图神经网络（Graph Neural Network）学习路线梳理

2020.7

参考系列文章：

[从图(Graph)到图卷积(Graph Convolution)：漫谈图神经网络模型 (一)]

[从图(Graph)到图卷积(Graph Convolution)：漫谈图神经网络模型 (二)]

[从图(Graph)到图卷积(Graph Convolution)：漫谈图神经网络模型 (三)]

1. 相关Survey：

- *A Comprehensive Survey on Graph Neural Networks* https://arxiv.org/abs/1901.00596
- *Deep Learning on Graphs: A Survey* https://arxiv.org/abs/1812.04202
- *Graph Neural Networks: A Review of Methods and Applications* https://arxiv.org/pdf/1812.08434

2. 历史脉络

1. 图神经网络的概念最早在2005年提出。2009年Franco博士在其论文 *The graph neural network model*, https://persagen.com/files/misc/scarselli2009graph.pdf 中定义了图神经网络的理论基础，第一种图神经网络（GNN）也是基于这篇论文。
2. 最早的GNN主要解决的还是如分子结构分类等严格意义上的图论问题。但实际上欧式空间(比如像图像 Image)或者是序列(比如像文本 Text)，许多常见场景也都可以转换成图(Graph)，然后就能使用图神经网络技术来建模。
3. 2009年后图神经网络也陆续有一些相关研究，但没有太大波澜。直到2013年，在图信号处理(Graph Signal Processing)的基础上，Bruna(这位是LeCun的学生)在文献 *Spectral networks and locally connected networks on graphs*, https://arxiv.org/abs/1312.6203中首次提出图上的基于频域(Spectral-domain)和基于空域(Spatial-domain)的卷积神经网络。
4. 其后至今，学界提出了很多基于空域的图卷积方式，也有不少学者试图通过统一的框架将前人的工作统一起来。而基于频域的工作相对较少，只受到部分学者的青睐。
5. 值得一提的是，图神经网络与图表示学习(Represent Learning for Graph)的发展历程也惊人地相似。2014年，在word2vec：*Distributed Representations of Words and Phrases and their Compositionality*, http://papers.nips.cc/paper/5

021-distributed-representations-of-words-andphrases的启发下，Perozzi等人提出了DeepWalk：*DeepWalk: Online Learning of Social Representations*, https://arxiv.org/abs/1403.6652，开启了深度学习时代图表示学习的大门。更有趣的是，就在几乎一样的时间，Bordes等人提出了大名鼎鼎的TransE：*Translating Embeddings for Modeling Multi-relational Data*, https://papers.nips.cc/paper/5071-translating-embeddings-for-modeling-multi-relational-data，为知识图谱的分布式表示(Represent Learning for Knowledge Graph)奠定了基础

2. 图神经网络（Graph Neural Network）

最早的图神经网络起源于Franco博士的论文：*The graph neural network model*, https://persagen.com/files/misc/scarselli2009graph.pdf

