



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

# Αναγνώριση Δορυφορικών Εικόνων και Αεροφωτογραφιών με Τεχνικές Μηχανικής Μάθησης

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΤΟΥ

**ΒΑΜΒΟΥΚΑΚΗ Δ. ΓΕΩΡΓΙΟΥ**

**Επιβλέπων:** Γεώργιος Στάμου  
Καθηγητής Ε.Μ.Π.

Αθήνα, Νοέμβριος 2023

---





ΕΘΝΙΚΟ ΜΕΤΕΩΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ  
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

# Αναγνώριση Δορυφορικών Εικόνων και Αεροφωτογραφιών με Τεχνικές Μηχανικής Μάθησης

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΤΟΥ

**ΒΑΜΒΟΥΚΑΚΗ Δ. ΓΕΩΡΓΙΟΥ**

**Επιβλέπων:** Γεώργιος Στάμου  
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 10η Νοεμβρίου 2023.

(Υπογραφή)

(Υπογραφή)

(Υπογραφή)

.....  
Γεώργιος Στάμου  
Καθηγητής Ε.Μ.Π.

.....  
Αθανάσιος Βουλόδημος  
Επικουρος Καθηγητής Ε.Μ.Π.

.....  
Στέφανος Κόλλιας  
Καθηγητής Ε.Μ.Π.

Αθήνα, Νοέμβριος 2023



(Υπογραφή)

.....  
**Γεώργιος Βαμβουκάκης**

Copyright © - All rights reserved. Με την επιφύλαξη παντός δικαιώματος.  
Γεώργιος Βαμβουκάκης, 2023.

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.

## Περίληψη

---

Η αναγνώριση και χαρτογράφηση της ξυλώδους βλάστησης σε αεροφωτογραφίες και δορυφορικές εικόνες είναι απαραίτητη για ποικίλλες εφαρμογές, όπως ο σχεδιασμός χρήσεων γης, η διαχείριση πόρων και η παρακολούθηση οικοσυστημάτων. Ωστόσο, η χειροκίνητη αναγνώριση και χαρτογράφηση της βλάστησης απαιτεί πολύ χρόνο, είναι εντάσεως εργασίας και είναι επιρρεπής σε λάθη. Στην παρούσα εργασία, εξετάζεται το θέμα της αναγνώρισης της βλάστησης με τη χρήση μηχανικής μάθησης (ML). Στόχος της παρούσας μελέτης είναι να παρέχει μια αποτελεσματική και ακριβή προσέγγιση για την εφαρμογή αλγορίθμων μηχανικής μάθησης για την αναγνώριση και χαρτογράφηση ξυλώδους και μη ξυλώδους βλάστησης σε αεροφωτογραφίες και δορυφορικές φωτογραφίες. Για να γίνει αυτό, συγκεντρώθηκε, επισημάνθηκε και επεξεργάστηκε ένα σημαντικό σύνολο δεδομένων από δορυφορικές εικόνες και αεροφωτογραφίες με τα οποία τροφοδοτήθηκαν οι αλγόριθμοι μηχανικής μάθησης. Δοκιμάστηκαν πολλαπλοί αλγόριθμοι ML και συγκρίθηκαν ως προς την ακρίβεια και την απόδοση και ο καλύτερος αλγόριθμος επιλέχθηκε για περαιτέρω ανάλυση. Τα ευρήματα της μελέτης δείχνουν πόσο καλοί είναι οι αλγόριθμοι μηχανικής μάθησης στην ακριβή ανίχνευση και χαρτογράφηση της βλάστησης με φτηνό υπολογιστικό κόστος για τον υπολογιστή. Η τεχνική που δημιουργήθηκε σε αυτή τη μελέτη μπορεί να προσφέρει μια πρακτική και αποτελεσματική απάντηση στο ζήτημα του εντοπισμού της βλάστησης σε αεροφωτογραφίες και δορυφορικές εικόνες. Η μελέτη αυτή προσθέτει στην ήδη υπάρχουσα γνώση, παρουσιάζοντας τη χρήση αλγορίθμων μηχανικής μάθησης στον τομέα της αναγνώρισης και χαρτογράφησης της βλάστησης. Επίσης επισημαίνεται η δυνατότητα που έχουν οι αλγόριθμοι ML να αλλάξουν ριζικά τον τρόπο με τον οποίο παρακολουθούμε και διαχειριζόμαστε τους φυσικούς μας πόρους. Τέλος, παρουσιάζεται μια πρότυπη εφαρμογή για την περαιτέρω βελτίωση ή αύξηση των δεδομένων εκπαίδευσης που μπορεί να οδηγήσει σε βελτιωμένα μοντέλα.

### Λέξεις Κλειδιά

αναγνώριση βλάστησης, μηχανική μάθηση, αεροφωτογραφίες, δορυφορικές εικόνες, U-Net, αναγνώριση εικόνας



## **Abstract**

---

The identification and mapping of woody vegetation in aerial photographs and satellite images is essential for a variety of applications, such as land use planning, resource management and ecosystem monitoring. However, manual identification and mapping of vegetation is time-consuming, labor-intensive and prone to human error. In this work, the issue of vegetation recognition is addressed using machine learning (ML). The objective of this study is to provide an efficient and accurate approach to apply machine learning algorithms for the identification and mapping of woody and non-woody vegetation in aerial and satellite imagery. To do this, a significant data set of satellite imagery and aerial photographs was collected, annotated and processed to feed the machine learning algorithms. Multiple ML algorithms were tested and compared in terms of accuracy and performance and the best algorithm was selected for further analysis. The findings of the study show how good machine learning algorithms are in accurately detecting and mapping vegetation with cheap computational cost for the computer. The technique developed in this study can provide a practical and effective answer to the issue of vegetation detection in aerial and satellite imagery. This study adds to pre-existing knowledge by presenting the use of machine learning algorithms in the field of vegetation identification and mapping. It also highlights the potential for ML algorithms to radically change the way we monitor and manage our natural resources. Finally, a model application is presented for further refinement or augmentation of training data that can lead to improved models.

### **Keywords**

vegetation recognition, machine learning, aerial photographs, satellite images, UNet, image recognition





## Ευχαριστίες

---

Θα ήθελα καταρχήν να ευχαριστήσω τον καθηγητή Δρ. Γεώργιο Στάμου και τα μέλη του εργαστηρίου Συστημάτων Τεχνητής Νοημοσύνης και Μάθησης για την επίβλεψη αυτής της διπλωματικής εργασίας και για την ευκαιρία που μου δώθηκε για να εκπονήσω τη διπλωματική εργασία μου. Επίσης ευχαριστώ ιδιαίτερα τον Δρ. Ηλία Συμεονάκη για τη παροχή των δεδομένων που χρησιμοποιήθηκαν στην εργασία και τον Αντώνη Κορκοφίγκα για την καθοδήγησή του και την εξαιρετική συνεργασία που είχαμε. Ακόμα, ένα μεγάλο ευχαριστώ στους φίλους και συμφοιτητές μου Πέτρο, Φάνη, Γιώργο και Αλέξανδρο, στη φίλη μου Άννα και στους συναδέλφους Γιώργο και Αιμίλιο για την υποστήριξή τους. Τέλος, θα ήθελα να ευχαριστήσω τον παππού μου Κωνσταντίνο Βαμβουκάκη για την υποστήριξή του και τη μητέρα μου για την καθοδήγηση και την ηθική συμπαράσταση που μου προσέφεραν όλα αυτά τα χρόνια.

Αθήνα, Νοέμβριος 2023

*Γεώργιος Βαμβουκάκης*



# Περιεχόμενα

---

<b>Περίληψη</b>	<b>5</b>
<b>Abstract</b>	<b>7</b>
<b>Ευχαριστίες</b>	<b>9</b>
<b>1 Εισαγωγή</b>	<b>17</b>
1.1 Αντικείμενο της διπλωματικής . . . . .	18
1.2 Οργάνωση του τόμου . . . . .	19
<b>I Θεωρητικό Μέρος</b>	<b>21</b>
<b>2 Μηχανική Μάθηση</b>	<b>23</b>
2.1 Ορισμός Μηχανικής Μάθησης . . . . .	23
2.2 Είδη Μηχανικής Μάθησης . . . . .	24
2.2.1 Επιβλεπόμενη Μάθηση . . . . .	24
2.2.2 Μή-Επιβλεπόμενη Μάθηση . . . . .	25
2.2.3 Ενισχυτική Μάθηση . . . . .	27
<b>3 Μοντέλα Μηχανικής Μάθησης - Μετρικές</b>	<b>29</b>
3.1 Μοντέλα . . . . .	29
3.1.1 Δέντρα Απόφασης . . . . .	29
3.1.2 Τυχαίο Δάσος . . . . .	30
3.1.3 Adaptive Boosting . . . . .	31
3.1.4 Gradient Tree Boosting . . . . .	31
3.1.5 Αλγόριθμος K-Πλησιέστερων Γειτόνων . . . . .	32
3.2 Μετρικές . . . . .	34
3.2.1 Ακρίβεια . . . . .	34
3.2.2 Ακρίβεια Προσέγγισης και Ανάκληση . . . . .	34
3.2.3 Περιοχή Κάτω από την Καμπύλη ROC . . . . .	35
3.2.4 Μέσο Απόλυτο Σφάλμα και Μέσο Τετραγωνικό Σφάλμα . . . . .	35
3.2.5 R-Τετράγωνο . . . . .	35
<b>4 Βαθιά Μάθηση</b>	<b>37</b>
4.1 Τεχνητά Νευρωνικά Δίκτυα . . . . .	37
4.1.1 Είδη Επιπέδων στα Μοντέλα Βαθιάς Μάθησης . . . . .	37

4.1.2	Συναρτήσεις Ενεργοποίησης	38
4.2	Ο Αλγόριθμος της Οπισθοδιάδοσης	38
4.2.1	Ο Κανόνας της Αλυσίδας	39
4.2.2	Εμπρόσθια Διάδοση	39
4.2.3	Συνάρτηση Κόστους	39
4.2.4	Οπισθοδρομική Διάδοση	40
4.2.5	Ενημέρωση Βαρών και Προκαταλήψεων	40
4.3	Perceptron Πολλαπλών Επιπέδων	40
4.3.1	Αρχιτεκτονική MLP	41
4.3.2	Συναρτήσεις Ενεργοποίησης	42
4.4	U-Net	42
4.4.1	Αρχιτεκτονική του U-Net	42
<b>II Πρακτικό Μέρος</b>		<b>45</b>
<b>5</b>	<b>Υλοποίηση - Πειράματα</b>	<b>47</b>
5.1	Δεδομένα	47
5.2	Πλαίσια Εργασίας	48
5.3	Πειράματα	48
5.3.1	Δημιουργία Μασκών από Σημεία σε Αεροφωτογραφίες	48
5.3.2	Δημιουργία Βελτιωμένων Μασκών με U-Net	49
5.3.3	Εκπαίδευση Μοντέλου Πρόβλεψης Ποσοτών Βλάστησης από Δορυφορικές Εικόνες	51
<b>6</b>	<b>Εφαρμογή Οπτικοποίησης και Βελτίωσης Δεδομένων</b>	<b>57</b>
6.1	Στόχος	57
6.2	Κώδικας - Χαρακτηριστικά	57
6.2.1	Βιβλιοθήκες που χρησιμοποιήθηκαν	57
6.2.2	Κλάση AutoScrollbar	58
6.2.3	Κλάση CanvasImage	59
6.2.4	Η κλάση MainWindow	61
6.3	Παράδειγμα Χρήσης	62
<b>III Επίλογος</b>		<b>73</b>
<b>7</b>	<b>Επίλογος</b>	<b>75</b>
7.1	Συμπεράσματα	75
7.2	Μελλοντικές Εργασίες	76
<b>Βιβλιογραφία</b>		<b>80</b>

## Κατάλογος Σχημάτων

---

2.1	Σχηματική αναπαράσταση της επιβλεπόμενης μηχανικής μάθησης. . . . .	25
2.2	Συσταδοποίηση δεδομένων 2 διαστάσεων σε 3 συστάδες. . . . .	26
2.3	Σχηματική αναπαράσταση ενός αυτοκωδικοποιητή. . . . .	27
3.1	Σχηματική αναπαράσταση Δέντρου Απόφασης για ταξινόμηση φρούτων. . . .	30
3.2	Σχηματική αναπαράσταση του αλγόριθμου K-NN για δεδομένα με 2 χαρακτη- ριστικά. . . . .	33
4.1	Σχηματική αναπαράσταση ενός MLP με 2 κρυφά επίπεδα. . . . .	41
4.2	Σχηματική αναπαράσταση του U-Net. . . . .	43
5.2	Συνάρτηση απώλειας κατά την εκπαίδευση U-Net . . . . .	50
5.3	Συντελεστής ομοιότητας Dice κατά το στάδιο επαλήθευσης του U-Net . . . .	51
5.1	Αεροφωτογραφίες και οι παραγώμενες μάσκες τους . . . . .	53
5.4	Αεροφωτογραφίες και οι παραγώμενες μάσκες τους απο το αρχικό μοντέλο και το U-Net . . . . .	54
5.5	Προβληματικές μάσκες U-Net . . . . .	55
6.1	Παράθυρο εκκίνησης . . . . .	63
6.2	Παράθυρο εισαγωγής εικόνας . . . . .	64
6.3	Παράθυρο ανάγνωσης εικόνας και μάσκας . . . . .	65
6.4	Παράθυρο περιήγησης εικόνας . . . . .	66
6.5	Ζουμ σε μέρος της εικόνας . . . . .	67
6.6	Η μεγενθυμένη εικόνα και οι τιμές της μάσκας σε κάθε εικονοστοιχείο . . . .	68
6.7	Επιλογή και αλλαγή τιμή μάσκας σε ένα εικονοστοιχείο . . . . .	69
6.8	Επιλογή και αλλαγή τιμή μάσκας σε πολλά εικονοστοιχεία . . . . .	70
6.9	Αποθήκευση επεξεργασμένης μάσκας . . . . .	71



## Κατάλογος Πινάκων

---

5.1	Τα μοντέλα που δοκιμάστηκαν και οι επιδόσεις τους . . . . .	48
5.2	Μοντέλα για ποσοστά βλάστησης από μάσκες U-Net και οι μετρικές τους . . .	52
5.3	Μοντέλα για ποσοστά βλάστησης από φιλτραρισμένες μάσκες U-Net και οι μετρικές τους . . . . .	52





## Κεφάλαιο **1**

### Εισαγωγή

---

**Ε**να από τα σημαντικότερα στοιχεία των οικοσυστημάτων της Γης είναι η βλάστηση, η οποία είναι απαραίτητη για τη διατήρηση της ζωής στον πλανήτη και τον έλεγχο του κλίματος. Ένας από τους μεγαλύτερους κινδύνους για τη σταθερότητα και τη βιοποικιλότητα του πλανήτη είναι η κλιματική αλλαγή, η οποία προκαλείται από την απελευθέρωση αερίων του θερμοκηπίου. Ως αποδέκτης άνθρακα, η βλάστηση, ιδίως τα δάση, απορροφά διοξείδιο του άνθρακα από την ατμόσφαιρα και μειώνει την ποσότητα του άνθρακα που μπορεί να προκαλέσει υπερθέρμανση του πλανήτη. Επιπλέον, η βλάστηση είναι απαραίτητη για την παρακολούθηση των πόρων, αφού προσφέρει βιότοπο για την άγρια ζωή, καθώς και τροφή, καταφύγιο και πόρους διαβίωσης για τον άνθρωπο.

Είναι ευρέως γνωστό ότι η βλάστηση επηρεάζει τη ρύθμιση του κλίματος. Για παράδειγμα, τα δάση αποθηκεύουν διοξείδιο του άνθρακα στη βιομάζα και το έδαφός τους απορροφώντας το από την ατμόσφαιρα. Όμως η αποψίλωση και η καταστροφή των δασών επιστρέφουν αυτόν τον άνθρακα στην ατμόσφαιρα, επιταχύνοντας την υπερθέρμανση του πλανήτη. Επιπλέον, ένα σημαντικό μέρος του οξυγόνου που απαιτείται για την υποστήριξη της ζωής στη Γη παράγεται από τα δέντρα. Τα δάση βοηθούν στη μείωση της ποσότητας των αερίων του θερμοκηπίου στην ατμόσφαιρα, τα οποία διαφορετικά επιδεινώνουν την υπερθέρμανση του πλανήτη, απορροφώντας και αποθηκεύοντας άνθρακα.

Επειδή παρέχει στον άνθρωπο πρόσβαση σε τροφή, καταφύγιο και βιοποριστικά μέσα, καθώς και χρησιμεύει ως βιότοπος για την άγρια ζωή, η παρακολούθηση της βλάστησης είναι απαραίτητη για την παρακολούθηση των πόρων μίας περιοχής. Για παράδειγμα, τα δάση αποτελούν προμήθεια ξύλου, το οποίο χρησιμοποιείται για την παραγωγή χαρτιού, δομικών υλικών και καυσίμων. Μια μεγάλη ποικιλία ειδών, πολλά από τα οποία είναι ευάλωτα ή απειλούμενα με εξαφάνιση, μπορούν επίσης να βρουν καταφύγιο σε αυτά. Επιπλέον, τα δάση προσφέρουν πόρους που περιλαμβάνουν μη ξυλώδη δασικά προϊόντα, όπως φαρμακευτικά βότανα και άλλα είδη που είναι απαραίτητα για την σύγχρονη οικονομία και την υγεία.

Η αξία της βλάστησης, ωστόσο, εκτείνεται πολύ πέρα από την ικανότητά της να ελέγχει τη θερμοκρασία και να υποστηρίζει τη ζωή όπως την ξέρουμε. Τα δέντρα απορροφούν και αποθηκεύουν νερό, μειώνοντας τον κίνδυνο πλημμυρών και παρέχοντας νερό για άρδευση και άλλες χρήσεις. Αυτός είναι ο λόγος για τον οποίο η βλάστηση είναι τόσο σημαντική για τη διαχείριση των υδάτων. Επιπλέον, η βλάστηση βοηθάει σε κρίσιμες λειτουργίες ενός οικοσυστήματος, όπως ο έλεγχος των εντόμων και η επικονίαση, οι οποίες είναι ζωτικής σημασίας για την ανάπτυξη των καλλιεργειών τροφίμων.

Για την αποτελεσματική διαχείριση των φυσικών πόρων και την αποτροπή της κλιματικής αλλαγής, η παρακολούθηση της βλάστησης είναι ζωτικής σημασίας. Πρόσφατα έχει σημειωθεί σημαντική πρόοδος στην παρακολούθηση και χαρτογράφηση της βλάστησης, ιδίως σε απομακρυσμένες και δυσπρόσιτες περιοχές, με τη χρήση δορυφορικών και αεροφωτογραφιών. Πλέον, είναι εφικτή η ακριβής αναγνώριση και χαρτογράφηση της βλάστησης από δορυφορικές φωτογραφίες με τη χρήση αλγορίθμων μηχανικής μάθησης, προσφέροντας μια χαμηλού κόστους και αποτελεσματική λύση στο πρόβλημα της παρακολούθησης της βλάστησης.

## 1.1 Αντικείμενο της διπλωματικής

Αν και διαθέτουμε πολλά δεδομένα δορυφορικών εικόνων και αεροφωτογραφιών, η διαδικασία αναγνώρισης της βλάστησης είναι αρκετά χρονοβόρα. Επιπλέον, για τη συνεχόμενη παρακολούθηση της βλάστησης μίας περιοχής απαιτείται να λαμβάνουμε συνεχώς νέα δεδομένα. Από τα παραπάνω προκύπτουν τα εξής προβλήματα :

- Η λήψη αεροφωτογραφιών που έχουν μεγαλύτερη ανάλυση και χρησιμοποιούνται για την ανάλυση της βλάστησης από μοντέλα μηχανικής μάθησης είναι αρκετά ακριβή εφόσον απαιτείται να γίνεται ανά τακτά χρονικά διαστήματα.
- Οι δορυφορικές εικόνες είναι ευκολότερο να αποκτηθούν αλλά έχουν χαμηλότερη ανάλυση. Επιπλέον, η παρουσία ατμοσφαιρικών φαινομένων μπορούν να επηρεάσουν την ακρίβεια της αναγνώρισης της βλάστησης, καθιστώντας δύσκολη τη διάκριση μεταξύ της βλάστησης και άλλων τύπων κάλυψης εδάφους.
- Η απόκτηση δεδομένων για την εκπαίδευση μοντέλων μηχανικής μάθησης είναι πολύ δύσκολη, χρονοβόρα και απαιτεί την απασχόληση εμπειρών ειδικών.

Αντικείμενο της διπλωματικής είναι η ανάπτυξη ενός συστήματος που θα μπορεί να ανιχνεύει αυτόματα τα ποσοστά βλάστησης σε περιοχές που απεικονίζονται σε δορυφορικές εικόνες με πολύ χαμηλότερη ανάλυση σε σχέση με αεροφωτογραφίες. Αυτό επιτυγχάνεται, συνοπτικά, με τα παρακάτω βήματα :

- Αρχική εκπαίδευση ενός μοντέλου πάνω σε δεδομένα σημείων πάνω σε αεροφωτογραφίες.
- Δημιουργία "μασκών" βλάστησης για αεροφωτογραφίες.
- Εκπαίδευση ενός μοντέλου αρχιτεκτονικής τύπου U-Net [1] πάνω στις μάσκες που έχουν δημιουργηθεί.
- Υπολογισμός ποσοστών βλάστησης ανά περιοχή που απεικονίζεται σε δορυφορικές εικόνες βάσει των αντίστοιχων αεροφωτογραφιών με τη χρήση του εκπαιδευμένου U-Net.
- Εκπαίδευση ενός μοντέλου αποκλειστικά πάνω σε δορυφορικές εικόνες με στόχο να υπολογίζει τα ποσοστά βλάστησης από αυτές.

- Δημιουργία proof-of-concept εφαρμογής σε python [2] που επιτρέπει στον χρήστη την ευκολότερη επισήμανση σημείων πάνω σε εικόνες για δημιουργία επιπλέον δεδομένων εκπαίδευσης.

Ένα τέτοιο σύστημα θα μπορούσε να χρησιμοποιηθεί από ειδικούς που μελετάνε την μεταβολή της βλάστησης σε διάφορες περιοχές, εξοικονομώντας χρόνο αλλά και πόρους καθώς μειώνεται η ανάγκη λήψης αεροφωτογραφιών για αυτόν τον σκοπό. Επιπλέον, με το εργαλείο επισήμανσης σημείων μπορεί να βελτιωθεί η ποιότητα των δεδομένων βελτιώνοντας και τα ίδια τα μοντέλα αναγνώρισης. Τέλος, η τεχνική αυτή θα μπορούσε να χρησιμοποιηθεί και για άλλους τύπους κάλυψης εδάφους για τους οποίους έχουμε φυσικά επαρκή δεδομένα.

## 1.2 Οργάνωση του τόμου

Η εργασία αυτή είναι οργανωμένη σε 7 κεφάλαια: Στο Κεφάλαιο 2 δίνεται το θεωρητικό υπόβαθρο των βασικών εννοιών της Μηχανικής Μάθησης. Περιγράφονται τα είδη Μηχανικής Μάθησης καθώς και οι βασικές υποκατηγορίες τους. Στο Κεφάλαιο 3 περιγράφονται τα πιο γνωστά μοντέλα μηχανικής μάθησης τα οποία χρησιμοποιούμε και συγκρίνουμε στα πλαίσια της εργασίας καθώς και μετρικές που χρησιμοποιούνται για την αξιολόγηση των μοντέλων. Στο Κεφάλαιο 4 περιγράφονται τα μοντέλα βαθιάς μηχανικής μάθησης και πώς αυτά δομούνται. Στο Κεφάλαιο 5, περνάμε στο πρακτικό μέρος όπου βλέπουμε τι δεδομένα και τι μοντέλα χρησιμοποιήσαμε καθώς και τα αποτελέσματα αυτών. Εκπαιδεύουμε μοντέλα τόσο για τη δημιουργία μασκών, όσο και για τον υπολογισμό ποσοστών βλάστησης. Στο Κεφάλαιο 6, εξηγούμε τις λειτουργίες της εφαρμογής επισήμανσης καθώς και δείχνουμε παράδειγμα χρήσης της. Στο Κεφάλαιο 7, συνοψίζουμε τα αποτελέσματα της εργασίας και προτείνουμε πιθανές μελλοντικές επεκτάσεις της.



## **Μέρος I**

### **Θεωρητικό Μέρος**

---



## Κεφάλαιο 2

# Μηχανική Μάθηση

---

Στο κεφάλαιο αυτό παρουσιάζονται οι βασικές ιδέες της μηχανικής μάθησης, ενός κλάδου της επιστήμης των υπολογιστών που χρησιμοποιεί στατιστικές τεχνικές για να επιτρέπει στα υπολογιστικά συστήματα να μαθαίνουν από δεδομένα [3]. Εξετάζονται θέματα που είναι ζωτικής σημασίας για την κατανόηση εξελιγμένων συστημάτων μάθησης όπως τα νευρωνικά δίκτυα. Το κεφάλαιο αυτό ολοκληρώνεται με μια εισαγωγή στην εκπαίδευση και αξιολόγηση μοντέλων, ξεκινώντας με μια σύντομη περίληψη των πιο κρίσιμων αρχών της μηχανικής μάθησης.

