Applied Mathematics and Nonlinear Sciences

<https://www.sciendo.com>

AtanK-A New SVM Kernel for Classification

Meng Tian, Hongkui Li[†]

School of Mathematics and Statistics, Shandong University of Technology,
Zibo 255049, PR China

Submission Info

Communicated by Juan Luis García Guirao
Received September 4th 2020
Accepted September 27th 2021
Available online December 31st 2021

Abstract

The efficiency of support vector machine in practice is closely related to the optimal selection of kernel functions and their hyper-parameters. A novel kernel, namely the arctangent kernel, is proposed in this paper. Compared with the Gaussian kernel, the new proposed kernel has a quick similarity descent in the neighborhood of the inspection sample and a moderate similarity descent toward the infinity of the inspection sample. The experimental results on two simulated data sets and some UCI data sets show that the new proposed kernel function has better effectiveness and robustness compared with the polynomial kernel, the Gaussian kernel, the exponential radial basis function, and the former proposed kernel with moderate decreasing.

Keywords: Arctangent kernel, Classification, Generalization, Kernel function, Support vector machine

1 Introduction

Support vector machine (SVM) is a classical kernel method. It has grown substantially due to its good generalization capability even in cases of low quantity training setting over the last decades. It comes from Vapnik's theory, and develops on the structural risk minimization principle [1]. Illuminated by SVM, many kernel based methods, such as kernel principle component analysis (KPCA) [2], kernel clustering (KC) [3] kernel Fisher discriminant analysis (KFDA) [4], kernel canonical correlation analysis (KCCA) [5], and kernel linear discriminant analysis (GDA) [6] are proposed to solve various different problems in machine learning community.

In most practical problems the data samples are linear non-separable, kernel methods map these points into a high dimensional space and even infinite-dimensional space (named feature space) to gain a better linear separability. The mapping is defined implicitly by the chosen kernel function which represents the inner product of each pair of data points in the feature space without computing their mapping functions directly. Thus the

[†]Corresponding author.

Email address: lhk8068@163.com

linear algorithm which has a poor performance in the default space produces a very impressive performance in a suitable feature space. This is the well-known “kernel trick”, and it is the key to the success of kernel methods. The importation of kernel functions makes the SVM more robust against the curse of dimensionality. Based on this technique of “kernel trick”, kernel methods have been numerous algorithms developed due to their attractive advantages in terms of theoretical guarantees, efficient training time, and delicate high-dimensional data handle ability.

Regardless of the prosper of kernel methods, a poorly selected kernel can lead to a worse prediction performance than that of their linear counterparts in the original input space. The reason lies in the nature of data is hard to catch. So the estimation of the underlying relationship of data is now converted into the choice of kernel function and its parameters. As we all know, selecting an appropriate kernel or an optional kernel parameter, namely the appropriate feature space is a crucial target. [1, 7, 8]

There are several commonly used kernel functions, such as the linear kernel ($K(\mathbf{x}, \mathbf{z}) = \langle \mathbf{x}, \mathbf{z} \rangle$), the polynomial kernel ($K(\mathbf{x}, \mathbf{z}) = (\langle \mathbf{x}, \mathbf{z} \rangle + 1)^n, n \in \mathbb{N}$), the Gaussian kernel ($K(\mathbf{x}, \mathbf{z}) = \exp(-\|\mathbf{x} - \mathbf{z}\|^2 / 2\sigma^2), \sigma > 0$) and the exponential radial basis kernel ($K(\mathbf{x}, \mathbf{z}) = \exp(-\|\mathbf{x} - \mathbf{z}\| / 2\sigma^2), \sigma > 0$) [7].

In the literature, there are three approaches to choose kernel function or kernel parameters. First, the kernel selection can be reduced to real-valued parameter optimization. Especially, the Gaussian kernel has shown to be an optimal choice of kernel function. Many operations are taken to select its hyperparameters for different tasks. The most generally used method is grid search technique and cross-validation [1, 7, 8]. This approach works well in practice, however it is quite time-consuming because it need train SVMs with all desired combinations of parameters and screens them according to the training accuracy. Kernel alignment, Fisher discriminant criterion, Bayes predictive method and independence criterion are proposed subsequently [9–12]. And these operations are all based on kernel matrix and do not require the information of the trained SVMs, so they can save running time compared with cross-validation strategy. However, these methods based on target function optimization have their own disadvantages such as the minimum is not global, and their preferred accuracy depending on the type of data to be processed. Some dynamic evolutionary algorithms have been adapted to optimize the SVM parameters [13] [14]. But they are still time-consuming because of the training of SVMs from the population evolved generation by generation to converge. So far, there is no uniform standard or paradigm to solve the kernel parameter optimization problem of the Gaussian kernel function.

