

Uncertainty-Aware Systems for Human-AI Collaboration: Generative and Conformal Models in Dynamic, Resource-Constrained Environments

Vasco Thomas Serrão Pearson
Instituto Superior Técnico, Lisboa, Portugal

October 2024

Abstract

Although machine learning models are now ubiquitous in high-stakes decision-making scenarios, full automation is prevented by these model’s limitations: their performance is often constrained by their training data, making them ineffective in the face of distribution shifts, and their lack of transparency undermines human trust in these systems. Human-AI collaboration (HAIC) has been proposed as a solution to these issues, leveraging the complementary strengths of human experts to mitigate the models’ limitations. Despite its promise, the leading framework for HAIC, learning to defer (L2D), has a critical shortcoming: it struggles in dynamic environments where machine learning models fail to generalize to out-of-distribution data, often producing inaccurate predictions. Therefore, estimating model uncertainty correctly and leveraging humans’ adaptability becomes essential in these scenarios. In this thesis, we propose two uncertainty-aware methods to enhance HAIC systems in dynamic environments. The first enhances L2D by employing distance-aware models, combining machine learning outputs with a density function to enhance robustness, thus providing reliable uncertainty estimates. The second method uses density-based conformal prediction to assess epistemic uncertainty, deciding whether an instance should be processed through L2D or deferred directly to human experts via rejection learning. We further extend both methods to handle cost-sensitive scenarios, limited human predictions, and capacity constraints. Using constraint programming, we optimize the assignments to ensure each decision-maker, human or machine, handles the instances they are most likely to classify correctly. Our results demonstrate that both approaches outperform state-of-the-art baselines, particularly in dynamic environments, while also improving calibration.

Keywords: human-ai collaboration, learning to defer, uncertainty estimation

1 Introduction

In recent years, advances in artificial intelligence (AI) and machine learning (ML) have led to models that match or surpass human accuracy in tasks across various sectors, including finance [2, 25], criminal justice [13], and healthcare [10, 15, 35]. These models demonstrate remarkable speed and scalability. However, despite their strengths, ML models face limitations in high-stakes scenarios due to their reliance on training data, struggles with generalization in dynamic environments, and lack of transparency [30, 33]. As a result, their applicability in fully automated decision-making systems remains constrained.

To mitigate these limitations, human-AI collaboration (HAIC) systems have been developed, combining the strengths of both AI and human experts [7, 8]. AI models excel at processing large datasets efficiently, while humans offer flexibility, adaptability, and the ability to engage in causal reasoning [14]. A key challenge in these systems is deciding when to defer decisions to humans: rejection learning addresses this by deferring uncertain cases to humans [6], while learning to defer (L2D) improves

upon this approach by estimating both AI and human confidence, allowing for better allocation of instances between the decision-makers [27, 29].

Despite its promise, L2D faces significant challenges in dynamic environments, where evolving data distributions can render previous model assumptions inaccurate [11]. Additionally, L2D methods often require human predictions for every instance, which are often unavailable due to prohibitive data acquisition costs. Most L2D systems also overlook human capacity constraints and fail to account for the varied costs of misclassifications, which can result in suboptimal decision-making in cost-sensitive applications [26].

To advance human-AI collaboration systems in dynamic environments, we introduce three main contributions. First, a distance-aware L2D system is proposed, designed to enhance robustness when data distributions change. This approach combines traditional L2D with density-aware models [3], leveraging density-based adjustments to better estimate confidence levels in unfamiliar regions of the feature space, thereby reducing overconfidence and improving calibration.

The second contribution is a hybrid system that integrates density-based conformal prediction [16, 28] to complement L2D with rejection learning in dynamic environments. By using conformal prediction, the system identifies instances that lie outside the training data distribution and defers these directly to human experts, applying a rejection learning strategy. For in-distribution data, the system utilizes L2D to make optimal assignment decisions based on reliable correctness estimates. Both these proposed systems support cost-sensitive scenarios, where different errors may have different costs. They also consider human work-capacity constraints, by leveraging constraint programming to calculate the optimal assignment of instances.

Finally, we conduct an experimental study in a cost-sensitive fraud detection context to validate the effectiveness of these systems. We perform tests under a wide array of conditions, comparing the performance of the proposed methods against a state-of-the-art L2D method, as well as against rejection learning and random deferral baselines, demonstrating that our approaches outperform the baselines across a set of diverse, realistic settings.

2 Background

2.1 Rejection Learning and Uncertainty Estimation

The simplest deferral approach in the literature is *rejection learning* [5, 6, 12]. In a HAIC setting, rejection learning defers to humans those instances the model rejects to predict. This technique often involves abstaining from predicting in cases of high uncertainty, allowing human experts to handle uncertain instances [20]. Hendrickx et al. [19] categorize rejection into two types: *ambiguity rejection*, which enables the model to abstain in scenarios where the target values are inherently ambiguous, and *novelty rejection*, where the model refrains from predicting instances that deviate significantly from the training data.

These types of rejection align with distinct types of uncertainty: *aleatoric uncertainty*, which stems from irreducible randomness within the data (such as class overlap) and leads to ambiguity rejection, and *epistemic uncertainty*, which arises from incomplete knowledge—whether due to out-of-distribution (OOD) data or uncertainty about the correct model to fit the data, leading to novelty rejection [21]. Several studies have proposed methods for distinguishing these types of uncertainty to improve applications like rejection learning [34] and active learning [31].

Density estimation methods are effective for identifying epistemic uncertainty by assessing where the feature space is sparsely populated. Bui and Liu [3] propose the density-softmax model, which uses

a classifier that incorporates density scores in the softmax layer, thereby improving uncertainty estimation, particularly in the presence of distribution shifts. Similarly, Hechtlinger et al. [16] propose a conformal prediction approach that combines density estimation with set-valued predictions, providing empty set predictions for instances in regions of high epistemic uncertainty and multi-class predictions where aleatoric uncertainty is detected. In our work, we will leverage these density-based techniques to enhance the robustness and adaptability of our HAIC systems.

2.2 Current L2D Methods

The L2D framework was introduced by Madras et al. [27] to address the shortcomings of confidence-based rejection learning, which does not account for the human decision-maker’s performance. By training a main classifier and a deferral mechanism, L2D considers both model and human errors, optimizing deferral decisions accordingly. Mozannar and Sontag [29] extended this by highlighting that Madras et al.’s [27] approach is not consistent, proposing a consistent surrogate loss that includes a separate class for deferral. Verma and Nalisnick [37] critique the L2D surrogate loss developed by Mozannar and Sontag [29] for miscalibration issues, introducing a one-vs-all (OvA) approach that improves calibration by training the classifier and deferral model independently. In the multi-expert setting, Keswani et al. [24] extend L2D to assign instances to multiple experts, while Verma et al. [38] propose a new consistent and calibrated loss, generalizing the loss from [37] to handle multi-expert scenarios.

