MACHINE LEARNING AND MATHEMATICAL PROGRAMMING BASED HYBRID SOLUTION PROPOSAL FOR CAPACITATED VEHICLE ROUTING PROBLEM

Master's Thesis

Özgür SANLI

Eskişehir 2022

MACHINE LEARNING AND MATHEMATICAL PROGRAMMING BASED HYBRID SOLUTION PROPOSAL FOR CAPACITATED VEHICLE ROUTING PROBLEM

Özgür SANLI

Master's Thesis

Department of Industry Engineering

Programme in Institute of Graduate

Supervisor: Asst. Prof. Dr. Zühal KARTAL

Eskişehir

Eskişehir Technical University

Institute of Graduate Programs

July 2022

01/07/2022

SUPERVISOR APPROVAL

Master's student Özgür SANLI, whom I supervise, has completed this thesis titled MACHINE LEARNING AND MATHEMATICAL PROGRAMMING BASED HYBRID SOLUTION PROPOSAL FOR CAPACITATED VEHICLE ROUTING PROBLEM. According to my inspections, the work is scientifically and ethically appropriate for the student to the thesis defense exam.

Supervisor

Asst. Prof. Dr. Zühal KARTAL

ABSTRACT

MACHINE LEARNING AND MATHEMATICAL PROGRAMMING BASED HYBRID SOLUTION PROPOSAL FOR CAPACITATED VEHICLE ROUTING PROBLEM

Özgür SANLI

Department of Industry Engineering

Programme in Institute of Graduate

Eskişehir Technical University, Institute of Graduate Programs, July 2022

Supervisor: Asst. Prof. Dr. Zühal KARTAL

In this study, a three-stage approach which hybridizes machine learning techniques and mathematical programming formulations, is proposed for the solution of capacitated vehicle routing problem (CVRP). In order to solve CVRP, in the first stage, it was decided the nodes to be assigned to which vehicles via machine learning algorithms, then in the second stage it was ensured that the resulting clusters' total demand amount do not exceed the vehicle capacity using a method, which is called capacity balancing algorithm. In the third stage, the vehicle started from the depot and visits all the assigned nodes to find the shortest (minimum as an alternative) travelled distance by using the traveling salesman problem (TSP) mathematical model. The final solution of the CVRP has been formed by combining all TSP routes. The machine learning algorithms that are used in this study are for supervised learning category; K-Nearest Neighborhood (K-NN) and Logistic Regression (LR) algorithms and for unsupervised learning category; K-Means algorithm. For the proposed approach, sensitivity analyzes were carried out using different datasets from the literature with a different number of vehicles. As a result, it has been shown that the proposed hybrid approach gives better results in most of the test problems than the solution of the mathematical model of CVRP.

Keywords: Capacitated Vehicle Routing Problem, Machine Learning and Mathematical Modelling Hybrid Approach, Logistic Regression, K-Means, K-NN.

ÖZET

KAPASITELİ ARAÇ ROTALAMA PROBLEMİ İÇİN MAKİNE ÖĞRENMESİ VE MATEMATİKSEL PROGRAMLAMA TEMELLİ HİBRİD BİR ÇÖZÜM ÖNERİSİ

Özgür SANLI

Endüstri Mühendisliği Anabilim Dalı

Lisansüstü Eğitim Bilim Dalı

Eskişehir Teknik Üniversitesi, Lisansüstü Eğitim Enstitüsü, Temmuz 2022

Danışman: Dr. Öğr. Üyesi Zühal KARTAL

Bu çalışmada, kapasiteli araç rotalama probleminin (KARP) çözümü için makine öğrenmesi teknikleri ile matematiksel programlama formülasyonlarını hibritleştiren üç aşamalı bir yaklaşım önerilmiştir. KARP'ın çözümü için ilk aşamada makine öğrenmesi algoritmaları ile düğümlerin hangi araçlara atanacağına karar verildikten sonra ikinci aşamada ortaya çıkan kümelerin toplam talep miktarının her bir aracın kapasitesini aşmaması kapasite dengeleme algoritması adı verilen bir metot tarafından garantilenmiştir. Üçüncü aşamada ise, her bir araç depodan tur oluşturmak için başlar ve gezgin satıcı problemi (GSP) matematiksel modelini kullanarak en kısa kat edilen mesafeyi bulmak için atanan tüm düğümleri ziyaret eder. KARP 'ın nihai çözümü, tüm TSP rotalarının birleştirilmesiyle oluşturulmuştur. Bu çalışmada kullanılan makine öğrenmesi algoritmaları denetimli öğrenme kategorisi altında; K-En yakın Komşuluk algoritması (K-NN) ve lojistik regresyon (LR) algoritmalarıyken; denetimsiz öğrenme kategorisi için, K-Ortalamalar (K-Means) algoritmasıdır. Önerilen yaklaşım için, farklı araç sayıları ile literatürden farklı veri setleri kullanılarak duyarlılık analizleri gerçekleştirilmiştir. Sonuç olarak önerilen hibrit yaklaşımın test problemlerinin çoğunda KARP'ın matematiksel modelinin çözümüne göre daha iyi sonuçlar verdiği gösterilmiştir.

Anahtar Sözcükler: Kapasiteli Araç Rotalama Problemi, Makine Öğrenmesi ve Matematiksel Modelleme Hibrid Yaklaşım, Lojistik Regresyon, K-Means, K-NN.

ACKNOWLEDGEMENTS

I would like to thank my supervisor Asst. Prof. Dr. Zühal Kartal for her consistent support and contribution to my thesis and for all her efforts to improve my study.

I also would like to express my sincere thanks to my parents, for supporting me all throughout my life and to my family, relatives and friends for believing in me, supporting me in every way and always being by my side. My brother Murat Sanli and my colleague Yusuf Taner Serin proofread my work, and I appreciate it.

Finally, I wish to express my deepest love and gratefulness to my wife Seda and my daughter Defne for their understanding and always make me feel their support and encouragement in the course of research and my life.

Özgür SANLI

01/07/2022

STATEMENT OF COMPLIANCE WITH ETHICAL PRINCIPLES AND RULES

I hereby truthfully declare that this thesis is an original work prepared by me; that I have behaved in accordance with the scientific ethical principles and rules throughout the stages of preparation, data collection, analysis and presentation of my work; that I have cited the sources of all the data and information that could be obtained within the scope of this study, and included these sources in the references section; and that this study has been scanned for plagiarism with "scientific plagiarism detection program" used by Eskişehir Technical University, and that "it does not have any plagiarism" whatsoever. I also declare that, if a case contrary to my declaration is detected in my work at any time, I hereby express my consent to all the ethical and legal consequences that are involved.

Özgür SANLI

CONTENTS

LIST OF TABLES

Page

LIST OF FIGURES

GLOSSARY OF SYMBOLS AND ABBREVIATIONS

CBA	:	Capacity Balancing Algorithm
CFRS	:	Cluster First Routing Second
CVRP	:	Capacitated Vehicle Routing Problem
DCVRP	:	Distance Constrained Vehicle Routing Problem
K-NN	:	K-Nearest-Neighbors
LR	:	Logistic Regression
Obj. F.	:	Objective Function
RFCS	:	Route First Cluster Second
TSP	:	Traveling Salesman Problem
VRP	:	Vehicle Routing Problem
VRPB	:	Vehicle Routing Problem with Backhauls
VRPBTW	:	VRPB with Time Windows
VRPPD	:	Vehicle Routing Problem with Pick up and Delivery
VRPPDTW	:	VRPPD with Time Windows
VRPTW	:	Vehicle Routing Problem with Time Windows

1. INTRODUCTION

Last decade, logistics and supply chain industry has caught a significant growth than it was estimated by the help of the emergence of COVID-19 pandemics. Especially scoping out the on-demand logistics market, it was $9.1 billion in 2019, now on-demand logistics market is aet to surpass $75.0 billion by 2030 [1]. It is obvious that goods transportation will be accelerated, planning and scheduling will need to be much more precise and punctual. In this situation, a 10% cost reduction in the sector will profit $ 7.5 billion. That is why it is necessary to manage logistic activities effectively. Logistic activities are mainly divided into five groups which are selection of the place of operation, storage and packaging of the product, inventory management, management of requests and orders, transportation, and distribution activities [2]. Strategic achievements in these areas improve the quality of services and reduce operating costs.

Planning of vehicles, determining distribution points and determining their routes refers to transportation and distribution activities from the strategic steps of logistics activities. This problem is known as Vehicle Routing Problem(VRP). VRP can be expressed in the form of a fleet vehicle taking the shortest route to the designated customers, services or products. VRP is one of the most examined and essential field of operation research, in literature [3].

VRP is a type of problem in which the cost of vehicles' routes is tried to be minimized after meeting the demands of customers who make requests for the same or different amounts and vehicles can be departed from one or more depots using a certain number of vehicles whose capacities are predetermined. In classical VRP, it is assumed that all vehicles return to the depot. Since the VRP is solved with the k-numbered vehicles, VRP is an improved version of traveling salesman problem (TSP) with additional constraints. For n nodes in the TSP, the number of routes to calculate is n! whereas for more than one vehicle in the VRP, the number of routes to calculate is $2^n * n!$. **Figure 1.1.** depicts the decomposed solution of VRP to TSP as an example.

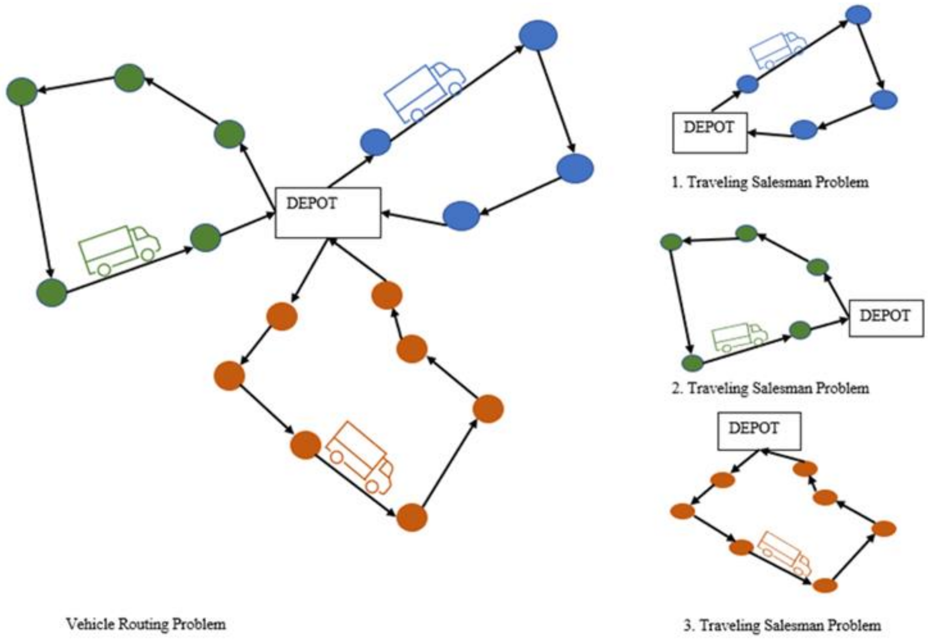


Figure 1.1. *Figural representational of 3 vehicle VRP decomposed to 3 different TSP*

In this thesis, a hybrid decomposed method based on Cluster First Routing Second (CFRS) method assisted with machine learning algorithms was proposed for capacitated (homogeneous) vehicle routing problem's solution (CVRP). The mathematical modeling formulation of the TSP was executed for solving the nodes allocated to a vehicle in a specified time to the commercial solver after the nodes (customers) to be assigned to the vehicles were defined using supervised or unsupervised machine learning methods. Machine Learning methods such as Logistic Regression (LR), K-Nearest Neighbors (K-NN), and K-Means algorithms were used to assign nodes to vehicles. After finding the nodes assigned to the vehicles, a method called Capacity Balancing Algorithm (CBA) is used to check for vehicle capacity. If vehicle capacity is exceeded, the vehicle assignments are revised (CBA). In CBA, the node with the lowest probability is selected from the nodes belonging to the vehicle whose capacity is exceeded, and this node is assigned to the vehicle with the second highest probability. If there is no capacity gap in the other selected vehicle, the assignment is made to the third vehicle. This process is continued until the vehicle capacity is not exceeded. Then, the assigned nodes of vehicles which clarified, are solved with TSP formulation by using a commercial solver. As far as known, no solution technique using machine learning algorithms and mathematical programming models with the help of such an algorithm (CBA) has been found in the literature. In the approach discussed, the nodes in the VRP are classified or clustered first using a machine learning algorithm, and then nodes that are assigned to vehicles are solved with a TSP mathematical model and the results are obtained.

In literature, it was found that after clustering, routing was performed to solve various types of vehicle routing problems. The closest studies to the method developed in this thesis are those aimed at using clustering algorithms and mathematical models together Donda and Cerda [4], and Assis et al. [5] can be given as an example. Donda and Cerda [4] proposed a 3-stage heuristic for the multi-depot heterogeneous fleet with VRPTW. According to this study, after the appropriate clusters have been found in the first stage, a clustering-based mathematical modelling formulation in the second stage performs the process of assigning clusters to vehicles. At final stage, vehicle routes are still found by the help of mathematical modelling formulation. In Assis et al. [5], the authors presented a clustering algorithm based on mathematical modelling for the operational management of crude oil supply. For the solution of CVRP, Rautela et al. [6], used a solution by clustering and then routing using the K-Means machine learning algorithm. According to Rautela et al., they use the K-Means algorithm until they find a solution that does not exceed the capacity of the vehicles, then use the cheapest link algorithm to route nodes allocated to the cars. In comparison, Geetha et al. [7], inspired by the capacity vehicle routing problem, proposed a K-Means algorithm that evaluates both customer demands and distance to the cluster when allocating a node to a vehicle. According to Alesiani et al. [8] in their study, after finding a reduced number of cluster center's using the K-Means algorithm, they used the TSPY-Python program package for solving the routing of these centers. Mostafa and Eltawil's [9] investigation with the K-means algorithm is the most similar solution method to this thesis. The authors solved the heterogeneous fleet size vehicle routing problem by mathematical modelling formulation of TSP after converting the K-Means algorithm to a form that assigns a similar number of nodes to each set. However, the study did not include what kind of an approach is followed if the capacity limit is exceeded. In this thesis, along with machine learning algorithms such as K-Means, LR, K-NN algorithm, a CBA is presented to prevent exceeding the capacity of vehicles. The interested reader is referred to Czuba and Pierzchala [10] for a survey of the solution of VRP using machine learning approaches. As a result, this study will fill a gap in the literature by classification and clustering supervised and unsupervised machine learning approaches with the use of an algorithm that prevents overcapacity, and then concluding mathematical models with commercial solvers.

The following sections of the thesis are organized as follows. In Chapter 2, the definition, variants, and solution methods of VRP is discussed. In Chapter 3, the methodology of proposed model is explained with the description and inner workings of machine learning algorithms and mathematical models that are used in solution of CVRP. In Chapter 4, numerical results of the proposed model are declared. In Chapter 5, the conclusion and discussion about future works of model are represented.

2. VEHICLE ROUTING PROBLEM (VRP)

The VRP was first presented to the literature in 1959 with the problem of "Truck Shipment" posed by Dantzig and Ramser [11]. In this problem, a solution was sought to the issue of how to distribute gasoline tankers leaving a depot to gas stations in the shortest way. In addition, for the first time, the mathematical programming formulation of the problem is presented in that study. A few years after this proposal, Clarke and Wright developed Dantzig and Ramser's approach in their study, proposing an effective greedy algorithm (1964) [12]. After these two studies, hundreds of models and algorithms related to many different VRP versions and optimal and approximate algorithms to solve this problem were proposed. For interested readers, the study of Laporte and Osman [13] and the study of Toth and Vigo [14] can be suggested for annals of VRP.

VRP is a set of minimum cost vehicle routes that, under certain constraints, will fulfill the demands of specific clients geographically dispersed from one or more distribution locations (depots) [15]. Each vehicle begins to move at the facility and returns to the same location after visiting all customers. Each vehicle in the fleet has a certain load capacity, and the capabilities of all vehicles are equivalent and have comparable expenses (insurance, maintenance, etc.). Vehicle capacities and client requests are predetermined (deterministic) [16]. Main constraints of classical VRP are shown in below:

1. Each customer's request is fulfilled by a vehicle.
2. The demands of all customers are fulfilled.
3. Each vehicle can support multiple customers as long as the capacity of the vehicle is not surpassed.
4. Each vehicle starts its tour from the depot and returns to the depot after meeting the customer's demands.
5. Each vehicle can be departed from the depot at once and return to the depot at once.

