LABEL PROVENANCE LEARNING: MODELING WHERE NOISY LABELS COME FROM

Anonymous authors
Paper under double-blind review

ABSTRACT

Noisy-label learning commonly represents supervision as input–label pairs, discarding the source of each label. Yet large datasets pool labels from different websites, collection protocols, or human annotators, whose errors need not follow the same pattern. We study this setting as *label provenance learning* (LPL): the learner observes (x_i, s_i, a_i), where s_i is a potentially corrupted label, a_i identifies its source, the clean label is latent, and no source is assumed trusted. Provenance can act through two channels, a *routing channel* describing how examples reach sources and a *labeling channel* describing how sources produce labels; each may or may not depend on the provenance, defining four nested prediction structures. We derive provenance-conditioned variational learning (PCVL), which fits any of these structures under a single objective, and a permutation diagnostic that selects among them before training, using no clean labels. On CIFAR-10N the diagnostic detects neither dependency, and the provenance-blind structure has the highest mean accuracy among the four; on dopanim it detects a source-dependent labeling channel, and activating only that channel improves balanced accuracy over both provenance-blind PCVL and DivideMix. The diagnostic's decisions agree with post-hoc oracle audits in every real-data setting. A controlled CIFAR-10 cross that varies the two mechanisms separately shows the labeling channel tracking its generating mechanism, with weaker evidence for routing; a matched stress test shows that shuffling provenance removes the gain. Provenance is therefore worth modeling when its conditional role is supported, rather than activated by default.

1 INTRODUCTION

Learning with noisy labels (LNL) aims to learn a clean-label classifier from potentially corrupted supervision (Song et al., 2023). In its common formulation, the learner receives input–label pairs (x_i, s_i) and must distinguish useful supervision from label errors. This representation leaves out a potentially informative part of the data collection process: where each label came from.

A single dataset often carries labels of several origins. A web-crawled collection may combine examples gathered for the same task from different websites. A manually annotated corpus may combine judgments from annotators with different expertise or readings of the labeling instructions. These sources need not share a noise pattern: their error rates, class confusions, and sensitivity to difficult examples may differ. Treating all labels as an undifferentiated pool discards this structure even when source identities are recorded.

Label provenance is not always informative, nor is it a quality certificate. Knowing that a label came from source a neither reveals whether that particular label is correct nor implies that any source is clean. It instead records the labeling context in which the latent class was measured. The central question is therefore not whether provenance should always be used, but whether it changes the data-collection mechanism in a given dataset.

A motivating case. Consider multi-site chest-radiograph screening. The latent target Y is the true condition, X is the radiograph, A identifies the radiologist or reading service, and S is the recorded diagnosis. Provenance can matter in two ways. First, readers have different miss profiles: a thoracic specialist and a general reader need not interpret the same subtle pattern equally well. Second, cases do not reach readers at random. Poor-quality images may be triaged to a senior reader using

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

information visible in X, while intensive-care and routine pre-operative pathways may be assigned to different reading rotations despite having different disease prevalence. The latter association does not mean that a dispatcher observes the unknown diagnosis; it means that assignment uses pathway information correlated with Y but absent from the pixels. A learner that discards A loses both the reader-specific measurement context and the structure of the collection process.

Motivated by this distinction, we make recorded source context an explicit variable in noisy-label learning and call the resulting setting *label provenance learning* (LPL). The training set is $\mathcal{D} = \{(x_i, s_i, a_i)\}_{i=1}^n$, where $x_i \in \mathcal{X}$ is an input, $s_i \in \{1, \dots, K\}$ is an observed label, $a_i \in \mathcal{A}$ records its provenance, and y_i is the unobserved clean label. The goal is a classifier for Y, not for S:

$$f : \mathcal{X} \to \Delta^{K-1}, \qquad \min_f \mathbb{E}_{(X,Y)}[\ell(f(X), Y)]. \tag{1}$$

The classification task remains unchanged: provenance is available during training, while the deployed classifier predicts from X alone. The conventional LNL interface cannot distinguish two ways in which provenance may matter: which examples reach a source and how that source turns latent classes into observed labels. LPL makes these routing and labeling mechanisms explicit and testable rather than assuming that source identity is always useful. We therefore propose *provenance-conditioned variational learning* (PCVL), a probabilistic model and training procedure that represents both mechanisms as channels in a joint generative model. PCVL trains this model together with a sample-selection objective that anchors the deployed classifier to semantic targets.

Contributions. (1) We formalize label provenance learning as noisy-label learning with recorded sources, in which each example retains one label, no source is assumed clean, source assignment may be informative, and the deployed classifier still predicts from X alone. (2) We identify and model two distinct roles of provenance: a routing channel describing which examples reach which source, and a labeling channel describing how a source turns a latent class into an observed label. Crossing their alternatives yields four configurations in a nested 2×2 family, all instantiated by a unified variational model. (3) We develop a warm-up-stage stratified permutation diagnostic that tests which additional provenance dependencies the data support, without using clean labels or test accuracy. Its decisions match a post-hoc oracle audit in every real setting we study. (4) We evaluate the resulting learner against source-blind and source-aware baselines on CIFAR-10N and dopanim, against a matched 2×2 family of controlled generating mechanisms, and against a true-versus-shuffled provenance intervention, showing that the value of source identity tracks the collection mechanism.

2 Related Work

Noisy-label learning (LNL) studies training from corrupted supervision (Song et al., 2023). Its main approaches include loss correction and transition estimation (Natarajan et al., 2013; Patrini et al., 2017; Li et al., 2021), robust objectives (Zhang & Sabuncu, 2018; Liu et al., 2020), and sample selection or correction (Han et al., 2018; Arazo et al., 2019; Li et al., 2020; Feng et al., 2022). These methods ordinarily operate on (X, S) and are therefore source-blind in our setting.

PCVL is most directly related to deep generative models of label noise. The M2 model scores latent classes through a class-conditional decoder (Kingma et al., 2014); CausalNL and InstanceGM extend this construction to instance-dependent corruption (Yao et al., 2021; Garg et al., 2023). InstanceGM corresponds to the provenance-free $\mathsf{R}_1\mathsf{L}_1$ specialization of our framework. PCVL additionally records the process that assigned an example to a source and allows the labeling mechanism to vary with that source.

Learning from crowds models annotator-specific error processes, from confusion-matrix estimation (Dawid & Skene, 1979; Whitehill et al., 2009; Welinder et al., 2010; Raykar et al., 2010) to neural annotator layers (Rodrigues & Pereira, 2018; Tanno et al., 2019; Chu et al., 2021). That line motivates our labeling channel but identifies it from redundant labels on shared examples, whereas LPL retains one label per example and additionally treats source assignment as a dependency to be tested rather than a design variable. Weak-supervision models (Ratner et al., 2016; 2020), multi-source adaptation (Mansour et al., 2008; Guo et al., 2018), and learning using privileged information (Vapnik & Vashist, 2009) likewise use auxiliary training-time structure. Appendix A gives a fuller comparison.

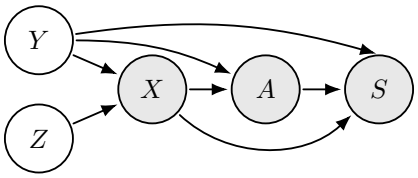

Figure 1: The maximal PCVL graphical model. Shaded nodes are observed during training and unshaded nodes are latent; S and A are absent at deployment. Edges into A define the routing channel, while edges into S define the labeling channel. The nested restrictions considered below remove $Y \to A$ and $A \to S$, respectively.

3 Provenance-Conditioned Variational Learning

PCVL is a unified provenance-conditioned latent-variable framework encompassing four data-collection structures. Its maximal form combines a routing likelihood, a provenance-conditioned labeling likelihood, a conditional variational autoencoder, and a reliable-sample-selection objective. Restricting the conditioning sets of the routing and labeling channels yields a nested 2×2 family, while a model-relative diagnostic selects a parsimonious form before training. Given $\mathcal{D} = \{(x_i, s_i, a_i)\}_{i=1}^n$, PCVL uses the observed label S and provenance A only during training; at deployment, its X-only classifier approximates the latent clean-label posterior $P^\star(Y \mid X)$.

3.1 Generative Model and Provenance Channels

Let $Y \in \{1, \dots, K\}$ denote the latent clean class and let Z be a continuous residual representation. The observed variables are the input X, its provenance $A \in \{1, \dots, M\}$, and the noisy label $S \in \{1, \dots, K\}$. The availability of A alone does not reveal how provenance enters the collection process: neither the dependencies governing source assignment nor the provenance dependence of label errors is known a priori. We therefore represent these two mechanisms by generic routing and labeling likelihoods, R and L, and begin with their maximal conditioning sets:

$$\begin{aligned} &Y \sim p(Y), \quad Z \sim p(Z), \\ &X \sim p(X \mid Y, Z), \quad A \sim \mathsf{R}(A \mid X, Y), \quad S \sim \mathsf{L}(S \mid X, Y, A). \end{aligned} \tag{2}$$

Under this generative ordering, the complete-data joint distribution factorizes as

$$p(x, s, a, y, z) = p(y)\, p(z)\, p(x \mid y, z)\, \mathsf{R}(a \mid x, y)\, \mathsf{L}(s \mid x, y, a). \tag{3}$$

The routing and labeling factors encode distinct roles of provenance, and their conditioning structures are hypotheses about the data-collection process rather than universal assumptions. Section 3.4 introduces a diagnostic procedure for determining which dependencies are supported by a dataset. We first define the channel forms considered in this work; Appendix B discusses their scope and representative collection scenarios.

Routing channel. The routing likelihood describes which source labels an example. We consider two nested families:

$$\mathsf{R}_1 : \; p(A \mid X), \qquad \mathsf{R}_2 : \; p(A \mid X, Y), \tag{4}$$

with $\mathsf{R}_1 \subset \mathsf{R}_2$. R_1 models example-dependent assignment but is constant across candidate classes; random or balanced assignment $p(A)$ is its degenerate special case. R_2 additionally permits $A \not\perp Y \mid X$, so the observed assignment can score candidate classes.

