# PLAME: Piecewise-Linear Approximate Measure for Additive Kernel SVM (Extended abstract)

Tsz Nam Chan*, Zhe Li†, Leong Hou U‡, Reynold Cheng§
*Department of Computer Science and Software Engineering, Shenzhen University
edisonchan@szu.edu.cn
†Alibaba Cloud
huoju.lz@alibaba-inc.com
‡State Key Laboratory of Internet of Things for Smart City
‡Department of Computer and Information Science, University of Macau
ryanlhu@um.edu.mo
§Department of Computer Science and Musketeers Foundation Institute of Data Science, The University of Hong Kong
ckcheng@cs.hku.hk

***Abstract*—Additive Kernel SVM has been extensively used in many applications, including human activity detection and pedestrian detection. Since training an additive kernel SVM model is very time-consuming, which is not scalable to large-scale datasets, many efficient solutions have been developed in the past few years. However, most of the existing methods normally fail to achieve one of these three important conditions which are (1) low classification error, (2) low memory space, and (3) low training time. In order to simultaneously fulfill these three conditions, we develop the new piecewise-linear approximate measure (PLAME) for training additive kernel SVM models. Experimental results verify that this approach can achieve the best trade-off between accuracy, memory space, and training time compared with different types of state-of-the-art methods.**

## I. Introduction

Kernel functions have been extensively used in different communities, including database [3], [6], [7], data mining [5], machine learning [15], and computer vision [9], [11], [13], [14], for supporting various fundamental tasks (e.g., classification, clustering, and visualization). Recently, additive kernels [5], [11], [14] have received significant attention due to a wide range of applications. Human activity detection systems [12] utilize SVM models with additive kernels to predict human activities, e.g., walk and sit down, from sensor data. Pedestrian detection systems [13] utilize SVM models with additive kernels to detect pedestrians in an image. In above studies, the scholars believe that using SVM models with additive kernels can generally provide superior performance in their applications.

However, training SVM models with additive kernels is computationally expensive, which cannot be scalable to support large-scale datasets. Although various types of (exact and approximate) methods have been developed for tackling this issue, all of them cannot simultaneously fulfill these three conditions, which are (1) low classification error (i.e., high accuracy), (2) low memory consumption, and (3) low training time (cf. Table I).

This work was supported by the NSFC grant 62202401, the Science and Technology Development Fund Macau SAR (0052/2023/RIA1, 0031/2022/A, 001/2024/SKL), the Research Grant of University of Macau (MYRG2022-00252-FST), Wuyi University Hong Kong and Macau joint Research Fund (2021WGALH14), the Hong Kong Jockey Club Charities Trust (Project 260920140), the University of Hong Kong (Project 109000579), the France/Hong Kong Joint Research Scheme 2020/21 (Project F-HKU702/20), and the HKU Outstanding Research Student Supervisor Award 2022-23.

TABLE I: Different methods for training SVM with additive kernels.

| Method | Classification error | Memory space | Training time |
|---|---|---|---|
| Kernel SVM solver [8] | low | low | high |
| Linear SVM solver [10] | high | low | low |
| Feature approximation [2], [9], [11], [14] | high | high | low |
| Function approximation [15] | high | low | low |
| PLAME (ours) [4] | low | low | low |

Therefore, we develop the new similarity measure, called Piecewise-Linear Approximate MEasure (PLAME), which directly utilizes the piecewise-linear function to approximate additive kernels. By incorporating PLAME into the linear SVM solver (with slight modification), we show that this approach can fulfill all conditions in Table I for training SVM models with additive kernels.

The rest of the paper is organized as follows. We first provide an overview of additive kernels in Section II. Then, we illustrate the main concept of PLAME in Section III. Lastly, we show some experimental results in Section IV.

## II. Overview of Additive Kernels

Additive kernels have been successfully used in both computer vision [13], [14] and machine learning [15] communities. Given $\mathbf{x}_i$ and $\mathbf{x}$ as two $d$-dimensional vectors, where $x_i^{(\ell)}$ and $x^{(\ell)}$ denote the $\ell^{\text{th}}$ dimensional values of these two vectors, respectively, Table II summarizes four famous additive kernel functions, which are $\chi^2$, JS, intersection, and Hellinger kernels.

Observe that these kernel functions exhibit the following additive property [5], [13] (cf. Definition 1).

