



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

ΔΙΑΤΜΗΜΑΤΙΚΟ ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ:
ΕΠΙΣΤΗΜΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

Αναγνώριση Καθηκόντων Χειριστή στο Επίπεδο Παραγωγής
μέσω Αλγορίθμων Βαθιάς Μηχανικής Μάθησης

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Ανδρέας Ε. Μαστακούρης

Επιβλέπων: Δημήτριος Σούντρης
Καθηγητής Ε.Μ.Π

Αθήνα,
Ιούλιος 2021



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ
ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ
ΥΠΟΛΟΓΙΣΤΩΝ

ΔΙΑΤΜΗΜΑΤΙΚΟ ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ:
ΕΠΙΣΤΗΜΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

Αναγνώριση Καθηκόντων Χειριστή στο Επίπεδο Παραγωγής
μέσω Αλγορίθμων Βαθιάς Μηχανικής Μάθησης

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Ανδρέας Ε. Μαστακούρης

Επιβλέπων: Δημήτριος Σούντρης
Καθηγητής Ε.Μ.Π

Εγκρίθηκε από την τριμελή επιτροπή την 2021:

.....
Δ. Σούντρης
Καθηγητής Ε.Μ.Π

.....
Γ. Βοσνιάκος
Καθηγητής Ε.Μ.Π

.....
Π. Μπενάρδος
Επ. Καθηγητής Ε.Μ.Π

Αθήνα,
Ιούλιος 2021

.....

Ανδρέας Ε. Μαστακούρης

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

Copyright © Ανδρέας Ε. Μαστακούρης, 2021

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

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

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

Περίληψη

Η παρούσα εργασία μελετά την αναγνώριση των ανθρώπινων δραστηριοτήτων (Human Activity Recognition - HAR) στο επίπεδο παραγωγής, έτσι ώστε να επιτευχθεί αποτελεσματικά η συνεργασία ενός χειριστή με έναν οποιοδήποτε ρομποτικό μηχανισμό. Η συγκεκριμένη μελέτη παρουσιάζει αρκετές δυσκολίες, καθώς η χρονική διάρκεια των κινήσεων που εκτελεί ένας χειριστής διαφέρει και επιπλέον ο ρυθμός με τον οποίο πραγματοποιεί μία κίνηση δεν παραμένει σταθερός. Για τη μελέτη λοιπόν του συγκεκριμένου προβλήματος ένα smartphone προσδένεται στον καρπό 2 χειριστών, οι οποίοι στη συνέχεια πραγματοποιούν επαναληπτικά ένα μεγάλο αριθμό πειραμάτων εξετάζοντας 5 συγκεκριμένες κινήσεις. Οι κινήσεις αυτές είναι αρκετά δύσκολα διαχωρίσιμες μεταξύ τους και επιπλέον δε διαφέρουν αισθητά από άλλες συνήθεις κινήσεις μη ενδιαφέροντος που ένας εργαζόμενος μπορεί να εκτελεί.

Για τον επιτυχή διαχωρισμό των 5 εξεταζόμενων κινήσεων επιστρατεύονται 2 διαφορετικά είδη νευρωνικών δικτύων. Αρχικά, τα σήματα των πειραμάτων τροφοδοτούνται στα δίκτυα των αυτοκωδικοποιητών με σκοπό τη συμπίεση των δεδομένων αυτών και την κατάλληλη εξαγωγή χαρακτηριστικών. Οι αυτοκωδικοποιητές που χρησιμοποιούνται αποτελούν καινοτόμες τεχνικές συμπίεσης δεδομένων, οι οποίες μελετώνται διεξοδικά από αρκετούς ερευνητές το τελευταίο διάστημα. Στη συνέχεια, τα χαρακτηριστικά αυτά τροφοδοτούνται στο τελευταίο επίπεδο ενός δικτύου χρονοσειρών LSTM, το οποίο εκπαιδεύεται για τον αποτελεσματικό διαχωρισμό των εξεταζόμενων κινήσεων. Με τον τρόπο αυτό, παρόλο που μέσα από τη μη παραμετρική ανάλυση των σημάτων δυσχεραίνεται ο άμεσος διαχωρισμός των κινήσεων, η χρήση νευρωνικών δικτύων διευκολύνει τη διαδικασία αυτή αγγίζοντας πολύ υψηλά ποσοστά σωστής ταξινόμησης.

Τέλος, αντικείμενο της παρούσας εργασίας αποτελεί η μελέτη των σημάτων μακράς διάρκειας (ΣΜΔ). Τα σήματα αυτά αποτελούν χρονοσειρές πολύ μεγαλύτερης διάρκειας από αυτές των 5 εξεταζόμενων κινήσεων και επιπλέον περιλαμβάνουν αρκετό θόρυβο λόγω της ύπαρξης χρονικών περιόδων από δραστηριότητες μη ενδιαφέροντος. Για το λόγο αυτό προτείνεται μία νέα μεθοδολογία, η οποία συνδυάζει τα 2 είδη εκπαιδευμένων νευρωνικών δικτύων που αναφέρθηκαν προηγούμεως, με σκοπό την αναγνώριση των επιθυμητών κινήσεων εντός των σημάτων μακράς διάρκειας σε πραγματικό χρόνο. Βέβαια, θα πρέπει να τονισθεί ότι η συγκεκριμένη μεθοδολογία παρουσιάζεται για πρώτη φορά και μπορεί να χρησιμοποιηθεί σε οποιαδήποτε εφαρμογή συνεργασίας χειριστή-ρομπότ η οποία πραγματοποιείται με χρήση σημάτων από αισθητήρες.

Λέξεις Κλειδιά

Παρακολούθηση ανθρώπινης δραστηριότητας, συνεργασία χειριστή-ρομπότ, βαθιά μάθηση, νευρωνικά δίκτυα χρονοσειρών LSTM, δίκτυα αυτοκωδικοποιητών, σήματα μακράς διάρκειας

Abstract

The present study deals with Human Activity Recognition (HAR) at the production level, a technique used for the collaboration between a human and any robotic mechanism. This technique presents several difficulties, as the duration of the movements performed by any operator varies and in addition, the rate at which he performs a movement does not remain constant. The present work studies this problem by attaching a smartphone onto the wrist of 2 operators, who subsequently perform a large number of experiments examining 5 specific movements. These movements are quite difficult to be distinguished from each other and they also do not differ significantly from other ordinary hand movements of no interest.

For the successful separation of the 5 specific movements, 2 different types of neural networks are employed. Initially, the experimental signals are fed to an autoencoder neural network so as to compress the signals and extract the appropriate features. The autoencoders used are considered as innovative data compression techniques that have been extensively studied by several researchers in recent years. These features are then fed to the last layer of an LSTM time series network, which is subsequently trained to effectively separate the 5 movements. In this way, although the non-parametric analysis of the signals makes it difficult to directly separate the movements, the use of neural networks facilitates the whole process, yielding to a high performance of correct classification.

The final goal of the present work is the study of long-term signals (LTSs). These signals are time series of much longer duration than those of the 5 aforementioned movements and they also include noisy parts of signals derived from activities of no interest. For this reason, a new methodology is proposed combining the 2 types of neural networks that have already been trained, in order to detect the 5 specific movements within the long-term signals. Finally, it should be emphasized that this methodology is presented for the first time and can be used in any application of human-robot collaboration based on sensor signals.

Kew Words

Human activity recognition, human-robot collaboration, deep learning, LSTM time series neural networks, autoencoder neural network, long-term signals

Ευχαριστίες

Αρχικά, θα ήθελα να ευχαριστήσω το Ίδρυμα Μποδοσάκη για την εμπιστοσύνη που μου έδειξε επιλέγοντάς με ως υπότροφο του ιδρύματος. Η οικονομική στήριξη που μου παρείχε καθόλη τη διάρκεια του μεταπτυχιακού προγράμματος ήταν καθοριστική για την επιτυχή ολοκλήρωση των σπουδών μου.

Επιπλέον, θα ήθελα να εκφράσω τις θερμές μου ευχαριστίες στον καθηγητή κ. Δημήτριο Σούντρη, αφενός μεν για την ανάθεση του ιδιαίτερου αυτού θέματος και αφετέρου για τη στήριξη που μου παρείχε καθόλη τη διάρκεια εκπόνησης της διπλωματικής εργασίας. Ιδιαίτερα ευγνώμων νιώθω επίσης για τους καθηγητές κ. Γεώργιο Βοσνιάχο και κ. Πανώριο Μπενάρδο, όπως και για τον υποψήφιο διδάκτορα κ. Δημοσθένη Μασούρο για τις χρήσιμες συμβουλές που μου παρείχαν τόσο για τη διεξαγωγή των πειραμάτων όσο και για την εξαγωγή των αποτελεσμάτων της εργασίας.

Επιπροσθέτως, η συνεργασία με τη συμφοιτήριά μου Γεωργία Ανδριοσοπούλου ήταν ιδιαίτερα σημαντική για την ολοκλήρωση των σπουδών μου. Οι συμβουλές και η υποστήριξή της διευκόλυναν σε τεράστιο βαθμό τις σπουδές μου και βελτίωσαν το ενδιαφέρον μου για την εμβάθυνση στο συγκεκριμένο αντικείμενο.

Τέλος, θα ήθελα να ευχαριστήσω τους γονείς μου Ευθύμιο Μαστακούρη κι Ελπίδα Τσαγκαράκου για τη στήριξη και την υπομονή τους, όχι μόνο κατά την εκπόνηση της συγκεκριμένης διπλωματικής εργασίας, αλλά και καθόλη τη διάρκεια τόσο των προπτυχιακών όσο και των μεταπτυχιακών σπουδών μου.

Ανδρέας Μαστακούρης
Αθήνα, Ιούλιος 2021

Περιεχόμενα

| | |
|--------------------------------------------------------|-----------|
| Περιεχόμενα | vii |
| Κατάλογος Σχημάτων | ix |
| Κατάλογος Πινάκων | xi |
| 1 Εισαγωγή | 1 |
| 1.1 Το Πρόβλημα και η Σημασία του | 2 |
| 1.2 Ανασκόπηση Βιβλιογραφίας | 3 |
| 1.3 Προτεινόμενη Μεθοδολογία και Συνεισφορές | 4 |
| 1.4 Οργάνωση της Εργασίας | 5 |
| 2 Νευρωνικά Δίκτυα Χρονοσειρών | 7 |
| 2.1 Αναδρομικά Νευρωνικά Δίκτυα RNN | 8 |
| 2.1.1 Υπολογιστικοί Γράφοι | 8 |
| 2.1.2 Αρχιτεκτονική Αναδρομικών Νευρωνικών Δικτύων | 9 |
| 2.1.3 Εκπαίδευση Αναδρομικών Νευρωνικών Δικτύων | 10 |
| 2.2 Το Πρόβλημα Μακροπρόθεσμων Εξαρτήσεων | 12 |
| 2.2.1 Δίκτυα Μακράς και Βραχείας Μνήμης LSTM | 13 |
| 2.3 Μέθοδοι Βελτιστοποίησης | 15 |
| 2.3.1 Αλγόριθμος Καθόδου Κλίσης | 15 |
| 2.3.2 Κάθοδος Κλίσης με Ορμή | 15 |
| 3 Νευρωνικά Δίκτυα Αυτοκωδικοποιητών | 17 |
| 3.1 LSTM Αυτοκωδικοποιητές | 18 |
| 3.2 Variational Αυτοκωδικοποιητές | 19 |
| 3.2.1 Πιθανοτική Μέθοδος Κύριων Συνιστωσών | 19 |
| 3.2.2 Μη Γραμμικός Μετασχηματισμός Δεδομένων | 22 |
| 3.3 Εκπαίδευση Μοντέλων VAE | 26 |
| 3.3.1 Διανυσματική Παράγωγος ως προς θ | 26 |
| 3.3.2 Διανυσματική Παράγωγος ως προς ϕ | 27 |
| 4 Προηγμένη Variational Συμπερασματολογία | 31 |
| 4.1 Semi-Amortized Συμπερασματολογία | 32 |
| 4.1.1 Στοχαστική Variational Συμπερασματολογία | 32 |
| 4.1.2 Υβριδική Μέθοδος AVI-SVI | 33 |
| 4.2 Λαπλασιανή Προσέγγιση | 34 |
| 4.2.1 Δίκτυα Piece-wise Linear | 34 |
| 4.2.2 Εκ των Υστέρων Συμπερασματολογία για ReLU Δίκτυα | 35 |
| 4.2.3 Variational Λαπλασιανοί Αυτοκωδικοποιητές | 37 |
| 4.2.4 Σύνδεση Λαπλασιανής Προσέγγισης με ReLU Δίκτυα | 38 |
| 4.3 Γραφική Αναπαράσταση Προηγμένων Μοντέλων | 39 |

| | |
|--------------------------------------------------------------------------|-----------|
| 5 Κανονικοποιημένες Ροές | 41 |
| 5.1 Εισαγωγικές Έννοιες στις Ροές | 42 |
| 5.2 Συνδυασμός Ροών με Μοντέλα VAE | 44 |
| 5.3 Κανονικοποιημένη Ροή Householder | 46 |
| 6 Περιγραφή Πειραμάτων | 49 |
| 6.1 Παρουσίαση Κινήσεων Ενδιαφέροντος | 50 |
| 6.2 Βήματα Εκτέλεσης Κινήσεων | 51 |
| 6.3 Μη Παραμετρική Ανάλυση Σημάτων | 54 |
| 7 Παρουσίαση Αποτελεσμάτων | 57 |
| 7.1 Αποτελέσματα από Σήματα Κινήσεων | 58 |
| 7.1.1 Εξαγωγή Χαρακτηριστικών από Σήματα Κινήσεων | 58 |
| 7.1.2 Ταξινόμηση Σημάτων Κινήσεων | 63 |
| 7.2 Αποτελέσματα από Σήματα Συνιστωσών Κινήσεων | 65 |
| 7.2.1 Εξαγωγή Χαρακτηριστικών από Σήματα Φάσεων | 65 |
| 7.2.2 Ταξινόμηση Σημάτων Συνιστωσών Κινήσεων | 69 |
| 7.3 Μελέτη Σημάτων Μακράς Διάρκειας | 69 |
| 7.3.1 Αλγόριθμος Απόφασης Αναγνώρισης Κινήσεων | 71 |
| 7.3.2 Αποτελέσματα Αλγορίθμου Απόφασης | 72 |
| 8 Μελέτη Απόκρισης Υπολογιστικών Πόρων | 77 |
| 8.1 Σύγχρονοι Τρόποι Συμπερασματολογίας | 78 |
| 8.1.1 Edge Computing | 78 |
| 8.1.2 Edge και Cloud Offloading | 79 |
| 8.2 Πειραματικό Πλαίσιο | 80 |
| 8.3 Πειραματικά Αποτελέσματα | 80 |
| 8.3.1 Συμπερασματολογία σε Πραγματικό Χρόνο | 81 |
| 8.3.2 Κατανάλωση Ισχύος Αρχιτεκτονικών | 84 |
| 8.4 Βελτιστοποιήσεις για Συμπερασματολογία σε Πραγματικό Χρόνο | 85 |
| 8.4.1 Μέθοδοι Pruning Μοντέλων LSTM | 85 |
| 8.4.2 Αποτελέσματα Μεθόδων Pruning | 86 |
| 8.5 Μελλοντικοί στόχοι | 87 |
| 8.5.1 Υλοποίηση Δικτύων σε FPGA | 87 |
| 8.5.2 Quantization και Approximate Computing | 88 |
| 9 Επίλογος | 89 |
| 9.1 Γενικά Συμπεράσματα | 90 |
| 9.2 Μελλοντική Εργασία | 91 |
| Βιβλιογραφικές Αναφορές | 93 |

Κατάλογος Σχημάτων

| | | |
|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| 2.1 | Αναδρομικό δίκτυο χωρίς δεδομένα εξόδου. Στο αριστερό τμήμα αναπαριστάται ο αναδρομικός υπολογιστικός γράφος, ενώ στο δεξί τμήμα ο ξεδιπλωμένος μη κυκλικός υπολογιστικός γράφος. Πηγή [26]. | 9 |
| 2.2 | Υπολογιστικός γράφος που απεικονίζει την αρχιτεκτονική ενός αναδρομικού δικτύου και τον τρόπο εκτίμησης της συνάρτησης κόστους. Πηγή [26]. | 9 |
| 2.3 | Αρχιτεκτονική κελιού του νευρωνικού δικτύου LSTM. Πηγή [33]. | 13 |
| 3.1 | Αρχιτεκτονική αυτοκωδικοποιητή σε συνδυασμό με τα νευρωνικά δίκτυα LSTM, χρησιμοποιώντας τη ξεδιπλωμένη μορφή του μη κυκλικού υπολογιστικού γράφου. Πηγή [44]. | 18 |
| 3.2 | Στατιστική μέθοδος κύριων συνιστωσών. | 20 |
| 3.3 | Αρχιτεκτονική variational αυτοκωδικοποιητή. | 25 |
| 3.4 | Reparametrization trick κατά την εκτίμηση των παραμέτρων ϕ του νευρωνικού δικτύου VAE. Πηγή [50]. | 28 |
| 3.5 | Αρχιτεκτονική variational αυτοκωδικοποιητή σε συνδυασμό με τα νευρωνικά δίκτυα LSTM. Στο τμήμα του κωδικοποιητή και στον αποκωδικοποιητή χρησιμοποιείται η ξεδιπλωμένη μορφή του μη κυκλικού υπολογιστικού γράφου. | 29 |
| 4.1 | Αρχιτεκτονική variational αυτοκωδικοποιητή σε συνδυασμό με τα νευρωνικά δίκτυα LSTM στο τμήμα του κωδικοποιητή και τα ReLU πολυστρωματικά δίκτυα εμπρόσθιας τροφοδότησης MLP στο τμήμα του αποκωδικοποιητή. | 40 |
| 5.1 | Αρχιτεκτονική variational αυτοκωδικοποιητή σε συνδυασμό τη χρήση των κανονικοποιημένων ροών στο μεσαίο επίπεδο. | 46 |
| 5.2 | Αντικατοπτρική περιστροφή διανύσματος \mathbf{z}_t ως προς ένα υπερεπίπεδο με χρήση του ορθοκανονικού πίνακα Householder. | 47 |
| 5.3 | Σχηματική αναπαράσταση της ροής Householder σε συνδυασμό με τον αυτοκωδικοποιητή VAE. Πηγή [61]. | 48 |
| 5.4 | Αρχιτεκτονική variational αυτοκωδικοποιητή σε συνδυασμό με τα νευρωνικά δίκτυα LSTM και την κανονικοποιημένη ροή Householder. | 48 |
| 6.1 | Προσάρτηση smartphone στον καρπό των 2 χειριστών για τη διεξαγωγή των 5 κινήσεων. | 51 |
| 6.2 | Παρουσίαση 4 φάσεων κατά την εκτέλεση του βιδώματος με χρήση γερμανικού κλειδιού. | 51 |
| 6.3 | Παρουσίαση 4 φάσεων κατά την εκτέλεση του βιδώματος με χρήση κατσαβιδιού. | 52 |
| 6.4 | Παρουσίαση 3 φάσεων κατά την εκτέλεση του σηκώματος και μεταφοράς ενός τεμαχίου. | 52 |
| 6.5 | Παρουσίαση 3 φάσεων κατά την εκτέλεση της εναπόθεσης ενός τεμαχίου. | 53 |
| 6.6 | Παραδείγματα πειραμάτων καθαρισμού επιφάνειας. | 53 |
| 6.7 | Σήματα επιτάχυνσης του άξονα Y για κάθε μία από τις 5 εξεταζόμενες κινήσεις. | 54 |
| 6.8 | Σύγκριση του εύρους τιμών της επιτάχυνσης Y για τις 5 κινήσεις. | 55 |
| 6.9 | Σύγκριση του εύρους τιμών της συνολικής επιτάχυνσης για τις 5 κινήσεις. | 55 |

| | | |
|------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| 7.1 | Ανακατασκευή επιτάχυνσης X για 8 τυχαία σήματα κινήσεων με χρήση των δικτύων HF-V-LSTM-AE (πρώτη), VL-LSTM/MLP-AE (δεύτερη) και SAV-LSTM/MLP-AE (τρίτη γραμμή). | 61 |
| 7.2 | 2D manifold της συμπιεσμένης αναπαράστασης z από τα σήματα κινήσεων βάσει των δικτύων HF-V-LSTM-AE (αριστερά), VL-LSTM/MLP-AE (μέση) και SAV-LSTM/MLP-AE (δεξιά). | 62 |
| 7.3 | Τιμή ποσότητας -ELBO του συνόλου επικύρωσης (validation set) κατά τη διάρκεια των 500 εποχών εκπαίδευσης των σημάτων κινήσεων, βάσει των μοντέλων HF-V-LSTM-AE (αριστερά), VL-LSTM/MLP-AE (μέση) και SAV-LSTM/MLP-AE (δεξιά). | 63 |
| 7.4 | Αρχιτεκτονική δικτύου LSTM σε συνδυασμό με τον αυτοκωδικοποιητή HF-V-LSTM-AE για την ταξινόμηση των 5 εξεταζόμενων κινήσεων. | 64 |
| 7.5 | Αξιολόγηση ταξινόμησης των 5 κινήσεων του δικτύου LSTM με χρήση καμπύλης ROC (αριστερά) και confusion matrix (δεξιά). | 65 |
| 7.6 | Ανακατασκευή επιτάχυνσης X για 8 τυχαία σήματα φάσεων με χρήση των δικτύων HF-V-LSTM-AE (πρώτη), VL-LSTM/MLP-AE (δεύτερη) και SAV-LSTM/MLP-AE (τρίτη γραμμή). | 67 |
| 7.7 | 2D manifold της συμπιεσμένης αναπαράστασης z από τα σήματα φάσεων βάσει των δικτύων HF-V-LSTM-AE (αριστερά), VL-LSTM/MLP-AE (μέση) και SAV-LSTM/MLP-AE (δεξιά). | 68 |
| 7.8 | Τιμή ποσότητας -ELBO του συνόλου επικύρωσης (validation set) κατά τη διάρκεια των 500 εποχών εκπαίδευσης των σημάτων φάσεων, βάσει των μοντέλων HF-V-LSTM-AE (αριστερά), VL-LSTM/MLP-AE (μέση) και SAV-LSTM/MLP-AE (δεξιά). | 68 |
| 7.9 | Αξιολόγηση ταξινόμησης των 15 συνιστωσών κινήσεων του δικτύου LSTM με χρήση καμπύλης ROC (αριστερά) και confusion matrix (δεξιά). | 69 |
| 7.10 | Διάσπαση ενδεικτικού σήματος μακράς διάρκειας σε N επιμέρους χρονοσειρές με χρήση αλληλεπικαλυπτόμενων παραθύρων με ποσοστό επικάλυψης 95%. | 70 |
| 7.11 | Ταξινόμηση σημάτων μακράς διάρκειας (ΣΜΔ) χρησιμοποιώντας την 1 ^η τεχνική επεικούς αξιολόγησης των αποτελεσμάτων. | 74 |
| 7.12 | Ταξινόμηση σημάτων μακράς διάρκειας (ΣΜΔ) χρησιμοποιώντας την 2 ^η τεχνική αυστηρής αξιολόγησης των αποτελεσμάτων. | 76 |
| 8.1 | Σύγκριση τεχνικών edge και cloud computing. | 79 |
| 8.2 | Σύγκριση χρόνων συμπερασματολογίας ανά παράθυρο για τους 4 υπολογιστικούς πόρους. | 83 |
| 8.3 | Σύγκριση κατανάλωσης ισχύος (σε Watt) μεταξύ των 4 υπολογιστικών πόρων και για τα 50 σήματα μακράς διάρκειας. | 84 |
| 8.4 | Σύγκριση μέσων χρόνων συμπερασματολογίας ανά παράθυρο για διάφορα ποσοστά μείωσης των παραμέτρων. | 87 |

Κατάλογος Πινάκων

| | | |
|-----|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| 1.1 | Κεφάλαια διπλωματικής εργασίας. | 6 |
| 6.1 | Εξεταζόμενες κινήσεις χειριστή στο επίπεδο παραγωγής. | 50 |
| 7.1 | Υπερπαράμετροι αυτοκωδικοποιητών για τη συμπίεση σημάτων των 5 κινήσεων. . . | 59 |
| 7.2 | Τιμές λογαριθμοποιημένης πιθανοφάνειας για τους αυτοκωδικοποιητές των 5 κινήσεων. . . | 60 |
| 7.3 | Βασικές υπερπαράμετροι δικτύου LSTM κατά την ταξινόμηση των 5 κινήσεων. . . | 64 |
| 7.4 | Υπερπαράμετροι αυτοκωδικοποιητών για τη συμπίεση σημάτων των 15 φάσεων. . . | 66 |
| 7.5 | Λογαριθμοποιημένη πιθανοφάνεια για τους αυτοκωδικοποιητές των 15 φάσεων. . . | 66 |
| 7.6 | Πρόβλεψη κινήσεων σε σήματα μακράς διάρκειας με χρήση των νευρωνικών LSTM και HF-V-LSTM-AE. Οι λανθασμένα εμφανιζόμενες προβλέψεις αναγράφονται με bold χαρακτήρες, ενώ οι υπογραμμισμένες προβλέψεις βρίσκονται σε λανθασμένη χρονική σειρά. | 72 |
| 7.7 | Αποτελέσματα ταξινόμησης των 5 κινήσεων σε 50 ΣΜΔ με χρήση των 2 τεχνικών αξιολόγησης της προτεινόμενης μεθοδολογίας. | 76 |
| 8.1 | Χρόνοι συμπερασματολογίας παραθύρων με χρήση του τοπικού server. | 82 |
| 8.2 | Χρόνοι συμπερασματολογίας παραθύρων με χρήση του cloud server. | 82 |
| 8.3 | Χρόνοι συμπερασματολογίας παραθύρων με χρήση των AGX και NX. | 83 |
| 8.4 | Αποτελέσματα μείωσης παραμέτρων των δικτύων LSTM του Κεφαλαίου 7. | 86 |

Κεφάλαιο 1

Εισαγωγή

Το παρόν Κεφάλαιο αποτελεί την εισαγωγή της παρούσας εργασίας και έχει ως στόχο να εισάγει τον αναγνώστη στις βασικές έννοιες του προβλήματος που πρόκειται να αντιμετωπιστεί. Συγκεκριμένα, στην Εν. 1.1 εξηγείται το πρόβλημα της παρακολούθησης των ανθρώπινων δραστηριοτήτων στο επίπεδο παραγωγής και τονίζεται η σημασία της επίλυσής του. Στην Εν. 1.2 πραγματοποιείται μία βιβλιογραφική ανασκόπηση των εργασιών που αφορούν το συγκεκριμένο αντικείμενο και επισημαίνονται ορισμένες ελλείψεις των έως τώρα προτεινόμενων μεθοδολογιών. Στην Εν. 1.3 αναλύεται συνοπτικά η μεθοδολογία της παρούσας εργασίας και επισημαίνονται οι συνεισφορές της συγκριτικά με άλλες προσεγγίσεις που έχουν προταθεί για το συγκεκριμένο αντικείμενο. Τέλος, στην Εν. 1.4 παρουσιάζεται η δομή του συγκεκριμένου τόμου.

1.1 Το Πρόβλημα και η Σημασία του

Η αναγνώριση των ανθρώπινων δραστηριοτήτων (Human Activity Recognition - HAR) αποτελεί ένα πρόβλημα που μελετάται διεξοδικά τα τελευταία χρόνια λόγω της μεγάλης βοήθειας που μπορεί να προσφέρει σε αρκετούς τομείς. Χαρακτηριστικό παράδειγμα αποτελεί η αναγνώριση των καθηκόντων ενός εργαζόμενου σε μια γραμμή παραγωγής [1–3], ενώ άλλα παραδείγματα μπορούν να παρατηρηθούν σε εφαρμογές παρακολούθησης της υγείας και της άθλησης των ανθρώπων [4–6], στη χρήση των κοινωνικών δικτύων [7, 8] κλπ.

Η τεχνική HAR πραγματοποιείται συνήθως με 3 διαφορετικούς τρόπους: (1) με χρήση βίντεο και εικόνων, (2) με χρήση σημάτων από αισθητήρες και (3) με χρήση ραδιοσυχνοτήτων. Στην πρώτη περίπτωση η αναγνώριση της ανθρώπινης διαστηριότητας πραγματοποιείται ύστερα από κατάλληλη ανάλυση διαφόρων εικόνων ή βίντεο από μία ή περισσότερες κάμερες, ενώ στη δεύτερη περίπτωση επιστρατεύονται διάφορα επιταχυνσιόμετρα ή γυροσκόπια που τοποθετούνται σε διαφορετικά σημεία του σώματος και καταγράφουν σήματα κατά τη διάρκεια των δραστηριοτήτων. Τέλος, στην τρίτη περίπτωση το ανθρώπινο σώμα κινείται εντός ενός πεδίου ραδιοσυχνοτήτων και ως εκ τούτου μελετώνται οι εναλλαγές του πεδίου κατά τη διάρκεια των διαφόρων κινήσεων.

Η παρούσα εργασία ασχολείται με την αναγνώριση κίνησης ενός χειριστή στο επίπεδο παραγωγής, έτσι ώστε να μελετηθεί η συνεργασία του εργαζόμενου αυτού με έναν οποιοδήποτε ρομποτικό μηχανισμό. Η μελέτη αυτή θεωρείται μία σύνθετη διαδικασία, καθώς το πλήθος των διαφορετικών κινήσεων του χειριστή είναι αρκετά μεγάλο, η χρονική διάρκεια των κινήσεων αυτών διαφέρει, ο ρυθμός με τον οποίο πραγματοποιεί μία κίνηση δεν παραμένει σταθερός κ.α. Παρόλες όμως τις δυσκολίες αυτές το συγκεκριμένο αντικείμενο θεωρείται αρκετά ενδιαφέρον, καθώς μέσα από τη συνεργασία χειριστή-ρομπότ δύναται να ελαττωθεί αρκετά ο χρόνος παραγωγής των προϊόντων και κατά συνέπεια να προκύψει μείωση στο συνολικό κόστος της επιχείρησης.

Η συνεργασία χειριστή-ρομπότ πραγματοποιείται συνήθως με τους 2 πρώτους τρόπους που αναφέρθηκαν προηγουμένως, ήτοι με χρήση κάμερας ή με τη βοήθεια αισθητήρων που καταγράφουν σήματα από διάφορα σημεία του σώματος. Στην παρούσα εργασία επιλέγεται να μελετηθεί η δεύτερη περίπτωση, η οποία εξετάζεται διεξοδικά από αρκετούς ερευνητές το τελευταίο διάστημα [9, 10]. Ορισμένοι λόγοι για την προτίμηση των αισθητήρων είναι η εύκολη και γρήγορη μεταφορά τους, το μικρό υπολογιστικό κόστος κατά την ανάλυση των σημάτων έναντι των βίντεο ή των εικόνων και η ασύρματη σύνδεση της συσκευής καταγραφής των σημάτων με τον ρομποτικό μηχανισμό [11].

Σε γενικές γραμμές, κατά την τεχνική HAR τα βήματα που ακολουθούνται έως ότου εξαχθούν πληροφορίες για τις δραστηριότητες ενός εργαζόμενου παρουσιάζονται συνοπτικά ακολούθως:

1. Προεπεξεργασία Δεδομένων: Στο στάδιο της προεπεξεργασίας (preprocessing) απομακρύνονται τα ελλιπή δεδομένα, εξαλείφεται ο θόρυβος των σημάτων και συνήθως πραγματοποιείται κανονικοποίησή τους για διευκόλυνση της ανάλυσης που θα ακολουθήσει.
2. Τμηματοποίηση: Στο στάδιο της τμηματοποίησης (segmentation) απομονώνονται τα πιο σημαντικά τμήματα των σημάτων για την μετέπειτα επεξεργασία τους.
3. Εξαγωγή Χαρακτηριστικών: Στο στάδιο αυτό εξάγονται τα απαραίτητα χαρακτηριστικά (feature extraction) από τα τμήματα των σημάτων του προηγούμενου βήματος. Αν το πλήθος των χαρακτηριστικών αυτών είναι αρκετά μεγάλο, τότε μπορούν να χρησιμοποιηθούν μέθοδοι μείωσης της διαστατικότητάς τους (dimensionality reduction).
4. Μηχανική Μάθηση: Στο στάδιο αυτό επιστρατεύονται αλγόριθμοι μηχανική μάθησης (machine learning), με σκοπό την ταξινόμηση των χαρακτηριστικών του προηγούμενου βήματος στις εξεταζόμενες δραστηριότητες.

Στις επόμενες ενότητες λοιπόν πραγματοποιείται αρχικά μία ανασκόπηση των ερευνητικών δημοσιεύσεων που αφορούν τη συνεργασία εργαζόμενου-ρομπότ στο περιβάλλον εργασίας, ενώ ακολούθως παρουσιάζεται η προτεινόμενη μεθοδολογία στην οποία βασίζεται η παρούσα μελέτη.

1.2 Ανασκόπηση Βιβλιογραφίας

Η χρήση ρομποτικών μηχανισμών στο περιβάλλον εργασίας και η συνεργασία τους με τους εργαζόμενους μελετάται διεξοδικά από αρκετούς ερευνητές, καθώς όπως προαναφέρθηκε μπορεί να προσφέρει αρκετά πλεονεκτήματα όπως η μείωση του φυσικού και γνωστικού φορτίου της εργασίας, η μείωση του χρόνου παραγωγής προϊόντων και η μείωση του κόστους της επιχείρησης [12, 13]. Παρόλο όμως τα πλεονεκτήματα που προσφέρει, η συνεργασία αυτή αποτελεί μία αρκετά σύνθετη και πολύπλοκη διαδικασία. Αυτό συμβαίνει διότι η λειτουργία κάθε ρομποτικού μηχανισμού προϋποθέτει την παρακολούθηση και την ακριβή κατανόηση των κινήσεων του εργαζόμενου, έτσι ώστε στη συνέχεια να μπορέσει να λειτουργήσει αυτόνομα χωρίς να παρεμποδίζει τις δραστηριότητες των ανθρώπινου δυναμικού. Ακολούθως λοιπόν παρουσιάζονται ορισμένες ερευνητικές εργασίες που αφορούν την παρακολούθηση της ανθρώπινης δραστηριότητας στο επίπεδο παραγωγής, οι μέθοδοι των οποίων μπορούν να χρησιμοποιηθούν για την επιτυχή συνεργασία χειριστή-ρομπότ στο περιβάλλον εργασίας.

Οι ερευνητές στο [14] παρουσίασαν μία καινοτόμο μέθοδο για την αναγνώριση των ανθρώπινων δραστηριοτήτων κατά τη διάρκεια συναρμολόγησης ποδηλάτων. Συγκεκριμένα, χρησιμοποίησαν αισθητήρες τύπου IMU και υπερήχων για την καταγραφή των σημάτων και προσπάθησαν να ταξινομήσουν 21 διαφορετικές δραστηριότητες. Παράλληλα, κατά τη διάρκεια των πειραμάτων κατέγραφαν σήματα από άσχετες δραστηριότητες και τις αντιστοίχισαν σε μία συγκεκριμένη κίνηση τύπου NULL, έτσι ώστε να δημιουργηθεί μία επιπλέον κλάση που ενσωματώνει όλες τις κινήσεις μη ενδιαφέροντος. Τέλος, χρησιμοποιώντας ως μέθοδο ταξινόμησης τα κρυφά Μαρκοβιανά μοντέλα (Hidden Markov Models) κατέληξαν στο συμπέρασμα ότι τα αποτελέσματα δεν είναι πάντα ενθαρρυντικά, καθώς το ποσοστό σωστής ταξινόμησης των επιθυμητών κινήσεων εξαρτάται από τις ομοιότητες που αυτές παρουσιάζουν με τις άσχετες δραστηριότητες της κλάσης NULL. Συνεπώς, το μειονέκτημα της συγκεκριμένης προσέγγισης είναι ότι στην κλάση NULL δεν μπορούν να ενσωματωθούν όλες οι πιθανές δραστηριότητες τις οποίες δεν επιθυμούμε να μελετήσουμε και συνεπώς η λογική αυτή μπορεί να μειώσει αισθητά την απόδοση των εξεταζόμενων μεθόδων.

Ορισμένοι από τους ερευνητές της παραπάνω μελέτης προσπάθησαν να επεκτείνουν την τεχνική HAR κατά τον έλεγχο της συναρμολόγησης αυτοκινήτου [15, 16]. Συγκεκριμένα, 8 διαφορετικοί άνθρωποι εκτέλεσαν 3800 πειράματα από 6 διαφορετικές δραστηριότητες, με αποτέλεσμα την καταγραφή σημάτων με διάρκεια έως και 480 λεπτά. Τα σήματα αυτά καταγράφηκαν από αρκετά διαφορετικά σημεία του σώματος των εργαζόμενων και στη συνέχεια εφαρμόστηκαν: (1) κατάλληλες μέθοδοι τμηματοποίησης τους, έτσι ώστε να επιτευχθεί η απομόνωση των σημαντικών τμημάτων από τα σήματα και (2) κατάλληλες μέθοδοι συνένωσης (fusion) των τμημάτων αυτών και (3) εφαρμογή αλγορίθμου ταξινόμησης στα συνενωμένα τμήματα. Παρόλο που η μελέτη αυτή είναι αρκετά πολύπλοκη, η προαναφερθείσα μεθοδολογία οδήγησε σε αρκετά υψηλά ποσοστά σωστής ταξινόμησης των επιθυμητών κινήσεων.

Κατά αντιστοιχία των παραπάνω μελετών, οι συγγραφείς στο [17] εφάρμοσαν την τεχνική HAR σε μία γραμμή παραγωγής μελετώντας 5 διαφορετικές δραστηριότητες. Συγκεκριμένα, προσαρμόζοντας IMU αισθητήρες στους καρπούς των εργαζόμενων κατέγραφαν σήματα επιτάχυνσης και γωνιακής ταχύτητας των χεριών τους, προσπαθώντας να ερμηνεύσουν αν έχει ολοκληρωθεί ο αριθμός των απαραίτητων διεργασιών στο επίπεδο παραγωγής. Οι δραστηριότητες που μελετήθηκαν ήταν βίδωμα με κατσαβίδι, βίδωμα με γερμανικό κλειδί, χρήση ηλεκτρικού τρυπανιού, χτύπημα με σφυρί και τέλος προστέθηκαν διάφορες δραστηριότητες αδράνειας των εργαζόμενων (π.χ. αλλαγή εργαλείων, ξεκούραση, ομιλία με άλλους ανθρώπους κ.α.), οι οποίες θεωρήθηκαν μία ξεχωριστή κατηγορία. Όσον αφορά το στάδιο της ταξινόμησης, η έρευνα βασίστηκε στον αλγόριθμο του k-κοντινότερου γείτονα και οδήγησε σε πολύ υψηλά ποσοστά σωστής ταξινόμησης των δραστηριοτήτων. Βέβαια, το μειονέκτημα της συγκεκριμένης μελέτης είναι ότι αρκετές από τις εξεταζόμενες κινήσεις είναι εύκολα ανιχνεύσιμες (π.χ. χρήση ηλεκτρικού τρυπανιού ή χτύπημα με σφυρί), γεγονός που ίσως οδήγησε στα αρκετά υψηλά ποσοστά απόδοσης.

Πριν την εφαρμογή των αλγορίθμων ταξινόμησης, αρκετές μελέτες στηρίζονται σε μεθόδους μείωσης της διαστατικότητας των χαρακτηριστικών με σκοπό την ταυτόχρονη μείωση του υπολογιστικού κόστους. Χαρακτηριστικό παράδειγμα αποτελούν οι μελέτες [18, 19], όπου οι συγγραφείς επιστράτευαν μεθόδους μείωσης διαστατικότητας των χαρακτηριστικών που προέκυψαν από τα σήματα παρακολούθησης της ανθρώπινης δραστηριότητας. Η πιο συχνή μέθοδος που χρησιμοποιείται ευρέως είναι η μέθοδος των Κύριων Συνιστωσών (PCA), η οποία αποτελεί έναν αλγόριθμο γραμμικού μετασχηματισμού των χαρακτηριστικών σε ένα περιστραμμένο επίπεδο και διατήρηση εκείνων που παρέχουν την περισσότερη πληροφορία στο επίπεδο αυτό. Το μειονέκτημα αυτής της μεθόδου είναι ότι σε αρκετές περιπτώσεις, λόγω της μεγάλης πολυπλοκότητας των χαρακτηριστικών, ο γραμμικός μετασχηματισμός τους δεν επαρκεί και συνεπώς απαιτείται η χρήση μη γραμμικών μετασχηματισμών (π.χ. με χρήση νευρωνικών δικτύων) για την αποτελεσματικότερη μείωση της διαστατικότητας των αρχικών δεδομένων.

Τέλος, αρκετές μελέτες χρησιμοποιούν μεθόδους βαθιάς μηχανικής μάθησης, ήτοι νευρωνικά δίκτυα, για την αναγνώριση των ανθρώπινων δραστηριοτήτων [20–23]. Ενδιαφέρουσα μελέτη αποτελεί η [24], στην οποία οι ερευνητές χρησιμοποίησαν σήματα τύπου IMU και sEMG και με εφαρμογή ενός αλγορίθμου μετέτρεψαν τα σήματα αυτά σε εικόνες. Στη συνέχεια, εφάρμοσαν μετασχηματισμό Fourier στις εικόνες αυτές και τις τροφοδότησαν σε συνελκτικά δίκτυα (CNN) για την κατηγοριοποίηση 6 διαφορετικών δραστηριοτήτων στο επίπεδο παραγωγής. Το μειονέκτημα της συγκεκριμένης μελέτης είναι ότι στο στάδιο ελέγχου περιλαμβάνονται σήματα μόνο από τις εξεταζόμενες κινήσεις και συνεπώς δεν έχει εξετασθεί με ποιόν τρόπο μπορούν να απορριφθούν κινήσεις μη ενδιαφέροντος, οι οποίες δεν περιλαμβάνονται στο στάδιο εκπαίδευσης.

Συγκεντρωτικά λοιπόν, οι έως τώρα προτεινόμενες μεθοδολογίες που στηρίζονται στη χρήση σημάτων από αισθητήρες για την παρακολούθηση των ανθρώπινων δραστηριοτήτων στο επίπεδο παραγωγής παρουσιάζουν τα εξής 2 συνήθη μειονεκτήματα: (1) οι κινήσεις που μελετώνται στις περισσότερες περιπτώσεις είναι εύκολα διαχωρίσιμες (π.χ. χτύπημα με σφυρί κ.α.) και (2) εισάγουν μία επιπλέον κλάση NULL για κινήσεις μη ενδιαφέροντος, με αποτέλεσμα να δημιουργείται μία γενικευμένη κλάση που σε αρκετές περιπτώσεις δυσχεραίνει τα αποτελέσματα. Στην παρούσα εργασία επιδιώκεται λοιπόν η αντιμετώπιση των 2 προαναφερθέντων προβλημάτων μελετώντας αρχικά 5 κινήσεις δύσκολα διαχωρίσιμες μεταξύ τους, ενώ στη συνέχεια παρουσιάζεται μία νέα μεθοδολογία για τον εντοπισμό κινήσεων μη ενδιαφέροντος σε σήματα μεγάλης διάρκειας. Η μεθοδολογία αυτή και οι συνεισφορές της εργασίας παρουσιάζονται συνοπτικά στην Εν. 1.3.

1.3 Προτεινόμενη Μεθοδολογία και Συνεισφορές

Σε συστήματα παραγωγής τα οποία απαιτούν τη συμμετοχή εργαζομένων, η παρακολούθηση και ακολούθως η αναγνώριση των δραστηριοτήτων τους κατά τη διάρκεια εργασίας τους μπορούν να χρησιμοποιηθούν αφενός μεν για την αξιολόγηση των εργαζομένων και αφετέρου για τη βελτίωση των υπηρεσιών που παρέχουν μέσα από την επιτήρηση των δραστηριοτήτων τους. Στα πλαίσια αυτής της ερευνητικής δραστηριότητας, η παρούσα εργασία ασχολείται με το αντικείμενο της αναγνώρισης διαφορετικού τύπου καθηκόντων χειριστή στο επίπεδο της παραγωγής, με σκοπό τη συνεργασία του εργαζόμενου αυτού με έναν οποιοδήποτε ρομποτικό μηχανισμό.

Για το σκοπό αυτό επιλέγεται η μελέτη 5 διαφορετικών δραστηριοτήτων που θα μπορούσαν να διαδραματίζονται σε περιβάλλον παραγωγής από κάποιον χειριστή, κατά τη διάρκεια των οποίων λαμβάνονται κατάλληλα σήματα χρονοσειρών με χρήση αισθητήρων. Η καταγραφή των σημάτων επιλέγεται να πραγματοποιηθεί με χρήση της εφαρμογής [Science Journal](#), στην οποία μπορεί οποιοσδήποτε να αποκτήσει πρόσβαση μέσω ενός smartphone. Η συγκεκριμένη εφαρμογή διατίθεται από την εταιρία Google και μπορεί να χρησιμοποιηθεί για την καταγραφή δεδομένων, εφόσον το smartphone προσαρμοστεί με κατάλληλο τρόπο σε οποιοδήποτε σημείο του σώματος.

Στην παρούσα εργασία επιλέγεται η προσάρτηση του smartphone στον καρπό 2 εργαζομένων, έτσι ώστε να καταγράφονται τα σήματα χρονοσειρών κατά τη διάρκεια εκτέλεσης των κινήσεων

τους. Να σημειωθεί ότι η συχνότητα δειγματοληψίας της συγκεκριμένης εφαρμογής είναι 15Hz, η οποία κρίνεται ικανοποιητική για την καταγραφή σημάτων των εξεταζόμενων δραστηριοτήτων. Στη συνέχεια, πραγματοποιείται επεξεργασία των σημάτων αυτών μέσω τεχνικών μηχανικής μάθησης για την αναγνώριση μοτίβων και την αντιστοίχισή τους με τα αντιπροσωπευτικά είδη των καθηκόντων.

Τα βήματα λοιπόν της συγκεκριμένης ερευνητικής εργασίας και οι συνεισφορές της έναντι των έως τώρα προτεινόμενων μεθοδολογιών συνοψίζονται ως εξής:

1. Καταγραφή Σημάτων: Προσαρτώντας ένα μόνο smartphone στον καρπό 2 εργαζομένων, πραγματοποιείται ένας μεγάλος αριθμός πειραμάτων από 5 διαφορετικές δραστηριότητες. Οι δραστηριότητες είναι πολύ δύσκολα ανιχνεύσιμες, καθώς δε διαφέρουν αισθητά από άλλες κινήσεις μη ενδιαφέροντος που ένας εργαζόμενος δύναται να εκτελέσει. Συνεπώς, το πρόβλημα που επιδιώκεται να επιλυθεί αποτελεί ένα δύσκολο πρόβλημα αναγνώρισης των ανθρώπινων δραστηριοτήτων στο επίπεδο παραγωγής, το οποίο στηρίζεται αποκλειστικά στην καταγραφή σημάτων από ένα μόνο σημείο του σώματος κάθε χειριστή.
2. Εξαγωγή Χαρακτηριστικών: Έχοντας καταγράψει όλα τα σήματα από τις 5 εξεταζόμενες δραστηριότητες, επιστρατεύονται διάφορες μέθοδοι εξαγωγής χαρακτηριστικών από τις χρονοσειρές αυτές. Οι μέθοδοι που χρησιμοποιούνται στην παρούσα εργασία αποτελούν καινοτόμες τεχνικές συμπίεσης δεδομένων, οι οποίες μελετώνται κατά κόρον από αρκετούς ερευνητές το τελευταίο διάστημα. Να σημειωθεί επίσης ότι ορισμένα από τα δίκτυα που θα παρουσιαστούν (π.χ. SA-VAE, VLAE και HF-VAE) δεν έχουν χρησιμοποιηθεί ξανά για τη μελέτη της συνεργασίας χειριστή-ρομπότ στο επίπεδο παραγωγής.
3. Ταξινόμηση Δραστηριοτήτων: Εκπαιδευοντας νευρωνικά δίκτυα χρονοσειρών LSTM διπλής κατεύθυνσης, πραγματοποιείται ταξινόμηση των 5 εξεταζόμενων δραστηριοτήτων. Βέβαια, να σημειωθεί ότι στο τελευταίο επίπεδο των δικτύων αυτών ενσωματώνονται τα χαρακτηριστικά των σημάτων που προέκυψαν από το βήμα 2, με σκοπό την προσθήκη επιπλέον πληροφορίας επιδιώκοντας την αύξηση των ποσοστών σωστής ταξινόμησης. Με αυτόν τον τρόπο ουσιαστικά συνδυάζονται 2 είδη νευρωνικών δικτύων, καταλήγοντας σε πολύ υψηλά ποσοστά σωστής κατηγοριοποίησης των 5 εξεταζόμενων δραστηριοτήτων.
4. Σήματα Μακράς Διάρκειας: Τέλος, χρησιμοποιώντας τα εκπαιδευμένα νευρωνικά δίκτυα παρουσιάζεται μία νέα μεθοδολογία, βάσει της οποίας προβλέπεται ποιές από τις 5 εξεταζόμενες κινήσεις υπάρχουν σε σήματα μακράς διάρκειας (ΣΜΔ) και με ποιά σειρά πραγματοποιήθηκαν. Τα σήματα αυτά ουσιαστικά αποτελούν χρονοσειρές πολύ μεγαλύτερης διάρκειας από τις 5 εξεταζόμενες δραστηριότητες, στις οποίες περιλαμβάνονται επιπλέον κινήσεις μη ενδιαφέροντος που πρέπει να "φιλτράρονται" και να "αγνοούνται" από οποιονδήποτε ρομποτικό μηχανισμό. Να σημειωθεί ότι η μεθοδολογία που προτείνεται παρουσιάζεται για πρώτη φορά και δε βασίζεται στην τεχνική των έως τώρα εργασιών του συγκεκριμένου αντικείμενου, οι οποίες εισάγουν μία επιπλέον κλάση NULL για την ενσωμάτωση όλων των κινήσεων μη ενδιαφέροντος.