2.1. Variants of Vehicle Routing Problem

The Capacity Vehicle Routing Problem refers to the most basic version of VRP that fits the aforementioned constraints (CVRP). CVRP assumes that the vehicles have homogeneous capacity (equivalent capacity).

By adding different constraints, VRP variants come up, the extensions of CVRP are shown in **Figure 2.1.** [14]. Each arrow represents a CVRP extension; for example, by adding a route length constraint to CVRP, the problem is transformed into a Distance Constrained Vehicle Routing Problem (DCVRP).

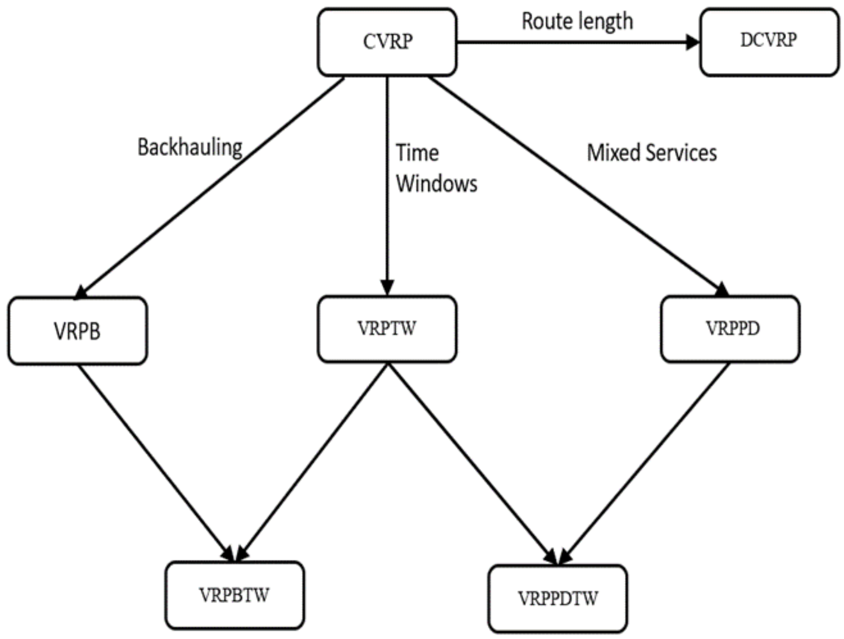


Figure 2.1. *Extensions of CVRP*

2.1.1. Distance constrained vehicle routing problem (DCVRP)

In Distance constrained VRP, the total distance that each vehicle assigned to the routes will travel is limited. The type of product to be transported, the type of vehicle to be used, type of driver elements can create distance capacitated problems. The limitation that the driver cannot be on the way continuously for more than a certain period of time, the condition that the transported product may deteriorate as a result of prolonged transportation, etc. if there are such situations, the distance constraint should be included in the model and the problem is called Distance Constrained VRP. Most known articles are Laporte, Nobert, and Desrochers [17, 18].

2.1.2. Vehicle routing problem with time windows (VRPTW)

VRP with Time Windows (VRPTW) is a CVRP version that introduces a new constraint that forces vehicles to be scheduled inside a specific time interval. Pullen and Webb [19], Knight and Hofer [20], and Madsen [21] conducted early VRPTW studies. Solomon [22] leads the well-known VRPTW research with different datasets.

2.1.3. Vehicle routing problem with backhauls (VRPB)

VRP with Backhauls (VRPB) is a variant of CVRP in which the customer set is divided into two subsets, one for delivery and second for pick-up. Martello and Toth [23] and Deif and Bodin [24] are well-known VRPB researchers that offered an adaptation of Clarke and Wright [12] VRP heuristic.

2.1.4. Vehicle routing problem with pickup and delivery (VRPPD)

VRP with Pickup and Delivery (VRPPD) evolved from CVRP, which was represented by two quantities: one for demands to be delivered to the customer and the other for commodities to be picked up from the customer. Savelsbergh and Sol [25], Desrosiers et al. [26], and Solomon and Desrosiers [27] all contributed to the advancement of VRPPD research.

2.1.5. VRPB with time windows (VRPBTW)

The time windows extension of VRPB is called VRPB with Time Windows (VRPBTW). VRPBTW model is a model developed from the basic model VRPB by adding constraint of time window on each customer. Cho and Wang [28] developed a model VRPBTW that minimizes the number of vehicles and routing time by starting from the depot and end at the depot.

2.1.6. VRPPD with time windows (VRPPDTW)

VRPPD with windows is composed by adding time windows constrain to a classical VRPPD. The VRPPDTW has a variety of practical applications, including the transport of the disabled and elderly people, sealift and airlift of cargo and troops, and pickup and delivery for overnight carriers or urban services. Perspectives on this growing field were offered by Savelsbergh and Sol [25], Desrosiers et al. [26], and Solomon and Desrosiers [27].

2.2. Solution Methods of VRP

The CVRP's objective functions differ as much as its constraints and variations. The objective function is not always to find the shortest route. The following are the primary objective functions of VRP that are commonly mentioned in the literature:

1. To minimize vehicle travel time.
2. Reduce total transportation expenses associated with the fixed costs of the vehicles used, in addition to reducing total distances travelled by vehicles.
3. To minimize the total number of vehicles that leave the depot.
4. If splitted distribution is to be made to customers, to minimize the penalties arising from fragmented distribution.
5. Serving as many customers as possible.

As the constraints and objective functions change in VRP, the solution space also changes its shape and it grows, so it becomes difficult to find the optimal point located in the solution space, therefore VRP is *NP-hard problem* [29]. In this case, various solution methods have been devised to find the optimal solution. As a result, heuristic solution methods for real-world problems that cannot be solved using exact solutions have been developed. VRP solution methods are shown in **Figure 2.2**.

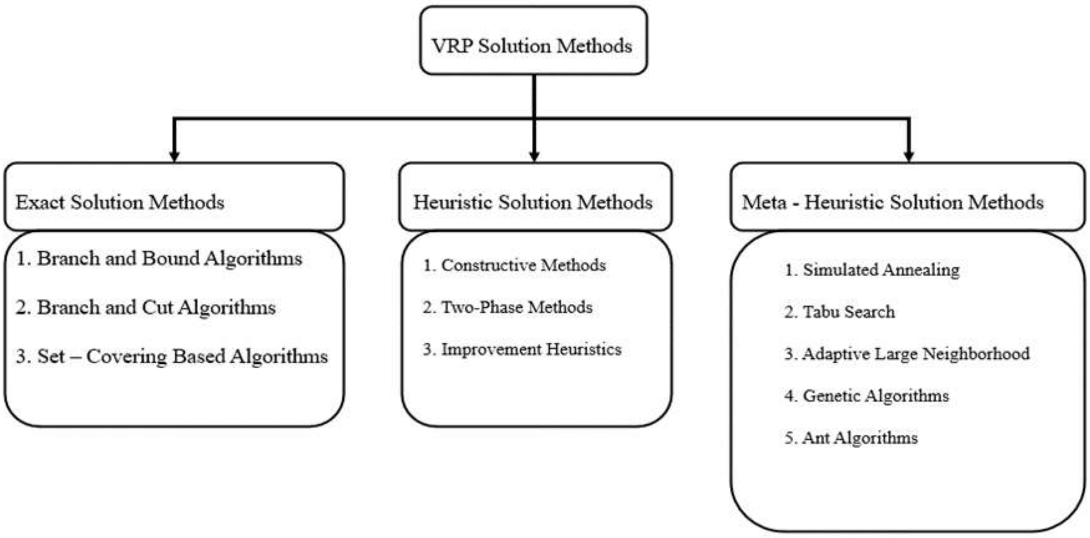


Figure 2.2. *VRP solution methods*

As the methods of solving VRPs are examined, it is known that exact solution methods, heuristic solution method and meta-heuristic solution methods are often used in the literature.

2.2.1. Exact Solution Methods

CVRP which is the basic version of VRP, it is seen that exact solution methods based on branch-bound, branch-cut and set-covering based algorithm problems are included in the study of Toth and Vigo [14]. Accordingly, the branch-bound algorithm for CVRP [30] and set-covering based [31] algorithms who give exact solutions derived on the basis of, can be counted as pioneers.

2.2.2. Heuristic Solution Methods

The classical heuristics developed for VRP can be classified in three groups as constructive methods, two-phase methods, and improvement heuristics. For constructive methods, Clark and Wright saving algorithm [12], which is still adapted in the literature for all types of VRP and TSP can be given as an example and is used quite often. Two-Phase methods are divided into two, one is “Cluster First; Route Second” (CFRS), and the other is “Route First; Cluster Second" (RFCS). On the other hand, improvement heuristics, can be divided into two, one is improving a single route, and the other is improving multiple routes. Algorithms such as in-route and inter-route insertion, binary Decoupling, 2-opt can be sorted in these two types.

2.2.3. Meta-Heuristic Solution Methods

Meta-heuristic solution methods have developed in a very large number over the years. Among these algorithms, especially in the early 1990s, Simulated Annealing [32]; tabu search algorithm [33, 34]; adaptive large neighborhood search [35], genetic algorithm [36] and ant algorithms [37] can be given as examples. As new technologies are emerging, new solution methods are developed for VRP. Last decade, one of the most important methods that has been increased the effectiveness of VRP is Machine Learning techniques.

2.2.4. Machine learning solutions methods for VRP

Machine learning is a scientific field of study that contributes for the development of various techniques and algorithms to enable computers to learn in a similar way like humans [38]. Machine learning is divided into three types: supervised, unsupervised and reinforcement learning. In supervised learning, the computer is trained with a previously correctly classified data source. From this training, the meaningful results that has obtained from models is used from computer, to perform with the models that has not encountered before. In most cases, supervised learning is utilized to solve classification and regression issues. Unsupervised learning is a type of machine learning in which the computer tries to generate hidden patterns in a given model by drawing a meaningful conclusion. It is commonly used for clustering and pattern finding tasks. Reinforcement learning exposes the machine to an environment where it must constantly teach itself via trial and error. This computer learns from its previous experiences and attempts to acquire the most relevant information in order to make appropriate decisions. Reinforcement learning is used to solve problems like self-driving vehicles and energy consumption.

Machine Learning is consists of sub-elements of mathematics, such as statistics, optimization, and probability. In the literature, the use of machine learning in the solution of the VRP have been first proposed by Hopfield and Tank [39], however, the usage of machine learning in VRP solutions has accelerated in recent years. Instead of performing complex mathematical operations on a computer, it has been a preferable situation for machine learning to investigate the solution space with the methods it has learned by making a quick generalization. To teach cutting-plane method [40], the method of solving by learning important nodes [41], learning using heuristic methods [42] are the main known supervised learning studies in the literature. As an example of studies related to unsupervised learning are learning the branch strategy [43], learning variable selection policy [44], constraint satisfaction problem [45] studies which can be given.

In the literature, hybrid solution approaches with machine learning based have also been used. These solution methods are cluster first, routing second (CFRS) and routing first, cluster second (RFCS) algorithms. In RFCS algorithms, first, a TSP solution is taken that covers all customers. The solution of TSP is decomposed into vehicle routes in such a way that they do not breach their capacity limits. In CFRS, the exact opposite of the operation performed in RFCS is done, customers are divided into vehicles first, and TSP mathematical models are solved so that the vehicles can go to the customers in the shortest way. Examples of assisted machine learning in the literature include CFRS; Beasley [46], Montoya [47], and RFCS; Fisher and Jaikumar [48], Dondo and Cerdá [4].

3. METHODOLOGY

In this section, the algorithms used in the CVRP solution are explained according to the chronology used in the solution method. **Figure 3.1.** depicts the stages involved in the CVRP solution. Machine learning methods are used to assign vehicles to nodes first. The capacities of the vehicles are handled with CBA, in the second stage, and balancing is performed if there is an excess capacity. Finally, TSP is used to solve the nodes assigned to each vehicle. The CVRP solution is obtained by combining the TSP solutions of each car in the fourth stage. The sections are sorted according to this chronology, and their definitions and roles in the model are visually shown in each section.

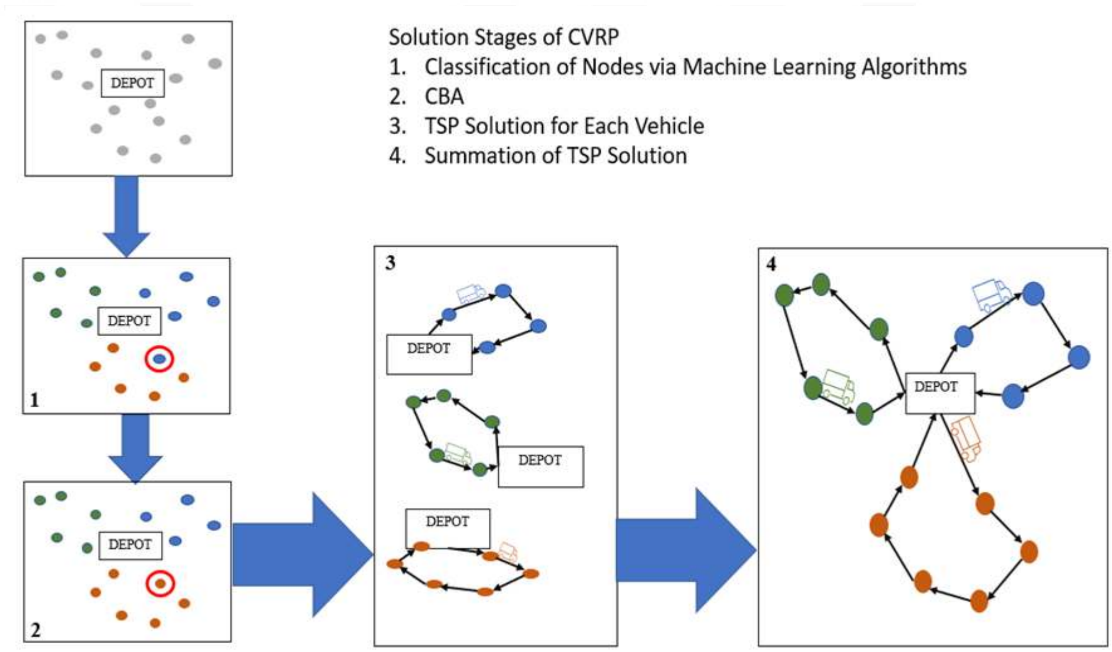


Figure 3.1. *Solution stages of CVRP*

3.1. CVRP Mathematical Formulation

Capacitated (homogeneous) vehicle routing problem (CVRP) is to minimize the total distance travelled by vehicles so that homogeneous vehicles with the same capacity do not violate the vehicle capacities of all nodes, starting from the depot. It is known that vehicle routing problems can be modeled based on two-index and three-index vehicle-flow formulations. In this section, a mixed integer mathematical programming model based on a two-index vehicle-flow formulation is used [49]. The parameters and decision variables used in the mathematical model can be described as follows:

Index:

i,j : Nodes(Customers)

Sets:

V = (0,..,N) sets of nodes, 0: depot

Parameters:

q_i : Demand of Node i,

C : Vehicle's Capacity

K : Number of Vehicles

d_{ij}: distance from node i to node j

Decision Variables:

x_{ij}: binary variable if a vehicle travels from i to j, x_{ij} equals 1 otherwise 0

u_i: continuous variable representing the load of the vehicle after visiting customer *i*.

