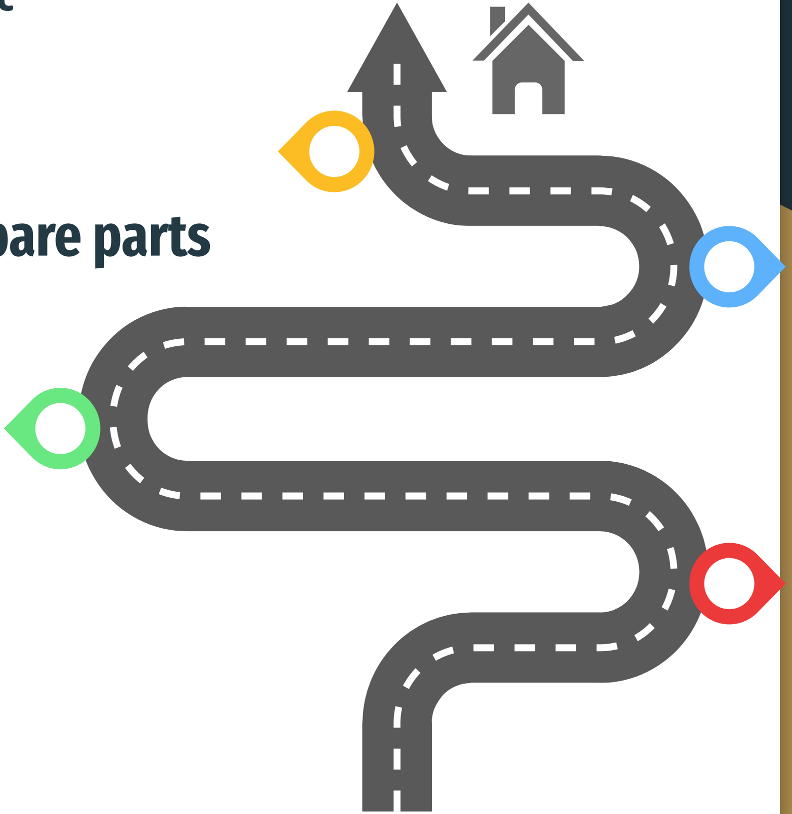
B.Tech Project

Inventory and distribution routing for a multi-product supply chain model for spare parts in Automobile company



Under the supervision of

Prof. Biswajit Mahanty

Mohammed Raafid M V | 20MF3IM33

Introduction

- Inventory management and Transportation are vital for supply chain efficiency.
- It ensures seamless movement of products from production to point of sale.
- Our core aim is having the right products available at the right place and time
- Enhancing efficiency, reducing costs, and maximizing operational potential

Problem Statement

- Consider a Manufacturing Plant at **Hosur** with **8** Distribution center from **Tamil Nadu**
- Let us consider the number of parts are **5 parts** and, **4 weeks** period
- Focus on 5 parts that are being supplied to the distribution centers via **5** transportation companies
- Truck capacity **7 Tonne** truck is used by all the vehicles from different transporters