However, L2D methods face several limitations in practical deployment. They struggle to adapt to evolving data distributions, which can cause model performance to degrade over time [26, 32]. Moreover, these frameworks rely heavily on labeled data from all experts, making them impractical in real-world scenarios due to high labeling costs. Most existing L2D methods also focus on minimizing zero-one loss, which overlooks cost-sensitive scenarios where misclassification costs can vary. Finally, they often do not consider human work capacity constraints when deciding who predicts on a given instance.

Recent advancements in L2D research have aimed to address some of these practical constraints, particularly the challenge of learning with limited expert data. Charusaie et al. [4], Hemmer et al. [18], and Tailor et al. [36] explore active learning, semi-supervised learning, and meta-learning, respectively, to reduce the need for extensive human annotations. On the other hand, Alves et al. [1] focus on cost and capacity constraints, which are crucial for real-world applications.

3 Density-Softmax

Existing L2D systems struggle in dynamic environments where data distributions shift over time. To address this, we incorporate density-aware modeling using the density-softmax approach [3] to develop an L2D system that is more robust against distribution shifts, ensuring improved calibration, reduced overconfidence on OOD data, and more detailed expert modeling in scenarios with limited data.

The density-softmax model, proposed by Bui and Liu [3], consists of a feature extractor, a normalizing flows-based density estimator, and a classifier. The feature extractor maps inputs into a latent space, which the density model uses to estimate the likelihood of observing each instance under the training distribution. Specifically, we use a RealNVP model to calculate the log-likelihood for each data point x . This is done by transforming x into a latent representation $z = f(x; \alpha)$, where α represents the parameters of the RealNVP model. The transformation maps the input space into a simpler distribution, usually a standard Gaussian, where the log-likelihood is computed. The resulting likelihood $p(f(x; \alpha))$ is then combined with the classifier logits to adjust predictions. For a classifier with logits $x^T \theta_{g_k}$ for each class k , the predictive probability is computed as:

$$p(y = i|x) = \frac{\exp(p(f(x; \alpha)) \cdot (x^T \theta_{g_i}))}{\sum_{j=1}^K \exp(p(f(x; \alpha)) \cdot (x^T \theta_{g_j}))}. \quad (1)$$

This integration ensures that for OOD instances (low-density regions), the classifier’s output probabilities are scaled down, reflecting higher uncertainty. As a result, the model is less likely to make overconfident predictions in unseen regions of the feature space.

These distance-aware classifiers are used in our L2D system, which is built using the one-vs-all (OvA) framework developed by Verma et al. [38]. The OvA approach constructs a classifier $h : \mathcal{X} \rightarrow \mathcal{Y}$ and a rejector $r : \mathcal{X} \rightarrow \{0, 1, \dots, J\}$. When $r(x_i) = 0$, the classifier h makes the decision, while for $r(x_i) = j$, the decision is deferred to the j th human expert. The classifier itself consists of K functions $g_k : \mathcal{X} \rightarrow \mathbb{R}$, where each function g_k estimates the probability of the instance belonging to class k . Similarly, the rejector has J functions $g_{\perp, j} : \mathcal{X} \rightarrow \mathbb{R}$ to estimate the likelihood of each expert making the correct decision. Each binary classifier can be trained independently by minimizing a proper binary loss (e.g., log-loss), allowing us to train these models even when expert data is limited (e.g. having one expert prediction per instance). Each of the $K + J$ binary classifiers is combined with a density function, as described in the density-softmax approach, to improve uncertainty estimation and en-

sure reliable correctness estimates for each decision-maker. Since the decision to assign an instance to a particular decision-maker relies on these correctness estimates, they must provide calibrated scores and reflect the uncertainty of each instance.

To handle epistemic uncertainty, particularly in low-density regions, we adjust the model’s estimate of expert correctness. Specifically, for any given expert, we blend the model’s predicted logits with a value that reflects the expert’s historical average performance when faced with high epistemic uncertainty. The adjusted logits are given by:

$$p(f(x; \alpha)) \hat{g}_{\perp, j} + (1 - p(f(x; \alpha))) \sigma^{-1}(\hat{p}_{j, \text{avg}}), \quad (2)$$

where σ^{-1} represents the inverse sigmoid function, $\hat{g}_{\perp, j}$ represents the logit score of the binary classifier for expert j , and $\hat{p}_{j, \text{avg}}$ is the expert’s average correctness on the training set. This adjustment ensures that the model reflects a more realistic estimate of expert accuracy in high-uncertainty situations, balancing between the model’s own estimates and historical expert performance.

4 Conformal Prediction for HAIC

In this section, we introduce a different approach to enhancing HAIC systems by integrating uncertainty estimation mechanisms. We hypothesize that while L2D methods perform well on familiar, in-distribution cases, they should avoid handling instances that lie outside the training distribution. For such OOD cases, the model’s probability estimates can be poorly calibrated, resulting in suboptimal decisions. In these scenarios, human experts are better suited to generalize and adapt due to their ability to leverage broader contextual knowledge and handle novel situations. Furthermore, in high-stakes environments, it may be preferable—or required—that OOD instances be reviewed by human experts to ensure accountability and explainability. To meet these requirements, we propose a system that defers OOD instances to human experts through rejection learning, while relying on L2D for in-distribution cases.

To implement this system, we utilize density-based conformal prediction [16, 28] to assess epistemic uncertainty on a per-instance basis. This method allows us to determine when an instance is likely to be OOD, signaling the need for expert intervention, and when it is preferable to proceed with L2D for decision assignment. Density-based conformal prediction uses a two-step training process involving a proper training set $D^{tr} = (X^{tr}, Y^{tr})$ and a calibration set $D^{cal} = (X^{cal}, Y^{cal})$. First, a class-conditional density estimator $\hat{p}(x|y)$ is built using D^{tr} . The calibration set is then used to determine the empirical $1 - \alpha$ quantile \hat{q}_y of the density values

for each class,

$$\hat{q}_y = \sup \left\{ t : \frac{1}{n_y} \sum_{z_i \in D_y^{cal}} \mathbb{I}(\hat{p}(x_i|y) \geq t) \geq 1 - \alpha \right\}, \quad (3)$$

where n_y is the number of elements belonging to class y in D^{cal} , $z_i = (x_i, y_i)$, $\mathbb{I}(\hat{p}(x_i|y) \geq t)$ equals 1 when the estimated probability $\hat{p}(x_i|y)$ is greater than or equal to t , and 0 otherwise, and $D_y^{cal} = \{z_i \in D^{cal} : y_i = y\}$ is the subset of calibration examples in class y . This quantile effectively acts as a threshold, allowing the model to define, for any new observation x_{n+1} , a prediction set

$$\mathcal{C}_\alpha(x_{n+1}) = \{y \in \mathcal{Y} : \hat{p}(x_{n+1}|y) \geq \hat{q}_y\}, \quad (4)$$

which includes all classes y for which the observed density is above the threshold. Instances that deviate significantly from the training distribution are predicted as the empty set and consequently deferred to human experts. Conversely, non-null predictions allow the L2D system to make optimal assignments based on the estimated correctness of both model and human experts.