Objective Function:

$$Min\ Z = \sum_{i \in V} \sum_{j \in V} d_{ij} x_{ij} \quad (3.1)$$

Constraints:

$$\sum_{i \in V} x_{ij} = 1 \qquad \forall j \in V \backslash \{0\} \quad (3.2)$$

$$\sum_{j \in V} x_{ij} = 1 \qquad \forall i \in V \backslash \{0\} \quad (3.3)$$

$$\sum_{i \in V} x_{i0} = K \quad (3.4)$$

$$\sum_{j \in V} x_{0j} = K \quad (3.5)$$

$$u_i - u_j + C x_{ij} \leq C - q_j \quad \forall\, i,j \in V \backslash \{0\}, i \neq j \quad (3.6)$$

$$q_i \leq u_i \leq C \quad \forall i \in V \backslash \{0\} \quad (3.7)$$

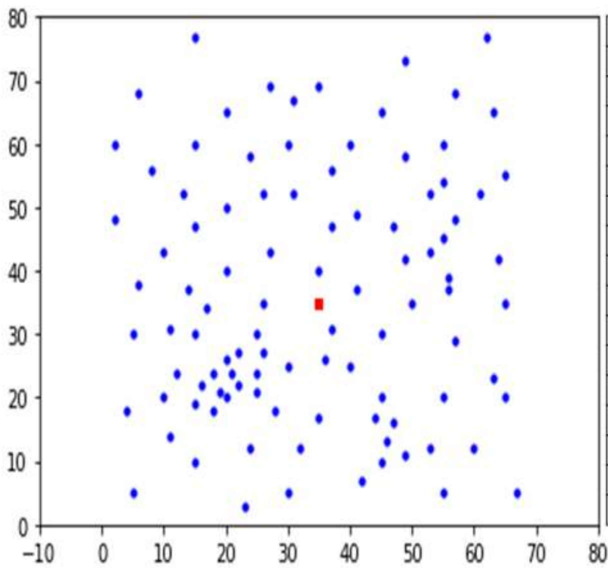
$$x_{ij} \in \{0,1\} \qquad \forall i,j \in V \quad (3.8)$$

In the model, the objective function (3.1) minimizes the distance to be traveled in the distribution to all nodes. Constraint (3.2) and (3.3) ensure that there is only one incoming to each node and only one outgoing from each node. Constraint (3.4) and (3.5) ensure K number of vehicles to leave the depot and the same number of vehicles to return to the depot. Constraint (3.6) is sub-tour elimination constraint. Sub-tour elimination constraint prevents the sub-tours of routes while guarantees any capacity of vehicles have not been exceeded. Constraint (3.7) ensures that the load of a vehicle does not be less than the node's demand and that the vehicle's capacity is not violated. Constraint (3.8) is called a sign constraint, x_{ij} is a binary variable.

CVRP mathematical formulation is used for two purposes in this thesis:

1. To obtain labeled data for supervised machine learning algorithms to train machine learning models for their future predictions.
2. To make comparison with the purposed model solutions.

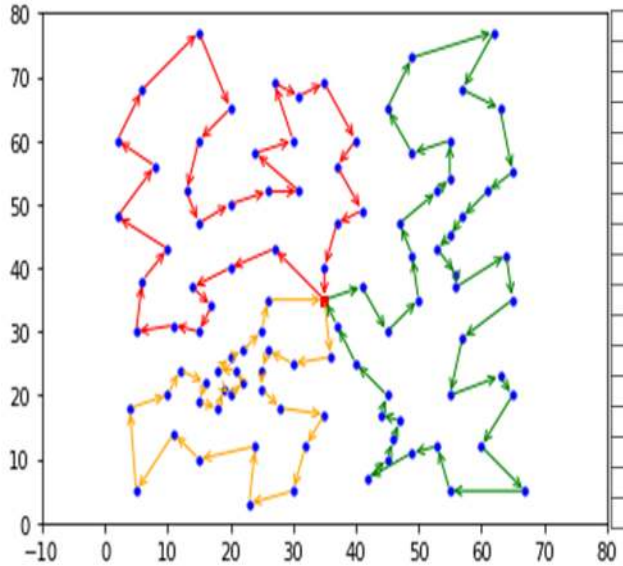
To acquire labeled data, one of the CVRP datasets in the literature is downloaded on Python utilising Numpy and Pandas Library .In this thesis R101 [22] dataset is used for to train supervised machine learning algorithms. **Figure 3.2.** shows the R101 dataset schema and first 15 rows of the dataset. The nodes are represented by blue dots, whereas the depot is represented by a red square.



Index	Node No	X-Coord.	Y-Coord.	Demand
0	0	35	35	0
1	1	41	49	10
2	2	35	17	7
3	3	55	45	13
4	4	55	20	19
5	5	15	30	26
6	6	25	30	3
7	7	20	50	5
8	8	10	43	9
9	9	55	60	16
10	10	30	60	16
11	11	20	65	12
12	12	50	35	19
13	13	30	25	23
14	14	15	10	20
15	15	30	5	8

Figure 3.2. *Schema of R101 and first 15 rows of R101 dataset*

The Python-MIP open-source platform with GUROBI solver 9.12 is used to solve this loaded dataset using the CVRP mathematical formulation. After performing solution on CVRP, a new column is inserted to the R101 dataset to identify which vehicles are assigned to which nodes. As explained previously, CVRP is an NP-Hard problem, hence an optimum solution for R101 could not be obtained, so the CVRP model was given 96 hours of CPU time limit to solve. This labeled R101 dataset, in which the CVRP mathematical solution is included, is utilized to train all supervised machine learning algorithms in all datasets. Due to the fact that the nodes of R101 are randomly dispersed and not clustered, the R101 dataset was used for training purposes. **Figure 3.3.** depicts the solution routes in R101 with 3 vehicles, as well as the first 15 rows in Labelled R101 with 3 vehicles. As can be seen **Figure 3.3.**, a new column named "Vehicle No" is created, which represents the node assignment.



Index	Node No	X-Coord.	Y-Coord.	Demand	Vehicle No.
0	0	35	35	0	0
1	1	41	49	10	2
2	2	35	17	7	1
3	3	55	45	13	3
4	4	55	20	19	3
5	5	15	30	26	2
6	6	25	30	3	1
7	7	20	50	5	2
8	8	10	43	9	2
9	9	55	60	16	3
10	10	30	60	16	2
11	11	20	65	12	2
12	12	50	35	19	3
13	13	30	25	23	1
14	14	15	10	20	1
15	15	30	5	8	1

Figure 3.3. *Solution of R101 with 3 vehicles and first 15 rows of labeled R101 dataset*

A comparison with the proposed model to CVRP mathematical formulation is another motivation. The results of CVRP with 2, 3, and 4 vehicles on a certain CPU time with datasets are included in **Table 3.1.** In computational results chapter, this proposed model comparison to CVRP mathematical formulation solution according to dataset will be analyzed.

Table 3.1. *CVRP results for datasets*

CVRP Results for Datasets						
	CVRP with 2 Vehicles		CVRP with 3 Vehicles		CVRP with 4 Vehicles	
	Obj. F.	Time(sec)	Obj. F.	Time(sec)	Obj. F.	Time(sec)
RC101	763.00	21600.00	720.23	21600.00	814.38	21600.00
C101	531.98	21600.00	564.08	21600.00	610.80	21600.00
Tai100a	1148.01	21600.00	1209.32	21600.00	1309.51	21600.00
Golden-1	4313.82	21600.00	4521.28	21600.00	4800.18	21600.00
CMT3	651.30	21600.00	696.94	21600.00	719.71	21600.00

3.2. Machine Learning Algorithms

Machine learning algorithms basically produce two types of output. If the output is numerical, it is called regression, and if it is categorical, it is called a classification problem. Machine Learning algorithms are generally grouped into three categories: supervised, unsupervised and reinforcement learning. To anticipate numerous categorization problems, LR and K-NN algorithms, which are supervised machine learning algorithms, were utilized in this work. The K-Means approach, which is an unsupervised machine learning algorithm, is also used to cluster the data.

Classification problems are used in many applications. Data mining [50, 51], protein interaction [52], cancer diagnosis [53], bankruptcy prediction [54] are few known examples. K-NN, LR, and K-Means machine learning algorithms, which are the most commonly utilized in optimization problems, were used in this thesis.

In this study, supervised machine learning algorithms perform classification in the solution of CVRP by making a prediction about which vehicles will be assigned to which customers. It is aimed to make a classification with supervised machine learning algorithms. K-NN and LR algorithms have classified the customers by vehicle and performed the assignments. Since K-NN and LR algorithms are supervised learning algorithms, they must be trained before they can be applied. **Figure 3.4.** describes the supervised learning workflow.

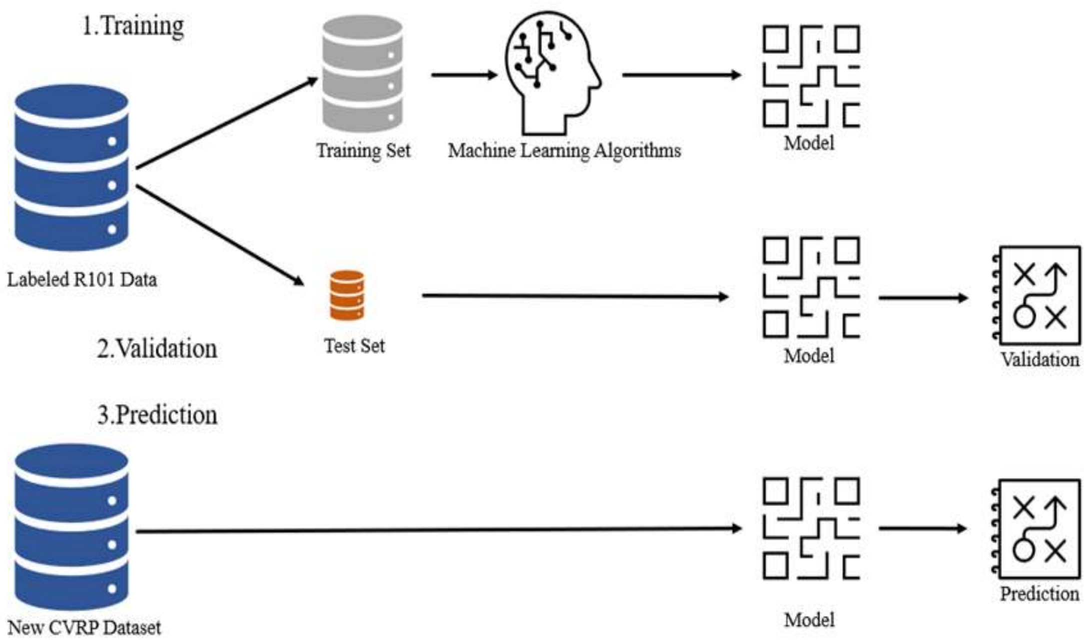


Figure 3.4. *The workflow of supervised learning*

Observe from **Figure 3.4.** that there are 3 stages in Supervised Learning algorithms. In training stage according to holdout method [55], labelled R101 Dataset is divided into a training set and test set. In this study, 70% training - 30% test data partitioning is used. By using Labelled Training Set, K-NN and LR algorithms discover the patterns of classification. In this machine learning modelling, only X-Coordinates and Y-Coordinates of R101 are used for training as features. It means that in this model there are only two features, it is expected from machine learning algorithms to predict the assignments of vehicles by using these two features. A machine learning model is created using algorithm parameters. Following the setup of the model, the accuracy test is carried out for validation. In this test, the model attempted to predict which assignments previously known and divided previously on a percent 30% test set. The prediction of model and the labelled dataset are compared, and the model will be utilized for new CVRP datasets based on the accuracy test results. The ratio of correctly predicted assigned vehicles to total nodes is known as accuracy. **Table 3.2.** represents the comparison of labeled R101 and LR predicted outcomes for CVRP with three vehicles over 15 nodes.

Table 3.2. *The comparison of labeled R101 and LR predicted outcomes for CVRP*

Index	Node No	X-Coord.	Y-Coord.	Demand	Vehicle No.	Prediction of LR Model
0	0	35	35	0	0	0
1	1	41	49	10	2	1
2	2	35	17	7	1	1
3	3	55	45	13	3	3
4	4	55	20	19	3	3
5	5	15	30	26	2	2
6	6	25	30	3	1	1
7	7	20	50	5	2	2
8	8	10	43	9	2	2
9	9	55	60	16	3	3
10	10	30	60	16	2	2
11	11	20	65	12	2	2
12	12	50	35	19	3	3
13	13	30	25	23	1	1
14	14	15	10	20	1	1
15	15	30	5	8	1	1

According to **Table 3.2**, Node 1 in red is mispredicted by LR model. Accuracy test results of labeled R101 dataset is shown in **Table 3.3.**

Table 3.3. *Accuracy test results of labeled R101*

	Accuracy Test Results of Labeled R101					
	CVRP with 2 vehicles		CVRP with 3 vehicles		CVRP with 4 vehicles	
	Trainset Accuracy	Test set Accuracy	Trainset Accuracy	Test set Accuracy	Trainset Accuracy	Test set Accuracy
K-NN Model	1	1	1	1	1	0.93
LR Model	0.98	0.96	0.98	0.96	0.98	0.96

Observe from **Table 3.3.,** accuracy test results are sufficient to make predictions on new datasets, more than 90% of accuracy test results is a good performance. Using only 2 features, resulted with higher accuracy test results. LR results are lower from K-NN because of its linearity on decision boundary. On the LR Model, there is a decrease in

Trainset Accuracy and Testset Accuracy; this decline indicates model overfitting, but this overfitting is insignificant to affect the problem model prediction for further datasets.

3.2.1. Logistic regression algorithm

The logistic regression as a general statistical model was originally developed and popularized primarily by Joseph [56]. LR is the form of the linear classification equation that has turned into a logarithmic maximization. LR makes an effective classification by selecting the features in the dataset. In a dataset with 1000 samples, LR model can make the classification in a matter of seconds. **Figure 3.5.** shows the transformation of linear regression to logistic regression.

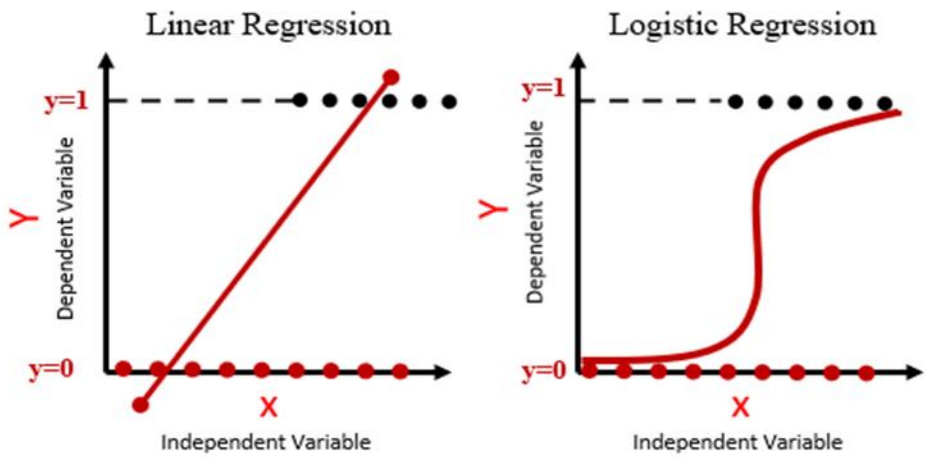


Figure 3.5. *Transformation of linear regression to logistic regression*

X is an independent variable, which indicates that it is the machine learning algorithm's input. Y is a dependent variable, which its value changes as a result of X. Y is a predicted value as well. As illustrated, the predicted value of y in linear regression is between 0 and 1. LR, on the other hand, predicts a value of y of 0 or 1.

Given that classes are discrete in supervised classification problems, the purpose of algorithms is to find *decision boundaries* between classes. Depending on the example of the problem, the decision boundaries may be complex and not geometrically linear. In general, different machine learning algorithms have different assumptions about the shape of decision boundaries. In the case of logistic regression, the assumption is that the decision boundaries are linear. That is, these are hyperplanes in the high-dimensional feature space, where the size of the feature space is simply determined by the number of elements in the feature vector of a training sample. The purpose of logistic regression is to find a way to divide data points to have an accurate estimate of the class of a particular observation using the features contained from labelled datasets. In this thesis, LR model utilizes X and Y coordinates of dataset as features to calculate node categorization and decision boundary.

LR has 12 parameters in Python Scikit-learn library, the most effective parameters are “penalty” and “C”. The regularization of the model is changed to Ridge regularization [57] or Lasso regularization [58] by utilizing the "penalty" parameter. “C” parameter calibrates the effect of regularization, by using small values it specifies stronger regularization. With using GridSearchCV function, which is used for models to find best solutions by adjusting parameters, the parameters of LR model are fixed on Ridge Regularization with C value equals to 0.05. The decision boundaries of the labeled R101 dataset with 2, 3, and 4 vehicles using the parameters above are shown in **Figures 3.6., 3.7., 3.8**. respectively. Observe from **Figure 3.6.** that decision boundaries are linear.

