

# Exploring the Regularized SGC in application to social network data

Patrick Pho<sup>1</sup>, Alexander V. Mantzaris<sup>1</sup>

<sup>1</sup> University of Central Florida, Department of Statistics and Data Science  
211C TCII, 4000 University Blvd, Orlando FL 32816-2370  
patrick.pho@ucf.edu, alexander.mantzaris@ucf.edu

## Abstract

Attributed graphs are a powerful tool to model real life systems which exist in many domains such as social science, biology, etc. Online social networks have a significant effect on human society and have become an important research topic for maintaining the integrity of the common social understanding. This includes combating disinformation and sensing predispositions that can disrupt peaceful discourse. Studying networks often involves labeling nodes into representative groups. Simple machine learning approaches or community detection have limitation in their capability to make use of both network topology and node features. Graph Neural Networks (GNNs) provide an efficient framework combining both sources of information to produce accurate node classification. In this work, we study the application of two variants of GNNs, namely Simple Graph Convolution (SGC) and its extension on a social network dataset as well as a comprehensive set of synthetic attributed graphs with varying network topology. The SGC provides fast and efficient framework while its extension improves interpretability of the result by highlighting key node features determining the class characteristics.

## Introduction

Attributed graph (network) provides powerful representation of real life complex systems where each element is regarded as node with associated attributes (features) and the connectivity information between elements forms edges. This graph-structured feature data is used to represent complex systems in various domains such as social science (social networks (Fan et al. 2019)), biology (biochemical pathways (Bove et al. 2020)), material science (molecular networks (Gilmer et al. 2017)). Behaviors of such systems are mostly defined by or dependent on their underlying network structure (Estrada 2012).

Online social networks have been at the forefront of the interest in networks research due to their rapid integration into every aspect human life and the profound impact they have on human behaviors (Althoff, Jindal, and Leskovec 2017; Camacho et al. 2020). The work of (Umberson and Karas Montez 2010) points out the important role of social relationships in defining physical and mental health status.

The fast growth of internet-based social media facilitates intercommunication between consumers and opens up a new opportunity for companies to interact with customers (Mangold and Faulds 2009). Beside many great benefits, social networks also become a medium to spread fake news leaving negative impact on individuals and society (Shu et al. 2017). Investigation into network structure often involve simplifying the collection of nodes into a handful set of communities or groups. The simplification process can be regarded as assigning nodes with distinct labels of their communities capturing aggregated behaviors of all existing nodes in the network. The principle underlying the ability to group nodes together in this fashion relies on homophily (McPherson, Smith-Lovin, and Cook 2001). Examples of this are found in the work of (Kahne and Bowyer 2018) which studies how social network connections created from friendships or interests can drive political engagements differently.

Labels of nodes can be inferred using standard classification methods such as logistic regression, which are predominantly reliant on the node feature information. However, these methods do not consider the supplementary node connectivity information. On the other hand, community detection algorithms (e.g. Louvain (Blondel et al. 2008)) take into account only node connectivity information and hence ignore rich information of node features.

Graph Neural Networks (GNNs) is a class of neural networks operating on the graph domain and is able to combine node attributes and connectivity to produce informative node representation (node embedding). GNNs produces expressive node representations (node embeddings) via an iterative message passing mechanism where each node aggregates its direct neighbors' feature vectors and updates its own feature vector with the aggregate information. The final node representation encapsulates structural information of a k-hop neighborhood. Many GNNs variants have been proposed and achieved state-of-the-art performance on a variety of tasks. Graph Convolutional Networks (GCN) (Kipf and Welling 2016) learn a graph representation via layer-wise propagation rules representing localized spectral filters. Graph Attention Network (GAT) (Veličković et al. 2017) utilizes an attention mechanism to account for neighbors' importance in aggregation phase. With the growth of the datasets and the databases that hold them the concerns on the processing time, complexity of the models and the ability to

interpret the results become a key concern.

The Simple Graph Convolution (SGC) (Wu et al. 2019) simplifies GCN by removing non-linear transitions between layers while still be able to retain its representational power. The work of (Pho and Mantzaris 2020) demonstrates expressive node embedding capability of SGC and explores flexible regularization mechanism to facilitate meaningful interpretation.

