Prior-Free Rare Category Detection

Jingrui He* Jaime Carbonell*

Abstract

Rare category detection is an open challenge in machine learning. It plays the central role in applications such as detecting new financial fraud patterns, detecting new network malware, and scientific discovery. In such cases rare categories are hidden among huge volumes of normal data and observations. In this paper, we propose a new method for rare category detection named SEDER, which requires no prior information about the data set. It implicitly performs semiparametric density estimation using specially designed exponentially families, and then picks the examples for labeling where the neighborhood density changes the most. SEDER can work in the cases where the data is not separable. Its unique feature over all existing methods lies in its prior-free nature, i.e. it does not require any prior information about the data set (e.g. the number of classes, the proportion of the different classes, etc.). Therefore, it is more suitable for real applications. Experimental results on both synthetic and real data sets demonstrate the superiority of SEDER.

1 Introduction.

Classical supervised learning methods require labeled examples representing each class, from which classifiers may be induced to predict class membership for unlabeled data. Whereas classifier induction has been well studied over the years, both for the balanced case [13], and the unbalanced case [18] [16] [12], very few methods have been proposed to discover classes in an unlabeled data set by proposing initial candidate examples of each class to a labeling oracle [14] [8] [10] [11]. Active learning [9] [6] focuses on the related problem of finding maximally discriminative examples to label, once each class has been discovered. If the data set is well balanced, we may use random sampling to find all the classes. On the other hand, if the data set is skewed, i.e. some classes dominate the data set (the majority classes) and the other classes rarely occur (the minority/rare classes), random sampling can prove extremely inefficient at discovering all the classes, especially the rare ones. It is often the case that these rare classes are of key importance; therefore, we need more sophisticated methods for rare category detection.

Rare category detection has a wealth of applications. For example, in financial fraud detection, the vast majority of financial transactions are legitimate, but a small number may be fraudulent; detecting early instances of the fraud patterns is a major first step towards systematically finding and stopping such illicit activity [3]. Another example is network intrusion detection. Systematically finding the early onset of new malicious network activities among huge volumes of routine network traffic is a critical unmet challenge [19]. Similarly, in astronomy, most of the objects in sky survey images are explainable by current theories and models, and only a tiny fraction of the objects may lead to new discoveries [14]. Rare category detection is also a bottleneck in reducing the sample complexity of active learning [2] [5].

Despite its importance, up until now, only a few methods have been proposed to address the rare category detection challenge in a general setting. For example, the method based on mixture models proposed in [14] is among the first attempts in this direction; in [8], the authors proposed a generic consistency algorithm, and proved upper bounds and lower bounds for this algorithm in some specific situations. Both of the two methods require that the support regions of the different classes be separable or near-separable to work well. The former also needs to be given the number of classes in the data set in order to train a reasonable mixture model [14]. More recently, in [10], the authors proposed NNDM algorithm for rare category detection, which is essentially a local-density-differential-sampling strategy. Different from the above two methods, NNDM does not depend on the separability assumption. In [11], the authors generalize the theoretical results for the binary case in [10] to the cases where we have multiple rare classes. However, NNDM needs to be given the number of classes as well as the proportion of the different classes in the data set, which is unrealistic in many real applications.

In this paper, we focus on the more challenging case where we do not have any prior information about the data set. The proposed method, SEmiparametric Density Estimation based Rare category detection (SEDER), implicitly performs semiparametric density estimation using specially designed exponentially fami-

*Carnegie Mellon University.

lies, and selects the examples with the largest norm of the gradients for labeling by the oracle. In this way, it focuses on the areas with the maximum change in the local density. Different from existing methods, SEDER does not require any prior information about the data set. Therefore, it is more suitable for real applications.

The rest of the paper is organized as follows. In Section 2, we introduce the specially designed exponentially families used in SEDER, and derive the scoring function. The complete algorithm of SEDER is presented in Section 3. In Section 4, we compare SEDER with state-of-the-art techniques on both synthetic and real data sets. Finally, we conclude the paper in Section 5.

2 Semiparametric Density Estimation for Rare Category Detection.

In rare category detection, we make the following assumptions: 1) the distribution of the majority classes is sufficiently smooth; and 2) the minority classes form compact clusters in the feature space. An example of the underlying distribution where these assumptions are satisfied is shown in Figure 1. Note that these assumptions are much more realistic than the separable/near-separable assumption assumed in [8] [14]. Based on our assumptions, abrupt changes in local density indicate the presence of rare classes. By sampling in these areas, we have high probability of finding examples from the rare classes. Following this line of reasoning, our proposed method SEDER implicitly estimates the density using specially designed exponential families, which essentially define a semiparametric model. At each data point, we set the score to be the norm of the gradient of the estimated density, which measures the maximum change rate of the local density, and pick the examples with the largest scores to be labeled by the oracle. Although the intuition of SEDER and NNDM [10] is quite similar: to pick the examples with the maximum change in the local density, NNDM is a nearest-neighbor-based method, it depends on the proportion of different classes to set the size of the neighborhood, and the scores of the examples roughly indicate the change in the local density; whereas SEDER is based on semiparametric density estimation, it is prior-free, i.e. it does not require any prior information about the data set, and the scores measure exactly the maximum change rate in the local density.

In this section, we first define some notations in subsection 2.1, and then introduce the specially designed exponential families in subsection 2.2. Finally we present the scoring function in subsection 2.3.

