

RESEARCH

Open Access



# Regularized Simple Graph Convolution (SGC) for improved interpretability of large datasets

Phuong Pho and Alexander V. Mantzaris\* 

\*Correspondence:  
alexander.mantzaris@ucf.edu  
Department of Statistics  
and Data Science, University  
of Central Florida (UCF),  
4000 Central Florida Blvd,  
Orlando 32816, USA

## Abstract

Classification of data points which correspond to complex entities such as people or journal articles is a ongoing research task. Notable applications are recommendation systems for customer behaviors based upon their features or past purchases and in academia labeling relevant research papers in order to reduce the reading time required. The features that can be extracted are many and result in large datasets which are a challenge to process with complex machine learning methodologies. There is also an issue on how this is presented and how to interpret the parameterizations beyond the classification accuracies. This work shows how the network information contained in an adjacency matrix allows improved classification of entities through their associations and how the framework of the SGC provide an expressive and fast approach. The proposed regularized SGC incorporates shrinkage upon three different aspects of the projection vectors to reduce the number of parameters, the size of the parameters and the directions between the vectors to produce more meaningful interpretations.

**Keywords:** Graph neural networks, Big data, Dimensionality reduction, Simple Graph Convolution, Interpretability, Graph Convolutional Network

## Introduction

Research into network based approaches in machine learning has been ongoing with the growing sizes of networks produced by users on commercial online social networking platforms. The ways users engage with each other directly on online social network platforms from independent activities have given rise to networks with billions of users (nodes) [1]. The information provided by the users can be used for investigating different phenomena. The interlinking information that produces edges which then can form a network, or series of networks, has been studied in the growing field of network science [2, 3]. Network science offers many tools and approaches to learn and to draw insight about the community structures [4], the centrality distribution [5] of the nodes (users), and provides models for how the networks can grow from an initial set of nodes [6, 7]. There are many applications for these insights such as in targeted advertising [8] where brands seek to have highly central nodes spread advertising content, and another application is in the effort to understand the ‘landscape’ of political polarization between communities in Twitter [9].

As well as the node interlinking information which can produce a network (graph), there are the attributes of the users which can provide important useful information in improving upon predictive analytics. A notable technique used by online shopping platforms is *collaborative filtering* [10] which works as a recommendation system for improving the shopping experience of customers by optimizing the product view to those items predicted to be of interest. These new link predictions are based upon past purchases and the historical records of other customers. Here 'clusters' are formed between groups of items in a multidimensional space of these choices [11, 12]. This relies on the observation that the items are not selected independently of previous purchases and that there is information gained from utilizing the data collected [13]. Another area where predictive analytics has used information in order to make predictions is in student performance [14]. Logistic regression has been used in situations where a model is required in a decision framework in order to predict achievements [15]. A key difference between the network based approach and these approaches is that the information contained in the network and how the links influence the node of concern are excluded from the model which can play a key role in economic behaviors predicted [16]. In the effort to fuse these sources together for models to incorporate [17] discusses how user identity linkage across online social networks can be accomplished.

Online social networks have been at the forefront of the interest in networks and approaches which use the interlinking information due to their size and the effect they have on human behaviors [18]. The network topologies of the virtual networks can find applications but they also carry over into the physical world. The techniques have been used in other domains such as examining the centrality in streets of urban spaces [19] which also can be seen as a continuation of the original network/topological graph theoretical formulation of Euler's investigation of the 'Seven Bridges of Königsberg' problem [20]. As these urban networks fit within networks of urban spaces themselves, multi-layer networks [21] are produced that span over the globe allowing an analysis of even global migration patterns [22]. In a similar fashion, it is also possible to consider academic literature as a network with similar properties governing its construction, such as homophily [23]. The nodes in such academic networks are publications and the links are the citations between the articles that provide information of association. There is active research in this field [24] which notes the key motivation is that researchers can spend considerable amounts of time searching for the relevant research in order to not allocate time on topics already explored with similar approaches. Being able to find associative research is of importance since it is possible for research to be directed in areas already investigated and waste time as well as materials in research such as studies requiring expensive lab equipment. Navigating the network to extract relevant research is therefore a key activity in preventing this. The work of [25] discusses how the investigators can seek from these datasets insight for the dynamics of the growth and the interconnectivity of scientific thought. With the growth of the citation datasets (such as the ones described in "Data" section) the concerns on the processing time, complexity of the models and the ability to interpret the results are becoming a key issue.

This then poses key questions about how to process and then reason about the results from large datasets with large variations. Questions about the results even require effort in their interpretation. Work such as [26] look at the problem from a conceptual

perspective on the areas of focus for big data and how the user can interact with the data that are results from a post analysis. The work of [27] provides a high level overview of the tools and approaches available in visually investigating the data and the results of different methods return. It is possible to include the full set of interpretable outcomes and the full set of relevant data features, but that does produce a challenge for the practitioner to determine which features are or prime interest. A dimensionality reduction approach [28] provides a more effective experience for the practitioner.

Graph Neural Networks (GNNs), [29], provides a methodological framework for combining node feature and the network data information in order to produce predictions within a machine learning paradigm. There are many applications ranging from image object positions in a non-euclidean space representation, molecular properties and citation networks [30]. Therefore this work deals with investigating an even more simple GNN, the Simple Graph Convolution (SGC) [31], which has a simpler methodological definition and a competitive predictive accuracy. It is in developing a modified SGC that the task of reducing the dimensionality in large datasets with a GNN will be explored. As will be shown in “[Methodology](#)” section, the simplicity of the model allows for it to be a basis for extensions that can incorporate constraints such as shrinkage upon the parameters. This is done in a manner similar to the regularization procedure of Lasso [32]. This will allow the large complex datasets to be processed in such a manner as to be interpretable and more accessible in terms of the computations resources required. Models other than the SGC would incorporate more complexity upon procedures already complex making large dataset investigations an increasingly large challenge to apply. The work effectively takes the SGC and extends it so that the model can introduce constraints upon the parameter vectors in such a manner as to allow the model to be more easily interpreted. This allows the parameters for each class to be more sparse and for each class to have less of an overlap between each other. Altogether this results in a parameter matrix which can more easily be inspected.

In “[Related work](#)” section, a selection of previous work highlights the development of the GNN which led to the SGC is acknowledged. The “[Data](#)” section describes the datasets used in exploring the results of the proposed modification on the SGC with regularization. The methodology is described in “[Methodology](#)” section, where the SGC formalism is presented and the proposed modification where the regularization upon the features and their parameters allows for a reduction of the redundant features and hence ease in the interpretability. The results are displayed in “[Results](#)” section where the ability for the SGC and the proposed SGC allow the model to fit data which would not be linearly separable but is made separable by incorporating the graph information, and then the application to a scientific citation dataset (Cora [33]) is shown.

### **Related work**

Convolutional Neural Networks (CNNs) [34] has brought a methodological approach for handling high dimensional problems more efficiently than other paradigms. As noted in [35] in conjunction with deep learning, CNNs have greatly improved the ability to classify sound and image data. The work of [36] introduces formally how graph based methods can be used with CNNs. A key contribution of [36] is that the extension of the model to generalize to graphs is founded upon localized graph filters instead of the

CNN's localized convolution filter (or kernel). It presents a spectral graph formulation and how filters can be defined in respect to individual nodes in the graph with a certain number of 'hops' distance. These 'hops' are representative of the number of edges traversed between nodes and is the result of the powers of the adjacency matrix where the number of walks can be calculated [5] (walks are paths which allow node revisits).

