



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

**Προβλέψεις χρονοσειρών σε έξυπνα δίκτυα ηλεκτρικής
ενέργειας με τεχνικές μηχανικής μάθησης και βαθιάς
μάθησης**

Διπλωματική Εργασία

Αλέξανδρος-Μενέλαος Τζώρτζης

Επιβλέπων: Κατσαρός Δημήτριος

Φεβρουάριος 2022



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

**Προβλέψεις χρονοσειρών σε έξυπνα δίκτυα ηλεκτρικής
ενέργειας με τεχνικές μηχανικής μάθησης και βαθιάς
μάθησης**

Διπλωματική Εργασία

Αλέξανδρος-Μενέλαος Τζώρτζης

Κατσαρός Δημήτριος: Κατσαρός Δημήτριος

Φεβρουάριος 2022



UNIVERSITY OF THESSALY
SCHOOL OF ENGINEERING
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

**Time series predictions in smart power grids with machine
learning and deep learning techniques**

Diploma Thesis

Alexandros-Menelaos Tzortzis

Supervisor: Katsaros Dimitrios

February 2022

Εγκρίνεται από την Επιτροπή Εξέτασης:

Κατσαρός Δημήτριος **Κατσαρός Δημήτριος**

Αναπληρωτής Καθηγητής, Τμήμα Ηλεκτρολόγων Μηχανικών και
Μηχανικών Υπολογιστών, Πανεπιστήμιο Θεσσαλίας

Μέλος

Δασκαλοπούλου Ασπασία

Αναπληρώτρια Καθηγήτρια, Τμήμα Ηλεκτρολόγων Μηχανικών και
Μηχανικών Υπολογιστών, Πανεπιστήμιο Θεσσαλίας

Μέλος

Ψαρράς Ιωάννης

Καθηγητής, Τομέας Ηλεκτρικών Βιομηχανικών Διατάξεων και Συστημάτων Αποφάσεων, Εθνικό Μετσόβιο Πολυτεχνείο

Ευχαριστίες

Αρχικά, η πραγματοποίηση και η ολοκλήρωση αυτής της πτυχιακής εργασίας θα ήταν αδύνατη χωρίς την υποστήριξη του διδακτορικού φοιτητή και ερευνητή Σωτήρη Πελέκη, στον οποίο εκφράζω θερμά τις ευχαριστίες και την ευγνωμοσύνη μου για τον πολύτιμο χρόνο που μου διέθεσε για να μου παρέχει σημαντικές εξηγήσεις επί του θέματος οι οποίες ήταν πολύτιμες για την επίτευξη αυτής της εργασίας.

Επίσης οφείλω ένα μεγάλο ευχαριστώ στον καθηγητή κύριο Ιωάννη Ψαρρά, που χωρίς την συμμετοχή του, δεν θα είχα την ευκαιρία να ασχοληθώ με το θέμα και έκανε την διπλωματική αυτή πραγματικότητα αυτό, καθώς και όλη την συνεργασία και κατανόηση που είχε σε όλη την διάρκεια της.

Φυσικά, θα ήθελα να ευχαριστήσω και τους καθηγητές μου στην επιτροπή παρουσίασης από το τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Θεσσαλίας. Η συμμετοχή του ήταν καθοριστική και τους είμαι ευγνώμων που ήταν πάνω από πρόθυμοι να με βοηθήσουν για την πραγματοποίησή της.

Τέλος, να ευχαριστήσω την οικογένειά μου που με στήριξε, με κατανόησε και με επικροτούσε σε κάθε μου προσπάθεια καθώς και όλους τους φίλους μου που έκαναν την εργασία να φαίνεται πιο εύκολη.

ΥΠΕΥΘΥΝΗ ΔΗΛΩΣΗ ΠΕΡΙ ΑΚΑΔΗΜΑΪΚΗΣ ΔΕΟΝΤΟΛΟΓΙΑΣ ΚΑΙ ΠΝΕΥΜΑΤΙΚΩΝ ΔΙΚΑΙΩΜΑΤΩΝ

«Με πλήρη επίγνωση των συνεπειών του νόμου περί πνευματικών δικαιωμάτων, δηλώνω ρητά ότι η παρούσα διπλωματική εργασία, καθώς και τα ηλεκτρονικά αρχεία και πηγαίοι κώδικες που αναπτύχθηκαν ή τροποποιήθηκαν στα πλαίσια αυτής της εργασίας, αποτελεί αποκλειστικά προϊόν προσωπικής μου εργασίας, δεν προσβάλλει κάθε μορφής δικαιώματα διανοητικής ιδιοκτησίας, προσωπικότητας και προσωπικών δεδομένων τρίτων, δεν περιέχει έργα/εισφορές τρίτων για τα οποία απαιτείται άδεια των δημιουργών/δικαιούχων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον και πληρούν τους κανόνες της επιστημονικής παράθεσης. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο, αρχεία ή/και πηγές άλλων συγγραφέων, αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή. Δηλώνω επίσης ότι τα αποτελέσματα της εργασίας δεν έχουν χρησιμοποιηθεί για την απόκτηση άλλου πτυχίου. Αναλαμβάνω πλήρως, ατομικά και προσωπικά, όλες τις νομικές και διοικητικές συνέπειες που δύναται να προκύψουν στην περίπτωση κατά την οποία αποδειχθεί, διαχρονικά, ότι η εργασία αυτή ή τμήμα της δεν μου ανήκει διότι είναι προϊόν λογοκλοπής».

Ο/Η Δηλών/ούσα

Αλέξανδρος-Μενέλαος Τζώρτζης

Διπλωματική Εργασία

Προβλέψεις χρονοσειρών σε έξυπνα δίκτυα ηλεκτρικής ενέργειας με τεχνικές μηχανικής μάθησης και βαθιάς μάθησης

Αλέξανδρος-Μενέλαος Τζώρτζης

Περίληψη

Περισσότερο από ποτέ άλλοτε, η βραχυπρόθεσμη πρόβλεψη φορτίου (STLF) είναι ζωτικής σημασίας για την καθημερινή λειτουργία των δικτύων ηλεκτρικής ενέργειας, λόγω της απελευθέρωσης της αγοράς ηλεκτρικής ενέργειας και της αυξανόμενης διείσδυσης των ανανεώσιμων πηγών ενέργειας (ΑΠΕ). Η βραχυπρόθεσμη πρόβλεψη φορτίου είναι μια αρκετά δύσκολη εργασία, καθώς οι χρονοσειρές ζήτησης ενέργειας χαρακτηρίζονται από μη γραμμικότητα, μη στασιμότητα και πολύπλοκη εποχικότητα και τάσεις, λόγω του γεγονότος ότι η ζήτηση ηλεκτρικής ενέργειας προέρχεται από διάφορα ηλεκτρικά φορτία που, με τη σειρά τους, εξαρτώνται από πολλές εξωτερικές μεταβλητές. Για να ξεπεραστούν οι προκλήσεις που προκύπτουν και να διασφαλιστεί η ακριβής πρόβλεψη ενέργειας, έχουν εμφανιστεί ανά τα χρόνια μέθοδοι πρόβλεψης που κυμαίνονται από τις παραδοσιακές μεθόδους πρόβλεψης και την ανάλυση χρονοσειρών έως τις τεχνικές μηχανικής μάθησης (ML) και βαθιάς μάθησης (DL), συχνά υποβοηθούμενες από μεθόδους συσσωμάτωσης μοντέλων (ensembles). Στο πλαίσιο της διπλωματικής αυτής, οι μέθοδοι αυτές επεκτείνονται ένα βήμα παραπέρα με την εισαγωγή μεθόδων μεταφοράς μάθησης μεταξύ 28 διαφορετικών χρονοσειρών εθνικής ζήτησης ηλεκτρικής ενέργειας ευρωπαϊκών χωρών. Ορόσημο της μελέτης αποτελεί μια στοχευμένη ανάλυση συσταδοποίησης που πραγματοποιείται στις ζητούμενες χρονοσειρές φανερώνοντας ότι δεδομένα τοπικά γειτονικών περιοχών παρουσιάζουν υψηλή συσχέτιση. Αυτό υποδεικνύει ότι μοντέλα εκπαιδευμένα σε δεδομένα γειτονικών χωρών και άρα κοινών συστάδων μπορούν συχνά να οδηγήσουν σε βελτίωση της προβλεπτικής ακρίβειας μέσω μεταφοράς μάθησης. Βασική συνεισφορά της διπλωματικής αυτής είναι η δημιουργία ενιαίων μοντέλων βασισμένων στη μεταφορά μάθησης τα οποία, με περιορισμένη εκπαίδευση, μπορούν να παράξουν επαρκείς προβλέψεις σε νέες εθνικές χρονοσειρές, εξυπηρετώντας έτσι ενδιαφερόμενους του ενεργειακού τομέα όπως διαχειριστές συστημάτων μεταφοράς και διανομής παγκοσμίως.

Λέξεις-κλειδιά:

Βαθιά Μάθηση, Μεταφορά μάθησης, Ενεργειακή ζήτηση, βραχυπρόθεσμη πρόβλεψη φορτίου, Πρόβλεψη επόμενης-μέρας, Ανάλυση χρονοσειρών, Επιχειρησιακή Έρευνα, Συστήματα Αποφάσεων

Diploma Thesis

Time series predictions in smart power grids with machine learning and deep learning techniques

Alexandros-Menelaos Tzortzis

Abstract

More than ever before, short-term load forecasting (STLF) is vital for the day-to-day operation of electricity grids, due to the deregulation of the electricity market and the increasing penetration of renewable energy sources (RES). Short-term load forecasting is quite a challenging task, as energy demand time series are characterized by non-linearity, non-stationarity and complex seasonality and trends, due to the fact that electricity demand comes from various electrical loads which, in turn, depend on many external variables. To overcome the challenges that arise and ensure accurate energy forecasting, forecasting methods have emerged over the years ranging from traditional forecasting methods and time series analysis to machine learning (ML) and deep learning (DL) techniques, often assisted by model ensembles. In the context of this thesis, these methods are extended one step further by introducing learning transfer methods between 28 different time series of national electricity demand of European countries. The milestone of the study is a targeted clustering analysis performed on the time series of interest revealing that data from locally adjacent regions show high correlation. This suggests that models trained on data from neighbouring countries and thus common clusters can often lead to improved forecasting accuracy through transfer learning. The key contribution of this thesis is the creation of unified models based on transfer learning that, with limited training, can produce adequate forecasts on new national time series, thus serving energy sector stakeholders such as transmission and distribution system operators worldwide.

Keywords:

Deep Learning, Transfer Learning, Energy Demand, short-term load forecasting, Day-ahead forecasting, Time series Analysis, Operational Research, Decision Systems

Πίνακας περιεχομένων

Ευχαριστίες	ix
Περίληψη	xii
Abstract	xiv
Πίνακας περιεχομένων	xv
Κατάλογος σχημάτων	xix
Κατάλογος πινάκων	xxi
1 Εισαγωγή	1
1.1 Σχετική βιβλιογραφία	3
1.2 Σκοπός Εργασίας & Συνεισφορά	5
1.3 Δομή Εργασίας	6
I Θεωρητικό Υπόβαθρο	7
2 Χρονοσειρές	9
2.1 Εισαγωγή	9
2.2 Ανάλυση Χρονοσειρών	9
2.3 Αποσύνθεση Χρονοσειρών	13
2.4 Μέθοδοι Προβλέψεων	15
2.5 Τεχνικές Προβλέψεων	18
2.6 Μοντέλα AR(I)MA	23
2.7 Καταλογισμός (Imputation)	25

3	Μηχανική Μάθηση	29
3.1	Κατηγορίες Μηχανικής Μάθησης	29
3.2	Νευρωνικά Δίκτυα	32
3.3	Machine Learning Life-cycle	40
3.4	Hyperparameter Tuning	43
4	Transfer Learning	45
4.1	Εισαγωγή	45
4.2	Βασικές Έννοιες & Ορισμοί	46
4.3	Κατηγοριοποίηση ειδών	47
4.4	Deep transfer learning	48
II	Λογισμικό & Πείραμα	51
5	Υλοποίηση ML Pipeline	53
5.1	Επιλογή τεχνολογιών	53
5.2	Περιβάλλον & Αρχιτεκτονική Pipeline	56
6	Ανάλυση Δεδομένων (EDA)	67
6.1	Mean Plots	67
6.2	Box-plots	68
6.3	ADF Test	70
6.4	ACF/PACF Plots	70
6.5	Dendrogram & Choropleth	71
6.6	Heatmaps	84
6.7	Annual lineplots & historgams	89
7	Πειραματική Περιγραφή	91
7.1	Δεδομένα	91
7.2	Επεξεργασία	92
7.3	Case studies	93
7.4	Αρχιτεκτονική & εκπαίδευση	94
7.5	Αξιολόγηση	95

III	Αποτελέσματα & Συμπεράσματα	97
8	Αποτελέσματα Πειραμάτων	99
8.1	MLflow UI	99
8.2	Σημασία Ensembling	100
8.3	Επίδοση transfer learning	102
9	Συμπεράσματα	107
9.1	Σύνοψη και συμπεράσματα	107
9.2	Μελλοντικές επεκτάσεις	109
	Βιβλιογραφία	111
	Παράρτημα	122

Κατάλογος σχημάτων

1.1	Ποσοστό πρωτογενούς ενέργειας από ανανεώσιμες πηγές	2
2.1	Στοιχεία Χρονοσειρών	11
2.2	Στασιμότητα χρονοσειρών	12
2.3	Διάστημα εμπιστοσύνης	18
2.4	Εκθετική Εξομάλυνση	20
2.5	Κατηγορίες παρεμβολών	27
3.1	Νευρώνας Perceptron	34
3.2	Multilayer Perceptron	35
3.3	Συμπεριφορά learning rate	38
4.1	Transfer Learning	46
4.2	Μέθοδοι Transfer Learning	48
4.3	Μέθοδος freezing/fine-tuning	50
4.4	Μέθοδος Head Replacement	50
5.1	Deployed Architecture	56
5.2	Pipeline	58
5.3	Δημιουργία εισόδων μοντέλου	63
6.1	Mean plots	68
6.2	Box plots	69
6.3	ACF/PACF plots	71
6.4	Correlation: Daily profiles	75
6.5	Time Profile	77
6.6	No normalization: Daily profiles	78

6.7	Energy Demand Choropleth	79
6.7	Normalization: Time profiles in local time	82
6.7	Normalization: Time profiles in UTC time	84
6.7	Heatmaps	88
6.8	Annual dataset visualization	90
7.1	Imputation	93
7.2	TestCase: all-tl	94
8.1	MLflow UI	100
8.1	Ensemble-Single Model Comparison	102
8.2	Test case MAPE barplots	103
8.3	baseline - Cluster-TL Difference	105

Κατάλογος πινάκων

7.1	Πίνακας υπερπαραμέτρων	95
9.1	Entrypoint parameters	122
9.2	Entrypoint parameters' description	123
9.3	Test case comparison	124
9.4	Best model hyperparams - Benchmark	125
9.5	Best model hyperparams - All-TL	126
9.6	Best model hyperparams - Cluster-TL	127

Nomenclature

ANN	Artificial neural network	LTLF	Long-term load forecasting
DL	Deep learning	LW	Lookback window
NN	Neural Network	FW	Forecast Horizon
ML	Machine Learning	EDA	Exploratory Data Analysis
TL	Transfer Learning	ETL	Extract - Transform - Load
MLP	Multi-layer perceptron	MLOps	Machine learning operations
LSTM	Long short-term memory	MAPE	Mean absolute percentage error
RNN	Recurrent Neural Network	RMSE	Root Mean Square error
CNN	Convolutional Neural Network	RES	Renewable Energy Storage
STLF	Short-term load forecasting	TSO	Transmission System Operator
MTLF	Mid-term load forecasting		

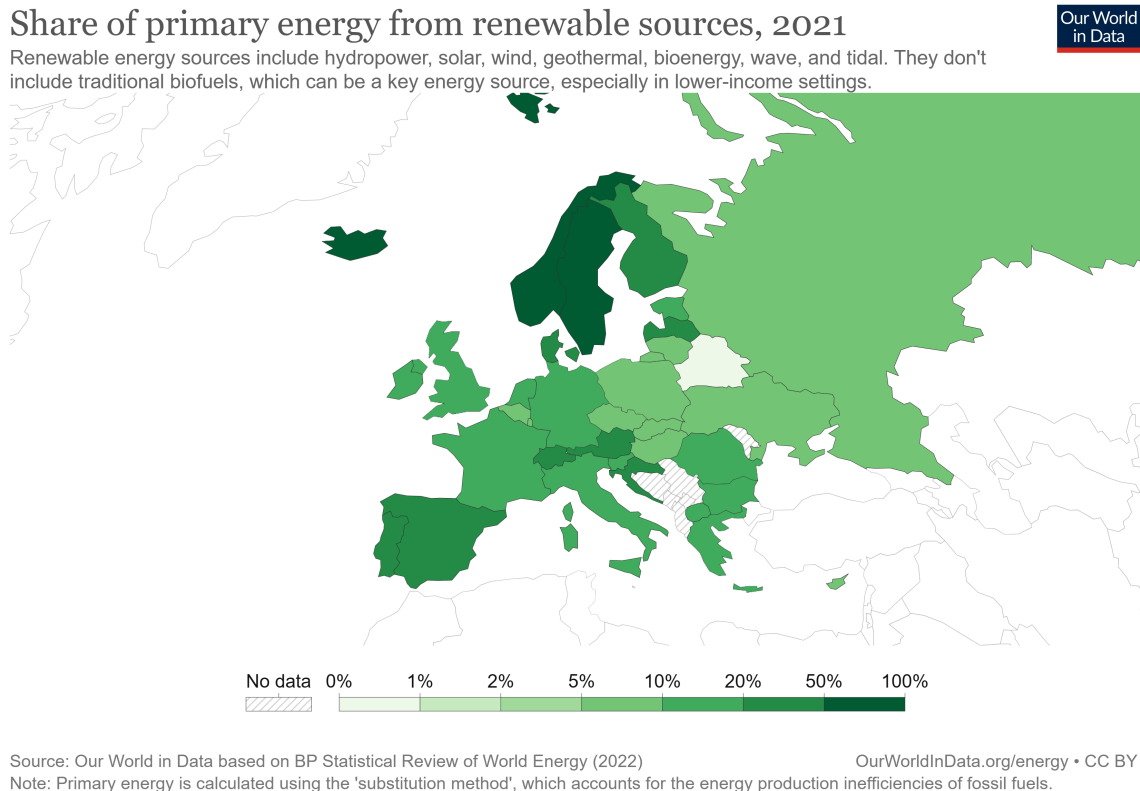
Κεφάλαιο 1

Εισαγωγή

Βρισκόμενοι στο 2022, παρατηρούμε ότι είναι απαραίτητη η παραγωγή γρήγορων και αξιόπιστων προβλέψεων. Οι προβλέψεις είναι πολύτιμες τόσο για μεγάλες, όσο και για τις επιχειρήσεις, διότι δίνουν τη δυνατότητα να λαμβάνουν τεκμηριωμένες επιχειρηματικές αποφάσεις και να αναπτύσσουν στρατηγικές με βάση τα αμέτρητα δεδομένα που παράγουν. Οι οικονομικές και επιχειρησιακές αποφάσεις λαμβάνονται με βάση τις τρέχουσες συνθήκες της αγοράς και τις προβλέψεις για το πώς φαίνεται το μέλλον. Τα δεδομένα του παρελθόντος συγκεντρώνονται και αναλύονται για να βρεθούν μοτίβα, τα οποία χρησιμοποιούνται για την πρόβλεψη μελλοντικών τάσεων και αλλαγών. Η πρόβλεψη επιτρέπει στην εκάστοτε εταιρεία να είναι προληπτική αντί για αντιδραστική.

Η ηλεκτρική ενέργεια ως προϊόν έχει πολύ διαφορετικά χαρακτηριστικά σε σύγκριση με ένα υλικό προϊόν. Για παράδειγμα, η ηλεκτρική ενέργεια δεν μπορεί να αποθηκευτεί αποδοτικά όταν πρόκειται για μεγάλες ποσότητες: είναι ακριβή, απαιτεί υψηλή συντήρηση και έχει περιορισμένη διάρκεια ζωής. Εξαιτίας αυτού, οι επιχειρήσεις κοινής ωφέλειας πρέπει να εξισορροπούν την προσφορά και τη ζήτηση κάθε στιγμή, και έτσι είναι ευνοϊκότερο να παράγεται αμέσως μόλις ζητείται. Οποιαδήποτε εμπορική εταιρεία ηλεκτρικής ενέργειας έχει διάφορους στρατηγικούς στόχους. Ένας από αυτούς τους στόχους είναι να παρέχει στους τελικούς χρήστες (απαιτήσεις της αγοράς) ασφαλή και σταθερή ηλεκτρική ενέργεια. Ως εκ τούτου, η πρόβλεψη φορτίου ηλεκτρικής ενέργειας είναι μια ζωτικής σημασίας διαδικασία στον προγραμματισμό της βιομηχανίας ηλεκτρικής ενέργειας και της λειτουργία των συστημάτων ηλεκτρικής ενέργειας. Οι ακριβείς προβλέψεις οδηγούν σε σημαντική εξοικονόμηση του κόστους λειτουργίας και συντήρησης, σε αυξημένη αξιοπιστία του συστήματος μεταφοράς ηλεκτρικής ενέργειας και σωστές αποφάσεις για τη μελλοντική ανάπτυξη.

Αυτό είναι ιδιαίτερα επίκαιρο στην σημερινή εποχή όπου οι ανανεώσιμες πηγές ενέργειας (ΑΠΕ) παίζουν όλο και πιο καθοριστικό ρόλο στο σημερινό σύστημα ηλεκτρικής ενέργειας όπως φαίνεται και στο σχήμα 1.1. Είναι πολύ σημαντικό για τους διαχειριστές του δικτύου να γνωρίζουν πόση ενέργεια θα παράγεται αλλά και πόση ζητείται στις επόμενες ώρες και ημέρες. Η αποθήκευση ηλεκτρικής ενέργειας είναι απαραίτητη στην περίπτωση που υπάρχει πλεονάζουσα παραγωγή ενέργειας από τις ΑΠΕ και μικρότερη ζήτηση φορτίου



Σχήμα 1.1: Στον χάρτη αυτόν φαίνεται το ποσοστό πρωτογενούς ενέργειας κάθε ευρωπαϊκής χώρας από ανανεώσιμες πηγές ενέργειας [1]

Η ζήτηση ηλεκτρικής ενέργειας εκτιμάται από τη συσσώρευση της κατανάλωσης σε τακτά χρονικά διαστήματα [2]. Οι προβλέψεις της ζήτησης συνήθως κατηγοριοποιούνται, όσον αφορά τη διάρκεια της πρόβλεψης, σε τρεις κατηγορίες: (short-term) βραχυπρόθεσμες προβλέψεις, μεσοπρόθεσμες (mid-term) προβλέψεις και μακροπρόθεσμες (long-term) προβλέψεις [3–6]. Δεδομένου ότι οι χρονοσειρές ενεργειακού φορτίου παρουσιάζουν μη γραμμικότητα, μη στασιμότητα, θόρυβο και διαφορετικές εποχιακές τάσεις, οι προβλέψεις σε μεγάλο βάθος χρόνου συνοδεύονται με μεγαλύτερη ανακρίβεια. Έτσι, το μεγαλύτερο ενδιαφέρον (και ο σκοπός τη διπλωματικής) βρίσκεται στην κατηγορία του short-term load forecasting

(STLF) για προβλέψεις στο όσο το δυνατόν πιο κοντινό μέλλον.

Η υψηλή μεταβλητότητα των δεδομένων ηλεκτρικής ενέργειας κάνει την κατασκευή τέτοιων μοντέλων μια μεγάλη πρόκληση, μια στην οποία χρειάζονται όσο περισσότερα δεδομένα γίνεται προκειμένου να μεγιστοποιηθεί η ακρίβεια των προβλέψεων. Παρόλα αυτά, εξαιτίας της πληθώρας και υψηλής ανάλυσής τους, το υπολογιστικό κόστος για την πλήρη χρήση τους είναι αποθαρρυντικό για την κατασκευή τέτοιων μοντέλων.

Στην παρούσα διπλωματική αντιμετωπίζουμε αυτό το πρόβλημα αξιοποιώντας τεχνικές μεταφοράς μάθησης, στις οποίες γνώση που αποκτήθηκε από την επίλυση ενός προβλήματος, να "μεταφέρεται" για να χρησιμοποιηθεί για να λυθεί ένα άλλο (συνήθως γειτονικό) πρόβλημα. Με αυτό τον τρόπο, ενώ μειώνουμε την χρονική ακρίβεια για την κατασκευή των μοντέλων χρησιμοποιώντας ήδη υπάρχουσα γνώση. Τα οφέλη όμως δεν σταματάνε εκεί, η ιδέα ενός παγκόσμιου μοντέλου πρόβλεψης στο οποίο, με λίγη επανεκπαίδευση, μπορεί να προσαρμοστεί στις ανάγκες μιας περιοχής είναι ιδιαίτερα δελεαστική και στους ενεργειακούς ενδιαφερόμενους (stakeholders). Έτσι, είναι σε θέση να ρυθμίσουν καλύτερα την ισορροπία παραγωγής - ζήτησης φορτίου, κάνοντας γρήγορες και έγκυρες προβλέψεις για την διακύμανση της ζήτησης. Ταυτόχρονα δίνεται η ευκαιρία για εύρεση έγκυρων προβλέψεων όταν δεν υπάρχουν προϋπάρχοντα δεδομένα (κατασκευή νέου υποσταθμού, προέκταση δικτύου κτλ).

Φυσικά, είναι σημαντικό να εξεταστούν τα δεδομένα προς χρήση, για να παρατηρηθούν μοτίβα και συμπεριφορές ώστε να είμαστε σε θέση να επιβεβαιώσουμε με μεγαλύτερη αυτοπεποίθηση, ότι η μεταφορά μάθησης πράγματι θα βοηθήσει σε καλύτερη απόδοση των μοντέλων πρόβλεψης.

1.1 Σχετική βιβλιογραφία

Τα οφέλη που προσφέρει η ακριβής πρόβλεψη ζήτησης ηλεκτρικής ενέργειας καθιστά την κατασκευή των ανάλογων μοντέλων εξαιρετικά πολύτιμη στον τομέα της ηλεκτρικής ενέργειας [7, 8]. Αρχικά για την πρόβλεψη δεδομένων, χρησιμοποιήθηκαν στατιστικές μέθοδοι και γραμμικά μοντέλα όπως αυτοπαλινδρομούμενα και κινητού μέσου όρου (ARIMA). Παρατηρήθηκε όμως ότι πολλές χρονοσειρές δεν είχαν γραμμική συμπεριφορά. Φυσικά αυτό είχε σαν αποτέλεσμα τα μοντέλα αυτά να είναι ανεπαρκή που οδήγησε στην χρήση μεθόδων μηχανικής μάθησης [9].

Στη βιβλιογραφία, έχει γίνει χρήση πολλών τεχνολογιών τόσο για mid-term [10, 11] όσο και για long-term forecasting [12, 13] με χρήση τεχνικών πρόβλεψης όπως μοντέλα ARIMA, SVMs και φυσικά νευρωνικών δικτύων [14]. Υπάρχει, επίσης μεγάλη πληθώρα ερευνών στην οποία γίνεται χρήση μεταφοράς μάθησης για την πρόβλεψη ζήτησεως ηλεκτρικής ενέργειας. Έχουν δημιουργηθεί μοντέλα mid-term (MTLF) και long-term load forecasting (LTLF) όπως για τον υπολογισμό του ανώτατου φορτίου ηλεκτρικής ενέργειας [15] ή την μηνιαία κατανάλωση σε τοπικά δίκτυα ηλεκτρικής ενέργειας [16].

Επιπλέον, η ανάγκη για μεγαλύτερη διείσδυση των μεθόδων αποθήκευσης ανανεώσιμων πηγών ενέργειας (ΑΠΕ) [17] οδηγεί στην ανάγκη για ακριβέστερες προβλέψεις που μπορούν να ικανοποιήσουν με επιτυχία τις απαιτήσεις ευελιξίας και απόκρισης στη ζήτηση που τίθενται στα συστήματα λειτουργίας μεταφοράς και διανομής. Ως αποτέλεσμα, το STLTF έχει εξεταστεί σε διάφορα ερευνητικά έργα και μελέτες [18–22].

Φυσικά, η χρήση μεταφοράς μάθησης για την πρόβλεψη δεδομένων έχει εξεταστεί σε αρκετές έρευνες για την απόδοση σε προβλήματα STLTF. Σε αυτή την κατηγορία βρίσκονται έρευνες όπως αυτή των *Gomez-Rosero, Capretz και Mir* [23] στην οποία έγινε χρήση ενός LSTM μοντέλου για την πρόβλεψη κατανάλωσης ηλεκτρικής ενέργειας από νοικοκυριά στο Λονδίνο. Ακόμα, υπάρχει η έρευνα των *Eunjung Lee and Wonjong Rhee* [24] στην οποία, χρησιμοποιήθηκε MLP (Multi-Layer Perceptron) για την πρόβλεψη ζήτησης κάνοντας χρήση του ICU repository [25] με δεδομένα από ηλεκτρικά φορτία 370 υποσταθμών στην Πορτογαλία με περίοδο δειγματοληψίας 15 λεπτών. Η μεταφορά μάθησης έχει επίσης αξιοποιηθεί σε μοντέλα Graphical Neural Networks (GNNs) εξετάζοντας την χωρική συσχέτιση που έχουν οι πηγές των δεδομένων μεταξύ τους [26] και σε Residual Neural Network (ResNet) μοντέλα ως ένας οργανικός τρόπος χαρτογράφησης (mapping) πολύπλοκων σχέσεων μεταξύ των φορτίων ενέργειας διαφόρων περιοχών [27]. Η ύπαρξη περιορισμένων ιστορικών δεδομένων είναι συχνό χαρακτηριστικό. Αξιοσημείωτη αναφορά αποτελεί η μελέτη των *Long Cai και Jie Gu* στην οποία γίνεται transfer learning στο GEFCom 2012: ένα σύνολο δεδομένων που αποτελείται από το ωριαίο ιστορικό φορτίου 20 ζωνών, το ωριαίο ιστορικό θερμοκρασίας 11 μετεωρολογικών σταθμών και έναν κατάλογο ομοσπονδιακών αργιών των ΗΠΑ σε μηνιαίο ιστορικό φορτίου που καταναλώνονται από διάφορες χώρες [28]. Στην έρευνά των *Chuong B Do και Andrew Y Ng*, γίνεται χρήση ενός μοντέλου δύο επιπέδων [29] για την δημιουργία νέων parameter functions για αναγνώριση κειμένου.

1.2 Σκοπός Εργασίας & Συνεισφορά

Στην παρούσα μελέτη, επικεντρωνόμαστε στη χρήση της μεταφοράς μάθησης για τη βραχυπρόθεσμη πρόβλεψη ζήτησης εθνικών φορτίων με αξιοποίηση δεδομένων γειτονικών χωρών. Εικάζουμε πως η ακρίβεια πρόβλεψης του φορτίου στη χώρα-στόχο μπορεί να αυξηθεί εάν τα πρότυπα κατανάλωσης ισχύος στις χώρες πηγής και στη χώρα-στόχο είναι παρόμοια.

Υπάρχουν διαφορές στους τρόπους με τους οποίους οι διάφορες ζώνες καταναλώνουν ενέργεια. Με αφορμή αυτό το γεγονός, διερευνάται η επίδραση βασικών εξωτερικών παραγόντων (γεωγραφία και κουλτούρα προέλευσης) στην ακρίβεια πρόβλεψης τους. Η ανάλυσή μας επικεντρώνεται στις προβλέψεις για την επόμενη ημέρα (day-ahead forecasting) και στις καθαρές συγκεντρωτικές χρονοσειρές φορτίου, χρησιμοποιώντας το σύνολο δεδομένων ENTSOE [30] που περιέχει τη ζήτηση ενέργειας από 28 ευρωπαϊκές χώρες. Οι κύριες συνεισφορές της μελέτης συνοψίζονται ως εξής:

1. Για την κατασκευή των μοντέλων χρησιμοποιείται MLP αρχιτεκτονική. Αποτελεί μια απλή αρχιτεκτονική, χαρακτηριστική όμως για την κατασκευή μοντέλων για πρόβλεψη χρονοσειρών και άρα ένα καλό σημείο εκκίνησης για benchmarking τεχνικών μεταφοράς μάθησης. Η απλότητα της επιτρέπει μεγάλη ευελιξία στην προσαρμογή των παραμέτρων της. Έτσι, γίνεται πιο ξεκάθαρη η εστίαση στις τεχνικές μεταφοράς μάθησης, χωρίς περιττές λεπτομέρειες και πολύπλοκες αρχιτεκτονικές.
2. Πραγματοποιείται διερευνητική ανάλυση δεδομένων (EDA) στην οποία εξετάζονται για ιδέες και μοτίβα που μπορεί να περιέχουν τα δεδομένα. Η διαδικασία αυτή συνεισφέρει στο να τονιστεί η αξία του προβλήματος transfer learning και να τεθεί η αντίστοιχη στοχοθεσία. Παράλληλα η μελέτη αυτή εξετάζει αν υπάρχει ομαδοποίηση (clustering) των δεδομένων η οποία μπορεί να οδηγήσει σε καλύτερη απόδοση των μοντέλων μεταφοράς μάθησης
3. Εξετάζονται τρεις διαφορετικές περιπτώσεις χρήσης μεταφοράς μάθησης. Στην πρώτη περίπτωση, κατασκευάζονται, εκπαιδεύονται και αξιολογούνται μοντέλα για κάθε χώρα του dataset ξεχωριστά. Τα μοντέλα αυτά λειτουργούν σαν σημείο αναφοράς (benchmark) για την απόδοση των μοντέλων στα οποία γίνεται χρήση transfer learning. Στην συνέχεια, ακολουθεί η δοκιμή μοντέλων στα οποία εξετάζονται η επίδοση ενός μοντέλου, εκπαιδευμένου σε όλες τις χώρες του dataset και αξιολόγηση του σε δεδομένου μιας ανεξάρτητης χώρας, κάνοντας χρήση transfer learning. Τέλος, δοκιμάζονται

transfer learning μοντέλα, εκπαιδευμένα σε χώρες που ομαλοποιήθηκαν σύμφωνα με το EDA, για να εξεταστεί πειραματικά η χρήση clustering των δεδομένων

4. Στο πλαίσιο της επιλογής όσο το δυνατόν καλύτερων τιμών των υπερπαραμέτρων των μοντέλων πραγματοποιείται προσεχτικό hyper-parameter tuning ενώ στη συνέχεια αξιοποιούνται τεχνικές ensembling οι οποίες έχουν αποδειχθεί ότι βελτιώνουν την απόδοση των βαθιών μοντέλων πρόβλεψης χρονοσειρών.

Από τα πειράματα, παρατηρήθηκε σημαντική βελτίωση της απόδοσης σε πολλά από τα ανάλογα δεδομένα μεταξύ του benchmark και των transfer learning test cases. Μάλιστα, φαίνεται μια ξεκάθαρη επίδοση στην περίπτωση που πραγματοποιήθηκε clustering, επιβεβαιώνοντας μερικώς την αρχική μας υπόθεση.

1.3 Δομή Εργασίας

Τα κεφάλαια 2-4 αποτελούν ένα ενιαίο μέρος: το *θεωρητικό υπόβαθρο* της διπλωματικής. Στο δεύτερο κεφάλαιο αναλύεται η δομή των χρονοσειρών: τον ορισμό και βασικές έννοιες τους, στην ανάλυση τους, την επεξεργασία τους καθώς και τεχνικών προβλέψεων τους. Το τρίτο κεφάλαιο γίνεται μια γενική ανάλυση του ευρύτερου κλάδου της μηχανικής μάθησης, τις κατηγορίες της καθώς και μια εισαγωγή στα νευρωνικά δίκτυα: τα βασικά χαρακτηριστικά τους και τη ρύθμισή τους (*parameter tuning*). Στο τέταρτο κεφάλαιο γίνεται μια πιο εξατομικευμένη παρατήρηση στην κατηγορία τεχνικών μηχανικής μάθησης ονόματι transfer learning, στην οποία εστιάζει και η παρούσα διπλωματική.

Στο κεφάλαιο 5 παρουσιάζεται η υλοποίηση του λογισμικού που κατασκευάστηκε με σκοπό να γίνουν οι μετρήσεις. Αναλύονται οι τεχνολογίες που χρησιμοποιήθηκαν, και οι αιτίες πίσω από την επιλογή τους. Ακόμα, γίνεται μια εκτεταμένη επεξήγηση, σε γενικό πλαίσιο, των επιμέρους σταδίων από τα οποία περνάνε τα δεδομένα για την εξαγωγή των συμπερασμάτων. Στην συνέχεια, ακολουθεί η πειραματική παρουσίαση. Λόγω της μεγάλης έκτασης της έρευνας που έγινε στα δεδομένα (EDA) θεωρήθηκε βοηθητικό να αποτελεί ένα ανεξάρτητο κεφάλαιο στην διπλωματική. Στο κεφάλαιο 7, περιγράφονται τα στάδια του ML pipeline που κατασκευάστηκε με μια πιο σκοπιά πιο εστιασμένη προς τα πειράματα που ακολούθησαν. Τέλος, στο κεφάλαιο 8, ακολουθούν τα συμπεράσματα της διπλωματικής και οι πιθανές μελλοντικές επεκτάσεις της.

Μέρος Ι

Θεωρητικό Υπόβαθρο

Κεφάλαιο 2

Χρονοσειρές

2.1 Εισαγωγή

Μια χρονοσειρά είναι μια ακολουθία σημείων δεδομένων που εμφανίζονται με διαδοχική σειρά κατά τη διάρκεια κάποιας χρονικής περιόδου. Δεν υπάρχει ελάχιστο ή μέγιστο χρονικό διάστημα που πρέπει να συμπεριληφθεί, επιτρέποντας τη συλλογή των δεδομένων με τρόπο που να παρέχει τις πληροφορίες που αναζητά ο αναλυτής που εξετάζει τη δραστηριότητα. Οι χρονοσειρές αποτελούν την κύρια πηγή δεδομένων που χρησιμοποιούνται από τις μεθόδους μηχανικής/στατιστικής μάθησης, για την ανάλυση και την εξαγωγή ουσιαστικών στατιστικών στοιχείων και χαρακτηριστικών σχετικά με τα δεδομένα.

2.2 Ανάλυση Χρονοσειρών

Η ανάλυση χρονοσειρών μπορεί να είναι χρήσιμη για να δούμε πώς μια δεδομένη μεταβλητή μεταβάλλεται με την πάροδο του χρόνου (ενώ ο ίδιος ο χρόνος, στα δεδομένα χρονοσειρών, είναι συχνά η ανεξάρτητη μεταβλητή). Η ανάλυση χρονοσειρών μπορεί επίσης να χρησιμοποιηθεί για να εξεταστεί πώς οι αλλαγές που σχετίζονται με το επιλεγμένο σημείο δεδομένων συγκρίνονται με τις μεταβολές σε άλλες μεταβλητές κατά την ίδια χρονική περίοδο.

2.2.1 Συστατικά Χρονοσειρών

Οι παραδοσιακές μέθοδοι ανάλυσης των χρονοσειρών ασχολούνται κυρίως με την αποσύνθεση της διακύμανσης της χρονοσειράς σε τέσσερα συστατικά της στοιχεία: *τάση*, *κυκλι-*

κότητα, εποχικότητα και τυχαιότητα.

Τάση : Ένα μοτίβο στα δεδομένα που δείχνει την κίνηση μιας σειράς προς σχετικά υψηλότερες ή χαμηλότερες τιμές κατά τη διάρκεια μιας μεγάλης χρονικής περιόδου. Με άλλα λόγια, μια τάση παρατηρείται όταν υπάρχει αυξανόμενη ή μειούμενη κλίση στη χρονοσειρά. Μια τάση θα μπορούσε να είναι *ανοδική, καθοδική, οριζόντια/στάσιμη*.

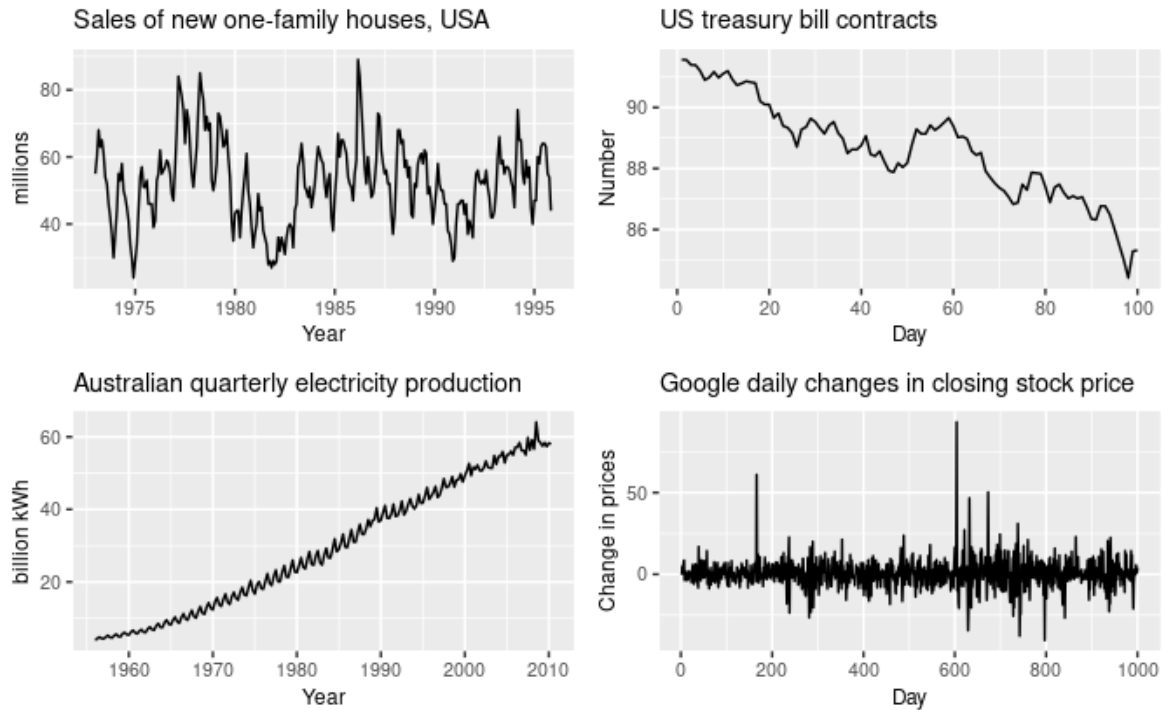
Εποχικότητα : Αναφέρεται σε περιοδικές διακυμάνσεις σε ορισμένους επιχειρηματικούς τομείς και κύκλους που συμβαίνουν τακτικά με βάση μια συγκεκριμένη εποχή. Μια εποχή μπορεί να αναφέρεται σε μια ημερολογιακή περίοδο, όπως το καλοκαίρι ή ο χειμώνας, ή μπορεί να αναφέρεται σε μια εμπορική περίοδο, όπως η περίοδος των διακοπών.

Κυκλικότητα : Φαινόμενο στο οποίο τα δεδομένα παρουσιάζουν αυξήσεις και μειώσεις που δεν έχουν σταθερή περίοδο. Η διάρκεια αυτών των διακυμάνσεων είναι συνήθως τουλάχιστον 2 έτη. Συχνά συγχέεται η κυκλική συμπεριφορά με την εποχιακή συμπεριφορά, αλλά στην πραγματικότητα είναι αρκετά διαφορετικές. Εάν οι διακυμάνσεις δεν έχουν σταθερή περίοδο, τότε είναι κυκλικές- εάν η περίοδος είναι αμετάβλητη και συνδέεται με κάποια πτυχή του ημερολογίου, τότε το μοτίβο είναι εποχιακό. Γενικά, η μέση διάρκεια των κύκλων είναι μεγαλύτερη από τη διάρκεια ενός εποχιακού προτύπου και το μέγεθος των κύκλων τείνει να είναι πιο μεταβλητό από το μέγεθος των εποχιακών προτύπων.

Τυχαιότητα : κινήσεις που εμφανίζονται ακανόνιστα και γενικά κατά τη διάρκεια σύντομων περιόδων. Οι ακανόνιστες μεταβολές δεν ακολουθούν ένα συγκεκριμένο μοντέλο και, ενώ δεν είναι προβλέψιμες, μπορούν όμως να εξηγηθούν με βάση κάποια γεγονότα. Στην πράξη, είναι όλες οι συνιστώσες των χρονολογικών σειρών που δεν μπορούν να αποδοθούν στην επίδραση των κυκλικών διακυμάνσεων ή των εποχιακών διακυμάνσεων ή εκείνων της τάσης.