2.1 Notation. In rare category detection, we are given a set of unlabeled examples $S = \{x_1, \ldots, x_n\}$, $x_i \in \mathbb{R}^d$, which come from m distinct classes, i.e. $y_i \in \{1, \ldots, m\}$, $\forall i \in \{1, \ldots, n\}$. Without loss of generality, assume that $\sum_{i=1}^n x_i = \vec{0}$ and $\frac{1}{n}\sum_{i=1}^n x_i^2 = 1$. The proportion of some classes is much smaller than that of the other classes. They are the so-called rare classes. Table 1 summarizes the notations used in this paper. Our goal is to request as few total labels as possible in order to find at least one example from each class, especially those rare classes which are of particular interest to us.

Table 1: Notations

Symbol	Definition
S	The set of unlabeled examples
n	The number of examples in S
m	The number of classes in S
x_i	The i^{th} unlabeled example
x_i^j	The j^{th} feature of x_i
d	The dimensionality of the feature space
y_i	The class label of x_i
$g_\beta(x)$	The density defined by specially designed exponential families
$g_0(x)$	The carrier density
β_0	The normalizing parameter in $g_\beta(x)$
$t(x)$	The $p \times 1$ vector of sufficient statistics
$t^j(x)$	The j^{th} component of $t(x)$
β_1	The $p \times 1$ parameter vector
β_1^j	The j^{th} component of β_1
σ^j	The bandwidth for the j^{th} feature
β	(β_1, β_0)
$\hat{\beta}$	The maximum likelihood estimate of β
$l(\beta)$	The log-likelihood of the data
$g_\beta^j(x^j)$	The marginal distribution of the j^{th} feature based on $g_\beta(x)$
$g^j(x^j)$	The true marginal distribution of the j^{th} feature
b^j	Positive parameter which is a function of β_1^j
$\hat{b}^j$	The maximum likelihood estimate of b^j
A	$\frac{1}{n}\sum_{k=1}^n \frac{\sum_{i=1}^n \exp(-\frac{(x_k^j - x_i^j)^2}{2(\sigma^j)^2})(x_i^j)^2}{\sum_{i=1}^n \exp(-\frac{(x_k^j - x_i^j)^2}{2(\sigma^j)^2})}$
B	$(\sigma^j)^2$
C	$\frac{1}{n}\sum_{k=1}^n (x_k^j)^2$
$D_i(x)$	$\frac{1}{n}\prod_{j=1}^d \frac{1}{\sqrt{2\pi b^j}\sigma^j} \exp(-\frac{(x^j - b^j x_i^j)^2}{2(\sigma^j)^2 b^j})$
s_i	The score of x_i

2.2 Specially Designed Exponential Families. Traditional density estimation methods belong to two categories [7]: by fitting a parametric model via maximum likelihood, or by nonparametric methods such as kernel density estimation. For the purpose of rare category detection, parametric models are not appropriate since we can not assume a specific form of the underlying distribution for a given data set. On the other hand, the estimated density based on nonparametric methods tends to be under-smoothed, and the examples from rare classes will be buried among numerous spikes in the estimated density.

As proposed in [7], these two kinds of methods can be combined by putting an exponential family through a kernel density estimator, the so-called specially designed exponential families. It is a favorably compromise between parametric and nonparametric density estimation: the nonparametric smoother allows local adaptation to the data, while the exponential term matches some of the data's global properties, and makes the density much smoother [7]. To be specific, the estimated density $g_\beta(x) = g_0(x)\exp(\beta_0+\beta_1^T t(x))$ [7]. Here, $x \in \mathbb{R}^d$, $g_0(x)$ is a carrier density, $t(x)$ is a $p\times 1$ vector of sufficient statistics, β_1 is a $p\times 1$ parameter vector, and β_0 is a normalizing parameter that makes $g_\beta(x)$ integrate to 1. In our application, we use the kernel density estimator with the Gaussian kernel as the carrier density, i.e. $g_0(x) = \frac{1}{n}\sum_{i=1}^n \prod_{j=1}^d \frac{1}{\sqrt{2\pi}\sigma^j}\exp(-\frac{(x^j-x_i^j)^2}{2(\sigma^j)^2})$, where x^j is the j^{th} feature of x, x_i^j is the j^{th} feature of the i^{th} data point, and σ^j is the bandwidth for the j^{th} feature. In SEDER, σ^j is determined by cross validation [15] on the j^{th} feature. Here, the parameters $\beta = (\beta_1, \beta_0)$ can be estimated according to the following theorem.

THEOREM 2.1. *The maximum likelihood estimate $\hat{\beta}$ of β satisfies the following conditions [7]: $\forall j \in \{1,\ldots,p\}$*

$$\int_{x^1}\cdots\int_{x^d} t^j(x) g_{\hat{\beta}}(x)dx^d\cdots dx^1 = \frac{1}{n}\sum_{i=1}^n t^j(x_i)$$

where $t^j(x)$ is the j^{th} component of the vector $t(x)$.

Proof Firstly, notice that β_0 is a normalizing parameter that makes $g_\beta(x)$ integrate to 1, i.e.

$$\beta_0 = -\log\int_{x^1}\cdots\int_{x^d} g_0(x)\exp(\beta_1^T t(x))dx^d\cdots dx^1$$

Therefore, $\forall j \in \{1,\ldots,p\}$

$$\begin{aligned}\frac{\partial\beta_0}{\partial\beta_1^j} &= -\frac{\int_{x^1}\cdots\int_{x^d} t^j(x)g_0(x)\exp(\beta_1^T t(x))dx^d\cdots dx^1}{\int_{x^1}\cdots\int_{x^d} g_0(x)\exp(\beta_1^T t(x))dx^d\cdots dx^1}\\ &= -\int_{x^1}\cdots\int_{x^d} t^j(x)g_0(x)\exp(\beta_0+\beta_1^T t(x))dx^d\cdots dx^1\\ &= -\int_{x^1}\cdots\int_{x^d} t^j(x)g_\beta(x)dx^d\cdots dx^1\end{aligned}$$

where β_1^j is the j^{th} component of the vector β_1.

