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

Σχεδίαση, υλοποίηση και εκπαίδευση μοντέλων Μηχανικής Μάθησης σε ενεργειακά και μετεωρολογικά δεδομένα για την πρόβλεψη φορτίου.

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΕΜΜΑΝΟΥΗΛ ΠΑΝΑΓΙΩΤΟΥ

Επιβλέπων : Βεσκούκης Βασίλειος
Αν. Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2020

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

Σχεδίαση, υλοποίηση και εκπαίδευση μοντέλων Μηχανικής Μάθησης σε ενεργειακά και μετεωρολογικά δεδομένα για την πρόβλεψη φορτίου.

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΕΜΜΑΝΟΥΗΛ ΠΑΝΑΓΙΩΤΟΥ

Επιβλέπων : Βεσκούκης Βασίλειος
Αν. Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 20η Οκτωβρίου 2020.

.....	.....	.....
Βεσκούκης Βασίλειος Αν. Καθηγητής Ε.Μ.Π.	Παπασπύρου Νικόλαος Καθηγητής Ε.Μ.Π.	Χατζηαργυρίου Νικόλαος Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2020

Εμμανουήλ Παναγιώτου

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

Copyright © Εμμανουήλ Παναγιώτου, 2020.
Με επιφύλαξη παντός δικαιώματος. All rights reserved.

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

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

Περίληψη

Η ορθή πρόβλεψη της συμπεριφοράς του καιρού καθώς και της παραγωγής και κατανάλωσης φορτίου είναι ιδιαίτερα σημαντική σε πολλαπλούς επιστημονικούς τομείς καθώς και σε επιχειρήσεις. Μια βελτιωμένη στρατηγική πρόβλεψης μπορεί να βοηθήσει εταιρείες παροχής ενέργειας να ανταποκριθούν στην ισορροπία ζήτησης/προσφοράς, αποφεύγοντας απώλειες και αναπτύσσοντας κέρδη. Σε αυτή την εργασία, μετεωρολογικά δεδομένα που έχουν συλλεχθεί από συσκευές και σένσορες IoT και συνδυάζονται με πανελλαδικά δεδομένα φορτίου, χρησιμοποιούνται για την κατασκευή προβλέψεων καθώς και για την βελτίωση υπαρχόντων προγνώσεων. Τα δεδομένα αποθηκεύονται και φιλτράρονται από κατάλληλες μεθόδους προεπεξεργασίας για την διασφάλιση της ποιότητάς τους καθώς και το ότι οι προβλέψεις θα είναι ανεξάρτητες τυχαίου θορύβου. Τα πανελλαδικά δεδομένα καιρού κανονικοποιούνται σε σχέση με τα δημογραφικά στοιχεία της εκάστοτε περιοχής, σε μια προσπάθεια εκτίμησης της κατανάλωσης της ηλεκτρικής ενέργειας. Για την δημιουργία προβλέψεων μελετώνται και υλοποιούνται διάφορα μοντέλα βαθιάς μηχανικής μάθησης, εστιάζοντας κυρίως σε αναδρομικά δίκτυα LSTM. Γίνεται πειραματισμός και αξιολόγηση σε σχέση με την ποιότητα και το πλήθος των δεδομένων εισόδου, την αποδοτικότητα των μοντέλων, την βελτίωση ήδη υπαρχόντων προγνώσεων, καθώς και ως προς τον αριθμό των ημερών για τις οποίες παράγεται πρόβλεψη. Τέλος αναπτύσσεται εργαλείο αυτόματης οπτικοποίησης των παραγόμενων χρονοσειρών πρόβλεψης, έτοιμο για παραγωγή, με φιλικό προς τον χρήστη περιβάλλον.

Λέξεις κλειδιά

Machine Learning, Time series Forecasting, Deep Learning, Data Preprocessing, Load Forecasting, CNN, RNN, LSTM.

Abstract

Proper forecasting of weather behavior as well as predicting load production and consumption is particularly important in multiple scientific fields as well as in businesses. An improved forecasting strategy can assist power or energy providing companies to meet the demand and supply equilibrium, avoiding losses and increasing profits. In this thesis, meteorological data collected from IoT devices and sensors, combined with nationwide load data are used to make predictions as well as to improve existing forecasts. The acquired data are stored and filtered by appropriate preprocessing methods not only ensuring their quality but also that the predictions will be independent of random noise. Nationwide weather data are normalized in relation to the demographics of each region, in an effort to estimate electricity consumption. To generate predictions, multiple machine learning models are built and analysed, focusing mainly on recurrent networks, such as LSTM. Experiments are conducted and evaluation is performed regarding the quality and quantity of the input data, the efficiency of the models, the improvement of existing forecasts, as well as the number of days for which forecasts are generated. Finally a ready for production tool with a user-friendly interface for automatic visualization of the generated predictions and time series is developed.

Key words

Machine Learning, Time series Forecasting, Deep Learning, Data Preprocessing, Load Forecasting, CNN, RNN, LSTM.

Ευχαριστίες

Ευχαριστώ θερμά τον επιβλέποντα καθηγητή αυτής της διατριβής, κ. Βεσκούκη Βασίλειο, για τη συνεχή καθοδήγηση και εμπιστοσύνη του. Ευχαριστώ επίσης τον Αν. Καθηγητή Τομέα Τοπογραφίας Δουλάμη Νικόλαο για τις πολύτιμες συμβουλές και τα χρήσιμα σχόλιά του καθώς και τα μέλη της επιτροπής κ. Παπασπύρου Νικόλαο και κ. Χατζηαργυρίου Νικόλαο. Θα ήθελα τέλος να ευχαριστήσω την οικογένειά μου και κυρίως τους γονείς μου, οι οποίοι με υποστήριξαν και έκαναν δυνατή την απερίσπαστη ενασχόλησή μου τόσο με την εκπόνηση της διπλωματικής μου, όσο και συνολικά με τις σπουδές μου.

Εμμανουήλ Παναγιώτου,

Αθήνα, 20η Οκτωβρίου 2020

Περιεχόμενα

Περίληψη 5

Abstract 7

Ευχαριστίες 9

Περιεχόμενα 11

Κατάλογος σχημάτων 13

1. Εισαγωγή 15
1.1 Σκοπός της εργασίας 15
1.2 Δομή της εργασίας 15
1.3 Τοποθέτηση του προβλήματος 16

2. Μηχανική Μάθηση 19
2.1 Διαδικασίες Μάθησης 19
2.1.1 Επιβλεπόμενη Μάθηση 19
2.1.2 Μάθηση χωρίς εκπαιδευτή 20
2.1.2.1 Ενισχυτική Μάθηση 20
2.1.2.2 Μη Επιβλεπόμενη Μάθηση 20
2.2 Νευρωνικά Δίκτυα 20
2.2.1 Το μοντέλο του τεχνητού Νευρώνα 20
2.2.2 Συναρτήσεις ενεργοποίησης 21
2.2.2.1 Sigmoid or Logistic 21
2.2.2.2 TanH or hyperbolic tangent 22
2.2.2.3 Softmax 22
2.2.2.4 ReLU 23
2.2.3 Αρχιτεκτονικές Νευρωνικών Δικτύων 24
2.2.4 Τεχνικές Εκπαίδευσης Νευρωνικών Δικτύων 26
2.2.4.1 Γραμμική Παλινδρόμηση (Gradient descent) 26
2.2.4.2 Οπισθοδιάδοση (Backpropagation) 26
2.2.4.3 Οπισθοδιάδοση στο χρόνο (BPTT) 27
2.3 Μοντέλα Βαθιάς Μάθησης (Deep Learning) 28
2.3.1 Συνελικτικά δίκτυα 28
2.3.2 Long Short Term Memory (LSTM) Neural Networks 29
2.3.3 Bidirectional LSTM 33

3. Απόκτηση, Επεξεργασία και Ανάλυση Δεδομένων χρονοσειράς 35
3.1 Εισαγωγή 35
3.2 Internet of Things (IoT) 35
3.2.1 Industrial Internet of Things (IIoT) 36
3.3 Δεδομένα καιρού & Συστήματα IoT 37

3.4 Αυτόματη ανάκτηση δεδομένων καιρού 37
3.5 Προεπεξεργασία δεδομένων 40
3.5.1 Χαμένες τιμές 41
3.5.2 Εξάλειψη Θορύβου 43
3.5.3 Κανονικοποίηση 45
3.6 Δεδομένα φορτίου 47

4. Πειράματα-Αποτελέσματα 51
4.1 Διαδικασία εκπαίδευσης 52
4.2 Επίδραση της προεπεξεργασίας των δεδομένων στην επίδοση 54
4.3 Επίδοση μοντέλων Μηχανικής Μάθησης 56
4.4 Επίδραση των δεδομένων στην ακρίβεια προβλέψεων 58
4.5 Αριθμητική βελτίωση μετεωρολογικής πρόγνωσης 59
4.6 Δεδομένα φορτίου ENTSO-E 61
4.7 Επίδραση γνώσης προηγούμενων ημερών στην απόδοση 64
4.8 Συνδυασμός δεδομένων φορτίου-καιρού 66
4.9 Επίδραση παραθύρου προηγούμενης γνώσης - μελλοντικής πρόβλεψης 68
4.10 Οπτικοποίηση Χρονοσειρών 71

5. Συμπεράσματα-Μελλοντικές εφαρμογές 75
5.1 Σύνοψη εργασίας και συμπερασμάτων 75
5.2 Μελλοντικές εφαρμογές (Future Work) 76

Βιβλιογραφία 77

Κατάλογος σχημάτων

1.1 Κατανομή των μοντέλων που χρησιμοποιούνται για την πρόβλεψη φορτίου στη βιβλιογραφία [Kust17]. 16

2.1 Η δομή ενός φυσικού νευρώνα του ανθρώπινου εγκεφάλου (**a**) σε σύγκριση με την μαθηματική δομή του τεχνητού νευρώνα (**b**) ο οποίος χρησιμοποιείται στα νευρωνικά δίκτυα. 21
2.2 Γραφική παράσταση της Σιγμοειδούς συνάρτησης. 22
2.3 Γραφική παράσταση της Υπερβολικής συνάρτησης. 22
2.4 Γραφική παράσταση της συνάρτησης ReLU. 23
2.5 Γραφική παράσταση της συνάρτησης ReLU σε σύγκριση με διάφορες περιπτώσεις της παραμετρικής ReLU ανάλογα με την τιμή της παραμέτρου α. 23
2.6 Η δομή ενός πλήρους συνδεδεμένου νευρωνικού δικτύου. 24
2.7 Μαθηματικές σχέσεις ενός κελιού RNN. 25
2.8 Το αναδρομικό δίκτυο, ”ξετυλιγμένο” στο χρόνο. 25
2.9 Γραφική απεικόνιση της εμπρόσθιας τροφοδότησης ενός αναδρομικού δικτύου. . . . 26
2.10 Απεικόνιση του περιθωρίου μεταξύ της χρήσιμης πληροφορίας και του σημείου αναγκαιότητάς της σε σειριακά δεδομένα. 29
2.11 Η δομή μιας πύλης αποτελούμενη από μια σιγμοειδή συνάρτηση σ που ελέγχει την ροή της πληροφορίας. 30
2.12 Η δομή κελιού ενός απλού αναδρομικού δικτύου (**a**) σε σύγκριση με την δομή του LSTM (**b**) 30
2.13 Forget πύλη του LSTM. 31
2.14 Store πύλη του LSTM. 31
2.15 Update πύλη του LSTM. 32
2.16 Output πύλη του LSTM. 32
2.17 Η δομή ενός αμφίδρομου LSTM [Yild18]. 33

3.1 Τα επίπεδα ενός IoT συστήματος. 36
3.2 Το script daily.sh 37
3.3 ER διάγραμμα της βάσης δεδομένων καιρού. 38
3.4 Γραφική παράσταση δεδομένων που ανακτήθηκαν. 39
3.5 Κομμάτι κώδικα για την διαχείρηση χαμένων τιμών και διπλοκαταχωρημένων εγγραφών με χρήση pandas-python. 42
3.6 Σήμα εισόδου θερμοκρασιών μιας εβδομάδας. Περιέχεται θόρυβος σε μορφή *spike noise* που παρατηρείται κυρίως στα σημεία καμπής. 43
3.7 Αποτελέσματα της αποθορυβοποίησης των δεδομένων εισόδου. Οι χρονοσειρές πριν (**a**) και μετά (**b**) την εφαρμογή. 44
3.8 Κώδικας για την εφαρμογή φίλτρων median και gaussian στα δεδομένα. 45
3.9 Κώδικας κανονικοποίησης των δεδομένων εισόδου. Γίνεται χρήση του MinMaxScaler της βιβλιοθήκης sklearn. 46
3.10 Χάρτης επίδρασης των δεδομένων καιρού ανά περιοχή. (**a**) για όλη την ηπειρωτική Ελλάδα και (**b**) για την περιφέρεια Θεσσαλίας. 48

3.11 Διάγραμμα ροής δεδομένων. Περιγράφεται σχηματικά η διαδικασία εκπαίδευσης των μοντέλων. 49
3.12 Διάγραμμα ροής δεδομένων. Περιγράφεται συλλογικά η διαδικασία απόκτησης, επεξεργασίας δεδομένων, εκπαίδευσης μοντέλων και εξαγωγής συμπερασμάτων. 49

4.1 Διάγραμμα ροής δεδομένων. Περιγράφεται σχηματικά η διαδικασία διεξαγωγής των πειραμάτων. 51
4.2 Κώδικας δημιουργίας των δεδομένων εκπαίδευσης σε μορφή εισόδων-αναμενόμενης εξόδου. 52
4.3 Δείγμα της μορφής των δεδομένων εκπαίδευσης του μοντέλου. Χωρίζονται σε δεδομένα εισόδου και αναμενόμενης εξόδου. 53
4.4 Κώδικας κατασκευής ενός μοντέλου LSTM. Χρησιμοποιείται η βιβλιοθήκη μηχανικής μάθησης keras. 53
4.5 Διαγράμματα ανάλυσης επίδοσης με εξάλειψη θορύβου. 54
4.6 Διαγράμματα ανάλυσης επίδοσης χωρίς εξάλειψη θορύβου. 55
4.7 Σύγκριση μοντέλων βαθιάς μάθησης LSTM, Stacked LSTM, Bidirectional LSTM και CNN. 57
4.8 Γραφική παράσταση των δεδομένων. Με μαύρο παριστάνονται οι πραγματικές μετρήσεις και με πράσινο οι προγνώσεις. 59
4.9 Παράδειγμα εισόδων (πραγματικές παρελθοντικές τιμές και προγνώσεις), πρόβλεψης του δικτύου και πραγματικής μελλοντικής μέτρησης. 60
4.10 Παράδειγμα εισόδων (πραγματικές παρελθοντικές τιμές και προγνώσεις), πρόβλεψης του δικτύου και πραγματικής μελλοντικής μέτρησης σε δεδομένα φορτίου. Παρουσιάζεται και η απόλυτη διαφορά των χρονοσειρών παρατήρησης-πρόβλεψης (κόκκινο) και παρατήρησης-πρόγνωσης (κίτρινο). 62
4.11 Μεταβολή του μέσου τετραγωνικού σφάλματος: (**a**) MSE του διανύσματος εξόδου, μήκους ίσο με αυτό του διανύσματος εισόδου. (**b**) MSE μέρους του διανύσματος εξόδου, μόνο της πρώτης ημέρας. (**c**) Τα δύο σφάλματα σε κοινό διάγραμμα. 65
4.12 Παράδειγμα εισόδων (παρελθοντικές τιμές) του φορτίου και της σταθμισμένης θερμοκρασίας. Τα δύο σήματα αυτά περιέχουν και τιμές πρόγνωσης μετά τη διακεκομμένη γραμμή. Με κόκκινο παρουσιάζεται η πραγματική μέτρηση (αναμενόμενη τιμή). . . 66
4.13 Μεταβολή του μέσου τετραγωνικού σφάλματος για όλα τα πιθανά ”σενάρια” παραθύρων παρελθοντικών και μελλοντικών τιμών. 68
4.14 Μεταβολή του μέσου τετραγωνικού σφάλματος των καλύτερων μοντέλων. 68
4.15 Χρονοσειρές πρόβλεψης για τα μοντέλα 5 παρελθοντικών ημερών. 69
4.16 Bar plot σφαλμάτων όλων των μοντέλων (**a**) για τις αντίστοιχες ημέρες ζητούμενης πρόβλεψης (**b**) μόνο για την πρώτη ημέρα. 70
4.17 Στιγμιότυπα του εργαλείου οπτικοποίησης. 71
4.18 Οπτικοποίηση προβλέψεων για τα μοντέλα 5 και 7 ημερών. 72
4.19 Οπτικοποίηση προβλέψεων για τα μοντέλα 14 και 30 ημερών. 73
4.20 Οπτικοποίηση περιπτώσεων όπου **(a)** η πρόβλεψή μας δεν είναι καλή και **(b)** η πρόβλεψή μας είναι πολύ καλή, καλύτερη και από την πρόγνωση ENTSO-E. 74

Κεφάλαιο 1

Εισαγωγή

1.1 Σκοπός της εργασίας

Σκοπός της παρούσας εργασίας είναι η ανάλυση και επεξεργασία ενεργειακών και μετεωρολογικών δεδομένων σε πανελλαδικό επίπεδο για την ανάπτυξη προβλέψεων με την χρήση τεχνικών μηχανικής μάθησης.

Μελετάται η διαχείριση των δεδομένων, αρχικά με την απόκτηση και την αποθήκευσή τους από πολλαπλές πηγές (βάσεις, συσκευές IoT, API). Έπειτα αναλύεται η προεπεξεργασία δεδομένων δίνοντας έμφαση στην αποθορυβοποίηση και στην κανονικοποίηση των δεδομένων, πράγμα που οδηγεί σε καλύτερη απόδοση τους αλγόριθμους μηχανικής μάθησης.

Ιδιαίτερη σημασία δίνεται στον πειραματισμό με διάφορα μοντέλα μηχανικής μάθησης, την εκτίμησή τους καθώς και τον τρόπο με τον οποίο λειτουργούν σε θεωρητικό επίπεδο. Επιπρόσθετα, διεξάγονται δοκιμές με διάφορα σύνολα χρονοσειρών και εκτιμάται η επίδρασή τους στην επίδοση. Εκτός από τις προβλέψεις, έμφαση δίνεται και στο πρόβλημα της βελτίωσης των προγνώσεων που δίνονται από τις πηγές παροχής δεδομένων, καλύτερη απόδοση επιτυγχάνεται με την προσθήκη αυτών των προγνώσεων ως επιπρόσθετη πληροφορία στις εισόδους των μοντέλων. Τέλος, αναπτύσσεται και χρησιμοποιείται εργαλείο οπτικοποίησης των χρονοσειρών σε φιλικό προς τον χρήστη περιβάλλον.

1.2 Δομή της εργασίας

Στο δεύτερο κεφάλαιο ορίζονται οι βασικές τεχνικές μηχανικής μάθησης (machine learning) που χρησιμοποιούνται, σε θεωρητικό επίπεδο. Αναλύονται και συγκρίνονται αλγόριθμοι, τεχνικές και αρχιτεκτονικές δικτύων.

Στο τρίτο κεφάλαιο παρουσιάζονται οι τρόποι απόκτησης δεδομένων καθώς και η αποθήκευσή τους σε βάσεις δεδομένων. Επίσης περιγράφονται τόσο σε θεωρητικό όσο και σε πρακτικό επίπεδο (υλοποίηση κώδικα) οι τεχνικές αποθορυβοποίησης και κανονικοποίησης των δεδομένων, στα πλαίσια προεπεξεργασίας τους για την μετέπειτα χρήση.

Στο τέταρτο κεφάλαιο παραθέτονται τα αποτελέσματα των πειραμάτων που εκτελέστηκαν. Παρουσιάζονται διαγράμματα, κώδικας και μετρικές. Συγκρίσεις γίνονται ως προς την ποιότητα και το πλήθος των δεδομένων, την αρχιτεκτονική των δικτύων, την βελτίωση ήδη υπαρχόντων προγνώσεων, καθώς και για την επίδραση του πλήθους των ζητούμενων ημερών στην ποιότητα της πρόβλεψης.

Η εργασία ολοκληρώνεται με το τελευταίο κεφάλαιο, όπου γίνεται συζήτηση για πιθανές μελλοντικές εφαρμογές, καθώς παρουσιάζονται περιληπτικά συμπεράσματα.

1.3 Τοποθέτηση του προβλήματος

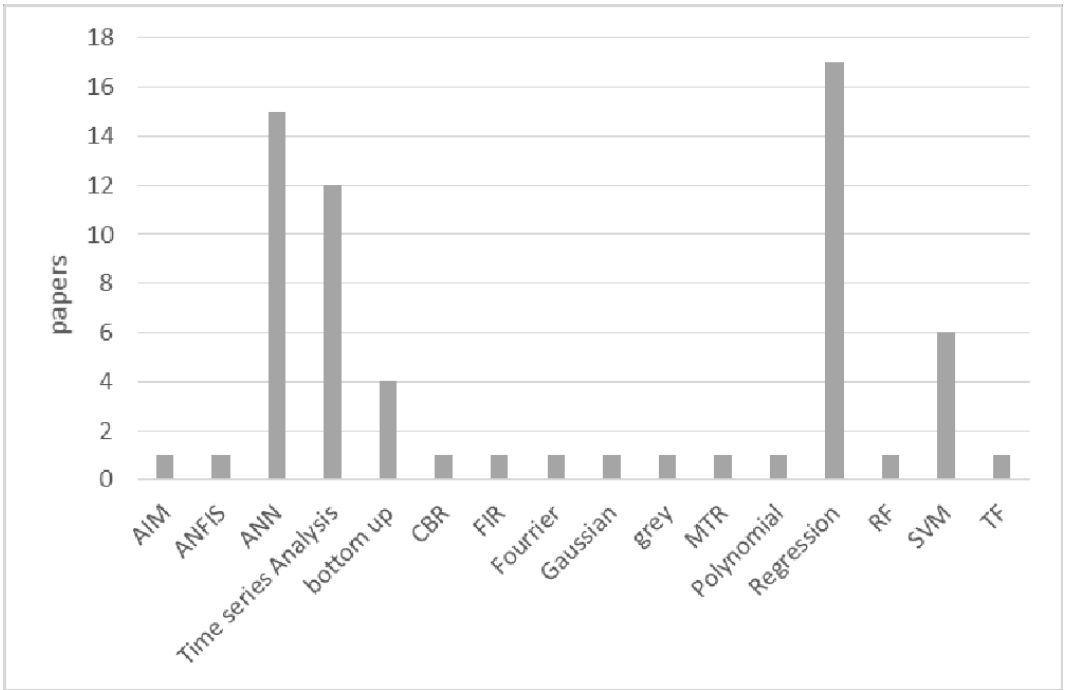
Η πρόβλεψη φορτίου είναι ένα βασικό εργαλείο σε διάφορες εφαρμογές του τομέα της ενέργειας και αποκτά μεγαλύτερη σημασία στις απελευθερωμένες αγορές ηλεκτρικής ενέργειας. Οι χρονοσειρές φορτίου επηρεάζονται από πολλούς παράγοντες που περιλαμβάνουν καιρικές συνθήκες, κοινωνικό-οικονομικές πτυχές καθώς και τυχαίες παραμέτρους [Hahn09, Sobh19]. Αυτές οι εξαρτήσεις εισάγουν μη-γραμμικά και ασταθή χαρακτηριστικά στις ζητούμενες χρονοσειρές. Οπότε η υψηλή απόδοση προβλέψεων που απαιτείται από τους διαχειριστές συστημάτων ηλεκτρικής ενέργειας, τους λειτουργούς και τους συμμετέχοντες στις αγορές, καθιστούν την πρόβλεψη φορτίου εξαιρετικά δύσκολη, ακόμα και για βραχυπρόθεσμα διαστήματα.

Οι προβλέψεις φορτίου είναι σημαντικές για μια αξιόπιστη και οικονομικά σταθερή λειτουργία ενός συστήματος ισχύος, όπως μια επιχείρηση παροχής ηλεκτρικής ενέργειας [Dela08]. Μια εκτίμηση φορτίου πάνω από την πραγματική, υποχρεώνει τους Διαχειριστές Μεταφοράς Ηλεκτρικής Ενέργειας (ΑΔΜΗΕ) να δεσμεύουν ακριβότερες μονάδες παραγωγής ενέργειας οδηγώντας σε υψηλότερες τιμές στην αγορά ενέργειας.

Από την άλλη, η υποτίμηση της ζήτησης φορτίου μπορεί να θέσει την ασφαλή λειτουργία του συστήματος σε κίνδυνο και να επιβάλλει την απρόβλεπτη εκκίνηση ακριβών μονάδων σε σύντομο χρονικό διάστημα. Επιπλέον, η αξιοπιστία των συστημάτων ισχύος διασφαλίζεται με την ανάλυση του δικτύου βάσει προβλέψεων φορτίου. Σε περιβάλλοντα με πολιτική ανοικτής αγοράς, ο καθορισμός της οριακής τιμής για άμεση παράδοση (spot price) και η χονδρική εμπορία ηλεκτρικής ενέργειας, επηρεάζεται σημαντικά από τη ζήτηση φορτίου.

Οι ακριβείς προβλέψεις φορτίου είναι ζωτικής σημασίας για την εγγύηση ενεργειακών συναλλαγών καθώς και για να διατηρηθούν τα μερίδια αγοράς για τους συμμετέχοντες, "παίκτες", μιας ανταγωνιστικής αγοράς ηλεκτρικής ενέργειας. Για αυτό το λόγο, οι Διαχειριστές Μεταφοράς Ηλεκτρικής Ενέργειας συνήθως επιβάλλουν κυρώσεις σε προμηθευτές ενέργειας με υψηλά σφάλματα πρόβλεψης φορτίου [Dela08].

Αρκετές έρευνες έχουν επικεντρωθεί στην αύξηση της απόδοσης μοντέλων πρόβλεψης φορτίου [Fall18, Hern14b]. Ιδιαίτερα κατευθύνονται σε λύσεις βασισμένες στη μηχανική μάθηση [Alma17, Side20] καθώς και σε στατιστικά μοντέλα ανάλυσης χρονοσειρών [Cont03].



Σχήμα 1.1: Κατανομή των μοντέλων που χρησιμοποιούνται για την πρόβλεψη φορτίου στη βιβλιογραφία [Kust17].

Σημαντική έρευνα έχει γίνει και για τον συνδυασμό διαφορετικών μοντέλων, δημιουργώντας υβριδικά δίκτυα. Καλύτερη απόδοση έχει επιτευχθεί συνδυάζοντας διαφορετικές μεθόδους τεχνητής νοημοσύνης, όπως ομαδοποίηση δεδομένων (data clustering) με εποπτευόμενη ή και μη εποπτευόμενη εκπαίδευση νευρωνικών δικτύων [Hern14a]. Για παράδειγμα, στο [Side20] προτείνεται η χρήση ενός "fuzzy clustering" επιπέδου, το οποίο κατηγοριοποιεί τα δεδομένα εισόδου σε πολλαπλές ομάδες. Έπειτα ένα δίκτυα συναρτήσεων πυρήνα (RBF) μετασχηματίζει τα δεδομένα εισόδου σε υψηλότερες διαστάσεις. Τέλος ένα συνελικτικό δίκτυο (CNN) δέχεται τις νέες αναπαραστάσεις δεδομένων και εκπαιδεύεται ώστε να παράγει την τελική πρόβλεψη.

Μελέτες έχουν γίνει και για την ανάλυση της επίδρασης εξωτερικών παραγόντων στην κατανάλωση/παραγωγή φορτίου. Οι καιρικές συνθήκες είναι βασικός παράγοντας που επηρεάζει τη ζήτηση ηλεκτρικής ενέργειας. Πολλά μοντέλα πρόβλεψης φορτίου βασίζονται σε παραμέτρους καιρού. Οι μετεωρολογικοί σταθμοί παρέχουν σημειακές μετρήσεις των καιρικών συνθηκών μιας περιοχής. Δεδομένου ότι το φορτίο κατανέμεται γεωγραφικά, ένας μετεωρολογικός σταθμός μπορεί να μην επαρκεί για την ανάλυση των παραλλαγών του φορτίου σε μια μεγαλύτερη περιοχή. Για αυτό το λόγο, μπορεί να έχει μεγάλη σημασία ο ορθός συνδυασμός της πληροφορίας με βάση τη γεωγραφική κατανομή [Sobh19].

Τις τελευταίες δεκαετίες, πολλαπλά μοντέλα πρόβλεψης που περιλαμβάνουν μεταβλητές καιρού έχουν αναπτυχθεί και δοκιμαστεί [Hipp01, Jing18, Xie18b]. Η πλειοψηφία των εγγράφων στη βιβλιογραφία πρόβλεψης φορτίου, μελετούν μεθοδολογίες πρόβλεψης φορτίου σε επίπεδα υψηλής τάσης [Fan12, Nowo16]. Η ανάπτυξη και εφαρμογή τεχνολογιών έξυπνου δικτύου (smart grid) τα τελευταία χρόνια, με τη διάδοση των ανανεώσιμων πηγών ενέργειας και των ηλεκτροκίνητων οχημάτων, απαιτεί νέες λύσεις όπως η πιθανοτική πρόβλεψη φορτίου και έχει δημιουργήσει την ανάγκη για ακριβείς προβλέψεις σε χαμηλά/μεσαία επίπεδα τάσης [Hynd10, Wang19, Quil15].

Ο καιρός είναι ίσως ο σημαντικότερος παράγοντας που καθορίζει τη μη-βιομηχανική κατανάλωση ηλεκτρικής ενέργειας. Μοντέλα βασισμένα σε μετεωρολογικά δεδομένα χρησιμοποιούνται συχνά στη πρόβλεψη ηλεκτρικού φορτίου. Φορείς που προβαίνουν σε προβλέψεις, έχουν χρησιμοποιήσει στο παρελθόν τη σχέση μεταξύ προφίλ καιρού και φορτίου περιοχών για την ανάπτυξη μοντέλων. Αν και η θερμοκρασία είναι το βασικό σήμα που χρησιμοποιείται συχνότερα, άλλα μεγέθη όπως η υγρασία και η ταχύτητα του ανέμου μπορούν επίσης να προσφέρουν σημαντική πληροφορία [Xie18a, Xie17]. Τα καιρικά δεδομένα προέρχονται κυρίως από τις παρατηρήσεις μετεωρολογικών σταθμών. Ενώ πολλοί δημόσιοι πάροχοι δεδομένων και ιδιωτικοί προμηθευτές λαμβάνουν δεδομένα από διαφορετικούς μετεωρολογικούς σταθμούς για να εξυπηρετήσουν πελάτες στη βιομηχανία παροχής ενέργειας, η διαθεσιμότητα και η ποιότητα των καιρικών δεδομένων είναι ανησυχητική για πολλές εταιρείες ηλεκτρικής ενέργειας [Hong10].

Τα όργανα ενός μετεωρολογικού σταθμού συνήθως συλλέγουν τις πληροφορίες από μία περιορισμένη γεωγραφική περιοχή. Τα δεδομένα αυτά όπως η θερμοκρασία ή η υγρασία αντικατοπτρίζουν την καιρική συμπεριφορά μιας συγκεκριμένης τοποθεσίας. Από την άλλη πλευρά, το φορτίο μπορεί να κατανέμεται σε μια τεράστια περιοχή μιας ζώνης εξυπηρέτησης πελατών. Σε ένα ηλεκτρικό δίκτυο παροχής ενέργειας, καθώς προχωράμε σε υψηλότερα επίπεδα ιεραρχίας φορτίου, το συνολικό φορτίο καλύπτει όλο και μεγαλύτερη γεωγραφική περιοχή. Επομένως, οι ενδείξεις ενός μετεωρολογικού σταθμού ενδέχεται να μην αντικατοπτρίζουν επαρκώς τις διακυμάνσεις φορτίου σε μια μεγάλη γεωγραφική περιοχή.

Στην παρούσα εργασία, εξετάζεται η δημιουργία και χρήση ενός μοναδικού χωρικά κανονικοποιημένου δείκτη θερμοκρασίας ο οποίος εξαρτάται από τις μετρήσεις πολλαπλών σταθμών στις μεγαλύτερες πόλεις της Ελλάδας. Οι τιμές αυτές σταθμίζονται σε σχέση με τη γεωγραφική τους τοποθεσία χρησιμοποιώντας, τα δημογραφικά στοιχεία της εκάστοτε περιοχής. Ο δείκτης αυτός χρησιμοποιείται στη συνέχεια για τον υπολογισμό προβλέψεων κατανάλωσης ηλεκτρικής ενέργειας σε πανελλαδικό επίπεδο.

Η κατανάλωση ενέργειας σε επίπεδο κρατών έχει εξίσου μεγάλη σημασία. Η ενέργεια παίζει ζωτικό ρόλο στην πρόοδο και την κοινωνικοοικονομική ανάπτυξη μιας χώρας. Διάφορες μελέτες και έρευνες συσχετίζουν άμεσα την κατανάλωση ενέργειας με την οικονομική, τεχνολογική και κοινω-

νική ανάπτυξη της χώρας [Oztu10, Payn10, Mahm14]. Η ζήτηση ηλεκτρικής ενέργειας αυξάνεται εκθετικά και οι διαθέσιμοι πόροι εξαντλούνται με ανησυχητικό ρυθμό. Επομένως, είναι απαραίτητη η κατάλληλη διαχείριση των πηγών ενέργειας για τη βελτιστοποίηση της χρήσης τους, την ελαχιστοποίηση του κόστος παραγωγής και την προστασία από περιβαλλοντικούς κινδύνους.

Εκτός από τα δεδομένα εκπαίδευσης, η απόδοση των μοντέλων εξαρτάται και από τον ορίζοντα πρόβλεψης, δηλαδή τις ζητούμενες ημέρες πρόβλεψης [Bouk18]. Δύο προκλήσεις σχετίζονται με τους στοχευμένους ορίζοντες πρόβλεψης. Στη βραχυπρόθεσμη περίπτωση, η ακρίβεια είναι ζωτικής σημασίας για τη βέλτιστη καθημερινή λειτουργική απόδοση της παροχής ηλεκτρικής ενέργειας. Στη μεσοπρόθεσμη περίπτωση, η σταθερότητα πρόβλεψης απαιτείται για τον ακριβή προγραμματισμό των προμηθειών καυσίμων και για τις έγκαιρες εργασίες συντήρησης. Για σταθερότητα προβλέψεων, πρέπει να διατηρηθεί ένα χαμηλό σφάλμα πρόβλεψης στο μεσοπρόθεσμο διάστημα. Έτσι, το μοντέλο πρόβλεψης πρέπει να συνεχίζει να λειτουργεί με ακρίβεια ή τουλάχιστον δεν θα πρέπει να είναι υπερβολικά ευαίσθητο στον χρόνο εντός του μεσοπρόθεσμου πλαισίου. Έχει λοιπόν ιδιαίτερο ενδιαφέρον και η μελέτη της απόδοσης των μοντέλων σε βραχυπρόθεσμα και μεσοπρόθεσμα "παράθυρα" ζητούμενης πρόβλεψης.

Συνοψίζοντας, σε αυτή την εργασία, αναλύονται μέθοδοι μηχανικής μάθησης για την πρόβλεψη της συμπεριφοράς του φορτίου σε πανελλαδικό επίπεδο. Μελετάται η επίδραση εξωτερικών παραγόντων (μετεωρολογικά δεδομένα), συνδυάζονται παρελθοντικές μετρήσεις καθώς και υπάρχουσες προγνώσεις για την βελτίωση της πρόβλεψης.