### 2.1 Ορισμός Μηχανικής Μάθησης

Ο τομέας της τεχνητής νοημοσύνης που είναι γνωστός ως μηχανική μάθηση (ML) ασχολείται με τη δημιουργία μεθόδων που επιτρέπουν στους υπολογιστές να μαθαίνουν από δεδομένα, να αναγνωρίζουν μοτίβα και να κάνουν κρίσεις με ελάχιστη ή και καθόλου ανθρώπινη συμμετοχή. Ένας αλγόριθμος, ο οποίος είναι μια ακολουθία εντολών που χρησιμοποιούνται για τη μετατροπή της εισόδου σε έξοδο, είναι απαραίτητος για την επίλυση ενός υπολογιστικού προβλήματος [4]. Για παράδειγμα, ένας αλγόριθμος που δέχεται ένα σύνολο αριθμών ως είσοδο και παράγει μια ταξινομημένη λίστα μπορεί να χρησιμοποιηθεί για την ταξινόμηση μιας λίστας αριθμών. Μπορεί να υπάρχουν πολλαπλοί αλγόριθμοι για μια δεδομένη λειτουργία και ο στόχος μας είναι να επιλέξουμε αυτόν που χρησιμοποιεί τη λιγότερη μνήμη και έχει το μικρότερο υπολογιστικό κόστος.

Υπάρχουν όμως και άλλα ζητήματα για τα οποία δεν έχουμε αλγόριθμο, όπως η διάκριση των ανεπιθύμητων μηνυμάτων ηλεκτρονικού ταχυδρομείου (spam) από τα πραγματικά. Γνωρίζουμε την είσοδο, η οποία είναι ένα αρχείο ηλεκτρονικού ταχυδρομείου στην πιο βασική του μορφή, και την έξοδο, η οποία είναι μια απάντηση ναι ή όχι που δείχνει αν το μήνυμα ηλεκτρονικού ταχυδρομείου είναι ανεπιθύμητο. Ωστόσο, δεν είμαστε σίγουροι για τον τρόπο μετατροπής της εισόδου σε έξοδο. Το spam μπορεί να οριστεί διαφορετικά στο πέρασμα του χρόνου και από διαφορετικούς ανθρώπους. Για αυτό, χρησιμοποιούμε δεδομένα για να αντισταθμίσουμε την άγνοιά μας. Προκειμένου να "μάθουμε" τι συνιστά ανεπιθύμητο, μπορούμε απλώς να συγκεντρώσουμε χιλιάδες περιπτώσεις ηλεκτρονικού ταχυδρομείου, μερικές από τις οποίες χαρακτηρίζονται ως ανεπιθύμητες και άλλες όχι. Αυτό σημαίνει ότι θέλουμε η μέθοδος για το έργο αυτό να εξάγεται αυτόματα από τον υπολογιστή. Εφόσον έχουμε ήδη αλγόριθμους για την ταξινόμηση αριθμών, δεν χρειάζεται να τους μάθουμε, αλλά υπάρχουν

πολλές ακόμη δραστηριότητες για τις οποίες δεν έχουμε μεθόδους αλλά διαθέτουμε πληθώρα δεδομένων [5].

Η μηχανική μάθηση επομένως είναι ο προγραμματισμός υπολογιστών για τη μεγιστοποίηση ενός κριτηρίου απόδοσης χρησιμοποιώντας παραδείγματα ή προηγούμενη γνώση. Για να επιτευχθεί αυτό, απαιτείται ένα μοντέλο με παραμέτρους που ρυθμίζονται χρησιμοποιώντας δεδομένα εκπαίδευσης ή προηγούμενη γνώση. Το μοντέλο μπορεί να εφαρμοστεί για να κάνει προβλέψεις, να μαθαίνει από δεδομένα ή να κάνει και τα δύο. Χρησιμοποιεί τις αρχές της στατιστικής για τη δημιουργία υπολογιστικών μοντέλων και απαιτείται η δημιουργία αποτελεσματικών αλγορίθμων που μπορούν να διαχειριστούν τεράστιες ποσότητες δεδομένων και να επιλύσουν το ζήτημα της βελτιστοποίησης της υλοποίησης του μοντέλου. Σε ορισμένες περιπτώσεις, η αποτελεσματικότητα της μεθόδου μάθησης ή εξαγωγής συμπερασμάτων, συμπεριλαμβανομένης της χωρικής και χρονικής πολυπλοκότητάς της, μπορεί να είναι εξίσου κρίσιμη με την ακρίβειά της [6].

Η δημοτικότητα του τομέα μεταξύ των επιστημόνων δεδομένων και των ακαδημαϊκών έχει αυξηθεί ως αποτέλεσμα της ικανότητάς του να εντοπίζει μοτίβα, να συλλέγει πληροφορίες και να αυτοματοποιεί τη λήψη αποφάσεων.

## 2.2 Είδη Μηχανικής Μάθησης

Βάσει των δεδομένων που διαθέτουμε και του τρόπου εκπαίδευσης, οι αλγόριθμοι μηχανικής μάθησης μπορούν να ταξινομηθούν σε τρεις βασικές κατηγορίες.

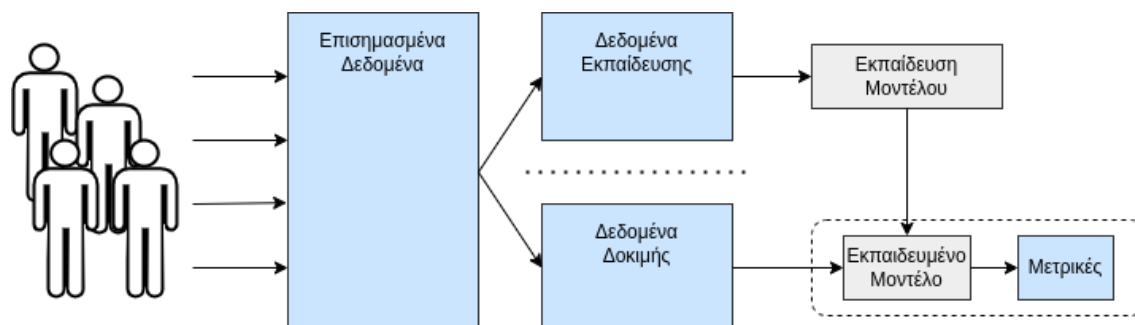
### 2.2.1 Επιβλεπόμενη Μάθηση

Η μορφή μηχανικής μάθησης που ονομάζεται επιβλεπόμενη μάθηση (supervised learning) στοχεύει στην ανάπτυξη μιας συνάρτησης που μετατρέπει τις εισόδους σε εξόδους με βάση επισημασμένα δεδομένα εκπαίδευσης. Σε αυτό το παράδειγμα, στον αλγόριθμο παρέχεται μια συλλογή επισημασμένων παραδειγμάτων (ζεύγη εισόδου/εξόδου), με στόχο την εκμάθηση της συσχέτισης μεταξύ εισόδων και εξόδων προκειμένου να προβλεφθούν τιμές για νέες, άγνωστες εισόδους [5]. Τα ακόλουθα βήματα περιλαμβάνονται συνήθως στη διαδικασία επιβλεπόμενης μάθησης:

1. Συγκέντρωση δεδομένων εκπαίδευσης με επισημάνσεις: Ο αλγόριθμος θα λάβει μια συλλογή ζευγών εισόδου/εξόδου για να κατανοήσει από αυτά την υποκείμενη σχέση μεταξύ εισόδων και εξόδων.
2. Επιλογή μοντέλου: Ένα μοντέλο είναι μια μαθηματική περιγραφή της σχέσης μεταξύ εισόδων και εξόδων, επομένως η επιλογή ενός κατάλληλου μοντέλου είναι σημαντική. Μπορεί να είναι ένα απλό γραμμικό μοντέλο ή ένα πιο περίπλοκο μη γραμμικό μοντέλο.
3. Εκπαίδευση του μοντέλου: Για να μειωθεί το σφάλμα πρόβλεψης, ο αλγόριθμος τροποποιεί τις παραμέτρους του μοντέλου με βάση τα δεδομένα εκπαίδευσης. Συνήθως, για το σκοπό αυτό χρησιμοποιούνται μέθοδοι βελτιστοποίησης όπως το gradient descent.



4. Αξιολόγηση του εκπαιδευμένου μοντέλου: Η ικανότητα γενίκευσης του εκπαιδευμένου μοντέλου αξιολογείται χρησιμοποιώντας ένα διαφορετικό σύνολο δεδομένων, γνωστό ως δεδομένα δοκιμής (test set).



Σχήμα 2.1: Σχηματική αναπαράσταση της επιβλεπόμενης μηχανικής μάθησης.

Η αναγνώριση ομιλίας (speech recognition), η επεξεργασία φυσικής γλώσσας (natural language processing - NLP), η ταξινόμηση εικόνων (image classification) και άλλοι τομείς μπορούν να επωφεληθούν από την επιβλεπόμενη μάθηση. Σε αυτές τις εφαρμογές, το μοντέλο εκπαιδεύεται σε ένα μεγάλο σύνολο επισημασμένων δεδομένων για να κατανοηθεί η θεμελιώδης σύνδεση μεταξύ εισόδων και εξόδων και στη συνέχεια εφαρμόζεται σε νέα δεδομένα για τη δημιουργία προβλέψεων.

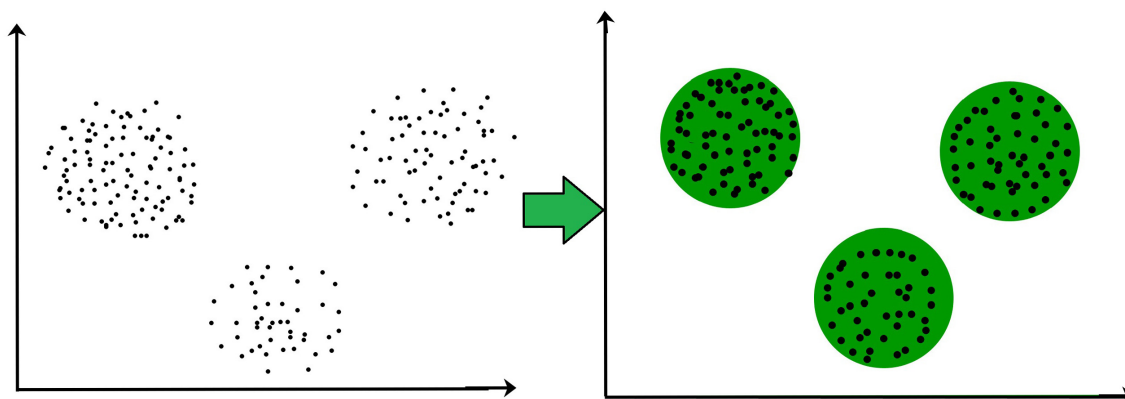
### 2.2.2 Μή-Επιβλεπόμενη Μάθηση

Στη μή-επιβλεπόμενη μάθηση (unsupervised learning), χρησιμοποιούνται αλγόριθμοι για την εύρεση μοτίβων σε σύνολα δεδομένων που περιέχουν σημεία δεδομένων που δεν έχουν κατηγοριοποιηθεί ούτε τους έχουν αποδοθεί ετικέτες. Έτσι, χωρίς καμία εξωτερική βοήθεια, οι αλγόριθμοι είναι ελεύθεροι να ταξινομήσουν, να επισημάνουν και να ομαδοποιήσουν τα σημεία δεδομένων μέσα στα σύνολα δεδομένων [7]

Το μοντέλο δεν απαιτεί επίβλεψη από τους χρήστες, καθώς στον αλγόριθμο μάθησης δεν δίνονται ετικέτες, οπότε αφήνεται στη δική του ευχέρεια να εντοπίσει τη δομή στα δεδομένα εισόδου. Η μή-επιβλεπόμενη μάθηση είναι όταν ένα σύστημα τεχνητής νοημοσύνης ομαδοποιεί αταξινομήτα δεδομένα με βάση τις ομοιότητες και τις διαφορές, παρόλο που δεν δίνονται κατηγορίες [8]. Η εύρεση κρυμμένων και ενδιαφερουσών μοτίβων σε μή επισημειωμένα δεδομένα είναι ο πρωταρχικός στόχος της μή-επιβλεπόμενης μάθησης.

Η μή-επιβλεπόμενη μάθηση, ανάλογα με τον τρόπο διαχείρισης των δεδομένων, χωρίζεται σε 4 βασικές κατηγορίες.

- **Συσταδοποίηση:** Στο πεδίο της μάθησης χωρίς επίβλεψη, η συσταδοποίηση είναι μια θεμελιώδης έννοια που διαδραματίζει κρίσιμο ρόλο στην ανακάλυψη μοτίβων ή δομών μέσα σε ένα σύνολο μη ταξινομημένων δεδομένων. Με τη χρήση αλγορίθμων συσταδοποίησης μή επιβλεπόμενης μάθησης, καθίσταται δυνατή η επεξεργασία δεδομένων και ο εντοπισμός φυσικών συστάδων ή ομάδων μέσα στα δεδομένα. Επιπλέον, ο αριθμός των συστάδων που θα πρέπει να εντοπίσουν οι αλγόριθμοι μπορεί να ρυθμιστεί ώστε να βελτιωθεί η λεπτομέρεια αυτών των ομάδων [9].



Σχήμα 2.2: Συσταδοποίηση δεδομένων 2 διαστάσεων σε 3 συστάδες.

Ουσιαστικά, η συσταδοποίηση μπορεί να θεωρηθεί ως το σημαντικότερο πρόβλημα μάθησης χωρίς επίβλεψη, καθώς πρόκειται για τη διαδικασία οργάνωσης αντικειμένων σε ομάδες με βάση τις ομοιότητές τους. Με άλλα λόγια, μια συστάδα είναι ένα σύνολο αντικειμένων που μοιράζονται ορισμένα κοινά στοιχεία και διαφέρουν από τα αντικείμενα που ανήκουν σε άλλες συστάδες. Συνολικά, η συσταδοποίηση προσφέρει μια ισχυρή προσέγγιση για την κατανόηση πολύπλοκων συνόλων δεδομένων και τη λήψη αποφάσεων βάσει δεδομένων.

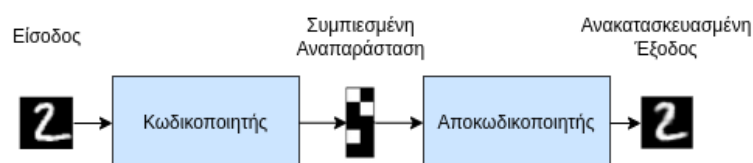
- **Ανίχνευση Ανωμαλιών:** Η ανίχνευση ανωμαλιών είναι μια κρίσιμη πτυχή της ανάλυσης δεδομένων που περιλαμβάνει τον εντοπισμό μοτίβων σε ένα σύνολο δεδομένων που βρίσκονται εκτός του αναμενόμενου εύρους. Με απλά λόγια, η ανίχνευση ανωμαλιών αναφέρεται στον εντοπισμό σημείων δεδομένων που αποκλίνουν σημαντικά από τα κανονικά πρότυπα που παρατηρούνται σε ένα σύνολο δεδομένων. Με τον εντοπισμό αυτών των ανωμαλιών, οι αναλυτές μπορούν να αποκτήσουν πολύτιμες πληροφορίες σχετικά με πιθανά ζητήματα που ενδέχεται να απαιτούν περαιτέρω διερεύνηση και επίλυση.
- **Κανόνες Συσχέτισης:** Η εξόρυξη κανόνων συσχέτισης είναι ένας τύπος μάθησης χωρίς επίβλεψη που περιλαμβάνει την ανακάλυψη ενδιαφερουσών σχέσεων μεταξύ μεταβλητών σε μεγάλα σύνολα δεδομένων χωρίς τη χρήση επισημασμένων δεδομένων. Σε αντίθεση με την επιβλεπόμενη μάθηση όπου παρέχονται επισημασμένα δεδομένα στον αλγόριθμο, η εξόρυξη κανόνων συσχέτισης περιλαμβάνει τον αλγόριθμο που κάνει παρατηρήσεις και εντοπίζει μοτίβα που θα μπορούσαν να είναι χρήσιμα για τη λήψη αποφάσεων ή προβλέψεων.

Αυτή η μέθοδος είναι συνήθως περιγραφική παρά προγνωστική και χρησιμοποιείται συχνά για την αποκάλυψη κρυφών σχέσεων μεταξύ μεταβλητών σε ένα σύνολο δεδομένων. Οι σχέσεις που ανακαλύπτονται συνήθως αναπαρίστανται με τη μορφή κανόνων ή συνόλων συχνών στοιχείων, τα οποία μπορούν να χρησιμοποιηθούν για την απόκτηση πληροφοριών σχετικά με τα δεδομένα [10].

Η εξόρυξη κανόνων συσχέτισης μπορεί να θεωρηθεί μια προηγμένη μορφή σεναρίου “what-if”, όπου ο αλγόριθμος προσπαθεί να εντοπίσει ενδιαφέρουσες σχέσεις μεταξύ

μεταβλητών στα δεδομένα. Εάν συμβεί ένα συγκεκριμένο γεγονός, τότε ο αλγόριθμος μπορεί να προβλέψει την πιθανότητα να συμβεί ένα άλλο γεγονός με βάση τις ανακαλυφθείσες συσχετίσεις.

- Αυτόκωδικοποιητής:** Οι αυτοκωδικοποιητές (Autoencoders) είναι μια ισχυρή και δημοφιλής τεχνική μή-επιβλεπόμενης μάθησης. Βασίζονται σε νευρωνικά δίκτυα για την εκτέλεση του έργου της μάθησης αναπαράστασης, το οποίο συνεπάγεται την εξεύρεση τρόπου μετατροπής αδόμητων δεδομένων σε μια πιο ωφέλιμη μορφή. Οι αυτοκωδικοποιητές είναι μια συγκεκριμένη κατηγορία νευρωνικών δικτύων, τα οποία είναι φτιαγμένα για να δέχονται μια είσοδο και να εξάγουν το ίδιο πράγμα με την είσοδο. Ο αυτοκωδικοποιητής συμπιέζει την είσοδο σε μία αναπαράσταση χαμηλότερης διάστασης πριν από την ανακατασκευή της εξόδου, η οποία είναι μια συμπυκνωμένη "περίληψη" ή συμπίεση της εισόδου [11]. Αποτελείται από τον κωδικοποιητή και τον αποκωδικοποιητή. Η είσοδος συμπιέζεται από τον κωδικοποιητή σε μία αναπαράσταση, και ο αποκωδικοποιητής πρέπει να λάβει αυτήν την αναπαράσταση και να την χρησιμοποιήσει για να ανακατασκευάσει την αρχική είσοδο. Μια μέθοδος κωδικοποίησης, μια μέθοδος αποκωδικοποίησης και μια συνάρτηση απώλειας για τη σύγκριση της εξόδου με τον στόχο είναι απαραίτητα για την κατασκευή ενός αυτοκωδικοποιητή. Ανάλογα με τον τύπο των δεδομένων εισόδου, για τη σχεδίαση του κωδικοποιητή και του αποκωδικοποιητή μπορούν να χρησιμοποιηθούν διαφορετικές αρχιτεκτονικές νευρωνικών δικτύων.



Σχήμα 2.3: Σχηματική αναπαράσταση ενός αυτοκωδικοποιητή.

Η αποθρομβοποίηση δεδομένων, η εξαγωγή χαρακτηριστικών και η ανίχνευση ανωμαλιών είναι μερικές μόνο από τις λειτουργίες για τις οποίες μπορούν να χρησιμοποιηθούν οι αυτοκωδικοποιητές εκτός από τη μείωση της διαστατικότητας ή τη συμπίεση. Οι αυτοκωδικοποιητές μπορούν να μάθουν να αναπαριστούν πολύπλοκα, υψηλών διαστάσεων δεδομένα με πιο συνοπτικό τρόπο, γεγονός που αποτελεί ένα από τα κύρια πλεονεκτήματά τους. Αυτό μπορεί να είναι χρήσιμο για μια ποικιλία εφαρμογών, όπως η μείωση των αναγκών αποθήκευσης για τεράστια σύνολα δεδομένων, η επιτάχυνση της επεξεργασίας ή η απλούστευση της ανάλυσης και ερμηνείας δύσκολων δεδομένων.

### 2.2.3 Ενισχυτική Μάθηση

Το πεδίο της ενισχυτικής μάθησης (Reinforcement Learning) εστιάζει στην ανατροφοδότηση και όχι στις σαφείς ετικέτες. Για να βελτιστοποιήσουν τις ανταμοιβές, τα μοντέλα αλληλεπιδρούν με το περιβάλλον τους και λαμβάνουν αποφάσεις με βάση τις δικές τους εκτιμήσεις των συνθηκών. Σε αντίθεση με την επιβλεπόμενη μάθηση, τα μοντέλα ενισχυτικής μάθησης βασίζονται αποκλειστικά στην εμπειρία μέσω της διαδικασίας δοκιμής και σφάλμα-

τος για να μάθουν νέες λειτουργίες. Αλληλεπιδρώντας με το περιβάλλον τους, η ενισχυτική μάθηση δίνει τη δυνατότητα στα μοντέλα να κάνουν αποτελεσματικά διαδοχικές κρίσεις. Το μοντέλο χρησιμοποιεί την ανατροφοδοτούμενη αξιολόγηση και επιβράβευση για να καθοδηγεί τις επιλογές συμπεριφοράς του και βελτιώνεται σταδιακά ανάλογα με τις εμπειρίες του [12]. Η ενισχυτική μάθηση διαφέρει από την επιβλεπόμενη μάθηση επειδή δεν εξαρτάται από ετικέτες, καθώς δίνεται έμφαση στην λήψη διαδοχικών βέλτιστων αποφάσεων [13]. Οι δύο υποκατηγορίες της ενισχυτικής μάθησης είναι η θετική και η αρνητική ενισχυτική μάθηση. Η θετική ενισχυτική μάθηση επιβραβεύει τις επιθυμητές αποφάσεις και συμπεριφορές, ενώ η αρνητική “τιμωρεί” τις δυσμενείς.

## Κεφάλαιο **3**

# Μοντέλα Μηχανικής Μάθησης - Μετρικές

---

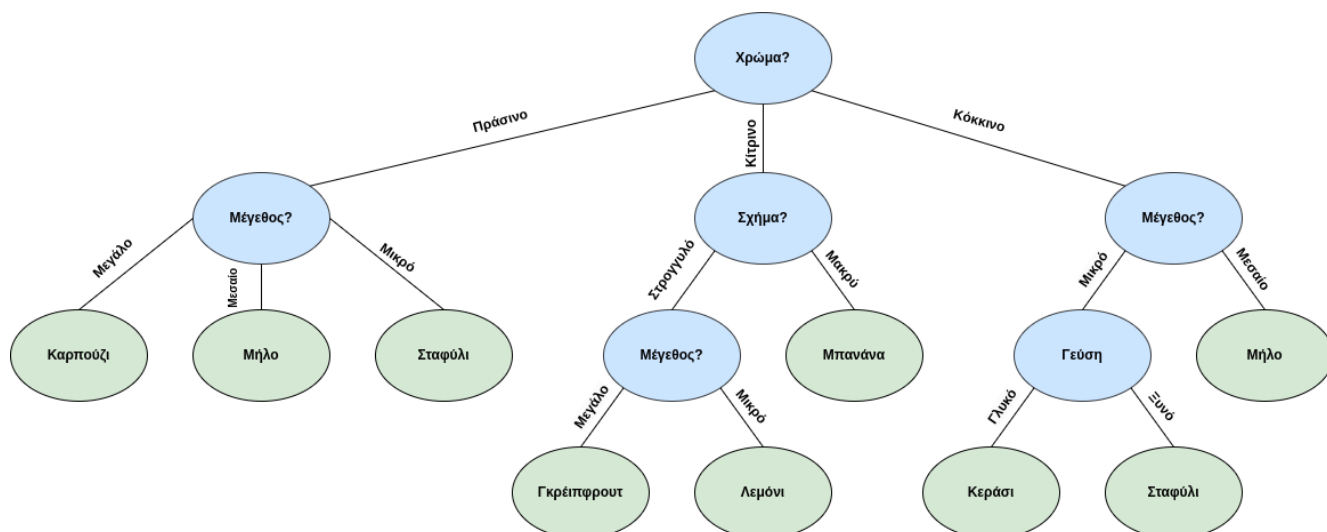
Στο κεφάλαιο αυτό γίνεται μια εκτενής περιγραφή των μοντέλων μηχανικής μάθησης που χρησιμοποιήθηκαν και μεταξύ των οποίων έγινε η σύγκριση για να βρεθεί το καταλληλότερο για τον σκοπό της παρούσας εργασίας. Επιπλέον, αναλύονται οι μετρικές συναρτήσεις που χρησιμοποιήθηκαν για την αξιολόγηση και τη σύγκριση των υποψήφιων μοντέλων.