Secondly, the log-likelihood of the data is $l(\beta) = \sum_{i=1}^n \log(g_\beta(x_i)) = \sum_{i=1}^n \log(g_0(x_i)) + n\beta_0 + \sum_{i=1}^n \beta_1^T t(x_i)$. Taking the partial derivative of $l(\beta)$ with respect to β_1^j, we have:

$$\begin{aligned}\frac{\partial l(\beta)}{\partial\beta_1^j} &= n\frac{\partial\beta_0}{\partial\beta_1^j} + \sum_{i=1}^n t^j(x_i)\\ &= -n\int_{x^1}\cdots\int_{x^d} t^j(x)g_\beta(x)dx^d\cdots dx^1 + \sum_{i=1}^n t^j(x_i)\end{aligned}$$

Setting the partial derivative to 0, we have that the maximum likelihood estimate $\hat{\beta}$ of β satisfies $\int_{x^1}\cdots\int_{x^d} t^j(x)g_{\hat{\beta}}(x)dx^d\cdots dx^1 = \frac{1}{n}\sum_{i=1}^n t^j(x_i)$, $\forall j \in \{1,\ldots,p\}$.■

In SEDER, we set the vector of sufficient statistics to be $t(x) = [(x^1)^2,\ldots,(x^d)^2]^T$ [1]. If we estimate the parameters according to Theorem 2.1, different parameters will be coupled due to the normalizing parameter β_0. Let β_1^j be the j^{th} component of the vector β_1. In order to de-couple the estimation of different β_1^js, we make the following changes. Firstly, we decompose β_0 into β_0^js such that $\sum_{j=1}^d \beta_0^j = \beta_0$, then $g_\beta(x)$ can be seen as a kernel density estimator with a 'special' kernel, i.e. $g_\beta(x) = \frac{1}{n}\sum_{i=1}^n\prod_{j=1}^d[\frac{1}{\sqrt{2\pi}\sigma^j}\exp(-\frac{(x^j-x_i^j)^2}{2(\sigma^j)^2})\exp(\beta_0^j+\beta_1^j(x^j)^2)]$. Next, we relax the constraint on β_0^js, and let them depend on x_i^j in such a way that

$$\int_{x^j}\frac{1}{\sqrt{2\pi}\sigma^j}\exp(-\frac{(x^j-x_i^j)^2}{2(\sigma^j)^2})\exp(\beta_{0i}^j+\beta_1^j(x^j)^2)dx^j = 1 \tag{2.1}$$

where β_{0i}^j implies the dependence of β_0^j on x_i^j. In this

[1] Note that the following analysis also applies to other forms of the sufficient statistics, such as $t(x) = [x^1,\ldots,x^d]^T$. In all our experiments, the second order sufficient statistics perform the best. So we use this form in SEDER.

way, the marginal distribution of the j^{th} feature is

$$
\begin{aligned}
&g_\beta^j(x^j)\\
&= \int_{x^1}\cdots\int_{x^{j-1}}\int_{x^{j+1}}\cdots\int_{x^d} g_\beta(x)dx^d\cdots dx^{j+1}\\
&dx^{j-1}\cdots dx^1\\
&= \frac{1}{n}\sum_{i=1}^{n}\{\frac{1}{\sqrt{2\pi}\sigma^j}\exp(-\frac{(x^j-x_i^j)^2}{2(\sigma^j)^2})\exp(\beta_{0i}^j+\beta_1^j(x^j)^2)\cdot\\
&\prod_{k\neq j}\int_{x^k}\frac{1}{\sqrt{2\pi}\sigma^k}\exp(-\frac{(x^k-x_i^k)^2}{2(\sigma^k)^2})\exp(\beta_{0i}^k+\beta_1^k(x^k)^2)dx^k\}\\
&= \frac{1}{n}\sum_{i=1}^{n}\frac{1}{\sqrt{2\pi}\sigma^j}\exp(-\frac{(x^j-x_i^j)^2}{2(\sigma^j)^2})\exp(\beta_{0i}^j+\beta_1^j(x^j)^2)
\end{aligned}
$$

To estimate the parameters in our current model, we have the following theorem.

THEOREM 2.2. *The maximum likelihood estimates $\hat{\beta}_1^j$ and $\hat{\beta}_{0i}^j$ of β_1^j and β_{0i}^j satisfy the following conditions:* $\forall j\in\{1,\ldots,d\}$

$$
\sum_{k=1}^{n}(x_k^j)^2 = \sum_{k=1}^{n}\frac{\sum_{i=1}^{n}\exp(\hat{\beta}_{0i}^j-\frac{(x_k^j-x_i^j)^2}{2(\sigma^j)^2})E_i^j((x^j)^2)}{\sum_{i=1}^{n}\exp(\hat{\beta}_{0i}^j-\frac{(x_k^j-x_i^j)^2}{2(\sigma^j)^2})} \tag{2.2}
$$

where $E_i^j((x^j)^2) = \int_{x^j}(x^j)^2\frac{1}{\sqrt{2\pi}\sigma^j}\exp(-\frac{(x^j-x_i^j)^2}{2(\sigma^j)^2})\exp(\hat{\beta}_{0i}^j+\hat{\beta}_1^j(x^j)^2)dx^j$.

