Isolation Kernel and Its Effect on SVM

Kai Ming Ting
School of Engineering and
Information Technology,
Federation University,
Australia
kaiming.ting@federation.edu.au

Yue Zhu
National Key Laboratory for Novel
Software Technology,
Nanjing University,
China
zhuy@lamda.nju.edu.cn

Zhi-Hua Zhou
National Key Laboratory for Novel
Software Technology,
Nanjing University,
China
zhouzh@lamda.nju.edu.cn

ABSTRACT

This paper investigates data dependent kernels that are derived directly from data. This has been an outstanding issue for about two decades which hampered the development of kernel-based methods. We introduce Isolation Kernel which is solely dependent on data distribution, requiring neither class information nor explicit learning to be a classifier. In contrast, existing data dependent kernels rely heavily on class information and explicit learning to produce a classifier. We show that Isolation Kernel approximates well to a data independent kernel function called Laplacian kernel under uniform density distribution. With this revelation, Isolation Kernel can be viewed as a data dependent kernel that adapts a data independent kernel to the structure of a dataset. We also provide a reason why the proposed new data dependent kernel enables SVM (which employs a kernel through other means) to improve its predictive accuracy. The key differences between Random Forest kernel and Isolation Kernel are discussed to examine the reasons why the latter is a more successful tree-based kernel.

CCS CONCEPTS

• **Computing methodologies** → *Machine learning*; *Supervised learning by classification*; *Kernel methods*;

KEYWORDS

Data dependent kernel, SVM classifiers, Random Forest, Isolation Forest

ACM Reference Format:
Kai Ming Ting, Yue Zhu, and Zhi-Hua Zhou. 2018. Isolation Kernel and Its Effect on SVM. In *KDD 2018: 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, August 19–23, 2018, London, United Kingdom.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3219819.3219990

1 INTRODUCTION

The development of kernel-based methods has been hampered by the need to manually design a suitable kernel for the task at hand. A poorly chosen kernel produces poor task-specific performance. This has remained to be an outstanding issue for about two decades.

One research attempt to address this issue is to use Multiple Kernel Learning (e.g., [5]) to learn a combination of multiple data-independent kernels to reduce the risk of choosing an unsuitable kernel for the task at hand. This approach provides a good way to combine multiple kernels in either linear or nonlinear form, where a weight is assigned to each kernel. The weights can be determined in different ways, including data dependent estimations that employ either class conditional probability, structural risk minimisation, Bayesian or boosting approaches. See [9] for a comprehensive survey. However, this approach does not produce a data dependent kernel directly from data.

An alternative research direction is to adapt the similarity measurements to the task automatically from data. One interesting approach is distance metric learning [18, 20, 21] which aims to warp the space such that points of the same class are brought closer and points of different classes are stretched further apart in the new space than in the given space. The approach transforms the space such that points in the transformed space achieved the stated warped effect when measured in Euclidean distance. The transformation, i.e., the focus of the approach, is formulated as a learning process which must have access to class information.

The third direction is to produce a data dependent kernel explicitly; and one approach is Random Forest (RF) kernel [2]. However, the evaluation of this approach has been limited to some specific application and regression only [6, 14].

One of the first methods to adapt a data independent kernel function to the structure of data is through a conformal transformation[1, 19]. In the classification context, the idea is to modify a given kernel function such that the spatial distances around the boundary between two classes is enlarged (and those outside the boundary region reduced). To achieve this objective, some knowledge of the boundary is required in order to learn a data dependent kernel for a given dataset. This objective bears some resemblance to that of distance metric learning, i.e., to reduce the distance between points of the same class and increase the distance between points of different classes. The key issue of this method is the need to know the boundary (e.g., the support vectors in SVM due to the use of a data independent kernel) in order to modify the kernel.

The proposed approach differs from the abovementioned approaches in two key aspects. First, unlike approaches such as Random Forest kernel, distance metric learning and conformal transformation, the proposed approach does not rely on class information. Thus, it has wider applications to both unsupervised and supervised learning tasks. Second, unlike multiple kernel learning which aims to learn a combination of multiple data independent kernels, the proposed approach derives a data dependent kernel directly from

KDD 2018, August 19–23, 2018, London, United Kingdom

ACM ISBN 978-1-4503-5552-0/18/08…$15.00
https://doi.org/10.1145/3219819.3219990

data. The mechanism to create the data dependent kernel is simpler than all these approaches.

This paper aims to (i) develop a *data dependent kernel which adapts to the density structure of a dataset*—the characteristic of the kernel; and (ii) identify the reason why the data dependent kernel enables kernel-based algorithms such as SVM to improve its predictive accuracy.

The contributions are:

(1) Creating a new data dependent kernel, called Isolation Kernel, which needs neither learning nor class information. In contrast, existing approaches to data dependent kernel require class information and learning.
(2) Asserting a necessary property of the partitioning mechanism to produce a successful Isolation Kernel, i.e., the partitioning mechanism must produce large isolating partitions in sparse region and small isolating partitions in dense region. This property leads to the required characteristic of the kernel: two points in sparse region are more similar than two points of the same inter-point distance in dense region.
(3) Providing a reason why the proposed new data dependent kernel enables SVM (which employs a kernel through other means) to improve its predictive accuracy.
(4) Comparing the kernel with radial basis function (RBF) kernel, Laplacian kernel, multiple kernel learning and distance metric learning, in the context of SVM classifiers.

This work is a step advancement in easing the bottleneck in the development of kernel-based methods, enabling a successful kernel to be derived directly from data. It is a simpler approach than existing approaches to data dependent kernels; and has demonstrably better or comparable predictive accuracy than data independent kernels and kernels derived from distance metric learning and multiple kernel learning when applied to SVM classifiers.

The rest of the paper is organised as follows. Section 2 provides the formal definitions of Isolation Kernel, its characteristics, required partitioning mechanism, relationship to a data independent kernel, and its ability to adapt to density structure of a dataset. Section 3 presents a reason why SVM produces a better predictive accuracy using the kernel. The empirical evaluation results on real datasets are presented in Section 4. The relationship with Random Forest is given in Section 5, followed by discussion and conclusions.

2 ISOLATION KERNEL: DEFINITIONS

Let $D = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}, \mathbf{x}_i \in \mathbb{R}^d$ be a dataset sampled from an unknown probability density function $\mathbf{x}_i \sim F$. Moreover, let $\mathcal{H}_\psi(D)$ denote the set of all partitions H that are admissible under the dataset D where each isolating partition $\theta \in H$ isolates one point from the rest of the points in a random subset $\mathcal{D} \in D$, and $|\mathcal{D}| = \psi$.

