软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software [doi: 10.13328/j.cnki.jos.007065]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

面向鲁棒图结构防御的过参数化图神经网络*

初 旭[1], 马辛宇[2,3], 林 阳[2,3], 王 鑫[1,4], 王亚沙[3,5], 朱文武[1,4], 梅 宏[3]

[1](清华大学 计算机科学与技术系, 北京 100084)

[2](北京大学 计算机学院, 北京 100871)

[3](高可信软件技术教育部重点实验室 (北京大学), 北京 100871)

[4](清华大学 北京信息科学与技术国家研究中心, 北京 102206)

[5](北京大学 软件工程国家工程中心, 北京 100871)

通信作者: 初旭, E-mail: chu_xu@tsinghua.edu.cn

摘 要: 图数据在现实应用中普遍存在, 图神经网络 (GNN) 被广泛应用于分析图数据, 然而 GNN 的性能会被图结构上的对抗攻击剧烈影响. 应对图结构上的对抗攻击, 现有的防御方法一般基于图内聚先验进行低秩图结构重构. 但是现有的图结构对抗防御方法无法自适应秩真值进行低秩图结构重构, 同时低秩图结构与下游任务语义存在错配. 为了解决以上问题, 基于过参数化的隐式正则效应提出过参数化图神经网络 (OPGNN) 方法, 并形式化证明所提方法可以自适应求解低秩图结构, 同时证明节点深层表征上的过参数化残差链接可以有效解决语义错配. 在真实数据集上的实验结果表明, OPGNN 方法相对于现有基线方法具有更好的鲁棒性, 同时, OPGNN 方法框架在不同的图神经网络骨干上如 GCN、APPNP 和 GPRGNN 上显著有效.

关键词: 图节点半监督分类; 图结构对抗防御; 过参数化; 隐式正则化; 图神经网络

中图法分类号: TP18

中文引用格式: 初旭, 马辛宇, 林阳, 王鑫, 王亚沙, 朱文武, 梅宏. 面向鲁棒图结构防御的过参数化图神经网络. 软件学报. http://www.jos.org.cn/1000-9825/7065.htm

英文引用格式: Chu X, Ma XY, Lin Y, Wang X, Wang YS, Zhu WW, Mei H. Over-parameterized Graph Neural Network Towards Robust Graph Structure Defending. Ruan Jian Xue Bao/Journal of Software (in Chinese). http://www.jos.org.cn/1000-9825/7065.htm

Over-parameterized Graph Neural Network Towards Robust Graph Structure Defending

CHU Xu[1], MA Xin-Yu[2,3], LIN Yang[2,3], WANG Xin[1,4], WANG Ya-Sha[3,5], ZHU Wen-Wu[1,4], MEI Hong[3]

[1](Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China)

[2](School of Computer Science, Peking University, Beijing 100871, China)

[3](Key Laboratory of High Confidence Software Technologies (Peking University), Ministry of Education, Beijing 100871, China)

[4](Beijing National Research Center for Information Science and Technology, Tsinghua University, Beijing 102206, China)

[5](National Research Center of Software Engineering, Peking University, Beijing 100871, China)

Abstract: Graph data is ubiquitous in real-world applications, and graph neural networks (GNNs) have been widely used in graph data analysis. However, the performance of GNNs can be severely impacted by adversarial attacks on graph structures. Existing defense methods against adversarial attacks generally rely on low-rank graph structure reconstruction based on graph community preservation priors. However, existing graph structure adversarial defense methods cannot adaptively seek the true low-rank value for graph structure reconstruction, and low-rank graph structures are semantically mismatched with downstream tasks. To address these problems, this study proposes the over-parameterized graph neural network (OPGNN) method based on the implicit regularization effect of over-parameterization. In addition, it formally proves that this method can adaptively solve the low-rank graph structure problem and also

* 基金项目: 国家科技攻关计划 (2020AAA0106300); 国家自然科学基金 (62250008, 62222209, 62102222, 61936011); 北京信息科学与技术国家研究中心基金 (BNR2023RC01003)
收稿时间: 2023-05-04; 修改时间: 2023-09-16; 采用时间: 2023-10-03; jos 在线出版时间: 2024-02-28

proves that over-parameterized residual links on node deep representations can effectively address semantic mismatch. Experimental results on real datasets demonstrate that the OPGNN method is more robust than existing baseline methods, and the OPGNN framework is notably effective on different graph neural network backbones such as GCN, APPNP, and GPRGNN.

Key words: semi-supervised classification of graph nodes; graph structure adversarial defense; over-parameterization; implicit regularization; graph neural network (GNN)

图 (graph) 是一种普适的数据格式, 图数据的节点特征和节点间邻接关系分别可以描述不同对象及对象间的复杂关系. 近年来被用于分析图数据的图神经网络 (graph neural network, GNN)[1−3]在药物发现[4]、社交网络[5]、交通预测[6]、知识表示[7]等众多应用中具有重要意义. 近期研究表明, 图神经网络的性能会被图结构 (graph structure) 上的扰动显著影响[8]. 由于图神经网络的消息传递机制 (message passing mechanism)[9]在多层网络中将图结构上变化导致的错误信息逐层积累放大, 同时结构上的扰动可能会屏蔽正确信息在节点间的传递, 因此图结构上的微小扰动即可剧烈降低图神经网络在图节点分类任务上的性能.

本文研究图结构被扰动过的直推式 (transductive) 图节点半监督学习[3], 即研究在存在非定向对抗攻击 (non-targeted adversarial attack) 下的图节点分类任务的对抗防御. 图节点任务上的定向攻击指通过扰动降低预先选定的节点集合内节点的分类准确率, 通常情况下通过扰动目标节点的邻接图结构或特征实现. 区别于定向攻击, 非定向攻击期望通过扰动降低整体测试节点集合的分类准确率, 因此非定向攻击相比定向攻击更难检测, 面向非定向攻击的对抗防御更加困难.

面向针对图结构的非定向对抗攻击, 一类有效的图节点分类对抗防御方法是基于图内聚 (community preservation) 先验的图结构学习 (graph structure learning, GSL) 方法[10]. 图内聚先验即: 对于下游任务而言具有相同标签的节点普遍具有相似的节点特征, 因此这些节点有更高的概率连边, 从而也更可能形成具有内聚性的图结构. 由图频谱理论 (graph spectral theory)[11]可以得知一个邻接矩阵的秩 (rank) 与该图中的连通分量的数量有关, 而低秩图具有更为密集的连通分量. 因此为了更好地实现图内聚性质, 通常对图的邻接矩阵做低秩约束. 然而求解矩阵的秩需要求解矩阵极大线性无关组的个数, 是一个 NP-难的问题, 在机器学习过程中想要直接计算矩阵秩并进行优化非常困难. 因此多数工作[12−14]会考虑优化矩阵秩的凸松弛问题: 优化矩阵的核范数 (nuclear norm) 作为代理优化目标. 核范数被定义为矩阵特征值的核, 其计算一般依赖于矩阵的奇异值分解 (singular value decomposition, SVD), 然而这一过程的时间复杂度高, 且在梯度反向传播过程中会出现数值不稳定的现象[15]. 另外, 有工作将矩阵的低秩优化问题转化为低秩约束问题[16], 然而如何选取合适的低秩真值依赖于高复杂度的超参数调优. 总之, 一个亟需解决的问题是现有方法无法自适应秩真值进行低秩图结构重构.

另一方面, 考虑到图结构的潜在度量空间与下游任务语义度量空间存错配, 因此往往需要将学习到的图结构与任务进行适配. 一种自然的想法是通过添加稀疏的残差链接以合并低秩图结构与稀疏的错配结构描述完整的图结构[17−19]. 当原始图结构未被攻击时, 稀疏的残差链接可以有效建模任务语义度量与图结构度量上的错配, 然而当图结构被攻击时, 低秩图结构学习、结构上的稀疏错配学习受到图滤波器学习共同作用, 图结构上稀疏的残差链接对稀疏错配的学习与低秩结构学习互相耦合影响, 形成一个难以优化的联合非凸问题, 使得所学图结构并不能有效适配到下游任务的语义度量空间. 因此, 另一个亟需解决的问题是解决所学低秩图结构与下游任务语义的错配.

为了解决现有基于图结构学习的直推式图节点半监督学习对抗防御方法中存在的: (1) 无法自适应秩真值进行低秩图结构重构. (2) 所学低秩图结构与下游任务语义的错配问题. 本文提出一种新的基于过参数化隐式正则效应的图神经网络, 称为过参数化图神经网络 (over-parameterized GNN, OPGNN) 方法. 本文证明, 在特定条件下, 无需提前指定矩阵秩的真值或进行核范数优化, 所提的 OPGNN 方法可以自适应求解低秩图结构邻接矩阵. 同时, 所提 OPGNN 方法将结构上的稀疏结构学习转化为节点深层表征 (即评分函数) 错配学习以解决语义错配问题, 本文证明当节点深层表征语义的错配稀疏时, 所提的 OPGNN 方法学习到的低秩图结构可以良好适配下游任务语义. 为了验证所提 OPGNN 方法的有效性, 本文验证 OPGNN 方法在多个真实图数据集 (Cora, Citeseer 及 Polblogs), 多种图神经网络骨干 (GCN[3]、APPNP[20]和 GPRGNN[21]等), 在 Metattack[22]针对图结构不同攻击强度下的半监督图节点分类性能. 实验结果表明 OPGNN 方法相对于现有基线方法具有更好的鲁棒性, 且攻击强度越高, 鲁棒性越显著.

本文第 1 节介绍与所提 OPGNN 方法有关的相关工作. 第 2 节介绍与所提 OPGNN 方法相关的预备知识. 第 3 节介绍本文与所提 OPGNN 方法相关的理论结果. 第 4 节介绍所提 OPGNN 方法的方法细节. 第 5 节展示所提 OPGNN 方法的实验性能. 最后, 在第 6 节对 OPGNN 方法进行总结并对未来工作进行展望.

1 相关工作

1.1 鲁棒图结构学习方法

面向针对图结构的非定向对抗攻击, 鲁棒图结构学习方法主要设计思路基于图内聚先验的低秩图结构重构. GCN-SVD[13]发现对图结构的攻击可能影响图在谱域中的高秩部分, 并使用截断的奇异值分解方法对邻接矩阵进行低秩逼近. Pro-GNN[23]对图的稀疏性、低秩性和特征平滑性的显式正则化来学习清除噪音后的图结构. SimP-GCN[24]利用额外的自监督任务约束原始节点特征空间和隐层节点表示空间的距离差异, 鼓励图神经网络学得一个保距映射, 尝试提升模型对抗噪音的鲁棒性. LRGNN[16]将原始图结构拆分为噪声矩阵和去噪后的真实图邻接矩阵, 并分别用稀疏性与低秩性对噪声矩阵与真实邻接矩阵进行约束. STABLE[25]与 SimP-GCN 类似, 利用基于对比学习的无监督学习在深层表示空间重构图结构, 使图特征的表示学习与图结构学习在无监督训练信号下相互促进. 低秩图结构重构对一系列具体任务具有重要意义, 如在社交图像理解[26]中, 可以通过在一种新颖深度协同嵌入方法中无缝整合多个低秩图结构重构, 取得显著的性能提升; 如在社交图像检索[27]中, 可以通过实现鲁棒图结构优化与下游任务无缝地兼容适配, 以显著提升下游任务的性能.

面向图结构度量空间和下游语义任务空间的错配, 一些鲁棒图结构学习方法在结构上适配语义错配. AGC 方法[17]将初始图结构的拉普拉斯矩阵与更新后的图拉普拉斯矩阵进行加权求和. GAUG-O[19]将优化的图连边概率与原始图结构加权叠加后得到新的连边概率, 再根据此概率进行采样. 基于通道维度注意力机制的将原始结构和残差结构结合的聚合方法, 如研究者提出了一种用于基于的通道注意力的混合消息传递机制[18], 该机制利用注意力权重为初始图结构和优化图结构的每一个元素分配注意力从而进行信息聚合.

与上述方法不同, 本文提出的 OPGNN 方法基于过参数化隐式正则效应自适应求解低秩图结构, 同时考虑节点深层表征 (即评分函数) 错配, 解决了当前工作中存在的低秩超参调参困难与结构任务语义错配问题.

1.2 基于过参数化的隐式正则方法

近期, 区别于传统的构造显式正则项的正则方法, 一系列工作揭示了通过过参数化待学习参数, 在特定的初始化条件及更新规则下, 无需显式构造正则项即可实现对机器学习算法的正则化效应. Gunasekar 等人揭示了线性矩阵分解过参数化可以学习到低秩的矩阵重构[28], 奠定了隐式正则化研究的重要基础. Arora 等人拓展了 Gunasekar 等人的工作, 揭示了多层过参数化具有更强的低秩隐式正则效应[29]. Vaškevičius 等人[30]和 Zhao 等人[31]同时独立发现了哈达玛积过参数化的稀疏正则效应, 区别是 Vaškevičius 等人关注最优恢复补全, 而 Zhao 等人更关注稀疏鲁棒回归. You 等人通过引入异步学习率证明了可以通过过参数化同时起到低秩和稀疏两种正则效应[32]. 最近, Liu 等人研究者设计了过参数化的卷积神经网络并验证了隐式正则效应对鲁棒图像分类的有效性[33]. 然而, 目前过参数化的图卷积神经网络仍缺乏足够研究, 本文提出了线性算子谱图神经网络和基于过参数化的 OPGNN 方法, 首次在鲁棒图数据分析领域利用过参数化的隐式正则效应提升了图神经网络在图节点半监督学习的结构对抗防御性能.

2 预备知识

2.1 方法相关预备知识

本节介绍本文方法相关的预备知识.

首先, 介绍与图相关的基本概念. 一个 N-节点带权特征图 (N-node weighted feature graph, 简称为图) 是一个四元组 $G=(V,E,A,\mathbf{f})$. 其中 $V=\{1,\ldots,N\}$ 称为节点集 (node set), $E=\{(i,j)\}\subset V\times V$ 称为边集 (edge set). 矩阵 $A=\{a_{ij}\}_{i,j}$ 称为图 G 的邻接权重矩阵 (简称为权重矩阵或邻接矩阵), 其满足 $a_{ij}\in(0,1]$ 仅当边 $(i,j)\in E$, $a_{ij}=0$ 仅当 $(i,j)\notin E$.

