Exploratory Machine Learning with Unknown Unknowns[*]

Peng Zhao, Yu-Jie Zhang, Zhi-Hua Zhou
National Key Laboratory for Novel Software Technology,
Nanjing University, Nanjing 210023, China
{zhaop, zhangyj, zhouzh}@lamda.nju.edu.cn

Abstract

In conventional supervised learning, a training dataset is given with ground-truth labels from a known label set, and the learned model will classify unseen instances to known labels. In real situations, when the learned models do not work well, learners generally attribute the model failure to the inadequate selection of learning algorithms or the lack of enough labeled training samples. In this paper, we point out that there is an important category of failure, which owes to the fact that there are *unknown* classes in the training data misperceived as other labels, and thus their existence is *unknown* from the given supervision. Such problems of unknown unknown classes can hardly be addressed by common re-selection of algorithms or accumulation of training samples. For this purpose, we propose the *exploratory machine learning*, where in this paradigm once learner encounters unsatisfactory learning performance, she can examine the possibility and, if unknown unknowns really exist, deploy the optimal strategy of feature space augmentation to make unknown classes observable and be enabled for learning. Theoretical analysis and empirical study on both synthetic and real datasets validate the efficacy of our proposal.

1 Introduction

Machine learning has achieved great success in many real-world applications. The success heavily relies on the suitable learning algorithm and sufficient supervised training data. Therefore, facing model failure, the learner would always doubt the inadequate selection of algorithms and the lack of data. A common practice is to try other algorithms or accumulate more data, and such an approach could work effectively when there are no other factors leading to the failure. In this paper, however, we point out that there is an important cause of model failure always ignored before: *unknown unknowns* hidden in the training dataset.

Specifically, we attribute the unknown unknowns to the fact that some training instances of certain *unknown* classes are wrongly perceived as others, and thus appear *unknown* to the learned model with the given supervision. This is always the case when the label space is misspecified due to the insufficient feature information. Consider the task of medical diagnosis, where we need to train a model for community healthcare centers based on their patient records, to help diagnose the cause of a patient with cough and dyspnea. As shown in Figure 1, there are actually three causes: two common ones (*asthma* and *pneumonia*), as well as an unusual one (*lung cancer*) whose diagnosis crucially relies on the computerized tomography (CT) scan device, yet too expensive to purchase. Thus, the community healthcare centers are not likely to diagnose patients with dyspnea as cancer, resulting in that the class of "lung cancer" becomes invisible and hidden in the collected training dataset. As a result, the learned model will be unaware of this unobserved class.

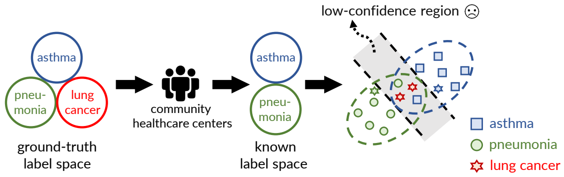


Figure 1: Unknown unknowns in medical diagnosis tasks. Patients with *lung cancer* are misdiagnosed as *asthma* or *pneumonia* due to the lack of CT scan devices, and thus appear as unknown to learned model. So "lung cancer" class becomes invisible in training data.

Similar phenomena occur in many other applications. For instance, the trace of a new-type aircraft was mislabeled as old-type ones until performance of aircraft detectors is found poor (i.e., capability of collected signals is inadequate), and the officer suspects that there are new-type aircrafts unknown previously. When feature information is insufficient, there is a high risk to misperceive some classes of training data as others, leading to existence of hidden classes. Especially, hidden classes are sometimes of more interest, like in above two cases. Thus, it is crucial to discover hidden unknown classes and classify known classes well simultaneously.

Conventional supervised learning (SL) cannot obtain a satisfied model when such *unknown unknowns* emerge in the training dataset, even if we could accumulate more data and re-select algorithms exhaustively. The reason lies in that the unknown factors are beyond the expressivity of training data. We thus require new ideas to tackle such unknown unknowns.

2 ExML: A New Learning Framework

The problem we are concerned with is essentially a class of *unknown unknowns*. In fact, how to deal with unknown

[*] This research was supported by NSFC (61921006).

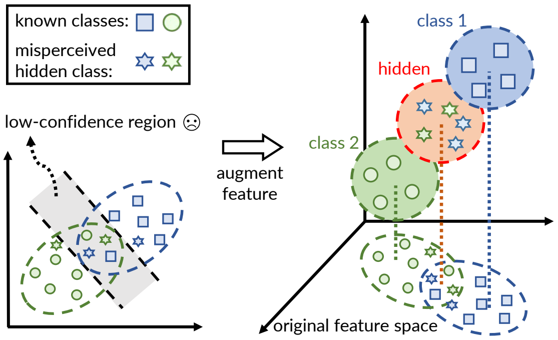


Figure 2: An example illustrates that an informative feature can substantially improve separability of low-confidence samples and make the hidden class distinguishable.

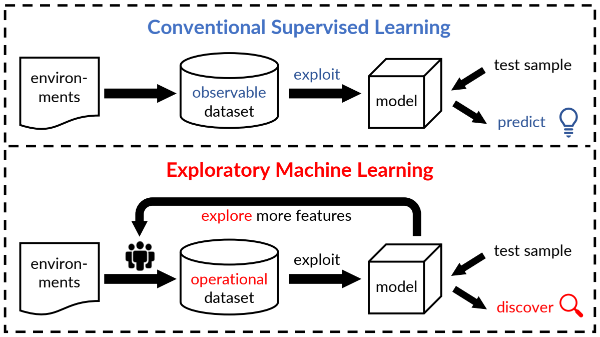


Figure 3: Comparison of two frameworks. SL exploits observable dataset for prediction. ExML explores more features based on operational dataset for discovery of hidden classes.

unknowns is the fundamental question of robust artificial intelligence (Dietterich 2017), and many studies have been devoted to addressing various aspects including changing distributions (Pan and Yang 2010; Gama et al. 2014), evolvable features (Hou, Zhang, and Zhou 2017; Hou and Zhou 2018), open categories (Scheirer et al. 2013; Geng, Huang, and Chen 2018), etc. Different from them, we study a new problem setting ignored previously, that is, the training dataset is badly advised by the *incompletely perceived label space* due to the *insufficient feature information*. This problem turns out to be quite challenging, since feature space and label space are entangled and *both* of them are unreliable.

The first challenge is that when the learning performance is undesired we do not know whether the issue is caused by the hidden unknown classes or not. To tackle that, we may accumulate more training data and re-select the learning algorithms. If the model failure persists, we would suspect the existence of unknowns. The second challenge is how to recognize the hidden unknown classes. Notably, it is infeasible to merely pick out instances with low predictive confidence as hidden classes, since we can hardly distinguish (i) instances from hidden classes that suffer from low-confidence predictions owing to the incomplete label space; (ii) instances from known classes that suffer from low-confidence predictions because of insufficient feature information. This characteristic reflects intrinsic hardness of learning with unknown unknowns due to feature deficiency, and thus it is necessary to ask for external feature information.

2.1 Exploratory Machine Learning

To handle unknown unknowns caused by feature deficiency, we resort to the human in the learning loop to interact with environments for enhancing the data collection, more specifically, actively augmenting the feature space. The idea is that when a learned model remains performing poorly even fed with much more data, learner will suspect existence of hidden classes and subsequently seek several candidate features to augment. Figure 2 shows a straightforward example that learner receives a dataset and observes that there are two classes with poor separability, resulting in a noticeable low-confidence region. After a proper feature augmentation, learner will then realize that there exists an additional class hidden in training data previously due to feature deficiency.

Enlightened by the above example, we introduce a new learning framework called *exploratory machine learning* (ExML), which explores more feature information to deal with unknown unknowns caused by feature deficiency. The terminology of exploratory learning is originally raised in the area of education, defined as an approach to teaching and learning that encourages learners to examine and investigate new material with the purpose of discovering relationships between existing background knowledge and unfamiliar content and concepts (Njoo and De Jong 1993; Spector et al. 2014). In the context of machine learning, our proposed framework encourages learners to *examine and investigate the training dataset via exploring new feature information, with the purpose of classifying known classes and discovering potentially hidden classes*. Figure 3 compares the proposed ExML to conventional supervised learning (SL). Conventional SL views the training dataset as an observable representation of environments and exploits it to train a model to predict the label. By contrast, ExML considers the training dataset is *operational*, where learners can examine and investigate the dataset by *exploring* more feature information, and thereby *discover* unknown unknowns due to feature deficiency.