An introduction to the motivation from basic principals can be found in [37], where the fundamental analysis operations of signals from regular grids (lattice structures) to more general graphs is developed. The authors in [38] utilize the theory of signals on graphs in order to show how a shift-invariant convolution filter can be formulated as a polynomial of adjacency matrices. The discussion of how low pass filters are an underlying principal in the GNN is discussed in [39] which is also described in the work of the SGC. [40] proposes a Graph Convolutional Network (GCN) by adapting Convolutional Neural Networks (CNNs) for graph-structured data. The GCN learns a graph representation via layer-wise propagation rules that represents localized spectral filters.

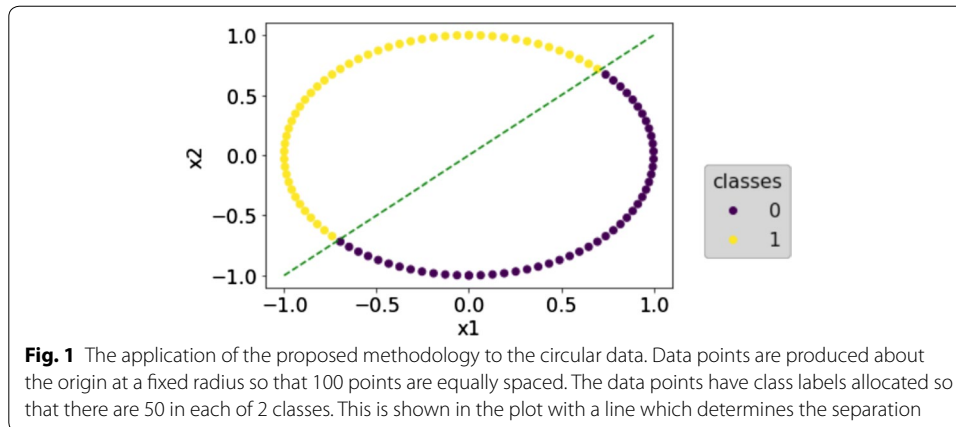
The GNN can allow for the augmentation of a users social network and their features to make a more accurate prediction and similarly for an academic paper that the features (keywords or low dimensional representation) with the citation links can more accurately place its relevance. The machine learning framework can introduce large overheads in the processing time especially for large datasets but fortunately research has shown that simpler GNN models display peak performance [41]. The work of [40] which introduces a semi-supervised approach to GNNs, shows in appendix B the performance of the methodology with the number of 'layers' employed in the model and how there is an actual degradation of the performance after a few layers. The SGC [31] provides an efficient framework to provide a similar model of the data associations as the GCN but avoid the necessity of the layers the GCN introduces. The methodology of the SGC, as shown in "Methodology" section allows a single layer of matrix computations with a non-linear activation function. This is similar to the processing steps taken for logistic regression which can be computed for large datasets very efficiently. Building upon this efficient model allows an investigator to explore further constraints which would be much more computationally demanding with the incorporation of layers.

## Data

Three different datasets are employed in order to explore the model proposed, with 2 of them being synthetic and the last being a real dataset which is well explored [33]. The first synthetic dataset has 2 dimensional features with data points placed in a circle and labels applied on opposite sides of the identity line ( $x_1 = x_2$ ). The other synthetic dataset also has 2 dimensions and placed in such a way which clustering or a non-network based model, relying upon distance measures, would incorrectly classify the node labels. More about these 2 datasets is described below.

### Circular data

Figure 1 shows the synthetic data produced with points allocated along a circle based at the origin. There are 100 points and 50 of them are allocated to each class placing them on either side of the identity line. A key aspect of this data is that the model will attempt to shrink the feature projections which can incur a penalty on the optimization



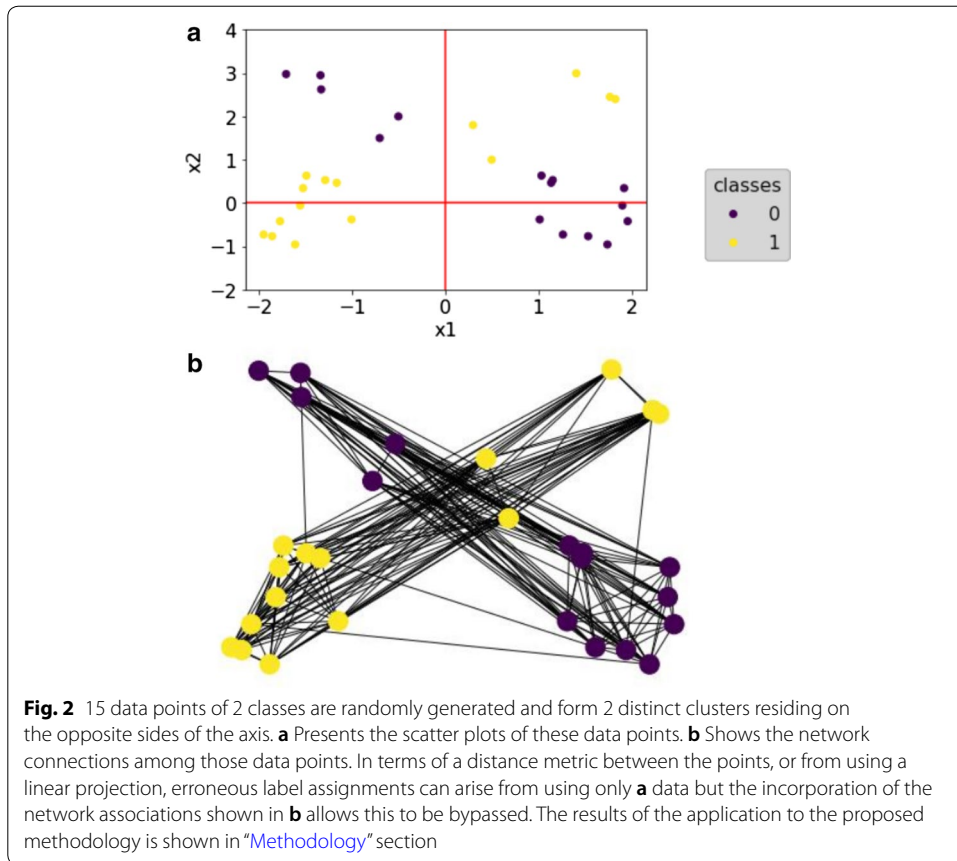
procedure. The compromise between the error function on the data and the regularization penalization term will require a balance as a single feature reduces the shrinkage penalty but the direction for the optimal fit uses both dimensions. This compromise is induced since the optimal projection will be with a vector containing non-negative parameters for each dimension in equal value at a direction  $x_1 = -x_2$  therefore highlighting the shrinkage of one of the parameters. The network data (the adjacency matrix) is a ring network connecting neighboring nodes.

#### Linearly inseparable data

Figure 2 shows 30 synthetically produced data points in 2 dimensions ( $x_1, x_2$ ) which form 4 distinct clusters. Each class has 15 data points randomly generated and it is separated into 2 clusters across the axis. We produce a non-disjoint network (single component) structure for the data points to be connected with a more dense connectivity set between points of the same label. This production is inline with the concept of modularity in networks [42] where the density of the edges between nodes of the same label is proportionately greater than the density between nodes with different labels. Without the network structure, distance metrics would produce erroneous results and the introduction of this information increases the accuracy. This allows the linear operations to produce a separation for the class labels.