In this work, we conduct simulation study to compare the performance of SGC and its regularized extension. We then explore the application of SGC and regularized SGC on social network attributed graph data. Since SGC framework involves mostly linear operation, it offers high efficiency and scalability compared with other frameworks relying on more 'layers' in order to improve accuracy. It is also of interest to investigate the effect of regularization on dimensionality reduction to highlight important node features and improve our understanding of how these input features affect the assessment of target class. Our framework can be used to build recommendation system for large social network where computational cost is high and thus fast and efficient framework is preferable. Our model can highlight useful node features that are important to determine the categories of products or communities of users.

## Data

### Synthetic datasets

We examine the performance of SGC and its regularized version under a variety of network topologies. Thirty attributed graph datasets are synthesized to imitate scale-free networks which are commonly found in real applications. Each graph contains three clusters (subgraphs) with 100 nodes per cluster. Each subgraph is constructed following the Barabási-Albert preferential attachment model (Barabási and Albert 1999). The interconnectivity between a pair of subgraphs is determined as follows.

- On each subgraph, a subset of nodes is chosen using weighted random sampling on degrees of the nodes.
- Random edges are generated between a pair of subsets of nodes. The probability of connecting a pair of nodes is **inter\_p**.

Hyperparameters (number of preferential attachment for the Barabási-Albert model, probability of random edges) are then established to control the topology of the whole graph. Samples of simulated graphs are illustrated in Figures 1.

The node feature matrix corresponding with each cluster is generated following multivariate normal distributions  $\mathbf{X}_c \sim \mathcal{N}(\mu_c, \Sigma)$ ,  $c = 1, 2, 3$ . We set the mean vectors as  $\mu_1 = (1, 0, 0)^T$ ,  $\mu_2 = (0, 1, 0)^T$ ,  $\mu_3 = (0, 0, 1)^T$  and the covariance matrix is  $\Sigma = \mathbf{I}_3$ .

### Real dataset

We further investigate the application of regularized SGC on Facebook Page-Page dataset (Rozemberczki, Allen, and Sarkar 2021; Fey and Lenssen 2019). This attributed graph contains Facebook pages as nodes and mutual likes between pages are regarded as edges. Node features are extracted

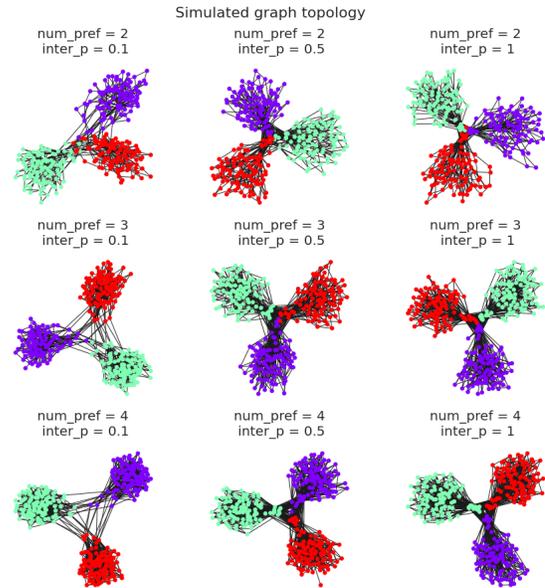


Figure 1: Visualizations various network topology from simulated attributed graphs, each with 3 communities (colored). Number of preferential attachment (`num_pref`) controls the overall amount of connectivity while the inter-graph connectivity probability (`inter_p`) governs the amount of linkings between subgraphs.

from the site descriptions summarizing the purpose of the site. Each page can be classified as one of the four categories: politicians, governmental organizations, television shows and companies.

Statistics	Value
Number of nodes	22,470
Number of edges	171,002
Number of isolated nodes	0
Number of classes	4
Node feature	128
Density	0.001

Table 1: Summary statistics of Facebook Page-Page dataset

## Methodology

SGC is introduced in (Wu et al. 2019) as a simplified GNN model developed from GCN (Kipf and Welling 2016) by removing non-linear activation functions between hidden layers and reparametrizing successive layers into one single layer. This modification reduces superfluous complexity of GCN while retains superb performance on many downstream tasks.