Το σχήμα 2.1 απεικονίζει παραδείγματα μοτίβων για χρονοσειρές. Οι μηνιαίες πωλήσεις κατοικιών (πάνω αριστερά) παρουσιάζουν έντονη εποχικότητα εντός κάθε έτους. Τα συμβόλαια κρατικών γραμματίων των ΗΠΑ (πάνω δεξιά) εμφανίζουν μια προφανής πτωτική τάση. Η τριμηνιαία παραγωγή ηλεκτρικής ενέργειας της Αυστραλίας (κάτω αριστερά) δείχνει μια ισχυρή αυξητική τάση, με έντονη εποχικότητα. Τέλος, Η ημερήσια μεταβολή της τιμής κλεισίματος της μετοχής της Google (κάτω δεξιά) δεν παρουσιάζει τάση, εποχικότητα ή κυκλική

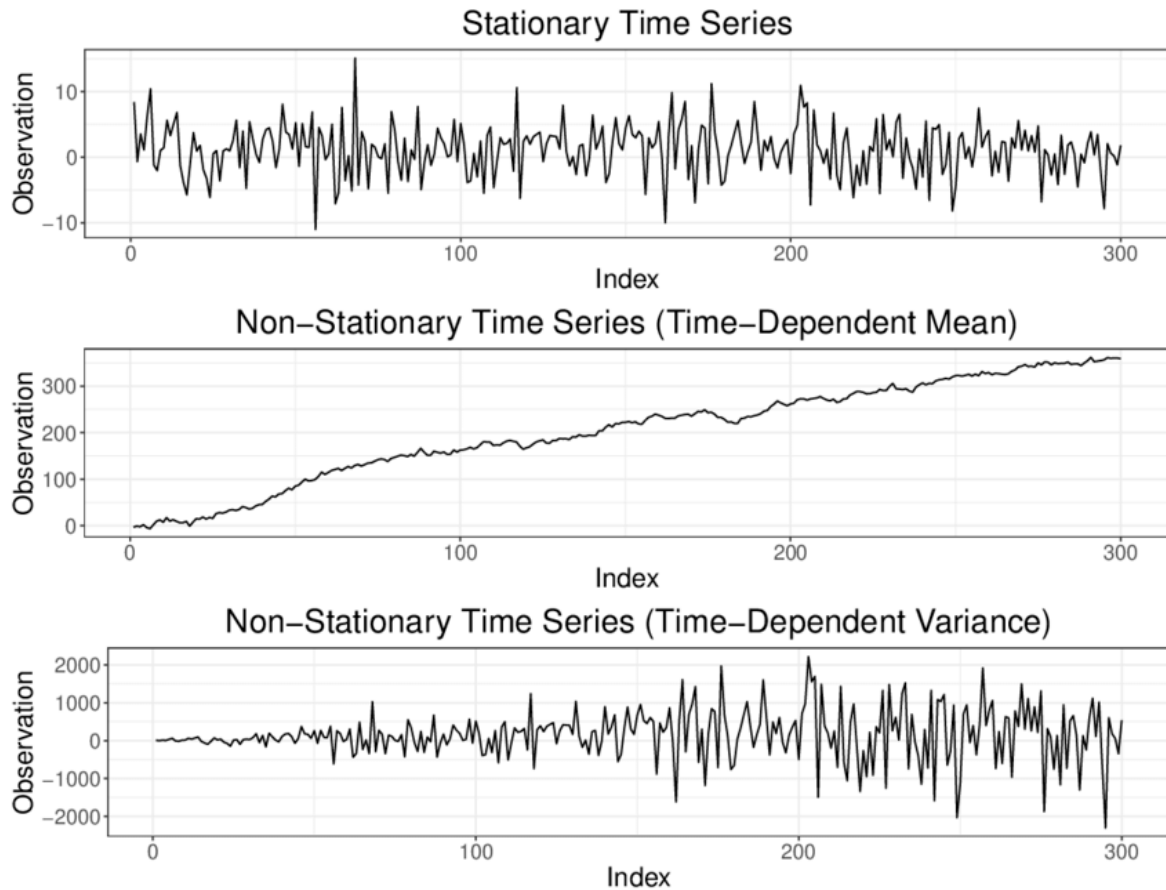
συμπεριφορά. Υπάρχουν τυχαίες διακυμάνσεις που δεν φαίνεται να είναι πολύ προβλέψιμες, και δεν υπάρχουν ισχυρά μοτίβα που θα βοηθούσαν στην ανάπτυξη ενός μοντέλου πρόβλεψης.



Σχήμα 2.1: Παραδείγματα μοτίβων χρονοσειρών [31].

2.2.2 Στασιμότητα (Stationarity)

Μια στάσιμη (stationary) χρονοσειρά είναι μια χρονοσειρά της οποίας οι ιδιότητες δεν εξαρτώνται από τη χρονική στιγμή κατά την οποία παρατηρείται. Έτσι, οι χρονοσειρές με τάσεις ή εποχικότητα δεν είναι στάσιμες - η τάση και η εποχικότητα θα επηρεάσουν την τιμή της χρονοσειράς σε διαφορετικές χρονικές στιγμές. Από την άλλη πλευρά, μια σειρά λευκού θορύβου είναι στάσιμη - δεν έχει σημασία πότε γίνεται η παρατήρηση, θα πρέπει να μοιάζει σχεδόν ίδια σε οποιαδήποτε χρονική στιγμή.



Σχήμα 2.2: Στο παραπάνω σχήμα συγκρίνονται γραφικά η τυπική μορφή μιας στάσιμης χρονοσειράς, με διαφορετικούς τύπους σειρών που μπορεί να υπάρχουν και μπορούν να μετασχηματιστούν για να προκύψει μια στάσιμη χρονοσειρά [32].

2.2.3 Διαφόριση (differencing)

Με τον όρο διαφóριση (differencing) ορίζουμε τον υπολογισμό της διαφοράς μεταξύ διαδοχικών παρατηρήσεων μιας χρονοσειράς. Η διαφóριση μπορεί να βοηθήσει στη σταθεροποίηση του μέσου όρου μιας χρονοσειράς αφαιρώντας τις μεταβολές στο επίπεδο μιας χρονοσειράς και συνεπώς εξαλείφοντας (ή μειώνοντας) την τάση και την εποχικότητα. Συχνά η διαφóριση χρησιμοποιείται για την μετατροπή χρονοσειράς από μη-στάσιμη σε στάσιμη. Ενίοτε τα διαφορίσιμα δεδομένα δεν φαίνονται να είναι στάσιμα και μπορεί να είναι απαραίτητο να διαφοροποιηθούν τα δεδομένα πολλαπλές φορές, ώστε να προκύψει μια στάσιμη σειρά. Σε περιπτώσεις έντονης εποχικότητας στην χρονοσειρά, ενδείκνυται να χρησιμοποιηθεί μια σειρά εποχικής διαφοροποίησης (seasonal differencing): διαφορά μεταξύ μιας παρατήρησης και της προηγούμενης παρατήρησης της ίδιας περιόδου.

2.3 Αποσύνθεση Χρονοσειρών

Η αποσύνθεση χρονοσειρών αναφέρεται φυσικά στην διάσπαση της αρχικής χρονοσειράς στις προαναφερόμενες συνιστώσες της.

$$Y_t = f(S_t, T_t, C_t, R_t)$$

- Y_t : παρατήρηση κατά τη χρονική περίοδο t .
- S_t : συνιστώσα εποχικότητας.
- T_t : συνιστώσα τάσης.
- C_t : συνιστώσα κύκλου.
- R_t : συνιστώσα τυχαιότητας.

Μια χρονοσειρά, επομένως, είναι ένας συνδυασμός f των συνιστωσών του. Οι τρόποι με τους οποίους μπορούν να συνδυαστούν είναι:

- Αθροιστικά: $Y_t = S_t + T_t + R_t + C_t$.
- Πολλαπλασιαστικά: $Y_t = S_t \times T_t \times R_t \times C_t$.

Η αποσύνθεση παρέχει έναν δομημένο τρόπο σκέψης σχετικά με ένα πρόβλημα πρόβλεψης χρονοσειρών, τόσο γενικά όσον αφορά την πολυπλοκότητα της μοντελοποίησης όσο και ειδικά όσον αφορά τον καλύτερο τρόπο καταγραφής κάθε μιας από αυτές τις συνιστώσες σε ένα συγκεκριμένο μοντέλο. Κάθε μία από αυτές τις συνιστώσες μπορεί να χρειαστεί να αξιοποιηθεί ή και να αντιμετωπιστεί κατά την προετοιμασία των δεδομένων ή την επιλογή και ρύθμιση του μοντέλου.

Κινητός μέσος όρος (Moving Average) Με την χρήση αυτής της μεθόδου γίνεται υπολογισμός του μέσου όρου των τιμών γύρω από μία παρατήρηση. Συνεισφέρει στην εξάλειψη μέρους της τυχαιότητας των δεδομένων και δίνει μία λογική εκτίμηση της συνιστώσας τάσης-κύκλου της παρατήρησης. Η μέθοδος αυτή έχει διάφορες μορφές, με κύρια και ενδεικτική την απλή moving average της μορφής:

$$TC_t = KMO(n)_t = \frac{1}{n} \sum_{i=-(n \text{ mod } 2)}^{n \text{ mod } 2} Y_{t+i}$$

Κλασική Μέθοδος Αποσύνθεσης αποτελεί την πιο απλή διαδικασία για την απομόνωση των 4 συνιστωσών μιας χρονοσειράς. Γίνεται η υπόθεση ότι ο συνδυασμός των συντελεστών της χρονοσειράς γίνεται πολλαπλασιαστικά, άρα είναι της μορφής:

$$Y_t = S_t \times T_t \times R_t \times C_t$$

Βήμα 1 Αρχικά, υπολογίζεται ο κινητός μέσος όρος n παρατηρήσεων, ανάλογα την φύση των δεδομένων (για παράδειγμα, κινητός μέσος όρος 12 παρατηρήσεων για ετήσια δεδομένα). Έτσι, εξαλείφεται (όπως αναφέρθηκε στην παράγραφο 2.3) η τυχαιότητα, ενώ η προσεκτική επιλογή του πλήθους των παρατηρήσεων, αγνοεί πιθανή εποχικότητα της χρονοσειράς. Απαλείφοντας τις δυο αυτές συνιστώσες, παρατηρείται ότι ο ΚΜΟ δίνει μια καλή εκτίμηση της συμπεριφοράς της χρονοσειράς όσον αφορά την τάση και την κυκλικότητα:

$$KMO(n) = T \times C$$

Βήμα 2 Στην συνέχεια, γίνεται διαίρεση των πραγματικών δεδομένων με τις αντίστοιχες τιμές των ΚΜΟ. προκύπτοντας έτσι οι λόγοι εποχικότητας, οι οποίοι, όμως, περιέχουν τυχαιότητα:

$$\Lambda E(n) = \frac{Y}{KMO(n)} = \frac{S \times T \times R \times C}{T \times C} = S \times R$$

Βήμα 3 Σκοπός είναι η απαλοιφή της τυχαιότητας από τους λόγους εποχικότητας του Βήματος 2. Αυτό γίνεται με τον υπολογισμό της μέσης τιμής των λόγων εποχικότητας που αναφέρονται σε αντίστοιχες χρονικές περιόδους. Οι υπολογισμένοι μέσοι όροι αποτελούν τους δείκτες εποχικότητας της αρχικής χρονοσειράς.

$$\Delta E(n) = \frac{\Lambda E(n)}{R} = \frac{S \times R}{R} = S$$

Βήμα 4 Σε αυτό το βήμα, γίνεται διαίρεση των πραγματικών δεδομένων με τους αντίστοιχους δείκτες εποχικότητας που υπολογίστηκαν στο βήμα 3, για την εύρεση της αποεποχικοποιημένης σειράς.

$$\frac{Y}{\Delta E(n)} = \frac{S \times T \times R \times C}{S} = T \times R \times C$$

Βήμα 5 Ακολουθεί απαλοιφή της τυχαιότητας από την απο-εποχικοποιημένη σειρά, ξανά με την χρήση του κατάλληλου ΚΜΟ.

$$\frac{T \times R \times C}{T \times C} = R$$

Βήμα 6 Αν κριθεί αναγκαίος, γίνεται ο υπολογισμός της τάσης από τη σειρά τάσης-κύκλου του Βήματος 5. Αυτό το βήμα απαιτεί την επιλογή του μοντέλου τάσης που αντιπροσωπεύει περισσότερο τη χρονοσειρά. Υποθέτοντας για παράδειγμα μια γραμμική τάση, γίνεται χρήση της γραμμικής παλινδρόμησης (*linear regression*).

Βήμα 7 Τέλος, για την εύρεση της κυκλικότητας, το μόνο που χρειάζεται είναι μια απλή διαίρεση της σειράς τάσης κύκλου με την τάση που υπολογίστηκε στο προηγούμενο βήμα:

$$\frac{T \times C}{T} = C$$

2.4 Μέθοδοι Προβλέψεων

Η πρόβλεψη είναι μια τεχνική που χρησιμοποιεί ιστορικά δεδομένα ως δεδομένα εισόδου για την πραγματοποίηση εκτιμήσεων για την κατεύθυνση των μελλοντικών τάσεων. Ενώ υπάρχουν πολλές μέθοδοι πρόβλεψης, η κατάλληλη επιλογή της δεν είναι μια τετριμμένη διαδικασία.

2.4.1 Στατιστική Πρόβλεψη

Η στατιστική πρόβλεψη (*statistical forecasting*) συνεπάγεται τη χρήση στατιστικών στοιχείων που βασίζονται σε ιστορικά δεδομένα για την πρόβλεψη του τι θα μπορούσε να συμβεί στο μέλλον. Είναι απλές διαδικασίες, καθώς μπορούν να εφαρμοστούν άμεσα, σχετικά γρήγορα και δεν προϋποθέτουν στατιστικές γνώσεις προκειμένου για την παραγωγή προβλέψεων (όταν χρησιμοποιούνται σαν “black box”). Η αδυναμία τους, όμως, βρίσκεται στην προϋπόθεση ότι η συμπεριφορά της χρονοσειράς στο παρελθόν, θα συνεχιστεί και στο μέλλον. Συνεπώς, αγνοεί εξωτερικές ενέργειες που εν δυνάμει επηρεάζουν την πορεία της.

2.4.2 Κριτική Πρόβλεψη

Με τον όρο κριτική πρόβλεψη (*judgemental forecasting*) αναφερόμαστε σε μια είδος πρόβλεψη που γίνεται με βάση υποκειμενικά δεδομένα. Τέτοιες προβλέψεις γίνονται συνή-

θως από ένα άτομο που θεωρείται ότι γνωρίζει καλά την πηγή που αφορά η χρονοσειρά για την οποία γίνεται η πρόβλεψη. Μπορεί να λαμβάνει υπόψη ποσοτικές πληροφορίες, αλλά βασίζεται σε μεγάλο βαθμό σε υποκειμενικό συναίσθημα. Προφανώς, τέτοιες προβλέψεις έχουν το μεγαλύτερο μειονέκτημά την ανθρώπινη προκατάληψη (αισιοδοξία/απαισιοδοξία) Συνεπώς, για μια τέτοια αναξιόπιστη μέθοδο, υπάρχουν τρία γενικά πλαίσια στα οποία χρησιμοποιείται η πρόβλεψη με βάση την κρίση:

- δεν υπάρχουν διαθέσιμα στοιχεία, οπότε οι στατιστικές μέθοδοι δεν είναι εφαρμόσιμες και η πρόβλεψη κατά κρίση είναι η μόνη εφικτή προσέγγιση.
- υπάρχουν διαθέσιμα στοιχεία, παράγονται στατιστικές προβλέψεις, οι οποίες στη συνέχεια προσαρμόζονται με τη χρήση κρίσης.
- υπάρχουν διαθέσιμα στοιχεία και παράγονται στατιστικές και κριτικές προβλέψεις ανεξάρτητα και στη συνέχεια συνδυάζονται.

2.4.3 Πρόβλεψη Προϋπολογισμού

Η συγκεκριμένη μέθοδος πρόβλεψης είναι περισσότερο μια φιλόδοξη στοχοθεσία για την πορεία της χρονοσειράς, παρά μια κανονική παραγωγή προβλέψεως. Παρ'όλα αυτά, η πρόβλεψη προϋπολογισμού (budget forecasting) βοηθά στην ποσοτικοποίηση των προσδοκιών και δημιουργεί μια βάση για τη σύγκριση των πραγματικών αποτελεσμάτων, ώστε να διαπιστωθεί κατά πόσο τα αποτελέσματα διαφέρουν από την αναμενόμενη απόδοση.

Αξίζει να αναφερθεί ονομαστικά, ότι ο συνδυασμός των παραπάνω μεθόδων για την πρόβλεψη των δεδομένων μιας χρονοσειράς καθορίζει την **τελική πρόβλεψη**.

2.4.4 Ορίζοντας Πρόβλεψης

Ο ορίζοντας πρόβλεψης (forecast horizon) είναι το χρονικό διάστημα στο μέλλον για το οποίο πρόκειται να προβλεφθούν οι τιμές της χρονοσειράς. Ανάλογα την τιμή του ορίζοντα πρόβλεψης, επιλέγεται και η κατάλληλη στατιστική μέθοδος πρόβλεψης. Οι πιθανοί ορίζοντες πρόβλεψης χωρίζονται σε τρεις κατηγορίες:

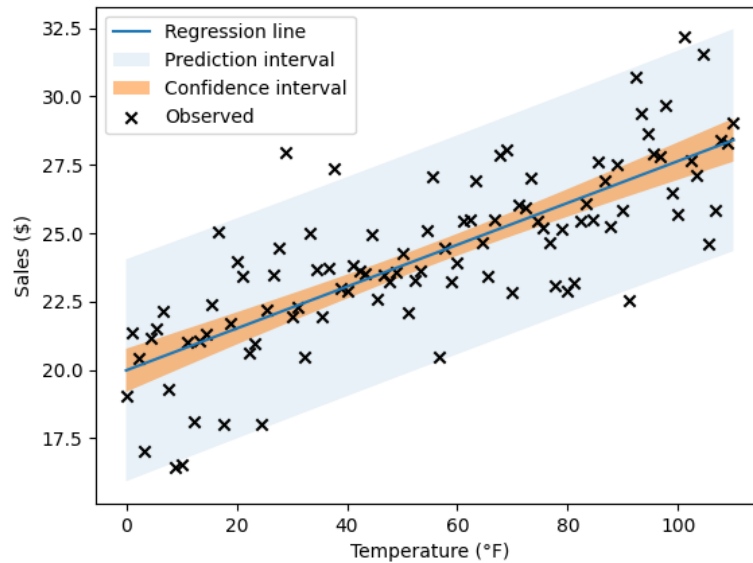
1. **Βραχυπρόθεσμη πρόβλεψη:** Είναι συνήθως μικρότερη των 3 μηνών αλλά έχει χρονική διάρκεια έως και 1 έτος στις αγορές για προγράμματα ή αναθέσεις εργασίας.

2. **Μεσοπρόθεσμη πρόβλεψη:** Διάρκειας συνήθως από 3 μήνες έως 1 έτος, αλλά έχει χρονικό εύρος από ένα έως τρία έτη. Χρησιμοποιείται για τον προγραμματισμό πωλήσεων, τον προγραμματισμό παραγωγής, τον ταμειακό προϋπολογισμό κ.ο.κ.
3. **Μακροπρόθεσμη πρόβλεψη:** Χρονικό εύρος τριών ή περισσότερων ετών και χρησιμοποιείται για το σχεδιασμό και την εγκατάσταση νέων εγκαταστάσεων.

2.4.5 Διαστήματα Εμπιστοσύνης/Πρόβλεψης

Είναι σημαντικό στην πρόβλεψη των δεδομένων της χρονοσειράς η αξιοποίηση των διαστημάτων εμπιστοσύνης (confidence intervals). Το διάστημα εμπιστοσύνης είναι το εύρος των τιμών που αναμένετε ότι θα βρίσκεται η εκτίμησή/πρόβλεψη ή γενικά μια παράμετρος (όπως για παράδειγμα μια παράμετρος του μοντέλου μας). Ένα διάστημα εμπιστοσύνης είναι ποσοστιαίο: αν κατασκευαστεί ένα διάστημα εμπιστοσύνης με επίπεδο εμπιστοσύνης (confidence level) 95%, υπάρχει βεβαιότητα ότι 95 στις 100 φορές η εκτίμηση θα βρίσκεται μεταξύ της ανώτερης και της κατώτερης τιμής που καθορίζεται από το διάστημα εμπιστοσύνης.

Ένα ενδιαφέρον είδος διαστήματος πρόβλεψης είναι το διάστημα πρόβλεψης (prediction interval), το οποίο έχει ως στόχο την πρόβλεψη του επόμενου σημείου που θα επιλεγθεί (sampled) από ένα πληθυσμό, λαμβάνοντας υπόψιν τα προηγούμενα δείγματα που επιλέχθηκαν. Προφανώς, έχοντας μεγαλύτερο πλήθος πιθανών τιμών, είναι σύνηθες το εύρος του διαστήματος εμπιστοσύνης να είναι μικρότερο του διαστήματος πρόβλεψης



Σχήμα 2.3: Στο παραπάνω πληθυσμό πωλήσεων με βάση την θερμοκρασία μιας περιοχής και κάνουμε χρήση της ενός μοντέλου γραμμικής παλινδρόμησης για την εύρεση μέσης τιμής πωλήσεων ανά θερμοκρασία. Το διάστημα εμπιστοσύνης δείχνει το εύρος του μέσου όρου των πωλήσεων σε μια δεδομένη θερμοκρασία, ενώ το διάστημα πρόβλεψης δείχνει το εύρος των πιθανών πωλήσεων σε μια πραγματική ημέρα με δεδομένη την θερμοκρασία [33]

2.5 Τεχνικές Προβλέψεων

2.5.1 Αφελής Μέθοδος

Η πιο απλή στατιστική μέθοδος. Δεν παράγει ακριβείς προβλέψεις αλλά πολλές φορές χρησιμοποιείται ως benchmark για άλλες μεθόδους. Η πρόβλεψη θεωρείται πως είναι ίση με την τελευταία παρατήρηση της διαθέσιμης χρονοσειράς.

$$F(t + 1) = Y(t)$$

2.5.2 Εκθετική Εξομάλυνση

Η εκθετική εξομάλυνση προτάθηκε στα τέλη της δεκαετίας του 1950 (Brown, 1959- Holt, 1957- Winters, 1960) και αποτέλεσε το κίνητρο για ορισμένες από τις πιο επιτυχημένες μεθόδους πρόβλεψης. Οι προβλέψεις παράγονται με μεθόδους εκθετικής εξομάλυνσης είναι

σταθμισμένοι μέσοι όροι παρελθοντικών παρατηρήσεων, με τα βάρη να φθίνουν εκθετικά καθώς οι παρατηρήσεις μεγαλώνουν. Με άλλα λόγια, όσο πιο πρόσφατη είναι η παρατήρηση τόσο μεγαλύτερο είναι το σχετικό βάρος. Οι μέθοδοι εξομάλυνσης είναι κατάλληλες για βραχυπρόθεσμες προβλέψεις ενός μεγάλου όγκου χρονοσειρών, ενώ αποδίδουν καλύτερα σε δεδομένα που παρουσιάζουν στασιμότητα ή μικρό ρυθμό ανάπτυξης ή μείωσης ως προς το χρόνο. Ανάλογα την την τάση της χρονοσειράς προς μελέτη, έχουμε διαφορετικούς τύπους που χρησιμοποιούνται.

Απλή Εκθετική Εξομάλυνση Χρησιμοποιείται για δεδομένα σταθερού επιπέδου: χωρίς ξεκάθαρη τάση ή εποχιακό μοτίβο. Σε κάθε βήμα t , υπολογίζουμε το σφάλμα της πρόβλεψης μας F_t , συγκριτικά με την πραγματική τιμή της χρονοσειράς Y_t . Στην συνέχεια, υπολογίζουμε την επόμενη πρόβλεψη F_{t+1} ως ένα συνδυασμό της προηγούμενης πρόβλεψης και της προηγούμενης παρατήρησης. Ανάλογα την τιμή ενός συντελεστή α , δίνεται περισσότερη σημασία στην προηγούμενη παρατήρηση ή πρόβλεψη.

Εκθετική εξομάλυνση γραμμικής τάσης (Holt) Αποτελεί μια προέκταση της απλής εκθετικής εξομάλυνσης (simple exponential smoothing), στην οποία προστίθεται ένα παραπάνω μοντέλο SES για τον προσδιορισμό της τάσης (είτε ανοδικής, είτε καθοδικής) της χρονοσειράς.

Εκθετική εξομάλυνση μη φθίνουσας τάσης (damping) Οι προβλέψεις που δημιουργούνται με τη γραμμική μέθοδο του Holt εμφανίζουν μια σταθερή τάση (αυξανόμενη ή μειούμενη) επ' αόριστον στο μέλλον. Εμπειρικά στοιχεία δείχνουν ότι αυτές οι μέθοδοι τείνουν να υπερπροβλέπουν (overfit), ειδικά για μεγαλύτερους ορίζοντες πρόβλεψης. Με αυτή την αφορμή, εισήχθηκε μια παράμετρος ϕ που "αποσβένει" την τάση σε μια επίπεδη γραμμή κάποια στιγμή στο μέλλον.

Η εξίσωση 2.1 είναι ο γενικός τύπος για την αποσβεστική εκθετική εξομάλυνση:

$$e_t = Y_t - F_t \quad (2.1)$$

$$S_t = S_{T-1} + \phi \times T_{t-1} + \alpha \times e_t \quad (2.2)$$

$$T_t = \phi \times T_{t-1} + \alpha \times \beta \times e_t \quad (2.3)$$

$$F_{t+1} = F_t + \phi \times T_t \quad (2.4)$$

όπου: (2.5)

- α : συντελεστής βαρύτητας πρόβλεψης & παρατήρησης ($0 < \alpha < 1$).
- β : συντελεστής τάσης χρονοσειράς ($0 < \beta < \alpha$).
- ϕ : συντελεστής απόσβεσης (damping).

Αξίζει να σημειωθεί, ότι η συγκεκριμένη μέθοδος, ως και η πιο γενική, μπορεί να χρησιμοποιηθεί ως οποιαδήποτε προηγούμενη με την κατάλληλη τιμή του συντελεστή απόσβεσης. Πιο συγκεκριμένα:

- $\phi = 0$: Απλή Εκθετική Εξομάλυνση.
- $\phi < 1$: Εκθετική εξομάλυνση φθίνουσας τάσης.
- $\phi \approx 1$: Εκθετική εξομάλυνση γραμμικής τάσης.
- $\phi > 1$: Εκθετική εξομάλυνση εκθετικής τάσης.

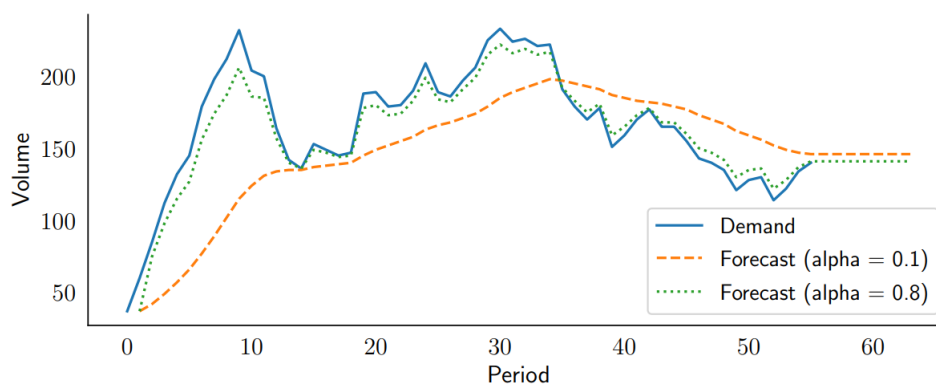


Figure 3.2: Simple smoothing

Σχήμα 2.4: Σε ένα απλό παράδειγμα SES, παρατηρούμε ότι η πρόβλεψη της χρονοσειράς είναι πιο αδιάφορη στις απότομες αλλαγές της χρονοσειράς, όταν η παράμετρος α είναι σημαντικά μικρότερη του 1 (μας ενδιαφέρει πιο πολύ το σφάλμα της πρόβλεψης, από ότι οι προηγούμενες παρατηρήσεις) [31].

2.5.3 Μοντέλα Παλινδρόμησης

Στην γενική μορφή τους, τα μοντέλα παλινδρόμησης περιγράφουν τη σχέση μεταξύ μεταβλητών με την προσαρμογή μιας γραμμής στα παρατηρούμενα δεδομένα. Τα μοντέλα γραμμικής παλινδρόμησης χρησιμοποιούν μια ευθεία γραμμή, ενώ τα μοντέλα λογιστικής και μη γραμμικής παλινδρόμησης χρησιμοποιούν μια καμπύλη γραμμή. Οι μέθοδοι αυτοί, επιτρέπουν την εκτίμηση της μεταβολής μιας εξαρτημένης μεταβλητής καθώς μεταβάλλονται οι ανεξάρτητες μεταβλητές.

Απλή Γραμμική Παλινδρόμηση: χρησιμοποιείται για την εκτίμηση της σχέσης μεταξύ δύο ποσοτικών μεταβλητών (για παράδειγμα, η ζήτηση σε ένα δίκτυο και η ώρα της μέρας). Συνήθως, χρησιμοποιείται όταν επιθυμείται να εξεταστεί:

- Η ισχύς της σχέσης μεταξύ των δύο μεταβλητών.
- η τιμή της εξαρτημένης σε μια ορισμένη τιμή της ανεξάρτητης μεταβλητής.

Βασική προϋπόθεση για την εφαρμογή της απλής γραμμικής παλινδρόμησης είναι ότι η τιμή μιας μεταβλητής εξαρτάται από την τιμή ή τη μεταβολή της τιμής κάποιας άλλης. Συχνά όμως δύο μεταβλητές μπορεί να σχετίζονται χωρίς να μπορεί να θεωρηθεί πως η τιμή της μίας επηρεάζει ή εξαρτάται από την τιμή της άλλης.

Συντελεστής συσχέτισης r : αποτελεί ένα μέτρο του βαθμού συσχέτισης που μπορεί να υπάρχει μεταξύ δύο μεταβλητών. Έχει εύρος τιμών από το -1 έως το 1 όπου:

- $r = 1$: ισχυρή ανάλογη σχέση μεταξύ των δύο μεταβλητών.
- $r = 0$: Δεν υπάρχει καμία σχέση ανάμεσα στις δύο μεταβλητές.
- $r = -1$: ισχυρή αντιστρόφως ανάλογη σχέση μεταξύ δύο μεταβλητών.

Συντελεστής R^2 : Η συσχέτιση των τιμών που προκύπτουν από την εξίσωση της ευθείας παλινδρόμησης και των πραγματικών τιμών συμβολίζεται με R . Στην πράξη η συσχέτιση αυτή χρησιμοποιείται στην τετραγωνική της μορφή και ως εκ τούτου είναι ένας συντελεστής πάντα θετικός ($0 < R^2 < 1$). Πρακτικά, ο συντελεστής R^2 δείχνει πόσο καλά τα δεδομένα ταιριάζουν στο μοντέλο παλινδρόμησης.

$$R^2 = \frac{\sum(\hat{Y}_i - \bar{Y})^2}{\sum(Y_i - \bar{Y})^2} = r_{xy}^2, \hat{Y}_i = \frac{\sum Y_i}{n}, \bar{Y} = a + bY_i$$

Στατιστικοί Δείκτες: Οι στατιστικοί δείκτες χρησιμοποιούνται για να προσδιοριστεί η σημαντικότητα μιας μεταβλητής πρόβλεψης Y έχει στατιστικά σημαντική σχέση με μια μεταβλητή αποτελέσματος X , ή να εκτιμηθεί η διαφορά μεταξύ δύο ή περισσότερων ομάδων (samples). Πιο συγκεκριμένα στην περίπτωση ενός μοντέλου (για παράδειγμα γραμμικής παλινδρόμησης) έχουμε ενδεικτικά:

- **f-test:** Χρησιμοποιείται για να εξεταστεί αν το μοντέλο που δημιουργήθηκε ταιριάζει (fit) με τα πραγματικά δεδομένα.
- **t-test:** προσδιορισμός ύπαρξης (ή μη) γραμμικής συσχέτισης μεταξύ των μεταβλητών απόκρισης Y και πρόβλεψης X .

Στα μοντέλα παλινδρόμησης, χρησιμοποιούνται μετρικές αποστάσεων (distance metrics) για να καθοριστεί ο βαθμός σύγκλισης προς το καλύτερο αποτέλεσμα. Ακόμη και ο ορισμός του "καλύτερου" αποτελέσματος πρέπει να εξηγείται ποσοτικά με κάποια μετρική. Οι πιο συνήθεις μετρικές που χρησιμοποιούνται είναι οι ακόλουθες:

- **Μέσο Σφάλμα (Mean Error):** Εύρεση ύπαρξης (ή μη) συστηματικότητας στο σφάλμα. Όσο πιο μικρή είναι η τιμή του, τόσο μικρότερο είναι προφανώς το σφάλμα.

$$ME = \frac{1}{n} \sum_{i=1}^n (Y_t - F_t)$$

- **Μέσο τετραγωνικό σφάλμα (MSE):** Έλεγχος του το μέτρου της αστοχίας (όχι όμως σχετικά με την κατεύθυνση της). Δίνει μεγαλύτερη βαρύτητα στα μεγάλα σφάλματα και μικρότερη στα μικρά.

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_t - F_t)^2$$

- **Ρίζα μέσου τετραγωνικού σφάλματος (RMSE):** Παρόμοια λειτουργικότητα με τον MSE. Με την χρήση της ρίζας, είναι εκφραζόμενος στις μονάδες της αρχικής χρονοσειράς.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (Y_t - F_t)^2}$$

- **Μέσο απόλυτο ποσοστιαίο σφάλμα (MAPE):** είναι το πιο συνηθισμένο μέτρο που χρησιμοποιείται για την πρόβλεψη του σφάλματος, πιθανώς επειδή οι μονάδες της

μεταβλητής είναι ποσοστιαίες, καθιστά ευκολότερη την κατανόηση. Λειτουργεί καλύτερα αν δεν υπάρχουν ακραίες τιμές στα δεδομένα και μηδενικά (αλγεβρικά υπάρχει ο κίνδυνος απειρισμού του παρονομαστή).

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{Y_t - F_t}{Y_t} \right| \times 100(\%)$$

- **Συμμετρικό μέσο απόλυτο ποσοστιαίο σφάλμα (SMAPE):** Το MAPE είναι ασύμμετρο και επιβάλλει μεγαλύτερη ποινή στα αρνητικά σφάλματα (όταν οι προβλέψεις είναι υψηλότερες από τις πραγματικές) από ό,τι στα θετικά σφάλματα. Έτσι, δημιουργήθηκε το sMAPE για να διορθώσει αυτό το ελάττωμα. Αυτό δεν το καθιστά βέβαια συμμετρικό, καθώς διαχειρίζεται τις απαισιόδοξες και αισιόδοξες προβλέψεις με διαφορετικό τρόπο.

$$sMAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{Y_t - F_t}{\frac{1}{2}(Y_t + F_t)} \right| \times 100(\%)$$

2.6 Μοντέλα AR(I)MA

Μια μοναδική κατηγορία μοντέλων, γνωστή ως μοντέλα AR(I)MA, χρησιμοποιείται για την πρόβλεψη της παρατήρησης τη χρονική στιγμή (t+1) με βάση ιστορικά δεδομένα από προηγούμενα χρονικά σημεία που λαμβάνονται με την ίδια παρατήρηση. Τα βασικά μοντέλα αυτής της κατηγορίας αναλύονται ακολούθως.

2.6.1 Μοντέλο Αυτοπαλινδρόμησης (AR)

Σε ένα μοντέλο πολλαπλής παλινδρόμησης, προβλέπουμε τη μεταβλητή που μας ενδιαφέρει χρησιμοποιώντας έναν γραμμικό συνδυασμό προβλεπτικών παραγόντων. Σε ένα μοντέλο παλινδρόμησης (autoregression), προβλέπουμε τη μεταβλητή ενδιαφέροντος χρησιμοποιώντας έναν γραμμικό συνδυασμό προηγούμενων τιμών της μεταβλητής. Ο όρος παλινδρόμηση υποδηλώνει ότι πρόκειται για παλινδρόμηση της μεταβλητής έναντι του εαυτού της. Έτσι, ένα μοντέλο παλινδρόμησης τάξης p μπορεί να γραφεί ως εξής:

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + e_t$$

όπου e_t είναι λευκός θόρυβος. Αυτό είναι σαν μια πολλαπλή παλινδρόμηση αλλά με καθυστερημένες τιμές των y_t ως προγνωστικούς παράγοντες. Αναφερόμαστε σε αυτό ως AR(p) μοντέλο, ένα παλινδρομικό μοντέλο τάξης p.

2.6.2 Μοντέλο κινητού μέσου όρου (MA)

Επίσης μια παραλλαγή μοντέλου πολλαπλής παλινδρόμησης, με πολλές ομοιότητες με το μοντέλο παλινδρόμησης. Αντί όμως να χρησιμοποιεί παρελθοντικές τιμές της μεταβλητής πρόβλεψης σε μια παλινδρόμηση, ένα μοντέλο κινητού μέσου όρου (moving average model) χρησιμοποιεί παρελθοντικά *σφάλματα* πρόβλεψης για να κάνει μια καλύτερη εκτίμηση για την τρέχουσα χρονική περίοδο. Η εξίσωση ενός μοντέλου κινητού μέσου όρου τάξης q είναι η ακόλουθη:

$$y_t = c + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \dots + \theta_q \epsilon_{t-q} + \epsilon_t$$

όπου ϵ_t είναι λευκός θόρυβος. Αναφερόμαστε σε αυτό ως μοντέλο MA(q), ένα μοντέλο κινητού μέσου όρου τάξης q . Φυσικά, δεν παρατηρούμε τις τιμές του ϵ_t , οπότε δεν πρόκειται πραγματικά για παλινδρόμηση με τη συνήθη έννοια.

Παρατηρείται ότι κάθε τιμή του y_t μπορεί να θεωρηθεί ως ένας σταθμισμένος κινητός μέσος όρος των τελευταίων σφαλμάτων πρόβλεψης. Ωστόσο, τα μοντέλα κινητού μέσου όρου δεν πρέπει να συγχέονται με την εξομάλυνση κινητού μέσου όρου σε προηγούμενο κεφάλαιο. Ένα υπόδειγμα κινητού μέσου χρησιμοποιείται για την πρόβλεψη μελλοντικών τιμών, ενώ η εξομάλυνση κινητού μέσου χρησιμοποιείται για την εκτίμηση του κύκλου τάσης των παρελθουσών τιμών.

ARIMA: Τα μοντέλα ARIMA συμβολίζονται γενικά ως ARIMA (p,d,q) όπου p είναι η τάξη του αυτοπαλινδρομούμενου μοντέλου, d είναι ο βαθμός της διαφοροποίησης (d -στη παράγωγος) και q είναι η τάξη του μοντέλου κινητού μέσου όρου. Εάν συνδυάσουμε τη διαφοροποίηση με νδρόμηση και ένα υπόδειγμα κινητού μέσου όρου, λαμβάνουμε ένα μη εποχικό υπόδειγμα ARIMA. Το πλήρες υπόδειγμα μπορεί να γραφεί ως εξής:

$$y'_t = c + \phi_1 y'_{t-1} + \dots + \phi_p y'_{t-p} + e_t \theta_1 \epsilon_{t-1} + \dots + \theta_q \epsilon_{t-q} + \epsilon_t$$

όπου y'_t είναι η διαφοροποιημένη σειρά (μπορεί να έχει διαφοροποιηθεί περισσότερες από μία φορές). Οι "προγνωστικοί παράγοντες" (predictors) στη δεξιά πλευρά περιλαμβάνουν τόσο τις καθυστερημένες τιμές του y_t όσο και τα καθυστερημένα σφάλματα.

2.7 Καταλογισμός (Imputation)

Πολλά σύνολα δεδομένων του πραγματικού κόσμου μπορεί να περιέχουν ελλιπείς τιμές για διάφορους λόγους. Συχνά κωδικοποιούνται ως NaNs, κενά ή οποιαδήποτε άλλα μέσα τοποθέτησης. Η εκπαίδευση ενός μοντέλου με ένα σύνολο δεδομένων που έχει πολλές ελλείπουσες τιμές μπορεί να επηρεάσει δραστικά την ποιότητα του μοντέλου μηχανικής μάθησης. Imputation είναι ένα σύνολο τεχνικών που χρησιμοποιούνται για την αντικατάσταση των δεδομένων που λείπουν με κάποια υποκατάστατη τιμή ώστε να διατηρηθούν τα περισσότερα δεδομένα/πληροφορίες του συνόλου δεδομένων.

2.7.1 Πλήρης ανάλυση περίπτωσης (CCA)

Η πλήρης ανάλυση περίπτωσης (complete case analysis) μια αρκετά απλή μέθοδο χειρισμού των ελλιπών δεδομένων, η οποία αφαιρεί άμεσα τις γραμμές που έχουν ελλιπή δεδομένα, δηλαδή εξετάζουμε μόνο τις γραμμές στις οποίες έχουμε πλήρη δεδομένα, δηλαδή τα δεδομένα δεν λείπουν. Η τεχνική αυτή χρησιμοποιείται με τις παραδοχές ότι τα δεδομένα λείπουν τυχαία (MAR) και αφαιρούνται πλήρως από τον πίνακα. Αν και εύκολο στην εφαρμογή, η αφαίρεση των δεδομένων από το σύνολο δεδομένων κάθε φορά δεν είναι εφικτή και μπορεί να οδηγήσει σε μείωση του μεγέθους του συνόλου δεδομένων σε μεγάλο βαθμό. Ταυτόχρονα, επηρεάζεται η κατανομή (distribution) των δεδομένων και προκαλεί προκατάληψη (bias) στα δεδομένα οδηγώντας σε εσφαλμένη ανάλυση του μοντέλου

2.7.2 Καταλογισμός αυθαίρετης τιμής

Πρόκειται για μια σημαντική τεχνική, καθώς μπορεί να χειριστεί τόσο τις αριθμητικές όσο και τις κατηγορικές μεταβλητές. Η τεχνική αυτή δηλώνει ότι ομαδοποιούμε τις ελλείπουσες τιμές σε μια στήλη και τις αντιστοιχίζουμε σε μια νέα (arbitrary) τιμή που απέχει πολύ από το εύρος της εν λόγω στήλης. Είναι επίσης εύκολο στην εφαρμογή, ενώ ταυτόχρονα διατηρεί την πληροφορία ότι τα δεδομένα λείπουν σε εκείνη την θέση. Παρ' όλα αυτά, μπορεί εν δυνάμει να στρεβλώσει την αρχική κατανομή των μεταβλητών, ενώ οι αυθαίρετες τιμές μπορούν να δημιουργήσουν ακραίες τιμές.

2.7.3 Καταλογισμός μέσης τιμής

Ο υπολογισμός του μέσου όρου (mean imputation) είναι μια μέθοδος κατά την οποία η τιμή που λείπει από μια συγκεκριμένη μεταβλητή αντικαθίσταται από τον μέσο όρο των διαθέσιμων περιπτώσεων. Η μέθοδος αυτή διατηρεί το μέγεθος του δείγματος και είναι εύκολη στη χρήση, αλλά η μεταβλητότητα των δεδομένων μειώνεται, οπότε οι τυπικές αποκλίσεις και οι εκτιμήσεις της διακύμανσης τείνουν να υποεκτιμούνται.

2.7.4 Καταλογισμός παλινδρόμησης

Μπορεί να χρησιμοποιηθεί απλή παλινδρόμηση, όπου η τιμή προβλέπεται από μια εξίσωση παλινδρόμησης. Για τη μέθοδο αυτή χρησιμοποιούνται οι πληροφορίες των πλήρων παρατηρήσεων για την πρόβλεψη των τιμών των παρατηρήσεων που λείπουν. Η παλινδρόμηση υποθέτει ότι οι τιμές εμπίπτουν απευθείας σε μια γραμμή παλινδρόμησης με μη μηδενική κλίση, οπότε συνεπάγεται συσχέτιση 1 μεταξύ των προβλεπτικών παραγόντων και της μεταβλητής αποτελέσματος που λείπει. Σε αντίθεση με τη μέθοδο αντικατάστασης μέσων όρων, ο υπολογισμός με παλινδρόμηση θα υπερεκτιμήσει τις συσχετίσεις, ωστόσο, οι διακυμάνσεις και οι συν-διακυμάνσεις υποεκτιμούνται.

