OUTLIER REMOVAL FOR ENHANCING KERNEL-BASED CLASSIFIER VIA THE DISCRIMINANT INFORMATION

Thee Chanyaswad *Mert Al* *S. Y. Kung*

Princeton University
Princeton, NJ, USA

ABSTRACT

Pattern recognition on big data can be challenging for kernel machines as the complexity grows with the squared number of training samples. In this work, we overcome this hurdle via the *outlying data sample removal* pre-processing step. This approach removes less-informative data samples and trains the kernel machines only with the remaining data, and hence, directly reduces the complexity by reducing the number of training samples. To enhance the classification performance, the outlier removal process is done such that the discriminant information of the data is mostly intact. This is achieved via the novel *Outlier-Removal Discriminant Information (ORDI)* metric, which measures the contribution of each sample toward the discriminant information of the dataset. Hence, the ORDI metric can be used together with the simple filter method to effectively remove insignificant outliers to both reduce the computational cost and enhance the classification performance. We experimentally show on two real-world datasets at the sample removal ratio of 0.2 that, with outlier removal via ORDI, we can simultaneously (1) improve the accuracy of the classifier by 1%, and (2) provide significant saving on the total running time by 1.5x and 2x on the two datasets. Hence, ORDI can provide a win-win situation in this performance-complexity tradeoff of the kernel machines for big data analysis.

Index Terms— Classification, big data, kernel machines, discriminant information, outlier removal

1. INTRODUCTION

Kernel methods [1, 2] have been successful techniques in pattern recognition for a variety of applications, e.g. speech [3], image [4], medical diagnosis [5], etc. The underlying mechanic of kernel methods is the kernel matrix. However, given N training samples, the kernel matrix scales to $\mathcal{O}(N^2)$ in complexity. This can be a limiting factor when dealing with a large dataset, which can consist of over a million training samples. Previous works have circumvented this challenge mostly via the kernel matrix approximation [6, 7, 8, 9, 10, 11]. On the other hand, an alternative approach that has not been thoroughly investigated is via *outlying data sample removal* or simply, *outlier removal*. As suggested by Vapnik [12], Burges and Scholkopf [13], and Blum and Langley [14], an ability to de-emphasize bad training samples can be crucial to the success of a learning machine, and increase the learning rate by confining the search space of hypotheses. From this perspective, the challenge of large training samples may be addressed effectively via the outlying data removal process [15, 16, 14].

This work focuses on the outlying data removal approach for large kernel machines. Though, it is important to point out that, these two lines of approaches are actually complementary since one works on the features, while the other works on the samples. The outlier removal approach to large-scaled kernel methods aims at trimming a subset of the training data samples *a priori*, and train the model only with the remaining subset of data [16]. This requires a metric to predict the quality of each data sample. Since this work focuses on the classification problem via kernel machines, we adopt Fisher's discriminant analysis [17, 18] as the basis of our quality measure. This scheme has been theoretically proven to be optimal for classification under the Gaussian assumption [19, 20, 21]; and experimentally shown to be effective for building a classifier [22, 17, 23], designing a compression algorithm [24], and selecting kernel functions [25].

Particularly, we develop and propose the *Outlier-Removal Discriminant Information (ORDI)* as the quantitative metric for measuring the contribution of each data sample toward the pattern recognition task. ORDI can be computed, and, more remarkably, can be *kernelized* very efficiently with $\mathcal{O}(N)$ complexity. Hence, the ORDI score of the kernel-embedded samples can be computed in linear time. Our proposed method then uses the ORDI score with a simple filter method to remove outlying data from the training set. Finally, we experimentally show on two real-world datasets that the ORDI filtering method can improve the classification accuracy by 1%, while effectively shortening the total running time by 1.5x and 2x on the two datasets. In addition, when compared to three related works, we show that the ORDI filtering method can significantly outperform all of them on the classification performance by as much as 15%. Hence, our ORDI filtering method has the potential to present a win-win situation for the performance-complexity tradeoff.

2. RELATED WORKS

There are two primary approaches for efficient large-scaled kernel machines. The prominent approach in the literature is via the kernel matrix approximation [6, 7, 8, 9, 10], such as the Nystrom approximation [11, 10] and the random kitchen sinks [6, 7]. The other approach is via the *data sample selection* or *outlying data sample removal* [15, 16, 14]. This approach uses only a subset of the training samples for the learning process.

These two methods are complimentary, i.e. both can be used jointly to provide maximum speed-up. Since our work follows the latter approach, we focus this discussion on the outlying data removal method. Blum and Langley [14] categorize works on outlying data removal for kernel machines into three types – the embedded method [14], the wrapper method [15, 16], and the filter method [26]. Our work falls into the *filter method*, and we experimentally compare our method to one previous filter method, as well as two previous wrapper methods. The embedded method, meanwhile, has not been shown in the literature to be effective in recent years.

The two wrapper methods iteratively train a classifier on the select samples, and use the distance between the decision hyperplane

and the remaining samples to select the next samples. The first uses the *active selection criterion* (ACT_WRAPPER) [15], which selects the samples that are closest to the current hyperplane. The other uses the *gradient selection criterion* (GRAD_WRAPPER) [15], which selects the most mis-classified samples.

