



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΣΥΣΤΗΜΑΤΩΝ ΑΠΟΦΑΣΕΩΝ ΚΑΙ ΔΙΟΙΚΗΣΗΣ

Ανάπτυξη Συστήματος Διαδικασιών Μηχανικής
Μάθησης για τη Βραχυπρόθεσμη Πρόβλεψη
Ζήτησης Ηλεκτρικής Ενέργειας

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ
ΘΕΟΔΟΣΙΟΣ ΠΟΥΝΤΡΙΔΗΣ

Επιβλέπων: Δημήτριος Ασκούνης
Καθηγητής Ε.Μ.Π.

Αθήνα, Μάρτιος 2023



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΣΥΣΤΗΜΑΤΩΝ ΑΠΟΦΑΣΕΩΝ ΚΑΙ ΔΙΟΙΚΗΣΗΣ

Ανάπτυξη Συστήματος Διαδικασιών Μηχανικής Μάθησης για τη Βραχυπρόθεσμη Πρόβλεψη Ζήτησης Ηλεκτρικής Ενέργειας

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ
ΘΕΟΔΟΣΙΟΣ ΠΟΥΝΤΡΙΔΗΣ

Επιβλέπων: Δημήτριος Ασκούνης
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 9^η Μαρτίου 2023.

(Υπογραφή)	(Υπογραφή)	(Υπογραφή)
Ιωάννης Ψαρράς Καθηγητής Ε.Μ.Π.	Δημήτριος Ασκούνης Καθηγητής Ε.Μ.Π.	Χρυσόστομος Δούκας Καθηγητής Ε.Μ.Π.

Αθήνα, Μάρτιος 2023

Copyright © Θεοδόσιος Πουντρίδης, 2023.
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.

(Υπογραφή)

Θεοδόσιος Πουντρίδης

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

Περίληψη

Η βραχυπρόθεσμη πρόβλεψη ζήτησης ηλεκτρικής ενέργειας αποτελεί σημαντικό εργαλείο για την διασφάλιση της εύρυθμης λειτουργίας των συστημάτων ηλεκτρικής ενέργειας και για την αγοραπωλησία ενέργειας. Επίσης, μειώνει το κόστος αγοράς και παραγωγής της ενέργειας από τους παρόχους, πράγμα που μπορεί να μειώσει την τιμή και για τους καταναλωτές. Για την επίτευξη του προαναφερθέντος στόχου είναι ιδιαίτερα χρήσιμη η χρήση μηχανικής μάθησης. Ωστόσο, η διαδικασία της εφαρμογής τεχνικών μηχανικής μάθησης δύναται να καταστεί μια περίπλοκη και χρονοβόρα διαδικασία. Στην παρούσα εργασία πραγματοποιείται ανάπτυξη εργαλείου λογισμικού με σκοπό την αυτοματοποίηση του κύκλου ζωής μηχανικής μάθησης της πρόβλεψης φορτίου με χρήση των μεθόδων διαδικασιών μηχανικής μάθησης (MLOps - Machine Learning Operations). Η εφαρμογή αυτή διευκολύνει και επιταχύνει κατά πολύ την διαδικασία αυτή. Από την εφαρμογή αυτή υποστηρίζεται πληθώρα μοντέλων τόσο μηχανικής μάθησης, όσο και βαθιάς μάθησης, καθώς και πολυμεταβλητές (multivariate) και πολλαπλές (multiple) χρονοσειρές. Επίσης, παρέχεται και υποστήριξη για συμμεταβλητές (covariates). Σε αυτό το πλαίσιο, πραγματοποιείται εφαρμογή του εν λόγω εργαλείου στο πρόβλημα πρόβλεψης πολλαπλών χρονοσειρών είκοσι χωρών της Ευρώπης. Γίνεται συγκριτική μελέτη για τα μοντέλα Light Gradient Boosting Machine (LightGBM) και Neural Basis Expansion Analysis for Time Series forecasting (NBeats). Για τις δύο μεθόδους αυτές εκτελείται βελτιστοποίηση υπερπαραμέτρων, η οποία δείχνει την σημαντικότητα των παραμέτρων του κάθε μοντέλου. Επιπλέον, πραγματοποιείται αξιολόγηση των βέλτιστων μοντέλων, και εξήγηση των αποτελεσμάτων τους και των πλεονεκτημάτων του κάθε μοντέλου. Επιπροσθέτως, για την χρονοσειρά της Πορτογαλίας πραγματοποιείται εξήγηση της συμβολής της κάθε χρονικής στιγμής του παραθύρου πρότερης γνώσης των προαναφερθέντων μοντέλων με χρήση SHapley Additive exPlanations (SHAP). Ακόμα, πραγματοποιείται σύγκριση με τα αποτελέσματα των μοντέλων αν εκπαιδευτούν σε κάθε χρονοσειρά ξεχωριστά. Τελικά, η αυτοματοποίηση που προσφέρει η εφαρμογή της παρούσας εργασίας, καθώς και η επιτάχυνση του κύκλου ζωής μηχανικής μάθησης και η υψηλή ακρίβεια των αποτελεσμάτων την καθιστά απαραίτητο εργαλείο για τους παρόχους ηλεκτρικής ενέργειας.

Λέξεις κλειδιά

Διαδικασίες Μηχανικής Μάθησης (MLOps), MLflow, Μηχανική Μάθηση (ML), Βαθιά Μάθηση (DL), Πρόβλεψη Χρονοσειρών, Πρόβλεψη Φορτίου Βραχυπρόθεσμου Χρόνου (STLF), NBeats, LightGBM, Πολλαπλές Χρονοσειρές

Abstract

Short-term load forecasting is an important tool for ensuring the smooth operation of power systems and for energy trading. It also reduces the cost of suppliers to purchase and produce energy, which can also reduce the price for consumers. To achieve the above-mentioned objective, the use of machine learning is very useful. However, the process of applying machine learning techniques can become a complex and time-consuming task. In this thesis, a software tool is developed to automate the machine learning life cycle of load forecasting using Machine Learning Operations (MLOps) methods. This application facilitates and accelerates this process greatly. The application supports a variety of both machine learning and deep learning models, as well as multivariate and multiple time series. Support for covariates is also provided. Then, the tool is applied to the multiple time series forecasting problem of twenty European countries. A comparative study of the Light Gradient Boosting Machine (LightGBM) model and the Neural Basis Expansion Analysis for Time Series forecasting (NBeats) model is done. For these two methods, hyperparameter optimization is performed, which shows the importance of the parameters of each model. In addition, the optimal models are evaluated for all timeseries, and their results are explained, along with the advantages of each model. In addition, for the Portuguese time series, an explanation of the contribution of each time point of the lookback window of the aforementioned models is carried out using SHapley Additive exPlanations (SHAP). Also, these results are compared with the results of the models if they are trained on each time series separately. Finally, the automation offered by the application of this thesis, as well as the acceleration of the machine learning life cycle and the high accuracy of the results makes it an indispensable tool for electricity providers.

Key words

Machine Learning Operations (MLOps), MLflow, Machine Learning (ML), Deep Learning (DL), Time Series Prediction, Short Term Load Forecast (STLF), NBeats, LightGBM, Multiple Time Series

Ευχαριστίες

Αρχικά, θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου κ. Δημήτρη Ασκούνη για την δυνατότητα να ασχοληθώ με το αντικείμενο της βραχυπρόθεσμης πρόβλεψης ζήτησης ηλεκτρικής ενέργειας, και την εμπιστοσύνη που μου έδειξε αναθέτοντάς μου αυτό το θέμα. Επίσης, θα ήθελα να ευχαριστήσω θερμά τον κ. Σωτήρη Πελέκη, για την καθοδήγησή και την βοήθειά του καθόλη την διάρκεια της εκπόνησης της παρούσας εργασίας. Τέλος, θα ήθελα να ευχαριστήσω την οικογένεια μου και τους φίλους μου για την ανιδιοτελή στήριξή τους κατά την διάρκεια των σπουδών μου

Περιεχόμενα

Λίστα Εικόνων	1
Λίστα Πινάκων	3
1 Εισαγωγή	5
1.1 Πρόβλεψη Ζήτησης Ηλεκτρικής Ενέργειας	5
1.2 Σκοπός και συνεισφορά	6
1.3 Διάρθρωση τόμου	7
2 State-of-the-Art εργαλεία και μέθοδοι	9
2.1 Η μεθοδολογία MLOps (Machine Learning Operations)	9
2.1.1 Διαδικασία επίλυσης προβλημάτων με χρήση μηχανικής μάθησης	9
2.1.2 Οι μέθοδοι DevOps και MLOps	11
2.1.3 Οι ρυθμίσεις (Setups) του MLOps	12
2.2 Σύντομη βιβλιογραφική ανάλυση για την βραχυπρόθεσμη πρόβλεψη ζήτησης ηλεκτρικής ενέργειας και την χρήση πολλαπλών χρονοσειρών	15
2.3 Μοντέλα που θα χρησιμοποιηθούν στην παρούσα εργασία	16
2.3.1 LightGBM	16
2.3.2 NBEATS	17
3 Ανάπτυξη της Εφαρμογής	19
3.1 Σύντομη παρουσίαση της εφαρμογής	19
3.2 Στάδιο Ελέγχου της σωλήνωσης	21
3.3 Αρχική φόρτωση και έλεγχος των δεδομένων	21
3.4 Επεξεργασία δεδομένων	23
3.4.1 Εντοπισμός ακραίων τιμών (Outlier detection)	23
3.4.2 Αντικατάσταση άγνωστων τιμών (imputation)	24
3.4.3 Χειρισμός πολυμεταβλητών ή/και πολλαπλών χρονοσειρών	25
3.5 Εκπαίδευση	25
3.5.1 Χειρισμός πολλαπλών η/και πολυμεταβλητών χρονοσειρών	26
3.6 Αξιολόγηση	26
3.6.1 Σύντομη περιγραφή του SHAP	27
3.6.2 Υλοποίηση SHAP για χρονοσειρές	28
3.7 Βελτιστοποίηση υπερπαραμέτρων (hyperparameter tuning) με χρήση της βιβλιοθήκης Optuna	29
3.7.1 Θεωρητική ανάλυση	29
3.7.2 Υλοποίηση	30
4 Πειραματική διαδικασία	32
4.1 Εισαγωγή	32
4.2 Προετοιμασία του συνόλου δεδομένων	32
4.3 Αρχική φόρτωση και έλεγχος των δεδομένων	33
4.4 Επεξεργασία δεδομένων	33
4.4.1 Εντοπισμός ακραίων τιμών (Outlier detection)	33
4.4.2 Αντικατάσταση άγνωστων τιμών (Imputation)	37
4.5 Επιλογή μοντέλου και μετρικής	39
4.6 Αναλυτικά αποτελέσματα μοντέλων Naive	39

4.7	Αναλυτικά αποτελέσματα μοντέλου NBEATS	41
4.7.1	Βελτιστοποίηση υπερπαραμέτρων	41
4.7.2	Αξιολόγηση μοντέλου και SHAP	45
4.7.3	Αποτελέσματα προσέγγισης ενός μοντέλου ανά χρονοσειρά	51
4.8	Αναλυτικά αποτελέσματα μοντέλου LightGBM	52
4.8.1	Βελτιστοποίηση υπερπαραμέτρων	52
4.8.2	Αξιολόγηση μοντέλου και SHAP	55
4.8.3	Αποτελέσματα προσέγγισης ενός μοντέλου ανά χρονοσειρά	61
4.9	Συγκεντρωτικά αποτελέσματα όλων των μοντέλων	62
5	Συμπεράσματα και Μελλοντικές προεκτάσεις	64
5.1	Συμπεράσματα και Επίλογος	64
5.2	Μελλοντικές προεκτάσεις	65
	Παραρτήματα	70
A	Παράμετροι εφαρμογής και η σημασία τους	70

Λίστα Εικόνων

2.1	Πιθανή ρύθμιση ανάπτυξης ενός μοντέλου μηχανικής μάθησης χωρίς τις αρχές MLOps	13
2.2	Πιθανή ρύθμιση ανάπτυξης ενός μοντέλου μηχανικής μάθησης με αυτοματοποίηση μέσω των σωληνώσεων	14
2.3	Πιθανή ρύθμιση ανάπτυξης μοντέλου μηχανικής μάθησης με αυτοματοποίηση μέσω συνεχούς παράδοσης της ρύθμισης σωληνώσεων	15
2.4	Η αρχιτεκτονική του NBEATS	18
3.1	Η αρχιτεκτονική της εφαρμογής	20
3.2	Τα στάδια της σωλήνωσης	21
3.3	Η αθροιστική φύση των τιμών SHAP	28
4.1	Οι τιμές που αφαιρέθηκαν από την χρονοσειρά του Βελγίου	34
4.2	Η τελική μορφή της χρονοσειράς του Βελγίου	34
4.3	Οι τιμές που αφαιρέθηκαν από την χρονοσειρά της Ελλάδας	35
4.4	Η τελική μορφή της χρονοσειράς της Ελλάδας	35
4.5	Οι τιμές που αφαιρέθηκαν από την χρονοσειρά της Πορτογαλίας	36
4.6	Η τελική μορφή της χρονοσειράς της Πορτογαλίας	36
4.7	Τιμές που αντικαταστάθηκαν από την χρονοσειρά της Νορβηγίας	38
4.8	Το MAPE των δεδομένων επικύρωσης για τα τρεξίματα της βελτιστοποίησης υπερπαραμέτρων του NBEATS	43
4.9	Το πόσο σημαντική είναι η κάθε υπερπαραμέτρος που βελτιστοποιήθηκε στο τελικό αποτέλεσμα του NBEATS	44
4.10	Αναλυτικά αποτελέσματα ανά υπερπαραμέτρο του NBEATS	44
4.11	Πρόβλεψη για το 2021 του NBEATS για την χρονοσειρά της Πορτογαλίας	46
4.12	Πρόβλεψη για την δεύτερη εβδομάδα του 2021 για την χρονοσειρά της Πορτογαλίας απο το μοντέλο NBEATS	47
4.13	Οι συντελεστές SHAP ενός δείγματος για την πρώτη τιμή της εξόδου του μοντέλου NBEATS	48
4.14	Οι συντελεστές SHAP ενός δείγματος για την δέκατη τρίτη τιμή της εξόδου του μοντέλου NBEATS	48
4.15	Οι συντελεστές SHAP ενός δείγματος για την εικοστή τέταρτη τιμή της εξόδου του μοντέλου NBEATS	48
4.16	Οι συντελεστές SHAP όλων των δειγμάτων για την πρώτη τιμή της εξόδου του μοντέλου NBEATS	49
4.17	Μέση τιμή των τιμών SHAP για την πρώτη τιμή της εξόδου του μοντέλου NBEATS	50
4.18	Οι συντελεστές SHAP όλων των δειγμάτων και η μέση τιμή τους για την δέκατη τρίτη τιμή της εξόδου του μοντέλου NBEATS	50
4.19	Οι συντελεστές SHAP όλων των δειγμάτων και η μέση τιμή τους για την εικοστή τέταρτη τιμή της εξόδου του μοντέλου NBEATS	51
4.20	Το MAPE των δεδομένων επικύρωσης για τα τρεξίματα της βελτιστοποίησης υπερπαραμέτρων του LightGBM	53
4.21	Το πόσο σημαντική είναι η κάθε υπερπαραμέτρος που βελτιστοποιήθηκε στο τελικό αποτέλεσμα του LightGBM	54

4.22	Αναλυτικά αποτελέσματα ανά υπερπαράμετρο του LightGBM	55
4.23	Πρόβλεψη για το 2021 του LightGBM για την χρονοσειρά της Πορτογαλίας	57
4.24	Πρόβλεψη για την δεύτερη εβδομάδα του 2021 για την χρονοσειρά της Πορτογαλίας απο το μοντέλο LightGBM	57
4.25	Οι συντελεστές SHAP ενός δείγματος για την πρώτη τιμή της εξοδου του μοντέλου LightGBM	58
4.26	Οι συντελεστές SHAP ενός δείγματος για την δέκατη τρίτη τιμή της εξοδου του μοντέλου LightGBM	58
4.27	Οι συντελεστές SHAP ενός δείγματος για την εικοστή τέταρτη τιμή της εξοδου του μοντέλου LightGBM	59
4.28	Οι συντελεστές SHAP όλων των δειγμάτων για την πρώτη τιμή της εξοδου του μοντέλου LightGBM	59
4.29	Μέση τιμή των τιμών SHAP για την πρώτη τιμή της εξοδου του μοντέλου LightGBM	60
4.30	Οι συντελεστές SHAP όλων των δειγμάτων και η μέση τιμή τους για την δέκατη τρίτη τιμή της εξοδου του μοντέλου LightGBM	60
4.31	Οι συντελεστές SHAP όλων των δειγμάτων και η μέση τιμή τους για την εικοστή τέταρτη τιμή της εξοδου του μοντέλου LightGBM	61

Λίστα Πινάκων

3.1	Μορφή αρχείου csv απλών χρονοσειρών	22
3.2	Μορφή αρχείου csv πολλαπλών η/και πολυμεταβλητών χρονοσειρών . .	22
4.1	Αριθμός ακραίων τιμών που αφαιρέθηκαν ανά χρονοσειρά	37
4.2	Αριθμός τιμών που αντικαταστάθηκαν ανά χώρα	39
4.3	Αποτελέσματα του μοντέλου Naive μιας ημέρας ανά χώρα	40
4.4	Αποτελέσματα του μοντέλου Naive μιας εβδομάδας ανά χώρα	41
4.5	Τιμές του μοντέλου NBEATS με το χαμηλότερο MAPE	43
4.6	Αποτελέσματα του μοντέλου NBEATS	46
4.7	Αποτελέσματα των μοντέλων NBEATS απλών χρονοσειρών	52
4.8	Τιμές του μοντέλου LightGBM με το χαμηλότερο MAPE	54
4.9	Αποτελέσματα του μοντέλου LightGBM	56
4.10	Αποτελέσματα των μοντέλων απλών χρονοσειρών LightGBM	62
4.11	Συνοπτικά αποτελέσματα όλων των μοντέλων	62

Κεφάλαιο 1

Εισαγωγή

Την σημερινή εποχή, τα δεδομένα παράγονται με πρωτοφανείς ρυθμούς, δημιουργώντας τεράστια ποσότητα πληροφορίας που απαιτεί ανάλυση. Η ανάλυση χρονοσειρών αποτελεί βασικό εργαλείο για την κατανόηση και την ανάλυση χρονικά εξαρτώμενων δεδομένων, επιτρέποντας στους αναλυτές διαφόρων τομέων να κάνουν προβλέψεις και να διαμορφώσουν αποτελεσματικές στρατηγικές διαχείρισης προβλημάτων.

Η ανάλυση χρονοσειρών αποτελεί τη μελέτη δεδομένων που συλλέγονται για κάποιο μέγεθος για μια συγκεκριμένη περίοδο, όπως οι μετρήσεις θερμοκρασίας, οι τιμές των μετοχών στο χρηματιστήριο ή τα στοιχεία πωλήσεων. Πρόκειται για μια ισχυρή τεχνική που επιτρέπει στους αναλυτές να εντοπίζουν τάσεις, μοτίβα και σχέσεις στα δεδομένα. Η ικανότητα μοντελοποίησης και πρόβλεψης μελλοντικών τάσεων και συμπεριφορών με βάση δεδομένα του παρελθόντος έχει καταστήσει την ανάλυση χρονοσειρών κρίσιμο εργαλείο σε διάφορους τομείς, όπως στην οικονομία, στην υγεία, στην μηχανική, στην υγειονομική περίθαλψη και στην περιβαλλοντική επιστήμη.

Στην εν λόγω εργασία θα ασχοληθούμε με την κατασκευή εφαρμογής για πρόβλεψη ζήτησης ηλεκτρικής ενέργειας. Παρακάτω ακολουθεί σύντομη ανάλυση του τομέα αυτού της ανάλυσης χρονοσειρών.

1.1 Πρόβλεψη Ζήτησης Ηλεκτρικής Ενέργειας

Η πρόβλεψη ζήτησης ηλεκτρικής ενέργειας αναφέρεται στη διαδικασία πρόβλεψης της μελλοντικής ζήτησης ηλεκτρικής ενέργειας από τους καταναλωτές σε μια δεδομένη γεωγραφική περιοχή ή αγορά. Αποτελεί βασικό εργαλείο για τους παρόχους ενέργειας προκειμένου να διασφαλίσουν ότι μπορούν να παρέχουν επαρκή ηλεκτρική ενέργεια για να καλύψουν τις ανάγκες των πελατών τους. Πιο συγκεκριμένα, η ακριβής πρόβλεψη φορτίου επιτρέπει στους παρόχους να λαμβάνουν αποφάσεις σχετικά με την παραγωγή και τη διανομή, να βελτιώνουν την αξιοπιστία του συστήματος και να βελτιστοποιούν την χρήση των πόρων για τη μείωση του κόστους.

Η πρόβλεψη ζήτησης ηλεκτρικής ενέργειας είναι μια σύνθετη διαδικασία, καθώς η ζήτηση εξαρτάται από μια πληθώρα παραγόντων όπως από τα καιρικά φαινόμενα και την ανθρώπινη δραστηριότητα. Αυτά δίνουν σε αυτές τις χρονοσειρές εποχικότητα. Οι μέθοδοι που μπορεί να χρησιμοποιηθούν για την πρόβλεψη ζήτησης βασίζονται σε στατιστική ανάλυση, μηχανική μάθηση, βαθιά μάθηση ή σε κάποιον συνδυασμό των προαναφερθέντων. Η ακρίβεια των μοντέλων πρόβλεψης φορτίου είναι κρίσιμη, καθώς επηρεάζει άμεσα την ικανότητα των παρόχων ενέργειας να παρέχουν ηλεκτρική ενέργεια στους πελάτες τους.

Η πρόβλεψη φορτίου έχει γίνει όλο και πιο σημαντική τα τελευταία χρόνια λόγω της αυξανόμενης ζήτησης για ανανεώσιμες πηγές ενέργειας, όπως η αιολική και η ηλιακή ενέργεια, οι οποίες εξαρτώνται σε μεγάλο βαθμό από τις καιρικές συνθήκες. Η πρόβλεψη φορτίου είναι επίσης κρίσιμη για τους παρόχους ενέργειας για την αγορά και πώληση ενέργειας στο χρηματιστήριο ενέργειας, όπου οι τιμές καθορίζονται με βάση την προσφορά και τη ζήτηση. Η ακριβής πρόβλεψη επιτρέπει στους παρόχους να αγοράζουν ή να παράγουν ηλεκτρική ενέργεια στο χαμηλότερο δυνατό κόστος, μειώνοντας το συνολικό κόστος της ενέργειας για τους καταναλωτές. Αυτό είναι ιδιαίτερα

σημαντικό σήμερα λόγω της ενεργειακής κρίσης.

Επίσης, τα αποτελέσματα των προβλέψεων ηλεκτρικής ενέργειας μπορούν να χρησιμοποιηθούν και από τον καταναλωτή για την ρύθμιση και μείωση ενέργειάς που καταναλώνεται στο κτήριό του. Πιο συγκεκριμένα, ο καταναλωτής μπορεί να ρυθμίσει την ενεργειακή συμπεριφορά του κτιρίου μέσω έξυπνων συσκευών, κάνοντάς τις να λειτουργούν όταν η ζήτηση είναι χαμηλή, πληρώνοντας έτσι χαμηλότερες τιμές για την ενέργεια που καταναλώνει.

Τέλος, στην παρούσα εργασία θα ασχοληθούμε με την βραχυπρόθεσμη πρόβλεψη ζήτησης ηλεκτρικής ενέργειας. Αυτή αποτελεί πρόβλεψη ενέργειας για χρόνο από μία μέρα έως μία εβδομάδα μετά την τελευταία γνωστή τιμή, με δεδομένα ανάλυσης από 15 μέχρι 60 λεπτών. Χρησιμοποιείται για εφαρμογές σχετικές με την προμήθεια της ηλεκτρικής ενέργειας και με το σύστημα διανομής. Εάν η πρόβλεψη είναι πολύ χαμηλή, μπορεί να υπάρξει έλλειψη ηλεκτρικής ενέργειας, με αποτέλεσμα διακοπές ρεύματος, και εάν η πρόβλεψη είναι πολύ υψηλή, τότε θα υπήρχε υπερεκτίμηση φορτίου, με αποτέλεσμα αυξημένο κόστος είτε υπερεκτίμηση παραγωγής που θα μπορούσε να οδηγήσει πάλι σε διακοπές ρεύματος.

1.2 Σκοπός και συνεισφορά

Ο σκοπός της παρούσας εργασίας είναι η περαιτέρω ανάπτυξη προϋπάρχουσας εφαρμογής για την διευκόλυνση της διαδικασίας της πρόβλεψης ζήτησης ηλεκτρικής ενέργειας με χρήση μεθόδων μηχανικής μάθησης (Machine Learning) και βαθιάς μάθησης (Deep Learning). Όπως θα αναλυθεί στα κεφάλαια παρακάτω, ο χρήστης μπορεί με αρκετά μεγάλη ευκολία να εκτελέσει το μεγαλύτερο τμήμα του κύκλου ζωής μηχανικής μάθησης (ML Lifecycle) αυτόματα, με την χρήση μιας εντολής. Τα αποτελέσματα αυτής της διαδικασίας αποθηκεύονται σε διακομιστή με τέτοιο τρόπο ώστε να είναι εύκολα προσβάσιμα στον χρήστη.

Για την επιτυχή πραγματοποίηση του παραπάνω στόχου, η εφαρμογή αυτή υποστήριζε τις παρακάτω λειτουργίες:

- Υποστήριξη απλών χρονοσειρών και συμμεταβλητών (covariates)
- Δυνατότητα εκπαίδευσης και αξιολόγησης πληθώρας μοντέλων τόσο μηχανικής μάθησης, όσο και βαθιάς μάθησης (NBEATS, Transformer, NHiTS, TCN, RNN, BlockRNN, TFT, LightGBM, RandomForest και Naive)

- Αποθήκευση αποτελεσμάτων στο MLflow

Σε αυτά, στο πλαίσιο της παρούσας εργασίας προστέθηκαν τα ακόλουθα:

- Υποστήριξη πολλαπλών (multiple) και πολυμεταβλητών χρονοσειρών
- Αυτόματη επεξεργασία των χρονοσειρών με ανίχνευση ακραίων τιμών (outlier detection) και αντικατάσταση άγνωστων τιμών (imputation)
- Υποστήριξη εξήγησης των αποτελεσμάτων και της σημαντικότητας του κάθε χαρακτηριστικού με χρήση SHAP [1]
- Δυνατότητα βελτιστοποίησης υπερπαραμέτρων τόσο με χρήση αναζήτησης πλέγματος (grid search), όσο και με χρήση πιο σύγχρονων μεθόδων (TPE), που βασίζονται στην στατιστική.

Πρέπει σε αυτό το σημείο να αναφερθεί ότι η εφαρμογή αυτή έχει και πολλές γενικές μεθόδους έτσι ώστε να μπορεί να χρησιμοποιηθεί και για την πρόβλεψη διαφορετικών χρονοσειρών.

Για να παρουσιαστεί η χρησιμότητα της εφαρμογής, πραγματοποιείται εφαρμογή της στο πρόβλημα της πρόβλεψης πολλαπλών χρονοσειρών ζήτησης ηλεκτρικής ενέργειας για 20 χώρες της Ευρώπης. Τα μοντέλα που χρησιμοποιήθηκαν είναι τόσο μηχανικής μάθησης (LightGBM), όσο και βαθιάς μάθησης (NBEATS). Για την διασφάλιση της επιτυχίας των προαναφερθέντων μοντέλων να επιλύσουν το παραπάνω πρόβλημα πραγματοποιήθηκαν τα εξής:

- Σύγκριση των μοντέλων με Naive μεθόδους για να διασφαλιστεί η χρησιμότητά τους
- Βελτιστοποίηση υπερπαραμέτρων για να έχουμε τα καλύτερα δυνατά αποτελέσματα. Σε αυτό το πλαίσιο, πραγματοποιήθηκε και ανάλυση της σημασίας της κάθε υπερπαραμέτρου
- Ανάλυση της σημαντικότητας του κάθε χαρακτηριστικού που δίνεται στο κάθε μοντέλο με χρήση της μεθόδου SHAP
- Σύγκριση των μοντέλων πολλαπλών χρονοσειρών με τα αντίστοιχά τους, αν αυτά εκπαιδεύονταν σε κάθε χώρα ξεχωριστά

Τα αποτελέσματα της ανάλυσης αυτής μπορούν να χρησιμοποιηθούν από τους διαχειριστές συστημάτων ηλεκτρικής ενέργειας ή από εταιρίες παραγωγής ενέργειας που συμμετέχουν στην αγορά ενέργειας για την βελτιστοποίηση της παραγωγής της. Παράλληλα, μέσω των προβλέψεων μεγάλης ακρίβειας που προσφέρει η εφαρμογή, οι εταιρίες ενέργειας μπορούν να καταστρώσουν βέλτιστες στρατηγικές αγοραπωλησίας ηλεκτρικής ενέργειας, μεγιστοποιώντας έτσι το κέρδος τους.

Τέλος, στόχος της χρήσης πολλαπλών χρονοσειρών και ενός μοντέλου για όλες τις χώρες είναι η μείωση της κατανάλωσης πόρων και της βιωσιμότητας της εφαρμογής μας. Έτσι, ο χρήστης θα μπορεί να χρησιμοποιεί ένα μόνο μοντέλο για την πρόβλεψη φορτίου πολλαπλών χωρών της Ευρώπης. Επίσης, οι χρονοσειρές των χωρών παράγονται με παρόμοιο τρόπο, και έτσι έχουν πολλά κοινά χαρακτηριστικά. Ένα μοντέλο πολλαπλών χρονοσειρών θα μπορούσε να χρησιμοποιήσει την γνώση από όλες τις χώρες πάνω στις οποίες έχει εκπαιδευτεί έτσι ώστε να προβλέψει με μεγαλύτερη ακρίβεια την ζήτηση κάθε χώρας. Τέλος, η χρήση περισσότερων δεδομένων μέσω της εκπαίδευσης σε πολλαπλές χρονοσειρές μειώνει την πιθανότητα υπερπροσαρμογής του μοντέλου στα δεδομένα (overfitting)[2].

1.3 Διάρθρωση τόμου

Στο κεφάλαιο 1 πραγματοποιείται εισαγωγή στην πρόβλεψη χρονοσειρών, και στον τομέα της βραχυπρόθεσμης πρόβλεψης ζήτησης ηλεκτρικής ενέργειας ειδικότερα. Επιπλέον, παρουσιάζεται εκτενώς ο σκοπός της παρούσας εφαρμογής, καθώς και του προβλήματος των πολλαπλών χρονοσειρών πάνω στο οποίο εφαρμόζεται.

Στο κεφάλαιο 2, γίνεται ανάλυση των state-of-the-art εργαλείων και μεθόδων που θα χρησιμοποιηθούν στην παρούσα εργασία. Πιο συγκεκριμένα, παρουσιάζεται η μεθοδολογία διαδικασιών μηχανικής μάθησης (MLOps), και πραγματοποιείται βιβλιογραφική ανάλυση για την πρόβλεψη ζήτησης ηλεκτρικής ενέργειας, και την χρήση

πολλαπλών χρονοσειρών. Τέλος, παρουσιάζονται τα μοντέλα μηχανικής και βαθιάς μάθησης που θα χρησιμοποιηθούν στην παρούσα εργασία.

Στο κεφάλαιο 3, παρουσιάζεται αναλυτικά η εφαρμογή που αναπτύχθηκε στο πλαίσιο αυτής της εργασίας. Πιο συγκεκριμένα, παρουσιάζονται τα 6 στάδια του MLflow που υλοποιήθηκαν: Το στάδιο ελέγχου της σωλήνωσης, το στάδιο αρχικής φόρτωσης των δεδομένων, το στάδιο επεξεργασίας δεδομένων, το στάδιο εκπαίδευσης, το στάδιο αξιολόγησης και το στάδιο βελτιστοποίησης υπερπαραμέτρων. Επίσης, γίνεται σύντομη θεωρητική ανάλυση των μεθόδων SHAP (SHapley Additive exPlanations) και TPE (tree-structured Parzen Estimator) που εφαρμόζονται στην βιβλιοθήκη optuna.

Στο κεφάλαιο 4, παρουσιάζεται η πειραματική διαδικασία που ακολουθήθηκε για το πρόβλημα των πολλαπλών χρονοσειρών. Συγκεκριμένα, αφού παρουσιαστεί το σύνολο δεδομένων πάνω στο οποίο θα δουλέψουμε, παρουσιάζονται τα αποτελέσματα της διαδικασίας ανίχνευσης ακραίων τιμών, και αντικατάστασης αγνώστων τιμών. Έπειτα, παρατίθενται τα αναλυτικά αποτελέσματα των μοντέλων Naive, NBEATS και LightGBM, μαζί με τις ακριβείς εντολές που εκτελέστηκαν. Τελικά, τα μοντέλα συγκρίνονται στον μέσο όρο του MAPE και του RMSE. Τα NBEATS και LightGBM συγκρίνονται με τα αντίστοιχά τους, αν αυτά εκπαιδεύονταν σε κάθε χώρα ξεχωριστά.

Στο κεφάλαιο 5 αναλύονται τα συμπεράσματα και οι μελλοντικές προεκτάσεις της παρούσας εργασίας.

Στο παράρτημα Α, παρουσιάζονται αναλυτικά όλες οι παράμετροι της εφαρμογής με την σημασία τους.