Ύστερα λοιπόν από την παρουσίαση της προτεινόμενης μεθοδολογίας και των συνεισφορών της παρούσας εργασίας, στην Εν. 1.4 που ακολουθεί συνοψίζεται η δομή του συγκεκριμένου τόμου και αναφέρονται επιγραμματικά τα περιεχόμενα των Κεφαλαίων που ακολουθούν.

1.4 Οργάνωση της Εργασίας

Ύστερα από το 1^ο Κεφάλαιο της Εισαγωγής, η παρούσα εργασία χωρίζεται σε επιπλέον 7 Κεφάλαια, όπως αυτά αναγράφονται στον Πίν. 1.1. Πιο αναλυτικά, στο Κεφάλαιο 2 παρουσιάζονται τα νευρωνικά δίκτυα χρονοσειρών, τα οποία εκπαιδεύονται για την ταξινόμηση των 5 εξεταζόμενων δραστηριοτήτων. Ακολούθως, στο Κεφάλαιο 3 αναλύονται τα νευρωνικά δίκτυα

αυτοκωδικοποιητών, τα οποία χρησιμοποιούνται για τη συμπίεση δεδομένων και την εξαγωγή χαρακτηριστικών. Τα χαρακτηριστικά αυτά προστίθενται στο τελευταίο επίπεδο των δικτύων LSTM, με στόχο την αύξηση των ποσοστών σωστής ταξινόμησης των 5 εξεταζόμενων κινήσεων. Τα Κεφάλαια 4 και 5 αφιερώνονται επίσης στα νευρωνικά δίκτυα αυτοκωδικοποιητών, παρουσιάζοντας ορισμένες προηγμένες τεχνικές συμπίεσης δεδομένων. Στο Κεφάλαιο 6 αναλύονται τα πειράματα που πραγματοποιήθηκαν από τους 2 χειριστές στο επίπεδο παραγωγής, ενώ στο Κεφάλαιο 7 παρουσιάζονται τα αποτελέσματα των νευρωνικών δικτύων της παρούσας μελέτης. Επιπλέον, στο Κεφάλαιο 8 συγκρίνονται 4 διαφορετικοί υπολογιστικοί πόροι και εξετάζεται εάν τα αποτελέσματα τις προτεινόμενης μεθοδολογίας μπορούν να εξαχθούν σε πραγματικό χρόνο και κατά συνέπεια να ενσωματωθούν στη γραμμή παραγωγής ενός εργοστασίου. Τέλος, στο Κεφάλαιο 9 συνοψίζονται τα συμπεράσματα της εργασίας και προτείνονται ορισμένες ιδέες και πιθανές επεκτάσεις βάσει των έως τώρα αποτελεσμάτων.

Πίνακας 1.1: Κεφάλαια διπλωματικής εργασίας.

| Κεφάλαιο | Τίτλος |
|----------|-----------------------------------------|
| 2 | Νευρωνικά Δίκτυα Χρονοσειρών |
| 3 | Νευρωνικά Δίκτυα Αυτοκωδικοποιητών |
| 4 | Προηγμένη Variational Συμπερασματολογία |
| 5 | Κανονικοποιημένες Ροές |
| 6 | Περιγραφή Πειραμάτων |
| 7 | Παρουσίαση Αποτελεσμάτων |
| 8 | Μελέτη Απόκρισης Υπολογιστικών Πόρων |
| 9 | Συμπεράσματα και Μελλοντική Εργασία |

Κεφάλαιο 2

Νευρωνικά Δίκτυα Χρονοσειρών

Στο παρόν Κεφάλαιο πραγματοποιείται μία εισαγωγή στα νευρωνικά δίκτυα και συγκεκριμένα σε αυτά που χρησιμοποιούνται για τη μοντελοποίηση χρονοσειρών. Συγκεκριμένα, στην Εν. 2.1 εξηγείται η αρχιτεκτονική και λειτουργία των απλών αναδρομικών δικτύων RNN, καθώς και ο τρόπος με τον οποίο αυτά εκπαιδεύονται. Στην Εν. 2.2 πραγματοποιείται μία σύντομη αναφορά στα προβλήματα που προκύπτουν κατά την εκπαίδευση των απλών αναδρομικών δικτύων και ως λύση προτείνονται τα δίκτυα μακράς και βραχείας μνήμης LSTM. Τέλος, στην Εν. 2.3 εξηγούνται οι αλγόριθμοι που μπορούν να χρησιμοποιηθούν κατά την εκπαίδευση των βαρών ενός νευρωνικού δικτύου και επισημαίνονται οι βασικές διαφορές και τα πλεονεκτήματα κάθε μεθόδου.

2.1 Αναδρομικά Νευρωνικά Δίκτυα RNN

Τα αναδρομικά νευρωνικά δίκτυα (Recurrent Neural Networks - RNN) προτάθηκαν για πρώτη φορά από τους συγγραφείς στο [25] και ανήκουν στην κατηγορία των νευρωνικών δικτύων που μοντελοποιούν δεδομένα, τα οποία παρουσιάζουν κάποια χρονική εξάρτηση μεταξύ τους. Η βασική ιδέα των συγκεκριμένων δικτύων θεωρείται το διαμοίρασμα των παραμέτρων του δικτύου στα δεδομένα διαφορετικών χρονικών στιγμών, έτσι ώστε να επιτευχθεί η μοντελοποίηση της χρονικής εξάρτησής τους. Πριν όμως την ανάλυση των συγκεκριμένων δικτύων απαιτείται μία σύντομη αναφορά στους υπολογιστικούς γράφους, οι οποίοι περιγράφουν σχηματικά τους υπολογισμούς που πραγματοποιούνται στα αναδρομικά νευρωνικά δίκτυα.

2.1.1 Υπολογιστικοί Γράφοι

Οι υπολογιστικοί γράφοι χρησιμοποιούνται ευρέως στην επιστήμη της μηχανικής μάθησης με στόχο τη σχηματική αναπαράσταση των υπολογισμών που απαιτούνται κατά την εκπαίδευση ενός αλγορίθμου. Με αυτή την τεχνική γίνεται πιο κατανοητός ο τρόπος διασύνδεσης των δεδομένων εισόδου και εξόδου δια μέσου των παραμέτρων του μοντέλου, ενώ παράλληλα απεικονίζονται γραφικά ποιά δεδομένα χρησιμοποιούνται για να εκτιμηθεί η συνάρτηση κόστους. Στην παρούσα υποενότητα ουσιαστικά παρουσιάζεται η ιδέα του ξεδιπλώματος ενός υπολογιστικού γράφου που αντιστοιχεί σε μία αλληλουχία γεγονότων, δηλαδή μίας τεχνικής που χρησιμοποιείται για να υποδηλώσει το διαμοίρασμα των παραμέτρων ενός μοντέλου σε δεδομένα διαφορετικών χρονικών στιγμών.

Έστω λοιπόν ένα δυναμικό σύστημα το οποίο περιγράφεται από την Εξ. (2.1):

$$\mathbf{s}_t = f(\mathbf{s}_{t-1}, x_t, \theta) \quad (2.1)$$

όπου \mathbf{s}_t θεωρείται η κατάσταση (state) του συστήματος τη χρονική στιγμή t , \mathbf{x}_t τα δεδομένα εισόδου τη χρονική στιγμή t και θ οι παράμετροι του νευρωνικού δικτύου. Η Εξ. (2.1) αναφέρεται σε μία αναδρομική σχέση με την έννοια ότι κάθε μία από τις χρονικές στιγμές t συνδέεται με την προηγούμενή της $t - 1$ για όλο το πεπερασμένο πλήθος χρονικών στιγμών τ του προβλήματος που εξετάζεται.

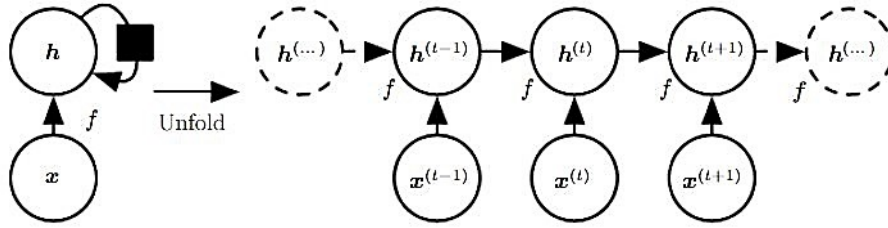
Για το πεπερασμένο αυτό πλήθος τ των χρονικών στιγμών ο υπολογιστικός αυτός γράφος μπορεί να ξεδιπλωθεί $\tau - 1$ φορές. Για παράδειγμα, αν στην Εξ. (2.1) θεωρηθεί ότι $\tau = 3$, τότε προκύπτει το εξής:

$$\mathbf{s}_3 = f(\mathbf{s}_2, \theta) = f(f(\mathbf{s}_1, \theta), \theta) \quad (2.2)$$

Η Εξ. (2.2) ουσιαστικά προέκυψε από την επαναλαμβανόμενη αντικατάσταση κάθε κατάστασης από την προηγούμενή της, με αποτέλεσμα να οδηγηθούμε σε ένα αποτέλεσμα το οποίο εν τέλει δεν περιλαμβάνει εσωτερικά κάποια επιπλέον εναλλαξιμότητα. Το αποτέλεσμα αυτό μπορεί να αναπαρασταθεί γραφικά χρησιμοποιώντας τους κατευθυνόμενους, μη κυκλικούς υπολογιστικούς γράφους (directed acyclic computational graphs), όπως παρουσιάζεται στο Σχ. 2.1. Στη συγκεκριμένη περίπτωση αναπαριστάται ένα ξεδιπλωμένο νευρωνικό δίκτυο χωρίς δεδομένα εξόδου σε κάθε χρονική στιγμή, ενώ παράλληλα η κατάσταση κάθε χρονικής στιγμής ουσιαστικά ισοδυναμεί με το κρυφό επίπεδο (hidden layer) του δικτύου και απεικονίζεται με την τιμή \mathbf{h} :

$$\mathbf{h}_t = f(\mathbf{h}_{t-1}, \mathbf{x}_t, \theta) \quad (2.3)$$

Κατά την εκπαίδευση του αναδρομικού νευρωνικού δικτύου ουσιαστικά το δίκτυο "μαθαίνει" να ενσωματώνει όλη την πληροφορία μέχρι τη χρονική στιγμή t στο κρυφό επίπεδο \mathbf{h}_t , με στόχο την πρόβλεψη της τιμής της επόμενης χρονικής στιγμής.

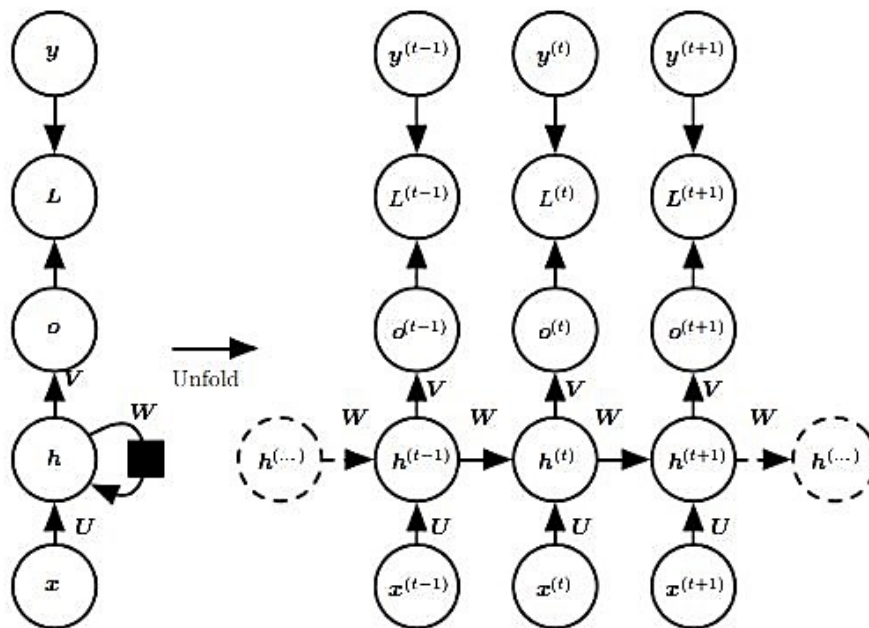


Σχήμα 2.1: Αναδρομικό δίκτυο χωρίς δεδομένα εξόδου. Στο αριστερό τμήμα αναπαριστάται ο αναδρομικός υπολογιστικός γράφος, ενώ στο δεξί τμήμα ο ξεδιπλωμένος μη κυκλικός υπολογιστικός γράφος. Πηγή [26].

2.1.2 Αρχιτεκτονική Αναδρομικών Νευρωνικών Δικτύων

Ύστερα από τη σύντομη αναφορά στους υπολογιστικούς γράφους, στην παρούσα υποενότητα παρουσιάζεται η αρχιτεκτονική των αναδρομικών δικτύων, καθώς και οι εξισώσεις από τις οποίες αυτά διέπονται. Αρχικά λοιπόν η αρχιτεκτονική των αναδρομικών νευρωνικών δικτύων απεικονίζεται στον υπολογιστικό γράφο του Σχ. 2.2. Συγκεκριμένα, το αναδρομικό δίκτυο αντιστοιχίζει μία ακολουθία δεδομένων εισόδου \mathbf{x}_t σε μία αντίστοιχη ακολουθία \mathbf{o}_t μέσω των κρυφών επιπέδων \mathbf{h}_t . Στη συνέχεια, για κάθε χρονική στιγμή t ορίζεται η συνάρτηση κόστους L_t , η οποία μετράει πόσο απέχουν οι εξόδοι του νευρωνικού δικτύου \mathbf{o}_t από τις επιθυμητές τιμές \mathbf{y}_t . Να σημειωθεί ότι στις περιπτώσεις όπου θέλουμε να μετατρέψουμε τις εξόδους του νευρωνικού δικτύου σε πιθανότητες, τότε η συνάρτηση κόστους L_t εσωτερικά περιλαμβάνει το μετασχηματισμό $\hat{\mathbf{y}}_t = \text{softmax}(\mathbf{o}_t)$.

Όσον αφορά τις παραμέτρους του νευρωνικού δικτύου, αυτές αποτελούνται από τους πίνακες \mathbf{U} , \mathbf{W} και \mathbf{V} , οι οποίοι μοιράζονται σε όλη την ακολουθία των δεδομένων που εισέρχονται στο δίκτυο. Ο πίνακας \mathbf{U} ουσιαστικά συνδέει τα δεδομένα εισόδου \mathbf{x}_t με το κρυφό επίπεδο του νευρωνικού \mathbf{h}_t , ο πίνακας \mathbf{W} μετασχηματίζει την κρυφή αναπαράσταση \mathbf{h}_t μεταξύ διαδοχικών χρονικών στιγμών, ενώ τέλος ο πίνακας \mathbf{V} αφορά το μετασχηματισμό των κρυφών αναπαράστασεων \mathbf{h}_t στις τιμές εξόδου \mathbf{o}_t κάθε χρονικής στιγμής του δικτύου.



Σχήμα 2.2: Υπολογιστικός γράφος που απεικονίζει την αρχιτεκτονική ενός αναδρομικού δικτύου και τον τρόπο εκτίμησης της συνάρτησης κόστους. Πηγή [26].

Για την εκτίμηση των δεδομένων εξόδου \mathbf{o}_t κάθε χρονικής στιγμής απαιτείται η εμπρόσθια τροφοδότηση των δεδομένων εισόδου \mathbf{x}_t εντός του δικτύου. Στις περισσότερες περιπτώσεις ως

μη γραμμική συνάρτηση ενεργοποίησης των δικτύων αυτών θεωρείται η συνάρτηση \tanh , δηλαδή η υπερβολική εφαπτομένη. Η εμπρόσθια τροφοδότηση ξεκινά υπό την υπόθεση μίας συγκεκριμένης κρυφής αναπαράστασης \mathbf{h}_0 η οποία τροφοδοτείται στην πρώτη χρονική στιγμή, όπως απεικονίζεται στο Σχ. 2.2. Στη συνέχεια, για κάθε χρονική στιγμή από $t = 1$ έως και $t = \tau$, οι εξισώσεις που χρησιμοποιούνται είναι οι εξής:

$$\mathbf{a}_t = \mathbf{b} + \mathbf{W}\mathbf{h}_{t-1} + \mathbf{U}\mathbf{x}_t \quad (2.4)$$

$$\mathbf{h}_t = \tanh(\mathbf{a}_t) \quad (2.5)$$

$$\mathbf{o}_t = \mathbf{c} + \mathbf{V}\mathbf{h}_t \quad (2.6)$$

$$\hat{\mathbf{y}}_t = \text{softmax}(\mathbf{o}_t) \quad (2.7)$$

όπου \mathbf{b} και \mathbf{c} είναι τα διανύσματα μεροληψίας (bias vectors), ενώ οι πίνακες \mathbf{U} , \mathbf{W} και \mathbf{V} είναι οι γραμμικοί μετασχηματισμοί των δεδομένων που αναφέρθηκαν προηγουμένως. Να σημειωθεί ότι η συνάρτηση softmax δε χρησιμοποιείται σε όλες τις περιπτώσεις, αλλά αποτελεί μία ενδεικτική συνάρτηση που επιλέγεται στην προκείμενη περίπτωση. Επιπλέον, όσον αφορά τη συνάρτηση κόστους, αυτή εκτιμάται από την Εξ. (2.8):

$$L(\{\mathbf{x}_1, \dots, \mathbf{x}_\tau\}, \{\mathbf{y}_1, \dots, \mathbf{y}_\tau\}) = \sum_t L_t \quad (2.8)$$

όπου σε κάθε χρονική στιγμή t η ποσότητα L_t επιλέγεται ανάλογα με το πρόβλημα που εξετάζεται (π.χ. το τετραγωνικό σφάλμα ή αρνητική λογαριθμοποιημένη πιθανοφάνεια μεταξύ των ποσοτήτων $\hat{\mathbf{y}}_t$ και \mathbf{y}_t).

Τέλος, να σημειωθεί ότι για την εκπαίδευση του αναδρομικού νευρωνικού δικτύου απαιτείται η οπισθοδιάδοση των διανυσματικών παραγώγων (gradients) της συνάρτησης κόστους L ως προς τις παραμέτρους του δικτύου. Το υπολογιστικό κόστος αυτό είναι ανάλογο του $O(\tau)$ και δεν μπορεί να μειωθεί περαιτέρω λόγω της διαδοχικής εξάρτησης κάθε χρονικής στιγμής από την προηγούμενη της. Η οπισθοδιάδοση αυτή των διανυσματικών παραγώγων μπορεί να πραγματοποιηθεί με τον αλγόριθμο οπισθοδιάδοσης του λάθους δια μέσω του χρόνου (Back Propagation Through Time - BPTT), όπως συνοπτικά παρουσιάζεται στην επόμενη υποενότητα.

2.1.3 Εκπαίδευση Αναδρομικών Νευρωνικών Δικτύων

Στη συγκεκριμένη υποενότητα παρουσιάζεται ο τρόπος εκτίμησης των διανυσματικών παραγώγων της συνάρτησης κόστους L ως προς τις παραμέτρους του αναδρομικού δικτύου. Για την εκτίμηση αυτών των διανυσματικών παραγώγων θα χρησιμοποιηθεί ο αλγόριθμος οπισθοδιάδοσης του λάθους δια μέσω του χρόνου (BPTT).

Σύμφωνα λοιπόν με το Σχ. 2.2, οι κόμβοι του υπολογιστικού γράφου περιλαμβάνουν τις παραμέτρους του νευρωνικού δικτύου \mathbf{U} , \mathbf{W} , \mathbf{V} , \mathbf{b} και \mathbf{c} , καθώς και όλες τις ποσότητες οι οποίες εξαρτώνται από το χρόνο t , δηλαδή \mathbf{x}_t , \mathbf{h}_t , \mathbf{o}_t και L_t . Εφόσον οι παράμετροι του νευρωνικού διαμοιράζονται σε όλα τα δεδομένα, πριν την εκτίμηση των διανυσματικών παραγώγων των παραμέτρων αυτών θα πρέπει αρχικά να μελετηθούν οι διανυσματικές παράγωγοι των ποσοτήτων που εξαρτώνται από το χρόνο t .

Έστω λοιπόν ότι αρχικά απαιτείται η εκτίμηση της διανυσματικής παραγώγου $\nabla_{\mathbf{o}_t} L$ για οποιαδήποτε χρονική στιγμή $t \leq \tau$. Σύμφωνα με την Εξ. (2.8) η συνολική συνάρτηση κόστους L ισούται με το άθροισμα όλων των επιμέρους σφαλμάτων L_t , συνεπώς η ποσότητα $\nabla_{\mathbf{o}_t} L$ γράφεται:

$$\nabla_{\mathbf{o}_t} L = \frac{\partial L}{\partial \mathbf{o}_t} = \frac{\partial L}{\partial L_t} \frac{\partial L_t}{\partial \mathbf{o}_t} = \frac{\partial L_t}{\partial \mathbf{o}_t} \quad (2.9)$$

όπου για κάθε χρονική στιγμή t η παράγωγος της ποσότητας L_t ως προς την ποσότητα \mathbf{o}_t μπορεί να εκτιμηθεί. Έστω στη συνέχεια ότι επιθυμούμε να εκτιμήσουμε τη διανυσματική παράγωγο $\nabla_{\mathbf{h}_t} L$ για οποιαδήποτε χρονική στιγμή $t \leq \tau$. Η παράγωγος αυτή δεν μπορεί να εκτιμηθεί με την ίδια ευκολία όσο αυτή της Εξ. (2.9). Αυτό συμβαίνει διότι σύμφωνα με το Σχ. 2.2, η κρυφή αναπαράσταση \mathbf{h}_t δεν επηρεάζει μόνο την ίδια χρονική στιγμή t αλλά και την ακριβώς επόμενη που ακολουθεί. Συνεπώς, η διανυσματική παράγωγος $\nabla_{\mathbf{h}_t} L$ θα αποτελείται από το άθροισμα 2 όρων, όπως παρουσιάζεται ακολούθως:

$$\begin{aligned} \nabla_{\mathbf{h}_t} L &= \left(\frac{\partial \mathbf{h}_{t+1}}{\partial \mathbf{h}_t} \right)^T (\nabla_{\mathbf{h}_{t+1}} L) + \left(\frac{\partial \mathbf{o}_t}{\partial \mathbf{h}_t} \right)^T (\nabla_{\mathbf{o}_t} L) \longrightarrow \\ \nabla_{\mathbf{h}_t} L &= \mathbf{W}^T (\nabla_{\mathbf{h}_{t+1}} L) + \mathbf{V}^T (\nabla_{\mathbf{o}_t} L) \end{aligned} \quad (2.10)$$

όπου ο δεύτερος όρος της Εξ. (2.10) μπορεί να εκτιμηθεί με χρήση της Εξ. (2.9). Αντιθέτως, η εκτίμηση του πρώτου όρου απαιτεί την εκτίμηση της αντίστοιχης παραγώγου της ποσότητας L ως προς \mathbf{h} την ακριβώς επόμενη χρονική στιγμή, δηλαδή την παράγωγο $\nabla_{\mathbf{h}_{t+1}} L$. Το ερώτημα λοιπόν που τίθεται είναι πώς θα εκτιμηθεί η συγκεκριμένη παράγωγος, όταν για κάθε μία χρονική στιγμή απαιτείται η εκτίμηση της αντίστοιχης παραγώγου για την επόμενη χρονική στιγμή;

Το πρόβλημα αυτό λοιπόν μπορεί να λυθεί αν μελετηθεί η συγκεκριμένη παράγωγος για την τελευταία χρονική στιγμή τ , για την οποία δεν υπάρχει επόμενη κρυφή αναπαράσταση $\mathbf{h}_{\tau+1}$. Αυτό όμως σημαίνει ότι ο πρώτος όρος της Εξ. (2.10) μηδενίζεται και συνεπώς η διανυσματική παράγωγος $\nabla_{\mathbf{h}_\tau} L$ προκύπτει:

$$\nabla_{\mathbf{h}_\tau} L = \mathbf{V}^T (\nabla_{\mathbf{o}_\tau} L) \quad (2.11)$$

η οποία μπορεί να εκτιμηθεί χρησιμοποιώντας την Εξ. (2.9). Ύστερα από την εκτίμηση της συγκεκριμένης παραγώγου μπορούν αναδρομικά να εκτιμηθούν όλες οι αντίστοιχες παράγωγοι $\nabla_{\mathbf{h}_t} L$ για όλες τις χρονικές στιγμές $t < \tau$.

Έχοντας λοιπόν εκτιμήσει τις διανυσματικές παραγώγους $\nabla_{\mathbf{o}_t} L$ και $\nabla_{\mathbf{h}_t} L$ για κάθε μία χρονική στιγμή t , στη συνέχεια μπορούν να εκτιμηθούν οι διανυσματικές παράγωγοι της ποσότητας L ως προς τις παραμέτρους \mathbf{U} , \mathbf{W} , \mathbf{V} , \mathbf{b} και \mathbf{c} του αναδρομικού νευρωνικού δικτύου. Οι παράγωγοι αυτές βασίζονται στις συσχετίσεις των ποσοτήτων \mathbf{o}_t και \mathbf{h}_t με τις παραμέτρους του νευρωνικού (δες τις Εξ. (2.4), (2.5), (2.6) και (2.7)) και παρουσιάζονται συνοπτικά ακολούθως:

$$\nabla_{\mathbf{c}} L = \sum_t \left(\frac{\partial \mathbf{o}_t}{\partial \mathbf{c}} \right)^T (\nabla_{\mathbf{o}_t} L) = \sum_t (\nabla_{\mathbf{o}_t} L) \quad (2.12)$$

$$\begin{aligned} \nabla_{\mathbf{b}} L &= \sum_t \left(\frac{\partial \mathbf{h}_t}{\partial \mathbf{b}} \right)^T (\nabla_{\mathbf{h}_t} L) = \sum_t \left(\frac{\partial}{\partial \mathbf{b}} \tanh(\mathbf{b} + \mathbf{W}\mathbf{h}_{t-1} + \mathbf{U}\mathbf{x}_t) \right)^T (\nabla_{\mathbf{h}_t} L) \longrightarrow \\ \nabla_{\mathbf{b}} L &= \sum_t \left(\frac{\partial \mathbf{h}_t}{\partial \mathbf{b}} \right)^T (\nabla_{\mathbf{h}_t} L) = \sum_t \text{diag}(1 - \mathbf{h}_t^2) (\nabla_{\mathbf{h}_t} L) \end{aligned} \quad (2.13)$$

$$\nabla_{\mathbf{v}} L = \sum_t \sum_i \left(\frac{\partial L}{\partial o_t^i} \right) (\nabla_{\mathbf{v} o_t^i}) = \sum_t (\nabla_{\mathbf{o}_t} L) \mathbf{h}_t^T \quad (2.14)$$

$$\nabla_{\mathbf{w}} L = \sum_t \sum_i \left(\frac{\partial L}{\partial h_t^i} \right) (\nabla_{\mathbf{w} h_t^i}) = \sum_t \text{diag}(1 - \mathbf{h}_t^2) (\nabla_{\mathbf{h}_t} L) \mathbf{h}_{t-1}^T \quad (2.15)$$

$$\nabla_{\mathbf{u}} L = \sum_t \sum_i \left(\frac{\partial L}{\partial h_t^i} \right) (\nabla_{\mathbf{u} h_t^i}) = \sum_t \text{diag}(1 - \mathbf{h}_t^2) (\nabla_{\mathbf{h}_t} L) \mathbf{x}_t^T \quad (2.16)$$

Ύστερα από τις εκτιμήσεις των παραπάνω παραγώγων απαιτείται η κατάλληλη επιλογή της μεθόδου βελτιστοποίησης, η οποία θα χρησιμοποιεί τις διανυσματικές αυτές παραγώγους με στόχο την ανανέωση των βαρών του δικτύου. Ορισμένες από τις μεθόδους αυτές αναλύονται στον Εν. 2.3, όπου εξηγείται με ποιόν τρόπο πραγματοποιείται η ανανέωση των παραμέτρων του δικτύου για την ελαχιστοποίηση της συνάρτησης κόστους L . Πριν όμως την παρουσίαση των συγκεκριμένων μεθόδων, στην Εν. 2.2 εξηγούνται οι δυσκολίες εκπαίδευσης των αναδρομικών νευρωνικών δικτύων RNN και αναφέρονται ορισμένες παραλλαγές που έχουν προταθεί και χρησιμοποιούνται ευρέως μέχρι και σήμερα.

2.2 Το Πρόβλημα Μακροπρόθεσμων Εξαρτήσεων

Στην υποενότητα 2.1.3 παρουσιάστηκε ο τρόπος με τον οποίο μπορούν να εκτιμηθούν οι διανυσματικές παράγωγοι της συνάρτησης κόστους L ως προς τις παραμέτρους του αναδρομικού νευρωνικού δικτύου. Παρόλο όμως που κατά τη μαθηματική ανάλυση που προηγήθηκε δε φαίνεται να δημιουργείται κάποιο πρόβλημα, στην πράξη οι εκτιμήσεις αυτές δημιουργούν αρκετά θέματα κατά την εκπαίδευση τέτοιου είδους δικτύων.

Το κύριο πρόβλημα που προκύπτει είναι ότι στις περισσότερες περιπτώσεις οι τιμές των διανυσματικών παραγώγων "εξασθενούν" (vanish) ή "εκρήγνυνται" (explode). Η πρώτη περίπτωση είναι συνηθέστερη και αναφέρεται στις υπερβολικά χαμηλές τιμές που μπορεί να λαμβάνουν οι διανυσματικές παράγωγοι, με αποτέλεσμα οι παράμετροι του δικτύου να μην ανανεώνονται κατά τη διάρκεια της εκπαίδευσης. Αντιθέτως, η δεύτερη περίπτωση αναφέρεται στις υπερβολικά υψηλές τιμές των διανυσματικών παραγώγων, οι οποίες μεταβάλλουν τις παραμέτρους του δικτύου με υπερβολικό τρόπο σε κάθε επανάληψη και έτσι αποτρέπουν τη σύγκλιση τους σε συγκεκριμένες τιμές κατά τη διάρκεια της εκπαίδευσης.

Το πρόβλημα αυτό συναντάται στις περισσότερες αρχιτεκτονικές των βαθιών νευρωνικών δικτύων. Βέβαια, στις περιπτώσεις των αναδρομικών δικτύων παρουσιάζεται πιο συχνά, καθώς το βάθος του δικτύου εξαρτάται από το πλήθος των χρονικών στιγμών τ των δεδομένων εισόδου, το οποίο σε αρκετές περιπτώσεις είναι αρκετά μεγάλο. Η κύρια αιτία του προβλήματος αυτού είναι ότι για την εκτίμηση των διανυσματικών παραγώγων μίας χρονικής στιγμής απαιτείται το γινόμενο των αντίστοιχων παραγώγων διαφορετικών χρονικών στιγμών.

Στην περίπτωση των αναδρομικών δικτύων το φαινόμενο αυτό παρατηρείται στην Εξ. (2.10) και συγκεκριμένα κατά την εκτίμηση της ποσότητας $\nabla_{\mathbf{h}_t} L$. Η εξίσωση αυτή αποτελεί μία αναδρομική σχέση η οποία συσχετίζει την παράγωγο της μίας χρονικής στιγμής με την επόμενη και συνεπώς, αν η σχέση αυτή "ξεδιπλωθεί" θα προκύψει ένα γινόμενο διαδοχικών διανυσματικών παραγώγων. Αν λοιπόν οι τιμές αυτές των παραγώγων είναι μικρότερες του 1 τότε το γινόμενο αυτό θα οδηγήσει σε αρκετά χαμηλές τιμές, ενώ αντίθετα αν οι τιμές των παραγώγων είναι μεγαλύτερες του 1 το γινόμενο θα λάβει υπερβολικά υψηλές τιμές.

Για την αντιμετώπιση του συγκεκριμένου προβλήματος έχουν προταθεί αρκετές διαφορετικές μέθοδοι, ορισμένες από τις οποίες παρουσιάζονται ακολούθως:

1. Echo State Networks (ESN) [27, 28]: Τα συγκεκριμένα δίκτυα θέτουν περιορισμούς στους πίνακες του αναδρομικού δικτύου (π.χ. $\mathbf{W} = \mathbf{U} = \mathbf{I}$) και εκπαιδεύουν μόνο τον πίνακα βαρών \mathbf{V} που αντιστοιχεί στα δεδομένα εξόδου.
2. Liquid State Networks (LSN) [29]: Τα νευρωνικά δίκτυα αυτά είναι αντίστοιχα με τα ESN, με τη διαφορά ότι χρησιμοποιούν νευρώνες τύπου spiking με δυαδικές τιμές εξόδου (binary outputs), σε αντίθεση με τα κρυφά επίπεδα \mathbf{h} των δικτύων ESN που λαμβάνουν συνεχείς τιμές.
3. Δίκτυα με Skip Connections [30]: Τα δίκτυα αυτά προσθέτουν συνδέσεις συγκεκριμένων χρονικών στιγμών με αρκετά παλαιότερες και με αυτόν τον τρόπο επιτυγχάνεται η διάδοση της πληροφορίας για τη μοντελοποίηση των δεδομένων.

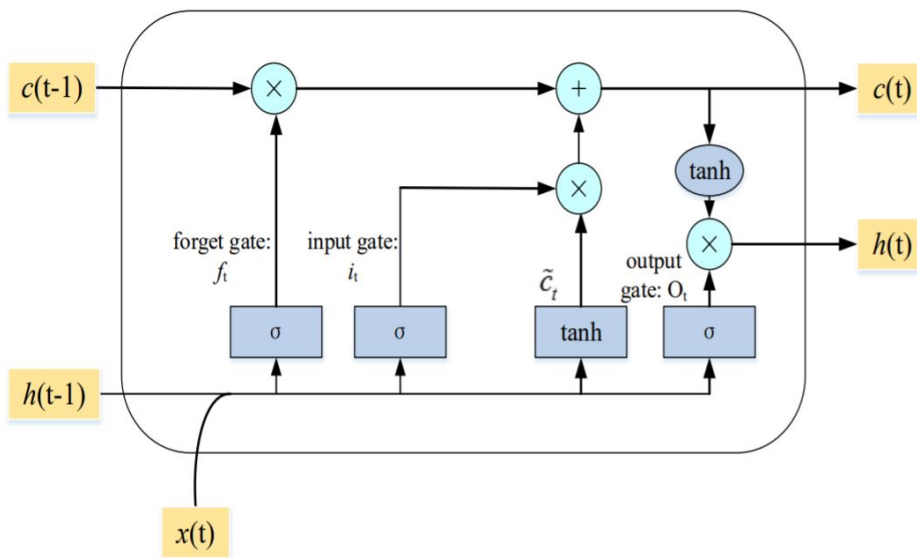
4. Gated RNNs [31, 32]: Τα συγκεκριμένα δίκτυα αντιμετωπίζουν το πρόβλημα της διάδοσης πληροφορίας με χρήση πυλών (gates). Οι πύλες αυτές συνήθως ισοδυναμούν με τη λογιστική συνάρτηση και επιτρέπουν τη διάδοση της πληροφορίας ανάλογα με την τιμή της κ.α.

Στη συγκεκριμένη εργασία μελετάται μία ειδική κατηγορία των δικτύων Gated RNNs, τα οποία ονομάζονται δίκτυα μακράς και βραχείας μνήμης. Η ανάλυση των δικτύων αυτών παρουσιάζεται στην υποενότητα που ακολουθεί και εξηγείται με ποιόν τρόπο τα δίκτυα αυτά καταφέρνουν να αντιμετωπίσουν το πρόβλημα της "εξασθένησης" ή της "έκρηξης" των διανυσματικών παραγώγων που παρουσιάζεται στα απλά αναδρομικά δίκτυα RNNs.

2.2.1 Δίκτυα Μακράς και Βραχείας Μνήμης LSTM

Τα νευρωνικά δίκτυα μακράς και βραχείας μνήμης (Long Short-Term Memory - LSTM) [32] αποτελούν μία εναλλακτική αρχιτεκτονική αναδρομικών δικτύων, τα οποία στηρίζονται στη χρήση πυλών για την αποφυγή της "εξασθένησης" ή της "έκρηξης" των διανυσματικών παραγώγων που παρουσιάζεται στα απλά αναδρομικά δίκτυα. Το ερώτημα λοιπόν που τίθεται είναι πώς η χρήση συγκεκριμένων πυλών μπορεί να βοηθήσει τη διάδοση της πληροφορίας από αρκετά παλαιότερες χρονικές στιγμές, έτσι ώστε να αποφευχθεί το πρόβλημα των διανυσματικών παραγώγων που μόλις προαναφέρθηκε;

Εστω λοιπόν το νευρωνικό δίκτυο LSTM που απεικονίζεται στο Σχ. 2.3. Σε αντίθεση με την αρχιτεκτονική των απλών νευρωνικών δικτύων RNN του Σχ. 2.2, τα δίκτυα LSTM δέχονται 2 ξεχωριστές ποσότητες στο κελί κάθε χρονικής στιγμής t : (1) το διάνυσμα \mathbf{h}_{t-1} που αποτελεί την κατάσταση της προηγούμενης χρονικής στιγμής (βραχεία μνήμη) και (2) το διάνυσμα \mathbf{c}_{t-1} που μεταφέρει την πληροφορία της μακροπρόθεσμης μνήμης από τις αρκετά προηγούμενες χρονικές στιγμές.



Σχήμα 2.3: Αρχιτεκτονική κελιού του νευρωνικού δικτύου LSTM. Πηγή [33].

Η σημασία της πληροφορίας που μεταφέρουν τα 2 αυτά διανύσματα μνήμης ελέγχεται με τη βοήθεια 3 πυλών (gates), οι οποίες περιγράφονται συνοπτικά ακολούθως:

1. Forget Gate f_t : Η συγκεκριμένη πύλη δέχεται ως ορίσματα τα δεδομένα εισόδου \mathbf{x}_t και το διάνυσμα της προηγούμενης κατάστασης \mathbf{h}_{t-1} και τα τροφοδοτεί σε μία σιγμοειδή συνάρτηση, παράγοντας ένα διάνυσμα με τιμές μεταξύ του 0 και 1. Το διάνυσμα αυτό στη συνέχεια πολλαπλασιάζεται στοιχείο με στοιχείο με τη διάνυσμα \mathbf{c}_{t-1} , έτσι ώστε να ελεγχθούν ποιά δεδομένα της μακράς μνήμης είναι χρήσιμα. Με άλλα λόγια λοιπόν η συγκεκριμένη πύλη

"φιλτράρει" τα δεδομένα των παλαιότερων χρονικών στιγμών, δεδομένου της τωρινής (λόγω \mathbf{x}_t) και της ακριβώς προηγούμενης (λόγω \mathbf{h}_{t-1}) κατάστασης.

2. **Input Gate \mathbf{i}_t** : Η συγκεκριμένη πύλη δέχεται ξανά ως ορίσματα τα δεδομένα εισόδου \mathbf{x}_t και το διάνυσμα της προηγούμενης κατάστασης \mathbf{h}_{t-1} και τα τροφοδοτεί σε μία σιγμοειδή συνάρτηση, παράγοντας ένα δεύτερο διάνυσμα με τιμές μεταξύ του 0 και 1. Το διάνυσμα αυτό στη συνέχεια πολλαπλασιάζεται στοιχείο με στοιχείο με τη διάνυσμα $\tilde{\mathbf{c}}_t$, το οποίο ουσιαστικά αποτελεί την κατάσταση του τωρινού κελιού, όπως και στα απλά αναδρομικά δίκτυα RNN (δηλαδή έχει προκύψει από το μη γραμμική συνάρτηση \tanh των δεδομένων \mathbf{x}_t και \mathbf{h}_{t-1}). Η πύλη λοιπόν αυτή φιλτράρει μόνο τη βραχεία μνήμη των δεδομένων \mathbf{x}_t και \mathbf{h}_{t-1} , με στόχο να εξαχθεί ένα κατάλληλο διάνυσμα βραχείας μνήμης με την κατάλληλη πληροφορία. Τα δεδομένα που εξάγονται από τον πολλαπλασιασμό αυτό στη συνέχεια αθροίζονται με τα αντίστοιχα δεδομένα της πύλης forget, παράγοντας το διάνυσμα μακράς μνήμης \mathbf{c}_t που θα οδηγηθεί στο επόμενο χρονικά κελί του δικτύου.
3. **Output Gate \mathbf{o}_t** : Όπως ακριβώς και στις προηγούμενες 2 περιπτώσεις, η πύλη output δέχεται ξανά ως ορίσματα τα δεδομένα εισόδου \mathbf{x}_t και το διάνυσμα της προηγούμενης κατάστασης \mathbf{h}_{t-1} και τα τροφοδοτεί σε μία σιγμοειδή συνάρτηση, παράγοντας ένα τρίτο διάνυσμα με τιμές μεταξύ του 0 και 1. Το διάνυσμα αυτό στη συνέχεια πολλαπλασιάζεται στοιχείο με στοιχείο με τη διάνυσμα μακράς μνήμης \mathbf{c}_t του προηγούμενου βήματος, αφού όμως πρώτα το δεύτερο τροφοδοτηθεί στη μη γραμμική συνάρτηση \tanh . Με αυτόν τον τρόπο η πύλη αυτή "φιλτράρει" την πληροφορία της μακράς μνήμης και διατηρεί την πληροφορία εκείνη που αντιστοιχεί στην βραχεία μνήμη του δικτύου. Με αυτόν τον τρόπο λοιπόν παράγεται εν τέλει το διάνυσμα βραχείας μνήμης \mathbf{h}_t που θα τροφοδοτηθεί στο επόμενο χρονικά κελί του δικτύου.

Χρησιμοποιώντας λοιπόν τα 3 είδη πυλών που μόλις προαναφέρθηκαν, το αναδρομικό δίκτυο LSTM κατορθώνει σε κάθε χρονική στιγμή t να μεταφέρεται η πληροφορία τόσο της βραχείας μνήμης (λόγω του διανύσματος \mathbf{h}_{t-1}) όσο και της μακράς μνήμης (λόγω του διανύσματος \mathbf{c}_{t-1}). Συνεπώς, κατά την εκπαίδευση του δικτύου LSTM η ανανέωση των βαρών πραγματοποιείται με φυσιολογικό τρόπο, καθώς η χρήση των πυλών αποτρέπουν την εμφάνιση της "εξασθένησης" ή της "έκρηξης" των διανυσματικών παραγώγων που παρουσιάζεται στα απλά αναδρομικά δίκτυα RNN. Οι μαθηματικές εξισώσεις που διέπουν το συγκεκριμένο δίκτυο αναγράφονται ακολούθως:

$$\mathbf{f}_t = \sigma(\mathbf{W}_{if}\mathbf{x}_t + \mathbf{b}_{if} + \mathbf{W}_{hf}\mathbf{h}_{t-1} + \mathbf{b}_{hf}) \quad (2.17)$$

$$\mathbf{i}_t = \sigma(\mathbf{W}_{ii}\mathbf{x}_t + \mathbf{b}_{ii} + \mathbf{W}_{hi}\mathbf{h}_{t-1} + \mathbf{b}_{hi}) \quad (2.18)$$

$$\tilde{\mathbf{c}}_t = \tanh(\mathbf{W}_{i\tilde{c}}\mathbf{x}_t + \mathbf{b}_{i\tilde{c}} + \mathbf{W}_{h\tilde{c}}\mathbf{h}_{t-1} + \mathbf{b}_{h\tilde{c}}) \quad (2.19)$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_{io}\mathbf{x}_t + \mathbf{b}_{io} + \mathbf{W}_{ho}\mathbf{h}_{t-1} + \mathbf{b}_{ho}) \quad (2.20)$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{c}}_t \quad (2.21)$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t) \quad (2.22)$$

όπου οι πίνακες \mathbf{W} και τα διανύσματα μεροληψίας \mathbf{b} σε όλες τις παραπάνω εξισώσεις αποτελούν τις παραμέτρους προς εκπαίδευση του νευρωνικού δικτύου LSTM. Επίσης, το σύμβολο σ αναφέρεται στη σιγμοειδή συνάρτηση, ενώ το σύμβολο \odot ισοδυναμεί με το Hadamard γινόμενο, ή διαφορετικά το γινόμενο στοιχείο με στοιχείο μεταξύ διαφορετικών διανυσμάτων.

Τέλος, θα πρέπει να σημειωθεί ότι οι παραπάνω εξισώσεις αποτελούν μία ενδεικτική περίπτωση των υπολογισμών εντός ενός κελιού LSTM. Μία εναλλακτική λύση θα ήταν η συγχώνευση (concatenation) των διανυσμάτων \mathbf{x}_t και \mathbf{h}_{t-1} στις Εξ. (2.17), (2.18), (2.19) και (2.20) και στη συνέχεια, για κάθε μία περίπτωση το διάνυσμα που προκύπτει να πολλαπλασιαστεί με τον αντίστοιχο πίνακα \mathbf{W} . Με αυτόν τον τρόπο, σε κάθε μία από τις εξισώσεις αυτές θα υπήρχε μόνο ένας πίνακας \mathbf{W} και ένα διάνυσμα μεροληψίας \mathbf{b} .

2.3 Μέθοδοι Βελτιστοποίησης

Στην παρούσα ενότητα παρουσιάζονται ορισμένες από τις ευρέως γνωστές μεθόδους εκπαίδευσης ενός νευρωνικού δικτύου, όπως αυτές αναλύονται διεξοδικά από τους συγγραφείς στο [34]. Η επιλογή της μεθόδου αποτελεί ένα σπουδαίο παράγοντα κατά την εκπαίδευση του μοντέλου και συνεπώς θα πρέπει να αναφερθούν τα πλεονεκτήματα και τα μειονεκτήματα κάθε μίας για την επιτυχή ανανέωση των βαρών ενός νευρωνικού δικτύου. Να σημειωθεί ότι κατά την εφαρμογή όλων των αλγορίθμων πραγματοποιούνται δύο πέρασματα εντός του δικτύου. Το πρώτο είναι ένα εμπρόσθιο πέρασμα και αφορά την εκτίμηση της συνάρτησης κόστους χρησιμοποιώντας τα βάρη της προηγούμενης επανάληψης, ενώ αντίθετα το δεύτερο είναι ένα οπίσθιο πέρασμα και αφορά την ανανέωση των βαρών χρησιμοποιώντας έναν από τους αλγορίθμους που ακολουθούν.

2.3.1 Αλγόριθμος Καθόδου Κλίσης

Ο αλγόριθμος καθόδου κλίσης (gradient descent) θεωρείται ένας από τους απλούς αλγορίθμους που χρησιμοποιούνται κατά την εκπαίδευση ενός νευρωνικού δικτύου. Αν υποθέσουμε ότι L θεωρείται η συνάρτηση κόστους και θ είναι όλες οι παράμετροι ενός νευρωνικού δικτύου, τότε η μέθοδος αυτή ανανεώνει τις παραμέτρους σύμφωνα με την Εξ. (2.23):

$$\theta' = \theta - \eta \nabla_{\theta} L(\theta) \quad (2.23)$$

όπου η είναι ο συντελεστής εκμάθησης. Ο συγκεκριμένος αλγόριθμος, αν και θεωρείται αρκετά απλός ως προς την υλοποίησή του, δεν μπορεί να εφαρμοσθεί στις περιπτώσεις όπου ο όγκος των δεδομένων είναι αρκετά μεγάλος. Ο λόγος είναι ότι σε αυτές τις περιπτώσεις το σύνολο δεδομένων δε "χωράει" στη μνήμη RAM του υπολογιστή και συνεπώς η εκτίμηση της συνάρτησης κόστους θεωρείται αδύνατη. Για την αντιμετώπιση του συγκεκριμένου προβλήματος έχουν προταθεί 2 παραλλαγές της συγκεκριμένης μεθόδου, δηλαδή η μέθοδος mini-batch και stochastic gradient descent των Εξ. (2.24) και (2.25) αντίστοιχα:

$$\theta' = \theta - \eta \nabla_{\theta} L(\theta; x_{i:i+N}, y_{i:i+N}) \quad (2.24)$$

$$\theta' = \theta - \eta \nabla_{\theta} L(\theta; x_i, y_i) \quad (2.25)$$

Η μέθοδος mini-batch ουσιαστικά στηρίζεται σε ένα υποσύνολο N δεδομένων ικανό να "χωρέσει" στη μνήμη του υπολογιστή, ενώ αντίθετα η στοχαστική μέθοδος εκτιμά τη διανυσματική παράγωγο της συνάρτησης κόστους για κάθε ένα ζεύγος δεδομένων x_i, y_i ξεχωριστά. Η χρήση ενός υποσυνόλου δεδομένων κατά την εκτίμηση της διανυσματικής παραγωγού έχει αποδειχθεί ότι οδηγεί πιο γρήγορα στη σύγκλιση του αλγορίθμου, ενώ παράλληλα οι ανανεώσεις των παραμέτρων παρουσιάζουν μικρότερη διακύμανση συγκριτικά με την περίπτωση της στοχαστικής μεθόδου κλίσης.