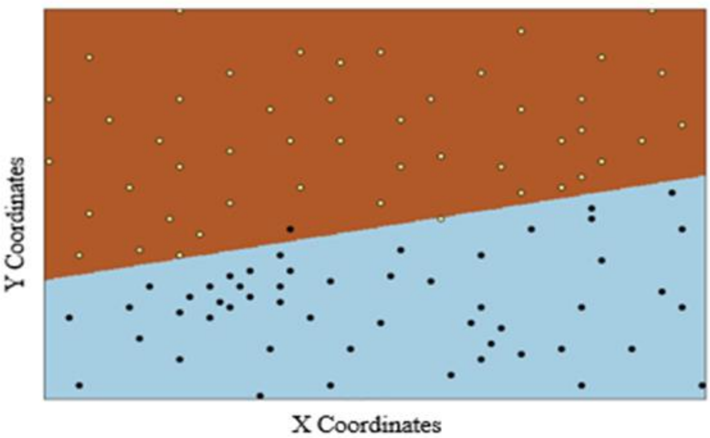


Figure 3.6. *Decision boundary of LR model for CVRP with 2 vehicles*

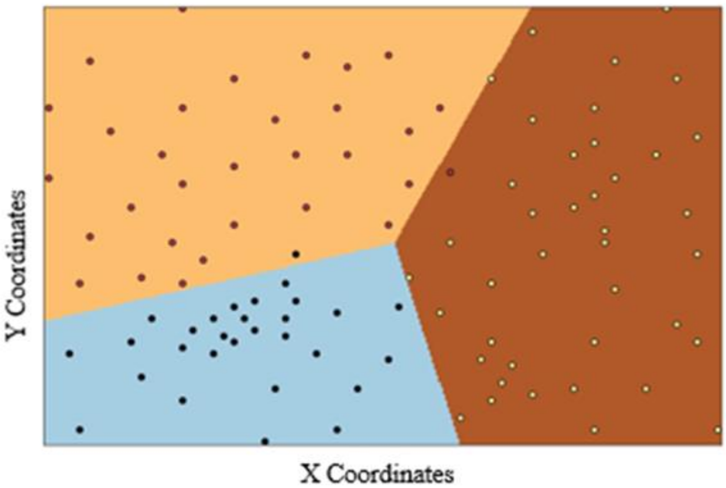


Figure 3.7. *Decision boundary of LR model for CVRP with 3 vehicles*

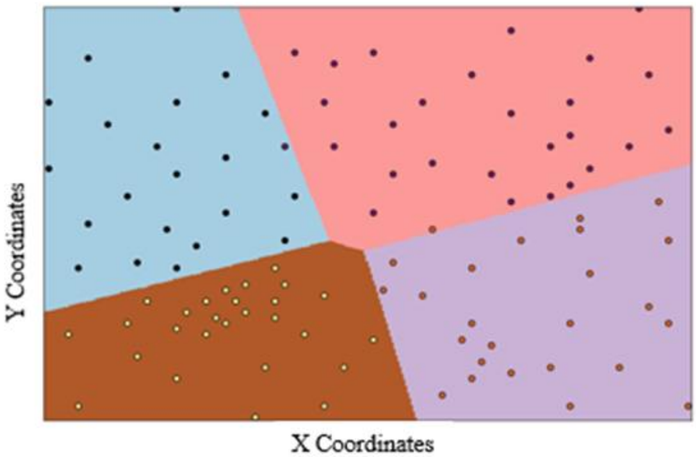


Figure 3.8. *Decision boundary of LR model for CVRP with 4 vehicles*

Since the decision boundaries of the LR model has been clarified, it is now be deployed to new datasets. This decision boundary regulates the assignment of vehicles to nodes. After the model has been validated, it can give predictions for new data. **Figure 3.9.** shows the schema for new data and its prediction using the LR model. CVRP prediction is made in this model with 2 vehicles.

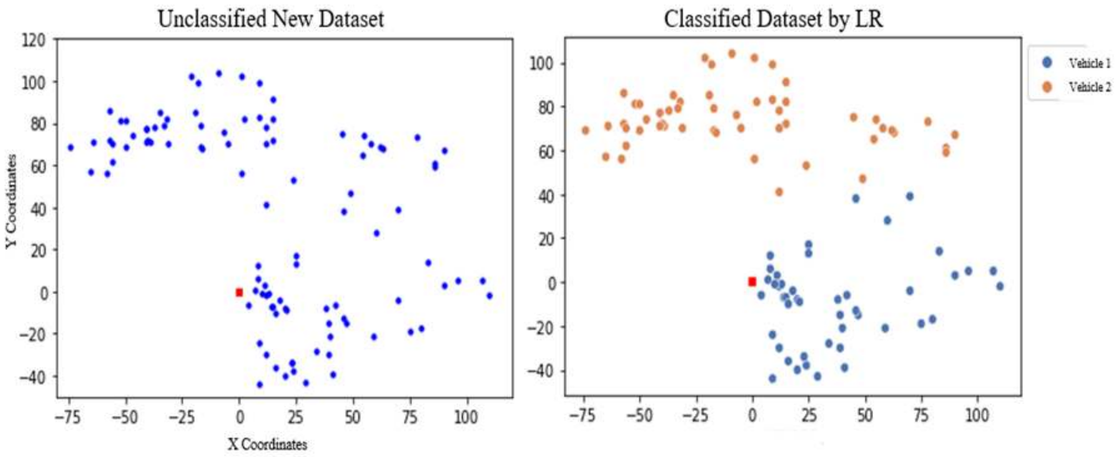


Figure 3.9. *Figure 3.9. Unclassified new dataset vs. classified new dataset by LR*

When **Figure 3.9.** is checked, the decision boundary of LR with 2 vehicles is applied to the new dataset. The "Vehicle Prediction of LR Model" column was placed into the new dataset after the vehicles were assigned using LR prediction, and the predictions were deployed to this column using the Python Pandas package. **Table 3.4.** shows the 15 new dataset selections that have been deployed.

Table 3.4. *Vehicle prediction of LR model on new dataset*

Index	Node No	X-Coord.	Y-Coord.	Demand	Vehicle Prediction of LR Model
1	1	41	49	10	2
2	2	35	17	7	4
3	3	55	45	13	2
4	4	55	20	19	3
5	5	15	30	26	1
6	6	25	30	3	4
7	7	20	50	5	1
8	8	10	43	9	1
9	9	55	60	16	2
10	10	30	60	16	2
11	11	20	65	12	1
12	12	50	35	19	3
13	13	30	25	23	4
14	14	15	10	20	4
15	15	30	5	8	4

3.2.2. K-Nearest Neighbor Algorithm

The K-NN is a nonparametric supervised learning method, it was first used by Evelyn Fix and Joseph Hodges in 1951 [59]. It is used for classification and regression predictions. In both cases, the input consists of the nearest K training sample in a dataset. In the K-NN classification, output is a class membership. An object is classified by the plural vote of its neighbors, and the object is assigned to the class that is most common among its nearest K neighbors (K is a positive integer, typically Decimals). A demonstration of K-NN working principle is shown in **Figure 3.10**.

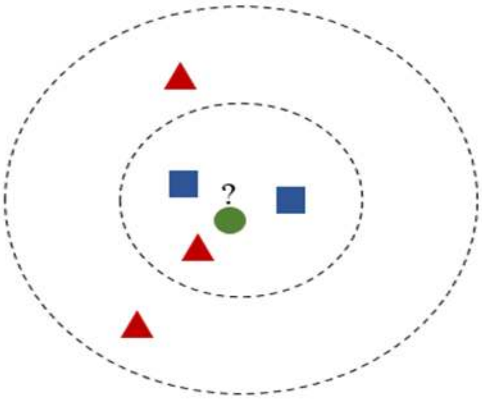


Figure 3.10. *K-NN working principle*

Figure 3.10. can be used as an example of the working principles of K-NN machine learning algorithm. K-NN method, for example, predicts the form of a circle. There are two labels, one for the triangle and the other for the square. If K=3, the method computes the circle's closest three neighbors; counting these neighbors, there are two squares and one triangle, hence the circle's prediction is square. If K=5 and there are 3 triangles and 2 squares, the circle prediction is triangle.

In the Python Scikit-Learn Library, K-NN includes seven parameters, the most popular and effective of which are "n_neighbors" and "weight." The number of nearby neighborhoods is determined by "n_neighbors." The "weight" parameter is used to determine the weight of nodes; for example, if "weight" is uniform, all nodes' weights are equal; on the other hand, if "weight" equals "distance," the closest nodes have a stronger impact on vehicle assignment to unclassified nodes. The "n neighbor" parameter is equal to 5 and the "weight" parameter is equal to "distance" according to GridSearchCV function, which finds the optimum parameters for K-NN. Figures **3.11., 3.12., 3.13.** illustrate the decision boundaries on K-NN Model of Labeled R101 with 2,3 and 4 vehicles respectively, using these parameters.

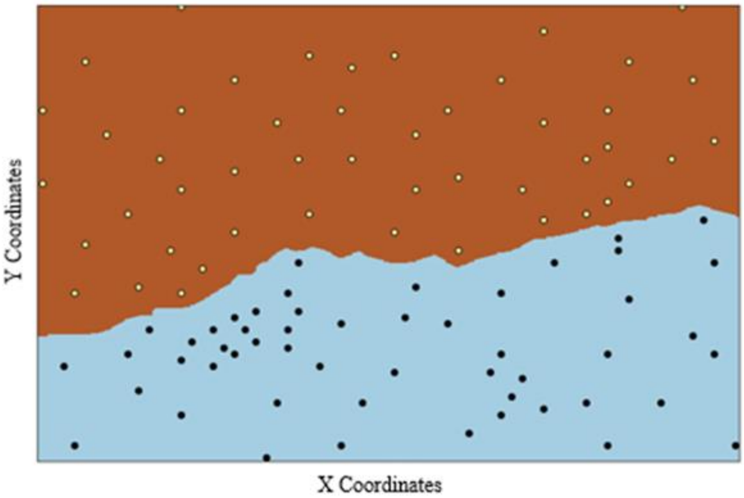


Figure 3.11. *Decision boundary of K-NN model for CVRP with 2 vehicles*

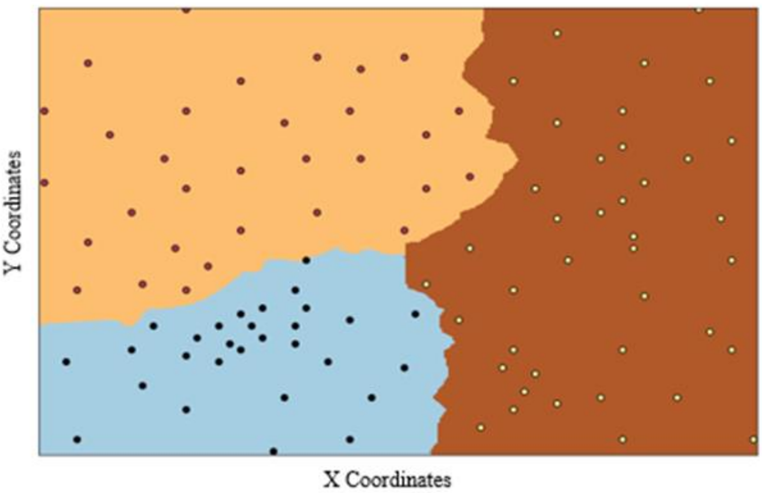


Figure 3.12. *Decision boundary of K-NN model for CVRP with 3 vehicles*

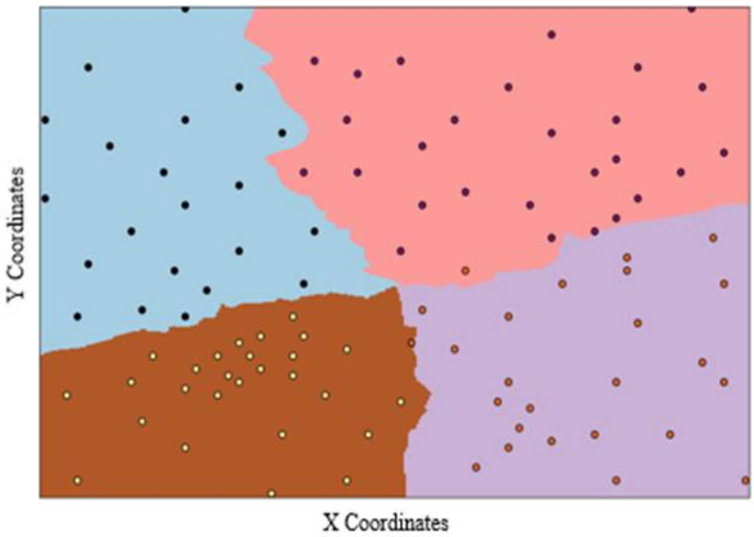


Figure 3.13. *Decision boundary of K-NN model for CVRP with 4 vehicles*

The Decision Boundary divides data points into regions, which corresponds to the class they belong to. The labelled R101 dataset is used to determine region separation. The classification of new datasets could be achieved with the help of these regions. In **Figure 3.11.**, for instance, if one of the new dataset's nodes is in the blue region, the first vehicle is allocated; if the node is in the brown region, the second vehicle is assigned. K-NN algorithm is an algorithm based on the local geometry of the distribution of data in the attribute hyperplane (and their relative distance measurements). Therefore, the decision boundary turns out to be nonlinear and non-uniform.

As decision boundaries of model is developed, it can be used for the new dataset. The assignments of vehicles to nodes are decided according to these decision boundaries. After validation of the model, prediction of new data can be produced by model. Schema of a new data and prediction of this new data by applying K-NN model is shown in **Figure 3.14**. In this model, CVRP with 4 vehicles prediction is made.

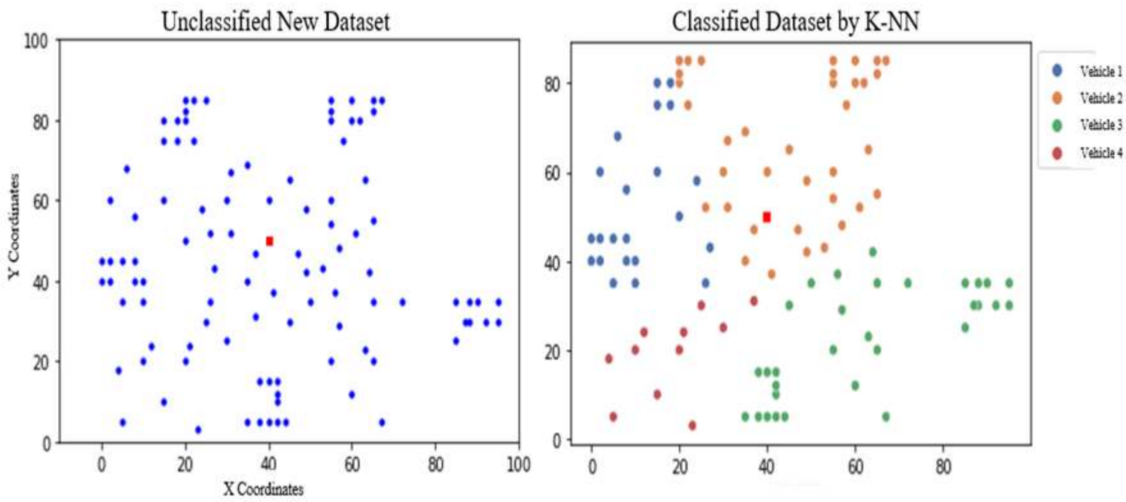


Figure 3.14. *Unclassified new dataset vs. classified new dataset by K-NN*

Decision boundary of K-NN with four vehicles is applied to new dataset if **Figure 3.13.** is checked. After assignment of vehicles are made by K-NN prediction, "Vehicle Prediction of K-NN Model" column inserted to the new dataset and the predictions deployed to this column by the help of Python Pandas library. The deployed 15 assignments of the new datasets are displayed in **Table 3.5**.

Table 3.5. *Vehicle Prediction of K-NN Model on new dataset*

Index	Node No	X-Coord.	Y-Coord.	Demand	Vehicle Prediction of K-NN Model
1	1	25	85	20	2
2	2	22	75	30	2
3	3	22	85	10	2
4	4	20	80	40	2
5	5	20	85	20	2
6	6	18	75	20	1
7	7	15	75	20	1
8	8	15	80	10	1
9	9	10	35	20	1
10	10	10	40	30	1
11	11	8	40	40	1
12	12	8	45	20	1
13	13	5	35	10	1
14	14	5	45	10	1
15	15	2	40	20	1

3.2.3. K-Means Algorithm

K-Means clustering algorithm is classified as unsupervised machine learning algorithms. The method is aimed at dividing a dataset consisting of n data objects into K clusters in which each observation belongs to the cluster with the nearest average (cluster centers or cluster center). K-Means algorithm, the similarity between clusters while minimizing inertia similarities are working to maximizing relies on most. The basic principle of operation of the algorithm is based on minimizing quadratic errors. The algorithm works in two steps; first, it assigns each node in the data to the nearest clustering region, then determines each clustering center based on the average of the data. Since K-Means is an unsupervised learning algorithm, there is no need to separate any training and test sets. **Figure 3.15.** shows an example of clustering. Red triangles demonstrate the cluster centers.