Labeling channel. The labeling likelihood describes how the assigned source produces the observed label. We consider two nested families:

$$\mathsf{L}_1 : \; p(S \mid X, Y), \qquad \mathsf{L}_2 : \; p(S \mid X, Y, A), \tag{5}$$

where L_1 shares one error process across sources and L_2 allows source-specific, instance-dependent corruption. No source is assumed clean and its reliability is not supplied to the learner. Crossing the two routing families with the two labeling families yields four *forms* of the single PCVL model, and we use that term throughout; they are nested specializations of the maximal graph, so they act as diagnostic outcomes and ablation controls rather than four separately tuned methods.

3.2 VARIATIONAL OBJECTIVE

For an observed (x_i, s_i, a_i) the log likelihood is

$$\ell_i = \log \sum_{y=1}^{K} \int p(y)p(z)p(x_i \mid y, z)\mathsf{R}(a_i \mid x_i, y)\mathsf{L}(s_i \mid x_i, y, a_i)\, dz. \tag{6}$$

The sum and integral in Eq. (6) couple the discrete semantic variable Y and the continuous residual Z, making both the evidence and its posterior intractable for the neural model. We therefore introduce an amortized variational distribution $q_{\psi,\rho}(Y, Z \mid X, S, A)$ with adequate support. Rewriting the evidence as an expectation under $q_{\psi,\rho}$ and applying Jensen's inequality gives

$$\begin{aligned}
\ell_i &= \log \mathbb{E}_{q_{\psi,\rho}}\left[\frac{p(Y)p(Z)p(x_i \mid Y, Z)\mathsf{R}(a_i \mid x_i, Y)\mathsf{L}(s_i \mid x_i, Y, a_i)}{q_{\psi,\rho}(Y, Z \mid x_i, s_i, a_i)}\right] \\
&\geq \mathbb{E}_{q_{\psi,\rho}}\Big[\log p(Y) + \log p(Z) + \log p(x_i \mid Y, Z) \\
&\qquad + \log \mathsf{R}(a_i \mid x_i, Y) + \log \mathsf{L}(s_i \mid x_i, Y, a_i) \\
&\qquad - \log q_{\psi,\rho}(Y, Z \mid x_i, s_i, a_i)\Big] =: \mathcal{L}_i(q).
\end{aligned} \tag{7}$$

Thus maximizing $\mathcal{L}_i(q)$ fits the generative factors while bringing the amortized posterior toward the exact posterior. Training equivalently minimizes the negative ELBO.

Variational family. The bound holds for any q with adequate support, but its factorization determines both approximation quality and deployment behavior. By the chain rule,

$$q_{\psi,\rho}(Y, Z \mid X, S, A) = q_\psi(Y \mid X, S, A)q_\rho(Z \mid X, S, A, Y). \tag{8}$$

The graph in Fig. 1 implies $Z \perp (S, A) \mid X, Y$. Mirroring this conditional independence yields the natural structured family

$$q_{\psi,\rho}(Y, Z \mid X, S, A) = q_\psi(Y \mid X, S, A)q_\rho(Z \mid X, Y). \tag{9}$$

This family retains the class-conditional dependence of Z on Y, rather than imposing a fully factorized mean-field approximation. The categorical factor in Eq. (9) is not available at deployment because it requires S and A; a flexible encoder could also copy S directly into the inferred class. Because our target is an X-only clean-label classifier, we follow provenance-free latent-variable LNL methods (Garg et al., 2023; Yao et al., 2021) and impose

$$q^{\mathrm{c}}_{\psi,\rho}(Y, Z \mid X, S, A) := q_\psi(Y \mid X)q_\rho(Z \mid X, Y) \equiv q^{\mathrm{c}}_{\psi,\rho}(Y, Z \mid X). \tag{10}$$

Thus $q_\psi(Y \mid X)$ is both the categorical variational factor and the deployed semantic classifier. This restriction may loosen the bound, but it requires class semantics to be recoverable from X alone and removes the direct $S \to Y$ copying path.[1]

Substituting Eq. (10) into Eq. (7) gives the classification-oriented ELBO

$$\begin{aligned}
\mathcal{L}'_i = \mathbb{E}_{q^{\mathrm{c}}_{\psi,\rho}}\Big[&\log p(x_i \mid Y, Z) + \log \mathsf{R}(a_i \mid x_i, Y) + \log \mathsf{L}(s_i \mid x_i, Y, a_i) \\
&+ \log p(Y) - \log q_\psi(Y \mid x_i) + \log p(Z) - \log q_\rho(Z \mid x_i, Y)\Big].
\end{aligned} \tag{11}$$

Pairing each prior term with its variational factor and collecting the negative log-likelihood terms converts Eq. (11) into a rate–distortion decomposition with three observed-variable distortions and two latent KL rates:

$$-\mathcal{L}'_i = \mathcal{D}_{X,i} + \mathcal{D}_{S,i} + \mathcal{D}_{A,i} + \mathcal{R}_{Y,i} + \mathcal{R}_{Z,i}, \tag{12}$$

$$\begin{aligned}
\mathcal{D}_{X,i} &:= -\mathbb{E}_q \log p(x_i \mid Y, Z), \\
\mathcal{D}_{S,i} &:= -\mathbb{E}_q \log \mathsf{L}(s_i \mid x_i, Y, a_i), \\
\mathcal{D}_{A,i} &:= -\mathbb{E}_q \log \mathsf{R}(a_i \mid x_i, Y), \\
\mathcal{R}_{Y,i} &:= \mathrm{KL}(q_\psi(Y \mid x_i) \| p(Y)), \\
\mathcal{R}_{Z,i} &:= \mathbb{E}_{q_\psi(Y \mid x_i)} \mathrm{KL}(q_\rho(Z \mid x_i, Y) \| p(Z)).
\end{aligned} \tag{13}$$

[1] Appendix G evaluates a Bayes-inspired alternative that combines an X-only classifier with the learned labeling likelihood, $q^{\mathrm{B}}(Y \mid X, S, A) \propto f_\omega(Y \mid X)\mathsf{L}(S \mid X, Y, A)$. Under class-dependent routing, the analogous correction additionally includes $\mathsf{R}(A \mid X, Y)$.

Operational weighted objective. The decomposition separates two roles that are mixed in the expanded ELBO. The distortion terms measure how well the latent representation explains the three observed variables X, S, and A, whereas the rate terms measure the information cost of departing from the priors over Y and Z. Inspired by β-VAE (Higgins et al., 2017), we expose their coefficients separately:

$$\mathcal{J}_{\beta,i} = \lambda_X \mathcal{D}_{X,i} + \lambda_S \mathcal{D}_{S,i} + \lambda_R \mathcal{D}_{A,i} + \beta_Y \mathcal{R}_{Y,i} + \beta_Z \mathcal{R}_{Z,i}. \tag{14}$$

Active observed channels receive unit weight and inactive channels zero. Each distortion is implemented as a per-example mean under its observation model, with reconstruction additionally averaged over observed dimensions so that image size does not become an implicit coefficient. Our primary experiments fix $(\lambda_X, \lambda_S, \lambda_R) = (1, 1, 1)$ for channels present in a model and $(\beta_Y, \beta_Z) = (0, 0)$, based on the sensitivity analysis in Appendix E. With uniform $p(Y)$, $\mathcal{R}_Y = \log K - \mathcal{H}(q_\psi)$, whereas $\mathcal{R}_Z$ limits within-class residual capacity. Their distinct effects are evaluated in the multi-β sensitivity study.

3.3 Identifiability and Latent Separation

Optimizing the variational objective does not by itself identify the intended clean-label semantics. There are two related ambiguities. First, the variational classifier and labeling channel induce the auxiliary predictive mixture

$$\widetilde{p}(s \mid x, a) = \sum_{k=1}^{K} q_\psi(Y = k \mid x)\, \mathsf{L}(s \mid x, k, a). \tag{15}$$

Different classifier–label-channel pairs can induce the same mixture while assigning different meanings to the latent class index. This auxiliary mixture uses the constrained variational weights, not the generative weights $p(Y \mid X, A)$, which also incorporate routing under R_2. Second, the image likelihood can redistribute class information between Y and Z: a more informative residual encoder can compensate for a less semantic Y, while the decoder preserves the same reconstruction likelihood. Neither the posterior factorization nor the variational objective rules out these observationally equivalent solutions.

We therefore introduce a direct semantic identification bias. Sample selection constrains the first ambiguity in Eq. (15) by supplying targets to $q_\psi(Y \mid X)$ without using the learned labeling channel. A change in q_ψ that can be offset by L while preserving the observable likelihood may still increase the selection loss unless it also preserves the selected or corrected targets. The additional objective thus narrows the set of compensating classifier–channel solutions by acting directly on the semantic variable. This remains an inductive bias rather than a recovery guarantee and requires the selector to retain a nonzero clean-label signal.

Following InstanceGM and CausalNL (Garg et al., 2023; Yao et al., 2021), we instantiate this constraint with DivideMix (Li et al., 2020). Its two X-only peer classifiers estimate clean-example probabilities from their loss distributions and use peer-refined targets to separate reliable from unreliable supervision. We apply the resulting DivideMix objective, denoted $\mathcal{L}_{\mathrm{sel}}$, directly to the same semantic classifier represented by $q_\psi(Y \mid X)$. Because this signal is computed independently of R and L, it anchors the interpretation of Y without allowing either learned observation channel to validate its own predictions. Our target is the deployed clean-label classifier rather than a unique decomposition of all information between Y and Z; the latter remains underdetermined by the observable likelihood. Appendix E.4 also reports an exploratory constraint on Z, and Appendix H explores source-conditioned selection; neither extension is part of the core method.

3.4 Testing Which Form Applies

The underlying collection process poses two structural questions: whether provenance carries class information beyond the image ($A \not\perp Y \mid X$), and whether labeling errors depend on provenance ($A \not\perp S \mid X, Y$). These distinguish R_1 from R_2 and L_1 from L_2, respectively. We diagnose these properties before comparing final performance rather than infer them afterward from test accuracy.

