

000 RECONSTRUCTIONNET: A NEURAL NETWORK AR- 001 CHITECTURE FOR UNCERTAINTY-AWARE PREDI- 002 CIONS WITH EXPLAINABILITY 003

004 **Anonymous authors**
 005 Paper under double-blind review
 006

011 ABSTRACT 012

013 Uncertainty estimation quantifies a model’s confidence in its predictions, foster-
 014 ing calibrated trust among users. Existing approaches face two key limitations: (1)
 015 most capture only a single type of uncertainty, and (2) they incur additional train-
 016 ing or inference overhead. We propose ReconstructionNet, a neural network that
 017 addresses these limitations by modeling the joint input–output distribution with
 018 class-specific autoencoders. This enables simultaneous prediction and estimation
 019 of both aleatoric and distributional uncertainty in a single pass. Across five real-
 020 world datasets, ReconstructionNet matches or surpasses baseline classifiers while
 021 producing uncertainty estimates with greater reliability, selectivity, robustness to
 022 false negatives, and strong OOD detection. Furthermore, ReconstructionNet’s ar-
 023 chitecture naturally supports uncertainty explanations, revealing how individual
 024 features contribute to prediction uncertainty without extra computation. MNIST
 025 and ISIC experiments demonstrate that these explanations highlight misclassified
 026 regions consistent with human intuition. Together, these contributions establish
 027 ReconstructionNet as a unified framework for trustworthy and interpretable AI.
 028

029 1 INTRODUCTION 030

031 Uncertainty estimation refers to the task of quantifying how uncertain a machine learning model is
 032 about its prediction for each instance. Reliable uncertainty estimates foster calibrated trust by alert-
 033 ing users to cases where the model is likely to be uncertain and erroneous (Toh et al., 2025). Com-
 034 mon methods for uncertainty estimation include Bayesian Neural Networks (BNNs) (Jospin et al.,
 035 2022), Monte Carlo Dropout (Gal & Ghahramani, 2016) and Deep Ensemble (Lakshminarayanan
 036 et al., 2017). These methods quantify one or more of three main types of uncertainty (Malinin &
 037 Gales, 2018): 1) Aleatoric (data), 2) Epistemic (model), and 3) Distributional uncertainty. While
 038 existing work in uncertainty estimation shows promise, it often faces several limitations. Most un-
 039 certainty estimates quantify only a single type of uncertainty and are unable to differentiate between
 040 various sources of uncertainty. Furthermore, many uncertainty estimation methods incur increased
 041 training and inference time.

042 We introduce **ReconstructionNet**, a neural network architecture designed to address the aforemen-
 043 tioned limitations. ReconstructionNet quantifies aleatoric and distributional uncertainty by modeling
 044 the joint input–output distribution with class-specific autoencoders, effectively measuring the dis-
 045 tance of an instance from the training data of each class. This design reduces epistemic uncertainty
 046 by constraining the state space and allows ReconstructionNet to distinguish between aleatoric and
 047 distributional uncertainty.

048 Beyond identifying when a model is uncertain, understanding which features contribute to prediction
 049 uncertainty is equally valuable, giving rise to the emerging field of uncertainty explanation (Wang
 050 et al., 2025; Antorán et al., 2021; Fan, 2025). Uncertainty explanations are typically represented as
 051 vectors of real values, where each value quantifies the significance and impact of an input feature on
 052 the model’s uncertainty. State-of-the-art methods include applying out-of-the-box eXplainable Ar-
 053 tificial Intelligence (XAI) techniques, such as Integrated Gradients (IG) (Sundararajan et al., 2017),
 054 to explain existing uncertainty estimates (Mougan & Nielsen, 2023; Iversen et al., 2024). Other
 055 approaches involve observing how input perturbations affect prediction uncertainty (Antorán et al.,

054 2021; Wang et al., 2025) and learning which features significantly reduce uncertainty. However,
 055 these methods require extra modules on top of uncertainty estimation, increasing inference time.
 056

057 ReconstructionNet’s design enables inbuilt uncertainty explanation: each class-specific autoencoder
 058 produces feature-wise reconstruction errors scaled by learned error weights for classification. Those
 059 weighed reconstruction errors quantify each feature’s contribution to the uncertainty, providing un-
 060 certainty explanations without additional modules or computation.

061 The contributions of this research are as follows:

- 062 1. Propose ReconstructionNet, a neural architecture that minimises epistemic uncertainty
 063 while quantifying and explaining both aleatoric and distributional uncertainty.
- 064 2. Provide a theoretical evaluation of ReconstructionNet’s uncertainty explanations.
- 065 3. Demonstrate the efficacy of ReconstructionNet for prediction, uncertainty estimation, and
 066 explanation on real-world applications in healthcare and finance.

069 2 RELATED WORK

071 2.1 UNCERTAINTY ESTIMATION

073 **Definition 1 (Uncertainty Estimation)** *For instance $\mathbf{x} \in \mathbb{R}^d$ and a model f , an uncertainty esti-*
 074 *mator $\sigma(\mathbf{x}; f) : \mathbb{R}^d \rightarrow \mathbb{R}$ assigns a real-valued measure of the prediction uncertainty for \mathbf{x} .*

075 Uncertainty estimates generally aim to quantify three types of uncertainty:

- 077 1. **Aleatoric:** Uncertainty arising from inherent noise in the training data. In classification
 078 problems, this manifests as overlapping classes.
- 079 2. **Epistemic:** Uncertainty from inadequate model parameter fit, reducible by expanding the
 080 dataset or narrowing the hypothesis space.
- 081 3. **Distributional:** Uncertainty caused by data shifts between the training and prediction set.

083 **Bayesian methods** capture epistemic uncertainty by modeling distributions over network weights.
 084 For instance, Bayesian Neural Networks (BNNs) (Jospin et al., 2022) sample from these distribu-
 085 tions to produce multiple predictions, with the variability reflecting uncertainty. Deep Ensembles
 086 (Lakshminarayanan et al., 2017) and Monte Carlo Dropout (MCD) (Gal & Ghahramani, 2016) ap-
 087 proximate this sampling via multiple models or stochastic weight activations within a single net-
 088 work. Despite their effectiveness, these approaches are computationally expensive, often requiring
 089 multiple forward passes or extended training.

090 **Evidential Deep Learning (EDL)** methods (Ulmer et al., 2023; Amini et al., 2020) are more com-
 091 putationally efficient than Bayesian approaches, requiring only a single forward pass and one trained
 092 model. They assume the output follows a well-characterized distribution, predicting the parameters
 093 of the output distribution from which uncertainty can be derived. For classification, this corre-
 094 sponds to Dirichlet parameters (Sensoy et al., 2018), with Posterior Networks (PN) (Charpentier
 095 et al., 2020) extending this via normalizing flows. Despite capturing aleatoric and distributional
 096 uncertainty, these methods rely on restrictive assumptions about the output distribution.

097 **Deterministic Uncertainty Methods (DUMs)** (Postels et al., 2022; Charpentier et al., 2023; Ze-
 098 lenka et al., 2023a) make minimal assumptions about the output distribution, estimating distribu-
 099 tional uncertainty as the distance of an instance from the training set. A recent DUM, the Re-
 100 construction Uncertainty Estimate (RUE) (Wang et al., 2024; Korte et al., 2024), uses a decoder
 101 to reconstruct inputs from the latent representation, with reconstruction error serving as the un-
 102 certainty measure. ReconstructionNet extends RUE by also capturing aleatoric uncertainty alongside
 103 distributional uncertainty.

105 2.2 UNCERTAINTY EXPLANATIONS

106 Knowing when a model is unreliable is valuable, but uncertainty explanations provide deeper insight
 107 by quantifying the contribution of each feature to the model’s overall uncertainty.

108 **Definition 2 (Uncertainty Explanation)** Given an instance $\mathbf{x} \in \mathbb{R}^d$, a model f , and an uncertainty
 109 estimator $\sigma(\mathbf{x}; f) : \mathbb{R}^d \rightarrow \mathbb{R}$, an uncertainty explanation method $\zeta(\mathbf{x}; f, \sigma) : \mathbb{R}^d \rightarrow \mathbb{R}^d$ assigns each
 110 input feature a real value reflecting its contribution to $f(\mathbf{x})$'s uncertainty.

111 Uncertainty explanation is a nascent field with two primary approaches:

113 **Gradient-Based Methods:** (Mougan & Nielsen, 2023; Iversen et al., 2024) applied standard eX-
 114 plainable Artificial Intelligence (XAI) methods, such as Integrated Gradients (IG) (Sundararajan
 115 et al., 2017), to explain uncertainty estimates. While easy to implement, gradient-based explana-
 116 tions can be sparse due to vanishing gradients, making them difficult to interpret.

117 **Perturbation-Based Methods** (Antorán et al., 2021; Wang et al., 2025) assess each feature's contri-
 118 bution to uncertainty by perturbing inputs and measuring the impact on the uncertainty score. Their
 119 accuracy depends on the number of perturbations, making them computationally expensive.

120 Both methods require an additional module for uncertainty explanation, increasing inference time.
 121 In contrast, (Zelenka et al., 2023b) computes predictions, uncertainty, and explanations in a sin-
 122 gle forward pass using one model. Based on prototype networks (Snell et al., 2017), it classifies
 123 instances by similarity to class prototypes, with uncertainty explanations derived from the inner
 124 product between the predicted prototype and the instance's feature map. Similarly, Reconstruc-
 125 tionNet leverages its architecture to provide ante-hoc uncertainty explanations efficiently during
 126 inference.

128 3 METHODOLOGY

130 We present ReconstructionNet, a neural network architecture which offers the following features:

- 132 1. Uses the joint input-output probability for classification.
- 133 2. Quantifies distributional and aleatoric uncertainty.
- 134 3. Generates uncertainty explanations.

136 3.1 CLASSIFICATION

138 Consider a classification dataset with N instances and C classes, where each instance i has an input
 139 vector $\mathbf{x}_i \in \mathbb{R}^d$ and label y_i . Let \mathcal{X}_j denote the set of training instances with target label j , containing
 140 N_j instances. For each instance i and class j , the true and predicted class probabilities are p_{ij} and
 141 \hat{p}_{ij} , respectively.