一个图信号 (graph signal) 被定义为如下函数 $\mathbf{f}: V \to \mathbb{R}^F$, 即将每个图节点映射至 $\mathbb{R}^F$ 空间中其对应的 $F \in \mathbb{N}$ 维图节点信号向量 (graph node signal vector).

为了方便阐释, 本文亦使用 $\mathbf{f}$ 代表图信号矩阵 (signal matrix), 即 $\mathbf{f} = (\mathbf{f}_1, \ldots, \mathbf{f}_N)^\top \in \mathbb{R}^{N\times F}$, 其中 $\mathbf{f}_i \in \mathbb{R}^F$ 表示第 i 个图节点的图信号向量 (简称为节点 i 的图信号). 第 i 个图节点的度 (degree) d_i 定义为 $d_i := \sum_{j=1}^{N} a_{ij}$, 图 G 的拉普拉斯矩阵 (graph Laplacian) $\mathbf{L}$ 定义为 $\mathbf{L} := D - A$, 其中 D 为图 G 的度矩阵 (degree matrix), 度矩阵是一个对角矩阵, 第 i 个对角元为第 i 个图节点的度, 即 $D := \mathrm{diag}(d_1, \ldots, d_N)$. 给定图 $G = (V, E, A, \mathbf{f})$, 其拉普拉斯矩阵 $\mathbf{L}$ 由邻接矩阵 A 确定, 因此亦可用图拉普拉斯矩阵替代邻接矩阵 A 表示一个图 G, 即 $G = (V, E, \mathbf{L}, \mathbf{f})$.

接下来介绍与图神经网络相关的概念. 一般来说, 图神经网络可以看做是对输入空间中原始图信号矩阵 $\mathbf{f} = (\mathbf{f}_1, \ldots, \mathbf{f}_N)^\top \in \mathbb{R}^{N\times F}$ 根据图的拉普拉斯矩阵 $\mathbf{L}$ (或图的邻接权重矩阵 A) 进行滤波的特征映射, 即一个图神经网络是如下映射: $\Theta_{\mathbf{L}}(\mathbf{f}): \mathbb{R}^{N\times F} \to \mathbb{R}^{N\times F_{\text{out}}}$, $\mathbf{f} \mapsto \mathbf{z} = (\mathbf{z}_1, \ldots, \mathbf{z}_N)^\top$, 其中 $\mathbf{z} \in \mathbb{R}^{N\times F_{\text{out}}}$ 表示图的深层表征矩阵 (deep latent representation matrix), 且 F_{out} 表示每个节点深层表征向量的维度. 当下游任务为图节点的 2-分类任务时, 一般取 $F_{\text{out}} = 1$. 从图信号的滤波视角出发, 谱图神经网络 (spectral GNNs) (本文聚焦于谱图神经网络, 基于启发式消息传递图神经网络 heuristic MPNNs 的相关研究留作未来研究方向) 具有较好的理论释义: 可以被看作在图拉普拉斯矩阵 $\mathbf{L}$ 的频谱 Λ 上基于一个多项式滤波器 $h(\Lambda)$ 对图信号矩阵 $\mathbf{f}$ 进行滤波[34], 即:

$$\mathbf{z} = \Theta_{\mathbf{L}}(\mathbf{f}) \overset{\text{def}}{=} U h(\Lambda) U^\top g(\mathbf{f}) \tag{1}$$

其中, $U \in \mathbb{R}^{N\times N}$ 是图拉普拉斯矩阵的特征向量构成的矩阵, 并满足 $U\Lambda U^\top = \mathbf{L}$ (图拉普拉斯矩阵 $\mathbf{L}$ 是一个对称半正定矩阵, 因此总存在矩阵 $U, \Lambda \in \mathbb{R}^{N\times N}$, 使分解 $U\Lambda U^\top = \mathbf{L}$ 成立), $g(\mathbf{f})$ 表示对原始图信号矩阵 $g(\mathbf{f})$ 进行特征变换, $g(\cdot)$ 一般由全连接神经网络参数化并在端到端优化中被学习.

公式 (1) 的滤波过程如下.

(1) 首先, 图傅里叶变换 (graph Fourier transform) $U^\top g(\mathbf{f})$ 将特征变换过的图信号矩阵 $g(\mathbf{f})$ 映射至图的谱域 (spectral domain).

(2) 然后, 图滤波器 $h(\Lambda)$ 对谱域上的投影 $U^\top g(\mathbf{f})$ 进行滤波, 滤波后的谱图投影变为 $h(\Lambda)U^\top g(\mathbf{f})$, 其中 $h(\cdot)$ 作用在一个矩阵的每一个元素上.

(3) 最后, 将滤波后的谱图投影 $h(\Lambda)U^\top g(\mathbf{f})$ 利用图傅里叶逆变换 (inverse graph Fourier transform) 矩阵 $U^\top$ 映射回空域 (spatial domain), 即 $\mathbf{z} = U h(\Lambda) U^\top g(\mathbf{f})$.

当滤波器 $h(\Lambda)$ 是一个多项式时, 显然有 $U h(\Lambda) U^\top = h(\mathbf{L})$, 因此谱图神经网络经常写为公式 (2):

$$\mathbf{z} = \Theta_{\mathbf{L}}(\mathbf{f}) \overset{\text{def}}{=} h(\mathbf{L}) g(\mathbf{f}) \tag{2}$$

相较于公式 (1), 公式 (2) 不需要显式地求解图拉普拉斯变换矩阵 U, 在频谱矩阵 Λ 存在取值相同的特征值时, 图拉普拉斯矩阵 U 不具有唯一解, 因此基于公式 (2) 设计的图神经网络具有更好的算法稳定性, 基于公式 (2) 的常见谱图神经网络有 GCN[3]、APPNP[20]和 GPRGNN[21]等.

2.2 图节点半监督学习对抗防御问题设定

图节点半监督学习: 给定图 $G = (V, E, A, \mathbf{f})$, 节点集合 V 中的每个图节点 i 都有一个标签 y_i, 其中一部分节点有真实标签, 一部分节点没有标签, 图节点半监督学习的目标是预测未标记节点的标签, 即找到一个函数 $\Theta: V \to Y$, 其中 Y 表示标签集合, 使得预测的标签与真实标签尽可能接近.

图节点半监督学习: 当图 $G = (V, E, A, \mathbf{f})$ 存在对抗攻击时, 即对邻接矩阵 A 和 $\mathbf{f}$ 进行扰动 (通常扰动幅度较小, 使得对抗攻击不易被察觉), 图节点半监督学习对抗防御的目标是使半监督学习模型的标签预测 $\Theta: V \to Y$ 相对于对抗攻击鲁棒.

3 理论分析

在介绍本文方法细节前, 本节阐释方法相关的理论结果, 以说明所提方法设计思路来源及其合理性. 本节首先给

出本文所考虑的一类线性图神经网络——线性算子谱图神经网络, 然后介绍该类线性算子谱图神经网络的相关性质.

在神经网络的理论分析中, 非线性激活函数 (non-linear activation function) 对理论分析带来极大的困难, 为了分析的便利性和结论的普适性, 常常考虑分析神经网络的线性情形[29,35], 以得出相对有效的结论. 另一方面, 在神经网络的实践中, 常常将线性情形得到的结论套入到非线性神经网络中, 设计新的网络结构或算法, 往往可以得到显著的实践结果[33].

在本节中, 我们关注一类基于谱图神经网络的线性图神经网络, 本文将这类线性图神经网络称为线性算子谱图神经网络 (linear operator spectral GNN). 在第 2 节中曾指出, 给定图即 $G=(V,E,\mathbf{L},\mathbf{f})$, 谱图神经网络 $\Theta_{\mathbf{L}}(\mathbf{f})$ 的一种一般形式由公式 (2) 给出: $\mathbf{z}:=\Theta_{\mathbf{L}}(\mathbf{f})=h(\mathbf{L})g(\mathbf{f})$, 其中图的深层表征矩阵 $\mathbf{z}=(\mathbf{z}_1,\ldots,\mathbf{z}_N)^{\top}\in\mathbb{R}^{N\times F_{\text{out}}}$. 由于本文所关注的图数据分析的下游任务为图节点分类, 因此可以将图节点的深层表征向量 (标量) 维度设置为 1, 即 $F_{\text{out}}=1$, 此时 $\mathbf{z}\in\mathbb{R}^N$. 本文将线性算子图神经网络定义为可以将 $\Theta_{\mathbf{L}}(\mathbf{f})$ 写做作用在谱滤波器 $h(\mathbf{L})$ 上的由 $g(\mathbf{f})$ 参与参数化的线性算子 (linear operator) $\mathcal{A}_{g(\mathbf{f})}$. 在不引起歧义时, 略去下标记为 $\mathcal{A}$. 正式的定义如下.

定义 1. 线性算子谱图神经网络. 给定 N-节点带权特征图 $G=(V,E,\mathbf{L},\mathbf{f})$, 给定一个图信号特征变换映射 $g(\cdot)$ 和一个谱滤波器 $h(\mathbf{L})$, 图 G 的线性算子谱图神经网络 $\mathcal{A}_{g(\mathbf{f})}:\mathbb{R}^{N\times N}\to\mathbb{R}^N$ 是一个由下述映射 (隐式) 指定的线性算子:

$$[\mathcal{A}_{g(\mathbf{f})}(h(\mathbf{L}))]_i=<A_{g(\mathbf{f})}^{(i)},h(\mathbf{L})>,\quad A_{g(\mathbf{f})}^{(i)}\in\mathbb{R}^{N\times N},\quad i=1,\ldots,N \tag{3}$$

其中, $<\cdot,\cdot>$ 表示矩阵的内积, 即 $<A,B>:=Trace(AB^{\top})$. 此时, 若 $\mathbf{z}\in\mathbb{R}^N$ 表示线性算子谱图神经网络滤波后的向量, 则:

$$\mathbf{z}\overset{\text{def}}{=}\Theta_{\mathbf{L}}(\mathbf{f})=\mathcal{A}_{g(\mathbf{f})}(h(\mathbf{L})) \tag{4}$$

在图节点分类任务中, 一般只有一个 N-节点带权特征图 G, 因此在不引起歧义时可略去标示图的下标 $g(\mathbf{f})$, 将线性算子谱图神经网络 $\mathcal{A}_{g(\mathbf{f})}(h(\mathbf{L}))$ 简记为 $\mathcal{A}(h(\mathbf{L}))$.

为了方便阐释, 本节考虑图节点的 2-分类任务 (多分类任务仅需考虑多个信道的线性算子谱图神经网络, 多个信道的评分函数及 *Softmax* 层). 在图节点 2-分类任务中, 常将类别标记为 0 类和 1 类, 并期望通过图神经网络拟合全部节点属于 0 类的真值评分函数 (score function) $\mathbf{z}^{\text{true}}:\mathbb{R}^{N\times F}\to\mathbb{R}^N$. 分类时, 仅需将拟合的评分函数 $\mathbf{z}=\mathcal{A}(h(\mathbf{L}))$ 输入给 Sigmoid 激活 $S(x)=1/(1+\mathrm{e}^{-x})$ 或通过硬阈值法 (hard-thresholding) 确定分类即可. 换句话说, 面向图节点的 2-分类任务时, 常常需要进行如下优化, 并期望所学习到的图拉普拉斯矩阵 $\mathbf{L}$ 具有更好的抵御结构上对抗攻击的能力:

$$\min_{\mathbf{L}\succcurlyeq 0}\|\mathcal{A}(h(\mathbf{L}))-\mathbf{z}^{\text{true}}\|_2^2 \tag{5}$$

考虑到线性算子谱图神经网络 $\mathcal{A}(\cdot)$ (或一般的基于图结构学习的图节点分类图神经网络) 是一个 $\mathbb{R}^{N\times N}$ 空间到 $\mathbb{R}^{N\times N}$ 空间的映射, 当图 G 规模较大时, $N\ll N^2$, 此时即使约束 $\mathbf{L}$ 为半正定矩阵, 即 $\mathbf{L}\succcurlyeq 0$, 优化问题式 (5) 仍然为一个欠定系统 (underdetermined system), 此时存在无穷多个全局最小值满足 $\mathcal{A}(h(\mathbf{L})=\mathbf{z}^{\text{true}}$. 对于这一类欠定系统, 仅优化公式 (5) 并不能保证得到一个具有泛化能力的线性算子谱图神经网络 $\mathcal{A}(h(\mathbf{L}))$.

为了得到具有泛化能力的 (线性算子谱) 图神经网络, 需要额外的归纳偏置 (inductive bias) 在公式 (5) 的最优解集合中找到具有泛化能力的子集. 一种常用的归纳偏置为图内聚 (community preservation) 先验, 该先验认为对于下游任务而言具有相同标签的节点普遍具有相似的节点特征, 因此这些节点有更高的概率连边, 从而也更可能形成具有内聚性的图结构. 由于拉普拉斯矩阵 $\mathbf{L}$ 的特征值为 0 的个数等于图连通分量个数[11], 因此可以通过压缩拉普拉斯矩阵 $\mathbf{L}$ 的秩 (rank) 以使得图结构具有更好的内聚性: 鼓励图结构具有更为密集的连通分量, 使得相似节点倾向于连通而不相似节点倾向于不连通. 此时在公式 (5) 的最小值解集合中找到使得符合图内聚先验的子集, 使得所学图神经网络具有良好泛化能力. 因此, 可以摈弃优化公式 (5) 转而优化公式 (6):

$$\min_{\mathbf{L}\succcurlyeq 0}\|\mathbf{L}\|_*,\ \text{s.t.}\ \mathcal{A}(h(\mathbf{L}))=\mathbf{z}^{\text{true}} \tag{6}$$

其中, $\|\cdot\|_*$ 表示矩阵的核范数 (nuclear norm), 即若 $\mathbf{L}=U\Lambda U^{\top}$ 则有 $\|\mathbf{L}\|_*:=Trace(\Lambda)$.