The filter method is based on the *Johnson-Lindenstrauss (JL) transform* (JL_TRANSFORM) [26], which can preserve the separability of the training data after trimming. One simple way to implement this method is via a random selection of the data samples [26].

3. PRELIMINARIES

3.1. Discriminant Information

Derived from the bases of Fisher's discriminant analysis [17] and mutual information to the utility subspace [21, 20], the *discriminant information (DI),* ψ, is a metric for determining the separability of the data for classification. It has been shown – both theoretically [20, 21] and empirically [17, 25, 23, 27] – to be indicative of the classification ability of learning machines. Hence, we use the discriminant information as the basis of our analysis on the importance of each data sample toward the learning process.

More formally, given a supervised training dataset $\{\mathbf{X} \in \mathbb{R}^{M\times N}, \mathbf{y} \in \mathbb{R}^N\}$ with M features and N samples, define the *scatter matrix* and the *between-class scatter matrix*, respectively, as

$$\bar{\mathbf{S}} = \bar{\mathbf{X}}\bar{\mathbf{X}}^T;\ \mathbf{S}_B = \sum_{l=1}^{L} N_l \bar{\boldsymbol{\mu}}_l \bar{\boldsymbol{\mu}}_l{}^T,$$

where $\bar{\mathbf{X}}$ is the centered data, $\bar{\boldsymbol{\mu}}_l$ is the centered class-l mean, N_l is the number of samples in class l, and L is the number of classes. Then, the *discriminant information* (ψ) is defined as $\psi = \mathrm{tr}(\bar{\mathbf{S}}^{-1}\mathbf{S}_B)$ [1, 20, 21, 18].

Next, suppose a sample $\mathbf{x}$ is removed from the training data. Let us denote the new discriminant information as $\mathrm{tr}(\bar{\mathbf{S}}'^{-1}\mathbf{S}'_B)$, where $\bar{\mathbf{S}}'$ and $\mathbf{S}'_B$ are derived from the remaining training data. Then define the *Outlier-Removal Discriminant Information (ORDI)* of $\mathbf{x}$ as,

$$d\psi(\mathbf{x}) = \mathrm{tr}(\bar{\mathbf{S}}^{-1}\mathbf{S}_B) - \mathrm{tr}(\bar{\mathbf{S}}'^{-1}\mathbf{S}'_B).$$

Since $d\psi(\mathbf{x})$ indicates the reduction in the discriminant information caused by removing $\mathbf{x}$, it can be used as a metric for data sample removal. However, the exact computation of $d\psi(\mathbf{x})$ can be expensive, as it requires at least one matrix inversion. Therefore, in Section 4.2, we derive an approximation of $d\psi(\mathbf{x})$ that can be computed efficiently even in the kernel-embedded feature space.

3.2. Linear Algebra

Our analysis requires the following theorems in linear algebra.

Theorem 1 (Merikoski-Sarria-Tarazaga [28])**.** *The non-increasingly ordered singular values of a matrix* $\mathbf{A} \in \mathbb{R}^{M\times N}$ *have the values of* $0 \le \sigma_i \le \frac{\|\mathbf{A}\|_F}{\sqrt{i}}$*, where* $\|\cdot\|_F$ *is the Frobenius norm of a matrix.*

Theorem 2 (von Neumann [29])**.** *Let* $\sigma_i(\mathbf{A})$ *and* $\sigma_i(\mathbf{B})$ *be the non-increasingly ordered singular values of* $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{M\times N}$*. Then,* $\mathrm{tr}(\mathbf{A}\mathbf{B}^T) \le \Sigma_{i=1}^{R}\sigma_i(\mathbf{A})\sigma_i(\mathbf{B})$*, where* $R = \min\{M, N\}$.

4. METHOD

4.1. Overview

We propose the *ORDI data sample filtering method* as follows. Given a training dataset $\{\mathbf{x}_i, y(i)\}_{i=1}^N$, derive $d\psi(\mathbf{x}_i)$ for all samples, and then use $d\psi(\mathbf{x}_i)$ as the metric for the filter method in the outlier removal process. Since high $d\psi(\mathbf{x}_i)$ means that removing $\mathbf{x}_i$ significantly reduces the discriminant information of the training data, the samples are sorted in the decreasing order of their $d\psi(\mathbf{x}_i)$, and the filter stage simply removes the lowest P samples. Finally, we note that the $d\psi(\mathbf{x}_i)$ metric can be used with the wrapper method as well. However, in this work, we consider the simpler filter method, and $d\psi(\mathbf{x}_i)$ for each sample is only derived once.

4.2. Outlier-Removal Discriminant Information

The main challenge of the ORDI filtering method is to compute $d\psi(\mathbf{x}_i)$ *efficiently* and *effectively*. In addition, since the main motivation of this work is from the kernel methods, the computation of $d\psi(\mathbf{x}_i)$ should also be *kernelizable*. The following theorem is our main theoretical result, which possesses all of the desired properties.