Locations

- Main Manufacturing Plant is at **Hosur**

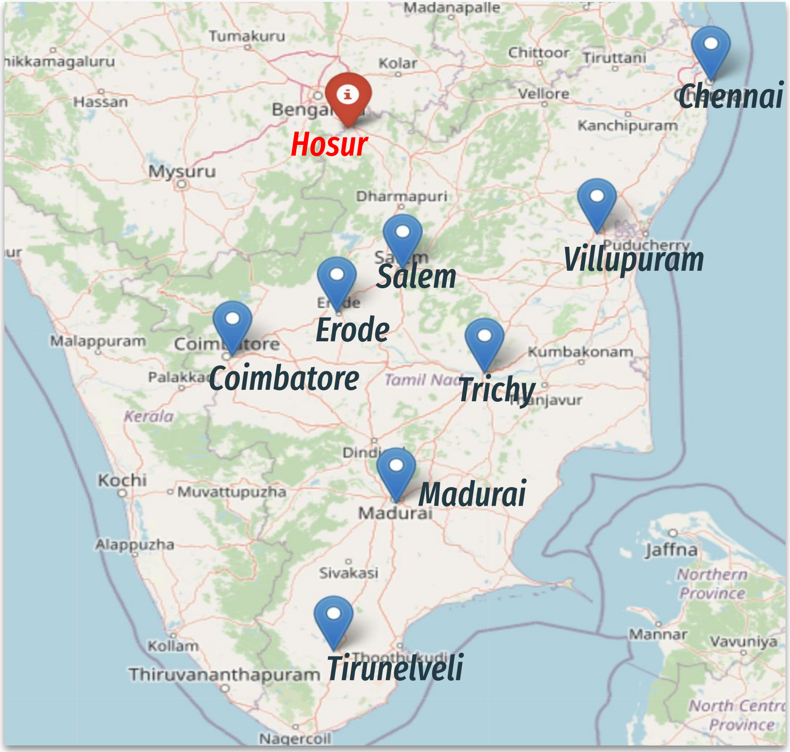
S.NO	NAME	CITY	LATITUDE	LONGITUDE
1	POPULAR PRIVATE LIMITED (*DC1*)	CHENNAI	13.06486	80.26754
2	TBF AUTO SOLUTIONS (*DC2*)	SALEM	11.6489	78.1591
3	POPULAR AUTO DISTRIBUTORS (*DC3*)	MADURAI	9.8784	78.1149
4	RKS AUTO AGENCIES (*DC4*)	ERODE	11.3323	77.7037
5	MVR AUTO SOLUTIONS (*DC5*)	TRICHY	10.8606	78.7121
6	XMR ENTERPRISES (*DC6*)	TIRUNELVELI	8.7555	77.6883
7	SPH ENTERPRISES LLP (*DC7*)	VILLUPURAM	11.925	79.4836
8	ADR ENTERPRISES (*DC8*)	COIMBATORE	10.998	76.99

Data

Considering 1 Distribution Center the demand data for DC1 is:

Distributor 1	Week 1	Week 2	Week 3	Week 4	Total
Part 1	16	16	15	15	62
Part 2	249	263	248	300	1060
Part 3	226	246	226	290	988
Part 4	376	418	350	459	1603
Part 5	28	28	23	25	104

Table 1



- **Inventory holding capacity** and **cost** for Distributor center 1 during all time periods in table 2

- **Transportation charges** and **Number of trucks available** for service based on destination to distributor center 1 in table 3

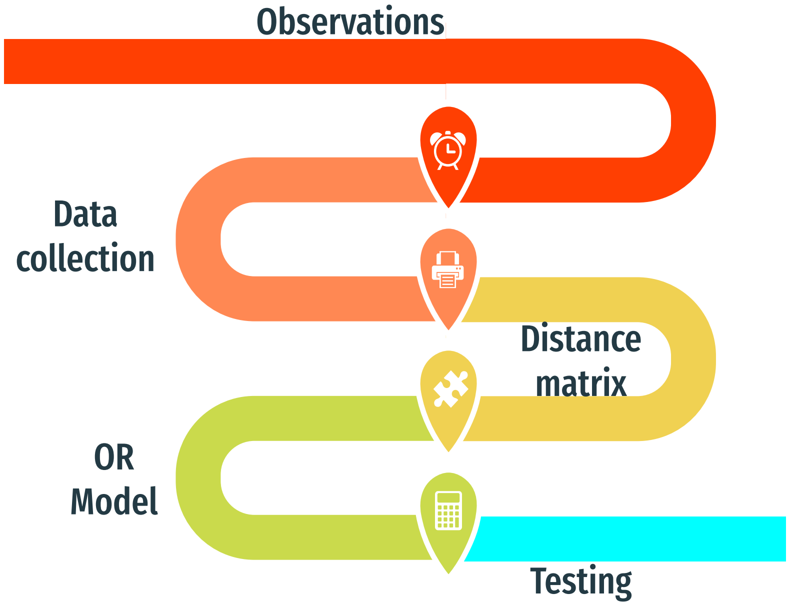
DC1	Inventory Capacity	Invetory Holding Cost	Part Weight (KG)
Part 1	967	91	2.575
Part 2	725	84	0.682
Part 3	258	6	0.287
Part 4	645	41	0.929
Part 5	226	7	0.510

Table 2

	Part 1	Part 2	Part 3	Part 4	Part 5	Number of trucks available
Transporter 1	19.67	18.2	2.19	7.1	2.06	2
Transporter 2	17.18	15.89	1.91	6.2	1.8	5
Transporter 3	19.13	17.7	2.13	6.9	2.01	4
Transporter 4	9.89	9.15	1.1	3.57	1.04	3
Transporter 5	38.63	35.73	4.31	13.94	4.05	3