另一方面, 描述图结构的度量空间与下游任务语义的度量空间存在错配, 即拉普拉斯矩阵 $\mathbf{L}$ 蕴含的相似度度量与任务语义上的节点特征 $\Theta_{\mathbf{L}}(\mathbf{f})$ 蕴含的相似度度量存在错配. 在图谱分析的语境下, 拉普拉斯矩阵蕴含的度指图的

谱度量 (spectral metric), 谱度量与拉普拉斯矩阵的特征值和特征向量有关, 提供了有关图结构、连通性等图谱属性信息. 在图神经网络与图节点半监督学习的语境下, 节点特征蕴含的相似度度量即为分类任务语义的相似度, 隶属于同一类别的图节点在该度量下应该接近. 由于当下很多图节点半监督分类任务中节点的类别不仅依赖于图的结构信息, 也与图的原始图信号相关, 因此拉普拉斯矩阵蕴含的相似度度量与任务语义上的节点特征蕴含的相似度度量存在错配. 现有图结构学习方法试图在图结构上添加稀疏的残差连接以描述低秩图结构与下游任务语义的错配. 在本节所述线性算子图神经网络的上下文语境下, 这种结构上的稀疏残差连接可以被描述为如下优化问题:

$$\min_{\mathbf{L}\succcurlyeq 0}\|\mathbf{L}\|_*+\lambda\|\mathbf{s}\|_1,\ \text{s.t.}\ \mathcal{A}(h(\mathbf{L}+\mathbf{s}))=\mathbf{z}^{\text{true}} \tag{7}$$

然而, 由于谱滤波器 $h(\cdot)$ 一般取为多项式滤波器, 即 $h(A)=\sum_{k=0}^{K}w_kA^k$, 优化问题式 (7) 中 $h(\mathbf{L}+\mathbf{s})$ 会包含 $\mathbf{L}^k\mathbf{s}^{K-k}$, 这使得低秩结构 $\mathbf{L}$ 与稀疏结构 $\mathbf{s}$ 在优化问题式 (7) 中联合非凸 (jointly non-convex), 极大地增加了有效学习线性算子谱图神经网络的难度.

因此为解决上述挑战, 本文提出将结构上的稀疏结构学习转化为节点深层表征 (即评分函数) 上的错配: 仍考虑一个残差连接 $\mathbf{s}$, 此时错配发生在低秩结构诱导的评分函数 $\mathbf{z}=\mathcal{A}(h(\mathbf{L}))$ 与真值评分函数间 $\mathbf{z}^{\text{true}}$, 即 $\mathbf{z}+\mathbf{s}=\mathcal{A}(h(\mathbf{L}))+\mathbf{s}=\mathbf{z}^{\text{true}}$, 对应的优化问题如下:

$$\min_{\mathbf{L}\succcurlyeq 0}\|\mathbf{L}\|_*+\lambda\|\mathbf{s}\|_1,\ \text{s.t.}\ \mathcal{A}(h(\mathbf{L}))+\mathbf{s}=\mathbf{z}^{\text{true}} \tag{8}$$

优化问题式 (8) 的求解仍然存在挑战: 由于核范数与 ℓ_1-范数局部不可微, 在基于梯度下降的优化过程中会出现数值不稳定现象[15].

为求解局部不可微的优化问题, 近期有一系列工作将局部不可微问题转化为全局可微问题并证明求解全局可微问题会得到该局部不可微问题相同的解[28,30,32]. 受此启发, 接下来本文证明, 通过对 $\mathbf{L}$ 和 $\mathbf{s}$ 进行过参数化 (overparameterization), 在特定更新规则及初始化下, 求解一个全局可微的 ℓ_2-范数优化问题可以得到优化问题式 (8) 的解. 具体地, 本文提出定理 1.

给定 N-节点带权特征图 $G=(V,E,\mathbf{L},\mathbf{f})$, 给定一个图信号特征变换映射 $g(\cdot)$ 和一个多项式谱滤波器 $h(\mathbf{L})=\sum_{k=1}^{K}w_k\mathbf{L}^k$ (假设 $k=0$ 对应的偏执项被评分函数 $\mathbf{z}$ 吸收). 令 $\mathcal{A}^*$ 表示 $\mathcal{A}$ 的伴随算子 (adjoint operator), 线性算子 $\mathcal{A}$ 的伴随算子定义为 $\mathcal{A}^*:\underline{R}^m\to\mathbb{R}^{n\times n}$, $\mathcal{A}^*(r)=\sum_i r_iA^{(i)}$. 考虑由如下参数矩阵/向量 $\mathbf{X}_t(\gamma)\in\mathbb{R}^{N\times N}$, $\mathbf{u}_t(\gamma)$, $\mathbf{v}_t(\gamma)\in\mathbb{R}^N$ 的梯度流 (gradient flow) 指定的微分方程:

$$\begin{aligned}\dot{\mathbf{X}}_{\mathbf{t}}(\gamma)&\overset{\text{def}}{=}\lim_{\tau\to 0}\frac{\mathbf{X}_{t+\tau}(\gamma)-\mathbf{X}_t(\gamma)}{\tau}=-\mathcal{A}^*(\mathbf{r}_t(\gamma))\sum_{k=1}^{K}w_k(\mathbf{X}_t(\gamma)\mathbf{X}_t(\gamma)^\top)^{k-1}\mathbf{X}_t(\gamma),\\ \begin{pmatrix}\dot{\mathbf{u}}_t(\gamma)\\ \dot{\mathbf{v}}_t(\gamma)\end{pmatrix}&\overset{\text{def}}{=}\lim_{\tau\to 0}\left(\begin{pmatrix}\mathbf{u}_{t+\tau}(\gamma)\\ \mathbf{v}_{t+\tau}(\gamma)\end{pmatrix}-\begin{pmatrix}\mathbf{u}_t(\gamma)\\ \mathbf{v}_t(\gamma)\end{pmatrix}\right)/\tau=-\alpha\cdot\begin{pmatrix}\mathbf{r}_t(\gamma)^\circ\mathbf{u}_t(\gamma)\\ -\mathbf{r}_t(\gamma)^\circ\mathbf{v}_t(\gamma)\end{pmatrix}\end{aligned} \tag{9}$$

其中, $\mathbf{r}^\circ\mathbf{u}$ 表示向量间的哈达玛积 (Hadamard product), 即 $\mathbf{r}^\circ\mathbf{u}:=\sum_i r_iu_i$; 残差向量 $\mathbf{r}_t(\gamma)$ 为 $\mathbf{r}_t(\gamma):=\mathcal{A}(h(\mathbf{X}_t(\gamma)\mathbf{X}_t(\gamma)^\top)+\mathbf{u}_t(\gamma)^\circ\mathbf{u}_t(\gamma)-\mathbf{v}_t(\gamma)^\circ\mathbf{v}_t(\gamma)-\mathbf{z}^{\text{true}}$.

不难看出, 公式 (9) 中的梯度流经过离散化即为下述 ℓ_2-范数优化问题的梯度下降更新:

$$\min_{\mathbf{X}\in\mathbb{R}^{N\times N},\mathbf{u},\mathbf{v}\in\mathbb{R}^N}\frac{1}{2}\|\mathcal{A}(h(\mathbf{X}\mathbf{X}^\top))+\mathbf{u}^\circ\mathbf{u}-\mathbf{v}^\circ\mathbf{v}-\mathbf{z}^{\text{true}}\|_2^2 \tag{10}$$

定理 1 指出, 在特定初始条件下 ($\gamma\to 0$), 线性算子谱图神经网络 $\mathcal{A}$ 可交换时, 梯度流式 (9) 会收敛到 ($t\to\infty$) 优化问题式 (8) 的解. 正式地, 定理 1 表述如下.

定理 1. 假设线性算子谱图神经网络 $\mathcal{A}$ 满足交换性, 即对于任意 $1\leqslant i\neq j\leqslant N$, 成立 $A^{(i)}_{g(\mathbf{f})}A^{(j)}_{g(\mathbf{f})}=A^{(j)}_{g(\mathbf{f})}A^{(i)}_{g(\mathbf{f})}$. 将微分方程梯度流的初始条件设置如下:

$$\mathbf{X}_0(\gamma)\overset{\text{def}}{=}\gamma I_{N\times N},\quad\begin{bmatrix}\mathbf{u}_0(\gamma)\\ \mathbf{v}_0(\gamma)\end{bmatrix}\overset{\text{def}}{=}\begin{bmatrix}\gamma\mathbf{1}_{N\times 1}\\ \gamma\mathbf{1}_{N\times 1}\end{bmatrix} \tag{11}$$

令 $\mathbf{L}_t(\gamma) := \mathbf{X}_t(\gamma)\mathbf{X}_t(\gamma)^\top$. 将 $\mathbf{L}_t(\gamma)$, $\mathbf{u}_t(\gamma)$ 和 $\mathbf{v}_t(\gamma)$ 在 t 趋于无穷的极限点记作 $\mathbf{L}_\infty(\gamma)$, $\mathbf{u}_\infty(\gamma)$ 和 $\mathbf{v}_\infty(\gamma)$, 即:

$$\mathbf{L}_\infty(\gamma) \overset{\text{def}}{=} \lim_{t\to\infty} \mathbf{L}_t(\gamma), \quad \begin{bmatrix} \mathbf{u}_\infty(\gamma) \\ \mathbf{v}_\infty(\gamma) \end{bmatrix} \overset{\text{def}}{=} \begin{bmatrix} \lim\limits_{t\to\infty} \mathbf{u}_t(\gamma) \\ \lim\limits_{t\to\infty} \mathbf{v}_t(\gamma) \end{bmatrix} \tag{12}$$

则当初始化足够小时, 即 $\gamma \to 0$ 时, 谱滤波器为 1 阶多项式时 (即 $h(\mathbf{L}) = w\mathbf{L}$), 公式 (12) 中极限点将收敛到问题 (8) 的解. 具体地:

$$\hat{\mathbf{L}} \overset{\text{def}}{=} \lim_{\gamma\to 0} \mathbf{L}_\infty(\gamma), \quad \begin{bmatrix} \hat{\mathbf{u}} \\ \hat{\mathbf{v}} \end{bmatrix} \overset{\text{def}}{=} \begin{bmatrix} \lim\limits_{\gamma\to 0} \mathbf{u}_\infty(\gamma) \\ \lim\limits_{\gamma\to 0} \mathbf{v}_\infty(\gamma) \end{bmatrix} \tag{13}$$

公式 (13) 是下述优化问题的解:

$$\min_{\mathbf{L}\in\mathbb{R}^{N\times N},\mathbf{u},\mathbf{v}\in\mathbb{R}^N} \|\mathbf{L}\|_* + \lambda\|\mathbf{u}^\circ\mathbf{u} - \mathbf{v}^\circ\mathbf{v}\|_1, \text{ s.t. } \mathcal{A}(h(\mathbf{L})) + \mathbf{u}^\circ\mathbf{u} - \mathbf{v}^\circ\mathbf{v} = \mathbf{z}^{\text{true}} \tag{14}$$

证明: 定理证明思路类似文献 [28,32], 由于欠定问题最优解存在, 仅需证明该解满足公式 (14) 的 Karush-Kuhn-Tucker (KKT) 条件即可, 具体证明细节见附录 A.

4 过参数化图神经网络 (OPGNN) 方法

4.1 方法概览

本节介绍过参数化图神经网络 (OPGNN) 方法, 其设计思路基于第 3 节定理 1 的结论. 方法的框架图如图 1 所示, 给定受攻击图 $G = (V, E, A, \mathbf{f})$, 将重构邻接矩阵 $\hat{A}$ (或基于 $\hat{A}$ 计算得到的重构拉普拉斯矩阵 $\hat{\mathbf{L}}$) 初始化, 并将其及图节点信号向量构成的图信号矩阵 $\mathbf{f}$ 输入给一个图神经网络骨架 (如 GCN、APPNP、GPRGNN 骨架) $\Theta_{\hat{\mathcal{A}}}(\mathbf{f})$. 图神经网络骨架将处理过的深层表征矩阵 $\mathbf{z} = \Theta_{\hat{\mathcal{A}}}(\mathbf{f})$ 输入给 *Softmax* 层得到全部节点属于任意类别的拟合概率矩阵 *Softmax*($\mathbf{z}$), 再将 *Softmax*($\mathbf{z}$) 输入给线性算子谱图神经网络损失函数. 线性算子谱图神经网络损失函数通过梯度下降反馈更新更好的重构图结构 $\hat{A}$, 重新输入图神经网络骨架 $\Theta_{\mathbf{f}}(\mathbf{f})$ 不断迭代得到更具泛化能力的图节点分类神经网络. 同时, 由于受攻击图结构仍具有一定参考意义, 攻击者往往不会大幅度更改受攻击的图结构, 因此可以将重构图结构和受攻击图结构输入重构误差损失函数, 对整体学习进程的收敛速度进行加速.

4.2 方法细节

在定理 1 中, 对拉普拉斯矩阵进行如过参数化 $\mathbf{L}_t(\gamma) = \mathbf{X}_t(\gamma)\mathbf{X}_t(\gamma)^\top$ 及按照梯度流公式 (9) 更新仅能保证其半正定性, 而良好定义的图拉普拉斯矩阵需要满足其每一行的行和为 0, 过参数化拉普拉斯矩阵及按公式 (9) 更新并不能满足这一性质. 因此考虑对图结构临界矩阵 A 进行过参数化, 即:

$$A_t(\gamma) \overset{\text{def}}{=} \mathbf{X}_t(\gamma)\mathbf{X}_t(\gamma)^\top \tag{15}$$

事实上, 由于 $\mathbf{L} = D - A$, 且谱滤波器 $h(\cdot)$ 为多项式, 因此考虑 $h(A) = h(\mathbf{X}\mathbf{X}^\top)$ 并不影响第 3 节的结论, 重构学习低秩的邻接矩阵 $\hat{A}$ 仍能体现内聚性先验.

