation is presented on the role of learning rate as an implicit regularization mechanism against label noise, extending previous findings through detailed static and temporal analyses. This chapter establishes both a methodological foundation for more rigorous evaluations and evidence supporting robust strategies based on learning rate variation.
- In Chapter 4, the Co-Teaching framework is examined from its algorithmic formulation and conventional sample retention policy to its limitations, proposing strategies based

on coordinated cyclic variations in learning and retention rates, evaluating through controlled experiments how this coordination, in synchronized or alternating manner, affects both generalization capacity and divergence between partner networks, quantified by metrics such as prediction disagreement and sample selection correlation;

- In Chapter 5, the CCT method is formally presented, integrating coordinated cyclic variations of learning and sample retention rates into the Co-Teaching framework, detailing its operational architecture, including the main algorithm, checkpoint mechanism, and initialization strategies. Additionally, a structured optimization framework is introduced, aimed at both obtaining robust baselines and defining adequate configurations for the cyclic parameters of the proposed method;
- In Chapter 6, experimental results obtained through CCT application on multiple datasets and noise scenarios are reported, including systematic comparisons with state-of-the-art methods, detailed ablative analyses to understand each component's contribution, and sensitivity studies to characterize the method's robustness to variations in its hyperparameters;
- In Chapter 7, a synthesis of the thesis's main contributions is presented, limitations of the conducted work are discussed, and directions for future research are identified.

2 LNL: BACKGROUND AND STATE OF THE ART

This chapter presents the fundamental concepts related to LNL. We begin with a description of the most common sources of label noise in practical scenarios, followed by a formal problem definition and a taxonomy of the main types of noise. Subsequently, we review the principal techniques proposed in the literature to address this challenge, with particular emphasis on classification methods that form the foundation of this thesis. Finally, we outline the specific focus of our contributions within this landscape.

2.1 SOURCES OF LABEL NOISE

Noisy labels are incorrect, ambiguous, or inconsistent annotations that do not adequately reflect the true content of the data. While it is possible to mitigate part of this noise through better annotation and curation practices, it tends to be inevitable in many real-world scenarios, such as automatically collected data or subjective annotations. This noise affects the ability of models to learn correct patterns and can be interpreted as part of the irreducible error in the predictive error decomposition, although the classical Bias-Variance decomposition was not originally conceived to exclusively capture the effects of noisy labels (CARNEIRO, 2024).

In general, label noise can arise in different ways and have various origins. The literature identifies four main sources of this noise (FRENAY; VERLEYSEN, 2014):

- Information insufficiency: The information available in data samples may not be sufficient to ensure accurate and consistent labeling. This can result from the absence of relevant data, limitations in descriptive language, or variations in information quality. For example, in chest radiographs, multiple pathologies often exhibit similar visual characteristics, generating diagnostic ambiguities even for experienced radiologists (IRVIN et al., 2019);
- Inter-annotator variability: When the labeling task is subjective, significant variability may exist among annotators, regardless of their level of expertise. This variability arises from ambiguity in certain classification decisions, where multiple interpretations may be considered valid. For example, in diabetic retinopathy, Krause et al. (2018) reported relevant variation in the degree of agreement among ophthalmologists;
- Inconsistent annotators: Labeling errors can be introduced by factors such as fatigue, inattention, implicit biases, or inadequate qualification of annotators (PANDEY et al., 2022). These factors reduce the reliability of annotations and are particularly evident in crowdsourcing contexts, where supervision and quality control are limited (IBRAHIM et al., 2024);
- Encoding or communication errors: Systematic failures in data processing pipelines, inconsistencies in annotation interfaces, or corruption during transfer and storage. These errors occur independently of data content or annotator capability, manifesting as noise

introduced by technical limitations of collection and processing systems (LIANG; LIU; YAO, 2022).

The distribution of these mechanisms can vary across different application domains. In computer vision, label noise associated with inter-annotator variability is frequently observed, particularly in categories with less defined semantic boundaries. In natural language processing, characteristics such as polysemy and contextual dependency tend to contribute to multiple types of label noise simultaneously.

On the other hand, in clinical artificial intelligence applications, the required technical specialization can intensify both variability issues and informational limitations. For example, the ChestX-ray14 dataset exhibits considerable label noise rates, stemming not only from the limitations of natural language processing methods used to automatically extract labels from radiological reports, but also from the intrinsic difficulties of medical image diagnosis (OAKDEN-RAYNER, 2020).

2.2 PROBLEM STATEMENT

The LNL problem spans multiple supervised learning paradigms, including classification, regression, and structured prediction tasks such as semantic segmentation and object detection, each presenting unique characteristics in how label corruption manifests and affects model performance. This thesis focuses specifically on the *supervised classification setting*, which represents the most extensively studied and theoretically developed area within LNL research. The insights and methodologies developed for noisy classification often serve as foundational principles that can be adapted to other supervised learning domains.

2.2.1 Supervised classification setting

Consider a c-class classification problem defined over a feature space $\mathcal{X} \subset \mathbb{R}^d$, where d denotes the input dimensionality, and a label space $\mathcal{Y} = \{1, \ldots, c\}$, where c is the number of classes. Throughout this thesis, we adopt a one-hot vectorial representation for labels to simplify the description of algorithms and theoretical analysis under label noise. Specifically, $\mathbf{y} \in 0, 1^c$ represents the one-hot encoding of the scalar class index $y \in \mathcal{Y}$, i.e., $\mathbf{y} = \mathbf{e}_y$ where $\mathbf{e}_j$ is the j-th canonical basis vector.

In the standard supervised learning setting, the training dataset is $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{|\mathcal{D}|}$ sampled i.i.d. from an unknown joint distribution $p(\mathbf{x}, \mathbf{y})$ over $\mathcal{X} \times \{0, 1\}^c$, where $\mathbf{y}_i$ denotes the true one-hot label of instance $\mathbf{x}_i$.

In the LNL setting, the training dataset is $\mathcal{D} = \{(\mathbf{x}_i, \tilde{\mathbf{y}}_i)\}_{i=1}^{|\mathcal{D}|}$, where the observed noisy labels $\tilde{\mathbf{y}}_i$ may differ from the true labels $\mathbf{y}_i$ due to various noise sources. This corresponds to sampling from a corrupted joint distribution $p(\mathbf{x}, \tilde{\mathbf{y}})$ over $\mathcal{X} \times \{0, 1\}^c$. The noise corruption process is commonly modeled by a class-conditional transition matrix $\mathbf{T} \in [0, 1]^{c \times c}$, where each

entry $T_{i,j} = p(\tilde{\mathbf{y}} = \mathbf{e}_j \mid \mathbf{y} = \mathbf{e}_i)$ represents the probability of observing a noisy label j when the ground-truth class is i, subject to the constraint $\sum_{j=1}^{c} T_{i,j} = 1$ for all $i \in \{1, \ldots, c\}$.

Our objective is to learn a robust classifier $f(\mathbf{x}; \theta) : \mathcal{X} \to \Delta^{c-1}$ parameterized by $\theta \in \Theta$, where Θ denotes the parameter space and Δ^{c-1} is the probability simplex in $\mathbb{R}^c$, i.e., the set of probability vectors $\mathbf{p} = (p_1, \ldots, p_c)^\top$ with $p_k \geq 0$ and $\sum_{k=1}^{c} p_k = 1$. The classifier should minimize the expected risk with respect to the clean distribution while being trained on the noisy dataset:

$$\mathcal{R}_{\mathcal{D}}(f) = \mathbb{E}_{\mathcal{D}}[\ell(f(\mathbf{x}; \theta), \tilde{\mathbf{y}})] = \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x}_i, \tilde{\mathbf{y}}_i) \in \mathcal{D}} \ell(f(\mathbf{x}_i; \theta), \tilde{\mathbf{y}}_i) \tag{2.1}$$

where $\mathbb{E}_{\mathcal{D}}$ denotes the empirical expectation over the training set, and ℓ is a suitable loss function, such as cross-entropy (CE).

The fundamental challenge is to develop training strategies that are robust to the label noise sources identified previously, enabling the model to learn true underlying patterns despite training on corrupted labels. The specific types and mathematical characterizations of these noise processes are detailed in the following sections.

2.2.2 Evaluation metrics

A typical metric for assessing the robustness of a method in the LNL setting is the prediction accuracy on clean test examples, which are not seen during training. A significant drop in test accuracy often indicates that the model has overfit to noisy labels. Consequently, test accuracy is widely adopted as the standard criterion for final evaluation in the LNL literature.

For a clean test dataset $\mathcal{D}_{\text{test}} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{|\mathcal{D}_{\text{test}}|}$, let $\hat{\mathbf{y}}_i = \mathbf{e}_{\arg\max_k f(\mathbf{x}_i; \theta)_k}$ be the predicted one-hot label of the i-th example. The test accuracy is formalized as:

$$\text{Test Accuracy} = \frac{1}{|\mathcal{D}_{\text{test}}|} \sum_{(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{D}_{\text{test}}} \mathbf{y}_i^\top \hat{\mathbf{y}}_i \,. \tag{2.2}$$

For hyperparameter optimization and model selection, validation accuracy should ideally be used on a validation dataset $\mathcal{D}_{\text{val}} = \{(\mathbf{x}_i, \tilde{\mathbf{y}}_i)\}_{i=1}^{|\mathcal{D}_{\text{val}}|}$ (which may be noisy):

$$\text{Validation Accuracy} = \frac{1}{|\mathcal{D}_{\text{val}}|} \sum_{(\mathbf{x}_i, \tilde{\mathbf{y}}_i) \in \mathcal{D}_{\text{val}}} \tilde{\mathbf{y}}_i^\top \hat{\mathbf{y}}_i \,. \tag{2.3}$$

When a clean test set is not available, validation accuracy may be used as a proxy for final evaluation. Although it may be affected by label noise, it still serves as a practical estimate of the model's performance, especially in real-world scenarios where clean annotations are difficult to obtain.

It is worth noting that, depending on the method, additional metrics such as label precision, label recall, and error correction rate may also be considered (SONG et al., 2023).

2.3 TYPES OF LABEL NOISE

2.3.1 Instance-independent noise

Instance-independent label noise assumes that the corruption process is conditionally independent of the data features given the true label (SONG et al., 2023). For the subsequent definitions, let $i \in \{1, \ldots, c\}$ denote the class index and $\tau \in [0, 1]$ the true label noise rate.

2.3.1.1 Symmetric

Symmetric noise, also known as uniform noise, occurs when the labels of samples are corrupted uniformly across all classes. Formally, the transition probabilities are

$$T_{i,i} = 1 - \tau, \quad T_{i,j} = \frac{\tau}{c-1} \quad \text{for } j \neq i. \tag{2.4}$$

which corresponds to the vectorial formulation:

$$p(\tilde{\mathbf{y}} = \mathbf{e}_i \mid \mathbf{y} = \mathbf{e}_i) = 1 - \tau, \quad p(\tilde{\mathbf{y}} = \mathbf{e}_j \mid \mathbf{y} = \mathbf{e}_i) = \frac{\tau}{c-1} \quad \text{for } j \neq i\,. \tag{2.5}$$

This implies that each true label has a uniform probability of $\frac{\tau}{c-1}$ of being corrupted to any of the other $c-1$ classes. Such noise commonly arises in automatic labeling, random annotation errors, or communication failures (LIANG; LIU; YAO, 2022).

2.3.1.2 Asymmetric

Asymmetric noise characterizes class-dependent label corruption, often arising from inherent visual or semantic similarities between specific classes, commonly observed in real-world datasets. Formally,

$$T_{i,i} = 1 - \sum_{k \neq i} T_{i,k}, \quad \text{with some } T_{i,j} > T_{i,k} \text{ for } j \neq k \neq i\,. \tag{2.6}$$

where some off-diagonal entries $T_{i,j}$ are larger than others, reflecting that certain classes are more prone to confusion. For instance, $T_{\text{dog,cat}} > T_{\text{dog,airplane}}$ indicates that a sample with true label “dog” is more likely to be mislabeled as “cat” than as “airplane”, due to semantic similarity.

2.3.1.2.1 Pair noise

A specific case of asymmetric noise is pair noise, where the label of one class can only be mapped to another visually similar class. In this case, only one specific transition is allowed:

$$T_{i,i} = 1 - \tau, \quad T_{i,j} = \tau, \quad T_{i,k} = 0 \text{ for } k \neq i, j\,. \tag{2.7}$$

which in vectorial form becomes:

$$p(\tilde{\mathbf{y}} = \mathbf{e}_i \mid \mathbf{y} = \mathbf{e}_i) = 1 - \tau, \quad p(\tilde{\mathbf{y}} = \mathbf{e}_j \mid \mathbf{y} = \mathbf{e}_i) = \tau, \quad p(\tilde{\mathbf{y}} = \mathbf{e}_k \mid \mathbf{y} = \mathbf{e}_i) = 0 \text{ for } k \neq i, j\,. \tag{2.8}$$

2.3.1.2.2 Adjacent noise

Another variant is adjacent noise, which models cyclic transitions between consecutive classes, such that each class i can only be mislabeled as the next class in the sequence:

$$T_{i,i} = 1 - \tau, \quad T_{i,(i \bmod c)+1} = \tau, \quad T_{i,k} = 0 \text{ otherwise}. \tag{2.9}$$

2.3.2 Instance-dependent noise

Instance-dependent noise models a more realistic scenario, where the corruption probability depends on both the true label and the specific characteristics of each input instance. Unlike instance-independent noise, this type accounts for the fact that some samples are intrinsically harder to label, reflecting the varying complexity of instances in the dataset. Formally, the corruption probability is defined as:

$$\rho_{i,j}(\mathbf{x}) = p(\tilde{\mathbf{y}} = \mathbf{e}_j \mid \mathbf{y} = \mathbf{e}_i, \mathbf{x}) \tag{2.10}$$

where $\mathbf{x}$ represents the instance features, making the transition matrix dependent on each individual sample.

This type of noise is particularly relevant in practical applications, where factors such as image quality, visual ambiguity, class overlap, or annotation limitations can influence labeling errors. For example, images with low resolution, occlusions, or poor lighting conditions are more prone to mislabeling. Instance-dependent noise captures this variability, providing a more faithful model of real-world noisy label scenarios (SONG et al., 2023).

2.3.3 Extensions to other learning domains

Although this thesis focuses on supervised classification tasks, the concept of label noise naturally extends to other ML domains, such as regression, segmentation, and detection, each presenting specific challenges related to the nature of labels and how label noise manifests.

In regression tasks, label noise does not manifest as incorrect classes, but rather as distortions in the continuous values of target variables, requiring specific modeling approaches and robust strategies. In semantic segmentation, for example, the challenge lies in pixel-level classification, where label noise can follow structural patterns (e.g., affecting contiguous image regions due to systematic annotation errors). In object detection, which combines aspects of classification and regression, noise can simultaneously affect class labels and bounding box coordinates. This scenario introduces additional complexity, especially in cases of partially labeled objects or poorly positioned boxes.

For in-depth discussions on mathematical formalizations and specific implementations of these domains, we refer to Yi et al. (2022), Liu et al. (2022) and Carneiro (2024).

2.4 STATE OF THE ART

The challenge of LNL has motivated extensive research, resulting in a rich body of methodological approaches. Several surveys have organized these techniques from different perspectives, each providing valuable insights into the field. Han et al. (2021) presents a framework centered on learning from noisy label representations, while Song et al. (2023) offers a detailed categorization emphasizing DL applications.

In this thesis, we structure our analysis following the organizational framework developed by Carneiro (2024), which systematically organizes LNL methodologies into complementary perspectives. This framework, illustrated in Figure 1, divides approaches into four main categories: loss function, training algorithms, data processing, and model architectures. Each category addresses the noisy label challenge through distinct mechanisms, collectively covering the primary intervention points in the learning pipeline.

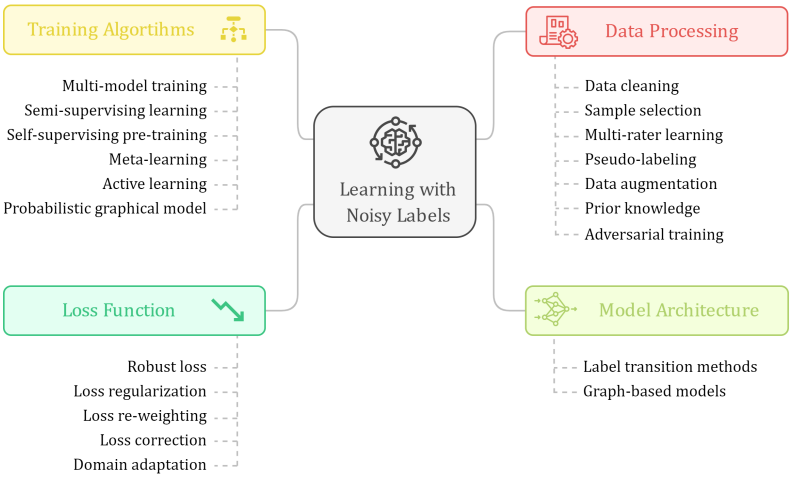


Figure 1 – Organizational framework of LNL methodologies.

Source: Adapted from Carneiro (2024).

Although the framework encompasses various strategies, not all subcategories have received equal attention or consolidation in recent DL research on noisy labels. Some approaches are more promising in realistic scenarios, while others remain context-limited or face scalability issues. Techniques such as active learning require reliable oracles for relabeling, often unfeasible at scale. Probabilistic graphical models, despite recent advances, involve challenges like variational inference and training of generative models, typically more expensive and less stable than conventional discriminative approaches.

Other methodological approaches present solid theoretical foundations but encounter similar practical barriers. Multi-rater learning, for example, offers formal guarantees regarding the identifiability of true labels; however, its adoption is limited by the high cost of obtaining multiple annotations per sample. Similarly, techniques based on synthetic label generation have not yet proven to be sufficiently reliable alternatives to human annotation, and the generalization of uncertainty estimates obtained through multiple annotators to single-annotation

scenarios remains little explored.

Additionally, strategies such as domain adaptation, prior knowledge incorporation, and adversarial training have been investigated. Domain adaptation assumes training (noisy) and test (clean) data belong to distinct domains, but requires prior access to test data, generally unavailable in benchmarks. Informative priors provide benefits but need specialized knowledge and face generalization challenges. Adversarial training proposes connections between label noise and input noise, but remains poorly validated empirically in LNL.

Given these considerations, this thesis focuses on the most established and widely adopted approaches in the LNL literature. These techniques constitute the core of the current state of the art, reconciling theoretical foundations, practical applicability, and robust empirical performance. In the following sections, each category is examined in detail, discussing its foundations and characteristics.

2.4.1 Loss function

In supervised learning, model training relies on empirical risk minimization:

$$\theta^* = \arg\min_{\theta \in \Theta} \mathbb{E}_{\mathcal{D}} \left[\ell \left(f(\mathbf{x}; \theta), \tilde{\mathbf{y}} \right) \right] \tag{2.11}$$

In classification, the most common loss functions are CE for multi-class problems and binary cross-entropy (BCE) for binary classification. However, these conventional functions pose challenges for LNL tasks due to their symmetric treatment of labels, exponentially penalizing deviations regardless of their reliability (HAN et al., 2021; CARNEIRO, 2024). This leads to memorization of noisy labels, especially in DNNs.

To address this issue, various strategies have been proposed for modifying loss functions, including robust formulations, regularization techniques, re-weighting schemes, correction methods, and domain adaptation approaches.

2.4.1.1 Robust loss

Robust loss functions are specifically designed to be resilient to label noise by embedding noise tolerance into their formulation. They mitigate the influence of corrupted samples without relying on prior estimates of noise rate or type, making them effective in settings where noise characteristics are unknown or hard to estimate.

One of the first theoretical studies on loss robustness to label noise was conducted by Manwani & Sastry (2013), defining noise tolerance as a learning algorithm's ability to produce the same optimal classifier on clean or noisy labels. They showed that 0–1 loss is fully robust, while Mean Squared Error (MSE) is tolerant only under symmetric noise. Ghosh, Kumar & Sastry (2017) extended this by identifying conditions where the expected risk under noisy labels matches that of clean labels, proving that Mean Absolute Error (MAE) satisfies these conditions, whereas CE remains sensitive to noise.

Addressing theoretical limitations identified by Long & Servedio (2008) regarding the vulnerability of convex loss functions to random classification noise in linear classifiers, Rooyen, Menon & Williamson (2015) proposed the unhinged loss:

$$\ell_{\text{unh}}\left(f(\mathbf{x};\theta),\tilde{\mathbf{y}}\right) = 1 - \tilde{\mathbf{y}}^{\top} f(\mathbf{x};\theta) \tag{2.12}$$

The key innovation lies in its symmetric property ($\ell(z) + \ell(-z) = \text{constant}$), which theoretically ensures robustness to label noise. This insight was subsequently formalized by Charoenphakdee, Lee & Sugiyama (2019), who proved that symmetry constitutes a sufficient condition for loss function robustness under specific noise assumptions.

While theoretically robust functions like MAE provide noise tolerance guarantees, they face practical limitations including slow convergence and degraded performance on complex multi-class tasks. To address this fundamental robustness-efficiency trade-off, Zhang & Sabuncu (2018) developed Generalized Cross-Entropy (GCE):

$$\ell_{\text{gce}}\left(f(\mathbf{x};\theta),\tilde{\mathbf{y}}\right) = \frac{1 - \left(\tilde{\mathbf{y}}^{\top} f(\mathbf{x};\theta)\right)^{q}}{q}, \quad q \in (0,1] \tag{2.13}$$

This parametric family allows tuning the trade-off between robustness and convergence: low q values retain CE's fast convergence, while $q = 1$ approaches MAE's robustness, adapting the loss to noise level and task complexity.

Drawing inspiration from the symmetry properties of Kullback-Leibler (KL) divergence, Wang et al. (2019) proposed Symmetric Cross-Entropy (SCE), which combines standard CE with its reverse formulation to enhance noise tolerance:

$$\ell_{\text{sce}}\left(f(\mathbf{x};\theta),\tilde{\mathbf{y}}\right) = -\alpha\left(\tilde{\mathbf{y}}^{\top}\log\left(f(\mathbf{x};\theta)\right)\right) - \beta\left(f(\mathbf{x};\theta)^{\top}\log\left(\tilde{\mathbf{y}}\right)\right) \tag{2.14}$$

where the first term (weighted by α) preserves CE's optimization properties, while the second term (weighted by β) adds robustness via the reverse cross-entropy (RCE) formulation.

Building on the observation that normalization can enhance loss robustness to label noise, Ma et al. (2020) proposed the Active Passive Loss (APL), which mitigates underfitting in normalized losses by combining "active" terms that favor correct class probabilities with "passive" terms that suppress incorrect ones. Similarly, Xu et al. (2019) introduced a Determinant-based Mutual Information (DMI) loss, preserving key monotonicity properties and providing robustness guarantees under arbitrary noise.

Recent research has explored abstention learning for highly ambiguous samples, allowing models to refrain from uncertain predictions. Thulasidasan et al. (2019) proposed the Deep Abstention Classifier (DAC), adding an abstention class ($c + 1$) controlled by a hyperparameter balancing prediction and abstention. Sachdeva et al. (2021) showed this mechanism is especially useful in open-set noise scenarios, where unseen class samples are mislabeled as known ones.

The evolution of robust loss functions reveals a clear progression: initial approaches emphasized theoretical guarantees (MAE, unhinged), later developments improved practical

viability (GCE, SCE), and specialized solutions address specific scenarios (DAC for open-set noise). This highlights a trade-off between theoretical guarantees and computational efficiency: MAE is robust but converges slowly, while hybrid approaches like GCE and SCE sacrifice some guarantees for better optimization.

Robust loss functions are most effective in low-noise settings and tasks with few classes. In more challenging scenarios, such as high noise or large-scale datasets, their performance improves when combined with sample selection or label correction. However, most robust losses address the effects of label noise rather than modeling its underlying generative process.

2.4.1.2 Loss regularization

While robust loss functions address label noise by modifying the loss function's intrinsic properties, regularization techniques take a complementary approach by constraining the learning process itself. Loss regularization methods impose additional constraints on either model parameters or training dynamics, limiting the model's capacity to memorize noisy patterns without requiring prior knowledge of noise characteristics.

The fundamental insight underlying regularization-based methods stems from the observation that DNNs exhibit a distinctive learning behavior: they tend to fit clean patterns before memorizing noisy labels (ARPIT et al., 2017). This phenomenon suggests that controlling the learning trajectory can naturally separate signal from noise. The general regularized formulation extends empirical risk minimization as:

$$\theta^* = \arg\min_{\theta \in \Theta} \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x}, \tilde{\mathbf{y}}) \in \mathcal{D}} \ell\left(f(\mathbf{x}; \theta), \tilde{\mathbf{y}}\right) + \Omega(\theta) \tag{2.15}$$

where $\Omega(\theta)$ is the regularization term, defined according to the specific constraint.

Classical regularization constrains model parameters to prevent overfitting. Architecture-based methods like dropout (JINDAL; NOKLEBY; CHEN, 2016) provide implicit regularization by randomly deactivating neurons, reducing co-adaptation and improving generalization under noise. Nested dropout (CHEN et al., 2024) further regularizes feature activations, offering structured robustness in noisy environments.

Parameter constraint methods explicitly limit how much model weights can deviate during training. Hu, Li & Yu (2020) formalized this approach by constraining parameter deviation from the initialization, where $\theta(0)$ denotes the model initialization:

$$\theta^* = \arg\min_{\theta \in \Theta} \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x}, \tilde{\mathbf{y}}) \in \mathcal{D}} \ell\left(f(\mathbf{x}; \theta), \tilde{\mathbf{y}}\right) + \lambda \|\theta - \theta(0)\|_2^2 \tag{2.16}$$

This approach prevents the model from straying far from its random initialization, assuming that clean patterns require smaller parameter changes than noise memorization.

The most sophisticated regularization approaches dynamically adjust the training process based on learning dynamics. Liu et al. (2020) developed Early-Learning Regularization

(ELR), which modulates gradient flow according to sample reliability estimates:

$$\theta^* = \arg\min_{\theta \in \Theta} \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x}_i, \tilde{\mathbf{y}}_i) \in \mathcal{D}} \ell(f(\mathbf{x}_i; \theta), \tilde{\mathbf{y}}_i) + \lambda \log(1 - f(\mathbf{x}_i; \theta)^\top \mathbf{t}_i) \tag{2.17}$$

where $\mathbf{t}_i$ represents target distributions maintained through exponential moving averages. This mechanism amplifies learning from samples that consistently align with the model's predictions while dampening the influence of inconsistent (potentially noisy) examples.

Moving beyond methods based solely on sample dynamics, Tan et al. (2021) proposed Co-learning, which combines distinct learning paradigms via regularization. The framework employs a shared encoder with two heads—one for supervised classification and another for contrastive learning—guided by two auxiliary losses: an intrinsic similarity loss enforcing consistency across sample transformations, and a structural similarity loss aligning representations of both tasks via KL divergence. This design allows feature-dependent information to implicitly regularize supervised learning, enhancing robustness without estimating noise rates.

Regularization techniques are effective across diverse scenarios, with performance varying by the underlying noise mechanism. Methods like dropout provide consistent improvements, while approaches that penalize deviations from initial weights offer strong resistance to early-stage label noise. Dynamic methods such as ELR excel when sample reliability is estimable over time, though they require careful parameter calibration.

Recent comparative studies highlight that regularization effectiveness depends on noise patterns: symmetric noise responds well to classical methods, whereas instance-dependent noise benefits from adaptive approaches. This has motivated hybrid architectures combining multiple regularization mechanisms, indicating that the future of regularization in LNL lies in principled combinations that exploit complementary strengths.

2.4.1.3 Loss reweighting

Loss reweighting methods address a fundamental challenge in LNL: how to automatically distinguish between reliable and unreliable samples without access to clean labels. Rather than treating all training examples equally, these approaches assign differential importance weights based on estimated sample reliability, effectively implementing a soft filtering mechanism during optimization.

The core challenge lies in developing principled weighting functions that can reliably estimate sample quality from observable features. The reweighted optimization problem modifies empirical risk minimization as:

$$\theta^* = \arg\min_{\theta \in \Theta} \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x}, \tilde{\mathbf{y}}) \in \mathcal{D}} \omega(\mathbf{x}, \tilde{\mathbf{y}}) \times \ell(f(\mathbf{x}; \theta), \tilde{\mathbf{y}}) \tag{2.18}$$

where $\omega(\mathbf{x}, \tilde{\mathbf{y}})$ represents the weight assigned to each sample based on its estimated reliability.

Liu & Tao (2016) formulated LNL as a domain shift problem, where the source domain contains noisy labels and the target domain has clean labels. Their approach employs

importance reweighting to correct for the distribution mismatch:

$$\omega(\mathbf{x}, \tilde{\mathbf{y}}) = \frac{p(Y = \tilde{\mathbf{y}} \mid X = \mathbf{x})}{p(\tilde{Y} = \tilde{\mathbf{y}} \mid X = \mathbf{x})} \tag{2.19}$$

This theoretical framework provides a principled foundation but requires estimating the noise transition probabilities, which can be challenging in practice.

An alternative strategy, Active Bias (CHANG; LEARNED-MILLER; MCCALLUM, 2017), uses prediction variance as an indicator of sample reliability. The central idea is that samples of intermediate difficulty exhibit higher variance across training iterations:

$$\omega(\mathbf{x}, \tilde{\mathbf{y}}) = \frac{s\,(\mathbf{x}, \tilde{\mathbf{y}}) + \epsilon}{Z} \tag{2.20}$$

where $s(\mathbf{x}, \tilde{\mathbf{y}})$ measures prediction variability over recent iterations for the specific sample-label pair, Z provides normalization across the dataset, and ϵ ensures numerical stability.

A particularly influential approach by Arazo et al. (2019) models the loss distribution using a two-component Beta mixture model (BMM):

$$p(\ell_i) = \sum_{k \in \{\text{noisy}, \text{clean}\}} \lambda_k \times p(\ell_i \mid k) \tag{2.21}$$

where each component $p(\ell_i \mid k)$ is modeled by a Beta distribution:

$$p(\ell_i \mid k) = \frac{\Gamma(\alpha_k + \beta_k)}{\Gamma(\alpha_k)\Gamma(\beta_k)} \ell_i^{\alpha_k - 1} (1 - \ell_i)^{\beta_k - 1} \tag{2.22}$$

with parameters $\alpha_k, \beta_k > 0$ and $\Gamma(\cdot)$ representing the Gamma function. The sample weight is computed from the posterior probability of belonging to the clean component:

$$w_i = \frac{\lambda_{k=\text{clean}} \times p(\ell_i \mid k = \text{clean})}{\sum_{j \in \{\text{noisy, clean}\}} \lambda_j \times p(\ell_i \mid j)} \tag{2.23}$$

Rather than direct reweighting, this method applies weights via label smoothing, interpolating between the noisy label and the model's prediction based on sample reliability.