142 Traditional neural networks predict the conditional probability of class j given instance \mathbf{x}_i . The
 143 class with the highest conditional probability is then selected as the final predicted class \hat{y}_i :

$$145 \quad \hat{y}_i = \arg \max_{j \in \{1, \dots, C\}} \Pr(\hat{y}_i = j \mid \mathbf{x}_i). \quad (1)$$

147 ReconstructionNet differs from traditional feedforward neural network classifiers in its inference
 148 process. ReconstructionNet instead predicts the joint probability of the target \hat{y}_i and input \mathbf{x}_i and
 149 the target class with the highest joint probability is the final predicted class (Equation 2):

$$150 \quad \hat{y}_i = \arg \max_{j \in \{1, \dots, C\}} \Pr(\hat{y}_i = j, \mathbf{x}_i). \quad (2)$$

152 Equation 2 is a valid classification formulation, equivalent to Equation 1 via Bayes' theorem:

$$153 \quad \arg \max_j \Pr(\hat{y}_i = j, \mathbf{x}_i) = \arg \max_j \Pr(\hat{y}_i = j \mid \mathbf{x}_i) \Pr(\mathbf{x}_i) = \arg \max_j \Pr(\hat{y}_i = j \mid \mathbf{x}_i). \quad (3)$$

155 To model joint input-output probability, we adopt the model architecture in Figure 1a. Given a
 156 classification problem with C classes, we construct C autoencoders g_1, \dots, g_C and train them si-
 157 multaneously, such that each autoencoder models the joint probability $\Pr(\hat{y}_i = j, \mathbf{x}_i)$. During
 158 inference, the model computes feature-wise reconstruction errors \mathbf{e}_{ij} for each autoencoder (Equa-
 159 tion 4). $\hat{\mathbf{x}}_{ij} = g_j(\mathbf{x}_i)$ is the reconstructed input of instance i by class- j autoencoder g_j ; $\hat{\mathbf{x}}_{ij}^k$ is the
 160 reconstruction for feature k .

$$161 \quad \mathbf{e}_{ij} = \begin{bmatrix} (\hat{\mathbf{x}}_{ij}^1 - \mathbf{x}_i^1)^2 & \dots & (\hat{\mathbf{x}}_{ij}^d - \mathbf{x}_i^d)^2 \end{bmatrix} \quad (4)$$

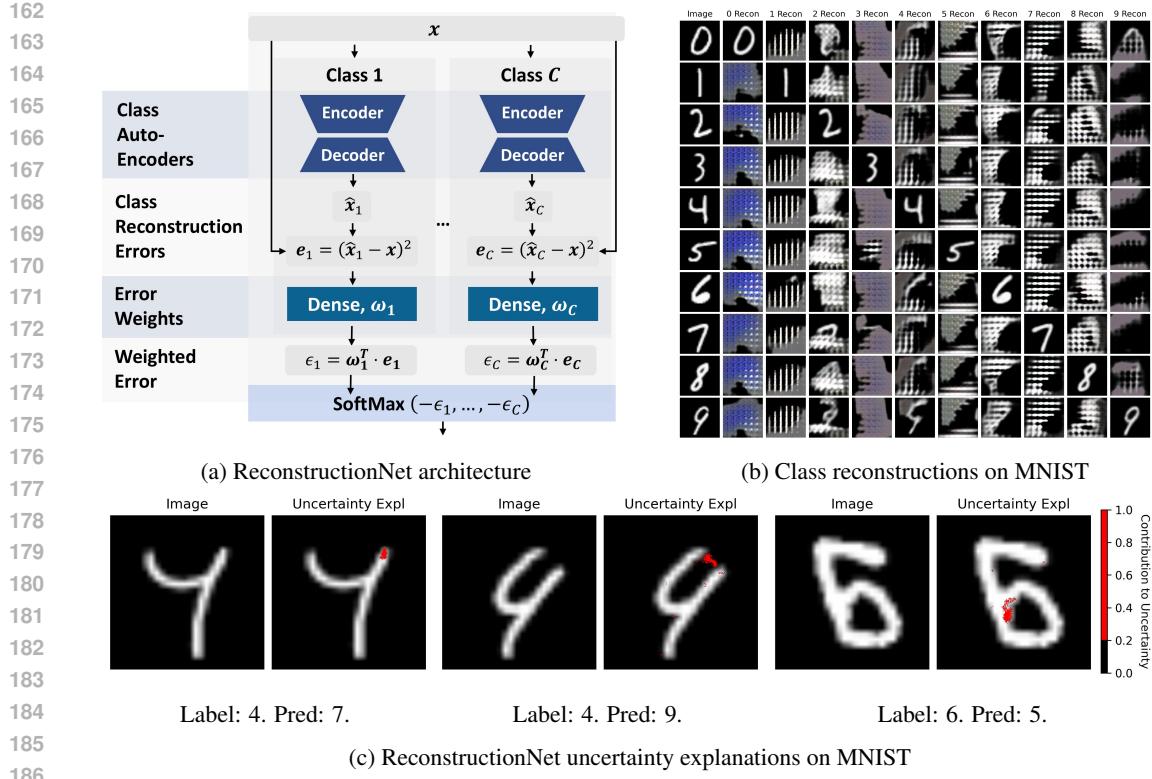


Figure 1: ReconstructionNet overview and examples. (a) ReconstructionNet architecture: Each class-specific autoencoder is trained to reconstruct only its own class, producing reconstruction errors inversely related to the joint input–output probability. After applying error weights and softmax normalization, the resulting probabilities are used for classification (Equation 2). (b) MNIST class reconstructions: Only the autoencoder for the true class yields a faithful reconstruction, leading to lower reconstruction error and a higher prediction probability. Other autoencoders generate artifacts (e.g., checkerboard patterns (Odena et al., 2016)) due to lack of training on mismatched classes. (c) MNIST uncertainty explanations: Weighted class reconstruction errors highlight uncertain regions. In the first example, the extra right vertical line increases uncertainty of it being a 7; in the middle, the missing connector raises uncertainty of it being a 9; in the last, the extra bottom-left vertical line increases uncertainty of it being a 5.

Next, it calculates a weighted reconstruction error ϵ_{ij} for each class j , where the weights ω_j are trainable parameters (Equation 5).

$$\epsilon_{ij} = \omega_j^T \cdot \mathbf{e}_{ij}. \quad (5)$$

Finally, prediction probability \hat{p}_{ij} is obtained by applying softmax to the negative weighted errors ϵ_i . Negation is used since the higher the reconstruction error, the lower the probability (Figure 1b).

$$\hat{p}_i = \text{softmax}(-\epsilon_i). \quad (6)$$

To train the ensemble of class autoencoders concurrently, we designed a loss function (Equation 7) consisting of two components, with β as a hyperparameter to balance both training objectives.

$$\mathcal{L}_{total} = \mathcal{L}_{CE} + \beta \cdot \mathcal{L}_{Class_MSE}, \quad \mathcal{L}_{CE} = - \sum_{i=1}^N \sum_{j=1}^C p_{ij} \log(\hat{p}_{ij}) \quad (7)$$

\mathcal{L}_{CE} is the cross-entropy loss; it optimizes the classification performance of the ReconstructionNet. To ensure that the class autoencoders also learn to model the joint probability $\Pr(\hat{y}_i = j, \mathbf{x}_i)$ while also maintaining predictive accuracy, we introduce the class-dependent mean squared error (MSE):

$$\mathcal{L}_{Class_MSE}^j = \frac{1}{N_j} \sum_{\mathbf{x}_i \in \mathcal{X}_j} \|\hat{\mathbf{x}}_{ij} - \mathbf{x}_i\|^2. \quad (8)$$

Where each class autoencoder g_j is trained to minimize reconstruction error for its corresponding instances in \mathcal{X}_j . The overall reconstruction loss, \mathcal{L}_{Class_MSE} , is then the average across all classes:

$$\mathcal{L}_{Class_MSE} = \frac{1}{N} \sum_{j=1}^C N_j \cdot \mathcal{L}_{Class_MSE}^j. \quad (9)$$

The formulation of \mathcal{L}_{Class_MSE} ensures each class autoencoder is trained exclusively to reconstruct samples from its ground truth class, thereby modeling the joint distribution of the input and target. Assume that the latent vector z_j from the encoder of class j follows an isotropic Gaussian distribution, $\Pr(\mathbf{x}_i, y_i = j | z_{ij}) = \mathcal{N}(\hat{\mathbf{x}}, \sigma^2 I)$ (Doersch, 2021; Odaibo, 2019). The probability density function of the distribution of all \mathbf{x} from class j can be expressed as:

$$\Pr(\mathbf{x}_i, y_i = j | z_{ij}) = \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{1}{2\sigma^2} \|\mathbf{x}_i - \hat{\mathbf{x}}_{ij}\|^2\right). \quad (10)$$

To model this distribution, we optimize our model to maximize the log-likelihood over the training dataset with target class j :

$$\ell(\hat{\mathbf{x}}, \sigma) = \sum_{i \in \mathcal{X}_j} \left[-\frac{d}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \|\mathbf{x}_i - \hat{\mathbf{x}}_{ij}\|^2 \right] \quad (11)$$

$$= \sum_{i \in \mathcal{X}_j} \left[-\frac{d}{2} \log(2\pi\sigma^2) \right] - \frac{N_j}{2\sigma^2} \cdot \mathcal{L}_{Class_MSE}^j. \quad (12)$$

We observe that maximizing the log-likelihood is equivalent to minimizing $\mathcal{L}_{Class_MSE}^j$. This demonstrates that minimizing \mathcal{L}_{Class_MSE} leads to the training of class autoencoders that model the joint distribution of the input and target, with reconstruction error inversely related to the joint probability. Additionally, the architecture and loss function of ReconstructionNet limit its hypothesis space, making it more resistant to epistemic uncertainty (Hüllermeier & Waegeman, 2021).