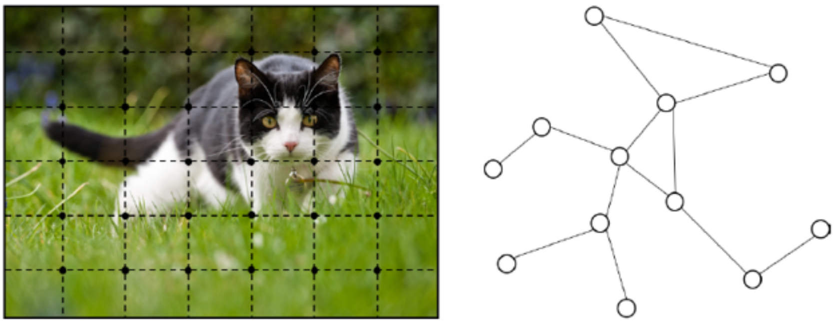


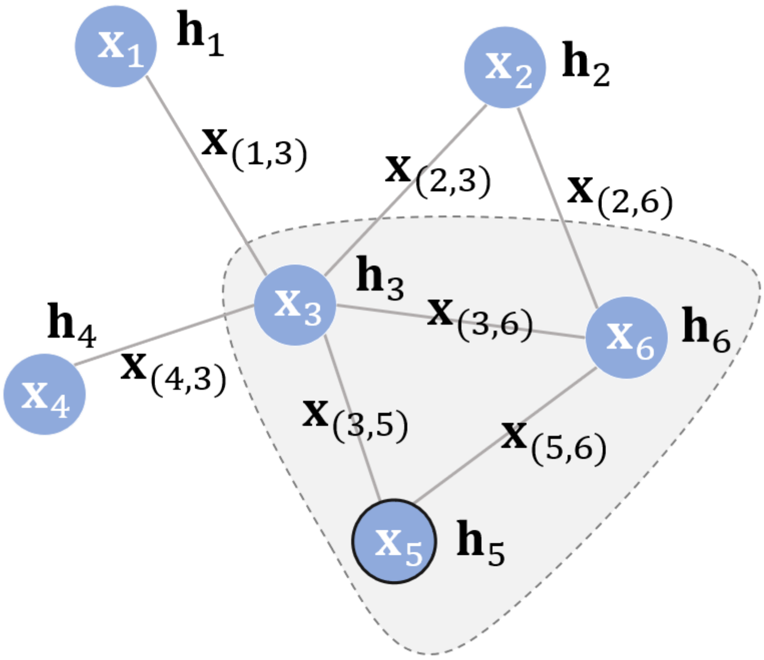
Fig. 1. Left: image in Euclidean space. Right: graph in non-Euclidean space

它的理论基础是**不动点**理论。给定一张图 GG，每个结点都有其自己的特征(feature), 本文中用x_v表示结点v的特征；连接两个结点的边也有自己的特征，本文中用$x(v,u)$表示结点v与结点u之间边的特征；GNN的学习目标是获得每个结点的图感知的隐藏状态h_v(state embedding)，这就意味着：对于每个节点，它的隐藏状态包含了来自邻居节点的信息。那么，如何让每个结点都感知到图上其他的结点呢？GNN通过**迭代式更新**所有结点的隐藏状态来实现，在$t+1$时刻，结点vv的隐藏状态按照如下方式更新：

$$h_v^{t+1} = f(x_v, x_co[v], h_n^te[v], x_ne[v])$$

上面这个公式中的 ff 就是隐藏状态的**状态更新**函数，在论文中也被称为**局部转移函数**(local transaction function)。公式中的xco[v]$\mathbf{x}co[v]$指的是与结点vv相邻的边的特征，xne[v]$\mathbf{x}ne[v]$指的是结点vv的邻居结点的特征，htne[v]$\mathbf{h}nte[v]$则指邻居结点在tt时刻的隐藏状态。注意 ff 是对所有结点都成立的，是一个全局共享的函数。

与深度学习（机器学习）相结合，利用神经网络(Neural Network)来拟合这个复杂函数 ff。值得一提的是，虽然看起来 ff 的输入是不定长参数，但在 ff 内部我们可以先将不定长的参数通过一定操作变成一个固定的参数，比如说用所有隐藏状态的加和来代表所有隐藏状态。