Wang et al. (2023) proposed Improved MAE (IMAE), which emphasizes samples with intermediate confidence levels while down-weighting both very easy and very hard examples:

$$\omega(\mathbf{x}, \tilde{\mathbf{y}}) = \exp\left(\nu \times (\tilde{\mathbf{y}}^\top f(\mathbf{x}; \theta)) \times \left(1 - \tilde{\mathbf{y}}^\top f(\mathbf{x}; \theta)\right)\right) \tag{2.24}$$

where $\nu > 0$ is a constant that controls the exponential base. This exponential weighting scheme automatically focuses learning on samples where the model exhibits moderate uncertainty.

While reweighting methods offer flexibility and theoretical appeal, they face limitations. The small-loss hypothesis, assuming low training loss indicates clean labels, often fails under class-dependent noise or limited model capacity, and most weighting functions require careful hyperparameter tuning. The shift from domain adaptation theory to adaptive confidence weighting reflects a move toward more practical, automated solutions, with evidence suggesting that integrating multiple reliability indicators is more effective than relying on single heuristics.

2.4.1.4 Loss correction

Loss correction methods aim to mitigate label noise by adjusting the supervision signal during training. Instead of treating observed labels as ground truth, they refine learning targets by interpolating model predictions with noisy labels or modeling the underlying corruption process. The key insight is that models can identify reliable versus corrupted signals, generating better targets than the original labels. This self-correction arises because DNNs tend to fit clean patterns before memorizing noise, allowing intervention during training.

One approach that operationalizes this idea is bootstrapping. Reed et al. (2015) introduced a method where model predictions are progressively incorporated into the supervision signal based on their agreement with the provided labels:

$$\theta^* = \arg\min_{\theta \in \Theta} \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x}, \tilde{\mathbf{y}}) \in \mathcal{D}} \ell\left(f(\mathbf{x};\theta), \beta\tilde{\mathbf{y}} + (1-\beta) f(\mathbf{x};\theta)\right) \tag{2.25}$$

where $\beta \in [0,1]$ balances the contribution of the noisy label $\tilde{\mathbf{y}}$ and the model prediction $f(\mathbf{x};\theta)$. The loss reduces to CE when both agree and shifts toward self-training otherwise.

A more sophisticated approach explicitly models the noise mechanism through transition matrices. Patrini et al. (2017) developed forward and backward correction strategies that account for systematic label corruption patterns. The backward approach applies the inverse transition matrix to adjust the loss computation:

$$\theta^* = \arg\min_{\theta \in \Theta} \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x}, \tilde{\mathbf{y}}) \in \mathcal{D}} \tilde{\mathbf{y}}^\top \hat{\mathbf{T}}^{-1} \left[\ell\left(f(\mathbf{x};\theta), \mathbf{y}_1\right), \cdots, \ell\left(f(\mathbf{x};\theta), \mathbf{y}_{|\mathcal{Y}|}\right)\right]^\top \tag{2.26}$$

while the forward variant modifies model predictions directly:

$$\theta^* = \arg\min_{\theta \in \Theta} \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x}, \tilde{\mathbf{y}}) \in \mathcal{D}} \ell\left(\hat{\mathbf{T}}^\top f(\mathbf{x};\theta), \tilde{\mathbf{y}}\right) \tag{2.27}$$

The success of these methods relies on accurately estimating the transition matrix $\hat{\mathbf{T}}$, a task that becomes increasingly difficult under high noise rates or in datasets with many classes.

To address this difficulty, Hendrycks et al. (2018) proposed the Gold Loss Correction (GLC) method, which uses a small clean validation set to estimate transition probabilities:

$$\hat{\mathbf{T}}_{\mathbf{y},\tilde{\mathbf{y}}} = \frac{1}{|\bar{\mathcal{D}}_{\mathbf{y}}|} \sum_{\mathbf{x} \in \bar{\mathcal{D}}_{\mathbf{y}}} p_\theta(\tilde{Y} = \tilde{\mathbf{y}} \mid X = \mathbf{x}) \tag{2.28}$$

where $\bar{\mathcal{D}}_{\mathbf{y}}$ denotes the set of clean samples with true label $\mathbf{y} \in \mathcal{Y}$, and p_θ is trained using the original noisy-label training set. This method relaxes the assumption of having no clean data by trading it for the requirement of a small, curated subset.

Complementing explicit matrix estimation, Lukasik et al. (2020) proposed label smoothing techniques that regularize labels by spreading probability mass across classes, reducing sensitivity to individual label errors while preserving class structure.

Despite their potential, loss correction methods face challenges. Bootstrapping requires careful tuning of the interpolation parameter β, as high values retain noise and low values risk premature convergence to errors. Matrix-based approaches rely on stable noise patterns captured by transition matrices and clean validation data, assumptions often unmet in practice. This progression motivates adaptive correction strategies that adjust to estimated noise, enhancing robustness across diverse scenarios.

2.4.2 Training algorithms

The development of training algorithms is among the most active areas in LNL research. Strategies include meta-learning methods that adjust sample weights or revise labels, self-supervised pretraining for more robust representations, and multi-network techniques to mitigate confirmation bias. Semi-supervised learning (SSL) treats part of the dataset as unlabeled, while probabilistic graphical models provide a framework to capture the label noise process. The following subsections discuss the main methods in this direction.

2.4.2.1 Meta-learning

Meta-learning in LNL aims to automatically distinguish reliable from noisy supervision during training. Unlike methods assuming prior noise knowledge, it adaptively reweights or relabels samples through hierarchical optimization guided by reliable validation signals. The framework comprises three elements: parameterization of the weighting/relabeling function, bilevel optimization linking meta-parameters to validation performance, and assumptions on reliable validation data availability. Different designs yield trade-offs in expressiveness, computational cost, and clean supervision reliance.

One of the earliest meta-learning methods for LNL was proposed by Dehghani et al. (2017), introducing an architecture that separates confidence estimation from classification. The method employs a feature extractor $g(\mathbf{x};\phi) : \mathcal{X} \to \mathcal{Z}$, classifier $f(\mathbf{z};\theta) : \mathcal{Z} \to \Delta^{|\mathcal{Y}|-1}$, and confidence estimator $\omega(\mathbf{z}, \tilde{\mathbf{y}};\gamma) : \mathcal{Z} \times [0,1]^{|\mathcal{Y}|} \to \mathbb{R}$ for label reliability. The approach uses a noisy training set $\mathcal{D}_t$ and clean validation set $\mathcal{D}_v$. Training proceeds in two stages, first training the confidence network:

$$\phi^*, \gamma^* = \arg\min_{\phi,\gamma} \frac{1}{|\mathcal{D}_v|} \sum_{(\mathbf{x},\tilde{\mathbf{y}},\mathbf{y}) \in \mathcal{D}_v} \ell\left(\omega(g(\mathbf{x};\phi), \tilde{\mathbf{y}};\gamma), \mathrm{diff}(\mathbf{y}, \tilde{\mathbf{y}})\right) \tag{2.29}$$

where $\mathrm{diff}(\mathbf{y}, \tilde{\mathbf{y}})$ quantifies the discrepancy between clean and noisy labels. In the second stage, the classifier is trained using confidence-based weighting:

$$\phi^*, \theta^* = \arg\min_{\phi,\theta} \frac{1}{|\mathcal{D}_t|} \sum_{(\mathbf{x},\tilde{\mathbf{y}}) \in \mathcal{D}_t} \omega(g(\mathbf{x};\phi), \tilde{\mathbf{y}};\gamma^*) \times \ell(f(g(\mathbf{x};\phi);\theta), \tilde{\mathbf{y}}) \tag{2.30}$$

While intuitive, this approach requires explicit supervision of label quality. To overcome this, the Learning to Reweight (L2R) method by Ren et al. (2018) introduces per-sample

meta-weights, avoiding direct modeling of confidence. This bilevel problem learns weights $\mathbf{w} \in \Delta^{|\mathcal{D}_t|-1}$ to minimize validation loss:

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \frac{1}{|\mathcal{D}_v|} \sum_{(\mathbf{x}_j, \mathbf{y}_j) \in \mathcal{D}_v} \ell(f(\mathbf{x}_j; \theta^*(\mathbf{w})), \mathbf{y}_j) \tag{2.31}$$

where $\theta^*(\mathbf{w})$ minimizes weighted training loss on noisy data. Despite its flexibility, L2R scales poorly due to numerous meta-parameters.

To address scalability, Shu et al. (2019) proposed Meta-Weight-Net (MWN), which models the weighting function $\omega(\cdot;\phi)$ as a neural network that maps loss values to adaptive sample weights. The method is formulated as a bilevel optimization problem:

$$\begin{aligned} \phi^* &= \arg\min_{\phi} \frac{1}{|\mathcal{D}_v|} \sum_{(\mathbf{x},\mathbf{y}) \in \mathcal{D}_v} \ell(f(\mathbf{x}; \theta_\phi^*), \mathbf{y}) \\ \text{subject to} \quad \theta_\phi^* &= \arg\min_{\theta} \frac{1}{|\mathcal{D}_t|} \sum_{(\mathbf{x},\tilde{\mathbf{y}}) \in \mathcal{D}_t} \omega(\ell(f(\mathbf{x};\theta), \tilde{\mathbf{y}}); \phi) \times \ell(f(\mathbf{x};\theta), \tilde{\mathbf{y}}) \end{aligned} \tag{2.32}$$

As a natural progression from pure weighting, Zhang et al. (2020) introduced a meta-parameter $\beta_i \in [0, 1]$ to convexly combine noisy labels and model predictions:

$$\hat{\mathbf{y}}_i(\beta_i) = \beta_i \tilde{\mathbf{y}}_i + (1 - \beta_i) f(\mathbf{x}_i; \theta), \tag{2.33}$$

allowing more flexible corrections beyond traditional weighting schemes.

Going beyond weighting-based adjustments, Zheng, Awadallah & Dumais (2021) introduced Meta Label Correction (MLC), where a meta-model trained on clean validation data learns to correct labels for the main classifier via bilevel optimization. In parallel, teacher-student frameworks are incorporated into meta-learning. For instance, Li et al. (2017) used knowledge distillation, with an expert model generating soft targets and a meta-learner adaptively fusing them with noisy supervision. This setup leverages pre-trained model biases while dynamically adjusting trust in expert guidance.

A recurring challenge in meta-learning is the reliance on clean validation data. To alleviate this, Zhang & Pfister (2021) proposed an automatic construction method based on training instability, selecting samples that exhibit high loss variation across consecutive epochs. Complementarily, Xu et al. (2021) leveraged Gaussian mixture models (GMMs) to detect pseudo-clean samples by modeling the loss distribution, exploiting the small-loss criterion.

Addressing the dependency on clean validation data, Xia, Lee & Chen (2023) proposed TCC-Net, which generates meta-data automatically during training. The method combines Co-Teaching with meta-learning in two stages: first, pre-training networks with a Contradictory Loss; then applying meta-co-teaching, dividing each mini-batch into training and meta portions based on loss ranking. Each network learns to weight samples from its peer via parameterized MLPs trained through bilevel optimization, creating a self-supervised meta-learning approach.

Despite recent advances, meta-learning approaches face notable limitations. They are often computationally expensive and sensitive to meta-parameter tuning. Importantly, most

methods conflate high-loss samples with noise, overlooking the distinction between aleatoric uncertainty (inherent ambiguity) and epistemic uncertainty (hard but informative cases). This is critical under instance-dependent noise, where sample difficulty may correlate with label corruption, biasing the exclusion of valuable data.

2.4.2.2 Multi-model training

Multi-model training is a core strategy in LNL, particularly effective in mitigating confirmation bias, a phenomenon where the model reinforces incorrect predictions by overly trusting noisy labels. The underlying premise is that independently trained models are unlikely to make the same mistakes on the same samples, especially in the early stages of training. This disagreement can be exploited as a reliability signal, allowing one model to act as a filter for the other during sample selection. In practice, such complementarity functions as an implicit form of cross-validation, enhancing robustness against overfitting to noise.

One key approach in this category leverages prediction divergence as a selection criterion. A straightforward example is the Decouple method (MALACH; SHALEV-SHWARTZ, 2017), which trains two DNNs exclusively on samples where their predictions disagree:

$$\mathcal{S} = \{(\mathbf{x}, \tilde{\mathbf{y}}) \in \mathcal{D} \mid \arg\max f(\mathbf{x}; \theta_1) \neq \arg\max f(\mathbf{x}; \theta_2)\} \tag{2.34}$$

where both models are jointly optimized on the disagreement set $\mathcal{S}$. While conceptually elegant, this strategy may include noisy samples in the disagreement set, limiting its effectiveness under high-noise conditions.

A second category establishes hierarchical relationships between models, where a specialized mentor model guides the training of the main student model. MentorNet (JIANG et al., 2018) exemplifies this approach by implementing a data-driven curriculum learning strategy, where a mentor network $g(\cdot; \phi)$ dynamically assigns weights to training samples:

$$\phi^*, \theta^* = \arg\min_{\phi,\theta} \frac{1}{|\mathcal{D}|} \sum_{i=1}^{|\mathcal{D}|} g(\mathbf{z}_i; \phi) \times \ell(f(\mathbf{x}_i; \theta), \tilde{\mathbf{y}}_i) + R(\phi, \theta), \tag{2.35}$$

where $\mathbf{z}_i$ represents features extracted from the sample i and $R(\phi, \theta)$ denotes regularization terms that encourage progressive sample selection and prevent overfitting.

A widely used strategy involves training multiple models that teach each other by leveraging the small-loss assumption. The Co-Teaching method (HAN et al., 2018) embodies this approach by having each model select a subset of samples with the smallest losses to train its peer:

$$\theta_1^* = \arg\min_{\theta_1} \frac{1}{|\mathcal{S}_2|} \sum_{(\mathbf{x},\tilde{\mathbf{y}}) \in \mathcal{S}_2} \ell(f(\mathbf{x}; \theta_1), \tilde{\mathbf{y}}), \quad \theta_2^* = \arg\min_{\theta_2} \frac{1}{|\mathcal{S}_1|} \sum_{(\mathbf{x},\tilde{\mathbf{y}}) \in \mathcal{S}_1} \ell(f(\mathbf{x}; \theta_2), \tilde{\mathbf{y}}) \tag{2.36}$$

where $\mathcal{S}_1$ and $\mathcal{S}_2$ denote the subsets of samples with the lowest losses, selected by models 1 and 2, respectively. Specifically, they are defined as $\mathcal{S}_1 = \{(\mathbf{x}, \tilde{\mathbf{y}}) \in \mathcal{D} \mid \ell(f(\mathbf{x}; \theta_1), \tilde{\mathbf{y}}) < \tau_{\text{loss}}\}$ and

$\mathcal{S}_2 = \{(\mathbf{x}, \tilde{\mathbf{y}}) \in \mathcal{D} \mid \ell(f(\mathbf{x};\theta_2), \tilde{\mathbf{y}}) < \tau_{\text{loss}}\}$. The threshold τ_{loss} is progressively annealed during training, allowing for increasingly strict filtering of potentially noisy samples over time.

To address premature convergence, where models become too similar too quickly, Co-Teaching+ (YU et al., 2019) extends the original Co-Teaching framework by using a disagreement criterion. It first filters samples based on prediction disagreement, then applies the small-loss criterion within this set, preserving model diversity longer and improving robustness. A more advanced alternative trains multiple models simultaneously with objectives balancing supervised loss and regularization. JoCoR (WEI et al., 2020) exemplifies this via joint minimization:

$$\theta_1^*, \theta_2^* = \arg\min_{\theta_1,\theta_2} \frac{1}{|\mathcal{S}|} \sum_{(\mathbf{x},\tilde{\mathbf{y}}) \in \mathcal{S}} \mathcal{L}(\mathbf{x}, \tilde{\mathbf{y}}, \theta_1, \theta_2) \tag{2.37}$$

where the subset $\mathcal{S}$ is selected based on joint loss thresholding $\mathcal{S} = \{(\mathbf{x}, \tilde{\mathbf{y}}) \in \mathcal{D} \mid \mathcal{L}(\mathbf{x}, \tilde{\mathbf{y}}, \theta_1, \theta_2) < \tau_{\text{loss}}\}$, and the loss function combines supervised terms with regularization based on the KL divergence:

$$\mathcal{L}(\mathbf{x}, \tilde{\mathbf{y}}, \theta_1, \theta_2) = (1-\lambda) \sum_{i=1}^{2} \ell(f(\mathbf{x};\theta_i), \tilde{\mathbf{y}}) + \lambda \text{KL}_{\text{sym}}[f(\mathbf{x};\theta_1) \| f(\mathbf{x};\theta_2)] \tag{2.38}$$

This formulation enforces stricter constraints on sample selection, requiring low loss in both models simultaneously for inclusion in the training set.

Expanding Co-teaching strategies, Zhang et al. (2021) proposed CJC-Net, which integrates cyclical training with a joint loss function in a three-stage architecture. The method combines simultaneous pre-training of two networks, followed by a noise elimination phase where learning rates are cyclically adjusted to alternate between overfitting and underfitting. During this phase, a joint loss function combining CE with regularization is employed to amplify the differences between clean and noisy sample losses. The cyclical training allows accumulating more robust loss statistics over time, while the Co-Teaching strategy prevents both networks from prematurely converging to the same errors. Finally, samples with high accumulated losses are removed, producing a cleaner dataset for final refinement.

More recent methods incorporate temporal information into the filtering process. SELF (NGUYEN et al., 2020) employs a temporal ensemble of predictions to identify samples with inconsistent labels:

$$\hat{\mathbf{y}}^{(t+1)} = \alpha \hat{\mathbf{y}}^{(t)} + (1-\alpha) f(\mathbf{x}; \theta^{(t)}) \tag{2.39}$$

where $\alpha \in [0, 1]$ controls the momentum of the temporal ensemble. Samples for which the noisy label $\tilde{\mathbf{y}}$ disagrees with the one-hot encoding of the updated pseudo-label $\hat{\mathbf{y}}^{(t+1)}$ are reclassified as unlabeled, enabling subsequent semi-supervised training via the Mean Teacher framework.

Although multi-model training is more effective at mitigating confirmation bias than single-model approaches, it has limitations. Premature convergence remains a challenge, especially in high-noise scenarios where model diversity decreases quickly. Moreover, computational and memory costs roughly double, restricting use in resource-constrained settings.

2.4.2.3 *Semi-supervised*

SSL has emerged as a fundamental strategy for addressing LNL, given the structural overlap between both scenarios. In essence, both SSL and LNL involve training datasets that cannot be entirely trusted for direct supervision: in SSL, some samples lack annotations, while in LNL, some available annotations are potentially incorrect. The key distinction lies in how these problematic samples are handled — SSL deals with explicitly unlabeled data, while LNL requires identifying reliable samples before applying semi-supervised techniques.

One class of methods focuses on enforcing consistency in model predictions across different views or perturbations of the same input. These approaches assume that augmented versions of a sample should produce similar outputs. A pioneering example is Ding et al. (2018), which combines traditional supervision with consistency regularization:

$$\theta^* = \arg\min_\theta \frac{1}{|\mathcal{D}_c|} \sum_{(\mathbf{x},\tilde{\mathbf{y}})\in\mathcal{D}_c} \ell(f(\mathbf{x};\theta),\tilde{\mathbf{y}}) + \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x},\tilde{\mathbf{y}})\in\mathcal{D}} \|f(\mathbf{x};\theta) - f(\mathbf{x};\hat{\theta})\|_2^2 \tag{2.40}$$

where $\mathcal{D}_c$ are clean samples, and $f(\mathbf{x};\hat{\theta})$ the model predictions under dropout perturbations.

Building on the consistency principle, Kong et al. (2019) leverages the assumption that decision boundaries should pass through low-density regions, employing Rényi entropy for samples classified as noisy:

$$\theta^* = \arg\min_\theta \frac{1}{|\mathcal{D}_c|} \sum_{(\mathbf{x},\tilde{\mathbf{y}})\in\mathcal{D}_c} \ell(f(\mathbf{x};\theta),\tilde{\mathbf{y}}) + \frac{1}{|\mathcal{D}_n|} \sum_{(\mathbf{x},\tilde{\mathbf{y}})\in\mathcal{D}_n} \mathbb{H}_r(f(\mathbf{x};\theta),\alpha) \tag{2.41}$$

where $\mathcal{D}_n = \mathcal{D} \setminus \mathcal{D}_c$ are non-pseudo-clean samples, and $\mathbb{H}_r(\cdot,\alpha)$ is the Rényi entropy.

Another influential line of work builds upon traditional pseudo-labeling by integrating advanced DA techniques. A prominent example is DivideMix (LI; SOCHER; HOI, 2020), which combines MixUp with SSL by partitioning the training data into labeled (clean) samples $\mathcal{D}_c$ and unlabeled (noisy) samples $\mathcal{D}_n$. The method operates through a two-stage augmentation process:

$$\hat{\mathcal{D}}_c = \{(\hat{\mathbf{x}},\tilde{\mathbf{y}}) \mid \hat{\mathbf{x}} = a(\mathbf{x}), a\sim\mathcal{A}, (\mathbf{x},\tilde{\mathbf{y}})\in\mathcal{D}_c\} \tag{2.42}$$

$$\hat{\mathcal{D}}_n = \{(\hat{\mathbf{x}},\mathbf{q}) \mid \mathbf{q} = \text{sharpen}(\mathbb{E}_{a\sim\mathcal{A}}[f(\hat{\mathbf{x}};\theta)]), \hat{\mathbf{x}} = a(\mathbf{x}), a\sim\mathcal{A}, \mathbf{x}\in\mathcal{D}_n\} \tag{2.43}$$

where $\mathcal{A}$ denotes a set of stochastic augmentation functions, and sharpen$(\cdot)$ enhances the distribution through temperature scaling. Subsequently, MixUp interpolation is applied between augmented data and shuffled version $\mathcal{W} = \text{shuffle}(\hat{\mathcal{D}}_c \cup \hat{\mathcal{D}}_n)$. Final optimization combines supervised and unsupervised terms:

$$\theta^* = \arg\min_\theta \frac{1}{|\mathcal{D}'_c|} \sum_{(\mathbf{x},\tilde{\mathbf{y}})\in\mathcal{D}'_c} \ell(f(\mathbf{x};\theta),\tilde{\mathbf{y}}) + \frac{\lambda_n}{|\mathcal{Y}|\cdot|\mathcal{D}'_n|} \sum_{(\mathbf{x},\tilde{\mathbf{y}})\in\mathcal{D}'_n} \|\tilde{\mathbf{y}} - f(\mathbf{x};\theta)\|_2^2 \tag{2.44}$$

Recent methods have sought to improve robustness by combining different SSL strategies, particularly consistency regularization and pseudo-labeling. One such example is the

SSR method (FENG; TZIMIROPOULOS; PATRAS, 2022), which integrates pseudo-labeling with consistency based on cosine similarity:

$$\theta^* = \arg\min_{\theta} \frac{1}{|\mathcal{D}_{pl}|} \sum_{(\mathbf{x},\hat{\mathbf{y}}) \in \mathcal{D}_{pl}} \ell(f(\mathbf{x};\theta), \hat{\mathbf{y}}) + \ell_{fc}(f(a_1(\mathbf{x});\theta), f(a_2(\mathbf{x});\theta)) \tag{2.45}$$

where $\mathcal{D}_{pl}$ contains pseudo-labeled samples, $a_1, a_2 \sim \mathcal{A}$ are different augmentation functions, and $\ell_{fc}(\cdot,\cdot)$ computes cosine distance-based consistency.

More advanced methods adapt SSL frameworks for noisy labels. FixMatch variants use noise-aware pseudo-label thresholds, while temporal ensembling (e.g., Mean Teacher) weights predictions via moving averages of sample confidence. Despite their promise, SSL approaches for LNL face two key challenges: false positives (noisy samples classified as clean) introduce bias, while correctly labeled but complex examples are often discarded, depriving the model of valuable information. These issues highlight the need for refined sample selection criteria that distinguish label noise from inherent sample difficulty.

2.4.2.4 Self-supervised pre-training

Self-supervised pretraining has emerged as a key strategy to address the initialization challenge in LNL, which occurs when randomly initialized models quickly overfit spurious patterns from noisy labels, hindering robust representation learning. The vulnerability of DNNs to noisy labels is particularly high early in training, before stable representations develop. Hendrycks, Lee & Mazeika (2019) showed that randomly initialized models tend to memorize noisy patterns, impairing robustness and uncertainty calibration. This behavior stems from differential memorization: DNNs learn simple patterns before complex ones (ARPIT et al., 2017), but noisy labels disrupt this hierarchy. Zheltonozhskii et al. (2022) formalized this as the "warm-up obstacle," showing that early stopping alone cannot prevent premature noise memorization.

Given a noisy dataset $\mathcal{D} = \{(\mathbf{x}_i, \tilde{\mathbf{y}}_i)\}_{i=1}^{M}$ with a nonzero probability of incorrect labels, the key challenge is avoiding premature convergence to noise-fitting solutions rather than meaningful representations during early epochs, with two prominent instantiations of this framework being particularly relevant for LNL:

- Contrastive learning: this approach learns representations invariant to meaningful transformations and capable of distinguishing instances. Given an unlabeled set $\mathcal{D}_u = \{\mathbf{x}_i\}_{i=1}^{N}$, the contrastive objective is:

$$\mathcal{L}_{\text{contrast}} = -\mathbb{E}_{\mathbf{x}\sim\mathcal{D}_u}\left[\log \frac{\exp\left(\text{sim}(\phi(\mathbf{a}_1(\mathbf{x})), \phi(\mathbf{a}_2(\mathbf{x})))/\zeta\right)}{\sum_{\mathbf{x}'\in\mathcal{B}\setminus\{\mathbf{x}\}} \exp\left(\text{sim}(\phi(\mathbf{a}_1(\mathbf{x})), \phi(\mathbf{a}_2(\mathbf{x}')))/\zeta\right)}\right] \tag{2.46}$$

  where $\text{sim}(\cdot,\cdot)$ measures similarity, $\mathbf{a}_1, \mathbf{a}_2$ are stochastic augmentations, $\phi : \mathcal{X} \to \mathbb{R}^d$ is the encoder, $\mathcal{B}$ the current batch, and ζ controls distribution concentration.

- Transformation prediction: an alternative approach exploits the prediction of applied transformations as an implicit regularization mechanism. Given a set of transformations $\mathcal{T} = \{t_k\}_{k=1}^{K}$ and a distribution $p(k)$, the objective is:

$$\mathcal{L}_{\text{transform}} = \mathbb{E}_{\mathbf{x}\sim\mathcal{D}_u,\, k\sim p(k)} \left[\ell\big(g(\phi(t_k(\mathbf{x})); \gamma), k\big) \right] \tag{2.47}$$

  where $g : \mathbb{R}^d \to \Delta^{K-1}$ is a classification head with parameters γ.

Zheltonozhskii et al. (2022), for example, proposed the Contrast-to-Divide (C2D) method, which combines contrastive pretraining with robust learning on noisy labels. First, representations are learned from unlabeled data by minimizing the contrastive loss:

$$\phi^* = \arg\min_{\phi} \mathcal{L}_{\text{contrast}}(\mathcal{D}_u; \phi), \tag{2.48}$$

then the model is fine-tuned on noisy labeled data by minimizing a robust loss:

$$\theta^* = \arg\min_{\theta} \mathcal{L}_{\text{robust}}(\mathcal{D}; f(\cdot;\theta)), \tag{2.49}$$

where $f(\cdot;\theta) = h(\phi^*(\cdot);\theta)$ with h being the task-specific head. This procedure leverages stable representations learned during pretraining to mitigate the impact of noisy labels.

Self-supervised pretraining provides strong initializations for LNL, capturing semantic structures before exposure to noisy labels, reducing early error memorization, and serving as implicit regularization by promoting natural data invariances. It can be applied without large labeled datasets, though drawbacks include increased computational cost, dependence on alignment between pretext and downstream tasks, and challenges in obtaining large unlabeled datasets in specialized domains. Today, it is standard in state-of-the-art LNL methods, improving robustness and uncertainty calibration (HENDRYCKS; LEE; MAZEIKA, 2019), with domain-specific adaptations Zheltonozhskii et al. (2022), marking pretraining as essential for modern hybrid pipelines combining robust representations with tailored noise correction.

2.4.3 Data processing

2.4.3.1 Data cleaning

Data cleaning methods aim to mitigate the impact of noisy labels by identifying and removing potentially mislabeled samples from the training set. Unlike loss correction or robust regularization, these methods operate directly on the dataset, refining its composition to increase the proportion of correctly labeled examples. Formally, given a noisy training set $\mathcal{D}$, the goal is to construct a cleaner subset $\mathcal{D}_c \subseteq \mathcal{D}$ that maximizes a label quality function:

$$\mathcal{D}_c = \arg\max_{\mathcal{S}\subseteq\mathcal{D}} Q(\mathcal{S}), \tag{2.50}$$

where $Q(\mathcal{S})$ measures the label purity of $\mathcal{S}$, favoring correctly labeled samples.

The theoretical foundation of modern DC stems from the observation that DNNs display a temporally structured memorization bias. In particular, Arpit et al. (2017) demonstrated that DNNs initially fit cleanly labeled examples (typically associated with lower losses) before eventually memorizing noisy labels. This insight supports the loss-based criterion, where a sample is clean if:

$$\mathcal{D}_c = \{(\mathbf{x}, \tilde{\mathbf{y}}) \in \mathcal{D} \mid \ell(f(\mathbf{x};\theta), \tilde{\mathbf{y}}) < \tau_{\text{loss}}\}, \tag{2.51}$$

The Co-Teaching method (HAN et al., 2018), previously described in Subsection 2.4.2.2, implements loss-based DC through a collaborative paradigm in which two networks are trained simultaneously, each selecting small-loss samples to supervise the other:

$$\theta_1^{(t+1)} = \theta_1^{(t)} - \eta \nabla_{\theta_1} \frac{1}{|\mathcal{S}_2^{(t)}|} \sum_{(\mathbf{x},\tilde{\mathbf{y}}) \in \mathcal{S}_2^{(t)}} \ell(f(\mathbf{x};\theta_1), \tilde{\mathbf{y}}) \tag{2.52}$$

$$\theta_2^{(t+1)} = \theta_2^{(t)} - \eta \nabla_{\theta_2} \frac{1}{|\mathcal{S}_1^{(t)}|} \sum_{(\mathbf{x},\tilde{\mathbf{y}}) \in \mathcal{S}_1^{(t)}} \ell(f(\mathbf{x};\theta_2), \tilde{\mathbf{y}}), \tag{2.53}$$

where $\mathcal{S}_1^{(t)}$ and $\mathcal{S}_2^{(t)}$ are the small-loss subsets selected by each model at iteration t.

The method ITLM (SHEN; SANGHAVI, 2019) refines this idea by alternating between sample selection and retraining, optimizing over the k cleanest samples:

$$\theta^* = \arg\min_{\theta} \frac{1}{k} \sum_{i \in \mathcal{I}_k} \ell(f(\mathbf{x}_i;\theta), \tilde{\mathbf{y}}_i), \tag{2.54}$$

where $\mathcal{I}_k$ indexes the k samples with the lowest loss, and k is dynamically adjusted based on the estimated noise rate.