## 3.1 Μοντέλα

### 3.1.1 Δέντρα Απόφασης

Τα Δέντρα Απόφασης (Decision Trees) αποτελούν γνωστό παράδειγμα μοντέλου επιβλεπόμενης μηχανικής μάθησης για ταξινόμηση δεδομένων (classification). Η έννοια αυτή προέρχεται από την τυπική δομή δέντρου, το οποίο αποτελείται από μια ρίζα, κόμβους, κλαδιά και φύλλα. Ομοίως, ένα Δέντρο Αποφάσεων αποτελείται από κόμβους που συμβολίζονται σχηματικά με κύκλους, με τις ακμές που συνδέουν τους κόμβους να αντιπροσωπεύουν τα κλαδιά. Ένα Δέντρο Αποφάσεων ξεκινά από τη ρίζα και γενικά εκτείνεται από αριστερά προς τα δεξιά με κατεύθυνση προς τα κάτω. Ο κόμβος εκκίνησης αναφέρεται ως κόμβος "ρίζας", ενώ ο κόμβος τερματισμού ονομάζεται κόμβος "φύλλο". Κάθε εσωτερικός κόμβος, ένας κόμβος που δεν είναι κόμβος-φύλλο, μπορεί να έχει δύο ή περισσότερα κλαδιά που εκτείνονται από αυτόν. Οι κόμβοι αντιστοιχούν σε συγκεκριμένα χαρακτηριστικά, ενώ τα κλαδιά υποδηλώνουν ένα εύρος τιμών. Αυτά τα εύρη τιμών χρησιμεύουν ως σημεία κατάτμησης για το σύνολο τιμών του συγκεκριμένου χαρακτηριστικού [14]. Ένα παράδειγμα δενδρικής δομής απεικονίζεται στο Σχήμα 3.1.

Σε ένα Δέντρο Απόφασης, η ταξινόμηση των δεδομένων καθορίζεται από τις τιμές των χαρακτηριστικών. Το δέντρο κατασκευάζεται χρησιμοποιώντας ταξινομημένα δεδομένα, με διαχωρισμούς κλάσεων με βάση χαρακτηριστικά που διαχωρίζουν βέλτιστα τα δεδομένα. Τα δεδομένα χωρίζονται σύμφωνα με αυτές τις τιμές των χαρακτηριστικών. Αυτή η διαδικασία εφαρμόζεται αναδρομικά σε κάθε υποσύνολο στοιχείων δεδομένων που υποδιαιρείται. Η διαδικασία ολοκληρώνεται όταν όλα τα στοιχεία δεδομένων εντός του τρέχοντος υποσυνόλου ανήκουν σε μία μόνο κλάση.



Σχήμα 3.1: Σχηματική αναπαράσταση Δέντρου Απόφασης για ταξινόμηση φρούτων.

### 3.1.2 Τυχαίο Δάσος

Το Τυχαίο Δάσος (Random Forest) είναι μια συλλογή μη κλαδωμένων δέντρων ταξινόμησης ή παλινδρόμησης (αναλόγως της φύσης του προβλήματος) που προέρχονται από τυχαία υποσύνολα δεδομένων εκπαίδευσης. Οι προβλέψεις γίνονται με τη συγκέντρωση των προβλέψεων του συνόλου (ψήφος πλειοψηφίας για ταξινόμηση ή μέσος όρος για παλινδρόμηση). Η κατασκευή κάθε δέντρου γίνεται ως εξής [15]:

- Επιλογή  $N$  τυχαίων δειγμάτων, όπου το  $N$  αντιπροσωπεύει τον αριθμό των δειγμάτων στα δεδομένα εκπαίδευσης, με αντικατάσταση στα αρχικά δεδομένα. Τα τυχαία αυτά δείγματα χρησιμεύουν ως δεδομένα εκπαίδευσης για την ανάπτυξη του δέντρου. Ένα μέρος των δεδομένων αυτών δεν χρησιμοποιούνται για εκπαίδευση αλλά χρησιμοποιούνται για την αμερόληπτη εκτίμηση του σφάλματος ταξινόμησης καθώς προστίθενται νέα δέντρα στο δάσος.
- Για  $M$  μεταβλητές εισόδου, επιλέξτε τη μεταβλητή  $m$  έτσι ώστε  $m \ll M$ . Στη συνέχεια, σε κάθε κόμβο επιλέγονται τυχαία  $m$  μεταβλητές από τις  $M$  διαθέσιμες και υπολογίζεται η βέλτιστη διαίρεση σε αυτές τις  $m$  μεταβλητές για να διαιρεθεί ο κόμβος. Κατά τη διάρκεια της ανάπτυξης του δάσους, η τιμή του  $m$  παραμένει σταθερή.
- Κάθε δέντρο αναπτύσσεται στο μέγιστο δυνατό μέγεθος του χωρίς κανένα κλάδεμα.

Σε σύγκριση με έναν ταξινομητή ενός δέντρου, το Τυχαίο Δάσος παρουσιάζει γενικά σημαντική βελτίωση των επιδόσεων. Το ποσοστό σφάλματος γενίκευσής του είναι συγκρίσιμο με εκείνο του Adaboost, αλλά παρουσιάζει και μεγαλύτερη ανθεκτικότητα στο θόρυβο των δεδομένων.

Στην ουσία, το Τυχαίο Δάσος είναι μια μέθοδος μηχανικής μάθησης που χρησιμοποιεί πολλαπλά Δέντρα Απόφασης για να ενισχύσει την ακρίβεια πρόβλεψης και να αποφύγει την υπερπροσαρμογή στα δεδομένα εκπαίδευσης. Με τον μέσο όρο των εξόδων από τα μεμονωμένα δέντρα, το μοντέλο μειώνει αποτελεσματικά τη διακύμανση και μειoriάζει την

επίδραση του θορύβου στα δεδομένα. Επιπλέον, η τυχαία επιλογή χαρακτηριστικών κατά τη διαδικασία επαγωγής βοηθά στη διαφοροποίηση των δέντρων, ενισχύοντας περαιτέρω την απόδοση του συνόλου.

### 3.1.3 Adaptive Boosting

Ο Yoan Freund και ο Robert Schapire εισήγαγαν τη μέθοδο μάθησης γνωστή ως AdaBoost, συντομογραφία των λέξεων Adaptive Boosting [16]. Προσαρμόζοντας επαναληπτικά τα βάρη των αδύναμων ταξινομητών με βάση την απόδοσή τους, συνδυάζει αρκετούς αδύναμους ταξινομητές για να δημιουργήσει έναν ισχυρό ταξινομητή, αυξάνοντας την ακρίβεια των προβλέψεων.

Ο τρόπος λειτουργίας του αλγορίθμου AdaBoost έχει ως εξής:

ΑΛΓΟΡΙΘΜΟΣ 3.1: Αλγόριθμος AdaBoost

---

```

1: procedure ADABOOST( $X, y, M$ )
2:   Αρχικοποίησε τα βάρη με την ίδια τιμή για κάθε σημείο στα δεδομένα  $w_i = \frac{1}{N}$  for  $i = 1, 2, \dots, N$ 
3:   for  $m = 1, 2, \dots, M$  do
4:     Προσάρμοσε έναν αδύναμο ταξινομητή  $h_m(x)$  στα δεδομένα  $X$  με βάρη  $w_i$ 
5:     Υπολόγισε το σταθμισμένο σφάλμα ταξινόμησης:  $\epsilon_m = \frac{\sum_{i=1}^N w_i (y_i \neq h_m(x_i))}{\sum_{i=1}^N w_i}$ 
6:     Υπολόγισε το βάρος του ταξινομητή:  $a_m = \frac{1}{2} \ln \left( \frac{1-\epsilon_m}{\epsilon_m} \right)$ 
7:     Προσάρμοσε τα βάρη των δεδομένων:  $w_i = w_i \cdot e^{-a_m y_i h_m(x_i)}$  for  $i = 1, 2, \dots, N$ 
8:     Κανονικοποίησε τα βάρη:  $w_i = \frac{w_i}{\sum_{i=1}^N w_i}$  for  $i = 1, 2, \dots, N$ 
9:   end for
10:  return  $F(x) = \sum_{m=1}^M a_m h_m(x)$ 
11: end procedure

```

---

Η επιτυχία του AdaBoost οφείλεται σε μεγάλο βαθμό στην ικανότητά του να αυξάνει προσαρμοστικά τα βάρη των δύσκολων περιπτώσεων, γεγονός που παρακινεί τους αδύναμους ταξινομητές να δίνουν μεγαλύτερη προσοχή σε αυτές τις περιπτώσεις στις επόμενες επαναλήψεις [17]. Το AdaBoost έχει χρησιμοποιηθεί με επιτυχία σε διάφορους τομείς, όπως η αναγνώριση ομιλίας, η όραση υπολογιστών και η επεξεργασία φυσικής γλώσσας [18].

Έχει όμως και ορισμένα μειονεκτήματα:

- Το AdaBoost μπορεί να είναι ευαίσθητο σε θορυβώδη δεδομένα και ακραίες τιμές, επειδή μπορεί να υπερτονίσει αυτές τις εμφανίσεις κατά τη διάρκεια της διαδικασίας μάθησης [19].
- Προβλήματα με την επεκτασιμότητα: Για μεγάλα σύνολα δεδομένων, ο αλγόριθμος μπορεί να γίνει υπολογιστικά ακριβός λόγω της ανάγκης για πολλαπλές επαναλήψεις [20].

### 3.1.4 Gradient Tree Boosting

Το Gradient Tree Boosting, επίσης γνωστό ως Gradient Boosted Decision Trees (GBDT) ή Gradient Boosting Machines (GBM), είναι μια ισχυρή μέθοδος μάθησης που έχει επιδειξει

επιτυχία σε διάφορες εργασίες μηχανικής μάθησης, συμπεριλαμβανομένης της παλινδρόμησης και της ταξινόμησης [21]. Η τεχνική συνδυάζει τα πλεονεκτήματα των δέντρων απόφασης και της ενίσχυσης, με αποτέλεσμα ένα ισχυρό και ερμηνεύσιμο μοντέλο ικανό να χειρίζεται δεδομένα μεγάλης κλίμακας και υψηλών διαστάσεων.

Το Gradient Tree Boosting είναι μια επαναληπτική μέθοδος μάθησης που συνδυάζει τις προβλέψεις πολλαπλών αδύναμων μαθητών, δέντρων απόφασης, για να σχηματίσει έναν ισχυρό προβλεπτή [21]. Η βασική ιδέα πίσω από την ενίσχυση είναι η βελτίωση της απόδοσης του συνόλου με την προσθήκη νέων δέντρων που αντισταθμίζουν τα σφάλματα που κάνουν τα υπάρχοντα δέντρα.

Ο αλγόριθμος Gradient Boosting μπορεί να συνοψιστεί στα ακόλουθα βήματα [21]:

### ΑΛΓΟΡΙΘΜΟΣ 3.2: Αλγόριθμος Gradient Boosting

---

```

1: procedure GRADIENTBOOSTING( $X, y, M, L$ )
2:   Αρχικοποίησε το μοντέλο με μια σταθερή πρόβλεψη  $F_0(x) = \arg \min_{\gamma} \sum_{i=1}^N L(y_i, \gamma)$ 
3:   for  $m = 1, 2, \dots, M$  do
4:     Υπολογίστε τις αρνητικές κλίσεις (ψευδο-υπολειμματικές) της συνάρτησης
     απώλειας:  $r_{im} = - \left[ \frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} \right]_{F=F_{m-1}}$  for  $i = 1, 2, \dots, N$ 
5:     Προσάρμοσε έναν αδύναμο μαθητή  $h_m(x)$  στις ψευδο-υπολειμματικές:  $h_m(x) \approx$ 
      $\arg \min_h \sum_{i=1}^N (r_{im} - h(x_i))^2$ 
6:     Υπολόγισε το βήμα  $a_m$ :  $a_m = \arg \min_a \sum_{i=1}^N L(y_i, F_{m-1}(x_i) + ah_m(x_i))$ 
7:     Ανανέωσε το μοντέλο:  $F_m(x) = F_{m-1}(x) + a_m h_m(x)$ 
8:   end for
9:   return  $F_M(x)$ 
10: end procedure

```

---

Το Gradient Tree Boosting διαθέτει αρκετές υπερπαραμέτρους που μπορούν να ρυθμιστούν για τη βελτίωση της απόδοσης του μοντέλου:

- Αριθμός δέντρων: Η αύξηση του αριθμού των δέντρων βελτιώνει γενικά την απόδοση του μοντέλου, αλλά μπορεί να οδηγήσει σε υπερπροσαρμογή αν οριστεί πολύ υψηλός [8].
- Βάθος δέντρων: Τα βαθύτερα δέντρα μπορούν να συλλάβουν πιο σύνθετες αλληλεπιδράσεις, αλλά είναι επιρρεπή σε υπερπροσαρμογή. Τα μικρότερα δέντρα είναι πιο ερμηνεύσιμα και λιγότερο επιρρεπή σε υπερπροσαρμογή [22].
- Ρυθμός μάθησης: Ένας μικρότερος ρυθμός μάθησης συνήθως βελτιώνει την απόδοση του μοντέλου μειώνοντας την υπερπροσαρμογή, αλλά απαιτεί περισσότερα δέντρα και αυξάνει τον χρόνο εκπαίδευσης [21].

### 3.1.5 Αλγόριθμος K-Πλησιέστερων Γειτόνων

Ο αλγόριθμος K-πλησιέστερων γειτόνων (K-Nearest Neighbors - K-NN) είναι ένας από τους απλούστερους αλλά αποτελεσματικούς αλγορίθμους μηχανικής μάθησης που χρησιμοποιούνται για εργασίες ταξινόμησης και παλινδρόμησης [23]. Σε αντίθεση με τους περισσότερους αλγορίθμους, ο K-NN δεν μαθαίνει ρητά ένα μοντέλο. Αντ' αυτού, χρησιμοποιεί μια



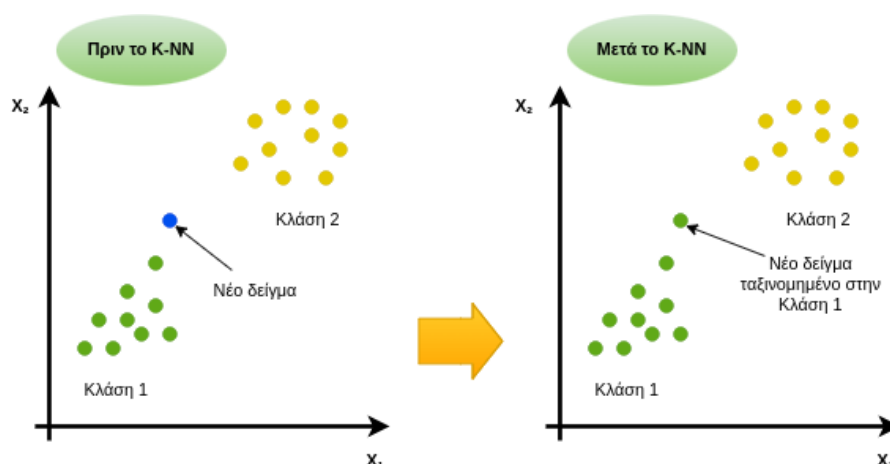
αποθηκευμένη βάση δεδομένων για την πραγματοποίηση προβλέψεων, ως εκ τούτου θεωρείται αλγόριθμος μάθησης βασισμένος σε περιπτώσεις (instance-based learning algorithm) [24].

Η αρχή πίσω από τον K-NN είναι διαισθητική: παρόμοια πράγματα βρίσκονται κοντά το ένα στο άλλο. Ο K-NN λειτουργεί αποθηκεύοντας όλα τα δείγματα εισόδου και ταξινομώντας τα νέα δείγματα με βάση ένα μέτρο ομοιότητας, όπως η απόσταση [25]. Για ένα νέο δείγμα, υπολογίζει την απόσταση από όλα τα άλλα δείγματα εκπαίδευσης. Στη συνέχεια, ο αλγόριθμος επιλέγει τους "K" πλησιέστερους γείτονες, όπου "K" είναι ένας προκαθορισμένος αριθμός. Η πρόβλεψη γίνεται με βάση την πλειοψηφική κλάση μεταξύ αυτών των γειτόνων για την ταξινόμηση ή τον μέσο όρο για εργασίες παλινδρόμησης.

Μαθηματικά, αν συμβολίσουμε ένα δείγμα με ένα διάνυσμα χαρακτηριστικών  $x \in R^n$ , η μετρική απόσταση που χρησιμοποιείται συχνά είναι η ευκλείδεια απόσταση [26]:

$$d(x_i, x_j) = \sqrt{\sum_{l=1}^n (x_{il} - x_{jl})^2} \quad (3.1)$$

όπου  $x_i, x_j$  είναι το  $i^{\text{οστο}}$  και  $j^{\text{οστο}}$  δείγμα στο σύνολο δεδομένων.



Σχήμα 3.2: Σχηματική αναπαράσταση του αλγόριθμου K-NN για δεδομένα με 2 χαρακτηριστικά.

Η επιλογή του σωστού αριθμού γειτόνων, "K", είναι ζωτικής σημασίας για την αποδοτικότητα του αλγορίθμου K-NN. Μια μικρή τιμή του "K" μπορεί να οδηγήσει σε προβλέψεις που επηρεάζονται σε μεγάλο βαθμό από το θόρυβο, ενώ ένα μεγάλο "K" μπορεί να προκαλέσει την ένταξη δειγμάτων από άλλες κλάσεις, οδηγώντας σε λανθασμένη ταξινόμηση ή ανακριβείς προβλέψεις [27]. Δεν υπάρχει καθολικά βέλτιστος αριθμός για το "K" και η επιλογή εξαρτάται συχνά από τα δεδομένα. Για την επιλογή του "K" συνήθως γίνεται έλεγχος με ένα σύνολο δεδομένων επικυρωποίησης [28].

Ενώ ο βασικός αλγόριθμος K-NN είναι απλός, διάφορες επεκτάσεις μπορούν να βελτιώσουν την αποδοτικότητά του. Μια τέτοια επέκταση είναι ο σταθμισμένος K-NN, όπου αντί κάθε ένας από τους "K" γείτονες να συνεισφέρει εξίσου στην τελική πρόβλεψη, οι συνεισφορές τους σταθμίζονται ανάλογα με την απόστασή τους από το δείγμα δοκιμής. Με αυτόν τον τρόπο, οι πιο κοντινοί γείτονες συμβάλλουν περισσότερο στην τελική πρόβλεψη από τους πιο

απομακρυσμένους.

Το K-NN έχει εφαρμοστεί με επιτυχία σε τομείς όπως η αναγνώριση εικόνων, η αναγνώριση βίντεο και τα συστήματα συστάσεων. Ωστόσο, πάσχει από περιορισμούς όπως η υπολογιστική αναποτελεσματικότητα για μεγάλα σύνολα δεδομένων και η κακή απόδοση σε χώρους υψηλών διαστάσεων - ένα φαινόμενο γνωστό ως "κατάρα της διαστατικότητας" [29]. Επίσης, η αποδοτικότητα του K-NN μπορεί να υποβαθμιστεί σημαντικά με άσχετα χαρακτηριστικά, δεδομένου ότι όλα τα χαρακτηριστικά συμβάλλουν στον υπολογισμό της απόστασης.

## 3.2 Μετρικές

Στην περιοχή της τεχνητής νοημοσύνης και της μηχανικής μάθησης, οι μετρικές παίζουν καθοριστικό ρόλο στην αξιολόγηση και εκτίμηση της απόδοσης των μοντέλων. Αυτές οι μετρικές παρέχουν πολύτιμες πληροφορίες σχετικά με την απόδοση και την αποτελεσματικότητα των μοντέλων μηχανικής μάθησης, επιτρέποντας στους ερευνητές και τους μηχανικούς να λαμβάνουν τεκμηριωμένες αποφάσεις. Η κατανόηση αυτών των μετρικών και των εφαρμογών τους είναι ζωτικής σημασίας για την αξιολόγηση και τη σύγκριση διαφορετικών αλγορίθμων σε διάφορους τομείς. Παρακάτω αναφέρονται οι πιο γνωστές από αυτές.

### 3.2.1 Ακρίβεια

Η ακρίβεια (accuracy) είναι ίσως η πιο βασική και ευρέως χρησιμοποιούμενη μετρική στη μηχανική μάθηση. Μετρά το ποσοστό των σωστά ταξινομημένων δειγμάτων επί του συνολικού αριθμού των δειγμάτων. Η ακρίβεια μπορεί να αναπαρασταθεί μαθηματικά ως εξής:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3.2)$$

όπου TP αντιπροσωπεύει τα αληθώς θετικά (True Positives), TN αντιπροσωπεύει τα αληθώς αρνητικά (True Negatives), FP αντιπροσωπεύει τα ψευδώς θετικά (False Positives) και FN αντιπροσωπεύει τα ψευδώς αρνητικά (False Negatives). Η ακρίβεια παρέχει μια συνολική εικόνα της απόδοσης ενός μοντέλου, αλλά μπορεί να μην είναι κατάλληλη για μη ισορροπημένα σύνολα δεδομένων [8].

### 3.2.2 Ακρίβεια Προσέγγισης και Ανάκληση

Η ακρίβεια προσέγγισης (precision) και η ανάκληση (recall) είναι μετρικές που χρησιμοποιούνται συνήθως, ιδίως σε εργασίες ταξινόμησης, όπου υπάρχει ανισορροπία μεταξύ των κλάσεων. Η ακρίβεια προσέγγισης μετρά το ποσοστό των σωστά προβλεπόμενων θετικών δειγμάτων επί του συνόλου των δειγμάτων που προβλέπονται ως θετικά. Η ανάκληση, από την άλλη πλευρά, υπολογίζει το ποσοστό των σωστά προβλεπόμενων θετικών δειγμάτων επί του συνόλου των πραγματικών θετικών δειγμάτων. Μαθηματικά, η ακρίβεια προσέγγισης και η ανάκληση μπορούν να οριστούν ως εξής:

$$precision = \frac{TP}{TP + FP} \quad (3.3)$$

$$recall = \frac{TP}{TP + FN} \quad (3.4)$$

Αυτές οι μετρικές χρησιμοποιούνται συχνά μαζί και μπορούν να συνδυαστούν σε μια ενιαία μετρική που ονομάζεται βαθμολογία F1, η οποία είναι ο αρμονικός μέσος όρος της ακρίβειας προσέγγισης και της ανάκλησης. Η βαθμολογία F1 παρέχει μια ισορροπημένη αξιολόγηση, ειδικά όταν το σύνολο δεδομένων έχει άνιση κατανομή κλάσεων [30].

### 3.2.3 Περιοχή Κάτω από την Καμπύλη ROC

Η καμπύλη ROC (Receiver Operating Characteristic) είναι μια γραφική αναπαράσταση της απόδοσης ενός ταξινομητή με την απεικόνιση του αληθούς θετικού ποσοστού (ευαισθησία) έναντι του ψευδώς θετικού ποσοστού (ειδικότητα) σε διάφορα κατώτατα όρια ταξινόμησης. Η περιοχή κάτω από αυτή την καμπύλη, γνωστή ως Area Under Curve - Receiver Operating Characteristic (AUC-ROC), είναι μια ευρέως χρησιμοποιούμενη μετρική για την αξιολόγηση της απόδοσης δυαδικών ταξινομητών. Η AUC-ROC κυμαίνεται από 0 έως 1, όπου οι υψηλότερες τιμές υποδηλώνουν καλύτερη απόδοση ταξινόμησης [31].

### 3.2.4 Μέσο Απόλυτο Σφάλμα και Μέσο Τετραγωνικό Σφάλμα

Το Μέσο Απόλυτο Σφάλμα (Mean Absolute Error - MAE) και το Μέσο Τετραγωνικό Σφάλμα (Mean Square Error - MSE) είναι δημοφιλείς μετρικές για την αξιολόγηση μοντέλων παλινδρόμησης. Το MAE υπολογίζει τη μέση απόλυτη διαφορά μεταξύ των προβλεπόμενων και των πραγματικών τιμών, ενώ το MSE υπολογίζει τη μέση τετραγωνική διαφορά. Αυτές οι μετρικές παρέχουν ένα μέτρο της προβλεπτικής ακρίβειας του μοντέλου και του μεγέθους των σφαλμάτων. Ενώ το MAE είναι πιο ανθεκτικό στις ακραίες τιμές, το MSE τιμωρεί περισσότερο τα μεγάλα σφάλματα [3].

### 3.2.5 R-Τετράγωνο

Το  $R^2$  είναι μια μετρική που χρησιμοποιείται σε εργασίες παλινδρόμησης για να εκτιμηθεί πόσο καλά ένα μοντέλο ταιριάζει στα δεδομένα. Μετρά το ποσοστό της διακύμανσης της εξαρτημένης μεταβλητής που μπορεί να εξηγηθεί από τις ανεξάρτητες μεταβλητές. Το  $R^2$  κυμαίνεται από 0 έως 1, όπου το 1 υποδηλώνει τέλεια προσαρμογή, ενώ τιμές κοντά στο 0 υποδηλώνουν κακή προσαρμογή [32].



