Лекция 6

Преподаватель: Ефимов Владислав

Кластеризация Снижение размерности II

План занятия

Нелинейные методы снижения размерности

Задача кластеризации

Алгоритм k-means

Алгоритмы DBSCAN и HDBSCAN

Кратко о других алгоритмах кластеризации

Метрики качества кластеризации

Обучение без учителя:

Нелинейные преобразования

MDS — многомерное шкалирование

Есть признаковое пространство и мы хотим снизить размерность … (Приэтом хотим чтобы потеряли минимум информации...)

- Бывают метричиские
- Бывают не метрические

Примеры:

PCA - (есть **kernel PCA** — которое делает PCA нелинейным. Но ядро тяжело считать/ Мы еще создаем новую размерность…)

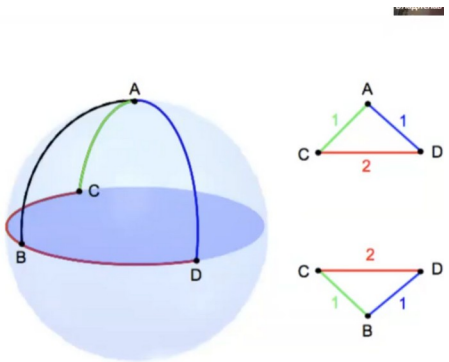
- **kernel trick —** преобразование в виде скалярного произведения K(xi, xj) = <функционал Ф(xi), Ф(xj) > - вместо подсчета Ф считаем все скалярное произведение, что вычислительно дешевле выходит.

ISOMAP
t-SNE
MDS
UMAP

MDS

Multidimensional Scaling (MDS)

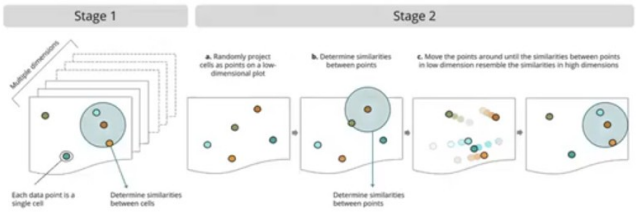
Понятно, что точно воспроизвести расстояния получится не всегда.



T-SNE

t-SNE

- t-SNE (t-distributed Stochastic Neighbour Embedding) — практически многомерное шкалирование.
- Вместо приближения исходных попарных расстояний/метрик мы пытаемся перенести «окрестность» точек из исходного пространства в пространство меньшей размерности.
- Полученные расстояния, скорее всего, не будут соотноситься с исходными.



Мы пытаемся ближние точки переместить в новое пространстве так, чтобы при преобразвании получилось примерно тоже самое (но расстояния могут быть разными, но они будут близки)

t-SNE

- Схожесть между объектами в исходном пространстве $\mathbb{R}^m$

$$p(i,j) = \frac{p(i\,|\,j) + p(j\,|\,i)}{2n}, \quad p(j\,|\,i) = \frac{\exp(-\|\mathbf{x}_j - \mathbf{x}_i\|^2/2\sigma_i^2)}{\sum_{k\neq i}\exp(-\|\mathbf{x}_k - \mathbf{x}_i\|^2/2\sigma_i^2)}.$$

- σ_i неявно задаётся пользователем через параметр perplexity:

$$Perp(P_i) = 2^{H(P_i)},$$
$$H(P_i) = -\sum_j p(j\,|\,i)\log_2 p(j\,|\,i)\,.$$

есть 2 распределения и мы пытаемся их сделать похожими

sigma i = perplexia — это информация показывающая на сколько данные шумные или не шумные

Это определяли схожность в исходном пространстве, а теперь определим схожесть в новом пространстве:

t-SNE

- Схожесть между объектами в целевом пространстве $\mathbb{R}^k, k << m$

$$q(i,j) = \frac{g(|\mathbf{y}_i - \mathbf{y}_j|)}{\sum_{k \neq l} g(|\mathbf{y}_i - \mathbf{y}_j|)},$$

- где $g(z) = \frac{1}{1+z^2}$ — распределение Коши (t-распределение Стьюдента с одной степенью свободы).
- Критерий

$$J_{t-SNE}(y) = KL(P\|Q) = \sum_i \sum_j p(i,j) \log \frac{p(i,j)}{q(i,j)} \to \min_{\mathbf{y}}.$$

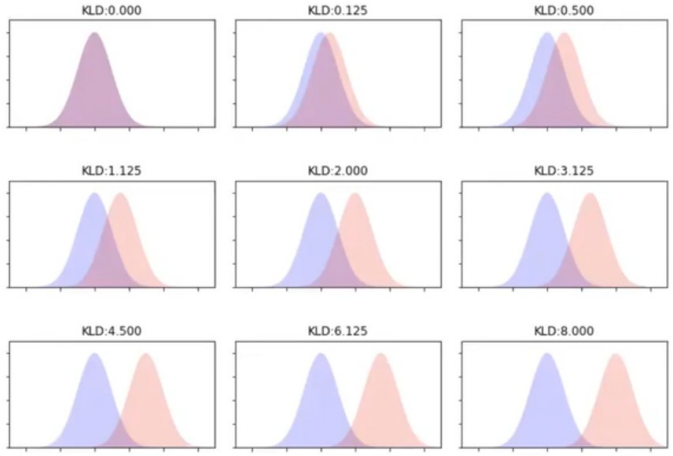
Критерий **Дельвергенция Кульбака Лейблера ---- KLD :**

Дивергенция Кульбака-Лейблера

- Насколько распределение P отличается от распределения Q?
- На этот вопрос отвечает метрика, которая называется дивергенция Кульбака-Лейблера:

$$KL(P\|Q) = \sum_z P(z) \log \frac{P(z)}{Q(z)}$$

- Не симметрична и не отрицательна.
- Последнее верно в силу неравенства Гиббса.



