Neural Networks 63 (2015) 117–132

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

# Approximate kernel competitive learning

Jian-Sheng Wu [a,b], Wei-Shi Zheng [a,c,*], Jian-Huang Lai [a,d]

[a] School of Information Science and Technology, Sun Yat-sen University, Guangzhou 510006, China
[b] SYSU-CMU Shunde International Joint Research Institute, Shunde, China
[c] Guangdong Province Key Laboratory of Computational Science, Guangzhou 510275, China
[d] Guangdong Province Key Laboratory of Information Security, China

## HIGHLIGHTS

- Kernel competitive learning (KCL) cannot be applied in large scale data problem.
- Propose a projection based approximate KCL method for large scale data problem.
- Provide theoretical analysis on why the approximation modelling would work for KCL.
- A pseudo-parallelled approximate computation framework for large scale KCL is developed.
- Experimentally show the effectiveness and efficiency of the proposals.

## ARTICLE INFO

*Article history:*
Received 18 April 2014
Received in revised form 23 September 2014
Accepted 14 November 2014
Available online 27 November 2014

*Keywords:*
Kernel clustering
Kernel competitive learning
Large scale computation

## ABSTRACT

Kernel competitive learning has been successfully used to achieve robust clustering. However, kernel competitive learning (KCL) is not scalable for large scale data processing, because (1) it has to calculate and store the full kernel matrix that is too large to be calculated and kept in the memory and (2) it cannot be computed in parallel. In this paper we develop a framework of approximate kernel competitive learning for processing large scale dataset. The proposed framework consists of two parts. First, it derives an approximate kernel competitive learning (AKCL), which learns kernel competitive learning in a subspace via sampling. We provide solid theoretical analysis on why the proposed approximation modelling would work for kernel competitive learning, and furthermore, we show that the computational complexity of AKCL is largely reduced. Second, we propose a pseudo-parallelled approximate kernel competitive learning (PAKCL) based on a set-based kernel competitive learning strategy, which overcomes the obstacle of using parallel programming in kernel competitive learning and significantly accelerates the approximate kernel competitive learning for large scale clustering. The empirical evaluation on publicly available datasets shows that the proposed AKCL and PAKCL can perform comparably as KCL, with a large reduction on computational cost. Also, the proposed methods achieve more effective clustering performance in terms of clustering precision against related approximate clustering approaches.



## 1. Introduction

Clustering, as an important kind of unsupervised learning approach, plays an important role in discovering the structure of data and exploratory in nature (Jain, 2010). Up to now, there are lots of clustering methods developed for various problems in a wide range of applications, e.g., engineering, computer science, life and medical science, earth science, social science and economics (Xu & Wunsch, 2005). Typical clustering methods are such as $k$-means (MacQueen, 1967), hierarchical clustering, kernel $k$-means (Schölkopf, Smola, & Müller, 1998), and spectral clustering (Ng, Jordan, & Weiss, 2001).

Due to the effectiveness on grouping data, in the past decades, competitive learning has received a lot of attention and has been widely applied in data clustering (Banerjee & Ghosh, 2004; Cottrell, Hammer, Hasenfuß, & Villmann, 2006; Fort, Letremy, & Cottrell, 2002; Inokuchi & Miyamoto, 2006; MacDonald & Fyfe, 2000; Martinetz, Berkovich, & Schulten, 1993; Mizutani & Miyamoto, 2005; Qin & Suganthan, 2004; Schleif, Zhu, & Hammer, 2013; Vesanto & Alhoniemi, 2000; Wang, Lai, & Zhu, 2010, 2012; Xu, Krzyzak, & Oja, 1993). Compared to the traditional iterative clustering algorithms,

* Corresponding author at: School of Information Science and Technology, Sun Yat-sen University, Guangzhou 510006, China. Tel.: +86 020 84110175.
*E-mail addresses:* jiansheng4211@gmail.com (J.-S. Wu), wszheng@ieee.org (W.-S. Zheng), stsljh@mail.sysu.edu.cn (J.-H. Lai).