Proof First of all, according to Equation (2.1), we have $\beta_{0i}^j = -\log\int_{x^j}\frac{1}{\sqrt{2\pi}\sigma^j}\exp(-\frac{(x^j-x_i^j)^2}{2(\sigma)^2})\exp(\beta_1^j(x^j)^2)dx^j$. Therefore,

$$
\begin{aligned}
\frac{\partial\beta_{0i}^j}{\beta_1^j} &= -\frac{\int_{x^j}\frac{1}{\sqrt{2\pi}\sigma^j}\exp(-\frac{(x^j-x_i^j)^2}{2(\sigma)^2})\exp(\beta_1^j(x^j)^2)(x^j)^2dx^j}{\int_{x^j}\frac{1}{\sqrt{2\pi}\sigma^j}\exp(-\frac{(x^j-x_i^j)^2}{2(\sigma)^2})\exp(\beta_1^j(x^j)^2)dx^j}\\
&= -\int_{x^j}\frac{1}{\sqrt{2\pi}\sigma^j}\exp(-\frac{(x^j-x_i^j)^2}{2(\sigma^j)^2})\\
&\exp(\beta_{0i}^j+\beta_1^j(x^j)^2)(x^j)^2dx^j\\
&= -E_i^j((x^j)^2)
\end{aligned}
$$

Then the log-likelihood of the data on the j^{th} component is

$$
\begin{aligned}
l(\beta_1^j) &= \sum_{k=1}^{n}\log(g_\beta^j(x_k^j))\\
&= \sum_{k=1}^{n}\log(\frac{1}{n}\sum_{i=1}^{n}\frac{1}{\sqrt{2\pi}\sigma^j}\exp(-\frac{(x_k^j-x_i^j)^2}{2(\sigma^j)^2})\cdot\\
&\exp(\beta_{0i}^j+\beta_1^j(x_k^j)^2))\\
&= \sum_{k=1}^{n}\log(\frac{1}{n\sqrt{2\pi}\sigma^j}\exp(\beta_1^j(x_k^j)^2)\cdot\\
&\sum_{i=1}^{n}\exp(-\frac{(x_k^j-x_i^j)^2}{2(\sigma^j)^2})\exp(\beta_{0i}^j))
\end{aligned}
$$

Taking the partial derivative of $l(\beta_1^j)$ with respect to β_1^j, we have:

$$
\begin{aligned}
\frac{\partial l(\beta_1^j)}{\partial\beta_1^j} &= \sum_{k=1}^{n}(x_k^j)^2+\sum_{k=1}^{n}\frac{\sum_{i=1}^{n}\exp(\beta_{0i}^j-\frac{(x_k^j-x_i^j)^2}{2(\sigma^j)^2})\frac{\partial\beta_{0i}^j}{\partial\beta_1^j}}{\sum_{i=1}^{n}\exp(\beta_{0i}^j-\frac{(x_k^j-x_i^j)^2}{2(\sigma^j)^2})}\\
&= \sum_{k=1}^{n}(x_k^j)^2-\sum_{k=1}^{n}\frac{\sum_{i=1}^{n}\exp(\beta_{0i}^j-\frac{(x_k^j-x_i^j)^2}{2(\sigma^j)^2})E_i^j((x^j)^2)}{\sum_{i=1}^{n}\exp(\beta_{0i}^j-\frac{(x_k^j-x_i^j)^2}{2(\sigma^j)^2})}
\end{aligned}
$$

Setting the partial derivative to 0, we have that the maximum likelihood estimate $\hat{\beta}_1^j$ and $\hat{\beta}_{0i}^j$ of β_1^j and β_{0i}^j satisfy $\sum_{k=1}^{n}(x_k^j)^2 = \sum_{k=1}^{n}\frac{\sum_{i=1}^{n}\exp(\hat{\beta}_{0i}^j-\frac{(x_k^j-x_i^j)^2}{2(\sigma^j)^2})E_i^j((x^j)^2)}{\sum_{i=1}^{n}\exp(\hat{\beta}_{0i}^j-\frac{(x_k^j-x_i^j)^2}{2(\sigma^j)^2})}$.■

Notice that according to Theorem 2.2, β_1^js can be estimated separately, which greatly simplifies our problem. At the first glance, Equation (2.2) is hard to solve. Next, we let $\beta_1^j = (1-\frac{1}{b^j})\frac{1}{2(\sigma^j)^2}$, where $b^j\neq 1$ is a positive parameter, the introduction of which will simplify this equation. According to Equation (2.1), β_{0i}^j can be expressed in terms of b^j, i.e.

$$
\begin{aligned}
\beta_{0i}^j &= -\log\int_{x^j}\frac{1}{\sqrt{2\pi}\sigma^j}\exp(-\frac{(x^j-x_i^j)^2}{2(\sigma^j)^2}+\beta_1^j(x^j)^2)dx^j\\
&= -\log\int_{x^j}\frac{1}{\sqrt{2\pi}\sigma^j}\exp(-\frac{(x^j)^2+b^j(x_i^j)^2-2b^jx^jx_i^j}{2(\sigma^j)^2b^j})dx^j\\
&= \frac{(1-b^j)(x_i^j)^2}{2(\sigma^j)^2}-\frac{1}{2}\log b^j
\end{aligned}
$$

Therefore, the estimated density becomes

$$
\tilde{g}_b(x) = \frac{1}{n}\sum_{i=1}^{n}\prod_{j=1}^{d}\frac{1}{\sqrt{2\pi b^j}\sigma^j}\exp(-\frac{(x^j-b^jx_i^j)^2}{2(\sigma^j)^2b^j}) \tag{2.3}
$$