2.7.5 Γραμμικός καταλογισμός

Η γραμμική παρεμβολή (linear interpolation) είναι μια τεχνική υπολογισμού που υποθέτει γραμμική σχέση μεταξύ των σημείων δεδομένων και χρησιμοποιεί μη ελλείπουσες τιμές από γειτονικά σημεία δεδομένων για τον υπολογισμό μιας τιμής για ένα σημείο δεδομένων που λείπει. Πολλές φορές, είναι η καλύτερη μια πιο ομαλή (smooth) μετάβαση μεταξύ των γνωστών τιμών και επιλέγονται τεχνικές ομαλής παρεμβολής (smooth interpolation).



Σχήμα 2.5: Τα σχήματα αποτυπώνονται τα διακριτά δεδομένα μας (κόκκινες τελείες) ενώ οι μπλε γραμμές δείχνουν την προσέγγιση της γραμμικής (αριστερά) και ομαλής (δεξιά) παρεμβολής για δεδομένα που βρίσκονται ενδιάμεσα των γνωστών τιμών [34].

Κεφάλαιο 3

Μηχανική Μάθηση

Η μηχανική μάθηση είναι μια εφαρμογή της τεχνητής νοημοσύνης (AI) που επιτρέπει στα συστήματα να μαθαίνουν και να βελτιώνονται από την εμπειρία χωρίς να προγραμματίζονται ρητά. Η μηχανική μάθηση επικεντρώνεται στην ανάπτυξη προγραμμάτων υπολογιστών που μπορούν να έχουν πρόσβαση σε δεδομένα και να τα χρησιμοποιούν για να μαθαίνουν μόνοι τους. Στις παρακάτω ενότητες ακολουθεί μια περιγραφή των κατηγοριών στις οποίες χωρίζεται η μηχανική μάθηση.

3.1 Κατηγορίες Μηχανικής Μάθησης

3.1.1 Επιβλεπόμενη μάθηση

Στην επιβλεπόμενη μάθηση (supervised learning), η μηχανή διδάσκεται μέσω παραδείγματος. Ο χειριστής παρέχει στον αλγόριθμο μηχανικής μάθησης ένα γνωστό σύνολο δεδομένων που περιλαμβάνει τις επιθυμητές εισόδους (features) και εξόδους (labels) και ο αλγόριθμος πρέπει να βρει μια μέθοδο για να καθορίσει πώς θα καταλήξει σε αυτές τις εισόδους και εξόδους. Μαθηματικώς, ορίζεται ως εξής:

Δεδομένου ενός χώρου παραδειγμάτων (feature space) X και ενός χώρου ετικετών (label space) Y , υπάρχει κάποιος στόχος συνάρτηση $y = f(x)$ όπου $y \in Y$, έτσι ώστε για κάθε $x \in X$, η συνάρτηση αυτή να εξάγει το σωστό y στο χώρο ετικετών. Ένας επιβλεπόμενος μαθητής ασχολείται με ένα σύστημα όπου η $f(x)$ μαθαίνεται από ένα δεδομένο σύνολο ζευγών (x, y) . Θέλουμε αυτή τη μάθηση όταν δεν γνωρίζουμε τι είναι η f . Ο στόχος μας είναι να βρούμε ένα συνάρτηση $g(x)$ που είναι κοντά (κατά προτίμηση πανομοιότυπη) στην $f(x)$ [35]

Ενώ ο χειριστής γνωρίζει τις σωστές απαντήσεις στο πρόβλημα, ο αλγόριθμος εντοπίζει

μοτίβα στα δεδομένα, μαθαίνει από τις παρατηρήσεις και κάνει προβλέψεις. Ο αλγόριθμος κάνει προβλέψεις και διορθώνεται από τον χειριστή - και αυτή η διαδικασία συνεχίζεται έως ότου ο αλγόριθμος επιτύχει ένα υψηλό επίπεδο ακρίβειας/απόδοσης. Κάτω από την ομπρέλα της επιβλεπόμενης μάθησης εμπίπτουν οι ακόλουθες υποκατηγορίες:

1. **Classification (Ταξινόμηση):** Το πρόγραμμα μηχανικής μάθησης πρέπει να βγάλει ένα συμπέρασμα από τις παρατηρούμενες τιμές και να καθορίσει αν σε ποια κατηγορία ανήκουν οι νέες παρατηρήσεις.
2. **Regression (Παλινδρόμηση):** Το πρόγραμμα μηχανικής μάθησης πρέπει να εκτιμήσει - και να κατανοήσει - τις σχέσεις μεταξύ των μεταβλητών. Η ανάλυση παλινδρόμησης επικεντρώνεται σε μια εξαρτημένη μεταβλητή και μια σειρά άλλων μεταβαλλόμενων μεταβλητών - γεγονός που την καθιστά ιδιαίτερα χρήσιμη για την πρόβλεψη και την πρόγνωση.
3. **Forecasting (Πρόβλεψη):** Η διαδικασία πραγματοποίησης προβλέψεων για το μέλλον με βάση τα δεδομένα του παρελθόντος και του παρόντος και χρησιμοποιείται συνήθως για την ανάλυση τάσεων.

3.1.2 Ημιεπιβλεπόμενη μάθηση

Η ημιεπιβλεπόμενη μάθηση (semi-supervised) είναι παρόμοια με την επιβλεπόμενη μάθηση, αλλά χρησιμοποιεί δεδομένα με και χωρίς ετικέτες (labels). Τα επισημασμένα δεδομένα είναι ουσιαστικά πληροφορίες που έχουν σημαντικές ετικέτες, ώστε ο αλγόριθμος να μπορεί να κατανοήσει τα δεδομένα, ενώ τα unsupervised δεδομένα δεν διαθέτουν αυτές τις πληροφορίες. Με τη χρήση αυτών των συνδυασμών, οι αλγόριθμοι μηχανικής μάθησης μπορούν να μάθουν να επισημαίνουν δεδομένα χωρίς ετικέτες.

3.1.3 Μη-επιβλεπόμενη μάθηση

Εδώ, ο αλγόριθμος μηχανικής μάθησης μελετά τα δεδομένα για τον εντοπισμό μοτίβων, χωρίς να υπάρχει ανθρώπινος χειριστής για την παροχή οδηγιών. Αντ' αυτού, η μηχανή καθορίζει τους συσχετισμούς και τις σχέσεις αναλύοντας τα διαθέσιμα δεδομένα. Σε μια διαδικασία μάθησης χωρίς επίβλεψη, ο αλγόριθμος μηχανικής μάθησης αφήνεται να ερμηνεύσει μεγάλα σύνολα δεδομένων και να αντιμετωπίσει τα δεδομένα αυτά αναλόγως. Ο αλγόριθμος

προσπαθεί να οργανώσει αυτά τα δεδομένα με κάποιο τρόπο για να περιγράψει τη δομή τους. Αυτό μπορεί να σημαίνει την ομαδοποίηση των δεδομένων σε ομάδες (clusters) ή την τακτοποίησή τους με τρόπο που να φαίνεται πιο οργανωμένος. Καθώς αξιολογεί περισσότερα δεδομένα, η ικανότητά του να λαμβάνει αποφάσεις για τα δεδομένα αυτά βελτιώνεται σταδιακά και γίνεται πιο εκλεπτυσμένη. Κάτω από την ομπρέλα της μάθησης χωρίς επίβλεψη, εμπίπτουν:

1. **Clustering (Ομαδοποίηση):** Η ομαδοποίηση περιλαμβάνει την ομαδοποίηση συνόλων παρόμοιων δεδομένων (βάσει καθορισμένων κριτηρίων). Είναι χρήσιμη για την κατάτμηση δεδομένων σε διάφορες ομάδες και την εκτέλεση ανάλυσης σε κάθε σύνολο δεδομένων για την εύρεση μοτίβων (e.g K Means Clustering).
2. **Anomaly Detection (Ανίχνευση ανωμαλιών):** Η ανωμαλία είναι ένα συνώνυμο για "ακραίο φαινόμενο" (outlier). Η ανίχνευση ανωμαλιών (ή ανίχνευση ακραίων τιμών) είναι ο εντοπισμός σπάνιων στοιχείων, γεγονότων ή παρατηρήσεων που δημιουργούν υποψίες επειδή διαφέρουν σημαντικά από την πλειονότητα των δεδομένων. Οι ανωμαλίες δραστηριότητες μπορεί να συνδέονται με κάποιου είδους προβλήματα ή σπάνια γεγονότα.
3. **Dimension reduction (Μείωση διαστάσεων):** Η μείωση των διαστάσεων μειώνει τον αριθμό των μεταβλητών που εξετάζονται για να βρεθούν οι ακριβείς πληροφορίες που απαιτούνται.

Ενισχυτική μάθηση

Η ενισχυτική μάθηση (reinforcement learning) επικεντρώνεται σε οργανωμένες διαδικασίες μάθησης, όπου σε έναν αλγόριθμο μηχανικής μάθησης παρέχεται ένα σύνολο ενεργειών, παραμέτρων και τελικών τιμών. Καθορίζοντας τους κανόνες, ο αλγόριθμος μηχανικής μάθησης προσπαθεί στη συνέχεια να εξερευνήσει διάφορες επιλογές και δυνατότητες, παρακολουθώντας και αξιολογώντας κάθε αποτέλεσμα για να καθορίσει ποιο είναι το βέλτιστο. Η ενισχυτική μάθηση διδάσκει στη μηχανή τη δοκιμή και το σφάλμα. Μαθαίνει από προηγούμενες εμπειρίες και αρχίζει να προσαρμόζει την προσέγγισή της ανάλογα με την κατάσταση, ώστε να επιτύχει το καλύτερο δυνατό αποτέλεσμα (e.g video games, robotics, self-driving cars)

3.2 Νευρωνικά Δίκτυα

Ένα Τεχνητό Νευρωνικό Δίκτυο (Artificial Neural Network) ή απλά ένα Νευρωνικό Δίκτυο (NN) είναι διασυνδεδεμένα στρώματα μικρών μονάδων που ονομάζονται κόμβοι (nodes) και εκτελούν μαθηματικές πράξεις για την ανίχνευση μοτίβων στα δεδομένα. Τα νευρωνικά δίκτυα προσπαθούν να προσδιορίσουν τις παραμέτρους μίας γενικευμένης μη γραμμικής εξίσωσης η οποία καλείται να μοντελοποιήσει την σχέση μεταξύ των μεταβλητών εισόδου και εξόδου. Λόγω της αύξησης του πλήθους διαθέσιμων δεδομένων, της μεγαλύτερης διαθέσιμης υπολογιστής ισχύς και της καλύτερης απόδοσης των αλγορίθμων που αξιοποιούνται για την εκπαίδευση και διαχείρισή τους, η χρήση των νευρωνικών δικτύων αυξήθηκε εκθετικά τα τελευταία χρόνια. Τα νευρωνικά δίκτυα μπορούν να χρησιμοποιηθούν σε εφαρμογές supervised, unsupervised και semi-supervised learning.

3.2.1 Τεχνητός Νευρώνας Perceptron

Πρόκειται για το βασικό δομικό στοιχείο ενός NN. Λαμβάνει σταθμισμένες τιμές, εκτελεί μαθηματικούς υπολογισμούς και παράγει έξοδο. Ονομάζεται επίσης μονάδα, κόμβος ή perceptron. Στην γενική περίπτωση ένας perceptron έχει n εισόδους και μία έξοδο. Τα δεδομένα εισόδου πολλαπλασιάζονται με τα αντίστοιχα βάρη (weights) τους, και τα γινόμενα αυτά αθροίζονται μεταξύ τους. Στην συνέχεια, προστίθεται στο άθροισμα μια γνωστή σταθερή τιμή ονόματι *προκατάληψη* (bias), με σκοπό την επιτάχυνση ή την καθυστέρηση της ενεργοποίησης ενός συγκεκριμένου κόμβου.

Το αποτέλεσμα όλης αυτής της διαδικασίας, λειτουργεί σαν τιμή εισόδου για την συνάρτηση ενεργοποίησης (activation function). Τέλος, η έξοδος της συνάρτησης ενεργοποίησης είναι κι η έξοδος του νευρώνα.

$$y = f\left(\sum_{i=0}^n x_i w_i + b\right)$$

όπου:

- w_i : συνιστώσα διανύσματος βαρών
- x_i : συνιστώσα διανύσματος είσοδο
- b : προκατάληψη (bias)

- f : συνάρτηση ενεργοποίησης
- y : έξοδος του νευρώνα και της συνάρτησης ενεργοποίησης

3.2.2 Συναρτήσεις Ενεργοποίησης

Οι συναρτήσεις ενεργοποίησης αποτελούν σαν μέθοδος προ-επεξεργασίας του εσωτερικού γινομένου είσοδος-βαρών, πριν αποτελέσουν την έξοδο του νευρώνα. Ανάλογα με την επιλογή της (ReLU, sigmoid, tanh), εισάγεται του φαινομένου της μη γραμμικότητας στο σύστημα. Οι βασικότερες συναρτήσεις με βάση την βιβλιογραφία παρατίθενται στη συνέχεια ακολουθούμενες από την μαθηματική τους έκφραση

- **Ταυτοτική Συνάρτηση (Identity Function):** είναι μια συνάρτηση που επιστρέφει πάντα την τιμή που χρησιμοποιήθηκε ως όρισμά της, αμετάβλητη. Αποτελεί την πιο απλή συνάρτηση και χρησιμοποιείται όταν έχει σημασία η ατόφια τιμή του εσωτερικού γινομένου

$$f(x) = x$$

- **Βηματική Συνάρτηση (Step Function):** Πρόκειται ουσιαστικά για έναν ταξινομητή με βάση μια τιμή κατωφλίου. Σε αυτό, αποφασίζουμε κάποια τιμή κατωφλίου (έστω a) για να αποφασίσουμε την έξοδο που ο νευρώνας θα πρέπει να ενεργοποιηθεί ή να απενεργοποιηθεί.

$$f(x) = \begin{cases} 0 & a \leq 1 \\ 1 & a > 1 \end{cases}$$

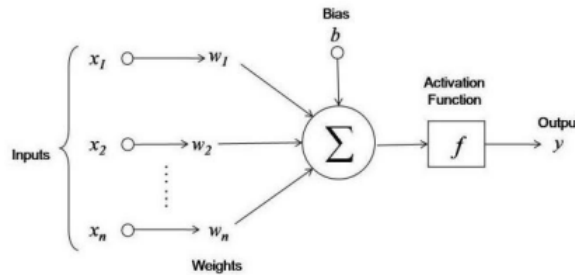
- **Ανορθωμένη Γραμμική Συνάρτηση (Rectified Linear Unit -ReLU-):** Είναι η πιο συχνά χρησιμοποιούμενη συνάρτηση ενεργοποίησης στα νευρωνικά δίκτυα. Η ανορθωμένη γραμμική συνάρτηση ενεργοποίησης ή ReLU είναι μια μη γραμμική συνάρτηση ή γραμμική συνάρτηση που θα εξάγει την είσοδο άμεσα αν είναι θετική, διαφορετικά θα εξάγει μηδέν.

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

- **Σιγμοειδής συνάρτηση (Sigmoid function):** Η σιγμοειδής συνάρτηση είναι μια μαθηματική συνάρτηση που έχει χαρακτηριστική καμπύλη σχήματος S. Όλες οι σιγμοειδείς συναρτήσεις (π.χ softmax, tanh) έχουν την ιδιότητα να αντιστοιχίζουν ολόκληρη την αριθμητική γραμμή σε ένα μικρό εύρος, όπως μεταξύ 0 και 1 ή -1 και 1. Έτσι, μια

χρήση μιας σιγμοειδούς συνάρτησης είναι η μετατροπή μιας πραγματικής τιμής σε μια τιμή που μπορεί να ερμηνευτεί ως πιθανότητα.

$$f(x) = \frac{e^x}{e^x + 1}$$



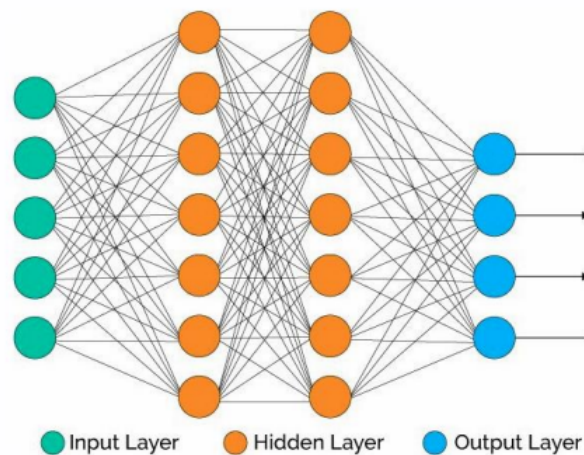
Σχήμα 3.1: Στο παραπάνω σχήμα βλέπουμε σχηματικά την συνάρτηση που καθιστά ένα νευρώνα

3.2.3 Πολυεπίπεδο Perceptron (MLP)

Το Multilayer Perceptron εμπίπτει στην κατηγορία των *feedforward* αλγορίθμων, αποτελείται αποκλειστικά από perceptrons, τα οποία οργανώνονται σε ένα ή περισσότερα στρώματα (layers). Κάθε perceptron ενός στρώματος, τροφοδοτείται με δεδομένα από κάθε perceptron του προηγούμενου layer και τροφοδοτεί με την σειρά του κάθε perceptron του επόμενου layer (forward pass). Μια στρώση που συνδέεται με τέτοιο τρόπο με εκείνη που ακολουθεί ονομάζεται πυκνή (dense).

Είναι πολύ ευέλικτες αρχιτεκτονικές και μπορούν να χρησιμοποιηθούν γενικά για την εκμάθηση μιας αντιστοίχισης από τις εισόδους στις εξόδους. Ταυτόχρονα, η απλότητά τους σαν δίκτυα μας επιτρέπουν να εξετάσουμε με μεγάλη ακρίβεια και δεν απαιτούν τη χρονοβόρα διαδικασία αναζήτησης σχεδιασμού δικτύου που απαιτείται για βαθύτερες και πιο περίπλοκες αρχιτεκτονικές της τελευταίας τεχνολογίας [36], γεγονός που επιτρέπει την ταχεία εγκατάσταση της πειραματικής προσέγγισης. Αντιθέτως, αποδίδουν αρκετά καλά για ένα ευρύ φάσμα εφαρμογών [37–39]. Επιπλέον, τα σχετικά λίγα εκπαιδευσιμα βάρη ενός δικτύου MLP μειώνουν καταλυτικά τον χρόνο εκπαίδευσης, επιτρέποντάς μας να δοκιμάσουμε μια μεγαλύτερη ποικιλία δοκιμών από ό,τι θα μπορούσαμε με μια δομή που ήταν πιο ακριβής από υπολογιστική άποψη.

Ομοίως με τα μοντέλα παλινδρόμησης (regression), χρησιμοποιούμε μετρικές αποστάσεις ή αλλιώς συναρτήσεις κόστους (cost functions) για τον έλεγχο της απόδοσης του μοντέλου. Οι συναρτήσεις αυτές συγκρίνουν το αποτέλεσμα της εξόδου με τις εισόδου και έχουν σαν έξοδο την διαφορά μεταξύ των τιμών τους. Φυσικά, αν ο αλγόριθμος σταματούσε εδώ, δεν θα ήταν σε θέση να μάθει τα βάρη που ελαχιστοποιούν τη συνάρτηση κόστους. Εάν ο αλγόριθμος υπολόγιζε μόνο μία επανάληψη, δεν θα υπήρχε πραγματική μάθηση (learning). Έτσι, εισάγεται η έννοια της οπισθοδιάδοσης (**backpropagation**)



Σχήμα 3.2: Μια μινιμαλιστική αποτύπωση ενός Multilayer Perceptron (MLP)

3.2.4 Backpropagation Algorithm

Η οπισθοδιάδοση είναι ο μηχανισμός μάθησης που επιτρέπει στο Multilayer Perceptron να προσαρμόζει επαναληπτικά τα βάρη στο δίκτυο, με στόχο την ελαχιστοποίηση της συνάρτησης κόστους. Το επίπεδο προσαρμογής καθορίζεται από τις κλίσεις (παράγωγο) της συνάρτησης κόστους ως προς αυτές τις παραμέτρους. Η παράγωγος μιας συνάρτησης μετρά την ευαισθησία στη μεταβολή της τιμής της συνάρτησης (τιμή εξόδου) σε σχέση με μια μεταβολή στο όρισμά της x (τιμή εισόδου) [40]. Με άλλα λόγια, η παράγωγος μας λέει την κατεύθυνση προς την οποία κινείται η συνάρτηση κόστους, και κατ' επέκταση πόσο πρέπει να αλλάξει η παράμετρος (προς θετική ή αρνητική κατεύθυνση) ώστε η συνάρτηση να ελαχιστοποιηθεί.

Αρχικά βρίσκουμε πόση αλλαγή χρειάζονται οι παράμετροι του τελευταίου επιπέδου. Υπάρχουν πολλές μέθοδοι υπολογισμού αυτής της αλλαγής, και αρκετές θα αναφερθούν αναλυτικά στο κεφάλαιο 3.2.5, με την πιο γνωστή να είναι η μέθοδος καθόδου κλήσης (gradient descent). Στην συνέχεια, με την χρήση του κανόνα αλυσίδας του Leibniz (chain rule), ανανεώνουμε τις τιμές των παραμέτρων των υπολοίπων layers για *κάθε* νευρώνα ατομικά, ξεκινώντας από το τέλος προς την αρχή. Αξίζει να αναφερθεί ότι στην περίπτωση των Multilayer Perceptron, συνήθως σε κάθε layer υπάρχουν παραπάνω από ένα νευρώνες, και άρα οι παράμετροι (weights, biases etc) του μοντέλου δεν είναι μοναδιαίες τιμές, αλλά συχνά αποτυπώνονται σε μορφή διανύσματος

Η εξίσωση 3.2.4, αντιπροσωπεύει τον κανόνα αλυσίδας για τον υπολογισμό της *ευαισθησίας* της συνάρτησης κόστους C από την μεταβολή της τιμής του διανύσματος w των βαρών του L -στου layer του μοντέλου

$$\frac{\partial C}{\partial w^{(L)}} = \frac{\partial C}{\partial y^{(L)}} \times \frac{\partial y^{(L)}}{\partial z^{(L)}} \times \frac{\partial z^{(L)}}{\partial w^{(L)}}$$

$$\begin{aligned} y^{(L)} &= f(z^{(L)}) = f(w^{(L)} \times x^L + b^{(L)}) \\ &= f(w^{(L)} \times y^{L-1} + b^{(L)}) \end{aligned}$$

όπου:

- C : Η συνάρτηση κόστους (Cost function)
- f : συνάρτηση ενεργοποίησης
- $y^{(L)}$: Το διάνυσμα εξόδου του L -στου layer του μοντέλου
- $z^{(L)}$: Το διάνυσμα εσωτερικού γινομένου βαρών-εισόδου του L -στου layer του μοντέλου
- $w^{(L)}$: διάνυσμα βαρών του L -στου layer του μοντέλου

Προφανώς, αυτή η διαδικασία είναι υπολογιστικά κοστοβόρα, ειδικά αν επαναλαμβάνεται για κάθε καινούργια είσοδο στο μοντέλο. Γι' αυτό, συχνά τα δεδομένα συνενώνονται συνήθως σε παρτίδες (batches). Με αυτό τον τρόπο αποτιμάται πρώτα η έξοδος όλων των δειγμάτων της παρτίδας κι έπειτα πραγματοποιείται ένα μόνο τρέξιμο της οπισθοδιάδοσης του σφάλματος για όλα τα δείγματα της παρτίδας αυτής

3.2.5 Αλγόριθμοι Βελτιστοποίησης

Οι λεγόμενοι βελτιστοποιητές (optimizers) είναι αλγόριθμοι ή μέθοδοι που χρησιμοποιούνται για την ελαχιστοποίηση μιας συνάρτησης κόστους ή για τη μεγιστοποίηση της αποδοτικότητας της παραγωγής. Οι βελτιστοποιητές είναι μαθηματικές συναρτήσεις που εξαρτώνται από τις παραμέτρους με τις οποίες "μαθαίνει" το μοντέλο, δηλαδή από τα βάρη και τις προκαταλήψεις (biases). Είναι σημαντική η κατάλληλη επιλογή του αλγορίθμου (optimizer) και εξαρτάται ανάλογα με την εφαρμογή και τα δεδομένα που δίνονται.

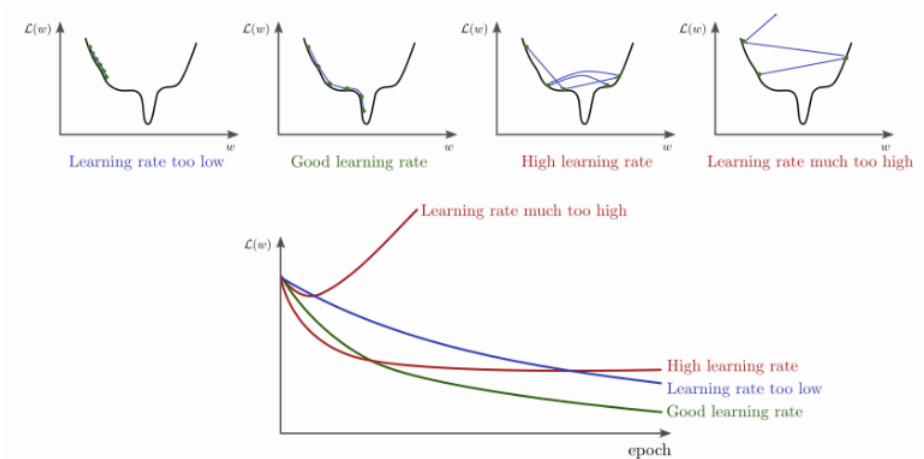
Gradient Descent: Ο αλγόριθμος Gradient Descent είναι ο πιο βασικός αλλά και ο πιο συχνά χρησιμοποιούμενος αλγόριθμος βελτιστοποίησης. Μειώνει επαναληπτικά τη συνάρτηση απωλειών κινούμενος προς την *αντίθετη* κατεύθυνση από αυτή της απότομης ανόδου. Μέσω της οπισθοδιάδοσης, η απώλεια μεταφέρεται από το ένα στρώμα στο άλλο και οι παράμετροι του μοντέλου, γνωστές και ως βάρη (και biases), τροποποιούνται ανάλογα με τις απώλειες έτσι ώστε η απώλεια να ελαχιστοποιηθεί.

$$W_{new}^{(L)} = W_{old}^{(L)} - \alpha \times \frac{\partial C}{\partial W^{(L)}}$$

$$b_{new}^{(L)} = b_{old}^{(L)} - \alpha \times \frac{\partial C}{\partial b^{(L)}}$$

όπου: α : ρυθμός εκμάθησης (learning rate)

Learning rate: Ο ρυθμός εκμάθησης (learning rate) είναι μια υπερπαραμέτρος που ελέγχει πόσο θα αλλάξει το μοντέλο ως απάντηση στο εκτιμώμενο σφάλμα κάθε φορά που ενημερώνονται τα βάρη του μοντέλου. Η επιλογή της τιμής αποτελεί πρόκληση, καθώς μια πολύ μικρή τιμή, κάνει μικρότερα βήματα στον αλγόριθμο, και μπορεί να οδηγήσει σε μια μακρά διαδικασία εκπαίδευσης που μπορεί να "κολλήσει" (σε κάποιο τοπικό ελάχιστο της συνάρτησης κόστους). Αντίθετα, μια πολύ μεγάλη τιμή μπορεί να οδηγήσει σε πολύ γρήγορη εκμάθηση ενός μη βέλτιστου συνόλου βαρών ή σε μια ασταθή διαδικασία εκπαίδευσης.



Σχήμα 3.3: Στα παραπάνω σχήματα, παρατηρούμε την επίδραση της τιμής του learning rate για την εύρεση ολικού ελαχίστου της συνάρτησης κόστους. Αναλυτικότερα, ξεκινώντας από δεξιά προς τα αριστερά, στο σχήμα α) με μικρό learning rate, το μοντέλο "πρόλαβε" να φτάσει στο ολικό ελάχιστο. Στο σχήμα β) με βέλτιστο learning rate, το μοντέλο φτάνει στο ολικό ελάχιστο. Στο σχήμα γ) και δ) βλέπουμε δυο περιπτώσεις με μεγάλο learning rate, στις οποίες βλέπουμε ασταθή βήματα του αλγορίθμου που όχι μόνο δεν οδηγούν στην ελάχιστη τιμή, αλλά και εκτροχιάζουν την πρόβλεψη μακριά από αυτήν[41]

Παρότι απλή στην κατανόηση και υλοποίηση, επειδή αυτή η μέθοδος υπολογίζει την κλίση (gradient) για ολόκληρο το dataset σε μία ενημέρωση, φυσικά απαιτεί μεγάλη μνήμη και είναι υπολογιστικά δαπανηρή. Επομένως, αξίζει να αναφερθούν παραλλαγές που συχνά χρησιμοποιούνται, για να περιοστούν οι αδυναμίες της μεθόδου

Stochastic Gradient Descent: Η πιο γνωστή παραλλαγή του Gradient Descent, ονομάται στοχαστική (*stochastic*) gradient descent χρησιμοποιείται για να καταπολεμήσει τον αργή ανανέωση των παραμέτρων. Πιο συγκεκριμένα, ενώ στην GD, πρέπει να διατρεχθούν όλα τα δείγματα του συνόλου εκπαίδευσης για να κάνετε μια ενιαία ενημέρωση για μια παράμετρο σε μια συγκεκριμένη επανάληψη, στην SGD, από την άλλη πλευρά, χρησιμοποιείται μόνο ένα ή μόνο ένα υποσύνολο (batch) δείγματος εκπαίδευσης από το σύνολο εκπαίδευσης για να γίνει η ανανέωση για μια παράμετρο σε μια συγκεκριμένη επανάληψη. Εάν χρησιμοποιηθεί υποσύνολο, τότε ο αλγόριθμος ονομάζεται **Minibatch SGD**.

SGD with Momentum: Σε περιπτώσεις που η συνάρτηση κόστους παρουσιάζει τοπικά υψηλή διαφοροποίηση της κλίσης (σαγματικό σημείο, τοπικό ελάχιστο κτλ), ο αλγόριθμος

μειώνει τον ρυθμό ανανέωσης τη τιμές των παραμέτρων. Ιδανικά, ο αλγόριθμος θα πρέπει να διατηρεί την "ορμή" με την οποία ανανέωνε την τιμή του. Για αυτό τον σκοπό, εισάγουμε τον εκθετικά σταθμισμένο μέσο όρο της εκάστοτε παραμέτρου, κάνοντας χρήση μιας νέας υπερπαραμέτρου γ . Η υπερπαραμέτρος αυτή, ονόματι ορμή (*momentum*), αντιπροσωπεύει την σημασία που δίνουμε στην προηγούμενη τιμή της εκάστοτε παραμέτρου, για τον υπολογισμό της νέας της τιμής της. Ο όρος ορμής γ τίθεται συνήθως στο 0,9 ή σε μια παρόμοια τιμή.

$$W_{new}^{(L)} = W_{old}^{(L)} - \alpha \times V_t \quad , \quad V_t = \gamma \times V_{t-1} + (1 - \gamma) \times \frac{\partial C}{\partial W^{(L)}}$$

$$b_{new}^{(L)} = b_{old}^{(L)} - \alpha \times V_t \quad , \quad V_t = \gamma \times V_{t-1} + (1 - \gamma) \times \frac{\partial C}{\partial b^{(L)}}$$

Adagrad: Ένα από τα μειονεκτήματα όλων των βελτιστοποιητών που εξηγήθηκαν είναι ότι ο ρυθμός μάθησης είναι σταθερός για όλες τις παραμέτρους και για κάθε κύκλο. Η διαίσθηση πίσω από το AdaGrad (Adaptive Gradient Descent) είναι ότι μπορούμε να χρησιμοποιήσουμε διαφορετικούς ρυθμούς μάθησης για κάθε νευρώνα για κάθε κρυφό στρώμα με βάση διαφορετικές επαναλήψεις. Σκοπός είναι να κάνει μεγάλες ενημερώσεις για λιγότερο συχνές παραμέτρους και ένα μικρό βήμα για συχνές παραμέτρους. Σε κάθε βήμα, χρησιμοποιούμε το άθροισμα των τετραγώνων των gradients για τον υπολογισμό της νέας τιμής των παραμέτρων:

$$W_{new}^{(L)} = W_{old}^{(L)} - \frac{\alpha}{\sqrt{\theta_t + \epsilon}} \times \frac{\partial C}{\partial W^{(L)}} \quad , \quad \theta_t = \sum_{t=1}^t \left(\frac{\partial C}{\partial W_{t-1}^{(L)}} \right)^2$$

$$b_{new}^{(L)} = b_{old}^{(L)} - \frac{\alpha}{\sqrt{\theta_t + \epsilon}} \times \frac{\partial C}{\partial b^{(L)}} \quad , \quad \theta_t = \sum_{t=1}^t \left(\frac{\partial C}{\partial b_{t-1}^{(L)}} \right)^2$$

Το ϵ είναι ένας όρος εξομάλυνσης που αποφεύγει τη διαίρεση με το μηδέν (συνήθως της τάξης του $1e-8$). Παρατηρείται επίσης, ότι χωρίς την πράξη τετραγωνικής ρίζας, ο αλγόριθμος έχει πολύ χειρότερες επιδόσεις.

Adam Ο βελτιστοποιητής Adam (Adaptive Moment Estimation) είναι μακράν ένας από τους πιο προτιμώμενους βελτιστοποιητές. Η ιδέα πίσω από τον βελτιστοποιητή Adam είναι να χρησιμοποιηθεί η έννοια της ορμής από το "SGD with momentum" και ο προσαρμοστικός ρυθμός μάθησης από το "Adadelta". Έτσι, συνδυάζει τα πλεονεκτήματα και των δύο

μεθόδων. Οι εξισώσεις 3.1 περιγράφουν την μαθηματική διαδικασία του αλγορίθμου για την ανανέωση των βαρών του μοντέλου (ομοίως και για τα biases):

$$W_{new}^{(L)} = W_{old}^{(L)} - \frac{\alpha}{\sqrt{\theta_t + \epsilon}} \times V_t \quad (3.1)$$

$$\theta_t = \sum_{t=1}^t \left(\frac{\partial C}{\partial W_{t-1}^{(L)}} \right)^2, \quad V_t = \gamma \times V_{t-1} + (1 - \gamma) \times \frac{\partial C}{\partial w^{(L)}} \quad (3.2)$$

3.3 Machine Learning Life-cycle

Η τυπική κυκλική διαδικασία που ακολουθείται από τους ειδικούς της μηχανικής μάθησης για την επίλυση προβλημάτων μπορεί να οριστεί ως "ο κύκλος ζωής της μηχανικής μάθησης". Ορίζει κάθε βήμα που ένας οργανισμός πρέπει να ακολουθήσει για να επωφεληθεί από τη μηχανική μάθηση και την τεχνητή νοημοσύνη για να αντλήσει πρακτική επιχειρηματική αξία. Χωρίζεται σε 7 ξεχωριστά στάδια, ως εξής:

1. Ορισμός προβλήματος:

Το πρώτο στάδιο ενός έργου μηχανικής μάθησης είναι ο ορισμός και η κατανόηση του προβλήματος που πρέπει να επιλυθεί μαζί με την επιχειρηματική αξία και την εφαρμογή του. Αυτό φάση ξεκινά με την ανάλυση των στόχων και των λόγων πίσω από ένα συγκεκριμένο πρόβλημα δήλωση. Ο απώτερος στόχος είναι η κατανόηση της δύναμης των δεδομένων και της χρησιμότητά τους στην επίλυση του συγκεκριμένου προβλήματος και την επίτευξη αποτελεσμάτων.

2. Συλλογή δεδομένων:

Αφού καθοριστεί με σαφήνεια ο στόχος, είναι απαραίτητο να αποκτηθούν τα δεδομένα που απαιτούνται από διάφορες διαθέσιμες πηγές δεδομένων. Σε αυτό το στάδιο, ορισμένα από τα ερωτήματα που αξίζει να εξεταστούν είναι:

- Ποια δεδομένα χρειάζονται για το έργο;
- Πού είναι διαθέσιμα αυτά τα δεδομένα;
- Πώς μπορούν να αποκτηθούν;
- Ποιος είναι ο πιο αποτελεσματικός τρόπος αποθήκευσης και πρόσβασης σε όλα αυτά;

Υπάρχουν διάφοροι τρόποι συλλογής δεδομένων για έργα μηχανικής μάθησης, όπως η εστίαση σε ομάδες, οι συνεντεύξεις, έρευνες και δεδομένα εσωτερικής χρήσης και χρηστών. Μια πολύ σημαντική πηγή, όμως είναι και τα δημόσια δεδομένα τα οποία είναι συνήθως δωρεάν (ανοικτά δεδομένα). Αυτά περιλαμβάνουν έρευνες και εμπορικές ενώσεις, όπως τράπεζες, εταιρείες που είναι εισηγμένες στο χρηματιστήριο και άλλες. Παρ' όλα αυτά, εάν τα δεδομένα δεν είναι δημόσια διαθέσιμα, υπάρχουν αρκετοί νομικοί περιορισμοί που αφορούν αυτή τη μέθοδο συλλογής.

3. Διαχείριση δεδομένων:

Ο τύπος και η ποιότητα των δεδομένων που χρησιμοποιούνται σε ένα μοντέλο μηχανικής μάθησης επηρεάζει σημαντικά την έξοδό του. Ως εκ τούτου, η επεξεργασία δεδομένων είναι μια χρήσιμη και χρονοβόρα διαδικασία καθαρισμού και μετατροπής των ακατέργαστων δεδομένων σε χρήσιμη μορφή. Κατά τη διάρκεια αυτής της φάσης οι ειδικοί στα δεδομένα διερευνούν, προ-επεξεργάζονται, εξαρτούν και μετασχηματίζουν δεδομένα πριν από τη μοντελοποίηση και την ανάλυση. Ελλείπουσες τιμές, διπλά δεδομένα, άκυρα δεδομένα, ακατάλληλη μορφοποίηση, θόρυβος είναι μερικά από τα είδη δεδομένων που θα πρέπει να χειριστούν καθ' όλη τη διάρκεια αυτού του σταδίου προετοιμασίας και πριν προχωρήσουν στην εξαγωγή συμπερασμάτων.

4. Διερευνητική Ανάλυση Δεδομένων

Η διερευνητική Ανάλυση Δεδομένων () είναι μια προσέγγιση για την ανάλυση συνόλων δεδομένων με σκοπό τη σύνοψη των κύριων χαρακτηριστικών τους. Η EDA χρησιμοποιείται για να δούμε τι μπορούν να μας πουν τα δεδομένα πριν από την εργασία μοντελοποίησης. Δημιουργούνται πολλαπλές απεικονίσεις που αναδεικνύουν μοτίβα και τάσεις στα δεδομένα, αποκαλύπτοντας έτσι τις σχέσεις/συσχετίσεις μεταξύ των μεταβλητών του προβλήματος. Γενικά, σε περίπτωση μεγάλου όγκου δεδομένων, η δημιουργία γραφημάτων είναι ο καλύτερος τρόπος για να εξερευνήσει και να κοινοποίηση των ευρημάτων, γεγονός που με τη σειρά του επιτρέπει την ταχύτερη και ουσιαστικότερη υλοποίηση των τελευταίων σταδίων του κύκλου ζωής της μηχανικής μάθησης. EDA γενικά οδηγεί σε ταχύτερη λήψη αποφάσεων τόσο κατά την ανάλυση όσο και βάσει των στατιστικών στοιχείων που αποκαλύπτονται για το σύνολο δεδομένων.

5. Ανάπτυξη μοντέλου

Αυτό το στάδιο περιλαμβάνει την επιλογή και την εκπαίδευση του μοντέλου μηχανικής μάθησης (ML). Μας βοηθάει στην ανακάλυψη προτύπων στα δεδομένα και ανάλογα με τη φύση του προβλήματος επιτηρούμενες ή μη επιτηρούμενες εργασίες μάθησης, όπως παλινδρόμηση, ταξινόμηση, πρόβλεψη και ομαδοποίηση αντίστοιχα. Κατά τη διάρκεια αυτής της φάσης, οι μαθηματικές, υπολογιστικές και επιχειρηματικές γνώσεις πρέπει να συνδυαστούν για την εκπαίδευση ενός αλγορίθμου ML που θα κάνει προβλέψεις ή εκτιμήσεις με βάση τα παρεχόμενα δεδομένα. Σε όλη τη διάρκεια του της διαδικασίας μοντελοποίησης ML, οι κατάλληλοι αλγόριθμοι μηχανικής μάθησης ή βαθιάς μάθησης (deep learning) πρέπει να επιλέγονται με βάση πολλαπλά κριτήρια (ταύτιση με τη φύση του συγκεκριμένου συνόλου δεδομένων, τους χρόνους εκπαίδευσης, την ανάγκη για υψηλή ακρίβεια, την ανάγκη για ερμηνευσιμότητα κλπ) Τέλος, συχνά οι αλγόριθμοι ML βοηθούν στον εντοπισμό βασικών χαρακτηριστικά με υψηλή προγνωστική αξία που οδηγούν στην καλύτερη κατανόηση του προβλήματος και των μεταβλητών του.

6. Αξιολόγηση μοντέλου

Είναι ένα κρίσιμο βήμα που θα καθορίσει την ποιότητα και την ακρίβεια των μελλοντικών προβλέψεων σε νέες καταστάσεις. Τα μοντέλα ML πρέπει να αξιολογούνται σύμφωνα με μετρικές αξιολόγησης (π.χ. ακρίβεια, ανάκληση, F1-score, για ταξινόμηση, μέσο τετραγωνικό σφάλμα, μέσο απόλυτο σφάλμα για παλινδρόμηση, διαφορές μετρικές ομοιότητας για ομαδοποίηση κ.λπ.) σε προηγουμένως αθέατα υποσύνολα του συνόλου δεδομένων (στην περίπτωση των εποπτευόμενης μάθησης). Προαιρετικά, η ρύθμιση των υπερπαραμέτρων είναι εξαιρετικά χρήσιμη, αλλά συχνά επίσης υπολογιστικά εντατική, προκειμένου να ανακαλυφθεί η καλύτερη υπερπαραμέτρος συνδυασμών που οδηγούν στα καλύτερα αποτελέσματα αξιολόγησης για μια επιλεγμένη οικογένεια μοντέλων. Η διασταυρωμένη επικύρωση K-folds [42] είναι μία από τις πιο δημοφιλείς μεθόδους για την αξιοποίηση κατά τη διάρκεια αυτής της διαδικασίας.

7. Αξιοποίηση μοντέλου

Το συγκεκριμένο βήμα δεν αναπτύσσεται στα πλαίσια της διπλωματικής και επομένως αναφέρεται επιγραμματικά για λόγους πληρότητας. Το τελικό βήμα, αυτό, περιλαμβάνει την εφαρμογή (deployment) ενός μοντέλου μηχανικής μάθησης σε ένα περιβάλλον πραγματικού κόσμου, ώστε να μπορεί να λαμβάνει αυτόματα αποφάσεις βάσει δεδομένων. Πριν από την ανάπτυξη ενός μοντέλου, είναι ζωτικής σημασίας να δοκιμαστεί και να αξιολογηθεί η ευρωστία, η συμβατότητα και η επεκτασιμότητά του.