Using iterative approaches, the INCV (CHEN et al., 2019) and O2U-Net (HUANG et al., 2019) methods are notable. INCV utilizes an algorithm combining cross-validation with progressive DC, randomly dividing data into two halves at each iteration to train the network on one part and classify the other, adding samples with consistent labels to the reliable set while removing those with higher loss, progressively refining the clean set. O2U-Net implements overfitting-to-underfitting cycles using cyclic learning rates, accumulating loss statistics over multiple epochs before performing dataset cleaning based on these statistics.

Given the limitations of purely loss-based methods that may discard informative samples, SELFIE (SONG; KIM; LEE, 2019) proposes a nuanced strategy distinguishing definitively clean examples from potentially recoverable ones. It splits the training data into a clean set $\mathcal{D}_c$, selected by small-loss, and a recoverable set $\mathcal{D}_r$, containing samples with consistent predicted labels over recent iterations. The training objective combines both subsets:

$$\mathcal{L} = \frac{1}{|\mathcal{D}_c|} \sum_{(\mathbf{x},\tilde{\mathbf{y}}) \in \mathcal{D}_c} \ell(f(\mathbf{x};\theta), \tilde{\mathbf{y}}) + \frac{1}{|\mathcal{D}_r|} \sum_{(\mathbf{x},\hat{\mathbf{y}}) \in \mathcal{D}_r} \ell(f(\mathbf{x};\theta), \hat{\mathbf{y}}) \tag{2.55}$$

where $\mathcal{D}_r$ includes samples with low entropy over recent epochs, indicating recoverability.

Alternatively, the Area Under the Margin (AUM) method (PLEISS et al., 2020) proposes a temporal metric that tracks the confidence evolution of model predictions throughout training.

For each sample, AUM measures the cumulative margin between the assigned label's logit and the maximum logit of competing classes:

$$\text{AUM}(\mathbf{x}, \tilde{\mathbf{y}}) = \sum_{t=1}^{T} \left[\tilde{\mathbf{y}}^\top f(\mathbf{x}; \theta^{(t)}) - \max_{k \neq \arg\max_j \tilde{\mathbf{y}}_j} f(\mathbf{x}; \theta^{(t)})_k \right] \tag{2.56}$$

where $\tilde{\mathbf{y}}^\top f(\mathbf{x}; \theta^{(t)})$ represents the logit corresponding to the observed label $\tilde{\mathbf{y}}$ at iteration t, and $\max_{k \neq \arg\max_j \tilde{\mathbf{y}}_j} f(\mathbf{x}; \theta^{(t)})_k$ is the maximum logit of competing classes. Samples with correct labels maintain higher prediction margins over time compared to mislabeled examples, providing an interpretable criterion for sample selection based on confidence stability.

To enhance robustness in loss-based filtering, Xia et al. (2022) which combines smooth and hard truncation strategies. For the smooth case, the robust mean loss is estimated as

$$\tilde{\mu} = \frac{1}{t} \sum_{i=1}^{t} \psi(\ell_i), \quad \text{where} \quad \psi(\ell_i) = \log\left(1 + \ell_i + \frac{\ell_i^2}{2}\right), \tag{2.57}$$

where $\ell_i \geq 0$ denotes sample loss, and $\psi(\cdot)$ is a non-decreasing influence function that attenuates large losses (outliers). Hard truncation removes extreme losses before computing robust mean, with sample selection guided by concentration inequalities and sample usage frequency.

From a different perspective, data-centric methods address noisy labels by focusing on data quality rather than the learning algorithm. They aim to evaluate and refine label reliability after training in a model-agnostic manner. Among the most representative approaches is Confident Learning (CL) (NORTHCUTT; JIANG; CHUANG, 2021), which directly estimates the joint distribution between noisy and true labels. Under a class-conditional noise assumption, expressed as $p(\tilde{Y} \mid Y, X) = p(\tilde{Y} \mid Y)$, CL defines class-specific confidence thresholds as

$$t_i = \frac{1}{|\mathcal{D}_{\tilde{\mathbf{y}}=\mathbf{e}_i}|} \sum_{\mathbf{x} \in \mathcal{D}_{\tilde{\mathbf{y}}=\mathbf{e}_i}} \mathbf{e}_i^\top f(\mathbf{x}; \theta), \tag{2.58}$$

where $\mathcal{D}_{\tilde{\mathbf{y}}=\mathbf{e}_i} = \{\mathbf{x} : \tilde{\mathbf{y}} = \mathbf{e}_i\}$ denotes the set of samples with observed noisy one-hot label $\mathbf{e}_i$, and $\mathbf{e}_i^\top f(\mathbf{x}; \theta)$ is the i-th component of the predicted probability vector from a model trained out-of-sample. Based on these thresholds, CL constructs confident subsets

$$\mathcal{D}_{i,j} = \{\mathbf{x} \in \mathcal{D}_{\tilde{\mathbf{y}}=\mathbf{e}_i} : \mathbf{e}_j^\top f(\mathbf{x}; \theta) \geq t_j,\ j = \arg\max_k \mathbf{e}_k^\top f(\mathbf{x}; \theta)\}, \tag{2.59}$$

representing transitions $i \to j$ from noisy to clean class. The joint distribution is estimated as

$$\hat{Q}_{i,j} = \frac{|\mathcal{D}_{i,j}|}{\sum_k |\mathcal{D}_{i,k}|}. \tag{2.60}$$

In general, although DC methods significantly reduce overfitting to incorrect labels and improve dataset quality, they still face important limitations. Aggressive removal of examples can exclude valuable and difficult instances, compromising model robustness. Additionally, the dependence on loss-based heuristics assumes that learning dynamics are consistent across different architectures and datasets, which is not always the case.

Furthermore, binary classification of samples as clean or noisy may oversimplify the problem, failing to capture the continuous spectrum of label reliability. Static cleaning thresholds disregard model confidence evolution during training. Finally, computational overhead and hyperparameter sensitivity (such as thresholds and selection ratios) impose practical challenges on these approaches.

2.4.3.2 Sample selection

Unlike data cleansing methods, sample selection techniques retain the training set by categorizing samples as clean or noisy and applying distinct strategies to each group. This approach acknowledges the informative potential of incorrectly labeled examples, avoiding premature discard of valuable data. The general formulation can be expressed as:

$$\theta^* = \arg\min_{\theta} \left[\frac{1}{|\mathcal{D}_c|} \sum_{(\mathbf{x},\tilde{\mathbf{y}})\in\mathcal{D}_c} \ell_c(f(\mathbf{x};\theta),\tilde{\mathbf{y}}) + \frac{1}{|\mathcal{D}_n|} \sum_{(\mathbf{x},\tilde{\mathbf{y}})\in\mathcal{D}_n} \ell_n(f(\mathbf{x};\theta),\tilde{\mathbf{y}}) \right] \quad (2.61)$$

$$\text{subject to} \quad \mathcal{D}_c = \text{clean}(\mathcal{D}), \quad \mathcal{D}_n = \mathcal{D} \setminus \mathcal{D}_c$$

where $\mathcal{D}_c$ and $\mathcal{D}_n$ are the clean and noisy data subsets, respectively. The function $\text{clean}(\cdot)$ determines this partition, while $\ell_c(\cdot)$ and $\ell_n(\cdot)$ are the loss functions for each category.

The most widely adopted theoretical foundation for defining $\text{clean}(\cdot)$ is the small-loss trick (ARPIT et al., 2017), already introduced in Section 2.4.3.1. Formally, the clean sample set $\mathcal{D}_c$ is defined using the same loss threshold τ_{loss}:

$$\mathcal{D}_c = \{(\mathbf{x},\tilde{\mathbf{y}}) \mid (\mathbf{x},\tilde{\mathbf{y}}) \in \mathcal{D},\ \ell_c(f(\mathbf{x};\theta),\tilde{\mathbf{y}}) < \tau_{\text{loss}}\}. \quad (2.62)$$

This threshold is challenging to define in practice, motivating various methods to operationalize the small-loss trick for clean sample selection.

One of the first methods to address this issue was Co-Teaching (previously discussed in Sections 2.4.2.2 and 2.4.3.1), proposed by Han et al. (2018), which introduces a dynamic schedule to adjust the proportion of selected samples during training:

$$R(t) = 1 - \epsilon \min\left(1, \left(\frac{t}{t_k}\right)^{\alpha}\right) \quad (2.63)$$

where $R(t)$ denotes the fraction of samples classified as clean at iteration t, and ϵ, α, and t_k are hyperparameters. This formulation reflects the intuition that, in the early stages of training ($R(t) \approx 1$), most samples should be considered reliable, with a gradual reduction as the model becomes more likely to memorize noisy labels.

Building on this approach, Han et al. (2020) proposed SIGUA, which implements a refined three-set strategy: the lowest $p\%$ loss samples form the reliable set used for standard gradient descent; the next $q\%$ constitute the unreliable set, used for weighted gradient ascent; and the remaining samples are discarded from the optimization process.

In parallel, uncertainty estimation-based approaches have emerged as promising alternatives. Along these lines, Köhler, Autenrieth & Beluch (2019) explored uncertainty measures such as variation ratio and prediction standard deviation, computed via deep ensembles or Monte Carlo dropout, assuming that more uncertain samples are more likely to contain incorrect labels.

Extending the small-loss criterion, methods such as SELF (HUANG; ZHANG; ZHANG, 2020) and ProMix (XIAO et al., 2023) incorporate mechanisms to improve sample selection. As discussed in Section 2.4.2.2, SELF uses temporal ensembling of predictions to identify likely mislabeled samples. Complementarily, ProMix applies a more restrictive dual criterion, selecting only samples with both small loss and high-confidence predictions:

$$\mathcal{D}_c = \left\{(\mathbf{x}, \tilde{\mathbf{y}}) \mid \ell_{\text{CE}}(f(\mathbf{x};\theta), \tilde{\mathbf{y}}) < \tau_{\text{loss}} \wedge \tilde{\mathbf{y}}^\top f(\mathbf{x};\theta) > \tau_{\text{conf}}\right\} \tag{2.64}$$

where $\tilde{\mathbf{y}}^\top f(\mathbf{x};\theta)$ denotes the predicted confidence, and $\tau_{\text{loss}}, \tau_{\text{conf}}$ are the respective thresholds. This strategy helps reduce false positives, as mislabeled samples may sometimes have low loss but rarely high confidence.

A distinct line of methods explores latent representations and geometric features of the data. Ortego et al. (2021) proposed the MOIT method, which employs intermediate representations extracted via a projection $g(\cdot;\phi) : \mathcal{V} \to \mathcal{Z}$, applied over the backbone $f(\cdot;\hat{\theta}) : \mathcal{X} \to \mathcal{V}$. The model is optimized using a contrastive loss that brings representations of samples from the same class closer in the space $\mathcal{Z}$, followed by selection via k-NN classification:

$$\mathcal{D}_c = \left\{(\mathbf{x}, \tilde{\mathbf{y}}) \in \mathcal{D} \;\middle|\; d(\mathbf{x}, \tilde{\mathbf{y}}) < \gamma\right\}, \quad d(\mathbf{x}, \tilde{\mathbf{y}}) = -\tilde{\mathbf{y}}^\top \log \hat{\mathbf{p}},\ \hat{\mathbf{p}} \in \Delta^{|\mathcal{Y}|-1} \tag{2.65}$$

Here, $\hat{\mathbf{p}}$ is the predicted probability distribution computed via k-NN in the representation space $\mathcal{Z}$. Similarly, the SSR method (FENG; TZIMIROPOULOS; PATRAS, 2022) classifies samples as clean only when the k-NN classifier's prediction matches the observed label.

Mixture model approaches effectively model the loss distribution. Arazo et al. (2019) proposed BMM, where training losses are modeled by Beta distributions, as shown in Equation (2.21), with the smallest loss component corresponding to clean samples. This inspired DivideMix (LI; SOCHER; HOI, 2020), employing a two-component GMM with SSL, treating noisy samples as unlabeled data. For open-set scenarios, Sachdeva et al. (2021) extended this in EvidentialMix, using a three-component GMM with subjective logic loss to quantify uncertainty. This distinguishes closed-set clean samples (lowest mean), closed-set noisy samples (highest mean), and open-set samples (intermediate mean).

With a specific focus on the sample retention curve, Yang et al. (2024) approached sample selection as a bi-level optimization problem:

$$\begin{aligned} R^* &= \arg\min_{R \in \mathcal{F}} \mathcal{L}_{\text{val}}(\theta^*, R, \mathcal{D}_{\text{val}}) \\ \text{subject to} \quad \theta^* &= \arg\min_{\theta} \mathcal{L}(\theta, R, \mathcal{D}) \end{aligned} \tag{2.66}$$

where $\mathcal{D}_{\text{val}}$ is the clean-labeled validation set, $\mathcal{F}$ is the search space for R defined from K base functions, and $\mathcal{L}_{\text{val}}$, $\mathcal{L}$ are the validation and training loss functions, respectively.

In general, sample selection methods face inherent limitations. They often assume that the loss distributions of clean and noisy samples are well separated, which may fail under high noise rates or ambiguous classes. Many also require prior knowledge of the noise rate or access to clean validation sets, restricting practical use. Their effectiveness ultimately depends on the quality of the selection function clean$(\cdot)$, which remains an open challenge, indicating that strategy choice should be guided by the dataset and application domain.

2.4.3.3 Data augmentation

Data augmentation enhances the robustness of DL models in the presence of noisy labels. Unlike approaches focused on label correction or sample selection, augmentation operates by expanding the dataset through transformations that preserve class semantics while introducing controlled variability. This expansion helps mitigate spurious correlations between noise patterns and data attributes, providing implicit regularization.

DA transformations, $\mathcal{A} : \mathcal{X} \to \mathcal{X}$, map input samples to modified versions that preserve the true label. The expanded dataset can be formalized as:

$$\mathcal{D}_{\text{aug}} = \{(\mathcal{A}(\mathbf{x}_i), \tilde{\mathbf{y}}_i) : (\mathbf{x}_i, \tilde{\mathbf{y}}_i) \in \mathcal{D}, \mathcal{A} \in \mathcal{A}_{\text{set}}\} \tag{2.67}$$

where $\mathcal{A}_{\text{set}}$ is the set of admissible transformations, focused on spatial operations such as rotation, translation, scaling, flipping, and cropping, designed to preserve class semantics and break spurious correlations between data artifacts and noisy labels. However, these classical approaches exhibit limited effectiveness under heavy noise, motivating the development of advanced methods (SONG et al., 2023).

A major advance in robust augmentation was proposed by Zhang et al. (2018) with MixUp. This approach represented a paradigm shift by proposing convex interpolation in both feature and label spaces, creating virtual examples as:

$$\mathbf{x}_{\text{mix}} = \lambda \mathbf{x}_i + (1-\lambda)\mathbf{x}_j\,, \quad \tilde{\mathbf{y}}_{\text{mix}} = \lambda \tilde{\mathbf{y}}_i + (1-\lambda)\tilde{\mathbf{y}}_j \tag{2.68}$$

where $\lambda \sim \text{Beta}(\alpha, \alpha)$ with $\alpha > 0$ controlling interpolation smoothness. MixUp smooths the decision space via convex combinations, promoting linear behavior and reducing corruption sensitivity. This regularization broadens training distribution around original samples. Effectiveness arises from noise dilution, as interpolation mitigates label corruption, especially when only one label is noisy.

A recent DA approach in the context of LNL is the Augmented Descent method (NISHI et al., 2021a), which formalizes a hierarchy between weak and strong transformations, differentiating levels of augmentation intensity. Weak augmentations ($\mathcal{A}_W$) correspond to traditional low-distortion transformations, while strong augmentations ($\mathcal{A}_S$) involve more aggressive operations, often derived from automatic search methods such as AutoAugment. This distinction can be defined using distortion metrics:

$$\mathcal{A}_W = \{a \in \mathcal{A} : d(a(\mathbf{x}), \mathbf{x}) \leq \epsilon_w\}, \quad \mathcal{A}_S = \{a \in \mathcal{A} : d(a(\mathbf{x}), \mathbf{x}) \leq \epsilon_s\} \tag{2.69}$$

where $d(\cdot, \cdot)$ represents a distance metric, and $\epsilon_w < \epsilon_s$ defines distortion thresholds.

The optimization process of AugDesc elegantly combines sample selection with differential augmentation. It employs weak augmentations to identify potentially clean samples using a binary clean criterion $\text{clean}(\mathbf{x}, \tilde{\mathbf{y}}) \in \{0, 1\}$, while strong augmentations are applied during backpropagation. Clean samples are trained using their original labels, whereas suspicious ones rely on pseudo-labels obtained through consensus across multiple weak transformations.

Parallel to training augmentation methods, there is growing interest in post-hoc techniques for enhancing inference robustness, such as Test-Time Augmentation (TTA). In this approach, the model is applied to multiple augmented test versions, and predictions are aggregated for the final decision. Given a trained model $f(\cdot; \theta^*)$ and test-time transformations $\mathcal{A}_{\text{test}}$, the TTA prediction is computed as:

$$\hat{\mathbf{y}}_{\text{TTA}} = \mathbb{E}_{a \sim \mathcal{A}_{\text{test}}} \left[f(a(\mathbf{x}_{\text{test}}); \theta^*) \right] \tag{2.70}$$

Smart & Carneiro (2023) reported significant improvements using TTA in noisy-label classification, showing that aggregating multiple predictions can help mitigate residual training errors. Its simplicity and independence from transformation parallelism make TTA an attractive option for enhancing robustness without substantially increasing inference time.

DA effectiveness in noisy label contexts can be understood from different theoretical perspectives. From an information theory standpoint, DA expands the data distribution support, reducing model dependence on specific features correlated with noise. This process functions as regularization that penalizes overly specialized solutions.

However, for DA to be effective, it must adhere to a central principle: class preservation. For every transformation $a \in \mathcal{A}$, the true label $\mathbf{y}^*$ must remain the most likely outcome, that is,

$$\arg\max_c p(c|\mathbf{x}) = \arg\max_c p(c|a(\mathbf{x})).$$

When this property is violated, additional noise is introduced into the training process, which proves especially detrimental in scenarios already compromised by label corruption.

Despite these benefits, DA is not without limitations. The so-called "augmentation curse" emerges when excessive transformations introduce additional noise, thereby amplifying rather than mitigating the negative effects of incorrect labels. Moreover, when the applied transformations are incompatible with the underlying nature of the data, they can violate the semantic preservation assumption, ultimately degrading model performance.

2.4.3.4 Pseudo-labeling

Pseudo-labeling goes beyond simple correction strategies by actively refining labels using knowledge gained during training. Unlike conventional bootstrapping, which relies on

fixed parameters and static interpolation between observed labels and predictions, modern methods are adaptive, automatically adjusting confidence and correction strategies based on model evolution and sample characteristics, creating a dynamic refinement framework.

Pseudo-labeling generates labels $\mathbf{y}_i^{\text{pseudo}}$ through temporal refinement functions:

$$\mathbf{y}_i^{\text{pseudo}}(t+1) = \mathcal{F}_t(\mathbf{x}_i, \tilde{\mathbf{y}}_i, f(\cdot;\theta_t), \mathcal{H}_i^{t-w:t}), \tag{2.71}$$

where $\mathcal{F}_t$ represents the temporal refinement function at time t, and $\mathcal{H}_i^{t-w:t} = \{f(\mathbf{x}_i;\theta_s)\}_{s=t-w}^{t}$ represents the prediction history for sample i within window $[t-w,t]$, enabling contextual adaptation and memory usage, contrasting static approaches.

One solution to mitigate prediction instability in instantaneous refinement methods was SELF (HUANG; ZHANG; ZHANG, 2020), previously discussed in Section 2.4.3.2. This approach introduces exponential moving averages to smooth pseudo-labels over time:

$$\mathbf{y}_i^{\text{pseudo}}(t+1) = \alpha \mathbf{y}_i^{\text{pseudo}}(t) + (1-\alpha) f(\mathbf{x}_i;\theta_t) \tag{2.72}$$

where α controls the forgetting rate, promoting gradual and stable updates of refined labels.

Contrasting with instantaneous refinement, SEAL (CHEN et al., 2021) proposes iterative pseudo-labeling with global memory. The idea is to accumulate predictive knowledge throughout training via temporal averaging of predictions for each instance:

$$\mathbf{y}_i^{\text{pseudo}}(t) = \frac{1}{t}\sum_{s=1}^{t} f(\mathbf{x}_i;\theta_s) \tag{2.73}$$

where t denotes the training epoch. This mechanism acts as a noise filter reducing prediction variance, typically a mixture of useful information and noise from incorrect labels. By computing temporal averages, the method assumes predictions gradually converge to an ideal distribution, where useful information remains consistent across epochs while noise cancels out, resulting in more reliable pseudo-labels.

A distinct approach that explicitly handles different types of label noise is proposed by the DSOS method (ALBERT et al., 2022), which categorizes samples into clean, *in-distribution* (ID) noise, and *out-of-distribution* (OOD) noise. Pseudo-labels $\mathbf{y}_i^{\text{pseudo}}(t)$ are generated through a two-stage process using detection mechanisms: $u_i(t) \in \{0,1\}$ indicates ID noise samples, while $v_i(t) \in [0,1]$ estimates the probability of OOD noise. The refinement process consists of two sequential steps. First, ID noise is corrected using a bootstrapping-based strategy:

$$\mathbf{y}_i^b(t) = (1-u_i(t))\tilde{\mathbf{y}}_i + u_i(t) f(\mathbf{x}_i;\theta_t), \tag{2.74}$$

then OOD samples undergo dynamic softening:

$$\mathbf{y}_i^{\text{pseudo}}(t) = \frac{\exp\left(\frac{v_i(t)\mathbf{y}_i^b(t)}{\alpha}\right)}{\sum_{j=1}^{c}\exp\left(\frac{v_i(t)y_{i,j}^b(t)}{\alpha}\right)}, \tag{2.75}$$

where $\alpha \in [0,1]$ is the temperature parameter. This scheme differentiates treatment: clean samples retain original labels, ID noise gets corrections, and OOD noise becomes uniform.

A fundamental limitation of pseudo-labeling is bias amplification risk. When a model develops systematic bias during training, this can be propagated and reinforced through pseudo-labels. This is especially problematic in scenarios with unbalanced classes or feature-dependent noise. Additionally, most pseudo-labeling methods assume model predictions are more reliable than original labels for suspicious samples. However, this assumption may be violated in early training stages or when architecture is insufficient to capture data complexity.

2.4.4 Model architecture

2.4.4.1 Label transition methods

Label transition methods are among the most theoretically grounded strategies in LNL, using the transition matrix $\mathbf{T} \in [0,1]^{c\times c}$ to model label corruption. Some approaches extend this to instance-dependent noise, allowing corruption to depend on input features. These methods can either estimate the transition matrix externally for loss correction or sample reweighting, or incorporate it directly into the model. Both provide theoretical guarantees of statistical consistency often absent in other methods.

The central challenge lies in the identifiability problem: reliably estimating $\mathbf{T}$ from noisy data is often ill-posed, especially for instance-dependent models. As discussed by Scott (2015), this arises because we have an under-constrained system with only one noisy label per training sample, making multiple mathematically equivalent solutions possible.

These methods follow two main approaches for statistical consistency. The risk-consistent approach by Patrini et al. (2017) constructs a clean risk estimator through loss correction. The classifier-consistent approach (HAN et al., 2021; XIA et al., 2019) provides direct guarantees about classifier convergence. Both formulate the problem as:

$$\theta_1^*, \theta_2^* = \arg\min_{\theta_1,\theta_2} \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x},\tilde{\mathbf{y}})\in\mathcal{D}} -\log \sum_{\mathbf{y}\in\{0,1\}^c} p(\tilde{\mathbf{y}}|\mathbf{y},\mathbf{x},\theta_1)\, p(\mathbf{y}|\mathbf{x},\theta_2) \tag{2.76}$$

where θ_1 parameterizes the transition model and θ_2 the clean label classifier. The decomposition allows each component to be optimized considering its specific function in the learning process.

One well-established strategy to circumvent the identifiability problem uses anchor points, defined as representative samples $\mathbf{x}_y$ for a particular class with one-hot representation $\mathbf{e}_y$, satisfying $p(Y = \mathbf{e}_y|\mathbf{x}_y) \approx 1$. Methods using anchor points (LIU; TAO, 2016; XIA et al., 2019) exploit this property to simplify transition matrix estimation. Under anchor point assumptions, transition matrix elements can be directly determined. In practice, finding true anchor points where $p(Y = \mathbf{e}_y|\mathbf{x}_y) = 1$ is challenging and may require manual assistance. Consequently, modern methods propose automatic selection of pseudo anchor points through $\mathbf{x}_y^* = \arg\max_{\mathbf{x}\in\mathcal{X}} p(\tilde{Y} = \mathbf{e}_y|\mathbf{x})$.

For cases where anchor points are not available, Xia et al. (2019) developed the Reweight T-Revision method, which initializes with a classifier trained on noisy data and subsequently detects pseudo anchor points. The method refines the transition matrix through the minimization of a weighted empirical risk that rebalances the contribution of different samples based on the confidence of their predictions.

Alternative strategies include separability, where, for example, Cheng et al. (2020) assumes that high-confidence samples can be identified as "distilled" and used for robust matrix estimation. In another approach, the clusterability method (ZHU; SONG; LIU, 2021) exploits the property that k-nearest neighbors share clean labels, allowing the formulation of consensus equation systems that relate observations from multiple samples to estimate $\mathbf{T}$ uniquely.

An alternative approach to circumvent identifiability problems involves mathematical decompositions that reformulate the original problem into more tractable subproblems. The Dual-T estimator (YAO et al., 2020) represents an important contribution in this direction, by proposing factorization of the transition matrix $T_{y,\tilde{y}}$ using an intermediate class variable Y'. This decomposition allows dividing the estimation process into two steps: one between Y and Y', and another between Y' and $\tilde{Y}$, replacing direct estimation of $p(\tilde{Y} \mid Y)$ with two simpler steps. The intermediate variable Y' is defined based on the initial classifier's posterior, and subsequent estimates are performed through frequency counting and anchor points.

Regularization approaches address the identifiability problem through penalty terms that guide optimization toward desirable solutions. The volume maximization method (LI et al., 2021) observes that when clean label posterior probabilities are sufficiently spread, the optimal transition matrix maximizes the volume of the simplex formed by its columns. The method adds a regularization term that penalizes small volume matrices, using the logarithm of the transition matrix determinant as a measure. This approach offers identifiability guarantees under relatively weak conditions on the data distribution.

Total variation regularization (ZHANG; NIU; SUGIYAMA, 2021) assumes that the "purest" clean label classifier exhibits the highest pairwise total variation. Regularization encourages predictions for different samples to be sufficiently distinct. This strategy provides an additional separation criterion, promoting unique identifiability of the transition matrix.

For methods that integrate transition matrix estimation directly into the architecture, specific structural modifications are necessary to explicitly incorporate corruption process modeling. The most established implementation of this approach, pioneered by Sukhbaatar et al. (2015), adds a noise adaptation layer parameterized by a matrix $\mathbf{W} \in [0,1]^{c \times c}$ on top of the base network:

$$p(\tilde{\mathbf{y}} = \mathbf{e}_j | \mathbf{x}; \theta, \mathbf{W}) = \sum_{i=1}^{|\mathcal{Y}|} W_{i,j} \cdot p(\mathbf{y} = \mathbf{e}_i | \mathbf{x}; \theta) \tag{2.77}$$

Training uses a strategy starting with $\mathbf{W} = \mathbf{I}$ and trace regularization $\lambda \text{tr}(\mathbf{W})$, where λ gradually increases during training. This allows the network to first learn clean label samples before progressively modeling noise through changes in $\mathbf{W}$.

Dedicated architectures extend this to complex scenarios. The c-model and s-model

(GOLDBERGER; BEN-REUVEN, 2017) first considered instance-dependent transitions, where output layers depend on clean labels and sample features. Later methods include dual networks distinguishing noise prediction from transition estimation, and masking networks (HAN et al., 2018) incorporating prior knowledge through GAN formulations.

Another research line explores the use of quality embedding (YAO et al., 2019), where each sample is associated with a confidence parameter that regulates the transition between labels. This approach avoids explicit transition matrix estimation, replacing it with a variational parameterization that directly models annotation reliability.

Despite their robust theoretical guarantees, transition methods face fundamental limitations that restrict their applicability. Extension to open-set scenarios is challenging, as samples from unobserved classes lack adequate representation in the transition matrix, and effectiveness is limited for purely symmetric noise, where the transition matrix approaches a uniform permutation. Moreover, as observed by Zhu, Song & Liu (2021), anchor point-based methods become problematic when the number of classes is high and training samples are limited, making reliable prototype identification difficult.

2.5 SCOPE OF OUR CONTRIBUTIONS

The CCT method proposed in this thesis is primarily situated within the *training-algorithm* category of LNL approaches—more specifically in the multi-model training subcategory—while incorporating a secondary component from the *data processing* category through the sample-selection mechanism inherited from the Co-Teaching framework. Within the broader landscape summarized in Table 1, this positioning reflects a strategy that preserves original labels, operates solely at the level of training dynamics, and complements or overcomes limitations found in the other major LNL categories.

Compared to loss-function methods, CCT avoids the intrinsic trade-off between theoretical guarantees and optimization efficiency that characterizes robust losses. While approaches such as GCE and SCE improve stability at the cost of weakening theoretical consistency, and MAE provides robustness but suffers from slow convergence, CCT retains the standard CE loss and introduces noise tolerance through algorithmic modulation of the training process rather than through loss redefinition.

Within the training-algorithm category, CCT builds upon the foundational Co-Teaching framework while introducing two key innovations: (1) cyclic variations of the learning rate, which enhance generalization and help escape local minima, replacing the monotonically decaying schedules typically used in Co-Teaching; and (2) a cyclic retention policy that dynamically adjusts the proportion of selected samples, avoiding the stagnation caused by fixed retention rates in later training stages.

CCT further differs from multi-model methods such as CJC-Net, which separate pretraining and noise-filtering phases and frequently rely on clean validation data for model selection. In contrast, CCT integrates cyclicity directly into the collaborative training loop,

Category	Example Methods	Main Advantages	Main Limitations	CCT Compatibility
Loss function	MAE, GCE, SCE, APL, ELR, Bootstrapping, GLC	Simple to implement; low computational overhead; architecture-agnostic.	Fundamental robustness–optimization trade-off; sensitivity under high noise rates or many classes; require careful tuning.	Yes
Training algorithms	MWN, Co-Teaching, JoCoR, CJC-Net, C2D, DivideMix	Directly manipulate training dynamics; robust without modifying loss or architecture; leverage model disagreement.	Require coordination among networks; diversity may collapse; sensitive to sample-selection instability.	**CCT family**
Data processing	INCV, O2U-Net, ProMix	Improve robustness via data filtering or semi-supervised refinement; exploit clean subsets or confident examples.	Depend on heuristics, warm-up phases, or partially clean subsets; may discard informative hard samples or amplify selection bias.	Yes
Model architecture	Noise Adaptation Layer, c-model	Explicitly model noise transitions; strong theoretical grounding.	Require strong noise assumptions; identifiability issues; higher architectural complexity and training cost.	Yes

Table 1 – Summary of the main categories of methods for LNL and the positioning of the proposed CCT.

maintaining diversity between networks and improving robustness even under noisy or limited validation conditions. Additionally, by introducing controlled learning-rate variation during vanilla pretraining, CCT yields stronger initializations than methods that rely on fixed-LR warm-up stages.