为了方便阐释理论结果, 第 3 节考虑了相对简单的 2-分类任务进行分析, 此时图深层表征 $\mathbf{z} = \Theta_{\hat{\mathcal{A}}}(\mathbf{f}) = \Theta_{\hat{\mathbf{X}}\hat{\mathbf{X}}^\top}(\mathbf{f}) \in \mathbb{R}^N$, 深层表征 $\mathbf{z}$ 表示全部节点属于 0 类的评分函数. 当面临 M-分类 ($M \geqslant 3$) 时, 图深层表征 $\mathbf{z} \in \mathbb{R}^{N\times F_{\text{out}}}$, 一般取 $F_{\text{out}} = M$, 此时图深层表征 $\mathbf{z}$ 的第 i 行第 j 个元素表示第 i 个节点属于第 j 类的评分, $j = 1, \ldots, M$. 在图节点分类任务为 M-分类时, 可以构造 M 个信道的谱滤波器 $\{h^{(j)}(\cdot)\}_{j=1}^M$:

$$h^{(j)}(A) \overset{\text{def}}{=} \sum_{k=1}^{K} w_k^j (A)^k, \quad j = 1, \ldots, M \tag{16}$$

其中, 信道上标 j 表示第 j 类. 相对应地, 与一般的多分类谱图神经网络相同, 将 M 个信道的评分函数构成图深层表征 $\mathbf{z} := (\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(M)}) \in \mathbb{R}^{N\times M}$, 其中,

$$\mathbf{z}^{(j)} = \Theta_{\hat{\mathcal{A}}}(\mathbf{f})^{(j)} \overset{\text{def}}{=} \mathcal{A}_{g(\mathbf{f})}(h^{(j)}(\hat{A})) \tag{17}$$

实践中真值评分函数 $\mathbf{z}^{\text{true}}$ 往往无法被直接观测, 更多的时候可观测的是表示图节点类别的独热编码 (one-hot

encoding) $\mathbf{Y}=(Y_1,\ldots,Y_N)^{\top}\in\mathbb{R}^{N\times M}$, 因此实践中需要将拟合的评分函数 $\mathbf{z}=\Theta_{\hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}}(\mathbf{f})$ 通过 *Softmax* 层映射至 $(0,1)$ 取值区间与独热编码 $Y=(y_1,\ldots,y_n)^{\top}$ 匹配后计算损失函数, 即对 $\mathbf{z}$ 的第 i 行第 j 列元素 $\mathbf{z}_i^{(j)}$ 考虑映射:

$$Softmax(\mathbf{z}_i^{(j)})=\frac{\exp(\mathbf{z}_i^{(j)})}{\sum_{j=1}^{M}\exp(\mathbf{z}_i^{(j)})} \tag{18}$$

记 $Softmax(\mathbf{z})$ 为 *Softmax* 函数公式 (18) 作用在 $\mathbf{z}$ 上的每一个函数得到的矩阵.

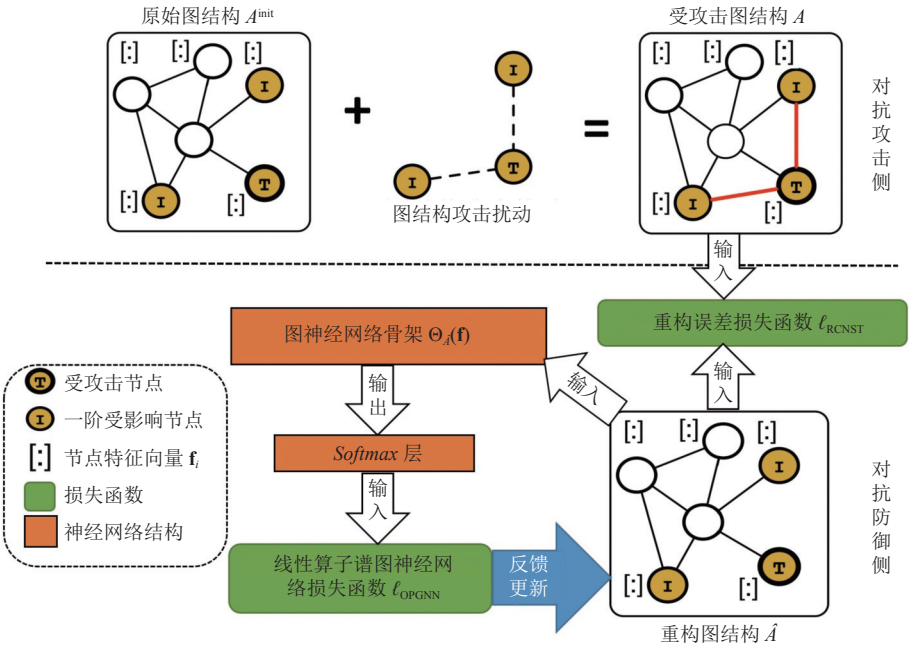


图 1 OPGNN 方法

同时, 注意到在 M-分类任务时到还需要匹配节点深层表征错配上的残差 $\mathbf{u}^{\circ}\mathbf{u}-\mathbf{v}^{\circ}\mathbf{v}=\mathbf{z}^{\text{true}}-\Theta_{\hat{A}}(\mathbf{f})$ 的维度, 此时需要 $\mathbf{u},\mathbf{v}\in\mathbb{R}^{N\times M}$, 故在 M-分类时, 过参数化参数矩阵初始化为:

$$\mathbf{X}_0(\gamma)\overset{\text{def}}{=}\gamma I_{N\times N},\quad \begin{bmatrix}\mathbf{u}_0(\gamma)\\ \mathbf{v}_0(\gamma)\end{bmatrix}\overset{\text{def}}{=}\begin{bmatrix}\gamma\mathbf{1}_{N\times M}\\ \gamma\mathbf{1}_{N\times M}\end{bmatrix} \tag{19}$$

基于第 3 节定理 1 的结论 OPGNN 方法的深度线性算子谱图神经网络损失函数 (分类任务实践中也可以将 l_2 损失函数更换为交叉熵损失函数, 本文表述仍使用 l_2 损失函数以兼容回归任务) 如下:

$$\ell_{\text{OPGNN}}\overset{\text{def}}{=}\min_{\mathbf{X}\in\mathbb{R}^{N\times N},\mathbf{u},\mathbf{v}\in\mathbb{R}^{N\times M}}\frac{1}{2}\|Softmax(\Theta'_{\hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}}(\Theta_{\hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}}(\mathbf{f})+\mathbf{u}^{\circ}\mathbf{u}-\mathbf{v}^{\circ}\mathbf{v}))-\mathbf{Y}\|_2^2 \tag{20}$$

另一方面注意到观测到的受攻击图结构 A 仍具有一定参考意义, 攻击者往往不会大幅度更改受攻击的图结构, 因此可以将重构图结构和受攻击图结构输入重构误差损失函数, 对整体学习进程的收敛速度进行加速, 具体的重构损失函数如下:

$$\ell_{\text{RCNST}}\overset{\text{def}}{=}\|\hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}-A\|_F \tag{21}$$

其中, $\|\cdot\|_F$ 表示矩阵的 Frobenius 范数.

最后, OPGNN 方法的整体优化目标如下:

$$L_{\text{OPGNN}}=\ell_{\text{OPGNN}}+\beta\ell_{\text{RCNST}} \tag{22}$$

其中, β 控制重构误差的正则强度, 适中的强度将加速重构结构学习.

本文所提 OPGNN 方法的算法流程如算法 1 所示.

算法 1. 面向图节点分类对抗防御的过参数化图神经网络 OPGNN.

输入: 图 $G=(V,E,A,\mathbf{f})$, 参数 $\mathbf{X}$, $\mathbf{u}$, $\mathbf{v}$, 图神经网络骨架网络参数 Θ, 训练迭代数 T, 图神经网络骨架网络参数学习率 η, $\mathbf{X}$ 学习率 τ, $\mathbf{u},\mathbf{v}$ 学习率 α, 图 G 的独热编码标签 $\mathbf{Y}$, $\mathbf{X}$, $\mathbf{u}$, $\mathbf{v}$ 参数初始化尺寸 γ, 重构损失函数权重 β, 谱滤波器阶数 K.

初始化: $\mathbf{X}_0(\gamma)\leftarrow\gamma I_{N\times N}$, $\mathbf{u}_0(\gamma)\leftarrow\gamma I_{N\times M}$, $\mathbf{v}_0(\gamma)\leftarrow\gamma I_{N\times M}$, 随机初始化网络参数 Θ

1. **while** $t\neq T$ **do**

2. #基于 (随机) 梯度下降训练图神经网络 Θ, Θ';

3. $\Theta\leftarrow\Theta-\eta\cdot\left[\partial\|Softmax(\Theta'_{\hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}}(\Theta_{\hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}}(\mathbf{f})+\mathbf{u}^{\circ}\mathbf{u}-\mathbf{v}^{\circ}\mathbf{v}))-\mathbf{Y}\|_2^2+\beta\|\hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}-A\|_F/\partial\Theta\right]$;

4. $\Theta'\leftarrow\Theta'-\eta\cdot\left[\partial\|Softmax(\Theta'_{\hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}}(\Theta_{\hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}}(\mathbf{f})+\mathbf{u}^{\circ}\mathbf{u}-\mathbf{v}^{\circ}\mathbf{v}))-\mathbf{Y}\|_2^2+\beta\|\hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}-A\|_F/\partial\Theta\right]$;

5. #更新 $\mathbf{X}$;

6. $X\leftarrow X-\tau\cdot\left[\partial\|Softmax(\Theta'_{\hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}}(\Theta_{\hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}}(\mathbf{f})+\mathbf{u}^{\circ}\mathbf{u}-\mathbf{v}^{\circ}\mathbf{v}))-\mathbf{Y}\|_2^2+\beta\|\hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}-A\|_F/\partial X\right]$;

7. #更新 $\mathbf{u},\mathbf{v}$;

8. $u\leftarrow u-\alpha\cdot[\partial\|Softmax(\Theta'_{\hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}}(\Theta_{\hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}}(\mathbf{f})+\mathbf{u}^{\circ}\mathbf{u}-\mathbf{v}^{\circ}\mathbf{v}))-\mathbf{Y}\|_2^2/\partial\mathbf{u}]$;

9. $v\leftarrow v-\alpha\cdot[\partial\|Softmax(\Theta'_{\hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}}(\Theta_{\hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}}(\mathbf{f})+\mathbf{u}^{\circ}\mathbf{u}-\mathbf{v}^{\circ}\mathbf{v}))-\mathbf{Y}\|_2^2/\partial\mathbf{v}]$;

10. **end while**

输出: 重构结构 $\hat{A}\leftarrow\mathbf{X}\mathbf{X}^{\top}$, 语义错配残差连接 $\mathbf{s}\leftarrow\mathbf{u}^{\circ}\mathbf{u}-\mathbf{v}^{\circ}\mathbf{v}$, 深度线性算子谱图神经网络 $\Theta'_{\hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}}(\Theta_{\hat{\mathbf{X}}\hat{\mathbf{X}}^{\top}}(\mathbf{f}))$.

5 实验分析

本节验证所提 OPGNN 方法的实验性能, 首先陈述实验设置; 然后展示 OPGNN 方法在多个数据集上与基线方法的性能对比, 并验证 OPGNN 方法在不同图神经骨架网络上的有效性; 最后对方法的组成成分进行分析.

5.1 实验设置

5.1.1 数据集

本文在 3 个常用半监督节点分类任务的基准数据集上验证了我们提出的 OPGNN 方法的有效性, 包括两个学术引用图数据集, 即 Cora 和 Citeseer 数据集, 以及一个网络博客相关的图数据集, 即 Polblogs 数据集. 这些数据集的统计数据如表 1 所示. 需要注意的是, 在 Polblogs 图中没有可用的节点特征. 在这种情况下, 我们采用和文献 [23,25] 等相同的方案, 将图信号矩阵设置为一个 $N\times N$ 的单位矩阵, 并类似文献 [25] 通过随机删减边进行样本增强.

表 1 实验数据集

数据集	节点集大小	边集大小	节点类别数	节点特征维度
Cora	2 485	5 069	7	1 433
Citeseer	2 110	3 668	6	3 703
Polblogs	1 222	16 714	2	—

5.1.2 基线方法与图神经骨干网络

为了充分验证 OPGNN 的鲁棒性与有效性, 本文将所提 OPGNN 方法与目前最先进的图结构对抗防御方法进行比较.

• RGCN[36]将隐藏层特征建模为高斯变量, 并在聚合邻居信息时对方差较大的节点分配低权重.

• GCN-Jaccard[37]使用节点特征的 Jaccard 相似度作为度量, 并消除连接不相似节点的连边以对抗结构上的扰动.

• GCN-SVD[13]发现对图结构的攻击可能影响图在谱域中的高秩部分, 并使用截断的奇异值分解方法对邻接矩阵进行低秩逼近.

• Pro-GNN[23]对图的稀疏性、低秩性和特征平滑性的显式正则化来学习清除噪音后的图结构.

• SimP-GCN[24]利用额外的自监督任务约束原始节点特征空间和隐层节点表示空间的距离差异, 鼓励 GCN 学

得一个保距映射, 尝试提升模型对抗噪音的鲁棒性.

• LRGNN[16]将原始图结构拆分为噪声矩阵和去噪后的真实图邻接矩阵, 并分别用稀疏性与低秩性对噪声矩阵与真实邻接矩阵进行约束.

• Elastic[38]构造了基于 ℓ_1 -范数和 ℓ_2 -范数的图节点信号表征光滑性正则化的消息传递聚合图神经网络, 通过鼓励表征之间的光滑性对图结构的攻击进行防御.

• STABLE[25]与 SimP-GCN 类似, 利用基于对比学习的无监督学习在深层表示空间重构图结构, 使图特征的表示学习与图结构学习在无监督训练信号下相互促进.

需要注意的是, GCN-Jaccard 和 Pro-GNN 方法都仅在节点特征可用时才有效. 此外, 我们还将以上图结构对抗防御方法与通用图卷积网络 GCN[3]和图注意力网络 GAT[39]进行了比较.

为了充分验证 OPGNN 的广泛适用能力, 本文在不同的图神经骨干网络上验证 OPGNN 的有效性, 包括:

• GCN[3]: 图卷积网络 GCN 从谱域图卷积的二阶切比雪夫多项式近似而来, 本质上是一个低通滤波器, 形如:

$$\mathbf{z} = \sigma(\mathbf{w}(I + \hat{A})\mathbf{f}) \tag{23}$$

其中, σ 为激活函数.

