



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΤΕΧΝΟΛΟΓΙΑΣ ΛΟΓΙΣΜΙΚΟΥ

Δικτυακή Υπηρεσία χωρικής βελτιστοποίησης με
αρχιτεκτονικής ενοποίησης με υπηρεσίες web
τρίτων μερών

Μελέτη και υλοποίηση

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Νικόλα Τασιόπουλου

Επιβλέπων: Βασίλειος Βεσκούκης
Καθηγητής Ε.Μ.Π.

Αθήνα, Μάρτιος 2024



Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών
Εργαστηρίου Τεχνολογίας Λογισμικού

Δικτυακή Υπηρεσία χωρικής βελτιστοποίησης με αρχιτεκτονικής ενοποίησης με υπηρεσίες web τρίτων μερών

Μελέτη και υλοποίηση

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Νικόλα Τασιόπουλου

Επιβλέπων: Βασίλειος Βεσκούκης
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 27^η Μαρτίου, 2024.

.....
Βασίλειος Βεσκούκης
Καθηγητής Ε.Μ.Π.

.....
Νικόλαος Παπασπύρου
Καθηγητής Ε.Μ.Π.

.....
Αριστείδης Παγουρτζής
Καθηγητής Ε.Μ.Π.

Αθήνα, Μάρτιος 2024

.....
ΝΙΚΟΛΑΑΣ ΤΑΣΙΟΠΟΥΛΟΣ
Διπλωματούχος Ηλεκτρολόγος Μηχανικός
και Μηχανικός Υπολογιστών Ε.Μ.Π.

Copyright © – All rights reserved Νικόλαας Τασιόπουλος, 2024.
Με επιφύλαξη παντός δικαιώματος.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

στην οικογένεια μου

Περίληψη

Τα αυξανόμενα κόστη και η αναποτελεσματικότητα στη μεταφορά εμπορευμάτων, ειδικά στο αστικό περιβάλλον όπου η παράδοση "τελευταίου μιλίου" αντιπροσωπεύει σημαντικό μέρος των συνολικών εξόδων μεταφοράς, αποτελούν σημαντική πρόκληση για τον τομέα των μεταφορών. Αυτή η διπλωματική εργασία παρουσιάζει μια εμπειριστατωμένη έρευνα και ανάπτυξη μιας αρχιτεκτονικής λογισμικού, όπου βελτιστοποιεί χωρικά δεδομένα που ενσωματώνονται τρίτες υπηρεσίες web, με στόχο την ενίσχυση της αποδοτικότητας και της ταχύτητας παράδοσης εμπορευμάτων σε μία περιοχή. Ως παράδειγμα χρησιμοποιήσαμε την περιοχή της Αττικής. Μέσω της εξερεύνησης μοντέλων συσταδοποίησης για βελτιστοποιημένες διαδικασίες διανομής, η έρευνα χρησιμοποιεί διαδρομές οδηγών από τον Οκτώβριο του 2023, αξιολογεί υφιστάμενες ομάδες και εφαρμόζει το Open Source Routing Machine (OSRM) για ακριβείς εκτιμήσεις διαδρομής και χρόνου, σε συνδυασμό με τα γεωχωρικά όρια των ταχυδρομικών κωδικών στην Ελλάδα. Η μελέτη εφαρμόζει το KMeans και τις παραλλαγές του, MiniBatch KMeans και KMeans Constrained, για την ομαδοποίηση. Επιπλέον, η διπλωματική διερευνά τη δημιουργία ενός RESTful API, σχεδιασμένου για ευελιξία και ευκολία χρήσης, επιτρέποντας την προσαρμογή των εισαγωγικών δεδομένων, την επιλογή αλγορίθμων ομαδοποίησης και τη χρήση του μηχανισμού δρομολόγησης. Τα ευρήματα αποδεικνύουν σημαντικό δυναμικό για βελτίωση των υπηρεσιών παράδοσης τελευταίου μιλίου, τονίζοντας την κρίσιμη ανάγκη για την υιοθέτηση καινοτόμων τεχνικών βελτιστοποίησης χωρικών δεδομένων για την αντιμετώπιση των σύγχρονων προκλήσεων στις υπηρεσίες μεταφοράς και παράδοσης.

Λέξεις Κλειδιά — Χωρική Βελτιστοποίηση, Γεωχωρικά δεδομένα, Συστάδα, Συσταδοποίηση, KMeans, MiniBatch KMeans, KMeans Constrained, RESTful API, Open Source Routing Machine

Abstract

The escalating costs and inefficiencies in goods transportation, particularly within urban settings where last-mile delivery represents a significant portion of the total transportation expenses, pose a considerable challenge for the logistics sector. This diploma thesis introduces an in-depth investigation and development of a web-based spatial optimization service that seamlessly integrates with third-party web services, aiming to enhance the efficiency and speed of goods delivery in the Attica region. Through the exploration of clustering models for optimized distribution processes, the research utilizes driver routes from October 2023, evaluates existing clusters, and employs the Open Source Routing Machine (OSRM) for precise route and timing estimations, in conjunction with the geographical delineations of postal codes in Greece. The study applies KMeans and its variants, MiniBatch KMeans and KMeans Constrained, to achieve effective clustering. Furthermore, the thesis details the creation of a RESTful API, designed for flexibility and ease of use, allowing for the adjustment of input data, selection of clustering algorithms, and choice of routing machine. The findings demonstrate a significant potential for enhancing last-mile delivery services in urban logistics, underscoring the critical need for innovative spatial optimization strategies to overcome the current obstacles in transportation and delivery services.

Keywords — Spatial Optimization, Logistics, Clustering, KMeans, MiniBatch KMeans, KMeans Constrained, RESTful API, Open Source Routing Machine, OSRM, PostgreSQL, SQL, PostGIS, Geographic Information Systems (GIS), Python, Flask

Ευχαριστίες

Θα ήθελα να εκφράσω την ευγνωμοσύνη μου στον επιβλέποντα καθηγητή μου, κ. Βεσκούκη, για την αμέριστη υποστήριξη και καθοδήγησή του καθ' όλη τη διάρκεια της ολοκλήρωσης της διπλωματικής μου. Η τεχνογνωσία και η διορατικότητά του ήταν ανεκτίμητες, βοηθώντας με να ξεπεράσω τις προκλήσεις αυτού του έργου.

Επιπλέον, θα ήθελα να ευχαριστήσω θερμά την Skroutz Last Mile. Η γενναιοδωρία τους στην παροχή των απαραίτητων δεδομένων για την έρευνά μας αξίζει ιδιαίτερης εκτίμησης. Επίσης, εκτιμώ βαθύτατα την προθυμία τους να υποστηρίξουν την ακαδημαϊκή κοινότητα. Για περισσότερες πληροφορίες σχετικά με τα δεδομένα, μπορείτε να επικοινωνήσετε με τον κ. Αγαμέμνονα Παπάζογλου (agamemnon@skroutzlastmile.gr).

Επίσης, θα ήθελα να ευχαριστήσω θερμά τους φίλους μου για τη συνεχή ενθάρρυνση και την ακλόνητη πίστη τους στις ικανότητές μου. Η υποστήριξή τους ήταν πηγή δύναμης και κινήτρων για μένα καθ' όλη τη διάρκεια αυτού του ταξιδιού.

Τέλος, θέλω να εκφράσω την ευγνωμοσύνη μου προς την μου οικογένεια. Η αγάπη, η υπομονή και η αμέριστη πίστη τους στην πορεία μου αποτελούσαν μια σταθερή πηγή έμπνευσης και αποφασιστικότητας.

Νικόλας Τασιόπουλος
Μάρτιος 2023

Περιεχόμενα

Περίληψη	vii
Abstract	ix
Ευχαριστίες	xi
Περιεχόμενα	xiii
Λίστα Σχημάτων	xv
Κατάλογος Πινάκων	xvii
1 Εισαγωγή	1
1.1 Το πρόβλημα	1
1.2 Μεταφορές και Δρομολόγηση	2
1.2.1 Μεταφορές	2
1.2.2 Τελευταίο μίλι - Δρομολόγιο Οδηγού	2
1.3 Κίνητρο και Στόχοι	3
2 Υπόβαθρο και εργαλεία	5
2.1 Αλγόριθμοι Συσταδοποίησης	5
2.1.1 Συστάδα (Cluster)	5
2.1.2 K-Means	5
2.1.3 KMeans MiniBatch	9
2.1.4 KMeans Constrained	11
2.2 Σύστημα δρομολόγησης - OSRM (Open Source Routing Machine)	14
2.2.1 OSRM Services	14
2.2.2 Τρόπος λειτουργίας	24
2.2.3 Αναδρομή στο Vehicle Routing Problem - VRP	27
2.3 Γεωχωρικά Δεδομένα και Συναρτήσεις	29
2.3.1 The geometry_columns table - OGC simple feature	29
2.3.2 Σύστημα χωρικής αναφοράς (Spatial Reference System) - SRS	29
2.3.3 Γεωχωρικές Βάσεις δεδομένων	30
2.3.4 Γεωχωρικά δεδομένα	30
2.3.5 Γεωχωρικές Πράξεις	31
2.4 Tools	34
2.4.1 Postman: Εργαλείο Διαχείρισης API	34
2.4.2 Datagrip	35
3 Παρουσίαση του προβλήματος	39
3.1 Δεδομένα Εισόδου	39
3.1.1 Δρομολόγια - Σημεία Παράδοσης/Παραλαβής	39

3.1.2	Γεωγραφικά όρια ταχυδρομικών κωδικών	41
3.1.3	Υπάρχουσα λύση του προβλήματος - Συστάδες	41
3.2	Δείκτες Αξιολόγησης	44
3.2.1	Αξιολόγηση Δρομολογίων	44
3.2.2	Αξιολόγηση Συσταδοποίησης	45
3.3	Βάση δεδομένων και Views	47
3.3.1	Πίνακες	47
3.3.2	SQL Views	48
3.3.3	ER Diagram	49
4	Λύση του προβλήματος	53
4.1	Υλοποίηση	53
4.1.1	Δεδομένα εισόδου	53
4.1.2	Ελεγχος και προεπεξεργασία των δεδομένων εισόδου	53
4.1.3	Επιλογή και Εκπαίδευση μοντέλου	57
4.1.4	Παραγωγή των νέων clusters	57
4.2	Παραμετροποιήσεις	61
4.2.1	Μοντέλα Συσταδοποίησης	61
4.2.2	Μηχανή Δρομολόγησης	61
4.3	Διαγράμματα	61
5	Χρήση της αρχιτεκτονικής	65
5.1	Περιγραφή του API	65
5.2	Παραδείγματα Χρήσης	65
5.3	Αποτελέσματα του API	66
6	Αποτελέσματα και επέκταση	69
6.1	Αποτελέσματα	69
6.1.1	Συστάδες	69
6.1.2	Πλήθος των tasks ανά συστάδα	69
6.1.3	Διαφορές Συστάδων	74
6.2	Αξιολογηση συστάδων	74
6.3	Επέκταση	83
	Βιβλιογραφία	85

Κατάλογος Σχημάτων

1.2.1	Διάγραμμα κεντρικών κόμβων	2
1.2.2	Διάγραμμα τελευταίου μιλίου	3
2.1.1	Activity diagram του K-Means	7
2.1.2	Δεδομένα εισόδου	7
2.1.3	Παράδειγμα του αλγορίθμου k-means με $k = 4$	8
2.1.4	Οριακές γραμμές Voronoi	8
2.1.5	Mini Batch K-means Activity Diagram	10
2.1.6	Παράδειγμα Mini Batch K-means με $k = 4$ και μέγεθος παρτίδας $b = 4$	10
2.1.8	Παράδειγμα συσταδοποίησης με τον αλγόριθμο k-means constrained με $k = 3$ και μέγιστο πλήθος συστάδων 4.	13
2.2.1	Route Service Endpoint	15
2.2.2	Οπτικοποίηση του response του route service για την εύρεση της διαδρομής απο μια αφετηρία σε μια προορισμό.	17
2.2.3	Route service - Παράμετρος overview	18
2.2.4	Αποτέλεσμα του request για την εύρεση του πλησιέστερου δρόμου απο την ΣΗΜΜΥ ΕΜΠ	19
2.2.5	Table Service Endpoint	21
2.2.7	Αποτέλεσμα του request για την εύρεση της βέλτιστης διαδρομής για το πρόβλημα του περιοδεύοντος πωλητή	23
2.2.6	Trip Service Endpoint	24
2.2.8	Παραλλαγές του προβλήματος δρομολόγησης οχημάτων	28
2.3.1	Οι τρεις βασικοί τύποι γεωχωρικών δεδομένων : point, linestring, polygon.	30
2.3.2	Οι τρεις τύποι multigeometries.	31
2.3.3	Παράδειγμα εξωτερικού δακτυλίου (με πορτοκαλί χρώμα) Εικόνα από [34].	32
2.3.4	Παράδειγμα διαφορών st_exteriorRing και st_convexHull.	32
2.4.1	Γραφικό Περιβάλλον του Postman	34
2.4.2	Datagrip χάρτης γεωχωρικών δεδομένων	35
2.4.3	Datagrip GUI	36
2.4.4	Datagrip Queries & Visualizations	37
3.1.1	Δεδομένα στάσεων.	40
3.1.2	Παράδειγμα παραγόμενου δρομολογίου.	41
3.1.3	Παράδειγμα των γεωγραφικών ορίων των ταχυδρομικών κωδικών.	42
3.1.4	Παράδειγμα συστάδων.	43
3.1.5	Παράδειγμα των συστάδων σε Αττική.	44
3.3.1	ER Diagram της βάσης δεδομένων. Με γκρι χρώμα φαίνονται τα geometry columns.	51
4.1.1	Στάσεις με τοποθεσία POINT(0 0)	54
4.1.2	Παράδειγμα επικάλυψης ταχυδρομικών κωδικών	55
4.1.3	Activity diagram για την επίλυση της επικάλυψης των ταχυδρομικών κωδικών	56
4.1.4	Επίλυση της επικάλυψης των ταχυδρομικών κωδικών	57
4.1.5	Παράδειγμα αποτελεσμάτων συσταδοποίησης με τον αλγόριθμο KMeans	58

4.1.6	Παράδειγμα αποτελεσμάτων συσταδοποίησης με τον αλγόριθμο MiniBatchKMeans	58
4.1.7	Παράδειγμα αποτελεσμάτων συσταδοποίησης με τον αλγόριθμο KMeans Constrained . . .	59
4.1.8	Παράδειγμα αποτελεσμάτων συσταδοποίησης με τον αλγόριθμο KMeans	59
4.1.9	Παράδειγμα συσταδοποίησης των ταχυδρομικών κωδικών	60
4.1.10	Activity diagram για την παραγωγή των νέων συστάδων	62
4.3.1	Sequence Diagram της αρχιτεκτονικής	63
4.3.2	Component diagram της αρχιτεκτονικής	64
4.3.3	Deployment diagram της αρχιτεκτονικής	64
5.1.1	Το endpoint του API	66
5.3.1	Αποτέλεσμα του API για τα clusters	67
5.3.2	Αποτέλεσμα του API για τα piles	67
6.1.1	KMeans Clustering	70
6.1.2	KMeans Clustering without Warehouse Tasks	70
6.1.3	Minibatch KMeans Clustering	71
6.1.4	Minibatch KMeans Clustering without Warehouse Tasks	71
6.1.5	Constrained KMeans Clustering	72
6.1.6	Mean Constrained KMeans Clustering	72
6.1.7	Heatmaps	73
6.1.8	Differences between KMeans and Old Solution	75
6.1.9	Differences between Minibatch KMeans and Old Solution	76
6.1.10	Differences between Constrained KMeans and Old Solution	77
6.1.11	Differences between KMeans and Minibatch KMeans	78
6.1.12	Differences between Constrained KMeans and Constrained KMeans Mean	79
6.2.1	Τα τυχαία tasks που επιλέχθηκαν για την αξιολογηση	80
6.2.2	Μέση διάρκεια ανά task	81
6.2.3	Μέση διάρκεια ανά task με τυπική απόκλιση	81
6.2.4	Μέγιστη διάρκεια ανά task	82
6.2.5	Διάμεση διάρκεια ανά task	82
6.2.6	Κατανομή δεδομένων ανά task	83

Κατάλογος Πινάκων

2.1	Επεξήγηση των παραμέτρων για το request του OSRM route service.	16
2.2	Επεξήγηση των παραμέτρων για το request του OSRM nearest service.	17
2.4	Πίνακας διάρκειας οδήγησης μεταξύ περιχών της Αττικής (σε δευτερόλεπτα)	22
2.5	Επεξήγηση παραμέτρων για το request του OSRM trip service.	23
3.1	Πεδία του πίνακα δρομολογίων.	40
3.2	Πεδία του shapefile των ταχυδρομικών κωδικών.	41
3.3	Πεδία του shapefile της υπάρχουσας λύσης.	43
3.4	Πεδία του πίνακα clustering_algorithms.	47
3.5	Πεδία του πίνακα clustering_results.	48
3.6	Πεδία του πίνακα new_piles.	48
4.1	Παράδειγμα επικάλυψης ταχυδρομικών κωδικών	54
5.1	Περιγραφή των URL παραμέτρων του API	65
5.2	Περιγραφή των BODY παραμέτρων του API	65
6.2	Μετρικές απόδοσης διαφορετικών αλγορίθμων	74

Κεφάλαιο 1

Εισαγωγή

Τα τελευταία χρόνια έχει γίνει όλο και πιο κοστοβόρα η μεταφορά των προϊόντων. Η μεταφορά των αγαθών, αποτελείται από τρία βασικά στάδια. Την παραλαβή των αγαθών από τον προμηθευτή, την μεταφορά των αγαθών από την γειτονία του προμηθευτή στην γειτονία του πελάτη και την τελική παράδοση των αγαθών στον πελάτη. Οι αστικές περιοχές επιβαρύνουν το κόστος των μεταφορών περισσότερο. Συγκεκριμένα, η δρομολόγηση αγαθών σε αστικές περιοχές μπορεί να φτάνει μέχρι και το 50% του συνολικού κόστους μεταφοράς [1]. Επιπλέον, αναμένεται μέχρι το 2030 να υπάρξει αύξηση των αστικών περιοχών με αποτέλεσμα το μεταφορικό κόστος των αστικών περιοχών σε σύγκριση με το συνολικό κόστος μεταφοράς να αυξηθεί μέχρι και το 68%. Ανεβάζοντας την συνολική τιμή των προϊόντων και μειώνοντας την προσβασιμότητα τους. Επομένως, η ανάγκη για αποτελεσματικότερες μεταφορές είναι πιο επιτακτική από ποτέ.

Στόχος αυτής της διπλωματικής είναι η υλοποίηση και η σχεδίαση ενός API λογισμικού για την αποθήκευση, σύγκριση και αξιολόγηση μοντέλων συσταδοποίησης στην Αττική για ευέλικτη, γρηγορότερη και αποτελεσματικότερη παράδοση αγαθών. Ωστόσο, μπορέσαμε να διαχωρίσουμε το πρόβλημα σε επιμέρους, γεγονός που επιτρέπει να λειτουργούν τα μέρη αυτά αυτόνομα.

Χρησιμοποιήθηκαν, δρομολόγια οδηγών του Οκτώβριο του 2023 στην Αττική, τα ήδη υπάρχοντα clusters, το εργαλείο OSRM για την κατασκευή των διαδρομών αλλά και των χρόνων. Τέλος, χρειάστηκαν τα γεωχωρικά όρια των ταχυδρομικών κωδικών της Ελλάδας.

Οι αλγόριθμοι συσταδοποίησης που χρησιμοποιήθηκαν είναι ο KMeans και 2 παραλλαγές του Minibatch Kmeans και KMeans Constrained. Για την μηχανή δρομολόγησης χρησιμοποιήθηκε το Open Source Routing Machine (OSRM).

Η υλοποίηση του λογισμικού έγινε σε Python και η αποθήκευση των δεδομένων σε μια βάση δεδομένων PostgreSQL.

1.1 Το πρόβλημα

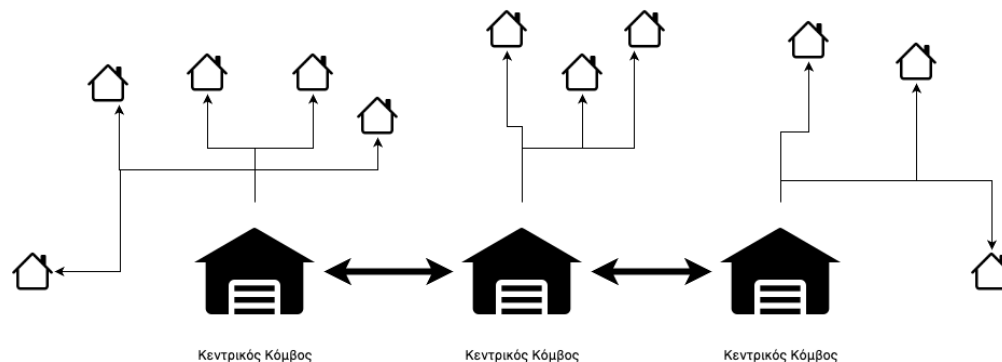
Το κόστος των μεταφορών και του τελευταίου μιλίου στις αστικές περιοχές αποτελεί σημαντική πρόκληση για τη διαχείριση της αλυσίδας εφοδιασμού και τον προγραμματισμό των μεταφορών. Το τελευταίο μίλι αναφέρεται στη μετακίνηση αγαθών από έναν κόμβο μεταφορών στον τελικό προορισμό. Είναι το λιγότερο αποδοτικό στάδιο της αλυσίδας εφοδιασμού και περιλαμβάνει έως και το 28% του συνολικού κόστους παράδοσης. Η αύξηση του οριακού κόστους μεταφοράς των ανθρώπων από έναν κόμβο μεταφοράς στον τελικό προορισμό τους αποτελεί σημαντική πρόκληση. Στα δίκτυα εμπορευματικών μεταφορών, τα δέματα μπορούν να παραδοθούν μαζί και αποτελεσματικά σε έναν κεντρικό κόμβο μέσω πλοίου, τρένου ή άλλων μέσων, αλλά στη συνέχεια πρέπει να φορτωθούν σε μικρότερα οχήματα για παράδοση σε μεμονωμένους πελάτες. Το πρόβλημα του τελευταίου μιλίου μπορεί επίσης να περιλαμβάνει την πρόκληση της πραγματοποίησης παραδόσεων σε αστικές περιοχές, οι οποίες συχνά συμβάλλουν στη συμφόρηση και

σε προβλήματα ασφάλειας. Έχουν προταθεί διάφορες καινοτόμες στρατηγικές και τεχνολογίες για την αντιμετώπιση αυτού του προβλήματος, όπως η χρήση μικρότερων οχημάτων, κινητών αποθηκών (moving depot) και δομών παράδοσης δύο κλιμακίων. Οι προσεγγίσεις αυτές αποσκοπούν στη μείωση του κόστους των εξωτερικών παραγόντων, όπως η συμφόρηση και η ρύπανση, και στη βελτίωση της αποτελεσματικότητας του τελευταίου σταδίου παράδοσης στις αστικές περιοχές. Η αποδοτικότητα κόστους και η εφαρμοσιμότητα αυτών των εναλλακτικών δομών διανομής επηρεάζονται από τις διαδικασίες εφοδιασμού, τα χαρακτηριστικά του παραλήπτη και τα τοπικά χαρακτηριστικά της πόλης [1]–[5].

1.2 Μεταφορές και Δρομολόγηση

1.2.1 Μεταφορές

Για τη βελτίωση της αποδοτικότητας στη διαχείριση των μεταφορικών διεργασιών, έχει υιοθετηθεί η τακτική της συσταδοποίησης των περιοχών εξυπηρέτησης και η συνακόλουθη ανάθεσή τους σε κεντρικούς μεταφορικούς κόμβους. Αυτοί οι κεντρικοί κόμβοι φέρουν την ευθύνη για την τοπική διαχείριση της περιοχής τους και για τη μεταφορά αγαθών προς άλλους κεντρικούς κόμβους, ενισχύοντας έτσι την ομαλότητα και την αποτελεσματικότητα του δικτύου διανομής.



Σχήμα 1.2.1: Διάγραμμα κεντρικών κόμβων

Η ηπειρωτική Ελλάδα, στο πλαίσιο αυτής της στρατηγικής, έχει διαιρεθεί σε τέσσερις βασικές περιοχές, με κάθε περιοχή να υποστηρίζεται από τον δικό της κεντρικό κόμβο. Αυτή η διαίρεση επιτρέπει την πιο οργανωμένη και εξειδικευμένη διαχείριση των μεταφορών, καθώς κάθε κόμβος εξυπηρετεί συγκεκριμένη γεωγραφική περιοχή.

Ειδικότερα, στην περιοχή της Αττικής, η οποία αντιμετωπίζει έναν τεράστιο όγκο μεταφορών, έχει καθοριστεί μόνο ένας κεντρικός κόμβος. Ωστόσο, η πολυπλοκότητα και η υψηλή ζήτηση σε αυτή την περιοχή απαιτούν περαιτέρω διαμέριση της Αττικής σε μικρότερες υπο-περιοχές για αποδοτικότερη διαχείριση. Πιθανές διαμερίσεις μπορεί να περιλαμβάνουν τις περιφερειακές ενότητες της Αττικής, τους δήμους, ή τους ταχυδρομικούς κωδικούς, αλλά οι λύσεις αυτές συναντούν προκλήσεις λόγω του μεγέθους τους, το οποίο είτε είναι υπερβολικά μικρό ή μεγάλο για την ιδανική διαχείριση μεταφορών. Συνεπώς, απαιτείται η ανάπτυξη μιας νέας διαίρεσης της Αττικής, η οποία θα προσφέρει την ιδανική ισορροπία για την αποτελεσματική διαχείριση μεταφορών.

Αυτή η προσέγγιση θα διευκολύνει όχι μόνο τον διαμερισμό των δεμάτων στους οδηγούς, αλλά και την εσωτερική οργάνωση των αποθηκών, επιτρέποντας μια πιο αποδοτική και οργανωμένη διαχείριση των πόρων και των δραστηριοτήτων της μεταφοράς.

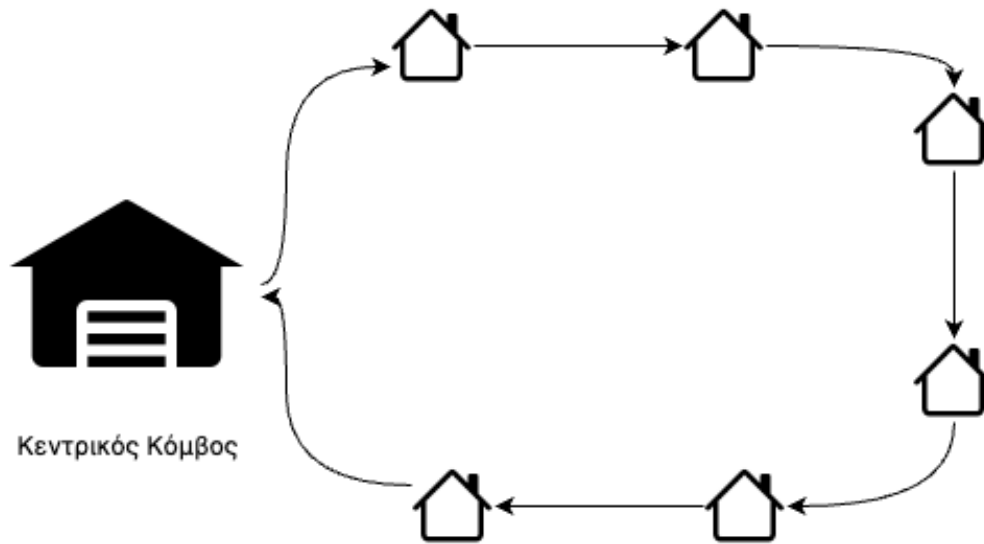
1.2.2 Τελευταίο μίλι - Δρομολόγιο Οδηγού

Στην υποενότητα αυτή, εστιάζουμε στην διαδικασία του "τελευταίου μιλίου" και την δρομολόγηση του οδηγού, ένα κρίσιμο στάδιο στην αλυσίδα παράδοσης που εξασφαλίζει την έγκαιρη και αποτελεσματική

παράδοση δεμάτων στους προορισμούς τους. Αυτή η διαδικασία αρχίζει με την άφιξη του οδηγού σε έναν κεντρικό κόμβο στην αρχή της βάρδιας του, όπου το όχημά του φορτώνεται με δέματα προς παράδοση.

Ο οδηγός, στη συνέχεια, αναλαμβάνει την ευθύνη για μια συγκεκριμένη γειτονιά ή περιοχή, όπου πρέπει να παραδώσει ή να παραλάβει δέματα. Αυτό επιτρέπει την αποδοτική διαχείριση των παραδόσεων, καθώς κάθε οδηγός εστιάζει σε μια περιορισμένη περιοχή, μειώνοντας τον χρόνο μετακίνησης μεταξύ των διευθύνσεων και βελτιστοποιώντας την ολοκλήρωση της παράδοσης των προϊόντων. Η διαδρομή του οδηγού καταλήγει πάντα στον κεντρικό κόμβο, ο οποίος λειτουργεί ως αφετηρία και τερματικός σταθμός της καθημερινής διαδρομής, επιτρέποντας την επαναφόρτωση, την επιστροφή δεμάτων που δεν παραδόθηκαν, και την προετοιμασία για την επόμενη βάρδια.

Η παρακάτω εικόνα παρουσιάζει ένα διαγραμματικό παράδειγμα της διαδικασίας τελευταίου μιλίου, απεικονίζοντας την δρομολόγηση του οδηγού από τον κεντρικό κόμβο προς τις διάφορες τοποθεσίες παράδοσης και πίσω, έτσι ώστε να παρέχει μια οπτική αναπαράσταση των βασικών στοιχείων και των βημάτων που εμπλέκονται στην διαδικασία τελευταίου μιλίου.



Σχήμα 1.2.2: Διάγραμμα τελευταίου μιλίου

1.3 Κίνητρο και Στόχοι

Στο πλαίσιο της παρούσας διπλωματικής εργασίας, υλοποιήσαμε μια αρχιτεκτονική λογισμικού, η οποία επιδιώκει την αποδοτική επίλυση του προβλήματος που περιγράφεται προηγουμένως. Αυτή η αρχιτεκτονική προσφέρει τη δυνατότητα εύκολης ένταξης και χρήσης διαφόρων υπολογιστικών μονάδων (modules), επιτρέποντας την απρόσκοπτη σύγκριση διαφόρων λύσεων που εκτελούνται με διαφορετικά σενάρια αλγορίθμων ή ρυθμίσεων (παραμέτρων) εκτέλεσης αλγορίθμων.

Επιπλέον, στο λογισμικό δίνονται δεδομένα που αφορούν μια περιοχή. Παρακολουθώντας το παράδειγμα της Αττικής, όπου διαθέτουμε δεδομένα στην επιθυμητή μορφή, το λογισμικό μας ενσωματώνει τα δεδομένα αυτά, χωρίς ωστόσο να περιορίζεται αποκλειστικά σε αυτά. Το εν λόγω λογισμικό διαρθρώνεται σε τέσσερα κύρια μέρη, τα οποία είναι σχεδιασμένα να λειτουργούν εντελώς αυτόνομα:

- Το πρώτο μέρος αφορά την αποθήκευση των δεδομένων σε μια βάση δεδομένων.
- Το δεύτερο μέρος αφορά την εκπαίδευση των μοντέλων συσταδοποίησης.
- Το τρίτο μέρος αφορά την δρομολόγηση των οδηγών, μια μηχανή δρομολόγησης.
- Τέλος, ένας συντονιστής (orchestrator) είναι υπεύθυνος για την επικοινωνία με τα υπόλοιπα συστήματα και την εκτέλεση των αλγορίθμων συσταδοποίησης.

Η ανάπτυξη του λογισμικού έγινε χρησιμοποιώντας τη γλώσσα προγραμματισμού Python, ενώ για την αποθήκευση των δεδομένων χρησιμοποιήθηκε η βάση δεδομένων PostgreSQL. Οι αλγόριθμοι KMeans, Minibatch Kmeans και KMeans Constrained εφαρμόστηκαν για την εκπαίδευση των μοντέλων συσταδοποίησης. Για τη δρομολόγηση των οδών αξιοποιήθηκε το Open Source Routing Machine (OSRM).

Η πολυπλοκότητα του προβλήματος, καθώς και οι ποικίλες προκλήσεις που παρουσιάστηκαν κατά τη διάρκεια της υλοποίησης, μας οδήγησαν στην ανάπτυξη μιας λύσης που χαρακτηρίζεται τόσο από την αποδοτικότητα όσο και από την ευελιξία της. Συγκεκριμένα, το λογισμικό προσφέρει τη δυνατότητα στους χρήστες να ενσωματώσουν νέους αλγορίθμους συσταδοποίησης, νέα συστήματα δρομολόγησης, διαφορετικούς τρόπους διαχείρισης των αποτελεσμάτων είτε API είτε python κώδικας. Επιπροσθέτως, επιτρέπει την προσαρμογή των αποτελεσμάτων του, από τα δεδομένα εισόδου, όπως τα δρομολόγια και οι γεωγραφικές περιοχές (π.χ., ταχυδρομικοί κωδικοί), προκειμένου να παραχθούν συγκρίσιμα αποτελέσματα.

Κεφάλαιο 2

Υπόβαθρο και εργαλεία

Σε αυτό το κεφάλαιο παρουσιάζονται αναλυτικά τα εργαλεία που χρησιμοποιήθηκαν για την υλοποίηση της παρούσας διπλωματικής εργασίας. Αρχικά, παρουσιάζονται οι αλγόριθμοι συσταδοποίησης που χρησιμοποιήθηκαν για την επίλυση του προβλήματος της συσταδοποίησης των περιοχών της Αθήνας. Στη συνέχεια, παρουσιάζεται το εργαλείο Open Source Routing Machine (OSRM) που χρησιμοποιήθηκε για την εξαγωγή διαφόρων δεδομένων, χρόνο και απόσταση, των διαδρομών. Τέλος, παρουσιάζονται οι γεωχωρικές λειτουργίες που χρησιμοποιήθηκαν.

2.1 Αλγόριθμοι Συσταδοποίησης

Πριν προχωρήσουμε στην παρουσίαση των αλγορίθμων συσταδοποίησης, κρίνεται απαραίτητη η διατύπωση του ορισμού της συστάδας.

2.1.1 Συστάδα (Cluster)

Συστάδα, ή αλλιώς Cluster, ονομάζεται μια ομάδα αντικειμένων που παρουσιάζουν συγγένεια και ομοιότητα μεταξύ τους. Στον τομέα της μηχανικής μάθησης, η συσταδοποίηση (Clustering) ορίζεται ως η μέθοδος κατά την οποία τα αντικείμενα κατηγοριοποιούνται σε ένα σύνολο ομάδων (συστάδων). Η ομαδοποίηση αυτή πραγματοποιείται με τρόπο ώστε τα αντικείμενα που ανήκουν στην ίδια συστάδα να παρουσιάζουν ομοιότητες και κοινά χαρακτηριστικά, σε αντίθεση με αντικείμενα άλλων συστάδων. Ο αριθμός και η σύνθεση των συστάδων δεν είναι προκαθορισμένα, γεγονός που κατατάσσει την τεχνική της συσταδοποίησης στην κατηγορία της μη επιβλεπόμενης μάθησης.[6]

2.1.2 K-Means

Ο αλγόριθμος του Lloyd [7] αποτελεί μια θεμελιώδη προσέγγιση στην ομαδοποίηση δεδομένων. Ο αλγόριθμος δέχεται σαν είσοδο σημεία και τον αριθμό των συστάδων που πρέπει να παράξει. Ο στόχος είναι να ελαχιστοποιηθεί το άθροισμα των τετραγώνων των αποστάσεων μεταξύ κάθε σημείου δεδομένων και του κεντροειδούς της συστάδας στην οποία έχει ανατεθεί.

Ο KMeans αποτελεί ένα κρίσιμο εργαλείο στην επίλυση προβλημάτων συσταδοποίησης στον τομέα της επεξεργασίας δεδομένων και της μηχανικής μάθησης.

Η συσταδοποίηση K-means αποτελεί μία δημοφιλή και απλή μέθοδο για την τμηματοποίηση ενός συνόλου δεδομένων σε k ομάδες, όπου k είναι ένας προκαθορισμένος αριθμός συστάδων.

Ο αλγόριθμος λειτουργεί ως εξής:

Algorithm 1 K-Means Clustering**Input:**

Dataset $X = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ where $\mathbf{x}_i \in \mathbb{R}^d$
 Number of clusters, k

Output:

A partitioning of the data into k clusters

```

1: procedure KMEANS( $X, k$ )
2:   Initialization: Randomly select  $k$  initial cluster centroids  $\mu_1, \mu_2, \dots, \mu_k$ 
3:   repeat
4:     for each data point  $\mathbf{x}_i \in X$  do
5:       Assign  $\mathbf{x}_i$  to the cluster  $C_j$  with the closest centroid:
6:        $j = \arg \min_l \|\mathbf{x}_i - \mu_l\|^2$ 
7:     end for
8:     for each cluster  $C_j$  do
9:       Update the centroid:
10:       $\mu_j \leftarrow \frac{1}{|C_j|} \sum_{\mathbf{x}_i \in C_j} \mathbf{x}_i$ 
11:    end for
12:  until stopping criteria is met
13: end procedure

```

Θα προχωρήσουμε στην περιγραφή του αλγορίθμου με εκτενέστερη ανάλυση και λεπτομέρειες. Αρχικά, πρέπει να προσδιορίσουμε τον βέλτιστο αριθμό συστάδων (k). Δυστυχώς, δεν υπάρχει μια απάντηση σε αυτό το ερώτημα. Ο βέλτιστος αριθμός των συστάδων είναι υποκειμενικός και εξαρτάται από τη μέθοδο που χρησιμοποιείται για τη μέτρηση των ομοιοτήτων και τις παραμέτρους που χρησιμοποιούνται για τον διαχωρισμό. Εδώ, υποθέτουμε ότι ο αριθμός των συστάδων (k) είναι γνωστός. Το επόμενο βήμα είναι η αρχικοποίηση k κέντρων συστάδων. Μια κοινή μέθοδος είναι η τυχαία αρχικοποίηση, επιλέγοντας τυχαία k σημεία και θεωρώντας αυτά ως αρχικά κέντρα. Για να αποφευχθεί η εύρεση τοπικών βέλτιστων κατά την εφαρμογή του k -means, ορισμένες υλοποιήσεις λαμβάνουν υπόψη πολλαπλές τυχαίες αρχικοποιήσεις.

Ο αλγόριθμος k -means περιλαμβάνει έναν επαναληπτικό βρόχο (repeat) που εκτελείται έως ότου ικανοποιηθεί μια συνθήκη τερματισμού. Αυτή η συνθήκη ικανοποιείται είτε μετά από έναν προκαθορισμένο αριθμό επαναλήψεων, είτε αν το συνολικό άθροισμα των αποστάσεων των σημείων με τα κέντρα των συστάδων γίνει μικρότερο από ένα προκαθορισμένο όριο. Μέσα στον βρόχο βρίσκονται δύο βρόχοι 'for'. Ο πρώτος αναθέτει σε κάθε σημείο x_i την ομάδα με το κέντρο που έχει τη μικρότερη απόσταση από το x_i . Μετά την εκχώρηση των σημείων σε συστάδες, ο δεύτερος βρόχος 'for' υπολογίζει τα νέα κέντρα των συστάδων. Τα κέντρα βρίσκονται υπολογίζοντας τον μέσο όρο όλων των σημείων κάθε ομάδας. [8]

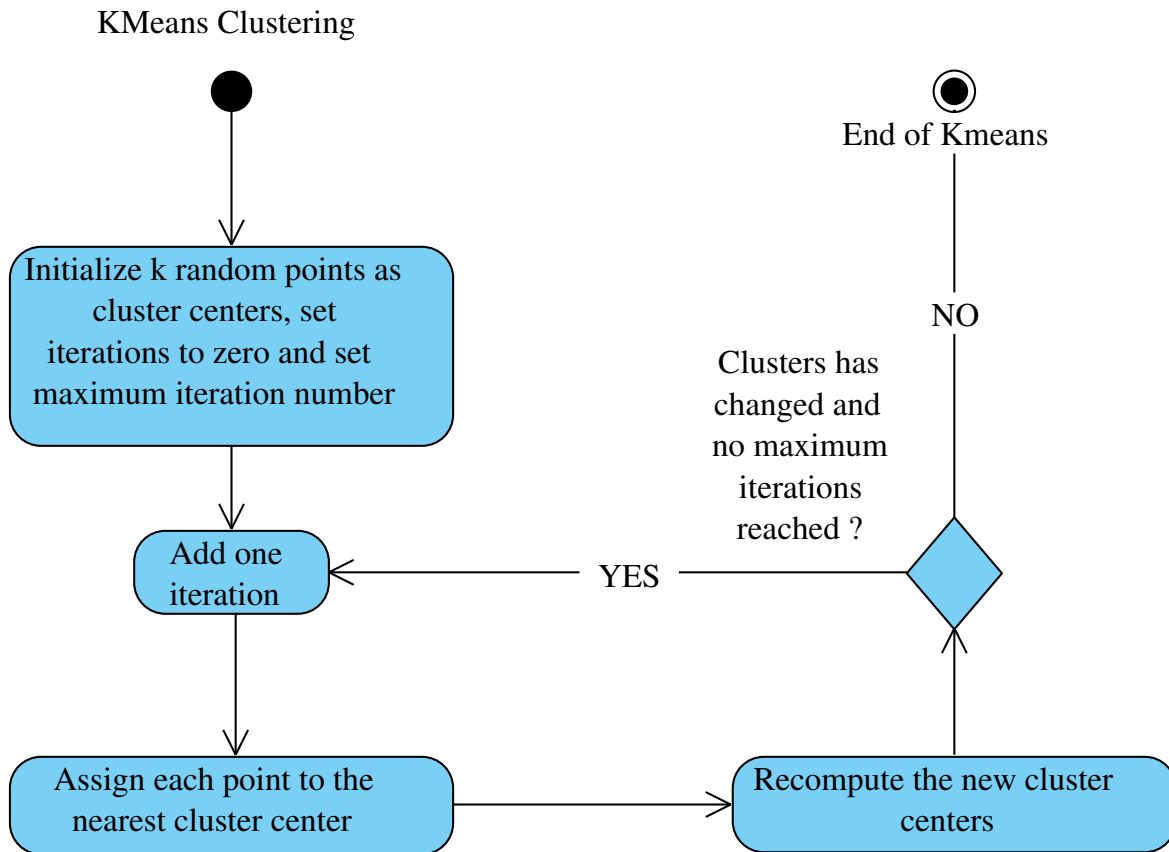
Όταν ικανοποιηθεί η συνθήκη διακοπής, ο αλγόριθμος επιστρέφει τις συστάδες με τα σημεία και τα κέντρα τους. Στο σχήμα 2.1.2 παρουσιάζεται ένα παράδειγμα του αλγορίθμου k -means. Τα δεδομένα εισόδου περιέχουν τέσσερα σύνολα δεδομένων που δημιουργήθηκαν τυχαία. Ο αλγόριθμος k -means εφαρμόζεται στα δεδομένα με $k = 4$ και τα αποτελέσματα της συσταδοποίησης παρουσιάζονται στο Σχήμα 2.1.3.