We develop an approach to implement ExML, consisting of three ingredients: rejection model, feature exploration, and model cascade. The rejection model identifies suspicious instances that potentially belong to hidden classes. Feature exploration guides which feature should be explored, and then retrains the model on the augmented feature space. Model cascade allows a layer-by-layer processing to refine the selection of suspicious instances. Theoretical analysis is provided to justify the superiority of the proposed framework. We present empirical evaluations on synthetic data to illustrate the idea and further validate the effectiveness on real datasets.

2.2 Problem Formulation

Training Dataset. The learner receives a training dataset $\widehat{D}_{tr} = \{(\widehat{\mathbf{x}}_i, \widehat{\mathbf{y}}_i)\}_{i=1}^{m}$, where $\widehat{\mathbf{x}}_i \in \widehat{\mathcal{X}} \subseteq \mathbb{R}^d$ is from the *observed* feature space, and $\widehat{\mathbf{y}}_i \in \widehat{\mathcal{Y}}$ is from the *incomplete* label space with N known classes. We consider the binary case for simplicity. Note that there exist training samples that are actually from hidden classes yet wrongly labeled as known classes due to feature deficiency.

Candidate Features and Cost Budget. Besides the training dataset, the learner can access a set of candidate features $\mathcal{C} = \{c_1, \ldots, c_K\}$, whose values are unknown before acquisition. For the example of medical diagnosis (Figure 1), a feature refers to signals returned from CT scan devices, only available after patients have taken the examination. Moreover, a certain cost will be incurred to acquire any candidate feature for any sample. The learner aims to identify top k informative features from the pool under a given budget B. For convenience, the cost of each acquisition is set as 1 uniformly and the learner desires to find the best feature, i.e., $k = 1$.

Testing Stage. Suppose the learner identifies the best feature as c_i, he/she will then augment the testing sample with this feature, leading to the augmented feature space $\mathcal{X}_i = (\widehat{\mathcal{X}} \cup \mathcal{X}^i) \subseteq \mathbb{R}^{d+1}$ where $\mathcal{X}^i$ is the feature space of c_i. The learned model requires to predict the label of the augmented testing sample, either classified to one of known classes or discovered as the hidden classes (abbrev. hc).

We finally note that several assumptions are introduced for simplicity, with the aim of avoiding distractions of an over-complicated setting and better understanding the essence of the new problem. Actually, our proposal still works when relaxing these assumptions by borrowing well-known techniques such as multi-class rejection (Zhang, Wang, and Qiao 2018), learning with non-uniform cost (Seldin et al. 2014). We emphasize that above aspects are not the current focus. These extensions will be considered as future works.

3 A Practical Approach

Due to the feature deficiency, the learner might be even unaware of the existence of hidden classes based on the observed training data. It is thus necessary to introduce the assumption that *instances with high predictive confidence are safe, i.e., they will be correctly predicted as one of known classes*. Learner will suspect the existence of hidden classes when the learned model performs badly.

We justify the necessity of above assumption. Actually, there are some previous works studying the problem of high-confidence false predictions without considering the issue of feature deficiency (Attenberg, Ipeirotis, and Provost 2015; Lakkaraju et al. 2017), in which there exist some instances wrongly predicted with high confidence. Since the model's performance is highly unreliable, to rectify that, they assume the existence of an oracle providing ground-truth labels for the given query. However, in present of feature deficiency as in our scenario, the problem would not be tractable unless there is an oracle able to provide ground-truth labels based on the insufficient feature information, which turns out to be an even stronger assumption that does not hold in reality generally. We leave high-confidence unknown unknowns due to the insufficient feature as future work to explore.

On the other hand, we emphasize that the introduced assumption does not trivialize the problem because low-predictive instances are not necessarily from hidden classes (as explained at the beginning of Section 2), which necessities more efforts. Following the methodology of ExML (examining the training dataset via exploring new feature information), we design a novel approach, which consists of three components: rejection model, feature exploration, and model cascade. Figure 4 illustrates main procedures, and we will describe details of each component subsequently.

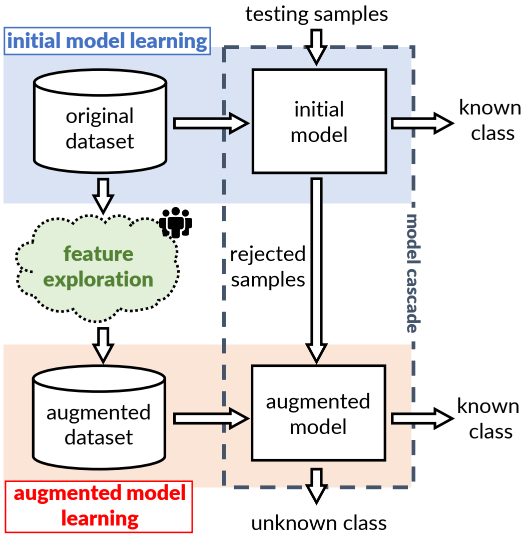


Figure 4: Overall procedure of ExML. Our approach begins with an initial model (blue part), followed by exploring the best candidate feature (green part). Afterwards, a model is retrained based on the augmented dataset, and finally cascaded with the initial model to discover the hidden class (red part).

3.1 Rejection Model

As shown in Figure 4, the learner starts from training an initial model on the original dataset, with capability of identifying low-confidence instances. As emphasized previously (cf. the beginning of Section 2), these low-confidence instances could come from either known or hidden classes, so they are only detected as suspicious and will be further refined.

We realize this goal by the learning with rejection technique (Cortes, DeSalvo, and Mohri 2016b), where the learned model will abstain from predicting instances whose maximum conditional probability lower than a given value $1 - \theta$. More precisely, we learn a function pair $f = (h, g)$, where $h : \widehat{\mathcal{X}} \mapsto \mathbb{R}$ is the *predictive* function for known classes and $g : \widehat{\mathcal{X}} \mapsto \mathbb{R}$ is the gate function to *reject* the hidden class. The sample $\widehat{\mathbf{x}}$ is identified to the hidden class if $g(\widehat{\mathbf{x}}) < 0$, and otherwise to the class of $\texttt{sign}(h(\widehat{\mathbf{x}}))$. Such rejection models can be trained via optimizing the following objective:

$$\min_f \ \mathbb{E}_{(\widehat{\mathbf{x}},\widehat{y})\sim\widehat{\mathcal{D}}}[\ell_{0/1}(f, \widehat{\mathbf{x}}, \widehat{\mathbf{y}}; \theta)], \tag{1}$$

where $\ell_{0/1}(f, \widehat{\mathbf{x}}, \widehat{\mathbf{y}}; \theta) = \mathbb{1}_{\widehat{y}\cdot h(\widehat{\mathbf{x}})<0} \cdot \mathbb{1}_{g(\widehat{\mathbf{x}})>0} + \theta \cdot \mathbb{1}_{g(\widehat{\mathbf{x}})\leq 0}$ is the 0-1 loss of the rejection model f parameterized by the threshold $\theta \in (0, 0.5)$ and $\widehat{\mathcal{D}}$ is the data distribution over $\widehat{\mathcal{X}} \times \widehat{\mathcal{Y}}$. A smaller θ will lead to more rejections but a higher predictive accuracy on known classes. To tackle