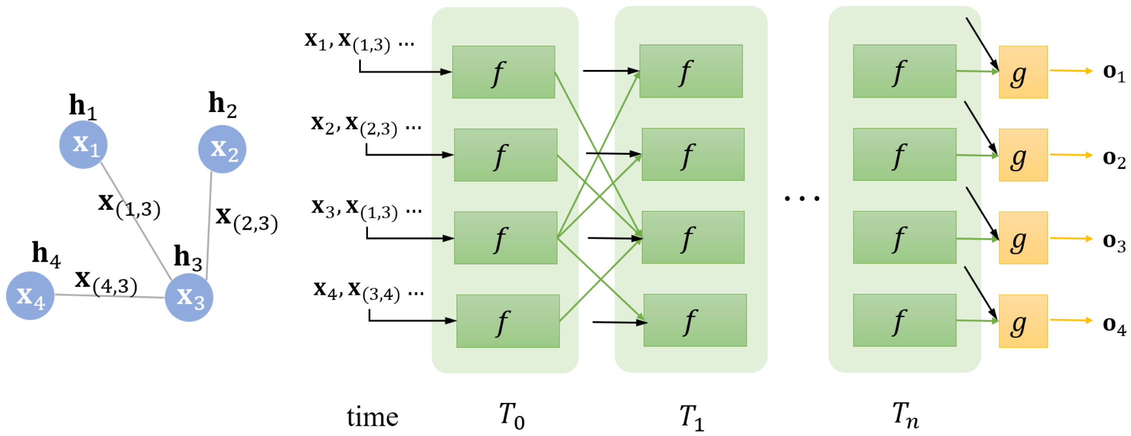


$$\mathbf{h}_5 = f\left(\mathbf{x}_5, \mathbf{x}_{(3,5)}, \mathbf{x}_{(5,6)}, \mathbf{h}_3, \mathbf{h}_6, \mathbf{x}_3, \mathbf{x}_6\right)$$

状态更新公式仅描述了如何获取每个结点的隐藏状态，除它以外，我们还需要另外一个函数 g来描述如何适应下游任务。

$$\mathbf{o}_v = g(\mathbf{h}_v, \mathbf{x}_v)$$

在原论文中，g又被称为**局部输出函数**(local output function)，与 f类似，g 也可以由一个神经网络来表达，它也是一个全局共享的函数。那么，整个流程可以用下面这张图表达：



对于不同的图来说，收敛的时刻可能不同，因为收敛是通过两个时刻p-范数的差值是否小于某个阈值 ϵ来判定的，比如：

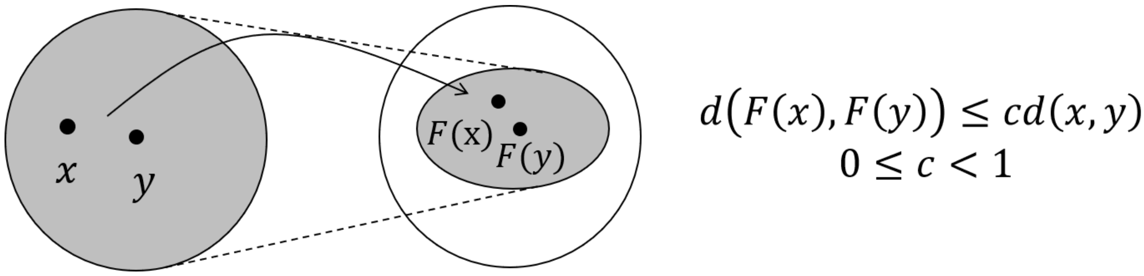
$$||\mathbf{H}^{t+1}||_2 - ||\mathbf{H}^t||_2 < \epsilon$$

Banach's Fixed Point Theorem:

GNN的理论基础是**不动点**(the fixed point)理论，这里的不动点理论专指**巴拿赫不动点定理**(Banach's Fixed Point Theorem)。首先我们用 F表示若干个 f 堆叠得到的一个函数，也称为**全局更新**函数，那么图上所有结点的状态更新公式可以写成:

$$\mathbf{H}^{t+1} = F(\mathbf{H}^t, \mathbf{X})$$

不动点定理指的就是，不论H^0是什么，只要 F是个**压缩映射**(contraction map)，H^0经过不断迭代都会收敛到某一个固定的点，我们称之为不动点。那压缩映射又是什么呢，一张图可以解释得明明白白：



上图的实线箭头就是指映射F, 任意两个点 x，y在经过F这个映射后，分别变成了 $F(x), F(y)$。压缩映射就是指，$d(F(x), F(y)) \leq cd(x, y), 0 \leq c < 1$。也就是说，经过 F 变换后的新空间一定比原先的空间要小，原先的空间被压缩了。想象这种压缩的过程不断进行，最终就会把原空间中的所有点映射到一个点上。

具体实现：

在具体实现中， F其实通过一个简单的**前馈神经网络**(Feed-forward Neural Network)即可实现。

$$\begin{aligned}\mathbf{h}_v^{t+1} &= f(\mathbf{x}_v, \mathbf{x}_c o[v], \mathbf{h}_n^t e[v], \mathbf{x}_n e[v]) \\ &= \sum_{u \in ne[v]} FNN([\mathbf{x}_v; \mathbf{x}_{(u,v)}; \mathbf{h}_u^t; \mathbf{x}_u])\end{aligned}$$