Table 3

Methodology



- **Observation**
  Understanding the current model, its objectives, and its relevance to recent industrial practices and trends is essential
- **Preliminary data collection**
  Gathering preliminary data is crucial. This data will be used for calculations and fed into the model.
- **Distance matrix procurement**
  Google API used, for distance matrix must be collected to accurately calculate transportation distances between various locations.
- **OR Model development**
  Capturing the objectives and converting operational conditions into constraints effectively.
- **Testing**
- **Full-scale implementation**
- **Conclusion and future scope**

Objective function, Decision variable and Constraints

- Decision variables (with notations)

- **prod_plant('plant', p,w)** - Production quantity of **part p** at the plant during the week w
- **transport_vars(i,p,t,w)** - Quantity delivered from plant to Distribution **center i**, part p, transportation **company t**, during **week**
- **inventory_dist2('plant',p,w)**- Inventory levels of products needed to be maintained at the **plant** of part p during week w
- **inventory_dist1(i,p,w)**- Inventory levels of part p that is needed to be maintained at the distribution center i during week w

Objective function:

Minimize prob = transport_vars * cost_price + inventory_dist *inventory_holding_cost1 +inventory_dist2 * inventory_holding_cost +prod_plant * prod_price

Constraints:

- **Demand**: transport_vars >= demand
- **Inventory**: For w=0, inventory_dist = initial_inventory,
  Otherwise, inventory_dist = inventory_dist - demand + transport_vars
- inventory_dist >= 0

- **Transportation**: transport_vars * weight_per_unit <= transport_capacity * truck_capacity
- **Production**: transport_vars <= production_capacity
- **Production Matching**: transport_vars = prod_plant
- **Inventory Capacity:**inventory_dist <= inventory_capacity
- inventory_dist >= inventory_min

Formulation:

Objective Function and **Constraints** are devised to yield an effective model for supply chain management decision-making, ensuring practical and **optimal** outcomes.

Implementation in Solver:

Using **Python** with the library **PuLP** (Python Linear Programming)

Code Link: https://colab.research.google.com/drive/

Result

- Total cost after optimization is **Rs. 1,58,394.38**
- The output from the solver is in figure 1
- Inventory_(0,_'part1',_'week2') = 40.0

  It indicates Inventory stored in Distribution Center 1 for part 1 in week 2 is 40

- Transport_(0,_'plant',_'part2',_'comp4',_'week1') = 249.0

  The quantity of part 2 to transported from plant to distribution center 1 by transporter company 4 in week 1 is 249