Κεφάλαιο 2

Μηχανική Μάθηση

Η Μηχανική Μάθηση (Machine Learning - ML), είναι ένα πεδίο της επιστήμης των υπολογιστών και της τεχνητής νοημοσύνης που στοχεύει στην δημιουργία συστημάτων ικανών να μαθαίνουν αυτόματα από κάποια γνώση και να κάνουν προβλέψεις με βάση αυτή. Το σύστημα βελτιώνει σταδιακά την απόδοσή του κατά την εκτέλεση μιας συγκεκριμένης εργασίας, χωρίς την ανάγκη ανθρώπινης επίβλεψης ή επιπλέον προγραμματισμού. Αυτό επιτυγχάνεται μέσω υπολογιστικών μοντέλων, τα οποία μαθαίνουν να αναγνωρίζουν πρότυπα από ένα σύνολο δεδομένων εκπαίδευσης. Ανάλογα με την εφαρμογή, αυτά τα δεδομένα μπορεί να είναι εικόνες, κυματομορφές, κείμενο ή οποιαδήποτε άλλη μορφή πληροφορίας μπορεί να ερμηνευθεί από έναν υπολογιστή. Οι πιο διαδεδομένες μέθοδοι ML σχετίζονται με την μορφή των δεδομένων εισόδου. Πιο συγκεκριμένα, αν τα δεδομένα εισόδου συνοδεύονται από επισημασμένα δεδομένα εξόδου, ετικέτες(labels), τότε λέμε ότι έχουμε επιβλεπόμενη μάθηση. Ο στόχος είναι το μοντέλο να μάθει από την διαφορά της πρόβλεψης και της πραγματικής τιμής της ετικέτας. Όταν τα δεδομένα δεν έχουν επισήμανση, ακολουθείτε μη επιβλεπόμενη μάθηση όπου, μέσω κατάλληλων αλγορίθμων, τα δεδομένα ομαδοποιούνται σε κατηγορίες ή αναζητάτε μια δομή αυτών. Βεβαίως υπάρχει και το ενδιάμεσο στάδιο, όπου έχουμε ετικέτες για μέρος των δεδομένων. Τότε η κύρια μέθοδος είναι αυτή της ημι-επιβλεπόμενης μάθησης.

2.1 Διαδικασίες Μάθησης

Όπως οι άνθρωποι, έτσι και τα τεχνητά συστήματα με νοημοσύνη, έχουν διαφορετικούς τρόπους μάθησης. Οι τρόποι αυτοί ανήκουν συνήθως σε μία από τις παρακάτω κατηγορίες: μάθηση με εκπαιδευτή (ή επιβλεπόμενη μάθηση), μάθηση χωρίς εκπαιδευτή. Αυτές οι μορφές μάθησης, υλοποιούνται υπολογιστικά με διάφορα μοντέλα και στοχεύουν στην προσομοίωση της ανθρώπινης μάθησης.

2.1.1 Επιβλεπόμενη Μάθηση

Στην επιβλεπόμενη μάθηση (ή μάθηση χωρίς εκπαιδευτή), για κάθε διάνυσμα εισόδου, υπάρχει ένα επιθυμητό διάνυσμα εξόδου. Αυτό είναι γνωστό στον εκπαιδευτή αλλά όχι και στο σύστημα μάθησης. Οπότε ο στόχος είναι να βρεθεί (από το σύστημα μάθησης) μια συνάρτηση απεικόνισης f τέτοια ώστε $f(x) = y$, μια συνάρτηση δηλαδή που θα προσεγγίζει όσο το δυνατόν καλύτερα την επιθυμητή έξοδο, δεδομένης μιας οποιασδήποτε εισόδου. Η διαφορά μεταξύ της επιθυμητής τιμής και της πραγματικής απόκρισης του δικτύου, λέγεται σήμα σφάλματος και με βάση αυτήν οφείλουν να μεταβληθούν βέλτιστα οι παράμετροι του συστήματος. Η διαδικασία με την οποία προσεγγίζεται αυτή η συνάρτηση απεικόνισης, διαφέρει ανάλογα με το μοντέλο που χρησιμοποιείται όμως η κύρια διαδικασία που ακολουθείται, είναι η κατ' επανάληψη σύγκριση των προβλέψεων του μοντέλου με τις πραγματικές τιμές και η διόρθωση των παραμέτρων του μοντέλου με βάση αυτήν. Η εκπαίδευση σταματάει όταν το μοντέλο επιτυγχάνει ικανοποιητική ακρίβεια στις προβλέψεις του, τόσο στα δεδομένα εισόδου, αλλά και σε δεδομένα που δεν έχουν εμφανιστεί κατά την διαδικασία της εκπαίδευσης (validation data). Έτσι διασφαλίζεται ότι η γνώση που αποκτάται κατά την εκπαίδευση μπορεί να γενικευθεί και δικαιολογείται η χρησιμότητα της συνάρτησης απεικόνισης.

Τα προβλήματα επιβλεπόμενης μάθησης, χωρίζονται σε προβλήματα ταξινόμησης (classification) και παλινδρόμησης (regression) .

- Ταξινόμηση: Γίνεται ανάθεση των δεδομένων σε διακριτές κατηγορίες ή κλάσεις. Σε αυτές τις περιπτώσεις το y είναι διακριτό. Ένα παράδειγμα είναι, με βάση μια εικοσιτετράωρη χρονοσειρά τιμών θερμοκρασίας να προβλεφθεί αν θα βρέξει την επόμενη ημέρα (y = 0 ή 1, δυαδική ταξινόμηση).

- Παλινδρόμηση: Προβλέπεται μια συνεχής ποσότητα. Για παράδειγμα, με βάση την εικοσιτετράωρη χρονοσειρά θερμοκρασίας, να προβλεφθεί ή πιθανότητα εκδήλωσης βροχής την επόμενη ημέρα (y ανήκει στο 0,1 συνεχές).

2.1.2 Μάθηση χωρίς εκπαιδευτή

Στην επιβλεπόμενη μάθηση, υπάρχει η καθοδήγηση του εκπαιδευτή που καθοδηγεί το μοντέλο στον σωστό δρόμο ώστε να αποκτήσει γνώση. Υπάρχουν δηλαδή ζεύγη δεδομένων εισόδου και επιθυμητής εξόδου. Όμως, αυτό δεν συμβαίνει πάντα. Σε ένα παιχνίδι σκακιού, δεν μπορούμε να καθορίσουμε αν μια κίνηση είναι καλή, αλλά ούτε και να έχουμε έναν αλάνθαστο δείκτη κρίσης της κάθε κίνησης. Οπότε ένα σύστημα εκμάθησης σκακιού, δεν θα μπορούσε να αναπτυχθεί χρησιμοποιώντας επιβλεπόμενη μάθηση. Ανάλογα με το αν υπάρχει κάποιος κριτής ή όχι, η μάθηση χωρίς εκπαιδευτή χωρίζεται σε: Ενισχυτική ,και Μη Επιβλεπόμενη, Μάθηση.

2.1.2.1 Ενισχυτική Μάθηση

Στην ενισχυτική μάθηση, η βέλτιστη κατεύθυνση του μοντέλου υπολογίζεται μέσω συνεχούς αλληλεπίδρασης με το περιβάλλον στοχεύοντας στην ελαχιστοποίηση ενός βαθμωτού δείκτη απόδοσης. Το σύστημα παρακολουθεί μια χρονική ακολουθία ερεθισμάτων που λαμβάνει από το περιβάλλον, τα οποία συνδυάζονται σε ένα καθολικό σήμα, αυξάνοντας την ποιότητα των επιμέρους ερεθισμάτων. Παράδειγμα ερεθισμάτων μπορεί να είναι οι πόντοι κάθε πίστας σε κάποιο παιχνίδι υπολογιστή καθώς και η "ζωή" του παίκτη. Ο στόχος της ενισχυτικής μάθησης, είναι η ελαχιστοποίηση μιας συνάρτησης τρέχοντος κόστους η οποία εξαρτάται από τα επιμέρους κόστη των ενεργειών που εκτελούνται.

2.1.2.2 Μη Επιβλεπόμενη Μάθηση

Στην μη επιβλεπόμενη μάθηση, δεν υπάρχει εξωτερικός εκπαιδευτής ούτε κριτής για να προσανατολίσει την διαδικασία της μάθησης. Οι παράμετροι του δικτύου αλλάζουν ανάλογα με ένα μέτρο ανεξάρτητο από την εργασία. Το δίκτυο προσπαθεί να "μάθει" στατιστικές ιδιότητες των δεδομένων εισόδου με στόχο την κατηγοριοποίηση τους σε κλάσεις. Η κατηγοριοποίηση γίνεται αυτόματα, κωδικοποιώντας τα δεδομένα ανάλογα με τα χαρακτηριστικά τους.

2.2 Νευρωνικά Δίκτυα

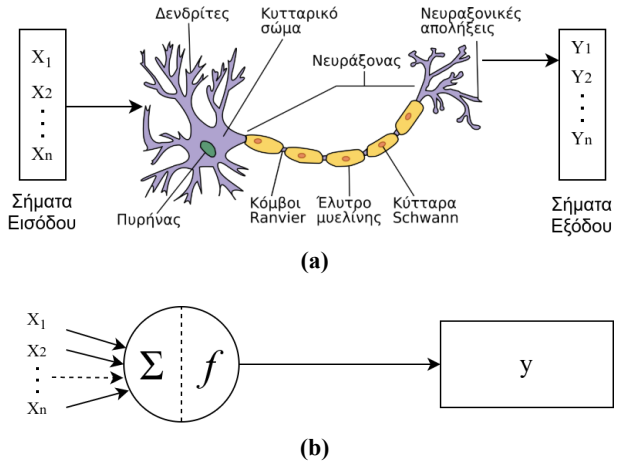
Η πιο διαδεδομένη προσέγγιση για την επίλυση προβλημάτων επιβλεπόμενης μάθησης, είναι η χρήση νευρωνικών δικτύων. Πρόκειται για ένα κύκλωμα διασυνδεδεμένων τεχνητών νευρώνων σε δομή γράφου. Δέχονται πολλαπλές εισόδους και μπορούν να παράγουν πολλαπλές προβλέψεις στην έξοδο. Τα νευρωνικά δίκτυα, είναι εμπνευσμένα από το Κεντρικό Νευρικό Σύστημα και προσπαθούν να προσομοιώσουν τις λειτουργίες του ανθρώπινου εγκεφάλου. Οι βασικές ιδιότητες των δικτύων, έχουν ερευνηθεί και προταθεί από την δεκαετία του 1940 από νευροφυσιολόγους, ψυχολόγους και μαθηματικούς.

2.2.1 Το μοντέλο του τεχνητού Νευρώνα

Η βασική μονάδα από την οποία αποτελείται το δίκτυο είναι οι νευρώνες. Κάθε νευρώνας δέχεται στην είσοδό του κάποια σήματα εισόδου $X = \{x_1, x_2, \ldots, x_n\}$. Κάθε στοιχείο x_i, του διανύσματος X, πολλαπλασιάζεται με ένα βάρος w_i ενός πίνακα βαρών W. Η λειτουργία αυτή προσομοιώνει τον ρόλο της σύναψης στον βιολογικό νευρώνα. Έπειτα όλα τα αποτελέσματα περνούν από έναν

αθροιστή, ο οποίος προσθέτει τα μεταβλημένα, από τα βάρη, σήματα εισόδου. Επίσης προστίθεται ένας όρος πόλωσης b ο οποίος συμβάλλει στην οριζόντια μετατόπιση του τελικού αποτελέσματος. Αυτό το βήμα μπορεί να αντικατασταθεί ώς μια επιπλέον είσοδος $x_0 = 1$ με κάποιο βάρος w_0. Το αποτέλεσμα, περνάει από μια συνάρτηση ενεργοποίησης (activation function) f η οποία δρα ως ένα φίλτρο που διαμορφώνει την τελική απόκριση του νευρώνα, περιορίζοντάς την συνήθως σε κάποιο διάστημα $[0, 1]$ ή $[-1, 1]$.

Τα παραπάνω διατυπώνονται μαθηματικά με την πολύ απλή έκφραση $y = f(X \cdot W + b)$, πρακτικά οι νευρώνες περιορίζουν τον χώρο συναρτήσεων, στον οποίο αναζητείται η συνάρτηση απεικόνισης f, στην γραμμική περίπτωση. Η ονομασία καθώς και η λογική προκύπτει από την αναλογία με την νευροφυσιολογία του εγκεφάλου. Όπως φαίνεται και στο Σχ. 2.1, η μαθηματική δομή του νευρώνα μοιάζει με την φυσική δομή ενός νευρώνα του εγκεφάλου.



Σχήμα 2.1: Η δομή ενός φυσικού νευρώνα του ανθρώπινου εγκεφάλου (**a**) σε σύγκριση με την μαθηματική δομή του τεχνητού νευρώνα (**b**) ο οποίος χρησιμοποιείται στα νευρωνικά δίκτυα.

2.2.2 Συναρτήσεις ενεργοποίησης

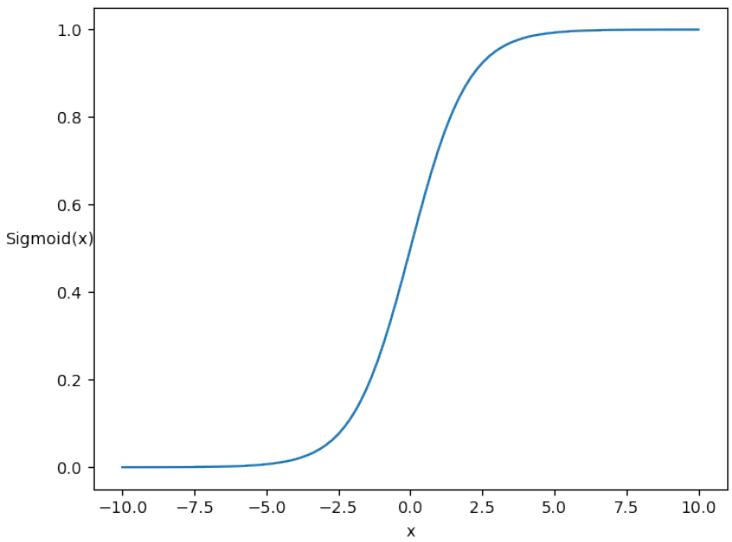
Η συνάρτηση ενεργοποίησης, στοχεύει κυρίως στην εισαγωγή μιας μη-γραμμικότητας (non-linearity) στο σύστημα, καθώς και στην φραγή των εξόδων των νευρώνων σε χαμηλά διαστήματα.

Η μη-γραμμικότητα είναι πολύ σημαντική αφού αν είχαμε μια γραμμική συνάρτηση της μορφής $f(x) = a \cdot x + b$, τότε ένα δίκτυο πολλαπλών επιπέδων οσηδήποτε μεγάλης πολυπλοκότητας , θα μπορούσε να αποτυπωθεί με την χρήση ενός μόνο επιπέδου, συνδυάζοντας σωστά τις γραμμικές συναρτήσεις ενεργοποίησης. Επίσης, τα περισσότερα προβλήματα στον πραγματικό κόσμο είναι μη-γραμμικά, οπότε μια μη-γραμμική συνάρτηση ενεργοποίησης, μπορούμε να υποθέσουμε πως εισάγει την απαραίτητη πολυπλοκότητα στο σύστημα, ώστε να μπορεί να κάνει προβλέψεις, για ένα εξίσου πολύπλοκο πρόβλημα.

Η επιλογή μιας συνάρτησης ενεργοποίησης, μπορεί να γίνει ανάλογα με το πρόβλημα (ταξινόμηση, παλινδρόμηση) αλλά και ανάλογα με την αποτελεσματικότητα, σύμφωνα με προηγούμενα αποτελέσματα από έρευνες. Οι πιο δημοφιλείς οικογένειες συναρτήσεων ενεργοποίησης είναι οι εξής:

2.2.2.1 Sigmoid or Logistic

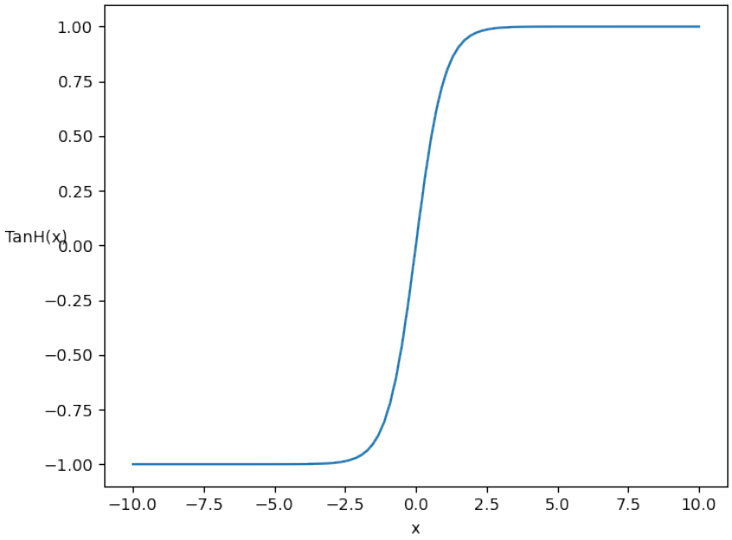
Οι σιγμοειδής συναρτήσεις, παίρνουν το όνομά τους από την γραφική αναπαράστασή τους. Ορίζονται ως $f(x) = \frac{1}{1+e^{-x}}$ και φράσουν την είσοδο στο διάστημα [0,1]. Αυτός είναι και ο λόγος που χρησιμοποιούνται συχνά στην δυαδική ταξινόμηση (binary classification) αφού η έξοδος μπορεί να ερμηνευθεί ως πιθανότητα, να ανήκει το δείγμα σε μια από τις δύο κλάσεις.



Σχήμα 2.2: Γραφική παράσταση της Σιγμοειδούς συνάρτησης.

2.2.2.2 TanH or hyperbolic tangent

Μπορεί να θεωρηθεί σιγμοειδής, αφού έχει παρόμοια λειτουργία. Όμως δίνει αποτελέσματα στο [-1,1]. Το πλεονέκτημα είναι ότι, αρνητικές τιμές εισόδου θα δώσουν αρνητική έξοδο και θετικές τιμές εισόδου θετική έξοδο αντίστοιχα. Χρησιμοποιείται συχνά για ταξινόμηση.
Ορίζεται ως $f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$



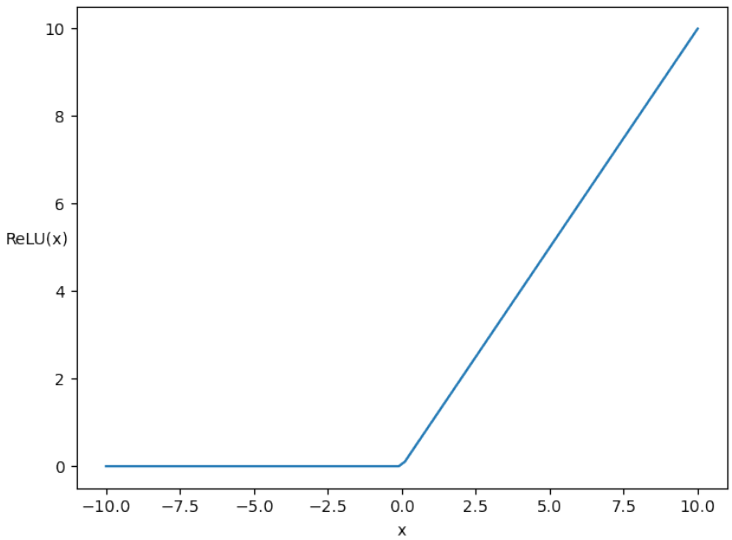
Σχήμα 2.3: Γραφική παράσταση της Υπερβολικής συνάρτησης.

2.2.2.3 Softmax

Περιγράφεται ως συνδυασμός πολλών σιγμοειδών συναρτήσεων. Χρησιμοποιείται για την ταξινόμηση πολλών κλάσεων αφού δέχεται στην είσοδο ένα διάνυσμα τιμών **z** και τις κανονικοποιεί, επιστρέφοντας μια κατανομή πιθανοτήτων. Αυτές οι κατανομές αντιπροσωπεύουν την πιθανότητα να ανήκει σε κάθε επιμέρους κλάση ένα δείγμα εισόδου. Η συνάρτηση Softmax ορίζεται μαθηματικά από τον παρακάτω τύπο: $softmax(\mathbf{z}) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$ για $i = 1, \ldots, K$ και $\mathbf{z} = \{z_1, \ldots, z_K\}$

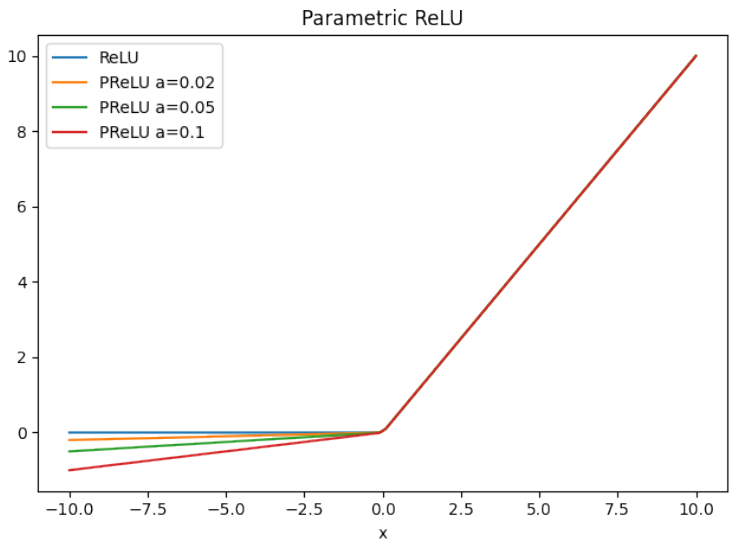
2.2.2.4 ReLU

Η Rectifier Linear Unit (ReLU) ορίζεται μαθηματικά ως $ReLU(x) = max(0, x)$. Είναι μια από τις απλούστερες συναρτήσεις οδηγώντας σε ταχύτερους χρόνους εκπαίδευσης [Kriz12]. Για αυτό το λόγο, σήμερα χρησιμοποιείται συχνά λόγω της χαμηλής πολυπλοκότητας αλλά και της υψηλής αποτελεσματικότητας.



Σχήμα 2.4: Γραφική παράσταση της συνάρτησης ReLU.

Προφανώς η ReLU ανήκει σε μια οικογένεια συναρτήσεων, οι οποίες διαφέρουν κυρίως ως προς την λειτουργία στις αρνητικές τιμές εισόδου. Για παράδειγμα, πολλοί ερευνητές πασχίζουν να αποδεχτούν πως το αρνητικό μέρος των σημάτων, πρακτικά διαγράφεται. Οπότε, προτάθηκαν οι Leaky συναρτήσεις [Zhan18], οι οποίες μειώνουν σημαντικά, αντί να διαγράφουν τελείως την αρνητική συνιστώσα. Μάλιστα, ο βαθμός μείωσης, σε κάποιες περιπτώσεις, μπορεί να θεωρηθεί και παράμετρος του δικτύου, όπως η μεταβλητή α στην Parametric ReLU $PReLU(x) = max(0, x) + min(\alpha \cdot x, 0)$.

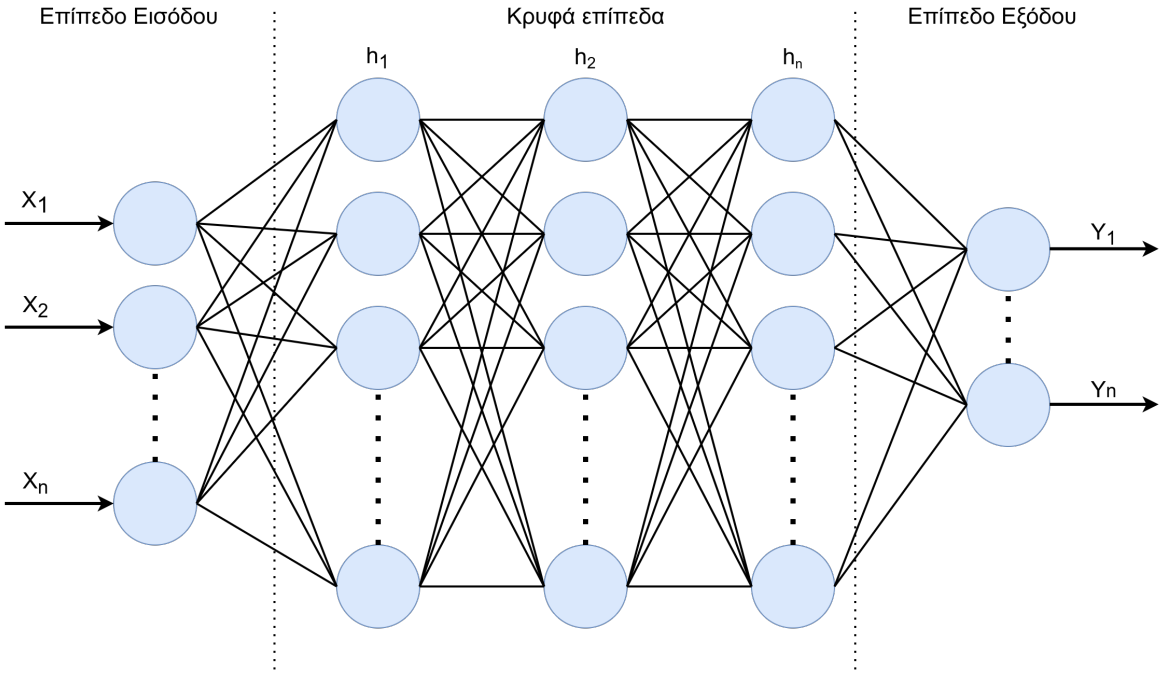


Σχήμα 2.5: Γραφική παράσταση της συνάρτησης ReLU σε σύγκριση με διάφορες περιπτώσεις της παραμετρικής ReLU ανάλογα με την τιμή της παραμέτρου α.

2.2.3 Αρχιτεκτονικές Νευρωνικών Δικτύων

Το νευρωνικό δίκτυο λοιπόν, αποτελείται από επίπεδα διασυνδεδεμένων νευρώνων. Έχει ένα επίπεδο εισόδου, πολλαπλά ενδιάμεσα επίπεδα και ένα επίπεδο εξόδου. Το επίπεδο εισόδου δέχεται τα δεδομένα εκπαίδευσης **X**. Κάθε επιμέρους τιμή του διανύσματος εισόδου **X** συνδέεται με όλους τους κόμβους του πρώτου κρυφού επιπέδου μέσω συνδεσμολογίας νευρώνων. Το ίδιο γίνεται και στα επόμενα κρυφά επίπεδα μέχρι το τελικό επίπεδο εξόδου το οποίο έχει διαστάσεις ίσες με την αναμενόμενη έξοδο y. Στόχος του δικτύου είναι να ελαχιστοποιήσει μία συνάρτηση κόστους C, μέτρο του πόσο καλά αποτελέσματα παράγει. Η αρχιτεκτονική των δικτύων εξαρτάται από το πλήθος των επιπέδων νευρώνων, τον τρόπο που συνδέονται καθώς και από τον αλγόριθμο μάθησης που χρησιμοποιείται. Υπάρχουν δύο βασικές κατηγορίες αρχιτεκτονικών δικτύων:

1. **Δίκτυα πρόσθιας τροφοδότησης (feedforward neural network):**
   Περιέχουν ένα επίπεδο εισόδου με διαστάσεις ίδιες με το σήμα δεδομένων εισόδου. Κάθε κόμβος εισόδου, συνδέεται με όλους τους κόμβους του επόμενου επιπέδου. Αυτό μπορεί να είναι το επίπεδο νευρώνων εξόδου, το οποίο παράγει αμέσως την έξοδο, αν το δίκτυο είναι στην πιο απλή του μορφή. Συνήθως όμως υπάρχουν πολλαπλά ενδιάμεσα επίπεδα νευρώνων τα οποία ονομάζονται "κρυφά". Τότε το νευρωνικό δίκτυο μπορεί να ερμηνευθεί ως ένας κατευθυνόμενος γράφος όπου το σήμα που τροφοδοτείται στην είσοδο, ρέει προς τα μπροστινά επίπεδα μέσω των κρυφών επιπέδων, εξού και το όνομα πρόσθια τροφοδότηση.



Σχήμα 2.6: Η δομή ενός πλήρους συνδεδεμένου νευρωνικού δικτύου.

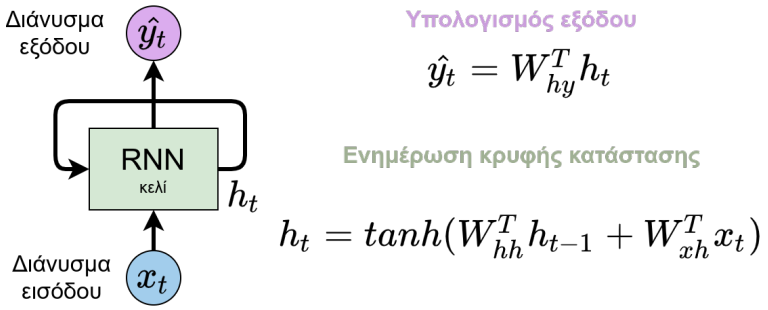
   Ο όρος *"κρυφό"* αναφέρεται στο γεγονός ότι αυτό το μέρος του νευρωνικού δικτύου, δεν είναι ορατό από την είσοδο και την έξοδο του δικτύου. Ο ρόλος τους, είναι να παράγουν χρήσιμα ενδιάμεσα αποτελέσματα κάνοντας υπολογισμούς, βοηθώντας το δίκτυο να εξάγει υψηλότερης τάξης πληροφορία από τα δεδομένα εισόδου. Ένα νευρωνικό δίκτυο λέγεται πλήρως συνδεδεμένο, όταν κάθε κόμβος σε κάθε επίπεδο, συνδέεται με κάθε άλλο κόμβο του επόμενου επιπέδου.

2. **Αναδρομικά δίκτυα (recurrent neural network):**
   Ανάδραση (feedback) θεωρείται ότι υπάρχει σε ένα σύστημα, όποτε η έξοδος ενός κόμβου ή στοιχείου, επηρεάζει την είσοδό του, εμφανίζοντας μια κυκλική διαδρομή στον γράφο αναπαράστασης της ροής του συστήματος. Τα αναδρομικά δίκτυα λοιπόν διαφέρουν σε σχέση με

ένα δίκτυο πρόσθιας τροφοδότησης, στο ότι έχουν τουλάχιστον έναν βρόχο ανάδρασης. Τα αναδρομικά αυτά δικτυα, είναι ο καλύτερος τρόπος για την μοντελοποίηση και την πρόβλεψη χρονοσειρών. Ο βασικός σκοπός των αναδράσεων, είναι πως επιτρέπουν στην πληροφορία να επιμείνει καθώς προχωράει ο χρόνος, πράγμα μη εφικτό με ένα παραδοσιακό δίκτυο πρόσθιας τροφοδότησης. Ο τρόπος με τον οποίο το αναδρομικό δίκτυο, περνάει πληροφορία από το ένα χρονικό βήμα, στο επόμενο, είναι κρατόντας μια εσωτερική, *"κρυφή"* κατάσταση του κάθε αναδρομικού κελιού. Με κάθε νέο δείγμα, αυτή η κατάσταση ενημερώνεται μέσω μιας αναδρομικής σχέσης μεταξύ τωρινής εισόδου, εξόδου, και της προηγούμενης τιμής της εσωτερικής κατάστασης.

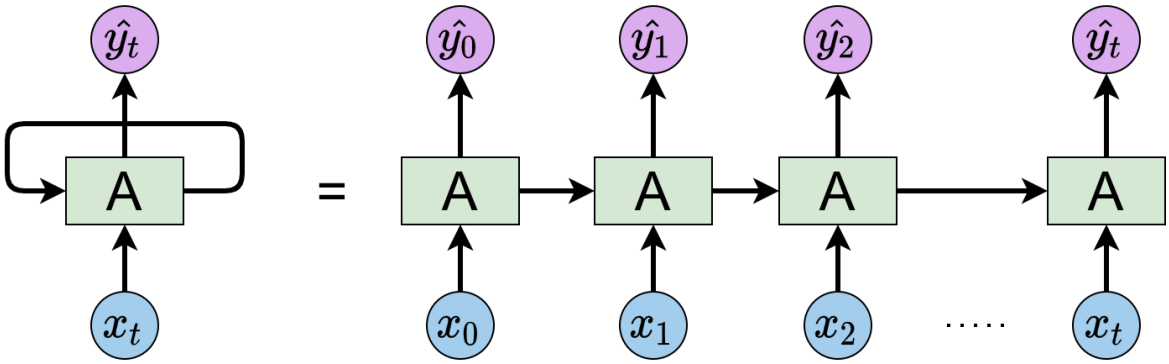
Η ενημέρωση εκφράζεται μαθηματικά από την σχέση $h_t = f_W(h_{t-1}, x_t)$ η οποία αντιστοιχεί στις πράξεις που γίνονται σε έναν κλασικό τεχνικό νευρώνα. Πολλαπλασιάζουμε τις εισόδους με έναν πίνακα βαρών W και περνάμε το αποτέλεσμα από μια μη-γραμμικότητα.

Η βασική διαφορά έγκειται στο ότι εφόσων έχουμε δύο εισόδους, την παλαιά κατάσταση και την τωρινή είσοδο, χρειαζόμαστε έναν πίνακα βαρών για την κάθε μία, όπως παρουσιάζεται και στο Σχ. 2.7. Η τελική έξοδος του κελιού, μπορεί να υπολογιστεί από την ανανεωμένη κατάσταση, πολλαπλασιάζοντάς την με έναν ξεχωριστό πίνακα βαρών.



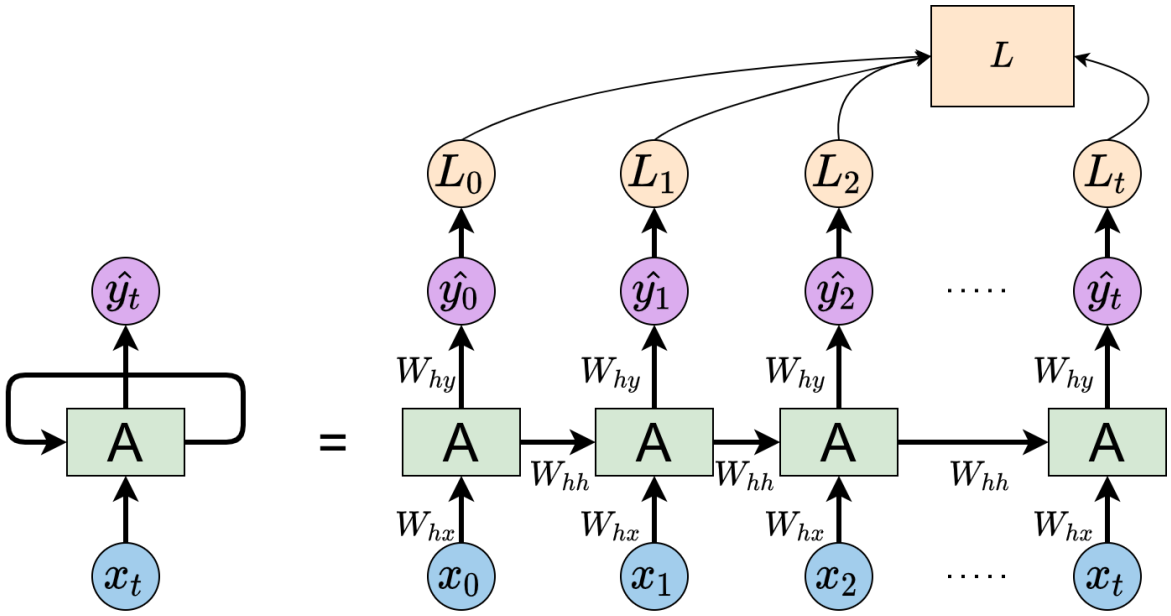
Σχήμα 2.7: Μαθηματικές σχέσεις ενός κελιού RNN.

Το τεράστιο πλεονέκτημα των αναδρομικών δικτύων, όταν έχουμε να κάνουμε με χρονοσειρές διαδοχικών δεδομένων φαίνεται στο παρακάτω σχήμα, όπου έχουμε αναπαραστήσει όλες τις καταστάσεις του δικτύου διαδοχικά.



Σχήμα 2.8: Το αναδρομικό δίκτυο, "ξετυλιγμένο" στο χρόνο.

