



BIRCH: A Micro-Clustering-Based Approach

BIRCH (Balanced Iterative Reducing and Clustering Using Hierarchies)

- A multiphase clustering algorithm (Zhang, Ramakrishnan & Livny, SIGMOD'96)
- Incrementally construct a CF (Clustering Feature) tree, a hierarchical data structure for multiphase clustering
 - Phase 1: Scan DB to build an initial in-memory CF tree (a multi-level compression of the data that tries to preserve the inherent clustering structure of the data)
 - Phase 2: Use an arbitrary clustering algorithm to cluster the leaf nodes of the CF-tree
- Key idea: Multi-level clustering
 - Low-level micro-clustering: Reduce complexity and increase scalability
 - High-level macro-clustering: Leave enough flexibility for high-level clustering
- *Scales linearly*: Find a good clustering with a single scan and improve the quality with a few additional scans

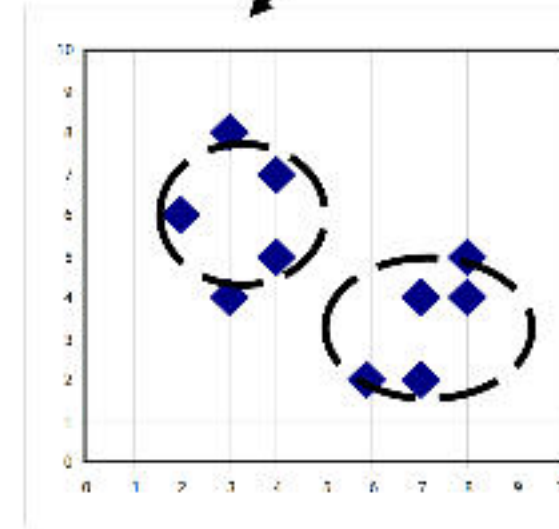
Clustering Feature Vector in BIRCH

□ Clustering Feature (CF): $CF = (N, LS, SS)$

□ N : Number of data points

□ LS : linear sum of N points: $\sum_{i=1}^N X_i$

□ SS : square sum of N points: $\sum_{i=1}^N X_i^2$



$CF = (5, (16,30),(54,190))$

(3,4)

(2,6)

(4,5)

(4,7)

(3,8)

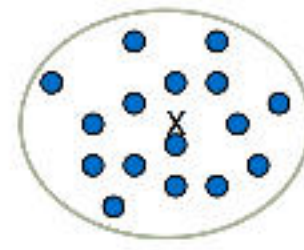
□ Clustering feature:

□ Summary of the statistics for a given sub-cluster: the 0-th, 1st, and 2nd moments of the sub-cluster from the statistical point of view

□ Registers crucial measurements for computing cluster and utilizes storage efficiently

Measures of Cluster: Centroid, Radius and Diameter

Centroid: \vec{x}_0



- the “middle” of a cluster
- n : number of points in a cluster
- \vec{x}_i is the i -th point in the cluster

$$\vec{x}_0 = \frac{\sum_i^n \vec{x}_i}{n}$$

Radius: R

- Average distance from member objects to the centroid
- The square root of average distance from any point of the cluster to its centroid

$$R = \sqrt{\frac{\sum_i^n (\vec{x}_i - \vec{x}_0)^2}{n}}$$

Diameter: D

- Average pairwise distance within a cluster
- The square root of average mean squared distance between all pairs of points in the cluster