The training and prediction algorithms are defined in Algorithms 1 and 2.

Algorithm 1: Training algorithm

Input: Training data
 $Z = (x_i, y_i), i = 1 \dots n$, Class list \mathcal{Y} ,
Confidence level α , Ratio p

Output: $\hat{p}_{list}, \hat{q}_{list}$

Initialize: $\hat{p}_{list} = list, \hat{q}_{list} = list$

for $y \in \mathcal{Y}$ **do**
 $X_y^{tr}, X_y^{cal} \leftarrow \text{SubsetData}(Z, \mathcal{Y}, p)$
 $\hat{p}_y \leftarrow \text{LearnDensityEstimator}(X_y^{tr})$
 $\hat{q}_y \leftarrow \text{Quantile}(\hat{p}_y(X_y^{cal}), \alpha)$
 $\hat{p}_{list}.append(\hat{p}_y)$
 $\hat{q}_{list}.append(\hat{q}_y)$
end for

return $\hat{p}_{list}, \hat{q}_{list}$

Algorithm 2: Prediction algorithm

Input: Input to be predicted x , Trained
 $\hat{p}_{list}, \hat{q}_{list}$, Class list \mathcal{Y}

Output: \mathcal{C}

Initialize: $\mathcal{C} = list$

for $y \in \mathcal{Y}$ **do**
if $\hat{p}_y(x) \geq \hat{q}_y$ **then**
 $\mathcal{C}.append(y)$
end if
end for

return \mathcal{C}

Despite the challenges of density estimation in high dimensions, this approach has proven effective

in previous work [16, 28]. The intuitive reason for this success is that we do not need the density estimates $\hat{p}(x|y)$ to be close to the actual density values $p(x|y)$, we only need the ordering imposed by $\hat{p}(x|y)$ to approximate the ordering defined by $p(x|y)$.

Hechtlinger et al. [16] show that $|P(y \in \mathcal{C}_\alpha(x_{n+1})) - (1 - \alpha)| \rightarrow 0$ as $\min_y n_y \rightarrow \infty$, ensuring the asymptotic validity of the model.

In conclusion, if a new instance results in a null set from conformal prediction, indicating high epistemic uncertainty, the system defers the instance to human experts using a rejection learning strategy. This strategy involves assigning the instance to the expert with the highest average correctness from the training data, while respecting capacity constraints. Conversely, when the prediction set is non-null, we employ the OvA L2D framework described above, assigning the instance to the most suitable decision-maker. Note that, in this system, the binary classifiers from the OvA approach are not combined with density models.

5 Cost and Capacity Constraints

To enhance the applicability of our HAIC systems in real-world scenarios, we outline how our approaches deal with cost-sensitive scenarios and how they manage assignments while respecting human work capacity constraints and limited data availability. We assume each instance i is represented by a feature vector x_i and a label $y_i \in \mathcal{Y} = \{0, 1\}$. The feature vector may have additional information available to experts, such as the ML model's score. In cost-sensitive settings, each instance has an associated misclassification cost c_i , representing the cost of an incorrect classification, while correct classifications incur no cost. Reflecting realistic conditions, we assume that not all experts provide predictions for every instance; rather, each expert j 's prediction $m_{i,j}$ for an instance i is treated as a separate instance within the training set. This setup allows flexibility, treating predictions from different experts for the same instance as distinct data points: $\{x_i, y_i, m_{i,j}, c_i\}$ and $\{x_i, y_i, m_{i,k}, c_i\}$. Our training set, therefore, is $S = \{x_i, y_i, m_{i,j}, c_i\}_{i=1, j=1}^{N, J}$, and our objective is to estimate the probability of correctness when deferring an instance to an expert or ML model, ultimately maximizing correctness under work capacity constraints.

5.1 Cost-Sensitive Learning

We adapt our methods to cost-sensitive scenarios, where each instance incurs a different error cost (e.g. FP and FN errors may have different costs). We follow the approach proposed by Zadrozny et al. [40], Elkan [9], and used by Alves et al. [1] in an L2D setting. This approach redefines the training distribution by weighting each instance according to its misclassification cost. In cost-sensitive settings,

our objective is no longer simply to minimize the error rate, but to minimize the total misclassification cost.

In standard learning setups, we typically minimize the expected error rate, represented by $\mathbb{E}_{(x,y)\sim D}[\mathbb{I}_{h(x)\neq y}]$, where D is the data distribution and $h(x)$ is the predicted label. However, in cost-sensitive contexts, each instance x_i carries an associated cost c_i , which we wish to incorporate into the objective. The objective then shifts to minimizing the expected misclassification cost, defined as $\mathbb{E}_{x,y,c\sim D}[c \cdot \mathbb{I}_{h(x)\neq y}]$.

To achieve this, Zadrozny et al. [40] show that we can redefine the data distribution as

$$\tilde{D}(x, y) = \frac{c}{\mathbb{E}_{c\sim D}[c]} D(x, y, c), \quad (5)$$

and under this modified distribution, the expected error rate is

$$\mathbb{E}_{x,y\sim \tilde{D}}[\mathbb{I}_{h(x)\neq y}] = \frac{1}{\mathbb{E}_{c\sim D}[c]} \mathbb{E}_{x,y,c\sim D}[c \cdot \mathbb{I}_{h(x)\neq y}]. \quad (6)$$

Thus, minimizing the error rate under \tilde{D} is equivalent to minimizing the expected misclassification cost under the original distribution D . In practical terms, this entails adjusting the weight of each instance during training according to its associated cost [9, 40]. To implement this approach in our OvA framework, we modify the standard log-loss functions, used for training both the classifier and the expert-correctness functions, by incorporating the misclassification costs as instance weights.

5.2 Capacity Constraints

To address the capacity constraints limitation, we propose a solution for incorporating human capacity constraints into both of our assignment systems. For the density-softmax approach, which functions as a purely L2D system, we apply the strategy proposed by Alves et al. [1], optimizing assignments under capacity constraints. For the conformal prediction approach, we extend this framework by creating a method that dynamically balances L2D with rejection learning while respecting capacity limitations.

Rather than processing the entire dataset at once, we impose constraints over batches of instances. This approach aligns with practical scenarios where data is processed in batches and allows the system to adapt dynamically to varying workload demands. We define a batch vector \mathbf{b} that assigns each instance i to a batch b containing n_b instances, and a human capacity matrix \mathbf{H} that sets the maximum workload each expert can handle per batch. Specifically, $H_{b,j}$ denotes the capacity for expert j within batch b .