Μαθηματικό Υπόβαθρο και Συνάρτηση Στόχου

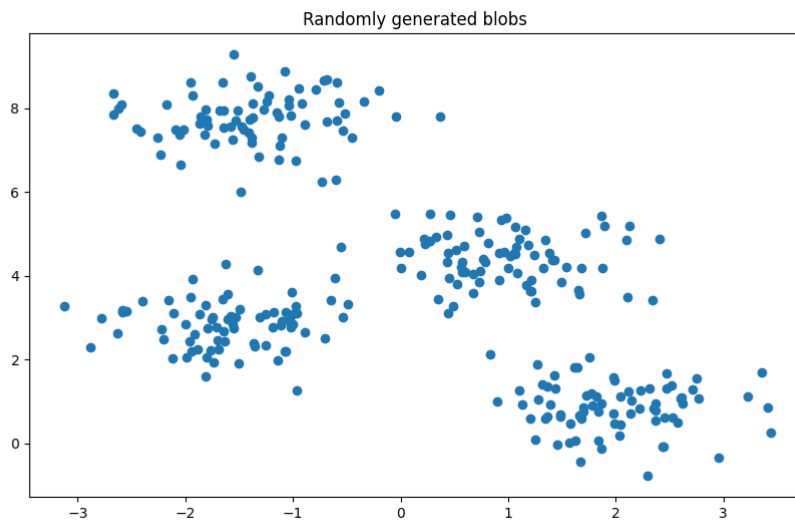
Η συνάρτηση στόχου του αλγορίθμου KMeans επιδιώκει την ελαχιστοποίηση του συνολικού αθροίσματος των τετραγωνικών αποστάσεων μεταξύ κάθε σημείου στο σύνολο δεδομένων και του κεντροειδούς της συστάδας S_i στην οποία ανήκει. Η συνάρτηση στόχου J που επιδιώκεται να ελαχιστοποιηθεί είναι η εξής:

$$J(C, \{S_i\}_{i=1}^k) = \sum_{i=1}^k \sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \mathbf{c}_i\|^2$$

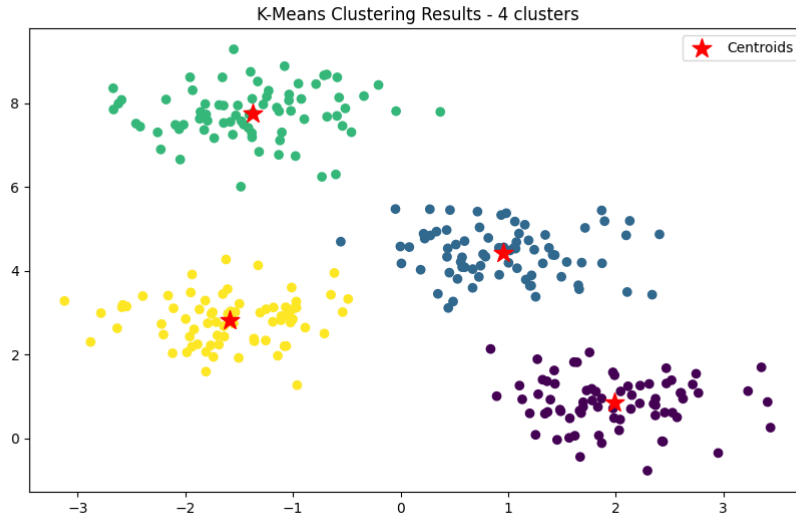
όπου:



Σχήμα 2.1.1: Activity diagram του K-Means

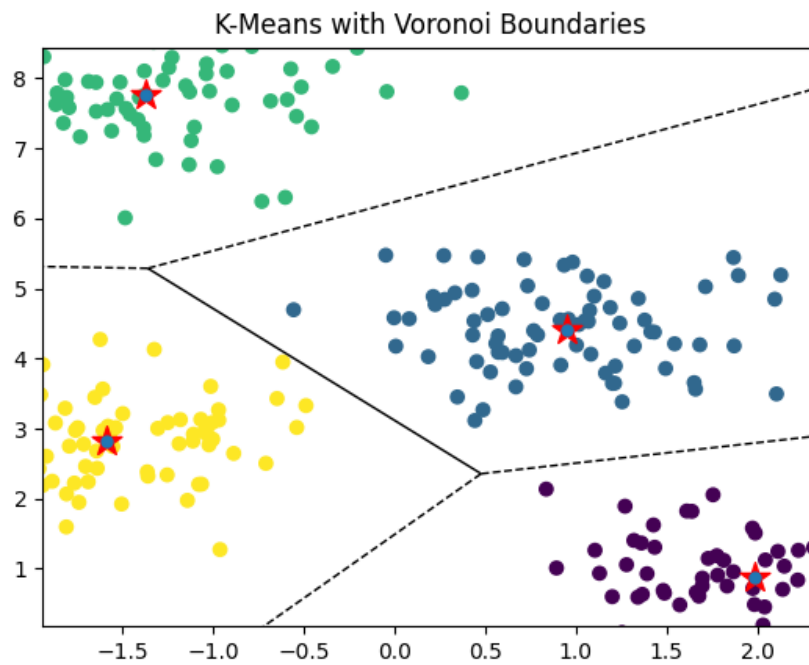


Σχήμα 2.1.2: Δεδομένα εισόδου



Σχήμα 2.1.3: Παράδειγμα του αλγορίθμου k-means με $k = 4$.

- C αντιπροσωπεύει το σύνολο των κέντρων των συστάδων,
- c_i είναι το κέντρο της συστάδας i ,
- S_i είναι το σύνολο των σημείων που ανήκουν στη συστάδα i ,
- και x είναι τα σημεία δεδομένων.



Σχήμα 2.1.4: Οριακές γραμμές Voronoi

2.1.3 KMeans MiniBatch

Ο αλγόριθμος Mini Batch K-means επιτυγχάνει υπολογιστική αποτελεσματικότητα επεξεργάζοντας μικρές παρτίδες του συνόλου δεδομένων σε κάθε επανάληψη. Αυτή η προσέγγιση, ενώ μειώνει τον χρόνο που απαιτείται για τη σύγκλιση, ενδέχεται να επηρεάσει την ποιότητα των συμπλεγμάτων που προκύπτουν λόγω της στοχαστικής φύσης της επιλογής παρτίδων [9] [10].

Σημαντική, λοιπόν είναι η επίδραση του μεγέθους της παρτίδας (batch size) στην ποιότητα του clustering και στην υπολογιστική αποτελεσματικότητα, υποδεικνύοντας ότι τα βέλτιστα μεγέθη παρτίδας αποτελούν σημείο ισορροπίας ανάμεσα στην ποιότητα και το υπολογιστικό κόστος.

Algorithm 2 MiniBatch KMeans Clustering

```

1: Input: Dataset  $X$ , Number of clusters  $k$ , Batch size  $b$ , Number of iterations  $T$ 
2: Output: Cluster centroids  $C$ 
3: procedure MINIBATCHKMEANS( $X, k, b, T$ )
4:   Initialize cluster centroids  $C$  randomly from  $X$ 
5:   for  $t = 1$  to  $T$  do
6:     Draw a random batch  $B$  from  $X$  with size  $b$ 
7:     for each sample  $x$  in  $B$  do
8:       Find the nearest centroid  $c$  in  $C$  to  $x$ 
9:       Assign  $x$  to cluster represented by  $c$ 
10:    end for
11:    for each cluster  $j$  do
12:      Update the centroid of  $j$  by computing the mean of all samples assigned to  $j$ , considering
the current batch
13:    end for
14:  end for
15: end procedure

```

Ο Mini Batch K-means ξεκινά με την τυχαία επιλογή αρχικών κέντρων για τις συστάδες και στη συνέχεια επεξεργάζεται μικρές ομάδες (παρτίδες) δεδομένων αντί ολόκληρου του συνόλου δεδομένων σε κάθε επανάληψη.

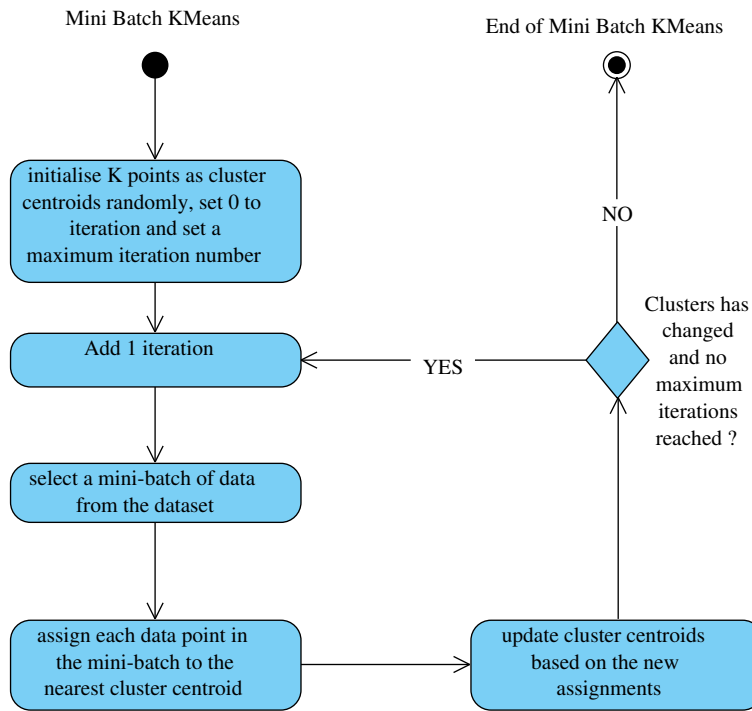
Στο επίκεντρο της διαδικασίας είναι η δυναμική ενημέρωση των κέντρων των συστάδων με βάση τα δεδομένα κάθε παρτίδας, επιτρέποντας στον αλγόριθμο να προσαρμόζεται και να βελτιώνει την ομαδοποίηση δυναμικά. Η στοχαστική φύση της επιλογής των παρτίδων εισάγει μια στοιχειώδη αβεβαιότητα στη διαδικασία, προκαλώντας μια ελαφριά μείωση στην ποιότητα των τελικών συστάδων σε σύγκριση με τον παραδοσιακό K-means.

Ενδιαφέρον παρουσιάζει η επιλογή του μεγέθους της παρτίδας, καθώς αυτό αποτελεί κρίσιμο παράγοντα για την ισορροπία μεταξύ υπολογιστικής αποδοτικότητας και ποιότητας ομαδοποίησης. Μικρότερες παρτίδες οδηγούν σε ταχύτερες επαναλήψεις αλλά μπορεί να απαιτούν περισσότερες επαναλήψεις για την επίτευξη σύγκλισης, ενώ μεγαλύτερες παρτίδες προσεγγίζουν τη συμπεριφορά του παραδοσιακού K-means, αυξάνοντας τον υπολογιστικό φόρτο αλλά βελτιώνοντας τη σταθερότητα των αποτελεσμάτων.

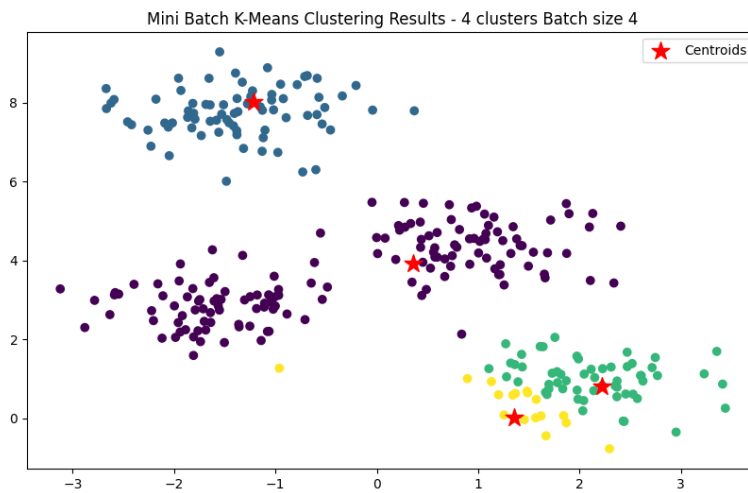
Παρά τις προκλήσεις, ο Mini Batch K-means παραμένει μια δυνατή επιλογή για την ομαδοποίηση μεγάλων συνόλων δεδομένων, προσφέροντας μια ελκυστική ισορροπία μεταξύ αποδοτικότητας και ποιότητας. Με την κατάλληλη προσαρμογή των παραμέτρων και την προσεκτική επιλογή του μεγέθους παρτίδας, ο αλγόριθμος μπορεί να παράγει σταθερά και αξιόπιστα αποτελέσματα, κάνοντάς τον ιδανικό για εφαρμογές στον τομέα της επιστήμης δεδομένων και της μηχανικής μάθησης.

Ενδεικτικά παραθέτουμε αποτελέσματα από την εφαρμογή του αλγορίθμου Mini Batch K-means στο σύνολο δεδομένων 2.1.2:

Η συνάρτηση μεταφοράς του αλγορίθμου Mini Batch K-means είναι ίδια με την συνάρτηση μεταφοράς του αλγορίθμου K-means, όπως περιγράφεται στην ενότητα 2.1.2.



Σχήμα 2.1.5: Mini Batch K-means Activity Diagram



Σχήμα 2.1.6: Παράδειγμα Mini Batch K-means με $k = 4$ και μέγεθος παρτίδας $b = 4$.

2.1.4 KMeans Constrained

Παρόλο που ο αλγόριθμος K-Means είναι ιδιαίτερα αποτελεσματικός, παρουσιάζει κάποιους περιορισμούς όταν αντιμετωπίζει περιπτώσεις με ειδικές απαιτήσεις στην διαδικασία της συσταδοποίησης.

Για παράδειγμα σε περιπτώσεις όπου η κατανομή των σημείων δεν είναι ομοιόμορφη οι τελικές συστάδες του KMeans θα έχουν μεγάλες ανισότητες στο μέγεθος τους [2.1.7b](#).

Περιορισμοί του K-Means

Σε ορισμένες περιπτώσεις, μπορεί να είναι επιθυμητό να επιβληθούν ορισμένοι περιορισμοί στη διαδικασία ομαδοποίησης. Τέτοιοι περιορισμοί μπορεί να περιλαμβάνουν τον καθορισμό ενός ελάχιστου ή/και μέγιστου μεγέθους για κάθε ομάδα (cluster) ή την ενσωμάτωση κάποιας προηγούμενης γνώσης σχετικά με τα δεδομένα. Η παραλλαγή constrained k-means [\[11\]](#) μπορεί να ικανοποιήσει τέτοιους περιορισμούς διατυπώνοντας το βήμα εκχώρησης σε ομάδες ως πρόβλημα ελαχίστου κόστους ροής (minimum cost flow - MCF). Τα προβλήματα MCF επιλύονται αποτελεσματικά με τη χρήση γραμμικών τεχνικών βελτιστοποίησης δικτύων.

Ο αλγόριθμος

Ο KMeans Constrained αλγόριθμος [\[11\]](#) τροποποιεί τον βασικό του K-Means [2.1.2](#) ενσωματώνοντας περιορισμούς στα βήματα εκχώρησης και ενημέρωσης. Ο αλγόριθμος συνοψίζεται ως εξής:

Οι περιορισμοί μπορεί να είναι αυστηροί (hard) ή ήπιοι (soft). Οι αυστηροί περιορισμοί πρέπει να τηρούνται πιστά αλλιώς ο αλγόριθμος αποτυγχάνει, ενώ οι ήπιοι περιορισμοί επηρεάζουν τον αλγόριθμο χωρίς αυστηρή επιβολή.

Algorithm 3 Constrained KMeans with Min and Max Cluster Sizes

```

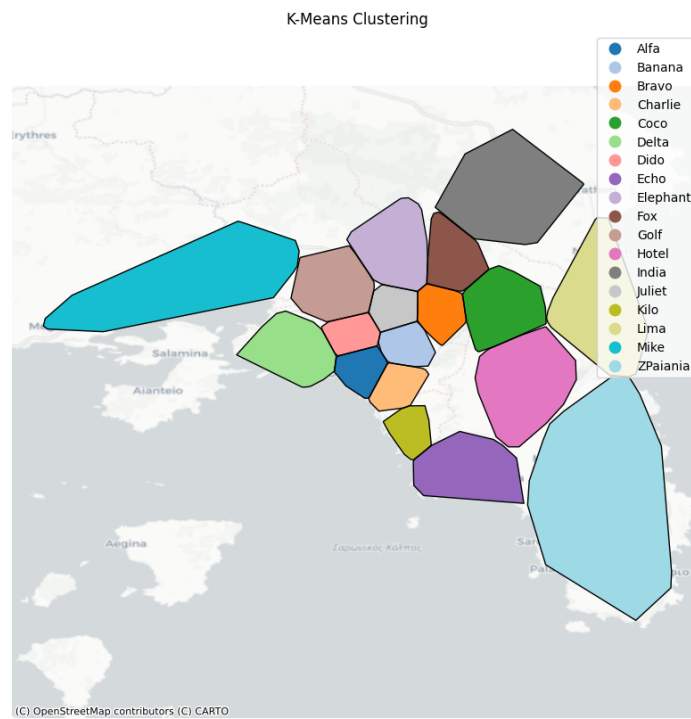
1: Input: Dataset  $X$ , Number of clusters  $k$ , Min size  $m_{\min}$ , Max size  $m_{\max}$ 
2: Output: Cluster centroids  $C$ , Cluster assignments for each  $x \in X$ 
3: procedure CONSTRAINEDKMEANS( $X, k, m_{\min}, m_{\max}$ )
4:   Initialize cluster centroids  $C$  randomly from  $X$ 
5:   Initialize cluster sizes  $S$  with zero for all  $k$  clusters
6:   repeat
7:     Assign each  $x \in X$  to the nearest centroid  $c \in C$ , subject to  $m_{\min} \leq S[c] \leq m_{\max}$ 
8:     for each centroid  $c$  in  $C$  do
9:       Update  $c$  by calculating the mean of all  $x \in X$  assigned to  $c$ , respecting  $m_{\min}$  and  $m_{\max}$ 
10:      Update  $S[c]$  with the number of  $x \in X$  assigned to  $c$ 
11:     end for
12:     Check for convergence or max iterations
13:   until stopping criteria is met
14:   return  $C$ , Cluster assignments
15: end procedure

```

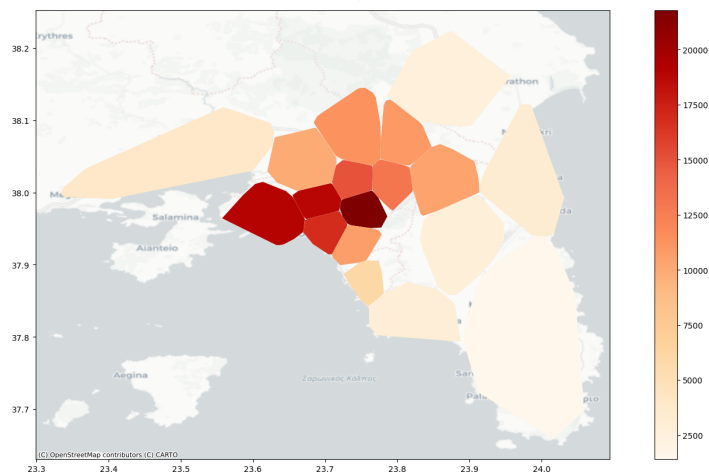
Παράδειγμα

Το Σχήμα [2.1.8](#) παρουσιάζει ένα παράδειγμα ομαδοποίησης με τον αλγόριθμο k-means constrained, όπου $k=3$ και μέγιστο μέγεθος συστάδων 75. Τα σημεία απεικονίζονται ως κύκλοι, τα κεντροειδή των ως αστέρια.

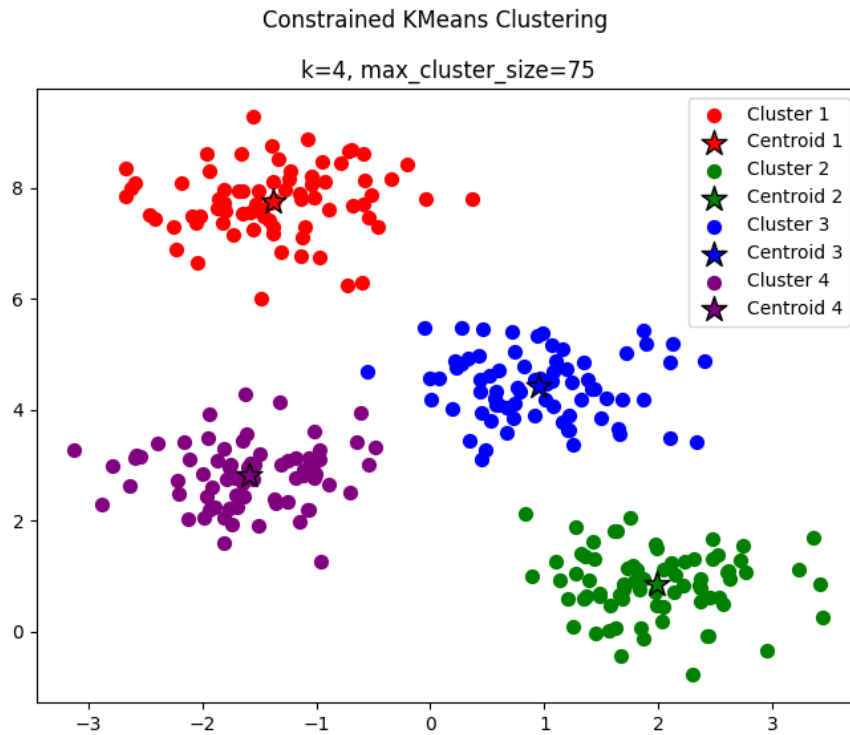
Όπως φαίνεται στο Σχήμα [2.1.8](#), ο αλγόριθμος k-means constrained παράγει συστάδες που ικανοποιούν τον περιορισμό μεγέθους, ενώ ταυτόχρονα ελαχιστοποιούν τη διακύμανση εντός της συστάδας.



(a) Παράδειγμα συσταδοποίησης με τον αλγόριθμο k-means στην Αθήνας.



(b) Οπτικοποίηση της πυκνότητας των σημείων των συστάδων στην Αθήνα.



Σχήμα 2.1.8: Παράδειγμα συσταδοποίησης με τον αλγόριθμο k-means constrained με $k = 3$ και μέγιστο πλήθος συστάδων 4.

Μαθηματικό Υπόβαθρο και Συνάρτηση Στόχου

Η βασική ιδέα του K-Means Constrained εστιάζει στην ελαχιστοποίηση της συνάρτησης στόχου του K-Means. Η συνάρτηση στόχου J που επιδιώκεται να ελαχιστοποιηθεί είναι η εξής:

$$J(C, \{S_i\}_{i=1}^k) = \sum_{i=1}^k \sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \mathbf{c}_i\|^2$$

όπου:

- C αντιπροσωπεύει το σύνολο των κέντρων των συστάδων,
- \mathbf{c}_i είναι το κέντρο της συστάδας i ,
- S_i είναι το σύνολο των σημείων που ανήκουν στη συστάδα i ,
- και \mathbf{x} είναι τα σημεία δεδομένων.

Οι περιορισμοί που εφαρμόζονται στον αλγόριθμο εξασφαλίζουν ότι κάθε συστάδα S_i έχει μέγεθος μεταξύ ενός ελάχιστου και ενός μέγιστου ορίου, δηλαδή $m_{\min} \leq |S_i| \leq m_{\max}$ για κάθε i .

2.2 Σύστημα δρομολόγησης - OSRM (Open Source Routing Machine)

Συστήματα δρομολόγησης είναι απαραίτητα για την εύρεση της βέλτιστης διαδρομής ανάμεσα σε δύο σημεία, την αφετηρία (Α) και τον προορισμό (Β), παρέχοντας σημαντικές πληροφορίες όπως η απόσταση, ο χρόνος και το κόστος της διαδρομής.

Στο πλαίσιο αυτού του κεφαλαίου, αναλύεται η λειτουργία της μηχανής δρομολόγησης Open Source Routing Machine (OSRM), περιγράφοντας την αρχιτεκτονική της, τους αλγόριθμους που χρησιμοποιεί και τις υπηρεσίες που παρέχει.

Το OSRM αποτελεί μια αποδοτική μηχανή δρομολόγησης, αναπτυγμένη σε C++, σχεδιασμένη ειδικά για την επεξεργασία δεδομένων οδικών δικτύων από το OpenStreetMap. Εχμεταλλεύεται δεδομένα του δικτύου, τα οποία καθορίζουν την τοπολογία και τα χαρακτηριστικά του, όπως δρόμοι, διασταυρώσεις, περιορισμούς στροφών, όρια ταχύτητας και άλλα, δίχως να ενσωματώνει δεδομένα για την κίνηση ή τις καιρικές συνθήκες.

Η λειτουργία της είναι αυτόνομη, χωρίς ανάγκη για εξωτερικά αιτήματα σε τρίτες υπηρεσίες, και μπορεί να εγκατασταθεί τοπικά σε έναν εξυπηρετητή. Η αρχιτεκτονική μας είναι σχεδιασμένη με ευελιξία, επιτρέποντας τη χρήση οποιασδήποτε υπηρεσίας δρομολόγησης με συμβατό διασύνδεση (interface), χωρίς να περιορίζομαστε στη χρήση μόνο του OSRM.

2.2.1 OSRM Services

Τα παρακάτω αποτελούν τα διαθέσιμα requests του OSRM API [12]. Τα requests αυτά μπορούν να χρησιμοποιηθούν για την εύρεση της βέλτιστης διαδρομής, την εύρεση του πλησιέστερου δρόμου σε ένα σημείο, τον υπολογισμό ενός πίνακα αποστάσεων/χρόνων μεταξύ πολλών σημείων, τον χαρτογραφισμό μιας ακολουθίας GPS σημείων στο οδικό δίκτυο και την παροχή λεπτομερών οδηγιών πλοήγησης. Τα requests αυτά είναι τα εξής:

Route Service

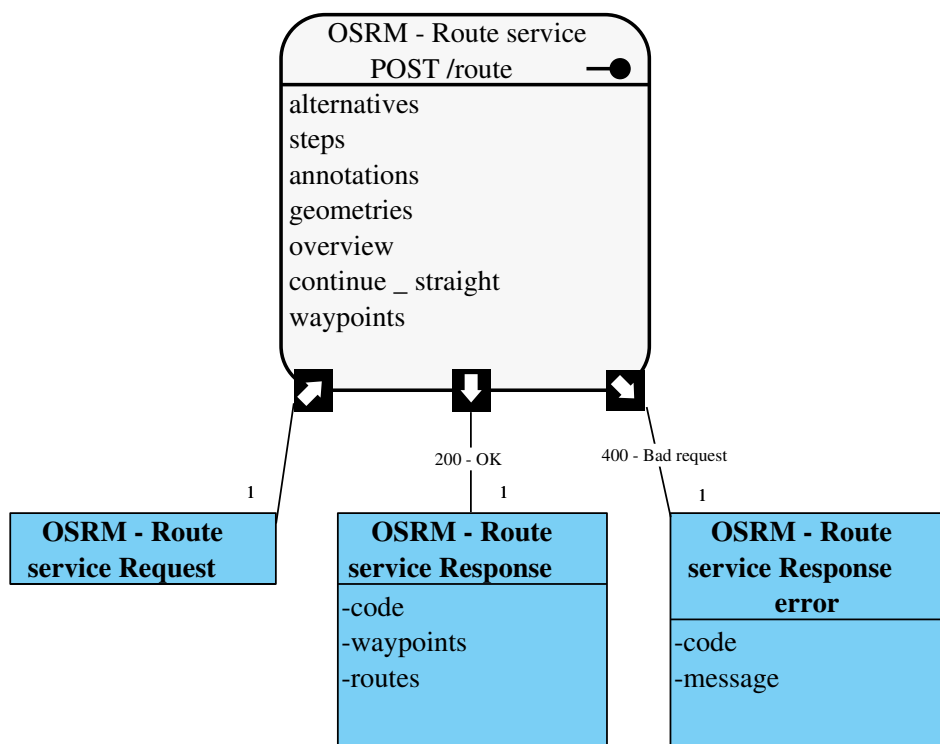
Το βασικό request για να βρεθεί το ελάχιστο μονοπάτι ανάμεσα σε μια σειρά από συντεταγμένες στην σειρά που δίνονται.

Παράδειγμα Για παράδειγμα αν θέλουμε να πάμε από Πειραιά στο Φλοίοσβο, θα χρησιμοποιήσουμε το request και η απάντηση θα είναι:

```

1  {
2  "code": "Ok",
3  "routes": [
4    {
5      "geometry": "m|pfFsykoCqGwFFg@jBmApAmCCkAq@a@mJpAgFzLuBp@\\{@`Bg@FiCsV\\
        mF0sZRcdnFqf@jCsN`Q_j@fGeM`CsLpAuBdBgAtI [|Bx@bC`CnBzFtBhXdCtExC`BtCp@xCa@~
        BgB",
6      "legs": [
7        {
8          "steps": [],
9          "summary": "",
10         "weight": 481,
11         "duration": 481,
12         "distance": 5776.4
13       }
14     ],
15     "weight_name": "routability",
16     "weight": 481,
17     "duration": 481,
18     "distance": 5776.4
19   }
20 ],
21 "waypoints": [

```

Σχήμα 2.2.1: Route Service Endpoint

```

22 {
23   "hint": "6xCJiLw7o0EAAAABwAAAAAAAAAAAA4AAAAogDIQBHSIOEAAAAAF
24   DmtQgQAAAAHAAAAAAAAADgAAAADBQEaQABpAWLmQgJiAGkBiOZCagAAbwOSwCIi",
25   "distance": 7.461109991,
26   "name": "Alexandrou Papanastasiou",
27   "location": [
28     23.658664,
29     37.938786
30   ]
31 },
32 {
33   "hint": "DRihjf__38PAAAADwAAANkAAAAuAAAAhYyEQQAAAABQmHJDk
34   lx0Qg8AAAAPAAAA2QAAAC4AAAADBQEaenFpAZ--QgJwcGkBsrlCAhEARwgSwCIi",
35   "distance": 35.198837453,
36   "name": "Poseidwnos",
37   "location": [
38     23.687546,
39     37.928607
40   ]
41 }
42 ]
43 }
    
```

Εκτός απο τις γενικές επιλογές για τις συντεταγμένες, υποστηρίζονται οι ακόλουθες επιλογές:

Παράμετρος	Τιμές	Περιγραφή
alternatives	true, false (προεπιλογή), ή Αριθμός	Αναζήτηση για εναλλακτικές διαδρομές. Η παροχή ενός αριθμού alternatives=n αναζητά έως και n εναλλακτικές διαδρομές.

steps	true, false (προεπιλογή)	Επιστροφή βημάτων διαδρομής για κάθε σκέλος της διαδρομής
annotations	true, false (προεπιλογή), nodes, distance, duration, datasources, weight, speed	Επιστρέφει επιπλέον μεταδεδομένα για κάθε συντεταγμένη κατά μήκος της διαδρομής.
geometries	polyline (προεπιλογή), polyline6, geojson	Μορφή γεωμετρίας διαδρομής που επιστρέφεται (επηρεάζει την περίληψη και ανά βήμα)
overview	simplified (προεπιλογή), full, false	Προσθήκη περίληψης γεωμετρίας είτε πλήρης, απλοποιημένη σύμφωνα με το υψηλότερο επίπεδο ζουμ που θα μπορούσε να εμφανιστεί, είτε καθόλου.
continue_straight	default (προεπιλογή), true, false	Εξαναγκάζει τη διαδρομή να συνεχίσει ευθεία στα σημεία διαδρομής περιορίζοντας τις αναστροφές εκεί ακόμη και αν θα ήταν πιο γρήγορο. Η προεπιλεγμένη τιμή εξαρτάται από το προφίλ.
waypoints	{index};{index};...	Θεωρεί τις εισαγόμενες συντεταγμένες που υποδεικνύονται από δεδομένους δείκτες ως σημεία διαδρομής στο επιστρεφόμενο αντικείμενο Match. Η προεπιλογή είναι να θεωρούνται όλες οι εισαγόμενες συντεταγμένες ως σημεία διαδρομής.

Πίνακας 2.1: Επεξήγηση των παραμέτρων για το request του OSRM route service.

Συμπίεση γεωμετρικών δεδομένων - Polylines Οι απαντήσεις με polylines για την γρηγρότερη αποστολή του request γίνονται με την μορφοποίηση google polylines.

Η μορφή πολυγραμμής της Google, γνωστή και ως Μορφότυπος Αλγορίθμου Κωδικοποιημένης Πολυγραμμής, Encoded Polyline Algorithm Format, είναι ένας αλγόριθμος συμπίεσης με απώλειες που χρησιμοποιείται για την αποθήκευση μιας σειράς από συντεταγμένες ως μία μοναδική συμβολοσειρά. Έχει σχεδιαστεί για την αποτελεσματική κωδικοποίηση και αποκωδικοποίηση πολυγραμμών, οι οποίες είναι ακολουθίες από σημεία γεωγραφικού πλάτους και μήκους που μπορούν να χρησιμοποιηθούν για την αναπαράσταση γραμμών σε έναν χάρτη. Ο αλγόριθμος κωδικοποιεί τις συντεταγμένες των σημείων χρησιμοποιώντας υπογεγραμμένες τιμές και μετατρέπει δυαδικές τιμές σε μια σειρά από κωδικούς χαρακτήρων για χαρακτήρες ASCII χρησιμοποιώντας το σχήμα κωδικοποίησης base64. Οι κωδικοποιημένες τιμές αθροίζονται με το 63 (ο χαρακτήρας ASCII '?') για να διασφαλιστεί η σωστή εμφάνιση αυτών των χαρακτήρων [13]–[15].

Μια οπτικοποίηση της απάντησης φαίνεται παρακάτω 2.2.2.

Αξιοσημείωτη είναι η παράμετρος overview, η οποία επηρεάζει την περίληψη της γεωμετρίας της διαδρομής. Αν η τιμή της είναι simplified, τότε η γεωμετρία της διαδρομής θα είναι απλοποιημένη σύμφωνα με το υψηλότερο επίπεδο ζουμ που θα μπορούσε να εμφανιστεί. Αν η τιμή της είναι full, τότε η γεωμετρία της διαδρομής θα είναι πλήρης. Αν η τιμή της είναι false, τότε δεν θα επιστραφεί καθόλου περίληψη της γεωμετρίας της διαδρομής. Μια συγκριτική απεικόνιση των τριών τιμών φαίνεται παρακάτω 2.2.3.

Είναι σημαντικό να αναφέρουμε ότι η απόσταση και ο χρόνος που απαιτούνται για να φτάσει κάποιος από το σημείο A στο σημείο B είναι διαφορετικοί από το φτάσει κάποιος από το B στο A. Αυτό οφείλεται στο γεγονός ότι το οδικό δίκτυο δεν είναι συμμετρικό.

Route from Piraeus to Floisvos
Duration: 481 seconds
Distance: 5776.4 meters



(C) OpenStreetMap contributors (C) CARTO

Σχήμα 2.2.2: Οπτικοποίηση του response του route service για την εύρεση της διαδρομής απο μια αφετηρία σε μια προορισμό.

Nearest Service

Εντοπίζει τον πλησιέστερο οδικό τμήμα απο ενα ζευγάρι συντεταγμένων. Υποστηρίζεται μόνο μία είσοδος γεωγραφικό μήκος,γεωγραφικό πλάτος για τις συντεταγμένες. Εκτός από τις γενικές επιλογές, οι ακόλουθες επιλογές υποστηρίζονται για αυτήν την υπηρεσία:

Παράμετρος	Τιμές	Περιγραφή
number	ακέραιος ≥ 1 (προεπιλογή 1)	Ο αριθμός των πλησιέστερων τμημάτων που πρέπει να επιστραφούν.

Πίνακας 2.2: Επεξήγηση των παραμέτρων για το request του OSRM nearest service.

Όσον αφορά στη χρήση της επιλογής waypoints, καθώς επιστρέφεται μόνο ένα αντικείμενο από αυτήν την υπηρεσία, η χρήση της με την επιλογή skip_waypoints ορισμένη σε true δεν είναι ιδιαίτερα χρήσιμη. Σε αυτήν την περίπτωση, θα επιστραφεί μόνο το πεδίο κωδικού.

