ТЕХНОСФЕРА

Лекция 3
Алгоритмы кластеризации II

Николай Анохин

12 октября 2015 г.

Краткое содержание предыдущих лекций

Дано. Признаковые описания N объектов $\mathbf{x} = (x_1, \ldots, x_m) \in \mathcal{X}$, образующие тренировочный набор данных X

Найти. Модель из семейства параметрических функций

$$H = \{h(\mathbf{x}, \theta) : \mathcal{X} \times \Theta \to \mathcal{Y} \mid \mathcal{Y} = \{1, \ldots, K\}\},$$

ставящую в соответствие произвольному $\mathbf{x} \in \mathcal{X}$ один из K кластеров так, чтобы объекты внутри одного кластера были похожи, а объекты из разных кластеров различались

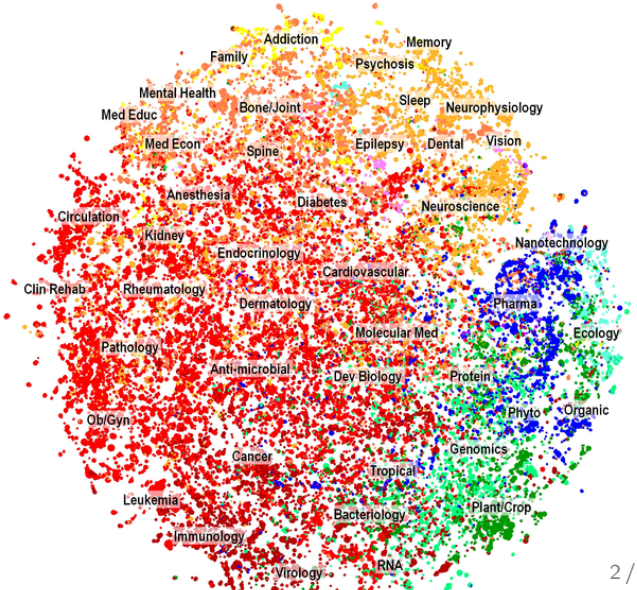
Краткое содержание предыдущих лекций

Рассмотрели классические алгоритмы кластеризации

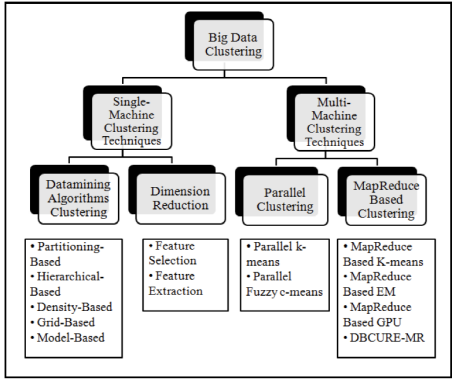
1. Смесь гауссовских распределений и k-means++
2. Hierarchical Clustering
3. DBSCAN

И как их использовать

1. Выбор количества кластеров
2. Оценка качества



Кластеризация больших данных[1]



[1]Big Data Clustering: Algorithms and Challenges

Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH)[2]

Идея метода: построить иерархию кластеров, которая позволит хранить ограниченное количество данных в виде агрегатов.

- *локальность:* каждая точка "кластеризуется" без сканирования всех других точек или имеющихся кластеров
- *выбросы:* точки в "густонаселенных" регионах принадлежат кластерам, а в "малонаселенных" – к выбросам
- *экономность:* используется вся доступная память, при этом минимизируется I/O
- *масштабируемость:* при определенных условиях обучается "онлайн" и требует единственного прохода по данным

[2]BIRCH: An efficient data clustering method for very large datasets

Меры компактности кластера

Центроид

$$\mathbf{x}_0 = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i$$

Радиус

$$R = \frac{1}{N}\sum_{i=1}^{N} d(\mathbf{x}_i, \mathbf{x}_0)$$

Диаметр

$$D = \frac{1}{N(N-1)}\sum_{i=1}^{N}\sum_{j=1}^{N} d(\mathbf{x}_i, \mathbf{x}_j)$$

Clustering Feature

Clustering feature – это объект, содержащий сжатую информацию о кластере.

