



Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής

Πρόγραμμα Μεταπτυχιακών Σπουδών

«Πληροφορική»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Μελέτη του Εκτιμώμενου Χρόνου Άφιξης πλοίων στο λιμάνι του Πειραιά με την χρήση Νευρωνικών Δικτύων A Study on Ships' Estimated Time of Arrival at the Port of Piraeus using Neural Networks
Όνοματεπώνυμο Φοιτητή	Μαρία Πιάχα
Πατρώνυμο	Περικλής
Αριθμός Μητρώου	ΜΠΠΛ/15057
Επιβλέπων	Ιωάννης Θεοδωρίδης, Καθηγητής Πανεπιστημίου Πειραιώς

Ημερομηνία Παράδοσης

Σεπτέμβριος 2020

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

(υπογραφή)

(υπογραφή)

Ιωάννης Θεοδωρίδης
Καθηγητής

Νικόλαος Πελέκης
Αναπληρωτής Καθηγητής

Άγγελος Πικράκης
Επίκουρος Καθηγητής

Περίληψη

Το συνέδριο για την ασφάλεια της ζωής στη θάλασσα (SOLAS) που πραγματοποιήθηκε το 1974 ορίζει στον κανονισμό 19/Κεφάλαιο V τον εξοπλισμό ναυσιπλοΐας που μεταφέρεται επί των πλοίων, ανάλογα με τον τύπο τους. Το 2000, ο Διεθνής Ναυτιλιακός Οργανισμός υιοθέτησε μια νέα απαίτηση (ως μέρος ενός αναθεωρημένου νέου κεφαλαίου V) για όλα τα πλοία, να φέρουν συστήματα αυτόματης αναγνώρισης (AIS). Το σύστημα αυτό αποσκοπεί στο καλύτερο και αποτελεσματικότερο έλεγχο της θαλάσσιας κυκλοφορίας από τις αρμόδιες αρχές, αφού τα σήματα που στέλνουν τα πλοία περιέχουν τόσο δυναμικές πληροφορίες όπως η θέση του πλοίου αλλά και στατικές όπως η ταυτότητα του πλοίου. Το εργαστήριο Datastories του Πανεπιστημίου Πειραιά έχει εγκαταστήσει κεραία για την καταγραφή των σημάτων AIS και τη δημιουργία αντίστοιχης βάσης δεδομένων.

Επιπλέον, για την καλύτερη διαχείριση και οργάνωση των εργασιών εντός του λιμένα είναι σημαντική η γνώση του Εκτιμώμενου Χρόνου Άφιξης. Στα πλαίσια της παρούσας διπλωματικής θα μελετηθεί η συγκεκριμένη χρονική μεταβλητή. Η μελέτη και η εκτίμηση του χρόνου αυτού γίνεται με τη χρήση νευρωνικού δικτύου, το οποίο θα λαμβάνει ως είσοδο μια σειρά από δεδομένα και ως έξοδο θα δίνει τον εκτιμώμενο χρόνο άφιξης του πλοίου. Τα δεδομένα εισόδου αποτελούνται από τα σήματα AIS που έχουν αποθηκευτεί στη βάση του πανεπιστημίου. Η εκτίμηση του αποτελέσματος πραγματοποιείται με βάση τον πραγματικό χρόνο άφιξης του πλοίου και την απόκλιση από αυτόν. Τέλος, θα μελετηθούν ποιοι παράγοντες μπορούν να επηρεάσουν το αποτέλεσμα, όπως οι καιρικές συνθήκες ή ο χρονικός ορίζοντας των δεδομένων που διαθέτουμε.

Λέξεις - Κλειδιά: Νευρωνικά Δίκτυα, AIS, Estimated Time of Arrival – ETA, Εκτιμώμενος Χρόνος Άφιξης, Μοντέλο Πρόβλεψης

Abstract

The 1978 SOLAS Convention stipulates in Regulation 19 / Chapter V the navigation equipment to be carried on board on ships, according to their type. In 2000, the International Maritime Organization (IMO) adopted a new requirement (as part of a revised new chapter V) for all ships, to have Automatic Identification Systems (AIS). This system aims at better and more effective control of maritime traffic by the competent authorities since the ship's signals contain both dynamic information such as vessel's position and static data such as vessel's identity. The Datastories Lab of the University of Piraeus has installed an antenna for recording AIS signals and creating a corresponding database.

In addition, the knowledge of Estimated Time of Arrival (ETA) is important for better management and organization of port operations. In this thesis, this time variable will be studied. The study and estimation of this time is done by using a neural network, which will take as input a series of data and as output will give the estimated time of arrival of the ship. Input data consists of AIS signals stored at the database of the university. The evaluation of the result shall be based on the actual time of arrival of the ship and the deviation from it. Finally, we will study which factors can affect the outcome, such as the weather conditions or the time horizon of the data we have.

Keywords: Neural Networks, AIS, Estimated Time of Arrival - ETA, Prediction Model

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή κ. Γιάννη Θεοδωρίδη, καθηγητή ΠΑΠΕΙ, για την υποστήριξη και καθοδήγηση του καθ' όλη τη διάρκεια της εκπόνησης της παρούσας διπλωματικής εργασίας αλλά και για την ευκαιρία που μου έδωσε να ασχοληθώ με το συγκεκριμένο θέμα. Επίσης, θα ήθελα να ευχαριστήσω την κα. Ευαγγελία Χονδροδήμα, Διδάκτορα ΕΜΠ, και τον κ. Γιάννη Κοντούλη, Υποψήφιο Διδάκτορα ΠΑΠΕΙ, μέλη του εργαστηρίου *Datastories*, για τις πολύτιμες συμβουλές τους και τη βοήθειά τους στην επίλυση των διάφορων προβλημάτων.

Τέλος, θα ήθελα να ευχαριστήσω την οικογένεια μου και τους φίλους μου για την υποστήριξή τους σε όλη τη διάρκεια των σπουδών μου.

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

1	ΕΙΣΑΓΩΓΗ	9
1.1	ΠΕΡΙΓΡΑΦΗ ΕΡΓΑΣΙΑΣ	9
1.2	ΣΧΕΤΙΚΕΣ ΕΡΓΑΣΙΕΣ	9
1.3	ΔΟΜΗ ΕΡΓΑΣΙΑΣ	10
2	ΔΕΔΟΜΕΝΑ ΕΡΓΑΣΙΑΣ – ΕΡΓΑΛΕΙΑ ΑΝΑΛΥΣΗΣ	11
2.1	ΣΥΣΤΗΜΑΤΑ ΠΑΡΑΚΟΛΟΥΘΗΣΗΣ ΠΛΟΙΩΝ	11
2.2	ΣΥΣΤΗΜΑ AIS	12
2.3	ΑΡΧΗ ΛΕΙΤΟΥΡΓΙΑΣ AIS	13
2.4	ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ R	14
3	ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ	16
3.1	ΕΙΣΑΓΩΓΗ	16
	<i>Βασικά Χαρακτηριστικά ενός Νευρωνικού Δικτύου</i>	16
	<i>Αρχιτεκτονικές Νευρωνικών Δικτύων</i>	21
	<i>Εκπαίδευση Νευρωνικών Δικτύων</i>	23
3.2	PERCEPTRON ΕΝΟΣ ΕΠΙΠΕΔΟΥ	24
	<i>Γενικά</i>	24
	<i>Εκπαίδευση Perceptron – Αλγόριθμος Σύγκλισης</i>	26
3.3	PERCEPTRON ΠΟΛΛΑΠΛΩΝ ΕΠΙΠΕΔΩΝ	27
4	ΠΕΙΡΑΜΑΤΙΚΗ ΔΙΑΔΙΚΑΣΙΑ	31
4.1	ΠΡΟΒΛΕΨΗ ΤΗΣ ΕΤΑ ΜΕ ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ	31
4.2	ΑΝΑΛΥΣΗ ΚΩΔΙΚΑ ΜΟΝΤΕΛΟΥ	33
4.3	ΜΕΤΡΗΣΕΙΣ ΣΦΑΛΜΑΤΩΝ ΓΙΑ ΤΗΝ ΑΞΙΟΛΟΓΗΣΗ ΤΟΥ ΜΟΝΤΕΛΟΥ	35
4.4	ΑΠΟΤΕΛΕΣΜΑΤΑ	36
5	ΣΥΜΠΕΡΑΣΜΑΤΑ	38
6	ΑΝΑΦΟΡΕΣ	39
	ΠΑΡΑΡΤΗΜΑ Α	40
	<i>DataEngineering.R</i>	40
	<i>auxiliaryFunctions.R</i>	44
	<i>PredictiveModel.R</i>	45
	ΠΑΡΑΡΤΗΜΑ Β	46

ΠΕΡΙΕΧΟΜΕΝΑ ΕΙΚΟΝΩΝ

ΕΙΚΟΝΑ 1 - ΑΡΧΙΚΗ ΣΕΛΙΔΑ MARINE TRAFFIC[15].....	11
ΕΙΚΟΝΑ 2 - ΑΡΧΙΚΗ ΣΕΛΙΔΑ ΑΠΕΙΚΟΝΙΣΗΣ ΤΟΥ ΠΑΝΕΠΙΣΤΗΜΙΟΥ ΠΕΙΡΑΙΑ[18].....	12
ΕΙΚΟΝΑ 3 - ΔΙΑΓΡΑΜΜΑΤΙΚΗ ΑΠΕΙΚΟΝΙΣΗ ΛΕΙΤΟΥΡΓΙΑΣ ΣΥΣΤΗΜΑΤΟΣ AIS[17].....	14
ΕΙΚΟΝΑ 4 - ΠΕΡΙΒΑΛΛΟΝ RSTUDIO.....	15
ΕΙΚΟΝΑ 5 - ΔΟΜΗ ΑΝΘΡΩΠΙΝΟΥ ΝΕΥΡΩΝΙΚΟΥ ΣΥΣΤΗΜΑΤΟΣ[8].....	18
ΕΙΚΟΝΑ 6 - ΔΟΜΗ ΝΕΥΡΩΝΑ[8].....	18
ΕΙΚΟΝΑ 7 - ΣΥΝΑΡΤΗΣΗ ΚΑΤΩΦΛΙΟΥ[8].....	19
ΕΙΚΟΝΑ 8 - ΣΙΓΜΟΕΙΔΗΣ ΣΥΝΑΡΤΗΣΗ ΕΝΕΡΓΟΠΟΙΗΣΗΣ[8].....	19
ΕΙΚΟΝΑ 9 - ΑΝΑΠΑΡΑΣΤΑΣΗ ΤΩΝ ΣΥΝΑΡΤΗΣΕΩΝ ΤΑΝΗ, RELU ΚΑΙ PLU[].....	21
ΕΙΚΟΝΑ 10 - ΔΙΚΤΥΟ ΠΡΟΣΘΙΑΣ ΤΡΟΦΟΔΟΤΗΣΗΣ ΕΝΟΣ ΕΠΙΠΕΔΟΥ[8].....	21
ΕΙΚΟΝΑ 11 - ΠΟΛΥΕΠΙΠΕΔΟ ΔΙΚΤΥΟ ΠΡΟΣΘΙΑΣ ΤΡΟΦΟΔΟΤΗΣΗΣ[8].....	22
ΕΙΚΟΝΑ 12 - ΑΝΑΔΡΟΜΙΚΟ ΔΙΚΤΥΟ[8].....	23
ΕΙΚΟΝΑ 13 - ΡΕΡΣΕΡΤΡΟΝ ROSENBLATT.....	24
ΕΙΚΟΝΑ 14 - ΤΟ ΥΠΕΡΕΠΙΠΕΔΟ ΩΣ ΟΡΙΟ ΑΠΟΦΑΣΗΣ[8].....	26
ΕΙΚΟΝΑ 15 - ΝΕΥΡΩΝΙΚΟ ΔΙΚΤΥΟ ΜΕ ΔΥΟ ΚΡΥΦΑ ΕΠΙΠΕΔΑ[8].....	27
ΕΙΚΟΝΑ 16 - ΔΙΑΓΡΑΜΜΑΤΙΚΗ ΑΠΕΙΚΟΝΙΣΗ ΕΚΜΑΘΗΣΗΣ ΜΕ ΤΟΝ ΑΛΓΟΡΙΘΜΟ BACKPROPAGATION[8] ..	29
ΕΙΚΟΝΑ 17 - ΔΟΜΗ ΝΕΥΡΩΝΙΚΟΥ ΔΙΚΤΥΟΥ	32
ΕΙΚΟΝΑ 18 - ΣΥΝΤΕΤΑΓΜΕΝΕΣ ΛΙΜΕΝΑ ΠΕΙΡΑΙΑ	34
ΕΙΚΟΝΑ 19 - ΠΑΡΑΔΕΙΓΜΑ ΜΕΤΑΒΛΗΤΗΣ ΤΥΠΟΥ LAG	34

ΠΕΡΙΕΧΟΜΕΝΑ ΠΙΝΑΚΩΝ

ΠΙΝΑΚΑΣ 1 - ΜΕΤΑΒΛΗΤΕΣ ΑΛΓΟΡΙΘΜΟΥ ΣΥΓΚΛΙΣΗΣ	27
ΠΙΝΑΚΑΣ 2 - ΣΥΝΟΠΤΙΚΗ ΠΑΡΟΥΣΙΑΣΗ ΑΛΓΟΡΙΘΜΟΥ ΣΥΓΚΛΙΣΗΣ	27
ΠΙΝΑΚΑΣ 3 - ΜΕΤΑΒΛΗΤΕΣ ΕΙΣΟΔΟΥ.....	31
ΠΙΝΑΚΑΣ 4 - ΑΠΟΤΕΛΕΣΜΑΤΑ ΕΦΑΡΜΟΓΗΣ ΣΥΝΔΥΑΣΜΟΥ ΝΕΥΡΩΝΩΝ ΣΤΟ VALIDATION DATASET	37
ΠΙΝΑΚΑΣ 5 - ΑΠΟΤΕΛΕΣΜΑΤΑ ΕΦΑΡΜΟΓΗΣ ΒΕΛΤΙΣΤΟΥ ΣΥΝΔΥΑΣΜΟΥ ΝΕΥΡΩΝΩΝ ΣΤΟ TEST DATASET 37	

ΟΡΟΛΟΓΙΑ – ΑΚΡΩΝΥΜΙΑ

IMO	International Maritime Organization
UTC	Coordinated Universal Time
NN	Neural Network
ETA	Estimated Time of Arrival
AIS	Automatic Identification System
SOLAS Convention	Convention for Safety of Life at Sea
RMSE	Root Mean Square Error
MAE	Mean Absolute Error
MLP	Multi-Layer Perceptron
MMSI	Maritime Mobile Service Identity
VHF	Very High Frequency
GPS	Global Positioning System
SMVs	Support Vector Machines
GMDSS	Global Maritime Distress and Safety System
VTS	Vessel Traffic Service
GNSS	Global Navigation Satellite System
TDMA	Time Division Multiple Access
DSC	Digital Selective Calling
HDLC	High-Level Data Link Control
GMSK	Gaussian Minimum Shift Keying
FM	Frequency Modulation
VLSI	Very Large Scale Integrated
ΔΝΟ	Διεθνής Ναυτιλιακός Οργανισμός
ΝΔ	Νευρωνικό Δίκτυο
ΠαΠει	Πανεπιστήμιο Πειραιά
ΒΔ	Βάση Δεδομένων

1 ΕΙΣΑΓΩΓΗ

1.1 Περιγραφή Εργασίας

Ο βέλτιστος προγραμματισμός των δραστηριοτήτων ενός λιμένα καθορίζεται σε μεγάλο βαθμό από την αβεβαιότητα των χρόνων άφιξης των πλοίων σε αυτόν. Στην πλειοψηφία τους οι χρόνοι αυτοί βασίζονται στην εκτίμηση των ναυτιλιακών πρακτόρων, ενώ μπορεί να μην ανανεώνονται τακτικά και να περιέχουν σημαντικές αποκλίσεις, ιδίως σε μεγάλους χρονικούς ορίζοντες.

Στην Ελλάδα, η γνώση του ορθού χρόνου άφιξης των πλοίων επηρεάζει σε μεγάλο βαθμό την οργάνωση και την αποτελεσματική διαχείριση του Πειραιά, καθώς αποτελεί το μεγαλύτερο λιμάνι της χώρας και ένα από τα μεγαλύτερα της Μεσογείου. Αποτελεί αναπτυξιακό μοχλό του διεθνούς εμπορίου, της τοπικής και εθνικής οικονομίας, ενώ παράλληλα συνιστά κομβικό σημείο για την ακτοπλοϊκή σύνδεση των νησιών με την ηπειρωτική Ελλάδα, διεθνές κέντρο κρουαζιέρας, κέντρο διαμετακομιστικού εμπορίου για την ευρύτερη Μεσόγειο, εξυπηρετώντας ετησίως περισσότερα από 24.000 πλοία κάθε τύπου και μεγέθους. Στα πλαίσια της συμβολής του στην ανάπτυξη της τοπικής και εθνικής οικονομίας, η αναβάθμιση των υποδομών και η χρήση μοντέλων πρόβλεψης είναι καθοριστικής σημασίας[16].

Ως *μοντέλο πρόβλεψης* μπορούμε να θεωρήσουμε τη διαδικασία εφαρμογής ενός στατιστικού μοντέλου ή ενός αλγορίθμου εύρεσης δεδομένων σε δεδομένα με σκοπό την πρόβλεψη νέων ή μελλοντικών παρατηρήσεων[2]. Στη ναυτιλία, τα μοντέλα αυτά μπορούν να χρησιμοποιηθούν στην εκτίμηση της πορείας που μπορεί να ακολουθήσει ένα πλοίο, στην εκτίμηση του όγκου των πλοίων που δέχεται ένα λιμάνι καθημερινά, στην εκτίμηση της ώρας άφιξης ενός πλοίου στο λιμάνι κατάπλου ή άλλων παραμέτρων. Η σωστή και όσο το δυνατόν εγκυρότερη εκτίμηση των παραπάνω συμβάλλει στην καλύτερη οργάνωση και διαχείριση του λιμένα από τις λιμενικές αρχές.

Στην παρούσα εργασία, θα μελετηθεί η εκτίμηση της ώρας άφιξης των πλοίων (Estimated Time of Arrival – ETA) στο λιμάνι, λαμβάνοντας υπόψη την ιστορικότητα των δεδομένων. Η ορθή πρόβλεψη της μεταβλητής αυτής σχετίζεται τόσο με τον βέλτιστο προγραμματισμό των εργασιών του λιμανιού, ιδιαίτερα σε περιπτώσεις άφιξης πολλών πλοίων σε μικρό χρονικό εύρος, αλλά και με την καλύτερη εξυπηρέτηση του επιβατικού κοινού. Εκτός από την ιστορικότητα των δεδομένων, υπάρχουν παράγοντες που μπορούν να επηρεάσουν σημαντικά την ώρα άφιξης στο χρονικό διάστημα μελέτης. Ένας τέτοιος παράγοντας είναι οι καιρικές συνθήκες που επικρατούν σε διάφορα τμήματα της πορείας. Οι απότομες αλλαγές δημιουργούν πιθανώς καθυστερήσεις με αποτέλεσμα το μοντέλο να οδηγηθεί σε αστοχία.

Η πρόβλεψη της ώρας άφιξης θα γίνει με τη χρήση νευρωνικού δικτύου, το οποίο αρχικά θα εκπαιδευτεί με τα ιστορικά δεδομένα και στη συνέχεια θα προβλέψει τα μελλοντικά.

1.2 Σχετικές Εργασίες

Παρά την εξέλιξη της τεχνολογίας, η πρόβλεψη της εκτιμώμενης ώρας άφιξης των πλοίων είναι μια καθημερινή πρόκληση για τους διαχειριστές των λιμανιών.

Στην έρευνά τους οι Gambardella et al. [5] προτείνουν ένα εργαλείο πρόβλεψης που υπολογίζει τις καθημερινές ροές των εμπορευματοκιβωτίων μέσα και έξω από ένα τερματικό, συνδυάζοντας δύο διαφορετικά μέρη. Το πρώτο χρησιμοποιεί παρελθοντικά δεδομένα για να προβλέψει τον αριθμό των εμπορευματοκιβωτίων που φορτώνονται σε ένα πλοίο. Το δεύτερο μέρος υπολογίζει τον συνολικό αριθμό των εμπορευματοκιβωτίων που αναμένεται να μεταφερθούν με φορτηγό, ως συνάρτηση της εκτιμώμενης ώρας άφιξης του πλοίου. Το μοντέλο που περιγράφεται στο [6] μπορεί να προβλέψει την εκτιμώμενη ώρα άφιξης του πλοίου χρησιμοποιώντας νευρωνικά δίκτυα

(Neural Networks – NN) για να μειώσει την αβεβαιότητα του χρόνου άφιξης στο λιμένα κατά περίπου 6 ώρες. Ωστόσο, αφορά χρονικό ορίζοντα 24 ωρών, χωρίς να λαμβάνει υπόψη του πληροφορίες σχετικά με τις καιρικές συνθήκες, αλλά προσμετρώντας την καθυστέρηση των πλοίων λόγω λιμενικών δραστηριοτήτων. Αυτό το μοντέλο προσομοίωσης ήταν σε θέση να μειώσει τις αλλαγές βάρδιας από τέσσερις σε δύο και να μειώσει τα λειτουργικά κόστη του τερματικού. Ένα από τα μέρη του μοντέλου προέβλεψε την ETA του πλοίου με βάση τα ιστορικά δεδομένα.

