# INDUSTRIAL TRAINING DEFENSE
# EE4T001

MALSAWMZUALA KHIANGTE
22EE01068

# Designing a Post-Disaster Humanitarian Supply Chain

Using Machine Learning and Multi-Criteria Decision-Making Techniques

# Contents

# 01. Introduction

Humanitarian supply chain is a set of coordinated actions meant to guarantee the effective transportation, storage and distribution of necessities in the event of natural or man made disaster

Urgency, uncertainty and coordination problems distinguish the humanitarian supply chain from other
supply chainsh text

# Problem Description

Disasters such as earthquakes, floods, fires, and landslides affect multiple regions and cause mass injuries. The most urgent goal is to minimize casualties, but rescue forces (RFs) are limited in number and capacity. In such high-pressure situations, delays or misallocation of resources can result in preventable loss of life

To overcome these limitations, it's crucial to:

- Determine the optimal number of rescuers per region.
- Prioritize which regions and which patients need help first.


# Research Questions

- How can **ML/AI** improve humanitarian supply chains?
- How should **transport, sourcing, and environment** be considered?
- How can disaster and disruption impacts be reduced?
- How can **MCDM** support humanitarian supply chains?

# Main Objectives of the Study

To prioritize affected regions

To classify injured individuals by severity

To optimize resource allocation

To support pre-disaster preparedness

# Methodology

# MCDM
## (Multi-Criteria Decision Making)

- Helps evaluate options with multiple, conflicting criteria.
- Useful in disaster management for quick trade-offs under uncertainty.
- In this study:
  - BWM → assign importance to criteria (population, hazard, access).
  - WASPAS → rank districts by priority.
  - Results guide resource allocation and highlight urgent areas.

*BWM is MCDM method developed by Rezaei (2015):*

# **BWM** (Best-Worst Method)

MCDM technique to assign weights to criteria (hazard, population, accessibility) using expert judgment.

**How it works (Steps):**

1. List decision criteria
➤ Example: Population, hazard exposure, accessibility

2. Select the Best and Worst criteria
➤ Most and least important as judged by experts

3. Rate the Best over all others (1–9 scale)
➤ Forms the Best-to-Others vector

$$A_B = (a_{B1}, a_{B2}, \ldots, a_{Bn})$$

4. Rate all others over the Worst (1–9 scale)
➤ Forms the Others-to-Worst vector

$$A_W = (a_{1W}, a_{2W}, \ldots, a_{nB})$$

5. Solve optimization model
➤ Minimizes max inconsistency to calculate optimal weights

$$\left|\frac{W_B}{W_j} - a_{Bj}\right| \le \varepsilon \quad \forall j$$

$$\left|\frac{W_j}{W_W} - a_{jW}\right| \le \varepsilon \quad \forall j$$

$$\sum W_j = 1 \qquad \forall j$$

$$W_j \ge 0$$

# WASPAS

## (Weighted Aggregated Sum Product Assessment)

MCDM method that ranks alternatives by weighted performance, combining WSM and WPM.

### How it works (Steps):

1. Use BWM weights

➤ The weights of criteria (from BWM) are input to WASPAS.

2. Normalize decision matrix

➤ Scores of each region on each criterion are normalized (e.g., min-max normalization).

$$\bar{x}_{ij} = \frac{x_{ij}}{\max_i x_{ij}} \text{ For profitable criteria}$$

$$\bar{x}_{ij} = \frac{\min_i x_{ij}}{x_{ij}} \text{ for non-profitable criteria}$$

3. Apply both WSM and WPM formulas

➤ Compute:

$Q^{(1)}$: Weighted sum score $$Q_i^{(1)} = \sum_{j=1}^{n} \bar{x}_{ij} w_{ij}$$

$Q^{(2)}$: Weighted product score $$Q_i^{(2)} = \prod_{j=1}^{n} \left(\bar{x}_{ij}\right)^{w_j}$$

4. Combine both scores with both equally important to get final WASPAS index:

$$Q_i = 0.5Q_i^{(1)} + 0.5Q_i^{(2)}$$

# Machine Learning Algorithm

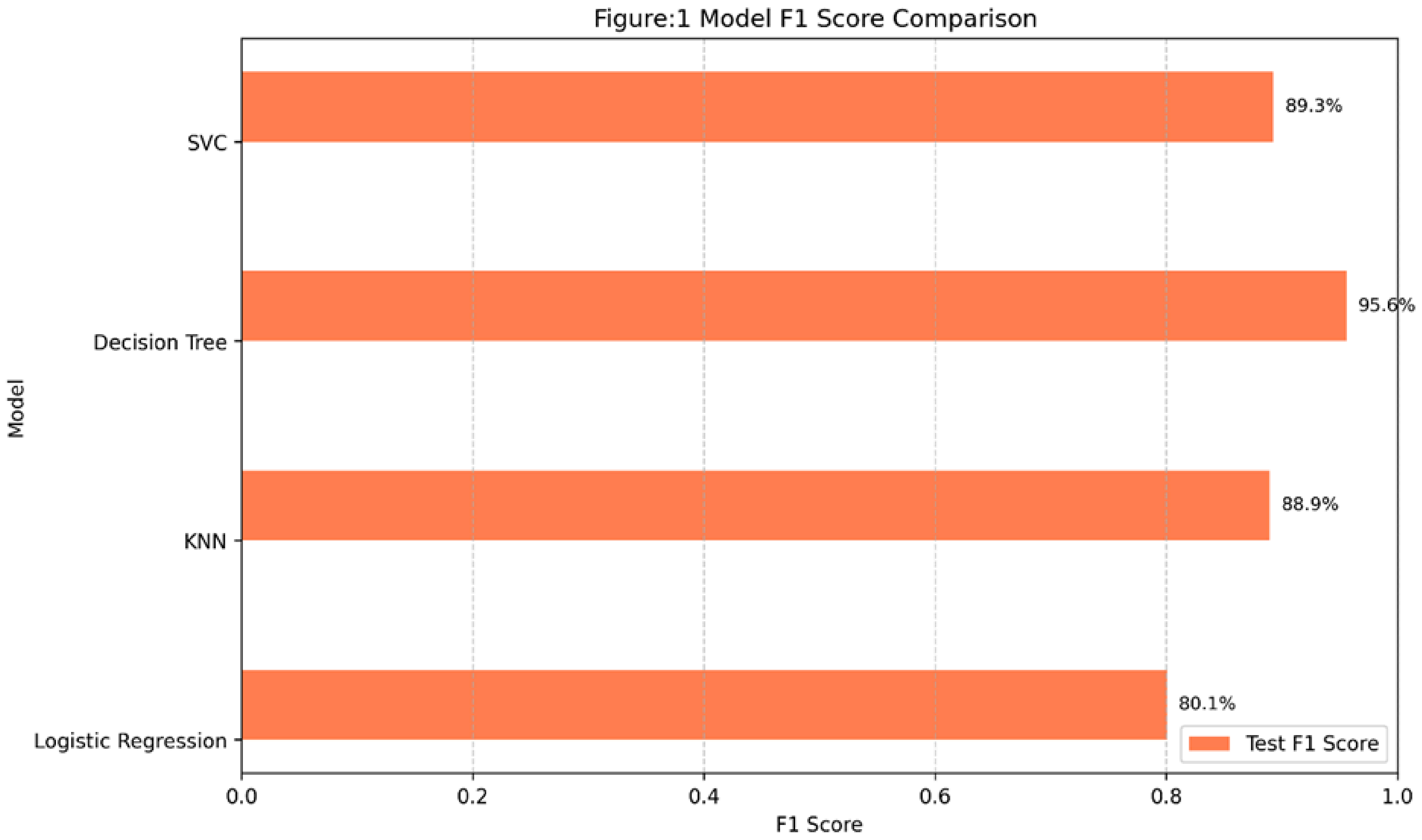
- Enables systems to learn from data and make predictions.
- In this study: predicts injury severity (very high, high, medium, low, very low) from **synthetic patient data** to prioritize rescue.