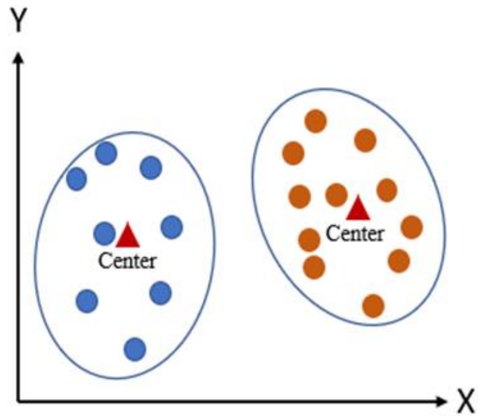


Figure 3.15. *K-Means clustering*

There are 8 parameters of K-Means Algorithm in Scikit-learn library. The most effective ones are “n_cluster” and “random_state”. “n_cluster” determines the number of clusters. Considering the CVRP with 2 vehicles, there needed to be 2 clusters therefore “n_cluster” will be equal to 2. “random_state” parameter use to make the randomness deterministic. “random_state” is given as 42, as in many python machine learning examples therefore this model also uses the same value [60]. The decision boundaries are generated utilizing R101 dataset, and the features are nodes coordinates, as illustrated in **Figure 3.16., 3.17., 3.18.**

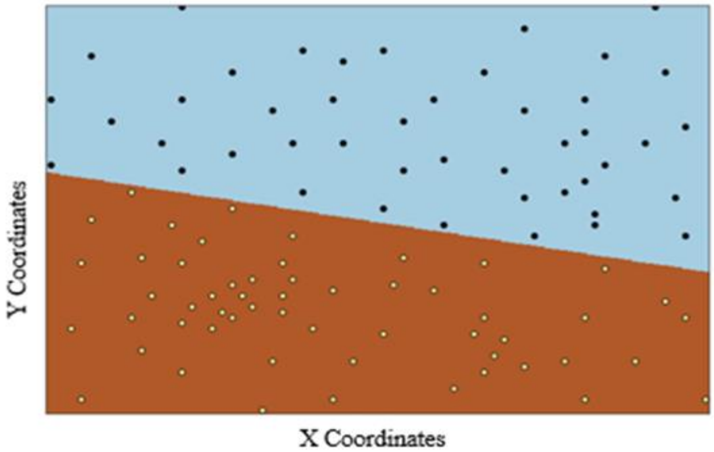


Figure 3.16. *Decision boundary of K-Means model for CVRP with 2 vehicles*

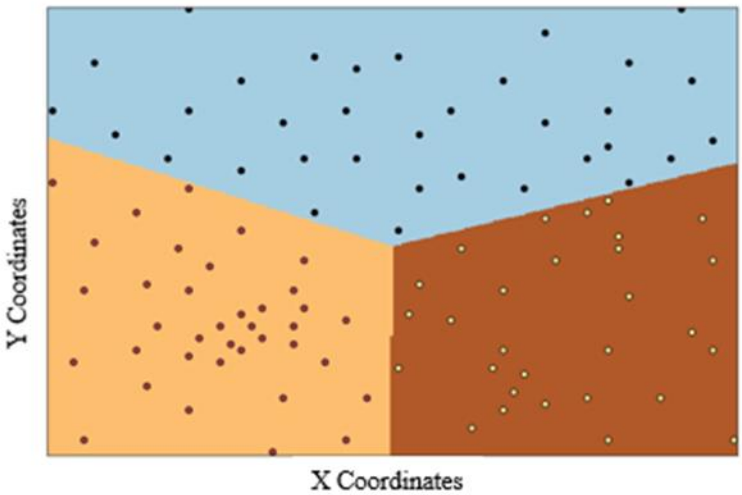


Figure 3.17. *Decision boundary of K-Means model for CVRP with 3 vehicles*

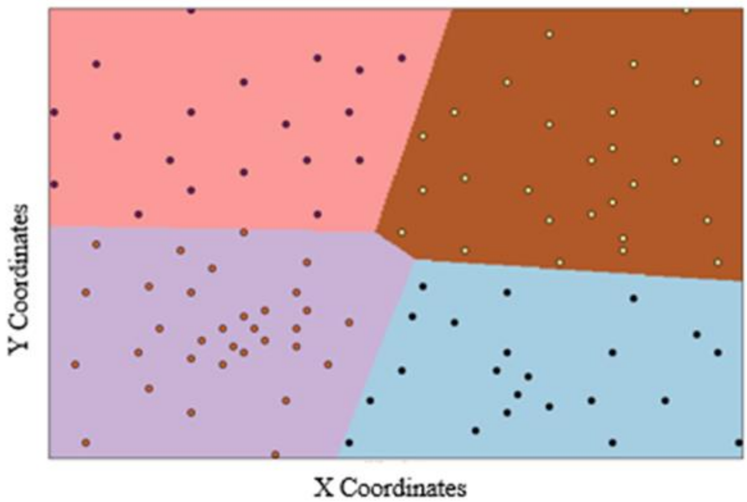


Figure 3.18. *Decision boundary of K-Means model for CVRP with 4 vehicles*

After construction of K-Means model, prediction of new data can be produced by the model. Schema of a new data and prediction of this new data by applying K-Means model is shown in **Figure 3.19**. In this model, CVRP with 3 vehicles prediction is made.

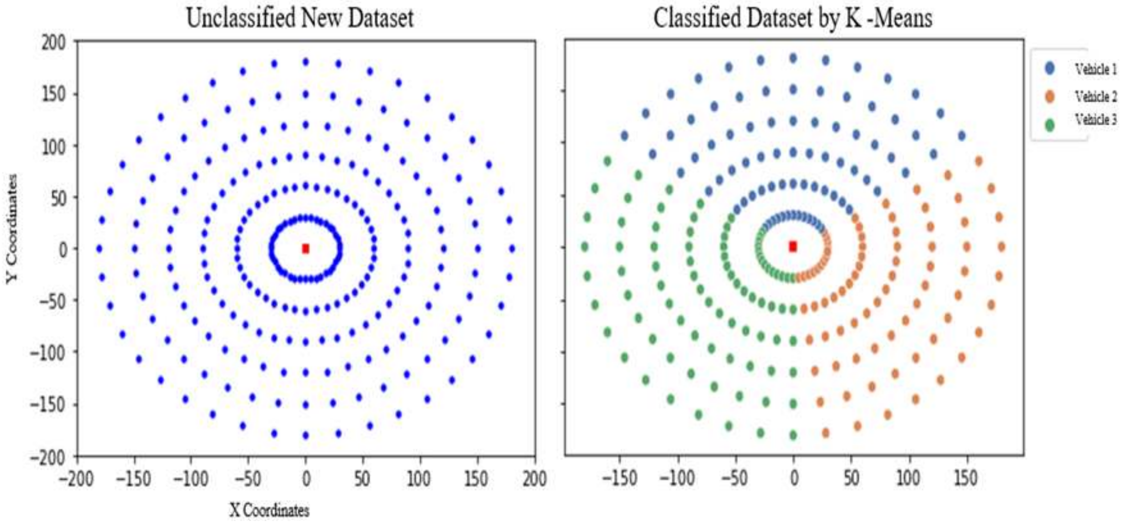


Figure 3.19. *Unclassified new dataset vs. classified new dataset by K-Means*

As seen in **Figure 3.17.** the template of K-Means Decision Boundary for 3 vehicles exactly fit to the new dataset. After K-Means algorithm assisted prediction are made, the assignments are inserted to the new dataset. In **Table 3.6.**, rearranged new dataset with assignment is shown which vehicle prediction of K-Means Model Column is added.

Table 3.6. *Vehicle prediction of K-Means model on new dataset*

Index	Node No	X-Coord.	Y-Coord.	Demand	Vehicle Prediction of K-Means Model
1	1	30.0	0.0	10	2
2	2	296.306	4.693	30	2
3	3	285.317	92.705	30	2
4	4	267.302	136.197	10	2
5	5	242.705	176.336	10	1
6	6	212.132	212.132	30	1
7	7	176.336	242.705	30	1
8	8	136.197	267.302	10	1
9	9	92.705	285.317	10	1
10	10	4.693	296.306	30	1
11	11	0.0	30.0	30	1
12	12	-4.693	296.306	10	1
13	13	-92.705	285.317	10	1
14	14	-136.19	267.302	30	1
15	15	-176.33	242.705	30	1

3.3. Capacity Balancing Algorithm

At the classification phase of the problem, the nodes are given to the vehicles based on the previously mentioned Decision Boundary regions. In certain scenarios, the assignments of nodes to the vehicle can exceed the vehicle's capacity. The capacity balancing algorithm (CBA) is used whenever a vehicle's capacity is surpassed. Machine learning algorithms calculate the probability of each node to each vehicle after determining the decision boundaries. CBA automatically disconnects node of vehicle which have the lowest probability of assignment if the vehicle has surpassed its capacity, allowing that node to be reassigned to the second closest vehicle. This procedure is repeated until the capacities of the vehicles are not exceeded.

A list is generated with the probability values of each vehicle to the nodes. Once machine learning algorithms have evaluated the data, this list would be extracted to new data. **Table 3.7.** includes new dataset with vehicle probability list.

Table 3.7. *New dataset with vehicle's probabilities*

Index	Node No	X-Coord.	Y-Coord	Demand	Vehicle Prediction of LR Model	Vehicle 1 Probability	Vehicle 2 Probability
1	62	4	-6	4	1	0.9999992370292082	0.00000076297079198
2	63	18	-4	394	1	0.999999720737659	0.0000000027926234
3	64	10	-1	3	1	0.9999969212831658	0.0000000030787168
4	65	16	-10	60	1	0.9999999740182084	0.00000025981792
5	66	96	5	899	1	0.9999999995982368	0.000040176319
6	67	60	28	169	1	0.9987049184963722	0.00012950815
7	68	75	-19	10	1	0.9999999999998265	0.0017352679
8	69	107	5	47	1	0.9999999999087481	0.000091251858
9	70	70	-4	37	1	0.9999999997471094	0.000025289059
10	71	110	-2	23	1	0.9999999999972146	0.000027854605
11	72	90	3	25	1	0.9999999996264921	0.0000374
12	73	80	-17	444	1	0.9999999999997864	0.000214
13	74	59	-21	3	1	0.9999999999993793	0.00621
14	75	83	14	99	1	0.9999998777306998	0.00000000122
15	47	46	38	119	1	0.587662138056631l	0.412337862

According to **Table 3.7.**, Vehicle Prediction's columns show the prediction of LR Model assignments. Vehicle 1 Probability and Vehicle 2 Probability columns show the probability of assignment. By using CBA, CBA checks for the capacity of vehicles, if vehicle 1 capacity is exceeded, CBA checks for lowest probability of Vehicle 1 assignment. In this example Node 47 or Index 15 has the lowest probability on Vehicle 1. CBA changes Node 47 assignment into Vehicle 2. An example of CBA in progress on New Dataset shown in **Figure 3.20.**

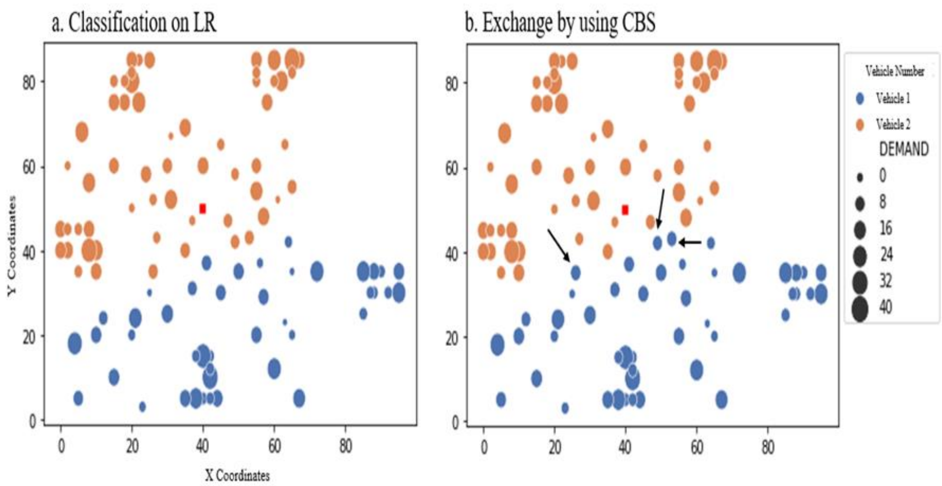


Figure 3.20. *CBA in progress*

Figure 3.20.a displays typical classification of nodes, **Figure 3.20.b** displays classification including CBA check for vehicle exceedance and arrows indicating swap of three nodes between Vehicle 2 from Vehicle 1. In this problem dataset vehicle capacity is 948, an overcapacity is recognized, and utilizing CBA, vehicle 2 capacity is reduced from 987 to 942. Vehicle 1 is swapped 3 nodes from Vehicle 2. The steps of procedure is shown in **Table 3.8**.

Table 3.8. *CBA in progress*

	Assigned Node Numbers by Using LR algorithm	Total Load	Assigned Node Numbers by Using CBA after LR Algorithm	Total Load
Vehicle 1	43	737	46	782
Vehicle 2	57	987	54	942

3.4. Traveling Salesman Problem Mathematical Formulation

The traveling salesman problem aims to minimize the total distance traveled during the visit of all nodes, starting from depot. The parameters and decision variables used in the mathematical model are described as follows [61]:

Index:

i,j : Nodes (Customers)

Sets:

V = (0,..,N) sets of nodes, 0: depot

n = Number of Nodes

Parameters:

d_{ij}: distance from node i to node j

Decision Variables:

x_{ij}: 1, if a vehicle travels from node i to node j, otherwise 0

y_i: positive integer number of node i (which shows the visit order)

Objective Function:

$$Min\ Z = \sum_{i\in V, j\in V} d_{ij}x_{ij} \qquad (3.9)$$

Constraints:

$$\sum_{j\in V\backslash\{i\}} x_{ij} = 1 \qquad \forall i \in V \qquad (3.10)$$

$$\sum_{i\in V\backslash\{j\}} x_{ij} = 1 \qquad \forall j \in V \qquad (3.11)$$

$$y_i - (n+1)x_{ij} \geq y_j - n \qquad \forall i \in V\backslash\{0\}, j \in V\backslash\{0, i\}$$

$$x_{ij} \in \{0,1\} \qquad \forall i \in V, j \in V \qquad (3.12)$$

$$y_i \geq 0 \qquad \forall i \in V \qquad (3.13)$$

The objective function of the TSP model (3.9) minimizes the total traveled distance. Constraints (3.10) and (3.11) ensure that there is only one incoming to each node i and one outgoing node i from each node. Constraint (3.12) is the classical MTZ sub-tour elimination constraint [62]. Constraint (3.13) states that the decision variables x(i, j) will take an integer value of 0-1.

After the classification (clustering) and capacity balancing operations are performed, a column showing all vehicle assignments is added to new dataset file. The vehicle assignments in this column are sorted and divided by the assignment of each vehicle, then TSP formulation with each vehicle is solved separately, the results obtained are then combined to get the CVRP result. The first 15 rows of a vehicle-assigned and sorted data sample are shown in **Table 3.9**.

Table 3.9. *Vehicle-assigned and sorted data sample*

Index	Node No	X-Coord.	Y-Coord.	Demand	Vehicle_Type
1	5	15	30	26	1
2	7	20	50	5	1
3	8	10	43	9	1
4	11	20	65	12	1
5	17	5	30	2	1
6	1	41	49	10	2

Table 3.9. *Vehicle-assigned and sorted data sample* (Continue)

7	3	55	45	13	2
8	9	55	60	16	2
9	10	30	60	16	2
10	20	45	65	9	2
11	27	35	40	16	3
12	4	55	20	19	3
13	12	50	35	19	3
14	21	45	20	11	3
15	22	45	10	18	3

The TSP mathematical model is run separately for each vehicle type in **Table 3.9.**, and then the CVRP solution is constructed by summing up these formulations' objective functions and the solution times. **Table 3.10.** shows TSP results of each vehicle's objective function, its solution time, and the Total CVRP solution for all algorithms in a 3-vehicle dataset.