Παράδειγμα Για παράδειγμα αν θέλουμε να βρούμε τον πλησιέστερο δρόμο απο την ΣΗΜΜΥ ΕΜΠ, θα χρησιμοποιήσουμε το [request](#)

, το οποίο θα έχει απάντηση:

```

1  {
2  "code": "Ok",
3  "waypoints": [
4    {
5      "hint": "jgoog-biD4WXAAAAlgAAAAAAAAAAAAAAAAAJ1d8QmKHeUIAAAAAAAAAAAAJcAAACW
6      AAAAAAAAAAAAAAAAAJBQEAM0ZqAW5_QwIw5moBRIBDAgAA7xAxNFat",
7      "nodes": [2063446108, 1302996113],
8      "distance": 23.753144772,
9      "name": "",
10     "location": [23.78296, 37.977966]
11   }
12 ]
13 }
```

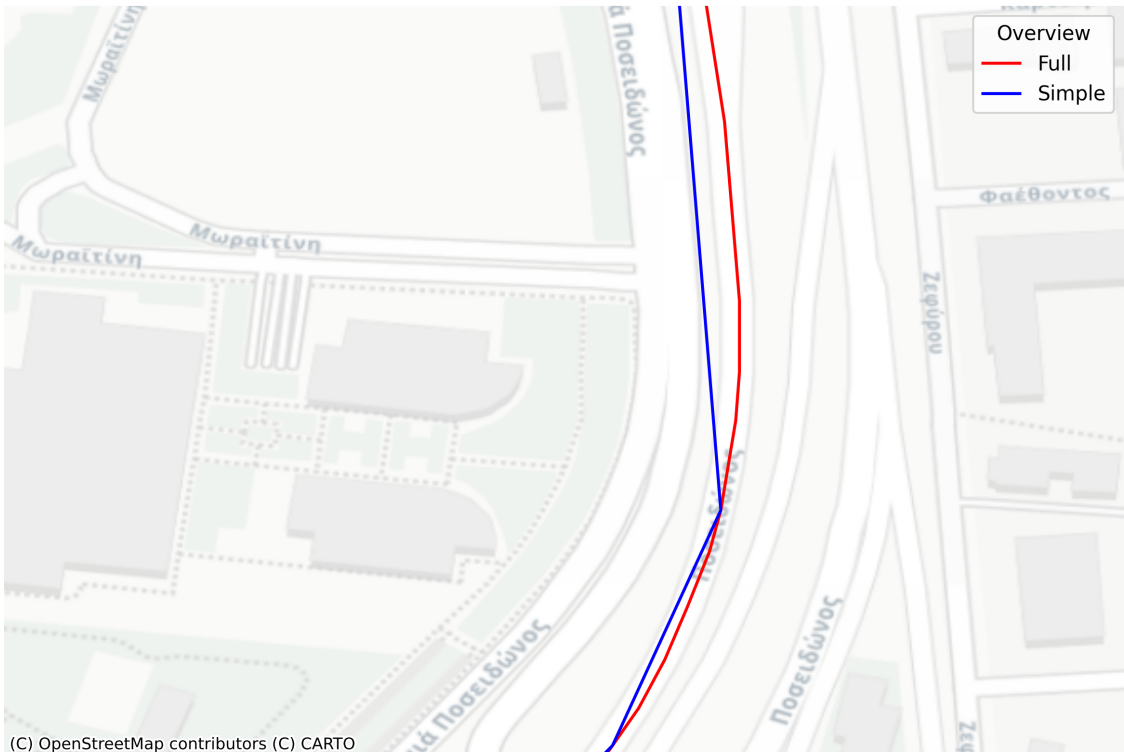
Route from Piraeus to Floisvos
Duration: 481 seconds Distance: 5776.4 meters
Overview: Full



(C) OpenStreetMap contributors (C) CARTO

(a) overview=full

Part of route from Piraeus to Floisvos
Duration: 481 seconds Distance: 5776.4 meters
Compare Overview: Full and Simple

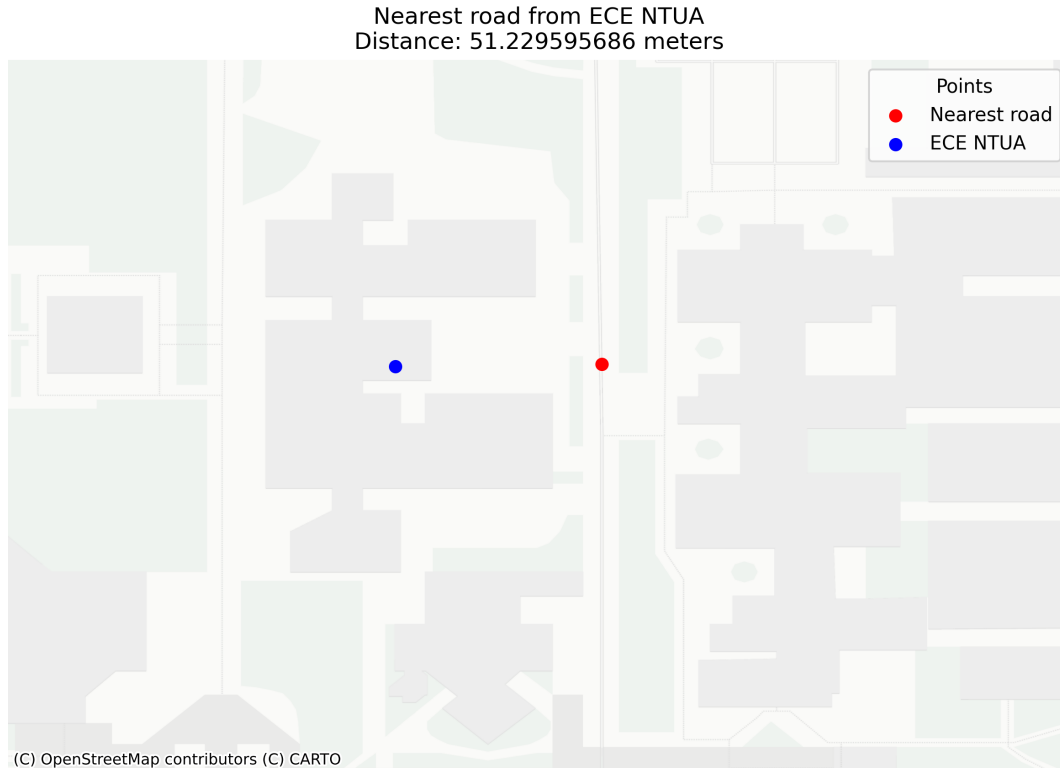


(C) OpenStreetMap contributors (C) CARTO

(b) Συγκριση των 2 τιμών της παραμέτρου overview

Σχήμα 2.2.3: Route service - Παράμετρος overview

Μια οπτικοποίηση της απάντησης φαίνεται παρακάτω 2.2.4.



Σχήμα 2.2.4: Αποτέλεσμα του request για την εύρεση του πλησιέστερου δρόμου απο την ΣΗΜΜΥ ΕΜΠ

Table Service

Το table service, υπολογίζει τη διάρκεια της ταχύτερης διαδρομής μεταξύ όλων των ζευγών των παρεχόμενων συντεταγμένων. Επιστρέφει τις διάρκειες ή τις αποστάσεις ή και τα δύο μεταξύ των ζευγών συντεταγμένων. Σημειώνεται ότι οι αποστάσεις δεν αντιπροσωπεύουν την κοντινότερη απόσταση μεταξύ δύο συντεταγμένων, αλλά μάλλον τις αποστάσεις των ταχύτερων διαδρομών. Η διάρκεια είναι σε δευτερόλεπτα και οι αποστάσεις σε μέτρα.

Επιλογή	Τιμές	Περιγραφή
sources	{index} ; {index} ; {index} ... ή all (προεπιλογή)	Χρήση τοποθεσίας με δεδομένο δείκτη ως πηγή.
destinations	{index} ; {index} ; {index} ... ή all (προεπιλογή)	Χρήση τοποθεσίας με δεδομένο δείκτη ως προορισμό.
annotations	duration (προεπιλογή), distance, ή duration,distance	Επιστροφή του ζητούμενου πίνακα ή πινάκων στην απόκριση.
fallback_speed	διπλό > 0	Αν δεν βρεθεί διαδρομή μεταξύ ζεύγους πηγής/προορισμού, υπολογίζεται η απόσταση as-the-crow-flies, στη συνέχεια χρησιμοποιείται αυτή η ταχύτητα για εκτίμηση διάρκειας.

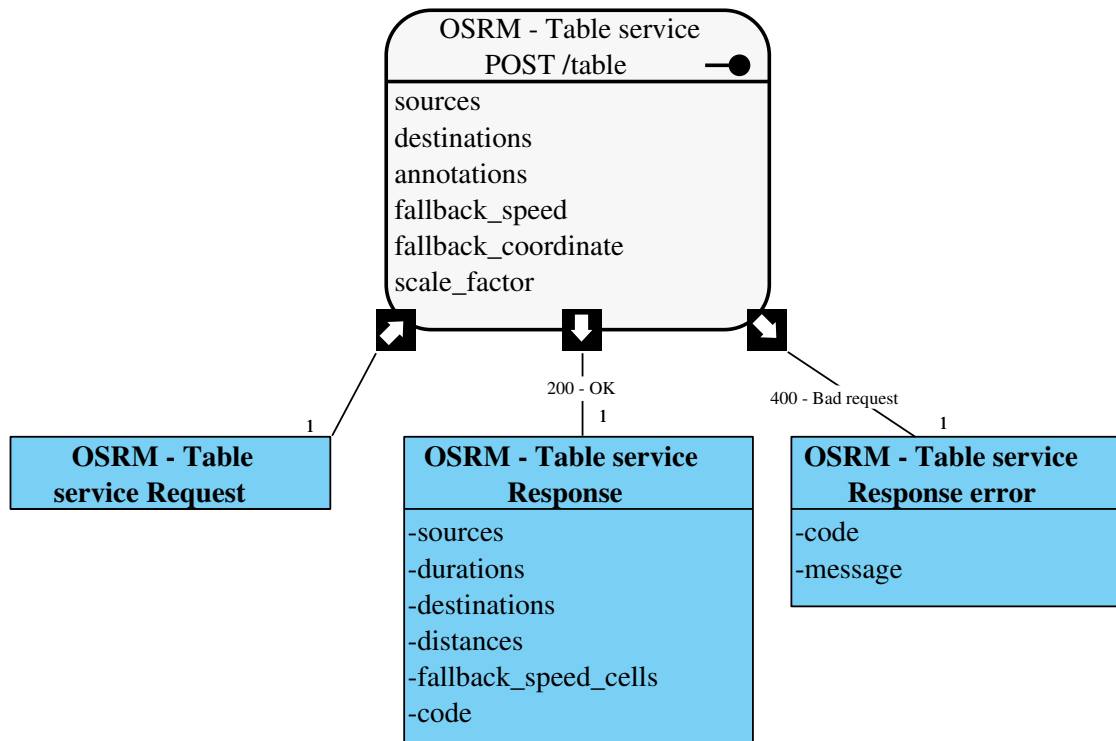
Επιλογή	Τιμές	Περιγραφή
fallback_coordinate	input (προεπιλογή), ή snapped	Κατά τη χρήση fallback_speed, χρησιμοποιείται η συντεταγμένη που παρέχει ο χρήστης (input), ή η τοποθεσία που έχει ευθυγραμμιστεί (snapped) για τον υπολογισμό αποστάσεων.
scale_factor	διπλό > 0	Χρησιμοποιείται σε συνδυασμό με annotations=durations. Κλιμακώνει τις τιμές διάρκειας του πίνακα με αυτόν τον αριθμό.

Παράδειγμα Για παράδειγμα αν θέλουμε να βρούμε τον πίνακα αποστάσεων μεταξύ περιοχών κοντά του Πειραιά, θα χρησιμοποιήσουμε το `request` και θα έχει απάντηση:

```

1  {
2  "code": "Ok",
3  "distances": [
4    [0,5508.7,6186.7,4848.7,3272.5],
5    [6249.5,0,2281,3804.3,6833.3],
6    [5447.7,1993.6,0,3002.6,6031.5],
7    [3819.9,2561.3,2656.7,0,4433.9],
8    [2239.7,4763.9,4899.3,4052.6,0]
9  ],
10 "destinations": [
11   {
12     "hint": "eDhJhX44SYVHAAAAmWAAAAAAAAAAAAAAAAAeZrQVrO
13     KUEAAAAAAAAAAAAACMAAAAaAAAAAAAAAAAAAADBQEAXQFpAS_mQ
14     gJ-AWkBVeZCAGAAALwISwCIi",
15     "distance": 7.533799006,
16     "name": "Parallilos",
17     "location": [23.658949,37.938735]
18   },
19   {
20     "hint": "DRlhjf__38CAAAADwAAANkAAAAuAAAA9URSQM6H
21     VEFQmHJDkIx0QgIAAAAPAAAA2QAAAC4AAAAADBQEAFXFpAfi-Q
22     gJib2kBxb1CAhEARwgSwCIi",
23     "distance": 51.223968877,
24     "name": "Poseidonos",
25     "location": [23.687445,37.928696]
26   },
27   {
28     "hint": "vuLxiv__38CAAAAEgAAAAAAAAAJAAAAdgMvQCOS
29     jUEAAAAAJLYYQqIAAAASAAAAAAAAAAAAkAAAAADBQEAAAbpAUJQ
30     gLGr2kBMslCAgAA3xQSwCIi",
31     "distance": 5.781541933,
32     "name": "",
33     "location": [23.703553,37.931337]
34   },
35   {
36     "hint": "e23Sjv__38JAAACQAAAHcBAAALAAAAcr6-QAAA
37     AADlboJD_B76QqkAAAAJAAAdwEAAAaAAAAADBQEAE5pAWDtQ
38     gKlFmkBIe1CAhKAHwISwCIi",
39     "distance": 14.772307861,
40     "name": "",
41     "location": [23.690769,37.940576]
42   },
43   {
44     "hint": "w10kgjlvkoX4AAAAABgAAADkAAAAuAAAAAGGVPQnXY
45     jz9h9D5BERMYQXwAAAAADAAAAHQAAABcAAAAADBQEAE5xJpAXoDQ
46     wLfFmkBGQRDAgEAjwoSwCIi",
47     "distance": 91.055235396,
48     "name": "",
49     "location": [23.663335,37.946234]
50   }
51 ],
52 "durations": [

```



Σχήμα 2.2.5: Table Service Endpoint

```

53 [0,460.2,527.8,547.3,367.7],
54 [546.5,0,228.2,428.8,616.8],
55 [471,210.3,0,353.3,541.3],
56 [383.3,249,264.6,0,471.4],
57 [248.6,395.7,460.7,467.2,0]
58 ],
59 "sources": [
60 {
61   "hint": "eDhJhX44SYVHAAAAAMwAAAAAAAAAAAAAAAAAeZrQVrO
62   KUEAAAAAAAAAACMAAAAaAAAAAAAAAAAAAAAAADBQEaxQFpAS_mQ
63   gJ-AWkBVeZCAGaALwISwCIi",
64   "distance": 7.533799006,
65   "name": "Parallilos",
66   "location": [23.658949,37.938735]
67 },
68 {
69   "hint": "DRiHjf__38CAAAADwAAANkAAAAuAAAA9URSQM6H
70   VEFQmHJDk1xOQgIAAAA2QAAAC4AAAAADBQEAFxFpAfi-Q
71   gJib2kBxb1CAhEARwgSwCIi",
72   "distance": 51.223968877,
73   "name": "Poseidonos",
74   "location": [23.687445,37.928696]
75 },
76 {
77   "hint": "vuLxiv__38CAAAAEgAAAAAAAAAJAAAAgMvQCOS
78   jUEAAAAAJlYYQQIAAAAASAAAAAaAAAAAaAAAAADBQEAAbBpAUJQ
79   gLGr2kBMslCAgAA3xQSwCIi",
80   "distance": 5.781541933,
81   "name": "",
82   "location": [23.703553,37.931337]
83 },
84 {

```

```

85   "hint": "e23Sjv___38JAAAAcQAAAAHcBAAALAAAAcr6-QAAA
86   AADlboJD_B76QAkAAAAJAAAAAwEAAAAsAAAADBQEAE5pAWDtQ
87   gKlFmkBIe1CAhkAHwISwCIi",
88   "distance": 14.772307861,
89   "name": "",
90   "location": [23.690769, 37.940576]
91 },
92 {
93   "hint": "w10kgjlvkoX4AAAABgAAADkAAAAuAAAAGGVPQnXY
94   jz9h9D5BERMYQXwAAAADAAAAHQAAABcAAAADBQEAE5xJpAXoDQ
95   wLfFmkBGQRDAgEAjwoSwCIi",
96   "distance": 91.055235396,
97   "name": "",
98   "location": [23.663335, 37.946234]
99 }
100 ]
101 }

```

Απο την απάντηση μπορούμε να δημιουργήσουμε πολύ απλά τον πίνακα αποστάσεων μεταξύ των πόλεων 2.4.

Αφετηρία	Προορισμός				
	Πειραιάς	Φλοίσβο	Πλ. Νταβάρη	Κ.Π.Ι. Σταύρος Νιάρχος	Στάδιο Καραϊσκάκη
Πειραιάς	0.0	460.2	527.8	547.3	367.7
Φλοίσβο	546.5	0.0	228.2	428.8	616.8
Πλ. Νταβάρη	471.0	210.3	0.0	353.3	541.3
Κ.Π.Ι. Σταύρος Νιάρχος	383.3	249.0	264.6	0.0	471.4
Στάδιο Καραϊσκάκη	248.6	395.7	460.7	467.2	0.0

Πίνακας 2.4: Πίνακας διάρκειας οδήγησης μεταξύ περιχών της Αττικής (σε δευτερόλεπτα)

Trip Service

Το trip service, λύνει το πρόβλημα του περιοδεύοντος πωλητή (TSP) για μια σειρά απο σημεία.

Η υπηρεσία trip λύνει το Πρόβλημα του Περιοδεύοντος Πωλητή χρησιμοποιώντας μια άπληστη ευρετική (αλγόριθμος εισαγωγής του πιο μακρινού) για 10 ή περισσότερα σημεία και χρησιμοποιεί βίαιη δύναμη για λιγότερα από 10 σημεία. Η επιστρεφόμενη διαδρομή δεν χρειάζεται να είναι η ταχύτερη διαδρομή. Καθώς το TSP είναι NP-hard επιστρέφει μόνο μια προσέγγιση.

Παράμετρος	Τιμές	Περιγραφή
roundtrip	true (προεπι-λογή), false	Η επιστρεφόμενη διαδρομή είναι μια κυκλική διαδρομή (η διαδρομή επιστρέφει στην πρώτη τοποθεσία)
source	any (προεπι-λογή), first	Η επιστρεφόμενη διαδρομή ξεκινά από οποιαδήποτε ή την πρώτη συντεταγμένη
destination	any (προεπι-λογή), last	Η επιστρεφόμενη διαδρομή τελειώνει σε οποιαδήποτε ή την τελευταία συντεταγμένη
steps	true, false (προεπιλογή)	Επιστρέφει οδηγίες διαδρομής για κάθε ταξίδι
annotations	true, false (προεπι-λογή), nodes, distance, duration, datasources, weight, speed	Επιστρέφει επιπλέον μεταδεδομένα για κάθε συντεταγμένη κατά μήκος της γεωμετρίας της διαδρομής

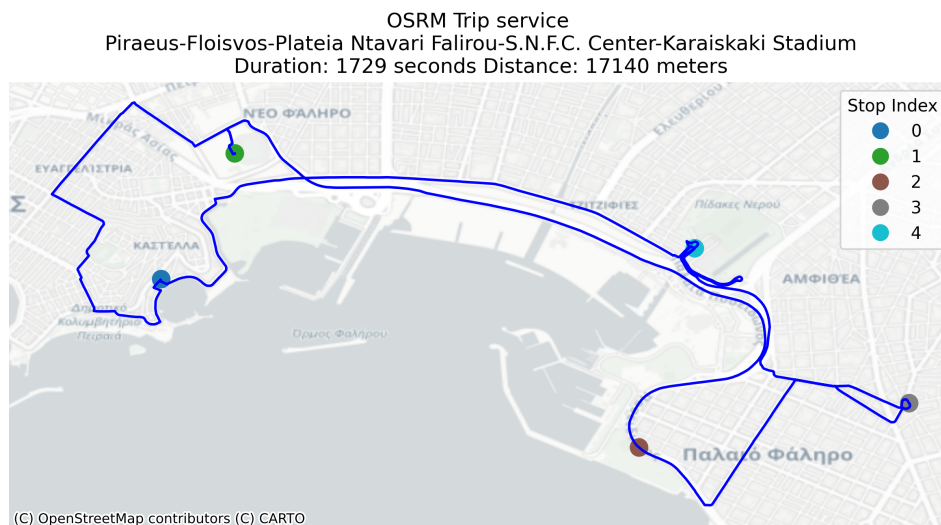
geometries	polyline (προεπιλογή), polyline6, geojson	Μορφή γεωμετρίας διαδρομής που επιστρέφεται (επηρεάζει την περίληψη και ανά βήμα)
overview	simplified (προεπιλογή), full, false	Προσθήκη περίληψης γεωμετρίας είτε πλήρης, απλοποιημένη ανάλογα με το υψηλότερο επίπεδο ζουμ που θα μπορούσε να εμφανιστεί, είτε κα- θόλου

Πίνακας 2.5: Επεξήγηση παραμέτρων για το request του OSRM trip service.

Διόρθωση Αρχικών και Τελικών Σημείων

Είναι δυνατόν να οριστεί ρητά το αρχικό ή το τελικό σημείο του ταξιδιού. Όταν το source ορίζεται ως **first**, η πρώτη συντεταγμένη χρησιμοποιείται ως αρχικό σημείο του ταξιδιού στην έξοδο. Όταν ο προορισμός ορίζεται ως **last**, η τελευταία συντεταγμένη θα χρησιμοποιηθεί ως προορισμός του ταξιδιού στην επιστρεφόμενη έξοδο. Εάν ορίσουμε **any**, οποιαδήποτε από τις συντεταγμένες μπορεί να χρησιμοποιηθεί ως το πρώτο ή τελευταίο σημείο στην έξοδο.

Παράδειγμα Για παράδειγμα αν θέλουμε να βρούμε την βέλτιστη διαδρομή μεταξύ περιοχών θα κά-
νουμε το [request](#). Οπτικοποιώντας το αποτέλεσμα του request, θα ακολουθήσουμε την παρακάτω δι-
αδρομή όπως φαίνονται στο σχήμα 2.2.7.



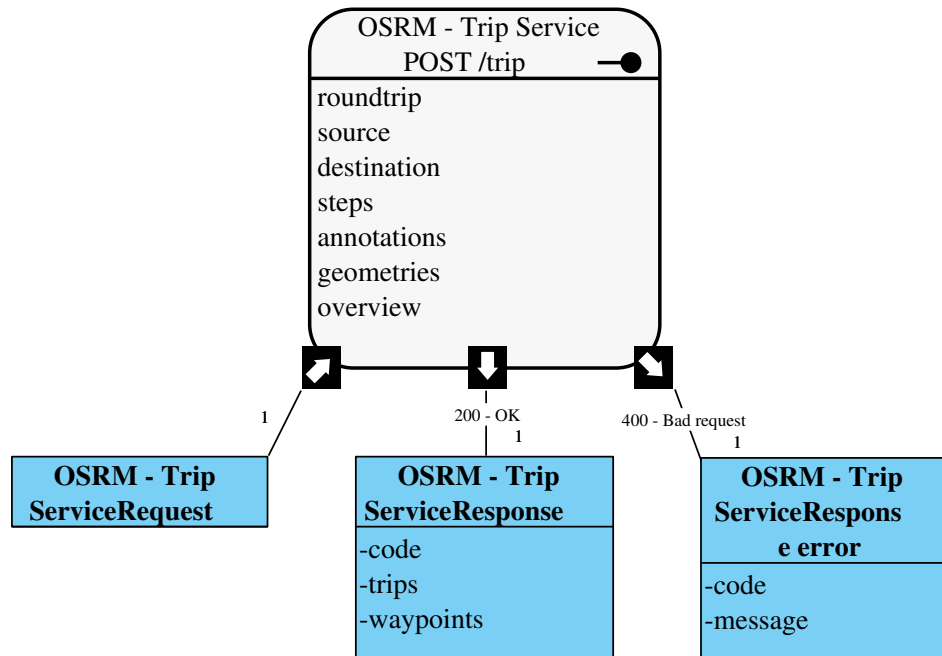
Σχήμα 2.2.7: Αποτέλεσμα του request για την εύρεση της βέλτιστης διαδρομής για το πρόβλημα του περιοδεύοντος πωλητή

Match

Χαρτογραφεί μια ακολουθία από GPS σημεία στο οδικό δίκτυο. Επειδή δεν έχει χρησιμοποιηθεί στην παρούσα εργασία, δεν θα αναλυθεί περαιτέρω.

Tile

Επιστρέφει τα δεδομένα του χάρτη σε μορφή raster tiles. Επειδή δεν έχει χρησιμοποιηθεί στην παρούσα εργασία, δεν θα αναλυθεί περαιτέρω.



Σχήμα 2.2.6: Trip Service Endpoint

2.2.2 Τρόπος λειτουργίας

Για την καλύτερη κατανόηση του προβλήματος της εύρεσης της συντομότερης διαδρομής, παρουσιάζεται πως λειτουργεί το OSRM.

Συντομότερη διαδρομή Σε αυτή την διπλωματική, ορίζουμε ως συντομότερη διαδρομή μεταξύ δύο σημείων την διαδρομή με τον λιγότερο χρόνο οδήγησης. Αυτός ο ορισμός είναι σύμφωνος και με το OSRM. Ο χρόνος οδήγησης εξαρτάται από την απόσταση, την ταχύτητα, τα όρια ταχύτητας, την κίνηση και την ποιότητα του δρόμου. Επομένως, η εύρεση της συντομότερης διαδρομής απαιτεί την εξερεύνηση των διαφόρων παραμέτρων που επηρεάζουν τον χρόνο οδήγησης σε πραγματικό χρόνο.

Στο OSRM, η εύρεση της συντομότερης διαδρομής δεν γίνεται πάντα με την εξερεύνηση όλων των διαδρομών. Αντίθετα, χρησιμοποιείται μια προσέγγιση που βασίζεται στον αλγόριθμο Dijkstra [2.2.2](#).

Ο αλγόριθμος Dijkstra

Ο αλγόριθμος του Dijkstra αναπτύχθηκε από τον Ολλανδό επιστήμονα Edsger W. Dijkstra το 1959 [\[16\]](#). Ο αλγόριθμος αυτός σχεδιάστηκε για την εύρεση της συντομότερης διαδρομής ανάμεσα σε ένα σημείο εκκίνησης και όλα τα άλλα σημεία σε έναν γράφο, ο οποίος περιέχει θετικά βάρη στις ακμές του. Η βασική ιδέα πίσω από τον αλγόριθμο είναι η εξέταση των πιο κοντινών μη εξερευνημένων κόμβων και η σταδιακή επέκταση της συντομότερης διαδρομής προς τα έξω, μέχρι να επιτευχθεί η εξερεύνηση όλων των κόμβων.

Στην γραμμή 2 του αλγορίθμου, αρχικοποιείται το σύνολο των κόμβων που δεν έχουν εξερευνηθεί. Στις γραμμές 3-6, αρχικοποιούνται οι αποστάσεις και οι προηγούμενοι κόμβοι για κάθε κόμβο. Στην γραμμή 7, η απόσταση του αρχικού κόμβου προς τον εαυτό του θέτεται σε μηδέν. Στην επανάληψη που ακολουθεί (γραμμές 8-20), εξετάζεται ο κόμβος με την μικρότερη απόσταση από τον αρχικό κόμβο και αφαιρείται από το σύνολο των μη εξερευνημένων κόμβων. Αυτή η αφαίρεση του ήδη εξερευνημένου κόμβου, εξασφαλίζει ότι η επανάληψη θα τερματίσει μόλις εξερευνηθούν όλοι οι κόμβοι. Στη συνέχεια, εξετάζονται όλοι οι γειτονικοί κόμβοι του εξερευνημένου κόμβου και ενημερώνονται οι αποστάσεις τους αν βρεθεί μικρότερη διαδρομή.

Algorithm 4 Dijkstra's Algorithm

```

1: procedure DIJKSTRA( $G, v$ )                                ▷  $G$  is the graph and  $v$  is the source vertex
2:    $Q \leftarrow V(G)$                                        ▷ Initialize the set of unvisited nodes
3:   for all  $vertex \in V(G)$  do
4:      $dist[vertex] \leftarrow \infty$                          ▷ Set initial distances to infinity
5:      $prev[vertex] \leftarrow \text{undefined}$                  ▷ Previous node in optimal path
6:   end for
7:    $dist[v] \leftarrow 0$                                     ▷ The distance from the source to itself is zero
8:   while  $Q \neq \emptyset$  do
9:      $u \leftarrow \min_{x \in Q} dist[x]$                        ▷ Node with the least distance
10:     $Q \leftarrow Q - \{u\}$                                   ▷ Remove  $u$  from  $Q$ 
11:    for all  $vertex \in \text{Adj}(u)$  do                       ▷ Where  $\text{Adj}(u)$  is the set of neighbors of  $u$ 
12:       $alt \leftarrow dist[u] + \text{length}(u, neighbor)$ 
13:      if  $alt < dist[neighbor]$  then
14:         $dist[neighbor] \leftarrow alt$ 
15:         $prev[neighbor] \leftarrow u$ 
16:      end if
17:    end for
18:  end while
19:  return  $dist, prev$ 
20: end procedure

```

Multilevel Dijkstra

Προσαρμόσιμος σχεδιασμός διαδρομής (CRP): Ευελιξία στην πλοήγηση Η έλευση του Προσαρμόσιμου Σχεδιασμού Διαδρομής, Customizable Route Planning (CRP) [17] σηματοδοτεί ένα σημαντικό ορόσημο στην εξέλιξη των συστημάτων πλοήγησης, καλύπτοντας την κρίσιμη ανάγκη για δυναμική, αποτελεσματική και προσαρμοσμένη στο χρήστη βελτιστοποίηση της διαδρομής. Αναπτυγμένο με γνώμονα τον σύγχρονο τρόπο μετακινήσεων και απαιτήσεων, το CRP ξεπερνά τα παραδοσιακά παραδείγματα σχεδιασμού διαδρομής εισάγοντας έναν αλγόριθμο Dijkstra πολλαπλών επιπέδων που προσαρμόζεται απρόσκοπτα στις μεταβαλλόμενες συνθήκες και τις προσωπικές προτιμήσεις.

Στον πυρήνα του CRP βρίσκεται ένα εξελιγμένο αλγοριθμικό πλαίσιο που έχει σχεδιαστεί για τη βελτιστοποίηση της επιλογής διαδρομής σε τεράστια οδικά δίκτυα. Σε αντίθεση με τις συμβατικές μεθόδους που υπολογίζουν στατικά τις συντομότερες διαδρομές, το CRP ενσωματώνει μια διαδικασία τριών σταδίων: προεπεξεργασία, προσαρμογή και αναζήτηση. Αυτή η καινοτόμος προσέγγιση εξασφαλίζει άνευ προηγουμένου ευελιξία, επιτρέποντας προσαρμογές σε πραγματικό χρόνο με βάση την κυκλοφορία, τον καιρό, το κλείσιμο δρόμων, ακόμη και κριτήρια που καθορίζονται από τον χρήστη, όπως γραφικές διαδρομές ή αποφυγή διοδίων.

Προεπεξεργασία: Δημιουργία ενός επεκτάσιμου υποβάθρου Το στάδιο της προεπεξεργασίας θέτει τις βάσεις για την αποτελεσματικότητα του CRP. Κατά τη διάρκεια αυτής της φάσης, ο αλγόριθμος κατασκευάζει μια ιεραρχική αναπαράσταση του οδικού δικτύου, αφαιρώντας τα βασικά χαρακτηριστικά, ενώ παραλείπει τις λεπτομέρειες που αφορούν τις μετρικές. Αυτό το ανεξάρτητο από τις μετρικές μοντέλο επιτρέπει στο σύστημα να εκτελεί το μεγαλύτερο μέρος των υπολογιστικών βαρών εκ των προτέρων, μειώνοντας σημαντικά το χρόνο που απαιτείται για τους επακόλουθους υπολογισμούς διαδρομών.

Εξατομίκευση: Προσαρμογή των δρομολογίων στις συνθήκες και τις προτιμήσεις σε πραγματικό χρόνο Η προσαρμογή είναι το σημείο στο οποίο το CRP ξεχωρίζει. Αξιοποιώντας τα προεπεξεργασμένα δεδομένα, το σύστημα μπορεί να ενσωματώσει γρήγορα διάφορες μετρικές δρομολόγησης χωρίς την ανάγκη για εκτενή επαναυπολογισμό. Είτε προσαρμόζεται σε ξαφνική κυκλοφοριακή συμφόρηση, είτε ενσωματώνει ενημερωμένες πληροφορίες για τα οδικά έργα, είτε προσαρμόζεται

στις μοναδικές προτιμήσεις των χρηστών, το στάδιο προσαρμογής του CRP εξασφαλίζει ότι ο σχεδιασμός διαδρομών παραμένει τόσο επίκαιρος όσο και ιδιαίτερα εξατομικευμένος.

Αναζήτηση γρήγορων, ακριβών οδηγιών Στην τελική φάση, το CRP χρησιμοποιεί τα προσαρμοσμένα δεδομένα για την γρήγορη εκτέλεση request, παρέχοντας στους χρήστες βελτιστοποιημένες οδηγίες που αντικατοπτρίζουν τις τελευταίες συνθήκες και τις ειδικές απαιτήσεις τους. Η διαδικασία των request επωφελείται από την πολυεπίπεδη δομή του αλγορίθμου, επιτρέποντας τον γρήγορο εντοπισμό των συντομότερων διαδρομών λαμβάνοντας υπόψη όλους τους σχετικούς παράγοντες.

Συμπέρασμα ως προς την λειτουργία Ο προσαρμοσμένος σχεδιασμός διαδρομής αντιπροσωπεύει μια αλλαγή παραδείγματος στην τεχνολογία πλοήγησης, προσφέροντας μια λύση που δεν είναι μόνο ταχύτερη και πιο ευέλικτη, αλλά και πιο προσαρμοσμένη στις ατομικές ανάγκες και προτιμήσεις των χρηστών. Ενσωματώνοντας προηγμένες αλγοριθμικές στρατηγικές με βαθιά κατανόηση των σύγχρονων προκλήσεων κινητικότητας, το CRP θέτει νέα πρότυπα για εξατομικευμένο, αποτελεσματικό και προσαρμοσμένο σχεδιασμό διαδρομών.

Αλγοριθμικές καινοτομίες και δομές δεδομένων Η ουσία της επιτυχίας του CRP έγκειται στην καινοτόμο χρήση της κατάτμησης σε πολλαπλά επίπεδα και στην εκμετάλλευση των ιεραρχικών ιδιοτήτων του οδικού δικτύου. Με τη διάσπαση του δικτύου σε διαχειρίσιμα τμήματα και επίπεδα, το CRP επιτυγχάνει μια ισορροπία μεταξύ κοκκομετρίας και υπολογιστικής απόδοσης. Αυτή η τμηματοποίηση επιτρέπει στον αλγόριθμο να πλοηγείται γρήγορα στο δίκτυο, εντοπίζοντας τις βέλτιστες διαδρομές και ελαχιστοποιώντας παράλληλα τους περιττούς υπολογισμούς για μακρινές ή άσχετες διαδρομές.

Ένα κρίσιμο στοιχείο του συστήματος CRP είναι η μοναδική δομή δεδομένων του, η οποία έχει σχεδιαστεί για να επιτρέπει την ταχεία προσαρμογή για ποικίλες μετρικές. Αυτή η δομή υποστηρίζει την αποτελεσματική αποθήκευση και ανάκτηση πληροφοριών, επιτρέποντας στο σύστημα να προσαρμόζεται γρήγορα στις αλλαγές των προτιμήσεων ή των συνθηκών δρομολόγησης. Η χρήση προηγμένων τεχνικών συμπίεσης γραφημάτων βελτιώνει περαιτέρω τις επιδόσεις του συστήματος, μειώνοντας το αποτύπωμα μνήμης και επιτρέποντας ταχύτερους χρόνους επεξεργασίας.

Τεχνικές προσαρμογής Ένα από τα χαρακτηριστικά που ξεχωρίζουν στο CRP είναι η δυνατότητα προσαρμογής του, η οποία βασίζεται σε ένα εξελιγμένο σύνολο αλγορίθμων για τη δυναμική προσαρμογή των μετρικών δρομολόγησης χωρίς την ανάγκη πλήρους αναθεώρησης του συστήματος. Η διαδικασία αυτή περιλαμβάνει τον επανυπολογισμό των βαρών και του κόστους που σχετίζονται με διαφορετικές διαδρομές στο δίκτυο, με βάση κριτήρια που ορίζονται από τον χρήστη ή δεδομένα που εισάγονται σε πραγματικό χρόνο. Η ευελιξία αυτής της διαδικασίας προσαρμογής είναι ζωτικής σημασίας για την προσαρμογή σε σενάρια όπως η κίνηση σε ώρες αιχμής, το κλείσιμο δρόμων ή οι εξατομικευμένες προτιμήσεις δρομολόγησης, όπως οι γραφικές έναντι των ταχύτερων διαδρομών.

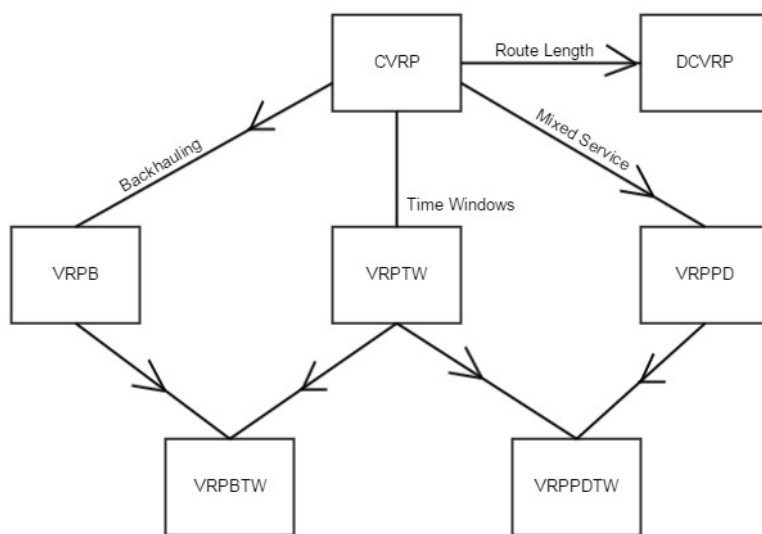
Η ανάπτυξη του Προσαρμοζόμενου Σχεδιασμού Διαδρομών αποτελεί άλμα προς τα εμπρός στον τομέα της τεχνολογίας βελτιστοποίησης διαδρομών. Ενσωματώνοντας αλγοριθμικές στρατηγικές αιχμής με μια προσέγγιση της πλοήγησης με επίκεντρο τον χρήστη, το CRP προσφέρει μια κλιμακούμενη, ευέλικτη και αποτελεσματική λύση στις προκλήσεις των σύγχρονων μεταφορών και της κινητικότητας. Καθώς το σύστημα συνεχίζει να εξελίσσεται, οι πιθανές εφαρμογές και ο αντίκτυπός του τόσο στην ατομική κινητικότητα όσο και στην ευρύτερη υποδομή μεταφορών υπόσχονται να είναι σημαντικές.

Τελική ανάλυση Σε τελική ανάλυση, το σύστημα Προσαρμοζόμενου Σχεδιασμού Διαδρομής (CRP) αποτελεί σημαντική πρόοδο στις τεχνολογίες πλοήγησης, αντιμετωπίζοντας τόσο τις τρέχουσες όσο και τις μελλοντικές προκλήσεις στη βελτιστοποίηση διαδρομών. Η πολυεπίπεδη προσέγγισή του όχι μόνο εξασφαλίζει αποτελεσματική και ευέλικτη δρομολόγηση σε σενάρια πραγματικού χρόνου, αλλά θέτει επίσης τις βάσεις για περαιτέρω καινοτομίες στον τομέα. Οι δυνατότητες του CRP να επηρεάσει ένα ευρύ φάσμα εφαρμογών -από την προσωπική πλοήγηση έως την εφοδιαστική μεγάλης κλίμακας και τον αστικό σχεδιασμό- είναι τεράστιες, υπογραμμίζοντας τη σημασία του ως εργαλείου για σύγχρονες λύσεις μεταφορών. Καθώς η τεχνολογία εξελίσσεται και η ζήτηση για εξυπνότερα, πιο προσαρμοστικά συστήματα

αυξάνεται, το CRP ξεχωρίζει ως απόδειξη της δύναμης του καινοτόμου σχεδιασμού αλγορίθμων και του αντίκτυπού του στη βελτίωση της καθημερινής μας ζωής και κινητικότητας. Η διερεύνηση των δυνατοτήτων, των προκλήσεων και των δυνατοτήτων του CRP στο [17] αποτέλεσε ένα καίριο βήμα προς τον επαναπροσδιορισμό του τρόπου με τον οποίο πλοηγούμαστε στον κόσμο μας, σήμερα.