Κεφάλαιο 2

State-of-the-Art εργαλεία και μέθοδοι

Στο παρόν κεφάλαιο θα πραγματοποιηθεί ανάλυση των εργαλείων και μεθόδων που θα χρησιμοποιηθούν στην παρούσα εργασία

2.1 Η μεθοδολογία MLOps (Machine Learning Operations)

2.1.1 Διαδικασία επίλυσης προβλημάτων με χρήση μηχανικής μάθησης

Η εφαρμογή τεχνικών μηχανικής μάθησης μπορεί να καταστεί μια περίπλοκη εργασία. Επιπλέον είναι κυκλική, καθώς οι ανάγκες και η φύση του προβλήματος μεταβάλλονται συνεχώς. Συνεπώς είναι αναγκαίο να συλλέγονται ανά τακτά χρονικά διαστήματα δεδομένα σχετικά με την απόδοση της λύσης και αν αυτή χειροτερεύσει πολύ να επαναλαμβάνεται η διαδικασία εκ νέου. Η διαδικασία αυτή ονομάζεται κύκλος ζωής ML (ML lifecycle) και χωρίζεται σε 7 στάδια:

1. Συλλογή Δεδομένων

Για κάθε εφαρμογή μεθόδων μηχανικής μάθησης είναι αναγκαία η συλλογή δεδομένων. Τα δεδομένα αυτά θα πρέπει να είναι αντιπροσωπευτικά των περιπτώσεων που απαντώνται στον πραγματικό κόσμο για το πρόβλημα που μελετάται. Επίσης, είναι σημαντικό η ποσότητα των δεδομένων να είναι αρκετή για την λύση που έχει επιλεγεί, ειδικά για τις περιπτώσεις που γίνεται χρήση βαθιάς μάθησης. Τέλος, πρέπει να επιλεγούν οι πηγές των δεδομένων που θα χρησιμοποιηθούν. Για τον σκοπό αυτό μπορούν να γίνουν στατιστικές έρευνες, χειροκίνητη ανάθεση ετικετών σε δεδομένα για εκπαίδευση ή και χρήση δεδομένων χρηστών. Τα σύνολα δεδομένων μπορούν να ληφθούν είτε με χρήση αυτών των μεθόδων από μια εταιρία-οργανισμό είτε να ληφθούν από το διαδίκτυο από δημόσια διαθέσιμα σύνολα δεδομένων.

2. Αρχική Επεξεργασία Δεδομένων (Data wrangling)

Το σύνολο δεδομένων που χρησιμοποιείται συνήθως βρίσκεται σε μια μορφή ακατάλληλη για ανάλυση και εκπαίδευση των μοντέλων. Επομένως, είναι αναγκαία μια αρχική μετατροπή των ακατέργαστων δεδομένων σε χρήσιμα δεδομένα για χρήση στα επόμενα στάδια. Η διαδικασία αυτή ονομάζεται αρχική επεξεργασία δεδομένων (Data wrangling). Περιλαμβάνει την εξέταση των δεδομένων για τον εντοπισμό τυχόν προβλημάτων, όπως ελλείπουσες τιμές, διπλές καταχωρίσεις ή ακραίες τιμές, αλλά και την αλλαγή της οργάνωσης των δεδομένων με δημιουργία νέων στηλών ή τη μετονομασία στηλών. Τέλος, πραγματοποιείται αποθήκευση των καθαρών και οργανωμένων δεδομένων σε κατάλληλη μορφή (πχ csv) έτσι ώστε να γίνει εύκολη πρόσβαση από το επόμενο βήμα.

3. Διερευνητική Ανάλυση Δεδομένων (EDA)

Η διερευνητική ανάλυση δεδομένων αποτελεί μια μέθοδο για την ανάλυση και τη σύνοψη συνόλων δεδομένων με σκοπό την ανακάλυψη μοτίβων και σχέσεων που μπορεί να μην είναι άμεσα προφανείς. Στόχος της EDA είναι να κατανοηθεί

η υποκείμενη δομή των δεδομένων και να εντοπιστούν πιθανά ζητήματα ή περιοχές για περαιτέρω διερεύνηση. Εάν, για παράδειγμα, προκύψει από την EDA ότι ορισμένα χαρακτηριστικά των δεδομένων δεν επηρεάζουν την έξοδο, γίνεται αυτά να αφαιρεθούν προκειμένου να επιταχυνθεί η διαδικασία εκπαίδευσης και να βελτιωθεί ενδεχομένως η απόδοση του μοντέλου. Για να βοηθήσει στον εντοπισμό μοτίβων και σχέσεων στα δεδομένα η EDA περιλαμβάνει συνήθως τη χρήση διαγραμμάτων, όπως ιστογράμματα, διαγράμματα διασποράς (scatter plots) και θηκογράμματα (box plots). Περιλαμβάνει επίσης τη χρήση περιγραφικών στατιστικών, όπως ο μέσος όρος, η διάμεσος και η τυπική απόκλιση, για τη σύνοψη των δεδομένων.

4. Προεπεξεργασία Δεδομένων

Σε αυτό το βήμα, πραγματοποιείται (ενδεχομένως) κλιμάκωση των δεδομένων σε ένα πιο κατάλληλο εύρος, και μπορεί να εφαρμοστούν τεχνικές για μείωση διαστατικότητας όπως Ανάλυση Κύριων Συνιστωσών (Principal Component Analysis). Αυτό μπορεί να βελτιώσει την απόδοση του μοντέλου και να μειώσει τον κίνδυνο υπερπροσαρμογής (overfitting). Τέλος, γίνεται ο διαχωρισμός των δεδομένων σε σύνολο για εκπαίδευση (training), επικύρωση (validation) και τεστ. Το σύνολο τεστ χρησιμοποιείται για την αξιολόγηση της απόδοσης ενός μοντέλου σε δεδομένα που συναντά για πρώτη φορά, ενώ το σύνολο επικύρωσης χρησιμοποιείται για την επιλογή μοντέλων, την επιλογή της αρχιτεκτονικής που θα χρησιμοποιηθεί, την προσαρμογή των υπερπαραμέτρων (hyperparameter tuning) και την αξιολόγηση της απόδοσης του μοντέλου σε δεδομένα που δεν χρησιμοποιήθηκαν στη διαδικασία εκπαίδευσης.

5. Εκπαίδευση Μοντέλου

Στο στάδιο της εκπαίδευσης το μοντέλο μαθαίνει να κάνει προβλέψεις από τα δεδομένα. Κατά τη διάρκεια του βήματος αυτού, στο μοντέλο δίνεται το σύνολο δεδομένων εκπαίδευσης, το οποίο μπορεί να είναι επισημασμένα (labelled) ή και όχι. Η πρώτη περίπτωση ονομάζεται επιβλεπόμενη μάθηση (Supervised Learning), ενώ η δεύτερη μη επιβλεπόμενη μάθηση. Επιπλέον, το μοντέλο που θα χρησιμοποιηθεί, μπορεί να είναι απλά μηχανικής μάθησης (Machine Learning, ML) ή βαθιάς μάθησης (Deep Learning, DL). Τα μοντέλα βαθιάς μάθησης, όπως τα perceptron πολλαπλών επιπέδων (Multi Layer Perceptron, MLP), είναι σε θέση να μαθαίνουν από μεγαλύτερα και πιο περίπλοκα σύνολα δεδομένων. Η διαδικασία εκπαίδευσης των DL μοντέλων αποτελείται από πολλαπλές επαναλήψεις, όπου το μοντέλο λαμβάνει ως είσοδο τα δεδομένα εκπαίδευσης, κάνει προβλέψεις και στη συνέχεια υπολογίζονται τα σφάλματα τα οποία χρησιμοποιούνται για την ενημέρωση των βαρών και των πολώσεων (biases). Αντίθετα, οι παραδοσιακές μέθοδοι μηχανικής μάθησης, όπως η γραμμική παλινδρόμηση και τα δέντρα αποφάσεων, χρησιμοποιούνται συνήθως σε απλούστερες εργασίες με μικρότερα σύνολα δεδομένων.

6. Αξιολόγηση Μοντέλου

Στο στάδιο της αξιολόγησης γίνεται αξιολόγηση της απόδοσης του μοντέλου σε δεδομένα που δεν χρησιμοποιήθηκαν στη διαδικασία εκπαίδευσης. Οι μετρικές που μπορεί να χρησιμοποιηθούν εξαρτώνται από το μοντέλο και από το πρόβλημα. Οι πιο συνήθεις επιλογές για προβλήματα κατηγοριοποίησης (classification) είναι:

- Accuracy

- Precision
- Recall
- F1 score

Για προβλήματα παλινδρόμησης (Regression) γίνεται χρήση των:

- Μέσο Απόλυτο Σφάλμα (Mean Absolute Error, MAE)
- Μέσο Τετραγωνικό Σφάλμα (Mean Squared Error, MSE)
- Ρίζα του Μέσου Τετραγωνικού Σφάλματος (Root Mean Squared Error, RMSE)
- Μέσο Απόλυτο Ποσοστιαίο Σφάλμα (Mean Absolute Percentage Error, MAPE)

Τέλος, ενώ για την τελική αξιολόγηση του μοντέλου γίνεται χρήση του συνόλου τεστ, υπάρχει και η δυνατότητα προσαρμογής των υπερπαραμέτρων του μοντέλου (hyperparameter tuning) με χρήση του συνόλου επικύρωσης και επανάληψη της εκπαίδευσης του μοντέλου με διαφορετικές υπερπαραμέτρους κάθε φορά. Το παραπάνω γίνεται και με χρήση k-fold cross-validation, που αποφεύγει την ανάγκη ύπαρξης ξεχωριστού συνόλου δεδομένων επικύρωσης διαχωρίζοντας τα δεδομένα εκπαίδευσης σε k "αναδιπλώσεις", όπου κάθε αναδίπλωση χρησιμοποιείται ως σύνολο επικύρωσης μία φορά και ως σύνολο εκπαίδευσης k-1 φορές.

7. Εγκατάσταση (Deployment) Μοντέλου

Σε αυτό το στάδιο, η ευθύνη του μοντέλου μεταφέρεται από την ομάδα επιστήμης δεδομένων στις ομάδες των μηχανικών. Ενσωματώνουν το μοντέλο στην εφαρμογή και το καθιστούν διαθέσιμο προς χρήση. Οι ομάδες αυτές είναι υπεύθυνες για την τακτική παρακολούθηση των επιδόσεων του μοντέλου, και εάν αυτές χειροτερεύουν, μπορεί να χρειαστεί να επαναληφθεί εκ νέου όλη διαδικασία για να ενημερωθεί το μοντέλο ώστε να ληφθούν υπόψη νέες τάσεις και νέα δεδομένα. Γίνεται επίσης αναφορά τυχόν σφαλμάτων ή απροσδόκητων προβλέψεων του μοντέλου στην ομάδα επιστήμης δεδομένων, η οποία χρησιμοποιεί αυτές τις πληροφορίες για βελτίωση του μοντέλου σε περίπτωση επανάληψης του κύκλου.

Είναι συνεπώς ξεκάθαρη η πολυπλοκότητα του κύκλου ζωής μηχανικής μάθησης. Σε αυτό το σημείο θα πραγματοποιηθεί σύντομη ανάλυση της μεθόδου DevOps, καθώς και της μεθόδου MLOps που βασίζεται στην πρώτη, και προσπαθεί να επιλύσει το προαναφερθέν πρόβλημα:

2.1.2 Οι μέθοδοι DevOps και MLOps

Το DevOps, ή αλλιώς αναπτυξιακές λειτουργίες (developmental operations), αναφέρεται σε ένα σύνολο πρακτικών που συνδυάζουν τις διαδικασίες εργασίας των προγραμματιστών λογισμικού με εκείνες των επιχειρησιακών ομάδων για τη δημιουργία ενός κοινού συνόλου πρακτικών που λειτουργεί ως υβρίδιο των δύο ρόλων. Ως αποτέλεσμα, ο κύκλος ανάπτυξης του λογισμικού επιταχύνεται και εξασφαλίζεται η συνεχής παράδοση των προϊόντων λογισμικού. Έτσι, μειώνεται το συνολικό κόστος καθώς πέφτει το κόστος συντήρησης αφού αυξάνεται η αποτελεσματικότητα της ροής εργασίας κατά τη συντήρηση των εφαρμογών λογισμικού.

Οι ομάδες ανάπτυξης λογισμικού συνήθως υιοθετούν τη μεθοδολογία Agile, η οποία είναι μια επαναληπτική προσέγγιση του κύκλου ανάπτυξης ενός λογισμικού που εστιάζει στην ελαστικότητα και τη συνεχή συνεργασία και επικοινωνία των εμπλεκόμενων μελών (stakeholders). Συγκεκριμένα, οι επιχειρησιακές ομάδες είναι υπεύθυνες για την ανάπτυξη, τη συντήρηση και τη συνεχή αξιολόγηση της εφαρμογής είτε με τη μορφή σφαλμάτων είτε με την αξιολόγηση των τελικών χρηστών. Επιπλέον είναι υπεύθυνες για τη διαβίβαση αυτών των πληροφοριών στις ομάδες ανάπτυξης. Από εκεί και πέρα, η ομάδα ανάπτυξης εισέρχεται στη φάση συντήρησης της εφαρμογής, όπου σχεδιάζει, κατασκευάζει, τεστάρει και προωθεί την επόμενη ενημερωμένη έκδοση της εφαρμογής. Επιπλέον, η αυτοματοποίηση της διαδικασίας αξιολόγησης και ανάπτυξης επιτρέπει την τελική ενσωμάτωση των προς παράδοση προϊόντων λογισμικού

Το MLOps επιτρέπει να αυτοματοποιηθούν και να αξιοποιηθούν πλήρως τα μοντέλα μηχανικής μάθησης υψηλής απόδοσης με ευκολία. Μπορεί να θεωρηθεί ως το σημείο τομής μεταξύ της μηχανικής μάθησης και των πρακτικών DevOps που αναφέρθηκαν παραπάνω.

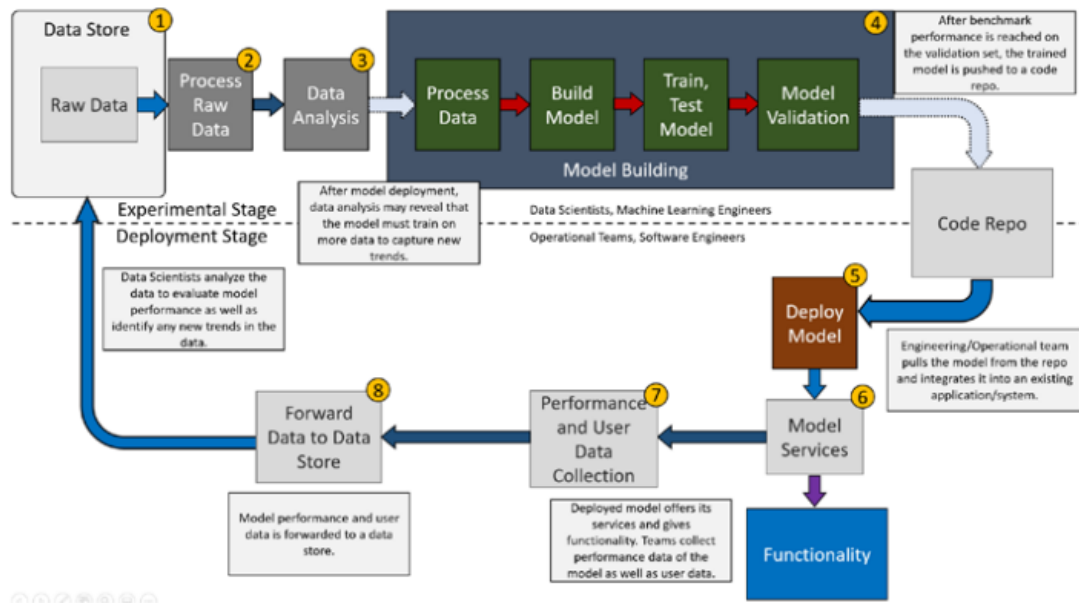
Το MLOps υιοθετεί τις αρχές του DevOps και τις εφαρμόζει στα μοντέλα μηχανικής μάθησης κατά αντιστοιχία με το λογισμικό, ενώνοντας τους κύκλους ανάπτυξης που ακολουθούν οι επιστήμονες δεδομένων και οι μηχανικοί μηχανικής μάθησης με αυτούς των επιχειρησιακών ομάδων, ώστε να συμβάλει στη συνεχή παράδοση μοντέλων μηχανικής μάθησης υψηλής απόδοσης. Η διαδικασία ανάπτυξης μοντέλων σε αυτό που ονομάζεται πειραματικό στάδιο, μπορεί να οδηγήσει σε εντυπωσιακές επιδόσεις και μπορεί να φανεί ως μία πολλά υποσχόμενη λύση. Ωστόσο, η πραγματικότητα είναι ότι τα περισσότερα μοντέλα απλώς δεν ξεπερνούν ποτέ αυτό το πειραματικό στάδιο, καθώς η ανάπτυξή τους είναι από μόνη της ένα αρκετά απαιτητικό εγχείρημα. Η διατήρηση των μοντέλων μετά την ανάπτυξή τους εξαντλεί επίσης πόρους, καθώς κάθε νέα ενημέρωση απαιτεί εκ νέου ενσωμάτωση τους στην εφαρμογή. Για τους λόγους αυτούς, τα περισσότερα μοντέλα απλά δεν ξεπερνούν ποτέ τη φάση του πρωτοτύπου. Μέχρι την εμφάνιση των MLOps, η ανάπτυξη λύσεων που δημιουργήθηκαν με τη χρήση της τελευταίας τεχνολογίας μηχανικής μάθησης αποτελούσε σημαντική πρόκληση για τις επιχειρήσεις λόγω του όγκου των πόρων που απαιτούνταν για την υλοποίησή τους. Αυτός είναι και ο λόγος για τον οποίο το MLOps είναι τόσο σημαντικό. Διευκολύνει αρκετά την ανάπτυξη και τη συντήρηση των λύσεων μηχανικής μάθησης αυτοματοποιώντας τα περισσότερα από τα δύσκολα μέρη για τους προγραμματιστές, επιταχύνοντας μαζί τις διαδικασίες ανάπτυξης και συντήρησης. Με μια πλήρως αυτοματοποιημένη ρύθμιση, οι ομάδες ανάπτυξης μπορούν να συμβαδίζουν με την τελευταία λέξη της τεχνολογίας μηχανικής μάθησης και να αναπτύσσουν γρήγορα νέα μοντέλα, βελτιώνοντας τις υπηρεσίες τους καθώς μπορούν να εστιάζουν πλέον την προσοχή τους στο να αναπτύξουν νεότερες, πιο υποσχόμενες αρχιτεκτονικές μοντέλων.

2.1.3 Οι ρυθμίσεις (Setups) του MLOps

Υπάρχουν τρεις διαφορετικές ρυθμίσεις που αντιπροσωπεύουν τα διάφορα στάδια της αυτοματοποίησης: χειροκίνητη εφαρμογή (manual implementation), συνεχή παράδοση μοντέλων (continuous model delivery) και συνεχή ολοκλήρωση/συνεχής παράδοση σωληνώσεων (continuous integration/continuous delivery of pipelines). Παρακάτω θα περιγραφούν σε ανάλυση υψηλού επιπέδου και συνοπτικά οι παραπάνω ρυθμίσεις. Για αναλυτικότερες και πιο κάθετες επεξηγήσεις του κάθε βήματος παραπέμπουμε τον αναγνώστη στο [3]

Η χειροκίνητη εφαρμογή αναφέρεται σε μια ρύθμιση όπου δεν εφαρμόζονται οι αρχές MLOps και τα πάντα υλοποιούνται με το χέρι. Τα βήματα για τη δημιουργία

ενός μοντέλου μηχανικής μάθησης εκτελούνται όλα χειροκίνητα. Οι ομάδες μηχανικών λογισμικού πρέπει να ενσωματώσουν χειροκίνητα τα μοντέλα στην εφαρμογή και οι επιχειρησιακές ομάδες πρέπει να βοηθήσουν να διασφαλιστεί η διατήρηση όλων των λειτουργιών μαζί με τη συλλογή δεδομένων και μετρήσεων απόδοσης του μοντέλου. Η διαδικασία αυτή φαίνεται σχηματικά στην Εικόνα 2.1

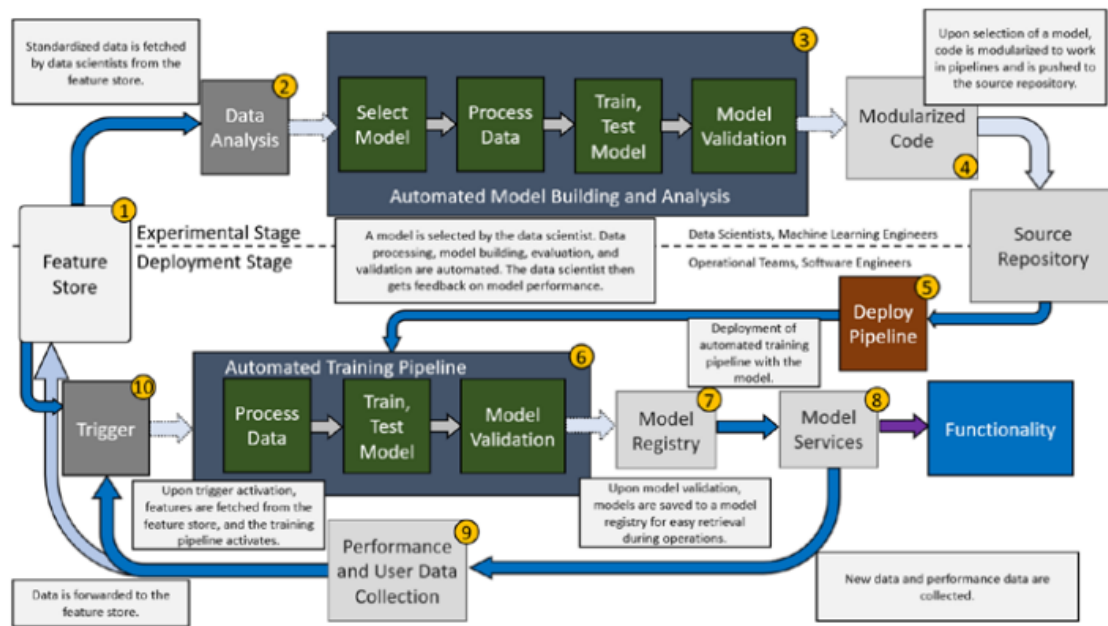


Σχήμα 2.1: Πιθανή ρύθμιση ανάπτυξης ενός μοντέλου μηχανικής μάθησης χωρίς τις αρχές MLOps. Τα βέλη με το διακεκομμένο περίγραμμα σημαίνουν ότι η μετάβαση στο επόμενο βήμα εξαρτάται από μια συνθήκη στο τρέχον βήμα. Για παράδειγμα, στο βήμα της επικύρωσης του μοντέλου, οι μηχανικοί μηχανικής μάθησης πρέπει να διασφαλίσουν ότι το μοντέλο πληροί κάποιες ελάχιστες προκαθορισμένες απαιτήσεις όσον αφορά την απόδοση πριν προωθήσουν το μοντέλο στο αποθετήριο τους.¹

Η συνεχής παράδοση μοντέλων είναι μια καλή μέση λύση μεταξύ μιας χειροκίνητης ρύθμισης και μιας πλήρως αυτοματοποιημένης. Εδώ χρησιμοποιούνται οι σωληνώσεις, οι οποίες επιτρέπουν την αυτοματοποίηση της πλευράς σχετικά με τη μηχανική μάθηση της διαδικασίας. Υπενθυμίζεται ότι η σωλήνωση είναι μια υποδομή που περιέχει μια ακολουθία στοιχείων που μετασχηματίζουν και χειρίζονται τις πληροφορίες καθώς περνούν μέσα από αυτήν.

Το κύριο χαρακτηριστικό αυτού του τύπου ρυθμίσεων είναι ότι το μοντέλο που αναπτύσσεται έχει εγκατεστημένες σωληνώσεις για τη συνεχή εκπαίδευσή του σε νέα δεδομένα, ακόμη και μετά το στάδιο της ανάπτυξης. Αναδύεται επίσης η αυτοματοποίηση του πειραματικού σταδίου ή του σταδίου ανάπτυξης του μοντέλου, μαζί με την ανάπτυξη του κώδικα σε επίπεδο στοιχείων, ώστε να είναι δυνατή η περαιτέρω αυτοματοποίηση στα επόμενα βήματα. Σε αυτή τη ρύθμιση, η συνεχής παράδοση αναφέρεται στην ταχεία ανάπτυξη και εφαρμογή νέων μοντέλων μηχανικής μάθησης. Μέσω αυτής της αυτοματοποίησης, τα μοντέλα μπορούν πλέον να δημιουργηθούν ή να ενημερωθούν με πολύ ταχύτερο ρυθμό. Η προαναφερθείσα διαδικασία μπορεί να φανεί σχηματικά στην Εικόνα 2.2

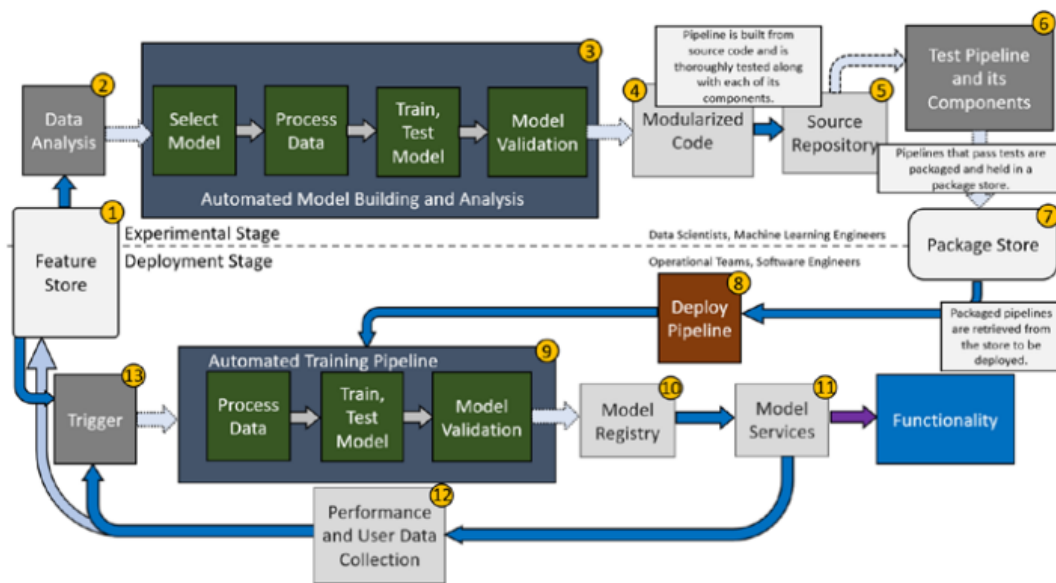
¹Πηγή: [3]



Σχήμα 2.2: Πιθανή ρύθμιση ανάπτυξης ενός μοντέλου μηχανικής μάθησης με αυτοματοποίηση μέσω των σωληνώσεων.²

Η συνεχής ολοκλήρωση/συνεχής παράδοση των σωληνώσεων αναφέρεται σε μια ρύθμιση όπου οι σωληνώσεις στο πειραματικό στάδιο δοκιμάζονται διεξοδικά με μια αυτοματοποιημένη διαδικασία για να διασφαλιστεί ότι όλα τα στοιχεία λειτουργούν όπως προβλέπεται. Από εκεί και πέρα, οι σωληνώσεις συσκευάζονται και αναπτύσσονται, όπου οι ομάδες ανάπτυξης λογισμικού αναπτύσσουν η σωλήνωση σε ένα δοκιμαστικό περιβάλλον, διεκπεραιώνουν πρόσθετες δοκιμές για να διασφαλίσουν τόσο τη συμβατότητα όσο και τη λειτουργικότητα και, στη συνέχεια, την προσθέτουν στο περιβάλλον παραγωγής. Σε αυτή τη ρύθμιση, οι σωληνώσεις μπορούν πλέον να δημιουργηθούν και να αναπτυχθούν με γρήγορο ρυθμό, επιτρέποντας στις ομάδες να δημιουργούν συνεχώς νέες οι οποίες βασίζονται στις πιο πρόσφατες αρχιτεκτονικές μηχανικής μάθησης χωρίς κανένα από τα εμπόδια πόρων που σχετίζονται με τις χειροκίνητες δοκιμές και την ενσωμάτωση. Η προαναφερθείσα διαδικασία μπορεί να φανεί σχηματικά στην Εικόνα 2.3

²Πηγή: [3]



Σχήμα 2.3: Πιθανή ρύθμιση ανάπτυξης μοντέλου μηχανικής μάθησης με αυτοματοποίηση μέσω συνεχούς παράδοσης της ρύθμισης σωληνώσεων.³

Το κύριο πρόβλημα της δεύτερης ρύθμισης που διορθώνεται με την παρούσα είναι αυτό της ανάπτυξης σωληνώσεων. Προηγουμένως, οι σωληνώσεις έπρεπε να τεσταριστούν χειροκίνητα από τις ομάδες μηχανικής μάθησης και τις επιχειρησιακές ομάδες για να διασφαλιστεί ότι τα συστατικά τους λειτουργούσαν και ότι η σωλήνωση ήταν συμβατή με την εφαρμογή. Αντίθετα, στη παρούσα ρύθμιση, οι δοκιμές αυτοματοποιούνται, επιτρέποντας στις ομάδες να δημιουργούν και να αναπτύσσουν σωληνώσεις πολύ πιο εύκολα από ό,τι πριν. Το μεγαλύτερο πλεονέκτημα είναι ότι οι επιχειρήσεις μπορούν πλέον να συμβαδίζουν με τις σημαντικές αλλαγές στα δεδομένα που απαιτούν τη δημιουργία νέων μοντέλων και νέων σωληνώσεων και μπορούν επίσης να αξιοποιήσουν τις τελευταίες τάσεις και αρχιτεκτονικές μηχανικής μάθησης, όλα χάρη στην ταχεία δημιουργία και ανάπτυξη των σωληνώσεων σε συνδυασμό με τη συνεχή παράδοση υπηρεσιών μοντέλων από την προηγούμενη ρύθμιση.

Στη παρούσα διπλωματική λόγω των πολλών πλεονεκτημάτων που παρουσιάζει υιοθετήσαμε και αναπτύξαμε τη λογική της συνεχούς παράδοσης μοντέλων, η οποία θα αναλυθεί διεξοδικά στην εφαρμοσμένη έκφρασή της στα αντίστοιχα κεφάλαια.

Το εργαλείο που επιλέχθηκε για να υλοποιηθούν οι προαναφερθείσες αρχές είναι το MLflow [4]. Το εργαλείο αυτό προσφέρεται για έλεγχο όλου του κύκλου ζωής μηχανικής μάθησης και για τον έλεγχο της σωλήνωσης μηχανικής μάθησης (ML pipeline) με ελάχιστες αλλαγές στον κώδικα.

2.2 Συντομη βιβλιογραφική ανάλυση για την βραχυπρόθεσμη πρόβλεψη ζήτησης ηλεκτρικής ενέργειας και την χρήση πολλαπλών χρονοσειρών

Το πρόβλημα της βραχυπρόθεσμης πρόβλεψης ηλεκτρικού φορτίου έχει δυο βασικούς τρόπους αντιμετώπισης. Ο πρώτος αφορά στατιστικά μοντέλα όπως το ARIMA

³Πηγή: [3]

[5] ή το sARIMA [6]. Ο δεύτερος είναι με την χρήση μηχανικής ή βαθιάς μάθησης. Πολλά μοντέλα βαθιάς μάθησης έχουν χρησιμοποιηθεί ανά τα χρόνια όπως MLP [7] και LSTM [8].

Η χρήση πολλαπλών χρονοσειρών για την εκπαίδευση ενός μοντέλου πρόβλεψης χρονοσειρών (ονομάζεται επίσης global προσέγγιση [2]) αποτελεί μια μέθοδο πρόβλεψης η οποία εμφανίζεται όλο και περισσότερο στην βιβλιογραφία για πρόβλεψη χρονοσειρών. Πιο συγκεκριμένα, οι μέθοδοι που κέρδισαν στον διαγωνισμό M4 [9] και M5 [10] χρησιμοποίησαν πολλαπλές χρονοσειρές. Στην πρώτη περίπτωση εφαρμόστηκε συνδυασμός εκθετικής εξομάλυνσης με Αναδρομικά Νευρωνικά Δίκτυα (Recurrent Neural Network, RNN), ενώ στην δεύτερη εκπαιδεύτηκαν μοντέλα με gradient boosting. Επιπλέον, στο [11] χρησιμοποιήθηκε RNN το οποίο εκπαιδεύτηκε σε πολλαπλές χρονοσειρές του Uber και εφαρμόστηκε χωρίς επανεκπαίδευση στο σύνολο δεδομένων του M3 διαγωνισμού [12]. Ακόμα, στο [13] το μοντέλο NBEATS δοκιμάζεται στο σύνολο M3 και M4.

Η χρήση πολλαπλών χρονοσειρών χρησιμοποιείται επίσης και στην πρόβλεψη φορτίου. Στο [14] χρησιμοποιήθηκε RNN με στόχο την βραχυπρόθεσμη πρόβλεψη ζήτησης ενέργειας στο νοικοκυριό με διαχωρισμό των καταναλωτών σε τυχαίες ομάδες. Οι χρονοσειρές κάθε ομάδας προβλέφθηκαν ξεχωριστά. Επίσης, στο [15], πραγματοποιήθηκε χρήση μοντέλου CNN (Convolutional Neural Network) για την πρόβλεψη κατανάλωσης ενέργειας από καταναλωτές. Δυο στρατηγικές δοκιμάστηκαν, μια με ένα μοντέλο για όλους τους καταναλωτές, και μια με ξεχωριστό μοντέλο για κάθε καταναλωτή. Το αποτέλεσμα έδειξε ότι το μοντέλο που χρησιμοποίησε πολλαπλές χρονοσειρές είχε καλύτερα αποτελέσματα. Τέλος, στο [16], το μοντέλο NBEATS χρησιμοποιείται μαζί με συσταδοποίηση (clustering) για την πρόβλεψη φορτίου για δίκτυα διανομής ενέργειας. Πολλαπλές χρονοσειρές που αναφέρονται σε καταναλωτές δίνονται στο ίδιο μοντέλο.

2.3 Μοντέλα που θα χρησιμοποιηθούν στην παρούσα εργασία

Παρακάτω ακολουθεί σύντομη παρουσίαση των μοντέλων που θα χρησιμοποιηθούν στη παρούσα εργασία.

2.3.1 LightGBM

Το μοντέλο Light Gradient Boosting Machine (LightGBM) [17] [18] αποτελεί ένα μοντέλο μηχανικής μάθησης το οποίο βασίζεται στα δέντρα αποφάσεων και στην τεχνική του gradient boosting.