2.3.2 Κάθοδος Κλίσης με Ορμή

Οι προαναφερθείσες παραλλαγές της μεθόδου καθόδου κλίσης στις περισσότερες περιπτώσεις καθυστερούν τη σύγκλιση σε κάποια συγκεκριμένη τιμή των παραμέτρων. Για το λόγο αυτό στην υποενοότητα αυτή παρουσιάζονται ορισμένοι αλγόριθμοι καθόδου κλίσης με ορμή (gradient descent with momentum), οι οποίοι οδηγούν σε μικρότερες διακυμάνσεις των τιμών των παραμέτρων και συνεπώς οδηγούν σε γρηγορότερη σύγκλισή τους.

Η λογική της μεθόδου καθόδου κλίσης με ορμή έγκειται στη χρήση της τεχνικής exponentially weighted moving average [35]. Συγκεκριμένα, αν υποθέσουμε ότι έχουμε ένα σύνολο δεδομένων το οποίο εξελίσσεται χρονικά, τότε για κάθε μία χρονική στιγμή η τεχνική αυτή εκτιμά την αναμενόμενη τιμή του συνόλου δεδομένων χρησιμοποιώντας τις τιμές των

προηγούμενων χρονικών στιγμών με κατάλληλα βάρη. Ουσιαστικά λοιπόν η μέθοδος αυτή εκτιμά έναν "κινητό" μέσο όρο που εξελίσσεται ως προς το χρόνο βάσει των τιμών των προηγούμενων χρονικών στιγμών. Αν λοιπόν ο "κινητός" αυτός μέσος όρος χρησιμοποιηθεί στις τιμές των διανυσματικών παραγώγων που εκτιμώνται διαδοχικά κατά τη διάρκεια της εκπαίδευσης, έχει αποδειχθεί ότι η σύγκλιση του αλγορίθμου βελτιώνεται.

Με χρήση λοιπόν της αναδρομικής Εξ. (2.26), η μέθοδος αυτή χρησιμοποιεί τις προηγούμενες χρονικά τιμές των διανυσματικών παραγώγων, ενώ η ανανέωση των βαρών πραγματοποιείται σύμφωνα με την Εξ. (2.27):

$$v_t = \gamma v_{t-1} + \eta \nabla_{\theta} L(\theta) \quad (2.26)$$

$$\theta' = \theta - v_t \quad (2.27)$$

όπου η ποσότητα γ αποτελεί συντελεστή της μεθόδου και τίθεται συνήθως ίσος με την τιμή 0.9. Οι αλγόριθμοι που έχουν αναπτυχθεί βασισμένοι στη ροπή ποικίλουν και για το λόγο αυτό ακολούθως παρουσιάζεται η πιο συνηθισμένη μέθοδος, η οποία χρησιμοποιείται σε πολλές εφαρμογές που στηρίζονται στη χρήση νευρωνικών δικτύων.

Αλγόριθμος Adaptive Momentum Estimation

Ο αλγόριθμος Adam [36] ουσιαστικά αποτελεί έναν συνδυασμό των μεθόδων Adagrad [37] και RMSprop [38]. Συγκεκριμένα, με χρήση της Εξ. (2.28) εκτιμάται ένας "κινητός" μέσος όρος των διανυσματικών παραγώγων για κάθε χρονική στιγμή t , ενώ παράλληλα με την Εξ. (2.29) πραγματοποιείται η ίδια διαδικασία και για τη μη κεντραρισμένη διασπορά (uncentered variance) των παραγώγων αυτών.

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) \nabla_{\theta} L(\theta) \quad (2.28)$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) (\nabla_{\theta} L(\theta))^2 \quad (2.29)$$

Για τη μείωση των σφαλμάτων στις αρχικές τιμές των παραπάνω ποσοτήτων πραγματοποιείται διόρθωση σύμφωνα με τις Εξ. (2.30) και (2.31), αντίστοιχα. Στις εξισώσεις αυτές ο παρανομαστής παύει να επιδρά μετά από ένα συγκεκριμένο πλήθος επαναλήψεων λόγω του όρου β^t .

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \quad (2.30)$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \quad (2.31)$$

Τέλος, η ανανέωση των βαρών του δικτύου πραγματοποιείται σύμφωνα με την Εξ. (2.32). Να σημειωθεί ότι συνήθεις τιμές των παραμέτρων είναι 0.9 για την παράμετρο β_1 , 0.999 για την παράμετρο β_2 και 10^{-8} για την ποσότητα ϵ .

$$\theta' = \theta - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon} \hat{m}_t \quad (2.32)$$

Ο αλγόριθμος Adam φαίνεται να λειτουργεί σε πολλές εφαρμογές και στις περισσότερες περιπτώσεις οδηγεί σε μικρότερες διακυμάνσεις των τιμών των παραμέτρων, με αποτέλεσμα την ταχύτερη σύγκλισή τους. Τέλος, θα πρέπει να τονισθεί ότι υπάρχουν πολλοί διαφορετικοί αλγόριθμοι που μπορούν να χρησιμοποιηθούν κατά την εκπαίδευση των νευρωνικών δικτύων (π.χ. Adadelta, Adamax, Nadam κ.α.), αλλά στην παρούσα εργασία η εξαγωγή των αποτελεσμάτων βασίζεται στη μέθοδο Adam που μόλις παρουσιάστηκε.

Κεφάλαιο 3

Νευρωνικά Δίκτυα Αυτοκωδικοποιητών

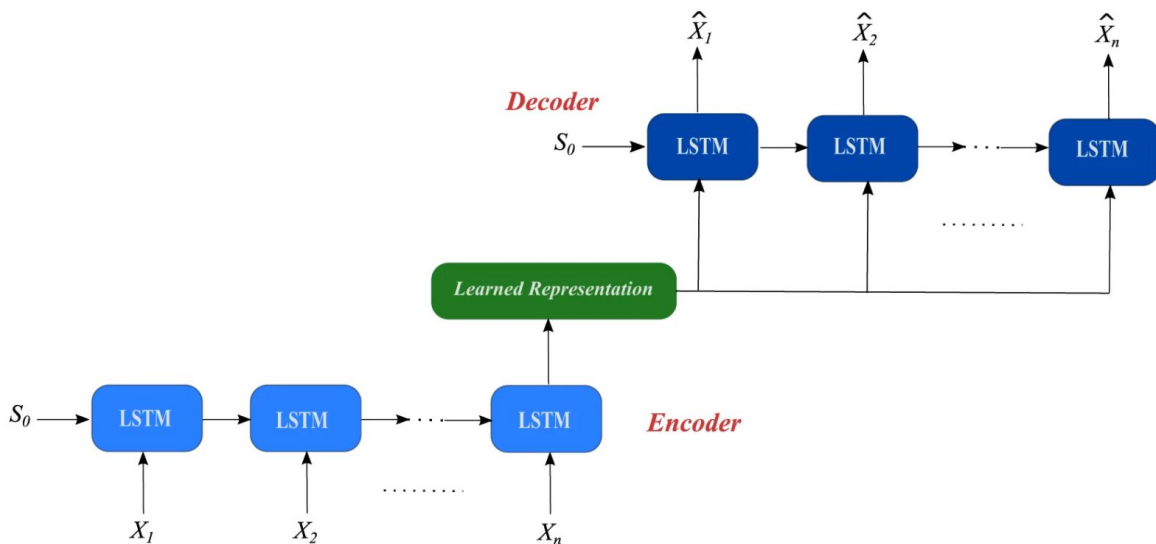
Στο παρόν Κεφάλαιο παρουσιάζονται τα νευρωνικά δίκτυα αυτοκωδικοποιητών, τα οποία μπορούν να προσαρμοστούν σε σήματα χρονοσειρών για την αποθυροβοποίηση και την εξαγωγή χαρακτηριστικών τους. Συγκεκριμένα, στην Εν. 3.1 πραγματοποιείται μία σύντομη αναφορά στους αυτοκωδικοποιητές σε συνδυασμό με τα νευρωνικά δίκτυα LSTM της Εν. 2.2, ενώ στην Εν. 3.2 παρουσιάζονται τα ίδια μοντέλα σε συνδυασμό όμως και με την Μπεϋζιανή θεωρία. Επιπλέον, στην Εν. 3.3 παρουσιάζεται η μεθοδολογία εκπαίδευσης των δικτύων αυτών καθώς και οι δυσκολίες που προκύπτουν κατά τη διαδικασία αυτή. Το βασικό πλεονέκτημα τέτοιου είδους γενικευμένων στατιστικών μοντέλων έναντι των αρχικών αυτοκωδικοποιητών έγκειται στο γεγονός ότι μοντελοποιούν την κατανομή των συμπιεσμένων δεδομένων, παρέχοντας τη δυνατότητα παραγωγής νέων δεδομένων σε περιπτώσεις που παρατηρείται έλλειψή τους.

3.1 LSTM Αυτοκωδικοποιητές

Οι αυτοκωδικοποιητές αποτελούν ένα είδος νευρωνικού δικτύου μη επιβλεπόμενης μάθησης, το οποίο εκπαιδεύεται ώστε να συμπιέζει και να κωδικοποιεί αποτελεσματικά τα δεδομένα εισόδου. Ακολούθως, εκπαιδεύεται έτσι ώστε να μπορεί να ανακατασκευάζει τα αρχικά δεδομένα χρησιμοποιώντας τη συμπιεσμένη αναπαράσταση [39]. Στόχος του δικτύου αυτού είναι να εξάγει μια αναπαράσταση δεδομένων όσο πιο κοντά γίνεται στην αρχική είσοδο. Ουσιαστικά λοιπόν ο αυτοκωδικοποιητής μειώνει τη διαστατικότητα των δεδομένων, μαθαίνοντας πώς να αγνοεί το θόρυβο από τα δεδομένα.

Το δίκτυο αποτελείται από 4 κύρια μέρη: τον κωδικοποιητή (encoder) που αποτελεί το μοντέλο μείωσης της διαστατικότητας, το μεσσαίο επίπεδο (bottleneck) που αποτελεί το επίπεδο της συμπιεσμένης αναπαράστασης των δεδομένων, τον αποκωδικοποιητή (decoder) δηλαδή το μοντέλο ανακατασκευής των αρχικών δεδομένων και το κόστος ανακατασκευής (reconstruction loss) που εκφράζει πόσο κοντά βρίσκεται η εξόδος του δικτύου στην αρχική είσοδο [40]. Οι αυτοκωδικοποιητές χρησιμοποιούνται σε ποικίλες εφαρμογές, όπως για παράδειγμα ανίχνευση παράτυπων σημείων-ανωμαλιών (anomaly detection), αποθρομβοποίηση εικόνων (image denoising), μείωση διαστατικότητας δεδομένων (dimensionality reduction), συσταδοποίηση (clustering) και δημιουργία νέων δεδομένων (generator) [41–43].

Στην παρούσα ενότητα παρουσιάζονται οι αυτοκωδικοποιητές που συνδυάζονται με τα νευρωνικά δίκτυα LSTM (δες Εν. 2.2), η αρχιτεκτονική των οποίων απεικονίζεται στο Σχ. 3.1. Ουσιαστικά λοιπόν η διαφορά των συγκεκριμένων μοντέλων από τα απλά μοντέλα αυτοκωδικοποιητών έγκειται στο γεγονός ότι τόσο στο τμήμα του κωδικοποιητή όσο και στον αποκωδικοποιητή χρησιμοποιούνται νευρωνικά δίκτυα LSTM έναντι των απλών πολυστρωματικών δικτύων MLP (Multilayer Perceptron), με στόχο την συμπίεση και την εξαγωγή χαρακτηριστικών των χρονοσειρών στο μεσσαίο επίπεδο. Η αρχιτεκτονική αυτή θα αναφέρεται από εδώ και στο εξής ως LSTM AutoEncoder ή συντομογραφικά ως LSTM-AE.



Σχήμα 3.1: Αρχιτεκτονική αυτοκωδικοποιητή σε συνδυασμό με τα νευρωνικά δίκτυα LSTM, χρησιμοποιώντας τη ξεδιπλωμένη μορφή του μη κυκλικού υπολογιστικού γράφου. Πηγή [44].

Αν λοιπόν το τμήμα του κωδικοποιητή θεωρείται ως ένας μη γραμμικός μετασχηματισμός f του αρχικού σήματος $\mathbf{x} \in \mathbb{R}^D$ με παραμέτρους ϕ , τότε τα συμπιεσμένα δεδομένα $\mathbf{z} \in \mathbb{R}^d$ προκύπτουν ως: $\mathbf{z} = f_{\phi}(\mathbf{x})$. Ακολούθως, τα συμπιεσμένα δεδομένα \mathbf{z} , μέσω ξανά του μη γραμμικού μετασχηματισμού g με παραμέτρους θ στο τμήμα του αποκωδικοποιητή επιδιώκουν την ανακατασκευή του σήματος \mathbf{x} μέσα από τις εκτιμήσεις: $\hat{\mathbf{x}} = g_{\theta}(\mathbf{z}) = g_{\theta}(f_{\phi}(\mathbf{x}))$.

Ως συνάρτηση κόστους για ένα υποσύνολο δεδομένων \mathbf{X} (γνωστό ως batch, το οποίο αποτελείται από N δεδομένα εισόδου \mathbf{x}) χρησιμοποιείται συνήθως η Εξ. (3.1):

$$L = \underbrace{\frac{1}{N} \sum_{i=1}^N \|\mathbf{x}_i - g_{\theta}(f_{\phi}(\mathbf{x}_i))\|_2^2}_{\text{reconstruction loss}} + \underbrace{\beta \sum_{i=1}^M |w_i|}_{\text{regularization term}} \quad (3.1)$$

όπου η μεταβλητή w_i αποτελεί κάθε μία από τις συνολικά M παραμέτρους που προκύπτουν από το τμήμα του κωδικοποιητή (με παραμέτρους ϕ) και του αποκωδικοποιητή (με παραμέτρους θ). Ο πρώτος όρος της Εξ. (3.1) ουσιαστικά αποτελεί το κόστος ανακατασκευής των σημάτων (δηλαδή το μέσο τετραγωνικό σφάλμα δεδομένων εισόδου-εξόδου), ενώ ο δεύτερος συνεισφέρει ως ποινικοποίηση των βαρών των 2 δικτύων. Ο λόγος που χρησιμοποιείται συνήθως η L1 νόρμα (ή σε αρκετές περιπτώσεις η L2 νόρμα) ως ποινικοποίηση είναι για να διατηρηθούν τα βάρη σε όσο το δυνατόν χαμηλότερες τιμές, ώστε κατά την εκμάθηση των παραμέτρων του δικτύου να αποφευχθούν φαινόμενα υπερεκπαίδευσης [26]. Τέλος, η παράμετρος β είναι επιλεγμένη παράμετρος με συγκεκριμένη τιμή.

Η εκπαίδευση των βαρών του δικτύου πραγματοποιείται με χρήση του αλγορίθμου οπισθοδιάδοσης του σφάλματος δια μέσω του χρόνου (BPTT). Κατά την εκτέλεση του αλγορίθμου αυτού πραγματοποιούνται δύο πέρασματα εντός του δικτύου. Το πρώτο πέρασμα αφορά την εκτίμηση της συνάρτησης κόστους της Εξ. (3.1) χρησιμοποιώντας τα βάρη της προηγούμενης επανάληψης, ενώ το δεύτερο πέρασμα αφορά την ανανέωση των βαρών χρησιμοποιώντας έναν από τους αλγορίθμους Gradient Descent, Adaptive Moment Estimation (Adam), Adagrad, Adadelta κ.α., όπως αυτοί αναλύονται διεξοδικά από τους συγγραφείς στο [34].

3.2 Variational Αυτοκωδικοποιητές

Η παρούσα εργασία ασχολείται κυρίως με τους variational αυτοκωδικοποιητές (Variational AutoEncoders ή VAE), οι οποίοι αποτελούν μια παραλλαγή των παραδοσιακών αυτοκωδικοποιητών έχοντας όμως την ίδια αρχιτεκτονική. Τα δίκτυα αυτά συνδυάζουν την αρχιτεκτονική των νευρωνικών δικτύων με τη Μπεϋζιανή θεωρία και αποτελούν γενικευμένα μοντέλα (generative models) νευρωνικών δικτύων που παρέχουν τη δυνατότητα όχι μόνο ανακατασκευής των αρχικών δεδομένων εισόδου, αλλά και παραγωγής νέων δεδομένων από μία συγκεκριμένη κατανομή. Πριν όμως την ανάλυση των νευρωνικών δικτύων VAE απαιτείται η ανάλυση και η κατανόηση της στατιστικής μεθόδου των Κύριων Συνιστωσών (Probabilistic Principal Component Analysis - PPCA), η οποία αποτελεί έναν αντίστοιχο στατιστικό τρόπο εξαγωγής χαρακτηριστικών μετασχηματίζοντας όμως με γραμμικό τρόπο τα δεδομένα εισόδου [45]. Η μέθοδος λοιπόν αυτή παρουσιάζεται στην υποενότητα που ακολουθεί.

3.2.1 Πιθανοτική Μέθοδος Κύριων Συνιστωσών

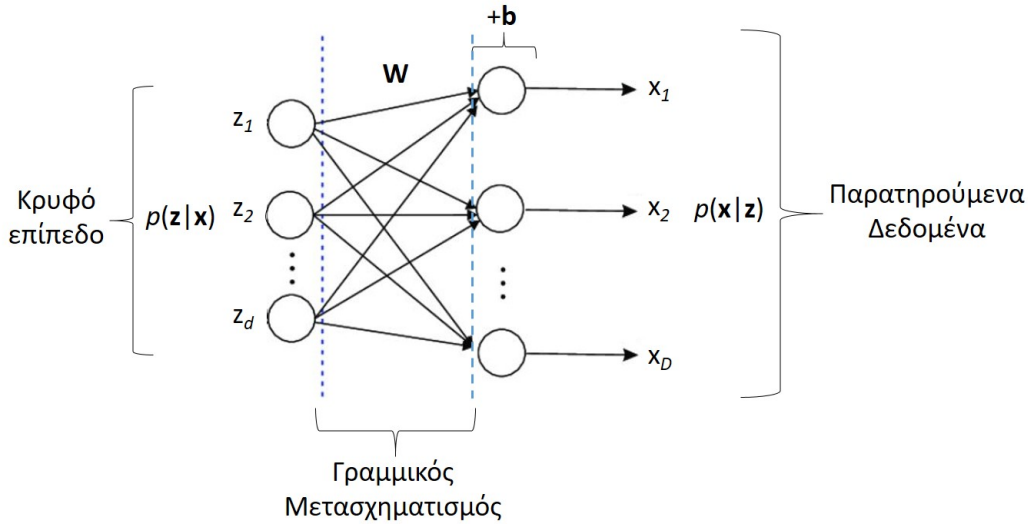
Έστω λοιπόν ένα σήμα εισόδου $\mathbf{x} \in \mathbb{R}^D$, το οποίο προκύπτει από τα κρυφά (ή συμπιεσμένα) δεδομένα $\mathbf{z} \in \mathbb{R}^d$ (όπου $D \gg d$) μέσω ενός γραμμικού μετασχηματισμού \mathbf{W} (δες Σχ. 3.2). Τότε, το στατιστικό μοντέλο που περιγράφει τα δεδομένα αυτά περιγράφεται από τις Εξ. (3.2) και (3.3):

$$p(\mathbf{z}) = N(\mathbf{0}, \mathbf{I}) \quad (3.2)$$

$$p(\mathbf{x}|\mathbf{z}) = N(\mathbf{W}\mathbf{z} + \mathbf{b}, \sigma^2\mathbf{I}) \quad (3.3)$$

όπου $p(\mathbf{z})$ η πρότερη κατανομή των κρυφών δεδομένων (prior distribution), $p(\mathbf{x}|\mathbf{z})$ η δεσμευμένη συνάρτηση πιθανοφάνειας (conditional likelihood), \mathbf{W} ο πίνακας γραμμικού μετασχηματισμού των δεδομένων \mathbf{z} , \mathbf{b} η σταθερά μεροληψίας (ή αλλιώς bias) και σ η τυπική απόκλιση των δεδομένων \mathbf{x} . Για το γραμμικό αυτό μετασχηματισμό των δεδομένων αρκεί λοιπόν η εκτίμηση των

προαναφερθόντων παραμέτρων. Ακολούθως περιγράφονται 2 ενδεικτικοί τρόποι εκτίμησης, όπως αυτοί παρουσιάζονται από τους συγγραφείς στο [45].



Σχήμα 3.2: Στατιστική μέθοδος κύριων συνιστωσών.

Μέθοδος Μέγιστης Πιθανοφάνειας

Για την εκτίμηση των παραμέτρων του στατιστικού μοντέλου αρκεί να εκτιμηθεί η συνάρτηση πιθανοφάνειας και στη συνέχεια να επιδιωχθεί η μεγιστοποίησή της [45]. Η συνάρτηση αυτή ουσιαστικά εκφράζει την πιθανότητα να παρατηρήσει κάποιος μόνο τα παρατηρούμενα δεδομένα, δηλαδή το σύνολο δεδομένων \mathbf{X} που αποτελείται από τα N σήματα \mathbf{x} . Εφόσον όμως στο συγκεκριμένο πρόβλημα υποθέτουμε ότι κάθε σήμα \mathbf{x}_i προέρχεται από την αντίστοιχη συμπιεσμένη αναπαράσταση \mathbf{z}_i μέσω ενός γραμμικού μετασχηματισμού, η συνάρτηση πιθανοφάνειας γράφεται:

$$p(\mathbf{X}|\theta) = \int_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\theta) d\mathbf{Z} = \prod_{i=1}^N \int_{\mathbf{z}} p(\mathbf{x}_i, \mathbf{z}_i|\theta) d\mathbf{z}_i = \prod_{i=1}^N \int_{\mathbf{z}} p(\mathbf{x}_i|\mathbf{z}_i, \theta) p(\mathbf{z}_i|\theta) d\mathbf{z}_i \longrightarrow$$

$$p(\mathbf{X}|\theta) = \prod_{i=1}^N \int_{\mathbf{z}} N(\mathbf{x}_i|\mathbf{W}\mathbf{z}_i + \mathbf{b}, \sigma^2\mathbf{I}) N(\mathbf{z}_i|\mathbf{0}, \mathbf{I}) d\mathbf{z}_i \quad (3.4)$$

όπου ουσιαστικά η παράμετρος θ , σύμφωνα με τις Εξ. (3.2) και (3.3), αποτελείται από τις παραμέτρους \mathbf{W} (διαστάσεων $D \times d$), το D -διάστατο διάνυσμα \mathbf{b} και την τυπική απόκλιση σ . Να σημειωθεί ότι κάθε σήμα \mathbf{x}_i θεωρείται ανεξάρτητο από τα υπόλοιπα $N-1$ σήματα και για το λόγο αυτό προκύπτει το γινόμενο στην Εξ. (3.4). Παρατηρώντας λοιπόν την παραπάνω εξίσωση έπεται ότι για την εκτίμηση της συνάρτησης πιθανοφάνειας θα πρέπει να εκτιμηθούν N ολοκληρώματα, κάθε ένα από τα οποία περιλαμβάνει το γινόμενο 2 κανονικών κατανομών (δηλαδή δεσμευμένης πιθανοφάνειας και πρότερης κατανομής) και στη συνέχεια να πραγματοποιηθεί το γινόμενο των επιμέρους αποτελεσμάτων.

Θεωρητικά, ο συγκεκριμένος υπολογισμός θεωρείται αρκετά χρονοβόρος υπολογιστικά και συνεπώς θα δυσχέραινε τη διαδικασία εκτίμησης της συνάρτησης πιθανοφάνειας. Σύμφωνα όμως με τη Μπεϋζιανή θεωρία, αν η δεσμευμένη πιθανοφάνεια $p(\mathbf{x}|\mathbf{z})$ ανήκει στην εκθετική οικογένεια κατανομών (π.χ. Gaussian, Bernoulli κ.α.), τότε υπάρχει σίγουρα μία πρότερη κατανομή $p(\mathbf{z})$ η οποία είναι συζυγής της πρώτης (conjugate prior) και η οποία οδηγεί σε γινόμενο των 2 αυτών κατανομών ίδιας μορφής με αυτήν της πρότερης. Εφόσον λοιπόν στην Εξ. (3.4) τόσο η δεσμευμένη συνάρτηση πιθανοφάνειας $p(\mathbf{x}|\mathbf{z})$ όσο και η πρότερη κατανομή $p(\mathbf{z})$ είναι Γκαουσιανές

κατανομές, το γινόμενο τους είναι θα είναι κι αυτό Γκαουσιανή κατανομή για την οποία μπορούμε να προσεγγίσουμε τις τιμές του ολοκληρώματός της. Με αυτόν τον τρόπο εκτιμούμε εύκολα τα N ολοκληρώματα και συνεπώς τη συνάρτηση πιθανοφάνειας $p(\mathbf{X}|\theta)$, ενώ ακολούθως πραγματοποιούμε την παραγωγή της για την εκτίμηση των παραμέτρων \mathbf{W} , \mathbf{b} και σ .

Μέθοδος Expectation Maximization

Ο αλγόριθμος Expectation Maximization (EM) είναι ένας ευρέως γνωστός επαναληπτικός αλγόριθμος που χρησιμοποιείται σε προβλήματα όπου υπάρχουν κρυφές τιμές [45], όπως δηλαδή στο πρόβλημα που εξετάζεται. Αποτελείται από 3 βήματα, τα οποία περιγράφονται συνοπτικά ακολούθως:

1. E-Step: Στο βήμα αυτό, χρησιμοποιώντας τον κανόνα του Bayes αρχικά εκτιμάται η εκ των υστέρων κατανομή των κρυφών δεδομένων (posterior distribution) σύμφωνα με την Εξ. (3.5).

$$p(\mathbf{Z}|\mathbf{X}, \theta^{old}) = \frac{p(\mathbf{X}|\mathbf{Z}, \theta^{old})p(\mathbf{Z}|\theta^{old})}{\int_{\mathbf{Z}} p(\mathbf{X}|\mathbf{Z}, \theta^{old})p(\mathbf{Z}|\theta^{old})d\mathbf{Z}} \rightarrow$$

$$p(\mathbf{Z}|\mathbf{X}, \theta^{old}) = \frac{\prod_{i=1}^N N(\mathbf{x}_i|\mathbf{W}\mathbf{z}_i + \mathbf{b}, \sigma^2\mathbf{I}) N(\mathbf{z}_i|\mathbf{0}, \mathbf{I})}{\prod_{i=1}^N \int_{\mathbf{z}} N(\mathbf{x}_i|\mathbf{W}\mathbf{z}_i + \mathbf{b}, \sigma^2\mathbf{I}) N(\mathbf{z}_i|\mathbf{0}, \mathbf{I}) d\mathbf{z}_i} \quad (3.5)$$

Να σημειωθεί ότι τόσο ο αριθμητής όσο και ο παρονομαστής της Εξ. (3.5) μπορούν να εκτιμηθούν λόγω της συζυγούς πρότερης κατανομής ως προς τη δεσμευμένη συνάρτηση πιθανοφάνειας, όπως δηλαδή εξηγήθηκε προηγουμένως.

2. M-Step: Στο βήμα αυτό αρχικά εκτιμάται η λογαριθμοποιημένη πιθανοφάνεια του συνόλου δεδομένων, δηλαδή του ζεύγους $\{\mathbf{X}, \mathbf{Z}\}$. Η συνάρτηση αυτή ορίζεται ως: $l(\theta) = \log p(\mathbf{X}, \mathbf{Z}|\theta)$. Εφόσον οι τιμές \mathbf{Z} είναι άγνωστες, ακολούθως εκτιμάται η αναμενόμενη τιμή της ποσότητας $l(\theta)$ ως προς τις τιμές \mathbf{Z} , σύμφωνα με την Εξ. (3.6):

$$Q(\theta, \theta^{old}) = E_{\mathbf{Z}|\mathbf{X}, \theta^{old}}\{l(\theta)\} = \int_{\mathbf{Z}} \log p(\mathbf{X}, \mathbf{Z}|\theta) \cdot p(\mathbf{Z}|\mathbf{X}, \theta^{old})d\mathbf{Z} \quad (3.6)$$

Ύστερα από την εκτίμηση της ποσότητας $Q(\theta, \theta^{old})$, ουσιαστικά πραγματοποιείται η μεγιστοποίησή της ως προς τις παραμέτρους θ . Κατά συνέπεια, προκύπτει το εξής: $\theta^{new} = \operatorname{argmax}_{\theta} Q(\theta, \theta^{old})$.

3. Έλεγχος τερματισμού σύμφωνα με το κριτήριο σύγκλισης. Διαφορετικά θέτουμε τη μεταβλητή $\theta^{old} \leftarrow \theta^{new}$ και επιστροφή στο Βήμα 1.

Το ερώτημα όμως που τίθεται είναι γιατί κάποιος να επιλέξει έναν επαναληπτικό αλγόριθμο για την εκτίμηση των παραμέτρων \mathbf{W} , \mathbf{b} και σ , εφόσον υπάρχει μία αναλυτική λύση για το ίδιο πρόβλημα όπως η μέθοδος της μέγιστης πιθανοφάνειας. Ορισμένα λοιπόν από τα πλεονεκτήματα του αλγορίθμου EM έναντι με της μεθόδους μέγιστης πιθανοφάνειας παρουσιάζονται ακολούθως:

1. Ο κύριος λόγος είναι η υπολογιστική πολυπλοκότητα, καθώς η αναλυτική λύση της μέγιστης πιθανοφάνειας οδηγεί σε πολυπλοκότητα $O(ND^2)$, σε αντίθεση με τον αλγόριθμο EM όπου η αντίστοιχη πολυπλοκότητα μειώνεται σε $O(NDd)$ (όπου N το πλήθος του συνόλου δεδομένων, ενώ D και d οι διαστατικότητες των παρατηρούμενων και συμπιεσμένων δεδομένων, αντίστοιχα). Αν λοιπόν η αρχική διάσταση των δεδομένων ισούται με $D = 1000$ και η συμπιεσμένη διάσταση με $d = 10$, τότε η υπολογιστική πολυπλοκότητα της μέγιστης πιθανοφάνειας είναι κατά 100 φορές μεγαλύτερη από την αντίστοιχη του αλγορίθμου EM.

2. Ο αλγόριθμος EM μπορεί να χρησιμοποιηθεί και σε πιο πολύπλοκα μοντέλα, όπως για παράδειγμα μοντέλα μειγμάτων πιθανοτικής ανάλυσης σε κύριες συνιστώσες (mixtures of PCA).
3. Ο αλγόριθμος EM μπορεί παράλληλα να χρησιμοποιηθεί για την εκτίμηση των παραμέτρων του μοντέλου και σε περιπτώσεις όπου υπάρχουν και άγνωστα δεδομένα στο σύνολο \mathbf{X} , σε αντίθεση με τη μέγιστη πιθανοφάνεια κ.α.

Σημείωση: Παρόλο που ο αλγόριθμος EM παρουσιάζει τα παραπάνω πλεονεκτήματα, σε αρκετές περιπτώσεις δεν εκτιμά τις βέλτιστες παραμέτρους \mathbf{W} , \mathbf{b} και σ όπως η μέθοδος μέγιστης πιθανοφάνειας. Αυτό συμβαίνει διότι κατά την επαναληπτική διαδικασία μπορεί να εγκλωβιστεί σε τοπικά μέγιστα, χωρίς να έχει τη δυνατότητα απεγκλωβισμού. Παρόλα αυτά όμως, σε περιπτώσεις όπου η διαστατικότητα των δεδομένων παραμένει μεγάλη, οι λύσεις του αλγορίθμου EM κρίνονται αρκετά ικανοποιητικές.

Τέλος, να σημειωθεί ότι και με τις 2 προαναφερθείσες μεθόδους, μόλις πραγματοποιηθεί η εκτίμηση των παραμέτρων \mathbf{W} , \mathbf{b} και σ , οι συμπιεσμένες αναπαραστάσεις \mathbf{z}_i των παρατηρούμενων δεδομένων \mathbf{x}_i προκύπτουν βάσει του [45] ως εξής:

$$\mathbf{z}_i = \frac{1}{\sigma^2} \Sigma \mathbf{W}^T (\mathbf{x}_i - \mathbf{b}) \quad (3.7)$$

$$\text{όπου } \Sigma = \left(\frac{1}{\sigma^2} \mathbf{W}^T \mathbf{W} + \mathbf{I} \right)^{-1} \quad (3.8)$$

3.2.2 Μη Γραμμικός Μετασχηματισμός Δεδομένων

Το ερώτημα που τίθεται στις προηγούμενες προσεγγίσεις είναι πώς θα αντιμετωπιζόταν το ίδιο πρόβλημα, αν αντί για γραμμικό μετασχηματισμό των δεδομένων \mathbf{z}_i στις Εξ. (3.2) και (3.3) χρησιμοποιούσαμε ένα μη γραμμικό μετασχηματισμό, όπως για παράδειγμα ένα νευρωνικό δίκτυο; Με αυτόν τον τρόπο, το μοντέλο που θα περιέγραφε τα δεδομένα θα μπορούσε να αναπαρασταθεί από τις Εξ. (3.9) και (3.10):

$$p(\mathbf{z}) = N(\mathbf{0}, \mathbf{I}) \quad (3.9)$$

$$p_{\theta}(\mathbf{x}|\mathbf{z}) = N(g_{\theta}(\mathbf{z}), \sigma^2 \mathbf{I}) \quad (3.10)$$

όπου η μόνη διαφορά των παραπάνω εξισώσεων από τις Εξ. (3.2) και (3.3) έγκειται στο μη γραμμικό μετασχηματισμό g_{θ} , ο οποίος θα μπορούσε να θεωρηθεί ένα νευρωνικό δίκτυο (αντίστοιχο του αποκωδικοποιητή) με τις παραμέτρους θ να αποτελούν τα βάρη του δικτύου. Να σημειωθεί ότι ο μη γραμμικός μετασχηματισμός δεδομένων χρησιμοποιείται κυρίως σε περιπτώσεις όπου ο όγκος των δεδομένων είναι αρκετά μεγάλος και επιπλέον η πολυπλοκότητα των δεδομένων δεν μπορεί να μοντελοποιηθεί με γραμμικούς μετασχηματισμούς (δεδομένα βίντεο, πολύπλοκα σχήματα κτλ.).

Το πρόβλημα που προκύπτει από τη χρήση του μη γραμμικού μετασχηματισμού είναι ότι η ιδιότητα της συζυγούς πρότερης κατανομής $p(\mathbf{z})$ ως προς τη δεσμευμένη συνάρτηση πιθανοφάνειας $p_{\theta}(\mathbf{x}|\mathbf{z})$ χάνεται και συνεπώς, τόσο κατά τη μέθοδο της μέγιστης πιθανοφάνειας (δες Εξ. (3.4)) όσο και κατά το E-step του αλγορίθμου EM (δες παρονομαστή της Εξ. (3.5)) τα αντίστοιχα ολοκληρώματα μετατρέπονται σε απροσέγγιστα (intractable integrals). Άρα, οι έως τώρα μέθοδοι καθίστανται ανίκανες να εκτιμήσουν τις παραμέτρους του μοντέλου που ορίζονται από τις Εξ. (3.9) και (3.10).

Μία ενδεικτική λύση για την αντιμετώπιση του συγκεκριμένου προβλήματος είναι να προσεγγιστεί η εκ των υστέρων κατανομή $p(\mathbf{Z}|\mathbf{X})$ στο E-step του αλγορίθμου EM (δες Εξ. (3.5)) μέσα από τη θεωρία της variational συμπερασματολογίας (variational inference). Η μέθοδος αυτή μας επιτρέπει να προσεγγίσουμε μια υψηλού διαστάσεων κατανομή με μια απλούστερη, λύνοντας

ένα πρόβλημα βελτιστοποίησης. Ακολούθως λοιπόν περιγράφεται η συγκεκριμένη μέθοδος βάσει της οποίας επιλύθηκε το πρόβλημα της χρήσης μη γραμμικών μετασχηματισμών στα συμπιεσμένα δεδομένα.

Πιο αναλυτικά, για δύο κατανομές $p(\mathbf{Z}|\mathbf{X})$ και $q(\mathbf{Z})$ ορίζεται ως απόκλιση (divergence) D μια μεταβλητή που μετρά τη διαφορά μεταξύ των κατανομών, έτσι ώστε $D(q(\mathbf{Z})||p(\mathbf{Z}|\mathbf{X})) \geq 0$ και $D(q(\mathbf{Z})||p(\mathbf{Z}|\mathbf{X})) = 0$ για $q(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X})$. Συνήθως χρησιμοποιείται η Kullback-Leibler (KL) divergence [46, 47], η οποία συχνά αναφέρεται και ως σχετική εντροπία (relative entropy) ή κέρδος (information gain) και δίνεται από την Εξ. (3.11).

Στόχος της variational συμπερασματολογίας είναι ο καθορισμός μιας κατανομής $q_\lambda(\mathbf{Z})$ (όπου λ οι παράμετροι της κατανομής αυτής, όπως π.χ. για μία Γκαουσιανή κατανομή το διάνυσμα της μέσης τιμής $\boldsymbol{\mu}$ και ο πίνακας συνδιασπορών $\boldsymbol{\Sigma}$), η οποία να βρίσκεται όσο το δυνατόν πιο κοντά στην εκ των υστέρων κατανομή $p(\mathbf{Z}|\mathbf{X})$ και το οποίο επιτυγχάνεται ελαχιστοποιώντας την KL divergence:

$$D_{KL}(q_\lambda(\mathbf{Z})||p(\mathbf{Z}|\mathbf{X})) = \int_{\mathbf{Z}} q_\lambda(\mathbf{Z}) \log \frac{q_\lambda(\mathbf{Z})}{p(\mathbf{Z}|\mathbf{X})} d\mathbf{Z} = E_q \left[\log \frac{q_\lambda(\mathbf{Z})}{p(\mathbf{Z}|\mathbf{X})} \right] \rightarrow$$

$$D_{KL}(q_\lambda(\mathbf{Z})||p(\mathbf{Z}|\mathbf{X})) = E_q[\log q_\lambda(\mathbf{Z})] - E_q[\log p(\mathbf{X}, \mathbf{Z})] + \log p(\mathbf{X}) \quad (3.11)$$

Στόχος λοιπόν αποτελεί η εκτίμηση των παραμέτρων λ της κατανομής q_λ που ελαχιστοποιούν την παραπάνω απόσταση. Συνεπώς, η βέλτιστη προσέγγιση (approximation) της εκ των υστέρων κατανομής προκύπτει από την Εξ. (3.12):

$$q_\lambda^*(\mathbf{Z}) = \operatorname{argmin}_\lambda D_{KL} \quad (3.12)$$

Το ερώτημα όμως που τίθεται είναι πώς μπορεί να ελαχιστοποιηθεί η απόσταση μιας κατανομής $q_\lambda(\mathbf{Z})$ από την εκ των υστέρων κατανομή $p(\mathbf{Z}|\mathbf{X})$ όταν η δεύτερη δεν είναι γνωστή; Αυτό σημαίνει ότι το πρόβλημα ελαχιστοποίησης της Εξ. (3.12) θα πρέπει να τροποποιηθεί κατάλληλα ώστε να είναι εφικτή η εκτίμηση της συνάρτησης $q_\lambda(\mathbf{Z})$. Θεωρώντας λοιπόν ως -ELBO (Evidence Lower Bound) τους 2 πρώτους όρους του δεξιού τμήματος της Εξ. (3.11), τότε η σχέση αυτή μπορεί να γραφεί ως εξής:

$$ELBO(\lambda) = E_q[\log p(\mathbf{X}, \mathbf{Z})] - E_q[\log q_\lambda(\mathbf{Z})] \quad (3.13)$$

$$\text{και } \log p(\mathbf{X}) = ELBO(\lambda) + D_{KL}(q_\lambda(\mathbf{Z})||p(\mathbf{Z}|\mathbf{X})) \quad (3.14)$$

Παρατηρώντας την Εξ. (3.14) είναι γνωστό ότι η απόσταση Kullback-Leibler στο δεξί τμήμα της εξίσωσης είναι ένας αριθμός μεγαλύτερος ή ίσος του μηδενός και συνεπώς η συγκεκριμένη εξίσωση γράφεται:

$$\log p(\mathbf{X}) \geq ELBO(\lambda) \quad (3.15)$$

Παρατηρείται λοιπόν ότι η ποσότητα ELBO αποτελεί ένα κάτω όριο της λογαριθμοποιημένης συνάρτησης πιθανοφάνειας. Συνεπώς, εφόσον δεν μπορούμε να μεγιστοποιήσουμε τη συνάρτηση αυτή, εναλλακτικά θα μπορούσαμε να μεγιστοποιήσουμε ένα κάτω όριό της. Άρα, το πρόβλημα της ελαχιστοποίησης της απόστασης KL μεταξύ των κατανομών $q(\mathbf{Z})$ και $p(\mathbf{Z}|\mathbf{X})$ ισοδυναμεί με τη μεγιστοποίηση του όρου ELBO. Από την παρατήρηση αυτή προκύπτει η ονομασία του όρου ELBO (Evidence Lower Bound), καθώς αποτελεί την κατώτερη τιμή της κατανομής του συνόλου δεδομένων \mathbf{X} (γνωστή και ως evidence), δηλαδή ισχύει $\log p(\mathbf{X}) \geq ELBO(\lambda)$. Συνεπώς, η βέλτιστη προσέγγιση της εκ των υστέρων κατανομής προκύπτει από την Εξ. (3.16):

$$q_\lambda^*(\mathbf{Z}) = \operatorname{argmax}_\lambda ELBO(\lambda) \quad (3.16)$$

Σημείωση: Η ποσότητα ELBO εκτιμάται σύμφωνα με την Εξ. (3.13). Η συγκεκριμένη μορφή μπορεί να τροποποιηθεί αναλύοντας την απο κοινού κατανομή $p(\mathbf{X}, \mathbf{Z})$ ως γινόμενο της δεσμευμένης συνάρτησης πιθανοφάνειας $p_\theta(\mathbf{X}|\mathbf{Z})$ και της πρότερης κατανομής $p(\mathbf{Z})$. Συνεπώς, η ποσότητα ELBO γράφεται:

$$\begin{aligned} ELBO(\lambda, \theta) &= E_q[\log p_\theta(\mathbf{X}|\mathbf{Z})] + E_q[\log p(\mathbf{Z})] - E_q[\log q_\lambda(\mathbf{Z})] \longrightarrow \\ ELBO(\lambda, \theta) &= E_q[\log p_\theta(\mathbf{X}|\mathbf{Z})] - E_q \left[\log \frac{q_\lambda(\mathbf{Z})}{p(\mathbf{Z})} \right] \longrightarrow \\ ELBO(\lambda, \theta) &= \underbrace{E_q[\log p_\theta(\mathbf{X}|\mathbf{Z})]}_{\text{reconstruction likelihood}} - \underbrace{D_{KL}(q_\lambda(\mathbf{Z})||p(\mathbf{Z}))}_{\text{regularization term}} \end{aligned} \quad (3.17)$$

Συγκεντρωτικά λοιπόν η χρήση μίας μη γραμμικής συνάρτησης κατά το μετασχηματισμό των συμπιεσμένων δεδομένων έχει οδηγήσει έως τώρα στα εξής συμπεράσματα:

1. Η εκτίμηση των παραμέτρων του μοντέλου δεν μπορεί να πραγματοποιηθεί ούτε με τη μέθοδο μέγιστης πιθανοφάνειας αλλά και ούτε με τον αλγόριθμο EM, λόγω της εμφάνισης απροσέγγιστων ολοκληρωμάτων.
2. Αντί για τη μεγιστοποίηση της λογαριθμοποιημένης συνάρτησης πιθανοφάνειας $\log p(\mathbf{X})$ μπορούμε να μεγιστοποιήσουμε ένα κάτω όριο της, δηλαδή την ποσότητα ELBO της Εξ. (3.17). Η συνάρτηση κόστους αυτή αποτελείται από 2 ειδών παραμέτρους: (α) τις παραμέτρους λ της κατανομής $q_\lambda(\mathbf{Z})$ (π.χ. για Γκαουσιανή κατανομή το διάνυσμα της μέσης τιμής $\boldsymbol{\mu}$ και ο πίνακας συνδιασπορών $\boldsymbol{\Sigma}$) και (β) τις παραμέτρους θ του νευρωνικού δικτύου.

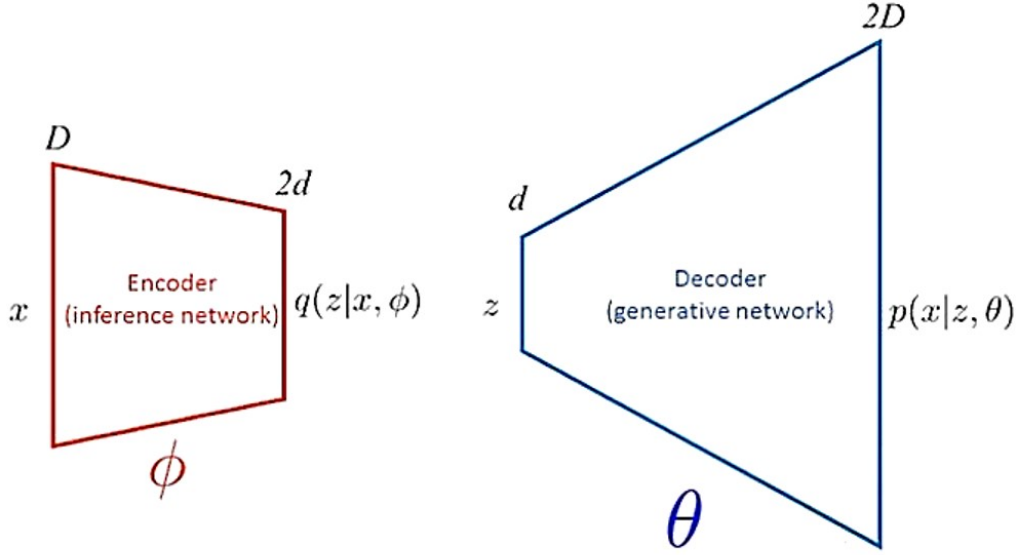
Για την εκτίμηση των 2 αυτών παραμέτρων λ, θ θα μπορούσε να χρησιμοποιηθεί η στοχαστική variational συμπερασματολογία (Stochastic Variational Inference - SVI) [48], η οποία βασίζεται στην εναλλάξ παραγωγή της ποσότητας ELBO πρώτα ως προς τη μία παράμετρο και στη συνέχεια ως προς την άλλη, ώστε επαναληπτικά να πραγματοποιηθεί η μεγιστοποίησή της. Ουσιαστικά λοιπόν αυτό θα παρομοιαζόταν με τον αλγόριθμο EM, τα 2 βήματα του οποίου θεωρούνται πια τα εξής:

1. Στο E-step εκτιμάται η κατανομή $q_\lambda(\mathbf{Z})$, η οποία θεωρείται προσέγγιση (approximation) της εκ των υστέρων κατανομής $p(\mathbf{Z}|\mathbf{X})$ που υπολογιζόταν στο αντίστοιχο βήμα του EM αλγορίθμου της Εξ. (3.5).
2. Στο M-step εκτιμώνται οι παράμετροι θ του νευρωνικού δικτύου, οι οποίες θεωρούνται αντίστοιχες με τις παραμέτρους \mathbf{W} και \mathbf{b} κατά το γραμμικό μετασχηματισμό της μεθόδου PPCA (δες υποενότητα 3.2.1).

Τέλος, να σημειωθεί ότι παραμένει ένα τελευταίο ερώτημα το οποίο δυσχεραίνει την αντιμετώπιση του συγκεκριμένου προβλήματος. Αν υποθέσουμε ότι έχουν εκτιμηθεί οι παράμετροι λ της κατανομής $q_\lambda(\mathbf{z})$ κάθε δεδομένου εκπαίδευσης καθώς και οι παράμετροι του νευρωνικού δικτύου θ , τότε για κάθε νέο δεδομένο δε θα μπορούσε να εκτιμηθεί η αντίστοιχη συμπιεσμένη αναπαράσταση. Αυτό συμβαίνει διότι: (1) οι παράμετροι λ έχουν εκτιμηθεί μόνο για κάθε ένα από τα δεδομένα του συνόλου εκπαίδευσης και (2) η οπίσθια τροφοδότηση των δεδομένων εξόδου \mathbf{x}_i στο νευρωνικό δίκτυο είναι αδύνατη λόγω της ύπαρξης μη γραμμικοτήτων. Αντιθέτως, αυτό δε συμβαίνει στο γραμμικό μετασχηματισμό της μεθόδου PPCA, όπου μπορεί να εκτιμηθεί ο ανάστροφος πίνακας \mathbf{W}^T και συνεπώς η συμπιεσμένη αναπαράσταση \mathbf{z}_i σύμφωνα με την Εξ. (3.7).

Η λύση προτάθηκε λοιπόν από τους συγγραφείς στο [49], οι οποίοι πρότειναν να χρησιμοποιηθεί ένα δεύτερο νευρωνικό δίκτυο με παραμέτρους ϕ ως κωδικοποιητής (inference

network), το οποίο θα δέχεται τα δεδομένα εισόδου και θα καταλήγει στις παραμέτρους λ για τα δεδομένα εκπαίδευσης (δες Σχ. 3.3). Με αυτόν τον τρόπο, κατά την εμφάνιση ενός νέου δεδομένου θα μπορούσε να χρησιμοποιηθεί ο συγκεκριμένος κωδικοποιητής ώστε να εκτιμηθεί η ζητούμενη συμπιεσμένη αναπαράστασή του. Βέβαια, να σημειωθεί ότι το επιπλέον αυτό νευρωνικό δίκτυο προσθέτει υπολογιστικό κόστος, αλλά καθίσταται αναγκαίο για τη συμπίεση δεδομένων τα οποία δεν ανήκουν στο σύνολο εκπαίδευσης.