Στην εργασία [7] προτείνονται δύο μοντέλα πρόβλεψης της ETA που χρησιμοποιούν NN και μηχανές διανυσμάτων υποστήριξης (Support Vector Machines - SVMs) αντίστοιχα για την εκτίμηση της ETA των πλοίων μεταφοράς εμπορευματοκιβωτίων στο λιμάνι του Ρότερνταμ για έναν μεσοπρόθεσμο χρονικό ορίζοντα. Προέκυψε ότι τα μοντέλα SVM υπερέχουν των NN κατά τη σύγκριση του απόλυτου μέσου σφάλματος (Mean Absolute Error -MAE) των μοντέλων σε όλες τις περιπτώσεις. Σε κάθε περίπτωση, τα δύο μοντέλα πέτυχαν σημαντικά καλύτερες προβλέψεις σε σύγκριση με την τρέχουσα κατάσταση, όπου η ETA βασίζεται στις εκτιμήσεις της ναυτιλιακής εταιρείας. Επίσης, παρά το γεγονός ότι και τα δύο μοντέλα λαμβάνουν υπόψη τις καιρικές συνθήκες, η επίδρασή τους δεν ήταν σημαντική στο κομμάτι της διαδρομής που μελετήθηκε, χωρίς αυτό να σημαίνει ότι δεν επηρεάζουν το χρόνο που ένα πλοίο φτάνει στο λιμάνι.

1.3 Δομή Εργασίας

Η συγκεκριμένη εργασία αποτελείται από πέντε κεφάλαια. Το κεφάλαιο 1 αποτελείται από την εισαγωγή και αφορά στην περιγραφή της εργασίας καθώς και στη σύνοψη εργασιών που αφορούν το συγκεκριμένο θέμα.

Στο κεφάλαιο 2 αναλύονται τα «εργαλεία» που χρησιμοποιήθηκαν. Γίνεται μια σύντομη αναφορά στους τρόπους παρακολούθησης των θέσεων των πλοίων και στη συνέχεια αναλύονται τα σήματα AIS, η χρησιμότητά τους και ο τρόπος λειτουργίας τους. Τέλος, γίνεται μια γρήγορη περιγραφή της γλώσσας R και του RStudio, που αποτελούν τα βασικά προγραμματιστικά εργαλεία.

Στο κεφάλαιο 3 γίνεται μια θεωρητική ανάλυση των Νευρωνικών Δικτύων. Πιο συγκεκριμένα, αναλύεται η δομή των δικτύων τύπου Perceptron καθώς και Πολυστρωματικά Νευρωνικά Δίκτυα (Multi-Layer Perceptron – MLP).

Στο κεφάλαιο 4 περιγράφεται η δομή του Νευρωνικού Δικτύου που χρησιμοποιήθηκε καθώς και ορισμένα βασικά σημεία του κώδικα, ενώ παρατίθενται συγκριτικά δεδομένα για τις διάφορες καταστάσεις του δικτύου.

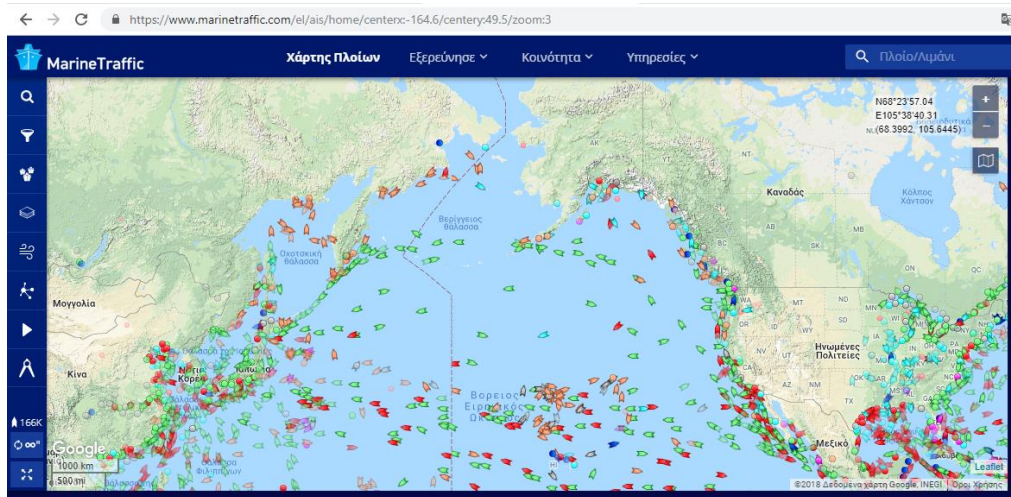
Τέλος, στο κεφάλαιο 5 παρουσιάζονται τα συμπεράσματα της παρούσας εργασίας.

2 ΔΕΔΟΜΕΝΑ ΕΡΓΑΣΙΑΣ – ΕΡΓΑΛΕΙΑ ΑΝΑΛΥΣΗΣ

Στο παρόν κεφάλαιο γίνεται μια εισαγωγή σε βασικές έννοιες που χρησιμοποιήθηκαν για την ανάπτυξη του μοντέλου. Πιο συγκεκριμένα, στην Ενότητα 2.1 γίνεται αναφορά στα συστήματα απεικόνισης των θέσεων των πλοίων. Στην Ενότητα 2.2 εισάγεται η έννοια του συστήματος AIS, η αρχή λειτουργίας του οποίου περιγράφεται λεπτομερώς στην Ενότητα 2.3. Τέλος, στην Ενότητα 2.4 γίνεται μία σύντομη αναφορά στην Γλώσσα Προγραμματισμού R, που αποτελεί το βασικό εργαλείο προγραμματισμού της εφαρμογής.

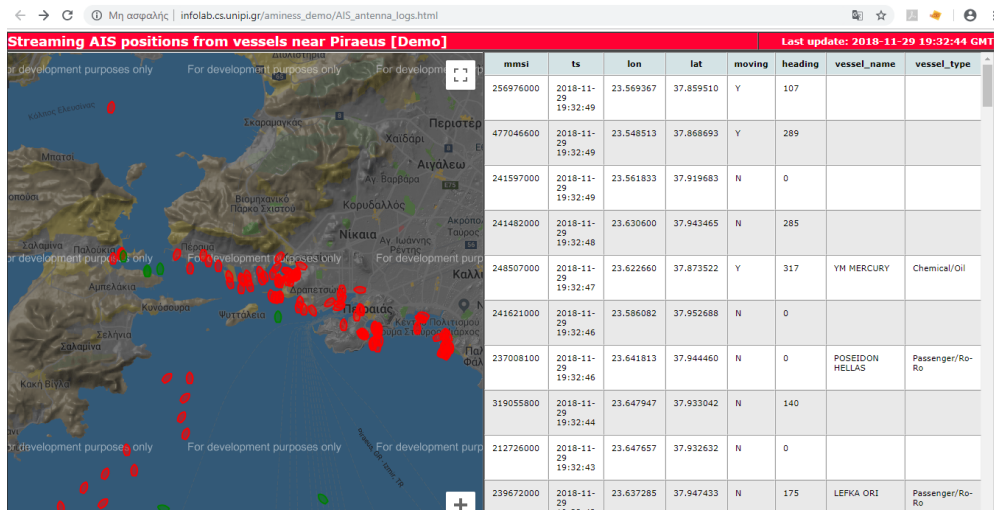
2.1 Συστήματα Παρακολούθησης Πλοίων

Οι θέσεις των πλοίων καθώς και η καταγραφή των παραμέτρων του ταξιδιού τους, όπως η αναμενόμενη πορεία τους ή ο χρόνος άφιξης του στο λιμάνι προορισμού τους, είναι καθοριστικής σημασίας για την ομαλή λειτουργία του λιμένα, αφού συμβάλλει αποφασιστικά στη βελτίωση της διαχείρισής του. Για το λόγο αυτό υπάρχουν διάφορες ιστοσελίδες που δείχνουν σε πραγματικό χρόνο τις θέσεις όλων των πλοίων ανά τον κόσμο. Ο πιο γνωστός, ίσως, ιστότοπος βρίσκεται στον ακόλουθο σύνδεσμο www.marinetraffic.com. Ο χρήστης, είτε είναι ο διαχειριστής του λιμανιού είτε απλός χρήστης, μπορεί να δει ανά πάσα στιγμή τις συντεταγμένες του πλοίου που τον ενδιαφέρει.



Εικόνα 1 - Αρχική Σελίδα Marine Traffic[15]

Ανάλογη με την παραπάνω ιστοσελίδα, είναι και αυτή του Πανεπιστημίου Πειραιά. Από το εργαστήριο Datastories του πανεπιστημίου έχει εγκατασταθεί κεραία, η οποία λειτουργεί ως σταθμός λήψης AIS σημάτων όλων των πλοίων που κινούνται στο λιμάνι του Πειραιά καθώς και στην ευρύτερη περιοχή του Αργοσαρωνικού μέχρι την Αίγινα. Τα σήματα αυτά αποθηκεύονται σε μία γεωπληροφοριακή βάση δεδομένων, η οποία περιλαμβάνει τις συντεταγμένες των πλοίων (γεωγραφικό μήκος, γεωγραφικό πλάτος), την Ταυτότητα Ναυτιλιακής Κινητής Υπηρεσίας (MMSI), τη χρονοσφραγίδα (timestamp), την ταχύτητα εδάφους και την κατεύθυνση του πλοίου. Για την απεικόνισή τους, ο ενδιαφερόμενος μπορεί να χρησιμοποιήσει την ιστοσελίδα http://infolab.cs.unipi.gr/aminess_demo/AIS_antenna_logs.html.



Εικόνα 2 - Αρχική Σελίδα Απεικόνισης του Πανεπιστημίου Πειραιά[18]

Στα πλαίσια της παρούσας διπλωματικής, τα δεδομένα που θα χρησιμοποιήσουμε προέρχονται από τη ΒΔ του ΠαΠει και αφορούν παρατηρήσεις που συλλέχθηκαν τη χρονική περίοδο Ιανουάριος 2016 – Δεκέμβριος 2016.

2.2 Σύστημα AIS

Σύμφωνα με τον κανονισμό 19 της SOLAS/Κεφάλαιο V - Απαιτήσεις μεταφοράς για συστήματα πλοήγησης και εξοπλισμού πλοίων - ορίζεται ο εξοπλισμός ναυσιπλοΐας που μεταφέρεται επί των πλοίων, ανάλογα με τον τύπο του πλοίου. Το 2000, ο *Διεθνής Ναυτιλιακός Οργανισμός - ΔΝΟ (International Marine Organization – IMO)* υιοθέτησε μια νέα απαίτηση (ως μέρος ενός αναθεωρημένου νέου κεφαλαίου V) για όλα τα πλοία, να φέρουν συστήματα αυτόματης αναγνώρισης (AIS) [19]. Ο ΔΝΟ ανέπτυξε και εφάρμοσε, αρχικά, το παραπάνω σύστημα με στόχο την αποφυγή συγκρούσεων, τον αποτελεσματικότερο έλεγχο της ναυτιλιακής κυκλοφορίας, τη βελτίωση του επιπέδου ασφαλείας κατά τον πλου, την αναγνώριση στόχων και την παρακολούθηση αυτών και γενικότερα την άσκηση ασφαλέστερης ναυτιλίας. Σήμερα, το σύστημα αυτό εξυπηρετεί πολλούς σκοπούς και χρησιμοποιείται από το σύνολο των Αρμόδιων Αρχών της Ναυτιλίας.

Ο κανονισμός απαιτεί συστήματα AIS να τοποθετηθούν πάνω σε όλα τα πλοία ολικής χωρητικότητας 300 τόνων και άνω που εκτελούν διεθνή δρομολόγια, σε φορτηγά πλοία 500 GT και άνω που δεν εκτελούν διεθνή δρομολόγια και σε όλα τα επιβατηγά πλοία ανεξάρτητα από το μέγεθός τους. Η απαίτηση αυτή ισχύει για όλα τα πλοία από τις 31 Δεκεμβρίου 2004. Το σύστημα AIS πρέπει να είναι σε λειτουργία σε όλα τα πλοία που το φέρουν ανά πάσα στιγμή, εκτός εάν προβλέπεται από διεθνείς συμφωνίες, κανόνες ή πρότυπα προστασίας των πληροφοριών πλοήγησης. Ένα κράτος μπορεί να απαλλάξει πλοία από τη μεταφορά AIS, όταν τα πλοία θα τεθούν μόνιμα εκτός λειτουργίας εντός δύο ετών από την ημερομηνία εφαρμογής. Τα πρότυπα απόδοσης για το AIS εγκρίθηκαν το 1998[19].

Το *Αυτόματο Σύστημα Αναγνώρισης (Automatic Identification System – AIS)* είναι ένα σύστημα αυτόματης ανταλλαγής ψηφιακών σημάτων μεταξύ των σταθμών λήψης AIS (πλοία, δορυφόροι, παράκτιοι σταθμοί παρακολούθησης κυκλοφορίας πλοίων), στη συχνότητα των υπερβραχέων κυμάτων (VHF)[1][17]. Μέσω του συστήματος αυτού στατικές και δυναμικές πληροφορίες των πλοίων (ταυτότητα, φορτίο, λιμένας απόπλου και κατάπλου και άλλες πληροφορίες) μπορούν να ανταλλάσσονται ηλεκτρονικά.

Οι πληροφορίες που λαμβάνονται μέσω του Αυτόματου Συστήματος Αναγνώρισης περιλαμβάνουν 3 είδη παραμέτρων:

- Στατικές παραμέτρους, δηλαδή:
 1. τη ναυτιλιακή κινητή δορυφορική ταυτότητα (MMSI)
 2. τον αριθμό αναγνώρισης IMO
 3. το όνομα του πλοίου (έως 20 χαρακτήρες) και το διακριτικό κλήσεως
 4. τις διαστάσεις του πλοίου στρογγυλοποιημένες σε ακέραιο αριθμό μέτρων
 5. τον τύπο του πλοίου (δεξαμενόπλοιο, κρουαζιερόπλοιο κλπ)
 6. τη θέση που είναι εγκατεστημένο το σύστημα πάνω στο πλοίο
 7. τον τύπο ηλεκτρονικής συσκευής προσδιορισμού στίγματος (απλό ή διαφορικό GPS)
- Δυναμικές παραμέτρους, δηλαδή:
 1. τη θέση του πλοίου
 2. τη συντονισμένη παγκόσμια ώρα (UTC)
 3. την αληθή πορεία από 0° έως 359°, όπως αυτή εισάγεται από τη γυροπυξίδα
 4. την πορεία ως προς το βυθό
 5. την ταχύτητα ως προς το βυθό
 6. τη ναυτιλιακή κατάσταση του πλοίου (εν πλω, αγκυροβολημένο, ακυβέρνητο κλπ.)
 7. το ρυθμός στροφής, δεξιά (+) ή αριστερά (-)
 8. το ρυθμός ανανέωσης αναφοράς
- Παραμέτρους ταξιδιού, δηλαδή:
 1. το βύθισμα του πλοίου
 2. τον τύπο του φορτίου
 3. τον προορισμό
 4. τον εκτιμώμενο χρόνο κατάπλου (μήνας, ημέρα, ώρα και λεπτό σε συγχρονισμένο παγκόσμιο χρόνο)

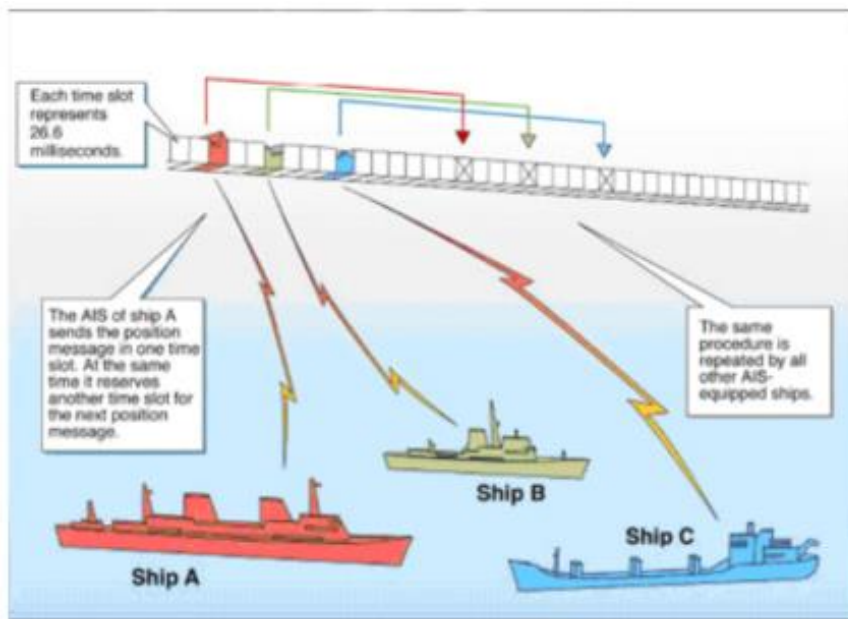
2.3 Αρχή Λειτουργίας AIS

Κάθε σύστημα AIS αποτελείται από έναν πομπό VHF, δύο δέκτες VHF TDMA, έναν δέκτη VHF DSC και από τυπικούς θαλάσσιους ηλεκτρονικούς συνδέσμους επικοινωνίας σε συστήματα πλοήγησης και αισθητήρων. Οι πληροφορίες θέσης και χρονισμού προέρχονται συνήθως από ένα εσωτερικό ή εξωτερικό σύστημα δορυφορικής πλοήγησης (π.χ. GPS), συμπεριλαμβανομένου ενός διαφορικού GNSS δέκτη μέσης συχνότητας για ακριβή θέση σε παράκτια και εγχώρια ύδατα. Έτσι, οι πληροφορίες για την κατεύθυνση, την πορεία και την ταχύτητα θα λαμβάνονται από όλα τα πλοία εξοπλισμένα με AIS[1][17].

Το AIS λειτουργεί κανονικά με αυτόνομο και συνεχή τρόπο, ανεξάρτητα από το αν λειτουργεί σε ανοιχτές θάλασσες ή σε παράκτιες ή εγχώριες περιοχές. Οι μεταδόσεις χρησιμοποιούν διαμόρφωση συχνότητας (FM GMSK) 9,6 kb σε κανάλια των 25 ή 12,5 kHz χρησιμοποιώντας πρωτόκολλα πακέτων HDLC. Παρόλο που είναι απαραίτητο μόνο ένα ραδιοφωνικό κανάλι, κάθε σταθμός μεταδίδει και λαμβάνει πάνω από δύο ραδιοφωνικά κανάλια για την αποφυγή προβλημάτων παρεμβολής και για να επιτρέπεται η εναλλαγή των καναλιών χωρίς απώλεια επικοινωνίας με άλλα πλοία. Το σύστημα παρέχει αυτόματη επίλυση διαμάχης μεταξύ του ίδιου και των άλλων σταθμών, διατηρώντας έτσι την ακεραιότητα των επικοινωνιών.

Κάθε σταθμός καθορίζει το δικό του πρόγραμμα εκπομπής, βάσει του ιστορικού της κυκλοφορίας και της γνώσης των μελλοντικών ενεργειών από άλλους σταθμούς. Μια αναφορά θέσης από έναν σταθμό AIS ταιριάζει σε μία από τις 2250 χρονικές θυρίδες που καθορίζονται κάθε 60 δευτερόλεπτα. Οι σταθμοί AIS συγχρονίζονται συνεχώς μεταξύ τους, για να αποφευχθεί η επικάλυψη των μεταδόσεων. Η επιλογή της θυρίδας από ένα σταθμό AIS γίνεται τυχαία μέσα σε ένα καθορισμένο διάστημα και με χρονικό όριο από 0 έως 8. Όταν ένας σταθμός αλλάξει την εκχώρησή του, ανακοινώνει εκ των προτέρων τόσο τη νέα θέση όσο και το χρονικό όριο για την τοποθεσία αυτή.

Με τον τρόπο αυτό, τόσο οι νέοι σταθμοί όσο και οι σταθμοί που βρίσκονται ξαφνικά σε περιοχή ραδιοσυχνότητας κοντά σε άλλα σκάφη, θα λαμβάνουν πάντοτε τα σκάφη αυτά[17].



Εικόνα 3 - Διαγραμματική Απεικόνιση Λειτουργίας Συστήματος AIS[17]

Η απαιτούμενη ικανότητα αναφοράς πλοίου σύμφωνα με το πρότυπο απόδοσης του ΔΝΟ ανέρχεται σε τουλάχιστον 2000 χρονοθυρίδες ανά λεπτό, αν και το σύστημα παρέχει 4500 χρονοθυρίδες ανά λεπτό. Σε περίπτωση υπερφόρτωσης του συστήματος, θα υπάρξει αποκλεισμός μόνο για τους στόχους που βρίσκονται πολύ μακριά, προκειμένου να δοθεί προτεραιότητα σε πιο κοντινούς στόχους, οι οποίοι αποτελούν πρωταρχικό μέλημα για τις ναυτιλιακές αρχές. Στην πράξη, η χωρητικότητα του συστήματος είναι σχεδόν απεριόριστη, επιτρέποντας να φιλοξενηθεί ταυτόχρονα μεγάλος αριθμός πλοίων.