那我们如何保证 f是个压缩映射呢，其实是通过限制 f 对H的偏导数矩阵的大小，这是通过一个对**雅可比矩阵**(Jacobian Matrix)的**惩罚项**(Penalty)来实现的。在代数中，有一个定理是: f 为压缩映射的等价条件是 f的梯度/导数要小于1。这个等价定理可以从压缩映射的形式化定义导出，我们这里使用 $||x||$表示 x 在空间中的**范数**(norm)。范数是一个标量，它是向量的长度或者模，$||x||$ 是 x在有限空间中坐标的连续函数。这里把 x 简化成1维的，坐标之间的差值可以看作向量在空间中的距离，根据压缩映射的定义，可以导出：

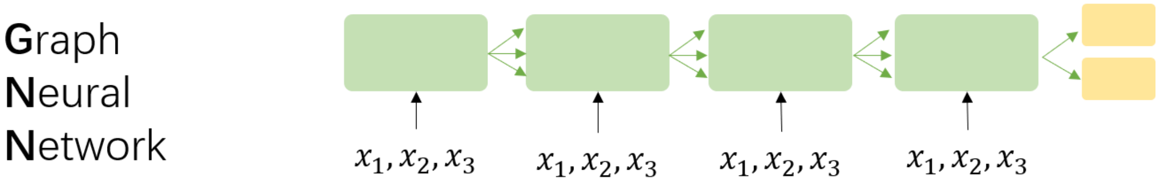
$$||F(x)-F(y)||\leq c||x-y||, 0\leq c<1$$
$$\frac{||F(x)-F(y)||}{||x-y||}\leq c$$
$$\frac{||F(x)-F(x-\Delta x)||}{||\Delta x||}\leq c$$
$$||F'(x)|| = ||\frac{\partial F(x)}{\partial x}||\leq c$$

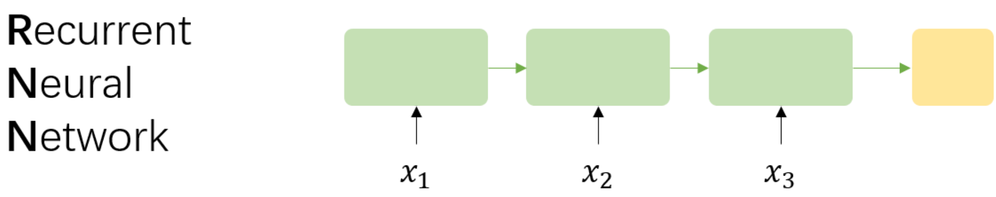
推广一下，即得到雅可比矩阵的罚项需要满足其范数小于等于c等价于压缩映射的条件。根据拉格朗日乘子法，将有约束问题变成带罚项的无约束优化问题，训练的目标可表示成如下形式：

$$J = Loss + \lambda \cdot \max(\frac{||\partial FNN||}{||\partial \mathbf{h}||} - c, 0), c\in(0,1)$$