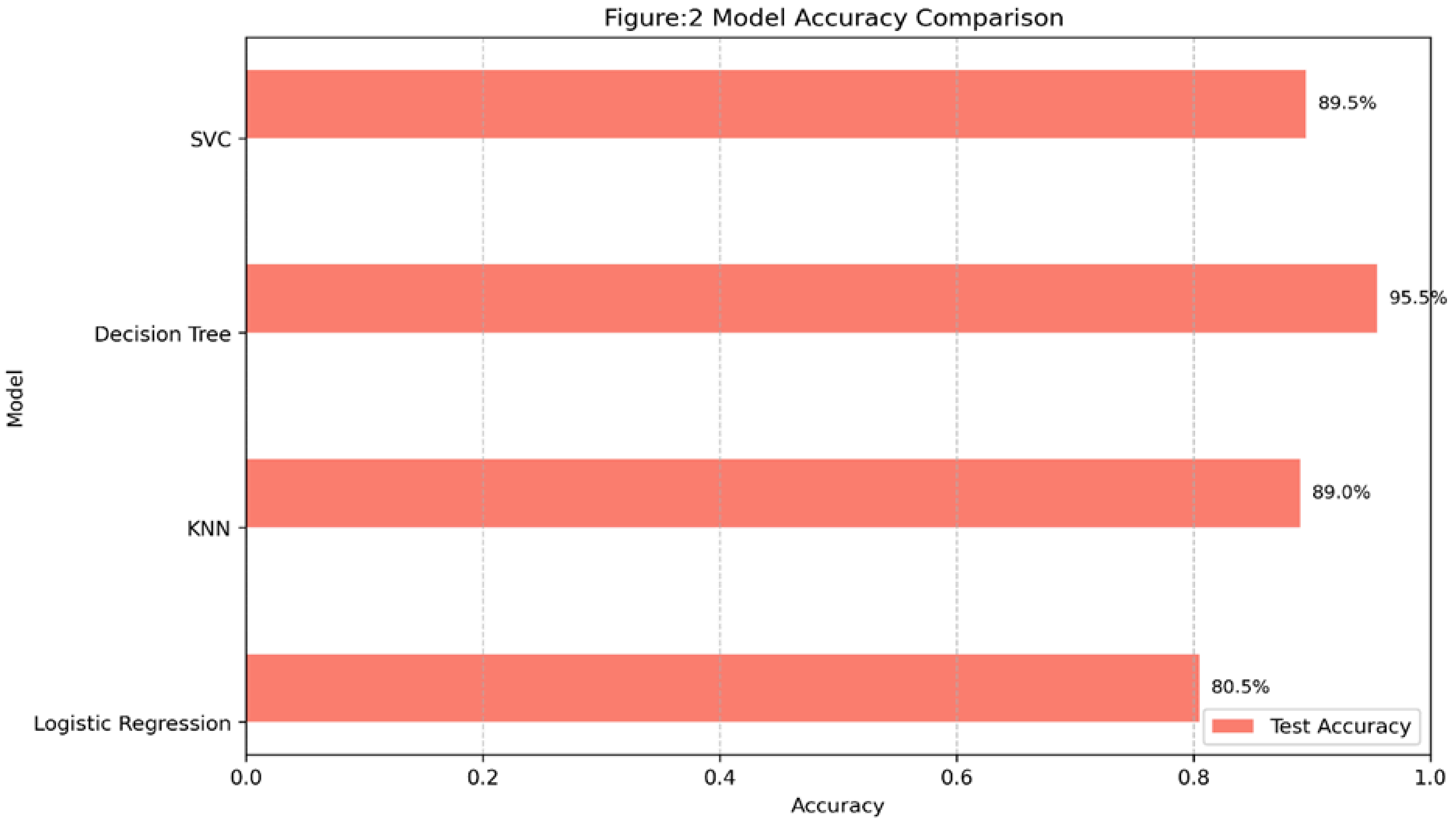
| Variable | Variable Type | Variable | Variable type |
|---|---|---|---|
| **Age** | Integer | **Pulse rate** | Integer |
| **Gender** | Boolean | **Mobility** | Boolean |
| **Pregnancy Status** | Boolean | **Trapped Duration** | Integer |
| **Consciousness Level** | Boolean | **Blood Pressure** | Integer |
| **Severe Bleeding** | Boolean | | |

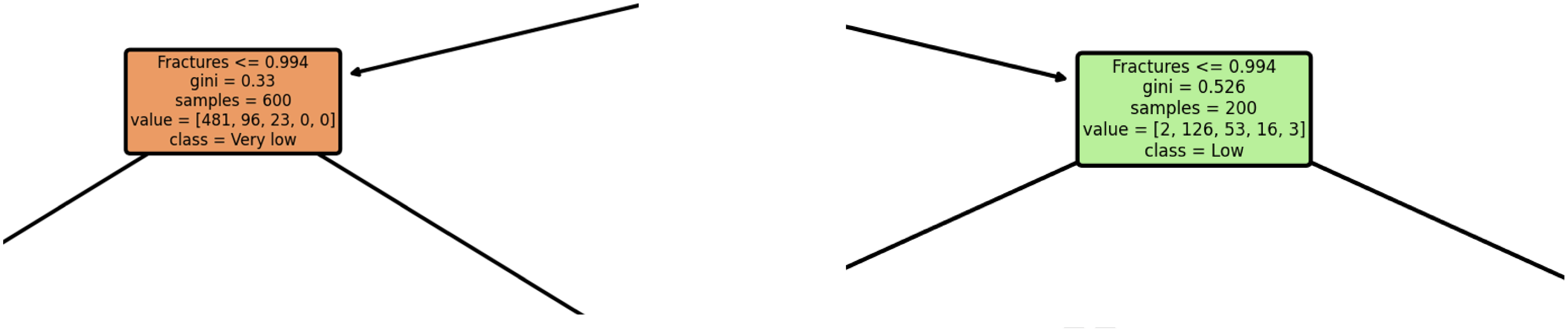
```python
def assign_condition(row):
    score = 0
    if row["Age"] < 10 or row["Age"] > 50:
        score += 2
    if row["Gender"] == 0:
        score += 1
    if row["Pregnancy_Status"] == 1:
        score += 2
    if row["Consciousness_Level"] == 0:
        score += 2
    if row["Severe_Bleeding"] == 1:
        score += 2
    if row["Fractures"] == 1:
        score += 2
    if row["Pulse_Rate"] > 120 or row["Pulse_Rate"] < 50:
        score += 2
    if row["Mobility"] == 0:
        score += 2
    if row["Trapped_Duration_Hours"] > 5:
        score += 1

    if score >= 8:
        return "Very High"
    elif score >= 6:
        return "High"
    elif score >= 4:
        return "Medium"
    elif score >= 2:
        return "Low"
    else:
        return "Very Low"

severity_map = {
    "Very Low": 0,
    "Low": 1,
    "Medium": 2,
    "High": 3,
    "Very High": 4
}
```

*(Note: Patient records are synthetic and do not represent real individuals.)*

- Compared algorithms (Decision Tree, KNN, SVM, etc.) using accuracy, precision, recall, F1-score.
- Decision Tree Classifier selected for best performance.




Fractures <= 0.994
gini = 0.33
samples = 600
value = [481, 96, 23, 0, 0]
class = Very low
Fractures <= 0.994
gini = 0.526
samples = 200
value = [2, 126, 53, 16, 3]
class = Low

# Bi-Objective Optimization

- Optimizes two competing goals simultaneously:
  - **Save lives** → minimize distance/time.
  - **Save resources** → minimize cost.
- Produces **Pareto-optimal** trade-offs instead of favoring one objective.
- Modeled as a **MILP** with linear equations and mixed decision variables.
- This study **integrates patient severity** (ML), **area risk** (MCDM), and real-world constraints (capacity, cost, time).

# Objective Funtions

## Objective 1: Minimize Total Distance Traveled by RFs

$$\text{Min } Z_1 = \sum_{I}\sum_{J}\sum_{O}\sum_{M}\sum_{S}\sum_{L} X_{i,j,o,m,s,l} \cdot D_{ij}$$

**Meaning:**
Reduces **total travel distance** to accelerate rescue operations and minimize delay in reaching patients.

## Objective 2: Minimize Total Operational Cost

$$\text{Min } Z_2 = \sum_{M} A_i \cdot Q_{i,m} + \sum_{I}\sum_{O}\sum_{L} C_0 \cdot N_{i,o,l} + \sum_{I} Y_i \cdot CA_i$$

**Meaning:**
Captures total cost of **Activating relief centers** + **Vehicle purchases** (by type and time)+**Rescuer deployment**

# Constraints:

| Constraint (Math) | Meaning |
|---|---|
| $\sum X_{i,j,o,m,s,l} \geq F_m$ | Enough rescuers assigned per area |
| $L \leq \sum Y_i \leq U$ | Limit number of active RCs |
| $\sum A_i Q_{i,m} + \sum C_0 N_{i,o,l} + \sum Y_i CA_i \leq B$ | Stay within total budget |
| $\sum X_{i,j,o,m,s} \leq N_{i,o,l} \cdot CP_o$ | Vehicle capacity constraint |
| $\sum X_{i,j,o,m,s,l} = 1$ | Each injured individual must be rescued |
| $\sum X_{i,j,o,m,s,l} = 1$ | Each injured has one injury condition |
| $\sum X_{i,j,o,m,s,l} = N_s$ | Rescuer allocation per condition |
| $Q_{i,m} \geq a_m$ | Rescuer count ≥ ML forecast |
| $Q_{i,m} \geq 0$ | Non-negative rescuer allocation |
| $N_{i,o,l} \geq 0$ | Non-negative vehicle count |
| $X_{i,j,o,m,s,l} \in \{0,1\}$ | Binary assignment decision |
| $Y_i \in \{0,1\}$ | Binary RC activation decision |