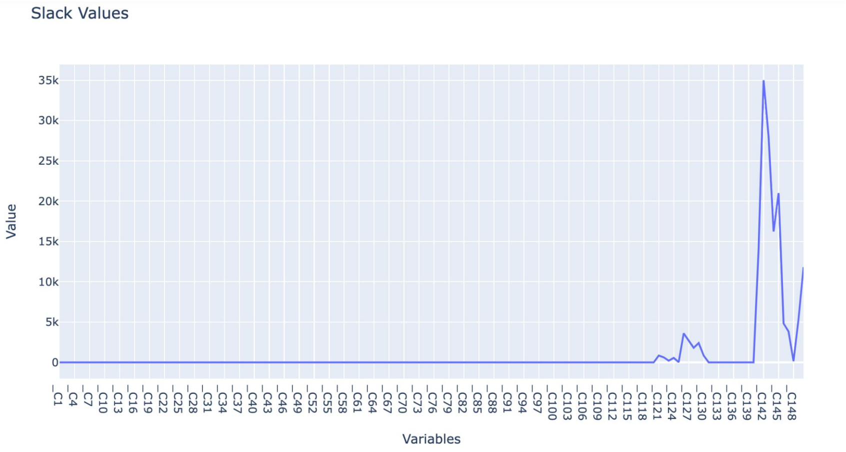
```
Total cost: 158394.37999999998
Inventory_(0,_'part1',_'week2') = 40.0
Inventory_(0,_'part1',_'week3') = 40.0
Inventory_(0,_'part1',_'week4') = 40.0
Inventory_(0,_'part2',_'week2') = 33.0
Inventory_(0,_'part2',_'week3') = 33.0
Inventory_(0,_'part2',_'week4') = 34.0
Inventory_(0,_'part3',_'week2') = 16.0
Inventory_(0,_'part3',_'week3') = 17.0
Inventory_(0,_'part3',_'week4') = 17.0
Inventory_(0,_'part4',_'week2') = 30.0
Inventory_(0,_'part4',_'week3') = 30.0
Inventory_(0,_'part4',_'week4') = 30.0
Inventory_(0,_'part5',_'week2') = 66.0
Inventory_(0,_'part5',_'week3') = 67.0
Inventory_(0,_'part5',_'week4') = 67.0
Transport_(0,_'plant',_'part1',_'comp4',_'week1') = 16.0
Transport_(0,_'plant',_'part1',_'comp4',_'week2') = 56.0
Transport_(0,_'plant',_'part1',_'comp4',_'week3') = 15.0
Transport_(0,_'plant',_'part1',_'comp4',_'week4') = 15.0
Transport_(0,_'plant',_'part2',_'comp4',_'week1') = 249.0
Transport_(0,_'plant',_'part2',_'comp4',_'week2') = 296.0
Transport_(0,_'plant',_'part2',_'comp4',_'week3') = 248.0
Transport_(0,_'plant',_'part2',_'comp4',_'week4') = 301.0
Transport_(0,_'plant',_'part3',_'comp4',_'week1') = 226.0
Transport_(0,_'plant',_'part3',_'comp4',_'week2') = 262.0
Transport_(0,_'plant',_'part3',_'comp4',_'week3') = 227.0
Transport_(0,_'plant',_'part3',_'comp4',_'week4') = 290.0
Transport_(0,_'plant',_'part4',_'comp4',_'week1') = 376.0
Transport_(0,_'plant',_'part4',_'comp4',_'week2') = 448.0
Transport_(0,_'plant',_'part4',_'comp4',_'week3') = 350.0
Transport_(0,_'plant',_'part4',_'comp4',_'week4') = 459.0
Transport_(0,_'plant',_'part5',_'comp4',_'week1') = 28.0
Transport_(0,_'plant',_'part5',_'comp4',_'week2') = 94.0
```

Figure 1

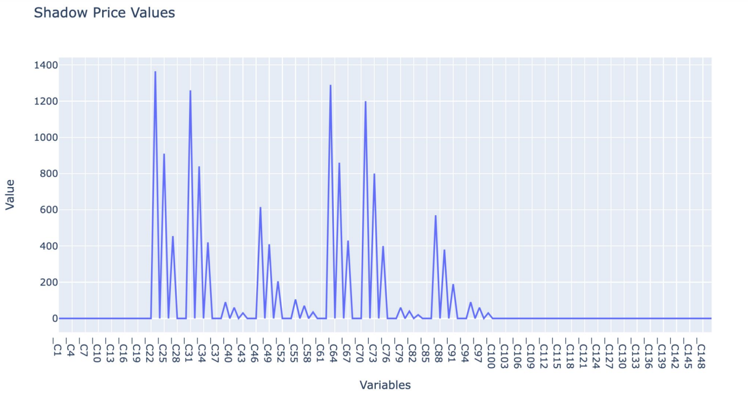
Sensitivity Analysis

160 constraints only **62** constraints are **slack** variables exhibited non zero values

Shadow Price is how much the objective function value would change for each unit in the right-hand side of a constraint