Conditional channel diagnostics. We use a stratified permutation test because both questions involve the latent Y and high-dimensional X. After the standard DivideMix warm-up and be-

fore activating the variational branch, we reuse a provenance-blind peer classifier g; the diagnostic therefore requires no additional representation training. The classifier supplies the proxy $\hat{y}_i = \arg\max_k g(k \mid x_i)$ and a penultimate feature vector. Mini-batch k-means partitions these features into 20 strata H_X, and strata with fewer than 30 examples are discarded.

For a discrete target V, a provenance value a, and a stratum h, let n_{avh} be the number of retained examples with $(A, V, H) = (a, v, h)$, with corresponding margins n_{ah}, n_{vh}, and n_h. The plug-in conditional mutual information is

$$\widehat{I}(A; V \mid H) = \sum_h \sum_{a,v} \frac{n_{avh}}{n} \log \frac{n_{avh} n_h}{n_{ah} n_{vh}}, \tag{16}$$

where zero-count cells contribute zero. The two observed statistics are

$$T_R = \widehat{I}(A; \hat{Y} \mid H_X), \qquad T_L = \widehat{I}(A; S \mid H_X, \hat{Y}). \tag{17}$$

Thus T_R is a proxy for $A \not\perp Y \mid X$, distinguishing R_1 from R_2, while T_L approximates $A \not\perp S \mid X, Y$, distinguishing L_1 from L_2. For each statistic, we independently permute A within its conditioning strata and recompute the statistic for B permutations. The finite-sample permutation p-value is

$$\widehat{p} = \frac{1 + \sum_{b=1}^{B} \mathbf{1}[T_b \geq T_{\text{obs}}]}{B + 1}, \tag{18}$$

and we denote the values obtained from T_R and T_L by p_R and p_L, respectively, testing at $\alpha = 0.05$. Stratification preserves the provenance and target margins within each conditioning cell while removing their residual association. When clean labels are available for controlled evaluation, we replace $\hat{Y}$ by Y only for a post-hoc oracle audit; these labels never select or train PCVL. Because $\hat{Y}$ and H_X come from a finite learned representation, the operational tests are model-relative diagnostics rather than exact population tests. Rejection supports the corresponding channel, whereas non-rejection may also reflect proxy error, coarse conditioning, or limited power.

Operational decision rule. The conditional routing diagnostic selects R_2 only when it rejects $A \perp Y \mid X$; otherwise it selects R_1. The labeling diagnostic selects L_2 on rejection and L_1 otherwise. These are pre-training model-selection diagnostics, not identifiability certificates. We report proxy–oracle agreement wherever released clean targets permit this audit. Together, the two decisions select one of the four forms defined in Section 3.1. PCVL starts from the maximal $\mathsf{R}_2\mathsf{L}_2$ graph and removes each unsupported provenance dependency. Under R_1, the independently encoded routing term is class-inert and can be omitted operationally; under R_2, $p(A \mid X, Y)$ remains candidate-class dependent. We report the selected form as PCVL and use the other forms only as matched structural controls. Further interpretation and implementation details appear in Appendices B and F.

3.5 Overall Objective

Combining the multi-β variational objective with the semantic constraint introduced in Section 3.3 gives

$$\mathcal{J} = \sum_{i=1}^{n} \mathcal{J}_{\beta,i} + \lambda_{\text{sel}} \mathcal{L}_{\text{sel}}. \tag{19}$$

The first term contains the three observed-variable likelihoods and the two multi-β latent-rate penalties derived from the ELBO. The second term is an identification bias outside the generative model that directly anchors the semantic variable. The interface coefficient λ_{sel} controls the relative influence of that external learner; it is allowed to depend on the selector family because different selectors expose objectives on different numerical scales, but not on the fitted form. We do not introduce an additional coefficient on the complete variational objective: its likelihood and rate components already have explicit coefficients, so an outer VAE weight would be redundant and would obscure which probabilistic term changed. Algorithm 1 summarizes diagnostic-guided model selection and the subsequent peer-training procedure.

4 Experiments

Our evaluation separates three questions: whether training-only diagnostics select an appropriate form, whether the fitted form responds to the underlying collection mechanism, and whether the

Algorithm 1 Diagnostic-guided PCVL with DivideMix

Input: $\mathcal{D} = \{(x_i, s_i, a_i)\}_{i=1}^{n}$, coefficients in Eq. (14), significance level α
Warm up two provenance-blind X-only DivideMix peers q_{ψ_1} and q_{ψ_2} on the observed labels
Reuse the warm-up predictions and features to compute T_R, T_L and their stratified-permutation p-values
Select R_2 if $p_R < \alpha$ and R_1 otherwise; select L_2 if $p_L < \alpha$ and L_1 otherwise
for each remaining training epoch **do**
  Each peer estimates clean-example probabilities and constructs the DivideMix partition used to train the other peer
  for $m \in \{1, 2\}$ **do**
    Form $\mathcal{L}_{\text{sel}}^{(m)}$ from the peer-selected reliable and unreliable subsets
    Draw differentiable samples from $q_{\psi_m}(Y \mid X)$ and $q_{\rho_m}(Z \mid X, Y)$ and evaluate the active PCVL terms
    Update peer m and its variational and generative components using Eq. (19)
  end for
end for
Output: X-only classifier $\frac{1}{2}[q_{\psi_1}(Y \mid X) + q_{\psi_2}(Y \mid X)]$

selected form improves on strong source-blind and source-aware baselines under real provenance. Controlled CIFAR-10 isolates mechanism changes; CIFAR-10N and dopanim test naturally occurring human provenance; and a matched true-versus-shuffled provenance intervention tests whether gains depend on informative source identity. Appendix E additionally decomposes the variational branch, compares all four channel forms, and reports the latent-rate sensitivity grid.

4.1 Experimental Setup

PCVL implementation. The diagnostic first selects one of the four forms in Section 3.1; PCVL then trains that form with independent semantic, routing, and labeling encoders, using DivideMix as the semantic sample-selection constraint, unit weights on active channels, and $(\beta_Y, \beta_Z) = (0, 0)$. The remaining forms are fitted only as matched structural controls.

Benchmarks and comparisons. Controlled CIFAR-10 crosses balanced or class-dependent routing with shared or source-dependent instance-dependent noise, producing four frozen manifests at approximately 50% corruption. CIFAR-10N (Wei et al., 2022) contributes top-5 and top-20 worker manifests, and dopanim (Herde et al., 2024a) contributes one retained hard label and annotator identity per image. We compare against ERM, GCE, Co-teaching, DivideMix, Source-Aware CE, and the source-aware transition model Annot-Mix.

Evaluation protocol. Matched methods share manifests, data order, augmentations, and initialization whenever their architectures permit. We use no clean training labels, no clean validation source, and no test-based checkpoint selection; structural choices come from the conditional-dependence diagnostics rather than from test accuracy, and coefficient grids are reported only as sensitivity analyses. The primary score is the mean over the final ten epochs of a pre-specified 200-epoch horizon, aggregated over three matched seeds for the primary benchmark and controlled-cross results; auxiliary sensitivity and stress analyses use fixed matched protocols over the same final-ten-epoch window. We report accuracy for CIFAR, balanced accuracy for dopanim, and NLL and ECE for calibration. Reconstruction models, selector thresholds, optimizer settings, and complete provenance construction appear in Appendices C and D.

4.2 Channel Diagnostics and Model Selection

The diagnostics give a clear dataset-specific answer: both CIFAR-10N subsets select the provenance-blind $\mathsf{R}_1\mathsf{L}_1$ form, whereas dopanim selects source-dependent labeling, $\mathsf{R}_1\mathsf{L}_2$; the proxy and oracle decisions agree in all three settings. We obtain the operational decision using the stratified permutation procedure of Section 3.4, with only $\hat{Y}$ and H_X from a provenance-blind reference

classifier. Released clean targets are then substituted for $\hat{Y}$ in a separate oracle audit; they never select a model or enter an optimizer. Table 1 reports both decisions.

Table 1: Deployable proxy diagnostics and post-hoc oracle agreement on real data. p_R tests R_1 against R_2 and p_L tests L_1 against L_2. All tests use $B = 2000$ permutations and $\alpha = 0.05$.

Dataset	Proxy p_R	Proxy p_L	Proxy form	Oracle form	Agreement
CIFAR-10N top5	0.7656	0.1584	$\mathsf{R}_1\mathsf{L}_1$	$\mathsf{R}_1\mathsf{L}_1$	Yes
CIFAR-10N top20	0.7366	0.6647	$\mathsf{R}_1\mathsf{L}_1$	$\mathsf{R}_1\mathsf{L}_1$	Yes
dopanim	0.2694	< 0.001	$\mathsf{R}_1\mathsf{L}_2$	$\mathsf{R}_1\mathsf{L}_2$	Yes

Neither CIFAR-10N subset rejects either null, while dopanim rejects only the labeling null. The agreement supports using the proxy as a practical diagnostic in these three settings. Neither audit detects class-dependent routing here, so this comparison does not assess routing-test power on positive cases.

4.3 Controlled Data–Model Cross

The controlled cross shows that fitted channel structure tracks the generating mechanism most clearly for labeling: every mechanism-matched PCVL form exceeds its within-row DivideMix control, and the matched form is the row maximum in three of four datasets. Table 2 crosses the four retained CIFAR-10 generating mechanisms with the four PCVL forms. The selector and all optimization settings are held fixed across three independent runs. The pre-specified analysis uses matched changes along one axis rather than requiring exact diagonal dominance.

Table 2: Controlled CIFAR-10 cross at approximately 50% label noise. Entries are mean $\pm$ standard deviation of accuracy over three independent runs. Comparisons are made within a data row.