3.4 Hyperparameter Tuning

3.4.1 Εισαγωγή

Στη μηχανική μάθηση, πρέπει να γίνει διάκριση μεταξύ παραμέτρων και υπερπαραμέτρων. Ένας αλγόριθμος μάθησης μαθαίνει ή εκτιμά τις παραμέτρους (όπως τα βάρη και biases) του μοντέλου για το δεδομένο σύνολο δεδομένων και στη συνέχεια συνεχίζει να ενημερώνει αυτές τις τιμές καθώς συνεχίζει να μαθαίνει. Αντίθετα, οι υπερπαραμέτροι είναι συγκεκριμένες για τον *ίδιο* τον αλγόριθμο, οπότε δεν μπορούν να υπολογιστούν οι τιμές τους από τα δεδομένα. Χρησιμοποιούμε τις υπερπαραμέτρους για να υπολογίσουμε τις παραμέτρους του μοντέλου, με διαφορετικές τιμές υπερπαραμέτρων να παράγουν διαφορετικές τιμές παραμέτρων του μοντέλου για ένα δεδομένο σύνολο δεδομένων. Χαρακτηριστικές υπερπαραμέτροι ενός μοντέλου είναι ο αριθμός κρυφών στρωμάτων, αριθμός κόμβων/νευρώνων ανά στρώμα, ο ρυθμός μάθησης (learning rate), ορμή (momentum), optimizer (Adam,SGD,RMSProp), συνάρτηση ενεργοποίησης (ReLU,SiLU etc) και η συνάρτηση απωλειών (MSE,MAE,MAPE etc)

Ο συντονισμός υπερπαραμέτρων (hyperparameter tuning) είναι η διαδικασία προσδιορισμού του σωστού συνδυασμού υπερπαραμέτρων που μεγιστοποιεί την απόδοση του μοντέλου. Λειτουργεί με την εκτέλεση πολλαπλών δοκιμών (trials) σε μια ενιαία διαδικασία εκπαίδευσης. Κάθε δοκιμή είναι μια πλήρης εκτέλεση της εφαρμογής εκπαίδευσής με τιμές για τις επιλεγμένες υπερπαραμέτρους, οι οποίες καθορίζονται εντός των ορίων που προκαθορίζονται. Αυτή η διαδικασία μόλις ολοκληρωθεί θα σας δώσει το σύνολο των τιμών των υπερπαραμέτρων που είναι οι καταλληλότερες για το μοντέλο ώστε να δώσει βέλτιστα αποτελέσματα. Η διαδικασία αυτή είναι εξαιρετικά σημαντική για την βέλτιστη απόδοση του μοντέλου [43] [44]

3.4.2 Tuning Methods

Μπορούμε να βρούμε αυτές τις βέλτιστες τιμές υπερπαραμέτρων χρησιμοποιώντας χειροκίνητες (manual) ή αυτοματοποιημένες (automated) μεθόδους. Όταν συντονίζουμε τις υπερπαραμέτρους χειροκίνητα, συνήθως ξεκινάμε χρησιμοποιώντας τις προεπιλεγμένες συνιστώμενες τιμές και στη συνέχεια αναζητούμε ένα εύρος τιμών πειραματικά. Αλλά ο χειροκίνητος συντονισμός είναι μια κουραστική, χρονοβόρα προσέγγιση, ειδικά όταν υπάρχουν πολλές υπερπαραμέτρους με μεγάλο εύρος τιμών. Οι αυτοματοποιημένες μέθοδοι συντονισμού

υπερπαραμέτρων χρησιμοποιούν έναν αλγόριθμο για την αναζήτηση των βέλτιστων τιμών, με τις πιο δημοφιλείς να είναι η αναζήτηση πλέγματος (grid search), η τυχαία αναζήτηση (random search) και η βελτιστοποίηση κατά Bayes (bayesian search).

Grid Search: Αποτελεί ένα είδος brute force αναζήτησης. Δημιουργούμε ένα πλέγμα πιθανών διακριτών τιμών υπερπαραμέτρων και στη συνέχεια προσαρμόζουμε το μοντέλο με κάθε πιθανό συνδυασμό. Καταγράφουμε τις επιδόσεις του μοντέλου για κάθε σύνολο και στη συνέχεια επιλέγουμε τον συνδυασμό που παρήγαγε τις καλύτερες επιδόσεις.

Random Search: Στη μέθοδο τυχαίας αναζήτησης, δημιουργούμε ένα πλέγμα πιθανών τιμών για τις υπερπαραμέτρους. Κάθε επανάληψη δοκιμάζει έναν τυχαίο συνδυασμό υπερπαραμέτρων από αυτό το πλέγμα, καταγράφει την απόδοση και τέλος επιστρέφει τον συνδυασμό υπερπαραμέτρων που παρείχε την καλύτερη απόδοση.

Bayesian Search: Ο συντονισμός και η εύρεση των σωστών υπερπαραμέτρων για το μοντέλο σας είναι ένα πρόβλημα βελτιστοποίησης. Θέλουμε να ελαχιστοποιήσουμε τη συνάρτηση απώλειας του μοντέλου μας αλλάζοντας τις παραμέτρους του μοντέλου. Η μέθοδος αυτή μας βοηθά να βρούμε το ελάχιστο σημείο στον ελάχιστο αριθμό βημάτων. Χρησιμοποιείται επίσης μια συνάρτηση απόκτησης που κατευθύνει τη δειγματοληψία σε περιοχές όπου είναι πιθανή η βελτίωση σε σχέση με την τρέχουσα καλύτερη παρατήρηση. Αν και η καλύτερη εναλλακτική για αναζήτηση βέλτιστων τιμών των παραμέτρων, δεν είναι αλάθητη. Υπάρχει κίνδυνος στις bayesian search τεχνικές, σε μεγάλο εύρος τιμών, να παγιδευτεί σε τοπικό ελάχιστο, μην οδηγώντας σε βέλτιστη λύση [45]. Έτσι, ενώ είναι σημαντικό να μελετηθούν οι υπερπαραμέτροι σε μεγάλο μέρος τιμών, πρέπει να γίνει με προσεκτική επιλογή των άκρων ώστε να επιτευχθούν τα βέλτιστα αποτελέσματα.

Κεφάλαιο 4

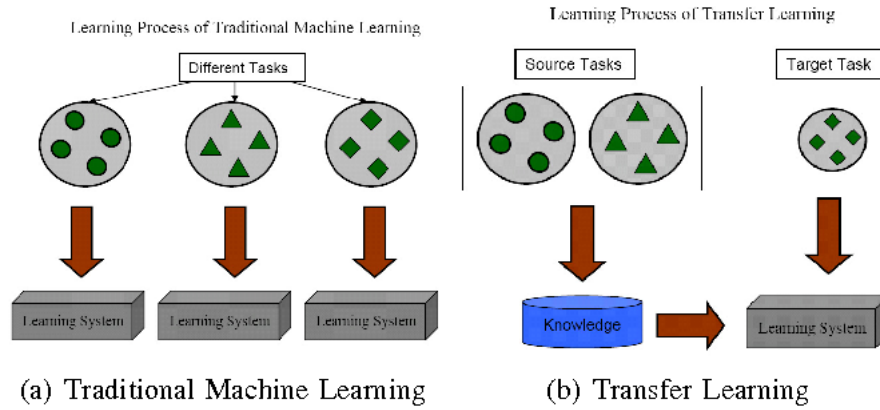
Transfer Learning

4.1 Εισαγωγή

Η μεταφορά μάθησης (transfer learning) είναι μια τεχνική μηχανικής μάθησης όπου ένα μοντέλο που έχει εκπαιδευτεί σε μια εργασία επαναχρησιμοποιείται σε μια δεύτερη συναφή εργασία. Εάν τα δύο μοντέλα έχουν αναπτυχθεί για να εκτελούν παρόμοιες εργασίες, τότε η γενικευμένη γνώση μπορεί να μοιραστεί μεταξύ τους. Αυτή η προσέγγιση στην ανάπτυξη της μηχανικής μάθησης μειώνει τους πόρους και την ποσότητα των επισημειωμένων δεδομένων που απαιτούνται για την εκπαίδευση νέων μοντέλων.

Τα περισσότερα μοντέλα που επιλύουν πολύπλοκα προβλήματα χρειάζονται *πάρα* πολλά δεδομένα, και η συγκέντρωση τεράστιων ποσοτήτων επισημασμένων (labeled) δεδομένων για μοντέλα με επίβλεψη μπορεί να είναι πραγματικά δύσκολη, ειδικά αν αναλογιστεί κανείς το χρόνο και την προσπάθεια που απαιτείται για την επισήμανση των σημείων δεδομένων.

Έτσι, η μεταφορά μάθησης γίνεται σημαντικό μέρος της εξέλιξης της μηχανικής μάθησης και χρησιμοποιείται όλο και περισσότερο ως τεχνική στο πλαίσιο της διαδικασίας ανάπτυξης.



Σχήμα 4.1: Σύγκριση μηχανικής μάθησης με και χωρίς την χρήση τεχνικών transfer learning [46]

4.2 Βασικές Έννοιες & Ορισμοί

Για την καλύτερη κατανόηση των εννοιών, θα δανειστούν ορισμοί από την εργασία των Pan και Yang "A Survey on Transfer Learning" [46]. Ορίζεται ένας domain D από δύο στοιχεία: ένα χώρο χαρακτηριστικών (feature space) X και μια οριακή κατανομή πιθανοτήτων $Pr\{X\}$.

Δεδομένου ενός domain, ορίζεται μια εργασία (task) T . Αυτή αποτελείται από δύο στοιχεία: έναν χώρο ετικετών (label space) Y και μια objective function $f(\cdot)$, που ταυτίζεται με την δεσμευμένη πιθανότητα του label space Y προς τον feature space X ($Pr\{Y|X\}$). Η objective function αποτελείται από ζεύγη $\{x_i, y_i\} \in \{X, Y\}$ και χρησιμοποιείται για την πρόβλεψη του αντίστοιχου label $f(x)$ μιας νέας περίπτωσης x .

$$D = \{X, Pr\{X\}\} \quad , \quad T = \{Y, f(\cdot)\} = \{Y, Pr\{Y|X\}\}$$

Με βάση τον ορισμό, δεδομένων των source domain D_S με αντίστοιχο task T_S , και target domain (D_T) με task T_T , ο σκοπός του transfer learning είναι να ανακαλυφθεί το $Pr\{Y_T|X_T\}$ του D_T , με πληροφορίες αποκτημένες από τα D_S και D_T όπου $D_S \neq D_T$ ή $T_S \neq T_T$. Όταν δύο domains (έστω D_S και D_T) είναι διαφορετικά, τότε είτε έχουν διαφορετικά feature spaces ($X_S \neq X_T$) ή οριακές κατανομές ($Pr\{X_S\} \neq Pr\{X_T\}$). Αντίστοιχα, δύο tasks (έστω T_S και T_T) είναι διαφορετικά, τότε είτε έχουν διαφορετικά label spaces ($Y_S \neq Y_T$), είτε διαφορετικές δεσμευμένες κατανομές ($Pr\{Y_S|X_S\} \neq Pr\{Y_T|X_T\}$).

4.3 Κατηγοριοποίηση ειδών

4.3.1 Κριτήριο: Domain/Task elements

Καθώς το domain D και task T είναι πλειάδες (tuples), αυτές οι ανισότητες οδηγούν σε τέσσερα σενάρια μάθησης μεταφοράς.

1. $X_S \neq X_T$: Υπάρχει διαφορά στα feature spaces του source και target. Ένα απλό παράδειγμα είναι η περίπτωση στην οποία για την πρόβλεψη της τιμής πώλησης ενός ρούχου (έστω κοινό το task -χωρίς να καθίσταται απαραίτητο-), χρησιμοποιούμε διαφορετικά χαρακτηριστικά του εν λόγω ρούχου. Αυτή η περίπτωση, ονομάζεται και ετερογενής (*heterogeneous*) transfer learning, ενώ η ισότητα (ίδια χαρακτηριστικά για το ρούχο) προφανώς ονομάζεται *homogeneous*.
2. $Pr\{X_S\} \neq Pr\{X_T\}$: Οι οριακές κατανομές πιθανότητας στο source και target διαφέρουν. Συνεχίζοντας το παράδειγμα του ρούχου, το ένα domain αφορά φθηνές, ενώ το άλλο ακριβές μάρκες ρούχων, αναμένουμε προφανώς πολύ διαφορετικές τιμές. Αυτή η περίπτωση επίσης ονομάζεται *domain adaptation*
3. $Y_S \neq Y_T$: Τα labels spaces μεταξύ δύο tasks διαφέρουν. Ένα ξεκάθαρο παράδειγμα είναι να γίνει μεταφορά μάθησης από ένα πρόβλημα regression, όπου τα labels έχουν εύρος τιμών, σε ένα πρόβλημα classification (διακριτές τιμές των labels)
4. $Pr\{Y_S|X_S\} \neq Pr\{Y_T|X_T\}$: οι δεσμευμένες πιθανότητες των source/target tasks είναι διαφορετικές (ένα feature έχει διαφορετικό "νόημα" ανάμεσα στα tasks). Για παράδειγμα, αν η τιμή αγοράς ενός αυτοκινήτου θεωρείται "φτηνή", πιθανόν η ίδια τιμή δεν θεωρείται τόσο "φτηνή" όταν εξετάζεται η αγορά μιας σοκολάτας. Αυτό το είδος bias ονομάζεται *context feature bias*

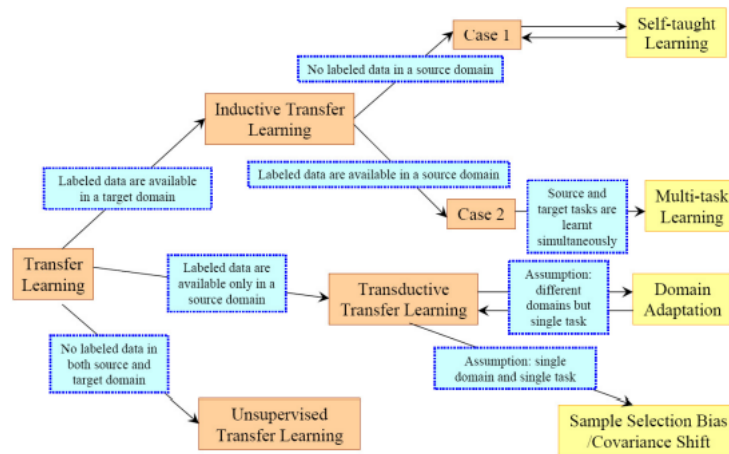
4.3.2 Κριτήριο: Source/Target Domain & Tasks

Όπως με τα στοιχεία που απαρτίζονται τα domains και tasks, είναι δυνατή και η κατηγοριοποίηση των τεχνικών transfer learning με βάση τα source και target domains/tasks. Αναλυτικότερα:

1. **Inductive Transfer learning:** Τα source και target domains είναι ίδια, αλλά τα source και target tasks είναι διαφορετικά αλλά σχετίζονται μεταξύ τους. Ανάλογα με το αν το

source domain περιέχει labeled δεδομένα ή όχι, αυτό μπορεί να χωριστεί περαιτέρω σε δύο υποκατηγορίες: *multi-task learning* και *self-taught learning* αντίστοιχα.

2. **Unsupervised Transfer Learning:** Τα source και target domains είναι γειτονικά, αλλά τα tasks είναι διαφορετικά αλλά σχετίζονται μεταξύ τους, ενώ επίσης, δεν υπάρχουν διαθέσιμα labeled data σε κανένα από τα domains.
3. **Transductive Transfer Learning:** Τα source και target tasks είναι τα ίδια, ενώ τα source και target domains είναι διαφορετικά αλλά σχετικά μεταξύ τους. Ανάλογα με το αν τα feature spaces ή τα marginal probabilities των δύο domains είναι ίδια, χωρίζεται σε *domain adaptation* και *sample selection bias* ή *covariate shift* αντίστοιχα



Σχήμα 4.2: Επισκόπηση των διαφόρων μεθόδων transfer learning [46]

4.4 Deep transfer learning

Τα συστήματα βαθιάς μάθησης (deep learning) είναι πολυεπίπεδες αρχιτεκτονικές που μαθαίνουν διαφορετικά χαρακτηριστικά σε διαφορετικά επίπεδα. Τα αρχικά στρώματα (feature extraction layers) συγκεντρώνουν χαρακτηριστικά υψηλότερου (γενικού) επιπέδου που περιορίζονται σε λεπτά (ειδικά) χαρακτηριστικά καθώς προχωράμε βαθύτερα στο δίκτυο (classifier layers). Η διασθητική αυτή περιγραφή είναι σημαντική για την κατανόηση της υπόλοιπης ενότητας.

Deep transfer learning (βαθιά μεταφορά μάθησης) ορίζεται μια εφαρμογή του transfer learning, πάνω στα νευρωνικά δίκτυα (deep learning), με όλο και περισσότερη αξιοποίηση σε

κλάδους όπως η όραση υπολογιστών (computer vision) και η επεξεργασία φυσικής γλώσσας (MLP) [47].

Η λογική πίσω από αυτό είναι ότι σε κοινές εργασίες (κοινά δεδομένα ή σκοπός), με την εκπαίδευση ενός μοντέλου για μια πηγαία (source) εργασία, έχει αποκτηθεί κάποια ποσότητα πληροφορίας και για την εργασία στόχου (task). Η γνώση αυτή μεταφράζεται σε weights και biases (παραμέτροι) του μοντέλου, κι έτσι μπορούν να αξιοποιηθούν για την εκπαίδευση του άλλου. Βασικές στρατηγικές που αναγράφονται στην βιβλιογραφία [48, 49] είναι οι εξής:

4.4.1 Warm Start

Η μέθοδος της ευνοϊκής αρχής (warm start) είναι ίσως η σημαντικότερη και λειτουργεί ως βάση για αυτές που ακολουθούν. Η βασική ιδέα εδώ είναι να αξιοποιηθούν τα βάρη του προ-εκπαιδευμένου μοντέλου μιας άλλης εργασίας, ως αρχικές τιμές των παραμέτρων για το μοντέλο της ζητούμενης εργασίας. Αξίζει να αναφερθεί, ότι με την αποφυγή στοχευμένης αρχικοποίησης των παραμέτρων, δίνει μια ευνοϊκότερη πορεία για την εκπαίδευση του μοντέλου, και μειώνονται οι πιθανότητες για σύγκλιση σε σαγματικά σημεία ή εκτροχιασμό της πρόβλεψης (ακατάλληλο learning rate)

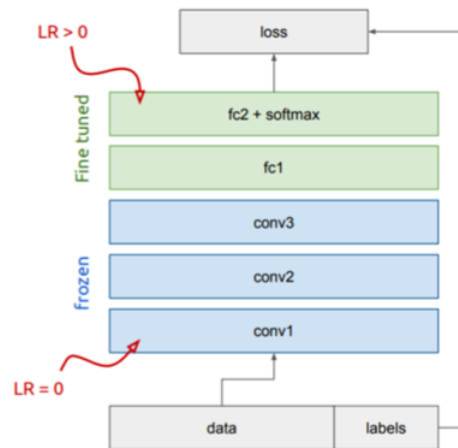
4.4.2 Bounded Epochs

Η ανάγκη για παραγωγή ποιοτικών προβλέψεων σε σύντομο χρονικό διάστημα, οδήγησε στην δημιουργία της μεθόδου των φραγμένων εποχών. Η μέθοδος αυτή, αξιοποιεί την τεχνική του *warm start*, αλλά η εκπαίδευση του μοντέλου στο ζητούμενο task γίνεται σε περιορισμένο αριθμό εποχών. Η λογική είναι ότι, μετά το *warm start*, χρειάζονται ελάχιστες διαπεράσεις των δεδομένων από το μοντέλο, για να επιτευχθεί η μέγιστη ακρίβεια του μοντέλου, δεδομένης της ελάχιστης επανεκπαίδευσης του

4.4.3 Freezing

Στην τεχνική αυτή, επίσης χρησιμοποιείται η τεχνική του *Warm-Start*, με τα βάρη του πηγαίου μοντέλου να λειτουργούν σαν αρχικοποίηση των βαρών του μοντέλου σκοπού. Στην συνέχεια όμως, ενώ συνεχίζεται η επανεκπαίδευση (fine-tuning) του μοντέλου, κάποια από τα αρχικά layers του μοντέλου "παγώνουν": δεν ανανεώνονται οι παράμετροι τους κατά την εκπαίδευση του target μοντέλου. Συνήθως τα επίπεδα που επιλέγονται είναι τα αρχικά στα

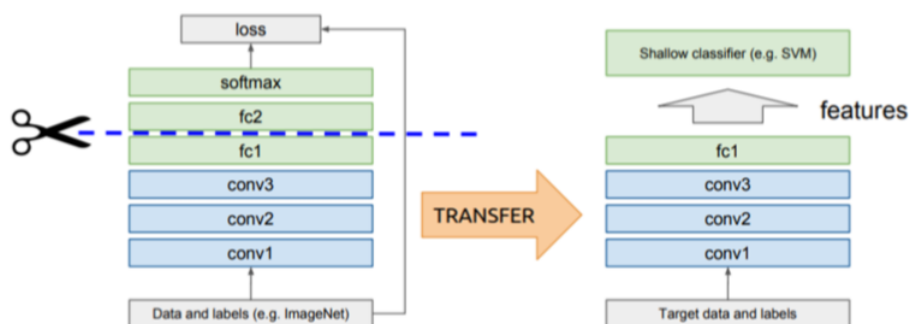
οποία γίνεται η εξαγωγή γενικών πληροφοριών, με την λογική ότι οι γενικές αυτές πληροφορίες είναι κοινές και για το source και για το target task. Έτσι, επιτυγχάνουμε επιτάχυνση της εκπαίδευση του μοντέλου μειώνοντας σε μεγάλο ποσοστό τις υπολογιστικές του ανάγκες.



Σχήμα 4.3: Απεικόνιση ενός μοντέλου στο οποίο γίνεται χρήση freezing με fine-tuning. Στο σχήμα φαίνεται ότι τα επίπεδα που έχουν "παγώσει" έχουν μηδενικό learning rate (LR), ενώ σε αυτά που συνεχίζεται η εκπαίδευση (fine-tuning) υπάρχει ένα θετικό (και σαφώς μικρό) learning rate [48]

4.4.4 Head Replacement

Η μέθοδος αυτή, όπως και με τις προηγούμενες, αξιοποιεί τα αρχικά επίπεδα ενός μοντέλου εκπαιδευμένου σε ένα γειτονικό task. Η ειδοποιός όμως διαφορά της είναι ότι εισάγει την έννοια της αντικατάστασης της κεφαλής του μοντέλου, με μια κεφαλή (classifier) πάνω στην target task. Επομένως



Σχήμα 4.4: Απεικόνιση δύο μοντέλων που επιτυγχάνεται transfer learning με την μέθοδο της "μεταφοράς κεφαλής" [48]

Μέρος II

Λογισμικό & Πείραμα

Κεφάλαιο 5

Υλοποίηση ML Pipeline

5.1 Επιλογή τεχνολογιών

Σκοπός είναι η υλοποίηση των σταδίων του ML lifecycle σε ένα ενιαίο framework όπου θα εκτελούνται ακολουθιακά. Αυτή η λειτουργία επιτεύχθηκε με την χρήση του *MLFlow* [50] : μιας open-source πλατφόρμας για τη διαχείριση του κύκλου ζωής της μηχανικής μάθησης. Αντιμετωπίζει τέσσερις πρωταρχικές λειτουργίες:

1. Παρακολούθηση πειραμάτων για την καταγραφή και τη σύγκριση παραμέτρων και αποτελεσμάτων (MLflow Tracking).
2. Συσκευασία κώδικα ML σε επαναχρησιμοποιήσιμη, αναπαραγώγιμη μορφή, προκειμένου να μοιραστεί με άλλους επιστήμονες δεδομένων ή να μεταφερθεί στην παραγωγή (MLflow Projects).
3. Διαχείριση και ανάπτυξη μοντέλων από διάφορες βιβλιοθήκες ML σε διάφορες πλατφόρμες εξυπηρέτησης μοντέλων και εξαγωγής συμπερασμάτων (MLflow Models).
4. Παροχή ενός κεντρικού αποθηκευτικού χώρου μοντέλων για τη συνεργατική διαχείριση του πλήρους κύκλου ζωής ενός μοντέλου MLflow, συμπεριλαμβανομένης της έκδοσης του μοντέλου, των μεταβάσεων σταδίων και των σχολίων (MLflow Model Registry).

Λόγω της ανεξαρτησίας του *MLFlow*, μπορεί να χρησιμοποιηθεί με οποιαδήποτε βιβλιοθήκη μηχανικής μάθησης και σε οποιαδήποτε γλώσσα προγραμματισμού, καθώς όλες οι λειτουργίες είναι προσβάσιμες μέσω ενός REST API και ενός CLI, ενώ περιλαμβάνει επίσης ένα Python API, R API και Java API

5.1.1 Διαχείριση δεδομένων

Το στάδιο της προ-επεξεργασίας των δεδομένων έγινε χρήση των βιβλιοθηκών *Darts* [51] και *Pandas* [52] για την αποθήκευση και διαχείριση των datasets. Η βιβλιοθήκη *Darts* είναι μια πολύ γνωστή, φιλική προς τον χρήστη βιβλιοθήκη, που αξιοποιείται για προβλέψεις και ανίχνευση ανωμαλιών σε χρονοσειρές: χαρακτηριστικά που την καθιστούν ιδανική για τον σκοπό του βήματος.

5.1.2 Διερευνητική Ανάλυση Δεδομένων

Για την ανάλυση των δεδομένων (EDA) έγινε χρήση ενός συνδυασμού των βιβλιοθηκών *Matplotlib* [53], *Seaborn* [54] και *Plotly* [55]. Η *Matplotlib* είναι μια εξαιρετική και απλή στην χρήση βιβλιοθήκη για απεικόνιση δεδομένων. Παρ' όλα αυτά, απλότητα της αποτελεί εμπόδιο στο να χρησιμοποιηθεί για πολύπλοκα γραφήματα λόγω του low-level interface της.

Έτσι, για την αντιμετώπιση αυτής της αδυναμίας, χρησιμοποιήθηκε η *Seaborn*: μια βιβλιοθήκη οπτικοποίησης δεδομένων *Python* που βασίζεται στην *Matplotlib*, αλλά με high-level interface. Αποτελεί επίσης μια πολύ διάσημη βιβλιοθήκη, με πολλούς χρήστες να την χρησιμοποιούν. Αυτός ήταν ένας σημαντικός παράγοντας για την επιλογή της καθώς βοήθησε στην επιτάχυνση της μάθησης της και υλοποίησης του κατάλληλου κώδικα για τις ανάγκες της διπλωματικής.

Η συνάρτηση *Plotly* είχε περιορισμένη χρήση στην δημιουργία των διαδραστικών γραφημάτων του EDA, στα οποία στη *Seaborn* υπήρχε περίπλοκη κατασκευή και θα έκανε τον κώδικα πιο δυσανάγνωστο. Αξίζει επίσης να αναφερθεί η χρήση της βιβλιοθήκης *Scipy*: μια βιβλιοθήκη που περιλαμβάνει θεμελιώδεις αλγόριθμοι για επιστημονικούς υπολογισμούς σε *Python* [56]. Αξιοποιήθηκε για την ομαδοποίηση των δεδομένων των χωρών, απαραίτητη για την μελέτη της απόδοσης του transfer learning σε ορισμένα test cases της διπλωματικής

5.1.3 Ανάπτυξη & αξιολόγηση μοντέλου

Ανάπτυξη μοντέλων: Για την υλοποίηση του μοντέλου, είναι υψίστης σημασίας η επιλογή των κατάλληλων τεχνολογιών (frameworks) για την οργάνωση των δεδομένων, την εκπαίδευση και την πρόβλεψη δεδομένων για το μοντέλο. Λόγω της διασημότητας και του εύρους του κλάδου της βαθιάς μάθησης, υπάρχει μεγάλη ποικιλία frameworks, ανάλογα με τις απαιτήσεις της εκάστοτε εργασίας. Ενώ οι απόψεις δίστανται στην επιλογή του καλύτερου, τα

πιο διαδεδομένα φαίνεται να είναι τρία: *Keras* [57], *Tensorflow* [58] και *Pytorch* [59]

Ενώ οποιοδήποτε framework θα πρόσφερε ικανοποιητικά αποτελέσματα [60], έπειτα από εκτεταμένη έρευνα επιλέχτηκε η χρήση του *Pytorch* για τους σκοπούς της διπλωματικής. Η ταχύτητα εκπαίδευσης αποτελεί ένα σημαντικό παράγοντα στην επιλογή, με το *Pytorch* να είναι η καλύτερη επιλογή εκ των τριών [61], [62]. Ακόμα, η συγκριτικά εύκολη χρήση του *Pytorch* είναι καθοριστική για την επιλογή framework όταν απαιτούνται σημαντικές προσαρμογές στην αρχιτεκτονική των μοντέλων. [62]. Τέλος, η ικανότητα του *Pytorch* να επαναλαμβάνει την ίδια ταξινόμηση και τμηματοποίηση μοντέλων, είναι ένα σημαντικό χαρακτηριστικό το οποίο σε εκδόσεις του *Keras* και του *TensorFlow* δεν επιτυγχάνεται λόγω ορισμένων ανεξέλεγκτων τυποποιήσεων [63].

Για την εκπαίδευση του μοντέλου χρησιμοποιήθηκε το *Lightning* [64]: ένα από τα πιο γνωστά wrappers του *Pytorch*. Κύριο χαρακτηριστικό του αποτελεί η τυποποίηση βασικών κοινών λειτουργιών όπως τα στάδια της εκπαίδευσης (forward/backward pass) και προεπεξεργασίας των δεδομένων. Σε συνδυασμό με την οργάνωση των λειτουργιών σε επώνυμες συναρτήσεις, ο κώδικας συχνά είναι λιγότερο επιρρεπής σε προγραμματιστικά λάθη, και αισθητά πιο ευανάγνωστος [65]

Η μέθοδος ensemble πραγματοποιήθηκε με την χρήση του *Ensemble-PyTorch* [66]. Το *Ensemble-PyTorch* είναι ένα ενοποιημένο framework για ensemble στην *PyTorch*, με ευχρηστό API για την εκπαίδευση/αξιολόγηση του ensemble, καθώς και υψηλή αποδοτικότητα στην εκπαίδευση με παραλληλισμό. Εναλλακτική μέθοδος ensemble μπορούσε να πραγματοποιηθεί με την χρήση της βιβλιοθήκης *Scikit-learn* [67]. Παρ' όλα αυτά, έχοντας ήδη κάνει χρήση της βιβλιοθήκης *PyTorch* για το μοντέλο πρόβλεψης, χρησιμοποιήθηκε η πρώτη επιλογή για λόγους συμβατότητας.

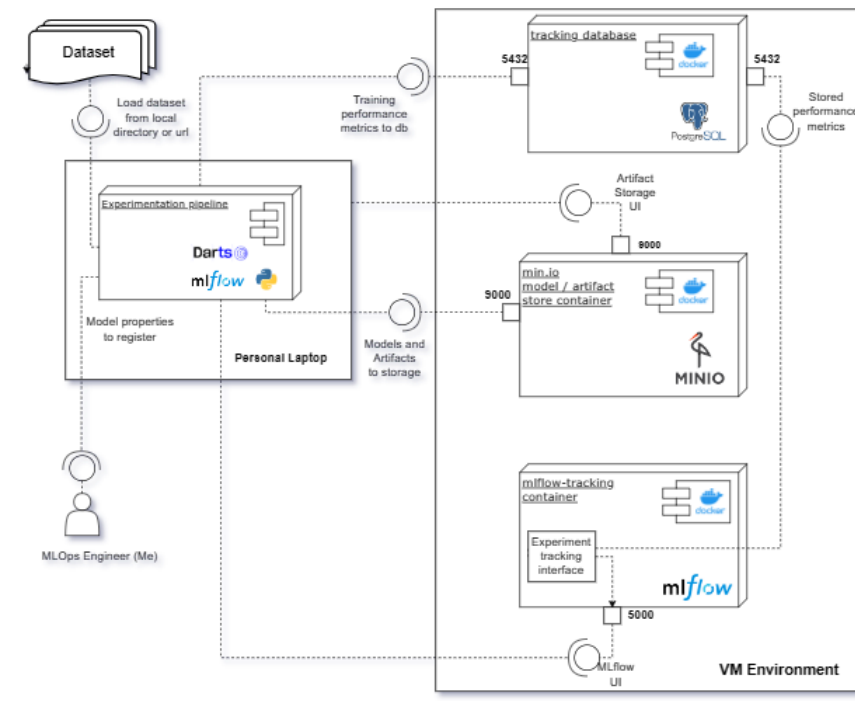
Εύρεση βέλτιστων παραμέτρων: Όπως έχει αναφερθεί, ο συντονισμός των υπερπαραμέτρων (hyper-parameter tuning) αποτελεί βασικό μέρος του ελέγχου της συμπεριφοράς ενός μοντέλου μηχανικής μάθησης. Εάν δεν συντονιστούν σωστά οι υπερπαραμέτροι του μοντέλου, οι εκτιμώμενες παράμετροι παράγουν μη βέλιστα αποτελέσματα, καθώς δεν ελαχιστοποιούν τη συνάρτηση απώλειας. Επομένως, είναι σημαντικό να επιλεχτεί κι η κατάλληλη βιβλιοθήκη για αυτό τον στόχο, που να συνδυάζει βέλτιστη ευχρηστία και απόδοση. Έτσι, εξετάζοντας συγκρίσεις που έγιναν μεταξύ των βιβλιοθηκών [68, 69] επιλέχτηκε σαν βιβλιοθήκη η *Optuna* [70] λόγω της ευχρηστίας του API, της αναλυτικής βιβλιογραφίας, της ταχύτητας και της ποικιλίας απεικονίσεων των δεδομένων.

Επαλήθευση μοντέλου: Βιβλιοθήκες όπως Darts [51] και Pandas [52] αποδείχτηκαν εξαιρετικά χρήσιμες προσφέροντας συναρτήσεις μετρικών (MAPE, RMSE, MASE etc) και δόμησης των δεδομένων (dataframes, timeseries etc)

5.2 Περιβάλλον & Αρχιτεκτονική Pipeline

Στο σχήμα 5.1 παρουσιάζεται το MLflow service που μας παραχωρήθηκε από το πανεπιστήμιο για την καταγραφή των πειραμάτων μας. Περιλαμβάνει έναν:

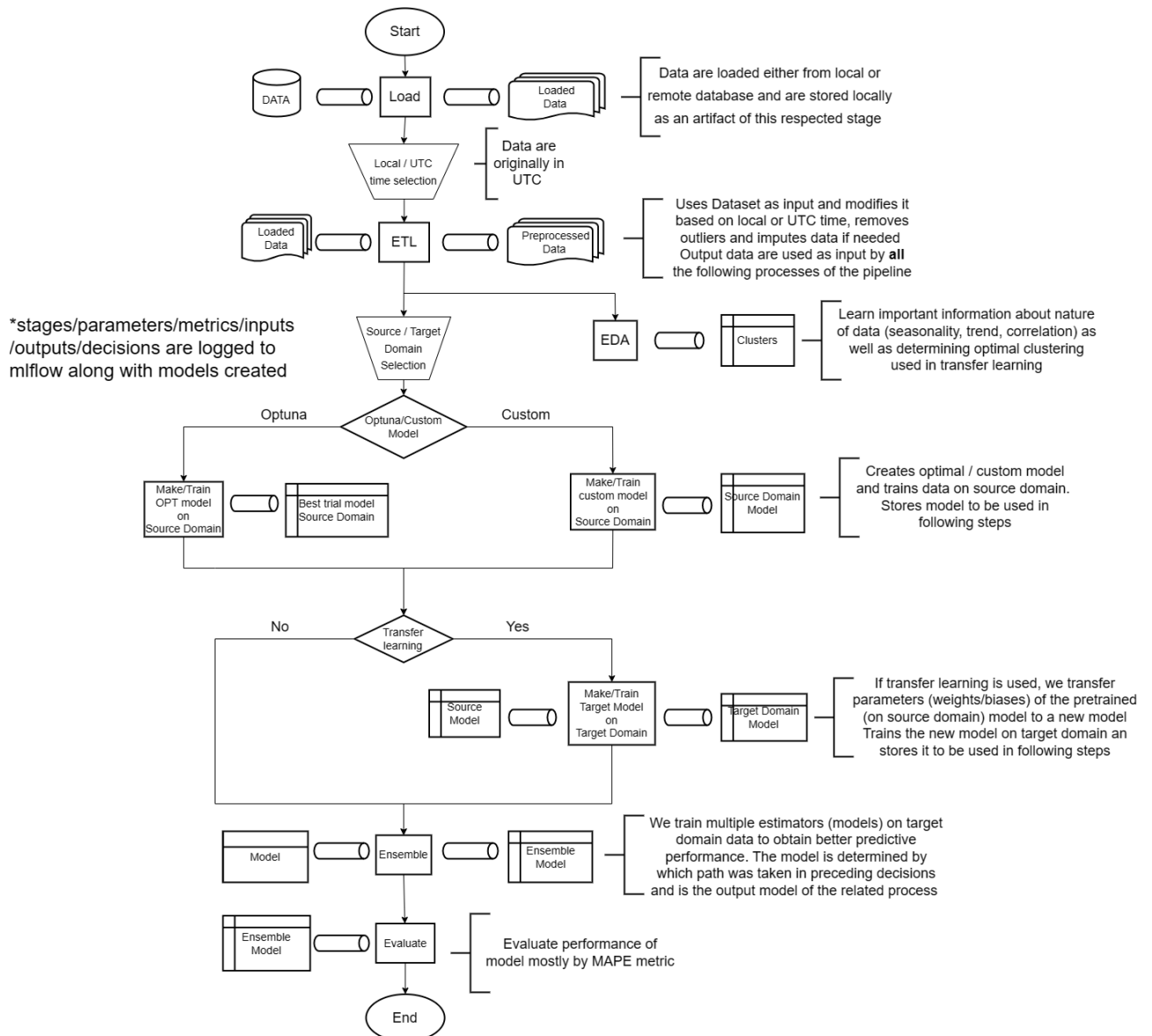
1. **MLflow client:** docker [71] υπεύθυνο για την χρήση και καταγραφή (logging) των δεδομένων κάνοντας χρήση της πλατφόρμας mlflow
2. **MinIO client:** για την αποθήκευση των μετρικών, artifacts του mlflow σε *MinIO* [72]
3. **PostgreSQL client:** για την δημιουργία ενός interface μεταξύ της βάσης δεδομένων και του UI (docker)



Σχήμα 5.1: Αρχιτεκτονικό διάγραμμα της τεχνολογίας που σχεδιάστηκε (deployment)

Στο σχήμα 5.2 παρουσιάζεται λεπτομερώς η αρχιτεκτονική του transfer learning pipeline που υλοποιήθηκε σε αυτή την εργασία. Η λειτουργία των σταδίων αναλύεται διεξοδικά στα κεφάλαια 5 και 6, αλλά στο παρόν κεφάλαιο γίνεται μια επιγραμματική περιγραφή

1. **Main:** Ανεξάρτητο στάδιο των υπολοίπων που λειτουργεί σαν δρομολογητής (driver) των υπόλοιπων σταδίων. Είναι υπεύθυνο ώστε να εκτελεί τα υπόλοιπα στάδια με την κατάλληλη ακολουθία, δίνοντας τις απαραίτητες παραμέτρους για την ομαλή λειτουργία των σταδίων και της επικοινωνίας μεταξύ τους
2. **Load:** τα δεδομένα φορτώνονται, είτε από τοπική είτε από απομακρυσμένη βάση δεδομένων και αποθηκεύονται εσωτερικά στο mlflow (artifacts). Το στάδιο αυτό είναι σημαντικό για την δίνει την δυνατότητα στον χρήστη να προσαρμόσει την είσοδο.πηγή των δεδομένων χωρίς να επηρεάσει τα υπόλοιπα στάδια του pipeline
3. **Preprocessing:** τα δεδομένα αναλύονται και διορθώνονται πιθανές ανωμαλίες (πχ outliers) στις χρονοσειρές. Στο στάδιο αυτό αξίζει να αναφερθεί η δυνατότητα μετατροπής των δεδομένων είτε σε UTC, είτε σε τοπική ζώνη ώρας τους πριν την επεξεργασία τους. Έτσι, ολοκληρώθηκε η επεξεργασία των δεδομένων και κάθε στάδιο που ακολουθεί χρησιμοποιεί τις χρονοσειρές όπως αποθηκεύτηκαν μετά την ολοκλήρωση του σταδίου
4. **Tuning:** Το στάδιο αυτό είναι προαιρετικό και περιλαμβάνει την χρήση του framework *Optuna* για την ρύθμιση των υπερπαραμέτρων του μοντέλου. Εναλλακτικά, ο χρήστης μπορεί να παρέχει τις τιμές των υπερπαραμέτρων από την είσοδο που θα χρησιμοποιηθούν στο ακόλουθο στάδιο
5. **Training:** Η εκπαίδευση του μοντέλου με χρήση είτε των παραμέτρων του προηγούμενου σταδίου, είτε δοσμένο χειροκίνητα στην είσοδο. Το μοντέλο λαμβάνει, όπως προαναφέρθηκε, τις χρονοσειρές από το στάδιο ETL και δημιουργείται ένα μοντέλο για την προσέγγιση (model fitting) των δεδομένων αυτών. Στην περίπτωση που γίνεται χρήση transfer learning, χρησιμοποιείται επιπλέον μοντέλο για την αρχικοποίηση των παραμέτρων με βάση αυτό
6. **Evaluation:** Αξιολόγηση των προβλέψεων του μοντέλου με χρήση γραφικών και μετρικών. Για την πληρότητα της αξιολόγησης, πραγματοποιείται μια διαδικασία ονόματι *model ensemble* (περισσότερα στο κεφάλαιο 5.2.5) στην οποία πολλά αντίγραφα του μοντέλου χρησιμοποιούνται για την αξιολόγηση των αποτελεσμάτων



Σχήμα 5.2: Απεικόνιση της διαδικασίας εκτέλεσης των σταδίων του machine learning lifecycle ως μέρος ενός pipeline

5.2.1 Extract - Transform - Load (ETL)

Το στάδιο αυτό περιλαμβάνει τον κώδικα εξαγωγής των δεδομένων. Το σύστημα παρέχει στον χρήστη του λογισμικού να πειράξει αυτό τον κώδικα και να επιλέξει τον τρόπο με τον οποίο θα διαβάζει τα δεδομένα. Η default επιλογή είναι η τοπική αναζήτηση φακέλου με τα αρχεία του dataset σε μορφή .csv. Δίνοντας όμως το κατάλληλο σύνδεσμο, δίνεται η επιλογή στον χρήστη να αναζητήσει το φάκελο με datasets χρησιμοποιώντας:

1. HTTP request

2. mlflow run artifact search
3. local file path
4. Amazon S3 bucket (integrated with Mlflow)

Ο κώδικας έχει σαν παραμέτρους εισόδου τον σύνδεσμο για την φόρτωση των αρχείων και ένα comma-separated string με τα ονόματα datasets που χρειαζόμαστε από τον εν λόγω φάκελο των datasets. Η διαδικασία είναι πολύ απλή: αναγνωρίζεται η κατηγορία του συνδέσμου, διαβάζονται τα datasets με βάση τις χώρες που παρείχε ο χρήστης και αποθηκεύονται σαν mlflow artifacts για χρήση από το επόμενο στάδιο

5.2.2 Data Preprocessing

Τα δεδομένα, αν και έγκυρα περιλαμβάνουν πολλές ανώμαλες τιμές (anomalies) οι οποίες οφείλονται σε σφάλματα των δεδομένων (discontinuities, outliers etc) και πρέπει να εξεταστούν και, αν κριθεί απαραίτητο, να επεξεργαστούν. Αρχικά, γίνεται έλεγχος την σωστής ανάγνωσης των εν λόγω αρχείων. Εξετάζεται αν έχουν ημερομηνίες σε αύξουσα σειρά και τον σωστό αριθμό και όνομα στηλών. Οι περισσότερες από αυτές τις στήλες χρησιμοποιούνται στο στάδιο του EDA στο κεφάλαιο 6 για την καλύτερη ανάλυση του.

Αρχικά, ενώνουμε τα αρχεία με βάση τις χώρες προέλευσης τους (merge). Η λογική πίσω από αυτό βρίσκεται στο γεγονός ότι κάποιες χώρες χωρίζονται σε πολλά BZ (bidding zones) και άρα πολλές και διαφορετικές χρονοσειρές. Συνεπώς, αυτές οι χώρες θα επηρεάζουν περισσότερο τα αποτελέσματα του μοντέλου από ότι άλλες χώρες (όπως η Γερμανία).

Σε πολλές χρονοσειρές υπάρχουν (εσφαλμένα) μηδενικές τιμές και outliers (τιμές πολύ μεγαλύτερες ή πολύ μικρότερες από τις γειτονικές τους) με κίνδυνο να προκαλέσουν προβλήματα στην εκπαίδευση των μοντέλων. Έτσι, γίνεται αρχικά αντικατάσταση τους με NaNs. Για την επιλογή αυτών των outliers, υπολογίζεται ο μέσος όρος και η τυπική απόκλιση για κάθε μήνα στο dataset και αφαιρούνται όσες τιμές απέχουν σημαντικά του μέσου όρου. Επιλέχθηκε ως default τιμή της τυπικής απόκλισης το 4.5% ώστε να αφαιρούνται οι τιμές με την μέγιστη πιθανότητα να είναι πράγματι outliers. Αξίζει να αναφερθεί πως εφόσον οι σειρές δεν εμφανίζουν τάση, δεν χρειάζεται να πραγματοποιηθεί de-trending.

Τα δεδομένα είναι αρχικά σε UTC (+0) time zone. Αν και πρακτική σύμβαση, το γεγονός ότι οι χώρες ρεαλιστικά βρίσκονται σε διαφορετικά time zones, και άρα πολλές κομβικές καθημερινές χρονικές στιγμές δεν είναι κοινές για κάθε χώρα (πχ ώρες αιχμής) ενδεχομένως

μπορεί να επηρεάσει σημαντικά την ακρίβεια του μοντέλου. Έτσι, σχεδιάστηκε η εκδοχή στην οποία οι χώρες διατηρούν το time zone τους. Στην περίπτωση, όμως, που πρέπει να διατηρηθούν οι χρονικές ζώνες της κάθε χώρας, πρέπει να γίνει έλεγχος αν μια χώρα βρίσκεται σε πολλαπλά time zones. Ευτυχώς, οι χώρες αφορούν την Ευρώπη (και τα Ευρωπαϊκά μέρη τους) όπου όλες οι χώρες βρίσκονται σε μια χρονική ζώνη (δίχως να υπολογίζονται φυσικά οι αποικίες μιας χώρας).