Relative to data-processing methods, CCT mitigates the limitations of approaches that rely on static or heuristic thresholds, which may inadvertently discard informative hard samples or depend on partially clean subsets. Unlike pseudo-labeling or correction-based strategies—often vulnerable to error propagation when model predictions are unreliable—CCT preserves original labels throughout training. Finally, unlike architecture-based methods grounded in transition-matrix modeling or graph structures, CCT operates with standard DNN architectures and introduces no structural overhead, reinforcing its practicality and applicability.

A notable advantage of CCT is its methodological transparency: both the cyclic learning-rate and retention schedules are explicitly defined, and their hyperparameters can be tuned through a simple univariate procedure. This design facilitates reproducibility and provides a clear lens through which the influence of training dynamics on robustness can be understood.

In addition, CCT retains the conceptual simplicity and orthogonality characteristic of training-dynamics–based approaches, making it compatible with a wide range of LNL strategies. Because it does not modify the loss function or require architectural changes, CCT can be seamlessly integrated with robust losses, data-driven filtering mechanisms, or noise-aware architectural modules, functioning as a lightweight and plug-and-play component within larger LNL pipelines.

Since CCT belongs to the broader family of training-dynamics–based approaches—and, more specifically, to the subcategory of multi-model training algorithms—it is natural to contextualize it relative to representative methods within this group. Table 2 provides a focused technical comparison, while a comprehensive empirical evaluation is presented later in this thesis.

Table 2 – Technical comparison between CCT and representative multi-model LNL methods.

Method	**Learning-Rate Schedule**	**Sample Selection**	**Retention Policy**	**Key Innovation**
Decouple	Fixed/Decay	Disagreement	None	Updates only on disagreement
Co-Teaching	Fixed/Decay	Small-loss	Fixed schedule	Cross-model teaching of clean samples
Co-Teaching+	Fixed/Decay	Small-loss + disagreement	Disagreement-driven dynamic	Maintains model divergence
JoCoR	Fixed/Decay	Small joint-loss	Implicit (agreement-weighted)	Joint-loss co-regularization
CJC-Net	Multi-phase	Small-loss	Phase-dependent	Cyclical multi-phase training with joint loss
Co-learning	Fixed/Decay	Cooperative (sup. + SSL)	Representation-weighted	Supervised + self-supervised co-training
TCC-Net	Two-phase	Small-loss	Meta-weighted	Contradictory loss + meta-co-teaching
CCT	**Cyclic**	**Small-loss**	**Cyclic coordinated**	**Cyclic schedules to preserve diversity and limit noise memorization**

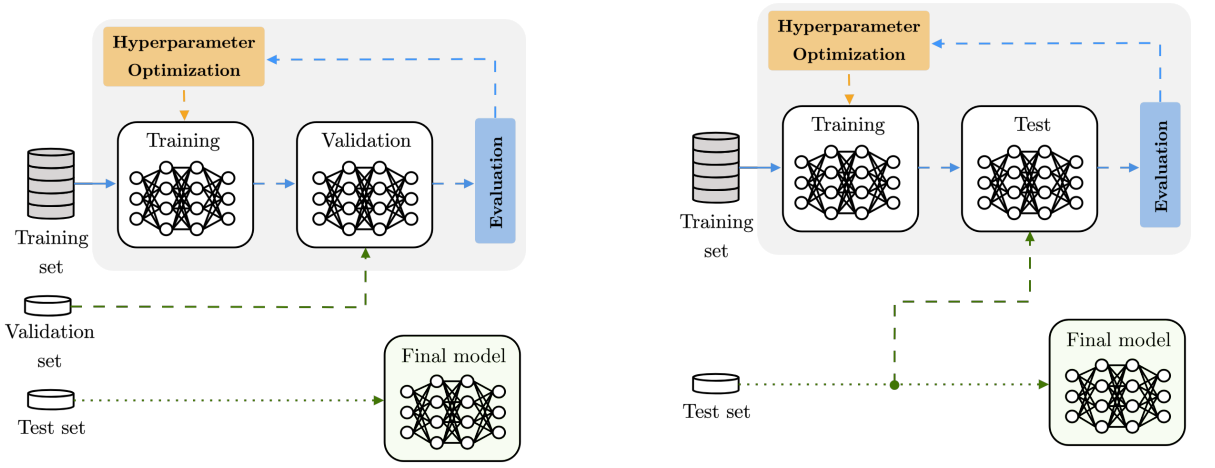
3 BASELINE TRAINING AND OPTIMIZATION METHODOLOGY

As previously discussed, DNNs tend to memorize noisy labels, compromising generalization and performance on unseen data. This issue has fueled increasing interest in the LNL field. Chapter 2 reviewed the main approaches to mitigate this effect, highlighting their methodologies, strengths, and limitations. However, two important aspects warrant closer examination: certain training and evaluation practices in the LNL literature may introduce biases that hinder fair comparisons, and vanilla models used as baselines are often not properly optimized, potentially underestimating their true performance.

This chapter addresses both points by examining methodological issues in current evaluation practices and investigating how proper optimization of vanilla models—particularly through learning rate tuning and scheduling strategies—affects their baseline performance. Our analysis demonstrates that reasonably tuned vanilla models can be considerably more competitive than typically reported in comparative studies.

3.1 REALISTIC TRAINING METHODOLOGY

In the conventional ML methodology, the standard procedure involves dividing the data into three distinct subsets: training, validation, and testing (GOODFELLOW; BENGIO; COURVILLE, 2016), as illustrated in Figure 2a. If a validation set is not previously available, it is typically derived from the split of the original training set. The test set, ideally, is separated in advance and kept unchanged to ensure a fair and realistic evaluation of the model's final performance. Within this framework, the validation set plays a crucial role in monitoring performance during training, optimizing hyperparameters, and selecting models.



(a) Flowchart of the conventional methodology (training → validation → test).

(b) Flowchart of the problematic methodology, where the test set is inappropriately used as the validation set.

Figure 2 – Comparative diagram between the conventional and problematic methodologies.

However, a critical analysis of the LNL literature reveals that several methods (HAN

et al., 2018; HUANG et al., 2019; YU et al., 2019; WEI et al., 2020; ZHANG et al., 2021; XIA; LEE; CHEN, 2023) diverge from the conventional protocol by using the test set for purposes that should be exclusive to validation, as illustrated in Figure 2b. Such practice constitutes a clear form of data leakage: any use of the test set for hyperparameter tuning, model selection, or early stopping contaminates the final evaluation and produces an optimistically biased estimate of generalization. According to standard ML principles (GOODFELLOW; BENGIO; COURVILLE, 2016), the test set must remain strictly unseen until all training and validation stages are complete, ensuring independence between phases and preserving the statistical validity of the evaluation. Violating this protocol results in inflated performance estimates, undermines reproducibility, and compromises fair comparison between methods, often leading to misleading conclusions about true generalization ability.

Another problematic aspect in LNL research relates to performance evaluation practices. Many studies report metrics as averages over the final training epochs, which can systematically obscure the peak performance achieved at earlier stages. In LNL, models often reach their optimal performance before the final stages of training, when the memorization of incorrect labels is already degrading generalization. Consequently, reporting only the average of the final epochs can underestimate a method's true potential and introduce biases that compromise fair comparisons, such that apparent improvements may reflect evaluation artifacts rather than genuine algorithmic advances.

3.2 BASELINE OPTIMIZATION

A precise evaluation of the performance of ML models and methods crucially depends on baseline optimization. This optimization encompasses everything from *vanilla* training with standard configurations and traditional methods to more advanced approaches. A well-calibrated baseline provides a solid reference to determine whether new methods represent genuine advances or merely incremental improvements.

In the ML literature, it is not uncommon to find cases where simple and well-tuned baselines outperform more complex methods for the desired task (OLIVER et al., 2018; GULRAJANI; LOPEZ-PAZ, 2021; NADO et al., 2022). Unfortunately, in the LNL literature, baseline optimization is not a recurring concern, compromising fair comparisons — especially considering that recent techniques increasingly incorporate diverse strategies, such as: self-supervised learning, SSL, meta-learning, robust DA techniques, specialized loss functions, ensemble learning, multi-round training, among others (SONG et al., 2023).

The implications of this lack of interest in baseline optimization are substantial: besides preventing the determination of the real performance achievable with properly optimized baselines, it creates an imbalance in comparisons. While newly proposed approaches receive extensive optimization and hyperparameter tuning, baseline methods remain under-optimized, which artificially increases the performance contrast between them. In this sense, when examining new techniques, a fundamental question arises: which improvements are truly

attributable to innovations when compared to a properly optimized baseline? Without this careful and balanced analysis, conclusions about progress in the field may be distorted, leading to an overestimation of the real gains of new methods compared to artificially weakened baselines.

To contribute partially to this discussion, we investigate the impact of optimizing one of the most influential hyperparameters in ML on the performance of vanilla models: the learning rate in the following subsections. We present an empirical analysis, extending experiments from the literature, to demonstrate how higher learning rates can increase robustness to label noise. Additionally, we propose the use of a specific learning rate scheduler for the vanilla model, demonstrating how this adjustment can make it reasonably competitive.

3.2.1 Learning rate as regularization for noisy labels

In DNN training, regularization is a fundamental component for mitigating overfitting and promoting generalization. In the context of LNL, this need becomes even more critical due to the high susceptibility of DNNs to memorize examples with incorrect labels. Research in LNL has continuously examined the impact of different regularization techniques, both explicit and implicit—such as dropout (CHEN et al., 2024), weight decay (GHIASI; SHAFAHI; ARDEKANI, 2023), DA (NISHI et al., 2021b), and early stopping (BAI et al., 2021)—in mitigating this phenomenon of unwanted memorization. However, the learning rate, widely recognized as the most critical hyperparameter in DL training (BENGIO, 2012), has received limited attention as a regularization mechanism in the specific context of LNL.

One of the most important discoveries about the learning rate in LNL comes from the work of Tanaka et al. (2018), which presented initial evidence that high learning rates can act as an implicit regularization mechanism against noisy labels. However, their experiments were limited to comparisons between only two learning rates in a symmetric noise scenario. In this section, we present a comprehensive empirical investigation that expands the understanding of the role of learning rate in LNL. Through systematic experiments across multiple scenarios and noise levels, our results not only corroborate the initial observations of Tanaka et al. (2018) but also more completely characterize the interaction between learning rate and noise memorization.

3.2.1.1 Experimental setup

To investigate the impact of the learning rate on robustness against noisy labels, we conducted experiments using the CIFAR-10 and CIFAR-100 datasets with the ResNet-18 architecture. Labels were artificially corrupted using two types of noise: symmetric (20% to 80%) and asymmetric (10% to 40%). For asymmetric noise, we followed the methodology proposed by Patrini et al. (2017) for CIFAR-10, while for CIFAR-100, we implemented a scheme where each sample's label was replaced with the subsequent class label.

The models were trained for 200 epochs using the stochastic gradient descent (SGD) optimizer with a momentum of 0.9 and a weight decay of 5×10^{-4}. The learning rate was kept fixed throughout the training process for each experiment, but it was varied across experiments according to the values specified in Table 3.

Model evaluation was conducted using the following metrics:

- Clean Loss: average loss computed over correctly labeled samples, indicating the model's fit to non-corrupted examples;
- Noisy Loss: average loss computed over mislabeled samples, providing insight into the degree of memorization of noisy labels;
- Clean/Noisy Ratio: ratio between the loss on correctly labeled and mislabeled samples, quantifying the model's ability to distinguish between clean and corrupted samples[1].

Experimental Parameters	**Description**	**Tested Values**
Noise	Type and level of noisy label	Symmetric: 20%, 40%, 60%, 80% Asymmetric: 10%, 20%, 30%, 40%
Training Hyperparameters	Optimization configurations	η: $10^{-3} \times \{1, 5, 10, 2.5 \times 10, 5 \times 10, 10^2\}$ Batch size: 64
Validation	Number of independent runs	3 runs to verify consistency

Table 3 – Summary of experimental setup parameters and their tested values for evaluating the robustness of different learning rates against noisy labels.

3.2.1.2 Loss behavior: clean examples vs. noisy

To investigate how the learning rate affects the propensity of DNNs to memorize incorrect labels, we initially analyzed its influence on the model's ability to distinguish between correctly labeled (clean) and corrupted (noisy) examples. This discriminative capacity is quantified through the ratio between average losses on clean and noisy examples (Clean/Noisy), with lower values indicating a model that learns to focus on reliable examples while penalizing noisy ones.

In Figure 3, we present an aggregated analysis based on the average value of this ratio over the 200 training epochs, using the inverse version of the metric (Noisy/Clean) to facilitate visual interpretation—in this case, higher values reflect better discriminative capacity.

[1] Although this ratio provides a compact and directly interpretable way of quantifying the separability between clean and noisy samples — which is the central aspect we aim to monitor — we acknowledge that aggregated metrics have natural limitations, such as sensitivity to outliers and the inability to reflect the full shape of the loss distributions. Even so, we adopt this metric because it captures, in a synthetic and comparable manner, the relative dynamics between clean and noisy losses throughout training.

Each subplot compares the model's behavior under different learning rates and noise levels (symmetric and asymmetric), with batch size fixed at 64.

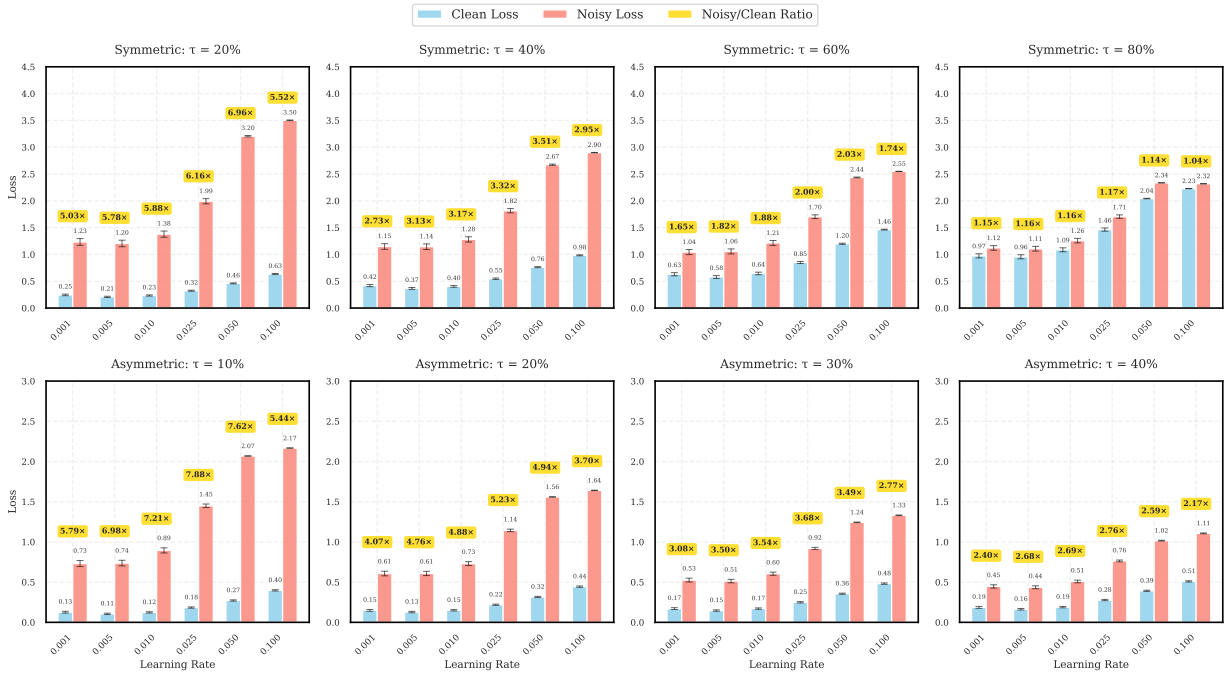


Figure 3 – Discrimination capacity under different fixed learning rates.

The results indicate a non-linear relationship between the learning rate and the Noisy/Clean ratio. Intermediate rates—especially $\eta = 0.01$ and $\eta = 0.025$—tend to produce the highest values of this ratio, which indicates greater separation between the average losses of clean and noisy examples. In contrast, very low learning rates ($\eta = 0.001$ to 0.005) result in lower discrimination, probably due to insufficient parameter updates, while high rates ($\eta \geq 0.05$) frequently lead to ratio degradation or instability under intense noise.

Furthermore, the behavior varies according to the type and intensity of noise:

- Under symmetric noise, the effect of the learning rate shows more stability, with a relatively well-defined effective regularization zone;
- In contrast, asymmetric noise tends to systematically reduce the Noisy/Clean ratio at all levels of η, suggesting greater difficulty in discrimination—a characteristic already discussed in the literature due to its more deceptive nature.

Although this aggregated analysis is useful for identifying general patterns, it does not allow evaluation of the *temporal dynamics* of discrimination over epochs. In particular, it does not reveal the temporal dynamics of the learning process, obscuring fundamental aspects such as when discriminative capacity emerges during training, whether there is stability or deterioration of this capacity over epochs, and how the evolution of discrimination relates to model performance.

To answer these questions, we conducted a detailed temporal analysis based on the original metric (Clean/Noisy), presented in Figure 4. This figure shows, over the 200 training

epochs, the evolution of average losses and the Clean/Noisy ratio, as well as test accuracy, for different noise and learning rate configurations.

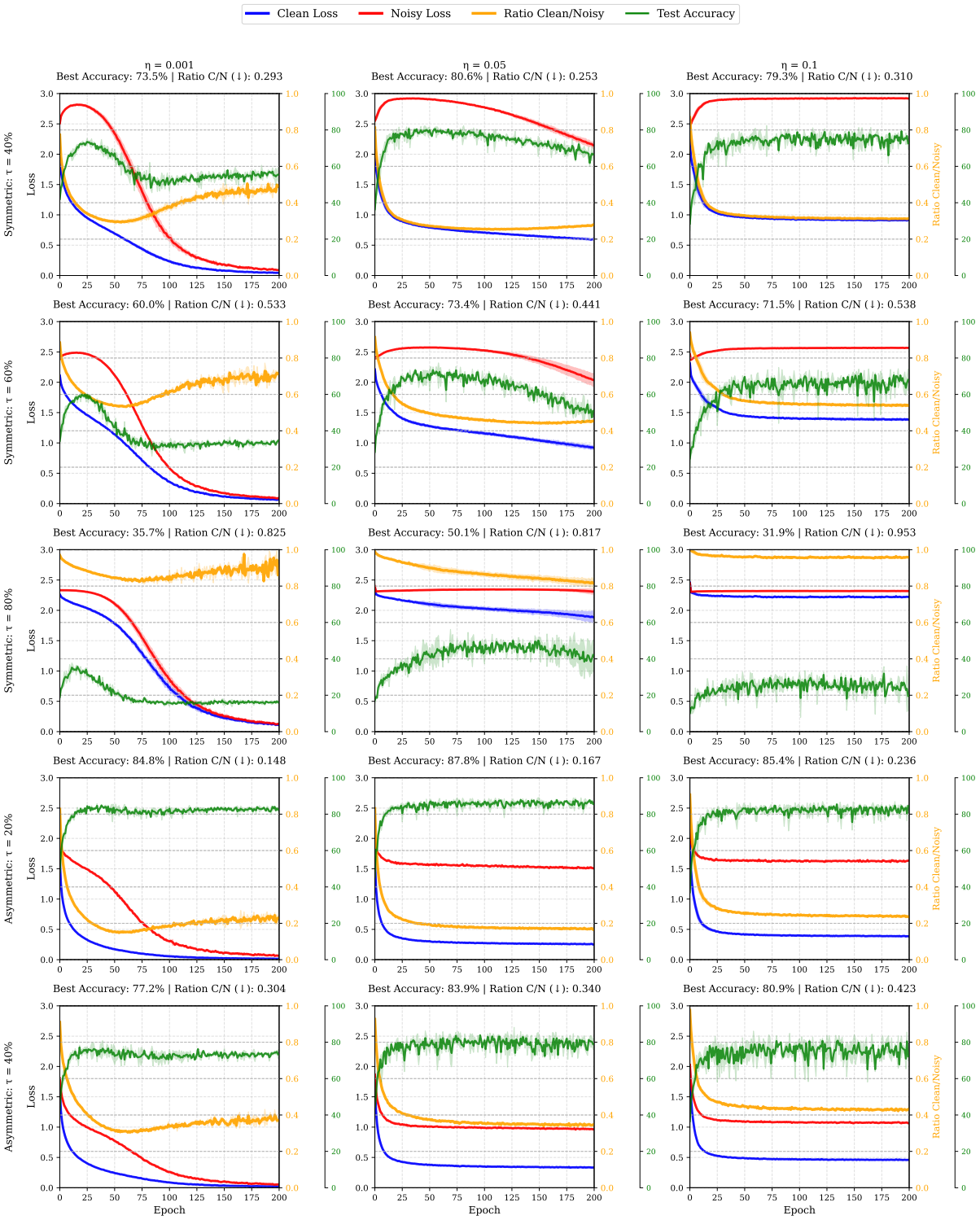


Figure 4 – Temporal evolution of losses and the Clean/Noisy ratio throughout training.

The analysis reveals important patterns about learning dynamics:

- Emergence dynamics of discriminative capacity: it is observed that different learning rates present distinct trajectories for developing discriminative capacity. For low learning rates

($\eta = 0.001$), gradual emergence of discrimination is observed, with the Clean/Noisy ratio evolving slowly but consistently. In contrast, intermediate rates ($\eta = 0.05$) demonstrate an ability to establish discrimination more rapidly in the initial epochs;

- Stability vs. deterioration: the stability of the ratio over epochs varies with η. While low rates tend to maintain or even gradually improve their discriminative capacity throughout training, very high rates ($\eta = 0.1$) frequently exhibit instability, with oscillations in the Clean/Noisy ratio that may indicate convergence difficulties or memorization tendencies. Notably, these high rates often demonstrate promising initial performance that deteriorates over time, suggesting that the duration of exposure to different η values may be as important as the value itself;

- Temporal correlation between discrimination and performance: in various scenarios, we observe that configurations that maintain a low Clean/Noisy ratio over time also present higher and more stable test accuracy. For example, $\eta = 0.05$ under 20% asymmetric noise simultaneously achieves the best accuracies and lowest loss ratio, suggesting that sustained capacity to distinguish clean from noisy examples is fundamental for generalization;

- Distinct learning phases: different phases in the learning process are observed. In scenarios with severe symmetric noise (60-80%), an initial phase is observed where all configurations show discriminative capacity, followed by a phase where only certain learning rates manage to maintain this capacity. This pattern suggests that the regularizing effect of learning rate becomes more critical as training progresses;

- Influence of noise nature: the type of noise also influences the dynamics. In the case of asymmetric noise, the Clean/Noisy ratio tends to remain more stable over time, while for symmetric noise, it shows greater sensitivity to the choice of η and the number of epochs.

The joint analysis of static and temporal perspectives extends the findings of Tanaka et al. (2018), suggesting that the effectiveness of η as an implicit regularizer depends not only on its absolute value, but also on its capacity to sustain discrimination over time.

In particular, rates that seem promising in the global analysis (see Figure 3) may deteriorate as epochs advance, while apparently less favorable configurations show greater temporal robustness. This reveals that intermediate values better balance discriminative learning, while extremes compromise robustness, especially under high noise.

Furthermore, the duration of exposure to the learning rate proves to be as decisive as its value. For example, $\eta = 0.1$ shows high initial capacity, but decays over time, whereas $\eta = 0.05$ maintains more stable performance. This observation raises the hypothesis that decay strategies—combining optimized high initial learning rates with the transition to lower values—could leverage the advantages of both regimes.

In practice, this issue may be even more critical for vanilla models, which, unlike specialized LNL methods, generally do not incorporate specific mechanisms designed to mitigate overfitting in the presence of noisy labels. For these models, the learning rate represents one of the few implicit regularization mechanisms available, along with choices such as batch size and training duration.

3.2.2 Impact of learning rate scheduling on vanilla models

In the training of DNNs in the LNL context, as discussed previously, the learning rate plays a fundamental role in both robustness and final model performance. Methods proposed in the LNL literature typically employ specific scheduling strategies for η, such as linear decay (HAN et al., 2018), step (LI; SOCHER; HOI, 2020), and cyclic variants (LI et al., 2018; LIU et al., 2020), carefully chosen to enhance the functioning of the proposed technique.

However, this scheduling choice is generally oriented toward tuning the results of the LNL method itself, and not necessarily optimized or suitable for the vanilla model used as a baseline (trained directly with noisy labels without any mitigation technique). In practice, this tends to introduce bias in experimental comparisons, since the baseline performance may be underestimated due to the use of a learning rate schedule that does not favor it. To illustrate this issue, we experimentally evaluate how different learning rate scheduling policies impact the performance of vanilla models in scenarios with noisy labels.

3.2.2.1 Experimental setup

Experiments were conducted on the CIFAR-10 and CIFAR-100 datasets using the CNN-7 architecture. Labels were artificially corrupted following two noise types: symmetric (40% and 80%) and asymmetric (40%), according to the label generation procedure described in Section 3.2.1.1. Models were trained for 100 epochs using the SGD optimizer with momentum 0.9, weight decay of 5×10^{-4}, batch size of 64, maximum learning rate of 2×10^{-1}, and minimum learning rate of 10^{-4}.

We evaluated five learning rate schedulers: step, linear, exponential, cosine annealing, and cosine annealing with periodic restarts (SGDR). These choices reflect the most widely used and well-validated learning-rate strategies in the recent LNL literature (HAN et al., 2018; LI et al., 2018; LI; SOCHER; HOI, 2020). The learning rate profiles of these schedulers are shown in Figure 5. Each configuration was repeated three times with different random seeds to ensure statistical robustness. For validation and test metrics, we applied a simple moving average (SMA) with a window of 10 epochs to smooth fluctuations and better capture the general trend of model performance over training, reducing the impact of noisy measurements in individual evaluations.

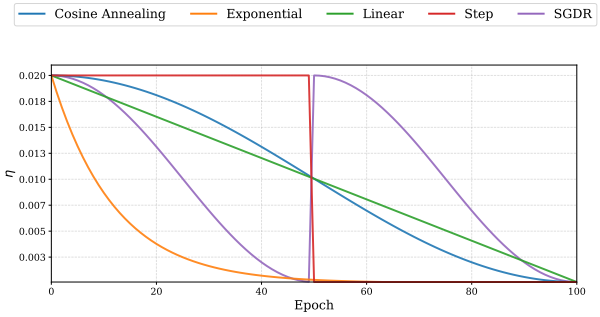


Figure 5 – Different learning rate scheduling policies evaluated in our experiments.

3.2.2.2 *Empirical results and observations*

The analysis of results illustrated in Figures 6 and 7, corresponding to the CIFAR-10 and CIFAR-100 datasets, shows that the step scheduler, despite its conceptual simplicity, presents superior performance in vanilla models when compared to more sophisticated strategies such as cosine annealing and SGDR. This superiority is consistently observed across both datasets: in CIFAR-10, for instance, step achieves the highest test accuracies under both symmetric and asymmetric noise (both with $\tau = 40\%$), while in CIFAR-100 it obtains the best results across all evaluated label noise scenarios.

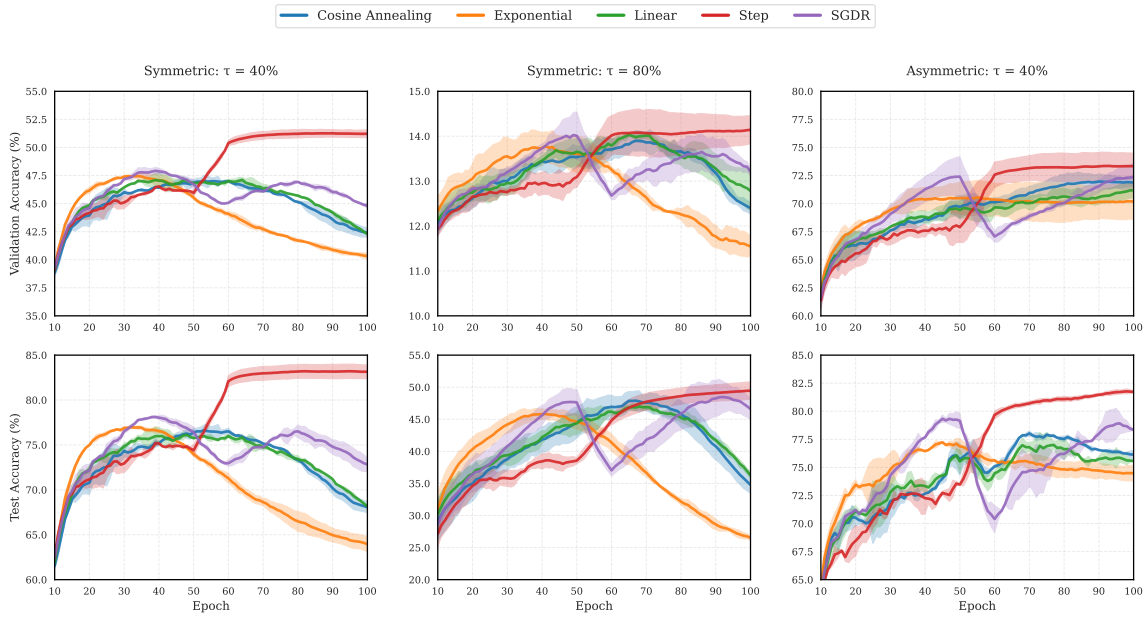


Figure 6 – Validation and test accuracy evolution of the vanilla CNN-7 model on CIFAR-10 under different label noise scenarios, comparing various learning rate schedulers. Curves are smoothed using SMA to reduce short-term fluctuations and highlight overall trends.