#### Methodology

[40] develops Graph Convolutional Networks (GCNs) by adapting Convolutional Neural Networks (CNNs) for graph-structured data and the work of [31] (proposing the Simple Graph Convolution (SGC)) builds upon it. The SGC removes the non-linear transitions between the layers in the model. This simplification speeds up processing time significantly yet still performs on par with GCNs and other state-of-the-art graph neural network models across multiple benchmark graph datasets. The model modification will allow easier interpretability of the parameters fitted by the optimization procedure with the application of a set of constraints. The constraints introduced into the loss function will force the stochastic gradient descent algorithm to find directions which have fewer non-zero values and less overlap for the parameters between the classes. This addresses the problem of how



to inspect effectively the matrix of parameters and the vectors of the parameters for each class. This takes inspiration from regularization methods.

We adopt the notations presented in [40] and [31] for the GCN and SGC respectively. A graph  $G = (V; \mathbf{A})$  can be defined as a collection of nodes (vertexes) set  $V = (v_1, v_2, \dots, v_N)$  containing  $N$  nodes and an adjacency matrix  $\mathbf{A} \in \mathbb{R}^{N \times N}$  where  $a_{ij}$  is the weighted edge between node  $v_i$  and  $v_j$  ( $a_{ij} = 0$  if  $v_i$  and  $v_j$  are not connected). We define the degree matrix  $\mathbf{D} = \text{diag}(d_1, d_2, \dots, d_N)$  as a diagonal matrix whose off-diagonal elements are zero and each diagonal element  $d_i$  capture the degree of node  $v_i$  and  $d_i = \sum_j a_{ij}$ . There is a feature matrix (also referred to as the design matrix)  $X \in \mathbb{R}^{N \times D}$  where each row  $x_i$  is the feature vector measured on each node of the graph. This can be thought of as each row is the feature data belonging to a node, and the columns to a different dimension of the features. Each node  $i$  has a class label from  $C$  classes and hence can be coded as one hot vector  $y_i \in \{0, 1\}^C$ .

The GCNs and SGC add self-loops and normalize the adjacency matrix to get the matrix  $\mathbf{S}$ :

$$\mathbf{S} = \tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \tag{1}$$

where  $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$  and  $\tilde{\mathbf{D}} = \text{diag}(\tilde{\mathbf{A}})$ . This normalization allows successive powers of the matrix to not influence the overall size the projections. The SGC removes non-linear transformation from the  $k$ th-layer of the GCN resulting in a linear model of the form:

$$\hat{Y} = \text{softmax}(\mathbf{S} \dots \mathbf{S} \mathbf{X} \Theta^{(1)} \Theta^{(2)} \dots \Theta^{(K)}). \tag{2}$$

The SGC classifier is then achieved by collapsing the repetitive multiplication of matrix  $\mathbf{S}$  into the  $k$ th power matrix  $\mathbf{S}^K$  and reparameterizing the successive weight matrices as  $\Theta = \Theta^{(1)} \Theta^{(2)} \dots \Theta^{(K)}$ .

$$\hat{Y} = \text{softmax}(\mathbf{S}^K \mathbf{X} \Theta). \tag{3}$$

The parameter  $k$  corresponds to the number of ‘hops’ which is the number of edge traversals in the network adjacency matrix  $\mathbf{S}$ .  $k$  can be thought of as accumulating information from a certain number of hops away from a node (as described visually in [31]). If  $k = 0$  the methodology becomes equivalent to a logistic regression application which is known to be scalable to large datasets. Since the SGC introduces the matrix  $\mathbf{S}$  as linear operation the same scalability applies. The weight matrix  $\Theta$  is trained by minimizing the cross entropy loss:

$$\mathcal{L} = \sum_{l \in \mathcal{Y}_L} \sum_{c \in C} Y_{lc} \ln \hat{Y}_{lc} \tag{4}$$

where  $\mathcal{Y}_L$  is a collection of labeled nodes.

As motivated in “Introduction”, the SGC shows how an efficient formulation of GNNs can be derived, it does not provide as well the ability to reduce the feature set. To reduce the number of parameter values, we introduce a flexible set of constraints as shrinkage operators in the loss for Eq. 4:

$$\begin{aligned} \mathcal{L}_R = \mathcal{L} + L_1 \times \sum_{c \in C} \left( \sum_{d=1}^D |\Theta_{R(\cdot,c)}|^4 \right)^{(-1)} + L_2 \times \sum_{c \in C} \|\Theta_{R(\cdot,c)}\|_2 \\ + L_3 \times \left( \sum_{c_1 \in C} \sum_{c_2 \in C} \left( |\Theta_{R(\cdot,c_1)}^T| \cdot |\Theta_{R(\cdot,c_2)}| : c_1 < c_2 \right) \right) \end{aligned} \tag{5}$$

The first component of  $\mathcal{L}_R$  is the loss from SGC being  $\mathcal{L}$ . Next,  $L_1$  is the shrinkage term for penalizing the number or parameters by reducing the penalization with a larger skew in the number of elements in the columns of  $\Theta_R$ . The term  $|\Theta_{R(\cdot,c)}|^4$  denotes the normalized vector for each class projection in the parameter matrix (which are columns) and that each element is raised to the power of 4. The  $L_2$  term is the total magnitude of the parameter vector so that the distribution of the terms are not influential but only the norm result. The term  $L_3$  is the term which penalizes class label projection which have large overlaps, so that vectors will be orthogonal or depending upon the value of  $L_3$  to support opposing directions. The parameters for the regularized fit using the shrinkage in the loss will be referred to as  $\Theta_R$ . To impose an orthogonality constraint between the projection vectors the term for the  $L_3$  is modified:



$$\begin{aligned} \mathcal{L}_R = & \mathcal{L} + L_1 \times \sum_{c \in C} \left( \sum_{d=1}^D |\Theta_{R(\cdot, c)}|^4 \right)^{(-1)} + L_2 \times \sum_{c \in C} \|\Theta_{R(\cdot, c)}\|_2 \\ & + L_3 \times \left( \sum_{c_1 \in C} \sum_{c_2 \in C} \left( |\Theta_{R(\cdot, c_1)}^T| \cdot |\Theta_{R(\cdot, c_2)}| : c_1 < c_2 \right)^2 \right). \end{aligned} \quad (6)$$

This methodology therefore delivers a formulation which is based upon an approach with layers as other ‘deep learning’ frameworks provide, but without the computational burdens that come along with it. The simplified model implementation is therefore capable to be run on a personal computer with Pytorch [43].