Τέλος, γίνεται αντικατάσταση των *NaNs* με αριθμητικές τιμές τροποποιώντας υπάρχουσες τεχνικές imputation [73]. Χρησιμοποιείται ένας σταθμισμένος μέσος όρος των ιστορικών δεδομένων και γραμμικής παρεμβολής. Τα βάρη της κάθε μεθόδου εξαρτώνται εκθετικά από την απόσταση της τρέχουσας *NaN* τιμής από την κοντινότερη ημερομηνία που έχει τιμή. Η τροποποίηση της υλοποίησης του paper, έγκειται στον ορισμό των ιστορικών δεδομένων. Πιο συγκεκριμένα, ο μέσος όρος των τιμών της ίδιας μέρας της εβδομάδας που επίσης έχουν λιγότερα από *WNcutoff* απόσταση από την ώρα της *NaN*, λιγότερο από *Ycutoff* από τον χρόνο της *NaN* και λιγότερο από *YDcutoff* από την ημέρα του χρόνου της *NaN*. Απόδειξη της αποτελεσματικότητας του imputation αποτελεί το σχήμα 7.1 στο οποίο φαίνεται η ασυνέχεια των δεδομένων για την χρονοσειρά της Ελλάδας και η επεξεργασία που έγινε στο imputation

5.2.3 Hyper-parameter Optimization (Optuna)

Η Optuna χρησιμοποιεί συνολικά την παρακάτω στρατηγικές για την εύρεση του καλύτερου συνδυασμού υπερπαραμέτρων.

- **Στρατηγική δειγματοληψίας (sampling):** Χρησιμοποιεί έναν αλγόριθμο δειγματοληψίας για την επιλογή του καλύτερου συνδυασμού υπερπαραμέτρων από έναν κατάλογο όλων των πιθανών συνδυασμών. Επικεντρώνεται σε περιοχές όπου οι υπερπαραμετροί δίνουν καλά αποτελέσματα και αγνοεί άλλες με αποτέλεσμα την εξοικονόμηση χρόνου.
- **Στρατηγική κλαδέματος (pruning):** Χρησιμοποιεί μια στρατηγική κλαδέματος που ελέγχει συνεχώς την απόδοση του αλγορίθμου κατά τη διάρκεια της εκπαίδευσης και κλαδεύει (τερματίζει) την εκπαίδευση για συγκεκριμένο συνδυασμό υπερπαραμέτρων, εάν δεν δίνει καλά αποτελέσματα. Αυτό έχει επίσης ως αποτέλεσμα την εξοικονόμηση χρόνου

Στην Optuna χρησιμοποιείται μια objective function, ορισμένη από τον χρήστη. Αυτή η συνάρτηση θα περιέχει την λογική για τη δημιουργία ενός μοντέλου, την εκπαίδευσή του και την αξιολόγησή του στο σύνολο δεδομένων επικύρωσης. Η συνάρτηση αυτή και κάθε εκτέλεση της, αξιολογεί ένα trial: ένα συνδυασμός τιμών των υπερπαραμέτρων. Μετά την αξιολόγηση, θα πρέπει να επιστρέφει μια ενιαία τιμή η οποία είναι γενικά η έξοδος της μετρικής αξιολόγησης ή της συνάρτησης απώλειας (ακρίβεια, MSE, MAE, MAPE etc) και πρέπει να ελαχιστοποιηθεί/μεγιστοποιηθεί

Αξίζει να σημειωθεί στον Trainer του μοντέλου, ότι γίνεται χρήση *EarlyStopping* και *Pruning*. *Early stopping* είναι μια μέθοδος που επιτρέπει να καθοριστεί ένας αυθαίρετα μεγάλος αριθμός εποχών (epochs) εκπαίδευσης και να σταματήσει η εκπαίδευση μόλις η απόδοση του μοντέλου σταματήσει να βελτιώνεται σε ένα σύνολο δεδομένων επικύρωσης (validation). Η απόδοση του μοντέλου φυσικά ελέγχεται από την ελαχιστοποίηση του validation loss, που υπολογίζεται στο validation loop. Ως έξοδο της μετρικής αξιολόγησης ορίζουμε την ελαχιστοποίηση της απώλειας επιβεβαίωσης (*validation loss*). Εντός της objective function, υπολογίζεται μια πρόταση (suggestion) για την τιμή κάθε υπερπαραμέτρου με βάση.

Ο πίνακας 7.1 έχει μια αναλυτική λίστα από τις παραμέτρους της optuna. Ενώ θα μπορούσαν να μελετηθούν όλες αυτές από την optuna για την αναζήτηση του απόλυτου βέλτιστου μοντέλου, ορισμένες επιλέχθηκαν στοχευμένα να μην διερευνηθούν ώστε να μην υπάρξει περιττή περιπλοκότητα στο μοντέλο. Για παράδειγμα, για την διερεύνηση της παραμέτρου του μέγιστου πλήθους των εποχών (*max_epochs*) θα χρειαζόντουσαν εκθετικά περισσότερα *trials*. Έτσι, είναι πιο απλό και παραγωγικό να τεθεί μια αυθαίρετα μεγάλη (σχεδόν απροσέγγιστη) τιμή για αυτήν και, κάνοντας χρήση της τεχνικής *EarlyStopping*, να βρεθεί για κάθε περίπτωση η καλύτερη τιμή.

Οι τιμές αυτές ορίζονται σε ένα εύρος που ορίζεται από τον χρήστη στην έναρξη της εκτέλεσης του optimization. Η διαδικασία δοκιμής διαφορετικών συνδυασμών υπερπαραμέτρων για την εύρεση του συνδυασμού που δίνει τα καλύτερα αποτελέσματα ονομάζεται *study* και συχνά αποτελείται από πολλά *trials*. Ένα study ολοκληρώνεται στην περίπτωση που:

1. εκτελεστεί ο μέγιστος αριθμός trials, ορισμένο από τον χρήστη
2. παρέλθει ένα συγκεκριμένο χρονικό διάστημα από την έναρξη της εκτέλεσης, επίσης ορισμένο από τον χρήστη
3. ο pruner κρίνει ότι δεν επιδέχονται άλλη βελτιστοποίηση οι υπερπαραμέτροι του μο-

ντέλου

Μόλις ολοκληρωθεί το study, αποθηκεύονται οι τιμές του καλύτερου trial ως pickle (pkl) και ως απλού κειμένου (.txt) αρχείου [74], καθώς και κάποια σημαντικά γραφήματα της διαδικασίας του study σαν artifacts στο mlflow run του optimization

5.2.4 Model Training

Όπως προαναφέρθηκε, το μοντέλο, η εκπαίδευση και οι προβλέψεις που προέκυψαν έγιναν χρησιμοποιώντας ως framework το pytorch lightning κάνοντας την χρήση ενός LightningModule. Ένα LightningModule οργανώνει τον κώδικα PyTorch σε 4 κύριες κατηγορίες:

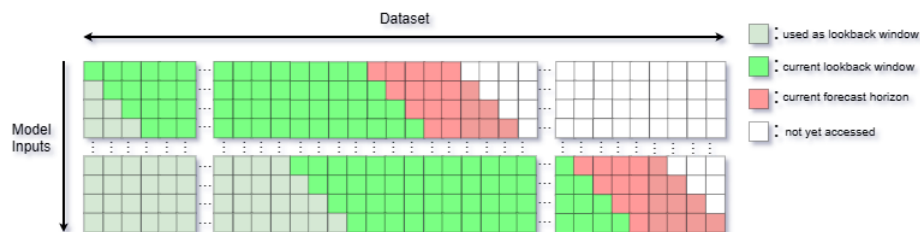
- **Constructor:** Αποτελεί την δημιουργία του μοντέλου και μια πρώτη επεξεργασία των εισόδων του όπως δεδομένα και (υπερ)παράμετροι
- **Dataloaders:** Τα δεδομένα της χρονοσειράς θα πρέπει να δίνονται με τρόπο ανάλογο (της εισόδου) της αρχιτεκτονικής του μοντέλου. Οι συναρτήσεις αυτές είναι υπεύθυνες για την οργάνωση των δεδομένων προκειμένου να λειτουργήσουν αποτελεσματικά σαν είσοδος για το μοντέλο για τις επαναληπτικές διαδικασίες
- **Train/Test/Validation/Predict Loops:** Οι επαναληπτικές διαδικασίες επεξεργασίας της εισόδου του μοντέλου για ένα βήμα της επεξεργασίας τους. Τα δεδομένα δίνονται στην είσοδο, επεξεργάζονται από το μοντέλο (forward pass) και υπολογίζεται η απώλεια (loss)
- **Optimizers και LR Schedulers:** Ο υπολογισμός του optimizer που χρησιμοποιείται και η ανανέωση των υπερπαραμέτρων που υπολογίζει

Τα περισσότερα στάδια, λόγω της εύκολης χρήσης του *Lightning* (βλέπε κεφάλαιο 5.1.3) είναι αρκετά τυποποιημένα και δεν χρειάζεται ιδιαίτερη εμβάθυνση. Αξιοσημείωτη όμως αποτελεί η κατασκευή του μοντέλου και οργάνωση των δεδομένων που γίνεται στον κατασκευαστή του LightningModule. Στον κατασκευαστή παρέχονται οι (υπερ)παράμετροι του μοντέλου και εσωτερικά καλούνται δύο ειδικές συναρτήσεις: η *make_hidden_layers* και η *feature_target_split*.

Η πρώτη προσθέτει στα στρώματα του μοντέλου εσωτερικά ζεύγη από γραμμικά στρώματα (linear layers) και συναρτήσεις ενεργοποίησης με γεωμετρία και τύπο ανάλογο με τις

παραμέτρους που δόθηκαν. Η δεύτερη, αξιοποιώντας την βιβλιοθήκη *Pandas* [52], λαμβάνει ένα dataframe με δεδομένα ανάλογα του loop (train/test/validation) καθώς και τα μεγέθη εισόδου/εξόδου (lookback window/forecast horizon) του μοντέλου. Χρησιμοποιείται για να γίνει μια πρώτη επεξεργασία των δεδομένων πριν δοθούν για περαιτέρω οργάνωση από τους dataloaders.

Πιο συγκεκριμένα, δομεί τα δεδομένα φορτίου (μια στήλη στο dataframe ονόματι *Load*) σε σειρές. Κάθε σειρά αποτελείται από δεδομένα πλήθους *lookback window* (έστω M) ακολουθούμενα από τα αμέσως επόμενα (χρονικά) δεδομένα πλήθους *forecast horizon* (έστω N). Κάθε γραμμή αντιστοιχεί στην προηγούμενη της μετακινώντας το συνολικό παράθυρο των δεδομένων ($M+N$) μετατοπισμένο κατά ένα δεδομένο δεξιά, όπως φαίνεται και στο σχήμα 5.3:



Σχήμα 5.3: Απεικόνιση του αλγορίθμου για την δημιουργία δεδομένων εισόδου και labels εξόδου. Κάθε κελί αντιπροσωπεύει ένα σταθερό πλήθος δεδομένων φορτίου, ενώ κάθε γραμμή είναι ένας συνδυασμός feature - target, αναλογικά με το συνολικό dataset (όλα τα κελιά). Οι σειρές δίνονται στο σύστημα χρονολογικά από πάνω προς τα κάτω και οι στήλες διαβάζονται χρονολογικά από δεξιά προς τα αριστερά

Έχοντας δομήσει τον κώδικα PyTorch σε ένα *LightningModule*, χρησιμοποιούμε την κλάση *Trainer*: υπεύθυνη για την αυτοματοποιημένη εκπαίδευση και πρόβλεψη των δεδομένων με βάση το μοντέλο. Ο *Trainer* του μοντέλου είναι παρόμοιος με τον *Trainer* του προηγούμενου σταδίου, χωρίς φυσικά την συνάρτηση *callback* για *pruning*, μια λειτουργία αποκλειστική της *Ortuna*.

5.2.5 Ensemble

Η κύρια αρχή εκπαίδευσης των παραμέτρων στα μοντέλα μηχανικής μάθησης είναι πως αρχικοποιούνται σε μια αρχική, και συχνά τυχαία, κατάσταση (*random state*) και βηματικά οι

παράμετροι προσεγγίζουν τις βέλτιστες τιμές τους. Έτσι, είναι προφανές ότι μια διαφορετική αρχικοποίηση των παραμέτρων, είναι δυνατό να έχουμε αποκλίσεις από την βέλτιστη τιμή. Έτσι, για να έχουμε την καλύτερη (ή την πιο αντιπροσωπευτική) πρόβλεψη του μοντέλου, είναι απαραίτητο να γίνουμε πολλές εκπαιδεύσεις περιπτώσεων (διαφορετικά random states -estimators-) του μοντέλου για να εξαχθεί ένα ενιαίο, βέλτιστο συμπέρασμα. Η διαδικασία αυτή επιτυγχάνεται με μεθόδους *ensemble learning*.

Για την υλοποίηση του, ο κώδικας διακλαδίζεται σε δύο περιπτώσεις, ανάλογα με τις παραμέτρους που το δόθηκαν. Στην περίπτωση που του δοθεί μονοπάτι (path) προς ήδη υλοποιημένο μοντέλο ensemble, τότε θα χρησιμοποιήσει εκείνο το μοντέλο για πρόβλεψη και αξιολόγηση των αποτελεσμάτων. Σε αντίθετη περίπτωση, κατασκευάζεται το forecasting μοντέλο, και χρησιμοποιείται σαν πρότυπο για τους estimators και την εκπαίδευση του ensemble μοντέλου.

Υπάρχουν διάφορες τεχνικές για ensemble learning με την βιβλιογραφία να επισημαίνει:

1. **Bagging:** ακρωνύμιο για 'Bootstrap Aggregation' είναι μια παράλληλη μέθοδος (οι estimators εκτελούνται ανεξάρτητα μεταξύ τους) που συνδυάζει τα αποτελέσματα πολλαπλών μοντέλων/estimators (aggregation) για να προκύψει ένα γενικευμένο αποτέλεσμα (μειωμένο variance). Δημιουργεί πρόσθετα δεδομένα για εκπαίδευση από το σύνολο δεδομένων με τη χρήση *bootstrapping*: μια τεχνική δειγματοληψίας στην οποία δημιουργούμε υποσύνολα παρατηρήσεων από το αρχικό σύνολο δεδομένων, με αντικατάσταση (τα στοιχεία μπορούν να επαναχρησιμοποιηθούν για μελλοντικά δείγματα)
2. **Boosting:** μια ακολουθιακή τεχνική η οποία όπως η bagging, συνδυάζει πολλαπλούς αδύναμους estimators για την δημιουργία ενός νέου ισχυρότερου. Αντίθετα όμως με την bagging, οι estimators βρίσκονται σε μια ακολουθία, με κάθε estimator στην ακολουθία να δίνει μεγαλύτερη σημασία στις παρατηρήσεις του συνόλου δεδομένων που αντιμετωπίστηκαν άσχημα από τους προηγούμενους.
3. **Stacking:** μια μέθοδος που χρησιμοποιεί προβλέψεις από πολλαπλά μοντέλα για τη δημιουργία ενός νέου μοντέλου το οποίο χρησιμοποιείται για την πραγματοποίηση προβλέψεων στο σύνολο δοκιμών. Λαμβάνει τις εξόδους των επιμέρους παράλληλων μοντέλων ως είσοδο και προσπαθεί να μάθει πώς να συνδυάσει καλύτερα τις προβλέψεις εισόδου για να κάνει μια καλύτερη πρόβλεψη εξόδου.

Αν και οποιαδήποτε μέθοδος ensemble θα είχε ικανοποιητικά αποτελέσματα, επιλέχθηκε

η χρήση της *bagging* ως μια από τις πιο χρησιμοποιημένες τεχνικές [75]. Αν και η *boosting* δυνητικά μπορεί να είναι αναβάθμιση της *bagging*, υπάρχει σημαντικό ρίσκο *over-fitting* του μοντέλου [76], κάτι ανεπιθύμητο ειδικά για τους σκοπούς τους *transfer learning*. Αξιόλογος παράγοντας αποτέλεσε και η ο συνδυασμός απλότητα και αξιοπιστίας της *bagging*, συγκριτικά με την *stacking* που, αν και αποτελεσματική, είναι χρονοβόρα, δυσκολότερη στην ανάπτυξη και συντήρηση και λιγότερο απλή στην ερμηνεία [77]

5.2.6 Evaluation

Κατά την αξιολόγηση ενός μοντέλου πρόβλεψης χρονοσειρών, μας ενδιαφέρει η απόδοση του μοντέλου σε δεδομένα που δεν χρησιμοποιήθηκαν για την εκπαίδευσή του. Στη μηχανική μάθηση, τα δεδομένα αυτά τα ονομάζουμε αφανή ή εκτός δείγματος δεδομένα.

Μπορούμε να το κάνουμε αυτό χωρίζοντας τα δεδομένα που έχουμε στη διάθεσή μας. Χρησιμοποιούμε κάποια για να προετοιμάσουμε το μοντέλο και κρατάμε πίσω κάποια δεδομένα και ζητάμε από το μοντέλο να κάνει προβλέψεις για εκείνη την περίοδο. Η αξιολόγηση αυτών των προβλέψεων θα παρέχει ένα καλό υποκατάστατο για το πώς θα αποδώσει το μοντέλο όταν το χρησιμοποιήσουμε επιχειρησιακά

Στην εφαρμοσμένη μηχανική μάθηση, συχνά χωρίζουμε τα δεδομένα μας σε ένα σύνολο εκπαίδευσης και ένα σύνολο δοκιμής: το σύνολο εκπαίδευσης που χρησιμοποιείται για την προετοιμασία του μοντέλου και το σύνολο δοκιμής που χρησιμοποιείται για την αξιολόγησή του.

Παρόλα αυτά, η φύση των δεδομένων παίζει σημαντικό ρόλο στην επιλογή αυτών των συνόλων. Αν δεν υπάρχει σχέση μεταξύ των παρατηρήσεων, ότι κάθε παρατήρηση είναι ανεξάρτητη, τα σύνολα μπορεί να είναι και ανεξάρτητα μεταξύ τους. Αυτό όμως δεν ισχύει για τα δεδομένα χρονοσειρών, όπου η χρονική διάσταση των παρατηρήσεων σημαίνει ότι δεν μπορούμε να τις χωρίσουμε τυχαία σε ομάδες. Αντ' αυτού, πρέπει να χωρίσουμε τα δεδομένα και να σεβαστούμε τη χρονική σειρά με την οποία παρατηρήθηκαν οι τιμές.

Στην πρόβλεψη χρονοσειρών, αυτή η αξιολόγηση των μοντέλων σε ιστορικά δεδομένα ονομάζεται *backtesting*: σε κάθε στιγμή στο σύνολο δεδομένων, εκπαιδεύεται το μοντέλο σε γνωστά/παρελθόντα δεδομένα εκείνη τη στιγμή και έπειτα δοκιμάζεται σε άγνωστα/μελλοντικά δεδομένα εκείνη τη στιγμή [78]

Στην αξιολόγηση του μοντέλου χρησιμοποιήθηκαν πολλαπλές μετρικές (MASE, RMSE etc) που αναπτύχθηκαν στο κεφάλαιο 2.5.3 με την κυριότερη σαν κύρια αυτή του MAPE

(Mean Absolute Percentage Error).

Κεφάλαιο 6

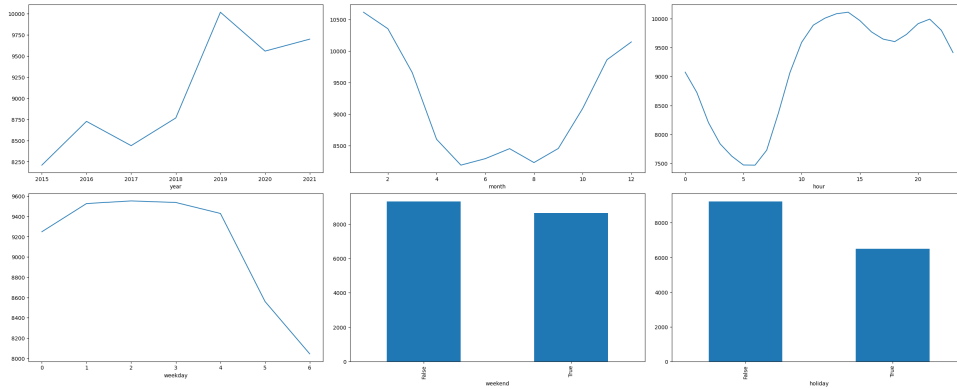
Ανάλυση Δεδομένων (EDA)

6.1 Mean Plots

Όπως αναλύθηκε στο κεφάλαιο 4, EDA είναι η διαδικασία εξαγωγής μοτίβων (συνήθως με οπτικό τρόπο) από τα δεδομένα. Για το EDA, διεξάχθηκαν αρκετές απεικονίσεις για την καλύτερη κατανόηση τους.

Αρχικά, έγινε μια γενική εξέταση της συμπεριφοράς (τάσης) των δεδομένων. Από τα γραφήματα στο σχήμα 6.1 συμπεράνουμε:

- μια ανοδική τάση στις τιμές των φορτίων κατά μέσο όρο: καθώς αυξάνεται το έτος, αυξάνεται το φορτίο
- από την μηνιαία παρατήρηση, μια ανοδική αύξηση της ζήτησης φορτίου από τον Σεπτέμβριο έως τον Δεκέμβριο και σταθερή μείωση από τον Ιανουάριο έως τον Μάιο (χειμερινοί μήνες)
- μια αύξηση της ζήτησης φορτίου κατά τις εργάσιμες ώρες (με μικρή πτώση το μεσημέρι) και πτώση αντισυμμετρική της αρχικής αύξησης καθώς πλησιάζει η νύχτα
- πως το φορτίο είναι μεγαλύτερο κατά μέσο όρο τις εργάσιμες ημέρες. Μάλιστα, παρατηρείται αλματώδης πτώση στο Σάββατο και στην Κυριακή (μέρες 5 και 6). Μάλιστα, αυτή η αρχή εφαρμόζεται και στην μέγιστη αριθμητική τιμή με τα σαββατοκύριακα και αργίες να έχουν μικρότερη μέγιστη τιμή από τις εργάσιμες μέρες.

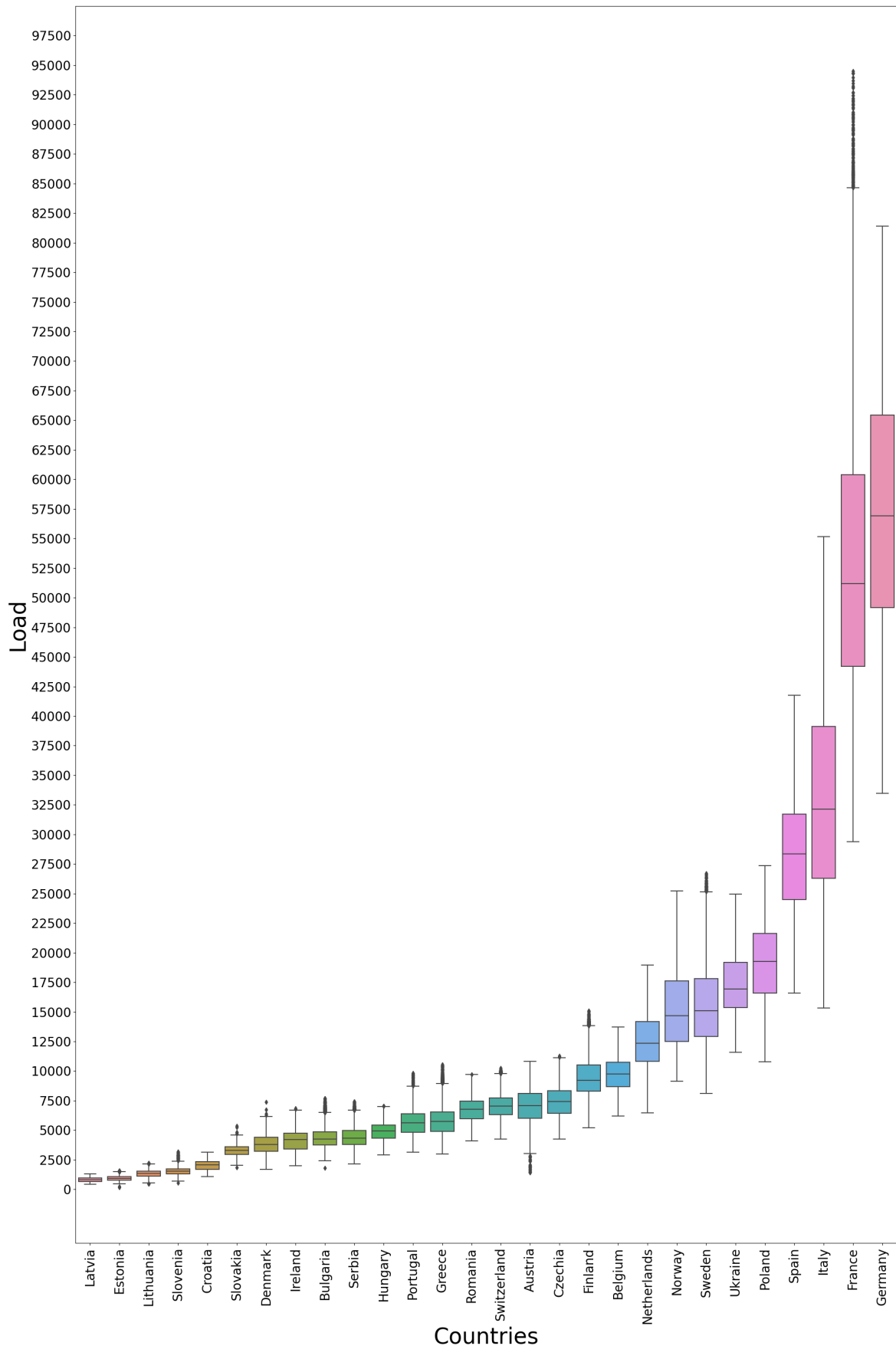


Σχήμα 6.1: Γραμμική απεικόνιση της μέσης τιμής του φορτίου των δεδομένων ανά έτος/μήνα/ώρα/μέρα της εβδομάδας. Επίσης, τα τελευταία δύο γραφήματα συγκρίνουν πόσα δεδομένα φορτίου είναι σε μέρες που είναι σαββατοκύριακα/διακοπές ή όχι

6.2 Box-plots

Με σκοπό το transfer learning, είναι σημαντικό να εξεταστεί η φύση των δεδομένων ανα χώρα. Έτσι, σχεδιάστηκαν σε μορφή box-plot οι κατανομές φορτίου των χωρών του dataset. Για την καλύτερη σύγκριση τους, τα box-plots, έγιναν σε κοινό γράφημα βάζοντας τα στην σειρά. Από το σχήμα 6.2, είναι εύκολο να εξαχθεί ότι τα δεδομένα των χωρών δεν κυμαίνονται στις ίδιες τιμές και άρα καθίσταται απαραίτητη η κανονικοποίησή (normalization) τους. Επίσης, είναι άξιο παρατήρησης πως η ζήτηση της ηλεκτρικής ενέργειας μιας χώρας οφείλεται στην παραγωγή και στον πληθυσμό της χώρας.

Αυτό γίνεται ξεκάθαρο, παρατηρώντας χώρες με μεγάλη βιομηχανία και πληθυσμό (Ιταλία, Ισπανία, Γερμανία κτλ) να έχουν μεγάλη ζήτηση, χώρες με μικρή βιομηχανία και πληθυσμό (πχ Βαλτικές χώρες) να έχουν την μικρότερη ζήτηση. Αυτό εξηγεί επίσης, γιατί χώρες με μεγάλη βιομηχανία και μικρό (αναλογικά) πληθυσμό, όπως οι Σκανδιναβικές, έχουν τόσο μεγάλη ζήτηση, συγκριτικά με χώρες με μικρή βιομηχανία, αλλά μεγαλύτερο πληθυσμό (πχ Ελλάδα).



Σχήμα 6.2: Side-by-side box-plots της κατανομής των φορτίων για κάθε χώρα του dataset

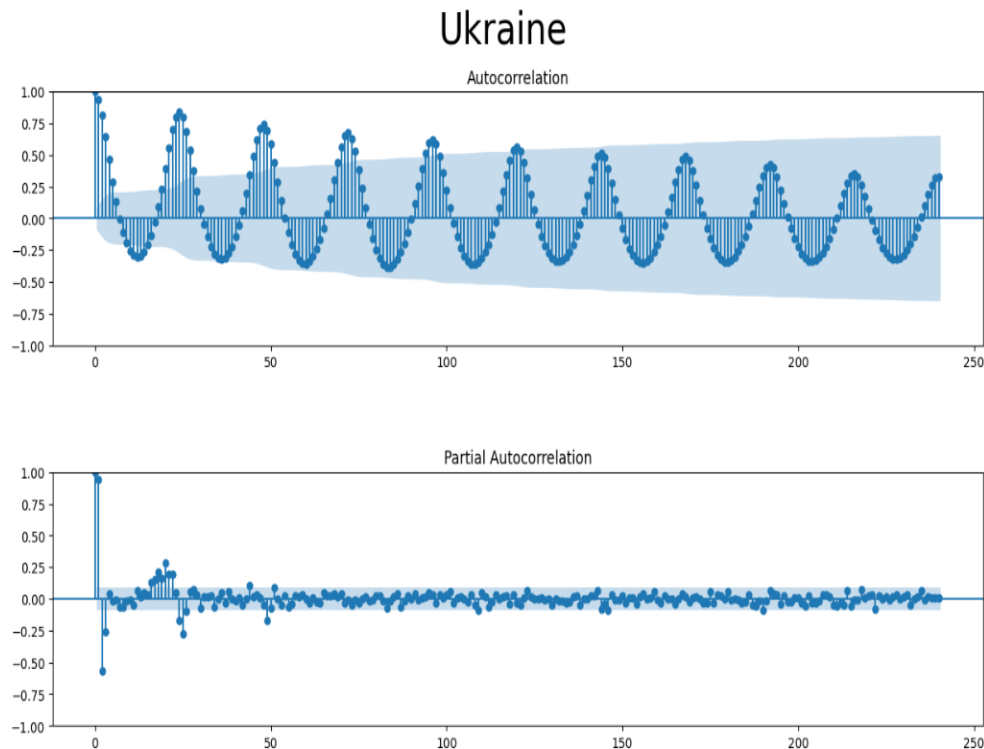
6.3 ADF Test

6.4 ACF/PACF Plots

Τα διαγράμματα ACF προσδιορίζουν τη συσχέτιση μεταξύ των δεδομένων στον τρέχοντα χρόνο και με τις παρελθοντικές τιμές (παρελθοντικές τιμές 1 μονάδας, παρελθοντικές τιμές 2 μονάδων, ..., παρελθοντικές τιμές n μονάδων). Τις χρησιμοποιούμε για να προσδιορίσουμε την εποχικότητα, την τάση, την κυκλικότητα και το υπόλειμμα τα οποία λαμβάνονται υπόψη. Τα διαγράμματα PACF εμφανίζουν είναι το ποσό της συσχέτισης μεταξύ μιας μεταβλητής και μιας υστερήσης της ίδιας που δεν εξηγείται από τις συσχετίσεις σε όλες τις υστερήσεις χαμηλότερης τάξης. Χρησιμοποιούνται κυρίως για να προσδιορίσουν πώς οι επιμέρους υστερήσεις επηρεάζουν τον παρόντα χρόνο χωρίς να υπάρχουν ενδιάμεσες υστερήσεις και καθιστούν σαφέστερο το ποιο μοντέλο μπορεί να εκτιμήσει καλύτερα τη χρονοσειρά.

- **Autoregressive(AR) Model:** Για ένα μοντέλο AR, σίγουρα θα παρατηρήσουμε το διάγραμμα ACF του να έχει ένα κωνικό ή ημιτονοειδές μοτίβο που συγκλίνει στο 0, ενδεχομένως με εναλλασσόμενα αρνητικά και θετικά πρόσημα. Όσο για το διάγραμμα PACF του, θα εμφανίζει σημαντικές τιμές στις πρώτες p υστερήσεις, στη συνέχεια μη σημαντική τιμή.
- **Μοντέλο κινητού μέσου(MA):** Για ένα υπόδειγμα MA, θα παρατηρήσουμε σίγουρα το διάγραμμα ACF του να έχει σημαντικές τιμές στις πρώτες q υστερήσεις, και στη συνέχεια μη σημαντικές τιμές. Όσο για το διάγραμμα PACF, θα εμφανίζει κωνικό ή ημιτονοειδές μοτίβο που συγκλίνει στο 0, ενδεχομένως με εναλλασσόμενα αρνητικά και θετικά πρόσημα.

Από τα διαγράμματα των χωρών (με παραδειγματικό το σχήμα 6.1), γίνεται εμφανές ότι τα δεδομένα μας μπορούν να μοντελοποιηθούν με ένα καθαρό μοντέλο AR. Μπορούμε, επίσης, εύκολα να παρατηρήσουμε μια ομόρφωνη (για όλες τις χώρες) υψηλή αυτοσυσχέτιση ανά 24 time lags (ώρες στο περιεχόμενο των χρονοσειρών) .το υψηλό μοτίβο εποχικότητας του συνόλου δεδομένων, δεδομένου ότι η γραφική ταλαντώνεται ημιτονοειδώς περίπου κάθε 24 time lags (όπως στις 24 ώρες της ημέρας). Έτσι, προκύπτει σαν συμπέρασμα ότι υπάρχει υψηλή περιοδική συμπεριφορά σε ημερήσιο επίπεδο, κάτι το οποίο μπορεί να επιβεβαιωθεί από το ημερήσιο mean plot στο γράφημα.



Σχήμα 6.3: Χαρακτηριστικό ζεύγος ACF/PACF plots από την χρονοσειρά της Ουκρανίας

6.5 Dendrogram & Choropleth

Για τους σκοπούς του transfer learning, είναι ανάγκη να ομαδοποιηθούν οι χώρες μεταξύ τους σε clusters που θα λειτουργήσουν σαν source domains. Η ομαδοποίηση έγινε με hierarchical clustering και η αποτύπωση του γίνεται με την χρήση δενδρογράμματος και χαρτών choropleth.

6.5.1 Έννοιες & Υλοποίηση

Οι χάρτες Choropleth απεικονίζουν διαιρεμένες γεωγραφικές περιοχές ή περιφέρειες που είναι χρωματισμένες, σκιασμένες ή διαμορφωμένες σε σχέση με μια μεταβλητή δεδομένων. Αυτό παρέχει έναν τρόπο απεικόνισης των τιμών σε μια γεωγραφική περιοχή, η οποία μπορεί να δείξει διακυμάνσεις ή μοτίβα σε όλη την εμφανιζόμενη τοποθεσία.

Το δενδρόγραμμα απεικονίζει τον τρόπο με τον οποίο αποτελείται κάθε ομάδα/cluster, σχεδιάζοντας μια σύνδεση σχήματος U μεταξύ μιας μη-ομάδας και των παιδιών της.

- Η κορυφή του συνδέσμου U υποδηλώνει μια ένωση clusters

- Τα δύο σκέλη του συνδέσμου U υποδεικνύουν ποια clusters ενώθηκαν
- Το μήκος των δύο σκελών του συνδέσμου U αντιπροσωπεύει την απόσταση μεταξύ των θυγατρικών ομάδων.

Αρχικά, πριν από την έναρξη του αλγορίθμου, κάθε χαρακτηριστικό είναι μια ομάδα. Η βιβλιοθήκη `scipy.linkage` χρησιμοποιεί αυτό που ονομάζεται "συσσωρευτική ομαδοποίηση".

1. Ξεκινά με κάθε παρατήρηση ως τη δική της ομάδα.
2. Οι δύο πλησιέστερες ομάδες ενώνονται σε μία ομάδα.
3. Οι επόμενες πλησιέστερες συστάδες ομαδοποιούνται μαζί
4. GOTO 1: η διαδικασία αυτή συνεχίζεται μέχρι να υπάρχει μόνο μία ομάδα που να περιέχει ολόκληρο το σύνολο των δεδομένων.

Κάθε φύλλο στο δενδρόγραμμα αντιπροσωπεύει ένα χαρακτηριστικό και κάθε κόμβος μια ομάδα. Ο άξονας y δείχνει την απόσταση μεταξύ των σημείων. Ο αριθμός των ομάδων στα δεδομένα θα εξαρτηθεί από την απόσταση που θα οριστεί ως κατώφλι. Εάν επιλεγθεί μια μικρή απόσταση, θα σχηματιστούν περισσότερες ομάδες. Αντίθετα, αν επιλέξουμε μια μεγάλη απόσταση ως κατώφλι, θα έχουμε λιγότερες ομάδες.

Οι πιο δημοφιλείς τρόποι ορισμού της απόστασης μεταξύ των ομάδων [79] είναι οι εξής μέθοδοι:

- **Single linkage:** Η απλή σύνδεση (κοντινότερος γείτονας) είναι η μικρότερη απόσταση μεταξύ ενός ζεύγους παρατηρήσεων σε δύο clusters. Μπορεί μερικές φορές να παράγει ομάδες όπου οι παρατηρήσεις σε διαφορετικά cluster είναι πιο κοντά μεταξύ τους από ό,τι οι παρατηρήσεις μέσα στις δικές τους ομάδες. Αυτές οι ομάδες μπορεί να φαίνονται διασκορπισμένες.
 - **Πρόβλημα:** τείνει να είναι εξαιρετικά μυωπική. Ένα αντικείμενο θα προστεθεί σε ένα σμήνος εφόσον βρίσκεται κοντά σε οποιοδήποτε από τα άλλα αντικείμενα του σμήνους, ακόμη και αν είναι σχετικά μακριά από όλα τα άλλα. Έτσι, η απλή σύνδεση έχει την τάση να παράγει μακριές, χορδωτές ομάδες.
- **Complete linkage:** Complete-linkage (πιο μακρινός γείτονας) είναι όταν η απόσταση μετράται μεταξύ του πιο απομακρυσμένου ζεύγους παρατηρήσεων σε δύο clusters.

Αυτή η μέθοδος παράγει συνήθως στενότερες ομάδες από ό,τι η απλή σύνδεση, αλλά αυτές οι στενές ομάδες μπορεί να καταλήξουν πολύ κοντά μεταξύ τους. Μαζί με το average-linkage, είναι μία από τις πιο δημοφιλείς μετρικές απόστασης.

- **Πρόβλημα:** μπορεί να είναι ιδιαίτερα ευαίσθητη στις ακραίες τιμές των δεδομένων.
- **Μέθοδος του Ward:** επιδιώκει να συνδέσει τις δύο ομάδες των οποίων η συγχώνευση οδηγεί στο μικρότερο άθροισμα τετραγώνων εντός της ομάδας (δηλαδή στην ελάχιστη διακύμανση εντός της ομάδας). Έχει την τάση να παράγει ομάδες ίσου μεγέθους που είναι συμπαγείς.

Αποφασίστηκε ότι η μέθοδος *ward* [80] [79] ήταν η καλύτερη επιλογή και παρείχε μια πιο λογική και διαισθητική ομαδοποίηση

Το δενδρόγραμμα που σχεδιάζεται μπορεί να χρησιμοποιηθεί για την ομαδοποίηση των χωρών με βάση την ομοιότητα των συνόλων δεδομένων τους για μεταγενέστερη μάθηση μεταφοράς. Δεδομένου ότι το δενδρόγραμμα σχεδιάζεται με βάση τη συσχέτιση των χωρών, αυτό σημαίνει ότι δεν επηρεάζεται από την κλίμακα ή την κατανομή των δεδομένων των χωρών. Για να έχουμε ένα πιο ολοκληρωμένο κριτήριο ομαδοποίησης, χρησιμοποιούμε θερμικά διαγράμματα με βάση τις χώρες και τις χρονικές συνδιακυμάνσεις, τα οποία επηρεάζονται από την κλίμακα/κατανομή των δεδομένων

Για την δημιουργία των clusters χρησιμοποιήθηκε η βιβλιοθήκη Scipy [56] της python. Η πειραματική ομαδοποίηση των χωρών γίνεται με βάση δύο παράγοντες: την συσχέτιση (correlation) τους, και με βάση τη μέση κατανάλωση τους ανα χρονικά διαστήματα (time profiles).

6.5.2 Μέθοδος 1: Correlation

Η πρώτη προσέγγιση αφορά το πόσο συσχετίζονται τα δεδομένα των χωρών. δηλαδή κοινή τάση και εποχικότητα. Τα δεδομένα αρχικά δίνονται σε μορφή pandas.DataFrame (df) στο οποίο κάθε γραμμή περιλαμβάνει την στήλη:

- φορτίου (Load)
- χώρα (country)
- χρονική στιγμή/ώρα (hour) που πάρθηκαν τα δεδομένα

- μήνας (month) που πάρθηκαν τα δεδομένα
- έτος (year) που πάρθηκαν τα δεδομένα

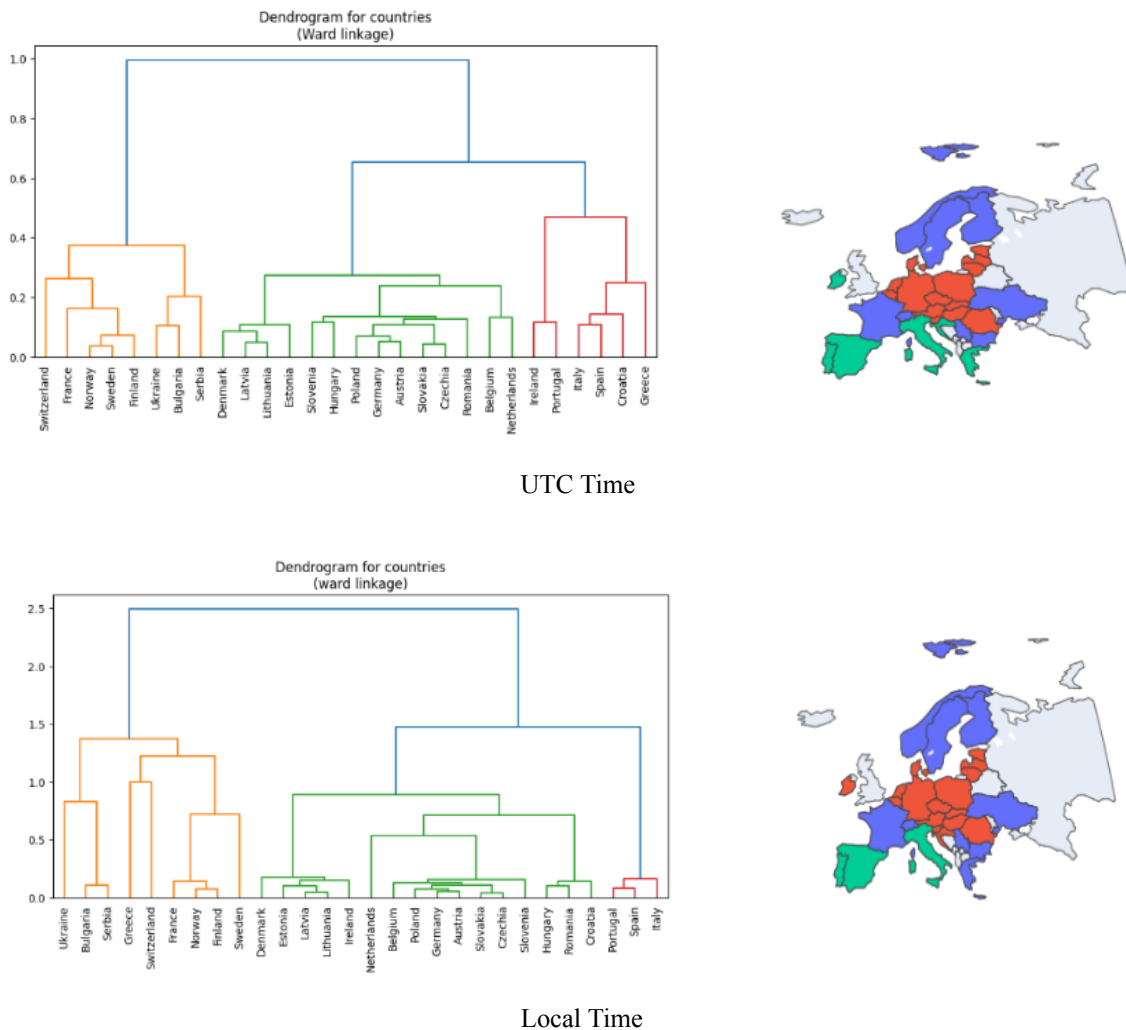
Για τον επεξεργασία του dataframe, πραγματοποιήθηκαν τα ακόλουθα βήματα:

1. **Μετατροπή dataframe σε pivot table:** Μας ενδιαφέρει από το αρχικό dataframe να έχουμε οργανωμένες τις σχέσεις μεταξύ της χώρας, του χρόνου και του εκάστοτε φορτίου. Έτσι, μετασχηματίζουμε το dataframe (df) σε ένα pivot table: κάθε στήλη αντιπροσωπεύει μια χώρα και κάθε γραμμή μια χρονική στιγμή (datetime). Σε κάθε κελί του table είναι η τιμή φορτίου (Load) μιας χώρας για μια χρονική στιγμή, με βάση την γραμμή και στήλη που βρίσκεται.
2. **Δημιουργία correlation matrix (corr):** Από αυτό το pivot table (dframe) υπολογίζουμε τον correlation matrix (corr): ένας πίνακας συσχετίσεων μεταξύ όλων των χωρών. Προκειμένου να γίνει το clustering με βάση το μέγιστο correlation, έχουμε ορίσει να μηδενίζονται οποιεσδήποτε correlations του corr matrix είναι κατώτερες από μια τιμή (πειραματικά προέκυψε 0.8)
3. **Μετατροπή correlation σε distance matrix (dissimilarity):** Το hierarchical clustering γίνεται με κριτήριο της αποστάσεις των χωρών του correlation matrix, αλλά τα correlations δεν είναι αποστάσεις. Εάν το correlation μεταξύ δύο χωρών είναι r , ορίζουμε σαν απόσταση $d = 1 - r$. Έτσι, μετατρέπουμε τον correlation matrix (corr) σε $dissimilarity = 1 - corr$.
4. **Μετατροπή για 1-D condensed distance matrix:** Η συνάρτηση `scipy.linkage` που πραγματοποιεί το clustering, χρειάζεται ως είσοδο έναν condensed distance matrix και όχι έναν 2-D redundant distance matrix όπως αυτός του προηγούμενου βήματος. Για την μετατροπή αυτή, χρησιμοποιείται η εντολή `scipy.squareform`
5. **Δημιουργία δενδρογράμματος:** Χρησιμοποιείται η εντολή `scipy.dendrogram` η οποία επιστρέφει το δενδρόγραμμα του clustering

Εξετάζουμε το δενδρόγραμμα πειραματικά για κάθε πιθανή δεκάδα (βήμα 0.1) του correlation. Επιλέχθηκε ως ελάχιστη τιμή συντελεστή συσχέτισης για την ομαδοποίηση την τιμή 0.8. Για τιμές μικρότερες ή ίσες του 0.8, παρατηρήθηκε μια έντονη συσχέτιση ανάμεσα σε χώρες με σχετικά κοντινή γεωγραφική θέση (Σκανδιναβικές, Μεσογειακές, Βαλτικές, χώρες

κεντρικής Ευρώπης κλπ). Το ίδιο φαινόμενο φαίνεται ακόμα πιο ξεκάθαρα και για τιμές μεγαλύτερες του 0.8, με όλη την νότια Ευρώπη, την κεντρική Ευρώπη και την Σκανδιναβική Ευρώπη και τις χώρες της Βαλτικής να είναι ξεχωριστές ομάδες

Στην εικόνα 6.4 αποτυπώνονται γραφικές για choropleth και το πηγαίο δενδρόγραμμα του για UTC και local time αντίστοιχα. Παρατηρούμε ότι τα δύο ζεύγη choropleth/dendrogram έχουν σημαντικές ομοιότητες στην ομαδοποίηση, σε βαθμό που είναι πιο εύκολο να αναπαριστούν οι διαφορές τους. Οι διαφορές βρίσκονται στις χώρες της Κροατίας και της Ελλάδας. Στην περίπτωση του UTC time zone, βρίσκονται με τις υπόλοιπες μεσογειακές χώρες (πχ Ισπανία). Αντίθετα, στην περίπτωση του local time zone, η Κροατία βρίσκεται με τις υπόλοιπες χώρες της Κεντρικής Ευρώπης (πχ Αυστρία), ενώ η Ελλάδα βρίσκεται με τις σκανδιναβικές και ορισμένες από τις γειτονικές της σλαβικές (πχ Βουλγαρία)



Σχήμα 6.4: Δενδρόγραμμα και σχετικό choropleth υπολογισμένο με βάση το correlation των χωρών σε UTC και local time

6.5.3 Μέθοδος 2: Time Profiles

Η δεύτερη προσέγγιση αφορά τα time profiles των χωρών. Βρίσκονται οι μέσες τιμές φορτίου ανά χώρα για κάθε ώρα/μήνα/έτος (χρονική διάρκεια) και σχεδιάζεται ένας διδιάστατος πίνακας χωρών - μέσων τιμών ανά χρονική διάρκεια.