2.2.3 Αναδρομή στο Vehicle Routing Problem - VRP

Οι Fisher και Jaikumar το 1981 [18] εξετάζουν μια κοινή παραλλαγή του προβλήματος δρομολόγησης οχημάτων (VRP) στο οποίο ένας στόλος οχημάτων παραδίδει προϊόντα που είναι αποθηκευμένα σε μια κεντρική αποθήκη για να ικανοποιήσει τις παραγγελίες των πελατών. Η απόφαση δρομολόγησης περιλαμβάνει τον προσδιορισμό του ποιος από τις απαιτήσεις θα ικανοποιηθεί κάθε όχημα και ποια διαδρομή θα ακολουθήσει κάθε όχημα για την εξυπηρέτηση της ζήτησης που του έχει ανατεθεί, ώστε να ελαχιστοποιηθεί το συνολικό κόστος παράδοσης. Παρουσιάζουν μια ευρετική μέθοδο για το πρόβλημα αυτό, στην οποία η ανάθεση των πελατών στα οχήματα επιτυγχάνεται με την επίλυση ενός γενικευμένου προβλήματος ανάθεσης με μια αντικειμενική συνάρτηση που προσεγγίζει το κόστος παράδοσης. Ο στόχος των Hertz, Laporte & Gendreau το 1994 [19] είναι να περιγράψουν το TABUROUTE, μια νέα ευρετική αναζήτηση tabu για το πρόβλημα δρομολόγησης οχημάτων με περιορισμούς χωρητικότητας και μήκους διαδρομής. Ο αλγόριθμος εξετάζει μια ακολουθία γειτονικών λύσεων που λαμβάνονται με την επανειλημμένη αφαίρεση μιας κορυφής από την τρέχουσα διαδρομή της και την επανατοποθέτησή της σε μια άλλη διαδρομή. Οι Rochat και Taillard [20] παρουσιάζουν μια πιθανολογική τεχνική για τη διαφοροποίηση, εντατικοποίηση και παραλληλισμό μιας τοπικής αναζήτησης προσαρμοσμένης για την επίλυση προβλημάτων δρομολόγησης οχημάτων. Αποδεικνύεται ότι οι αποτελεσματικές αναζητήσεις tabu πρώτου επιπέδου για προβλήματα δρομολόγησης οχημάτων μπορούν να βελτιωθούν σημαντικά με αυτή την τεχνική. Έπειτα παρουσιάστηκε μια ενοποιημένη ευρετική αναζήτηση tabu για το πρόβλημα δρομολόγησης οχημάτων με χρονικά παράθυρα και για δύο σημαντικές γενικεύσεις: τα περιοδικά και τα προβλήματα δρομολόγησης οχημάτων με πολλαπλές αποθήκες με χρονικά παράθυρα [21]. Η απόδοση της ευρετικής αξιολογείται με τη σύγκρισή της με εναλλακτικές μεθόδους σε περιπτώσεις αναφοράς του προβλήματος δρομολόγησης οχημάτων με χρονικά παράθυρα. Αργότερα το πρόβλημα μετασχηματίστηκε σε πρόβλημα δρομολόγησης οχημάτων με χρονικά παράθυρα (VRPTW) [22]. Οι διαδρομές πρέπει να σχεδιάζονται με τέτοιο τρόπο ώστε κάθε σημείο να επισκέπτεται μόνο μία φορά από ακριβώς ένα όχημα εντός ενός συγκεκριμένου χρονικού διαστήματος-όλες οι διαδρομές ξεκινούν και καταλήγουν στην αποθήκη και οι συνολικές απαιτήσεις όλων των σημείων μιας συγκεκριμένης διαδρομής δεν πρέπει να υπερβαίνουν τη χωρητικότητα του οχήματος. Σε αυτή την εργασία, [23] ασχολούνται με ένα πρόβλημα δρομολόγησης οχημάτων συλλογής απορριμμάτων σε πραγματικές συνθήκες με χρονικά παράθυρα (VRPTW), λαμβάνοντας υπόψη τα πολλαπλά ταξίδια απόρριψης και τα μεσημεριανά διαλείμματα των οδηγών. Οι Pisinger και Ropke [24], παρουσίασαν μια ενοποιημένη ευρετική που είναι σε θέση να επιλύσει πέντε διαφορετικές παραλλαγές του προβλήματος δρομολόγησης οχημάτων: το πρόβλημα δρομολόγησης οχημάτων με χρονικά παράθυρα (VRPTW), το πρόβλημα δρομολόγησης οχημάτων με χωρητικότητα (CVRP), το πρόβλημα δρομολόγησης οχημάτων με πολλές αποθήκες (MDVRP), το πρόβλημα δρομολόγησης οχημάτων που εξαρτάται από την τοποθεσία (SDVRP) και το ανοιχτό πρόβλημα δρομολόγησης οχημάτων (OVRP). Οι Golden Raghavan και Wasil [25] μελέτησαν το πρόβλημα δρομολόγησης οχημάτων : τελευταίες εξελίξεις και νέες προκλήσεις. Χρήση γενετικού αλγορίθμου για την επίλυση του γενικευμένου προβλήματος προσανατολισμού. Chvatal-Gomory Rank-1 Cuts που χρησιμοποιούνται σε μια αποσύνθεση Dantzig-Wolfe του προβλήματος δρομολόγησης οχημάτων με χρονικά παράθυρα. Τα προβλήματα δρομολόγησης, όπως το πρόβλημα του περιοδεύοντος πωλητή και το πρόβλημα δρομολόγησης οχημάτων, μελετώνται ευρέως τόσο λόγω της κλασικής ακαδημαϊκής ελκυστικότητάς τους όσο και λόγω των πολυάριθμων εφαρμογών τους στην πραγματική ζωή.



Σχήμα 2.2.8: Παραλλαγές του προβλήματος δρομολόγησης οχημάτων

2.3 Γεωχωρικά Δεδομένα και Συναρτήσεις

2.3.1 The geometry_columns table - OGC simple feature

Ο ρόλος και η σημασία των χωρικών δεδομένων

Η διαχείριση χωρικών πληροφοριών καθίσταται κρίσιμη για εφαρμογές με επίγνωση του πλαισίου, όπως η παρακολούθηση της θέσης ή της έκτασης χρηστών, αισθητήρων και κτιρίων. Η βιβλιοθήκη που παρουσιάζεται από [26] υλοποιεί την προδιαγραφή απλών χαρακτηριστικών του OGC, επιτρέποντας τη διαλειτουργικότητα μεταξύ διαφορετικών εφαρμογών και συστημάτων. Περαιτέρω, οι Web Feature Services (WFS) και Web Map Services (WMS) διερευνώνται από [27] για την ανταλλαγή γεωχωρικών δεδομένων σε πραγματικό χρόνο, ενώ οι [28] βελτιώνουν την απόδοση χωρικών πράξεων μέσω αλγορίθμων. Τέλος, οι [29] τονίζουν τον ρόλο των προτύπων OGC στην ανάπτυξη γεωχωρικών υπηρεσιών.

Τεχνικές και πρότυπα στη χωρική ανάλυση

Η χρήση του Dimensionally Extended Nine-Intersection Model (DE-9IM) ή Clementini-Matrix, που καθορίζεται στην προδιαγραφή OGC "Simple Features for SQL", επιτρέπει τον υπολογισμό χωρικών σχέσεων μεταξύ γεωμετριών [30]. Προσπάθειες για την αποφυγή της δημιουργίας τοπολογίας και τη βελτίωση της αποτελεσματικότητας περιλαμβάνουν την ανάπτυξη λειτουργιών χωρικής ανάλυσης με βάση τις προδιαγραφές OGC [31]. Η εισαγωγή της μεθόδου ευρετηρίασης διπλών σπόρων (DWSI) βελτιώνει την αποτελεσματικότητα των αλγορίθμων επικάλυψης διανυσματικών δεδομένων [32]. Η ανάγκη για τυποποίηση στην ανταλλαγή και ενσωμάτωση γεωχωρικών δεδομένων οδηγεί στην ανάπτυξη του προτύπου OGC/ISO Simple Feature Access (SFA), ενισχύοντας την αυτοματοποιημένη συλλογιστική με γεωμετρικά δεδομένα [33].

Για τα παρακάτω χρησιμοποιήθηκαν οι πηγές **postgisDocs** , [34]

2.3.2 Σύστημα χωρικής αναφοράς (Spatial Reference System) - SRS

Ένα σύστημα χωρικής αναφοράς (SRS) είναι ένα πλαίσιο που χρησιμοποιείται για την ακριβή μέτρηση των θέσεων στην επιφάνεια της Γης ως συντεταγμένες. Για παράδειγμα τα δύο γνωστά SRS είναι

- WGS84 lot lat με SRS ID EPSG:4326 - Παγκόσμιο σύστημα γεωδαισίας
- Web Mercator (EPSG:3857) - Σύστημα προβολής για τον χάρτη του Google

Το EPSG είναι συντομογραφία για το [European Petroleum Survey Group](#).

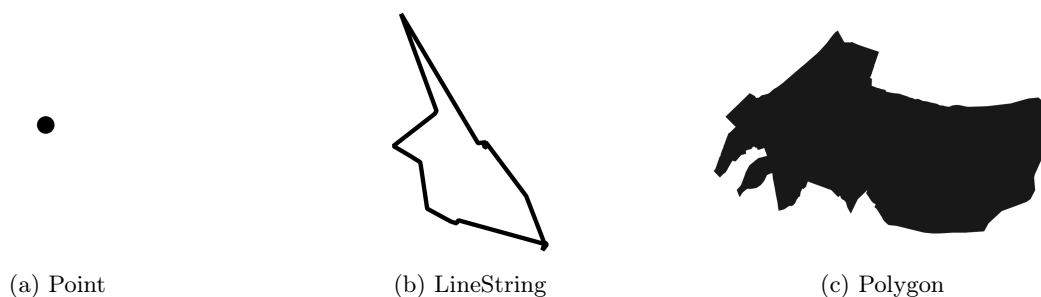
Ένα SRS περιλαμβάνει κάτι περισσότερο από τον τρόπο προσδιορισμού ενός συγκεκριμένου σημείου στη Γη. Περιλαμβάνει ένα ολοκληρωμένο πλαίσιο που αποτελείται από

- την επιλογή ενός ελλειψοειδούς αναφοράς
- το σημείο αναφοράς που συνδέει αυτό το ελλειψοειδές με πραγματικές θέσεις στη Γη
- το σύστημα αναφοράς συντεταγμένων (CRS) που χρησιμοποιείται για τη χαρτογράφηση αυτών των σημείων.

Το SRS είναι μια ευρύτερη έννοια που διασφαλίζει ότι δεδομένα από διαφορετικές πηγές μπορούν να ενσωματωθούν και να χρησιμοποιηθούν μαζί με ακρίβεια στο ίδιο περιβάλλον GIS[34], [36].

Σύστημα αναφοράς συντεταγμένων - CRS

Ένα σύστημα αναφοράς συντεταγμένων (CRS) αποτελεί συστατικό στοιχείο του SRS. Καθορίζει τον τρόπο προσδιορισμού τα σημεία στο επιλεγμένο ελλειψοειδές αναφοράς χρησιμοποιώντας ένα σύνολο συντεταγμένων. Ο πιο δημοφιλής τύπος CRS για χρήση σε ένα ελλειψοειδές αναφοράς είναι το γεωγραφικό σύστημα συντεταγμένων, επίσης γνωστό ως γεωδαιτικό σύστημα συντεταγμένων ή απλά γεωγραφικό μήκος και πλάτος. Αυτό το σύστημα εντοπίζει σημεία στο ελλειψοειδές σχεδιάζοντας γραμμές γεωγραφικού μήκους (μεσημβρινούς) από τον ένα πόλο στον άλλο και γραμμές γεωγραφικού πλάτους



Σχήμα 2.3.1: Οι τρεις βασικοί τύποι γεωχωρικών δεδομένων : point, linestring, polygon.

(παράλληλους) γύρω από το ελλειψοειδές. Αυτή η έννοια της χρήσης ενός συστήματος συντεταγμένων δεν ισχύει μόνο για την υδρόγειο σφαίρα αλλά και για οποιοδήποτε ελλειψοειδές σχήμα, αποδεικνύοντας τον θεμελιώδη ρόλο του στο πλαίσιο ενός SRS [34], [36].

Άλλα SRS συστήματα θα βρείτε [εδώ](#).

2.3.3 Γεωχωρικές Βάσεις δεδομένων

Οι γεωχωρικές βάσεις δεδομένων επιτρέπουν την αποθήκευση γεωμετρικών δεδομένων, που συνήθως σχετίζονται με τη γεωγραφία, σε συνηθισμένους πίνακες βάσεων δεδομένων και προσφέρουν μοναδικούς τύπους δεδομένων ειδικά για γεωμετρικά αντικείμενα. Προσφέρουν μοναδικές λειτουργίες και ευρετήρια που επιτρέπουν τον χειρισμό των δεδομένων και την υποβολή ερωτημάτων μέσω της χρήσης κάτι παρόμοιο με τη Structured Query Language (SQL). Χρησιμοποιείται συχνά μόνο για την αποθήκευση χωρικών δεδομένων, μια χωρική βάση δεδομένων είναι ικανή για πολλά περισσότερα [34].

Η πλέον εδραιωμένη γεωχωρική βάση δεδομένων είναι η PostgreSQL με την επέκταση PostGIS. Ο πυρήνας της λειτουργικότητας της PostGIS βασίζεται στην PostgreSQL, η οποία υποστηρίζει την γεωχωρική δημιουργία index με το GiST (Generalized Search Tree) και έναν αποτελεσματικό query planner ειδικά φτιαγμένο για τα γεωχωρικά δεδομένα. Η απόφαση να αναπτύξει το PostGIS πάνω στην PostgreSQL υπογραμμίζει την ανθεκτικότητα και την ευελιξία της PostgreSQL ως πλατφόρμα ORDBMS (object-relational database management system), διαχωρίζοντάς την σημαντικά από άλλα συστήματα βάσεων δεδομένων ανοικτού κώδικα.

2.3.4 Γεωχωρικά δεδομένα

Τρεις θεμελιώδεις τύποι γεωχωρικών δεδομένων μπορούν να χρησιμοποιηθούν για την αναπαράσταση οποιοδήποτε δυο διαστάσεων χωρικών αντικειμένων: τα σημεία (Points), οι γραμμές (Linestrings) και τα πολύγωνα (Polygons).

Well-Known Text (WKT)

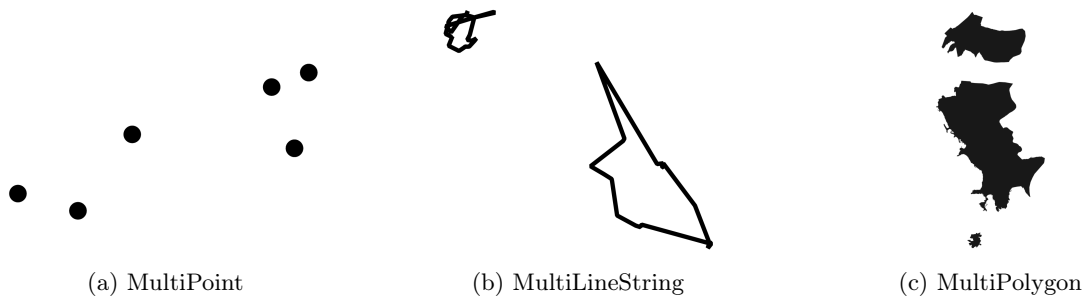
Well-known text representation είναι ένα πρότυπο της Open Geospatial Consortium (OGC) για την αναπαράσταση γεωμετριών ως κείμενο.

Οι λειτουργίες `ST_AsText` και `ST_GeomFromText` είναι λειτουργίες του OGC που συναντώνται συνήθως σε πολλές χωρικές βάσεις δεδομένων συμβατές με το OGC, οι οποίες μετατρέπουν εμπρός και πίσω μεταξύ των αρχείων μιας βάσης δεδομένων εγγενή δυαδική μορφή σε μορφή εμφάνισης κειμένου.

Multi geometries

Τα multiGeometries όπως τα MultiPoint, MultiLineString και MultiPolygon είναι συλλογές από τα αντίστοιχα σημεία, γραμμές και πολύγωνα.

- MultiPoint: Μια συλλογή από points και τίποτε άλλο.



Σχήμα 2.3.2: Οι τρεις τύποι multigeometries.

- MultiLineString: Μια συλλογή από linestrings και τίποτε άλλο.
- MultiPolygon: Μια συλλογή από polygons και τίποτε άλλο.

Geometry Collection

Το GeometryCollection είναι ένας τύπος δεδομένων στο PostGIS που μπορεί να περιέχει ετερογενείς γεωμετρικές. Σε αντίθεση με τα multigeometries όπου οι γεωμετρικές που τις αποτελούν πρέπει να είναι του ίδιου τύπου, ο τύπος δεδομένων GeometryCollection μπορεί να περιλαμβάνει σημεία, γραμμές (linestrings), πολύγωνα και τα αντίστοιχα τους σε μορφή συλλογής. Μπορεί ακόμη και να περιέχει άλλα GeometryCollections. Εν ολίγοις, μπορείτε να συμπεριλάβετε κάθε τύπο γεωμετρίας που είναι γνωστός στο PostGIS μέσα σε ένα GeometryCollection.

2.3.5 Γεωχωρικές Πράξεις

Γιατί σε όλα ST_ ;

Παρατηρούμε ότι όλες οι γεωχωρικές πράξεις στο PostGIS αρχίζουν με το ST_ . Αυτό γίνεται μόνο στην PostGIS και ισχύει για μοναδικές πράξεις που υπάρχουν μόνο στην PostGIS αλλά και για πράξεις που υπάρχουν στη SQL. Το ST είναι μια συντομογραφία για το "spatial temporal".

ST_MakePolygon

Με ένα κλειστό linestring που χρησιμεύει ως αναπαράσταση του εξωτερικού δακτυλίου, η ST_MakePolygon κατασκευάζει ένα πολύγωνο. Έχει την επιλογή να λάβει ως δεύτερο όρισμα έναν πίνακα κλειστών γραμμοσειρών για εσωτερικούς δακτυλίους. Το ST_MakePolygon δεν επικυρώνει τα linestrings εισόδου. Αυτό σημαίνει ότι τα ανοιχτά linestrings ή linestrings που δεν μπορούν να δημιουργήσουν πολύγωνα, δημιουργούν ένα λανθασμένο ή μάλλον λάθος πολύγωνο, όπως ένα με τρύπες εκτός του εξωτερικού δακτυλίου ή έναν εσωτερικό δακτύλιο που δεν περιέχεται εξ ολοκλήρου από τον εξωτερικό δακτύλιο. Υπάρχει όφελος στην ταχύτητα αφού η επικύρωση του αρχικού linestring απουσιάζει εντελώς.

ST_Union

Το st_union εφαρμόζεται σαν aggregation function, λειτουργεί σαν το st_union, επιστρέφοντας την ένωση όλων των γεωμετριών σε ένα multi-geometry πεδίο.

ST_ExteriorRing

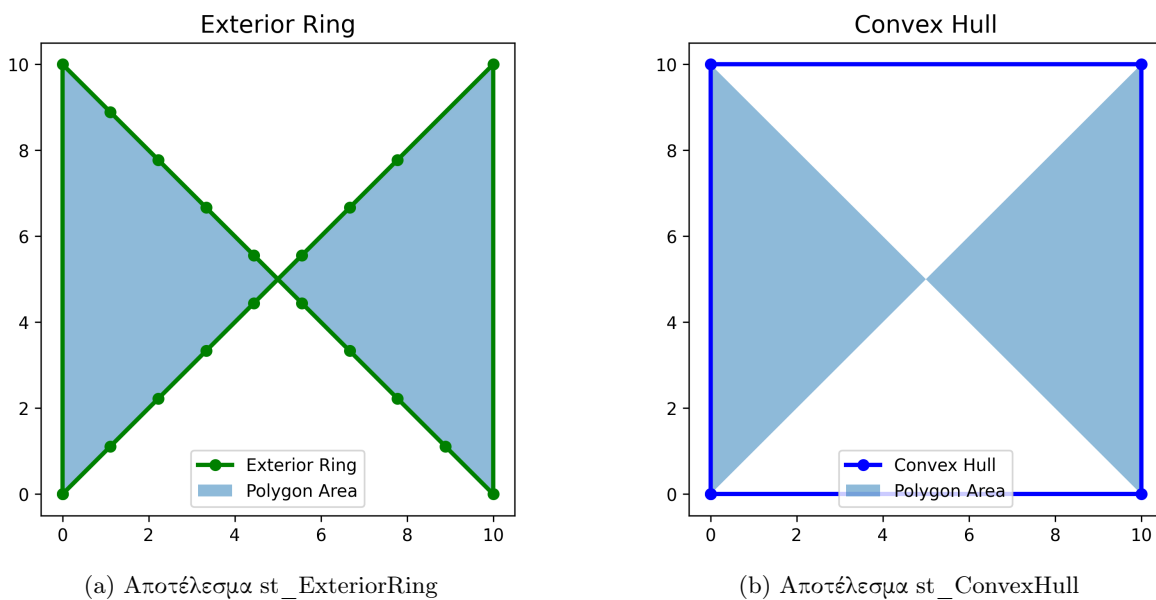
Η συνάρτηση ST_ExteriorRing επιστρέφει τον εξωτερικό δακτύλιο ενός πολυγώνου.

ST_ConvexHull

Έχει σχεδιαστεί για τη δημιουργία του ελάχιστου κυρτού πολυγώνου που περικλείει ένα σύνολο από σημεία, γραμμές ή πολύγωνα σε ένα χωρικό σύνολο δεδομένων.



Σχήμα 2.3.3: Παράδειγμα εξωτερικού δακτυλίου (με πορτοκαλί χρώμα) Εικόνα από [34].



Σχήμα 2.3.4: Παράδειγμα διαφορών `st_exteriorRing` και `st_convexHull`.

ST_Multi

Η συνάρτηση ST_Multi χρησιμοποιείται συχνά στο PostGIS για τη μετατροπή σημείων, γραμμοσειρών και πολυγώνων στις αντίστοιχες μορφές πολλαπλών στοιχείων, ανεξάρτητα από το αν αποτελούνται από μία μόνο γεωμετρία. Εάν η γεωμετρία υπάρχει ήδη ως παραλλαγή πολλαπλών στοιχείων, θα παραμείνει ως έχει. Ο πρωταρχικός σκοπός αυτής της συνάρτησης είναι η διατήρηση της ομοιομορφίας σε μια στήλη βάσης δεδομένων, διασφαλίζοντας ότι κάθε γεωμετρία σε αυτήν έχει τον ίδιο τύπο, διατηρώντας έτσι τη συνοχή.

ST_Intersects και ST_Intersection

Η έννοια της τομής αναφέρεται σε διάφορους τρόπους με τους οποίους οι γεωμετρίες μπορεί να επικαλύπτονται ή να έρχονται σε επαφή μεταξύ τους.

Στο PostGIS, υπάρχουν δύο συναρτήσεις που έχουν σχεδιαστεί για τον χειρισμό των τομών. Η πρώτη, ST_Intersects, αξιολογεί δύο γεωμετρίες και επιστρέφει true αν έχουν επικαλυπτόμενα τμήματα. Η δεύτερη συνάρτηση, ST_Intersection, παράγει μια νέα γεωμετρία που περικλείει την περιοχή που μοιράζονται οι δύο γεωμετρίες εισόδου. Εάν δεν υπάρχει επικάλυψη μεταξύ των γεωμετριών, το αποτέλεσμα αυτής της συνάρτησης θα είναι μια κενή γεωμετρία.

ST_Contains και ST_Within

Οι όροι "Contains" και "Within" περιγράφουν μια αμφίδρομη σχέση μεταξύ γεωμετριών. Εάν η γεωμετρία A βρίσκεται πλήρως εντός της γεωμετρίας B, τότε λέμε ότι η B περιέχει (contains) την A και αντίστροφα, η A βρίσκεται εντός (within) της B. Αυτές οι σχεσιακές έννοιες ορίζονται στο PostGIS μέσω των συναρτήσεων ST_Contains και ST_Within αντίστοιχα. Μια αξιοσημείωτη απαίτηση για να "περιέχει" μια γεωμετρία A τη γεωμετρία B είναι ότι η επικάλυψη του ορίου της A με τη B δεν αποτελεί πλήρως τη B. Αυτό σημαίνει ότι η B δεν πρέπει να τοποθετείται εξ ολοκλήρου στο όριο της A. Ενώ μια γεωμετρία δεν περιλαμβάνει το σύνορό της στη σχέση περιέλευσης, περιλαμβάνει εγγενώς τον εαυτό της.

ST_Dump

Η ST_Dump περιγράφεται ως μια συνάρτηση χρήσιμη για την αποσύνθεση γεωμετριών πολλαπλών και συλλογών στις επιμέρους γεωμετρίες που τις αποτελούν. Σε αντίθεση με την ST_GeometryN, η οποία διερευνά μόνο ένα επίπεδο γεωμετρίας, η ST_Dump εμβαθύνει αναδρομικά στις πολυ- και συλλεκτικές γεωμετρίες για να τις αποδομήσει διεξοδικά. Το ST_Dump δεν επιστρέφει άμεσα μια γεωμετρία-αντίθετα, παράγει σειρές αντικειμένων geometry_dump. Αυτός ο προσαρμοσμένος τύπος, αναπόσπαστο στοιχείο του PostGIS, αποτελείται από δύο στοιχεία: μια διαδρομή και ένα geom. Το path είναι ένας μονοδιάστατος πίνακας που επισημαίνει το βάθος στο οποίο εντοπίστηκε η γεωμετρία, βοηθώντας στην πιθανή ανακατασκευή της αρχικής γεωμετρίας. Το geom περιέχει την πραγματική αποσυντεθειμένη γεωμετρία για την καθορισμένη διαδρομή. Η λειτουργία αυτή επιτρέπει τη λεπτή αποσυναρμολόγηση πολύπλοκων γεωμετριών, συμπεριλαμβανομένων εκείνων με καμπύλα σχήματα, παρέχοντας λεπτομερή εικόνα της χωρικής δομής.

Άλλες γεωχωρικές πράξεις

Μερικά ακόμα παραδείγματα είναι τα εξής:

- **ST_GeomFromText:** Μετατρέπει το κείμενο σε γεωχωρικό αντικείμενο.
- **ST_GeomFromEWKT:** Μετατρέπει το κείμενο σε γεωχωρικό αντικείμενο με τον κωδικό αναφοράς.
- **ST_SetSRID:** Ορίζει τον κωδικό αναφοράς του srs των γεωχωρικών δεδομένων.
- **ST_AsText:** Μετατρέπει τα γεωχωρικά δεδομένα σε μορφή κειμένου.
- **ST_Centroid:** Επιστρέφει το κέντρο μάζας του γεωχωρικού αντικειμένου.

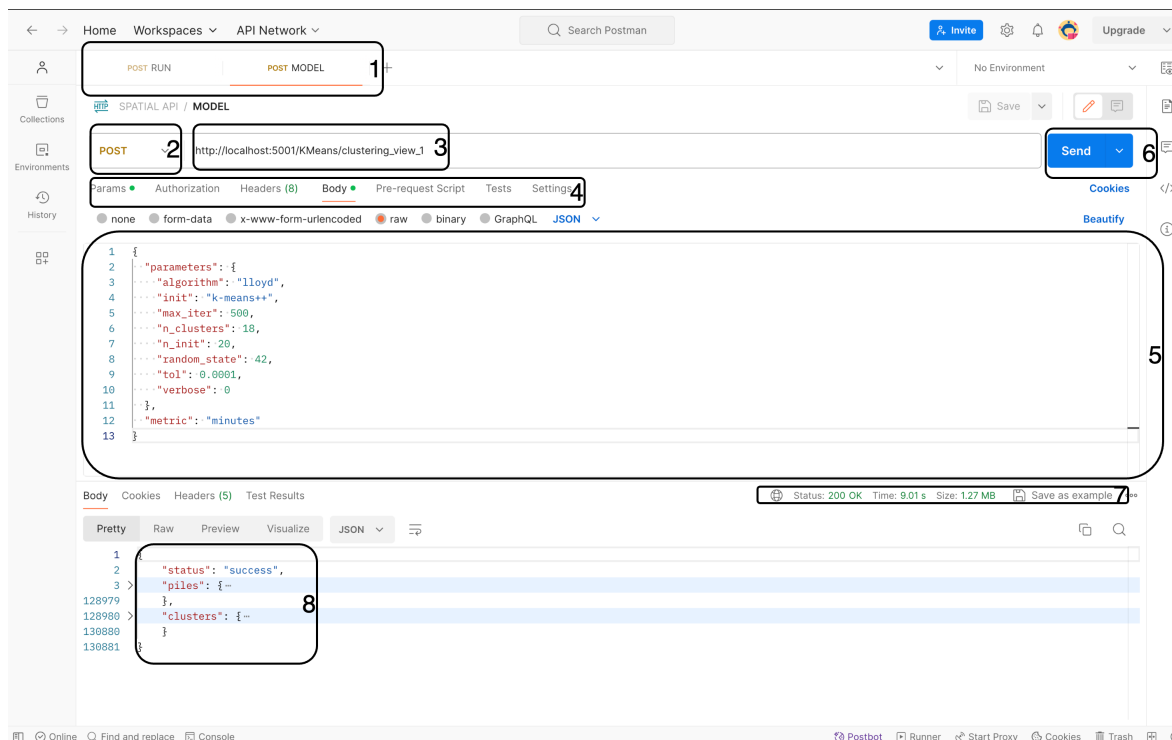
- **ST_Distance**: Υπολογίζει την απόσταση μεταξύ δύο γεωχωρικών αντικειμένων.
- **ST_Collect**: Συλλέγει διαφορετική ή όμοια γεωχωρικά αντικείμενα σε μία στήλη geometryCollection [2.3.4](#).

2.4 Tools

Παρακάτω θα αναλυθούν τα εργαλεία που χρησιμοποιήθηκαν κατά την διάρκεια της υλοποίησης της εφαρμογής.

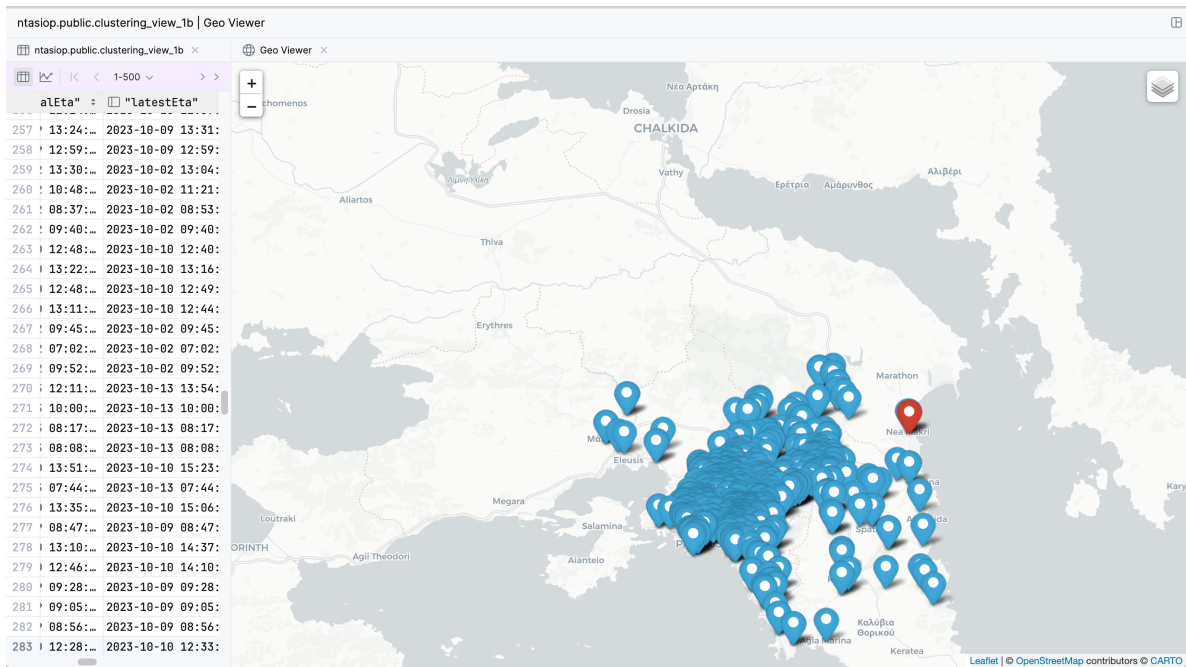
2.4.1 Postman: Εργαλείο Διαχείρισης API

Το Postman αποτελεί ένα εργαλείο που εξυπηρετεί στην ευκολότερη και αποδοτικότερη διαχείριση API αιτημάτων, προσφέροντας δυνατότητες για αυτοματοποίηση, αποθήκευση, σύγκριση, και μορφοποίηση των αποτελεσμάτων τους. Ειδικότερα, στο πλαίσιο της παρούσας εργασίας, χρησιμοποιήθηκε για την αποτελεσματική οργάνωση και διαχείριση των API αιτημάτων.



Σχήμα 2.4.1: Γραφικό Περιβάλλον του Postman

Στο ανωτέρω σχήμα [2.4.1](#), παρουσιάζεται το γραφικό περιβάλλον του Postman. Στο **παραλληλόγραμμο 1**, διακρίνονται τα ενεργά API αιτήματα. Στο **παραλληλόγραμμο 2**, απεικονίζονται οι διαθέσιμες μέθοδοι που υποστηρίζονται από το Postman. Στο **παραλληλόγραμμο 3**, φαίνεται το επιλεγμένο URL για την υποβολή του αιτήματος. Στο **παραλληλόγραμμο 4**, παρουσιάζονται οι υποστηριζόμενοι τύποι παραμέτρων για το συγκεκριμένο αίτημα. Στο **παραλληλόγραμμο 5**, αναφέρονται οι παράμετροι που έχουν επιλεγεί για το αίτημα. Το **παραλληλόγραμμο 6** αντιστοιχεί στο κουμπί υποβολής του αιτήματος. Στο **παραλληλόγραμμο 7**, εμφανίζονται το κατάσταση, ο χρόνος απόκρισης και το μέγεθος της απάντησης του αιτήματος. Τέλος, στο **παραλληλόγραμμο 8**, διατίθεται η απάντηση του αιτήματος.

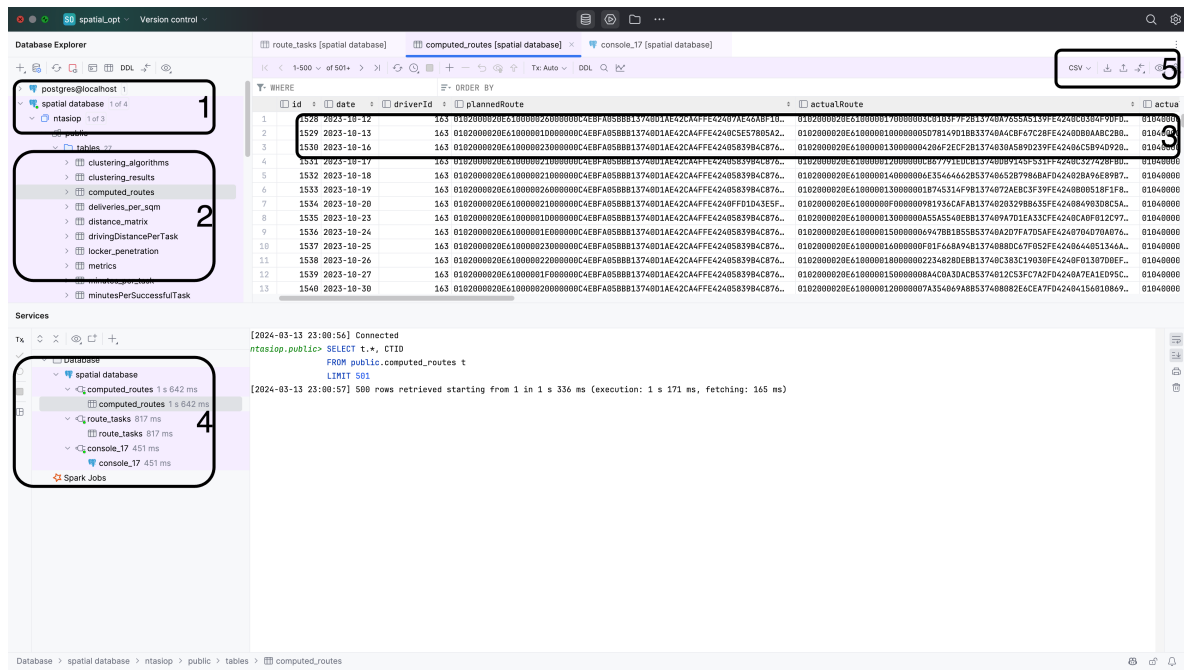


Σχήμα 2.4.2: Datagrip χάρτης γεωχωρικών δεδομένων

2.4.2 Datagrip

Το DataGrip [37] είναι ένα εργαλείο που χρησιμοποιείται για την διαχείριση βάσεων δεδομένων. Υποστηρίζει διάφορα συστήματα βάσεων δεδομένων και ενσωματώνει ένα ευρύ φάσμα εργαλείων για την αποτελεσματική αναζήτηση, επεξεργασία και πλοήγηση σε βάσεις δεδομένων. Η φιλική προς το χρήστη διεπαφή του DataGrip και η έξυπνη βοήθεια κωδικοποίησης εξορθολογίζουν τις λειτουργίες των βάσεων δεδομένων, καθιστώντας το απαραίτητο εργαλείο.

Περιβάλλον χρήστη του Datagrip



Σχήμα 2.4.3: Datagrip GUI

Στο παραπάνω σχήμα 2.4.3 φαίνεται το γραφικό περιβάλλον του Datagrip. Στο αριστερό μέρος της εικόνας, παραλληλόγραμμα 1 φαίνονται οι βάσεις δεδομένων που έχουν συνδεθεί στο Datagrip. Στο παραλληλόγραμμα 2 φαίνονται οι πίνακες της βάσης δεδομένων που έχει επιλεγεί. Στο παραλληλόγραμμα 3 φαίνεται ένα δείγμα των δεδομένων που περιέχει ο πίνακας. Στο παραλληλόγραμμα 4 φαίνονται οι ανοιχτές συνδέσεις με την βάση δεδομένων. Στο παραλληλόγραμμα 5 φαίνονται τα εργαλεία που προσφέρει το Datagrip για την διαχείριση των δεδομένων, εκεί βρίσκεται και η επιλογή για την οπτικοποίηση των γεωχωρικών δεδομένων.

Queries & Visualizations

Το Datagrip προσφέρει την δυνατότητα να εκτελείς queries στην βάση δεδομένων και να οπτικοποιείς τα αποτελέσματα.

The screenshot displays the Datagrip interface with a SQL query in the console and its results visualized on a map. The query is: `select * from clustering_view_1b`. The results are shown in a table with columns: type, state, result, and originalEtm. The map shows a cluster of blue location pins on a map of Greece, with a red pin highlighting a specific location.

1 points to the SQL query in the console.

2 points to the table of query results.

3 points to the map visualization of the results.

4 points to the map controls (layers, zoom, etc.).

type	state	result	originalEtm
customer_delivery	closed	succeeded	2023-10-06 10:17
bulk_merchant_pickup	closed	<null>	2023-10-10 13:10
customer_delivery	closed	succeeded	2023-10-06 10:08
customer_delivery	closed	succeeded	2023-10-06 08:22
customer_delivery	closed	<null>	2023-10-06 10:01
customer_delivery	closed	succeeded	2023-10-10 13:03
customer_delivery	closed	<null>	2023-10-10 07:54
bulk_merchant_pickup	closed	succeeded	2023-10-10 13:00
bulk_merchant_pickup	closed	succeeded	2023-10-10 12:23
bulk_merchant_pickup	closed	succeeded	2023-10-10 13:06
customer_delivery	closed	succeeded	2023-10-10 12:20
customer_delivery	closed	<null>	2023-10-06 07:48
bulk_merchant_pickup	closed	succeeded	2023-10-10 12:26
customer_delivery	closed	<null>	2023-10-06 10:13
customer_delivery	closed	<null>	2023-10-06 07:13
bulk_merchant_pickup	closed	succeeded	2023-10-10 13:11
bulk_merchant_pickup	closed	succeeded	2023-10-10 13:57

Σχήμα 2.4.4: Datagrip Queries & Visualizations

Στο σχήμα 2.4.4 φαίνεται ένα query που εκτελείται στην βάση δεδομένων και τα αποτελέσματα του οπτικοποιούνται σε έναν χάρτη. Στο πάνω μέρος, παραλληλόγραμμα 1 φαίνεται το query, `select * from clustering_view_1b` που εκτελείται. Στα αριστερά, παραλληλόγραμμα 2, φαίνονται τα αποτελέσματα του query με την μορφή πίνακα. Στα δεξιά, παραλληλόγραμμα 3 φαίνεται ο χάρτης που παράγεται από τα αποτελέσματα του query. Ο χάρτης διαδραστικός και ο χρήστης μπορεί να επιλέξει το γεωχωρικό δεδομένο που θέλει να εξετάσει πατώντας πάνω στο σημείο που το αντιπροσωπεύει. Στο παραλληλόγραμμα 4 φαίνονται οι επιλογές που προσφέρει το Datagrip για την αλλαγή του τύπου του χάρτη.

Κεφάλαιο 3

Παρουσίαση του προβλήματος

Σε αυτό το κεφάλαιο παρουσιάζουμε το πρόβλημα που αντιμετωπίσαμε και την δομή που έχουν τα δεδομένα μας. Η ανάλυση ξεκινά με την παρουσίαση των δεδομένων εισόδου, όπου εξετάζεται η φύση, η δομή και η προέλευση των δεδομένων. Στη συνέχεια, αναφερόμαστε στους δείκτες αξιολόγησης, δηλαδή στα κριτήρια με βάση τα οποία θα αξιολογηθεί η αποδοτικότητα και η επιτυχία των προτεινόμενων λύσεων. Επιπλέον, αφιερώνουμε μια ενότητα στην ανάλυση των Ταχυδρομικών Κωδικών, εξετάζοντας τον ρόλο τους στη διαδικασία δρομολόγησης και διανομής, και τη συνεισφορά τους στην αποδοτικότητα του συνολικού συστήματος. Τέλος, προχωράμε σε μια λεπτομερή ανάλυση των SQL Views και του ER Diagram, παρέχοντας ένα σαφές οπτικό αποτύπωμα της βάσης δεδομένων και των σχέσεων μεταξύ των διάφορων οντοτήτων.

3.1 Δεδομένα Εισόδου

Τα δεδομένα εισόδου είναι κατά κύριο λόγο δρομολόγια οδηγών με στάσεις παράδοσης και παραλαβής. Παράλληλα, διαθέσιμη έγινε και η υφιστάμενη λύση συσταδοποίησης που πρέπει να βελτιστοποιήσουμε. Επιπλέον, διαθέτουμε και τα γεωχωρικά όρια των ταχυδρομικών οδηγών της Ελλάδας. Ακολουθεί λεπτομερής παρουσίαση της μορφής των δεδομένων.