## Results

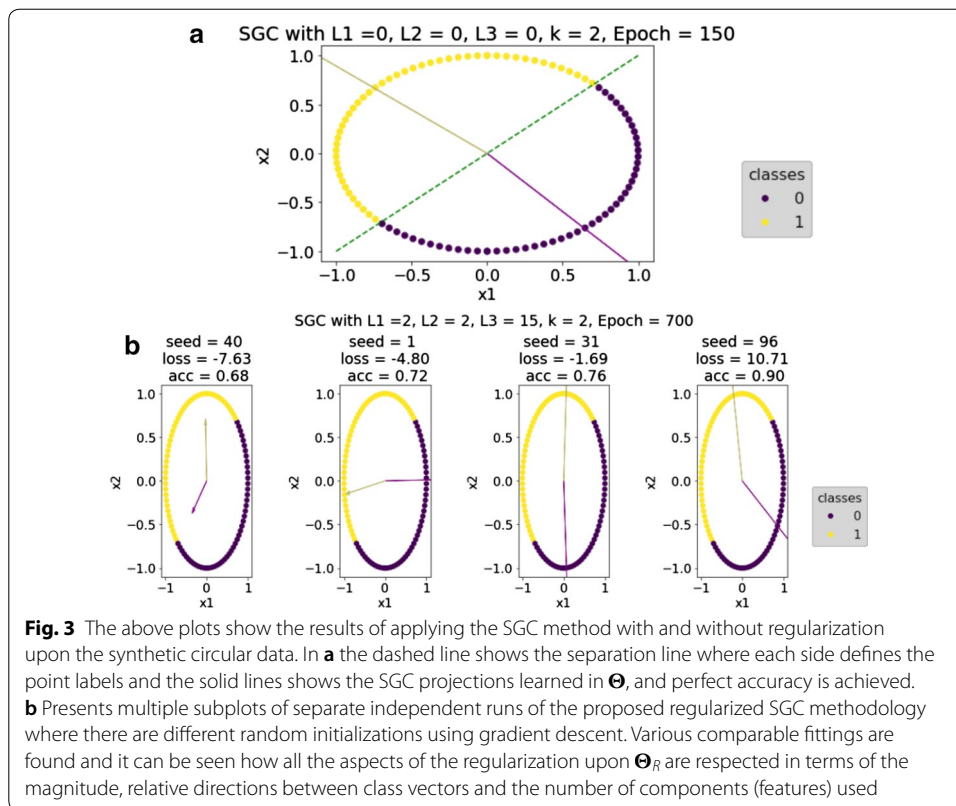
Here we present the results of applying the proposed methodology to the datasets described in “Data” section. The synthetic circularly placed datapoints with labels allocated on the sides of the identity line of 2 dimensions, described in “Circular data” section of the “Data” section. The synthetic datapoints placed along 2 dimensions without a linear separation of the labels based upon a distance metric but possible with the network information is described in “Linearly inseparable data” section and the results for it shown in the subsection of “Results” section, “Synthetic linearly inseparable data”. The results of the application to the real dataset of [33] (Cora citation dataset) is shown in the subsection of “Results” section “Application to the Cora dataset”. The methods of logistic regression, SGC and the regularized SGC are applied and the results are compared revealing that the fitted parameter vectors for each class have less overlap between themselves so that their characteristics for the classes can be more effectively interpreted.

### Synthetic circular data

The results of applying the SGC methodology with and without regularization on the synthetic circular data, is shown in Fig. 3. The points in the dataset have 2 features  $x_1$  and  $x_2$ . In Subfigure (a) the SGC model produces a perfect accuracy with 2 parameters used the projection vectors pointing to the proper direction of each class. The regularized SGC returns different solutions due to the regularization in the parameter matrix  $\Theta$ . Subfigure (b) shows the regularized parameter vectors under different initializations of the learning algorithm which applies constraints. These constraints reduce the number of parameters used, the size of the vectors as a norm, and the direction between the vectors to be more informative. It can be seen how different random initializations produce different loss values and accuracy depending upon the local optima arrived at. These different stable points do show that the shrinkage factors are affecting the vectors for each class in  $\Theta$ .

A separation based upon the identity line for the 2 dimensions, represents a situation where there is equal weight upon all the features of the data and the inference scheme must make a choice in the penalization. The choice results in a decrease in the loss of accuracy in order to decrease the penalization from the regularization from the 3 components calculated from the projections;  $L_1$ ,  $L_2$  and  $L_3$  as discussed in “Methodology” section. The variation shows that the model is able to explore a with range of vectors for





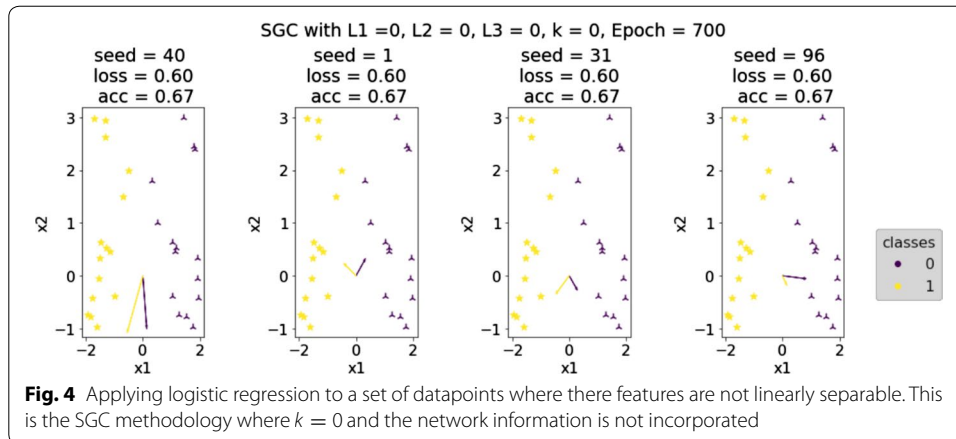
**Fig. 3** The above plots show the results of applying the SGC method with and without regularization upon the synthetic circular data. In **a** the dashed line shows the separation line where each side defines the point labels and the solid lines shows the SGC projections learned in  $\Theta$ , and perfect accuracy is achieved. **b** Presents multiple subplots of separate independent runs of the proposed regularized SGC methodology where there are different random initializations using gradient descent. Various comparable fittings are found and it can be seen how all the aspects of the regularization upon  $\Theta_R$  are respected in terms of the magnitude, relative directions between class vectors and the number of components (features) used

the matrix columns of  $\Theta_R$ . From the range the choice with the largest accuracy (lowest loss) can be chosen.

### Synthetic linearly inseparable data

In this subsection, we apply the SGC method with and without regularization on the linearly inseparable data which contains feature coordinates and a network of associations. The dataset used here is described in “Data” section’s subsection “Linearly inseparable data” where the coordinate space of the datapoints and the network are displayed. The key aspect which this dataset emphasizes is that the features alone without the network information cannot produce a linear separation, but with the incorporation of the network information (with linear operators) this classification then becomes possible. Figure 4 shows the results of applying the SGC to the dataset without the network information being used  $k = 0$ , and is effectively an application of logistic regression. The methodology cannot separate the data correctly with a pair of linear projections but that can be alleviated as seen in the next figures by incorporating the network information as well ( $k > 0$ ).

