Big Data Computing - HW 1 - G24

Local space analysis for MRPrintStatistics

April 5, 2025

1 Local space indicator M_L analysis

The local space indicator M_L represents the maximum amount of main memory required by a single invocation of a map or reduce function, for storing the input and any data structure needed by the invocation. The maximum is taken over all rounds and all invocations in each round.
We analyze the local space indicator M_L for the MapReduce implementation of `MRPrintStatistics`. Given:

- N = total number of input points
- L = number of partitions
- d = dimension of each point ($d \geq 1$)
- K = number of centroids ($K \geq 1$)
- A B = demographic groups

1.1 Function overview

The `MRPrintStatistics` function assigns each point to its closest centroid and counts points by centroid and demographic group. It is implemented as follows:

Algorithm 1 `MRPrintStatistics`

1: **function** FIND_CLOSEST_CENTROID(point)
2: Extract `point_x` and `point_label` from point ▷ $O(d)$ space
3: Calculate distances to all centroids ▷ $O(K)$ space
4: Create list of `(distance, centroid, label)` tuples ▷ $O(K \cdot d)$ space
5: Find closest centroid ▷ $O(d)$ space
6: Return `(centroid, label)` ▷ $O(d)$ space
7: **end function**
8: Map `find_closest_centroid` to `inputPoints` ▷ $O(k \cdot d)$ space for each point
9: Aggregate counts of `(centroid, label)` pairs ▷ $O(\frac{N}{L})$ space for keys in partition

1.2 Bound derivation

From the analysis, we see that the maximum additional space during the Map phase for each point is $O(K \cdot d)$, and the maximum additional space during the Reduce phase (specifically, the local aggregation step for `reduceByKey`) is $O(\frac{N}{L})$.
Therefore, the overall M_L is the maximum of these requirements:

$$M_L = \max(\text{Map Phase}, \text{Reduce Phase}) \tag{1}$$

$$= \max(O(K \cdot d), O(\frac{N}{L})) \tag{2}$$

$$= O(K \cdot d + \frac{N}{L}) \tag{3}$$

$$= O(\frac{N}{L}) \tag{4}$$

Since K (number of clusters) and d (dimensionality), in our case, we can assume that are much smaller than $\frac{N}{L}$ (number of points per partition), the $O(\frac{N}{L})$ cost dominates, so the complexity is sublinear with respect to the input size.