In binary classification, each instance i is assigned a decision-maker a_i from $\{1, \dots, J+2\}$: if $a_i = y +$

1, the ML model predicts class y for the instance, while $a_i = j+2$ defers the decision to the j th human expert. We then frame the assignment optimization as

$$\mathbf{A}^* = \arg \max_{A \in \{0,1\}^{n_b \times (2+J)}} \sum_{i=1}^{n_b} \sum_{a_i=1}^{J+2} \hat{\mathbb{P}}(\text{correct} | x_i, a_i) A_{i,a_i}, \quad (7)$$

subject to two constraints: (1) $\sum_{i=1}^{n_b} A_{i,a_i} = H_{b,a_i}$ ensuring each expert meets his capacity, and (2) $\sum_{a_i=1}^{J+2} A_{i,a_i} = 1$ to ensure each instance has a unique assignment.

For the conformal prediction approach, balancing between L2D and RL is necessary. We introduce a binary function $f_\alpha(x_i)$, which indicates whether an instance will be assigned through L2D or RL, based on the coverage level α for the conformal prediction method. We then optimize this balance alongside the assignment matrix \mathbf{A} , with the objective defined in Equation 8 below. This optimization is subject to the same constraints mentioned above, and an additional constraint ensuring that when rejection learning is applied ($f_\alpha(x_i) = 0$), the instance is deferred to a human expert, preventing OOD instances from being handled by the model. Here, we also downweigh instances (through w_i) based on the coverage level α at which they fall into the null set, and if the proportion of null set predictions exceeds the expected rate from training data. This adaptation allows the system to respond to distribution shifts effectively, prioritizing expert work capacity for OOD instances when required.

6 Experimental Setup

6.1 Dataset

The bank-account-fraud (BAF) tabular dataset, developed by Jesus et al. [22], is a synthetic version of a real-world fraud detection dataset created with CTGAN [39]. It contains one million records spanning eight months, with each row representing a bank account application labeled as fraudulent (1) or legitimate (0). Fraudulent applications constitute only about 1% of the dataset, reflecting a significant class imbalance. The dataset includes 30 features (19 numeric, 6 binary, and 5 categorical) detailing applicant and application characteristics. A positive prediction indicates fraud, leading to application rejection, while a negative prediction suggests legitimacy, resulting in account opening. Misclassification costs differ: false positives (rejecting legitimate applications) lead to customer loss, whereas false negatives (accepting fraudulent applications) result in financial losses for the bank.

6.2 Misclassification Costs

Fraud detection is a cost-sensitive task that requires balancing the costs of false positives (c_{FP}) and false negatives (c_{FN}), making standard metrics like accu-

$$(\mathbf{A}^*, \alpha^*) = \arg \max_{\mathbf{A} \in \{0,1\}^{n_b \times (2+J)}, \alpha} \sum_{i=1}^{n_b} \sum_{a_i=1}^{J+2} \left(f_\alpha(x_i) \frac{1}{w_i} \hat{\mathbb{P}}(\text{corr.} \mid x_i, a_i) + (1 - f_\alpha(x_i)) \hat{\mathbb{P}}(\text{corr.} \mid X_{\text{train}}, a_i) \right) A_{i,a_i} \quad (8)$$

racy inadequate. Jesus et al. [22] apply a Neyman-Pearson criterion, aiming to maximize recall at a fixed 5% false positive rate, which aligns with industry goals to detect fraud while minimizing customer attrition. Although this criterion suits fraud detection’s class imbalance and error trade-offs, it doesn’t provide direct values for c_{FP} and c_{FN} necessary for cost-sensitive learning.

In cost-sensitive tasks, assuming no cost for correct classifications, the objective is to minimize the expected misclassification cost:

$$\frac{1}{N} \sum_{i=1}^N [\lambda \mathbb{I}[y_i = 0 \wedge \hat{y}_i = 1] + \mathbb{I}[y_i = 1 \wedge \hat{y}_i = 0]], \quad (9)$$

where $\lambda = c_{FP}/c_{FN}$. We derive this cost trade-off parameter based on the Neyman-Pearson criterion, allowing us to set $c_{FP} = \lambda$ and $c_{FN} = 1$ for our misclassification cost re-weighting approach.

Elkan [9] shows that we can relate a binary classifier’s optimal threshold t to c_{FP} and c_{FN} . The positive class is predicted if $(1 - p)c_{FP} \leq p c_{FN}$, where p is the probability of fraud. At equality, we have:

$$\lambda_t = \frac{t}{1 - t}, \quad (10)$$

which provides a theoretical basis to set $c_{FP} = \lambda_t$ and $c_{FN} = 1$ based on the Neyman-Pearson criterion’s optimal threshold. We use the alert model described in the following section to determine the value of λ_t through its optimal threshold, which yields $\lambda_t = 0.056$.

6.3 Alert Setup

To create a realistic HAIC scenario, we incorporate an alert model, as used in practical settings like fraud detection, to screen and flag suspicious instances for human review. This setup allows human experts to focus on a critical subset of the feature space, optimizing their work capacity. Our assignment systems will function alongside an alert model, which will screen instances and raise alerts; our systems will then be trained and tested on these flagged instances.

The alert model is trained on data from the first three months of the data, validated on the fourth month, and deployed over months 4 to 8 (see Figure 1). We employ LightGBM for the alert model due to its high performance with tabular data and computational efficiency on large datasets [23]. The model minimizes the binary cross-entropy loss during training.



Figure 1: Training, validation, and deployment splits for the alert model and assignment systems.

On the validation set, the alert model achieved a recall of 58.48% at a 5% false positive rate (FPR), using a threshold of $t = 0.0526$. During the deployment phase, it achieved a recall of 50.93% with a 4.2% FPR on flagged instances and a ROC-AUC of 0.9. Over the deployment period, the model flagged about 29,000 instances, maintaining a fraud prevalence of 12%, which provides the data for training and testing our assignment systems.

6.4 Noise injection

To evaluate our system’s ability to handle data drift and detect OOD instances, we introduce noise into the test set, which comprises alerts from the eighth month. This approach creates a known subset of modified data, enabling targeted assessment of our systems on these altered instances.

We generate five test set variants, each with a unique noise configuration: low, medium, and high noise settings, where all features are modified with progressively increasing noise levels; a categorical noise setting, where only categorical features are affected; and a numerical noise setting, where only numerical features are altered. For numerical features, data is standardized (mean of 0, unit variance), then modified by adding zero-mean Gaussian noise with standard deviations of 1.0 (low), 1.5 (medium and numerical), and 2.0 (high), and finally transformed back to the original scale. Discrete values are rounded, and values in bounded features are clipped to stay within their limits. For categorical and binary features, we randomly switch values with probabilities of 0.3 (low), 0.4 (medium and categorical), and 0.5 (high).