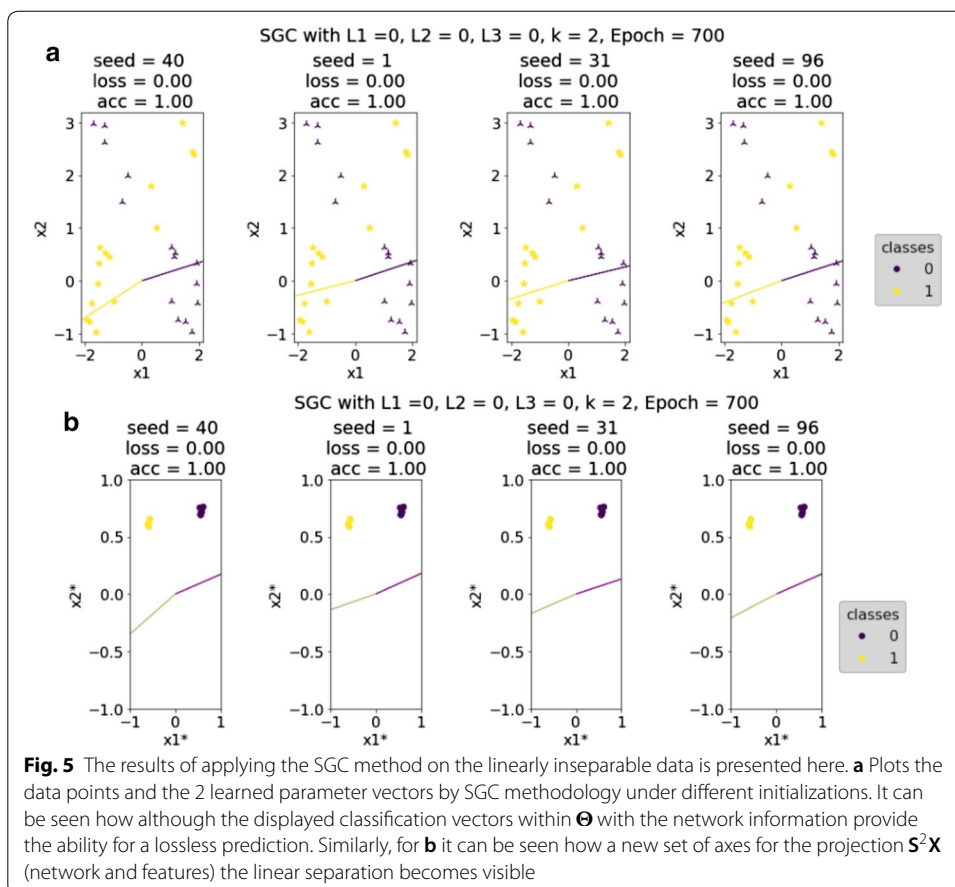
Using the SGC (by setting the shrinkage parameters to 0), in Fig. 5, Subfigure (a) shows that although the vectors for the class projections, as columns in  $\Theta$ , do not enable a separation between the groups the network information enables a perfect accuracy to be produced. This is because although the support for an erroneous class can be accumulated for a point, the feature space ‘communicated’ to it from the edge connections of features overrides the nodes’ own features in these cases.



Each plot is an independent run with slight changes in  $\Theta$ . Subfigure (b) shows a set of plots but where the axes  $x1^*$  and  $x2^*$  for each data point represents the projection of the features with the ‘neighborhoods’ of the points. With  $k = 2$ ,  $S^2$ , aggregates the weights from ‘2 hops’ distance in the network, so that the multiplication of  $S^2X$  is shown on these new axes. It can then be understood why the data is then ‘linearly’ separable after this transformation. This emphasizes how the network information can be used to improve the accuracy and maintain model simplicity.

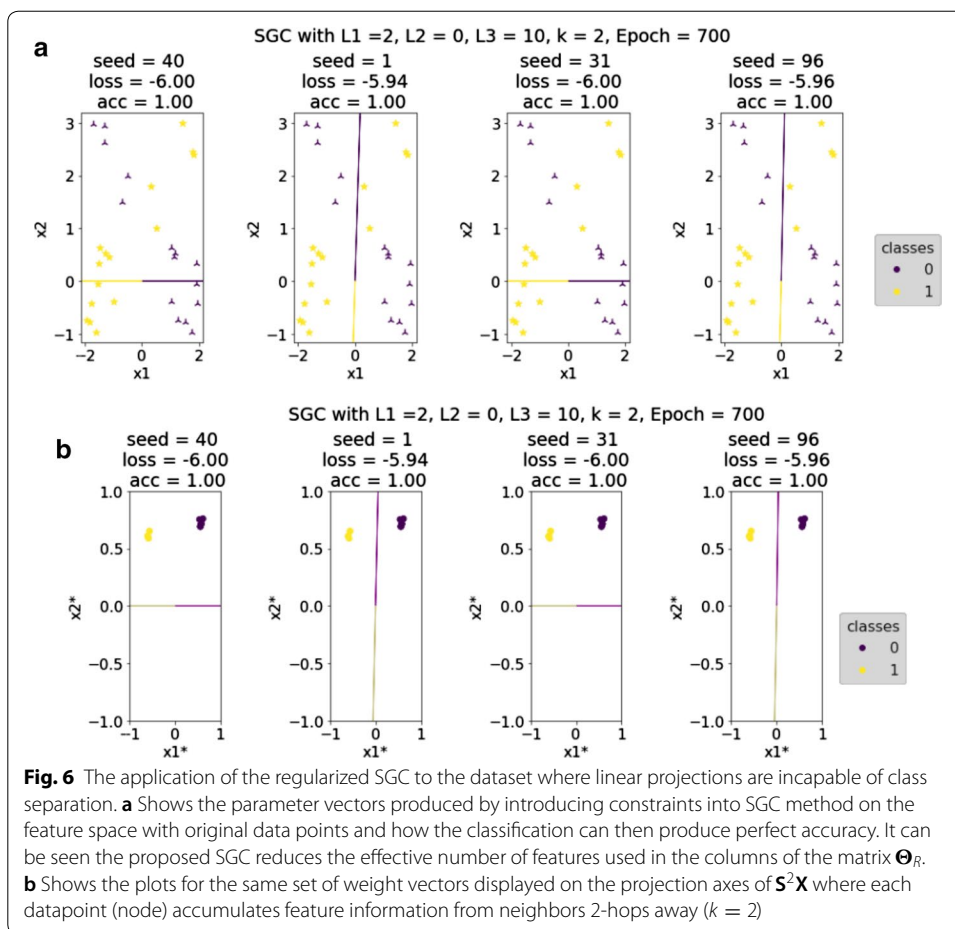
In Fig. 6 the regularized SGC is applied to the dataset (with  $L2 = 0$ ) and the parameter vectors for each class from  $\Theta_R$  are plotted in both Subfigures (a) and (b). The constraints (shrinkages) are placed on the sum of the elements within  $\Theta_{\cdot,j_R}$  and the direction of the vectors which reduces the total value summation for feature extraction. In Subfigure (a) the projection vectors of  $\Theta_R$  are shown and as with Fig. 5 the results produce a perfect accuracy. What can be seen is that the model explores alternative parameterizations which are not found previously without regularization. The projections all display a drop of a features dimension. Subfigure (b) shows the  $S^2X$  projection and that perfect accuracy can still be achieved. In each of the plots it can be seen how various equivalent (in terms of the accuracy) projections can be searched which reduce effectively the number of features used for each class being predicted. The ability for the non-linearly separable data to be correctly classified without introduction of new parameters or ‘layers’ in the CNN enables explorations to be done more efficiently on large datasets in terms of time and processing capabilities.

The change of the  $L3$  constraint is utilized so that the projection vectors for each class are fit to be orthogonal to each other (shown in Eq. 6. This allows the possibility for a smaller number of classes to be fit, if the class number is not known and differs from the previous application in that the support for each class would be seen as a separate linear function’s projected value. Figure 7 shows the results in Subfigure (a) and Subfigure (b) where the vector fit in the space of the data points is seen and how the data is transformed into different axes using the network data respectively. It can be seen how the constraint for the orthogonality is preserved and the accuracy for the fits is still achieved for this problem.