In this section, we briefly present the original SGC. An attributed graph data set contains a graph  $G = (V; E)$  of  $N$  nodes (vertices)  $V = (v_1, v_2, \dots, v_N)$  and edges  $(v_i, v_j) \in E$ . Often, the connectivity in  $E$  is represented by an adjacency matrix  $\mathbf{A} \in R^{N \times N}$  where each element  $a_{ij}$  repre-

sents an edge between node  $v_i$  and  $v_j$  ( $a_{ij} = 0$  if  $v_i$  and  $v_j$  are disconnected). Each node might have a feature vector  $x_{v_i} \in R^D$  which is stacked together to form a node attribute matrix  $\mathbf{X} \in R^{N \times D}$ . 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  $d_i = \sum_j a_{ij}$ . Each node  $i$  receives a 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  $\mathbf{S}$ :

$$\mathbf{S} = \tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \quad (1)$$

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  $k^{\text{th}}$ -layer of the GCN resulting in a linear model of the form:

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

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

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

The parameter  $k$  corresponds to the number of 'hops' which is the number of edge traversals in the network adjacency matrix  $\mathbf{S}$ . This  $k$  parameter can be thought of as accumulating information from a certain number of hops away from a node (as described visually in (Wu et al. 2019)). 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:

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

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

## Regularized SGC

The work of (Pho and Mantzaris 2020) illustrates SGC's expressive power in node classification task and proposes a flexible regularization methodology to reduce the number of parameters and highlight a sparse set of important features. They introduce a flexible set of constraints in terms of shrinkage parameters  $L_1$ ,  $L_2$ , and  $L_3$  in the loss  $\mathcal{L}$  from Eq 4:

$$\begin{aligned} \mathcal{L}_R = \mathcal{L} + L_1 \times \sum_{c \in C} \left( \sum_{d=1}^D |\Theta_{R(\cdot, c)}|^4 \right)^{(-1)} + \\ L_2 \times \sum_{c \in C} \|\Theta_{R(\cdot, c)}\|_2 + \\ L_3 \times \left( \sum_{c_1 \in C} \sum_{c_2 \in C} \left( |\Theta_{R(\cdot, c_1)}^T| \cdot |\Theta_{R(\cdot, c_2)}| : c_1 \prec c_2 \right)^2 \right) \end{aligned} \quad (5)$$

where  $\Theta_R$  is the parameters for the regularized fitted SGC and  $|\Theta_{R(\cdot, c)}|^4$  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.  $L_1$  regulates the number of parameters by inducing the penalization with a larger skew in the number of elements in the columns of  $\Theta_R$ . The  $L_2$  term controls the total magnitude of the parameter vector while the  $L_3$  term penalizes class label projection which have large overlaps.

In this work, we modify  $L_3$  term to impose an orthogonality constraint between the projection vectors:

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

## Experiment

Two GNN frameworks, namely SGC (Wu et al. 2019) and regularized SGC (Pho and Mantzaris 2020) are adopted for our experiment with multi-class node classification task. For both models, we set the number of hops  $k = 2$ . All models are trained using the AdamW optimizer (Loshchilov and Hutter 2017) with a learning rate of 0.2.

### Synthetic datasets

For simulation study, we set the proportion of training, validation, and test sets are 20%, 25%, 45% respectively. Both models are trained for 100 epochs. Optimal values of  $L_1$  and  $L_3$  are chosen from a grid search of  $20 \times 20$  values, each varies from  $10^{-5}$  to  $10^2$ , to maximize the validation accuracy. Note that we utilize the orthogonality constraint for the penalty  $L_3$ .

### Real dataset

For Facebook Page-Page dataset, we pre-process by normalizing feature vector of each node to sum-up to one. The training, validating, and test proportions are 5%, 25%, and 50% respectively.

The number of training epochs are set as 200. Optimal regularization parameters of SGC  $L_1$  and  $L_3$  are chosen from a grid search of  $50 \times 50$  values, each varies from  $10^{-5}$  to  $10^2$ , to maximize validation accuracy. The  $L_3$  term is modified to imposes orthogonality constraint on the weight vectors.

We implement the regularized SGC and SGC using PyTorch (Paszke et al. 2019) and Deep Graph Library (Wang et al. 2019) and release the source code in our Github repository<sup>1</sup>. The experiments are conducted using Kaggle kernels<sup>2</sup>.

<sup>1</sup><https://github.com/phuongpho/reg-sgc>

<sup>2</sup><https://www.kaggle.com/code>

## Results

### Synthetic datasets

Figure 2 displays line graphs with test accuracy on the vertical axis and probability of intergraph connectivity on the horizontal axis. Overall, both SGC and regularized SGC achieve decent classification capability under different network structures. Our proposed framework not only performs on par with the SCG but also induces sparsity on the fitted parameters, which facilitates further investigation on important features defining class membership.