3.1.1 Δρομολόγια - Σημεία Παράδοσης/Παραλαβής

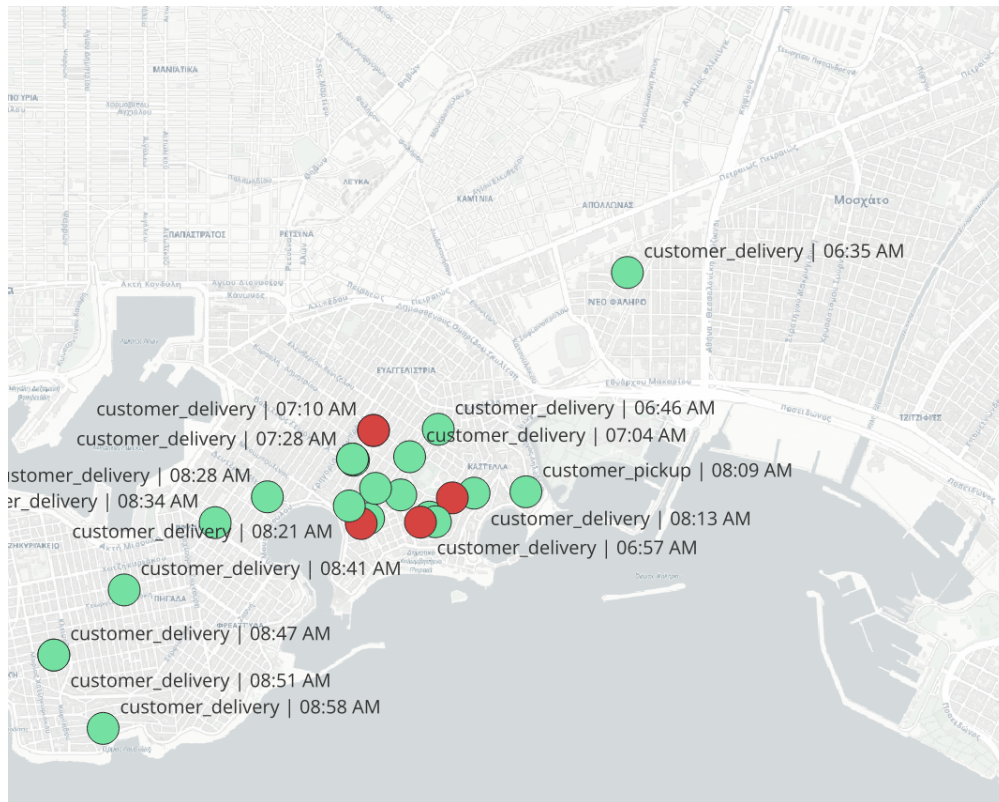
Κατα κύρια βάση, τα δρομολόγια παράγονται την προηγούμενη μέρα και αφορούν την επόμενη. Ωστόσο, κατά την διάρκεια μιας τυπικής διαδρομής, όπως αναφέραμε και στην εισαγωγή 1.2, υπάρχει περίπτωση να υπάρξουν αλλαγές στο δρομολόγιο, λόγω μη αναμενόμενων περιστατικών, όπως κίνηση, καιρικές συνθήκες, κλπ. Επομένως, μερικές στάσεις μπορεί να μην ολοκληρωθούν ποτέ και ο οδηγός να πρέπει να επιστρέψει στην αποθήκη πριν ολοκληρώσει όλες τις αναμενόμενες στάσεις. Λόγω, των ίδιων των δεδομένων, υπάρχει περίπτωση κάποιοι οδηγοί να πάρουν δέματα, άρα και στάσεις από άλλους οδηγούς, συναντώντας τους σε ένα σημείο κατά την διάρκεια του δρομολογίου τους.

Παρακάτω παραθέτουμε ένα παράδειγμα των δεδομένων 3.1.1. Πρόκειται για στάσεις οδηγών για παράδοσεις ή παραλαβές δεμάτων. Στο παράδειγμα, με κόκκινο χρώμα εμφανίζονται οι στάσεις που απέτυχαν, ενώ με πράσινο χρώμα εμφανίζονται οι στάσεις που ολοκληρώθηκαν επιτυχώς. Επιπλέον, παρατηρούμε ότι έχουμε πληροφορίες για την ώρα παράδοσης ή παραλαβής και τον τύπο της στάσης.

Τα δεδομένα δρομολογίων οδηγών αφορούν μια περίοδο από Οκτώβριο μέχρι Δεκέμβριο του 2023, με κάθε δρομολόγιο να διαρκεί από 5 έως 8 ώρες και να έχει κατά μέσο όρο 50 στάσεις.

Μορφή των δεδομένων

Παρακάτω παρατίθεται ένας πίνακας που περιγράφει τη μορφή των δεδομένων δρομολογίων.



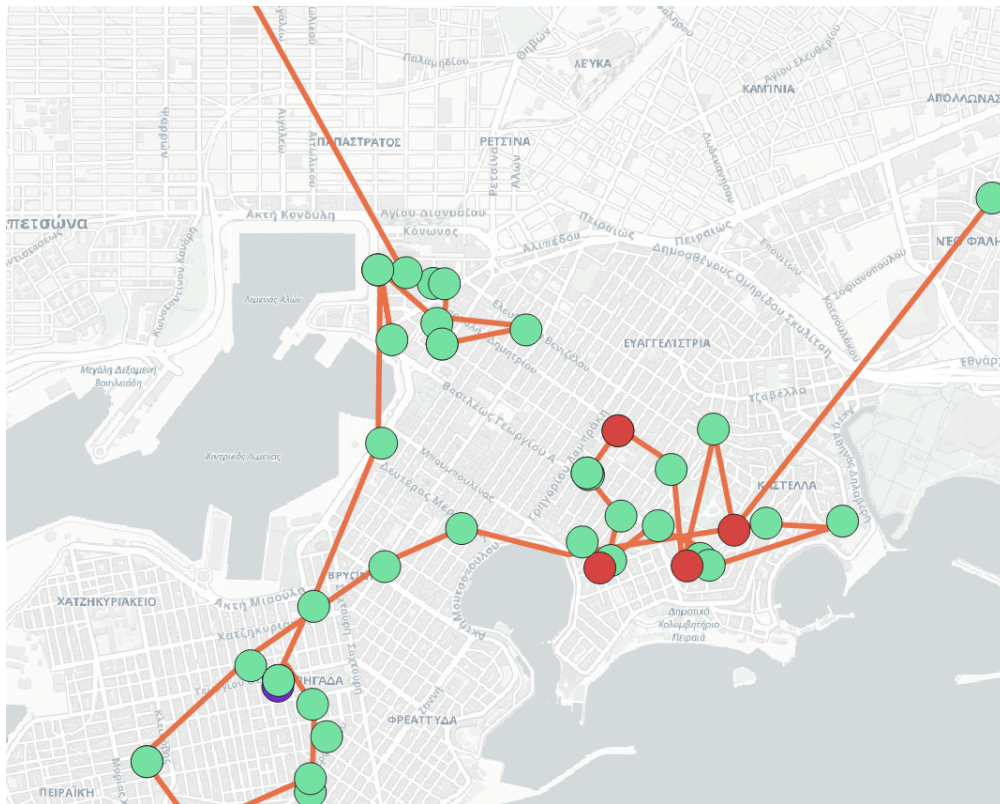
Σχήμα 3.1.1: Δεδομένα στάσεων.

Πεδίο	Τύπος	Περιγραφή
date	date	Ημερομηνία δρομολογίου
driverId	integer	Αριθμός αναγνώρισης οδηγού
type	enum	Προκαθορισμένος τύπος παράδοσης ή παραλαβής για την στάση
result	enum	Αποτέλεσμα της παράδοσης/παραλαβής (succeeded ή failed)
location	geometry(Point)	Γεωγραφικό σημείο της στάσης
latestEta	timestamp	Η ώρα που έγινε η παράδοση/παραλαβή
relatedOrderCount	integer	Ο αριθμός των δεμάτων που παραδόθηκαν/παραλήθηκαν

Πίνακας 3.1: Πεδία του πίνακα δρομολογίων.

Computed routes

Από τα παραπάνω δεδομένα μπορούμε να κατασκευάσουμε ένα παραγόμενο δρομολόγιο για συγκεκριμένο ζευγάρι οδηγού και ημερομηνία. Συγκεκριμένα, με χρήση των σημείων των στάσεων και τον χρόνο ολοκλήρωσης μίας στάσης μπορούμε να κατασκευάσουμε ένα LineString που αντιπροσωπεύει την διαδρομή του οδηγού. Ακολουθεί ένα παράδειγμα παραγόμενου δρομολογίου [3.1.2](#).



Σχήμα 3.1.2: Παράδειγμα παραγόμενου δρομολογίου.

3.1.2 Γεωγραφικά όρια ταχυδρομικών κωδικών

Τα γεωγραφικά όρια των ταχυδρομικών κωδικών είναι κρίσιμα για την επίλυση του προβλήματος και παρέχονται σε μορφή shapefile. Αυτά τα όρια μπορούν να χρησιμοποιηθούν για την δημιουργία συστάδων περιοχών, όπως αναφέραμε στην εισαγωγή 1.3.

Μορφή των δεδομένων

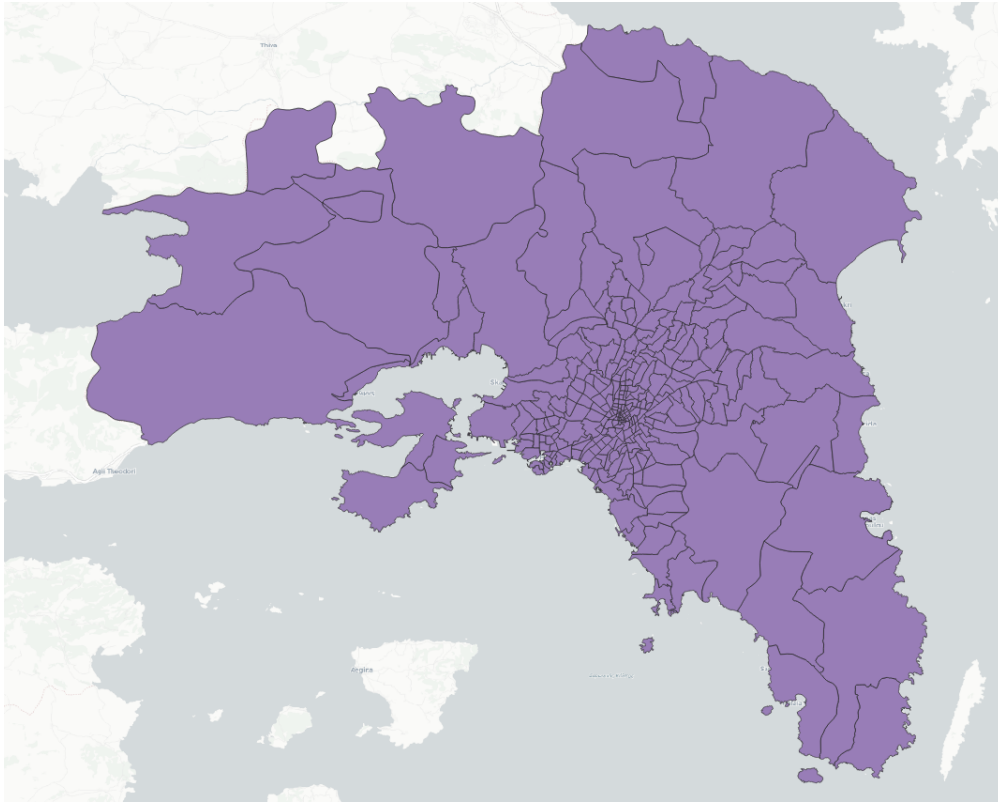
Δίνεται πίνακας που περιγράφει τη μορφή των δεδομένων των ταχυδρομικών κωδικών.

Πεδίο	Τύπος	Περιγραφή
id	integer	Μοναδικός αριθμός αναγνώρισης
Shape	geometry(MultiPolygon)	Γεωγραφικό σχήμα
prefecture	string	Νομός
municipal	string	Δήμος
pile	string	Συστάδα στην οποία ανήκει ο ΤΚ.

Πίνακας 3.2: Πεδία του shapefile των ταχυδρομικών κωδικών.

3.1.3 Υπάρχουσα λύση του προβλήματος - Συστάδες

Απο τον προηγούμενο πίνακα, παράγεται επίσης η υπάρχουσα λύση στο πρόβλημα συσταδοποίησης, η οποία αποτελείται από συστάδες περιοχών στην Αττική, χωρίς να έχει πραγματοποιηθεί βελτιστοποίηση για την αποτελεσματική διαίρεση των περιοχών. Αυτά τα clusters αποτελούν και η βάση σύγκρισης για τα νέα αποτελέσματα.



Σχήμα 3.1.3: Παράδειγμα των γεωγραφικών ορίων των ταχυδρομικών κωδικών.

Για την κατασκευή τους χρησιμοποιήσαμε τον παρακάτω αλγόριθμο:

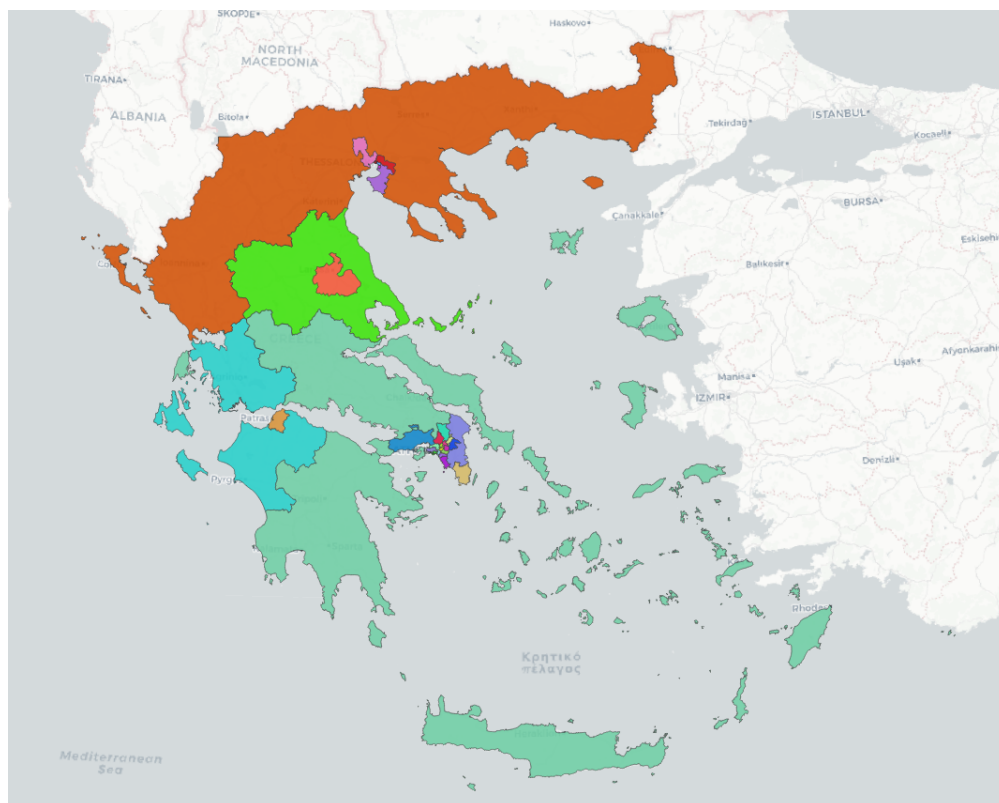
Algorithm 5 Generate Clusters From Zips

```

1: Zipcodes ← Load all rows from the zipcodes table
2: Initialize Data as an empty list
3: for each pile in Zipcodes grouped by pile do
4:   ShapeUnion ← ST_Union of all shapes in the current group
5:   MultiGeometry ← Convert ShapeUnion into a multi-geometry
6:   DumpedGeometry ← ST_Dump of MultiGeometry
7:   ExteriorRing ← ST_ExteriorRing of DumpedGeometry
8:   Polygon ← ST_MakePolygon of ExteriorRing
9:   MultiPolygon ← ST_Multi of Polygon
10:  ZipcodesList ← Concatenate all zip codes in the current group separated by ','
11:  Add {id : row_number, shape1 : MultiPolygon, name : pile, zipcodes : ZipcodesList} to Data
12: end for
13: Initialize FinalResults as an empty list
14: for each name in Data grouped by name do
15:   ShapeUnion ← ST_Union of all shape1 in the current group
16:   MinId ← Minimum id in the current group
17:   MinZipcodes ← Minimum zipcodes in the current group
18:   Add {name : name, shape : ShapeUnion, id : MinId, zipcodelist : MinZipcodes} to
     FinalResults
19: end for
20: return FinalResults

```

Παράδειγμα των συστάδων παρουσιάζεται στο σχήμα 3.1.4.



Σχήμα 3.1.4: Παράδειγμα συστάδων.

Μορφή των δεδομένων

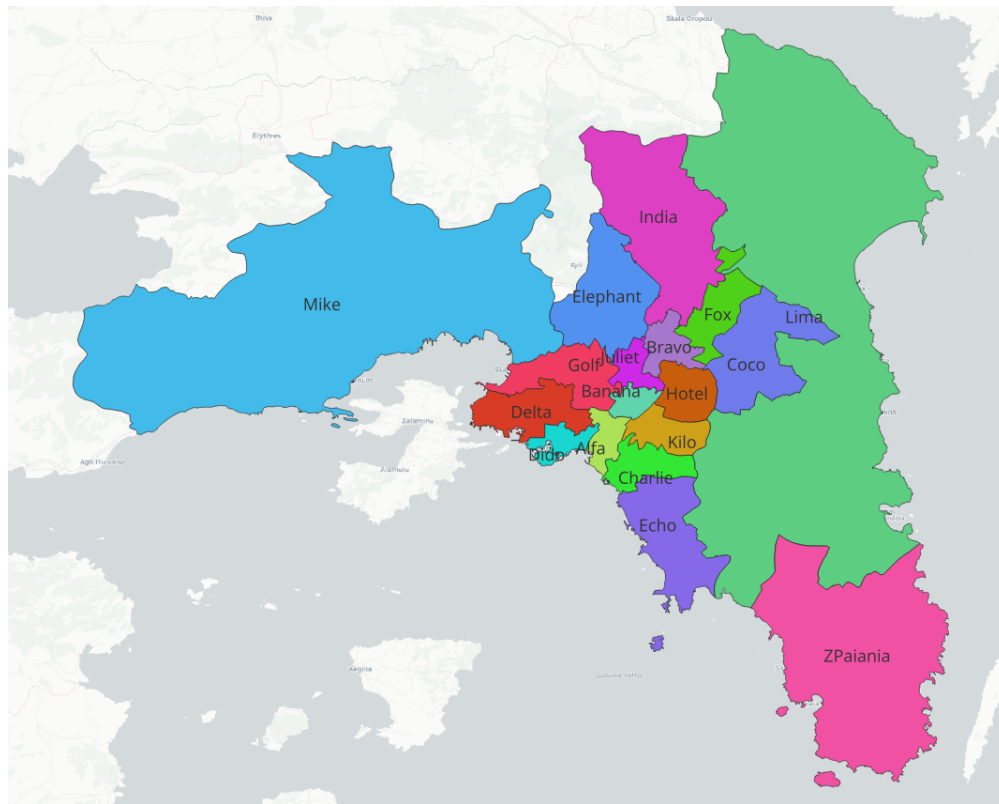
Ακολουθεί πίνακας που περιγράφει τη μορφή των δεδομένων της υπάρχουσας λύσης.

Πεδίο	Τύπος	Περιγραφή
id	integer	Μοναδικός αριθμός αναγνώρισης
name	string	Όνομα συστάδας
shape	geometry(MultiPolygon)	Γεωγραφικό σχήμα
zipCodeList	array	Λίστα με τους ταχυδρομικούς κωδικούς της συστάδας

Πίνακας 3.3: Πεδία του shapefile της υπάρχουσας λύσης.

Παράδειγμα των συστάδων σε Αττική

Ένα παράδειγμα, μόνο των συστάδων της Αττικής, παρουσιάζεται στο σχήμα 3.1.5.



Σχήμα 3.1.5: Παράδειγμα των συστάδων σε Αττική.

3.2 Δείκτες Αξιολόγησης

Στην παρούσα ενότητα παρουσιάζονται τα κριτήρια και οι μεθοδολογίες που χρησιμοποιούνται για την αξιολόγηση της αποδοτικότητας και της αποτελεσματικότητας των δρομολογίων. Η πολυπλοκότητα της αξιολόγησης μιας διαδρομής μεταφοράς απορρέει από την πολύπλευρη φύση των εμπλεκόμενων κριτηρίων, τα οποία μπορούν να επηρεάσουν σημαντικά το αποτέλεσμα της διαδικασίας αξιολόγησης. Το παρόν έγγραφο περιγράφει δύο βασικές προσεγγίσεις για την αξιολόγηση δρομολογίων: την αξιολόγηση με βάση τον χρόνο και την αξιολόγηση με βάση την απόσταση. Οι μέθοδοι αυτές είναι από τις πλέον διαδεδομένες και κρίσιμες στο πλαίσιο του σχεδιασμού και της ανάλυσης των μεταφορών. Η σημασία τους, οι μεθοδολογίες και οι επιπτώσεις τους συζητούνται λεπτομερώς για την παροχή μιας ολοκληρωμένης κατανόησης της αξιολόγησης δρομολογίων.

3.2.1 Αξιολόγηση Δρομολογίων

Η αξιολόγηση μιας διαδρομής μεταφοράς είναι μια πολύπλοκη διαδικασία που απαιτεί την εις βάθος κατανόηση των διαφόρων παραγόντων που μπορούν να επηρεάσουν το αποτέλεσμα της αξιολόγησης. Η αξιολόγηση εξαρτάται από τα καθορισμένα κριτήρια και τους στόχους της αξιολόγησης. Στην παρούσα έρευνα, διερευνούμε δύο διαφορετικές μεθοδολογίες για την αξιολόγηση διαδρομών: αξιολογήσεις με βάση τον χρόνο και αξιολογήσεις με βάση την απόσταση. Και οι δύο προσεγγίσεις είναι κομβικές στις μελέτες μεταφορών και προσφέρουν μοναδικές γνώσεις για την επιλογή και τη βελτιστοποίηση διαδρομών.

Αξιολόγηση με βάση τον χρόνο

Η εκτίμηση της χρονικής αποδοτικότητας μιας διαδρομής είναι μια κρίσιμη μετρική στην αξιολόγηση διαδρομών. Η διάρκεια που απαιτείται για τη διέλευση μιας διαδρομής αποτελεί πρωταρχικό παράγοντα που επηρεάζει την επιλογή διαδρομής τόσο για τα άτομα όσο και για τις εμπορευματικές μεταφορές. Η

διάρκεια αυτή επηρεάζεται από διάφορες μεταβλητές, συμπεριλαμβανομένων των συνθηκών κυκλοφορίας, της ώρας της ημέρας, της ημέρας της εβδομάδας και των εποχιακών διακυμάνσεων, καθιστώντας την έναν εξαιρετικά μεταβλητό και δυναμικό παράγοντα στο σχεδιασμό διαδρομών. Η αξιολόγηση με βάση το χρόνο αποσκοπεί στην ποσοτικοποίηση και ανάλυση αυτών των πτυχών για τον προσδιορισμό των πιο αποδοτικών σε χρόνο διαδρομών υπό διαφορετικές συνθήκες. Η αξιολόγηση αυτής της μετρικής αποτελεί πρόκληση λόγω της ευαισθησίας της σε ξαφνικές αλλαγές στη δυναμική της κυκλοφορίας και στους περιβαλλοντικούς παράγοντες.

Αξιολόγηση με βάση την απόσταση

Η αξιολόγηση μιας διαδρομής με βάση την απόσταση είναι μια άλλη θεμελιώδης μετρική στην αξιολόγηση διαδρομών. Σε αντίθεση με τον χρόνο, η απόσταση μιας διαδρομής είναι μια σχετικά σταθερή παράμετρος, η οποία καθορίζεται κυρίως από τη γεωγραφική θέση των σημείων αφετηρίας και προορισμού. Η σημασία της αξιολόγησης της απόστασης έγκειται στην ικανότητά της να προσφέρει μια σταθερή μετρική για τη σύγκριση της χωρικής αποτελεσματικότητας διαφορετικών διαδρομών. Αυτή η μετρική είναι ιδιαίτερα χρήσιμη στον υλικοτεχνικό σχεδιασμό και την ανάλυση κόστους, όπου η απόσταση συσχετίζεται άμεσα με την κατανάλωση καυσίμων, τη φθορά και άλλα λειτουργικά κόστη. Η σταθερότητα και η αντικειμενικότητα της απόστασης ως μετρικής την καθιστούν μια απλή και αξιόπιστη μέθοδο για την αξιολόγηση διαδρομών.

3.2.2 Αξιολόγηση Συσταδοποίησης

Σε αυτό το σημείο, παρουσιάζουμε μια νέα μέθοδο για την αξιολόγηση των χωρικών περιοχών που χρησιμοποιούνται στις διαδρομές των οδηγών. Η προσέγγιση αυτή αποσκοπεί στην ανάλυση της αποτελεσματικότητας των διαδρομών με βάση τη συγκέντρωση μεμονωμένων διαδρομών σε διακριτές συστάδες. Η αξιολόγηση κάθε χωρικής περιοχής πραγματοποιείται με τη χρήση ιστορικών δεδομένων διαδρομών. Κατά τη διάρκεια μιας διαδρομής παράδοσης ή παραλαβής, οι οδηγοί πλοηγούνται από το σημείο εκκίνησης στον προορισμό τους, κάνοντας καθορισμένες στάσεις. Ωστόσο, ο αριθμός και η διάταξη αυτών των στάσεων, ενώ αρχικά καθορίζονται από το σύστημα δρομολόγησης, ενδέχεται να μεταβληθούν κατά τη διάρκεια του ταξιδιού. Επομένως, είναι ζωτικής σημασίας να διακρίνονται μόνο οι στάσεις που πραγματοποιούνται στην τελική εκτελούμενη διαδρομή.

Επιπλέον, κατά τη συζήτηση των μετρήσεων χρόνου και απόστασης, θεωρείται ότι αυτές μετρώνται από την προηγούμενη στάση. Εάν δεν υπάρχει προηγούμενη στάση, που σημαίνει ότι ο οδηγός βρίσκεται στο σημείο τερματισμού (αφετηρία), θεωρείται άσχετη με τη συμπερίληψη στη μετρική αξιολόγησης. Αυτός ο χωρικός δείκτης αξιολόγησης έχει σχεδιαστεί για να είναι ανεξάρτητος από τον οδηγό, ενσωματώνοντας στάσεις σε μια συγκεκριμένη περιοχή κατά τη διάρκεια μιας προκαθορισμένης περιόδου. Από αυτές τις στάσεις, υπολογίζονται ο μέσος χρόνος και η απόσταση των δρομολογίων που πραγματοποιούνται στην περιοχή με τη χρήση της περιγραφόμενης μεθοδολογίας.

Το ερώτημα που παρέχεται χρησιμεύει ως υπολογιστική βάση για την αξιολόγηση αυτή, συγκεντρώνοντας δεδομένα ανά οδηγό. Φιλτράρει τις στάσεις που δεν σχετίζονται με την ανάλυση (όπως οι παραλαβές και οι παραδόσεις σε αποθήκες) και περιλαμβάνει μόνο τις στάσεις που ολοκληρώθηκαν επιτυχώς ή απλώς επιχειρήθηκαν εντός των καθορισμένων χωρικών περιοχών. Το ερώτημα υπολογίζει τον μέσο χρόνο που δαπανάται ανά επιτυχημένη εργασία και συγκεντρώνει τις ολοκληρωμένες στάσεις, προσφέροντας έτσι μια αποχρώσα κατανόηση της επιχειρησιακής αποδοτικότητας εντός των καθορισμένων περιοχών.

Αυτή η χωρική μετρική αξιολόγησης προσφέρει μια πρωτοποριακή προοπτική για τη βελτιστοποίηση δρομολογίων, εστιάζοντας όχι απλώς στην ατομική απόδοση αλλά στη συλλογική αποδοτικότητα εντός προκαθορισμένων γεωγραφικών ορίων. Αναλύοντας τα συγκεντρωτικά δεδομένα των δρομολογίων που έχουν ομαδοποιηθεί με βάση τη χωρική εγγύτητα, η μέθοδος αυτή παρέχει πληροφορίες σχετικά με τη λειτουργική δυναμική των δραστηριοτήτων παράδοσης και παραλαβής, αναδεικνύοντας τις περιοχές αποδοτικότητας και εκείνες που χρήζουν βελτίωσης. Μια τέτοια προσέγγιση επιτρέπει μια πιο λεπτομερή και ολοκληρωμένη ανάλυση της αποτελεσματικότητας των δρομολογίων, συμβάλλοντας σημαντικά στον στρατηγικό σχεδιασμό και τη βελτιστοποίηση των λειτουργιών μεταφοράς και εφοδιαστικής.

Παρακάτω παρουσιάζουμε και τον αλγόριθμο που χρησιμοποιείται για την αξιολόγηση συσταδοποίησης.

Algorithm 6 Calculate Cluster Metrics from Spatial Data

Require:

spatial_data: The spatial area data for evaluation.

date: Optional date for filtering tasks.

Ensure:

clusterMetrics: A list of metrics.

- 1: *routeTasks* \leftarrow Retrieve tasks within *spatial_data*.
 - 2: **if** *date* is provided **then**
 - 3: Filter *routeTasks* by *date*.
 - 4: **end if**
 - 5: Sort *routeTasks* by driver, date, and latestEta.
 - 6: Calculate *nextEta* for each task by shifting latestEta by one position.
 - 7: *groupedTasks* \leftarrow Group *routeTasks* by driver, date, and cluster.
 - 8: Aggregate within *groupedTasks* to calculate:
 - Number of completed tasks.
 - *minutesSpent* by taking the difference between *nextEta* and *latestEta* in minutes for each task, then summing these differences within each group.
 - 9: Sort computed routes by driver and date.
 - 10: Aggregate total driving distance for each driver and date.
 - 11: *perDayMetrics* \leftarrow Calculate metrics per day :
 - Average minutes spent per task.
 - Total number of completed tasks.
 - Average distance per task.
 - 12: *clusterAverageMinutes* \leftarrow Calculate average minutes spent per task.
 - 13: *clusterAverageDistance* \leftarrow Calculate average distance per task.
 - 14: *clusterMetrics* \leftarrow Combine *clusterAverageMinutes* and *clusterAverageDistance*.
 - 15: **return** *clusterMetrics*
-

3.3 Βάση δεδομένων και Views

Όπως είπαμε και στον προηγούμενο μέρος [2.3.3](#), η βάση δεδομένων που χρησιμοποιήθηκε είναι η PostgreSQL με την επέκταση PostGIS. Η βάση δεδομένων περιέχει τα δεδομένα που περιγράψαμε στο Κεφάλαιο [3.1](#).

3.3.1 Πίνακες

Οι πίνακες για τους οποίους έχουμε ήδη εξηγήσει τα δεδομένα που αποθηκεύουν είναι οι εξής:

- **route_tasks**: Περιέχει τα δεδομένα των στάσεων [3.1.1](#).
- **computed_routes**: Περιέχει τα δεδομένα των υπολογισμένων δρομολογίων [3.1.1](#).
- **zipcodes**: Περιέχει τα δεδομένα των ταχυδρομικών κωδικών.
- **piles**: Περιέχει την υπάρχουσα λύση των συστάδων [3.1.3](#).

Επιπλέον, θα χρειαστούμε έναν πίνακα για την αποθήκευση των αποτελεσμάτων των clustering μοντέλων. Έναν για τις παραγόμενες συστάδες και έναν ακόμα για τις παραμέτρους των μοντέλων.

Πίνακας clustering_algorithms

Ο πίνακας `clustering_algorithms` περιέχει τις παραμέτρους των clustering μοντέλων. Το λογισμικό για κάθε νέο πεδίο στους αλγορίθμους συσταδοποίησης φτιάχνει ένα νέα στήλη στον πίνακα. Επομένως ο πίνακας αυτός είναι δυναμικός και μπορεί να αλλάξει ανάλογα με τον αλγόριθμο που χρησιμοποιείται. Ενδεικτικά, ο πίνακας περιέχει τα εξής πεδία:

Όνομα Στήλης	Τύπος	Περιγραφή
id	integer	Μοναδικός αριθμός αναγνώρισης
name	varchar(255)	Όνομα αλγορίθμου
n_clusters	integer	Αριθμός συστάδων
max_iter	integer	Μέγιστος αριθμός επαναλήψεων

Πίνακας 3.4: Πεδία του πίνακα `clustering_algorithms`.

Πίνακας clustering_results

Ο πίνακας `clustering_results` περιέχει τα αποτελέσματα των clustering μοντέλων. Περιέχει τα εξής πεδία:

Όνομα Στήλης	Τύπος	Περιγραφή
id	integer	Μοναδικός αριθμός αναγνώρισης
date_range	daterange	Εύρος ημερομηνιών για τα task που έγινε fit το μοντέλο
city	varchar(255)	Όνομα πόλης για την οποία υπάγονται τα task που έγινε fit το μοντέλο
attributes	text[]	Λίστα απο types των tasks που έγινε fit το μοντέλο, προεπιλογή σε 'all'
algorithmid	integer	Αναγνωριστικό αλγορίθμου, αναφορά σε clustering_algorithms
cluster_id	integer	Αναγνωριστικό συστάδας
cluster_points	geometry(MultiPoint)	Γεωμετρικά σημεία των task που έγινε fit το μοντέλο
convex_hull	geometry	Κυρτό περίβλημα της συστάδας
centroid	geometry(Point)	Γεωμετρικό κέντρο της συστάδας
cluster_stats	jsonb	Στατιστικά στοιχεία της συστάδας σε μορφή JSON
label	varchar	Ετικέτα της συστάδας

Πίνακας 3.5: Πεδία του πίνακα clustering_results.

Πίνακας new_piles

Ο πίνακας new_piles περιέχει τα αποτελέσματα συστάδων των ταχυδρομικών κωδικών απο τα clustering_results. Περιέχει τα εξής πεδία:

Όνομα Στήλης	Τύπος	Περιγραφή
id	serial	Μοναδικός αριθμός αναγνώρισης, πρωτεύον κλειδί
weighted_metric_params	varchar	Παράμετροι βαρυμένης μετρικής συνήθως χρόνος ή απόσταση
shape	geometry(MultiPolygon)	Γεωμετρικό σχήμα σε μορφή MultiPolygon
num_of_zips	integer	Αριθμός ταχυδρομικών κωδικών που περιλαμβάνονται
cluster_id	integer	Αναγνωριστικό της συστάδας
zips	text	Ταχυδρομικοί κώδικες σε μορφή κειμένου
cluster_stats	jsonb	Στατιστικά στοιχεία της συστάδας σε μορφή JSONB
algorithmid	integer	Αναγνωριστικό του αλγορίθμου, αναφορά σε clustering_algorithms
label	varchar	Ετικέτα της συστάδας

Πίνακας 3.6: Πεδία του πίνακα new_piles.

3.3.2 SQL Views

Για τη βελτίωση της αποδοτικότητας και την ευκολία διαχείρισης διαφόρων λειτουργιών και ερωτημάτων εντός της βάσης δεδομένων, υιοθετήθηκε η χρήση των SQL Views. Τα SQL Views αναφέρονται σε ερωτήματα τα οποία έχουν αποθηκευτεί στη βάση δεδομένων και λειτουργούν ως εικονικοί πίνακες. Ένα

View είναι δυνατόν να συνθέτει δεδομένα από μία ή περισσότερες πηγές πινάκων. Η χρήση των Views ενδείκνυται για την εφαρμογή περιορισμών πρόσβασης σε δεδομένα, την παροχή προσαρμοσμένων προβολών δεδομένων, καθώς και την απλοποίηση και επιτάχυνση εκτελεστικών ερωτημάτων που γίνονται συχνά.

Συγκεκριμένα, χρησιμοποιήθηκαν Views για την αποδοτική εξαγωγή δεδομένων από τους πίνακες `route_tasks` και `zipcodes`.

Το view `clustering_views` περιέχει τα δεδομένα που χρησιμοποιήθηκαν για την εκπαίδευση των clustering μοντέλων. Συγκεκριμένα το `clustering_view_1`, περιέχει τις στάσεις των οδηγών απο 01/10/2023 έως 15/10/2023 για την περιοχή της Αττικής. Υπάρχουν και άλλα δύο views που αφορούν την εκπαίδευση των μοντέλων. Το `clustering_view_2` περιέχει τις στάσεις των οδηγών απο 01/12/2023 έως 31/12/2023 για την περιοχή της Αττικής εξαιρουμένων των στάσεων που αφορούν τους κεντρικούς κόμβους (αποθήκες). Το `clustering_view_3` περιέχει τις στάσεις των οδηγών απο 01/10/2023 έως 31/12/2023 για την περιοχή της Αττικής, τα οποία έχουν ολοκληρωθεί επιτυχώς, αφορούν πελάτες (παράδοση ή παραλαβή) και έχουν προγραμματιστεί απο την προηγούμερα μέρα. Το view `clustering_view_1` αποτελούν 180 χιλιάδες στάσεις, το `clustering_view_2` 400 χιλιάδες και το `clustering_view_3` 700 χιλιάδες.

Επιπροσθέτως, το View `overlapping_zipcodes` αποτυπώνει τις περιοχές με επικαλύψεις ζωνών ταχυδρομικών κωδίκων.

Για την ταχεία ανάκτηση δεδομένων σχετικών με τις συστάδες της Αττικής, χρησιμοποιήθηκε επίσης το View `ath_piles`, το οποίο παρέχει πρόσβαση στην υφιστάμενη διαμόρφωση των συστάδων της περιοχής αναφοράς (βλ. 3.1.3).