The second approach is combing several base kernels instead of using only one specific kernel to achieve better results. In the recent years, multiple kernel learning method (MKL) are proposed to overcome the difficulty of the Gaussian kernel optimization [15–17]. MKL adopts a set of predefined based kernels instead of selecting a single kernel function. The combination function of kernel functions has several forms, such as linear function, nonlinear function, or a data-dependent combination. Sincerely, choosing a suitable combination form, and further constructing the specific combination function is a challenging work. Many approaches are proposed to pursuit the combination coefficients by a joint optimization algorithm, or by using a two-stage alternate procedure. Testing results show that using multiple kernels can improve classification accuracy in practice compared with the single kernel approach, but the computational time of MKL is higher for determining the kernel function combination form.

The third way is creating new kernel functions to evade the kernel selection in the practice setting. A. Jacot et al. prove that the evolution of an ANN during training can also be described by a kernel, and proposed a new kernel-the Neural Tangent Kernel [18]. Built on the Cauchy distribution, Basak proposed the Cauchy kernel and observed the new proposed kernel has a long-tailed property which triggers its special performance sometime [19]. N. E. Ayat et al. propose a new kernel with moderate decreasing for pattern recognition [20]. Subsequently, Zhang and Wang presented a kernel function modifying the kernel proposed above, and the new proposed kernel can obtain better generalization performance compared with the old one [21]. Recently, some sets of generalized kernels based on orthogonal functions are proposed to simplify the kernel parameter optimization [22, 23]. M. Tian et al. analyzed the construction modes and the properties of several kinds of kernels based on orthogonal polynomials, and highlighted the similarities and differences among them [24]. But the specific construction of

these kernels makes the calculation of kernel function time-consuming.

A new kernel function based on the arctangent function is presented in this paper. The obtained kernel function has concentrated on a high similarity in the neighborhood of the inspection sample and a low similarity toward infinity. Meanwhile, this kernel has a quick similarity descent rate near to the inspection sample and a slow similarity descent rate far away from the inspection sample. We provide experimental results showing their good performances on synthetic and UCI benchmark data. The paper focuses on the classification tasks and considers the SVM algorithm is popular in various applications. The applicability and robustness of the proposed kernel will be investigated here.

The remaining part of this paper is organized as follows. Section 2 shows a clear description of the SVM for classification. Section 3 provides a detailed introduction of the new presented kernel. We give experimental results and discussions on performance in section 4. Section 5 concludes this article.

2 Review of Support Vector Classification

The support vector machine is a classical machine learning approach. In recent years, SVM has become a hot topic in pattern recognition, and SVM has been successfully applied in many practice fields. In particular, a classical SVM classifier is capable of producing a linear decision boundary between two different classes, and maximizing the margin in the feature space between them at the same time. Introductions to SVM for binary classification task are summarized in the following.

Consider a set of training labeled sample represented by $\{(\mathbf{x}_i, y_i)\}_{i=1}^l$, where $\mathbf{x}_i \in \mathbf{X} \subset \mathbb{R}^n$, $\mathbf{X}$ denotes the input space, and $\mathbb{R}^n$ denotes n -dimensional vector space. Here $y_i \in \{-1, 1\}$ is the label associated with $\mathbf{x}_i$. The SVM classifier training process which builds the separating hyperplane between these two classes can be formulated by that:

$$\begin{aligned} \min \quad & \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^l \xi_i \\ \text{s.t.} \quad & y_i(\omega^T \mathbf{x}_i + b) \geq 1 - \xi_i, \xi_i \geq 0, i = 1, \dots, l \end{aligned}$$

where ω and b define a linear classifier in the feature space. Here $\xi = (\xi_1, \dots, \xi_l)^T$ is the vector of slack variables and C is the penalty parameter of the error term defined by the users.

The Lagrange dual operation converts the optimization problem above into the dual formulation, which can be rewritten as:

$$\begin{aligned} \max \quad & \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle \\ \text{s.t.} \quad & 0 \leq \alpha_i \leq C, \sum_{i=1}^l \alpha_i y_i = 0. \end{aligned}$$

where $\{\alpha_i\}_{i \in \{1, \dots, l\}}$ are Lagrangian multipliers obtained by computing the optimization problems corresponding to each separation constraint. The samples with $\alpha_i > 0$ are named as support vectors and they are integral to define the separating boundary.