Κάθε κελί μεταφέρει πληροφορία στο επόμενο (μέσω του υπολογισμού του h_t). Όπως βλέπουμε στο Σχ. 2.9, σε κάθε στιγμή οι πίνακες βαρών W παραμένουν ίδιοι και υπολογίζονται διαφορετικές έξοδοι για κάθε είσοδο. Για κάθε τιμή εξόδου, υπολογίζεται μια τιμή συνάρτησης κόστους και για τον υπολογισμό του συνολικού κόστους, προστίθονται οι επιμέρους τιμές.



Σχήμα 2.9: Γραφική απεικόνιση της εμπρόσθιας τροφοδότησης ενός αναδρομικού δικτύου.

2.2.4 Τεχνικές Εκπαίδευσης Νευρωνικών Δικτύων

Χρησιμοποιώντας την συνάρτηση κόστους αποφασίζεται, αν ένα νευρωνικό δίκτυο, έχει καλή απόδοση ή όχι. Όμως αυτό από μόνο του δεν επαρκεί. Πρέπει με κάποιο τρόπο να κατευθύνουμε το σύστημα προς την βέλτιστη περιοχή ελαχίστου κόστους. Αυτό επιτυγχάνεται μέσω αλγορίθμων που υπολογίζουν την κατάλληλη κατεύθυνση αλλαγής των παραμέτρων του δικτύου σε κάθε επίπεδο.

2.2.4.1 Γραμμική Παλινδρόμηση (Gradient descent)

Από την μαθηματική ανάλυση, γνωρίζουμε πως η κλίση (gradient) μιας συνάρτησης ∇f σε κάποιο συγκεκριμένο σημείο, αντιστοιχεί σε ένα διάνυσμα, κάθετο στην επιφάνεια, προσανατολισμένο προς την κατεύθυνση όπου η συνάρτηση αυξάνεται με τον μεγαλύτερο ρυθμό. Εφόσων ο αλγόριθμος Γραμμικής Παλινδρόμησης στοχεύει στην ελαχιστοποίηση της συνάρτησης κόστους, τα βάρη του δικτύου θα πρέπει να αλλάζουν προς την κατεύθυνση του διανύσματος $-\nabla f$.

Πράγματι για την ενημέρωση των πινάκων βαρών, χρησιμοποιείται η εξίσωση $W_+ = W - \eta \nabla f$. Το $'\eta'$ είναι μια σταθερά που λέγεται ρυθμός μάθησης (learning rate) και ρυθμίζει, τι ποσοστό της παραγώγου θα χρησιμοποιήσουμε, ή σχηματικά αντιστοιχεί στο μέγεθος του βήματος που πρέπει να πάρουμε προς την βέλτιστη κατεύθυνση την εκάστοτε στιγμή.

Μια συνήθης λύση που στοχεύει στην εύρεση του ολικού ελαχίστου είναι η στοχαστική γραμμική παλινδρόμηση (Stochastic Gradient Descent - SGD) [Bott10]. Η βασική διαφορά είναι ότι ο υπολογισμός γίνεται με την χρήση τυχαίων δειγμάτων, και όχι ολόκληρου του συνόλου των δεδομένων εκπαίδευσης. Έτσι μειώνεται η πιθανότητα σύγκλισης του αλγορίθμου σε τοπικό ελάχιστο, καθώς αυξάνεται και η ταχύτητα εκτέλεσης του αλγορίθμου.

Διάφορες τεχνικές βελτιστοποίησης έχουν αναπτυχθεί και αποδειχθεί να συγκλίνουν επιτυχώς. Επί του παρόντος, οι πιο διαδεδομένες είναι αυτές με αυτόματο καθορισμό του ρυθμού μάθησης όπως οι Adagrad [Duch11], Adadelta [Zeil12] and Adam [King14].

2.2.4.2 Οπισθοδιάδοση (Backpropagation)

Η εκπαίδευση ενός δικτύου λοιπόν, απαιτεί αλγόριθμους που να υπολογίζουν αποδοτικά τις παραγώγους των συναρτήσεων κόστους με στόχο την ορθότερη ανανέωση των παραμέτρων. Ένας αποδοτικός τρόπος υπολογισμού της κλίσης (gradient) εισήχθη με τον αλγόριθμο backpropagation [Rume85,

LeCu98]. Ο αλγόριθμος της οπισθοδιάδοσης υπολογίζει μεθοδικά τις παραγώγους μιας σύνθετης μαθηματικής έκφρασης χρησιμοποιώντας τον κανόνα αλυσίδας, ενώ αποθηκεύει προσωρινά τα ενδιάμεσα αποτελέσματα. Οι παράγωγοι αυτοί της συνάρτησης κόστους, είναι αδιαμφισβήτητα το σημαντικότερο στοιχείο που οδηγεί τα νευρωνικά δίκτυα στο να φτάσουν στο ελάχιστο το κόστος, να επιτύχουν δηλαδή στην εκπαίδευσή τους.

Συγκεκριμένα, η ανανέωση των βαρών γίνεται από τον αλγόριθμο Gradient Descent και βρίσκεται από τον τύπο $w_{ij}(l,k+1) - w_{ij}(l,k) = -\eta \frac{\partial J}{\partial w_{ij}(l,k)}$. Όπου $w_{ij}(l,k)$ είναι το συναπτικό βάρος που συνδέει τον νευρώνα j του προηγούμενου επιπέδου $(l-1)$ με τον νευρώνα i του επιπέδου l, η είναι ο ρυθμός εκπαίδευσης και J το κόστος. Οπότε ο στόχος είναι ο υπολογισμός του $\frac{\partial J}{\partial w_{ij}(l,k)}$.

Ο υπολογισμός του γίνεται συναρτήσει της διέγερσης u του νευρώνα, ο οποίος δίνεται από τη σχέση $u_i^{(k)} = \sum_{j=1}^{N} w_{ij}(l,k) y_j^{(k)}(l-1) + w_{i0}(l,k)$ και της εξόδου του y η οποία βρίσκεται από την $y_i^{(k)}(l) = f(u_i^{(k)}(l))$. Από τον κανόνα της αλυσίδας ισχύει ότι $\frac{\partial J}{\partial w_{ij}(l,k)} = \frac{\partial J}{\partial u_i(l)^{(k)}} \frac{\partial u_i(l)^{(k)}}{\partial w_{ij}(l,k)}$, όπου $\frac{\partial u_i(l)^{(k)}}{\partial w_{ij}(l,k)} = y_j^{(k)}(l-1)$.

Στην περίπτωση που χρησιμοποιείται τετραγωνικό σφάλμα ανάμεσα στην αναμενόμενη έξοδο d και την πραγματική τιμή y, το σφάλμα J ισούται με $J = \frac{1}{P}\sum_{p=1}^{P} \|d^{(p)} - y^{(p)}\|^2 = \frac{1}{P}\sum_{p=1}^{P}\sum_{i=1}^{m}[d_i^{(p)} - y_i^{(p)}]^2$. Άρα προκύπτει για το τελευταίο επίπεδο $-\frac{\partial J}{\partial u_i^{(k)}(L)} = -\frac{\partial J}{\partial y_i^{(k)}(L)} \frac{\partial y_i^{(k)}(L)}{\partial u_i^{(k)}(L)}$ όπου $\frac{\partial J}{\partial y_i^{(k)}(L)} = -(d_i^{(k)} - y_i^{(k)})$. Άρα αντικαθιστόντας $-\frac{\partial J}{\partial y_i^{(k)}(L)} = (d_i^{(k)} - y_i^{(k)}) \frac{\partial f(u_i^{(k)}(L)}{\partial u_i^{(k)}(L)} = (d_i^{(k)} - y_i^{(k)}) f'(u_i^{(k)}(L))$.

Οπότε τελικά συμπεραίνουμε πως η παράγωγος που αναζητείται, εξαρτάται από την διαφορά ανάμεσα στην επιθυμητή και την πραγματική έξοδο, αλλά και απο την παράγωγο της συνάρτησης ενεργοποίησης. Ανάλογα με τον τύπο της συνάρτησης ενεργοποίησης, όπως έχουν αναλυθεί στο κεφάλαιο 2.2.2, η παράγωγος μπορεί να υπολογιστεί ως εξής:

- Σιγμοειδής: $f(u) = \frac{1}{1+e^{-u}}$, η παράγωγος $f'(u) = f(u)(1 - f(u)$
- Υπερβολική: $f(u) = \frac{e^u - e^{-u}}{e^u + e^{-u}}$, η παράγωγος $f'(u) = (1 + f(u))(1 - f(u))$
- Γραμμική: $f(u) = u$, η παράγωγος $f'(u) = 1$

Για τα υπόλοιπα επίπεδα, ισχύει ότι $-\frac{\partial J}{\partial u_i^{(k)}(l)} = -\sum_{\mu=1}^{N(l+1)} \frac{\partial J}{\partial u_\mu^{(k)}(l+1)} \frac{\partial u_\mu^{(k)}(l+1)}{\partial y_i^{(k)}(l)} \frac{\partial y_i^{(k)}(l)}{\partial u_i^{(k)}(l)}$, το οποίο σημαίνει πως οι παράγωγοι κάθε επιπέδου εξαρτώνται από την παράγωγο του νευρώνα μ που βρίσκεται στο επόμενο επίπεδο $(l+1)$. Οπότε κάθε φορά που ολοκληρώνεται ένας υπολογισμός, ξεκινώντας από το τελευταίο επίπεδο, η πληροφορία προωθείται στο προηγούμενό του. Εξού και το όνομα του αλγορίθμου "οπισθοδιάδοσης" (back propagation).

2.2.4.3 Οπισθοδιάδοση στο χρόνο (BPTT)

Για την εκπαίδευση και την αλλαγή των βαρών αναδρομικών νευρωνικών δικτύων, χρησιμοποιείται ο αλγόριθμος Backpropagation Through Time (BPTT). Η λογική είναι παρόμοια με την απλή οπισθοδιάδοση, επιτυγχάνεται ανανέωση των βαρών υπολογίζοντας την παράγωγο του κόστους ως προς τα βάρη. Αρχικά δίνεται μια ακολουθία στο δίκτυο και παράγονται οι προβλέψεις. Έπειτα "ξεδιπλώνεται" το δίκτυο και υπολογίζονται τα κόστη για κάθε χρονοστιγμή. Τέλος επανασυναρμολογείται το δίκτυο και ανανεώνονται τα βάρη κατάλληλα.

2.3 Μοντέλα Βαθιάς Μάθησης (Deep Learning)

Η βαθιά μηχανική μάθηση (Deep Learning) ανήκει στο ευρύτερο πεδίο της μηχανικής μάθησης. Μέθοδοι όπως βαθιά νευρωνικά δίκτυα, αναδρομικά δίκτυα και συνελικτικά νευρωνικά δίκτυα, χρησιμοποιούνται επιτυχώς τα τελευταία χρόνια για την επίλυση προβλημάτων όρασης υπολογιστών, αναγνώρισης φωνής, επεξεργασίας φυσικής γλώσσας, βιοεπιστήμης και ανάλυσης-πρόβλεψης χρονοσειρών. Ο όρος "βαθύ" προκύπτει από την ύπαρξη πολλαπλών επιπέδων στα δίκτυα αυτά. Η βαθιά μάθηση, είναι η μοντέρνα εκδοχή της μηχανικής μάθησης. Η αποτελεσματική εκπαίδευση των δικτύων απαιτεί πληθώρα δεδομένων, συχνά στην τάξη των πολλών χιλιάδων δειγμάτων. Επίσης, αν και δεν είναι απαραίτητη, η παράλληλη επεξεργασία των δεδομένων σε κάρτα Γραφικών (GPU) με κατάλληλες βιβλιοθήκες έχει επιταχύνει ιδιαίτερα την διάρκεια εκπαίδευσης, σε σύγκριση με την εκτέλεση στην κεντρική μονάδα επεξεργασίας (CPU). Η εκθετική αύξηση των δεδομένων λόγω του διαδικτύου, καθώς και η ραγδαία, ωθούμενη από την βιομηχανία βιντεοπαιχνιδιών, ανάπτυξη στις αποδόσεις των καρτών γραφικών, είναι οι λόγοι για τους οποίους, η βαθιά μάθηση αποτελεί την τελευταία λέξη της τεχνολογίας (state of the art) σε ότι αφορά την ανάπτυξη μοντέλων τεχνητής νοημοσύνης.

2.3.1 Συνελικτικά δίκτυα

Η ανάλυση εικόνων με στόχο την ταξινόμηση, αναγνώριση προτύπων, κατάτμηση ή την ανάπτυξη συστημάτων σύστασης, αποτελεί ένα σημαντικό ποσοστό των εφαρμογών για τις οποίες προτείνουν λύσεις τα δίκτυα βαθιάς μάθησης.

Συγκεκριμένα, για προβλήματα που περιέχουν δεδομένα εικόνας, χρησιμοποιούνται δίκτυα με επίπεδα συνέλιξης. Για τον υπολογισμό της πράξης της συνέλιξης μεταξύ εικόνων, χρησιμοποιείται ο τύπος της δισδιάστατης συνέλιξης $C(i,j) = \sum_{m=0}^{(M_a-1)} \sum_{n=0}^{(N_a-1)} A(m,n) * B(i-m, j-n)$ ο οποίος εφαρμόζεται μεταξύ δύο πινάκων από pixels (εικόνων) A και B και παράγει έναν νέο πίνακα C. Ο πρώτος πίνακας αντιστοιχεί στην εικόνα εισόδου, ο δεύτερος, είναι ένα φίλτρο (kernel) το οποίο "περνάει" πάνω από την εικόνα, φιλτράροντάς την και παράγοντας τον πίνακα εξόδου C.

Η αρχιτεκτονική των συνελικτικών νευρωνικών δικτύων, αποτελείται (όπως και στα νευρωνικά δίκτυα), από ένα επίπεδο εισόδου και εξόδου, καθώς και από πολλαπλά κρυφά επίπεδα. Κάθε κρυφό επίπεδο, αποτελείται από διαδοχικά επίπεδα συνέλιξης, συνάρτησης ενεργοποίησης, υποδειγματοληψίας της εικόνας καθώς και από ένα ή περισσότερα πλήρως συνδεδεμένα επίπεδα (fully connected). Τα διαδοχικά αυτά επίπεδα έχουν τις εξής λειτουργίες:

- **Συνελικτικά (Convolutional):** Στοχεύουν στον υπολογισμό της εικόνας εξόδου ή χάρτη χαρακτηριστικών (feature map), μετά την εφαρμογή του φίλτρου (kernel) του οποίου οι τιμές είναι εκπαιδεύσιμες (trainable) από το δίκτυο και το πέρασμα του αποτελέσματος από μια συνάρτηση ενεργοποίησης. Πρακτικά, η ορθή προσαρμογή αυτών των παραμέτρων είναι αυτό που ωθεί το δίκτυο στην σωστή ανάλυση της εικόνας, πετυχαίνοντας το ζητούμενο αποτέλεσμα.

- **Υποδειγματοληψία (Pooling):** Μειώνει τις διαστάσεις του πίνακα εισόδου συνδυάζοντας τις εξόδους του προηγούμενου επιπέδου. Συνδυάζει pixels περιοχών της εικόνας εισόδου, τυπικά 2×2, υπολογίζοντας το μέγιστο (max pooling) ή το μέσο (average pooling) των τιμών.

- **Πλήρως συνδεδεμένο επίπεδο (fully connected layer):** Οι τελικοί χάρτες χαρακτηριστικών που προκύπτουν ύστερα από τις πολλές εναλλαγές συνελικτικών και pooling επιπέδων, περνούν από ένα πλήρως συνδεδεμένο νευρωνικό δίκτυο, όπως αυτό που περιγράφεται στην ενότητα 2.2.3, για την εξαγωγή του τελικού αποτελέσματος.

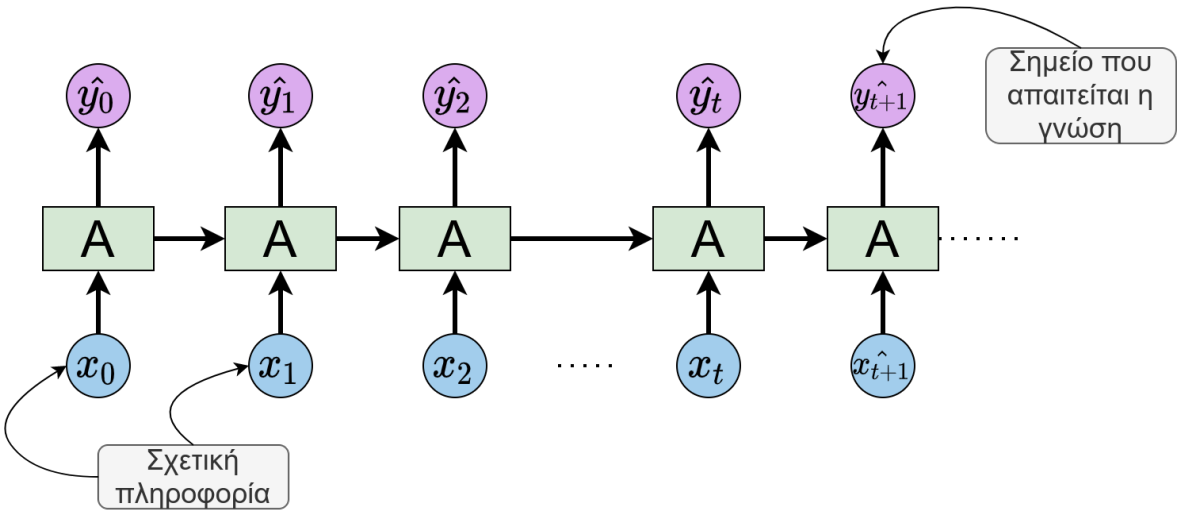
Τα συνελικτικά νευρωνικά δίκτυα, έχουν χρησιμοποιηθεί επιτυχώς και για προβλήματα ανάλυσης χρονοσειρών [Fawa19], αφού τα δεδομένα αυτά , μπορούν να θεωρηθούν ως πίνακες εισόδου και άρα το συνελικτικό δίκτυο θα μπορέσει να ανιχνεύσει πρότυπα αυτών.

2.3.2 Long Short Term Memory (LSTM) Neural Networks

Μια ειδική κατηγορία αναδρομικών δικτύων (RNN) που περιγράφονται στην ενότητα 2.2.3, είναι τα LSTM. Προτάθηκαν αρχικά από τους Hochreiter και Schmidhuber το 1997 [Hoch97] και έχουν μελετηθεί και αναπτυχθεί από ερευνητές έκτοτε. Τα αποτελέσματα χρήσης των δικτύων σε δεδομένα χρονοσειρών είναι πολλά υποσχόμενα, προσφέροντας λύσεις σε πληθώρα σύγχρονων προβλημάτων.

Τα LSTM είναι σχεδιασμένα έτσι ώστε να λύσουν ένα από τα βασικά θέματα που έχουν τα RNN, την εκμάθηση μακροσκελών εξαρτήσεων. Για παράδειγμα, ένα RNN σχεδιασμένο να δέχεται 3 λέξεις μιας πρότασης, για να προβλέπει την επόμενη, δεν έχει την δυνατότητα να "θυμάται" σημαντική προηγούμενη πληροφορία η οποία μπορεί να βρισκόταν στην προηγούμενη πρόταση. Το πρόβλημα αυτό, είχε εντοπιστεί από τον Hochreiter [Hoch91] και Bengio [Beng94].

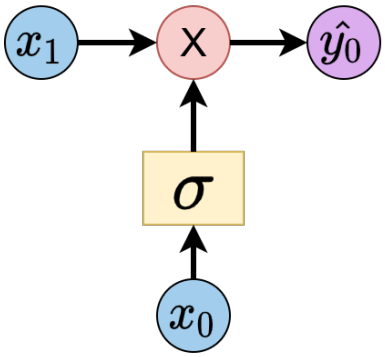
Όσο το χάσμα μεταξύ της σχετικής πληροφορίας και το σημείο στο οποίο είναι απαραίτητη μεγαλώνει, τόσο δυσκολότερα μπορεί να "μάθει" ένα RNN, όπως φαίνεται και στο Σχ. 2.10 .



Σχήμα 2.10: Απεικόνιση του περιθωρίου μεταξύ της χρήσιμης πληροφορίας και του σημείου αναγκαιότητάς της σε σειριακά δεδομένα.

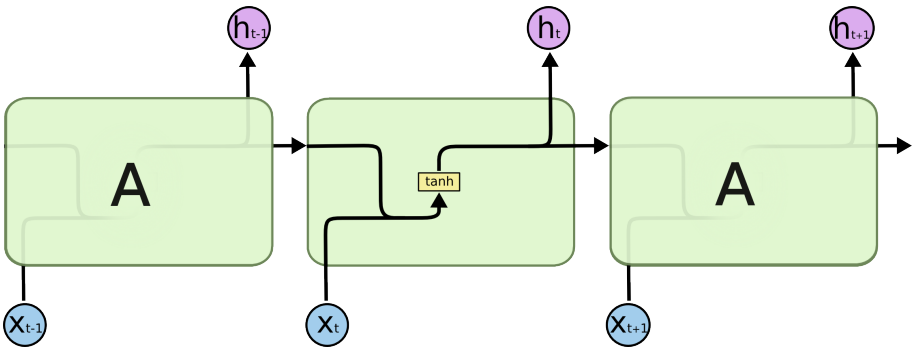
Όπως και για τα RNN στην ενότητα 2.2.3 η περιγραφή του LSTM θα γίνει με την ανάλυση του βασικού κελιού ενός δικτύου LSTM [cola]. Αρχικά στο Σχ. 2.12 παρουσιάζεται η δομή μιας μονάδας LSTM σε σύγκριση με αυτή ενός απλού αναδρομικού δικτύου.

Η βασική ιδέα στηρίζεται στην ύπαρξη διασυνδεδεμένων *πύλων* (gates) οι οποίες ελέγχουν (ενθαρρύνουν ή όχι) την ροή της πληροφορίας από ένα σημείο στο επόμενο. Η δομή μιας τέτοιας πύλης (Σχ. 2.11) εμπεριέχει μια σιγμοειδή συνάρτηση από την οποία περνάει ένα διάνυσμα εισόδου, και πολλαπλασιάζεται έπειτα με ένα δεύτερο διάνυσμα για να παραχθεί η τελική έξοδος.

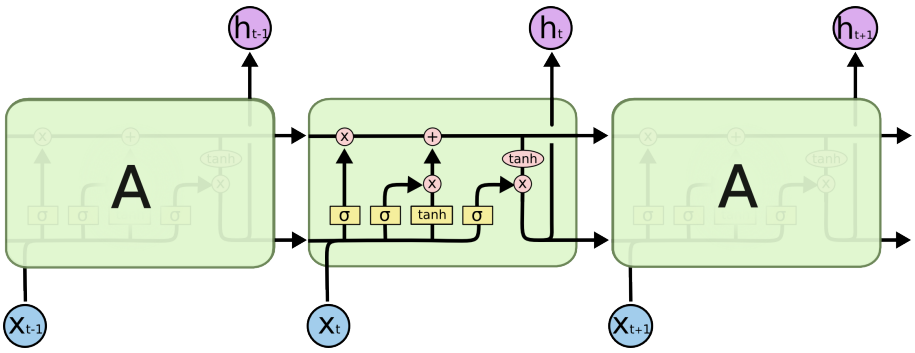


Σχήμα 2.11: Η δομή μιας πύλης αποτελούμενη από μια σιγμοειδή συνάρτηση σ που ελέγχει την ροή της πληροφορίας.

Ονομάζονται πύλες, διότι η σιγμοειδής συνάρτηση συμπιέζει τις τιμές των διανυσμάτων στο [0,1] και πολλαπλασιάζοντας με κάποιο άλλο διάνυσμα, ελέγχουμε πόσο αυτό, θα επηρεάσει το αποτέλεσμα.



(a)

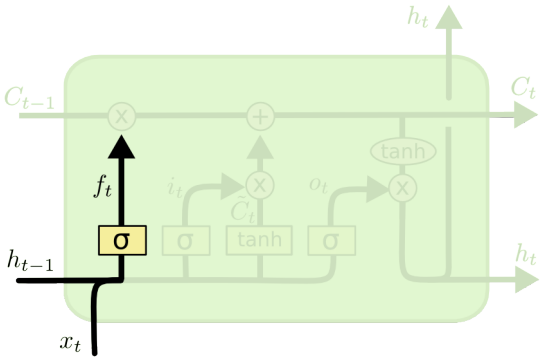


(b)

Σχήμα 2.12: Η δομή κελιού ενός απλού αναδρομικού δικτύου (**a**) σε σύγκριση με την δομή του LSTM (**b**)

Η ανάλυση του *"κελιού"* ενός LSTM μπορεί να γίνει με τα παρακάτω βήματα:

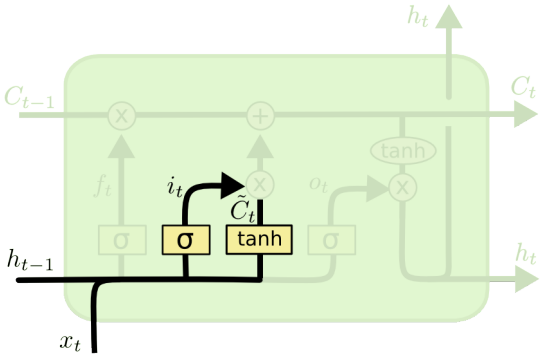
- **Forget gate:** Σε αυτό το σημείο, διαλέγουμε την άχρηστη πληροφορία, που περιέχεται στην προηγούμενη κρυφή κατάσταση h_{t-1} και στην νέα είσοδο x_t, και την "ξεχνάμε" μέσω της πύλης. Παράγεται ένας αριθμός στο [0,1] ο οποίος πολλαπλασιάζεται με τις τιμές της εσωτερικής κατάστασης C_{t-1}, διαλέγοντας τι πληροφορία θα παραμείνει.



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \ + \ b_f\right)$$

Σχήμα 2.13: Forget πύλη του LSTM.

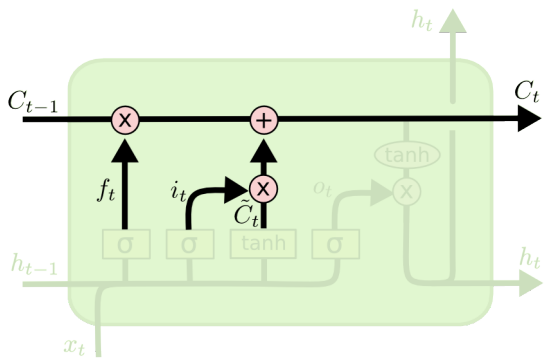
- **Store & Update gate:** Στο επόμενο βήμα, αποφασίζεται, τι νέα πληροφορία θα αποθηκεύσουμε στην εσωτερική κατάσταση C. Αυτό γίνεται, αρχικά με την επιλογή των τιμών που θα ενημερωθούν (σιγμοειδής) καθώς και μέσω της πύλης tanh η οποία παράγει ένα υποψήφιο διάνυσμα τιμών $\tilde{C}_t$ για αυτές.



$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \ + \ b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \ + \ b_C)$$

Σχήμα 2.14: Store πύλη του LSTM.

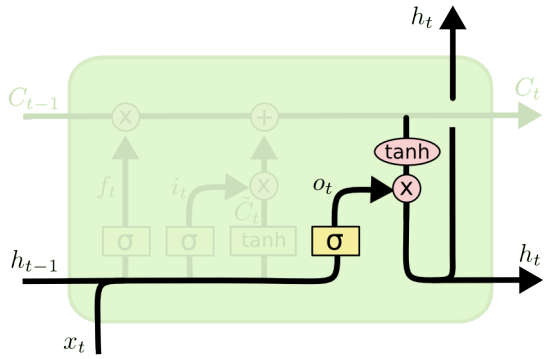
Έπειτα πραγματοποιούμε τις αλλαγές του πρώτου βήματος, ξεχνώντας τις άχρηστες πληροφορίες, και προσθέτουμε τις νέες πληροφορίες $\tilde{C}_t$ πολλαπλασιαζόμενες με το ποσοστό i_t της αλλαγής αυτής.



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Σχήμα 2.15: Update πύλη του LSTM.

- **Output gate:** Τέλος αποφασίζεται, τι αποτέλεσμα θα παραχθεί στην έξοδο. Για να επιτευχθεί αυτό, η κατάσταση που υπολογίστηκε από τα προηγούμενα βήματα C_t φιλτράρεται, περνώντας την από μια συνάρτηση $tanh$ και πολλαπλασιάζεται με την έξοδο μιας σιγμοειδής πύλης, όπου τελικά αποφασίζεται τι πληροφορία θα περάσει στο επόμενο στάδιο.



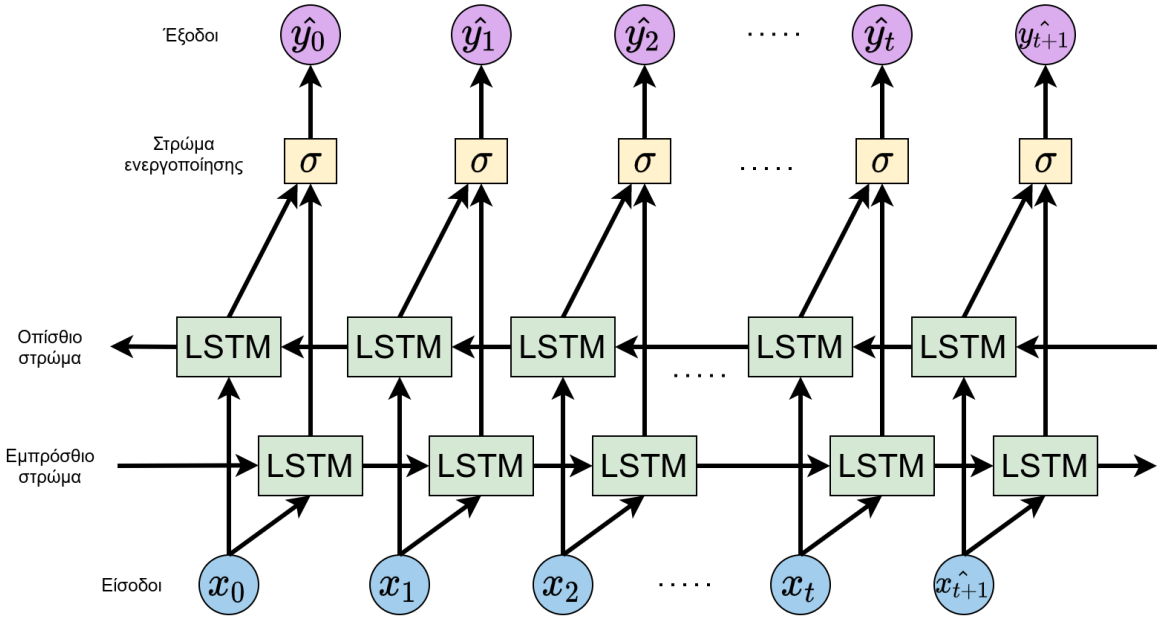
$$o_t = \sigma\left(W_o\ [h_{t-1}, x_t]\ +\ b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

Σχήμα 2.16: Output πύλη του LSTM.

2.3.3 Bidirectional LSTM

Μια παραλλαγή του LSTM που χρησιμοποιείται με επιτυχία είναι το ”αμφίδρομο” (bidirectional) LSTM [Huan15]. Όπως περιγράφεται στην προηγούμενη ενότητα, τα LSTM χρησιμοποιούν την πληροφορία μιας χρονοσειράς, όπου κάθε χρονική στιγμή, επηρεάζει την συμπεριφορά του επόμενου κελιού, οπότε η χρονοσειρά ”διαβάζεται” από την αρχή μέχρι το τέλος. Όμως, αυτή η σειρά μπορεί να αναστραφεί και να παρέχει χρήσιμη πληροφορία.

Αρχικά η είσοδος, προσπελάσεται από την αρχή μέχρι το τέλος, και ύστερα ανάστροφα. Έπειτα, η πληροφορία των κρυφών καταστάσεων συνδυάζεται για τον υπολογισμό της κρυφής κατάστασης του εκάστοτε κελιού. Πρακτικά υπολογίζεται η εμπρόσθια κρυφή κατάσταση $\rightarrow h_t$ την χρονική στιγμή t, έπειτα η ανάδρομη $\leftarrow h_t$ και τότε η κρυφή κατάσταση του κελιού τη στιγμή t μπορεί να υπολογιστεί από την τομή τους $h_t = \rightarrow h_t || \leftarrow h_{-t}$. Η παραπάνω διαδικασία περιγράφεται σχηματικά (Σχ. 2.17) παρακάτω.



Σχήμα 2.17: Η δομή ενός αμφίδρομου LSTM [Yild18].

Κεφάλαιο 3

Απόκτηση, Επεξεργασία και Ανάλυση Δεδομένων χρονοσειράς

3.1 Εισαγωγή

Συχνά, ο καλύτερος τρόπος πρόβλεψης της συμπεριφοράς κάποιου γεγονότος στο μέλλον είναι η παρατήρηση και ανάλυση παρελθόντων γεγονότων.

Αυτό είναι ένα είδος προχωρημένης ανάλυσης που χρησιμοποιείται για την πρόβλεψη μελλοντικών γεγονότων χρησιμοποιώντας δεδομένα, στατιστικούς αλγόριθμους, αναλυτικά ερωτήματα και τεχνικές μηχανικής μάθησης για να βρεθεί η πιθανότητα μελλοντικών αποτελεσμάτων βάσει ιστορικών δεδομένων. Η τεχνική χρησιμοποιείται καθημερινά από οργανισμούς για τον εντοπισμό πιθανών κινδύνων και ευκαιριών.

Το Predictive analytics αναλύει πληροφορίες από υπάρχοντα σύνολα δεδομένων με στόχο την εύρεση προτύπων και τάσεων. Σε αντίθεση με την παραδοσιακή επιχειρησιακή δραστηριότητα σε αυτόν τον τομέα, η οποία σε καμία περίπτωση δε συμβαδίζει με τα σημεία των καιρών, η προγνωστική ανάλυση αναλύει πρότυπα και τις τάσεις και στη συνέχεια τα χρησιμοποιεί για να προβλέψει πιθανά αποτελέσματα.

Η σημασία της ποιότητας, του μεγέθους και της προεπεξεργασίας του συνόλου των δεδομένων είναι καθοριστική, και πολλές φορές μεγαλύτερη από τον αλγόριθμο ή το στατιστικό μοντέλο που εφαρμόζεται. Σε αυτό το κεφάλαιο, περιγράφεται η διαδικασία απόκτησης δεδομένων με αυτόματο τρόπο, η αποθήκευσή τους καθώς και η προεπεξεργασία τους.

3.2 Internet of Things (IoT)

Το Διαδίκτυο των πραγμάτων (Internet of Things) αναφέρεται στην διασύνδεση, μέσω του διαδικτύου, υπολογιστικών συσκευών. Η διασύνδεση αυτή είναι ενσωματωμένη σε καθημερινά αντικείμενα, παρέχοντάς τους τη δυνατότητα να στείλουν και να λάβουν δεδομένα.

