

Машинное обучение, ФКН ВШЭ

Семинар №18

1 Лапласиан графа

Пусть задан некоторый взвешенный неориентированный граф $G = (V, E)$, где $V = \{v_1, \dots, v_n\}$ – множество вершин. Мы также будем обозначать вершину v_i индексом i . Матрицу смежности обозначим $W = (w_{ij})_{i,j=1,\dots,n}$. Степенью вершины v_i назовём

$$d_i = \sum_{j=1}^n w_{ij}$$

Также введём матрицу D как диагональную матрицу с элементами $D_{ii} = d_i$. Ненормированным Лапласианом графа называется матрица

$$L = D - W$$

Лапласиан графа обладает следующими свойствами:

1. Для произвольного вектора $f \in \mathbb{R}^n$

$$f^T L f = \frac{1}{2} \sum_{i,j} w_{ij} (f_i - f_j)^2$$

Докажем это.

$$\begin{aligned} f^T L f &= f^T D f - f^T W f = \sum_{i=1}^n d_i f_i^2 - \sum_{i,j} f_i f_j w_{ij} = \\ &= \frac{1}{2} \left(\sum_{i=1}^n d_i f_i^2 - 2 \sum_{i,j} f_i f_j w_{ij} + \sum_{j=1}^n d_j f_j^2 \right) = \frac{1}{2} \left(\sum_{i=1}^n \sum_{j=1}^n w_{ij} f_i^2 - 2 \sum_{i,j} f_i f_j w_{ij} + \sum_{j=1}^n \sum_{i=1}^n w_{ij} f_j^2 \right) = \\ &= \frac{1}{2} \sum_{i,j} w_{ij} (f_i - f_j)^2 \end{aligned}$$

2. L – симметричная, положительно полуопределённая матрица.

Симметричность следует из того, что граф неориентированный. Положительная полуопределённость следует из того, что все веса неотрицательные $w_{ij} \geq 0 \forall i, j$, а также из предыдущего пункта:

$$\forall f \in \mathbb{R}^n \quad f^T L f \geq 0.$$

3. Наименьшее собственное значение равно 0 и соответствует вектору $\mathbf{1} = (1, \dots, 1)^T$.

2 Graph-cut подход

Пусть мы задали на нашем множестве точек взвешенный неориентированный граф, где вес ребра w_{ij} означает степень сходства объектов v_i, v_j (чем больше w_{ij} , тем больше похожи объекты v_i, v_j). Введём обозначения: $W(A, B) = \sum_{i \in A, j \in B} w_{ij}$, \bar{A} – дополнение множества A . Тогда логично решать задачу кластеризации выборки на k непересекающихся кластеров A_1, \dots, A_k , минимизируя функционал

$$\text{cut}(A_1, \dots, A_k) = \frac{1}{2} \sum_{i=1}^k W(A_i, \bar{A}_i). \quad (2.1)$$

На практике минимизация функционала 2.1 зачастую приводит к тому, что в кластера выделяются отдельные объекты, что является нежелательным. Поэтому данный функционал нормируют следующим образом:

$$\text{RatioCut}(A_1, \dots, A_k) = \frac{1}{2} \sum_{i=1}^k \frac{W(A_i, \bar{A}_i)}{|A_i|}, \quad (2.2)$$

где $|A|$ – количество вершин во множестве A .

Рассмотрим оптимизационную задачу для случая, когда $k = 2$:

$$\min_{A \subset V} \text{RatioCut}(A, \bar{A}).$$

Переформулируем её через Лапласиан. Для этого введём вектор $f = (f_1, \dots, f_n)^T$.

$$f_i = \begin{cases} \sqrt{|\bar{A}|/|A|}, & v_i \in A \\ -\sqrt{|A|/|\bar{A}|}, & v_i \in \bar{A} \end{cases}$$

И запишем

$$\begin{aligned} f^T L f &= \frac{1}{2} \sum_{i,j} w_{ij} (f_i - f_j)^2 \\ &= \frac{1}{2} \sum_{i \in A, j \in \bar{A}} w_{ij} \left(\sqrt{\frac{|\bar{A}|}{|A|}} + \sqrt{\frac{|A|}{|\bar{A}|}} \right)^2 + \frac{1}{2} \sum_{i \in \bar{A}, j \in A} w_{ij} \left(-\sqrt{\frac{|A|}{|\bar{A}|}} - \sqrt{\frac{|\bar{A}|}{|A|}} \right)^2 \\ &= \frac{1}{2} \left(\sqrt{\frac{|\bar{A}|}{|A|}} + \sqrt{\frac{|A|}{|\bar{A}|}} \right)^2 (W(A, \bar{A}) + W(\bar{A}, A)) \\ &= \frac{1}{2} \left(\frac{|\bar{A}|}{|A|} + 2 + \frac{|A|}{|\bar{A}|} \right) (W(A, \bar{A}) + W(\bar{A}, A)) \\ &= \frac{1}{2} \left(\frac{|\bar{A}| + |A|}{|A|} + \frac{|A| + |\bar{A}|}{|\bar{A}|} \right) (W(A, \bar{A}) + W(\bar{A}, A)) \\ &= n \text{RatioCut}(A, \bar{A}) \end{aligned}$$