To solve real-world classification problems, the dot product $\langle \mathbf{x}_i, \mathbf{x}_j \rangle$ in the former algorithm can be replaced with a kernel function $K(\mathbf{x}_i, \mathbf{x}_j)$. The kinds of kernel K satisfying some defined characterization of dot product are all samples $\mathbf{x}_i$ and $\mathbf{x}_j$, which can be formulated as:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle, K : \mathbf{X} \times \mathbf{X} \rightarrow \mathbb{R},$$

where Φ denotes the feature map that their pairwise inner product of the mapped samples of all $\mathbf{x}_i$ and $\mathbf{x}_j$ in the augmented space $\mathbb{F}$, i.e. $\Phi : \mathbf{X} \rightarrow \mathbb{F}$. So the kernel function defines implicitly the nonlinear mapping.

In practice, some theorems have been proved and provided to verify whether a function can be a kernel function. In these theorems, the original and most common condition for a symmetric function $K(\mathbf{x}_i, \mathbf{x}_j)$ to be a kernel was raised by Mercer in 1909, namely a kernel is the function must be symmetric, continue and positive definite.

Fig. 1 Trends near the zero for AtanK kernel with different σ values.

3 The New Proposed Kernel Function

Based on unimodality and asymptotic of kernel functions, we proposed a new kind of kernel function. This kernel function is formulated as:

$$K(\mathbf{x}, \mathbf{z}) = \frac{2}{\pi} \arctan \frac{\sigma^2}{\|\mathbf{x} - \mathbf{z}\|^2}, (\sigma \neq 0) \quad (1)$$

where σ is a normalization constant. For convenience, the kernel function is denoted as AtanK in this paper.

To prove the AtanK kernel is a Mercer kernel. We give the following theorem which is helpful to prove positive definite of it.

Theorem 1. [1] Let $X \in \mathbb{R}^n, f : (0, \infty) \rightarrow \mathbb{R}$, and $K : \mathbb{X} \times \mathbb{X} \rightarrow \mathbb{R}$ be defined by $K(\mathbf{x}, \mathbf{z}) = f(\|\mathbf{x} - \mathbf{z}\|^2)$. If f is completely monotonic, then $K(\mathbf{x}, \mathbf{z})$ is a Mercer kernel.

Corollary 2. The AtanK kernel function in Eq. (1), $X \in \mathbb{R}^n$, defined on a compact domain and $\sigma \neq 0$, is a Mercer kernel.

Proof. Obviously, the AtanK function is continuous, symmetric and $K \in C^\infty$. Let $r = \|\mathbf{x} - \mathbf{z}\|^2$, and Eq. (1) can be simplified as in the following form:

$$K(r) = \frac{2}{\pi} \arctan \frac{\sigma^2}{r}$$

If $0 < r_1 < r_2$, we have

$$\begin{aligned} K(r_1) - K(r_2) &= \frac{2}{\pi} \arctan \frac{\sigma^2}{r_1} - \frac{2}{\pi} \arctan \frac{\sigma^2}{r_2} \\ &= \frac{2}{\pi} \left(-\frac{\sigma^2}{r_1^2 + \sigma^4} \right) (r_1 - r_2) > 0, \end{aligned}$$

Where $0 < r_1 < r_2$. So $K(r)$ is completely monotonic. Taking Theorem 1 in consideration, the corollary is proved.

Fig.1 shows the output of AtanK kernel function for different σ s. For clarity, the scanning range of x is $[-2, 2]$, and z is fixed at zero. One can observe that the AtanK kernel is unimodality and symmetric around the z .

N. E. Ayat et al. pointed that a good kernel function should have different decrease speed in the neighborhood and the infinity of the inspection sample. That is to say, a quick and a moderate decrease speed should behold in the neighborhood and the infinity of the inspection sample, respectively [20]. And the authors pointed that the Gaussian kernel satisfies the requirement in the neighborhood of the inspection sample but not the requirement in the infinity of the inspection sample. Based on this, the authors proposed a kernel with moderate decreasing (denoted as KMOD), whose analytic expression is as:

$$KMOD(\mathbf{x}, \mathbf{z}) = L \left[\exp \left(\frac{\gamma}{\|\mathbf{x} - \mathbf{z}\|^2 + \sigma^2} \right) - 1 \right] \quad (2)$$

Fig. 2 The trends near and far from the zero for Gauss, ERBF, UKF and AtanK kernels with different σ values.