Σχήμα 3.3: Αρχιτεκτονική variational αυτοκωδικοποιητή.

Συνεπώς, ύστερα από τη προσθήκη του κωδικοποιητή για την εκτίμηση των παραμέτρων λ , απαιτείται η εκτίμηση των παραμέτρων ϕ και θ που μεγιστοποιούν την ποσότητα $ELBO(\phi, \theta)$ της Εξ. (3.18):

$$ELBO(\phi, \theta) = \underbrace{E_q[\log p_\theta(\mathbf{X}|\mathbf{Z})]}_{\text{reconstruction likelihood}} - \underbrace{D_{KL}(q_\lambda(\mathbf{Z}|\mathbf{X}, \phi) || p(\mathbf{Z}))}_{\text{regularization term}} \quad (3.18)$$

Σύμφωνα λοιπόν με το Σχ. 3.3, κατά την εκπαίδευση του νευρωνικού δικτύου η πορεία που ακολουθεί ένα τυχαίο σήμα \mathbf{x} εντός του δικτύου είναι η εξής:

1. Αρχικά, το σήμα $\mathbf{x} \in \mathbb{R}^D$ εισέρχεται στον κωδικοποιητή με παραμέτρους ϕ και καταλήγει ενδεικτικά σε μία Γκαουσιανή κατανομή $q_\lambda(\mathbf{z}|\mathbf{x}, \phi)$, οι παράμετροι λ της οποίας αποτελούνται από το d -διάστατο διάνυσμα $\boldsymbol{\mu}$ και τα d σε πλήθος διαγώνια στοιχεία του πίνακα συνδιασπορών $\boldsymbol{\Sigma}$ διαστάσεων $d \times d$.
2. Ύστερα από την εκτίμηση της κατανομής $q_\lambda(\mathbf{z}|\mathbf{x}, \phi)$ πραγματοποιείται δειγματοληψία ενός τυχαίου δείγματος, παράγοντας τη συμπιεσμένη αναπαράσταση $\mathbf{z} \in \mathbb{R}^d$.
3. Το συγκεκριμένο δείγμα οδηγείται στο τμήμα του αποκωδικοποιητή με παραμέτρους θ για την εκτίμηση της δεσμευμένης συνάρτησης πιθανοφάνειας $p_\theta(\mathbf{x}|\mathbf{z})$.
4. Τέλος, εφόσον όλες οι ζητούμενες κατανομές είναι γνωστές, εκτιμάται η ποσότητα $ELBO$ σύμφωνα με την Εξ. (3.18).

Σημείωση: Ο λόγος που μοντελοποιούνται μόνο τα διαγώνια στοιχεία του πίνακα $\boldsymbol{\Sigma}$ είναι διότι το πλήθος των στοιχείων του πίνακα αυτού αυξάνονται εκθετικά με την αύξηση της διαστατικότητας των συμπιεσμένων δεδομένων. Το πρόβλημα αυτό είναι γνωστό ως κατάρα των μεγάλων διαστάσεων (ή διαφορετικά curse of dimensionality).

Τέλος, θα πρέπει να σημειωθεί ότι η χρήση ενός νευρωνικού δικτύου ως κωδικοποιητή προσθέτει ένα επιπλέον σφάλμα μοντελοποίησης, το οποίο είναι γνωστό ως *amortization error*. Αυτό προκύπτει από το γεγονός ότι διαφορετικά δεδομένα εκπαίδευσης "μοιράζονται" (*amortize*) τις ίδιες παραμέτρους ϕ του νευρωνικού, ώστε να καταλήξουν στις παραμέτρους λ της κατανομής $q_\lambda(\mathbf{z}|\mathbf{x}, \phi)$. Για το λόγο αυτό η συγκεκριμένη μέθοδος είναι γνωστή ως *Amortized Variational Inference - AVI*. Η ιδανική λύση θα ήταν να πραγματοποιηθεί βελτιστοποίηση των παραμέτρων λ για κάθε δεδομένο του συνόλου εκπαίδευσης ξεχωριστά, αλλά αυτό θα δυσχέραινε τη συμπίεση νέων δεδομένων στα οποία δεν έχει εκπαιδευτεί το δίκτυο. Συνεπώς, η χρήση του κωδικοποιητή αυξάνει τα σφάλμα εκτίμησης των παραμέτρων λ , αλλά καθίσταται αναγκαία για την εξαγωγή χαρακτηριστικών από δεδομένα που δε βρίσκονται στο σύνολο εκπαίδευσης.

3.3 Εκπαίδευση Μοντέλων VAE

Όπως παρουσιάστηκε στην υποενότητα 3.2.2, η συνάρτηση κόστους για την εκμάθηση του *variational* αυτοκωδικοποιητή είναι η ποσότητα $ELBO(\phi, \theta)$ της Εξ. (3.18). Η ποσότητα αυτή αποτελείται από τις παραμέτρους θ του αποκωδικοποιητή (*decoder*) και τις παραμέτρους ϕ του κωδικοποιητή (*encoder*) για τις οποίες θα πρέπει να πραγματοποιηθεί εκπαίδευση των 2 δικτύων.

Για την εκμάθηση των συγκεκριμένων παραμέτρων μπορεί να χρησιμοποιηθεί ένας από τους αλγόριθμους *Gradient Descent*, *Adaptive Moment Estimation (Adam)* κ.α., όπως αυτοί αναλύονται διεξοδικά στο [34]. Σε όλους τους αλγόριθμους αυτούς απαιτείται η εκτίμηση της παραγώγου της ποσότητας $ELBO$ ως προς τις εξεταζόμενες παραμέτρους. Στη συγκεκριμένη ενότητα λοιπόν παρουσιάζεται ο τρόπος εκτίμησης των ζητούμενων παραγώγων και επισημαίνονται τα προβλήματα που προκύπτουν κατά τη διαδικασία αυτή.

3.3.1 Διανυσματική Παράγωγος ως προς θ

Για την εκτίμηση των παραμέτρων θ του αποκωδικοποιητή απαιτείται η εκτίμηση της διανυσματικής παραγώγου της ποσότητας $L(\phi, \theta) = ELBO(\phi, \theta)$ ως προς θ , δηλαδή η ποσότητα $\nabla_\theta L(\phi, \theta)$. Χρησιμοποιώντας λοιπόν την Εξ. (3.18) προκύπτει:

$$\begin{aligned} \nabla_\theta L(\phi, \theta) &= \nabla_\theta E_q[\log p_\theta(\mathbf{X}|\mathbf{Z})] - \nabla_\theta D_{KL}(q_\lambda(\mathbf{Z}|\mathbf{X}, \phi) || p(\mathbf{Z})) \longrightarrow 0 \\ \nabla_\theta L(\phi, \theta) &= \nabla_\theta \int_{\mathbf{Z}} q_\lambda(\mathbf{Z}|\mathbf{X}, \phi) \log p_\theta(\mathbf{X}|\mathbf{Z}) d\mathbf{Z} \xrightarrow{\mathbf{x}_i: i.i.d.} \\ \nabla_\theta L(\phi, \theta) &= \sum_{i=1}^N \int_{\mathbf{z}} q_\lambda(\mathbf{z}_i|\mathbf{x}_i, \phi) \nabla_\theta \log p_\theta(\mathbf{x}_i|\mathbf{z}_i) d\mathbf{z}_i \end{aligned} \quad (3.19)$$

Σύμφωνα με την Εξ. (3.19) απαιτείται αρχικά η εκτίμηση της ποσότητας $\nabla_\theta \log p_\theta(\mathbf{x}_i|\mathbf{z}_i)$. Εφόσον όμως η δεσμευμένη πιθανοφάνεια $p_\theta(\mathbf{x}_i|\mathbf{z}_i)$ αποτελεί την έξοδο του αποκωδικοποιητή, η παράγωγος της κατανομής αυτής ως προς θ μπορεί να εκτιμηθεί με τον αλγόριθμο οπισθοδιάδοσης του λάθους. Επιπλέον, για μείωση του υπολογιστικού κόστους και την αποφυγή εκτίμησης ολοκληρωμάτων πραγματοποιούνται 2 στοχαστικές μέθοδοι οι οποίες παρουσιάζονται συνοπτικά ακολούθως.

Μέθοδος Mini-Batching

Η Εξ. (3.19) προϋποθέτει την εκτίμηση N διαφορετικών ολοκληρωμάτων για την εκτίμηση της ποσότητας $\nabla_\theta L(\lambda, \theta)$. Εναλλακτικά, εφόσον ο συγκεκριμένος εκτιμητής είναι αμερόληπτος (*unbiased estimate*) ως προς την πραγματική διανυσματική παράγωγο της ποσότητας $ELBO$, θα μπορούσε να εκτιμηθεί μόνο ένα ολοκλήρωμα συγκεκριμένου ζεύγους δεδομένων $\mathbf{x}_i, \mathbf{z}_i$ και το

αποτέλεσμα που προκύπτει να πολλαπλασιαστεί N φορές. Με αυτό τον τρόπο, παρόλο που ο υπολογισμός δεν είναι ακριβής για το σύνολο των N δεδομένων, η συγκεκριμένη προσέγγιση δεν οδηγεί σε αρκετά αποκλίνουσες τιμές μειώνοντας παράλληλα αισθητά το υπολογιστικό κόστος. Συνεπώς, επιλέγοντας ένα $i \sim \{1, \dots, N\}$, η ποσότητα $\nabla_{\theta} L(\lambda, \theta)$ γράφεται:

$$\nabla_{\theta} L(\phi, \theta) = N \int_{\mathbf{z}} q_{\lambda}(\mathbf{z}_i | \mathbf{x}_i, \phi) \nabla_{\theta} \log p_{\theta}(\mathbf{x}_i | \mathbf{z}_i) d\mathbf{z}_i \quad (3.20)$$

Μέθοδος Monte-Carlo

Η συγκεκριμένη μέθοδος μπορεί να χρησιμοποιηθεί για τη μετατροπή ενός ολοκληρώματος σε άθροισμα, χρησιμοποιώντας δείγματα από την κατανομή του ολοκληρώματος. Χρησιμοποιώντας λοιπόν τη μέθοδο Monte-Carlo η Εξ. (3.20) γράφεται:

$$\nabla_{\theta} L(\phi, \theta) = N \nabla_{\theta} \log p_{\theta}(\mathbf{x}_i | \mathbf{z}_i^*), \text{ όπου } \mathbf{z}_i^* \sim q_{\lambda}(\mathbf{z}_i | \mathbf{x}_i, \phi) \quad (3.21)$$

Τέλος, να σημειωθεί ότι ο συγκεκριμένος τρόπος εκτίμησης της ποσότητας $\nabla_{\theta} L(\phi, \theta)$ θεωρείται διπλά στοχαστικός, καθώς βασίζεται σε 2 στοχαστικές μεθόδους διευκόλυνσης της διαδικασίας εκτίμησης.

3.3.2 Διανυσματική Παράγωγος ως προς ϕ

Για την εκτίμηση των παραμέτρων ϕ στο τμήμα του κωδικοποιητή απαιτείται η εκτίμηση της διανυσματικής παραγώγου της ποσότητας $L(\phi, \theta) = ELBO(\phi, \theta)$ ως προς ϕ , δηλαδή η ποσότητα $\nabla_{\phi} L(\phi, \theta)$. Χρησιμοποιώντας λοιπόν την Εξ. (3.17) προκύπτει:

$$\nabla_{\phi} L(\phi, \theta) = \nabla_{\phi} E_q[\log p_{\theta}(\mathbf{X} | \mathbf{Z})] - \nabla_{\phi} D_{KL}(q_{\lambda}(\mathbf{Z} | \mathbf{X}, \phi) || p(\mathbf{Z})) \longrightarrow$$

$$\nabla_{\phi} L(\phi, \theta) = \nabla_{\phi} \int_{\mathbf{Z}} q_{\lambda}(\mathbf{Z} | \mathbf{X}, \phi) \log p_{\theta}(\mathbf{X} | \mathbf{Z}) d\mathbf{Z} - \nabla_{\phi} D_{KL}(q_{\lambda}(\mathbf{Z} | \mathbf{X}, \phi) || p(\mathbf{Z})) \xrightarrow{\text{mini-batching}}$$

$$\nabla_{\phi} L(\phi, \theta) = N \nabla_{\phi} \int_{\mathbf{z}} q_{\lambda}(\mathbf{z}_i | \mathbf{x}_i, \phi) \log p_{\theta}(\mathbf{x}_i | \mathbf{z}_i) d\mathbf{z}_i - \nabla_{\phi} D_{KL}(q_{\lambda}(\mathbf{z}_i | \mathbf{x}_i, \phi) || p(\mathbf{z}_i)) \quad (3.22)$$

Σύμφωνα με την Εξ. (3.22), η απόσταση Kullback-Leibler μεταξύ των κανονικών κατανομών $q_{\lambda}(\mathbf{Z} | \mathbf{X}, \phi)$ και $p(\mathbf{Z})$ αποτελεί έναν όρο κλειστής μορφής ως προς τις παραμέτρους ϕ . Συνεπώς, η παραγωγή της συγκεκριμένης ποσότητας ως προς ϕ μπορεί να εκτιμηθεί σε κλειστή μορφή. Αντιθέτως, το κύριο πρόβλημα που τίθεται στην Εξ. (3.22) είναι ο πρώτος όρος, όπου η διανυσματική παράγωγος ∇_{ϕ} μπορεί να εισέλθει εντός του ολοκληρώματος αλλά όχι απευθείας στη συνάρτηση $\log p_{\theta}(\mathbf{x}_i | \mathbf{z}_i)$. Αυτό συμβαίνει διότι η ίδια η συνάρτηση κατανομής του ολοκληρώματος $q_{\lambda}(\mathbf{z}_i | \mathbf{x}_i, \phi)$ εξαρτάται από τις παραμέτρους ϕ του κωδικοποιητή και συνεπώς δεν μπορεί να χρησιμοποιηθεί απευθείας η μέθοδος Monte-Carlo. Σύμφωνα λοιπόν με τους συγγραφείς στο [49] η λύση που προτάθηκε είναι η εναλλαγή μεταβλητής, η οποία ονομάζεται reparametrization trick και εξηγείται αναλυτικά ακολούθως.

Reparametrization Trick

Έστω λοιπόν ότι ζητείται να εκτιμηθεί η παραγωγή ενός ολοκληρώματος I ως προς μία μεταβλητή ϕ , η οποία όμως περιλαμβάνεται στην κατανομή του ολοκληρώματος, όπως δηλαδή στην Εξ. (3.23):

$$I = \frac{\partial}{\partial \phi} \int_{\mathbf{z}} p(\mathbf{z} | \phi) f(\phi, \mathbf{z}) d\mathbf{z} \quad (3.23)$$

Στην παραπάνω εξίσωση η παράγωγος ως προς ϕ μπορεί να εισέλθει εντός του ολοκληρώματος, αλλά δεν μπορεί να εφαρμοσθεί στη συνάρτηση f (εφόσον η κατανομή p περιλαμβάνει τη συγκεκριμένη παράμετρο) και συνεπώς δεν μπορεί να πραγματοποιηθεί Monte-Carlo εκτίμηση του ολοκληρώματος. Αν όμως βρεθεί μία συνάρτηση g τέτοια ώστε η μεταβλητή z να μπορεί να αντικατασταθεί από τη συνάρτηση $z = g(\phi, x, \epsilon)$, όπου ϵ μία τυχαία μεταβλητή, τότε η Εξ. (3.23) θα γραφόταν:

$$I = \frac{\partial}{\partial \phi} \int_{\epsilon} r(\epsilon) f(\phi, g(\phi, x, \epsilon)) d\epsilon \quad (3.24)$$

Συνεπώς, με χρήση αρχικά του Monte-Carlo εκτιμητή η τιμή του παραπάνω ολοκληρώματος προκύπτει:

$$I = \frac{\partial}{\partial \phi} \int_{\epsilon} r(\epsilon) f(\phi, g(\phi, x, \epsilon)) d\epsilon \xrightarrow{\text{Monte-Carlo}} \\ I = \frac{\partial}{\partial \phi} f(\phi, g(\phi, x, \epsilon^*)), \quad \text{όπου } \epsilon^* \sim r(\epsilon) \quad (3.25)$$

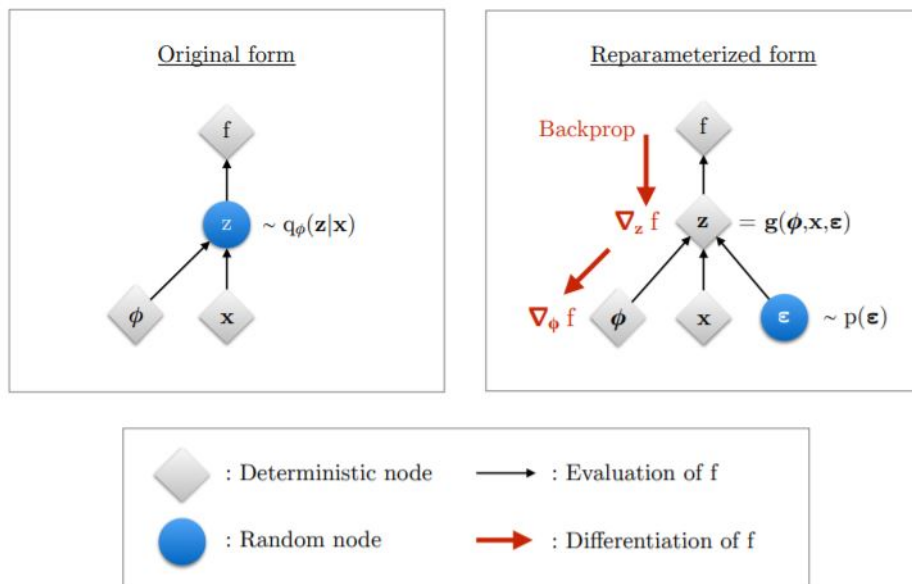
Η επιλογή της συνάρτησης g είναι σημαντική για την πραγματοποίηση της προαναφερθείσας μεθόδου. Βέβαια, στη συγκεκριμένη περίπτωση όπου θεωρούμε ότι η μεταβλητή z εξαρτάται από τις παραμέτρους ϕ του κωδικοποιητή δια μέσου των παραμέτρων λ της κατανομής $q_{\lambda}(\mathbf{Z}|\mathbf{X}, \phi)$, είναι προφανές ότι ισχύει: $z = g(\phi, x, \epsilon) = \mu_{\phi}(x) + \sigma_{\phi}(x)\epsilon$. Συνεπώς, η συνάρτηση g αποτελεί το γραμμικό μετασχηματισμό της μεταβλητής $\epsilon \sim N(0, 1)$ ως προς τη μεταβλητή $z \sim N(\mu_{\phi}(x), \sigma_{\phi}(x))$.

Αντικαθιστώντας λοιπόν τη συνάρτηση $f(\phi, z) = \log p_{\theta}(\mathbf{x}_i|\mathbf{z}_i)$ από την Εξ. (3.22), τότε αυτή στη συνέχεια γράφεται:

$$\nabla_{\phi} L(\phi, \theta) = N \nabla_{\phi} \int_{\mathbf{z}} q_{\lambda}(\mathbf{z}_i|\mathbf{x}_i, \phi) \log p_{\theta}(\mathbf{x}_i|\mathbf{z}_i) d\mathbf{z}_i - \nabla_{\phi} D_{KL}(q_{\lambda}(\mathbf{z}_i|\mathbf{x}_i, \phi)||p(\mathbf{z}_i)) \xrightarrow{\text{reparametrization}}$$

$$\nabla_{\phi} L(\phi, \theta) = N \nabla_{\phi} \log p_{\theta}(\mathbf{x}_i|g(\phi, \mathbf{x}_i, \boldsymbol{\epsilon}^*)) - \nabla_{\phi} D_{KL}(q_{\lambda}(\mathbf{z}_i|\mathbf{x}_i, \phi)||p(\mathbf{z}_i)), \quad \text{όπου } \boldsymbol{\epsilon}^* \sim N(\mathbf{0}, \mathbf{I}) \quad (3.26)$$

Η σχηματική αναπαράσταση της τεχνικής reparametrization trick πραγματοποιείται στο Σχ. 3.4, όπου απεικονίζεται παραστατικά η χρησιμότητα της τεχνικής εναλλαγής μεταβλητής κατά την παραγωγή ως προς τις παραμέτρους ϕ του κωδικοποιητή:



Σχήμα 3.4: Reparametrization trick κατά την εκτίμηση των παραμέτρων ϕ του νευρωνικού δικτύου VAE. Πηγή [50].

Σημείωση: Η τεχνική reparametrization δεν μπορεί να εφαρμοσθεί σε όλα τα είδη των κατανομών. Ενδεικτικές κατανομές στις οποίες εφαρμόζεται είναι η Gaussian και η Gamma, ενώ αντίθετως δεν μπορεί να εφαρμοσθεί στις διακριτές κατανομές.

Συγκεντρωτικά λοιπόν, ο αλγόριθμος εκπαίδευσης του νευρωνικού δικτύου VAE συνοψίζεται ακολούθως:

- Έστω N δεδομένα εισόδου $\mathbf{x} \in \mathbb{R}^D$ και τα αντίστοιχα συμπιεσμένα δεδομένα $\mathbf{z} \in \mathbb{R}^d$
- Επιλογή ενός $i \sim \{1, \dots, N\}$ και εκτίμηση των στοχαστικών διανυσματικών παραγώγων της ποσότητας ELBO ως προς θ και ϕ , όπως αναγράφεται ακολούθως:

– Παράγωγος ως προς θ :

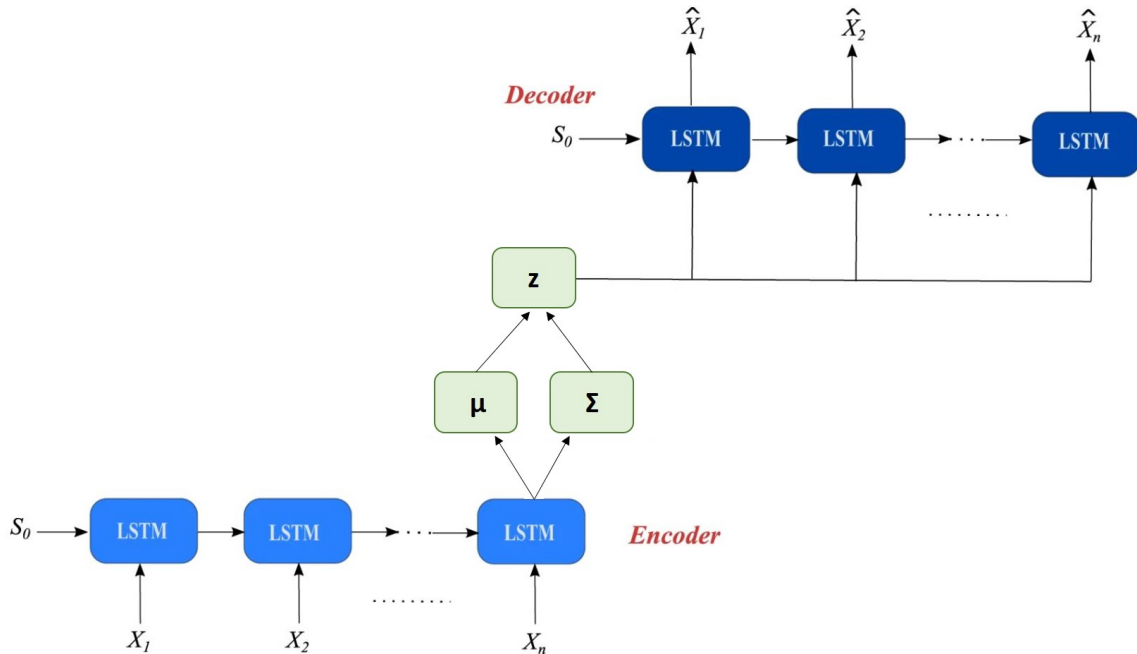
$$\nabla_{\theta} L(\phi, \theta) = N \nabla_{\theta} \log p_{\theta}(\mathbf{x}_i | \mathbf{z}_i^*), \text{ όπου } \mathbf{z}_i^* \sim q_{\lambda}(\mathbf{z}_i | \mathbf{x}_i, \phi)$$

– Παράγωγος ως προς ϕ :

$$\nabla_{\phi} L(\phi, \theta) = N \nabla_{\phi} \log p_{\theta}(\mathbf{x}_i | g(\phi, \mathbf{x}_i, \boldsymbol{\epsilon}^*)) - \nabla_{\phi} D_{KL}(q_{\lambda}(\mathbf{z}_i | \mathbf{x}_i, \phi) || p(\mathbf{z}_i)), \text{ όπου } \boldsymbol{\epsilon}^* \sim N(\mathbf{0}, \mathbf{I})$$

- Ανανέωση των παραμέτρων θ και ϕ επιλέγοντας μία από τις gradient-based optimization μεθόδους που παρουσιάζονται από τους συγγραφείς στο [34].

Να σημειωθεί ότι στην παρούσα εργασία χρησιμοποιείται το συγκεκριμένο νευρωνικό δίκτυο VAE που μόλις παρουσιάστηκε, με τη μόνη διαφορά να έγκειται στο γεγονός ότι ως κωδικοποιητής και αποκωδικοποιητής χρησιμοποιείται το νευρωνικό δίκτυο LSTM. Η συγκεκριμένη αρχιτεκτονική δε δημιουργεί κάποιο πρόβλημα στην προαναφερθείσα εκπαίδευση του δικτύου. Η μόνη διαφορά είναι ότι απαιτείται η κατάλληλη οπισθοδιάδοση (propagation) των στοχαστικών διανυσματικών παραγώγων εντός των νευρωνικών δικτύων LSTM. Η συγκεκριμένη αρχιτεκτονική απεικονίζεται στο Σχ. 3.5 και θα αναφέρεται από εδώ και στο εξής ως Variational LSTM AutoEncoder, ή συντομογραφικά ως V-LSTM-AE.



Σχήμα 3.5: Αρχιτεκτονική variational αυτοκωδικοποιητή σε συνδυασμό με τα νευρωνικά δίκτυα LSTM. Στο τμήμα του κωδικοποιητή και στον αποκωδικοποιητή χρησιμοποιείται η ξεδιπλωμένη μορφή του μη κυκλικού υπολογιστικού γράφου.

Κεφάλαιο 4

Προηγμένη Variational Συμπερασματολογία

Στο Κεφάλαιο αυτό παρουσιάζονται ορισμένες μέθοδοι που μπορούν να συνδυαστούν με τα νευρωνικά δίκτυα VAE, με στόχο την αποτελεσματικότερη συμπίεση και εξαγωγή χαρακτηριστικών από δεδομένα χρονοσειρών. Συγκεκριμένα, στην Εν. 4.1 παρουσιάζεται η semi-amortized συμπερασματολογία, η οποία μπορεί να μειώσει το amortization σφάλμα που προκαλείται από το δίκτυο του κωδικοποιητή βασιζόμενη στη στοχαστική variational συμπερασματολογία (stochastic variational inference). Επιπλέον, στην Εν. 4.2 παρουσιάζεται η μεθοδολογία των νευρωνικών δικτύων VAE σε συνδυασμό με τη Λαπλασιανή προσέγγιση, η οποία αφενός μεν μειώνει περαιτέρω το amortization σφάλμα και αφετέρου εκτιμά έναν ολοκληρωμένο πίνακα συνδιασπορών στο μεσσαίο επίπεδο, εμπλουτίζοντας έτσι τις συσχετίσεις των συμπιεσμένων δεδομένων.

4.1 Semi-Amortized Συμπερασματολογία

Σύμφωνα με την Εν. 3.2, οι variational αυτοκωδικοποιητές αποτελούνται από 2 δίκτυα βάσει των οποίων επιτυγχάνεται η συμπίεση των δεδομένων εισόδου: (1) το δίκτυο του κωδικοποιητή (inference network) με χρήση του οποίου τα δεδομένα εισόδου \mathbf{x} συμπιέζονται στην αναπαράσταση \mathbf{z} και (2) το δίκτυο του αποκωδικοποιητή (generative network) με βάση το οποίο τα συμπιεσμένα δεδομένα οδηγούνται στην ανακατασκευή των δεδομένων εισόδου. Οι συγκεκριμένες αρχιτεκτονικές ονομάζονται amortized variational αυτοκωδικοποιητές, καθώς η χρήση ενός νευρωνικού δικτύου ως κωδικοποιητή προσθέτει ένα επιπλέον σφάλμα μοντελοποίησης που προκύπτει από το γεγονός ότι διαφορετικά δεδομένα εκπαίδευσης "μοιράζονται" (amortize) τις ίδιες παραμέτρους ϕ του νευρωνικού, ώστε να καταλήξουν στις παραμέτρους λ της κατανομής $q_\lambda(\mathbf{z}|\mathbf{x}, \phi)$ των συμπιεσμένων δεδομένων τους.

Στην παρούσα ενότητα λοιπόν παρουσιάζεται ένας τρόπος μείωσης του amortization σφάλματος χρησιμοποιώντας συνδυαστικά τη στοχαστική variational συμπερασματολογία (SVI) [48] μαζί με την amortized variational συμπερασματολογία (AVI) [49]. Συγκεκριμένα, προτείνεται μία υβριδική προσέγγιση βασισμένη στη δημοσίευση [51], όπου η AVI χρησιμοποιείται για την αρχικοποίηση των variational παραμέτρων λ της κατανομής $q_\lambda(\mathbf{z}|\mathbf{x}, \phi)$ και στη συνέχεια χρησιμοποιείται η SVI με στόχο την κατάλληλη αναπροσαρμογή των παραμέτρων αυτών για βελτίωση της απόδοσης του μοντέλου. Πριν όμως την παρουσίαση του υβριδικού αυτού μοντέλου αρχικά πραγματοποιείται μία σύντομη αναφορά στη μέθοδο SVI στην υποενότητα που ακολουθεί.

4.1.1 Στοχαστική Variational Συμπερασματολογία

Η στοχαστική variational συμπερασματολογία (SVI) προτάθηκε από τους συγγραφείς στο [48] και επέλυσε το πρόβλημα της εφαρμογής της variational συμπερασματολογίας σε πολύ μεγάλο όγκο δεδομένων χρησιμοποιώντας όμως μόνο το δίκτυο του αποκωδικοποιητή. Στις περιπτώσεις αυτές, για την εκπαίδευση των μοντέλων απαιτείται η χρήση υποσυνόλων του συνολικού όγκου των δεδομένων (δηλ. batches) και συνεπώς, οι συγγραφείς της συγκεκριμένης μεθοδολογίας πρότειναν να εφαρμοσθεί η μέθοδος της ανόδου κλίσης (gradient ascent) στην ποσότητα $L(\lambda, \theta) = ELBO(\lambda, \theta)$ της Εξ. (3.17), όπως παρουσιάζεται ακολούθως:

1. Έστω $\mathbf{x} \in \mathbb{R}^D$ ένα από τα N δεδομένα του συνόλου δεδομένων εισόδου \mathbf{X} .
2. Αρχικοποίηση των παραμέτρων λ της κατανομής $q_\lambda(\mathbf{z}|\mathbf{x})$
3. Για $k = 0, \dots, K - 1$ εφαρμογή gradient ascent:

$$\lambda_{k+1} = \lambda_k + \alpha \nabla_\lambda L(\lambda_k, \theta, \mathbf{x})$$

4. Ανανέωση των παραμέτρων θ του αποκωδικοποιητή χρησιμοποιώντας την στοχαστική διανυσματική παράγωγο $\nabla_\theta L(\lambda, \theta)$ της Εξ. (3.21).

Στην παραπάνω μεθοδολογία ο αριθμός K ισούται με τον πλήθος των επαναλήψεων της μεθόδου SVI και α είναι ο συντελεστής εκμάθησης. Ουσιαστικά λοιπόν η συγκεκριμένη μεθοδολογία βελτιστοποιεί την ποσότητα $ELBO(\lambda, \theta)$ για κάθε ένα δεδομένο εκπαίδευσης ξεχωριστά, εκτιμώντας τις βέλτιστες δυνατές παραμέτρους λ της κατανομής $q_\lambda(\mathbf{z}|\mathbf{x})$ στο μεσαίο επίπεδο. Επιπλέον, σύμφωνα με τα παραπάνω βήματα παρατηρείται ότι οι παράμετροι λ βελτιστοποιούνται ξεχωριστά από τις παραμέτρους θ , γεγονός που αποτρέπει τις παραμέτρους του αποκωδικοποιητή θ να εγκλωβιστούν σε τοπικά μέγιστα.

Τα μειονεκτήματα της συγκεκριμένης μεθόδου αναφέρονται περιληπτικά ακολούθως:

1. Σε κάθε μία εποχή και για κάθε ένα δεδομένο εισόδου \mathbf{x} απαιτείται η επαναληπτική μέθοδος gradient ascent για την εκτίμηση των παραμέτρων λ της κατανομής $q_\lambda(\mathbf{z}|\mathbf{x})$ στο μεσαίο επίπεδο. Αν λοιπόν ο αριθμός των επαναλήψεων K που απαιτείται είναι αρκετά μεγάλος, τότε η συγκεκριμένη επαναληπτική διαδικασία καθυστερεί αρκετά την εκπαίδευση του μοντέλου.
2. Όπως αναφέρθηκε και στην Εν. 3.2, η συγκεκριμένη μεθοδολογία δεν μπορεί να εφαρμοσθεί σε δεδομένα τα οποία δεν έχουν χρησιμοποιηθεί κατά την εκπαίδευση του δικτύου. Αυτό συμβαίνει διότι οι παράμετροι λ έχουν εκτιμηθεί μόνο για κάθε ένα από τα δεδομένα του συνόλου εκπαίδευσης και συνεπώς οι αντίστοιχες παράμετροι είναι άγνωστες για νέα δεδομένα. Κατά συνέπεια, η SVI δεν μπορεί να χρησιμοποιηθεί αυτούσια σε νέα δεδομένα για την εξαγωγή χαρακτηριστικών.

Το ερώτημα λοιπόν που τίθεται είναι εάν είναι εφικτό η προαναφερθείσα μεθοδολογία να εφαρμοσθεί παράλληλα με την amortized variational συμπερασματολογία, ώστε αφενός μεν να ελεγχθεί εάν η συνδυαστική μέθοδος βελτιώνει τη μοντελοποίηση και αφετέρου να δοθεί η δυνατότητα συμπίεσης νέων δεδομένων με χρήση του κωδικοποιητή. Η υβριδική αυτή μεθοδολογία παρουσιάστηκε στο [51] και περιγράφεται συνοπτικά ακολούθως.

4.1.2 Υβριδική Μέθοδος AVI-SVI

Ακολούθως παρουσιάζεται μία μεθοδολογία ταυτόχρονης χρήσης της amortized με την stochastic variational συμπερασματολογία. Η συγκεκριμένη μέθοδος ουσιαστικά χρησιμοποιεί ένα δίκτυο κωδικοποιητή στα δεδομένα εισόδου \mathbf{x} και καταλήγει αρχικά στις παραμέτρους λ της κατανομής $q_\lambda(\mathbf{z}|\mathbf{x}, \phi)$. Στη συνέχεια, χρησιμοποιώντας το βήμα 3 της μεθόδου SVI αναπροσαρμόζει τις παραμέτρους αυτές με στόχο τη βελτίωση της συμπιεσμένης αναπαράστασης \mathbf{z} . Συνεπώς, η διαφορά της μεθόδου αυτής από την απλή SVI έγκειται στο γεγονός ότι η πρόβλεψη των παραμέτρων λ πραγματοποιείται από το δίκτυο του κωδικοποιητή και στη συνέχεια εφαρμόζεται η μέθοδος gradient ascent για τη βελτίωση των συγκεκριμένων παραμέτρων.

Η προαναφερθείσα μεθοδολογία θα αναφέρεται από εδώ και στο εξής ως semi-amortized variational συμπερασματολογία, καθώς με χρήση της μεθόδου SVI μειώνεται το amortization σφάλμα που προκύπτει από τη μέθοδο AVI. Επιπλέον, ο αντίστοιχος αυτοκωδικοποιητής ορίζεται ως SA-VAE (Semi-Amortized VAE) και για την εκπαίδευσή του ακολουθούνται τα βήματα που ακολουθούν:

1. Έστω $\mathbf{x} \in \mathbb{R}^D$ ένα από τα N δεδομένα του συνόλου δεδομένων εισόδου \mathbf{X} .
2. Αρχικοποίηση των παραμέτρων λ της κατανομής $q_\lambda(\mathbf{z}|\mathbf{x}, \phi)$ από τον κωδικοποιητή. Δηλαδή $\lambda_0 = \text{enc}_\phi(\mathbf{x})$
3. Για $k = 0, \dots, K - 1$ εφαρμογή gradient ascent:

$$\lambda_{k+1} = \lambda_k + \alpha \nabla_\lambda L(\lambda_k, \theta, \mathbf{x})$$

4. Ανανέωση των παραμέτρων θ του αποκωδικοποιητή βάσει της παραγώγου $\frac{dL(\lambda, \theta)}{d\theta}$.
5. Ανανέωση των παραμέτρων ϕ του κωδικοποιητή βάσει της παραγώγου $\frac{dL(\lambda, \theta)}{d\phi}$.

Σύμφωνα με τα βήματα 4 και 5 της υβριδικής μεθόδου, για την εκπαίδευση του αυτοκωδικοποιητή SA-VAE απαιτείται η εκτίμηση της παραγώγου της ποσότητας ELBO ως προς τις παραμέτρους θ και ϕ . Σε αντίθεση όμως με την amortized variational συμπερασματολογία της Εν. 3.3, κατά την εκπαίδευση του υβριδικού δικτύου απαιτείται η οπισθοδιάδοση της ολικής παραγώγου μέσα από την

επαναληπτική διαδικασία του βήματος 3 που μόλις προαναφέρθηκε. Με άλλα λόγια, απαιτείται η οπισθοδιάδοση της παραγώγου μέσα από την επαναληπτική διαδικασία της μεθόδου gradient ascent, όπως δηλαδή παρουσιάζεται στο [52].

Εστω λοιπόν ότι εκτελούμε τη μέθοδο gradient ascent του βήματος 3 με μία μόνο επανάληψη, δηλαδή ισχύει: $\lambda_1 = \lambda_0 + \alpha \nabla_{\lambda} L(\lambda_0, \theta, \mathbf{x})$. Αν στη συνέχεια επιθυμούμε να εκτιμήσουμε την παράγωγο $\frac{dL}{d\phi}$, τότε η συγκεκριμένη ποσότητα γράφεται:

$$\frac{dL}{d\phi} = \frac{d\lambda_0}{d\phi} \frac{dL}{d\lambda_0} \quad (4.1)$$

όπου ο πρώτος όρος του παραπάνω γινομένου μπορεί να εκτιμηθεί με χρήση του αλγορίθμου οπισθοδιάδοσης του λάθους στο τμήμα του κωδικοποιητή. Αντιθέτως, ο δεύτερος όρος δεν μπορεί να εκτιμηθεί άμεσα, καθώς για την εκτίμηση της ποσότητας ELBO έχει προηγηθεί η μέθοδος gradient ascent εκτιμώντας την ποσότητα λ_1 . Για την εκτίμηση λοιπόν του δεύτερου όρου της Εξ. (4.1) χρησιμοποιούνται τα fast Hessian-vector products [53, 54], όπως παρουσιάζονται ακολούθως:

$$\frac{dL}{d\lambda_0} = \frac{d\lambda_1}{d\lambda_0} \frac{dL}{d\lambda_1} = (\mathbf{I} + \alpha H_{\lambda, \lambda} L(\lambda_0, \theta, \mathbf{x})) \frac{dL}{d\lambda_1} \quad (4.2)$$

όπου η ποσότητα $H_{\lambda, \lambda}$ αποτελεί τον Hessian πίνακα της ποσότητας ELBO ως προς την παράμετρο λ_0 και α είναι ο συντελεστής εκμάθησης της μεθόδου gradient ascent. Επιπλέον, όσον αφορά το δεύτερο όρο του γινομένου της Εξ. (4.2), η παράγωγος της ποσότητας ELBO ως προς την παράμετρο λ_1 εκτιμάται σύμφωνα με την τεχνική reparametrization trick της υποενότητας 3.3.2. Η διαδικασία αυτή είναι παρόμοια και για περισσότερες επαναλήψεις της μεθόδου gradient ascent στο Βήμα 3 της υβριδικής μεθόδου. Τέλος, με αντίστοιχο τρόπο μπορούν να εκτιμηθεί και η παράγωγος της ποσότητας ELBO ως προς τις παραμέτρους θ .

4.2 Λαπλασιανή Προσέγγιση

Στη συγκεκριμένη ενότητα παρουσιάζεται ο τρόπος με τον οποίο μπορούν να συνδυαστούν τα νευρωνικά δίκτυα VAE με τη Λαπλασιανή προσέγγιση, όπως δηλαδή παρουσιάστηκε από τους συγγραφείς στο [55]. Πριν όμως την παρουσίαση της Λαπλασιανής μεθόδου, θα πρέπει να γίνει μία σύντομη αναφορά στη νευρωνικά δίκτυα εμπρόσθιας τροφοδότησης MLP τα οποία χρησιμοποιούν ως συνάρτηση ενεργοποίησης τη ReLU (Rectified Linear Unit). Τα δίκτυα συνήθως ονομάζονται Piece-wise Linear και αναλύονται περιληπτικά ακολούθως.

4.2.1 Δίκτυα Piece-wise Linear

Εστω ένα νευρωνικό δίκτυο εμπρόσθιας τροφοδότησης MLP με L επίπεδα, όπου ως συνάρτηση ενεργοποίησης χρησιμοποιείται η ReLU. Θεωρώντας ως \mathbf{h} τις τιμές των κρυφών επιπέδων και ως \mathbf{x} την έξοδο του νευρωνικού δικτύου, τότε εξάγονται οι Εξ. (4.3) και (4.4):

$$\mathbf{h}_{l+1} = \text{ReLU}(\mathbf{W}_l \mathbf{h}_l + \mathbf{b}_l), \quad l = 0, \dots, L-1 \quad (4.3)$$

$$\mathbf{x} = \mathbf{W}_L \mathbf{h}_L + \mathbf{b}_L \quad (4.4)$$

Τα δίκτυα αυτά ονομάζονται piece-wise linear, διότι χωρίζουν την είσοδο σε περιοχές εντός των οποίων το δίκτυο τοπικά συμπεριφέρεται γραμμικά [56]. Η εφαρμογή της ReLU ως συνάρτηση ενεργοποίησης είναι ουσιαστικά ισοδύναμη με τον πολλαπλασιασμό μίας μάσκας \mathbf{O} , η οποία ισοδυναμεί με έναν διαγώνιο πίνακα όπου τα διαγώνια στοιχεία του σ_i ισοδυναμούν με:

$$o_i = \begin{cases} 1, & \text{αν } \mathbf{w}_i^T \mathbf{h} + b_i > 0 \\ 0, & \text{αλλιώς} \end{cases}$$

Με χρήση της συγκεκριμένης μάσκας \mathbf{O} και πραγματοποιώντας αλυσιδωτά τον πολλαπλασιασμό των βαρών \mathbf{W} με τους διαγώνιους πίνακες \mathbf{O} σε κάθε ένα επίπεδο του δικτύου, τα δεδομένα εξόδου \mathbf{x} της Εξ. (4.4) ουσιαστικά γράφονται στη μορφή:

$$\begin{aligned} \mathbf{x} &= \mathbf{W}_L \text{ReLU}(\mathbf{W}_{L-1} \mathbf{h}_{L-1} + \mathbf{b}_{L-1}) + \mathbf{b}_L \rightarrow \\ \mathbf{x} &= \mathbf{W}_L \mathbf{O}_{L-1} (\mathbf{W}_{L-1} \mathbf{h}_{L-1} + \mathbf{b}_{L-1}) + \mathbf{b}_L \rightarrow \\ \mathbf{x} &= \mathbf{W}_L \mathbf{O}_{L-1} \mathbf{W}_{L-1} \dots \mathbf{O}_0 \mathbf{W}_0 \mathbf{z} + \dots \rightarrow \\ \mathbf{x} &= g_\theta(\mathbf{z}) = \mathbf{W}_z \mathbf{z} + \mathbf{b}_z \end{aligned} \quad (4.5)$$

όπου η συνάρτηση g_θ αποτελεί το piece-wise linear δίκτυο και \mathbf{z} τα δεδομένα εισόδου στο δίκτυο στο πρώτο επίπεδο (δηλαδή $\mathbf{z} = \mathbf{h}_0$). Με αυτόν τρόπο όμως ο πίνακας \mathbf{W}_z θα μπορούσε να θεωρηθεί αντίστοιχος με τον πίνακα \mathbf{W} της πιθανοτικής μεθόδου κύριων συνιστωσών της Εξ. (3.3). Έτσι, παρόλο που στο δίκτυο χρησιμοποιήθηκε η μη γραμμική συνάρτηση μετασχηματισμού ReLU των δεδομένων εισόδου \mathbf{z} , η θεώρηση ότι η συνάρτηση αυτή ισοδυναμεί με ένα διαγώνιο πίνακα με τιμές 0 ή 1 σε κάθε ένα επίπεδο του δικτύου οδηγεί φαινομενικά σε ένα γραμμικό τρόπο σύνδεσης των κρυμμένων μεταβλητών \mathbf{z} με τις μεταβλητές εξόδου \mathbf{x} .

Βέβαια, να σημειωθεί ότι ο πίνακας \mathbf{W}_z δεν μπορεί να θεωρηθεί ακριβώς ως ένας πίνακας γραμμικού μετασχηματισμού, για τον οποίο μπορούμε να εκτιμήσουμε μεγέθη όπως π.χ. ο ανάστροφος \mathbf{W}_z^T , ο αντίστροφος \mathbf{W}_z^{-1} κ.α. Κατά συνέπεια δεν μπορούμε να πραγματοποιήσουμε οπίσθια τροφοδότηση των δεδομένων εξόδου \mathbf{x} στο δίκτυο, ώστε να εκτιμήσουμε τα δεδομένα εισόδου \mathbf{z} . Αυτό συμβαίνει διότι ο πίνακας αυτός έχει προκύψει ουσιαστικά βάσει των μάσκων o_i , τα οποία περιλαμβάνουν μηδενικές τιμές και συνεπώς μετασχηματίζουν μη γραμμικά τα δεδομένα σε κάθε ένα επίπεδο του δικτύου.

4.2.2 Εκ των Υστέρων Συμπερασματολογία για ReLU Δίκτυα

Ύστερα από την προαναφερθείσα μεθοδολογία το ερώτημα που προκύπτει όμως είναι το εξής. Αν θεωρήσουμε ως αποκωδικοποιητή ένα δίκτυο MLP με ReLU συναρτήσεις ενεργοποίησης, ο φαινομενικά γραμμικός μετασχηματισμός της Εξ. (4.5) μήπως θα μπορούσε να βοηθήσει στη μείωση του amortization σφάλματος στο τμήμα του κωδικοποιητή; Ο συλλογισμός αυτός προκύπτει από το γεγονός ότι στην πιθανοτική μέθοδο των κύριων συνιστωσών η εκ των υστέρων κατανομή των κρυφών δεδομένων $p(\mathbf{z}|\mathbf{x})$ προέκυπτε αποκλειστικά από το αντίστοιχο τμήμα του αποκωδικοποιητή (αλλά με γραμμικό μετασχηματισμό, δες Σχ. 3.2), σύμφωνα με τις Εξ. (4.6) και (4.7):

$$p(\mathbf{z}|\mathbf{x}) = N\left(\frac{1}{\sigma^2} \mathbf{\Sigma} \mathbf{W}^T (\mathbf{x} - \mathbf{b}), \mathbf{\Sigma}\right) \quad (4.6)$$

$$\text{όπου } \mathbf{\Sigma} = \left(\frac{1}{\sigma^2} \mathbf{W}^T \mathbf{W} + \mathbf{I}\right)^{-1} \quad (4.7)$$

Αν λοιπόν σε κάθε επανάληψη της εκπαίδευσης του νευρωνικού δικτύου VAE, εκτός από την προσεγγιστική κατανομή $q_\lambda(\mathbf{Z}|\mathbf{X}, \phi)$ που εκτιμάται από τον κωδικοποιητή, καταφέρουμε να εκτιμήσουμε και τον πίνακα \mathbf{W}_z από το τμήμα του αποκωδικοποιητή, έπεται ότι μπορούμε στη συνέχεια να εκτιμήσουμε και την πραγματική κατανομή $p(\mathbf{z}|\mathbf{x})$ από τις Εξ. (4.6) και (4.7). Συνεπώς, η αρχική εκτίμηση $q_\lambda(\mathbf{Z}|\mathbf{X}, \phi)$ θα μπορούσε να ανανεωθεί κατάλληλα χρησιμοποιώντας και την κατανομή $p(\mathbf{z}|\mathbf{x})$, όπως παρουσιάζεται στον Αλγ. 1.

Αλγόριθμος 1 Εκ των υστέρων συμπερασματολογία για τα piece-wise linear δίκτυα

Είσοδος: δεδομένα \mathbf{x} , inference δίκτυο enc_φ , piece-wise linear generative δίκτυο g_θ , πλήθος ανανέωσης T , decay a_t .