	Fitted PCVL form				
Synthetic dataset	$\mathsf{R}_1\mathsf{L}_1$	$\mathsf{R}_1\mathsf{L}_2$	$\mathsf{R}_2\mathsf{L}_1$	$\mathsf{R}_2\mathsf{L}_2$	DivideMix
SYN-$\mathsf{R}_1\mathsf{L}_1$	95.90 ± 0.10	95.74 ± 0.14	95.88 ± 0.11	95.65 ± 0.13	93.82 ± 0.22
SYN-$\mathsf{R}_1\mathsf{L}_2$	95.04 ± 0.16	95.94 ± 0.09	95.07 ± 0.15	95.87 ± 0.12	94.73 ± 0.19
SYN-$\mathsf{R}_2\mathsf{L}_1$	95.70 ± 0.12	95.83 ± 0.10	95.87 ± 0.13	95.91 ± 0.11	93.82 ± 0.24
SYN-$\mathsf{R}_2\mathsf{L}_2$	95.12 ± 0.17	95.74 ± 0.12	95.18 ± 0.14	95.95 ± 0.08	94.65 ± 0.21

The matched gains over DivideMix range from 1.21 to 2.08 points. The only departure from a row maximum occurs on SYN-$\mathsf{R}_2\mathsf{L}_1$, where the matched form falls short by 0.04 points.

Labeling responds to the generating mechanism more clearly than routing does. Under R_1, switching the fitted labeling channel from L_1 to L_2 changes accuracy by -0.16 points on SYN-$\mathsf{R}_1\mathsf{L}_1$ but by $+0.90$ points on SYN-$\mathsf{R}_1\mathsf{L}_2$, a difference-in-differences of 1.06 points; under R_2 the corresponding interaction is 0.73 points. The routing interactions have the same sign but are much smaller, 0.19 points with L_1 and 0.28 with L_2.

Complementary-source stress test. The matched intervention isolates the value of informative provenance: PCVL improves when A identifies the source-specific confusion, and the gain disappears when that correspondence is broken. We construct a stronger CIFAR-10 intervention with nine balanced sources and approximately 60% noise. When an example is corrupted, source a maps class y to $(y + a + 1) \bmod 10$. The matched control permutes A within each clean class, preserving X, Y, S, and every source–class count while breaking the source–label correspondence. Accordingly, $I(Y; A \mid S)$ decreases from 1.3136 to 0.0066 nats after shuffling.

Quantitatively, true provenance improves accuracy by 1.56 points over DivideMix, 1.85 points over provenance-blind PCVL, and 2.13 points over the shuffled control. Shuffled provenance does not improve on the blind model, isolating the predictive value of the source–label correspondence rather than source identity alone.

Table 3: Complementary-source CIFAR-10 stress test at 59.78% realized noise. Metrics summarize the final ten epochs of the fixed 200-epoch protocol; Std. denotes variation across those epochs.

Form	Accuracy	Std.	NLL	ECE
DivideMix	94.61	0.11	0.2784	0.0968
PCVL, provenance-blind	94.32	0.06	0.2055	0.0124
PCVL, shuffled A	94.04	0.07	0.2228	0.0178
PCVL, true A	**96.17**	0.06	**0.1297**	**0.0065**

4.4 Results on Human Provenance

On human-provenance data, diagnostic-guided PCVL attains the highest primary metric in all three settings. Table 4 compares PCVL with source-blind and provenance-aware baselines. Every method deploys an X-only classifier; the matched four-form comparison is deferred to Appendix E.1.

Table 4: Main results on human-provenance benchmarks. Calibration entries are NLL/ECE. All methods train for 200 epochs. Best accuracy in each setting is bold.

	CIFAR-10N top5		CIFAR-10N top20		dopanim	
Method	Acc.	NLL / ECE	Acc.	NLL / ECE	Bal. Acc.	NLL / ECE
ERM	76.88 ± 0.33	1.186/0.157	81.80 ± 0.34	0.886/0.118	49.72 ± 0.43	2.096/0.298
GCE	79.60 ± 0.40	1.359/0.158	84.94 ± 0.07	0.832/0.109	50.99 ± 0.23	2.461/0.337
Co-teaching	84.51 ± 0.24	0.681/0.081	89.55 ± 0.15	0.405/0.047	53.69 ± 1.35	1.969/0.229
DivideMix	78.15 ± 0.81	0.845/0.045	85.50 ± 0.40	0.585/0.056	64.63 ± 0.11	1.207/0.063
Source-Aware CE	78.38 ± 0.10	1.363/0.172	83.36 ± 0.01	1.002/0.130	50.99 ± 0.48	1.732/0.250
Annot-Mix	77.92 ± 0.31	0.919/0.104	82.42 ± 0.17	0.741/0.077	57.88 ± 0.45	1.694/0.224
PCVL (diagnostic-guided)	$\mathbf{88.59 \pm 0.10}$	0.450/0.049	$\mathbf{92.85 \pm 0.11}$	0.273/0.024	$\mathbf{68.81 \pm 0.18}$	1.028/0.110

The performance pattern follows the diagnosed mechanisms. Provenance need not be activated on CIFAR-10N for PCVL to outperform the listed baselines, whereas dopanim benefits from retaining source identity in the labeling channel. Appendix E.1 further shows that adding the unsupported routing channel does not help.

5 Limitations

LPL assumes provenance is observed, stable, and fixed before the label is produced. Channel estimation requires adequate source–class coverage, and the proxy diagnostic can lose power through representation error or coarse stratification. The evidence for the two channels is uneven. Source-dependent labeling is supported by the controlled cross, by the dopanim diagnostic, and by the resulting accuracy gain, whereas neither real benchmark exhibits class-dependent routing, the controlled routing interactions are small and descriptive, and fitting R_2 without diagnostic support reduces accuracy. Our controlled routing intervention is also a single mechanism. This study therefore establishes routing as a hypothesis the diagnostic can test, not as a demonstrated source of gain. Our comparisons also report means over three seeds without a pre-specified paired significance test, so small differences between PCVL forms should be read as descriptive.

6 Conclusion

We formulate label provenance learning through separate routing and labeling channels. PCVL combines their generative model with a source-blind semantic classifier, sample-selection supervision, and a diagnostic that chooses among nested channel structures. Controlled and human-provenance results show that the value of source identity depends on the collection mechanism: CIFAR-10N supports the provenance-blind form, whereas dopanim benefits from source-dependent labeling. Provenance should therefore be modeled when its conditional role is supported, rather than activated indiscriminately.

AI USE STATEMENT

In this work, we used generative AI tools while preparing this manuscript. Its use was limited to language editing and manuscript organization: tightening prose, improving the abstract and contribution statements, normalizing terminology, and identifying redundant or missing text. All research contributions are the authors' own. The problem formulation, model, diagnostic procedure, experimental design, implementation, and all reported results were produced by the authors without AI generation. No text was accepted without author review, and the authors take full responsibility for the entire content of this paper, including any remaining errors.

ETHICS STATEMENT

This work introduces no new human-subject data and uses only publicly available research benchmarks. Provenance identifiers are treated as anonymous categorical variables; we neither attempt to identify individual annotators nor interpret model-estimated source effects as judgments of personal competence. Nevertheless, provenance may correlate with systematic biases in data collection or annotation. Models using such information should therefore be evaluated for source imbalance and should not be used for consequential assessment of individual annotators.

REPRODUCIBILITY STATEMENT

We provide the generative formulation, variational objective, channel specializations, and optimization protocol in the main paper, with additional derivations, dataset construction details, hyperparameter settings, and sensitivity analyses in the appendix. We also report the random-seed protocol and the aggregation rule used for all main experimental results.

REFERENCES

Eric Arazo, Diego Ortego, Paul Albert, Noel E. O'Connor, and Kevin McGuinness. Unsupervised label noise modeling and loss correction. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 312–321. PMLR, 2019.

Stephen H. Bach, Bryan He, Alexander Ratner, and Christopher Ré. Learning the structure of generative models without labeled data. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 273–282. PMLR, 2017.

Zhendong Chu, Jing Ma, and Hongning Wang. Learning from crowds by modeling common confusions. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(7):5832–5840, 2021. doi: 10.1609/aaai.v35i7.16730.

Alexander Philip Dawid and Allan M. Skene. Maximum likelihood estimation of observer errorrates using the EM algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 28(1):20–28, 1979. doi: 10.2307/2346806.

Xiuwen Fang and Mang Ye. Robust federated learning with noisy and heterogeneous clients. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10072–10081, 2022.

Chen Feng, Georgios Tzimiropoulos, and Ioannis Patras. SSR: An efficient and robust framework for learning with unknown label noise. In *33rd British Machine Vision Conference (BMVC)*, number 372. BMVA Press, 2022.

Arpit Garg, Cuong Nguyen, Rafael Felix, Thanh-Toan Do, and Gustavo Carneiro. Instance-dependent noisy label learning via graphical modelling. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 2288–2298, 2023.

Jiang Guo, Darsh Shah, and Regina Barzilay. Multi-source domain adaptation with mixture of experts. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language*

Processing, pp. 4694–4703. Association for Computational Linguistics, 2018. doi: 10.18653/v1/D18-1498.

Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor W. Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Dan Hendrycks, Mantas Mazeika, Duncan Wilson, and Kevin Gimpel. Using trusted data to train deep networks on labels corrupted by severe noise. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Marek Herde, Denis Huseljic, Lukas Rauch, and Bernhard Sick. dopanim: A dataset of doppelganger animals with noisy annotations from multiple humans. In *Advances in Neural Information Processing Systems Datasets and Benchmarks Track*, volume 37, pp. 51085–51117, 2024a. doi: 10.52202/079017-1617.

Marek Herde, Lukas Lührs, Denis Huseljic, and Bernhard Sick. Annot-mix: Learning with noisy class labels from multiple annotators via a mixup extension. In *ECAI 2024 – 27th European Conference on Artificial Intelligence*, Frontiers in Artificial Intelligence and Applications, pp. 2910–2918. IOS Press, 2024b. doi: 10.3233/FAIA240829.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. β-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with Gumbel-Softmax. In *International Conference on Learning Representations*, 2017.

Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. MentorNet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2304–2313. PMLR, 2018.

Diederik P. Kingma, Danilo J. Rezende, Shakir Mohamed, and Max Welling. Semi-supervised learning with deep generative models. In *Advances in Neural Information Processing Systems*, volume 27, 2014.

Junnan Li, Richard Socher, and Steven C. H. Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. In *International Conference on Learning Representations*, 2020.

Xuefeng Li, Tongliang Liu, Bo Han, Gang Niu, and Masashi Sugiyama. Provably end-to-end label-noise learning without anchor points. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 6403–6413. PMLR, 2021.