## Κεφάλαιο 4

# Βαθιά Μάθηση

---

Στο κεφάλαιο αυτό περιγράφεται η αρχή της Βαθιάς Μάθησης (Deep Learning). Η βαθιά μάθηση είναι ένα υποσύνολο της μηχανικής μάθησης, η οποία είναι ένας κλάδος της τεχνητής νοημοσύνης που στοχεύει στη μίμηση του προτύπου μάθησης του ανθρώπινου εγκεφάλου για την επίλυση σύνθετων προβλημάτων. Η βαθιά μάθηση χρησιμοποιεί ένα μοντέλο υπολογισμού εμπνευσμένο από τη δομή του εγκεφάλου, ένα δίκτυο τεχνητών νευρώνων ή μονάδων - εξ ου και η ονομασία "Τεχνητά Νευρωνικά Δίκτυα" (Artificial Neural Networks - ANNs). Παρακάτω θα αναλύσουμε τα επιμέρους τμήματα και τα είδη των στρωμάτων (layers) που χρησιμοποιούνται για τη δημιουργία μοντέλων καθώς και ο αλγόριθμος που χρησιμοποιείται για την εκπαίδευση και βελτιστοποίηση αυτών. Επιπλέον θα παρουσιαστούν τα μοντέλα βαθιάς μάθησης που έχουν χρησιμοποιηθεί στη παρούσα εργασία.

### 4.1 Τεχνητά Νευρωνικά Δίκτυα

Τα τεχνητά νευρωνικά δίκτυα (Artificial Neural Networks - ANNs) είναι εμπνευσμένα από τη δομή και τις λειτουργικές πτυχές των βιολογικών νευρωνικών δικτύων. Ένα απλό ANN αποτελείται από τρεις τύπους επιπέδων: ένα επίπεδο εισόδου, ένα ή περισσότερα κρυφά επίπεδα και ένα επίπεδο εξόδου. Κάθε επίπεδο αποτελείται από κόμβους ή τεχνητούς νευρώνες, οι οποίοι συνδέονται με σταθμισμένες ακμές που αντιπροσωπεύουν την ισχύ ή την κατεύθυνση του σήματος μεταξύ των κόμβων [33].

Η θεμελιώδης εξίσωση που περιγράφει τον υπολογισμό σε έναν μόνο νευρώνα είναι:

$$z = w_1 * x_1 + w_2 * x_2 + \dots + w_n * x_n + b \quad (4.1)$$

$$a = f(z) \quad (4.2)$$

όπου  $z$  είναι το σταθμισμένο άθροισμα των εισόδων  $x$  (συν έναν όρο προκατάληψης  $b$ ),  $a$  είναι η ενεργοποίηση του νευρώνα και  $f$  είναι η συνάρτηση ενεργοποίησης. Τα βάρη  $w$  είναι οι παράμετροι που μαθαίνει το δίκτυο κατά την εκπαίδευση [33].

#### 4.1.1 Είδη Επιπέδων στα Μοντέλα Βαθιάς Μάθησης

Τα πιο συνηθισμένα είδη επιπέδων που χρησιμοποιούνται στα μοντέλα βαθιάς μάθησης είναι τα παρακάτω:

1. Γραμμικά επίπεδα (Linear layers): Σε ένα γραμμικό (ή πλήρως συνδεδεμένο) επίπεδο, κάθε νευρώνας στο επίπεδο συνδέεται με κάθε νευρώνα στο προηγούμενο επίπεδο. Τα βάρη και οι προκαταλήψεις μαθαίνονται μέσω της εκπαίδευσης, επιτρέποντας στο επίπεδο να απεικονίζει τα δεδομένα εισόδου σε έναν χώρο που είναι ευκολότερο για το μοντέλο να κατανοήσει [34].
2. Συνελικτικά επίπεδα (Convolutional layers): Αυτά τα επίπεδα χρησιμοποιούν έναν πυρήνα ή φίλτρο που σαρώνει τα δεδομένα εισόδου, εκτελώντας πολλαπλασιασμό πινάκων ανά στοιχείο και δημιουργώντας έναν χάρτη χαρακτηριστικών [35]. Χρησιμοποιούνται συχνά για εργασίες επεξεργασίας εικόνας.
3. Επίπεδα συγκέντρωσης (Pooling layers): Αυτά τα επίπεδα ακολουθούν τα συνελικτικά επίπεδα σε ένα συνελικτικό νευρωνικό δίκτυο (Convolutional Neural Network - CNN), χρησιμεύοντας στη μείωση της διαστατικότητας και της υπολογιστικής πολυπλοκότητας και βοηθώντας στην αποφυγή της υπερπροσαρμογής.
4. Επίπεδα κανονικοποίησης (Normalization layers): Αυτά τα επίπεδα, όπως η κανονικοποίηση παρτίδας (batch normalization), βοηθούν να γίνει το δίκτυο ταχύτερο και πιο σταθερό μέσω της κανονικοποίησης της εισόδου των επιπέδων [36].
5. Αναδρομικά επίπεδα (Recurrent layers): Χρησιμοποιούνται στα Αναδρομικά Νευρωνικά Δίκτυα (Recurrent Neural Networks - RNNs). Τα επίπεδα αυτά διατηρούν μια κρυφή κατάσταση που αποθηκεύει χρονικές πληροφορίες [37].

#### 4.1.2 Συναρτήσεις Ενεργοποίησης

Οι συναρτήσεις ενεργοποίησης εισάγουν μη γραμμικότητα στο δίκτυο, η οποία επιτρέπει στο μοντέλο να μαθαίνει σύνθετα μοτίβα. Ορισμένες από τις πιο συχνά χρησιμοποιούμενες συναρτήσεις ενεργοποίησης είναι οι εξής:

1. Σιγμοειδής (sigmoid): Η σιγμοειδής συνάρτηση συμπιέζει την είσοδο σε ένα εύρος μεταξύ 0 και 1. Χρησιμοποιήθηκε ευρέως στα πρώιμα νευρωνικά δίκτυα, αλλά έπεσε σε δυσμένεια λόγω προβλημάτων όπως η εξαφάνιση των κλίσεων [37].
2. ReLU (Rectified Linear Unit): Η συνάρτηση ReLU είναι σήμερα μια από τις πιο ευρέως χρησιμοποιούμενες συναρτήσεις ενεργοποίησης. Ορίζεται ως  $f(x) = \max(0, x)$ , καθιστώντας την υπολογιστικά αποδοτική. Ωστόσο, μπορεί να υποφέρει από ένα πρόβλημα γνωστό ως "πεθαμένη ReLU" [38].
3. Υπερβολική εφαπρομένη (Tanh): Η συνάρτηση tanh συμπιέζει την είσοδο σε ένα εύρος μεταξύ -1 και 1, παρέχοντας μια έξοδο με κέντρο το 0, η οποία μπορεί να διευκολύνει τη μάθηση για το επόμενο επίπεδο [39].

## 4.2 Ο Αλγόριθμος της Οπισθοδιάδοσης

Προκειμένου να κατανοήσουμε το φαινόμενο που τροφοδοτεί τα νευρωνικά δίκτυα, είναι ζωτικής σημασίας να εμβαθύνουμε στην έννοια της Οπισθοδιάδοσης (Backpropagation), μια

συντομογραφία για την “οπισθοδρομική διάδοση των σφαλμάτων”. Αυτός ο αλγόριθμος επιτρέπει στο νευρωνικό δίκτυο να μαθαίνει από τα σφάλματα και να βελτιώνει την απόδοσή του μέσω προσαρμογών των βαρών και των προκαταλήψεων [40].

Η οπισθοδιάδοση αποτελεί τη ραχοκοκαλιά της εκπαίδευσης στα νευρωνικά δίκτυα, επιτρέποντας τη βελτιστοποίηση των βαρών που στη συνέχεια ελαχιστοποιεί τη συνάρτηση κόστους. Η οπισθοδιάδοση είναι ουσιαστικά μια εφαρμογή του κανόνα της αλυσίδας για τον υπολογισμό των κλίσεων της συνάρτησης κόστους σε σχέση με τα βάρη του δικτύου [41].

### 4.2.1 Ο Κανόνας της Αλυσίδας

Ο κανόνας της αλυσίδας είναι ένας βασικός κανόνας παραγωγίσιμης που βοηθά στην αντιμετώπιση σύνθετων συναρτήσεων. Με μαθηματικούς όρους, εάν μια μεταβλητή  $z$  εξαρτάται από τη μεταβλητή  $y$ , η οποία εξαρτάται από τη  $x$ , δηλαδή  $z = z(y)$  και  $y = y(x)$ , τότε η παράγωγος της  $z$  ως προς  $x$  εκφράζεται ως εξής:

$$\frac{dz}{dx} = \frac{dz}{dy} \cdot \frac{dy}{dx} \quad (4.3)$$

Ο κανόνας της αλυσίδας αποτελεί την ουσία της οπισθοδιάδοσης και χρησιμοποιείται επανειλημμένα για τον υπολογισμό της παραγωγού της συνάρτησης κόστους σε σχέση με οποιοδήποτε βάρος στο δίκτυο.

### 4.2.2 Εμπρόσθια Διάδοση

Πριν ασχοληθούμε με την οπισθοδιάδοση, ας αναφερθούμε στην εμπρόσθια διάδοση. Σε ένα νευρωνικό δίκτυο, οι εισοδοί διαδίδονται μέσω του δικτύου για τη δημιουργία μιας εξόδου. Κατά τη διάρκεια αυτού του προς τα εμπρός περάσματος, το δίκτυο χρησιμοποιεί αρχικά τυχαία βάρη και προκαταλήψεις για τον υπολογισμό της εξόδου. Στη συνέχεια, η έξοδος συγκρίνεται με την αναμενόμενη έξοδο και η διαφορά δίνει το σφάλμα ή το κόστος [33].

Για ένα απλό δίκτυο με ένα κρυφό επίπεδο, η έξοδος  $O$  για μια είσοδο  $x$  υπολογίζεται ως εξής:

$$O = \sigma(W_2 \cdot \sigma(W_1 \cdot x + b_1) + b_2) \quad (4.4)$$

Εδώ,  $\sigma$  είναι η συνάρτηση ενεργοποίησης,  $W_1$  και  $W_2$  είναι τα βάρη και  $b_1$  και  $b_2$  είναι οι προκαταλήψεις [42].

### 4.2.3 Συνάρτηση Κόστους

Η συνάρτηση κόστους, γνωστή και ως συνάρτηση απωλειών, ποσοτικοποιεί το σφάλμα μεταξύ της πρόβλεψης του δικτύου και της πραγματικής εξόδου. Μια από τις συνήθως χρησιμοποιούμενες συναρτήσεις κόστους είναι το μέσο τετραγωνικό σφάλμα (MSE), το οποίο υπολογίζεται ως εξής για  $n$  νευρώνες εξόδου και  $m$  παραδείγματα εκπαίδευσης:

$$C = \frac{1}{2n} \sum_x \|y(x) - a(x)\|^2 \quad (4.5)$$

όπου  $y(x)$  είναι η αναμενόμενη έξοδος και  $a(x)$  είναι η έξοδος του δικτύου.

#### 4.2.4 Οπισθοδρομική Διάδοση

Η οπισθοδιάδοση αρχίζει μόλις ολοκληρωθεί το εμπρόσθιο πέρασμα μέσω του δικτύου και υπολογιστεί το κόστος. Ο στόχος είναι να προσαρμοστούν τα βάρη και οι προκαταλήψεις ώστε να ελαχιστοποιηθεί το κόστος. Η προσαρμογή γίνεται με τον υπολογισμό της κλίσης της συνάρτησης κόστους σε σχέση με κάθε βάρος και προκατάληψη στο δίκτυο [43].

Ο αλγόριθμος οπισθοδιάδοσης υπολογίζει την κλίση σφάλματος του επίπεδο εξόδου ( $\delta^L$ ) και στη συνέχεια διαδίδει αυτά τα σφάλματα προς τα πίσω μέσω του δικτύου. Για κάθε επίπεδο " $l$ " που πηγαίνει προς τα πίσω από το επίπεδο εξόδου στο επίπεδο εισόδου, το σφάλμα  $\delta^l$  υπολογίζεται ως εξής:

$$\delta^l = ((W^{l+1})^T \delta^{l+1}) \odot \sigma'(z^l) \quad (4.6)$$

Εδώ,  $\sigma'$  είναι η παράγωγος της συνάρτησης ενεργοποίησης,  $W$  είναι τα βάρη και  $z$  είναι η σταθμισμένη είσοδος στους νευρώνες του επιπέδου  $l$ . Το  $\odot$  δηλώνει το γινόμενο *Hadamard* ή τον πολλαπλασιασμό κατά στοιχείο.

#### 4.2.5 Ενημέρωση Βαρών και Προκαταλήψεων

Αφού το σφάλμα διαδοθεί προς τα πίσω μέσω του δικτύου, ενημερώνονται τα βάρη και οι προκαταλήψεις. Αυτό γίνεται με τη χρήση της καθόδου κλίσης (gradient descent), όπου τα βάρη και οι προκαταλήψεις προσαρμόζονται ανάλογα με την αρνητική κλίση της συνάρτησης κόστους στο τρέχον σημείο [44].

Για κάθε βάρος  $w$  και προκατάληψη  $b$ , ο κανόνας ενημέρωσης έχει ως εξής:

$$w' = w - \eta \frac{\partial C}{\partial w} \quad (4.7)$$

$$b' = b - \eta \frac{\partial C}{\partial b} \quad (4.8)$$

Εδώ, " $\eta$ " είναι ο ρυθμός μάθησης, μια παράμετρος που καθορίζει το μέγεθος των βημάτων που γίνονται κατά την κάθοδο [39].

Η οπισθοδιάδοση, σε συνδυασμό με την κάθοδο κλίσης, επιτρέπει στα νευρωνικά δίκτυα να μαθαίνουν από τα λάθη τους και να βελτιώνουν την απόδοσή τους με την πάροδο του χρόνου. Η κατανόηση αυτού του θεμελιώδους αλγορίθμου είναι απαραίτητη για κάθε άτομο που δουλεύει πάνω στη βαθιά μάθηση.

### 4.3 Perceptron Πολλαπλών Επιπέδων

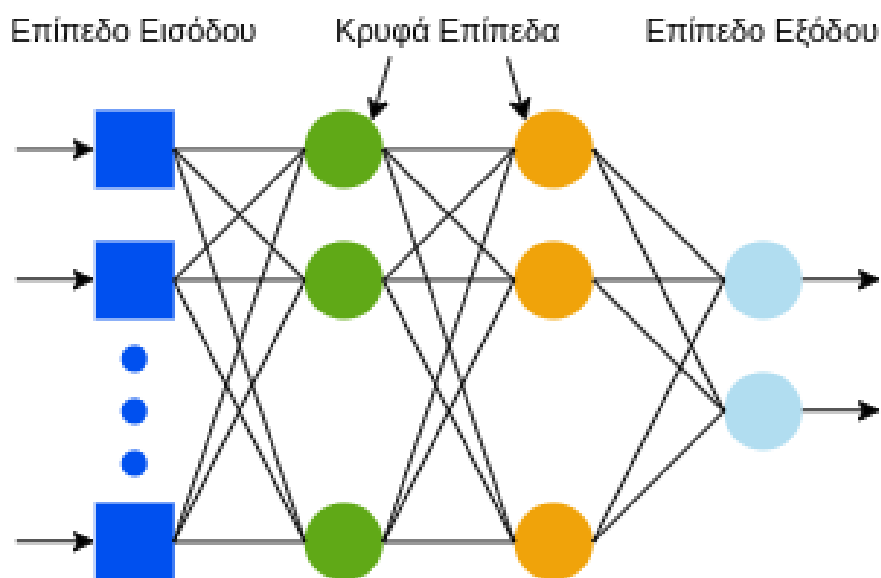
Το Perceptron πολλαπλών επιπέδων (Multi-Layer Perceptron - MLP) είναι μια κατηγορία τεχνητών νευρωνικών δικτύων πρόσθιας τροφοδότησης, ένα βασικό συστατικό στον τομέα της μηχανικής μάθησης και της βαθιάς μάθησης. Μπορεί να θεωρηθεί ως γενίκευση του απλού Perceptron, ενός πρώιμου μοντέλου νευρωνικού δικτύου [45].



Τα MLP μπορούν να εφαρμοστούν σε μια ευρεία ποικιλία εργασιών, όπως η παλινδρόμηση, η ταξινόμηση και η εξαγωγή χαρακτηριστικών. Ωστόσο, έχουν επίσης αρκετούς περιορισμούς, όπως ο κίνδυνος υπερπροσαρμογής και οι δυσκολίες στην εκπαίδευση με πολλά κρυφά επίπεδα. Οι πρόσφατες προόδους στη βαθιά μάθηση, όπως η εμφάνιση τεχνικών κανονικοποίησης (π.χ. dropout) και άλλων αρχιτεκτονικών, όπως τα συνελκτικά νευρωνικά δίκτυα (Convolutional Neural Network - CNN) και τα αναδρομικά νευρωνικά δίκτυα (Recurrent Neural Network - RNN), έχουν αμβλύνει ορισμένα από αυτά τα ζητήματα.

### 4.3.1 Αρχιτεκτονική MLP

Η αρχιτεκτονική ενός Perceptron πολλαπλών επιπέδων περιλαμβάνει τουλάχιστον τρία επίπεδα κόμβων (ή νευρώνων): ένα επίπεδο εισόδου, ένα ή περισσότερα "κρυφά" επίπεδα και ένα επίπεδο εξόδου. Κάθε κόμβος σε ένα επίπεδο συνδέεται με όλους τους κόμβους στο επόμενο επίπεδο, αποτελώντας μια πλήρως συνδεδεμένη δομή. Κάθε κόμβος αντιπροσωπεύει έναν τεχνητό νευρώνα και εφαρμόζει μια συνάρτηση ενεργοποίησης στην καθαρή είσοδό του για να παράγει την καθαρή έξοδό του.



Σχήμα 4.1: Σχηματική αναπαράσταση ενός MLP με 2 κρυφά επίπεδα.

Τυπικά, ένα πλήρως συνδεδεμένο επίπεδο ενός MLP μπορεί να αναπαρασταθεί από την ακόλουθη εξίσωση:

$$y = f(Wx + b) \quad (4.9)$$

όπου :

- $x$  είναι το διάνυσμα εισόδου
- $W$  είναι ο πίνακας βαρών
- $b$  είναι το διάνυσμα μεροληψίας
- $f$  είναι η συνάρτηση ενεργοποίησης

- $y$  είναι το διάνυσμα εξόδου

Το επίπεδο εισόδου λαμβάνει δεδομένα εισόδου. Τα κρυφά επίπεδα εκτελούν υπολογισμούς και μεταδίδουν τις πληροφορίες στο επίπεδο εξόδου. Το επίπεδο εξόδου παράγει τις τελικές εξόδους. Σημείωση ότι το "βάθος" του MLP εξαρτάται από τον αριθμό των κρυφών επιπέδων και το "πλάτος" εξαρτάται από τον αριθμό των νευρώνων σε κάθε επίπεδο.

### 4.3.2 Συναρτήσεις Ενεργοποίησης

Κάθε κόμβος στο MLP εφαρμόζει μια συνάρτηση ενεργοποίησης στην είσοδο του δικτύου του για να παράγει έξοδο. Μπορεί να χρησιμοποιηθεί ένα ευρύ φάσμα συναρτήσεων ενεργοποίησης, όπως ενδεικτικά: σιγμοειδής, υπερβολική εφαπτομένη, ReLU (Rectified Linear Unit) και softmax. Η επιλογή της συνάρτησης ενεργοποίησης εξαρτάται συχνά από τη συγκεκριμένη εφαρμογή και τα χαρακτηριστικά των δεδομένων εισόδου.

Για παράδειγμα, η σιγμοειδής συνάρτηση,  $f(z) = 1/(1 + e^{-z})$ , και η συνάρτηση υπερβολικής εφαπτομένης,  $f(z) = \tanh(z)$ , χρησιμοποιούνται συχνά για προβλήματα δυαδικής ταξινόμησης, καθώς συμπιέζουν τις εισόδους τους στο εύρος  $(0, 1)$  και  $(-1, 1)$ , αντίστοιχα. Η συνάρτηση ReLU,  $f(z) = \max(0, z)$ , χρησιμοποιείται ευρέως σε μοντέλα βαθιάς μάθησης λόγω της υπολογιστικής της αποδοτικότητας και της αποτελεσματικότητάς της στην άμβλυνση του προβλήματος της εξαφανιζόμενης κλίσης (vanishing gradient) [46]. Η συνάρτηση softmax είναι ιδιαίτερα χρήσιμη στο επίπεδο εξόδου για προβλήματα ταξινόμησης πολλών κλάσεων, καθώς δίνει μια πιθανολογική ερμηνεία στους νευρώνες εξόδου.

## 4.4 U-Net

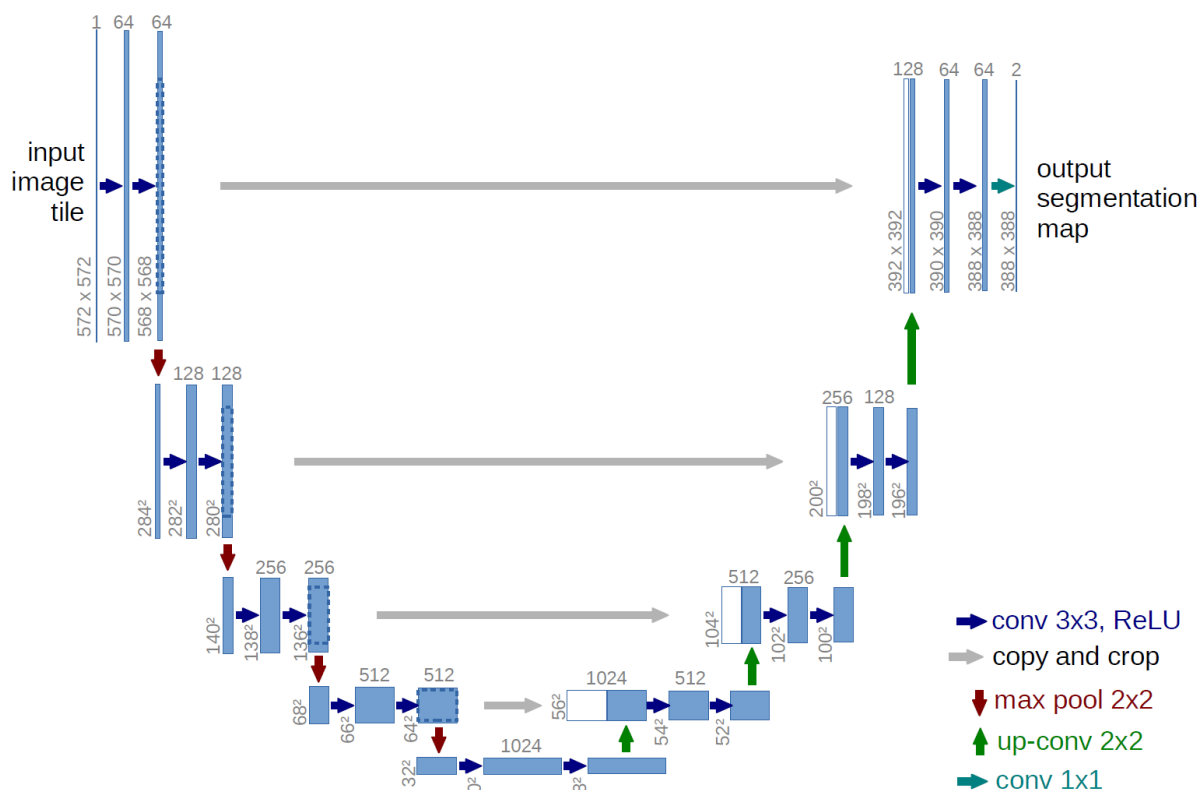
Το μοντέλο U-Net είναι ένας τύπος συνελκτικού νευρωνικού δικτύου (CNN) που είναι ιδιαίτερα κατάλληλο για την κατάτμηση εικόνων, καθιστώντας το ιδιαίτερα πολύτιμο στον τομέα της ιατρικής απεικόνισης [1]. Σε αντίθεση με τα παραδοσιακά συνελκτικά δίκτυα, το U-Net χαρακτηρίζεται από τη μοναδική αρχιτεκτονική του, η οποία μοιάζει με το γράμμα "U", εξ ου και το όνομά του.

### 4.4.1 Αρχιτεκτονική του U-Net

Η αρχιτεκτονική U-Net αποτελείται από ένα μονοπάτι συρρίκνωσης και ένα μονοπάτι επέκτασης, το οποίο προσδίδει στο δίκτυο το σχήμα U [1]. Αυτή η συγκεκριμένη αρχιτεκτονική επιτρέπει στο μοντέλο να εκμεταλλεύεται τόσο τα τοπικά χαρακτηριστικά όσο και τα ευρύτερα συμφραζόμενα χαρακτηριστικά από την εικόνα εισόδου.

#### Μονοπάτι Συρρίκνωσης

Το μονοπάτι συρρίκνωσης (contracting path) ακολουθεί την τυπική αρχιτεκτονική ενός συνελκτικού δικτύου. Αποτελείται από επαναλαμβανόμενες εφαρμογές δύο  $3 \times 3$  συνελκτικών επιπέδων (χωρίς συμπλήρωση - padding), κάθε μία από τις οποίες ακολουθείται από



Σχήμα 4.2: Σχηματική αναπαράσταση του U-Net.