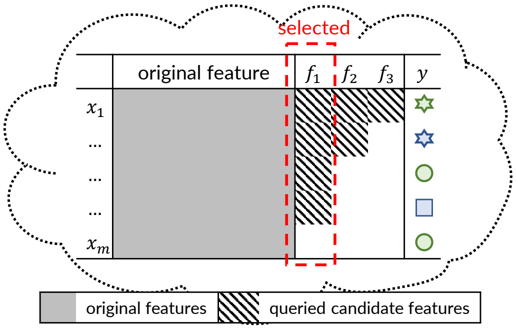


Figure 5: Feature Exploration

the difficulty of non-convex optimization arising from the indicator function, Cortes, DeSalvo, and Mohri (2016b) introduce the following calibrated surrogate loss function $\ell_{surr} := \ell_{surr}(f, \widehat{\mathbf{x}}, \widehat{y}; \theta)$ defined as

$$\ell_{surr} = \max\left\{1 + \frac{1}{2}\big(g(\widehat{\mathbf{x}}) - \widehat{y} \cdot h(\widehat{\mathbf{x}})\big), \theta\big(1 - \frac{g(\widehat{\mathbf{x}})}{1-2\theta}\big), 0\right\}$$

to approximate the original $\ell_{0/1}$ loss. Since the distribution is unknown we cannot directly measure the risk, we choose the model that minimizes the empirical risk:

$$\min_{f \in \mathbb{H}\times\mathbb{H}} \frac{1}{m}\sum_{i=1}^{m} \ell_{surr}(f, \widehat{\mathbf{x}}_i, \widehat{y}_i; \theta) + C_h\|h\|_{\mathbb{H}}^2 + C_g\|g\|_{\mathbb{H}}^2, \quad (2)$$

where C_h and C_g are regularization parameters, and $\mathbb{H}$ is the RKHS induced by kernel $K : \widehat{\mathcal{X}} \times \widehat{\mathcal{X}} \mapsto \mathbb{R}$. By the representer theorem (Schölkopf and Smola 2002), the optimizer of (2) is in the form of $h(\widehat{\mathbf{x}}) = \sum_{i=1}^{m} u_i K(\widehat{\mathbf{x}}, \widehat{\mathbf{x}}_i)$ and $g(\widehat{\mathbf{x}}) = \sum_{i=1}^{m} w_i K(\widehat{\mathbf{x}}, \widehat{\mathbf{x}}_i)$, where u_i and w_i are coefficients to learn. So (2) can be reformulated as quadratic programming and solved efficiently. For more details we refer the reader to the seminal work of Cortes, DeSalvo, and Mohri (2016b).

3.2 Feature Exploration

If the initial model is unqualified (for instance, it rejects too many samples for achieving desired accuracy), the learner will suspect the existence of hidden classes and explore new features to augment. In our setting, the learner requires to select the best feature from K candidates and retrain a model based on the augmented data, as shown in Figure 5.

We emphasize that the conventional feature selection is not feasible here, because it requires to know values of candidate features, while these values are unknown before acquisitions. To address the challenge, we propose a novel procedure—*feature exploration*—to adaptively identify the most informative feature under the cost budget, *without* requiring feature values in advance. To address the issue, there are two fundamental questions to answer:

(1) how to measure the quality of candidate features?

(2) how to allocate the budget to identify the best feature?

We answer the above two questions in the following.

Feature quality measure. Denote by $\mathcal{D}_i$ the data distribution over $\mathcal{X}_i \times \widehat{\mathcal{Y}}$, where $\mathcal{X}_i$ is the augmented feature space of the i-th candidate feature. We use the *Bayes risk* on $\mathcal{D}_i$ as feature quality measure:

$$R_i^* = R_i(f_i^*) = \min_f \; \mathbb{E}_{(\mathbf{x},\widehat{\mathbf{y}})\sim\mathcal{D}_i}\big[\ell_{0/1}(f, \mathbf{x}, \widehat{\mathbf{y}}; \theta)\big], \quad (3)$$

where $R_i(f)$ is the expected $0/1$ risk of function f over $\mathcal{D}_i$, and f_i^* minimizes $R_i(f)$ over all measurable functions. The Bayes risk essentially reflects the minimal error that any rejection model can attain on the augmented data distribution, whose value will be smaller when the selected augmented feature improves the separability more significantly (and is believed more informative).

Due to the inaccessibility of the underlying distribution $\mathcal{D}_i$, we approximate the Bayes risk by its empirical version over the augmented data $D_i = \{(\mathbf{x}_j, \widehat{\mathbf{y}}_j)\}_{j=1}^{n_i} \sim \mathcal{D}_i$,

$$\widehat{R}_{D_i} = \widehat{R}_i(\widehat{f}_i) = \sum\nolimits_{j=1}^{n_i} \ell_{0/1}(\widehat{f}_i, \mathbf{x}_j, \widehat{y}_j; \theta), \quad (4)$$

where $\mathbf{x}_j \in \mathcal{X}_i, \widehat{\mathbf{y}}_j \in \widehat{\mathcal{Y}}$, and $\widehat{f}_i$ is the rejection model learned by empirical risk minimization over surrogate loss (2) on augmented dataset D_i. Based on the feature quality measure (3) and its empirical version (4), we now introduce the budget allocation strategy to identify the best candidate feature.

Budget allocation strategy. Without loss of generality, suppose features are sorted according to their quality, i.e., $R_1^* \leq \cdots \leq R_K^*$. Our goal is to identify the best feature within the limited budget, and meanwhile the model retrained on augmented data should have good generalization ability.

We first propose the uniform allocation strategy as follows, under the guidance of criterion (3).

Uniform Allocation For each candidate feature c_i, $i \in [K]$, learner allocates $\lfloor B/K \rfloor$ budget and obtains an augmented dataset D_i. So we can compute the empirical feature measure by (4), and select the feature with the smallest risk. The above strategy is simple yet effective, which can provably identify the best feature with high probability (Theorem 1).

Median Elimination We further propose another variant inspired by the bandit theory to improve the budget allocation efficiency. Specifically, we adopt the technique of *median elimination* (ME) (Even-Dar, Mannor, and Mansour 2006), which removes one half of poor candidate features after every iteration and only the best one remains in the end. As a result, the algorithm can avoid allocating too many budgets on poor features. More specifically, the elimination proceeds in $T = \lceil \log_2 K \rceil$ episodes, in each episode, $\lfloor B/T \rfloor$ budget is allocated uniformly to all remaining candidate features, and the learner could query their values for updating the corresponding augmented datasets D_i. Then, the score $\widehat{R}_{D_i}$ is calculated on the current augmented datasets D_i and the half features with high $\widehat{R}_{D_i}$ are eliminated. In the last, only one candidate feature i_s will be left and its augmented dataset D_{i_s} contains around $\lfloor B/\log K \rfloor$ samples, which is the largest one among all the candidate features. Algorithm details are presented in Appendix B.

As shown in Figure 5, poor features are eliminated earlier, budget left for the selected feature is thus improved from $\lfloor B/K \rfloor$ to $\lfloor B/\log K \rfloor$ by ME, which ensures better generalization ability of the learned model. Meanwhile, median elimination can explore the best candidate feature more efficiently than uniform allocation, as shown in the bandit theory (Even-Dar, Mannor, and Mansour 2006). We finally remark that our paper currently focuses on the best feature, and the framework is ready for identifying the top k features ($k > 1$) by introducing more sophisticated techniques (Kalyanakrishnan et al. 2012; Chen, Li, and Qiao 2017).

3.3 Model Cascade

After feature exploration, learner will retrain a model on augmented data. Considering that the augmented model might not always be better than the initial model, particularly when the budget is not enough or candidate features are not quite informative, we propose the *model cascade* mechanism to cascade the augmented model with the initial one. Concretely, high-confidence predictions are accepted in the initial model, the rest suspicious are passed to the next layer for feature exploration, those augmented samples with high confidence will be accepted by the augmented model, and the remaining suspicious continue to the next layer for further refinements.

Essentially, our approach can be regarded as a *layer-by-layer processing for identifying instances of hidden classes*, and the procedures can be stopped until human discovers remaining suspicious are indeed with certain hidden structures. For simplicity, we only implement a two-layer architecture.

4 Theoretical Analysis

This section presents theoretical results. We first investigate the attainable excess risk of supervised learning, supposing that the best feature were *known* in advance. Then, we provide the result of ExML to demonstrate the effectiveness of our proposed criterion and budget allocation strategies.

For each candidate feature c_i, we denote the corresponding hypothesis space as $\mathcal{H}_i, \mathcal{G}_i = \{\mathbf{x} \mapsto \langle \mathbf{w}, \Phi_i(\mathbf{x}) \rangle \mid \|\mathbf{w}\|_{\mathbb{H}_i} \leq \Lambda_i\}$, where Φ_i and $\mathbb{H}_i$ are induced feature mapping and RKHS of kernel K_i in the augmented feature space.