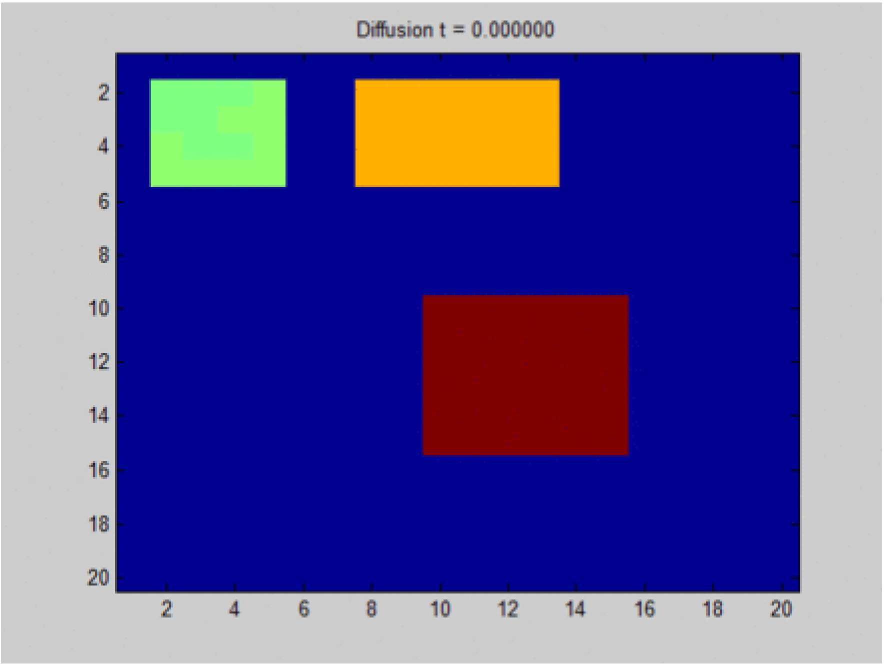
其中λ是超参数，与其相乘的项即为雅可比矩阵的罚项。

- 模型学习
- GNN与RNN





- GNN局限性：



3. 门控神经网络（Gated Graph Neural Network）

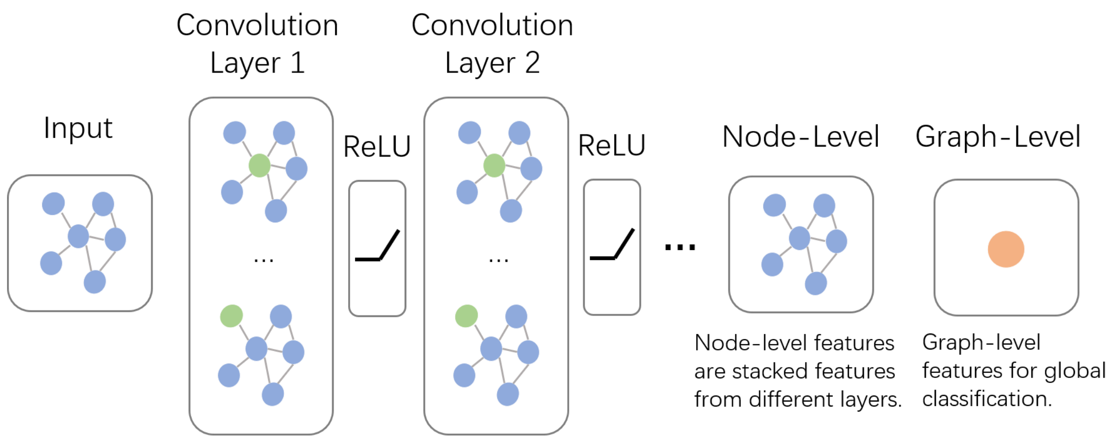
门控图神经网络(Gated Graph Neural Network, GGNN) *Gated graph sequence neural networks*, https://arxiv.org/abs/1511.05493就出现了。虽然在这里它们看起来类似，但实际上，它们的区别非常大，其中最核心的不同即是**门控神经网络不以不动点理论为基础**。这意味着：f不再需要是一个压缩映射；迭代不需要到收敛才能输出，可以迭代固定步长；优化算法也从 AP 算法转向 BPTT。

- 状态更新
- GNN与GCNN

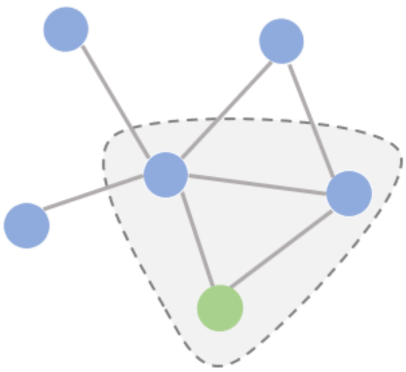
4. 图卷积

在本篇中，我们将着大量笔墨介绍**图卷积神经网络中的卷积操作**。接下来，我们将首先介绍一下图卷积神经网络的大概框架，借此说明它与基于循环的图神经网络的区别。接着，我们将从头开始为读者介绍卷积的基本概念，以及其在物理模型中的涵义。最后，我们将详细地介绍两种不同的卷积操作，分别为**空域卷积**和**频域卷积**，与其对应的经典模型。读者不需有任何信号处理方面的基础，傅里叶变换等概念都会在本文中详细介绍。