Yuncheng Li, Jianchao Yang, Yale Song, Liangliang Cao, Jiebo Luo, and Li-Jia Li. Learning from noisy labels with distillation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 1910–1918, 2017.

Huafeng Liu, Mengmeng Sheng, Zeren Sun, Yazhou Yao, Xian-Sheng Hua, and Heng-Tao Shen. Learning with imbalanced noisy data by preventing bias in sample selection. *IEEE Transactions on Multimedia*, 26:7426–7437, 2024. doi: 10.1109/TMM.2024.3368910.

Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. In *Advances in Neural Information Processing Systems*, volume 33, pp. 20331–20342, 2020.

Yang Liu, Hao Cheng, and Kun Zhang. Identifiability of label noise transition matrix. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 21475–21496. PMLR, 2023.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation with multiple sources. In *Advances in Neural Information Processing Systems*, volume 21, pp. 1041–1048, 2008.

Nathan Mantel and William Haenszel. Statistical aspects of the analysis of data from retrospective studies of disease. *Journal of the National Cancer Institute*, 22(4):719–748, 1959. doi: 10.1093/jnci/22.4.719.

Nagarajan Natarajan, Inderjit S. Dhillon, Pradeep K. Ravikumar, and Ambuj Tewari. Learning with noisy labels. In *Advances in Neural Information Processing Systems*, volume 26, 2013.

Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1944–1952, 2017.

Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1406–1415, 2019.

Alexander Ratner, Stephen H. Bach, Henry Ehrenberg, Jason Fries, Sen Wu, and Christopher Ré. Snorkel: Rapid training data creation with weak supervision. *The VLDB Journal*, 29(2–3):709–730, 2020. doi: 10.1007/s00778-019-00552-1.

Alexander J. Ratner, Christopher M. De Sa, Sen Wu, Daniel Selsam, and Christopher Ré. Data programming: Creating large training sets, quickly. In *Advances in Neural Information Processing Systems*, volume 29, pp. 3567–3575, 2016.

Vikas C. Raykar, Shipeng Yu, Linda H. Zhao, Gerardo Hermosillo Valadez, Charles Florin, Luca Bogoni, and Linda Moy. Learning from crowds. *Journal of Machine Learning Research*, 11: 1297–1322, 2010. URL https://www.jmlr.org/papers/v11/raykar10a.html.

Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to reweight examples for robust deep learning. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4334–4343. PMLR, 2018.

Filipe Rodrigues and Francisco Câmara Pereira. Deep learning from crowds. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, pp. 1611–1618, 2018. doi: 10.1609/aaai.v32i1.11506.

Jakob Runge. Conditional independence testing based on a nearest-neighbor estimator of conditional mutual information. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pp. 938–947. PMLR, 2018.

Jun Shu, Qi Xie, Lixuan Yi, Qian Zhao, Sanping Zhou, Zongben Xu, and Deyu Meng. Meta-Weight-Net: Learning an explicit mapping for sample weighting. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 34(11):8135–8153, 2023. doi: 10.1109/TNNLS.2022.3152527.

Ryutaro Tanno, Ardavan Saeedi, Swami Sankaranarayanan, Daniel C. Alexander, and Nathan Silberman. Learning from noisy labels by regularized estimation of annotator confusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11244–11253, 2019.

Vladimir Vapnik and Akshay Vashist. A new learning paradigm: Learning using privileged information. *Neural Networks*, 22(5–6):544–557, 2009. doi: 10.1016/j.neunet.2009.06.042.

Zhuowei Wang, Jing Jiang, Bo Han, Lei Feng, Bo An, Gang Niu, and Guodong Long. SemiNLL: A framework of noisy-label learning by semi-supervised learning. *Transactions on Machine Learning Research*, 2022. URL https://openreview.net/forum?id=qzM1Tw5i7N.

Jiaheng Wei, Zhaowei Zhu, Hao Cheng, Tongliang Liu, Gang Niu, and Yang Liu. Learning with noisy labels revisited: A study using real-world human annotations. In *International Conference on Learning Representations*, 2022.

Peter Welinder, Steve Branson, Serge J. Belongie, and Pietro Perona. The multidimensional wisdom of crowds. In *Advances in Neural Information Processing Systems*, volume 23, 2010.

Jacob Whitehill, Paul Ruvolo, Tingfan Wu, Jacob Bergsma, and Javier Movellan. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *Advances in Neural Information Processing Systems*, volume 22, 2009.

Jingyi Xu, Zihan Chen, Tony Q. S. Quek, and Kai Fong Ernest Chong. FedCorr: Multi-stage federated learning for label noise correction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10184–10193, 2022.

Yu Yao, Tongliang Liu, Mingming Gong, Bo Han, Gang Niu, and Kun Zhang. Instance-dependent label-noise learning under a structural causal model. In *Advances in Neural Information Processing Systems*, volume 34, pp. 4409–4420, 2021.

Zhilu Zhang and Mert R. Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

A Extended Related Work

Noisy-label learning. Noisy-label learning (LNL) studies training from corrupted labels (Song et al., 2023). Classical approaches derive unbiased or corrected risks under class-conditional corruption (Natarajan et al., 2013; Patrini et al., 2017), while later work estimates transition matrices from anchor points or alternative geometric assumptions (Li et al., 2021; Liu et al., 2023). These results also make the limitation explicit: instance-dependent transitions are generally not identified from a single noisy label without additional assumptions or observations. Clean auxiliary data provide one such source of information (Li et al., 2017; Hendrycks et al., 2018), but LPL assumes none.

Modern deep LNL methods instead exploit optimization dynamics. MentorNet learns a curriculum (Jiang et al., 2018); Co-teaching exchanges small-loss examples between networks (Han et al., 2018); and mixture-based methods estimate sample reliability from the loss distribution (Arazo et al., 2019; Li et al., 2020). ELR (Liu et al., 2020) and GCE (Zhang & Sabuncu, 2018) regularize or robustify the objective, SemiNLL (Wang et al., 2022) organizes LNL as semi-supervised learning, and SSR uses feature-space selection and relabeling (Feng et al., 2022). Sample reweighting can also be learned from auxiliary supervision (Ren et al., 2018; Shu et al., 2019). These methods are source-blind and, in the two-channel language, occupy the $(\mathsf{R}_1, \mathsf{L}_1)$ corner. Class-aware selection for imbalanced noisy data (Liu et al., 2024) further motivates treating class conditioning in a selector as a stratification choice rather than a contribution.

Learning from crowds. A long line of work models annotator-specific error processes. Dawid & Skene (1979) estimate a per-annotator confusion matrix by expectation maximization; later work adds item difficulty and annotator ability (Whitehill et al., 2009), multidimensional annotator representations (Welinder et al., 2010), and joint classifier–annotator estimation (Raykar et al., 2010). Neural formulations attach per-annotator layers or confusion structure to a shared backbone (Rodrigues & Pereira, 2018; Tanno et al., 2019; Chu et al., 2021), and Annot-Mix (Herde et al., 2024b) combines a source-aware transition model with mixup. Our labeling channel L_2 belongs to this family.

Two differences define the LPL setting. First, identification in that literature usually rests on redundancy: several annotators label the same example, so disagreement pins down each one's error process. LPL retains a single label per example, so the labeling channel must instead be identified from an instance-dependent backbone and an external semantic anchor. Second, that literature treats task assignment as a design variable, whom to ask and when, rather than as a generative dependency that a given dataset may or may not exhibit. Our routing channel makes that dependency explicit and testable, which is what distinguishes R_2 from R_1.

Weak supervision. Data programming represents heuristic labeling functions as noisy supervision sources and estimates latent labels with a generative model (Ratner et al., 2016; 2020). Because dependencies among sources can bias that model, subsequent work learns dependency structure from

unlabeled data (Bach et al., 2017). LPL shares the principle that the label acquisition process should be modeled, but does not expose executable labeling functions, abstentions, or multiple source outputs per example; provenance may be a rule but may equally be a worker, vendor, web pipeline, or model.

Deep generative models of label noise. Our variational treatment descends from semi-supervised deep generative models, whose M2 construction enumerates candidate classes and scores each by how well it explains the observed input under a class-conditional decoder (Kingma et al., 2014). CausalNL (Yao et al., 2021) and InstanceGM (Garg et al., 2023) adapt this to instance-dependent label corruption with a latent clean label and latent representation, and InstanceGM is the direct $(\mathsf{R}_1, \mathsf{L}_1)$ ancestor of our model: it contains the enumerate-and-score machinery but observes no provenance in either channel. We use the Gumbel-softmax reparameterization for the discrete latent class (Jang et al., 2017).

Multiple data sources. Multi-source domain adaptation combines populations whose input distributions differ (Mansour et al., 2008; Guo et al., 2018; Peng et al., 2019), while robust federated learning addresses heterogeneous clients whose local labels may be corrupted (Xu et al., 2022; Fang & Ye, 2022). In those settings the source indexes a domain or optimization client. In LPL, A instead identifies the process that produced the observed label for a particular example; routing and labeling can therefore vary even when all inputs belong to one prediction domain and training is centralized.

Privileged information. Learning using privileged information (LUPI) studies features available at training time but not at test time (Vapnik & Vashist, 2009). The routing channel is of this type. This framing also explains why a routing signal that is not recoverable from X can still improve an X-only classifier: it sharpens the training-time target, not the test-time input.

B Scope and Motivation of the Considered Channel Forms

The routing and labeling forms in Section 3.1 are a deliberately scoped set of provenance hypotheses, not an exhaustive enumeration of every parent subset in the graphical model. They isolate whether provenance participates in example assignment and label production while holding the instance-dependent semantic labeling backbone fixed. This focus keeps the comparison on the role of observed source identity rather than on the separate problem of selecting a general noise family.