για μια συνάρτηση ενεργοποίησης ReLU και μια  $2 \times 2$  max pooling πράξη με  $\text{stride} = 2$  για υποδειγματοληψία (downsampling) [1]. Μετά από κάθε πράξη υποδειγματοληψίας, ο αριθμός των καναλιών χαρακτηριστικών διπλασιάζεται.

### Μονοπάτι Επέκτασης

Το μονοπάτι επέκτασης (expanding path) αποτελείται από μια αναδειγματοληψία (up-sampling) του χάρτη χαρακτηριστικών ακολουθούμενο από ένα συνελκτικό επίπεδο  $2 \times 2$  ("up-convolution"), μια συνένωση με τον αντίστοιχα περικομμένο χάρτη χαρακτηριστικών από το μονοπάτι συρρίκνωσης και δύο συνελκτικά επίπεδα  $3 \times 3$ , η κάθε μία ακολουθούμενη από μια ενεργοποίηση ReLU [1]. Η περικοπή είναι απαραίτητη λόγω της απώλειας εικονοστοιχείων που βρίσκονται στα όρια της εικόνας σε κάθε συνελκτικό επίπεδο. Στο τελευταίο επίπεδο, χρησιμοποιείται ένα  $1 \times 1$  συνελκτικό επίπεδο για την αντιστοίχιση κάθε διανύσματος χαρακτηριστικών  $64$  διαστάσεων στον επιθυμητό αριθμό κλάσεων.

### Συνδέσεις Παράκαμψης

Ένα βασικό χαρακτηριστικό του U-Net είναι η παρουσία συνδέσεων παράκαμψης (skip connections), όπου η έξοδος κάθε επιπέδου στο μονοπάτι συρρίκνωσης τροφοδοτείται απευθείας στο αντίστοιχο επίπεδο στο μονοπάτι επέκτασης. Αυτό επιτρέπει στο μοντέλο να αξιοποιήσει τα χαρακτηριστικά υψηλής ανάλυσης από το μονοπάτι συρρίκνωσης απευθείας κατά τη διαδικασία αναδειγματοληψίας, βοηθώντας έτσι το δίκτυο να ανακτήσει τις λεπτομέρειες που μπορεί να χαθούν κατά την κωδικοποίηση των χαρακτηριστικών [1].

### **Εκπαίδευση του μοντέλου U-Net**

Το μοντέλο U-Net εκπαιδεύεται σε λίγες εικόνες εκπαίδευσης και χρησιμοποιεί επαύξηση δεδομένων (data augmentation) για να αποτρέψει την υπερπροσαρμογή [1]. Αυτό είναι ιδιαίτερα επωφελές για βιοϊατρικές εφαρμογές, όπου τα επισημασμένα δεδομένα μπορεί να είναι σπάνια.

Η εκπαίδευση του U-Net περιλαμβάνει την ελαχιστοποίηση μιας συνάρτησης σφάλματος, η οποία στην αρχική της εφαρμογή ήταν η softmax κατά εικονοστοιχείο πάνω στον τελικό χάρτη χαρακτηριστικών σε συνδυασμό με τη συνάρτηση απώλειας cross-entropy [1].

Το τελικό επίπεδο του U-Net παράγεται μια ταξινόμηση κατά εικονοστοιχείο, η οποία είναι ο χάρτης κατάτμησης της εικόνας εισόδου.

Η αρχιτεκτονική του U-Net του επιτρέπει να κάνει ακριβείς προβλέψεις με τη χρήση αναδειγματοληψίας και συνδέσεων παράκαμψης, διατηρώντας παράλληλα υψηλή ανάλυση καθ' όλη τη διάρκεια λόγω της χρήσης μη συμπληρωμένων συνελκτικών επιπέδων [1].

**Μέρος **

**Πρακτικό Μέρος**

---



## Κεφάλαιο 5

### Υλοποίηση - Πειράματα

---

Στο κεφάλαιο αυτό, παρέχουμε μια λεπτομερή περιγραφή των πειραμάτων μας, καθώς αυτά αποτελούν κομβικό μέρος της έρευνάς μας και υποστηρίζουν την προσέγγισή μας με εμπειρικά στοιχεία, δοκιμάζοντας την ευρωστία και την αξιοπιστία των προτεινόμενων μοντέλων. Η πειραματική μας διαδικασία μπορεί σε γενικές γραμμές να χωριστεί σε διάφορα στάδια: συλλογή και επισήμανση δεδομένων, επιλογή του πλαισίου εργασίας (framework), επιλογή και εφαρμογή μοντέλων και ανάλυση των αποτελεσμάτων.

#### 5.1 Δεδομένα

Τα θεμέλια της μελέτης μας βασίζονται σε δύο τύπους δεδομένων εικόνας: αεροφωτογραφίες και δορυφορικές εικόνες. Οι αεροφωτογραφίες αυτές προέρχονται από την περιοχή Limpopo της Νότιας Αφρικής το 2010 και έχουν επισημανθεί περίπου 15,000 σημεία με τις ετικέτες woody, non-woody, non-vegetated στα πλαίσια της εργασίας [47]. Καθώς μας ενδιαφέρει η αναγνώριση ξυλώδους βλάστησης, χωρίσαμε τα σημεία σε 2 κλάσεις, ξηλώδης βλάστηση και μή ξηλώδης/χωρίς βλάστηση. Κάθε σημείο βρίσκεται στο κέντρο μίας αεροφωτογραφίας 90x90 μέτρων και κάθε εικονοστοιχείο αντιστοιχεί σε περιοχή περίπου 0.5x0.5 μέτρων.

Οι δορυφορικές εικόνες είναι πολύ χαμηλότερης ανάλυσης, αλλά έχουμε στη διαθεσή μας πολύ περισσότερες και σε διάφορες χρονικές στιγμές από το 1988 έως το 2018 και κάθε εικονοστοιχείο τους αντιστοιχεί σε περιοχή 30x30 μέτρων.

Οι αεροφωτογραφίες είναι εικόνες τύπου RGB και επομένως κάθε εικονοστοιχείο αποτελείται από τρεις τιμές. Επιπλέον, χρησιμοποιήσαμε το Δείκτη Ορατής Βλάστησης (Visible Vegetation Index - VVI) ο οποίος χρησιμοποιείται για την εκτίμηση φωτοσυνθετικής δραστηριότητας και ορίζεται ως:

$$VVI = \left[ \left( 1 - \frac{R - R_0}{R + R_0} \right) \cdot \left( 1 - \frac{G - G_0}{G + G_0} \right) \cdot \left( 1 - \frac{B - B_0}{B + B_0} \right) \right] \quad (5.1)$$

όπου  $R$ ,  $G$ ,  $B$  είναι οι αντίστοιχες τιμές του εικονοστοιχείου, και  $R_0 = 30$ ,  $G_0 = 50$ ,  $B_0 = 1$  είναι οι τιμές RGB που χρησιμοποιούνται ως σημείο αναφοράς για το πράσινο χρώμα. Επομένως καταλήγουμε με 4 τιμές για κάθε σημείο εκπαίδευσης.

Όσον αφορά τις δορυφορικές εικόνες, περιέχουν 54 τιμές ανά εικονοστοιχείο.

## 5.2 Πλαίσια Εργασίας

Όσον αφορά τα πλαίσια εργασίας με τα οποία εργαστήκαμε, χρησιμοποιήσαμε το Scikit-Learn (sk-learn) και το PyTorch, τα οποία χρησιμοποιούνται ευρέως στην κοινότητα της μηχανικής μάθησης. Το Scikit-Learn [48] χρησιμοποιήθηκε για τις παραδοσιακές μεθόδους μηχανικής μάθησης, ενώ το PyTorch [49] μας επέτρεψε να υλοποιήσουμε ένα προηγμένο μοντέλο βαθιάς μάθησης, το U-Net. Το χαρακτηριστικό του δυναμικού γράφου υπολογισμών του PyTorch ήταν ιδιαίτερα χρήσιμο στην πρωτοτυποποίηση και την αποσφαλμάτωση, ενώ η ολοκληρωμένη βιβλιοθήκη του Scikit-Learn διευκόλυνε την προεπεξεργασία, την επιλογή μοντέλων και τον υπολογισμό μετρικών.

## 5.3 Πειράματα

Στην εργασία χρησιμοποιείται μια σειρά μοντέλων, καθένα από τα οποία επιλέχθηκε για τη συνάφεια του με τη φύση των δεδομένων και του προβλήματός μας. Ο πειραματισμός μας περιλάμβανε παραδοσιακά μοντέλα μηχανικής μάθησης, όπως αυτά περιγράφηκαν στην Ενότητα 3.1. Χρησιμοποιήσαμε επίσης μοντέλα βαθιάς μάθησης, συγκεκριμένα ένα δίκτυο Perceptron Πολλαπλών Επιπέδων MLP για την κατηγοριοποίηση των σημείων και ένα δίκτυο U-Net για τη δημιουργία μασκών όπως αυτά περιγράφηκαν στις Ενότητες 4.3 και 4.4. Τα μοντέλα αξιολογήθηκαν με βάση την ακρίβεια πρόβλεψης και τη γενίκευση, καθώς δεν μας ενδιέφερε ιδιαίτερα η υπολογιστική τους αποτελεσματικότητα.

### 5.3.1 Δημιουργία Μασκών από Σημεία σε Αεροφωτογραφίες

Για την έρευνά μας, η κύρια πηγή δεδομένων εκπαίδευσης ήταν μια συλλογή από αεροφωτογραφίες σχολιασμένες με 15.000 σημεία. Τα σημεία αυτά χρησίμευαν ως ετικέτες που αντιπροσώπευαν τρεις διακριτές κατηγορίες: ξυλώδη βλάστηση, μη ξυλώδη βλάστηση και περιοχές χωρίς βλάστηση. Αυτά τα επισημασμένα σημεία αποτέλεσαν ανεκτίμητη πηγή για την έρευνά μας, παρέχοντας στα μοντέλα ταξινόμησης ένα επαρκές σύνολο δεδομένων για την εκμάθηση των διακριτικών χαρακτηριστικών των διαφόρων τύπων κάλυψης γης.

Εκπαιδεύσαμε διάφορα μοντέλα πάνω στα ίδια δεδομένα και διαλέξαμε το καλύτερο από αυτά όπως φαίνεται στον πίνακα 5.1

Μοντέλα	Accuracy	Precision	Recall	F1
Random Forest	0.8056	0.8068	0.8034	0.8042
Decision Tree	0.7111	0.7105	0.7096	0.7098
Gradient Boosting	0.8361	0.8381	0.8339	0.8349
<b>Ada Boost</b>	<b>0.8389</b>	<b>0.8413</b>	<b>0.8365</b>	<b>0.8376</b>
MLP	0.7917	0.8029	0.8034	0.7911
K-Neighbors	0.8056	0.8056	0.8034	0.8047

Πίνακας 5.1: Τα μοντέλα που δοκιμάστηκαν και οι επιδόσεις τους

Στο σύνθετο τοπίο της ανάπτυξης μοντέλων μηχανικής μάθησης, η επικύρωση της απόδοσης του μοντέλου είναι υψίστης σημασίας για να διασφαλιστεί η αξιοπιστία και η γενίκευση.



Η ικανότητα ενός μοντέλου να αποδίδει καλά σε νέα δεδομένα αποτελεί κρίσιμο μέτρο της αποτελεσματικότητας και της χρηστικότητας του σε πραγματικές εφαρμογές.

Για τον σκοπό αυτό, επιλέξαμε να εφαρμόσουμε τη διασταυρωμένη επικύρωση 10 φορές (10-fold cross-validation). Σε αυτή την προσέγγιση, το σύνολο δεδομένων χωρίζεται σε δέκα ίσα υποσύνολα. Το μοντέλο εκπαιδεύεται δέκα φορές, χρησιμοποιώντας κάθε φορά εννέα υποσύνολα ως δεδομένα εκπαίδευσης και το υπόλοιπο υποσύνολο ως δεδομένα επικύρωσης. Η διαδικασία αυτή διασφαλίζει ότι κάθε σημείο δεδομένων χρησιμοποιείται για επικύρωση ακριβώς μία φορά, παρέχοντας μια ολοκληρωμένη αξιολόγηση της απόδοσης του μοντέλου.

Ο απώτερος στόχος αυτού του σταδίου δεν ήταν μόνο η διάκριση των 2 κατηγοριών βλάστησης αλλά και η δημιουργία ολοκληρωμένων μασκών του συνόλου των περιοχών που αναπαρίστανται στις αεροφωτογραφίες. Αυτές οι μάσκες χρησιμεύουν ως οπτικές αναπαραστάσεις των ταξινομήσεων, επιτρέποντάς μας να αποτυπώσουμε την πολυπλοκότητα του τοπίου σε μια μορφή που είναι οπτικά ερμηνεύσιμη.

Μπορούμε να δούμε παραδείγματα μασκών που δημιουργήθηκαν με αυτό το μοντέλο στην εικόνα 5.1.

### 5.3.2 Δημιουργία Βελτιωμένων Μασκών με U-Net

Παρόλο που το αρχικό μας μοντέλο επιδεικνύει ελπιδοφόρα αποτελέσματα στη διαφοροποίηση μεταξύ ξυλώδους και μη ξυλώδους/όχι βλάστησης, υπάρχουν παρατηρήσιμες ανακρίβειες που δεν μπορούν να αγνοηθούν. Η πολυπλοκότητα που υπάρχει στις αεροφωτογραφίες, οδηγεί μερικές φορές σε λανθασμένη ταξινόμηση.

Υπό το πρίσμα αυτών των προκλήσεων, διερευνούμε την υιοθέτηση της αρχιτεκτονικής U-Net, γνωστής για την αποτελεσματικότητά της στην κατάτμηση εικόνων, για τη βελτίωση της ταξινόμησης της ξυλώδους. Η αρχιτεκτονική του U-Net, που χαρακτηρίζεται από τις συμμετρικές διευρυνόμενες και συρρικνούμενες διαδρομές του είναι ικανή να περιγράφει περίπλοκα μοτίβα και δομές μέσα σε εικόνες. Αυτό το καθιστά κατάλληλη λύση για την ενίσχυση της κατάτμησης μικρότερων, ομαδοποιημένων ξυλώδων φυτών που προηγουμένως είχαν ταξινομηθεί λανθασμένα ή αγνοηθεί.

Αξιοποιούμε τις μάσκες που δημιουργούνται από το αρχικό μοντέλο ως δεδομένα εκπαίδευσης για το U-Net. Αυτές οι μάσκες, αν και ατελείς, παρέχουν ένα θεμελιώδες σύνολο δεδομένων που περιλαμβάνει τα γενικά χαρακτηριστικά της ξυλώδους και μη ξυλώδους βλάστησης. Το μοντέλο U-Net έχει σχεδιαστεί για να μαθαίνει από αυτά τα χαρακτηριστικά, βελτιώνοντας την ικανότητά του να διακρίνει και να τμηματοποιεί τις πολύπλοκες, ομαδοποιημένες δομές της ξυλώδους βλάστησης με αυξημένη ακρίβεια.

Η χρήση του U-Net σε αυτό το πλαίσιο ξεπερνά τους περιορισμούς του αρχικού μοντέλου, προσφέροντας βελτιωμένη οριοθέτηση μεταξύ της ξυλώδους και της μη ξυλώδους βλάστησης, ιδίως σε περιοχές όπου η βλάστηση αυτή είναι πυκνά συσσωρευμένη και διαπλεκόμενη.

Η εκπαίδευσή του έγινε πάνω σε 87 εικόνες με ρυθμό μάθησης  $1e-5$  με χρήση του βελτιστοποιητή Adam και συνάρτηση απώλειας Cross Entropy για 10 εποχές εκπαίδευσης.

Παρακάτω μπορούμε να δούμε τη συνάρτηση απώλειας κατά τη διάρκεια της εκπαίδευσης:



Σχήμα 5.2: Συνάρτηση απώλειας κατά την εκπαίδευση U-Net

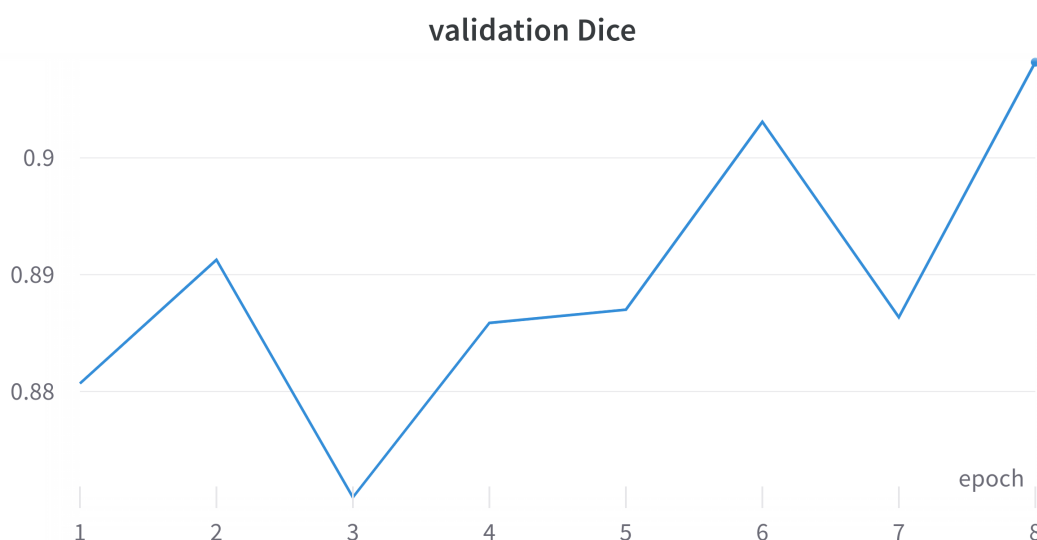
Επιπλέον, χρησιμοποιήσαμε 9 εικόνες από το σύνολο δεδομένων για επαλήθευση. Για αυτό το σκοπό, χρησιμοποιήσαμε τη βαθμολογία επαλήθευσης Dice ως κύριο μέτρο απόδοσης όπως φαίνεται στην Εικόνα 5.3. Η βαθμολογία Dice, που συχνά αναφέρεται ως συντελεστής ομοιότητας Dice (Dice Similarity Coefficient - DSC), είναι ένας δείκτης χωρικής επικάλυψης που μετρά την ομοιότητα μεταξύ δύο δειγμάτων που παίρνουν δυαδικές τιμές. Δεδομένων δύο δυαδικών μασκών,  $A$  (η αληθινή μάσκα) και  $B$  (προβλεπόμενη μάσκα), η βαθμολογία Dice υπολογίζεται ως εξής:

$$\text{Dice}(A, B) = \frac{2 \times |A \cap B|}{|A| + |B|} \quad (5.2)$$

όπου:

- $|A \cap B|$  αναπαριστά τα κοινά εικονοστοιχεία που "συμφωνούν" μεταξύ της αληθινής μάσκας  $A$  και της μάσκας  $B$ .
- $|A|$  και  $|B|$  είναι ο συνολικός αριθμός εικονοστοιχείων στην αληθινή μάσκα και στην προβλεπόμενη μάσκα αντίστοιχα.

Μπορούμε εύκολα να διαπιστώσουμε στην πράξη την καλύτερη πιστότητα των νέων μασκών από την εικόνα 5.4.



Σχήμα 5.3: Συντελεστής ομοιότητας Dice κατά το στάδιο επαλήθευσης του U-Net

### 5.3.3 Εκπαίδευση Μοντέλου Πρόβλεψης Ποσοτών Βλάστησης από Δορυφορικές Εικόνες

Έχοντας εκπαιδεύσει το U-Net μπορούμε τώρα να παράγουμε νέες, καλύτερες μάσκες για τις περιοχές που μας ενδιαφέρουν. Για κάθε περιοχή επομένως, παράγουμε μία μάσκα η οποία χρησιμοποιείται για να υπολογίσουμε το ποσοστό κάλυψης ξυλώδους βλάστησης. Αυτό γίνεται απλά μετρώντας τα μαύρα εικονοστοιχεία στις μάσκες και προκύπτει το ποσοστό κάλυψης από τον τύπο:

$$woody\_coverage = \frac{black\_pixels}{total\_pixels} \quad (5.3)$$

Όπως έχουμε ήδη αναφέρει, διαθέτουμε αεροφωτογραφίες μεγέθους 90x90 μέτρων και κάθε εικονοστοιχείο στη δορυφορική εικόνα αντιστοιχεί σε περιοχή 30x30 μέτρων. Συνεπώς, για κάθε αεροφωτογραφία της οποίας έχουμε υπολογίσει τη μάσκα, παίρνουμε τις τιμές των 9 εικονοστοιχείων της δορυφορικής εικόνας που της αντιστοιχούν. Άρα τελικά φτιάχνουμε τα δεδομένα εκπαίδευσης με στόχο το ποσοστό ξυλώδους βλάστησης και χαρακτηριστικά τις τιμές των 9 εικονοστοιχείων που αντιστοιχούν στη περιοχή, συνολικά λοιπόν 486 τιμές.

Η βιβλιοθήκη sklearn διαθέτει τα μοντέλα που χρησιμοποιήσαμε παραπάνω και για εργασίες παλινδρόμησης. Αρχικά δοκιμάσαμε να εκπαιδεύσουμε το μοντέλο με τα ποσοστά βλάστησης όπως αυτά προκύπτουν από τις μάσκες του U-Net, με αποτελέσματα όπως φαίνονται στον Πίνακα 5.2

Προσπαθώντας να διερευνήσουμε πώς μπορεί να βελτιωθεί το σφάλμα στις προβλέψεις, ανατρέξαμε στα δεδομένα, τις παραγόμενες μάσκες. Αν και το U-Net ήταν πολύ καλύτερο στην κατάτμηση αντικειμένων στις εικόνες, σε περιπτώσεις όπου δεν υπήρχε βλάστηση, το U-Net δημιουργούσε μάσκες που είχαν κοντά 100% βλάστηση. Αντίθετα, οι μάσκες που είχαν παραχθεί από το αρχικό μοντέλο είχαν λιγότερο αυτό το πρόβλημα καθώς η ταξινόμηση

Μοντέλα	MAE	MSE
<b>Random Forest</b>	<b>0.3184</b>	<b>0.1371</b>
Decision Tree	0.3637	0.2546
Gradient Boosting	0.3221	0.1405
Ada Boost	0.3544	0.1533
MLP	39.6259	2338.8642

Πίνακας 5.2: Μοντέλα για ποσοστά βλάστησης από μάσκες U-Net και οι μετρικές τους

Μοντέλα	MAE	MSE
<b>Random Forest</b>	<b>0.2388</b>	<b>0.0786</b>
Decision Tree	0.3074	0.1551
Gradient Boosting	0.2444	0.0826
Ada Boost	0.2536	0.0852
MLP	27.8208	1248.1066

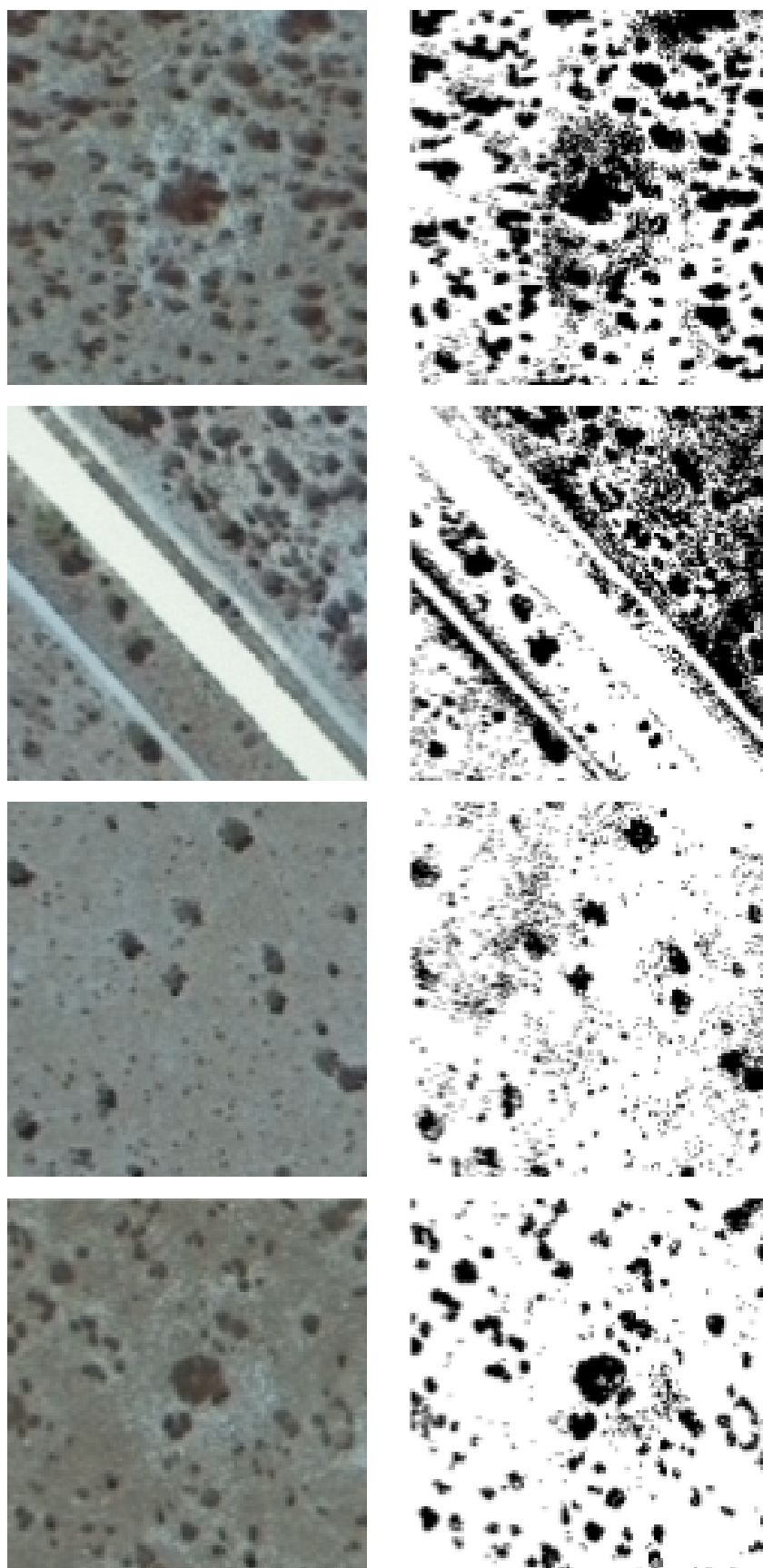
Πίνακας 5.3: Μοντέλα για ποσοστά βλάστησης από φιλτραρισμένες μάσκες U-Net και οι μετρικές τους

γινόταν ανα εικονοστοιχείο. Στις εικόνες 5.5 παρακάτω φαίνεται καλύτερα το πρόβλημα.