3.2 UNCERTAINTY ESTIMATION

By modelling the joint input-output probability, we can quantify:

1. **Aleatoric Uncertainty:** When instances lie in overlapping class regions, several classes have similarly *high* probabilities above threshold θ_1 , signalling high aleatoric uncertainty.

$$\exists \mathbf{C}_i \subseteq \{1, \dots, C\}, |\mathbf{C}_i| \geq 2 \text{ s.t.} \\ \Pr(\hat{y}_i = c_1, \mathbf{x}_i) \approx \Pr(\hat{y}_i = c_2, \mathbf{x}_i) \geq \theta_1, \forall c_1, c_2 \in \mathbf{C}_i. \quad (13)$$

2. **Distributional Uncertainty:** When instances lie beyond the training distribution, the joint probabilities of all classes are similarly *low*, below some threshold θ_2 , signaling high distributional uncertainty.

$$\forall c \in \{1, \dots, C\}, \Pr(\hat{y}_i = c, \mathbf{x}_i) \leq \theta_2. \quad (14)$$

Modelling joint probabilities allows differentiation between uncertainty types: instances show high aleatoric uncertainty when their most probable classes have probabilities above θ_1 , and high distributional uncertainty when all class probabilities are below θ_2 .

The notion of aleatoric uncertainty, as illustrated in Equation 13, is nicely captured by Shannon entropy, reflecting evenly spread high prediction probabilities across overlapping classes:

Definition 3 (Aleatoric Uncertainty) *For an instance \mathbf{x}_i , aleatoric uncertainty is quantified using the Shannon entropy (Shannon, 1948) of the prediction probabilities \hat{p}_{ij} :*

$$\sigma_{aleatoric}(\mathbf{x}_i) = - \sum_{j=1}^C \hat{p}_{ij} \log \hat{p}_{ij}.$$

Distributional uncertainty in Equation 14 is captured by the predicted class's reconstruction error. By design, the predicted class has the lowest error and highest probability among all classes; therefore, if its error is high (probability low), all other classes also have low probabilities.

270 **Definition 4 (Distributional Uncertainty)** For an instance \mathbf{x}_i , distributional uncertainty is the re-
 271 construction error of the predicted class \hat{y}_i 's autoencoder:
 272

$$\sigma_{dist}(\mathbf{x}_i) = \|\mathbf{e}_{i\hat{y}_i}\|_1.$$

274 3.3 UNCERTAINTY EXPLANATION
 275

276 The weighted reconstruction errors ζ of the predicted class \hat{y} serve as uncertainty explanations, as
 277 they represent feature uncertainties scaled by their importance to the prediction.
 278

279 **Definition 5 (ReconstructionNet Explanation)** The ReconstructionNet Explanation for instance
 280 \mathbf{x}_i and its predicted class \hat{y}_i is the weighted reconstruction errors of predicted class \hat{y}_i :

$$\zeta(\mathbf{x}_i) = \omega_{\hat{y}_i} \odot \mathbf{e}_{i\hat{y}_i}.$$

283 ReconstructionNet uncertainty explanations exhibit the following three properties: (1) Implementation
 284 Invariance (Sundararajan et al., 2017), (2) Sensitivity (Sundararajan et al., 2017), and (3)
 285 Consistency (Lundberg & Lee, 2017). An explanation that satisfies all three properties is (1) consistent
 286 across different implementations, (2) does not attribute irrelevant features incorrectly, and (3)
 287 preserves the relative importance of features across models.
 288

An uncertainty explanation is *implementation-invariant* if, for a pair of functionally equivalent prediction models, the same explanation is generated for any instance. A pair of functionally equivalent models are models that yield the same output for a given set of inputs.

Property 1 (Implementation Invariance) Given two functionally equivalent prediction models, f and f' , an explanation function χ is implementation-invariant if and only if, for any instance \mathbf{x} , the explanations derived from both models using χ are equivalent: $\chi(\mathbf{x}; f) = \chi(\mathbf{x}; f')$.

ReconstructionNet's uncertainty explanations ζ , defined as the weighted reconstruction errors of the predicted class (Definition 5), are implementation-invariant: for two functionally equivalent models, both the error weights ω and the reconstruction errors remain identical.

A *sensitive* explanation function allocates zero attribution to irrelevant features for prediction. A feature is considered irrelevant if a change in its value does not impact the model's prediction.

Property 2 (Sensitivity) An explanation function χ is sensitive if it assigns a zero feature attribution value, $\chi(\mathbf{x}; f)_i = 0$, to features i that are irrelevant to the prediction.

ReconstructionNet's uncertainty explanations ζ are sensitive. Since the error weight ω encodes each feature's contribution to the final prediction, any change in feature uncertainty that does not affect the prediction must have a weight of zero, yielding a weighted reconstruction error ζ of zero.

An explanation is *consistent* if a feature's uncertainty attribution does not decrease when the model is altered to increase that feature's contribution.

Property 3 (Consistency) Let f' be a modification of f where feature i 's contribution is increased. For an instance \mathbf{x} and $\mathbf{x}^{\setminus i}$ with $x_i = 0$, an explanation χ is consistent if:

$$f'(\mathbf{x}) - f'(\mathbf{x}^{\setminus i}) \geq f(\mathbf{x}) - f(\mathbf{x}^{\setminus i}) \quad \text{then} \quad \chi(\mathbf{x}; f')_i \geq \chi(\mathbf{x}; f)_i.$$

ReconstructionNet's uncertainty explanations, ζ , are consistent. For any instance, reconstruction errors stay the same between f and f' , since the joint input-output probability modeled by the autoencoders is independent of feature contributions. Thus, only the error weights ω can change; if a feature's weight increases in f' , its uncertainty attribution also increases, satisfying consistency.

317 4 EXPERIMENTS
 318

320 4.1 DATASETS
 321

322 We use tabular datasets (Covid, Diabetes, Fund) to evaluate uncertainty reliability, selectivity, and
 323 robustness, medical image datasets (ISIC, OCTMNIST) for OOD detection, and MNIST and ISIC
 to assess uncertainty explanation correctness. The datasets are summarized as follows:

- 324 1. **Covid** (Hinns et al., 2021) is a tabular dataset of United Kingdom’s COVID-19 policies
 325 and regional case counts, labelled by whether R_t (effective reproduction number) > 1 .
 326 2. **Diabetes** (Mustafa, 2023) is a binary tabular dataset of demographics, pre-existing conditions, and vital signs, with labels indicating diabetes status.
 327 3. **Fund** (Kovvuri et al., 2023) is a binary tabular dataset from 4,330 funds, using macroeconomic indicators, fund allocations, HHI, and past performance to predict if a fund’s net asset value (NAV) exceeds the previous quarter’s.
 328 4. **ISIC** (Codella et al., 2019; Tschandl et al., 2018) is a dermoscopic image dataset of seven skin conditions. For OOD detection, we created three datasets with decreasing similarity to ISIC: BCN-IN (images from seen classes of BCN20000 (Combalia et al., 2019)), BCN-OUT (images from the unseen Scar class), and ChestMNIST (Wang et al., 2017).
 329 5. **OCTMNIST** (Kermany et al., 2018) is a retinal OCT dataset with four classes. For OOD
 330 detection, we used three datasets of decreasing similarity to OCTMNIST: OCTDL-IN (im-
 331 ages from seen classes of OCTDL (Kulyabin et al., 2024)), OCTDL-OUT (images from
 332 unseen classes), and ChestMNIST (Wang et al., 2017).
 333 6. **MNIST** (Deng, 2012) is an image dataset of ten handwritten digits.

341 4.2 BASELINES
 342

343 We compared ReconstructionNet against six recent state-of-the-art uncertainty estimation methods:
 344 (1) Entropy, (2) MCD, (3) DE, (4) PN, (5) BNN, and (6) EDL. Implementation and tuning details of
 345 the baselines are provided in the appendix.

346 4.3 EVALUATION METRICS
 347

348 We evaluate prediction performance with Area Under receiver operating characteristic Curve (AUC)
 349 and accuracy. To evaluate uncertainty estimation performance, we used the following metrics:

- 351 1. **Correlation** (Mi et al., 2022; Upadhyay et al., 2022) measures reliability as the Pearson
 352 correlation between uncertainty and error, with higher values indicating better reliability.
 353 2. **AURC** (Ding et al., 2020) quantifies selectivity by measuring the area under the risk-
 354 coverage curve (AURC), with lower values indicating better selectivity.
 355 3. **σ -Risk Score** measures robustness to false negatives as errors for instances with normalized
 356 uncertainty below $\sigma = \{0.1, 0.2, 0.3, 0.4\}$, with lower values indicating greater resilience.
 357 4. **OOD Detection** (Lakshminarayanan et al., 2017; Postels et al., 2020; Malinin & Gales,
 358 2018) measures how well uncertainty distinguishes in-distribution from OOD instances
 359 using AUROC, with higher values indicating better detection.

360 4.4 RESULTS
 361

362 4.4.1 PREDICTION PERFORMANCE
 363

364 Table 1: Model prediction performance. NN refers to an MLP for tabular and a ResNet for image
 365 datasets. The best-performing model for each metric is **bolded**, while the second-best is underlined.

	Covid (Tabular)		Diabetes (Tabular)		Fund (Tabular)		ISIC (Image)		OCTMNIST (Image)	
	AUC (\uparrow)	Acc (\uparrow)	AUC (\uparrow)	Acc (\uparrow)	AUC (\uparrow)	Acc (\uparrow)	AUC (\uparrow)	Acc (\uparrow)	AUC (\uparrow)	Acc (\uparrow)
RN (Ours)	0.95 ± 0.00	0.88 ± 0.00	0.97 ± 0.00	0.92 ± 0.01	0.75 ± 0.01	0.71 ± 0.00	0.91 ± 0.01	0.76 ± 0.01	0.99 ± 0.00	0.91 ± 0.02
NN	0.95 ± 0.00	0.89 ± 0.00	0.98 ± 0.00	0.93 ± 0.01	0.60 ± 0.02	0.71 ± 0.00	0.87 ± 0.01	0.70 ± 0.01	0.99 ± 0.01	0.88 ± 0.03
MCD	0.95 ± 0.00	0.89 ± 0.00	0.98 ± 0.00	0.93 ± 0.01	0.59 ± 0.02	0.71 ± 0.00	0.87 ± 0.01	0.70 ± 0.01	0.99 ± 0.01	0.88 ± 0.03
DE	0.95 ± 0.00	0.89 ± 0.00	0.98 ± 0.00	0.93 ± 0.00	0.59 ± 0.01	0.71 ± 0.00	0.89 ± 0.00	<u>0.72 ± 0.01</u>	0.99 ± 0.00	0.89 ± 0.02
PN	0.96 ± 0.01	0.88 ± 0.01	0.97 ± 0.00	0.92 ± 0.01	0.53 ± 0.00	0.71 ± 0.00	0.70 ± 0.01	0.64 ± 0.01	0.94 ± 0.02	0.80 ± 0.05
BNN	0.91 ± 0.02	0.85 ± 0.02	<u>0.97 ± 0.01</u>	0.91 ± 0.01	0.54 ± 0.01	0.71 ± 0.00	0.60 ± 0.02	0.52 ± 0.02	0.99 ± 0.00	0.87 ± 0.02
EDL	0.92 ± 0.02	0.79 ± 0.18	0.87 ± 0.07	0.95 ± 0.02	0.58 ± 0.05	<u>0.62 ± 0.17</u>	0.53 ± 0.01	0.51 ± 0.01	0.96 ± 0.00	0.83 ± 0.02

366 Table 1 compares model performance. ReconstructionNet (RN) achieved strong results across
 367 datasets, with second-highest AUC and accuracy on COVID and Diabetes, and the highest AUC

378 and accuracy on Fund, ISIC, and OCTMNIST. Notably, on Fund, ReconstructionNet outperformed
 379 others in AUC despite similar accuracy, suggesting it learns discriminative features robust to data
 380 shifts (caused by the COVID-19 pandemic) and is resilient to epistemic uncertainty.
 381

382 4.4.2 UNCERTAINTY ESTIMATION PERFORMANCE

384 Table 2: Uncertainty estimation performance. The best-performing model for each metric is **bolded**,
 385 while the second-best model is underlined.