Routing forms. $\mathsf{R}_1 = p(A \mid X)$ represents assignment based on observable instance properties, such as difficulty-based triage or acquisition characteristics visible in the image. $p(A)$, including randomized or balanced task allocation, is its degenerate special case when the assignment probabilities are constant in X. $\mathsf{R}_2 = p(A \mid X, Y)$ permits residual class association after conditioning on the modeled input. This can arise when referral pathways, preliminary tests, or other omitted allocation variables are correlated with the latent class; it does not require a dispatcher to observe the unknown clean label. The two forms give a nested comparison between a class-inert assignment process and candidate-class-dependent routing evidence. A restricted $p(A \mid Y)$ model could describe explicitly class-stratified assignment, but it is an incomparable branch relative to $p(A \mid X)$ rather than another step in this ladder. Selecting among all such routing restrictions is outside our scope.

Labeling forms. $\mathsf{L}_1 = p(S \mid X, Y)$ represents a source-shared error process, as when provenance is administrative or all sources follow a common labeling mechanism. $\mathsf{L}_2 = p(S \mid X, Y, A)$ allows expertise, protocol, acquisition site, or model version to change how a particular instance and latent class produce the observed label. Neither form assumes that any source is clean. We adopt the same instance-dependent semantic backbone in both forms so that their only structural difference is the provenance input. Class-dependent channels $p(S \mid Y)$ and $p(S \mid Y, A)$ are nested restrictions of these two forms, respectively. Whether the additional dependence on X is necessary is an orthogonal noise-family selection problem that we do not address here. Likewise, a channel such as $p(S \mid X, A)$ can describe observed labels but removes the explicit statistical link from S to the latent clean class; we treat that semantic-informativeness question as outside the LPL comparison.

The 2×2 generating cross covers qualitatively distinct provenance settings: routing may depend only on the example or additionally on the latent class, while labeling may be shared or source-dependent. The diagnostics in Section 3.4 therefore test only the provenance dependencies that distinguish these four collection mechanisms.

C Benchmark and Provenance Construction Details

Canonical synthetic-condition names. Each name SYN-$\mathsf{R}_j\mathsf{L}_k$ records the routing and labeling factors used to generate the data. It does not describe the model subsequently fitted to that dataset. Table 5 lists the complete 2×2 family. For each base image dataset and severity, all four conditions use the same 50,000 CIFAR-10 training images, clean-label order, four equally sized sources, and clean test set. Full SHA-256 digests and the audited arrays are distributed with the manifests; the table gives digest prefixes for compact traceability.

Table 5: Frozen controlled manifests. MI and CMI are plug-in estimates in nats. The source-label signal is $I(A; S \mid Y)$.

Dataset	Noise	$I(A;Y)$	$I(A;S \mid Y)$	SHA-256 prefix
$\mathsf{R}_1\mathsf{L}_1$	0.5050	0.00028	0.00235	`a7c2b2edab08`
$\mathsf{R}_1\mathsf{L}_2$	0.4960	0.00028	0.47217	`b842d41f68d5`
$\mathsf{R}_2\mathsf{L}_1$	0.5050	0.09637	0.00198	`62942bd659a5`
$\mathsf{R}_2\mathsf{L}_2$	0.5010	0.09637	0.44156	`d4a81b4b4647`

Base InstanceGM IDN labels. Let $v_i \in [0, 255]^{3072}$ be the raw vectorized CIFAR-10 image and let y_i be its clean class. Following the released InstanceGM corruption code, we sample an example-specific flip probability $r_i \sim \mathrm{TruncNormal}(0.5, 0.1^2; [0, 1])$ and a fixed tensor $W \in \mathbb{R}^{K \times 3072 \times K}$ with i.i.d. standard-normal entries. For $s \neq y_i$, define

$$\pi_{is} = r_i \frac{\exp(v_i^\top W_{y_i,:,s})}{\sum_{c \neq y_i} \exp(v_i^\top W_{y_i,:,c})}, \qquad \pi_{iy_i} = 1 - r_i. \tag{20}$$

We draw $s_i \sim \mathrm{Cat}(\pi_i)$ with generator seed 123. This single frozen realization has noise rate 0.50502 and supplies the labels for both $\mathsf{R}_1\mathsf{L}_1$ and $\mathsf{R}_2\mathsf{L}_1$ exactly, rather than merely matching them in expectation.

Controlled routing. For R_1, seed 2027 randomly permutes the examples and assigns them round-robin to four sources, producing exactly 12,500 examples per source and making $p(A)$ a balanced degenerate case of $p(A \mid X)$. For R_2, seed 3031 permutes examples within each clean class and assigns exact counts from the following 10×4 count table:

$$C = 1250\,\mathbf{1}_{10\times 4} + 750 \begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & -1 & 1 \\ -1 & 0 & 0 & 1 \\ 1 & 0 & 0 & -1 \\ 1 & 0 & -1 & 0 \\ -1 & 0 & 1 & 0 \end{bmatrix}. \tag{21}$$

Every class still has 5,000 examples and every source 12,500, while $I(A;Y) = 0.09637$ nats. This realizes $p(A \mid Y)$, a controlled special case of $p(A \mid X, Y)$.

Controlled labeling. L_1 uses the same frozen base labels for every source, so provenance does not enter the IDN mechanism. For L_2, we rerun Eq. (20) independently inside each source with nominal rates $(0.2, 0.4, 0.6, 0.8)$ and seeds $123 + 1000(a + 1)$. Thus both error rates and wrong-class distributions can depend on A, while their average target rate remains 0.5. Within each routing

family, the L_1 and L_2 manifests share exactly the same source assignments. Across routing families, the two L_1 manifests share exactly the same noisy-label vector. The audit verifies these invariants, the source counts, label ranges, realized noise rates, MI/CMI contrasts, and file hashes before any training job starts.

CIFAR-N manifests. CIFAR-10N attaches human worker IDs to the CIFAR-10 training images (Wei et al., 2022). We expand each image-batch range in the released side-information files into per-image worker assignments and rank workers by total annotation count, with ascending worker ID breaking ties. The top k workers are remapped in this order to provenance IDs $0, \ldots, k-1$. CIFAR-10N contains three annotation slots. Images with no selected worker are dropped and singleton assignments are retained directly. For overlaps, we process images by increasing number of eligible workers and then image index, assigning each image to the eligible worker minimizing, lexicographically, its current retained count, raw annotation count, worker ID, and slot. Counts are updated immediately, producing a deterministic greedy balance across selected workers. At $k = 20$, this retains 25,690 images, including 6,360 overlap resolutions. The manifests are fixed across model seeds.

dopanim conversion. dopanim contains 15,734 iNaturalist images from 15 species in four visually confusable groups, with 52,795 annotations on 10,484 images from 20 annotators (Herde et al., 2024a). The annotation campaign divided training images into ten task batches assigned to annotators. We use the released data splits and construct a local "rand-1"-style training subset: for every annotated training image, we choose one available annotator uniformly using a fixed seed and retain that annotator's ID as A. The observed label is the class attaining the maximum of the selected annotator's raw per-class likelihood vector, with ties broken uniformly. Duplicate submissions for the same image–annotator pair are resolved by retaining the first record in numeric annotation-ID order. The selected labels and source IDs remain fixed across training runs. CIFAR models operate on 32×32 inputs, whereas dopanim uses 224×224 inputs with an ImageNet-style ResNet-18 and correspondingly resized variational encoder and decoder.

D Experimental Protocol Details

Baselines. ERM uses all observed labels directly. GCE (Zhang & Sabuncu, 2018), Co-teaching (Han et al., 2018), and DivideMix (Li et al., 2020) represent robust loss and sample-selection baselines. Source-Aware CE concatenates a provenance embedding during training and distills to an X-only classifier, without modeling corruption. Annot-Mix (Herde et al., 2024b) is a source-aware transition model with joint image–source–label mixup; the dopanim benchmark reports it as the strongest overall noisy-annotation method across its seven variants (Herde et al., 2024a). We retain its marginal label likelihood without PCVL losses and use our backbone and fixed single-annotation splits. The reported runs use RAdam, learning rate 0.001, batch size 128, mixup parameter 1, and a 200-epoch schedule. No method is told that any source is clean.

PCVL and selector settings. DivideMix is the sample-selection constraint in the primary PCVL experiments and otherwise retains its published optimizer, schedule, augmentations, and selection rule. Its clean-probability threshold is 0.5 on controlled CIFAR-10 and CIFAR-10N and 0.2 on dopanim. PCVL uses independent encoders for the semantic classifier, router, and labeling channel. The diagnostically selected form is reported as PCVL; the remaining forms are matched structural controls.

Training and reporting. Compared methods receive the same frozen manifest, mini-batch order, augmentations, and initialization seed whenever their architectures permit. Coefficients are fixed before inspecting clean human-benchmark test results. We use no clean training labels, clean validation source, or test-selected checkpoint. The reported score averages the final ten epochs of the fixed training horizon; epoch-wise curves are used only to inspect stability. Accuracy is primary on CIFAR and balanced accuracy on dopanim, with NLL and ECE reported for calibration. Coefficient sweeps diagnose numerical sensitivity and do not select channel forms.

Relation to established stratified tests. The diagnostic in Section 3.4 is implemented as a finite-sample stratified permutation test. The Mantel–Haenszel framework provides a classical stratified

alternative (Mantel & Haenszel, 1959), while conditional-permutation tests similarly preserve local covariate structure under the null (Runge, 2018). We use conditional mutual information as the statistic because it applies uniformly to the multi-valued provenance, class, and label variables considered here.

E Additional Experimental Protocol and Tables

E.1 Complete Four-Form Results on Human Provenance

Table 6 fits every form with the same selector and training protocol. At fixed routing, the labeling contrast isolates source-dependent labeling; at fixed labeling, the routing contrast isolates candidate-class-dependent routing.

Table 6: Matched PCVL form comparison. Entries are accuracy (balanced accuracy for dopanim), mean $\pm$ population standard deviation over three seeds.