# Solution Methodology – LP-Metric Method

- **LP-Metric:** converts multi-objective problems into a single objective by minimizing distance to the ideal solution.
- Balances two conflicting goals:
  - **Minimize travel distance**
  - **Minimize operational cost**
- Produces a **compromise solution** that is well-balanced, not extreme in either objective.

$$LP = \left\{ \sum_{j=1}^{k} W_j \times \left[ \frac{f_j\left(X^{*j}\right) - f_j(X)}{f_j\left(X^{*j}\right) - f_j\left(X^{-j}\right)} \right]^P \right\}^{\frac{1}{P}}$$

Where:

- $f_j(X)$: Value of objective $j$ at solution $X$
- $f_j^*$: **Ideal** (best possible) value of objective $j$
- $f_j^-$: **Anti-ideal** (worst-case) value of objective $j$
- $w_j$: Weight for objective $j$ (often equally weighted)
- $p$: Norm (typically 1 or 2)

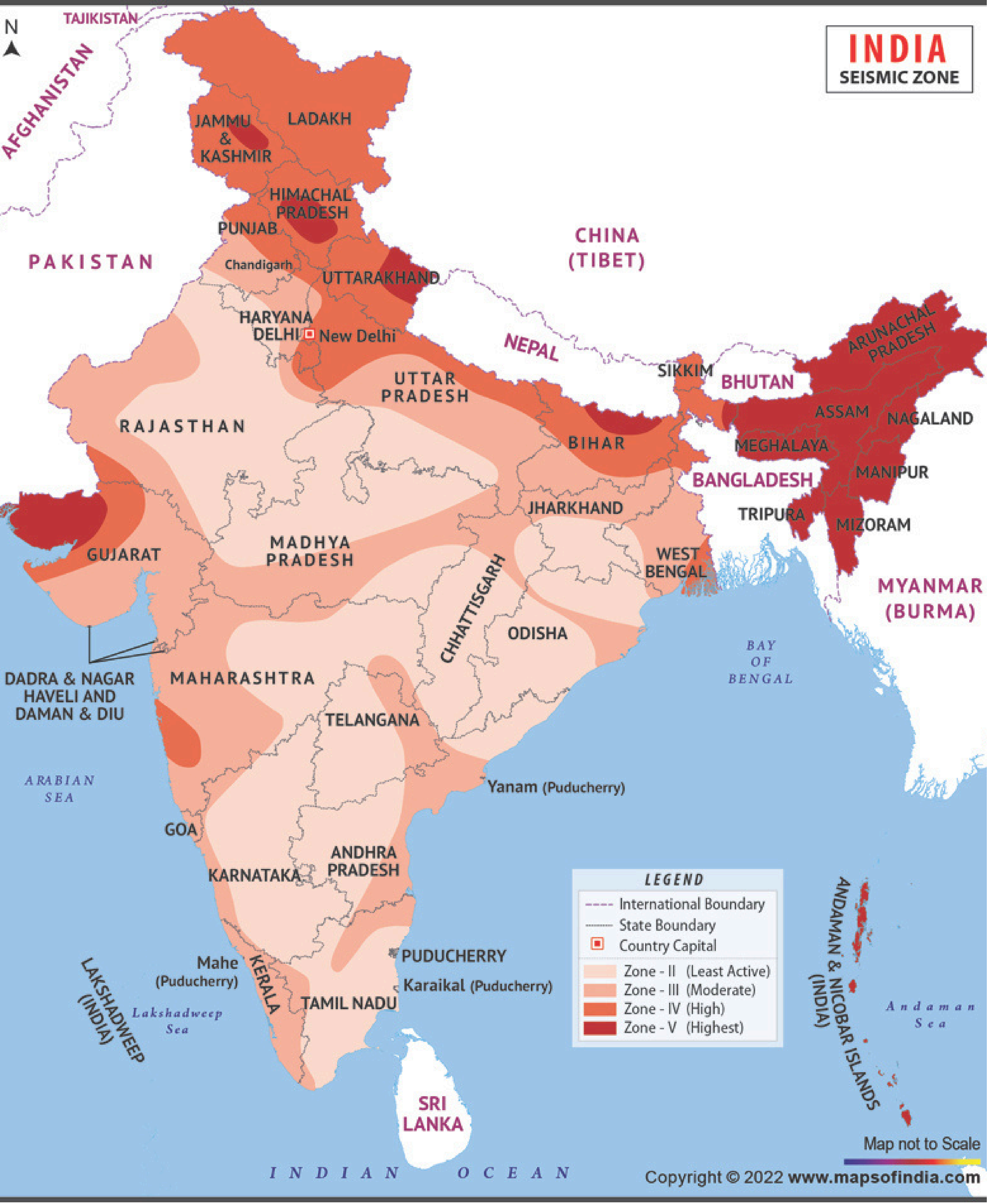
# Model Assumptions

**All patients are assigned to a relief centre (RC)**

**Patient data is synthetic, and severity is predicted using a trained ML model**

**Each patient is limited to their 3 nearest RCs**

**Severe patients must be rescued earlier**




# Case Study

A synthetic case study was developed for Mizoram, India, simulating a hypothetical earthquake to test disaster response strategies using machine learning, MCDM, and bi-objective optimization. Mizoram was chosen because it lies in Seismic Zone V, the highest earthquake risk category in India, making it highly vulnerable to severe seismic events.

# District Data & Prioritization (MCDM Approach)

## District Data Preparation

- 11 districts of Mizoram considered.
- Key attributes: Population, Area Size, Earthquake Hazard Factor, Accessibility.
- Data sourced from 2011 Census and updated using growth estimates.
- Criteria normalized and prepared for ranking.

