**Project No: 2**

**Project Title: Traveling Salesman Problem**

**Prepared by: Rose Mary Jose**

**Date: 01-07-23**

Prepared for “Thoughtware Training Private Limited”, under Guidance of its CEO Mr. Pattabhi Raman. The project will be subject to further research, modification and exclusive use of “Thoughtware Training Private Limited”

**Objective of the project**

The Traveling Salesman Problem (TSP) is a classic optimization problem in computer science and operations research. It is a combinatorial problem that seeks to find the shortest possible route for a salesman who must visit a given set of cities and return to the starting city, visiting each city exactly once.

the TSP can be defined as follows:

Given a set of cities and the distances between each pair of cities, the objective is to find the shortest possible route that starts from a particular city, visits each remaining city exactly once, and returns to the starting city.

The TSP is a well-known NP-hard problem, meaning that there is no known efficient algorithm that can solve all instances of the problem optimally in polynomial time. Therefore, finding an exact solution to large TSP instances is computationally challenging.

The TSP has numerous applications in logistics, transportation, network design, and other fields where finding an optimal or near-optimal route is essential.

**Dataset**

For finding the optimal route travelled by a salesman a dataset with the cities to be visited along with the distance matrix between those cities is created as the dataset. The created dataset consists of 14 popular cities in South India. The distance between each city is considered and the resulting distance matrix is displayed below.

import pandas as pd  
import numpy as np  
from python\_tsp.exact import solve\_tsp\_dynamic\_programming  
import io

## reading dataset  
data = pd.read\_excel('TSP\_dataset.xlsx')  
data

location Bengaluru Chennai Kochi Trivandrum Salem Pondicherry   
0 Bengaluru 0.0 345.7 547.4 735.9 201.1 313.2   
1 Chennai 345.7 0.0 690.1 778.5 344.7 150.8   
2 Kochi 547.4 690.1 0.0 199.3 349.2 568.6   
3 Trivandrum 735.9 778.5 199.3 0.0 537.0 657.9   
4 Salem 201.1 344.7 349.2 537.0 0.0 222.4   
5 Pondicherry 313.2 150.8 568.6 657.9 222.4 0.0   
6 Madurai 433.5 455.6 265.1 318.5 233.5 332.3   
7 Mysore 143.2 480.9 394.1 680.9 335.4 446.0   
8 Coimbatore 362.9 505.7 189.3 387.4 167.0 382.3   
9 Mangaluru 350.7 704.0 418.2 610.0 558.5 669.1   
10 Vellore 211.3 137.7 575.6 766.3 229.3 153.6   
11 Kozhikode 355.9 683.9 182.4 374.2 345.3 560.5   
12 Shivamogga 306.4 659.7 642.0 1051.2 514.2 624.8   
13 Hubballi 412.1 765.4 861.7 1156.9 619.9 730.5   
  
 Madurai Mysore Coimbatore Mangaluru Vellore Kozhikode Shivamogga   
0 433.5 143.2 362.9 350.7 211.3 355.9 306.4   
1 455.6 480.9 505.7 704.0 137.7 683.9 659.7   
2 265.1 394.1 189.3 418.2 575.6 182.4 642.0   
3 318.5 680.9 387.4 610.0 766.3 374.2 1051.2   
4 233.5 335.4 167.0 558.5 229.3 345.3 514.2   
5 332.3 446.0 382.3 669.1 153.6 560.5 624.8   
6 0.0 378.1 211.4 636.3 400.7 359.7 748.4   
7 378.1 0.0 198.8 255.2 352.6 209.6 243.5   
8 211.4 198.8 0.0 457.0 391.5 183.2 446.8   
9 636.3 255.2 457.0 0.0 560.4 232.6 200.8   
10 400.7 352.6 391.5 560.4 0.0 558.6 524.7   
11 359.7 209.6 183.2 232.6 558.6 0.0 403.7   
12 748.4 243.5 446.8 200.8 524.7 403.7 0.0   
13 854.1 496.5 666.4 361.8 630.4 599.9 210.7   
  
 Hubballi   
0 412.1   
1 765.4   
2 861.7   
3 1156.9   
4 619.9   
5 730.5   
6 854.1   
7 496.5   
8 666.4   
9 361.8   
10 630.4   
11 599.9   
12 210.7   
13 0.0

The data set considered is a distance matrix of shape 14 x 14.

**Model Selection**

The objective of the project is to find the shortest possible route that starts from a particular city, visits each remaining city exactly once, and returns to the starting city. TSP is an optimization problem in supply chain management. Optimization in supply chain management refers to the process of finding the best possible solution or configuration that maximized efficiency, minimizes costs, and distances, improves overall performance within the supply chain network. It involves using mathematical modelling, algorithms, and analytical techniques to optimize various aspects of the supply chain, including inventory levels, transportation routes, production schedules, facility locations, and resource allocation. There are different types of optimization problem in supply chain management such as, cost minimization, service level maximization, resource utilization, risk mitigation, sustainability, TSP, etc.

To achieve optimization in supply chain management, various techniques and tools are employed, such as linear programming, network optimization, simulation, predictive analytics, and machine learning. These methods help in analyzing large amount of data, considering multiple constraints and objectives, and finding the best possible solutions that optimize the supply chain operations.

Here the problem is to find the shortest possible route that starts from a particular city, visits each remaining city exactly once, and returns to the starting city. In order to do the optimization firstly the mathematical formulation is to be done.