$$D = \sqrt{\frac{\sum_i^n \sum_j^n (\vec{x}_i - \vec{x}_j)^2}{n(n-1)}}$$

The CF Tree Structure in BIRCH

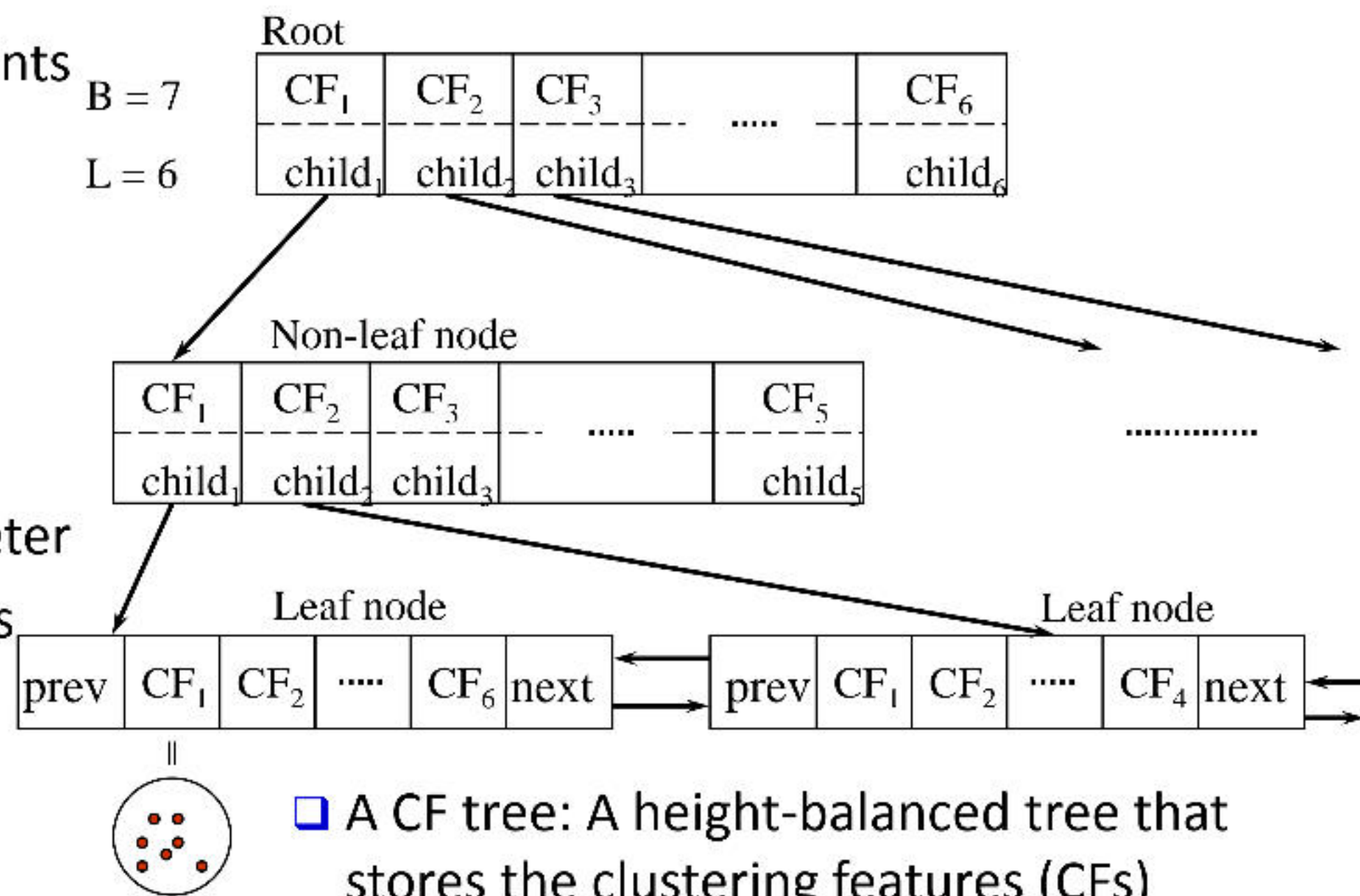
- Incremental insertion of new points (similar to B+-tree)

- For each point in the input

- Find closest leaf entry
- Add point to leaf entry and update CF
- If entry diameter $>$ max_diameter
 - split leaf, and possibly parents

- A CF tree has two parameters

- Branching factor: Maximum number of children
- Maximum diameter of sub-clusters stored at the leaf nodes



- A CF tree: A height-balanced tree that stores the clustering features (CFs)
- The non-leaf nodes store sums of the CFs of their children

BIRCH: A Scalable and Flexible Clustering Method

- An integration of agglomerative clustering with other (flexible) clustering methods
 - Low-level micro-clustering
 - Exploring CP-feature and BIRCH tree structure
 - Preserving the inherent clustering structure of the data
 - Higher-level macro-clustering
 - Provide sufficient flexibility for integration with other clustering methods
- Impact to many other clustering methods and applications
- Concerns
 - Sensitive to insertion order of data points → 对插入数据的顺序敏感
 - Due to the fixed size of leaf nodes, clusters may not be so natural
 - Clusters tend to be spherical given the radius and diameter measures