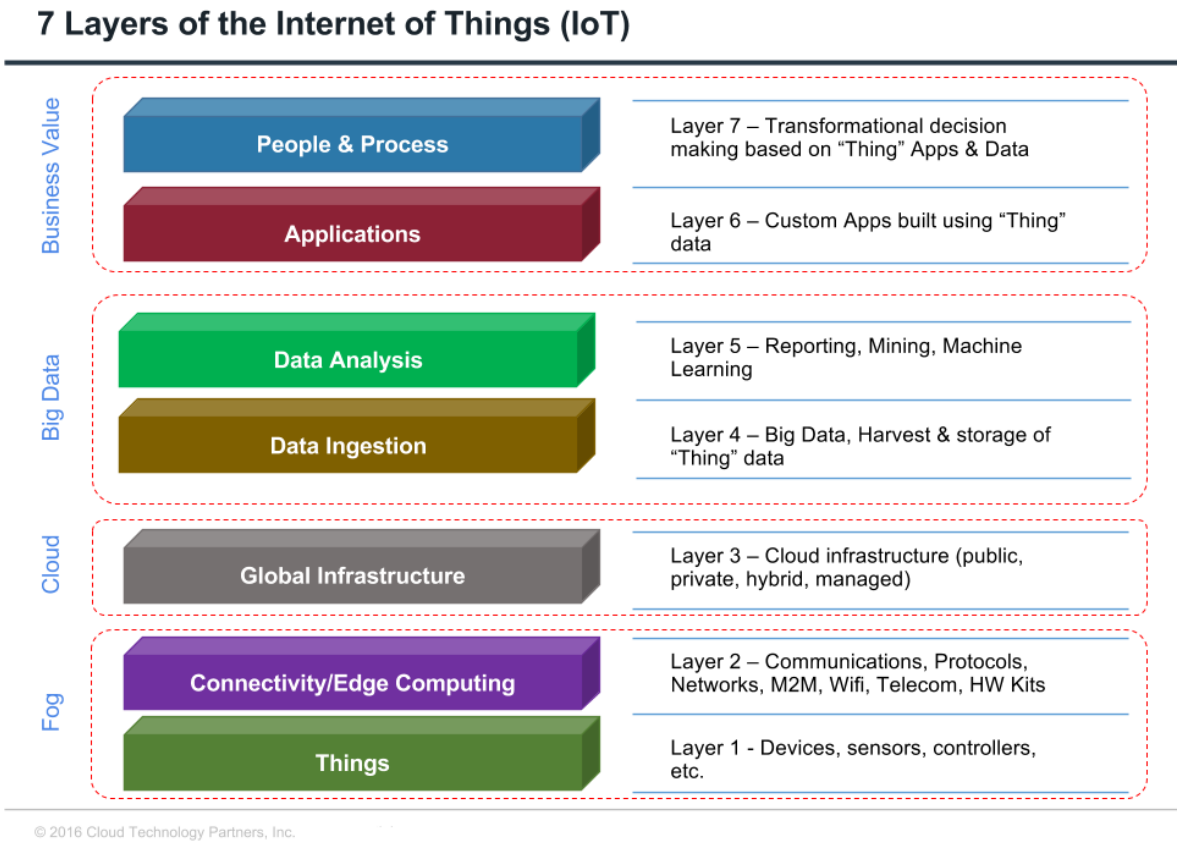
Μέσω αυτού, δημιουργείται μια παγκόσμια υποδομή για την κοινωνία των πληροφοριών, παρέχοντας προηγμένες υπηρεσίες μέσω διασυνδεδεμένων (φυσικών και εικονικών) συσκευών (”πράγματα”) που βασίζονται στις υφιστάμενες και εξελισσόμενες διαλειτουργικές τεχνολογίες πληροφόρησης και επικοινωνίας.

Το IoT χρησιμοποιείται σε μια πληθώρα επιστημονικών πεδίων καθώς και καθημερινών εφαρμογών όπως, αυτόματη διαχείριση (ενεργειακή) κτιρίων, παρακολούθηση περιβάλλοντος, ιατροφαρμακευτική περίθαλψη, έξυπνες πόλεις (smart city), βιομηχανία (smart industrie) και γεωργία (smart agriculture). Επίσης εφαρμόζεται επιτυχώς στην ενεργειακή διαχείριση οργανισμών, εταιρειών και ολόκληρων πόλεων [Berg16], εξοικονομώντας χρήματα και βοηθώντας στη προστασία και διατήρηση του περιβάλλοντος.

Η μετάβαση στην ψηφιακή εποχή είναι γεγονός. Η χρησιμότητα του IoT είναι μεγάλη. Η επεξεργασία και ανάλυση δεδομένων για την εξαγωγή συμπερασμάτων, λήψη αποφάσεων και κατασκευή προβλέψεων για το μέλλον καθιστά την ζήτηση από τους υποψήφιους αγοραστές ακόμα μεγαλύτερη.

Το IoT μπορεί να αναλυθεί ως μια πολυεπίπεδη διαδικασία όπως φαίνεται στο παρακάτω σχήμα (3.1).

- Το πρώτο επίπεδο (πράγματα) αποτελείται από τις συσκευές, τους αισθητήρες και ελεγκτές που παράγουν τα ακατέργαστα δεδομένα.
- Έπειτα περνάμε στο στάδιο συνδεσιμότητας. Η πληροφορία διαδίδεται μέσω πρωτοκόλλων του διαδικτύου.
- Τα δεδομένα δημοσιεύονται σε κάποιο δημόσιο ή ιδιωτικό cloud infrastructure για την έυκολη και γρήγορη διαχείριση της πληροφορίας για κάθε εμπλεκόμενο.
- Στο στάδιο της αποθήκευσης και ανάλυσης των δεδομένων πραγματοποιείται το ξεκαθάρισμα και η προεπεξεργασία των δεδομένων, ώστε να διατηρηθεί η χρήσιμη πληροφορία, επίσης αναπτύσσονται αλγόριθμοι αναφοράς και πρόβλεψης. Ισχυρή είναι η παρουσία της μηχανικής μάθησης σε αυτό το επίπεδο τα τελευταία χρόνια.
- Τέλος, επιστρέφεται η φιλτραρισμένη τελική πληροφορία, απόφαση ή πρόβλεψη. Αυτή είναι και η επιχειρηματική αξία που αποφέρει η επένδυση σε ένα τέτοιο σύστημα.



Σχήμα 3.1: Τα επίπεδα ενός IoT συστήματος.

3.2.1 Industrial Internet of Things (IIoT)

Το Βιομηχανικό Ίντερνετ των πραγμάτων (Industrial Internet of Things) αναφέρεται σε διασυνδεδεμένους αισθητήρες, όργανα και άλλες συσκευές που είναι δικτυωμένες μαζί με βιομηχανικές εφαρμογές υπολογιστών, συμπεριλαμβανομένης της παραγωγής και της διαχείρισης ενέργειας.

Αυτή η συνδεσιμότητα επιτρέπει τη συλλογή δεδομένων, την ανταλλαγή και την ανάλυση, διευκολύνοντας ενδεχομένως βελτιώσεις στην παραγωγικότητα και την αποδοτικότητα, καθώς και άλλα οικονομικά οφέλη.

3.3 Δεδομένα καιρού & Συστήματα IoT

Μολονότι τα δεδομένα καιρού συχνά ενσωματώνονται σε καταναλωτικά προϊόντα IoT όπως εφαρμογές εβδομαδιαίας πρόβλεψης καιρού, δεν υπήρξε μεγάλη ώθηση για την ενσωμάτωση αυτών των πληροφοριών σε βιομηχανικά συστήματα.

Ωστόσο, σύμφωνα με μια αναφορά [Dutt02], ανωμαλότητες στον καιρό, διαταράσσουν τη λειτουργική και οικονομική απόδοση του 70% των επιχειρήσεων παγκοσμίως. Επίσης, σε παρόμοια έρευνα, αναφέρεται πως πάνω από τις μισές επιχειρήσεις, δεν μπορούν να αποφύγουν οικονομικές απώλειες που σχετίζονται με ανωμαλίες καιρού. Τέλος, Το 95% των ερωτηθέντων παραδέχθηκαν απώλειες 10% στα κέρδη λόγω δυσμενών καιρικών συνθηκών.

Με την κλιματική αλλαγή λοιπόν να αυξάνεται, αναμένεται ανάλογη δραστική μεταβολή στα ακραία καιρικά φαινόμενα καθώς και στους αριθμούς που αναφέρονται στις οικονομικές απώλειες λόγω αυτών. Επομένως, είναι φυσική η στροφή προς το Διαδίκτυο των πραγμάτων και τις τεχνικές μηχανικής μάθησης για την εύρεση λύσεων.

3.4 Αυτόματη ανάκτηση δεδομένων καιρού

Το αρχικό πλάνο διεξαγωγής των πειραματισμών, προϋπέθετε την απόκτηση δεδομένων καιρού για συγκεκριμένες γεωγραφικές τοποθεσίες, οι οποίες αντιστοιχούσαν σε περιοχές αιολικών ή και ηλιακών πάρκων στην Κρήτη. Βασική συνθήκη είναι η απόκτηση ζεύγων παρατηρήσεων αλλά και προβλέψεων για τα δεδομένα καιρού για κάθε χρονική στιγμή και τοποθεσία με χρονικό διάστημα μιας ώρας. Για αυτό το λόγο αναπτύχθηκε πρόγραμμα σε Python, το οποίο δεδομένων των συντεταγμένων των σημείων ενδιαφέροντος θα κατεβάζει τα απαιτούμενα δεδομένα χρονοσειράς από το API του Dark Sky [dark].

Τα περισσότερα API καιρού, όπως και το Dark Sky που χρησιμοποιήθηκε, δεν προσφέρουν παρελθοντικές προβλέψεις τους. Συνεπώς κρίθηκε αναγκαία η ανάπτυξη προγράμματος, το οποίο θα εκτελείται την ίδια ώρα και κάθε ημέρα της εβδομάδας, ώστε να αποθηκεύει τις παρατηρήσεις τιμών της ίδιας ημέρας, αλλά και τις προβλέψεις της επόμενης.

Η διαδικασία αυτή, προφανώς απαιτεί την ύπαρξη ενός σταθερού υπολογιστή/εξυπηρετητή (server) ο οποίος θα είναι πάντοτε ενεργός, και θα εκτελεί την εκάστοτε διεργασία κάθε φορά που χρειάζεται.

Η αυτοματοποίηση σε τακτά χρονικά διαστήματα επιτυγχάνεται σε Unix περιβάλλον με την χρήση της λειτουργίας cron ή cron job [mcro].

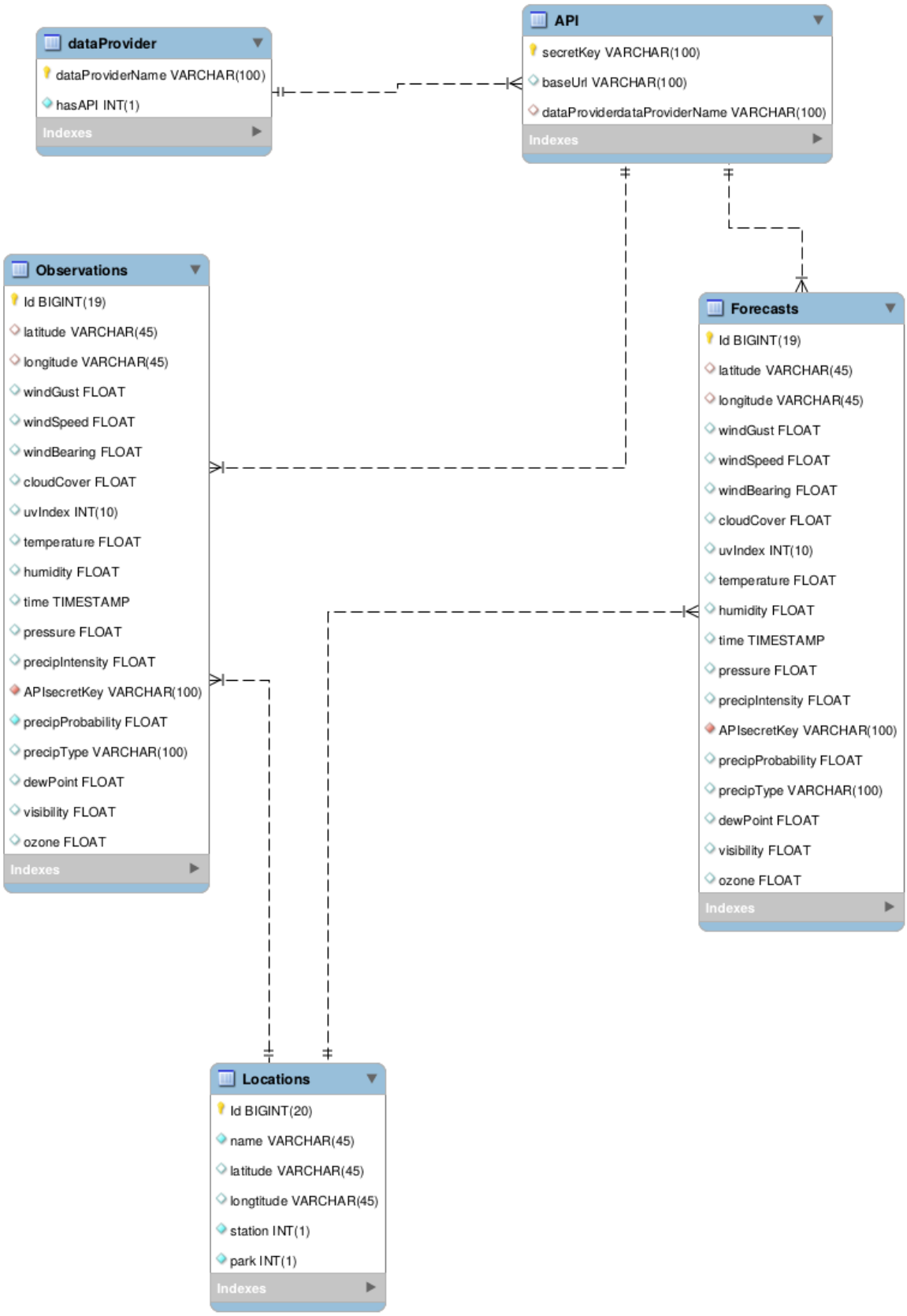
Πρόκειται για έναν προγραμματιστή εργασιών βασισμένο στον χρόνο. Η σύνταξη για την δρομολόγηση εργασιών, ακολουθεί το εξής συντακτικό minute hour day_of_month month day_of_week command_to_run περιέχοντας τις προφανείς επιτρεπόμενες τιμές για κάθε πεδίο, εκτός του αστερίσκου (*) ο οποίος υποδεικνύει ότι η εργασία πρέπει να εκτελεστεί για κάθε τιμή του αντίστοιχου πεδίου.

Συγκεκριμένα, σε υπολογιστικό μηχάνημα, που είχαμε στη διάθεσή μας, ορίστηκε μια cron εργασία η οποία μια φορά κάθε ημέρα, εκτελεί το script daily.sh το οποίο τρέχει, το ένα μετά το άλλο, δύο προγράμματα python.

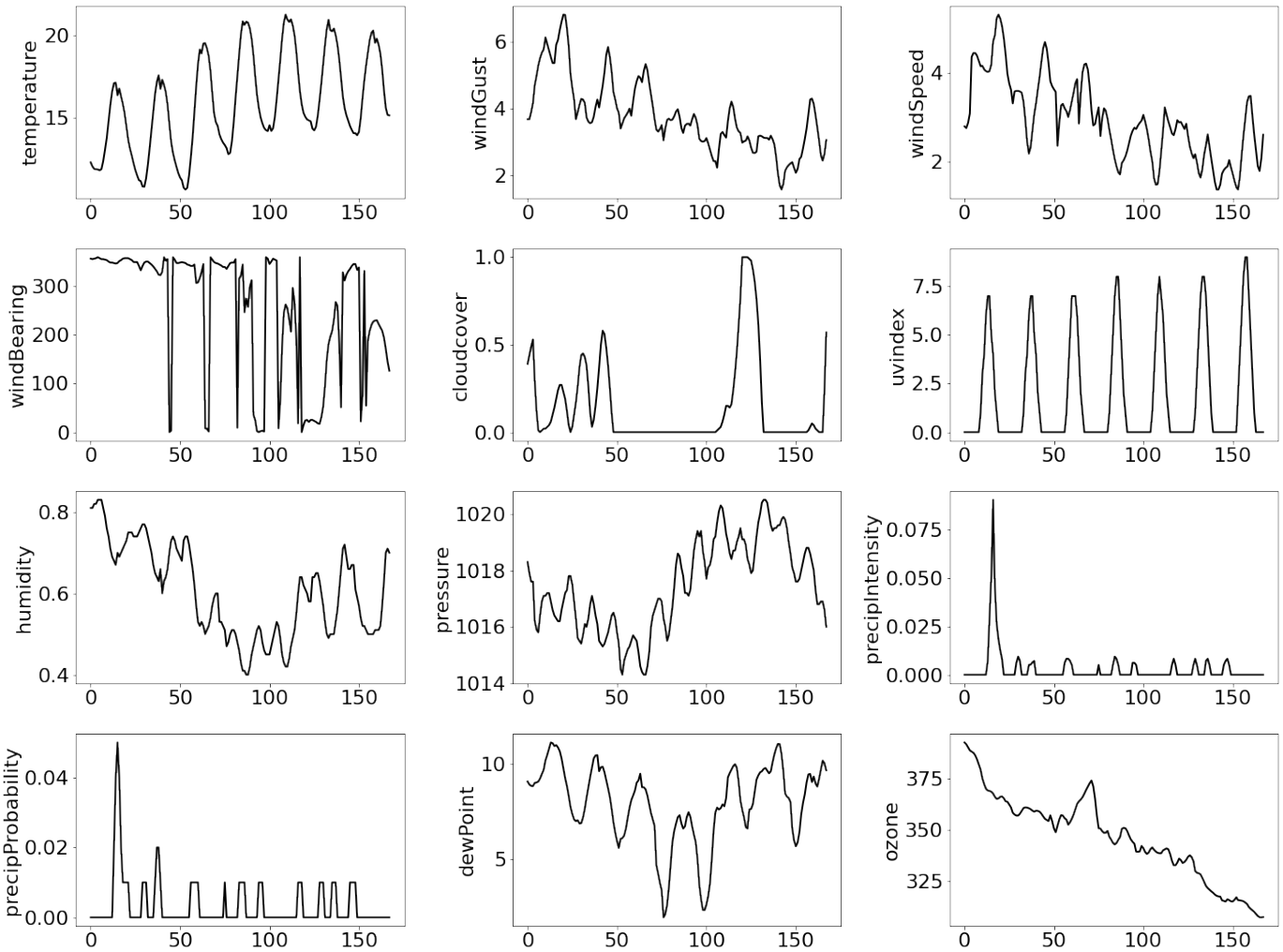
```
1 python3 ~/Thesis/Scripts/fillDBobservations.py
2 python3 ~/Thesis/Scripts/fillDBpredictions.py
```

Σχήμα 3.2: Το script daily.sh

Αυτά ”χτυπούν” τα αντίστοιχα endpoints του API και κατεβάζουν έτσι τις ημερήσιες παρατηρήσεις και προβλέψεις αντίστοιχα, για την εκάστοτε μέρα. Έπειτα τα αποτελέσματα αποθηκεύονται σε μια βάση δεδομένων MySQL αλλά και σε μορφή .json αρχείων. Η δομή της βάσης φαίνεται στο παρακάτω ER διάγραμμα Σχ.3.3.



Σχήμα 3.3: ER διάγραμμα της βάσης δεδομένων καιρού.



Σχήμα 3.4: Γραφική παράσταση δεδομένων που ανακτήθηκαν.

3.5 Προεπεξεργασία δεδομένων

Πριν την εφαρμογή αλγορίθμων για την εξόρυξη γνώσης, το ερευνώμενο σύνολο δεδομένων πρέπει να επεξεργαστεί κατάλληλα για να βελτιστοποιηθεί η απόδοση του συστήματος.

Η προεπεξεργασία των δεδομένων είναι απαραίτητη, καθώς τα αρχικά δεδομένα πάσχουν από διαφόρων ειδών προβλήματα. Σε αυτά συγκαταλέγονται η ύπαρξη αλληλοσυγκρουόμενων πληροφοριών, η ύπαρξη ασυνεπειών ως προς την κωδικοποίηση, την ονοματοδοσία πεδίων και τις μονάδες μέτρησης, καθώς και η ύπαρξη χαμένων τιμών και θορύβου. Ιδιαίτερα σε δεδομένα κυματομορφών καιρού ή και φορτίου, όπου η μέτρηση εξαρτάται από σένσορες καθώς και την τυχαιότητα της φύσης, η ύπαρξη θορύβου, τυχαία δηλαδή κυμαινόμενων δεδομένων χωρίς ουσιαστικό περιεχόμενο, είναι συχνή και πρέπει να αντιμετωπιστεί. Τα προβληματικά αυτά δεδομένα καλούνται "ακάθαρτα" και η διαδικασία αντιμετώπισης των προβλημάτων τους καλείται "καθαρισμός δεδομένων" (data cleansing).

Η προεπεξεργασία των δεδομένων περιλαμβάνει τον καθαρισμό τους, αλλά δεν περιορίζεται σε αυτόν. Ιδιαίτερα στον τομέα της μηχανικής μάθησης, συχνά επιβάλλεται ο μετασχηματισμός των δεδομένων και ιδιαίτερα η κανονικοποίησή τους. Η κανονικοποίηση είναι η μετατροπή αριθμητικών τιμών σε άλλες, πιο "κατάλληλες" για τον εκάστοτε αλγόριθμο, αριθμητικές τιμές (για παράδειγμα μετασχηματισμός στο διάστημα [0,1]).

Ένα επιπλέον θέμα που εμπίπτει στην προεπεξεργασία των δεδομένων είναι η μείωση ή αύξηση του όγκου τους. Ειδική περίπτωση μείωσης των δεδομένων, με βαρύνουσα σημασία, είναι η επιλογή σημαντικών χαρακτηριστικών, η επιλογή δηλαδή εκείνων των μεταβλητών ή πεδίων που είναι απαραίτητες για την εξόρυξη της γνώσης και η διαγραφή των ακραίων τιμών (outliers) που μπορεί να δυσχεραίνουν την διαδικασία της πρόβλεψης. Αντίθετα, σε κάποιες περιπτώσεις όπου κάποιες σημαντικές κλάσεις στο σύνολο των δεδομένων υποεκπροσωπούνται, εφαρμόζονται τεχνικές επέκτασης όπου δημιουργούνται νέα, τυχαία δεδομένα που προσομοιώνουν τις τιμές των πραγματικών που ανήκουν σε υποεκπροσωπούμενη κλάση. Για παράδειγμα, σε ένα πρόβλημα εντοπισμού απάτης, μπορεί 95 στις 100 επιχειρήσεις στο σύνολο των δεδομένων να λειτουργούν νόμιμα, οπότε και ο αλγόριθμος πρόβλεψης να επιτυγχάνει ακρίβεια 95% προβλέποντας μόνο νόμιμες επιχειρήσεις. Προφανώς αυτό δεν είναι ο στόχος, και άρα σε αυτή τη περίπτωση μπορεί να εφαρμοστεί επέκταση της κλάσης των επιχειρήσεων που είναι απάτες.

Επιπλέον, η αυτόματη ανάκτηση δεδομένων, έρχεται συνήθως με προβλήματα τα οποία δεν περιορίζονται μόνο στην ανάγκη συγχώνευσης διάσπαρτων στοιχείων από διάφορες πηγές και την αποθήκευσή τους σε μια ενιαία βάση. Για παράδειγμα, πολλές φορές διαπιστώνεται η ύπαρξη, διπλοκαταχωρημένων εγγραφών, εγγραφών με αντικρουόμενο περιεχόμενο, τιμές που παραβιάζουν λογικούς κανόνες, εσφαλμένες τιμές (πχ αρνητικές ποσότητες πωλήσεων). Επίσης, ένα πολύ συνηθισμένο πρόβλημα είναι η ύπαρξη χαμένων τιμών, η έλλειψη δηλαδή τιμών σε ορισμένα πεδία καταχωρημένων εγγραφών.

3.5.1 Χαμένες τιμές

Η ύπαρξη χαμένων τιμών (missing values) είναι ένα από τα συνηθέστερα προβλήματα των δεδομένων του πραγματικού κόσμου. Οι λόγοι για αυτό το φαινόμενο είναι πολλοί. Κάποιες πληροφορίες μπορεί να μην ήταν διαθέσιμες την ώρα της καταχώρησης ή μπορεί να διαγράφηκαν αργότερα από λάθος. Η αστοχία υλικού (για παράδειγμα σένσορες) ή λογισμικού είναι ένας άλλος λόγος ύπαρξης χαμένων τιμών.

Οι χαμένες τιμές είναι ένα σημαντικό πρόβλημα στην επεξεργασία δεδομένων γιατί μπορεί να οδηγήσουν σε πλήρη αποτυχία τους αλγορίθμους που εφαρμόζονται. Για τον λόγο αυτό, η επίλυση αυτού του προβλήματος από την αρχή, αλλάζοντας δηλαδή την δομή των προβληματικών γραμμών με χαμένες τιμές, είναι προτιμότερη και περισσότερο ελεγχόμενη.

Στον πίνακα 3.1 παρατηρούμε πως λείπουν τιμές για κάποια πεδία στα δεδομένα φορτίου.

Row	time	temperature	load_observation	load_forecast
2090	2015-03-29 02:00:00.0	10.8	NaN	4326.00
2091	2015-03-29 03:00:00.0	10.5	NaN	NaN
2985	2015-05-05 08:00:00.0	23.0	NaN	6131.00
7150	2015-10-25 21:00:00.0	6.9	NaN	4899.00
10314	2016-03-05 17:00:00.0	7.8	NaN	6471.00

Πίνακας 3.1: Παράδειγμα πραγματικών δεδομένων που περιέχουν μη καταχωρημένες (NaN) τιμές.

Ορισμένοι τρόποι αντιμετώπισης των χαμένων τιμών είναι οι ακόλουθοι:

- **Διαγραφή ολόκληρης της γραμμής.** Ο τρόπος αυτός δεν ενδείκνυται γιατί προκαλεί απώλεια χρήσιμης πληροφορίας. Εφαρμόζεται μόνο σε περιπτώσεις κατά τις οποίες λείπει η τιμή της κλάσης ή σε περιπτώσεις όπου η γραμμή περιέχει πολλές χαμένες τιμές. Δεν χρησιμοποιείται σε δεδομένα χρονοσειράς, αφού απώλεια μιας ολόκληρης γραμμής οδηγεί σε ασυνέπεια στον χρόνο.

- **Αναζήτηση και καταχώρηση της πραγματικής τιμής.** Θεωρητικά αυτή θα ήταν η καλύτερη λύση. Ωστόσο, συνήθως είναι η λιγότερο εφικτή. Κατά κανόνα τα δεδομένα είναι πάρα πολλά και η αναζήτηση των τιμών είναι απαγορευτικά χρονοβόρα, ενώ σε πολλές περιπτώσεις είναι αδύνατη η εύρεση της πραγματικής τιμής.

- **Χρήση μιας σταθερής τιμής για όλες τις χαμένες τιμές.** Όπως για παράδειγμα η τιμή 0. Ο τρόπος δεν ενδείκνυται γιατί οι αλγόριθμοι που θα επεξεργαστούν τα δεδομένα μπορεί να εκλάβουν την τιμή αυτή ως έγκυρη, να τη συμπεριλάβουν στην επεξεργασία και να οδηγηθούν σε εσφαλμένα συμπεράσματα.

- **Αντικατάσταση της χαμένης τιμής.** Αυτό συνήθως γίνεται με τη μέση τιμή της στήλης αν το πεδίο είναι αριθμητικό.

- **Πρόβλεψη της χαμένης τιμής.** Σύμφωνα με τον τρόπο αυτό, το πρόβλημα της χαμένης τιμής αντιμετωπίζεται σαν πρόβλημα κατηγοριοποίησης (αν το πεδίο είναι ονομαστικό) ή παλινδρόμησης (αν το πεδίο είναι αριθμητικό). Αναπτύσσεται ένα μοντέλο ικανό να υπολογίζει τις τιμές της στήλης με τη χαμένη τιμή από τα δεδομένα των άλλων στηλών. Η πιθανότερη τιμή που θα υπολογιστεί με τη χρήση του μοντέλου αντικαθιστά τη χαμένη τιμή.

Η αντικατάσταση των κενών με τη μέση τιμή, είναι ο τρόπος με τον οποίο επιλύεται το πρόβλημα στην παρούσα εργασία. Συγκεκριμένα, για την αποθήκευση και διαχείριση των δεδομένων, χρησιμοποιείται το module pandas της python.

Αυτή η βιβλιοθήκη καθιστά εύκολη την αντιμετώπιση του προβλήματος χαμένων τιμών καθώς και την ύπαρξη διπλοκαταχωρημένων εγγραφών. Με το παρακάτω κομμάτι κώδικα, διαγράφονται οι

duplicate εγγραφές και αντικαθιστάται με τη μέση τιμή το κάθε κενό πεδίο στις στήλες load_observation και load_forecast.

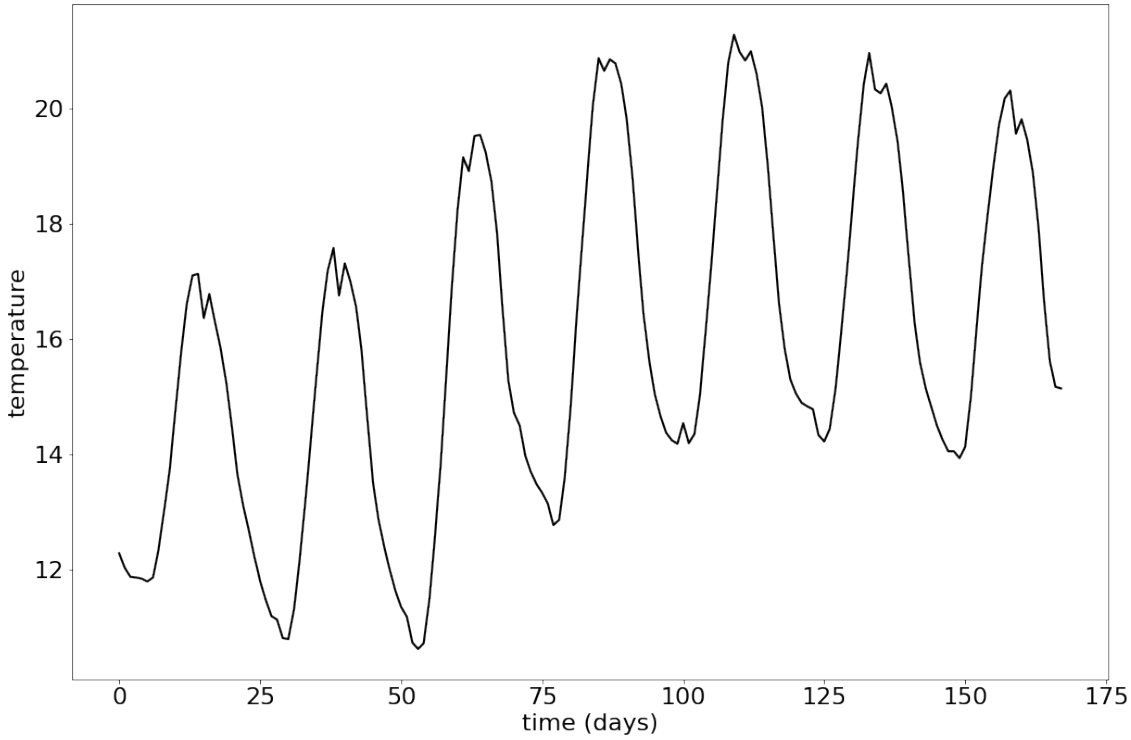
```
if(dataset.isnull().values.any()):
  print("Some row misses values")
  dataset['load_forecast'].fillna((dataset['load_forecast'].mean()), inplace=True)
  dataset['load_forecast'].fillna((dataset['load_forecast'].mean()), inplace=True)
  print("Replaced with mean")
  dataset = dataset.drop_duplicates(subset='time', keep="last")
  print("Dropped duplicates")
```

Σχήμα 3.5: Κομμάτι κώδικα για την διαχείρηση χαμένων τιμών και διπλοκαταχωρημένων εγγραφών με χρήση pandas-python.

3.5.2 Εξάλειψη Θορύβου

Θορυβώδη ονομάζονται τα δεδομένα τα οποία περιέχουν εσφαλμένες τιμές που έχουν προκύψει τυχαία, τιμές δηλαδή που δεν προσφέρουν χρήσιμη πληροφορία στην ανάλυση. Η ύπαρξη θορύβου προκαλεί προβλήματα στους αλγορίθμους εξόρυξης και πρέπει να αντιμετωπιστεί στα πλαίσια της προεπεξεργασίας των δεδομένων.

Στο παρακάτω σχήμα (3.6) παρουσιάζεται σήμα εισόδου το οποίο περιέχει θόρυβο.



Σχήμα 3.6: Σήμα εισόδου θερμοκρασιών μιας εβδομάδας. Περιέχεται θόρυβος σε μορφή *spike noise* που παρατηρείται κυρίως στα σημεία καμπής.

Η τακτική αντιμετώπισης θορύβου που χρησιμοποιείται στην παρούσα εργασία, είναι η εφαρμογή φίλτρων στα σήματα εισόδου. Στην επεξεργασία σήματος, ένα ψηφιακό φίλτρο είναι ένα σύστημα που εκτελεί μαθηματικές πράξεις σε ένα δείγμα εισόδου για να μειώσει ή να ενισχύσει ορισμένες πτυχές αυτού του σήματος. Τα φίλτρα που χρησιμοποιούνται για την αποθορυβοποίηση των σημάτων παρουσιάζονται παρακάτω.

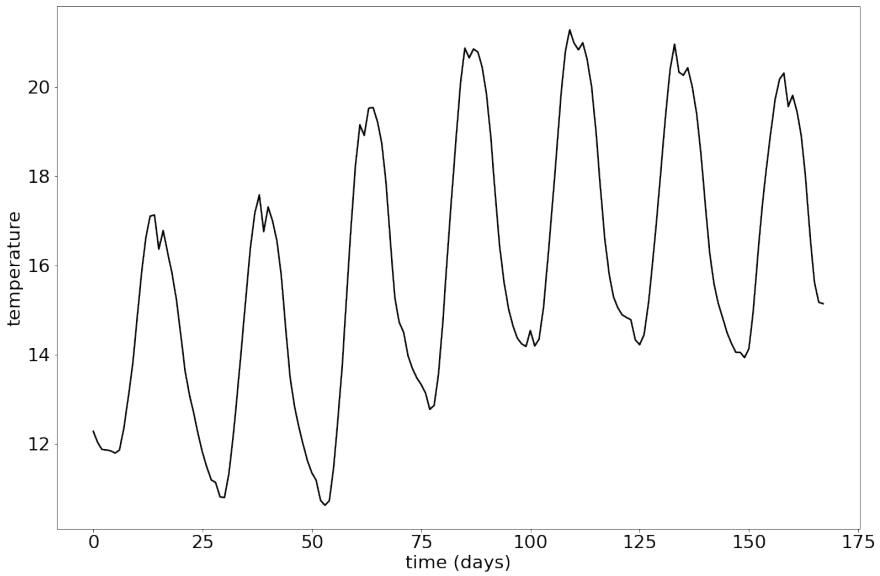
- **Median filter.** Το φίλτρο της διαμέσου τιμής είναι μια μη γραμμική τεχνική επεξεργασίας σήματος χρήσιμη για την κατάπνιξη θορύβου. Προτάθηκε ως εργαλείο ανάλυσης χρονοσειρών [Tuke77] και αργότερα τέθηκε σε χρήση και στην επεξεργασία εικόνων.

  Η διαδικασία εκτελείται αφήνοντας ένα παράθυρο να κυλίεται πάνω από τα σημεία μιας ακολουθίας, αντικαθιστώντας την τιμή στο κέντρο του παραθύρου με τη διάμεσο τιμή των πρωτοτύπων τιμών του παραθύρου. Στην έξοδο παράγεται μια ακολουθία η οποία συνήθως είναι πιο ομαλή από την αρχική.

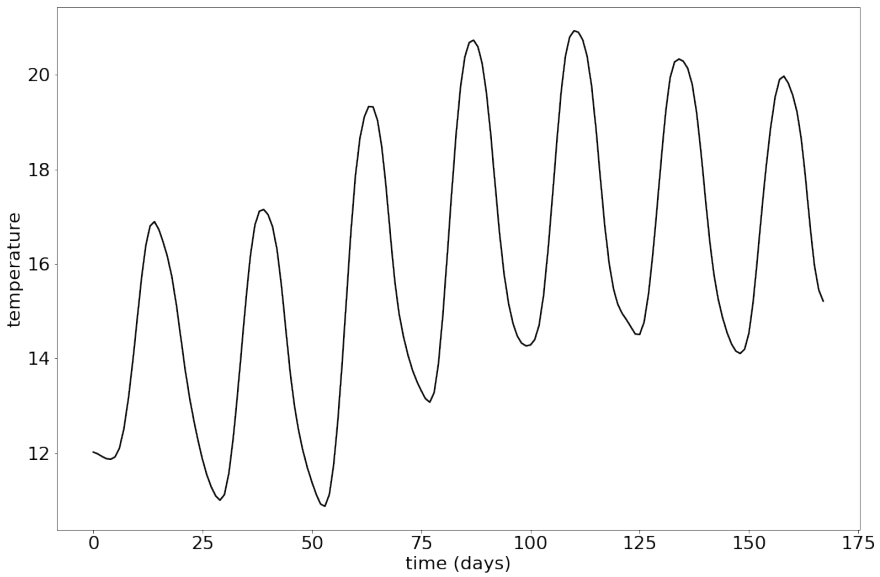
- **Gaussian filter.** Το Γκαουσιανό φίλτρο, στηρίζεται στην πράξη της συνέλιξης μεταξύ δύο σημάτων που έχει αναλυθεί στην ενότητα 2.3.1. Το σήμα εισόδου, συνελίσεται με έναν πίνακα (kernel) το οποίο έχει τιμές μιας κανονικής ή Γκαουσιανής κατανομής εξού και το όνομα του φίλτρου.