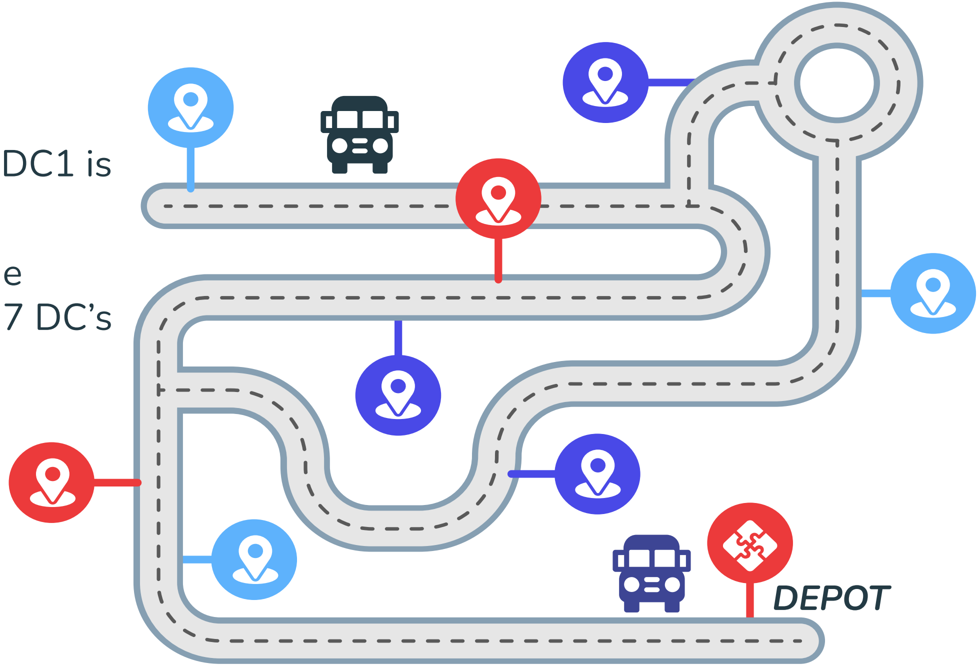

```
Total number of constraints: 160
Number of slack variables greater than zero: 29
Number of slack variables lesser than zero: 23
```




```
Total number of variables: 160
Number of points of shadow price greater than 1: 30
```

Capacitated Vehicle Routing

- Total weight of the parts transported to DC1 is **2300 KG**
- Total 8 DC's are there, Assume that same quantity is transported to the remaining 7 DC's
- Total truck capacity is **7000 KG**
- There are **5 transportes**, transport **company 4** has **minimum cost** of transportation there is **3 trucks** available with them



Distance Matrix Using Google Maps API

	Hosur	Chennai	Salem	Madurai	Erode	Trichy	Tirunelveli	Villupuram	Coimbatore
Hosur	0	316	172	413	235	301	546	237	331
Chennai	316	0	339	463	404	317	615	165	500
Salem	151	338	0	241	67	129	374	174	163
Madurai	387	463	236	0	199	146	156	299	208
Erode	214	406	69	205	0	140	338	242	95
Trichy	281	317	130	146	140	0	299	153	209
Tirunelveli	525	615	374	156	338	299	0	451	346
Villupuram	239	165	174	299	240	152	451	0	336
Coimbatore	310	502	165	217	98	208	351	338	0

Routing

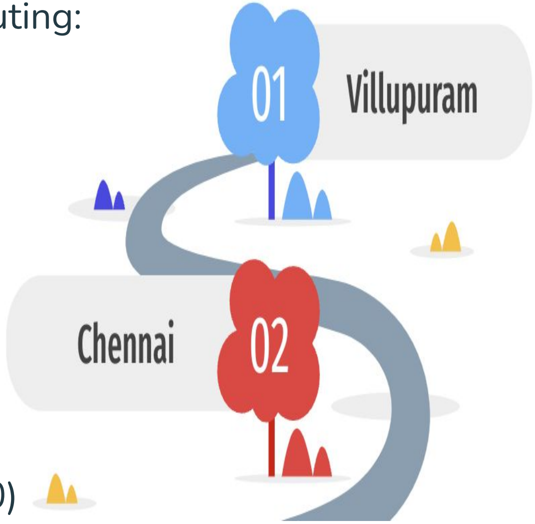
Route 1

- Every Route starts and ends at the main plant Hosur
- Using CVRP constraints with Google OR Tool the result of routing:

```
Route for vehicle 2:
 0 Load(0) ->  7 Load(2300) ->  1 Load(4600) ->  0 Load(4600)
Distance of the route: 718050m
Load of the route: 4600

Route for vehicle 3:
 0 Load(0) ->  5 Load(2300) ->  3 Load(4600) ->  6 Load(6900) ->  0 Load(6900)
Distance of the route: 1130661m
Load of the route: 6900

Route for vehicle 4:
 0 Load(0) ->  2 Load(2300) ->  4 Load(4600) ->  8 Load(6900) ->  0 Load(6900)
Distance of the route: 645695m
Load of the route: 6900
```