Εξοδος: full-covariance Γκαουσιανή εκ των υστέρων κατανομή $q(\mathbf{z}|\mathbf{x}, \phi)$.

```

 $\boldsymbol{\mu}_0 \leftarrow enc_\varphi(\mathbf{x})$  ▷ εκτίμηση μέσης τιμής από δίκτυο κωδικοποιητή
for  $t = 0$  to  $T - 1$  do
   $g_\theta(\boldsymbol{\mu}_t) \leftarrow \mathbf{W}_t \boldsymbol{\mu}_t + \mathbf{b}_t$  ▷ εκτίμηση  $\mathbf{W}_t$  και  $\mathbf{b}_t$  από τον αποκωδικοποιητή
   $\boldsymbol{\Sigma}_t \leftarrow (\sigma^{-2} \mathbf{W}_t^T \mathbf{W}_t + \mathbf{I})^{-1}$  ▷ εκτίμηση πίνακα συνδιασπορών  $\boldsymbol{\Sigma}_t$  βάσει της Εξ. (4.7)
   $\boldsymbol{\mu}' \leftarrow \sigma^{-2} \boldsymbol{\Sigma}_t \mathbf{W}_t^T (\mathbf{x} - \mathbf{b}_t)$  ▷ εκτίμηση μέσης τιμής  $\boldsymbol{\mu}'$  βάσει της Εξ. (4.6)
   $\boldsymbol{\mu}_{t+1} \leftarrow (1 - a_t) \boldsymbol{\mu}_t + a_t \boldsymbol{\mu}'$  ▷ γραμμικό interpolation των 2 μέσων τιμών κωδικοποιητή με αυτή της Εξ. (4.6)
end for

 $\boldsymbol{\Sigma}_T \leftarrow (\sigma^{-2} \mathbf{W}_T^T \mathbf{W}_T + \mathbf{I})^{-1}$  ▷ ολοκληρωμένος πίνακας συνδιασπορών κρυφών δεδομένων
 $q(\mathbf{z}|\mathbf{x}, \phi) \leftarrow N(\boldsymbol{\mu}_T, \boldsymbol{\Sigma}_T)$  ▷ εκτίμηση κατανομής  $q(\mathbf{z}|\mathbf{x}, \phi)$  κρυφών δεδομένων
 $p_\theta(\mathbf{x}|\mathbf{z}) \leftarrow \mathbf{W}_T \mathbf{z} + \mathbf{b}_T$  ▷ όπου  $\mathbf{z}$  δείγμα από την κατανομή  $q(\mathbf{z}|\mathbf{x}, \phi)$ 

```

Σύμφωνα λοιπόν με τον παραπάνω αλγόριθμο, κατά την εκπαίδευση του νευρωνικού δικτύου η πορεία που ακολουθεί ένα τυχαίο σήμα \mathbf{x} εντός του δικτύου είναι η εξής:

1. Αρχικά, το σήμα $\mathbf{x} \in \mathbb{R}^D$ εισέρχεται στον κωδικοποιητή με παραμέτρους ϕ και καταλήγει ενδεικτικά σε μία Γκαουσιανή κατανομή $q_\lambda(\mathbf{z}|\mathbf{x}, \phi)$, οι παράμετροι λ της οποίας αποτελούνται από το d -διάστατο διάνυσμα $\boldsymbol{\mu}$ και τα d σε πλήθος διαγώνια στοιχεία του πίνακα συνδιασπορών $\boldsymbol{\Sigma}$ διαστάσεων $d \times d$.
2. Ύστερα από την εκτίμηση της κατανομής $q_\lambda(\mathbf{z}|\mathbf{x}, \phi)$, η μέση τιμή $\boldsymbol{\mu}$ της κατανομής αυτής τροφοδοτείται στο τμήμα του αποκωδικοποιητή και εξάγεται ο φαινομενικά γραμμικός μετασχηματισμός \mathbf{W} και το διάνυσμα μεροληψίας \mathbf{b} της Εξ. (4.5).
3. Με χρήση του πίνακα \mathbf{W} εκτιμάται ο πίνακας συνδιασπορών $\boldsymbol{\Sigma}$ της κατανομής $q_\lambda(\mathbf{z}|\mathbf{x}, \phi)$ βάσει της Εξ. (4.7), αντίστοιχα με τη πιθανοτική μέθοδο των κύριων συνιστωσών.
4. Επιπλέον εκτιμάται μία δεύτερη μέση τιμή $\boldsymbol{\mu}'$ της κατανομής $q_\lambda(\mathbf{z}|\mathbf{x}, \phi)$ βάσει της Εξ. (4.6), αντίστοιχα με την πιθανοτική μέθοδο των κύριων συνιστωσών.
5. Πραγματοποιείται γραμμικό interpolation των μέσων τιμών $\boldsymbol{\mu}$ και $\boldsymbol{\mu}'$ και θέτουμε το αποτέλεσμα στη μεταβλητή $\boldsymbol{\mu}$.
6. Επιστροφή στο βήμα 2 και επανεκτέλεση των βημάτων 3, 4 και 5 T φορές, με τη διαφορά όμως ότι στις επόμενες επανλήψεις του βήματος 2 ως μέση τιμή $\boldsymbol{\mu}$ χρησιμοποιείται αυτή του βήματος 5 της προηγούμενης επανάληψης (κι όχι την αρχική τιμή που προέκυψε από το τμήμα του κωδικοποιητή).
7. Ύστερα από την εκτίμηση της τελικής κατανομής $q_\lambda(\mathbf{z}|\mathbf{x}, \phi)$ πραγματοποιείται δειγματοληψία ενός τυχαίου δείγματος, παράγοντας τη συμπιεσμένη αναπαράσταση $\mathbf{z} \in \mathbb{R}^d$. Να σημειωθεί ότι με τη συγκεκριμένη μεθοδολογία ο πίνακας συνδιασπορών $\boldsymbol{\Sigma}$ της $q_\lambda(\mathbf{z}|\mathbf{x}, \phi)$ παύει να είναι διαγώνιος και αποτελεί έναν ολοκληρωμένο πίνακα συνδιασπορών που ερμηνεύει τη συσχέτιση των συμπιεσμένων δεδομένων \mathbf{z} .
8. Το συγκεκριμένο δείγμα οδηγείται στο τμήμα του αποκωδικοποιητή με παραμέτρους θ για την εκτίμηση της δεσμευμένης συνάρτησης πιθανοφάνειας $p_\theta(\mathbf{x}|\mathbf{z})$.

9. Τέλος, εφόσον όλες οι ζητούμενες κατανομές είναι γνωστές, εκτιμάται η ποσότητα ELBO σύμφωνα με την Εξ. (3.18). Στη συνέχεια πραγματοποιείται παραγωγή της ποσότητας αυτής ως προς θ και ϕ για την εκπαίδευση του δικτύου (δες υποενότητες 3.3.1 και 3.3.2 αντίστοιχα).

4.2.3 Variational Λαπλασιανοί Αυτοκωδικοποιητές

Ακολούθως παρουσιάζεται η ιδέα της Λαπλασιανής προσέγγιση (Laplace Approximation) σε συνδυασμό με τους VAE αυτοκωδικοποιητές, επεκτείνοντας την ιδέα του VAE στον Variational Laplace AutoEncoder (VLAE). Βέβαια, να σημειωθεί ότι η συγκεκριμένη προσέγγιση δεν είναι εφικτό να εφαρμοσθεί στα συγκεκριμένα δίκτυα όπως θα παρουσιαστεί αχολούθως και για το λόγο αυτό στην υποενότητα 4.2.4 πραγματοποιείται η σύνδεση της συγκεκριμένης μεθοδολογίας με τη χρήση των ReLU δικτύων που παρουσιάστηκαν στην υποενότητα 4.2.2.

Έστω λοιπόν ότι η από κοινού κατανομή $p_\theta(\mathbf{x}, \mathbf{z}) = p_\theta(\mathbf{x}|\mathbf{z})p(\mathbf{z})$ μπορεί να εκτιμηθεί και στη συνέχεια, όπως και στους αυτοκωδικοποιητές VAE, επιδιώκεται η εκτίμηση της εκ των υστέρων κατανομής $p_\theta(\mathbf{z}|\mathbf{x}) \propto p_\theta(\mathbf{x}, \mathbf{z})$. Η Λαπλασιανή μέθοδος προσεγγίζει την εκ των υστέρων κατανομή $p_\theta(\mathbf{z}|\mathbf{x})$ με μία Γκαουσιανή κατανομή $q_\lambda(\mathbf{z}|\mathbf{x})$ κεντραρισμένη στη μέγιστη τιμή της λογαριθμοποιημένης κατανομής $\log p_\theta(\mathbf{x}, \mathbf{z})$ (η οποία μεγιστοποιείται στην ίδια τιμή με την κατανομή $p_\theta(\mathbf{x}, \mathbf{z})$ και κατά συνέπεια με την $p_\theta(\mathbf{z}|\mathbf{x})$), ενώ παράλληλα θεωρεί πίνακα συνδιασπορών της $q_\lambda(\mathbf{z}|\mathbf{x})$ αυτόν που προκύπτει ανάλογα την καμπυλότητα (δηλαδή τη 2η παράγωγο) της συνάρτησης $\log p_\theta(\mathbf{x}, \mathbf{z})$ στη μέγιστη τιμή που εκτιμήθηκε προηγουμένως.

Η συνολική διαδικασία λοιπόν της Λαπλασιανής μεθόδου μπορεί λοιπόν να χωριστεί σε δύο μέρη: (1) τη διαδικασία εκτίμησης της μέγιστης τιμής (mode) της εκ των υστέρων κατανομής και (2) την προσέγγιση της κατανομής αυτής με μία Γκαουσιανή κεντραρισμένη στη μέγιστη τιμή που εκτιμήθηκε στο πρώτο βήμα. Το βήμα μεγιστοποίησης ουσιαστικά πραγματοποιείται από την Εξ. (4.8) μέσω επαναληπτικής διαδικασίας βάσει κάποιας μέθοδου ανόδου κλίσης (gradient-based optimization).

$$\nabla_{\mathbf{z}} \log p_\theta(\mathbf{x}, \mathbf{z})|_{\mathbf{z}=\boldsymbol{\mu}} = \mathbf{0} \quad (4.8)$$

Έπειτα από την εκτίμηση της μέγιστης τιμής της από κοινού λογαριθμοποιημένης κατανομής $\log p_\theta(\mathbf{x}, \mathbf{z})$, το ανάπτυγμα Taylor δεύτερου βαθμού της κατανομής αυτής οδηγεί στις Εξ. (4.9) και (4.10):

$$\log p_\theta(\mathbf{x}, \mathbf{z}) \approx \log p_\theta(\mathbf{x}, \boldsymbol{\mu}) - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Lambda}(\mathbf{x} - \boldsymbol{\mu}), \quad (4.9)$$

$$\text{όπου } \boldsymbol{\Lambda} = -\nabla_{\mathbf{z}}^2 \log p_\theta(\mathbf{x}, \mathbf{z})|_{\mathbf{z}=\boldsymbol{\mu}} \quad (4.10)$$

Σύμφωνα με τις Εξ. (4.9) και (4.10) παρατηρείται ότι η μορφή της από κοινού λογαριθμοποιημένης κατανομής $\log p_\theta(\mathbf{x}, \mathbf{z})$ είναι ισοδύναμη με αυτή μίας λογαριθμοποιημένης Γκαουσιανής κατανομής. Συνεπώς, κατά τη Λαπλασιανή προσέγγιση και η εκ των υστέρων κατανομή θα είναι Γκαουσιανή σύμφωνα με την Εξ. (4.11):

$$q(\mathbf{z}|\mathbf{x}) = N(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \text{ όπου } \boldsymbol{\Sigma}^{-1} = \boldsymbol{\Lambda} \quad (4.11)$$

Εφόσον εκτιμήθηκε η εκ των υστέρων κατανομή $q(\mathbf{z}|\mathbf{x})$ των κρυφών δεδομένων \mathbf{z} , αυτή χρησιμοποιείται στη συνέχεια για εκτίμηση του όρου ELBO στην Εξ. (3.13) που απαιτείται για την εκπαίδευση του δικτύων κωδικοποιητή και αποκωδικοποιητή. Η παραπάνω διαδικασία παρουσιάζεται αναλυτικότερα στον Αλγ. 2.

Αλγόριθμος 2 Variational Λαπλασιανό Αυτοκωδικοποιητές

Είσοδος: δεδομένα \mathbf{x} , inference δίκτυο enc_φ , piece-wise linear generative δίκτυο g_θ , πλήθος ανανέωσεων T .

Αρχικοποίηση $\boldsymbol{\mu}_0 \leftarrow enc_\varphi(\mathbf{x})$.

Εκτίμηση της από κοινού λογαριθμοποιημένης: $logp_\theta(\mathbf{x}, \mathbf{z}) = logp_\theta(\mathbf{x}|\mathbf{z}) + logp(\mathbf{z})$

for $t = 0$ to $T - 1$ **do**

 Ανανέωση μέσης τιμής $\boldsymbol{\mu}$ της $logp_\theta(\mathbf{x}, \mathbf{z})$ (π.χ. με gradient descent)

end for

$\boldsymbol{\Sigma} \leftarrow (-\nabla_{\mathbf{z}}^2 logp_\theta(\mathbf{x}, \mathbf{z}|\mathbf{z} = \boldsymbol{\mu}))^{-1}$

 ▷ Εκτίμηση πίνακα συνδιασπορών κατανομής $q(\mathbf{z}|\mathbf{x})$

$q(\mathbf{z}|\mathbf{x}) \leftarrow N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$

 ▷ Προσέγγιση κατανομής $p_\theta(\mathbf{z}|\mathbf{x})$ με Γκαουσιανή κατανομή

$\boldsymbol{\Sigma} = \mathbf{L}\mathbf{L}^T$

 ▷ Cholesky Decomposition στον $\boldsymbol{\Sigma}$

$\mathbf{z} \leftarrow \boldsymbol{\mu} + \mathbf{L} \cdot \boldsymbol{\varepsilon}$

 ▷ όπου δείγμα $\boldsymbol{\varepsilon} \sim N(\mathbf{0}, \mathbf{I})$

Υπολογισμός του ELBO: $L(\phi, \theta) = logp_\theta(\mathbf{x}, \mathbf{z}) - logq(\mathbf{z}|\mathbf{x})$

$\theta \leftarrow +\alpha \nabla_\theta L(\phi, \theta)$

 ▷ Ανανέωση του generative μοντέλου

$\phi \leftarrow \phi + \alpha \nabla_\phi L(\phi, \theta)$

 ▷ Ανανέωση του inference μοντέλου

Όπως όμως προαναφέρθηκε στην αρχή της παρούσας ενότητας η παραπάνω μεθοδολογία της Λαπλασιανής προσέγγισης είναι αδύνατον να εφαρμοστεί. Αυτό συμβαίνει διότι: (1) η εκτίμηση της μέσης τιμής $\boldsymbol{\mu}$ της λογαριθμοποιημένης κατανομής $logp_\theta(\mathbf{x}, \mathbf{z})$ με κάποιο gradient-based optimization αλγόριθμο αποτελεί μία χρονοβόρα διαδικασία που θα πρέπει να επαναλαμβάνεται σε κάθε εποχή της εκπαίδευσης του αυτοκωδικοποιητή VLAE και (2) απαιτείται η δεύτερη παράγωγος της κατανομής $logp_\theta(\mathbf{x}, \mathbf{z})$ για την εκτίμηση του πίνακα συνδιασπορών $\boldsymbol{\Sigma}$, η οποία θεωρείται επίσης αρκετά χρονοβόρα διαδικασία. Για το λόγο αυτό στην υποενότητα 4.2.4 που ακολουθεί περιγράφεται συνοπτικά η σύνδεση των Αλγ. 1 και 2 και αποδεικνύεται ότι η μεθοδολογία των ReLU δικτύων της υποενότητας 4.2.2 αποτελεί μία ειδική περίπτωση της Λαπλασιανής μεθόδου της παρούσας ενότητας.

4.2.4 Σύνδεση Λαπλασιανής Προσέγγισης με ReLU Δίκτυα

Ακολουθώς πραγματοποιείται η σύγκριση της Λαπλασιανής προσέγγισης με τη μεθοδολογία που παρουσιάστηκε στην υποενότητα 4.2.2 από τη χρήση των ReLU νευρωνικών δικτύων. Έστω λοιπόν το δίκτυο του αποκωδικοποιητή που περιγράφεται από τις Εξ. (3.9) και (3.10). Σύμφωνα με τις εξισώσεις αυτές, εφόσον η δεσμευμένη συνάρτηση πιθανοφάνειας $p_\theta(\mathbf{x}|\mathbf{z})$ και η εκ των προτέρων κατανομή $p(\mathbf{z})$ των κρυφών δεδομένων \mathbf{z} είναι γκαουσιανές κατανομές, έπεται ότι και η από κοινού κατανομή $p_\theta(\mathbf{x}, \mathbf{z}) = p_\theta(\mathbf{x}|\mathbf{z})p(\mathbf{z})$ θα είναι Γκαουσιανή. Συνεπώς, η λογαριθμοποιημένη αυτή κατανομή προκύπτει:

$$logp_\theta(\mathbf{x}, \mathbf{z}) = -\frac{1}{2\sigma^2} (\mathbf{x} - g_\theta(\mathbf{z}))^T (\mathbf{x} - g_\theta(\mathbf{z})) - \frac{1}{2} \mathbf{z}^T \mathbf{z} + C \quad (4.12)$$

Για την εύρεση της μέγιστης τιμής της εκ των υστέρων κατανομής $p_\theta(\mathbf{z}|\mathbf{x})$, η από κοινού λογαριθμοποιημένη κατανομή $logp_\theta(\mathbf{z}, \mathbf{x})$ παραγωγίζεται ως προς \mathbf{z} και η παράγωγος τίθεται ίση με 0:

$$\nabla_{\mathbf{z}} \log p_{\theta}(\mathbf{x}, \mathbf{z}) = -\frac{1}{\sigma^2} \frac{\partial g_{\theta}(\mathbf{z})^T}{\partial \mathbf{z}} (g_{\theta}(\mathbf{z}) - \mathbf{x}) - \mathbf{z} = \mathbf{0} \quad (4.13)$$

Αντικαθιστώντας το μη γραμμικό μετασχηματισμό $g_{\theta}(\mathbf{z})$ από την Εξ. (4.5) των ReLU δικτύων, η Εξ. (4.13) γράφεται:

$$\boldsymbol{\mu} = \frac{1}{\sigma^2} \left(\frac{1}{\sigma^2} \mathbf{W}_z^T \mathbf{W}_z + \mathbf{I} \right)^{-1} \mathbf{W}_z^T (\mathbf{x} - \mathbf{b}_z) \quad (4.14)$$

η οποία είναι ισοδύναμη με τη συνάρτηση ανανέωσης της μέσης τιμής $\boldsymbol{\mu}$ του Αλγ. 1. Επίσης, από τις Εξ. (4.10) και (4.14) παρατηρείται ότι ισχύει:

$$\boldsymbol{\Lambda} = -\nabla_{\mathbf{z}}^2 \log p_{\theta}(\mathbf{x}, \mathbf{z})|_{\mathbf{z}=\boldsymbol{\mu}} = \left(\frac{1}{\sigma^2} \mathbf{W}_z^T \mathbf{W}_z + \mathbf{I} \right) \quad (4.15)$$

τιμή η οποία είναι ισούται με $\boldsymbol{\Sigma}^{-1}$ της Εξ. (4.7). Συνεπώς, η χρήση των τοπικά γραμμικών νευρωνικών δικτύων (piece-wise linear networks) αποτελούν μία ειδική περίπτωση της Λαπλασιανής προσέγγισης, στην οποία θεωρείται ότι η από κοινού κατανομή των δεδομένων $p_{\theta}(\mathbf{x}, \mathbf{z})$ ακολουθεί κανονική κατανομή. Τέλος, να σχολιαστεί ότι η εφαρμογή της Λαπλασιανής προσέγγισης απαιτεί τη δεύτερη παράγωγο της λογαριθμοποιημένης Γκαουσιανής κατανομής $p_{\theta}(\mathbf{x}, \mathbf{z})$ (δηλαδή το Hessian πίνακα) για την εκτίμηση του πίνακα συνδιασπορών $\boldsymbol{\Sigma}$, με αποτέλεσμα να απαιτεί μεγαλύτερο υπολογιστικό κόστος από τα piece-wise linear νευρωνικά δίκτυα της υποενότητας 4.2.2.

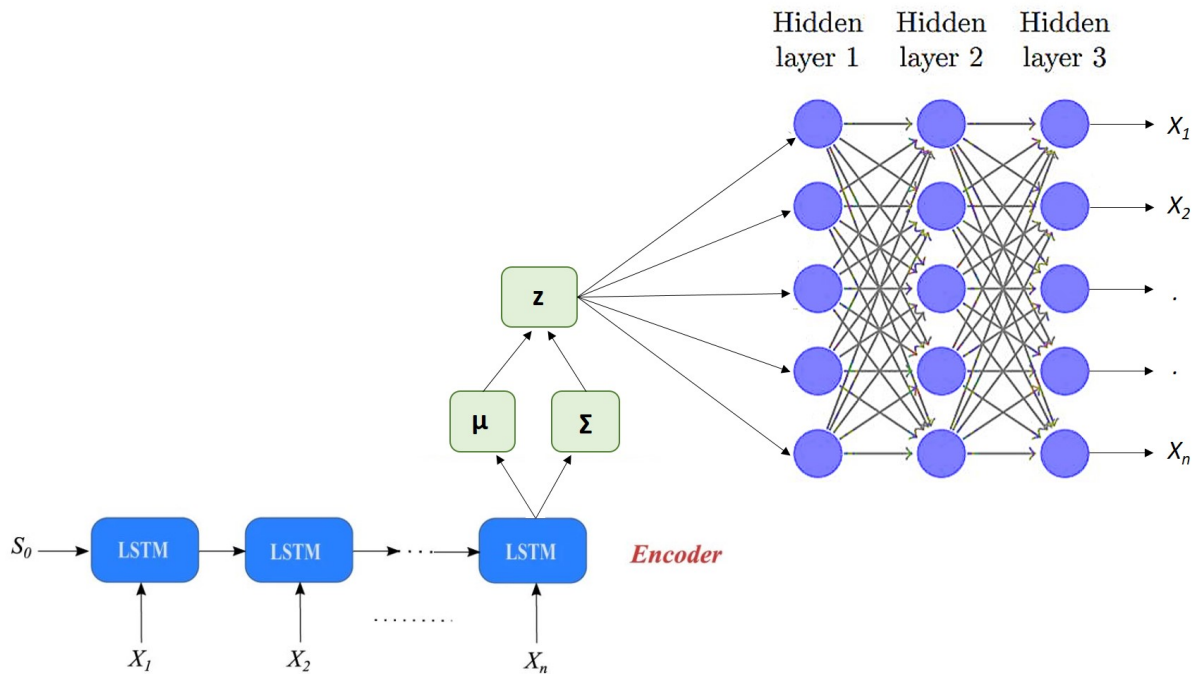
4.3 Γραφική Αναπαράσταση Προηγμένων Μοντέλων

Έως τώρα έχουν παρουσιαστεί 2 προηγμένες μέθοδοι variational συμπερασματολογίας, οι οποίες χρησιμοποιούνται για να βελτιώσουν την απόδοση των μοντέλων VAE. Συγκεκριμένα, στην Εν. 4.1 παρουσιάστηκε η μέθοδος της semi-amortized συμπερασματολογίας, ενώ στην Εν. 4.2 αναλύθηκε η Λαπλασιανή μέθοδος και πώς αυτή μπορεί να ενσωματωθεί στα δίκτυα VAE. Παρόλα αυτά όμως, οι μέθοδοι αυτές εισάγουν έναν περιορισμό στο τμήμα του αποκωδικοποιητή των νευρωνικών δικτύων. Συγκεκριμένα, τόσο στη μία μέθοδο όσο και στην άλλη απαιτείται η χρήση μόνο νευρωνικών δικτύων εμπρόσθιας τροφοδότησης MLP για να μπορέσει να πραγματοποιηθεί η εκπαίδευση. Οι λόγοι που οδηγούν στον περιορισμό αυτό είναι οι εξής:

1. Όσον αφορά τη semi-amortized συμπερασματολογία, σύμφωνα με την Εξ. (4.2) απαιτείται η εκτίμηση του Hessian πίνακα της ποσότητας ELBO ως προς την παράμετρο λ . Ο πίνακας αυτός ουσιαστικά αποτελεί τη δεύτερη παράγωγο της ποσότητας ELBO, η οποία όμως δεν μπορεί να εκτιμηθεί με τη βιβλιοθήκη Pytorch [57] εκτός και αν το δίκτυο του αποκωδικοποιητή είναι δίκτυο εμπρόσθιας τροφοδότησης MLP. Αυτό συμβαίνει διότι η βιβλιοθήκη αυτή δεν παρέχει τη δυνατότητα οπισθοδιάδοσης διανυσματικών παραγώγων 2ης ή μεγαλύτερης τάξης σε νευρωνικά δίκτυα διαφορετικού τύπου από το δίκτυο MLP.
2. Η Λαπλασιανή μεθοδολογία μπορεί να εφαρμοσθεί κυρίως στις περιπτώσεις όπου το δίκτυο του αποκωδικοποιητή είναι ένα πολυστρωματικό δίκτυο εμπρόσθιας τροφοδότησης MLP με ReLU συναρτήσεις ενεργοποίησης. Αντιθέτως, τα νευρωνικά δίκτυα LSTM δεν μπορούν να χρησιμοποιηθούν εύκολα για την εκτίμηση του πίνακα \mathbf{W}_z της Εξ. (4.5). Αυτό συμβαίνει διότι ως μη γραμμική συνάρτηση ενεργοποίησης χρησιμοποιούν τη συνάρτηση \tanh , ενώ παράλληλα στηρίζονται στη λογιστική συνάρτηση για την εφαρμογή των πυλών.

Εφόσον όμως στην παρούσα εργασία μελετούνται χρονοσειρές, για την εφαρμογή της semi-amortized συμπερασματολογίας και της Λαπλασιανής μεθόδου στα σήματα των χρονοσειρών θα χρησιμοποιηθούν δίκτυα LSTM στο τμήμα του κωδικοποιητή, ενώ αντιθέτως στο τμήμα του αποκωδικοποιητή ReLU πολυστρωματικά δίκτυα εμπρόσθιας τροφοδότησης MLP. Η συγκεκριμένη

αρχιτεκτονική απεικονίζεται στο Σχ. 4.1 και θα αναφέρεται από εδώ και στο εξής ως SAV-LSTM/MLP-AE (δηλαδή Semi-Amortized Variational LSTM/MLP AutoEncoder) για τη semi-amortized συμπερασματολογία και VL-LSTM/MLP-AE (δηλαδή Variational Laplace LSTM/MLP AutoEncoder) για τη Λαπλασιανή μέθοδο, αντίστοιχα.



Σχήμα 4.1: Αρχιτεκτονική variational αυτοκωδικοποιητή σε συνδυασμό με τα νευρωνικά δίκτυα LSTM στο τμήμα του κωδικοποιητή και τα ReLU πολυστρωματικά δίκτυα εμπρόσθιας τροφοδότησης MLP στο τμήμα του αποκωδικοποιητή.

Κεφάλαιο 5

Κανονικοποιημένες Ροές

Στο παρόν Κεφάλαιο πραγματοποιείται μία εισαγωγή στις κανονικοποιημένες ροές, οι οποίες θεωρούνται ειδικές μέθοδοι για το μετασχηματισμό μίας απλής στατιστικής κατανομής σε μία πιο περίπλοκη χρησιμοποιώντας κατάλληλες συναρτήσεις μετασχηματισμού. Συγκεκριμένα, στην Εν. 5.1 παρουσιάζεται η μαθηματική ανάλυση που διέπει τις κανονικοποιημένες ροές και απαριθμούνται ορισμένα πλεονεκτήματά τους έναντι άλλων παρόμοιων μεθόδων. Ακολούθως, στην Εν. 5.2 οι κανονικοποιημένες ροές ενσωματώνονται στα νευρωνικά δίκτυα VAE και παρουσιάζεται ο τρόπος με τον οποίο μπορούμε να δημιουργήσουμε υβριδικά και πιο αποτελεσματικά μοντέλα συμπίεσης δεδομένων. Τέλος, στην Εν. 5.3 αναλύεται η κανονικοποιημένη ροή Householder, η οποία χρησιμοποιείται στη συγκεκριμένη εργασία με στόχο τη βελτίωση της απόδοσης των απλών αυτοκωδικοποιητών VAE.

5.1 Εισαγωγικές Έννοιες στις Ροές

Οι κανονικοποιημένες ροές (Normalizing Flows - NF) [58, 59] αποτελούν μία κατηγορία μεθόδων, βάσει των οποίων μπορούμε να μετασχηματίσουμε μία απλή στατιστική κατανομή σε μία πιο περίπλοκη χρησιμοποιώντας ειδικές συναρτήσεις μετασχηματισμού. Οι συναρτήσεις αυτές πρέπει να έχουν συγκεκριμένες ιδιότητες, όπως για παράδειγμα θα πρέπει να είναι διαφορίσιμες και αντιστρέψιμες (differentiable και invertible functions). Εφαρμόζοντας λοιπόν κάποια μέθοδο κανονικοποιημένων ροών, τα δεδομένα της απλής κατανομής ρέουν (flow) μέσα από τις συναρτήσεις αυτές και υπόκεινται σε διαδοχικούς μετασχηματισμούς. Με αυτόν τον τρόπο παράγονται δεδομένα από μία πιο περίπλοκη κατανομή με διαφορετικές ιδιότητες από την αρχική. Ακολούθως λοιπόν πραγματοποιείται μία εισαγωγή στη μαθηματική μοντελοποίηση των κανονικοποιημένων ροών και επισημαίνονται ορισμένα πλεονεκτήματα έναντι άλλων μεθόδων.

Μαθηματική Μοντελοποίηση Ροών

Έστω ένα παρατηρούμενο σήμα $\mathbf{x} \in \mathbb{R}^D$ μίας κατανομής $p(\mathbf{x})$, το οποίο θεωρούμε ότι προκύπτει από το ίδιας διαστατικότητας, άγνωστο σήμα $\mathbf{z} \in \mathbb{R}^D$ της απλής κατανομής $p(\mathbf{z})$ με χρήση ενός μετασχηματισμού. Ας υποθέσουμε στη συνέχεια ότι ο μετασχηματισμός αυτός ισοδυναμεί με μία συνάρτηση $\mathbf{f} = (f_1, \dots, f_D) : \mathbb{R}^D \rightarrow \mathbb{R}^D$, δηλαδή ισχύει: $\mathbf{x} = \mathbf{f}(\mathbf{z})$. Χρησιμοποιώντας τον κανόνα αλλαγής μεταβλητής, οι 2 αυτές κατανομές συνδέονται μεταξύ τους μέσα από την Εξ. (5.1):

$$\begin{aligned} p(\mathbf{x}) &= p(\mathbf{z}) \left| \det \left(\frac{d\mathbf{z}}{d\mathbf{x}} \right) \right| = p(\mathbf{z}) \left| \det \left(\frac{d\mathbf{x}}{d\mathbf{z}} \right)^{-1} \right| \longrightarrow \\ p(\mathbf{x}) &= p(\mathbf{z}) \left| \det \left(\frac{d\mathbf{f}(\mathbf{z})}{d\mathbf{z}} \right) \right|^{-1} \longrightarrow \\ p(\mathbf{x}) &= p(\mathbf{z}) |\det \nabla_{\mathbf{z}} \mathbf{f}(\mathbf{z})|^{-1} \end{aligned} \quad (5.1)$$

όπου η ποσότητα \det αποτελεί την ορίζουσα της 1ης παραγώγου (ή αλλιώς Ιακωβιανής μήτρας) της συνάρτησης $\mathbf{f}(\mathbf{z})$ ως προς τη μεταβλητή \mathbf{z} και εξασφαλίζει ότι η συνάρτηση $p(\mathbf{x})$ είναι μία κατανομή που ολοκληρώνει στη μονάδα. Στόχος στην προκείμενη περίπτωση είναι να βρεθεί η κατάλληλη συνάρτηση \mathbf{f} , η οποία θα μετασχηματίζει κάθε δεδομένο \mathbf{z} της απλής κατανομής $p(\mathbf{z})$ στο αντίστοιχο δεδομένο \mathbf{x} της κατανομής $p(\mathbf{x})$.

Αν λοιπόν υποθέσουμε ότι έχουμε ένα σύνολο δεδομένων \mathbf{X} που αποτελείται από N δεδομένα \mathbf{x} , τότε η εκτίμηση της συνάρτησης \mathbf{f} θα μπορούσε να πραγματοποιηθεί με τη μέθοδο της μέγιστης πιθανοφάνειας. Συγκεκριμένα, θεωρώντας όλα τα δεδομένα $\mathbf{x}_i, \mathbf{z}_i$ ανεξάρτητα μεταξύ τους (i.i.d) και λογαριθμοποιώντας τη συνάρτηση πιθανοφάνειας $p(\mathbf{X})$ του συνόλου δεδομένων, η Εξ. (5.1) γράφεται στη μορφή:

$$\log p(\mathbf{X}) = \sum_{i=1}^N [\log p(\mathbf{z}_i) - \log |\det \nabla_{\mathbf{z}} \mathbf{f}(\mathbf{z}_i)|] \quad (5.2)$$

Το πρόβλημα που προκύπτει στην Εξ. (5.2) είναι ότι τα δεδομένα \mathbf{z}_i δεν είναι γνωστά. Συνεπώς, για να επιτευχθεί η πρόσβαση σε κάθε \mathbf{z}_i θα πρέπει να χρησιμοποιηθεί ο αντίστροφος μετασχηματισμός της συνάρτησης \mathbf{f} , δηλαδή $\mathbf{z}_i = \mathbf{f}^{-1}(\mathbf{x}_i)$. Αυτό βέβαια προϋποθέτει ότι η συνάρτηση \mathbf{f} είναι πλήρως αντιστρέψιμη (invertible), δηλαδή με χρήση της συγκεκριμένης συνάρτησης κάθε δεδομένο \mathbf{z}_i αντιστοιχίζεται σε ένα μοναδικό δεδομένο \mathbf{x}_i και αντίστροφα. Συνεπώς, η λογαριθμοποιημένη συνάρτηση πιθανοφάνειας της Εξ. (5.2) γράφεται:

$$\log p(\mathbf{X}) = \sum_{i=1}^N [\log p(\mathbf{f}^{-1}(\mathbf{x}_i)) - \log |\det \nabla_{\mathbf{x}} \mathbf{f}(\mathbf{f}^{-1}(\mathbf{x}_i))|] \quad (5.3)$$

όπου αν θεωρήσουμε ότι η συνάρτηση \mathbf{f} έχει γνωστή μορφή, τότε μπορούμε να εκτιμήσουμε τις παραμέτρους της παραγωγίζοντας τη λογαριθμοποιημένη συνάρτηση πιθανοφάνειας της Εξ. (5.3). Το ερώτημα όμως που τίθεται είναι ποιά μπορεί να είναι η μορφή της συγκεκριμένης συνάρτησης \mathbf{f} , η οποία συνδέει τα δεδομένα των κατανομών $p(\mathbf{x})$ και $p(\mathbf{z})$;

Για να απαντηθεί το παραπάνω ερώτημα θα πρέπει να δοθεί έμφαση στη μορφή των προαναφερθέντων εξισώσεων. Συγκεκριμένα, από το δεύτερο όρο της Εξ. (5.2) παρατηρείται ότι για την εκτίμηση της συνάρτησης πιθανοφάνειας απαιτείται ο υπολογισμός της Ιακωβιανής μήτρας της συνάρτησης \mathbf{f} ως προς $\mathbf{z} = \mathbf{f}^{-1}(\mathbf{x})$. Η μήτρα αυτή ουσιαστικά ισούται με:

$$\nabla_{\mathbf{z}}\mathbf{f}(\mathbf{z}) = \begin{bmatrix} \frac{\partial f_1}{\partial z_1} & \cdots & \frac{\partial f_1}{\partial z_D} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_D}{\partial z_1} & \cdots & \frac{\partial f_D}{\partial z_D} \end{bmatrix} \quad (5.4)$$

Η εκτίμηση της παραπάνω μήτρας αποτελεί μία χρονοβόρα διαδικασία με υπολογιστική πολυπλοκότητα $O(D^3)$. Ίδια πολυπλοκότητα προκύπτει και για την εκτίμηση της αντίστροφης συνάρτησης \mathbf{f}^{-1} της Εξ. (5.3). Για τη μείωση λοιπόν του υπολογιστικού κόστους έχουν προταθεί 2 διαφορετικοί τρόποι επιλογής της συνάρτησης \mathbf{f} , όπως παρουσιάζονται ακολουθώς:

1. General Normalizing Flows: Στη συγκεκριμένη κατηγορία ροών η συνάρτηση \mathbf{f} επιλέγεται με τέτοιο τρόπο, ώστε η Ιακωβιανή μήτρα να είναι συνήθως ένας άνω ή κάτω τριγωνικός πίνακας. Με αυτή την τεχνική έπεται ότι η ορίζουσα της μήτρας θα ισούται με το γινόμενο των διαγώνιων στοιχείων της. Συνεπώς, το υπολογιστικό κόστος μειώνεται σε $O(D)$ για την ορίζουσα της Ιακωβιανής μήτρας και $O(D^2)$ για την εκτίμηση της αντίστροφης συνάρτησης \mathbf{f}^{-1} . Εναλλακτικά, η συνάρτηση \mathbf{f} θα μπορούσε να επιλεγεί με τέτοιο τρόπο ώστε η Ιακωβιανή μήτρα να προκύψει ένας διαγώνιος πίνακας. Αν και η συγκεκριμένη τεχνική υπεραπλουστεύει το πρόβλημα, η υπολογιστική πολυπλοκότητα μειώνεται δραματικά σε $O(D)$ για την εκτίμηση τόσο της ορίζουσας όσο της αντίστροφης συνάρτησης \mathbf{f}^{-1} . Σε κάθε περίπτωση, η λογαριθμοποιημένη συνάρτηση πιθανοφάνειας της Εξ. (5.3) γράφεται:

$$\log p(\mathbf{X}) = \sum_{i=1}^N \left[\log p(\mathbf{f}^{-1}(\mathbf{x}_i)) - \sum_{j=1}^D \log \partial_j f_j(\mathbf{f}^{-1}(\mathbf{x}_i)) \right] \quad (5.5)$$

2. Volume Preserving Flows: Στη συγκεκριμένη περίπτωση η συνάρτηση \mathbf{f} επιλέγεται με τέτοιο τρόπο, ώστε η ορίζουσα της Ιακωβιανής μήτρας να ισούται 1. Κατά συνέπεια, ο λογάριθμος της ορίζουσας αυτής μηδενίζεται. Για το λόγο αυτό οι ροές αυτού του τύπου χρησιμοποιούνται αρκετά συχνά, καθώς μειώνουν δραματικά το υπολογιστικό κόστος εκτίμησης της συνάρτησης \mathbf{f} . Βέβαια στις περιπτώσεις όπου η ορίζουσα ισούται με 1, τίθεται ο περιορισμός ότι ο "όγκος" (volume) της κατανομής $p(\mathbf{X})$ των παρατηρούμενων δεδομένων πρέπει να είναι ίσος με τον αντίστοιχο "όγκο" της απλής κατανομής $p(\mathbf{Z})$. Συνεπώς, ο περιορισμός αυτός μπορεί να οδηγήσει σε μία απλουστευμένη κατανομή $p(\mathbf{X})$. Στην περίπτωση αυτή, η λογαριθμοποιημένη συνάρτηση πιθανοφάνειας της Εξ. (5.3) απλοποιείται ως εξής:

$$\log p(\mathbf{X}) = \sum_{i=1}^N \left[\log p(\mathbf{f}^{-1}(\mathbf{x}_i)) - \log \left| \det \nabla_{\mathbf{x}} \mathbf{f}(\mathbf{f}^{-1}(\mathbf{x}_i)) \right| \right] \quad (5.6)$$

Ορισμένα πλεονεκτήματα που προκύπτουν από τη χρήση των κανονικοποιημένων ροών αναφέρονται συνοπτικά ακολούθως:

1. Το πρόβλημα μεγιστοποίησης αναφέρεται στην ίδια τη συνάρτηση πιθανοφάνειας, την οποία μπορούμε να εκτιμήσουμε αναλυτικά για το σύνολο των N δεδομένων. Αυτό αποτελεί ένα ισχυρό πλεονέκτημα έναντι των variational αυτοκωδικοποιητών, στους οποίους μεγιστοποιούμε ένα κάτω όριο της συνάρτησης αυτής, γνωστό ως ELBO (δες Εν. 3.2).
2. Ύστερα από την εκτίμηση της συνάρτησης \mathbf{f} αλλά και της αντίστροφης \mathbf{f}^{-1} , για κάθε δεδομένο \mathbf{x}_i μπορεί να εκτιμηθεί η ακριβής τιμή της λογαριθμοποιημένης συνάρτησης πιθανοφάνειας $\log p(\mathbf{x}_i)$ (density estimation). Αυτό δε συμβαίνει στα μοντέλα αυτοκωδικοποιητών, όπου ουσιαστικά μπορούμε να εκτιμήσουμε μόνο την τιμή της δεσμευμένης συνάρτησης πιθανοφάνειας $\log p(\mathbf{x}_i|\mathbf{z}_i)$ που προκύπτει από την κρυφή αναπαράσταση \mathbf{z}_i .
3. Ύστερα από την εκτίμηση της αντίστροφης συνάρτησης \mathbf{f}^{-1} , μπορούμε να εκτιμήσουμε ακριβώς την κρυφή αναπαράσταση \mathbf{z}_i από τα παρατηρούμενα δεδομένα \mathbf{x}_i μέσω της σχέσης $\mathbf{z}_i = \mathbf{f}^{-1}(\mathbf{x}_i)$. Συνεπώς, η εκ των υστέρων συμπερασματολογία για τα κρυφά δεδομένα πραγματοποιείται με ακρίβεια. Αυτό δε συμβαίνει στους variational αυτοκωδικοποιητές, όπου για κάθε δεδομένο \mathbf{x}_i εκτιμάται μία προσεγγιστική εκ των υστέρων κατανομή $q(\mathbf{z}|\mathbf{x}, \phi)$ των κρυφών δεδομένων, από την οποία προκύπτει η συμπίεσμένη αναπαράσταση \mathbf{z}_i ύστερα από κατάλληλη δειγματοληψία.
4. Τέλος, στις περιπτώσεις όπου η κατανομή των δεδομένων \mathbf{z} επιλέγεται να είναι αρκετά απλή (π.χ. Standard Gaussian), μπορεί να πραγματοποιηθεί αρκετά εύκολα δειγματοληψία (sampling) από την κατανομή αυτή. Συνεπώς, με χρήση της νέας κρυφής αναπαράστασης \mathbf{z}_{new} μπορεί να δημιουργηθούν νέα δεδομένα σύμφωνα με τη σχέση: $\mathbf{x}_{new} = \mathbf{f}(\mathbf{z}_{new})$, εμπλουτίζοντας το αρχικό σύνολο δεδομένων. Κατά συνέπεια, οι κανονικοποιημένες ροές ανήκουν στην κατηγορία των γενικευμένων μοντέλων (generative models), με βάση τα οποία μπορούν να παραχθούν νέα δεδομένα από μία απλή κατανομή.

Στην παρούσα εργασία μελετάται πώς μπορούν να συνδυαστούν οι κανονικοποιημένες ροές μαζί με τους variational αυτοκωδικοποιητές, με στόχο τη βελτίωση της προσεγγιστικής κατανομής $q(\mathbf{z}|\mathbf{x}, \phi)$ στο μεσοαίο επίπεδο. Μία σύντομη εισαγωγή σε τέτοιους είδους μεθόδους παρουσιάζεται στην ενότητα που ακολουθεί.

5.2 Συνδυασμός Ροών με Μοντέλα VAE

Όπως παρουσιάστηκε στην Εν. 3.2, τα νευρωνικά δίκτυα VAE εκτιμούν μία προσεγγιστική κατανομή των κρυφών δεδομένων $q(\mathbf{z}|\mathbf{x}, \phi)$ από το δίκτυο του κωδικοποιητή. Στη συνέχεια, από την κατανομή αυτή πραγματοποιείται δειγματοληψία και η συμπίεσμένη αναπαράσταση \mathbf{z} οδηγείται στο τμήμα του αποκωδικοποιητή για την εκτίμηση της δεσμευμένης συνάρτησης πιθανοφάνειας $p(\mathbf{x}|\mathbf{z})$. Το πρόβλημα αυτής της μεθοδολογίας είναι ότι στις περισσότερες περιπτώσεις η κατανομή $q(\mathbf{z}|\mathbf{x}, \phi)$ επιλέγεται να είναι Γκαουσιανή με διαγώνιο πίνακα συνδιασπορών, έτσι ώστε να μειωθεί αισθητά το υπολογιστικό κόστος κατά τη διάρκεια της εκπαίδευσης των μοντέλων. Παρόλα αυτά, ο διαγώνιος πίνακας συνδιασπορών δεν ερμηνεύει τις συσχετίσεις των συμπίεσμένων δεδομένων και συνεπώς αυξάνει το σφάλμα μοντελοποίησης τέτοιου είδους νευρωνικών δικτύων.

Για τη μείωση του συγκεκριμένου σφάλματος έχουν προταθεί διάφοροι τρόποι αντιμετώπισης. Ενδεικτικό παράδειγμα αποτελεί η χρήση της Λαπλασιανής μεθόδου που παρουσιάστηκε στην Εν. 4.2, όπου χρησιμοποιώντας και το τμήμα του αποκωδικοποιητή ο πίνακας συνδιασπορών Σ της κατανομής $q(\mathbf{z}|\mathbf{x}, \phi)$ παύει να είναι διαγώνιος, ερμηνεύοντας με αποτελεσματικότερο τρόπο τις

συσχετίσεις των συμπίεσμένων δεδομένων. Στην παρούσα ενότητα παρουσιάζεται ένας εναλλακτικός τρόπος βελτίωσης της κατανομής $q(\mathbf{z}|\mathbf{x}, \phi)$ στο τμήμα του κωδικοποιητή, η οποία μπορεί να επιτευχθεί ύστερα από την αναπροσαρμογή της κατανομής αυτής με χρήση των κανονικοποιημένων ροών.