Theorem 3. *Given a supervised training dataset* $\{\mathbf{X} \in \mathbb{R}^{M\times N}, \mathbf{y} \in \mathbb{R}^N\}$ *and a kernel function* $k(\mathbf{x}_i, \mathbf{x}_j)$*, the Outlier-Removal Discriminant Information (ORDI)* $d\psi(\mathbf{x})$ *of the sample* $\mathbf{x}$ *is bounded by*

$$d\psi(\mathbf{x}) \le \frac{\beta\kappa_{\mathbf{x}}}{\rho(\kappa_{\mathbf{x}}-\rho)} + \frac{H_{4,1/2}(\delta_{\mathbf{x}}+\kappa_{\mathbf{x}})}{\rho(N_{\mathbf{x}}-1)} + \frac{\kappa_{\mathbf{x}}(\delta_{\mathbf{x}}+\kappa_{\mathbf{x}})}{\rho(\kappa_{\mathbf{x}}-\rho)(N_{\mathbf{x}}-1)},$$

where the variables are defined as follows. $\beta = \sum_{l=1}^{L} N_l k(\boldsymbol{\mu}_l, \boldsymbol{\mu}_l)$*;* $\kappa_{\mathbf{x}} = k(\mathbf{x}, \mathbf{x})$*;* $N_{\mathbf{x}}$ *is the number of the training samples in the class as that* $\mathbf{x}$ *belongs to (including* $\mathbf{x}$*);* $H_{4,1/2}$ *is the generalized harmonic number;* $\rho > 0$ *is the ridge parameter;* $\boldsymbol{\mu}_l$ *is the class-mean of the* l^{th} *class, and* N_l *is the number of samples in the* l^{th} *class;* $\boldsymbol{\mu}_{\mathbf{x}}$ *is the class-mean of the class* $\mathbf{x}$ *belongs to; and*

$$\begin{aligned}\delta_{\mathbf{x}} =& N_{\mathbf{x}}[k(\boldsymbol{\mu}_{\mathbf{x}}, \boldsymbol{\mu}_{\mathbf{x}})^2 - 4k(\boldsymbol{\mu}_{\mathbf{x}}, \boldsymbol{\mu}_{\mathbf{x}}) \cdot k(\mathbf{x}, \boldsymbol{\mu}_{\mathbf{x}}) \\ &+ 2\kappa_{\mathbf{x}} k(\boldsymbol{\mu}_{\mathbf{x}}, \boldsymbol{\mu}_{\mathbf{x}}) + 2k(\mathbf{x}, \boldsymbol{\mu}_{\mathbf{x}})^2]^{1/2}.\end{aligned}$$

Proof. The proof works on the kernel-embedded feature space: $\mathbf{x} \in \mathbb{R}^M \mapsto \boldsymbol{\phi} \in \mathbb{R}^J$, where $k(\mathbf{x}_i, \mathbf{x}_j) = \boldsymbol{\phi}_i^T\boldsymbol{\phi}_j$. The discriminant information is defined as $\psi = \bar{\mathbf{S}}^{-1}\mathbf{S}_B$, where $\bar{\mathbf{S}} = \boldsymbol{\Phi}\boldsymbol{\Phi}^T$ and $\mathbf{S}_B = \sum_{l=1}^{L} n_l\boldsymbol{\nu}_l\boldsymbol{\nu}_l^T = \sum_{l=1}^{L}\mathbf{S}_l$, where $\boldsymbol{\nu}_l$ is the class-mean of the l^{th} class. For notational clarity and simplicity, we assume that all of the data are centered, and that the data mean does not change when one sample is removed. This is justifiable when N is very large, which is the scenario we consider here.

Removing a sample $\boldsymbol{\phi}$ causes the two scatter matrices to change: $\bar{\mathbf{S}}' = \bar{\mathbf{S}} - \boldsymbol{\phi}\boldsymbol{\phi}^T$ and $\mathbf{S}'_B = \mathbf{S}_B + \mathbf{E}_B$, where $\mathbf{E}_B$ is derived as follows. Suppose the removed sample $\boldsymbol{\phi}$ belongs to class $l = \ell$, then $\mathbf{S}'_B = \mathbf{S}_B - \mathbf{S}_\ell + \mathbf{S}'_\ell$, where $\mathbf{S}'_\ell = (N_\ell - 1)\boldsymbol{\nu}'_\ell\boldsymbol{\nu}'^T_\ell$, and $\boldsymbol{\nu}'_\ell = (\boldsymbol{\nu}_\ell N_\ell - \boldsymbol{\phi})/(N_\ell - 1)$. With algebraic modification, we have

$$\mathbf{E}_B = \frac{(N_\ell\boldsymbol{\nu}_\ell\boldsymbol{\nu}_\ell^T - N_\ell\boldsymbol{\nu}_\ell\boldsymbol{\phi}^T - N_\ell\boldsymbol{\phi}\boldsymbol{\nu}_\ell^T + \boldsymbol{\phi}\boldsymbol{\phi}^T)}{(N_\ell - 1)} \tag{1}$$

From this, we have that $\mathrm{rank}(\mathbf{E}_B) \le 4$ [30], and we can write

$$d\psi(\mathbf{x}) = \mathrm{tr}[\bar{\mathbf{S}}^{-1}\mathbf{S}_B - \bar{\mathbf{S}}'^{-1}\mathbf{S}'_B] = \mathrm{tr}[\bar{\mathbf{S}}^{-1}\mathbf{S}_B - (\bar{\mathbf{S}} - \boldsymbol{\phi}\boldsymbol{\phi}^T)^{-1}(\mathbf{S}_B + \mathbf{E}_B)] \tag{2}$$