Η δεύτερη είναι μια μορφή ensembling, η οποία σκοπό έχει να βελτιώσει την απόδοση μοντέλων (που συνήθως βασίζονται σε δέντρα) που μόνα τους θα είχαν κακές επιδόσεις. Πιο συγκεκριμένα, η μέθοδος αυτή εκπαιδεύει κάθε μοντέλο σε μία συνάρτηση της αρνητικής παραγώγου της συνάρτησης σφάλματος του προηγούμενου. Τελικά λαμβάνεται η έξοδος του τελευταίου μοντέλου ως συνολική έξοδος.

Η συνάρτηση σφάλματος που χρησιμοποιείται στον αλγόριθμο του LightGBM εξαρτάται τόσο από το πόσο καλά έχει προβλέψει το κάθε δέντρο τα δεδομένα εκπαίδευσης, όσο και από το πόσο περίπλοκο είναι το δέντρο. Έτσι, δέντρα τα οποία είναι μεγάλα σε μέγεθος ή έχουν μεγάλα βάρη αποφεύγονται, και το τελικό δέντρο πραγματοποιεί πιο γρήγορες προβλέψεις

Εκτός από τα προαναφερθέντα, στο μοντέλο LightGBM εφαρμόζονται και ορισμένες βελτιστοποιήσεις με σκοπό την αύξηση της ταχύτητας πρόβλεψης. Αρχικά, το δέντρο δεν δοκιμάζει όλες τις τιμές ενός χαρακτηριστικού για να βρει τον καλύτερο διαχωρισμό, αλλά οι παραπάνω ομαδοποιούνται σε ιστογράμματα. Έτσι, οι τιμές που χρειάζεται να ελεγχθούν για κάθε κόμβο του δέντρου είναι πολύ λιγότερες.

Επιπλέον, πραγματοποιείται ένωση των χαρακτηριστικών που είναι αμοιβαίως αποκλειόμενα, δεν λαμβάνουν δηλαδή ποτέ μηδενικές τιμές μαζί. Η τεχνική αυτή ονομάζεται Exclusive Feature Bundling technique (EFB), και μειώνει των αριθμών των χαρακτηριστικών του συνόλου δεδομένων, και επομένως και τον χρόνο εκπαίδευσης.

Τέλος, κάθε δέντρο που προστίθεται δεν εκπαιδεύεται σε όλα τα χαρακτηριστικά που υπάρχουν στο σύνολο δεδομένων. Αντιθέτως, τα χαρακτηριστικά χωρίζονται σε 2 σύνολα, αυτό του οποίου τα χαρακτηριστικά έχουν μεγάλη παράγωγο του σφάλματος, και αυτό με μικρή παράγωγο του σφάλματος. Το επόμενο δέντρο έπειτα εκπαιδεύεται σε όλα τα χαρακτηριστικά του πρώτου συνόλου, και σε ένα δείγμα μόνο των στοιχείων του δεύτερου. Η λογική πίσω από αυτή την στρατηγική είναι ότι τα χαρακτηριστικά με την μεγαλύτερη παράγωγο του σφάλματος, είναι αυτά που έχει μάθει λιγότερο το προηγούμενο δέντρο. Αυτό σημαίνει ότι είναι και αυτά στα οποία χρειάζεται να βελτιωθεί η απόδοση του μοντέλου. Η μέθοδος αυτή ονομάζεται Gradient-based One Side Sampling technique (GOSS).

2.3.2 NBEATS

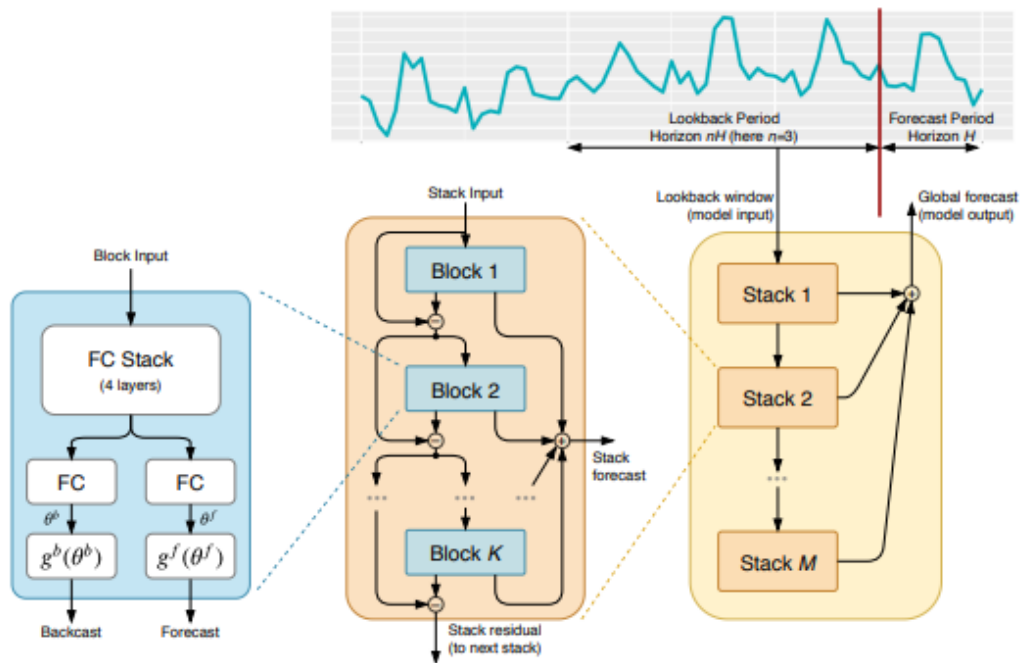
Το μοντέλο Neural Basis Expansion Analysis for Time Series forecasting (NBeats [13]) αποτελεί ένα μοντέλο βαθιάς μάθησης, το οποίο βασίζεται στα perceptron πολλαπλών επιπέδων (Multi Layer Perceptron, MLP), και στα υπολλειματικά δίκτυα (residual networks).

Πιο συγκεκριμένα, το θεμελιώδες δομικό συστατικό του NBEATS είναι τα μπλοκ (blocks). Αυτά περιέχουν έναν αριθμό από πλήρως συνδεδεμένα επίπεδα (Fully Connected Layers), και έχουν 2 εξόδους οι οποίες ονομάζονται backcast και forecast. Η πρώτη αποτελεί πρόβλεψη των δεδομένων εισόδου "προς τα πίσω", ενώ η δεύτερη πρόβλεψη των δεδομένων μετά την είσοδο.

Πολλά μπλοκ φτιάχνουν μια στοίβα (stack). Τα μπλοκ ενώνονται σε σειρά, και η είσοδος του ενός είναι η διαφορά της εισόδου του προηγούμενου μπλοκ και της πρόβλεψης των δεδομένων εισόδου (backcast) που θεωρείται το κομμάτι της εισόδου που έχει προβλεφθεί επιτυχώς. Η έξοδος της στοίβας είναι το άθροισμα των προβλέψεων μετά την είσοδο (forecast) όλων των μπλοκ που την αποτελούν. Το παραπάνω ονομάζεται διπλή υπολλειματική στοίβα (Double Residual Stacking).

Τέλος, πολλές στοίβες μπορούν να τοποθετηθούν σε σειρά, και να προκύψει το τελικό αποτέλεσμα του μοντέλου αθροίζοντας όλες τις προβλεψεις των στοιβών.

⁴Πηγή: [13]



Σχήμα 2.4: Η αρχιτεκτονική του NBEATS. ⁴

Κεφάλαιο 3

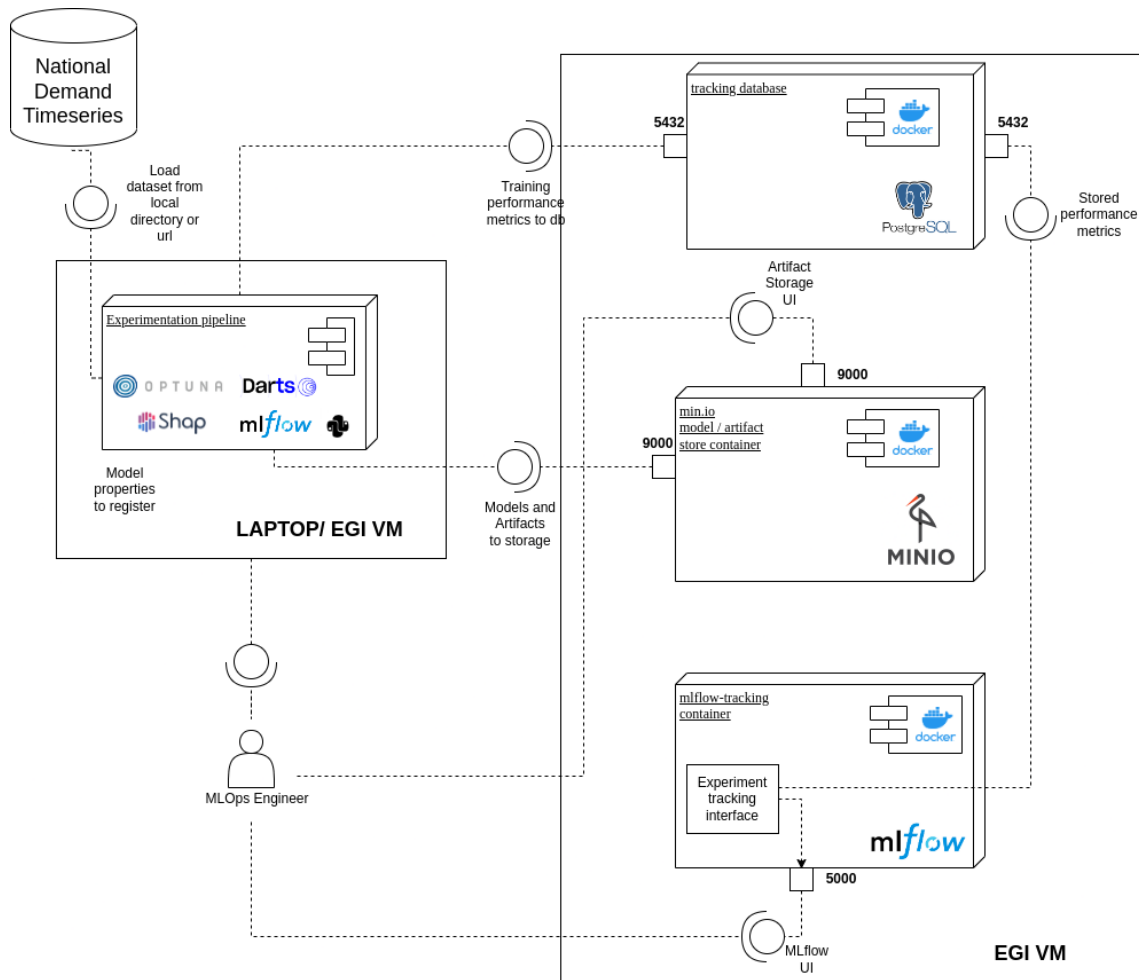
Ανάπτυξη της Εφαρμογής

Σε αυτό το σημείο θα παρουσιαστεί η εφαρμογή αναλυτικά, και θα δοθεί έμφαση στις λειτουργίες που προστέθηκαν κατά την εκπόνηση της παρούσας εργασίας. Επίσης, παρέχεται το github της εφαρμογής ¹

3.1 Σύντομη παρουσίαση της εφαρμογής

Όπως προαναφέρθηκε, σκοπός της παρούσας εφαρμογής είναι η αυτοματοποίηση του κύκλου ζωής μηχανικής μάθησης (ML Lifecycle) από την αρχική επεξεργασία δεδομένων μέχρι και την αξιολόγηση του μοντέλου για προβλήματα βραχυπρόθεσμης πρόβλεψης ζήτησης ηλεκτρικής ενέργειας (STLF), αλλά και για χρονοσειρές γενικότερα. Η αυτοματοποίηση αυτή της σωλήνωσης μηχανικής μάθησης (ML pipeline), καθώς και η εύκολη πρόσβαση στα αποτελέσματα των διεργασιών, επιτεύχθη με χρήση του MLflow [4]. Επίσης, η εφαρμογή προσφέρει την δυνατότητα εκπαίδευσης μοντέλων μηχανικής μάθησης (ML) και βαθιάς μάθησης (DL) (NBEATS, Transformer, NHiTS, TCN, RNN, BlockRNN, TFT, LightGBM, RandomForest και Naive), για προβλήματα τόσο απλών, όσο και πολυμεταβλητών (multivariate) ή/ και πολλαπλών (multiple) χρονοσειρών. Ακόμα, παρέχεται και υποστήριξη για συμμεταβλητές (covariates). Ο αρχικός σκελετός της παρούσας εφαρμογής βασίστηκε σε προϋπάρχοντα κώδικα που ήταν διαθέσιμος από το εργαστήριο. Η αρχιτεκτονική της εφαρμογής φαίνεται στο παρακάτω διάγραμμα:

¹https://github.com/pelekhs/energy-forecasting/tree/master/PILOT1_RDN



Σχήμα 3.1: Η αρχιτεκτονική της εφαρμογής

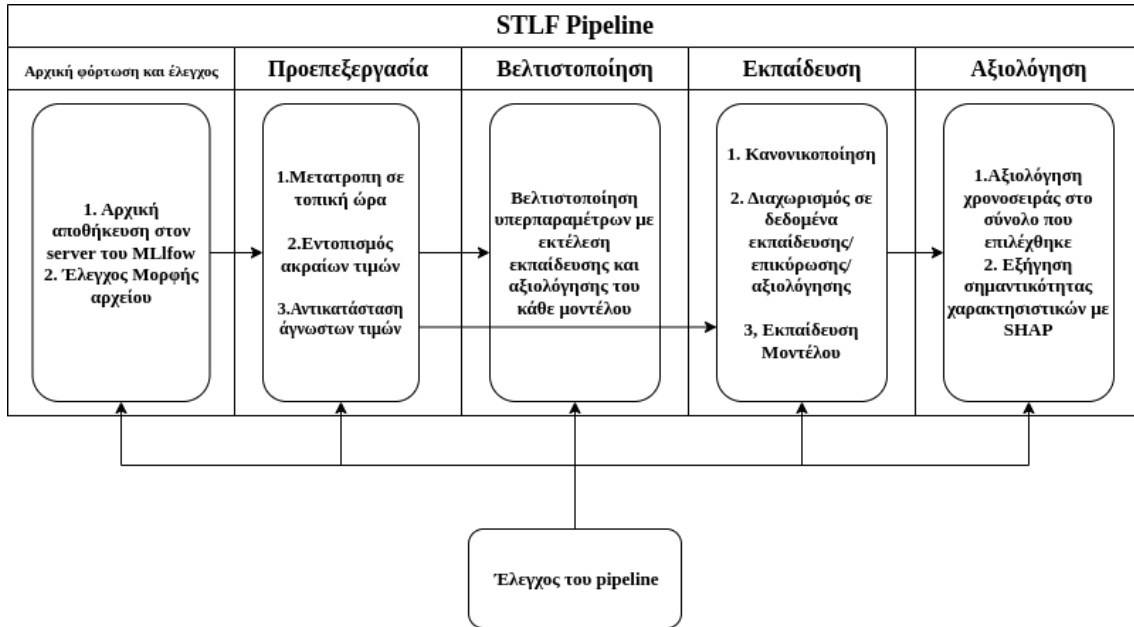
Στα δεξιά φαίνεται ο διακομιστής MLflow στον οποίο αποθηκεύονται όλα τα αποτελέσματα και είναι υπεύθυνος για την διεπαφή χρήστη (UI) τόσο του MLflow, όσο και του Minio. Στην πρώτη διεπαφή φαίνονται τα αποτελέσματα των τρεξιμάτων, ενώ στην δεύτερη δίνεται πρόσβαση στον χρήστη στα αρχεία που είναι αποθηκευμένα στον διακομιστή. Στην παρούσα διπλωματική προστέθηκαν λειτουργικότητες στο αριστερό κομμάτι της εφαρμογής που είναι η σωλήνωση (pipeline) της εφαρμογής. Όλος ο κώδικας είναι γραμμένος σε python[19], ενώ για τα μοντέλα έγινε χρήση της βιβλιοθήκης darts[20]. Ακόμα, σημαντικές βιβλιοθήκες είναι η optuna[21] και shap[1], που χρησιμοποιήθηκαν για την βελτιστοποίηση υπερπαραμέτρων και την εξήγηση σημαντικότητας των αποτελεσμάτων αντίστοιχα.

Η σωλήνωση μηχανικής μάθησης χωρίστηκε σε 6 στάδια (stages) τα οποία καλούνται με την βοήθεια του MLflow:

- Στάδιο ελέγχου της σωλήνωσης (αρχείο experimentation_pipeline.py)
- Αρχική φόρτωση και έλεγχος των δεδομένων (αρχείο load_raw_data.py)
- Επεξεργασία δεδομένων (αρχείο etl.py)
- Εκπαίδευση μοντέλου (αρχείο training.py)
- Αξιολόγηση μοντέλου (αρχείο evaluation.py)

- Βελτιστοποίηση υπερπαραμέτρων (hyperparameter tuning) με χρήση της βιβλιοθήκης Optuna [22] [24] (αρχείο optuna.py)

Τα παραπάνω στάδια περιγράφονται συνοπτικά στην παρακάτω εικόνα:



Σχήμα 3.2: Τα στάδια της σωλήνωσης

Παρακάτω ακολουθεί αναλυτική περιγραφή του καθενός από τα προαναφερθέντα στάδια

3.2 Στάδιο Ελέγχου της σωλήνωσης

Αυτό το στάδιο είναι υπεύθυνο για τον έλεγχο της ροής της σωλήνωσης μηχανικής μάθησης, και για την κλήση των υπόλοιπων σταδίων με τις κατάλληλες παραμέτρους. Συγκεκριμένα, αφού ληφθούν οι αρχικές παράμετροι από το MLflow, πραγματοποιείται αρχικά κλήση του σταδίου αρχικής φόρτωσης και ελέγχου των δεδομένων. Αν η εκτέλεση αυτού είναι επιτυχής, γίνεται κλήση του σταδίου επεξεργασίας δεδομένων. Στο σημείο αυτό γίνεται έλεγχος του πεδίου `opt_test`, που καθορίζει το αν επιθυμούμε να κάνουμε βελτιστοποίηση υπερπαραμέτρων. Εάν αυτό είναι αληθές, ξεκινά η εκτέλεση του σταδίου βελτιστοποίησης υπερπαραμέτρων, και όταν αυτό τελειώσει σταματά και η εκτέλεση της εφαρμογής. Στην αντίθετη περίπτωση, εκτελούνται διαδοχικά τα στάδια εκπαίδευσης και αξιολόγησης του μοντέλου με τις υπερπαραμέτρους που δίνονται και σταματά η εκτέλεση της εφαρμογής.

3.3 Αρχική φόρτωση και έλεγχος των δεδομένων

Αυτό το στάδιο ξεκινά με την αρχική ανάγνωση των αρχείων δεδομένων. Τα αρχεία αυτά ενδεχομένως να βρίσκονται αποθηκευμένα τοπικά στον υπολογιστή που γίνεται η εκτέλεση της εφαρμογής, ή στον διακομιστή του MLflow. Πρέπει να είναι σε μορφή csv. Σε κάθε περίπτωση γίνεται κατέβασμα των αρχείων τοπικά στον υπολογιστή που εκτελείται η εφαρμογή. Έπειτα, πραγματοποιείται ανάγνωση του αρχείου csv από

την βιβλιοθήκη pandas [23] και γίνεται έλεγχος της εγκυρότητάς του. Η μορφή που απαιτείται να έχει το εν λόγω αρχείο εξαρτάται από την μορφή του προβλήματος που εξετάζεται. Συγκεκριμένα, αν το πρόβλημα αφορά απλές χρονοσειρές, η μορφή του αρχείου πρέπει να είναι σαν τον πίνακα παρακάτω:

Πίνακας 3.1: Μορφή αρχείου csv απλών χρονοσειρών

Date	Load
2015-04-09 00:00:00	7893
2015-04-09 00:00:00 + resolution	8023
2015-04-09 00:00:00 + 2 × resolution	8572
...	...

Εδώ, το Date είναι η στήλη για τις ημερομηνίες και το Load είναι η στήλη του φορτίου/τιμής της απλής χρονοσειράς. Οι έλεγχοι που πραγματοποιούνται σε αυτή την περίπτωση είναι οι εξής:

- Αν το csv έχει ακριβώς 2 στήλες με ονόματα Date, Load.
- Αν οι ημερομηνίες της στήλης Date βρίσκονται σε χρονολογική σειρά. Αν αυτό δεν συμβαίνει ενδεχομένως ο χρήστης έχει θέσει το πεδίο day_first (το οποίο δείχνει στην pandas το αν πρέπει να διαβαστούν οι ημερομηνίες θεωρώντας ότι πρώτα εμφανίζεται η ημέρα και μετά ο μήνας ή το αντίστροφο) λανθασμένα

Κάθε csv που δεν ικανοποιεί τους παραπάνω ελέγχους προκαλεί και την ανάλογη εξαίρεση.

Στην περίπτωση που το πρόβλημα αφορά πολλαπλές ή/και πολυμεταβλητές χρονοσειρές, η μορφή του αρχείου πρέπει να είναι σαν τον πίνακα παρακάτω:

Πίνακας 3.2: Μορφή αρχείου csv πολλαπλών η/και πολυμεταβλητών χρονοσειρών

Index	Day	ID	Source	Source Code	Timeseries ID	00:00:00	...
0	2015-04-09	0	Portugal	PT	0	5248	...
1	2015-04-09	1	Spain	ES	1	25497	...
...

Εδώ Index είναι ο αύξοντας αριθμός της γραμμής, ID το μοναδικό αναγνωριστικό της συγκεκριμένης συνιστώσας, Source η πηγή από την οποία ελήφθη αυτή η συνιστώσα (στην εφαρμογή που θα αναλυθεί εδώ μια χώρα) και Source Code ο κωδικός αυτής της πηγής (εδώ απλά ο κωδικός της χώρας). Οι στήλες με τις ώρες περιέχουν φορτίο/τιμή της κάθε χρονοσειράς. Οι έλεγχοι που πραγματοποιούνται σε αυτή την περίπτωση είναι οι εξής:

- Αν υπάρχουν στο csv οι στήλες Day, ID, και οι ώρες 00:00:00, 00:00:00 + resolution, 00:00:00 + 2 × resolution, ..., 24:00:00 - resolution.
- Αν δεν υπάρχει οποιαδήποτε από τις στήλες Timeseries ID, Source Code, Source, αυτή τίθεται ίση με την υποχρεωτική στήλη ID
- Αν οι ημέρες που αντιστοιχούν σε κάθε ID βρίσκονται σε χρονολογική σειρά

- Αν όλες οι χρονοσειρές που υπάρχουν στο αρχείο έχουν τον ίδιο αριθμό συνιστωσών (components)

Όπως και προηγουμένως, κάθε csv που δεν ικανοποιεί τους παραπάνω ελέγχους προκαλεί και την ανάλογη εξαίρεση.

Μετά την ολοκλήρωση των παραπάνω ελέγχων, το csv ανεβαίνει στον διακομιστή του MLflow, και η εκτέλεση της διαδικασίας σωλήνωσης προχωράει στο επόμενο στάδιο.

3.4 Επεξεργασία δεδομένων

Σε αυτό το στάδιο πραγματοποιείται η αρχική επεξεργασία των χρονοσειρών από την εφαρμογή. Πιο συγκεκριμένα, αφού γίνουν διαθέσιμες οι χρονοσειρές τοπικά ως pandas dataframes, γίνεται ενδεχομένως μετατροπή τους σε τοπική ώρα της χώρας στην οποία ανήκουν. Αυτό συμβαίνει αν το πεδίο `convert_to_local_tz` είναι αληθές. Έπειτα, αφού περιοριστούν οι χρονοσειρές στις επιθυμητές χρονιές, διαγραφούν οι επαναλαμβανόμενες τιμές, και μετατραπεί η ανάλυση των χρονοσειρών στην επιθυμητή, γίνεται εντοπισμός ακραίων τιμών και αντικατάσταση άγνωστων τιμών. Οι διαδικασίες αυτές θα περιγραφούν αναλυτικότερα στις ακριβώς επόμενες υποενότητες. Μετά από τα προαναφερθέντα, οι επεξεργασμένες χρονοσειρές μετατρέπονται εκ νέου τοπικά σε csv, και ανεβαίνουν στον διακομιστή του MLflow.

Πρέπει επίσης να αναφερθεί ότι τα αποτελέσματα του εντοπισμού ακραίων τιμών και της αντικατάστασης άγνωστων τιμών αποθηκεύονται ως artifacts στους φακέλους `outlier_detection_results` και `imputation_results` αντίστοιχα (στον φάκελο `features` βρίσκεται η τελική χρονοσειρά)

Πιο συγκεκριμένα, στον φάκελο `outlier_detection_results` αποθηκεύονται csvs με τις τιμές που αφαιρέθηκαν από την κάθε χρονοσειρά, καθώς και εικόνες που δείχνουν την χρονοσειρά μετά την αφαίρεση των ακραίων τιμών, και με τις τιμές αυτές επισημασμένες. Στον φάκελο `imputation_results` αποθηκεύονται csvs με τις τιμές που αντικαταστάθηκαν, καθώς και εικόνες με τις χρονοσειρές μετά την αντικατάσταση.

3.4.1 Εντοπισμός ακραίων τιμών (Outlier detection)

Σε πολλές χρονοσειρές υπάρχουν μηδενικές τιμές, και επίσης πολλές άλλες ακραίες τιμές (δηλαδή τιμές πολύ μεγαλύτερες ή πολύ μικρότερες από τις γειτονικές τους) οι οποίες πιθανόν οφείλονται σε λάθος και θα προκαλούσαν προβλήματα στην εκπαίδευση των μοντέλων, και στην κανονικοποίηση των δεδομένων. Για αυτό δίνεται η επιλογή χρήσης της συνάρτησης `remove_outliers` η οποία αντικαθιστά τις προαναφερθείσες τιμές με άγνωστες τιμές (NaN). Η συνάρτηση αυτή καλείται στην περίπτωση που το πεδίο `rmv_outliers` είναι αληθές.

Πιο συγκεκριμένα, αφού υπολογιστεί ο μέσος όρος και η τυπική απόκλιση κάθε μήνα στην χρονοσειρά, αφαιρούνται όσες τιμές βρίσκονται περισσότερο από `std_dev` τυπικές αποκλίσεις μακριά από τον μέσο όρο αυτόν. Ως προεπιλεγμένη τιμή του `std_dev` επιλέχθηκε το 4.5 έτσι ώστε να αφαιρούνται μόνο οι τιμές που βρίσκονται πολλές τυπικές αποκλίσεις μακριά από τον μέσο όρο, οι οποίες είναι το πιθανότερο να είναι λανθασμένες, και να κρατούνται στην χρονοσειρά οι τιμές που βρίσκονται λιγότερες τυπικές αποκλίσεις μακριά από τον μέσο όρο, οι οποίες είναι πιθανότερο να είναι μακριά από τον μέσο όρο απλά λόγω κάποιου εξωτερικού γεγονότος (πχ καιρού στις χρονοσειρές της παρούσας εργασίας).

Πρέπει βέβαια να σημειωθεί ότι η χρονοσειρά πάνω στην οποία εφαρμόζεται η εν λόγω συνάρτηση χρειάζεται να εμφανίζει μηνιαία εποχικότητα. Διαφορετικά, η συμπεριφορά κάθε μήνα θα είναι διαφορετική, και δεν θα προκύψουν έγκυρα αποτελέσματα για τις ακραίες τιμές.

3.4.2 Αντικατάσταση άγνωστων τιμών (imputation)

Πολλές χρονοσειρές περιέχουν άγνωστες τιμές. Αυτές θα πρέπει να αντικατασταθούν με κάποια αριθμητική τιμή, καθώς διαφορετικά δεν θα μπορεί να πραγματοποιηθεί η εκπαίδευση των μοντέλων.

Ένας απλός τρόπος για να συμβεί αυτό θα ήταν η γραμμική παρεμβολή (linear interpolation), που αντικαθιστά τις διαδοχικές άγνωστες τιμές ενός διαστήματος με τιμές που σχηματίζουν ευθεία η οποία έχει άκρα στις 2 κοντινότερες γνωστές τιμές. Η μέθοδος αυτή υποστηρίζεται και από το `darts`, που χρησιμοποιείται από την παρούσα εφαρμογή [24]. Αυτή η μέθοδος, αν και γρήγορη, δεν λειτουργεί καλά με χρονοσειρές που εμφανίζουν εποχικότητα. Σε αυτές τις περιπτώσεις, αν το διάστημα συνεχόμενων άγνωστων τιμών είναι αρκετά μεγάλο, η εποχικότητα της χρονοσειράς χάνεται στο διάστημα αυτό. Εναλλακτικά, θα μπορούσε να γίνει χρήση ιστορικών δεδομένων. Τότε ωστόσο, θα χανόταν οποιαδήποτε τοπική διαφοροποίηση (αύξηση ή μείωση σε σχέση με τις ιστορικές τιμές αντίστοιχων περιόδων) της χρονοσειράς που θα συνέβαινε σε διάστημα κοντά στις άγνωστες τιμές. Αυτή η διαφοροποίηση λαμβάνεται υπόψιν από την γραμμική παλινδρόμηση.

Είναι επομένως ξεκάθαρα τα πλεονεκτήματα και τα μειονεκτήματα αμφοτέρων των προαναφερθέντων μεθόδων. Έτσι, η υλοποίηση της συνάρτησης αντικατάστασης αγνώστων τιμών με αριθμητικές τιμές, βασίστηκε στο [25], που γίνεται συνδυασμός των δύο μεθόδων. Αυτό γίνεται χρησιμοποιώντας τον σταθμισμένο μέσο όρο ιστορικών δεδομένων και γραμμικής παρεμβολής. Τα βάρη της κάθε μεθόδου εξαρτώνται εκθετικά από την απόσταση της τρέχουσας άγνωστης τιμής από την κοντινότερη ημερομηνία που έχει τιμή σύμφωνα με τους τύπους:

$$w = e^{ad_i}$$

$$result = wL + (1 - w)H$$

όπου L η γραμμική παρεμβολή, H τα ιστορικά δεδομένα και d_i η προαναφερθείσα απόσταση. Το a αποτελεί μια σταθερά που καθορίζει το πόσο γρήγορα θα μειώνεται η συμβολή της γραμμικής παλινδρόμησης. Ως προκαθορισμένη τιμή επιλέχθηκε το 0,3. Επομένως, όσο αυξάνεται η απόσταση, μειώνεται η συνεισφορά της γραμμικής παρεμβολής και αυξάνεται η συνεισφορά των ιστορικών δεδομένων.

Τα ιστορικά δεδομένα για μια άγνωστη τιμή (εδώ τροποποιείται λίγο ο ορισμός του [25] με σκοπό την εξάρτηση του H από πιο πρόσφατες και σχετικές τιμές) ορίζονται ως ο μέσος όρος των τιμών της ίδιας μέρας της εβδομάδας που επίσης έχουν λιγότερο από WN_{cutoff} (προκαθορισμένη τιμή 1 λεπτό) απόσταση από την ώρα της άγνωστης τιμής, λιγότερο από Y_{cutoff} (προκαθορισμένη τιμή 3 χρόνια) απόσταση από τον χρόνο της άγνωστης τιμής και λιγότερο από YD_{cutoff} (προκαθορισμένη τιμή 30 μέρες) απόσταση από την ημέρα του χρόνου της άγνωστης τιμής.

Οι αργίες λαμβάνονται υπόψιν από την παρούσα υλοποίηση. Συγκεκριμένα, γίνεται χρήση μιας απλοποιημένης μεθόδου του [26]. Θεωρούνται είτε Σάββατα (αν η πραγματική μέρα είναι Παρασκευή) ή Κυριακές (σε κάθε άλλη περίπτωση). Τέλος, πρέπει να αναφερθεί ότι, παρόλο που στο [25] η μέθοδος αντικατάστασης άγνωστων

τιμών εφαρμόζεται σε έξυπνους μετρητές, δουλεύει αρκετά καλά και για άλλες χρονοσειρές που εμφανίζουν παρόμοια χαρακτηριστικά (όπως πχ ημερήσια, μηνιαία και ετήσια εποχικότητα). Τέλος, εφόσον ο χρήστης δεν επιθυμεί να εφαρμόσει αντικατάσταση άγνωστων τιμών με την παραπάνω μέθοδο, αλλά προτιμά να πραγματοποιήσει απλή γραμμική παλινδρόμηση, μπορεί να θέσει το πεδίο `l_interpolation` αληθές.

3.4.3 Χειρισμός πολυμεταβλητών ή/και πολλαπλών χρονοσειρών

Σε περίπτωση που το πρόβλημα αφορά πολυμεταβλητές ή/και πολλαπλές χρονοσειρές, η διαδικασία που αναφέρθηκε παραπάνω εκτελείται για κάθε συνιστώσα ξεχωριστά. Στο τέλος της εκτέλεσης του παρόντος σταδίου, πραγματοποιείται επιπλέον περικοπή της κάθε συνιστώσας κάθε χρονοσειράς, έτσι ώστε να διατηρείται μόνο το κοινό κομμάτι τους. Έτσι, όλες οι συνιστώσες μιας χρονοσειράς τελειώνουν και ξεκινούν από τις ίδιες ημερομηνίες.

3.5 Εκπαίδευση

Το στάδιο της εκπαίδευσης ξεκινά με την φόρτωση των χρονοσειρών στον υπολογιστή από τον διακομιστή MLflow.

Έπειτα, πραγματοποιείται διαχωρισμός της κάθε κάθε χρονοσειράς σε χρονοσειρές εκπαίδευσης, επικύρωσης και αξιολόγησης. Η πρώτη ημερομηνία των δεδομένων επικύρωσης και αξιολόγησης καθορίζεται από τα πεδία `cut_date_val` και `cut_date_test` αντίστοιχα. Επίσης, τα δεδομένα αξιολόγησης κόβονται μετά την ημερομηνία `test_end_date`. Πρέπει να σημειωθεί ότι στην περίπτωση που `cut_date_val = cut_date_test`, τότε το σύνολο δεδομένων επικύρωσης και αξιολόγησης συμπίπτουν.