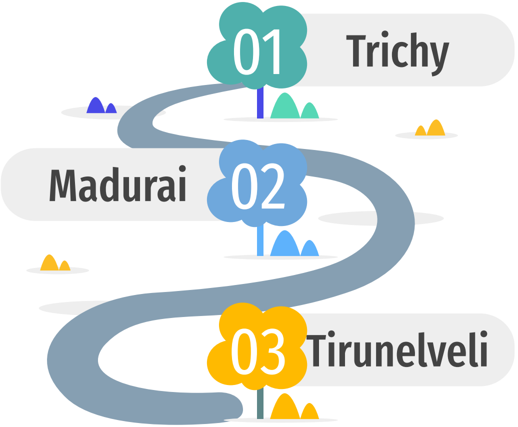


- It shows the route followed by each truck
  0 Load(0) -> 7 Load(2300) -> 1 Load(4600) -> 0 Load(4600)

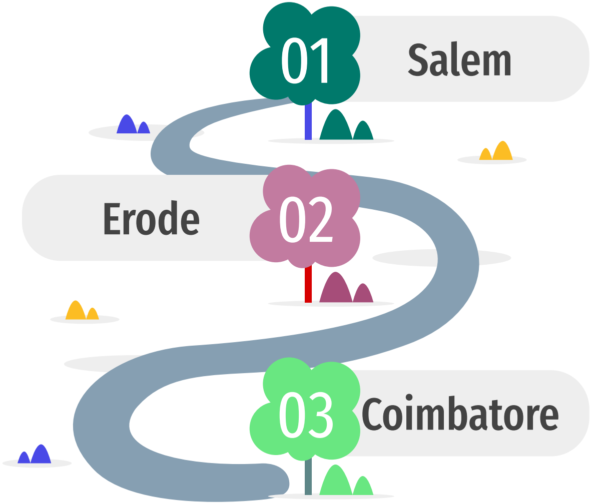
- Indicates Start from Hosur -> Villupuram -> Chennai -> Hosur

Conclusion

Route 2



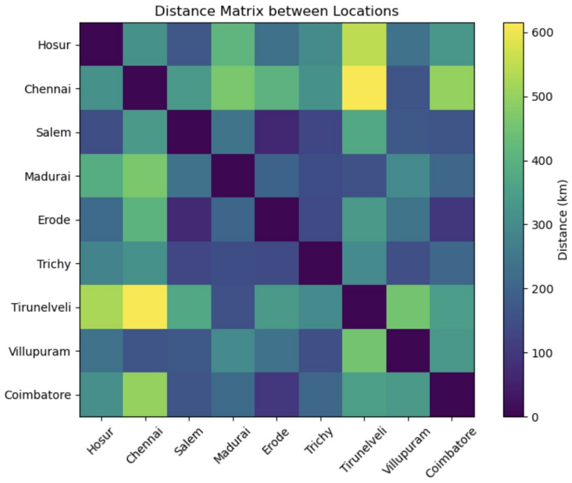
Route 3



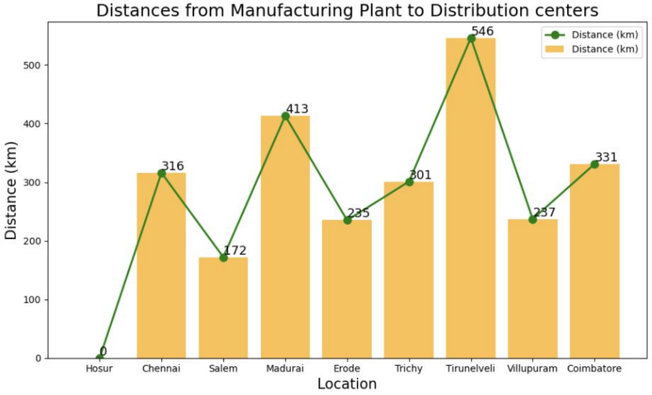
Due to route optimization, the total distance traveled significantly decreased from **3554.0 km** to **2497.4 km**, representing a **reduction of 1056.6 km**, or distance reduce by **29.73%**