Το εύρος κάλυψης του συστήματος είναι παρόμοιο με άλλες εφαρμογές VHF και στην ουσία εξαρτάται από το ύψος της κεραίας. Η διάδοσή του είναι ελαφρώς καλύτερη από αυτή του ραντάρ, λόγω του μεγαλύτερου μήκους κύματος, οπότε είναι δυνατό να «βλέπουμε» και πίσω από τα νησιά, αν δεν είναι πολύ ψηλά. Με τη βοήθεια επαναληπτών, η κάλυψη για σταθμούς πλοίων και σταθμούς VTS μπορεί να βελτιωθεί σημαντικά. Το σύστημα είναι συμβατό με ψηφιακά συστήματα επιλεκτικής κλήσης, επιτρέποντας στα συστήματα ξηράς GMDSS να δημιουργούν με ακρίβεια κανάλια AIS και έτσι να εντοπίζουν και να παρακολουθούν πλοία εξοπλισμένα με AIS.

2.4 Γλώσσα Προγραμματισμού R

Η υλοποίηση του μοντέλου πραγματοποιήθηκε στη γλώσσα προγραμματισμού R, η οποία είναι μια γλώσσα προγραμματισμού ανοικτού κώδικα που παρέχει στον χρήστη τη δυνατότητα να κάνει υπολογιστική στατιστική και γραφήματα. Προέρχεται από τη γλώσσα προγραμματισμού S, η οποία δημιουργήθηκε στα Εργαστήρια Bell από τον John Chambers, γι' αυτό και έχουν πολλές ομοιότητες [20][21]. Μερικά από τα βασικά εργαλεία για τις στατιστικές αναλύσεις που παρέχει η R είναι:

- δημιουργία τυχαίων δειγμάτων
- διακριτές και συνεχείς μεταβλητές (Poisson, Gamma, Exponential κτλ)
- έλεγχοι υποθέσεων

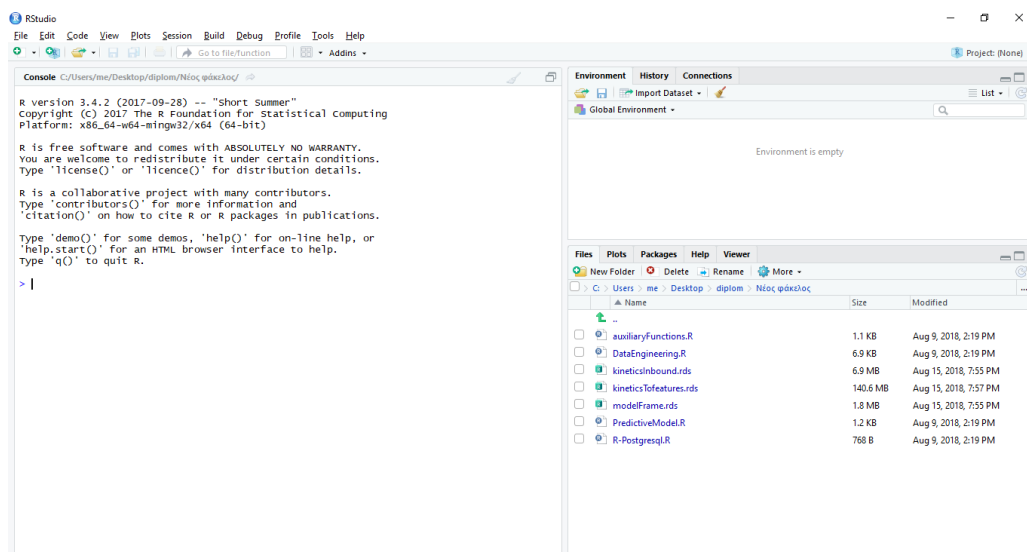
- στατιστικά τεστ (Kolmogorov-Smirnoff)
- δημιουργία γραφημάτων (ιστόγραμμα, qq plot, pie chart, bar chart κτλ)

Είναι GNU λογισμικό, δηλαδή ο καθένας μπορεί να κάνει βελτιώσεις στον πηγαίο κώδικά της και να τις δημοσιεύει και επίσης διανέμεται δωρεάν. Αυτός είναι και ο βασικός λόγος που έχουν γίνει πολλές βελτιώσεις από τότε που δημιουργήθηκε. Οι αρχικοί δημιουργοί ήταν οι Robert Gentleman και Ross Ihaka, αργότερα όμως συνέβαλλαν και άλλοι στην ανάπτυξη της.

H R μπορεί να κάνει τα ίδια πράγματα με την S και με πολύ λιγότερο κώδικα. Είναι open source περιβάλλον, άρα όλοι μπορούν να έχουν πρόσβαση στον κώδικά της και να κάνουν διορθώσεις. Επιπλέον, επιτρέπει στον χρήστη να αλληλεπιδρά με άλλες γλώσσες (C/C++, Java, Python), με αρχεία δεδομένων (Excel, Access) και με άλλα στατιστικά πακέτα (SAS, Stata, SPSS, Minitab). Τέλος, το περιβάλλον της R παρέχεται δωρεάν.

H R δεν συνιστάται για ανάλυση μεγάλων δεδομένων, αφού το βασικό μειονέκτημά της είναι ότι καταναλώνει πολλή μνήμη. Είναι «αργή» γλώσσα και άρα ως προς τον χρόνο εκτέλεσης των εντολών δεν είναι τόσο αποδοτική. Παρόλα αυτά, επειδή η R διαθέτει πάνω από 5000 πακέτα, χρησιμοποιείται σε πολλούς επιστημονικούς τομείς καθώς και από εταιρείες που κάνουν αναλύσεις δεδομένων, όπως η Google, LinkedIn, Facebook, ενώ η οικονομία, η χημεία, η ιατρική και η φαρμακευτική είναι μερικοί από τους τομείς που χρησιμοποιούν τη συγκεκριμένη γλώσσα προγραμματισμού.

Το προγραμματιστικό περιβάλλον που χρησιμοποιήθηκε είναι το RStudio. Πρόκειται για ένα ευρέως χρησιμοποιούμενο επαγγελματικό λογισμικό ανοιχτού κώδικα που συνδυάζει αξιόπιστη ανάλυση δεδομένων με εργαλεία για την αποτελεσματική ανταλλαγή δεδομένων. Παρακάτω φαίνεται μια αναπαράσταση του περιβάλλοντος του RStudio [20][21].



Εικόνα 4 - Περιβάλλον RStudio

3 ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Στο παρόν κεφάλαιο γίνεται στην Ενότητα 3.1 μια εισαγωγή στις βασικές αρχές των Νευρωνικών Δικτύων. Αφού δίνεται ένας σύντομος ορισμός, αναλύονται οι ιδιότητές τους καθώς και οι διάφοροι τύποι αρχιτεκτονικών. Επιπλέον, γίνεται αναφορά στις συναρτήσεις ενεργοποίησης αλλά και στους τρόπους εκπαίδευσής τους. Στην Ενότητα 3.2, παρουσιάζεται η δομή και ο τρόπος λειτουργίας ενός ΝΔ ενός επιπέδου, το οποίο αποτελεί την απλούστερη μορφή ΝΔ και καλείται Perceptron του Rosenblatt ή απλά Perceptron. Τέλος, στην Ενότητα 3.3 παρουσιάζονται τα πολυστρωματικά νευρωνικά δίκτυα ή Multilayer Perceptron (MLPs), καθώς και ο αλγόριθμος οπίσθιας διάδοσης της ενεργοποίησης/σφάλματος ή Back Propagation Algorithm.

3.1 Εισαγωγή

Βασικά Χαρακτηριστικά ενός Νευρωνικού Δικτύου

Τα τεχνητά νευρωνικά δίκτυα έχουν σαν στόχο την προσομοίωση των βιολογικών νευρωνικών δικτύων του ανθρώπινου εγκεφάλου [8]. Ουσιαστικά, πρόκειται για ένα σύνολο από ισχυρά μαθηματικά εργαλεία που μπορούν να μοντελοποιήσουν άγνωστα μη γραμμικά συστήματα, χρησιμοποιώντας μόνο δεδομένα εισόδου-εξόδου από αυτά. Το δίκτυο, συνήθως, υλοποιείται με χρήση ηλεκτρονικών εξαρτημάτων ή προσομοιώνεται με λογισμικό σε υπολογιστή. Για την επίτευξη καλής απόδοσης, τα νευρωνικά δίκτυα χρησιμοποιούν μια τεράστια διασύνδεση απλών υπολογιστικών κυττάρων που αναφέρονται ως "νευρώνες" ή "μονάδες επεξεργασίας".

Ένας ορισμός που μπορεί να δοθεί, επομένως, είναι ο εξής:

Ένα νευρωνικό δίκτυο είναι ένας μαζικά παράλληλος κατανεμημένος επεξεργαστής αποτελούμενος από απλές μονάδες επεξεργασίας που έχει μια φυσική τάση για την αποθήκευση των βιωματικών γνώσεων και τη διάθεση αυτών για χρήση. Ένα νευρωνικό δίκτυο διατηρεί τις βασικές αρχές της παράλληλης κατανεμημένης επεξεργασίας και της ευέλικτης μάθησης σε δίκτυα απλών διασυνδεδεμένων κόμβων. Μοιάζει με τον εγκέφαλο από δύο απόψεις:

1. Η γνώση του δικτύου αποκτάται από το περιβάλλον του μέσα από μια διαδικασία μάθησης.
2. Τα συναπτικά βάρη (synaptic weights) χρησιμοποιούνται για την αποθήκευση της αποκτηθείσας γνώσης [8].

Ένα νευρωνικό δίκτυο παράγει την υπολογιστική του δύναμη μέσω της μαζικά παράλληλης κατανεμημένης δομής του και της ικανότητάς του να μαθαίνει και, επομένως, να γενικεύει, γεγονός που του επιτρέπει να βρίσκει καλές προσεγγιστικές λύσεις σε πολύπλοκα (μεγάλης κλίμακας) προβλήματα. Στην πράξη, ένα πολύπλοκο πρόβλημα ενδιαφέροντος αποσυντίθεται σε διάφορα σχετικά απλά καθήκοντα και τα νευρωνικά δίκτυα αναλαμβάνουν ένα υποσύνολο των εργασιών που ταιριάζουν με τις εγγενείς δυνατότητές τους.

Τα νευρωνικά δίκτυα προσφέρουν τις ακόλουθες χρήσιμες ιδιότητες και δυνατότητες:

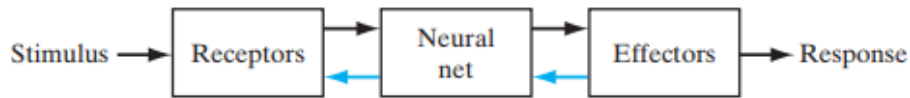
1. *Μη Γραμμικότητα:* Ένα νευρωνικό δίκτυο, που αποτελείται από διασυνδεδεμένους μη γραμμικούς νευρώνες, είναι από μόνο του μη γραμμικό. Η μη γραμμικότητα είναι μια εξαιρετικά σημαντική ιδιότητα, ιδιαίτερα αν ο υποκείμενος φυσικός μηχανισμός που είναι υπεύθυνος για την παραγωγή του σήματος εισόδου είναι εγγενώς μη γραμμικός.
2. *Αντιστοίχιση Εισόδου – Εξόδου:* Ένα δημοφιλές παράδειγμα μάθησης, που ονομάζεται μάθηση υπό επίτηρηση, περιλαμβάνει την τροποποίηση των συναπτικών βαρών ενός νευρωνικού δικτύου, εφαρμόζοντας ένα σύνολο εκπαιδευτικών παραδειγμάτων ή εργασιών. Κάθε παράδειγμα αποτελείται από ένα μοναδικό σήμα εισόδου και μια αντίστοιχη επιθυμητή απάντηση. Στο δίκτυο παρουσιάζεται με ένα παράδειγμα, που συλλέγεται τυχαία από το σύνολο, και τα συναπτικά βάρη τροποποιούνται για να ελαχιστοποιηθεί η διαφορά μεταξύ της επιθυμητής απόκρισης και της πραγματικής απόκρισης του δικτύου που παράγεται από το σήμα εισόδου σύμφωνα με ένα κατάλληλο στατιστικό κριτήριο. Η εκπαίδευση του δικτύου επαναλαμβάνεται για πολλά παραδείγματα στο σύνολο, έως ότου το δίκτυο φθάσει σε σταθερή κατάσταση όπου

δεν υπάρχουν άλλες σημαντικές αλλαγές στα συναπτικά βάρη. Τα παραδείγματα που χρησιμοποιήθηκαν αρχικά, μπορεί να εφαρμοστούν ξανά κατά τη διάρκεια του σταδίου της εκπαίδευσης, αλλά σε διαφορετική σειρά. Έτσι, το δίκτυο μαθαίνει από τα παραδείγματα κατασκευάζοντας μια χαρτογράφηση εισόδου-εξόδου για το πρόβλημα.

3. *Προσαρμοστικότητα*: Τα νευρωνικά δίκτυα έχουν μια ενσωματωμένη δυνατότητα προσαρμογής των συναπτικών βαρών στις αλλαγές στο περιβάλλον. Συγκεκριμένα, ένα νευρωνικό δίκτυο εκπαιδευμένο να λειτουργεί σε ένα συγκεκριμένο περιβάλλον, μπορεί εύκολα να ανακατασκευαστεί για να ανταπεξέλθει στις μικρές αλλαγές των περιβαλλοντικών συνθηκών λειτουργίας. Επιπλέον, όταν λειτουργεί σε μη στατικό περιβάλλον (δηλαδή σε ένα περιβάλλον, όπου τα στατιστικά μεγέθη αλλάζουν με το χρόνο), ένα νευρωνικό δίκτυο μπορεί να σχεδιαστεί να μεταβάλλει τα συναπτικά του βάρη σε πραγματικό χρόνο. Η φυσική αρχιτεκτονική ενός νευρωνικού δικτύου για ταξινόμηση προτύπων, επεξεργασία σήματος και έλεγχο εφαρμογών, σε συνδυασμό με την προσαρμοστική ικανότητα του δικτύου, το καθιστούν χρήσιμο εργαλείο στην προσαρμοστική ταξινόμηση προτύπων, στην προσαρμοστική επεξεργασία σήματος και στον προσαρμοστικό έλεγχο. Ως γενικός κανόνας μπορεί να ειπωθεί, ότι όσο πιο προσαρμοστικό κάνουμε ένα σύστημα διασφαλίζοντας συνεχώς ότι το σύστημα παραμένει σταθερό, τόσο ισχυρότερη θα είναι η απόδοσή του όταν το σύστημα θα πρέπει να λειτουργήσει σε μη σταθερό περιβάλλον.
4. *Ενδεικτική Απόκριση*: Στο πλαίσιο της ταξινόμησης προτύπων, ένα νευρωνικό δίκτυο μπορεί να σχεδιαστεί για να παρέχει πληροφορίες όχι μόνο για την επιλογή του κατάλληλου μοτίβου αλλά και για την ορθότητα της απόφασης.
5. *Πληροφορία Σχετική με το Περιεχόμενο*: Η γνώση αντιπροσωπεύεται από την ίδια τη δομή και κατάσταση ενεργοποίησης ενός νευρωνικού δικτύου. Κάθε νευρώνας στο δίκτυο επηρεάζεται ενδεχομένως από τη συνολική δραστηριότητα όλων των άλλων νευρώνων στο δίκτυο. Συνεπώς, οι συναφείς πληροφορίες αντιμετωπίζεται φυσικά από το νευρωνικό δίκτυο.
6. *Ανοχή σε Σφάλματα*: Ένα νευρωνικό δίκτυο, που υλοποιείται σε μορφή hardware, έχει τη δυνατότητα να είναι εγγενώς ανεκτικό σε σφάλματα ή ικανό για αξιόπιστους υπολογισμούς, με την έννοια ότι η απόδοσή του υποβαθμίζεται σταδιακά κάτω από δυσμενείς συνθήκες λειτουργίας.
7. *Δυνατότητα Υλοποίησης σε VLSI*: Η μαζικά παράλληλη φύση ενός νευρωνικού δικτύου το καθιστά δυνητικά γρήγορο για τον υπολογισμό ορισμένων εργασιών. Το ίδιο χαρακτηριστικό κάνει το νευρωνικό δίκτυο κατάλληλο για υλοποίηση χρησιμοποιώντας τεχνολογία πολύ μεγάλης κλίμακας (Very-Large-Scale-Integrated - VLSI). Ένα πολύ σημαντικό πλεονέκτημα του VLSI είναι ότι προσφέρει ένα αξιόπιστο μέσο για τον εντοπισμό πολύπλοκων συμπεριφορών με ιεραρχικό τρόπο.
8. *Ομοιογένεια Ανάλυσης και Σχεδιασμού*: Τα νευρωνικά δίκτυα απολαμβάνουν καθολικότητα ως επεξεργαστές πληροφοριών, με την έννοια ότι χρησιμοποιείται ο ίδιος συμβολισμός σε όλα τα πεδία που αφορούν την εφαρμογή των νευρωνικών δικτύων. Αυτή η δυνατότητα εκδηλώνεται με διαφορετικούς τρόπους:
 - i. Οι νευρώνες αντιπροσωπεύουν ένα κοινό συστατικό για όλα τα νευρωνικά δίκτυα.
 - ii. Αυτή η ικανότητα καθιστά δυνατή την ανταλλαγή θεωριών και αλγορίθμων μάθησης σε διαφορετικές εφαρμογές των νευρωνικών δικτύων.
 - iii. Τα αρθρωτά δίκτυα μπορούν να κατασκευαστούν μέσω ομαλής ενσωμάτωσης τμημάτων.
9. *Νευροβιολογική Αναλογία*: Ο σχεδιασμός ενός νευρωνικού δικτύου παρουσιάζει αναλογία με τον εγκέφαλο, ο οποίος είναι ζωντανή απόδειξη ότι η ανεκτική σε λάθη παράλληλη επεξεργασία δεν είναι μόνο φυσικά εφικτή, αλλά γρήγορη και ισχυρή. Οι νευροβιολόγοι εξετάζουν τα (τεχνητά) νευρωνικά δίκτυα ως ερευνητικό εργαλείο για την ερμηνεία νευροβιολογικών φαινομένων. Από την άλλη πλευρά, οι μηχανικοί αναζητούν στη νευροβιολογία νέες ιδέες για να λύσουν προβλήματα πιο περίπλοκα από εκείνα που βασίζονται σε συμβατικές τεχνικές σχεδιασμού.

Όπως αναφέρθηκε, ένα νευρωνικό δίκτυο παρουσιάζει αναλογίες με το ανθρώπινο νευρικό σύστημα. Για να εντοπίσουμε την ομοιότητα αυτή αρκεί να δούμε τις βασικές δομές των δύο παραπάνω.

Το ανθρώπινο νευρικό σύστημα μπορεί να θεωρηθεί ως ένα σύστημα τριών σταδίων, όπως απεικονίζεται παρακάτω σχήμα.

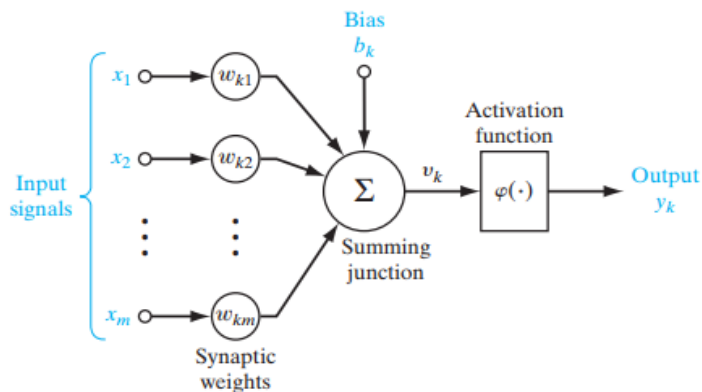


Εικόνα 5 - Δομή Ανθρώπινου Νευρικού Συστήματος[8]

Κέντρο του συστήματος είναι ο εγκέφαλος, που αντιπροσωπεύεται από το νευρωνικό δίκτυο (Neural Network), το οποίο δέχεται συνεχώς πληροφορίες, τις επεξεργάζεται και λαμβάνει τις κατάλληλες αποφάσεις. Στο σχήμα φαίνονται δύο σειρές βέλη. Αυτά που δείχνουν από αριστερά προς τα δεξιά δείχνουν την εμπρόσθια μετάδοση σημάτων που φέρουν πληροφορίες μέσω του συστήματος. Τα βέλη που δείχνουν από δεξιά προς τα αριστερά (μπλε χρώμα) σημαίνουν την ύπαρξη ανατροφοδότησης στο σύστημα. Οι υποδοχείς (Receptors) μετατρέπουν τα ερεθίσματα από το ανθρώπινο σώμα ή το εξωτερικό περιβάλλον σε ηλεκτρικούς παλμούς που μεταφέρουν πληροφορίες στο νευρωνικό δίκτυο (εγκέφαλος). Οι τελεστές (Effectors) μετατρέπουν τους ηλεκτρικούς παλμούς που παράγονται από το νευρωνικό δίκτυο σε αποκρίσεις, ως εξόδους του συστήματος[8].