Έστω λοιπόν ο αυτοκωδικοποιητής VAE του Σχ. 3.3, ο οποίος εκτιμά την προσεγγιστική κατανομή $q(\mathbf{z}_0) = q(\mathbf{z}|\mathbf{x}, \phi)$ από το τμήμα του κωδικοποιητή. Ας υποθέσουμε ότι η κατανομή αυτή είναι μία Γκαουσιανή με διαγώνιο πίνακα συνδιασπορών, η οποία δεν ερμηνεύει τις συσχετίσεις των συμπίεσμένων δεδομένων. Χρησιμοποιώντας όμως τη συλλογιστική των κανονικοποιημένων ροών της προηγούμενης ενότητας, στην απλοϊκή αυτή κατανομή $q(\mathbf{z}_0)$ θα μπορούσε να εφαρμοσθεί ένας κατάλληλος μετασχηματισμός $\mathbf{f} = (f_1, \dots, f_d) : \mathbb{R}^d \rightarrow \mathbb{R}^d$ (όπου d η διάσταση των κρυφών δεδομένων $\mathbf{z} \in \mathbb{R}^d$), με στόχο τη μετατροπή της σε μία Γκαουσιανή $q(\mathbf{z}_1)$ με ολοκληρωμένο πίνακα συνδιασπορών (full covariance). Ουσιαστικά λοιπόν τα αρχικά κρυφά δεδομένα \mathbf{z}_0 μετασχηματίζονται κατάλληλα στα δεδομένα \mathbf{z}_1 μέσω της σχέσης: $\mathbf{z}_1 = \mathbf{f}(\mathbf{z}_0)$. Συνεπώς, οι 2 κατανομές αυτές συνδέονται μεταξύ ως εξής:

$$q(\mathbf{z}_1) = q(\mathbf{z}_0) \left| \det \left(\frac{d\mathbf{z}_0}{d\mathbf{z}_1} \right) \right| = q(\mathbf{z}_0) \left| \det \left(\frac{d\mathbf{z}_1}{d\mathbf{z}_0} \right) \right|^{-1} \longrightarrow$$

$$\log q(\mathbf{z}_1) = \log q(\mathbf{z}_0) - \log \left| \det \left(\frac{d\mathbf{f}(\mathbf{z}_0)}{d\mathbf{z}_0} \right) \right| \quad (5.7)$$

όπου ουσιαστικά η κατανομή $q(\mathbf{z}_1)$ αποτελεί την τροποποιημένη εκ των υστέρων κατανομή των κρυφών δεδομένων. Να σημειωθεί ότι για τον επιτυχή μετασχηματισμό της αρχικής κατανομής $q(\mathbf{z}_0)$, σε αρκετές περιπτώσεις μπορεί να απαιτούνται παραπάνω από ένας μετασχηματισμοί. Αν λοιπόν υποθέσουμε ότι η συνάρτηση \mathbf{F} αποτελείται από K διαδοχικούς μετασχηματισμούς, δηλαδή ισχύει: $\mathbf{F} = \mathbf{f}_1 \circ \mathbf{f}_2 \circ \dots \circ \mathbf{f}_K$, τότε η Εξ. (5.7) γράφεται:

$$\log q(\mathbf{z}_K) = \log q(\mathbf{z}_0) - \sum_{i=1}^K \log \left| \det \left(\frac{d\mathbf{f}_i(\mathbf{z}_i)}{d\mathbf{z}_i} \right) \right| \quad (5.8)$$

Συνεπώς, χρησιμοποιώντας συνδυαστικά την Εξ. (3.18) της ποσότητας ELBO με την Εξ. (5.8) της αναπροσαρμοσμένης εκ των υστέρων κατανομής των κρυφών δεδομένων, η ποσότητα ELBO των αυτοκωδικοποιητών VAE γράφεται:

$$ELBO(\phi, \theta) = E_{q(\mathbf{z}_0)}[\log p_\theta(\mathbf{X}|\mathbf{Z}_K)] - D_{KL}(q_\lambda(\mathbf{Z}_K|\mathbf{X}, \phi) || p(\mathbf{Z}_K)) \longrightarrow$$

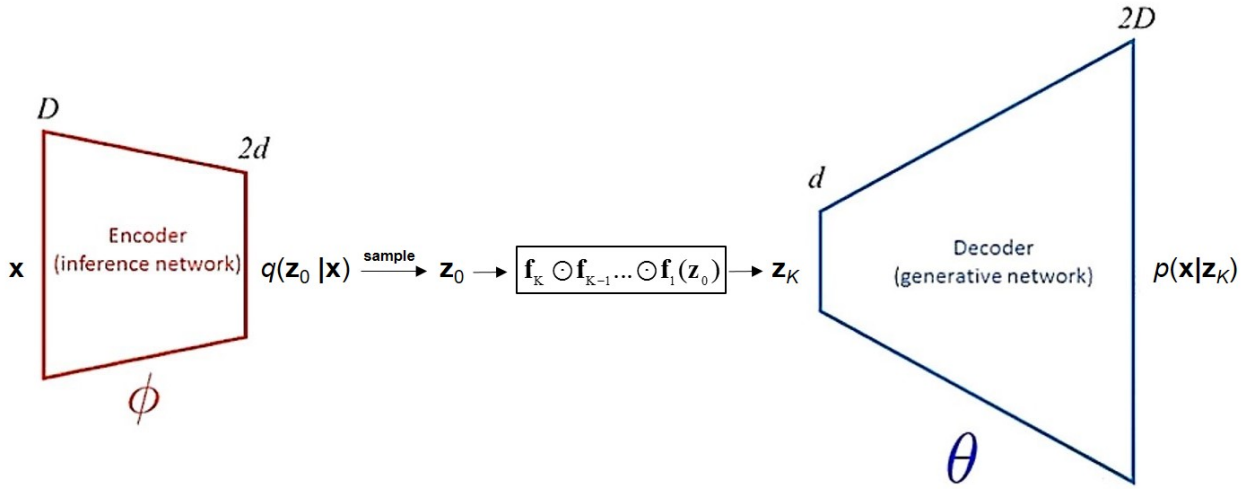
$$ELBO(\phi, \theta) = E_{q(\mathbf{z}_0)}[\log p_\theta(\mathbf{X}|\mathbf{Z}_K)] - E_{q(\mathbf{z}_0)}[\log q(\mathbf{Z}_K)] + E_{q(\mathbf{z}_0)}[\log p(\mathbf{Z}_K)] \longrightarrow$$

$$ELBO(\phi, \theta) = E_{q(\mathbf{z}_0)}[\log p_\theta(\mathbf{X}|\mathbf{Z}_K)] - E_{q(\mathbf{z}_0)} \left[\log q(\mathbf{Z}_0) - \sum_{i=1}^K \log \left| \det \left(\frac{d\mathbf{f}_i}{d\mathbf{Z}_i} \right) \right| \right] + E_{q(\mathbf{z}_0)}[\log p(\mathbf{Z}_K)] \rightarrow$$

$$ELBO(\phi, \theta) = E_{q(\mathbf{z}_0)} \left[\log p_\theta(\mathbf{X}|\mathbf{Z}_K) + \sum_{i=1}^K \log \left| \det \nabla_{\mathbf{z}} \mathbf{f}_i \right| \right] - D_{KL}(q(\mathbf{Z}_0) || p(\mathbf{Z}_K)) \quad (5.9)$$

Η προαναφερθείσα μεθοδολογία ουσιαστικά ενσωματώνει τη μέθοδο των κανονικοποιημένων ροών στη συνάρτηση κόστους ELBO των variational αυτοκωδικοποιητών. Η συγκεκριμένη τεχνική παρουσιάστηκε για πρώτη φορά από τους συγγραφείς στο [60], οι οποίοι χρησιμοποίησαν έναν μετασχηματισμό της μορφής: $\mathbf{F} = \mathbf{f}_1 \circ \mathbf{f}_2 \circ \dots \circ \mathbf{f}_K$, όπου οι συναρτήσεις \mathbf{f}_i επιλέχθηκαν να είναι απλοί μη γραμμικοί μετασχηματισμοί των συμπίεσμένων δεδομένων \mathbf{z}_i . Βέβαια, θα πρέπει να σημειωθεί ότι οι συναρτήσεις \mathbf{f}_i επιλέχθηκαν με τέτοιο τρόπο, ώστε να είναι εφικτή η εκτίμηση της Ιακωβιανής μήτρας στην Εξ. (5.9). Κατά συνέπεια, με χρήση της τεχνικής αυτής κατάφεραν να βελτιώσουν την κατανομή των συμπίεσμένων δεδομένων στο μεσαίο επίπεδο.

Η προαναφερθείσα μεθοδολογία αναπαριστάται στο Σχ. 5.1, απεικονίζοντας γραφικά τη συνδυαστική χρήση των κανονικοποιημένων ροών με τα νευρωνικά δίκτυα VAE.



Σχήμα 5.1: Αρχιτεκτονική variational αυτοκωδικοποιητή σε συνδυασμό τη χρήση των κανονικοποιημένων ροών στο μεσσαίο επίπεδο.

5.3 Κανονικοποιημένη Ροή Householder

Στην παρούσα εργασία μελετάται η κανονικοποιημένη ροή Householder (HF) σε συνδυασμό με τους variational αυτοκωδικοποιητές. Η συγκεκριμένη τεχνική παρουσιάστηκε για πρώτη φορά από τους συγγραφείς στο [61] και ανήκει στην κατηγορία των volume preserving ροών, δηλαδή η συνάρτηση μετασχηματισμού \mathbf{f} επιλέγεται με τέτοιο τρόπο ώστε η ορίζουσα της Ιακωβιανής μήτρας να ισούται με 1. Ακολούθως λοιπόν παρουσιάζεται η συγκεκριμένη μεθοδολογία και εξηγείται με ποιόν τρόπο μπορεί να χρησιμοποιηθεί η συγκεκριμένη ροή στο μεσσαίο επίπεδο ενός variational αυτοκωδικοποιητή, με στόχο τη βελτίωση της συμπίεσμνης αναπαράστασης των δεδομένων.

Εστω λοιπόν ότι το δίκτυο του κωδικοποιητή εκτιμά μία κανονική κατανομή $q(\mathbf{z}_0)$, από την οποία στη συνέχεια παράγεται ένα δείγμα \mathbf{z}_0 (δες Σχ. 5.1). Η κατανομή αυτή χαρακτηρίζεται από το διάνυσμα της μέσης τιμής $\boldsymbol{\mu}_0$ και το διαγώνιο πίνακα συνδιασπορών $\boldsymbol{\Sigma}_0 = \text{diag}(\boldsymbol{\sigma}^2)$, δηλαδή ισχύει $q(\mathbf{z}_0) \sim N(\boldsymbol{\mu}_0, \text{diag}(\boldsymbol{\sigma}^2))$. Παράλληλα, ως θεωρήσουμε ως διάνυσμα \mathbf{z}_1 ένα νέο άγνωστο δείγμα μίας κανονικής κατανομής $q(\mathbf{z}_1) \sim N(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ με ολοκληρωμένο πίνακα συνδιασπορών. Πραγματοποιώντας αποσύνθεση (eigenvalue decomposition) του πίνακα $\boldsymbol{\Sigma}_1$ προκύπτει το εξής:

$$\boldsymbol{\Sigma}_1 = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^T \longrightarrow \boldsymbol{\Lambda} = \mathbf{U}^T\boldsymbol{\Sigma}_1\mathbf{U} \quad (5.10)$$

όπου ο πίνακας \mathbf{U} είναι ο ορθοκανονικός πίνακας των ιδιοδιανυσμάτων και ο πίνακας $\boldsymbol{\Lambda}$ είναι ο διαγώνιος πίνακας των ιδιοτιμών του $\boldsymbol{\Sigma}_1$. Αν όμως ταυτίσουμε το διαγώνιο πίνακα των ιδιοτιμών $\boldsymbol{\Lambda}$ του άγνωστου δείγματος \mathbf{z}_1 με το διαγώνιο πίνακα συνδιασπορών $\boldsymbol{\Sigma}_0$ του γνωστού δείγματος \mathbf{z}_0 , τότε έπεται ότι το δείγμα \mathbf{z}_0 έχει προκύψει από το δείγμα \mathbf{z}_1 μέσω του μετασχηματισμού: $\mathbf{z}_0 = \mathbf{U}^T\mathbf{z}_1$. Συνεπώς, εφόσον ο πίνακας των ιδιοδιανυσμάτων \mathbf{U} είναι ορθοκανονικός (δηλαδή $\mathbf{U}^{-1} = \mathbf{U}^T$), έπεται ότι το δείγμα \mathbf{z}_1 προκύπτει: $\mathbf{z}_1 = \mathbf{U}\mathbf{z}_0$ με κατανομή $q(\mathbf{z}_1) \sim N(\mathbf{U}\boldsymbol{\mu}_0, \mathbf{U}\text{diag}(\boldsymbol{\sigma}^2)\mathbf{U}^T)$.

Αν λοιπόν με κάποιο τρόπο εκτιμηθεί ένας ορθοκανονικός πίνακας αντίστοιχος του πίνακα \mathbf{U} , τότε τα δεδομένα \mathbf{z}_0 που προκύπτουν από τον κωδικοποιητή θα μπορούσαν να τροποποιηθούν κατάλληλα με σκοπό την παραγωγή νέων δεδομένων \mathbf{z}_1 από μία κατανομή με ολοκληρωμένο πίνακα συνδιασπορών (αντίστοιχο του $\boldsymbol{\Sigma}_1 = \mathbf{U}\text{diag}(\boldsymbol{\sigma}^2)\mathbf{U}^T$). Συνεπώς, στόχος θεωρείται η εύρεση κατάλληλων μεθόδων για την εκτίμηση ενός ορθοκανονικού πίνακα, αντίστοιχο του πίνακα των ιδιοδιανυσμάτων \mathbf{U} .

Θεώρημα 1^ο. Για κάθε ένα $d \times d$ ορθοκανονικό πίνακα \mathbf{U} υπάρχει σίγουρα ένας full-rank $d \times K$ πίνακας \mathbf{Y} (basis) και ένας τριγωνικός $K \times K$ πίνακας \mathbf{S} (kernel, όπου $K \leq d$), έτσι ώστε:

$$\mathbf{U} = \mathbf{I} - \mathbf{Y}\mathbf{S}\mathbf{Y}^T \quad (5.11)$$

όπου η τιμή K ορίζεται ως βαθμός (degree) του ορθοκανονικού πίνακα \mathbf{U} . Επιπλέον, αν γνωρίζουμε τη συγκεκριμένη τιμή K , τότε ο ορθοκανονικός πίνακας \mathbf{U} μπορεί να γραφεί ως εξής.

Θεώρημα 2^ο. Οποιοσδήποτε ορθοκανονικός πίνακας \mathbf{U} του βαθμού K μπορεί να γραφεί ως γινόμενο K μετασχηματισμών Householder [62, 63]:

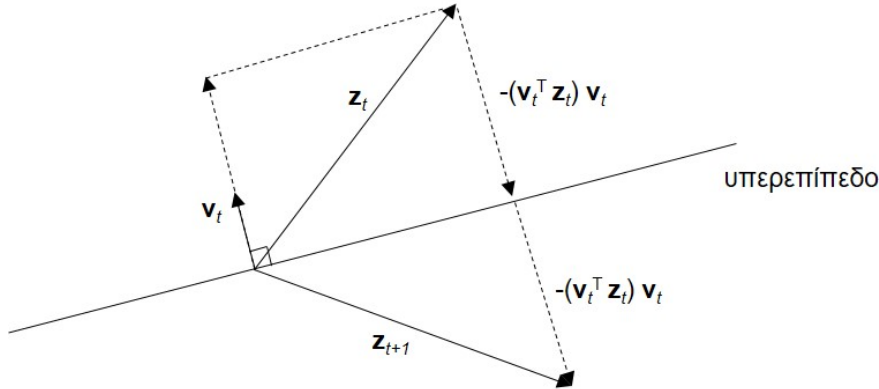
$$\mathbf{U} = \mathbf{H}_K \circ \mathbf{H}_{K-1} \circ \dots \circ \mathbf{H}_1 \quad (5.12)$$

όπου ουσιαστικά κάθε πίνακας \mathbf{H} αποτελεί ένα ξεχωριστό μετασχηματισμό Householder, ο οποίος με τη σειρά του είναι κι αυτός ορθοκανονικός [64]. Κατά συνέπεια, αν θεωρήσουμε ότι η συνάρτηση \mathbf{F} αποτελείται από K διαδοχικούς μετασχηματισμούς: $\mathbf{F} = \mathbf{f}_K \circ \mathbf{f}_{K-1} \circ \dots \circ \mathbf{f}_1 = \mathbf{H}_K \circ \mathbf{H}_{K-1} \circ \dots \circ \mathbf{H}_1$, τότε τα δεδομένα \mathbf{z}_0 που προκύπτουν από τον κωδικοποιητή θα μπορούσαν να τροποποιηθούν σύμφωνα με την Εξ. (5.13):

$$\mathbf{z}_K = \mathbf{H}_K \circ \mathbf{H}_{K-1} \circ \dots \circ \mathbf{H}_1 \circ \mathbf{z}_0 \quad (5.13)$$

όπου τα δεδομένα \mathbf{z}_K θα προέρχονται από την κατανομή $q(\mathbf{z}_K)$ με ολοκληρωμένο πίνακα συνδιασπορών. Το ερώτημα όμως που προκύπτει είναι πώς μπορούμε να εκτιμήσουμε έναν πίνακα Householder \mathbf{H} , έτσι ώστε να μετασχηματίσουμε κατάλληλα τα αρχικά δεδομένα \mathbf{z}_0 ;

Για να απαντηθεί το συγκεκριμένο ερώτημα απαιτείται μία σύντομη ανάλυση γραμμικής άλγεβρας. Έστω λοιπόν ένα διάνυσμα \mathbf{z}_t (όπου $t = 1, \dots, K$) και ένα μοναδιαίο διάνυσμα \mathbf{v}_t (δηλαδή $\|\mathbf{v}_t\| = 1$), όπως αυτά απεικονίζονται στο Σχ. 5.2. Στόχος είναι να βρεθεί ένας ορθοκανονικός πίνακας \mathbf{H} , ο οποίος όταν πολλαπλασιαστεί με το διάνυσμα \mathbf{z}_t οδηγεί σε ένα αντίστοιχο διάνυσμα \mathbf{z}_{t+1} αντικατοπτρικό του πρώτου ως προς το επίπεδο που ορίζεται κάθετα στο μοναδιαίο διάνυσμα \mathbf{v}_t .



Σχήμα 5.2: Αντικατοπτρική περιστροφή διανύσματος \mathbf{z}_t ως προς ένα υπερεπίπεδο με χρήση του ορθοκανονικού πίνακα Householder.

Ο πίνακας αυτός \mathbf{H} είναι γνωστός ως ορθοκανονικός πίνακας Householder και μπορεί να εκτιμηθεί σύμφωνα με την Εξ. (5.14):

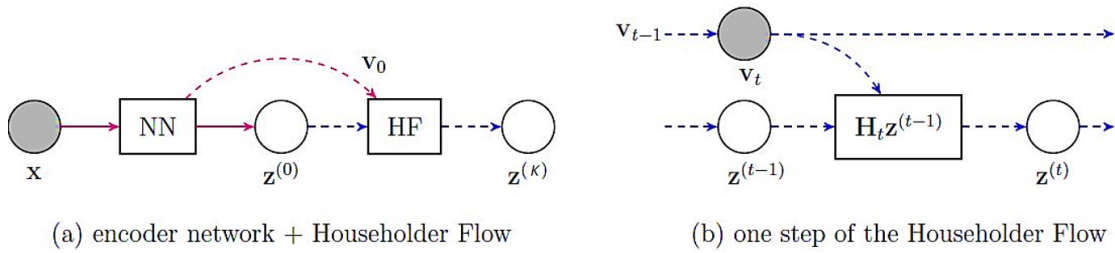
$$\begin{aligned} \mathbf{z}_{t+1} &= \mathbf{z}_t - 2(\mathbf{v}_t^T \mathbf{z}_t) \mathbf{v}_t \longrightarrow \\ \mathbf{z}_{t+1} &= \mathbf{z}_t - 2\mathbf{v}_t \mathbf{v}_t^T \mathbf{z}_t \longrightarrow \\ \mathbf{z}_{t+1} &= (\mathbf{I} - 2\mathbf{v}_t \mathbf{v}_t^T) \mathbf{z}_t \end{aligned} \quad (5.14)$$

Θα πρέπει να σημειωθεί ότι η Εξ. (5.14) ισχύει στις περιπτώσεις όπου το διάνυσμα \mathbf{v}_t είναι μοναδιαίο. Σε διαφορετική περίπτωση, η παραπάνω εξίσωση τροποποιείται αναλόγως:

$$\mathbf{z}_{t+1} = \underbrace{\left(\mathbf{I} - 2 \frac{\mathbf{v}_t \mathbf{v}_t^T}{\|\mathbf{v}_t\|^2} \right)}_{\text{Householder}} \mathbf{z}_t \quad (5.15)$$

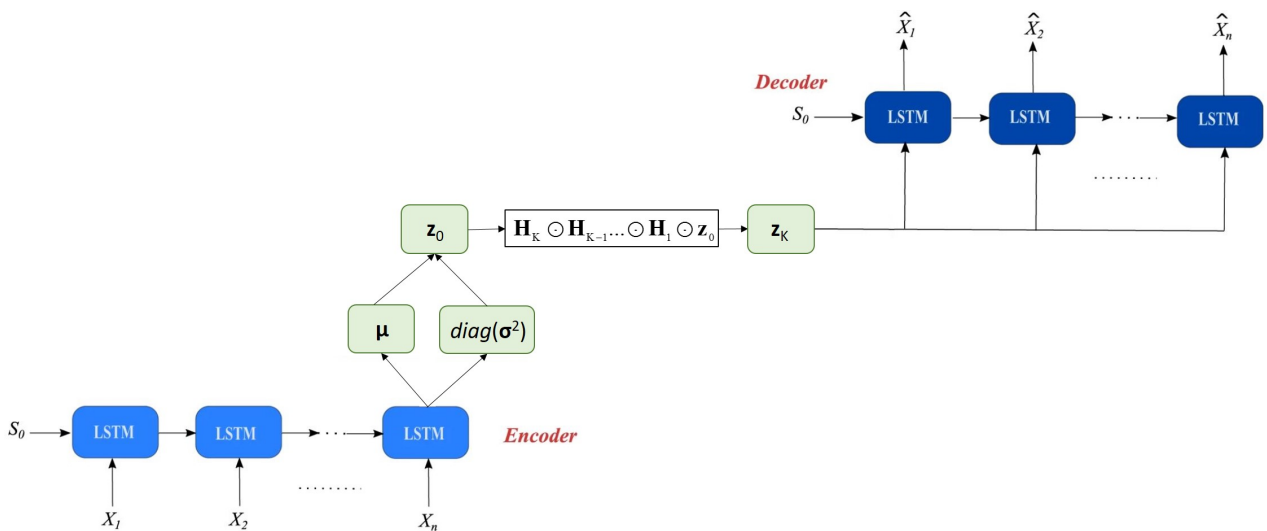
Σύμφωνα με την Εξ. (5.15) έπεται ότι για την εκτίμηση των πινάκων \mathbf{H}_t (όπου $t = 1, \dots, K$) απαιτείται πρώτα η εκτίμηση των κατάλληλων διανυσμάτων \mathbf{v}_t . Για το λόγο αυτό, τα διανύσματα αυτά παράγονται από το τελευταίο επίπεδο του κωδικοποιητή με χρήση ενός γραμμικού μετασχηματισμού, όπως αναπαριστάται στο Σχ. 5.3. Συνεπώς, τα διανύσματα αυτά εκπαιδεύονται ως βάρη του δικτύου, έτσι ώστε να προκύψουν τα τελικά δεδομένα \mathbf{z}_K από μία κατανομή με ολοκληρωμένο πίνακα συνδιασπορών.

Όσον αφορά το πλήθος K των Householder μετασχηματισμών, θα πρέπει να τονισθεί ότι η τιμή αυτή αποτελεί μία υπερπαραμέτρο προς διερεύνηση, με στόχο τον καλύτερο μετασχηματισμό των δεδομένων \mathbf{z}_0 . Επιπλέον, εφόσον οι Householder πίνακες \mathbf{H}_t είναι ορθοκανονικοί, η ορίζουσα των πινάκων αυτών ισούται με 1 και κατά συνέπεια ο λογάριθμος της ποσότητας αυτής μηδενίζεται στην Εξ. (5.9). Κατά συνέπεια, το υπολογιστικό κόστος της συγκεκριμένης ροής είναι αρκετά μικρότερο συγκριτικά με άλλα είδη ροών που έχουν προταθεί.



Σχήμα 5.3: Σχηματική αναπαράσταση της ροής Householder σε συνδυασμό με τον αυτοκωδικοποιητή VAE. Πηγή [61].

Τέλος, εφόσον στη συγκεκριμένη εργασία μελετώνται χρονοσειρές, τα νευρωνικά δίκτυα του κωδικοποιητή και του αποκωδικοποιητή επιλέγονται να είναι δίκτυα LSTM (δες Σχ. 5.4). Συνεπώς, το μοντέλο αυτό θα αναφέρεται από εδώ και στο εξής ως HF-V-LSTM-AE, ή ολογράφως Householder Flow Variational LSTM AutoEncoder.



Σχήμα 5.4: Αρχιτεκτονική variational αυτοκωδικοποιητή σε συνδυασμό με τα νευρωνικά δίκτυα LSTM και την κανονικοποιημένη ροή Householder.

Κεφάλαιο 6

Περιγραφή Πειραμάτων

Στο παρόν Κεφάλαιο παρουσιάζονται αναλυτικά τα πειράματα που πραγματοποιήθηκαν από τους 2 χειριστές στο επίπεδο παραγωγής. Συγκεκριμένα, στην Εν. 6.1 πραγματοποιείται αρχικά μία σύντομη αναφορά στον εξοπλισμό που απαιτείται κατά τη διεξαγωγή των πειραμάτων και στη συνέχεια αναλύονται οι 5 κινήσεις που επέλεγονται να μελετηθούν. Στην Εν. 6.2 παρουσιάζονται αναλυτικά τα βήματα που εκτελούνται κατά τη διάρκεια των κινήσεων, διασπώντας ουσιαστικά κάθε μία από αυτές σε επιμέρους συνιστώσες κινήσεις. Τέλος, στην Εν. 6.3 πραγματοποιείται μία σύντομη μη παραμετρική ανάλυση των σημάτων, επισημαίνοντας την αβεβαιότητα που προκύπτει λόγω της μεγάλης επαναληψιμότητας των πειραμάτων από τους 2 χειριστές.

6.1 Παρουσίαση Κινήσεων Ενδιαφέροντος

Όπως αναφέρθηκε στην Εν. 1.3, η παρούσα εργασία μελετά 5 πιθανές κινήσεις ενός χειριστή στο επίπεδο παραγωγής, έτσι ώστε στη συνέχεια να μπορεί να επιτευχθεί η συνεργασία του με ένα ρομποτικό μηχανισμό. Η καταγραφή των σημάτων αυτών πραγματοποιείται με χρήση της εφαρμογής Science Journal της εταιρίας Google, στην οποία μπορεί οποιοσδήποτε να αποκτήσει πρόσβαση με χρήση ενός smartphone. Εναλλακτικά θα μπορούσε κάποιος να χρησιμοποιήσει οποιαδήποτε άλλη συσκευή καταγραφής σημάτων, καθώς η μεθοδολογία που θα παρουσιαστεί στη συνέχεια δε διαφοροποιείται καθόλου για τις περιπτώσεις όπου η συνεργασία χειριστή-ρομπότ πραγματοποιείται με χρήση σημάτων από αισθητήρες.

Η εφαρμογή Science Journal παρέχει τη δυνατότητα καταγραφής 8 διαφορετικών σημάτων από το σημείο του σώματος στο οποίο προσδένεται το smartphone. Τα σήματα αυτά συνοψίζονται ακολούθως:

1. Επιτάχυνση σε κάθε έναν από τους άξονες x,y,z (σε m/s^2) (3 σήματα),
2. Συνολική επιτάχυνση (linear acceleration) και στους 3 άξονες μαζί, χωρίς να συμπεριλαμβάνεται η δύναμη της βαρύτητας (ξανά σε m/s^2) (4ο σήμα),
3. Φωτεινότητα (σε Exposure Value (EV) λόγω iPhone), όπου ουσιαστικά μετράται η φωτεινότητα της μπροστινής κάμερας του κινητού τηλεφώνου (5ο σήμα),
4. Δύναμη του γεωμαγνητικού πεδίου (σε MicroTesla- μT), όπου ουσιαστικά χρησιμοποιούνται αισθητήρες για την καταγραφή του μαγνητικού πεδίου στους 3 άξονες και στη συνέχεια συγχωνεύονται για την εκτίμηση της συνολικής δύναμης του μαγνητικού πεδίου (6ο σήμα),
5. Προσανατολισμός με χρήση πυξίδας (σε degrees) (7ο σήμα),
6. Ένταση του ηχητικού κύματος (σε dB) (8ο σήμα)

Το smartphone επιλέγεται να προσδεθεί στον καρπό 2 διαφορετικών χειριστών, οι οποίοι στη συνέχεια πραγματοποιούν ένα μεγάλο αριθμό πειραμάτων από 5 διαφορετικές δραστηριότητες. Οι δραστηριότητες είναι πολύ δύσκολα ανιχνεύσιμες, καθώς δε διαφέρουν αισθητά από συνήθειες κινήσεις μη ενδιαφέροντος που μπορεί να εκτελεί ένας εργαζόμενος. Πιο συγκεκριμένα, οι 5 κινήσεις που επιλέγονται να μελετηθούν είναι οι εξής:

Πίνακας 6.1: Εξεταζόμενες κινήσεις χειριστή στο επίπεδο παραγωγής.

| A/A | Κίνηση | Συντομογραφία |
|-----|--------------------------------------------|---------------|
| 1 | Βίδωμα κοχλία με χρήση γερμανικού κλειδιού | ΓΚ |
| 2 | Βίδωμα κοχλία με χρήση κατσαβιδιού | ΚΚ |
| 3 | Σήκωμα τεμαχίου | ΣΤ |
| 4 | Εναπόθεση τεμαχίου | ΕΤ |
| 5 | Καθαρισμός επιφάνειας | ΚΕ |

Να σημειωθεί ότι η συχνότητα δειγματοληψίας της εφαρμογής Science Journal ανέρχεται στα 15 Hz, η οποία κρίνεται ικανοποιητική για τη μελέτη των 5 προαναφερθέντων κινήσεων. Βέβαια, σε περίπτωση μελέτης κινήσεων οι οποίες είναι πιο πολύπλοκες και περιλαμβάνουν αρκετά απότομες μεταβολές των χειριών των εργαζομένων, ίσως απαιτείται η καταγραφή σημάτων με μεγαλύτερη συχνότητα δειγματοληψίας. Ακολούθως λοιπόν παρουσιάζονται τα βήματα που ακολουθούνται για την εκτέλεση κάθε μίας από τις 5 εξεταζόμενες κινήσεις.

6.2 Βήματα Εκτέλεσης Κινήσεων

Ύστερα από τον καθορισμό των 5 εξεταζόμενων κινήσεων, στην ενότητα αυτή παρουσιάζονται αναλυτικά τα βήματα που ακολουθούνται για την εκτέλεσή τους. Αρχικά λοιπόν, πριν από τη διενέργεια των πειραμάτων τα smartphone προσαρτώνται στους καρπούς 2 χειριστών, όπως απεικονίζεται στο Σχ. 6.1. Βέβαια, η θέση πρόσδεσης των συσκευών καταγραφής των σημάτων επαφίεται στη διευκόλυνση του εκάστοτε εργαζόμενου να εκτελέσει τις ζητούμενες κινήσεις. Παρόλα αυτά, στην παρούσα εργασία επιλέγεται ο καρπός και στους 2 χειριστές, καθώς αφενός μεν δε δυσχεραίνεται η εκτέλεση των πειραμάτων και αφετέρου τα σήματα που λαμβάνονται παρουσιάζουν μεγαλύτερες μεταβολές σε χαμηλά σημεία του χεριού.



Σχήμα 6.1: Προσάρτηση smartphone στον καρπό των 2 χειριστών για τη διεξαγωγή των 5 κινήσεων.

Ο συνολικός αριθμός των πειραμάτων που διενεργούνται από τους 2 χειριστές στο επίπεδο παραγωγής ανέρχεται στα 625, δηλαδή 125 πειράματα για κάθε μία από τις 5 κινήσεις. Τα αναλυτικά βήματα εκτέλεσης των κινήσεων αυτών παρουσιάζονται συνοπτικά ακολούθως:

1. Βιδώμα με χρήση γερμανικού κλειδιού (ή ΓΚ)

Σε αυτή την περίπτωση η καταγραφή σημάτων ξεκινά με τη μεταφορά του κοχλίου στην οπή και ολοκληρώνεται με τη διαδικασία του βιδώματος με χρήση γερμανικού κλειδιού. Η συνολική διάρκεια της κίνησης ανέρχεται στα 8-10 secs. Η κίνηση του βιδώματος που εκτελεί ο κάθε χειριστής μπορεί να χωριστεί στις 4 φάσεις (συνιστώσες κινήσεις) που αναγράφονται ακολούθως και παρουσιάζονται στο Σχ. 6.2:

1. Πιάσιμο και σήκωμα κοχλίας
2. Μεταφορά κοχλίας πάνω από την οπή
3. Τοποθέτηση βίδας στην οπή
4. Βιδώμα κοχλίας με γερμανικό κλειδί

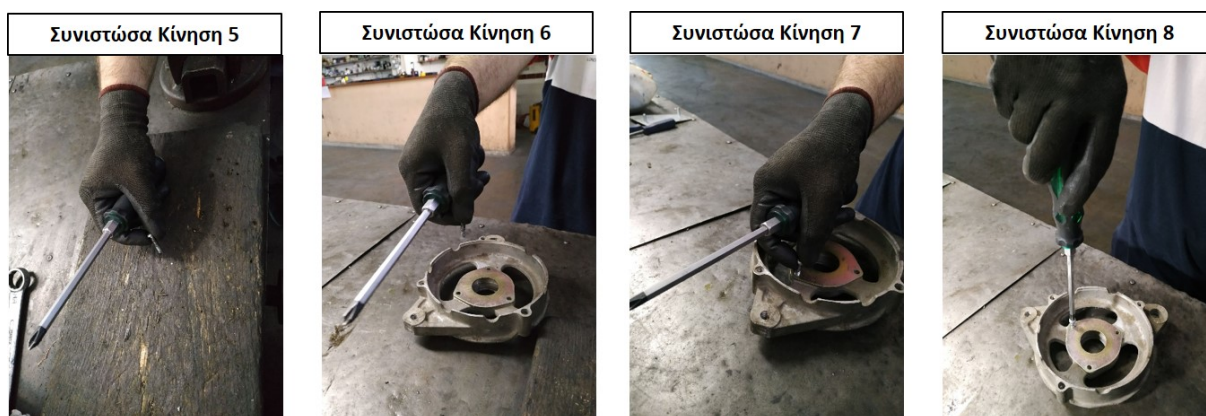


Σχήμα 6.2: Παρουσίαση 4 φάσεων κατά την εκτέλεση του βιδώματος με χρήση γερμανικού κλειδιού.

2. Βιδώμα κοχλία με χρήση κατσαβιδιού (ή ΚΚ)

Η κίνηση αυτή είναι σχεδόν ίδια με την προηγούμενη. Συγκεκριμένα, η καταγραφή σημάτων ξεκινά με τη μεταφορά του κοχλία στην οπή και ολοκληρώνεται με τη διαδικασία του βιδώματος με χρήση κατσαβιδιού. Η συνολική διάρκεια της κίνησης ανέρχεται ξανά περίπου στα 8 έως και 10 secs. Συνεπώς, η μόνη διαφορά με την κίνηση του γερμανικού κλειδιού έγκειται στον τρόπο βιδώματος του κοχλία. Αυτό συνεπάγεται ότι ο κάθε χειριστής εκτελεί τις 4 συνιστώσες κινήσεις που αναγράφονται ακολούθως (συνεχίζοντας την αρίθμηση από το τέλος των προηγούμενων συνιστωσών κινήσεων) και παρουσιάζονται στο Σχ. 6.3:

- | | |
|-------------------------------------|---------------------------------------|
| 5. Πιάσιμο και σήκωμα κοχλία | 7. Τοποθέτηση βίδας στην οπή |
| 6. Μεταφορά κοχλία πάνω από την οπή | 8. Βιδώμα κοχλία με χρήση κατσαβιδιού |



Σχήμα 6.3: Παρουσίαση 4 φάσεων κατά την εκτέλεση του βιδώματος με χρήση κατσαβιδιού.

3. Σήκωμα τεμαχίου (ή ΣΤ)

Η συγκεκριμένη κίνηση περιλαμβάνει το σήκωμα ενός τεμαχίου από 3 διαφορετικούς πάγκους, κάθε ένας από τους οποίους έχει διαφορετικό ύψος και προσανατολισμό στο χώρο. Σε κάθε μία από τις 3 αυτές περιπτώσεις συλλέχθηκαν λίγα περισσότερα από 40 πειράματα, ώστε εν τέλει να προκύψουν τα συνολικά 125 της συγκεκριμένη κίνησης. Η διάρκειά της είναι μικρότερη από τις 2 προηγούμενες περιπτώσεις, με μέσο όρο τα 5.5 secs. Η συγκεκριμένη κίνηση μπορεί να χωριστεί στις 3 φάσεις μικρότερης διάρκειας που αναγράφονται και παρουσιάζονται στο Σχ. 6.4:

9. Σήκωμα τεμαχίου από τον πάγκο με τεντωμένα χέρια
10. Μάζεμα χεριών μεταφέροντας το τεμάχιο κοντά στο σώμα
11. Περιστροφή σώματος προς οποιαδήποτε κατεύθυνση για μεταφορά



Σχήμα 6.4: Παρουσίαση 3 φάσεων κατά την εκτέλεση του σηκώματος και μεταφοράς ενός τεμαχίου.

4. Εναπόθεση Τεμαχίου (ή ΕΤ)

Η συγκεκριμένη κίνηση ουσιαστικά αποτελεί την ακριβώς αντίθετη διαδικασία από το σήκωμα τεμαχίου. Τα πειράματα διεξήχθησαν στους ίδιους πάγκους, με τη διαφορά να έγκειται στην κίνηση των χεριών για την εναπόθεση (αντί για σήκωμα) του τεμαχίου. Σε κάθε μία από τις 3 αυτές περιπτώσεις συλλέχθηκαν ξανά λίγα περισσότερα από 40 πειράματα, ώστε εν τέλει να προκύψουν τα συνολικά 125 της συγκεκριμένη κίνησης. Επιπλέον, και σε αυτή την περίπτωση η μέση διάρκεια κάθε πειράματος ανέρχεται στα 5.5 secs. Τέλος, οι αντίστοιχες 3 συνιστώσες κινήσεις στις οποίες μπορεί να χωρισθεί η συγκεκριμένη δραστηριότητα αναγράφονται ακολουθώντας και ταυτόχρονα παρουσιάζονται στο Σχ. 6.4:

12. Διατήρηση τεμαχίου με τεντωμένα χέρια σε υψηλή θέση
13. Περιστροφή σώματος κρατώντας το τεμάχιο με μαζεμένα χέρια
14. Εναπόθεση τεμαχίου σε πάγκο τεντώνοντας τα χέρια



Σχήμα 6.5: Παρουσίαση 3 φάσεων κατά την εκτέλεση της εναπόθεσης ενός τεμαχίου.

5. Καθαρισμός Επιφάνειας (ή ΚΕ)

Τέλος, η συγκεκριμένη κίνηση περιλαμβάνει τον καθαρισμό 3 διαφορετικών επιφανειών (διαφορετικού ύψους και προσανατολισμού στο χώρο) με 3 διαφορετικούς τρόπους. Για κάθε έναν τρόπο εκτελέστηκαν περίπου 14 πειράματα σε κάθε επιφάνεια (δηλαδή συνολικά 42 πειράματα με τον ίδιο τρόπο και στις 3 επιφάνειες μαζί) με αποτέλεσμα και για τους 3 τρόπους να συλλεχθούν συνολικά 125 πειράματα. Η μέση διάρκεια κάθε πειράματος ανέρχεται στα 6 secs. Παραδείγματα των 3 τρόπων καθαρισμού σε μία από τις επιφάνειες απεικονίζονται στο Σχ. 6.6:



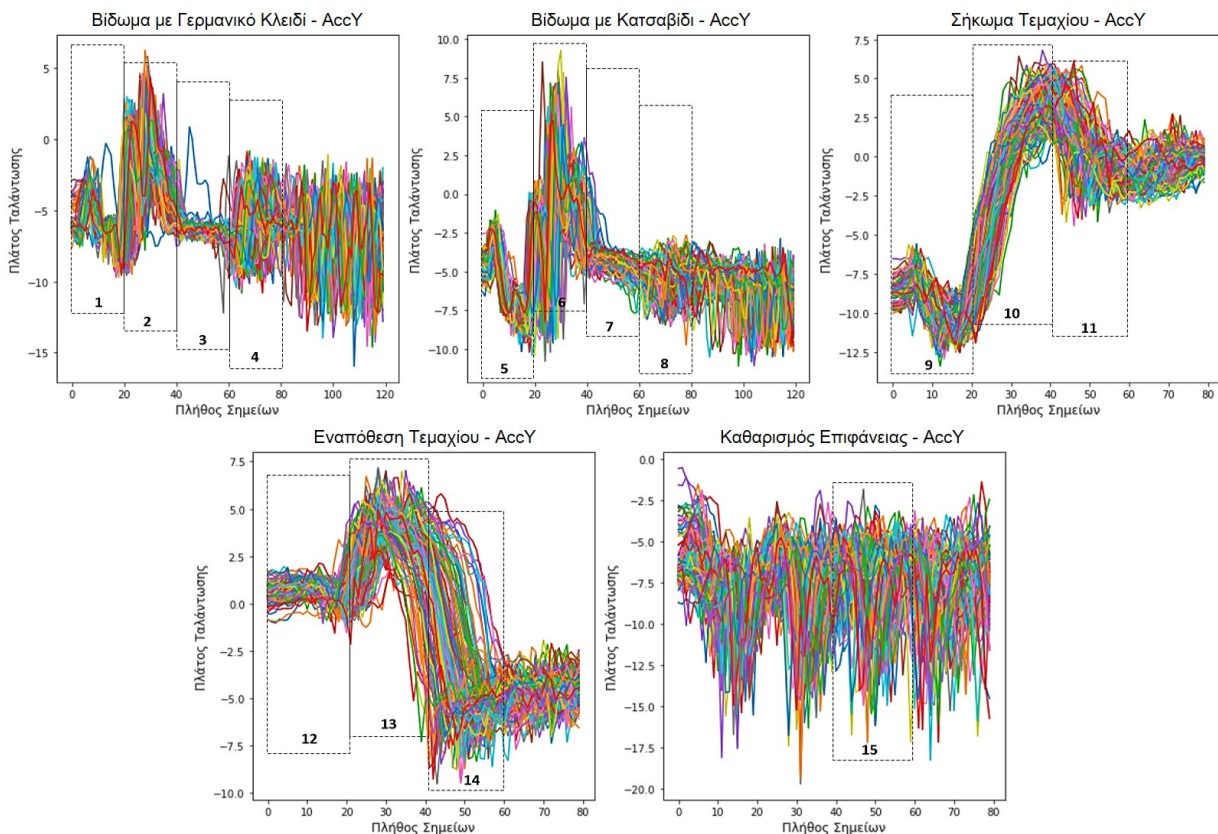
Σχήμα 6.6: Παραδείγματα πειραμάτων καθαρισμού επιφάνειας.

6.3 Μη Παραμετρική Ανάλυση Σημάτων

Ύστερα από τη συλλογή των 625 σημάτων από τις 5 εξεταζόμενες κινήσεις, στην ενότητα αυτή πραγματοποιείται μία μη παραμετρική ανάλυση των χρονοσειρών αυτών. Συγκεκριμένα, στο Σχ. 6.7 απεικονίζεται ενδεικτικά η επιτάχυνση στον άξονα Y για τα 125 πειράματα κάθε μίας κίνησης, όπου είναι εμφανής η αβεβαιότητα (uncertainty) που προκύπτει λόγω της επαναληψιμότητας των πειραμάτων. Να σημειωθεί ότι η επιλογή της επιτάχυνσης Y είναι ενδεικτική, ενώ αντίστοιχα αποτελέσματα παρατηρούνται και για τις υπόλοιπες 7 καταγραφές σημάτων.

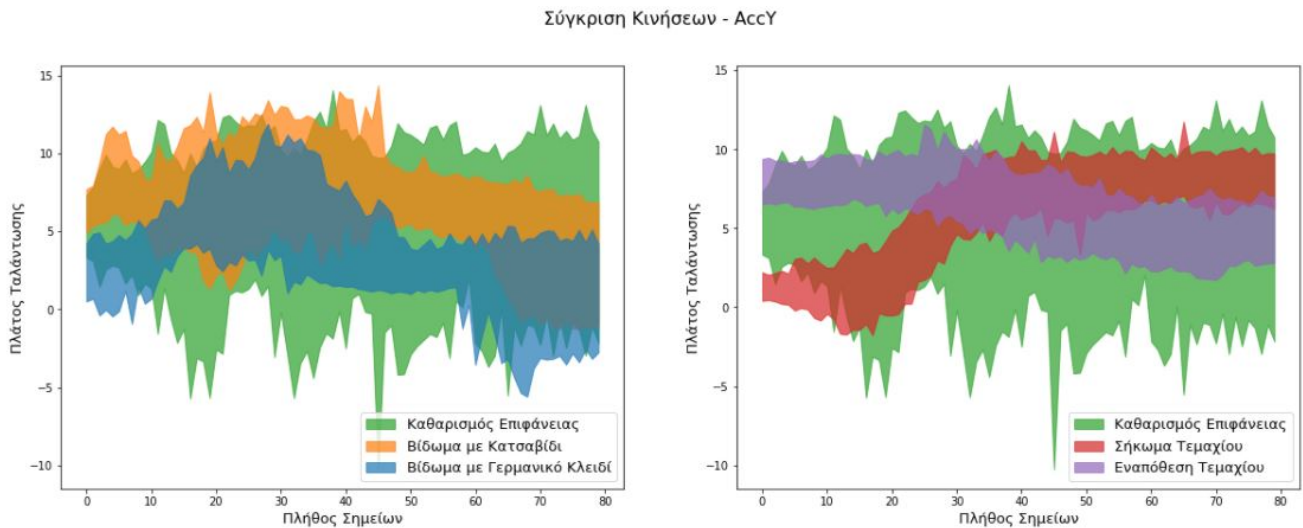
Όσον αφορά τη χρονική διάρκεια των 5 εξεταζόμενων κινήσεων, είναι προφανές ότι αυτές διαφέρουν. Συγκεκριμένα, ενώ οι 2 πρώτες κινήσεις βιδώματος με γερμανικό κλειδί και κατσαβίδι αποτελούνται από περίπου 120 περίπου σημεία (δηλαδή 8 secs, αφού η συχνότητα δειγματοληψίας είναι 15 Hz), οι υπόλοιπες 3 κινήσεις αποτελούνται από περίπου 80 σημεία, δηλαδή 5.5 secs. Η διαφορά αυτή της χρονικής διάρκειας μπορεί να επιλυθεί αγνοώντας τα τελευταία δευτερόλεπτα των 2 πρώτων κινήσεων, στα οποία η διαδικασία του βιδώματος επαναλαμβάνεται περιστρέφοντας τον κοχλία με γερμανικό κλειδί ή κατσαβίδι χωρίς να προσδίδεται επιπλέον πληροφορία.

Επιπλέον, με χρήση κατάλληλων παραθύρων (όρια με διακεκομμένες γραμμές) στο Σχ. 6.7 απαριθμούνται οι 15 φάσεις που μπορούν να αναλυθούν οι 5 ολόκληρες κινήσεις, όπως παρουσιάστηκαν αναλυτικά στην Εν. 6.2. Ο μέσος χρόνος διάρκειας των συνιστωσών κινήσεων ανέρχεται περίπου στα 1.5-2 secs για όλες τις κινήσεις, τα οποία αντιστοιχούν σε τουλάχιστον 20 σημεία στα σήματα καταγραφής. Βέβαια, να σημειωθεί ότι για τις 2 πρώτες κινήσεις του βιδώματος, οι συνιστώσες 4 και 8 διαρκούν αρκετά περισσότερο έως ότου ολοκληρωθεί το βίδωμα. Εφόσον όμως οι συγκεκριμένες φάσεις επαναλαμβάνονται χωρίς μεταβολές στο πλάτος ταλάντωσης, τα 20 πρώτα σημεία κατά την έναρξη του βιδώματος του κοχλία θεωρούνται ικανοποιητικά για να αντιπροσωπεύσουν τις συνιστώσες αυτές. Έτσι, ο συνολικός αριθμός σημάτων φάσεων ανέρχεται στα 1875 , δηλαδή $(4 + 4) \times 125 = 1000$ από τις κινήσεις ΓΚ και ΚΚ, $(3 + 3) \times 125 = 750$ από τις κινήσεις ΣΤ και ΕΤ και τέλος 125 από την κίνηση ΚΕ.



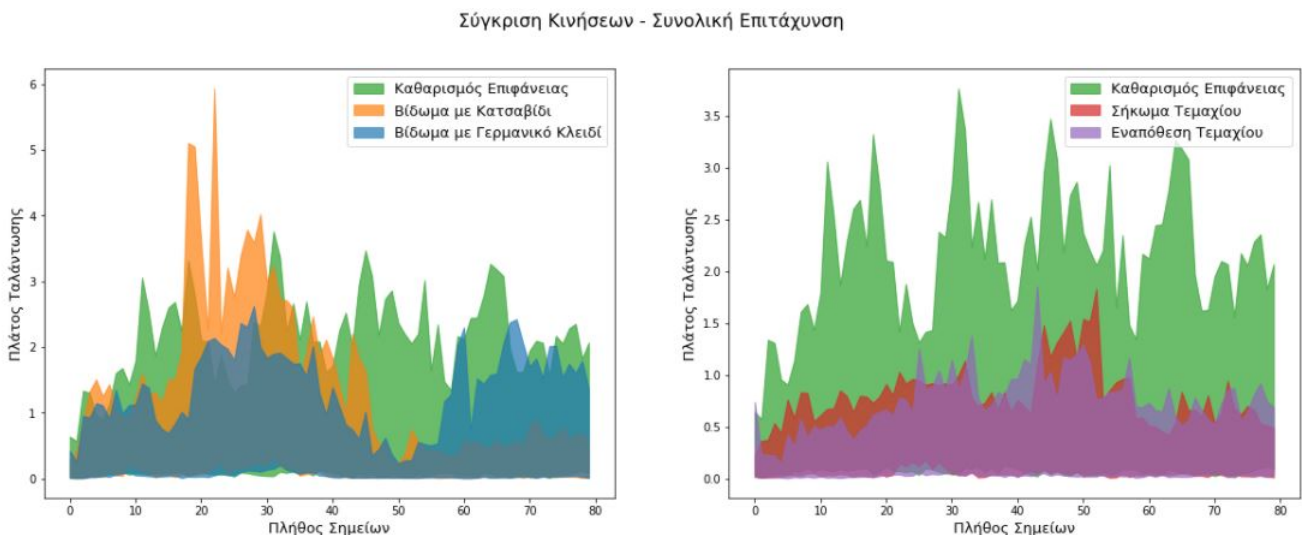
Σχήμα 6.7: Σήματα επιτάχυνσης του άξονα Y για κάθε μία από τις 5 εξεταζόμενες κινήσεις.

Ακολούθως, στο Σχ. 6.8 απεικονίζεται το εύρος τιμών του πλάτους ταλάντωσης της επιτάχυνσης Y για όλη τη διάρκεια της κάθε κίνησης. Σύμφωνα λοιπόν με αποτελέσματα που προκύπτουν είναι προφανές ότι το εύρος τιμών του καθαρισμού επιφάνειας φαίνεται να επικαλύπτει ως επί το πλείω τα εύρη τιμών των υπολοίπων κινήσεων. Το φαινόμενο αυτό είναι πιο έντονο στις κινήσεις σηκώματος και εναπόθεσης τεμαχίου (δεξιά), όπου οι τιμές της επιτάχυνσης Y των 2 αυτών κινήσεων φαίνεται να επικαλύπτονται σχεδόν εξ' ολοκλήρου από τις αντίστοιχες του καθαρισμού επιφάνειας. Τέλος, όπως επαληθεύεται και από το Σχ. 6.7, οι 2 πρώτες κινήσεις βιδώματος του κοχλίου με γερμανικό κλειδί και κατσαβίδι φαίνεται να παρουσιάζουν παρόμοια μορφή της επιτάχυνσης Y , με αποτέλεσμα την επικάλυψη των μεταξύ τους τιμών σε αρκετά μεγάλο εύρος της συνολικής κίνησης.