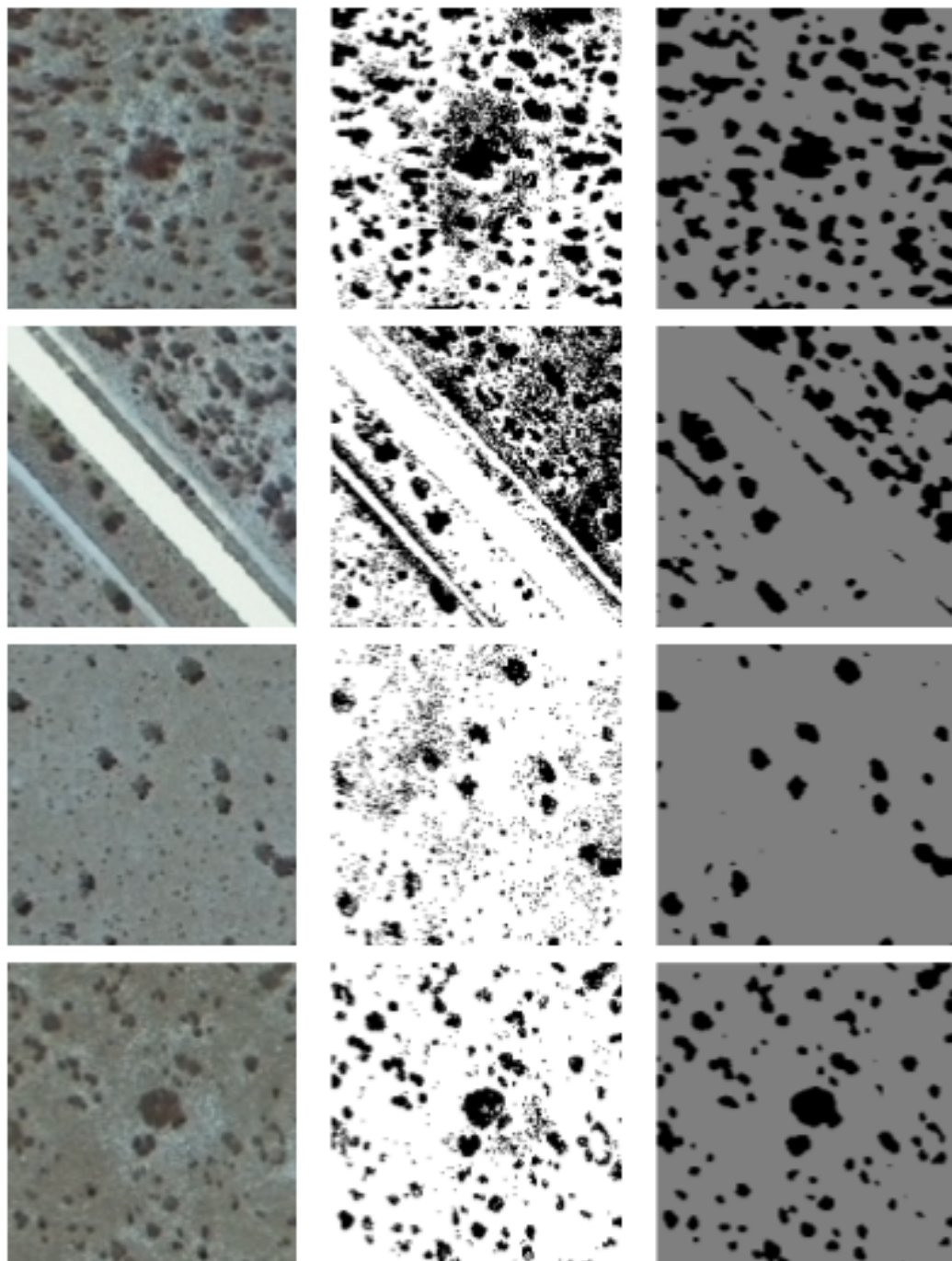
Για να επιλύσουμε αυτό το πρόβλημα προσπαθήσαμε να εκμεταλλευτούμε αυτή τη διαφορά στις 2 μάσκες. Έτσι, χρησιμοποιήσαμε τις μάσκες για να φιλτράρουμε αυτά τα δεδομένα από το σύνολο δεδομένων μας. Με αυτόν τον τρόπο, ξεσκαρτάραμε σχετικά εύκολα περίπου το 15% των δεδομένων που παρουσίαζαν μεγάλη διαφορά (μεγαλύτερη του 50%) στα ποσοστά βλάστησης μεταξύ των 2 μασκών, όταν το ποσοστό αυτό ξεπερνούσε το 95% στις μάσκες U-Net.

Έτσι τελικά, επανεκπαιδεύσαμε τα μοντέλα μας και είχαμε τα εξής αποτελέσματα όπως φαίνονται στον Πίνακα 5.3

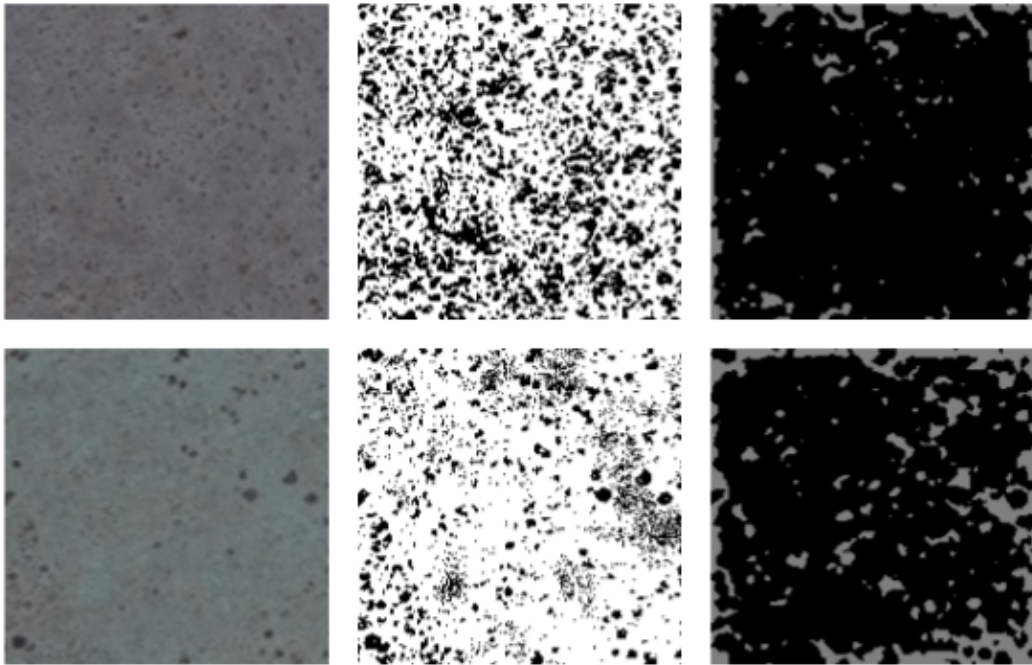
Παρατηρούμε ότι με το φιλτράρισμα των δεδομένων ότι βελτιώθηκαν σημαντικά όλα τα μοντέλα. Όπως και για τα μοντέλα ταξινόμησης εφαρμόσαμε τη διασταυρωμένη επικύρωση 10 φορές (10-fold cross-validation), για να επιβεβαιώσουμε την αποτελεσματικότητα των μοντέλων.



Σχήμα 5.1: Αεροφωτογραφίες και οι παραγόμενες μάσκες τους



Σχήμα 5.4: Αεροφωτογραφίες και οι παραγόμενες μάσκες τους απο το αρχικό μοντέλο και το U-Net



Σχήμα 5.5: Προβληματικές μάσκες U-Net





## Κεφάλαιο **6**

# Εφαρμογή Οπτικοποίησης και Βελτίωσης Δεδομένων

---

### 6.1 Στόχος

Η ανάπτυξη μιας αποτελεσματικής λύσης εξαρτάται συχνά από τη δημιουργία πρακτικών εργαλείων που επιταγχύνουν τη διαδικασία και βελτιώνουν τα αποτελέσματα. Στην προσπάθειά μας να αξιοποιήσουμε τη μηχανική μάθηση για την ακριβή ανάλυση της βλάστησης από αεροφωτογραφίες, αναπτύξαμε μια πρωτότυπη εφαρμογή που μπορεί να χρησιμεύσει ως κρίσιμο στοιχείο της βελτίωσης της μεθοδολογίας μας. Αυτό το διαδραστικό εργαλείο έχει σχεδιαστεί για να ανοίγει οποιαδήποτε μάσκα αεροφωτογραφίας και να επιτρέπει αλλαγές στις επισημάνσεις της.

Η εφαρμογή αυτή εξυπηρετεί δύο σκοπούς. Πρώτον, επιτρέπει την ενδεδειγμένη επιθεώρηση των παραγόμενων μασκών και διευκολύνει τη λεπτομερή ρύθμιση των δεδομένων με βάση την ανθρώπινη εμπειρία και αξιολόγηση. Δεύτερον, παρέχει τη δυνατότητα διόρθωσης τυχόν ανακριβειών ή ασυνεπειών στους αρχικούς σχολιασμούς. Αυτή η διαδραστική λειτουργία διασφαλίζει ότι τα μοντέλα μας εκπαιδεύονται συνεχώς με τα πιο ακριβή δεδομένα, βελτιώνοντας τη συνολική απόδοση. Όπως είδαμε άλλωστε στο προηγούμενο κεφάλαιο ένα σημαντικό βάρος της ευθύνης για τα σφάλματα των μοντέλων πέφτει στις λανθασμένες μάσκες.

Οι επόμενες ενότητες θα εμβαθύνουν στα χαρακτηριστικά, τη διαδικασία ανάπτυξης και τις πιθανές βελτιώσεις αυτής της πρωτότυπης εφαρμογής, παρέχοντας έτσι μια ολοκληρωμένη κατανόηση της σημασίας της στην εργασία μας.

### 6.2 Κώδικας - Χαρακτηριστικά

Σε αυτήν την ενότητα θα εξηγήσουμε αναλυτικά τα χαρακτηριστικά της εφαρμογής καθώς και τον τρόπο υλοποίησής τους.

#### 6.2.1 Βιβλιοθήκες που χρησιμοποιήθηκαν

Τα θεμέλια κάθε εφαρμογής Python βρίσκονται στις βιβλιοθήκες και τις κλάσεις που χρησιμοποιεί. Αυτές οι βιβλιοθήκες ενσωματώνουν μια πληθώρα συναρτήσεων, μεθόδων και κλάσεων, σχεδιασμένες για να βελτιώσουν και να επιταχύνουν τη διαδικασία ανάπτυξης

εφαρμογών. Η εφαρμογή μας για την οπτικοποίηση εικόνων και μάσκας δεν αποτελεί εξαίρεση. Αξιοποιεί μια συλλογή βιβλιοθηκών Python για να χειριστεί εργασίες που κυμαίνονται από την απόδοση του γραφικού περιβάλλοντος χρήστη έως την επεξεργασία και τον χειρισμό δεδομένων εικόνας και μάσκας.

Ακολουθεί μια επεξήγηση των βιβλιοθηκών και των κλάσεων που αποτελούν τη ραχοκοκαλιά της εφαρμογής μας, προσφέροντας μια ματιά στους ρόλους και τη σημασία τους.

```
import numpy as np

import tkinter as tk
from tkinter import IntVar, ttk, \
    messagebox, StringVar, \
    DoubleVar, filedialog

from PIL import Image, ImageTk
```

Το Tkinter είναι η τυπική εργαλειοθήκη γραφικού περιβάλλοντος χρήστη GUI στην Python. Παρέχει μια ισχυρή αντικειμενοστραφή διεπαφή στο Tk GUI toolkit. Η εφαρμογή μας χρησιμοποιεί το Tkinter για την απόδοση της διεπαφής χρήστη, την καταγραφή των εισόδων του χρήστη και την εμφάνιση εικόνων και μασκών.

Το PIL βοηθάει στο άνοιγμα, την επεξεργασία και την αποθήκευση πολλών διαφορετικών μορφών αρχείων εικόνας. Σε αυτή την εφαρμογή, χρησιμοποιείται ιδιαίτερα για το άνοιγμα εικόνων και μασκών, την αποθήκευση μασκών και τη μετατροπή δεδομένων εικόνας σε μορφή που μπορεί εύκολα να αποδοθεί στον καμβά του Tkinter.

Το NumPy είναι μια βιβλιοθήκη η οποία προσθέτει υποστήριξη για μεγάλους πίνακες, μαζί με μια μεγάλη συλλογή μαθηματικών συναρτήσεων υψηλού επιπέδου για τη λειτουργία αυτών των πινάκων. Στην εφαρμογή μας, το NumPy χρησιμοποιείται για τον αποτελεσματικό χειρισμό πινάκων δεδομένων εικόνας και μάσκας.

### 6.2.2 Κλάση *AutoScrollbar*

Καθώς εμβαθύνουμε στην ανάπτυξη της εφαρμογής οπτικοποίησης και επεξεργασίας εικόνας και μάσκας, η ανάγκη για εξελιγμένα στοιχεία γραφικού περιβάλλοντος που βελτιώνουν την αλληλεπίδραση και την εμπειρία του χρήστη γίνεται ύψιστης σημασίας. Ένα τέτοιο κρίσιμο στοιχείο είναι η γραμμή κύλισης, ένα στοιχείο γραφικού περιβάλλοντος που διευκολύνει την πλοήγηση σε μεγάλες εικόνες και μάσκες, επιτρέποντας στους χρήστες να μετακινηθούν απρόσκοπτα μέσα στο περιεχόμενο. Ωστόσο, μια στατική γραμμή κύλισης μπορεί να είναι οπτικά ενοχλητική όταν δεν χρειάζεται. Αυτό μας φέρνει στην υλοποίηση της κλάσης *AutoScrollbar* - μια δυναμική γραμμή κύλισης που εμφανίζεται μόνο όταν το περιεχόμενο υπερβαίνει την περιοχή εμφάνισης.

```

class AutoScrollbar(ttk.Scrollbar):
    """ A scrollbar that hides itself if it's not needed. Works only for grid geometry manager """
    def set(self, lo, hi):
        if float(lo) <= 0.0 and float(hi) >= 1.0:
            self.grid_remove()
        else:
            self.grid()
            ttk.Scrollbar.set(self, lo, hi)

    def pack(self, **kw):
        raise tk.TclError('Cannot use pack with the widget ' + self.__class__.__name__)

    def place(self, **kw):
        raise tk.TclError('Cannot use place with the widget ' + self.__class__.__name__)

```

Η κλάση *AutoScrollbar* είναι μια βελτίωση της παραδοσιακής γραμμής κύλισης, σχεδιασμένη να είναι διακριτική. Αποκρύπτεται αυτόματα όταν δεν χρειάζεται και γίνεται ορατή και ενεργή όταν το περιεχόμενο ξεπερνά την περιοχή προβολής. Ας αναλύσουμε τη δομή και τις λειτουργίες της κλάσης *AutoScrollbar*.

Η μέθοδος *set* καλείται με δύο ορίσματα, *lo* και *hi*, τα οποία αντιπροσωπεύουν τις κλασματικές τιμές της θέσης της γραμμής κύλισης. Όταν ολόκληρο το περιεχόμενο είναι ορατό μέσα στο παράθυρο (δηλαδή,  $lo \leq 0$  και  $hi \geq 1$ ), η γραμμή κύλισης αφαιρείται από το παράθυρο, αποκρύπτοντάς την ουσιαστικά από το χρήστη. Εάν το περιεχόμενο υπερχειλίσει, η γραμμή κύλισης προστίθεται ξανά στο παράθυρο, καθιστώντας την ορατή και λειτουργική.

### 6.2.3 Κλάση CanvasImage

Τώρα που έχουμε δημιουργήσει μια λειτουργική και ευέλικτη γραμμή κύλισης με την κλάση *AutoScrollbar*, ήρθε η ώρα να επικεντρωθούμε στην εμφάνιση και τον χειρισμό εικόνων και μασκών μέσα στην εφαρμογή μας. Η κλάση *CanvasImage* εξυπηρετεί αυτόν ακριβώς τον σκοπό, προσφέροντας εκτεταμένες δυνατότητες, όπως μεγέθυνση, σάρωση και επιλογή εικόνας, καθιστώντας εύκολη την αλληλεπίδραση των χρηστών με τις εικόνες. Ας αναλύσουμε αυτή την κλάση, ας εξερευνήσουμε τις μεθόδους της και ας κατανοήσουμε τις λειτουργίες της.

```

class CanvasImage:
    """ Display and zoom image """
    def __init__(self, placeholder, path, parent):
        ...
        if path:
            self.load_img(path)

```

Η κλάση *CanvasImage* ξεκινά την αρχικοποίησή της με τη μέθοδο *\_\_init\_\_* με την οποία γίνεται η ρύθμιση διαφόρων χαρακτηριστικών και ρυθμίσεων που είναι απαραίτητες για την εμφάνιση μιας εικόνας και την ενεργοποίηση των αλληλεπιδράσεων του χρήστη. Ενσωματώνει τη γραμμή *AutoScrollbar* για ομαλή πλοήγηση, δεσμεύει πολυάριθμα συμβάντα για αλληλεπιδράσεις όπως μεγέθυνση, σάρωση και επιλογή και χειρίζεται μεγάλες εικόνες. Στη συνέχεια, χρησιμοποιείται η μέθοδος *load\_img* για να φορτώσει μια εικόνα και να επαναφέρει το γραφικό περιβάλλον χρήστη στην προεπιλεγμένη του κατάσταση, προετοιμάζοντάς το για την εμφάνιση της νέας εικόνας.

```

# Bind events to the Canvas
self.canvas.bind('<Configure>', lambda event: self.__show_image()) # canvas is resized
self.canvas.bind('<ButtonPress-1>', self.__move_from) # remember canvas position
self.canvas.bind('<ButtonPress-3>', self.__select_start) # remember canvas position
self.canvas.bind('<B3-Motion>', self.__select_update) # remember canvas position
self.canvas.bind('<B1-Motion>', self.__move_to) # move canvas to the new position
self.canvas.bind('<MouseWheel>', self.__wheel) # zoom for Windows and MacOS, but not Linux
self.canvas.bind('<Button-5>', self.__wheel) # zoom for Linux, wheel scroll down
self.canvas.bind('<Button-4>', self.__wheel) # zoom for Linux, wheel scroll up

```

Διάφορες μέθοδοι και δεσμεύσεις γεγονότων επιτρέπουν στους χρήστες να αλληλεπιδρούν με την εικόνα. Η κλάση διαχειρίζεται την κίνηση του ποντικιού, τα κλικ και τα γεγονότα του τροχού του ποντικιού για να διευκολύνει ενέργειες όπως η επιλογή, η σάρωση και το ζουμ.

Η μέθοδος `__wheel` χειρίζεται συγκεκριμένα τις ενέργειες ζουμ που ξεκινούν από τον τροχό του ποντικιού, προσαρμόζοντας κατάλληλα την κλίμακα της εικόνας. Ελέγχει τη θέση του ποντικιού για να διασφαλίσει ότι η μεγέθυνση γίνεται μόνο όταν το ποντίκι βρίσκεται πάνω από την εικόνα.

```

def __select_update(self, event):
    self.update_mouse_img_coords(event)

    self.selected_x1_var.set(self.img_mouse_x)
    self.selected_y1_var.set(self.img_mouse_y)
    self.redraw()

def __select_start(self, event):
    self.update_mouse_img_coords(event)

    self.selected_x0_var.set(self.img_mouse_x)
    self.selected_x1_var.set(self.img_mouse_x)

    self.selected_y0_var.set(self.img_mouse_y)
    self.selected_y1_var.set(self.img_mouse_y)

    self.parent.selected_value_var.set(None)

def motion(self, event):
    self.update_mouse_img_coords(event)
    # print('{}', '{}'.format(x, y))

```

Η κλάση παρέχει δυνατότητες για την επιλογή ενός τμήματος της εικόνας και την παρακολούθηση των συντεταγμένων, όπως φαίνεται σε μεθόδους όπως `__select_start`, `__select_update` και `motion`. Η μέθοδος `motion`, για παράδειγμα, ενημερώνει τις συντεταγμένες της εικόνας καθώς κινείται το ποντίκι, επιτρέποντας την παρακολούθηση της θέσης του ποντικιού στην εικόνα σε πραγματικό χρόνο.

Η μέθοδος `add_text` επιτρέπει την προσθήκη επισημάνσεων κειμένου στην εικόνα. Αυτό μπορεί να χρησιμοποιηθεί για την εμφάνιση πληροφοριών ή σχολίων απευθείας στην εικόνα, στην περίπτωση που οι τιμές των μασκών σε κάθε εικονοστοιχείο.

```

def smaller(self):
    """ Resize image proportionally and return smaller image """
    w1, h1 = float(self.imwidth), float(self.imheight)
    w2, h2 = float(self.__huge_size), float(self.__huge_size)
    aspect_ratio1 = w1 / h1
    aspect_ratio2 = w2 / h2 # it equals to 1.0
    if aspect_ratio1 == aspect_ratio2:
        image = Image.new('RGB', (int(w2), int(h2)))
        k = h2 / h1 # compression ratio
        w = int(w2) # band length
    elif aspect_ratio1 > aspect_ratio2:
        image = Image.new('RGB', (int(w2), int(w2 / aspect_ratio1)))
        k = h2 / w1 # compression ratio
        w = int(w2) # band length
    else: # aspect_ratio1 < aspect_ratio2
        image = Image.new('RGB', (int(h2 * aspect_ratio1), int(h2)))
        k = h2 / h1 # compression ratio
        w = int(h2 * aspect_ratio1) # band length
    i, j, n = 0, 1, round(0.5 + self.imheight / self.__band_width)
    while i < self.imheight:
        print('\rOpening image: {j} from {n}'.format(j=j, n=n), end='')
        band = min(self.__band_width, self.imheight - i) # width of the tile band
        self.__tile[1][3] = band # set band width
        self.__tile[2] = self.__offset + self.imwidth * i * 3 # tile offset (3 bytes per pixel)
        self.__image.close()
        self.__image = Image.open(self.path) # reopen / reset image
        self.__image.size = (self.imwidth, band) # set size of the tile band
        self.__image.tile = [self.__tile] # set tile
        cropped = self.__image.crop((0, 0, self.imwidth, band)) # crop tile band
        image.paste(cropped.resize((w, int(band * k)+1), self.__filter), (0, int(i * k)))
        i += band
        j += 1
    print('\r' + 30*' ' + '\r', end='') # hide printed string
    return image

```

Για τεράστιες εικόνες, η κλάση *CanvasImage* χρησιμοποιεί μια δομή πυραμίδας για να επιτρέπει την αποτελεσματική μεγέθυνση και σάρωση χωρίς να φορτώνει ολόκληρη την εικόνα στη μνήμη. Η μέθοδος *smaller* δημιουργεί μια μικρότερη έκδοση της εικόνας για το σκοπό αυτό.

Η μέθοδος *\_\_show\_image* είναι υπεύθυνη για την επανασχεδίαση της εικόνας στον καμβά, ειδικά μετά από αλληλεπιδράσεις όπως η μεγέθυνση και η σάρωση. Εξασφαλίζει ότι το τμήμα της εικόνας που εμφανίζεται αντιστοιχεί στην τρέχουσα κατάσταση του καμβά και στις αλληλεπιδράσεις του χρήστη.