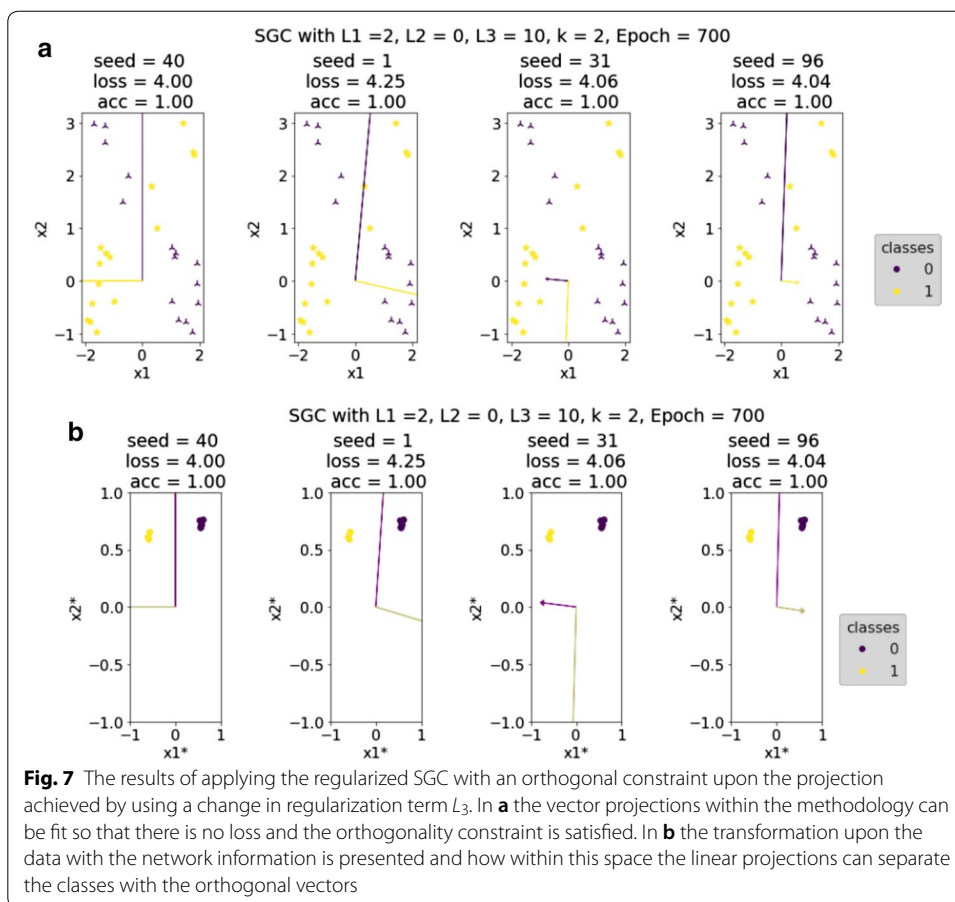
### Application to the Cora dataset

Here is presented the application of the SGC methodology and the proposed regularized SGC to the dataset of Cora [33]. The purpose is to examine the capability of both the SGC and the regularized SGC to a dataset with a large number of features. There are many situations in big data applications where the datasets have large numbers of features due to larger data gathering schemes and a requirement to select key features without supervision. The SGC has been applied to the Cora dataset [44], and here the performance with a regularized version is mainly directed at the interpretability in highlighting the key variables in the feature set while also applying other constraints. Figure 8 present the results with 2 Subfigures with heatmaps displaying the parameter values fitted for each class in  $\Theta$  and  $\Theta_R$ . The dataset classifies each document as belonging to one of 7 different classes where the SGC then produces a parameter matrix  $\Theta$  with 7 columns and  $d$  rows for the feature number. The constraint upon  $L_3$  is set so that the projection vectors between classes are in opposing direction so that class feature loadings are differentiated by their placement in a histogram of the values. With the SGC applied, Subfigure (a) shows the weights of the parameters for each class (a single column in  $\Theta$ ) as a separate heatmap with a legend for the values indicated. Analogously the same set of results but produced with the regularized SGC proposed here is shown in Subfigure (b). The approach produces a new parameter matrix  $\Theta_R$  introduces the regularizations in the



inference scheme for the parameters by penalizing their total sum and directions to be as informative about the features in terms of accuracy prediction and lack of overlap (removing redundancy within large feature spaces typical of large datasets). It can be seen there are fewer variables highlighted for the practitioner to examine, which looks to investigate and highlight which variables are important for the class membership determination. The 7 cells highlighted in the bottom right are padding.

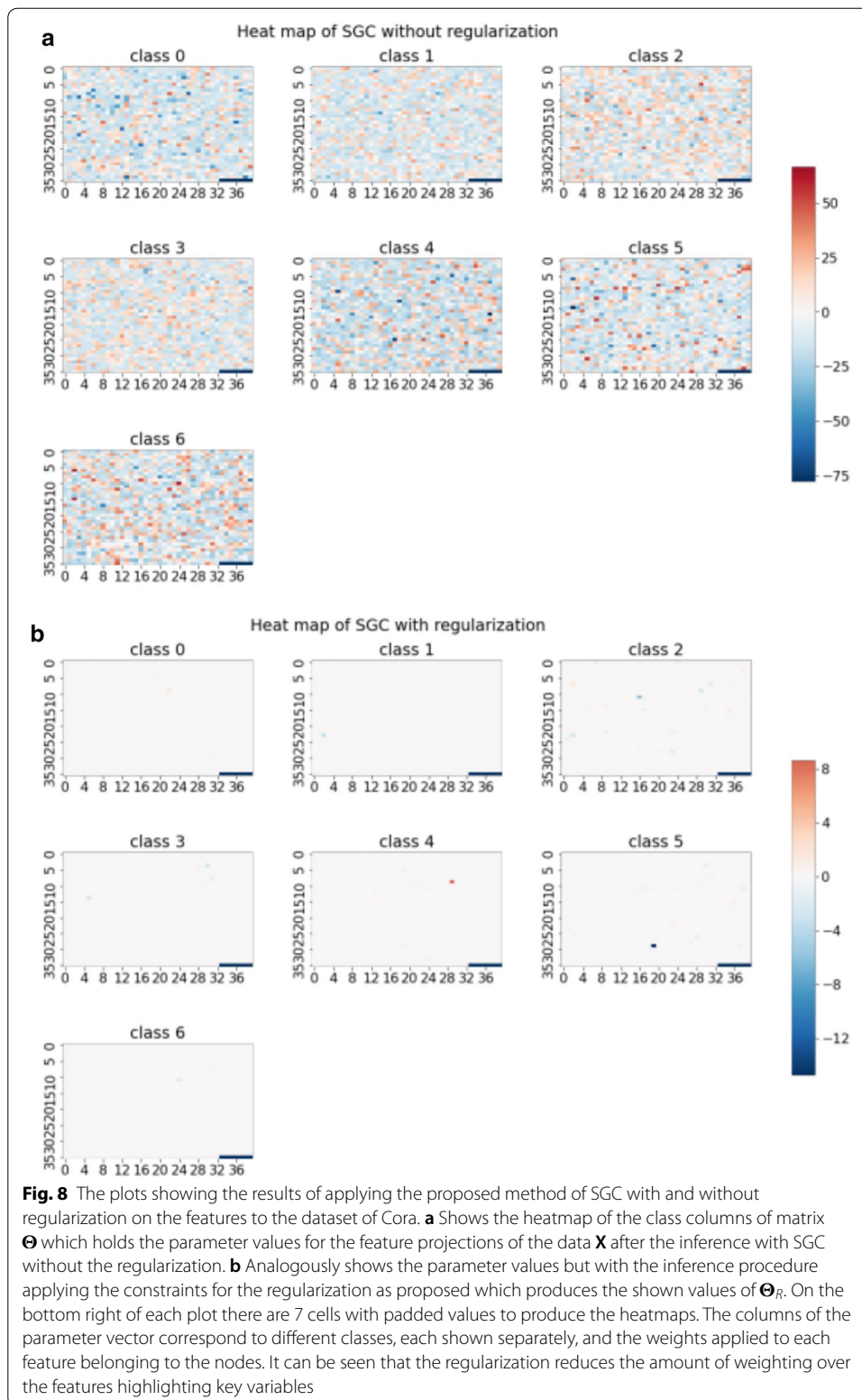
Using the data in the heatmaps shown in Fig. 8 a histogram of the values for the parameter values for each class and the features is created for the SGC and the values from the regularized SGC held in the matrices  $\Theta$  (Subfigure a) and  $\Theta_R$  (Subfigure b). In Fig. 9, Subfigure (a) shows how there is a smaller group of features which provide positive contribution to the class identification and that an apparent 2 mode distribution can be made out. Each plot belongs to a different class in the dataset and are a different column in  $\Theta$ . Subfigure (b) shows the parameter value distribution within  $\Theta_R$ . The effect of the regularization can be seen in comparison with Subfigure (a) where the number of feature values at value 0 are the majority. This makes the exploration and backtracking process the features easier.

