Image and Vision Computing 57 (2017) 89–101

Contents lists available at ScienceDirect

Image and Vision Computing

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

On the effect of hyperedge weights on hypergraph learning☆

Sheng Huang[a, b, *], Ahmed Elgammal[c], Dan Yang[a, b]

[a]Ministry of Education Key Laboratory of Dependable Service Computing in Cyber Physical Society, Chongqing 400044, China
[b]School of Software Engineering at Chongqing University, Chongqing 400044, China
[c]Department of Computer Science at Rutgers University, Piscataway, NJ 08854, USA

ARTICLE INFO

Article history:
Received 2 February 2015
Received in revised form 21 October 2016
Accepted 22 October 2016
Available online 27 November 2016

Keywords:
Hypergraph learning
Transductive learning
Graph learning
Image clustering
Image classification

ABSTRACT

Hypergraph is a powerful representation for several computer vision, machine learning, and pattern recognition problems. In the last decade, many researchers have been keen to develop different hypergraph models. In contrast, no much attention has been paid to the design of hyperedge weighting schemes. However, many studies on pairwise graphs showed that the choice of edge weight can significantly influence the performances of such graph algorithms. We argue that this also applies to hypergraphs. In this paper, we empirically study the influence of hyperedge weights on hypergraph learning via proposing three novel hyperedge weighting schemes from the perspectives of geometry, multivariate statistical analysis, and linear regression. Extensive experiments on ORL, COIL20, JAFFE, Sheffield, Scene15 and Caltech256 datasets verified our hypothesis for both classification and clustering problems. For each of these classes of problems, our empirical study concludes with suggesting a suitable hypergraph weighting scheme. Moreover, the experiments also demonstrate that the combinations of such weighting schemes and conventional hypergraph models can achieve competitive classification and clustering performances in comparison with some recent state-of-the-art algorithms.

1. Introduction

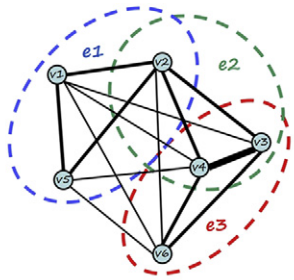
Hypergraph is a commonly used machine learning technique, which represents the structure of data via measuring the similarity between groups of points [1–10]. As a generalization of a graph, the edge of hypergraph (hyperedge) can own any number of vertices rather than two vertices as a graph (Fig. 1 gives an example of hypergraph). This property endows hypergraph a strong descriptive ability of data, particularly, for depicting the complex high-order data relation, since each edge often represents a data relation in graph learning. According to this merit, hypergraph is not only as same as a graph which is a general way to address the fundamental learning tasks, such as, classification and clustering [6,11-13], but also is a suitable technique to address the recently hot computer vision and machine learning issues, such as attribute learning [14], multi-label learning [15–17], multi-view learning [18,19], matching [20–22] and image annotation [1,23].

Recently, some impressive hypergraph models have been proposed [2,8,11,25-27], and their hypergraph learning approaches were successfully applied to tackle extensive tasks [9,10,28-31]. Generally speaking, hypergraph models can be roughly divided into two categories. The first category uses spectral clustering techniques to partition the vertices via constructing a simple pairwise graph from the original hypergraph. Representative methods include clique expansion [25], star expansion [25] and clique averaging [26], etc. Approaches in the second category define a hypergraph Laplacian using analogies from the simple pairwise graph Laplacian. Representative methods in this category include Zhou's normalized Laplacian [11], Bolla's Laplacian [27], etc. However, interestingly, as was shown in Ref. [32], all of the previous algorithms, despite their very different formulations, can be reduced to two graph constructions, the star expansion and the clique expansion, and they are equivalent to each other under specific conditions.

Besides hypergraph models which mainly focus on hypergraph partition, the quality of hypergraph is also an important factor that affects the performance of hypergraph learning in different applications. A good hypergraph should well reflect the real relations of data. So, in the last decade, there also exist several studies about hypergraph construction, e.g. Refs. [11,25-27,33-35]. But, to the best of our knowledge, there are no prior research that formally discusses the effect of the strategy used for assigning hyperedge weights in hypergraph learning, which we call "weighting scheme". In this paper, we aim at addressing this gap. In graph learning, which is the

☆ This paper has been recommended for acceptance by Sven Dickinson, PhD.
* Corresponding author.
E-mail address: huangsheng@cqu.edu.cn (S. Huang).

http://dx.doi.org/10.1016/j.imavis.2016.10.009
0262-8856/



(a)

	e1	e2	e3	e4	e5		e16	e17	e18
v1	1	0	0	1	1	•••	0	0	0
v2	1	1	0	1	0		0	0	0
v3	0	1	1	0	1		1	1	0
v4	0	1	1	0	0		1	0	0
v5	1	0	0	0	0		0	1	1
v6	0	0	1	0	0		0	0	1

(b)