- 图卷积：使用于图的可学习卷积核
- 图卷积框架（Framework）

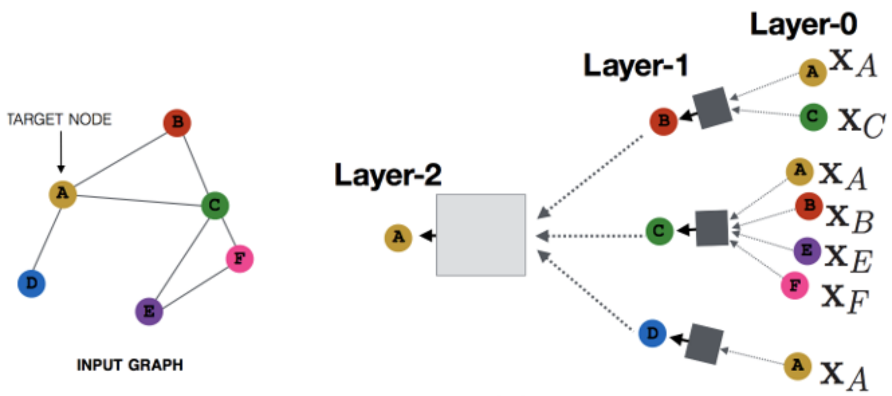


- 空域卷积（Spatial Convolution)



$$\mathbf{h}_v^{l+1} = \sum_{u \in ne[v]} \mathbf{h}_u^l$$

消息传递网络(Message Passing Neural Network)



图采样与聚合(Graph Sample and Aggregate)

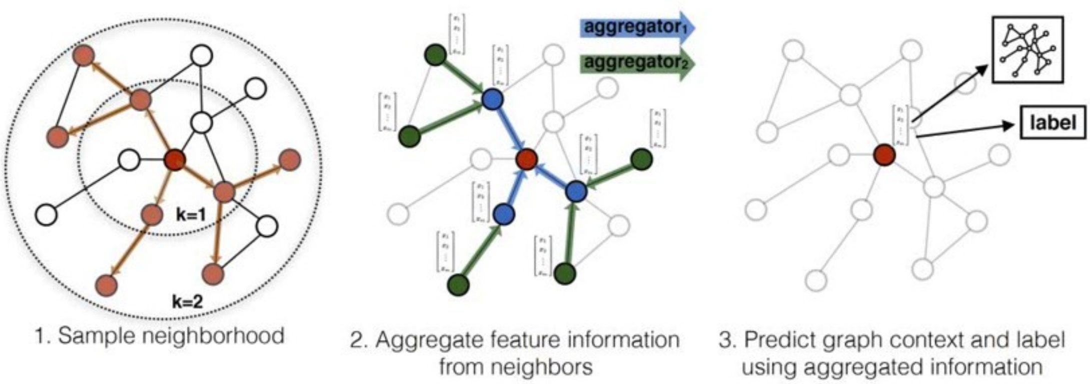
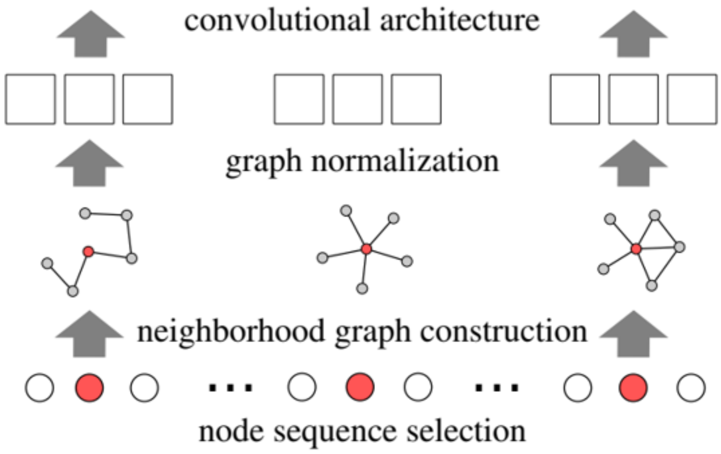


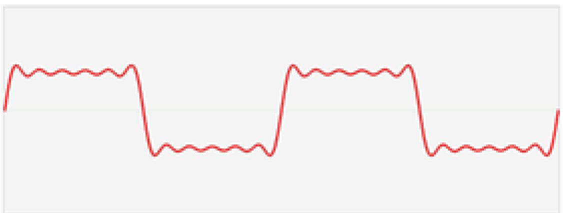
Figure 1: Visual illustration of the GraphSAGE sample and aggregate approach.