Form	CIFAR-10N top5	CIFAR-10N top20	dopanim
PCVL-$\mathsf{R}_1\mathsf{L}_1$	$\mathbf{88.59 \pm 0.10}$	$\mathbf{92.85 \pm 0.11}$	67.16 ± 0.55
PCVL-$\mathsf{R}_1\mathsf{L}_2$	88.50 ± 0.21	92.56 ± 0.13	$\mathbf{68.81 \pm 0.18}$
PCVL-$\mathsf{R}_2\mathsf{L}_1$	88.50 ± 0.04	92.67 ± 0.05	67.43 ± 0.62
PCVL-$\mathsf{R}_2\mathsf{L}_2$	88.06 ± 0.06	92.12 ± 0.02	65.58 ± 0.85

On both CIFAR-10N manifests, the diagnostic selects the best mean form, $\mathsf{R}_1\mathsf{L}_1$. The gaps between PCVL forms there are small relative to their seed variation and we ran no pre-specified paired significance test, so we report them as descriptive differences rather than as statistically reliable ones. On dopanim, the selected $\mathsf{R}_1\mathsf{L}_2$ form improves balanced accuracy by 1.65 points over the provenance-blind form and by 4.18 points over DivideMix. Adding the unsupported routing channel reduces performance. Top5 and top20 also provide a source-density replication, although their absolute accuracies are not a pure provenance contrast because the retained image sets differ.

E.2 Decomposing the Variational Branch

We isolate the two observed-variable terms added to the sample-selection learner by crossing $\lambda_X, \lambda_S \in \{0, 1\}$ in the provenance-blind $\mathsf{R}_1\mathsf{L}_1$ form. This is a factorial decomposition, not a sequence of separately tuned models: all four runs use the frozen Step-4 $\mathsf{R}_1\mathsf{L}_1$ manifest (realized noise rate 50.50%), network, optimizer, seed, and 200-epoch schedule, with $\beta_Y = \beta_Z = 0$. Thus the $(0, 0)$ cell is the same-trainer DivideMix control, $(1, 0)$ adds only image reconstruction, $(0, 1)$ adds only the label likelihood, and $(1, 1)$ is the complete PCVL objective for this form.

Table 7: Factorial decomposition of the PCVL variational branch on the frozen Step-4 $\mathsf{R}_1\mathsf{L}_1$ manifest. Results are means over the final ten epochs of the fixed 200-epoch protocol.

Objective	λ_X	λ_S	Accuracy (%)	NLL	ECE
DivideMix	0	0	93.82	0.3077	0.0903
+ reconstruction	1	0	94.12	0.3031	0.0866
+ label likelihood	0	1	95.37	0.1434	**0.0064**
Full PCVL ($\mathsf{R}_1\mathsf{L}_1$)	1	1	**95.90**	**0.1428**	0.0073

Table 7 reports the result, and both observed-variable terms contribute in this matched decomposition. Reconstruction alone improves accuracy by 0.30 points over DivideMix, while the label likelihood provides the larger individual gain of 1.55 points and substantially reduces both NLL and ECE. Their combination reaches the highest accuracy, improving by 2.08 points over DivideMix and by 0.53 points over the label-likelihood-only model. The corresponding factorial interaction is 0.23 points, so the full gain is not merely the sum of two isolated effects. The label-likelihood-only model has the lowest ECE by a small margin, whereas the full model has the lowest NLL.

E.3 Multi-Beta Grid and Latent Diagnostics

The full sweep uses

$$\beta_Y, \beta_Z \in \{0, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}. \tag{22}$$

We run this sensitivity grid on CIFAR-10N-top5 for 50 epochs and three seeds, with the main-table coefficients fixed before inspecting the resulting clean test accuracies. The grid therefore measures sensitivity rather than selecting a model using human-benchmark test labels. Table 8 reports the complete grid. Unit categorical rate regularization pushes $q(Y \mid X)$ close to the uniform prior and sharply reduces accuracy, whereas β_Z continuously controls the capacity of Z with little effect on classification over most of the tested range. Across the same runs, the posterior entropy rises from approximately 0.07 to 2.29 nats when β_Y reaches one, while the analytic Z-KL decreases monotonically as β_Z increases.

Table 8: Accuracy (%) for the β_Y–β_Z sensitivity grid on CIFAR-10N-top5. Each entry is the mean over three 50-epoch runs.

$\beta_Y \backslash \beta_Z$	0	10^{-4}	10^{-3}	10^{-2}	10^{-1}	1
0	85.02	84.87	84.72	84.92	84.46	84.49
10^{-4}	84.85	84.97	85.00	84.91	84.45	84.67
10^{-3}	85.03	84.88	85.09	84.94	84.86	84.66
10^{-2}	84.98	85.04	85.11	84.61	84.78	84.64
10^{-1}	**85.29**	85.03	85.06	85.10	85.14	84.87
1	48.72	45.38	39.45	46.48	38.75	45.36

E.4 Exploratory Residual Constraint

The core PCVL objective anchors the deployed categorical classifier but does not claim a unique decomposition of image information between Y and Z. We explored an additional first-moment constraint on the continuous variable. Let $\mu_{ik} = \mathbb{E}_{q_\rho(Z|x_i,k)}[Z]$, $\pi_{ik} = q_\psi(Y = k \mid x_i)$, $N_k = \sum_i \pi_{ik}$, $m_k = N_k^{-1} \sum_i \pi_{ik}\mu_{ik}$, and $m = n^{-1} \sum_i \sum_k \pi_{ik}\mu_{ik}$. The exploratory penalty is

$$\mathcal{L}_{\text{res}} = \frac{1}{K} \sum_{k=1}^{K} \|m_k - m\|_2^2, \tag{23}$$

with stop-gradient through π. It discourages class-dependent shifts in the aggregate posterior mean, but does not control higher-order or nonlinear class information in Z.

We first cross $\lambda_{\text{sel}} \in \{0, 0.1, 0.3, 1, 3\}$ with $\lambda_r \in \{0, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$ at $d_Z = 25$. Table 9 shows that sample selection supplies the predictive gain, whereas the residual coefficient has no consistent effect.

Table 9: Exploratory interaction between sample selection and residual regularization on CIFAR-10N-top5. Entries are 50-epoch clean test accuracy (mean $\pm$ standard deviation over three seeds).

$\lambda_{\text{sel}} \backslash \lambda_r$	0	10^{-4}	10^{-3}	10^{-2}	10^{-1}	1
0	74.76 ± 4.36	75.80 ± 4.19	76.38 ± 3.58	76.07 ± 3.60	74.38 ± 4.48	74.68 ± 4.58
0.1	82.43 ± 0.18	82.47 ± 0.12	82.28 ± 0.20	82.53 ± 0.04	82.46 ± 0.22	82.54 ± 0.13
0.3	83.01 ± 0.34	83.26 ± 0.20	83.24 ± 0.17	83.29 ± 0.25	83.21 ± 0.20	83.25 ± 0.28
1	84.91 ± 0.31	84.89 ± 0.27	84.81 ± 0.35	84.96 ± 0.08	84.95 ± 0.25	84.91 ± 0.09
3	83.33 ± 0.48	83.51 ± 0.17	82.73 ± 0.63	83.02 ± 0.65	83.06 ± 0.44	83.30 ± 0.62

We next fix $\lambda_{\text{sel}} = 1$ and $\beta_Y = \beta_Z = 0$, and vary $d_Z \in \{25, 50, 100\}$ jointly with $\lambda_r \in \{0, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$. In addition to accuracy, the released result file reports the residual loss, analytic Z-KL, reconstruction NLL, test NLL, and 15-bin ECE.

Table 10 reports this grid. At $\lambda_r = 0$, increasing d_Z from 25 to 100 raises the total analytic Z-KL from 341.39 to 1128.47 and improves reconstruction NLL from -0.64 to -0.69, while accuracy

Table 10: CIFAR-10N-top5 accuracy across the 50-epoch latent-capacity and residual-regularization grid. Rows vary d_Z and columns vary λ_r; entries are mean $\pm$ standard deviation over three seeds.

$d_Z \backslash \lambda_r$	0	10^{-4}	10^{-3}	10^{-2}	10^{-1}	1
25	85.02 ± 0.14	84.73 ± 0.49	84.86 ± 0.35	85.11 ± 0.36	85.01 ± 0.30	84.90 ± 0.22
50	85.09 ± 0.19	84.85 ± 0.06	85.07 ± 0.04	84.97 ± 0.11	84.85 ± 0.15	84.85 ± 0.21
100	85.08 ± 0.18	85.02 ± 0.23	85.13 ± 0.03	84.88 ± 0.14	85.15 ± 0.03	85.15 ± 0.25

remains unchanged. At every dimension, positive λ_r sharply contracts the residual first-moment statistic. The total Z-KL initially decreases but can rebound under stronger residual weights, showing that matching class-conditional means does not control all information carried by Z. Across all 18 capacity–weight cells, accuracy remains in $[84.73, 85.15]\%$, test NLL in $[0.5142, 0.5325]$, and ECE in $[0.0459, 0.0502]$, without a monotonic trend. Although $\mathcal{L}_{\text{res}}$ itself falls by orders of magnitude, it improves neither prediction nor calibration. We therefore exclude it from the core PCVL objective and retain this study as a negative exploratory result.

F Neural Parameterization and Stochastic Training

We instantiate the abstract generative factors as $p_\theta(X \mid Y, Z)$, $p_\phi(S \mid X, Y, A)$, and $\mu_{\omega_\mathsf{R}}(A \mid X, Y)$; the distinct symbols identify their neural parameter blocks, while q_ψ and q_ρ remain variational distributions. The L_2 labeling channel is a free neural logit function rather than a transition matrix:

$$p_\phi(S = \cdot \mid x, Y = k, A = a) = \operatorname{softmax} r_\phi(h_\kappa(x), e_k, e_a), \tag{24}$$

where $h_\kappa(x)$ is an image representation and e_k, e_a are learned class and provenance embeddings. Setting L_1 removes e_a. The source-blind semantic classifier, router, and labeling channel use independent observed-X encoders:

$$q_\psi(Y = \cdot \mid x) = \operatorname{softmax} c_\psi(h_\psi(x)), \tag{25}$$

$$\mu_{\omega_\mathsf{R}}(A = \cdot \mid x, Y = k) = \operatorname{softmax} g_{\omega_\mathsf{R}}(h_\mathsf{R}(x), e'_k). \tag{26}$$

Under R_1, dropping e'_k makes the routing score constant across candidate classes. Because its encoder is independent, this separable nuisance term does not update the classifier or labeling channel and is omitted in the operational objective. Under R_2, the class embedding lets the observed provenance score candidate classes. The router always conditions on the observed input x, not on the decoder reconstruction.