Table 3.10. *Combined TSP for CVRP*

	Vehicle 1		**Vehicle 2**		**Vehicle 3**		**Total Solution of CVRP**	
	Obj. F.	Time (sec)	Obj. F.	Time (sec)	Obj. F.	Time (sec)	Total Obj. F.	Total Time (Sec)
K-Means	233.80	682.40	236.20	5.40	228.10	15.60	698.10	703.40
LR	181.90	0.60	253.30	233.10	268.70	222.10	703.90	455.80
K-NN	232.70	3.10	255.10	97.40	286.60	31.50	774.40	132.00

4. COMPUTATIONAL RESULTS

Python and Python libraries such as Scikit-learn, Python-MIP, Pandas, and NumPy were used to implement machine learning algorithms and to solve TSP and CVRP in this thesis. Mathematical modeling formulations and machine learning algorithms have been built on a single software platform with the help of these libraries. The results of the test problems were obtained by operating the AMD Ryzen 5 3500 U with Radeon Vega Mobile Gfx 2.10 GHz CPU computer with the GUROBI 9.12 commercial solver on the Python-MIP.

Before obtaining the test results, the CVRP mathematical formulation was run for 6 hours, 24 hours, and 96 hours, and the difference between the objective function of the mathematical model ran for 6 hours and the objective function result of the mathematical model ran for 96 hours was less than 1%. As a result, it was determined that running the CVRP mathematical model for 6 hours would be enough and this requires less time demand for each approach. Testing models for 6 hours also would provide a more balanced comparison of the two models.

4.1. Datasets

Five datasets were used to test the LR, K-NN, and K-Means algorithms. Algorithms for VRP, including 2-3-4-vehicle versions, were performed on these datasets, and experimental results were produced on a total of 45 test problems. For each problem type, the CVRP mathematical modeling formulation was run for 6 hours (21.600 seconds) with the GUROBI 9.12 solver, and the best values obtained by the solver were reported. The 100-node RC101 [22], C101 [22], Tai100a [63], CMT3 [53], and 240-node Golden-1 [64] datasets were used. These datasets' node inputs were used without alteration throughout the experiments. The datasets were derived for different types of VRP, which are often used in the literature, and were selected to reflect various characteristics. RC101 is a random clustered dataset which is shown in **Figure 4.1.**

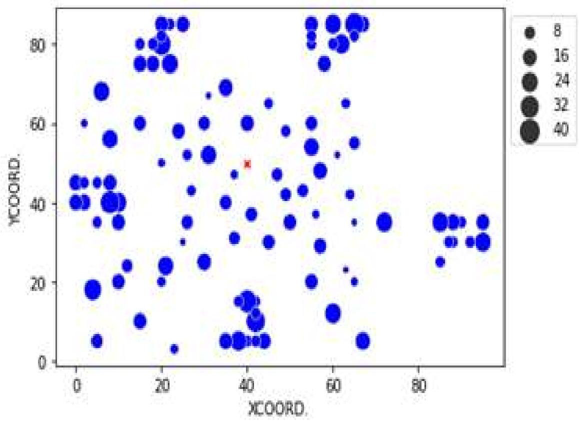


Figure 4.1. *RC101 dataset*

C101 is a dataset consisting of clustered nodes. Depicts on **Figure 4.2.**

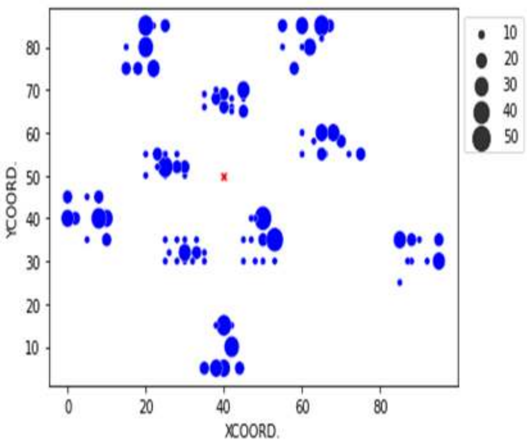


Figure 4.2. *C101 dataset*

In Tai100a dataset, nodes are non-uniformally spread in several clusters. The number of clusters and their compactness are variable. Tai100a Dataset is shown in **Figure 4.3.**

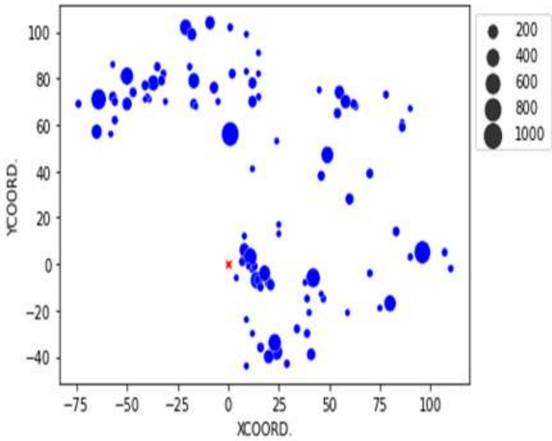


Figure 4.3. *Tai100a dataset*

In CMT3 dataset, nodes are non-uniformly spread but there is no clustering in the dataset. Illustrated on **Figure 4.4.**

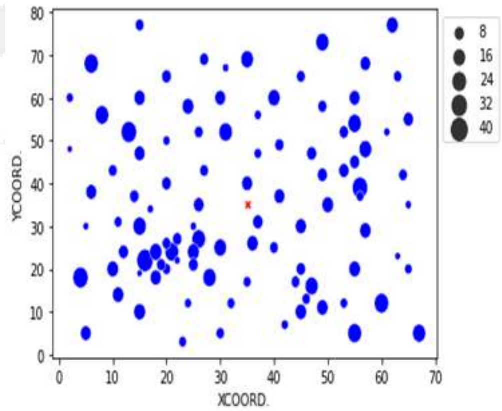


Figure 4.4. *CMT3 dataset*

In Golden-1 dataset, nodes are located in concentric circles around the depot, which is shown in **Figure 4.5**.

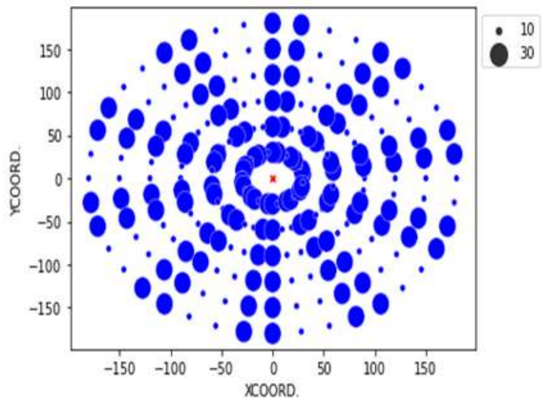


Figure 4.5. *Golden-1 dataset*

The vehicle capacities in the problems were calculated by dividing the total amount of demand in the dataset by the number of vehicles, then this value was increased by 10% to get the actual capacity of our problems. The capacity for each dataset is shown in **Table 4.1**.

Table 4.1. *Capacities of vehicles*

	Capacities of Each Vehicle		
	CVRP with 2 Vehicles	CVRP with 3 Vehicles	CVRP with 4 Vehicles
RC101	948	631	474
C101	996	663	497
Tai100a	8361	5574	4180
Golden-1	2640	1760	1320
CMT3	802	535	400

4.2. CBA results

After classification (clustering) process of machine learning algorithms, vehicle capacity was exceeded in 37 of the 45 test problems, CBA is used to balance the loads of vehicles based on the probability values of machine learning algorithms. **Table 4.2.** shows Dataset and Machine Learning Algorithms which the capacity balancing algorithm is applied and which it is not applied(45 test instances).

Table 4.2. *CBA and machine learning algorithms in datasets; + refer the usage of CBA, and – refer that does not require CBA*

CBA and Machine Learning Algorithms									
Datasets	LR CVRP with			K-NN CVRP with			K-Means CVRP with		
	2-Veh.	3-Veh.	4-Veh.	2-Veh.	3-Veh.	4-Veh.	2-Veh.	3-Veh.	4-Veh.
RC101	+	+	+	+	+	+	-	+	+
C101	+	+	+	+	+	+	+	+	+
Tai100a	-	+	+	-	+	+	+	+	+
Golden_1	+	+	+	+	+	+	-	+	-
CMT3	-	+	+	-	+	+	+	-	+

According to the results obtained from **Table 4.2.**, the capacity balancing algorithm was used by nine out of 15 problems of CVRP with 2 vehicles. For 3-vehicle CVRP and 4-vehicle CVRP problem, only the K-Means algorithm was not needed for the Golden_1 and CMT3 dataset. K-Means algorithm is based on iteration of intra-cluster similarity maximization. It is thought that since K-Means clustering is done as a result of aggregating nodes at certain points, it is assumed that the Golden-1 and CMT3 datasets would not have capacity outages.

4.3. CVRP with 2 vehicles Solution Results

In this section, firstly, it is aimed to analyse the effect of vehicle numbers on algorithms. For this purpose, the algorithms' results for each number of vehicles and the outputs obtained from running CVRP mathematical formulation for 6 hours with the GUROBI 9.12 solver were compared. A comparison of the 2-Vehicles Machine Learning Model solutions and CVRP mathematical formulation solution are presented in **Table 4.3.**

Table 4.3. *Solutions of machine learning models&TSP and CVRP mathematical formulation for 2 vehicles*

	RC101		C101		Tai100a		Golden_1		CMT3	
	Obj. F.	Total Time (sec)	Obj. F.	Total Time (sec)	Obj. F.	Total Time (sec)	Obj. F.	Total Time (sec)	Obj. F.	Total Time (sec)
K-Means	668.7	11011.93	569.87	21600	1046.4	11022.88	4335.3	21600	673.01	3086.32
LR	675.5	11478	573.45	21600	1035.2	10843.48	4337.5	21600	651.3	48.76
K-NN	692.8	10858.63	578.74	21600	1033.8	10821.55	4352	21600	659.4	34.56
CVRP	763	21600	531.98	21600	1148.0	21600	4313.8	21600	651.30	21600

On the RC101 and Tai100a datasets, K-Means&TSP solution performed better than the CVRP with mathematical formulation solution. While LR&TSP solution performed better than the CVRP solution in the RC101 and Tai100a datasets, it achieved the same result as the CVRP solution in the CMT3 dataset. When K-NN&TSP solution is analyzed, it is seen that K-Means&TSP solution of the algorithm exhibits the same behavior. The reason why machine learning algorithms consistently perform better in RC101 and Tai100a datasets is that the node distributions in these datasets are based on random placement, so it is interpreted that the classification performed by machine learning algorithms is more successful than CVRP mathematical formulation solution. Furthermore, when the objective functions and CPU times used are analyzed, it is obtained that all machine learning algorithms in the RC101 dataset produce superior results on average by 10% in half of the time as the CVRP solution. Machine learning & TSP algorithms, in the C101 and Golden_1 dataset, each algorithm used the entire 6-hour time-period. However, in these datasets, none of the machine learning algorithms were able to access CVRP's result, on average, they were able to access the objective function values at 90% proximity. It has been found that machine learning models in the CMT3 dataset approach the objective function value obtained by CVRP by 95% by using 25% less CPU time than CVRP. It is seen that LR&TSP and K-NN&TSP reached the results in less than 1 minute of CPU time on the CMT3 dataset. Therefore, it is obvious that the performance of algorithms also varies depending on the datasets.

To better understand the behavior of machine learning algorithms in relation to datasets, the routes of the Tai100a dataset, which outperforms CVRP in all machine learning algorithms in 2-vehicle datasets, are plotted and displayed in **Figure 4.6.**

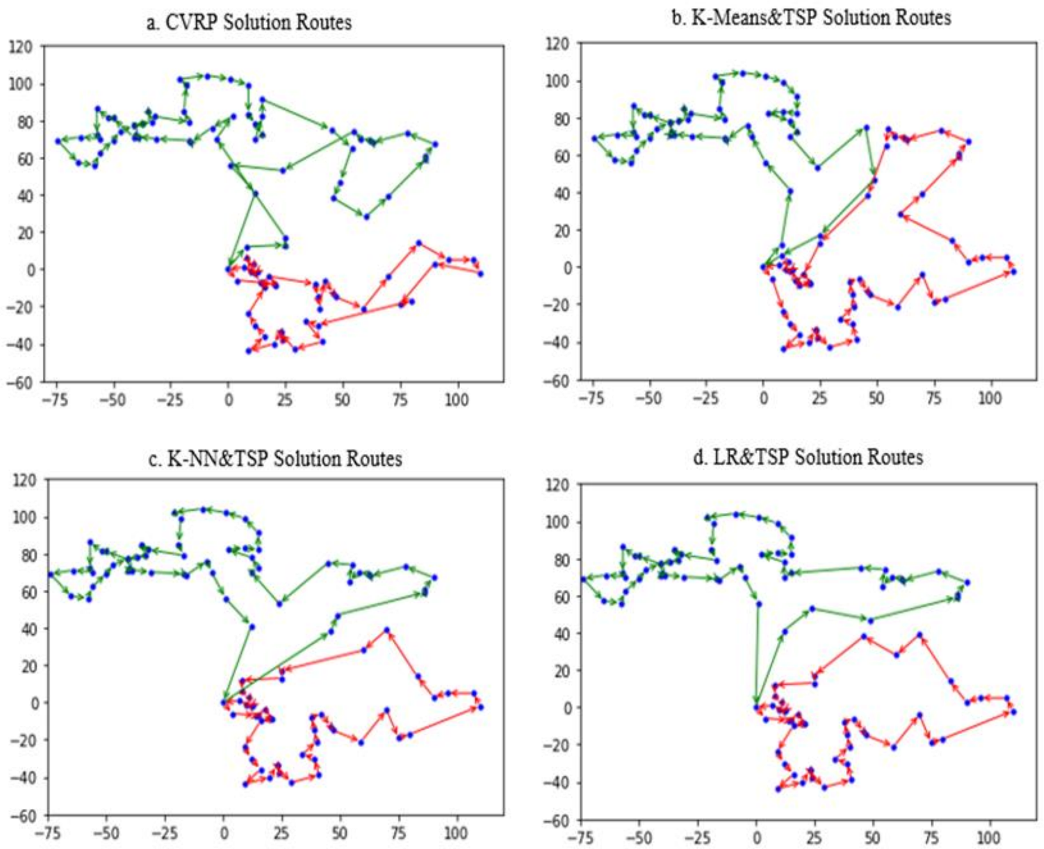


Figure 4.6. *Routes of Tai100a*

The number of nodes, vehicle capacities and route lengths of the algorithms used in the Tai100a dataset solution are presented in **Table 4.4**. In this dataset, only the K-Means algorithm used the capacity balancing algorithm.

Table 4.4. *Nodes-Vehicle Capacity-Route Length of Algorithms in Tai100a dataset*

	CVRP		**K-Means**		**LR**		**K-NN**	
	1.Veh.	2.Veh.	1.Veh.	2.Veh.	1.Veh.	2.Veh.	1.Veh.	2.Veh.
Assigned Number of Nodes	40	60	52	48	46	54	45	55
Vehicle's Load	6957	8246	8055	7148	7364	7839	7245	7958
Length of Routes to Assigned Nodes	425.76	722.24	523.87	522.52	431.74	602.13	441.96	593.28
Total length of Routes	1148		1046.39		1033.87		1035.24	

When the CVRP mathematical formulation solution result for the Tai100a dataset is reviewed, it is seen that the distribution of node numbers on routes is more unevenly located than other algorithms. Routes also vary depending on the amount of demand, however, the fact that the number of nodes on the routes of machine learning algorithms is closer to each other indicates that the vehicle’s loads are distributed evenly. As a result of the solutions of K-Means&TSP, it seems that the number of nodes to which the vehicles are assigned is formed in the closest way to each other by the help of CBA. However, the objective function value obtained from the solution of K-Means&TSP is higher by a small difference compared to the other two algorithms.

When the route structure was analyzed, the fact that the lengths of the CVRP solution's routes were positioned significantly different from each other also generated the highest value in the objective function. Because the node numbers are distributed similarly in the solutions generated with machine learning algorithms, the route lengths are equally similar to each other, and it can be interpreted that the objective function values are lower than the CVRP mathematical formulation's solution.

4.4. CVRP with 3 Vehicles Solution Results

Table 4.5. compares the performance of algorithms as a consequence of solving 3-vehicle Machine Learning Models and subsequent classifications using TSP and CVRP mathematical formulation.