Data	UE	Correlation (\uparrow)	AURC (\downarrow)	$\sigma = 0.1$ (\downarrow)	$\sigma = 0.2$ (\downarrow)	$\sigma = 0.3$ (\downarrow)	$\sigma = 0.4$ (\downarrow)
Covid	RN (Ours)	0.823 ± 0.016	0.028 ± 0.002	0.005 ± 0.001	0.006 ± 0.002	0.007 ± 0.002	0.012 ± 0.005
	MLP Entropy	0.664 ± 0.012	0.030 ± 0.002	0.003 ± 0.004	<u>0.016 ± 0.006</u>	<u>0.022 ± 0.006</u>	<u>0.032 ± 0.005</u>
	MLP MCD	0.639 ± 0.012	0.032 ± 0.003	0.004 ± 0.005	0.018 ± 0.009	0.029 ± 0.006	0.042 ± 0.005
	MLP DE	0.603 ± 0.031	0.030 ± 0.003	0.011 ± 0.003	0.020 ± 0.007	0.032 ± 0.007	0.040 ± 0.005
	PN Epis	0.313 ± 0.082	0.038 ± 0.011	0.021 ± 0.010	0.034 ± 0.015	0.041 ± 0.017	0.051 ± 0.016
	PN Alea	<u>0.690 ± 0.034</u>	<u>0.029 ± 0.007</u>	0.016 ± 0.005	0.022 ± 0.007	0.032 ± 0.006	0.039 ± 0.005
	BNN	0.030 ± 0.027	0.132 ± 0.018	0.104 ± 0.070	0.159 ± 0.023	0.161 ± 0.023	0.132 ± 0.015
Diabetes	EDL	0.674 ± 0.068	0.115 ± 0.094	0.238 ± 0.105	0.073 ± 0.009	0.058 ± 0.004	0.045 ± 0.005
	RN (Ours)	<u>0.902 ± 0.011</u>	<u>0.008 ± 0.001</u>	0.000 ± 0.000	<u>0.001 ± 0.000</u>	0.001 ± 0.000	0.001 ± 0.000
	MLP Entropy	0.864 ± 0.009	0.007 ± 0.002	0.000 ± 0.000	0.000 ± 0.000	0.001 ± 0.000	0.001 ± 0.000
	MLP MCD	0.845 ± 0.022	<u>0.008 ± 0.002</u>	0.000 ± 0.000	0.000 ± 0.000	0.001 ± 0.000	<u>0.002 ± 0.001</u>
	MLP DE	0.733 ± 0.031	<u>0.011 ± 0.001</u>	0.000 ± 0.000	<u>0.001 ± 0.000</u>	<u>0.005 ± 0.001</u>	<u>0.011 ± 0.003</u>
	PN Epis	0.525 ± 0.297	0.020 ± 0.021	0.006 ± 0.009	0.016 ± 0.023	0.023 ± 0.030	0.028 ± 0.033
	PN Alea	0.812 ± 0.022	0.009 ± 0.001	<u>0.001 ± 0.000</u>	<u>0.001 ± 0.000</u>	<u>0.002 ± 0.001</u>	<u>0.003 ± 0.001</u>
Fund	BNN	-0.619 ± 0.012	0.214 ± 0.016	0.262 ± 0.024	0.222 ± 0.016	0.185 ± 0.016	0.136 ± 0.013
	EDL	0.942 ± 0.017	0.046 ± 0.034	0.000 ± 0.000	<u>0.001 ± 0.002</u>	<u>0.006 ± 0.006</u>	<u>0.011 ± 0.010</u>
	RN (Ours)	0.385 ± 0.008	0.141 ± 0.003	0.039 ± 0.004	0.048 ± 0.005	0.077 ± 0.015	0.126 ± 0.023
	MLP Entropy	0.131 ± 0.034	<u>0.212 ± 0.023</u>	0.064 ± 0.043	0.132 ± 0.063	0.187 ± 0.036	0.241 ± 0.012
	MLP MCD	0.128 ± 0.038	0.236 ± 0.018	0.073 ± 0.067	0.124 ± 0.062	0.165 ± 0.046	0.210 ± 0.028
	MLP DE	0.022 ± 0.045	0.287 ± 0.032	0.364 ± 0.331	0.330 ± 0.192	0.283 ± 0.031	0.286 ± 0.045
	PN Epis	0.016 ± 0.001	0.291 ± 0.017	0.279 ± 0.028	0.278 ± 0.025	0.278 ± 0.022	0.279 ± 0.021

407 **Aleatoric Uncertainty:** Table 2 summarizes uncertainty estimation performance. Reconstruction-
 408 Net outperformed all models on COVID except σ -risk at $\sigma = 0.1$, where it matched MLP Entropy
 409 and MCD. On Diabetes and Fund, it consistently ranked among the top across metrics.
 410

411 Table 3: Out-of-distribution (OOD) detection performance, measured using AUROC. OOD datasets
 412 are presented in order of increasing deviation from the in-distribution. The best-performing model
 413 for each metric is **bolded**, while the second-best model is underlined.

	ISIC			OCTMNIST		
	BCN-IN (\uparrow)	BCN-OUT (\uparrow)	ChestMNIST (\uparrow)	OCTDL-IN (\uparrow)	OCTDL-OUT (\uparrow)	ChestMNIST (\uparrow)
RN (Ours)	0.777 ± 0.086	0.846 ± 0.070	0.919 ± 0.097	0.783 ± 0.082	0.866 ± 0.049	1.000 ± 0.000
ResNet Entropy	0.742 ± 0.010	0.757 ± 0.027	0.664 ± 0.072	0.674 ± 0.031	0.386 ± 0.066	0.826 ± 0.060
ResNet MCD	<u>0.746 ± 0.004</u>	<u>0.770 ± 0.016</u>	0.678 ± 0.080	0.661 ± 0.034	0.392 ± 0.064	0.844 ± 0.060
ResNet DE	0.720 ± 0.020	0.703 ± 0.029	<u>0.728 ± 0.055</u>	0.739 ± 0.009	0.497 ± 0.029	0.900 ± 0.028
PN Epis	0.674 ± 0.021	0.686 ± 0.039	0.654 ± 0.050	0.565 ± 0.052	0.483 ± 0.087	0.470 ± 0.145
PN Alea	0.660 ± 0.026	0.669 ± 0.041	0.631 ± 0.065	0.625 ± 0.063	0.492 ± 0.025	0.500 ± 0.107
BNN	0.556 ± 0.011	0.540 ± 0.056	0.547 ± 0.059	<u>0.742 ± 0.014</u>	<u>0.734 ± 0.016</u>	0.896 ± 0.015
EDL	0.533 ± 0.011	0.539 ± 0.023	0.619 ± 0.024	0.633 ± 0.012	0.669 ± 0.011	0.760 ± 0.013

424 **Distributional Uncertainty:** Table 3 presents the OOD detection performance of all uncertainty
 425 estimates. ReconstructionNet achieves the highest AUROC across all OOD datasets and improves
 426 as datasets deviate further from in-distribution data, showing it effectively ranks dataset dissimilarity.
 427 In contrast, other estimates like Entropy, MCD, PN, and BNN drop on highly dissimilar datasets (see
 428 ISIC) or struggle with unseen classes, as seen on OCTMNIST.

429 **Distinguishing Between Aleatoric and Distributional:** To illustrate ReconstructionNet’s ability
 430 to separate uncertainty types, we visualize aleatoric and distributional uncertainty over time on the
 431 Fund dataset (Figure 2). Aleatoric uncertainty is highest before January 2020, while during the 2020

recession (per the National Bureau of Economic Research), distributional uncertainty predominates, demonstrating ReconstructionNet’s discriminative capability.

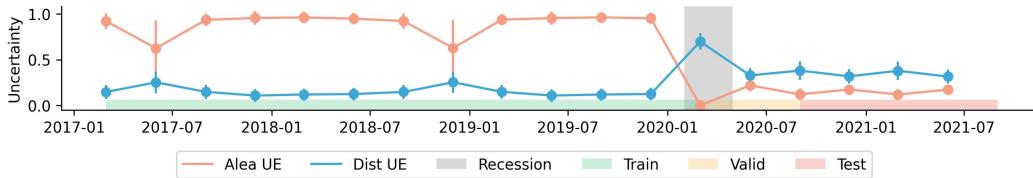


Figure 2: Aleatoric (Alea) and distributional (Dist) uncertainty over time on the Fund dataset. Uncertainty values are min–max normalized, and error bars represent one standard deviation. In early 2020, distributional uncertainty overtook aleatoric uncertainty due to COVID-19–induced shifts, while aleatoric uncertainty dropped as increasing class imbalance reduced label ambiguity.