Οι συνάψεις ή οι νευρικές απολήξεις είναι στοιχειώδεις δομικές και λειτουργικές μονάδες που μεσολαβούν στις αλληλεπιδράσεις μεταξύ των νευρώνων. Το πιο κοινό είδος συνάψεως είναι η χημική σύναψη, η οποία λειτουργεί ως εξής: Μια προσυναπτική διαδικασία απελευθερώνει μια ουσία που διαχέεται κατά μήκος της συναπτικής σύνδεσης μεταξύ νευρώνων και στη συνέχεια ενεργεί σε μια μετασυναπτική διαδικασία. Έτσι, η σύναψη μετατρέπει ένα προσυναπτικό ηλεκτρικό σήμα σε χημικό σήμα και στη συνέχεια πίσω σε ένα μετασυναπτικό ηλεκτρικό σήμα.

Όσον αφορά το νευρωνικό δίκτυο, ένας νευρώνας (*Neuron*) είναι μια μονάδα επεξεργασίας πληροφοριών που είναι θεμελιώδους σημασίας για τη λειτουργία αυτού[8].

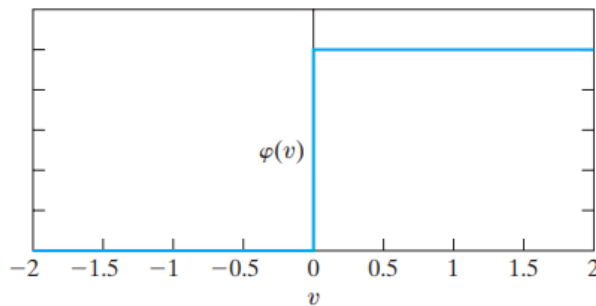


Εικόνα 6 - Δομή Νευρώνα[8]

Στο παραπάνω διάγραμμα μπορούμε να διακρίνουμε τα τρία βασικά δομικά συστατικά ενός νευρωνικού δικτύου.

- Ένα σύνολο συνάψεων ή συνδετικών συνδέσεων, κάθε μία από τις οποίες χαρακτηρίζεται από το βάρος ή τη δύναμή του. Συγκεκριμένα, ένα σήμα x_j στην είσοδο της συνάψεως j που συνδέεται με το νευρώνα k πολλαπλασιάζεται με το συναπτικό βάρος w_{kj} .
- Ένας αθροιστής των σημάτων εισόδου, τα οποία είναι σταθμισμένα με τις αντίστοιχες συναπτικές δυνάμεις του νευρώνα.
- Μια συνάρτηση απαρίθωσης (integration function) ή συνάρτηση ενεργοποίησης (activation function) για τον περιορισμό του εύρους της εξόδου ενός νευρώνα. Η συνάρτηση ενεργοποίησης αναφέρεται επίσης ως συνάρτηση περιορισμού, επειδή περιορίζει το επιτρεπτό εύρος του σήματος εξόδου σε κάποια πεπερασμένη τιμή. Πέραν των δύο βασικών τύπων συναρτήσεων (Συνάρτηση Κατωφλίου και Σιγμοειδής Συνάρτηση), σε προβλήματα που δεν μπορούν να αναπαρασταθούν εύκολα από γραμμικές συναρτήσεις, χρησιμοποιείται ως συνάρτηση ενεργοποίησης ένα υβρίδιο ανάμεσα στην υπερβολική εφασπτομένη και την γραμμική συνάρτηση [8][9][14]. Πιο συγκεκριμένα:
 - Συνάρτηση Κατωφλίου, η οποία περιγράφεται από τον ακόλουθο τύπο:

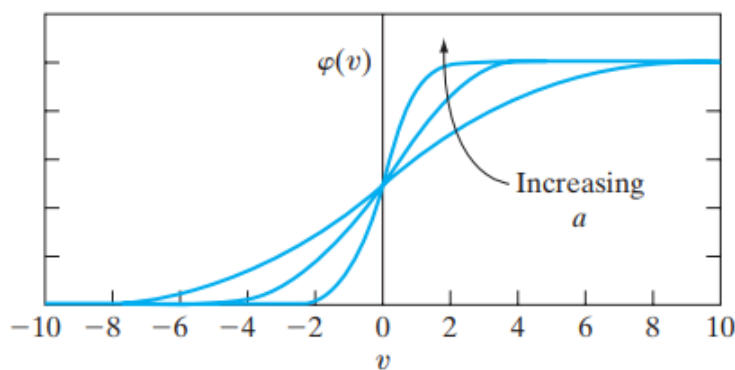
$$\varphi(v) = \begin{cases} 1 & \text{εάν } v \geq 0 \\ 0 & \text{εάν } v < 0 \end{cases}$$



Εικόνα 7 - Συνάρτηση Κατωφλίου[8]

- Σιγμοειδής Συνάρτηση, η οποία είναι αυστηρά αύξουσα, αποτελεί την πιο κοινή συνάρτηση ενεργοποίησης στην κατασκευή των νευρωνικών δικτύων. Μία μορφή της συνάρτησης αυτής είναι η λογιστική συνάρτηση και δίνεται από τη σχέση:

$$\varphi(v) = \frac{1}{1 + \exp(-av)}$$



Εικόνα 8 - Σιγμοειδής Συνάρτηση Ενεργοποίησης[8]

- iii. Τμηματικά γραμμική συνάρτηση (Piecewise-linear function) η οποία είναι ένα υβρίδιο της υπερβολικής εφαπτομένης (\tanh) και της Ανορθωτικής Γραμμικής Συνάρτησης (Rectified Linear Unit - ReLU) [9]. Η υπερβολική εφαπτομένη που αποτελεί μία ιδανική επιλογή ως συνάρτηση ενεργοποίησης, εφόσον το νευρωνικό δίκτυο δεν έχει μεγαλώσει σε βάθος και δίνεται από την παρακάτω σχέση:

$$\tanh x = \frac{\sinh x}{\cosh x} = \frac{e^x - e^{-x}}{e^x + e^{-x}} = \frac{e^{2x} - 1}{e^{2x} + 1}.$$

Στα βαθιά νευρωνικά δίκτυα (Deep Neural Networks), προτιμώνται οι γραμμικές μονάδες ανόρθωσης – ReLU που έχουν ως συνάρτηση ενεργοποίησης την:

$$f(z) = \max(0, z)$$

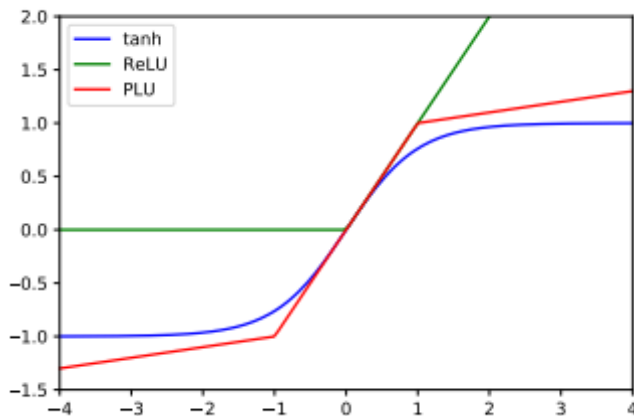
Οι ReLU μπορούν να βελτιστοποιηθούν γιατί είναι παρόμοιες με τις γραμμικές μονάδες, με τη διαφορά ότι οι ReLU δίνουν μηδέν για οποιαδήποτε μη θετική τιμή. Αυτό κάνει τις μερικές παραγώγους τους να παραμένουν μεγάλες όταν η μονάδα είναι ενεργή. Οι μερικές παράγωγοι δεν είναι απλά μεγάλες, είναι και συνεχείς. Έτσι, η δεύτερη παράγωγος της συνάρτησης είναι 0 σχεδόν παντού και η πρώτη παράγωγος είναι 1 όπου η μονάδα είναι ενεργή. Αυτό σημαίνει ότι η κατεύθυνση της κλίσης (gradient) είναι πολύ πιο χρήσιμη για τη μάθηση απ' ό,τι θα ήταν με οποιαδήποτε συνάρτηση ενεργοποίησης εισήγαγε δευτέρας-τάξης επιδράσεις. Οι ReLU περιγράφονται συνολικά από την ακόλουθη σχέση [12],[13]

$$\mathbf{h} = f(\mathbf{W}\mathbf{T}\mathbf{x} + \mathbf{b})$$

Όταν μια γραμμική συνάρτηση $h(x)$ μετασχηματίζεται από την υπερβολική εφαπτομένη, δηλ. $g(x) = \tanh(h(x))$, η προκύπτουσα συνάρτηση $g(x)$ είναι μη γραμμική και ομαλή. Όταν το ReLU εφαρμόζεται επίσης στο $h(x)$, το αποτέλεσμα είναι μια τμηματικά γραμμική συνάρτηση με παράγωγο είτε 0 ή ∇h . Η προσέγγιση μιας ομαλής, εξαιρετικά μη γραμμικής συνάρτησης χρησιμοποιώντας ένα μοντέλο δικτύου απαιτεί πολλά ReLU, υπονοώντας ένα δίκτυο με πολλά επίπεδα. Για να αυξήσουμε τη μη γραμμικότητα που δημιουργείται από κάθε ενεργοποίηση, προτείνεται η τμηματικά γραμμική συνάρτηση ενεργοποίησης PLU:

$$PLU(x) \equiv \max(\alpha(x+c) - c, \min(\alpha(x-c) + c, x))$$

Το PLU είναι μια ακατέργαστη προσέγγιση της \tanh και είναι μη φραγμένη για να αποφύγει την εξάλειψη των κλίσεων. Η ReLU συνάρτηση αποτελείται από δύο γραμμικά τμήματα με κλίση 0 ή 1. Το PLU έχει τρία γραμμικά τμήματα με κλίσεις α ή 1, όπου το α είναι μια παράμετρος που πρέπει να επιλεγεί ή να εκπαιδευτεί. Όπως φαίνεται και στο παρακάτω σχήμα, η PLU μπορεί να προσεγγίζει μη γραμμικές συναρτήσεις πολύ καλύτερα από την ReLU [9],[10],[11].



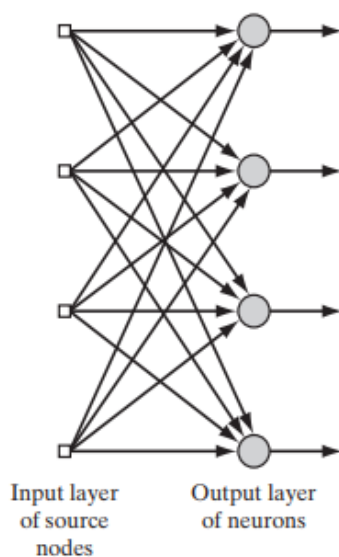
Εικόνα 9 - Αναπαράσταση των συναρτήσεων tanh, ReLU και PLU[10]

Αρχιτεκτονικές Νευρωνικών Δικτύων

Ανάλογα με το είδος των συνδέσεων των νευρώνων που απαρτίζουν ένα νευρωνικό δίκτυο, διακρίνονται διαφορετικές αρχιτεκτονικές από τις οποίες οι πιο βασικές είναι οι παρακάτω[8]:

1. Ενός Επιπέδου Δίκτυα Πρόσθιας Τροφοδότησης (Single-Layer Feedforward Networks)

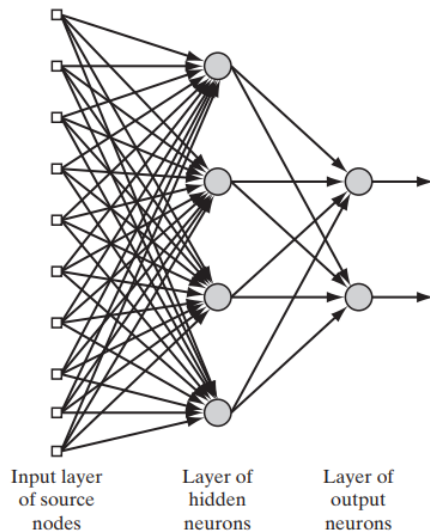
Σε ένα νευρωνικό δίκτυο, οι νευρώνες οργανώνονται με τη μορφή επιπέδων. Στην απλούστερη μορφή του, έχουμε ένα επίπεδο εισόδου αποτελούμενο από κόμβους εισόδου που συνδέεται απευθείας σε ένα επίπεδο νευρώνων εξόδου, αλλά όχι αντίστροφα. Δηλαδή, το δίκτυο είναι αυστηρώς τύπου πρόσθιας τροφοδότησης.



Εικόνα 10 - Δίκτυο Πρόσθιας Τροφοδότησης ενός Επιπέδου[8]

2. Πολυεπίπεδα Δίκτυα Πρόσθιας Τροφοδότησης (Multilayer Feedforward Networks)

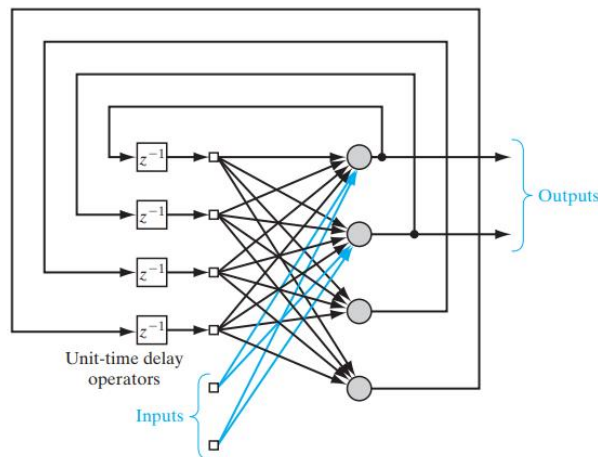
Τα δίκτυα που ανήκουν σε αυτή την κατηγορία αποτελούνται από ένα ή περισσότερα κρυφά επίπεδα, των οποίων οι υπολογιστικοί κόμβοι ονομάζονται κρυφοί νευρώνες ή κρυφές μονάδες. Ο όρος "κρυφός" αναφέρεται στο γεγονός ότι αυτό το τμήμα του νευρωνικού δικτύου δεν είναι άμεσα ορατό ούτε από την είσοδο ούτε από την έξοδο του δικτύου. Με την προσθήκη ενός ή περισσότερων κρυφών επιπέδων, το δίκτυο είναι σε θέση να εξαγει στατιστικά υψηλότερης τάξης από την είσοδό του. Με μια ευρύτερη έννοια, το δίκτυο αποκτά μια γενική προοπτική λόγω των επιπλέον συναπτικών συνδέσεων. Οι κόμβοι στο επίπεδο εισόδου του δικτύου τροφοδοτούν τα αντίστοιχα στοιχεία του πρότυπου ενεργοποίησης (διάνυσμα εισόδου), που αποτελούν τα σήματα εισόδου που εφαρμόζονται στους νευρώνες (υπολογιστικοί κόμβοι) του δεύτερου επιπέδου (δηλ. του πρώτου κρυφού επιπέδου). Τυπικά, οι νευρώνες σε κάθε επίπεδο του δικτύου έχουν ως εισόδους τους τα σήματα εξόδου μόνο του προηγούμενου επιπέδου. Το σύνολο των σημάτων εξόδου των νευρώνων στο επίπεδο εξόδου (τελικό επίπεδο) του δικτύου αποτελεί τη συνολική απόκριση του δικτύου στο πρότυπο ενεργοποίησης που παρέχεται από τους κόμβους στο επίπεδο εισόδου (πρώτο επίπεδο του δικτύου). Ένα χαρακτηριστικό του παραπάνω δικτύου είναι το γεγονός ότι κάθε κόμβος σε κάθε επίπεδο συνδέεται με κάθε άλλο κόμβο του επόμενου επιπέδου. Το δίκτυο αυτό καλείται πλήρως συνδεδεμένο. Σε αντίθετη περίπτωση (απουσία ορισμένων συνδέσεων) λέμε ότι το δίκτυο είναι μερικώς συνδεδεμένο.



Εικόνα 11 - Πολυεπίπεδο Δίκτυο Πρόσθιας Τροφοδότησης[8]

3. Αναδρομικά Δίκτυα (Recurrent Networks)

Ένα αναδρομικό νευρωνικό δίκτυο διαθέτει τουλάχιστον ένα βρόγχο ανάδρασης σε σχέση με το δίκτυο πρόσθιας τροφοδότησης. Αυτό σημαίνει ότι ένα τέτοιο δίκτυο μπορεί να αποτελείται από ένα μεμονωμένο επίπεδο νευρώνων, όπου κάθε νευρώνας τροφοδοτεί το σήμα εξόδου του πίσω, στις εισόδους των άλλων νευρώνων. Η ύπαρξη βρόγχων ανάδρασης έχει μεγάλη επίδραση στην ικανότητα μάθησης και την απόδοση του δικτύου.



Εικόνα 12 - Αναδρομικό Δίκτυο[8]

Εκπαίδευση Νευρωνικών Δικτύων

Όπως αναφέρθηκε στην αρχή του κεφαλαίου, το νευρωνικό δίκτυο έχει την τάση να αποθηκεύει και να επαναχρησιμοποιεί βιωματική γνώση[8]. Μπορούμε σύντομα να ορίσουμε την έννοια της γνώσης ως το σύνολο των αποθηκευμένων πληροφοριών ή μοντέλων που χρησιμοποιούνται από ένα άτομο ή μια μηχανή για να ερμηνεύουν, να προβλέπουν και να ανταποκρίνονται κατάλληλα στον έξω κόσμο. Μία βασική λειτουργία ενός νευρωνικού δικτύου είναι να μάθει ένα μοντέλο του περιβάλλοντος στο οποίο είναι ενσωματωμένο και να διατηρεί το μοντέλο επαρκώς συνεπές με τον πραγματικό κόσμο, έτσι ώστε να επιτευχθούν οι καθορισμένοι στόχοι της εφαρμογής. Η γνώση του περιβάλλοντος αποτελείται από δύο είδη πληροφοριών:

1. Την ισχύουσα κατάσταση, που αντιπροσωπεύεται από γεγονότα για το τι είναι και τι ήταν γνωστό και καλείται προγενέστερη γνώση/πληροφορία.

2. Τις παρατηρήσεις του περιβάλλοντος, που λαμβάνονται με τη βοήθεια αισθητήρων, σχεδιασμένων να διερευνούν το περιβάλλον στο οποίο υποτίθεται ότι λειτουργεί το νευρωνικό δίκτυο. Οι παρατηρήσεις αυτές, αν και αποτελούν το σύνολο των πληροφοριών που χρησιμοποιούνται για την εκπαίδευση του νευρωνικού δικτύου, υπόκεινται σε σφάλματα λόγω της ύπαρξης θορύβου στους αισθητήρες.

Αν και ο τρόπος αναπαράστασης της γνώσης είναι αρκετά περίπλοκος, υπάρχουν τέσσερις γενικοί κανόνες που εφαρμόζονται.

1. Οι όμοιες είσοδοι που προέρχονται από όμοιες κλάσεις συνήθως παράγουν όμοιες αναπαραστάσεις στο εσωτερικό του δικτύου και, συνεπώς, πρέπει να ταξινομούνται ως μέλη της ίδιας κατηγορίας.
2. Για τα στοιχεία που πρόκειται να ταξινομηθούν ως χωριστές κλάσεις, θα πρέπει να δίνονται επαρκώς διαφορετικές αναπαραστάσεις στο δίκτυο.
3. Εάν ένα συγκεκριμένο χαρακτηριστικό είναι σημαντικό, τότε θα πρέπει να χρησιμοποιηθεί ένας μεγάλος αριθμός νευρώνων για την αναπαράστασή του στο δίκτυο.
4. Προηγούμενες πληροφορίες και ομοιότητες πρέπει να ενσωματωθούν στο σχεδιασμό ενός νευρωνικού δικτύου όποτε είναι διαθέσιμες, έτσι ώστε να απλοποιηθεί η σχεδίαση του δικτύου εφόσον δεν είναι υποχρεωμένο να τα μαθαίνει.

Η πιο αξιοσημείωτη ιδιότητα των νευρωνικών δικτύων είναι η δυνατότητα μάθησης, η οποία μπορεί να κατηγοριοποιηθεί ως εξής:

- Επιβλεπόμενη μάθηση, η οποία απαιτεί τη διαθεσιμότητα ενός στόχου ή μιας επιθυμητής απόκρισης για την υλοποίηση μιας συγκεκριμένης αντιστοίχισης εισόδου-εξόδου, ελαχιστοποιώντας μια συνάρτηση κόστους
- Μη επιβλεπόμενη μάθηση, η υλοποίηση της οποίας βασίζεται στην παροχή ενός «ανεξάρτητου από την εργασία» μέτρου της ποιότητας της αναπαράστασης που απαιτείται να μάθει το δίκτυο
- Ενισχυτική μάθηση, στην οποία η αντιστοίχιση εισόδου-εξόδου εκτελείται μέσω της συνεχούς αλληλεπίδρασης ενός συστήματος μάθησης με το περιβάλλον του, ώστε να ελαχιστοποιείται ένας βαθμωτός δείκτης απόδοσης

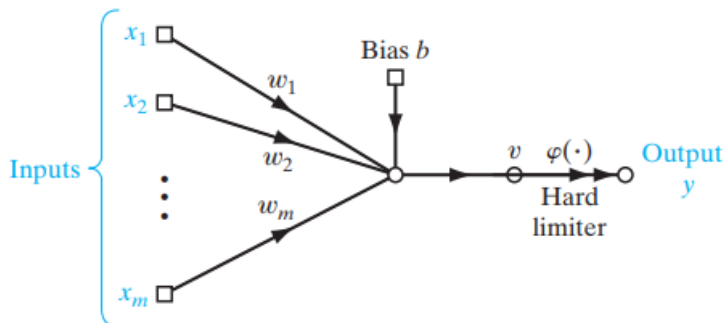
Η επιβλεπόμενη μάθηση βασίζεται στη διαθεσιμότητα ενός δείγματος εκπαίδευσης χαρακτηρισμένων παραδειγμάτων, με κάθε παράδειγμα να αποτελείται από ένα σήμα εισόδου και την αντίστοιχη επιθυμητή απόκριση[8][14]. Στην πράξη, η συλλογή των παραδειγμάτων αυτών είναι μια χρονοβόρα διαδικασία, κυρίως για μεγάλης κλίμακας προβλήματα μάθησης. Αντίθετα, η μη επιβλεπόμενη μάθηση βασίζεται αποκλειστικά σε μη χαρακτηρισμένα παραδείγματα, τα οποία απολούνται απλώς από ένα σύνολο σημάτων εισόδου.