3.3.3 ER Diagram

Παρακάτω παρουσιάζεται το ER Diagram της βάσης δεδομένων.

public.shipping_orders		
id	int4	
whichWarehouse	varchar(2147483647)	N
driverFk	int4	N
merchantLocationId	int4	N
routingProjectDate	date	N
createdAt	timestamp	N

public.route_tasks		
id	int4	
date	date	
relatedOrderIds	text	N
type	varchar(2147483647)	
state	varchar(2147483647)	
result	varchar(2147483647)	N
originalEta	timestamp(6)	
latestEta	timestamp(6)	N
completedAt	timestamp(6)	N
assignedFrom	varchar(2147483647)	
plannedLocation	int4	
completedLocation	int4	N
plannedServiceTime	int4	
createdAt	timestamp(6)	
updatedAt	timestamp(6)	
previousDriverId	int4	N
driverId	int4	N
failReason	varchar(2147483647)	N
messageToDriver	varchar(1024)	N
successReason	varchar(2147483647)	N
updatedBy	varchar(2147483647)	
reasonText	varchar(1024)	N
arrivedAt	timestamp(6)	N
taskAttributes	text	N
latestEtaUpdatedAt	timestamp(6)	N

public.zipcodes		
id	int4	
shape	int4	
zip	int4	
prefecture	varchar(2147483647)	
municipal	varchar(2147483647)	
name	varchar(2147483647)	
type	varchar(2147483647)	N
pile	varchar(2147483647)	
warehouse	varchar(2147483647)	N

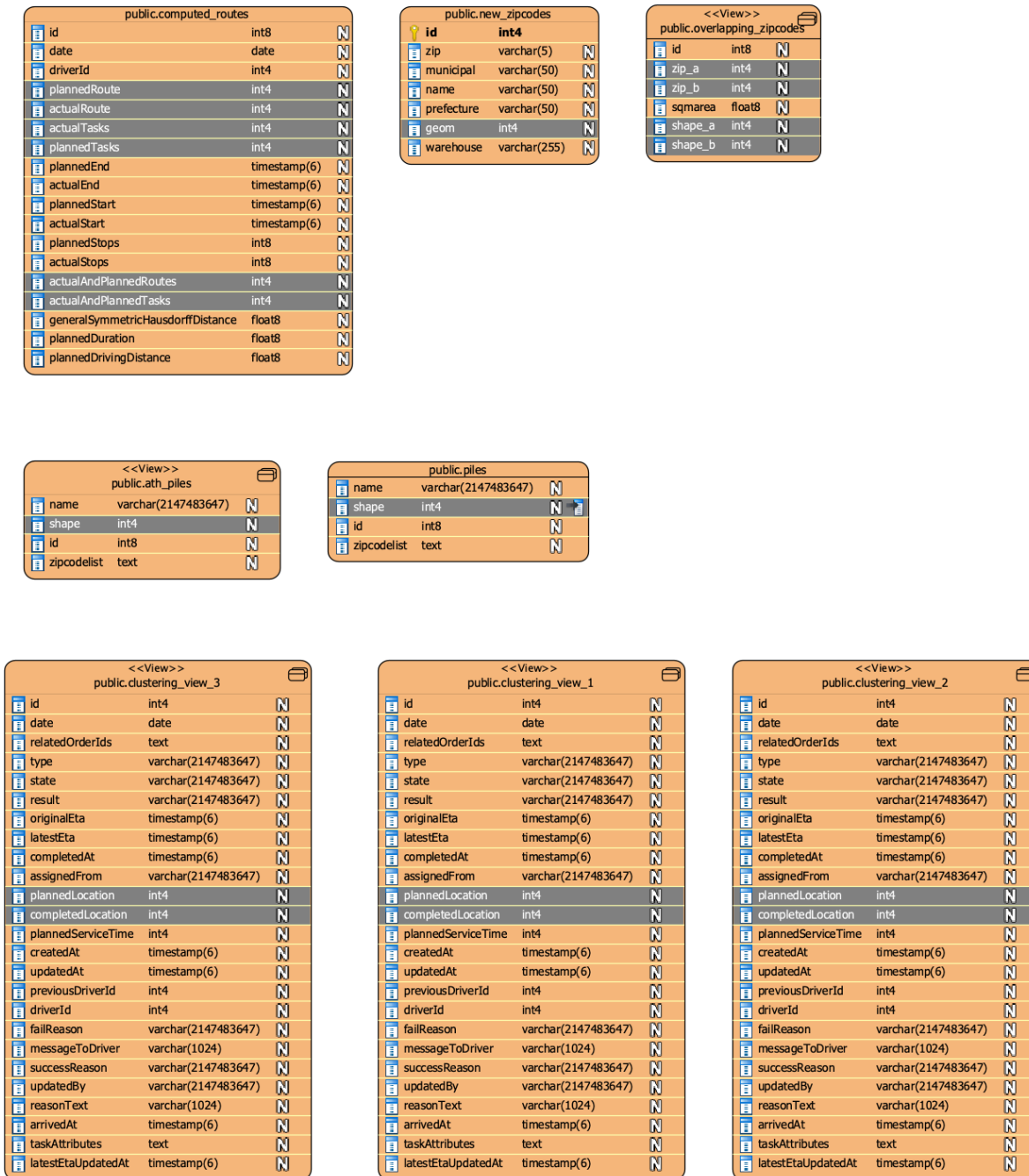
public.clustering_algorithms		
id	int4	
name	text	
n_clusters	int4	
init	text	N
max_iter	int4	N
n_init	int4	N
created_at	timestamp(6)	N
algorithm	text	N
copy_x	bool	N
random_state	int4	N
tol	float8	N
verbose	int4	N
batch_size	int4	N
compute_labels	bool	N
init_size	text	N
max_no_improvement	int4	N
reassignment_ratio	float8	N
n_jobs	int4	N
size_max	int4	N
size_min	int4	N
affinity	text	N
compute_distances	bool	N
compute_full_tree	text	N
connectivity	text	N
distance_threshold	text	N
linkage	text	N
memory	text	N
metric	text	N

public.clustering_results		
id	int4	
date_range	int4	U
city	varchar(255)	U
attributes	int4	
algorithmid	int4	U
cluster_id	int4	U
cluster_points	int4	
convex_hull	int4	
centroid	int4	
cluster_stats	int4	
label	varchar(2147483647)	N

public.new_piles		
id	int4	
weighted_metric_params	varchar(2147483647)	N U
shape	int4	N
num_of_zips	int4	N
cluster_id	int4	N
zips	text	N U
cluster_stats	int4	N
algorithmid	int4	N U
label	varchar(2147483647)	N

clustering_results_algorithmid_fkey

new_piles_algorithmid_fkey



Σχήμα 3.3.1: ER Diagram της βάσης δεδομένων. Με γκρι χρώμα φαίνονται τα geometry columns.

Κεφάλαιο 4

Λύση του προβλήματος

Η προσεκτική ανάλυση του προβλήματος που πραγματοποιήσαμε μέχρι στιγμής, μας οδηγεί στην κρίσιμη στιγμή της πρότασης μιας συνολικής λύσης. Στο πλαίσιο αυτού του κεφαλαίου, θα αναδείξουμε την προσέγγιση και τη μεθοδολογία που ακολουθήσαμε για την ανάπτυξη του προτεινόμενου λογισμικού. Θα σχεδιάσουμε μια αρχιτεκτονική λογισμικού που επιτρέπει την ευέλικτη ενσωμάτωση νεότερων και πιο προηγμένων τεχνολογικών λύσεων στο εκάστοτε τμήμα του λογισμικού. Μέσα από αυτή τη διαδικασία, θα παρουσιάσουμε επίσης τα κρίσιμα διαγράμματα και μοντέλα που απεικονίζουν λεπτομερώς τη λειτουργική δομή και τη δυναμική του λογισμικού, επιδιώκοντας να προσφέρουμε μια σαφή και συγκροτημένη εικόνα της προτεινόμενης λύσης.

4.1 Υλοποίηση

Στην ενότητα αυτή αναλύεται η διαδικασία υλοποίησης μιας λύσης που αποσκοπεί στην επίλυση του προβλήματος που περιγράψαμε στο Κεφάλαιο 1.1. Η επίτευξη του στόχου αυτού περιλαμβάνει την αντιμετώπιση μιας σειράς υποπροβλημάτων, τα οποία απαιτούν λεπτομερή εξέταση και επεξεργασία.

- **Ελεγχος και προεπεξεργασία των δεδομένων εισόδου:** Τα δεδομένα εισόδου.
- **Επιλογή και Εκπαίδευση μοντέλου:** Επιλογή και εκπαίδευση του μοντέλου που θα χρησιμοποιηθεί
- **Παραγωγή των νέων clusters:** Παραγωγή των νέων clusters με βάση το μοντέλο που επιλέχθηκε

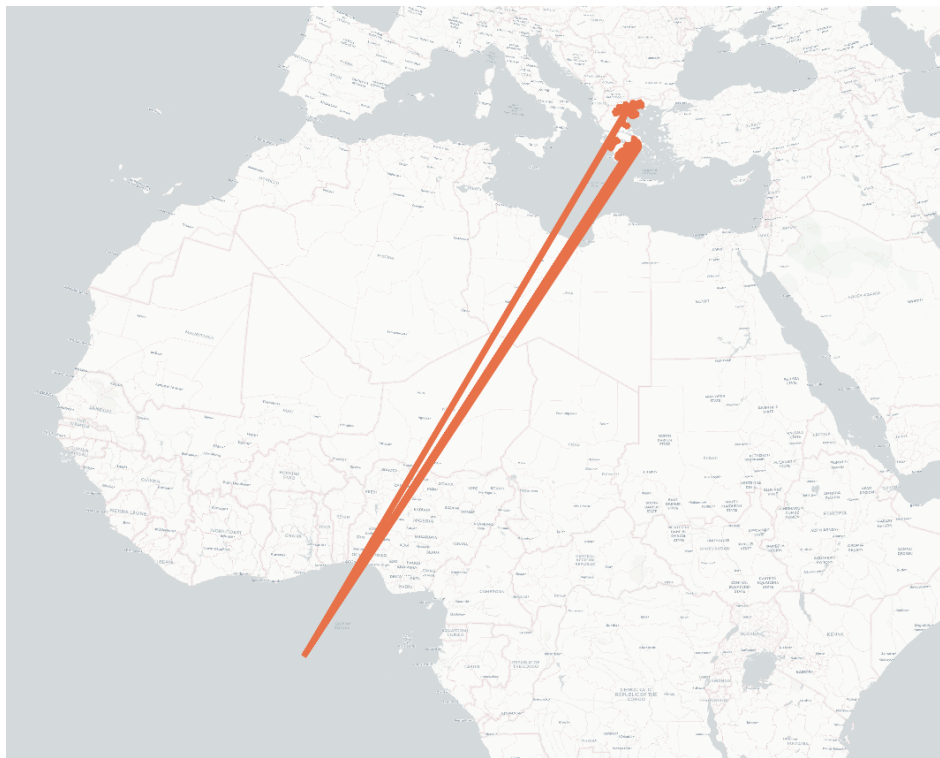
Σε αυτό το κεφάλαιο θα περιγράψουμε την υλοποίηση της κάθε μίας από τις παραπάνω διαδικασίες.

4.1.1 Δεδομένα εισόδου

Όπως αναφέρθηκε και στο Κεφάλαιο 3.1, πρόκειται για τα δεδομένα στάσεων και τα δεδομένα των ταχυδρομικών κωδικών.

4.1.2 Έλεγχος και προεπεξεργασία των δεδομένων εισόδου

Αρχικά, πρέπει να ελεγχθούν τα δεδομένα εισόδου. Απο τα δεδομένα των στάσεων μερικές φορές προκύπτουν αντιφάσεις όσο αφορά την γεωγραφική τοποθεσία της στάσης και την διεύθυνση. Συγκεκριμένα, αυτό που παρατηρήσαμε είναι η τοποθεσία POINT(0 0) που προφανώς πρόκειται για κάποιο λάθος συστήματος απο πρευράς δεδομένων. Εμείς για να μην επηρεάσουμε τα δεδομένα μας αφαιρέσαμε τις στάσεις που είχαν αυτή την τοποθεσία.



Σχήμα 4.1.1: Στάσεις με τοποθεσία POINT(0 0)

Επιπλέον, παρατηρήσαμε ότι μερικά σχήματα ταχυδρομικών κωδικών είχαν επικαλύψεις. Αυτό σημαίνει πρακτικά ότι ένα σημείο μπορεί να ανήκει σε 2 ταχυδρομικούς κωδικούς. Ενδεικτικά, παραθέτουμε μερικά παραδείγματα :

Ταχυδρομικός Κωδικός A	Ταχυδρομικός Κωδικός B	Επικαλυπτούμενη επιφάνεια
62500	62100	81870404
34100	34150	32704700
19003	19023	2077749
55534	55535	884011
59100	59132	705311
15239	15236	576337
56238	56224	470681
34131	34100	268988

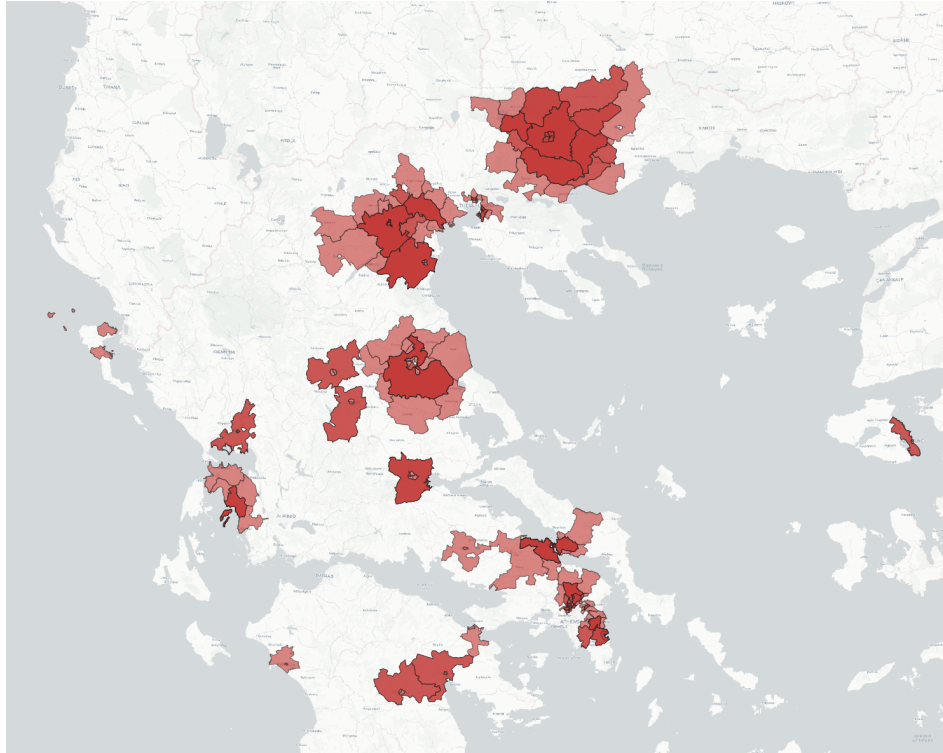
Πίνακας 4.1: Παράδειγμα επικάλυψης ταχυδρομικών κωδικών

Κάποια παραδείγματα φαίνονται παρακάτω στην εικόνα [4.1.2](#). Απο αυτά προκύπτει ότι οι περισσότερες επικαλύψεις δεν αφορούν τα δεδομένα, αλλά αργότερες ενημερώσεις των ταχυδρομικών κωδικών στις εκάστοτε περιοχών.

Για να αντιμετωπίσουμε αυτό το πρόβλημα αλλάξαμε τα σχήματα των ταχυδρομικών κωδικών δίνοντας την επικάλυψη σε ένα από τα 2 μέρη που συμμετείχαν στην επικάλυψη. Για να μπορέσουμε να κάνουμε αυτή την αλλαγή χρησιμοποιήσαμε ο παρακάτω αλγόριθμος [7](#).

Η διαδικασία προεπεξεργασίας και επικύρωσης των δεδομένων αποτελεί θεμέλιο για την ακρίβεια και την ορθότητα των τελικών clusters.

Παραθέτουμε και το αντίστοιχο activity diagram στην εικόνα [4.1.4](#).



Σχήμα 4.1.2: Παράδειγμα επικάλυψης ταχυδρομικών κωδικών

Algorithm 7 Resolve Zip Code Area Overlaps

Require:

Z : A set of all zip codes.

$shape(\cdot)$: A function that returns the geometric shape of a zip code area.

$intersect(\cdot, \cdot)$: A function that returns the geometric intersection of two shapes.

$difference(\cdot, \cdot)$: A function that returns the geometric difference between two shapes.

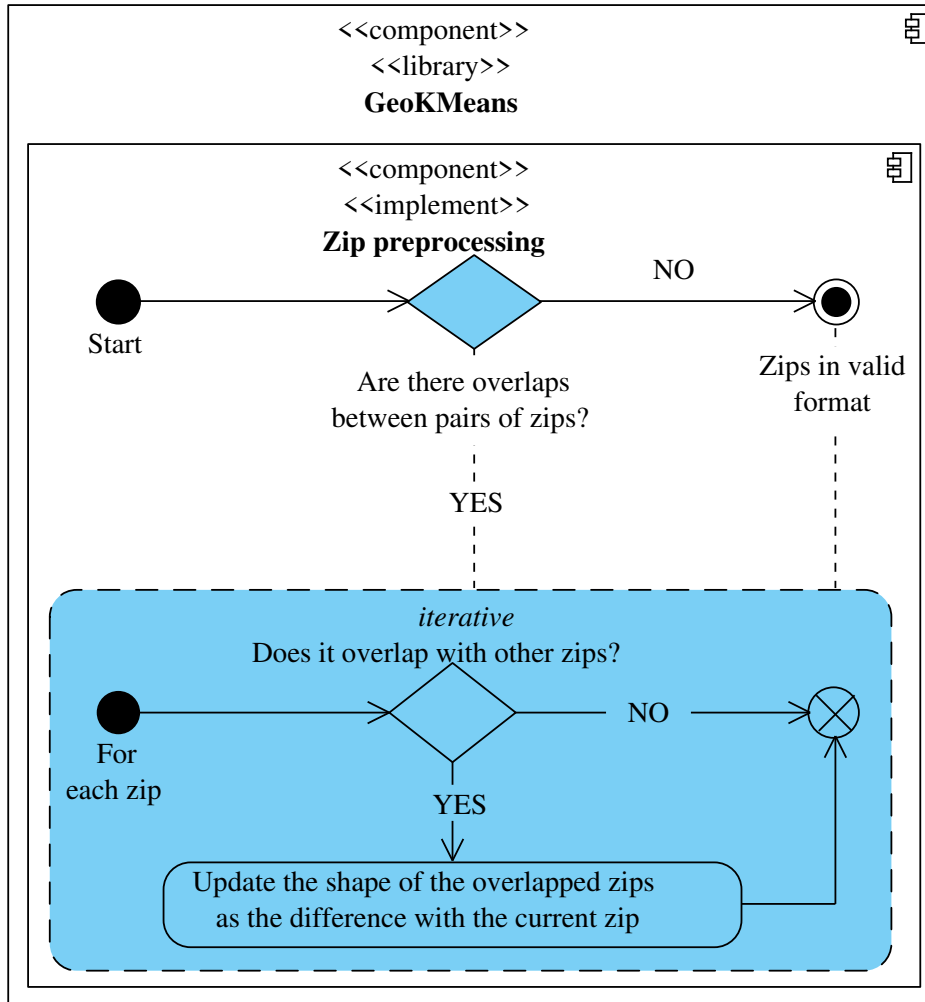
$union(\cdot, \cdot)$: A function that returns the geometric union of two shapes.

$overlappingZips$: A set of pairs (A, B) indicating overlapping zip codes.

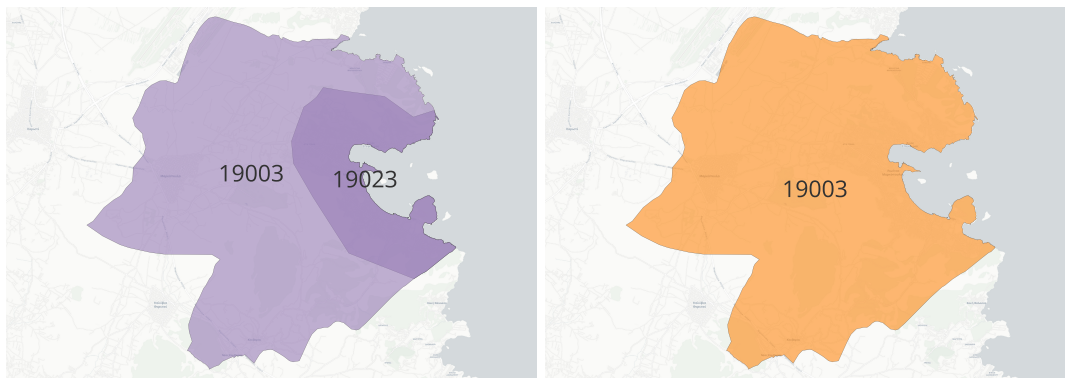
```

1: for each pair  $(A, B)$  in  $overlappingZips$  do
2:    $intersectedZipsA \leftarrow \emptyset$ 
3:   for each  $Z$  in  $Z$  do
4:     if  $Z \neq A$  then
5:        $intersectedZipsA \leftarrow intersectedZipsA \cup \{intersect(shape(Z), shape(A))\}$ 
6:     end if
7:   end for
8:   for each  $Z$  in  $intersectedZipsA$  do
9:      $shape(Z) \leftarrow difference(shape(Z), shape(A))$ 
10:     $shape(A) \leftarrow union(shape(A), intersect(shape(Z), shape(A)))$ 
11:   end for
12: end for

```

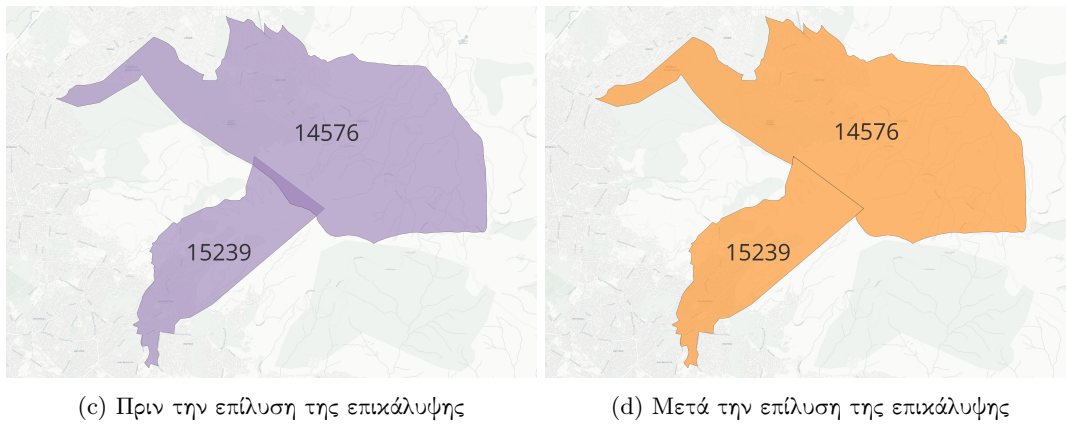


Σχήμα 4.1.3: Activity diagram για την επίλυση της επικάλυψης των ταχυδρομικών κωδικών



(a) Πριν την επίλυση της επικάλυψης

(b) Μετά την επίλυση της επικάλυψης



Σχήμα 4.1.4: Επίλυση της επικάλυψης των ταχυδρομικών κωδικών

4.1.3 Επιλογή και Εκπαίδευση μοντέλου

Στο επόμενο βήμα της διαδικασίας υλοποίησης, μετά την προσεκτική προεπεξεργασία και τον καθορισμό των δεδομένων εισόδου, επικεντρωθήκαμε στην επιλογή του μοντέλου που θα χρησιμοποιηθεί για τη συσταδοποίηση των δεδομένων. Αυτή η επιλογή κρίνεται καθοριστικής σημασίας, καθώς το κατάλληλο μοντέλο θα πρέπει να ανταποκρίνεται επαρκώς στις απαιτήσεις και τις προκλήσεις του προβλήματος που αντιμετωπίζουμε.

Επιλέξαμε τον αλγόριθμο KMeans ως τη βασική μεθοδολογία για την συσταδοποίηση, λόγω της απλότητας, της αποδοτικότητας και της ευρείας αποδοχής του στην κοινότητα της επιστήμης δεδομένων. Προκειμένου να επιτύχουμε ακόμη καλύτερη επίδοση και να αντιμετωπίσουμε ειδικότερες προκλήσεις, ενσωματώσαμε επίσης δύο παραλλαγές του KMeans: το MiniBatchKMeans και το KMeans Constrained.

Τα αποτελέσματα της συσταδοποίησης αποτυπώνονται σε παραδείγματα που δείχνουν την αποτελεσματικότητα των επιλεγμένων αλγορίθμων, καθώς και την ικανότητά τους να ομαδοποιούν τα δεδομένα με τρόπο που είναι συνεπής με τους στόχους της μελέτης μας.

4.1.4 Παραγωγή των νέων clusters

Στην τελική φάση της διαδικασίας υλοποίησης, μετά την επιτυχή εκπαίδευση των μοντέλων και τη δημιουργία αρχικών clusters, προχωρούμε στην παραγωγή νέων συστάδων, οι οποίες ομαδοποιούν τους ταχυδρομικούς κωδικούς με βάση τα αποτελέσματα που προέκυψαν. Για την καλύτερη κατανόηση και διαχείριση, οι νέες συστάδες θα αναφέρονται στο εξής ως "συστάδες" και τα αρχικά clusters ως "clusters".

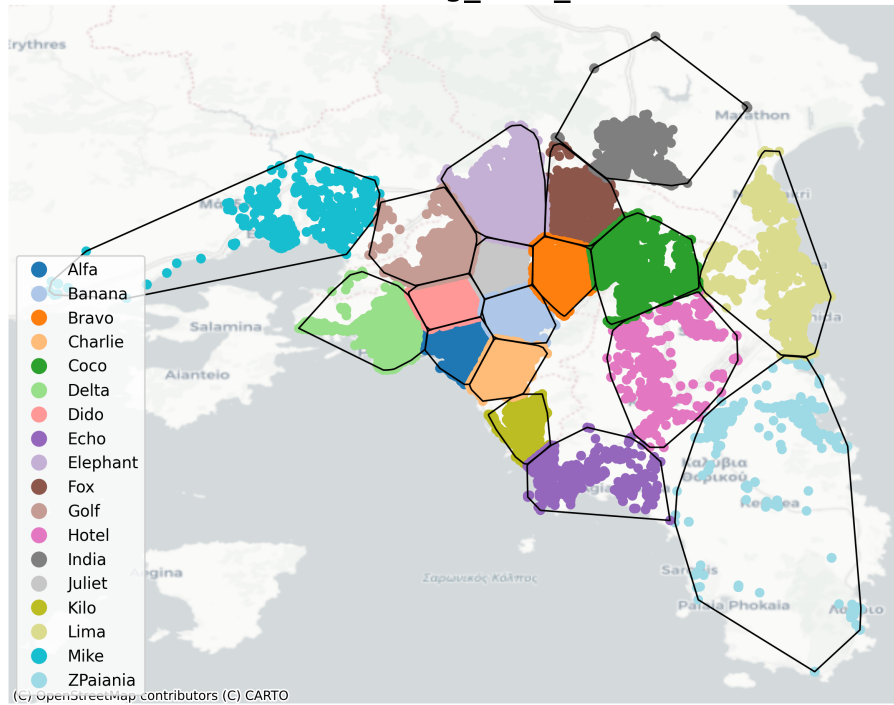
Η δημιουργία των νέων συστάδων επιτυγχάνεται μέσω ενός νέου αλγορίθμου που αναπτύξαμε, ο οποίος διαδραματίζει κρίσιμο ρόλο στην επίτευξη της τελικής μας στόχευσης. Ο αλγόριθμος αυτός ακολουθεί τρία βασικά βήματα.

Ξεκινάμε με την αξιολόγηση των MultiPolygons που προκύπτουν από τον αλγόριθμο KMeans, χρησιμοποιώντας τη μέθοδο `intersects` της βιβλιοθήκης `shapely` για να εντοπίσουμε περιοχές που έχουν κοινά σημεία με τους ταχυδρομικούς κωδικούς.

Αν μια περιοχή από τα clusters τέμνει μόνο με έναν ταχυδρομικό κωδικό, τότε αυτός ο ταχυδρομικός κωδικός ανήκει αυτόματα στην αντίστοιχη συστάδα. Στην εικόνα 4.1.9 ο ταχυδρομικός κωδικός 17778 ανήκει στο cluster Alfa, άρα η νέα συστάδα Alfa θα περιέχει τον ταχυδρομικό κωδικό 17778.

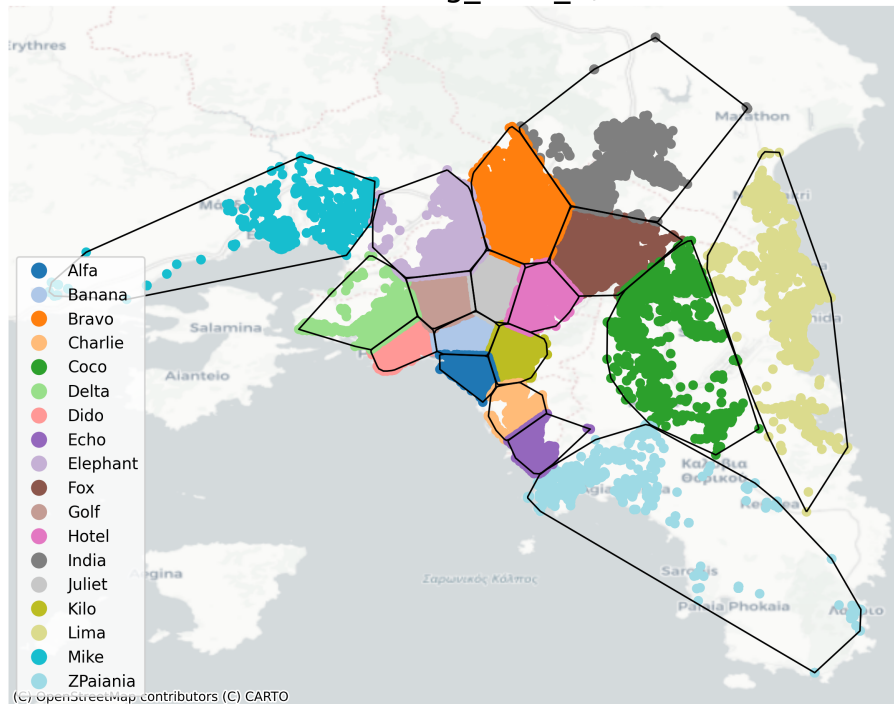
Η πρόκληση προκύπτει όταν πολλοί ταχυδρομικοί κωδικοί εμπίπτουν σε μια συστάδα. Σε αυτή την περίπτωση, επιλέγουμε εκείνον με τη μεγαλύτερη επικάλυψη, λαμβάνοντας υπόψη εξαιρέσεις και πιθανές αναγκαιότητες ειδικής αντιμετώπισης. Για παράδειγμα στην εικόνα 4.1.9 ο ταχυδρομικός κωδικός 11855, ανήκει σε τρεις clusters, αλλά έχει παραπάνω από 33% επικάλυψη με το cluster Alfa, άρα ανήκει στο cluster Alfa.

KMeans Clustering view: clustering_view_1, k=18



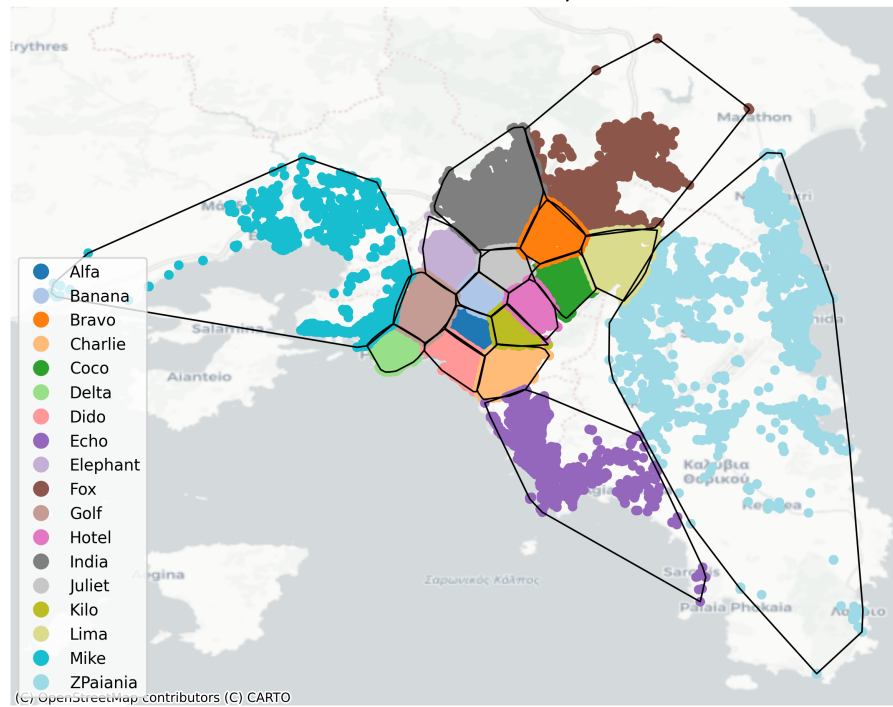
Σχήμα 4.1.5: Παράδειγμα αποτελεσμάτων συσταδοποίησης με τον αλγόριθμο KMeans

MiniBatchKMeans Clustering view: clustering_view_1, k=18

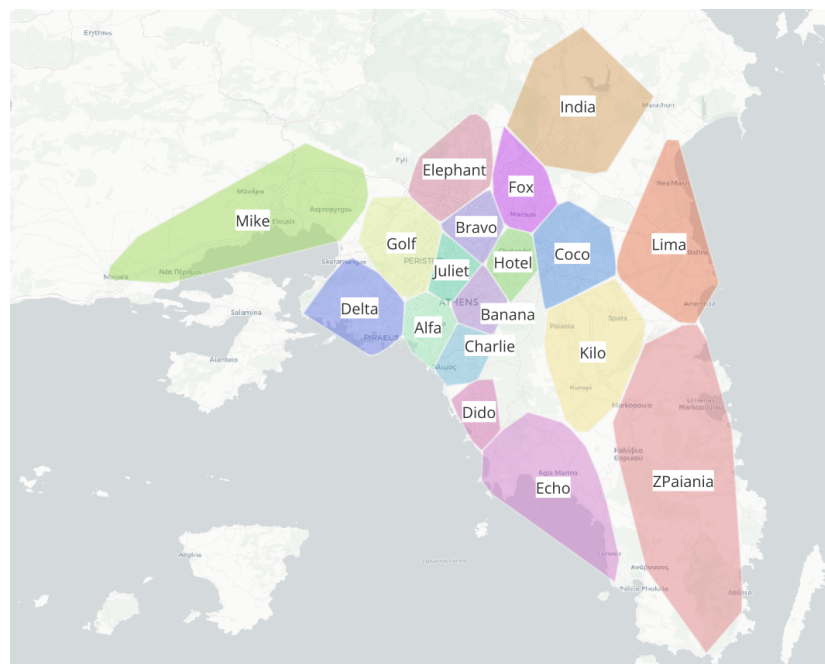


Σχήμα 4.1.6: Παράδειγμα αποτελεσμάτων συσταδοποίησης με τον αλγόριθμο MiniBatchKMeans

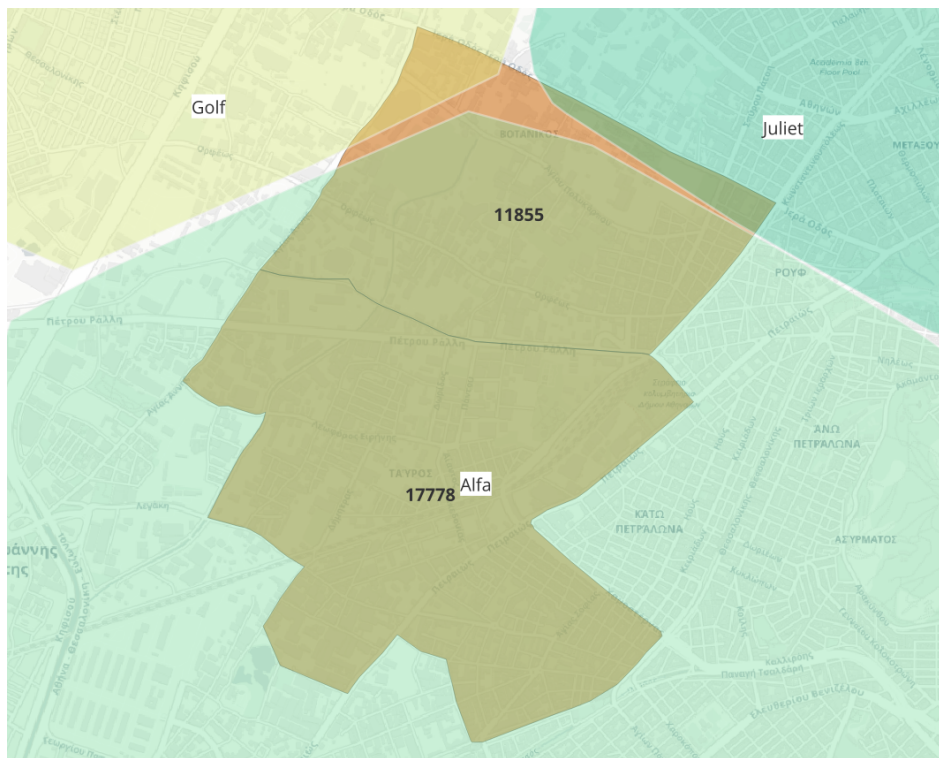
KMeans Constrained Clustering
 view: clustering_view_1b, k=18
 Cluster sizes: Max: 9530, Min: 4000



Σχήμα 4.1.7: Παράδειγμα αποτελεσμάτων συσταδοποίησης με τον αλγόριθμο KMeans Constrained



Σχήμα 4.1.8: Παράδειγμα αποτελεσμάτων συσταδοποίησης με τον αλγόριθμο KMeans



Σχήμα 4.1.9: Παράδειγμα συσταδοποίησης των ταχυδρομικών κωδικών

Για κρίσιμες περιπτώσεις όπου δεν είναι ξεκάθαρο εάν ένας ταχυδρομικός κωδικός ανήκει σε μια συστάδα, προχωρούμε σε μια πιο λεπτομερή αξιολόγηση, όπως η επιλογή βάσει του μικρότερου μέσου όρου διάρκειας των στάσεων των οδηγών από ιστορικά δεδομένα.

Παραθέτουμε, το αντίστοιχο activity diagram [4.1.10](#) στην εικόνα και ο αλγόριθμος [8](#) που χρησιμοποιήθηκε.

Algorithm 8 Generate New Clusters**Require:**

Zips: A list of zip codes .

shape(zip): Geometric shape representing the area of *zip*.

intersectingPiles(zip): Function returning a list of piles intersecting with *zip*.

intersectedAreaCluster(zip, Pile): Function returning the intersected area percentage of *zip* with a given pile.

weightedMetric(areas): Function returning the weighted metric for a list of areas.

```

1: for each zip in Zips do
2:   chosenPile  $\leftarrow$  NULL
3:   possiblePiles  $\leftarrow$  intersectingPiles(zip)
4:   L  $\leftarrow$  length of possiblePiles
5:   if L = 1 then
6:     chosenPile  $\leftarrow$  possiblePiles[0]
7:   else
8:     minimumArea  $\leftarrow$  110/L
9:     intersectedAreas  $\leftarrow$  list of intersectedAreaCluster(zip, Pile) for each pile in possiblePiles
10:    sortedIntersectedAreas  $\leftarrow$  sort intersectedAreas in descending order
11:    first  $\leftarrow$  sortedIntersectedAreas[0]
12:    if first  $\geq$  minimumArea then
13:      chosenPile  $\leftarrow$  possiblePiles[ index of first in intersectedAreas ]
14:    else
15:      Calculate the weighted metric for each cluster in possiblePiles
16:      chosenPile  $\leftarrow$  possiblePiles[ index of max(weighted metric) ]
17:    end if
18:  end if
19: end for

```

4.2 Παραμετροποιήσεις

Για μελλοντική χρήση του λογισμικού έχει γίνει προβλεψη για νέα μοντέλα συσταδοποίησης, καθώς και άλλη μηχανή δρομολόγησης.

4.2.1 Μοντέλα Συσταδοποίησης

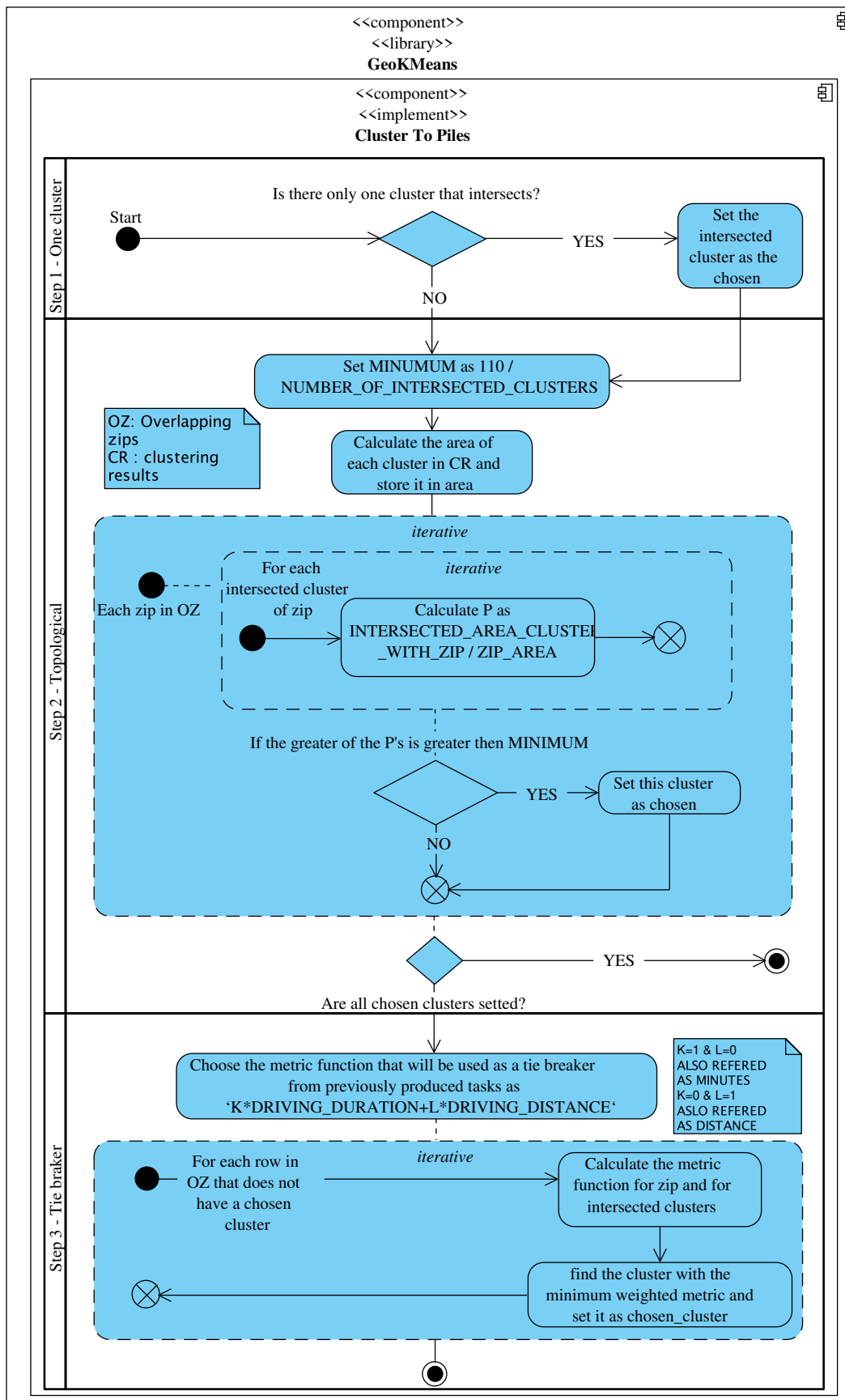
Για την προσθήκη νέων μοντέλων συσταδοποίησης, θα πρέπει να προστεθεί το ονομα και η υλοποίηση του αλγορίθμου στην μέθοδο `makeClusterModelFromNameAndParams` της κλάσης `GeoKMeans`. Επιπλέον, θα πρέπει το νέο μοντέλο να έχει την μέθοδο `fit`.