Table 4.5. *Solutions of machine learning models&TSP and CVRP mathematical formulation for 3 vehicles*

	RC101		**C101**		**Tai100a**		**Golden_1**		**CMT3**	
	Obj. F.	Total Time (sec)	Obj. F.	Total Time (sec)	Obj. F.	Total Time (sec)	Obj. F.	Total Time (sec)	Obj. F.	Total Time (sec)
K-Means	698.1	703.47	583.6	3994.05	1177.2	98.25	4527.3	8097.1	685.4	12.46
LR	704.0	455.89	579.1	14691.14	1051.1	2562.25	4393.3	10266.22	664.3	2.44
K-NN	774.5	132.13	654.7	14888.13	1439.9	117.31	4602.7	21600	706.8	9.51
CVRP	720.2	21600	564.0	21600	1209.3	21600	4521.2	21600	696.9	21600

The solution of K-Means&TSP turned out to be better than the CVRP mathematical solution in RC101, Tai100a and CMT3 datasets. Especially considering the CPU time of the CVRP mathematical model, Tai100a (98.25 sec.) and CMT3 (12.46 sec.) in the datasets, it is observed that K-Means&TSP achieves better results in a much shorter CPU time. When the LR&TSP solution is examined, it is seen that the algorithm achieves better results in less CPU time in all datasets except the C101 dataset than the CVRP solution. Especially in the CMT3 dataset, a much better result was achieved in 2.44 seconds than CVRP mathematical formulation. The results of the K-NN&TSP solution, when the averages of all datasets were evaluated, it was found that the K-NN&TSP approximates the result at the level of 90% using about 30% of the CPU time.

When 3-Vehicle Machine Learning algorithms and the solution results of TSP are analyzed in the C101 dataset, it is seen that all of them are higher than the CVRP's solution (**Table 4.5.**). Therefore, to closely examine the behavior of the algorithm on this dataset, the routes of the algorithms according to C101 dataset were plotted in **Figure 4.7.** and analyzed.

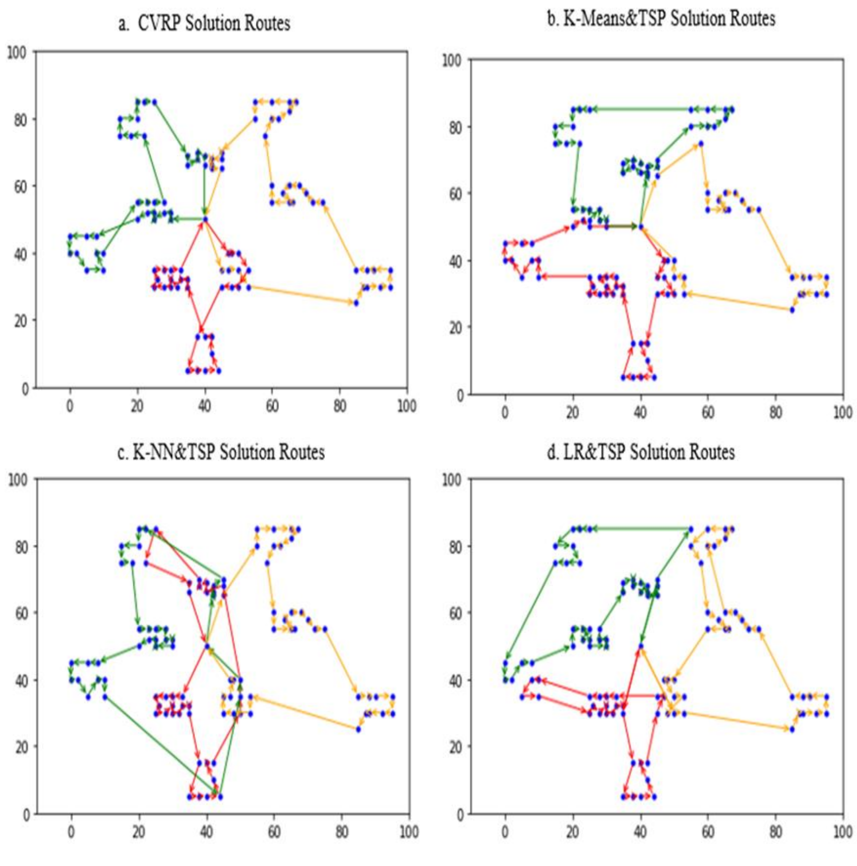


Figure 4.7. *Routes of C101*

Table 4.6. shows the number of nodes, the capacity of the vehicles employed, and the route lengths of the methods used in the C101 dataset solution. After classifying and grouping all machine learning algorithms, a capacity balancing algorithm was performed.

Table 4.6. *Nodes-Vehicle Capacities-Route Lengths of Algorithms in C101 dataset*

	CVRP			**K-Means**			**LR**			**K-NN**		
	1.Veh	2.Veh	3.Veh	1.Veh	2.Veh	3.Veh	1.Veh	2.Veh	3.Veh	1.Veh	2.Veh	3.Veh
Assigned Number of Nodes	38	28	34	40	25	35	28	37	35	28	38	34
Vehicle's Load	650	510	650	660	490	660	490	660	660	510	640	660
Length of Routes to Assigned Nodes	227.9	142.0	194.0	208.6	188.1	186.8	152.7	210.1	216.2	181.6	243.1	229.9
Total length of Routes		564.07			583.61			579.16			654.76	

The capacities of the vehicles for the C101 dataset were determined as 663. It is worth noting that the use of vehicle capacities is very close to the upper limit of capacity in solutions. The reason for this situation can be explained as the fact that all three algorithms used capacity balancing algorithm. When the CVRP solution is studied in 1. and 2. vehicles route, it was seen that the vehicle routes intersected each other, as shown in **Figure 4.7.a**. It was considered that the reason for this circumstance is due to vehicle capacity limitations. There is no situation where the routes intersect each other in K-Means&TSP and LR&TSP solutions. It is thought that the objective function values closest to the CVRP solution of this situation causes the coming from these two solutions. In addition, this dataset includes clustered data, as is known from the literature. LR&TSP algorithm finds the closest result to the CVRP mathematical model among other machine learning algorithms. Similar to CVRP solution, the overlap of the routes of the 3 vehicles can be observed as a result of K-NN&TSP solution. It can be concluded that the reason why the K-NN&TSP solution performs the worst in terms of objective functions is due to the placement of routes (**Figure 4.7.c.**).

In the K-Means&TSP solution, the route with the highest number of nodes was encountered (**Table 4.6.**). However, the length of the route with the maximum number of 40 nodes was 208.6, and it was observed that the length of the cluster, which is located in a balanced way close to each other, occurs at shorter distances than routes where the number of nodes is less. (For example, K-NN 38 nodes:243.1). One of the reasons that machine learning algorithms find a worse objective function than CVRP mathematical formulation is that the dataset used in supervised learning consists of randomly and non-clustered distributed nodes, while the C101 dataset can be explained as a clustered dataset.

4.5. CVRP with 4 Vehicles Solution Results

The comparison of the 4-Vehicle Machine Learning Model with the Result of CVRP Mathematical Modeling Formulation is presented in **Table 4.7**.

Table 4.7. *Solutions of machine learning models&TSP and CVRP mathematical formulation for 4 vehicles*

	RC101		C101		Tai100a		Golden_1		CMT3	
	Obj. F.	Total Time (sec)	Obj. F.	Total Time (sec)	Obj. F.	Total Time (sec)	Obj. F.	Total Time (sec)	Obj. F.	Total Time (sec)
K-Means	795.8	172.64	665.5	790.22	1330.6	5418.18	4716.4	5396.5	711.5	16.05
LR	802.0	749.11	684.5	5663.93	1360.8	5571.99	4915.9	7432.02	701.1	8.44
K-NN	836.9	417.46	714.7	454.51	1635.1	10815.48	4822.7	5483.51	710.3	38.87
CVRP	814.3	21600	610.8	21600	1309.5	21600	4800.1	21600	719.7	21600

The results of the K-Means&TSP algorithm turned out to be better for in the RC101, Golden_1 and CMT3 datasets than CVRP mathematical formulation's solution. In the C101 and Tai100a datasets, which did not have a better result as an objective function, the solution given by the GUROBI 9.12 commercial solver for CVRP was approached by 92% and 98%, respectively, and the times used compared to the commercial solver were also quite low (790 sec; 1330 sec). In its solution with LR&TSP, a better result was obtained in the RC101 and CMT3 datasets, and this solution was achieved in just 8.44 seconds in the CMT3 dataset. In the datasets C101, Tai100a and Golden_1, the proximity percentages of the objective functions according to CVRP mathematical formulation's solution were 90%, 96% and 97%, respectively. K-NN&TSP solutions, on the other hand, showed a similar trend to the 3-vehicles solutions and derived the highest objective function value in the RC101, C101, Tai100a datasets. The CPU times used between 25% and 30% of the time is given for CVRP solution. Therefore, it has been observed that the proposed algorithms have access to better solutions in short CPU times.

It is observed that 4-vehicle machine learning algorithms achieve better results, especially in the CMT3 dataset. For this purpose, the routes formed to monitor the behavior of algorithms in this dataset are shown in **Figure 4.8**.

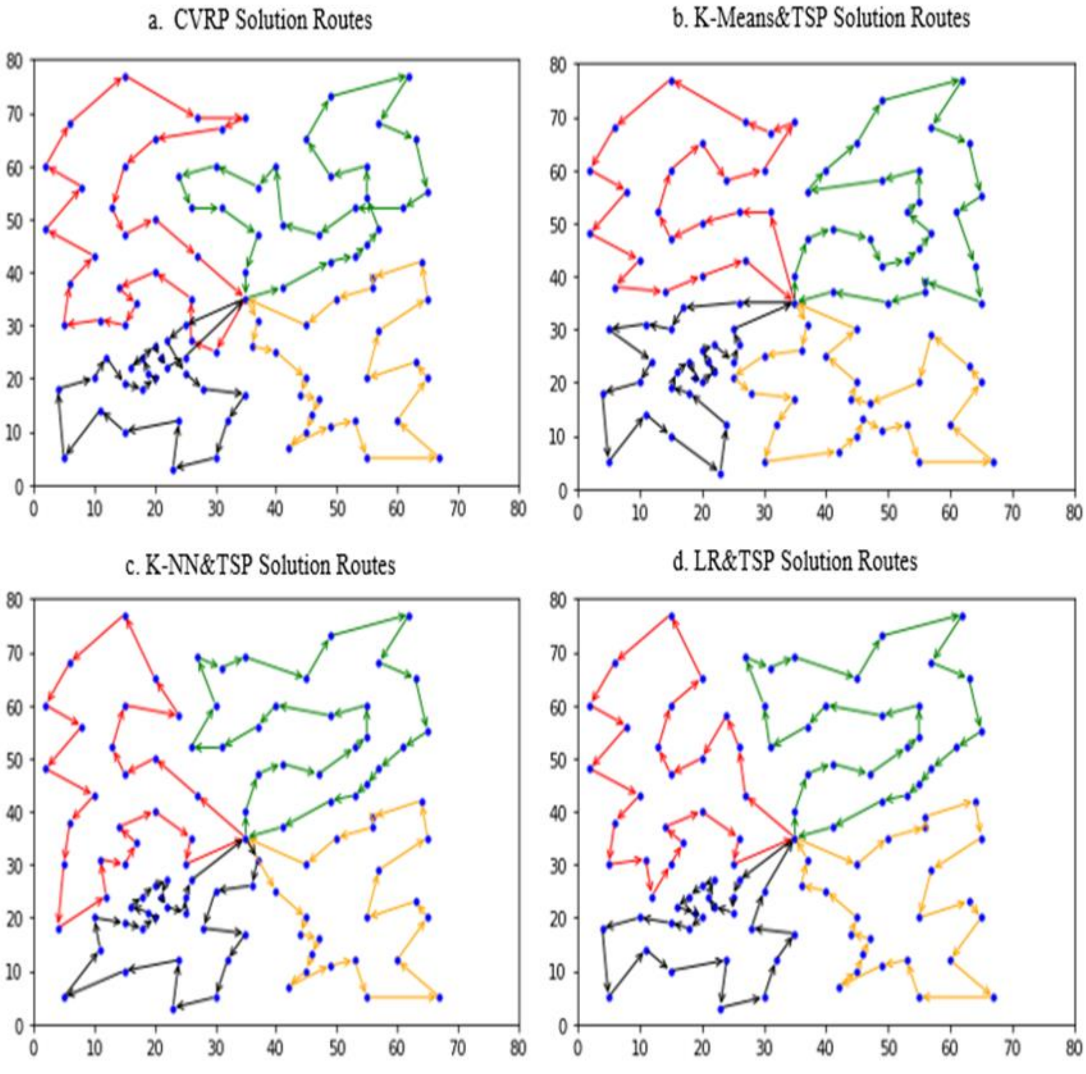


Figure 4.8. *Routes of CMT3*

The number of nodes, vehicle capacities and route lengths of the algorithms used in the CMT3 dataset solution are presented in **Table 4.8**. All machine learning algorithms have used the capacity balancing algorithm.

Table 4.8. *Nodes-Vehicle Capacity-Route Lengths of Algorithms in CMT3 dataset*

	CVRP				**K-Means**			
	1.Veh.	2.Veh.	3.Veh.	4.Veh.	1.Veh.	2.Veh.	3.Veh.	4.Veh.
Assigned Number of Nodes	24	26	25	25	25	22	26	27
Vehicle's Load	361	378	354	365	371	322	385	380
Length of Routes to Assigned Nodes	162.01	191.75	205.69	160.24	174.29	191.79	155.44	190.04
Total length of Routes	719.69				711.56			
	LR				**K-NN**			
	1.Veh.	2.Veh.	3.Veh.	4.Veh.	1.Veh.	2.Veh.	3.Veh.	4.Veh.
Assigned Number of Nodes	24	27	24	25	24	28	22	26
Vehicle's Load	312	386	361	399	338	395	329	396
Length of Routes to Assigned Nodes	197.32	191.45	162.01	150.35	206.45	197.33	159.5	147.07
Total length of Routes	701.13				710.35			

When we look at the solutions on **Figure 4.8.**, in the CVRP routes of 1. and 4 vehicles; it has been observed that vehicle routes pass through each other. The situation where vehicle routes pass through each other is observed only in the CVRP solution (**Figure 4.8.a.**). It is believed that such a solution leads to the formation of the highest value of the CVRP's objective function. It is seen that the routes obtained from the LR&TSP and K-NN&TSP solutions are very similar to each other (**Figure 4.8.c.; Figure 4.8.d.**). When **Table 4.8.** is analyzed, it is observed that the lengths of the vehicle routes are also formed very closely to each other. In addition, it seems that the most successful objective function was achieved by the LR&TSP solution. When the LR&TSP solution is examined (**Figure 4.8.d.&Table 4.8**), the node numbers of the routes were formed very close to each other and with low distance totals. It is noteworthy that the K-NN&TSP

solution with the lowest route length (147.07) is quite close to the K-Means&TSP solution in terms of route schemas, although it differs quite significantly in terms of objective functions. The reason for this situation can be interpreted as the fact that the CMT3 dataset consists of randomly distributed nodes, which can lead to similar objective function values even for different route placements.

4.6. K-Means&TSP Solution Results

Another analysis carried out in this chapter is the comparison of algorithms with the CVRP solution when all the vehicle numbers are taken into account. For this purpose, in **Table 4.9.**, the proximity of K-means& TSP and its solutions to CVRP solutions in terms of objective function on a vehicle-by-vehicle (%) basis and their proximity in terms of CPU time required are given. Objective Function (Obj. F.) Gap(%) and Time Gap(%) calculations were developed in order to better understand the tables. Obj. F. Gap(%) is calculated as Machine Learning Algorithms&TSP Solution/CVRP Mathematical Solution*100 Time Gap(%) is equal to Machine Learning Algorithms used time/CVRP used Time *100.

Table 4.9. *K-Means&TSP solution comparison with CVRP according to Obj. F. Gap(%) and Time Gap(%)*

	K-Means					
	2-Vehicles		3-Vehicles		4-Vehicles	
	Obj. F. Gap(%)	Time Gap(%)	Obj. F. Gap(%)	Time Gap(%)	Obj. F. Gap(%)	Time Gap(%)
RC101	114.1	50.98	97.42	3.26	102.33	0.8
C101	93.35	100	96.65	18.49	91.77	3.66
Tai100a	109.71	51.03	102.73	0.45	98.41	25.08
Golden_1	99.5	100	99.87	37.49	101.78	24.98
CMT3	96.77	14.29	101.68	0.06	101.15	0.07
On Average	102.686	63.26	99.67	11.95	99.088	10.918
On overall with all datasets and vehicle numbers Obj. F. Gap(%) is 100.48% in 28.70% Time Gap.						