Replacing $\hat{\beta}_1^j$ and $\hat{\beta}_{0i}^j$ with functions of $\hat{b}^j$ (the maximum likelihood estimate of b^j) in the definition

of $E_i^j((x^j)^2)$, we have $E_i^j((x^j)^2) = \hat{b}^j(\sigma^j)^2 + (\hat{b}^j)^2(x_i^j)^2$, and Equation (2.2) becomes

$$\sum_{k=1}^{n}(x_k^j)^2 = n\hat{b}^j(\sigma^j)^2 + (\hat{b}^j)^2\sum_{k=1}^{n}\frac{\sum_{i=1}^{n}\exp(\frac{(1-\hat{b}^j)(x_i^j)^2}{2(\sigma^j)^2} - \frac{(x_k^j-x_i^j)^2}{2(\sigma^j)^2})(x_i^j)^2}{\sum_{i=1}^{n}\exp(\frac{(1-\hat{b}^j)(x_i^j)^2}{2(\sigma^j)^2} - \frac{(x_k^j-x_i^j)^2}{2(\sigma^j)^2})}$$

In general, the value of $\hat{\beta}_1^j$ is very close to 0, and $g_{\hat{\beta}}(x)$ is a smoothed version of $g_0(x)$. Therefore, $\hat{b}^j$ should be close to 1, and we can re-write the above equation as follows.

$$\frac{1}{n}\sum_{k=1}^{n}(x_k^j)^2 \approx \hat{b}^j(\sigma^j)^2 + (\hat{b}^j)^2\frac{1}{n}\sum_{k=1}^{n}\frac{\sum_i \exp(-\frac{(x_k^j-x_i^j)^2}{2(\sigma^j)^2})(x_i^j)^2}{\sum_i \exp(-\frac{(x_k^j-x_i^j)^2}{2(\sigma^j)^2})}$$

This is a second-degree polynomial equation of $\hat{b}^j$, and the roots can be easily obtained by Vieta's theorem[2], i.e. $\forall j \in \{1,\ldots,d\}$

$$\hat{b}^j = \frac{-B + \sqrt{B^2 + 4AC}}{2A} \tag{2.4}$$

where $A = \frac{1}{n}\sum_{k=1}^{n}\frac{\sum_{i=1}^{n}\exp(-\frac{(x_k^j-x_i^j)^2}{2(\sigma^j)^2})(x_i^j)^2}{\sum_{i=1}^{n}\exp(-\frac{(x_k^j-x_i^j)^2}{2(\sigma^j)^2})}$, $B = (\sigma^j)^2$, and $C = \frac{1}{n}\sum_{k=1}^{n}(x_k^j)^2$.

THEOREM 2.3. *Let $g^j(x^j)$ be the true density for the j^{th} feature. If $\frac{1}{n}\sum_{i=1}^{n}\frac{x_i^j}{g^j(x_i^j)}\cdot\frac{dg^j(x_i^j)}{dx_i^j} \geq -1 + O(1)$, then $\hat{b}^j \leq 1$ and $\hat{\beta}_1^j \leq 0$.*

Proof For the sake of simplicity, let $z = x^j$, $h = \sigma^j$, and $f(z) = g^j(x^j)$. Then $A = \frac{1}{n}\sum_{k=1}^{n}\frac{\sum_{i=1}^{n}\exp(-\frac{(z_k-z_i)^2}{2h^2})(z_i)^2}{\sum_{i=1}^{n}\exp(-\frac{(z_k-z_i)^2}{2h^2})}$, $B = h^2$, and $C = \frac{1}{n}\sum_{k=1}^{n}(z_k)^2$. Consider the following regression problem where the true regression function $r(z) = z^2$, the noise has mean 0, and we use kernel regression to estimate this function. Then $A - C$ is the bias of kernel regression on the training data, i.e. $A - C = \frac{1}{n}\sum_{i=1}^{n}h^2(\frac{1}{2}r''(z_i) + \frac{r'(z_i)f'(z_i)}{f(z_i)})\int z^2k(z)dz + O(h^2)$ [17], where $k(z)$ is the Gaussian kernel used in kernel regression, i.e. $k(z) = \frac{1}{\sqrt{2\pi}}\exp(-\frac{z^2}{2})$. Therefore, $A - C = h^2 + \frac{h^2}{n}\sum_{i=1}^{n}\frac{2z_if'(z_i)}{f(z_i)} + O(h^2)$, and $A + B - C \geq 0$ if and only if $\frac{1}{n}\sum_{i=1}^{n}\frac{z_if'(z_i)}{f(z_i)} \geq -1 + O(1)$. Given that $A + B - C \geq 0$, we can show that $\hat{b}^j = \frac{-B+\sqrt{B^2+4AC}}{2A} \leq \frac{-B+\sqrt{B^2+4A(A+B)}}{2A} = 1$, and $\hat{\beta}_1^j = (1 - \frac{1}{\hat{b}^j})\frac{1}{2(\sigma^j)^2} \leq 0$.■

At the beginning of Section 2, we have made the following assumptions: 1) the distribution of the majority classes is sufficiently smooth; and 2) the minority classes form compact clusters in the feature space. In this case, the first order derivative of the density would be close to 0 for most examples, and have large absolute values for a few examples near the rare classes. Therefore, the condition in Theorem 2.3 is always satisfied, and the exponential term appended to the carrier density decreases away from the origin.

2.3 Scoring Function. Once we have estimated all the parameters using Equation (2.4), we can measure the change in the local density at each data point based on the estimated density in Equation (2.3). Note that at each data point, if we pick a different direction, the change in local density would be different. In SEDER, we measure the change along the gradient, which gives the maximum change at each data point.