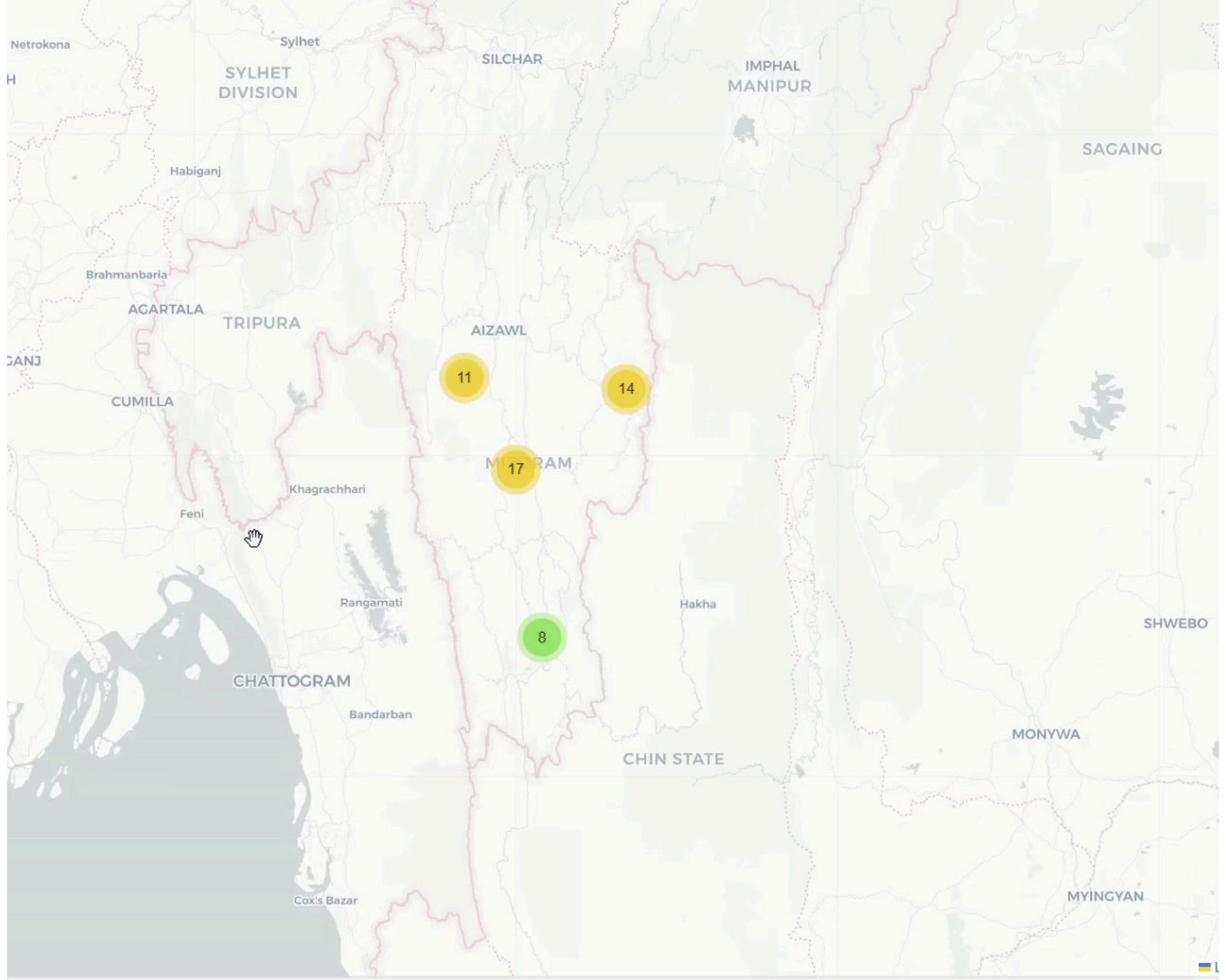
## Prioritization

- Best-Worst Method (BWM): Calculated weights of criteria
- WASPAS Method: Aggregated normalized values to rank districts.
- Final scores used as priority weights ($\alpha_m$) in optimization model.

| | District | Area_ID | Population | Area Size | Hazard Factor | Accessibility | WASPAS_Score | Rank |
|---|---|---|---|---|---|---|---|---|
| 0 | Aizawl | 1 | 400309 | 3577.00 | 7 | 10 | 0.781367 | 1 |
| 2 | Champhai | 3 | 125745 | 3185.83 | 9 | 7 | 0.702890 | 2 |
| 1 | Lunglei | 2 | 161428 | 4536.00 | 9 | 6 | 0.679851 | 3 |
| 4 | Mamit | 5 | 86364 | 3025.00 | 8 | 7 | 0.636324 | 4 |
| 5 | Serchhip | 6 | 64937 | 1421.00 | 7 | 8 | 0.631177 | 5 |
| 7 | Saiha | 8 | 56574 | 1399.00 | 10 | 3 | 0.607730 | 6 |
| 3 | Kolasib | 4 | 83955 | 1382.00 | 6 | 8 | 0.597058 | 7 |
| 6 | Lawngtlai | 7 | 117894 | 2557.00 | 8 | 4 | 0.573286 | 8 |
| 8 | Hnahthial | 9 | 28500 | 1133.00 | 8 | 5 | 0.572594 | 9 |
| 10 | Saitual | 11 | 30000 | 1500.00 | 7 | 7 | 0.569012 | 10 |
| 9 | Khawzawl | 10 | 28500 | 1152.00 | 6 | 6 | 0.510333 | 11 |

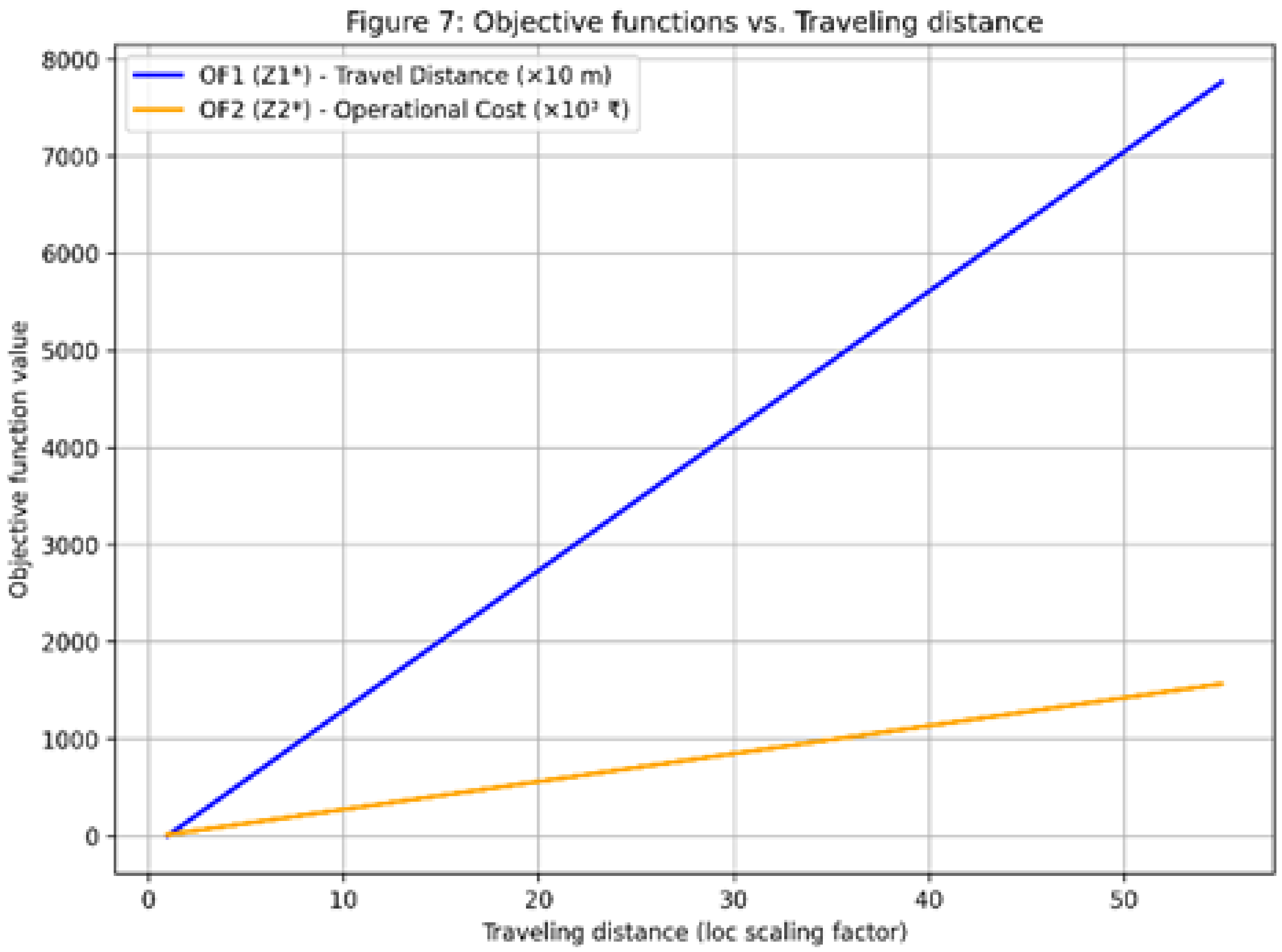
# Machine Learning for Severity Prediction

- Synthetic dataset: patients with randomized age, gender, vitals, injury, entrapment time.
- ML model: Decision Tree Classifier trained to predict 5 severity levels (0–4).