Where L , γ and σ are three controlling parameters. By using mathematical formula $e^x \approx 1 + x$, Ref. [21] simplifies the formulation of the KMOD kernel, and proposed a new Mercer kernel, namely the UKF kernel, i.e.

$$UKF(\mathbf{x}, \mathbf{z}) = L(|\mathbf{x} - \mathbf{z}|^2 + \sigma^2)^{-\alpha} (\sigma \neq 0)$$

Note that these two kinds of kernel functions both need a pair of parameters. Obviously, optimizing these two parameters is a very time-consuming target. For simplicity, Ref. [20] set the parameters chosen from some predetermined sets and Ref. [21] set $L = \sigma^2$ and $\alpha = 1$. This procedures in Ref. [21] make certain the $UKF(\mathbf{x}, \mathbf{z})$ within the interval of $(0, 1]$. And it makes the $UKF(\mathbf{x}, \mathbf{z})$ do the comparison between the Gaussian kernel and the ERBF kernel easily.

Fig. 2 shows the kernel outputs of the Gaussian kernel, the ERBF kernel, the UKF kernel, and the AtanK kernel for the parameters 0.2, 1, and 2, respectively. In order to give a detail comparison, the curves of the Gaussian kernel, the ERBF kernel, the UKF kernel, and the AtanK kernel are all given here. We discarded the comparison between the UKF kernel and the KMOD kernel due to the reason that Ref. [21] have shown the UKF can learn more separation information compared with the KMOD kernel. For each parameter, the kernel

curves near to and far away from the neighborhood of the inspection sample zero in two-dimension plane are both shown.

From Fig.2, we can see that the AtanK kernel satisfies the quick decrease in the neighborhood of the inspection sample and the moderate decrease far away from the inspection sample. When $\mathbf{x}$ is near to 0, the AtanK kernel has a quicker decrease compared with the Gauss kernel, and when $\mathbf{x}$ is far away from 0, the AtanK kernel has a slower decrease compared with the ERBF kernel. The AtanK kernel is not too different from the UKF kernel, while the UKF kernel has two sets of parameters to be chosen.

Seen from the kernel function discussed above, we find that when describing the change of the kernel function value with respect to the inspection sample in two-dimensional space, the curve of the kernel function presents a unimodal function curve whose maximum point is exactly the inspection sample, whether the kernel function constructed based on the inner product form of $\langle \mathbf{x}, \mathbf{z} \rangle$ or the norm form of $\|\mathbf{x} - \mathbf{z}\|$. It is shown that the kernel function is a kind of expression of similarity. Generally speaking, the kernel functions constructed based on the norm form $\|\mathbf{x} - \mathbf{z}\|$ are unimodal and symmetrical about the inspection sample, such as the Gaussian kernel, the exponential radial basis kernel, the arctangent kernel. While the kernel function constructed based on the inner product form $\langle \mathbf{x}, \mathbf{z} \rangle$ does not satisfies symmetry, such as the polynomial kernel, the spline kernel, and some kinds of orthogonal polynomial kernels.

This paper discusses the different trends of similarity change near to and far away from the inspection sample, which in fact provides intuitive guidance for the similarity measurement requirements around the inspection sample. Various similarity measurement methods provide different inner product, or similarity, between mapping vectors in different feature spaces, which makes it possible to find the optimal feature spaces with better classification performance. However, because of the diversity and particularity of the real problems, we cannot determine which kernel function is the optimal kernel function for the specific problem. Therefore, we all choose the Gaussian kernel function as the universal kernel. As we all know, in the field of mathematical statistics, the normal distribution is a key type of data distribution. The existence of the maximum likelihood probability theorem makes the normal distribution be the highest peak in the classical statistical field. However, with the development of statistical technology, people have found that some specific data distribution cannot be characterized by the normal distribution in real life. Gradually, some new distributions such as the Chi-square distribution and the student distribution, are springing up. It can be imagined that the construction of kernel function is also faced with the same problem. The universal kernel function represented by the Gaussian kernel cannot describe all the similarity measurement. It is necessary to construct new kernel functions constantly to provide more possibilities for different practical problems and diverse similarity measurements.

4 Experimental results

The effectiveness of the proposed kernel function is assessed on 2 artificial datasets and 11 UCI benchmark datasets compared with the polynomial kernel, the Gaussian kernel, the UKF kernel, and the exponential radial basis kernel [26]. For simplicity, the polynomial kernel, the Gaussian kernel and the exponential radial basis kernel, are denoted as Poly, Gauss and ERBF, respectively.

4.1 Checkerboard Dataset Classification

To evaluate the performance and robustness of the AtanK kernel, we have performed a simple classification task in two dimensions. The results of the AtanK kernel in relation to the four kernel functions mentioned above (Poly, Gauss, UKF and ERBF) are performed on the checkerboard dataset.