The latter term can be expanded via the Woodbury identity [30]:

$$(\bar{\mathbf{S}} - \boldsymbol{\phi}\boldsymbol{\phi}^T)^{-1}(\mathbf{S}_B + \mathbf{E}_B) = (\bar{\mathbf{S}}^{-1} + \frac{\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}}{1 - \boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}})(\mathbf{S}_B + \mathbf{E}_B)$$

Substitute this term into Eq. (2),

$$d\psi(\mathbf{x}) = \mathrm{tr}[\frac{\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\mathbf{S}_B}{\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\boldsymbol{\phi} - 1} + \bar{\mathbf{S}}^{-1}\tilde{\mathbf{E}}_B + \frac{\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\mathbf{E}_B}{\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\boldsymbol{\phi} - 1}], \tag{3}$$

where $\tilde{\mathbf{E}}_B = -\mathbf{E}_B$. To find the upper-bound for $d\psi(\mathbf{x})$, we derive such bound for the three summands separately since trace is additive.

The first term: $\mathrm{tr}[\frac{\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\mathbf{S}_B}{\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}-1}]$. With Theorem 2, and the fact that $\boldsymbol{\phi}\boldsymbol{\phi}^T$ is a rank-1 matrix,

$$\mathrm{tr}[\frac{\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\mathbf{S}_B}{\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}-1}] \leq \frac{\sum_i \sigma_i(\boldsymbol{\phi}\boldsymbol{\phi}^T)\sigma_i(\bar{\mathbf{S}}^{-1}\mathbf{S}_B\bar{\mathbf{S}}^{-1})}{(\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}-1)} \leq \frac{(\boldsymbol{\phi}^T\boldsymbol{\phi})\sigma_1(\bar{\mathbf{S}}^{-1}\mathbf{S}_B\bar{\mathbf{S}}^{-1})}{(\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}-1)}, \quad (4)$$

where σ_1 is the highest singular value. $\sigma_1(\bar{\mathbf{S}}^{-1}\mathbf{S}_B\bar{\mathbf{S}}^{-1})$ is equivalent to the spectral norm, so we use the submultiplicative property,

$$\sigma_1(\bar{\mathbf{S}}^{-1}\mathbf{S}_B\bar{\mathbf{S}}^{-1}) = \left\|\bar{\mathbf{S}}^{-1}\mathbf{S}_B\bar{\mathbf{S}}^{-1}\right\|_2 \leq \left\|\bar{\mathbf{S}}^{-1}\right\|_2^2 \|\mathbf{S}_B\|_2 \quad (5)$$

We then consider these two terms individually. First, $\left\|\bar{\mathbf{S}}^{-1}\right\|_2^2 = \sigma_1(\bar{\mathbf{S}}^{-1}) = 1/\sigma_J(\bar{\mathbf{S}})$, where $\sigma_J(\bar{\mathbf{S}})$ is the smallest singular value of $\bar{\mathbf{S}}$. Since typically, $\bar{\mathbf{S}}$ is regularized by a ridge $\rho > 0$, i.e. $\bar{\mathbf{S}} + \rho\mathbf{I}$, it can be said that $\sigma_J(\bar{\mathbf{S}}) > \rho$, so we can bound $\left\|\bar{\mathbf{S}}^{-1}\right\|_2^2 < 1/\rho$. Next, consider the norm $\|\mathbf{S}_B\|_2 = \left\|\sum_{l=1}^{L} N_l\boldsymbol{\nu}_l\boldsymbol{\nu}_l^T\right\|_2$. With the triangular inequality, we can write,

$$\|\mathbf{S}_B\|_2 \leq \sum_{l=1}^{L} N_l \left\|\boldsymbol{\nu}_l\boldsymbol{\nu}_l^T\right\|_2 = \sum_{l=1}^{L} N_l\boldsymbol{\nu}_l^T\boldsymbol{\nu}_l = \beta. \quad (6)$$

Hence, we can bound Eq. (5) as $\sigma_1(\bar{\mathbf{S}}^{-1}\mathbf{S}_B\bar{\mathbf{S}}^{-1}) \leq \beta/\rho^2$. To conclude the upper-bound derivation of Eq. (4), we consider its denominator. Since we are interested in the upper-bound, we can assume that $\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\boldsymbol{\phi} > 1$. This assumption is conservative because, when the assumption does not hold, it would overestimate the importance of the sample. Hence, we would not discard the samples we should keep. Then, we use the property proved by von Neumann (cf. [31, 30]) to write $\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\boldsymbol{\phi} = \mathrm{tr}(\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}) = \mathrm{tr}(\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}\boldsymbol{\phi}^T) = c\sigma_1(\bar{\mathbf{S}}^{-1})\boldsymbol{\phi}^T\boldsymbol{\phi}$, where $c \in [-1, 1]$. Hence, for simplicity, we use the following approximation: $\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\boldsymbol{\phi} \approx \sigma_1(\bar{\mathbf{S}}^{-1})\boldsymbol{\phi}^T\boldsymbol{\phi} = (\boldsymbol{\phi}^T\boldsymbol{\phi})/\rho$. Then, we finally arrive at the upper-bound for Eq. (4):

$$\mathrm{tr}[\frac{\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\mathbf{S}_B}{\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}-1}] \leq \frac{\beta(\boldsymbol{\phi}^T\boldsymbol{\phi})}{\rho(\boldsymbol{\phi}^T\boldsymbol{\phi}-\rho)} \quad (7)$$