Η ενισχυτική μάθηση βρίσκεται κάπου ανάμεσα στην επιβλεπόμενη και τη μη επιβλεπόμενη μάθηση. Λειτουργεί μέσω συνεχών αλληλεπιδράσεων μεταξύ ενός συστήματος μάθησης και του περιβάλλοντος. Το σύστημα μάθησης εκτελεί μια ενέργεια και μαθαίνει από την απόκριση που του παρέχει το περιβάλλον ως προς αυτή την ενέργεια. Στην πράξη, ο ρόλος του εκπαιδευτή στην επιβλεπόμενη μάθηση αντικαθίσταται από αυτόν του κριτή, ο οποίος μπορεί να είναι ενσωματωμένος στο μηχανισμό μάθησης[8].

3.2 Perceptron ενός Επιπέδου

Γενικά

Το perceptron είναι η απλούστερη μορφή νευρωνικού δικτύου που μπορεί να σχεδιαστεί και να μελετηθεί, καθώς αποτελείται από ένα και μοναδικό νευρώνα[8]. Χρησιμοποιείται για την ταξινόμηση προτύπων, τα οποία είναι γραμμικά διαχωρίσιμα. Στην ουσία, αποτελείται από ένα μεμονωμένο νευρώνα με προσαρμόσιμα συναπτικά βάρη και μεροληψία (bias).



Εικόνα 13 - Perceptron Rosenblatt

Το Perceptron του Rosenblatt βασίζεται στο μοντέλο ενός νευρώνα των McCulloch/Pitts. Οι μόνες συνδέσεις που υπάρχουν είναι αυτές μεταξύ των εισόδων και του νευρώνα. Κάθε σύνδεση του νευρώνα με το εισερχόμενο σήμα x_i συνδέεται με το αντίστοιχο βάρος w_i . Οι εξισώσεις που περιγράφουν το παραπάνω μοντέλο είναι οι εξής:

$$u = \sum_{i=1}^n x_i w_i - \theta$$

Η έξοδος του νευρώνα είναι η $y=f(u)$ που μπορεί να γραφεί και ως:

$$y = f\left(\sum_{i=1}^n x_i w_i - \theta\right)$$

Στον παραπάνω τύπο συμβολίζουμε με u την διέγερση του νευρώνα, δηλαδή το συνολικό σήμα που λαμβάνει. Η συνάρτηση ενεργοποίησης f απεικονίζει το διάνυσμα εισόδου $x=[x_1, x_2, \dots, x_n]^T$ στην έξοδο y . Η παράμετρος θ ονομάζεται κατώφλι ενεργοποίησης και είναι το κατώφλι πάνω από το οποίο ενεργοποιείται ο νευρώνας. Διακρίνουμε επομένως τρεις καταστάσεις στις οποίες μπορεί να βρεθεί η διέγερση:

$$u > 0 \text{ εάν } \sum_{i=1}^n x_i w_i > \theta$$

$$u = 0 \text{ εάν } \sum_{i=1}^n x_i w_i = \theta$$

$$u < 0 \text{ εάν } \sum_{i=1}^n x_i w_i < \theta$$

Η συνάρτηση ενεργοποίησης f τροφοδοτείται από τη διέγερση u και δίνει την έξοδο του νευρώνα. Είναι μη γραμμική και μπορεί να έχει μία από τις παρακάτω μορφές:

$$f(u) = \begin{cases} 1, & \text{αν } u > 0 \\ 0, & \text{αν } u \leq 0 \end{cases}$$

Ή

$$f(u) = \begin{cases} 1, & \text{αν } u > 0 \\ -1, & \text{αν } u \leq 0 \end{cases}$$

Η ενεργητικότητα του αισθητήρα εξαρτάται από τα βάρη των συνδέσεων, τις τιμές των εισόδων και την τιμή του κατωφλίου. Αρκετά συχνά, το κατώφλι θ θεωρείται ως επιπλέον εσωτερικό βάρος $w_0=\theta$, ξεχωριστό από τα υπόλοιπα, και αντιστοιχεί στην είσοδο $x_0=-1$, με την αρχική εξίσωση να μετασχηματίζεται στην παρακάτω:

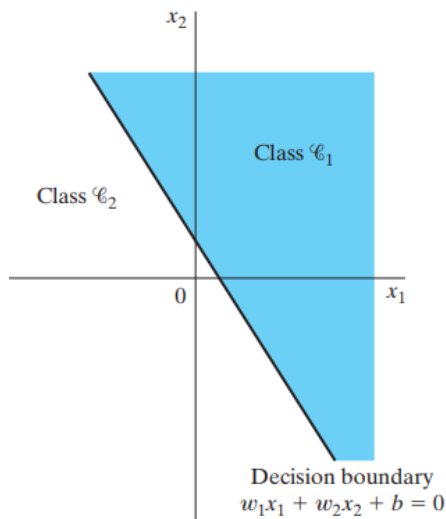
$$u = \sum_{i=1}^n x_i w_i - \theta = \sum_{i=1}^n x_i w_i - x_0 w_0 = \sum_{i=1}^n x_i w_i$$

Στην τελευταία, αν προσθέσουμε μια είσοδο $x_0=-1$, τότε το βάρος $w_0=-\theta$ αντιστοιχίζεται στη σταθερή είσοδο $x_0=1$ και ονομάζεται πόλωση (b). Ο όρος αυτός δεν έχει καμία φυσική σημασία και θεωρείται ως επιπρόσθετο ερέθισμα που αθροίζεται με τα σήματα εισόδου.

Όπως αναφέρθηκε, το perceptron χρησιμοποιείται για την ταξινόμηση προτύπων και για αυτό θα πρέπει να μπορεί να επιτυγχάνει γραμμικό διαχωρισμό. Σύμφωνα με τον Rosenblatt, αν τα διανύσματα εκπαίδευσης προέρχονται από δύο γραμμικά διαχωρίσιμες κλάσεις, τότε ο αλγόριθμος του perceptron συγκλίνει σε πεπερασμένο αριθμό επαναλήψεων. Στην απλούστερη δυνατή μορφή του perceptron, υπάρχουν δύο περιοχές απόφασης διαχωριζόμενες από υπερεπίπεδο το οποίο ορίζεται από την:

$$\sum_{i=1}^n x_i w_i + b = 0$$

Για παράδειγμα, στο παρακάτω σχήμα το όριο της απόφασης παίρνει τη μορφή ευθείας γραμμής. Ένα σημείο το οποίο βρίσκεται πάνω από τη διαχωριστική γραμμή αντιστοιχίζεται στην κλάση 1, ενώ ένα σημείο που βρίσκεται κάτω από αυτή αντιστοιχίζεται στην κλάση 2. Η επίδραση της πόλωσης b μετατοπίζει το όριο απόφασης μακριά από την αρχή των αξόνων.



Εικόνα 14 - Το Υπερεπίπεδο ως Όριο Απόφασης[8]

Εκπαίδευση Perceptron – Αλγόριθμος Σύγκλισης

Για να λειτουργήσει σωστά το perceptron, πρέπει οι δύο κλάσεις να είναι γραμμικά διαχωρίσιμες, αρά, και τα προς ταξινόμηση πρότυπα θα πρέπει να είναι επαρκώς διαχωρισμένα το ένα από το άλλο έτσι ώστε να εξασφαλίσουμε ότι η επιφάνεια απόφασης θα είναι υπερεπίπεδο. Άρα, στόχος της εκπαίδευσης του δικτύου είναι η αυτόματη εκμάθηση των παραμέτρων, ώστε να βρίσκει τη διαχωριστική γραμμή ανάμεσα στις δύο κλάσεις. Δηλαδή, θα πρέπει να υπολογίζεται ένα διάνυσμα βαρών τέτοιο ώστε:

$w^T x > 0$ για κάθε διάνυσμα εισόδων x που ανήκει στην κλάση 1

$w^T x \leq 0$ για κάθε διάνυσμα εισόδων x που ανήκει στην κλάση 2

Η εκπαίδευση γίνεται υπό επιτήρηση, δηλαδή δίνεται μια επιθυμητή έξοδος d για κάθε πρότυπο x χρησιμοποιώντας έναν επαναληπτικό αλγόριθμο. Τα πρότυπα παρουσιάζονται κυκλικά και όταν τελειώσουν επαναλαμβάνονται από την αρχή. Η διαδικασία αυτή μεταβάλλει το διάνυσμα βαρών εφόσον υπάρχει σφάλμα ταξινόμησης, δηλαδή όταν η επιθυμητή απόκριση γι' αυτό το πρότυπο διαφέρει από αυτή του δικτύου. Στους παρακάτω πίνακες παρουσιάζονται οι μεταβλητές και τα συνοπτικά βήματα του αλγόριθμου σύγκλισης ενός Perceptron με n εισόδους αντίστοιχα.

Μεταβλητές και Παράμετροι		
$x(n)$	Διάνυσμα εισόδων	$[+1, x_1(n), x_2(n), \dots, x_m(n)]^T$

$w(n)$	Διάνυσμα βαρών	$[b, w_1(n), w_2(n), \dots, w_m(n)]^T$
b	Πόλωση	
$y(n)$	Πραγματική απόκριση	
$d(n)$	Επιθυμητή απόκριση	
η	Παράμετρος ρυθμού μάθησης, <1	

Πίνακας 1 - Μεταβλητές Αλγόριθμου Σύγκλισης

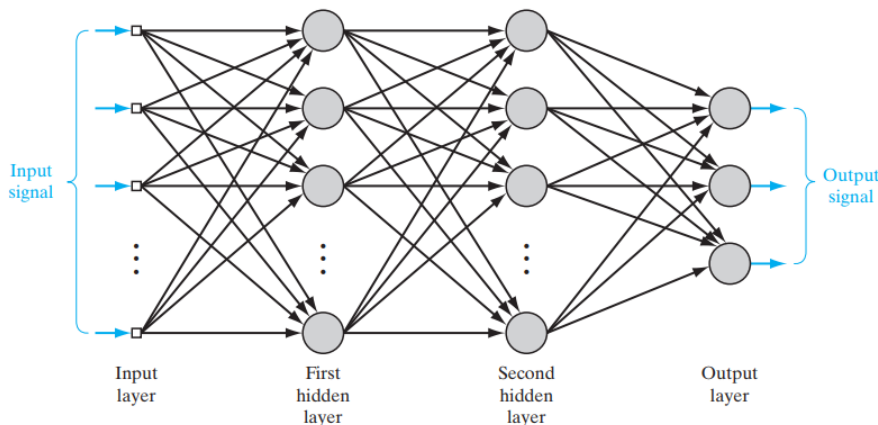
Βήματα Αλγόριθμου Σύγκλισης		
Βήμα 1	Αρχικοποίηση	Θέσε $w(0)=0$. Εκτέλεσε τους ακόλουθους υπολογισμούς για χρονικό βήμα $n=1,2,\dots$
Βήμα 2	Ενεργοποίηση	Στο χρονικό βήμα n , ενεργοποίησε το perceptron εφαρμόζοντας το διάνυσμα εισόδων $x(n)$ και την επιθυμητή απόκριση $d(n)$
Βήμα 3	Υπολογισμός Πραγματικής Απόκρισης	Υπολόγισε την πραγματική απόκριση ως $y(n)=\text{sgn}[w^T(n)x(n)]$
Βήμα 4	Προσαρμογή Διανύσματος Βαρών	Ενημέρωσε το διάνυσμα βαρών ώστε $w(n+1)=w(n)+\eta[d(n)-y(n)]x(n)$ όπου $d(n) = \begin{cases} +1 & \text{εάν } X(n) \text{ ανήκει στην κλάση 1} \\ -1 & \text{εάν } X(n) \text{ ανήκει στην κλάση 2} \end{cases}$
Βήμα 5	Συνέχιση	Αύξησε το χρονικό βήμα κατά 1 και επέστρεψε στο βήμα 2

Πίνακας 2 - Συνοπτική Παρουσίαση Αλγόριθμου Σύγκλισης

3.3 Perceptron Πολλαπλών Επιπέδων

Ένα νευρωνικό δίκτυο με ένα ή περισσότερα κρυφά επίπεδα (multilayer perceptron) εμφανίζει τα παρακάτω βασικά χαρακτηριστικά:

- Το μοντέλο κάθε νευρώνα στο δίκτυο περιλαμβάνει μια μη γραμμική συνάρτηση ενεργοποίησης, η οποία είναι διαφορίσιμη
- Το δίκτυο περιέχει ένα ή περισσότερα επίπεδα, τα οποία παραμένουν κρυφά για τους κόμβους των επιπέδων εισόδου και εξόδου
- Το δίκτυο επιδεικνύει μεγάλη διασυνδεσιμότητα, ο βαθμός της οποίας καθορίζεται από τα συναπτικά βάρη του δικτύου.



Εικόνα 15 - Νευρωνικό Δίκτυο με Δύο Κρυφά Επίπεδα[8]

Οι νευρώνες εξόδου απαρτίζουν το επίπεδο εξόδου, ενώ οι υπόλοιποι νευρώνες, οι οποίοι δεν αποτελούν μέρος ούτε της εισόδου ούτε της εξόδου, απαρτίζουν τα κρυφά επίπεδα του δικτύου. Το πρώτο κρυφό επίπεδο τροφοδοτείται από το επίπεδο εισόδου του δικτύου και οι έξοδοι που παράγονται από το επίπεδο αυτό εφαρμόζονται στο επόμενο κρυφό επίπεδο. Η διαδικασία αυτή συνεχίζεται για όλα τα επίπεδα του δικτύου[8].

Ο σχεδιασμός των νευρώνων είναι τέτοιος, ώστε να εκτελεί τόσο τον υπολογισμό του λειτουργικού σήματος, που εκφράζεται ως μια συνεχής μη γραμμική συνάρτηση του σήματος εισόδου και των συναπτικών βαρών, όσο και τον υπολογισμό του διανύσματος κλίσης, που αφορά την κλίση της επιφάνειας σφάλματος με τα βάρη που συνδέονται στις εισόδους ενός νευρώνα.

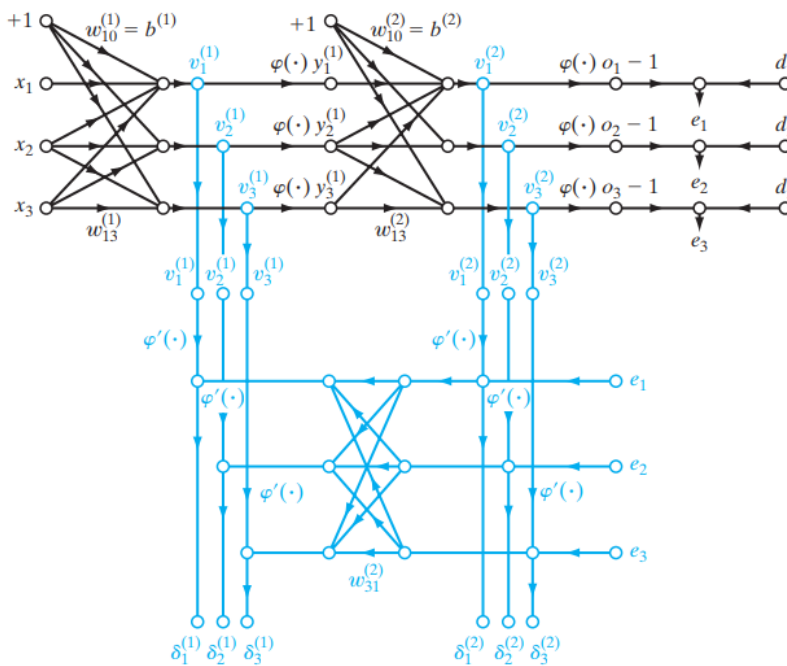
Ο αλγόριθμος οπίσθιας διάδοσης της ενεργοποίησης/σφάλματος (Back Propagation Algorithm) έχει καθιερωθεί ως ένας υπολογιστικά αποτελεσματικός και χρήσιμος αλγόριθμος για την εκπαίδευση των perceptron πολλαπλών επιπέδων[8][14]. Οφείλει το όνομά του στο γεγονός ότι μερικές παράγωγοι της συνάρτησης κόστους αναφορικά με τις ελεύθερες παραμέτρους του δικτύου, καθορίζονται μέσω της οπισθοδιάδοσης των σημάτων σφάλματος διαμέσου του δικτύου, επίπεδο προς επίπεδο. Η υπολογιστική ισχύς του αλγορίθμου έγκειται στα δύο κύρια γνωρίσματά του:

- Την τοπική μέθοδο για την ενημέρωση των συναπτικών βαρών και πολώσεων του perceptron πολλαπλών επιπέδων
- Την αποτελεσματική μέθοδο για τον υπολογισμό όλων των μερικών παραγώγων της συνάρτησης κόστους αναφορικά με τις ελεύθερες παραμέτρους

Η μέθοδος αυτή για την εκπαίδευση του νευρωνικού δικτύου λαμβάνει χώρα σε δυο φάσεις[8]. Αρχικά, στη φάση που εξελίσσεται προς τα εμπρός, τα συναπτικά βάρη του δικτύου είναι σταθερά και το σήμα εισόδου διαδίδεται διαμέσου του δικτύου μέχρι να φτάσει στην έξοδο. Το σήμα εισόδου καλείται και λειτουργικό σήμα καθώς εκτελεί μια χρήσιμη λειτουργία στην έξοδο του δικτύου και υπολογίζεται ως συνάρτηση των εισόδων και των σχετιζόμενων βαρών που εφαρμόζονται στο εκάστοτε νευρώνα. Αντίθετα, στη φάση που εξελίσσεται προς τα πίσω, παράγεται ένα σήμα σφάλματος μέσω της σύγκρισης της εξόδου του δικτύου με μια επιθυμητή απόκριση. Το σήμα σφάλματος – ο υπολογισμός του απαιτεί μία εξαρτώμενη από το σφάλμα συνάρτηση – που προκύπτει διαδίδεται επίσης διαμέσου του δικτύου αλλά με κατεύθυνση προς τα πίσω, ενώ λαμβάνουν χώρα και διαδοχικές προσαρμογές των συναπτικών βαρών.

Ο αλγόριθμος διατρέχει το δείγμα εκπαίδευσης ως εξής:

1. Αρχικοποίηση. Υποθέτοντας ότι δεν είναι διαθέσιμη πρότερη πληροφορία, επέλεξε τα συναπτικά βάρη και τα κατώφλια από μια ομοιόμορφη κατανομή, της οποίας ο μέσος είναι μηδέν και της οποίας η διακύμανση επιλέγεται με τέτοιο τρόπο, ώστε να κάνει την τυπική απόκλιση των τοπικών πεδίων των νευρώνων να βρίσκεται στο όριο μεταξύ του γραμμικού και του σταθερού μέρους της σιγμοειδούς συνάρτησης ενεργοποίησης.
2. Παρουσιάσεις των παραδειγμάτων εκπαίδευσης. Παρουσίασε στο δίκτυο μια εποχή παραδειγμάτων εκπαίδευσης. Για κάθε παράδειγμα που περιλαμβάνει το δείγμα, εκτέλεσε την ακολουθία των προς τα εμπρός και προς τα πίσω υπολογισμών που περιγράφεται στα βήματα 3 και 4 αντίστοιχα.
3. Υπολογισμός προς τα εμπρός. Έστω ότι ένα παράδειγμα εκπαίδευσης στη συγκεκριμένη εποχή (epoch) συμβολίζεται ως $(x(n), d(n))$, με το διάνυσμα εισόδων $x(n)$ να εφαρμόζεται στο επίπεδο εισόδου των αισθητηριακών κόμβων και το διάνυσμα επιθυμητών αποκρίσεων $d(n)$ να παρουσιάζεται στο επίπεδο εξόδου των υπολογιστικών κόμβων. Υπολόγισε τα τοπικά πεδία και τα λειτουργικά σήματα του δικτύου προχωρώντας προς τα εμπρός, σε όλη την έκταση του δικτύου, επίπεδο προς επίπεδο.
4. Υπολογισμός προς τα πίσω.
5. Επανάληψη. Επανάλαβε τους υπολογισμούς των βημάτων 3 και 4 παρουσιάζοντας νέες εποχές παραδειγμάτων εκπαίδευσης στο δίκτυο, μέχρι να ικανοποιηθεί το επιλεγόμενο κριτήριο τερματισμού.