Στην έξοδο, παράγεται ένα σήμα εξόδου, το οποίο είναι πιο ομαλό από το αρχικό. Μεγαλύτερη λείανση (smoothing) του σήματος επιτυγχάνεται για μεγαλύτερη τιμή της τυπικής απόκλισης σ του τύπου της κανονικής (Gauss) κατανομής $f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}$

Τα αποτελέσματα της εφαρμογής των φίλτρων σε δεδομένα εισόδου, φαίνονται στο Σχ. 3.7. Η αλλαγή είναι εμφανής, τα θορυβώδη spikes έχουν εξομαλυνθεί, διατηρώντας την πραγματική δομή των δεδομένων.



(a)



(b)

Σχήμα 3.7: Αποτελέσματα της αποθορυβοποίησης των δεδομένων εισόδου. Οι χρονοσειρές πριν (**a**) και μετά (**b**) την εφαρμογή.

Στην πράξη, εφαρμόζουμε διαδοχικά ένα median και ένα gaussian φίλτρο στα δεδομένα εισόδου, όπως φαίνεται στο παρακάτω κομμάτι κώδικα.

```
if filter_on == 1:
  for inp in INPUTS:
    dataset[inp] = medfilt(dataset[inp], 3)
    dataset[inp] = gaussian_filter1d(dataset[inp], 1.2)
```

Σχήμα 3.8: Κώδικας για την εφαρμογή φίλτρων median και gaussian στα δεδομένα.

3.5.3 Κανονικοποίηση

Η κανονικοποίηση (normalization) είναι μια διαδικασία μετασχηματισμού δεδομένων, κατά την οποία αριθμητικές τιμές αντικαθίστανται με άλλες, πιο κατάλληλες, αριθμητικές τιμές. Η κανονικοποίηση των δεδομένων γίνεται ώστε να αντιμετωπιστούν δυσκολίες ορισμένων μεθόδων ανάλυσης και εξόρυξης γνώσης. Για παράδειγμα, τα Νευρωνικά Δίκτυα λειτουργούν καλύτερα όταν οι τιμές εισόδου κυμαίνονται στην περιοχή [0,1]. Πρακτικά το διάστημα διακύμανσης των δεδομένων προσαρμόζεται στο εύρος εξόδου των συναρτήσεων ενεργοποίησης (σιγμοειδής).

Μια επαρκής εξομάλυνση, όχι μόνο των μεταβλητών εξόδου του δικτύου (μέσω των συναρτήσεων ενεργοποίησης) αλλά και των εισόδων, πριν από τη διαδικασία εκπαίδευσης, είναι πολύ σημαντική για την επίτευξη καλών αποτελεσμάτων καθώς και για τη σημαντική μείωση του χρόνου υπολογισμών, άρα και εκπαίδευσης [Sola97].

Υπάρχουν διάφορες μέθοδοι κανονικοποίησης των αριθμητικών τιμών. Ορισμένες από τις πλέον χρησιμοποιούμενες είναι οι ακόλουθες:

- **Κανονικοποίηση ελάχιστου-μέγιστου.** Με αυτήν τη μέθοδο κανονικοποίησης, οι αριθμητικές τιμές αντιστοιχίζονται με άλλες, οι οποίες κυμαίνονται εντός μιας προκαθορισμένης περιοχής τιμών. Η αντιστοίχιση γίνεται με γραμμικό μετασχηματισμό. Αν θεωρήσουμε μια μεταβλητή , όπου η μεγαλύτερη τιμή της είναι max_A και η μικρότερη τιμή της είναι η min_A, μπορούμε να αντιστοιχίσουμε όλες τις τιμές με άλλες που κυμαίνονται εντός μιας περιοχής με κατώτερο όριο την new_min_A και ανώτερο όριο την new_max_A σύμφωνα με τη

  $x' = \frac{x - min_A}{max_A - min_A}(new_max_A - new_min_A) + new_min_A$

  όπου x η εκάστοτε τιμή της μεταβλητής και x' η νέα τιμή. Η μέθοδος αυτή έχει το πλεονέκτημα ότι ο χρήστης προκαθορίζει την περιοχή τιμών, για παράδειγμα μπορεί να μετασχηματίσει τις τιμές έτσι ώστε να κυμαίνονται στην περιοχή [0,1], ορίζοντας σαν new_min_A την τιμή 0 και σαν new_max_A την τιμή 1. Επίσης, με τη μέθοδο αυτή διατηρείται η αναλογία μεταξύ των τιμών που υπήρχε στα αρχικά δεδομένα και είναι πολύ απλός και ο αντίστροφος μετασχηματισμός των δεδομένων στο αρχικό διάστημα.

- **Κανονικοποίηση z-score.** Η στατιστική αυτή μέθοδος, πραγματοποιεί μετασχηματισμό των αριθμητικών τιμών, χρησιμοποιώντας τη μέση τιμή και την τυπική απόκλιση τους. Για μία μεταβλητή , με μέση τιμή μ και τυπική απόκλιση σ, ο μετασχηματισμός των τιμών γίνεται σύμφωνα με τη σχεση

  $x' = \frac{x - \mu}{\sigma}$

  όπου x η εκάστοτε τιμή της μεταβλητής και x' η νέα τιμή.

  Η μέθοδος αυτή είναι ιδιαίτερα κατάλληλη σε περιπτώσεις όπου τα δεδομένα περιέχουν ακραίες τιμές, γιατί η κανονικοποίηση ελάχιστου-μέγιστου θα συγκέντρωνε τη μεγάλη πλειοψηφία των τιμών σε ένα ελάχιστο τμήμα της περιοχής των νέων τιμών. Επίσης, η μέθοδος δίνει κανονικοποιημένες τιμές, δηλαδή τιμές των οποίων η μέση τιμή είναι μηδενική.

Η μέθοδος κανονικοποίησης που χρησιμοποιείται στη παρούσα εργασία, είναι αυτή του ελαχίστου-μεγίστου. Η υλοποίηση γίνεται με το παρακάτω κομμάτι κώδικα χρησιμοποιώντας την βιβλιοθήκη sklearn της python.

```
scalers  = []
for inp in INPUTS:
  scaler = MinMaxScaler(feature_range = (0, 1))

  temp =  dataset

  train_set = temp[0:int(len(temp)*train_days_percentage)].reset_index(drop=True)
  test_set = temp[int(len(temp)*train_days_percentage):].reset_index(drop=True)


  train_set = train_set[inp].values.reshape(-1,1)
  test_set = test_set[inp].values.reshape(-1,1)

  scaler.fit(array(train_set))
  scalers.append(scaler)
```

Σχήμα 3.9: Κώδικας κανονικοποίησης των δεδομένων εισόδου. Γίνεται χρήση του MinMaxScaler της βιβλιοθήκης sklearn.

Όπως φαίνεται και από το σχήμα 3.4, κάθε σήμα εισόδου έχει διαφορετικές τιμές. Για αυτό το λόγο, εκπαιδεύεται ένας min-max scaler για κάθε είσοδο ξεχωριστά. Τα αντικείμενα αυτά αποθηκεύονται και μπορούν να χρησιμοποιηθούν στην φάση της δοκιμής (testing phase) για την αντιστροφή των προβλέψεων στις πραγματικές μονάδες. Οι τιμές ελαχίστου μεγίστου, με τις οποίες κανονικοποιούμε τα δεδομένα, προκύπτουν μόνο από το σύνολο εκπαίδευσης καθώς η γνώση του συνόλου δοκιμής δεν πρέπει να θεωρείται γνωστή.

3.6 Δεδομένα φορτίου

Κύριος σκοπός της παρούσας εργασίας είναι η μελέτη των δεδομένων φορτίου και η διεξαγωγή προβλέψεων για αυτά. Οι προβλέψεις φορτίου είναι σημαντικές για μια αξιόπιστη και οικονομικά σταθερή λειτουργία ενός συστήματος ισχύος, όπως μια επιχείρηση παροχής ηλεκτρικής ενέργειας εφόσων, μια εκτίμηση φορτίου πάνω από την πραγματική, υποχρεώνει τους Διαχειριστές Μεταφοράς Ηλεκτρικής Ενέργειας (ΑΔΜΗΕ) να δεσμεύουν ακριβότερες μονάδες παραγωγής ενέργειας οδηγώντας σε υψηλότερες τιμές στην αγορά ενέργειας.

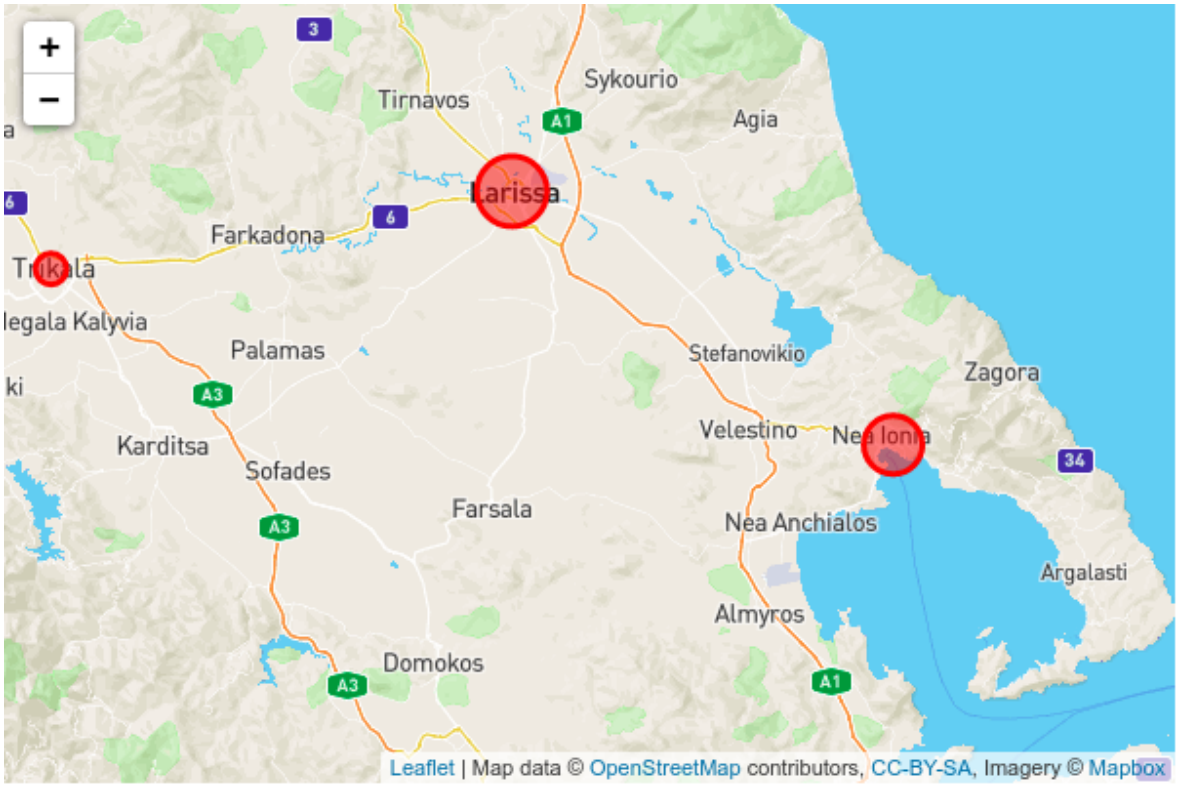
Τα δεδομένα αυτά δημοσιοποιούνται για ευρωπαϊκές χώρες από το European Network of Transmission System Operators for Electricity (ENTSO-E), το οποίο συλλέγει δεδομένα που του αποστέλλουν οι κατά τόπους διαχειριστές συστημάτων διανομής (Transmission System Operators - TSOs), στην Ελλάδα ο ΑΔΜΗΕ. Αν και η ανάλυση πιο συγκεκριμένων τιμών ,σε επίπεδο περιφέρειας για παράδειγμα, θα είχαν ενδιαφέρον, τέτοια δεδομένα δεν διατίθενται από το ENTSO-E. Μπορούν να αναζητηθούν σε δεδομένα μετρήσεων SCADA που διαθέτει ο ίδιος ο ΑΔΜΗΕ, ωστόσο οι απαιτούμενες διορθώσεις τους (συμμετοχή ΑΠΕ) είναι εκτός της εμβέλειας της εργασίας αυτής. Επίσης, η χωρική αναφορά των δεδομένων SCADA με την ακρίβεια που απαιτείται για αξιοποίηση της στη διερεύνηση της παρούσας εργασίας, δεν είναι διαθέσιμη. Οπότε οι πειραματισμοί έγιναν σε δεδομένα πανελλαδικού επιπέδου παρεχόμενα από το ENTSO-E. Προφανώς, όλες οι τεχνικές προεπεξεργασίας που περιγράφονται σε αυτό το κεφάλαιο, εφαρμόζονται και στα δεδομένα φορτίου.

Η κατανάλωση ενέργειας σε επίπεδο κρατών έχει εξίσου μεγάλη σημασία. Η ενέργεια παίζει ζωτικό ρόλο στην πρόοδο και την κοινωνικοοικονομική ανάπτυξη μιας χώρας. Ο καιρός επίσης είναι βασική κινητήρια δύναμη της κατανάλωσης ηλεκτρικής ενέργειας. Μοντέλα βασισμένα σε μετεωρολογικά δεδομένα χρησιμοποιούνται συχνά στη πρόβλεψη ηλεκτρικού φορτίου.

Καθώς είχαμε στην διάθεσή μας δεδομένα θερμοκρασίας από το 2015 ως το 2019 του Εθνικού Αστεροσκοπείου Αθηνών για κάποιες από τις μεγαλύτερες πόλεις της ηπειρωτικής Ελλάδας, αποφασίστηκε η συγχώνευση των τιμών θερμοκρασίας των παραπάνω περιοχών σε μια μόνο τιμή. Αυτή η πανελλαδική τιμή θερμοκρασίας, μπορεί να συνδυαστεί με τα δεδομένα φορτίου για την βελτίωση των προβλέψεων. Η συγχώνευση γίνεται μέσω της στάθμισης των τιμών θερμοκρασίας, ανάλογα με τον πληθυσμό της εκάστοτε περιοχής.

Παρακάτω (Σχ. 3.10) παρουσιάζεται ένας χάρτης στον οποίο φαίνεται ο βαθμός επίδρασης των τιμών της κάθε περιοχής στο τελικό αποτέλεσμα.

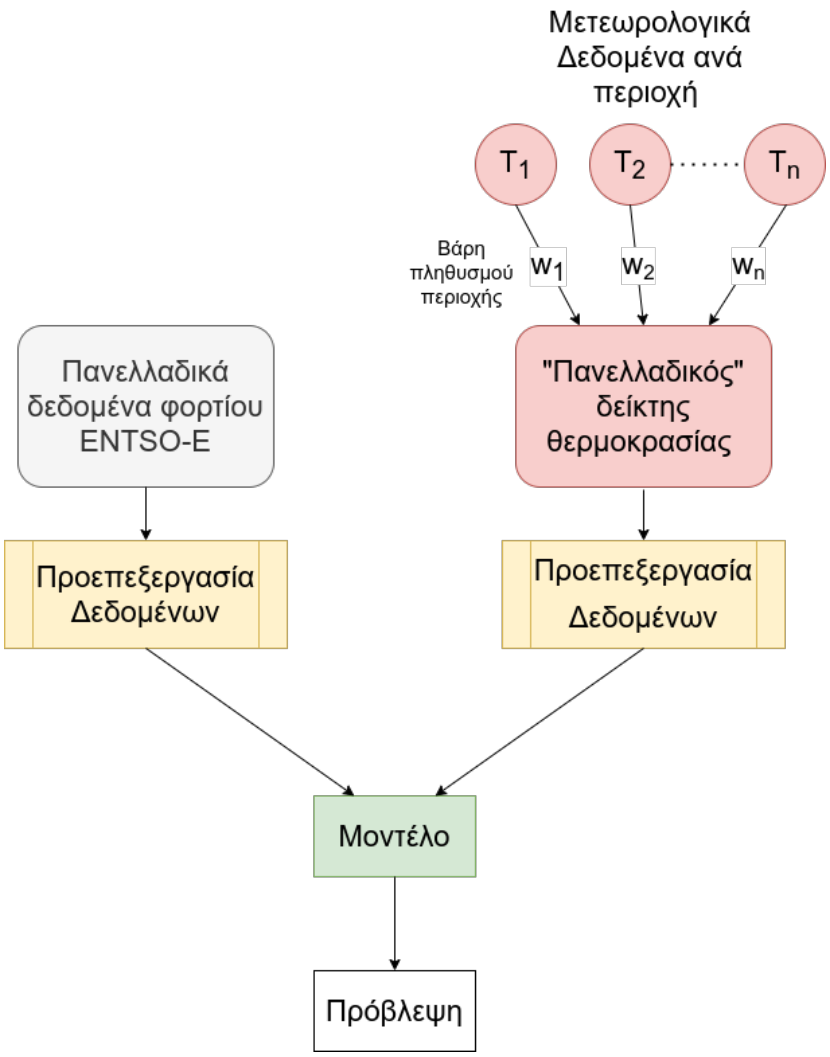
(a)



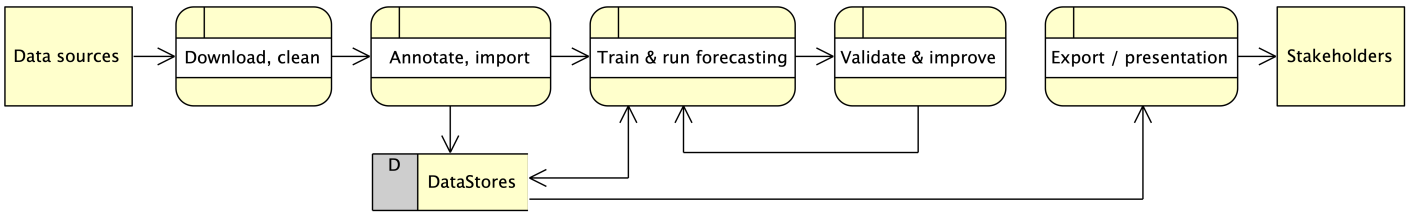
(b)

Σχήμα 3.10: Χάρτης επίδρασης των δεδομένων καιρού ανά περιοχή. (**a**) για όλη την ηπειρωτική Ελλάδα και (**b**) για την περιφέρεια Θεσσαλίας.

Τέλος, στο σχήμα Σχ. 3.11, παρουσιάζεται διάγραμμα ροής των δεδομένων φορτίου και καιρού για την διεξαγωγή πρόβλεψης.



Σχήμα 3.11: Διάγραμμα ροής δεδομένων. Περιγράφεται σχηματικά η διαδικασία εκπαίδευσης των μοντέλων.

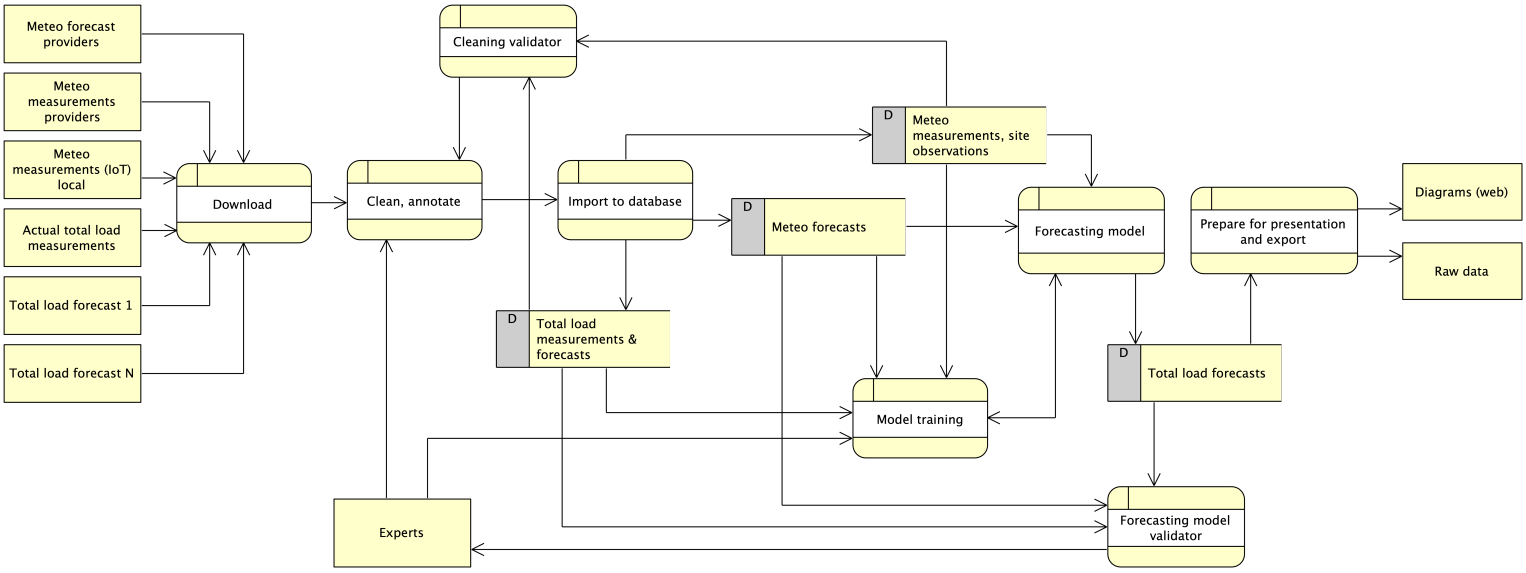


Σχήμα 3.12: Διάγραμμα ροής δεδομένων. Περιγράφεται συλλογικά η διαδικασία απόκτησης, επεξεργασίας δεδομένων, εκπαίδευσης μοντέλων και εξαγωγής συμπερασμάτων.

Κεφάλαιο 4

Πειράματα-Αποτελέσματα

Σε αυτό το κεφάλαιο, παρουσιάζεται η διαδικασία εκτέλεσης των πειραμάτων που αναλύθηκαν. Παρουσιάζονται τα αποτελέσματα και τα συμπεράσματα.



Σχήμα 4.1: Διάγραμμα ροής δεδομένων. Περιγράφεται σχηματικά η διαδικασία διεξαγωγής των πειραμάτων.

4.1 Διαδικασία εκπαίδευσης

Μετά την προεπεξεργασία των δεδομένων, ακολουθεί η διαδικασία της εκπαίδευσης η οποία είναι κοινή με μικρές παραλλαγές για όλα τα πειράματα που ακολουθούν. Τα κανονικοποιημένα και αποθορυβοποιημένα δεδομένα, χωρίζονται σε σύνολο εκπαίδευσης (training set) και σύνολο δοκιμής (testing set) είτε βάσει ενός ποσοστού (για παράδειγμα 80-20) είτε χωρίζεται με βάση την ημερομηνία. Στη παρούσα εργασία ο διαχωρισμός γίνεται με βάση την ημερομηνία και συγκεκριμένα επιλέγονται χρονοσειρές έτους 2015-2018 για την εκπαίδευση και το έτος 2019 για το σύνολο δοκιμής.

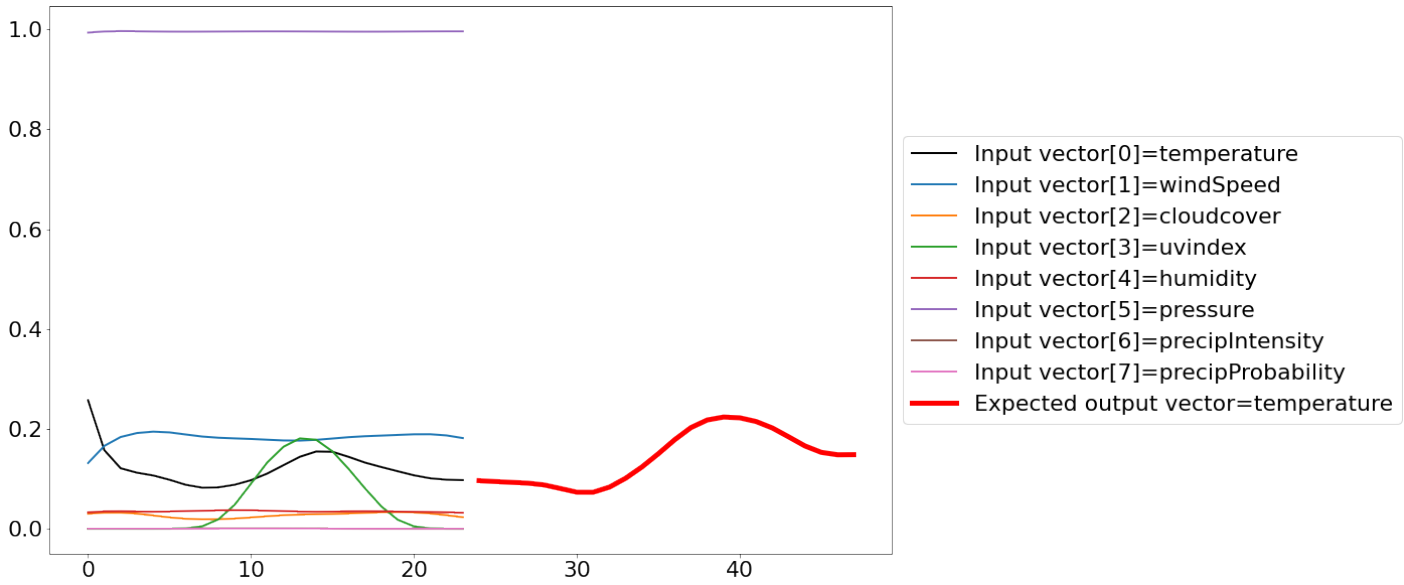
Επίσης, για να εκπαιδευτούν τα μοντέλα, πρέπει να οριστεί μια μεταβλητή η οποία ελέγχει το διάστημα ημερών εισόδου, δηλαδή τις τιμές με βάση τις οποίες το μοντέλο θα κάνει την πρόβλεψή του, καθώς και μια για τον έλεγχο των αναμενόμενων τιμών εξόδου. Τα ζευγάρια εισόδου-εξόδου, δημιουργούνται με μια συνάρτηση *data_split* η οποία δέχεται το σύνολο των δεδομένων και τα "σπάει" κατάλληλα σε πολλά ζευγάρια χρησιμοποιώντας ένα ολισθόμενο παράθυρο σταθερού μήκους το οποίο μετατοπίζεται ανα μία ώρα (εφόσων και το dataset είναι ωριαίας ακρίβειας).

```
1  def data_split(sequences, out, n_timestamp, n_pred):
2      X = []
3      y = []
4      for i in range(len(out)):
5        end_ix = i + n_timestamp
6        end_iy = end_ix + n_pred
7        seq_x = []
8        if end_iy > len(out)-1:
9          break
10       for seq in sequences:
11         seq_x.append(seq[i:end_ix])
12
13       seq_y = out[end_ix:end_iy]
14       X.append(seq_x)
15       y.append(seq_y)
16     return array(X), array(y)
```

Σχήμα 4.2: Κώδικας δημιουργίας των δεδομένων εκπαίδευσης σε μορφή εισόδων-αναμενόμενης εξόδου.

Η συνάρτηση επιστρέφει έναν πίνακα X ο οποίος περιέχει όλα τα διανύσματα εισόδου καθώς και έναν πίνακα y ο οποίος περιέχει την αναμενόμενη απόκριση για κάθε είσοδο στην αντίστοιχη θέση. Έτσι για ένα διάστημα 5 ετών, ξεκινώντας από το 2015 μέχρι και το 2018 για το training set δημιουργούνται 26064 ζευγάρια-σημεία δεδομένων και για το testing set, το οποίο περιέχει την χρονιά 2019, προκύπτουν 17312 ζευγάρια. Αυτά τα σύνολα, κρίθηκαν αναγκαία και ικανά, τόσο για την εκπαίδευση, όσο και για την εκτίμηση της επίδοσης του μοντέλου.

Στο σχήμα Σχ. 4.3 παρουσιάζεται ένα ζευγάρι κυματομορφών εισόδου και αναμενόμενης εξόδου.



Σχήμα 4.3: Δείγμα της μορφής των δεδομένων εκπαίδευσης του μοντέλου. Χωρίζονται σε δεδομένα εισόδου και αναμενόμενης εξόδου.

Ο ορισμός του μοντέλου (LSTM) γίνεται πολύ απλά και σε λίγες γραμμές κώδικα, χρησιμοποιώντας την βιβλιοθήκη μηχανικής μάθησης keras της python. Το μοντέλο αρχικοποιείται ως Sequential, δηλαδή τα στοιχεία του θα οριστούν σειριακά ως κελιά.

```
model = Sequential()
model.add(LSTM(50, activation='relu', return_sequences=True,
input_shape=(X_train.shape[1], X_train.shape[2])))
model.add(LSTM(50, activation='relu'))
model.add(Dense(y_train.shape[1], activation='linear'))
model.compile(optimizer = 'adam', loss = 'mean_squared_error')
```

Σχήμα 4.4: Κώδικας κατασκευής ενός μοντέλου LSTM. Χρησιμοποιείται η βιβλιοθήκη μηχανικής μάθησης keras.

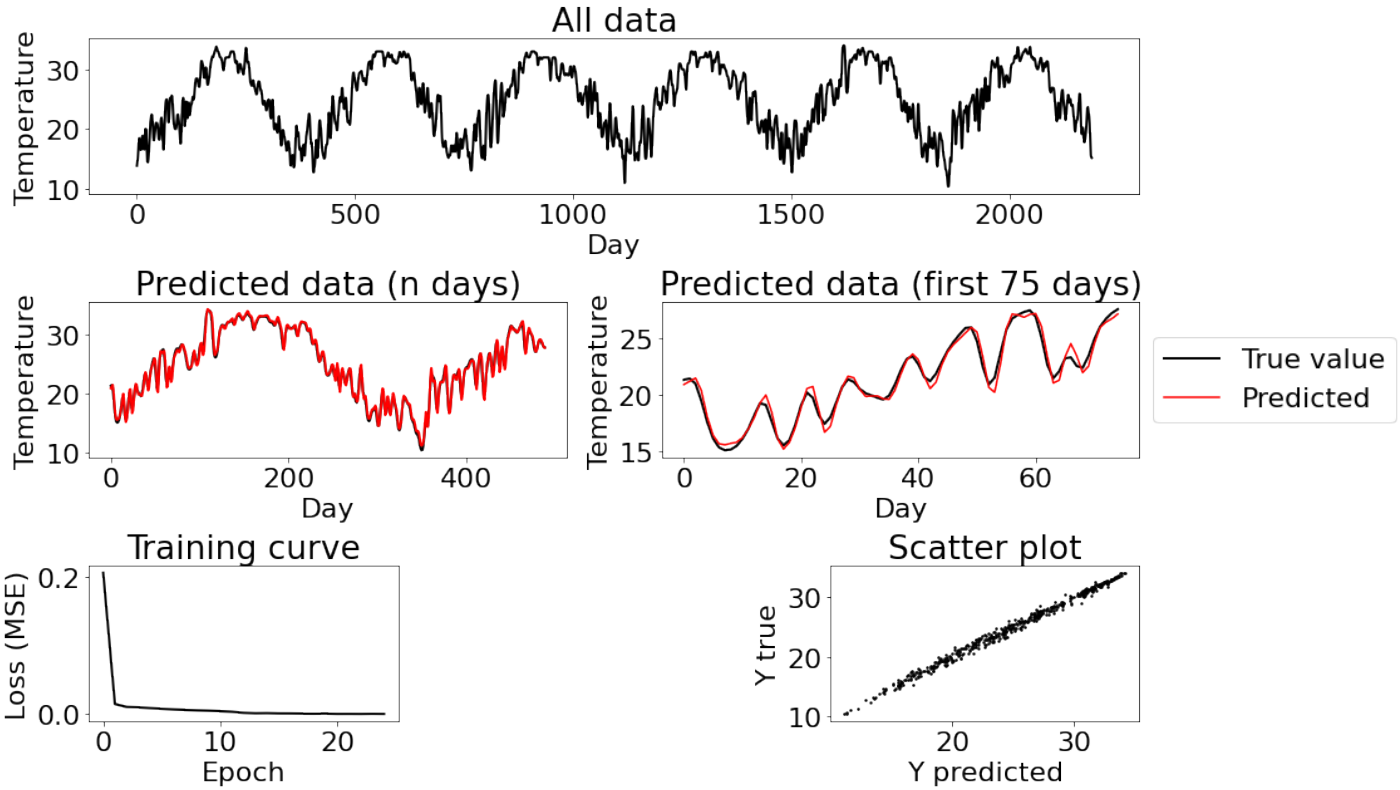
Επομένως, η ανάπτυξη ενός μοντέλου μηχανικής μάθησης απλοποιείται πολύ και μπορεί να χρησιμοποιηθεί ακόμα και από ερευνητές οι οποίοι δεν έχουν άριστη γνώση της λειτουργίας του κάθε στοιχείου ή και του κώδικα που απαιτείται. Ο λόγος έγκειται στο ότι τα κελιά αυτά μπορούν να θεωρηθούν ”μαύρα κουτιά”.

Η εκπαίδευση ξεκινάει με την εντολή model.fit η οποία δέχεται ως παραμέτρους, τους πίνακες X και y που έχουν παραχθεί από την συνάρτηση data_split, τον αριθμό των εποχών εκπαίδευσης, δηλαδή του πλήθους επαναλήψεων ανάγνωσης και επεξεργασίας του συνόλου δεδομένων, καθώς και το batch size. Επιστρέφεται ένα αντικείμενο το οποίο είναι χρήσιμο για την ανάλυση της επίδοσης του μοντέλου,, αφού παρατηρείται η τιμή της συνάρτησης κόστους ανά εποχή.