The second term: $\mathrm{tr}(\bar{\mathbf{S}}^{-1}\tilde{\mathbf{E}}_B)$. From Theorem 2,

$$\mathrm{tr}(\bar{\mathbf{S}}^{-1}\tilde{\mathbf{E}}_B) \leq \sum_{i=1}^{4} \sigma_i(\bar{\mathbf{S}}^{-1})\sigma_i(\tilde{\mathbf{E}}_B), \quad (8)$$

since $\mathrm{rank}(\tilde{\mathbf{E}}_B) \leq 4$. Then, using Theorem 1 and the scaling-by-scalar property of the norm, we get $\sigma_i(\tilde{\mathbf{E}}_B) \leq \frac{\left\|\tilde{\mathbf{E}}_B\right\|_F}{\sqrt{i}} = \frac{\|\mathbf{E}_B\|_F}{\sqrt{i}}$. Substitute in the definition of $\mathbf{E}_B$ and use the triangular inequality,

$$\sigma_i(\tilde{\mathbf{E}}_B) \leq \frac{\left\|N_\ell\boldsymbol{\nu}_\ell\boldsymbol{\nu}_\ell^T - N_\ell\boldsymbol{\nu}_\ell\boldsymbol{\phi}^T - N_\ell\boldsymbol{\phi}\boldsymbol{\nu}_\ell^T\right\|_F + \left\|\boldsymbol{\phi}\boldsymbol{\phi}^T\right\|_F}{(N_\ell-1)\sqrt{i}}$$

Then, using the property that $\|\mathbf{A}\|_F^2 = \mathrm{tr}(\mathbf{A}^T\mathbf{A})$,

$$\sigma_i(\tilde{\mathbf{E}}_B) \leq \frac{1}{(N_\ell-1)\sqrt{i}}\Big[N_\ell[(\boldsymbol{\nu}_\ell^T\boldsymbol{\nu}_\ell)^2 - 4(\boldsymbol{\nu}_\ell^T\boldsymbol{\nu}_\ell)(\boldsymbol{\nu}_\ell^T\boldsymbol{\phi}) + 2(\boldsymbol{\nu}_\ell^T\boldsymbol{\nu}_\ell)(\boldsymbol{\phi}^T\boldsymbol{\phi}) + 2\boldsymbol{\phi}^T\boldsymbol{\nu}_\ell]^{1/2} + \boldsymbol{\phi}^T\boldsymbol{\phi}\Big]$$

and with the definition of $\delta_{\mathbf{x}}$, we get $\sigma_i(\tilde{\mathbf{E}}_B) \leq \frac{(\delta_{\mathbf{x}}+\boldsymbol{\phi}^T\boldsymbol{\phi})}{(N_\ell-1)\sqrt{i}}$. Substitute into Eq. (8) and use the bound on $\sigma_1(\bar{\mathbf{S}}^{-1}) < 1/\rho$ from the derivation of the upper-bound of the first term, and we arrive at,

$$\mathrm{tr}(\bar{\mathbf{S}}^{-1}\tilde{\mathbf{E}}_B) \leq \sum_{i=1}^{4} \frac{(\delta_{\mathbf{x}}+\boldsymbol{\phi}^T\boldsymbol{\phi})}{\rho(N_\ell-1)\sqrt{i}} \leq \frac{(\delta_{\mathbf{x}}+\boldsymbol{\phi}^T\boldsymbol{\phi})}{\rho(N_\ell-1)} H_{4,1/2} \quad (9)$$

The third term: $\mathrm{tr}[\frac{\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\mathbf{E}_B}{\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}-1}]$. Using similar techniques as in the first term, we can write

$$\mathrm{tr}[\frac{\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\mathbf{E}_B}{\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}-1}] \leq \frac{(\boldsymbol{\phi}^T\boldsymbol{\phi})\sigma_1(\bar{\mathbf{S}}^{-1}\mathbf{E}_B\bar{\mathbf{S}}^{-1})}{(\boldsymbol{\phi}^T\boldsymbol{\phi}/\rho-1)} \quad (10)$$

Similar to the first term, we have $\sigma_1(\bar{\mathbf{S}}^{-1}\mathbf{E}_B\bar{\mathbf{S}}^{-1}) \leq \left\|\bar{\mathbf{S}}^{-1}\right\|_2^2 \|\mathbf{E}_B\|_2$. From the derivation in the second term and Theorem 1, we readily get $\|\mathbf{E}_B\|_2 = \sigma_1(\mathbf{E}_B) \leq \|\mathbf{E}_B\|_F \leq \frac{(\delta_{\mathbf{x}}+\boldsymbol{\phi}^T\boldsymbol{\phi})}{(N_\ell-1)}$. Using the upper-bound of $\left\|\bar{\mathbf{S}}^{-1}\right\|_2^2$ derived in the derivation in the first term, we can bound the singular value as $\sigma_1(\bar{\mathbf{S}}^{-1}\mathbf{E}_B\bar{\mathbf{S}}^{-1}) \leq \frac{(\delta_{\mathbf{x}}+\boldsymbol{\phi}^T\boldsymbol{\phi})}{\rho^2(N_\ell-1)}$. Substitute this into Eq. (10), and we get,

$$\mathrm{tr}[\frac{\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\mathbf{E}_B}{\boldsymbol{\phi}^T\bar{\mathbf{S}}^{-1}\boldsymbol{\phi}-1}] \leq \frac{(\boldsymbol{\phi}^T\boldsymbol{\phi})(\delta_{\mathbf{x}}+\boldsymbol{\phi}^T\boldsymbol{\phi})}{\rho(\boldsymbol{\phi}^T\boldsymbol{\phi}-\rho)(N_\ell-1)} \quad (11)$$