Each modified test set comprises 80% original and 20% noisy data. To account for statistical variation, we repeat the noise injection five times per setting with different random seeds and report average performance across these iterations.

6.5 Synthetic Experts

To address the lack of datasets with human predictions, we generate synthetic expert decisions by

simulating human error based on input features and the alert model score, a more realistic approach than solely relying on label noise. Following Alves et al.’s [1] approach, we define each expert’s probability of error using instance-dependent label noise. Specifically, for a given instance x_i with label y_i , the probability of error is influenced by the normalized features \bar{x}_i and the alert model’s score $M(x_i)$. The expert’s error probabilities are given by:

$$\begin{cases} \mathbb{P}(m_{j,i} = 1 | y_i = 0, \mathbf{x}_i) = \sigma\left(\beta_0 - \alpha \frac{\mathbf{w} \cdot \bar{\mathbf{x}}_i + w_M M(\mathbf{x}_i)}{\sqrt{\|\mathbf{w}\|^2 + w_M^2}}\right) \\ \mathbb{P}(m_{j,i} = 0 | y_i = 1, \mathbf{x}_i) = \sigma\left(\beta_1 + \alpha \frac{\mathbf{w} \cdot \bar{\mathbf{x}}_i + w_M M(\mathbf{x}_i)}{\sqrt{\|\mathbf{w}\|^2 + w_M^2}}\right) \end{cases} \quad (11)$$

where β_0 , β_1 , α , w , and w_M control the base error rate and the influence of the features and alert model score.

We align the expected cost of an expert’s decisions with the main classifier’s training misclassification cost $\mathbb{E}[C]_h$. We sample each expert’s target cost $\mathbb{E}[C]_j$ from $T_{\mathbb{E}[C]_j} \sim \mathcal{N}(\mathbb{E}[C]_h, 0.2\mathbb{E}[C]_h)$. Based on the target misclassification cost, we first sample each expert’s FPR randomly and then calculate the corresponding FNR required to achieve $T_{\mathbb{E}[C]_j}$. We then adjust β_0 and β_1 to match the sampled FPR and FNR. Our experiments consider 1 to 5 experts, sampling from a pool of 15 synthetic experts with different properties across trials to enhance robustness.

6.6 Data Availability and Capacity Constraints

To reflect the practical constraints of expert label availability, we simulate realistic scenarios of limited expert labels across four data availability conditions. The first condition assumes expert labels are available for all training instances, a common but unrealistic assumption in previous L2D approaches [17, 38]. The remaining three conditions reflect limited data availability, where each expert labels only a fraction of the dataset: either 1/5, 1/20, or 1/40 of the total instances. Each subset of training data is randomly assigned to experts across five seeds for each data availability scenario to introduce variability. In the density-softmax approach, the density models are trained on the same subsets used for training the expert models.

In addition to data availability constraints, we impose uniform capacity constraints on experts, assuming all experts have similar work capacities, and we experiment with different deferral rates (the percentage of flagged instances assigned to human experts) from 10% to 50%. This way we can assess the system’s ability to identify and defer OOD instances effectively, even under low deferral rates.

Our experiments explore these scenarios under four main variables: noise, which includes five levels and types as detailed in Section 6.4; number of

experts, which ranges from 1 to 5 experts; deferral rate, which varies from 10% to 50% in 10% increments; and data availability, which includes varying levels of expert-labeled training data as outlined above. These combinations result in 500 unique testing scenarios. Given this extensive number, results will often focus on representative settings within this range.

6.7 Baselines

Learning to Defer: For the L2D baseline, we implement the OvA L2D algorithm by Verma et al. [38], using a standard LightGBM model for each binary classifier instead of our distance-aware models. This setup mirrors our density-softmax approach but without the distance-awareness feature, allowing us to isolate the impact of our contributions. It also serves as a baseline for the conformal prediction approach, as the same L2D method is applied to instances not classified as OOD.

Rejection Learning: In this approach, we use density-based conformal prediction to measure uncertainty and determine which instances should be deferred to a human expert. Instances predicted as the empty set are randomly deferred, while the rest are assigned to the ML model. The coverage level, α , is chosen to align the number of null set predictions with expert work capacity.

Random Assignment: As a simple baseline, we assign a randomly selected subset of test instances to the experts, ensuring the number matches the experts’ work capacity.

6.8 System Training

Density-Based Conformal Prediction: To implement density-based conformal prediction, we split our data into a proper training set (alert data from months 4 to 6) and a calibration set (month 7). Following the method by Messoudi et al. [28], we train an MLP to minimize the weighted log-loss and use its penultimate dense layer as a feature extractor. These feature vectors are then used to perform Gaussian kernel density estimation (KDE) with a bandwidth of 0.5. KDE models are trained separately for each class on the proper training data. Next, we determine the empirical $1 - \alpha$ quantiles \hat{q}_y using the calibration set, which allows us to generate prediction sets as outlined in our algorithms.

Density-Softmax: For the density-softmax approach, we employ the same MLP feature vectors from the penultimate layer as inputs for RealNVP models, which estimate the likelihood for each instance. We train a separate RealNVP density model for each expert and data availability scenario using the training and validation subsets labeled by each respective expert. These density models are then integrated with binary classifiers, as described in our density-softmax methodology. The training

of these binary classifiers is described below.

Learning to Defer: We use LightGBM models to train both the primary ML classifier h and the expert-specific models that estimate their correctness. These models are trained on alerts from months 4 to 6 and validated on month 7. The primary classifier is trained to predict true labels, while the expert models are trained to predict a label of 1 if the expert’s prediction is correct and 0 otherwise. Both the primary classifier and expert models are optimized to minimize the weighted log-loss, following the cost-sensitive learning method described in Section 5.1.

7 Results

7.1 Density-Softmax Approach

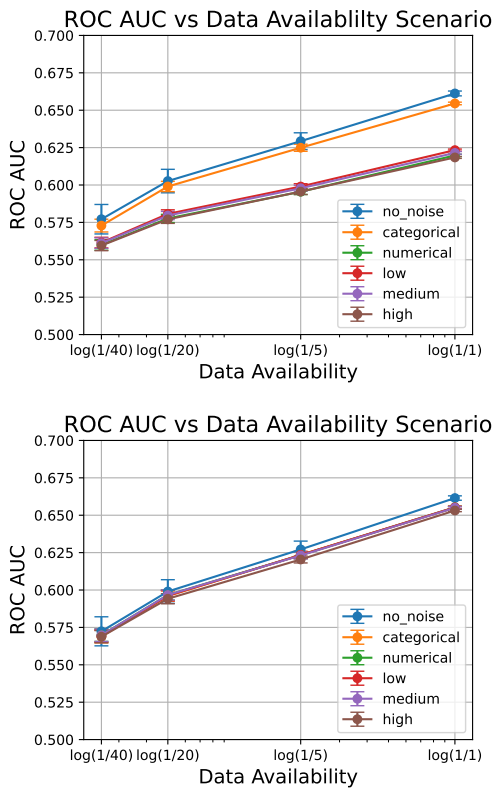


Figure 2: Mean ROC-AUC for the binary classifiers that estimate the correctness of the experts. On the bottom plot, the binary classifiers are combined with the density models. Values are calculated for each model and averaged, with error bars representing the 95% confidence intervals for the mean, accounting for randomness in the injection of noise and selection of training data.