Definition 2.1. For any two points $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, Isolation Kernel of $\mathbf{x}$ and $\mathbf{y}$ wrt D is defined to be the expectation taken over the probability distribution on all partitioning $H \in \mathcal{H}_\psi(D)$ that both $\mathbf{x}$ and $\mathbf{y}$ fall into the same isolating partition $\theta \in H$:

$$K_\psi(\mathbf{x}, \mathbf{y}|D) = \mathbb{E}_{\mathcal{H}_\psi(D)}[\mathbb{I}(\mathbf{x}, \mathbf{y} \in \theta \mid \theta \in H)] \quad (1)$$

where $\mathbb{I}(B)$ is the indicator function which outputs 1 if B is true; otherwise, $\mathbb{I}(B) = 0$.

In practice, Isolation Kernel would be estimated from a finite number of partitionings $H_i \in \mathcal{H}_\psi(D), i = 1, \dots, t$ as follows:

$$K_\psi(\mathbf{x}, \mathbf{y}|D) = \frac{1}{t}\sum_{i=1}^{t} \mathbb{I}(\mathbf{x}, \mathbf{y} \in \theta \mid \theta \in H_i) \quad (2)$$

Lemma 2.2. *$K_\psi(\mathbf{x}, \mathbf{y}|D)$ is a valid kernel.*

Proof. We only need to show that the matrix produced by K_ψ is a positive semi-definite (PSD) matrix.

Equation 2 can be re-expressed as follows:

$$K_\psi(\mathbf{x}, \mathbf{y}|D) = \frac{1}{t}\sum_{i=1}^{t}\sum_{\theta \in H_i} \mathbb{I}(\mathbf{x} \in \theta)\mathbb{I}(\mathbf{y} \in \theta)$$

Let $\phi_i(\mathbf{x}) = \mathbb{I}(\mathbf{x} \in \theta \mid \theta \in H_i)$; and $K_\psi^i = \phi_i^\top \phi_i$. Because K_ψ^i is in a quadratic form, it is PSD. The sum of PSD matrices, $K_\psi = \frac{1}{t}\sum_{i=1}^{t} K_\psi^i$, is also PSD. □

Provided H produces larger isolating partitions in the sparse region than those in the dense region, points of equal inter-point distance are more likely to fall in the same isolating partition in sparse region than that in dense region.

This H leads to a characteristic of K_ψ described below.

Let $\mathcal{X}_S$ and $\mathcal{X}_T$ be two subsets of points in sparse and dense regions of $\mathbf{R}^d$, respectively, i.e., the probability density $P(\mathcal{X}_S) < P(\mathcal{X}_T)$; and $\| \mathbf{x} - \mathbf{y} \|$ be the distance between $\mathbf{x}$ and $\mathbf{y}$.

Characteristic of K_ψ: $\forall \mathbf{x}, \mathbf{y} \in \mathcal{X}_S$ and $\forall \mathbf{x}', \mathbf{y}' \in \mathcal{X}_T$ such that $\| \mathbf{x} - \mathbf{y} \| = \| \mathbf{x}' - \mathbf{y}' \|$, K_ψ satisfies the following condition:

$$K_\psi(\mathbf{x}, \mathbf{y}) > K_\psi(\mathbf{x}', \mathbf{y}') \quad (3)$$

We provide empirical evidence to support this kernel characteristic in this section and the next.

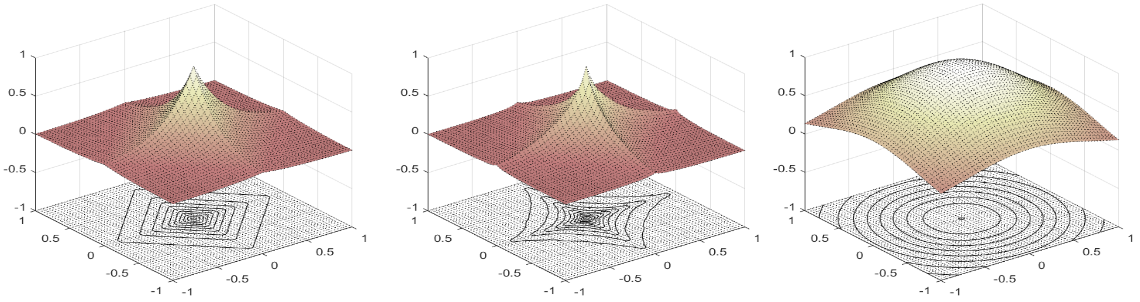
Expressed in words: the characteristic of Isolation Kernel is that **two points in sparse region are more similar than two points of equal inter-point distance in dense region**. Interestingly, this characteristic is akin to the one as judged by human, as suggested by psychologists (e.g., [10]).

2.1 The required partitioning mechanism

To obtain the above kernel characteristic, *the required property of the partitioning mechanism H is to create large partitions in sparse region and small partitions in dense region.*

Isolation methods e.g., iForest [12], are a partitioning mechanism which has the above property. An isolation method isolates every point from the rest of the points in the training set. When this is done randomly, it automatically produces large isolating partitions in sparse region and small isolating partitions in dense region.

We show in the next two subsections that when H is implemented using iForest (see Appendix), (a) K_ψ approximates well to Laplacian kernel under uniform density distribution; and (b) K_ψ measures two points to be more similar in sparse region than two points of the same inter-point distance in dense region. This is because such points in the sparse region are more likely to fall into the same isolating partition than that in dense region.



(a) Laplacian kernel

(b) Isolation Kernel

(c) RBF kernel

Figure 1: Laplacian, Isolation Kernel and RBF with reference to point (0, 0) on a 2-dimensional dataset with uniform density distribution. Parameters used: Isolation and Laplacian use the same $\psi = 256$; RBF uses $\gamma = 1$; and Isolation uses $t = 10000$.

2.2 K_ψ under uniform density distribution

2.2.1 Completely random trees in Breiman's analysis. Interestingly, Breiman [2] described a kernel, created based on *completely random trees which are generated without data.* For $d \geq 5$ and the number of leaf nodes $T \leq \exp(d/2)$ [2], the kernel is said to be approximated[1] by a Laplacian kernel:

$$L(\mathbf{x}, \mathbf{y}) = \exp(-\lambda \sum_{j=1}^{d} |x_j - y_j|) \qquad (4)$$

where $\mathbf{x} =< x_1, \cdots, x_j, \cdots, x_d >$; and λ determines the sharpness of the kernel.