The original training population is a set of 592 points, uniformly distributed in the unit square. We divide the square into 9 grids, and set these points of adjacent grids into two classes alternately. In the experiment, we select randomly 496 training samples and the last 96 samples as testing samples.

For simplicity, we set the coefficient n of the Poly kernel in the set of $\{1,2,3,4,5\}$, and the values of σ of

Table 1 Results on the Checkerboard dataset.

Kernel	Poly	Gauss	ERBF	UKF	AtanK
Parameter	$n=5$	$\sigma=0.5$	$\sigma=0.7$	$\sigma=1.1$	$\sigma=0.7$
Training accuracy (%)	67.14	98.18	100	97.18	94.15
Testing accuracy (%)	71.87	92.71	89.58	92.71	94.79
Number of SV	381	129	199	183	128

Table 2 Results on the Cosexp dataset.

Kernel	Poly	Gauss	ERBF	UKF	AtanK
Parameter	$n=2$	$\sigma=1.1$	$\sigma=0.7$	$\sigma=0.5$	$\sigma=1.3$
Training accuracy (%)	75.20	88.20	98.00	98.40	78.20
Testing accuracy (%)	75.00	71.00	71.00	70.00	72.00
Number of SV	281	193	134	103	128

the Gauss kernel and the ERBF kernel are chosen from $\{0.5, 0.7, 0.9, 1.1, 1.3\}$ in the artificial datasets. And we choose the same parameter $C = 10$ for 2 artificial datasets.

Table 1 lists the chosen parameter, the training accuracy and the minimal number of support vectors corresponding to several different kernels when they obtain their highest testing accuracy. Fig.3 shows the default checkerboard data set and the best classification resulting boundaries with different kernels.

From the results of Table 1, it can be seen that the AtanK kernel have both the highest testing accuracy and the lowest support vector numbers in spite of its ordinary training accuracy in these 5 kernel functions. As we all know the support vector ratio can be seen as an index reflecting the generalization error rate, so we take the number of SV into consideration here. On this index, the AtanK kernel is pretty much the same as the Gaussian kernel. In all, the performance of Poly is the worst. The Gaussian kernel is the best in the training accuracy, while its generalization is not so amazing. Seen from the testing accuracy performance, the Gaussian kernel and the UKF kernel have the same results, and both fall behind the AtanK kernel. And the AtanK kernel has the best classification performance in these kernels discussed above.

Seen from Fig. 3, the polynomial kernel has the worst performance that it does not find the adjacent grids structure at all. Though the ERBF kernel has the highest training accuracy, but its classification boundary is not smooth enough. Part of the reason lies in the quick decrease in the neighborhood of $\mathbf{x}$, and the local structure can be over-preserved. Fig. 3 shows that the Gaussian kernel, the UKF kernel and the AtanK kernel give the similar classification boundaries, and they all find out the adjacent grids structure of the checkerboard dataset.

4.2 Cosexp Dataset Classification

We give another artificial data set-the Cosexp dataset to evaluate the performance of the new proposed kernel [25]. Let the function $f(x) = 0.95\cos(\frac{e^x-1}{2})$ slice the xoy -plane. We uniformly and randomly take 600 sample points within the interval $[0, 5]$. The sample points above the curve and below the curve are labeled as the positive class and the negative class, respectively. In the subsection, we select randomly 500 training samples and the last 100 samples as testing samples. The parameter setting is as same as that in the former section.

The results of the AtanK kernel compared with the four kernel functions are investigated on the Cosexp dataset in Table 2. Table 2 shows the chosen parameter, the training accuracy and the minimal number of support vectors obtained with these kernels when they each obtain their highest testing accuracy. From the results of Table 2, it can be seen that the AtanK kernel and the polynomial kernel receive the better test accuracy, in spite of their naive training accuracy. And the UKF, ERBF and AtanK all have the good performance in terms of support vector numbers.

Fig. 4 shows the best classification resulting boundaries with different kernels on the Cosexp dataset. Seen from Fig. 4, we can see that the classification boundaries obtained by the ERBF kernel and the UKF kernel are more identical to the default boundary. They both have good local structure preserving ability, and it may be responsible for the result. At the same time, the difference of boundries obtained by the Gaussian kernel and the

Fig. 3 Two-dimensional Checkerboard dataset and classification boundaries of 5 sets of kernels on Checkerboard dataset.

AtanK kernel shows that these two kernel functions have an obvious difference in classification, although they both have symmetry and unimodality.