4.4.3 UNCERTAINTY EXPLANATION PERFORMANCE

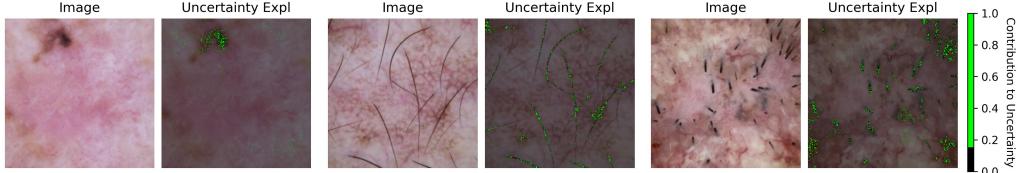


Figure 3: Uncertainty explanation illustration using images from the ISIC dataset. Positive attributions were min–max normalized, thresholded (pixel uncertainty > 0.15 shown in green), and overlaid for clarity. The highlighted regions “explain” the prediction uncertainty. AKIEC refers to Actinic Keratosis, BCC to Basal Cell Carcinoma, and BKL to Benign Keratosis.

To highlight the practical value of our uncertainty explanations in identifying regions that contribute to misclassification, we qualitatively assess them on the ISIC dataset. In Figure 3a, the image was misclassified as BCC instead of AKIEC, with uncertainty concentrated on the mole at the top-left corner, likely because such dark, mole-like spots resemble features more typical of BCC than AKIEC (Lee, 2017). In Figures 3b and 3c, both images were misclassified as BKL, with uncertainty concentrated on the hairs covering a substantial portion of the lesion, as such occlusions can obscure diagnostic features. These visual explanations align with human intuition, demonstrating their effectiveness in pinpointing input features that confuse the model.

5 CONCLUSION

This paper proposed ReconstructionNet, a neural network for reliable uncertainty estimation alongside classification. Unlike models based on conditional probability, ReconstructionNet uses class-specific autoencoders to model the input–output joint distribution, predicting the class with maximal joint probability (or minimal reconstruction error). This approach quantifies aleatoric and distributional uncertainty while minimizing epistemic uncertainty in a single training session. Across five real-world datasets, ReconstructionNet achieved comparable or improved classification performance, with uncertainty estimates showing superior reliability, selectivity, robustness to false negatives, and strong OOD detection. Its inbuilt, cost-free explanations highlight input features contributing to uncertainty on MNIST and ISIC, with theoretical properties of (1) Implementation Invariance, (2) Sensitivity, and (3) Consistency. While ReconstructionNet performs best on large, balanced datasets, this limitation suggests future directions, such as explicitly quantifying epistemic uncertainty.

486 6 REPRODUCIBILITY STATEMENT

488 For implementation details, see the code repository: <https://anonymous.4open.489 science/r/ReconstructionNet-4F8C/>. For tabular datasets, input features were normalized using min–max scaling prior to training. For image datasets, pixel intensities were scaled from 490 [0, 255] to [0, 1].

493 REFERENCES

- 495 Alexander Amini, Wilko Schwarting, Ava Soleimany, and Daniela Rus. Deep evidential 496 regression. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina 497 Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 498 2020, December 6-12, 2020, virtual*, volume 33, pp. 14927–14937, virtual, 2020. Curran 499 Associates, Inc. URL <https://proceedings.neurips.cc/paper/2020/hash/aab085461de182608ee9f607f3f7d18f-Abstract.html>.
- 502 Javier Antorán, Umang Bhatt, Tameem Adel, Adrian Weller, and José Miguel Hernández-Lobato. 503 Getting a CLUE: A method for explaining uncertainty estimates. In *9th International Conference 504 on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, Austria, 2021. 505 OpenReview.net. URL <https://openreview.net/forum?id=XSLF1XFq5h>.
- 507 Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. 508 Signature Verification using a "Siamese" Time Delay Neural Network. In *Advances in Neural 509 Information Processing Systems*, volume 6, USA, 1993. Morgan-Kaufmann. URL https://proceedings.neurips.cc/paper_files/paper/1993/hash/288cc0ff022877bd3df94bc9360b9c5d-Abstract.html.
- 512 Bertrand Charpentier, Daniel Zügner, and Stephan Günnemann. Posterior Network: Uncertainty 513 Estimation without OOD Samples via Density-Based Pseudo-Counts. In *Advances in Neural 514 Information Processing Systems*, volume 33, pp. 1356–1367, Online, 2020. Curran Associates, 515 Inc.
- 516 Bertrand Charpentier, Chenxiang Zhang, and Stephan Günnemann. Training, architecture, and prior 517 for deterministic uncertainty methods. *CoRR*, abs/2303.05796, 2023. doi: 10.48550/ARXIV. 518 2303.05796. URL <https://doi.org/10.48550/arXiv.2303.05796>.
- 520 Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic 521 minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.
- 523 Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with 524 application to face verification. In *2005 IEEE computer society conference on computer vision 525 and pattern recognition (CVPR'05)*, volume 1, pp. 539–546, USA, 2005. IEEE.
- 526 Noel Codella, Veronica Rotemberg, Philipp Tschandl, M. Emre Celebi, Stephen Dusza, David Gut- 527 man, Brian Helba, Aadi Kalloo, Konstantinos Liopyris, Michael Marchetti, Harald Kittler, and 528 Allan Halpern. Skin Lesion Analysis Toward Melanoma Detection 2018: A Challenge Hosted by 529 the International Skin Imaging Collaboration (ISIC), March 2019.
- 531 Marc Combaliá, Noel CF Codella, Veronica Rotemberg, Brian Helba, Veronica Vilaplana, Ofer 532 Reiter, Cristina Carrera, Alicia Barreiro, Allan C Halpern, Susana Puig, et al. Bcn20000: Dermo- 533 scopic lesions in the wild. *arXiv preprint arXiv:1908.02288*, 2019.
- 534 Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE 535 Signal Processing Magazine*, 29(6):141–142, 2012.
- 537 Yukun Ding, Jinglan Liu, Jinjun Xiong, and Yiyu Shi. Revisiting the Evaluation of Uncertainty 538 Estimation and Its Application to Explore Model Complexity-Uncertainty Trade-Off. In *2020 539 IEEE CVPRW*, pp. 22–31, Seattle, WA, USA, June 2020. IEEE. ISBN 978-1-72819-360-1. doi: 10.1109/CVPRW50498.2020.00010.

- 540 Carl Doersch. Tutorial on Variational Autoencoders, 2021. URL <http://arxiv.org/abs/1606.05908>.
- 541
- 542
- 543 Xiuyi Fan. Position paper: Integrating explainability and uncertainty estimation in medical AI. In
544 *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN)*, 2025.
- 545
- 546 Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian Approximation: Representing Model
547 Uncertainty in Deep Learning. In *Proceedings of The 33rd ICML*, pp. 1050–1059, USA, June
548 2016. PMLR.
- 549
- 550 Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recogni-
551 tion, 2015. URL <https://arxiv.org/abs/1512.03385>.
- 552
- 553 James Hinns, Xiuyi Fan, Siyuan Liu, Veera Raghava Kovvuri, Mehmet Orcun Yalcin, and
554 Markus Roggenbach. An initial study of machine learning underspecification using feature at-
555tribution explainable ai algorithms: A covid-19 virus transmission case study. In *PRICAI 2021:*
556 *Trends in Artificial Intelligence: 18th Pacific Rim International Conference on Artificial Intelli-
557 gence, PRICAI 2021, Hanoi, Vietnam, November 8–12, 2021, Proceedings, Part I 18*, pp. 323–
558 335, Vietnam, 2021. Springer.
- 559
- 560 Eyke Hüllermeier and Willem Waegeman. Aleatoric and epistemic uncertainty in machine learning:
561 An introduction to concepts and methods. *Machine learning*, 110(3):457–506, 2021.
- 562
- 563 Pascal Iversen, Simon Witzke, Katharina Baum, and Bernhard Y. Renard. Identifying Drivers of
564 Predictive Aleatoric Uncertainty, 2024. URL <http://arxiv.org/abs/2312.07252>.
- 565
- 566 Laurent Valentin Jospin, Hamid Laga, Farid Boussaid, Wray Buntine, and Mohammed Bennamoun.
567 Hands-On Bayesian Neural Networks—A Tutorial for Deep Learning Users. *IEEE Computa-
568 tional Intelligence Magazine*, 17(2):29–48, May 2022. ISSN 1556-6048. doi: 10.1109/MCI.
569 2022.3155327.
- 570
- 571 Daniel S Kermany, Michael Goldbaum, Wenjia Cai, Carolina CS Valentim, Huiying Liang, Sally L
572 Baxter, Alex McKeown, Ge Yang, Xiaokang Wu, Fangbing Yan, et al. Identifying medical diag-
573 noses and treatable diseases by image-based deep learning. *cell*, 172(5):1122–1131, 2018.
- 574
- 575 Lennard Korte, Li Rong Wang, and Xiuyi Fan. Confidence estimation in analyzing intravascular
576 optical coherence tomography images with deep neural networks. In *2024 IEEE Conference on
577 Artificial Intelligence (CAI)*, pp. 358–364, Singapore, 2024. IEEE.
- 578
- 579 Veera Raghava Reddy Kovvuri, Hsuan Fu, Xiuyi Fan, and Monika Seisenberger. Fund performance
580 evaluation with explainable artificial intelligence. *Finance Research Letters*, 58:104419, 2023.
- 581
- 582 Mikhail Kulyabin, Aleksei Zhdanov, Anastasia Nikiforova, Andrey Stepichev, Anna Kuznetsova,
583 Mikhail Ronkin, Vasilii Borisov, Alexander Bogachev, Sergey Korotkikh, Paul A Constable, et al.
584 Octdl: Optical coherence tomography dataset for image-based deep learning methods. *Scientific
data*, 11(1):365, 2024.
- 585
- 586 Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and Scalable Predictive
587 Uncertainty Estimation using Deep Ensembles. In *Advances in Neural Information Processing
588 Systems*, volume 30, pp. 6402–6413, USA, 2017. Curran Associates, Inc.
- 589
- 590 Peter K. Lee. *Actinic Keratosis, Basal, and Squamous Cell Carcinoma*, chapter 17. McGraw-Hill
591 Education, New York, NY, 2017. URL accessmedicine.mhmedical.com/content.aspx?aid=1177005687.
- 592
- 593 Scott Lundberg and Su-In Lee. A unified approach to interpreting model predictions, 2017. URL
594 <https://arxiv.org/abs/1705.07874>.
- 595
- Andrey Malinin and Mark Gales. Predictive Uncertainty Estimation via Prior Networks. In *Advances in Neural Information Processing Systems*, volume 31. Curran Asso-
596 ciates, Inc., 2018. URL <https://proceedings.neurips.cc/paper/2018/hash/3ea2db50e62ceefceaf70a9d9a56a6f4-Abstract.html>.

