

# DS-GA 1008 Midterm HW

Long Chen  
Center for Data Science, New York University  
lc3424@nyu.edu

Fall 2021

## 1 Theory

### 1.1 GDL Blueprint and Compositional Architectures

1.

Let  $H, W$  the height and width of image. From description:

$$\begin{aligned}\Omega_1 &\in \mathbb{R}^{C_1 \times H \times W} \\ \Omega_2 &\in \mathbb{R}^{C_2 \times H//2 \times W//2} \\ \Omega_3 &\in \mathbb{R}^{C_3}\end{aligned}$$

By setting,  $\varphi_1(x)$  is a convolution with some filter  $F$  where the kernel size is  $x \times 2$  and stride of  $2 \times 2$ . With the definition:

$$\begin{aligned}\varphi_1(x) &: \mathcal{X}(\Omega_1, \mathcal{C}_1) \mapsto \mathcal{X}(\Omega_2, \mathcal{C}_2) \\ \varphi_2(x) &: \mathcal{X}(\Omega_2, \mathcal{C}_2) \mapsto \mathcal{X}(\Omega_3, \mathcal{C}_3)\end{aligned}$$

for  $i \in \{1, \dots, H//2\}$  and  $j \in \{1, \dots, W//2\}$ , we have:

$$\begin{aligned}\varphi_1(x)(\tilde{u}_{i,j}) &= x[2i, 2j] + x[2i+1, 2j] + x[2i, 2j+1] + x[2i+1, 2j+1] \\ \varphi_2(x)(\tilde{u}) &= f(\{\varphi_1(x)(\tilde{u}_{i,j})\})\end{aligned}$$

where  $f : \Omega_2 \mapsto \Omega_3$ . The description concludes  $\Phi(x) = \varphi_2 \circ \varphi_1(x)$ .

2.

Any dataset *s.t* for  $\forall (x_i, y_1), (x_j, y_2), y_1 \neq y_2, \varphi_1(x_i) = \varphi_1(x_j)$ , the input for  $\varphi_2$  will be the same and, since  $\varphi_2$  must be a one-to-one function, the output of  $\Phi(x) = \varphi_2 \circ \varphi_1(x)$  should be same for the two classes. Consider an example where class 1 has alternating value of 1, -1 for each row and class 2 has all zero entries. The average pooling  $\varphi_1$  will yield all zero entries as output and thus the two classes will not be separable.

3.

Similarly, if  $\varphi_2 \circ \varphi_1(x_i) = \varphi_2 \circ \varphi_1(x_j)$ ,  $\forall i, j$ , the two classes will not be separable. Consider a case where  $H = W$ . Consider where class 1 images where only two consecutive columns have value 1's and else 0's (images containing a single *vertical* line with width 2), and where class 2 an image where only consecutive rows have has value 1's and else 0's (images containing a single *horizontal* line with width 2).

Consider that  $\varphi_2$  is global average pooling. Easily can we verify that  $\Phi(x_i)$  is a constant for all  $x_i$  in the same class.

Intuitively, the output of  $\varphi_1(x_i)$  of one class should be a 90-degree rotation of the  $\varphi_1(x_j)$  counterpart, where  $x_j$  is a 90-degree rotation of  $x_i$ . Therefore, all  $\Phi(x_i)$  is a constant for all  $x_i$  regardless of its class, since  $\varphi_2$  is a global average pooling function.

## 1.2 Invariant Polynomials

1.

Since the wording of this question is not clear (as discussed on Campuswire), here I provide solutions to both cases:

**Case 1 (invariant to both groups):** A candidate choice would be:

$$p(x_1, \dots, x_d) = \sum_{k=0}^K x_k.$$

**Case 2 (NOT invariant to  $\mathcal{C}_d$  or  $\mathcal{S}_d$ ):** A candidate choice would be:

$$p(x_1, \dots, x_d) = \sum_{k=0}^K a_k x_k,$$

where  $a_k = k+1$ . In such case, if we have rotation or permutation, the coefficient for  $x_i$  will be different for at least (2)  $x_i$ .

2.

A candidate choice would be:

$$p(x_1, \dots, x_d) = \sum_{k=0}^K x_{ik} x_{ik+1 \bmod K}.$$

That is the sum of product of two adjacent  $x_i$  with start-to-end connection and thus the cycled input will yield same output. However for permuted input, since the order of the input is changed, there will be at least (1)  $x_i$  with different neighbor and thus yielding different output.

3.