• APPNP[20]: APPNP 利用自定义 PageRank (personalized PageRank) 核来推导谱域图卷积. APPNP 的模型结构定义如下:

$$\mathbf{z} = \sum_{k=0}^{K} \alpha(1-\alpha)^k(\hat{A}^k g(\mathbf{f})) \tag{24}$$

其中, $\alpha \in (0,1]$ 是一个控制伸缩系数的超参数, $g(\cdot)$ 是一个参数化的神经网络对原始图信号矩阵 $\mathbf{f}$ 进行变换. APPNP 骨架首次分离了特征变换和传播两个步骤, 提高了其可扩展性.

• GPRGNN[21]: GPRGNN 使用单项式基函数逼近谱域图卷积, 也可以理解为基于广义 PageRank 核的图神经网络, 形式如下:

$$\mathbf{z} = \sum_{k=0}^{K} w_k(\hat{A}^k g(\mathbf{f})) \tag{25}$$

其中, w_k 是可学习的参数, $g(\mathbf{f})$ 是特征变换过的图信号矩阵.

5.1.3 超参数设置与实现细节

对于每一个数据集, 本文随机挑选其中 10% 的节点作为训练集, 10% 作为验证集, 以及剩余的 80% 作为测试集进行实验. 3 个数据集均为分类任务, 本文使用分类的准确率 (Accuracy) 作为评测指标. 对于所有实验, 本文用 10 个不同随机种子下的所有结果的均值与标准差作为汇报的方法性能. 为了公平对照, 所有方法的数据集划分方式是相同的. 所有本文 OPGNN 相关方法的图神经网络骨架学习率 η 固定为 1E–3, $\mathbf{X}$, $\mathbf{u}$, $\mathbf{v}$ 参数初始化尺寸 γ 固定为 5E–3. 在实验中剩余的超参数都在验证集上根据损失函数与准确率进行筛选. 具体地, 在方法 OPGNN 中, 低秩结构过参数化参数 $\mathbf{X}$ 的学习率 τ 和语义错配过参数化参数 $\mathbf{u}$, $\mathbf{v}$ 的学习率 α 从 {5E–5, 2E–5, 1E–5}中筛选, 重构误差损失函数的权重 β 从 $\{0.01, 0.1, 1, 5, 10\}$ 中进行筛选, 谱滤波器 $h(\cdot)$ 的阶数 K 从 $\{2, 3, 4\}$ 中筛选. 图神经网络骨架、$\mathbf{X}$, $\mathbf{u}$, $\mathbf{v}$ 参数的优化器均使用 Adam 优化器. 在 Cora 和 Citeseer 数据集上的训练轮次 T 取 2 000, 在 Polblogs 数据集上的训练训练轮次 T 取 3 000. 在本次实验中涉及的对原数据集的抗扰动方法为目前常用的非定向攻击算法 Metattack[22]. 实验运行于装备了 Nvidia RTX 3090 GPU 的机器, 方法框架搭建主要基于 PyTorch 1.11.0 及 DeepRobust[40]深度学习库.

5.2 定量实验

5.2.1 OPGNN 方法与先进基线方法的性能对比

由于现有主要基线方法都使用了 GCN 作为图神经网络骨干, 只有少数基线, 如 STABLE[25]、Elastic[38]使用或构造了更强表达能力的图神经网络骨干, 因此为了较为公平的与现有基线方法比较, 本节考察以 GCN 为图神经网

络骨干的 OPGNN 方法, 称之为 OPGNN-GCN.

本节考虑 6 种不同扰动率, 扰动强度分别为{0, 5%, 10%, 15%, 20%, 25%}, 扰动率为 a% 时指对当前图结构 a% 比例数目的边进行增减扰动. OPGNN-GNN 与基线方法 (基线的准确率参考表 2 中基线方法原论文自行汇报的结果) 在 Cora、Citeseer 和 Polblogs 的实验结果如表 2 所示, 可以看到本文所提方法在 Cora 和 Citeseer 上显著优于全部基线方法, 在 Polblogs 上的性能与 STABLE 方法相仿, 考虑到 Polblogs 并无节点特征向量, 因此主要性能来自于谱滤波器, 由于 Polblogs 使用了增强的 GCN, 因此其具有相对更好的谱滤波性能, 而 OPGNN-GCN 仅基于原始 GCN, 可以看出 OPGNN 的方法重构的图结构也具有很强的类别判别能力. 此外, 可以看到基线方法在扰动率加大时性能衰减较为严重, 而本文所提 OPGNN-GCN 方法在较大扰动率时具有更好的鲁棒性, 性能衰减更慢.

表 2　本文 OPGNN 方法与基线方法在不同结构扰动率下在 Cora、Citeseer 和 Polblogs 数据集上的准确率 (%)

数据集	扰动率	Elastic	STABLE	GCN	GAT	RGCN	GCN-Jaccard	GCN-SVD	Pro-GNN	SimP-GCN	LRGNN	OPGNN-GCN
Cora	0	84.76±0.53	85.58±0.56	83.50±0.44	83.97±0.65	83.09±0.44	82.05±0.51	80.63±0.45	82.98±0.23	83.69±0.45	83.42±0.30	**85.85±0.23**
	5	82.00±0.39	81.40±0.54	76.55±0.79	80.44±0.74	77.42±0.39	79.13±0.59	78.39±0.54	82.27±0.45	79.03±1.22	80.90±0.84	**83.71±0.45**
	10	76.18±0.46	80.49±0.61	70.39±1.28	75.61±0.59	72.22±0.38	75.16±0.76	71.47±0.83	79.03±0.59	75.74±1.66	77.47±0.87	**82.83±0.44**
	15	74.41±0.97	78.55±0.44	65.10±0.71	69.78±1.28	66.82±0.39	71.03±0.64	66.69±1.18	76.40±1.27	72.65±2.94	76.73±0.58	**78.89±0.20**
	20	69.64±0.62	**77.80±1.10**	59.56±2.72	59.94±0.92	59.27±0.37	65.71±0.89	58.94±1.13	73.32±1.56	70.11±6.39	72.86±0.93	75.59±0.43
	25	–	–	47.53±1.96	54.78±0.74	50.51±0.78	60.82±1.08	52.06±1.19	69.72±1.69	66.41±7.36	70.11±1.00	**73.84±0.65**
Citeseer	0	74.86± 0.53	**75.82±0.41**	71.95±0.55	73.26±0.83	71.20±0.83	72.10±0.63	70.65±0.32	73.28±0.69	74.25±0.66	73.13±0.33	75.79±0.60
	5	73.28±0.59	74.08±0.58	70.88±0.62	72.89±0.83	70.50±0.43	70.51±0.97	68.84±0.72	72.93±0.57	73.67±0.63	72.78±0.58	**74.88±0.42**
	10	73.41±0.36	73.45±0.40	67.55±0.89	70.63±0.48	67.71±0.30	69.54±0.56	68.87±0.62	72.51±0.75	73.07±1.37	72.11±1.23	**73.75±0.37**
	15	67.51±0.45	73.15±0.53	64.52±1.11	69.02±1.09	65.69±0.37	65.95±0.94	63.26±0.96	72.03±1.11	73.09±1.46	71.18±0.60	**73.66±0.45**
	20	65.65±1.95	72.76±0.53	62.03±3.49	61.04±1.52	62.49±1.22	59.30±1.40	58.55±1.09	70.02±2.28	70.08±3.55	66.11±0.76	**72.84±0.45**
	25	–	–	56.94±2.09	61.85±1.12	55.35±0.66	59.89±1.47	57.18±1.87	68.95±2.78	71.30±2.45	63.60±0.60	**72.32±0.37**
Polblogs	0	95.57±0.26	**95.95±0.27**	95.69±0.38	95.35±0.20	95.22±0.14	–	95.31±0.18	–	95.81±0.40	94.50±0.23	94.92±0.42
	5	90.08±1.06	93.80±0.12	73.07±0.80	83.69±1.45	74.34±0.19	–	89.09±0.22	–	72.97±2.20	93.30±0.33	**94.62±0.31**
	10	84.05±1.94	**92.46±0.77**	70.72±1.13	76.32±0.85	71.04±0.34	–	81.24±0.49	–	72.40±2.51	88.15±0.66	91.73±0.70
	15	72.17±0.74	90.04±0.72	64.96±1.91	68.80±1.14	67.28±0.38	–	68.10±3.73	–	67.54±2.92	86.22±1.38	**90.20±1.79**
	20	71.76±0.92	88.46±0.33	51.27±1.23	51.50±1.63	59.89±0.34	–	57.33±3.15	–	57.33±3.49	83.39±0.61	**89.20±1.93**
	25	–	–	49.23±1.36	51.19±1.49	56.02±0.56	–	48.66±9.93	–	56.40±2.87	75.53±1.06	**87.98±2.72**

5.2.2　OPGNN 方法对于不同图神经网络骨架的普遍适用

对比表 2 中 GCN 方法和 OPGNN-GCN 可以看出, 相对于原始的 GCN, 所提的过参数化图神经网络具有更强的鲁棒性, 这意味着过参数化在 GCN 图神经网络骨架上具有隐式正则效应. 为了验证这一效应的普适性, 我们在 APPNP 和 GPRGNN 图神经网络骨架上套用本文所提 OPGNN 方法, 将对应的过参数化网络分别称作 OPGNN-APPNP 和 OPGNN-GPRGNN.

仍考虑 6 种不同扰动率, 扰动强度分别为{0, 5%, 10%, 15%, 20%, 25%}, 考察原始 APPNP 和 OPGNN-APPNP 在不同扰动率下的分类准确率, 同时考察原始 GPRGNN 和 OPGNN-GPRGNN 在不同扰动率下的分类准确率, 结果分别如表 3 和表 4 所示. 结果表明, 在不同数据集上, OPGNN-APPNP 和 OPGNN-GPRGNN 在较强扰动率下相对原始图神经网络骨干具有更好的鲁棒性, 且扰动率越高优势越显著. 结合 OPGNN-GCN 相较于原始 GCN 的优势可以看出, 本文所提 OPGNN 方法框架对于不同的常见谱图神经网络就有较好的普适性.

5.2.3　OPGNN 方法的消融实验与敏感度分析

为了验证所提 OPGNN 方法中对低秩结构的有效学习以及对语义错配残差链接的有效学习, 我们考察两种消融基线: (i) OPGNN* : 仅考虑基于过参数化学习低秩结构 $\hat{A}=\mathbf{X}\mathbf{X}^{\top}$, 而不学习语义错配残差链接 $\mathbf{s}=\mathbf{u}^{\circ}\mathbf{u}-\mathbf{v}^{\circ}\mathbf{v}$. (ii) OPGNN† : 仅考虑基于过参数化学习语义错配残差链接 $\mathbf{s}=\mathbf{u}^{\circ}\mathbf{u}-\mathbf{v}^{\circ}\mathbf{v}$, 而不学习低秩结构 $\hat{A}=\mathbf{X}\mathbf{X}^{\top}$.

表 3 原始 APPNP 和过参数化 OPGNN-APPNP 实验结果 (%)

扰动率	Cora		Citeseer		Polblogs	
	GPRGNN	OPGNN-GPRGNN	GPRGNN	OPGNN-GPRGNN	GPRGNN	OPGNN-GPRGNN
0	**80.49±0.35**	77.95±0.22	73.28±0.62	**74.14±0.18**	**94.60±0.44**	93.19±0.44
5	**76.86±0.36**	75.17±0.35	71.90±0.61	**72.38±0.29**	68.69±0.84	**89.52±2.04**
10	74.57±0.59	**74.88±0.55**	**69.69±0.37**	69.39±0.71	64.81±1.85	**73.33±3.65**
15	73.14±0.48	**74.16±0.39**	67.02±0.83	**69.14±0.83**	49.23±1.35	**72.08±5.22**
20	69.65±0.49	**72.90±0.31**	64.44±0.46	**65.52±0.84**	48.60±1.47	**65.74±5.32**
25	66.44±0.74	**71.21±0.46**	64.29±1.04	**67.90±0.96**	47.60±1.95	**64.15±7.62**

表 4 原始 GPRGNN 和过参数化 OPGNN-GPRGNN 实验结果 (%)

扰动率	Cora		Citeseer		Polblogs	
	GPRGNN	OPGNN-GPRGNN	GPRGNN	OPGNN-GPRGNN	GPRGNN	OPGNN-GPRGNN
0	**80.80±0.38**	80.26±0.70	72.70±0.50	**73.38±0.43**	**94.42±0.43**	93.64±0.34
5	**77.11±0.33**	75.91±0.27	71.85±0.57	**72.79±0.36**	68.27±1.14	**90.60±1.59**
10	74.57±0.50	**75.03±0.74**	69.75±0.34	**69.76±0.81**	65.46±0.86	**76.41±4.11**
15	73.61±0.56	**74.36±0.49**	67.48±0.86	**69.73±0.71**	49.55±1.04	**71.90±6.71**
20	69.47±0.66	**72.90±0.67**	64.49±0.42	**66.80±0.89**	48.65±1.32	**67.61±6.28**
25	66.25±0.61	**71.48±0.37**	64.56±0.75	**68.42±0.87**	47.58±1.93	**65.69±6.78**

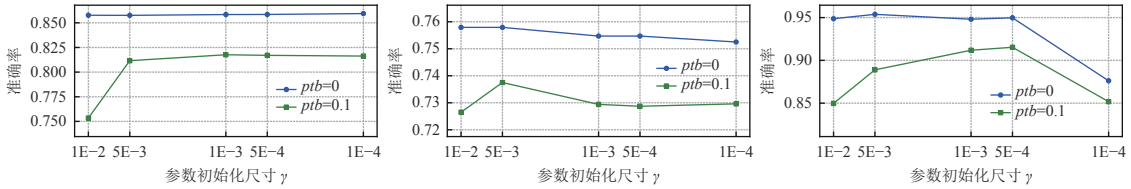
我们基于 GCN 图神经网络骨架实现 OPGNN* 和 OPGNN†, 在 6 种不同扰动率下考察上述两种 OPGNN 的消融方法在 Cora、Citeseer 和 Polblogs 数据集上的分类准确率, 与 OPGNN-GCN 的结果对比如表 5 所示. 表中结果表明, OPGNN-GCN 显著优于其消融基线方法 OPGNN* 和 OPGNN†, 验证了对低秩结构学习以及对语义错配残差链接学习的有效性和必要性. 在结构扰动率为 0 时, 仅考虑基于过参数化学习语义错配残差链接的变体 OPGNN† 要好于即考虑结构重构也考虑语义错配的 OPGNN, 这是因为在结构扰动率为 0 时, 过参数化的 OPGNN 学习不需要重构的邻接矩阵引入了更大的不确定性, 但在结构扰动率较大时, 过参数化的 OPGNN 学习并重构了更真实的邻接矩阵, 因此其性能更好. 另外可以看出 OPGNN* 的性能显著优于 OPGNN† 的性能, 说明了内聚性先验的强正则效应. 另一方面, 图低秩结构的学习也一定程度上弱化了原始图结构度量空间语义和下游任务语义的错配问题. 注意到, 表 5 中仅考虑基于过参数化学习语义错配残差链接的消融模型 OPGNN† 在 Polblogs 数据集上的准确率弱于表 2 中 GCN 方法. 由于 Polblogs 数据集不存在节点属性信息, 图分类单纯依赖于图结构拓扑. 消融模型 OPGNN† 不进行图低秩结构学习, 此时深层表征建模了图结构语义, 此时使用过参数化建模深层语义的错配链接相当于对图结构引入了额外的扰动, 因此消融模型 OPGNN† 在 Polblogs 数据集上的准确率弱于基线 GCN 方法. 这一观察从侧面说明了同时进行图低秩结构学习和语义错配学习的必要性.