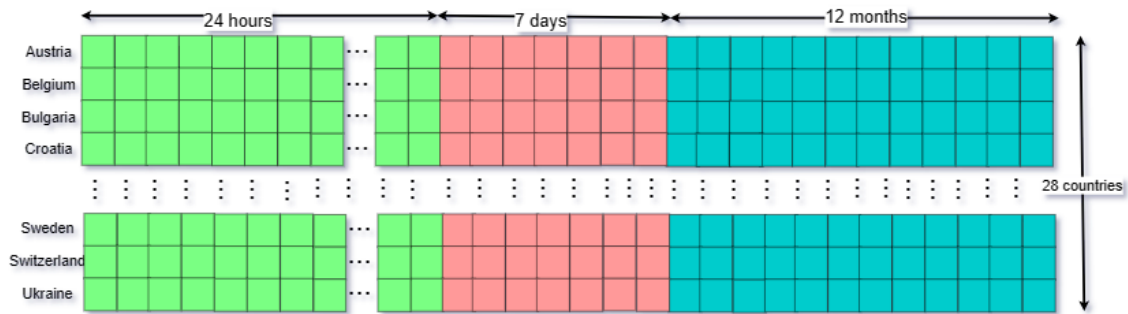
Η μέθοδος υπολογισμού των clusters είναι ο ίδιος για UTC και local. Αναλυτικά, δίνεται αρχικά ένα dataframe στο οποίο κάθε γραμμή περιλαμβάνει την στήλη:

- φορτίου (Load)
- χώρα (country)
- χρονική στιγμή/ώρα (hour) που πάρθηκαν τα δεδομένα
- μήνας (month) που πάρθηκαν τα δεδομένα
- έτος (year) που πάρθηκαν τα δεδομένα

Όπως φαίνεται και στο σχήμα 6.5, για κάθε ξεχωριστή χώρα του dataframe, και για κάθε μια από τις δοθείσες time covariates (λίστα με τις ονομασίες των ανάλογων στηλών), ομαδοποιώ τα δεδομένα με βάση το εκάστοτε ζεύγος (country, time_cov) και βρίσκεται η μέση τιμή του φορτίου. Η πράξη αυτή επιστρέφει ένα vector (time profile) με τις μέσες τιμές για κάθε χώρα:

- 24 τιμές για κάθε ώρα της μέρας (hour στήλη)
- 7 τιμές για κάθε μέρα της εβδομάδας (week στήλη)
- 12 τιμές για μήνα του χρόνου (month στήλη)

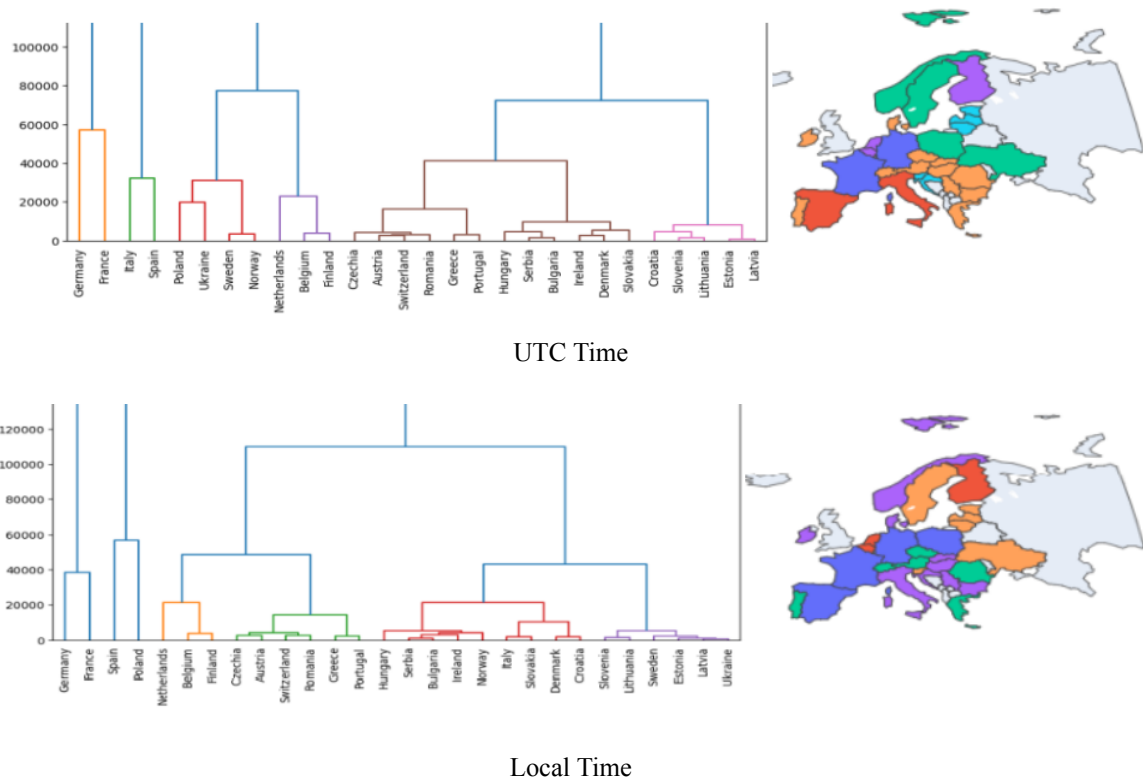
Στην συνέχεια, κανονικοποιούμε τις τιμές για κάθε time profile Έτσι, δημιουργείται ένα νέο dataframe χωρών, vectors όπου κάθε στήλη αντιπροσωπεύει μια χώρα και έχει $24 + 12 + 7 = 43$ τιμές. Το όνομα της στήλης είναι και το όνομα της χώρας. Το νέο dataframe δίνεται στην `scipy.pdist` που υπολογίζει την ευκλείδεια απόσταση των τιμών και επιστρέφει ένα distance matrix. Το `pdist` υπολογίζει by default με βάση τις γραμμές, και όχι τις στήλες. Άρα πριν δοθεί στην `pdist`, γίνεται `transpose` του dataframe ώστε κάθε γραμμή να αντιπροσωπεύει μια χώρα και όχι η στήλη. Στην συνέχεια εφαρμόζουμε hierarchical clustering, όπως στην περίπτωση των correlations, μέσω της linkage και δημιουργούμε το δενδρόγραμμα



Σχήμα 6.5: Σχέδιο που απεικονίζει την δομή του πίνακα των time profiles. Το συγκεκριμένο παράδειγμα καλύπτει την περίπτωση του μεγαλύτερου διανύσματος με ωριαίες, ημερήσιες και μηνιαίες τιμές

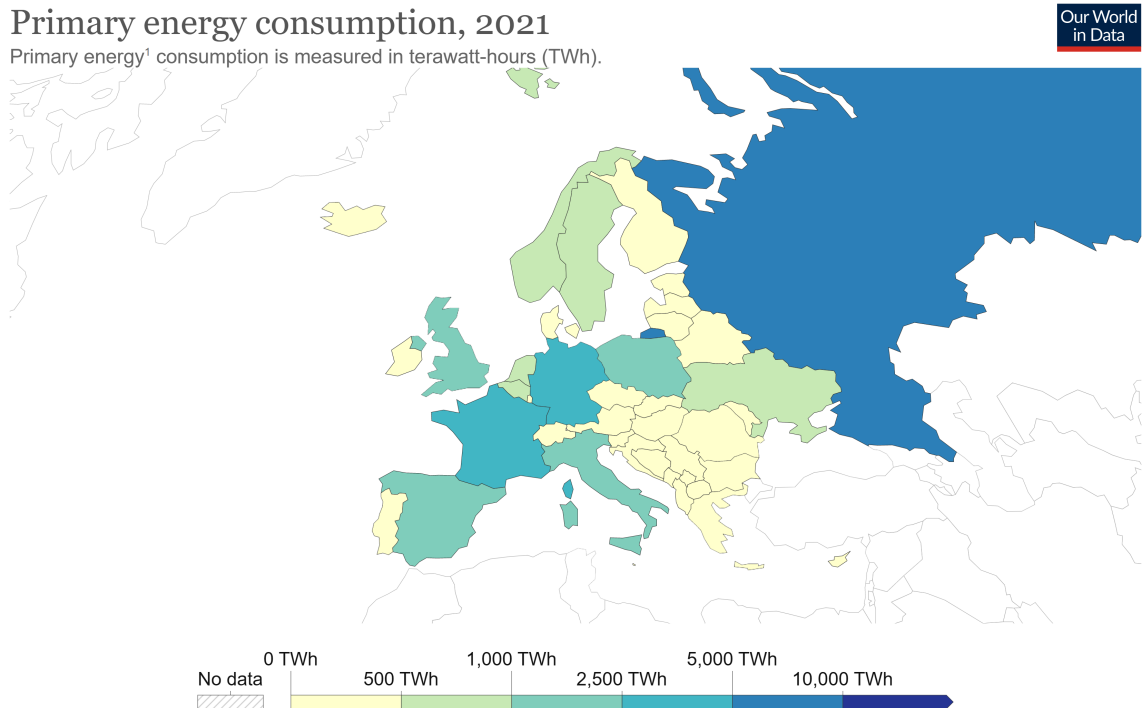
6.5.4 Έκδοχή 2.1: No normalization

Στην έκδοση του αλγορίθμου clustering, τα δεδομένα αρχικά δεν κανονικοποιήθηκαν. Εξετάστηκαν τα σύνολα hour-month, hour-year, year-month-hour. Στο σχήμα 6.6 ακολουθούν τα ζεύγη choropleth/dendrogram σε UTC και local time αντίστοιχα. Συγκρίνοντας τα ζεύγη choropleth/dendrogram για UTC και local time, παρατηρούνται αρκετές διαφορές μεταξύ τους. Στην περίπτωση του UTC time, οι βόρειες χώρες (Σκανδιναβικές + Βέλγιο/Ολλανδία) βρίσκονται είτε στο ίδιο είτε σε στενά συνδεδεμένο cluster και μπορούν να ενοποιηθούν σε ένα. Αντίθετα, σε local time, Ολλανδία/Βέλγιο/Φινλανδία συνδέονται με χώρες όπως Ελλάδα/Πορτογαλία, άλλα όχι με σκανδιναβικές. Μάλιστα οι σκανδιναβικές βρίσκονται όλες σε διαφορετικά clusters. Συνεπώς, υπάρχει μια ανεξήγητη ομαδοποίηση των χωρών, ειδικά στην περίπτωση του local time, καθώς δεν φαίνεται να συνδέονται με κάποιο τρόπο (πληθυσμιακά, γεωγραφικά, βιομηχανικά).



Σχήμα 6.6: Δενδρόγραμμα και σχετικό choropleth υπολογισμένο με βάση το daily profile των χωρών σε UTC και local time

Από το clustering σε UTC στο σχήμα 6.6, παρατηρούνται σημαντικές ομοιότητες με το χάρτη ζήτησης ηλεκτρικής ενέργειας του πανεπιστημίου της Οξφόρδης [81] (βλέπε και σχήμα 6.7). Έτσι, φαίνεται πως το παρών clustering δεν λαμβάνει υπόψη της αλλαγές που γίνονται με βάση τα time profiles, ομαδοποιώντας απλά με βάση το μέγεθος (magnitude) των τιμών φορτίου των χωρών. Αυτή υπήρξε η αφορμή για την επεξεργασία του αλγορίθμου clustering



1. **Primary energy:** Primary energy is the energy as it is available as resources – such as the fuels that are burnt in power plants – before it has been transformed. This relates to the coal before it has been burned, the uranium, or the barrels of oil. Primary energy includes energy that is needed by the end user, plus inefficiencies and energy that is lost when raw resources are transformed into a useable form. You can read more on the different ways of measuring energy in our article.

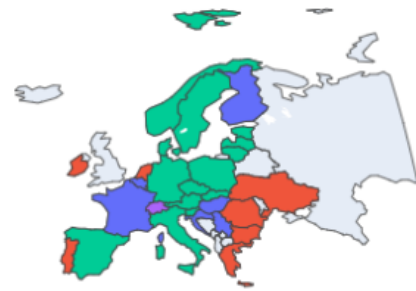
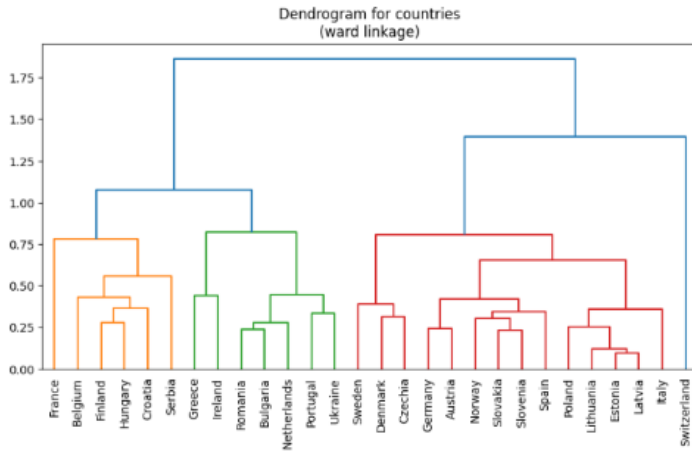
Σχήμα 6.7: Δενδρόγραμμα και σχετικό choropleth υπολογισμένο με βάση το correlation των χωρών σε local time

6.5.5 Εκδοχή 2.2: Normalized Time Profiles

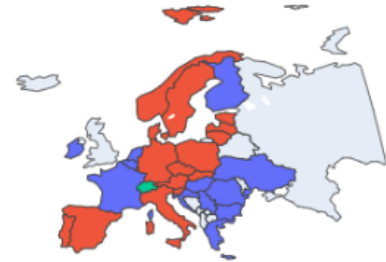
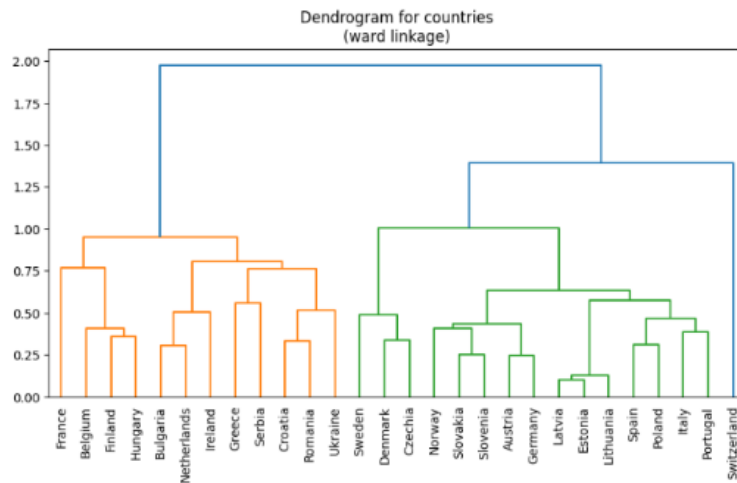
Σε αυτή την εκδοχή, όλες μέσες τιμές για κάθε ξεχωριστό time profile (πχ daily) μιας χώρας κανονικοποιούνται μεταξύ τους. Επίσης, δοκιμάστηκε η εισαγωγή ενός επιπλέον χαρακτηριστικού (feature) στους vectors της κάθε χώρας: την μέση τιμή φορτίου της εκάστοτε χώρας, κανονικοποιημένο προς την μέγιστη από τις μέσες τιμές των χωρών. Έτσι, δημιουργείται ένα νέο dataframe χωρών, vectors όπου κάθε χώρα αντιπροσωπεύεται από ένα ολικό προφίλ $24 + 12 + 7 + 1 = 44$ διαστάσεων. Το νέο feature αποδείχθηκε περιττό (δεν παρατηρήθηκε αλλαγή/βελτίωση με την προσθήκη του) το οποίο αποδόθηκε στο ότι το clustering γίνεται σε κανονικοποιημένα δεδομένα. Συνεπώς, για τα ακόλουθα πειράματα, επιλέχθηκε η αφαίρεση του. Για την περίπτωση αυτή, εξετάζουμε για daily (24 διαστάσεις), daily-weekly (24 + 7 = 31 διαστάσεις), daily-weekly-monthly (24 + 12 + 7 = 43 τιμές) profiles αποκλειστικά σε local timezone (λόγω της ομοιότητας του UTC με το energy demand)

Όπως φαίνεται στο σχήμα 6.8, με UTC χρόνο το daily profile των χωρών φαίνεται το clustering να επηρεάζεται έντονα από τις ζώνες ώρας. Αυτό γίνεται κατανοητό, με το ένα cluster (Πορτογαλία, Ιρλανδία) να είναι οι χώρες με UTC+0, και ένα άλλο να είναι επί το πλείστον οι χώρες με UTC+2 (Βουλγαρία, Ρουμανία, Βαλτικές κτλ) αφήνοντας τις υπόλοιπες χώρες με UTC+1 να καλύπτουν τα υπόλοιπα clusters (μωβ και κόκκινα στο choropleth). Αυτό ενθαρρύνει την εξέταση και ομαδοποίηση των χωρών με βάση το local time τους.

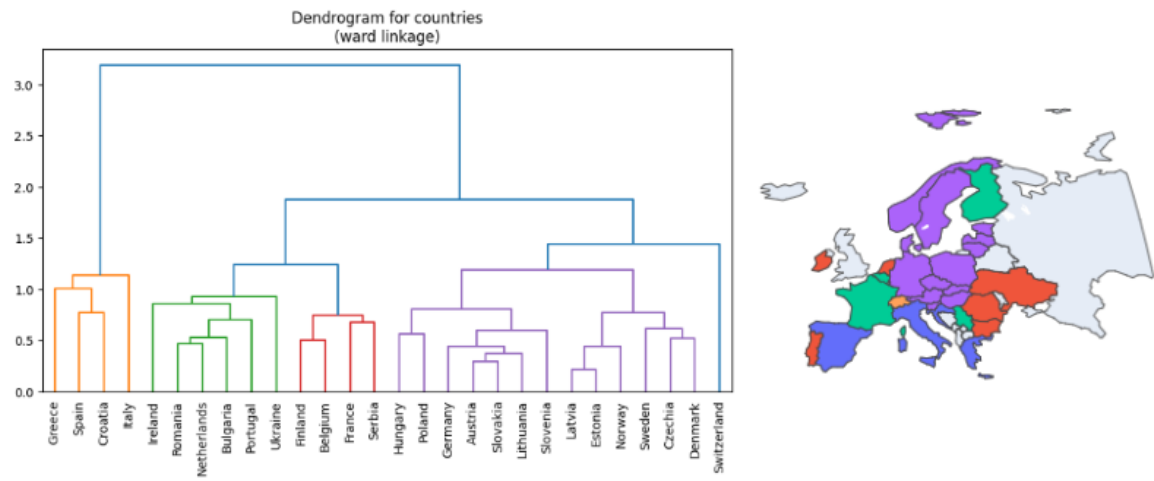
Απο το local time-based clustering στο σχήμα 6.7 με τον συνδυασμό daily, weekly και monthly profiles, προέκυψαν τα clusters των μεσογειακών χωρών (Ελλάδα, Ιταλία, Ισπανία, Κροατία), των σλαβικών χωρών, καθώς και της κεντρικής και βόρειας Ευρώπης. Οι Σκανδιναβικές Χώρες (Σουηδία, Νορβηγία, Δανία) αν και στο ίδιο cluster με αυτές της κεντρικής Ευρώπης, φαίνεται από το δενδρόγραμμα ότι ομαδοποιούνται ισχυρά μεταξύ τους. Συνεπώς παρατηρείται μια ομαδοποίηση με βάση γεωγραφικά ή και πολιτισμικά κριτήρια με τις ανάλογα κοινές χώρες να βρίσκονται στις ίδιες ομάδες.



Daily profile in UTC time

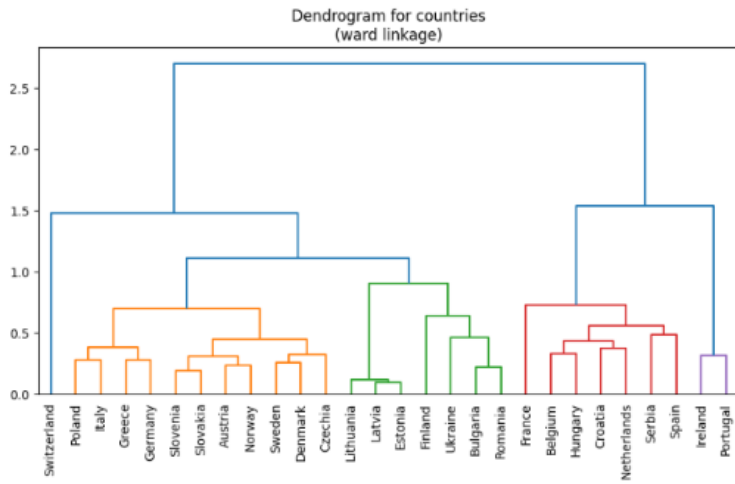


Daily and weekly profile in local time

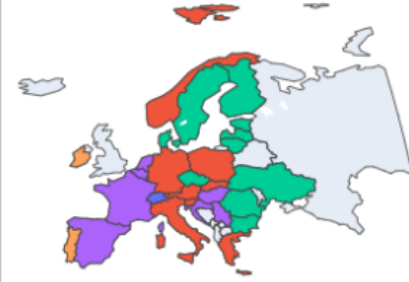
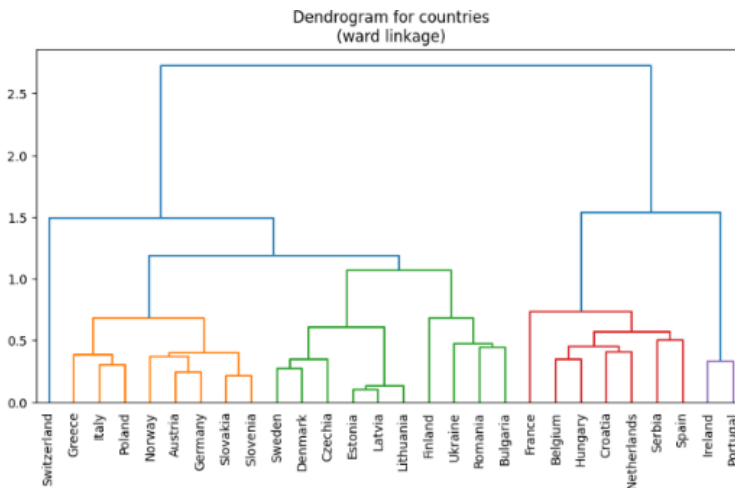


Daily, weekly and monthly profile in local time

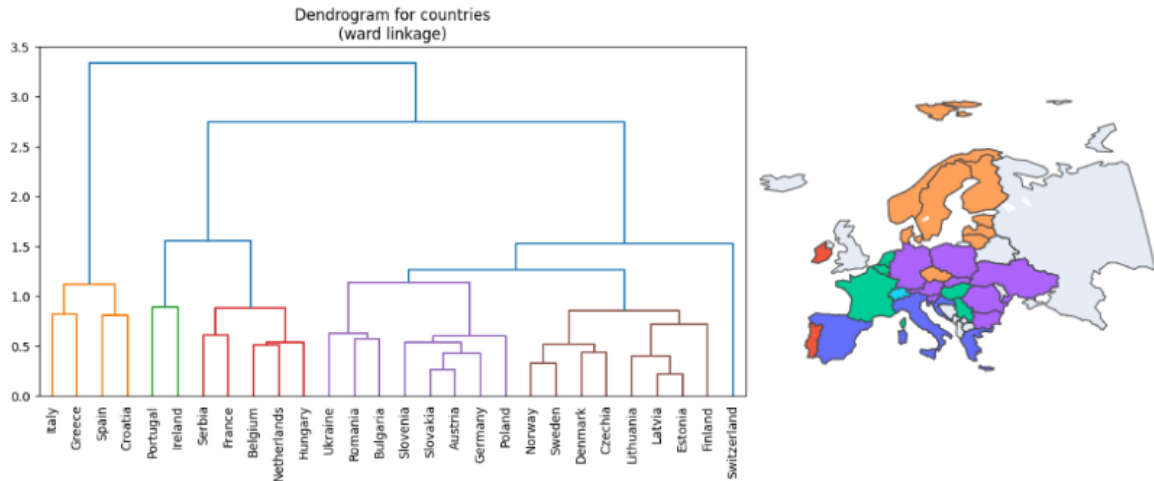
Σχήμα 6.7: Δενδρογράμματα και σχετικά choropleths υπολογισμένα με βάση time profiles των χωρών σε local time



Daily profile in UTC time



Daily and weekly profile in UTC time



Daily, weekly and monthly profile in UTC time

Σχήμα 6.7: Δενδρογράμματα και σχετικά choropleths υπολογισμένα με βάση time profiles των χωρών σε UTC time

6.6 Heatmaps

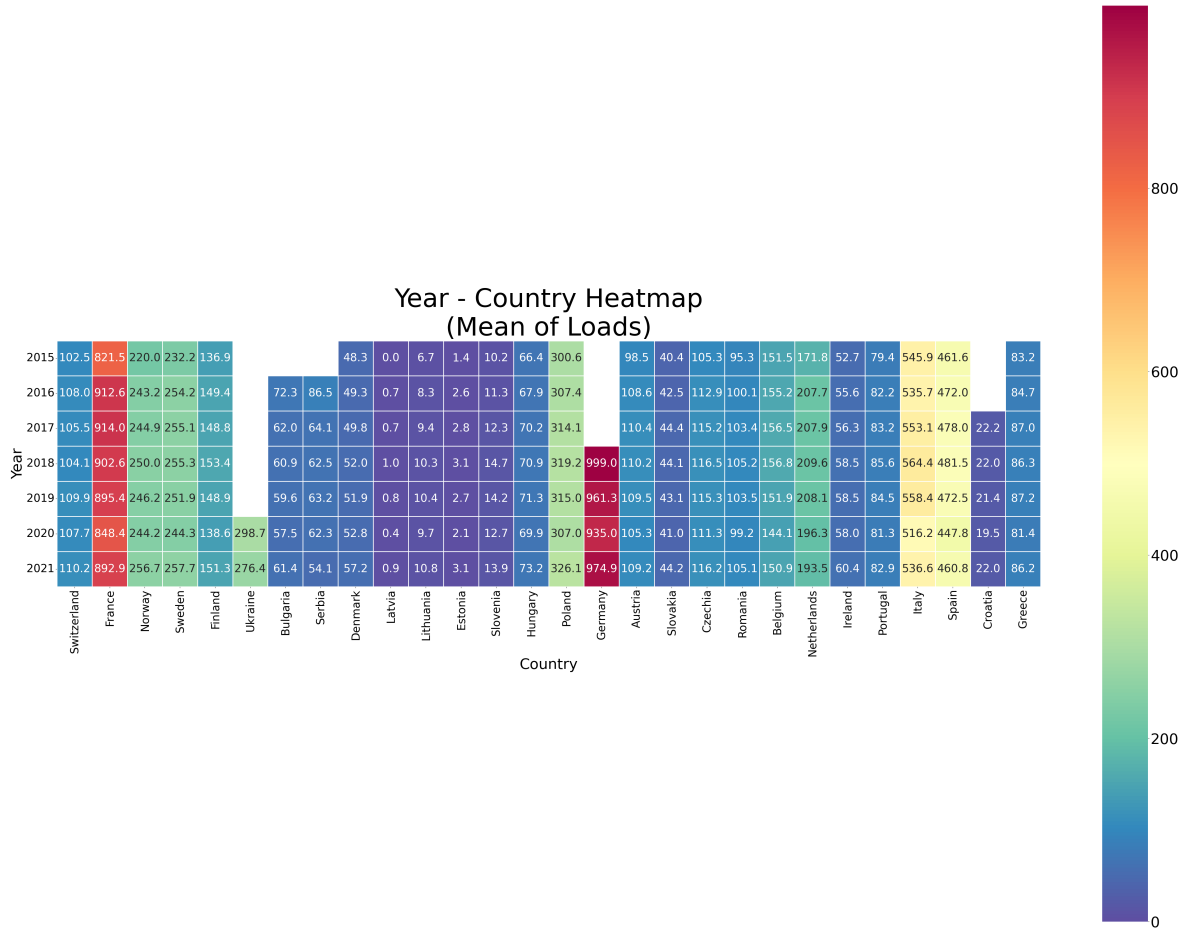
Ως εξέλιξη των mean plots στο κεφάλαιο 6.1, κατασκευάστηκαν γραφήματα για να αποτυπώσουν την συσχέτιση των φορτίων των χωρών με βάση ορισμένα time profiles (daily, monthly, yearly). Η καλύτερη απεικόνιση για αυτό το σκοπό αποδείχτηκε αυτή των *heatmaps*: μια τεχνική απεικόνισης δεδομένων που δείχνει το μέγεθος (magnitude) ενός φαινομένου ως χρώμα σε δύο διαστάσεις.

Το μέγεθος του φορτίου εξετάστηκε με διαστάσεις την χώρα προέλευσης και την τιμή του εκάστοτε time profile. Για κάθε time profile, δημιουργήθηκαν δύο ξεχωριστά heatmaps. Και τα δύο αποτυπώνουν την μέση τιμή του φορτίου μιας χώρας για μια τιμή του time profile (πχ για το 2019 στο yearly profile) συγκριτικά με όλες τις τιμές στις οποίες κυμαίνονται τα φορτία. Όμως, η ειδοποιός διαφορά ανάμεσα στα δύο βρίσκεται στο γεγονός πως η πρώτη περίπτωση ο μέσος όρος γίνεται με βάση τις τιμές όλων των χωρών, ενώ στην δεύτερη περίπτωση γίνεται με βάση την ίδια χώρα.

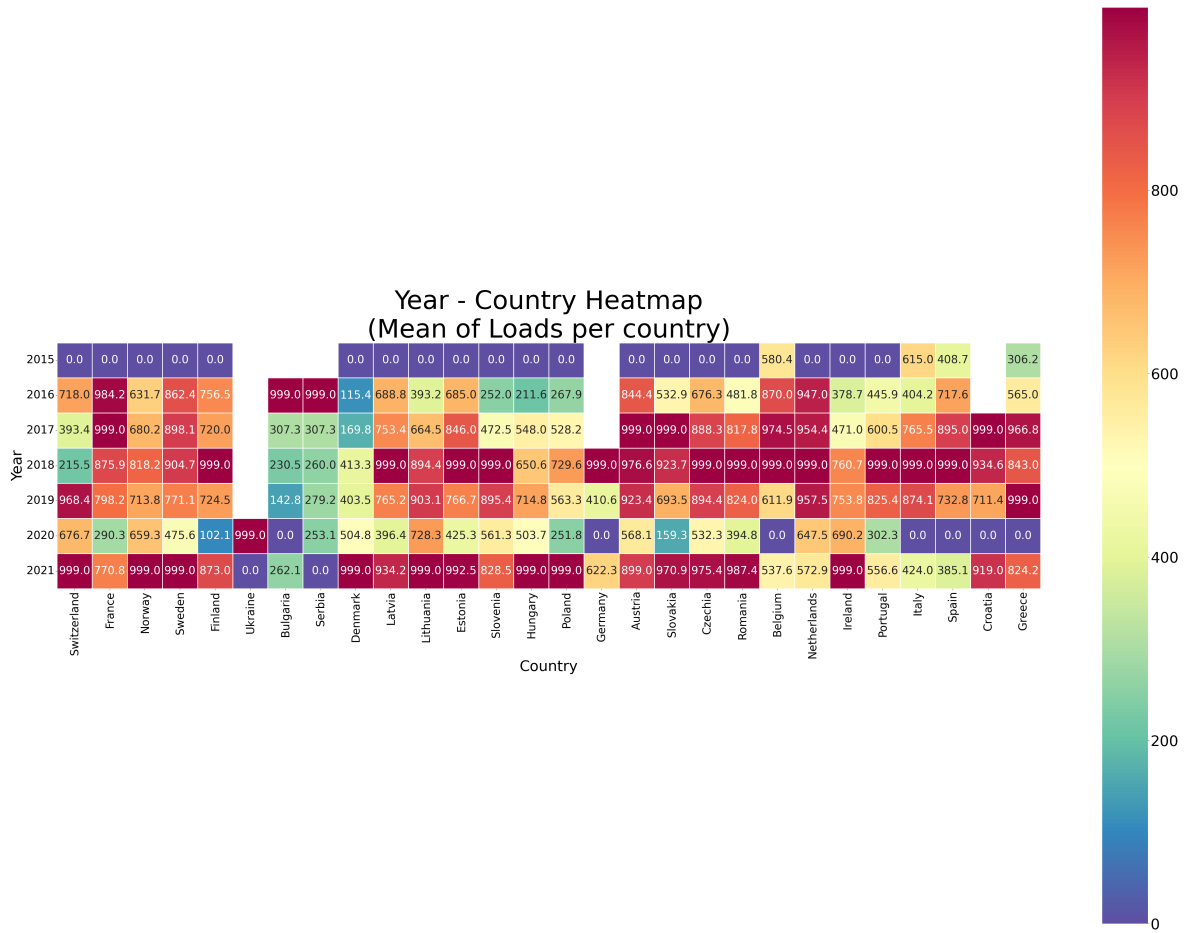
Έτσι, έχουμε μια οπτική των τιμών φορτίου ανά χώρα συγκριτικά με τις υπόλοιπες χώρες που παρέχονται, αλλά και μια εικόνα της διακύμανσης του φορτίου σε μια χώρα, με το πέρασμα των ετών/μηνών/ωρών

Από τους παραγόμενους χάρτες θερμότητας, μπορούμε να συμπεράνουμε ότι:

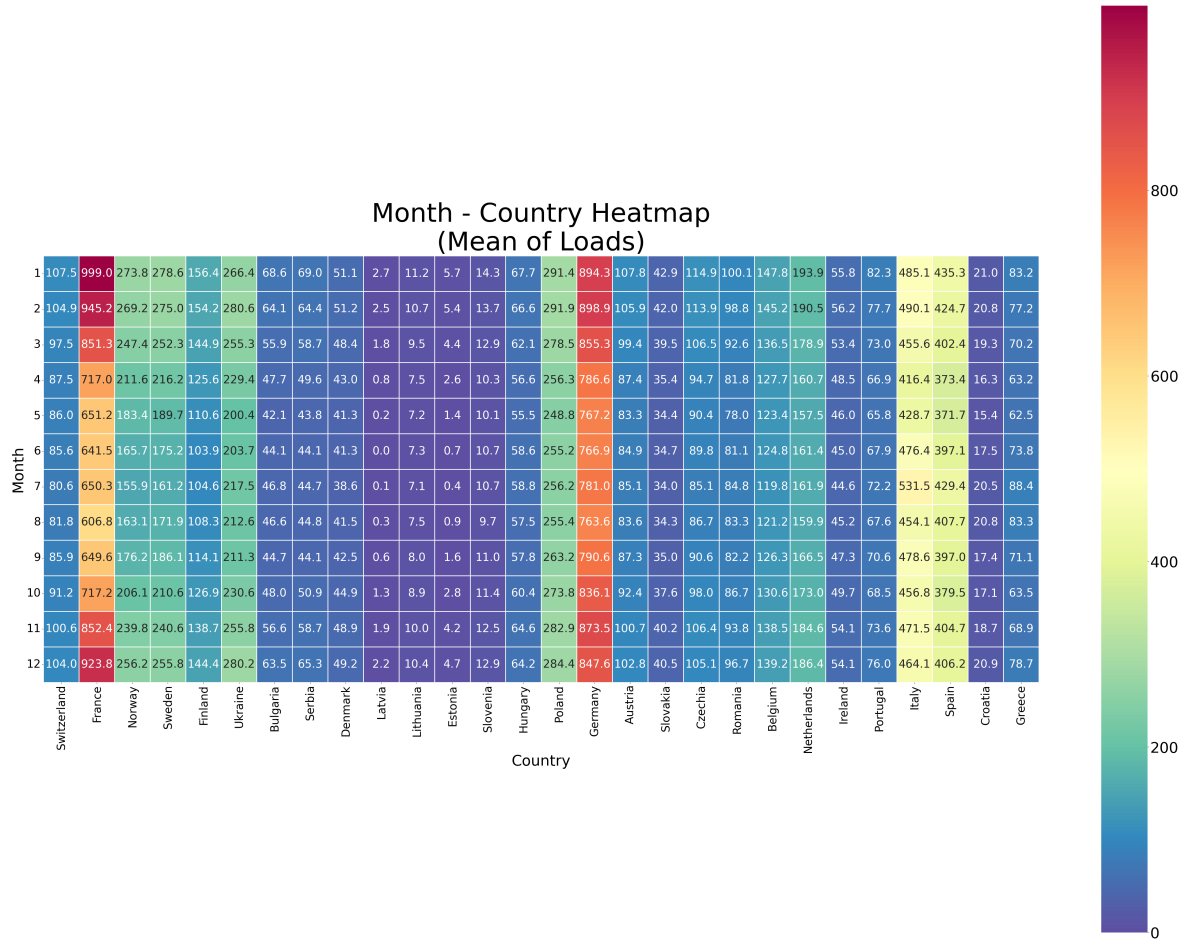
- **ετήσια:** ήταν σαφές ότι έλειπαν δεδομένα από συγκεκριμένες χώρες. Σε αυτό το heatmap, γίνεται πιο συγκεκριμένο για ποια έτη έχουμε έλλειψη και από ποια σύνολα δεδομένων (χώρες). Επίσης, παρατηρούμε (ανά στήλη και στον χάρτη θερμότητας στο σύνολό του) ότι οι μεγαλύτερες τιμές υπάρχουν κυρίως στο κάτω μέρος του χάρτη θερμότητας, γεγονός που σημαίνει ότι η ζήτηση ενέργειας αυξάνεται καθώς φτάνουμε στην παρούσα χρονική στιγμή. Είναι λογικό, ότι οι χώρες με μεγαλύτερο πληθυσμό και παραγωγή θα "κυριαρχήσουν" στον χάρτη θερμότητας με τις μεγαλύτερες τιμές σε όλα τα έτη (Γαλλία, Γερμανία κ.λπ.)
- **μηνιαία:** υπάρχει γενικά τεράστια ζήτηση ενέργειας κατά τους πρώτους (Ιανουάριος-Μάρτιος) και τους τελευταίους (Οκτώβριος-Δεκέμβριος) μήνες κάθε έτους, οι οποίοι συμπίπτουν σε μεγάλο βαθμό με τους μήνες με τις χαμηλότερες θερμοκρασίες και επομένως με μεγαλύτερη ανάγκη για ενέργεια μέσα στα σπίτια και τις επιχειρήσεις.
- **καθημερινά:** το μεγαλύτερο μέρος της ζήτησης ενέργειας παρατηρείται τις εργάσιμες ημέρες/ώρες, κάτι φυσικά αναμενόμενο δεδομένων των mean plots του κεφαλαίου 6.1
- ομόφωνα από όλα τα heatmaps, βλέπουμε κοινές τιμές σε γεωγραφικά γειτονικές χώρες. Χαρακτηριστικά παραδείγματα ομάδων είναι οι σκανδιναβικές, οι βαλτικές, και οι σλαβικές χώρες