图结构序列化(PATCHY-SAN)

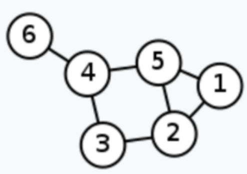


- 频域卷积（Spectral Convolution）

傅里叶变换(Fourier Transform)

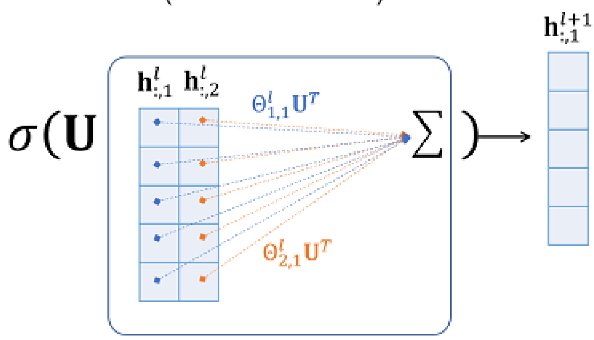


图上的傅里叶变换

Labeled graph	Degree matrix	Adjacency matrix	Laplacian matrix
	$\begin{pmatrix} 2&0&0&0&0&0\\0&3&0&0&0&0\\0&0&2&0&0&0\\0&0&0&3&0&0\\0&0&0&0&3&0\\0&0&0&0&0&1 \end{pmatrix}$	$\begin{pmatrix} 0&1&0&0&1&0\\1&0&1&0&1&0\\0&1&0&1&0&0\\0&0&1&0&1&1\\1&1&0&1&0&0\\0&0&0&1&0&0 \end{pmatrix}$	$\begin{pmatrix} 2&-1&0&0&-1&0\\-1&3&-1&0&-1&0\\0&-1&2&-1&0&0\\0&0&-1&3&-1&-1\\-1&-1&0&-1&3&0\\0&0&0&-1&0&1 \end{pmatrix}$

频域卷积网络(Spectral CNN)

$$\mathbf{h}_{:,j}^{l+1} = \sigma\left(\mathbf{U}\sum_{i=1}^{d_l}\Theta_{i,j}^{l}\mathbf{U}^T\mathbf{h}_{:,i}^{l}\right), (j = 1 \ldots d_{l+1})$$



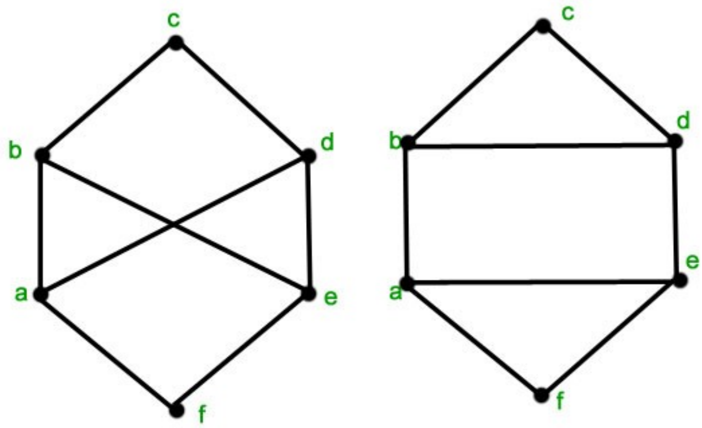
切比雪夫网络(ChebNet)