THEOREM 2.4. *Using the estimated density in Equation (2.3), $\forall x \in \mathbb{R}^d$, the maximum change rate of the density at x is $\sqrt{\sum_{l=1}^{d}\frac{(\sum_{i=1}^{n}D_i(x)(x^l - b^lx_i^l))^2}{((\sigma^l)^2b^l)^2}}$, where $D_i(x) = \frac{1}{n}\prod_{j=1}^{d}\frac{1}{\sqrt{2\pi b^j}\sigma^j}\exp(-\frac{(x^j-b^jx_i^j)^2}{2(\sigma^j)^2b^j})$ is the contribution of x_i to the estimated density at x.*

Proof of $\forall x \in \mathbb{R}^d$, let the gradient vector be $w \in \mathbb{R}^d$. We have $\forall l \in \{1,\ldots,d\}$

$$w_l = \frac{\partial\tilde{g}_b(x)}{\partial x^l} = \frac{1}{n}\sum_{i=1}^{n}(-\frac{x^l - b^lx_i^l}{(\sigma^l)^2b^l})\prod_{j=1}^{d}\frac{\exp(-\frac{(x^j-b^jx_i^j)^2}{2(\sigma^j)^2b^j})}{\sqrt{2\pi b^j}\sigma^j}$$
$$= -\sum_{i=1}^{n}\frac{D_i(x)(x^l - b^lx_i^l)}{(\sigma^l)^2b^l}$$

where w_l is the l^{th} component of w.

Therefore, the maximum change rate of the density at x is

$$\|w\|_2 = \sqrt{\sum_{l=1}^{d}(-\sum_{i=1}^{n}\frac{D_i(x)(x^l - b^lx_i^l)}{(\sigma^l)^2b^l})^2} = \sqrt{\sum_{l=1}^{d}\frac{(\sum_{i=1}^{n}D_i(x)(x^l - b^lx_i^l))^2}{((\sigma^l)^2b^l)^2}}$$■

If the distribution of the majority classes is sufficiently smooth, and the minority classes form compact clusters in the feature space, the minority classes are always located in the regions where the density changes

[2]Note that the other root $\frac{-B-\sqrt{B^2+4AC}}{2A}$ is disregarded since it is negative.

the most. Therefore, in SEDER, to discover the rare classes, we set the score of each example to be the maximum change rate of the density at this example, i.e. $\forall k \in \{1, \ldots, n\}$

$$s_k = \sqrt{\sum_{l=1}^{d} \frac{(\sum_{i=1}^{n} D_i(x_k)(x_k^l - b^l x_i^l))^2}{((\sigma^l)^2 b^l)^2}} \tag{2.5}$$

where s_k is the score of x_k. We pick the examples with the largest scores for labeling until we find at least one example from each class.

3 Algorithm.

The intuition of SEDER is to select the examples with the maximum change in the local density for labeling by the oracle. As introduced in subsection 2.3, the scores of the examples measure the maximum change rate in the local density, and they do not take into account the fact that nearby examples tend to have the same class label. Therefore, if we ask the oracle to label all the examples with large scores, we may repeatedly select examples from the most distinctive rare class, rather than discovering all the rare classes. To address this problem in SEDER, we make use of the following heuristic: if $x_i \in S$ has been labeled, $\forall x_k \in S$, $x_k \neq x_i$, if $\forall j \in \{1, \ldots, d\}$, $|x_i^j - x_k^j| \leq 3\sigma^j$, we would preclude x_k from being selected. In other words, if an unlabeled example is very close to a previously labeled one, it is quite likely that the labels of the two examples are the same, and labeling that example will not have a high probability of detecting a new rare class. The size of the neighborhood is set to $3\sigma^j$ such that the estimated density for the examples outside this neighborhood using Gaussian kernel is hardly affected by the labeled example. It should be pointed out that the feedback strategy is orthogonal to the remaining parts of the proposed algorithm. In our experiments, we find that despite its simplicity, the current strategy leads to satisfactory performance.

The proposed method, SEDER, is summarized in Algorithm 1. It works as follows. Firstly, we initialize the set I of selected examples and the set L of their labels to empty sets. Then step 2 to step 5 calculate the parameters in our model. Step 6 to step 8 calculate the score for each example in S. Finally, step 9 to step 13 gradually include the example with the maximum score into I and its label into L until we run out of the labeling budget. In each round, the selected example should be far away from all the labeled examples.

Note that: 1) unlike the methods proposed in [10] [14], SEDER does not need to be given the number of classes in S or any other information, hence it is more suitable for real applications; 2) in SEDER, we do not need to explicitly calculate the density at each example; 3) SEDER does not depend on the assumption that different classes be separable or near-separable.

Algorithm 1 SEmiparametric Density Estimation based Rare category detection (SEDER)

Input: Unlabeled data set S
Output: The set I of selected examples and the set L of their labels

1: Initialize $I = \phi$ and $L = \phi$.
2: **for** $j = 1 : d$ **do**
3: Calculate the bandwidth σ^j using cross validation [15].
4: Calculate the maximum likelihood estimate $\hat{b}^j$ of the parameter b^j according to Equation (2.4).
5: **end for**
6: **for** $i = 1 : n$ **do**
7: Calculate the score s_i of the i^{th} example according to Equation (2.5) using the estimated parameters.
8: **end for**
9: **while** the labeling budget is not exhausted **do**
10: Set $S' = \{x | x \in S, \forall i \in I, \exists j \in \{1, \ldots, d\}, \text{ s.t. } |x^j - x_i^j| > 3\sigma^j\}$
11: Query $x = \text{argmax}_{x_i \in S'} s_i$ for its label y_x
12: $I = I \cup \{x\}$, $L = L \cup \{y_x\}$;
13: **end while**