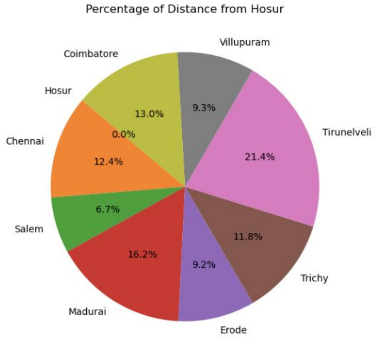
Data Visualisation

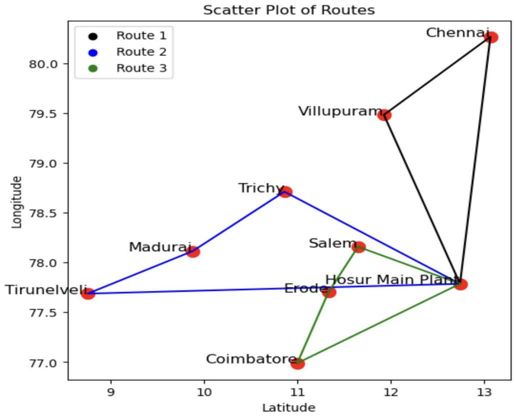
Visual Representation of distances from **Hosur** to the 8 distribution center



Heat Map of Distance Matrix

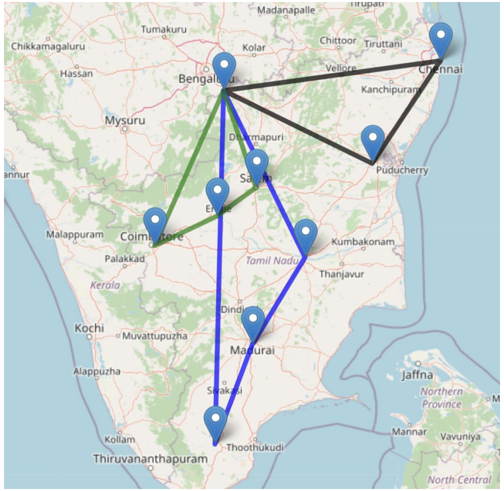




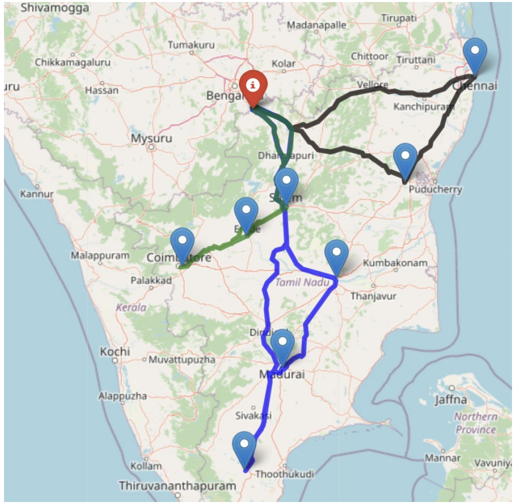


Scatter Plot of routes followed by the 3 trucks

- Total distance traveled was estimated through Haversine distance **2052.22 km**
- Actual distance travelled is **2497.408 km** calculated by google maps with distance matrix
- Rectified the error percentage in the distances are **17.79%**



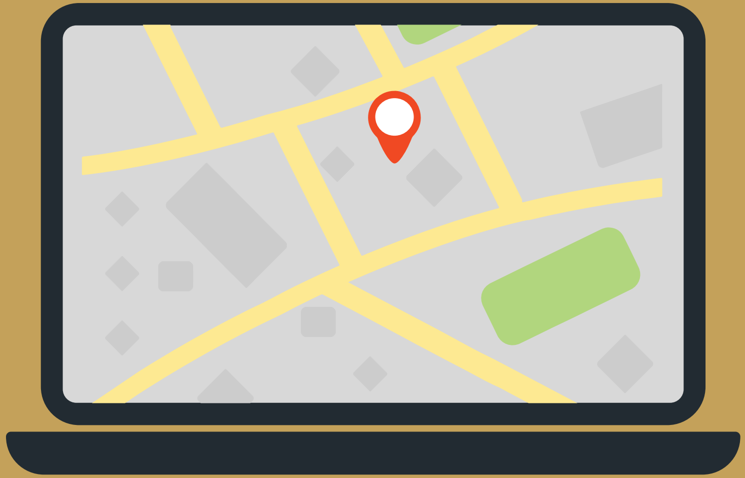
Distance calculated by Haversine distance



Route by Google Maps

Future Scope

- If one more plant was opened in Tamil Nadu due to the high demand of products, finding the proper location by reducing transportation costs and inventory costs
- Time window constraints can be added for loading and unloading for efficiently reaching the product
- Identify the traffic and tame taken for the travel in the routes where truck travel using Google Maps
- Different sizes of items can packed into a fixed volume of bins, in a way that minimizes the number of bins used by utilising the volume truck.



THANK YOU