## 6.2.4 Η κλάση MainWindow

Η κλάση *MainWindow* κληρονομεί από την *tkinter.ttk.Frame* και χρησιμεύει ως η βασική διεπαφή με την οποία αλληλεπιδρούν οι χρήστες. Είναι το σημείο όπου ενσωματώνονται οι λειτουργίες φόρτωσης εικόνων, εμφάνισης και χειρισμού τιμών, προσφέροντας μια ολοκληρωμένη εμπειρία χρήστη.

```

class MainWindow(tk.Frame):
    """ Main window class """
    def __init__(self, mainframe, path): ...

    def load_file(self):
        pass

    def save_func(self):
        file = self.file_path.get()
        # txt_file_path = os.path.splitext(file)[0] + ".txt"
        txt_file_path = os.path.splitext(file)[0] + "_mask.tiff"

        # np.savetxt(txt_file_path, self.values.transpose())
        mask = Image.fromarray(self.values.transpose())
        mask.save(txt_file_path)
        messagebox.showinfo("Saved", "Matrix is saved to " + txt_file_path)

    def __update_values(self):
        canvas = self.canvas
        try:
            val = self.selected_value_var.get()
        except Exception as ex:
            messagebox.showinfo("Error", "Value should be float")
            return

        x0 = min([canvas.selected_x0_var.get(), canvas.selected_x1_var.get()])
        x1 = max([canvas.selected_x0_var.get(), canvas.selected_x1_var.get()])
        y0 = min([canvas.selected_y0_var.get(), canvas.selected_y1_var.get()])
        y1 = max([canvas.selected_y0_var.get(), canvas.selected_y1_var.get()])

        for x in range(x0-1, x1):
            for y in range(y0-1, y1):
                try:
                    self.values[x, y] = val
                except Exception as ex:
                    pass
        canvas.redraw()

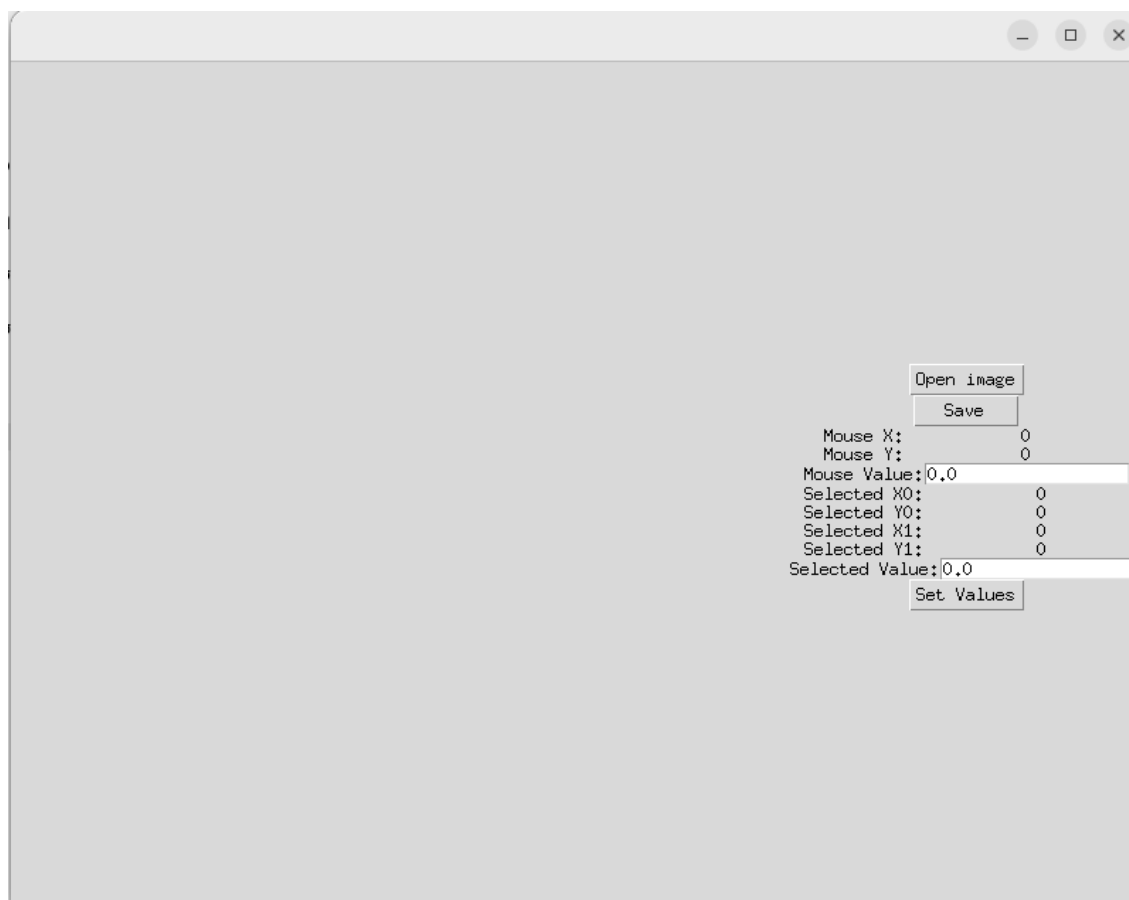
```

Η κλάση αρχικοποιείται με ένα filepath προς την εικόνα. Ρυθμίζει το μέγεθος του κύριου παραθύρου και ενσωματώνει την κλάση *CanvasImage* για το χειρισμό της απεικόνισης και των αλληλεπιδράσεων της εικόνας. Περιλαμβάνει ένα παράθυρο διαλόγου αρχείων για τη φόρτωση εικόνων και μια λειτουργία αποθήκευσης που μετατρέπει τις τιμές της επιλεγμένης περιοχής σε μάσκα και την αποθηκεύει ως αρχείο TIFF. Εμφανίζει δυναμικά τη θέση του ποντικιού και τις συντεταγμένες και τις τιμές της επιλεγμένης περιοχής, ενισχύοντας την αλληλεπίδραση του χρήστη. Μια κρίσιμη συνάρτηση, η *\_\_update\_values*, επιτρέπει στους χρήστες να ορίσουν τιμές για μια επιλεγμένη περιοχή στην εικόνα, οι οποίες στη συνέχεια μπορούν να αποθηκευτούν ως μάσκα.

### 6.3 Παράδειγμα Χρήσης

Σε αυτήν την ενότητα θα δούμε συνοπτικά ένα παράδειγμα χρήσης της εφαρμογής που περιγράψαμε προηγουμένως.

Κάνοντας εκκίνηση την εφαρμογή εμφανίζεται ένα κενό παράθυρο όπως φαίνεται στην Εικόνα 6.1 με τα κουμπιά που ενσωματώνουν τις λειτουργίες του ανοίγματος, αποθήκευσης και επεξεργασίας εικόνας και μάσκας.



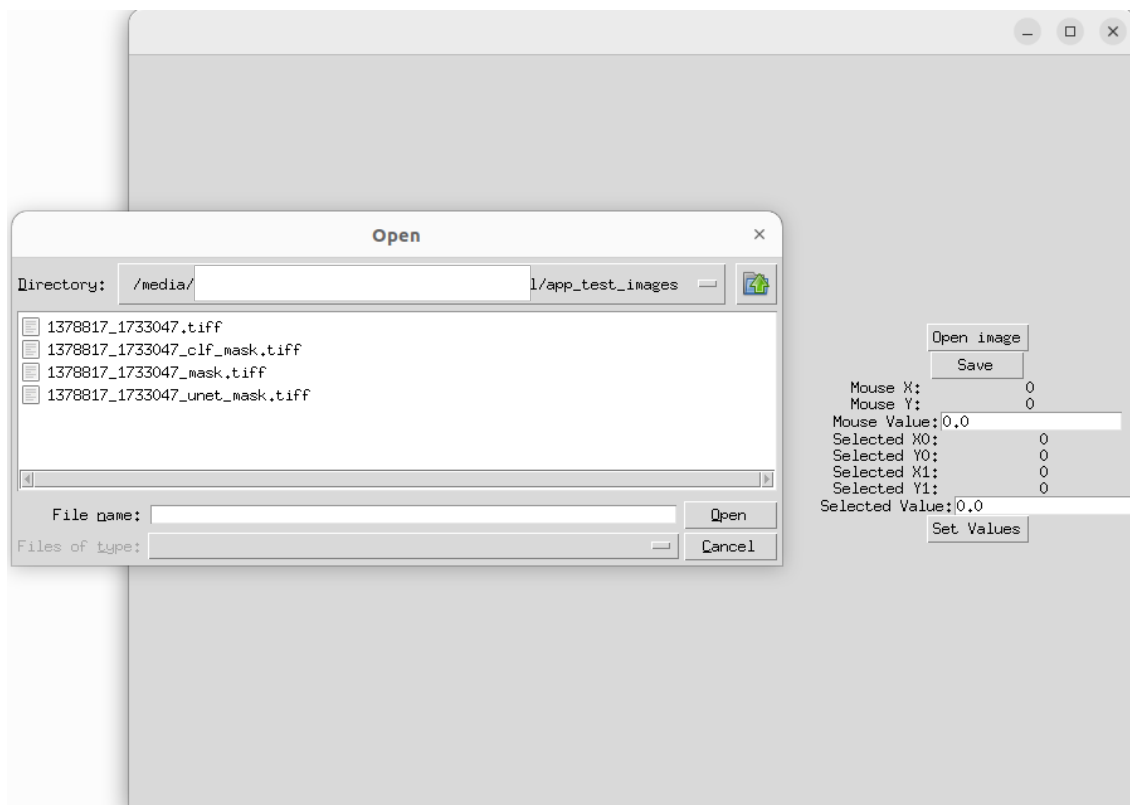
Σχήμα 6.1: Παράθυρο εκκίνησης

Πατώντας στο κουμπί *Open Image* ανοίγει ένα δεύτερο παράθυρο πλοήγησης αρχείων για να δώσουμε την εικόνα εισόδου, όπως φαίνεται στην Εικόνα 6.2

Διαλέγοντας την εικόνα που θέλουμε να φορτώσουμε (στο παράδειγμα αυτό την εικόνα 1378817\_1733047.tiff) φορτώνει και την αντίστοιχη μάσκα, όπως βλέπουμε και στην Εικόνα 6.3 η οποία πρέπει να έχει ίδιο όνομα αλλά με την κατάληξη "\_mask.tiff". Η αρχική εικόνα όπως βλέπουμε έχει φορτωθεί και είναι zoomed out, επομένως δεν υπάρχει και μπάρα κύλισης.

Χρησιμοποιώντας τον τροχό του ποντικιού μπορούμε να κάνουμε ζουμ στην εικόνα (βλ. Εικόνα 6.4). Τότε γίνεται και ορατή και η μπάρα κύλισης. Μπορούμε επίσης να κάνουμε σάρωση της εικόνας κρατώντας πατημένο το αριστερό κλικ και μετακινώντας το ποντίκι. Ο τρόπος σάρωσης, μεγέθυνσης και σμίκρυνσης της εικόνας μοιάζει σε πολύ μεγάλο βαθμό με τον τρόπο περιήγησης σε πολλά γνωστά προγράμματα προβολής εικόνων και του Google Maps.

Κάνοντας ζουμ, αρχικά μπορούμε να διαχωρίσουμε τα εικονοστοιχεία πολύ εύκολα (Εικόνα 6.5), και κάνοντας περεταιίρω ζουμ αρχίζουμε να μπορούμε να διακρίνουμε τις τιμές που έχει σε κάθε εικονοστοιχείο η μάσκα (Εικόνα 6.6).

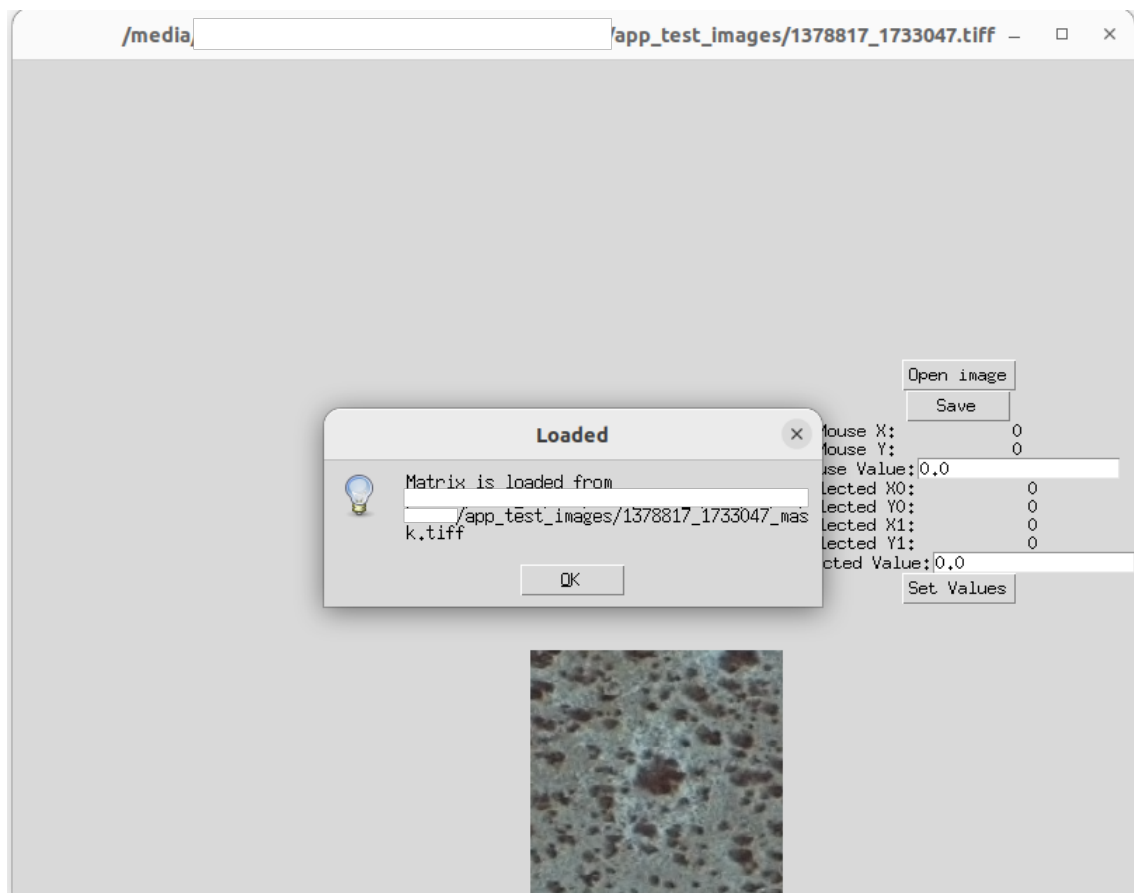


Σχήμα 6.2: Παράθυρο εισαγωγής εικόνας

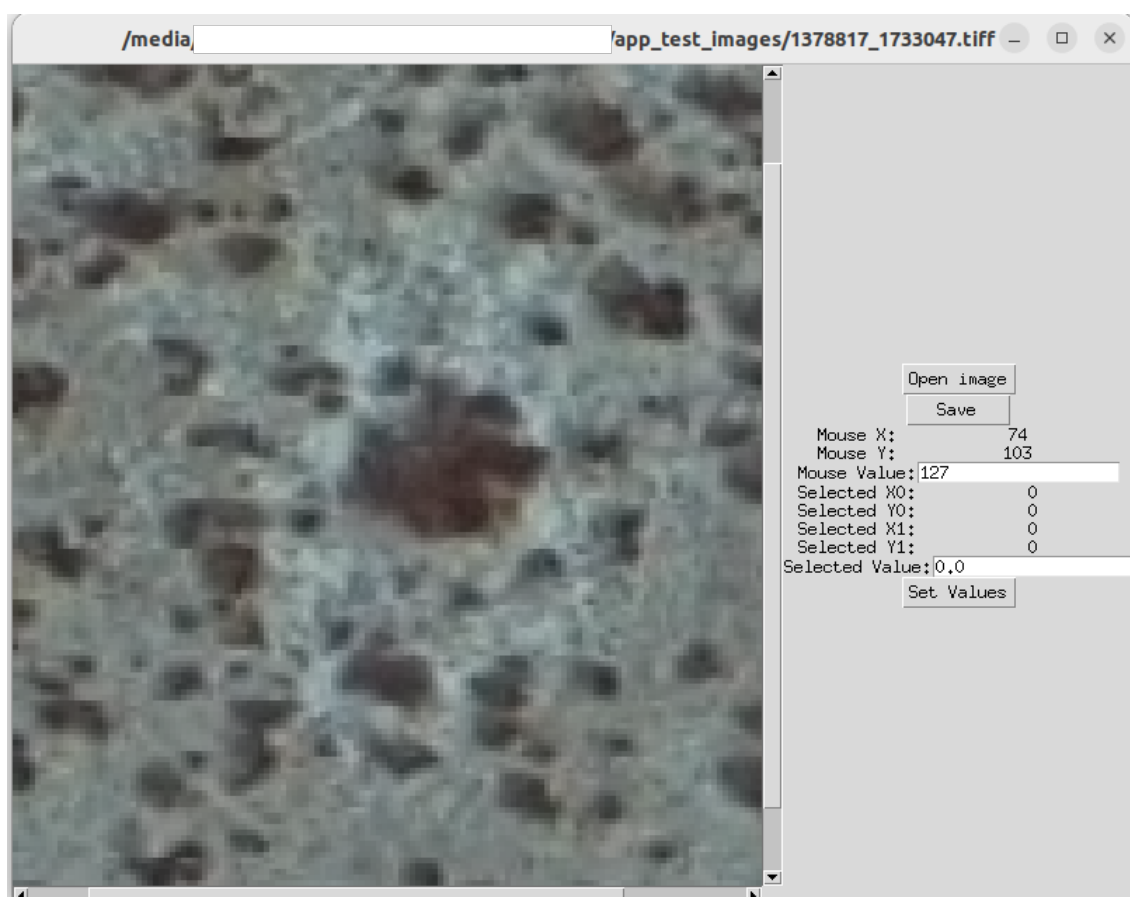
Μπορούμε να δούμε τη τρέχουσα τιμή της μάσκας ενός εικονοστοιχείου μετακινώντας το ποντίκι πάνω σε αυτό. Επιπλέον μπορούμε να διαλέξουμε ένα (Εικόνα 6.7) ή περισσότερα εικονοστοιχεία (Εικόνα 6.8) με δεξί κλικ ή κρατώντας το δεξί κλικ και μετακινώντας το ποντίκι (για πολλαπλή επιλογή). Στα δεξιά φαίνονται οι συντεταγμένες των επιλεγμένων εικονοστοιχείων. Συμπληρώνοντας το πεδίο *Selected Value* και πατώντας το κουμπί *Set Values* από κάτω μπορούμε να αλλάξουμε τις τιμές της μάσκας και βλέπουμε να ενημερώνεται απευθείας και το παράθυρο της εφαρμογής με τις νέες τιμές.

Τέλος, πατώντας το κουμπί *Save* μπορούμε να αποθηκεύσουμε την μάσκα με ό,τι αλλαγές έχουν γίνει σε αυτή (Εικόνα 6.9).

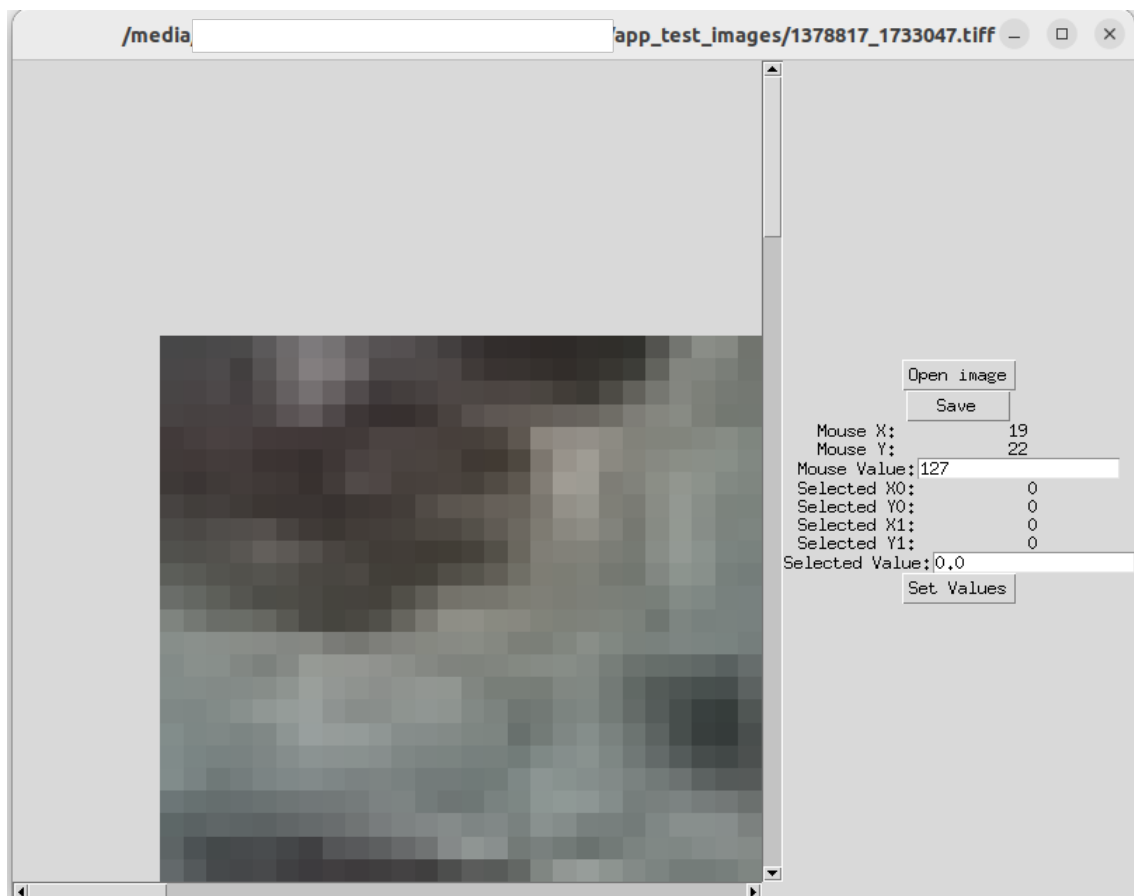




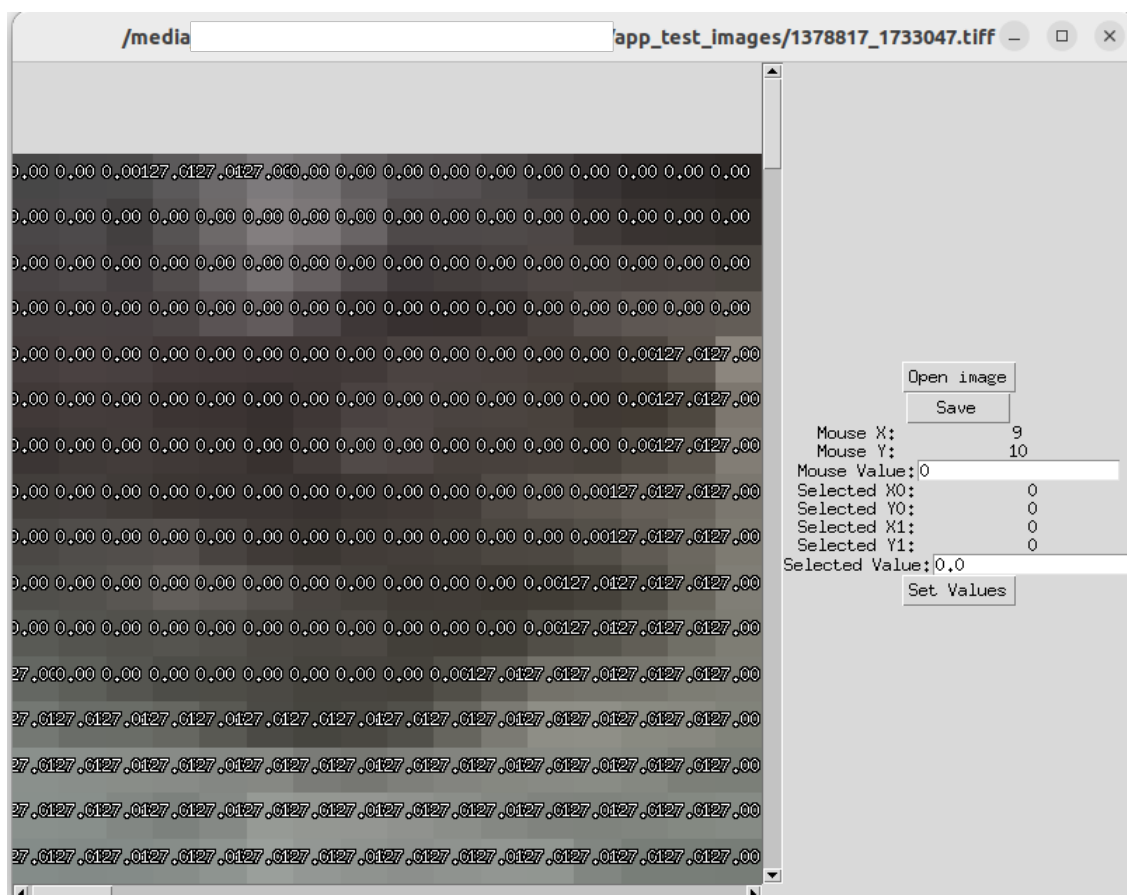
Σχήμα 6.3: Παράθυρο ανάγνωσης εικόνας και μάσκας