A particularly interesting aspect is that under severe noise conditions, such as in the symmetric scenario with $\tau = 80\%$, step maintains the vanilla model's performance relatively stable, while other schedulers suffer substantial drops. This observation is relevant because many specialized LNL methods adopt schedulers such as linear decay, cosine annealing, or SGDR, calibrated to maximize performance within their own architectures. However, applying

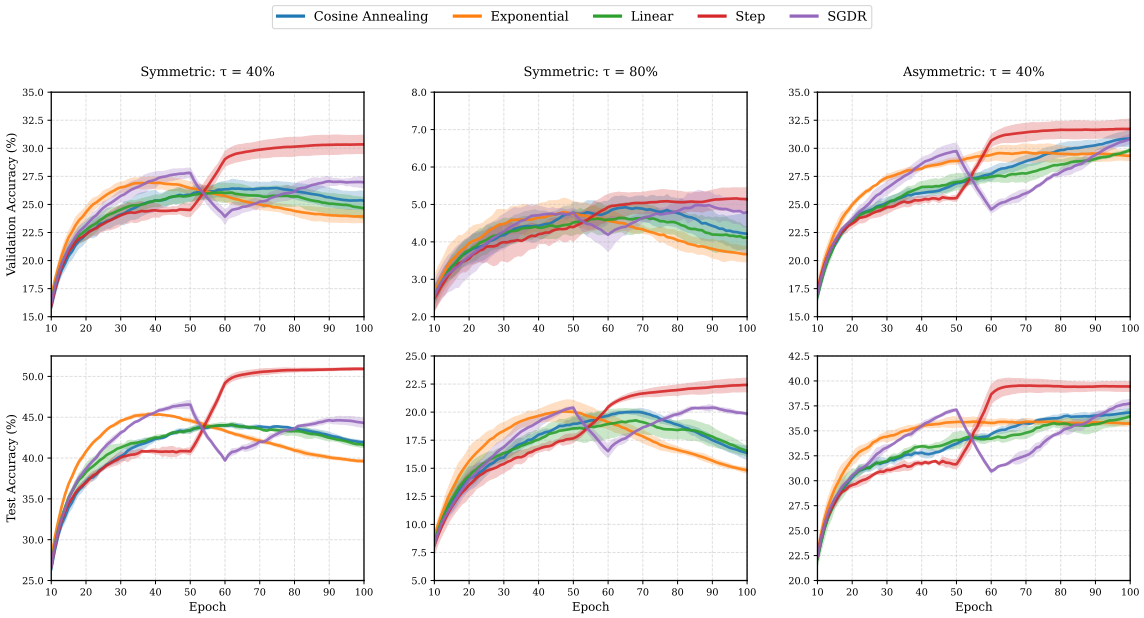


Figure 7 – Validation and test accuracy evolution of the vanilla CNN-7 model on CIFAR-100 under different label noise scenarios, comparing various learning rate schedulers. Curves are smoothed using SMA to reduce short-term fluctuations and highlight overall trends.

these same strategies to vanilla models may compromise their performance, introducing methodological bias in comparisons. Consequently, the supposed inferiority of vanilla models may be overestimated due to unfavorable configuration choices.

The effectiveness of Step can be attributed to its capacity to introduce a form of implicit temporal regularization, which makes it especially advantageous for vanilla models that lack specialized mechanisms to mitigate label noise effects. Abrupt learning rate reductions promote clear transitions between optimization phases: an initial phase with a high learning rate favors parameter space exploration, followed by a more conservative phase with a reduced rate, less prone to memorizing corrupted labels. This dynamic can be understood through the mechanism of "escaping" from problematic regions of the parameter space. Drastic reductions help the model escape from parameter space regions where it began memorizing noisy examples, forcing a change in the optimization regime. With the abrupt rate drop, the model loses the capacity to perform adjustments large enough to continue adapting to spurious patterns. In contrast, schedulers with smooth reductions keep the model longer in a "critical zone," where incremental adjustments still allow gradual noise assimilation through parametric drift.

This juxtaposition between conceptual simplicity and practical effectiveness raises important questions about the adequacy of scheduling strategies employed in comparative evaluations. While SGDR uses periodic restarts and cosine annealing applies a smooth decay, our results indicate that such sophisticated techniques do not always benefit vanilla models and, in some cases, may even impair their performance. For instance, exponential decay leads to an early degradation of learning capacity, whereas oscillatory strategies tend to reintroduce instabilities during critical convergence phases.

These findings highlight the importance of reevaluating recurring experimental prac-

tices in LNL literature, particularly regarding learning rate scheduler selection. For vanilla models, which serve as the starting point and essential baseline for comparisons, careful scheduler selection can not only improve performance but also promote fairer comparisons and reveal with greater precision the effective contributions of proposed specialized techniques.

While our analysis demonstrates the superiority of step scheduling for vanilla models under noisy conditions, several important questions remain open for future investigation. Specifically, our experiments employed fixed step parameters without systematic optimization of these choices. Key aspects warranting further exploration include: (1) the optimal timing for learning rate reduction (when to step), (2) the appropriate magnitude of reduction (how much to reduce), and (3) the interaction between step timing and different noise levels or dataset characteristics. These limitations present opportunities for more sophisticated optimization strategies. In Chapter 5, we address this question through univariate optimization that explores both optimal timing and magnitude for step scheduling.

4 CO-TEACHING AND CYCLIC TRAINING

In the previous chapter, we presented an overview of the main methods developed in the LNL literature to mitigate the negative impacts of label noise on the training of DNNs. Among these approaches, the Co-Teaching method stands out for its effectiveness and conceptual simplicity, as it combines the co-training strategy with loss-based sample selection. In this chapter, we explore the workings of Co-Teaching, discuss its limitations, and propose improvements through a cyclic training strategy.

4.1 CO-TEACHING FRAMEWORK

Empirical evidence suggests that low-loss samples are more likely to be correctly labeled (JIANG et al., 2018). Training a model predominantly on such instances within each mini-batch increases its robustness to label noise. Building on this assumption and extending the original co-training framework, Han et al. (2018) proposed the Co-Teaching method, in which two neural networks are trained simultaneously, each selecting a subset of small-loss instances from its mini-batch to update the other. A scheduling function $R(t)$, where t is the current training epoch, controls the proportion of selected samples over time, gradually reducing the influence of noisy data.

4.1.1 Algorithm description

Co-Teaching leverages the tendency of DNNs to learn simple, generalizable patterns before memorizing noisy labels (ARPIT et al., 2017). By focusing on low-loss samples, the method filters out likely mislabeled data, maximizing exposure to clean instances while minimizing the impact of incorrect labels.

The method employs two DNNs (denoted as networks 1 and 2), with parameters θ_1 and θ_2, trained simultaneously. Given a noisy dataset $\mathcal{D}$, at each training iteration, a mini-batch $\overline{\mathcal{D}}$ is sampled. Each network selects a fraction $R(t)$ of the samples in the mini-batch—those associated with the lowest loss values—forming the subsets $\overline{\mathcal{D}}_1$ and $\overline{\mathcal{D}}_2$. These presumably clean samples are then exchanged between the networks: each model updates its parameters using the instances selected by its peer. The overall procedure of the Co-Teaching method is detailed in Algorithm 1.

This mutual teaching mechanism is central to the effectiveness of Co-Teaching, as it reduces the risk of error confirmation caused by individual bias. Since the networks are initialized with different weights and follow distinct optimization trajectories, they tend to make mistakes on different instances throughout training. Consequently, cross-sample selection fosters diversity and complementarity, which helps mitigate the effects of label noise.

Algorithm 1: Co-Teaching algorithm

Input : Noisy training dataset $\mathcal{D}_{\text{train}}$
Initial weights θ_1, θ_2
Learning rate $\eta(t)$
Retention schedule $R(t)$
Number of epochs T
Batch size B

Output: Trained models θ_1, θ_2

for $t = 0$ **to** $T-1$ **do**
  Shuffle $\mathcal{D}_{\text{train}}$
  for $n = 1$ **to** $\left\lceil \frac{|\mathcal{D}_{train}|}{B} \right\rceil$ **do**
    Fetch n-th mini-batch $\overline{\mathcal{D}}$ from $\mathcal{D}_{\text{train}}$
    Select $\overline{\mathcal{D}}_1 = \arg\min_{\mathcal{D}':|\mathcal{D}'| \geq R(t)|\overline{\mathcal{D}}|} \ell(\theta_1; \mathcal{D}')$
    Select $\overline{\mathcal{D}}_2 = \arg\min_{\mathcal{D}':|\mathcal{D}'| \geq R(t)|\overline{\mathcal{D}}|} \ell(\theta_2; \mathcal{D}')$
    Update $\theta_1 \leftarrow \theta_1 - \eta(t) \nabla \ell(\theta_1; \overline{\mathcal{D}}_2)$
    Update $\theta_2 \leftarrow \theta_2 - \eta(t) \nabla \ell(\theta_2; \overline{\mathcal{D}}_1)$
  end
  Adjust retention rate $R(t)$
end
return θ_1, θ_2

4.1.2 Conventional retention strategy $R(t)$

DNNs, even when trained on datasets containing noisy labels, tend to first learn clean and simple patterns during the initial stages of training. In the original Co-Teaching method, the value of $R(t)$ is set higher at the beginning of training, allowing a greater number of low-loss samples to be retained in each mini-batch. As training progresses, $R(t)$ is gradually reduced to prevent the networks from overfitting noisy labels, thus favoring the retention of instances that are more likely to be correctly labeled and filtering out those potentially corrupted by noise.

In its standard implementation, Co-Teaching employs the linear decay variant of Equation 2.63 (with $\alpha = 1$), where t_k denotes the epoch at which $R(t)$ stops decreasing, and ϵ indicates the maximum fraction of samples to be ignored during training after epoch t_k.

This decreasing retention policy reflects the premise that, during the early stages of training, it is beneficial to expose the networks to a broader set of data in order to establish robust representations. As training progresses, the policy becomes more selective, aiming to prevent the memorization of noisy patterns. After epoch t_k, $R(t)$ remains constant at $1 - \epsilon$, operating under the assumption that this value approximates the actual proportion of clean samples in the dataset.

4.1.3 Limitations of traditional Co-Teaching

4.1.3.1 Convergence issues

Although Co-Teaching has proven effective in mitigating the accumulation of errors during training, it does not fully resolve this issue. As noted by Yu et al. (2019), this is because, as training progresses, the two neural networks tend to converge, reaching a consensus in their predictions. This convergence reduces their ability to distinguish clean samples, leading to the memorization of noisy patterns and, consequently, a degradation in generalization.

This redundant selection leads to the formation of increasingly similar sets $\overline{\mathcal{D}}_1$ and $\overline{\mathcal{D}}_2$, not only in their clean samples but also in their shared errors. As a result, the parameters of the networks converge, their predictions become nearly identical, and the benefit of cross-teaching — in which one network corrects the other's mistakes — is lost. This reduces the diversity of perspectives that Co-Teaching aims to preserve, undermining its ability to filter noise and increasing the risk of overfitting.

Additionally, in this scenario, the monotonically decreasing learning rate (η), as proposed in the original method (HAN et al., 2018), often proves insufficient to overcome such limitations. With progressively smaller η values, the models may become trapped in local minima, limiting their ability to explore broader regions of the optimization space that could be reached with higher learning rates (TANAKA et al., 2018).

4.1.3.2 Weaknesses of fixed retention policy

The retention policy in which $R(t)$ gradually decreases until epoch t_k and remains constant thereafter has been widely adopted in the literature. However, this approach is not necessarily optimal (YANG et al., 2024). Setting a fixed value for R beyond a certain point imposes a fundamental trade-off in the training process: balancing the benefits of sample diversity against the risks of memorizing noisy labels:

- A high value of R exposes the models to many potentially noisy samples, whose patterns may be memorized. This leads to overfitting and compromises the generalization ability of the models;
- Conversely, a low R value limits the models' exposure to the necessary diversity of training data, restricting their ability to learn more complex and robust patterns from the data.

Additionally, this fixed retention policy does not account for variations in the model's susceptibility to memorizing noisy samples after epoch t_k. Nor does it consider the impact of hyperparameters — such as the learning rate — on the model's robustness to label noise.

4.2.1 Cyclic learning rate scheduling

As discussed in subsection 4.1.3.1, one of the main challenges faced by methods such as Co-Teaching is the gradual loss of diversity between models throughout training. Since monotonically decreasing learning rates often prove insufficient to mitigate this problem, a direct alternative, in this context, is to adopt a learning rate variation policy that increases, even if momentarily, the divergence between models.

However, defining this policy is not trivial, because, as evaluated in the previous chapter, the learning rate plays a complex role in LNL, directly influencing the robustness of the models. One of the main challenges lies in the delicate balance between two opposing effects: high rates tend to reduce noise memorization, but may compromise overall performance; while lower rates favor better performance, but increase the models' propensity to rapidly memorize incorrect labels. Finding the optimal balance point between these extremes is particularly challenging in the LNL context.

A strategy with potential to address these two aspects — promoting divergence between models and achieving better performance — is the adoption of cyclic learning rate scheduling, allowing controlled oscillations of η. This approach favors the exploration of different regions of the parameter space when η is high (helping to escape local minima and saddle points) and more stable convergence when η is low (SMITH, 2017). Among the cyclic strategies proposed, Stochastic Gradient Descent with Warm Restarts (SGDR) stands out for its simplicity and efficiency. SGDR employs a cosine annealing schedule with restarts to adjust η periodically:

$$\eta(t) = \eta_{\min} + \frac{1}{2}(\eta_{\max} - \eta_{\min})\left(1 + \cos\left(\frac{t \bmod T}{T}\pi\right)\right) \tag{4.1}$$

where $\eta_{\min}$ and $\eta_{\max}$ denote the minimum and maximum learning rates, respectively, t represents the current epoch, and T defines the cycle length between restarts. Fig. 8(a) illustrates this cyclic variation of η using SGDR, showing how the cosine annealing schedule smoothly transitions between learning rate extremes.

This cyclic approach combines periods of broad exploration of the parameter space (when η is high) with periods of refinement and convergence (when η is low). The periodic and sharp increases in η temporarily induce model divergence, a particularly beneficial characteristic for overcoming potential overfitting conditions in the Co-Teaching framework. This controlled divergence allows for more efficient exploration of the optimization space, ultimately leading to more robust models with greater generalization capacity.

4.2.2 Cyclic schedule for sample retention

As previously discussed, using a fixed retention rate R throughout training imposes a delicate trade-off between the risk of excessive exposure to samples with noisy labels and the

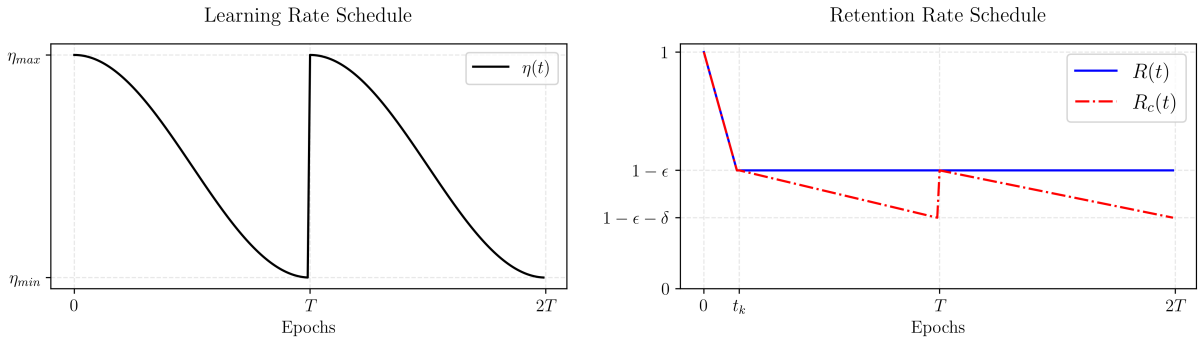


Figure 8 – a) Variation of the learning rate using the SGDR technique. b) Comparison between the retention rates $R(t)$ and $R_c(t)$.

limitation of sample diversity. To circumvent this limitation, we propose using a cyclic sample retention policy, synchronized with the cyclic scheduling of the learning rate η.

This approach exploits the fact that the robustness of models to noisy labels varies throughout the η cycle. Specifically, during training phases where the learning rate is high, models tend to exhibit greater robustness to noise. Under these circumstances, higher values of R can be employed, allowing models to explore a larger subset of the training set and learn more robust patterns from the data. Conversely, during phases with reduced learning rates, when solutions are refined, models become more susceptible to memorizing samples with noisy labels. To mitigate this effect, lower values of R can be used, directing models to focus on more reliable patterns and reducing the influence of potentially noisy samples.

$$R_c(t) = \begin{cases} 1-\epsilon\frac{t}{t_k}, & \text{if } t \in [0, t_k) \\ 1-\epsilon-\delta\frac{t-t_k}{T-t_k}, & \text{if } t \in [t_k, T) \\ 1-\epsilon-\delta\frac{(t \bmod T)}{T}, & \text{if } t \geq T \end{cases} \tag{4.2}$$

After epoch t_k, the retention values follow a cyclic pattern, oscillating between $1-\epsilon$ and $1-(\epsilon+\delta)$ every T epoch cycle, for the remainder of the training. Fig. 8 (b) illustrates the differences between the standard policy $R(t)$ and our cyclic retention proposal, $R_c(t)$.

4.2.3 Considerations on the synchronization between η and R

The synchronization between the learning rate $\eta(t)$ and the retention rate $R(t)$ is grounded in two main ideas:

- Robustness during high-variability phases: During exploration phases (high η), models exhibit increased robustness to noisy samples due to their enhanced ability to escape spurious local minima, allowing broader exposure to potentially corrupted data without compromising generalization;
- Susceptibility during convergence: During refinement phases (low η), models tend to

adjust their weights with higher precision, making them more susceptible to memorizing incorrect patterns. This calls for greater selectivity in the training samples.

Based on these observations, two main synchronization strategies between η and R can be naturally considered:

- In-phase synchronization: R is high when η is high and decreases as η decreases. This strategy builds on the assumption that the model's robustness at higher learning rates allows it to consider a larger number of samples, even in the presence of noise;
- Inverse synchronization: R is low when η is high and increases as η decreases. This approach adopts a more conservative stance during the early stages of exploration and a more inclusive one during refinement.

Cyclic synchronization provides a mechanism for balancing exploration and refinement. Furthermore, it introduces periodic moments of "rejuvenation" during training, where increases in η promote model divergence, and increases in R broaden the pool of considered samples. These cycles help preserve model diversity over time, mitigating the premature convergence often observed in traditional Co-Teaching approaches.

4.3 EXPERIMENTAL SETUP

We evaluated the effectiveness of cyclic variations in learning rate and retention rate within the Co-Teaching framework under noisy label conditions. Experiments were conducted on the CIFAR-10 dataset with two types of label noise: symmetric (30% and 60%) and asymmetric (20% and 40%). For the asymmetric noise settings, we followed the protocol proposed by Patrini et al. (2017).

We adopted the CNN-7 architecture and implemented three Co-Teaching variants:

- Traditional Co-Teaching (CoT): standard implementation with linearly decreasing learning and retention rates, following the original formulation;
- Co-Teaching using SGDR (CoT-S): hybrid implementation where only the learning rate follows a cyclic behavior via SGDR, while the retention policy maintains conventional behavior;
- Full Cyclic Co-Teaching (CoT-C): our approach combines SGDR with a cyclic retention policy, explored in two configurations:
  - In-phase synchronization (CoT-C-In phase): the maximum values of the retention rate R temporally coincide with the maximum values of the learning rate η;
  - Inverse synchronization (CoT-C-Inverse): the maximum values of the retention rate R coincide with the minimum values of the learning rate η.

All models were trained for 200 epochs using SGD with momentum 0.9, weight decay of 5×10^{-4}, batch sizes of 64, and an initial learning rate of 0.02. In the traditional setting, the learning rate decayed linearly starting from epoch 50. In the cyclic variants, η oscillated between 0.02 and 0.001 with a cycle period of $T = 50$ epochs, and the retention rate oscillated with an amplitude of $\delta = 0.1$ around the standard schedule starting from epoch t_k. To ensure statistical validity, all experiments were repeated three times with different random seeds. Table 4 summarizes the complete set of experimental parameters.

Parameter	**Values / Description**
Noise Types	Symmetric: 30% and 60%; Asymmetric: 20% and 40%
Co-Teaching Variants	CoT: linear learning and retention rates CoT-S: cyclic learning rate (SGDR), standard retention CoT-C: cyclic learning and retention (in-phase/inverse)
Training Setup	Optimizer: SGD (momentum 0.9, weight decay 5×10^{-4}) Initial learning rate ($\eta_{\max}$): 2×10^{-2} Batch sizes: 64 Epochs: 200
Cyclic Parameters	Cycle period T: 50 epochs Minimum learning rate $\eta_{\min}$: 10^{-3} Retention amplitude δ: 0.1 Standard retention start: $t_k = 10$, $\epsilon = \tau$
Validation Protocol	3 independent runs with different random seeds

Table 4 – Simplified summary of experimental parameters for evaluating cyclic strategies in Co-Teaching under noisy label conditions.

To investigate how our approach influences the convergence—or divergence—between the peer networks, we employ three complementary metrics to quantify different aspects of their evolution during training. Specifically:

- Prediction disagreement (PD): measures the fraction of samples for which the predictions of the two networks differ. Formally,

$$\mathrm{PD}(f_1, f_2) = \mathbb{E}_{\mathbf{x} \sim \mathcal{D}} \left[\mathbb{1}(\hat{y}_1 \neq \hat{y}_2) \right], \tag{4.3}$$

  where $\hat{y}_1 = \arg\max_k [f_1(\mathbf{x})]_k$ and $\hat{y}_2 = \arg\max_k [f_2(\mathbf{x})]_k$ are the predicted classes of networks f_1 and f_2, respectively. The indicator function $\mathbb{1}(\cdot)$ returns 1 if its argument is true, and 0 otherwise. PD ranges from 0 (complete agreement) to 1 (complete disagreement).

- Sample selection correlation (SC): quantifies the overlap between the training examples selected by each network, using the Jaccard index. Given selection masks $M_1, M_2 \subseteq \{1, 2, \ldots, |\overline{\mathcal{D}}|\}$ from networks f_1 and f_2, respectively, the correlation is computed as:

$$\mathrm{SC}(M_1, M_2) = \frac{|M_1 \cap M_2|}{|M_1 \cup M_2|}. \tag{4.4}$$

The score ranges from 0 (no overlap) to 1 (complete overlap), indicating how much both networks agree on the examples chosen for training.

4.4 RESULTS

4.4.1 Generalization

Table 5 presents the test accuracies achieved by different Co-Teaching variants under various noise conditions. The results demonstrate that our proposed cyclic approaches consistently outperform the traditional Co-Teaching baseline across all evaluated scenarios.

Among the tested configurations, the CoT-C-In phase variant achieves the highest performance in all settings. Notably, the performance gains are more pronounced at higher noise levels, suggesting that the cyclic approach is particularly effective when handling severely corrupted datasets.

Method	Symmetric		Asymmetric	
	30%	60%	20%	40%
CoT	88.22 ±0.16	81.87 ±0.22	89.43 ±0.16	85.25 ±0.14
CoT-S	88.67 ±0.22	82.72 ±0.43	89.64 ±0.26	85.73 ±0.18
CoT-C-Inverse	89.87 ±0.01	84.53 ±0.01	90.88 ±0.02	87.17 ±0.45
CoT-C-In phase	**90.22** ±0.07	**85.17** ±0.20	**91.28** ±0.08	**87.69** ±0.29

Table 5 – Test accuracies (%) for evaluating cyclic strategies in Co-Teaching under noisy label conditions. We report the mean ± standard deviation of the best accuracies over 3 runs for each method.

Both cyclic variants (CoT-C-Inverse and CoT-C-In phase) demonstrate superior robustness compared to the baseline, with the in-phase synchronization strategy yielding the greatest improvements. The CoT-S variant, which applies only cyclic learning rates while keeping a standard retention policy, shows modest gains over traditional Co-Teaching. This indicates that the combination of cyclic learning rates and cyclic retention schedules is responsible for the enhanced performance.

These improvements can be further analyzed by examining network behavior during training, as detailed in the following divergence analysis.

4.4.2 Divergence analysis

4.4.2.1 Prediction disagreement

Figure 9 illustrates the different patterns of prediction disagreement between models over time, under various types and levels of label noise. In the case of high symmetric noise (60%), we observe that traditional Co-Teaching tends to exhibit a gradual increase in disagreement, reaching values that remain elevated throughout the training epochs. Under moderate

asymmetric noise (40%), however, disagreement decreases more progressively, although there is still no clear indication of spontaneous reconvergence between the models.

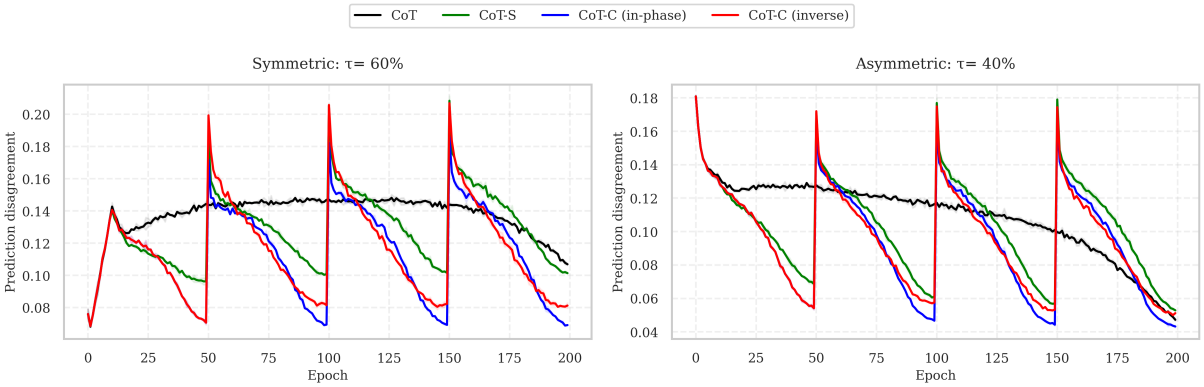


Figure 9 – Evolution of prediction disagreement on CIFAR-10 for cyclic strategies in Co-Teaching, under label noise conditions.

In contrast, the cyclical variants display a characteristic oscillatory pattern, with abrupt peaks (moments of increased disagreement) followed by gradual declines (periods of greater agreement), which occur in synchrony with the learning rate cycles. This periodic behavior suggests that these methods induce controlled phases of divergence, followed by reconciliation phases between the networks.

In practice, a certain level of disagreement between the models is beneficial for developing complementary perspectives, while extremes can be detrimental. Very low disagreement is not necessarily desirable—it may indicate that the networks are becoming replicas of each other, thus losing the benefits of collaborative training. On the other hand, very high disagreement may signal that the networks are following completely different paths. In this sense, the cyclical pattern creates "divergence moments" that encourage exploration of new regions in the data space, followed by convergence phases that consolidate the acquired knowledge. This dynamic balance contrasts with the behavior observed in traditional Co-Teaching, where the models gradually diverge without a clear mechanism for reconvergence.

4.4.2.2 Sample selection correlation

Figure 10 shows the evolution of the correlation between the sets of samples selected by the partner networks under different scenarios (types and levels) of label noise. Initially, all methods exhibit high correlation values due to the retention rate being close to 100% during the early epochs — at this stage, virtually all samples are used in the co-training process, resulting in high overlap between the sets selected by each network. Subsequently, a sharp drop is observed, occurring in parallel with the rapid reduction of the retention rate.

In traditional Co-Teaching, the correlation stabilizes at an intermediate level, with a slow downward trend throughout training. This early stabilization — a consequence of a constant retention rate and a monotonically and slowly decreasing learning rate — suggests a

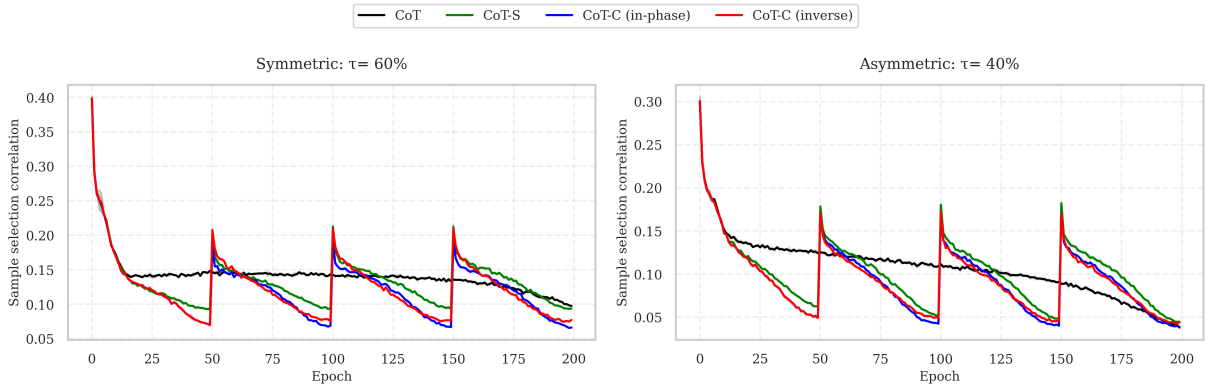


Figure 10 – Evolution of sample selection correlation on CIFAR-10 for cyclic strategies in Co-Teaching, under label noise conditions.

possible reduction in model adaptability, potentially limiting the diversity of examples selected collaboratively.

In contrast, the cyclic variants introduce an oscillatory dynamic in the correlation between the networks. In each cycle, we observe a peak in correlation followed by a gradual decline. These peaks occur when the learning rate rises — moments in which the networks tend to agree more on which examples are trustworthy, resulting in greater overlap. Such moments represent higher consensus, functioning as consolidation phases in which both networks focus on the most evidently clean examples. The decline in correlation reflects periods of progressive specialization, during which the networks develop complementary perspectives and divide the responsibility of filtering different regions of the data space.

Notably, even when only the learning rate is cyclic and the retention rate remains constant, the correlation curve already exhibits this pulsating behavior — alternating between phases of higher and lower agreement between the networks. However, when the retention rate is also made cyclic, the minimum values reached by the correlation become lower, indicating that the networks achieve deeper specialization during disagreement phases. This suggests that the combination of cycles in both rates enhances the diversity of the selected sets and increases the complementarity between the models.

This rhythmic alternation between phases of higher consensus and deeper specialization is driven by the cyclic combination of both rates. In the in-phase mode, the peaks of the retention rate coincide with the peaks of the learning rate. This synchronization favors moments when the model is more prone to learning while also using a broader set of trustworthy examples — creating well-defined cycles of consolidation and specialization, which contribute to the superior performance observed.

In the inverse mode, the peaks of the retention rate coincide with the troughs of the learning rate. In this configuration, the model focuses on a larger number of examples when the learning rate is more conservative, which may mitigate the risks of overfitting but also reduces the impact of aggressive exploration phases. Thus, the alternation still exists, but the consensus-specialization dynamic unfolds in a less synchronized manner.

5 PROPOSED METHOD: CYCLIC CO-TEACHING

This chapter introduces CCT, a novel learning framework designed to enhance model robustness in the presence of noisy labels. Inspired by the classical Co-Teaching paradigm, CCT integrates cyclical variations in both the learning rate and the sample retention rate, aiming to foster more adaptive and resilient training dynamics. The motivations behind these choices were discussed in previous chapters; here, the focus shifts to the formal description of the proposed method and its operational architecture.