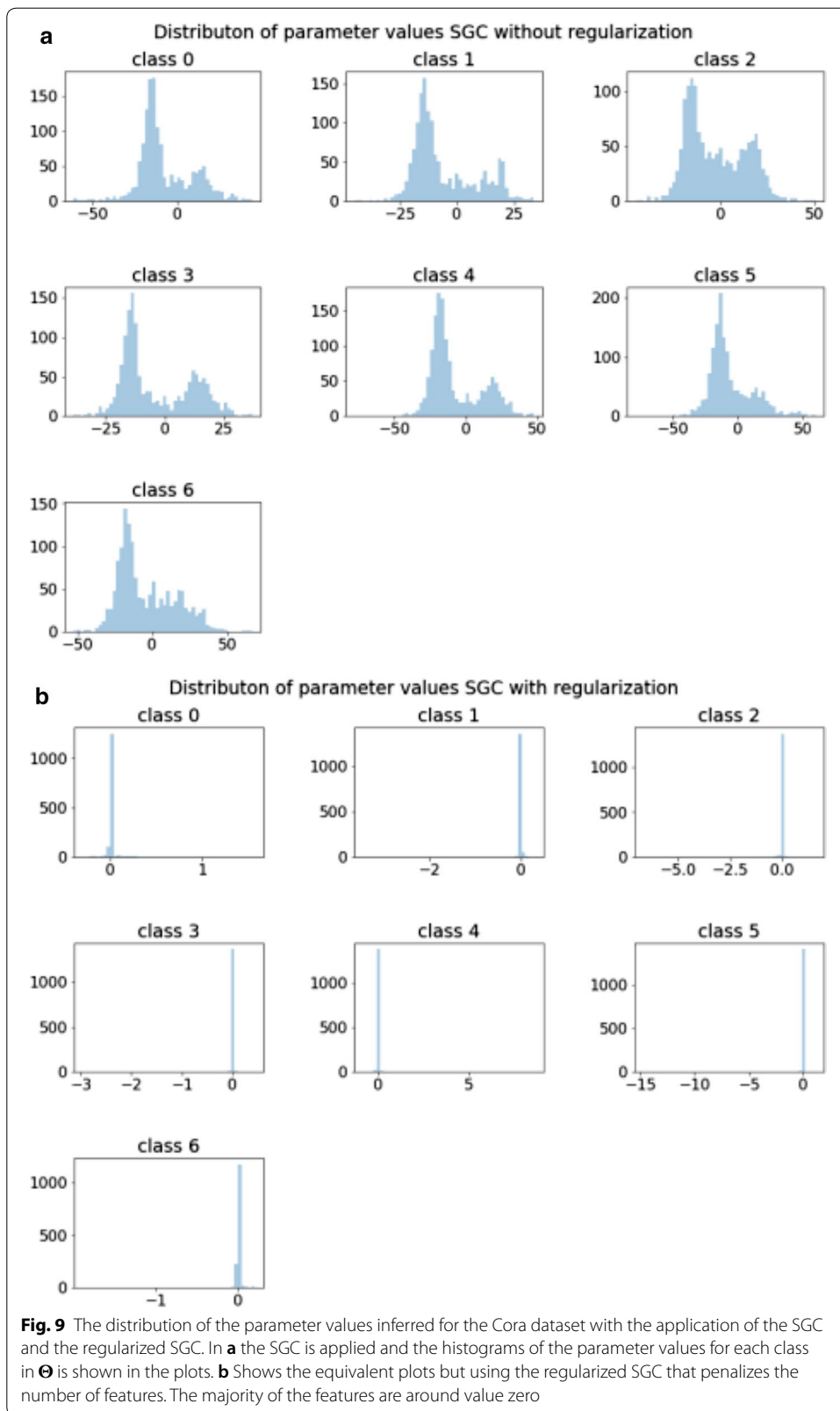


### Discussion

This work proposes a model extension of the Simple Graph Convolution (SGC) which aims at producing a smaller and more meaningful set of projections in which classification labels are presented. It addresses a key issue with interpretability of model applications in big data where many features may be used which are redundant and remove the ability for a practitioner to examine the weights. A key reason for why the SGC was chosen to be extended with this capability is that the operations are linear in the methodology with the exception of the softmax function application can be run relatively efficiently in comparison to methodologies relying on more parameterizations and more ‘layers’ in order to improve accuracy.

The SGC incorporating the network information can produce accurate classification of points in a feature space which is not linearly separable by utilizing the network information via linear operations. The results demonstrated this capability on a small dataset where the network projection effectively linearizes the search by having information from the node ‘neighborhood’ accumulated from ‘k-hops’ distance (relying upon the powers of the adjacency matrix). This allows for fast run times and the application to services which rely upon small delays. The methodology was applied to the Cora citation dataset which has a large number of features and the reduction is







significant in the number of features highlighted to the user. This provides a set small enough to explore manually if required.

## Conclusion and future work

The SGC model extension presented here allows for a more explainable set of results to be presented to the user. The regularization terms reduces the number of non-zero parameters and the overlap between parameterizations of the different classes. Future work could entail a more in depth exploration of how the network can be ‘decomposed’ in such a way as to minimize the number of label alterations. Producing a network separation by eliminating edges can find applications in social networks where polarized communities must be isolated as a means of inoculation.

## Abbreviations

SGC: Simple Graph Convolution; GNN: Graph Neural Network; GCN: Graph Convolutional Network; CNN: Convolutional Neural Networks.

## Acknowledgements

Not applicable.

## Authors’ contributions

Conceptualization, PP and AV; Formal analysis, PP and AV; Investigation, PP and AV; Methodology, PP and AV; Software, PP; Supervision, AV; Validation, PP; Visualization, PP; Writing original draft, PP and AV; Writing—review and editing, PP and AV. Both authors read and approved the final manuscript.

## Funding

This work was partially supported by grant FA8650-18-C-7823 from the Defense Advanced Research Projects Agency (DARPA). The views and opinions contained in this article are the authors and should not be construed as official or as reflecting the views of the University of Central Florida, DARPA, or the U.S. Department of Defense.