Supervised learning with known best feature. Suppose the best feature were known in advance, we could obtain B samples augmented with this particular feature. Let f_{SL} be the model learned by supervised learning via minimizing (2). From learning theory literatures (Bousquet, Boucheron, and Lugosi 2003; Cortes, DeSalvo, and Mohri 2016b), for any $\delta > 0$, with probability at least $1 - \delta$, we have

$$R_1(f_{\mathrm{SL}}) - R_1^* \leq \mathcal{O}\left(\sqrt{\frac{(\kappa_1 \Lambda_1)^2}{B}} + \sqrt{\frac{\log(1/\delta)}{2B}}\right) + R_{ap}, \tag{5}$$

where $R_{ap} = C_\theta \left(\inf_{f \in \mathcal{H}_1 \times \mathcal{G}_1} R_1^{surr}(f) - \inf_f R_1^{surr}(f)\right)$ is the approximation error measuring how well hypothesis spaces $\mathcal{H}_1, \mathcal{G}_1$ approach the target, in terms of the expected surrogate risk $R_1^{surr}(f) = \mathbb{E}_{(\mathbf{x}, \widehat{\mathbf{y}}) \sim \mathcal{D}_1}[\ell_{surr}(f, \mathbf{x}, \widehat{\mathbf{y}}; \theta)]$. The constant factor is $C_\theta = 1/((1-\theta) \cdot (1 - 2\theta))$.

The above result theoretically reveals that if the best feature were *known* in advance, the excess risk of supervised learning would converge to the inevitable approximate error in the rate of $\mathcal{O}(1/\sqrt{B})$, with a given feature budget B.

Exploratory learning with unknown best feature. In our setting, the best feature is unfortunately unknown ahead of time. More importantly, since values of K candidate features are unavailable, it is *infeasible* to perform the feature selection. We show that by means of ExML (feature exploration), the excess risk also converges, in the rate of $\mathcal{O}(\sqrt{K/B})$, yet *without* requiring to know the best feature.

Theorem 1. *Let c_{i_s} be the identified feature and $\widehat{f}_{i_s}$ be the augmented model returned by* ExML *with uniform allocation. Then, with probability at least $1 - \delta$, we have*

$$R_{i_s}(\widehat{f}_{i_s}) - R_1^* \leq \mathcal{O}\left(\sqrt{\frac{(\kappa\Lambda)^2}{\lfloor B/K \rfloor}} + \sqrt{\frac{\log(3/\delta)}{2\lfloor B/K \rfloor}}\right) + R_{ap}, \tag{6}$$

where $\Lambda = \max_{i \in [K]} \Lambda_i$, $\kappa = \max_{i \in [K]} \sup_{\mathbf{x} \in \mathcal{X}_i} K_i(\mathbf{x}, \mathbf{x})$.

Remark. Comparing the excess risk bounds of (5) and (6), we can observe that ExML exhibits a similar convergence tendency to SL with *known* best feature, yet *without* requiring to know the best feature. An extra $\sqrt{K}$ times factor is paid for exploration of the best feature. We note that under certain mild technical assumptions, the dependence can be further reduced to $\sqrt{\log K}$ by median elimination (Even-Dar, Mannor, and Mansour 2006), as poor candidate features have been removed in the earlier episodes.

5 Experiments

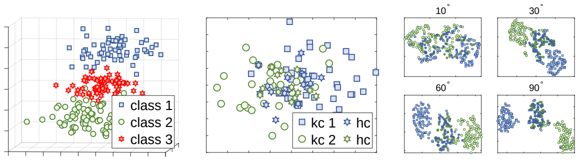
In this section, we conduct experiments to examine empirical performance of the proposed exploratory machine learning (ExML). Specifically, we provide evaluations on synthetic data for visualizing the superiority of ExML to conventional supervised learning in handling unknown unknowns. Then, we report results on real-world datasets to demonstrate the effectiveness of the overall method, as well as the usefulness of feature exploration and model cascade modules. In all experiments, we denote by $B = b \cdot mk$ the feature exploration budget, where m is number of training samples, K is number of candidate features, $b \in [0, 1]$ is the budget ratio.

5.1 Synthetic Data for Illustration

We first illustrate the advantage of exploratory machine learning over conventional supervised learning in discovering the hidden classes on the synthetic data.

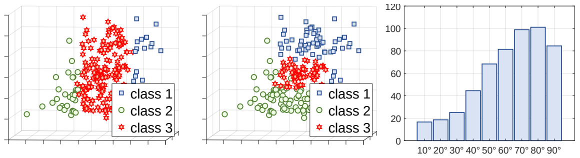
Setting. Following the illustrative example in Figure 1, we generate a 3-dim dataset containing 3 classes, whose ground-truth distribution is shown as Figure 6(a). However, as shown in Figures 6(b), only the first two dimensions are observable in the training stage, resulting in a hidden class (hc) located in the intersection area of known classes (kc1 and kc2).

Specifically, we generate instances of each class from Gaussian distributions. Means and variances are $[-a, 0, -z]$ and $\sigma \cdot \mathbf{I}_{3\times3}$ for the first known class, $[a, 0, z]$ and $\sigma \cdot \mathbf{I}_{3\times3}$ for the second known class as well as $[0, 0, 0]$ and $\sigma/2 \cdot \mathbf{I}_{3\times3}$ for the hidden class, where $\mathbf{I}_{3\times3}$ is a 3×3 identity matrix. We set $\sigma = 3a$ and $z = 5a$. In the training stage, the third-dim is unobservable and the hidden class (hc) is randomly labeled as another two. Each class contains 100 instances in the training data. Besides, we generate 9 candidate features in various qualities, whose angle to the horizon varies from 10° to 90°, the larger the better. Figure 6(c) plots the augmented feature space via t-SNE. The budget ratio is $b = 20\%$. In the testing



(a) ground-truth (b) training data (c) candidates

Figure 6: Visualization of synthetic data: (a) ground-truth distribution; (b) training data (only first two dimensions are observable); (c) t-SNE of candidate features with various qualities (a larger angle implies a better feature quality).



(a) SL (b) ExML (c) allocation

Figure 7: Visualization of results: (a) SL; b) ExML; (c) budget allocation of ExML with median elimination.

stage, the learner requires to predict on the 3-dim data, where the third dimension is the selected candidate features.

Contenders. We compare ExML to SL (with rejection model). For all rejection models, we employ the Gaussian kernel $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|_2^2/\gamma)$ with bandwidth $\gamma = \text{median}_{\mathbf{x}_i,\mathbf{x}_j \in D}(\|\mathbf{x}_i - \mathbf{x}_j\|_2^2)$ and set C_h, C_g to 1.

- **SL** trains the rejection model (Cortes, DeSalvo, and Mohri 2016b) with the given dataset on the original feature space, following the paradigm of conventional supervised learning. The threshold θ is choose as one achieving best accuracy on the testing data from the pool $[0.1, 0.2, 0.3, 0.4]$.
- **ExML** is our proposal with cascade models and using median elimination for feature exploration. The threshold for the initial rejection model is selected by cross validation to ensure 95% accuracy on high-confidence samples. The threshold θ for the augmented rejection model is choose as one achieving best accuracy on the testing data from the pool $[0.1, 0.2, 0.3, 0.4]$. The budget ratio is 20%.

Results. We first conduct SL to train a rejection model based on the 2-dim training data, and then perform ExML to actively augment the feature within the budget to discover unknown unknowns. Figures 7(a) and 7(b) plot the results, demonstrating a substantial advantage of ExML over SL in discovering the hidden class and predicting known classes. Furthermore, Figure 7(c) reports budget allocation of each candidate feature over 50 times repetition. We can see that the allocation clearly concentrates to more informative features (with larger angles), which validates the effectiveness of median elimination for the best feature exploration.

5.2 Benchmark Data for Evaluation

Dataset and Setting. We further evaluate on a UCI benchmark dataset *Mfeat* (van Breukelen et al. 1998), which is a multi-view dataset[1] containing 2000 samples and 6 views of features extracted by various methods. Their semantic information and statistics are:

- Fac: profile correlations, 216-dim;
- Pix: pixel averages in 2×3 windows, 240-dim;
- Kar: Karhunen-Love coefficients, 64-dim;
- Zer: Zernike moments, 47-dim;
- Fou: Fourier coefficients of the character shapes, 76-dim;
- Zer: morphological features, 6-dim.

The domain knowledge sorts the features by their quality as: Fac $>$ Pix $>$ Kar $>$ Zer $>$ Fou $>$ Mor, in a descending order.