Combined the results of Table 2 and Fig. 4, we found the Poly kernel has the best classification accuracy despite it fails to find the structure of default separating boundary, while the UKF kernel has the worst classification accuracy despite it catch the more vivid structure of default separating boundary. It can be seen as a reflection of the difficulty of realizing the structural risk minimization in kernel selection. From this point of view, the AtanK kernel has a naive training accuracy performance, a better testing classification result, and a rougher separating boundary structure, so it is a well-balanced choice.

4.3 UCI Benchmark

We evaluate the classification performance improvement of our proposed kernel over the four commonly used kernels mentioned above on 11 data sets from the UCI benchmark repository [26]. The specifications of these data sets are listed in Table 3. The regularization parameter C in SVM is chosen from the set of $\{2^{-4}, 2^{-3}, 2^{-2}, 2^{-1}, 2^0, 2^1, 2^2, 2^3, 2^4\}$. And the description of the hyperparameter values of kernels are summarized in Table 4.

Fig. 4 Boundary between domains of positive and negative samples corresponding to Cosexp dataset and the classification boundaries of 5 sets of kernels on Cosexp dataset.

In the experiment, the experimental methodology is set as follows: For a given data set, we throw randomly two-thirds of samples into the training set and the remaining one-third into the testing set, the stratified sampling is exceeded in this program. For single dataset, five kernel functions are adopted separately. To ensure the objectivity of experimental results, we run 10 randomly independent performances to evaluate the classification performance by the best classification accuracy rate by average. The average classification accuracies with standard deviations over 10 trials using the optimal parameter obtained by cross validation are summarized in Table 5. In Table 5, the bold font denotes the best classification performance across the kernel functions compared. The t-test results are summarized in Win-tie-loss (W-T-L), and they are attached at the bottoms of Table 5.

According to the prediction accuracy in Table 5, we note that the AtanK kernel obtains the best accuracy on 6 out of 11 datasets. The ERBF kernel and the UKF kernel fall behind, give better performance on 2 datasets, and the Gaussian kernel provides the best accuracy on only one dataset. Of these five kernel functions, the polynomial kernel has the worst behavior in the experiment. The W-T-L summarization shows that the AtanK kernel has an obvious advantage compared with the Poly kernel, the Gauss kernel, the ERBF kernel and the

Table 3 The specification of data sets for experiments.

Data set	Number of features	Number of samples	Number of classes
Heart	13	270	2
Ringnorm	20	1000	2
German	24	1000	2
Wdbc	30	569	2
Ionosphere	34	351	2
Sonar	60	208	2
Splice	60	1000	2
Australian	14	690	2
Twonorm	20	1000	2
Yeast	8	1136	2
IJK	16	2241	2

Table 4 The values corresponding to kernel parameters in experiments.

Kernel	Parameters	Values
Poly	n	1,2,3,4,5,6,7,8
Gauss	σ	$2^{-4}, 2^{-3}, 2^{-2}, 2^{-1}, 2^0, 2^1, 2^2, 2^3, 2^4$
ERBF	α	$2^{-4}, 2^{-3}, 2^{-2}, 2^{-1}, 2^0, 2^1, 2^2, 2^3, 2^4$
AtanK	n	$2^{-4}, 2^{-3}, 2^{-2}, 2^{-1}, 2^0, 2^1, 2^2, 2^3, 2^4$
UKF	n	$2^{-4}, 2^{-3}, 2^{-2}, 2^{-1}, 2^0, 2^1, 2^2, 2^3, 2^4$
	a	1

Table 5 The classification accuracy of different kernels on benchmark data sets (mean(%) $\pm$ std).