It is interesting to note that the data independent completely random trees used in the analysis [2] is equivalent to data dependent iForest when the data is uniform density distribution. Therefore, this Laplacian kernel approximates Isolation Kernel K_ψ very well under the uniform density distribution.

2.2.2 Laplacian kernel in a new light. Let ψ be the number of leaf nodes in a (data independent) completely random tree, Breiman's analysis [2] has λ in the Laplacian kernel relates to ψ and d as: $\lambda = \frac{\log(\psi)}{d}$.

Though not shown in [2], the Laplacian kernel can be re-expressed in terms of ψ as:

$$L_\psi(\mathbf{x}, \mathbf{y}) = \exp(-\frac{\log(\psi)}{d} \sum_{j=1}^{d} |x_j - y_j|) = \psi^{-\frac{1}{d} \sum_{j=1}^{d} |x_j - y_j|} \qquad (5)$$

This new expression has an interesting meaning in relation to Isolation Kernel K_ψ. That is, when K_ψ is approximately equivalent to L_ψ under uniform density distribution, ψ—the number of training points used to train each isolation tree—has the similar sharpness interpretation as λ (of Laplacian kernel shown in Equation 4), i.e., the larger ψ is, the sharper the L_ψ and K_ψ distributions.

In summary, Isolation Kernel K_ψ that employs iForest is well approximated by a variant of Laplacian kernel L_ψ under uniform density distribution because the completely random trees generated by iForest is exactly the same as data independent completely random trees used in Breiman's analysis [2] under this distribution.

Hereafter we refer to Laplacian kernel as defined in L_ψ above.

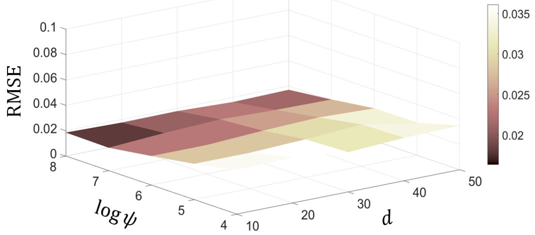


Figure 2: RMSE between Isolation Kernel and Laplacian kernel on uniform density distribution datasets with different values of d (number of dimensions) and ψ.

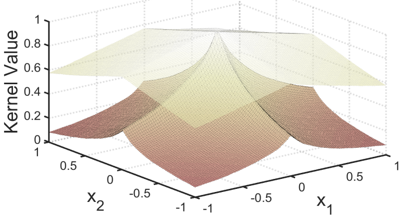
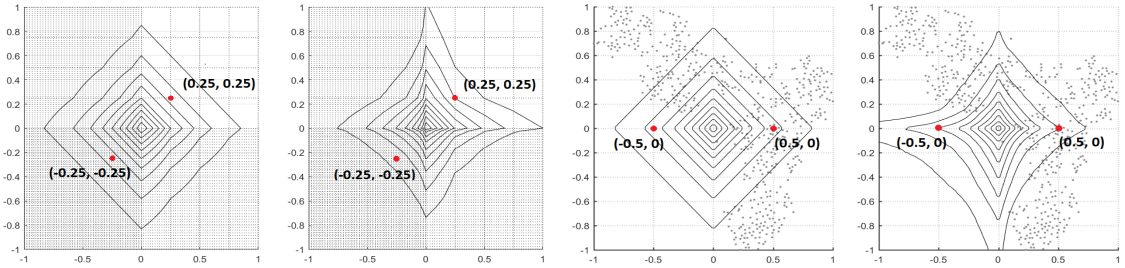


Figure 3: Distribution of K_ψ derived from iForest with $\psi = 8$ (top) and 1024 (bottom) on the same uniform density distribution. Higher ψ produces sharper K_ψ distribution.

Figure 1 compares the contours of the three kernels: Laplacian, Isolation and RBF wrt point $(0, 0)$ on a dataset with uniform density distribution. As can be observed, Isolation Kernel is very similar to Laplacian kernel. The contours of both Isolation Kernel and Laplacian Kernel have diamond-like shapes, which are much sharper than that of RBF kernel having circular shapes.

In order to further validate the relation between Isolation Kernel and Laplacian kernel, we randomly generate d-dimensional data under a uniform distribution $U \sim [-1, 1]^d$, where d varies in $\{10, 20, 30, 40, 50\}$. Isolation Kernels and Laplacian Kernels are constructed using these datasets to examine their differences, where $\psi \in \{16, 32, 64, 128, 256\}$. The RMSE (root-mean-square error) between Isolation Kernel and Laplacian kernel is calculated to measure the difference in each combination of $d \times \psi$. Figure 2 shows that the RMSE between the two kernels are small for all values of ψ and d. This shows that Isolation Kernel is a good approximation to Laplacian kernel under uniform density distribution.

[1] Issues of relating this approximation to RF kernel are discussed in Section 5.2.



(a) Laplacian (b) Isolation (c) Laplacian (d) Isolation

Figure 4: Laplacian Kernel versus Isolation Kernel: Contours with reference to $(0, 0)$. (a) & (b) - Uniform distributions of different densities in four quadrants: Highest in bottom-left, lowest in top-right, and medium in top-left & bottom-right; (c) & (d) - A 2-dimensional visualization of the multi-dimensional Forest dataset, transformed by t-SNE [17].

2.2.3 *K_ψ in terms of δ-distance.* Isolation Kernel K_ψ can be redefined as approximating the probability distribution of two points of δ-distance falling into the same isolating partition, i.e., $K_\psi(\mathbf{x}, \cdot), \forall_\mathbf{y} \; \| \mathbf{x} - \mathbf{y} \| = \eta\delta$, where $\eta \geq 0$ is an integer indicating the multiples of δ from $\mathbf{x}$.

Re-express $K_\psi(\mathbf{x}, \cdot)$ of δ-distance as $K_\psi(\mathbf{x}|\delta)$ for all points in a grid G of cell length δ which have $\eta\delta$ distance from $\mathbf{x} \in G$:

$$K_\psi(\mathbf{x}|\delta) \approx \psi^{-\eta\delta} \tag{6}$$

where ψ is the sharpness parameter; $\eta\delta = \frac{1}{d}\sum_{j=1}^{d} |x_j - y_j|$; and $K_\psi(\mathbf{x}|\delta)$ is inversely proportional to δ and η.

Figure 3 shows the effect of ψ under a uniform density distribution: the higher ψ is, the sharper the K_ψ distribution.