Finally, putting the upper-bounds for the three additive terms in Eq. (3) together; replacing N_ℓ with $N_{\mathbf{x}}$; and using the kernel trick $k(\mathbf{x}_i, \mathbf{x}_j) = \boldsymbol{\phi}_i^T\boldsymbol{\phi}_j$, we have completed the proof. □

Theorem 3 provides an approximate worst-case value of $d\psi(\mathbf{x})$. This upper-bound can be used as a metric to determine the importance of each sample toward the overall discriminant information. The rationale is that, the higher the value, the more important the data sample toward the classification task. Moreover, it incorporates the kernel trick, so the $d\psi(\mathbf{x})$ in the kernel-embedded feature space can readily be computed from the kernel function.

4.3. Complexity Analysis

We consider the complexity with respect to the number of kernel operations, i.e. $k(\cdot,\cdot)$. For example, given N samples, to compute the full kernel matrix, the complexity is $\mathcal{O}(N^2)$. The effect of outlying data removal on the computational complexity of the kernel matrix is the reduction from $\mathcal{O}(N^2)$ to $\mathcal{O}(N'^2)$, where N' is the number of remaining training samples. However, the outlier removal process itself has an overhead computational cost, and we consider that of the ORDI filtering method here.

For a given $\mathbf{x}$, $d\psi(\mathbf{x})$ only requires three kernel operations, viz. $k(\mathbf{x},\mathbf{x})$, $k(\mathbf{x},\boldsymbol{\mu}_{\mathbf{x}})$, and $k(\boldsymbol{\mu}_{\mathbf{x}},\boldsymbol{\mu}_{\mathbf{x}})$ (or only one for shift-invariant kernels). Since the filter method requires only one pass through the dataset, the overhead of the ORDI filtering method is $\mathcal{O}(N)$. With only linear additional complexity, the ORDI filtering method can have significant saving on the overall computational complexity.

5. EXPERIMENTS

5.1. Datasets

We use two datasets as follows. (a) *Human Activity Recognition Using Smartphones (HAR)* [32] has 561 features derived from mobile sensors. The classification task is to predict one of the 6 activities the subject is performing. There are 2,947 samples for testing, and 7,352 samples for training. (b) *Sensorless Drive Diagnosis (Drive)* [33] has 48 features derived from electric current drive signals, and the classification task is to predict one of the 11 car conditions. There are 11,000 samples for testing, and 47,509 samples for training.

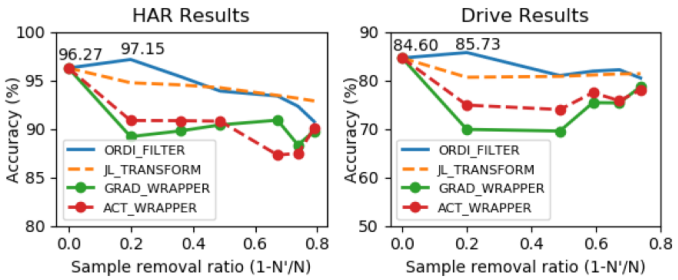

Fig. 1. Classification results on the HAR (Left) and Drive (Right) datasets. Given the initial N training samples, N' is the remaining samples after the outlier removal process. The x-axis is the sample removal ratio (1-N'/N). The two annotated numbers on both plots are the accuracy at ratio 0 and 0.2 using our ORDI filtering method.

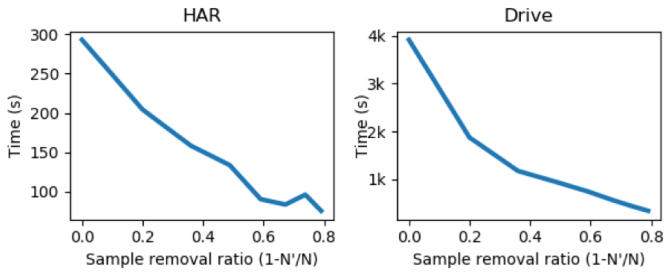

Fig. 2. The total running time (ORDI filtering + SVM training + testing) across various sample removal ratios for the the HAR (Left), and Drive (Right) datasets.