表 5 OPGNN 方法与消融基线方法的实验结果 (%)

扰动率	Cora			Citeseer			Polblogs		
	OPGNN-GCN	OPGNN*	OPGNN†	OPGNN-GCN	OPGNN*	OPGNN†	OPGNN-GCN	OPGNN*	OPGNN†
0	**85.85±0.23**	84.15±0.32	84.54±0.33	**75.79±0.60**	73.13±0.50	72.32±0.67	94.92±0.42	94.79±0.80	**95.56±0.53**
5	**83.71±0.45**	83.03±0.33	78.74±0.85	**74.88±0.42**	73.27±0.43	71.19±0.66	94.62±0.31	**94.81±0.38**	72.74±0.82
10	**82.83±0.44**	80.28±0.50	72.92±1.49	**73.75±0.37**	71.65±0.78	68.36±1.22	**91.89±0.50**	91.73±0.70	70.89±1.29
15	**78.89±0.20**	78.55±0.37	68.53±1.53	**73.66±0.45**	71.96±0.37	65.41±0.79	**90.20±1.79**	90.10±1.77	64.16±4.45
20	**75.59±0.43**	75.44±0.60	58.03±0.98	**71.84±0.45**	70.52±0.47	56.94±1.24	**89.20±1.93**	89.00±0.48	50.31±2.40
25	73.84±0.65	**73.97±0.63**	52.91±1.29	**72.32±0.37**	70.29±0.26	57.63±1.21	**87.98±2.72**	86.23±2.45	48.63±2.38

接下来, 本节考察 OPGNN 方法 (OPGNN-GCN) 中在有结构扰动 (默认扰动率 10%) 和无结构扰动时, 参数初始化尺寸 γ、谱滤波器阶数 K、重构误差强度 β、低秩结构过参数化参数 $\mathbf{X}$、学习率 τ 和语义错配残差链接过参数化参数 $\mathbf{u}$, $\mathbf{v}$ 学习率 α 的敏感度.

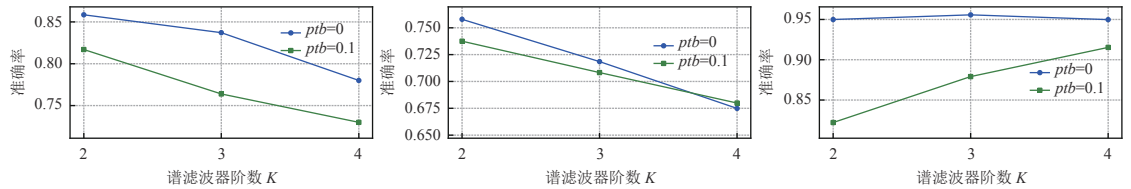
关于参数初始化尺寸 γ 的敏感度在 Cora、Citeseer 和 Polblogs 数据集上的表现如图 2 所示, 其中, *ptb* 表示扰动强度参数, 结果显示有结构扰动相较于无结构扰动, OPGNN 方法对初始化尺寸 γ 更敏感. 实验表现与定理 1 结论契合, 在尺寸参数适中时表现良好, 初始化尺寸过小时需要过长的迭代时间到达理想的极小值, 过大的初始化尺寸与定理 1 假设不符合. 参数初始化尺寸 γ 的敏感度的实验结果从侧面说明了定理 1 的合理性和 OPGNN 方法的有效性.



(a) Cora (b) Citeseer (c) Polblogs

图 2 OPGNN 方法中参数初始化尺寸 γ 的敏感度

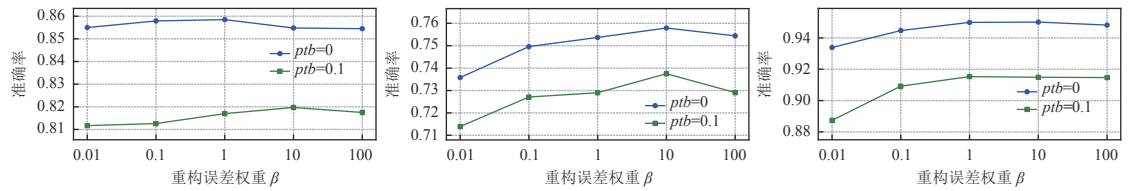
关于谱滤波器阶数 K 的敏感度在 Cora、Citeseer 和 Polblogs 数据集上的表现如图 3 所示. OPGNN 方法在 Cora 和 Citeseer 和 Polblogs 上呈现相反的趋势, 在 Cora 和 Citeseer 上的性能随阶数增高而减小, 这是因为越高阶的多项式滤波器具有更强的过平滑化 (over-smooth) 效应. Polblogs 数据集上的表现随阶数增高而变高, 由于 Polblogs 数据集没有图信号, 因此不受过平滑化效应的影响, 同时需要更高阶的多项式去探索更大的节点邻域更充分挖掘结构上的类别判别特征.



(a) Cora (b) Citeseer (c) Polblogs

图 3 OPGNN 方法中谱滤波器阶数 K 的敏感度

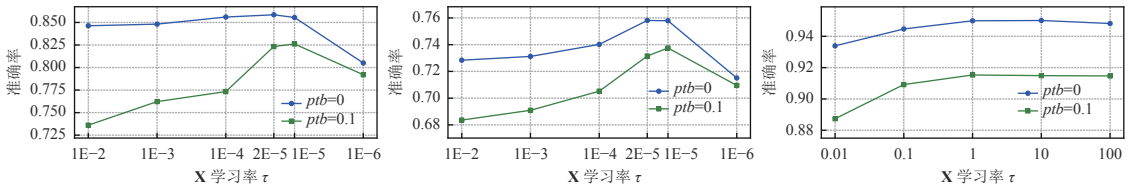
关于重构误差强度 β 的敏感度在 Cora、Citeseer 和 Polblogs 数据集上的表现如图 4 所示. 可以看到在无论在有无结构扰动时, 一个强度适中的重构误差都有助于性能提升, 这是因为重构误差可以为优化低秩结构重构提供有效的训练信号, 加速算法收敛, 避免由于迭代时间过长拟合噪音造成过拟合.



(a) Cora (b) Citeseer (c) Polblogs

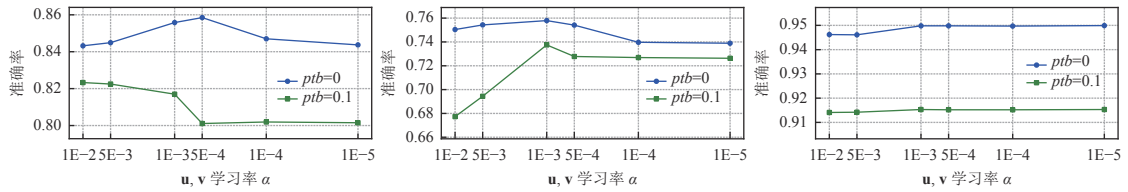
图 4 OPGNN 方法中重构误差强度 β 的敏感度

关于低秩结构过参数化参数 $\mathbf{X}$ 学习率 τ 和语义错配残差链接过参数化参数 $\mathbf{u}$, $\mathbf{v}$ 学习率 α 的敏感度分别如图 5、图 6 所示. 可以看到 OPGNN 方法对于 τ 相对 α 更为敏感, 侧面说明重构低秩结构能够一定程度上弱化语义错配. Polblogs 数据集上对 α 不敏感是因为该数据集无图信号矩阵, 因此类别判别特征完全依赖于图结构, 此时再考虑结构与下游语义错配没有意义. 同时可以看到, 使 OPGNN 算法性能达到最优的学习率 τ 和 α 并不一致, 这是因为定理 1 中的梯度流更新公式 (9) 蕴含了异步学习率, 本质上通过异步学习率控制稀疏隐式正则化相对低秩隐式正则化的强度. 图 5 和图 6 的结果说明了定理 1 中建模异步学习率的合理性和 OPGNN 方法的有效性.



(a) Cora (b) Citeseer (c) Polblogs

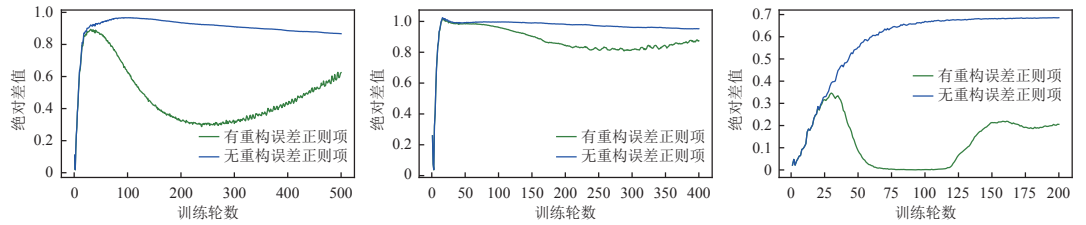
图 5 OPGNN 方法中低秩结构过参数化参数 **X** 学习率 τ 的敏感度



(a) Cora (b) Citeseer (c) Polblogs

图 6 OPGNN 方法中语义错配残差链接过参数化参数 **u**, **v** 学习率 α 的敏感度

为了检验重构损失函数对学习收敛速度的影响, 我们考察在结构扰动率 10% 时, 训练过程中训练集与验证集交叉熵损失绝对差值 (因为训练集损失会逐渐被优化至 0, 因此该差值可作为泛化误差的代理) 在有重构误差正则项和无重构误差正则项的差异. 在 3 个数据集上的结果如图 7 所示, 可以看到, 在有重构误差正则项时, 训练集与验证机交叉熵损失绝对插值呈现明显的先上升, 然后下降, 最后上升的三阶段过程. 初始时训练集和验证集的交叉熵损失都很高而差异较小; 在学习初期, 第 1 个阶段是模型学习在训练集损失快速下降验证集损失缓慢下降而呈现上升趋势; 第 2 个阶段是模型学习并收敛到有泛化能力的特征, 使得验证集损失下降, 此时训练集和验证集的交叉熵损失在第 2 个阶段差异变小; 此后, 模型逐渐过拟合训练集的噪音, 使得模型在验证集的损失变大, 进而使得训练集和验证集的交叉熵在第 3 个阶段的差异变大. 为了避免模型过拟合训练集中的噪音, 一般采取早停法 (early stopping), 在有重构误差损失函数时, 可以在第 2 个阶段采用早停法得到有泛化能力的模型. 相对应的, 在无重构误差正则项时, 在训练过程中, 训练集和验证集交叉熵损失绝对值一直较大, 这意味着模型无法快速收敛到有泛化能力的特征, 不断过拟合训练集的噪音, 使得通过早停法进行模型选择的策略失效.



(a) Cora 训练集与验证集交叉熵损失绝对差值 (b) Citeseer 训练集与验证集交叉熵损失绝对差值 (c) Polblogs 训练集与验证集交叉熵损失绝对差值

图 7 重构损失函数对经验泛化误差的影响

为了检验在面对高强度的结构攻击时重构误差损失函数的效用, 我们考察结构扰动率 25% 时 (更高强度的结构攻击在实际应用中不常见, 因为在实践中高强度攻击过于容易被检测, 同时攻击成本过高, 因此在广泛应用的评测框架 DeepRobust[40]深度学习库中, 也至多考察结构扰动率 25%) 不同重构误差正则强度 β 的 OPGNN 方法在 3 个数据集上的实验效果. 结果如表 6 所示, 表中结果显示, 在高强度结构攻击时, 强度适中的重构误差损失函数依然对学习有正面促进作用, 具体地, 依然有 75% 的可信边为学习低秩结构重构提供有效的训练信号, 隐式地加速算法收敛, 避免由于迭代时间过长拟合噪音造成过拟合, 使得对抗防御算法有效.

表 6　结构扰动率 25% 时不同重构误差强度 β 的 OPGNN 方法在 3 个数据集上的实验结果

β	Cora (%)	Citeseer (%)	Polblogs (%)
0.1	73.74±0.70	69.53±0.33	83.53±0.53
1	**73.84±0.65**	70.21±0.37	**87.98±2.72**
10	73.39±0.51	**72.32±0.37**	86.44±4.23
100	73.31±0.47	70.28±0.36	86.35±4.32

References:

[1] Bruna J, Zaremba W, Szlam A, LeCun Y. Spectral networks and locally connected networks on graphs. arXiv:1312.6203, 2014.