2.3 K_ψ is adaptive to different densities

Recall that ψ is the sharpness parameter of Laplacian kernel L_ψ and also its approximation K_ψ. In contrast to L_ψ which is data independent, K_ψ has the following data dependent characteristic: **K_ψ decreases at a slower rate in sparse region than in dense region, given a fixed ψ. In other words, the distribution of K_ψ is sharper in dense region than in sparse region.**

Figure 4 compares the contours of the two kernels under two non-uniform distributions. These examples show that Laplacian kernel has the same contour irrespective of the data distribution. In contrast, Isolation Kernel adapts its contour to the local data distribution, i.e., from the peak at $(0, 0)$, the contour decreases at a slower rate in the sparsest region (top right quadrant) than in dense regions (the other three quadrants). This leads to the kernel characteristic: two points in sparse region are more similar than two points of equal inter-point distance in dense region. This is demonstrated by the following points in Figure 4: Isolation Kernel has a higher value at $(0.25, 0.25)$ in the sparsest region than that at $(-0.25, -0.25)$ in the densest region, where both points have the same distance from the origin (see Figure 4(b)). Similarly in Figure 4(d), Isolation Kernel has a higher value at $(0.5, 0)$ than that at $(-0.5, 0)$. In each of the above cases, the first point (in the sparse region) is more similar to the point at the origin than the second point (in the dense region). Note that the corresponding points in Laplacian kernel, shown in Figures 4(a) & 4(c), have the same same similarity, as long as the distances are the same.

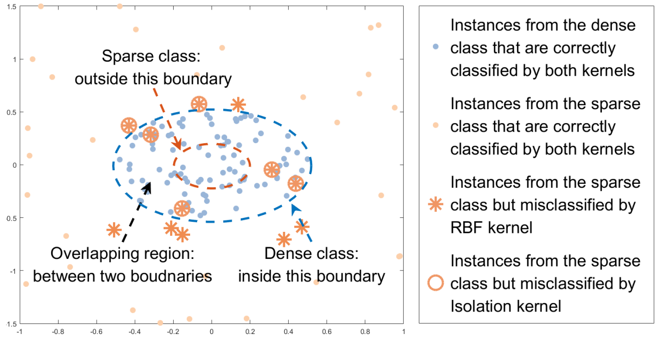


Figure 5: An example plot of data_20.

The ability to adapt to the density structure of a dataset is the key advantage of Isolation Kernel over data independent kernels.

This characteristic can be interpreted in terms of δ-distance as: *the K_ψ similarity of two points of δ-distance is higher in sparse region than that in dense region.*

In contrast, none of the existing kernels have the abovementioned characteristic. RF kernels and distance metric learning are heavily influenced by class information. No statement can be made in terms of δ-distance. Data independent kernels have uniform distribution in terms of δ-distance i.e., every data independent kernel has the same similarity for any two points of δ-distance everywhere in the feature space (i.e., it is translation invariant).

3 K_ψ IN ACTION IN SVM CLASSIFICATIONS

Knowing Isolation Kernel's ability to adapt to the density structure, we examine its effectiveness in dealing with datasets which have classes of varied densities in the SVM classification context.

Synthetic datasets. The datasets used have the density ratio of positive (dense) class to negative (sparse) class increased from 1:1 to 20:1. Specifically, we generate 500 2-dimensional points, with an equal number of positive points and negative points. Note that all points are uniformly distributed in each class. We vary the areas of different classes to control the density ratio. Besides, we also control the overlapping region between the two classes in order to make sure that negative class has a certain number of points. Figure 5 gives an example plot with density ratio 20:1.

Table 1: SVM classifier results. Each result is an average over 10 independent runs. The evaluation methodology is the same as stated in Section 4.1.

Data	Algorithm	Accuracy	FPR	FNR
data_1	SVM_RBF	0.973±0.008	0.008±0.002	0.046±0.016
	SVM_Lap	0.960±0.002	0.031±0.003	0.048±0.004
	SVM_IK	0.966±0.006	0.032±0.009	0.036±0.009
data_10	SVM_RBF	0.959±0.008	0.082±0.017	0.000±0.000
	SVM_Lap	0.965±0.002	0.070±0.002	0.000±0.000
	SVM_IK	0.967±0.005	0.064±0.010	0.002±0.002
data_20	SVM_RBF	0.953±0.008	0.094±0.015	0.000±0.000
	SVM_Lap	0.966±0.002	0.067±0.004	0.000±0.000
	SVM_IK	0.971±0.005	0.056±0.010	0.001±0.003

Table 1 presents the accuracy, false positive rate, and false negative rate of SVM with Isolation, Laplacian and the widely applied RBF kernels. It shows that Isolation Kernel always has lower false positive rates than RBF and Laplacian for datasets having density ratio > 1. This means that SVM using Isolation Kernel resulted a smaller number of points belonging to the sparse class being incorrectly classified to dense class than SVM using either RBF or Laplacian kernels. As the density ratio increases, all kernels have zero or close to zero false negative rates. Figure 5 demonstrates the abovementioned scenario using the data distribution of data_20. The results show that SVM with Isolation Kernel has higher accuracy than SVM with either RBF or Laplacian kernel when the density ratio between classes is high in the overlapping region.

Real datasets. Because we do not know the nature of the class overlapping region in real datasets, we focus on the points which are misclassified by SVM classifiers. For each misclassified point, its nearest neighbour of a different class is identified. Then, the density[2] is computed for each of these points. Let ρ_i be the density of the i-th point, and $\bar{\rho}_i$ be the density of the i-th point's nearest neighbor of a different class. We report mean maximum density—the average of $\max(\rho_i, \bar{\rho}_i)$; mean minimum density—the average of $\min(\rho_i, \bar{\rho}_i)$; and mean density ratio—the average of $\max(\rho_i, \bar{\rho}_i)/\min(\rho_i, \bar{\rho}_i)$. Table 2 shows these results from four real datasets.

It is interesting to note that points which are misclassified by Laplacian only has the highest density ratio; and by Isolation only has the lowest ratio on all these datasets. This is despite the fact that Isolation has a smaller number of misclassified points than Laplacian on two datasets (i.e., Wilt and German); but larger on the other two datasets. This result indicates that, in SVM classifications, Isolation Kernel produces better predictive accuracy than Laplacian kernel in regions where class density varies hugely. The reverse is true in regions where class density ratio is low.

Summary for Sections 2 and 3

(1) The partitioning requirement to create Isolation Kernel is: sparse region has large isolating partitions and dense region has small isolating partitions. We show that there is one successful partitioning method based on trees to create Isolation Kernel. It is a completely random partitioning method called Isolation Forest which automatically creates isolating partitions of the required size, depending on data distribution, where no partitioning selection criterion is required.