Определение 1

Пусть кластер C содержит N d-мерных объектов $\mathbf{x}_i$. Clustering feature (CF) для C определяется как тройка $CF = (N, LS, SS)$, где

$$LS = \sum_{i=1}^{N} \mathbf{x}_i, \quad SS = \sum_{i=1}^{N} \mathbf{x}_i^2$$

Утверждение 1

Пусть $CF_1 = (N_1, LS_1, SS_1)$ и $CF_2 = (N_2, LS_2, SS_2)$ – CF для кластеров C_1 и C_2. Тогда CF для кластера, полученного слиянием C_1 и C_2, определяется как

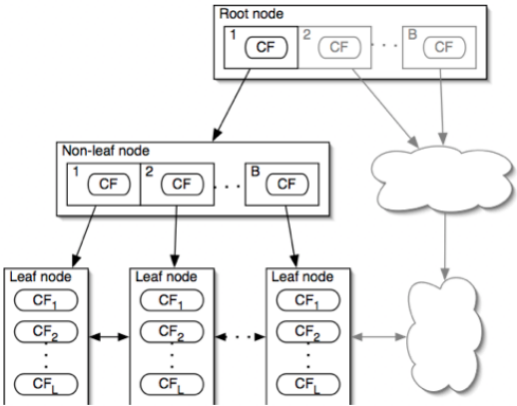
$$CF = (N_1 + N_2, LS_1 + LS_2, SS_1 + SS_2)$$

CF Tree

B – максимальное количество детей у внутреннего узла

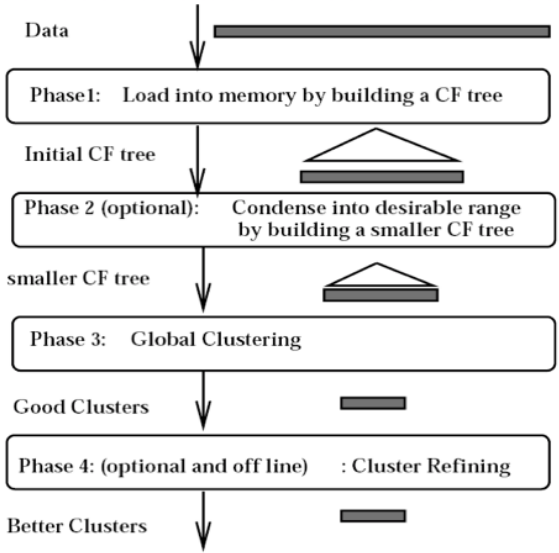
L – максимальное количество детей у листа

T – Максимальная компактность (R или D) ребенка листа

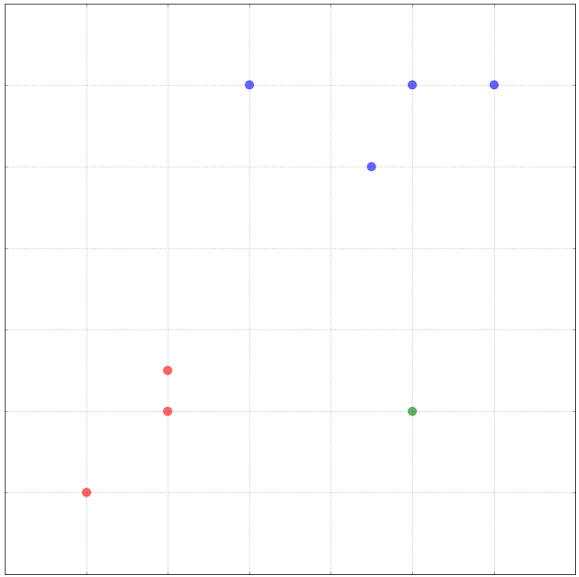


При разделении узла выбираем две наиболее удаленные CF и лепим к ним ближайшие

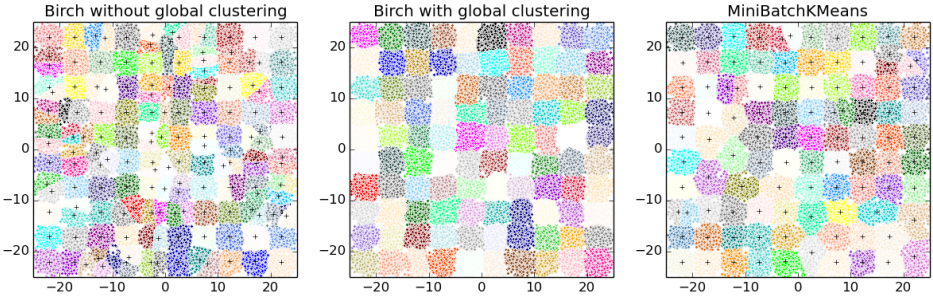
BIRCH


Data
Phase1: Load into memory by building a CF tree
Initial CF tree
Phase 2 (optional): Condense into desirable range by building a smaller CF tree
smaller CF tree
Phase 3: Global Clustering
Good Clusters
Phase 4: (optional and off line) : Cluster Refining
Better Clusters