4 Experimental Results.

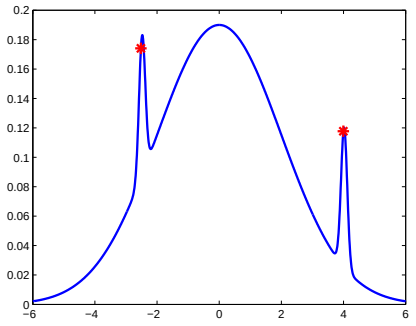
In this section, we compare SEDER with NNDM [10], Interleave (the best method proposed in [14]), random sampling (RS) and SEDER with $b^j = 1$ for $j = 1, \ldots, d$ (abbreviated as Kernel, which is equivalent to using kernel density estimator to estimate the density and to get the scores) on both synthetic and real data sets. For this purpose, we run these methods until all the classes have discovered, and compare the number of label requests by each method in order to find a certain number of classes. Note that SEDER, NNDM and Kernel are deterministic, whereas the results for Interleave and random sampling are averaged over 100 runs.

Here we would like to emphasize that only SEDER, RS and Kernel do not need any prior information about the data set, whereas NNDM and Interleave need extra information about the data set as inputs, such as the number of classes and the proportion of the different classes. When such prior information is not available, which is quite common in real applications, NNDM and Interleave are not applicable.

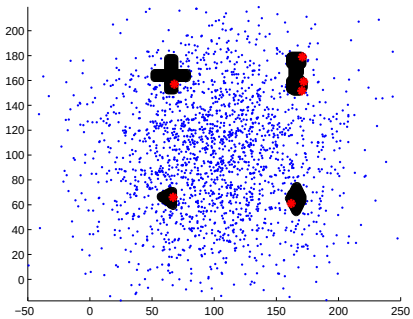
4.1 SYNTHETIC DATA SETS Figure 1(a) shows the underlying distribution of a 1-dimensional

synthetic data set. The majority class with 2000 examples has a Gaussian distribution with a large variance; whereas the minority classes with 50 examples each correspond to the two lower-variance peaks. As can be seen from this figure, the first two examples selected by SEDER (red stars) are both from the regions where the density changes the most.

Figure 1(b) shows a 2-dimensional synthetic data set. The majority class has 2000 examples (blue dots) with a Gaussian distribution. The four minority classes (black circles) all have different shapes, and each has 267, 280, 84 and 150 examples respectively. This data set is similar to the one used in [11]. To discover all the classes, SEDER only needs to label 6 examples, which are represented by red stars in the figure; whereas random sampling needs to label more than 50 examples on average.



(a) Underlying distribution of a 1-dimensional synthetic data set



(b) 2-dimensional synthetic data set

Figure 1: Synthetic data sets: red stars represent the examples selected by SEDER

4.2 REAL DATA SETS In this subsection, we present the experimental results on some real data sets. The properties of the data sets are summarized in Table 2. Notice that all these data sets are skewed: the proportion of the smallest class is less than 5%. For the last three data sets (Page Blocks, Abalone and Shuttle), it is even less than 1%. We refer to these three data sets as 'extremely' skewed; whereas the remaining two data sets (Ecoli and Glass) are referred to as 'moderately' skewed.

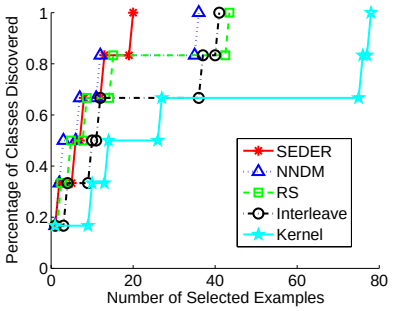
Table 2: Properties of the data sets used

Data Set	n	d	m	Largest Class	Smallest Class
Ecoli [1]	336	7	6	42.56%	2.68%
Glass [1]	214	9	6	35.51%	4.21%
Page Blocks [1]	5473	10	5	89.77%	0.51%
Abalone [1]	4177	7	20	16.50%	0.34%
Shuttle [4]	4515	9	7	75.53%	0.13%

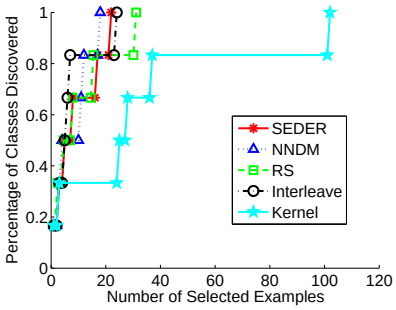
First, let us focus on the 'moderately' skewed data sets, which are shown in Figure 2. With Ecoli data set, to discover all the classes, NNDM needs 36 label requests, Interleave needs 41 label requests on average, RS needs 43 label requests on average, Kernel needs 78 label requests, and SEDER only needs 20 label requests; with Glass data set, to discover all the classes, NNDM needs 18 label requests, Interleave needs 24 label requests on average, RS needs 31 label requests on average, Kernel needs 102 label requests, and SEDER needs 22 label requests. Therefore, if the data set is 'moderately' skewed, the performance of SEDER is better than or comparable with NNDM, which requires more prior information than SEDER, including the number of classes in the data set and the proportion of the different classes.

Next, let us look at the 'extremely' skewed data sets. For example, in Shuttle data set, the largest class has 580 times more examples than the smallest class. With Page Blocks data set (Figure 3(a)), to discover all the classes, SEDER needs 36 label requests, NNDM needs 23 label requests, Interleave needs 77 label requests on average, RS needs 199 label requests on average, and Kernel needs more than 1000 label requests; with Abalone data set (Figure 3(b)), to discover all the classes, SEDER needs 316 label requests, NNDM needs 179 label requests, Interleave needs 333 label requests on average, RS needs 483 label requests on average[3], and Kernel needs more than 1000 label requests; with Shuttle data set (Figure 3(c)), to discover all the classes,

[3]Note that with Abalone data set, the results of NNDM and Interleave are slightly different from [10]. This is due to the effect of normalization on the data.