Table 4.9. when analyzed, it is seen that in 3 vehicle and 4 vehicle instances for K-Means&TSP, the objective function results are not better than CVRP mathematical formulation obtained on general average. In the same problem examples, the time required to obtain a solution is on average about 11.5% of the time given to CVRP. However, in 2-vehicle examples, this solution technique was more successful, especially

in randomly distributed datasets of nodes, the result is better than the CVRP solution result. In these samples, 63.26% of the time of CVRP mathematical formulation was also used. Therefore, the K-Means algorithm has generally provided more successful solutions in examples with a low number of vehicles.

4.7. K-NN&TSP Solution Results

In **Table 4.10.**, the gap of K-NN&TSP and its solutions to CVRP mathematical formulation solutions in terms of objective function and CPU time requirement is shown on each vehicle type (percent).

Table 4.10. *K-NN&TSP solution comparison with CVRP according to Obj. F. Gap(%) and Time Gap(%)*

K-NN						
	2-Vehicles		3-Vehicles		4-Vehicles	
	Obj. F. Gap(%)	Time Gap(%)	Obj. F. Gap(%)	Time Gap(%)	Obj. F. Gap(%)	Time Gap(%)
RC101	110.12	50.27	87.81	0.61	97.3	1.93
C101	91.92	100	86.15	68.93	85.46	2.1
Tai100a	111.04	50.1	83.98	0.54	80.09	50.03
Golden_1	99.12	100	98.23	100	99.53	25.39
CMT3	98.77	0.16	98.6	0.04	101.31	0.18
On Average	102.194	60.106	90.954	34.024	92.738	15.926
On overall with all datasets and vehicle numbers Obj. F. Gap(%) is 95.3% in 36.68% Time Gap.						

When the K-NN&TSP solutions were investigated, it was discovered that K-NN&TSP used 36.68 percent of the total time to achieve a result with a 4.70 percent worse objective function than CVRP solutions. According to CVRP solution formulation, the K-NN&TSP solution's effectiveness declines as the number of vehicles increases. K-NN&TSP solution, according to the CVRP solution, was the solution method with the shortest CPU time. As a result, this solution technique may be a good choice in situations when decision makers need alternative quicker solutions.

4.8. LR&TSP Solution Results

On a vehicle-by-vehicle basis, **Table 4.11.** shows the proximity of LR&TSP and its solutions to CVRP mathematical solutions in terms of objective function and CPU time required (percent).

Table 4.11. *LR&TSP solution comparison with CVRP according to Obj. F. Gap(%) and Time Gap(%)*

LR						
	2-Vehicles		3-Vehicles		4-Vehicles	
	Obj. F. Gap(%)	Time Gap(%	Obj. F. Gap(%)	Time Gap(%	Obj. F. Gap(%)	Time Gap(%
RC101	112.95	53.14	96.61	2.11	101.53	3.47
C101	92.77	100	97.4	68.01	89.23	26.22
Tai100a	110.89	50.2	115.05	11.86	96.22	25.8
Golden_1	99.45	100	102.91	47.53	97.65	34.41
CMT3	100	0.23	104.9	0.01	102.65	0.04
On	103.212	60.714	103.374	25.904	97.456	17.988
On overall with all dataset and vehicle numbers Obj. F. Gap(%) is 101.34% in 34.86% Time Gap.						

LR&TSP solution was able to achieve an average of 1.34% better result than CVRP mathematical formulation solution by using 34.86% of the time used by CVRP in total. When LR&TSP solutions were analyzed, even if there was a decrease compared to CVRP for 4-vehicle test instances, it was the most successful method in terms of all test problems. In addition, since about a third of the time required for the CVRP solution is used as the solution time, decision makers may prefer the LR&TSP solution as a method that provides both fast and successful results.

5. CONCLUSION

In this thesis, a method for solving VRP using supervised and unsupervised machine learning methods is applied. LR and K-NN algorithms from supervised machine learning algorithms and K-Means algorithms from unsupervised machine learning algorithms have been included, because they are the most frequently applied algorithms in the literature. As a method, CVRP was solved primarily on a dataset from the literature, and supervised machine learning algorithms were trained through this dataset solution. The trained algorithms have been used to classify or cluster on other datasets that are often used for various types of vehicle routing problems from the literature. With the classifications obtained from supervised machine learning algorithms, the clustering results obtained from unsupervised machine learning algorithms are solved in such a way that each cluster is formed into a single TSP. When the nodes assigned to the vehicles exceed the vehicle's capacity, a procedure called the capacity balancing algorithm is applied. To the best of author's knowledge, such a study has not been found in the literature.

Of the proposed hybrid approaches, LR&TSP solutions have been the most successful algorithm in terms of performance, LR&TSP algorithm has achieved better objective function values on average within less CPU time than CVRP mathematical model. After all, the three proposed methods were able to achieve quite successful results in much less time than the solution of the CVRP mathematical model. For the following studies, the proposed methods can be made to solve more complex vehicle routing problems by including features that will include time window vehicle routing problem.

REFERENCES

[1] Prescient & Strategic Intelligence. (2020). On-Demand logistics market report: By vehicle type (light commercial vehicle, medium/heavy commercial vehicle), end use (B2B, B2C), application (e-commerce, ındustrial, moving and shifting, P2P delivery) – latest trends, recent developments, and demand forecast through 2030, https://www.psmarketresearch.com/market-analysis/on-demand-logistics-market. (Date of Access: 11.07.2022)

[2] Tek, Ö.B. and Ozgul, E. (2010). *Modern pazarlama ilkeleri: Uygulamalı yönetimsel yaklaşım.* 3. baskı, İzmir: Birleşik Matbaacılık.

[3] Toth, P. and Vigo, D. (2000). *The vehicle routing problem*. SIAM.

[4] Dondo, R. and Cerdá, J. (2007). A cluster-based optimization approach for the multi depot heterogeneous fleet vehicle routing problem with time windows. *European Journal of Operational Research,* 176, 1478-1507.

[5] Asis, L.S., Eduardo, C. and Grossmann E.I. (2021). A MILP-based clustering strategy for integrating the operational management of crude oil supply. *Computers & Chemical Engineering*, 145.

[6] Rautela, A., Sharma, S.K. and Bhardwaj, P. (2019). Distribution planning using capacitated clustering and vehicle routing problem. *Journal of Advances in Management Research*, JAMR-12-2018-0113.

[7] Geetha, S., Poonthalir, G. and Vanathi, P.T. (2009). Improved K-means algorithm for capacitated clustering problem. *INFOCOMP Journal of Computer Science*, 8 (4), 52–59.

[8] Alesiani, F., Ermiş, G. and Konstantinos, G. (2022). Constrained clustering for the capacitated vehicle routing problem (CC-CVRP). *Applied Artificial Intelligence*, 36 (1).

[9] Mostafa, N. and Eltawir A. (2017). Solving the heterogeneous capacitated vehicle routing problem using k-means clustering and valid inequalities. *Proceedings of the International Conference on Industrial Engineering and Operations Management Rabat,* Morocco.

[10] Czuba, P. and Pierzchala, D. (2021). Machine learning methods for solving vehicle routing problems.

[11] Dantzig, G.B. and Ramser, J.H. (1959). The truck dispatching problem. *Management Science*, 6 (1), 80-91.

[12] Clarke, G. and Wright, J.W. (1964). Scheduling of vehicles from a depot to a number of delivery points. *Operations Research*, 12, 568-581.

[13] Laporte, G. and Osman, H.İ. (1995). Routing problems: A bibliography, *Annals of Operations Research*, 227-262.

[14] Toth, P. and Vigo, D. (2014). *Vehicle routing problems, methods and applications*. SIAM.

[15] Laporte, G. (1992). The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, 59 (3), 345-358.

[16] Kumar, S.N. and Panneerselvam, R. (2012). A survey on the vehicle routing problem and ıts variants. *Intelligent Information Management*, 4, 66-74.

[17] Laporte, G., Desrochers, M. and Nobert, Y. (1984). Two exact algorithms for the distance constrained vehicle routing problem. *Networks*, 14, 161-172.

[18] Laporte, G., Nobert, Y. and Desrochers, M. (1985). Optimal routing under capacity and distance restrictions. *Operations Research*, 33, 1050-1073.

[19] Pullen, H. and Webb, M. (1967). A computer application to a transport scheduling problem. *Computer Journal*, 10, 10-13.

[20] Knight, K. and Hofer, J. (1968). Vehicle scheduling with timed and connected calls: A case study. *Operational Research Quarterly*, 19, 299-310.

[21] Madsen, O.B.G. (1976). Optimal scheduling of trucks-A routing problem with tight due times for delivery. H. Strobel, R. Genser and M. Etschmaier (Editors), In *Optimization applied to transportation systems* (p. 126-136). IIASA, International Institute for Applied System Analysis, Laxenburgh, Austria.

[22] Solomon, M.M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, 35, 254-265.

[23] Martello, S. and Toth P. (1990). *Knapsack problems: Algorithms and computer ımplementations.* Wiley, Chichester, UK.

[24] Deif, I. and Bodin, L.D. (1984). Extension of the clarke and wright algorithm for solving the vehicle routing problem with backhauling. *Proceedings of the Babson College Conference on Software Uses in Transportation and Logistics Management*, pp. 75-96, Babson Park, MA. A. Kidder (Editor).

[25] Savelsbergh, M.W.P. and Sol, M. (1995). The general pickup and delivery problem. *Transportation Science*, 29, 17-29.

[26] Desrosiers, J., Dumas, Y., Solomon, M.M. and Soumis, F. (1995). Time constrained routing and scheduling. M.O. Ball, T.L. Magnanti, C.L. Monma, and G.L. Nemhauser (Editors), In *Network routing, handbooks in operations research and management science 8* (pp. 35-139). North-Holland, Amsterdam.

[27] Solomon, M.M. and Desrosiers, J. (1988). Time window constrained routing and scheduling problems. *Transportation Science*, 22, 1-13.

[28] Cho, Yuh-Jen. and Wang, Sheng-Der. (2005). A threshold accepting metaheuristics for the vehicle routing problem with backhauls and time window. *Journal of the Eastern Asia Society for Transportation Studies*, 6, 3022-3037.

[29] Bent, R. and Van Hentenryck, P. (2004). A two-stage hybrid local search for the vehicle routing problem with time windows. *Transportation Science*, 38 (4), 515-530.

[30] Christofides, N., Mingozzi A. and Toth P. (1981). State-space relaxation procedures for the computation of bounds to routing problems. *Networks*, 11, 145-164.

[31] Agarwal, Y., Mathur, K., and Salkin, H.M. (1989). A set-partitioning-based exact algorithm for the vehicle routing problem. *Networks*, 19, 731-749.

[32] Osman, I.H. (1993). Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. *Annals of Operations Research*, 41, 421-451.

[33] Glover, F. and Laguna, M. (1993). Tabu search. C.R. Reeves (Editor), In *Modern heuristic techniques for combinatorial problems* (pp. 70-150), Blackwell, Oxford, UK.

[34] Barbarosoglu, G. and Ozgur, D. (1999). A tabu search algorithm for the vehicle routing problem. *Computers and Operations Research*, 26, 255-270.

[35] Ropke, S. and Pisinger, D. (2006). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science*, 40, 455-472.

[36] Holland, J. H. (1975). *Adaptation in natural and Artificial systems*. Ann Arbor, MI: The University of Michigan Press.

[37] Colorni, A., Dorigo, M. and Maniezzo V. (1991). Distributed optimization by ant colonies. F. Varela and P. Bourgine (Editor), In *Proceedings of the European Conference on Artificial Life*, Elsevier, Amsterdam.

[38] Blum, C. and Roli, A. (2003). Metaheuristics in combinatorial optimization: overview and conceptual comparison. *ACM Computing Surveys*, 35 (3), 268-308.

[39] Hopfield, J. J. and Tank, D.W. (1985). 'Neural' computation of decisions in optimization problems. *Biological Cybernetics*, 52 (3), 141–152.

[40] Baltean-Lugojan, R., Bonami, P., Misener, R. and Tramontani, A. (2018). Selecting cutting planes for quadratic semidefinite outer approximation via trained neural networks.

[41] He, H., Daume III, H. and Eisner, J.M. (2014). Learning to search in branch and bound algorithms in Advances in neural information processing systems. 3293-3301.

[42] Khalil, E.B., Dilkina, B., Nemhauser, G.L., Ahmed, S. and Shao, Y. (2017). Learning to run heuristics in tree search. *International Joint Conferences on Artificial Intelligence Organization*, 659– 666.

[43] Khalil, E.B., Le Bodic, P., Song, L., Nemhauser, G. and Dilkina, B. (2016). Learning to branch inmixed integer programming. *Thirtieth Association for the Advancement of Artificial Intelligence Conference on Artificial Intelligence.*

[44] Lederman, G., Rabe, M.N., Lee E.A. and Seshia, S.A. (2018). Learning heuristics for automated reasoning through deep reinforcement learning. arXiv preprint arXiv:1807.08058.

[45] Amizadeh, S., Matusevych, S. and Weimer, M. (2018). Learning to solve circuit-sat: An unsupervised differentiable approach.

[46] Beasley, J.E. (1983). Route first-cluster second methods for vehicle routing. *Omega*, 11 (4), 403-408.

[47] Montoya, J.A., Guéret, C., Mendoza, J.E. and Villegas, J.G. (2014). A route first cluster-second heuristic for the Green Vehicle Routing Problem. *ROADEF 2014*, Bordeaux, France.

[48] Fisher, M.L. and Jaikumar, R. (1981). A generalized assignment heuristic for vehicle routing. *Networks*, 11 (2), 109–124.

[49] Christofides, N., Mingozzi, A., Toth, P., Sandi, C. (1979). *Combinatorial optimization*. Chichester, UK: Wiley, 315-338.

[50] Hand, D. J. (2007). Principles of data mining. *Drug Safety*, 30 (7), 621–622.

[51] Wu, X., Kumar, V., Quinlan, J. R., Ghosh, J., Yang Q. (2008). Top 10 algorithms in data mining. *Knowledge and Information Systems*, 14 (1), 1–37.

[52] Bock, J.R. and Gough, D.A. (2001). Predicting protein–protein interactions from primary structure. *Bioinformatics*, 17 (5), 455–460.

[53] Mangasarian, O.L., Street, W.N. and Wolberg W.H. (1995). Breast cancer diagnosis and prognosis via linear programming. *Operations Research*, 43 (4), 570–577.

[54] Min, J.H. and Lee, Y.C. (2005). Bankruptcy prediction using support vector machine with optimal choice of kernel function parameters. *Expert Systems with Applications*, 28 (4), 603–614.

[55] Sammut, C. and Webb, G.I. (2011). *Encyclopedia of machine learning*. Springer, 288.

[56] Cramer, J.S. (2002). *The origins of logistic regression*. Vol. 119, Tinbergen Institute, 167–178.

[57] Tikhonov, A. N. (1943). *On the stability of inverse problems. Doklady Akademi Nauk SSSR,* 39 (5), 195–198.

[58] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society*, 58 (1), 267–288.

[59] Fix, E. and Hodges, L. (1951). *Discriminatory analysis, nonparametric discrimination: Consistency properties*. Randolph Field, Texas: USAF School of Aviation Medicine.

[60] https://scikit-learn.org/stable/glossary.html#term-random_state (Date of Access: 11.07.2022)

[61] Applegate, D.L., Bixby, R.E., Chantal, V. and Cook, W.J. (2006). *The traveling salesman problem: a computational study*. Princeton University Press.

[62] Miller, E., Tucker A.W. and Zemlin R.A. (1960). Integer programming formulations and traveling salesman problems. *Journal of the ACM*, 7, 326-329.

[63] Rochat, Y. and Taillard, E. (1995). Probabilistic diversification and ıntensification in local search for vehicle routing. *Journal of Heuristics*, 1 (1), 147-167.

[64] Golden, B.L., Raghavan, S., Wasil, E.A. (2008). *The vehicle routing problem: Latest advances and new challenges*. Operations Research/Computer Science Interfaces Series, Springer.

CURRICULUM VITAE

Name Surname : Özgür SANLI

Foreign Language : English

<u>Education and Professional Background:</u>

2004, War Academy

15 years, Turkish Army