Μετά την διαδικασία του διαχωρισμού, εκτελείται ενδεχομένως η κανονικοποίηση της κάθε χρονοσειράς. Για τον σκοπό αυτό, γίνεται χρήση του `scale` του `darts` [27], που πραγματοποιεί `MinMaxScaling`. Αυτό σημαίνει ότι κάθε τιμή της κάθε χρονοσειράς (έστω x) αντικαθίσταται από την τιμή:

$$x = \frac{x - x_{min}}{x_{max} - x_{min}}$$

όπου x_{min} , x_{max} η μέγιστη και η ελάχιστη τιμή της χρονοσειράς. Επομένως, οι τιμές της χρονοσειράς πάνω στην οποία εφαρμόζεται η κανονικοποίηση βρίσκονται πλέον στο διάστημα $[0, 1]$, με την ελάχιστη τιμή της χρονοσειράς να μετασχηματίζεται σε 0, και την μέγιστη σε 1. Πρέπει να σημειωθεί ότι το x_{min} και το x_{max} υπολογίζονται μόνο στα δεδομένα εκπαίδευσης. Τα δεδομένα επικύρωσης και αξιολόγησης, έτσι, μπορεί να περιλαμβάνουν τιμές εκτός του διαστήματος $[0, 1]$. Εφόσον ο χρήστης δεν επιθυμεί να κανονικοποιήσει τα δεδομένα του, μπορεί να θέσει το πεδίο `scale` ψευδές.

Σε περίπτωση που υπάρχουν συμμεταβλητές, οι παραπάνω διαδικασίες του διαχωρισμού και της κανονικοποίησης εκτελούνται με τον ίδιο τρόπο.

Μετά το τέλος της διαδικασίας προεπεξεργασίας, ξεκινούμε την διαδικασία εκπαίδευσης του μοντέλου που ζήτησε ο χρήστης. Οι υπερπαραμέτροι αυτού του μοντέλου μπορούν να δοθούν από τον χρήστη στο αρχείο `config.yml` σε μορφή `YAML`. Πιο συγκεκριμένα, οι υπερπαραμέτροι πρέπει να έχουν την μορφή:

```
hyperparams_entrypoint:
  parameter_1: value
  parameter_2: value
  ...
```

όπου `hyperparams_entrypoint` δίνεται από τον χρήστη στην εντολή εκτέλεσης της εφαρμογής. Το αρχείο αυτό μπορεί να έχει πολλούς συνδυασμούς υπερπαραμέτρων, και ο χρήστης επιλέγει κάθε φορά ποιες θα ληφθούν υπόψιν. Σημειωτέον ότι οι παραπάνω παράμετροι μπορούν να βρίσκονται σε οποιαδήποτε σειρά επιθυμεί ο χρήστης, αρκεί να μην επαναλαμβάνονται.

Μετά το πέρας της διαδικασίας εκπαίδευσης, αποθηκεύεται στον διακομιστή του MLflow το μοντέλο με το χαμηλότερο `loss` (MSE) σε περίπτωση που γίνεται εκπαίδευση μοντέλων βαθιάς μάθησης, και το τελικό μοντέλο σε περίπτωση που πραγματοποιείται εκπαίδευση μοντέλων μηχανικής μάθησης.

3.5.1 Χειρισμός πολλαπλών η/και πολυμεταβλητών χρονοσειρών

Αξίζει να γίνει ιδιαίτερη αναφορά στο γεγονός ότι η κανονικοποίηση και ο διαχωρισμός δεδομένων γίνεται για κάθε χρονοσειρά του μοντέλου ξεχωριστά σε περίπτωση που το πρόβλημα αφορά πολλαπλές χρονοσειρές. Βέβαια, τα χρονικά σημεία στα οποία γίνεται ο διαχωρισμός είναι πάντα τα ίδια για όλες τις χρονοσειρές. Επιπλέον, εφόσον ζητηθεί από τον χρήστη κανονικοποίηση, γίνεται αποθήκευση στον διακομιστή του MLflow όλων των κανονικοποιητών (`scalers`) που χρειάστηκαν, ένας για κάθε χρονοσειρά.

3.6 Αξιολόγηση

Σε αυτό το στάδιο πραγματοποιείται η τελική αξιολόγηση του μοντέλου στα δεδομένα που ζητήθηκαν από τον χρήστη. Αρχικά, φορτώνεται το μοντέλο και τα δεδομένα τοπικά από τον διακομιστή MLflow. Έπειτα, πραγματοποιείται εκ νέου η διαδικασία διαχωρισμού και κανονικοποίησης των δεδομένων που παρουσιάστηκε στο στάδιο εκπαίδευσης.

Σε αυτό το σημείο θα παρουσιαστεί η συνάρτηση `backtesting`, η οποία πραγματοποιεί προβλέψεις σε ορίζοντα πρόβλεψης (`forecast_horizon`) ο οποίος θα της δοθεί (εν προκειμένω μια μέρα). Πιο συγκεκριμένα, για κάθε χρονοσειρά που δίνουμε στην συνάρτηση, και ξεκινώντας από το `cut_date_val` μέχρι το τέλος του συνόλου αξιολόγησης, κάνει διαδοχικά προβλέψεις `forecast_horizon` βήματα μακριά. Αυτό γίνεται με βήμα (`stride`) ίσο του `forecast_horizon`. Αυτή η διαδικασία συμβαδίζει με τον στόχο της παρούσας εργασίας (Βραχυπρόθεσμη πρόβλεψη ζήτησης ηλεκτρικής ενέργειας), καθώς πρέπει γνωρίζοντας το φορτίο μέχρι μια δεδομένη στιγμή, να πραγματοποιηθεί πρόβλεψη μιας ημέρας μετά. Την επόμενη μέρα, θεωρούνται οι πραγματικές τιμές που παρατηρήθηκαν τελικά γνωστές, και επαναλαμβάνεται η διαδικασία πρόβλεψης για την μεθεπόμενη ημέρα. Έτσι, τελικά προκύπτει μια χρονοσειρά πρόβλεψης, η οποία συγκρίνεται με τις πραγματικές τιμές με χρήση των ακόλουθων μετρικών:

- Μέσο Απόλυτο Σφάλμα (Mean Absolute Error, MAE), με τύπο:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |A_i - F_i|$$

- Μέσο Απόλυτο Κανονικοποιημένο Σφάλμα (Mean Absolute Scaled Error, MASE), με τύπο:

$$\text{MASE} = \frac{\text{MAE}}{\frac{1}{n-1} \sum_{i=2}^n |A_i - A_{i-1}|}$$

- Ρίζα του Μέσου Τετραγωνικού Σφάλματος (Root Mean Squared Error, RMSE), με τύπο:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (A_i - F_i)^2}$$

- Μεσο Απόλυτο Ποσοστιαίο Σφάλμα (Mean Absolute Percentage Error, MAPE), με τύπο:

$$\text{MAPE} = \frac{1}{n} \sum_{i=1}^n \left| \frac{A_i - F_i}{A_i} \right| \times 100\%$$

- Συμμετρικό Μεσο Απόλυτο Ποσοστιαίο Σφάλμα (symmetric Mean Absolute Percentage Error, SMAPE), με τύπο:

$$\text{SMAPE} = \frac{1}{n} \sum_{i=1}^n \frac{|F_i - A_i|}{(F_i + A_i)/2} \times 100\%$$

Στους παραπάνω τύπους, A_i είναι η πραγματική τιμή, F_i είναι η προβλεπόμενη τιμή και n είναι ο αριθμός των παρατηρήσεων. Παράγονται επίσης γραφικές παραστάσεις της χρονοσειράς που προβλέφθηκε μαζί με τις πραγματικές τιμές, καθώς και αρχεία csv με τις προβλέψεις και την πραγματική χρονοσειρά.

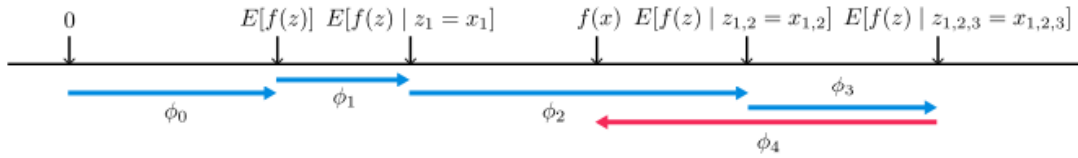
Το για ποιες χρονοσειρές εκτελείται η παραπάνω συνάρτηση εξαρτάται από τον τύπο του προβλήματος που πρέπει να επιλυθεί. Σε περίπτωση που ο χρήστης έχει εισάγει στην εφαρμογή μία χρονοσειρά, η συνάρτηση θα εκτελεστεί μόνο για το σύνολο δεδομένων αξιολόγησης αυτής. Το ίδιο θα συμβεί και στην περίπτωση που ο χρήστης έχει δώσει πολλαπλές η/και πολυμεταβλητές χρονοσειρές, με την μόνη διαφορά να είναι ότι ο χρήστης επιλέγει την χρονοσειρά που επιθυμεί να αξιολογηθεί βάσει του ID της ή το Source Code της. Αν ο χρήστης θέσει το πεδίο `evaluate_all_ts` αληθές, θα αξιολογηθεί το μοντέλο στις χρονοσειρές αξιολόγησης που προέρχονται από όλες τις χρονοσειρές, και θα αποθηκεύσει στο MLFlow τους μέσους όρους των προαναφερθέντων μετρικών. Επιπλέον, θα αποθηκευτούν αναλυτικά αποτελέσματα των μετρικών όλων των χρονοσειρών σε μορφή csv.

Επιπλέον, δίνεται η δυνατότητα ανάλυσης της εξόδου των μοντέλων βαθιάς και μηχανικής μάθησης με την χρήση της βιβλιοθήκης SHAP (SHapley Additive exPlanations). Παρακάτω ακολουθεί σύντομη παρουσίαση του τρόπου με τον οποίο λειτουργεί αυτή η βιβλιοθήκη, και της υλοποίησης του από την παρούσα εφαρμογή

3.6.1 Σύντομη περιγραφή του SHAP

Το SHAP (SHapley Additive exPlanations) αποτελεί μια βιβλιοθήκη για την εξήγηση της εξόδου οποιουδήποτε μοντέλου μηχανικής μάθησης. Βασίζεται στην έννοια των τιμών Shapley από τη θεωρία συνεργατικών παιγνίων, η οποία μετρά τη συμβολή κάθε παίκτη σε ένα παίγνιο στο συνολικό αποτέλεσμα. Σύμφωνα με το [28], οι συντελεστές SHAP (ϕ_i) προκύπτουν από τις τιμές Shapley μιας συνάρτησης δεσμευμένου μέσου όρου της εξόδου του μοντέλου. Αυτές, μπορούν να χρησιμοποιηθούν για την εξήγηση της συμβολής όλων των χαρακτηριστικών ενός μοντέλου. Κάθε ϕ_i δείχνει πόσο μεταβάλλεται η έξοδος του μοντέλου, δεδομένης της τρέχουσας τιμής του αντίστοιχου χαρακτηριστικού. Ένα από τα πιο χρήσιμα χαρακτηριστικά των τιμών αυτών είναι η αθροιστική φύση τους. Αυτό σημαίνει ότι ξεκινώντας από την $E[f(z)]$, η οποία αποτελεί την τιμή που θα πρόβλεπε το μοντέλο αν δεν ήταν γνωστή συγκεκριμένη τιμή κανενός

χαρακτηριστικού του (ϕ_0), και προσθέτοντας διαδοχικά όλους τους υπόλοιπους συντελεστές (ϕ_i), θα προέκυπτε τελικά η τιμή του μοντέλου για τις γνωστές πλέον τιμές όλων των χαρακτηριστικών του $f(z)$.



Σχήμα 3.3: Η αθροιστική φύση των τιμών SHAP.²

Σε περίπτωση που το μοντέλο είναι μη γραμμικό, ή τα χαρακτηριστικά δεν είναι ανεξάρτητα, η σειρά που προστίθενται τα ϕ_i έχει σημασία. Έτσι, οι τιμές των ϕ_i μεταβάλλονται. Για να υπάρχουν συγκεκριμένες τιμές SHAP σε αυτή την περίπτωση, απλά λαμβάνεται ο μέσος όρος των ϕ_i για όλες τις πιθανές διατάξεις.

Στην παρούσα εφαρμογή γίνεται χρήση του Kernel Explainer για την εκτίμηση των τιμών SHAP. Η μέθοδος αυτή δεν κάνει κάποια υπόθεση σχετικά με τον τύπο του μοντέλου που χρησιμοποιείται (Model Agnostic), και για αυτό είναι σχετικά αργή σε σχέση με άλλες μεθόδους. Φυσικά το πρώτο είναι απαραίτητο για την δεδομένη εφαρμογή, καθώς είναι επιθυμητό να μπορεί η μέθοδος αυτή να εφαρμοστεί σε όλα τα μοντέλα που μπορεί να επιθυμεί να εκπαιδεύσει ο χρήστης. Στην μέθοδο αυτή γίνεται χρήση μιας ειδικής σταθμισμένης γραμμικής παλινδρόμησης για τον υπολογισμό της εκτίμησης των συντελεστών SHAP. Αυτό γίνεται για όλα τα δείγματα που θα δώσει στην συνάρτηση ο χρήστης.

3.6.2 Υλοποίηση SHAP για χρονοσειρές

Σύμφωνα με όσα αναλύθηκαν παραπάνω, για να γίνει χρήση του SHAP, είναι αναγκαία η ύπαρξη συγκεκριμένων χαρακτηριστικών των μοντέλων. Όμως, τα μοντέλα βαθιάς και μηχανικής μάθησης που υποστηρίζονται από την παρούσα εφαρμογή παράγουν από τις χρονοσειρές που λαμβάνουν δείγματα εκπαίδευσης που ουσιαστικά αποτελούν κομμάτια σταθερού μεγέθους (παράθυρο προηγούμενης γνώσης και ορίζοντας εκπαίδευσης³ για τις εισόδους και τις εξόδους αντίστοιχα) της χρονοσειράς. Αυτό σημαίνει ότι ως χαρακτηριστικά του κάθε μοντέλου θεωρούνται οι χρονικές μονάδες μετά την αρχή του κάθε δείγματος. Επιπλέον, εφόσον το μοντέλο αυτό δέχεται και συμμεταβλητές, και αυτές λειτουργούν με παρόμοιο τρόπο. Πιο συγκεκριμένα, οι παρελθοντικές συμμεταβλητές γενικά χρειάζεται να είναι γνωστές για όλη την είσοδο, ενώ οι μελλοντικές για το διάστημα της εισόδου και της εξόδου. Σε αυτή την περίπτωση, συνεπώς, θεωρούμε επιπλέον τις χρονικές μονάδες μετά την αρχή της κάθε συμμεταβλητής για όλο το διάστημα που χρειάζεται από το μοντέλο να είναι γνωστή ως χαρακτηριστικά.

Πριν ξεκινήσει η εφαρμογή της μεθόδου SHAP, πρέπει να επιλεγούν τα δείγματα του συνόλου αξιολόγησης για τα οποία θα υπολογιστούν οι συντελεστές SHAP. Αυτά επιλέγονται τυχαία, και ο χρήστης έχει την δυνατότητα να καθορίσει με το πεδίο `shap_data_size` τον αριθμό τους(αν είναι int) ή το ποσοστό τους(αν είναι float).

²Πηγή: [28]

³Αυτά τα δύο αναφέρονται ως `input_chunk_length` και `output_chunk_length` αντίστοιχα στον κώδικα

Αφότου παραχθούν τα δείγματα, υπολογίζονται οι τιμές SHAP για κάθε ένα, και παράγονται σχετικές γραφικές παραστάσεις. Πιο συγκεκριμένα, παράγεται ένα διάγραμμα beeswarm, το οποίο δείχνει για όλα τα δείγματα που λήφθηκαν πόσο ψηλή ή χαμηλή ήταν η τιμή κάθε χαρακτηριστικού και η τιμή των συντελεστών SHAP για αυτό το χαρακτηριστικό. Επιπλέον, παράγεται ένα ραβδογράφημα που δείχνει απλά τον μέσο όρο της απόλυτης τιμής των συντελεστών SHAP για κάθε χαρακτηριστικό. Το διάγραμμα αυτό αποτελεί την πιο σύντομη έκφραση της σημασίας του κάθε χαρακτηριστικού που παράγεται από το SHAP. Τέλος, παράγονται ενδεικτικά 3 διαγράμματα force plot, που δείχνουν για ένα τυχαίο δείγμα την ακριβή τιμή των συντελεστών SHAP, και μοιάζουν με το σχήμα 3.3. Δείχνουν δηλαδή την προσθετική μορφή των συντελεστών αυτών όπως αναλύθηκε παραπάνω. Αυτά τα 3 διαγράμματα αναφέρονται σε μια έξοδο το καθένα, και πιο συγκεκριμένα στις εξόδους 0, το μισό του παράθυρο προηγούμενης γνώσης και το παράθυρο προηγούμενης γνώσης μείον ένα. Τα διαγράμματα αυτά παρουσιάζονται και σχολιάζονται εκτενέστερα στο Κεφάλαιο 4.

3.7 Βελτιστοποίηση υπερπαραμέτρων (hyperparameter tuning) με χρήση της βιβλιοθήκης Optuna

Σε αυτό το στάδιο πραγματοποιείται βελτιστοποίηση υπερπαραμέτρων με χρήση της Optuna. Αρχικά, θα γίνει μια σύντομη περιγραφή της θεωρίας πίσω από την μέθοδο αναζήτησης υπερπαραμέτρων αυτής.

3.7.1 Θεωρητική ανάλυση

Η optuna [22] [21] αποτελεί μια βιβλιοθήκη που πραγματοποιεί βελτιστοποίηση υπερπαραμέτρων. Η διαδικασία αυτή γίνεται από την optuna με πολλούς τρόπους, στην παρούσα εργασία όμως χρησιμοποιούμε δύο: το tree-structured Parzen Estimator (TPE), και την αναζήτηση πλέγματος (grid search).

Η πρώτη μέθοδος [29] αποτελεί μια πιθανοτική μέθοδο, η οποία σε κάθε τρέξιμο παράγει 2 πυκνότητες πιθανότητας βάσει των παρατηρήσεων (υπερπαραμέτροι που δοκιμάστηκαν σε προηγούμενα τρεξίματα της μεθόδου). Η πρώτη πυκνότητα πιθανότητας $l(x)$ παράγεται από τις παρατηρήσεις των οποίων η τιμή του σφάλματος είναι μικρότερη από y^* , ενώ η δεύτερη από τις παρατηρήσεις με μεγαλύτερο σφάλμα από το y^* . Το y^* υπολογίζεται κάθε φορά τέτοιο ώστε η πιθανότητα η παρατήρηση του επόμενου τρεξίματος να είναι μικρότερη από αυτό να είναι ίση με γ , όπου γ ένα σταθερό ποσοστό των παρατηρήσεων. Οι υπερπαραμέτροι που προτείνονται στο κάθε τρέξιμο διαλέγονται επιλέγοντας τον συνδυασμό με την μικρότερη τιμή $g(x) / l(x)$ από τυχαίες υποψήφιες παρατηρήσεις που διαλέχθηκαν με πυκνότητα πιθανότητας $l(x)$. Οι g και l , στην περίπτωση συνεχών υπερπαραμέτρων αποτελούν ένα μείγμα από γκαουσιανές με κέντρο κάθε παρατήρηση, και ίδιο βάρος. Στην περίπτωση διακριτών υπερπαραμέτρων με N τιμές, για κάθε τιμή i η πιθανότητα να επιλεγεί αυτή είναι ανάλογη του $Np_i + C_i$, όπου p_i η πιθανότητα επιλογής της i τιμής πριν την έναρξη της βελτιστοποίησης, και C_i οι φορές που επιλέχθηκε αυτή η τιμή για το g ή το l .

Η μέθοδος αναζήτησης πλέγματος απλά δοκιμάζει όλους τους συνδυασμούς υπερπαραμέτρων με σκοπό να βρει αυτήν με το μικρότερο σφάλμα. Είναι επομένως αρκετά πιο αργή σε σχέση με την TPE, είναι ωστόσο δεδομένο ότι θα βρει την ελάχιστη τιμή του σφάλματος, πράγμα που δεν ισχύει για την TPE.

Τέλος στο πλαίσιο της βελτιστοποίησης υπερπαραμέτρων χρησιμοποιείται μέθοδος [30] για τον υπολογισμό της σημαντικότητας κάθε υπερπαραμέτρου στην τιμή του

σφάλματος. Η μέθοδος αυτή βασίζεται στην Functional Analysis of Variance (fANOVA), που αποτελεί μια μέθοδο αποσύνθεσης της απόκλισης μιας συνάρτησης βάσει της συμβολής σε αυτήν κάθε υποσυνόλου των μεταβλητών της. Για τον υπολογισμό της σημασίας κάθε υπερπαραμέτρου θα εκπαιδευτεί μοντέλο random forest το οποίο θα προβλέπει την τιμή του σφάλματος κάθε τρεξίματος της optuna βάσει των υπερπαραμέτρων του. Η μέθοδος αποσύνθεσης που προαναφέρθηκε εφαρμόζεται έπειτα στην συνάρτηση σφάλματος αυτού του μοντέλου.

3.7.2 Υλοποίηση

Στην παρούσα εφαρμογή, όπως έχει προαναφερθεί, βελτιστοποίηση υπερπαραμέτρων εκτελείται μόνο εφόσον το πεδίο opt_test είναι αληθές. Σε αυτή την περίπτωση, αφού εκτελεστούν τα στάδια αρχικής φόρτωσης και ελέγχου των δεδομένων και επεξεργασίας δεδομένων, η εφαρμογή διαβάζει τις υπερπαραμέτρους του μοντέλου που επιθυμεί να εκπαιδευτεί ο χρήστης από το αρχείο config_orp.yml. Είναι γραμμένο σε μορφή YAML, και περιέχει πολλούς συνδυασμούς υπερπαραμέτρων. Ο χρήστης επιλέγει ποιον συνδυασμό θα λάβει υπόψιν η εφαρμογή επιλέγοντας το κατάλληλο hyperparams_entrypoint όταν εκτελεί την αντίστοιχη εντολή. Πιο συγκεκριμένα, οι συνδυασμοί υπερπαραμέτρων θα πρέπει να γράφονται ως ακολούθως:

```
hyperparams_entrypoint:
  parameter_range: ["range", start, end, step]
  parameter_list: ["list", value_1, value_2, ..., value_n]
  parameter_constant: value
  ...
```

Για τις παραπάνω μορφές ισχύουν τα εξής:

- Για όσες υπερπαραμέτρους γράφονται με την μορφή του parameter_range, η αναζήτηση θα πραγματοποιηθεί για τις τιμές [start, start + step, ..., end]. Τα start, end και step μπορούν να είναι μορφής int και float.
- Για όσες υπερπαραμέτρους γράφονται με την μορφή του parameter_list, η αναζήτηση θα πραγματοποιηθεί για τις τιμές [value_1, value_2, ..., value_n]. Τα value_i μπορούν να είναι μορφής int, float και string.
- Για όσες υπερπαραμέτρους γράφονται με την μορφή του parameter_constant, αυτές θεωρούνται σταθερές και δεν αλλάζουν μεταξύ των διάφορων τρεξιμάτων.

Σημειωτέον ότι οι παραπάνω μορφές μπορούν να βρίσκονται σε οποιαδήποτε σειρά επιθυμεί ο χρήστης, αρκεί να μην επαναλαμβάνονται.

Αφού πραγματοποιηθεί η προαναφερθείσα ανάγνωση, ξεκινούν να εκτελούνται τα τρεξίματα της optuna, τον αριθμό των οποίων τον καθορίζει ο χρήστης (μέσω του πεδίου n_trials). Πρέπει να σημειωθεί ότι είναι δυνατή η εκτέλεση τόσο της προαναφερθείσας μεθόδου tre για αναζήτηση υπερπαραμέτρων, όσο και της μεθόδου αναζήτησης πλέγματος (grid search). Ο χρήστης επιλέγει ποια από τις δυο μεθόδους επιθυμεί να εκτελεστεί, ανάλογα με το πλήθος των συνδυασμών υπερπαραμέτρων που πρέπει να ελεγχθούν.

Η διαδικασία που εκτελείται είναι παρόμοια με τα δύο στάδια της εκπαίδευσης και της αξιολόγησης που αναφέρθηκε παραπάνω. Η πιο σημαντική διαφορά εντοπίζεται στο στάδιο της αξιολόγησης: Η συνάρτηση backtesting δεν εκτελείται για το

σύνολο αξιολόγησης, αλλά για το σύνολο επικύρωσης. Αυτό συμβαίνει διότι τα δεδομένα αξιολόγησης πρέπει να είναι εντελώς άγνωστα στο μοντέλο. Ένα πιο σωστό όνομα για αυτή την διαδικασία επομένως, είναι διαδικασία επικύρωσης (validation).

Φυσικά, όπως και στο στάδιο αξιολόγησης, το κάθε μοντέλο μπορεί να επικυρωθεί με διάφορους τρόπους αναλόγως του τύπου του συνόλου δεδομένων που ενδιαφέρει τον χρήστη. Έτσι, σε περίπτωση που αυτό είναι μια απλή χρονοσειρά, η backtesting εκτελείται μια φορά στην χρονοσειρά επικύρωσης. Στην περίπτωση που το σύνολο δεδομένων, όμως, είναι πολλαπλές χρονοσειρές, γίνεται η συνάρτηση backtesting να εκτελεστεί για μία μόνο από τις χρονοσειρές ή και για όλες, παράγοντας στην τελευταία περίπτωση τον μέσο όρο των μετρικών για όλες τις χρονοσειρές. Στην περίπτωση αυτή, ακόμα, παράγεται αρχείο για κάθε τρέξιμο της αναζήτησης υπερπαραμέτρων με τα αναλυτικά αποτελέσματα για κάθε χρονοσειρά.

Η βελτιστοποίηση υπερπαραμέτρων, χρειάζεται μια μετρική του σφάλματος του μοντέλου για να λειτουργήσει. Η μετρική αυτή επιλέγεται από τον χρήστη μέσω του πεδίου `loss_function`. Η προεπιλεγμένη τιμή είναι MAPE, αλλά υποστηρίζονται και οι μετρικές SMAPE, MASE, MAE και RMSE. Η βιβλιοθήκη `optuna` αποθηκεύει τα αποτελέσματα του κάθε τρεξίματός της και τα επιστρέφει επίσης και στην γραμμή εντολών.

Στην παρούσα εφαρμογή γίνεται χρήση της δυνατότητας της βιβλιοθήκης `optuna` να αποθηκεύει το αντικείμενο `study` σε ένα αρχείο με κατάλληλη `.db` (εν προκειμένω `memory.db`). Έτσι, εάν για κάποιον λόγο η εκτέλεση της βελτιστοποίησης διακοπεί, μπορεί να δοθεί πάλι η ίδια εντολή, και η βελτιστοποίηση θα έχει χάσει μόνο το τρέχον τρέξιμο. Μετά το πέρας της διαδικασίας, επιπλέον, από αυτό το αρχείο γίνεται να έχει ο χρήστης πλήρη πρόσβαση στα αποτελέσματα της βελτιστοποίησης, και να πραγματοποιήσει περαιτέρω ανάλυση από αυτή που γίνεται αυτόματα αν το επιθυμεί.

Στον διακομιστή `MLflow` αποθηκεύεται το μοντέλο που πέτυχε την χαμηλότερη μετρική σφάλματος, μαζί με τις υπερπαραμέτρους που του αντιστοιχούν. Επιπλέον, αποθηκεύονται και διαγράμματα, τα οποία μάλιστα ανανεώνονται όσο προχωρά η εκτέλεση του παρόντος σταδίου. Πιο συγκεκριμένα, αποθηκεύεται διάγραμμα ιστορικού της διαδικασίας που δείχνει την πρόοδο της μετρικής λάθους σε σχέση με τα τρεξίματα που γίνονταν. Επίσης, αποθηκεύεται διάγραμμα σημαντικότητας των υπερπαραμέτρων που χρησιμοποιεί την μέθοδο που έχει εξηγηθεί παραπάνω. Τέλος, γίνεται αποθήκευση διαγραμμάτων που δείχνουν για κάθε υπερμεταβλητή την τιμή της συνάρτησης σφάλματος.

Κεφάλαιο 4

Πειραματική διαδικασία

4.1 Εισαγωγή

Η εφαρμογή χρησιμοποιείται πάνω στο σύνολο δεδομένων από τον οργανισμό European Network of Transmission System Operators for Electricity (ENTSO-E) [31] που περιλαμβάνει αρχεία από 42 χρονοσειρές διάφορων χωρών. Όπως θα δειχθεί παρακάτω, οι χρονοσειρές αυτές μειώθηκαν σε 20, καθεμία από τις οποίες αναφερόταν σε μια χώρα. Τα παραπάνω έχουν ανάλυση μιας ώρας, και αποτελούν την ζήτηση ηλεκτρικής ενέργειας σε kWh των διαχειριστών συστημάτων διανομής (TSOs) των χωρών Βέλγιο, Τσεχία, Δανία, Εσθονία, Φινλανδία, Γαλλία, Ελλάδα, Ουγγαρία, Ιταλία, Λετονία, Λιθουανία, Κάτω Χώρες, Νορβηγία, Πορτογαλία, Ρουμανία, Σλοβακία, Σλοβενία, Ισπανία, Σουηδία και Ελβετία. Ως δεδομένα εκπαίδευσης θεωρούνται οι χρονοσειρές από 2015-04-09 ως 2019-12-31. Ως δεδομένα επικύρωσης θα οριστούν από 2020-01-01 έως 2020-12-31. Ως δεδομένα αξιολόγησης θεωρούνται οι χρονοσειρές από 2021-01-01 έως 2022-01-01. Ο παραπάνω διαχωρισμός πραγματοποιείται έτσι για όλες τις χώρες. Η εφαρμογή εκτελείται σε μηχάνημα με χαρακτηριστικά: 32 CPU cores, 64GB RAM, NVIDIA V100 32GB.

4.2 Προετοιμασία του συνόλου δεδομένων

Αρχικά, κάποια αρχεία (DE-LU MBA.csv, Ukraine BEI CTA.csv, Ukraine IPS CTA.csv) δεν έχουν τα δεδομένα τους με χρονολογική σειρά. Έτσι, πραγματοποιείται ταξινόμηση των ημερομηνιών σε αύξουσα σειρά.

Τα αρχεία δεν έχουν σταθερό όνομα ανάλογα με την χώρα στην οποία ανήκει το καθένα. Για αυτό, πραγματοποιείται μετονομασία των αρχείων με ονόματα σχετικά με την χώρα προέλευσής τους.

Καποιες χώρες (Ιταλία, Ουκρανία, Δανία, Νορβηγία, Σουηδία) είναι χωρισμένες σε πολλαπλές BZ (Betting Zones). Αυτό έχει ως αποτέλεσμα μεγαλύτερες διακυμάνσεις στα δεδομένα και προβλήματα σχετικά με το ότι μια χώρα με πολλές χρονοσειρές θα επηρέαζε τα αποτελέσματα του υπερμοντέλου περισσότερο από ότι θα έπρεπε (πχ η Ιταλία με 6 χρονοσειρές θα επηρέαζε περισσότερο το αποτέλεσμα από ότι η Γερμανία με μία χρονοσειρά). Για αυτό, γίνεται άθροιση των csvs που αναφέρονται στην ίδια χώρα. Τελικά, δηλαδή, έχουμε 1 csv που δείχνει το φορτίο για κάθε χώρα.

Ακολουθώς, με σκοπό την διατήρηση τουλάχιστον ενός χρόνου στο σύνολο δεδομένων εκπαίδευσης, απορρίφθηκαν όλες οι χρονοσειρές που δεν είχαν δεδομένα πριν την 2019-01-01 00:00:00 UTC. Συνεπώς, στο σύνολο δεδομένων περιλαμβάνονται οι εξής 20 χώρες: Βέλγιο, Τσεχία, Δανία, Εσθονία, Φινλανδία, Γαλλία, Ελλάδα, Ουγγαρία, Ιταλία, Λετονία, Λιθουανία, Κάτω Χώρες, Νορβηγία, Πορτογαλία, Ρουμανία, Σλοβακία, Σλοβενία, Ισπανία, Σουηδία και Ελβετία. Όλες οι παραπάνω χώρες έχουν δεδομένα που ξεκινούν από την ημερομηνία 2015-04-09 23:00:00 UTC, και σταματούν την ημερομηνία 2022-04-01 01:00:00 UTC (εκτός της Ιταλίας, που σταματά την 2022-03-31 22:00:00 UTC)

Έπειτα, τα csvs μετατράπηκαν σε ένα της μορφής που περιγράφεται στο σχήμα

3.2 για πολλαπλές χρονοσειρές. Πιο συγκεκριμένα, το πρόβλημα αντιμετωπίζεται σαν ένα πρόβλημα πολλαπλών χρονοσειρών μιας συνιστώσας η καθεμία (το φορτίο κάθε χώρας). Αυτή η χρονοσειρά αποθηκεύεται στο `vm`, και ξεκινάει η εκτέλεση της διαδικασίας σωλήνωσης. Δοκιμάζονται πολλές προσεγγίσεις για την λύση του προβλήματος, οι οποίες θα αναφερθούν απο την υποενότητα 4.5 και μετά μαζί με τις ακριβείς εντολές που εκτελέστηκαν. Αρχικά, θα γίνει ανάλυση των αποτελεσμάτων των σταδίων αρχικής φόρτωσης και ελέγχου των δεδομένων και επεξεργασίας των δεδομένων, που είναι κοινά για όλες τις εντολές. Αποτελούν, εξάλλου, βήματα που πραγματοποιούνται πριν την εκπαίδευση μοντέλων στον κύκλο ζωής μηχανικής μάθησης.

4.3 Αρχική φόρτωση και έλεγχος των δεδομένων

Αρχικά πραγματοποιούνται οι έλεγχοι που αναφέρθηκαν στο κεφάλαιο 3, και το αρχείο της χρονοσειράς αποθηκεύεται στο `MLflow`. Προφανώς, τα αρχεία που δίδονται στην εφαρμογή πληρούν όλες τις προϋποθέσεις που αναλύθηκαν.