Data set	Poly	Gauss	ERBF	UKF	AtanK
Heart	76.00 $\pm$ 0.0457	78.44 $\pm$ 0.0393	78.59 $\pm$ 0.0149	77.22 $\pm$ 0.0417	78.78 $\pm$ 0.0376
Ringnorm	95.09 $\pm$ 0.0102	96.29 $\pm$ 0.0049	98.44 $\pm$ 0.0024	97.37 $\pm$ 0.0457	97.84 $\pm$ 0.0070
German	72.72 $\pm$ 0.0184	73.38 $\pm$ 0.0103	73.05 $\pm$ 0.0252	73.26 $\pm$ 0.0192	74.16 $\pm$ 0.0073
Wdbc	94.74 $\pm$ 0.0096	97.00 $\pm$ 0.0145	97.21 $\pm$ 0.0050	97.74 $\pm$ 0.0096	96.84 $\pm$ 0.0253
Ionosphere	86.15 $\pm$ 0.0248	88.37 $\pm$ 0.0092	92.91 $\pm$ 0.0121	92.48 $\pm$ 0.0184	91.28 $\pm$ 0.0212
Sonar	91.43 $\pm$ 0.0316	92.86 $\pm$ 0.0363	90.43 $\pm$ 0.0415	92.43 $\pm$ 0.0234	91.29 $\pm$ 0.0549
Splice	80.36 $\pm$ 0.0401	82.99 $\pm$ 0.0229	81.95 $\pm$ 0.0183	83.83 $\pm$ 0.0268	84.55 $\pm$ 0.0130
Australian	82.74 $\pm$ 0.0334	81.57 $\pm$ 0.0148	83.48 $\pm$ 0.0319	83.70 $\pm$ 0.0247	82.87 $\pm$ 0.0340
Twonorm	96.41 $\pm$ 0.0101	96.32 $\pm$ 0.0047	96.26 $\pm$ 0.0046	96.38 $\pm$ 0.0065	96.59 $\pm$ 0.0102
Yeast	66.78 $\pm$ 0.0144	65.80 $\pm$ 0.0091	67.28 $\pm$ 0.0270	66.33 $\pm$ 0.0210	67.47 $\pm$ 0.0204
IJK	96.80 $\pm$ 0.0045	98.11 $\pm$ 0.0091	98.14 $\pm$ 0.0087	98.30 $\pm$ 0.0054	98.26 $\pm$ 0.0076
W-T-L	10-0-1	7-3-1	9-0-2	7-2-2	—

Fig. 5 Result of Friedman's test of the testing accuracy.

UKF kernel on 10 datasets, 7 datasets, 9 datasets and 7 datasets out of 11 datasets in Table 5.

Besides comparing the testing accuracy of these kernels for each data set, we take an overall comparison on 11 UCI data sets by combining the nonparametric Friedman's test and the Tukey's honestly significant difference criterion. The Friedman's test is an extension of the sign test and is better known, it is widely useful for a much larger range of treatments [27]. The overall evaluation of the classification accuracy of these kernels is displayed in Fig. 5.

Seen from Fig. 5, the performance of the polynomial kernel is outperformed by other kernel functions. The performance of the proposed AtanK has the best rank of these five kernel functions, and it has a statistically significantly accuracy on 11 datasets. It means the AtanK kernel is a better and robust kernel compared with the polynomial kernel, the ERBF kernel, the Gaussian kernel, and the UKF kernel.

5 Conclusion

Kernel selection is fundamental to kernel based methods. Driven by the similarity measure defined by the arctangent function, we described a new universal kernel in this paper. Comprehensive experimental shows the new proposed AtanK kernel can properly learn separating boundary for many classification application. Schematic diagrams show the AtanK kernel is a new set of SVM kernels that allows a quick decreasing around the sample and a moderate decreasing toward infinity.

Experiments done on the checkerboard dataset and the Cosexp dataset show the ability of the AtanK kernel in separating the patterns. Thus we find the new kernel can preserve the all data on closeness information while still penalize the far neighborhood more reliably. Moreover, a comparison is done with previously constructed SVM models based on the UCI benchmark datasets, which was obtained by using the polynomial kernel, the Gaussian kernel, the ERBF kernel, and the UKF kernel. The result revealed that the AtanK kernel can improve classification accuracy.

Further numerical experiments are required in order to assess the power of our evolved kernel. And an automatic algorithm of the AtanK kernel parameters optimization will be valuable. Many mature kernel parameter optimization tricks can extend to the new kernel easily.

Acknowledgements: The work described in this paper was partially supported by Open Project Foundation of Intelligent Information Processing Key Laboratory of Shanxi Province (2017003) and the Doctoral research initiation fund of Shandong University of Technology (4041-418047).