- показывает, как одно распределени отличается от другого

t-SNE. Оптимизация

- Схожесть между объектами в целевом пространстве $\mathbb{R}^k, k << m$

$$q(i,j) = \frac{g(|\mathbf{y}_i - \mathbf{y}_j|)}{\sum_{k \neq l} g(|\mathbf{y}_i - \mathbf{y}_j|)},$$

где $g(z) = \frac{1}{1+z^2}$ — распределение Коши (t-распределение Стьюдента с одной степенью свободы).

- Критерий

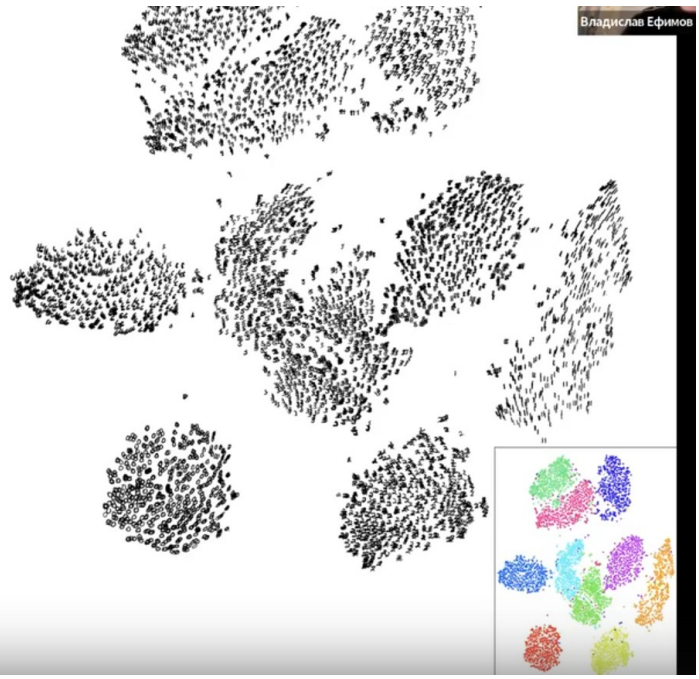
$$J_{t-SNE}(y) = KL(P\|Q) = \sum_i \sum_j p(i,j) \log \frac{p(i,j)}{q(i,j)} \to \min_{\mathbf{y}}.$$

- Оптимизируем этот критерий с помощью градиентного спуска.

t-SNE -нестабильный — отличается от запуска к запуску

t-SNE. Итоги

- Оригинальная статья.
- Примеры.
- Демо и советы.
- t-SNE может быть нестабильным.
- Размеры полученных сгустков могут ничего не значить.
- Расстояния между кластерами могут ничего не значить.
- Полностью шумовые данные могут выдать структуру.



UMAP

- развитие t-SNE
  - более быстрый, т.к отстуствует нормализация

UMAP

- UMAP (Uniform Manifold Approximation and Projection for Dimension Reduction) — развитие подхода t-SNE.
- Более быстрый за счёт отсутствия нормализации для исходного и нового пространств.
- Старается сохранить не только локальную, но и глобальную структуру в данных.

UMAP

- Схожесть между объектами в исходном пространстве $\mathbb{R}^m$

$$p_{ij} = p(i,j) = p(i\,|\,j) + p(j\,|\,i) - p(i\,|\,j)p(j\,|\,i),$$

$$p(i\,|\,j) = \exp(-\frac{d(x_i, x_j) - \rho_i}{\sigma_i}).$$

- $d(x_i, x_j)$ — расстояние между точками (необязательно евклидово), ρ_i — расстояние от i-го элемента до ближайшего соседа, σ_i ищутся, исходя из уравнения (k — гиперпараметр, число соседей):

$$\sum_j p(i,j) = \log_2 k.$$

UMAP

- Схожесть между объектами в целевом пространстве $\mathbb{R}^k, k < m$

$$q_{ij} = q(i,j) = (1 + a(y_i - y_j)^{2b})^{-1},$$

где $a \approx 1.93$ и $b \approx 0.79$ — гиперпараметры и их значения по-умолчанию.

- Критерий — fuzzy set cross entropy, кросс-энтропия для нечётких множеств

$$CE(X,Y) = \sum_i \sum_j \left[p_{ij}(X) \log \frac{p_{ij}(X)}{q_{ij}(Y)} + (1 - p_{ij}(X)) \log \frac{(1 - p_{ij}(X))}{(1 - q_{ij}(Y))} \right].$$

- Оптимизируемся все также с помощью градиентного спуска.

КРИТЕРИЙ тоже меняется

t-SNE vs UMAP

- Если мы сравним критерии для t-SNE и UMAP, то увидим, что UMAP включает в себя критерий t-SNE.
- Но имеет еще дополнительный член, который активируется на объектах далеких друг от друга.
- Из-за этого при оптимизации мы получаем, что UMAP старается сохранить не только локальную структуру, но и глобальную.