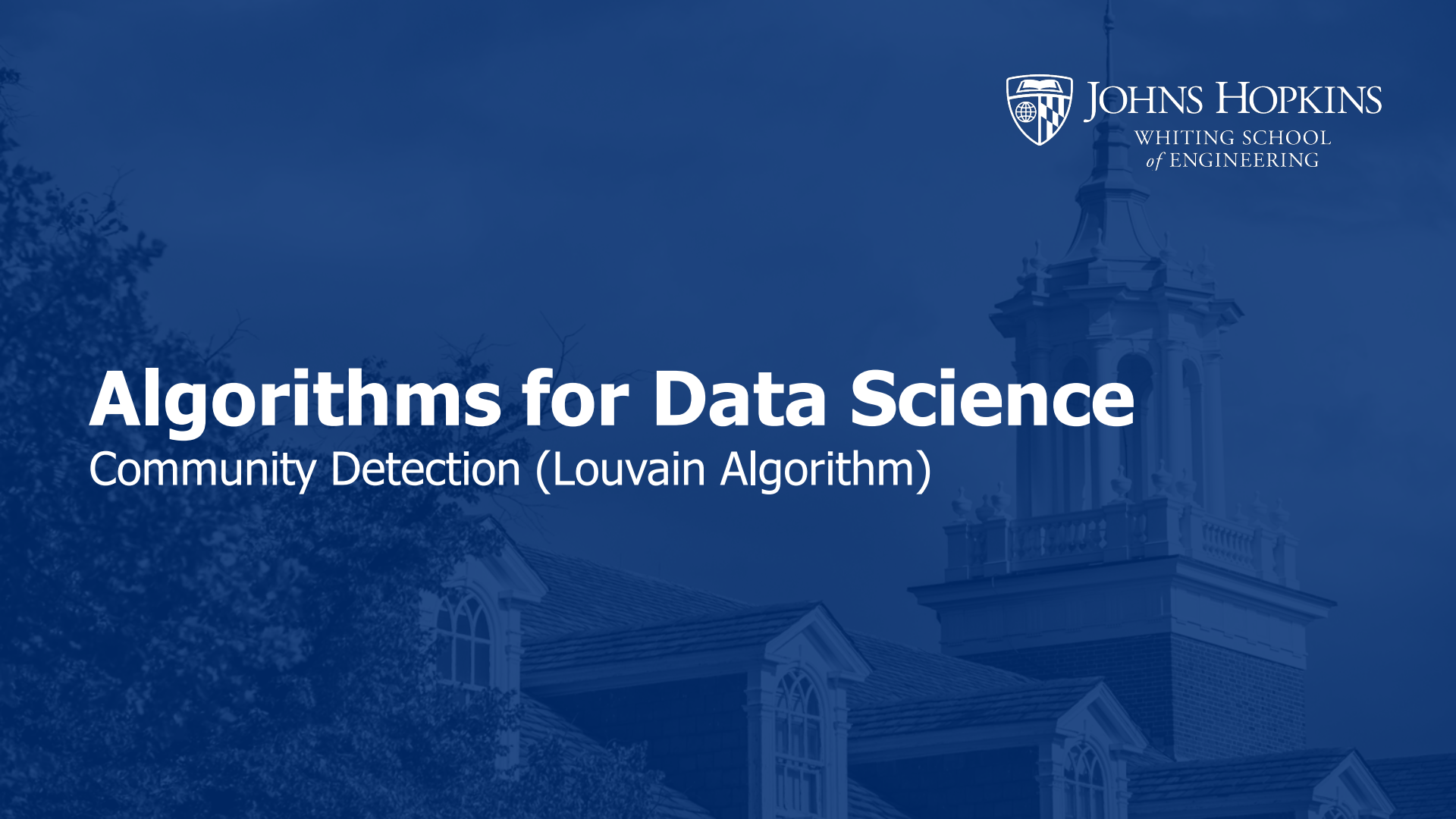
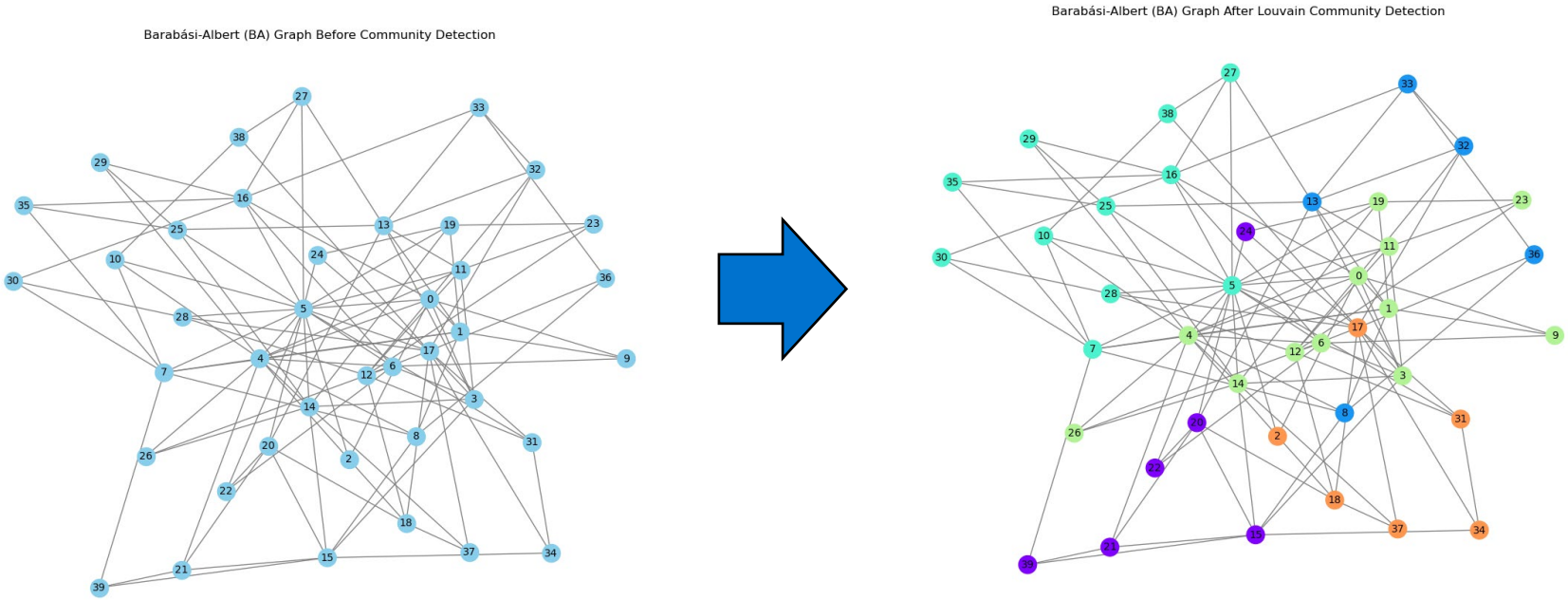