[2] Defferrard M, Bresson X, Vandergheynst P. Convolutional neural networks on graphs with fast localized spectral filtering. In: Proc. of the 30th Int'l Conf. on Neural Information Processing Systems. Barcelona: Curran Associates Inc., 2016. 3844–3852.

[3] Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks. arXiv:1609.02907, 2017.

[4] Wang YH, Min YS, Chen X, Wu J. Multi-view graph contrastive representation learning for drug-drug interaction prediction. In: Proc. of the 2021 Web Conf. Ljubljana: Association for Computing Machinery, 2021. 2921–2933. [doi: 10.1145/3442381.3449786]

[5] Qiu JZ, Tang J, Ma H, Dong YX, Wang KS, Tang J. DeepInf: Social influence prediction with deep learning. In: Proc. of the 24th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. London: Association for Computing Machinery, 2018. 2110–2119. [doi: 10.1145/3219819.3220077]

[6] Yu B, Yin HT, Zhu ZX. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In: Proc. of the 27th Int'l Joint Conf. on Artificial Intelligence. Stockholm: AAAI Press, 2018. 3634–3640.

[7] Wang Q, Mao ZD, Wang B, Guo L. Knowledge graph embedding: A survey of approaches and applications. IEEE Trans. on Knowledge and Data Engineering, 2017, 29(12): 2724–2743. [doi: 10.1109/TKDE.2017.2754499]

[8] Dai HJ, Li H, Tian T, Huang X, Wang L, Zhu J, Song L. Adversarial attack on graph structured data. In: Proc. of the 35th Int'l Conf. on Machine Learning. Stockholm: JMLR.org, 2018. 1115–1124.

[9] Gilmer J, Schoenholz SS, Riley PF, Vinyals O, Dahl GE. Neural message passing for quantum chemistry. In: Proc. of the 34th Int'l Conf. on Machine Learning. Sydney: JMLR.org, 2017. 1263–1272.

[10] Zhu YQ, Xu WZ, Zhang JH, Du YQ, Zhang JY, Liu Q, Yang C, Wu S. A survey on graph structure learning: Progress and opportunities. arXiv:2103.03036, 2022.

[11] Chung F. Spectral Graph Theory. Providence: American Mathematical Society, 1997.

[12] Friedland S, Lim LH. Nuclear norm of higher-order tensors. Mathematics of Computation, 2018, 87(311): 1255–1281. [doi: 10.1090/mcom/3239]

[13] Entezari N, Al-Sayouri SA, Darvishzadeh A, Papalexakis EE. All you need is low (rank): Defending against adversarial attacks on graphs. In: Proc. of the 13th Int'l Conf. on Web Search and Data Mining. Houston: Association for Computing Machinery, 2020. 169–177. [doi: 10.1145/3336191.3371789]

[14] Luo DS, Cheng W, Yu WC, Zong B, Ni JC, Chen HF, Zhang X. Learning to drop: Robust graph neural network via topological denoising. In: Proc. of the 14th ACM Int'l Conf. on Web Search and Data Mining. Association for Computing Machinery, 2021. 779–787. [doi: 10.1145/3437963.3441734]

[15] Ionescu C, Vantzos O, Sminchisescu C. Matrix backpropagation for deep networks with structured layers. In: Proc. of the 2015 IEEE Int'l Conf. on Computer Vision. Santiago: IEEE, 2015. 2965–2973. [doi: 10.1109/ICCV.2015.339]

[16] Xu H, Xiang LY, Yu JH, Cao AQ, Wang XB. Speedup robust graph structure learning with low-rank information. In: Proc. of the 30th ACM Int'l Conf. on Information and Knowledge Management. Association for Computing Machinery, 2021. 2241–2250. [doi: 10.1145/3459637.3482299]

[17] Li RY, Wang S, Zhu FY, Huang JZ. Adaptive graph convolutional neural networks. In: Proc. of the 32nd AAAI Conf. on Artificial Intelligence, the 30th Innovative Applications of Artificial Intelligence Conf. and the 8th AAAI Symp. on Educational Advances in Artificial Intelligence. New Orleans: AAAI Press, 2018. 3546–3553.

[18] Liu SQ, Chen Y, Xie XF, Siow JK, Liu Y. Retrieval-augmented generation for code summarization via hybrid GNN. arXiv:2006.05405, 2021.

[19] Zhao T, Liu Y, Neves L, Woodford O, Jiang M, Shah N. Data augmentation for graph neural networks. Proc. of the AAAI Conf. on Artificial Intelligence, 2021, 35(12): 11015–11023. [doi: 10.1609/aaai.v35i12.17315]

[20] Gasteiger J, Bojchevski A, Günnemann S. Predict then propagate: Graph neural networks meet personalized PageRank. arXiv:

1810.05997, 2019.
[21] Chien EL, Peng JH, Li P, Milenkovic O. Adaptive universal generalized PageRank graph neural network. arXiv:2006.07988, 2021.
[22] Zügner D, Günnemann S. Adversarial attacks on graph neural networks via Meta learning. arXiv:1902.08412, 2019.
[23] Jin W, Ma Y, Liu XR, Tang XF, Wang SH, Tang JL. Graph structure learning for robust graph neural networks. In: Proc. of the 26th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. Association for Computing Machinery, 2020. 66–74. [doi: 10.1145/3394486.3403049]
[24] Jin W, Derr T, Wang YQ, Ma Y, Liu ZT, Tang JL. Node similarity preserving graph convolutional networks. In: Proc. of the 14th ACM Int'l Conf. on Web Search and Data Mining. Association for Computing Machinery, 2021. 148–156. [doi: 10.1145/3437963.3441735]
[25] Li K, Liu Y, Ao X, Chi JF, Feng JH, Yang H, He Q. Reliable representations make a stronger defender: Unsupervised structure refinement for robust GNN. In: Proc. of the 28th ACM SIGKDD Conf. on Knowledge Discovery and Data Mining. Washington: Association for Computing Machinery, 2022. 925–935. [doi: 10.1145/3534678.3539484]
[26] Li ZC, Tang JH, Mei T. Deep collaborative embedding for social image understanding. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2019, 41(9): 2070–2083. [doi: 10.1109/TPAMI.2018.2852750]
[27] Li ZC, Tang JH, Zhang LY, Yang J. Weakly-supervised semantic guided hashing for social image retrieval. Int'l Journal of Computer Vision, 2020, 128(8): 2265–2278. [doi: 10.1007/s11263-020-01331-0]
[28] Gunasekar S, Woodworth B, Bhojanapalli S, Neyshabur B, Srebro N. Implicit regularization in matrix factorization. In: Proc. of the 31st Int'l Conf. on Neural Information Processing Systems. Long Beach: Curran Associates Inc., 2017. 6152–6160.
[29] Arora S, Cohen N, Hu W, Luo YP. Implicit regularization in deep matrix factorization. In: Proc. of the 33rd Int'l Conf. on Neural Information Processing Systems. Vancouver: Curran Associates Inc., 2019. 7413–7424.
[30] Vaškevičius T, Kanade V, Rebeschini P. Implicit regularization for optimal sparse recovery. In: Proc. of the 33rd Int'l Conf. on Neural Information Processing Systems. Vancouver: Curran Associates Inc., 2019. 2972–2983.
[31] Zhao P, Yang Y, He QC. High-dimensional linear regression via implicit regularization. Biometrika, 2022, 109(4): 1033–1046. [doi: 10.1093/biomet/asac010]
[32] You C, Zhu ZH, Qu Q, Ma Y. Robust recovery via implicit bias of discrepant learning rates for double over-parameterization. In: Proc. of the 34th Int'l Conf. on Neural Information Processing Systems. Vancouver: Curran Associates Inc., 2020. 17733–17744.
[33] Liu S, Zhu ZH, Qu Q, You C. Robust training under label noise by over-parameterization. In: Proc. of the 39th Int'l Conf. on Machine Learning. Baltimore: JMLR.org, 2022. 14153–14172.
[34] Ortega A, Frossard P, Kovačević J, Moura JMF, Vandergheynst P. Graph signal processing: Overview, challenges, and applications. Proc. of the IEEE, 2018, 106(5): 808–828. [doi: 10.1109/JPROC.2018.2820126]
[35] Wang XY, Zhang MH. How powerful are spectral graph neural networks. In: Proc. of the 39th Int'l Conf. on Machine Learning. Baltimore: JMLR.org, 2022. 23341–23362.
[36] Zhu DY, Zhang ZW, Cui P, Zhu WW. Robust graph convolutional networks against adversarial attacks. In Proc. of the 25th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. Anchorage: Association for Computing Machinery, 2019. 1399–1407. [doi: 10.1145/3292500.3330851]
[37] Wu HJ, Wang C, Tyshetskiy Y, Docherty A, Lu K, Zhu LM. Adversarial examples for graph data: Deep insights into attack and defense. In: Proc. of the 28th Int'l Joint Conf. on Artificial Intelligence. Macao: AAAI Press, 2019. 4816–4823.
[38] Liu XR, Jin W, Ma Y, Li YX, Liu H, Wang YQ, Yan M, Tang JL. Elastic graph neural networks. In: Proc. of the 38th Int'l Conf. on Machine Learning. Virtual Event: JMLR.org, 2021. 6837–6849.
[39] Veličković P, Cucurull G, Casanova A, Romero A, Liò P, Bengio Y. Graph attention networks. arXiv:1710.10903, 2018.
[40] Li YX, Jin W, Xu H, Tang JL. Deeprobust: A PyTorch library for adversarial attacks and defenses. arXiv:2005.06149, 2020.

附录 A. 定理 1 的证明

证明思路类似文献 [28,32], 由于欠定问题最优解存在, 仅需证明该解满足 (14) 的 Karush-Kuhn-Tucker (KKT) 条件即可.

令 $\mathbf{s} := \mathbf{u} \circ \mathbf{u} - \mathbf{v} \circ \mathbf{v}$, 考虑问题式 (14) 的拉格朗日函数:

$$\mathcal{L}(\mathbf{L}, \mathbf{s}, \boldsymbol{\nu}, \boldsymbol{\Gamma}) = Trace(L) + \lambda \|\mathbf{s}\|_1 + \boldsymbol{\nu}^{\top}(\mathbf{z}^{\text{true}} - \mathcal{A}(h(\mathbf{L})) - \mathbf{s}) - \langle \mathbf{X}, \boldsymbol{\Gamma} \rangle,$$

其中, $\boldsymbol{\nu}$ 和 $\boldsymbol{\Gamma}$ 为对偶变量. 由 Karush-Kuhn-Tucker (KKT) 条件可知, 公式 (14) 的最优解 $(\hat{\mathbf{L}}, \hat{\mathbf{s}})$ 需要满足如下条件——存在对偶变量 $\boldsymbol{\nu}$ 使得下述条件成立:

$$\mathcal{A}(\hat{\mathbf{L}})+\hat{\mathbf{s}}=\mathbf{y},\quad \hat{\mathbf{L}}\succcurlyeq 0\quad (\mathbf{I}-\mathcal{A}^*(\boldsymbol{\nu}))\cdot\hat{\mathbf{L}}=0,\quad \mathbf{I}\succcurlyeq\mathcal{A}^*(\boldsymbol{\nu}),\quad \boldsymbol{\nu}\in\lambda\cdot sign(\hat{\mathbf{s}}),$$

其中, $\mathcal{A}(\hat{\mathbf{L}})+\hat{\mathbf{s}}=\mathbf{y}$ 由欠定方程组系统的最优解自动满足 (即公式 (10) 存在解使该式取值为 0), $\hat{\mathbf{L}}\succcurlyeq 0$ 由分解构造 $\hat{\mathbf{L}}=\hat{\mathbf{X}}\hat{\mathbf{X}}^\top$ 自动满足, 因此只需要证明存在对偶变量 $\boldsymbol{\nu}$ 使得后 3 个条件成立即可.

接下来证明: 当 $h(A):=\sum_{k=1}^{K}w_kA^k$, 由初始条件

$$\mathbf{X}_0(\gamma)\overset{\text{def}}{=}\gamma I_{N\times N},\quad \begin{bmatrix}\mathbf{u}_0(\gamma)\\ \mathbf{v}_0(\gamma)\end{bmatrix}\overset{\text{def}}{=}\begin{bmatrix}\gamma\mathbf{1}_{N\times 1}\\ \gamma\mathbf{1}_{N\times 1},\end{bmatrix},$$

及梯度流

$$\dot{\mathbf{X}}_t(\gamma)=-\mathcal{A}^*(\mathbf{r}_t(\gamma))\sum_{k=1}^{K}w_k(\mathbf{X}_t(\gamma)\mathbf{X}_t(\gamma)^\top)^{k-1}\mathbf{X}_t(\gamma)\overset{K=1}{=}-\mathcal{A}^*(\mathbf{r}_t(\gamma))w\mathbf{X}_t(\gamma),\begin{bmatrix}\dot{\mathbf{u}}_t(\gamma)\\ \dot{\mathbf{v}}_t(\gamma)\end{bmatrix}=-\alpha\cdot\begin{bmatrix}\mathbf{r}_t(\gamma)^\circ\mathbf{u}_t(\gamma)\\ -\mathbf{r}_t(\gamma)^\circ\mathbf{v}_t(\gamma)\end{bmatrix}\tag{A1}$$

得到的极限点

$$\mathbf{L}_\infty(\gamma)\overset{\text{def}}{=}\lim_{t\to\infty}\mathbf{L}_t(\gamma)\overset{\text{def}}{=}\lim_{t\to\infty}\mathbf{X}_t(\gamma)\mathbf{X}_t(\gamma)^\top,\quad \begin{bmatrix}\mathbf{u}_\infty(\gamma)\\ \mathbf{v}_\infty(\gamma)\end{bmatrix}\overset{\text{def}}{=}\begin{bmatrix}\lim_{t\to\infty}\mathbf{u}_t(\gamma)\\ \lim_{t\to\infty}\mathbf{v}_t(\gamma),\end{bmatrix},$$

取

$$\xi_T(\gamma):=-\int_0^T w\mathbf{r}_t(\gamma)\mathrm{d}t,\ \xi_\infty(\gamma):=\lim_{T\to\infty}\xi_T(\gamma),$$

对于

$$\boldsymbol{\nu}(\gamma):=\frac{\xi_\infty(\gamma)}{\log(1/\gamma)}\tag{A2}$$

在 γ 足够小时 (即 $\gamma\to 0$) 使得 $(\mathbf{I}-\mathcal{A}^*(\boldsymbol{\nu}))\cdot\hat{\mathbf{L}}=0,\ \mathbf{I}\succcurlyeq\mathcal{A}^*(\boldsymbol{\nu}),\ \boldsymbol{\nu}\in\lambda\cdot sign(\hat{\mathbf{s}})$ 成立 (取 $\lambda=1/\alpha$), 即:

$$\mathbf{I}\succcurlyeq\lim_{\gamma\to 0}\mathcal{A}^*(\boldsymbol{\nu}(\gamma)),\ \lim_{\gamma\to 0}[\mathbf{I}-\mathcal{A}^*(\boldsymbol{\nu}(\gamma))]\cdot\hat{\mathbf{L}}=0,\ \lim_{\gamma\to 0}\boldsymbol{\nu}(\gamma)\in\alpha^{-1}\cdot sign(\hat{\mathbf{s}}).$$

(a) 接下来先证明: $\mathbf{I}\succcurlyeq\lim_{\gamma\to 0}\mathcal{A}^*(\boldsymbol{\nu}(\gamma)),\ \lim_{\gamma\to 0}[\mathbf{I}-\mathcal{A}^*(\boldsymbol{\nu}(\gamma))]\cdot\hat{\mathbf{L}}=0$.