**Mathematical Formulation**

Mathematical formulation refers to the process of representing the problem mathematically using variables, constraints, and an objective function. From the type of objective function optimization technique is chosen. There are different types of optimization technique such as linear programming, integer programming, non-linear programming etc.

Decision variables: these are the variables that represent the quantities or decisions to be optimized. They are typically denoted by symbols and can take on specific values later.

Constraints: they represent the limitations or restrictions on the decision variables. They define the feasible region of solutions. Constraints can be mathematical equations or inequalities that the decision variables must satisfy.

Objective function: it defines the goal of the optimization problem. It quantifies the measure to be maximized or minimized. It is typically a mathematical expression involving the decision variables.

Parameters: they are the fixed value given in the problem and influence the optimization process. These are known values and do not change during the optimization.

The objective of the TSP is:

min

where,

=

The constraint of the tsp problem is that one city should only be visited once. The salesman should be back to the origin city.

**Optimization technique**

In this project to solve the TSP ‘python-tsp’ library is used. The ‘python-tsp’ is a Python library that provides tools and algorithms for solving the Traveling Salesperson Problem (TSP). The TSP is a classic optimization problem in computer science and operations research, where the goal is to find the shortest possible route that visits a set of cities and returns to the starting city.

The python-tsp library offers various algorithms and methods to solve the TSP, including exact algorithms like dynamic programming and branch-and-bound, as well as approximate algorithms like nearest neighbor and 2-opt optimization. It provides a convenient and easy-to-use interface for defining TSP instances, solving them, and obtaining optimal or near-optimal solutions.

optimal\_route, distance = solve\_tsp\_dynamic\_programming(distance\_matrix)

print(optimal\_route)

In this project solve\_tsp\_dynamic\_programming function from the ‘python\_tsp’ library is used for finding the best route for traveling. The solve\_tsp\_dynamic\_programming function takes the distance matrix as input and returns the optimal route as a list of city indices and the distance covered is also calculated. The optimal route represents the order in which the cities should be visited to minimize the total distance. The total distance the salesman travel through the optimal route is found in KMs.

The optimal route for the salesman to travel is:

[0, 7, 12, 13, 9, 11, 2, 3, 6, 8, 4, 5, 1, 10]

The order of the city the salesman need to travel:

Bengaluru-> Mysore-> Shivamogga-> Hubballi-> Mangalore-> Kozhikode-> Kochi-> Trivandrum-> Madurai-> Coimbatore-> Salem-> Pondicherry-> Chennai-> Vellore-> Bengaluru.

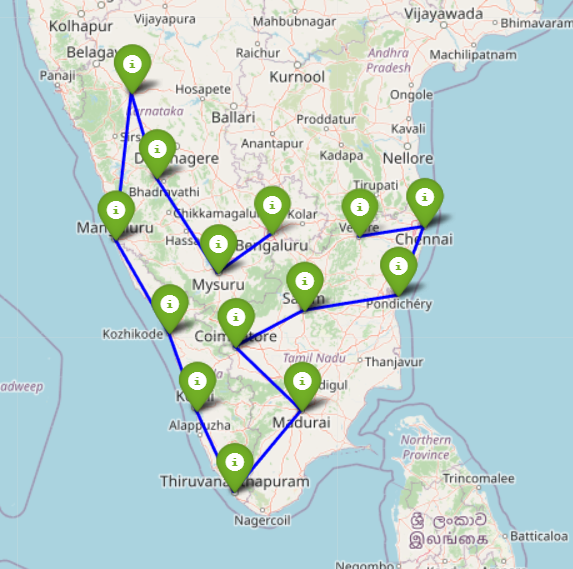
The minimum distance to be traveled is:

The minimum distance taken for covering all cities: 2992.6 KM

The optimal route is visually represented using Folium library from python.

location\_df = pd.read\_excel("TSP\_location\_coordinates.xlsx")  
location\_df = location\_df.set\_index('location')

import folium  
from geopy.geocoders import Nominatim  
  
# Create a map centered at a specific location (e.g., first city)  
geolocator = Nominatim(user\_agent="tsp\_map")  
loc = geolocator.geocode(optimal\_routes[0]) # Get the coordinates of the first city  
m = folium.Map(location=[loc.latitude, loc.longitude], zoom\_start=6)  
  
# Assuming you have a DataFrame called 'location\_df' containing the latitude and longitude  
# Columns 'latitude' and 'longitude' in the DataFrame correspond to the coordinates  
for i in range(len(optimal\_routes) - 1):  
 city1 = optimal\_routes[i]  
 city2 = optimal\_routes[i + 1]  
  
 # Get the coordinates of the cities from the DataFrame  
 location1 = location\_df.loc[city1]  
 location2 = location\_df.loc[city2]  
  
 folium.PolyLine(  
 locations=[(location1['latitude'], location1['longitude']), (location2['latitude'], location2['longitude'])],  
 color='blue',  
 weight=2.5,  
 opacity=1  
 ).add\_to(m)  
m



**Conclusion**

Solving the Traveling Salesman Problem (TSP) using dynamic programming offers an efficient and optimal solution for finding the shortest possible route that visits all given cities exactly once and returns to the starting city. Overall, dynamic programming is a powerful technique for solving combinatorial optimization problems like the TSP and provides a reliable method to find the optimal solution for relatively small instances of the problem. For larger instances, other algorithms like genetic algorithms, ant colony optimization, or simulated annealing may be employed to find near-optimal solutions efficiently.