5.2. Procedure

For all experiments, we use support vector machine (SVM) [34] as the classifier with $C = 1$. Other parameters are selected via the 3-fold cross-validation including the kernel function, and the ridge parameter ρ in $d\psi(\mathbf{x})$. The set of possible kernels includes linear, poly2, poly3, and Gaussian with the $\gamma \in 10^{-\{1,2,3\}}$ (see [12] for the definitions); and the set of possible ρ is $10^{\{-2,-1,\ldots,2\}}$. For the proposed ORDI filtering method (ORDI_FILTER), each sample is first ranked in descending order by its $d\psi(\mathbf{x})$ derived via Theorem 3. Then, samples are removed such that there remains roughly equal number of samples in each class. This is to ensure that the supervised training process has samples in every class.

5.3. Results

Fig. 1 reports the classification results with outlier removal on the two datasets. Let N be the initial number of available samples, and N' be the number of samples used to train SVM. Then, the x-axis corresponds to the *sample removal ratio* (1-N'/N). The ratio of 0 is therefore the result when SVM is trained on the entire N samples.

5.3.1. Comparison to Other Methods

The results on both datasets show that our method (ORDI_FILTER) has the highest classification accuracy among other three methods across almost all sample removal ratios. First, comparing to the two wrapper methods (GRAD_WRAPPER and ACT_WRAPPER), our ORDI filtering method yields significantly higher accuracy across all sample removal ratios, and is better by as much as 8% and 15% at 0.2 ratio on the HAR and Drive datasets, respectively.

Compared to the JL transform, ORDI_FILTER performs better at lower removal ratios, and by as much as 2.5% and 5% at the ratio of 0.2 on the HAR and Drive datasets, respectively. Although the JL transform performs slightly better at very high removal ratio ($>$ 0.7), such high ratios are not favorable in practice since it might intolerably hurt the generalization ability of the classifier [12][1].

5.3.2. Comparison to No Outlier Removal

The left-most point on the two plots in Fig. 1 corresponds to the baseline performance when all samples are used for training. The accuracy is 96.27% for HAR and 84.60% for Drive, as annotated. The result from the ORDI filtering method shows that, by removing $0.2N$ training samples, there is a gain in the classification accuracy by $\sim$ 1% on both datasets. This maybe partially explained by the observation in [19, 14] that restricting the search space can facilitate the learning process. Determining how much data to be removed should be data-dependent and is a topic for future research.

[1]For example, consider SVM, whose bound on test error is approximately $\propto 1/N'$ (cf. [12], pg 143). Assume that the test error bound using all samples is 5%; then the sample removal ratio of 0.2 would yield the error bound of 6.25%, whereas the ratio of 0.7 would yield the error bound of 16.67%.

5.4. Complexity Consideration

As discussed in Section 4.3, outlier removal via ORDI can reduce the computational complexity from $\mathcal{O}(N^2)$ to $\mathcal{O}(N'^2)$ with a small additional cost of $\mathcal{O}(N)$. We experimentally test this analysis on the two datasets. The results of the total running time (ORDI filtering + SVM training + testing), as reported in Fig. 2, confirm the computational saving provided by the ORDI filtering method. Moreover, combining this time saving performance with the classification performance in Fig. 1 at the sample removal ratio of 0.2, the ORDI filtering method provides the computational savings of 1.5x and 2x on HAR and Drive, respectively, while also providing the accuracy performance gain of 1% on both datasets. These results show that the ORDI filtering outlier removal method presents a win-win situation in terms of classification performance and computational efficiency.

6. CONCLUSION

In this work, we present an outlying data sample removal method to be used with kernel machines for big data analysis. We propose the Outlier-Removal Discriminant Information (ORDI) filtering method to remove non-discriminative training samples. ORDI ensures that the remaining subset of training data preserves most of the separability and, since ORDI can be kernelized, it can be computed efficiently in the kernel-embedded space. Via experiments on two datasets, we show that the ORDI outlier removal pre-processing step can both improve the classification performance and reduce the total running time of the kernel machine. Hence, it presents a promising prospect for kernel machines on big data analysis.

Acknowledgement

This material is based on work supported in part by the Brandeis Program of Defense Advanced Research Project Agency (DARPA) and Space and Naval Warfare System Center Pacific (SSC Pacific) under Contract No. 66001-15-C-4068. We thank Prof. Pei-Yuan Wu and Prof. Morris Chang for the valuable insight and support.

7. REFERENCES

[1] S. Y. Kung, *Kernel Methods and Machine Learning*, Cambridge University Press, Cambridge, UK, 2014.

[2] Bernhard Scholkopf and Alexander J. Smola, *Learning with kernels: support vector machines, regularization, optimization, and beyond*, MIT press, 2002.

[3] Bjorn Schuller, Gerhard Rigoll, and Manfred Lang, "Speech emotion recognition combining acoustic features and linguistic information in a hybrid support vector machine-belief network architecture," in *Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04). IEEE International Conference on*. 2004, vol. 1, pp. I–577, IEEE.

[4] Edgar Osuna, Robert Freund, and Federico Girosit, "Training support vector machines: an application to face detection," in *Computer vision and pattern recognition, Proceedings., IEEE computer society conference on*. 1997, pp. 130–136, IEEE.

[5] Isabelle Guyon, Jason Weston, Stephen Barnhill, and Vladimir Vapnik, "Gene selection for cancer classification using support vector machines," *Machine Learning*, vol. 46, no. 1, pp. 389–422, 2002.

[6] Ali Rahimi and Benjamin Recht, "Random features for large-scale kernel machines," in *Advances in neural information processing systems*, 2007, pp. 1177–1184.