We also observe that as the network gets denser (indicated by increasing  $\text{num\_pref}$ ), the models tend to perform better. This can be explained as on sparse graphs, nodes tend to have few neighbors, and the node representations produced by aggregating neighborhood features would become less smooth. Consequently, the models would have difficulty classifying nodes in these ineffective mapping spaces.

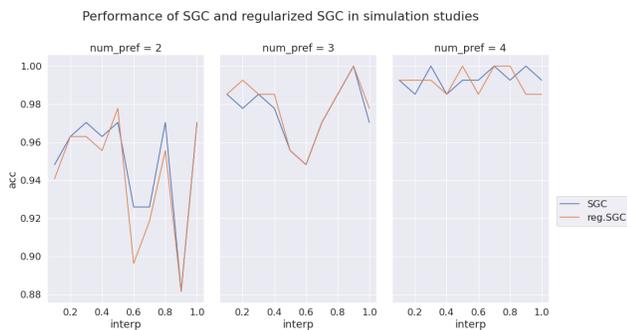


Figure 2: Performance of the SGC and regularized SGC under various network settings.

### Real dataset

In this section, we discuss the application of the proposed methodology to the Facebook Page-Page dataset. The purpose is to examine the capability of both the SGC and the regularized SGC on real life social network data with large number of features.

Figure 3 displays the weight vectors fitted on Facebook Page-Page dataset using SGC (Subfigure (a)) and regularized SGC (Subfigure (b)). Each plot contains four heat maps showing the magnitude of weight vectors corresponding to four classes. In Subfigure (a), the fitted weight vectors  $W$  ranges from 20 to  $-30$ . The model utilizes all input features to produce discriminative scores for node classification and achieves a good accuracy of 84.5%. The majority of weight values are non-zero indicating that under SGC model, most features have considerable impact on the class likelihood. Analogously the same set of results but produced with the regularized SGC is shown in Subfigure (b). The components of the weight vectors  $W_R$  are significantly reduced and the model perform equivalently with an accuracy of 84.2%. It's clear that penalty term  $L_1$  and  $L_3$  produce a strong shrinkage effect while maintain predictive power of the model.

In figure 4, we further investigate the distribution of the weight vectors of SGC with and without regularization. The

fitted weight vectors of SGC shown in Subfigure (a) seem to follow bimodal distributions with the two mode values locate around  $-20$  and  $20$ . The weight distribution suggests that almost half of the input features have positive impact on class likelihood while the remaining half tends to decrease class likelihood. Subfigure (b) shows the results of regularization incorporated in SGC framework. The penalization remarkably shifts the distribution of weight values into unimodal indicating that the majority of the components are shrunk toward zero.

Figure 5 compares model performances between SGC and its regularized extension. The accuracy, macro precision, and macro recall of SGC are all above 80% suggesting a great predictive capability. It can be seen that the incorporation of regularization renders a similar performance while further enhances the interpretability of the model. The sparsity of the fitted weight  $W_R$  facilitates the investigation of important input features defining the characteristics of each class.

## Discussion

In this paper, we examine the capability of SGC and its regularized extension on social network dataset. The SGC framework combines the network information and node features via linear operations to produce accurate classification. This allows for fast run times and the application to services which rely upon small delays. The incorporation of regularization significantly reduces the number of input features highlighted to the user while still maintain similar predictive capability. This result in selective sets of important features for practitioner to explore manually if interested. Simulation study shows that our proposed model performs on par with the SGC on thirty attributed graph datasets with varying network structures.

## References

- Althoff, T.; Jindal, P.; and Leskovec, J. 2017. 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*, 537–546.
- Barabási, A.-L.; and Albert, R. 1999. Emergence of scaling in random networks. *science*, 286(5439): 509–512.
- Blondel, V. D.; Guillaume, J.-L.; Lambiotte, R.; and Lefebvre, E. 2008. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10): P10008.
- Bove, P.; Micheli, A.; Milazzo, P.; and Podda, M. 2020. Prediction of Dynamical Properties of Biochemical Pathways with Graph Neural Networks. In *BIOINFORMATICS*, 32–43.
- Camacho, D.; Panizo-LLedot, A.; Bello-Organ, G.; Gonzalez-Pardo, A.; and Cambria, E. 2020. The four dimensions of social network analysis: An overview of research methods, applications, and software tools. *Information Fusion*, 63: 88–120.
- Estrada, E. 2012. *The structure of complex networks: theory and applications*. Oxford University Press.

