JioInstitute

Minimum Spanning Tree for Indian Supply Chain Optimization using Kruskal’s Algorithm

Group Name: -

1) Ashutosh Singh
2) Ashutosh Patil
3) Saris Singh

Professor: - 1) Sudipta Roy

Teacher Assistant: - 1) Priyanka Mishra

INDEX

1. Introduction

In modern supply chain management, minimizing transportation costs while maintaining efficient connectivity between suppliers, warehouses, and distribution centres is critical. However, with increasing network complexity and dynamic logistics requirements, determining the most cost-effective structure for supply routes poses a significant challenge.

This project applies Kruskal's Algorithm to find a Minimum Spanning Tree (MST) over a graph of major Indian cities connected by cost-weighted routes. The MST represents the most cost-efficient subset of connections covering all locations without forming cycles.

2. Data Used

Locations (Nodes):
- Delhi, Mumbai, Bangalore, Chennai, Kolkata, Hyderabad

Routes (Edges with Costs):
Each route represents a transportation link between two cities, with an associated cost (in ₹):

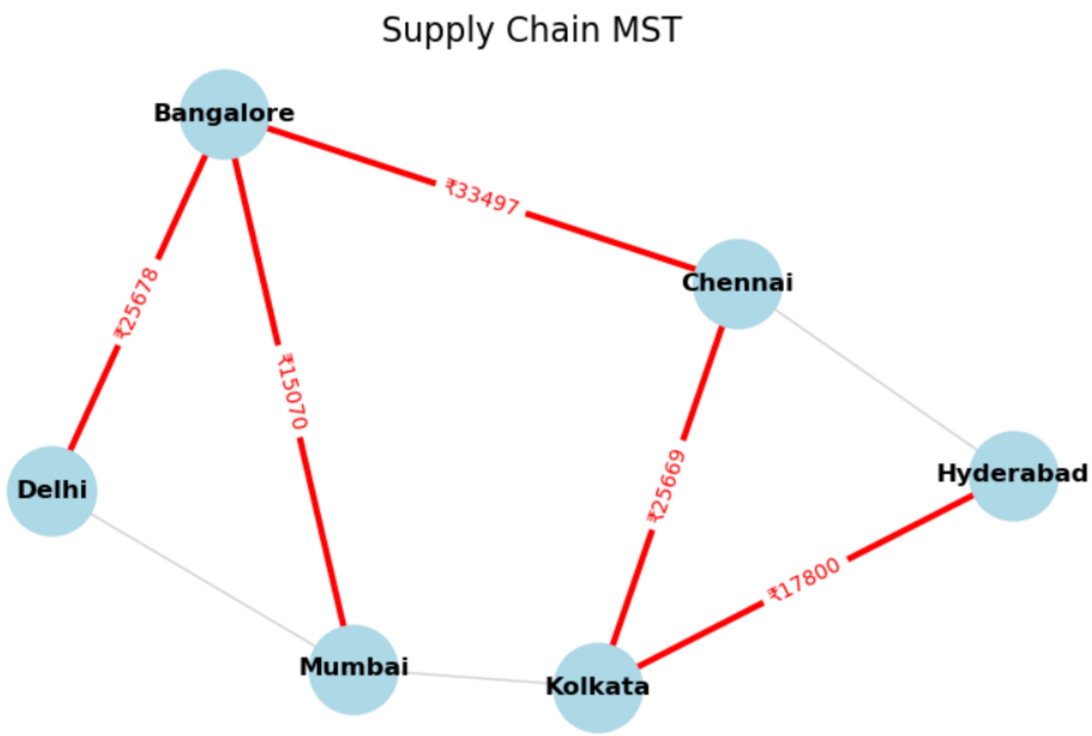
From	To	Cost (₹)
Delhi	Mumbai	45,322
Delhi	Bangalore	25,678
Mumbai	Bangalore	15,070
Mumbai	Kolkata	79,800
Bangalore	Chennai	33,497
Chennai	Kolkata	25,669
Chennai	Hyderabad	48,990
Kolkata	Hyderabad	17,800

4. Exploratory Data Analysis (EDA)

- Total Cities (Nodes): 6
- Total Routes (Edges): 8
- Max Cost: ₹79,800 (Mumbai - Kolkata)
- Min Cost: ₹15,070 (Mumbai - Bangalore)
- Graph Type: Undirected weighted graph

Visuals were plotted using Network X. MST was clearly highlighted using red edges with fare labels.

- The major analysis was done based on route wise mapping so that minimum cost would be required for the supply, considering the facts that multiple deliveries could be possible along the major stoppages.



5. Key Results

Minimum Spanning Tree (MST) Output:
Mumbai - Bangalore: ₹15070
Delhi - Bangalore: ₹25678
Chennai - Kolkata: ₹25669
Kolkata - Hyderabad: ₹17800
Bangalore - Chennai: ₹33497

Total Minimum Cost: ₹117714

The MST includes 5 edges (for 6 nodes), forming the most cost-effective connection plan without any loops.

Time Complexity:

- Sorting edges: O (E log E)
- Processing edges: up to O (E log V) using simple methods like union–find with path compression.

6. Future Enhancements

- Dynamic Data Input: Allow importing from external data files (e.g., CSV/Excel).
- Interactive Map Visualization: Use geolocation data and libraries
- Add Constraints: Consider truck capacity, road traffic, or delivery times.
- ML for Cost Prediction: Predict future costs using ML models.
- Integration with Real-World APIs: Use live logistics or map data.

7. Conclusion

This project successfully demonstrated how Kruskal's algorithm can be applied to real-world logistics problems to minimize transportation costs. It provides a foundational framework that can be extended with real-world data and constraints for scalable solutions.