Пример



Сравнение скорости




```
Birch without global clustering as the final step took 4.19 seconds
n_clusters : 158
Birch with global clustering as the final step took 4.61 seconds
n_clusters : 100
Time taken to run MiniBatchKMeans 5.61 seconds
```

Итоги

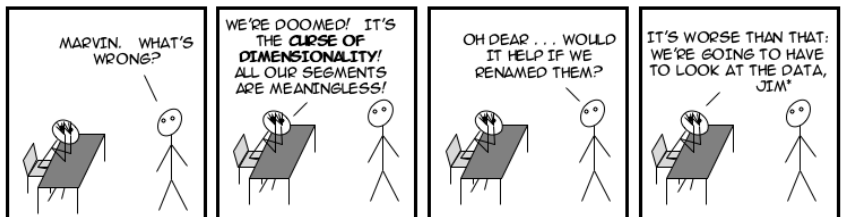
+ Алгоритм работает за $O(N)$
+ Можно контролировать количество используемой памяти
+ В некоторых случаях поддерживается онлайн кластеризация
– Тяжелый подбор параметров
– Только эллиптические кластеры

Другие алгоритмы[3][4][5]

Cls. Algorithms	*Size of Data*	*Cls. Quality*	*Scalability*	*Stability*
EM	Large	High	Low	Suffer from
FCM	Large	High	Low	Suffer from
DENCLUE	Huge	Partially	High	Suffer from
OptiGrid	Huge	Partially	High	Suffer from
BIRCH	Huge	Partially	High	Suffer from

[3] DENCLUE
[4] OptiGrid
[5] Fuzzy c-means

Multidimensional Scaling



Идея метода

Перейти в пространство меньшей размерности так, чтобы расстояния между объектами в новом пространстве были подобны расстояниям в исходном пространстве.

t-Stochastic Neighbour Embedding (t-SNE)[6]

Схожесть между объектами в исходном пространстве

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

Схожесть между объектами в целевом пространстве

$$q(i,j) = \frac{(1 + \|\mathbf{y}_i - \mathbf{y}_j\|^2)^{-1}}{\sum_{k \neq l}(1 + \|\mathbf{y}_k - \mathbf{y}_l\|^2)^{-1}}$$

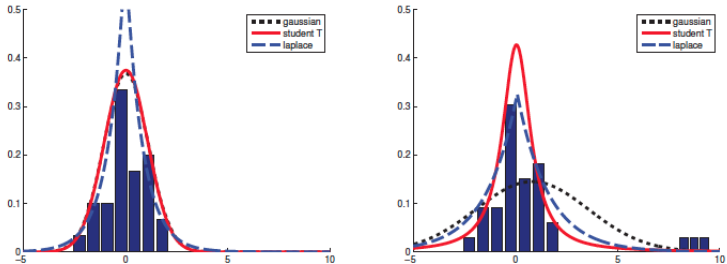
Критерий

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

[6] http://lvdmaaten.github.io/tsne/

t-распределение

$$\tau(\mu, \sigma^2, \nu) \propto \left[1 + \frac{1}{\nu}\left(\frac{x-\mu}{\sigma}\right)^2\right]^{-\frac{\nu+1}{2}}$$



Уильям Госсет 1908 (Student)

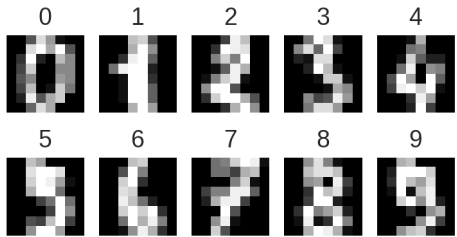
Дивергенция Кульбака-Лейблера

Насколько распределение P отличается от распределения Q?

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

Digits Dataset

около 1800 картинок 8x8 с рукописными цифрами



t-SNE

MNIST Dataset

70000 картинок 20x20 с рукописными цифрами

t-SNE

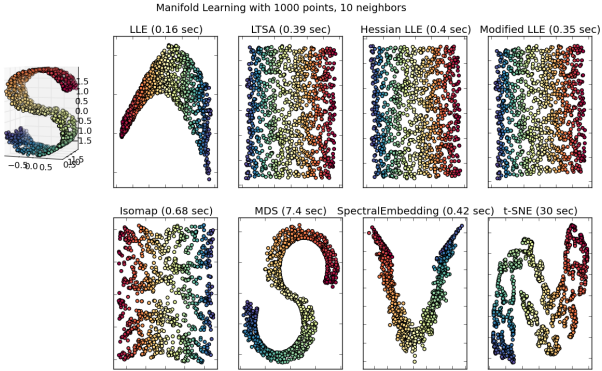
Еще примеры

CalTech

S&P 500

Words

Еще алгоритмы



Вопросы