Fig. 1. (a) An example of hypergraph which has 18 hyperedges (15 pairwise edges + 3 three-order hyperedges), and (b) its corresponding vertex–edge incident matrix which is a common tool to depict a hypergraph. (This figure is referenced from our earlier work [24].)

pairwise case of hypergraph, extensive studies have already shown that the choice of edge weight can significantly affect the results of the graph-based algorithms. The Heat-Kernel and Dot-Product weighting schemes are considered as the two most representative weighting schemes of edges [36–39]. Therefore, we hypothesize that the choice of hyperedge weights also plays a crucial role in hypergraph learning. This motivated us to investigate if there exists a preferable hyperedge weighting scheme in hypergraph learning. Moreover, we believe that different choices of hyperedge weights provide alternative ways to explain the hypergraph from different perspectives. In this paper, we empirically discuss the influence of the choice of hyperedge weight to hypergraph learning via presenting and evaluating three novel hyperedge weighting schemes.

As several hypergraph algorithms have been proposed, a few hyperedge weighting schemes have been heuristically and marginally mentioned in such papers. For example, Huang et al. [3] proposed a probabilistic hypergraph-based image retrieval system. In this system, the hyperedge is generated by *k*-nearest neighbor searching, and its weight is the sum of the pairwise edge weights between the centroid (seed point) of hyperedge and its neighbors. Zhang et al. [7] presented an unsupervised hypergraph-based feature selection method, which measures the high-order similarity of the vertices in a hyperedge using multidimensional interaction information (MII). For addressing 3-D object retrieval, Gao et al. [2] calculated the hyperedge weight via directly summing the weights of all pairwise edges whose end points are all in the same hyperedge. Clearly, the computation of such hyperedge weight is actually the inverse process of the clique expansion. So, if we use the mean operation to replace the sum operation, such way will be the inverse process of the clique averaging. In contrast to the previous three methods, Yu et al. [6] defined the hyperedge weight as a parameter of the hypergraph learning model via imposing a sparsity constraint. Thus, the hyperedge weights can be adaptively learned as the graph model optimized. The initial hyperedge weights of this method are constructed by following Huang's way [3], and the global optimal weights still cannot be guaranteed. Certainly, there are other hyperedge weighting schemes [4,5], but most of them are associated with very specific tasks.

Complementary to the previously proposed hyperedge weights, we carefully design three novel hyperedge weights from the perspectives of geometry, multivariate statistical analysis and linear regression [40,41] (see Fig. 2). First, from the perspective of geometry, a hyperedge can be regarded as a high-order simplex [32]. Thus, the volume of simplex is an intuitive hyperedge weight, which provides a reasonable dissimilarity measure for a point set. Motivated by some studies from geometry [42], we present three ways to compute the volume of the simplex for different situations. It is worthwhile to note that these three ways actually define the mathematical relationships between hyperedges and vertices, a hyperedge and its pairwise edges, and a hyperedge and its sub-hyperedges, respectively. Second, from the perspective of data mining and multivariate statistical analysis, a hyperedge can be naturally regarded as a cluster in the sample space, thus the trace of the scatter matrix of the samples in the same hyperedge should be a good hyperedge weight. Finally, from the perspective of linear regression [40,41], the linear reconstruction error of homogenous samples should be smaller than that of inhomogeneous samples. So, we consider a hyperedge as a small subset of samples, and use the local linear reconstruction error of each point in the hyperedge to measure the similarity of the point set.

In order to verify the importance of hyperedge weighting scheme in hypergraph learning, three state-of-the-art hypergraph models including Zhou's normalized Laplacian [11], clique expansion and star expansion [25], are adopted to evaluate the different hyperedge weight selection strategies for two learning problems, namely clustering and classification. Representative hyperedge weighting schemes for classification and clustering are concluded from our experimental results on ORL, COIL20, Sheffield and JAFFE databases. The experimental results also demonstrate that a carefully chosen hyperedge weight can significantly improve the performance

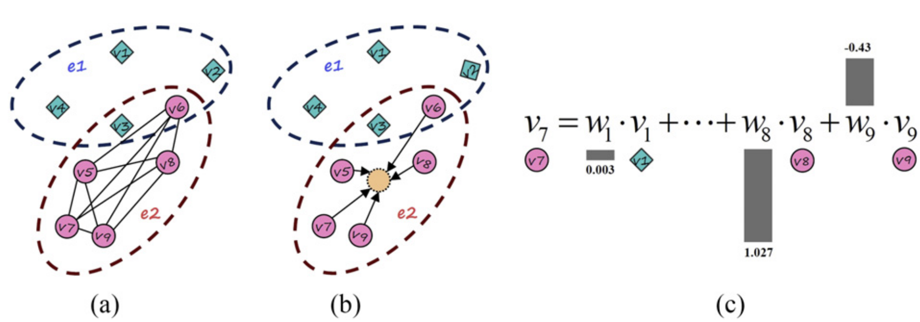


Fig. 2. Three explanations of the hyperedge $e2$, (a) a 4-simplex, (b) a cluster and (c) a linear combination of homogenous vertices.