Both channels are trained by scoring the observed value of their output variable. The labeling channel contributes $\log p_\phi(s_i \mid x_i, \hat{Y}, a_i)$ at the observed s_i, and the routing channel contributes $\log \mu_{\omega_\mathsf{R}}(a_i \mid x_i, \hat{Y})$ at the observed a_i. The router is never asked to produce a substitute $\hat{A}$ for use elsewhere: A is observed during training, and feeding a prediction built from $\hat{Y}$ back into a term already conditioned on Y would create an unanchored self-validating loop.

Writing $\pi_{ik} = q_\psi(Y = k \mid x_i)$ and $q_{\rho,ik} = q_\rho(Z \mid x_i, k)$, define

$$A_{ik} = \mathbb{E}_{q_{\rho,ik}}[\log p_\theta(x_i \mid Z, k)] - \mathrm{KL}(q_{\rho,ik} \| p(Z)). \tag{27}$$

The operational ELBO can then be expanded over classes as

$$\mathcal{L}'_i = \sum_{k=1}^{K} \pi_{ik} \Big[\log p(k) + A_{ik} + \log \mu_{\omega_\mathsf{R}}(a_i \mid x_i, k) + \log p_\phi(s_i \mid x_i, k, a_i) - \log \pi_{ik} \Big]. \tag{28}$$

Exact evaluation requires K encoder–decoder passes. Our implementation uses a single-sample differentiable categorical estimate for Y and a reparameterized Gaussian sample for Z. The source-blind restriction on $q_\psi(Y \mid X)$ is deliberate rather than the tightest family available; Appendix G evaluates a channel-conditioned, Bayes-inspired alternative that performed worse in the reported experiment.

F.1 Optimization Effects of the Routing Forms

The routing-dependent part of the negative ELBO in Eq. (12) is

$$\mathcal{D}_{A,i} = -\sum_{k=1}^{K} \pi_{ik} r_{ik}, \qquad r_{ik} := \log \mathsf{R}(a_i \mid x_i, k), \tag{29}$$

where $\pi_{ik} = q_\psi(Y = k \mid x_i)$. This expression makes the optimization effect of each routing form explicit.

R_1: an X-dependent but class-inert objective. Because $\mathsf{R}(a_i \mid x_i, k) = \mathsf{R}(a_i \mid x_i)$ is constant across k, Eq. (29) reduces to

$$\mathcal{D}_{A,i} = -\log \mathsf{R}(a_i \mid x_i), \qquad \nabla_\psi \mathcal{D}_{A,i} = 0. \tag{30}$$

Thus R_1 cannot directly change the categorical posterior. With an independent X-router it is inert for the deployed classifier, so its separable routing term is omitted. Constant $p(A)$ is included as the special case in which this nuisance router ignores X.

R_2: class-dependent candidate reweighting. Now r_{ik} varies with k. If t_{ij} is the semantic logit whose softmax is π_{ij}, then

$$\frac{\partial \mathcal{D}_{A,i}}{\partial t_{ij}} = -\pi_{ij} \left(r_{ij} - \sum_{k=1}^{K} \pi_{ik} r_{ik} \right). \tag{31}$$

The observed provenance therefore increases candidate classes that better explain the assignment and suppresses those that explain it worse. Unlike R_1, this is a direct gradient into $q_\psi(Y \mid X)$.

Proposition 1 (Routing-form gradients)**.** *Under Eq.* (10) *and independent channel encoders,* R_1 *contributes no gradient to the semantic classifier, whereas* R_2 *contributes the direct class-dependent gradient in Eq.* (31)*. All routing effects are training-time only because A is absent from the deployed posterior.*

F.2 Extended Interpretation of Routing Gradients

The absence of A from the deployed posterior is forced by the task: a new test input has not been assigned to any training-time labeling source. Routing therefore acts as privileged information only during training (Vapnik & Vashist, 2009). Under our independent parameterization, an X-only router has no gradient path to either the semantic classifier or labeling channel. Under R_2, the observed provenance instead scores candidate classes directly. A shared encoder would introduce an additional multi-task regularization path, but that is an architectural choice rather than a dependency implied by the ELBO, so it is not part of the primary model.

For R_2, informational and estimation regimes should be distinguished. When $H(Y \mid X) > 0$, the condition $I(A; Y \mid X) > 0$ means that provenance carries class information unavailable in X. When Y is effectively determined by X, as in curated image benchmarks, this conditional mutual information vanishes; routing can still help a finite-sample, finite-capacity learner estimate the mapping, but it does not add population-level information. Controlled CIFAR experiments primarily probe this estimation regime, whereas real annotation pipelines may exhibit the informational regime.

A class-dependent routing component can hurt when the targeted dependence is weak or absent and finite-sample density is insufficient to prevent fitting source noise, because its likelihood then distorts candidate-class gradients. This differs from diminishing returns when two valid channels resolve overlapping examples, and from estimation error when a genuinely present but flexible channel is fitted from sparse source–class cells. These distinctions motivate the diagnostics in Section 3.4.

G Tempered Bayes Correction as a Posterior Ablation

Eq. (10) restricts the variational posterior to a source-blind $q_\psi(Y \mid X)$, sampled independently of the observed label and provenance. Since the model itself conditions S and A on Y, one may

ask whether the observed channels should correct the source-blind classifier inside the variational posterior. We define the tempered family

$$q_{ik}^{\mathrm{B}}(\tau_R, \tau_L) \propto \bar{f}_\omega(k \mid x_i)\, \mathsf{R}(a_i \mid x_i, k)^{\tau_R} \mathsf{L}(s_i \mid x_i, k, a_i)^{\tau_L}. \tag{32}$$

For a channel that is absent or constant across candidate classes, its factor cancels during normalization. The main source-blind posterior is exactly $(\tau_R, \tau_L) = (0, 0)$, while $(1, 1)$ includes both available channel factors without tempering. This construction is inspired by Bayes' rule: the discriminative classifier need not equal the generative model's $p(Y \mid X)$. Intermediate temperatures vary the influence of channel evidence relative to image semantics. In this ablation we vary only (τ_R, τ_L) while holding the generative model, likelihood coefficients, selector, and optimization protocol fixed.

Table 11 reports a matched two-dimensional sensitivity screen on the controlled $\mathsf{R}_2\mathsf{L}_2$ CIFAR-10 condition. Both channel factors vary across candidate classes in this setting, so the experiment can separately control their contributions to posterior inference. Every cell uses the same frozen data, fitted model family, selector, and 100-epoch budget; only the two posterior temperatures change.

Table 11: Tempered Bayes-posterior sensitivity on controlled $\mathsf{R}_2\mathsf{L}_2$ CIFAR-10. Entries are mean accuracy (%) over the final ten epochs of matched 100-epoch runs. Rows vary routing evidence τ_R and columns vary labeling evidence τ_L; $(0, 0)$ is the source-blind posterior used by PCVL.

$\tau_R \backslash \tau_L$	0	0.01	0.03	0.1	0.3	1
0	92.15	92.07	92.11	73.22	88.78	69.43
0.01	92.00	92.02	92.24	71.84	67.97	92.00
0.03	92.33	91.80	92.26	65.69	10.00	91.88
0.1	91.52	91.67	54.09	56.92	35.88	90.30
0.3	91.51	91.60	91.30	85.72	76.79	39.77
1	91.28	87.26	90.39	35.56	83.44	10.00

Small temperatures remain close to the 92.15% source-blind control: the best cell reaches 92.33%, but the gain is only 0.18 points and does not persist as either temperature increases. Stronger correction is highly unstable, with several settings falling below 70% and the untempered $(1, 1)$ posterior reaching chance-level accuracy. This non-monotonic behavior is consistent with joint optimization allowing a learned channel to move the posterior toward classes that explain the observed noisy label without improving clean-label prediction, although accuracy alone cannot identify the optimization mechanism. The screen therefore favors the stable source-blind family in Eq. (10); it does not establish that channel-conditioned posteriors are generally inferior. We retain Eq. (32) as a sensitivity analysis rather than a required component of PCVL.

H Exploratory Source-Conditioned Sample Selection

The main method uses pooled sample selection to anchor the semantic classifier. As a separate exploratory extension, we ask whether DivideMix's scalar-loss mixture can instead be estimated independently within each provenance group. We call this direct source-wise variant Provenance-DivideMix. Both methods retain the same backbone, optimizer, peer-training procedure, and 200-epoch budget; only the population used to fit the loss mixture changes. This extension is not part of PCVL's probabilistic model or primary objective.

Table 12 reports both datasets. On CIFAR-10N top5, Provenance-DivideMix improves accuracy by 2.24 points over DivideMix, whereas on dopanim it changes accuracy by -0.23 points. These single-seed results suggest that source-wise fitting may provide a limited gain on particular datasets, but they do not support a general improvement claim. Nor is the construction applicable to every selector. DivideMix estimates a scalar-loss mixture, so changing its reference population leaves the rest of its training rule intact. In contrast, splitting a feature-based selector such as SSR by provenance changes its KNN graph and can destroy the global class geometry on which selection depends. Provenance-DivideMix is therefore a simple feasibility exploration rather than a general provenance wrapper for noisy-label methods. Establishing when source-wise selection is beneficial, and how to incorporate provenance into graph-based selectors without fragmenting their representation space, remains future work.

Table 12: Preliminary source-conditioned DivideMix results. Accuracy, NLL, and ECE are means over the final ten epochs of one seed; Std. is the corresponding within-run standard deviation of accuracy, not cross-seed uncertainty.

Dataset	Selector	Accuracy (%) ↑	Std.	NLL ↓	ECE ↓
CIFAR-10N top5	DivideMix	76.87	0.19	0.8810	0.0461
	Provenance-DivideMix	79.11	0.15	0.8024	0.0453
dopanim	DivideMix	64.90	0.37	1.1925	0.0762
	Provenance-DivideMix	64.67	0.40	1.1998	0.0777