In the training stage, we randomly sample 600 instances to form the labeled training data. This procedure repeats 10 times to generate different configurations. Since Mfeat is a multi-class dataset, we randomly sample 5 configurations to convert it into the binary classification task, where each known class and hidden class contain three original classes, and the instances from the hidden class are randomly mislabeled as one of known classes. There are in total 50 random configurations for training. As for the candidate features, each one of six views (features) is taken as original feature and the rest are prepared in the candidate set. Before training, we normalize all the features to the range $[0, 1]$. We evaluate all contenders on the testing data containing 1400 instances.

Contenders. Apart from SL, we include two ExML variants: $\text{ExML}_{\text{csd}}^{\text{UA}}$ and $\text{ExML}_{\text{aug}}^{\text{ME}}$ for ablation studies. Here aug/csd denotes the final model is only the augmented or cascaded with the initial model; UA/ME refers to feature exploration by uniform allocation or median elimination.

- $\mathbf{ExML}_{\mathbf{csd}}^{\mathbf{UA}}$ is our proposal with cascade model and using *uniform allocation* for feature exploration.
- $\mathbf{ExML}_{\mathbf{aug}}^{\mathbf{ME}}$ is our proposal *without* cascade model and using median elimination for feature exploration.

All ExML-type methods use the same parameters. SL and ExML are configured by the same setting as those in synthetic experiments. The budget ratio b varies from 10% to 30%.

Measure. We measure the performance of all the methods by the classification. Additionally, we introduce the *recall* to measure the effectiveness of feature exploration, defined as the ratio of the number of cases when identified feature is one of its top 2 features to the total number.

- **Accuracy**: the mean and standard deviation of the predictive accuracy on testing dataset over 50 configurations, where the true label of hidden classes are observable.
- **Recall**: the ratio of the number of cases when identified feature is one of top 2 features to the total number, where features quality is measured by the accuracy of augmented model trained on whole data with this particular feature.

Results. Table 1 reports mean and std of the predictive accuracy, and all features are sorted in descending order by their quality. We first compare SL to (variants of) ExML. When the original features are in high quality (Kar, Pix, Fac), SL could achieve favorable performance and there is no need to explore new features. However, in the case where uninformative original features are provided, which is of more interest for ExML, SL degenerates severely and $\text{ExML}_{\text{aug}}^{\text{ME}}$ (the single

[1] http://archive.ics.uci.edu/ml/datasets/Multiple+Features

Table 1: Evaluation on *Mfeat* dataset. Features are sorted by descending qualities. Bold font indicates algorithms significantly outperforms than others (paired t-test at 95% significance level).

Feature	Budget	SL	ExML$^{\text{ME}}_{\text{aug}}$	ExML$^{\text{UA}}_{\text{csd}}$	ExML	Recall
Fac	10%	**93.39 ± 1.66**	71.80 ± 9.55	92.39 ± 2.79	92.40 ± 2.78	48%
	20%	**93.39 ± 1.66**	82.26 ± 7.52	91.95 ± 3.32	92.00 ± 3.27	46%
	30%	**93.39 ± 1.66**	89.29 ± 4.72	92.20 ± 3.33	92.50 ± 2.86	44%
Pix	10%	**92.19 ± 2.47**	70.53 ± 8.27	90.54 ± 6.27	90.55 ± 6.31	58%
	20%	**92.19 ± 2.47**	81.70 ± 7.16	90.84 ± 6.17	90.87 ± 6.09	54%
	30%	**92.19 ± 2.47**	88.67 ± 4.14	90.45 ± 5.74	**91.82 ± 4.26**	68%
Kar	10%	**86.87 ± 3.43**	70.25 ± 10.2	85.55 ± 4.94	85.90 ± 4.85	56%
	20%	**86.87 ± 3.43**	81.46 ± 6.88	85.21 ± 5.46	**86.49 ± 4.81**	54%
	30%	86.87 ± 3.43	86.01 ± 5.41	86.52 ± 4.71	**88.18 ± 3.57**	56%
Zer	10%	73.82 ± 8.82	69.61 ± 10.7	72.96 ± 10.4	**76.17 ± 8.52**	82%
	20%	73.82 ± 8.82	**80.86 ± 8.02**	77.31 ± 7.89	**81.72 ± 7.33**	82%
	30%	73.82 ± 8.82	**86.07 ± 5.51**	81.11 ± 6.79	**86.33 ± 5.04**	86%
Fou	10%	68.73 ± 9.07	69.42 ± 9.68	68.88 ± 11.8	**75.92 ± 8.81**	82%
	20%	68.73 ± 9.07	82.11 ± 6.48	77.93 ± 8.27	**85.03 ± 4.39**	88%
	30%	68.73 ± 9.07	**89.90 ± 3.69**	82.45 ± 5.20	**89.35 ± 3.89**	92%
Mor	10%	57.47 ± 15.3	69.09 ± 11.3	66.58 ± 13.5	**71.07 ± 11.1**	80%
	20%	57.47 ± 15.3	**79.60 ± 10.1**	73.61 ± 8.86	**79.74 ± 9.92**	84%
	30%	57.47 ± 15.2	**87.44 ± 7.34**	78.31 ± 9.00	**86.98 ± 7.07**	90%

ExML model without model cascade) achieves better performance even with the limited budget. Besides, from the last column, we can see that informative candidates (top 2) are selected to strengthen the poor original features, which validates the effectiveness of the proposed budget allocation strategy (namely, the median elimination mechanism).

Since the ExML$^{\text{ME}}_{\text{aug}}$ is not guaranteed to outperform SL, particularly with the limited budget on poor candidate features, we propose the cascade structure. Actually, ExML approach (aka, ExML$^{\text{ME}}_{\text{csd}}$) achieves roughly *best-of-two-worlds* performance, in the sense that it is basically no worse or even better than the best of SL and ExML$^{\text{ME}}_{\text{aug}}$. It turns out that even ExML$^{\text{UA}}_{\text{csd}}$ could behave better than ExML$^{\text{ME}}_{\text{aug}}$. These results validate the effectiveness of the model cascade component.

5.3 Real Data of Activities Recognition

We additionally examine the effectiveness on a real-world dataset called *RealDisp*[2], which is an activities recognition task (Baños et al. 2012). There are 9 on-body sensors used to capture various actions of participants. Each sensor is placed on different parts of the body and provides 13-dimensional features including 3-dim from acceleration, 3-dim from gyro, 3-dim from magnetic field orientation and another 4-dim from quaternions. Hence, we have 117 features in total.

Dataset. Three types of actions (*walking*, *running*, and *jogging*) are included to form the dataset containing 2000 instances, where 30% of them are used for training and the remaining 70% for testing. In the training data, one sensor is deployed and the class of jogging is randomly misperceived as walking or running. The learner would explore the rest eight candidate features to discover the unknown unknowns. Thus, there are 9 partitions, and each is repeated for 10 times by sampling the training instances randomly.

Results. Figure 8 shows the mean and std of accuracy, our approach ExML (aka, ExML$^{\text{ME}}_{\text{csd}}$) outperforms others, validating the efficacy of our proposal. In addition, Figure 9 illustrates the budget allocation when the budget ratio $b = 30\%$. The i-th row denotes the scenario when the i-th sensor is the original feature, and patches with colors indicate the fraction of budget allocated to each candidate feature. The number above a patch means the attainable accuracy of the model trained on the whole training dataset with the particular feature. We highlight the top two candidate features of each row in white, and use blue color to indicate selected feature is not in top two. The results show that ExML with median elimination can select the top two informative features to augment for all the original sensors. The only exception is the 9-th sensor, but quality of the selected feature (91.8%) does not deviate too much from the best one (93.6%). These results reflect the effectiveness of our feature exploration strategy.

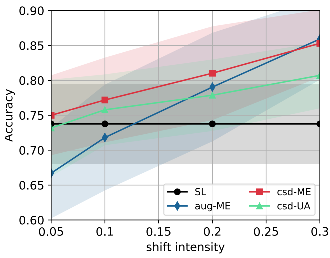


Figure 8: Performance comparisons of all contenders.

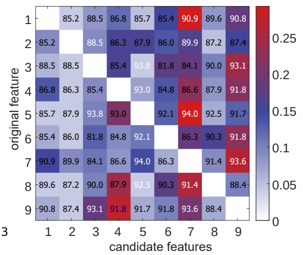


Figure 9: Budget allocation (median elimination).