5. 图表示

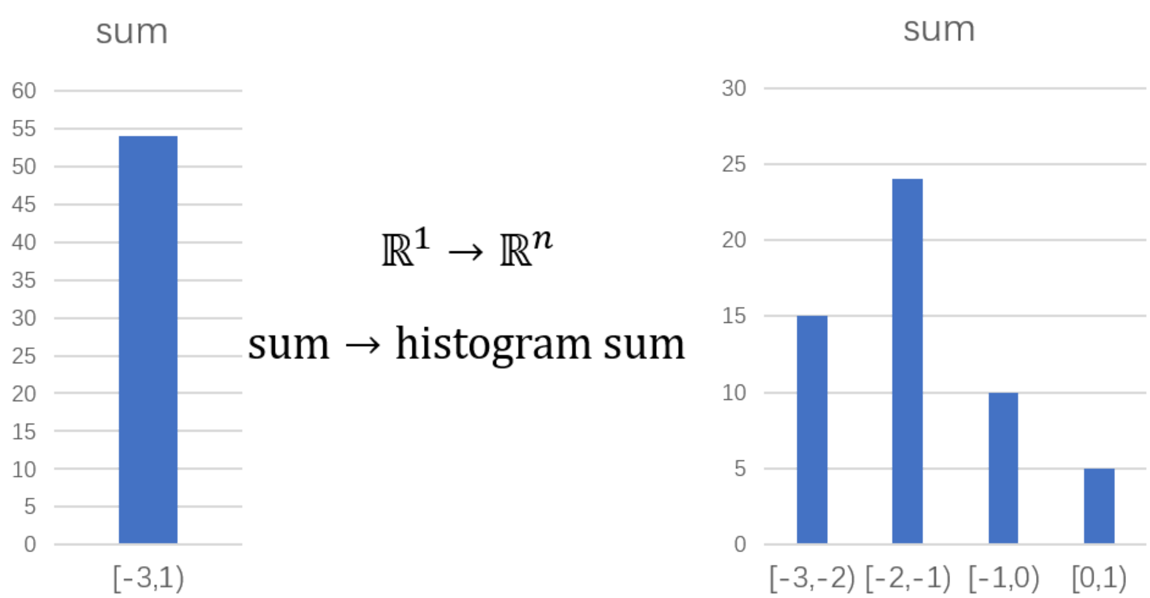
该部分主要关注在**得到了各个结点的表示后，如何生成整个图的表示**

- 图读出操作(ReadOut)

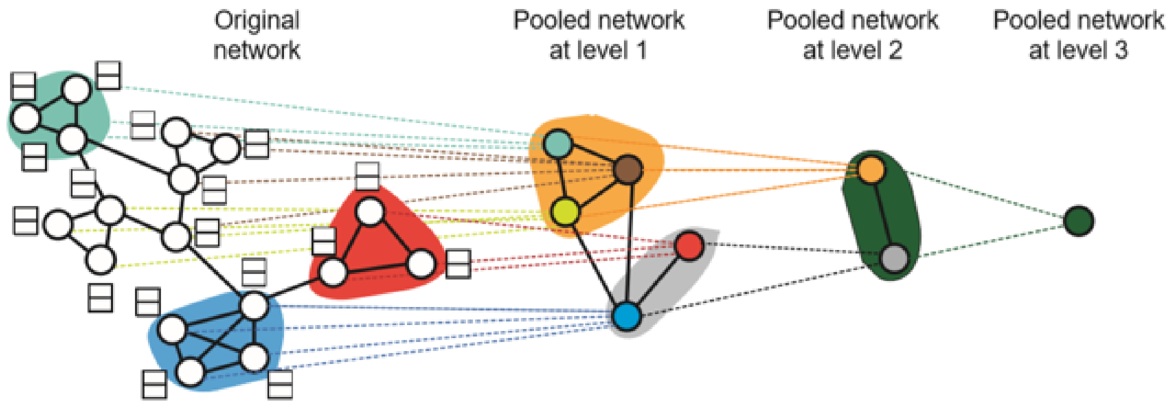
图重构(Graph Isomorphism)



基于统计的方法(Statistics Category)



基于学习的方法(Learning Category)：采样加全连接(Sample And FC)、全局结点(Global Node)、可微池化(Differentiable Pooling)



- DiffPool

同时完成了两个任务：结点聚类(Soft Clustering)与结点表示(Node Representation)。这两个任务是由两个不共享参数的GCN模块分别完成的，下文用 SC 和 NR 分别表示这两个模块。NR 模块与传统的GCN一样，输入是各结点的隐藏状态，通过图上的传播，输出是传播后各个结点的表示。SC 模块则不同，它的输入虽然也是各结点的隐藏表示，但其输出的是各结点属于不同聚类簇的概率(注意这里每一层聚类簇的数目是预先定义的)。上图中最左侧每个结点右上方的表格即代表这个。

0	1	0
1	0	0
0	0	1
1	0	0
0	1	0
0	1	0

$$\boldsymbol{S}^l \in \mathbb{R}^{6\times 3}$$