Fan, W.; Ma, Y.; Li, Q.; He, Y.; Zhao, E.; Tang, J.; and Yin, D. 2019. Graph neural networks for social recommendation. In *The World Wide Web Conference*, 417–426.

Fey, M.; and Lenssen, J. E. 2019. Fast Graph Representation Learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*.

Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; and Dahl, G. E. 2017. Neural message passing for quantum chemistry. In *International Conference on Machine Learning*, 1263–1272. PMLR.

Kahne, J.; and Bowyer, B. 2018. The political significance of social media activity and social networks. *Political Communication*, 35(3): 470–493.

Kipf, T. N.; and Welling, M. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Loshchilov, I.; and Hutter, F. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Mangold, W. G.; and Faulds, D. J. 2009. Social media: The new hybrid element of the promotion mix. *Business horizons*, 52(4): 357–365.

McPherson, M.; Smith-Lovin, L.; and Cook, J. M. 2001. Birds of a feather: Homophily in social networks. *Annual review of sociology*, 27(1): 415–444.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Kopf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; and Chintala, S. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Wallach, H.; Larochelle, H.; Beygelzimer, A.; d’Alché-Buc, F.; Fox, E.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 32*, 8024–8035. Curran Associates, Inc.

Pho, P.; and Mantzaris, A. V. 2020. Regularized Simple Graph Convolution (SGC) for improved interpretability of large datasets. *Journal of Big Data*, 7(1): 1–17.

Rozemberczki, B.; Allen, C.; and Sarkar, R. 2021. Multi-scale attributed node embedding. *Journal of Complex Networks*, 9(2): cnab014.

Shu, K.; Sliva, A.; Wang, S.; Tang, J.; and Liu, H. 2017. Fake news detection on social media: A data mining perspective. *ACM SIGKDD explorations newsletter*, 19(1): 22–36.

Umberson, D.; and Karas Montez, J. 2010. Social relationships and health: A flashpoint for health policy. *Journal of health and social behavior*, 51(1\_suppl): S54–S66.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; and Bengio, Y. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Wang, M.; Zheng, D.; Ye, Z.; Gan, Q.; Li, M.; Song, X.; Zhou, J.; Ma, C.; Yu, L.; Gai, Y.; Xiao, T.; He, T.; Karypis, G.; Li, J.; and Zhang, Z. 2019. Deep Graph Library: A Graph-Centric, Highly-Performant Package for Graph Neural Networks. *arXiv preprint arXiv:1909.01315*.

Wu, F.; Souza, A.; Zhang, T.; Fifty, C.; Yu, T.; and Weinberger, K. 2019. Simplifying graph convolutional networks. In *International conference on machine learning*, 6861–6871. PMLR.