4.2.2 Μηχανή Δρομολόγησης

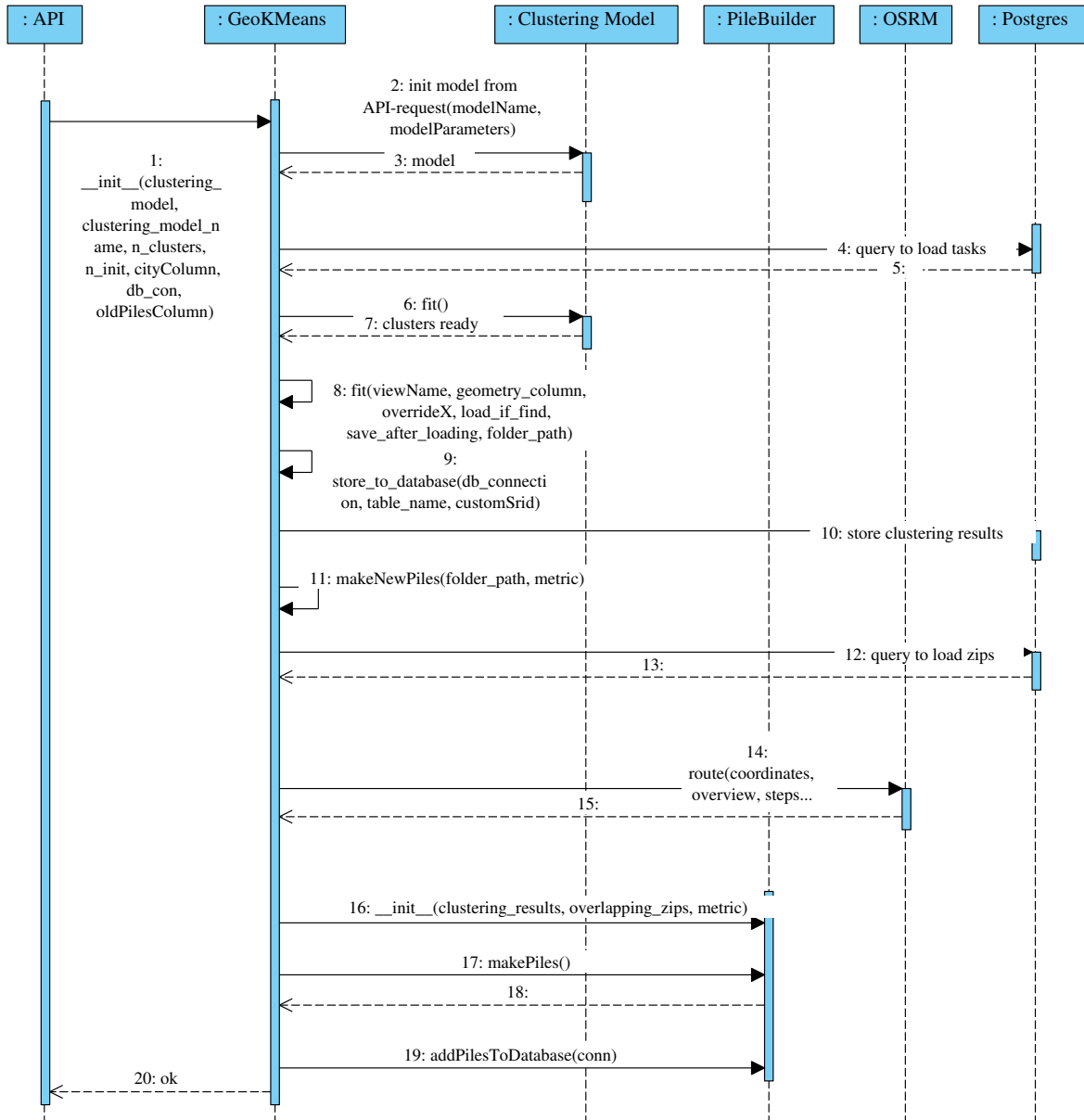
Για την προσθήκη νέας μηχανής δρομολόγησης, θα πρέπει να αντικατασταθούν οι 3 μέθοδοι της κλάσης `pyOSRM` με τις αντίστοιχες της νέας μηχανής δρομολόγησης.

4.3 Διαγράμματα

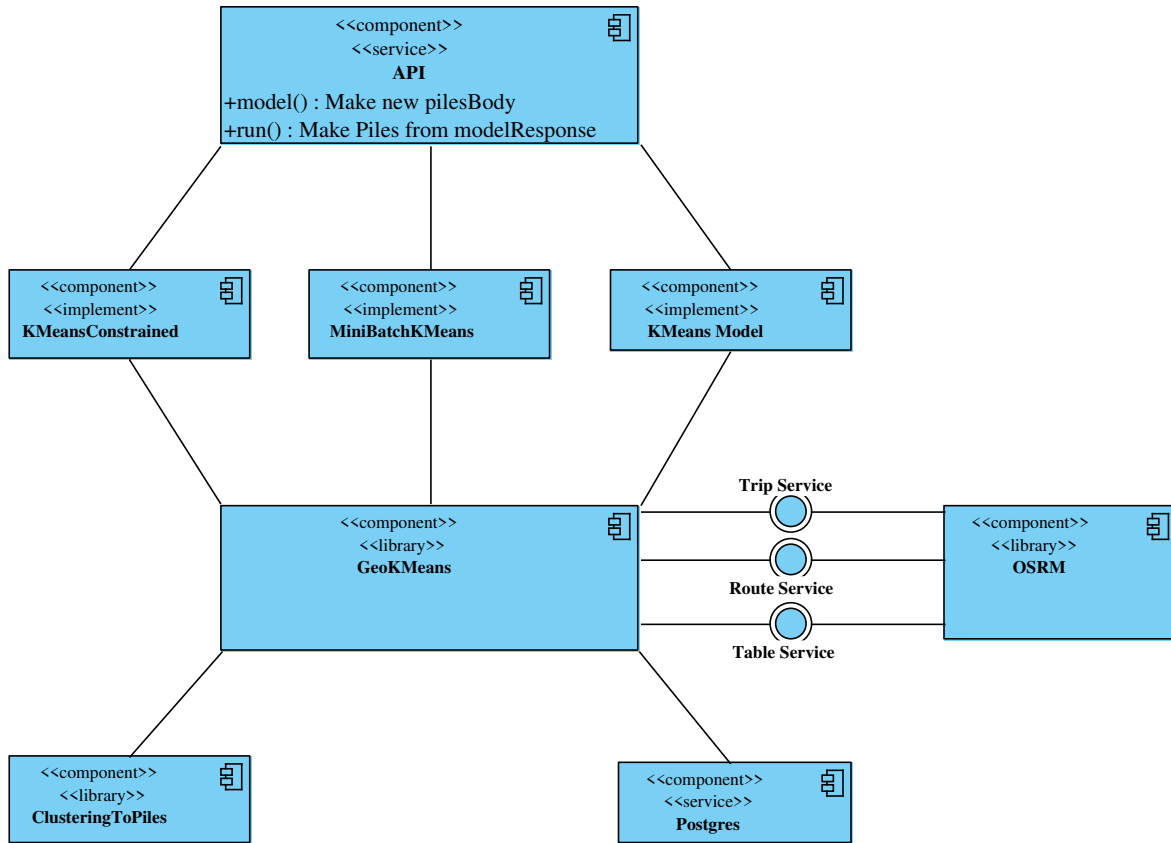
Παρακάτω παραθέτουμε το sequence, το component και το deployment diagram της αρχιτεκτονικής.



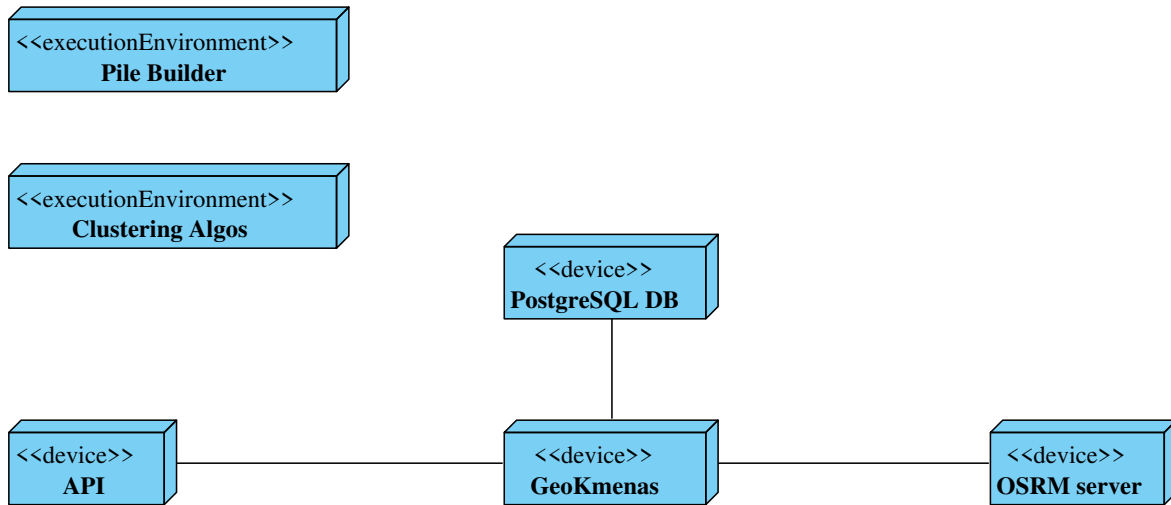
Σχήμα 4.1.10: Activity diagram για την παραγωγή των νέων συστάδων



Σχήμα 4.3.1: Sequence Diagram της αρχιτεκτονικής



Σχήμα 4.3.2: Component diagram της αρχιτεκτονικής



Σχήμα 4.3.3: Deployment diagram της αρχιτεκτονικής

Κεφάλαιο 5

Χρήση της αρχιτεκτονικής

Για την αποτελεσματικότερη χρήση του λογισμικού, δημιουργήσαμε ένα RESTful API. Στο κεφάλαιο αυτό, θα παρουσιάσουμε τον τρόπο χρήσης του API και θα δώσουμε παραδείγματα χρήσης του.

5.1 Περιγραφή του API

Το API που δημιουργήσαμε είναι ένα RESTful API. Αυτό σημαίνει ότι χρησιμοποιεί το πρωτόκολλο HTTP για την επικοινωνία μεταξύ του client και του server. Έχει μόνο ένα endpoint όπου μπορεί να καλεστεί και επιστρέφει τα αποτελέσματα του αλγορίθμου συσταδοποίησης και τα νέα clusters.

Για την χρήση του API, ο client πρέπει να κάνει ένα POST request για συγκεκριμένο αλγόριθμο `algo` και για συγκεκριμένα δεδομένα εισόδου `data`. Το API που θα πρέπει να καλεστεί είναι το `/algo/data`.

Παράμετρος	Αποδεκτές Τιμές
<code>algo</code>	KMeans, MiniBatchKMeans, KMeansConstrained
<code>data</code>	<code>clustering_view_1</code> , <code>clustering_view_2</code> , <code>clustering_view_3</code>

Πίνακας 5.1: Περιγραφή των URL παραμέτρων του API

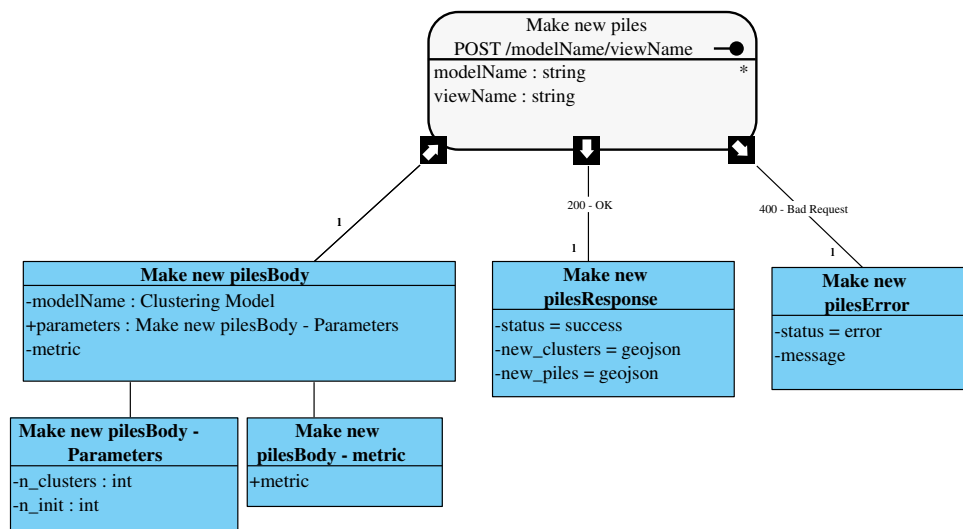
Για να οριστούν διαφορετικές παραμέτροι από τις προεπιλεγμένες, μπορούν να προστεθούν στο Body του request. Οι παράμετροι πρέπει να οριστούν σε JSON μορφή και πρέπει να έχουν την εξής μορφή:

Παράμετρος	Μορφή
<code>parameters</code>	JSON για τους παραμέτρους του αλγορίθμου από το URL parameter.
<code>metric</code>	<code>minutes</code> , <code>distance</code> ή σε μορφή <code>minutes*K+distance*L</code>

Πίνακας 5.2: Περιγραφή των BODY παραμέτρων του API

5.2 Παραδείγματα Χρήσης

Για παράδειγμα για να καλέσουμε τον αλγόριθμο KMeans με τα δεδομένα `clustering_view_1` και τις παραμέτρους `n_clusters=18`, `n_init=20`, `random_state=42`, `tol=0.0001`, `init=k-means++` και την μετρική `minutes`, θα πρέπει να κάνουμε request στο `/KMeans/clustering_view_1` με το παρακάτω body:



Σχήμα 5.1.1: Το endpoint του API

```

1  {
2      "parameters": {
3          "n_clusters": 18,
4          "init": "k-means++",
5          "n_init": 20,
6          "random_state": 42,
7          "tol": 1e-5
8      },
9      "metric": "minutes"
10 }

```

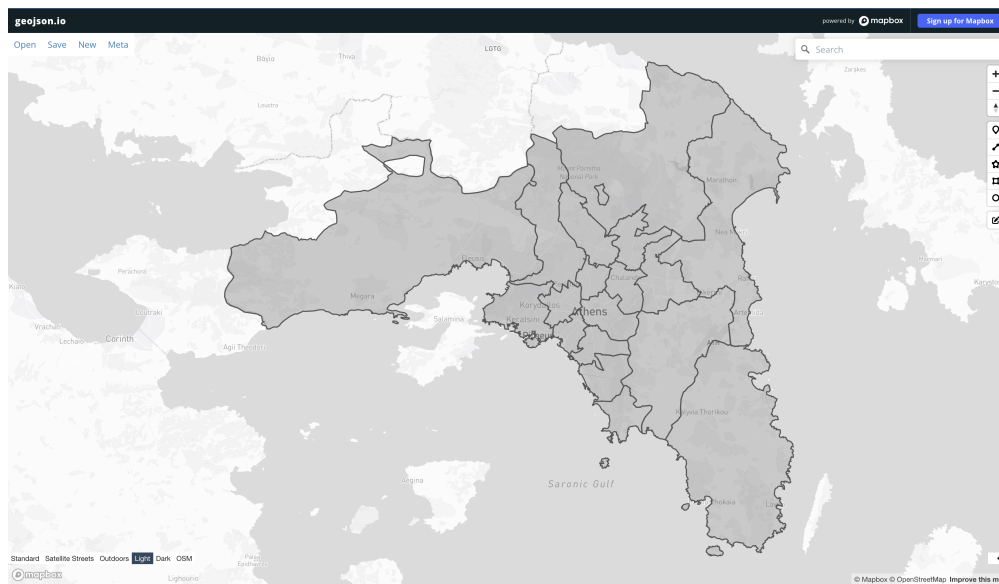
5.3 Αποτελέσματα του API

Το API επιστρέφει τα αποτελέσματα του αλγορίθμου συσταδοποίησης σε μορφή GeoJSON στην παράμετρο `clusters` και τα νέα `clusters` στην παράμετρο `piles`. Χρησιμοποιώντας την πλατφόρμα `geojson.io` [38] παραθέτουμε μια οπτικοποίηση των δύο `geojson` που απάντησε το λογισμικό μας για το παραπάνω `request` στα σχήματα 5.3.1 και 5.3.2.

Περισσότερα αποτελέσματα παρουσιάζονται στο κεφάλαιο 6.



Σχήμα 5.3.1: Αποτέλεσμα του API για τα clusters



Σχήμα 5.3.2: Αποτέλεσμα του API για τα piles

Κεφάλαιο 6

Αποτελέσματα και επέκταση

Στο κεφάλαιο αυτό παρουσιάζονται τα αποτελέσματα της εφαρμογής των αλγορίθμων συσταδοποίησης στα δεδομένα των tasks. Επίσης, παρουσιάζονται οι μετρικές απόδοσης των αλγορίθμων και προτάσεις για επέκταση της εργασίας.

Τα αποτελέσματα αφορούν τους αλγορίθμους με τις παρακάτω παραμέτρους:

Reference	Αλγόριθμος	n_clusters	init	max_iter	n_init	tol
KMeans 14 M	KMeans	18	k-means++	500	20	0.0001
KMeans 14 D	KMeans	18	k-means++	500	20	0.0001
KMeans 23 M	KMeans	18	k-means++	1000	100	0.0001
KMeans 23 D	KMeans	18	k-means++	1000	100	0.0001
MiniBatchKMeans 19 M	MiniBatchKMeans	18	k-means++	100	10	0
MiniBatchKMeans 19 D	MiniBatchKMeans	18	k-means++	100	10	0
KMeansConstrained 53 M	KMeansConstrained	18	k-means++	500	30	0.0001
KMeansConstrained 47 M	KMeansConstrained	18	k-means++	500	30	0.0001

Ο αλγόριθμος Old Solution αποτελεί την προηγούμενη λύση που χρησιμοποιούνταν.

Το M ή D στο τέλος του reference αναφέρετε στην συνάρτηση weighted metric του αλγορίθμου 8. Για το M, μας ενδιαφέρει μόνο ο χρόνος εκτέλεσης του task, ενώ για το D, μας ενδιαφέρει μόνο η απόσταση του task από το προηγούμενο.

Είναι επίσης, σημαντικό να πούμε ότι αλγόριθμοι εκπαιδεύτηκαν μόνο με το clustering_view_1. Το view 1b, είναι το ίδιο με το view 1, αλλά με την αφαίρεση των warehouse tasks.

6.1 Αποτελέσματα

Η ενότητα αυτή περιέχει τα αποτελέσματα των αλγορίθμων συσταδοποίησης και μια ανάλυση των αποτελεσμάτων.

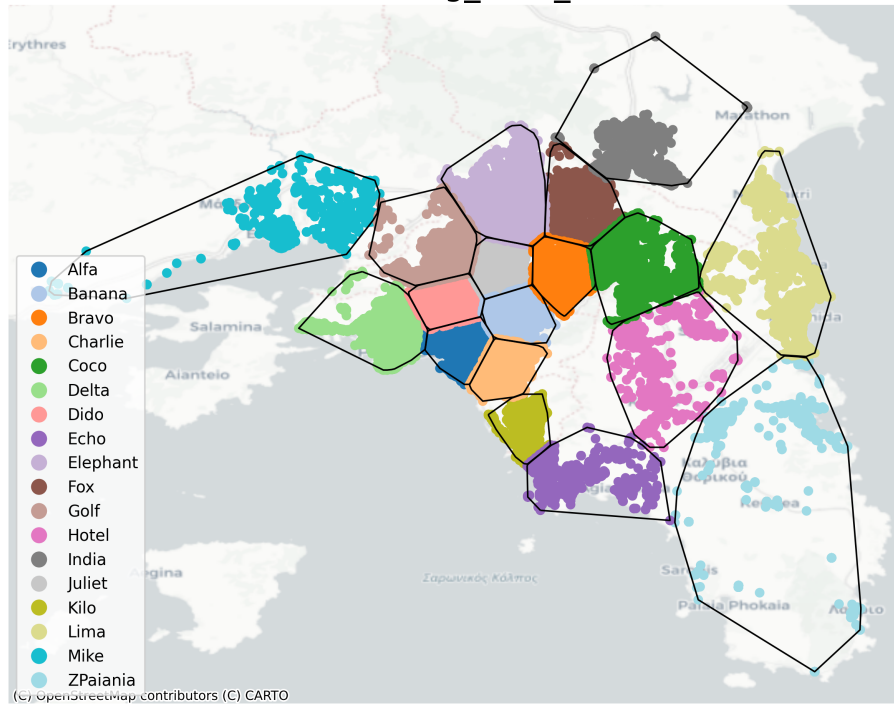
6.1.1 Συστάδες

Στις παρακάτω εικόνες παρουσιάζονται οι συστάδες που προέκυψαν από τους αλγορίθμους συσταδοποίησης. Η περιοχή κάθε συστάδας υπολογίζεται από το convex hull των σημείων της.

6.1.2 Πλήθος των tasks ανά συστάδα

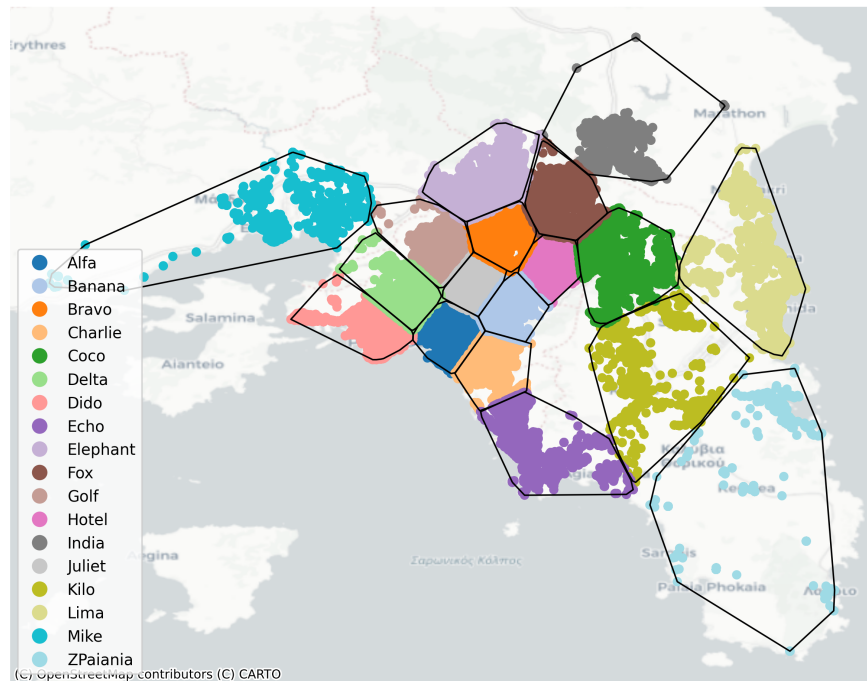
Στις παρακάτω εικόνες παρουσιάζεται το πλήθος των tasks ανά συστάδα.

KMeans Clustering
view: clustering_view_1, k=18



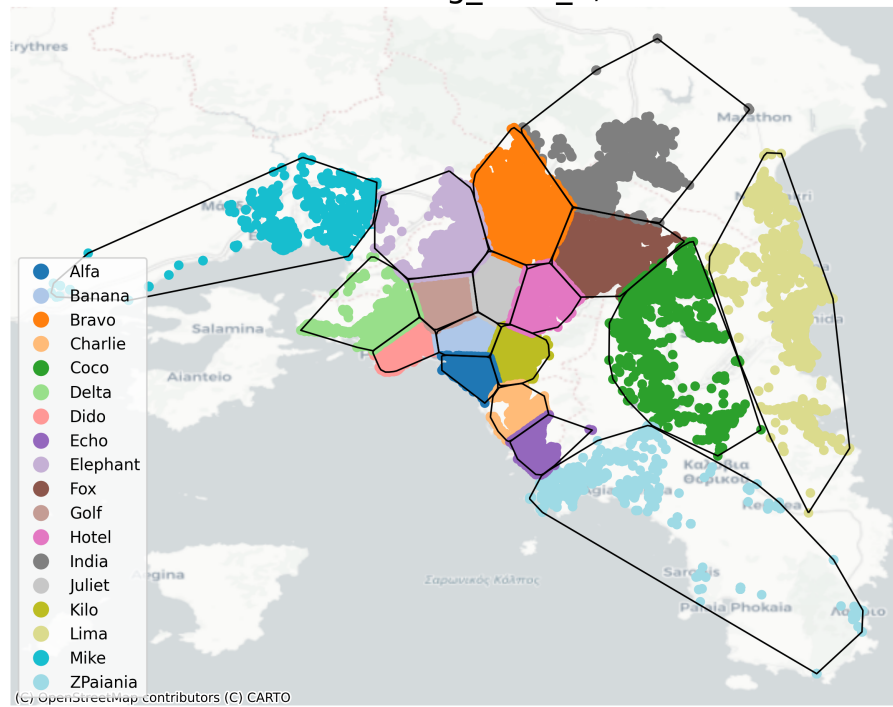
Σχήμα 6.1.1: KMeans Clustering

KMeans Clustering
view: clustering_view_1, k=18
Without Warehouse Tasks



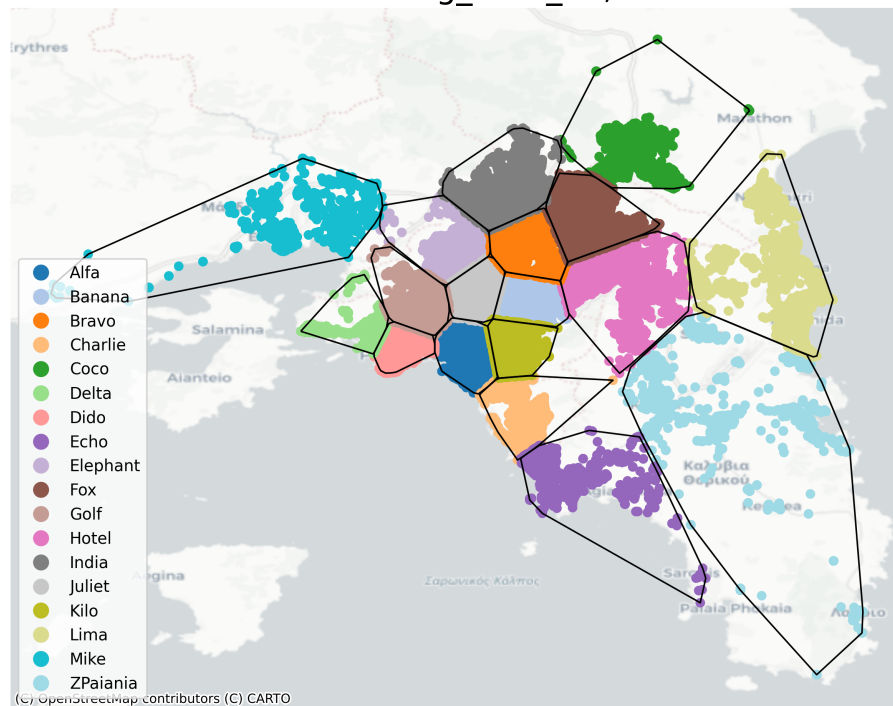
Σχήμα 6.1.2: KMeans Clustering without Warehouse Tasks

MiniBatchKMeans Clustering view: clustering_view_1, k=18



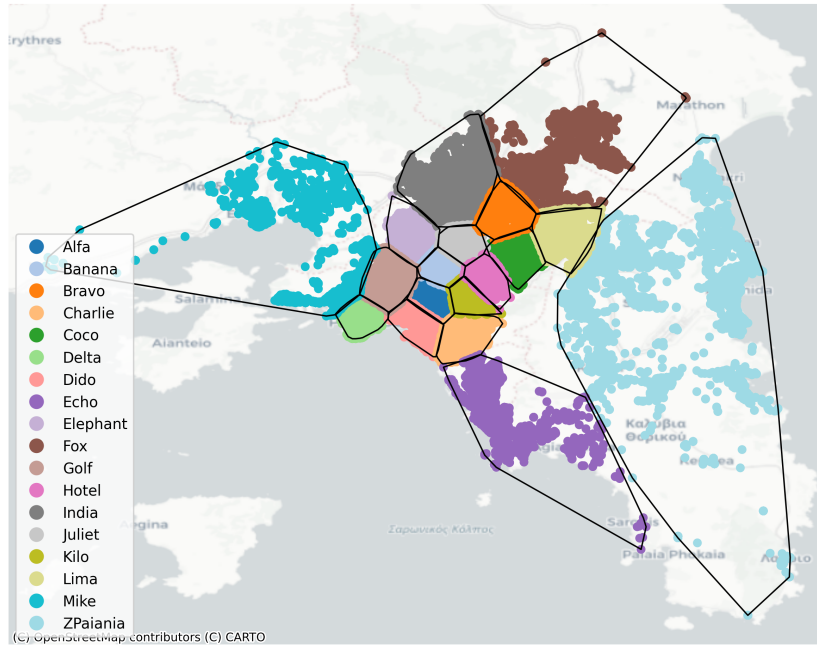
Σχήμα 6.1.3: Minibatch KMeans Clustering

MiniBatchKMeans Clustering view: clustering_view_1b, k=18



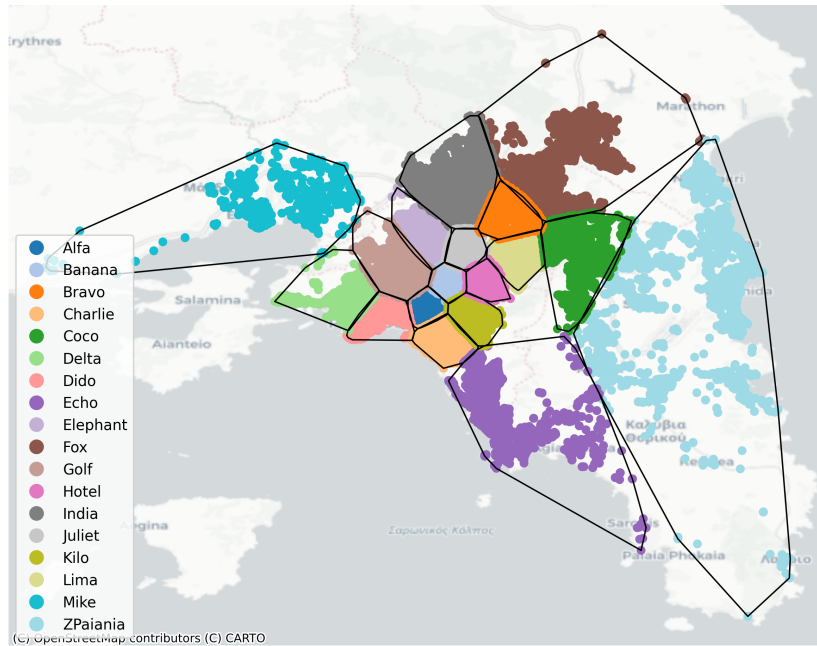
Σχήμα 6.1.4: Minibatch KMeans Clustering without Warehouse Tasks

KMeans Constrained Clustering
view: clustering_view_1b, k=18
Cluster sizes: Max: 9530, Min: 4000

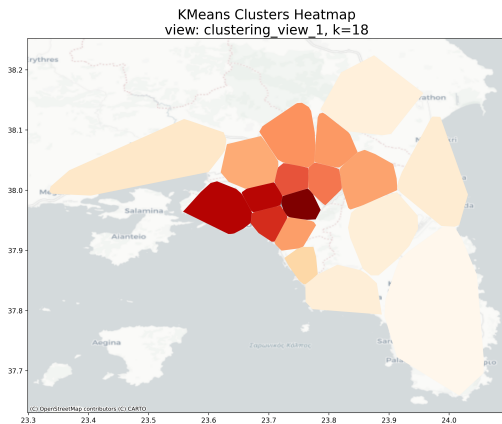


Σχήμα 6.1.5: Constrained KMeans Clustering

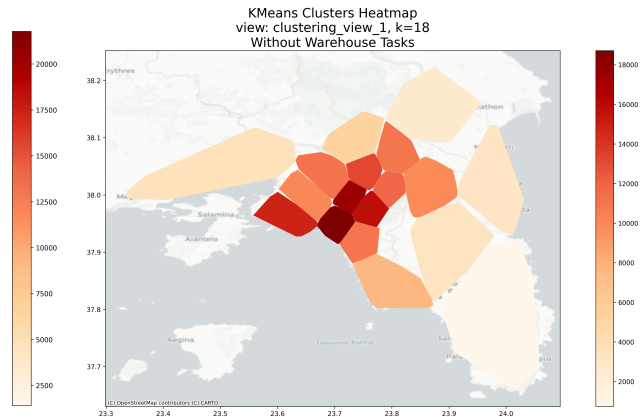
KMeans Constrained Clustering
view: clustering_view_1b, k=18
Without Warehouse Tasks
Cluster sizes: Max: 10046.0, Min: 3349.0



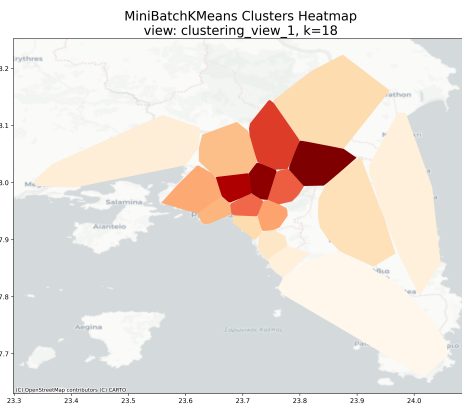
Σχήμα 6.1.6: Mean Constrained KMeans Clustering



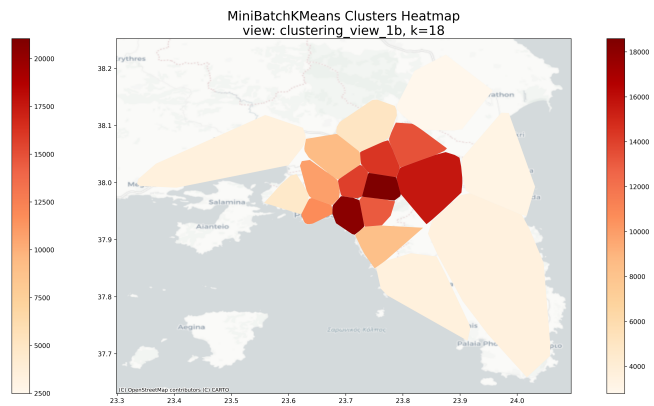
(a) KMeans Heatmap



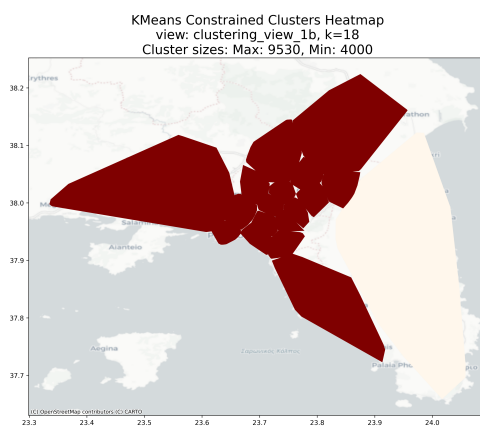
(b) KMeans without Warehouse Tasks - Heatmap



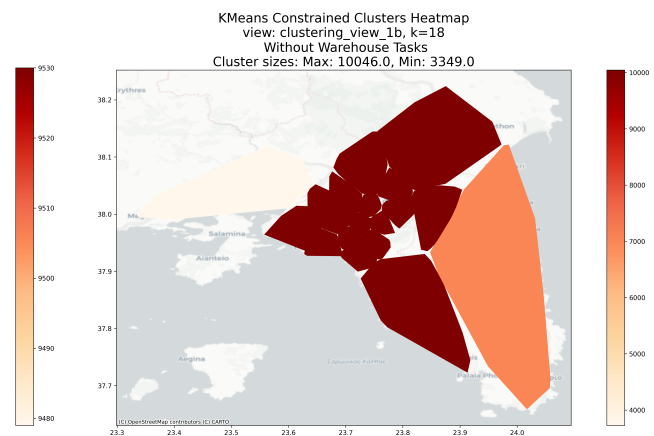
(c) Minibatch KMeans Heatmap



(d) Minibatch KMeans without Warehouse Tasks Heatmap

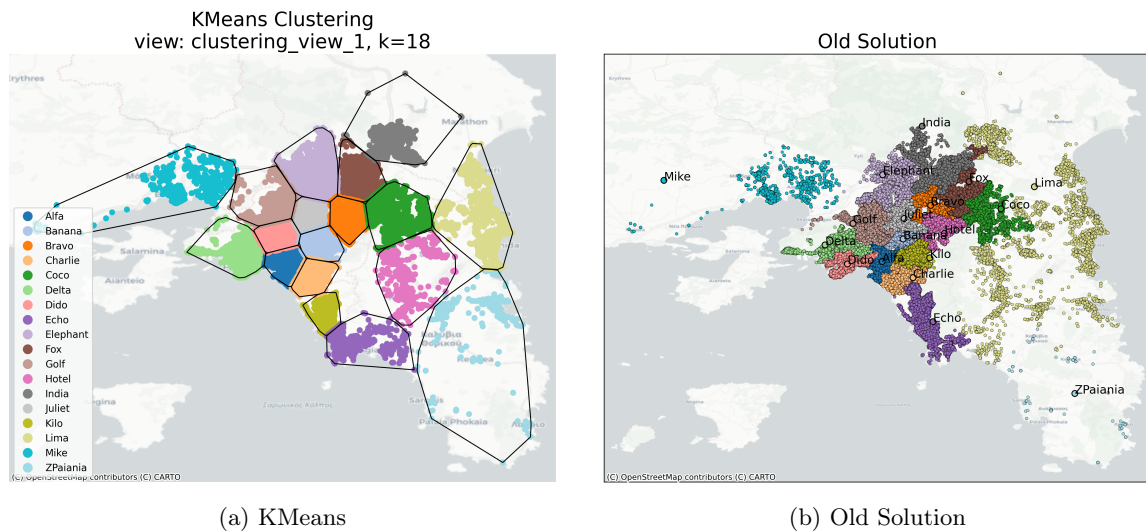


(e) Constrained KMeans Heatmap



(f) Constrained KMeans without Warehouse Tasks Heatmap

Σχήμα 6.1.7: Heatmaps



6.1.3 Διαφορές Συστάδων

Στις παρακάτω εικόνες παρουσιάζονται οι διαφορές των συστάδων μεταξύ των αλγορίθμων. Με πορφυρό χρώμα σημειώνονται τα σημεία που ανήκουν σε διαφορετικές συστάδες και με γκρι τα σημεία που ανήκουν στην ίδια συστάδα.

6.2 Αξιολογήση συστάδων

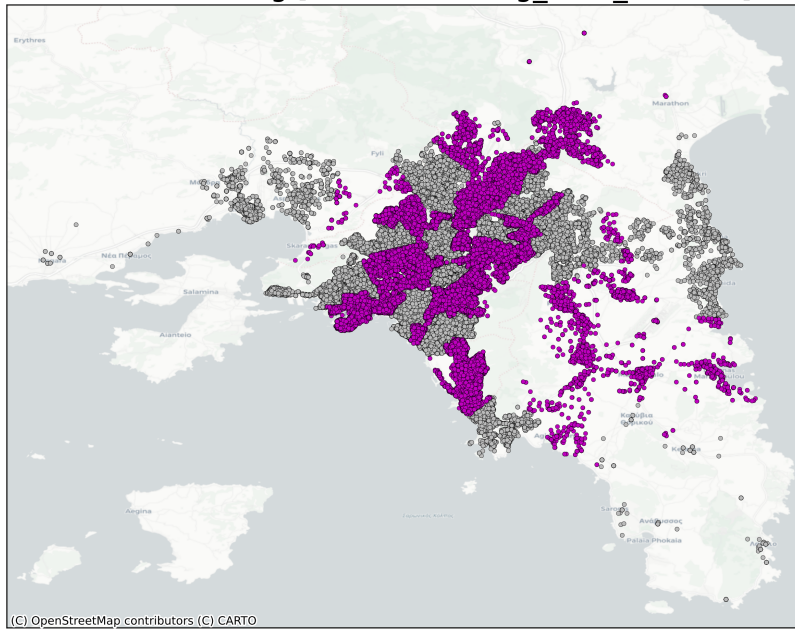
Στη μεθοδολογία αξιολόγησής μας, χρησιμοποιήσαμε ένα σύνολο δεδομένων που περιλαμβάνει 500 tasks ανά ταχυδρομικό κώδικα, φτάνοντας συνολικά σε 139.375 tasks στο σύνολο της βάσης δεδομένων. Παρατηρήθηκε ότι ορισμένοι ταχυδρομικοί κώδικες συνδέονταν με ανεπαρκή αριθμό εργασιών, κάτι που απαιτούσε τη χρήση όλων των διαθέσιμων εργασιών σε αυτές τις περιοχές για την ανάλυσή μας. Επιπλέον, αποκλείστηκαν από την αξιολόγηση tasks που σχετίζονταν με λειτουργίες αποθηκών, καθώς δεν συμβάδιζαν με τους στόχους συσταδοποίησης μας. Οι επιλεγμένες tasks για την αξιολόγησή μας απεικονίζονται στην εικόνα 6.2.1.