| ge | Gender | Pregnancy_Status | Consciousness_Level | Severe_Bleeding | Fractures | Pulse_Rate | Mobility | Trapped_Duration_Hours | Blood_Pressure | Area ID | Condition_Label | AreaNam |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 47 | 0 | 1 | 1 | 0 | 0 | 61 | 1 | 5 | 133 | 7 | 1 | Lawngtlai |
| 30 | 0 | 1 | 0 | 1 | 0 | 88 | 0 | 9 | 133 | 4 | 4 | Kolasib |
| 17 | 0 | 0 | 1 | 0 | 0 | 55 | 1 | 7 | 80 | 4 | 1 | Kolasib |
| 45 | 0 | 1 | 1 | 1 | 0 | 81 | 1 | 5 | 176 | 3 | 2 | Champhai |
| 25 | 0 | 0 | 1 | 0 | 0 | 57 | 1 | 7 | 91 | 1 | 1 | Aizawl |
| 33 | 0 | 1 | 1 | 0 | 0 | 58 | 1 | 5 | 139 | 3 | 1 | Champhai |
| 34 | 1 | 0 | 1 | 0 | 0 | 58 | 1 | 9 | 138 | 6 | 0 | Serchhip |
| 38 | 1 | 0 | 1 | 0 | 0 | 59 | 1 | 5 | 132 | 4 | 0 | Kolasib |
| 48 | 1 | 0 | 1 | 0 | 0 | 61 | 1 | 9 | 180 | 2 | 0 | Lunglei |
| 18 | 0 | 0 | 1 | 0 | 1 | 55 | 0 | 11 | 108 | 4 | 3 | Kolasib |
| 26 | 1 | 0 | 1 | 1 | 0 | 77 | 1 | 6 | 147 | 5 | 1 | Mamit |
| 49 | 0 | 0 | 0 | 1 | 0 | 91 | 0 | 12 | 152 | 7 | 3 | Lawngtlai |
| 17 | 0 | 0 | 1 | 0 | 0 | 55 | 1 | 9 | 120 | 1 | 1 | Aizawl |
| 39 | 0 | 0 | 1 | 0 | 0 | 59 | 1 | 2 | 141 | 3 | 0 | Champhai |
| 48 | 1 | 0 | 1 | 0 | 0 | 61 | 1 | 4 | 173 | 7 | 0 | Lawngtlai |
| 47 | 1 | 0 | 1 | 0 | 0 | 61 | 1 | 11 | 125 | 7 | 0 | Lawngtlai |
| 41 | 0 | 0 | 1 | 1 | 0 | 80 | 1 | 9 | 135 | 7 | 2 | Lawngtlai |
| 27 | 0 | 0 | 1 | 1 | 0 | 77 | 1 | 1 | 132 | 7 | 1 | Lawngtlai |
| 38 | 1 | 0 | 1 | 0 | 0 | 59 | 1 | 10 | 141 | 2 | 0 | Lunglei |
| 27 | 0 | 0 | 1 | 0 | 0 | 57 | 1 | 6 | 104 | 6 | 1 | Serchhip |
| 36 | 0 | 0 | 1 | 0 | 0 | 59 | 1 | 1 | 139 | 7 | 0 | Lawngtlai |
| 3 | 1 | 0 | 1 | 0 | 0 | 52 | 1 | 1 | 80 | 5 | 1 | Mamit |
| 14 | 0 | 0 | 1 | 0 | 0 | 54 | 1 | 5 | 114 | 2 | 0 | Lunglei |
| 37 | 1 | 0 | 1 | 0 | 0 | 59 | 1 | 8 | 135 | 1 | 0 | Aizawl |
| 52 | 0 | 1 | 1 | 0 | 0 | 62 | 1 | 5 | 172 | 3 | 2 | Champhai |
| 37 | 0 | 0 | 0 | 1 | 1 | 89 | 0 | 8 | 155 | 1 | 4 | Aizawl |
| 54 | 1 | 0 | 0 | 1 | 0 | 92 | 1 | 11 | 171 | 7 | 2 | Lawngtlai |
| 40 | 1 | 0 | 1 | 1 | 0 | 80 | 1 | 10 | 155 | 4 | 1 | Kolasib |

# Bi-Objective Optimization Model




To evaluate the emergency response planning, a bi-objective mixed-integer programming model was built using **Pyomo** and solved using **Gurobi optimizer**.

**Objectives:**

1. Minimize rescue distance ($Z_1$)
2. Minimize operational cost ($Z_2$)
3. Balance trade-off via LP-Metric ($Z_3$)

# Key Constraints & Parameters

- Relief Centers: Exactly 4 active (Aizawl, Lunglei, Champhai, Lawngtlai).
- Budget: ₹5 lakh total for activation, training, vehicles, and transport.
- Vehicles: ≤20; cost ₹15–20/km; capacity 4 patients.
- Rescuers: Assigned by severity (1–4 per patient); 10–20 per center.
- Patient Assignment: Each patient to 1 of 3 nearest centers, within ~20 km.

- **LP-metric ($Z_3$):** used normalized $Z_1$ (distance) and $Z_2$ (cost) to study trade-offs.
- **Limitation:** $Z_1$ relied on Euclidean distances → stayed near worst case, keeping $Z_3 \approx 0.5$.
- **Insight:** highlights need for road-network data and proper scaling in real applications.

| | Z1* (×10 m) | Z1_worst (×10 m) | Z2* (×10^3 ₹) | Z2_worst (×10^3 ₹) | Z1_combined (×10 m) | Z2_combined (×10^3 ₹) | Z3 |
|---|---|---|---|---|---|---|---|
| **0** | 2.2755 | 3.1985 | 13.6397 | 50.2551 | 3.1985 | 13.6397 | 0.500000 |
| **1** | 863.6355 | 864.9230 | 185.9846 | 219.9271 | 864.9125 | 185.9825 | 0.495891 |
| **2** | 1725.9345 | 1727.2130 | 358.4426 | 392.3869 | 1727.2130 | 358.4426 | 0.500000 |
| **3** | 2588.2345 | 2589.5130 | 530.9026 | 564.8469 | 2589.5130 | 530.9026 | 0.500000 |
| **4** | 3450.5350 | 3451.8125 | 703.3625 | 737.3070 | 3451.8125 | 703.3625 | 0.500000 |
| **5** | 4312.8335 | 4314.1125 | 875.8225 | 909.7667 | 4314.1125 | 875.9225 | 0.501473 |
| **6** | 5175.1330 | 5176.4105 | 1048.3821 | 1082.2266 | 5176.4105 | 1048.2821 | 0.498523 |

# Sensitivity Analysis

Sensitivity analysis tested the model's robustness by varying patient numbers, severity profiles, and budget, and observing their effects on distance ($Z_1$), cost ($Z_2$), and resource allocation (vehicles and rescuers).

*Fig:3 Sensitivity analysis of $Z_1$ vs $N_s$*

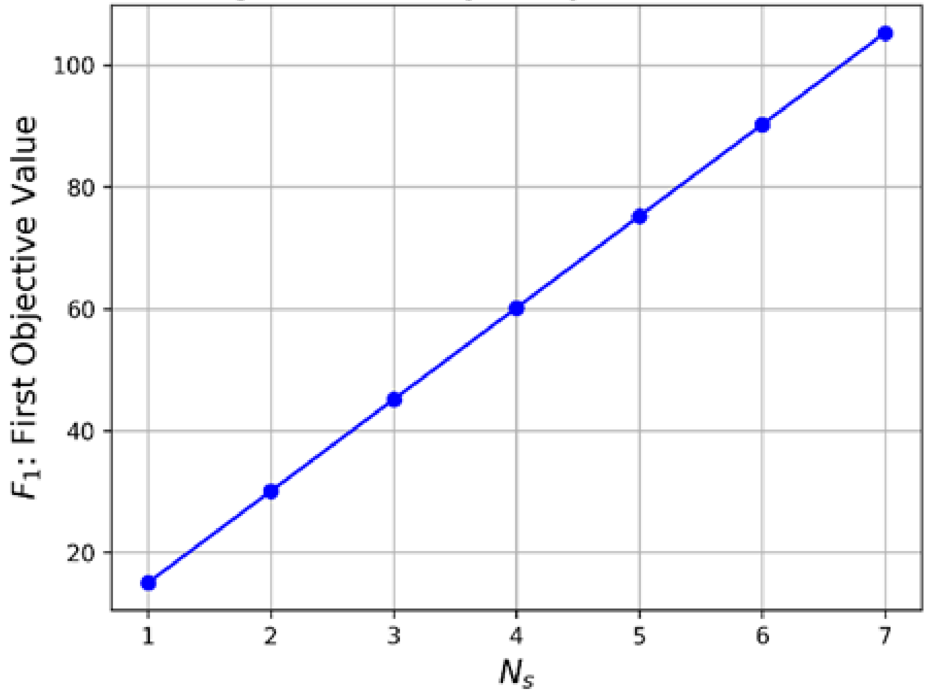
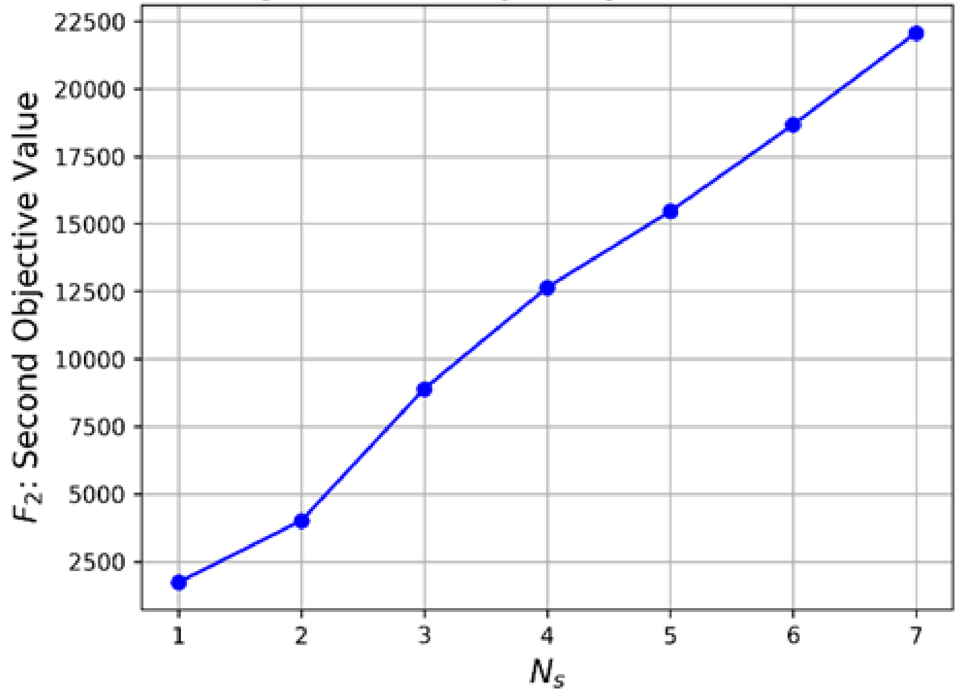