We begin with an overview of the framework's architecture, illustrated in Figure 11. Then, Section 5.2 presents the optimization procedure and the strategies used in its implementation.

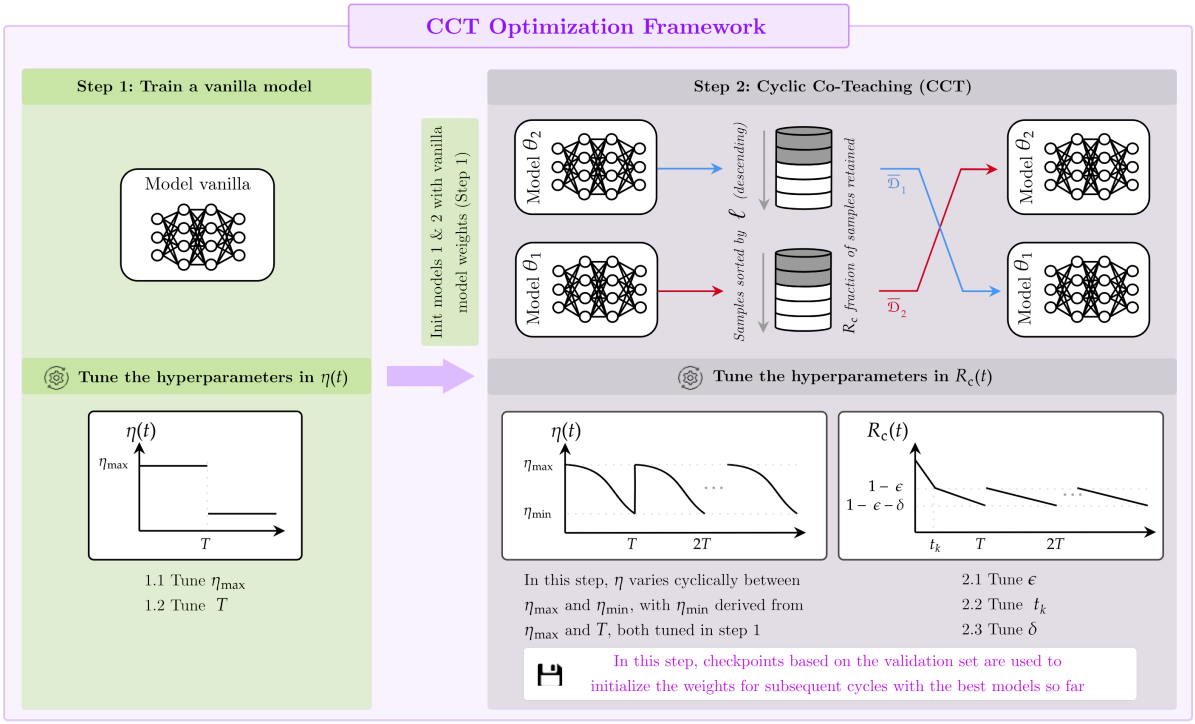


Figure 11 – Diagram of the CCT method and the proposed optimization framework. The CCT method trains two networks, θ_1 and θ_2, within a Co-Teaching structure that applies cyclical variations to both learning rates and sample retention rates. For each minibatch $\bar{\mathcal{D}}$, the networks select subsets with the lowest losses ($\bar{\mathcal{D}}_1$ and $\bar{\mathcal{D}}_2$) using the cyclical retention function R_c, which are then used by the opposite network to update its parameters. These cyclical adjustments in learning rates and retention rates promote deeper exploration and refinement, while a checkpoint mechanism ensures each new cycle resumes from the best weights found so far. The optimization framework is divided into two steps: in the first step, the learning rate hyperparameters η are tuned through conventional pre-training, resulting in an optimized vanilla model. In the second step, the weights of θ_1 and θ_2 in the co-training context are initialized from this model, and the learning rate varies cyclically based on the hyperparameters defined in the first step, while the hyperparameters of the cyclical retention function R_c are adjusted. The entire optimization process is carried out by monitoring the performance of the model(s) on a validation set $\mathcal{D}_{val}$.

5.1 CCT FRAMEWORK

The CCT method is formalized in Algorithm 2, where θ_i^* represents the optimal weights for model i, selected based on the highest validation accuracy achieved during training.

Algorithm 2: Cyclic Co-Teaching (CCT)

Input : Noisy dataset $\mathcal{D}_{\text{train}}$, validation dataset $\mathcal{D}_{\text{val}}$
Initialized weights θ_1, θ_2
Learning rate parameters $\eta_{\max}, \eta_{\min}$
Sample retention parameters ϵ, t_k, δ
Number of cycles C, epochs T, batch size B

Output: Best weights θ_1^*, θ_2^*

Initialize best weights: $\theta_1^* = \theta_1$, $\theta_2^* = \theta_2$
Initialize accuracies: $\text{acc}_1^* = 0$, $\text{acc}_2^* = 0$
for $c \leftarrow 1$ **to** C **do**
  $\theta_1 \leftarrow \theta_1^*$, $\theta_2 \leftarrow \theta_2^*$
  for $t \leftarrow 0$ **to** $T-1$ **do**
    Shuffle $\mathcal{D}_{\text{train}}$
    for $n \leftarrow 1$ **to** $\left\lceil \frac{|\mathcal{D}_{train}|}{B} \right\rceil$ **do**
      Fetch mini-batch $\bar{\mathcal{D}}$ from $\mathcal{D}_{\text{train}}$
      Select $\bar{\mathcal{D}}_1 = \arg\min_{\mathcal{D}'} \ell(\theta_1; \mathcal{D}')$
      Select $\bar{\mathcal{D}}_2 = \arg\min_{\mathcal{D}'} \ell(\theta_2; \mathcal{D}')$
      $\theta_1 \leftarrow \theta_1 - \eta(t) \nabla \ell(\theta_1; \bar{\mathcal{D}}_2)$
      $\theta_2 \leftarrow \theta_2 - \eta(t) \nabla \ell(\theta_2; \bar{\mathcal{D}}_1)$
    end
    Compute acc_1, acc_2 on $\mathcal{D}_{\text{val}}$
    if $acc_1 > acc_1^*$ **then**
      $\theta_1^* \leftarrow \theta_1, \text{acc}_1^* \leftarrow \text{acc}_1$
    end
    if $acc_2 > acc_2^*$ **then**
      $\theta_2^* \leftarrow \theta_2, \text{acc}_2^* \leftarrow \text{acc}_2$
    end
    Update $\eta(t)$ using Eq. (4.1);
    Update $R_c(t)$ using Eq. (4.2)
  end
end
return θ_1^*, θ_2^*

5.2 CCT OPTIMIZATION FRAMEWORK

This section presents a systematic two-step optimization framework for the CCT method. Step 1 identifies effective learning rate parameters using step scheduling and produces a well-tuned vanilla model. Step 2 builds on this foundation to optimize the CCT-specific cyclic retention parameters. Both phases employ univariate optimization, which is more efficient than

grid search. For example, for six variables with n possible values each, univariate optimization requires only $6n$ evaluations compared to n^6 for a full grid search.

Univariate optimization adjusts one hyperparameter at a time while keeping the others fixed at default or predefined values, observing the effect on model performance. This approach is simple to implement, computationally lightweight, and practical even when optimizing multiple hyperparameters.

5.2.1 Step 1: Vanilla model optimization with step scheduling

Building on the step scheduling methodology from Chapter 3, this phase identifies effective learning rate parameters and trains a robust vanilla model under noisy conditions. The optimization proceeds systematically: first, the best value for $\eta^*_{\max}$ is determined within a predefined range by training the model from scratch for a fixed number of epochs, varying $\eta_{\max}$ while keeping a constant learning rate in each run. The value yielding the highest average performance during the final training epochs is selected as $\eta^*_{\max}$.

Next, with $\eta^*_{\max}$ fixed, the best values of T^* and $\eta^*_{\min}$ are determined by retraining the model from scratch using $\eta^*_{\max}$ as the initial learning rate. The learning rate is reduced at epoch $t = T$ by a factor of $\frac{1}{T}$, resulting in $\eta_{\min} = \eta^*_{\max}/T$. The T value that provides the best average validation performance is chosen as T^*, and $\eta^*_{\min} = \eta^*_{\max}/T^*$.

This approach differs from existing LNL methodologies in several key aspects: unlike approaches that rely on large auxiliary datasets (HENDRYCKS; LEE; MAZEIKA, 2019), self-supervised techniques (TAN et al., 2021; ZHELTONOZHSKII et al., 2022), or clean validation sets, our optimization operates directly with potentially noisy validation data. Additionally, while methods such as O2U-Net and CJC-Net use fixed learning rates and require clean validation sets, our univariate procedure demonstrates robust performance even when the validation set contains noisy labels.

This approach is simple to implement and consistent with standard deep neural network training practices, which do not assume prior knowledge of the appropriate learning rate. In LNL scenarios, such procedures are practically inevitable: strong reference models and reliable estimates of the initial learning rate are rarely available or clearly defined. In our case, this phase yields a well-adjusted vanilla model that not only provides initialization weights but also establishes the foundational cyclic learning rate parameters ($\eta^*_{\max}$, T^*, $\eta^*_{\min}$) for subsequent CCT training.

5.2.2 Step 2: Cyclic training with CCT hyperparameter tuning

With the vanilla model foundation established in Phase 1, this phase optimizes the CCT-specific cyclic retention parameters (ϵ, t_k, δ) through cyclic training using Algorithm 2. The optimization adjusts one retention parameter at a time while keeping the learning rate parameters ($\eta^*_{\max}$, T^*, $\eta^*_{\min}$) fixed.

Although step scheduling produced the highest empirical performance in the vanilla setting, it is important to note that the learning-rate behavior required by CCT is fundamentally different from that of standard single-model training. CCT relies on maintaining controlled disagreement between the two networks so that the sample-exchange mechanism remains effective throughout training. In this context, the periodic warm-restart phases of SGDR are particularly advantageous: temporary increases in the learning rate periodically amplify model divergence, counteracting premature agreement and strengthening the filtering dynamics that drive the method. This explains why SGDR becomes more suitable for CCT, even in cases where step scheduling happens to yield the best results for the vanilla baseline.

The sequence of parameter optimization is as follows: First, determine ϵ, the baseline retention rate. If the noise rate τ is known, set $\epsilon = \tau$; otherwise, optimize ϵ with $t_k = 0$ and $\delta = 0$. Next, with ϵ fixed, optimize t_k (keeping $\delta = 0$) to decide when the cyclic retention pattern begins. Finally, optimize δ with ϵ and t_k fixed to set the amplitude of retention rate oscillations around ϵ.

Note that assigning $\epsilon = \tau$ applies only to the specific situation in which the noise rate is known. In more realistic scenarios — where τ is unknown or the validation set itself may contain noisy labels — ϵ is simply treated as a standard hyperparameter within the optimization process. The computational advantages of this univariate approach and the specific impact of optimizing ϵ on model performance are further discussed in subsection 7.3.

This phase leverages the vanilla model from Phase 1 to explore the CCT hyperparameter space systematically, producing the final optimized CCT model with both effective learning rate schedules and adaptive sample retention strategies.

The complete optimization framework is in Algorithm 3.

Algorithm 3: CCT optimization framework

Input : Noisy training dataset $\mathcal{D}_{\text{train}}$
Validation dataset $\mathcal{D}_{\text{val}}$
Learning rate range $\mathcal{L} \subseteq \mathbb{R}$
Cycle length range $\mathcal{T} \subseteq \mathbb{Z}$
Cyclical sample retention rate hyperparameter set: $\mathcal{E} \subseteq \mathbb{R}$, $\mathcal{T}_k \subseteq \mathbb{Z}$, $\mathcal{F} \subseteq \mathbb{R}$
Number of cycles C, batch size B

Output: Optimized weights θ_1^*, θ_2^* and hyperparameters $\eta_{\max}^*$, T^*, ϵ^*, t_k^*, δ^*

Step 1: Optimize the hyperparameters for η and train a vanilla model
foreach $\eta_{\max} \in \mathcal{L}$ **do**
  Train the DNN θ_1 from scratch on $\mathcal{D}_{\text{train}}$ using $\eta_{\max}$ for $T_{\max} = \max(\mathcal{T})$ epochs
  Evaluate on $\mathcal{D}_{\text{val}}$
end
Set $\eta_{\max}^*$ as the value with best validation accuracy
foreach $T \in \mathcal{T}$ **do**
  Train DNN θ_1 from scratch on $\mathcal{D}_{\text{train}}$ with $\eta_{\max}^*$ for T epochs
  Continue training with learning rate $\eta_{\max}^*/T$ for $2T_{\max} - T$ epochs
  Save best checkpoint on validation set
end
Set T^* and θ_1^* from best validation accuracy
Set $\eta_{\min} = \eta_{\max}^*/T^*$
Step 2: Optimize R_c hyperparameters and train with CCT
foreach $\epsilon \in \mathcal{E}$ **do**
  Set $t_k = 0$, $\delta = 0$
  Run Algorithm 2
end
Set ϵ^* as the ϵ with the best average validation accuracy
foreach $t_k \in \mathcal{T}_k$ **do**
  Set $\epsilon = \epsilon^*$, $\delta = 0$
  Run Algorithm 2
end
Set t_k^* as the t_k with the best average validation accuracy
foreach $\delta \in \mathcal{F}$ **do**
  Set $\epsilon = \epsilon^*$, $t_k = t_k^*$
  Run Algorithm 2
end
Set δ^* as the best value and θ_1^*, θ_2^* as the corresponding models from Algorithm 2
return θ_1^*, θ_2^* *and tuned learning rate and retention parameters*

6 EXPERIMENTAL SETUP

This section provides a detailed overview of the evaluation methodology and experimental settings used in our investigations.

6.1 EVALUATION AND TUNING METHODOLOGY

Following the methodological guidelines established in Chapter 3, our experimental evaluation adopts the conventional ML methodology to ensure unbiased model assessment and fair comparisons. This approach addresses the problematic practices identified in current LNL literature, implementing the following experimental protocols:

- **Validation set**: used to monitor model performance during training and to tune hyperparameters. When a clean validation set is not available, it is common practice to extract a subset of the noisy training set for this purpose, even though this introduces potential biases;
- **Stopping criterion**: since the ideal number of training epochs is difficult to determine, we adopt the early stopping technique, halting training once the validation performance no longer improves. This strategy prevents overfitting to noisy labels and avoids unnecessary computational cost;
- **Best model selection**: we report test set results for the model checkpoint that achieved the best validation performance during training.

For datasets without predefined validation sets, we adopt a 95%/5% train/validation split from the noisy training data, ensuring consistent evaluation protocols across all experiments.

6.2 EVALUATION AND LABEL CORRUPTION

Building on the above methodology, we assess our approach under label corruption by reporting test accuracy, as defined in Subsection 2.2.2. For hyperparameter optimization, we use validation accuracy to avoid data leakage and follow standard ML training practices. As commonly done in LNL works (HAN et al., 2018; YU et al., 2019; WEI et al., 2020; TAN et al., 2021), we report the average performance over the last 10 training epochs to account for training stability and provide a more robust evaluation.

For the synthetic datasets, since they are originally clean, we manually corrupt the labels using the label transition matrix T defined in Section 2.2.1. We adopt two representative structures for T: symmetric and asymmetric.

6.3 DATASETS

We use both synthetic datasets (CIFAR-10, CIFAR-100, and Tiny-ImageNet) and real-world datasets (Animal-10N, Food-101N, and Clothing1M) for our experiments. Note that for Tiny-ImageNet, Food-101N, and Clothing1M, a clean validation set is available, while for the remaining datasets, we extract a portion of the noisy training set to serve as a validation set.

- **CIFAR-10**[1]: a widely used dataset in computer vision consisting of 60K color images (50K for training and 10K for testing). Each image has dimensions of 32×32 pixels and belongs to one of the 10 distinct classes in the dataset. For simulation purposes, in addition to symmetric noise, we explored asymmetric noise following the methodology presented by Patrini et al. (2017), where noisy labels were generated through the mapping "truck" $\rightarrow$ "automobile", "bird" $\rightarrow$ "airplane", "deer" $\rightarrow$ "horse", and "cat" $\leftrightarrow$ "dog". It is worth noting that with this methodology for generating asymmetric noisy labels, only half of the classes in the dataset have noisy labels, and therefore, the actual noise rate in the entire dataset is half of the noise rate in the noisy classes. This means that when the asymmetric noise rate is 40%, this rate applies to the samples that had their labels swapped. Considering all samples (including all classes), the noise rate is 20%.

- **CIFAR-100**: similar to CIFAR-10, this dataset consists of 60K color images (50K for training and 10K for testing) with dimensions of 32×32 pixels. However, in this case, each image belongs to one of the 100 distinct classes in the dataset. Once again, in addition to simulating symmetric noise, we also included asymmetric noise, specifically adjacent noise, where the label of a class is replaced by the label of the subsequent class.

- **Tiny-ImageNet**[2]: is a smaller version (in terms of resolution and number of classes) of the ImageNet dataset, consisting of 120K color images (100K for training, 10K for validation, and 10K for testing) distributed across 200 classes with a resolution of 64×64 pixels.

- **Animal-10N**[3]: a dataset of online images labeled by humans for 5 pairs of animals with visually similar patterns, namely: (cat, lynx), (jaguar, cheetah), (wolf, coyote), (chimpanzee, orangutan), and (hamster, guinea pig). The dataset consists of 55K color images (50K for training and 5K for testing), each with dimensions of 64×64 pixels. Due to the ambiguous visual patterns of the images, labeling errors naturally occur, resulting in a dataset with approximately 8% label noise (SONG; KIM; LEE, 2019).

- **Food-101N**[4]: this dataset is designed for food classification, consisting of approximately 310K training images distributed across 101 categories, based on the Food-101 taxonomy

[1] https://www.cs.toronto.edu/~kriz/cifar.html
[2] https://www.kaggle.com/c/tiny-imagenet
[3] https://dm.kaist.ac.kr/datasets/animal-10n
[4] https://kuanghuei.github.io/Food-101N

(LEE et al., 2018). The images were collected from sources such as Google, Bing, Yelp, and TripAdvisor, with labels automatically assigned using keywords, leading to a label noise rate of around 20%. For evaluating the performance of models trained on this dataset, the test set from Food-101 is used, ensuring consistency between the categories in both datasets (BOSSARD; GUILLAUMIN; GOOL, 2014).

- **Clothing1M**: a large-scale dataset containing more than one million images distributed across 14 categories of clothing, including items such as shirts, sweaters, jackets, and dresses. The dataset was created from images obtained from online shopping websites, where labels were assigned based on keywords found in the surrounding text provided by sellers (XIAO et al., 2015). Due to the nature of this labeling process, a significant amount of noise is present in the labels, with an estimated noise rate of nearly 40% (LIANG; LIU; YAO, 2022). In addition to the noisy label samples, the dataset includes subsets with clean labels, comprising 50K training images, 14K validation images, and 10K test images. The complexity and irregularity of the label noise distribution, including symmetric, asymmetric, and random noise types, make Clothing1M a valuable dataset for studying LNL techniques.

Detailed information regarding the preprocessing operations applied to these datasets during the experiments is available in Appendix A.

6.4 BASELINES

For comparative purposes, we evaluate the performance of our method with the following state-of-the-art methods:

- **Vanilla (Standard)**: the DNN is trained directly on the noisy dataset without employing any LNL techniques;
- **Decoupling** (MALACH; SHALEV-SHWARTZ, 2017): involves the simultaneous training of two neural networks that are updated based on samples for which the predictions of the two models are different;
- **Co-Teaching** (HAN et al., 2018): also involves the simultaneous training of two neural networks, where each network selects only samples with small loss values to update the weights of the other network in the pair;
- **Co-Teaching+** (YU et al., 2019): employs a similar approach to Co-Teaching, but where the networks instruct each other only when there are discrepancies in predictions;
- **O2U-Net** (HUANG et al., 2019): oscillates a pre-trained network between overfitting and underfitting conditions through cyclic adjustment of the learning rate. During this

process, it records the normalized loss values of the samples for later identification of potentially mislabeled ones;

- **JoCoR** (WEI et al., 2020): based on the Co-Teaching strategy, employs a joint loss to make two networks agree with each other;
- **CJC-Net** (ZHANG et al., 2021): combines cyclic training method with joint loss and Co-Teaching strategies using pre-trained networks;
- **Co-learning** (TAN et al., 2021): trains a single encoder network with two heads (one dedicated to self-supervised training and the other to supervised training) that mutually restrict and aim to maximize agreement between them in the latent space;
- **TCC-Net** (XIA; LEE; CHEN, 2023): proposes a robust two-stage training, where in the first stage, neural networks are trained with a loss function robust to label noise, while in the second stage, Co-Teaching with meta-learning is employed.

6.5 TRAINING SETTINGS

6.5.1 Network structure and optimizer

In our experiments, we utilized five different architectures: a 7-layer CNN and a 9-layer CNN, whose details are provided in Appendix B, along with the ResNet-18, PreAct ResNet-18, and ResNet-34 neural networks. For the CIFAR-10 and CIFAR-100 datasets, we employed the 7-layer CNN, the 9-layer CNN, and the ResNet-18. For Tiny-ImageNet, we specifically used the PreAct ResNet-18 model. The Animal-10N dataset was trained using the 9-layer CNN, while Food-101N and Clothing1M were trained using the ResNet-34 and ResNet-18, respectively.

In all experiments, both for training the vanilla model and for training the CCT models, we used SGD as the optimization algorithm, with a momentum of 0.9 and a weight decay of 5×10^{-4}, except for the ResNet-18 training on the Clothing1M dataset, where we applied a weight decay of 10^{-3}. We also used a fixed mini-batch size of 64 and employed the traditional CE as the loss function for training the DNNs.

6.5.2 CCT method training

The vanilla model is trained without specific LNL techniques. As described in subsection 5.2, we first determine the optimal value of $\eta_{\max}$. For each $\eta_{\max}$ value, we train the model from scratch for 100 epochs with a fixed learning rate. The selected optimal value $\eta^*_{\max}$ is the one that yields the best average performance on the validation set during the last 10 epochs of training.

To determine the optimal value of T, we retrain the vanilla model from scratch using $\eta^*_{\max}$ as the initial learning rate, which is then reduced by a factor of $\frac{1}{T}$ at epoch $t = T$. For each

value of T, training continues for up to 200 epochs, with checkpoints saving the model state that exhibits the best validation set performance. Training is terminated if no improvement is observed after 20 consecutive epochs. The optimal value T^* is selected based on the best average validation set performance, and the minimum learning rate is set as $\eta_{\min} = \eta^*_{\max}/T^*$.

For the cyclic training phase, all models are initialized using weights from the best vanilla models previously obtained. During SGDR implementation, we employ the optimized values of $\eta_{\max}$, T, and $\eta_{\min}$. For the $R_c(t)$ curve, following Co-Teaching-based approaches (YU et al., 2019; WEI et al., 2020; ZHANG et al., 2021), we assume prior knowledge of τ and set $\epsilon = \tau$, where t_k and δ are the hyperparameters to be optimized. Training continues for a maximum of 8 cycles or until the average validation accuracy shows no improvement over the course of one cycle.

Table 6 summarizes the CCT hyperparameters, their search ranges, and the relationships adopted for most datasets. Specifically, for the Clothing1M dataset, we used a different search range for $\eta_{\max}$: $[0.001, 0.006]$ with step size 0.001. This narrower range reflects the characteristics of Clothing1M: its high noise rate, large number of classes, and very large dataset make early training sensitive to aggressive learning-rate peaks, causing instability when $\eta_{\max}$ is too large. For new datasets, a simple guideline is to run a small sweep of candidate upper bounds and select the largest value that keeps the early validation loss stable.

Step	Hyperparameter	Search Range/Relationships
Vanilla	$\eta_{\max}$	$[0.02, 0.025, 0.030, \cdots, 0.045, 0.05]$
	T	$[50, 75, 100]$
	$\eta_{\min}$	$\eta^*_{\min} = \eta^*_{\max}/T^*$
Cyclic	ϵ	$\epsilon = \tau$
	t_k	$[0, 5, 10]$
	δ	$[0, 0.025, 0.05, 0.075, 0.1]$

Table 6 – Parameter search spaces and relations in CCT optimization framework

For all experiments, results are computed using a 10-epoch simple moving average, with each experiment repeated five times with different random seeds. The reported performance metrics represent the mean and standard deviation of these moving averages across the five runs.

6.5.3 Exceptions

For Food-101N and Clothing1M, all previous mentions of "epochs" are replaced by "mini-epochs", where a mini-epoch is defined as some fraction of an epoch. This change is made to ensure there are sufficient updates of the cyclic rates in the CCT method, allowing

the models to converge adequately. Specifically, a mini-epoch is defined as 1/2 and 1/20 of an epoch for Food-101N and Clothing1M, respectively.

6.6 CODE AVAILABILITY AND REPRODUCIBILITY

All source code, training scripts, configuration files, and the complete implementation of the CCT method used in this thesis are publicly available at:

https://github.com/Kamassury/CCT

This repository includes all experiments, hyperparameter tuning routines, dataset preprocessing pipelines, baseline configurations, and evaluation protocols required for the full reproducibility of the results presented in Chapter 7.

7 RESULTS AND ANALYSIS

In this section, we compare and discuss the performance achieved by the CCT method relative to the baselines described in subsection 6.4. Additionally, we compare it with an optimized vanilla model and a modified version of the Co-Teaching method. Specifically, the optimized vanilla model represents the model that achieves the best performance through optimization solely by varying the learning rate (using the step decay policy) and applying the early stopping strategy, without any LNL technique interventions. The Enhanced Co-Teaching is a modified version of the original Co-Teaching method, distinguished by its use of SGD as the optimization algorithm, step decay learning rate policy (as in the vanilla optimized), and the early stopping strategy.

It is important to note that the methods compared in each table utilize the same neural network architecture, as specified in the corresponding legend. For example, when results are indicated to have been obtained using a 7-layer CNN, this implies that the results of all other methods listed in the table were also achieved using this neural network architecture.

To ensure a fair comparison, we present the baseline results exactly in the format used in their original publications. Although all methods perform multiple runs, some authors report only the mean, while others provide the mean and standard deviation; our tables preserve these formats. In the case of CCT, to maintain compatibility with the reference works, we report the mean and standard deviation only in scenarios where the compared methods also provide this information.

Furthermore, to preserve fairness, we do not re-tune the hyperparameters of baseline methods; each baseline is evaluated using the configuration recommended by its original publication. This avoids unintended advantages and ensures consistency with prior LNL comparisons.

7.1 COMPARISON WITH STATE-OF-THE-ART METHODS

7.1.1 CIFAR-10

The experimental results for the CIFAR-10 dataset, comparing various neural architectures (7-layer CNN, 9-layer CNN, and ResNet-18), label noise types, and noise levels, are presented in Tables 7, 8, and 9.

For the CIFAR-10 dataset, except in some scenarios with $\tau \geq 50\%$ symmetric noise for the 7-layer and 9-layer CNN architectures, the CCT method consistently demonstrated superior performance compared to state-of-the-art methods. Notably, the most significant gains were observed with the 7-layer CNN in scenarios with 80% symmetric noise and 40% asymmetric noise, where improvements of nearly $31pp$[1] and $11pp$ in test accuracy, respectively,

[1] *pp* denotes percentage points.

were achieved. Additionally, a substantial gain of approximately $9pp$ was observed for CCT compared to Co-Learning in the 40% asymmetric noise condition using ResNet-18.

Dataset	**CIFAR-10**			
Noise type	**Symmetric**			**Asymmetric**
Method/Noise ratio	20%	50%	80%	40%
Vanilla (ZHANG et al., 2021)	69.18 ±0.52	42.71 ±0.42	16.24 ±0.39	69.43 ±0.33
Decoupling (MALACH; SHALEV-SHWARTZ, 2017)	79.32 ±0.40	40.22 ±0.30	15.31 ±0.43	68.72 ±0.30
Co-Teaching (HAN et al., 2018)	78.23 ±0.27	71.30 ±0.13	26.58 ±2.22	73.78 ±0.22
Co-Teaching+ (YU et al., 2019)	78.71 ±0.34	57.05 ±0.54	24.19 ±2.74	68.84 ±0.20
JoCoR (WEI et al., 2020)	85.73 ±0.19	79.41 ±0.25	27.78 ±3.06	76.36 ±0.49
CJC-Net (ZHANG et al., 2021)	90.79 ±0.09	**88.51** ±0.05	29.21 ±0.11	71.61 ±0.15
Optimized Vanilla	88.20 ±0.02	81.93 ±0.28	54.83 ±2.33	84.27 ±0.24
Enhanced Co-Teaching	90.45 ±0.05	85.90 ±0.24	25.27 ±1.93	85.18 ±0.21
CCT	**91.44** ±0.13	88.29 ±0.19	**60.20** ±2.18	**87.39** ±0.12

Table 7 – Test accuracies (%) on CIFAR-10 dataset under various noise conditions using a 7-layer CNN architecture. Baseline results are taken directly from Zhang et al. (2021). Reported values for our methods correspond to the mean ± standard deviation over 5 runs using a 10-epoch simple moving average. Best results are shown in bold.

Dataset	**CIFAR-10**							
Noise type	**Symmetric**				**Asymmetric**			
Method/Noise ratio	20%	40%	50%	80%	10%	20%	30%	40%
Vanilla (ZHANG et al., 2021)	76.42	56.08	–	17.67	83.83	–	–	–
Co-Teaching (HAN et al., 2018)	82.66	77.42	–	22.60	85.83	–	–	–
O2U-Net (HUANG et al., 2019)	85.24	79.64	–	34.93	88.22	–	–	–
CJC-Net (ZHANG et al., 2021)	92.41	90.13	–	36.85	92.87	–	–	–
TCC-Net (XIA; LEE; CHEN, 2023)	91.94	–	**89.27**	**73.79**	–	91.17	89.23	82.71
Optimized Vanilla	89.59	86.61	83.64	56.85	92.35	91.48	90.07	87.12
Enhanced Co-Teaching	90.90	88.01	86.32	24.63	91.32	90.36	88.35	69.06
CCT	**92.49**	**90.28**	88.96	60.55	**93.08**	**92.33**	**91.32**	**87.85**

Table 8 – Test accuracies (%) on CIFAR-10 dataset under various noise conditions using a 9-layer CNN architecture. Baseline results are taken directly from Zhang et al. (2021) and Xia, Lee & Chen (2023). Reported values for our methods correspond to the mean over 5 runs using a 10-epoch simple moving average. Best results are shown in bold.

For the CIFAR-10 dataset, except in some scenarios with $\tau \geq 50\%$ symmetric noise for the 7-layer and 9-layer CNN architectures, the CCT method consistently demonstrated superior performance compared to state-of-the-art methods. Notably, the most significant gains were observed with the 7-layer CNN in scenarios with 80% symmetric noise and 40% asymmetric noise, where improvements of nearly $31pp$[2] and $11pp$ in test accuracy, respectively, were achieved. Additionally, a substantial gain of approximately $9pp$ was observed for CCT compared to Co-Learning in the 40% asymmetric noise condition using ResNet-18.