Εικόνα 16 - Διαγραμματική Απεικόνιση Εκμάθησης με τον Αλγόριθμο BackPropagation[8]

Αν και η σχεδίαση ενός νευρωνικού δικτύου με τη χρήση του αλγόριθμου οπίσθιας διάδοσης της ενεργοποίησης/σφάλματος έγκειται στην εμπειρία του κάθε σχεδιαστή, υπάρχουν μέθοδοι που μπορούν να βελτιώσουν σημαντικά την απόδοσή του.

1. Στοχαστική έναντι μαζικής ενημέρωσης: Η στοχαστική διαδικασία μάθησης με οπίσθοδιάδοση σφάλματος είναι υπολογιστικά γρηγορότερη από τη διαδικασία μαζικής μάθησης, ιδιαίτερα όταν το δείγμα δεδομένων εκπαίδευσης είναι μεγάλο.
2. Μεγιστοποίηση του περιεχομένου της πληροφορίας: Σ' αυτή την περίπτωση θα πρέπει να χρησιμοποιηθούν παραδείγματα που θα παρουσιάζουν το μεγαλύτερο σφάλμα εκπαίδευσης ή θα είναι τελείως διαφορετικά από αυτά που έχουν χρησιμοποιηθεί μέχρι τώρα. Επιπλέον, σε διαδικασίες ταξινόμησης προτύπων εφαρμόζεται η τεχνική της τυχαιοποίησης, που εξασφαλίζει ότι τα διαδοχικά παραδείγματα δεν θα ανήκουν στην ίδια κλάση.
3. Συνάρτηση ενεργοποίησης: Από θέμα ταχύτητας μάθησης, προτιμούμε τη χρήση μιας σιγμοειδούς συνάρτησης, η οποία θα είναι περιττή συνάρτηση του ορίσματος της

$$\varphi(-u) = -\varphi(u)$$

4. Τιμές στόχου: Είναι σημαντικό οι επιθυμητές τιμές απόκρισης να επιλέγονται μέσα από το εύρος τιμών της σιγμοειδούς συνάρτησης ενεργοποίησης, αλλιώς ο αλγόριθμος τείνει να οδηγεί τις ελεύθερες παραμέτρους στο άπειρο επιβραδύνοντας τη διαδικασία μάθησης.
5. Κανονικοποίηση των εισόδων: Κάθε μεταβλητή εισόδου θα πρέπει να περνά από μια διαδικασία προεπεξεργασίας, έτσι ώστε η μέση τιμή της, υπολογισμένη για όλο το δείγμα εκπαίδευσης, να είναι κοντά στο μηδέν. Σε αντίθετη περίπτωση, θα είναι μικρή σε σύγκριση με την τυπική απόκλιση. Η διαδικασία της κανονικοποίησης περιλαμβάνει τα στάδια της απομάκρυνσης μέσων τιμών, της αποσυσχέτισης και της εξίσωσης συνδιακυμάνσεων.
6. Αρχικοποίηση: Όταν στα συναπτικά βάρη ανατίθενται μεγάλες αρχικές τιμές, είναι πιθανό οι νευρώνες να οδηγηθούν στον κορεσμό και η διαδικασία μάθησης να επιβραδυνθεί. Αν, όμως, ανατεθούν μικρές τιμές, ο αλγόριθμος ΒΚ μπορεί να λειτουργεί σε μια πολύ επίπεδη περιοχή γύρω από το σημείο αρχής της επιφάνειας σφάλματος.

- Επομένως, η αρχικοποίηση των συναπτικών βαρών θα πρέπει να γίνεται με τέτοιο τρόπο, ώστε η τυπική απόκλιση του τοπικού πεδίου ενός νευρώνα να βρίσκεται στο όριο μεταξύ των μερών γραμμικής και κορεσμένης λειτουργίας της σιγμοειδούς συνάρτησης ενεργοποίησής του.
7. Μάθηση από υπαινικτικά στοιχεία: Η μάθηση που βασίζεται σε δείγμα παραδειγμάτων έχει να κάνει με μία άγνωστη συνάρτηση αντιστοίχισης εισόδου – εξόδου. Γενικεύοντας τον τρόπο αυτό μάθησης, μπορούμε να συμπεριλάβουμε και την μάθηση από υπαινικτικά στοιχεία, που αφορούν σε οποιαδήποτε πληροφορία είναι διαθέσιμη σχετικά με την άγνωστη συνάρτηση.
 8. Ρυθμοί μάθησης: Ιδανικά, όλοι οι νευρώνες θα πρέπει να έχουν τον ίδιο ρυθμό μάθησης. Τα τελευταία επίπεδα όμως παρουσιάζουν μεγαλύτερες αποκλίσεις από τα επίπεδα που βρίσκονται κοντά στην είσοδο. Επομένως, η παράμετρος που συνδέεται με το ρυθμό μάθησης θα πρέπει να έχει μικρότερη τιμή στα τελευταία επίπεδα και μεγαλύτερη στα αρχικά επίπεδα.

4 Πειραματική Διαδικασία

Στο κεφάλαιο αυτό περιγράφεται η μεθοδολογία που ακολουθήθηκε για την ανάπτυξη του μοντέλου πρόβλεψης της ETA. Αρχικά, παρουσιάζονται οι παράμετροι που λήφθηκαν υπόψη για την υλοποίηση, στη συνέχεια αναλύονται ορισμένα σημεία του κώδικα. Τέλος, γίνεται αξιολόγηση του μοντέλου, για την οποία χρησιμοποιούνται δύο μεταβλητές μέτρησης σφάλματος, το μέσο απόλυτο σφάλμα (MAE) των προβλέψεων και το μέσο ριζικό τετραγωνικό σφάλμα (RMSE).

4.1 Πρόβλεψη της ETA με Νευρωνικά Δίκτυα

Για την ανάπτυξη του μοντέλου θα πρέπει να καθοριστούν οι μεταβλητές εισόδου και οι παράμετροι του δικτύου. Οι παράμετροι που είναι απαραίτητες για την ανάπτυξη κάθε νευρωνικού δικτύου είναι οι εξής:

1. Οι μεταβλητές εισόδου στο νευρωνικό δίκτυο
2. Ο αριθμός των κρυφών επιπέδων
3. Ο αριθμός των νευρώνων
4. Ο αριθμός των εξόδων του δικτύου

Η έξοδος του νευρωνικού δικτύου καθορίζεται από το πρόβλημα πρόβλεψης, το οποίο στη δική μας περίπτωση είναι η εκτιμώμενη ώρα άφιξης ETA ενός πλοίου. Επομένως, το επίπεδο εξόδου θα αποτελείται από έναν κόμβο.

Οι εισοδοί στο νευρωνικό δίκτυο είναι τα δεδομένα AIS για κάθε ταξίδι, όπως παρουσιάζονται στον Πίνακα 1. Επομένως, υπάρχουν 5 μεταβλητές εισόδου για το νευρωνικό δίκτυο, βάσει του οποίου γίνονται οι προβλέψεις σχετικά με την ETA.

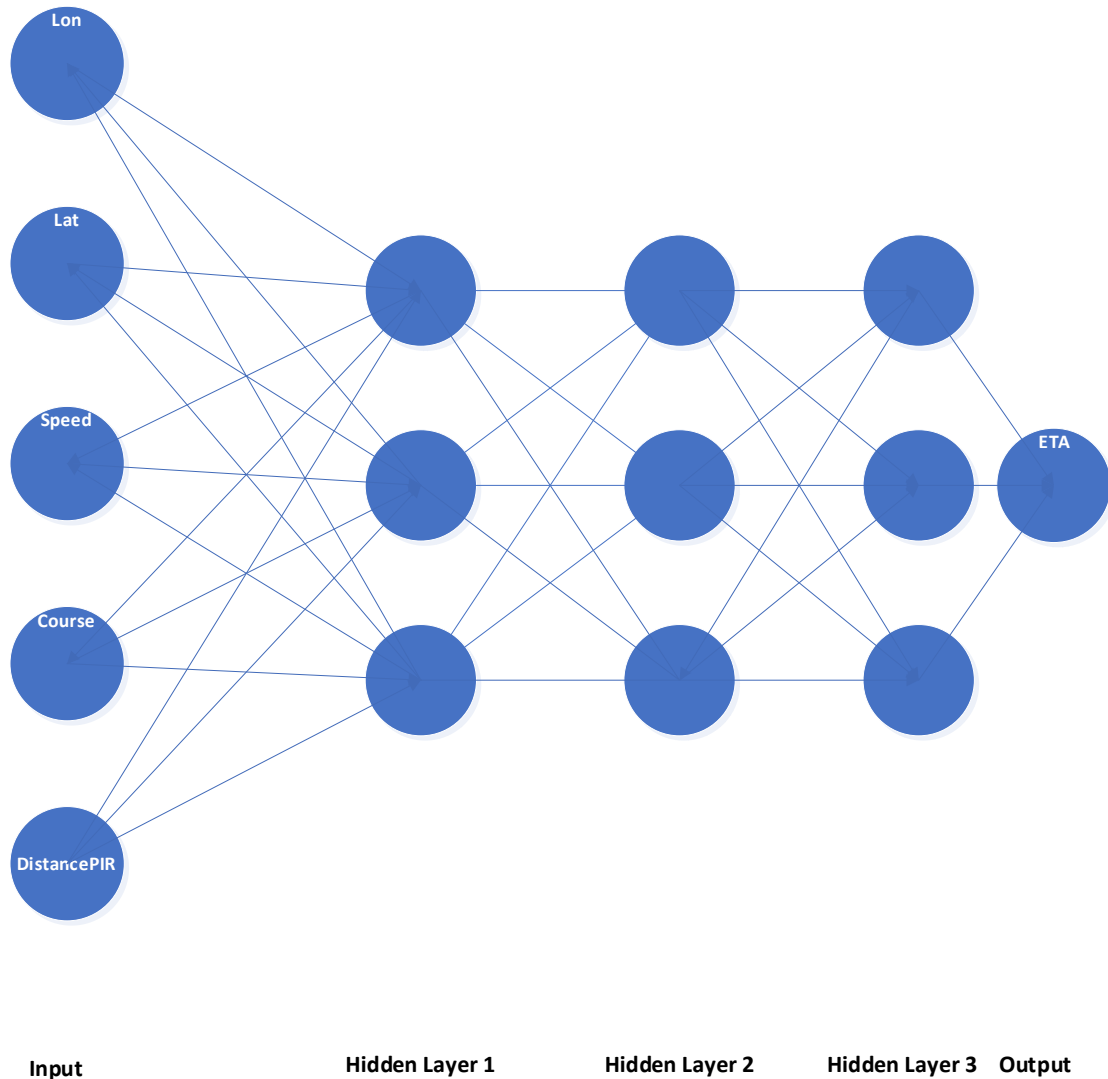
Μεταβλητές Εισόδου	
lon	course
lat	distancePIR
speed	

Πίνακας 3 - Μεταβλητές Εισόδου

Όσον αφορά τον αριθμό των κρυφών επιπέδων, για τα περισσότερα από τα προβλήματα, ένα κρυφό επίπεδο είναι επαρκές για την επίτευξη καλών αποτελεσμάτων [14]. Οι καταστάσεις στις οποίες η απόδοση βελτιώνεται με βαθύτερα νευρωνικά δίκτυα (2ο, 3ο επίπεδο κ.ο.κ.) είναι πολύ λίγες. Ωστόσο, τα βαθύτερα νευρωνικά δίκτυα μπορούν να αντιπροσωπεύουν λειτουργίες οποιουδήποτε είδους και μορφής, υπό την προϋπόθεση ότι υπάρχει αρκετά μεγάλο σύνολο δεδομένων στο οποίο μπορούν να εκπαιδευτούν[14].

Τέλος, ο αριθμός των νευρώνων στο κρυφό επίπεδο είναι καθοριστικής σημασίας. Χρησιμοποιώντας πολύ λίγους νευρώνες μπορεί το μοντέλο να οδηγηθεί σε αποτυχία, πράγμα που σημαίνει ότι ο αριθμός των νευρώνων είναι ανεπαρκής για να αναγνωρίσει τα μοτίβα που είναι κρυμμένα στο σύνολο δεδομένων. Από την άλλη πλευρά, η χρήση πολλών νευρώνων οδηγεί σε υπεργενίκευση, μια περίπτωση όπου το νευρωνικό δίκτυο έχει μεγαλύτερη ικανότητα πληροφόρησης από αυτή που περιέχεται στο σετ εκπαίδευσης.

Το νευρωνικό δίκτυο που χρησιμοποιήσαμε αποτελείται από 3 κρυφά επίπεδα που το καθένα έχει 50, 20 και 10 νευρώνες αντίστοιχα. Ενδεικτικά, η δομή του νευρωνικού δικτύου φαίνεται στην Εικόνα 17. Επιπλέον, για την εκπαίδευση του δικτύου χρησιμοποιείται ο αλγόριθμος οπίσθιας διάδοσης της ενεργοποίησης/σφάλματος (Back Propagation Algorithm).



Εικόνα 17 - Δομή Νευρωνικού Δικτύου

Τα δεδομένα που έχουμε στη διάθεσή μας αφορούν σήματα που καταγράφηκαν σε διάστημα ενός έτους, Ιανουάριος 2016 – Δεκέμβριος 2016, στην ευρύτερη περιοχή του λιμανιού του Πειραιά και του Αργοσαρωνικού. Από τα δεδομένα μας, το 50% θα χρησιμοποιηθεί για να εκπαιδευτεί το νευρωνικό δίκτυο, το 30% για επικαιροποίηση του μοντέλου και το 20% για τον έλεγχο των δεδομένων. Επίσης, το σφάλμα που προκύπτει από την ανάλυση των δεδομένων και με βάση το οποίο θα εξεταστεί η αποτελεσματικότητα του μοντέλου εξετάζεται στη φάση του ελέγχου, καθώς αφορά δεδομένα τα οποία δεν έχει «δει» το νευρωνικό ούτε στη φάση της εκπαίδευσης ούτε σε αυτήν της επικαιροποίησης.

Το Στάδιο της Εκπαίδευσης

Κατά τη διάρκεια της φάσης εκπαίδευσης, παραδείγματα του διανύσματος εισόδου παρουσιάζονται στο νευρωνικό δίκτυο με στόχο την εκπαίδευση του δικτύου, ώστε να κάνει προβλέψεις. Ο στόχος αυτής της φάσης με μάθηση υπό επίτηρηση είναι να παρουσιάσει το σύνολο των εισόδων παράλληλα με τις αντίστοιχες εξόδους, προκειμένου να βρεθούν τα βέλτιστα βάρη που συνδέουν τα διάφορα στρώματα, του νευρωνικού δικτύου. Για την υλοποίηση αυτής της διαδικασίας χρησιμοποιείται ο αλγόριθμος back-propagation.

Το Στάδιο της Επικαιροποίησης

Τα δεδομένα της επικαιροποίησης χρησιμοποιούνται για τον συντονισμό των παραμέτρων του δικτύου. Είναι ανεξάρτητα από αυτά της εκπαίδευσης και λαμβάνουν υπόψη μία μεταβλητή μέτρησης σφάλματος για τον βέλτιστο προσδιορισμό του συνόλου των παραμέτρων. Για κάθε σύνολο παραμέτρων, ένα νευρωνικό δίκτυο εκπαιδεύεται στο σύνολο δεδομένων εκπαίδευσης και στη συνέχεια επικυρώνεται στο σύνολο δεδομένων επικαιροποίησης. Στόχος της φάσης αυτής είναι η επιλογή του νευρωνικού δικτύου που επιτυγχάνει ελαχιστοποίηση του σφάλματος.

Το Στάδιο του Ελέγχου

Κατά τη διάρκεια της φάσης ελέγχου προσδιορίζεται το πραγματικό σφάλμα του νευρωνικού δικτύου. Ένα νέο σύνολο δεδομένων παρουσιάζεται στο μοντέλο και πραγματοποιούνται οι προβλέψεις. Το μέσο απόλυτο σφάλμα από την πραγματική ώρα των αφίξεων είναι το πραγματικό λάθος του μοντέλου. Η διαφορά της παρούσας φάσης με τη φάση της επικαιροποίησης είναι ότι η τελευταία είναι εμμέσως «εξαναγκασμένη», αφού έχει γίνει προσπάθεια ελαχιστοποίησης του σφάλματος με τον καθορισμό των παραμέτρων του μοντέλου. Συνήθως, το σφάλμα στο στάδιο αυτό είναι λίγο μεγαλύτερο από το προηγούμενο, δεδομένου ότι δεν μπορεί να γίνει βελτιστοποίηση σε σύνολο δεδομένων που έχουν παρουσιαστεί στο δίκτυο μόνο μία φορά.

4.2 Ανάλυση Κώδικα Μοντέλου

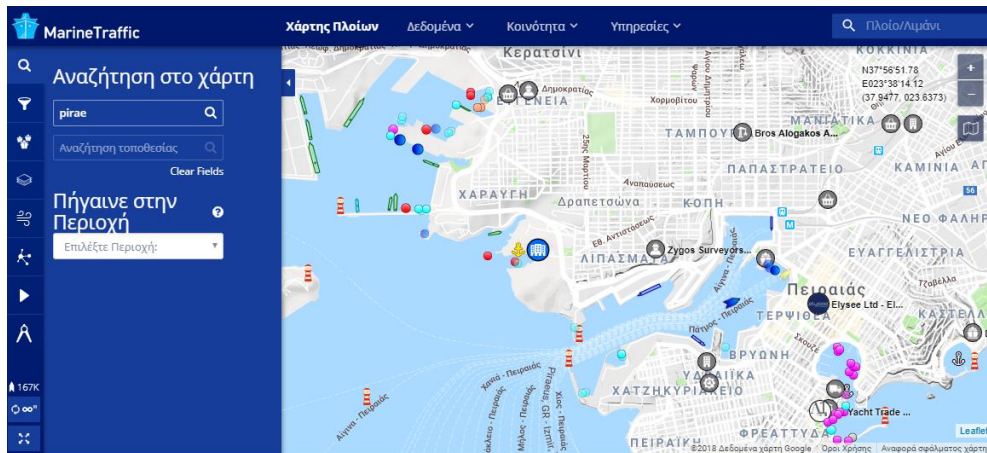
Βασικό κομμάτι στην υλοποίηση του μοντέλου αποτελεί η επεξεργασία των δεδομένων και ο μετασχηματισμός τους, έτσι ώστε να μπορέσουμε να εξάγουμε τις μεταβλητές τόσο της εισόδου όσο και τη μεταβλητή εξόδου, που τελικά θα μοντελοποιηθούν από το δίκτυο.

Τα αρχικά δεδομένα μας χωρίζονται σε δύο κατηγορίες, τα κινηματικά που αφορούν στην πληροφορία των σημάτων AIS και τα στατικά που περιέχουν πληροφορίες, όπως το αναγνωριστικό του πλοίου, το όνομά του, τον τύπο του και άλλα. Το σύνολο των δεδομένων αποθηκεύεται σε μορφή excel. Τα κινηματικά δεδομένα, που αρχικά είναι ομαδοποιημένα ανά μήνα, εντοπίζονται σε μία ενιαία λίστα. Για το συγκεκριμένο πρόβλημα, θα επικεντρωθούμε στα επιβατηγά πλοία τα οποία βρίσκονται σε κίνηση, καθώς διαθέτουν σταθερή πορεία σε σχέση με άλλους τύπους πλοίων, όπως για παράδειγμα τα αλιευτικά. Για το λόγο αυτό, με τη χρήση των κατάλληλων εντολών από το αρχείο των στατικών δεδομένων κρατάμε τα MMSI που είναι τύπου «Passenger» ή «passenger», ενώ από το αρχείο των κινηματικών δεδομένων κρατάμε τις εγγραφές με status=0 (το πλοίο σε κίνηση) και, κατόπιν, αντιστοιχίζουμε σε κάθε εγγραφή των κινηματικών το πλοίο που αφορούν (inner join στο mmsi).

Επειδή τα δεδομένα περιέχουν σφάλματα μετρήσεων και είναι πιθανόν να περιέχουν γεωγραφικές συντεταγμένες που βρίσκονται εκτός της περιοχής ενδιαφέροντός μας, θα πρέπει τα δεδομένα που περιέχουν σφάλματα να εξαιρεθούν. Αυτό γίνεται με τη βοήθεια της συνάρτησης *quantile* που υποστηρίζει η R. Πρόκειται για συνάρτηση κατανομής που κατηγοριοποιεί τα δεδομένα ανάλογα με τη συχνότητα εμφάνισής τους. Έτσι, εξαιρούμε την ακραίες τιμές των παρατηρήσεων του γεωγραφικού μήκους και γεωγραφικού πλάτους, όπως φαίνεται παρακάτω.

```
floorCutlon <- quantile(fullKineticsTomodel$lon,0.001)
cealingCutlon<- quantile(fullKineticsTomodel$lon,0.999)
```

Μία βασική παράμετρος που θα πρέπει να υπολογιστεί είναι η απόσταση κάθε σημείου από το λιμάνι του Πειραιά. Για γίνει αυτό κάνουμε την παραδοχή ότι το λιμάνι αντιπροσωπεύεται από ένα σταθερό σημείο, τις συντεταγμένες του οποίου λαμβάνουμε από τη Marine Traffic.