6 Conclusion

In this paper, we identify that aside from the inadequate selection of learning algorithms or the lack of enough labeled training samples, unknown unknowns could also lead to the model failure. In particular, we are concerned with the scenario where some instances in the training dataset belong to an unknown hidden class but are wrongly perceived as known classes, due to the insufficient feature information. To address this issue, we propose the *exploratory machine learning* (ExML) to encourage the learner to examine and investigate the training dataset by exploring more features to discover potentially hidden classes. Following this idea, we design an approach consisting of three procedures: rejection model, feature exploration, and model cascade. By leveraging techniques from bandit theory, we prove the rationale and efficacy of the feature exploration procedure. Experiments validate the effectiveness of our approach.

There remain many directions for future investigations. For instance, as mentioned in Section 2.2, we can borrow more advanced techniques to further relax some model assumptions introduced in the current work (such as binary known classes, uniform cost, best feature exploration, etc). In particular, it is interesting to consider a personalized cost for each candidate feature, since we usually need to pay a higher price to obtain more informative features in real-world applications. Moreover, in addition to the feature exploration proposed in this paper, we argue that there are many other possibilities for ExML to deal with unknown unknowns, by means of adaptive interactions with environments.

[2] http://archive.ics.uci.edu/ml/datasets/REALDISP+Activity+Recognition+Dataset

References

Attenberg, J.; Ipeirotis, P.; and Provost, F. 2015. Beat the Machine: Challenging Humans to Find a Predictive Model's Unknown Unknowns. *ACM Journal of Data and Information Quality* 1–17.

Baños, O.; Damas, M.; Pomares, H.; Rojas, I.; Tóth, M. A.; and Amft, O. 2012. A benchmark dataset to evaluate sensor displacement in activity recognition. In *Proceedings of 12th ACM Conference on Ubiquitous Computing*, 1026–1035.

Bansal, G.; and Weld, D. S. 2018. A Coverage-Based Utility Model for Identifying Unknown Unknowns. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*, 1463–1470.

Bartlett, P. L.; and Wegkamp, M. H. 2008. Classification with a Reject Option using a Hinge Loss. *Journal of Machine Learning Research* 1823–1840.

Bousquet, O.; Boucheron, S.; and Lugosi, G. 2003. Introduction to Statistical Learning Theory. In *Advanced Lectures on Machine Learning (Machine Learning Summer Schools 2003)*, 169–207.

Bousquet, O.; and Zhivotovskiy, N. 2019. Fast classification rates without standard margin assumptions. *arXiv preprint* arXiv:1910.12756.

Cai, X.-Q.; Zhao, P.; Ting, K. M.; Mu, X.; and Jiang, Y. 2019. Nearest Neighbor Ensembles: An Effective Method for Difficult Problems in Streaming Classification with Emerging New Classes. In *Proceedings of the 19th International Conference on Data Mining (ICDM)*, 970–975.

Chen, L.; Li, J.; and Qiao, M. 2017. Nearly Instance Optimal Sample Complexity Bounds for Top-k Arm Selection. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, 101–110.

Chow, C. K. 1970. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory* 41–46.

Cortes, C.; DeSalvo, G.; and Mohri, M. 2016a. Boosting with Abstention. In *Advances in Neural Information Processing Systems 29*, 1660–1668.

Cortes, C.; DeSalvo, G.; and Mohri, M. 2016b. Learning with Rejection. In *Proceedings of International Conference on Algorithmic Learning Theory*, 67–82.

Da, Q.; Yu, Y.; and Zhou, Z.-H. 2014. Learning with Augmented Class by Exploiting Unlabeled Data. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence*, 1760–1766.

Dhurandhar, A.; and Sankaranarayanan, K. 2015. Improving Classification Performance Through Selective Instance Completion. *Machine Learning* 425–447.

Dietterich, T. G. 2017. Steps Toward Robust Artificial Intelligence. *AI Magazine* 3–24.

Even-Dar, E.; Mannor, S.; and Mansour, Y. 2006. Action Elimination and Stopping Conditions for the Multi-Armed Bandit and Reinforcement Learning Problems. *Journal of Machine Learning Research* 7: 1079–1105.

Gama, J.; Zliobaite, I.; Bifet, A.; Pechenizkiy, M.; and Bouchachia, A. 2014. A survey on concept drift adaptation. *ACM Computing Surveys* 46(4): 44:1–44:37.

Geng, C.; Huang, S.-J.; and Chen, S. 2018. Recent Advances in Open Set Recognition: A Survey. *arXiv preprint* arXiv:1811.08581.

Herbei, R.; and Wegkamp, M. H. 2006. Classification with reject option. *Canadian Journal of Statistics* 709–721.

Hou, B.-J.; Zhang, L.; and Zhou, Z.-H. 2017. Learning with Feature Evolvable Streams. In *Advances in Neural Information Processing Systems 30*, 1417–1427.

Hou, C.; and Zhou, Z.-H. 2018. One-Pass Learning with Incremental and Decremental Features. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40(11): 2776–2792.

Huang, S.; Xu, M.; Xie, M.; Sugiyama, M.; Niu, G.; and Chen, S. 2018. Active Feature Acquisition with Supervised Matrix Completion. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1571–1579.

Kalyanakrishnan, S.; Tewari, A.; Auer, P.; and Stone, P. 2012. PAC Subset Selection in Stochastic Multi-armed Bandits. In *Proceedings of the 29th International Conference on Machine Learning*, 227—–234.

Lakkaraju, H.; Kamar, E.; Caruana, R.; and Horvitz, E. 2017. Identifying Unknown Unknowns in the Open World: Representations and Policies for Guided Exploration. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, 2124–2132.

Liu, S.; Garrepalli, R.; Dietterich, T. G.; Fern, A.; and Hendrycks, D. 2018. Open Category Detection with PAC Guarantees. In *Proceedings of the 35th International Conference on Machine Learning*, 3175–3184.

Melville, P.; Provost, F. J.; and Mooney, R. J. 2005. An Expected Utility Approach to Active Feature-Value Acquisition. In *Proceedings of the 5th IEEE International Conference on Data Mining*, 745–748.

Mohri, M.; Rostamizadeh, A.; and Talwalkar, A. 2018. *Foundations of Machine Learning*. The MIT Press, second edition.

Mu, X.; Ting, K. M.; and Zhou, Z.-H. 2017. Classification Under Streaming Emerging New Classes: A Solution Using Completely-Random Trees. *IEEE Transactions on Knowledge and Data Engineering* 29(8): 1605–1618.

Mu, X.; Zhu, F.; Du, J.; Lim, E.-P.; and Zhou, Z.-H. 2017. Streaming Classification with Emerging New Class by Class Matrix Sketching. In Singh, S. P.; and Markovitch, S., eds., *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, 2373–2379.

Njoo, M.; and De Jong, T. 1993. Exploratory learning with a computer simulation for control theory: Learning processes and instructional support. *Journal of research in science teaching* 821–844.

Pan, S. J.; and Yang, Q. 2010. A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering* 1345–1359.

Scheirer, W. J.; de Rezende Rocha, A.; Sapkota, A.; and Boult, T. E. 2013. Toward Open Set Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1757–1772.

Scheirer, W. J.; Jain, L. P.; and Boult, T. E. 2014. Probability Models for Open Set Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2317–2324.

Schölkopf, B.; and Smola, A. J. 2002. *Learning with Kernels: support vector machines, regularization, optimization, and beyond*. The MIT Press.

Seldin, Y.; Bartlett, P. L.; Crammer, K.; and Abbasi-Yadkori, Y. 2014. Prediction with Limited Advice and Multiarmed Bandits with Paid Observations. In *Proceedings of the 31th International Conference on Machine Learning*, 280–287.

Settles, B. 2012. *Active Learning*. Morgan & Claypool Publishers.

Shim, H.; Hwang, S. J.; and Yang, E. 2018. Joint Active Feature Acquisition and Classification with Variable-Size Set Encoding. In *Advances in Neural Information Processing Systems 31*, 1375–1385.

Spector, J. M.; Merrill, M. D.; Elen, J.; and Bishop, M. 2014. *Handbook of Research on Educational Communications and Technology*. Springer, fourth edition.

van Breukelen, M.; Duin, R. P. W.; Tax, D. M. J.; and den Hartog, J. E. 1998. Handwritten digit recognition by combined classifiers. *Kybernetika* 381–386.