Furthermore, across all architectures, levels, and noise types, the optimized vanilla models consistently outperformed the vanilla models reported in the literature. Although the

[2] *pp* denotes percentage points.

Dataset	CIFAR-10				
Noise type	**Symmetric**			**Asymmetric**	
Method/Noise ratio	20%	50%	80%	30%	40%
Vanilla (TAN et al., 2021)	84.81 ±0.24	61.49 ±0.58	28.98 ±0.26	81.99 ±0.31	76.30 ±0.34
Decoupling (MALACH; SHALEV-SHWARTZ, 2017)	85.75 ±0.31	61.93 ±0.82	27.23 ±0.84	81.83 ±0.26	74.97 ±0.38
Co-Teaching (HAN et al., 2018)	90.29 ±0.19	63.45 ±3.89	28.03 ±1.67	86.58 ±1.32	74.25 ±0.38
Co-Teaching+ (YU et al., 2019)	88.63 ±0.32	76.27 ±2.80	30.37 ±1.69	86.22 ±0.26	81.25 ±0.75
JoCoR (WEI et al., 2020)	90.43 ±0.25	66.00 ±0.53	29.19 ±1.64	86.41 ±0.45	73.95 ±1.00
Co-learning (TAN et al., 2021)	92.21 ±0.31	84.49 ±0.34	61.20 ±2.29	86.89 ±0.87	81.42 ±0.52
Optimized Vanilla	91.96 ±0.36	86.48 ±0.17	57.86 ±0.32	92.07 ±0.16	89.93 ±0.18
Enhanced Co-Teaching	92.89 ±0.07	87.82 ±0.32	26.27 ±2.47	91.04 ±0.32	81.57 ±0.08
CCT	**93.80** ±0.03	**89.36** ±0.15	**63.88** ±2.76	**92.62** ±0.08	**91.27** ±0.21

Table 9 – Test accuracies (%) on CIFAR-10 dataset with different types and levels of label noise obtained using ResNet-18. The results of the other methods are taken directly from Tan et al. (2021). Reported values for our methods correspond to the mean and standard deviation over 5 runs using a 10-epoch simple moving average. The best result in each case is highlighted in bold.

Enhanced Co-Teaching surpassed the optimized vanilla models in many cases, this was not always true for scenarios with very high noise rates. This finding highlights the importance of establishing robust baselines with optimized vanilla models to ensure a fair comparison with methods designed specifically for handling noisy labels.

7.1.2 CIFAR-100

For CIFAR-100, across all tested configurations, the CCT method predominantly achieved the best results, with the most notable performance gains in scenarios with 80% symmetric noise for the 7-layer (Table 10) and 9-layer CNNs (Table 11), and for the 40% asymmetric noise scenario using ResNet-18 (Table 12). Similar to the CIFAR-10 findings, optimized vanilla models showed significantly improved performance compared to vanilla models in the literature. For ResNet-18, these optimized vanilla models already exceeded all state-of-the-art results (Co-Learning), once again underscoring the critical importance of optimizing the baseline models.

In Appendix C, we present the simple moving average curves of test accuracy throughout the training process using ResNet-18 for both CIFAR-10 and CIFAR-100.

7.1.3 Tiny-ImageNet

Our experimental results for the Tiny-ImageNet dataset are detailed in Table 13, which also includes the performances of reference methods under different simulated conditions. It is observed that in all cases, the optimized vanilla models consistently outperform the results reported in the literature.

Furthermore, the superiority of the CCT performance is evident once again, with a performance gain of approximately $6.55pp$ for the case with 20% symmetric noise and about $8.68pp$ for the scenario with asymmetric noise.

Dataset	CIFAR-100			
Noise type	**Symmetric**			**Asymmetric**
Method/Noise ratio	20%	50%	80%	40%
Vanilla (ZHANG et al., 2021)	35.14 ±0.44	16.97 ±0.40	4.41 ±0.25	27.29 ±0.25
Decoupling (MALACH; SHALEV-SHWARTZ, 2017)	33.10 ±0.12	15.25 ±0.20	3.89 ±0.16	26.11 ±0.39
Co-Teaching (HAN et al., 2018)	43.73 ±0.16	34.96 ±0.50	15.15 ±0.46	28.35 ±0.25
Co-Teaching+ (YU et al., 2019)	49.27 ±0.03	40.04 ±0.70	13.44 ±0.37	33.62 ±0.39
JoCoR (WEI et al., 2020)	53.01 ±0.04	43.49 ±0.46	15.49 ±0.98	32.70 ±0.35
CJC-Net (ZHANG et al., 2021)	60.04 ±0.15	52.57 ±0.17	10.41 ±0.09	47.77 ±0.14
Optimized Vanilla	59.16 ±0.05	49.07 ±0.42	27.04 ±0.52	41.53 ±0.03
Enhanced Co-Teaching	62.35 ±0.05	56.92 ±1.01	14.90 ±1.40	45.98 ±0.67
CCT	**64.70** ±0.36	**59.40** ±0.01	**42.18** ±0.30	**51.86** ±0.27

Table 10 – Test accuracies (%) on CIFAR-100 dataset under various noise conditions using a 7-layer CNN architecture. Baseline results are taken directly from Zhang et al. (2021). Reported values for our methods correspond to the mean ± standard deviation over 5 runs using a 10-epoch simple moving average. Best results are shown in bold.

Dataset	CIFAR-100							
Noise type	**Symmetric**				**Asymmetric**			
Method/Noise ratio	20%	40%	50%	80%	10%	20%	30%	40%
Vanilla (ZHANG et al., 2021)	49.32	34.74	–	7.25	59.75	–	–	–
Co-Teaching (HAN et al., 2018)	53.79	46.47	–	12.23	57.53	–	–	–
O2U-Net (HUANG et al., 2019)	60.53	50.30	–	20.44	64.50	–	–	–
CJC-Net (ZHANG et al., 2021)	70.13	66.26	–	12.78	71.67	–	–	–
TCC-Net (XIA; LEE; CHEN, 2023)	68.93	–	62.44	18.82	–	66.99	62.92	54.63
Optimized Vanilla	65.67	60.04	56.58	31.44	69.89	66.87	61.47	51.64
Enhanced Co-Teaching	53.31	52.88	48.59	12.06	53.76	53.62	51.56	41.91
CCT	**70.79**	**66.31**	**63.50**	**42.35**	**72.47**	**70.41**	**67.79**	**61.05**

Table 11 – Test accuracies (%) on CIFAR-100 dataset under various noise conditions using a 9-layer CNN architecture. Baseline results are taken directly from Zhang et al. (2021) and Xia, Lee & Chen (2023). Reported values for our methods correspond to the mean over 5 runs using a 10-epoch simple moving average. Best results are shown in bold.

7.1.4 Real-world datasets

The results presented in Table 14 demonstrate the performance of the vanilla model, Enhanced Co-Teaching, and CCT method on several real-world noisy datasets: Animal-10N, Food-101N, and Clothing1M. These datasets are known for containing label noise, providing a challenging scenario to evaluate the effectiveness of the models.

For the Animal-10N dataset, the best-performing vanilla model shows good results, surpassing other methods like Co-learning. The CCT method achieves an improvement of approximately $1.4pp$ over TCC-Net.

For the Food-101N dataset, the optimized vanilla model stands out by outperforming most approaches, only trailing behind Co-learning. On the other hand, our CCT method not only surpasses the optimized vanilla model but also achieves performance notably close to

Dataset	CIFAR-100				
Noise type	**Symmetric**			**Asymmetric**	
Method/Noise ratio	20%	50%	80%	30%	40%
Vanilla (TAN et al., 2021)	57.79 ±0.44	33.75 ±0.46	8.64 ±0.22	51.06 ±0.44	42.49 ±0.23
Decoupling (MALACH; SHALEV-SHWARTZ, 2017)	56.18 ±0.32	31.58 ±0.54	7.71 ±0.23	49.86 ±0.54	41.51 ±0.67
Co-Teaching (HAN et al., 2018)	64.28 ±0.32	32.62 ±0.51	6.65 ±0.71	49.53 ±0.79	40.62 ±0.79
Co-Teaching+ (YU et al., 2019)	55.40 ±0.71	26.49 ±0.45	8.57 ±1.55	47.12 ±0.73	38.98 ±0.54
JoCoR (WEI et al., 2020)	62.29 ±0.71	30.19 ±0.60	6.84 ±0.92	49.04 ±0.91	39.72 ±0.76
Co-learning (TAN et al., 2021)	66.58 ±0.15	54.54 ±0.43	35.45 ±0.79	56.97 ±1.22	47.62 ±0.79
Optimized Vanilla	69.36 ±0.06	60.80 ±0.44	35.97 ±1.37	66.61 ±0.59	52.02 ±0.16
Enhanced Co-Teaching	70.77 ±0.57	65.44 ±0.05	14.96 ±0.06	64.70 ±0.02	56.50 ±0.22
CCT	**74.36** ±0.15	**67.36** ±0.02	**47.53** ±0.07	**73.85** ±0.40	**67.86** ±0.10

Table 12 – Test accuracies (%) on CIFAR-100 dataset with different types and levels of label noise obtained using ResNet-18. The results of the other methods are taken directly from Tan et al. (2021). Reported values for our methods correspond to the mean and standard deviation over 5 runs using a 10-epoch simple moving average. The best result in each case is highlighted in bold.

Noise type	**Symmetric**		**Asymmetric**
Method/Noise ratio	20%	50%	45%
Vanilla (YU et al., 2019)	35.56	19.58	26.14
Decoupling (MALACH; SHALEV-SHWARTZ, 2017)	36.28	22.61	26.10
Co-Teaching (HAN et al., 2018)	45.60	37.09	27.41
Co-Teaching+ (YU et al., 2019)	47.73	41.19	26.54
Optimized Vanilla	52.82	45.71	35.50
Enhanced Co-Teaching	37.06	29.97	27.99
CCT	**54.28**	**47.37**	**36.09**

Table 13 – Test accuracies (%) on the Tiny-ImageNet dataset for various label types and noise levels using PreAct ResNet-18. The results of the other methods are taken directly from Yu et al. (2019). Reported values for our methods correspond to the mean over 5 runs using a 10-epoch simple moving average. The best result in each case is highlighted in bold.

the state-of-the-art represented by Co-learning. This result is even more significant when considering that CCT adopts a much simpler strategy compared to the advanced techniques used by Co-learning, which include complex DA (such as MixUp), a customized loss function with terms that incorporate intrinsic and structural similarity information, and a hybrid framework that combines supervised and self-supervised learning.

Lastly, for the Clothing1M dataset, consistent with the results on Animal-10N and Food-101N, the optimized vanilla model demonstrates strong performance, outperforming methods such as JoCoR and TCC-Net. When compared to CJC-Net, the CCT method achieves an improvement of $0.08pp$.

The results highlight the versatility of the CCT method in various noisy label scenarios, demonstrating its ability to maintain consistent and reliable performance. By achieving competitive results across multiple datasets, the method proves its robustness and potential as an effective strategy for learning in noisy label environments. These results also indicate that, when properly optimized, the vanilla model can achieve competitive performance on

Datasets	**Animal-10N**	**Food-101N**	**Clothing1M**
Methods/DNN	**9-layer CNN**	**ResNet-34**	**ResNet-18**
Vanilla[†]	82.68	84.50	67.22
Decoupling (MALACH; SHALEV-SHWARTZ, 2017)	79.22	85.53	68.48
Co-Teaching (HAN et al., 2018)	82.43	61.91	69.21
Co-Teaching+ (YU et al., 2019)	50.66	81.61	59.32
JoCoR (WEI et al., 2020)	82.82	77.94	70.30
Co-learning (TAN et al., 2021)	82.95	**87.57**	–
O2U-Net (HUANG et al., 2019)	–	–	71.95
CJC-Net (ZHANG et al., 2021)	–	–	72.55
TCC-Net (XIA; LEE; CHEN, 2023)	83.22	–	70.46
Optimized vanilla	82.96	86.19	71.79
Enhanced Co-Teaching	83.48	84.33	71.11
CCT	**84.71**	87.42	**72.63**

[†]Vanilla results were obtained from: Xia, Lee & Chen (2023) for Animal-10N and Clothing1M, and Tan et al. (2021) for Food-101N

Table 14 – Test accuracies (%) on different real-world datasets with their respective DNNs. For Animal-10N, Food-101N, and Clothing1M, the baseline results are taken directly from Xia, Lee & Chen (2023), Tan et al. (2021), and both Zhang et al. (2021) and Xia, Lee & Chen (2023), respectively. Reported values for our methods correspond to the mean of the best test accuracy over 5 runs. The best result for each dataset is highlighted in bold.

real-world noisy datasets.

7.2 ABLATION STUDY

In this section, we investigate the effects of removing different proposed strategies to assess their impact on the performance of the proposed method. The results of this analysis are presented in Table 15. Specifically:

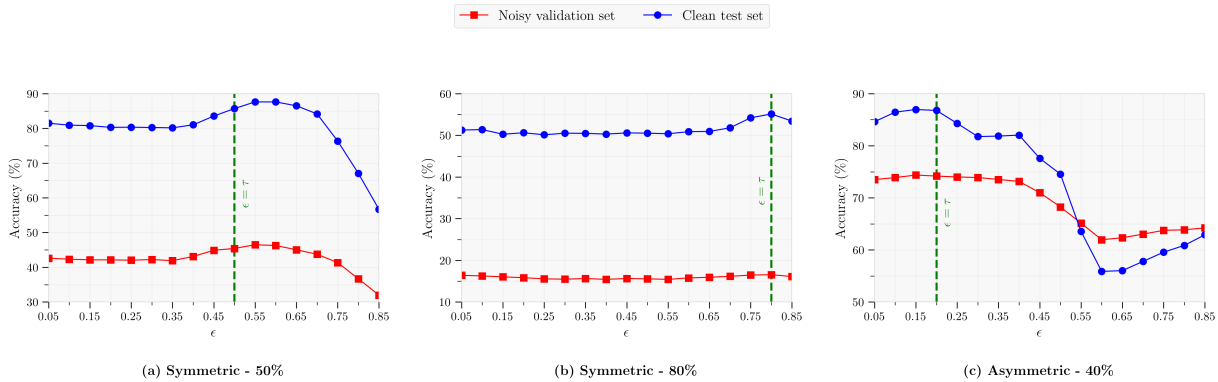
- To evaluate the impact of checkpoint usage in CCT, we trained the models in such a way that, in each subsequent cycle, training continued from the state of the models at the last epoch of the previous cycle. Decreases in performance when this strategy is absent highlight its importance for CCT's cyclic training in achieving better performance;
- In the absence of the hyperparameter optimization process, we evaluated the performance obtained with the standard Co-Teaching settings ($t_k = 10$ and $\delta = 0$). The performance reductions compared to CCT highlight the significance of optimizing the hyperparameters, particularly the sample retention rate, for improved performance;
- To study the effect of omitting vanilla pre-training, we conducted the cyclic training phase using $\eta_{\max} = 0.02$, $T = 100$, and $\eta_{\min} = \eta_{\max}/T$ with models starting from scratch. Notably, vanilla pre-training is a crucial component for the high performance of CCT, especially in scenarios with high noise rates.

Noise type	**Symmetric**		**Asymmetric**
Method/Noise ratio	50%	80%	40%
CCT	**88.29**	**60.20**	**87.39**
CCT w/o checkpoint	87.85	57.09	86.69
CCT w/o optimization of t_k and δ	86.44	56.76	87.07
CCT w/o vanilla pre-training	88.05	24.54	85.52

Table 15 – Results of the ablation study in terms of test accuracy (%) on CIFAR-10 using the 7-layer CNN. We report mean over 5 runs using a 10-epoch simple moving average. The best result is highlighted in bold.

7.3 IMPACT OF CYCLIC RETENTION RATE TUNING ON CCT

In this section, we discuss the impact of the $R_c(t)$ hyperparameters on the performance of the CCT method. Following the univariate optimization procedure proposed in subsection 5.2, in the absence of prior knowledge of the label noise rate (τ), optimizing ϵ is the first step to take. In Fig. 12, we demonstrate how the average performance of CCT varies with respect to ϵ for different label noise scenarios.



(a) Symmetric - 50% (b) Symmetric - 80% (c) Asymmetric - 40%

Figure 12 – Impact of hyperparameter ϵ optimization in CCT (with $t_k = 0$ and $\delta = 0$) on the performance of the 7-layer CNN on CIFAR-10 with noisy labels. Results are averaged over 5 runs using a 10-epoch simple moving average.

Noise Type	**Hyperparameter**				**Accuracy (%)**	
	τ	ϵ	t_k	δ	Validation	Test
Sym. 50%	Known	0.50	0	0.075	46.38	88.29
	Optimized	0.55	5	0.050	46.41	88.37
Sym. 80%	Known	0.80	5	0.100	17.69	60.20
	Optimized	0.80	5	0.100	17.69	60.20
Asym. 40%	Known	0.20	10	0.025	73.98	87.33
	Optimized	0.15	5	0.050	74.05	87.46

Table 16 – Impact of hyperparameters t_k and δ on CCT performance for CIFAR-10 with noisy labels. We report mean over 5 runs using a 10-epoch simple moving average.

It is observed that the best value of ϵ (which maximizes the average performance on the test set) is close to or coincides with $\epsilon = \tau$. On the other hand, when ϵ is optimized on the noisy validation set, the resulting performance is equal to or better than that obtained by setting $\epsilon = \tau$. This can be seen in Fig. 12 for $t_k = 0$ and $\delta = 0$, as well as in Table 16 after optimizing t_k and δ. Note that varying t_k and δ is beneficial even when the optimized value of ϵ is used.

8 CONCLUSION

In this thesis, we present a comprehensive investigation into LNL through the lens of training dynamics, which culminated in the development of the CCT method. Initially, we highlight aspects of a realistic training methodology in LNL, with emphasis on hyperparameter optimization and early stopping criteria based on noisy validation sets. Through extensive empirical analysis, we expanded the understanding of how higher learning rates act as an implicit regularization mechanism against noisy labels, providing systematic evidence across diverse scenarios and extending initial observations from the literature. This investigation, although independent, reinforced the importance of careful hyperparameter optimization even for vanilla models, which, when properly optimized, were capable of rivaling or even surpassing sophisticated LNL methods.

Centrally, we proposed the CCT method, which integrated into the Co-Teaching framework strategies such as pre-training, coordinated cyclic variations of learning rate and sample retention rate, along with a holistic and straightforward approach to hyperparameter optimization, making it easily reproducible. CCT sought to overcome the premature convergence limitation present in traditional Co-Teaching frameworks. By introducing cyclic variations, the method maintained diversity between models throughout training and progressively modulated the filtering of potentially noisy samples through predefined cycles. This design exploited the fact that different phases of cyclic training exhibited distinct levels of robustness to label noise, enabling a more dynamic and effective approach to collaborative learning.

The experimental results obtained across multiple architectures, noise types and intensities, as well as various datasets, demonstrated that CCT consistently outperformed state-of-the-art methods. The gains were particularly significant under severe noise conditions, traditionally challenging for existing approaches, representing substantial advances in the ability to handle extreme label corruption scenarios.

Beyond the algorithmic contributions, this thesis evidenced critical methodological issues in LNL research practices. We identified and discussed recurring evaluation problems, including inappropriate use of test sets for validation and systematic underestimation of baseline models. Our analyses showed that properly optimized vanilla models were capable of matching, and sometimes surpassing, the performance of sophisticated LNL methods, suggesting that a substantial portion of the progress reported in the field could reflect evaluation artifacts rather than genuine algorithmic advances. This finding reinforced the need to establish fair baselines and maintain rigorous experimental protocols.

From a practical standpoint, the results also indicated that the CCT method possessed potential for application in real-world scenarios where label quality could not be guaranteed. Unlike approaches requiring clean validation sets or prior knowledge about noise characteristics, CCT proved promising even with potentially noisy validation data, benefiting from its univariate optimization framework. This flexibility, combined with its relative computational efficiency compared to more complex approaches, suggested that CCT could constitute a

practical alternative in contexts with imperfect annotations.

We acknowledge, however, some limitations of the proposed approach. First, although cyclic training presented clear benefits, the optimal length and amplitude of cycles remained dataset-dependent, requiring empirical adjustments. Second, the computational overhead associated with the simultaneous training of two networks, although common in multi-model approaches, may have limited its applicability in resource-constrained environments. Finally, our analysis focused on classification tasks with closed-set noise, leaving open the investigation of CCT's effectiveness in regression problems or open-set noise scenarios, where samples from unseen classes were introduced during training.

8.1 FUTURE WORKS

The findings and limitations of this research suggest several promising directions for future investigation:

- Adaptive cyclic scheduling: The development of adaptive mechanisms that automatically adjust cycle parameters based on training dynamics would eliminate the need for manual tuning. This could involve meta-learning approaches or reinforcement learning frameworks that optimize the cyclic schedule in response to observed learning curves and validation performance, potentially leading to more robust and generalizable solutions;
- Integration with SSL: Recent advances in self-supervised pretraining have shown promise for LNL. Combining CCT's cyclic dynamics with self-supervised objectives could yield synergistic benefits, where unsupervised representations guide sample selection while cyclic training prevents premature convergence to noisy patterns. This integration could be particularly valuable in domains with limited labeled data;
- Theoretical analysis of cyclic dynamics: While our empirical results demonstrate CCT's effectiveness, a rigorous theoretical analysis of why cyclic variations enhance robustness remains an open problem. Future work should investigate the optimization landscape under cyclic scheduling, potentially drawing connections to recent advances in understanding neural network training dynamics and implicit regularization;
- Application to other learning paradigms: Extending CCT principles to SSL, few-shot learning, and continual learning scenarios could broaden their impact. The cyclic modulation of learning dynamics might help address challenges like catastrophic forgetting in continual learning or improve pseudo-label quality in semi-supervised settings.

The research presented in this thesis establishes cyclic training dynamics as a powerful principle for LNL, opening new avenues for developing robust DL systems that can effectively leverage imperfect supervision in real-world applications.

8.2 PUBLISHED PAPERS

The main contributions of this thesis were consolidated in the following publication:

- J. K. S. Kamassury, H. Pickler, F. R. Cordeiro, and D. Silva. **CCT: A Cyclic Co-Teaching Approach to Train Deep Neural Networks With Noisy Labels**. In: *IEEE Access* (2025). doi: https://doi.org/10.1109/ACCESS.2025.3548510.

Additional publications produced during the Ph.D., either related to this thesis or co-authored with other researchers, include:

- H. Pickler, J. K. S. Kamassury, and D. Silva. **Benchmarking Noisy Label Detection Methods**. Submitted to *Information Sciences* (Elsevier), currently under review (2025). Also available as a preprint on arXiv: https://doi.org/10.48550/arXiv.2510.16211.

- E. Bitencourt, J. Kamassury, and D. Silva. **Análise de funções de perda para segmentação de lesões cutâneas usando redes neurais convolucionais e transformações polares**. In: *Anais Complementares do IX Congresso Latino-Americano de Engenharia Biomédica (CLAIB 2022)* and the *XXVIII Congresso Brasileiro de Engenharia Biomédica (CBEB 2022)*, Online. Galoá, 2024.

BIBLIOGRAPHY

ALBERT, P. et al. Addressing out-of-distribution label noise in webly-labelled data. In: **2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)**. [s.n.], 2022. p. 2393–2402. Available at: https://doi.org/10.1109/WACV51458.2022.00245.

ARAZO, E. et al. Unsupervised label noise modeling and loss correction. In: CHAUDHURI, K.; SALAKHUTDINOV, R. (Ed.). **Proceedings of the 36th International Conference on Machine Learning**. PMLR, 2019. (Proceedings of Machine Learning Research, v. 97), p. 312–321. Available at: https://proceedings.mlr.press/v97/arazo19a.html.

ARPIT, D. et al. A closer look at memorization in deep networks. In: PRECUP, D.; TEH, Y. W. (Ed.). **Proceedings of the 34th International Conference on Machine Learning**. PMLR, 2017. (Proceedings of Machine Learning Research, v. 70), p. 233–242. Available at: https://proceedings.mlr.press/v70/arpit17a.html.

BAI, Y. et al. Understanding and improving early stopping for learning with noisy labels. In: RANZATO, M. et al. (Ed.). **Advances in Neural Information Processing Systems**. [S.l.]: Curran Associates, Inc., 2021. v. 34, p. 24392–24403.

BENGIO, Y. Practical recommendations for gradient-based training of deep architectures. In: MONTAVON, G.; ORR, G. B.; MüLLER, K.-R. (Ed.). **Neural Networks: Tricks of the Trade: Second Edition**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. p. 437–478. ISBN 978-3-642-35289-8. DOI: https://doi.org/10.1007/978-3-642-35289-8_26.

BERNHARDT, M. et al. Active label cleaning for improved dataset quality under resource constraints. **Nature Communications**, v. 13, n. 1, p. 1161, 2022. Available at: https://doi.org/10.1038/s41467-022-28818-3.

BOSSARD, L.; GUILLAUMIN, M.; GOOL, L. V. Food-101 – mining discriminative components with random forests. In: FLEET, D. et al. (Ed.). **Computer Vision – ECCV 2014**. Cham: Springer International Publishing, 2014. p. 446–461. ISBN 978-3-319-10599-4. Available at: https://doi.org/10.1007/978-3-319-10599-4_29.

CARNEIRO, G. **Machine Learning with Noisy Labels**. 1. ed. [S.l.]: Academic Press, 2024.

CHANG, H.-S.; LEARNED-MILLER, E.; MCCALLUM, A. Active bias: Training more accurate neural networks by emphasizing high variance samples. In: **Advances in Neural Information Processing Systems**. Curran Associates, Inc., 2017. v. 30, p. 7002–7012. Available at: https://papers.nips.cc/paper_files/paper/2017/hash/2f37d10131f2a483a8dd005b3d14b0d9-Abstract.html.

CHAROENPHAKDEE, N.; LEE, J.; SUGIYAMA, M. On symmetric losses for learning from corrupted labels. In: CHAUDHURI, K.; SALAKHUTDINOV, R. (Ed.). **Proceedings of the 36th International Conference on Machine Learning**. PMLR, 2019. (Proceedings of Machine Learning Research, v. 97), p. 961–970. Available at: https://proceedings.mlr.press/v97/charoenphakdee19a.html.

CHEN, P. et al. Understanding and utilizing deep neural networks trained with noisy labels. In: CHAUDHURI, K.; SALAKHUTDINOV, R. (Ed.). **Proceedings of the 36th International Conference on Machine Learning**. [S.l.]: PMLR, 2019. (Proceedings of Machine Learning Research, v. 97), p. 1062–1070. Available at: https://proceedings.mlr.press/v97/chen19g.html.

CHEN, P. et al. Beyond class-conditional assumption: A primary attempt to combat instance-dependent label noise. **Proceedings of the AAAI Conference on Artificial Intelligence**, v. 35, n. 13, p. 11442–11450, May 2021. Available at: https://ojs.aaai.org/index.php/AAAI/article/view/17363.

CHEN, Y. et al. Compressing features for learning with noisy labels. **IEEE Transactions on Neural Networks and Learning Systems**, v. 35, n. 2, p. 2124–2138, 2024. Available at: https://doi.org/10.1109/TNNLS.2022.3186930.

CHENG, J. et al. Learning with bounded instance and label-dependent label noise. In: III, H. D.; SINGH, A. (Ed.). **Proceedings of the 37th International Conference on Machine Learning**. PMLR, 2020. (Proceedings of Machine Learning Research, v. 119), p. 1789–1799. Available at: https://proceedings.mlr.press/v119/cheng20c.html.

DEHGHANI, M. et al. Learning to learn from weak supervision by full supervision. In: **NeurIPS Workshop on Meta-Learning**. [s.n.], 2017. Available at: https://meta-learn.github.io/2017/papers/metalearn17_dehghani.pdf.

DENG, J. et al. Imagenet: A large-scale hierarchical image database. In: **2009 IEEE Conference on Computer Vision and Pattern Recognition**. [s.n.], 2009. p. 248–255. Available at: https://doi.org/10.1109/CVPR.2009.5206848.

DING, Y. et al. A semi-supervised two-stage approach to learning from noisy labels. In: **2018 IEEE Winter Conference on Applications of Computer Vision (WACV)**. [s.n.], 2018. p. 1215–1224. Available at: https://doi.org/10.1109/WACV.2018.00138.

FENG, C.; TZIMIROPOULOS, G.; PATRAS, I. Ssr: An efficient and robust framework for learning with unknown label noise. In: **33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022**. BMVA Press, 2022. Available at: https://bmvc2022.mpi-inf.mpg.de/0372.pdf.

FORET, P. et al. Sharpness-aware minimization for efficiently improving generalization. In: **Proceedings of the 9th International Conference on Learning Representations (ICLR)**. [s.n.], 2021. Available at: https://openreview.net/forum?id=6Tm1mposlrM.

FRENAY, B.; VERLEYSEN, M. Classification in the Presence of Label Noise: A Survey. **IEEE Transactions on Neural Networks and Learning Systems**, v. 25, n. 5, p. 845–869, 2014. Available at: https://doi.org/10.1109/TNNLS.2013.2292894.

GHIASI, M. A.; SHAFAHI, A.; ARDEKANI, R. **Improving Robustness with Adaptive Weight Decay**. 2023. ArXiv. DOI: https://doi.org/10.48550/arXiv.2210.00094.

GHOSH, A.; KUMAR, H.; SASTRY, P. S. Robust loss functions under label noise for deep neural networks. **Proceedings of the AAAI Conference on Artificial Intelligence**, v. 31, n. 1, Feb. 2017. DOI: https://doi.org/10.1609/aaai.v31i1.10894.

GOLDBERGER, J.; BEN-REUVEN, E. Training deep neural-networks using a noise adaptation layer. In: **Proceedings of the 5th International Conference on Learning Representations (ICLR)**. [s.n.], 2017. Available at: https://openreview.net/forum?id=H12GRgcxg.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning**. Cambridge, MA: MIT Press, 2016.

GULRAJANI, I.; LOPEZ-PAZ, D. In search of lost domain generalization. In: **Proceedings of the 9th International Conference on Learning Representations (ICLR)**. [S.l.: s.n.], 2021. Available at: https://openreview.net/forum?id=lQdXeXDoWtI.

HAN, B. et al. SIGUA: Forgetting may make learning with noisy labels more robust. In: III, H. D.; SINGH, A. (Ed.). **Proceedings of the 37th International Conference on Machine Learning**. [S.l.]: PMLR, 2020. (Proceedings of Machine Learning Research, v. 119), p. 4006–4016. Available at: https://proceedings.mlr.press/v119/han20c.html.

HAN, B. et al. **Masking: A New Perspective of Noisy Supervision**. 2018. ArXiv. Available at: https://doi.org/10.48550/arXiv.1805.08193.