http://dx.doi.org/10.1016/j.neunet.2014.11.003
0893-6080/

such as $k$-means and kernel $k$-means, competitive learning has the advantages to avoid being trapped in a local minimum resulted by non-optimal initialisations (Inokuchi & Miyamoto, 2006; MacDonald & Fyfe, 2000; Mizutani & Miyamoto, 2005; Qin & Suganthan, 2004; Schleif et al., 2013; Wang et al., 2010) and has the ability to avoid learning extremely small clusters or even empty clusters (Banerjee & Ghosh, 2004) due to the adoption of the online update rule and winner update rule.

Most of the developed competitive learning approaches (Ahalt, Krishnamurthy, Chen, & Melton, 1990; Cottrell et al., 2006; Desieno, 1988; Fort et al., 2002; Kohonen, 1990; Martinetz et al., 1993; Vesanto & Alhoniemi, 2000; Xu et al., 1993) are based on the assumption that the clusters can be linearly separated in the data space. To overcome this shortage, recently, kernel competitive learning (KCL) (Inokuchi & Miyamoto, 2006; MacDonald & Fyfe, 2000; Mizutani & Miyamoto, 2005; Qin & Suganthan, 2004; Wang et al., 2010) and graph based multi-prototype competitive learning (Wang, Lai et al., 2012) have been developed to handle with the nonlinearly separable datasets. By mapping data points from the data space into a much higher or even infinite dimensional space, called the Reproduced Kernel Hilbert Space (RKHS), induced by a kernel function that is usually implicitly defined, as kernel $k$-means, kernel competitive learning would be likely to find linearly separated hyperplane in the RKHS space, which can yield arbitrary clustering shapes in the original data space (Xu & Wunsch, 2005). After initialising using a graph-based method, Wang, Lai et al. (2012) performs a multi-prototype competitive learning to refine the clustering and identify clusters of an arbitrary shape.

However, the large scale computational complexity and space complexity challenge the kernel competitive learning and the graph based multi-prototype competitive learning on dealing with large scale datasets. It is because nowadays, there are huge of data (e.g. world wide webpages, digital images and video surveillance data) created every day due to the development of computer science and information techniques. For example, the amount of digital data created and replicated from now to the year 2020 will reach at least 35 trillion gigabytes (Digital Universe Study, 2010).

To address the high computational complexity and space complexity problems, works in Schleif et al. (2013); Schleif, Zhu, Gisbrecht, and Hammer (2012) approximated the kernel competitive learning by avoiding using the full kernel matrix. By directly applying the Nystroḿ method (Williams & Seeger, 2001) to approximate the kernel matrix when calculating the distances between data points and cluster prototypes, Schleif et al. (2012) reduce the computational complexity and space complexity from $O(\tau cn^2)$ and $O(n^2)$ to $O(nm^2 + m^3 + 2\tau cnm)$ and $O(mn)$, respectively, where $\tau$ is the number of iterations and $m$ is the number of sampled data points. By using the core set points of each cluster, rather than using the whole data points of each cluster, to update the cluster prototypes after the reassignment of cluster labels, Schleif et al. (2013) can avoid calculating the whole kernel matrix and the quadratic computation when calculating the distances between data points and the cluster prototypes. However, it has to calculate the sub kernel matrix when computing the distances between data points and prototypes and costs $O(nd/\epsilon^2)$ (Bădoiu & Indvk, 2002) (or $O(1/\epsilon^8)$ when applying the probabilistic speedup method Tsang, Kwok, & Cheung, 2005), where $\epsilon$ is appropriately set to be $10^{-6}$ (Tsang et al., 2005), to calculate a core set for each cluster in each iterative step, which prevents its application to high dimensional clustering problem and clustering problem with a large number of clusters. For example, it almost costs 3 days to cluster the Caltech 101 data points if $\epsilon$ is set to 0.1 and much more time is used if $\epsilon$ is set smaller. Moreover, as stated in Fort et al. (2002), due to the batch update of the cluster prototypes after the reassignment of cluster labels for data points, both of them are depending on the initialisation and will generate imbalanced clusters. Note that imbalanced clustering will likely yield extremely small clusters or empty clusters (Bradley, Bennett, & Demiriz, 2000; Kashima, Ide, Kato, & Sugiyama, 2009). This can be a considerable concern for clustering problems with a large number of clusters and a high dimensionality (Bradley et al., 2000). Although the sizes of clusters may differ differently, more balanced clustering (or say less imbalanced clustering) is preferred in several real-life applications, such as large retail chains, and marketing campaign (Banerjee & Ghosh, 2004). In addition, balanced clustering is helpful to alleviate the sensitivity to clustering initialisation and avoid outlier clusters (Banerjee & Ghosh, 2004) as well.