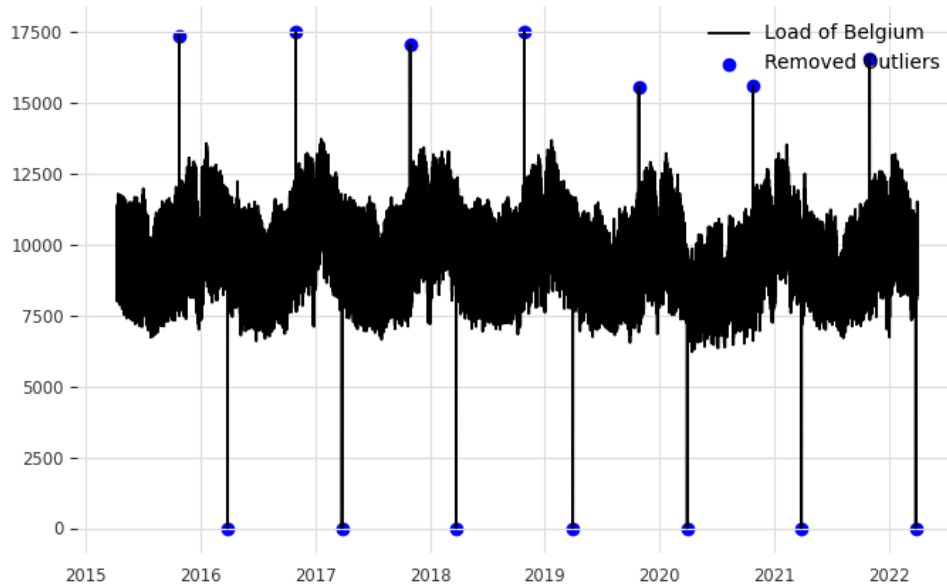
4.4 Επεξεργασία δεδομένων

Έπειτα, ακολουθεί το στάδιο της επεξεργασίας δεδομένων. Συγκεκριμένα, αρχικά γίνεται η επιλογή οι χρονοσειρές να εκφράζονται στην τοπική ώρα της χώρας στην οποία αντιστοιχούν. Αυτό γίνεται έτσι ώστε όλες οι χρονοσειρές να συμβαδίζουν με την ώρα με την οποία γίνονται οι ανθρώπινες δραστηριότητες στην κάθε χώρα. Προφανώς, κυρίως αυτές επηρεάζουν την ζήτηση ηλεκτρικής ενέργειας.

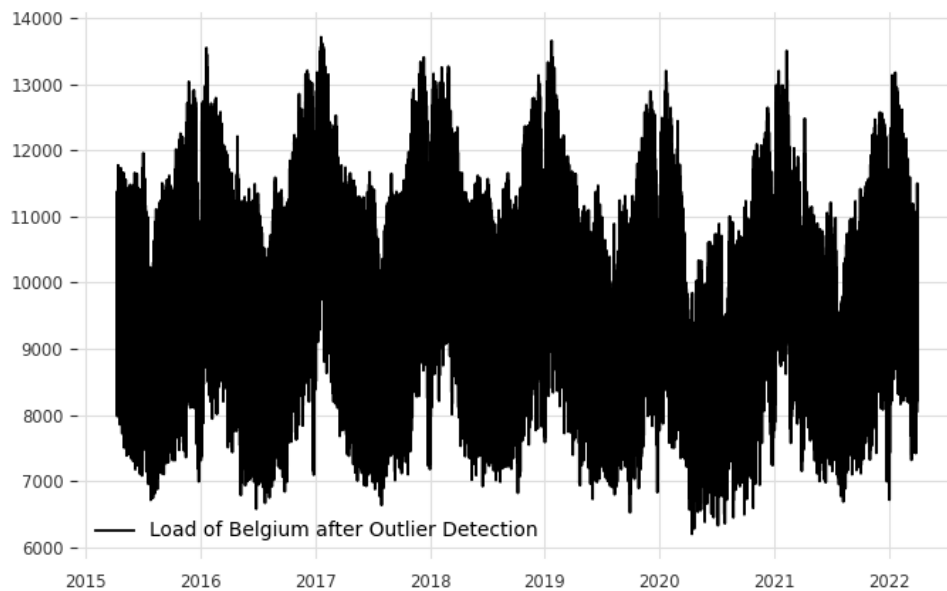
4.4.1 Εντοπισμός ακραίων τιμών (Outlier detection)

Ακολούθως, πραγματοποιείται για κάθε χρονοσειρά ξεχωριστά εντοπισμός ακραίων τιμών. Ενδεικτικά θα παρουσιαστούν για το Βέλγιο, την Ελλάδα και την Πορτογαλία διαγράμματα με τις τιμές που αφαιρέθηκαν από την κάθε χρονοσειρά, καθώς και η τελική τους μορφή.

Όπως παρατηρείται στην περίπτωση του Βελγίου, αφαιρέθηκαν μόνο οι τιμές που ήταν πολύ ψηλά ή χαμηλά (περισσότερες από 4.5 κανονικές κατανομές πάνω από τον μέσο όρο του μήνα τους). Οι τιμές αυτές θα έκαναν την κανονικοποίηση μεγίστου ελαχίστου να μην δουλεύει σωστά.

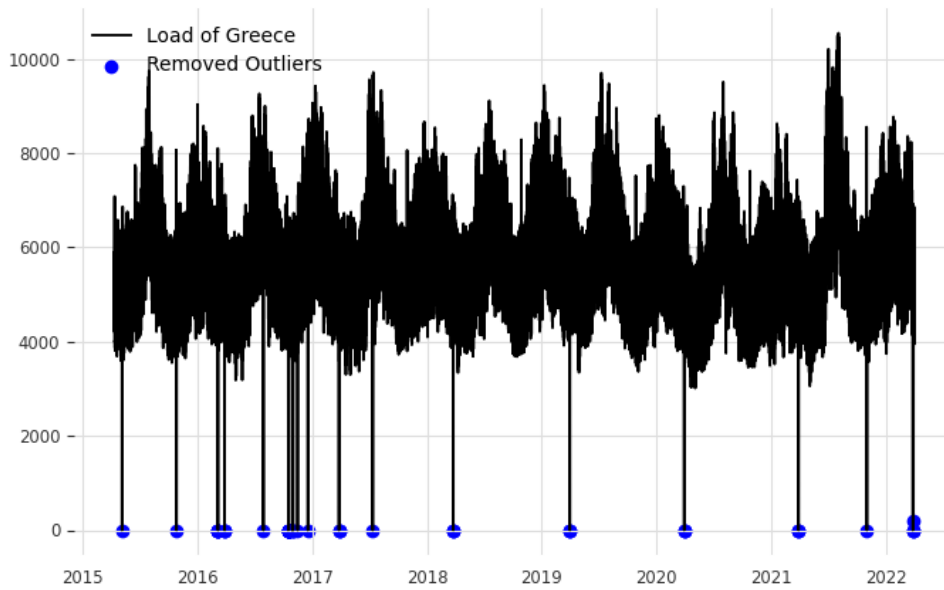


Σχήμα 4.1: Οι τιμές που αφαιρέθηκαν από την χρονοσειρά του Βελγίου

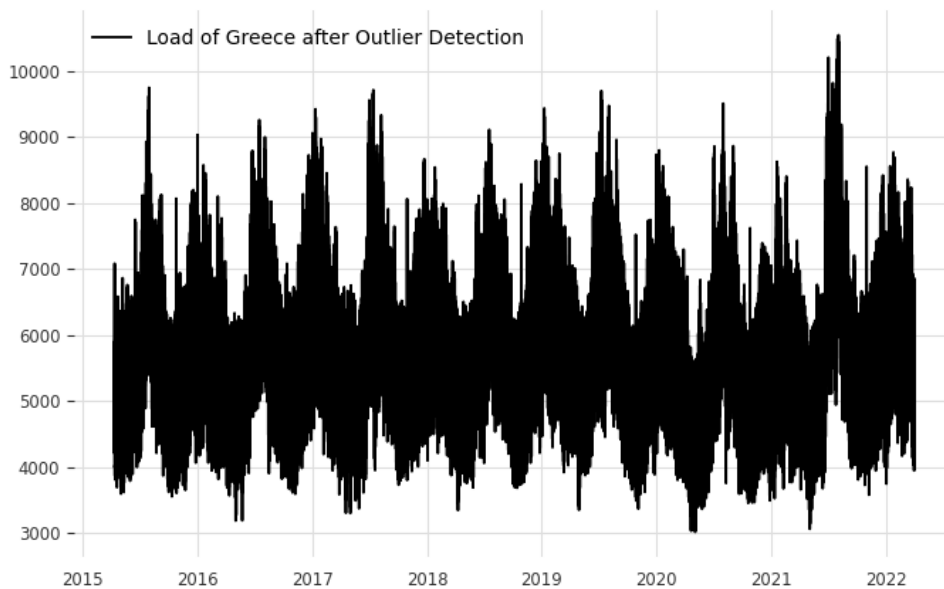


Σχήμα 4.2: Η τελική μορφή της χρονοσειράς του Βελγίου

Παρατηρείται ότι στην περίπτωση της Ελλάδας αφαιρέθηκαν μόνο οι τιμές που είναι ίσες με μηδέν. Συμβάντα όπως ο παρατεταμένος καύσωνας του 2021 παραμένουν στα δεδομένα όπως θα έπρεπε, καθώς δεν οφείλονται σε λανθασμένα δεδομένα, ούτε δημιουργούν πολύ μεγάλο πρόβλημα στην κανονικοποίηση.

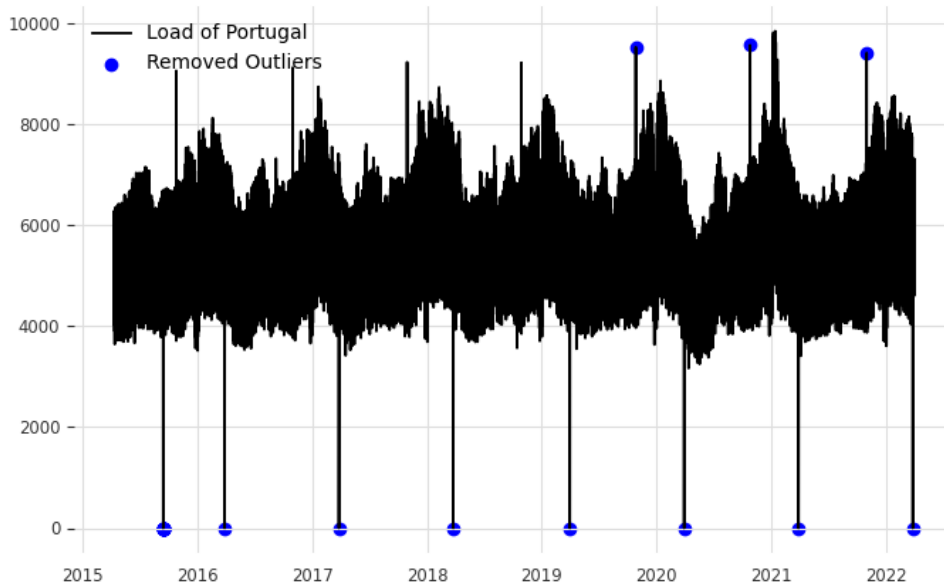


Σχήμα 4.3: Οι τιμές που αφαιρέθηκαν από την χρονοσειρά της Ελλάδας

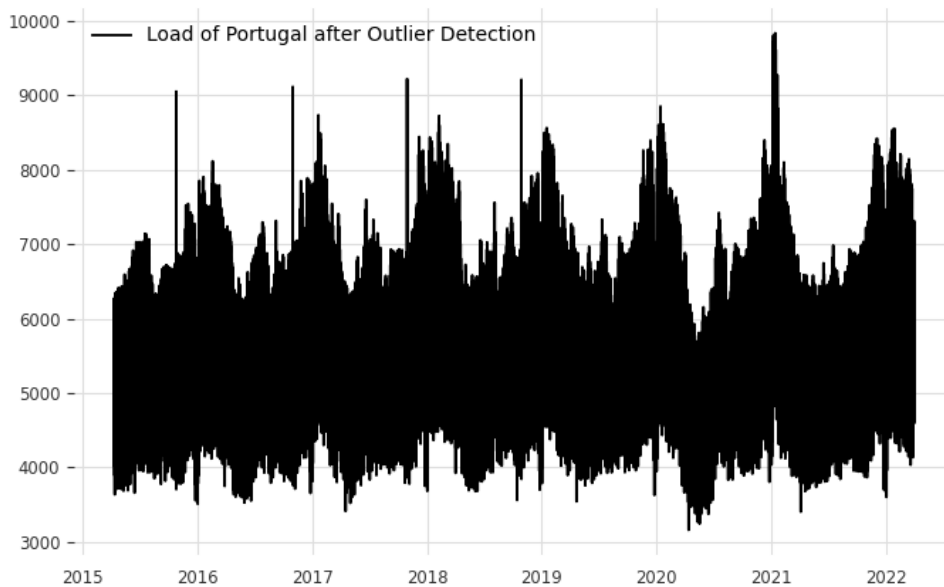


Σχήμα 4.4: Η τελική μορφή της χρονοσειράς της Ελλάδας

Για την χρονοσειρά της Πορτογαλίας αφαιρέθηκαν όλες οι μηδενικές τιμές, καθώς και κάποιες τιμές το 2020, 2021 και 2022 που ήταν κατά πολύ υψηλότερες από τον μέσο όρο του μήνα τους. Ομοίως με την περίπτωση της Ελλάδας, συμβάντα παρατεταμένης ψηλής ζήτησης, όπως στην αρχή του 2021, δεν θεωρούνται ακραίες τιμές.



Σχήμα 4.5: Οι τιμές που αφαιρέθηκαν από την χρονοσειρά της Πορτογαλίας



Σχήμα 4.6: Η τελική μορφή της χρονοσειράς της Πορτογαλίας

Παρακάτω παρουσιάζεται ο αριθμός των ακραίων τιμών που αφαιρέθηκαν για κάθε χρονοσειρά. Οι ακριβείς τιμές και οι ημερομηνίες και ώρες που αφαιρέθηκαν είναι διαθέσιμες στον διακομιστή του MLflow.

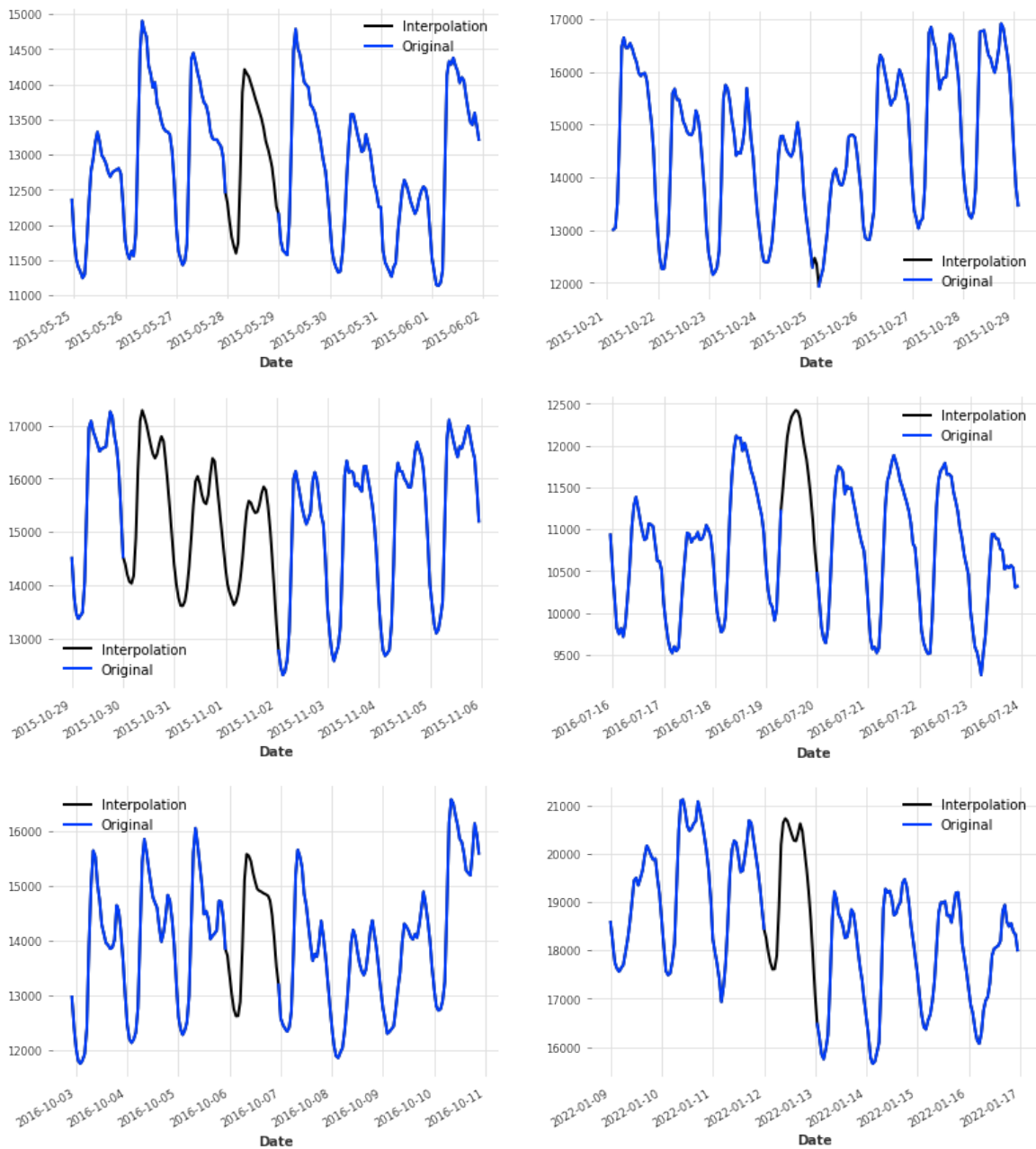
Πίνακας 4.1: Αριθμός ακραίων τιμών που αφαιρέθηκαν ανά χρονοσειρά

Χώρα	Αριθμός ακραίων τιμών
Βέλγιο	14
Τσεχία	227
Δανία	14
Εσθονία	230
Φινλανδία	95
Γαλλία	95
Ελλάδα	50
Ουγγαρία	18
Ιταλία	55
Λετονία	80
Λιθουανία	198
Κάτω Χώρες	7
Νορβηγία	273
Πορτογαλία	34
Ρουμανία	461
Σλοβακία	211
Σλοβενία	62
Ισπανία	26
Σουηδία	107
Ελβετία	62

4.4.2 Αντικατάσταση άγνωστων τιμών (Imputation)

Όπως αναλύθηκε στο Κεφάλαιο 3, μετά την αφαίρεση ακραίων τιμών, ακολουθεί η αντικατάσταση των άγνωστων τιμών (μαζί με αυτές που αφαιρέθηκαν από το προηγούμενο βήμα) με τιμές υπολογισμένες από το συνδυασμό ιστορικών δεδομένων και γραμμικής παλινδρόμησης. Στον διακομιστή του MLflow αποθηκεύονται όλες οι τελικές γραφικές παραστάσεις των χρονοσειρών μετά την αντικατάσταση, καθώς και csvs με τις ακριβείς τιμές που αντικαταστάθηκαν. Λόγω μεγάλης ομοιότητας με τα διαγράμματα που παρουσιάστηκαν παραπάνω, αυτά δεν θα παρατεθούν εδώ. Αντί αυτού, θα παρουσιαστεί με μεγαλύτερη λεπτομέρεια η λειτουργία του αλγορίθμου αντικατάστασης άγνωστων τιμών για την χρονοσειρά της Νορβηγίας.

Σχήμα 4.7: Τιμές που αντικαταστάθηκαν από την χρονοσειρά της Νορβηγίας



Στο παραπάνω διάγραμμα φαίνονται μερικά από τα διαστήματα με κενές τιμές που αντικαταστάθηκαν για την εν λόγω χρονοσειρά. Με μαύρη γραμμή συμβολίζονται οι νέες τιμές που προστέθηκαν, ενώ με μπλε συμβολίζεται η ήδη υπάρχουσα χρονοσειρά. Παρατηρείται ότι στα μικρά διαστήματα (όπως πχ αυτό της 2015-10-16) κυριαρχεί η γραμμική παλινδρόμηση. Όσο αυτά μεγαλώνουν, οι ιστορικές προβλέψεις παίζουν τον μεγαλύτερο ρόλο. Γενικότερα, οι καμπύλες των ωρών που έχουν προστεθεί είναι παρόμοιας μορφής και πλάτους με τις γειτονικές τους. Αυτό δείχνει ότι η αντικατάσταση ήταν επιτυχής.

Παρακάτω παρατίθεται ο πίνακας των αριθμών των τιμών που αντικαταστάθηκαν ανα χρονοσειρά.

Παρατηρείται ότι οι τιμές αυτές είναι ίδιες για όλες τις χρονοσειρές με τον αριθμό

Πίνακας 4.2: Αριθμός τιμών που αντικαταστάθηκαν ανά χώρα

Χώρα	Αριθμός τιμών
Βέλγιο	14
Τσεχία	227
Δανία	14
Εσθονία	230
Φινλανδία	95
Γαλλία	95
Ελλάδα	50
Ουγγαρία	18
Ιταλία	55
Λετονία	80
Λιθουανία	198
Κάτω Χώρες	7
Νορβηγία	273
Πορτογαλία	34
Ρουμανία	461
Σλοβακία	211
Σλοβενία	62
Ισπανία	26
Σουηδία	107
Ελβετία	62

των τιμών στις οποίες έγινε αφαίρεση ακραίων τιμών. Συνεπώς, δεν υπήρχαν άγνωστες τιμές στην αρχή σε καμία χρονοσειρά.

4.5 Επιλογή μοντέλου και μετρικής

Τα μοντέλα που θα επιλεγούν είναι το LightGBM και το NBEATS. Το πρώτο αποτελεί μια γρήγορη και ευέλικτη μέθοδο μηχανικής μάθησης, ενώ το δεύτερο έχει υψηλές αποδόσεις σε δεδομένα χωρών της Ευρώπης παρόμοια με της παρούσας εργασίας [32] [33]. Επίσης, το LightGBM έχει χρησιμοποιηθεί σε προβλήματα πρόβλεψης φορτίου [34], και από τον νικητή του M5 [10]. Το NBEATS ακόμα χρησιμοποιείται για δεδομένα φορτίου στο [35]. Από τις μετρικές που αποθηκεύονται από την εφαρμογή, αυτές που θα αναλυθούν αποτελούν το MAPE και το RMSE.

4.6 Αναλυτικά αποτελέσματα μοντέλων Naive

Σε αυτό το σημείο θα παρουσιαστούν τα αποτελέσματα των μοντέλων Naive μιας εβδομάδας και μίας ημέρας. Τα μοντέλα αυτά θεωρούν ως πρόβλεψη μιας συγκεκριμένης ώρας απλά την τιμή μια εβδομάδα και μια ημέρα πριν αντίστοιχα. Παρακάτω παρουσιάζεται η εντολή που εκτελέστηκε έτσι ώστε να προκύψουν τα αποτελέσματα που θα παρουσιαστούν:

```
nohup mlflow run --experiment-name theo_official
--entry-point exp_pipeline . -P convert_to_local_tz=true
-P country=$country -P cut_date_test=20210101
```

```

-P cut_date_val=20200101 -P darts_model=Naive
-P day_first=true -P device=gpu -P forecast_horizon=24
-P hyperparams_entrypoint=naive_d -P ignore_previous_runs=t
-P l_interpolation=false -P multiple=false -P num_workers=4
-P opt_test=false -P resolution=60 -P retrain=False
-P rmv_outliers=true -P scale=false
-P series_csv=../../RDN/Load_Data/country_dfs/$country.csv
-P test_end_date=20220101 -P time_covs=None
-P year_range=2015-2022 --env-manager=local

```

Στην θέση του \$country τοποθετήθηκε κάθε φορά η χώρα για την οποία εκτελείται η εν λόγω εντολή. Έτσι, η αντικατάσταση των άγνωστων τιμών γίνεται για την σωστή χώρα κάθε φορά (η παράμετρος country είναι υπεύθυνη για αυτό), και το αρχείο που διαβάζεται είναι το σωστό (παράμετρος series_csv). Επιπλέον, στην παράμετρο hyperparams_entrypoint επιλέχθηκε naive_d για την ημερήσια naive, και naive_w για την εβδομαδιαία naive μέθοδο. Η σημασία των υπόλοιπων παραμέτρων εξηγείται λεπτομερώς στο παράρτημα.

Η παραπάνω εντολή, επομένως, εκτελέστηκε 2 φορές για την κάθε χώρα. Παρακάτω παρουσιάζονται αναλυτικά για την κάθε χώρα τα αποτελέσματα του μοντέλου Naive μιας ημέρας και μιας εβδομάδας για τα δεδομένα αξιολόγησης (01-01-2021 έως 01-01-2022):

Πίνακας 4.3: Αποτελέσματα του μοντέλου Naive μιας ημέρας ανά χώρα

ID χρονοσειράς	RMSE	MAPE
Βέλγιο	751.22	5.41
Τσεχία	694.97	6.21
Δανία	355.34	5.87
Εσθονία	88.29	6.24
Φινλανδία	511.19	3.73
Γαλλία	4108.46	5.31
Ελλάδα	452.39	5.53
Ουγγαρία	422.78	5.91
Ιταλία	4791.55	9.62
Λετονία	70.05	5.05
Λιθουανία	121.87	5.57
Κάτω Χώρες	1130.31	6.22
Νορβηγία	767.15	3.4
Πορτογαλία	552.65	6.52
Ρουμανία	601.33	5.99
Σλοβακία	242.05	4.9
Σλοβενία	190.86	7.57
Ισπανία	2518.81	5.98
Σουηδία	1126.03	5.07
Ελβετία	547.24	5.47
Μέσος Όρος	1002.23	5.78

Πίνακας 4.4: Αποτελέσματα του μοντέλου Naive μιας εβδομάδας ανά χώρα

ID χρονοσειράς	RMSE	MAPE
Βέλγιο	633.76	4.58
Τσεχία	559.76	4.98
Δανία	306.4	5.5
Εσθονία	89.98	6.95
Φινλανδία	791.96	5.76
Γαλλία	5853.07	7.15
Ελλάδα	686.3	7.98
Ουγγαρία	399.03	5.65
Ιταλία	2848.98	5.37
Λετονία	52.09	4.31
Λιθουανία	100.84	4.96
Κάτω Χώρες	1002.53	5.6
Νορβηγία	1285.48	5.56
Πορτογαλία	403.98	4.31
Ρουμανία	432.15	4.32
Σλοβακία	196.12	3.96
Σλοβενία	171.29	6.4
Ισπανία	1990.63	4.51
Σουηδία	1500.64	6.87
Ελβετία	587.44	6.26
Μέσος Όρος	994.62	5.55

Παρατηρείται ότι ο μέσος όρος MAPE για το Naive μιας εβδομάδας (5.55) είναι ελαφρώς χαμηλότερος από τον αντίστοιχο για το ημερήσιο (5.78). Αυτό είναι αναμενόμενο, αφού εκτός της ημερήσιας εποχικότητας, οι χρονοσειρές έχουν και εβδομαδιαία εποχικότητα. Αυτό σημαίνει ότι το Naive μιας εβδομάδας τις λαμβάνει υποψιν αμφότερες, ενώ το Naive μια ημέρας μόνο την μια. Το ίδιο ισχύει και για το RMSE. Βέβαια υπάρχουν και κάποιες χρονοσειρές (όπως πχ αυτή της Σουηδίας) για τις οποίες η εβδομαδιαία Naive μέθοδος είναι χειρότερη από την ημερήσια. Αυτό μπορεί να οφείλεται στην ύπαρξη μεγαλύτερων διακυμάνσεων μεταξύ των εβδομάδων.

4.7 Αναλυτικά αποτελέσματα μοντέλου NBEATS

Συνεχίζοντας την πειραματική διαδικασία, πραγματοποιείται αξιολόγηση του μοντέλου NBeats. Πρέπει να σημειωθεί ότι για όλα τα παρακάτω πειράματα εφαρμόστηκε early stopping. Συγκεκριμένα, ελέγχεται εάν το MSE στο σύνολο αξιολόγησης έχει μειωθεί κατά 6×10^{-6} , σε σχέση με όσο ήταν δέκα εποχές πιο πριν. Εάν αυτό συμβεί τότε η διαδικασία εκπαίδευσης συνεχίζει, εάν όχι τότε σταματάει.

4.7.1 Βελτιστοποίηση υπερπαραμέτρων

Η εντολή που θα εκτελεστεί για να γίνει βελτιστοποίηση υπερπαραμέτρων με την βιβλιοθήκη optuna για το μοντέλο NBeats είναι η ακόλουθη:

```
nohup mlflow run --experiment-name theo_official --entry-point exp_pipeline .
-P convert_to_local_tz=true -P cut_date_test=20210101 -P cut_date_val=20200101
```

```

-P darts_model=NBEATS -P day_first=true -P device=gpu -P evaluate_all_ts=true
-P forecast_horizon=24 -P from_mongo=false -P grid_search=false
-P hyperparams_entrypoint=NBEATS_thesis_multiple -P ignore_previous_runs=t
-P l_interpolation=false -P multiple=true -P n_trials=50 -P num_workers=4
-P opt_test=true -P resolution=60 -P retrain=False -P rmv_outliers=true
-P scale=true -P series_csv=../../RDN/Load_Data/Multiple_Country_Series.csv
-P test_end_date=20220101 -P time_covs=None
-P year_range=2015-2022 --env-manager=local

```

Η ακριβής σημασία της κάθε παραμέτρου εξηγείται σε Παράρτημα στο τέλος. Οι παράμετροι που δεν αναφέρονται στην εντολή με `-P` διατηρούν την προεπιλεγμένη τιμή τους. Στην παραπάνω εντολή δείχνουμε στο MLflow να αποθηκεύσει όλα τα αποτελέσματα στο πείραμα (experiment) `theo_official` μέσω της παραμέτρου `-experiment-name`. Εκεί θα αποθηκευτούν και όλα τα υπόλοιπα τρεξίματα της εν λόγω εργασίας. Είναι, τέλος, αναγκαία η επιλογή `-env-manager=local` έτσι ώστε να μην δημιουργηθεί καινούργιο περιβάλλον `conda` από την αρχή, αλλά η εντολή να εκτελεστεί χρησιμοποιώντας το προϋπάρχον.

Άξια αναφοράς είναι ακόμα η μορφή του σημείου εισόδου (entrypoint) `NBEATS_thesis_multiple` στο αρχείο `config_opt.yml`:

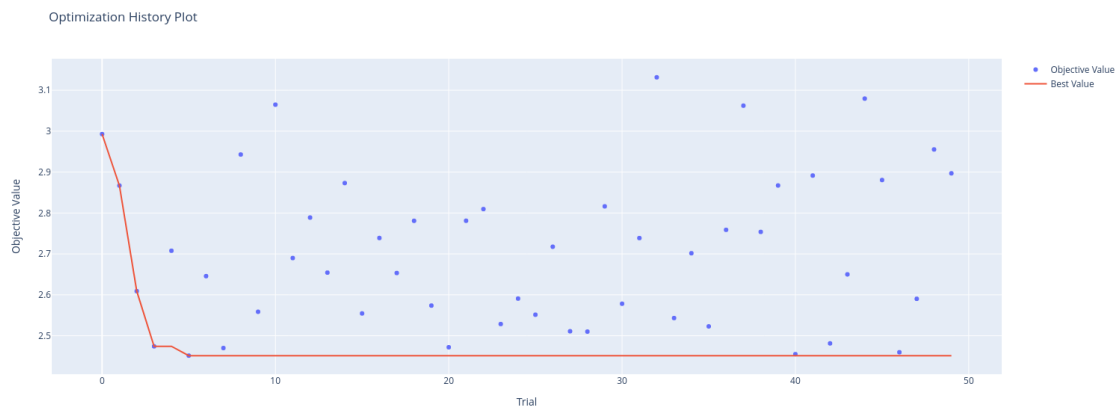
```

NBEATS_thesis_multiple:
  input_chunk_length: ["range", 120, 312, 24]
  output_chunk_length: 24
  num_stacks: ["range", 1, 10, 1]
  num_blocks: ["range", 1, 6, 1]
  num_layers: ["range", 1, 10, 1]
  generic_architecture: True
  layer_widths: 64
  expansion_coefficient_dim: 5
  n_epochs: 200
  random_state: 0
  nr_epochs_val_period: 2
  batch_size: 1024
  scale: ["list", "True", "False"]

```

Όπως έχει εξηγηθεί στην ενότητα 3, οι παράμετροι που βρίσκονται σε λίστα είναι αυτές για τις οποίες γίνεται προσπάθεια βελτιστοποίησης. Επιπλέον, η παράμετρος `nr_epochs_val_period` δείχνει κάθε πόσες εποχές θα εκτελείται επικύρωση του μοντέλου, προκειμένου να αξιολογηθεί κατά πόσο είναι καλύτερο από το τρέχων αποθηκευμένο. Σε αυτή την περίπτωση, το αντικαθιστά. Τέλος, γίνεται έλεγχος και κατά πόσο βελτιώνει ή χειροτερεύει το αποτέλεσμα η κανονικοποίηση της χρονοσειράς. Αυτό πραγματοποιείται με την παράμετρο `scale`.