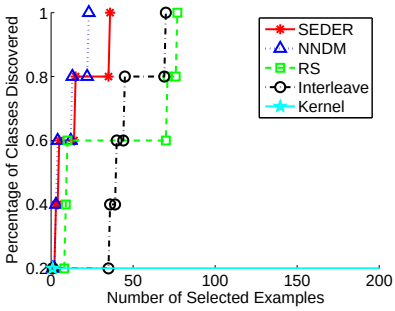


(a) Ecoli

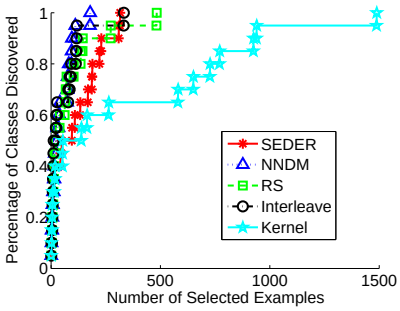


(b) Glass

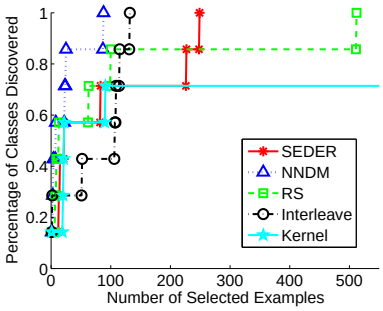
Figure 2: Real data sets: 'moderately' skewed



(a) Page Blocks



(b) Abalone



(c) Shuttle

Figure 3: Real data sets: 'extremely' skewed

SEDER needs 249 label requests, NNDM needs 87 label requests, Interleave needs 140 label requests on average, RS needs 512 label requests on average, and Kernel needs more than 1000 label requests.

Based on the above results, we have the following observations. First, SEDER, RS and Kernel require no prior information about the data set, and yet SEDER is significantly better than RS and Kernel in all the experiments. Second, if the data is not separable, the performance of Interleave is worse than SEDER (except Figure 3(c)), even though it is given the additional information about the number of classes in the data set. Finally, although NNDM is better than SEDER for the 'extremely' skewed data sets, in real applications, it is very difficult to estimate the number of classes in the data set, not to mention the proportion of the different classes. If the information provided to NNDM is not accurate enough, the performance of NNDM may be negatively affected. Moreover, when such information is not available, NNDM is not applicable at all.

5 Conclusion.

In this paper, we have proposed a new method for rare category detection named SEDER, which requires no prior information about the data set. It implicitly estimates the density using specially designed exponential families, which is essentially a semiparametric approach, and selects examples with the maximum norm of the gradient in the estimated density for labeling by an oracle.

To the best of our knowledge, SEDER is the first method tailored for the very challenging case where no prior information about the data set is available. Therefore, we expect it be more suitable for many real applications. The proposed method is based on sound theoretical analysis and its effectiveness is demonstrated by extensive experimental evaluations.

References

[1] A. Asuncion and D. Newman. UCI machine learning repository, 2007.
[2] M. Balcan, A. Beygelzimer, and J. Langford. Agnostic active learning. In *ICML*, pages 65–72, 2006.
[3] S. Bay, K. Kumaraswamy, M. Anderle, R. Kumar, and D. Steier. Large scale detection of irregularities in accounting data. In *ICDM*, pages 75–86, 2006.
[4] P. Brazdil and J. Gama. Statlog repository. In *http://www.niaad.liacc.up.pt/old/statlog/datasets /shuttle/shuttle.doc.html*, 1991.
[5] S. Dasgupta. Coarse sample complexity bounds for active learning. In *NIPS*, 2005.
[6] P. Donmez and J. Carbonell. Paired sampling in density-sensitive active learning. In *ISAIM*, 2008.
[7] B. Efron and R. Tibshirani. Using specially designed exponential families for density estimation. In *Proc. of the Annals of Statistics*, pages 2431–2461, 1996.
[8] S. Fine and Y. Mansour. Active sampling for multiple output identification. In *COLT*, pages 620–634, 2006.
[9] Y. Guo and D. Schuurmans. Discriminative batch mode active learning. In *NIPS*, 2007.
[10] J. He and J. Carbonell. Nearest-neighbor-based active learning for rare category detection. In *NIPS*, 2007.
[11] J. He and J. Carbonell. Rare class discovery based on active learning. In *ISAIM*, 2008.
[12] K. Huang, H. Yang, I. King, and K. Lyu. Learning classifiers from imbalanced data based on biased minimax probability machine. In *CVPR*, pages II–558–II–563, 2004.
[13] T. Mitchell. *Machine Learning*. McGraw-Hill Science Engineering, 1997.
[14] D. Pelleg and A. Moore. Active learning for anomaly and rare-category detection. In *NIPS*, 2004.
[15] D. Scott. *Multivariate Density Estimation: Theory, Practice, and Visualization*. Wiley-Interscience, 1992.
[16] Y. Sun, M. Karmel, and Y. Wang. Boosting for learning multiple classes with imbalanced class distribution. In *ICDM*, pages 592–602, 2006.
[17] L. Wasserman. *All of Nonparametric Statistics*. Springer, 2005.
[18] G. Wu and E. Chang. Aligning boundary in kernel space for learning imbalanced dataset. In *ICDM*, pages 265–272, 2004.
[19] J. Wu, H. Xiong, P. Wu, and J. Chen. Local decomposition for rare class analysis. In *KDD*, pages 814–823, 2007.