(2) Isolation Kernel K_ψ employing the iForest implementation approximates the data independent L_ψ kernel under uniform density distribution only. K_ψ adapts the L_ψ kernel to the density structure of a given dataset such that two points of δ-distance are more similar in sparse region than that in the dense region. In contrast, for the L_ψ kernel, two points of δ-distance have the same similarity everywhere in the space—all data independent kernels have this characteristic.

(3) In the overlapping region with high density ratio between classes, Isolation Kernel has lower error rate than data independent Laplacian kernel, when used in SVM classifications. The reverse is true in regions with low density ratio.

Table 2: Densities and ratios of SVM misclassified points. The number in bracket is the number of misclassified testing points, obtained from one run of a 80/20 split into training and testing subsets.

Dataset	Misclassified by	Density		Ratio
		Maximum	Minimum	
Wilt	Laplacian (37)	0.5596	0.1947	7.2
	Both kernels (15)	0.4952	0.2649	2.3
	Isolation (6)	0.2988	0.2415	1.5
German	Laplacian (9)	0.0626	0.0010	62.6
	Both kernels (37)	0.1053	0.0245	7.9
	Isolation (6)	0.0715	0.0275	1.8
Spam	Laplacian (14)	0.0991	0.0169	6.4
	Both kernels (48)	0.0506	0.0120	4.8
	Isolation (15)	0.0330	0.0169	2.2
Qsar	Laplacian (4)	0.2618	0.0218	16.9
	Both kernels (27)	0.1606	0.0356	6.3
	Isolation (6)	0.1579	0.0340	4.5

4 SVM EXPERIMENTS ON REAL DATASETS

In this section, we conduct an extensive experimental study using SVM classifiers on real datasets to further validate the effectiveness of Isolation Kernel.

4.1 Experimental setup

We conduct experiments on a total of 23 datasets: 14 UCI datasets [11] with small to median size (less than 5000 points); and 2 UCI datasets and 2 LIBSVM datasets with large data sizes (between 8000 and 50000 points), In addition, 5 UCI multi-class datasets are also used, where the number of points ranges from 168 to 10000. These datasets have the number of classes ranges from 3 to 100. All 23 datasets are available on public websites[3]. The properties of these datasets are provided in the first four columns of Table 3.

[2] To estimate the density of a point on a dataset, we use Kernel Density Estimation, where a multi-variate Gaussian is applied as the kernel function, and the bandwidth for dimension i [15] is set to $b_i = \sigma_i \left\{ \frac{4}{(d+2)n} \right\}^{1/(d+4)}$, where σ_i is the standard deviation of dimension i, and n is the dataset size.

[3] UCI datasets can be downloaded from https://archive.ics.uci.edu/ml/datasets.html, and LIBSVM datasets from https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html

Table 3: SVM results in accuracy: Comparing RBF kernel, GMML, SimpleMKL, Laplacian kernel and Isolation Kernel. GMML and SimpleMKL have "out of memory" error on 2 datasets; SimpleMKL cannot return any result within 24 hours on 5 datasets. The first and second subsets consists of 2-class datasets having less than and more than 5000 points, respectively; the third subset consists of multi-class datasets. The numbers following ± are standard errors.

	#inst	#fea	#class	RBF	GMML	SimpleMKL	Laplacian	Isolation
GPS	163	6	2	0.855±0.018	0.855±0.028	0.867±0.018	0.836±0.026	0.842±0.022
Heart	270	13	2	0.833±0.027	0.844±0.034	0.826±0.035	0.852±0.024	0.852±0.025
Breast	277	9	2	0.750±0.018	0.768±0.016	0.746±0.009	0.761±0.012	0.771±0.015
Ionosphere	351	34	2	0.949±0.006	0.941±0.020	0.958±0.009	0.946±0.014	0.955±0.011
Vote	435	16	2	0.943±0.013	0.949±0.010	0.940±0.007	0.956±0.008	0.961±0.013
ILPD	583	11	2	0.716±0.004	0.726±0.008	0.720±0.003	0.713±0.002	0.720±0.008
WBC	683	9	2	0.972±0.003	0.974±0.004	0.969±0.004	0.977±0.003	0.975±0.007
Austra	690	14	2	0.857±0.016	0.864±0.014	0.854±0.015	0.859±0.017	0.871±0.011
German	1000	24	2	0.753±0.008	0.754±0.006	0.705±0.003	0.759±0.006	0.767±0.012
Parkinson	1040	28	2	0.999±0.001	1.000±0.000	0.998±0.002	1.000±0.000	1.000±0.000
QSAR	1055	42	2	0.880±0.007	0.870±0.005	0.864±0.007	0.873±0.009	0.869±0.012
Messidor	1151	19	2	0.690±0.021	0.734±0.014	0.673±0.023	0.692±0.019	0.694±0.018
Spam	4141	58	2	0.932±0.004	0.912±0.003	> 24 hours	0.944±0.002	0.940±0.003
Wilt	4839	5	2	0.946±0.039	0.984±0.001	> 24 hours	0.946±0.021	0.985±0.014
Mushrooms	8124	112	2	1.000±0.000	1.000±0.000	> 24 hours	1.000±0.000	1.000±0.000
Phishing	11055	30	2	0.965±0.001	0.959±0.003	> 24 hours	0.968±0.002	0.967±0.001
a8a	32561	123	2	0.846±0.003	memory error	memory error	0.846±0.003	0.847±0.003
IJCNN	49990	22	2	0.980±0.001	memory error	memory error	0.978±0.001	0.978±0.002
Urban	168	147	9	0.835±0.022	0.841±0.029	0.882±0.027	0.847±0.033	0.841±0.020
Air	359	64	3	0.956±0.010	0.933±0.012	0.942±0.013	0.967±0.008	0.967±0.013
Forest	523	27	4	0.895±0.014	0.910±0.011	0.964±0.011	0.891±0.015	0.912±0.011
Vowel	528	10	11	0.983±0.010	0.979±0.010	0.982±0.007	0.979±0.010	0.989±0.007
Corel	10000	67	100	0.364±0.005	0.387±0.005	> 24 hours	0.457±0.003	0.466±0.004
Isolation has #wins/#draws/#losses				19/1/3	14/3/4	11/1/4	13/5/5	-

We compare Isolation Kernel with two kernels and two methods using SVM in classification tasks:

(1) SVM_RBF: SVM with RBF kernel.
(2) GMML+SVM_RBF: Geometric mean metric learning (GMML) [21] transforms feature space such that points of the same class are brought closer, and points of different classes are pulled further apart. Then SVM with RBF kernel is applied on the transformed data.
(3) SimpleMKL [13] learns a combination of multiple user-defined data independent kernels.
(4) SVM_Laplacian: SVM with Laplacian kernel.
(5) SVM_Isolation: SVM with Isolation Kernel.