## Availability of data and materials

The dataset, Cora, is presented in [33] and is available at <https://docs.dgl.ai/en/0.4.x/api/python/data.html>.

## Competing interests

The authors declare that they have no competing interests.

Received: 2 July 2020 Accepted: 8 October 2020

Published online: 20 October 2020

## References

- Krallman A, Pelletier MJ, Adams FG. @ size vs.# impact: Social media engagement differences amongst facebook, twitter, and instagram. In: *Celebrating America's Pastimes: Baseball, Hot Dogs, Apple Pie and Marketing?* Berlin: Springer; 2016. p. 557–61.
- Newman M. *Networks*. Oxford: Oxford University Press; 2018.
- Barabási A-L, et al. *Network science*. Cambridge: Cambridge University Press; 2016.
- Girvan M, Newman ME. Community structure in social and biological networks. *Proc Natl Acad Sci*. 2002;99(12):7821–6.
- Borgatti SP. Centrality and network flow. *Soc Netw*. 2005;27(1):55–71.
- Jeong H, Nédá Z, Barabási A-L. Measuring preferential attachment in evolving networks. *EPL (Europhys Lett)*. 2003;61(4):567.
- Albert R, Barabási A-L. Statistical mechanics of complex networks. *Rev Mod Phys*. 2002;74(1):47.
- Lafin P, Mantzaris AV, Ainley F, Otley A, Grindrod P, Higham DJ. Discovering and validating influence in a dynamic online social network. *Soc Netw Anal Min*. 2013;3(4):1311–23.
- Soares FB, Recuero R, Zago G. Influencers in polarized political networks on twitter. In: *Proceedings of the 9th international conference on social media and society*. 2018. p. 168–77.
- Ricci F, Rokach L, Shapira B. Introduction to recommender systems handbook. In: *Recommender systems handbook*. Berlin: Springer; 2011. p. 1–35.
- Su X, Khoshgoftaar TM. A survey of collaborative filtering techniques. *Advances in artificial intelligence*. 2009;2009.
- Unga, LH, Foster DP. Clustering methods for collaborative filtering. In: *AAAI workshop on recommendation systems*, Menlo Park, CA, vol. 1. 1998. p. 114–29.
- Zhang R, Tran T. An information gain-based approach for recommending useful product reviews. *Knowl Inf Syst*. 2011;26(3):419–34.

14. Kabakchieva D. Student performance prediction by using data mining classification algorithms. *Int J Comput Sci Manag Res*. 2012;1(4):686–90.
15. Thai-Nghe N, Drumond L, Krohn-Grimberghe A, Schmidt-Thieme L. Recommender system for predicting student performance. *Procedia Comput Sci*. 2010;1(2):2811–9.
16. Jackson MO. Networks in the understanding of economic behaviors. *J Econ Perspect*. 2014;28(4):3–22.
17. Shu K, Wang S, Tang J, Zafarani R, Liu H. User identity linkage across online social networks: a review. *Acm Sigkdd Explor Newsl*. 2017;18(2):5–17.
18. Althoff T, Jindal P, Leskovec J. Online actions with offline impact: how online social networks influence online and offline user behavior. In: *Proceedings of the tenth ACM international conference on web search and data mining*. 2017. p. 537–46.
19. Crucitti P, Latora V, Porta S. Centrality measures in spatial networks of urban streets. *Phys Rev E*. 2006;73(3):036125.
20. Euler L. *Solutio problematis ad geometriam situs pertinentis*. *Commentarii academiae scientiarum Petropolitanae*. 1741: 128–40.
21. Kivelä M, Arenas A, Barthelemy M, Gleeson JP, Moreno Y, Porter MA. Multilayer networks. *J Complex Netw*. 2014;2(3):203–71.
22. Belyi A, Bojic I, Sobolevsky S, Sitko I, Hawelka B, Rudikova L, Kurbatski A, Ratti C. Global multi-layer network of human mobility. *Int J Geogr Inf Sci*. 2017;31(7):1381–402.
23. McPherson M, Smith-Lovin L, Cook JM. Birds of a feather: homophily in social networks. *Ann Rev Sociology*. 2001;27(1):415–44.
24. Son J, Kim SB. Academic paper recommender system using multilevel simultaneous citation networks. *Decis Support Syst*. 2018;105:24–33.
25. Radicchi F, Fortunato S, Vespignani A. Citation networks. In: *Models of science dynamics*. Berlin: Springer; 2012. p. 233–57.
26. Suthaharan S. Big data classification: problems and challenges in network intrusion prediction with machine learning. *ACM SIGMETRICS Perform Eval Rev*. 2014;41(4):70–3.
27. Caldarola EG, Rinaldi AM. Big data visualization tools: a survey. *Research Gate* 2017.
28. Reddy GT, Reddy MPK, Lakshmana K, Kaluri R, Rajput DS, Srivastava G, Baker T. Analysis of dimensionality reduction techniques on big data. *IEEE Access*. 2020;8:54776–88.
29. Scarselli F, Gori M, Tsoi AC, Hagenbuchner M, Monfardini G. The graph neural network model. *IEEE Trans Neural Netw*. 2008;20(1):61–80.
30. Zhou J, Cui G, Zhang Z, Yang C, Liu Z, Wang L, Li C, Sun M. Graph neural networks: a review of methods and applications. *arXiv preprint arXiv:1812.08434* 2018.
31. Wu F, Zhang T, Souza AHd, Fifty C, Yu T, Weinberger KQ. Simplifying graph convolutional networks. In: *36th international conference on machine learning, ICML 2019, 2019-June, 2019*. p. 11884–94.
32. Tibshirani R. Regression shrinkage and selection via the lasso. *J R Stat Soc Ser B (Methodol)*. 1996;58(1):267–88.
33. Mccallum A. Cora research paper classification dataset. [people.cs.umass.edu/mccallum/data.html](http://people.cs.umass.edu/mccallum/data.html). KDD 2001.
34. LeCun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. *Proc IEEE*. 1998;86(11):2278–324.
35. LeCun Y, Bengio Y, Hinton G. Deep learning. *Nature*. 2015;521(7553):436–44.
36. Defferrard M, Bresson X, Vandergheynst P. Convolutional neural networks on graphs with fast localized spectral filtering. In: *Advances in neural information processing systems*. 2016. p. 3844–52.
37. Shuman DI, Narang SK, Frossard P, Ortega A, Vandergheynst P. The emerging field of signal processing on graphs: extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Process Mag*. 2013;30(3):83–98.
38. Sandryhaila A, Moura JM. Discrete signal processing on graphs. *IEEE Trans Signal Process*. 2013;61(7):1644–56.
39. NT H, Maehara T. Revisiting graph neural networks: All we have is low-pass filters. *arXiv preprint arXiv:1905.09550* 2019.
40. Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks. In: *5th international conference on learning representations, ICLR 2017—conference track proceedings 2016*.
41. Shchur O, Mumme M, Bojchevski A, Günnemann S. Pitfalls of graph neural network evaluation. *arXiv preprint arXiv:1811.05868* 2018.
42. Newman ME. Modularity and community structure in networks. *Proc Natl Acad Sci*. 2006;103(23):8577–82.
43. Ketkar N. Introduction to pytorch. In: *Deep learning with Python*. Berlin: Springer; 2017. p. 195–208.
44. Bidoki NH, Mantzaris AV, Sukthankar G. Exploiting weak ties in incomplete network datasets using simplified graph convolutional neural networks. *Mach Learn Knowl Extract*. 2020;2(2):125–46.

## Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.