In the density-softmax approach, we evaluate the classifier h and the expert models, both with and without the density models. The results for models without density-based adjustments are used directly in the L2D algorithm for the conformal

prediction method and the L2D baseline. The distance-aware models are used in the density-softmax L2D method.

Incorporating density models enhances calibration and performance under conditions of distribution shift. Figure 2) show that performance improvements are substantial for the expert models. Without density models, noise negatively impacts ROC-AUC in most scenarios except categorical noise. However, with density models, the differences between ROC-AUC values in noisy and no-noise settings become negligible. Moreover, the Expected Calibration Error (ECE) improves consistently with density models across different levels of labeled data. Notably, using the full set or one-fifth does not affect calibration when the density-softmax method is employed, indicating robust calibration performance. For the classifier h , adding density models also reduces the ECE across different noise scenarios.

To illustrate the system’s ability to detect OOD data, we use t-SNE to visualize the density scores under the high-noise setting. As shown in Figure 3, density scores in regions with high noise are significantly lower (close to 0), reflecting the method’s capacity to identify OOD data. On the other hand, they remain close to 1 in in-distribution areas.

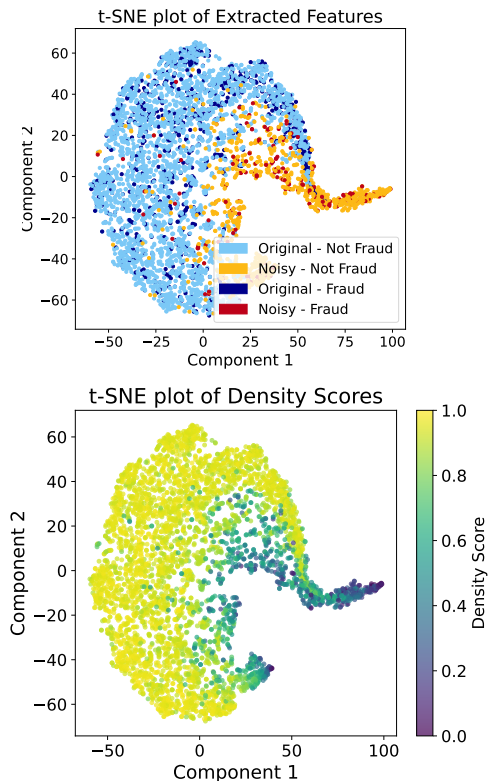


Figure 3: t-SNE visualization of the density scores on the test set in the high noise setting.

7.2 Conformal Prediction Approach

In the conformal prediction approach, our goal is to assess how effectively the method identifies noisy and OOD data by classifying these instances as empty sets. As shown in Figure 4, in higher noise settings, the conformal prediction method predicts nearly all noisy data as empty sets, even at low α values. This indicates that the model effectively deals with data drift by deferring uncertain predictions to experts. Note that at low α values, we expect very little empty set predictions, while higher α values gradually increase the number of empty set predictions until all instances are classified as empty sets at $\alpha = 1$.

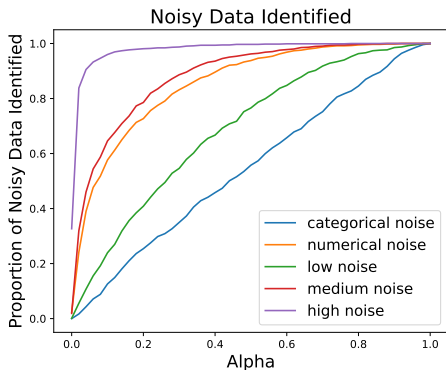


Figure 4: Effectiveness of the conformal prediction method in identifying OOD data. The plot shows the proportion of noisy data predicted as the null set for each coverage level.

Additionally, as more OOD data is deferred to experts, calibration improves for non-null set predictions. By filtering out these OOD instances, the conformal prediction method leaves the L2D system to operate on data where calibration estimates are more reliable. This rejection learning approach ensures that the remaining in-distribution data is more representative of the training set, allowing L2D to provide consistent and accurate estimates.

7.3 Misclassification Cost Analysis

Finally, we analyze the misclassification cost for different deferral strategies under varying noise conditions. Table 1 compares five strategies: density-softmax (DS), conformal prediction (CP), L2D, rejection learning (RL), and random assignment. For the 5 settings shown, we fix the deferral rate at 40%, the number of experts at 5, and data availability at one-fifth (one expert prediction per instance). The deferral rate is set at 40% so that expert work capacity is enough to handle the noisy data (20%) along with in-distribution instances. Each setting is run 5 times, where in each run we select 5 experts out of a pool of 15 and use different random seeds for training data selection and noise introduc-

tion. Across all noise settings, the proposed DS and CP methods consistently outperform the baseline strategies. In particular, DS and CP exhibit lower misclassification costs in low-noise scenarios, while in medium to high-noise settings, CP emerges as the best-performing strategy. This is due to the fact that as the noise levels get higher, having a hard cutoff and just sending OOD samples to humans leads to better results than having a model score “fade” caused by the density model, where OOD instances may sometimes still be assigned to the classifier.

The baseline methods (L2D, RL, and random assignment) perform less effectively, with the random assignment approach consistently yielding the highest misclassification costs. Interestingly, in the extended experiments where we test the 500 scenarios described in Section 6.6, L2D outperforms RL around 94% of the time in lower noise settings (categorical, numerical, and low noise) with higher deferral rates (40% and 50%), while RL outperforms L2D in higher noise conditions (medium and high noise) around 78% of the time. These results highlight L2D’s effectiveness in scenarios where data drift is minimal and it can reliably model decision-maker correctness. In contrast, rejection learning proves advantageous when data that deviates from the training distribution is identifiable and can be deferred to human experts. This also justifies the conformal prediction method’s advantage in high noise settings.