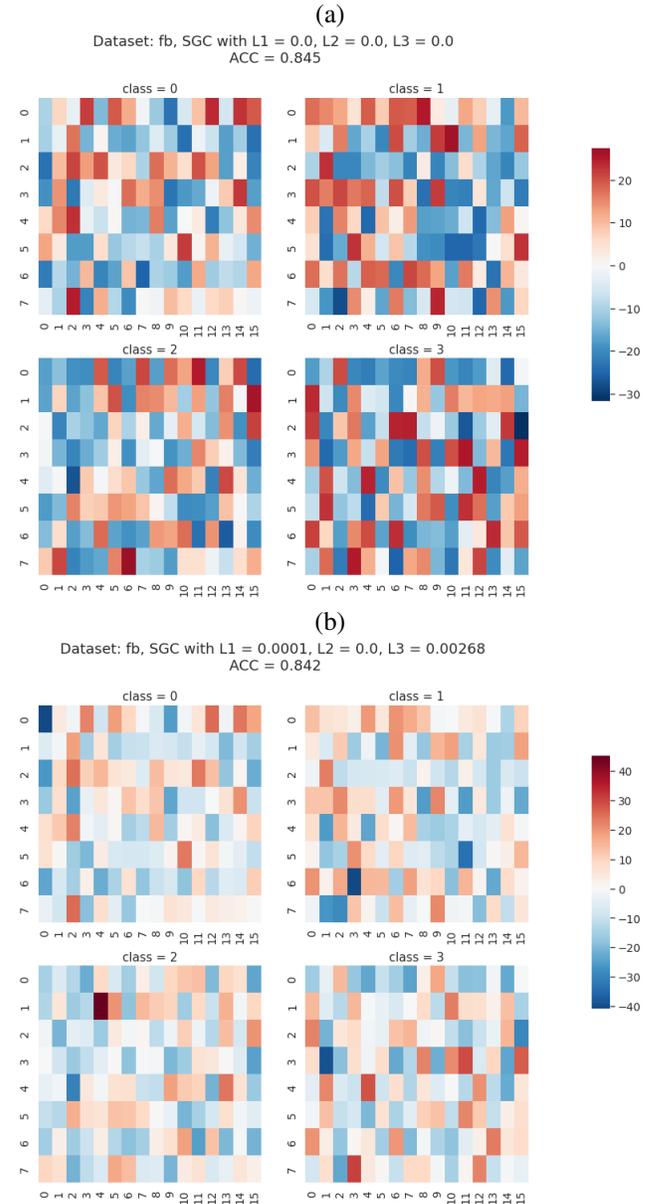


Figure 3: Heat maps of weight vectors fitted on Facebook Page-Page dataset. Each heat map displays the magnitude of weight vector corresponding to each class. Subfigure (a) presents the weight  $\mathbf{W}$  of SGC model without regularization while Subfigure (b) shows the weight  $\mathbf{W}_R$  of SGC model with regularization. It can be seen that applying regularization significantly reduces the amount of weight highlighting key features.

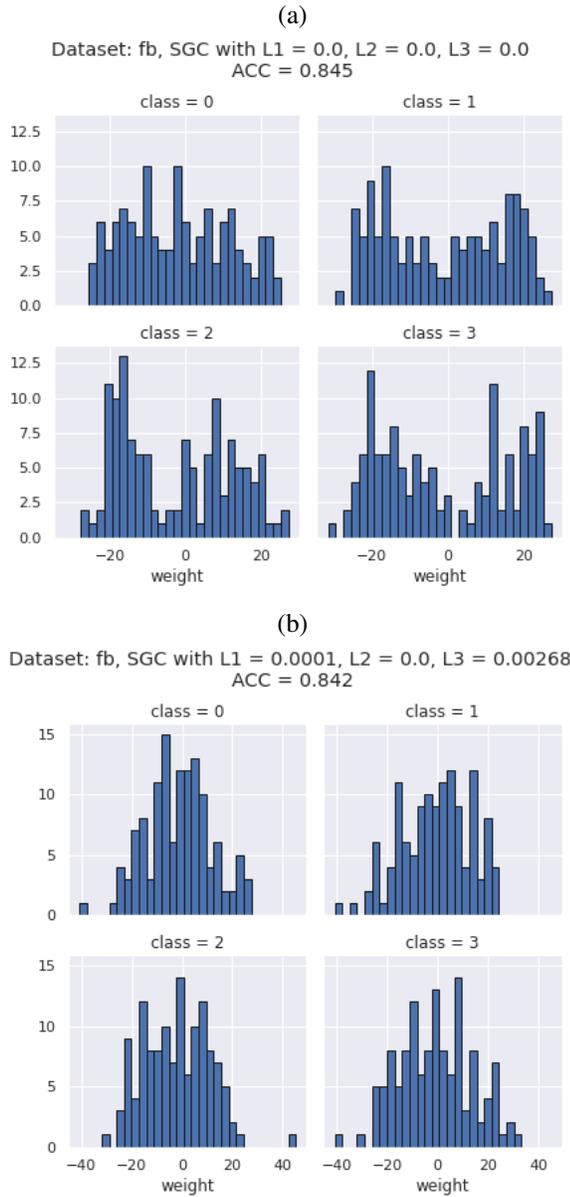


Figure 4: Distribution of the weight values inferred on Facebook Page-Page dataset. In Subfigure (a) the SGC is applied and the histograms of the parameter values for each class in  $\mathbf{W}$  is shown in the plots. Subfigure (b) shows the equivalent plots but using the regularized SGC that penalizes the number of features. The majority of the features are around value zero.



Figure 5: Accuracy, macro precision, and macro recall of SGC and regularized SGC evaluated on Facebook Page-Page dataset. The application of regularization improves interpretability significantly with almost no reduction on predictive capability.