In the clustering literature, there are also related works on large scale clustering. These works include: (1) $k$-means algorithm (Huang, 1998; Nistér & Stewénius, 2006; Ordonez & Omiecinski, 2004; Philbin, Chum, Isard, Sivic, & Zisserman, 2007; Wang, Wang, Ke, Zeng, & Li, 2012), (2) data sampling and summarisation techniques (Guha, Rastogi, & Shim, 1998; Kaufman & Rousseeuw, 2005; Ng & Han, 2002; Zhang, Ramakrishnan, & Livny, 1996), (3) distributed models (Chen, Song, Bai, Lin, & Chang, 2011; Cordeiro et al., 2011; Ene, Im, & Moseley, 2011) and incremental clusterings (Lühr & Lazarescu, 2009; Zhang, Liu, & Wang, 2007), (4) random sampling for approximating some specific non-linear kernels (Chitta, Jin, & Jain, 2012; Kar & Karnick, 2012; Pham & Pagh, 2013; Rahimi & Recht, 2007), and (5) data sampling for kernel $k$-means (Chitta, Jin, Havens, & Jain, 2011) and spectral clustering (Fowlkes, Belongie, Chung, & Malik, 2004; Williams & Seeger, 2001). However, these works have their weaknesses in one of the following points: (1) relying on the assumption about linearly separable model (Cordeiro et al., 2011; Ene et al., 2011; Guha et al., 1998; Huang, 1998; Kaufman & Rousseeuw, 2005; Ng & Han, 2002; Nistér & Stewénius, 2006; Ordonez & Omiecinski, 2004; Philbin et al., 2007; Wang, Wang et al., 2012; Zhang et al., 1996), (2) not scalable for nonlinear extension (Nistér & Stewénius, 2006; Philbin et al., 2007; Wang, Wang et al., 2012), and (3) incurring several limitations, such as sensitive to prototype initialisation, trapped in local minimum and yielding imbalanced partition (Chitta et al., 2011, 2012; Kar & Karnick, 2012; Pham & Pagh, 2013; Rahimi & Recht, 2007). Although kernel competitive learning can help alleviate the above limitations (1) and (3), it is indeed necessary to develop a scalable approach for kernel competitive learning for processing large scale data.

In this work, in order to make kernel competitive learning tractable for large scale data, we propose an approximate kernel competitive learning in this paper for pursuing robust approximate kernel competitive learning for clustering large scale data in a nonlinear way. In order to accelerate the large scale learning, we further propose a pseudo-parallelled approximate kernel competitive learning framework based on a set-based kernel competitive learning strategy.

### 1.1. Contributions

Motivated by the approximation idea used in approximate kernel $k$-means that endows low time and space complexity for clustering, in this paper, we wish to combine kernel competitive learning and the approximation idea together so as to develop a large scale clustering method. However, it does not mean it is straightforward to apply the approximation idea used in approximate kernel $k$-means to kernel competitive learning, because we will analyse that a direct use would contradict the constraint used in the approximation idea that the prototypes must be bounded in a sampled subspace, which is necessary to avoid the calculation and storing of the full kernel matrix. In order to solve this problem, we introduce the projection strategy into the approximation framework and combine it with kernel competitive learning so as to develop a novel large scale kernel competitive learning