Algorithms for Data Science

Community Detection (Louvain Algorithm)

Louvain Algorithm: Finding Communities



How the Louvain Algorithm Works

Modularity:

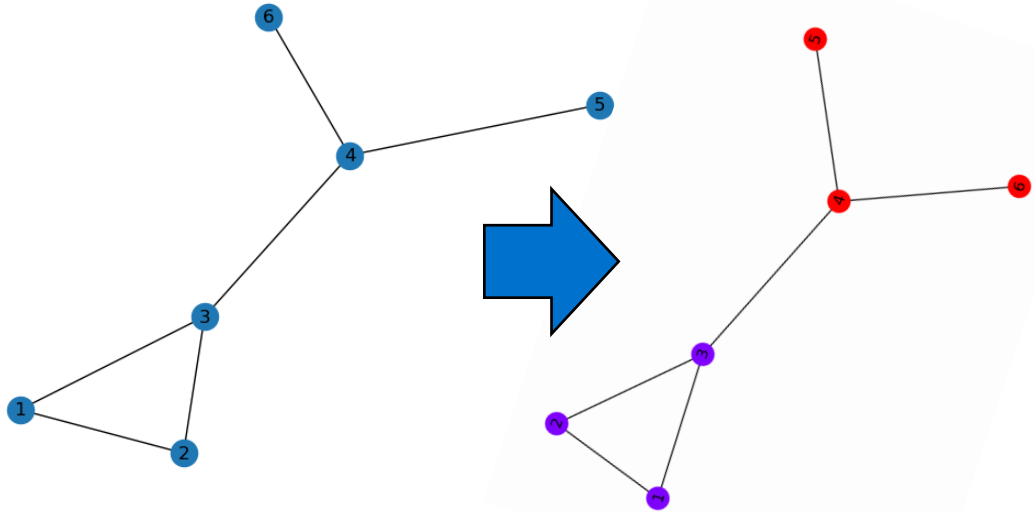
$$Q = \frac{1}{2m} \sum_{i,j} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j$$

Phase I: Local Moving Phase

- Each node is assigned to its best neighboring community.
- Modularity is recomputed after each reassignment.

Phase II: Aggregation Phase

- Communities are merged into "super nodes".
- The process repeats on the new graph.



Community Detection Algorithm Comparison

Algorithm	Approach	Complexity	Best For
Louvain	Greedy Modularity	$O(n \log n)$	Large Graphs
Girvan-Newman	Edge Betweeness	$O(n^3)$	Small Graphs
Label Propagation	Spreading Labels	$O(n)$	Real-time Updates
Spectral Clustering	Eigenvectors of Laplacian	$O(n^2)$	Moderate-Sized Graphs