*Fig : 4 Sensitivity analysis of $Z_2$ vs $N_s$*



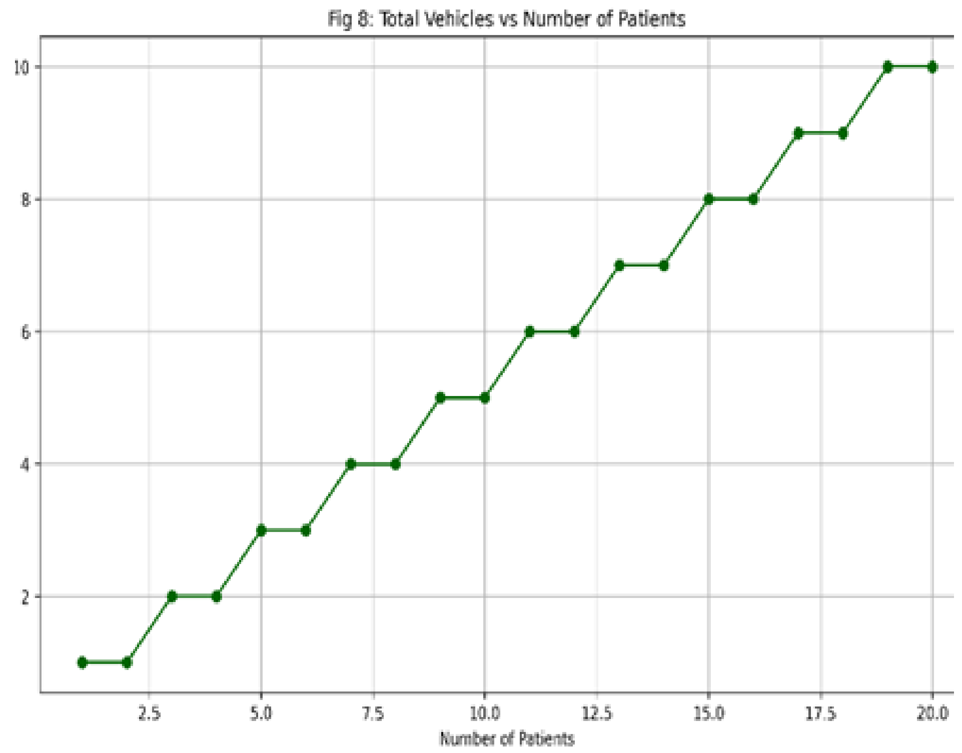




*Fig: 5 Sensitivity of First Objective on Number of Patients*

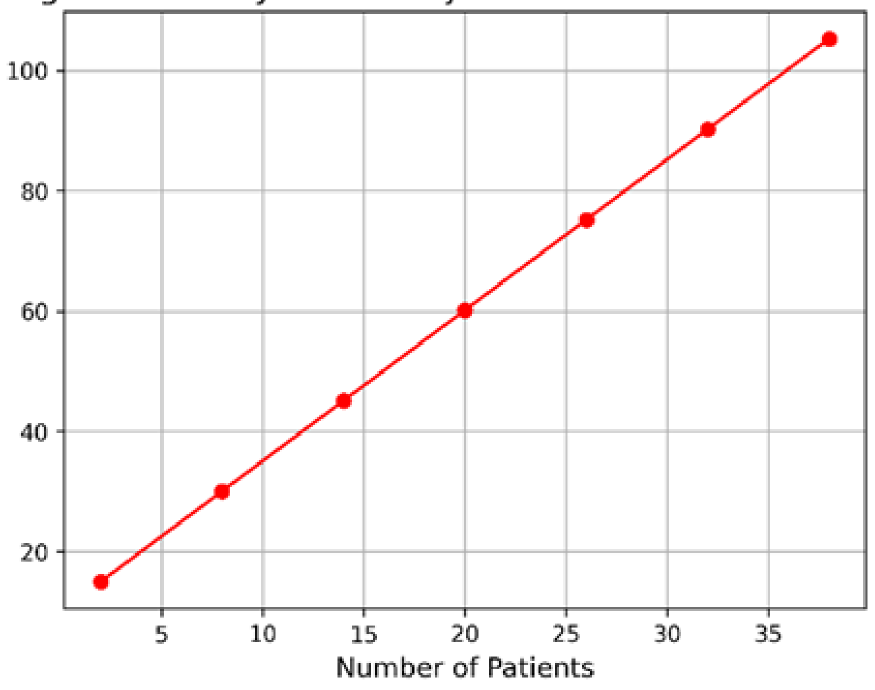
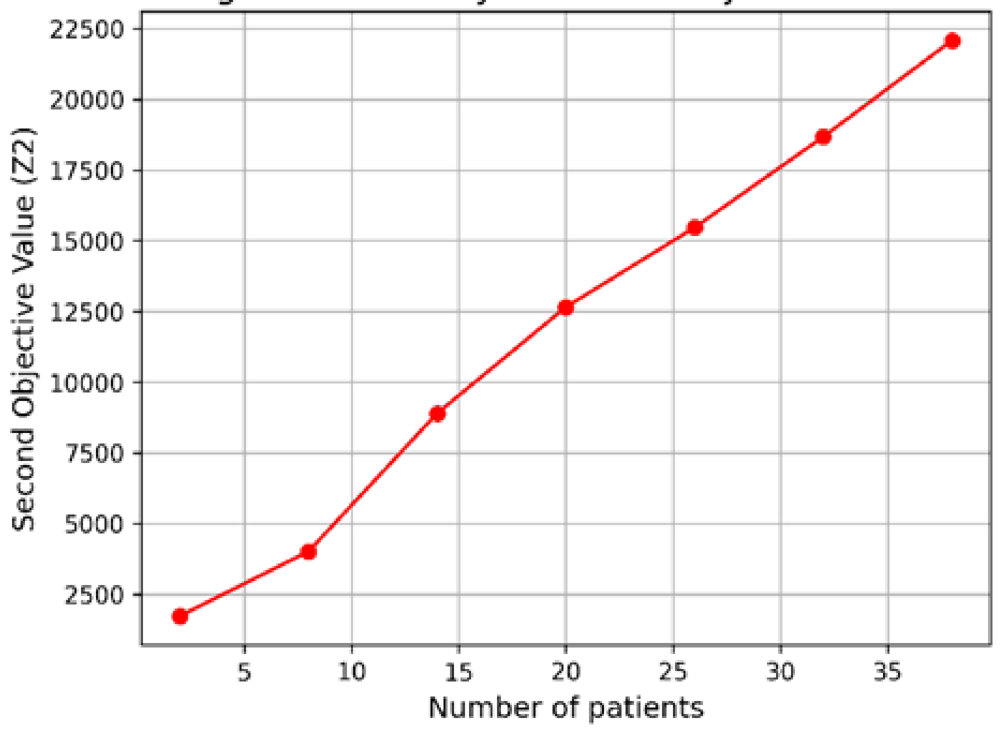
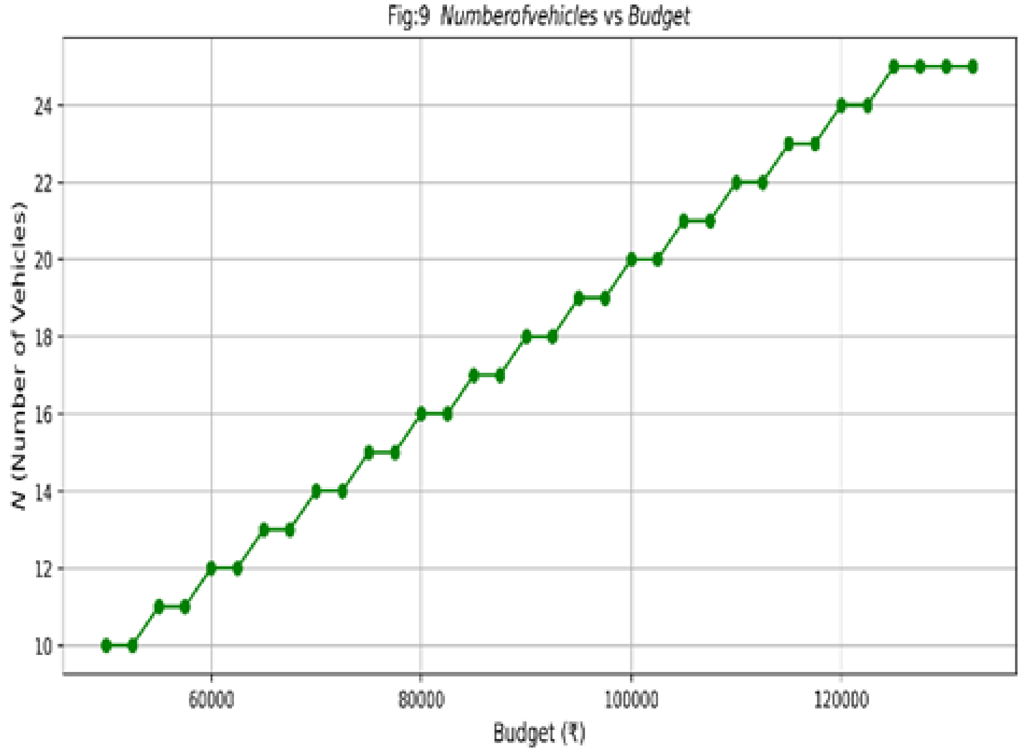