In Figure 5 we analyze the misclassification cost per 100 instances as we vary the deferral rate and the number of experts. This analysis reveals three key points. First, misclassification cost decreases as more instances are deferred to human experts, which aligns with the better performance of experts compared to the classifier h in cases of distribution shift. Second, increasing the deferral rate amplifies the advantage of having more experts, as shown by the wider gaps between the lines at high deferral rates. At low deferral rates, the system primarily defers noisy and OOD instances to human experts, but once the deferral rate exceeds 20%—matching the proportion of noisy data—experts also handle in-distribution cases, making a larger expert team beneficial for instance assignment, as we can assign instances to the expert based on their predicted performance. Lastly, in the high-noise setting, the conformal prediction method shows a sharper reduction in misclassification cost at low deferral rates compared to the density-softmax approach. This difference arises from the way each method handles OOD instances: the conformal prediction approach defers all instances predicted as empty sets directly to human experts, while the density-softmax approach may still assign certain OOD instances to the clas-

Table 1: Comparison of deferral strategies under different noise settings. The reported values are the expected misclassification cost per 100 instances with 95% confidence intervals for the mean across five runs. For each noise setting, the best-performing strategy is in bold, and the second-best is underlined.

Setting	Deferral Strategy					
	Noise	CP	DS	L2D	RL	Random
numerical		<u>5.06</u> \pm 0.28	4.93 \pm 0.39	5.07 \pm 0.46	5.36 \pm 0.32	5.25 \pm 0.06
categoryal		<u>4.81</u> \pm 0.13	4.80 \pm 0.07	5.07 \pm 0.21	5.23 \pm 0.10	5.31 \pm 0.23
low		<u>5.18</u> \pm 0.13	5.17 \pm 0.10	5.55 \pm 0.27	5.39 \pm 0.22	5.70 \pm 0.19
medium		5.25 \pm 0.14	<u>5.41</u> \pm 0.17	5.84 \pm 0.15	5.71 \pm 0.22	6.04 \pm 0.18
high		4.83 \pm 0.21	<u>5.48</u> \pm 0.09	5.89 \pm 0.18	5.75 \pm 0.12	6.08 \pm 0.07

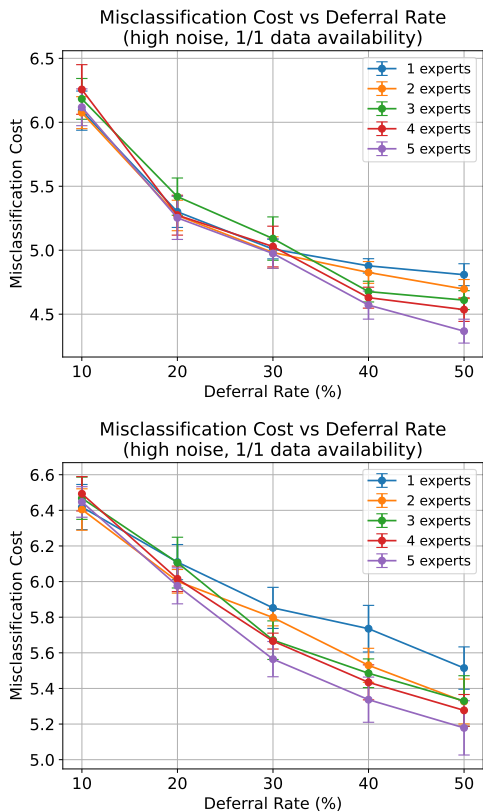


Figure 5: Misclassification Cost per 100 instances vs Deferral Rate. On top, we have the results for the conformal prediction approach, and below for the density-softmax approach. Error bars represent 95% confidence intervals for the mean across 5 runs. The results are shown for the high noise setting and the full data availability setting but are generalizable to other settings.

sifier if the optimization deems it advantageous.

8 Conclusions and Future Work

Machine learning (ML) models are increasingly used in high-stakes decisions due to their speed, scalability, and strong performance. However, these models often lack transparency and adaptability

in dynamic contexts, especially when data distributions shift over time. Human-AI collaboration (HAIC) addresses these limitations by combining the strengths of both ML models and human experts. The learning to defer (L2D) framework builds on this collaboration by modeling human decision-making and deferring uncertain cases to experts. Yet, L2D systems face key challenges in dynamic environments, where distribution shifts can reduce model reliability and affect the assignment process. Additionally, L2D must handle human work capacity constraints, limited data availability, and varying error costs. To address these limitations, we introduced two new HAIC methods designed to improve robustness in dynamic environments and enhance the applicability of these systems.

The first, a distance-aware L2D approach, combines classifiers with density functions to detect distribution shifts. This method showed improvements in misclassification cost and calibration, indicating that density-aware adjustments help the model to better detect instances that differ from the training data. The second approach employs density-based conformal prediction to balance rejection learning with L2D, identifying instances that do not conform to the training data distribution and deferring them directly to human experts. This strategy achieved the lowest misclassification cost in high-noise settings, effectively deferring challenging cases to human experts, while also improving calibration on instances where L2D is employed.

Future work could focus on refining the density-softmax approach by allowing the classifier h 's probability of correctness to approach zero under high uncertainty, aligning it more closely with conformal prediction. Additionally, exploring the system's performance under varying cost ratios for false positives and negatives would enhance its adaptability. Incorporating fairness incentives could also help mitigate biases in both human and ML decision-makers, promoting more equitable outcomes in high-stakes scenarios.