References

- [1] V. N. Vapnik, 1. (1998), *The Nature of Statistical Learning Theory*, 2nd ed., Springer-Verlag, New York, USA, pp. 493–520.
- [2] B. Schölkopf, A.J. Smola, K.-R. Müller, 2. (1999), Kernel principal component analysis," in: M. Press (Ed.), *Advances in kernel methods: support vector learning*, pp.327–352.
- [3] J. Liu, X Liu, J Xiong, et al., 3. (2020), Optimal Neighborhood Multiple Kernel Clustering with Adaptive Local Kernels. *IEEE Transactions on Knowledge and Data Engineering*, pp. 99.
- [4] S. Mika, et al., 4. (1999), Fisher discriminant analysis with kernels, *Neural Networks for Signal Processing IX*", in: *Proceedings of the 1999 IEEE Signal Processing Society Workshop*, pp.41–48.
- [5] D.R. Hardoon, S. Szedmak, J. Shawe-Taylor, 5. (2004), Canonical correlation analysis: an overview with application to learning method, *Neural Computation*, **16**(12), pp.2639–2664.
- [6] G. Baudat, F. Anouar, 6. (2000), Generalized discriminant analysis using a kernel approach, *Neural Computation*, **12**(10), pp.2385–2404.
- [7] J. Shawe-Taylor, N. Cristianini, 7. (2004), *Kernel Methods for Pattern Analysis*, Cambridge University Press, Cambridge, pp. 25–46.
- [8] L. Sun, et al., 8. (2018), Research on parameter selection method for support vector machines, *Applied Intelligence*, **48**, pp. 331–342.
- [9] Y. Wang, et al., 9. (2017), Multiple kernel learning with hybrid kernel alignment maximization, *Pattern Recognition*, **70**, pp. 104–111.
- [10] S. Yin, J. Yin, 10. (2016), Tuning kernel parameters for SVM based on expected square distance ratio, *Information Sciences*, **370-371**, pp. 92–102.
- [11] F. Ong, P. Milanfar and P. Getreuer, 11. (2019), Local kernels that approximate bayesian regularization and proximal operators, *IEEE Transactions on Image Processing*, **28**(6), pp. 3007–3019.
- [12] T. Wang, W. Li, 12. (2018), Kernel learning and optimization with Hilbert-Schmidt independence criterion, *International Journal of Machine Learning and Cybernetics*, **9** pp. 1707–1717.
- [13] J. Cervantes, X. Li, W. Yu, 13. (2014), Imbalanced data classification via support vector machines and genetic algorithms, *Connection Science*, **26**(4), pp. 335–348.
- [14] M. N. Kapp, R. Sabourin, P. Maupin, 14. (2012), A dynamic model selection strategy for support vector machine classifiers, *Applied Soft Computing*, **12**(8), pp. 2550–2565.
- [15] C. Cortes, M. Mohri, A. Rostamizadeh, 15. (2010), Two-stage learning kernel algorithms, in: *Proceedings of the 27th International Conference on Machine Learning*, pp. 239–246.
- [16] S. Niazmardi, A. Safari, S. Homayouni, 16. (2017), Similarity-based multiple kernel learning algorithms for classification of remotely sensed images, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, **10**(5), pp. 2012–2021.
- [17] R. Kannao, P. Guha, 17. (2017), Success based locally weighted Multiple Kernel combination," *Pattern Recognition*, **68**, pp. 38–51.
- [18] A. Jacot, F. Gabriel, C. Hongler, 18. (2018), Neural tangent kernel: convergence and generalization in neural networks, in: *Advances in neural information processing systems*, pp. 8571–8580.
- [19] J. Basak, 19. (2008), A least square kernel machine with box constraints, in: *Proceeding of the International Conference on Pattern Recognition*, pp. 1–4.
- [20] N. E. Ayat, et al. 20. (2001), KMOD-A new support vector machine kernel with moderate decreasing for pattern recognition, application to digit image recognition, in: *Proceeding of the 6th International Conference on document analysis recognition*, pp. 1215–1221.
- [21] R. Zhang, W. Wang, 21. (2011), Facilitating the applications of support vector machine by using a new kernel, application to digit image recognition, *Expert Systems with Applications*, **38**(11), pp. 14225–14230.
- [22] J. Zhao, et al., 22. (2013), An adaptive support vector regression based on a new sequence of unified orthogonal polynomials, *Pattern Recognition*, **46**(3), pp. 899–913.
- [23] V. H. Moghaddam, J. Hamidzadeh, 23. (2013), New Hermite orthogonal polynomial kernel and combined kernels in support vector machine classifier, *Pattern Recognition*, **60**, pp. 921–935.
- [24] M. Tian, W. Wang, 24. (2017), Some sets of orthogonal polynomial kernel functions, *Applied Soft Computing*, **61**, pp. 742–756.
- [25] A. Rakotomamonjy, et al., 25. (2008), SimpleMKL, *Journal of Machine Learning Research*, **9**, pp.2491–2521.
- [26] D. Dua, C. Graff, 26. (2019), UCI Machine Learning Repository, <http://archive.ics.uci.edu/ml>, Irvine, CA: University of California, School of Information and Computer Science.
- [27] M. Gönen, E. Alpaydın, 27. (2011), Multiple kernel learning algorithms, *Journal of Machine Learning Research*, **12**, pp. 2211–2268.