*Fig: 6 Sensitivity of second Objective on Fm*



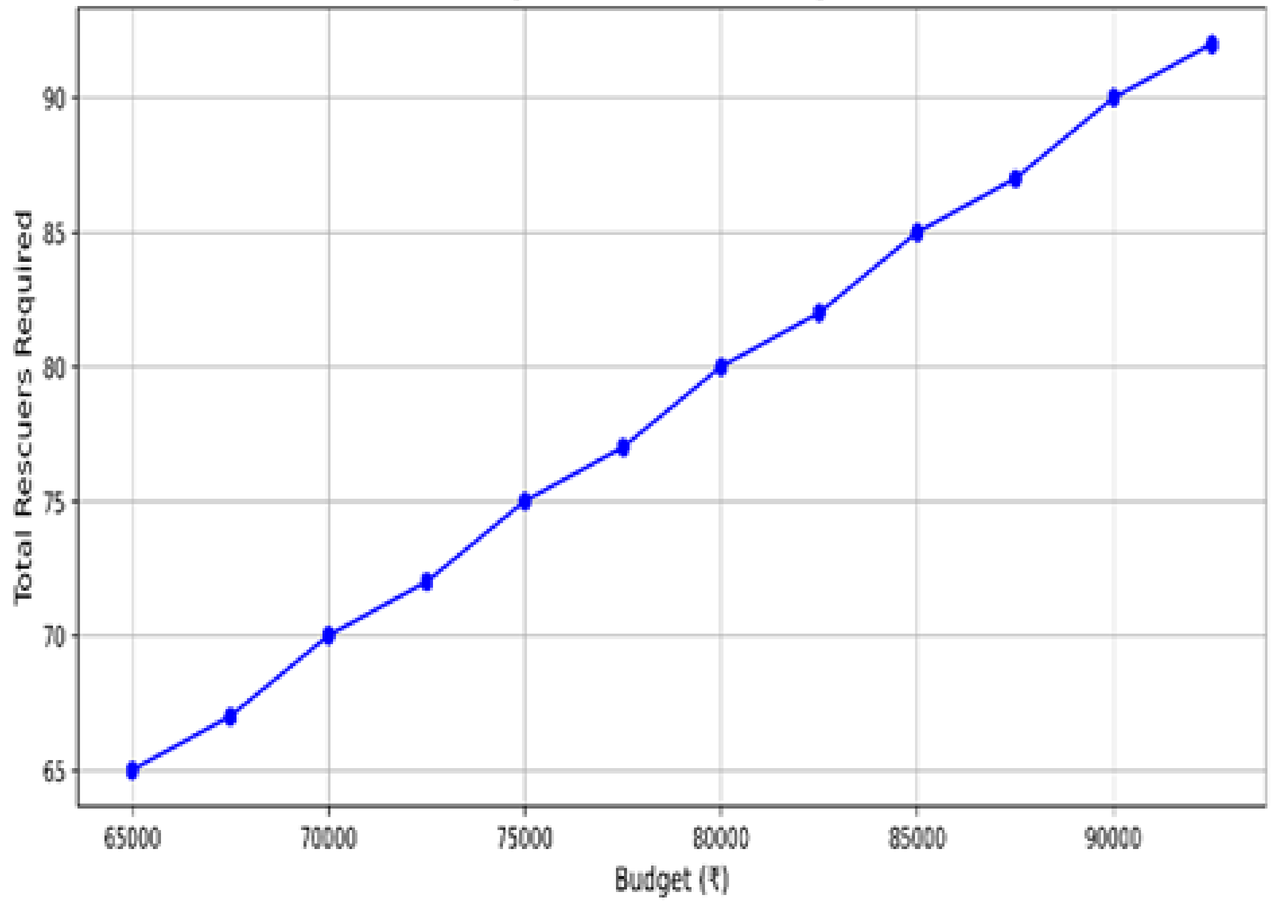


Fig: 10 Total Rescuers vs Budget

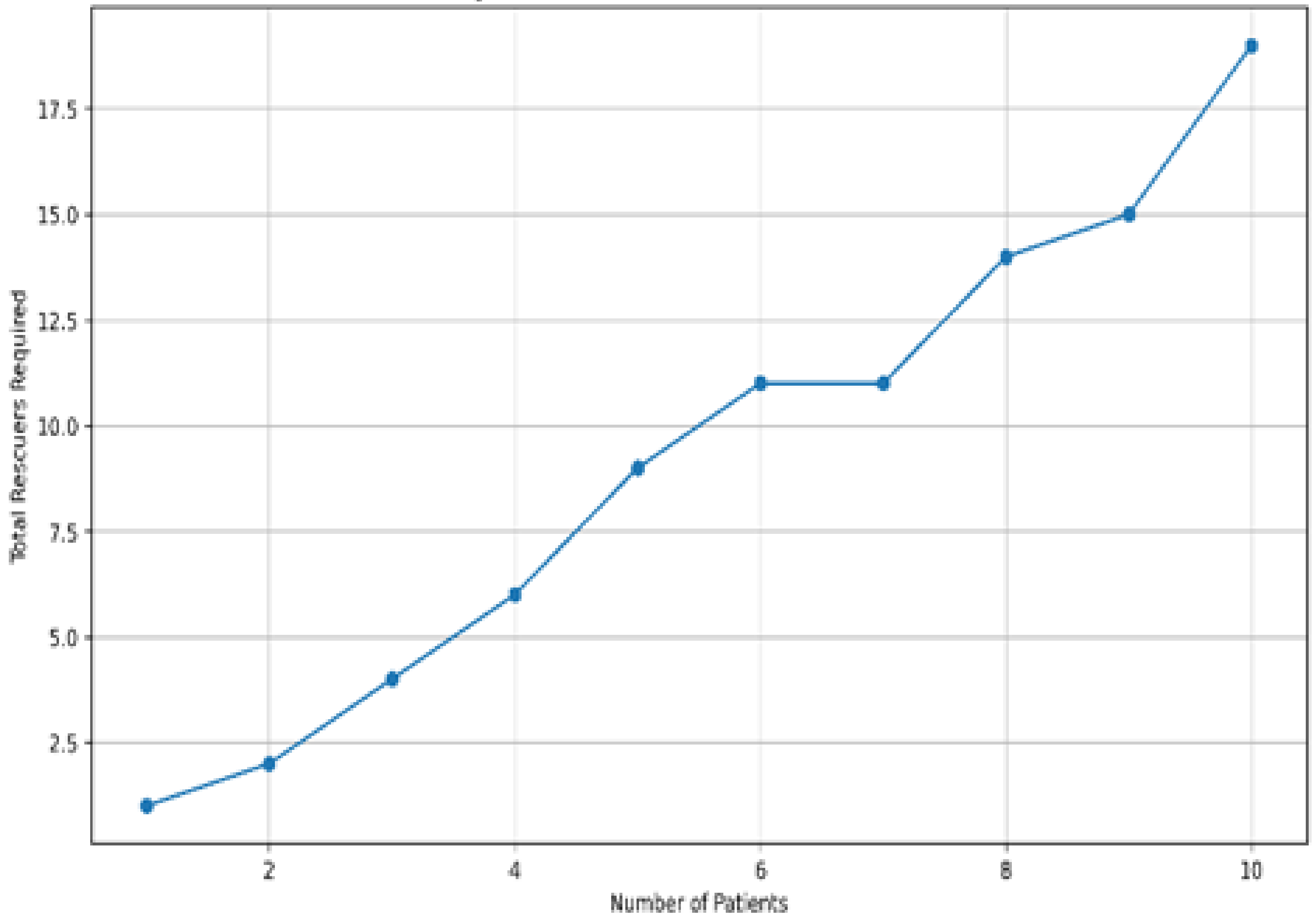


Fig:11 Total Rescuers vs Number of Patients

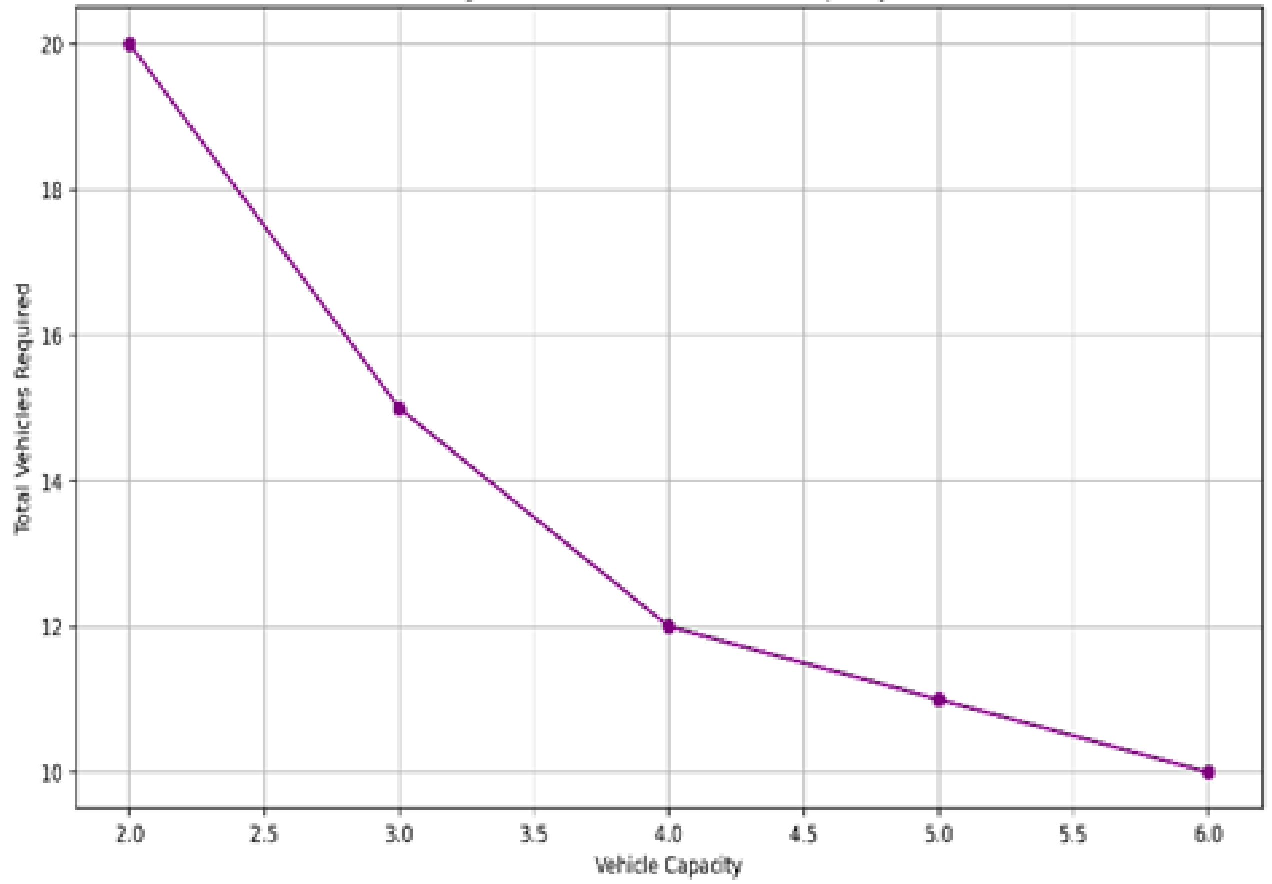


Fig: 12 Total Vehicles vs Vehicle Capacity

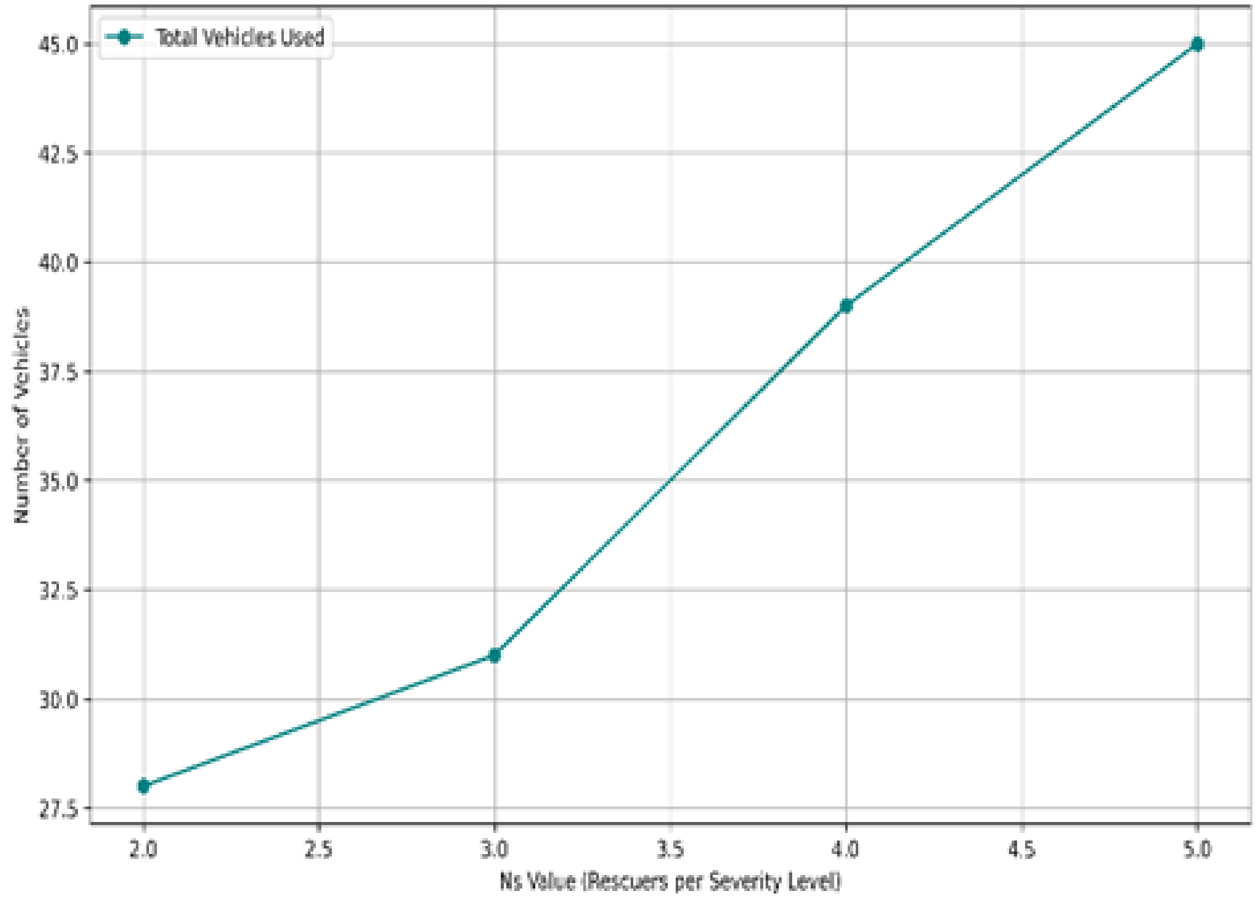


Fig 13 : Vehicles vs Ns Value

# Key Takeaways

- Integrating ML, optimization, and MCDM enables smarter disaster response planning.
- The Mizoram case showed effective balance of distance, cost, and resource allocation under realistic constraints.
- Future improvements need GIS data, dynamic conditions, and validation with real disasters (e.g., Cyclone Remal 2024) for real-world use.

# References


- Behl, A. and Dutta, P. (2019). Humanitarian supply chain management: a thematic literature review and future directions of research. Annals of Operations Research, 1001–1044.
- Choudhary, A. G. (2023). Designing a post-disaster humanitarian supply chain using machine learning and multi-criteria decision-making techniques.
- Brintrup, A. P. (2020). Supply chain data analytics for predicting supplier disruptions: a case study in complex asset manufacturing. International Journal of Production Research, 3330–3341.
- Budak, A. K. (2020). Real-time location systems selection by using a fuzzy MCDM approach: an application in humanitarian relief logistics. Applied Soft Computing, 106322.

# Thank you