For brevity, the above SVM classifiers are denoted as RBF, GMML, SimpleMKL, Laplacian and Isolation, respectively.

Table 4: Search ranges of parameters used.

	Description	Candidates
RBF	RBF parameter	$\gamma \in \{2^m \| m = -10, \ldots, 4, 5\}$
GMML	step length of geodesic	$t \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$
	RBF parameter	$\gamma \in \{2^m \| m = -10, \ldots, 4, 5\}$
Laplacian	ψ: sampling size	$\psi \in \{2^m \| m = 2, 3, \ldots, 12\}$
Isolation		

Parameter settings used in the experiments are listed in Table 4. The parameter settings are selected via 5-fold cross-validation on the training set. SVM's parameter C is fixed as 1 (which is the default setting, and not very sensitive). This is the same for all kernels and methods compared. For Isolation Kernel, the default settings of iForest [12] are used: the number of trees is 100, and maximum tree height $h = \log_2(\psi)$.

SimpleMKL employs 16 RBF kernels which have $\gamma \in \{2^m | m = -10, \ldots, 4, 5\}$. Polynomial kernels only or in combination with RBF kernels were also attempted; but the results were worse than the results reported in the next section. Unlike SVM with a single kernel, SimpleMKL has no parameters that require search.

The SVM implementation of LIBSVM [4] is used. Matlab 2017a is applied for the implementation; and each experiment is conducted on a single core of Intel Xeon(R) E5-2620 with 32G memory.

Rather than using the pre-computed kernel matrix, as required by LIBSVM in order to use kernels other than data independent kernels, we have modified LIBSVM to enable partial kernel matrix computation when Isolation Kernel is used.

We report the averaged accuracy and standard error of 5 independent runs on each dataset, where each run is using a 80/20 split of the original dataset into training and testing subsets.

4.2 Experimental results

(a) Accuracy. Table 3 reports the results of SVM using different kernels and methods. Isolation is better than RBF on 14 out of 18 binary classification datasets and on all 5 multi-class datasets. It is

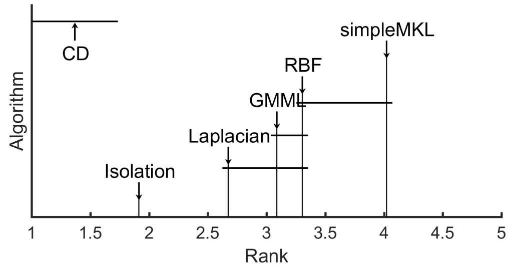


Figure 6: Nemenyi Test at 0.05 significance level. If two algorithms are connected by a CD (critical difference) line, then there is no significant difference between them.

Table 5: Runtime (second) for training + testing.

	RBF	GMML	SimpleMKL	Laplacian	Isolation
IJCNN	54+14	mem err	mem err	56+15	4297+716
Corel	26+7	23+5	>24 hrs	27+9	3975+810

better than GMML on 14 out of 21 datasets, ties on 3 and loses on 4 It is better than SimpleMKL on 11 out of 16 datasets, ties on 1, and loses on 4. It is better than Laplacian on 13 out of 23 datasets, ties on 5, and loses on 5.

The above result shows that Isolation Kernel is an effective data dependent kernel, which adapts well to local data distributions.

Figure 6 shows the result of a significance test called Nemenyi Test [7], which is a post-hoc test based on Friedman test. Isolation is significantly better than the other contenders at 0.05 significance level[4]. GMML and simpleMKL, when used in SVM classifiers, do not perform better than simply using the data independent RBF or Laplacian kernels.

(b) Runtime. Table 5 presents the runtimes on the two largest datasets of the 2-class and multi-class tasks for the three kernels. Isolation Kernel works much slower than the other two kernels. This is because 100 trees are used in calculation. Because the trees are completely random trees, each works independently, they can be easily parallelized to reduce the runtime.

Note that GMML and SimpleMKL could not complete on IJCNN because of their large memory requirements; and SimpleMKL could not complete within 24 hours on Corel.

5 RELATION WITH RF KERNEL

5.1 Breiman's explanation of RF classifier behaviour in terms of kernel

By defining RF kernel of two points as the average number of shared leaf nodes of Random Forest, Breiman [2] explains the behavior of the RF classifier in terms of this kernel: as a means to locate boundaries, which trade-offs between the kernel's 'symmetry' and 'skewness'. They correspond to correlation and strength of the ensemble, respectively. 'Symmetry' kernel is produced from completely random splits; and 'skewed' kernel is generated from splits which favour pure nodes, which has high strength or classification accuracy. This skewness is also conjectured to enable nonlinear classification [2].

This interpretation has inspired others to use RF as a similarity measure in distance-based neighbourhood methods. Breiman and Cutler [3] first describe two methods to generate RF similarity: one generates RF similarity from a labelled dataset; and the other from an unlabelled dataset. Shi and Horvath [14] applied the second method for tumor discovery using the RF similarity in a distance-based clustering algorithm. Davis and Ghahramani [6] attempted to generalise RF kernel to use different partitioning methods; however, the evaluation is conducted for the regression tasks only.

The applications of RF similarity have been limited for two reasons. First, the theory requires that the trees are trained using bootstrap samples and grown to the largest size. For large datasets, the similarity generation process is prohibitively expensive in terms of time and space. Second, to apply a supervised learning method to unsupervised learning, the RF similarity generation process demands a second synthetic dataset, to act as a second class as opposed to the given unlabelled dataset as the first class (see [3, 14]). Only then, a RF classifier can be generated. This further increases the time and space complexities of the entire process.

5.2 The differences between Isolation Forest and Random Forest as kernels

This section answers two questions to uncover the differences between Isolation Forest and Random Forest as kernels.

a) How does the 'symmetry' of RF kernel differ from the characteristic of Isolation Kernel, when both are using completely random trees?

They are not the same completely random trees. First, completely random trees cannot be generated by RF and are not actually used in RF. For ease of analysis, completely random trees were attempted as a proxy to explain one aspect of RF's behaviour, i.e., correlation [2]. Second, relying on class information, each completely random tree is grown to the largest where each leaf node has one class only (not necessarily with one point); and it requires to use the entire given dataset. As a result, each completely random tree (though data dependent) does not necessarily produce large partitions in sparse region and small partitions in dense region.