Ακολουθώντας την επιλογή των εργασιών, πραγματοποιήσαμε προσομοιώσεις δρομολογίων καθημερινά, καταλήγοντας σε συνολικά 12.138 δρομολόγια. Οι προσομοιώσεις αυτές διευκόλυναν μια εκτεταμένη αξιολόγηση των αποτελεσμάτων συσταδοποίησης, τα οποία συζητούνται αναλυτικά στις επόμενες ενότητες. Η αξιολόγησή μας επικεντρώνεται στην αποδοτικότητα και την ακρίβεια των αλγορίθμων συσταδοποίησης στην ομαδοποίηση των εργασιών, βελτιστοποιώντας έτσι τη διαδικασία δρομολόγησης. Τα ευρήματα από την ανάλυσή μας είναι καθοριστικά για την κατανόηση των επιπτώσεων των διάφορων στρατηγικών συσταδοποίησης στην επιχειρησιακή αποδοτικότητα.

Αλγόριθμος	Μέση διάρκεια	Διάμεση διάρκεια	Τυπική απόκλιση	Conf. Interval
KMeans 23 M	248.3	139.0	803.3	(244 , 252.7)
MiniBatchKMeans 19 M	234.6	138.5	636.0	(231.2 , 238.1)
KMeans 23 D	229.3	138.8	517.5	(226.5 , 232.1)
KMeans 14 D	227.5	138.8	515.7	(224.8 , 230.3)
MiniBatchKMeans 19 D	226.9	138.6	528.3	(224.0 , 229.7)
KMeans 14 M	225.9	138.9	507.4	(223.2 , 228.7)
Old Solution	223.6	139.3	468.6	(221.1 , 226.1)
KMeansConstrained 53 M	223.3	139.4	451.1	(220.9 , 225.8)
KMeansConstrained 47 M	222.8	139.6	430.3	(220.5 , 225.2)

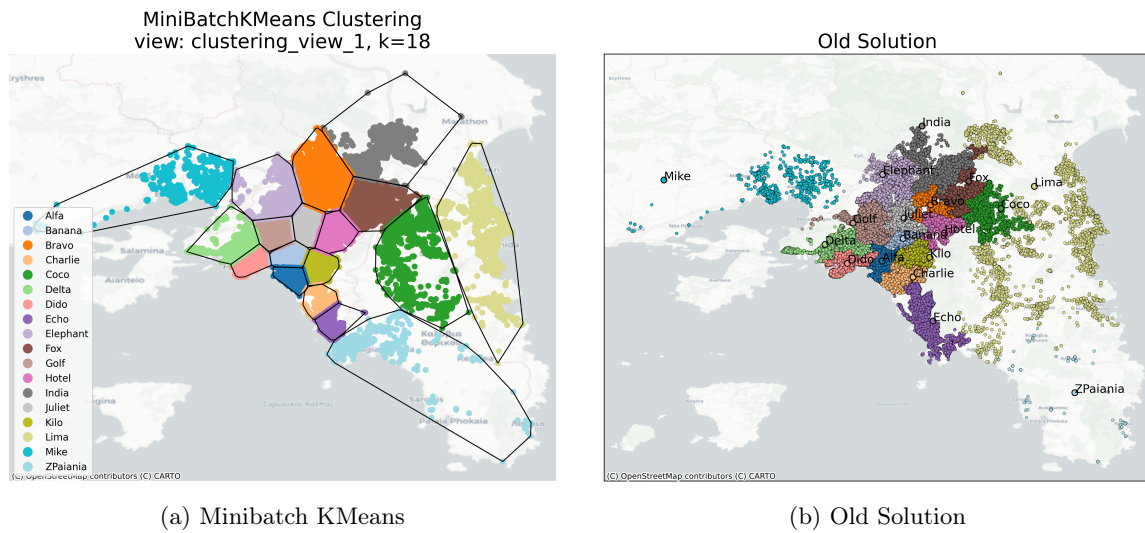
Πίνακας 6.2: Μετρικές απόδοσης διαφορετικών αλγορίθμων

Difference: KMeans Clustering [view: clustering_view_1, k=18] - Old Solution

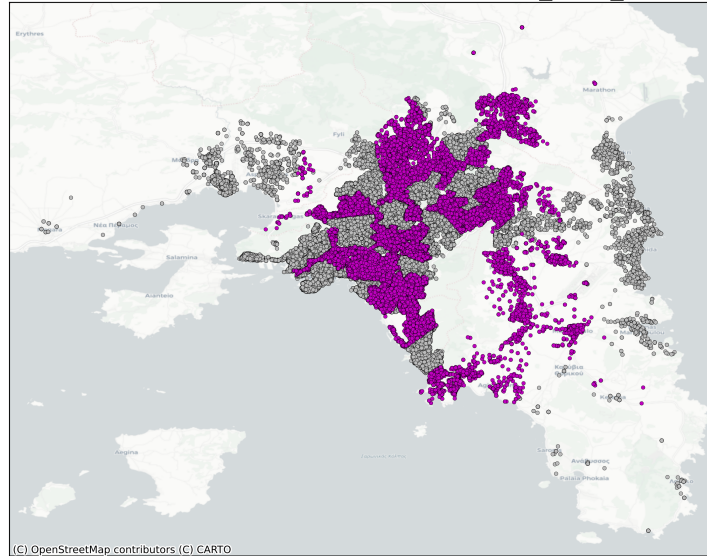


(c) Differences of KMeans and Old Solution

Σχήμα 6.1.8: Differences between KMeans and Old Solution



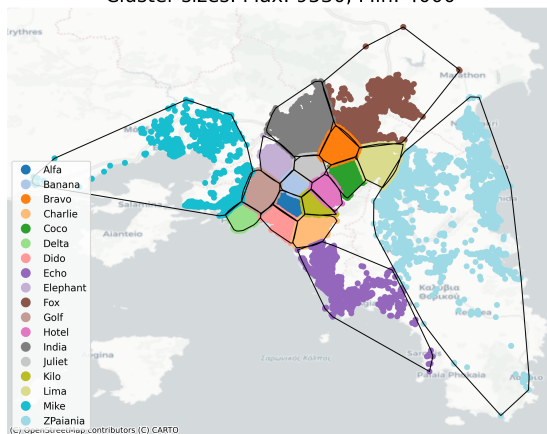
Difference: MiniBatchKMeans Clustering [view: clustering_view_1, k=18] - Old Solution



(c) Differences of Minibatch KMeans and Old Solution

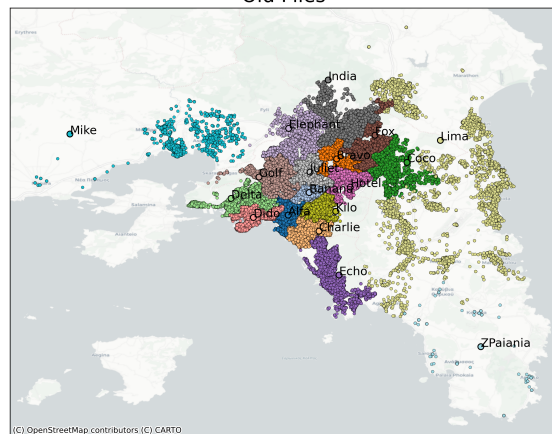
Σχήμα 6.1.9: Differences between Minibatch KMeans and Old Solution

KMeans Constrained Clustering
view: clustering_view_1b, k=18
Cluster sizes: Max: 9530, Min: 4000



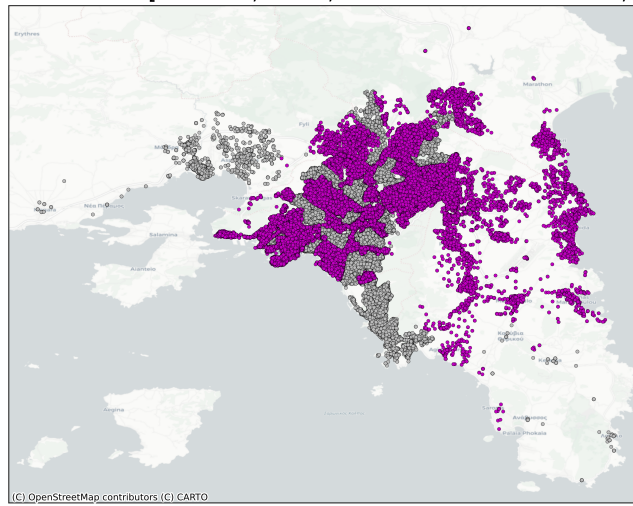
(a) Constrained KMeans

Old Piles



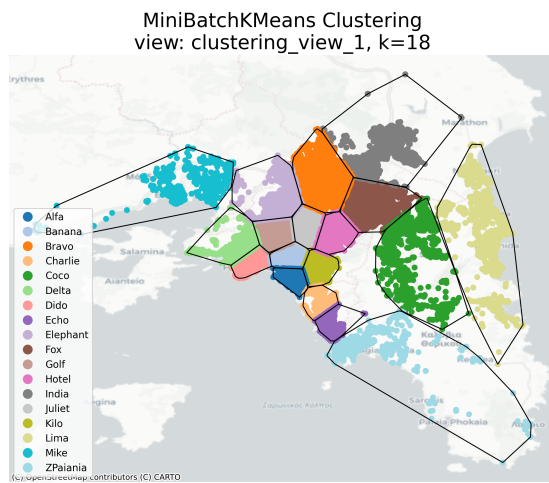
(b) Old Solution

Difference: KMeans Constrained [view: 1b, k=18, Cluster sizes: Max: 9530, Min: 4000] - Old Piles

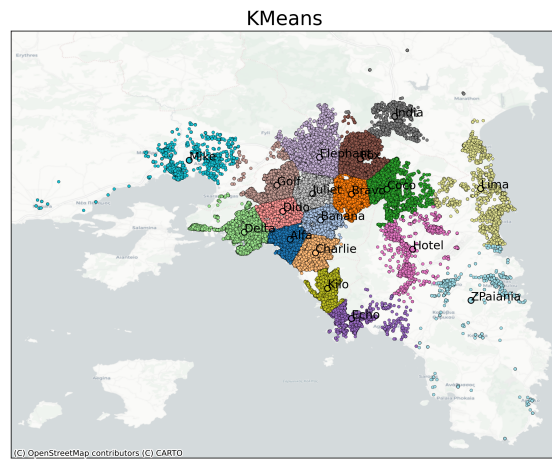


(c) Differences of Constrained KMeans and Old Solution

Σχήμα 6.1.10: Differences between Constrained KMeans and Old Solution

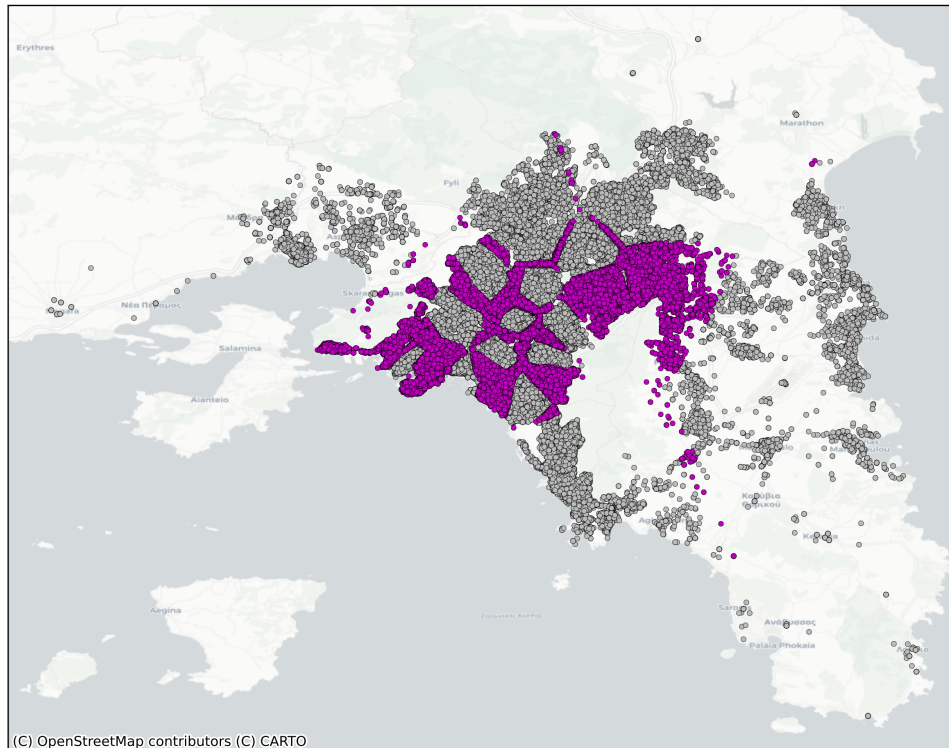


(a) KMeans



(b) Minibatch KMeans

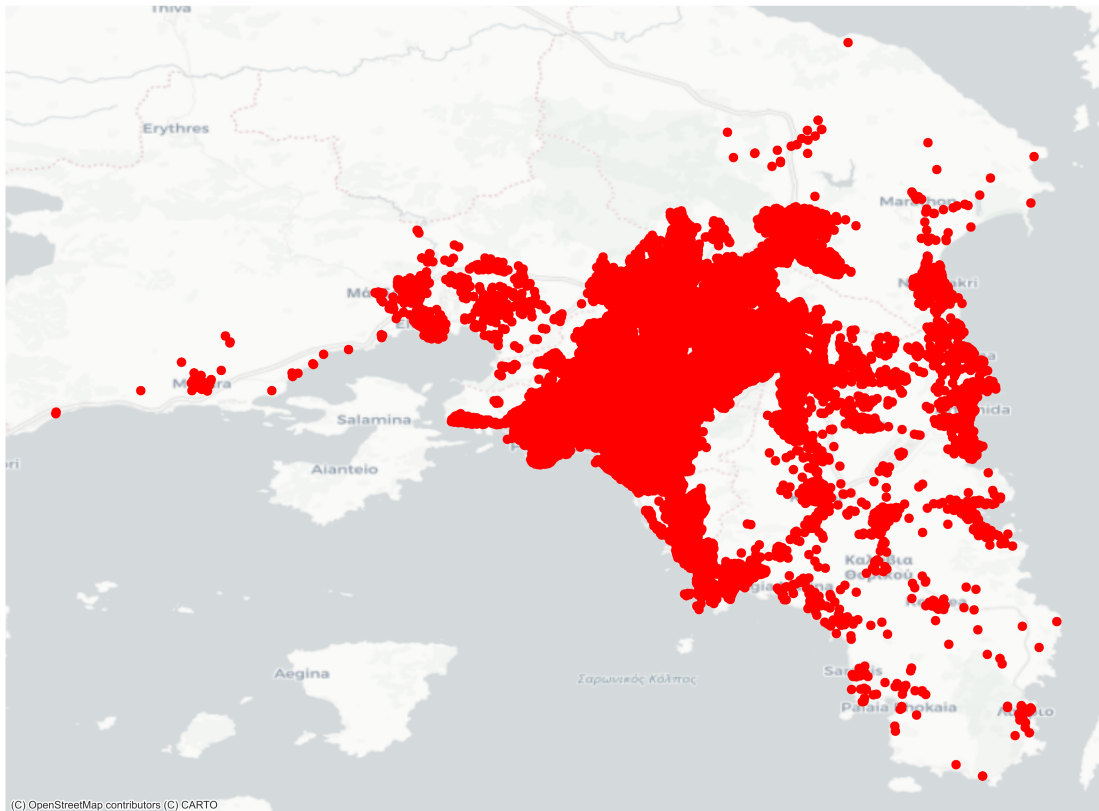
Difference: KMeans Constrained [view: 1b, k=18,
Cluster sizes: Max: 10046.0, Min: 3349.0] - KMeans Constrained



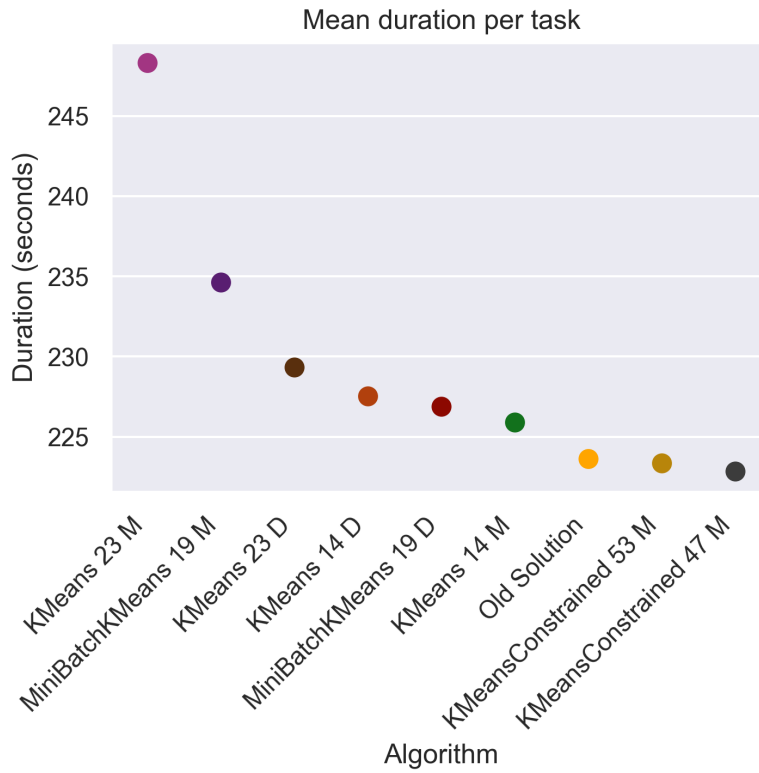
(c) Differences of Constrained KMeans and Constrained KMeans Mean

Σχήμα 6.1.12: Differences between Constrained KMeans and Constrained KMeans Mean

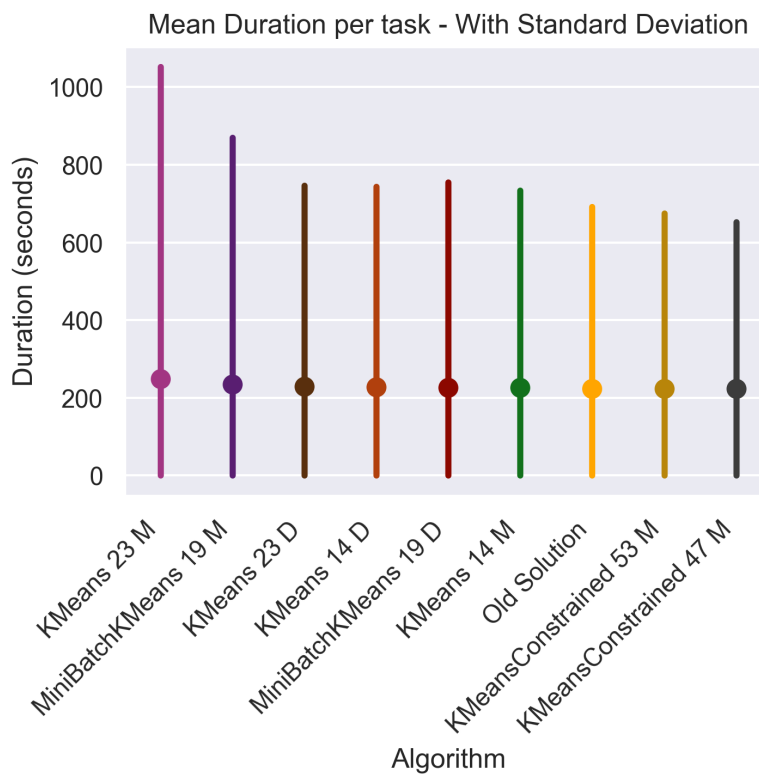
Tasks that used for evaluation



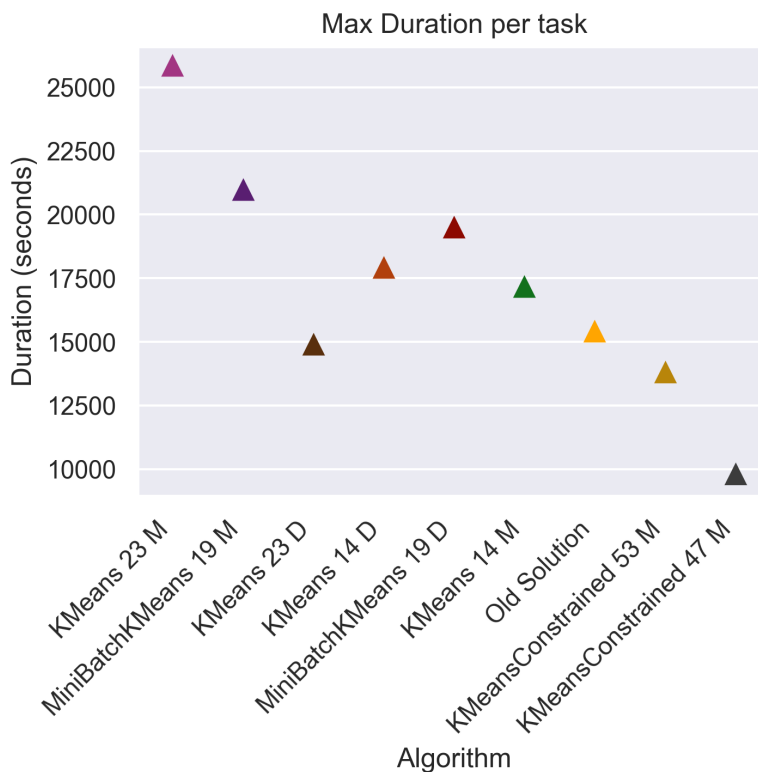
Σχήμα 6.2.1: Τα τυχαία tasks που επιλέχθηκαν για την αξιολογηση



Σχήμα 6.2.2: Μέση διάρκεια ανά task

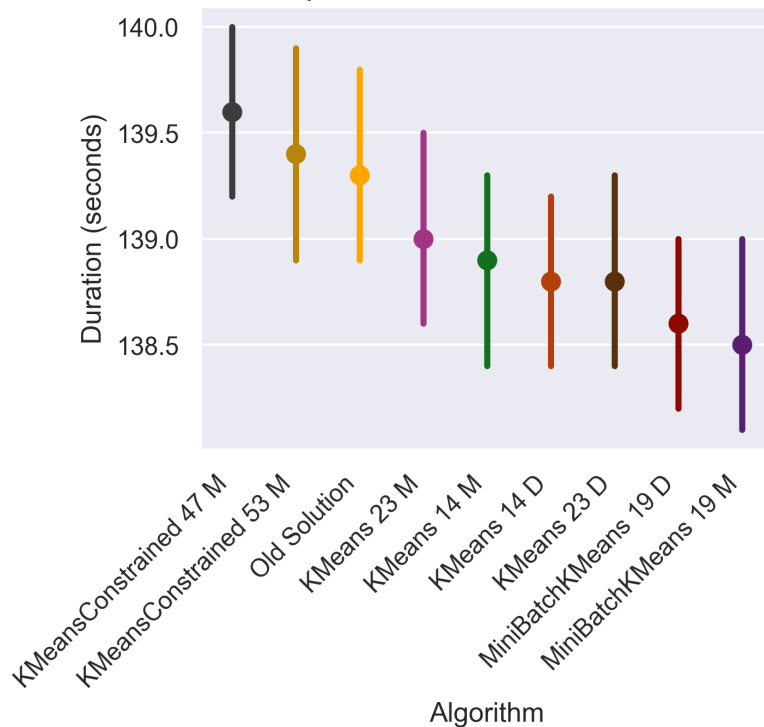


Σχήμα 6.2.3: Μέση διάρκεια ανά task με τυπική απόκλιση

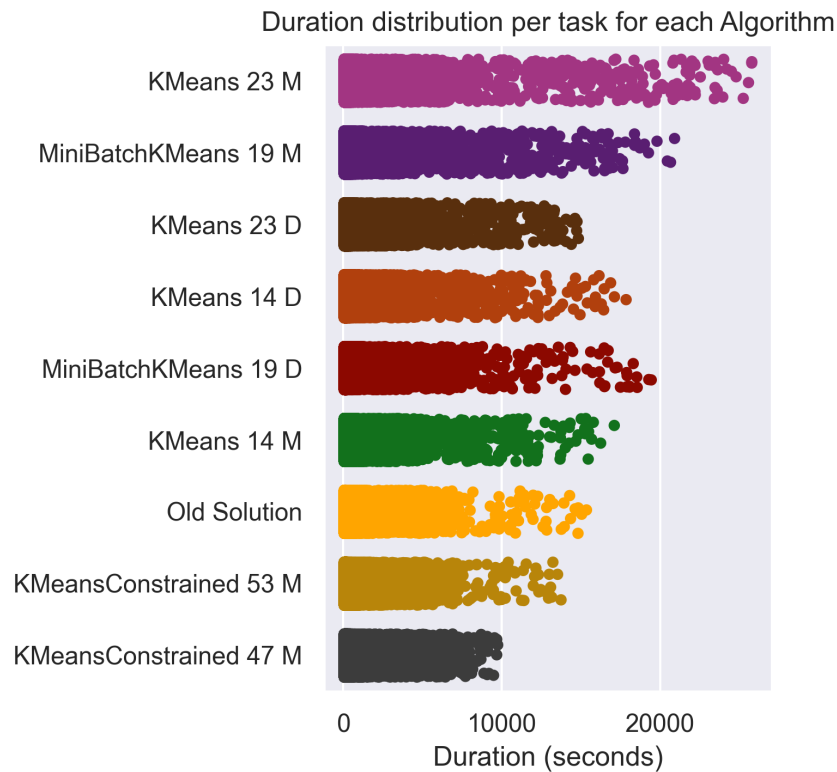


Σχήμα 6.2.4: Μέγιστη διάρκεια ανά task

Median Duration per task - With 95% Confidence Interval



Σχήμα 6.2.5: Διάμεση διάρκεια ανά task



Σχήμα 6.2.6: Κατανομή δεδομένων ανά task

6.3 Επέκταση

Ως πιθανές διευρύνσεις της παρούσας εργασίας, προτείνουμε την εξέταση μιας σειράς παραμέτρων που επηρεάζουν την αποδοτικότητα και την ακρίβεια των υφιστάμενων αλγορίθμων συσταδοποίησης, καθώς και την εφαρμογή και αξιολόγηση νέων ή εναλλακτικών αλγορίθμων. Επιπλέον, η εξέταση της επίδρασης της τροποποίησης των δεδομένων εισόδου, όπως η αφαίρεση ή η τροποποίηση δεδομένων με παρόμοιες ή κοντινές γεωγραφικές τοποθεσίες, θα μπορούσε να προσφέρει περαιτέρω ενδείξεις για τη βελτιστοποίηση των μεθόδων συσταδοποίησης. Τέλος, η διερεύνηση της επιρροής της αλλαγής γεωγραφικών ορίων, μέσω διαφορετικών διαμερίσεων, θα μπορούσε να παράσχει σημαντικές προοπτικές σχετικά με την κλιμακωσιμότητα και την προσαρμοστικότητα των αλγορίθμων.

Ένας ενδιαφέρων τρόπος για την επέκταση της εργασίας αυτής είναι η ανάπτυξη της ως μιας λύσης Software as a Service (SaaS), η οποία θα παρέχει στους χρήστες τη δυνατότητα να ενσωματώνουν αλγορίθμους που συμμορφώνονται με το API που προσφέρουμε. Με αυτό τον τρόπο, οι χρήστες-πάροχοι θα μπορούν να προσφέρουν τους δικούς τους αλγορίθμους, ενώ οι τελικοί χρήστες (καταναλωτές) θα έχουν την ελευθερία να επιλέξουν ποιους αλγορίθμους θα χρησιμοποιήσουν για την επεξεργασία των γεωχωρικών τους δεδομένων. Αυτό το μοντέλο υλοποίησης ενθαρρύνει την καινοτομία και την συνεργασία μεταξύ των χρηστών, προσφέροντας μια πλατφόρμα που συνδυάζει την ευελιξία και την ευρεία γκάμα εφαρμογών.

Η υλοποίηση της παραπάνω πρότασης μπορεί να γίνει μέσω του προτύπου Web Processing Service (OGC WPS) του Open Geospatial Consortium. Το WPS αποτελεί ένα πρότυπο που ορίζει τους κανόνες για την τυποποίηση των requests και των responses (αιτήσεις και απαντήσεις) για την κλήση υπηρεσιών επεξεργασίας γεωχωρικών δεδομένων ως μια υπηρεσία web, όπως η επικάλυψη πολυγώνων. Κύρια σημεία σχετικά με το WPS περιλαμβάνουν την ορισμό του τρόπου με τον οποίο ένας χρήστης μπορεί να ζητήσει την εκτέλεση μιας γεωχωρικής διαδικασίας και τον χειρισμό του αποτελέσματος από αυτήν τη διαδικασία. Επιτρέπει τη δημοσίευση, την ανεύρεση και τη δέσμευση γεωχωρικών διαδικασιών με έναν τυποποιημένο

και διαλειτουργικό τρόπο, υποστηρίζει πολλαπλές μορφές εισόδου και εξόδου και διαδικασίες μεγάλης διάρκειας, και διευκολύνει την εκτέλεση υπολογιστικών διαδικασιών που συνήθως συνδυάζουν ραστερικά, διανυσματικά, και/ή καλύμματα δεδομένων για την παραγωγή νέων ραστερικών, διανυσματικών, και/ή καλύμματα δεδομένων. Με την εφαρμογή του WPS στην προτεινόμενη SaaS λύση, οι χρήστες θα μπορούν να εκτελούν αλγορίθμους επεξεργασίας δεδομένων απομακρυσμένα, επιτρέποντας την αξιοποίηση των γεωχωρικών τους δεδομένων με μεγαλύτερη αποτελεσματικότητα και ευελιξία [39], [40].

Βιβλιογραφία

- [1] M. Ostermeier, A. Heimfarth, and A. Hübner, “Cost-optimal truck-and-robot routing for last-mile delivery,” en, *Networks*, vol. 79, no. 3, pp. 364–389, 2022. DOI: [10.1002/net.22030](https://doi.org/10.1002/net.22030). [Online]. Available:
- [2] L. Ranieri, S. Digiesi, B. Silvestri, and M. Roccotelli, “A review of last mile logistics innovations in an externalities cost reduction vision,” en, *Sustainability*, vol. 10, no. 3, p. 782, Mar. 2018. DOI: [10.3390/su10030782](https://doi.org/10.3390/su10030782). [Online]. Available:
- [3] L. G. Marujo, G. V. Goes, M. A. D’Agosto, A. F. Ferreira, M. Winkenbach, and R. A. Bandeira, “Assessing the sustainability of mobile depots: The case of urban freight distribution in rio de janeiro,” en, *Transportation Research Part D: Transport and Environment*, vol. 62, pp. 256–267, 2018. DOI: [10.1016/j.trd.2018.02.022](https://doi.org/10.1016/j.trd.2018.02.022). [Online]. Available:
- [4] B. Kin, J. Spoor, S. Verlinde, C. Macharis, and T. Van Woensel, “Modelling alternative distribution set-ups for fragmented last mile transport: Towards more efficient and sustainable urban freight transport,” en, *Case Studies on Transport Policy*, vol. 6, no. 1, pp. 125–132, 2018. DOI: [10.1016/j.cstp.2017.11.009](https://doi.org/10.1016/j.cstp.2017.11.009). [Online]. Available:
- [5] M. Jahangiriesmaili, S. Bahrami, and M. J. Roorda, “Solution of two-echelon facility location problems by approximation methods,” en, *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2610, no. 1, pp. 1–9, 2017. DOI: [10.3141/2610-01](https://doi.org/10.3141/2610-01). [Online]. Available:
- [6] Ν. Γκόλια, “Διερεύνηση Χρήσης Τεχνικών Μηχανικής Μάθησης για Ταξινόμηση (Classification) Βιογραφικών,” Mar. 18, 2021. [Online]. Available: (visited on 02/13/2024).
- [7] S. Lloyd, “Least squares quantization in pcm,” *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 1982. DOI: [10.1109/TIT.1982.1056489](https://doi.org/10.1109/TIT.1982.1056489).
- [8] Μ. Ξεφτέρης, “Αποδοτικοί αλγόριθμοι για την συσταδοποίηση χρονοσειρών μέσω προσέγγισης με gaussian processes,” en, Jul. 2021. [Online]. Available:
- [9] G. Schwartzman, “Mini-batch k -means terminates within $O(d/\epsilon)$ iterations,” no. arXiv:2304.00419, Apr. 2023, arXiv:2304.00419 [cs]. [Online]. Available:
- [10] D. Sculley, “Web-scale k -means clustering,” en, in *Proceedings of the 19th international conference on World wide web*, Raleigh North Carolina USA: ACM, Apr. 2010, pp. 1177–1178, ISBN: 9781605587998. DOI: [10.1145/1772690.1772862](https://doi.org/10.1145/1772690.1772862). [Online]. Available:
- [11] P. Bradley, K. Bennett, and A. Demiriz, “Constrained k -means clustering,” Aug. 2000.
- [12] *Routing machine*, <https://project-osrm.org/docs/v5.24.0/api/>. [Online]. Available:
- [13] *Interactive polyline encoder utility*, <https://developers.google.com/maps/documentation/utilities/polylineutility>. [Online]. Available:
- [14] *Encoded polyline algorithm format*, <https://developers.google.com/maps/documentation/utilities/polylinealgorithm>. [Online]. Available:
- [15] *Google maps api polyline string decoder*, <https://github.com/geodav-tech/decode-google-maps-polyline>. [Online]. Available:
- [16] E. W. Dijkstra, “A note on two problems in connexion with graphs,” en, *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, Dec. 1959. DOI: [10.1007/BF01386390](https://doi.org/10.1007/BF01386390). [Online]. Available:
- [17] D. Delling, A. V. Goldberg, T. Pajor, and R. F. Werneck, “Customizable route planning in road networks,” en, *Transportation Science*, vol. 51, no. 2, pp. 566–591, 2017. DOI: [10.1287/trsc.2014.0579](https://doi.org/10.1287/trsc.2014.0579). [Online]. Available:

- [18] M. L. Fisher and R. Jaikumar, “A generalized assignment heuristic for vehicle routing,” en, *Networks*, vol. 11, no. 2, 1981. DOI: [10.1002/net.3230110205](https://doi.org/10.1002/net.3230110205). [Online]. Available:
- [19] M. Gendreau, A. Hertz, and G. Laporte, “A tabu search heuristic for the vehicle routing problem,” en, *Management Science*, vol. 40, no. 10, 1994. DOI: [10.1287/mnsc.40.10.1276](https://doi.org/10.1287/mnsc.40.10.1276). [Online]. Available:
- [20] Y. Rochat and É. D. Taillard, “Probabilistic diversification and intensification in local search for vehicle routing,” en, *Journal of Heuristics*, vol. 1, no. 1, 1995. DOI: [10.1007/BF02430370](https://doi.org/10.1007/BF02430370). [Online]. Available:
- [21] J.-F. Cordeau, G. Laporte, and A. Mercier, “A unified tabu search heuristic for vehicle routing problems with time windows,” en, *Journal of the Operational Research Society*, vol. 52, no. 8, 2001. DOI: [10.1057/palgrave.jors.2601163](https://doi.org/10.1057/palgrave.jors.2601163). [Online]. Available:
- [22] O. Bräysy and M. Gendreau, “Vehicle routing problem with time windows, part ii: Metaheuristics,” en, *Transportation Science*, vol. 39, no. 1, 2005. DOI: [10.1287/trsc.1030.0057](https://doi.org/10.1287/trsc.1030.0057). [Online]. Available:
- [23] B.-I. Kim, S. Kim, and S. Sahoo, “Waste collection vehicle routing problem with time windows,” en, vol. 33, no. 12, 2006. DOI: [10.1016/j.cor.2005.02.045](https://doi.org/10.1016/j.cor.2005.02.045). [Online]. Available:
- [24] D. Pisinger and S. Ropke, “A general heuristic for vehicle routing problems,” en, vol. 34, no. 8, 2007. DOI: [10.1016/j.cor.2005.09.012](https://doi.org/10.1016/j.cor.2005.09.012). [Online]. Available:
- [25] *The Vehicle Routing Problem: Latest Advances and New Challenges* (Operations Research/Computer Science Interfaces), en. Boston MA: Springer US, vol. 43, ISBN: 9780387777771 9780387777788. DOI: [10.1007/978-0-387-77778-8](https://doi.org/10.1007/978-0-387-77778-8). [Online]. Available:
- [26] T. Schwarz, N. Hoenle, M. Grossmann, and D. Nicklas, “A library for managing spatial context using arbitrary coordinate systems,” in *IEEE Annual Conference on Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second*, Orlando, FL, USA: IEEE, 2004, pp. 48–52, ISBN: 9780769521060. DOI: [10.1109/PERCOMW.2004.1276904](https://doi.org/10.1109/PERCOMW.2004.1276904). [Online]. Available:
- [27] C. Zhang and W. Li, “The roles of web feature and web map services in real-time geospatial data sharing for time-critical applications,” en, *Cartography and Geographic Information Science*, vol. 32, no. 4, pp. 269–283, 2005. DOI: [10.1559/152304005775194728](https://doi.org/10.1559/152304005775194728). [Online]. Available:
- [28] J. Martinez-Llario, J. H. Weber-Jahnke, and E. Coll, “Improving dissolve spatial operations in a simple feature model,” en, *Advances in Engineering Software*, vol. 40, no. 3, pp. 170–175, 2009. DOI: [10.1016/j.advengsoft.2008.03.014](https://doi.org/10.1016/j.advengsoft.2008.03.014). [Online]. Available:
- [29] *Geospatial Web Services: Advances in Information Interoperability* (Advances in Geospatial Technologies). IGI Global, 2011, ISBN: 9781609601928 9781609601942. DOI: [10.4018/978-1-60960-192-8](https://doi.org/10.4018/978-1-60960-192-8). [Online]. Available:
- [30] C. Strobl, “Dimensionally extended nine-intersection model (de-9im),” en, in *Encyclopedia of GIS*, S. Shekhar and H. Xiong, Eds. Boston, MA: Springer US, 2008, pp. 240–245, ISBN: 9780387308586 9780387359731. DOI: [10.1007/978-0-387-35973-1_298](https://doi.org/10.1007/978-0-387-35973-1_298). [Online]. Available:
- [31] Z. Xie, Z. Ye, and L. Wu, “A design for polygon overlay algorithm in the simple feature model,” in *2008 Seventh International Conference on Grid and Cooperative Computing*, 2008, pp. 680–685. DOI: [10.1109/GCC.2008.41](https://doi.org/10.1109/GCC.2008.41).
- [32] J. Fan, C. Zhou, T. Ma, M. Ji, Y. Zhou, and T. Xu, “Dwsi: An approach to solving the polygon intersection-spreading problem with a parallel union algorithm at the feature layer level,” *Boletim de Ciências Geodésicas*, vol. 20, no. 1, pp. 159–182, 2014. DOI: [10.1590/s1982-21702014000100011](https://doi.org/10.1590/s1982-21702014000100011). [Online]. Available:
- [33] S. Stephen and T. Hahmann, “Formal qualitative spatial augmentation of the simple feature access model,” en, 18 pages, 537305 bytes, 2019. DOI: [10.4230/LIPICS.COSIT.2019.15](https://doi.org/10.4230/LIPICS.COSIT.2019.15). [Online]. Available:
- [34] R. Obe and L. Hsu, “Postgis in action,” 2011. [Online]. Available:
- [35] *Postgis documentation*, <https://postgis.net/docs/>. [Online]. Available:
- [36] *Volume 8: Ogc cdb spatial reference system guidance (best practice)*. [Online]. Available:
- [37] *Datagrip*, <https://www.jetbrains.com/datagrip/>. [Online]. Available:
- [38] *Geojson.io*, <https://geojson.io/>. [Online]. Available:
- [39] “Coupling ogc wps and w3c prov for provenance-aware geoprocessing workflows,” DOI: [10.1016/j.cageo.2020.104419](https://doi.org/10.1016/j.cageo.2020.104419). [Online]. Available:

- [40] “Simplifying the deployment of ogc web processing services wps for environmental modelling - introducing tethys wps server,” DOI: [10.1016/j.envsoft.2019.01.021](https://doi.org/10.1016/j.envsoft.2019.01.021). [Online]. Available: