BÀI 9 - PHÉP CHIẾU NGẪU NHIÊN

Hạn nộp bài: hết ngày 27/06/2024.

1 Bài tập lý thuyết

Bài 1. Xét ma trận dữ liệu $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{p \times n}$. Đặt $d_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|_2$ là khoảng cách giữa $\mathbf{x}_i$ và $\mathbf{x}_j$ trong không gian Euclide. Bài toán Multidimensional Scaling cố gắng tìm $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_n] \in \mathbb{R}^{k \times n}$ sao cho tổng bình phương nhỏ nhất

$$\min_{\mathbf{Y}=[\mathbf{y}_1,\ldots,\mathbf{y}_n]} \sum_{i,j} \left(\|\mathbf{y}_i - \mathbf{y}_j\|^2 - d_{ij}^2 \right) \tag{1}$$

với điều kiện $\sum_i^n \mathbf{y}_i = \mathbf{0}$. Đặt

$$\mathbf{K} = -\frac{1}{2}\mathbf{HDH}$$

với $\mathbf{D} = \left[d_{ij}^2 \right]$ và $\mathbf{H} = \mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^T$ là ma trận centering. Chứng mình rằng bài toán cực tiểu trong (1) tương đương với

$$\min_{\mathbf{Y} \in \mathbb{R}^{k \times n}} \left\| \mathbf{Y}^T\mathbf{Y} - \mathbf{K} \right\|_F^2, \tag{2}$$

trong đó $\|\cdot\|_F$ là chuẩn Frobenius.

2 Bài tập thực hành

Bài 2. Để thực hiện Locality Sensitive Hashing bằng phép chiếu ngẫu nhiên, ta tiến hành các bước sau:

Bước 1. Phát sinh n siêu phẳng ngẫu nhiên đi qua gốc toạ độ. Việc này tương đương với việc tạo ra n vector ngẫu nhiên để làm vector pháp tuyến cho các siêu phẳng.

Bước 2. Mã hoá các điểm dữ liệu dựa theo vị trí của nó so với bộ siêu phẳng vừa tạo. Cụ thể, một điểm $\mathbf{x}$ bất kì được mã hoá thành $\mathbf{x}' = \overline{x_1 x_2 \cdots x_n}$, trong đó

$$x_k = \begin{cases} 1, & \text{nếu } \mathbf{x} \text{ thuộc phần dương của siêu phẳng thứ } k, \\ 0, & \text{nếu } \mathbf{x} \text{ thuộc phần âm của siêu phẳng thứ } k. \end{cases}$$

Các mã này được dùng để phân cụm các điểm dữ liệu. Khi cần thực hiện truy vấn, thuật toán tìm kiếm sẽ tiến hành:

1. Mã hoá dữ liệu truy vấn $\mathbf{q}$ thành $\mathbf{q}'$;

2. Tìm tập các điểm $\left\{\mathbf{x}_i'\right\}_{i \in I}$ gần $\mathbf{q}'$ (theo khoảng cách Hamming);

3. Trong $\{\mathbf{x}_i\}_{i \in I}$, chọn ra điểm tương đồng nhất với $\mathbf{q}$.

Tham khảo: Random Projection for Locality Sensitive Hashing.

Yêu cầu:

a) Phát sinh tập dữ liệu $\mathcal{D}$ gồm 1000 điểm có phân phối đều $\mathcal{U}(-10; 10)$ với số chiều là 20 và 1 điểm dữ liệu truy vấn $\mathbf{q}$ có cùng phân phối và số chiều.

b) Viết hàm `generate_random_hyperplanes(num_planes, dimensions)` để phát sinh các siêu phẳng ngẫu nhiên:

- `num_planes` là số siêu phẳng cần phát sinh;
- `dimensions` là số chiều của các siêu phẳng.

Áp dụng hàm này để phát sinh 10 siêu phẳng có số chiều 20.

c) Viết hàm `lsh_hash_points(points, hyperplanes)` để mã hoá các điểm dữ liệu với một bộ siêu phẳng cho trước:

- `points` là các điểm dữ liệu;
- `hyperplanes` là các siêu phẳng.

Áp dụng hàm này để mã hoá tập dữ liệu $\mathcal{D}$ trong câu a).

d) Viết hàm `query_lsh(hash_table, query_point, hyperplanes)` để mã hoá điểm dữ liệu cần truy vấn, tính các khoảng cách Hamming và trả về danh sách các điểm có mã tương đồng:

- `hash_table` là bộ dữ liệu đã được mã hoá;
- `query_points` là điểm dữ liệu cần truy vấn;
- `hyperplanes` là các siêu phẳng.

Áp dụng hàm này để tìm các điểm có mã tương đồng với điểm $\mathbf{q}$ trong câu a).

e) Trong những điểm tìm được ở câu d), điểm nào tương đồng với $\mathbf{q}$ nhất (theo khoảng cách Euclide)?

f) Sử dụng thuật toán neareast neighbors thông thường để tìm điểm dữ liệu trong $\mathcal{D}$ gần q nhất và so sánh với kết quả ở câu e).

g) Chạy lại các bước trên nhiều lần và nhận xét.