References

- [1] J. V. Alves, D. Leitão, S. Jesus, M. O. P. Sampaio, J. Liébana, P. Saleiro, M. A. T. Figueiredo, and P. Bizarro. Cost-sensitive learning to defer to multiple experts with workload constraints, 2024.
- [2] J. O. Awoyemi, A. O. Adetunmbi, and S. A. Oluwadare. Credit card fraud detection using machine learning techniques: A comparative analysis. In *2017 International Conference on Computing Networking and Informatics (IC-CNI)*, pages 1–9, 2017. doi: 10.1109/ICCNI.2017.8123782.
- [3] H. M. Bui and A. Liu. Density-softmax: Scalable and calibrated uncertainty estimation under distribution shifts, 2023.
- [4] M.-A. Charusaie, H. Mozannar, D. Sontag, and S. Samadi. Sample efficient learning of predictors that complement humans, 2022.
- [5] C. Chow. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, 16(1):41–46, 1970. doi: 10.1109/TIT.1970.1054406.
- [6] C. Cortes, G. DeSalvo, and M. Mohri. Learning with rejection. In R. Ortner, H. U. Simon, and S. Zilles, editors, *Algorithmic Learning Theory*, pages 67–82, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46379-7.
- [7] M. De-Arteaga, R. Fogliato, and A. Chouldechova. A case for humans-in-the-loop: Decisions in the presence of erroneous algorithmic scores. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI ’20. ACM, Apr. 2020. doi: 10.1145/3313831.3376638. URL <http://dx.doi.org/10.1145/3313831.3376638>.
- [8] D. Dellermann, P. Ebel, M. Söllner, and J. M. Leimeister. Hybrid intelligence. *Business amp; Information Systems Engineering*, 61(5): 637–643, Mar. 2019. ISSN 1867-0202. doi: 10.1007/s12599-019-00595-2. URL <http://dx.doi.org/10.1007/s12599-019-00595-2>.
- [9] C. Elkan. The foundations of cost-sensitive learning. *Proceedings of the Seventeenth International Conference on Artificial Intelligence: 4-10 August 2001; Seattle*, 1, 05 2001.
- [10] A. Esteva, B. Kuprel, R. Novoa, J. Ko, S. Swetter, H. Blau, and S. Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542, 01 2017. doi: 10.1038/nature21056.
- [11] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and H. Bouchachia. A survey on concept drift adaptation. *ACM Computing Surveys (CSUR)*, 46, 04 2014. doi: 10.1145/2523813.
- [12] Y. Geifman and R. El-Yaniv. Selective classification for deep neural networks, 2017.
- [13] S. Goel, R. Shroff, J. Skeem, and C. Slobogin. *The accuracy, equity, and jurisprudence of criminal risk assessment*. 05 2021. ISBN 9781788972819. doi: 10.4337/9781788972826.00007.
- [14] A. Gopnik and H. M. Wellman. Reconstructing constructivism: causal models, bayesian learning mechanisms, and the theory theory. *Psychological bulletin*, 138 6:1085–108, 2012. URL <https://api.semanticscholar.org/CorpusID:2496804>.
- [15] V. Gulshan, L. Peng, M. Coram, M. Stumpe, D. Wu, A. Narayanaswamy, S. Venugopalan, K. Widner, T. Madams, J. Cuadros, R. Kim, R. Raman, P. Nelson, J. Mega, and D. Webster. Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *JAMA*, 316, 11 2016. doi: 10.1001/jama.2016.17216.
- [16] Y. Hechtlinger, B. Póczos, and L. Wasserman. Cautious deep learning, 2019.
- [17] P. Hemmer, S. Schellhammer, M. Vössing, J. Jakubik, and G. Satzger. Forming effective human-ai teams: Building machine learning models that complement the capabilities of multiple experts, 2022.
- [18] P. Hemmer, L. Thede, M. Vössing, J. Jakubik, and N. Kühl. Learning to defer with limited expert predictions, 2023.
- [19] K. Hendrickx, L. Perini, D. V. der Plas, W. Meert, and J. Davis. Machine learning with a reject option: A survey, 2024.
- [20] D. Hendrycks and K. Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks, 2018.
- [21] E. Hüllermeier and W. Waegeman. Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *Machine Learning*, 110(3):457–506, Mar. 2021. ISSN 1573-0565. doi: 10.1007/s10994-021-05946-3. URL <http://dx.doi.org/10.1007/s10994-021-05946-3>.

- [22] S. Jesus, J. Pombal, D. Alves, A. Cruz, P. Saleiro, R. P. Ribeiro, J. Gama, and P. Bizarro. Turning the tables: Biased, imbalanced, dynamic tabular datasets for ml evaluation, 2022. URL <https://arxiv.org/abs/2211.13358>.
- [23] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu. Lightgbm: A highly efficient gradient boosting decision tree. In *Neural Information Processing Systems*, 2017. URL <https://api.semanticscholar.org/CorpusID:3815895>.
- [24] V. Keswani, M. Lease, and K. Kenthapadi. Towards unbiased and accurate deferral to multiple experts, 2021.
- [25] A. E. Khandani, A. J. Kim, and A. W. Lo. Consumer credit-risk models via machine-learning algorithms. *Journal of Banking Finance*, 34(11):2767–2787, 2010. ISSN 0378-4266. doi: <https://doi.org/10.1016/j.jbankfin.2010.06.001>. URL <https://www.sciencedirect.com/science/article/pii/S0378426610002372>.
- [26] D. Leitão, P. Saleiro, M. A. T. Figueiredo, and P. Bizarro. Human-ai collaboration in decision-making: Beyond learning to defer, 2022.
- [27] D. Madras, T. Pitassi, and R. Zemel. Predict responsibly: improving fairness and accuracy by learning to defer. *Advances in neural information processing systems*, 31, 2018.
- [28] S. Messoudi, S. Rousseau, and S. Destercke. *Deep Conformal Prediction for Robust Models*, pages 528–540. 06 2020. ISBN 978-3-030-50145-7. doi: 10.1007/978-3-030-50146-4_39.
- [29] H. Mozannar and D. Sontag. Consistent estimators for learning to defer to an expert, 2021.
- [30] A. Nguyen, J. Yosinski, and J. Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 427–436, 2015. doi: 10.1109/CVPR.2015.7298640.
- [31] V.-L. Nguyen, S. Destercke, and E. Hüllermeier. Epistemic uncertainty sampling, 2019.
- [32] J. C. Perdomo, T. Zrnic, C. Mendler-Dünner, and M. Hardt. Performative prediction, 2021.
- [33] W. Saeed and C. Omlin. Explainable ai (xai): A systematic meta-survey of current challenges and future opportunities. *Knowledge-Based Systems*, 263: 110273, 2023. ISSN 0950-7051. doi: <https://doi.org/10.1016/j.knosys.2023.110273>. URL <https://www.sciencedirect.com/science/article/pii/S0950705123000230>.
- [34] R. Senge, S. Bösner, K. Dembczyński, J. Haasenritter, O. Hirsch, N. Donner-Banzhoff, and E. Hüllermeier. Reliable classification: Learning classifiers that distinguish aleatoric and epistemic uncertainty. *Information Sciences*, 255: 16–29, 2014. ISSN 0020-0255. doi: <https://doi.org/10.1016/j.ins.2013.07.030>. URL <https://www.sciencedirect.com/science/article/pii/S0020025513005410>.
- [35] S. Somanchi, S. Adhikari, A. Lin, E. Eneva, and R. Ghani. Early prediction of cardiac arrest (code blue) using electronic medical records. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD ’15*, page 2119–2126, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450336642. doi: 10.1145/2783258.2788588. URL <https://doi.org/10.1145/2783258.2788588>.
- [36] D. Tailor, A. Patra, R. Verma, P. Manggala, and E. Nalisnick. Learning to defer to a population: A meta-learning approach, 2024.
- [37] R. Verma and E. Nalisnick. Calibrated learning to defer with one-vs-all classifiers, 2022.
- [38] R. Verma, D. Barrejón, and E. Nalisnick. Learning to defer to multiple experts: Consistent surrogate losses, confidence calibration, and conformal ensembles, 2023.
- [39] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni. Modeling tabular data using conditional gan, 2019. URL <https://arxiv.org/abs/1907.00503>.
- [40] B. Zadrozny, J. Langford, and N. Abe. Cost-sensitive learning by cost-proportionate example weighting. In *Third IEEE International Conference on Data Mining*, pages 435–442, 2003. doi: 10.1109/ICDM.2003.1250950.