Σχήμα 6.8: Σύγκριση του εύρους τιμών της επιτάχυνσης Y για τις 5 κινήσεις.

Αντίστοιχα με την επιτάχυνση Y , στο Σχ. 6.9 παρουσιάζεται το εύρος τιμών της συνολικής επιτάχυνσης για τα 625 σήματα κινήσεων. Όπως παρατηρείται λοιπόν και σε αυτήν την περίπτωση υπάρχει σημαντική αλληλεπικάλυψη των τιμών των διαφόρων κινήσεων, γεγονός που αποδεικνύει τη δυσκολία διαχωρισμού των εξεταζόμενων δραστηριοτήτων. Στόχος λοιπόν από εδώ και στο εξής είναι να επιστρατευθούν οι τεχνικές βαθιάς μηχανικής μάθησης των Κεφαλαίων 2, 3, 4 και 5, έτσι ώστε να εξετασθεί σε τί βαθμό οι κινήσεις αυτές μπορούν να αναγνωρισθούν.



Σχήμα 6.9: Σύγκριση του εύρους τιμών της συνολικής επιτάχυνσης για τις 5 κινήσεις.

Τέλος, εκτός από το πρόβλημα ταξινόμησης των 5 αυτών δραστηριοτήτων, στο Κεφάλαιο που ακολουθεί μελετάται επιπλέον το πρόβλημα κατηγοριοποίησης των 15 συνολικά συνιστωσών κινήσεων. Οι λόγοι της μελέτης αυτής είναι 2 και παρουσιάζονται συνοπτικά ακολούθως:

- Εξέταση της ευαισθησίας των προτεινόμενων δικτύων, δηλαδή εάν τα δίκτυα που επιστρατεύονται μπορούν να εκπαιδευτούν με πολύ λίγα σημεία όπως αυτά των 15 συνιστωσών κινήσεων, ώστε στη συνέχεια να τις ταξινομήσουν κατάλληλα.
- Αν σε κάποιο σήμα μεγάλης διάρκειας και άγνωστης προέλευσης η αναγνώριση τόσο των ολόκληρων κινήσεων όσο και των συνιστωσών κινήσεων μπορούν να συνεισφέρουν ταυτόχρονα στον εντοπισμό των επιθυμητών δραστηριοτήτων εντός του άγνωστου σήματος.

Στο επόμενο Κεφάλαιο λοιπόν παρουσιάζεται η προτεινόμενη μεθοδολογία και εξάγονται τα αποτελέσματα της επεξεργασίας των σημάτων που αντιστοιχούν στις ολόκληρες κινήσεις και στις συνιστώσες κινήσεις, αντίστοιχα.

Κεφάλαιο 7

Παρουσίαση Αποτελεσμάτων

Στο παρόν Κεφάλαιο παρουσιάζονται τα αποτελέσματα της ανάλυσης των σημάτων τα οποία συλλέχθηκαν από τους 2 χειριστές κατά τη διάρκεια των πειραμάτων. Συγκεκριμένα, στην Εν. 7.1 αναλύονται τα 625 σήματα των 5 εξεταζόμενων κινήσεων, από τα οποία στην αρχή εξάγονται κατάλληλα χαρακτηριστικά με χρήση αυτοκωδικοποιητών και στη συνέχεια πραγματοποιείται ταξινόμησή τους χρησιμοποιώντας τα δίκτυα LSTM. Ακολούθως, στην Εν. 7.2 πραγματοποιείται η ίδια ακριβώς διαδικασία για τα 1875 σήματα των 15 συνιστωσών κινήσεων, τα οποία προκύπτουν από τη διάσπαση των αρχικών σημάτων σε επιμέρους χρονοσειρές. Τέλος, στην Εν. 7.3 μελετώνται τα σήματα μακράς διάρκειας, τα οποία αποτελούν χρονοσειρές πολύ μεγαλύτερης διάρκειας από αυτές των κινήσεων και επιπλέον περιλαμβάνουν τμήματα σημάτων από άλλες κινήσεις μη ενδιαφέροντος. Για το λόγο αυτό προτείνεται μία νέα μεθοδολογία, η οποία συνδυάζει τα νευρωνικά δίκτυα των Εν. 7.1 και 7.2 που έχουν εκπαιδευτεί με σκοπό την αναγνώριση των επιθυμητών κινήσεων εντός των σημάτων μακράς διάρκειας.

7.1 Αποτελέσματα από Σήματα Κινήσεων

Στην παρούσα ενότητα ουσιαστικά παρουσιάζονται τα αποτελέσματα που προκύπτουν από την ανάλυση των 625 σημάτων των 5 εξεταζόμενων κινήσεων που περιγράφηκαν στο Κεφάλαιο 6. Τα βήματα που ακολουθούνται για την εξαγωγή των αποτελεσμάτων αυτών παρουσιάζονται συνοπτικά ακολούθως:

1. Εκπαίδευση νευρωνικών δικτύων αυτοκωδικοποιητών στα 625 σήματα των 5 κινήσεων, χρησιμοποιώντας όλα τα δυνατά μοντέλα αυτοκωδικοποιητών που παρουσιάστηκαν στα Κεφάλαια 3, 4 και 5. Στόχος της διαδικασίας αυτής είναι να βρεθεί το μοντέλο αυτοκωδικοποιητή που εξάγει τα βέλτιστα χαρακτηριστικά μέσα από το φιλτράρισμα των σημάτων αυτών, έτσι ώστε αυτά στη συνέχεια να ενσωματωθούν στα μοντέλα του βήματος 2.
2. Εκπαίδευση νευρωνικών δικτύων LSTM για την ταξινόμηση των σημάτων στις 5 εξεταζόμενες κινήσεις. Να σημειωθεί ότι στο τελευταίο επίπεδο των νευρωνικών αυτών προστίθενται τα χαρακτηριστικά των σημάτων του προηγούμενου βήματος, με σκοπό την προσθήκη επιπλέον πληροφορίας κατά τη διαδικασία ταξινόμησης.

Για το λόγο αυτό, στην υποενότητα 7.1.1 μελετώνται όλα τα διαφορετικά μοντέλα αυτοκωδικοποιητών και επιλέγεται εκείνο που βελτιστοποιεί τη διαδικασία συμπίεσης των εξεταζόμενων σημάτων. Ακολούθως, στην υποενότητα 7.1.2 συνδυάζονται τα χαρακτηριστικά αυτά με τα νευρωνικά δίκτυα LSTM με σκοπό την ταξινόμηση των 625 σημάτων που προέκυψαν από τα πειράματα των 5 κινήσεων.

Η επεξεργασία των σημάτων πραγματοποιήθηκε με χρήση μίας GPU Tesla T4/NVIDIA-SMI 418.67 με 15 Gb RAM. Η διαδικασία εκπαίδευσης των νευρωνικών δικτύων απαιτεί το διαχωρισμό των σημάτων σε σύνολο εκπαίδευσης (training set), επικύρωσης (validation set) και τέλος ελέγχου (test set). Τα δεδομένα λοιπόν διαχωρίστηκαν ως εξής:

- Training Set: χρησιμοποιήθηκε το 80% του συνόλου δεδομένων. Συνεπώς, από τα 625 σήματα κινήσεων τα 500 χρησιμοποιήθηκαν για την εκπαίδευση των μοντέλων.
- Validation Set: χρησιμοποιήθηκε το 8% του συνόλου δεδομένων, δηλαδή 50 σήματα για τη διαδικασία επικύρωσης.
- Test Set: χρησιμοποιήθηκε το 12% του συνόλου δεδομένων, δηλαδή 70 σήματα για τη διαδικασία ελέγχου.

Θα πρέπει να τονισθεί ότι ο προαναφερθέν διαχωρισμός των σημάτων ισχύει τόσο για την υποενότητα 7.1.1 των αυτοκωδικοποιητών που ακολουθεί, όσο και για την υποενότητα 7.1.2 της ταξινόμησης των κινήσεων με τη βοήθεια των δικτύων LSTM.

7.1.1 Εξαγωγή Χαρακτηριστικών από Σήματα Κινήσεων

Στην παρούσα υποενότητα παρουσιάζονται τα αποτελέσματα όλων των αυτοκωδικοποιητών που εξηγήθηκαν στα Κεφάλαια 3, 4 και 5, έτσι ώστε να διερευνηθεί ποιά από τις εξεταζόμενες αρχιτεκτονικές οδηγεί στην αποτελεσματικότερη συμπίεση των σημάτων που προκύπτουν από τα 625 πειράματα των 5 κινήσεων. Αρχικά λοιπόν θα πρέπει να επιλεχθούν ορισμένες υπερπαραμέτροι των δικτύων, ώστε στη συνέχεια να πραγματοποιηθεί η εκπαίδευσή τους. Βέβαια, οι υπερπαραμέτροι αυτές θα πρέπει να είναι κοινές για όλα τα είδη των αυτοκωδικοποιητών, με σκοπό την τίμια σύγκριση μεταξύ τους. Ορισμένες λοιπόν από τις υπερπαραμέτρους αυτές συνοψίζονται ακολούθως:

1. Αριθμός Επιπέδων Δικτύου: Η συγκεκριμένη υπερπαράμετρος αναφέρεται στον αριθμό των επιπέδων (layers) σε κάθε ένα από τα δίκτυα κωδικοποιητή και αποκωδικοποιητή. Συγκεκριμένα, εάν πρόκειται για δίκτυα LSTM η υπερπαράμετρος αυτή αναφέρεται στο πλήθος των δικτύων αυτών που είναι προσκολλημένα (stacked) το ένα πάνω στο άλλο. Αντιθέτως, αν πρόκειται για δίκτυα MLP η παράμετρος αυτή αναφέρεται προφανώς στο πλήθος των layers που αποτελείται το δίκτυο.
2. Νευρώνες ανά Επίπεδο: Η συγκεκριμένη υπερπαράμετρος αναφέρεται στο πλήθος των νευρώνων στο τμήμα κωδικοποιητή και αποκωδικοποιητή. Συγκεκριμένα, αν το δίκτυο που εξετάζεται είναι δίκτυο LSTM, τότε η υπερπαράμετρος αυτή αναφέρεται στη διάσταση του διανύσματος \mathbf{h} που παρουσιάστηκε στην υποενότητα 2.2.1. Αντιθέτως, για το τμήμα του αποκωδικοποιητή των δικτύων SAV-LSTM/MLP-AE και VL-LSTM/MLP-AE, η υπερπαράμετρος αυτή αναφέρεται στο πλήθος των νευρώνων κάθε επιπέδου του δικτύου MLP.
3. Διάσταση Δεδομένων \mathbf{z} : Η συγκεκριμένη υπερπαράμετρος αναφέρεται προφανώς στο πλήθος των νευρώνων που επιλέγονται στο μεσσαίο επίπεδο (bottleneck). Εφόσον κάθε σήμα κινήσεων αποτελείται από 80 σημεία 8 διαφορετικών αισθητήρων, έπεται ότι ο συνολικός αριθμός σημείων κάθε σήματος ανέρχεται στα 640. Συνεπώς, για τη συμπίεσμένη αναπαράσταση \mathbf{z} θα πρέπει να επιλεγεί ένας αριθμός μικρότερος του 640.
4. Συντελεστής Εχμάρησης: Η συγκεκριμένη υπερπαράμετρος αναφέρεται στο συντελεστή των μεθόδων καθόδου κλίσης, οι οποίες παρουσιάστηκαν στην Εν. 2.3. Η μέθοδος βελτιστοποίησης η οποία χρησιμοποιείται σε όλα τα δίκτυα που ακολουθούν είναι ο αλγόριθμος Adam. Τέλος, εφόσον η βιβλιοθήκη pytorch [57] εκτελεί μόνο αλγορίθμους καθόδου (κι όχι ανόδου) κλίσης, κατά τη διάρκεια της εκπαίδευσης πραγματοποιείται ελαχιστοποίηση της ποσότητας -ELBO αντί για μεγιστοποίηση της ποσότητας ELBO των Κεφαλαίων 3, 4 και 5.

Σημείωση: Τα δίκτυα SAV-LSTM/MLP-AE και VL-LSTM/MLP-AE χρησιμοποιούν δίκτυα MLP στο τμήμα του αποκωδικοποιητή. Αντιθέτως, το απλό δίκτυο V-LSTM-AE και το HF-V-LSTM-AE χρησιμοποιούν δίκτυα LSTM τόσο στον κωδικοποιητή όσο και στον αποκωδικοποιητή. Για να πραγματοποιηθεί όμως μία τίμια σύγκριση των αποτελεσμάτων, στη συνέχεια παρουσιάζονται τα αποτελέσματα και του απλού δικτύου V-LSTM-AE με δίκτυο MLP στο τμήμα του αποκωδικοποιητή, το οποίο ονομάζεται V-LSTM/MLP-AE. Η διαδικασία αυτή ακολουθείται διότι η χρήση του δικτύου MLP σε χρονοσειρές ίσως μειώνει την απόδοση των μοντέλων και κατά συνέπεια επιθυμούμε να τονίσουμε τη σημασία της χρήσης των δικτύων LSTM.

Η επιλογή των τελικών υπερπαραμέτρων πραγματοποιείται ύστερα από κατάλληλη διερεύνηση των πιθανών τιμών τους. Οι υπερπαράμετροι λοιπόν που επιλέγονται και οδηγούν στις βέλτιστες δυνατές αποδόσεις των αυτοκωδικοποιητών παρουσιάζονται στον Πιν. 7.1:

Πίνακας 7.1: Υπερπαράμετροι αυτοκωδικοποιητών για τη συμπίεση σημάτων των 5 κινήσεων.

| Δίκτυο Κωδικ/τή | Δίκτυο Αποκωδ/τή | Αριθμός Επιπέδων | Νευρώνες \mathbf{h} ανά Επίπεδο | Νευρώνες bottleneck \mathbf{z} | Συν/στής Εχμάρησης |
|-----------------|------------------|------------------|-----------------------------------|----------------------------------|--------------------|
| LSTM | LSTM | 2 | 400 | 250 | $5 \cdot 10^{-4}$ |
| | MLP | | 350 | | |

Σύμφωνα με τον Πιν. 7.1, το πλήθος των νευρώνων \mathbf{z} στο μεσσαίο επίπεδο όλων των εξεταζόμενων αυτοκωδικοποιητών ανέρχεται στην τιμή 250. Συνεπώς, τα 640 σημεία κάθε σήματος συμπίεζονται σε 250 χαρακτηριστικά, μειώνοντας ουσιαστικά το πλήθος των διαστάσεων του σήματος κατά 60%. Ακολούθως, για τη σύγκριση μεταξύ των διάφορων παραλλαγών των αυτοκωδικοποιητών πραγματοποιείται εκτίμηση της τιμής της λογαριθμοποιημένης πιθανοφάνειας στο σύνολο δεδομένων ελέγχου, δηλαδή στα 70 σήματα κινήσεων. Προφανώς, υψηλότερες τιμές

της λογαριθμοποιημένης πιθανοφάνειας των παρατηρούμενων δεδομένων συνεπάγονται ένα μοντέλο που ανακατασκευάζει τα δεδομένα εισόδου με μεγαλύτερη ικανότητα και κατά συνέπεια η εξαγωγή των χαρακτηριστικών στο μεσαίο επίπεδο είναι πιο αποτελεσματική.

Οι τιμές της λογαριθμοποιημένης συνάρτησης πιθανοφάνειας για όλα τα μοντέλα αυτοκωδικοποιητών αναγράφονται στον Πίν. 7.2:

Πίνακας 7.2: Τιμές λογαριθμοποιημένης πιθανοφάνειας για τους αυτοκωδικοποιητές των 5 κινήσεων.

| | V-LSTM/MLP-AE | V-LSTM-AE | SAV-LSTM/MLP-AE |
|--------------------------|---------------|----------------|-----------------|
| Λογαρ/νη Πιθανοφάνεια | 238.90 | 733.07 | 241.19 |
| | HF-V-LSTM-AE | VL-LSTM/MLP-AE | |
| — — | 752.45 | 745.15 | |

Παρατηρώντας τον Πίν. 7.2, ορισμένα συμπεράσματα που μπορούν να εξαχθούν είναι τα εξής:

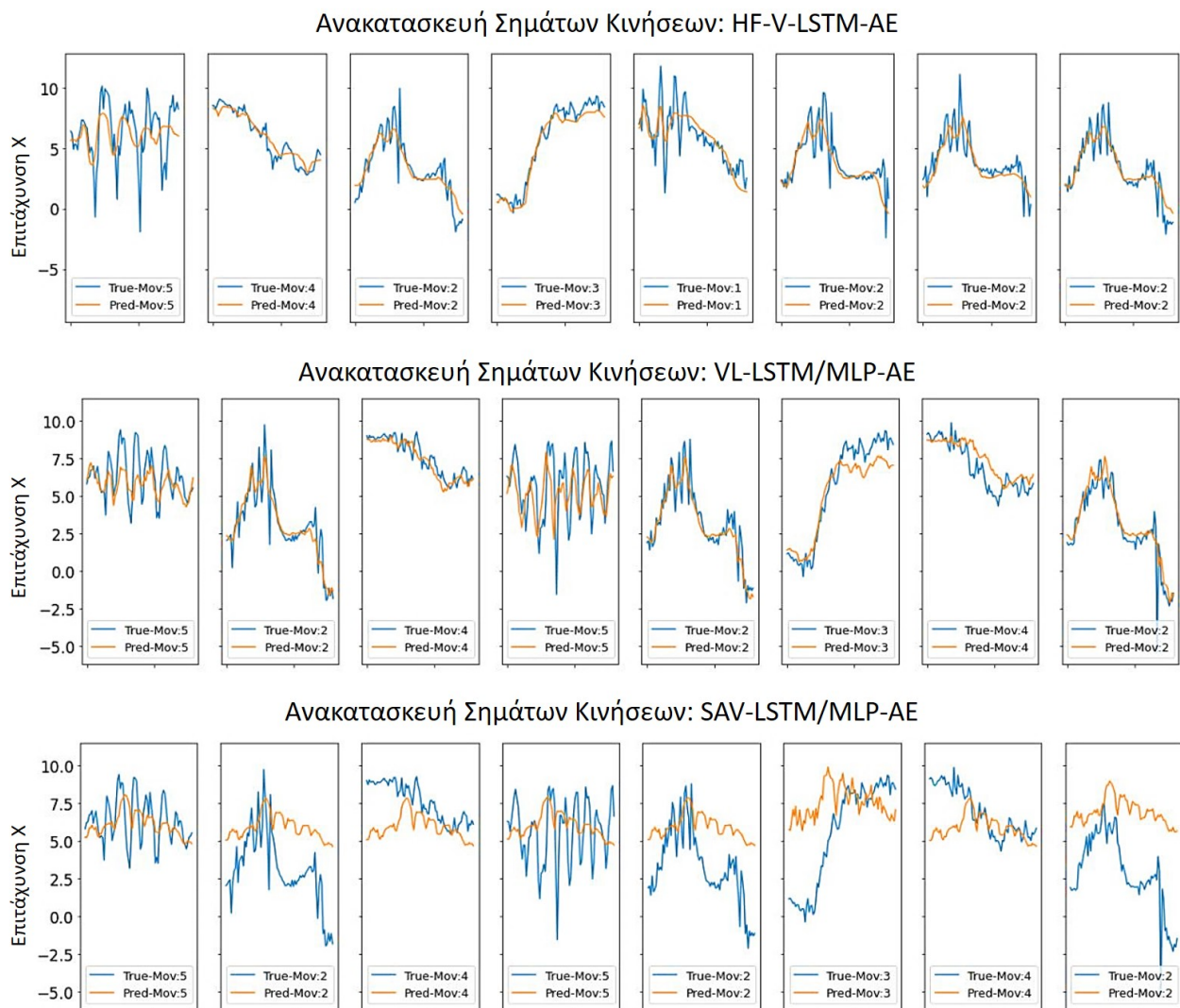
1. Οι τιμές της λογαρ/νης πιθανοφάνειας για τα δίκτυα V-LSTM/MLP-AE και V-LSTM-AE ισούνται με 238.90 και 733.07, αντίστοιχα. Η τεράστια διαφορά των 2 αυτών τιμών προφανώς οφείλεται στην αδυναμία του δικτύου MLP στο τμήμα του αποκωδικοποιητή να ανακατασκευάσει τα σήματα εισόδου. Με αυτόν τον τρόπο τονίζεται η σπουδαιότητα της χρήσης δικτύων LSTM για σήματα χρονοσειρών.
2. Το δίκτυο SAV-LSTM/MLP-AE, βασιζόμενο στη semi-amortized variational συμπερασματολογία, φαίνεται να βελτιώνει σε κάποιο βαθμό την απόδοση του δικτύου V-LSTM/MLP-AE. Παρόλα αυτά, η απόδοσή του παραμένει σχετικά χαμηλή συγκριτικά τα αντίστοιχα δίκτυα τα οποία χρησιμοποιούν δίκτυο LSTM στο τμήμα του αποκωδικοποιητή.
3. Το δίκτυο HF-V-LSTM-AE οδηγεί στη βέλτιστη δυνατή τιμή της συνάρτησης πιθανοφάνειας. Αυτό οφείλεται στο γεγονός ότι η χρήση της κανονικοποιημένης ροής Householder εκτιμά έναν πλήρη πίνακα συνδιασπορών στο μεσαίο επίπεδο, με αποτέλεσμα να ερμηνεύονται αποτελεσματικότερα οι συσχετίσεις των συμπιεσμένων δεδομένων. Συνεπώς, ύστερα από τη δειγματοληψία δεδομένων \mathbf{z} από μία πλήρη Γκαουσιανή κατανομή, η ανακατασκευή των σημάτων είναι πιο αποτελεσματική.
4. Τέλος, το δίκτυο VL-LSTM/MLP-AE οδηγεί στη 2η μεγαλύτερη τιμή της συνάρτησης πιθανοφάνειας. Μάλιστα, η τιμή αυτή είναι υψηλότερη ακόμα και από το δίκτυο V-LSTM-AE, το οποίο στηρίζεται σε δίκτυα LSTM στο τμήμα του αποκωδικοποιητή. Το φαινόμενο αυτό οφείλεται στο γεγονός ότι με τη χρήση των ReLU δικτύων εκτιμήθηκε μία Γκαουσιανή κατανομή με πλήρη πίνακα συνδιασπορών στο μεσαίο επίπεδο, η οποία όμως αυτή τη φορά προήλθε και με τη βοήθεια του αποκωδικοποιητή. Με αυτόν τον τρόπο, παρόλο που σαν αποκωδικοποιητής χρησιμοποιήθηκε δίκτυο MLP, μειώθηκε δραματικά το amortization σφάλμα στο τμήμα του κωδικοποιητή και αυξήθηκε σημαντικά η συνάρτηση πιθανοφάνειας.

Η αποτελεσματικότητα των εξεταζόμενων αυτοκωδικοποιητών μπορεί να αξιολογηθεί με επιπλέον 3 τρόπους. Αρχικά, μπορεί να αναπαρασταθεί γραφικά η ανακατασκευή ορισμένων σημάτων κινήσεων από τα 70 δεδομένα ελέγχου και να πραγματοποιηθεί η σύγκρισή τους με τα αρχικά σήματα εισόδου. Επιπλέον, αρκετά χρήσιμη μπορεί να θεωρηθεί η γραφική αναπαράσταση των συμπιεσμένων δεδομένων \mathbf{z} από ορισμένα σήματα ελέγχου, καθώς με τον τρόπο αυτό ουσιαστικά απεικονίζεται γραφικά πώς τα συμπιεσμένα δεδομένα διαχωρίζονται μεταξύ τους για διαφορετικά είδη κινήσεων. Τέλος, οι γραφικές παραστάσεις της αντικειμενικής συνάρτησης -ELBO κατά τη διάρκεια της εκπαίδευσης αποτελούν έναν επιπρόσθετο τρόπο αξιολόγησης των αυτοκωδικοποιητών. Οι 3 λοιπόν μέθοδοι αυτές παρουσιάζονται ακολούθως.

Ανακατασκευή Σημάτων Εισόδου

Η αποτελεσματικότητα των εξεταζόμενων αυτοκωδικοποιητών εξετάζεται αρχικά μέσα από την ανακατασκευή της επιτάχυνσης X για 8 τυχαία σήματα από τα συνολικά 70 που ανήκουν στο στάδιο ελέγχου. Τα 3 δίκτυα που επιλέγονται να εξετασθούν είναι τα HF-V-LSTM-AE, VL-LSTM/MLP-AE και SAV-LSTM/MLP-AE, τα αποτελέσματα των οποίων παρουσιάζονται στην πρώτη, δεύτερη και τρίτη γραμμή του Σχ. 7.1, αντίστοιχα. Να σημειωθεί ότι η επιλογή της μελέτης των 3 αυτών δικτύων είναι τελείως ενδεικτική, καθώς τα 2 πρώτα παρουσιάζουν παρόμοια απόδοση με το δίκτυο V-LSTM-AE, ενώ το SAV-LSTM/MLP-AE είναι συγκρίσιμο με το δίκτυο V-LSTM/MLP-AE. Συνεπώς, οι ανακατασκευές των σημάτων εισόδου για τους 2 εναπομείναντες αυτοκωδικοποιητές αναμένεται να είναι παρόμοιες με τα 3 δίκτυα που επιλέχθηκαν να μελετηθούν.

Σύμφωνα λοιπόν με τα αποτελέσματα του Σχ. 7.1, τα δίκτυα HF-V-LSTM-AE και VL-LSTM/MLP-AE οδηγούν σε αρκετά ικανοποιητική ανακατασκευή των σημάτων εισόδου. Το γεγονός αυτό είναι αναμενόμενο, καθώς σύμφωνα με τον Πίν. 7.2 η λογαριθμοποιημένη πιθανοφάνεια για τα δίκτυα αυτά παρουσιάζει τις μεγαλύτερες τιμές. Αντιθέτως, το δίκτυο SAV-LSTM/MLP-AE δε φαίνεται ικανό να οδηγεί σε επαρκή ανακατασκευή, γεγονός που οφείλεται στους εξής 2 λόγους: (1) η κατανομή των συμπιεσμένων δεδομένων χαρακτηρίζεται από διαγώνιο πίνακα συνδιασπορών και δεν ερμηνεύει τις μεταξύ τους συσχετίσεις και (2) το δίκτυο MLP δυσχεραίνει την ανακατασκευή χρονοσειρών. Τέλος, η απόδοση των δικτύων αυτών είναι παρόμοια και για τα υπόλοιπες 7 χρονοσειρές που έχουν καταγραφεί με τη χρήση του smartphone.

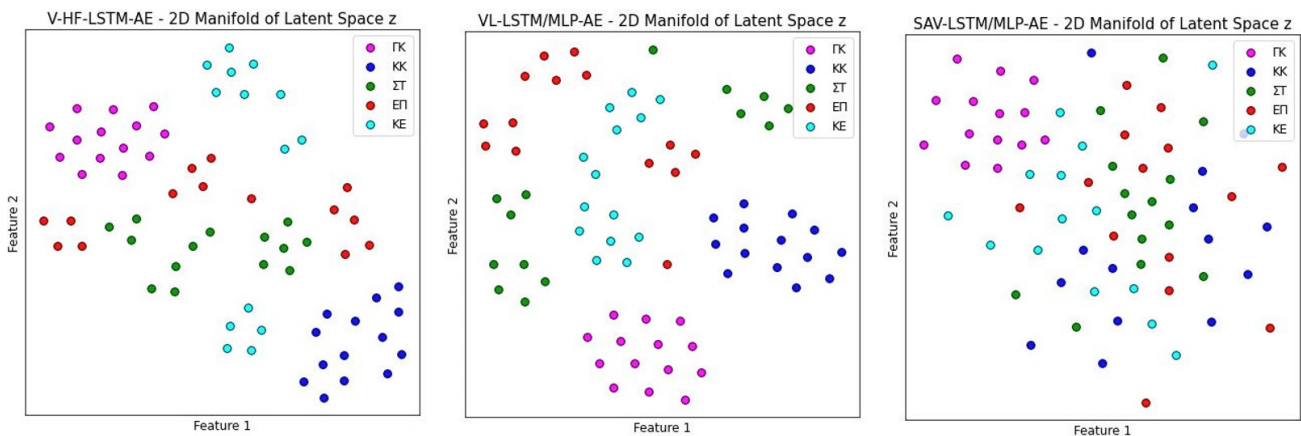


Σχήμα 7.1: Ανακατασκευή επιτάχυνσης X για 8 τυχαία σήματα κινήσεων με χρήση των δικτύων HF-V-LSTM-AE (πρώτη), VL-LSTM/MLP-AE (δεύτερη) και SAV-LSTM/MLP-AE (τρίτη γραμμή).

2D Manifold Συμπιεσμένων Δεδομένων

Για τα 3 αυτά δίκτυα αυτοκωδικοποιητών στη συνέχεια πραγματοποιείται γραφική αναπαράσταση των συμπιεσμένων δεδομένων \mathbf{z} για τα 70 σήματα που ανήκουν στο στάδιο ελέγχου. Η διαστατικότητα των δεδομένων αυτών είναι 250, όσοι δηλαδή και οι νευρώνες των αυτοκωδικοποιητών στο μεσαίο επίπεδο (δες Πίν. 7.1). Συνεπώς, για να πραγματοποιηθεί η διδιάστατη αναπαράστασή τους απαιτείται μείωση της διαστατικότητας των συμπιεσμένων δεδομένων. Αυτό επιτυγχάνεται με τη μέθοδο t-SNE [65], η οποία αποτελεί μία πιθανοτική μέθοδο συμπίεσης χαρακτηριστικών αντίστοιχη της στατιστικής μεθόδου κύριων συνιστωσών (PPCA), δημιουργώντας ένα νέφος (manifold) δεδομένων μειωμένης διαστατικότητας.

Σύμφωνα λοιπόν με το Σχ. 7.2, τα δίκτυα αυτοκωδικοποιητών HF-V-LSTM-AE και VL-LSTM/MLP-AE οδηγούν σε αρκετά εύκολα διαχωρίσιμες περιοχές των συμπιεσμένων δεδομένων \mathbf{z} για τις 5 διαφορετικές κινήσεις. Με αυτόν τον τρόπο διευκολύνεται η ανακατασκευή των σημάτων εισόδου αυξάνοντας ταυτόχρονα τη συνάρτηση πιθανοφάνειας των δεδομένων αυτών. Αντιθέτως, το δίκτυο SAV-LSTM/MLP-AE δε φαίνεται ικανό να εξάγει διαχωρίσιμα χαρακτηριστικά. Συνεπώς, αυτός είναι ο λόγος που κατά την ανακατασκευή των χρονοσειρών εισόδου του Σχ. 7.1, τα ανακατασκευασμένα σήματα αποκλίνουν αισθητά από τα αντίστοιχα της εισόδου και ταυτόχρονα κυμαίνονται εντός ενός πολύ μικρού εύρους τιμών.



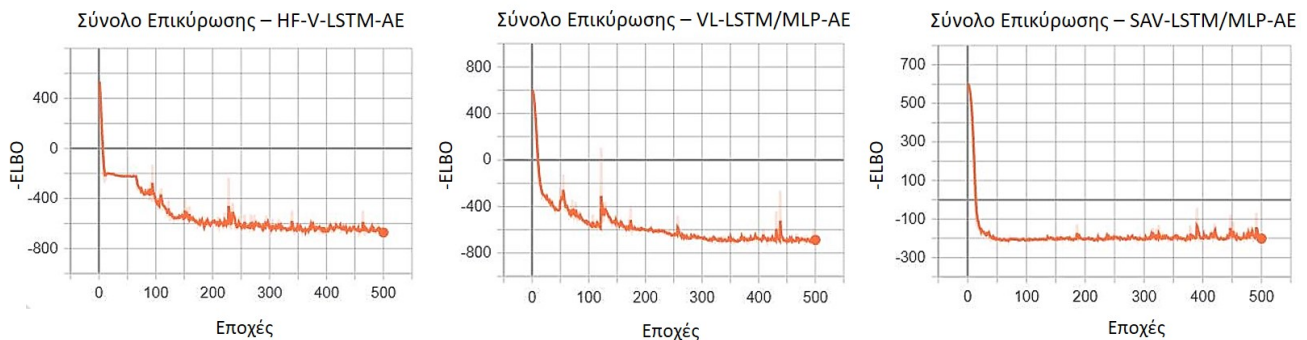
Σχήμα 7.2: 2D manifold της συμπιεσμένης αναπαράστασης \mathbf{z} από τα σήματα κινήσεων βάσει των δικτύων HF-V-LSTM-AE (αριστερά), VL-LSTM/MLP-AE (μέση) και SAV-LSTM/MLP-AE (δεξιά).

Αντικειμενική Συνάρτηση -ELBO

Τελευταίο κριτήριο αξιολόγησης των αυτοκωδικοποιητών αποτελούν οι τιμές της αντικειμενικής συνάρτησης -ELBO κατά τη διάρκεια της εκπαίδευσης των μοντέλων αυτών. Όπως αναφέρθηκε και προηγουμένως, η βιβλιοθήκη pytorch [57] εκτελεί μόνο αλγορίθμους καθόδου κλίσης και για το λόγο αυτό πραγματοποιείται ελαχιστοποίηση της ποσότητας -ELBO αντί για μεγιστοποίηση της ποσότητας ELBO των Κεφαλαίων 3, 4 και 5. Στο Σχ. 7.3 λοιπόν απεικονίζονται οι τιμές της ποσότητας -ELBO για τα 55 δεδομένα επικύρωσης (validation set) κατά τη διάρκεια των 500 εποχών εκπαίδευσης. Να σημειωθεί ότι τα δεδομένα επικύρωσης χρησιμοποιούνται κατά τη διάρκεια της εκπαίδευσης των δικτύων με σκοπό μόνο την αποφυγή φαινομένων υπερεκπαίδευσης και δε συνεισφέρουν καθόλου στην ανανέωση των βαρών.

Σύμφωνα λοιπόν με τα αποτελέσματα που προκύπτουν, παρατηρείται ότι η πόσότητα -ELBO για τα δίκτυα HF-V-LSTM-AE και VL-LSTM/MLP-AE ανέρχεται σε τιμές που προσεγγίζουν το -700. Αντιθέτως, η αντίστοιχη τιμή για το δίκτυο SAV-LSTM/MLP-AE είναι περίπου -200. Το γεγονός αυτό είναι αναμενόμενο, καθώς η ποσότητα ELBO αποτελεί το κατώτερο όριο της λογαριθμοποιημένης συνάρτησης πιθανοφάνειας των παρατηρούμενων δεδομένων, οι τιμές της οποίας παρουσιάστηκαν στον Πίν. 7.2 για όλα τα δίκτυα. Συνεπώς, τα δίκτυα HF-V-LSTM-AE

και VL-LSTM/MLP-AE (μαζί με το δίκτυο V-LSTM-AE) οδηγούν σε πολύ μικρότερο σφάλμα συγκριτικά με το δίκτυο SAV-LSTM/MLP-AE (ή το V-LSTM/MLP-AE) και για το λόγο αυτό επιτυγχάνεται αποτελεσματικότερη εξαγωγή χαρακτηριστικών στο μεσσαίο επίπεδο αυτών των αυτοκωδικοποιητών.



Σχήμα 7.3: Τιμή ποσότητας -ELBO του συνόλου επικύρωσης (validation set) κατά τη διάρκεια των 500 εποχών εκπαίδευσης των σημάτων κινήσεων, βάσει των μοντέλων HF-V-LSTM-AE (αριστερά), VL-LSTM/MLP-AE (μέση) και SAV-LSTM/MLP-AE (δεξιά).

7.1.2 Ταξινόμηση Σημάτων Κινήσεων

Ύστερα από την εκπαίδευση των αυτοκωδικοποιητών, στην παρούσα υποενότητα παρουσιάζεται το πρόβλημα της ταξινόμησης των 5 εξεταζόμενων κινήσεων. Εφόσον τα σήματα που μελετώνται είναι χρονοσειρές, τα νευρωνικά δίκτυα που επιστρατεύονται για το πρόβλημα της ταξινόμησης είναι τα δίκτυα LSTM που παρουσιάστηκαν στην Εν. 2.2. Βέβαια, να σημειωθεί ότι κατά τη διάρκεια της εκπαίδευσης στο τελευταίο επίπεδο των δικτύων προστίθενται τα χαρακτηριστικά που εξάγονται με τη βοήθεια των αυτοκωδικοποιητών. Συνεπώς, εφόσον τα χαρακτηριστικά αυτά είναι εύκολα διαχωρίσιμα για διαφορετικά είδη κινήσεων (δες Σχ. 7.2), έπεται ότι προστίθεται επιπλέον πληροφορία στα δίκτυα LSTM για τη σωστή ταξινόμηση των σημάτων.

Αντίστοιχα λοιπόν με τα δίκτυα των αυτοκωδικοποιητών, έτσι και σε αυτή την περίπτωση απαιτείται η επιλογή κατάλληλων υπερπαραμέτρων πριν την έναρξη της εκπαίδευσης του δικτύου. Ορισμένες από αυτές τις υπερπαραμέτρους αναλύονται συνοπτικά ακολούθως:

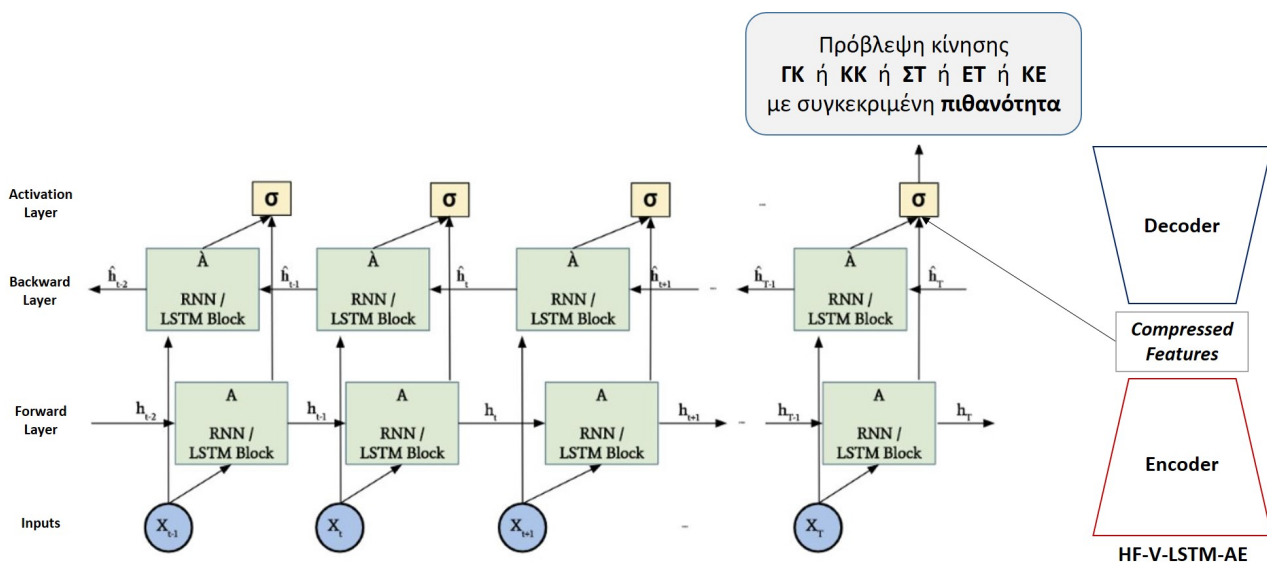
1. Αριθμός Επιπέδων Δικτύου: Εφόσον το εξεταζόμενο δίκτυο είναι LSTM, η υπερπαραμέτρος αυτή αναφέρεται στο πλήθος των δικτύων που είναι προσκολλημένα (stacked) το ένα πάνω στο άλλο και στα οποία η πληροφορία ρέει προς την ίδια κατεύθυνση.
2. Νευρώνες ανά Επίπεδο: Η συγκεκριμένη υπερπαραμέτρος αναφέρεται στη διάσταση του διανύσματος \mathbf{h} εντός του δικτύου LSTM, όπως αυτή παρουσιάστηκε στην υποενότητα 2.2.1.
3. Κατευθύνσεις Δικτύου LSTM: Η υπερπαραμέτρος αυτή αναφέρεται στο τελευταίο επίπεδο του δικτύου LSTM, στο οποίο υπάρχει δυνατότητα προσκόλλησης ενός επιπλέον δικτύου LSTM μέσα στο οποίο η πληροφορία ρέει προς την αντίθετη κατεύθυνση συγκριτικά με τα υπόλοιπα επίπεδα του δικτύου. Με αυτό τον τρόπο μοντελοποιούνται οι συσχετίσεις των σημείων ενός σήματος και προς την αντίθετη κατεύθυνση, με αποτέλεσμα να προστίθεται επιπλέον πληροφορία στο στάδιο της ταξινόμησης των σημάτων.
4. Συντελεστής Εχμάρησης: Όπως και στα δίκτυα των αυτοκωδικοποιητών, η συγκεκριμένη υπερπαραμέτρος αναφέρεται στο συντελεστή των μεθόδων καθόδου κλίσης, οι οποίες παρουσιάστηκαν στην Εν. 2.3. Η μέθοδος βελτιστοποίησης που επιλέγεται και σε αυτήν την περίπτωση είναι ο αλγόριθμος Adam.

Ύστερα από την παρουσίαση των βασικών υπερπαραμέτρων του δικτύου LSTM, αυτές που οδήγησαν στα βέλτιστα αποτελέσματα κατά το στάδιο της ταξινόμησης παρουσιάζονται συνοπτικά στον Πίν. 7.3:

Πίνακας 7.3: Βασικές υπερπαραμέτροι δικτύου LSTM κατά την ταξινόμηση των 5 κινήσεων.

| Αριθμός Επιπέδων | Νευρώνες h ανά Επίπεδο | Κατευθύνσεις Δικτύου | Συν/στής Εκμάθησης |
|---------------------|-----------------------------|-------------------------|-----------------------|
| 1 | 100 | 2 | 10^{-3} |

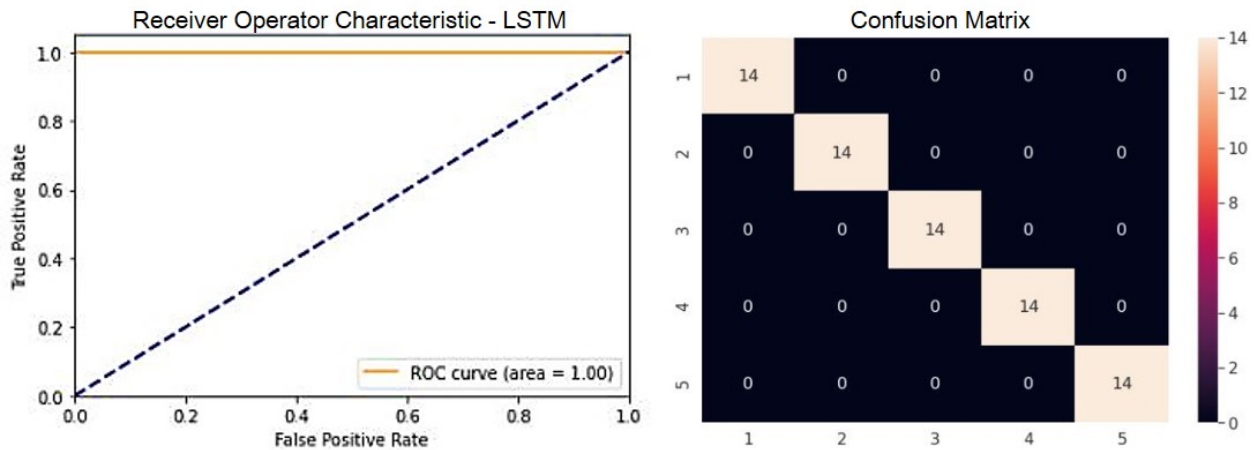
Η αρχιτεκτονική του δικτύου αυτού αναπαρίσταται γραφικά με τη βοήθεια του Σχ. 7.4. Φαίνεται λοιπόν ότι το δίκτυο αποτελείται από 2 στρώματα αντίθετης κατεύθυνσης της πληροφορίας, στα οποία προστίθενται επιπλέον τα χαρακτηριστικά που προέρχονται από το μεσαίο επίπεδο του αυτοκωδικοποιητή. Το δίκτυο αυτοκωδικοποιητή που επιλέγεται είναι το δίκτυο HF-V-LSTM-AE, το οποίο βάσει της υποενότητας 7.1.1 ουσιαστικά οδήγησε στην αποτελεσματικότερη συμπίεση των σημάτων, δημιουργώντας χαρακτηριστικά εύκολα διαχωρίσιμα μεταξύ τους (δες Σχ. 7.2). Με αυτόν τον τρόπο διευκολύνεται η διαδικασία της ταξινόμησης, καθώς προστίθεται επιπλέον πληροφορία στο δίκτυο LSTM για το διαχωρισμό των 5 εξεταζόμενων κινήσεων.



Σχήμα 7.4: Αρχιτεκτονική δικτύου LSTM σε συνδυασμό με τον αυτοκωδικοποιητή HF-V-LSTM-AE για την ταξινόμηση των 5 εξεταζόμενων κινήσεων.

Ύστερα από την επιλογή των κατάλληλων υπερπαραμέτρων ακολουθεί η εκπαίδευση 100 εποχών της αρχιτεκτονικής του Σχ. 7.4 για την ταξινόμηση των 5 εξεταζόμενων κινήσεων. Η απόδοση του δικτύου αυτού εξετάζεται στα 70 δεδομένα ελέγχου με χρήση της καμπύλης ROC Curve [66, 67], η οποία αναπαριστά το ρυθμό με τον οποίο κάθε κίνηση ταξινομείται στη σωστή κίνηση (True Positive Rate) συναρτήσει του ρυθμού με τον οποίο κάθε κίνηση αντιστοιχίζεται σε λάθος κίνηση (False Positive Rate ή False Alarm). Από την καμπύλη αυτή εξάγεται ο συντελεστής AUC (Area Under Curve), ο οποίος ισούται με το εμβαδόν κάτω από την καμπύλη ROC. Ο αριθμός αυτός κυμαίνεται μεταξύ των τιμών 0 έως 1, με τιμές κοντά στο 1 να υποδεικνύουν τέλεια συνολική απόδοση ταξινόμησης των 5 κινήσεων, ενώ τιμές κοντά στο 0.5 να υποδηλώνουν αναποτελεσματική ταξινόμηση [67]. Τέλος, ένα τελευταίο κριτήριο αξιολόγησης του μοντέλου αποτελεί ο Confusion Matrix, στον οποίο αναπαρίσταται γραφικά οι ταξινομήσεις των κινήσεων σε κάθε κλάση. Οι τιμές της διαγωνίου του πίνακα αυτού αντιπροσωπεύουν τα σήματα που έχουν ταξινομηθεί στις σωστές κλάσεις, ενώ οι υπόλοιπες τιμές αφορούν τις λανθασμένες κατηγοριοποιήσεις.

Σύμφωνα λοιπόν με το Σχ. 7.5, παρατηρείται ότι οι 5 εξεταζόμενες κινήσεις ταξινομούνται σωστά και για τα 70 σήματα ελέγχου. Συγκεκριμένα, η καμπύλη ROC ισούται με 1 για όλα τα πιθανά ποσοστά ψευδοσυναγεγμένων και αυτό συνεπάγεται συντελεστή AUC ίσο με 1. Κατά συνέπεια, το δίκτυο LSTM διπλής κατεύθυνσης σε συνδυασμό με τα χαρακτηριστικά του αυτοκωδικοποιητή HF-V-LSTM-AE οδηγούν σε τέλειο διαχωρισμό των εξεταζόμενων κινήσεων, με αποτέλεσμα το ποσοστό σωστής ταξινόμησης να ανέρχεται στο 100%.



Σχήμα 7.5: Αξιολόγηση ταξινόμησης των 5 κινήσεων του δικτύου LSTM με χρήση καμπύλης ROC (αριστερά) και confusion matrix (δεξιά).

7.2 Αποτελέσματα από Σήματα Συνιστωσών Κινήσεων

Στην ενότητα αυτή παρουσιάζονται τα αποτελέσματα της ανάλυσης των 1875 σημάτων που αντιστοιχούν στις 15 συνιστώσες κινήσεις. Η διαδικασία που πρόκειται να ακολουθηθεί είναι παρόμοια με αυτή των 5 κινήσεων που παρουσιάστηκε στις Εν. 6.2 και 6.3, με τη διαφορά ότι απαιτείται εκ νέου καθορισμός των υπερπαραμέτρων των δικτύων που πρόκειται να εκπαιδευτούν. Αρχικά λοιπόν η εκπαίδευση περιλαμβάνει ξανά το διαχωρισμό των σημάτων σε σύνολο εκπαίδευσης, επικύρωσης και ελέγχου. Τα δεδομένα λοιπόν διαχωρίστηκαν ως εξής:

- **Training Set:** χρησιμοποιήθηκε ξανά το 80% του συνόλου δεδομένων. Συνεπώς, από τα 1875 σήματα συνιστωσών κινήσεων τα 1500 χρησιμοποιήθηκαν για την εκπαίδευση των μοντέλων.
- **Validation Set:** χρησιμοποιήθηκε το 8% του συνόλου δεδομένων, δηλαδή 159 σήματα για τη διαδικασία επικύρωσης.
- **Test Set:** χρησιμοποιήθηκε το 12% του συνόλου δεδομένων, δηλαδή 216 σήματα για τη διαδικασία ελέγχου.