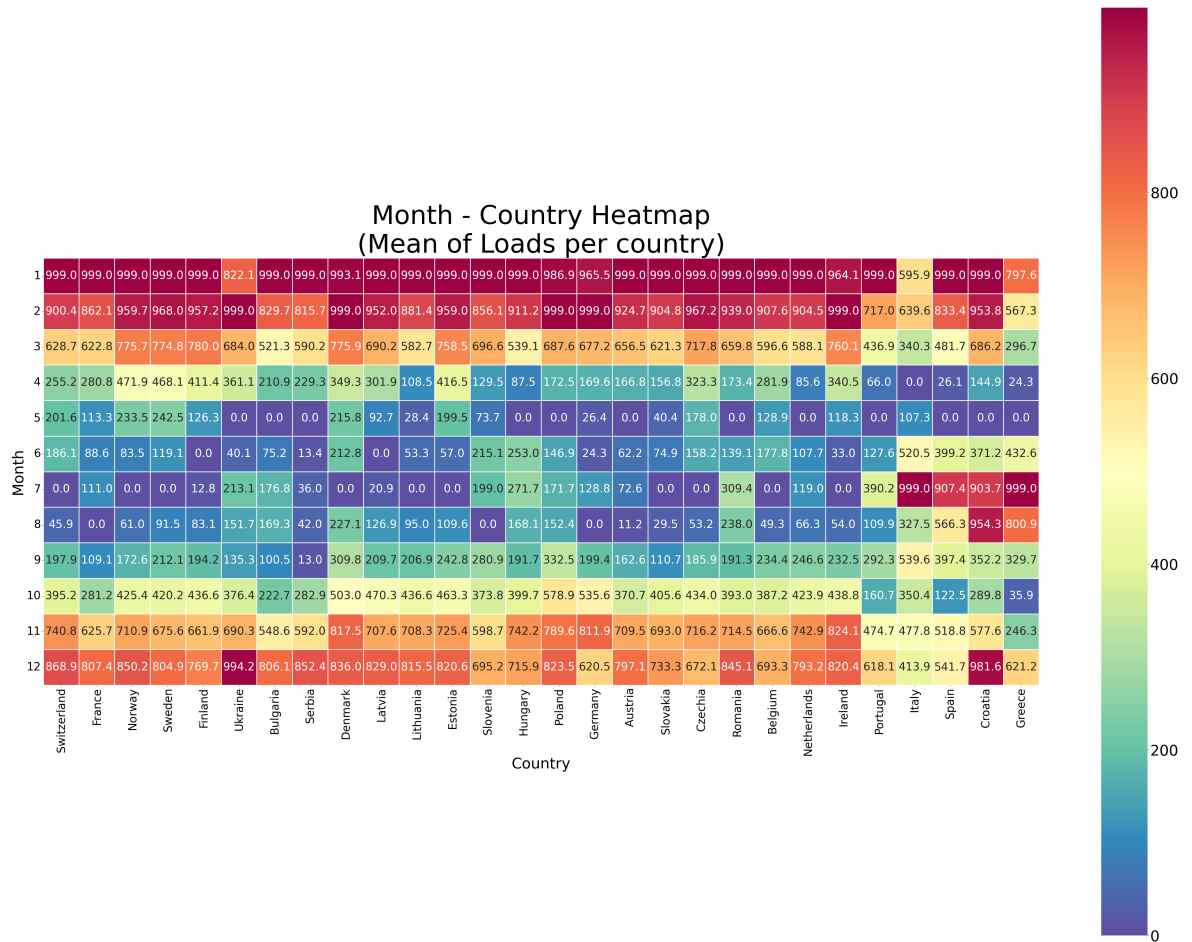
Yearly total mean heatmap



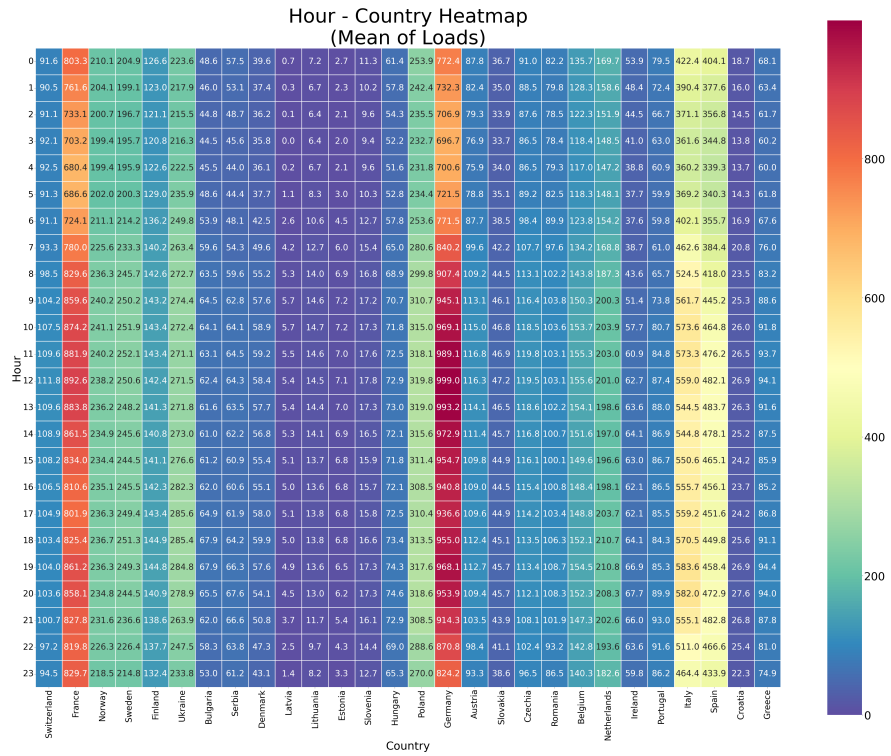
Yearly country-wise mean heatmap



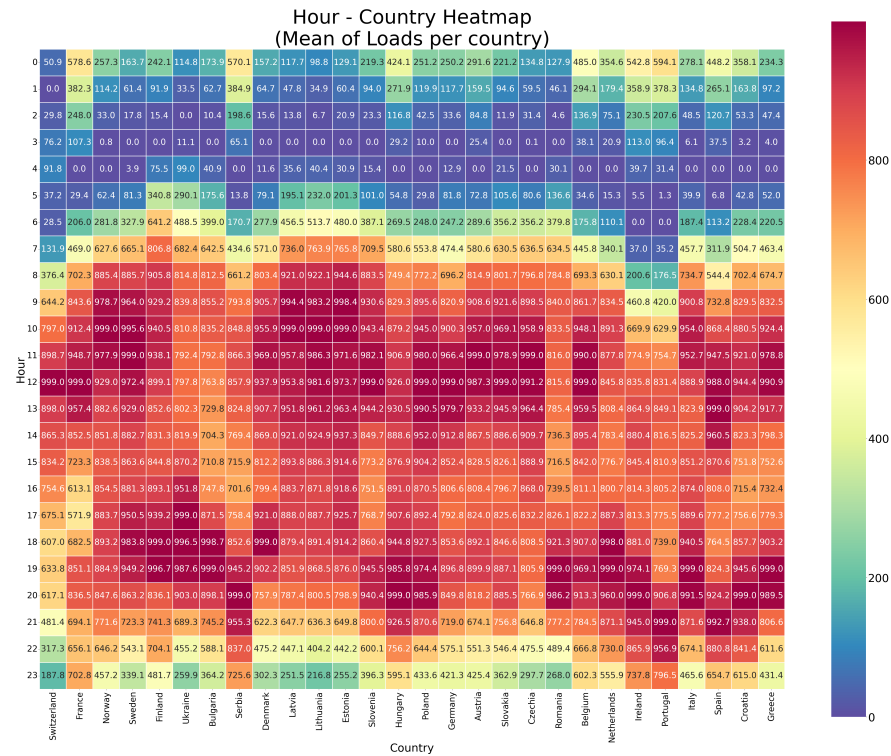
Monthly total mean heatmap



Monthly total mean heatmap per country



Daily total mean heatmap



Daily country-wise mean heatmap

Σχήμα 6.7: Heatmaps που απεικονίζουν την σχέση των μέσων τιμών του φορτίου συγκριτικά α) με τις υπόλοιπες χώρες και β) με τον χρόνο

6.7 Annual lineplots & historgams

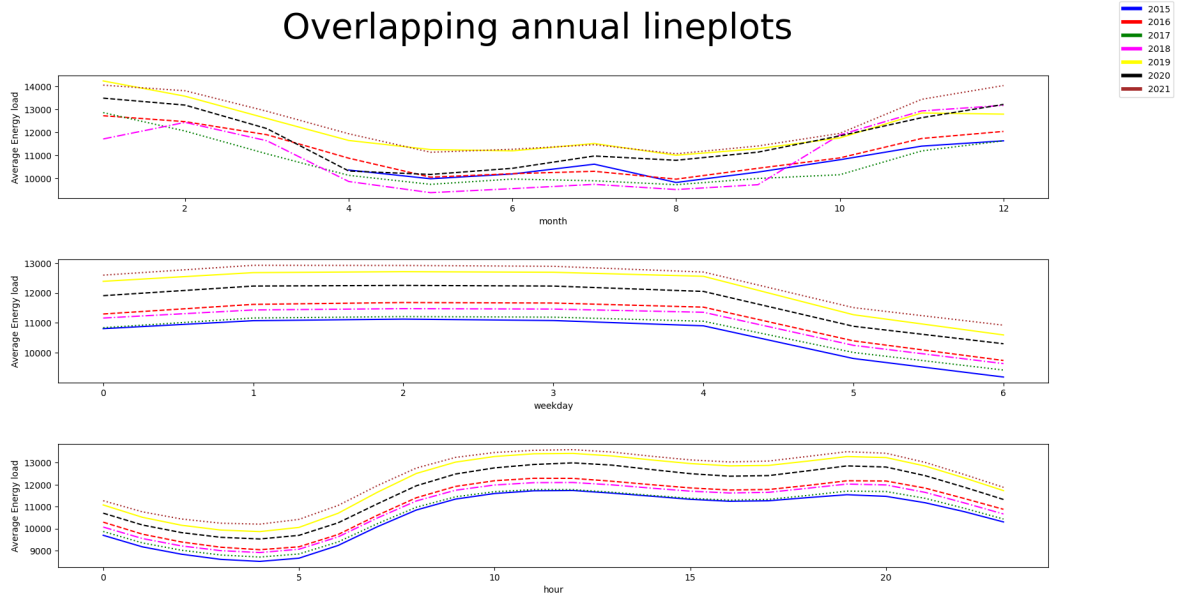
Στο σχήμα 6.8 σχεδιάστηκαν γραμμικές αποτυπώσεις των φορτίων όλων των χωρών, όπου κάθε γραμμή είναι ένα διαφορετικό έτος. Σκοπός αυτής της απεικόνισης είναι να αναδειχθεί πώς οι τιμές μετατοπίζονται κατά μέσο όρο κάθε χρόνο με βάση την εβδομάδα ή ημέρα ή ώρα του έτους. Στον άξονα x, εμφανίζουμε τον χρόνο ημερομηνίας (52/6/24 τιμές για τις εβδομάδες ή ημέρες ή ώρες του έτους αντίστοιχα). Στον άξονα y, είναι η μέση τιμή ενός συγκεκριμένου άξονα x για το συγκεκριμένο έτος

Ενώ δεν είναι τόσο εμφανές στο διάγραμμα των εβδομάδων έτους (yearweek), στα διαγράμματα των εβδομάδων/ωρών υπάρχει σαφής άνοδος ολόκληρων των γραμμών με κάθε έτος, που σημαίνει σταθερή αύξηση του μέσου ενεργειακού φορτίου ανά έτος. Σημαντική εξαίρεση στον κανόνα αυτό αποτελεί το έτος 2020, το οποίο είναι χαμηλότερο από το 2019-2021. Μια λογική εξήγηση να είναι η συμπεριφορά και οι απαιτήσεις για ενέργεια κατά τη διάρκεια της πανδημίας του covid-19 εκείνη την περίοδο.

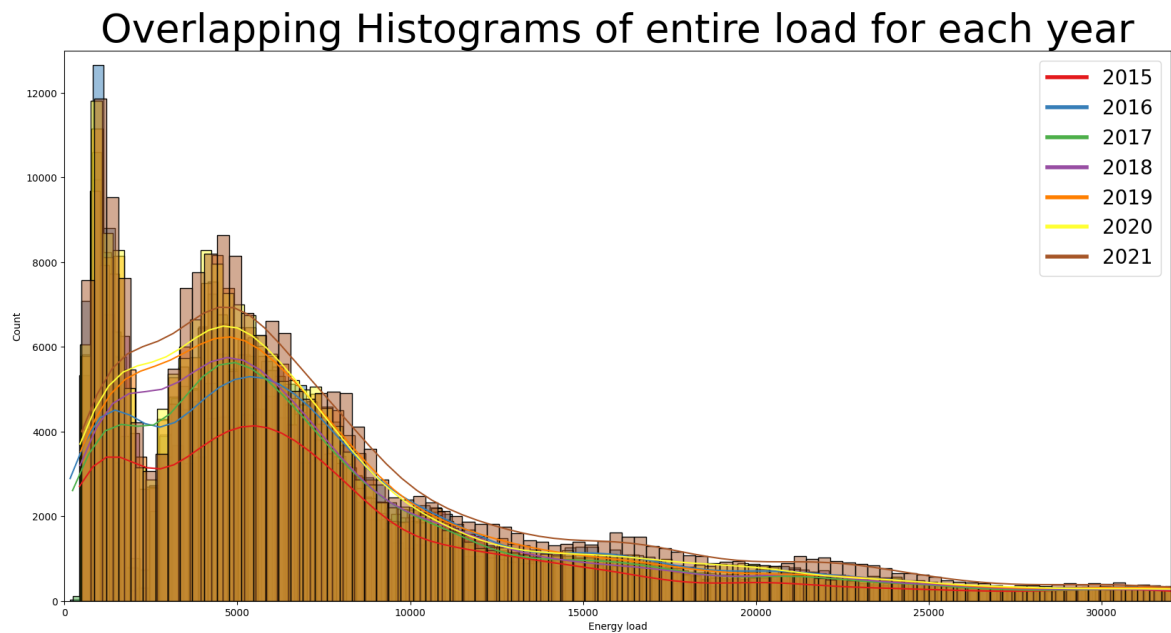
Τέλος, στο διάγραμμα της εβδομάδας του έτους, μπορούμε να δούμε ότι το διάγραμμα γραμμής για το 2022 ακολουθεί γραμμική προσέγγιση σε αντίθεση με όλα τα άλλα, γεγονός που μπορεί να αποδοθεί στον υπολογισμό (και κυρίως στην έλλειψη) δεδομένων κατά τη διάρκεια του έτους αυτού.

Η ίδια συμπεριφορά, γίνεται επίσης ξεκάθαρη με την χρήση ιστογραμματος. Τα ιστογράμματα μπορούν να εξηγηθούν ως αναπαραστάσεις δεδομένων που μοιάζουν με ραβδογράμματα και κατατάσσουν ένα εύρος κλάσεων σε στήλες κατά μήκος του οριζόντιου άξονα x. μια αναπαράσταση δεδομένων που μοιάζει με ραβδογράμματα και κατατάσσει ένα εύρος κλάσεων σε στήλες κατά μήκος του οριζόντιου άξονα x. Ο κατακόρυφος άξονας y αναπαριστά τον αριθμό καταμέτρησης ή το ποσοστό εμφάνισης στα δεδομένα για κάθε στήλη. Οι στήλες μπορούν να χρησιμοποιηθούν για την οπτικοποίηση μοτίβων κατανομής δεδομένων

Έτσι, δημιουργήθηκε το σχήμα 6.8 στο οποίο αναπαρίσταται το dataset με την μορφή αποκαλυπτομένων ιστογραμμάτων, με κάθε ιστόγραμμα να αντιπροσωπεύει ένα ξεχωριστό έτος από το οποίο υπάρχουν δεδομένα.



Γραμμική απεικόνιση των δεδομένων του dataset ανά έτος



Απεικόνιση συνολικού dataset σε μορφή ιστογράμματος ανά χρόνο

Σχήμα 6.8: Απεικονίσεις των δεδομένων του συνολικού dataset σε μορφή γραμμικού διαγράμματος και ιστογράμματος αντίστοιχα

Κεφάλαιο 7

Πειραματική Περιγραφή

Σκοπός των πειραμάτων είναι να εξεταστεί η απόδοση των τεχνικών transfer learning ανάμεσα σε δεδομένα διαφόρων χωρών. Η πειραματική διαδικασία ακολουθεί τα στάδια του pipeline όπως ορίστηκαν στο κεφάλαιο 5. Σε αυτό το κεφάλαιο, αναλύονται αρχικά στην φύση των δεδομένων που χρησιμοποιήθηκαν στα πειράματα. Είναι σημαντικό να αναφερθεί η δομή τους, και η πηγή τους τόσο για την καλύτερη κατανόηση, όσο και την αξιοπιστία των προβλέψεων. Έπειτα, αναφέρονται η διάταξη των πειραμάτων (test cases) για την εξαγωγή συμπερασμάτων.

7.1 Δεδομένα

Τα δεδομένα πάρθηκαν δεδομένα από την ENTSO-E: το Ευρωπαϊκό Δίκτυο Συστημάτων Μεταφοράς Διαχειριστών Ηλεκτρικής Ενέργειας [30]. Αποτελεί μια ένωση για τη συνεργασία των ευρωπαϊκών διαχειριστών συστημάτων μεταφοράς (ΔΣΜ). Το 39 μέλη ΔΣΜ, που εκπροσωπούν 35 χώρες, είναι υπεύθυνοι για την ασφαλή και συντονισμένη λειτουργία του ευρωπαϊκού συστήματος ηλεκτρικής ενέργειας, του μεγαλύτερου διασυνδεδεμένου ηλεκτρικού δικτύου στην στον κόσμο.

Τα δεδομένα είναι αρχικά αποθηκευμένα σε CSV (comma-separated values) αρχεία, με κάθε αρχείο να ένα μοναδικό συνδιασμό χώρας, ζώνη ώρας και bidding zone. Bidding zone ονομάζεται η μεγαλύτερη γεωγραφική περιοχή εντός της οποίας οι συμμετέχοντες στην αγορά μπορούν να ανταλλάσσουν ενέργεια χωρίς κατανομή δυναμικότητας (άρθρο 2 παράγραφος 3 του κανονισμού 543/2013, της 14ης Ιουνίου 2013, σχετικά με την υποβολή και τη δημοσίευση δεδομένων στις αγορές ηλεκτρικής ενέργειας και άρθρο 2 παράγραφος 65 του

κανονισμού (ΕΕ) 2019/943 του Ευρωπαϊκού Κοινοβουλίου και του Συμβουλίου, της 5ης Ιουνίου 2019, για την εσωτερική αγορά ηλεκτρικής ενέργειας (κανονισμός για την αγορά ηλεκτρικής ενέργειας)).

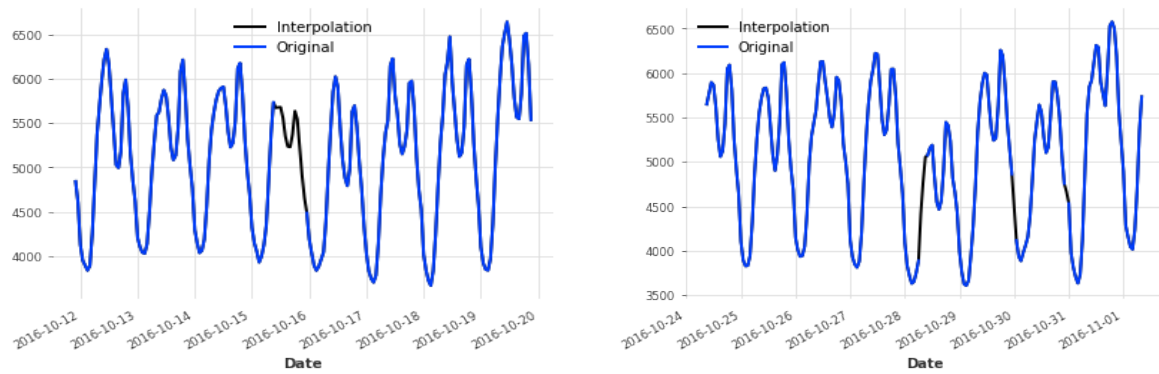
Τα αρχεία περιλαμβάνουν τρεις στήλες: η πρώτη αντιπροσωπεύει την χρονική στιγμή (σε μορφή Year-Month-Day Hour:Minute:Second) σε UTC(+0) ζώνη ώρας, το φορτίο που Load εκείνης της στιγμής και η τιμή (Price) του εν λόγω φορτίου. Τα δεδομένα περιλαμβάνουν ωριαίες τιμές του φορτίου των χωρών από το 2015 έως και το 2022. Για τους σκοπούς της διπλωματικής, μας αφορά αποκλειστικά η κατανάλωση της ηλεκτρικής ενέργειας, και άρα το κόστος αγνοείται.

7.2 Επεξεργασία

Τα δεδομένα των χωρών οφείλουν να περάσουν αρκετά στάδια επεξεργασίας πριν να είναι σε θέση να θεωρηθούν κατάλληλες χρονοσειρές για χρήση ως είσοδος από τα μοντέλα μηχανικής μάθησης. Έχοντας ήδη γίνει, στο κεφάλαιο 5.2.2, μια εξονυχιστική ανάλυση της όλης διαδικασίας, οφείλουμε εδώ να αναλύσουμε παραδοχές και λεπτομέρειες που έγιναν με βάση την φύση των πειραμάτων

Έπειτα από την ανάλυση που έγινε στα δεδομένα στο κεφάλαιο 6, αποδείχθηκε ότι στο local time-based clustering υπάρχει υψηλή συσχέτιση μεταξύ χωρών κοινής γεωγραφίας ή κουλτούρας. Έτσι, η μελέτη συνεχίστηκε προσαρμόζοντας τις τιμές του dataset για την τοπική ώρα των δεδομένων. Έτσι, θα εξετάσουμε το ενδεχόμενο το μοντέλο να μπορεί να αναγνωρίσει αυτή την συσχέτιση ώστε να πετύχει καλύτερη απόδοση.

Μια παραδοχή που γίνεται είναι στην daylight saving time, στην οποία την άνοιξη οι χώρες αλλάζουν την ώρα σε μια ώρα μπροστά, και το φθινόπωρο μια ώρα πίσω με σκοπό την μεγιστοποίηση φως ηλίου. Όταν μετατρέπουμε την ώρα μιας χώρας από UTC (που δεν έχει DST) σε local time (που έχει DST) δημιουργείται μια ανύπαρκτη ώρα: δεν υπάρχει σε μια συγκεκριμένη ζώνη ώρας όπου τα ρολόγια μετακινήθηκαν προς τα εμπρός λόγω της θερινής ώρας. Για αυτές τις τιμές επιλέχθηκε να γίνει 'shift_forward', το οποίο θα μετατοπίσει την ανύπαρκτη ώρα προς τα εμπρός στην πλησιέστερη υπάρχουσα ώρα.



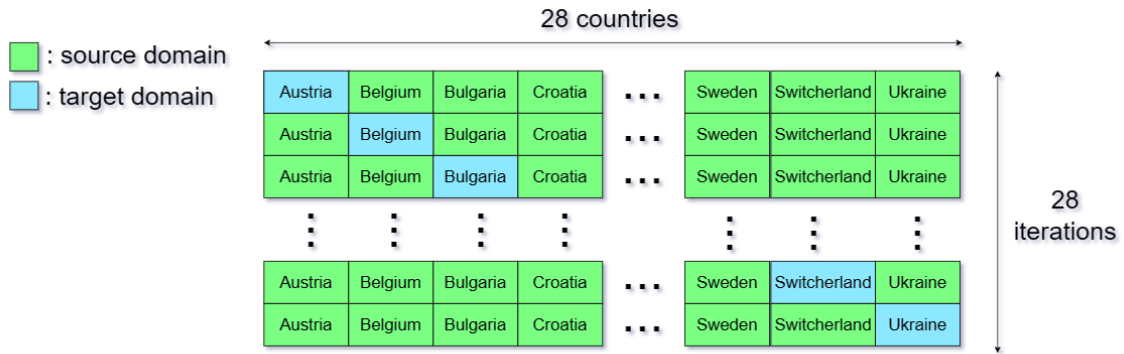
Σχήμα 7.1: Απεικόνιση δύο ανεξάρτητων χρονικών περιόδων στην χρονοσειρά της Ελλάδας στην οποία τα δεδομένα πραγματοποιήθηκε imputation. Με μαύρη γραμμή φαίνεται οι τιμές που έγιναν imputed και με μπλε η αρχική τιμή της χρονοσειράς

Για την επιλογή των source και target domains, δημιουργήθηκαν τρία ανεξάρτητα case studies, το καθένα να εξυπηρετεί στην εξαγωγή διαφορετικών συμπερασμάτων

7.3 Case studies

Case 1: Benchmark: Εξετάζουμε κάθε χώρα του dataset εξατομικευμένα. Υπάρχει ένα μοντέλο για κάθε χώρα, το οποίο εκπαιδεύεται με ιστορικά δεδομένα της εκάστοτε χώρας για την εξαγωγή μελλοντικών τιμών. Έτσι δημιουργούμε ένα σημείο αναφοράς (baseline) για να εξετάσουμε την απόδοση του transfer learning στα datasets ανά χώρα. Έτσι, έχουμε 28 διαφορετικά πειράματα, ένα για κάθε χώρα του dataset. Εξετάζουμε την απόδοση των προβλέψεων κάθε μοντέλου, εκπαιδευμένου με τα δεδομένα της εκάστοτε χώρας, στα ίδια δεδομένα.

Case 2: All-TL: Αποτελεί την πρώτη εφαρμογή transfer learning. Εξετάζουμε την επίδοση του μοντέλου πρόβλεψης δεδομένων μιας χώρας όταν εκπαιδεύεται με τα δεδομένα όλων των υπόλοιπων χωρών. Όπως φαίνεται και στο σχήμα 7.2, το dataset μας περιέχει δεδομένα από 28 διαφορετικές χώρες. Το test case αυτό εκτελεί 28 πειράματα στα οποία διαλέγεται μια διαφορετική χώρα σαν target domain και όλες οι υπόλοιπες σαν source domain. Στην συνέχεια, δημιουργείται ένα μοντέλο, εκπαιδευμένο στο source domain, και οι παράμετροι του χρησιμοποιούνται (μέσω τεχνικών transfer learning) για την κατασκευή ενός νέου μοντέλου με σκοπό την πρόβλεψη δεδομένων του target domain



Σχήμα 7.2: Απεικόνιση της χρήσης των χωρών για την υλοποίηση του 1ου testcase. Κάθε γραμμή αντιπροσωπεύει μια ξεχωριστή εκτέλεση του αλγορίθμου. Τα δεδομένα εκπαίδευσης (και οι ανάλογες χώρες) είναι με πράσινο χρώμα και η χώρα για την οποία προβλέπονται δεδομένα είναι με μπλε.

Case 3: Cluster-TL: Σύμφωνα με το συμπέρασμα του κεφάλαιο 6, χώρες με κοινή γεωγραφία ή και κουλτούρα, έχουν γειτονική συμπεριφορά στη ζήτηση ηλεκτρικής ενέργειας τους. Με βάση αυτό, δημιουργήθηκε ένα testcase το οποίο εξετάζει την επίδοση της μεταφοράς μάθησης μεταξύ χωρών της ίδιας ομάδας. Σε κάθε εκτέλεση, δημιουργείται ένα μοντέλο στο οποίο προβλέπεται η ζήτηση της εκάστοτε χώρας, έχοντας όμως εκπαιδευτεί μόνο με τα δεδομένα των υπόλοιπων χωρών της ίδιας ομάδας (cluster). Εφαρμόζουμε παρόμοια πολιτική με το προηγούμενο case study, κατασκευάζοντας 28 πειράματα (ένα για κάθε χώρα του dataset) όπου στο καθένα μια διαφορετική χώρα αποτελεί το target domain. Αντί όμως να χρησιμοποιηθούν όλες οι χώρες του dataset ως source domain, επιλέγουμε μόνο τις χώρες που βρίσκονται στο ίδιο cluster με την εκάστοτε χώρα του target domain

7.4 Αρχιτεκτονική & εκπαίδευση

Ο πίνακας 7.1 παρουσιάζει τις παραμέτρους που χρησιμοποιούνται από το μοντέλο, είτε το εύρος τιμών που εξετάζονται για το tuning τους (optuna). Κατά την πλήρη επανεκπαίδευση του δικτύου, όπως αναφέρεται στο κεφάλαιο 5.2.3 γίνεται χρήση EarlyStopping. Οι παράμετροι *check_val_every_n_epoch* και *patience* του πίνακα 7.1 αφορούν αντίστοιχα το πλήθος εποχών ανά το οποίο εκτελεί validation στα δεδομένα και το μέγιστο πλήθος εποχών που μπορεί να μην υπάρχει βελτίωση του loss function πριν διακόψει την εκπαίδευση

Η MLP αρχιτεκτονική που χρησιμοποιούν τα μοντέλα, απαιτεί ένα προκαθορισμένο μέγεθος εισόδου και εξόδου του μοντέλου. Έτσι, τα δεδομένα προσαρμόζονται κατάλληλα προκειμένου να χρησιμοποιηθούν σαν είσοδο για το μοντέλο (βλέπε κεφάλαιο 5.2.4) και μετά κατασκευάζεται το ίδιο το μοντέλο. Η κατασκευή του εξαρτάται άρικτα από την χρήση ή μη transfer learning. Στην περίπτωση που δεν χρησιμοποιείται, κατασκευάζεται μοντέλο με βάση της υπερπαραμέτρους της optuna (αν έχει γίνει) ή της καθορισμένες από τον χρήστη στην είσοδο. Αντίθετα, όταν γίνεται χρήση transfer learning, αναζητείται στο περιβάλλον του *Mlflow* η εκτέλεση του αντίστοιχου source μοντέλου και χρησιμοποιείται σαν βάση για το μοντέλο στο target domain.

Arguments	Value Range	Type
n_trials	100	Fixed
max_epochs	1500	Fixed
n_layers	[2,6]	Integer Continuous
layer_sizes	128,256,512,1024,2048	Discrete
l_window	168,336,504,672	Discrete
f_horizon	24	Fixed
l_rate	$[1e^{-5}, 1e^{-4}]$	Float Continuous
activation	ReLU, SiLU	Discrete
optimizer_name	Adam	Discrete
batch_size	256,512,1024	Discrete
n_estimators	20	Fixed
num_workers	10	Discrete
check_val_every_n_epoch	2	Discrete
patience	10	Discrete

Πίνακας 7.1: Πίνακας των παραμέτρων που χρησιμοποιούνται στην optuna, είτε ως σταθερές τιμές (fixed) είτε ως διακριτό ή συνεχές εύρος τιμών με σκοπό να βρεθεί η βέλτιστη

7.5 Αξιολόγηση

Τα δεδομένα μας χρησιμοποιούνται για ένα εξειδικευμένο είδος πρόβλεψης STLf: day-ahead forecasting. Σε αυτή την περίπτωση, δίνονται ιστορικά δεδομένα και γίνονται προβλέ-

ψεις (και φυσικά αξιολογούνται) για μελλοντικά δεδομένα των επόμενων 24 ωρών, κάνοντας χρήση της πρακτικής του backtesting (βλέπε κεφάλαιο 5.2.6)

Αξίζει να αναφερθεί ότι μετά την εκπαίδευση, για κάθε πείραμα που εκτελέστηκε χρησιμοποιήθηκε ένα πλήθος 20 όμοιων μοντέλων (estimators) με την τελική πρόβλεψη του συνόλου των μοντέλων μας (ensemble model) να προκύπτει από τη μέση τιμή (voting average) των προβλέψεων όλων των μοντέλων. Σαν κύρια μετρική, χρησιμοποιήθηκε το μέσο απόλυτο ποσοστιαίο σφάλμα (MAPE) ενώ επιπλέον υπολογίζονται χάριν πληρότητας και τα MSE, RMSE, MAE και sMAPE. Τέλος, απεικονίζουμε στο ίδιο σχεδιάγραμμα, τις γραφικές της προβλέψιμης και της πραγματικής χρονοσειράς. Λόγω του μεγέθους της χρονοσειράς, επιλέχτηκε ένα συγκεκριμένο (παραμετροποιήσιμο) πλήθος συνεχών δειγμάτων για την πιο ξεκάθαρη απεικόνιση τους.

Η αξιολόγηση των μοντέλων πραγματοποιείται:

1. στο τέλος της εκπαίδευσης των source και target μοντέλων. Εκεί εξετάζονται οι προβλέψεις των μοντέλων σε γνωστά μελλοντικά δεδομένα (test set) του source και target domain αντίστοιχα
2. στο τέλος της εκπαίδευσής του *ensemble* των target μοντέλων. Εκεί, εξετάζονται φυσικά έχοντας ως test-set τα ανάλογα δεδομένα της χώρας που βρίσκεται στο target domain

Μέρος ΙΙΙ

Αποτελέσματα & Συμπεράσματα

Κεφάλαιο 8

Αποτελέσματα Πειραμάτων

Αναφορικά με τα πειράματα που υλοποιήθηκαν, παρακάτω ακολουθούν οι συγκρίσεις αποδόσεις ανάμεσα στα διαφορετικά πειράματα. Η σύγκριση γίνεται καθαρά με την χρήση μετρικών που αναφέρθηκαν στο κεφάλαιο 7. Τα αποτελέσματα των μετρικών MAPE για κάθε πείραμα κάθε test case συγκεντρώθηκαν στον πίνακα 9.3.

Τα αποτελέσματα του *baseline* είναι για το μοντέλο μετά την εκπαίδευση και το *ensembling*. Τα αποτελέσματα του *All-TL* και *Cluster-TL* πάρθηκαν από το target μοντέλο, έχοντας προηγηθεί φυσικά transfer learning για την δημιουργία του και μετά από *ensembling* για καλύτερη αξιοπιστία των αποτελεσμάτων.

8.1 MLflow UI

Η οπτικοποίηση των παραμέτρων του μοντέλου, των υπερπαραμέτρων, των μετρήσεων απόδοσης και των χρόνων εκπαίδευσης όλων των διαφόρων μοντέλων και αλγορίθμων που εκπαιδεύονται είναι δυνατή μέσω ενός διαισθητικού διοικητικού περιβάλλοντος εργασίας χρήστη (βλέπε σχήμα 8.1). Στο σχήμα αυτό μπορεί να φανεί πως μπορούν επίσης να παρακολουθούνται τα καταχωρημένα μοντέλα και εξάγονται με ευκολία τα συμπεράσματα.

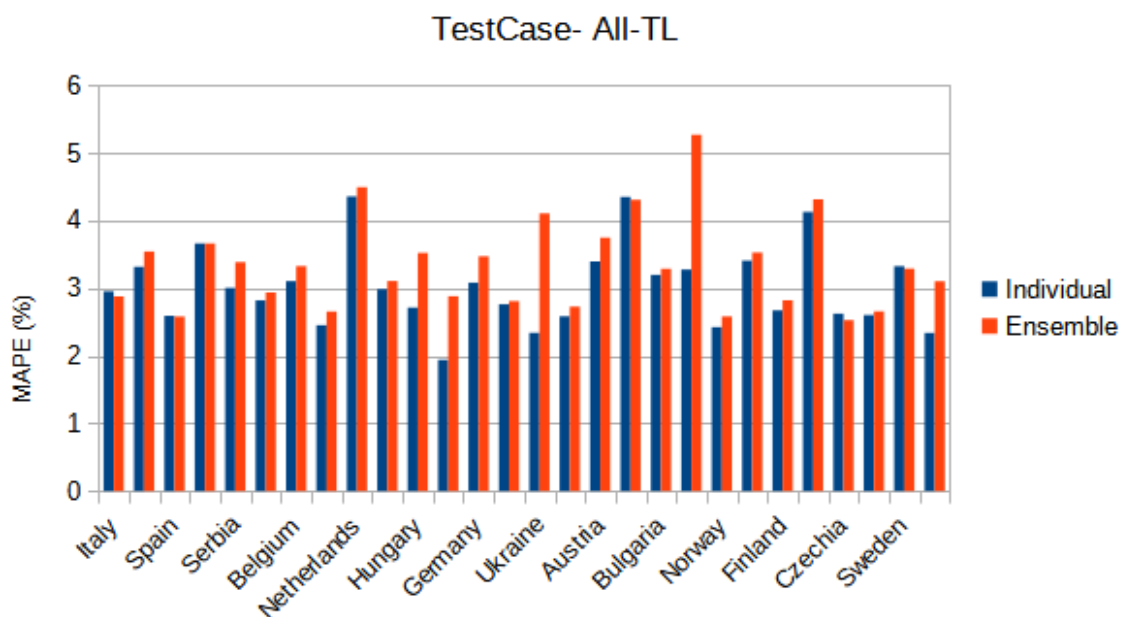
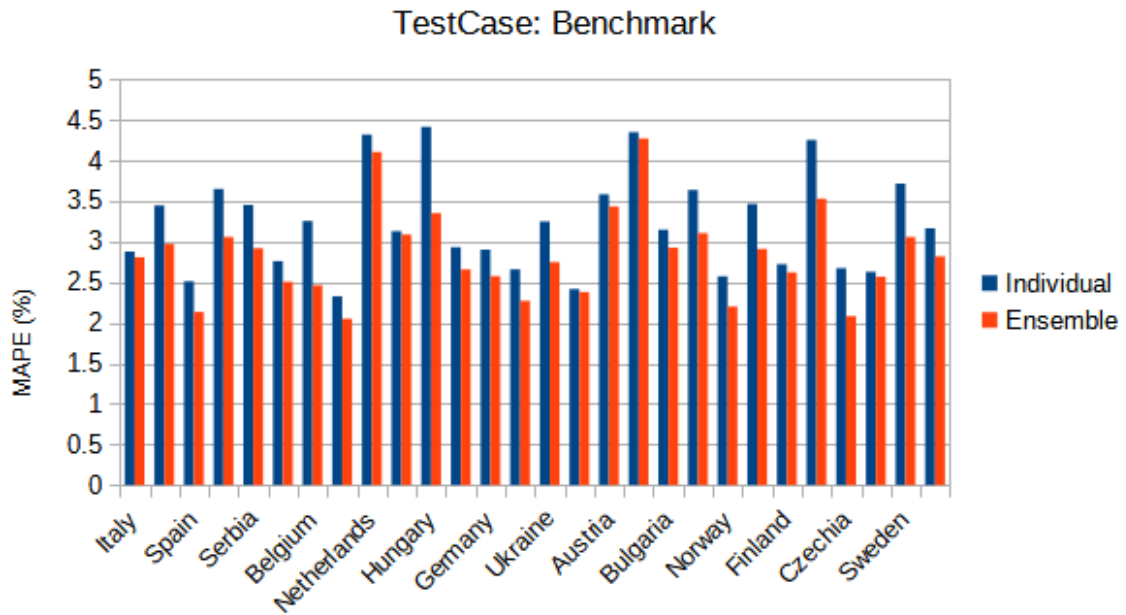
The top screenshot displays a table of 202 matching runs. The table has columns for Start Time, Run Name, Models, Parameters, and epochs. The parameters shown include activation (ReLU, SiLU), amsgrad (False), batch_size (1024), betas ((0.9, 0.999)), countries (Slovakia, Germany, Ro...), dir_in (./original_d..., /tmp/tmpxo..., /tmp/tmp1o...), and epochs (500). The bottom screenshot shows the details of a specific run, including artifacts like data, model, and training logs, and a section for making predictions using the logged model.

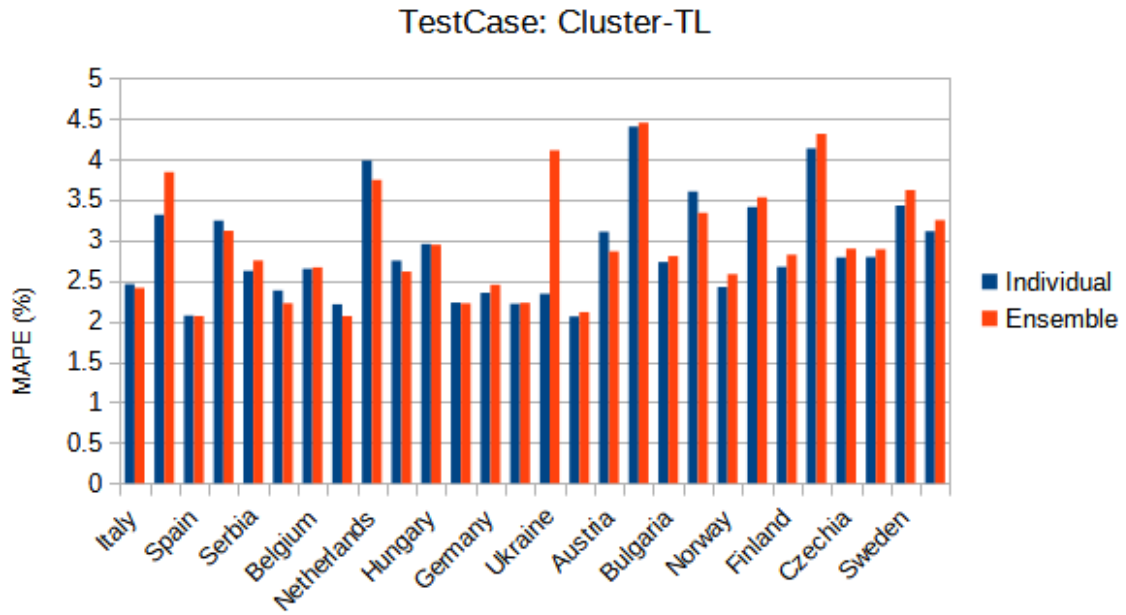
Σχήμα 8.1: Στην πρώτη εικόνα αυτή απεικονίζεται στο UI του mlflow κάποια απο τα πειράματα που ακολουθήθηκαν. Στην δεύτερη εικόνα φαίνεται εσωτερικά η αποθήκευση των μοντέλων pytorch

8.2 Σημασία Ensembling

Αρχικά κρίθηκε αναγκαίο να αποδειχθεί η σημασία χρήσης τεχνικών ensemble για την αξιοπιστία των πειραμάτων. Η χρήση μοναδικών μοντέλων ανά περίπτωση δεν θα έδινε αντιπροσωπευτική εικόνα για την αξιοπιστία της απόδοσης των μοντέλων. Κάτι τέτοιο παρατηρείται ξεκάθαρα στο σχήμα 8.1, απεικονίζεται η σύγκριση μεταξύ του μοντέλου μετά απο

ensembling και του μοντέλου χωρίς (single). Θεωρώντας την εκάστοτε ensemble μετρική MAPE σαν την μέση τιμή του αποτελέσματος του μοντέλου, βλέπουμε μεγάλη απόκλιση από την αντίστοιχη περίπτωση του μοναδιαίου μοντέλου. Μάλιστα, και στις τρεις περιπτώσεις υπάρχει σημαντική απόκλιση των MAPE από την ensemble σχεδόν για κάθε περίπτωση πειράματος. Αυτό το φαινόμενο μας αποδεικνύει την σημασία του ensembling και την αστάθεια των αποτελεσμάτων αν βασιζόμαστε αποκλειστικά στην μοναδιαία περίπτωση

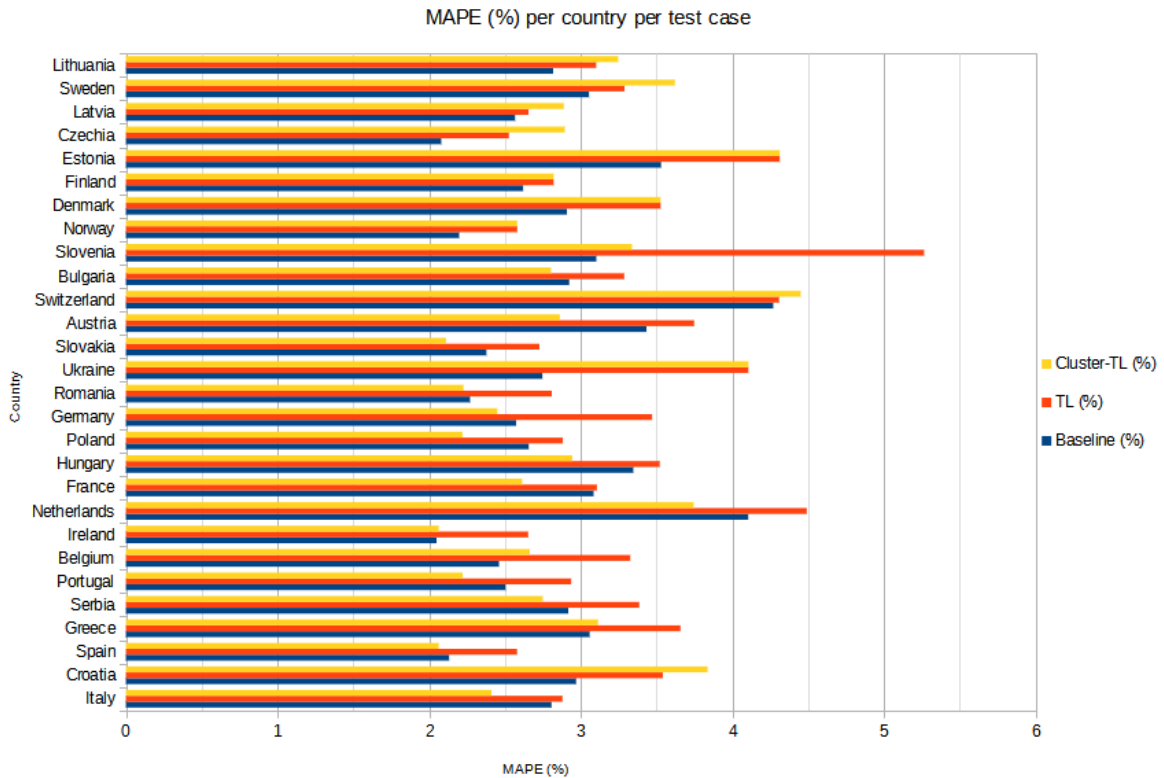




Σχήμα 8.1: Barplots που φαίνεται η σύγκριση του εκάστοτε μοντέλου με και χωρίς την χρήση ensemble με βάση την μετρική MAPE

8.3 Επίδοση transfer learning

Το σχήμα 8.2 εκφράζει τα την επίδοση των transfer learning τεχνικών, συγκρίνοντας τις μετρικές των test cases μεταξύ τους για τα αντίστοιχα πειράματα. Περιλαμβάνει τρεις μπάρες ανά χώρα, με την κάθε μια να αντιπροσωπεύει ένα ξεχωριστό test case. Αν και συμπυκνωμένο, είναι ξεκάθαρο, το transfer learning αποδίδει σε μεγάλο βαθμό ενάντια στην τυπική εκπαίδευση μοντέλων μηχανικής μάθησης, Υπάρχουν φυσικά κάποιες παρατηρήσεις για κάθε test case και οι οποίες θα αναλυθούν εκτενώς παρακάτω



Σχήμα 8.2: Ο πίνακας αυτός αντικατοπτρίζει τις αριθμητικές τιμές των μετρικών MAPE για κάθε πείραμα, για κάθε test case που εξετάστηκε στα πλαίσια της διπλωματικής. Έτσι, παρουσιάζεται μια πιο απτή αναπαράσταση των αποτελεσμάτων με σκοπό την καλύτερη κατανόηση της επίδοσης των μοντέλων

8.3.1 Αξιολόγηση All-TL

Η περίπτωση του *All-TL* test case δεν φαίνεται να παρουσιάζει σημαντική βελτίωση συγκριτικά με το baseline, έχοντας μεγάλη αύξηση του MAPE. Η υπόθεση μας για την αιτία πτώσης στην επίδοση περιστρέφεται γύρω από τους ακόλουθους λόγους:

- **Υπερ-γενίκευση μοντέλου:** Όπως αναφέρθηκε, τα μοντέλα αυτού του test case έχουν εκπαιδευτεί σε ένα μεγάλο εύρος δεδομένων. Τα δεδομένα αυτά αναφέρονται σε πολλές και συχνά ανεξάρτητες μεταξύ τους χώρες με διαφορετικές ενεργειακές κατανομές και ανάγκες. Επομένως, για τον σκοπό της ακριβής πρόβλεψης των δεδομένων μιας μεμονωμένης χώρας, η οποία μάλιστα δεν βρισκόταν στο training set, θα αποτελούσε σίγουρα πρόκληση για τις δυνατότητές του
- **περιορισμούς μοντέλου:** Στα μοντέλα χρησιμοποιήθηκε περιορισμένος αριθμός estimators στο ensemble (10 αντί για 20-30), ενώ ταυτόχρονα εξετάστηκαν επίσης για

πολύ μικρό αριθμό νευρώνων ανά layer του μοντέλου από το προβλεπόμενο (<336 έναντι τιμών έως και 600).