This will not be possible. By definition,  $\mathcal{C}_d \subset \mathcal{S}_d$ . It's not possible to find a polynomial function that is invariant to  $\mathcal{S}_d$  while is *not* invariant to its subset.

## 2 Coding Question

1.

With SBM, it's possible that our prediction “flips” the class prediction as the model is unsupervised. Such situation will reflect on a worse score on our current loss function, while the prediction is totally fine in separating the two classes.

2.

This setting of loss function will eliminate the effect of “permuted label” and yields a stable score even if our predicted label is permuted (or flipped in our binary case).

3.

Since we do not use node values in our model, the only thing to differentiate a community are the edges. Intuitively, the model catches connection (or the amount of connections in our unweighted edge case) among nodes. If we have  $p = q$ , the expected number of connections between all nodes will be the same, and thus the model will not have any feature to detect communities.

4.

Code attached in companion .py file. See the following two types of visualizations of communities:

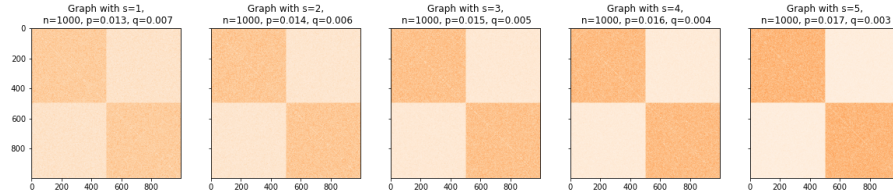


Figure 1: Heatmap of the sum of adjacency matrices for 100 randomly generated networks of different values of  $s$ . Note that with larger value of  $s$ , there are more intra-community connections and less inter-community connections.

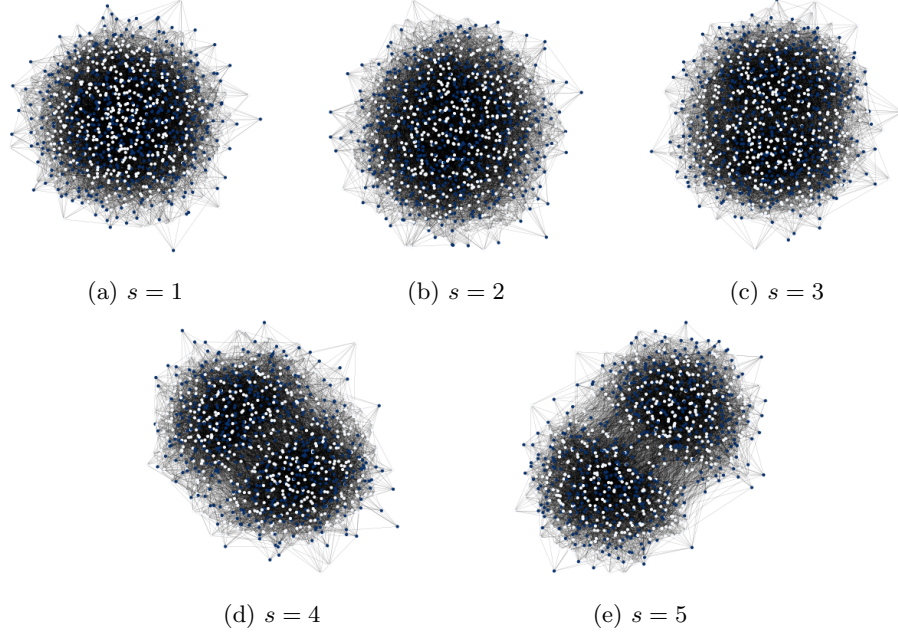


Figure 2: Network visualization of different values of  $s$  with  $n=1000$ . Blue and white nodes indicate two classes. Note that with larger value of  $s$ , we could have clearer community clusters.

## 5.

Code attached in companion .py file.

Let  $a = \frac{1}{\|V\|} \sum_i \mathbf{1}[\text{sign}(\hat{y}_i) = y_i]$ ,  $b = \frac{1}{\|V\|} \sum_i \mathbf{1}[\text{sign}(\hat{y}_i) = -y_i]$ . Intuitively,  $a + b = 1$ , so:

$$\max \max(a, b) = 1$$

$$\min \max(a, b) = 0.5$$

Therefore, we verify  $0 \leq \text{Overlap} \leq 1$ .

When overlap is 0, it means that model did no more than random guessing, while when overlap is 1, the model identifies the cluster perfectly.

## 6.

Code attached in companion .py file.

**7.**

Code attached in companion .py file.

Average overlap score for the sample graphs using spectral is: 0.0772.

Average overlap score for the sample graphs using SBM is: 0.286.