HAN, B. et al. **A Survey of Label-noise Representation Learning: Past, Present and Future**. 2021. ArXiv. Available at: https://doi.org/10.48550/arXiv.2011.04406.

HAN, B. et al. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In: **Advances in Neural Information Processing Systems**. Curran Associates, Inc., 2018. v. 31, p. 8527–8537. Available at: https://papers.nips.cc/paper_files/paper/2018/hash/a19744e268754fb0148b017647355b7b-Abstract.html.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. **The Elements of Statistical Learning: Data Mining, Inference, and Prediction**. 2. ed. New York: Springer, 2009.

HENDRYCKS, D.; LEE, K.; MAZEIKA, M. Using pre-training can improve model robustness and uncertainty. In: **Proceedings of the 36th International Conference on Machine Learning**. PMLR, 2019. (Proceedings of Machine Learning Research, v. 97), p. 2712–2721. Available at: https://proceedings.mlr.press/v97/hendrycks19a.html.

HENDRYCKS, D. et al. Using trusted data to train deep networks on labels corrupted by severe noise. In: **Advances in Neural Information Processing Systems**. Curran Associates, Inc., 2018. v. 31, p. 10477–10488. Available at: https://papers.nips.cc/paper_files/paper/2018/hash/ad554d8c3b06d6b97ee76a2448bd7913-Abstract.html.

HU, W.; LI, Z.; YU, D. Simple and effective regularization methods for training on noisily labeled data with generalization guarantee. In: **Proceedings of the 8th International Conference on Learning Representations (ICLR)**. [s.n.], 2020. Available at: https://iclr.cc/virtual_2020/poster_Hke3gyHYwH.html.

HUANG, J. et al. O2u-net: A simple noisy label detection approach for deep neural networks. In: **2019 IEEE/CVF International Conference on Computer Vision (ICCV)**. [s.n.], 2019. p. 3325–3333. Available at: https://doi.org/10.1109/ICCV.2019.00342.

HUANG, L.; ZHANG, C.; ZHANG, H. Self-adaptive training: beyond empirical risk minimization. In: LAROCHELLE, H. et al. (Ed.). **Advances in Neural Information Processing Systems**. Curran Associates, Inc., 2020. v. 33, p. 19365–19376. Available at: https://proceedings.neurips.cc/paper/2020/hash/e0ab531ec312161511493b002f9be2ee-Abstract.html.

HUANG, Y. et al. Uncertainty-aware learning against label noise on imbalanced datasets. **Proceedings of the AAAI Conference on Artificial Intelligence**, v. 36, n. 6, p. 6960–6969, Jun. 2022. Available at: https://ojs.aaai.org/index.php/AAAI/article/view/20654.

IBRAHIM, S. et al. **Learning From Crowdsourced Noisy Labels: A Signal Processing Perspective**. 2024. ArXiv. Available at: https://doi.org/10.48550/arXiv.2407.06902.

IRVIN, J. et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. **Proceedings of the AAAI Conference on Artificial Intelligence**, v. 33, n. 01, p. 590–597, Jul. 2019. Available at: https://doi.org/10.1609/aaai.v33i01.3301590.

JIANG, L. et al. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In: **Proceedings of the 35th International Conference on Machine Learning**. PMLR, 2018. (Proceedings of Machine Learning Research, v. 80), p. 2304–2313. Available at: https://proceedings.mlr.press/v80/jiang18c.html.

JINDAL, I.; NOKLEBY, M.; CHEN, X. Learning deep networks from noisy labels with dropout regularization. In: **2016 IEEE 16th International Conference on Data Mining (ICDM)**. [s.n.], 2016. p. 967–972. Available at: https://doi.org/10.1109/ICDM.2016.0121.

KONG, K. et al. Recycling: Semi-supervised learning with noisy labels in deep neural networks. **IEEE Access**, v. 7, p. 66998–67005, 2019. Available at: https://doi.org/10.1109/ACCESS.2019.2918794.

KRAUSE, J. et al. Grader variability and the importance of reference standards for evaluating machine learning models for diabetic retinopathy. **Ophthalmology**, v. 125, n. 8, p. 1264–1272, 2018. ISSN 0161-6420. Available at: https://doi.org/10.1016/j.ophtha.2018.01.034.

KöHLER, J. M.; AUTENRIETH, M.; BELUCH, W. H. **Uncertainty Based Detection and Relabeling of Noisy Image Labels**. 2019. ArXiv. Available at: https://doi.org/10.48550/arXiv.1906.11876.

LEE, K.-H. et al. Cleannet: Transfer learning for scalable image classifier training with label noise. In: **Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)**. [s.n.], 2018. p. 5447–5456. Available at: https://doi.org/10.1109/CVPR.2018.00571.

LI, J. et al. Cyclic annealing training convolutional neural networks for image classification with noisy labels. In: **2018 25th IEEE International Conference on Image Processing (ICIP)**. [s.n.], 2018. p. 21–25. Available at: https://doi.org/10.1109/ICIP.2018.8451331.

LI, J.; SOCHER, R.; HOI, S. C. H. Dividemix: Learning with noisy labels as semi-supervised learning. In: **Proceedings of the 8th International Conference on Learning Representations (ICLR)**. [s.n.], 2020. Available at: https://iclr.cc/virtual_2020/poster_HJgExaVtwr.html.

LI, X. et al. Provably end-to-end label-noise learning without anchor points. In: MEILA, M.; ZHANG, T. (Ed.). **Proceedings of the 38th International Conference on Machine Learning**. PMLR, 2021. (Proceedings of Machine Learning Research, v. 139), p. 6403–6413. Available at: https://proceedings.mlr.press/v139/li21l.html.

LI, Y. et al. Learning from noisy labels with distillation. In: **2017 IEEE International Conference on Computer Vision (ICCV)**. [s.n.], 2017. p. 1928–1936. Available at: https://doi.org/10.1109/ICCV.2017.211.

LIANG, X.; LIU, X.; YAO, L. Review–A Survey of Learning from Noisy Labels. **ECS Sens. Plus**, v. 1, n. 2, p. 021401, jun. 2022. Available at: https://doi.org/10.1149/2754-2726/ac75f5.

LIU, S. et al. Early-learning regularization prevents memorization of noisy labels. In: **Advances in Neural Information Processing Systems**. Curran Associates, Inc., 2020. v. 33, p. 20331–20342. Available at: https://proceedings.neurips.cc//paper_files/paper/2020/hash/ea89621bee7c88b2c5be6681c8ef4906-Abstract.html.

LIU, T.; TAO, D. Classification with noisy labels by importance reweighting. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 38, n. 3, p. 447–461, 2016. Available at: https://doi.org/10.1109/TPAMI.2015.2456899.

LIU, X. et al. Towards robust adaptive object detection under noisy annotations. In: **2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**. [s.n.], 2022. p. 14187–14196. Available at: https://doi.org/10.1109/CVPR52688.2022.01381.

LONG, P. M.; SERVEDIO, R. A. Random classification noise defeats all convex potential boosters. In: **Proceedings of the 25th International Conference on Machine Learning**. ACM, 2008. p. 608–615. Available at: https://doi.org/10.1145/1390156.1390233.

LOSHCHILOV, I.; HUTTER, F. SGDR: Stochastic gradient descent with warm restarts. In: **Proceedings of the 5th International Conference on Learning Representations (ICLR)**. [s.n.], 2017. Available at: https://openreview.net/forum?id=Skq89Scxx.

LUKASIK, M. et al. Does label smoothing mitigate label noise? In: III, H. D.; SINGH, A. (Ed.). **Proceedings of the 37th International Conference on Machine Learning**. PMLR, 2020. (Proceedings of Machine Learning Research, v. 119), p. 6448–6458. Available at: https://proceedings.mlr.press/v119/lukasik20a.html.

MA, X. et al. Normalized loss functions for deep learning with noisy labels. In: III, H. D.; SINGH, A. (Ed.). **Proceedings of the 37th International Conference on Machine Learning**. PMLR, 2020. (Proceedings of Machine Learning Research, v. 119), p. 6543–6553. Available at: https://proceedings.mlr.press/v119/ma20c.html.

MALACH, E.; SHALEV-SHWARTZ, S. Decoupling "when to update" from "how to update". In: **Advances in Neural Information Processing Systems**. Curran Associates, Inc., 2017. v. 30, p. 960–970. Available at: https://papers.nips.cc/paper_files/paper/2017/hash/58d4d1e7b1e97b258c9ed0b37e02d087-Abstract.html.

MANWANI, N.; SASTRY, P. S. Noise tolerance under risk minimization. **IEEE Transactions on Cybernetics**, v. 43, n. 3, p. 1146–1151, 2013. Available at: https://doi.org/10.1109/TSMCB.2012.2223460.

NADO, Z. et al. **Uncertainty Baselines: Benchmarks for Uncertainty & Robustness in Deep Learning**. 2022. ArXiv. DOI: https://doi.org/10.48550/arXiv.2106.04015.

NATARAJAN, N. et al. Learning with noisy labels. In: BURGES, C. et al. (Ed.). **Advances in Neural Information Processing Systems**. Curran Associates, Inc., 2013. v. 26. Available at: https://papers.nips.cc/paper_files/paper/2013/hash/3871bd64012152bfb53fdf04b401193f-Abstract.html.

NGUYEN, D. T. et al. Self: Learning to filter noisy labels with self-ensembling. In: **Proceedings of the 8th International Conference on Learning Representations (ICLR)**. [s.n.], 2020. Available at: https://iclr.cc/virtual_2020/poster_HkgsPhNYPS.html.

NISHI, K. et al. Augmentation strategies for learning with noisy labels. In: **2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**. [s.n.], 2021. p. 8018–8027. Available at: https://doi.org/10.1109/CVPR46437.2021.00793.

NISHI, K. et al. Augmentation strategies for learning with noisy labels. In: **2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2021. p. 8018–8027. DOI: https://doi.org/10.1109/CVPR46437.2021.00793.

NORTHCUTT, C.; JIANG, L.; CHUANG, I. Confident Learning: Estimating Uncertainty in Dataset Labels. **Journal of Artificial Intelligence Research**, v. 70, p. 1373–1411, abr. 2021. Available at: https://doi.org/10.1613/jair.1.12125.

OAKDEN-RAYNER, L. Exploring large-scale public medical image datasets. **Academic Radiology**, v. 27, n. 1, p. 106–112, 2020. Available at: https://doi.org/10.1016/j.acra.2019.10.006.

OLIVER, A. et al. Realistic Evaluation of Deep Semi-Supervised Learning Algorithms. In: BENGIO, S. et al. (Ed.). **Advances in Neural Information Processing Systems**. Curran Associates, Inc., 2018. v. 31. Available at: https://proceedings.neurips.cc/paper/2018/hash/c1fea270c48e8079d8ddf7d06d26ab52-Abstract.html.

ORTEGO, D. et al. Multi-objective interpolation training for robustness to label noise. In: **2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2021. p. 6602–6611. DOI: https://doi.org/10.1109/CVPR46437.2021.00654.

PANDEY, R. et al. Modeling and mitigating human annotation errors to design efficient stream processing systems with human-in-the-loop machine learning. **International Journal of Human-Computer Studies**, v. 160, p. 102772, 2022. ISSN 1071-5819. Available at: https://doi.org/10.1016/j.ijhcs.2022.102772.

PATRINI, G. et al. Making deep neural networks robust to label noise: A loss correction approach. In: **2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. [s.n.], 2017. p. 2233–2241. Available at: https://doi.org/10.1109/CVPR.2017.240.

PLEISS, G. et al. Identifying mislabeled data using the area under the margin ranking. In: **Advances in Neural Information Processing Systems**. Curran Associates, Inc., 2020. v. 33, p. 17044–17056. Available at: https://papers.nips.cc/paper/2020/hash/c6102b3727b2a7d8b1bb6981147081ef-Abstract.html.

RASCHKA, S. **Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning**. 2018. ArXiv. DOI: https://doi.org/10.48550/arXiv.1811.12808.

REED, S. et al. **Training Deep Neural Networks on Noisy Labels with Bootstrapping**. 2015. ArXiv. Available at: https://doi.org/10.48550/arXiv.1412.6596.

REN, M. et al. Learning to reweight examples for robust deep learning. In: DY, J.; KRAUSE, A. (Ed.). **Proceedings of the 35th International Conference on Machine Learning**. [S.l.]: PMLR, 2018. (Proceedings of Machine Learning Research, v. 80), p. 4334–4343. Available at: https://proceedings.mlr.press/v80/ren18a.html.

ROLNICK, D. et al. **Deep Learning is Robust to Massive Label Noise**. 2018. ArXiv. Available at: https://doi.org/10.48550/arXiv.1705.10694.

ROOYEN, B. van; MENON, A. K.; WILLIAMSON, R. C. **Learning with Symmetric Label Noise: The Importance of Being Unhinged**. 2015. ArXiv. Available at: https://doi.org/10.48550/arXiv.1505.07634.

SACHDEVA, R. et al. Evidentialmix: Learning with combined open-set and closed-set noisy labels. In: **2021 IEEE Winter Conference on Applications of Computer Vision (WACV)**. [s.n.], 2021. p. 3606–3614. Available at: https://doi.org/10.1109/WACV48630.2021.00365.

SCOTT, C. A Rate of Convergence for Mixture Proportion Estimation, with Application to Learning from Noisy Labels. In: LEBANON, G.; VISHWANATHAN, S. V. N. (Ed.). **Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics**. San Diego, California, USA: PMLR, 2015. (Proceedings of Machine Learning Research, v. 38), p. 838–846. Available at: https://proceedings.mlr.press/v38/scott15.html.

SHEN, Y.; SANGHAVI, S. Learning with bad training data via iterative trimmed loss minimization. In: CHAUDHURI, K.; SALAKHUTDINOV, R. (Ed.). **Proceedings of the 36th International Conference on Machine Learning**. PMLR, 2019. (Proceedings of Machine Learning Research, v. 97), p. 5739–5748. Available at: https://proceedings.mlr.press/v97/shen19e.html.

SHU, J. et al. Meta-weight-net: Learning an explicit mapping for sample weighting. In: WALLACH, H. et al. (Ed.). **Advances in Neural Information Processing Systems**. Curran Associates, Inc., 2019. v. 32. Available at: https://proceedings.neurips.cc/paper/2019/hash/e58cc5ca94270acaceed13bc82dfedf7-Abstract.html.

SMART, B.; CARNEIRO, G. Bootstrapping the relationship between images and their clean and noisy labels. In: **2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)**. [s.n.], 2023. p. 5333–5343. Available at: https://doi.org/10.1109/WACV56688.2023.00531.

SMITH, L. N. Cyclical learning rates for training neural networks. In: **2017 IEEE Winter Conference on Applications of Computer Vision (WACV)**. [s.n.], 2017. p. 464–472. Available at: https://doi.org/10.1109/WACV.2017.58.

SONG, H.; KIM, M.; LEE, J.-G. SELFIE: Refurbishing unclean samples for robust deep learning. In: CHAUDHURI, K.; SALAKHUTDINOV, R. (Ed.). **Proceedings of the 36th International Conference on Machine Learning**. PMLR, 2019. (Proceedings of Machine Learning Research, v. 97), p. 5907–5915. Available at: https://proceedings.mlr.press/v97/song19b.html.

SONG, H. et al. Learning from noisy labels with deep neural networks: A survey. **IEEE Transactions on Neural Networks and Learning Systems**, v. 34, n. 11, p. 8135–8153, 2023. Available at: https://doi.org/10.1109/TNNLS.2022.3152527.

SUKHBAATAR, S. et al. **Training Convolutional Networks with Noisy Labels**. 2015. ArXiv. Available at: https://doi.org/10.48550/arXiv.1406.2080.

TAN, C. et al. Co-learning: Learning from noisy labels with self-supervision. In: **Proceedings of the 29th ACM International Conference on Multimedia (MM)**. ACM, 2021. p. 1405–1413. Available at: https://doi.org/10.1145/3474085.3475622.

TANAKA, D. et al. Joint optimization framework for learning with noisy labels. In: **2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition**. [s.n.], 2018. p. 5552–5560. Available at: https://doi.org/10.1109/CVPR.2018.00582.

THULASIDASAN, S. et al. Combating label noise in deep learning using abstention. In: CHAUDHURI, K.; SALAKHUTDINOV, R. (Ed.). **Proceedings of the 36th International Conference on Machine Learning**. PMLR, 2019. (Proceedings of Machine Learning Research, v. 97), p. 6234–6243. Available at: https://proceedings.mlr.press/v97/thulasidasan19a.html.

WANG, X. et al. IMAE for noise-robust learning: Mean absolute error does not treat examples equally and gradient magnitude's variance matters. In: **ICLR 2023 Workshop on Trustworthy and Reliable Large-Scale Machine Learning Models**. [s.n.], 2023. Available at: https://openreview.net/forum?id=oK44liEinV.

WANG, Y. et al. Symmetric cross entropy for robust learning with noisy labels. In: **2019 IEEE/CVF International Conference on Computer Vision (ICCV)**. [s.n.], 2019. p. 322–330. Available at: https://doi.org/10.1109/ICCV.2019.00041.

WEI, H. et al. Combating noisy labels by agreement: A joint training method with co-regularization. In: **2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**. [s.n.], 2020. p. 13723–13732. Available at: https://doi.org/10.1109/CVPR42600.2020.01374.

XIA, Q.; LEE, F.; CHEN, Q. Tcc-net: A two-stage training method with contradictory loss and co-teaching based on meta-learning for learning with noisy labels. **Information Sciences**, v. 639, p. 119008, 2023. Available at: https://doi.org/10.1016/j.ins.2023.119008.

XIA, X. et al. Sample selection with uncertainty of losses for learning with noisy labels. In: **Proceedings of the 10th International Conference on Learning Representations (ICLR)**. [s.n.], 2022. Available at: https://openreview.net/forum?id=xENf4QUL4LW.

XIA, X. et al. **Are Anchor Points Really Indispensable in Label-Noise Learning?** 2019. ArXiv. Available at: https://arxiv.org/abs/1906.00189.

XIAO, R. et al. Promix: Combating label noise via maximizing clean sample utility. In: ELKIND, E. (Ed.). **Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23**. International Joint Conferences on Artificial Intelligence Organization, 2023. p. 4442–4450. Available at: https://doi.org/10.24963/ijcai.2023/494.

XIAO, T. et al. Learning from massive noisy labeled data for image classification. In: **2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. [s.n.], 2015. p. 2691–2699. Available at: https://doi.org/10.1109/CVPR.2015.7298885.

XU, Y. et al. $\mathcal{L}_{\mathrm{DMI}}$: A novel information-theoretic loss function for training deep nets robust to label noise. In: WALLACH, H. et al. (Ed.). **Advances in Neural Information Processing Systems**. Curran Associates, Inc., 2019. v. 32. Available at: https://proceedings.neurips.cc/paper/2019/hash/8a1ee9f2b7abe6e88d1a479ab6a42c5e-Abstract.html.

XU, Y. et al. Faster meta update strategy for noise-robust deep learning. In: **2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**. [s.n.], 2021. p. 144–153. Available at: https://doi.org/10.1109/CVPR46437.2021.00021.

YANG, H. et al. Searching to exploit memorization effect in deep learning with noisy labels. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 46, n. 12, p. 7833–7849, 2024. Available at: https://doi.org/10.1109/TPAMI.2024.3394552.

YAO, J. et al. Deep learning from noisy image labels with quality embedding. **IEEE Transactions on Image Processing**, v. 28, n. 4, p. 1909–1922, 2019. Available at: https://doi.org/10.1109/TIP.2018.2877939.

YAO, Y. et al. Dual t: Reducing estimation error for transition matrix in label-noise learning. In: LAROCHELLE, H. et al. (Ed.). **Advances in Neural Information Processing Systems**. Curran Associates, Inc., 2020. v. 33, p. 7260–7271. Available at: https://papers.nips.cc/paper/2020/hash/512c5cad6c37edb98ae91c8a76c3a291-Abstract.html.

YI, R. et al. Learning from pixel-level label noise: A new perspective for semi-supervised semantic segmentation. **IEEE Transactions on Image Processing**, v. 31, p. 623–635, 2022. Available at: https://doi.org/10.1109/TIP.2021.3134142.

YU, X. et al. How does disagreement help generalization against label corruption? In: CHAUDHURI, K.; SALAKHUTDINOV, R. (Ed.). **Proceedings of the 36th International Conference on Machine Learning**. PMLR, 2019. v. 97, p. 7164–7173. Available at: https://proceedings.mlr.press/v97/yu19b.html.

ZHANG, C. et al. Understanding deep learning requires rethinking generalization. In: **Proceedings of the 5th International Conference on Learning Representations (ICLR)**. [s.n.], 2017. Available at: https://openreview.net/forum?id=Sy8gdB9xx.

ZHANG, H. et al. mixup: Beyond empirical risk minimization. In: **Proceedings of the 6th International Conference on Learning Representations (ICLR)**. [s.n.], 2018. Available at: https://openreview.net/forum?id=r1Ddp1-Rb.

ZHANG, Q. et al. Cjc-net: A cyclical training method with joint loss and co-teaching strategy net for deep learning under noisy labels. **Information Sciences**, v. 579, p. 186–198, 2021. Available at: https://doi.org/10.1016/j.ins.2021.08.008.

ZHANG, Y.; NIU, G.; SUGIYAMA, M. Learning noise transition matrix from only noisy labels via total variation regularization. In: MEILA, M.; ZHANG, T. (Ed.). **Proceedings of the 38th International Conference on Machine Learning**. PMLR, 2021. (Proceedings of Machine Learning Research, v. 139), p. 12501–12512. Available at: https://proceedings.mlr.press/v139/zhang21n.html.

ZHANG, Z.; PFISTER, T. Learning fast sample re-weighting without reward data. In: **2021 IEEE/CVF International Conference on Computer Vision (ICCV)**. [s.n.], 2021. p. 705–714. Available at: https://doi.org/10.1109/ICCV48922.2021.00076.

ZHANG, Z.; SABUNCU, M. Generalized cross entropy loss for training deep neural networks with noisy labels. In: BENGIO, S. et al. (Ed.). **Advances in Neural Information Processing Systems**. Curran Associates, Inc., 2018. v. 31. Available at: https://papers.nips.cc/paper_files/paper/2018/hash/f2925f97bc13ad2852a7a551802feea0-Abstract.html.

ZHANG, Z. et al. Distilling effective supervision from severe label noise. In: **2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**. [s.n.], 2020. p. 9291–9300. Available at: https://doi.org/10.1109/CVPR42600.2020.00931.

ZHELTONOZHSKII, E. et al. Contrast to divide: Self-supervised pre-training for learning with noisy labels. In: **2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)**. [s.n.], 2022. p. 387–397. Available at: https://doi.org/10.1109/WACV51458.2022.00046.

ZHENG, G.; AWADALLAH, A. H.; DUMAIS, S. Meta label correction for noisy label learning. **Proceedings of the AAAI Conference on Artificial Intelligence**, v. 35, n. 12, p. 11053–11061, May 2021. Available at: https://ojs.aaai.org/index.php/AAAI/article/view/17319.

ZHU, Z.; SONG, Y.; LIU, Y. Clusterability as an alternative to anchor points when learning with noisy labels. In: MEILA, M.; ZHANG, T. (Ed.). **Proceedings of the 38th International Conference on Machine Learning**. PMLR, 2021. (Proceedings of Machine Learning Research, v. 139), p. 12912–12923. Available at: https://proceedings.mlr.press/v139/zhu21e.html.

APPENDIX A – DATASET PREPROCESSING

- **CIFAR-10/CIFAR-100**: in the preprocessing process of both datasets, operations are applied including random cropping to dimensions 32×32 (including padding of 4 pixels on each side of the image), random horizontal flipping with a probability of 50%, and normalization with the following values for CIFAR-10 and CIFAR-100, respectively:
  - $\mu = (0.491, 0.482, 0.446)$, $\sigma = (0.247, 0.243, 0.261)$
  - $\mu = (0.507, 0.486, 0.440)$, $\sigma = (0.267, 0.256, 0.276)$

  For validation/testing samples, we apply only the normalization used for the training samples.
- **Tiny-ImageNet**: following the same procedure as Yu et al. (2019) for the training set samples, we apply random resized cropping to 56×56, random horizontal flipping with a probability of 50%, and normalization with the following values:
  - $\mu = (0.480, 0.448, 0.397)$, $\sigma = (0.230, 0.226, 0.226)$

  Specifically for the validation/testing samples, we resize the samples to 64×64 and then perform a centered crop of 56×56. Finally, we use the same normalization adopted for the training set.
- **Animal-10N**: as done by Xia, Lee & Chen (2023), for the training set, we apply random horizontal flipping (with a probability of 50%) and normalization with the following values:
  - $\mu = (0.485, 0.456, 0.406)$, $\sigma = (0.229, 0.224, 0.225)$

  In the validation/testing set, we exclusively apply the same normalization used for the training samples.
- **Food-101N**: For the preprocessing of this dataset, we adopted the approach proposed by Tan et al. (2021), in which, for the training set, we applied random resized crop to 128×128 with a scale range of $[0.08, 1.0]$, random horizontal flipping with a probability of 50%, and normalization using the following values:
  - $\mu = (0.485, 0.456, 0.406)$, $\sigma = (0.229, 0.224, 0.225)$

  In the test set, we applied image resizing to 128×128 pixels, followed by a center crop of 128×128, and used the same normalization applied to the training set samples.
- **Clothing1M**: following the approaches of Zhang et al. (2021) and Xia, Lee & Chen (2023), we added the 50K clean-labeled samples to the remaining noisy-labeled samples in the training set. For preprocessing, both the training and test images were resized to 256×256, center-cropped to 224×224, and normalized using the following values:

- $\mu = (0.485, 0.456, 0.406)$, $\sigma = (0.229, 0.224, 0.225)$

APPENDIX B – NEURAL ARCHITECTURES

Table 17 provides the architecture details of the 7-layer CNN (WEI et al., 2020; ZHANG et al., 2021) and the 9-layer CNN (HAN et al., 2018; YU et al., 2019) used in experiments with CIFAR-10 and CIFAR-100.

7-layer CNN	**9-layer CNN**
	3×3 Conv, 128 LReLU
3×3 Conv, 64 BN, ReLU	3×3 Conv, 128 LReLU
3×3 Conv, 64 BN, ReLU	3×3 Conv, 128 LReLU
2×2 Max-pooling, stride 2	2×2 Max-pooling, stride 2
	Dropout $p = 0.25$
	3×3 Conv, 128 LReLU
3×3 Conv, 128 BN, ReLU	3×3 Conv, 128 LReLU
3×3 Conv, 128 BN, ReLU	3×3 Conv, 128 LReLU
2×2 Max-pooling, stride 2	2×2 Max-pooling, stride 2
	Dropout $p = 0.25$
3×3 Conv, 196 BN, ReLU	3×3 Conv, 512 LReLU
3×3 Conv, 196 BN, ReLU	3×3 Conv, 256 LReLU
2×2 Max-pooling, stride 2	3×3 Conv, 128 LReLU
	Avg-pooling
Dense: $256 \rightarrow 10/100$	Dense: $128 \rightarrow 10/100$

Table 17 – Details of the architectures used in the experiments

APPENDIX C – TEST ACCURACY MOVING AVERAGES

In Figure 13 we present the test accuracy evolution obtained with Vanilla, Vanilla optimized, Enhanced Co-Teaching, and CCT methods, using the ResNet-18 model on CIFAR-10 and CIFAR-100 datasets, under different levels and types of label noise. A 10-epoch moving average window was applied to smooth fluctuations and highlight learning trends. The Vanilla model follows the standard training configuration proposed by Tan et al. (2021).

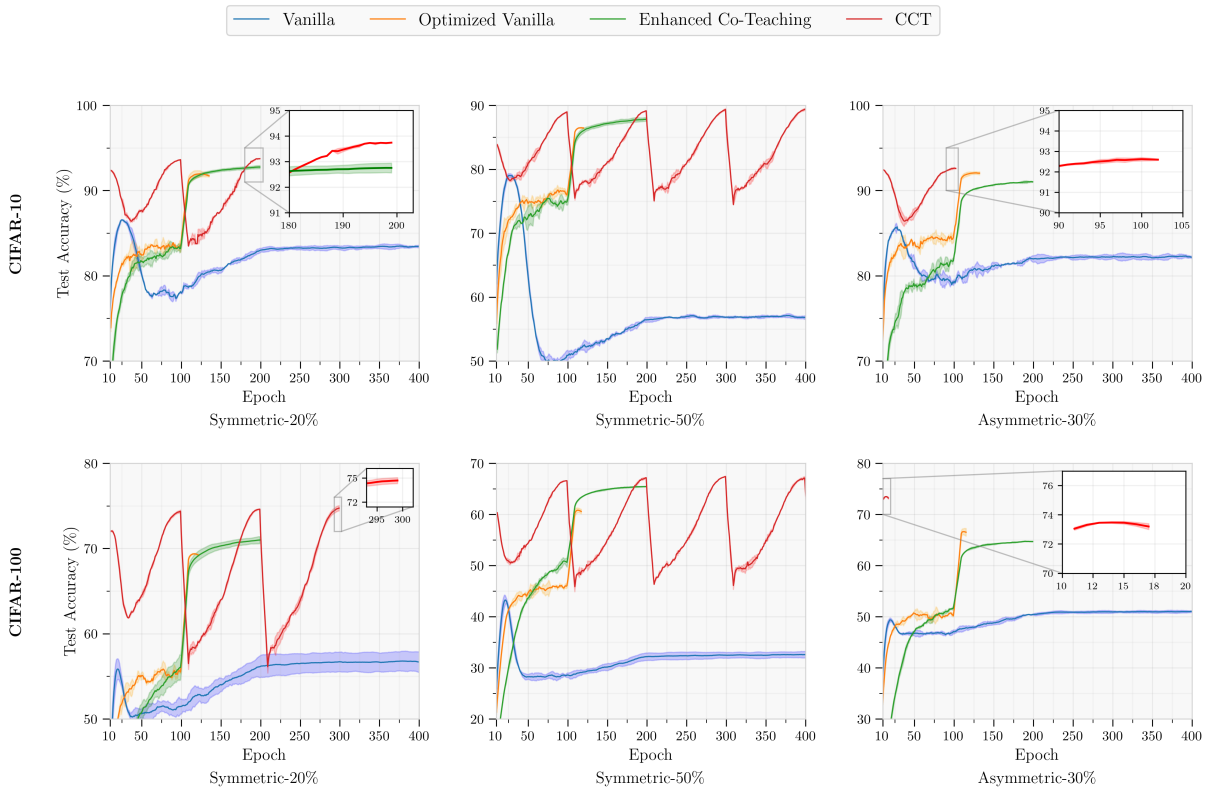


Figure 13 – Moving average of test accuracies (%) on the CIFAR-10 and CIFAR-100 datasets using ResNet-18, for different levels and types of label noise, with a 10-epoch window.