Η βελτιστοποίηση παραμέτρων, που χρησιμοποιεί το TPE, εκτελείται για 50 τρεξίματα (trials). Η τιμή MAPE στο σύνολο επικύρωσης (01-01-2020 έως 31-12-2020) χρησιμοποιείται για την αξιολόγηση των διάφορων μοντέλων. Η εξέλιξη αυτής της διαδικασίας φαίνεται στο διάγραμμα παρακάτω:



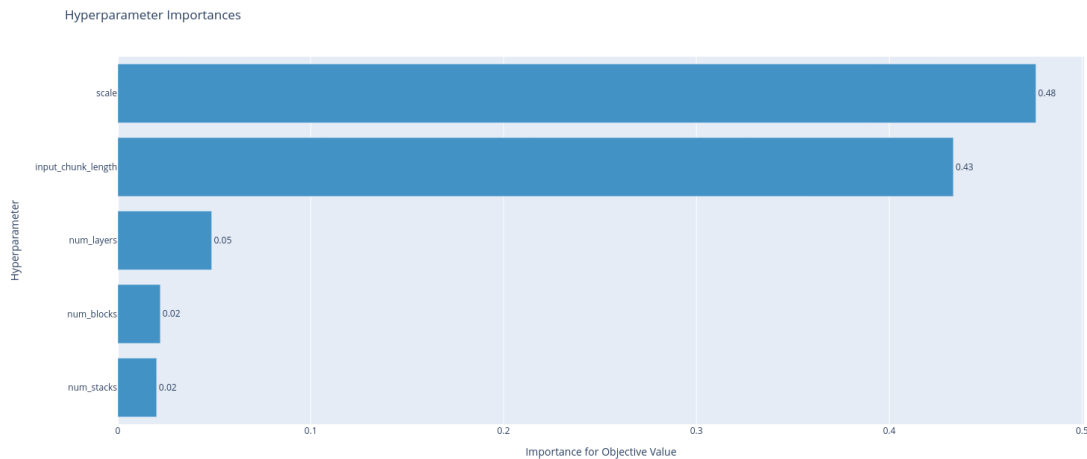
Σχήμα 4.8: Το MAPE των δεδομένων επικύρωσης για τα τρεξίματα της βελτιστοποίησης υπερπαραμέτρων

Οι υπερπαραμέτροι του καλύτερου μοντέλου είναι οι εξής:

Πίνακας 4.5: Τιμές του μοντέλου NBEATS με το χαμηλότερο MAPE

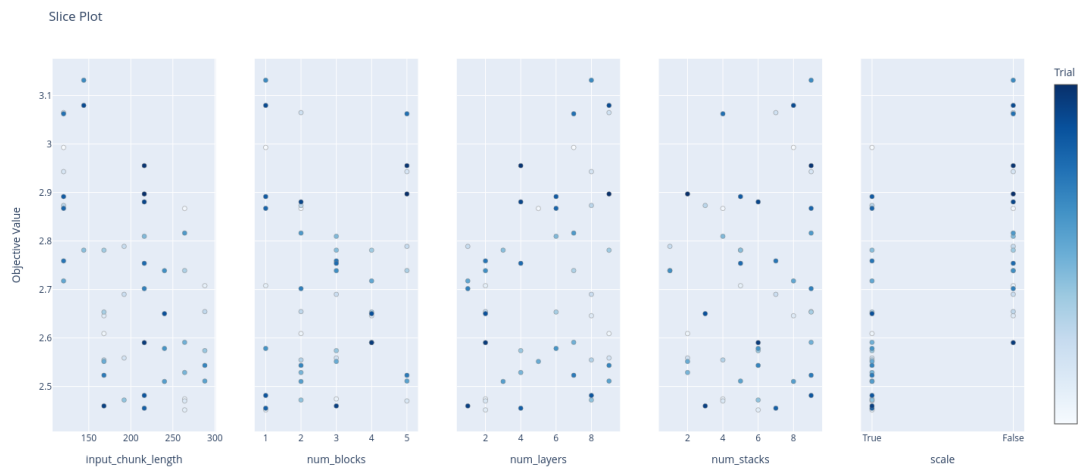
Παράμετρος	Τιμή
input_chunk_length	264
output_chunk_length	24
num_stacks	6
num_blocks	1
num_layers	2
generic_architecture	True
layer_widths	64
expansion_coefficient_dim	5
n_epochs	200
random_state	0
nr_epochs_val_period	2
batch_size	1024
scale	True

Οι τιμές με έντονη γραφή αποτελούν εκείνες για τις οποίες εκτελέστηκε βελτιστοποίηση. Επίσης, η τιμή της μετρικής MAPE στο σύνολο δεδομένων επικύρωσης είναι 2.45 και η τιμή του RMSE 387.7. Το καλύτερο μοντέλο είχε χρόνο εκπαίδευσης και επικύρωσης 01:57:07. Παρακάτω παρουσιάζεται και η σημαντικότητα (importance) της κάθε υπερπαραμέτρου σύμφωνα με τον αλγόριθμο της optuna που αναφέρθηκε στο κεφάλαιο 3:



Σχήμα 4.9: Το πόσο σημαντική είναι η κάθε υπερπαράμετρος που βελτιστοποιήθηκε στο τελικό αποτέλεσμα του NBEATS

Όπως φαίνεται, οι πιο σημαντικές υπερπαράμετροι είναι το αν η χρονοσειρα είναι κανονικοποιημένη ή όχι, και το παράθυρο πρότερης γνώσης (input chunk length ή lookback window). Αυτές, εξηγούν το 48% και 43% της απόκλισης του MAPE αντίστοιχα. Ακολουθούν οι num_layers(5%), num_blocks(2%) και num_stacks(2%). Με σκοπό την πιο λεπτομερή παρουσίαση του πως επηρεάζει το MAPE η κάθε υπερπαράμετρος, θα παρουσιαστεί διάγραμμα που δείχνει την τιμή mape που παρήγαγε κάθε δοκιμή και την τιμή της αντίστοιχης υπερπαραμέτρου. Υπάρχει ένα υποδιάγραμμα για κάθε υπερπαράμετρο:



Σχήμα 4.10: Αναλυτικά αποτελέσματα ανά υπερπαράμετρο του NBEATS

Αρχικά, η επιλογή της κανονικοποίησης του μοντέλου προκαλεί μείωση στο MAPE. Αυτό είναι αναμενόμενο, καθώς οι χρονοσειρές έχουν διαφορετικά μεγέθη. Έτσι, σε περίπτωση μη εφαρμογής κανονικοποίησης ορισμένες χρονοσειρές (αυτές με τις μεγαλύτερες τιμές) θα λαμβάνονταν υποψιν πολύ περισσότερο από άλλες.

Επιπλέον, η υπερπαράμετρος του παράθυρου πρότερης γνώσης, έχει ελάχιστο την τιμή 264 (11 ημέρες). Γενικότερα, φαίνεται ότι όσο μεγαλύτερο το παράθυρο, τόσο μικρότερη η τιμή του MAPE. Αυτό συμβαίνει διότι το μοντέλο λαμβάνει υπόψιν του

μεγαλύτερο διάστημα πριν την κάθε πρόβλεψη.

Ακόμα, παρατηρείται ότι μικρές τιμές της παραμέτρου `num_blocks` οδηγούν σε μικρότερο MAPE. Αυτό συμβαίνει διότι το μοντέλο καταφέρνει να αποσυνθέσει τις χρονοσειρές με μικρό αριθμό block. Ομοίως, μικρές τιμές της παραμέτρου `num_layers` φαίνεται να παράγουν μικρότερο mape. Αντιθέτως, η παράμετρος `num_stacks` φαίνεται να έχει ελάχιστο την τιμή 6, και το MAPE να αυξάνεται για μικρότερες και μεγαλύτερες τιμές. Η μικρότερη τιμή MAPE, επομένως, παρατηρείται σε μια μετρίως βαθιά αρχιτεκτονική, καθώς ο μεγάλος αριθμός των χρονοσειρών περιπλέκει αρκετά την διαδικασία μάθησης και ευνοεί αυτού του είδους τις αρχιτεκτονικές. Βέβαια, η συνεισφορά των τελευταίων 2 τιμών στην τιμή του MAPE είναι αρκετά μικρότερη από τις προηγούμενες όπως φάνηκε και από το διάγραμμα 4.9.

4.7.2 Αξιολόγηση μοντέλου και SHAP

Από την βελτιστοποίηση υπερπαραμέτρων έχει προκύψει το καλύτερο μοντέλο του οποίου οι παράμετροι παρουσιάστηκαν στον πίνακα 4.5. Για αυτό το μοντέλο, το οποίο βρίσκεται αποθηκευμένο στον διακομιστή του MLflow, εκτελείται η ακόλουθη εντολή με σκοπό την αξιολόγησή του για όλες τις χρονοσειρές στα δεδομένα αξιολόγησης (01-01-2021 έως 01-01-2022):

```
nohup mlflow run . -e eval -P mode=remote
-P series_uri=$series_uri -P evaluate_all_ts=true -P model_type=pl
-P scaler_uri=$scaler_uri -P cut_date_test=20210101 -P test_end_date=20220101
-P model_uri=$model_uri -P forecast_horizon=24 -P stride=None -P retrain=False
-P input_chunk_length=264 -P output_chunk_length=24 -P cut_date_val=20200101
-P day_first=true -P multiple=true -P resolution=60 -P analyze_with_shap=false
--env-manager=local --experiment-name=theo_official
```

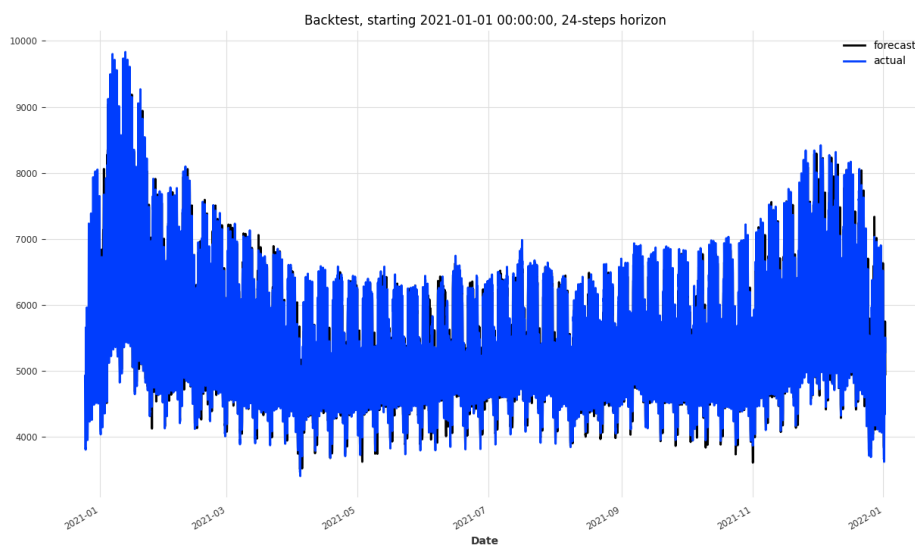
Η παραπάνω εντολή εκτελεί μόνο το στάδιο της αξιολόγησης. Για τον σκοπό αυτό έχει προστεθεί η επιλογή `-e eval`. Επιπλέον, στην θέση των `$series_uri`, `$scaler_uri`, `$model_uri` θα μπουν τα uris της χρονοσειράς, του αντικειμένου που πραγματοποιεί την κανονικοποίηση, και του μοντέλου αντίστοιχα. Αυτά είναι διαθέσιμα στο MLflow, στα αποτελέσματα τις προηγούμενης εντολής για την βελτιστοποίηση των υπερπαραμέτρων.

Παρακάτω παρουσιάζονται τα αποτελέσματα της αξιολόγησης του μοντέλου NBEATS για όλες τις χρονοσειρές στα δεδομένα αξιολόγησης, μαζί με τον μέσο όρο τους που αποθηκεύεται στον διακομιστή του MLflow:

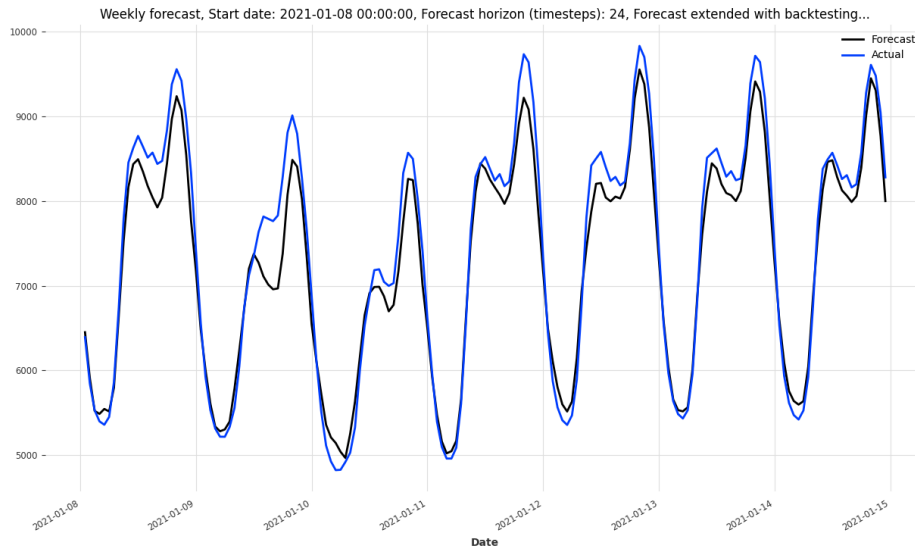
Πίνακας 4.6: Αποτελέσματα του μοντέλου NBEATS

ID χρονοσειράς	RMSE	MAPE
Βέλγιο	362.57	2.6
Τσεχία	249.72	2.25
Δανία	183.16	3.05
Εσθονία	49.39	3.65
Φινλανδία	325.02	2.33
Γαλλία	1885.99	2.42
Ελλάδα	306.62	3.48
Ουγγαρία	224.98	2.95
Ιταλία	1513.32	2.72
Λετονία	34.57	2.65
Λιθουανία	57.36	2.74
Κάτω Χώρες	860.71	4.78
Νορβηγία	508.47	2.19
Πορτογαλία	229.19	2.5
Ρουμανία	251.84	2.41
Σλοβακία	112.56	2.11
Σλοβενία	120.18	3.63
Ισπανία	970.42	2.24
Σουηδία	617.12	2.79
Ελβετία	407.27	4.26
Μέσος Όρος	463.5	2.89

Παρατηρείται βελτίωση σε σχέση με τα Naive μοντέλα για όλες τις χώρες και τις μετρικές, καθώς και για τον μέσο όρο τους. Στον διακομιστή του MLflow αποθηκεύονται επιπλέον και διαγράμματα που δείχνουν τις προβλέψεις του μοντέλου για όλες τις χώρες. Ενδεικτικά παρατίθενται παρακάτω τα διαγράμματα για την Πορτογαλία:



Σχήμα 4.11: Πρόβλεψη για το 2021 του NBEATS για την χρονοσειρά της Πορτογαλίας



Σχήμα 4.12: Πρόβλεψη για την δεύτερη εβδομάδα του 2021 για την χρονοσειρά της Πορτογαλίας απο το μοντέλο NBEATS

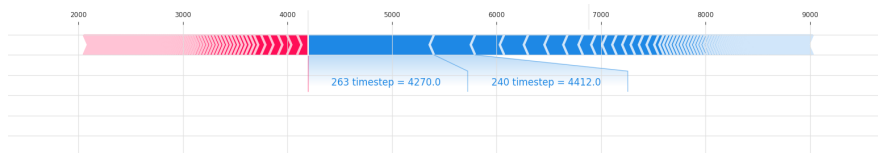
Στα διαγράμματα εμφανίζεται με μπλε γραμμή η πρόβλεψη του μοντέλου, και με μαύρη η πραγματική τιμή της χρονοσειράς. Το πρώτο διάγραμμα αναφέρεται σε όλη την περίοδο αξιολόγησής (01-01-2021 έως 01-01-2022), ενώ το δεύτερο στην δεύτερη εβδομάδα αυτής (08-01-2021 έως 15-01-2021). Όπως παρατηρείται γενικά η πρόβλεψη του μοντέλου είναι αρκετά κοντά στην πραγματική μορφή της χρονοσειράς. Το μοντέλο, ακόμα, καταφέρνει να αποτυπώσει αρκετά καλά την μορφή της ημερήσιας καμπύλης κατανάλωσης φορτίου.

Με σκοπό την καλύτερη κατανόηση της σημασίας κάθε χρονικού σημείου του παρθύρου πρότερης γνώσης, εκτελείται εξήγηση του αποτελέσματος του μοντέλου με την βιβλιοθήκη SHAP. Για να συμβεί αυτό, εκτελείται η εντολή:

```
nohup mlflow run . -e eval -P mode=remote -P series_uri=$series_uri
-P evaluate_all_ts=false -P model_type=pl -P scaler_uri=$scaler_uri
-P cut_date_test=20210101 -P test_end_date=20220101
-P model_uri=$model_uri -P forecast_horizon=24 -P stride=None -P retrain=False
-P input_chunk_length=264 -P output_chunk_length=24 -P cut_date_val=20200101
-P day_first=true -P eval_method=ts_code -P eval_series=PT -P multiple=true
-P resolution=60 -P analyze_with_shap=true -P size=100 --env-manager=local
--experiment-name=theo_official
```

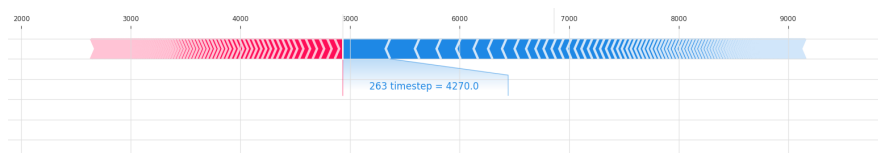
Όπως και προηγουμένως, η παραπάνω εντολή εκτελεί μόνο το στάδιο της αξιολόγησης. Για τον σκοπό αυτό έχει προστεθεί η επιλογή `-e eval`. Επιπλέον, στην θέση των `$series_uri`, `$scaler_uri`, `$model_uri` θα μπουν τα `uris` της χρονοσειράς, του αντικειμένου που πραγματοποιεί την κανονικοποίηση, και του μοντέλου αντίστοιχα. Αντιθέτως με την προηγούμενη εντολή, ωστόσο, δεν επιλέγουμε να πραγματοποιηθεί αξιολόγηση σε όλες τις χρονοσειρές. Αντιθέτως, θέτοντας `eval_series=PT` και `evaluate_all_ts=false`, η αξιολόγηση πραγματοποιείται μόνο στην χρονοσειρά της Πορτογαλίας. Η μέθοδος SHAP μπορεί να εφαρμοστεί σε μόνο μια χρονοσειρά αξιολόγησης. Επίσης, θέτοντας `size=100`, επιλέγονται 100 τυχαία δείγματα από την χρονοσειρά αξιολόγησης της Πορτογαλίας. Παρακάτω παρουσιάζονται τα διαγράμματα που παράγονται από αυτή την διαδικασία.

Αρχικά επιλέγεται το δείγμα για το οποίο θα σχεδιαστούν τα αναλυτικά διαγράμματα των τιμών των συντελεστών SHAP. Αυτό που επιλέγεται είναι το δείγμα 51 που ξεκινά την 2021-04-27 05-00-00. Για αυτό το δείγμα θα σχεδιαστούν τρία τέτοια διαγράμματα.



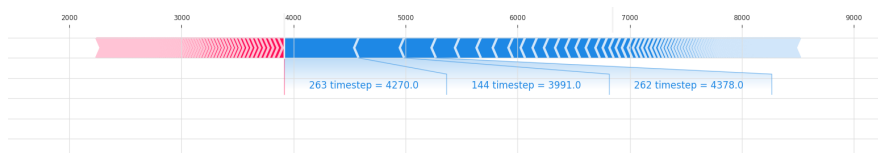
Σχήμα 4.13: Οι συντελεστές SHAP ενός δείγματος για την πρώτη τιμή της εξοδου του μοντέλου NBEATS

Παραπάνω παρουσιάζεται το διάγραμμα που αναφέρεται στην πρώτη έξοδο του τυχαίου δείγματος. Αναγράφεται με την έντονη κάθετη κόκκινη γραμμή η πρόβλεψη του μοντέλου για αυτό το δείγμα και αυτή την έξοδο. Τα βέλη δείχνουν πόσο ψηλότερα (κόκκινα) ή χαμηλότερα (μπλε) μεταβάλλει την πρόβλεψη κάθε feature. Όπως περιγράφηκε στο κεφάλαιο 3, το μέγεθος των βελών αυτών είναι ο μέσος όρος των τιμών SHAP για το κάθε χαρακτηριστικό και για αυτό το δείγμα. Αν γινόταν άθροιση των συνεισφορών αυτών των βελών ξεκινώντας από τον μέσο όρο της πρόβλεψης του μοντέλου για αυτή την είσοδο, θα προέκυπτε η πρόβλεψη του μοντέλου. Οι τιμές δίπλα στα = είναι απλά οι τιμές του κάθε χαρακτηριστικού στο εν λόγω δείγμα. Όπως ήταν αναμενόμενο, την μεγαλύτερη συμβολή στο αποτέλεσμα την έχει η τιμή την αμέσως προηγούμενη ώρα της πρόβλεψης ($\text{input_chunk_length} - 1 = 263$ timestep). Επίσης αναμενόμενα, λόγω της ημερήσιας εποχικότητας, η τιμή 24 ώρες πριν την πρόβλεψη (240) έχει την δεύτερη μεγαλύτερη συμβολή στο αποτέλεσμα.



Σχήμα 4.14: Οι συντελεστές SHAP ενός δείγματος για την δέκατη τρίτη τιμή της εξοδου του μοντέλου NBEATS

Παραπάνω έχει σχεδιαστεί το διάγραμμα που αναφέρεται στην δέκατη τρίτη έξοδο του ίδιου δείγματος. Παρατηρείται ότι η τιμή μία ώρα πριν την πρώτη πρόβλεψη έχει την μεγαλύτερη συμβολή στο αποτέλεσμα. Αυτό ενδεχομένως να συμβαίνει διότι είναι η πιο κοντινή τιμή στην έξοδο αυτή που έχει στην διάθεσή του το μοντέλο.

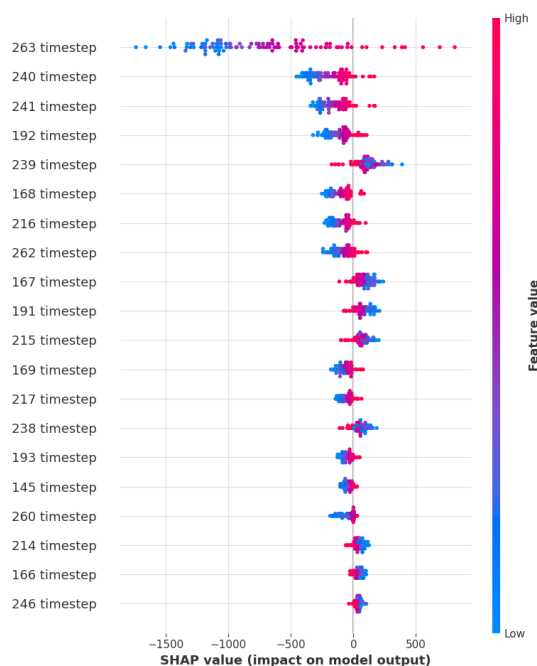


Σχήμα 4.15: Οι συντελεστές SHAP ενός δείγματος για την εικοστή τέταρτη τιμή της εξοδου του μοντέλου NBEATS

Τέλος σχεδιάζεται το διάγραμμα που αναφέρεται στην εικοστή τέταρτη (και τελευταία) έξοδο του ίδιου δείγματος. Και σε αυτή την περίπτωση, την μεγαλύτερη

συμβολή στο αποτέλεσμα την έχει η τιμή μία ώρα πριν την πρώτη πρόβλεψη. Αυτό εξηγείται από το γεγονός ότι είναι η πιο κοντινή πραγματική τιμή που έχει στην διάθεση του το μοντέλο στην εικοστή τέταρτη έξοδο. Επίσης, είναι ακριβώς μια ημέρα πριν την εν λόγω έξοδο. Αυτή την φορά, η χρονική στιγμή (timestamp) 144 έχει την δεύτερη μεγαλύτερη συμβολή. Αυτό πάλι εξηγείται από την ημερήσια εποχικότητα, αφού βρίσκεται 6 περίπου μέρες πριν την εν λόγω έξοδο.

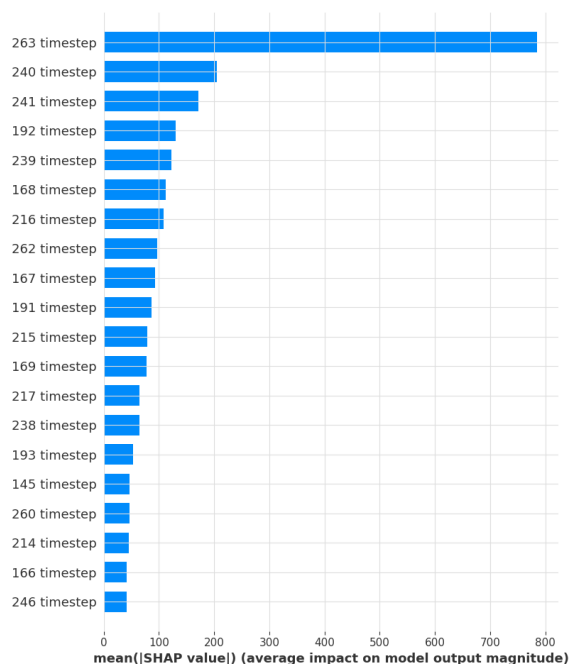
Στην συνέχεια, θα παρουσιαστούν διαγράμματα που δείχνουν τους συντελεστές SHAP όλων των δειγμάτων (100 δείγματα στο σύνολο).



Σχήμα 4.16: Οι συντελεστές SHAP όλων των δειγμάτων για την πρώτη τιμή της εξόδου του μοντέλου NBEATS

Παραπάνω εμφανίζεται το διάγραμμα που δείχνει τους συντελεστές SHAP για την πρώτη έξοδο και τα 20 πιο σημαντικά χαρακτηριστικά (αυτά δηλαδή που έχουν το μεγαλύτερο μέσο άθροισμα των απόλυτων τιμών των δεικτών SHAP). Κάθε σημείο τοποθετείται στον χ άξονα ανάλογα με την τιμή του δείκτη SHAP του αντίστοιχου δείγματος για αυτό το χαρακτηριστικό. Αν το σημείο είναι κόκκινο σημαίνει ότι η τιμή του χαρακτηριστικού είναι μεγάλη για αυτό το δείγμα σε σχέση με τα υπόλοιπα, ενώ αν είναι μπλε το αντίθετο. Μπορούμε να συμπεράνουμε, συνεπώς, το πως συσχετίζεται το κάθε χαρακτηριστικό με την τιμή της εξόδου. Για παράδειγμα, από το διάγραμμα παραπάνω παρατηρούμε ότι μεγάλες τιμές της χρονικής στιγμής (timestep) 263 επηρεάζουν θετικά την είσοδο (την κάνουν μεγαλύτερη από τον μέσο όρο), και μικρές τιμές την επηρεάζουν αρνητικά.

Η πιο σημαντική πληροφορία που γίνεται να αντληθεί από το παραπάνω διάγραμμα είναι το πόσο σημαντικό είναι το κάθε χαρακτηριστικό για την πρώτη έξοδο της πρόβλεψης του μοντέλου. Τα χαρακτηριστικά είναι τοποθετημένα στο διάγραμμα με φθίνουσα μέση τιμή της απόλυτης τιμής των δεικτών SHAP. Αυτή, όπως έχει αναφερθεί και προηγουμένως, δείχνει την σημαντικότητα της κάθε χρονικής στιγμής. Η προαναφερθείσα πληροφορία παρουσιάζεται πιο απλοποιημένα στο παρακάτω διάγραμμα:

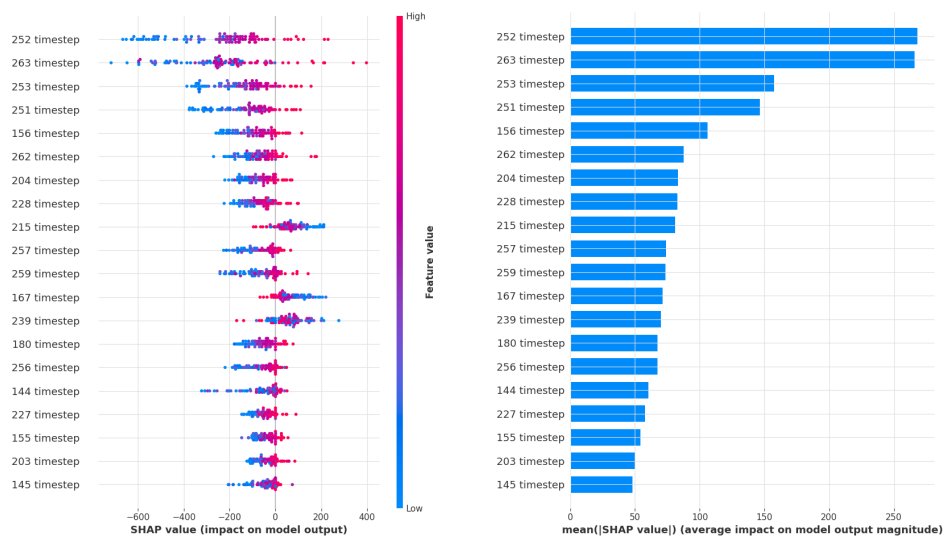


Σχήμα 4.17: Μέση τιμή των τιμών SHAP για την πρώτη τιμή της εξόδου του μοντέλου NBEATS

Παρατηρείται και πάλι ότι οι πιο σημαντικές χρονικές στιγμές βρίσκονται περίπου σε ακαίρια πολλαπλάσια των 24 ωρών μακριά από την πρώτη έξοδο (263 χρονική στιγμή: 1 ώρα πριν, 240 χρονική στιγμή: 24 ώρες πριν, 192 χρονική στιγμή 72 ώρες πριν κ.ο.κ.). Αυτό οφείλεται στην ημερήσια εποχικότητα της χρονοσειράς. Η πιο σημαντική χρονική στιγμή είναι 1 ώρα πριν την πρώτη έξοδο, που είναι αναμενόμενο, καθώς είναι η κοντινότερή της τιμή που είναι γνωστή στο μοντέλο.

Για την δέκατη τρίτη τιμή της εξόδου, παρακάτω παρουσιάζονται τα αντίστοιχα διαγράμματα:

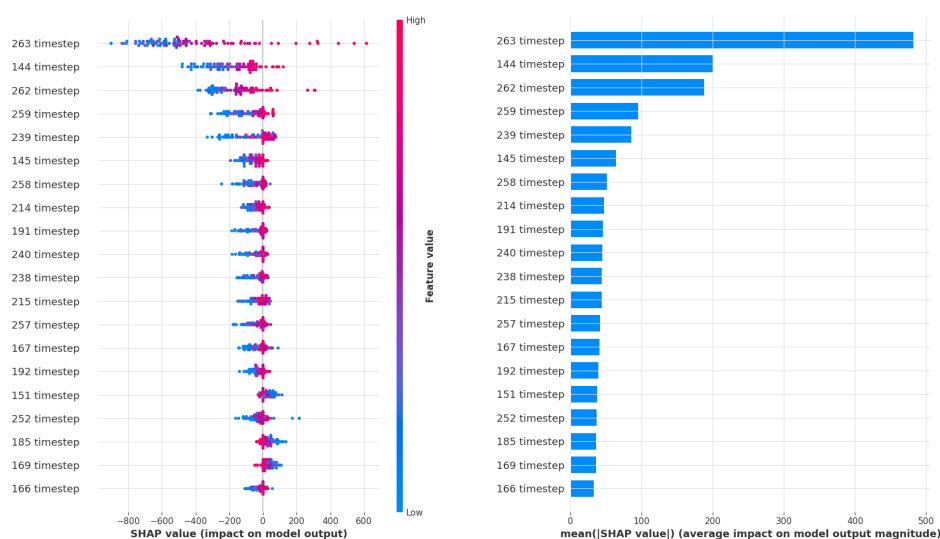
Σχήμα 4.18: Οι συντελεστές SHAP όλων των δειγμάτων και η μέση τιμή τους για την δέκατη τρίτη τιμή της εξόδου του μοντέλου NBEATS



Όπως είναι αναμενόμενο, η τιμή ακριβώς 24 ώρες πριν την δέκατη τρίτη έξοδο (χρονική στιγμή 252) είναι η πιο σημαντική λόγω εποχικότητας. Ακολουθεί πολύ κοντά και η τιμή 1 ώρα πριν την πρώτη έξοδο (χρονική στιγμή 263), επειδή είναι η κοντινότερη τιμή στην έξοδο που είναι γνωστή στο μοντέλο. Σημαντικές επίσης φαίνεται να είναι και οι τιμές κοντά στις προαναφερθείσες (χρονική στιγμή 253, 251, 262).

Τέλος θα παρουσιαστούν τα αντίστοιχα διαγράμματα για την εικοστή τέταρτη τιμή της εξόδου:

Σχήμα 4.19: Οι συντελεστές SHAP όλων των δειγμάτων και η μέση τιμή τους για την εικοστή τέταρτη τιμή της εξόδου του μοντέλου NBEATS



Εδώ και πάλι η χρονική στιγμή 263 είναι η πιο σημαντική για τους λόγους που αναλύθηκαν για την περίπτωση της δέκατης τρίτης εισόδου. Δεύτερη πιο σημαντική χρονική στιγμή, ωστόσο, είναι η 144. Αυτή απέχει περίπου 6 ημέρες από την έξοδο που αναλύεται εδώ. Αυτό δικαιολογείται από την εβδομαδιαία εποχικότητα της χρονοσειράς.