**Definition 1.** *If the kernel function $K(\mathbf{x}_i, \mathbf{x})$ is the sum of $d$ one-dimensional kernel functions (denoted as $k(x_i^{(\ell)}, x^{(\ell)})$), $K(\mathbf{x}_i, \mathbf{x})$ is the additive kernel function.*

TABLE II: Four representative additive kernel functions.

| Kernel name | $K(\mathbf{x}_i, \mathbf{x})$ |
|---|---|
| $\chi^2$ | $\sum_{\ell=1}^{d} \frac{2x_i^{(\ell)} x^{(\ell)}}{x_i^{(\ell)} + x^{(\ell)}}$ |
| JS | $\sum_{\ell=1}^{d} \frac{1}{2} x_i^{(\ell)} \log_2 \left( \frac{x_i^{(\ell)} + x^{(\ell)}}{x_i^{(\ell)}} \right) + \frac{1}{2} x^{(\ell)} \log_2 \left( \frac{x_i^{(\ell)} + x^{(\ell)}}{x^{(\ell)}} \right)$ |
| Intersection | $\sum_{\ell=1}^{d} \min(x_i^{(\ell)}, x^{(\ell)})$ |
| Hellinger | $\sum_{\ell=1}^{d} \sqrt{x_i^{(\ell)} x^{(\ell)}}$ |

$$K(\mathbf{x}_i, \mathbf{x}) = \sum_{\ell=1}^{d} k(x_i^{(\ell)}, x^{(\ell)}) \tag{1}$$

As an example, $k(x_i^{(\ell)}, x^{(\ell)}) = \frac{2x_i^{(\ell)} x^{(\ell)}}{x_i^{(\ell)} + x^{(\ell)}}$ for $\chi^2$ kernel function.

## III. PLAME: Piecewise-Linear Approximate Measure

In our work [4], we propose a novel approximate measure, which is $k_{PL}(x_i^{(\ell)}, x^{(\ell)})$, to replace $k(x_i^{(\ell)}, x^{(\ell)})$. As a remark, we denote $x_i^{(\ell)}$ as the $\ell^{\text{th}}$ dimensional value of the $i^{\text{th}}$ training data. First, after we know the region, e.g., $[L(\ell), U(\ell)]$[1], of $x_i^{(\ell)}$ for each dimension, we can partition this region into $P$ intervals $\{I_1, I_2, ..., I_P\}$, where $I_p = [l_p, u_p]$ $(1 \leq p \leq P)$. Then, we can define this piecewise-linear approximate measure (PLAME) based on these intervals (cf. Equation 2).

$$k_{PL}(x_i^{(\ell)}, x^{(\ell)}) = \begin{cases} m_{x^{(\ell)}}(I_1) x_i^{(\ell)} + c_{x^{(\ell)}}(I_1) & \text{if } x_i^{(\ell)} \in I_1 \\ m_{x^{(\ell)}}(I_2) x_i^{(\ell)} + c_{x^{(\ell)}}(I_2) & \text{if } x_i^{(\ell)} \in I_2 \\ \vdots & \vdots \\ m_{x^{(\ell)}}(I_P) x_i^{(\ell)} + c_{x^{(\ell)}}(I_P) & \text{if } x_i^{(\ell)} \in I_P \end{cases} \tag{2}$$

where the slope $m_{x^{(\ell)}}(I_p)$ and the intercept $c_{x^{(\ell)}}(I_p)$ (for $\chi^2$ kernel) are denoted as:

$$m_{x^{(\ell)}}(I_p) = \frac{2{x^{(\ell)}}^2}{(x^{(\ell)} + u_p)(x^{(\ell)} + l_p)} \tag{3}$$

$$c_{x^{(\ell)}}(I_p) = \frac{2x^{(\ell)} l_p u_p}{(x^{(\ell)} + u_p)(x^{(\ell)} + l_p)} \tag{4}$$

## IV. Experimental Results

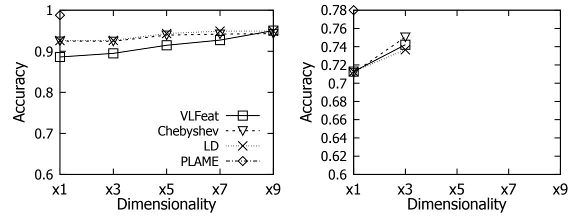
In our experiments, we adopt one small-scale dataset, skin [8], and one large-scale dataset, casas [1], for testing the accuracy and efficiency of different methods (cf. Table I). Observe that our method, PLAME, can achieve low memory space consumption (cf. Figure 1), low classification error (cf. Table III), and low training time (cf. Figure 2) compared with other state-of-the-art methods.