由链式法则 $\dot{\mathbf{L}}_t(\gamma)=\dot{\mathbf{X}}_t(\gamma)\mathbf{X}_t^\top(\gamma)+\mathbf{X}_t(\gamma)\dot{\mathbf{X}}_t^\top(\gamma)$ 及梯度流更新规则 $\dot{\mathbf{X}}_t(\gamma)=-\mathcal{A}^*(\mathbf{r}_t(\gamma))w\mathbf{X}_t(\gamma)$ 可知:

$$\dot{\mathbf{L}}_t(\gamma)=-\mathcal{A}^*(\mathbf{r}_t(\gamma))w\mathbf{L}_t(\gamma)-w\mathbf{L}_t(\gamma)\mathcal{A}^*(\mathbf{r}_t(\gamma)).$$

由于 $\mathcal{A}$ 可交换, 因此根据梯度流 (A1) 有:

$$\mathbf{L}_t(\gamma)=\exp(\mathcal{A}^*(\xi_t(\gamma)))\mathbf{L}_0(\gamma)\exp(\mathcal{A}^*(\xi_t(\gamma))).$$

因为初始化选择使得 $\mathbf{L}_0(\gamma)=\mathbf{X}_0\mathbf{X}_0^\top=\gamma^2 I$, 故 $\mathbf{L}_\infty(\gamma)=\gamma^2\exp(2\mathcal{A}^*(\xi_\infty(\gamma)))$. 由于假设 $\{A_i\}_{i=1}^N$ 可交换, 因此对于任意 $i=1,\ldots,N$, A_i 可被同一个正交矩阵 U 对角化, 由于 $\mathcal{A}^*(r)=\sum_i r_iA^{(i)}$, 故 $\mathcal{A}^*(\xi_\infty(\gamma))$ 也被 U 对角化, 因此对于任意 $i=1,\ldots,N$, 有:

$$\lambda_k(\mathbf{L}_\infty(\gamma))=\gamma^2\cdot\exp(2\lambda_i(\mathcal{A}^*(\xi_\infty(\gamma))))=\exp(2\lambda_i(\mathcal{A}^*(\xi_\infty(\gamma)))+2\log\gamma),$$

其中, $\lambda_i(\cdot)$ 表示正交矩阵 U 对应的第 i 大的特征值.

同时由于 $\hat{\mathbf{L}}:=\lim_{\gamma\to 0}\mathbf{L}_\infty(\gamma)$, 因此:

$$\lambda_i(\mathbf{L}_\infty(\gamma))\overset{\gamma\to 0}{\to}\lambda_i(\hat{\mathbf{L}}),\quad \forall i=1,\ldots,N.$$

注意到 $\hat{\mathbf{X}}\succcurlyeq 0$, 因此对于 $i=1,\ldots,N$, $\lambda_i(\hat{\mathbf{L}})\geqslant 0$.

(i) 当 $\lambda_i(\hat{\mathbf{L}})>0$:

$$\exp(2\lambda_i(\mathcal{A}^*(\xi_\infty(\gamma)))+2\log\gamma)\to\lambda_i(\hat{\mathbf{L}})\Rightarrow\lambda_i\left(\mathcal{A}^*\left(\frac{\xi_\infty(\gamma)}{\log(1/\gamma)}\right)\right)-1-\frac{\log\lambda_k(\hat{\mathbf{X}})}{2\log(1/\gamma)}\to 0.$$

当 $\boldsymbol{\nu}(\gamma):=\xi_\infty(\gamma)/\log(1/\gamma)$ 时, 可以得到:

$$\lim_{\gamma\to 0}\lambda_i(\mathcal{A}^*(\boldsymbol{\nu}(\gamma)))=1.$$

(ii) 当 $\lambda_i(\hat{\mathbf{L}})=0$:

$$\exp\left(2\lambda_i(\mathcal{A}^*(\xi_\infty(\gamma)))+2\log\gamma\right)\to 0.$$

根据极限定义, 对任意 $\epsilon\in(0,1/2)$, 存在 γ^ϵ, 当 $\gamma<\gamma^\epsilon$ 时, $\exp\left(2\lambda_i(\mathcal{A}^*(\xi_\infty(\gamma)))+2\log\gamma\right)\leqslant\epsilon$, 因此有:

$$\lambda_i\left(\mathcal{A}^*\left(\frac{\xi_\infty(\gamma)}{\log(1/\gamma)}\right)\right)-1<\frac{\log\epsilon}{2\log(1/\gamma)}<0,$$

由此可得到 $\lim_{\gamma\to 0}\lambda_i(\mathcal{A}^*(\boldsymbol{\nu}(\gamma)))<1$.

综合 (i)(ii) 可知, 对于任意 $i=1,\ldots,N$, 有 $\lim_{\gamma\to 0}\lambda_i(\mathcal{A}^*(\boldsymbol{\nu}(\gamma)))\leqslant 1$.

因此当 $\boldsymbol{\nu}(\gamma):=\xi_\infty(\gamma)/\log(1/\gamma)$ 时:

$$I\geqslant\lim_{\gamma\to 0}\mathcal{A}^*(\boldsymbol{\nu}(\gamma)) \tag{A3}$$

另外根据 (i) 和 (ii) 的讨论, 由特征向量构成的对角阵:

$$\Lambda_{\mathcal{A}^*(\boldsymbol{\nu}(\gamma))},\Lambda_{\hat{\mathbf{L}}}\text{满足}\lim_{\gamma\to 0}(I-\Lambda_{\mathcal{A}^*(\boldsymbol{\nu}(\gamma))})\Lambda_{\hat{\mathbf{L}}}=0,$$

因此有:

$$\lim_{\gamma\to 0}U(I-\Lambda_{\mathcal{A}^*(\boldsymbol{\nu}(\gamma))})\Lambda_{\hat{\mathbf{L}}}U^\top=\lim_{\gamma\to 0}(I-\mathcal{A}^*(\boldsymbol{\nu}(\gamma)))\cdot\hat{\mathbf{L}}=0 \tag{A4}$$

(b) 接下来仅需再证明: $\lim_{\gamma\to 0}\boldsymbol{\nu}(\gamma)\in\alpha^{-1}\cdot sign(\hat{\mathbf{s}})$.

根据梯度流 (A1) 有:

$$\mathbf{u}_t(\gamma)=\mathbf{u}_0(\gamma)^\circ\exp(\alpha/w\xi_t(\gamma)),\ \mathbf{v}_t(\gamma)=\mathbf{v}_0(\gamma)^\circ\exp(-\alpha/w\xi_t(\gamma)) \tag{A5}$$

由于 $\mathbf{s}:=\mathbf{u}^\circ\mathbf{u}-\mathbf{v}^\circ\mathbf{v}$, 因此:

$$\mathbf{s}_\infty^i(\gamma)=\gamma^2\cdot\exp\left(2\alpha\cdot\xi_\infty^i(\gamma)\right)-\gamma^2\cdot\exp\left(-2\alpha\cdot\xi_\infty^i(\gamma)\right),$$

其中, $\mathbf{s}_\infty^i(\gamma)$ 和 $\xi_\infty^i(\gamma)$ 分别是 $\mathbf{s}_\infty(\gamma)$ 和 $\xi_\infty(\gamma)$ 的第 i 个元素.

令 $\hat{\mathbf{s}}:=\lim_{\gamma\to 0}\mathbf{s}_\infty(\gamma)$ 并且 $\hat{\mathbf{s}}_i=\lim_{\gamma\to 0}\mathbf{s}_\infty^i(\gamma)$. 接下来证明当 $\boldsymbol{\nu}(\gamma)=\xi_\infty(\gamma)/\log(1/\gamma)$ 时, $\lim_{\gamma\to 0}\nu_i(\gamma)=1/\alpha sign(\hat{\mathbf{s}}_i)$.

根据 $\hat{\mathbf{s}}_i>0$, $\hat{\mathbf{s}}_i<0$ 及 $\hat{\mathbf{s}}_i=0$ 分 3 种情况讨论.

(i) 当 $\hat{\mathbf{s}}_i>0$ 时, 由公式 (A5) 得知一定有 $\lim_{\gamma\to 0}\xi_\infty(\gamma)=+\infty$ 使得 $\exp(2\alpha\xi_\infty(\gamma))\to+\infty$, $\exp(2\alpha\xi_\infty(\gamma))\to 0$.

此时:

$$2\alpha\cdot\xi_\infty^i(\gamma)-2\log(1/\gamma)\overset{\gamma\to 0}{\to}\log\hat{\mathbf{s}}_i\Rightarrow\lim_{\gamma\to 0}\boldsymbol{\nu}_i(\gamma)=1/\alpha.$$

(ii) 当 $\hat{\mathbf{s}}_i<0$ 时, 类似 (i) 此时 $\lim_{\gamma\to 0}\xi_\infty(\gamma)=-\infty$ 使得 $\exp(2\alpha\xi_\infty(\gamma))\to 0$, $\exp(2\alpha\xi_\infty(\gamma))\to+\infty$, 于是:

$$-2\alpha\cdot\xi_\infty^i(\gamma)+2\log(1/\gamma)\overset{\gamma\to 0}{\to}\log\hat{\mathbf{s}}_i\Rightarrow\lim_{\gamma\to 0}\boldsymbol{\nu}_i(\gamma)=-1/\alpha.$$

(iii) 当 $\hat{\mathbf{s}}_i=0$ 时, 由公式 (A5) 得知当 $\gamma\to 0$ 此时一定有 $\gamma^2\exp(2\alpha\xi_\infty(\gamma))\to 0$ 且 $\gamma^2\exp(-2\alpha\xi_\infty(\gamma))\to 0$. 根据极限定义, 对任意 $\epsilon\in(0,1/2)$, 存在 γ^ϵ, 当 $\gamma<\gamma^\epsilon$ 时:

$$\gamma^2\cdot\max\left\{\exp\left(2\alpha\cdot\xi_\infty^i(\gamma)\right),\exp\left(-2\alpha\cdot\xi_\infty^i(\gamma)\right)\right\}\leqslant\epsilon\Rightarrow 2\alpha\cdot\left|\frac{\xi_\infty^i(\gamma)}{\log(1/\gamma)}\right|-2<\frac{\log\epsilon}{\log(1/\gamma)}<0\Rightarrow\lim_{\gamma\to 0}|\boldsymbol{\nu}_i(\gamma)|<1/\alpha.$$

综合 (i)(ii)(iii) 可知 $\lim_{\gamma\to 0}\boldsymbol{\nu}(\gamma)\in\alpha^{-1}\cdot sign(\hat{\mathbf{s}})$.

综合 (a)(b) 可知, 当 $\boldsymbol{\nu}(\gamma):=\xi_\infty(\gamma)/\log(1/\gamma)$ 时:

$$\mathbf{I}\geqslant\lim_{\gamma\to 0}\mathcal{A}^*(\boldsymbol{\nu}(\gamma)),\quad\lim_{\gamma\to 0}[\mathbf{I}-\mathcal{A}^*(\boldsymbol{\nu}(\gamma))]\cdot\hat{\mathbf{L}}=0,\quad\lim_{\gamma\to 0}\boldsymbol{\nu}(\gamma)\in\alpha^{-1}\cdot sign(\hat{\mathbf{s}}),\ \mathcal{A}(\hat{\mathbf{L}})+\hat{\mathbf{s}}=\mathbf{y},\quad\hat{\mathbf{L}}\geqslant 0,$$

满足 KKT 条件, 此时 $(\hat{\mathbf{L}},\hat{\mathbf{s}})=(\hat{\mathbf{X}}\hat{\mathbf{X}}^\top,\hat{\mathbf{u}}^\circ\hat{\mathbf{u}}-\hat{\mathbf{v}}^\circ\hat{\mathbf{v}})$ 即为问题式 (14) 的最优解.

初旭(1992－), 男, 博士, 助理研究员, CCF 专业会员, 主要研究领域为机器学习, 数据分析.

王亚沙(1975－) , 男, 博士, 教授, 博士生导师, CCF 杰出会员, 主要研究领域为机器学习, 数据分析, 普适计算.

马辛宇(1999－), 男, 博士生, CCF 学生会员, 主要研究领域为时间序列分析, 图数据分析.

朱文武(1963－), 男, 博士, 教授, 博士生导师, CCF 会士, 主要研究领域为多媒体大数据, 机器学习.

林阳(1998－), 男, 博士生, CCF 学生会员, 主要研究领域为数据挖掘, 自然语言处理.

梅宏(1963－), 男, 博士, 教授, 博士生导师, CCF 会士, 主要研究领域为软件工程, 系统软件, 大数据分析.

王鑫(1988－), 男, 博士, 助理研究员, CCF 高级会员, 主要研究领域为多媒体智能, 媒体大数据, 机器学习.