Εικόνα 18 - Συντεταγμένες Λιμένα Πειραιά[15]

Με τη χρήση της συνάρτησης υπολογισμού της απόστασης, βρίσκουμε την απόσταση κάθε σημείου από το σταθερό σημείο που ορίσαμε παραπάνω και την εισάγουμε ως επιπλέον στήλη στα κινηματικά δεδομένα. Οι νέες εγγραφές ταξινομούνται με βάση το mmsi και το timestamp. Στο στάδιο αυτό γίνεται επιπλέον η παραδοχή ότι σε απόσταση 1Km το πλοίο έχει φτάσει στο λιμάνι, οπότε και «κρατάμε» όλα τα σημεία με απόσταση μεγαλύτερη από αυτή. Δεδομένου ότι ένα πλοίο μέχρι να αγκυροβολήσει, χρειάζεται να κάνει μανούβρες, η παραπάνω παραδοχή γίνεται για να είναι πιο εύκολος ο προσδιορισμός ο χαρακτηρισμός του ταξιδιού ως εισερχόμενο ή εξερχόμενο.

Το πιο σημαντικό κομμάτι για την υλοποίηση του μοντέλου είναι να εξετάσουμε αν το πλοίο εισέρχεται ή εξέρχεται από το λιμάνι προκειμένου να εξαιρεθούν τα «εξερχόμενα» ταξίδια. Για να το πετύχουμε αυτό δημιουργούμε μία νέα μεταβλητή τύπου lag, η οποία περιέχει την απόσταση κάθε σημείου από τον Πειραιά μετατοπισμένη κατά μία θέση προς τα κάτω, ενώ η πρώτη θέση συμπληρώνεται με την τιμή N/A. Υπολογίζουμε τη διαφορά των δύο στηλών και κρατάμε εκείνες τις εγγραφές όπου η διαφορά είναι διάφορη του μηδενός, αφού οι αλλαγές προσήμου δηλώνουν αλλαγή στην κατεύθυνση. Παρακάτω φαίνεται ο τρόπος που χρησιμοποιείται η νέα μεταβλητή.

	A	B	C	D	E	F	G	H
1	DistancePir	lagDist	Diff		sign1	sign2	Diff	
2	10	n/a			+	n/a		
3	5	10	5		+	+	0	
4	3	5	2		+	+	0	
5	2	3	1		+	+	0	
6	1	2	1		+	+	0	
7		1	1					
8								

Εικόνα 19 - Παράδειγμα Μεταβλητής Τύπου Lag

Με την παραπάνω διαδικασία έχουμε καταφέρει να δημιουργήσουμε ένα πίνακα με όλες τις αλλαγές των κατευθύνσεων των τροχιών των πλοίων και παράλληλα δημιουργούμε ένα μετρητή με βάση τις παραπάνω αλλαγές. Στη συνέχεια, συμπληρώνουμε με τα κινηματικά δεδομένα όλα τα σημεία του ταξιδιού του κάθε πλοίου. Επειδή η αλλαγή της κατεύθυνσης φαίνεται στο πρώτο σημείο του νέου ταξιδιού, προκειμένου να αντιστοιχίσουμε σε κάθε γεωγραφικό σημείο ένα TripId, ακολουθούμε την αντίθετη διαδικασία. Κάνοντας τη μεταβλητή τύπου «lead» καταφέρνουμε, επομένως, να αντιστοιχίσουμε όλα τα σημεία σε κάποιο ταξίδι.

Στο τελικό στάδιο χαρακτηρισμού του ταξιδιού ως εισερχόμενο ή εξερχόμενο, υπολογίζουμε την απόσταση του πρώτου και του τελευταίου σημείου κάθε ταξιδιού από το λιμάνι, οπότε και η διαφορά των δύο μας δίνει τον επιθυμητό χαρακτηρισμό. Έτσι, έχουμε κρατήσει τα ταξίδια εκείνα με κατεύθυνση προς το λιμάνι.

Επιπλέον, υπολογίζουμε για κάθε ταξίδι την απόσταση κάθε σημείου από το τελευταίο σημείο της διαδρομής, δηλαδή την είσοδο στο λιμάνι, σύμφωνα με τα παρακάτω.

```
###calculate for each observation the time to PIR
kineticsToModel[,timeToPort:=timestamp[.N]-timestamp, by=tripID]
kineticsToModel[,timeToPort:=round(as.numeric(timeToPort),2)]
```

Έτσι, οι μεταβλητές που ορίζουν το dataset είναι οι ακόλουθες.

```
varToModel <- c("lon", "lat", "speed", "course", "distancePIR", "timeToPort")
modelFrame <- kineticsToModel[,names(kineticsToModel) %in% varToModel, with=F]
```

Από αυτές, ως είσοδοι στο δίκτυο λαμβάνονται οι μεταβλητές lon, lat, speed, course και distancePIR ενώ ως έξοδος του δικτύου ορίζεται η μεταβλητή timeToPort.

```
predictiveModel_NN<-h2o.deeplearning(x=1:5, y="timeToPort",
activation="Rectifier",
training_frame = splits[[1]],
hidden=c(50,20,10),
standardize =T)
```

4.3 Μετρήσεις Σφαλμάτων για την Αξιολόγηση του Μοντέλου

Για την αξιολόγηση της απόδοσης του μοντέλου αλλά και του αποτελέσματος θα χρησιμοποιήσουμε δύο μετρήσεις σφαλμάτων: το Μέσο Απόλυτο Σφάλμα (Mean Absolute Error - MAE) και το Μέσο Σφάλμα Ρίζας (Root Mean Squared Error - RMSE). Η πρώτη μέτρηση μας δίνει μια ένδειξη του μέσου σφάλματος σε ώρες, ενώ η δεύτερη τη διακύμανση των σφαλμάτων πρόβλεψης. Έτσι μπορούμε να αξιολογήσουμε άμεσα τα αποτελέσματα της πρόβλεψης.

Το Μέσο Απόλυτο Σφάλμα είναι μια μέτρηση σφάλματος, που χρησιμοποιείται συνήθως στη στατιστική, για να μετρηθεί ο μέσος όρος της διακύμανση των προβλεπόμενων τιμών με αυτές των πραγματικών. Στην ουσία, αφορά στο μέσο όρο όλων των μέσων σφαλμάτων πρόβλεψης, ενώ μετράται στην ίδια κλίμακα με τα δεδομένα. Δίνεται από τον ακόλουθο τύπο:

$$\text{Μέσο Απόλυτο Σφάλμα} = \frac{1}{n} \sum_{i=1}^n |f_i - y_i|$$

$$\text{Μέσο Απόλυτο Σφάλμα} = \frac{1}{n} \sum_{i=1}^n |ETA_i - ATA_i|$$

όπου f_i : η εκτιμώμενη ώρα άφιξης του πλοίου (ETA)

y_i : η πραγματική ώρα άφιξης του πλοίου (ATA)

n : ο συνολικός αριθμός παρατηρήσεων που θα χρησιμοποιηθούν

Το Μέσο Σφάλμα Ρίζας είναι μια συνηθισμένη μέτρηση σφάλματος στη στατιστική, που συνήθως χρησιμοποιείται για την αξιολόγηση των αριθμητικών προβλέψεων. Σε σύγκριση με το Μέσο Απόλυτο Σφάλμα, το Μέσο Σφάλμα Ρίζας ενισχύει και εντοπίζει καλύτερα τα σφάλματα. Επομένως, χρησιμεύει ως δείκτης για τη διακύμανση των σφαλμάτων πρόβλεψης. Το Μέσο Σφάλμα Ρίζας δίνεται από τον ακόλουθο τύπο:

$$\text{ΜέσοΣφάλμα Ρίζας} = \sqrt{\frac{1}{n} \sum_{i=1}^n (f_i - y_i)^2}$$

$$\text{ΜέσοΣφάλμα Ρίζας} = \sqrt{\frac{1}{n} \sum_{i=1}^n (ETA_i - ATA_i)^2}$$

όπου f_i : η εκτιμώμενη ώρα άφιξης του πλοίου (ETA)

y_i : η πραγματική ώρα άφιξης του πλοίου (ATA)

n : ο συνολικός αριθμός παρατηρήσεων που θα χρησιμοποιηθούν

4.4 Αποτελέσματα

Όπως έχει αναφερθεί παραπάνω, τα δεδομένα έχουν χωριστεί σε επιμέρους υποσύνολα με ποσοστά 50%, 30% και 20% (εκπαίδευση, επικαιροποίηση και έλεγχος). Αυτό το πετυχαίνουμε με την παρακάτω εντολή:

```
splits <- h2o.splitFrame(modelData, ratios = c(0.5, 0.3))
```

Ένα από τα ορίσματα που δέχεται η συνάρτηση `h2o.deeplearning()` καλείται `hidden` και αντιπροσωπεύει τον αριθμό των επιπέδων καθώς και τον αριθμό των κρυφών νευρώνων. Για παράδειγμα, παραμετροποιώντας το όρισμα ως `hidden=c(50,50)` σημαίνει ότι το νευρωνικό δίκτυο αποτελείται από 2 επίπεδα με 50 κρυφούς νευρώνες το καθένα.

Προκειμένου να εκπαιδύσουμε το νευρωνικό δίκτυο, χρησιμοποιήσαμε διαφορετικούς συνδυασμούς κρυφών νευρώνων ανά επίπεδο, εφαρμόζοντάς τους στο `training dataset`. Κατόπιν, εκτιμήσαμε τα σφάλματα και την ακρίβεια «τρέχοντας» το νευρωνικό 10 φορές για κάθε συνδυασμό νευρώνων. Τα αποτελέσματα για κάθε τρέξιμο του σταδίου της εκπαίδευσης παρουσιάζονται αναλυτικά στο Παράρτημα Β.

Ακολουθούμε την ίδια διαδικασία, χρησιμοποιώντας και εφαρμόζοντας τους ίδιους συνδυασμούς νευρώνων στο validating dataset. Κατόπιν, για κάθε συνδυασμό υπολογίζουμε το Μ.Ο. και την τυπική απόκλιση του RMSE και επιλέγουμε αυτόν με τα μικρότερα αποτελέσματα. Έτσι, όπως φαίνεται και στον Πίνακα 4, ο συνδυασμός νευρώνων που επιλέχθηκε είναι ο 50/20/10 με Μ.Ο. $\mu=0,50830758$ και τυπική απόκλιση $\sigma=0,024759068$.

	Αριθμός Κρυφών Επιπέδων	Αριθμός Νευρώνων (Επίπεδο 1)	Αριθμός Νευρώνων (Επίπεδο 2)	Αριθμός Νευρώνων (Επίπεδο 3)	MAE (sec)	RMSE (sec)
Δοκιμή 1	3	50	20	10	172.9149	0.5199607
Δοκιμή 2	3	50	20	10	177.8938	0.5349323
Δοκιμή 3	3	50	20	10	172.8635	0.5198062
Δοκιμή 4	3	50	20	10	160.6912	0.4832038
Δοκιμή 5	3	50	20	10	172.5722	0.5189302
Δοκιμή 6	3	50	20	10	160.8236	0.4836017
Δοκιμή 7	3	50	20	10	179.4534	0.5396224
Δοκιμή 8	3	50	20	10	155.6006	0.4678962
Δοκιμή 9	3	50	20	10	176.8861	0.5319024
Δοκιμή 10	3	50	20	10	160.6966	0.4832199

Πίνακας 4 - Αποτελέσματα Εφαρμογής Συνδυασμού Νευρώνων στο Validation Dataset

Αφού έχουμε επιλέξει το βέλτιστο συνδυασμό νευρώνων για το validating dataset, τον εφαρμόζουμε στο testing dataset, τρέχοντας το νευρωνικό δίκτυο 10 φορές. Τα αποτελέσματα παρουσιάζονται στον Πίνακα 5, από τα οποία προκύπτει $\mu=0,7241$ και $\sigma=0,04987$.

	Αριθμός Κρυφών Επιπέδων	Αριθμός Νευρώνων (Επίπεδο 1)	Αριθμός Νευρώνων (Επίπεδο 2)	Αριθμός Νευρώνων (Επίπεδο 3)	MAE (sec)	RMSE (sec)
Δοκιμή 1	3	50	20	10	193.037	0.7122022
Δοκιμή 2	3	50	20	10	217.0257	0.8007074
Δοκιμή 3	3	50	20	10	210.008	0.7748158
Δοκιμή 4	3	50	20	10	208.1124	0.7678222
Δοκιμή 5	3	50	20	10	185.2272	0.6833882
Δοκιμή 6	3	50	20	10	173.0915	0.638614
Δοκιμή 7	3	50	20	10	188.785	0.6965144
Δοκιμή 8	3	50	20	10	182.9002	0.674803
Δοκιμή 9	3	50	20	10	209.0164	0.7711575
Δοκιμή 10	3	50	20	10	195.3988	0.7209158

Πίνακας 5 - Αποτελέσματα Εφαρμογής Βέλτιστου Συνδυασμού Νευρώνων στο Test Dataset

5 ΣΥΜΠΕΡΑΣΜΑΤΑ

Στην παρούσα εργασία αναπτύχθηκε μοντέλο για τη μελέτη του χρόνου άφιξης των πλοίων στο λιμάνι του Πειραιά με τη χρήση νευρωνικού δικτύου και δεδομένων που αντλήθηκαν από τη βάση του Πανεπιστημίου. Το μοντέλο πέτυχε ικανοποιητικά αποτελέσματα, με μία ακρίβεια στην εκτίμηση του χρόνου άφιξης της τάξης του 80%, δεδομένου ότι ο χρονικός ορίζοντας των δεδομένων είναι βραχυπρόθεσμος. Επιπλέον, η απόκλιση που υπολογίζεται από το απόλυτο μέσο σφάλμα είναι της τάξης των 2-3 λεπτών.

Ένας παράγοντας που θα μπορούσε να επηρεάσει το χρόνο αυτό, είναι καιρικές συνθήκες που επικρατούν καθ' όλη τη διάρκεια του ταξιδιού. Με μια δεύτερη σκέψη, όμως, μπορούμε να πούμε ότι η επίδραση του παράγοντα αυτού δεν είναι σημαντική. Από τη μία, η ταχύτητα με την οποία κινείται το πλοίο αντανακλά εν μέρει και τις καιρικές συνθήκες που επικρατούν στην περιοχή. Από την άλλη, το πιο βασικά σε κάθε δρομολόγιο είναι η άφιξη του πλοίου στο χρόνο που δίνεται από τον πράκτορα. Επομένως, ο καπετάνιος προσαρμόζει την ταχύτητα στις συνθήκες, καθώς μπορεί πάντα να ελαττώνει ή να αυξάνει δεδομένων των συνθηκών. Σε κάθε περίπτωση, βέβαια, το παραπάνω μπορεί να παρουσιάζει σημαντικές αποκλίσεις σε ακραία καιρικά φαινόμενα, όπου για παράδειγμα, έχει απαγορευτεί ο απόπλους και απαιτείται η παραμονή του πλοίου σε ενδιάμεσο λιμάνι. Σε κάθε περίπτωση όμως, η ενσωμάτωση των καιρικών συνθηκών δεν θα είχε σημαντική επίδραση στο χρόνο άφιξης, δεδομένου ότι τα πλοία κατά το κομμάτι της διαδρομής που εξετάζεται, κινούνται σχετικά κοντά στην ακτή και όχι στην ανοιχτή θάλασσα.

Ο τρόπος διαχείρισης των προηγούμενων λιμανιών πριν το πλοίο φτάσει στο λιμάνι του τελικού προορισμού, στη συγκεκριμένη περίπτωση στο λιμάνι του Πειραιά, αποτελεί έναν επιπρόσθετο παράγοντα που μπορεί να επηρεάσει την πρόβλεψη και ο οποίος πρέπει να μελετηθεί. Καθώς υπάρχει αβεβαιότητα στο χρόνο που παραμένουν τα πλοία στο λιμάνι, αυτό μπορεί να οδηγήσει σε σημαντικές αποκλίσεις του αρχικού προγραμματισμού και επομένως να απαιτεί επανασχεδιασμό των δραστηριοτήτων και του χρονοδιαγράμματος.

Καθώς η όσο το δυνατόν ακριβής πρόβλεψη του κατάπλου ενός πλοίου είναι απαραίτητη για τη σωστή οργάνωση και διαχείριση του λιμένα, προτείνεται η διεύρυνση του χρονικού ορίζοντα και η ενσωμάτωση περισσότερων δεδομένων για την αποτελεσματικότερη εκπαίδευση του μοντέλου, αφού το νευρωνικό δίκτυο απαιτεί αρκετά μεγάλο αριθμό δεδομένων για την καταγραφή όλων των μοτίβων. Τέλος, εξίσου σημαντική είναι και η μελέτη των καιρικών συνθηκών σε όλη τη διάρκεια του ταξιδιού και η συσχέτισή τους με την εκτίμηση της πρόβλεψης.

6 ΑΝΑΦΟΡΕΣ

1. Adam Weintrit, "Navigational Systems and Simulators, Marine Navigation and Safety of Sea Transportation". Taylor & Francis Group, LLC (2011)
2. Yu Zheng, "Trajectory Data Mining: An Overview", Microsoft Research
3. Galit Shmueli, "To explain or to Predict", Institute of Mathematical Statistics, 2010
4. Heywood, C., Connor, C., Browning, D., Smith, M. C., & Wang, B. (2009, April). GPS tracking of intermodal transportation: System integration with delivery order system. In Systems and Information Engineering Design Symposium, 2009. SIEDS'09. (pp. 191-196). IEEE
5. Gambarella, L. M., A.E. Rizzoli and M. Zaffalon. Simulation and planning of an intermodal 28 container terminal. Simulation, Vol. 71, No 2, 1998, pp. 107-116.
6. Fancello, G., C. Pani, M. Pisano, P. Serra and P. Fadda. Prediction of arrival times and human resources allocation for container terminal.
7. Parolas, I., Tavasszy L., Kourouniotti I. and vanDuin R., Prediction of vessels' estimated time of arrival (ETA) using machine learning – a port of Rotterdam case study. 96 46 th Annual Meeting of the Transportation Research 47 Board, January 2017
8. Haykin, S. Neural Networks and Learning Machines, Pearson Education, 3rd edition, 2008
9. Goodfellow, Y. Bengio, και A. Courville, Deep Learning. Cambridge, Massachusetts: The MIT Press, 2016
10. PLU: The Piecewise Linear Unit Activation Function, Andrei Nicolae, University of Washington, Seattle, 2018 (<https://arxiv.org/pdf/1809.09534.pdf>)
11. On the Importance of Normalisation Layers in Deep Learning with Piecewise Linear Activation Units, Zhibin Liao, Gustavo Carneiro, Department of Computer Science, University of Adelaide Australia
12. Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, "Image Net Classification with Deep Convolutional Neural Networks", University of Toronto, 2012
13. Adam Coates, Honglak Lee, Andrew Y. Ng, "An Analysis of Single-Layer Networks in Unsupervised Feature Learning", Stanford University and University of Michigan
14. Heaton, J. (2008). Introduction to Neural Networks with Java, Second Edition. St. Louis: Heaton Research, Inc.
15. Marine Traffic Homepage, <https://help.marinetraffic.com/hc/en-us/articles/204581828-What-is-the-Automatic-Identification-System-AIS->
16. Οργανισμός Λιμένος Πειραιώς Homepage, <http://www.olp.gr/el/the-port-of-piraeus/strategy-vision>
17. Navigation Center Homepage, <https://www.navcen.uscg.gov/?pageName=AISmain>
18. http://infolab.cs.unipi.gr/aminess_demo/AIS_antenna_logs.html
19. International Maritime Organization, <http://www.imo.org>
20. <https://www.r-project.org/>
21. [https://en.wikipedia.org/wiki/R_\(programming_language\)](https://en.wikipedia.org/wiki/R_(programming_language))

ΠΑΡΑΡΤΗΜΑ Α

Στο παράρτημα αυτό παρουσιάζεται ο κώδικας σε R της εφαρμογής. Χωρίζεται σε τρία μέρη. Αρχικά, το αρχείο *DataEngineering.R* (1) όπου γίνεται η επεξεργασία των δεδομένων προκριμένου μα μπορούν να εισαχθούν ως είσοδοι στο ΝΔ. Ο κώδικας αυτός καλεί το αρχείο *auxiliaryFunctions.R* (2) όπου υπολογίζεται για κάθε σημείο η απόσταση του από ένα σταθερό σημείο. Τέλος, το αρχείο *PredictiveModel.R* (3) που αποτελεί και την υλοποίηση του ΝΔ.