4.7.3 Αποτελέσματα προσέγγισης ενός μοντέλου ανά χρονοσειρά

Για λόγους σύγκρισης των αποτελεσμάτων με την προσέγγιση του μοντέλου πολλαπλών χρονοσειρών, εκτελέστηκε το ίδιο πείραμα με ένα μοντέλο να αντιστοιχεί σε κάθε χώρα. Οι παράμετροι του μοντέλου αυτού είναι ίδιες με του καλύτερο μοντέλο που προέκυψε από την βελτιστοποίηση υπερπαραμέτρων παραπάνω. Το κάθε μοντέλο, δηλαδή, εκπαιδεύτηκε στην κάθε χώρα ξεχωριστά με ίδιο διαχωρισμό δεδομένων. Έπειτα, αξιολογήθηκε μόνο στην χρονοσειρά της χώρας στην οποία εκπαιδεύτηκε. Στον παρακάτω πίνακα παρουσιάζονται τα αναλυτικά αποτελέσματα για κάθε χώρα:

Πίνακας 4.7: Αποτελέσματα των μοντέλων NBEATS απλών χρονοσειρών

ID χρονοσειράς	RMSE	MAPE
Βέλγιο	382.41	2.77
Τσεχία	225.85	2.05
Δανία	176.59	3.07
Εσθονία	47.9	3.63
Φινλανδία	279.87	2.07
Γαλλία	1923.83	2.45
Ελλάδα	281.86	3.34
Ουγγαρία	213.6	2.86
Ιταλία	1916.45	3.83
Λετονία	30.34	2.39
Λιθουανία	55.89	2.63
Κάτω Χώρες	719.83	4.21
Νορβηγία	458.79	2.09
Πορτογαλία	216.41	2.32
Ρουμανία	230.18	2.3
Σλοβακία	108.38	2.15
Σλοβενία	123.55	3.77
Ισπανία	1127.65	2.64
Σουηδία	620.42	2.93
Ελβετία	392.54	4.19
Μέσος Όρος	476.61	2.88

Όπως παρατηρούμε κάποιες χώρες βελτίωσαν τις μετρικές τους, ενώ κάποιες άλλες χειροτέρευσαν. Ο μέσος όρος του MAPE και του RMSE είχε ελάχιστη μείωση (MAPE 2.88 από 2.89, RMSE 476.61 από 463.5). Περαιτέρω σχολιασμός θα ακολουθήσει παρακάτω στα συγκεντρωτικά αποτελέσματα του 4.9

4.8 Αναλυτικά αποτελέσματα μοντέλου LightGBM

Συνεχίζοντας την πειραματική διαδικασία, πραγματοποιείται αξιολόγηση του μοντέλου LightGBM.

4.8.1 Βελτιστοποίηση υπερπαραμέτρων

Η εντολή που θα εκτελεστεί για να γίνει βελτιστοποίηση υπερπαραμέτρων με την βιβλιοθήκη optuna για το μοντέλο LightGBM είναι η ακόλουθη:

```
nohup mlflow run --experiment-name theo_official --entry-point exp_pipeline .
-P convert_to_local_tz=true -P cut_date_test=20210101
-P cut_date_val=20200101 -P darts_model=LightGBM -P day_first=true
-P device=gpu -P evaluate_all_ts=true -P forecast_horizon=24
-P hyperparams_entrypoint=LightGBM_thesis_multiple
-P ignore_previous_runs=t -P l_interpolation=false
-P multiple=true -P num_workers=4 -P opt_test=true
-P resolution=60 -P retrain=False -P rmv_outliers=true
-P series_csv=../../RDN/Load_Data/Multiple_Country_Series.csv
```

```
-P stride=None -P test_end_date=20220101 -P year_range=2009-2022
-P grid_search=true -P time_covs=None --env-manager=local
```

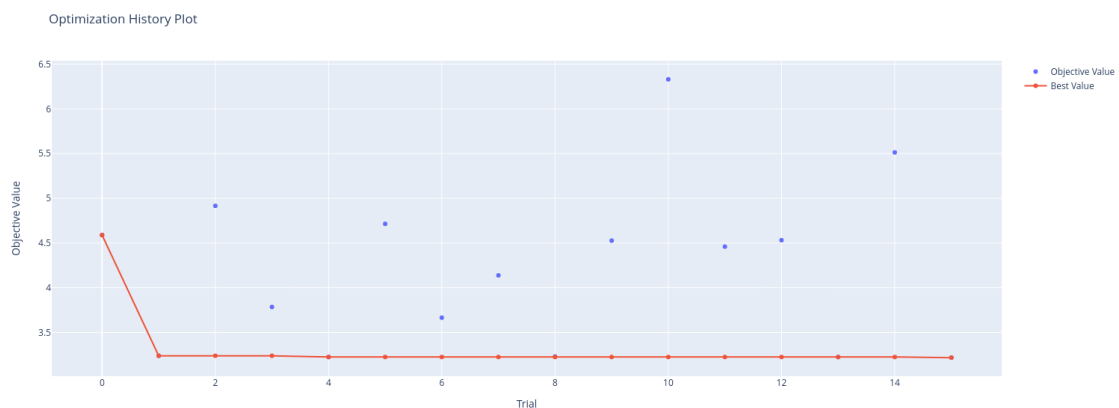
Όπως αναφέρθηκε και την περίπτωση του NBEATS, η ακριβής σημασία της κάθε παραμέτρου εξηγείται σε Παράρτημα στο τέλος. Οι παράμετροι που δεν αναφέρονται στην εντολή με `-P` διατηρούν την προεπιλεγμένη τιμή τους. Είναι άξιο αναφοράς το πεδίο `grid_search` που έχει τεθεί αληθές. Αυτό σημαίνει ότι η βιβλιοθήκη `ortuna` θα εκτελέσει αναζήτηση πλέγματος για όλες τις τιμές που θα δοθούν στο αρχείο `config_opt.yaml` για βελτιστοποίηση. Αυτό επιλέγεται εδώ, καθώς το πλήθος των πιθανών συνδυασμών είναι πολύ μικρό (16), και συνεπώς γίνεται να ελεγχθούν όλοι. Η βελτιστοποίηση θα τερματιστεί μόνη της όταν αναζητηθούν όλοι οι αναγκαίοι συνδυασμοί ή όταν ξεπεραστεί η τιμή του πεδίου `n_trials`, που εδώ λαμβάνει την προεπιλεγμένη τιμή του (100).

Παρακάτω παρουσιάζεται και η μορφή του σημείου εισόδου (`entrypoint`) `LightGBM_thesis_multiple` στο αρχείο `config_opt.yml`, το οποίο χρησιμοποιείται από την παραπάνω εντολή:

```
LightGBM_thesis_multiple:
  lags: ["range", 120, 312, 24]
  lags_past_covariates: null
  lags_future_covariates: None
  future_covs_as_tuple: true
  random_state: 0
  scale: ["list", "True", "False"]
```

Όπως έχει εξηγηθεί στην ενότητα 3, οι παράμετροι που βρίσκονται σε λίστα είναι αυτές για τις οποίες γίνεται προσπάθεια βελτιστοποίησης. Όπως φαίνεται παραπάνω τα `lags` είναι ουσιαστικά το παράθυρο πρότερης γνώσης του μοντέλου. Επομένως, θα δοκιμαστούν οι ίδιες τιμές που δοκιμάστηκαν και στο μοντέλο NBEATS. Τέλος, γίνεται έλεγχος και κατά πόσο βελτιώνει ή χειροτερεύει το αποτέλεσμα η κανονικοποίηση της χρονοσειράς. Αυτό πραγματοποιείται με την παράμετρο `scale`.

Όπως και στην περίπτωση του NBEATS, η τιμή MAPE στο σύνολο επικύρωσης (01-01-2020 έως 31-12-2020) χρησιμοποιείται για την αξιολόγηση των μοντέλων. Η εξέλιξη της διαδικασίας αναζήτησης πλέγματος φαίνεται στο διάγραμμα παρακάτω:



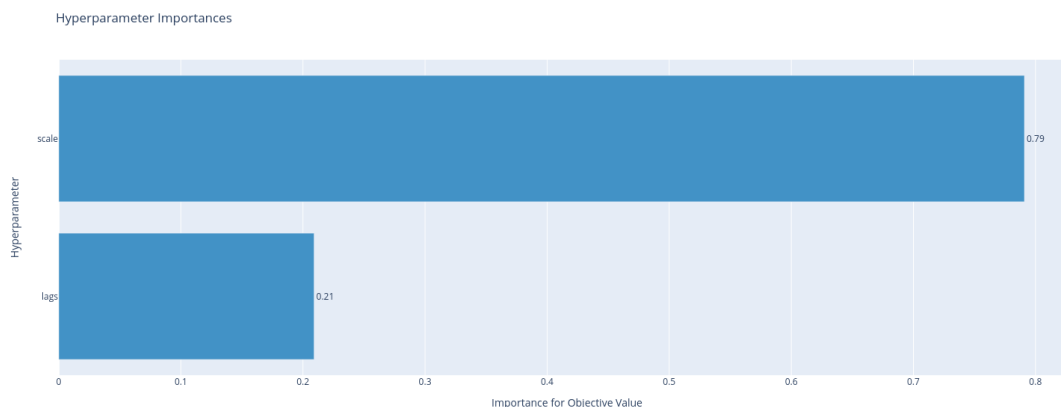
Σχήμα 4.20: Το MAPE των δεδομένων επικύρωσης για τα τρεξίματα της βελτιστοποίησης υπερπαραμέτρων του LightGBM

Η βελτιστοποίηση έτρεξε για 16 τρεξίματα. Οι υπερπαράμετροι του καλύτερου μοντέλου είναι οι εξής:

Πίνακας 4.8: Τιμές του μοντέλου LightGBM με το χαμηλότερο MAPE

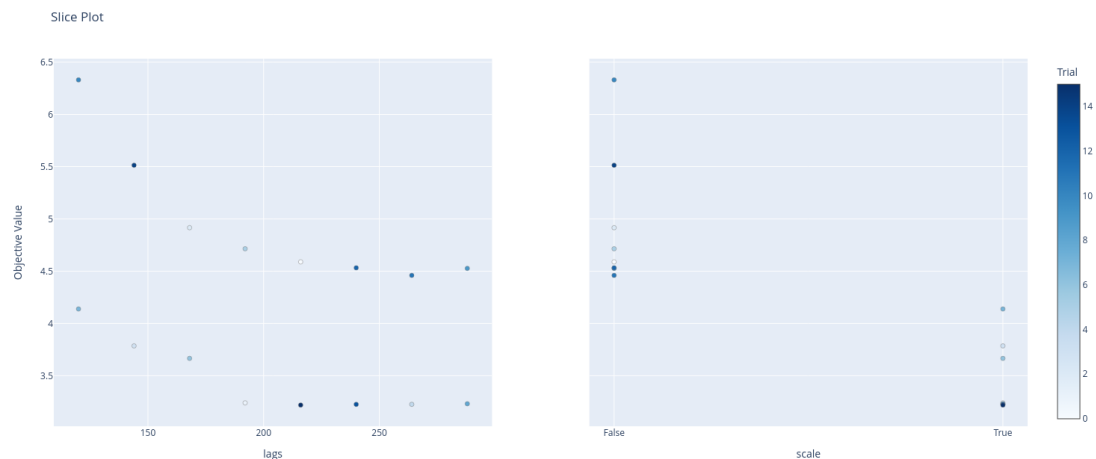
Παράμετρος	Τιμή
lags	216
lags_past_covariates	null
lags_future_covariates	None
future_covs_as_tuple	True
random_state	0
scale	True

Οι τιμές με έντονη γραφή αποτελούν εκείνες για τις οποίες εκτελέστηκε βελτιστοποίηση. Επίσης, η τιμή της μετρικής MAPE στο σύνολο δεδομένων επικύρωσης είναι 3.22 και η τιμή του RMSE 541.5. Η εκπαίδευση και η επικύρωση του καλύτερου μοντέλου κράτησε 00:04:14. Παρακάτω παρουσιάζεται και η σημαντικότητα (importance) της κάθε υπερπαραμέτρου σύμφωνα με τον αλγόριθμο της optuna που αναφέρθηκε στο κεφάλαιο 3:



Σχήμα 4.21: Το πόσο σημαντική είναι η κάθε υπερπαράμετρος που βελτιστοποιήθηκε στο τελικό αποτέλεσμα του LightGBM

Όπως παρατηρείται, το κατά πόσο η χρονοσειρά είναι κανονικοποιημένη είναι η πιο σημαντική παράμετρος (79% της απόκλισης του MAPE εξηγείται). Ακολουθεί το παράθυρο πρότερης γνώσης (εδώ lags) που εξηγεί το 29% της απόκλισης του MAPE. Τα αποτελέσματα αυτά επιβεβαιώνονται και από το παρακάτω διάγραμμα που δείχνει την τιμή mape που παρήγαγε κάθε δοκιμή και την τιμή της αντίστοιχης υπερπαραμέτρου. Υπάρχει ένα υποδιάγραμμα για κάθε υπερπαράμετρο:



Σχήμα 4.22: Αναλυτικά αποτελέσματα ανά υπερπαράμετρο του LightGBM

Στο παραπάνω διάγραμμα επιβεβαιώνεται ότι η κανονικοποίηση βελτιώνει το MAPE. Αυτό είναι αναμενόμενο, καθώς οι χρονοσειρές έχουν διαφορετικά μεγέθη. Έτσι, σε περίπτωση μη εφαρμογής κανονικοποίησης ορισμένες χρονοσειρές (αυτές με τις μεγαλύτερες τιμές) θα λαμβάνονταν υποψιν πολύ περισσότερο από άλλες.

Επιπλέον, μεγαλύτερο παράθυρο πρότερης γνώσης συνεπάγεται μικρότερο MAPE. Αυτό συμβαίνει διότι το μοντέλο λαμβάνει υπόψιν του μεγαλύτερο διάστημα πριν την κάθε πρόβλεψη. Βέβαια ελάχιστη τιμή MAPE παρουσιάζεται για τιμή παραθύρου ίση με 216. Για τιμές μεγαλύτερες από αυτό παρατηρείται ελάχιστη άνοδος του MAPE. Τέλος, είναι φανερό στο διάγραμμα του παραθύρου πρότερης γνώσης οι 2 καμπύλες, καθεμία από τις οποίες σχετίζεται με το αν έγινε κανονικοποίηση ή όχι.

4.8.2 Αξιολόγηση μοντέλου και SHAP

Από την βελτιστοποίηση υπερπαραμέτρων έχει προκύψει το καλύτερο μοντέλο του οποίου οι παράμετροι παρουσιάστηκαν στον πίνακα 4.8. Για αυτό το μοντέλο, το οποίο βρίσκεται αποθηκευμένο στον διακομιστή του MLflow, εκτελείται η ακόλουθη εντολή με σκοπό την αξιολόγησή του για όλες τις χρονοσειρές στα δεδομένα αξιολόγησης (01-01-2021 έως 01-01-2022):

```
nohup mlflow run . -e eval -P mode=remote
-P series_uri=$series_uri -P evaluate_all_ts=true -P model_type=pkl
-P scaler_uri=$scaler_uri -P cut_date_test=20210101 -P test_end_date=20220101
-P model_uri=$model_uri -P forecast_horizon=24 -P stride=None -P retrain=False
-P input_chunk_length=216 -P output_chunk_length=24 -P cut_date_val=20200101
-P day_first=true -P multiple=true -P resolution=60 -P analyze_with_shap=false
--env-manager=local --experiment-name=theo_official
```

Η παραπάνω εντολή εκτελεί μόνο το στάδιο της αξιολόγησης. Για τον σκοπό αυτό έχει προστεθεί η επιλογή `-e eval`. Είναι ακριβώς ίδια με την αντίστοιχη του NBEATS, εκτός από τις εξής διαφορές:

- Στην θέση των `$series_uri`, `$scaler_uri`, `$model_uri` θα μπουν τα `uris` της χρονοσειράς, του αντικείμενου που πραγματοποιεί την κανονικοποίηση, και του μοντέλου αντίστοιχα, τα οποία έχουν αποθηκευτεί στο τρέξιμο της βελτιστοποίησης υπερπαραμέτρων του LightGBM (και όχι του NBEATS)

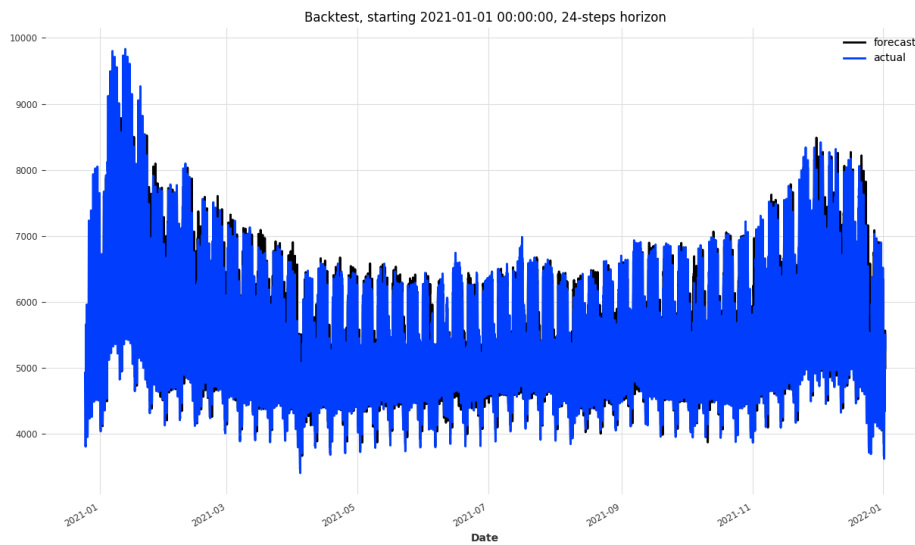
- Το `input_chunk_length` (παράθυρο πρότερης γνώσης) θα είναι ίσο με 216, που είναι το παράθυρο του μοντέλου το οποίο εξετάζεται.
- Ο τύπος του μοντέλου εδώ είναι `rkl` και όχι `pl`, καθώς το LightGBM αποθηκεύεται σε μορφή αρχείου `pickle`.

Παρακάτω παρουσιάζονται τα αποτελέσματα της αξιολόγησης του μοντέλου LightGBM για όλες τις χρονοσειρές στα δεδομένα αξιολόγησης, μαζί με τον μέσο όρο τους που αποθηκεύεται στον διακομιστή του MLflow:

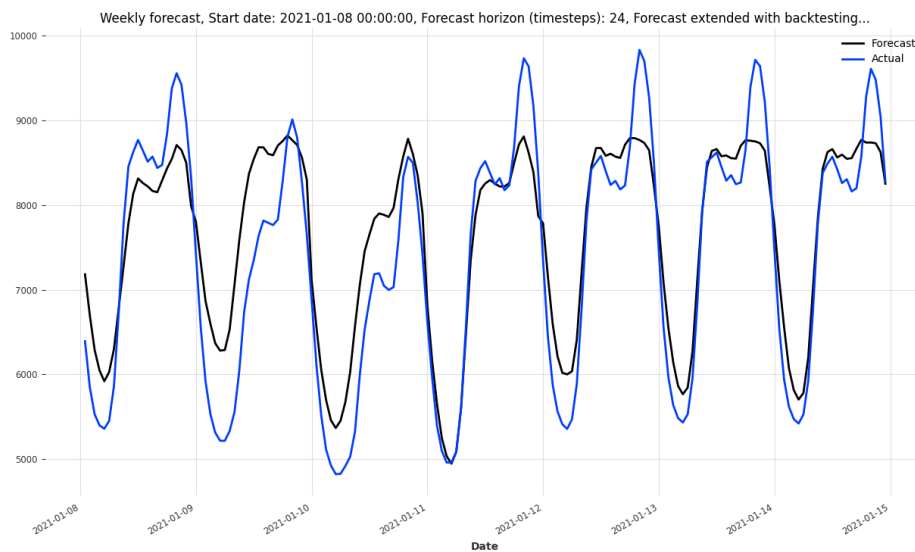
Πίνακας 4.9: Αποτελέσματα του μοντέλου LightGBM

ID χρονοσειράς	RMSE	MAPE
Βέλγιο	415.7	2.94
Τσεχία	336.35	2.85
Δανία	195.77	3.35
Εσθονία	56.34	4.12
Φινλανδία	393.02	2.93
Γαλλία	2653.94	3.46
Ελλάδα	349.11	4.1
Ουγγαρία	265.35	3.5
Ιταλία	1912.06	3.4
Λετονία	36.57	2.68
Λιθουανία	70.04	3.06
Κάτω Χώρες	799.88	4.33
Νορβηγία	579.17	2.56
Πορτογαλία	262.35	2.86
Ρουμανία	261.78	2.59
Σλοβακία	129.69	2.43
Σλοβενία	119.84	3.97
Ισπανία	1292.16	2.79
Σουηδία	776.03	3.61
Ελβετία	425.39	4.5
Μέσος Όρος	566.5	3.3

Για όλες τις χώρες και τις μετρικές το αποτέλεσμα είναι καλύτερο από τις Naive μεθόδους. Σε σύγκριση με το μοντέλο NBETS, είναι χειρότεο για όλες τις χώρες και μετρικές, με την εξαίρεση των Κάτω Χωρών (Για αμφότερες τις μετρικές), και της Σλοβενίας (μόνο για το RMSE). Ο μέσος όρος των τιμών αυτών είναι ομοίως καλύτερος από τα Naive και χειρότερος από το NBEATS. Περαιτέρω σύγκριση των 2 αυτών μοντέλων θα πραγματοποιηθεί στην ενότητα 4.9. Στον διακομιστή του MLflow αποθηκεύονται επιπλέον και διαγράμματα που δείχνουν τις προβλέψεις του μοντέλου για όλες τις χώρες. Ενδεικτικά παρατίθενται παρακάτω τα διαγράμματα για την Πορτογαλία:



Σχήμα 4.23: Πρόβλεψη για το 2021 του LightGBM για την χρονοσειρά της Πορτογαλίας



Σχήμα 4.24: Πρόβλεψη για την δεύτερη εβδομάδα του 2021 για την χρονοσειρά της Πορτογαλίας απο το μοντέλο LightGBM

Στα διαγράμματα, όπως και για την περίπτωση του μοντέλου NBEATS, εμφανίζεται με μπλε γραμμή η πρόβλεψη του μοντέλου, και με μαύρη η πραγματική τιμή της χρονοσειράς. Το πρώτο διάγραμμα αναφέρεται σε όλη την περίοδο αξιολόγησής (01-01-2021 έως 01-01-2022), ενώ το δεύτερο στην δεύτερη εβδομάδα αυτής (08-01-2021 έως 15-01-2021). Όπως παρατηρείται γενικά η πρόβλεψη του μοντέλου είναι κοντά στην πραγματική μορφή της χρονοσειράς, ωστόσο έχει αρκετά μεγαλύτερη διαφορά από την πραγματική τιμή σε σχέση με το NBEATS(διάγραμμα 4.11). Το μοντέλο, ακόμα, καταφέρνει να αποτυπώσει καλά την μορφή της ημερήσιας καμπύλης κατανάλωσης φορτίου, ωστόσο για την εβδομάδα που αναλύεται εδώ πολλές φορές γίνεται υπερε-

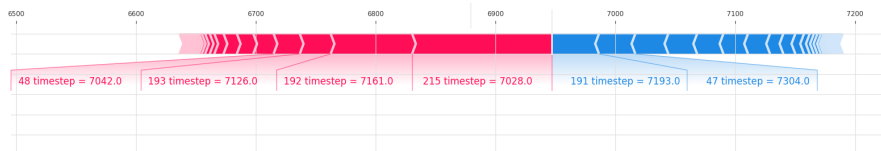
κτίμηση ή και υποεκτίμηση της κατανάλωσης.

Με σκοπό την καλύτερη κατανόηση της σημασίας κάθε χρονικού σημείου του παραθύρου πρότερης γνώσης, εκτελείται εξήγηση του αποτελέσματος του μοντέλου με την βιβλιοθήκη SHAP, όπως έγινε και για το NBEATS. Για τον σκοπό αυτόν, εκτελείται η εντολή:

```
nohup mlflow run . -e eval -P mode=remote -P series_uri=$series_uri
-P evaluate_all_ts=false -P model_type=pkl -P scaler_uri=$scaler_uri
-P cut_date_test=20210101 -P test_end_date=20220101
-P model_uri=$model_uri -P forecast_horizon=24 -P stride=None -P retrain=False
-P input_chunk_length=216 -P output_chunk_length=24 -P cut_date_val=20200101
-P day_first=true -P eval_method=ts_code -P eval_series=PT -P multiple=true
-P resolution=60 -P analyze_with_shap=true -P size=100 --env-manager=local
--experiment-name=theo_official
```

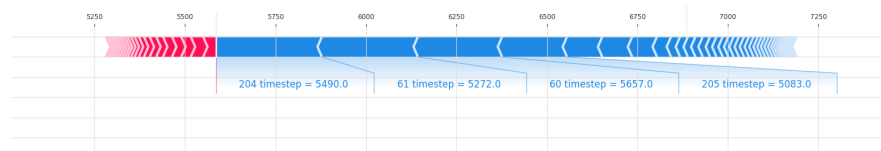
Όπως και προηγουμένως, η παραπάνω εντολή εκτελεί μόνο το στάδιο της αξιολόγησης. Για τον σκοπό αυτό έχει προστεθεί η επιλογή `-e eval`. Αντιθέτως με την προηγούμενη εντολή, δεν θα πραγματοποιηθεί αξιολόγηση σε όλες τις χρονοσειρές. Αντιθέτως, αφού θέτουμε `eval_series=PT` και `evaluate_all_ts=false`, η αξιολόγηση γίνεται μόνο για την Πορτογαλία. Η μέθοδος SHAP μπορεί να εφαρμοστεί σε μόνο μια χρονοσειρά αξιολόγησης. Επίσης, θέτοντας `size=100`, επιλέγονται 100 τυχαία δείγματα από την χρονοσειρά αξιολόγησης της Πορτογαλίας. Παρακάτω παρουσιάζονται τα διαγράμματα που παράγονται από αυτή την διαδικασία.

Αρχικά επιλέγεται το δείγμα για το οποίο θα σχεδιαστούν τα αναλυτικά διαγράμματα των τιμών των συντελεστών SHAP. Αυτό που επιλέγεται είναι το δείγμα 24 που ξεκινά την 2021-12-13 13:00:00. Για αυτό το δείγμα θα σχεδιαστούν τρία τέτοια διαγράμματα.



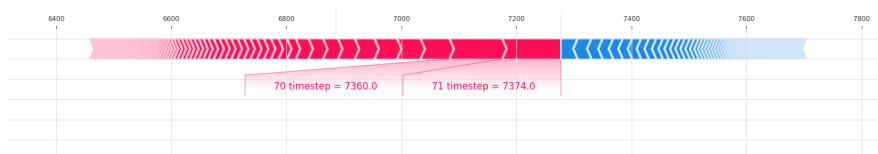
Σχήμα 4.25: Οι συντελεστές SHAP ενός δείγματος για την πρώτη τιμή της εξόδου του μοντέλου LightGBM

Για την πρώτη τιμή της εξόδου σημαντικότερες είναι οι τιμές 215 (1 ώρα πριν) και 192 (24 ώρες πριν). Αυτό συμβαίνει λόγω της ημερήσιας εποχικότητας της χρονοσειράς. Αξιοσημείωτο είναι ότι εδώ αυτές οι δυο χρονικές στιγμές (timestamps) έχουν αρνητική επιρροή στην έξοδο (το βελάκι τους είναι κόκκινο). Αυτό σημαίνει ότι η τιμή τους έχει αρνητική επίδραση στην έξοδο του μοντέλου



Σχήμα 4.26: Οι συντελεστές SHAP ενός δείγματος για την δέκατη τρίτη τιμή της εξόδου του μοντέλου LightGBM

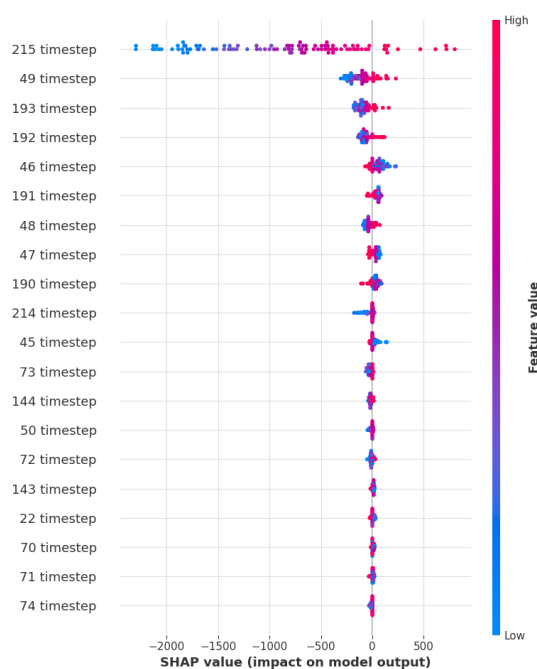
Ομοίως για την δέκατη τρίτη είσοδο, πιο σημαντικές είναι οι τιμές που απέχουν 24 ώρες από αυτήν (χρονική στιγμή 204), και μια εβδομάδα (χρονική στιγμή 60). Αυτό οφείλεται στην ημερήσια και στην εβδομαδιαία εποχικότητα της χρονοσειράς.



Σχήμα 4.27: Οι συντελεστές SHAP ενός δείγματος για την εικοστή τέταρτη τιμή της εξόδου του μοντέλου LightGBM

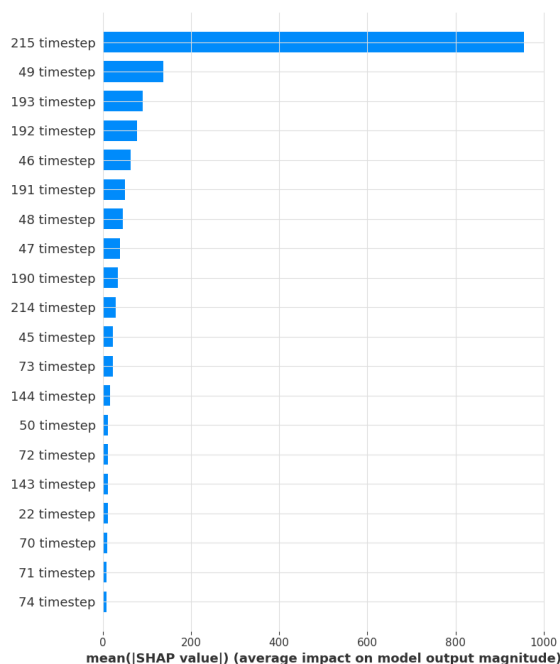
Και εδώ οι σημαντικότερες χρονικές στιγμές απέχουν περίπου μια εβδομάδα από την έξοδο (Η 71 χρονική στιγμή απέχει ακριβώς μια εβδομάδα από την εικοστή τέταρτη τιμή της εξόδου, ενώ η 70 μια εβδομάδα και μια ώρα). Η εξήγηση είναι ίδια με προηγουμένως.

Παρακάτω παρουσιάζεται αναλυτικά η τιμή SHAP της πρώτης εξόδου για τις 20 πιο σημαντικές χρονικές στιγμές και για όλα τα δείγματα.



Σχήμα 4.28: Οι συντελεστές SHAP όλων των δειγμάτων για την πρώτη τιμή της εξόδου του μοντέλου LightGBM

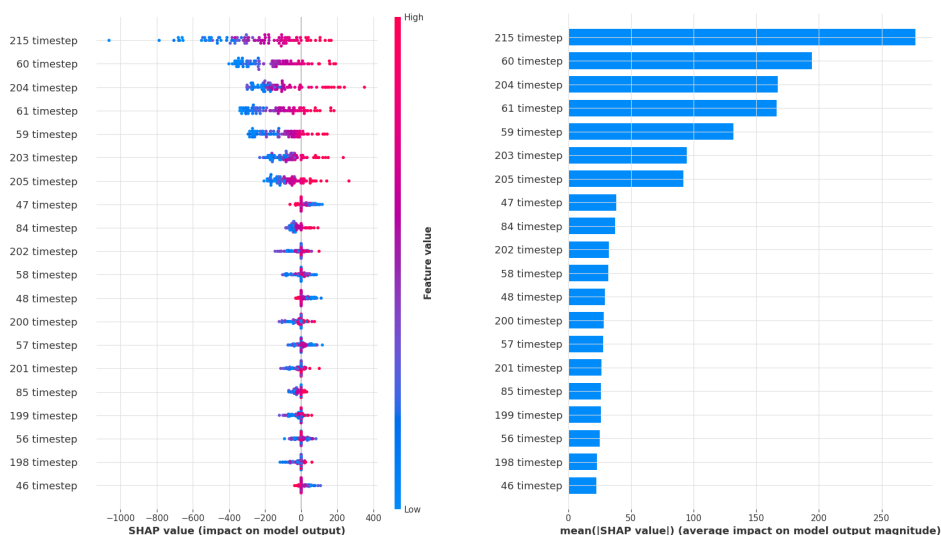
Όπως ήταν αναμενόμενο, πιο σημαντικές είναι οι χρονικές στιγμές μία ώρα και περίπου μια εβδομάδα (για την χρονική στιγμή 48, 6 ημέρες και 23 ώρες) μακριά από την εν λόγω έξοδο. Αυτό οφείλεται στην εποχικότητα της χρονοσειράς. Παρακάτω, παρουσιάζεται και το διάγραμμα της μέσης τιμής της απόλυτης τιμής των συντελεστών SHAP για κάθε χρονική στιγμή. Αυτό δείχνει πιο συνοπτικά τα ίδια αποτελέσματα που αναφέρθηκαν προηγουμένως.



Σχήμα 4.29: Μέση τιμή των τιμών SHAP για την πρώτη τιμή της εξόδου του μοντέλου LightGBM

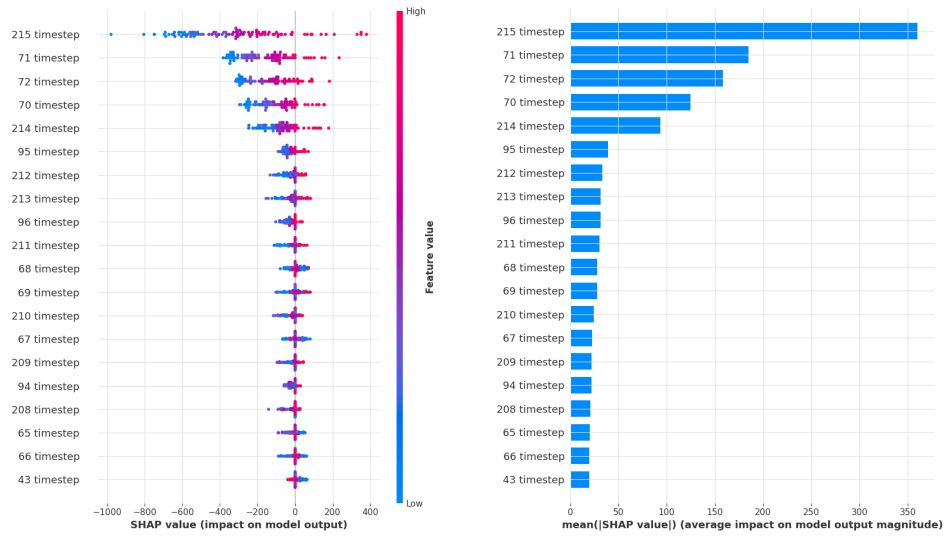
Παρακάτω παρουσιάζονται τα αντίστοιχα διαγράμματα για την δέκατη τρίτη έξοδο. Και πάλι η πρώτη τριάδα των σημαντικότερων χρονικών στιγμών είναι η κοντινότερη χρονική στιγμή των πραγματικών δεδομένων που έχει το μοντέλο στην διάθεσή του (215), η χρονική στιγμή μια εβδομάδα πριν την εν λόγω έξοδο (60), και 24 ώρες πριν (204). Η αιτιολόγηση είναι ίδια με παραπάνω.