[7] Ali Rahimi and Benjamin Recht, "Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning," in *Advances in neural information processing systems*, 2009, pp. 1313–1320.

[8] Quoc Le, Tamas Sarlos, and Alex Smola, "Fastfood-approximating kernel expansions in loglinear time," in *Proceedings of the international conference on machine learning*, 2013, vol. 85.

[9] Po-Sen Huang, Li Deng, Mark Hasegawa-Johnson, and Xiaodong He, "Random features for kernel deep convex network," in *Acoustics, Speech and Signal Processing (ICASSP), International Conference on*. 2013, pp. 3143–3147, IEEE.

[10] Petros Drineas and Michael W. Mahoney, "On the nystroem method for approximating a gram matrix for improved kernel-based learning," *journal of machine learning research*, vol. 6, no. Dec, pp. 2153–2175, 2005.

[11] Christopher Williams and Matthias Seeger, "Using the nystroem method to speed up kernel machines," in *Proceedings of the 14th Annual Conference on Neural Information Processing Systems*, 2001, pp. 682–688.

[12] Vladimir Vapnik, *The nature of statistical learning theory*, Springer science & business media, 2013.

[13] Christopher JC Burges and Bernhard Scholkopf, "Improving the accuracy and speed of support vector machines," in *Advances in neural information processing systems*, 1997, pp. 375–381.

[14] Avrim L. Blum and Pat Langley, "Selection of relevant features and examples in machine learning," *Artificial Intelligence*, vol. 97, no. 1, pp. 245–271, 1997.

[15] Antoine Bordes, Seyda Ertekin, Jason Weston, and Leon Bottou, "Fast kernel classifiers with online and active learning," *Journal of Machine Learning Research*, vol. 6, no. Sep, pp. 1579–1619, 2005.

[16] Gaston Baudat and Fatiha Anouar, "Feature vector selection and projection using kernels," *Neurocomputing*, vol. 55, no. 1, pp. 21–38, 2003.

[17] RA Fisher, "The use of multiple measurements in taxonomic problems," *Annals of eugenics*, vol. 7, no. 2, pp. 179–188, 1936.

[18] RA Fisher, "Statistical methods for research workers (ed. 10) oliver and boyd," *Ltd, Edinburgh*, 1946.

[19] Jerome Friedman, Trevor Hastie, and Robert Tibshirani, *The elements of statistical learning*, vol. 1, Springer series in statistics Springer, Berlin, 2001.

[20] S. Y. Kung, "Compressive privacy: From information /estimation theory to machine learning [lecture notes]," *IEEE Signal Processing Magazine*, vol. 34, no. 1, pp. 94–112, 2017.

[21] S Y Kung, "A compressive privacy approach to generalized information bottleneck and privacy funnel problems," *Journal of the Franklin Institute*, 2017.

[22] Ross D. King, Cao Feng, and Alistair Sutherland, "Statlog: comparison of classification algorithms on large real-world problems," *Applied Artificial Intelligence an International Journal*, vol. 9, no. 3, pp. 289–333, 1995.

[23] Thee Chanyaswad, J. Morris Chang, and Sun-Yuan Kung, "A compressive multi-kernel method for privacy-preserving machine learning," in *Neural Networks (IJCNN), 2017 International Joint Conference on*. 2017, pp. 4079–4086, IEEE.

[24] S. Y. Kung, "Discriminant component analysis for privacy protection and visualization of big data," *Multimedia Tools and Applications*, pp. 1–36, 2015.

[25] Thee Chanyaswad, Mert Al, J. Morris Chang, and S. Y. Kung, "Differential mutual information forward search for multi-kernel discriminant-component selection with an application to privacy-preserving classification," in *Machine Learning for Signal Processing (MLSP), 2017 IEEE 27th International Workshop on*. 2017, IEEE.

[26] Maria-Florina Balcan, Avrim Blum, and Santosh Vempala, "Kernels as features: On kernels, margins, and low-dimensional mappings," *Machine Learning*, vol. 65, no. 1, pp. 79–94, 2006.

[27] Thee Chanyaswad, J. Morris Chang, Prateek Mittal, and SY Kung, "Discriminant-component eigenfaces for privacy-preserving face recognition," in *Machine Learning for Signal Processing (MLSP), International Workshop on*. 2016, IEEE.

[28] Jorma Kaarlo Merikoski, Humberto Sarria, and Pablo Tarazaga, "Bounds for singular values using traces," *Linear Algebra and its Applications*, vol. 210, pp. 227–254, 1994.

[29] J. von Neumann, "Some matrix inequalities and metrization of metric space," *Tomsk Univ.Rev*, vol. 1, pp. 286–296, 1937.

[30] Roger A. Horn and Charles R. Johnson, *Matrix analysis*, Cambridge university press, 2012.

[31] Leon Mirsky, "A trace inequality of john von neumann," *Monatshefte fr mathematik*, vol. 79, no. 4, pp. 303–306, 1975.

[32] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, and Jorge Luis Reyes-Ortiz, "A public domain dataset for human activity recognition using smartphones.," in *ESANN*, 2013.

[33] M. Lichman, "Uci machine learning repository," 2013.

[34] Corinna Cortes and Vladimir Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.