Также заметим, что

$$\sum_{i=1}^n f_i = |A| \sqrt{\frac{|\bar{A}|}{|A|}} - |\bar{A}| \sqrt{\frac{|A|}{|\bar{A}|}} = 0.$$

И

$$\sum_{i=1}^n f_i^2 = |A| \frac{|\bar{A}|}{|A|} + |\bar{A}| \frac{|A|}{|\bar{A}|} = n.$$

Таким образом мы свели задачу минимизации функционала 2.2 к следующей задаче оптимизации:

$$\begin{aligned} f^T L f &\rightarrow \min_f, \\ \langle f, \mathbf{1} \rangle &= 0, \\ \|f\| &= \sqrt{n}. \end{aligned}$$

При этом f обязан принимать дискретные значения, согласно тому, как мы вводили этот вектор. Решать дискретную задачу оптимизации сложно, поэтому будем решать полученную задачу приближённо.

$$\mathcal{L} = f^T L f + \lambda_1 f^T \mathbf{1} + \lambda_2 (\|f\| - \sqrt{n})$$

Запишем условие минимума.

$$\nabla_f \mathcal{L} = 2L f + \lambda_1 \mathbf{1} + \lambda_2 \frac{1}{\|f\|} f = \mathbf{0}$$

Домножим это уравнение слева на $\mathbf{1}^T$, тогда, пользуясь ограничениями получим

$$\begin{aligned} 2\mathbf{1}^T L f + \lambda_1 n &= 0 \\ \lambda_1 = \lambda_1^T &= -\frac{2}{n} f^T L \mathbf{1} = 0 \end{aligned}$$

В последней строчке мы воспользовались тем, что матрица L симметричная, а также, что $\mathbf{1}$ – собственный вектор матрицы L с собственным значением 0. В таком случае, условие минимума переписывается следующим образом:

$$2L f + \lambda_2 \frac{1}{\|f\|} f = \mathbf{0}.$$

Учитывая, что $\|f\| = \sqrt{n}$, получаем, что f – собственный вектор матрицы L с собственным значением $\alpha = (-\lambda_2)/(2\sqrt{n})$. Тогда минимизируемый функционал можно переписать как $f^T L f = \alpha \|f\|^2 = \alpha n$. Таким образом, минимум достигается на собственном векторе со вторым минимальным собственным значением α (минимальное собственное значение равно 0 и достигается на собственном векторе $\mathbf{1}$).

Кластеризацию точек можно получить из полученного решения, с помощью решающего правила

$$\begin{cases} v_i \in A, & \text{если } f_i \geq 0 \\ v_i \in \bar{A}, & \text{если } f_i < 0 \end{cases}$$

Но, вместо решающего правила, найденные собственные вектора используют как признаки в новом пространстве, в котором можно провести кластеризацию простым метрическим методом, например, k-means. Таким образом, мы получаем следующий алгоритм:

Дано: матрица смежности W , необходимое число кластеров k .

1. Считаём Лапласиан графа $L = D - W$.
2. Находим собственные векторы Лапласиана u_1, \dots, u_n .
3. Выбираем k собственных векторов с наименьшими k собственными значениями. Записываем матрицу $U = (u_1, \dots, u_k)$, где u_k – k -ый столбец.
4. i -ую строчку матрицы U используем в качестве нового объекта y_i .
5. Кластеризуем новую выборку $\{y_1, \dots, y_n\}$ с помощью k-means.

Ранее мы рассмотрели оптимизационную задачу для случая, когда $k = 2$. Аналогичным образом можно рассмотреть случай, когда k – произвольное.

Введём матрицу H :

$$H_{ij} = \begin{cases} 1/\sqrt{|A_j|}, & v_i \in A_j \\ 0, & \text{иначе} \end{cases}$$

Рассмотрим столбец h_k , он соответствует множеству A_k . В случае, когда множества не пересекаются $H^T H = I$. Запишем

$$\begin{aligned} h_k^T L h_k &= \frac{1}{2} \sum_{i,j} w_{ij} (H_{ik} - H_{jk})^2 \\ &= \frac{1}{2} \sum_{i \in A_k, j \in \overline{A_k}} w_{ij} \frac{1}{|A_k|} + \frac{1}{2} \sum_{i \in \overline{A_k}, j \in A_k} w_{ij} \frac{1}{|A_k|} \\ &= \frac{W(A_k, \overline{A_k})}{|A_k|} \end{aligned}$$

Также мы можем записать $h_k^T L h_k = (H^T L H)_{kk}$. Тогда

$$\text{RatioCut}(A_1, \dots, A_k) = \text{tr}(H^T L H)$$

Получаем следующую оптимизационную задачу:

$$\begin{aligned} \text{tr}(H^T L H) &\rightarrow \min_H, \\ H^T H &= I. \end{aligned}$$

Можно показать, что в таком случае матрица H должна состоять из первых (по возрастанию собственных значений) собственных векторов матрицы L .