4.2 Επίδραση της προεπεξεργασίας των δεδομένων στην επίδοση

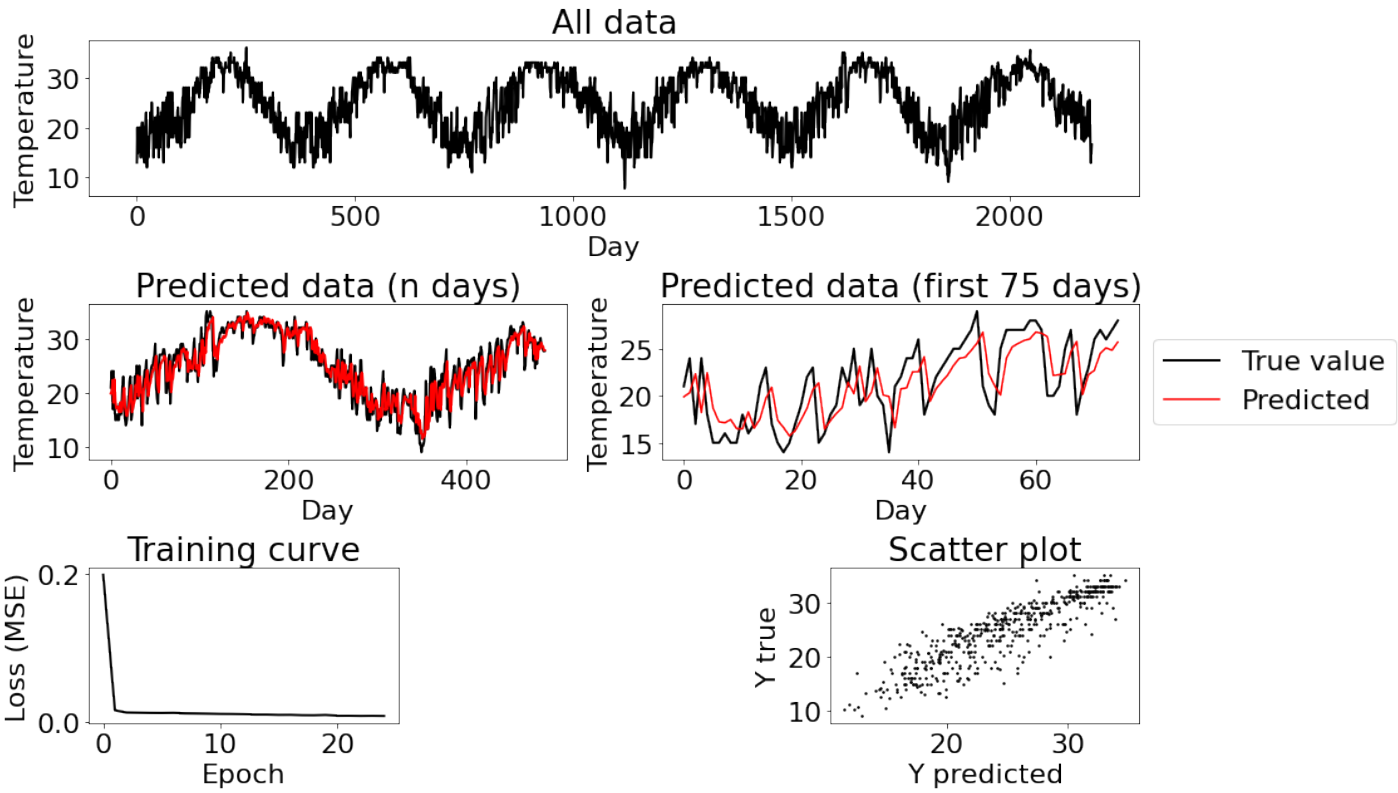
Σε αυτό το σημείο έγινε πειραματισμός για την επίδοση του συστήματος σε συνάρτηση με το αν εφαρμόστηκε αποθορυβοποίηση στα δεδομένα. Στην ενότητα 3.5.2 αναφερθήκαμε στις τεχνικές εξάλειψης θορύβου που εφαρμόζονται καθώς και στη σημασία και την επίδραση που έχουν στην βέλτιστη λειτουργία των αλγορίθμων μηχανικής μάθησης.

Στην πράξη, εκπαιδεύσαμε το ίδιο δίκτυο σε ένα μικρό πλήθος δεδομένων. Αρχικά εφαρμόζοντας τα φίλτρα αποθορυβοποίησης, πήραμε πολύ καλά αποτελέσματα όπως φαίνεται στο σχήμα 4.5.



Σχήμα 4.5: Διαγράμματα ανάλυσης επίδοσης με εξάλειψη θορύβου.

Έπειτα, απενεργοποιήσαμε τα φίλτρα και εκπαιδεύσαμε εκ νέου το μοντέλο. Τα αποτελέσματα ήταν, αναμενόμενα, χειρότερα όπως φαίνεται και στο σχήμα 4.6.



Σχήμα 4.6: Διαγράμματα ανάλυσης επίδοσης χωρίς εξάλειψη θορύβου.

Η πραγματική και συνολική διαφορά των πειραμάτων, φαίνεται και στα Scatter plots. Αυτά τα διαγράμματα (κατω δεξιά) απεικονίζουν σημεία της μορφής (πρόβλεψη, πραγματικότητα) στο δισδιάστατο επίπεδο. Προφανώς αν το σύστημα ήταν ιδανικό και δεν έκανε λάθος προβλέψεις, όλα τα σημεία θα βρισκόταν στην ευθεία $x = y$.

Όσο τα σημεία αποκλίνουν από αυτή την ευθεία. τόσο χειρότερη επίδοση έχει το σύστημα. Η διαφορά λοιπόν είναι εμφανής. Όταν τα δεδομένα κανονικοποιούνται, τα σημεία συγκεντρώνονται γύρω από την ευθεία, ενώ όταν δεν χρησιμοποιείται κανονικοποίηση έχουμε απόκλιση.

4.3 Επίδοση μοντέλων Μηχανικής Μάθησης

Η επιλογή του μοντέλου μηχανικής μάθησης, όπως περιγράφεται και στο κεφάλαιο 2 είναι καθοριστική για την επίτευξη της καλύτερης δυνατής πρόβλεψης. Για αυτό το λόγο, πριν την εκτέλεση πιο σύνθετων πειραμάτων, βασικός στόχος υπήρξε η επιλογή του καλύτερου δυνατού μοντέλου για την ανάλυση χρονοσειρών.

Η εκπαίδευση των μοντέλων γίνεται σε ένα μικρό σύνολο δεδομένων και η επίδοση τους μετριέται στα δεδομένα δοκιμής (test set). Όμως υπάρχουν διάφορες μετρικές με βάση τις οποίες μπορούν να συγκριθούν δύο συστήματα. Προφανώς οι μετρήσεις αυτές είναι συναρτήσεις ομοιότητας μεταξύ διανυσμάτων πρόβλεψης και των πραγματικών παρατηρήσεων. Οι πιο γνωστές και ευρέως χρησιμοποιούμενες είναι οι εξής:

- **Mean Absolute Error**

  Το μέσο απόλυτο σφάλμα είναι ο μέσος όρος της διαφοράς μεταξύ των πραγματικών τιμών και των προβλέψεων. Μας δίνει το μέτρο για το πόσο μακριά ήταν οι προβλέψεις από την πραγματική έξοδο. Ωστόσο, δεν μας δίνουν καμία πληροφορία για την κατεύθυνση του σφάλματος. Μαθηματικά, αναπαρίσταται ως:

  $$MAE(y_i, \hat{y}_i) = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|$$

  όπου y το διάνυσμα πραγματικών τιμών και $\hat{y}$ το διάνυσμα προβλέψεων.

- **Mean Squared Error**

  Το μέσο τετράγωνο σφάλμα (MSE) είναι αρκετά παρόμοιο με το μέσο απόλυτο σφάλμα, η μόνη διαφορά είναι ότι το MSE παίρνει το μέσο όρο του τετραγώνου της διαφοράς μεταξύ των διανυσμάτων παρατηρήσεων και των προβλέψεων. Δίνεται από τη σχέση

  $$MSE(y_i, \hat{y}_i) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

  όπου y το διάνυσμα πραγματικών τιμών και $\hat{y}$ το διάνυσμα προβλέψεων. Το πλεονέκτημα του MSE είναι ότι είναι πιο εύκολο να υπολογιστεί η παράγωγός του ,οδηγώντας σε ταχύτερη εκπαίδευση, σε αντίθεση με το μέσο απόλυτο σφάλμα το οποίο απαιτεί πιο περίπλοκους υπολογισμούς. Καθώς παίρνουμε το τετράγωνο του σφάλματος, η επίδραση των μεγαλύτερων σφαλμάτων γίνεται πιο έντονη σε σχέση με τα μικρότερα, επομένως το μοντέλο, μαθαίνοντας, μπορεί να εστιάσει περισσότερο στα μεγαλύτερα λάθη.

- **R^2 (R-squared)**

  Αποτελεί μια κανονικοποιημένη εκδοχή του MSE. Δεν προσφέρει περισσότερη πληροφορία από το μέσο τετραγωνικό σφάλμα, όμως είναι χρήσιμο καθώς δεν εξαρτάται από την κλίμακα των δεδομένων και επειδή είναι συχνά ευκολότερο να ερμηνευθεί από ανθρώπους. Πρακτικά επιστρέφει μια τιμή στο διάστημα [0,1] όπου τιμή κοντινότερη στη μονάδα σημαίνει μικρή απόκλιση πρόβλεψης-πραγματικότητας, ενώ τιμή κοντά στο 0 σημαίνει μεγάλη απόσταση των διανυσμάτων. Μαθηματικά ορίζεται ως:

  $$R^2(y_i, \hat{y}_i) = 1 - \frac{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}$$

  όπου y το διάνυσμα πραγματικών τιμών και $\hat{y}$ το διάνυσμα προβλέψεων και $\bar{y}$ ο μέσος όρος των τιμών του y.

Με βάση τις παραπάνω μετρικές αξιολογούνται τα μοντέλα LSTM ενός επιπέδου, Stacked LSTM δύο επιπέδων, Bidirectional LSTM και CNN. Η δομή αυτών των δικτύων καθώς και η λειτουργία τους, έχουν αναλυθεί στην ενότητα 2.3. Τα αποτελέσματα των δοκιμών φαίνονται στον πίνακα 4.1 και είναι εμφανής η υπεροχή του Stacked LSTM τόσο στο MSE όσο και στο MAE και R^2. Αυτός είναι και ο λόγος που το μοντέλο αυτό, επιλέγεται ως "βασικό" και χρησιμοποιείται στους υπόλοιπους πειραματισμούς. Η επιλογή επιβεβαιώνεται και από την βιβλιογραφία καθώς τα Stacked LSTM χρησιμοποιούνται για ανάλυση χρονοσειρών καιρού ακόμα και αν πρόκειται για τιμές πολλών γεωγραφικών περιοχών [Kare18].

Model	**MSE**	R^2	**MAE**
LSTM	1.12	0.97	0.82
Stacked LSTM	0.24	0.99	0.38
Bidirectional LSTM	1.63	0.95	1.02
CNN	2.09	0.94	1.1

Πίνακας 4.1: Σύγκριση επίδοσης μοντέλων βαθιάς μάθησης. Μετρικές MSE, MAE και R-squared.

Τα μοντέλα που εκπαιδεύτηκαν, ορίζονται στο παρακάτω κελί κώδικα.

```
if model_type == 1:
    # Single cell LSTM
    model = Sequential()
    model.add(LSTM(50, activation='relu',
    input_shape = (X_train.shape[1], 1)))
    model.add(Dense(1))
if model_type == 2:
    # Stacked LSTM
    model = Sequential()
    model.add(LSTM(50, activation='relu', return_sequences=True,
    input_shape=(X_train.shape[1], 1)))
    model.add(LSTM(50, activation='relu'))
    model.add(Dense(1))
if model_type == 3:
    # Bidirectional LSTM
    model = Sequential()
    model.add(Bidirectional(LSTM(50, activation='relu'),
    input_shape=(X_train.shape[1], 1)))
    model.add(Dense(1))
if model_type == 4:
    # CNN
    model = Sequential()
    model.add(Conv1D(64, kernel_size=2, activation='relu',
    input_shape=(X_train.shape[1], 1)))
    model.add(MaxPooling1D(pool_size=2))
    model.add(Flatten())
    model.add(Dense(50, activation='relu'))
    model.add(Dense(1))
```

Σχήμα 4.7: Σύγκριση μοντέλων βαθιάς μάθησης LSTM, Stacked LSTM, Bidirectional LSTM και CNN.

4.4 Επίδραση των δεδομένων στην ακρίβεια προβλέψεων

Η επιτυχία της εκπαίδευσης ενός αναδρομικού δικτύου για την πρόβλεψη χρονοσειράς εξαρτάται κυρίως από τα δεδομένα. Ακόμα και αν έχουν υποστεί την απαραίτητη προεπεξεργασία, η φύση των δεδομένων μπορεί να επηρεάσει την απόδοση του συστήματος πρόβλεψης. Δεδομένα με εποχικότητα, τείνουν να διευκολύνουν τα LSTM στις προβλέψεις τους, αφού είναι πιθανώς απλούστερη η εύρεση μοτίβων σε αυτές τις χρονοσειρές. Συνεπώς, είναι σημαντική και η διαδικασία της επιλογής κατάλληλων χρονοσειρών για την εκπαίδευση. Σε ένα πρόβλημα many-to-one όπως είναι η πρόβλεψη μιας μοναδικής χρονοσειράς (θερμοκρασία) βασιζόμενη σε πολλαπλές παρελθοντικές χρονοσειρές, όπως η θερμοκρασία, υγρασία, πίεση, πιθανότητα βροχής και άλλες που αναπαριστώνται γραφικά στο σχημα Σχ. 3.4, μπορεί κάποιες χρονοσειρές να μην έχουν χρησιμότητα.

Η επιλογή των χρήσιμων χρονοσειρών, προφανώς δεν μπορεί να γίνει αυτόματα και για αυτό το λόγο έγιναν δύο πειραματισμοί για σύγκριση. Αρχικά, δοκιμάστηκε η παροχή, στο μοντέλο, όλων των χρονοσειρών. Έπειτα εκτελέστηκε εκπαίδευση με χρήση μόνο της θερμοκρασίας, υγρασίας, πίεσης, πιθανότητας και έντασης βροχής, ταχύτητας ανέμου, νεφοκάλυψης και δείκτη υπεριώδους ακτινοβολίας. Αυτές οι χρονοσειρές επιλέχθηκαν με το χέρι διότι κρίθηκαν ως αυτές με την περισσότερη επίδραση στην έκβαση της πρόβλεψης για την μελλοντική θερμοκρασία.

Input Data	**MSE**	R^2
temperature	1.49	0.82
temperature, windGust, windSpeed, windBearing, cloudcover, uvindex, humidity, pressure, precipIntensity, precipProbability, dewPoint, visibility, ozone	1.33	0.83
temperature, windSpeed, cloudcover, uvindex, humidity, pressure, precipIntensity, precipProbability	1.23	0.84

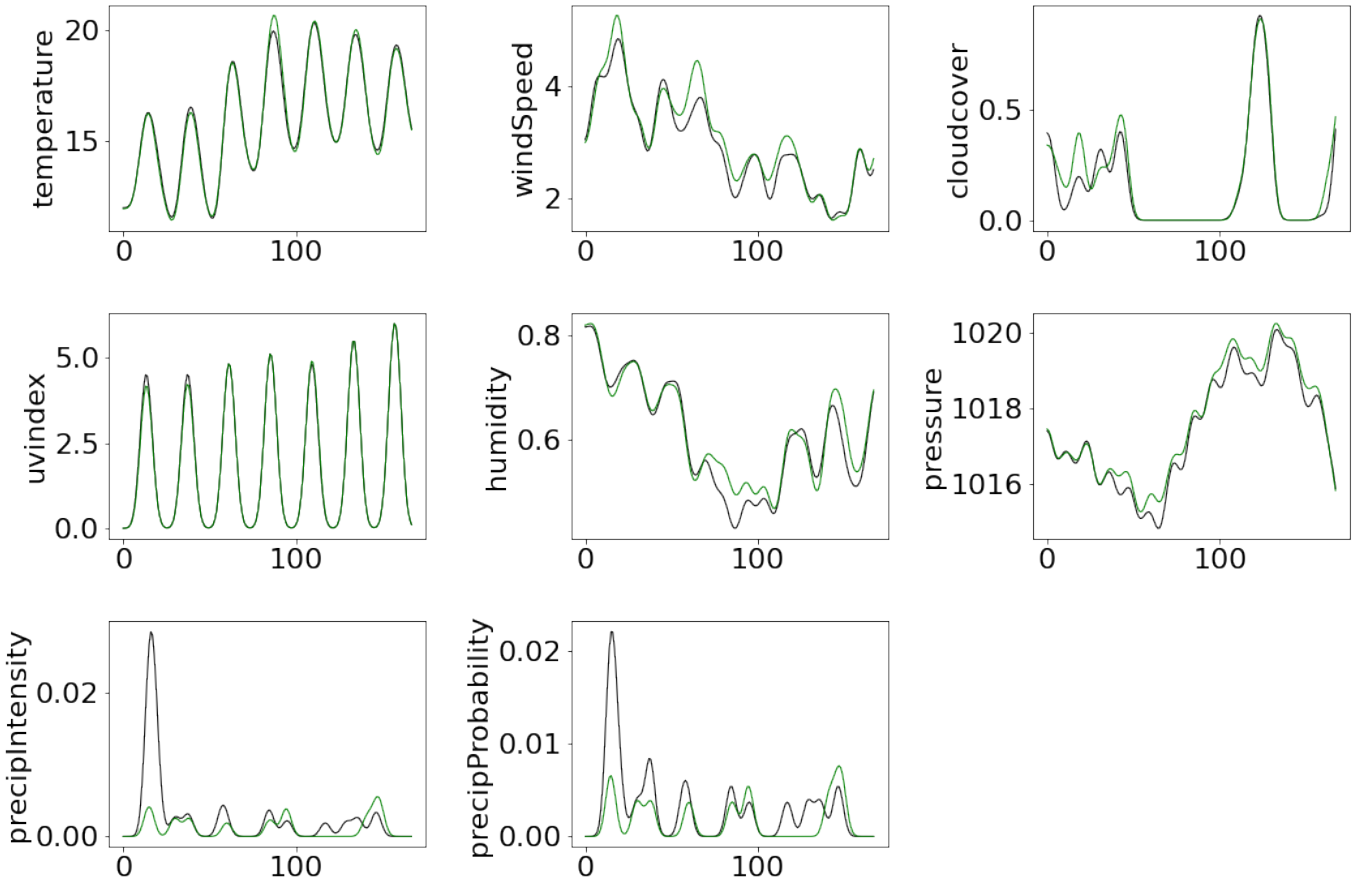
Πίνακας 4.2: Σύγκριση επίδοσης Stacked LSTM σε σχέση με τα δεδομένα εισόδου. Μετρικές MSE και R-squared.

Πράγματι, όπως φαίνεται και στον πίνακα 4.2, το μοντέλο με τα λιγότερα δεδομένα, έκανε συλλογικά καλύτερες προβλέψεις από αυτό που εκπαιδεύτηκε με το σύνολο των χρονοσειρών, αποδεικνύοντας για ακόμη μια φορά πως η ποιότητα των δεδομένων είναι σημαντικότερη από την ποσότητα.

4.5 Αριθμητική βελτίωση μετεωρολογικής πρόγνωσης

Συχνά και ιδιαίτερα σε δεδομένα χρονοσειρών, υπάρχουν προγνώσεις της συμπεριφοράς διαφόρων μεγεθών οι οποίες προκύπτουν από στατιστικά μοντέλα πρόβλεψης. Επομένως, σε αυτή την ενότητα αναλύεται η διαδικασία διόρθωσης και βελτίωσης αυτών των προγνώσεων με τη χρήση τεχνικών βαθιάς μηχανικής μάθησης. Ένα υποσύνολο των κανονικοποιημένων δεδομένων πραγματικών τιμών αλλά και προγνώσεων, φαίνεται στο σχήμα 4.8.

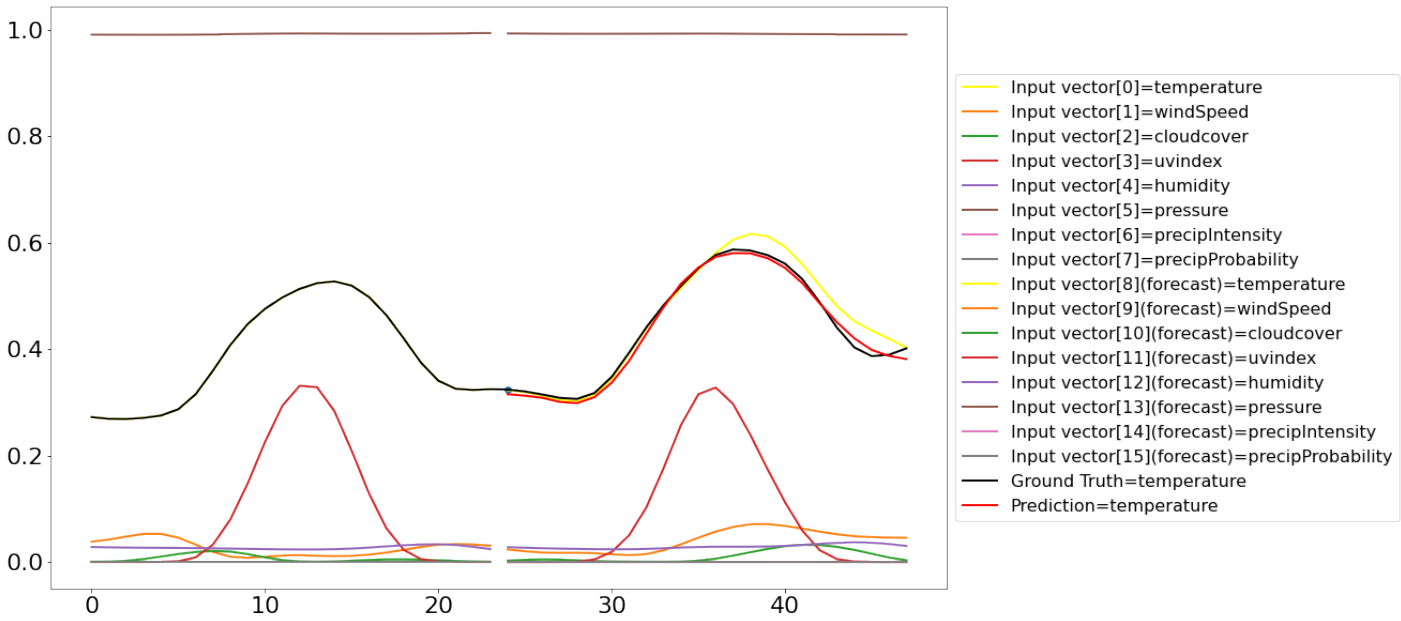
Οι μετεωρολογικές προγνώσεις που ανακτώνται, παράγονται με συγκεκριμένο τρόπο από μαθηματικά μοντέλα. Η προσέγγισή μας, δεν στοχεύει στην δημιουργία νέων μετεωρολογικών προγνώσεων, αλλά στην αριθμητική επεξεργασία ιστορικών χρονοσειρών, μέσω μοντέλων μηχανικής μάθησης, με στόχο την βελτίωση της πρόβλεψης στα μετρούμενα δεδομένα.



Σχήμα 4.8: Γραφική παράσταση των δεδομένων. Με μαύρο παριστάνονται οι πραγματικές μετρήσεις και με πράσινο οι προγνώσεις.

Συγκεκριμένα, οι μελλοντικές προγνώσεις, προστίθενται σαν επιπλέον πληροφορία στις εισόδους του μοντέλου μηχανικής μάθησης.

Στο παρακάτω σχήμα (Σχ. 4.9) παρουσιάζονται οι είσοδοι (παρελθοντικές τιμές και προγνώσεις καιρού), η πραγματική τιμή που μετρήθηκε και η πρόβλεψη του δικτύου. Με κίτρινο χρώμα, φαίνεται η (μετεωρολογική) πρόβλεψη και με μαύρο και κόκκινο, η πραγματική μέτρηση και η πρόβλεψη του δικτύου αντίστοιχα. Είναι προφανές πως το δίκτυο έχει μάθει να “διορθώνει” τις προγνώσεις και να παράγει καλύτερα αποτελέσματα. Παρατηρείται επίσης ότι το πρόβλημα αυτό, όπως φαίνεται και από τα δεδομένα, είναι πολύ ευκολότερο, αφού οι προγνώσεις είναι πολύ καλές και “ακολουθούν” τα σήματα πραγματικών τιμών.



Σχήμα 4.9: Παράδειγμα εισόδων (πραγματικές παρελθοντικές τιμές και προγνώσεις), πρόβλεψης του δικτύου και πραγματικής μελλοντικής μέτρησης.

Συγκεκριμένα στο παράδειγμα του σχήματος το μέσο τετραγωνικό σφάλμα μέτρησης-πρόγνωσης είναι 0.0029 και μέτρησης-πρόβλεψης 0.001, πράγμα που σημαίνει πως υπάρχει βελτίωση (μείωση του σφάλματος) κατά 65.5%.

Γενικά και σε όλο το σύνολο δοκιμής, η απόδοση του μοντέλου φαίνεται στον πίνακα 4.3

Data	**MSE**	R^2
Our Prediction based on Observations only	1.23	0.84
Our Prediction based on Observations and Forecasts	0.0981	0.9873
Dark Sky Forecasts	0.146	0.9812

Πίνακας 4.3: Σύγκριση σφαλμάτων: (1) Προβλέψεων βασισμένων στις παρελθοντικές παρατηρήσεις, (2) Προβλέψεων βασισμένων σε παρελθοντικές παρατηρήσεις & μελλοντικές προγνώσεις, (3) Προγνώσεων. Όλα τα σφάλματα υπολογίζονται πάνω στο ίδιο σύνολο δεδομένων και σε συνάρτηση με τις πραγματικές μελλοντικές παρατηρήσεις.

Στον πίνακα 4.3 παρατηρείται πτώση σφάλματος κατά 92% όταν προστίθενται οι μελλοντικές προγνώσεις στα δεδομένα εισόδου. Πράγμα που σημαίνει ότι το μοντέλο, όπως περιμέναμε, αποδίδει πολύ καλύτερα. Επιπλέον, είναι σημαντικό πως αποδίδει καλύτερα και από τις ίδιες τις προγνώσεις. Συγκεκριμένα υπάρχει μείωση σφάλματος (άρα βελτίωση πρόγνωσης) κατά 32.8%. Αυτή η βελτίωση στην πρόγνωση, μπορεί να είναι πολύ σημαντική για προβλήματα όπου μια καλύτερη πρόγνωση μπορεί να επιφέρει περισσότερα κέρδη ή να οδηγήσει στην αποφυγή καταστροφών.

4.6 Δεδομένα φορτίου ENTSO-E

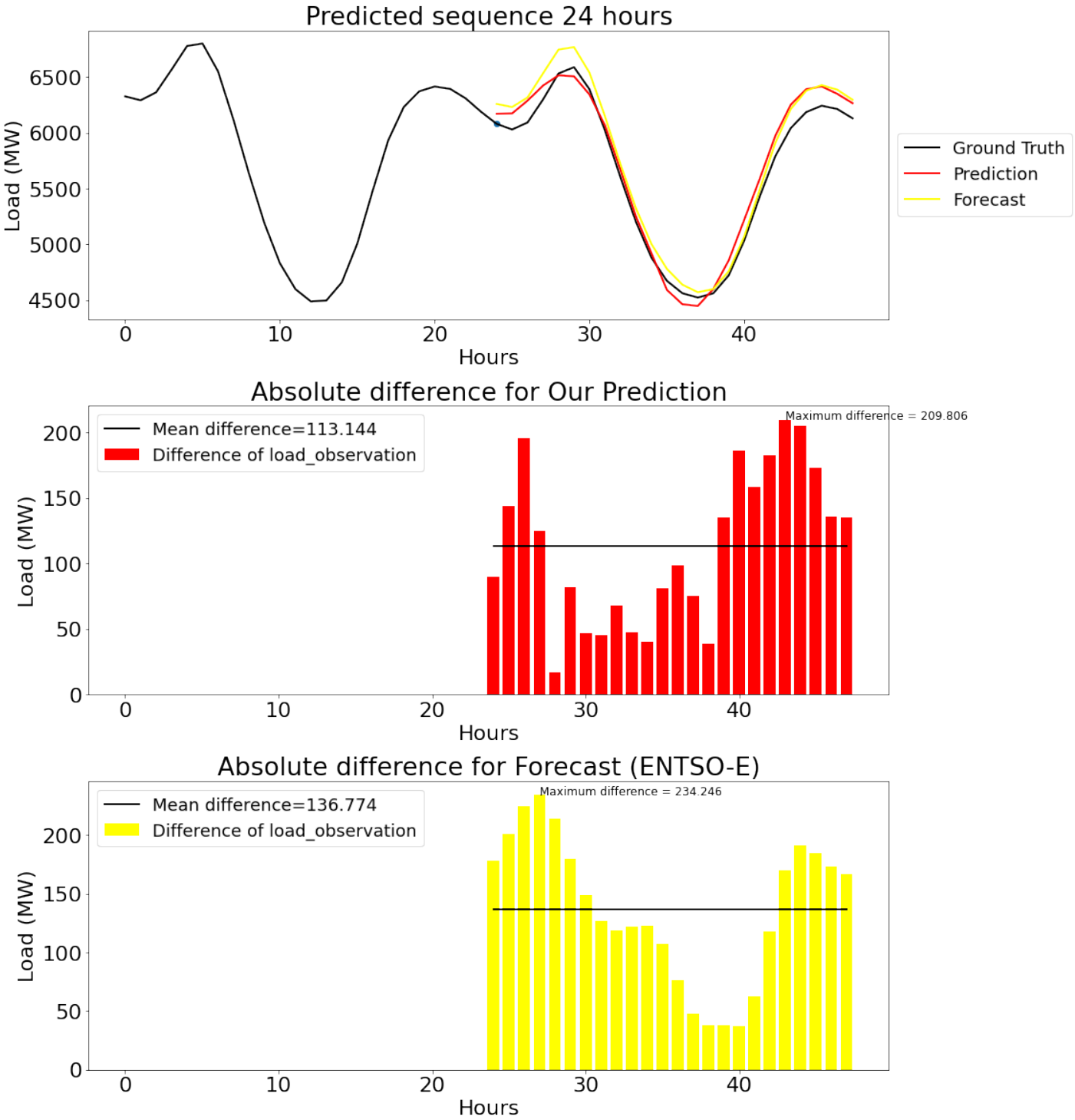
Σκοπός της εργασίας δεν είναι μόνο η μελέτη δεδομένων καιρού, αλλά κυρίως και η σχέση τους με δεδομένα ρεύματος-φορτίου. Τα δεδομένα αυτά δημοσιοποιούνται για ευρωπαϊκές χώρες από το European Network of Transmission System Operators for Electricity (ENTSO-E).

Το φορτίο σε ένα σύστημα ισχύος αναφέρεται ως η ωριαία μέση ενεργή ισχύς που απορροφάται από όλες τις εγκαταστάσεις που είναι συνδεδεμένες στο δίκτυο μεταφοράς ή στο δίκτυο διανομής. Το φορτίο είναι η τιμή, σε μια δεδομένη στιγμή, της ηλεκτρικής ισχύος που παρέχεται ή απορροφάται σε οποιοδήποτε σημείο ενός συστήματος, όπως καθορίζεται από μια στιγμιαία μέτρηση ή από την ολοκλήρωση της ισχύος κατά τη διάρκεια μιας δεδομένης χρονικής περιόδου. Το φορτίο μπορεί να αναφέρεται σε έναν καταναλωτή, μια συσκευή (εργοστασιακή ή οικιακή), μια ομάδα καταναλωτών ή συσκευών ή ένα δίκτυο. Στο συνολικό φορτίο συμπεριλαμβάνονται, η ισχύς που καταναλώνεται από το δίκτυο καθώς επίσης και (+) οι απώλειες του δικτύου αλλά εξαιρούνται (-) η κατανάλωση για αποθήκευση Υδροηλεκτρικής ενέργειας με αντλία και (-) η κατανάλωση βοηθητικών συσκευών (όπως γεννήτριες έκτακτης ανάγκης). Όσον αφορά τη μέθοδο υπολογισμού για τις 24 τιμές φορτίου, οι χώρες χρησιμοποιούν τις μέσες τιμές του φορτίου 10, 15 ή 60 λεπτά πριν από την ώρα της τελικής τιμής [Hirt18].

Προφανώς, ενδιαφέρουσα θα ήταν η μελέτη δεδομένων φορτίου σε μικρότερο πλαίσιο. Για παράδειγμα ο συνδυασμός δεδομένων καιρού και φορτίου για την πρόβλεψη παραγωγής ενέργειας ενός αιολικού πάρκου ή την κατανάλωση ενέργειας μια επιχείρησης ή περιοχής. Όμως η πρόσβαση σε τέτοιου είδους δεδομένα, είναι ιδιαίτερα δύσκολη και για αυτό το λόγο χρησιμοποιήθηκαν τα πανελλαδικά δεδομένα φορτίου της ENTSO-E. Προφανώς όμως, η ίδια ακριβώς μέθοδος, μπορεί να χρησιμοποιηθεί και να εφαρμοστεί σε δεδομένα μικρότερης κλίμακας.

Για τα δεδομένα φορτίου, υπάρχουν και προγνώσεις, οι οποίες προσφέρονται από το ENTSO-E και παράγονται από μοντέλο το οποίο έχει αναπτυχθεί από εξωτερικό προμηθευτή. Επομένως, αναπτύχθηκε μοντέλο βελτίωσης υπάρχουσας πρόγνωσης, όπως αναλύθηκε στην προηγούμενη ενότητα.

Στο παρακάτω σχήμα (Σχ. 4.10) παρουσιάζεται ένα παράδειγμα αποτελέσματος του μοντέλου. Αρχικά παρουσιάζονται οι χρονοσειρές πραγματικής τιμής (μαύρο) πρόβλεψης του μοντέλου (κόκκινο) και πρόγνωσης (κίτρινο). Οι τιμές στον άξονα είναι σε κανονικές μονάδες (MW) αφού έχουν "αποκανονικοποιηθεί". Το μοντέλο προβλέπει ένα εικοσιτετράωρο "μπροστά" με είσοδο τις παρατηρήσεις του προηγούμενου εικοσιτετραώρου και την πρόγνωση για το ζητούμενο εικοσιτετράωρο. Είναι εμφανές πως η πρόβλεψη του μοντέλου, είναι καλύτερη από την πρόγνωση σε αυτή τη περίπτωση. Πράγματι, στα δύο διαγράμματα που ακολουθούν, παρουσιάζεται η απόλυτη διαφορά (σφάλμα) και η πρόβλεψη (κόκκινο) είναι καλύτερη από την πρόγνωση (κίτρινο), τόσο στη μέση τιμή (113.1 - 136.7) όσο και στην μέγιστη απόκλιση που έχουν οι χρονοσειρές (209.8 - 234.2).



Σχήμα 4.10: Παράδειγμα εισόδων (πραγματικές παρελθοντικές τιμές και προγνώσεις), πρόβλεψης του δικτύου και πραγματικής μελλοντικής μέτρησης σε δεδομένα φορτίου. Παρουσιάζεται και η απόλυτη διαφορά των χρονοσειρών παρατήρησης-πρόβλεψης (κόκκινο) και παρατήρησης-πρόγνωσης (κίτρινο).

Συλλογικά και για όλο το σύνολο δοκιμής, η απόδοση του μοντέλου φαίνεται στον πίνακα 4.4

Data	**MSE**	R^2
Our Prediction based on Observations and Forecasts	0.001	0.9676
ENTOS-E Forecasts	0.0044	0.8513

Πίνακας 4.4: Σύγκριση σφαλμάτων: (1) Προβλέψεων βασισμένων σε παρελθοντικές παρατηρήσεις & μελλοντικές προγνώσεις, (2) Προγνώσεων. Όλα τα σφάλματα υπολογίζονται πάνω στο ίδιο σύνολο δεδομένων και σε συνάρτηση με τις πραγματικές μελλοντικές παρατηρήσεις.

Η βελτίωση είναι αξιοσημείωτη, καθώς παρατηρείται πτώση μέσου τετραγωνικού σφάλματος κατά 78%. Η σύγκριση αυτή αφορά τα μέσα τετραγωνικά σφάλματα μεταξύ παρατηρήσεων - προβλέψεων και παρατηρήσεων - προγνώσεων.