- 594 Lu Mi, Hao Wang, Yonglong Tian, Hao He, and Nir N. Shavit. Training-Free Uncertainty Estimation
 595 for Dense Regression: Sensitivity as a Surrogate. *Proceedings of AAAI*, 36(9):10042–10050,
 596 2022. ISSN 2374-3468. doi: 10.1609/aaai.v36i9.21243.
- 597 Carlos Mougan and Dan Saattrup Nielsen. Monitoring Model Deterioration with Explainable Uncer-
 598 tainty Estimation via Non-parametric Bootstrap. *Proceedings of the AAAI Conference on Artifi-*
599 cial Intelligence, 37(12):15037–15045, 2023. ISSN 2374-3468. doi: 10.1609/aaai.v37i12.26755.
 600 URL <https://ojs.aaai.org/index.php/AAAI/article/view/26755>.
- 601 Mohammed Mustafa. Diabetes prediction dataset, version 1. <https://www.kaggle.com/datasets/iammustafatz/diabetes-prediction-dataset/data>, 2023.
- 602 Stephen Odaibo. Tutorial: Deriving the Standard Variational Autoencoder (VAE) Loss Function,
 603 2019. URL <http://arxiv.org/abs/1907.08956>.
- 604 Augustus Odena, Vincent Dumoulin, and Chris Olah. Deconvolution and checkerboard artifacts.
 605 *Distill*, 1(10):e3, 2016.
- 606 Janis Postels, Hermann Blum, Cesar Cadena, Roland Siegwart, Luc Van Gool, and Federico
 607 Tombari. Quantifying aleatoric and epistemic uncertainty using density estimation in latent space.
 608 *CoRR*, abs/2012.03082, 2020. URL <https://arxiv.org/abs/2012.03082>.
- 609 Janis Postels, Mattia Segù, Tao Sun, Luca Daniel Sieber, Luc Van Gool, Fisher Yu, and Federico
 610 Tombari. On the practicality of deterministic epistemic uncertainty. In Kamalika Chaudhuri,
 611 Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato (eds.), *International*
 612 *Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, vol-
 613 *ume 162 of Proceedings of Machine Learning Research*, pp. 17870–17909, USA, 2022. PMLR.
 614 URL <https://proceedings.mlr.press/v162/postels22a.html>.
- 615 Murat Sensoy, Lance Kaplan, and Melih Kandemir. Evidential Deep Learning to Quantify Classifi-
 616 cation Uncertainty. In *Advances in Neural Information Processing Systems*, volume 31, Canada,
 617 2018. Curran Associates, Inc.
- 618 C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27
 619 (3):379–423, 1948. ISSN 0005-8580. doi: 10.1002/j.1538-7305.1948.tb01338.x. URL <https://ieeexplore.ieee.org/abstract/document/6773024>.
- 620 Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learn-
 621 ing. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob
 622 Fergus, S. V. N. Vishwanathan, and Roman Garnett (eds.), *Advances in Neural Informa-*
623 tion Processing Systems 30: Annual Conference on Neural Information Processing Systems
624 2017, December 4-9, 2017, Long Beach, CA, USA, pp. 4077–4087, USA, 2017. Curran
 625 Associates, Inc. URL <https://proceedings.neurips.cc/paper/2017/hash/cb8da6767461f2812ae4290eac7cbc42-Abstract.html>.
- 626 Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In
 627 *International conference on machine learning*, pp. 3319–3328, Australia, 2017. PMLR.
- 628 Sing Yee Toh, Chang Cai, Li Rong Wang, Xiaoyin Bai, Joanne Ngeow, and Xiuyi Fan. The effect
 629 of explainable ai and uncertainty quantification on medical students’ perspectives of decision-
 630 making ai: A cancer screening case study. In *Proceedings of the Extended Abstracts of the CHI*
 631 *Conference on Human Factors in Computing Systems, CHI EA ’25, New York, NY, USA, 2025*.
 632 Association for Computing Machinery. ISBN 9798400713958. doi: 10.1145/3706599.3719791.
 633 URL <https://doi.org/10.1145/3706599.3719791>.
- 634 Philipp Tschandl, Cliff Rosendahl, and Harald Kittler. The HAM10000 dataset, a large collection
 635 of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific Data*, 5(1):
 636 180161, August 2018. ISSN 2052-4463. doi: 10.1038/sdata.2018.161.
- 637 Dennis Ulmer, Christian Hardmeier, and Jes Frellsen. Prior and Posterior Networks: A Survey
 638 on Evidential Deep Learning Methods For Uncertainty Estimation. *Transactions on Machine*
639 Learning Research, 2023, 2023.

648 Uddeshya Upadhyay, Shyamgopal Karthik, Yanbei Chen, Massimiliano Mancini, and Zeynep
 649 Akata. BayesCap: Bayesian Identity Cap for Calibrated Uncertainty in Frozen Neural Networks.
 650 In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner
 651 (eds.), *Computer Vision – ECCV 2022*, pp. 299–317, Cham, 2022. Springer Nature Switzerland.
 652 ISBN 978-3-031-19775-8. doi: 10.1007/978-3-031-19775-8_18.

653
 654 Hanjing Wang, Bashirul Azam Biswas, and Qiang Ji. Optimization-Based Uncertainty Attribution
 655 Via Learning Informative Perturbations. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Rus-
 656 sakovsky, Torsten Sattler, and Gü̈l Varol (eds.), *Computer Vision – ECCV 2024*, volume 15136,
 657 pp. 237–253. Springer Nature Switzerland, Switzerland, 2025. ISBN 978-3-031-73228-7 978-3-
 658 031-73229-4. doi: 10.1007/978-3-031-73229-4_14. URL https://link.springer.com/10.1007/978-3-031-73229-4_14.

659
 660 Li Rong Wang, Thomas C. Henderson, and Xiuyi Fan. An Uncertainty Estimation Model for Al-
 661 gorithmic Trading Agent. In Soon-Geul Lee, Jinung An, Nak Young Chong, Marcus Strand,
 662 and Joo H. Kim (eds.), *Intelligent Autonomous Systems 18*, pp. 459–465, Korea, 2024. Springer
 663 Nature Switzerland. ISBN 978-3-031-44981-9. doi: 10.1007/978-3-031-44981-9_39.

664
 665 Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammad Bagheri, and Ronald M Sum-
 666 mers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised
 667 classification and localization of common thorax diseases. In *Proceedings of the IEEE conference*
 668 *on computer vision and pattern recognition*, pp. 2097–2106, 2017.

669
 670 Claudio Zelenka, Andrea Göhring, Daniyal Kazempour, Maximilian Hünemörder, Lars Schmarje,
 671 and Peer Kröger. A simple and explainable method for uncertainty estimation using attribute
 672 prototype networks. In *Proceedings of the IEEE/CVF International Conference on Computer*
 673 *Vision*, pp. 4570–4579, 2023a.

674
 675 Claudio Zelenka, Andrea Göhring, Daniyal Kazempour, Maximilian Hünemörder, Lars Schmarje,
 676 and Peer Kröger. A Simple and Explainable Method for Uncertainty Estimation using At-
 677 tribute Prototype Networks. In *2023 IEEE/CVF International Conference on Computer Vi-*
 678 *sion Workshops (ICCVW)*, pp. 4572–4581, France, 2023b. IEEE. ISBN 979-8-3503-0744-
 679 3. doi: 10.1109/ICCVW60793.2023.00491. URL [https://ieeexplore.ieee.org/](https://ieeexplore.ieee.org/document/10350670/)
 680 document/10350670/.

681 A APPENDIX

682 A.1 DATASET DETAILS

686 A.1.1 COVID-19 VIRUS TRANSMISSION DATASET

688 (Hinns et al., 2021) is a tabular binary classification dataset comprising 3,553 instances. Each in-
 689 stance is characterized by 32 continuous features describing the United Kingdom’s COVID-19 poli-
 690 cies and daily case counts across 12 regions, covering the period from February 2020 to February
 691 2021. The dataset is labelled with a binary class indicating whether $R_t > 1$, where R_t represents the
 692 effective reproduction number. A value of $R_t > 1$ signifies an increasing spread of COVID-19. We
 693 randomly divided the dataset into three subsets: training (70%), validation (10%), and test (20%).

694 A.1.2 DIABETES DIAGNOSIS DATASET

696 (Mustafa, 2023) is a tabular binary classification dataset comprising 100,000 instances, each re-
 697 presenting a patient. Each instance is described by eight features detailing the patient’s demographics
 698 (e.g., age and gender), pre-existing conditions (e.g., prevalence of heart disease), and vital signs
 699 (e.g., blood glucose level), and has a binary label indicating whether the patient suffers from dia-
 700 betes (0 indicating the patient is diabetes-free and 1 indicating the patient suffers from diabetes). We
 701 divided the dataset into three subsets for our experiments: training (80%), validation (10%), and test
 (10%).