## References


[1] UCI machine learning repository. http://archive.ics.uci.edu/ml/index.php.

[2] M. Boroumand and J. J. Fridrich. Applications of explicit non-linear feature maps in steganalysis. *IEEE TIFS*, 13(4):823–833, 2018.


[1]Since we need to perform normalization before training SVM classifier [8], the region for each dimension is usually $[0, 1]$.

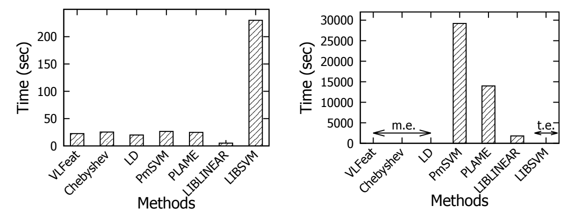



(a) skin (b) casas

Fig. 1: Trade-off between the memory space (i.e., dimensionality of the feature vectors) and the accuracy of feature approximation methods.

TABLE III: Accuracy of all methods (t.e.: the running time of this method takes more than three days and m.e.: the memory consumption of this method is more than 16GB.), where the top-2 accuracy values are in bold type for each dataset and the dimensionality of feature approximation methods is fixed as $\times 7$ of the original dimensionality.

| Method | LIBSVM | LIBLINEAR | VLFeat | Chebyshev | LD | PmSVM | PLAME |
|---|---|---|---|---|---|---|---|
| skin | **0.992** | 0.895 | 0.927 | 0.942 | 0.949 | 0.919 | **0.988** |
| casas | t.e. | 0.71 | m.e. | m.e. | m.e. | **0.728** | **0.78** |




(a) skin (b) casas

Fig. 2: Response time (sec) for all methods in the training phase, fixing the dimensionality of feature approximation methods as $\times 7$ of the original dimensionality.


[3] T. N. Chan, P. L. Ip, L. H. U, B. Choi, and J. Xu. SAFE: A share-and-aggregate bandwidth exploration framework for kernel density visualization. *PVLDB*, 15(3):513–526, 2022.

[4] T. N. Chan, Z. Li, L. H. U, and R. Cheng. PLAME: piecewise-linear approximate measure for additive kernel SVM. *IEEE TKDE*, 35(10):9985–9997, 2023.

[5] T. N. Chan, L. H. U, R. Cheng, M. L. Yiu, and S. Mittal. Efficient algorithms for kernel aggregation queries. *IEEE TKDE*, 34(6):2726–2739, 2022.

[6] T. N. Chan, L. H. U, B. Choi, and J. Xu. SLAM: efficient sweep line algorithms for kernel density visualization. In *SIGMOD*, pages 2120–2134. ACM, 2022.

[7] T. N. Chan, M. L. Yiu, and L. H. U. KARL: Fast kernel aggregation queries. In *ICDE*, pages 542–553, 2019.

[8] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM TIST*, 2:27:1–27:27, 2011.

[9] O. Chum. Low dimensional explicit feature maps. In *ICCV*, pages 4077–4085, 2015.

[10] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *JMLR*, 9:1871–1874, 2008.

[11] F. Li, G. Lebanon, and C. Sminchisescu. Chebyshev approximations to the histogram $\chi^2$ kernel. In *CVPR*, pages 2424–2431, 2012.

[12] W. Li and N. Vasconcelos. Complex activity recognition via attribute dynamics. *IJCV*, 122(2):334–370, 2017.

[13] S. Maji, A. C. Berg, and J. Malik. Efficient classification for additive kernel SVMs. *IEEE TPAMI*, 35(1):66–77, 2013.

[14] A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. *IEEE TPAMI*, 34(3):480–492, 2012.

[15] J. Wu and H. Yang. Linear regression-based efficient SVM learning for large-scale classification. *IEEE TNNLS*, 26(10):2357–2369, 2015.