In contrast, each isolation tree isolates each point from the rest of the points in a small data subset, sampled from the given dataset. This is done without the class information, i.e., each isolation tree partitions the space purely based on data distribution.

In other words, even under uniform density distribution, the completely random trees generated by iForest and the ones from Breiman's analysis [2] are not the same; and Random Forest does not produce trees which have the same property as that produced by iForest. With reference to the definition, RF kernel complies with Equation 1, but does not satisfy the condition in Equation 3 because of its class-dependency.

b) How do the above differences affect their kernel characteristics?

According to Breiman's conjecture [2], RF kernel aims to track the sign of the margin of x (defined as $P(+1|x) - P(-1|x)$, where +1 and -1 are the two class labels). In contrast, Isolation Kernel

[4] This test is conducted by setting the datasets/methods which have no results to zero accuracy. The conclusion is the same even by eliminating 7 such datasets in the test.

Table 6: SVM results: Comparing two implementations of RF kernel with Isolation Kernel. The two largest datasets are not included as both RF kernels have runtime > 24 hours.

	RF_DG	RF_B	Isolation
GPS	0.842±0.026	0.788±0.086	0.842±0.022
Heart	0.804±0.032	0.630±0.099	0.852±0.025
Breast	0.732±0.020	0.700±0.018	0.771±0.015
Ionosphere	0.910±0.007	0.913±0.008	0.955±0.011
Vote	0.963±0.006	0.966±0.005	0.961±0.013
ILPD	0.677±0.026	0.627±0.076	0.720±0.008
WBC	0.972±0.001	0.965±0.005	0.975±0.007
Austra	0.868±0.009	0.772±0.096	0.871±0.011
German	0.756±0.007	0.381±0.096	0.767±0.012
Parkinson	1.000±0.000	1.000±0.000	1.000±0.000
QSAR	0.856±0.009	0.739±0.129	0.869±0.012
Messidor	0.651±0.010	0.674±0.016	0.694±0.018
Spam	0.943±0.005	0.776±0.168	0.940±0.003
Wilt	0.983±0.002	0.945±0.038	0.985±0.014
Mushrooms	1.000±0.000	1.000±0.000	1.000±0.000
Phishing	0.966±0.002	0.947±0.024	0.967±0.001
Urban	0.818±0.029	0.824±0.031	0.841±0.020
Air	0.939±0.023	0.794±0.150	0.967±0.013
Forest	0.884±0.011	0.890±0.012	0.912±0.011
Vowel	0.898±0.024	0.596±0.183	0.989±0.007
Corel	0.316±0.070	0.257±0.098	0.466±0.004
#w/#d/#l	16/4/2	18/2/1	-

aims to measure similarity of two points such that points are more similar in sparse region than two points of the same inter-point distance in dense region. Note that class has no role in the similarity measurement of Isolation Kernel.

Under uniform density distribution, Isolation Kernel approximates well Laplacian kernel. RF Kernel is a rough approximation to Laplacian kernel because there are three inconsistencies in using Laplacian kernel to approximate RF kernel, even under uniform density distribution. First, the data independent completely random trees used in the analysis [2] are not a good approximation to RF trees because Random Forest cannot produce completely random trees, and RF trees are classification trees which are generated based on labelled data. Second, it does not make sense to RF kernel if the data has no class information; yet the Laplacian kernel requires no class information. Third, to make it consistent with the requirement of RF of using largest trees, Breiman [2] has advocated the Laplacian kernel to be sharper (having larger λ) the better. However, like any data independent kernel, λ of the Laplacian kernel is to be tuned for a given dataset.

5.3 Comparison with two implementations of RF kernel

In addition to Breiman's suggestion of using the largest trees [2], we also use the suggestion from Davis and Ghahramani [6] which randomly selects a height level ℓ_i between level 1 and the maximum height of tree i; and the similarity measurement is based on level ℓ_i for tree i. We denote the above implementations as RF_B and RF_DG, respectively.

Table 6 shows the comparison results. Isolation Kernel is significantly better than both implementations of RF kernel on many datasets, e.g., Corel, Vowel, Air Urban, ILPD, Ionosphere, Breast and Heart. On the only two datasets (Vote and Spam) RF is better, the difference in accuracy is marginal.

While RF_DG generally performs better than RF_B, they are both generally worse than data independent RBF and Laplacian kernels (wrt the results shown in Table 3), e.g., in comparison with RBF, RF_DG wins on 7 datasets, loses on 12 and draws on 2.

6 DISCUSSION

Why do existing methods, such as RF kernel and distance metric learning (which exploit class information), perform worse than Isolation Kernel (which has no access to class information) in classification tasks? We conjecture that a possible issue is due to these methods' over-reliance on the class information provided in the training set. This is equivalent to the overfitting effect in building a classifier. Class noise can easily upset this reliance.

Both Isolation Kernel and conformal transformation [1] adapt a (given or presumed) data-independent kernel to the structure of a dataset. But, the data dependent methodology employed determines the kind of structure of a dataset an approach adapts to: Isolation Kernel adapts to the density structure; and conformal transformation adapts to the class distribution.

Englund and Verikas [8] proposes a modified kernel based on the path between the two points under measurement in each tree in RF. Their evaluation using SVM, though small in scope, has showed some promise.

In order to produce a maximum margin classifier based on SVM, the data dependent similarity must be positive semi-definite (PSD). Mass-based similarity [16] has a similar characteristic as that of Isolation Kernel, but it is not PSD and it cannot be readily converted to PSD because its self-similarity is not constant.

6.1 Meaning of 'Data-dependent'

In the literature, the term 'data-dependent' has been used to mean different things. In the context of multiple kernel learning (e.g., [9]), the term means using a dataset to learn a weight for each user-defined data-independent kernel, in a (linear or non-linear) combination of multiple kernels. In the context of distance metric learning [20], the term means the use of class information and the training set to learn the metric. In the context of conformal transformation [1], the term means modifying a data independent kernel to the class distribution of the data. Like distance metric learning, class information in the data plays a key role here. Similarly, RF kernel [2] produces a classifier from class-labelled data.

Here, the term 'data dependent' means data distribution dependent, specifically, not knowing the class information. The data distribution dependency is the main contributor in producing a data dependent kernel. In addition, the term has no relation to learning, i.e., no learning is involved. These are the key differences from the term's usage in all the above contexts.

6.2 Future work

To strengthen the appeal of Isolation Kernel, the characteristic stated in Section 2 shall have a formal proof.