702 A.1.3 FUND PERFORMANCE EVALUATION DATASET
 703

704 (Kovvuri et al., 2023) is a tabular binary classification dataset comprising 77,940 instances, designed
 705 to predict whether a fund’s net asset value (NAV) exceeds its NAV from the previous quarter. Each
 706 instance represents the state of one of 4,330 funds between March 2017 and June 2021, sampled
 707 quarterly. Each instance is characterized by 18 continuous features, including macroeconomic in-
 708 dicators (such as stock market returns (ST), exchange rate returns (EXR), and interest rates (IR)),
 709 country-level equity investment percentages and net asset data for the fund (covering 11 countries
 710 and “Other Country”), the Herfindahl–Hirschman index (HHI), and a past performance metric com-
 711 puted as the sum of class labels from the past four quarters (“L4f Gain”). We split the dataset into
 712 three subsets based on date: instances before March 2020 were assigned to the training set (66.7%),
 713 instances between March 2020 and June 2020 to the validation set (11.1%), and instances after June
 714 2020 to the test set (22.2%).
 715

716 A.1.4 ISIC
 717

718 (Codella et al., 2019; Tschandl et al., 2018) is a dermoscopic image dataset containing instances from
 719 seven skin conditions: melanoma (MEL), melanocytic nevus (NV), basal cell carcinoma (BCC), ac-
 720 tinic keratosis (AKIEC), benign keratosis (BKL), dermatofibroma (DF), and vascular lesion (VASC).
 721 We followed the 2018 ISIC Challenge split, with 10,015 training, 193 validation, and 1,512 test im-
 722 ages. For the Out-of-Distribution (OOD) detection experiments, we constructed three OOD datasets
 723 with decreasing similarity to ISIC: (1) BCN-IN: 1,512 dermoscopic images of seen classes from the
 724 BCN20000 dataset (Combalia et al., 2019), (2) BCN-OUT: 313 images from the unseen class Scar,
 725 and (3) ChestMNIST: 1,512 images from the ChestMNIST dataset (Wang et al., 2017).
 726

727 A.1.5 OCTMNIST
 728

729 (Kermany et al., 2018) is a retinal OCT image dataset comprising four classes: 47% Normal, 34%
 730 CNV (Choroidal Neovascularization), 11% DME (Diabetic Macular Edema), and 8% Drusen. We
 731 split the dataset into three subsets: 97,477 training, 10,832 validation, and 1,000 test instances. For
 732 the OOD detection experiments, we constructed three OOD datasets with decreasing similarity to
 733 OCTMNIST: (1) **OCTDL-IN**: 618 OCTDL (Kulyabin et al., 2024) images from seen classes, (2)
 734 **OCTDL-OUT**: 1,000 OCTDL images from unseen classes, and (3) **ChestMNIST**: 1,512 images
 735 from the ChestMNIST dataset (Wang et al., 2017).
 736

737 A.2 BASELINE DETAILS
 738

739 We compared ReconstructionNet with recent state-of-the-art uncertainty estimates.

- 740 1. **Entropy** (Shannon, 1948) is derived from prediction probability.
 741
- 742 2. **Monte Carlo Dropout (MCD)** (Gal & Ghahramani, 2016) We apply dropout ($p = 0.5$)
 743 to the penultimate layer of the model and keep dropout active during inference to yield T
 744 predictions (where $T = 100$ for the Covid and Fund datasets, and $T = 10$ for all other
 745 datasets), following the hyperparameters in (Gal & Ghahramani, 2016).
 746
- 747 3. **Deep Ensemble (DE)** (Lakshminarayanan et al., 2017) uses five models, with the standard
 748 deviation of their predictions as the uncertainty estimate.
 749
- 750 4. **Posterior Network (PN)** (Charpentier et al., 2020) computes aleatoric uncertainty as the
 751 inverse of the maximum prediction probability and epistemic uncertainty as the inverse of
 752 the maximum of the Dirichlet distribution parameters.
 753
- 754 5. **Bayesian Neural Networks (BNN)** (Jospin et al., 2022) estimate uncertainty as the stan-
 755 dard deviation of $T = 100$ predictions obtained with different weight samples.
 756
- 757 6. **Evidential Deep Learning (EDL)** (Sensoy et al., 2018) estimates uncertainty using the
 758 entropy of the predicted probabilities.
 759

756 A.3 METRIC DETAILS
 757

758 A.3.1 CORRELATION
 759

760 (Mi et al., 2022; Upadhyay et al., 2022) measures the reliability of the uncertainty estimate as the
 761 Pearson’s correlation coefficient between the uncertainty estimate and the prediction error (mea-
 762 sured as the absolute difference between the one-hot label and the predicted probability). A higher
 763 correlation indicates a more reliable estimate, as prediction errors are likely to be high when the
 764 model’s uncertainty is high, and vice versa.

765

766 A.3.2 AURC (AREA UNDER RISK-COVERAGE CURVE)
 767

768 (Ding et al., 2020) quantifies the selectivity of the uncertainty estimate, indicating its usefulness in
 769 selective prediction i.e. making predictions only for confident instances. This is computed as the
 770 area under the risk-coverage curve, which plots the 0/1 loss (Risk) for instances with uncertainty
 771 scores within the $\alpha\%$ -percentile (Coverage) against the coverage. A selective uncertainty estimate
 772 would yield a low AURC, as it effectively reduces loss across all possible uncertainty thresholds.

773

774 A.3.3 SIGMA-RISK SCORE
 775

776 Evaluates the resilience of the uncertainty estimate to false negatives (incorrect instances with low
 777 uncertainty), which can lead to significant costs if many are left undetected. It is computed as the
 778 0/1 loss of instances with normalized uncertainty less than $\sigma = \{0.1, 0.2, 0.3, 0.4\}$. A lower σ -risk
 779 score indicates greater robustness to false negatives. To ensure robustness to outliers, we apply min-
 780 max normalization with the minimum and maximum values computed after outlier removal with the
 781 interquartile range method.

782

783 A.3.4 OOD DETECTION
 784

785 (Lakshminarayanan et al., 2017; Postels et al., 2020; Malinin & Gales, 2018) Evaluates how effec-
 786 tively the uncertainty estimate can distinguish between in-distribution and OOD instances. This is
 787 quantified using AUROC (Area Under the Receiver Operating Characteristic Curve), where the true
 788 label indicates whether an instance is OOD and the target score is the uncertainty estimate. A higher
 789 AUROC reflects a more reliable uncertainty measure capable of identifying OOD inputs.

790

791 A.4 IMPLEMENTATION DETAILS
 792

793 The use of error weights ω allows the model to decouple the modeling of joint prediction proba-
 794 bilities from the classification task, enabling simultaneous optimization of both objectives. It also
 795 dynamically scales errors across classes to address variations in reconstruction difficulty and adjusts
 796 the contribution of each feature to the prediction, acknowledging that not all reconstruction errors
 797 equally impact the final outcome.

798 For image datasets, to reduce the size of the ReconstructionNet model, we trained class-specific
 799 decoders sharing a common encoder instead of training separate class-specific autoencoders. The
 800 encoder was based on a ResNet18 (He et al., 2015) backbone, and the decoders were implemented
 801 as inverted ResNet18 backbones, replacing convolutional layers with transposed convolutions.

802 Hyperparameters for all models were determined using grid search on the validation set. For Re-
 803 constructionNet, we tuned the compression ratio (which determines the latent vector length as its
 804 product with the feature count) along with the number of encoder and decoder layers, the width
 805 of intermediate layers, and the loss function parameter β ($\beta = 1.25, 2, 0.75$ for Covid, Diabetes,
 806 and Fund datasets). Both the MLP and ReconstructionNet models were trained on the training set
 807 with oversampling using SMOTE (Chawla et al., 2002). All models were trained using the Adam
 808 optimizer with early stopping. Each experiment was repeated five times, and we report the mean
 809 and standard deviation of each metric.

810 A.5 VERIFYING UNCERTAINTY EXPLANATION PROPERTIES WITH TOY EXAMPLES
811812 A.5.1 IMPLEMENTATION INVARIANCE
813814 We illustrate the property of implementation invariance with a simple example. Suppose we have
815 two ReconstructionNet models, f and f' , that differ in architecture:

- 816 •
- f
- : Each class autoencoder is shallow and linear;
-
- 817 •
- f'
- : Each class autoencoder is deeper with nonlinearities.
-
- 818

819 If for every input \mathbf{x} , the two ReconstructionNet models yield identical prediction probabilities \hat{p} for
820 all classes, then:
821

- 822 1. The predicted classes
- $c^* = \arg \max_j \hat{p}_j$
- are identical,
-
- 823 2. The weighted reconstruction errors
- ϵ
- are identical,
-
- 824 3. The uncertainty explanations
- $\zeta(\mathbf{x})$
- , which are the feature-wise weighted reconstruction er-
-
- 825 rors of the predicted classes, are identical.
-
- 826

827 Thus, ReconstructionNet's uncertainty explanations are *implementation-invariant* under this defini-
828 tion of functional equivalence.
829830 A.5.2 SENSITIVITY
831832 We illustrate the sensitivity property with a simple example. Consider a dataset where each input \mathbf{x}
833 has three features, (x^1, x^2, x^3) . Suppose that for the predicted class c^* :

- 834 • The weighted reconstruction errors depend only on
- x^1
- and
- x^2
- ;
-
- 835 • The weight for
- x^3
- is zero:
- $w_{c^*}^3 = 0$
- , indicating that changes in
- x^3
- do not affect the class
-
- 836 prediction.
-
- 837

838 The uncertainty explanation for input \mathbf{x} is given by:
839

840
$$\zeta(\mathbf{x}) = [w_{c^*}^1 \|x^1 - \hat{x}_{c^*}^1\|^2 \quad w_{c^*}^2 \|x^2 - \hat{x}_{c^*}^2\|^2 \quad w_{c^*}^3 \|x^3 - \hat{x}_{c^*}^3\|^2] \quad (15)$$

841

842 Since $w_{c^*}^3 = 0$, the explanation assigns zero attribution to x^3 , which is irrelevant to the predic-
843 tion. Hence, ReconstructionNet's uncertainty explanations satisfy the *sensitivity* property: features
844 irrelevant to the prediction receive zero attribution in the weighted reconstruction error.845 A.5.3 CONSISTENCY
846847 We illustrate the consistency property with a simple example. Consider an input $\mathbf{x} = (x^1, x^2, x^3)$;
848 For the predicted class c^* , the uncertainty explanation is:
849

850
$$\zeta(\mathbf{x}) = [w_{c^*}^1 \|x^1 - \hat{x}_{c^*}^1\|^2 \quad w_{c^*}^2 \|x^2 - \hat{x}_{c^*}^2\|^2 \quad w_{c^*}^3 \|x^3 - \hat{x}_{c^*}^3\|^2] \quad (16)$$

851

852 Now consider a modification f' of the model f where feature x^2 is made more important for the
853 prediction (i.e., its weight increases in the softmax over reconstruction errors).