Wang, W.; and Qiao, X. 2018. Learning Confidence Sets using Support Vector Machines. In *Advances in Neural Information Processing Systems 31*, 4934–4943.

Yuan, M.; and Wegkamp, M. H. 2010. Classification Methods with Reject Option Based on Convex Risk Minimization. *Journal of Machine Learning Research* 111–130.

Zhang, C.; Wang, W.; and Qiao, X. 2018. On reject and refine options in multicategory classification. *Journal of the American Statistical Association* 730–745.

Zhang, Y.-J.; Zhao, P.; Ma, L.; and Zhou, Z.-H. 2020. An Unbiased Risk Estimator for Learning with Augmented Classes. In *Advances in Neural Information Processing Systems 33*, 10247–10258.

Appendix A Related Work

In this section, we discuss some related topics of the exploratory machine learning.

Open Category Learning. Open category learning is also named as learning with new classes, which focuses on handling unknown classes appearing only in the testing phase (Scheirer et al. 2013; Scheirer, Jain, and Boult 2014; Da, Yu, and Zhou 2014; Mu, Ting, and Zhou 2017; Mu et al. 2017; Liu et al. 2018; Cai et al. 2019; Zhang et al. 2020). Although these studies also care about the unknown classes detection, they differ from us significantly and are not able to apply to our situations. Specifically, first, they do not consider deficiency of the feature quality; second, there exist unknown classes in training data in our setting, while in theirs unknown classes only appear in the testing stage.

High-Confidence False Predictions. High-confidence false predictions appear due to model's unawareness of such kind of mistakes. Such instances and predictions are also referred to as a kind of "unknown unknowns" (Attenberg, Ipeirotis, and Provost 2015; Lakkaraju et al. 2017; Bansal and Weld 2018). As the model gives high-confidence for the false predictions, it is almost not possible to rectify the model's performance. As a result, existing studies typically ask for external human expert to help identifying high-confidence false predictions and then retrain the model with the guidance. Although these works also consider unknown unknowns and resort to external human knowledge, their setting and methodology differ from ours: our unknown unknowns are caused due to feature deficiency, so the learner requires to augment features rather than querying labels.

Active Learning. Active learning aims to achieve greater accuracy with fewer labels by asking queries of unlabeled data to be labeled by the human expert (Settles 2012). Besides, there are some works querying features (Melville, Provost, and Mooney 2005; Dhurandhar and Sankaranarayanan 2015; Huang et al. 2018), which tries to improve learning with missing features via as fewer as possible queries of entry values (the feature of an instance). Unlike their settings, in our work we augment new features to help the identification of the unknown classes rather than querying missing values of the given feature to improve the performance of known classes classification.

Learning with Rejection. Learning with rejection gives the classifier an option to reject an instance instead of providing a low-confidence prediction (Chow 1970). Plenty of works are proposed to design effective algorithms (Yuan and Wegkamp 2010; Cortes, DeSalvo, and Mohri 2016a; Wang and Qiao 2018; Shim, Hwang, and Yang 2018) and establish theoretical foundations (Herbei and Wegkamp 2006; Bartlett and Wegkamp 2008; Cortes, DeSalvo, and Mohri 2016b; Bousquet and Zhivotovskiy 2019). As aforementioned, methods of rejection cannot be directly applied in exploratory machine learning, because it will result in inaccurate rejections of instances from known classes, and meanwhile, it cannot exploit new features like exploratory machine learning.

Appendix B More Algorithm Details

This section presents more details for the feature exploration algorithm shown in Section 3.2, which attempts to identify the best feature from the K candidates with B budget. For better exploiting the budget, our algorithm essentially employing the idea of median elimination (Even-Dar, Mannor, and Mansour 2006). The algorithm proceeds in $T = \lceil \log_2 K \rceil$ episodes, where half of the poor features are removed every episode and only the best one remains in the last.

In each episode, in total $\lfloor B/T \rfloor$ budget is allocated uniformly to all remaining candidate features, and the learner could query their values for updating the corresponding augmented datasets D_i. Then, the score $\widehat{R}_{D_i}$ is calculated on the current augmented datasets D_i and the half features with high $\widehat{R}_{D_i}$ are eliminated. In the last, only one candidate feature i_s will be left and its augmented dataset D_{i_s} contains around $\lfloor B/\log K \rfloor$ samples, which is the largest among all candidate features.

Algorithm 1 Median Elimination for Feature Exploration

```
Input: Feature exploration budget $B$, original dataset
    $\widehat{D}_{tr} = \{(\widehat{\mathbf{x}}_i, \widehat{\mathbf{y}}_i)\}_{i=1}^m$, candidate feature pool
    $\mathcal{C}\{c_1, \ldots, c_K\}$, threshold $\theta \in (0, 1)$.
Output: Selected feature $c_{i_s} \in \mathcal{C}$ and corresponding augmented model $\widehat{f}_{i_s}$.
Initialize: dataset $D_i = \varnothing$ for each feature $c_i \in \mathcal{C}$, set of
    active features $\mathcal{C}_1 = \mathcal{C}$, $T = \lceil \log_2 K \rceil$.
for $t = 1, \ldots, T$ do
    Randomly select $n_t = \lfloor B/(T|\mathcal{C}_t|) \rfloor$ samples from
        $\widehat{D}_{tr}$ and query active features $c_i \in \mathcal{C}_t$;
    Update $D_i$ with selected samples and train a model $\widehat{f}_i$
        on $D_i$ by ERM (2), for all $c_i \in \mathcal{C}_t$;
    Compute $\widehat{R}_{D_i}$ according to (4), for all $c_i \in \mathcal{C}_t$;
    Update $\mathcal{C}_{t+1}$ as half of features in $\mathcal{C}_t$ with lower $\widehat{R}_{D_i}$;
end for
```

Appendix C Proof of Theorem 1

Proof of Theorem 1. The excess risk of the learned model $\widehat{f}_{i_s}$ can be decomposed into three parts,

$$
\begin{aligned}
& R_{i_s}(\widehat{f}_{i_s}) - R_1^* \\
= & \underbrace{R_{i_s}(\widehat{f}_{i_s}) - \widehat{R}_{i_s}(\widehat{f}_{i_s})}_{\texttt{term (a)}} + \underbrace{\widehat{R}_{i_s}(\widehat{f}_{i_s}) - \widehat{R}_1(\widehat{f}_1)}_{\texttt{term (b)}} + \underbrace{\widehat{R}_1(\widehat{f}_1) - R_1^*}_{\texttt{term (c)}},
\end{aligned}
$$

where $\texttt{term (a)}$ is the generalization error of the learned model $\widehat{f}_{i_s}$ and $\texttt{term (b)}$ is the difference between empirical criterion of the selected feature and that of the best feature, where $\widehat{f}_1$ refers to the model trained on the best feature with a $\lfloor B/K \rfloor$ budget. Besides, $\texttt{term (c)}$ captures the excess risk of $\widehat{f}_1$ relative to the Bayes risk. Notice that $\texttt{term (b)} < 0$ since the empirical criterion of the selected feature is the lowest among all candidates. Thus, to prove the theorem, it is sufficient to bound `term (a)` and `term (c)`.

We bound `term (a)` based on the following lemma on the generalization error of the rejection model, which can be regarded as a two-side counterpart of (Cortes, DeSalvo, and Mohri 2016b, Theorem 1).

Lemma 1. *Let $\mathcal{H}$ and $\mathcal{G}$ be the kernel-based hypotheses $\mathcal{H}$, $\mathcal{G} = \{\mathbf{x} \mapsto \langle \mathbf{w}, \Phi(\mathbf{x}) \rangle \mid \|\mathbf{w}\|_{\mathbb{H}} \leq \Lambda\}$. Then for any $\delta > 0$, with probability of $1 - \delta$ over the draw of a sample D of size n from $\mathcal{D}$, the following holds for all $f \in \mathcal{H} \times \mathcal{G}$:*

$$|R(f) - \widehat{R}(f)| \leq (2+\theta)\sqrt{\frac{(\kappa\Lambda)^2}{n}} + \sqrt{\frac{\log(2/\delta)}{2n}}, \quad (7)$$

where $\kappa^2 = \sup_{\mathbf{x}\in\mathcal{X}} K(\mathbf{x}, \mathbf{x})$ and $K : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is the kernel function associated with $\mathbb{H}$.