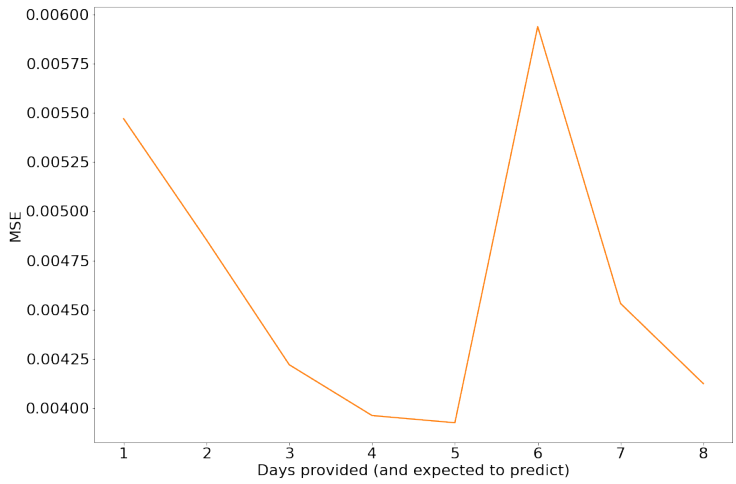
4.7 Επίδραση γνώσης προηγούμενων ημερών στην απόδοση

Εκτός από την παροχή νέων στοιχείων (όπως η πρόγνωση) για την βελτίωση της απόδοσης των μοντέλων, σημασία έχει και η μελέτη του αριθμού προηγούμενων ημερών που δίνεται στο μοντέλο σαν γνώση, για την πρόβλεψη της μελλοντικής συμπεριφοράς. Συγκεκριμένα, σε αυτή την ενότητα, εκπαιδεύονται μοντέλα τα οποία δέχονται x ημέρες στην είσοδο και προβλέπουν x ημέρες αντίστοιχα στην έξοδο, έχοντας προφανώς και τις x προγνώσεις της ENTSO-E για τις ζητούμενες ημέρες, σαν επιπρόσθετες πληροφορίες. Αυτό το πείραμα εκτελείται αρχικά για μία ημέρα εισόδου, όπου ζητείται και μια ημέρα πρόβλεψης στην έξοδο. Έπειτα το παράθυρο αυξάνεται κατά ένα εικοσιτετράωρο, ζητόντας δηλαδή 2 ημέρες πρόβλεψης, και η διαδικασία συνεχίζεται μέχρι τις 8 ημέρες. Το πείραμα σταματάει στις 8 ημέρες εισόδου, αφού στόχος είναι ο έλεγχος συμπεριφοράς της αναμενόμενης εβδομαδιαίας περιοδικότητας.

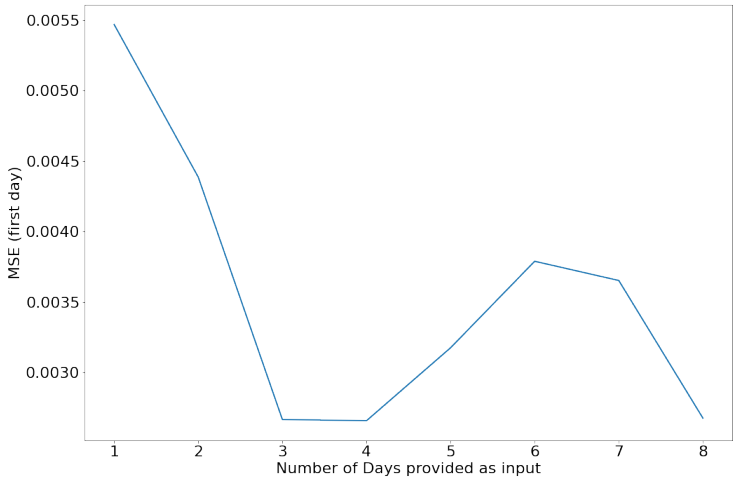
Τα αποτελέσματα φαίνονται στο παρακάτω σχήμα (Σχ. 4.11), όπου παρουσιάζεται η μεταβολή του μέσου τετραγωνικού σφάλματος για κάθε μοντέλο. Σύγκριση των μοντέλων γίνεται στην περίπτωση αυτή στο ίδιο σύνολο δοκιμής. Αρχικά παρουσιάζεται η μεταβολή όπου το διάνυσμα εξόδου αντιστοιχεί σε όλες τις ημέρες δηλαδή τόσες όσες και στην είσοδο (πορτοκαλί). Αυτό το πείραμα αντιστοιχεί σε μεσοπρόθεσμη πρόβλεψη. Όμως ενδιαφέρον έχει και η μέτρηση της επίδοσης στην βραχυπρόθεσμη περίπτωση της μίας ημέρας. Πρακτικά, από το διάνυσμα πρόβλεψης-εξόδου κρατείται μόνο το μέρος τη πρώτης ημέρας για όλα τα μοντέλα και συγκρίνεται με την πραγματική μέτρηση αυτής της ημέρας. Τέλος παρουσιάζονται σε κοινό διάγραμμα οι μεταβολές, όπου και φαίνεται πως η πρόβλεψη της πρώτης ημέρας και μόνο (μπλε χρώμα) είναι πάντοτε καλύτερη, προφανώς, αφου το διάνυσμα που συγκρίνεται είναι πολύ μικρότερο και άρα περιέχει λιγότερα λάθη.

Συνεπώς, το σφάλμα της πρώτης ημέρας (μπλε), αν και "ακολουθεί" την κατανομή για όλες τις ημέρες (πορτοκαλί), έχει μικρότερο σφάλμα. Επίσης, μέχρι την τέταρτη ημέρα, παρατηρούμε πτώση σφάλματος και στις δύο περιπτώσεις, πράγμα το οποίο μας κάνει να εικάζουμε πως περισσότερη πληροφορία (παρελθοντική αλλά και πρόγνωσης) θα κάνει τα μοντέλα να αποδίδουν καλύτερα. Όμως στην έκτη και όγδοη ημέρα παρατηρούμε αύξηση σφάλματος, οπότε αυτό δεν ισχύει απαραίτητα. Ενδιαφέρον έχει η μείωση του σφάλματος (στο ελάχιστο) μετά την έβδομη ημέρα, παρότι υπάρχει προηγούμενη αύξηση. Αυτό μπορεί να δικαιολογηθεί και να ερμηνευθεί λόγο πιθανής εποχικότητας των δεδομένων φορτίου ανα εβδομάδα. Εν γένει η ύπαρξη τάξης και εποχικότητας στα δεδομένα κάνει τα LSTM να λειτουργούν καλύτερα, αφού είναι πιθανότερο να εντοπίσουν και να μάθουν μοτίβα.

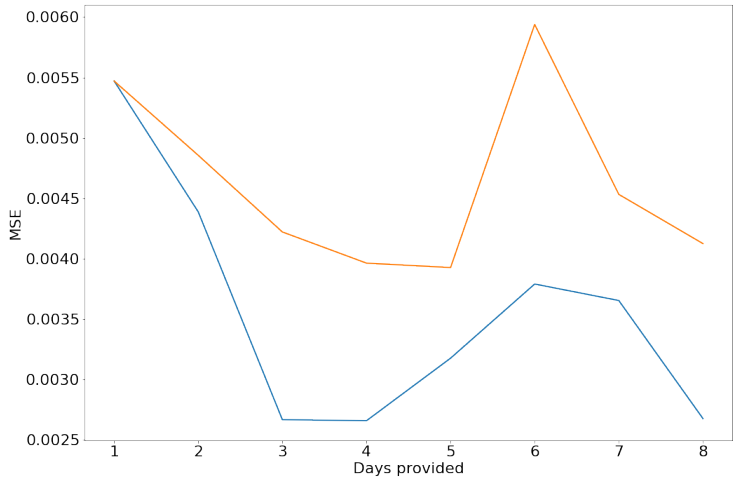
Συνοψίζοντας, μπορούμε να συμπεράνουμε (από το σχήμα) ότι το πλήθος των προηγούμενων ημερών μπορούν να επηρεάσουν σημαντικά την απόδοση του μοντέλου πρόβλεψης. Σημασία όμως πάντοτε έχει και η ερμηνεία των αποτελεσμάτων αυτών, ιδιαίτερα όταν πρόκειται για ανάλυση χρονοσειρών που αφορούν δεδομένα καιρού ή φορτίου. Βέβαια, συνήθως σε προβλήματα μηχανικής μάθησης, περισσότερα δεδομένα οδηγούν σε καλύτερες προβλέψεις στην γενική περίπτωση. Συνεπώς, το πλήθος των παρελθοντικών ημερών που θα χρησιμοποιηθούν μπορεί να εξαρτάται και από την εκάστοτε εφαρμογή, τον αριθμό των αναμενόμενων "επανεκπαιδεύσεων" των μοντέλων καθώς και τους διαθέσιμους υπολογιστικούς πόρους.



(a)



(b)



(c)

Σχήμα 4.11: Μεταβολή του μέσου τετραγωνικού σφάλματος: (**a**) MSE του διανύσματος εξόδου, μήκους ίσο με αυτό του διανύσματος εισόδου. (**b**) MSE μέρους του διανύσματος εξόδου, μόνο της πρώτης ημέρας. (**c**) Τα δύο σφάλματα σε κοινό διάγραμμα.

4.8 Συνδυασμός δεδομένων φορτίου-καιρού

Σε αυτή την ενότητα έγινε απόπειρα πρόσθεσης δεδομένων καιρού σαν επιπλέον πληροφορία για την πρόβλεψη δεδομένων φορτίου. Παρόμοια ανάλυση έχει γίνει στην ενότητα 4.4 για τα δεδομένα καιρού.

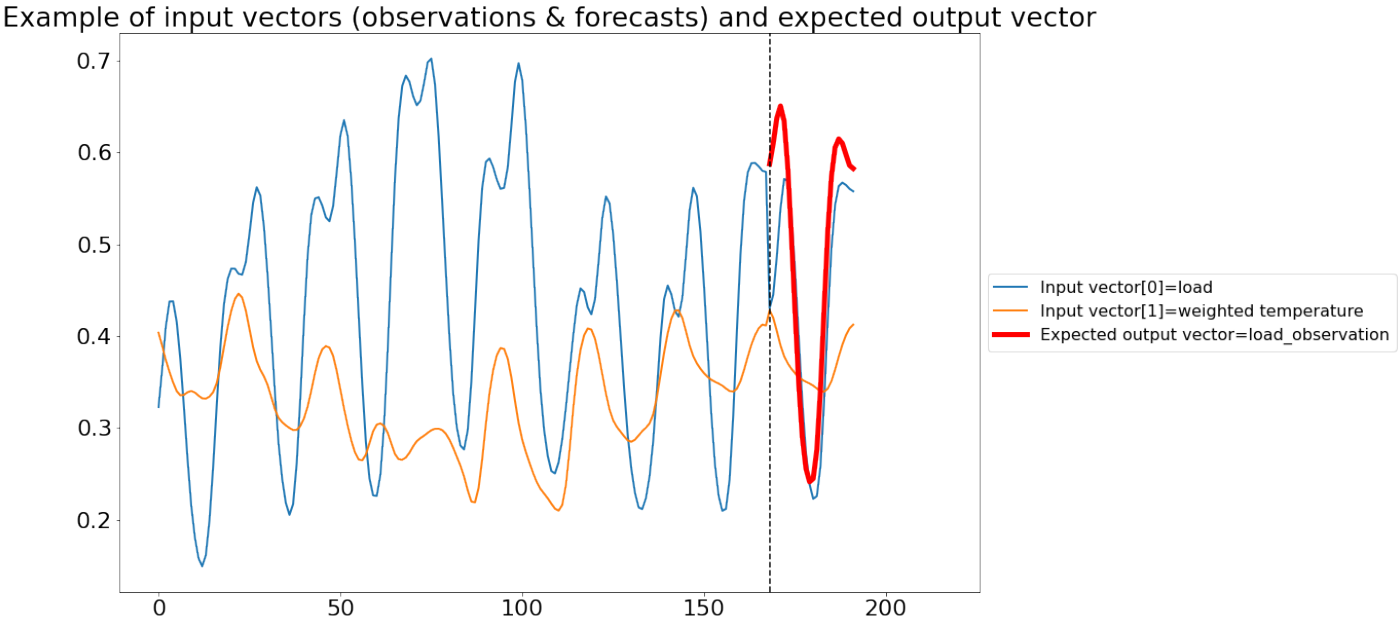
Συγκεκριμένα, είχαμε στην διάθεσή μας δεδομένα θερμοκρασίας από το 2015 ως το 2019 του Εθνικού Αστεροσκοπείου Αθηνών για κάποιες από τις μεγαλύτερες πόλεις της Ελλάδας. Αναλυτικά οι περιοχές: Αιτωλικό, Αλεξανδρούπολη, Άργος, Αθήνα, Χαλκίδα, Ελευθερούπολη, Φλώρινα, Ιωάννινα, Ισθμός (Κορίνθου), Καλαμάτα, Λαμία, Λάρισα, Πάτρα, Σπάρτη, Θεσσαλονίκη, Τρίκαλα, Τρίπολη, Βόλος.

Εφόσων όμως το φορτίο αντιπροσωπεύει τιμές σε πανελλαδικό επίπεδο, αποφασίστηκε η κανονικοποίηση των τιμών θερμοκρασίας των παραπάνω περιοχών σε μια μόνο τιμή. Το νέο αυτό σταθμισμένο μέτρο μπορεί να ερμηνευθεί χονδρικά ως μια "γενική θερμοκρασία" της Ελλάδας. Συγκεκριμένα το νέο αυτό μέτρο προκύπτει μαθηματικά ως εξής

$$T_{weighted} = \sum_{i=1}^{N} temp_i \frac{population_i}{TotalPopulation}$$

Πρακτικά, κάθε τιμή θερμοκρασίας, πολλαπλασιάζεται με ένα βάρος, σταθμίζεται δηλαδή ανάλογα με τον πληθυσμό της περιοχής σε σχέση με τον συνολικό πληθυσμό της Ελλάδας.

Ένα παράδειγμα εισόδου του μοντέλου παρουσιάζεται στο σχήμα Σχ. 4.12. Παρατηρείται ότι υπάρχει κάποια σχέση μεταξύ του σήματος φορτίου (μπλε) και της σταθμισμένης θερμοκρασίας (πορτοκαλί) αφού οι δύο χρονοσειρές δείχνουν να έχουν αύξηση και αντίστοιχα μείωση σε παρόμοια χρονικά διαστήματα.



Σχήμα 4.12: Παράδειγμα εισόδων (παρελθοντικές τιμές) του φορτίου και της σταθμισμένης θερμοκρασίας. Τα δύο σήματα αυτά περιέχουν και τιμές πρόγνωσης μετά τη διακεκομμένη γραμμή. Με κόκκινο παρουσιάζεται η πραγματική μέτρηση (αναμενόμενη τιμή).

Παρότι προσφέρουμε εππλέον πληροφορία στο μοντέλο, όπως αποδείχθηκε στο πείραμα της ενότητας 4.4, η ανάλογα καλύτερη επίδοση δεν είναι εγγυημένη. Συγκεκριμένα, για το σύνολο δοκιμής, η απόδοση του μοντέλου με και χωρίς την επιπρόσθετη πληροφορία την σταθμισμένης θερμοκρασίας, φαίνεται στον πίνακα 4.5.

Παρατηρούμε ότι η επιπρόσθετη πληροφορία βοηθάει ελάχιστα (της τάξης του 15%). Προφανώς δεν περιμέναμε μεγάλη μεταβολή της απόδοσης, αφού η πρόγνωση που προσφέρει η ENTSO-E είναι πολυτιμότερη πληροφορία από αυτήν της σταθμισμένης θερμοκρασίας.

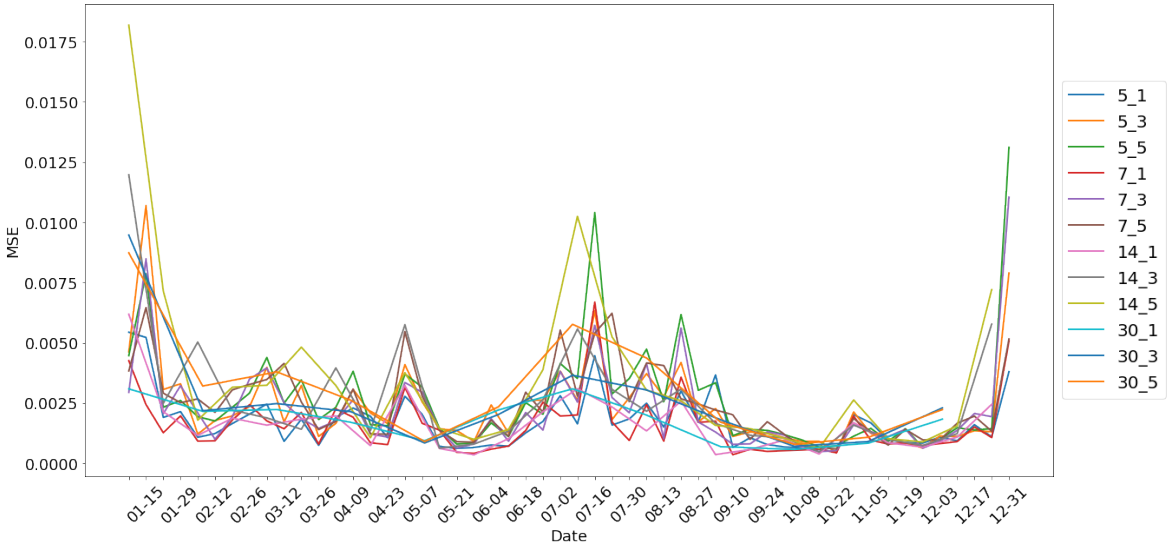
Data	**MSE**	R^2
Our Prediction based on Load	0.0025	0.9201
Our Prediction based on Load & Temperature	0.0021	0.9272

Πίνακας 4.5: Σύγκριση σφαλμάτων (1) χωρίς την σταθμησμένη θερμοκρασία, (2) με την επιπρόσθετη σταθμισμένη θερμοκρασία.

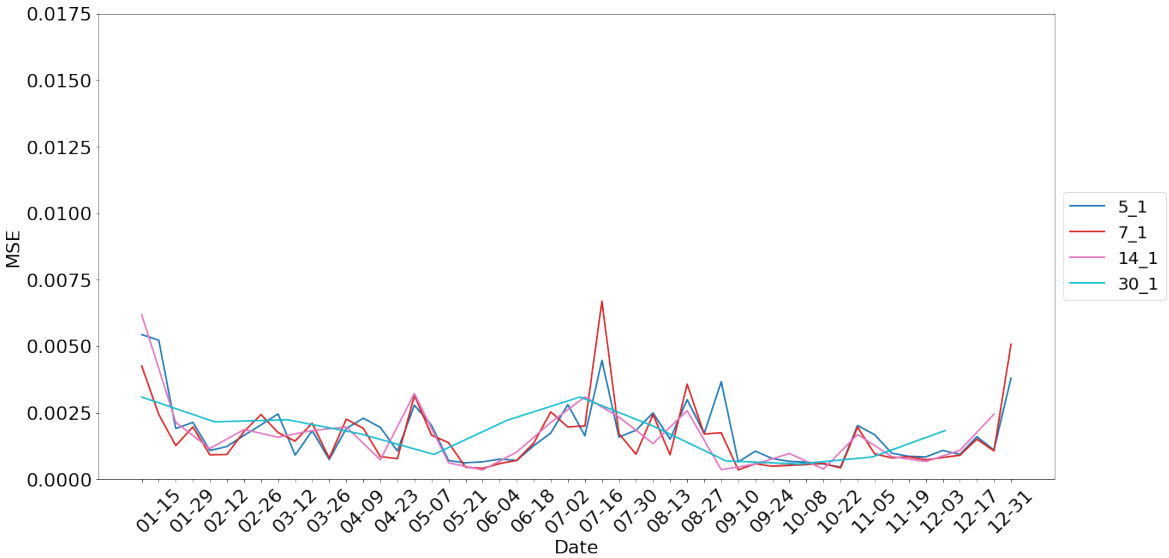
Εάν τα δεδομένα ήταν μικρότερης κλίμακας, για παράδειγμα ανάλυση φορτίου ενός αιολικού πάρκου, τότε ενδεχομένως τα δεδομένα καιρού υψηλότερης ακρίβειας της εκάστοτε περιοχής να παρείχαν πολύτιμη πληροφορία.

4.9 Επίδραση παραθύρου προηγούμενης γνώσης - μελλοντικής πρόβλεψης

Στα προηγούμενα παραδείγματα αυτού του κεφαλαίου, το ζητούμενο ήταν η πρόβλεψη της συμπεριφοράς του φορτίου μιας ημέρας, δεδομένης της παρελθοντικής συμπεριφοράς του για μια ημέρα ή μια εβδομάδα. Σε αυτή την ενότητα, παρουσιάζονται τα αποτελέσματα πειραματισμών με το "παράθυρο" των παρελθοντικών γεγονότων σε τιμές 5, 7, 14, 30 ημερών καθώς και του παραθύρου ζητούμενων προβλέψεων σε τιμές 1, 3, 5 ημερών. Σε όλες τις περιπτώσεις το μοντέλο δέχεται σαν είσοδο, τόσο τις πραγματικές παρελθοντικές μετρήσεις όσο και τις προγνώσεις της ENTSO-E για τις ζητούμενες ημέρες. Όλες οι διατάξεις εκπαιδεύτηκαν στο διάστημα τιμών φορτίου 2015-2018 και δοκιμάζονται στο 2019. Στα παρακάτω σχήματα (Σχ. 4.13, 4.14), παρουσιάζεται η μεταβολή του μέσου τετραγωνικού σφάλματος για το σύνολο του χρόνου 2019 (σύνολο δοκιμής). Αν και περίπλοκο, το διάγραμμα παρουσιάζει μια παρόμοια συμπεριφορά του σφάλματος για όλα τα πειράματα.



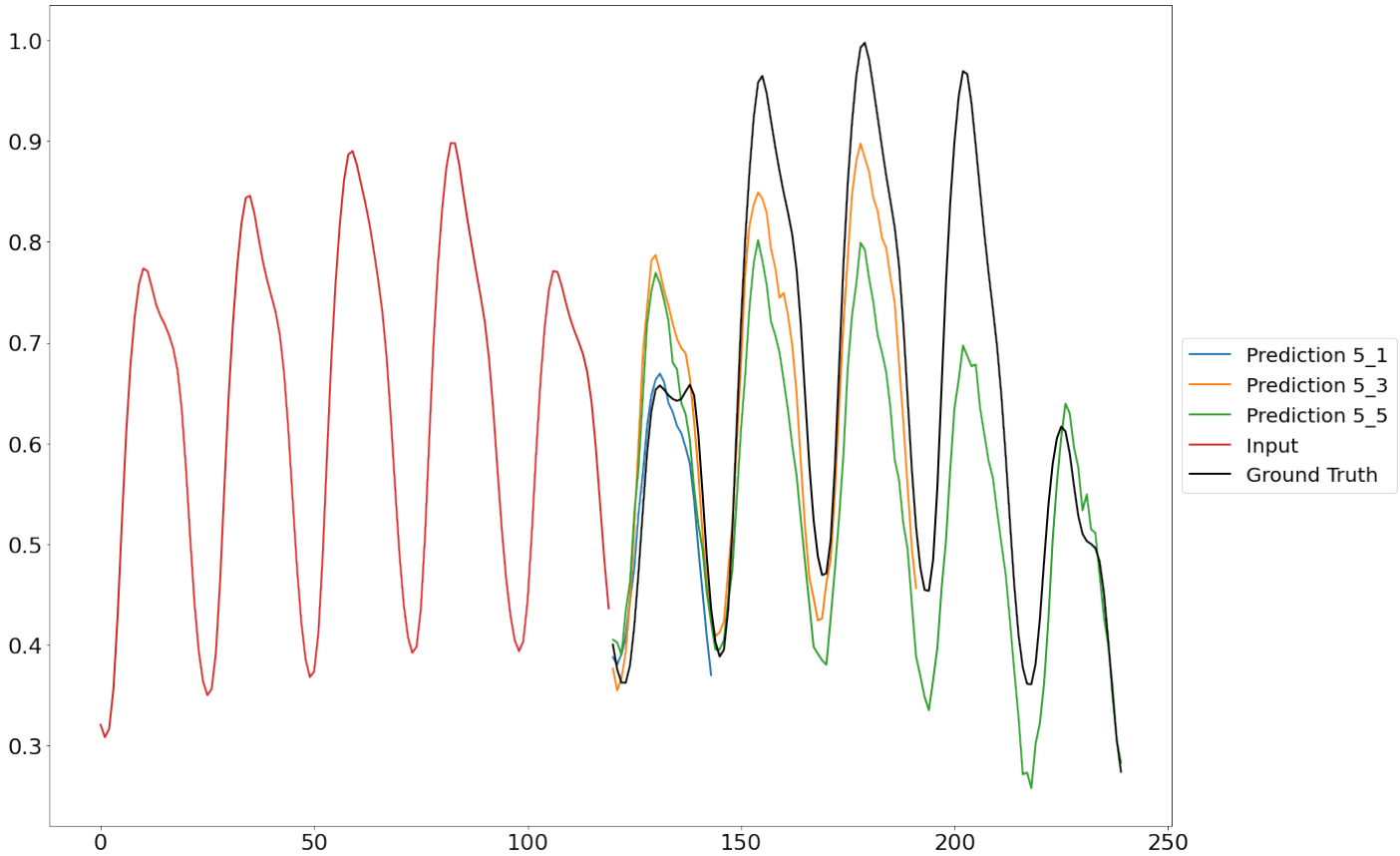
Σχήμα 4.13: Μεταβολή του μέσου τετραγωνικού σφάλματος για όλα τα πιθανά "σενάρια" παραθύρων παρελθοντικών και μελλοντικών τιμών.



Σχήμα 4.14: Μεταβολή του μέσου τετραγωνικού σφάλματος των καλύτερων μοντέλων.

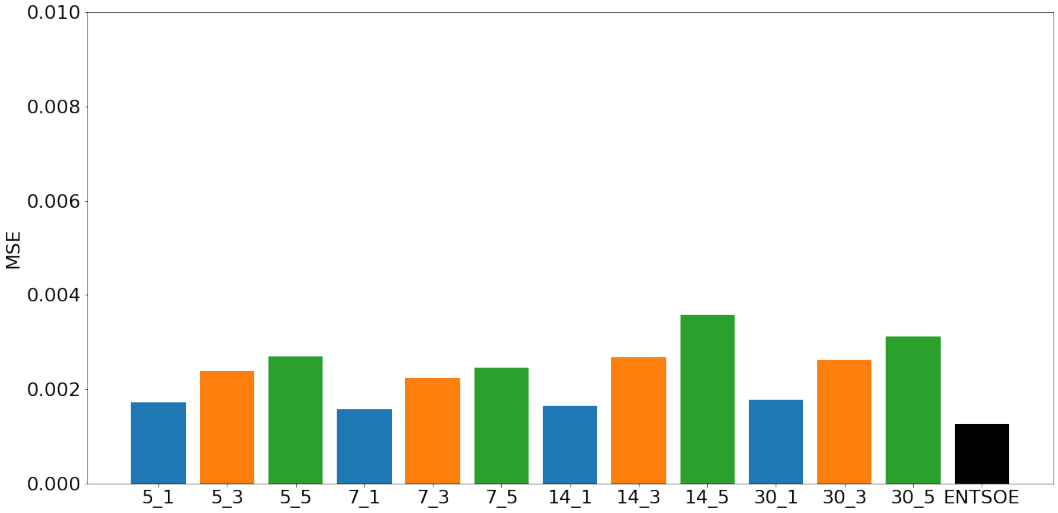
Αυτό το συμπέρασμα, το ότι δηλαδή παρόλο που κάθε πείραμα δέχεται διαφορετικού μήκους χρονοσειρές στην είσοδο, το σφάλμα ακολουθεί χονδρικά την ίδια κατανομή, αποδεικνύει το πόσο βασισμένο στα δεδομένα (data driven) είναι ένα σύστημα πρόγνωσης μηχανικής μάθησης.

Στο σχήμα που ακολουθεί (Σχ. 4.15) παρουσιάζονται οι χρονοσειρές εξόδου των μοντέλων που έχουν παρελθοντικό παράθυρο εισόδων 5 ημερών και προβλέπουν 1, 3 και 5 ημέρες στο μέλλον αντίστοιχα. Για την πρώτη ημέρα, παρατηρείται πως το μοντέλο 5_1 έχει καλύτερη απόδοση, όπως ήταν αναμενόμενο αφού έχει εκπαιδευτεί συγκεκριμένα για αυτό. Για τις πρώτες 3 ημέρες, καλύτερη απόδοση έχει το μοντέλο 5_3, ενώ το 5_5 έχει καλύτερη απόδοση στο σύνολο των 5 ημερών.

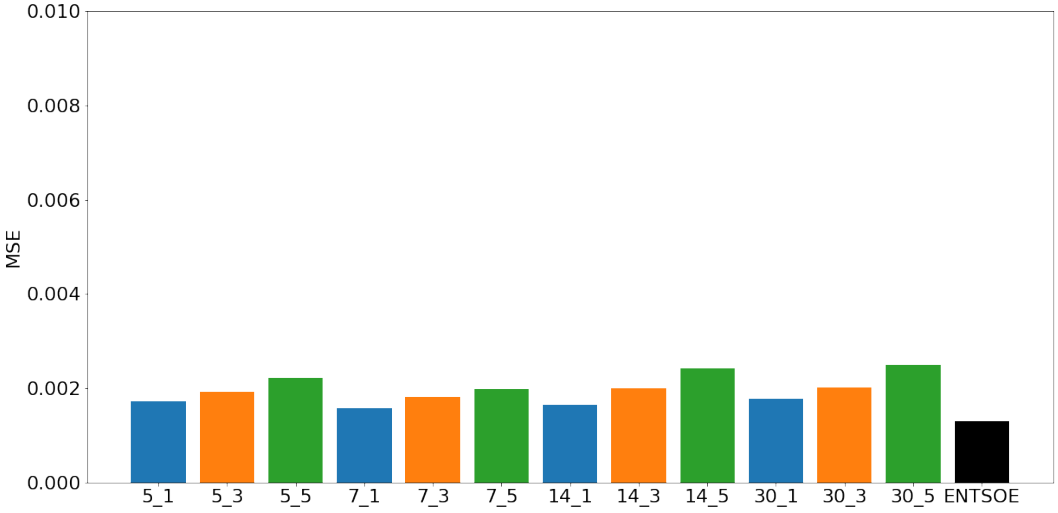


Σχήμα 4.15: Χρονοσειρές πρόβλεψης για τα μοντέλα 5 παρελθοντικών ημερών.

Αυτή η συμπεριφορά είναι λογική και αναμένεται σε όλο το σύνολο εκπαίδευσης και για όλα τα μοντέλα. Πράγματι, στο σχήμα (Σχ. 4.16) φαίνεται πως το σφάλμα αυξάνεται καθώς αυξάνονται οι ημέρες ζητούμενης πρόβλεψης, πράγμα λογικό αφού το πρόβλημα πρόβλεψης 5 ημερών είναι πολύ δυσκολότερο από αυτό πρόβλεψης μίας ημέρας. Επίσης φαίνεται πως το σφάλμα αυξάνεται και για την πρώτη ημέρα, επιβεβαιώνοντας την συμπεριφορά του σχήματος 4.15.



(a)



(b)

Σχήμα 4.16: Bar plot σφαλμάτων όλων των μοντέλων (**a**) για τις αντίστοιχες ημέρες ζητούμενης πρόβλεψης (**b**) μόνο για την πρώτη ημέρα.

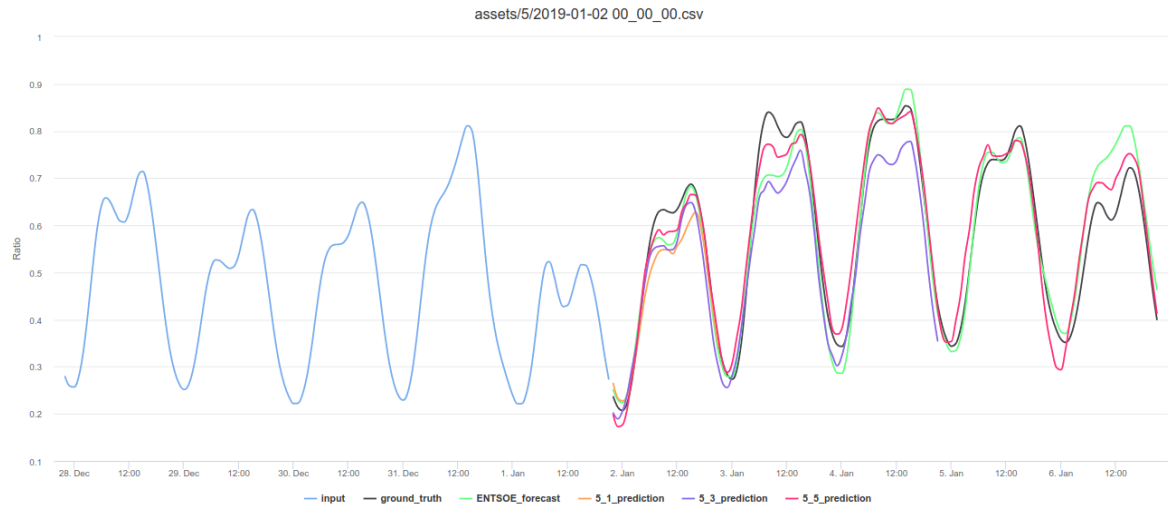
Συλλογικά καλύτερο, αποδεικνύεται οτι είναι το μοντέλο 7_1, δηλαδή αυτό το οποίο προβλέπει μια ημέρα δεδομένης μιας εβδομάδας. Αυτό ίσως να οφείλεται στην εποχικότητα που μπορεί να έχουν τα δεδομένα φορτίου ανα εβδομάδα. Σύγκριση γίνεται και σε σχέση με την πρόγνωση του ENTSO-E (μαύρο). Παρατηρούμε παρόμοια απόδοση με το καλύτερο μοντέλο 7_1.

4.10 Οπτικοποίηση Χρονοσειρών

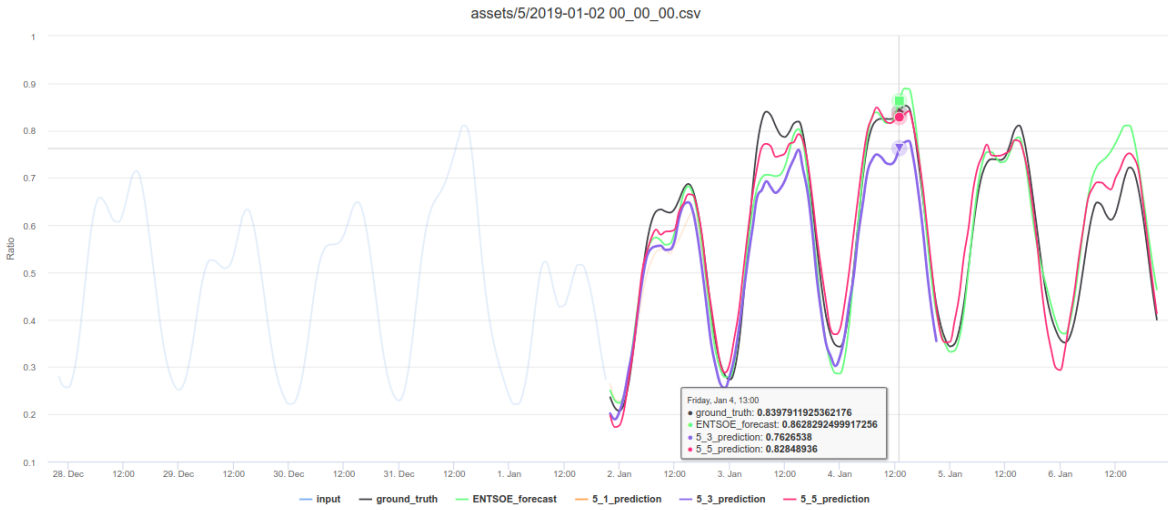
Το σύστημα πρόβλεψης που αναπτύχθηκε έχει αποδεδειγμένα καλά αποτελέσματα και θα μπορούσε να εφαρμοστεί, με τα κατάλληλα δεδομένα, σε μικρότερης κλίμακας εφαρμογές, χωρίς να απαιτούνται αλλαγές στον κώδικα. Για αυτό το λόγο, αναπτύχθηκε μια "production-ready" εφαρμογή οπτικοποίησης και παρακολούθησης των χρονοσειρών. Η ανάπτυξη έγινε σε συνεννόηση με απόφοιτο διπλωματικό φοιτητή του κ. Βεσκούκη, Ιωάννη Τζαννέτο. Χρησιμοποιήθηκε το web framework Angular CLI καθώς και η βιβλιοθήκη Highcharts JS η οποία ενδείκνυται για την διαδραστική οπτικοποίηση χρονοσειρών και όχι μόνο.