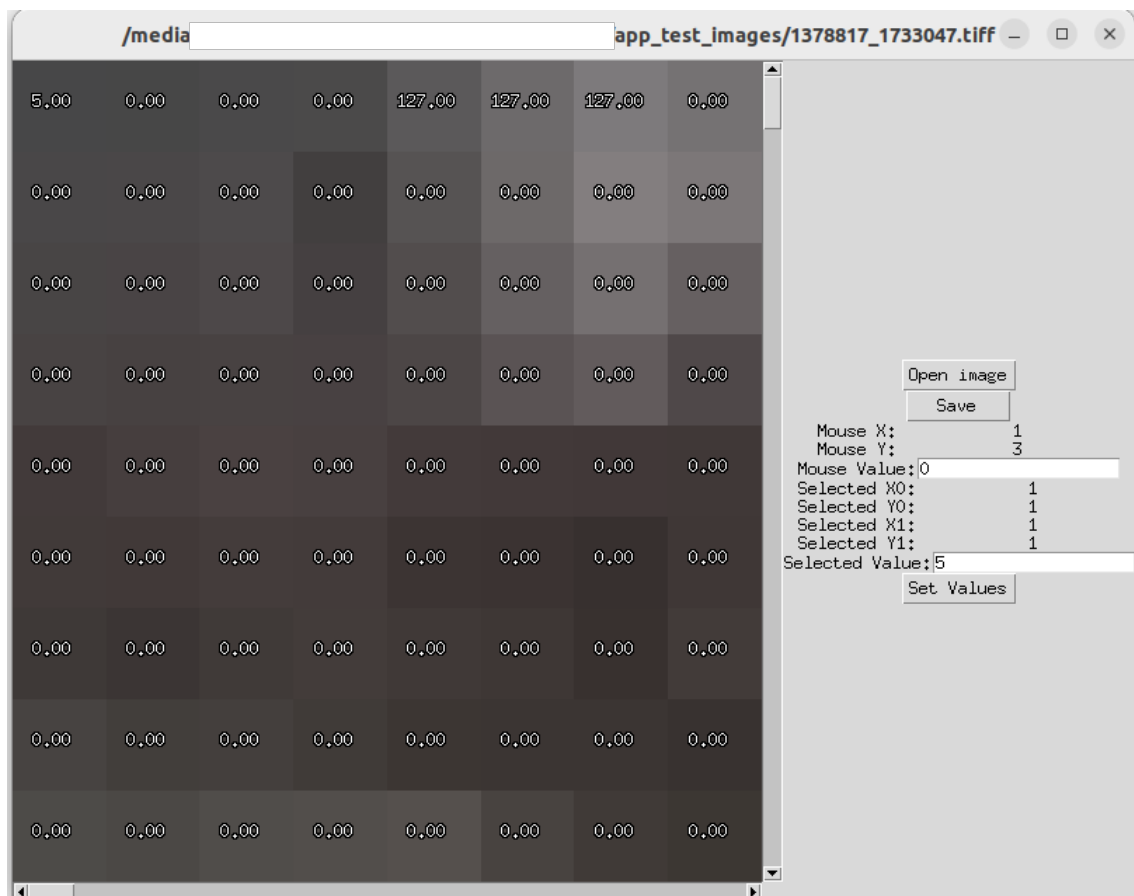
Σχήμα 6.4: Παράθυρο περιήγησης εικόνας



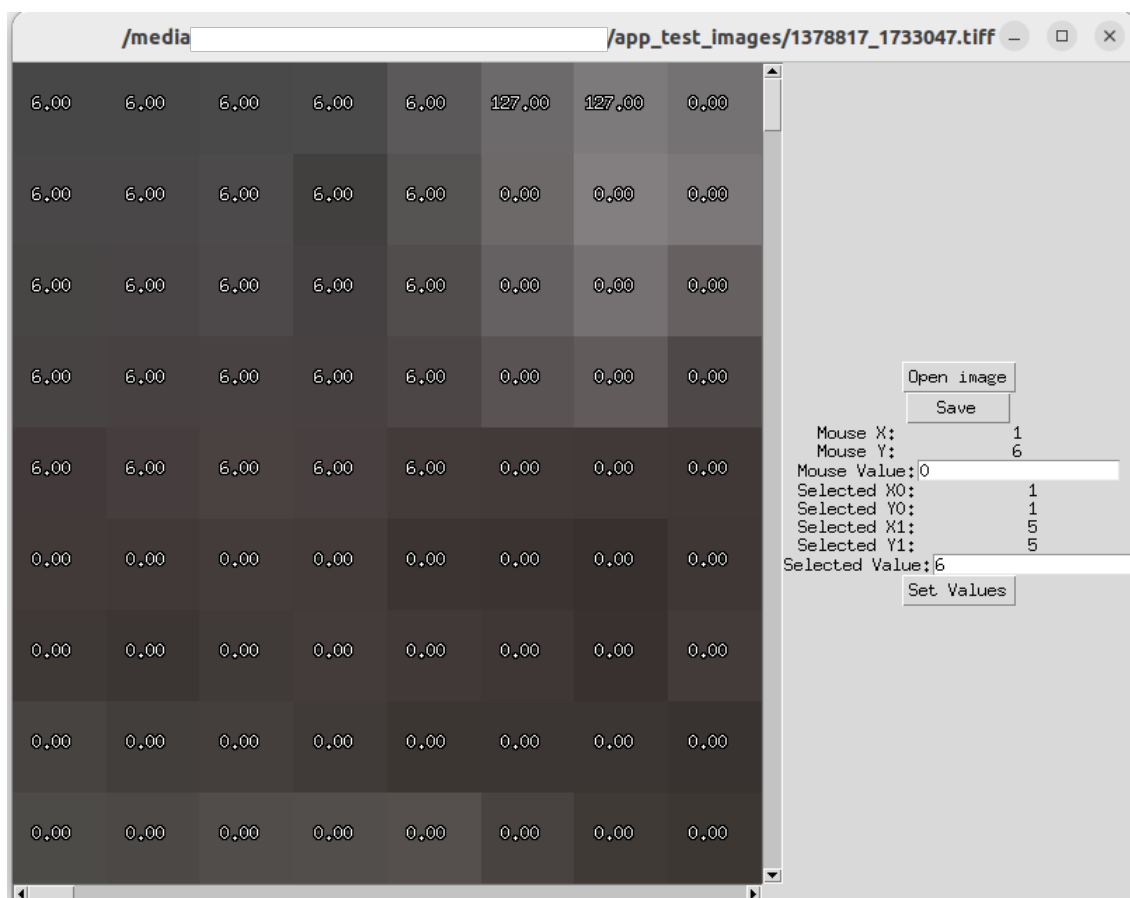
Σχήμα 6.5: Ζουμ σε μέρος της εικόνας



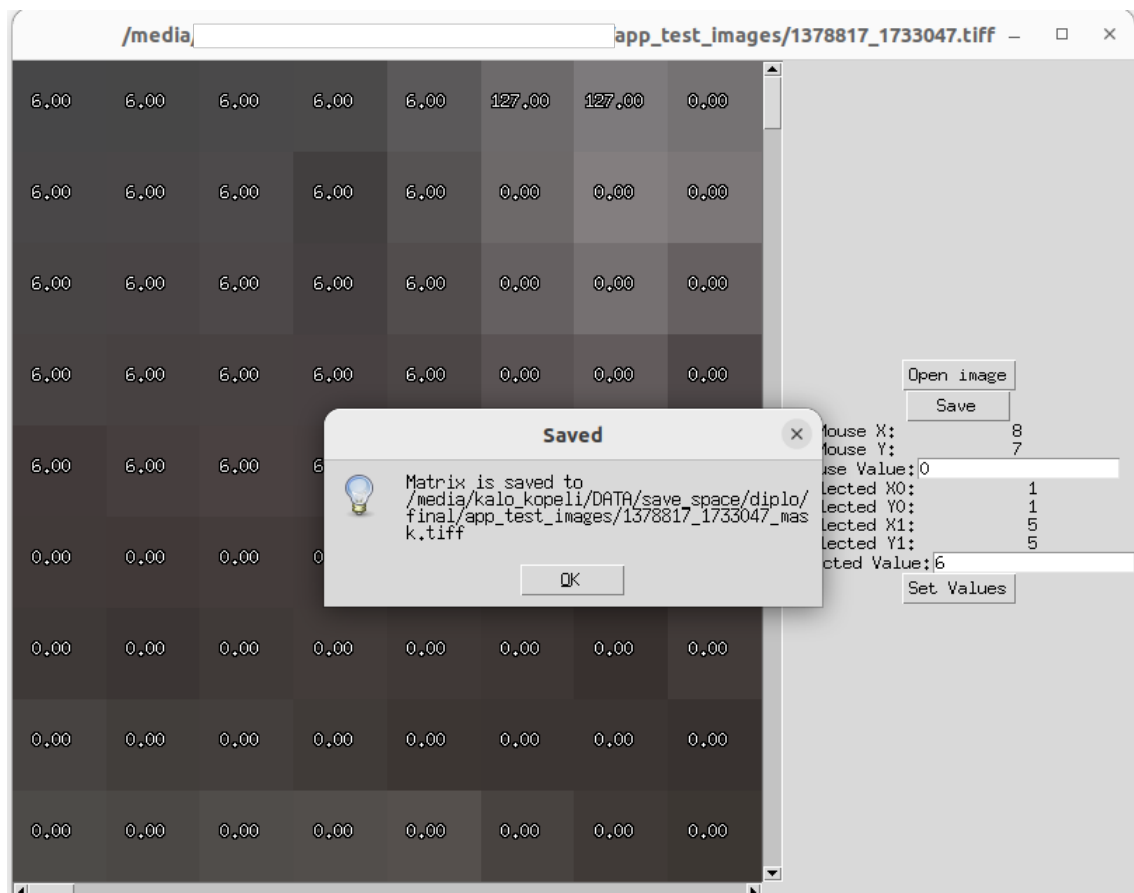
Σχήμα 6.6: Η μεγενθυμένη εικόνα και οι τιμές της μάσκας σε κάθε εικονοστοιχείο



Σχήμα 6.7: Επιλογή και αλλαγή τιμή μάσκας σε ένα εικονοστοιχείο



Σχήμα 6.8: Επιλογή και αλλαγή τιμή μάσκας σε πολλά εικονοστοιχεία



Σχήμα 6.9: Αποθήκευση επεξεργασμένης μάσκας





**Μέρος **

**Επίλογος**

---



## Κεφάλαιο **7**

# Επίλογος

---

### 7.1 Συμπεράσματα

Η παρούσα εργασία περιέγραψε μια περίπλοκη και καινοτόμο εφαρμογή μοντέλων μηχανικής μάθησης για την ταξινόμηση τύπων βλάστησης σε αεροφωτογραφίες, τη δημιουργία πιο εξελιγμένων масκών και τελικά την εκμετάλλευση αυτής της γνώσης για την πρόβλεψη ποσοστών βλάστησης από πιο εύκολα διαθέσιμες δορυφορικές εικόνες. Το ταξίδι ξεκίνησε με τη δημιουργία ενός συνόλου δεδομένων από αεροφωτογραφίες σχολιασμένες με περίπου 15,000 σημεία που προσδιορίζουν ξυλώδεις, μη-ξυλώδεις και χωρίς βλάστηση περιοχές. Αυτό το σύνολο δεδομένων αποτέλεσε τον ακρογωνιαίο λίθο της διαδικασίας εκπαίδευσης των αρχικών μοντέλων.

Διεξήχθη μία διαδικασία αξιολόγησης διαφόρων μοντέλων μηχανικής μάθησης για να εντοπίσουμε το μοντέλο με τις καλύτερες επιδόσεις όσον αφορά την ακρίβεια ταξινόμησης και άλλες σχετικές μετρήσεις. Το επιλεγμένο μοντέλο χρησιμοποιήθηκε για τη δημιουργία масκών μεγάλων εκτάσεων γης, μια διαδικασία που θα ήταν απαγορευτικά χρονοβόρα και ανακριβής εάν εκτελούνταν χειροκίνητα.

Βασίζόμενοι στο μοντέλο ταξινόμησης, υιοθετήσαμε μια αρχιτεκτονική U-Net, η οποία έχει σχεδιαστεί για εργασίες τμηματοποίησης εικόνων. Εκπαιδευμένο στις μάσκες που παρήγαγε το μοντέλο ταξινόμησης, το μοντέλο U-Net μας παρείχε βελτιωμένες μάσκες που αναπαριστούσαν με μεγαλύτερη ακρίβεια τα όρια μεταξύ ξυλώδους βλάστησης και περιοχών χωρίς ξυλώδη βλάστηση.

Αυτές οι μάσκες που δημιουργήθηκαν από το U-Net αλλά και από το προηγούμενο μοντέλο ταξινόμησης χρησίμευσαν ως σημείο αναφοράς για την εξαγωγή των ποσοστών ξυλώδους και μη ξυλώδους βλάστησης σε περιοχές όπου οι δορυφορικές εικόνες ήταν διαθέσιμες. Αυτό το βήμα παρείχε κρίσιμες πληροφορίες, οι οποίες στη συνέχεια χρησιμοποιήθηκαν ως δεδομένα εκπαίδευσης για ένα νέο μοντέλο. Στόχος αυτού του νέου μοντέλου ήταν να προβλέψει τα ποσοστά βλάστησης απευθείας από δορυφορικές εικόνες, διευρύνοντας το πεδίο εφαρμογής της μεθοδολογίας μας, δεδομένης της ευρείας διαθεσιμότητας τέτοιων εικόνων.

Επιπλέον, αναπτύχθηκε μια πρωτότυπη εφαρμογή, η οποία επιτρέπει στους χρήστες να ανοίγουν οποιαδήποτε μάσκα αεροφωτογραφίας και να τροποποιούν τις επισημάνσεις της. Αυτό το διαδραστικό εργαλείο δίνει τη δυνατότητα στους χρήστες να τελειοποιήσουν τα δεδομένα, βελτιώνοντας τη συνολική ακρίβεια των μοντέλων και επιτρέποντας ενδεχομένως τη διόρθωση τυχόν αρχικών σφαλμάτων σχολιασμού.

Συνολικά, η έρευνά μας συμβάλλει στην τηλεπισκόπηση και τις περιβαλλοντικές μελέτες. Με τη χρήση αλγορίθμων μηχανικής μάθησης και προηγμένων τεχνικών τμηματοποίησης εικόνων, δείξαμε μια πορεία για την αξιολόγηση και πρόβλεψη της κατανομής της βλάστησης από ευρέως διαθέσιμες δορυφορικές εικόνες.

## 7.2 Μελλοντικές Εργασίες

Υπάρχουν διάφορες κατευθύνσεις που θα μπορούσε να ακολουθήσει η μελλοντική έρευνα για να αξιοποιήσει το έργο που παρουσιάζεται στην παρούσα εργασία. Πρώτον, η εφαρμογή πιο προηγμένων τεχνικών βαθιάς μάθησης, θα μπορούσαν να βελτιώσουν την ακρίβεια ταξινόμησης και τμηματοποίησης.

Δεύτερον, η πρωτότυπη εφαρμογή θα μπορούσε να επεκταθεί ώστε να περιλαμβάνει λειτουργίες για τον σχολιασμό νέων εικόνων, δημιουργώντας έτσι περισσότερα δεδομένα εκπαίδευσης και βελτιώνοντας την απόδοση του μοντέλου με την πάροδο του χρόνου. Η εφαρμογή αυτή θα μπορούσε επίσης να επεκταθεί έτσι ώστε να μπορεί να γίνεται και εκπαίδευση των μοντέλων επί τόπου με τα νέα δεδομένα μέσω ενός γραφικού περιβάλλοντος.

Τέλος, η ενσωμάτωση των μοντέλων σε ένα λειτουργικό σύστημα πραγματικού χρόνου θα μπορούσε να παρέχει σχεδόν στιγμιαία ανάλυση της βλάστησης, ένα εργαλείο με τεράστιες δυνατότητες για τους περιβαλλοντολόγους, τους σχεδιαστές χρήσεων γης και τους ερευνητές της κλιματικής αλλαγής.

Εν κατακλείδι, η μελέτη αυτή σηματοδοτεί ένα ουσιαστικό βήμα για την αξιοποίηση της τεχνητής νοημοσύνης στον τομέα της τηλεπισκόπησης και της περιβαλλοντικής επιστήμης. Τα αποτελέσματα υπογραμμίζουν τις δυνατότητες για περαιτέρω πρόοδο και βελτιώσεις σε αυτό το διεπιστημονικό ερευνητικό πεδίο.

## Βιβλιογραφία

---

- [1] Olaf Ronneberger, Philipp Fischer και Thomas Brox. *U-net: Convolutional networks for biomedical image segmentation. Medical Image Computing and Computer-Assisted Intervention-MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III* 18, σελίδες 234-241. Springer, 2015.
- [2] Guido Van Rossum και Fred L. Drake. *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA, 2009.
- [3] Gareth James, Daniela Witten, Trevor Hastie και Robert Tibshirani. *An introduction to statistical learning*, τόμος 112. Springer, 2013.
- [4] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.
- [5] Ethem Alpaydin. *Introduction to machine learning*. MIT press, 2020.
- [6] Shahzad Khan. *Ethem Alpaydin. Introduction to Machine Learning (Adaptive Computation and Machine Learning Series)*. The MIT Press, 2004. ISBN: 0 262 01211 1. *Natural Language Engineering*, 14(1):133-137, 2008.
- [7] Horace B Barlow. *Unsupervised learning. Neural computation*, 1(3):295-311, 1989.
- [8] T. Hastie, R. Tibshirani και J.H. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer series in statistics. Springer, 2009.
- [9] Oded Maimon και Lior Rokach. *Data mining and knowledge discovery handbook*. 2005.
- [10] Krzysztof J Cios, Roman W Swiniarski, Witold Pedrycz, Lukasz A Kurgan, Krzysztof J Cios, Roman W Swiniarski, Witold Pedrycz και Lukasz A Kurgan. *Unsupervised learning: association rules. Data Mining: A Knowledge Discovery Approach*, σελίδες 289-306, 2007.
- [11] Pierre Baldi. *Autoencoders, unsupervised learning, and deep architectures. Proceedings of ICML workshop on unsupervised and transfer learning*, σελίδες 37-49. JMLR Workshop and Conference Proceedings, 2012.
- [12] Marco A Wiering και Martijn Van Otterlo. *Reinforcement learning. Adaptation, learning, and optimization*, 12(3):729, 2012.
- [13] Leslie Pack Kaelbling, Michael L Littman και Andrew W Moore. *Reinforcement learning: A survey. Journal of artificial intelligence research*, 4:237-285, 1996.

- [14] Yan Yan Song και LU Ying. *Decision tree methods: applications for classification and prediction*. *Shanghai archives of psychiatry*, 27(2):130, 2015.
- [15] Leo Breiman. *Random forests leo breiman and adele cutler*. *Random Forests-Classification Description*, 106, 2015.
- [16] Yoav Freund, Robert E Schapire και others. *Experiments with a new boosting algorithm*. *icml*, τόμος 96, σελίδες 148–156. Citeseer, 1996.
- [17] Robert E Schapire. *The boosting approach to machine learning: An overview*. *Nonlinear estimation and classification*, σελίδες 149–171, 2003.
- [18] Trevor Hastie, Saharon Rosset, Ji Zhu και Hui Zou. *Multi-class adaboost*. *Statistics and its Interface*, 2(3):349–360, 2009.
- [19] Philip M Long και Rocco A Servedio. *Random classification noise defeats all convex potential boosters*. *Proceedings of the 25th international conference on Machine learning*, σελίδες 608–615, 2008.
- [20] Manuel Fernández-Delgado, Eva Cernadas, Senén Barro και Dinani Amorim. *Do we need hundreds of classifiers to solve real world classification problems?* *The journal of machine learning research*, 15(1):3133–3181, 2014.
- [21] Jerome H Friedman. *Greedy function approximation: a gradient boosting machine*. *Annals of statistics*, σελίδες 1189–1232, 2001.
- [22] Tianqi Chen και Carlos Guestrin. *Xgboost: A scalable tree boosting system*. *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, σελίδες 785–794, 2016.
- [23] Thomas Cover και Peter Hart. *Nearest neighbor pattern classification*. *IEEE transactions on information theory*, 13(1):21–27, 1967.
- [24] David W Aha, Dennis Kibler και Marc K Albert. *Instance-based learning algorithms*. *Machine learning*, 6:37–66, 1991.
- [25] Naomi S Altman. *An introduction to kernel and nearest-neighbor nonparametric regression*. *The American Statistician*, 46(3):175–185, 1992.
- [26] Elena Deza, Michel Marie Deza, Michel Marie Deza και Elena Deza. *Encyclopedia of distances*. Springer, 2009.
- [27] Trevor Hastie και Robert Tibshirani. *Discriminant adaptive nearest neighbor classification and regression*. *Advances in neural information processing systems*, 8, 1995.
- [28] Ron Kohavi και others. *A study of cross-validation and bootstrap for accuracy estimation and model selection*. *Ijcai*, τόμος 14, σελίδες 1137–1145. Montreal, Canada, 1995.

- [29] Kevin Beyer, Jonathan Goldstein, Raghu Ramakrishnan και Uri Shaft. *When is “nearest neighbor” meaningful? Database Theory–ICDT’99: 7th International Conference Jerusalem, Israel, January 10–12, 1999 Proceedings 7*, σελίδες 217–235. Springer, 1999.
- [30] Marina Sokolova και Guy Lapalme. *A systematic analysis of performance measures for classification tasks. Information processing & management*, 45(4):427–437, 2009.
- [31] Tom Fawcett. *An introduction to ROC analysis. Pattern recognition letters*, 27(8):861–874, 2006.
- [32] Norman R Draper και Harry Smith. *Applied regression analysis*, τόμος 326. John Wiley & Sons, 1998.
- [33] Ian Goodfellow, Yoshua Bengio και Aaron Courville. *Deep learning*. MIT press, 2016.
- [34] Xavier Glorot και Yoshua Bengio. *Understanding the difficulty of training deep feedforward neural networks. Proceedings of the thirteenth international conference on artificial intelligence and statistics*, σελίδες 249–256. JMLR Workshop and Conference Proceedings, 2010.
- [35] Yann LeCun, Yoshua Bengio και Geoffrey Hinton. *Deep learning. nature*, 521(7553):436–444, 2015.
- [36] Sergey Ioffe και Christian Szegedy. *Batch normalization: Accelerating deep network training by reducing internal covariate shift. International conference on machine learning*, σελίδες 448–456. pmlr, 2015.
- [37] Sepp Hochreiter και Jürgen Schmidhuber. *Long short-term memory. Neural computation*, 9(8):1735–1780, 1997.
- [38] Vinod Nair και Geoffrey E Hinton. *Rectified linear units improve restricted boltzmann machines. Proceedings of the 27th international conference on machine learning (ICML-10)*, σελίδες 807–814, 2010.
- [39] Yann A. LeCun, Léon Bottou, Genevieve B. Orr και Klaus Robert Müller. *Efficient BackProp*, σελίδες 9–48. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
- [40] David E Rumelhart, Geoffrey E Hinton και Ronald J Williams. *Learning representations by back-propagating errors. nature*, 323(6088):533–536, 1986.
- [41] Michael A Nielsen. *Neural networks and deep learning*, τόμος 25. Determination press San Francisco, CA, USA, 2015.
- [42] Robert Hecht-Nielsen. *Theory of the backpropagation neural network. Neural networks for perception*, σελίδες 65–93. Elsevier, 1992.
- [43] Paul J Werbos. *Backpropagation through time: what it does and how to do it. Proceedings of the IEEE*, 78(10):1550–1560, 1990.

- [44] Sebastian Ruder. *An overview of gradient descent optimization algorithms*. arXiv preprint arXiv:1609.04747, 2016.
- [45] Frank Rosenblatt. *The perceptron: a probabilistic model for information storage and organization in the brain*. *Psychological review*, 65(6):386, 1958.
- [46] Xavier Glorot, Antoine Bordes και Yoshua Bengio. *Deep sparse rectifier neural networks*. *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, σελίδες 315–323. JMLR Workshop and Conference Proceedings, 2011.
- [47] Thomas P Higginbottom, Elias Symeonakis, Hanna Meyer και Sebastianvan der Linden. *Mapping fractional woody cover in semi-arid savannahs using multi-seasonal composites from Landsat data*. *ISPRS Journal of Photogrammetry and Remote Sensing*, 139:88–102, 2018.
- [48] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg και others. *Scikit-learn: Machine learning in Python*. *the Journal of machine Learning research*, 12:2825–2830, 2011.
- [49] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga και others. *Pytorch: An imperative style, high-performance deep learning library*. *Advances in neural information processing systems*, 32, 2019.