Σχήμα 4.30: Οι συντελεστές SHAP όλων των δειγμάτων και η μέση τιμή τους για την δέκατη τρίτη τιμή της εξόδου του μοντέλου LightGBM



Τέλος, τα διαγράμματα για την εικοστή τέταρτη έξοδο παρουσιάζονται παρακάτω. Και πάλι οι χρονικές στιγμές περίπου μια εβδομάδα πριν (70, 71, 72) και οι τελευταία γνωστή από το μοντέλο (215) είναι οι πιο σημαντικές.

Σχήμα 4.31: Οι συντελεστές SHAP όλων των δειγμάτων και η μέση τιμή τους για την εικοστή τέταρτη τιμή της εξόδου του μοντέλου LightGBM



4.8.3 Αποτελέσματα προσέγγισης ενός μοντέλου ανά χρονοσειρά

Για λόγους σύγκρισης των αποτελεσμάτων με την προσέγγιση του μοντέλου πολλών χρονοσειρών, εκτελέστηκε και για το LightGBM το ίδιο πείραμα με ένα μοντέλο να αντιστοιχεί σε κάθε χώρα. Οι παράμετροι διατηρήθηκαν ίδιες με αυτές της βελτιστοποίησης υπερπαραμέτρων παραπάνω. Στον παρακάτω πίνακα παρουσιάζονται τα αναλυτικά αποτελέσματα για κάθε χώρα:

Πίνακας 4.10: Αποτελέσματα των μοντέλων απλών χρονοσειρών LightGBM

ID χρονοσειράς	RMSE	MAPE
Βέλγιο	432.59	3.02
Τσεχία	317.03	2.68
Δανία	219.2	3.8
Εσθονία	54.53	4.06
Φινλανδία	329.8	2.42
Γαλλία	2461.56	3.04
Ελλάδα	332.04	3.76
Ουγγαρία	258.57	3.53
Ιταλία	1819.53	3.33
Λετονία	37.63	2.78
Λιθουανία	71.41	3.24
Κάτω Χώρες	824.98	4.51
Νορβηγία	498.79	2.28
Πορτογαλία	278.26	2.86
Ρουμανία	251.38	2.47
Σλοβακία	127.8	2.37
Σλοβενία	115.14	3.8
Ισπανία	1192.33	2.62
Σουηδία	698.71	3.2
Ελβετία	402.64	4.24
Μέσος Όρος	536.2	3.2

Όπως παρατηρούμε κάποιες χώρες βελτίωσαν τις μετρικές τους, ενώ κάποιες άλλες χειροτέρευσαν. Ο μέσος όρος του MAPE και του RMSE είχε μικρή μείωση (MAPE 3.2 από 3.3, RMSE 536.2 από 566.5). Περαιτέρω σχολιασμός θα ακολουθήσει παρακάτω στα συγκεντρωτικά αποτελέσματα του 4.9

4.9 Συγκεντρωτικά αποτελέσματα όλων των μοντέλων

Στον παρακάτω πίνακα παρουσιάζονται συνοπτικά τα μέσα αποτελέσματα όλων των μοντέλων που αναλύθηκαν παραπάνω:

Πίνακας 4.11: Συνοπτικά αποτελέσματα όλων των μοντέλων

Είδος Μοντέλου	Χρονοσειρά	RMSE	MAPE
Naive μιας ημέρας	Δεδομένα αξιολόγησης	1002.23	5.78
Naive μιας εβδομάδας	Δεδομένα αξιολόγησης	994.62	5.55
NBEATS πολλαπλών χρονοσειρών	Δεδομένα επικύρωσης	387.7	2.45
	Δεδομένα αξιολόγησης	463.5	2.89
NBEATS απλών χρονοσειρών	Δεδομένα αξιολόγησης	476.61	2.88
LightGBM πολλαπλών χρονοσειρών	Δεδομένα επικύρωσης	541.5	3.22
	Δεδομένα αξιολόγησης	566.5	3.3
LightGBM απλών χρονοσειρών	Δεδομένα αξιολόγησης	536.2	3.2

Όπως παρατηρείται το NBEATS είναι το καλύτερο μοντέλο για όλες τις μετρικές τόσο στα δεδομένα αξιολόγησης, όσο και στα δεδομένα επικύρωσης. Ωστόσο, το LightGBM έχει ορισμένα πλεονεκτήματα. Αρχικά, οι μετρικές του χειροτέρευαν λιγότερο στα δεδομένα αξιολόγησης από ότι του NBEATS. Αυτό σημαίνει ότι το μοντέλο αυτό κατάφερε να γενικεύσει καλύτερα τα αποτελέσματά του σε σύγκριση με το NBEATS. Επιπλέον, το LightGBM είχε πολύ χαμηλότερο χρόνο εκπαίδευσης και επικύρωσης από το NBEATS (00:04:14 και 01:57:07 αντίστοιχα). Αυτό σημαίνει χαμηλότερη κατανάλωση υπολογιστικών πόρων, πράγμα το οποίο αυξάνει την βιωσιμότητα των διαδικασιών που εκτελούνται.

Επιπλέον, παρατηρούμε μικρή χειροτέρευση των μετρικών στα μοντέλα πολλαπλών χρονοσειρών σε σχέση με τα απλά. Αυτό μπορεί να οφείλεται σε κάποιες διαφορές μεταξύ των χρονοσειρών των χωρών (πχ οι βόρειες χώρες καταναλώνουν λιγότερο το καλοκαίρι από τις νότιες, καθώς οι τελευταίες έχουν υψηλές θερμοκρασίες και επομένως την ανάγκη κλιματισμού). Ωστόσο, η διαφορά αυτή είναι πολύ μικρή (0.01 στο MAPE του NBEATS και 0.1 του LightGBM). Επομένως, είναι προτιμότερη η χρήση των μοντέλων πολλαπλών χρονοσειρών, αφού αυτή συνεπάγεται κατανάλωση λιγότερων πόρων (αποθήκευση ενός μοντέλου αντί για πολλαπλά). Το παραπάνω συμπέρασμα ισχύει ειδικά για το NBEATS, που έχει αρκετά μεγάλη κατανάλωση πόρων.

Κεφάλαιο 5

Συμπεράσματα και Μελλοντικές προεκτάσεις

5.1 Συμπεράσματα και Επίλογος

Στην παρούσα εργασία αναπτύχθηκε εφαρμογή με σκοπό την διευκόλυνση της πρόβλεψης κατανάλωσης ηλεκτρικής ενέργειας με χρήση τεχνικών τόσο μηχανικής μάθησης (Machine Learning), όσο και βαθιάς μάθησης (Deep Learning). Η εφαρμογή αυτή αυτοματοποιεί ένα μεγάλο μέρος του κύκλου ζωής μηχανικής μάθησης, και υποστηρίζει ποικίλες λειτουργίες οι οποίες αναλύθηκαν στο κεφάλαιο 3:

- Επεξεργασία δεδομένων, εκπαίδευση και αξιολόγηση μοντέλων με μια εντολή
- Δυνατότητα βελτιστοποίησης υπερπαραμέτρων με στατιστικές μεθόδους (tpe) και αναζήτηση πλέγματος (grid search)
- Δυνατότητα εξήγησης σημασίας του κάθε χαρακτηριστικού του μοντέλου με χρήση SHAP
- Υποστήριξη πολλαπλών ή/και πολυμεταβλητών χρονοσειρών

Η εφαρμογή αυτή δοκιμάζεται στο σύνολο δεδομένων [31], και στο πρόβλημα των πολλαπλών χρονοσειρών. Συγκεκριμένα, το σύνολο δεδομένων αποτελείται από 20 χρονοσειρές (Βέλγιο, Τσεχία, Δανία, Εσθονία, Φινλανδία, Γαλλία, Ελλάδα, Ουγγαρία, Ιταλία, Λετονία, Λιθουανία, Κάτω Χώρες, Νορβηγία, Πορτογαλία, Ρουμανία, Σλοβακία, Σλοβενία, Ισπανία, Σουηδία και Ελβετία) οι οποίες χωρίζονται σε σύνολα εκπαίδευσης (2015-04-09 23:00:00 έως 31-12-2019), επικύρωσης (01-01-2020 έως 31-01-2020) και αξιολόγησης (01-01-2021 έως 01-01-2022). Στις παρακάτω χρονοσειρές εκτελείται επεξεργασία, η οποία περιλαμβάνει τον εντοπισμό ακραίων τιμών (outlier detection) και την αντικατάσταση άγνωστων τιμών (imputation).

Στην παρούσα εργασία δοκιμάζονται 2 μοντέλα, ένα βαθιάς μάθησης (NBEATS), και ένα μηχανικής μάθησης (LightGBM). Σε αυτά τα μοντέλα εκτελείται βελτιστοποίηση υπερπαραμέτρων με σκοπό την κατανόηση της σημασίας της κάθε υπερπαραμέτρου, καθώς και την επιλογή του καλύτερου μοντέλου. Τα συμπεράσματα που προέκυψαν από αυτή την διαδικασία είναι τα εξής:

- Για αμφότερες τις μεθόδους προκύπτει ότι οι σημαντικότερες υπερπαραμέτροι είναι η κανονικοποίηση ή όχι της χρονοσειράς, και το παράθυρο πρότερης γνώσης, όπως παρουσιάζεται στα διαγράμματα 4.21 και 4.9. Για το NBEATS, ακολουθούν οι `um_layers`, `num_blocks` και `num_stacks` με πολύ μικρότερη συμβολή στο MAPE.
- Για αμφότερα τα μοντέλα, η επιλογή της κανονικοποίησης του μοντέλου προκαλεί μείωση στο MAPE. Αυτό δικαιολογείται από τα διαφορετικά μεγέθη των χρονοσειρών
- Για αμφότερα τα μοντέλα, όσο μεγαλώνει το παράθυρο πρότερης γνώσης, τόσο μικραίνει το MAPE. Στην περίπτωση του NBEATS, η τιμή του καλύτερου μοντέλου είναι 264, ενώ για το LightGBM η αντίστοιχη τιμή είναι 216,

- Για το NBEATS μικρές τιμές της παραμέτρου `num_blocks` (τιμή βέλτιστου μοντέλου 1) και της `num_layers` (τιμή βέλτιστου μοντέλου 2) οδηγούν σε μικρότερο MAPE. Αντιθέτως, η παράμετρος `num_stacks` έχει ελάχιστο την τιμή 6, και το MAPE αυξάνεται για τιμές πιο μεγάλες και πιο μικρές από αυτό.

Έπειτα, πραγματοποιήθηκε αξιολόγηση των μοντέλων στο σύνολο αξιολόγησης, και εκτελέστηκε η μέθοδος SHAP για την εξήγηση της σημασίας της κάθε χρονικής στιγμής του παραθύρου πρότερης γνώσης στον καθορισμό της εξόδου. Για την μέθοδο SHAP, για αμφότερα τα μοντέλα σημαντικές χρονικές στιγμές ήταν η τελευταία διαθέσιμη πραγματική τιμή, οι χρονικές στιγμές περίπου μια ημέρα πριν την έξοδο, και οι χρονικές στιγμές περίπου μια εβδομάδα πριν την έξοδο, όπως παρουσιάζεται στις υποενότητες 4.8.2 και 4.7.2. Αυτό οφείλεται στην ημερήσια και στην εβδομαδιαία εποχικότητα του συνόλου δεδομένων.

Όσο για την αξιολόγηση των μοντέλων, το NBEATS είναι το μοντέλο με το μικρότερο MAPE και RMSE στα δεδομένα αξιολόγησης και στα δεδομένα επικύρωσης. Επομένως, αν η εφαρμογή έχει ως κύριο μέλημα την υψηλή ακρίβεια του μοντέλου, αυτή είναι η επιλογή που προτείνεται.

Το LightGBM από την άλλη εμφανίζει μικρότερη αύξηση των μετρικών του στα δεδομένα αξιολόγησης από ότι το NBEATS. Αυτό σημαίνει ότι το LightGBM γενίκευσε καλύτερα τα αποτελέσματά του σε σύγκριση με το NBEATS. Επιπλέον, το LightGBM είχε πολύ χαμηλότερο συνολικό χρόνο εκπαίδευσης και επικύρωσης από το NBEATS. Αυτό σημαίνει χαμηλότερη κατανάλωση υπολογιστικών πόρων, που αυξάνει την βιωσιμότητα των μεθόδων που εφαρμόζονται. Αν, επομένως, η εφαρμογή απαιτεί ένα σχετικά "ελαφρύ" και ευέλικτο μοντέλο, που επανεκπαιδεύεται γρήγορα σε καινούργια δεδομένα και δεν καταναλώνει πολλούς πόρους, το LightGBM είναι η επιλογή που προτείνεται.

Έπειτα, εκτελείται για τις ίδιες παραμέτρους αξιολόγησης μοντέλων που έχουν εκπαιδευτεί στην κάθε χρονοσειρά ξεχωριστά. Παρατηρούμε μικρή χειροτέρευση των μετρικών στα μοντέλα πολλαπλών χρονοσειρών σε σχέση με τα απλά. Αυτό οφείλεται σε κάποιες διαφορές μεταξύ της μορφής των χρονοσειρών των χωρών (πχ οι βόρειες χώρες καταναλώνουν λιγότερο το καλοκαίρι από τις νότιες, καθώς οι τελευταίες έχουν υψηλές θερμοκρασίες και επομένως την ανάγκη κλιματισμού). Ωστόσο, η διαφορά αυτή είναι ανεπαίσθητη. Επομένως, είναι καλύτερο να χρησιμοποιούνται μοντέλα πολλαπλών χρονοσειρών, αφού αυτό συνεπάγεται κατανάλωση λιγότερων πόρων (αποθήκευση ενός μοντέλου αντί για πολλαπλά). Το παραπάνω ισχύει ειδικά για το NBEATS, που έχει αρκετά μεγάλη κατανάλωση πόρων.

Με το πρόβλημα των πολλαπλών χρονοσειρών, η εφαρμογή που αναπτύχθηκε στα πλαίσια τη παρούσας εργασίας απέδειξε την χρησιμότητά της. Θα μπορούσε να χρησιμοποιηθεί για πληθώρα προβλημάτων πρόβλεψης χρονοσειρών, σε τομείς ακόμα πιο γενικούς από την βραχυπρόθεσμη πρόβλεψη ζήτησης ηλεκτρικής ενέργειας. Το μόνο που απαιτείται από τον χρήστη είναι η μετατροπή των δεδομένων στην κατάλληλη μορφή, και η ορθή επιλογή των παραμέτρων.

5.2 Μελλοντικές προεκτάσεις

Αρχικά, περισσότερα μοντέλα θα μπορούσαν να προστεθούν (πχ το perceptron πολλαπλών επιπέδων MLP), μαζί με την δυνατότητα ensembling των μοντέλων. Αυτές οι προσθήκες θα μπορούσαν να προσφέρουν μεγαλύτερη πληθώρα εργαλείων στον χρήστη, και την δυνατότητα να επιτύχει ακόμα καλύτερα αποτελέσματα. Οι μέθο-

δοι προεπεξεργασίας θα μπορούσαν επιπλέον να γενικευτούν για χρονοσειρές άλλων μορφών, με την ταυτόχρονη προσθήκη εναλλακτικών μεθόδων από τις ήδη υπάρχουσες. Η ανίχνευση ακραίων τιμών και η αντικατάσταση άγνωστων τιμών αποτελούν 2 μεθόδους που θα μπορούσαν να έχουν εναλλακτικές.

Επιπλέον, η εφαρμογή θα μπορούσε να γίνει κομμάτι ενός συστήματος το οποίο θα προσέφερε στον χρήστη γραφικό περιβάλλον (frontend) προκείμενου να πραγματοποιηθούν οι προαναφερθείσες διεργασίες.

Ακόμα, θα μπορούσαμε να ενσωματώσουμε την εφαρμογή με άλλες βάσεις δεδομένων, προκειμένου να λαμβάνονται από εκεί οι χρονοσειρές απευθείας.

Τέλος, όσον αφορά το πρόβλημα πολλαπλών χρονοσειρών που αναλύθηκε παραπάνω, θα μπορούσε να πραγματοποιηθεί επιλογή ενός τμήματος των προαναφερθεισών χωρών με συσταδοποίηση (clustering), το οποίο θα περιείχε τις χώρες με παρόμοια χαρακτηριστικά. Αυτό θα μπορούσε να αυξήσει την αποτελεσματικότητα των μοντέλων.

Αναφορές

- [1] *SHAP documentation*. διεύθυν.: <https://shap.readthedocs.io/en/latest/> (επίσκεψη 05/03/2023).
- [2] P. Montero-Manso και R. J. Hyndman, «Principles and algorithms for forecasting groups of time series: Locality and globality», *International Journal of Forecasting*, τόμ. 37, σσ. 1632–1653, 4 Οκτ. 2021, issn: 01692070. doi: 10.1016/j.ijforecast.2021.03.004.
- [3] S. Alla και S. K. Adari, «What Is MLOps?», *Beginning MLOps with MLFlow*, σσ. 79–124, 2021. doi: 10.1007/978-1-4842-6549-9_3.
- [4] *MLflow documentation*. διεύθυν.: <https://mlflow.org/docs/latest/index.html> (επίσκεψη 03/03/2023).
- [5] L. Wei και Z. Zhen-gang, «Based on Time Sequence of ARIMA Model in the Application of Short-Term Electricity Load Forecasting», στο *2009 International Conference on Research Challenges in Computer Science*, Δεκ. 2009, σσ. 11–14. doi: 10.1109/ICRCCS.2009.12.
- [6] H. Musbah και M. El-Hawary, «SARIMA Model Forecasting of Short-Term Electrical Load Data Augmented by Fast Fourier Transform Seasonality Detection», στο *2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE)*, IEEE, Μάι. 2019, σσ. 1–4, isbn: 978-1-7281-0319-8. doi: 10.1109/CCECE.2019.8861542.
- [7] H. Hippert, C. Pedreira και R. Souza, «Neural networks for short-term load forecasting: a review and evaluation», *IEEE Transactions on Power Systems*, τόμ. 16, αρθμ. 1, σσ. 44–55, 2001. doi: 10.1109/59.910780.
- [8] S. Bouktif, A. Fiaz, A. Ouni και M. Serhani, «Optimal Deep Learning LSTM Model for Electric Load Forecasting using Feature Selection and Genetic Algorithm: Comparison with Machine Learning Approaches †», *Energies*, τόμ. 11, σ. 1636, 7 Ιούν. 2018, issn: 1996-1073. doi: 10.3390/en11071636.
- [9] S. Makridakis, E. Spiliotis και V. Assimakopoulos, «The M4 Competition: Results, findings, conclusion and way forward», *International Journal of Forecasting*, τόμ. 34, σσ. 802–808, 4 Οκτ. 2018, issn: 01692070. doi: 10.1016/j.ijforecast.2018.06.001.
- [10] S. Makridakis, E. Spiliotis, V. Assimakopoulos κ.ά., «The M5 uncertainty competition: Results, findings and conclusions», *International Journal of Forecasting*, τόμ. 38, σσ. 1365–1385, 4 Οκτ. 2022, issn: 01692070. doi: 10.1016/j.ijforecast.2021.10.009.
- [11] N. Laptev, J. Yosinski, L. E. Li και S. Smyl, «Time-series extreme event forecasting with neural networks at uber», στο *International conference on machine learning*, sn, τόμ. 34, 2017, σσ. 1–5.
- [12] S. Makridakis και M. Hibon, «The M3-Competition: results, conclusions and implications», *International Journal of Forecasting*, τόμ. 16, σσ. 451–476, 4 Οκτ. 2000, issn: 01692070. doi: 10.1016/S0169-2070(00)00057-1.
- [13] B. N. Oreshkin, D. Carpov, N. Chapados και Y. Bengio, «N-BEATS: Neural basis expansion analysis for interpretable time series forecasting», 2020.

- [14] H. Shi, M. Xu και R. Li, «Deep Learning for Household Load Forecasting—A Novel Pooling Deep RNN», *IEEE Transactions on Smart Grid*, τόμ. 9, αρθμ. 5, σσ. 5271–5280, 2018. doi: 10.1109/TSG.2017.2686012.
- [15] M. Voß, C. Bender-Saebelkampf και S. Albayrak, «Residential Short-Term Load Forecasting Using Convolutional Neural Networks», στο *2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, 2018, σσ. 1–6. doi: 10.1109/SmartGridComm.2018.8587494.
- [16] M. Grabner, Y. Wang, Q. Wen, B. Blažič και V. Štruc, «A Global Modeling Approach for Load Forecasting in Distribution Networks», *arXiv preprint arXiv:2204.00493*, Απρ. 2022.
- [17] G. Ke, Q. Meng, T. Finley κ.ά., «LightGBM: A Highly Efficient Gradient Boosting Decision Tree», στο *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio κ.ά., επιμελητές, τόμ. 30, Curran Associates, Inc., 2017.
- [18] *LightGBM documentation*. διεύθυν.: <https://github.com/Microsoft/LightGBM> (επίσκεψη 03/03/2023).
- [19] *python documentation*. διεύθυν.: <https://www.python.org/> (επίσκεψη 05/03/2023).
- [20] *darts documentation*. διεύθυν.: <https://unit8co.github.io/darts/> (επίσκεψη 05/03/2023).
- [21] *Optuna documentation*. διεύθυν.: <https://optuna.org/> (επίσκεψη 03/03/2023).
- [22] T. Akiba, S. Sano, T. Yanase, T. Ohta και M. Koyama, «Optuna: A Next-generation Hyperparameter Optimization Framework», στο *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019. doi: 10.1145/3292500.3330701.
- [23] *Pandas documentation*. διεύθυν.: <https://pandas.pydata.org/> (επίσκεψη 03/03/2023).
- [24] *Missing Values Filler - Darts 0.22.0 documentation*. διεύθυν.: https://unit8co.github.io/darts/generated_api/darts.dataprocessing.transformers.missing_values_filler.html?highlight=missing%20value#missing-values-filler (επίσκεψη 03/03/2023).
- [25] J. Peppanen, X. Zhang, S. Grijalva και M. J. Reno, «Handling bad or missing smart meter data through advanced data imputation», στο *2016 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, IEEE, Σεπτ. 2016, σσ. 1–5, ISBN: 978-1-5090-5167-0. doi: 10.1109/ISGT.2016.7781213.
- [26] T. Hong, «Short Term Electric Load Forecasting», North Carolina State University, 2010, σσ. 101–109.
- [27] *Scaler - Darts 0.22.0 documentation*. διεύθυν.: https://unit8co.github.io/darts/generated_api/darts.dataprocessing.transformers.scaler.html?highlight=scaling#scaler (επίσκεψη 03/03/2023).
- [28] S. M. Lundberg και S.-I. Lee, «A Unified Approach to Interpreting Model Predictions», στο *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio κ.ά., επιμελητές, τόμ. 30, Curran Associates, Inc., 2017.
- [29] J. Bergstra, R. Bardenet, Y. Bengio και B. Kégl, «Algorithms for Hyper-Parameter Optimization», στο *Advances in Neural Information Processing Systems*, J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira και K. Weinberger, επιμελητές, τόμ. 24, Curran Associates, Inc., 2011.

-
- [30] F. Hutter, H. Hoos και K. Leyton-Brown, «An Efficient Approach for Assessing Hyperparameter Importance», στο *Proceedings of the 31st International Conference on Machine Learning*, E. P. Xing και T. Jebara, επιμελητές, σειρά Proceedings of Machine Learning Research, τόμ. 32, Beijing, China: PMLR, 22–24 Jun 2014, σσ. 754–762.
- [31] ENTSO-E. διεύθυν.: <https://www.entsoe.eu/> (επίσκεψη 05/03/2023).
- [32] S. Pelekis, E. Karakolis, F. Silva κ.ά., «In Search of Deep Learning Architectures for Load Forecasting: A Comparative Analysis and the Impact of the Covid-19 Pandemic on Model Performance», IEEE, Ιούλ. 2022, σσ. 1–8, ISBN: 978-1-6654-6390-4. doi: 10.1109/IISA56318.2022.9904363.
- [33] S. Pelekis, I.-K. Seisopoulos, E. Spiliotis κ.ά., «A comparative assessment of deep learning models for day-ahead load forecasting: Investigating key accuracy drivers», *arXiv preprint arXiv:2302.12168*, Φεβ. 2023.
- [34] J. Park και E. Hwang, «A Two-Stage Multistep-Ahead Electricity Load Forecasting13; Scheme Based on LightGBM and Attention-BiLSTM», *Sensors*, τόμ. 21, σ. 7697, 22 Νοέ. 2021, ISSN: 1424-8220. doi: 10.3390/s21227697.
- [35] B. N. Oreshkin, G. Dudek, P. Peřka και E. Turkina, «N-BEATS neural network for mid-term electricity load forecasting», *Applied Energy*, τόμ. 293, σ. 116 918, Ιούλ. 2021, ISSN: 03062619. doi: 10.1016/j.apenergy.2021.116918.

Παραρτήματα

A Παράμετροι εφαρμογής και η σημασία τους

- **series_csv:** Αρχείο της χρονοσειράς. Προκαθορισμένη τιμή: ../RDN/Load_Data/2009-2021-global-load.csv
- **series_uri:** Υπερσύνδεσμος από τον οποίο γίνεται λήψη της χρονοσειράς. Προκαθορισμένη Τιμή: online_artifact
- **resolution:** Αλλαγή της χρονικής ακρίβειας των δεδομένων (σε λεπτά). Προκαθορισμένη Τιμή: 15
- **year_range:** Επιλογή εύρους χρονολογιών που θα συμπεριληφθούν στα δεδομένα. Προκαθορισμένη Τιμή: 2009-2019
- **time_covs:** Προαιρετική επιλογή covariate στη χρονοσειρά. Επιτρεπόμενες Τιμές: None, PT. Προκαθορισμένη Τιμή: PT
- **darts_model:** Η βασική αρχιτεκτονική του μοντέλου που πρόκειται να εκπαιδευτεί. Επιτρεπόμενες Τιμές: NBEATS, NHiTS, Transformer, RNN, TCN, BlockRNN, TFT, LightGBM, RandomForest, Naive, AutoARIMA. Δυνατότητα Πολλαπλών Τιμών: False. Προκαθορισμένη Τιμή: RNN
- **hyperparams_entrpoint, _h:** Το entry point του αρχείου config.yml του hyperparams το οποίο περιέχει τις επιθυμητές υπερπαραμέτρους για το επιλεγμένο μοντέλο. Προκαθορισμένη Τιμή: LSTM1
- **cut_date_val:** Η αρχική ημερομηνία του συνόλου επικύρωσης (δίνεται με τη μορφή YYYYMMDD). Προκαθορισμένη Τιμή: 20190101
- **cut_date_test:** Η αρχική ημερομηνία του συνόλου ελέγχου (δίνεται με τη μορφή YYYYMMDD). Προκαθορισμένη Τιμή: 20200101
- **test_end_date:** Η τελική ημερομηνία του συνόλου ελέγχου (δίνεται με τη μορφή YYYYMMDD). Προκαθορισμένη Τιμή: None
- **device:** Επιτρεπόμενες Τιμές: gpu, cpu. Δυνατότητα Πολλαπλών Τιμών: False. Προκαθορισμένη Τιμή: gpu
- **forecast_horizon:** Προκαθορισμένη Τιμή: 96
- **stride:** Προκαθορισμένη Τιμή: None
- **retrain:** Δυνατότητα επανεκπαίδευσης του μοντέλου κατά τη διάρκεια του backtesting Προκαθορισμένη Τιμή: False
- **ignore_previous_runs:** Δυνατότητα να αγνοηθούν τα προηγούμενα βηματικά τρεξίματα ενώ εκτελείται η σωλήνωση. Προκαθορισμένη Τιμή: True
- **scale:** Δυνατότητα κλιμάκωσης της χρονοσειράς στόχου. Προκαθορισμένη Τιμή: True

- **scale_covs**: Δυνατότητα κλιμάκωσης των συμμεταβλητών. Προκαθορισμένη Τιμή: True
- **day_first**: Δυνατότητα η ημερομηνία να εμφανίζει την ημέρα πριν τον μήνα. Προκαθορισμένη Τιμή: True
- **country**: Η χώρα της οποίας αναφέρεται το σύνολο δεδομένων. Προκαθορισμένη Τιμή: Portugal
- **std_dev**: Ο αριθμός με τον οποίο θα πολλαπλασιαστεί η τυπική απόκλιση της περιόδου του κάθε μήνα του dataframe. Το αποτέλεσμα χρησιμοποιείται ύστερα ως τιμή κατωφλιού όπως περιγράφεται παραπάνω. Προκαθορισμένη Τιμή: 4.5
- **max_thr**: Αν υπάρχει μια υποχρονοσειρά από διαδοχικά NaNs μεγαλύτερη του max_thr τότε δεν μετασχηματίζεται και επιστρέφεται με τις NaNs τιμές. Αν ισούται με -1, κάθε τιμή θα μετασχηματιστεί ανεξάρτητα του πόσο μεγάλη είναι η υπο-χρονοσειρά που αποτελείται από τα διαδοχικά NaNs. Προκαθορισμένη Τιμή: -1
- **a**: Το βάρος το οποίο δείχνει το πόσο γρήγορα το βάρος του απλού interpolation μειώνεται όσο η απόσταση στην πλησιέστερη μη-NaN τιμή αυξάνεται. Προκαθορισμένη Τιμή: 0.3
- **wncutoff**: Τα ιστορικά δεδομένα θα λάβουν μόνο υπόψιν τις ημερομηνίες οι οποίες έχουν απόσταση το πολύ wncutoff από την τρέχουσα κενή τιμή του αριθμού της εβδομάδας. Προκαθορισμένη Τιμή: 0.000694
- **ycutoff**: Τα ιστορικά δεδομένα θα λάβουν μόνο υπόψιν τις ημερομηνίες οι οποίες έχουν απόσταση το πολύ ycutoff από την τρέχουσα κενή τιμή του αριθμού της χρονιάς. Προκαθορισμένη Τιμή: 3
- **ydcutoff**: Τα ιστορικά δεδομένα θα λάβουν μόνο υπόψιν τις ημερομηνίες οι οποίες έχουν απόσταση το πολύ ydcutoff από την τρέχουσα κενή τιμή του αριθμού του yearday. Προκαθορισμένη Τιμή: 30
- **shap_data_size**: Μέγεθος του συνόλου δεδομένων του shap σε δείγματα. Προκαθορισμένη Τιμή: 10
- **analyze_with_shap**: Δυνατότητα να γίνει shap ανάλυση στο μοντέλο. Μόνο τα global προβλεπτικά μοντέλα υποστηρίζονται. Προκαθορισμένη Τιμή: False
- **multiple**: Δυνατότητα εκπαίδευσης σε πολλαπλές χρονοσειρές. Προκαθορισμένη Τιμή: False
- **eval_series**: Η χώρα στην οποία θα τρέξει το backtesting. Μόνο για πολλαπλές χρονοσειρές. Προκαθορισμένη Τιμή: Portugal
- **n_trials**: Πλήθος δοκιμών τις οποίες θα εκτελέσει η optuna. Προκαθορισμένη Τιμή: 100
- **opt_test**: Δυνατότητα εκτέλεσης optuna. Προκαθορισμένη Τιμή: False
- **from_mongo**: Αληθές αν το σύνολο δεδομένων διαβάζεται από MongoDB. Προκαθορισμένη Τιμή: False

- **mongo_name:** Ποιο mongo αρχείο διαβάζεται. Προκαθορισμένη Τιμή: rdn_load_data
- **num_workers:** Πλήθος νημάτων τα οποία θα χρησιμοποιηθούν από τη pytorch. Προκαθορισμένη Τιμή:4
- **eval_method:** Ποιο είδος ID ορίζεται στην eval_series: Αν είναι ts_id τότε κοιτάμε τη στήλη Timeseries ID, αλλιώς κοιτάμε τη στήλη Source Code. Επιτρεπόμενες Τιμές: ts_id, ts_code. Προκαθορισμένη Τιμή:ts_ID
- **l_interpolation:** Αληθές αν χρησιμοποιείται μόνο γραμμικό interpolation. Προκαθορισμένη Τιμή: False
- **rmv_outliers:** Αληθές αν θέλουμε να αφαιρεθούν οι ακραίες τιμές. Προκαθορισμένη Τιμή: True
- **loss_function:** Συνάρτηση κόστους η οποία θα χρησιμοποιηθεί στην optuna. Επιτρεπόμενες Τιμές: Mape, Smape, Mase, Mae, Rmse. Προκαθορισμένη Τιμή: Mape
- **evaluate_all_ts:** Δυνατότητα επικύρωσης των μοντέλων σε όλες τις χρονοσειρές και επιστροφή του μέσου όρου των μετρικών τους. Προκαθορισμένη Τιμή: False
- **convert_to_local_tz:** Δυνατότητα μετατροπής του χρόνου. Προκαθορισμένη Τιμή: True
- **grid_search:** Δυνατότητα τρεξίματος grid search ή χρήση tpe στην optuna. Προκαθορισμένη Τιμή:False
- **input_chunk_length:** input_chunk_length των δειγμάτων του shap. Λαμβάνεται υπόψιν μόνο αν δεν βρισκόμαστε στο στάδιο επικύρωσης global προβλεπτικού μοντέλου. Προκαθορισμένη Τιμή: None