Πρακτικά, στήνεται ένας server ο οποίος, κατόπιν αιτήματος, αποκτά με GET τα απαραίτητα δεδομένα χρονοσειρών και τα παρουσιάζει σε ένα φιλικό προς τον χρήστη (user friendly) περιβάλλον. Ο χρήστης μπορεί να προσπελάσει τις τιμές χρονοσειρών που εμφανίζονται ανα ημερομηνία, καθώς επίσης και να διαλέξει τις προβλέψεις των μοντέλων της ενότητας 4.9 που θα εμφανίζονται.

Στιγμιότυπα της εφαρμογής φαίνονται στο παρακάτω σχήμα (Σχ. 4.17). Ο χρήστης περνώντας πάνω από τα σήματα μπορεί να δει αναλυτικά τις εκάστοτε τιμές.



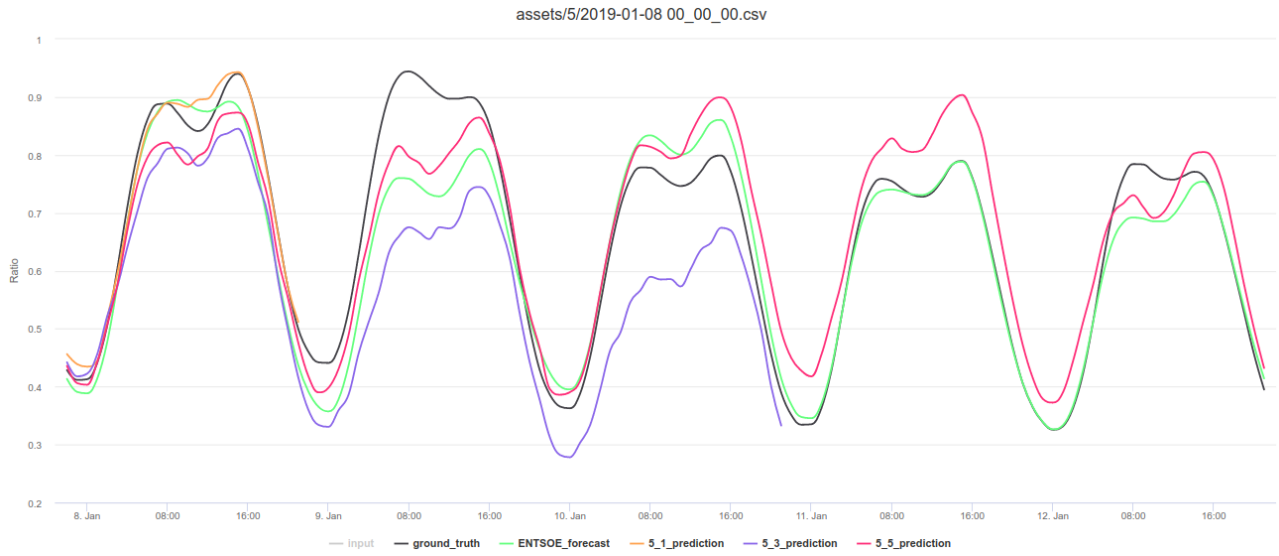
(a)



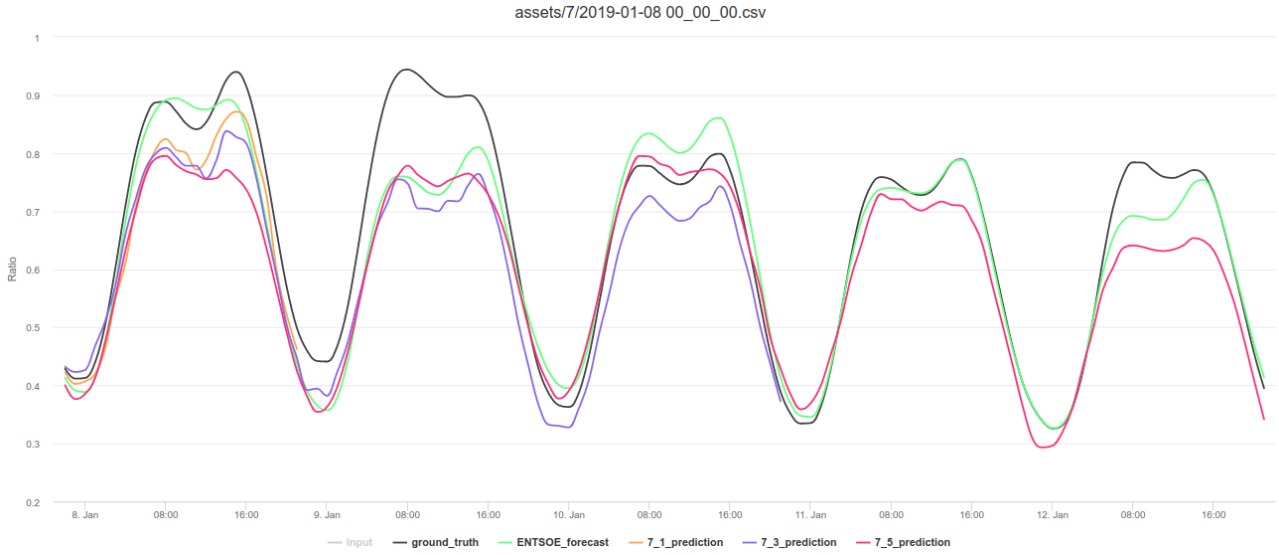
(b)

Σχήμα 4.17: Στιγμιότυπα του εργαλείου οπτικοποίησης.

Στα παρακάτω σχήματα (Σχ. 4.18, 4.19) παρουσιάζονται διαγράμματα όλων των μοντέλων για την ίδια ημέρα. Η ταξινόμηση γίνεται με βάση το πλήθος παρελθοντικών μετρήσεων που λαμβάνει το κάθε μοντέλο στην είσοδο, δηλαδή 5, 7, 14 και 30 ημέρες.

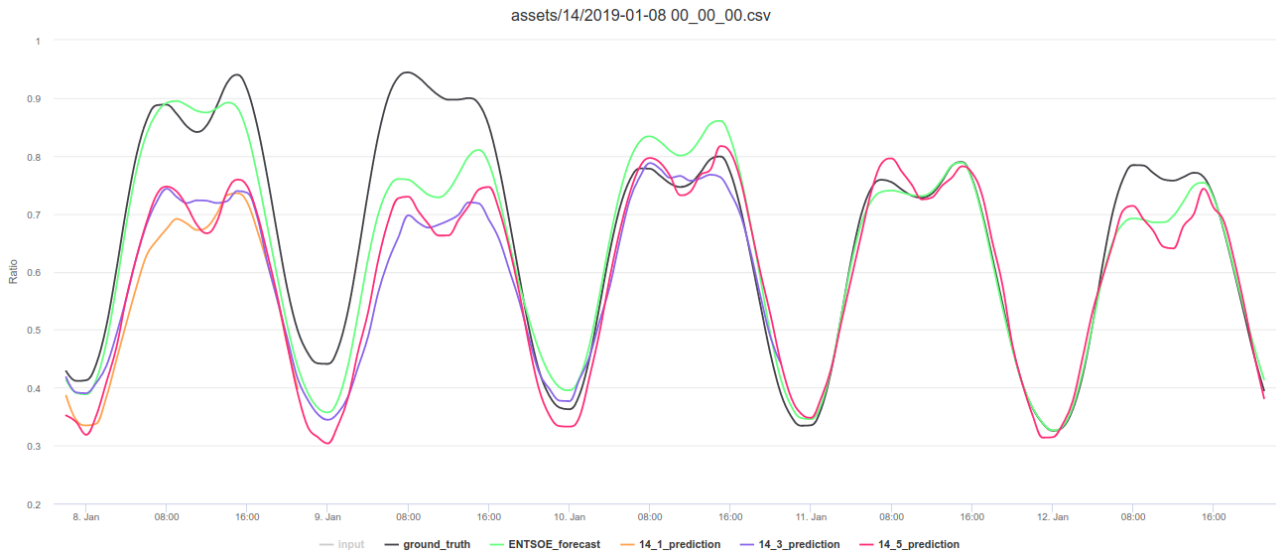


(a)

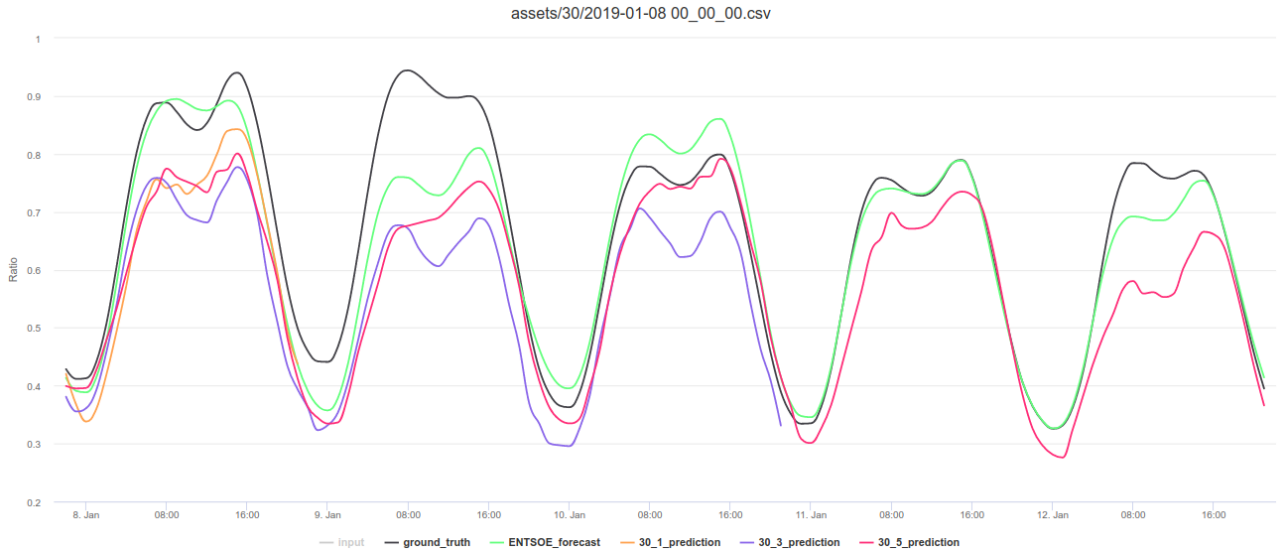


(b)

Σχήμα 4.18: Οπτικοποίηση προβλέψεων για τα μοντέλα 5 και 7 ημερών.



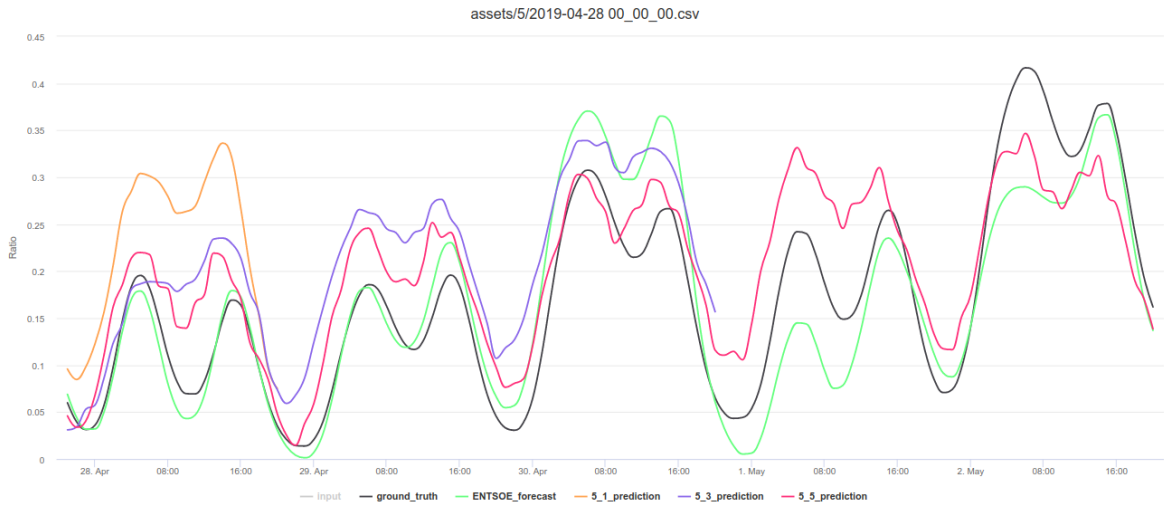
(a)



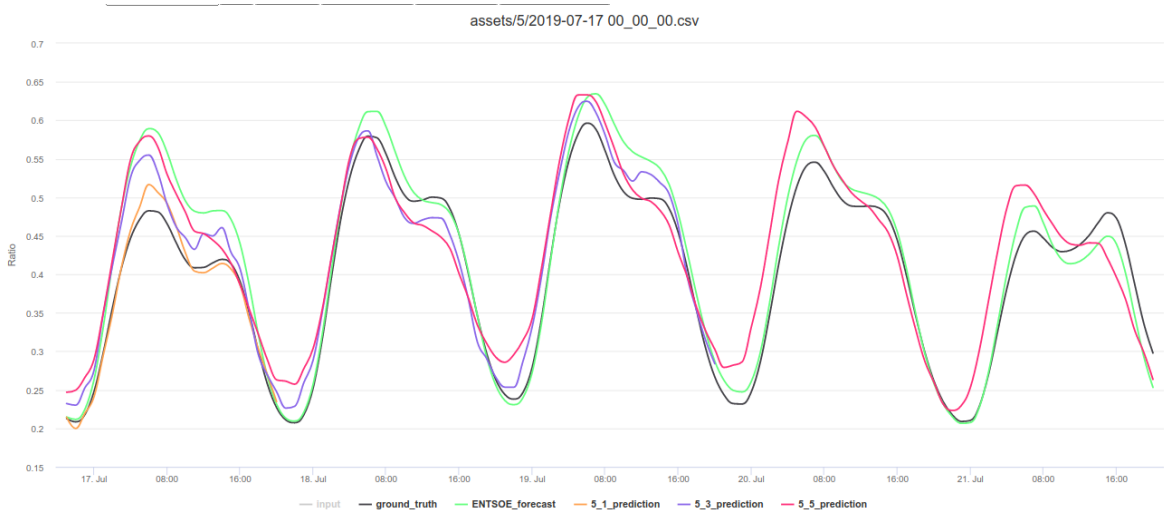
(b)

Σχήμα 4.19: Οπτικοποίηση προβλέψεων για τα μοντέλα 14 και 30 ημερών.

Στο σχήμα Σχ. 4.20 παρουσιάζονται μια όχι και τόσο καλή, αλλά και μια πολύ καλή πρόβλεψη. Συγκεκριμένα, στο διάγραμμα **(a)** φαίνεται η πρόβλεψη για την πρώτη ημέρα (πορτοκαλί) να είναι σε υψηλότερα επίπεδα από την πραγματική τιμή (μαύρο) αλλά και από την πρόγνωση ENTSO-E (πράσινο). Από την άλλη, στο διάγραμμα **(b)** η πρόβλεψή μας (πορτοκαλί) ακολουθεί πολύ καλύτερα την πραγματική τιμή (μαύρο) σε σχέση με την πρόγνωση ENTSO-E (πράσινο).



(a)



(b)

Σχήμα 4.20: Οπτικοποίηση περιπτώσεων όπου **(a)** η πρόβλεψή μας δεν είναι καλή και **(b)** η πρόβλεψή μας είναι πολύ καλή, καλύτερη και από την πρόγνωση ENTSO-E.

Κεφάλαιο 5

Συμπεράσματα-Μελλοντικές εφαρμογές

5.1 Σύνοψη εργασίας και συμπερασμάτων

Στην παρούσα εργασία μελετήθηκαν συστήματα πρόβλεψης δεδομένων καιρού και φορτίου. Αρχικά αναπτύχθηκαν εργαλεία αυτόματης ανάκτησης δεδομένων καιρού συγκεκριμένων περιοχών, τόσο των παρατηρήσεων όσο και των προβλέψεων. Αυτά αποθηκεύτηκαν σε βάση δεδομένων για περαιτέρω μελέτη σε επόμενα βήματα. Έπειτα δόθηκε μεγάλη έμφαση στην προεπεξεργασία των δεδομένων. Έγινε αποθορυβοποίηση και κανονικοποίηση σε όλες τις χρονοσειρές ξεχωριστά, αφού αυτή η διαδικασία είναι απαραίτητη για την ορθότερη λειτουργία των αλγορίθμων πρόβλεψης.

Έπειτα υλοποιήθηκαν και αναλύθηκαν διάφορα μοντέλα μηχανικής μάθηση συμπεριλαμβανομένων των LSTM, CNN, Stacked LSTM και Bidirectional LSTM. Η μελέτη τους έγινε, τόσο θεωρητικά όσο και πρακτικά, αξιολογόντας την απόδοσή τους στο ίδιο σύνολο δεδομένων. Η σύγκριση έγινε με βάση τις μετρικές MSE, MAE και R^2 και καλύτερη απόδοση αποδείχθηκε να έχει το μοντέλο Stacked LSTM.

Εκτελέστηκαν παραδείγματα χρησιμοποιώντας το καλύτερο Stacked LSTM πάνω σε δεδομένα καιρού, όπου μελετήθηκε η χρησιμότητα των σημάτων (θερμοκρασία, υγρασία κ.α) και η συμβολή τους στην βελτίωση της απόδοσης. Το συμπέρασμα είναι πως η φύση των δεδομένων μπορεί να επηρεάσει σε μεγάλο βαθμό την απόδοση του συστήματος πρόβλεψης.

Επίσης δόθηκε μεγάλη έμφαση στην βελτίωση των ήδη υπάρχοντων προγνώσεων, το οποίο επιτεύχθηκε εκπαιδεύοντας τα μοντέλα με τις προγνώσεις της ζητούμενης ημέρας σαν μια επιπλέον είσοδο. Στόχος δεν είναι η παραγωγή νέων μετεωρολογικών προγνώσεων, αφού αυτές παράγονται με συγκεκριμένο τρόπο από μαθηματικά μοντέλα, αλλά η αριθμητική επεξεργασία ιστορικών χρονοσειρών, μέσω μοντέλων μηχανικής μάθησης, για την βελτίωση της πρόβλεψης στα μετρούμενα δεδομένα. Τα αποτελέσματα ήταν ικανοποιητικά αφού το σύστημα κατάφερε να αναγνωρίσει πρότυπα στα δεδομένα και να παράξει προβλέψεις κατα μέσο όρο καλύτερες από τις προγνώσεις της ίδιας ημέρας. Πειραματισμοί έγιναν επίσης σε ότι αφορά την έκταση ημερών προηγούμενης γνώσης των μοντέλων καθώς και στο πλήθος των απαιτούμενων ημερών πρόβλεψης.

Μεγάλο μέρος της εργασίας αναφέρεται στη μελέτη πανελλαδικών δεδομένων φορτίου της ENTSO-E. Η βελτίωση της πρόγνωσης ήταν επιτυχημένη και σε αυτή τη περίπτωση. Για τον συνδυασμό δεδομένων φορτίου και καιρού, είναι απαραίτητη η κατασκευή μιας συνολικής τιμής θερμοκρασίας σε πανελλαδικό επίπεδο. Για αυτό το λόγο, στη παρούσα εργασία, εισάγεται η ιδέα της χωρικής κανονικοποίησης τιμών θερμοκρασίας από τις μεγαλύτερες πόλεις της ηπειρωτικής Ελλάδας. Οι τιμές θερμοκρασίας σταθμίζονται ανα περιοχή και σε σχέση με τα δημογραφικά στοιχεία της εκάστοτε πόλης, παράγοντας το τελικό κανονικοποιημένο σήμα θερμοκρασίας που χρησιμοποιείται από το μοντέλο. Καλύτερη διαχείριση του προβλήματος θα μπορούσε να επιτευχθεί εάν είχαμε στη διάθεσή μας χωρικά αναφερόμενα δεδομένα φορτίου.

Τέλος, αναπτύχθηκε εργαλείο οπτικοποίησης των χρονοσειρών με το οποίο ο χρήστης μπορεί να προσπελάσει χρονοσειρές και προβλέψεις διάφορων μοντέλων σε ένα φιλικό προς τον χρήστη περιβάλλον.

5.2 Μελλοντικές εφαρμογές (Future Work)

Δεδομένης της πεπερασμένης χρονικής διάρκειας εκπόνησης της διπλωματικής εργασίας, ο κύριος προσανατολισμός ήταν στην εκτέλεση πειραμάτων με μοντέλα LSTM τα οποία έχουν αποδειχθεί τα τελευταία χρόνια ιδανικά για προβλέψεις χρονοσειρών. Όμως το πεδίο της βαθιάς μηχανικής μάθησης είναι ταχύτατα ανερχόμενο και μοντέλα και μέθοδοι πρόβλεψης αναπτυσσονται και βελτιώνονται συνεχώς. Ενδιαφέρον θα είχε η μελέτη state-of-the-art Γενετικών δικτύων όπως τα Generative Adversarial Networks (GAN).

Τα δίκτυα αυτά, αποτελούνται από δύο μοντέλα, τον Discriminator και τον Generator και στόχος τους είναι η παραγωγή δεδομένων, παρόμοιων με αυτά που βρίσκονται στο σύνολο εκπαίδευσης. Συγκεκριμένα, ο Generator παράγει πιθανές χρονοσειρές, δεδομένων κάποιων τιμών (τυχαίων ή άλλον σημάτων από το σύνολο δεδομένων) και ο Discriminator αποφασίζει αν το σήμα εισόδου που λαμβάνει είναι πραγματικό (για παράδειγμα μια πραγματική αλληλουχία θερμοκρασιών από το σύνολο δεδομένων) ή ψεύτικο, έχει παραχθεί δηλαδή από τον Generator. Η διαδικασία της εκπαίδευσης είναι ένα παιχνίδι minimax μεταξύ των δύο μοντέλων και μέσο αυτής, ο Discriminator γίνεται όλο και καλύτερος στο να κατηγοριοποιεί τα σήματα ως "ψεύτικα" ή αληθινά και ο Generator παράγει όλο και ορθότερα σήματα εξόδου. Τελικά το μοντέλο του Generator μπορεί να χρησιμοποιηθεί για την πρόβλεψη πιθανών χρονοσειρών, μόλις εκπαιδευτεί για επαρκές χρονικό διάστημα.

Επιπρόσθετα, όπως έχει αναφερθεί και σε προηγούμενα κεφάλαια, θα είχε μεγάλο ενδιαφέρον η μελέτη χρονοσειρών μεγαλύτερης ακρίβειας και για μικρότερες γεωγραφικές περιοχές, με αξιοποίηση επιπλέον μετρήσεων από μια ή περισσότερες συσκευές IoT σε κοντινή απόσταση. Η λήψη χρονοσειρών μεγαλύτερης ακρίβειας, σε τοπικό επίπεδο, μπορεί να γίνει μέσω συστημάτων βιομηχανικού αυτομάτου ελέγχου (SCADA). Αυτή η διαδικασία όμως δεν πραγματοποιήθηκε στην παρούσα εργασία, αφού η πρόσβαση σε τέτοιου είδους δεδομένα δεν είναι εφικτή, όπως αναφέρθηκε και στην εισαγωγή.

Η διερεύνηση της ιδέας της χωρικής κανονικοποίησης των δεδομένων καιρού θα μπορούσε να αναλυθεί σε εκτενέστερο βαθμό, εάν είχαμε στη διάθεσή μας και χωρικά αναφερόμενα δεδομένα φορτίου. Ο συνδυασμός και η ακριβής χρήση των δεδομένων αυτών, όπως για παράδειγμα η επιλογή των κατάλληλων μετεωρολογικών σταθμών για την εκάστοτε περιοχή, είναι ένα πρόβλημα από μόνο του και θα μπορούσε να λυθεί με διάφορες τεχνικές.

Στην περίπτωση που χωρικά αναφερόμενα δεδομένα φορτίου, χωρικής ανάλυσης μετρήσεων οι οποίες ενδεχομένως να αξιοποιούν και εργαλεία μηχανικής μάθησης, είναι διαθέσιμα, μπορεί να μελετηθεί η διάκριση του φορτίου που πραγματικά σχετίζεται με τον καιρό (οικιακό) από εκείνο που σχετίζεται με βιομηχανίες και εργοστασιακές εγκαταστάσεις, στο οποίο καταμετρώνται μεγάλες καταναλώσεις κυρίως μέσης τάσης, οι οποίες δεν χρειάζεται να ανήκουν στην πρόβλεψη διότι δηλώνουν οι ίδιες το φορτίο. Εάν υπάρχει επιπλέον πληροφορία της χωρικής κατανομής των βιομηχανικών/εργοστασιακών καταναλωτών, η διεξαγωγή των πειραμάτων μπορεί να γίνει ακριβέστερη.

Τέλος, ένα πρόβλημα το οποίο απασχολεί τους ερευνητές στο πεδίο της μηχανικής μάθησης γενικότερα, αλλά έχει και ενδιαφέρον στα πλαίσια της εργασίας, είναι ο εντοπισμός και η πρόβλεψη για "ειδικές" περιπτώσεις. Η ανίχνευση δηλαδή μοτίβων που δεν εμφανίζονται σε μεγάλη συχνότητα στο σύνολο των δεδομένων. Για παράδειγμα, δεν διερευνήθηκαν "ειδικές ημέρες" όπως για παράδειγμα οι αργίες ή τα σαββατοκύριακα. Μια έρευνα προς αυτή τη κατεύθυνση, θα μπορούσε να βελτιώσει την ορθότητα των προβλέψεων για τις συγκεκριμένες περιπτώσεις.

Όλα τα παραπάνω, μπορούν να αναλυθούν και σε επίπεδο τεχνολογίας λογισμικού. Η ανάπτυξη εφαρμογών πρόβλεψης απαιτεί κατάλληλη αρχιτεκτονική που επιτρέπει την αυτόματη ροή δεδομένων (μετρήσεων, προγνώσεων κ.α.), καθώς και συνεχή λειτουργία εκπαίδευσης και δοκιμής των μοντέλων.

Βιβλιογραφία

[Alma17] Abdulaziz Almalaq and George Edwards, "A review of deep learning methods applied on load forecasting", in *2017 16th IEEE international conference on machine learning and applications (ICMLA)*, pp. 511–516, IEEE, 2017.

[Beng94] Yoshua Bengio, Patrice Simard and Paolo Frasconi, "Learning long-term dependencies with gradient descent is difficult", *IEEE transactions on neural networks*, vol. 5, no. 2, pp. 157–166, 1994.

[Berg16] Michael A Berger, Liesel Hans, Kate Piscopo and Michael D Sohn, "Exploring the energy benefits of advanced water metering", *Lawrence Berkeley National Laboratory*, 2016.

[Bott10] Léon Bottou, "Large-scale machine learning with stochastic gradient descent", in *Proceedings of COMPSTAT'2010*, pp. 177–186, Springer, 2010.

[Bouk18] Salah Bouktif, Ali Fiaz, Ali Ouni and Mohamed Adel Serhani, "Optimal deep learning lstm model for electric load forecasting using feature selection and genetic algorithm: Comparison with machine learning approaches", *Energies*, vol. 11, no. 7, p. 1636, 2018.

[cola] "Understanding LSTM Networks", `https://colah.github.io/posts/2015-08-Understanding-LSTMs/`. Accessed: 2020-10-15.

[Cont03] Javier Contreras, Rosario Espinola, Francisco J Nogales and Antonio J Conejo, "ARIMA models to predict next-day electricity prices", *IEEE transactions on power systems*, vol. 18, no. 3, pp. 1014–1020, 2003.

[dark] "Dark Sky API", `https://darksky.net/dev`. Accessed: 2020-10-15.

[Dela08] Erik Delarue and William D'haeseleer, "Adaptive mixed-integer programming unit commitment strategy for determining the value of forecasting", *Applied Energy*, vol. 85, no. 4, pp. 171–181, 2008.

[Duch11] John Duchi, Elad Hazan and Yoram Singer, "Adaptive subgradient methods for online learning and stochastic optimization.", *Journal of machine learning research*, vol. 12, no. 7, 2011.

[Dutt02] John A Dutton, "Opportunities and priorities in a new era for weather and climate services", *Bulletin of the American Meteorological Society*, vol. 83, no. 9, pp. 1303–1312, 2002.

[Fall18] Seyedeh Narjes Fallah, Ravinesh Chand Deo, Mohammad Shojafar, Mauro Conti and Shahaboddin Shamshirband, "Computational intelligence approaches for energy load forecasting in smart energy management grids: state of the art, future challenges, and research directions", *Energies*, vol. 11, no. 3, p. 596, 2018.

[Fan12] S. Fan and R. J. Hyndman, "Short-Term Load Forecasting Based on a Semi-Parametric Additive Model", *IEEE Transactions on Power Systems*, vol. 27, no. 1, pp. 134–141, 2012.

[Fawa19] Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar and Pierre-Alain Muller, "Deep learning for time series classification: a review", *Data Mining and Knowledge Discovery*, vol. 33, no. 4, pp. 917–963, 2019.

[Hahn09] Heiko Hahn, Silja Meyer-Nieberg and Stefan Pickl, "Electric load forecasting methods: Tools for decision making", *European journal of operational research*, vol. 199, no. 3, pp. 902–907, 2009.

[Hern14a] L. Hernandez, C. Baladron, J. M. Aguiar, B. Carro, A. J. Sanchez-Esguevillas, J. Lloret and J. Massana, "A Survey on Electric Power Demand Forecasting: Future Trends in Smart Grids, Microgrids and Smart Buildings", *IEEE Communications Surveys Tutorials*, vol. 16, no. 3, pp. 1460–1495, 2014.

[Hern14b] Luis Hernandez, Carlos Baladron, Javier M Aguiar, Belén Carro, Antonio J Sanchez-Esguevillas, Jaime Lloret and Joaquim Massana, "A survey on electric power demand forecasting: future trends in smart grids, microgrids and smart buildings", *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1460–1495, 2014.

[Hipp01] H. S. Hippert, C. E. Pedreira and R. C. Souza, "Neural networks for short-term load forecasting: a review and evaluation", *IEEE Transactions on Power Systems*, vol. 16, no. 1, pp. 44–55, 2001.

[Hirt18] Lion Hirth, Jonathan Mühlenpfordt and Marisa Bulkeley, "The ENTSO-E Transparency Platform–A review of Europe's most ambitious electricity data platform", *Applied energy*, vol. 225, pp. 1054–1067, 2018.

[Hoch91] Hochreiter, "DIPLOMARBEIT IM FACH INFORMATIK Untersuchungen zu dynamischen neuronalen Netzen", Master's thesis, Technische Universität München, 1991. (German).

[Hoch97] Sepp Hochreiter and Jürgen Schmidhuber, "LSTM can solve hard long time lag problems", in *Advances in neural information processing systems*, pp. 473–479, 1997.

[Hong10] Tao Hong, Pu Wang, Anil Pahwa, Min Gui and Simon M Hsiang, "Cost of temperature history data uncertainties in short term electric load forecasting", in *2010 IEEE 11th International Conference on Probabilistic Methods Applied to Power Systems*, pp. 212–217, IEEE, 2010.

[Huan15] Zhiheng Huang, Wei Xu and Kai Yu, "Bidirectional LSTM-CRF models for sequence tagging", *arXiv preprint arXiv:1508.01991*, 2015.

[Hynd10] R. J. Hyndman and S. Fan, "Density Forecasting for Long-Term Peak Electricity Demand", *IEEE Transactions on Power Systems*, vol. 25, no. 2, pp. 1142–1153, 2010.

[Jing18] XIE Jingrui and HONG Tao, "Load forecasting using 24 solar terms", *Journal of Modern Power Systems and Clean Energy*, vol. 6, no. 2, pp. 208–214, 2018.

[Kare18] Zahra Karevan and Johan AK Suykens, "Spatio-temporal stacked LSTM for temperature prediction in weather forecasting", *arXiv preprint arXiv:1811.06341*, 2018.

[Kavi16] Mike Kavis, "Investor's Guide to IoT–Part 1: Understanding The Ecosystem", *Cloud Technology Partners*, 2016.

[King14] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization", *arXiv preprint arXiv:1412.6980*, 2014.

[Kriz12] Alex Krizhevsky, Ilya Sutskever and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks", in *Advances in neural information processing systems*, pp. 1097–1105, 2012.

[Kust17] Corentin Kuster, Yacine Rezgui and Monjur Mourshed, "Electrical load forecasting models: A critical systematic review", *Sustainable cities and society*, vol. 35, pp. 257–270, 2017.

[LeCu98] Yann LeCun, Léon Bottou, Yoshua Bengio and Patrick Haffner, "Gradient-based learning applied to document recognition", *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[Mahm14] Anzar Mahmood, Nadeem Javaid, Adnan Zafar, Raja Ali Riaz, Saeed Ahmed and Sohail Razzaq, "Pakistan's overall energy potential assessment, comparison of LNG, TAPI and IPI gas projects", *Renewable and sustainable energy reviews*, vol. 31, pp. 182–193, 2014.

[mcro] "mcron - Mellor's cron daemon", `https://www.gnu.org/software/mcron/manual/mcron.pdf`. Accessed: 2020-10-15.

[Nowo16] Jakub Nowotarski, Bidong Liu, Rafał Weron and Tao Hong, "Improving short term load forecast accuracy via combining sister forecasts", *Energy*, vol. 98, pp. 40–49, 2016.

[Oztu10] Ilhan Ozturk, "A literature survey on energy–growth nexus", *Energy policy*, vol. 38, no. 1, pp. 340–349, 2010.

[Payn10] James E Payne, "A survey of the electricity consumption-growth literature", *Applied energy*, vol. 87, no. 3, pp. 723–731, 2010.

[Quil15] F. L. Quilumba, W. Lee, H. Huang, D. Y. Wang and R. L. Szabados, "Using Smart Meter Data to Improve the Accuracy of Intraday Load Forecasting Considering Customer Behavior Similarities", *IEEE Transactions on Smart Grid*, vol. 6, no. 2, pp. 911–918, 2015.

[Rume85] David E Rumelhart, Geoffrey E Hinton and Ronald J Williams, "Learning internal representations by error propagation", Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[Side20] George Sideratos, Andreas Ikonomopoulos and Nikos D Hatziargyriou, "A novel fuzzy-based ensemble model for load forecasting using hybrid deep neural networks", *Electric Power Systems Research*, vol. 178, p. 106025, 2020.

[Sobh19] Masoud Sobhani, Allison Campbell, Saurabh Sangamwar, Changlin Li and Tao Hong, "Combining weather stations for electric load forecasting", *Energies*, vol. 12, no. 8, p. 1510, 2019.

[Sola97] J Sola and Joaquin Sevilla, "Importance of input data normalization for the application of neural networks to complex industrial problems", *IEEE Transactions on nuclear science*, vol. 44, no. 3, pp. 1464–1468, 1997.

[Tuke77] John W Tukey, *Exploratory data analysis*, vol. 2, Reading, MA, 1977.

[Wang19] Y. Wang, Q. Chen, T. Hong and C. Kang, "Review of Smart Meter Data Analytics: Applications, Methodologies, and Challenges", *IEEE Transactions on Smart Grid*, vol. 10, no. 3, pp. 3125–3148, 2019.

[Xie17] Jingrui Xie and Tao Hong, "Wind speed for load forecasting models", *Sustainability*, vol. 9, no. 5, p. 795, 2017.

[Xie18a] J. Xie, Y. Chen, T. Hong and T. D. Laing, "Relative Humidity for Load Forecasting Models", *IEEE Transactions on Smart Grid*, vol. 9, no. 1, pp. 191–198, 2018.

[Xie18b] J. Xie and T. Hong, "Temperature Scenario Generation for Probabilistic Load Forecasting", *IEEE Transactions on Smart Grid*, vol. 9, no. 3, pp. 1680–1687, 2018.

[Yild18] Özal Yildirim, "A novel wavelet sequence based on deep bidirectional LSTM network model for ECG signal classification", *Computers in biology and medicine*, vol. 96, pp. 189–202, 2018.

[Zeil12] Matthew D Zeiler, "Adadelta: an adaptive learning rate method", *arXiv preprint arXiv:1212.5701*, 2012.

[Zhan18] Guoqiang Zhang and Haopeng Li, "Effectiveness of scaled exponentially-regularized linear units (SERLUs)", *arXiv preprint arXiv:1807.10117*, 2018.