DataEngineering.R

```
#####Data Engineering for Boat Arrival Prediction Model#####

### load libs
library(data.table);library(doParallel);library(foreach);library(zoo);library(devtools)

### load helper user defined functions
source("/PiachaMSc/BoatArrivalPrediction/auxiliaryFunctions.R")

options(scipen=999)

###list files
# setwd("./kinematics/")

###get names of kinetics files
fileNames <- list.files(paste0(getwd()))

###load ship id
shipID <- fread("ship_id.csv")

###isolate passenger ships
shipID_passenger <- shipID[type %like% "Passenger" | type %like% "passenger"]

###get data by name to a list
dataList <- lapply(fileNames, fread)

###bind list into a data.table
longKinetics <- rbindlist(dataList)

###save indermediate file
saveRDS(longKinetics, "longKinetics.rds")

###load intermidiate file
system.time(longKinetics <- readRDS("longKinetics.rds"))

###verify all ais type are position reports
table(longKinetics$type)
# 1    2    3    18    19
# 60328736 486998 8337485 645770 10318
### drop type variable
longKinetics[,type:=NULL]
```



```

####keep only boats with status=0 "under way using engine"
longKinetics <- longKinetics[status==0]

#### set primary key for shipID
setkeyv(shipID_passenger,"mmsi")

####set primary key for kinetics
setkeyv(longKinetics,"mmsi")

####merge shipid_passenger with kinetics
fullKinetics <- merge(longKinetics,shipID_passenger)

####turn UNIX timestamps into datetime R types
fullKinetics[,timestamp:=as.POSIXct(as.numeric(as.character(timestamp)),
                                     origin="1970-01-01",
                                     tz="Europe/Athens")]

####delete initial kinetics table for memory benefit
rm(longKinetics)

####keep only relevant variables
varsToKeep <- c("mmsi","timestamp","lon","lat","speed","heading","course")

####subset relevant variables
fullKineticsTomodel <- fullKinetics[,names(fullKinetics) %in% varsToKeep,
                                     with=F]

####save intermediate file
saveRDS(fullKineticsTomodel,"kineticsTofeatures.rds")

####read intermediate file
fullKineticsTomodel <- readRDS("kineticsTofeatures.rds")

####keep only non NA observations
fullKineticsTomodel <- fullKineticsTomodel[complete.cases(fullKineticsTomodel)]

#### clean outliers in lon lat

floorCutlon <- quantile(fullKineticsTomodel$lon,0.001)
cealingCutlon<- quantile(fullKineticsTomodel$lon,0.999)

floorCutlat <- quantile(fullKineticsTomodel$lat,0.001)
cealingCutlat<- quantile(fullKineticsTomodel$lat,0.999)

fullKineticsTomodel <- fullKineticsTomodel[lon > floorCutlon & lon < cealingCutlon]
fullKineticsTomodel <- fullKineticsTomodel[lat > floorCutlat & lat < cealingCutlat]

#### add pireus flag 37.9405547,23.6245785
fullKineticsTomodel <- fullKineticsTomodel[,c("lonPIR","latPIR"):=.(23.62457, 37.94055)]

#### calculate distance of each geo point from PIR

```

```

####build distibuted process
vessel <- unique(fullKineticsToModel$mmsi)
cl<-makeCluster(detectCores())
registerDoParallel(cl)

distpar <- foreach (i = vessel) %dopar% distVessel(i)
dist_par_all <- rbindlist(distpar)
stopCluster(cl)

#### save intermediate file
saveRDS(dist_par_all,"kineticsTofeatureswithDist.rds")

####load intermediate file
dist_par_all <- readRDS("kineticsTofeatureswithDist.rds")

####order by mmsi and timestamp

kineticsToModel <- dist_par_all[order(mmsi,timestamp)]

##remove low observations vessels
vesselsToCut <- kineticsToModel[,.N,by=mmsi][N<100]$mmsi
kineticsToModel <- kineticsToModel[!mmsi %in% vesselsToCut]

####characterize trip as inbound or outbound

####create lag of distance [lag=1]
kineticsToModel[, lagDistancePort := shift(distancePIR,
                                           1L,
                                           fill=NA,
                                           type="lag"),
                by=mmsi]

####calculate difference of lagged distance with distance
kineticsToModel[,DifferenceLag:=lagDistancePort-distancePIR]
kineticsToModel <- kineticsToModel[!DifferenceLag==0]

####find change in trend of distance
kineticsToModel[,sign1:=sign(DifferenceLag)]

kineticsToModel[, sign2 := shift(sign1,
                                 1L,
                                 fill=NA,
                                 type="lag"),
                by=mmsi]

kineticsToModel <- kineticsToModel[complete.cases(kineticsToModel)]

####flag change of trip
kineticsToModel[,flagOfChange:=ifelse(sign1==sign2,0,1)]

####remove time points were boat is inactive
kineticsToModel <- kineticsToModel[! sign1==0 ]

```

```

kineticsToModel <- kineticsToModel[! sign2==0 ]

###create a table only with points of change
onlyChanges <- kineticsToModel[flagOfChange==1]
onlyChanges[,tripID:=1:nrow(onlyChanges)]
onlyChanges <- onlyChanges[,c("mmsi", "timestamp", "tripID")]

###set primary keys
setkeyv(onlyChanges,c("mmsi", "timestamp"))
setkeyv(kineticsToModel,c("mmsi", "timestamp"))

### create unique tripID's
fullTripInfo <- merge(kineticsToModel,onlyChanges,all.x = T)

###shift to match point of new trip
fullTripInfo[,tripID:=shift(tripID,1,fill=NA,type = "lead")]

###fill NA's over each number wuth that number[tripID]
fullTripInfo[,tripID:=na.locf(tripID,fromLast = T,na.rm = T)]

###remove repetitive tripID==1
fullTripInfo <- fullTripInfo[!tripID==1]

###check if the vessel is moving towards or away from PIR
fullTripInfo[,c("first", "last"):=.(distancePIR[1],distancePIR[.N]),
  by=(tripID)][,direction:=first-last]

###characterize trip as inbound ori outbound
fullTripInfo[,tripType:=ifelse(direction<0,"outbound", "inbound")]

###keep only inbound trips
dataToModel <- fullTripInfo[tripType=="inbound"]

###save intermediate file
saveRDS(dataToModel,"kineticsInbound.rds")
kineticsInbound <- readRDS("kineticsInbound.rds")

###remove variables we dont need anymore
varsToDrop <-c("lonPIR", "latPIR", "lagDistancePort", "DifferenceLag", "sign1", "sign2",
  "flagOfChange", "first", "last", "direction", "tripType")
kineticsToModel <- kineticsInbound[,!names(kineticsInbound) %in% varsToDrop,with=F]

###calculate for each observation the time to PIR
kineticsToModel[,timeToPort:=timestamp[.N]-timestamp,
  by=tripID]
kineticsToModel[,timeToPort:=round(as.numeric(timeToPort),2)]

###build model frame
varToModel <- c("lon", "lat", "speed", "course", "distancePIR")
modelFrame <- kineticsToModel[,names(kineticsToModel) %in% varToModel,with=F]

```

```
###save data to model
saveRDS(modelFrame,"modelFrame.rds")
```

auxiliaryFunctions.R

```
###function to calculate distance
```

```
earth.dist <- function (long1, lat1, long2, lat2)
{
  rad <- pi/180
  a1 <- lat1 * rad
  a2 <- long1 * rad
  b1 <- lat2 * rad
  b2 <- long2 * rad
  dlon <- b2 - a2
  dlat <- b1 - a1
  a <- (sin(dlat/2))^2 + cos(a1) * cos(b1) * (sin(dlon/2))^2
  c <- 2 * atan2(sqrt(a), sqrt(1 - a))
  R <- 6378.145
  d <- R * c
  return(d)
}
```

```
### function to calculate distance of every point to PIR
```

```
distVessel <- function(Indvessel){
  library(data.table)
  vesselfullKineticsTomodel <- fullKineticsTomodel[mmsi==Indvessel]
  distList <- list()
  for (i in 1:nrow(vesselfullKineticsTomodel)){
    distList[[i]] <- earth.dist(vesselfullKineticsTomodel[i,]$lon,
                               vesselfullKineticsTomodel[i,]$lat,
                               vesselfullKineticsTomodel[i,]$lonPIR,
                               vesselfullKineticsTomodel[i,]$latPIR)
    print(paste0("prcComp: ",round(i/nrow(vesselfullKineticsTomodel)*100 ,3), " %")) }
  distVec <- unlist(distList)
  vesselfullKineticsTomodel[,distancePIR:=distVec]
  vesselfullKineticsTomodel}
```

PredictiveModel.R

```
#####Boat Arrival Prediction Model#####

### load libs
library(data.table);library(h2o)
set.seed(80)
options(scipen=999)

### load helper user defined functions
setwd("C:/PiachaMSc/BoatArrivalPrediction/kinematics/")

###read model data

#start H2O on your local machine using all available cores
h2o.init(nthreads=-1)

data <- readRDS("modelFrame.rds")

modelData <- as.h2o(data)

### split data into training test
splits <- h2o.splitFrame(modelData, ratios = c(0.5, 0.3))

#specify the parameters of the DL model
predictiveModel_NN<-h2o.deeplearning(x=1:5, y="timeToPort",
  activation="Rectifier",
  training_frame = splits[[1]],
  validation_frame = splits[[2]],
  hidden=c(30,30,30),
  standardize =T)

predictions <- h2o.predict(predictiveModel_NN,splits[[3]])

predFrame <- cbind(as.data.table(splits[[3]]),as.data.table(round(predictions)))

predFrame[,MAE:=mean((abs(timeToPort-predict)))]

predFrame[,RMSE:=sqrt((mean(abs(timeToPort-predict))^2)/nrow(predFrame))]

#save models
saveRDS(predictiveModel_NN,"DNN_Boat.rds")
saveRDS(predFrame,"predictions.rds")
```

ΠΑΡΑΡΤΗΜΑ Β

Στο παράρτημα αυτό παρουσιάζονται τα αποτελέσματα του προγράμματος. Για κάθε dataset, παραμετροποιούμε την μεταβλητή hidden και «τρέχουμε» το πρόγραμμα 10 φορές για κάθε αλλαγή των παραμέτρων και στη συνέχεια λαμβάνουμε τους μέσους όρους των σφαλμάτων.

- **Training Dataset**

Αριθμός Νευρώνων ανά Επίπεδο: 30/30/30

	Αριθμός Κρυφών Επιπέδων	Αριθμός Νευρώνων (Επίπεδο 1)	Αριθμός Νευρώνων (Επίπεδο 2)	Αριθμός Νευρώνων (Επίπεδο 3)	MAE (sec)	RMSE (sec)
Δοκιμή 1	3	30	30	30	183.2149	0.427502
Δοκιμή 2	3	30	30	30	148.7741	0.34714
Δοκιμή 3	3	30	30	30	193.8664	0.4523555
Δοκιμή 4	3	30	30	30	157.2314	0.3668736
Δοκιμή 5	3	30	30	30	172.2797	0.4019865
Δοκιμή 6	3	30	30	30	151.6262	0.353795
Δοκιμή 7	3	30	30	30	189.6716	0.4425677
Δοκιμή 8	3	30	30	30	159.2322	0.3715423
Δοκιμή 9	3	30	30	30	176.8333	0.4126116
Δοκιμή 10	3	30	30	30	188.8242	0.4405903

Αριθμός Νευρώνων ανά Επίπεδο: 100/100/100

	Αριθμός Κρυφών Επιπέδων	Αριθμός Νευρώνων (Επίπεδο 1)	Αριθμός Νευρώνων (Επίπεδο 2)	Αριθμός Νευρώνων (Επίπεδο 3)	MAE (sec)	RMSE (sec)
Δοκιμή 1	3	100	100	100	162.1247	0.3782914
Δοκιμή 2	3	100	100	100	180.8401	0.4219608
Δοκιμή 3	3	100	100	100	197.7159	0.4613377
Δοκιμή 4	3	100	100	100	151.7067	0.3539827
Δοκιμή 5	3	100	100	100	171.2041	0.3994768
Δοκιμή 6	3	100	100	100	182.3611	0.4255097
Δοκιμή 7	3	100	100	100	156.7569	0.3657667
Δοκιμή 8	3	100	100	100	156.2973	0.3646941
Δοκιμή 9	3	100	100	100	184.8533	0.4313249
Δοκιμή 10	3	100	100	100	190.970	0.4455971

Αριθμός Νευρώνων ανά Επίπεδο: 50/30/10

	Αριθμός Κρυφών Επιπέδων	Αριθμός Νευρώνων (Επίπεδο 1)	Αριθμός Νευρώνων (Επίπεδο 2)	Αριθμός Νευρώνων (Επίπεδο 3)	MAE (sec)	RMSE (sec)
Δοκιμή 1	3	50	30	10	167.9545	0.3918943
Δοκιμή 2	3	50	30	10	157.6988	0.3679644
Δοκιμή 3	3	50	30	10	194.0636	0.4528157
Δοκιμή 4	3	50	30	10	179.9063	0.419782
Δοκιμή 5	3	50	30	10	169.3196	0.3950796
Δοκιμή 6	3	50	30	10	194.8134	0.4545651
Δοκιμή 7	3	50	30	10	152.7858	0.3565007
Δοκιμή 8	3	50	30	10	156.8857	0.366067
Δοκιμή 9	3	50	30	10	185.7017	0.4333046
Δοκιμή 10	3	50	30	10	157.6713	0.3679001

Αριθμός Νευρώνων ανά Επίπεδο: 40/30/20

	Αριθμός Κρυφών Επιπέδων	Αριθμός Νευρώνων (Επίπεδο 1)	Αριθμός Νευρώνων (Επίπεδο 2)	Αριθμός Νευρώνων (Επίπεδο 3)	MAE (sec)	RMSE (sec)
Δοκιμή 1	3	40	30	20	177.8364	0.4149522
Δοκιμή 2	3	40	30	20	157.1906	0.3667784
Δοκιμή 3	3	40	30	20	172.9052	0.403446
Δοκιμή 4	3	40	30	20	161.9139	0.3777997
Δοκιμή 5	3	40	30	20	161.4255	0.37666
Δοκιμή 6	3	40	30	20	149.3517	0.3484878
Δοκιμή 7	3	40	30	20	160.5087	0.3745208
Δοκιμή 8	3	40	30	20	211.005	0.4923455
Δοκιμή 9	3	40	30	20	202.3186	0.4720772
Δοκιμή 10	3	40	30	20	193.512	0.4515287

Αριθμός Νευρώνων ανά Επίπεδο: 50/20/10

	Αριθμός Κρυφών Επιπέδων	Αριθμός Νευρώνων (Επίπεδο 1)	Αριθμός Νευρώνων (Επίπεδο 2)	Αριθμός Νευρώνων (Επίπεδο 3)	MAE (sec)	RMSE (sec)
Δοκιμή 1	3	50	20	10	205.7531	0.4800912
Δοκιμή 2	3	50	20	10	186.6	0.4354005
Δοκιμή 3	3	50	20	10	161.6377	0.3771552
Δοκιμή 4	3	50	20	10	157.1838	0.3667628
Δοκιμή 5	3	50	20	10	169.2472	0.3949106
Δοκιμή 6	3	50	20	10	182.1859	0.425101
Δοκιμή 7	3	50	20	10	172.2528	0.4019237
Δοκιμή 8	3	50	20	10	181.1005	0.4225683
Δοκιμή 9	3	50	20	10	182.2313	0.425207
Δοκιμή 10	3	50	20	10	155.8124	0.3635628

- **Validating Dataset**

Αριθμός Νευρώνων ανά Επίπεδο: 30/30/30

	Αριθμός Κρυφών Επιπέδων	Αριθμός Νευρώνων (Επίπεδο 1)	Αριθμός Νευρώνων (Επίπεδο 2)	Αριθμός Νευρώνων (Επίπεδο 3)	MAE (sec)	RMSE (sec)
Δοκιμή 1	3	30	30	30	187.8022	0.5647275
Δοκιμή 2	3	30	30	30	168.622	0.507052
Δοκιμή 3	3	30	30	30	183.9325	0.5530912
Δοκιμή 4	3	30	30	30	178.7058	0.5373741
Δοκιμή 5	3	30	30	30	188.2222	0.5659904
Δοκιμή 6	3	30	30	30	173.8975	0.5229154
Δοκιμή 7	3	30	30	30	170.5662	0.5128981
Δοκιμή 8	3	30	30	30	159.4717	0.4795366
Δοκιμή 9	3	30	30	30	181.5129	0.5458153
Δοκιμή 10	3	30	30	30	159.704	0.480235

M.O. RMSE: 0,52091085

Τυπική Απόκλιση RMSE: 0,027770514

Αριθμός Νευρώνων ανά Επίπεδο: 100/100/100

	Αριθμός Κρυφών Επιπέδων	Αριθμός Νευρώνων (Επίπεδο 1)	Αριθμός Νευρώνων (Επίπεδο 2)	Αριθμός Νευρώνων (Επίπεδο 3)	MAE (sec)	RMSE (sec)
Δοκιμή 1	3	100	100	100	161.8797	0.4867776
Δοκιμή 2	3	100	100	100	170.8121	0.5136375
Δοκιμή 3	3	100	100	100	165.8566	0.4987362
Δοκιμή 4	3	100	100	100	188.6249	0.5672013
Δοκιμή 5	3	100	100	100	172.1539	0.5176723
Δοκιμή 6	3	100	100	100	157.4938	0.4735891
Δοκιμή 7	3	100	100	100	175.2344	0.5269355
Δοκιμή 8	3	100	100	100	188.2222	0.5659904
Δοκιμή 9	3	100	100	100	183.8032	0.5527023
Δοκιμή 10	3	100	100	100	179.7049	0.5403784

M.O. RMSE: 0,52436206

Τυπική Απόκλιση RMSE: 0,030726817

Αριθμός Νευρώνων ανά Επίπεδο: 50/30/10

	Αριθμός Κρυφών Επιπέδων	Αριθμός Νευρώνων (Επίπεδο 1)	Αριθμός Νευρώνων (Επίπεδο 2)	Αριθμός Νευρώνων (Επίπεδο 3)	MAE (sec)	RMSE (sec)
Δοκιμή 1	3	50	30	10	184.1327	0.553693
Δοκιμή 2	3	50	30	10	163.0606	0.4903286
Δοκιμή 3	3	50	30	10	177.8916	0.534926
Δοκιμή 4	3	50	30	10	176.6951	0.531328
Δοκιμή 5	3	50	30	10	188.4051	0.5665404
Δοκιμή 6	3	50	30	10	178.7295	0.5374455
Δοκιμή 7	3	50	30	10	185.0113	0.5563349
Δοκιμή 8	3	50	30	10	167.0525	0.5023324
Δοκιμή 9	3	50	30	10	189.243	0.5690599
Δοκιμή 10	3	50	30	10	230.5582	0.6932961

M.O. RMSE: 0,55352848

Τυπική Απόκλιση RMSE: 0,052598731

Αριθμός Νευρώνων ανά Επίπεδο: 40/30/20

Μελέτη του Εκτιμώμενου Χρόνου Άφιξης πλοίων
στο λιμάνι του Πειραιά με την χρήση Νευρωνικών Δικτύων

	Αριθμός Κρυφών Επιπέδων	Αριθμός Νευρώνων (Επίπεδο 1)	Αριθμός Νευρώνων (Επίπεδο 2)	Αριθμός Νευρώνων (Επίπεδο 3)	MAE (sec)	RMSE (sec)
Δοκιμή 1	3	40	30	20	173.9336	0.5230239
Δοκιμή 2	3	40	30	20	162.9136	0.4898866
Δοκιμή 3	3	40	30	20	175.6099	0.5280646
Δοκιμή 4	3	40	30	20	205.9799	0.6193882
Δοκιμή 5	3	40	30	20	176.8861	0.5319024
Δοκιμή 6	3	40	30	20	182.3722	0.548399
Δοκιμή 7	3	40	30	20	212.0137	0.6375321
Δοκιμή 8	3	40	30	20	172.5216	0.5187781
Δοκιμή 9	3	40	30	20	194.1319	0.5837609
Δοκιμή 10	3	40	30	20	167.5733	0.5038984

M.O. RMSE: 0,54846342

Τυπική Απόκλιση RMSE: 0,046739193

Αριθμός Νευρώνων ανά Επίπεδο: 50/20/10

	Αριθμός Κρυφών Επιπέδων	Αριθμός Νευρώνων (Επίπεδο 1)	Αριθμός Νευρώνων (Επίπεδο 2)	Αριθμός Νευρώνων (Επίπεδο 3)	MAE (sec)	RMSE (sec)
Δοκιμή 1	3	50	20	10	172.9149	0.5199607
Δοκιμή 2	3	50	20	10	177.8938	0.5349323
Δοκιμή 3	3	50	20	10	172.8635	0.5198062
Δοκιμή 4	3	50	20	10	160.6912	0.4832038
Δοκιμή 5	3	50	20	10	172.5722	0.5189302
Δοκιμή 6	3	50	20	10	160.8236	0.4836017
Δοκιμή 7	3	50	20	10	179.4534	0.5396224
Δοκιμή 8	3	50	20	10	155.6006	0.4678962
Δοκιμή 9	3	50	20	10	176.8861	0.5319024
Δοκιμή 10	3	50	20	10	160.6966	0.4832199

M.O. RMSE: 0,50830758

Τυπική Απόκλιση RMSE: 0,024759068

- **Test Dataset**

Αριθμός Νευρώνων ανά Επίπεδο: 50/20/10

	Αριθμός Κρυφών Επιπέδων	Αριθμός Νευρώνων (Επίπεδο 1)	Αριθμός Νευρώνων (Επίπεδο 2)	Αριθμός Νευρώνων (Επίπεδο 3)	MAE (sec)	RMSE (sec)
Δοκιμή 1	3	50	20	10	193.037	0.7122022
Δοκιμή 2	3	50	20	10	217.0257	0.8007074
Δοκιμή 3	3	50	20	10	210.008	0.7748158
Δοκιμή 4	3	50	20	10	208.1124	0.7678222
Δοκιμή 5	3	50	20	10	185.2272	0.6833882
Δοκιμή 6	3	50	20	10	173.0915	0.638614
Δοκιμή 7	3	50	20	10	188.785	0.6965144
Δοκιμή 8	3	50	20	10	182.9002	0.674803
Δοκιμή 9	3	50	20	10	209.0164	0.7711575
Δοκιμή 10	3	50	20	10	195.3988	0.7209158

M.O. RMSE: 0,7241

Τυπική Απόκλιση RMSE: 0,04987