Therefore, the generalization error of the $\widehat{f}_{i_s}$ can be directly bounded by,

$$\texttt{term (a)} \leq (2+\theta)\sqrt{\frac{(\kappa_s\Lambda_s)^2}{\lfloor B/K \rfloor}} + \sqrt{\frac{\log(2/\delta')}{2\lfloor B/K \rfloor}}, \quad (8)$$

with probability at least $1 - \delta'$.

Then we process to analyze the excess risk of $\widehat{f}_1$ trained on the best feature with surrogate loss ℓ_{surr}, where `term (c)` can be decomposed into two parts as

$$\begin{aligned}\texttt{term (c)} &= \widehat{R}_1(\widehat{f}_1) - R_1^* \\ &= \underbrace{\widehat{R}_1(\widehat{f}_1) - R_1(\widehat{f}_1)}_{\texttt{term (c-1)}} + \underbrace{R_1(\widehat{f}_1) - R_1^*}_{\texttt{term (c-2)}}.\end{aligned}$$

Based on Lemma 1, `term (c-1)` is bounded by

$$\texttt{term (c-1)} \leq (2+\theta)\sqrt{\frac{(\kappa_1\Lambda_1)^2}{\lfloor B/K \rfloor}} + \sqrt{\frac{\log(2/\delta')}{2\lfloor B/K \rfloor}}, \quad (9)$$

which holds with probability at least $1 - \delta'$.

Before presenting the analysis on `term (c-2)`, we introduce the results on consistency and the generalization error of the surrogate loss function. First, we show that the excess risk with respect to 0/1 loss for any function f is bound by the excess risk with respect to the surrogate loss ℓ_{surr}.

Lemma 2 (Theorem 3 of Cortes, DeSalvo, and Mohri (2016b))**.** *Let $R^{surr}(f) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\ell_{surr}(f, \mathbf{x}, y; \theta)]$ denote the expected risk in terms of the surrogate loss of a pair $f = (h, g)$. Then, the excess error of f is upper bounded by its surrogate excess error as follows,*

$$R(f) - R^* \leq C_\theta \left(R^{surr}(f) - \inf_f R^{surr}(f) \right),$$

where $C_\theta = \frac{1}{(1-\theta)(1-2\theta)}$ and $R^* = \inf_f R(f)$.

Besides, the generalization error over the surrogate loss is bounded in the following lemma.

Lemma 3. *Let $\mathcal{H}$ and $\mathcal{G}$ be the kernel-based hypotheses, defined as $\mathcal{H}$, $\mathcal{G} = \{\mathbf{x} \mapsto \langle \mathbf{w}, \Phi(\mathbf{x}) \rangle \mid \|\mathbf{w}\|_{\mathbb{H}} \leq \Lambda\}$. Then for any $\delta > 0$, with probability of $1 - \delta$ over the draw of a sample D of size n from $\mathcal{D}$, the following holds for all $f \in \mathcal{H} \times \mathcal{G}$:*

$$|R^{surr}(f) - \widehat{R}^{surr}(f)| \leq \frac{2-2\theta}{1-2\theta}\sqrt{\frac{(\kappa\Lambda)^2}{n}} + B\sqrt{\frac{\log(2/\delta)}{2n}},$$

where $K : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is the kernel function associated with $\mathbb{H}$ and $\kappa^2 = \sup_{\mathbf{x}\in\mathcal{X}} K(\mathbf{x}, \mathbf{x})$. Moreover, $B = \sup_{f\in\mathcal{H}\times\mathcal{G}} \sup_{(\mathbf{x},y)\in\mathcal{X}\times\mathcal{Y}} \ell_{surr}(f, \mathbf{x}, y; \theta) \leq 1 + \max\{1, \theta/(1-2\theta)\} \cdot \kappa\Lambda$.

Lemma 3 can be obtained by the standard analysis of generalization error based on the Rademacher complexity. One may refer to Chapter 3 of the seminal textbook (Mohri, Rostamizadeh, and Talwalkar 2018). Then, based on the consistency property of the surrogate loss function demonstrated in Lemma 2, we have

$$\begin{aligned}&\texttt{term (c-2)} \\ &\leq C_\theta\big(R_1^{surr}(\widehat{f}_1) - \inf_f R_1^{surr}(f)\big) \\ &= C_\theta\Big(R_1^{surr}(\widehat{f}_1) - \inf_{f\in\mathcal{H}_1\times\mathcal{G}_1} R_1^{surr}(f) \\ &\qquad + \inf_{f\in\mathcal{H}_1\times\mathcal{G}_1} R_1^{surr}(f) - \inf_f R_1^{surr}(f)\Big) \\ &= C_\theta\Big(R_1^{surr}(\widehat{f}_1) - \inf_{f\in\mathcal{H}_1\times\mathcal{G}_1} R_1^{surr}(f)\Big) + R_{ap}.\end{aligned}$$

Denote $f_1^* = \inf_{f\in\mathcal{H}_1\times\mathcal{G}_1} R_1^{surr}(f)$. The first term of (10) can be further bounded by the generalization error bound of the the surrogate loss function and the optimality of $\widehat{f}_1$ for the ERM problem (2) in hypotheses $\mathcal{H}_1 \times \mathcal{G}_1$ as,

$$\begin{aligned}&R_1^{surr}(\widehat{f}_1) - R_1^{surr}(f_1^*) \\ &= R_1^{surr}(\widehat{f}_1) - \widehat{R}_1^{surr}(\widehat{f}_1) + \widehat{R}_1^{surr}(\widehat{f}_1) - R_1^{surr}(f_1^*) \\ &\leq R_1^{surr}(\widehat{f}_1) - \widehat{R}_1^{surr}(\widehat{f}_1) + \widehat{R}_1^{surr}(f_1^*) - R_1^{surr}(f_1^*) \\ &\leq 2 \sup_{f\in\mathcal{H}_1\times\mathcal{G}_1} |R_1^{surr}(f) - \widehat{R}_1^{surr}(f)| \\ &\leq \frac{4-4\theta}{1-2\theta}\sqrt{\frac{(\kappa_1\Lambda_1)^2}{\lfloor B/K \rfloor}} + 2B\sqrt{\frac{\log(2/\delta')}{2\lfloor B/K \rfloor}}.\end{aligned}$$

Here, for simplicity, we analyze the version of ERM problem in terms of the constraint on the RKHS norm instead of in terms of its Lagrange multiplier as (2), as they have identical regularization paths. Thus, term (c-2) is bounded by

$$\begin{aligned}&\texttt{term (c-2)} \\ &\leq C_\theta\left(\frac{4-4\theta}{1-2\theta}\sqrt{\frac{(\kappa_1\Lambda_1)^2}{\lfloor B/K \rfloor}} + 2B\sqrt{\frac{\log(2/\delta')}{2\lfloor B/K \rfloor}}\right) + R_{ap}\end{aligned} \quad (10)$$

with probability at least $1 - \delta'$.

Combining (8), (9) and (10), and setting $\kappa = \sup_{i\in[K]} \kappa_i$, $\Lambda = \sup_{i\in[K]} \Lambda_i$ and $\delta' = \delta/3$, we have

$$R_{i_s}(\widehat{h}_{i_s}, \widehat{g}_{i_s}) - R_1^*$$

$$\begin{aligned}
&\le (4+2\theta)\sqrt{\frac{(\kappa\Lambda)^2}{\lfloor B/K\rfloor}} + 2\sqrt{\frac{\log(6/\delta)}{2\lfloor B/K\rfloor}} \\
&\qquad + C_\theta\left(\frac{4-4\theta}{1-2\theta}\sqrt{\frac{(\kappa\Lambda)^2}{\lfloor B/K\rfloor}} + 2B\sqrt{\frac{\log(6/\delta)}{2\lfloor B/K\rfloor}}\right) + R_{ap} \\
&= \left(4+2\theta+\frac{4}{(1-2\theta)^2}\right)\sqrt{\frac{(\kappa\Lambda)^2}{\lfloor B/K\rfloor}} \\
&\qquad + 2(1+C_\theta B)\sqrt{\frac{\log(6/\delta)}{2\lfloor B/K\rfloor}} + R_{ap},
\end{aligned}$$

which holds with probability at least $1-\delta$. This ends the proof of Theorem 1. $\square$