- 854 • The reconstruction errors
- $\|x^j - \hat{x}_{c^*}^j\|^2$
- remain unchanged, since the autoencoders are un-
-
- 855 changed.
-
- 856 • The weight for
- x^2
- increases:
- $w_{c^*}^{2'} > w_{c^*}^2$
- .
-
- 857

858 Then the uncertainty attribution for x_2 becomes:
859

860
$$\zeta(\mathbf{x})^{2'} = w_{c^*}^{2'} \|x_2 - \hat{x}_{c^*}^2\|^2 \geq w_{c^*}^2 \|x_2 - \hat{x}_{c^*}^2\|^2 = \zeta(\mathbf{x})^2. \quad (17)$$

861

862 Thus, increasing a feature's contribution to the prediction does not decrease its uncertainty attribu-
863 tion, demonstrating that ReconstructionNet's uncertainty explanations satisfy the *consistency* prop-
erty.

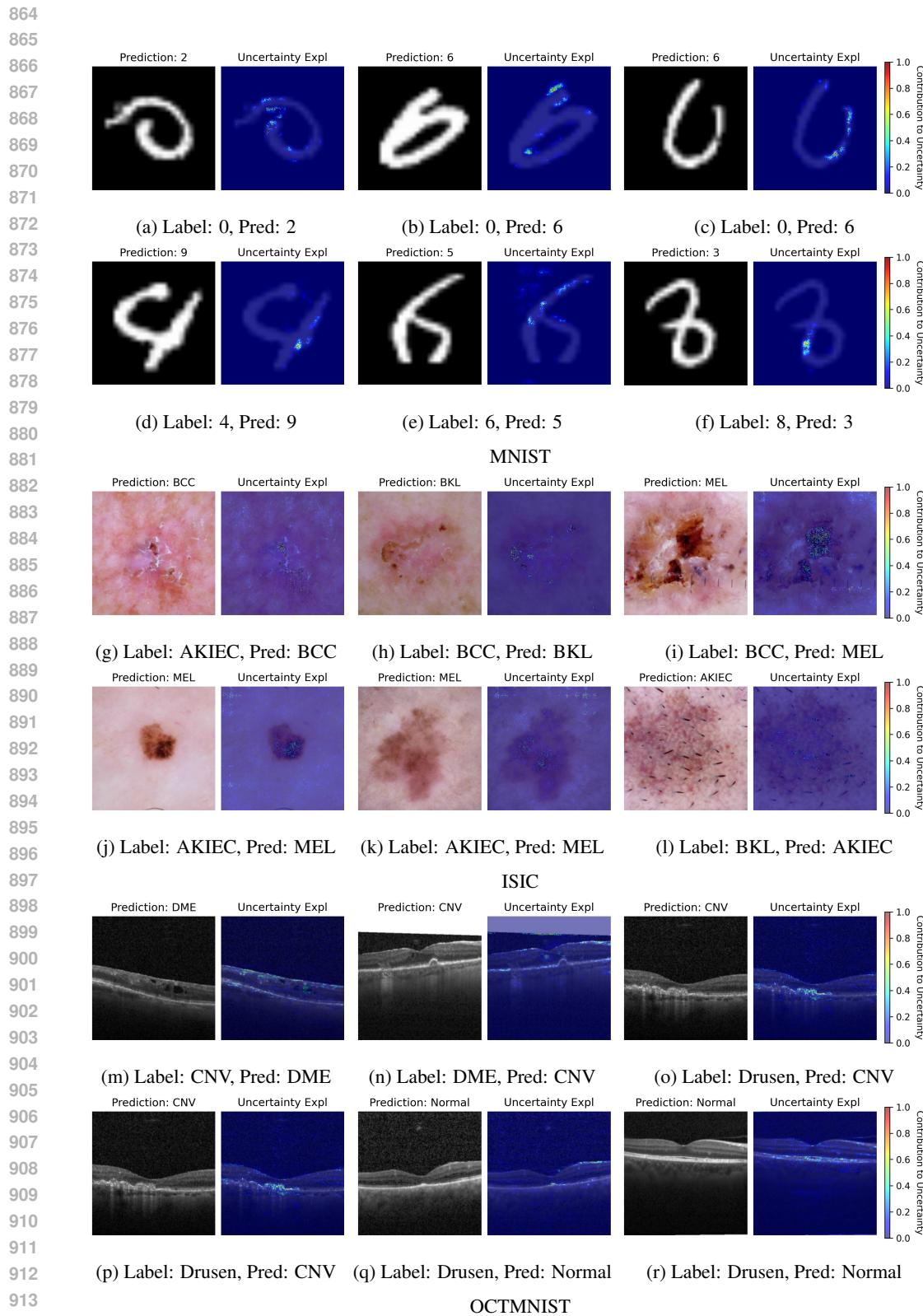


Figure 4: Uncertainty explanation illustration for MNIST, ISIC and OCTMNIST datasets. Positive uncertainty explanations were min-max normalized, gamma-corrected, and overlaid on the images for clarity. The highlighted regions “explain” the prediction uncertainty.

918
919

A.6 ADDITIONAL UNCERTAINTY EXPLANATION EXAMPLES

920

MNIST: The examples illustrate how the model’s uncertainty explanations correspond to ambiguous digit features that resemble other classes. In Figure 4a, the model predicted a 2 instead of a 0, showing uncertainty around the missing connection in the 0. In Figure 4b, it predicted a 6 instead of a 0, with uncertainty focused on the tail of the 6. In Figure 4c, the model again predicted a 6 instead of a 0, highlighting uncertainty on the right curve that resembles a 0. In Figure 4d, it predicted a 9 instead of a 4, with uncertainty at the connecting bump of the 4 (which is atypical in a 9). In Figure 4e, the model predicted a 5, with uncertainty concentrated on the top tail of the digit. Finally, in Figure 4f, it predicted a 3 instead of an 8, highlighting the lower-left connecting curve, whose removal would make the digit resemble a 3 more strongly.

929

ISIC: The examples demonstrate how the model’s uncertainty explanations capture visual features that increase ambiguity in lesion classification, leading to misclassifications. In Figures 4g and 4h, the model misdiagnoses the dermoscopic image, with the uncertainty explanations highlighting a scab-like region. In Figures 4i, 4j and 4k, images were misclassified as MEL (melanoma) instead of BCC or AKIEC, with uncertainty concentrated on the uneven pigmentation of the lesion, a feature often associated with melanoma. In Figure 4l, the image was misclassified as AKIEC with uncertainty explanations highlighting the presence of hairs covering a substantial portion of the lesion.

936

OCTMNIST These cases illustrate how the model’s uncertainty highlights regions that contribute to misclassification in OCT images. In Figure 4m, the model classified the OCT image as DME, with uncertainty concentrated around one of the fluid pockets. In Figure 4n, the model predicted CNV instead of DME, again highlighting a fluid pocket. This reflects the ambiguity introduced by fluid pockets, which are common to both conditions. Figures 4o, 4p, 4q, and 4r are Drusen images misclassified as other conditions, with uncertainty explanations highlighting the deposits (uneven bumps in the OCT images), which are key features of Drusen and major contributors to the model’s uncertainty in each incorrect prediction.

944

945

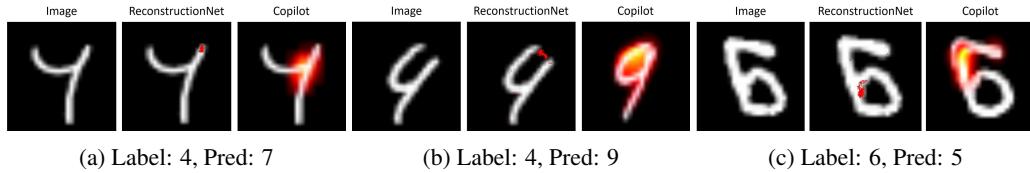
A.7 COMPARISON WITH LLMs

946

947

948

949



950

Figure 5: Comparison of uncertainty explanations from ReconstructionNet and Microsoft Copilot on the MNIST dataset. Positive uncertainty explanations from ReconstructionNet were min–max normalized, with pixels having attribution greater than 0.2 shown in red and overlaid on the images for clarity. Microsoft Copilot explanations were generated using the prompt: “This image has been misclassified as [image class prediction]. Generate an image highlighting the regions contributing to prediction uncertainty in red.” The highlighted regions are intended to “explain” the prediction uncertainty.

955

963

Figure 5 compares uncertainty explanations from ReconstructionNet and a publicly available, off-the-shelf large language model (LLM; Microsoft Copilot). We observe that ReconstructionNet produces more selective explanations, highlighting only small regions that contribute to uncertainty, whereas Copilot often highlights large areas (notably, in Figure 5b the entire digit is highlighted). Moreover, the regions identified by ReconstructionNet are more accurate. For example, in Figure 5a, where the digit 4 is misclassified as 7, ReconstructionNet correctly highlights the additional vertical line on the right, while Copilot highlights the connecting region that is common to both 4 and 7 and should not confuse the model. Similarly, in Figure 5c, where a 6 is misclassified as 5, ReconstructionNet pinpoints the extra vertical line on the bottom left that distinguishes 5 from 6, whereas Copilot highlights the top-left vertical line, which can occur in both digits.

972 A.8 RELATION TO CONTRASTIVE LEARNING AND SIAMESE NETWORKS
973

974 Note that, although the training procedure of ReconstructionNet shares similarities with contrastive
975 learning (Chopra et al., 2005), as both aim to minimize the distance between instances of the same
976 class while maximizing the distance between instances of different classes, there are key differences
977 in how these distances are computed. In contrastive learning, the distance is explicitly calculated
978 as the L2-norm between the embeddings of a pair of instances. In ReconstructionNet, the dis-
979 tance between an instance and the instances of a particular class is represented as the reconstruction
980 error of the corresponding class autoencoder. This distinction in distance measurement leads to
981 differences in the training loss. While contrastive learning incorporates the L2-norm distance be-
982 tween pairs of instances, ReconstructionNet uses class reconstruction error for its loss function.
983 Furthermore, in contrastive learning, the distances between instances of different classes are ex-
984 plicitly maximized and included in the loss function. In contrast, ReconstructionNet exclusively
985 trains each class autoencoder with instances belonging to its specific class. The architecture of
986 ReconstructionNet shares similarities with Siamese networks (Bromley et al., 1993) in its use of
987 sub-networks. However, unlike Siamese networks, which share weights across sub-networks, Re-
988 constructionNet allows each class autoencoder to have distinct weights.

989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025