Η φύση του test case και η αρχιτεκτονική των μοντέλων σίγουρα επηρέασαν την αποδοτικότητα του και επομένως χρειάζονται περισσότεροι πειραματισμοί για να βεβαιωθούμε για την πιστότητα των ισχυρισμών μας.

8.3.2 Αξιολόγηση Cluster-TL

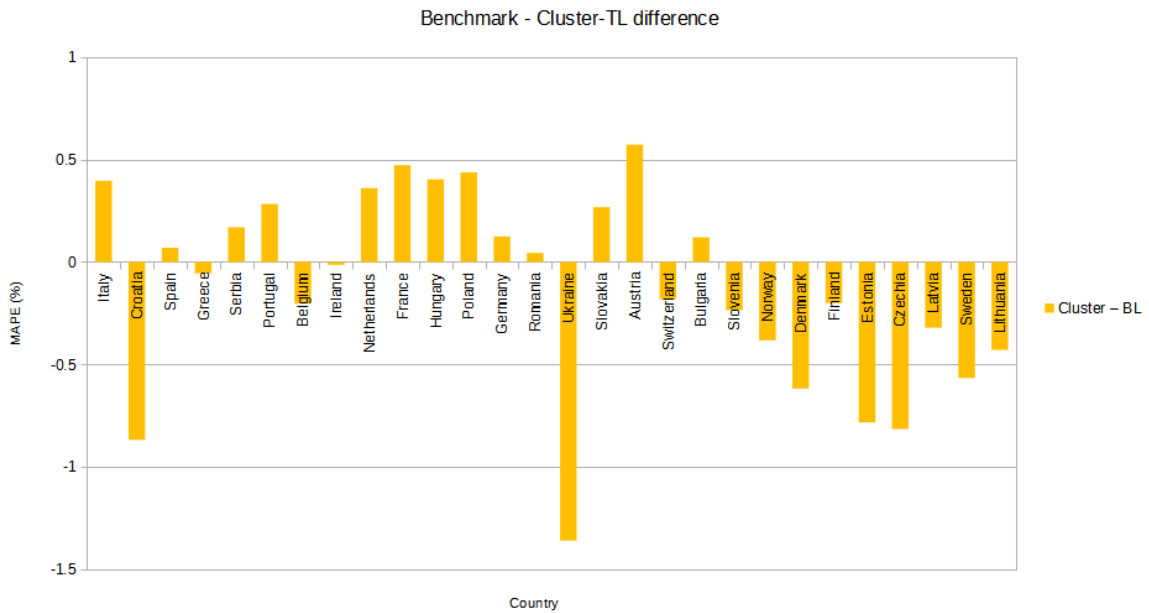
Αντίθετα με το *All-TL*, το συγκεκριμένο test case έχει σημαντική βελτίωση σε αρκετές περιπτώσεις πρόβλεψης, έχοντας ξεκάθαρη μείωση του MAPE σε σχέση με το baseline μοντέλο. Στο σχήμα 8.3 φαίνεται η διαφορά (difference) ανάμεσα στο MAPE των δύο test cases. Μάλιστα, η μέθοδος του clustering είναι ιδιαίτερη παραγωγική στην περίπτωση του πρώτου cluster (μεσογειακές χώρες), αλλά **κυρίως** του δευτέρου cluster (Σερβία έως Πολωνία).

Φυσικά, υπάρχουν περιπτώσεις που η μέθοδος clustering δεν αποδίδει ανάλογα με τις υπόλοιπες περιπτώσεις του ίδιου cluster (Κροατία/Ελλάδα, Ιρλανδία κλπ). Αυτή η συμπεριφορά μπορεί να οφείλεται σε αρκετούς παράγοντες όπως:

- **δυσανάλογη οικονομία/πληθυσμός:** Το test case που γίνεται clustering βασίζεται στην θεώρηση ότι δημιουργούνται σύνολα με όμοιες χώρες. Παρ'όλα αυτά, όσο και ακριβής να γίνει η ομαδοποίηση, υπάρχουν πάντα αποκλείσεις. Για παράδειγμα μοντέλα εκπαιδευμένα σε χώρες με οικονομία, βιομηχανία και πληθυσμό όπως της Ιταλίας και Ισπανίας, αναμενόμενα θα έχουν κακή απόδοση αν ζητηθεί να προβλέψουν τις ενεργειακές ανάγκες όπως αυτές της Κροατίας ή ακόμα και της Ελλάδας
- **εξωτικές περιπτώσεις (outliers):** Χώρες όπως η Ελβετία έχουν εξωτική οικονομία, οργάνωση και πληθυσμό, επομένως καθίσταται δύσκολη η ομαδοποίηση τους με άλλες χώρες. Ταυτόχρονα, η ποσότητα των δεδομένων επηρεάζει σημαντικά την απόδοση. Για παράδειγμα, τα δεδομένα που υπήρχαν για την Ουκρανία ήταν αποκλειστικά για τις χρονιές 2019, 2020 και 2021, ενώ σε πολλές χώρες έφταναν όσο παλιά όσο το 2015. Ένα τόσο περιορισμένο training set έχει πολλές πιθανότητες να μην οδηγήσει σε ακριβείς προβλέψεις κάτι το οποίο επιβεβαιώνεται από το σχήμα 8.3

Σε αυτό το σημείο, οφείλεται να αναφερθεί η αναπάντεχα κακή απόδοση των τελευταίων χωρών του σχήματος 8.3. Το μεγάλο αυτό συνεχές σύνολο χωρών έχει αποκλειστικά τις χώρες του τέταρτου cluster. Ένα cluster στο οποίο περιλαμβάνονται σκανδιναβικές και βαλτικές

χώρες: χώρες με πολύ διαφορετική οικονομία, πλούτο και παραγωγή (δυσανάλογες) Έτσι, σε αυτή την δυσανάλογη κατηγοριοποίηση πιστεύουμε ότι οφείλεται η κακή απόδοση του test case, παρόλα αυτά, χρειάζεται να γίνουν επιπλέον προσεκτικά πειράματα, προκειμένου να σιγουρευτούμε για την εγκυρότητα του ισχυρισμού αυτού.



Σχήμα 8.3: Barplots που φαίνεται η σύγκριση των μετρικών MAPE μεταξύ του *baseline* και *Cluster-TL* test cases. Κάθε μπάρα, αντιπροσωπεύει την αριθμητική διαφορά της μετρικής MAPE ανάμεσα στα δύο testcases. Θετική τιμή σημαίνει θετικός αριθμός και άρα το MAPE του baseline ήταν μεγαλύτερο (χειρότερο), ενώ αρνητική τιμή σημαίνει φυσικά το αντίθετο

Κεφάλαιο 9

Συμπεράσματα

9.1 Σύνοψη και συμπεράσματα

Η παρούσα διπλωματική εστίασε στη μελέτη της εφαρμογής μεθόδων μεταφοράς μάθησης με σκοπό την βελτίωση της προβλεπτικής ακρίβειας νευρωνικών δικτύων στα αποτελέσματα. Σαν αρχιτεκτονική, επιλέχτηκε η χρήση ενός πολυστρωματικού δικτύου perceptron με αρχιτεκτονική (υπερπαράμετροι) εξατομικευμένοι για κάθε ξεχωριστή περίπτωση πειράματος.

Έγινε προσεκτική μελέτη των δεδομένων, με σκοπό να προκύψουν πορίσματα για την φύση και την συμπεριφορά τους, με σκοπό την ομαδοποίηση δεδομένων όμοιων χωρών (clustering). Από αυτή την ανάλυση προέκυψε ότι τα δεδομένα των χωρών, όταν δίνονται ως προς την τοπική ζώνη ώρας της εκάστοτε χώρας, παρουσιάζουν ομοιότητες για χώρες με γειτονική γεωγραφία και κουλτούρα. Έτσι, μπορέσαμε να ομαδοποιήσουμε τις χώρες στις χώρες:

1. Μεσογείου
2. κεντρικής και δυτικής Ευρώπης
3. κεντρικής και ανατολικής Ευρώπης
4. Σκανδιναβίας και Βαλτικής

Εκτελέστηκαν πειράματα κάνοντας χρήση της τεχνικής *WARM START*. Η τεχνική αυτή πρεσβεύει ότι η πληροφορία (παράμετροι) ενός μοντέλου που εκπαιδεύτηκε σε γειτονικό πρόβλημα μπορεί να αποτελέσει την βάση (αρχική τιμή) των παραμέτρων για το μοντέλο του ζητούμενου στόχου. Εκτελέστηκαν δύο διαφορετικά είδη πειραμάτων:

- **All-TL**: Εξετάζουμε την επίδοση του μοντέλου πρόβλεψης δεδομένων μιας χώρας όταν εκπαιδεύεται με τα δεδομένα όλων των υπόλοιπων χωρών
- **Cluster-TL**: Σε κάθε εκτέλεση, δημιουργείται ένα μοντέλο στο οποίο προβλέπεται η ζήτηση της εκάστοτε χώρας, έχοντας όμως εκπαιδευτεί μόνο με τα δεδομένα των υπόλοιπων χωρών της ίδιας ομάδας (cluster). Τα clusters έχουν προκύψει από την ερευνητική ανάλυση που έγινε στα δεδομένα στο κεφάλαιο 6

Αναφορικά με την περίπτωση του *All-TL*, παρατηρούμε ότι δεν υπήρξε ιδιαίτερη βελτίωση συγκριτικά με το αρχικό (baseline) μοντέλο. Η χρήση ολόκληρου του πλήθους δεδομένων για την εκπαίδευση ενός πηγαίου μοντέλου, είναι πιθανό να οδηγεί σε υπερ-γενίκευση του μοντέλου. Ταυτόχρονα, ο χρόνος που χρειάζεται για την κατασκευή του μοντέλου είναι σημαντικά περισσότερος, και εκ πρώτης όψεως δεν προτιμάται ενάντια της αρχικής περίπτωσης. Παρόλα αυτά, είναι σημαντικό να αναφερθεί ότι χρειάζεται παραπάνω πειραματισμός, καθώς η εκπαίδευση του δεν έγινε με εσκεμμένα σημαντικά πιο περιορισμένο αριθμό νευρώνων και στιβάδων από το προτεινόμενο που μπορεί να ευθύνεται για την κακή απόδοση.

Όσο για την περίπτωση του *Cluster-TL*, παρατηρούμε σημαντική βελτίωση σε αρκετές περιπτώσεις συγκριτικά με το baseline. Ιδιαίτερη βελτίωση γίνεται στις περιπτώσεις που οι ομάδες αποτελούνται καθαρά από χώρες με πολύ γειτονικές ενεργειακές ανάγκες. Σε αντίθετη περίπτωση, βλέπουμε σημαντική πτώση της απόδοσης με το baseline να αποτελεί καλύτερη επιλογή.

Συνολικά, μπορούμε να δηλώσουμε ότι η εφαρμογή της μάθησης μεταφοράς είναι επωφελής για την αύξηση της ακρίβειας πρόβλεψης του μοντέλου και για τη συντόμευση της περιόδου εκπαίδευσης. Η μεταφορά μάθησης πιθανόν να έχει μικρή έως μηδαμινή επίδραση στην προβλεπόμενη ακρίβεια του μοντέλου για σύνολα δεδομένων που απέχουν πολύ μεταξύ τους. Αντίθετα, μπορεί να παράγει το ίδιο σφάλμα σε ένα μικρό τμήμα της περιόδου εκπαίδευσης σε σύγκριση με την εκπαίδευση από την αρχή. Λόγω της ισχυρής συνάφειας της διδαχθείσας γνώσης, η μεταφορά μάθησης δίνει αυξημένη ακρίβεια πρόβλεψης σε σύνολα δεδομένων που είναι πανομοιότυπα μεταξύ τους. Στην αντίθετη περίπτωση, η επανεκπαίδευση σταματά να λειτουργεί και συνιστάται να γίνονται προβλέψεις απευθείας από το αρχικό μοντέλο.

9.2 Μελλοντικές επεκτάσεις

Στην παρούσα διπλωματική, η ανάλυση των δεδομένων έγινε αποκλειστικά από την παράμετρο ζήτησης ηλεκτρικής ενέργειας (energy load) χωρίς να μας απασχολούν άλλοι παράγοντες. Παρά την απλότητα αυτής της σύμβασης, τα μοντέλα μας, αποδείχτηκαν αρκετά αποτελεσματικά στην πρόβλεψη μελλοντικών φορτίων. Υπάρχει, ωστόσο, αρκετό περιθώριο επιπλέον έρευνας για την απόδοση των μοντέλων όταν λαμβάνονται υπόψη κι άλλοι παράμετροι (κόστος, καιρικές συνθήκες κ.α) τόσο στην εκπαίδευση όσο και στην ομαδοποίηση (clustering) των χωρών.

Επίσης, η αρχιτεκτονική με την οποία εργαστήκαμε ήταν αρκετά απλή. Παρόλα αυτά, μέσα στην απλότητα της το MLP επέβαλε ορισμένους περιορισμούς για την δομή των μοντέλων, ωστόσο ήταν σε θέση να μας παρέχει κάποια πολύ σημαντικά αρχικά συμπεράσματα σχετικά με το πόσο αξιόλογη είναι η χρήση της μεταφοράς μάθησης στον τομέα των προβλέψεων. Παραπάνω, όμως έρευνα χρειάζεται να γίνει με χρήση πιο σύνθετων αρχιτεκτονικών, όπως επαναληπτικά (recurrent) ή συνελκτικά (convolutional) δίκτυα ή ακόμα, και βαθύτερες αρχιτεκτονικές πολυστρωματικών δικτύων perceptron (MLP). Μια τέτοια έρευνα θα βοηθήσει για να ξεκαθαριστεί η σχέση ανάμεσα στο μέγεθος/φύση των μοντέλων και τον όγκο των δεδομένων. Θα μπορούσε να αποδειχθεί μια υπόθεση πως περιορισμένα δεδομένα στόχου, κάνοντας χρήση ενός απλού MLP, θα προσαρμοστούν καλύτερα συγκρίνοντάς τα με ένα βαρύ δίκτυο LSTM

Συνεχίζοντας, η διπλωματική αυτή εστίασε στην χρήση της τεχνικής transfer learning της ευνοϊκής αρχής (*WARM START*). Είναι η βασικότερη και επιλέχτηκε για να εξεταστεί αν αξίζει η χρήση transfer learning σε τέτοιες περιπτώσεις προβλημάτων. Παρόλα αυτά, όπως αναλύθηκε στο κεφάλαιο 4 υπάρχουν αρκετές τεχνικές μεταφοράς μάθησης οι οποίες δυνητικά μπορεί να αποδίδουν καλύτερα μόνες τους ή σε συνδυασμό με αυτή της ευνοϊκής αρχής. Για παράδειγμα, η τεχνική του παγώματος (*freezing*) σε συνδυασμό με τις περιορισμένες εποχές (*bounded epochs*) αποτελεί μια υπολογιστικά ελαφρύτερη μέθοδος η οποία μπορεί να οδηγήσει καθολικά ή σε συγκεκριμένες περιπτώσεις σε καλύτερη απόδοση. Σε κάθε περίπτωση, υπάρχει ανάγκη για περισσότερα έρευνα πριν προβούμε σε οποιαδήποτε συμπεράσματα

Ως αποτέλεσμα της έρευνας που παρουσιάζεται σε αυτή την διπλωματική, πιστεύουμε ακράδαντα ότι πρέπει να αναπτυχθεί μια πλατφόρμα συγκέντρωσης δεδομένων και χρονοσειρών σε ολόκληρη τη βιομηχανία. Επιπλέον, ενθαρρύνεται η ανάπτυξη υπερμοντέλων που

μπορούν να χρησιμοποιηθούν ή να επανεκπαιδευτούν, όπως ακριβώς στον τομέα της επεξεργασίας εικόνων [82]. Αυτές οι μέθοδοι μπορούν να έχουν σημαντικό θετικό αντίκτυπο στις επιχειρήσεις και τους οργανισμούς, επειδή απομονώνουν την ποσότητα των δεδομένων εκπαίδευσης από την ακρίβεια των προβλέψεων. Οποιαδήποτε επιχείρηση, ανεξάρτητα από το πόσο λίγα δεδομένα έχει στη διάθεσή της, μπορεί να λειτουργήσει επανεκπαιδύοντας ένα από τα διαθέσιμα υπερμοντέλα για να αναπτύξει μοντέλα που παρέχουν ακριβείς προβλέψεις με λογικό κόστος. Η μεταφορά της μάθησης, σύμφωνα με τους συγγραφείς, θα είναι μια πολύ αποδοτική στρατηγική για τις επιχειρήσεις και τους οργανισμούς ώστε να αξιοποιήσουν και να ενσωματώσουν τη γνώση που έχουν στη διάθεσή τους στο μέλλον, όταν τα ανοικτά δεδομένα και μοντέλα θα χρησιμοποιούνται ευρύτερα.

Βιβλιογραφία

- [1] Hannah Ritchie, Max Roser, and Pablo Rosado. Energy. *Our World in Data*, 2022. <https://ourworldindata.org/energy>.
- [2] Gamze Nalcaci, Ayse Özmen, and Gerhard Wilhelm Weber. Long-term load forecasting: models based on mars, ann and lr methods. *Central European Journal of Operations Research 2018 27:4*, 27:1033–1049, 3 2018.
- [3] Eisa Almeshaiei and Hassan Soltan. A methodology for electric power load forecasting. *Alexandria Engineering Journal*, 50:137–144, 2011.
- [4] Ümmühan Basaran Filik, Ömer Nezih Gerek, and Mehmet Kurban. A novel modeling approach for hourly forecasting of long-term electric energy demand. *Energy Conversion and Management*, 52:199–211, 1 2011.
- [5] Hong Wang, Khalid A. Alattas, Ardashir Mohammadzadeh, Mohammad Hosein Sabzalian, Ayman A. Aly, and Amir Mosavi. Comprehensive review of load forecasting with emphasis on intelligent computing approaches. *Energy Reports*, 8:13189–13198, 11 2022.
- [6] Hesham K. Alfares and Mohammad Nazeeruddin. Electric load forecasting: Literature survey and classification of methods. *International Journal of Systems Science*, 33:23–34, 1 2002.
- [7] Xinquan Wei, Weiyan Zheng, and Xiangjun Duan. Iop conference series: Earth and environmental science a review of load forecasting of the distributed energy system short-term load forecasting system for power system based on big data yang donghui-research on short-term load forecasting based on feature similarity using pso algorithm. *Journal of Physics Conference Series*, 2021.

- [8] V. Y. Kondaiah, B. Saravanan, P. Sanjeevikumar, and Baseem Khan. A review on short-term load forecasting models for micro-grid application. *The Journal of Engineering*, 2022:665–689, 7 2022.
- [9] Ahmed Tealab. Time series forecasting using artificial neural networks methodologies: A systematic review. *Future Computing and Informatics Journal*, 3:334–340, 12 2018.
- [10] Zhongyi Hu, Yukun Bao, Raymond Chiong, and Tao Xiong. Mid-term interval load forecasting using multi-output support vector regression with a memetic algorithm for feature selection. *Energy*, 84:419–431, 5 2015.
- [11] Boris N. Oreshkin, Grzegorz Dudek, Paweł Pełka, and Ekaterina Turkina. N-beats neural network for mid-term electricity load forecasting. *Applied Energy*, 293:116918, 7 2021.
- [12] Swasti R. Khuntia, José L. Rueda, and Mart A.M.M. van der Meijden. Forecasting the load of electrical power systems in mid- and long-term horizons: A review. *IET Generation, Transmission and Distribution*, 10:3971–3977, 12 2016.
- [13] Nooriya A. Mohammed and Ammar Al-Bazi. An adaptive backpropagation algorithm for long-term electricity load forecasting. *Neural Computing and Applications*, 34:477–491, 1 2022.
- [14] Arunesh Kumar Singh, Ibraheem, S. Khatoon, Md Muazzam, and D. K. Chaturvedi. Load forecasting techniques and methodologies: A review. *ICPCES 2012 - 2012 2nd International Conference on Power, Control and Embedded Systems*, 2012.
- [15] Mostafa Askari and Farshid Keynia. Mid-term electricity load forecasting by a new composite method based on optimal learning mlp algorithm. *IET Generation, Transmission and Distribution*, 14:845–852, 3 2020.
- [16] Seung Min Jung, Sungwoo Park, Seung Won Jung, and Eenjun Hwang. Monthly electric load forecasting using transfer learning for smart cities. *Sustainability 2020, Vol. 12, Page 6364*, 12:6364, 8 2020.
- [17] Bjarne Steffen and Anthony Patt. A historical turning point? early evidence on how the russia-ukraine war changes public support for clean energy policies. *Energy Research Social Science*, 91:102758, 9 2022.

- [18] Shahab Bahrami, Yu Christine Chen, and Vincent W.S. Wong. Deep Reinforcement Learning for Demand Response in Distribution Networks. *IEEE Transactions on Smart Grid*, 12(2):1496–1506, 3 2021.
- [19] Sotiris Pelekis, Evangelos Karakolis, Francisco Silva, and Schoinas et al. In Search of Deep Learning Architectures for Load Forecasting: A Comparative Analysis and the Impact of the Covid-19 Pandemic on Model Performance. In *2022 13th International Conference on Information, Intelligence, Systems and Applications (IISA)*, pages 1–8. IEEE, 7 2022.
- [20] Evangelos Karakolis, Sotiris Pelekis, Spiros Mouzakitis, Ourania Markaki, Katerina Pappapostolou, Giorgos Korbakis, and John Psarras. Artificial Intelligence for Next Generation Energy Services Across Europe - The I-ENERGY Project. In *ES 2021 : 19th International Conference e-Society 2021*, pages 61–68, Lisbon, 2022.
- [21] Marco Pau, Panagiotis Kapsalis, Zhiyu Pan, George Korbakis, Dario Pellegrino, and Antonello Monti. MATRYCS A Big Data Architecture for Advanced Services in the Building Domain. *Energies*, 15(7):2568, 4 2022.
- [22] Katharina Aurelia Wehrmeister, Efthimios Bothos, and Marinakis et al. The BD4NRG Reference Architecture for Big Data Driven Energy Applications. *13th International Conference on Information, Intelligence, Systems and Applications, IISA 2022*, 2022.
- [23] Santiago Gomez-Rosero, Miriam A. M. Capretz, and Syed Mir. Transfer learning by similarity centred architecture evolution for multiple residential load forecasting. *Smart Cities 2021, Vol. 4, Pages 217-240*, 4:217–240, 2 2021.
- [24] Eunjung Lee and Wonjong Rhee. Individualized short-term electric load forecasting with deep neural network based transfer learning and meta learning. *IEEE Access*, 9:15413–15425, 2021.
- [25] Dheeru Dua and Casey Graff. Uci machine learning repository, 2017. [Online] Available (visited on 14/12/2022).
- [26] Di Wu and Weixuan Lin. Efficient residential electric load forecasting via transfer learning and graph neural networks. *IEEE Transactions on Smart Grid*, 2022.

- [27] Zhenyuan Zhang, Pengfei Zhao, Peng Wang, and Wei Jen Lee. Transfer learning featured combining short-term load forecast with small-sample conditions. *Conference Record - IAS Annual Meeting (IEEE Industry Applications Society)*, 2021-October, 2021.
- [28] Long Cai, Jie Gu, and Zhijian Jin. Two-layer transfer-learning-based architecture for short-term load forecasting. *IEEE Transactions on Industrial Informatics*, 16:1722–1732, 3 2020.
- [29] Chuong B. Do and Andrew Y. Ng. Transfer learning for text classification. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems*, volume 18. MIT Press, 2005.
- [30] Entso-e | the european association for the cooperation of transmission system operators (tsos) for electricity. [Online] Available (visited on 14/12/2022).
- [31] R.J. Hyndman and G Athanasopoulos. *Forecasting: Principles and Practice (2nd ed)*. OTexts: Melbourne, Australia, third edition, 2018.
- [32] André Bauer. *Automated Hybrid Time Series Forecasting: Design, Benchmarking, and Use Cases*. PhD thesis, Universität Würzburg, Fakultät für Mathematik und Informatik, 01 2021.
- [33] Louis Cialdella. Understanding the difference between prediction and confidence intervals for linear models in python | casual inference, 9 2020.
- [34] Linear/smooth interpolation | wiktionary. [Online] Available (visited on 14/12/2022).
- [35] Dan Roth, Ben Zhou, C. Cervantes, and C. Cheng. Introduction to machine learning. *CS 446 Machine Learning*, pages 1–11, 8 2016.
- [36] Thomas Elsken, Jan Hendrik Metzen, and Frank Hutter. Neural architecture search: A survey. *Journal of Machine Learning Research*, 20:1–21, 8 2018.
- [37] Dozdar Mahdi Ahmed, Masoud Muhammed Hassan, and Ramadhan J. Mstafa. A review on deep sequential models for forecasting time series data. *Applied Computational Intelligence and Soft Computing*, 2022, 2022.

- [38] Saptarshi Sengupta, Sanchita Basak, and Pallabi Saikia et al. A review of deep learning with special emphasis on architectures, applications and recent trends. *Knowledge-Based Systems*, 194, 4 2020.
- [39] Konstantinos Benidis, Syama Sundar Rangapuram, and Valentin Flunkert et al. Danielle maddix, dmmaddix@amazon.com, amazon research, 1900 university ave. *Journal of the ACM*, 37:37, 1096.
- [40] Derivative: Official wikipedia page. [Online] Available (visited on 14/12/2022).
- [41] Jiajun Wu Fei-Fei Li and Ruohan Gao. Cs231n: Convolutional neural networks for visual recognition - stanford.
- [42] Payam Refaeilzadeh, Lei Tang, and Huan Liu. *Cross-Validation*. Springer, Boston, MA, 2009.
- [43] Hilde J. P. Weerts, Andreas C. Mueller, and Joaquin Vanschoren. Importance of tuning hyperparameters of machine learning algorithms, 7 2020.
- [44] Jia Wu, Xiu Yun Chen, Hao Zhang, Li Dong Xiong, Hang Lei, and Si Hao Deng. Hyperparameter optimization for machine learning models based on bayesian optimization. *Journal of Electronic Science and Technology*, 17:26–40, 3 2019.
- [45] Adrian E. Bayer and Uros Seljak. The look-elsewhere effect from a unified bayesian and frequentist perspective. *Journal of Cosmology and Astroparticle Physics*, 2020, 7 2020.
- [46] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22:1345–1359, 2010.
- [47] Mohammadreza Iman, Khaled Rasheed, and Hamid Reza Arabnia. A review of deep transfer learning and recent advancements, 1 2022.
- [48] Tamoghna Ghosh Dipanjan Sarkar, Raghav Bali. *Hands-on Transfer Learning with Python*. Packt, 2020.
- [49] gunes (<https://stats.stackexchange.com/users/204068/gunes>). Transfer learning for regression problems. Cross Validated. URL:<https://stats.stackexchange.com/q/545879> (version: 2021-09-24).

- [50] Matei A. Zaharia, Andrew Chen, and Aaron Davidson. Accelerating the machine learning lifecycle with mlflow. *IEEE Data Eng. Bull.*, 41:39–45, 2018.
- [51] Abhinandan Jain. Darts - multibody modeling, simulation and analysis software. In *Computational Methods in Applied Sciences book series (COMPUTMETHODS, volume 53)*, pages 433–441, 01 2020.
- [52] Wes McKinney et al. Data structures for statistical computing in python. In *Proceedings of the 9th Python in Science Conference*, volume 445, pages 51–56. Austin, TX, 2010.
- [53] John D Hunter. Matplotlib: A 2d graphics environment. *Computing in science & engineering*, 9(3):90–95, 2007.
- [54] Michael Waskom, Olga Botvinnik, and Drew O’Kane. mwaskom/seaborn: v0.8.1 (september 2017), 9 2017. [Online] Available (visited on 14/12/2022).
- [55] Plotly Technologies Inc. Collaborative data science, 2015. [Online] Available (visited on 14/12/2022).
- [56] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, and Haberland. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.
- [57] Francois Chollet et al. Keras, 2015. [Online] Available (visited on 14/12/2022).
- [58] Martín Abadi, Paul Barham, and Chen et al. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pages 265–283, 2016.
- [59] Adam Paszke, Sam Gross, Francisco Massa, and Lerer et al. *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. Curran Associates, Inc., 2019.
- [60] I. Stancin and A. Jovic. An overview and comparison of free python libraries for data mining and big data analysis. *2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2019 - Proceedings*, pages 977–982, 5 2019.
- [61] Stefan Braun. Lstm benchmarks for deep learning frameworks, 6 2018.

- [62] Chirodea, Mihai Cristian, and Ovidiu Constantin Novac. Comparison of tensorflow and pytorch in convolutional neural network - based applications. *Proceedings of the 13th International Conference on Electronics, Computers and Artificial Intelligence, ECAI 2021*, 7 2021.
- [63] Saeed S. Alahmari, Dmitry B. Goldgof, Peter R. Mouton, and Lawrence O. Hall. Challenges for the repeatability of deep learning models. *IEEE Access*, 8:211860–211868, 2020.
- [64] Pytorch lightning | the ultimate pytorch research framework. scale your models, without the boilerplate. [Online] Available (visited on 14/12/2022).
- [65] William Falcon and Kyunghyun Cho. A framework for contrastive self-supervised learning and designing a new approach, 8 2020.
- [66] Ensemble pytorch — a unified ensemble framework for pytorch. [Online] Available (visited on 14/12/2022).
- [67] Lars Buitinck, Gilles Louppe, Mathieu Blondel, and Fabian Pedregosa et al. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013.
- [68] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2623–2631, 7 2019.
- [69] Shashank Shekhar, Adesh Bansode, and Asif Salim. A comparative study of hyperparameter optimization tools. *2021 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*, pages 1–6, 12 2021.
- [70] Akiba, Takuya Sano, and Shotaro Yanase et al. Optuna: A next-generation hyperparameter optimization framework. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2623–2631, 7 2019.
- [71] Dirk Merkel. Docker: lightweight linux containers for consistent development and deployment. *Linux journal*, 2014(239):2, 2014.

- [72] Minio | high performance, kubernetes native object storage. [Online] Available (visited on 14/12/2022).
- [73] Jouni Peppanen, Xiaochen Zhang, Santiago Grijalva, and Matthew J. Reno. Handling bad or missing smart meter data through advanced data imputation. *2016 IEEE Power and Energy Society Innovative Smart Grid Technologies Conference, ISGT 2016*, 12 2016.
- [74] Guido Van Rossum. *The Python Library Reference, release 3.8.2*. Python Software Foundation, 2020.
- [75] Pyae Pyae Phyo, Yung Cheol Byun, and Namje Park. Short-term energy forecasting using machine-learning-based ensemble voting regression. *Symmetry 2022, Vol. 14, Page 160*, 14:160, 1 2022.
- [76] David Opitz and Richard Maclin. Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research*, 11:169–198, 8 1999.
- [77] Linxi Meng, Will Treem, Graham A. Heap, and Jingjing Chen. A stacking ensemble machine learning model to predict alpha-1 antitrypsin deficiency-associated liver disease clinical outcomes based on uk biobank data. *Scientific Reports 2022 12:1*, 12:1–18, 10 2022.
- [78] Kenniy Olorunnimbe and Herna Viktor. Deep learning in the stock market—a systematic survey of practice, backtesting, and applications. *Artificial Intelligence Review*, pages 1–53, 6 2022.
- [79] V. Vijaya, Shweta Sharma, and Neha Batra. Comparative study of single linkage, complete linkage, and ward method of agglomerative clustering. *Proceedings of the International Conference on Machine Learning, Big Data, Cloud and Parallel Computing: Trends, Prespectives and Prospects, COMITCon 2019*, pages 568–573, 2 2019.
- [80] Laura Ferreira and David B. Hitchcock. A comparison of hierarchical methods for clustering functional data. <http://dx.doi.org/10.1080/03610910903168603>, 38:1925–1949, 10 2009.
- [81] University of Oxford Our World In Data. Statistical review of world electricity demand, 2015-2021.

-
- [82] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 3 2010.

Παράρτημα

Arguments	Entrypoints					
	Main	Load	ETL	Optuna	Train	Eval
stages	✓	✗	✗	✗	✗	✗
src_countries	✓	✗	✗	✗	✗	✗
tgt_countries	✓	✗	✗	✗	✗	✗
countries	✗	✓	✓	✓	✓	✓
dir_in	✓	✓	✓	✓	✓	✓
local_tz	✓	✗	✓	✗	✗	✗
seed	✓	✗	✗	✓	✓	✓
n_trials	✓	✗	✗	✓	✗	✗
max_epochs	✓	✗	✗	✓	✓	✓
n_layers	✓	✗	✗	✓	✓	✓
layer_sizes	✓	✗	✗	✓	✓	✓
l_window	✓	✗	✗	✓	✓	✓
f_horizon	✓	✗	✗	✓	✓	✓
l_rate	✓	✗	✗	✓	✓	✓
activation	✓	✗	✗	✓	✓	✓
optimizer_name	✗	✓	✗	✓	✓	✓
batch_size	✓	✗	✗	✓	✓	✓
transfer_mode	✓	✗	✗	✓	✓	✓
n_estimators	✓	✗	✗	✗	✗	✓
num_workers	✓	✗	✗	✓	✓	✗

Πίνακας 9.1: Πίνακας με τα στάδια του pipeline (entrypoints) και οι παράμετροι που προορίζονται για αυτά

Arguments	Description
stages	ονόματα entrypoints προς εκτέλεση διαχωρισμένα με κόμμα
src_countries	ονόματα αρχείων csv εισόδου που χρησιμοποιούνται από το source μοντέλο
tgt_countries	ονόματα αρχείων csv εισόδου που χρησιμοποιούνται από το μοντέλο στόχου
countries	src_countries ή tgt_countries ανάλογα την εκτέλεση ή μη transfer learning (για χρήση στο training)
dir_in	Path φακέλου με τα προεπεξεργασμένα (post-ETL) datasets. Στην περίπτωση που εκτελείται το ETL, αντιπροσωπεύει το φάκελο εισόδου του
dir_out	Φάκελος αποθήκευσης των datasets post-ETL
local_tz	σημαία για προεπεξεργασία ETL με βάση την τοπική (True) ή UTC (False) ζώνη ώρας της χώρας
seed	ψευδοτυχαίος αριθμός με σκοπό της αρχικοποίησης (random state) του μοντέλου
n_trials	πλήθος trials στην Optuna (βλέπε κεφάλαιο 5.2.3)
max_epochs	πλήθος(ή εύρος *) εποχών που χρησιμοποιεί το μοντέλο για σύγκλιση
n_layers	πλήθος* επιπέδων που χρησιμοποιεί το μοντέλο για σύγκλιση
layer_sizes	πλήθος* του αριθμού νευρώνων κάθε επιπέδου που χρησιμοποιεί το μοντέλο για σύγκλιση
l_window	πλήθος* των τιμών εισόδου που χρησιμοποιεί το μοντέλο για σύγκλιση
f_horizon	πλήθος* των τιμών εξόδου που χρησιμοποιεί το μοντέλο για σύγκλιση
l_rate	τιμή* του learning rate που χρησιμοποιεί το μοντέλο για σύγκλιση
activation	συνάρτηση ενεργοποίησης* που χρησιμοποιεί το μοντέλο για σύγκλιση
optimizer_name	όνομα optimizer* που χρησιμοποιεί το μοντέλο για σύγκλιση
batch_size	τιμή batch size* που χρησιμοποιεί το μοντέλο για σύγκλιση
transfer_mode	κατηγορία τεχνικής transfer learning που χρησιμοποιείται (βλέπε enum στο model_utils.py)
n_estimators	πλήθος estimators στο ensembling (βλέπε κεφάλαιο 5.2.5)
num_workers	πλήθος accelerators (GPUs) ή processors (CPU) που θα χρησιμοποιηθούν από το μοντέλο

Πίνακας 9.2: Πίνακας των παραμέτρων του pipeline και μιας ενδεικτικής περιγραφή τους. Αστερίσκο στην περιγραφή έχουν οι παράμετροι οι οποίες χρησιμοποιούνται με διαφορετικό format σε δύο entrypoints. Στην περίπτωση του train, λειτουργούν σαν σταθερές τιμές, ενώ στην περίπτωση του Optuna σαν εύρος τιμών που θα εξεταστεί στο tuning του μοντέλου

Country	Cluster no.	Benchmark	All-TL	Cluster-TL
Italy	1	2.807	2.879	2.411
Croatia	1	2.969	3.542	3.837
Spain	1	2.132	2.58	2.063
Greece	1	3.059	3.658	3.114
Serbia	2	2.918	3.386	2.749
Portugal	2	2.504	2.937	2.221
Belgium	2	2.461	3.327	2.664
Ireland	2	2.049	2.653	2.063
Netherlands	2	4.105	4.491	3.745
France	2	3.085	3.107	2.613
Hungary	2	3.347	3.521	2.944
Poland	2	2.657	2.882	2.22
Germany	3	2.574	3.471	2.45
Romania	3	2.27	2.809	2.226
Ukraine	3	2.747	4.106	4.106
Slovakia	3	2.378	2.727	2.111
Austria	3	3.434	3.748	2.862
Switzerland	3	4.27	4.308	4.451
Bulgaria	3	2.924	3.287	2.804
Slovenia	3	3.103	5.265	3.338
Norway	4	2.198	2.581	2.581
Denmark	4	2.909	3.527	3.527
Finland	4	2.62	2.821	2.821
Estonia	4	3.53	4.313	4.313
Czechia	4	2.08	2.526	2.895
Latvia	4	2.567	2.656	2.888
Sweden	4	3.055	3.289	3.621
Lithuania	4	2.818	3.102	3.247

Πίνακας 9.3: Πίνακας σύγκρισης απόδοσης μεταξύ των test cases

COUNTRIES	Benchmark			
	l_rate	l_window	layer_sizes	n_layers
Austria	0.00007	216	113	1
Belgium	0.00088	168	105,389,287	3
Bulgaria	0.00018	216	178	1
Croatia	0.00074	216	151	1
Czechia	0.00064	240	311,157	2
Denmark	0.00022	264	117,147	2
Estonia	0.00027	192	476	1
Finland	0.00003	312	313,122,385,135	4
France	0.00006	312	127	1
Germany	0.00004	336	469,331,289,326,260	5
Greece	0.00021	192	483	1
Hungary	0.00085	240	50,140,160,130	4
Ireland	0.00007	216	293,411	2
Italy	0.00005	336	434	1
Latvia	0.00001	312	343,448	2
Lithuania	0.00041	192	100,170,170,60	4
Netherlands	0.00024	168	170,160	2
Norway	0.00023	264	293	1
Poland	0.00067	192	160	1
Portugal	0.00005	240	244,100	2
Romania	0.00027	240	342	1
Serbia	0.00069	144	422	1
Slovakia	0.00005	312	202	1
Slovenia	0.00023	216	254,338,401	3
Spain	0.00008	168	420	1
Sweden	0.00048	168	444	1
Switzerland	0.00015	168	198	1
Ukraine	0.00093	48	446,249	2

Πίνακας 9.4: Στον πίνακα απεικονίζονται οι καλύτερες τιμές των υπερπαραμέτρων για το *Benchmark* test case

	All-TL			
COUNTRIES	l_rate	l_window	layer_sizes	n_layers
Austria	0.00025	264	150,200	2
Belgium	0.0006	168	190,190,200	3
Bulgaria	0.00023	168	130,120	2
Croatia	0.00015	240	170,140,120	3
Czechia	0.00042	216	200,150,200	3
Denmark	0.00091	216	160,110,110,90	4
Estonia	0.00025	288	160,150,100	3
Finland	0.00093	240	150,140	2
France	0.00026	336	160,120,100	3
Germany	0.00025	240	170,150,180	3
Greece	0.00023	168	130,120	2
Hungary	0.00002	192	127,272	2
Ireland	0.00026	264	200,200	2
Italy	0.00024	192	120,180,140,80	4
Latvia	0.00023	168	130,120	2
Lithuania	0.00002	192	127,272	2
Netherlands	0.0001	240	150,150,170	3
Norway	0.0003	240	90,190,130	3
Poland	0.00002	192	127,272	2
Portugal	0.00059	216	150,100	3
Romania	0.00018	168	124,16,383	3
Serbia	0.00049	168	140,190,170	3
Slovakia	0.0009	192	160,170	2
Slovenia	0.00023	264	154,100,239	3
Spain	0.00012	240	150,100,180,190	4
Sweden	0.00023	168	130,120	2
Switzerland	0.00023	168	130,120	2
Ukraine	0.00015	168	170,160,80	3

Πίνακας 9.5: Στον πίνακα απεικονίζονται οι καλύτερες τιμές των υπερπαραμέτρων για το *All-TL* test case

	Cluster-TL			
COUNTRIES	l_rate	l_window	layer_sizes	n_layers
Austria	0.00066	312	110,90	2
Belgium	0.00049	144	70,150,180	3
Bulgaria	0.00022	288	170,80	2
Croatia	0.00031	264	170,90,160	3
Czechia	0.00027	168	110,90,110	3
Denmark	0.00048	312	110,70,170,170	4
Estonia	0.00017	264	90,130,100,130,110,70,180,50	8
Finland	0.00011	192	110,200,90,200	4
France	0.00035	168	150,180	2
Germany	0.0002	192	450,384,177	3
Greece	0.00007	264	202,222,329	3
Hungary	0.00012	240	216	1
Ireland	0.00003	168	152,340	2
Italy	0.00003	168	237	1
Latvia	0.00015	192	218,209,468	3
Lithuania	0.00062	144	133,492,232	3
Netherlands	0.00032	216	311,234	2
Norway	0.00005	216	253	1
Poland	0.00036	336	160,170	2
Portugal	0.00025	168	141,207,138	3
Romania	0.00016	216	163344	2
Serbia	0.00006	216	474,232	2
Slovakia	0.00001	192	207	1
Slovenia	0.00005	168	304,280	2
Spain	0.0003	144	175,369	2
Sweden	0.00031	240	385,369	2
Switzerland	0.00018	288	459,159	2
Ukraine	0.00008	312	195,129,432,421	4

Πίνακας 9.6: Στον πίνακα απεικονίζονται οι καλύτερες τιμές των υπερπαραμέτρων για το *Cluster-TL* test case