It is possible to modify the current implementation of Isolation Forest such that it produces non-axis-parallel splits rather than axis-parallel splits. This will yield a data independent kernel other than L_ψ under uniform density distribution.

In a broader perspective, it is interesting to explore Isolation Kernels using non-tree-based implementations and identify their corresponding data independent kernels under uniform distribution.

The time cost of Isolation Kernel is high in comparison with data-independent kernels. How to reduce this cost is another interesting research question.

We have identified that datasets having classes with varied densities in the class-overlap regions are the scenarios in which Isolation Kernels are a better choice than data independent kernels. A formal analysis may reveal other contributing factors.

Indeed, the contour of Isolation Kernel (shown in Figure 1) is more similar to that of ℓ_p for $0 < p < 1$. A good approximation, as provided by Breiman [2] for Laplacian kernel, is required to substantiate it.

This paper has focused on SVM on classification tasks only. The Isolation Kernel itself is generic and can be applied to any data mining methods/tasks which require a similarity/dissimilarity measure. Domains which have classes of varied densities shall receive more attention.

7 CONCLUDING REMARKS

Viewing at the highest level, RF Kernel [2] provides the first piece of the zigsaw puzzle of our understanding of tree-based kernels for kernel-based methods. This paper contributes two important pieces of the puzzle, i.e., the space partitioning requirement and the characteristic of Isolation Kernel. This understanding enables different partitioning mechanisms to be used that are based on data distribution only, which need neither class information nor learning to be classifier. We show that iForest is one such mechanism.

On the surface, iForest may be considered as a special case of RF, i.e., when the latter is completely random, which was never advocated to be used by Breiman [2]. This paper unveils the important differences between RF and iForest which determine not only the tree structure requirement, but also the kernel characteristic, besides the obvious (but often overlooked) difference that iForest is unsupervised and RF is supervised.

It is recommended to use Isolation Kernel for SVM when a dataset has hugely varied densities between classes in the overlap region. When the densities between classes are approximately the same, a Laplacian kernel will do better.

ACKNOWLEDGEMENTS

This material is based upon work partially supported by the Air Force Office of Scientific Research, Asian Office of Aerospace Research and Development (AOARD) under award number: FA2386-17-1-4034 and 111 Project (B14020) (Kai Ming Ting); and the National Key R&D Program of China (2018YFB1004300) and the Collaborative Innovation Center of Novel Software Technology and Industrialization (Zhi-Hua Zhou). Comments from the reviewers have helped to improve the presentation of this paper.

REFERENCES

[1] Shun-Ichi Amari and Si Wu. Improving support vector machine classifiers by modifying kernel functions. *Neural Network*, 12(6):783–789, 1999.

[2] Leo Breiman. Some infinity theory for predictor ensembles. *Technical Report 577. Statistics Dept. UCB.*, 2000.

[3] Leo Breiman and A. Cutler. Random Forests Manual v4.0. *www.stat.berkeley.edu/~breiman/Using_random_forests_v4.0.pdf*, 2003.

[4] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011.

[5] Corinna Cortes, Marius Kloft, and Mehryar Mohri. Learning kernels using local rademacher complexity. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, pages 2760–2768, 2013.

[6] Alex Davies and Zoubin Ghahramani. The random forest kernel and creating other kernels for big data from random partitions. *arXiv:1402.4293*, 2014.

[7] Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.

[8] Cristofer Englund and Antanas Verikas. A novel approach to estimate proximity in a random forest: An exploratory study. *Expert Systems with Applications*, 39 (17):13046–13050, 2012.

[9] Mehmet Gönen and Ethem Alpaydin. Multiple kernel learning algorithms. *Journal Machine Learning Research*, 12:2211–2268, 2011.

[10] Carol L. Krumhansl. Concerning the applicability of geometric models to similarity data: The interrelationship between similarity and spatial density. *Psychological Review*, 85(5):445–463, 1978.

[11] M. Lichman. UCI machine learning repository, 2013. URL http://archive.ics.uci.edu/ml.

[12] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *Proceedings of the IEEE International Conference on Data Mining*, pages 413–422, 2008.

[13] Alain Rakotomamonjy, Francis R Bach, Stéphane Canu, and Yves Grandvalet. SimpleMKL. *Journal of Machine Learning Research*, 9(Nov):2491–2521, 2008.

[14] Tao Shi and Steve Horvath. Unsupervised learning with random forest predictors. *Journal of Computational and Graphical Statistics*, 15(1):118–138, 2006.

[15] Bernard W Silverman. *Density estimation for statistics and data analysis*, volume 26. CRC press, 1986.

[16] Kai Ming Ting, Ye Zhu, Mark Carman, Yue Zhu, and Zhi-Hua Zhou. Overcoming key weaknesses of distance-based neighbourhood methods using a data dependent dissimilarity measure. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1205–1214, 2016.

[17] Laurens van der Maaten and Geoffrey E. Hinton. Visualizing high-dimensional data using t-SNE. *Journal of Machine Learning Research*, 9(2):2579–2605, 2008.

[18] Kilian Q. Weinberger and Lawrence K. Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10 (2):207–244, 2009.

[19] Si Wu and Shun-Ichi Amari. Conformal transformation of kernel functions: A data-dependent way to improve support vector machine classifiers. *Neural Processing Letters*, 15(1):59–67, 2002.

[20] Eric P. Xing, Andrew Y. Ng, Michael I. Jordan, and Stuart Russell. Distance metric learning, with application to clustering with side-information. In *Proceedings of the 15th International Conference on Neural Information Processing Systems*, pages 521–528, 2002.

[21] Pourya Zadeh, Reshad Hosseini, and Suvrit Sra. Geometric mean metric learning. In *International Conference on Machine Learning*, pages 2464–2471, 2016.

APPENDIX: ISOLATION FOREST (IFOREST)

Here, we provide the pertinent details of Isolation Forest [12] which generates an ensemble of completely random trees, where each tree has the required property mentioned in Section 2.1.

Given a training sample of size ψ, a completely random tree randomly selects an attribute and its split point with uniform distribution; and splits the training sample accordingly into two subsets. This is done at each internal node of the tree recursively until every point is isolated from the rest of the points in the training sample. The final tree has ψ leaf nodes.

Each leaf node is thus an isolating partition. The isolating partitions are large in sparse region (as a result of a few splits); and they are small in dense region (because more splits are required in order to isolate a point in dense region). These isolating partitions are generated via a randomised process, without any attribute selection criteria—in contrast to that used to build classification trees.