Ύστερα από το διαχωρισμό των δεδομένων, στην υποενότητα 7.2.1 που ακολουθεί παρουσιάζονται τα αποτελέσματα τις εξαγωγής χαρακτηριστικών από τα σήματα συνιστωσών κινήσεων, ενώ στην υποενότητα 7.2.2 πραγματοποιείται η ταξινόμηση των σημάτων σε κάθε μία από τις 15 εξεταζόμενες κλάσεις.

7.2.1 Εξαγωγή Χαρακτηριστικών από Σήματα Φάσεων

Αντίστοιχα με τα σήματα κινήσεων, έτσι και σε αυτή την περίπτωση των συνιστωσών κινήσεων εξετάζονται οι 5 διαφορετικοί αυτοκωδικοποιητές που αναφέρθηκαν στην υποενότητα 7.1.1. Η επιλογή των τελικών υπερπαραμέτρων πραγματοποιείται ύστερα από κατάλληλη διερεύνηση των

πιθανών τιμών τους. Οι υπερπαράμετροι λοιπόν που επιλέγονται και οδηγούν στις βέλτιστες δυνατές αποδόσεις των αυτοκωδικοποιητών παρουσιάζονται στον Πιν. 7.4:

Πίνακας 7.4: Υπερπαράμετροι αυτοκωδικοποιητών για τη συμπίεση σημάτων των 15 φάσεων.

| Δίκτυο Κωδικ/τή | Δίκτυο Αποκωδ/τή | Αριθμός Επιπέδων | Νευρώνες h ανά Επίπεδο | Νευρώνες bottleneck z | Συν/στής Εκμάθησης |
|--------------------|---------------------|---------------------|---------------------------|--------------------------|-----------------------|
| LSTM | LSTM | 2 | 300 | 80 | $5 \cdot 10^{-4}$ |
| | MLP | | 350 | | |

Σύμφωνα με τον Πίν. 7.4, το πλήθος των νευρώνων \mathbf{z} στο μεσσαίο επίπεδο όλων των αυτοκωδικοποιητών ανέρχεται στην τιμή 80. Συνεπώς, τα 160 σημεία κάθε σήματος φάσεων (8 καταγραφές \times 20 σημεία/καταγραφή) συμπιέζονται σε 80 χαρακτηριστικά, μειώνοντας ουσιαστικά το πλήθος των διαστάσεων του σήματος κατά 50%. Ακολούθως, πραγματοποιείται σύγκριση μεταξύ των διάφορων παραλλαγών των αυτοκωδικοποιητών εκτιμώντας ξανά τις τιμές της λογαριθμοποιημένης πιθανοφάνειας στα 216 δεδομένα που ανήκουν στο σύνολο ελέγχου.

Οι τιμές λοιπόν της λογαριθμοποιημένης συνάρτησης πιθανοφάνειας για όλα τα μοντέλα αυτοκωδικοποιητών αναγράφονται στον Πίν. 7.5:

Πίνακας 7.5: Λογαριθμοποιημένη πιθανοφάνεια για τους αυτοκωδικοποιητές των 15 φάσεων.

| | V-LSTM/MLP-AE | V-LSTM-AE | SAV-LSTM/MLP-AE |
|--------------------------|---------------|----------------|-----------------|
| Λογαρ/νη Πιθανοφάνεια | 206.78 | 237.56 | 211.36 |
| | HF-V-LSTM-AE | VL-LSTM/MLP-AE | |
| - - | 238.98 | 214.85 | |

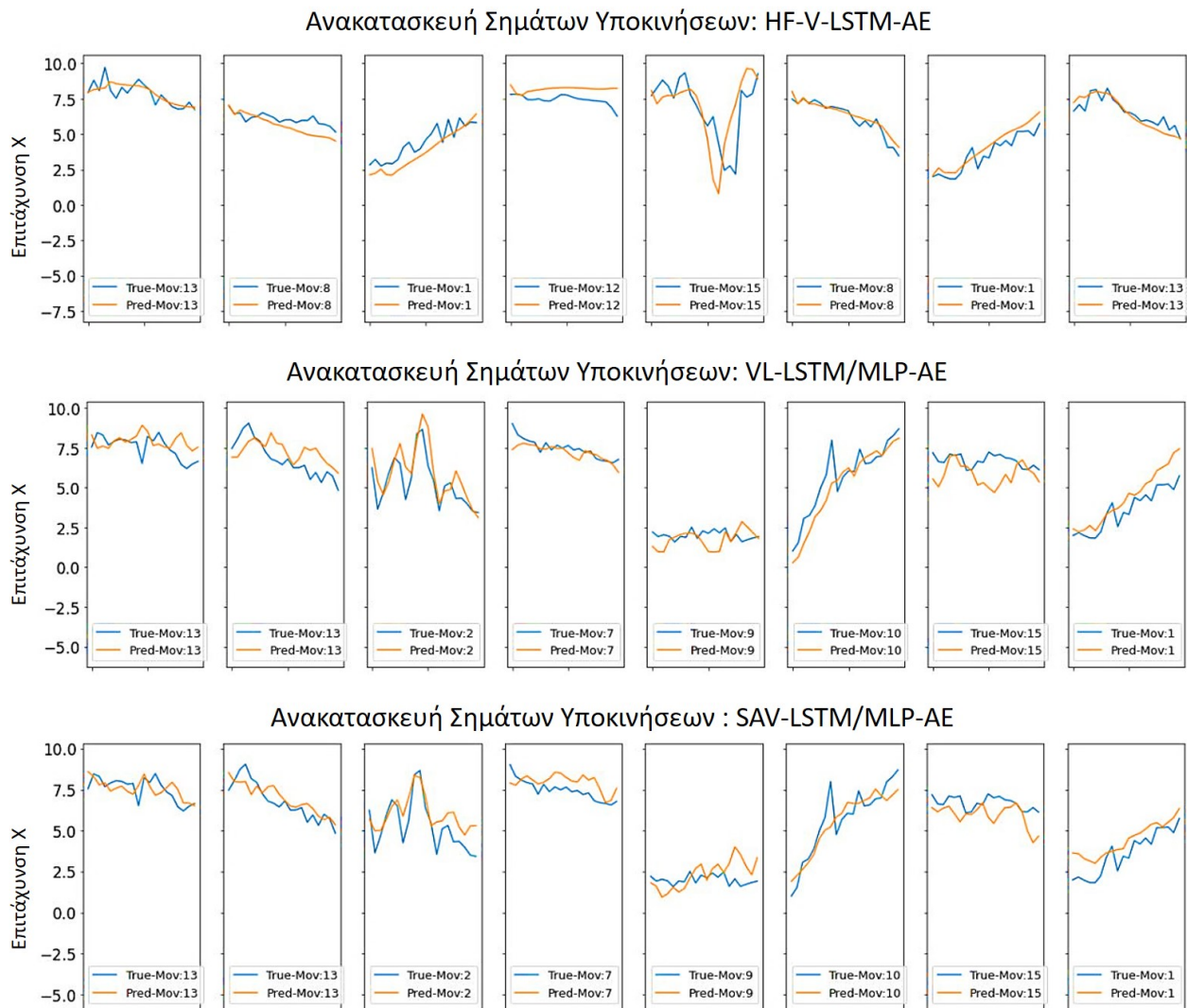
Παρατηρώντας τον Πίν. 7.5, ορισμένα συμπεράσματα που μπορούν να εξαχθούν είναι τα εξής:

1. Οι διαφορές των τιμών της συνάρτησης πιθανοφάνειας δεν είναι τόσο αισθητές για τα 5 δίκτυα αυτοκωδικοποιητών, παρόλο που ορισμένα στηρίζονται στο δίκτυο MLP στο τμήμα του αποκωδικοποιητή. Αυτό συμβαίνει λόγω του πλήθους των σημείων από τα οποία αποτελούνται τα σήματα των συνιστωσών κινήσεων. Συγκεκριμένα, τα 20 σημεία των σημάτων αυτών δε θεωρείται αρκετά μεγάλο νούμερο και κατά συνέπεια ίσως το δίκτυο MLP κατορθώνει να μοντελοποιήσει τις συσχετίσεις των σημείων και να ανατασχευάσει τα σήματα εισόδου.
2. Αντίστοιχα με τα σήματα των κινήσεων, έτσι κι σε αυτή την περίπτωση η υψηλότερη απόδοση επιτυγχάνεται από το δίκτυο HF-V-LSTM-AE, με τιμή συνάρτησης πιθανοφάνειας ίση με 238.98. Αυτό οφείλεται ξανά στην ύπαρξη Γκαουσιανής κατανομής με ολοκληρωμένο πίνακα συνδιασπορών στο μεσσαίο επίπεδο, με αποτέλεσμα να μοντελοποιούνται αποτελεσματικά οι συσχετίσεις των συμπιεσμένων δεδομένων.
3. Ύστερα από το δίκτυο HF-V-LSTM-AE, ακολουθούν τα δίκτυα V-LSTM-AE και VL-LSTM/MLP-AE με τιμές πιθανοφάνειας ίσες με 237.56 και 214.85, αντίστοιχα. Αν και οι τιμές αυτές δε διαφέρουν αισθητά, η χρήση LSTM στο τμήμα του αποκωδικοποιητή φαίνεται να βελτιώνει λίγο την ανακατασκευή των σημάτων εισόδου.
4. Τέλος, τη μικρότερη απόδοση επιτυγχάνουν τα δίκτυα V-LSTM/MLP-AE και SAV-LSTM/MLP-AE με τιμές πιθανοφάνειας ίσες με 206.78 και 211.36, αντίστοιχα. Φαίνεται λοιπόν ότι και σε αυτή την περίπτωση η semi-amortized συμπερασματολογία βελτιώνει λίγο τα αποτελέσματα, αλλά ξανά παραμένει λιγότερο αποτελεσματική από τα υπόλοιπα 3 δίκτυα που αναφέρθηκαν προηγουμένως.

Επιλέγοντας ξανά τα δίκτυα HF-V-LSTM-AE, VL-LSTM/MLP-AE και SAV-LSTM/MLP-AE, στη συνέχεια εξετάζεται η αποτελεσματικότητά τους με τη βοήθεια των 3 τρόπων αξιολόγησης της υποενότητας 7.1.1. Αρχικά λοιπόν αναπαριστάται γραφικά η ανακατασκευή 8 τυχαίων σημάτων συνιστωσών κινήσεων από τα 216 δεδομένα ελέγχου, ενώ στη συνέχεια πραγματοποιείται γραφική αναπαράσταση των συμπιεσμένων δεδομένων \mathbf{z} . Τέλος, η απόδοση των 3 αυτών αυτοκωδικοποιητών εξετάζεται μέσα από τη γραφική παράσταση της ποσότητας -ELBO για τα 159 δεδομένα του συνόλου επικύρωσης.

Ανακατασκευή Σημάτων Εισόδου

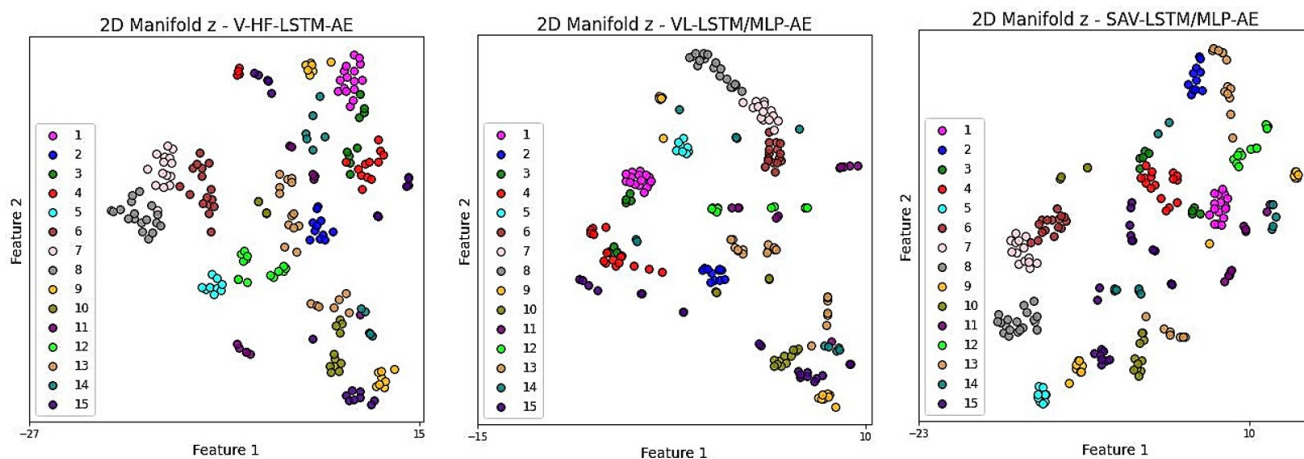
Σύμφωνα με τα αποτελέσματα του Σχ. 7.6, όλα τα δίκτυα φαίνεται να οδηγούν σε αρκετά ικανοποιητική ανακατασκευή των σημάτων εισόδου και δεν παρουσιάζεται κάποια αισθητή διαφορά στην απόδοσή τους. Το γεγονός αυτό είναι αναμενόμενο, καθώς σύμφωνα με τον Πίν. 7.5 η λογαριθμοποιημένη πιθανοφάνεια για τα δίκτυα αυτά δεν παρουσιάζει μεγάλες αποκλίσεις. Έτσι, παρόλη τη χρήση του δικτύου MLP στο τμήμα του αποκωδικοποιητή, η μικρή διάρκεια των συγκεκριμένων σημάτων επιτρέπει την ικανοποιητική ανακατασκευή τους από όλα τα εξεταζόμενα δίκτυα και συνεπώς την αποτελεσματική εξαγωγή χαρακτηριστικών στο μεσοαίο επίπεδο.



Σχήμα 7.6: Ανακατασκευή επιτάχυνσης X για 8 τυχαία σήματα φάσεων με χρήση των δικτύων HF-V-LSTM-AE (πρώτη), VL-LSTM/MLP-AE (δεύτερη) και SAV-LSTM/MLP-AE (τρίτη γραμμή).

2D Manifold Συμπιεσμένων Δεδομένων

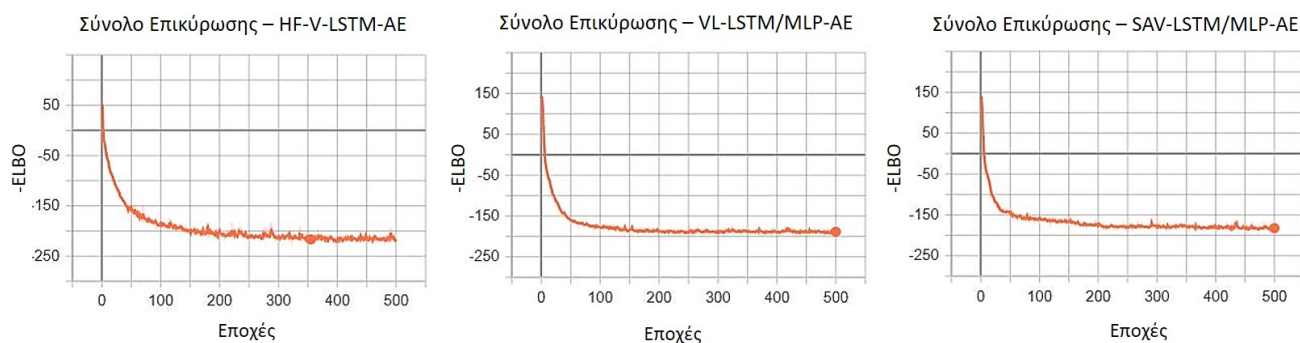
Χρησιμοποιώντας τα 3 αυτά δίκτυα αυτοκωδικοποιητών στη συνέχεια πραγματοποιείται γραφική αναπαράσταση των συμπιεσμένων δεδομένων \mathbf{z} για τα 216 σήματα που ανήκουν στο σύνολο ελέγχου. Η διαστατικότητα των δεδομένων αυτών είναι 80, όσοι δηλαδή και οι νευρώνες των αυτοκωδικοποιητών στο μεσαίο επίπεδο (δες Πίν. 7.4). Συνεπώς, για τη διδιάστατη αναπαράστασή τους απαιτείται μείωση της διαστατικότητας των συμπιεσμένων δεδομένων με τη μέθοδο t-SNE [65]. Σύμφωνα λοιπόν με το Σχ. 7.7, όλα τα δίκτυα αυτοκωδικοποιητών οδηγούν σε αρκετά εύκολα διαχωρίσιμες περιοχές των συμπιεσμένων δεδομένων \mathbf{z} για τις 15 διαφορετικές φάσεις. Με αυτόν τον τρόπο διευκολύνεται η ανακατασκευή των σημάτων εισόδου αυξάνοντας ταυτόχρονα τη συνάρτηση πιθανοφάνειας των δεδομένων αυτών.



Σχήμα 7.7: 2D manifold της συμπιεσμένης αναπαράστασης \mathbf{z} από τα σήματα φάσεων βάσει των δικτύων HF-V-LSTM-AE (αριστερά), VL-LSTM/MLP-AE (μέση) και SAV-LSTM/MLP-AE (δεξιά).

Αντικειμενική Συνάρτηση -ELBO

Τελευταίο κριτήριο αξιολόγησης των αυτοκωδικοποιητών αποτελούν οι τιμές της αντικειμενικής συνάρτησης -ELBO. Στο Σχ. 7.8 λοιπόν απεικονίζονται οι τιμές της ποσότητας -ELBO για τα 159 δεδομένα επικύρωσης (validation set) κατά τη διάρκεια των 500 εποχών εκπαίδευσης. Σύμφωνα λοιπόν με τα αποτελέσματα που προκύπτουν, παρατηρείται ότι η ποσότητα -ELBO για το δίκτυο HF-V-LSTM-AE ανέρχεται σε τιμές μικρότερες του -200, ενώ για τα δίκτυα VL-LSTM/MLP-AE και SAV-LSTM/MLP-AE είναι περίπου -199 και -195, αντίστοιχα. Οι διαφορές αυτές δεν είναι τόσο αισθητές, γεγονός που είναι αναμενόμενο καθώς η ποσότητα ELBO αποτελεί το κατώτερο όριο της λογαριθμοποιημένης συνάρτησης πιθανοφάνειας που παρουσιάστηκε στον Πίν. 7.5.



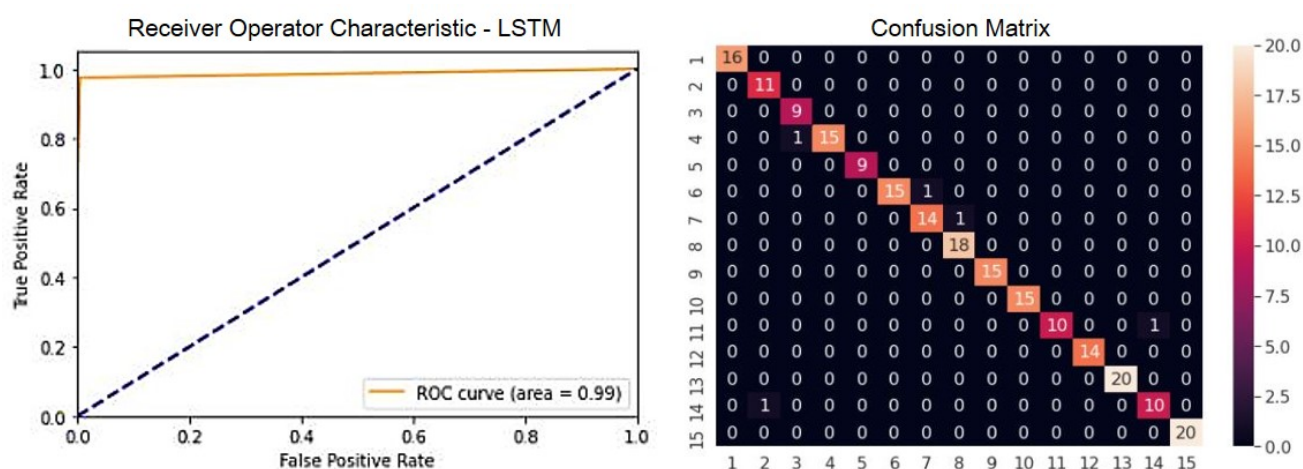
Σχήμα 7.8: Τιμή ποσότητας -ELBO του συνόλου επικύρωσης (validation set) κατά τη διάρκεια των 500 εποχών εκπαίδευσης των σημάτων φάσεων, βάσει των μοντέλων HF-V-LSTM-AE (αριστερά), VL-LSTM/MLP-AE (μέση) και SAV-LSTM/MLP-AE (δεξιά).

Συγκεντρωτικά, όλα τα δίκτυα αυτοκωδικοποιητών φαίνεται να επιτυγχάνουν αποτελεσματική εξαγωγή χαρακτηριστικών στο μεσαίο επίπεδο. Στην υποενότητα 7.2.2 που ακολουθεί το δίκτυο HF-V-LSTM-AE, το οποίο οδηγεί ξανά στα βέλτιστα αποτελέσματα, συνδυάζεται με το δίκτυο LSTM για την κατηγοριοποίηση των 1875 σημάτων των 15 εξεταζόμενων συνιστωσών κινήσεων.

7.2.2 Ταξινόμηση Σημάτων Συνιστωσών Κινήσεων

Ύστερα από την εκπαίδευση των αυτοκωδικοποιητών, στην παρούσα υποενότητα παρουσιάζεται το πρόβλημα της ταξινόμησης των 15 εξεταζόμενων φάσεων. Αντίστοιχα με την υποενότητα 7.1.2 κι εφόσον τα σήματα που μελετώνται είναι χρονοσειρές, τα νευρωνικά δίκτυα που επιστρατεύονται για το πρόβλημα της ταξινόμησης είναι τα δίκτυα LSTM. Επιπλέον, και αυτή τη φορά στο τελευταίο επίπεδο των δικτύων προστίθενται τα χαρακτηριστικά που εξάγονται από τον αυτοκωδικοποιητή HF-V-LSTM-AE της προηγούμενης υποενότητας, όπως δηλαδή αναπαριστάται γραφικά στο Σχ. 7.4. Τέλος, όσον αφορά τις υπερπαραμέτρους του δικτύου LSTM αυτές ταυτίζονται με τις τιμές του Πίν. 7.3.

Σύμφωνα λοιπόν με το Σχ. 7.9, παρατηρείται ότι από τις συνολικά 216 συνιστώσες κινήσεις οι 211 ταξινομούνται σωστά. Αυτό συνεπάγεται ένα πολύ υψηλό ποσοστό σωστής ταξινόμησης των φάσεων ίσο με 97.7%. Για το λόγο αυτό, η καμπύλη ROC είναι σχεδόν ίση με 1 για όλα τα πιθανά ποσοστά ψευδοσυναγεργμών και ο συντελεστής AUC ισούται με 0.99. Κατά συνέπεια, το δίκτυο LSTM διπλής κατεύθυνσης σε συνδυασμό με τα χαρακτηριστικά του αυτοκωδικοποιητή HF-V-LSTM-AE οδηγούν σε αρκετά ικανοποιητικό διαχωρισμό των εξεταζόμενων φάσεων, παρόλο που πολλές από αυτές μπορούν να παρομοιαστούν αρκετά μεταξύ τους.



Σχήμα 7.9: Αξιολόγηση ταξινόμησης των 15 συνιστωσών κινήσεων του δικτύου LSTM με χρήση καμπύλης ROC (αριστερά) και confusion matrix (δεξιά).

7.3 Μελέτη Σημάτων Μακράς Διάρκειας

Ύστερα από τη μελέτη των 625 σημάτων κινήσεων και των 1875 σημάτων φάσεων, ο επόμενος στόχος που τίθεται είναι η αναγνώριση των 5 εξεταζόμενων κινήσεων μέσα σε καταγραφές σημάτων μακράς διάρκειας (ΣΜΔ) που εκτελεσθεί από έναν χειριστή. Τα σήματα αυτά περιλαμβάνουν αφενός μεν κάποια από τις 5 επιθυμητές δραστηριότητες της Εν. 6.1, αφετέρου όμως περιλαμβάνουν και θόρυβο είτε λόγω της εναλλαγής της κίνησης είτε λόγω της ύπαρξης χρονικών περιόδων στις οποίες ο χειριστής δεν πραγματοποιεί κάποια από τις 5 δραστηριότητες (π.χ. περίοδοι αδράνειας, ομιλίας κ.α.). Όσον αφορά λοιπόν τη μαθηματική μοντελοποίηση, στόχος είναι να εφαρμοσθεί μία αρχιτεκτονική νευρωνικών δικτύων η οποία θα προσαρμοστεί στα σήματα μακράς διάρκειας και η

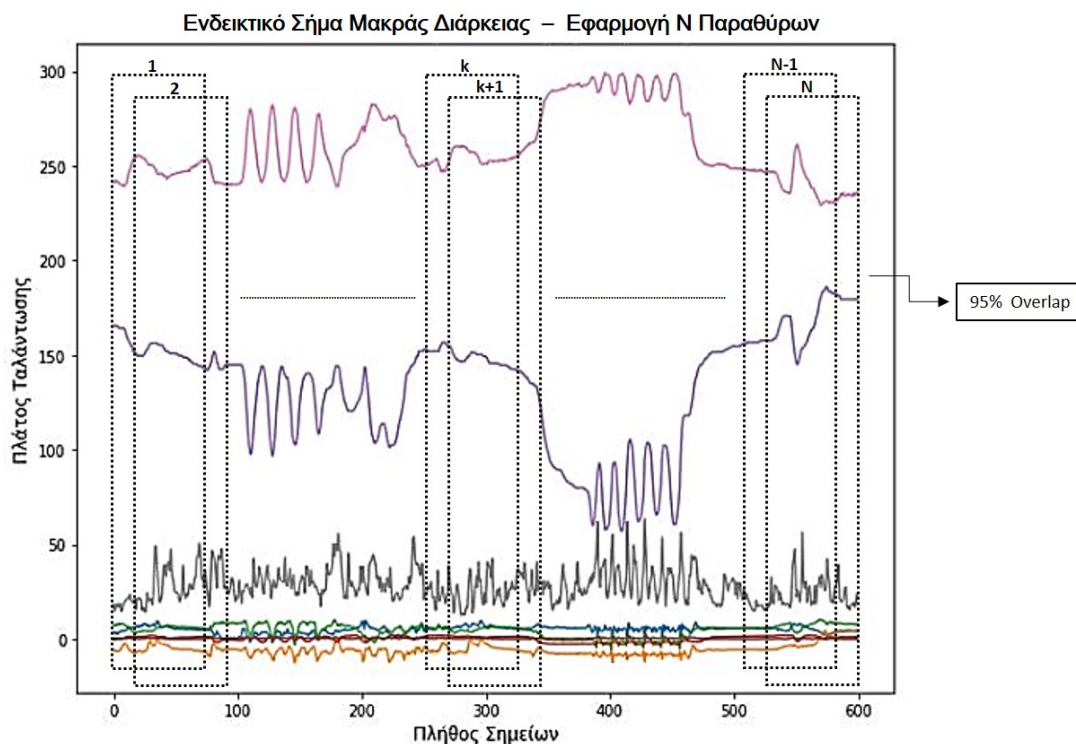
οποία θα "φιλτράρει" τα δεδομένα με στόχο την αναγνώριση των 5 εξεταζόμενων κινήσεων που έχουν πραγματοποιηθεί στις χρονοσειρές αυτές.

Έστω λοιπόν ένα πείραμα μεγάλης διάρκειας με 8 καταγραφές σημάτων, δηλαδή επιτάχυνση x, y, z , φωτεινότητα κλπ., το οποίο περιλαμβάνει τουλάχιστον μία εκ των 5 επιθυμητών κινήσεων. Σκοπός είναι να βρεθεί αφενός μεν ποιές κινήσεις περιλαμβάνονται στο συγκεκριμένο σήμα και αφετέρου με ποιά σειρά πραγματοποιήθηκαν. Για την επίτευξη των στόχων αυτών εφαρμόζονται αλληλοεπικαλυπτόμενα παραθύρα κατά τη διάρκεια του σήματος με συγκεκριμένο ποσοστό επικάλυψης (π.χ. ενδεικτικά 95%). Η διαδικασία προσαρμογής αλληλοεπικαλυπτόμενων παραθύρων επαναλαμβάνεται 2 φορές στην εξεταζόμενη χρονοσειρά μακράς διάρκειας. Στην 1η περίπτωση το μήκος παραθύρου ισούται με τον αριθμό σημείων που αποτελούνται τα σήματα των 5 εξεταζόμενων κινήσεων, ήτοι ίσο με 80, ενώ στη δεύτερη ισούται με τον αριθμό των σημείων που αποτελούνται τα αντίστοιχα σήματα των 15 συνιστωσών κινήσεων, δηλαδή ίσο με 20.

Συνεπώς, ύστερα από την εφαρμογή των αλληλοεπικαλυπτόμενων παραθύρων στη χρονοσειρά προκύπτουν τα εξής:

- N τμήματα χρονοσειρών μήκους 80 σημείων έκαστο, αντίστοιχα με αυτά των 5 κινήσεων και
- M τμήματα χρονοσειρών μήκους 20 σημείων έκαστο, αντίστοιχα με αυτά των 15 φάσεων.

Ενδεικτικό παράδειγμα εφαρμογής των N αλληλοεπικάλυπτων παραθύρων σε μία τυχαία χρονοσειρά μακράς διάρκειας 600 σημείων απεικονίζεται στο Σχ. 7.10. Η χρονοσειρά αυτή αποτελείται από 8 διαφορετικές καταγραφές σημάτων, όσες δηλαδή παρουσιάστηκαν στην Εν. 6.1 μέσω της εφαρμογής Science Journal. Με τον τρόπο αυτό τα N σήματα αυτά μπορούν να τροφοδοτηθούν στα ήδη εκπαιδευμένα μοντέλα των υποενοτήτων 7.1.1 και 7.1.2 για να προβλεφθεί μία από τις 5 κινήσεις με συγκεκριμένη πιθανότητα. Τέλος, αντίστοιχη διαδικασία πραγματοποιείται και για τα M σήματα διάρκειας 20 σημείων, με τη διαφορά ότι τα σήματα αυτά θα τροφοδοτηθούν στα νευρωνικά δίκτυα συνιστωσών κινήσεων των υποενοτήτων 7.2.1 και 7.2.2, ώστε να προβλεφθεί μία από τις 15 φάσεις με συγκεκριμένη πιθανότητα.



Σχήμα 7.10: Διάσπαση ενδεικτικού σήματος μακράς διάρκειας σε N επιμέρους χρονοσειρές με χρήση αλληλοεπικαλυπτόμενων παραθύρων με ποσοστό επικάλυψης 95%.

7.3.1 Αλγόριθμος Απόφασης Αναγνώρισης Κινήσεων

Το ερώτημα που τίθεται είναι πώς μπορούμε να είμαστε σίγουροι για τα αποτελέσματα των ταξινομήσεων, εφόσον στη χρονοσειρά μακράς διάρκειας περιλαμβάνονται χρονικά διαστήματα όπου ο χειριστής δεν πραγματοποιεί κάποια από τις κινήσεις και τις συνιστώσες κινήσεις με τις οποίες εκπαιδεύτηκαν τα νευρωνικά δίκτυα των Εν. 7.1 και 7.2;

Για την αντιμετώπιση του προβλήματος αυτού υλοποιείται ένας αλγόριθμος απόφασης, τα 5 βήματα του οποίου παρουσιάζονται συνοπτικά ακολούθως:

1. Τροφοδότηση των N επιμέρους χρονοσειρών μήκους 80 σημείων το καθένα στα 2 νευρωνικά δίκτυα κινήσεων HF-V-LSTM-AE και LSTM και διατήρηση μόνο εκείνων των τμημάτων από τα οποία προβλέπεται μία κίνηση με πιθανότητα μεγαλύτερη ενός κατωφλίου. Ενδεικτική τιμή πιθανότητας θεωρείται η 0.9. Για κάθε ένα από τα N τμήματα κινήσεων που διατηρείται, κρατείται επίσης και το χρονικό διάστημα εντός του σήματος μακράς διάρκειας στο οποίο εντοπίζεται το τμήμα αυτό.
2. Αντίστοιχα, τροφοδότηση των M επιμέρους χρονοσειρών μήκους 20 σημείων το καθένα στα 2 δίκτυα συνιστωσών κινήσεων HF-V-LSTM-AE και LSTM και διατήρηση μόνο εκείνων των τμημάτων από τα οποία προβλέπεται μία συνιστώσα κίνηση με πιθανότητα μεγαλύτερη ενός δεύτερου κατωφλίου. Ενδεικτική τιμή πιθανότητας θεωρείται η 0.7. Για κάθε ένα από τα M τμήματα φάσεων που διατηρείται, κρατείται επίσης και το χρονικό διάστημα εντός του σήματος μακράς διάρκειας στο οποίο εντοπίζεται το τμήμα αυτό.
3. Δημιουργία μίας μεγάλης λίστας η οποία περιέχει όλες τις προβλέψεις κινήσεων και συνιστωσών κινήσεων που διατηρήθηκαν στα 2 προηγούμενα βήματα μαζί με τα αντίστοιχα χρονικά διαστήματα στα οποία παρατηρήθηκαν εντός του σήματος μακράς διάρκειας.
4. Διατρέχοντας τις προβλέψεις κινήσεων και συνιστωσών κινήσεων της λίστας του βήματος 3, στη συνέχεια πραγματοποιείται αναγνώριση μίας από τις 5 εξεταζόμενες κινήσεις μόνο εάν ισχύει κάποια από τις ακόλουθες προϋποθέσεις:

Κίνηση 1 ή 2 (Βίδωμα με Γερμανικό Κλειδί ή με Κατσαβίδι):

α: Πρόβλεψη της κίνησης 1 ή 2 από τα δίκτυα κινήσεων με πιθανότητα μεγαλύτερη από 0.9 ΚΑΙ τουλάχιστον 2 από τις συνολικά 4 φάσεις που απαρτίζουν την κίνηση αυτή από τα δίκτυα συνιστωσών κινήσεων με πιθανότητα μεγαλύτερη του 0.7 ή,

β: Πρόβλεψη τουλάχιστον 3 από τις συνολικά 4 φάσεις της κίνησης αυτής από τα δίκτυα συνιστωσών κινήσεων με πιθανότητα μεγαλύτερη του 0.8.

Κίνηση 3 ή 4 (Σήκωμα Τεμαχίου ή Εναπόθεση Τεμαχίου):

γ: Πρόβλεψη της κίνησης 3 ή 4 από τα δίκτυα κινήσεων με πιθανότητα μεγαλύτερη από 0.9 ΚΑΙ τουλάχιστον 1 από τις συνολικά 3 φάσεις που απαρτίζουν την κίνηση αυτή από τα δίκτυα συνιστωσών κινήσεων με πιθανότητα μεγαλύτερη του 0.7 ή,

δ: Πρόβλεψη τουλάχιστον 2 από τις συνολικά 3 συνιστώσες κινήσεις της κίνησης αυτής από τα δίκτυα συνιστωσών κινήσεων με πιθανότητα μεγαλύτερη του 0.8.

Κίνηση 5 (Καθαρισμός Επιφάνειας)

ε: Τέλος, για τον καθαρισμό επιφάνειας θα πρέπει να υπάρχει πρόβλεψη τόσο από τα δίκτυα κινήσεων με πιθανότητα μεγαλύτερη από 0.9, όσο και από τα δίκτυα συνιστωσών κινήσεων με πιθανότητα μεγαλύτερη του 0.7.

5. Τέλος, πραγματοποιείται εκτίμηση της σειράς εκτέλεσης των κινήσεων. Χρησιμοποιώντας λοιπόν τις τελικές προβλέψεις του βήματος 4 και ανατρέχοντας στη λίστα του βήματος 3, εξάγονται τα χρονικά διαστήματα εντός του ΣΜΔ στα οποία παρατηρήθηκε κάθε μία τελική κίνηση. Ανάλογα λοιπόν με τα διαστήματα αυτά καθορίζεται επίσης και η σειρά εκτέλεσης των τελικών κινήσεων.

7.3.2 Αποτελέσματα Αλγορίθμου Απόφασης

Εφαρμόζοντας τα 5 βήματα του παραπάνω αλγορίθμου απόφασης, στην παρούσα υποενότητα παρουσιάζονται ορισμένα αποτελέσματα για 50 σήματα μακράς διάρκειας. Τα σήματα αυτά περιλαμβάνουν αφενός μεν κάποια από τις εξεταζόμενες 5 κινήσεις της Εν. 6.1, αφετέρου όμως περιλαμβάνουν και θόρυβο είτε λόγω της εναλλαγής της κίνησης είτε λόγω της ύπαρξης χρονικών περιόδων στις οποίες οι χειριστές εκτελούν ορισμένες κινήσεις μη ενδιαφέροντος. Επιπλέον, εκτός από τις προβλέψεις του αλγορίθμου, στον Πίν. 7.6 αναγράφεται τόσο η χρονική διάρκεια του σήματος μακράς διάρκειας όσο και ο χρόνος εξαγωγής των αποτελεσμάτων με χρήση του προαναφερθέν αλγορίθμου.

Πίνακας 7.6: Πρόβλεψη κινήσεων σε σήματα μακράς διάρκειας με χρήση των νευρωνικών LSTM και HF-V-LSTM-AE. Οι λανθασμένα εμφανιζόμενες προβλέψεις αναγράφονται με bold χαρακτήρες, ενώ οι υπογραμμισμένες προβλέψεις βρίσκονται σε λανθασμένη χρονική σειρά.

| A/A ΣΜΔ | Εκτελεσθείσες Κινήσεις | Πρόβλεψη Μοντέλων | Διάρκεια ΣΜΔ (sec) | Διάρκεια Πρόβλεψης (sec) |
|------------|---------------------------|----------------------|-----------------------|-----------------------------|
| 1 | {1, 2} | { <u>2</u> , 1} | 40.60 | 2.26 |
| 2 | {1, 3} | {1, 3} | 22.47 | 1.19 |
| 3 | {2, 4} | {2, 4} | 22.93 | 1.25 |
| 4 | {2, 5} | {2, 5} | 25.73 | 1.40 |
| 5 | {3, 2} | {3, 2} | 26.27 | 1.43 |
| 6 | {3, 1} | {3, 1} | 24.40 | 1.33 |
| 7 | {4, 1} | {4, 1} | 23.87 | 1.29 |
| 8 | {4, 3} | {4, 3} | 14.47 | 0.73 |
| 9 | {4, 5} | {4, 5} | 17.47 | 0.92 |
| 10 | {5, 1} | {5, 1} | 24.80 | 1.34 |
| 11 | {1, 2, 3} | {1, 2, 3, 4 } | 39.80 | 2.20 |
| 12 | {1, 3, 4} | {1, 3, 4} | 31.13 | 1.73 |
| 13 | {1, 2, 5} | {1, 2, 5} | 44.00 | 2.40 |
| 14 | {3, 4, 5} | {3, 4, 5} | 23.00 | 1.23 |
| 15 | {4, 2, 3} | {4, 1 , 2, 3} | 32.73 | 1.79 |
| 16 | {1, 4, 5} | {1, 4, 5} | 33.33 | 1.84 |
| 17 | {5, 2, 3} | {5, 2, 4 , 3} | 37.40 | 2.03 |
| 18 | {2, 5, 1} | {2, 5, 1} | 45.27 | 2.52 |
| 19 | {4, 5, 1} | {4, 5} | 34.53 | 1.87 |
| 20 | {5, 2, 4} | {5, 3 , 2, 4} | 36.47 | 1.97 |
| 21 | {2, 4, 5} | {2, 4, 5} | 31.13 | 1.83 |
| 22 | {4, 3, 5} | {4, 3, 5} | 26.93 | 1.49 |
| 23 | {5, 3, 4} | {5, 3, 4} | 26.33 | 1.43 |
| 24 | {3, 5, 1} | {3, 4 , 5, 1} | 35.8 | 1.95 |
| 25 | {3, 1, 5} | {3, 1, 4 , 5} | 34.93 | 1.94 |
| 26 | {5, 1, 4} | {5, 1} | 34.27 | 1.87 |
| 27 | {1, 4, 3} | {1, 4, 3} | 30.60 | 1.67 |

| | | | | |
|----|-----------------|---------------------------------|-------|------|
| 28 | {5, 1, 3, 4} | {5, 1, 3, 4} | 52.27 | 3.01 |
| 29 | {5, 2, 4, 1} | {5, <u>1</u> , 4, 2} | 57.53 | 3.21 |
| 30 | {5, 3, 4, 2} | {5, 3, 4, 2} | 48.53 | 2.74 |
| 31 | {3, 1, 2, 4} | {3, 2, 4} | 53.13 | 2.98 |
| 32 | {1, 5, 3, 4} | {2, 5, 3, 4} | 30.97 | 2.21 |
| 33 | {1, 4, 2, 5} | {4, 2, 5} | 48.20 | 2.76 |
| 34 | {2, 3, 4, 5} | {2, <u>5</u> , 3, 4} | 47.73 | 2.46 |
| 35 | {2, 4, 5, 1} | { <u>1</u> , 3, 4, 5, 2} | 56.47 | 3.19 |
| 36 | {2, 5, 4, 1} | { <u>1</u> , 5, 4, 2} | 59.13 | 3.25 |
| 37 | {1, 3, 5, 4} | {1, <u>4</u> , 3, 5} | 56.00 | 3.14 |
| 38 | {2, 4, 5, 3} | {2, 4, 5, 3} | 51.80 | 2.89 |
| 39 | {2, 5, 4, 3} | {2, 5, 4, 3} | 54.27 | 3.03 |
| 40 | {3, 2, 4, 5} | {3, 2, <u>5</u> , <u>4</u> } | 48.53 | 2.68 |
| 41 | {3, 5, 2, 4} | {3, 5, 1, 2, 4} | 50.40 | 2.89 |
| 42 | {4, 5, 2, 3} | {4, 5, 1, 2, 3} | 49.47 | 2.85 |
| 43 | {1, 4, 2, 5, 3} | {1, 4, 2, 5, 3} | 65.46 | 3.78 |
| 44 | {5, 2, 1, 3, 4} | {5, <u>1</u> , <u>2</u> , 3, 4} | 63.26 | 3.67 |
| 45 | {4, 1, 5, 2, 3} | {4, <u>5</u> , <u>1</u> , 2, 3} | 62.00 | 3.54 |
| 46 | {3, 5, 4, 1, 2} | {3, 5, 4, 1, 2} | 64.53 | 3.67 |
| 47 | {5, 2, 4, 3, 1} | {5, 2, <u>1</u> , 4, 3} | 57.07 | 3.21 |
| 48 | {4, 3, 5, 1, 2} | {4, 3, 5, <u>2</u> , <u>1</u> } | 59.33 | 3.30 |
| 49 | {1, 5, 4, 3, 2} | {5, 4, 3, <u>1</u> , 2} | 54.93 | 3.05 |
| 50 | {3, 1, 4, 5, 2} | {3, <u>2</u> , 4, 5, <u>1</u> } | 54.47 | 3.07 |

Η ερμηνεία των αποτελεσμάτων του Πίν. 7.6 θα μπορούσε να πραγματοποιηθεί χρησιμοποιώντας ξανά καμπύλες ROC [66, 67] και τον confusion matrix. Παρόλα αυτά, για να εφαρμοσθούν οι 2 συγκεκριμένες τεχνικές απαιτείται οι προβλέψεις του μοντέλου να αντιστοιχίζονται μία προς μία με τις πραγματικές τιμές. Σύμφωνα όμως με τον Πίν. 7.6 παρατηρούνται τα εξής 2 προβλήματα:

- Σε ορισμένα σήματα μακράς διάρκειας τα μοντέλα προβλέπουν παραπάνω κινήσεις από τις πραγματικές. Για παράδειγμα, οι πραγματικές κινήσεις που περιλαμβάνονται στο ΣΜΔ 11 είναι οι {1, 2, 3}, ενώ ο αλγόριθμος απόφασης που παρουσιάστηκε προβλέπει τις κινήσεις {1, 2, 3, 4}.
- Σε ορισμένα ΣΜΔ ο αλγόριθμος απόφασης προβλέπει λιγότερες κινήσεις από τις εκτελεσθείσες. Χαρακτηριστικό παράδειγμα αποτελεί το ΣΜΔ 19, για το οποίο οι εκτελεσθείσες κινήσεις είναι οι {4, 5, 1} ενώ προβλέπονται οι κινήσεις {4, 5}.

Οι 2 περιπτώσεις αυτές καθιστούν ανέφικτη την εξαγωγή των καμπύλων ROC και για το λόγο αυτό επιστρατεύονται 2 επιπλέον κλάσεις, οι οποίες εξισώνουν τον αριθμό των προβλέψεων με τις εκτελεσθείσες κινήσεις των ΣΜΔ. Η πρώτη κλάση ονομάζεται ADD και προστίθεται σε όλες τις εκτελεσθείσες κινήσεις των ΣΜΔ για τα οποία προβλέπεται μία επιπλέον κλάση. Για παράδειγμα, στο ΣΜΔ 11 οι πραγματικές κινήσεις θα θεωρούνται οι {1, 2, 3, ADD}, ενώ προβλεφθείσες κινήσεις οι {1, 2, 3, 4}. Επιπλέον, η δεύτερη κλάση που χρησιμοποιείται είναι η MISS, η οποία προστίθεται σε όλες τις προβλεφθείσες κινήσεις στις οποίες παραλείπεται μία κίνηση. Για παράδειγμα στο ΣΜΔ 19 οι εκτελεσθείσες κινήσεις θα θεωρούνται οι {4, 5, 1}, ενώ προβλεφθείσες κινήσεις οι {4, 5, MISS}. Με την τεχνική αυτή ουσιαστικά μπορούμε να αξιολογήσουμε πόσες φορές προστέθηκαν επιπλέον προβλέψεις ή παρατηρήθηκαν ελλείψεις στα αποτελέσματα των μοντέλων.

Τέλος, πριν την παρουσίαση των αποτελεσμάτων υπάρχει ένα επιπλέον θέμα το οποίο θα πρέπει να διερευνηθεί. Ας υποθέσουμε ότι για ένα ΣΜΔ προβλέπεται ίδιο πλήθος κινήσεων με όσες είναι οι πραγματικές που περιλαμβάνονται σ' αυτό, με τη διαφορά όμως ότι η σειρά πρόβλεψης δεν είναι

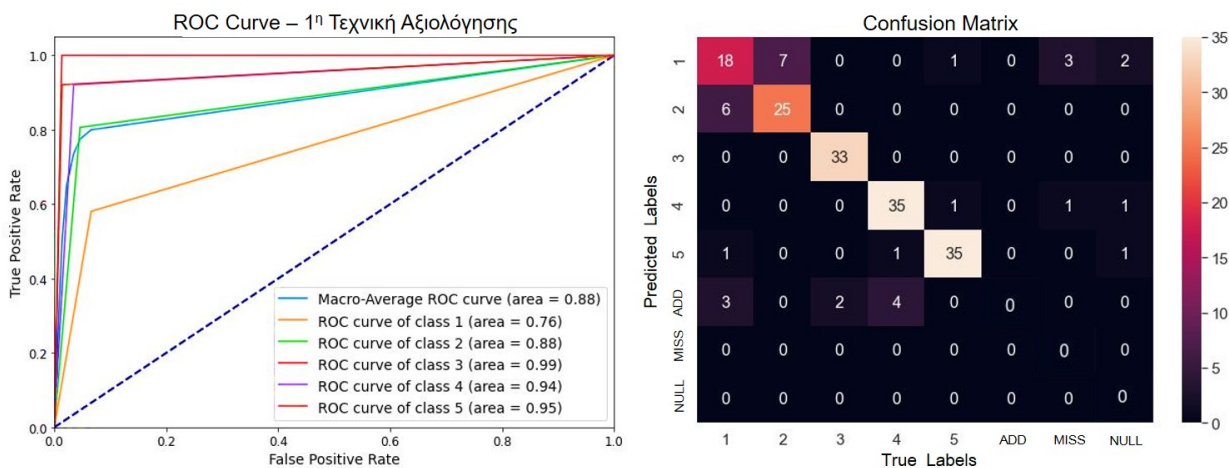
ίδια για τα 2 σύνολα. Ενδεικτικό παράδειγμα αποτελεί το ΣΜΔ 34, για το οποίο οι εκτελεσθείσες κινήσεις είναι οι {2, 3, 4, 5}, ενώ οι προβλεφθείσες οι {2, 5, 3, 4}. Για το παράδειγμα αυτό θα μπορούσαμε να θεωρήσουμε τις εξής 2 τεχνικές αξιολόγησης των αποτελεσμάτων:

1. Η 1η τεχνική αφορά μία επιεική αντιμετώπιση της συγκεκριμένης πρόβλεψης, για την οποία μπορούμε να θεωρήσουμε ότι μόνο η κίνηση 5 ταξινομήθηκε λανθασμένα ως προς τη χρονική σειρά. Αυτό συμβαίνει διότι οι κινήσεις 3 και 4 που ακολουθούν εμφανίζονται μετά την κίνηση 2 τόσο στις πραγματικές τιμές όσο και στις προβλέψεις κι επιπλέον εμφανίζονται με τη σωστή χρονική σειρά (δηλαδή 3, 4 αντί για 4, 3).
2. Η 2η τεχνική αφορά μία πιο αυστηρή αντιμετώπιση της συγκεκριμένης πρόβλεψης για την οποία μπορούμε να θεωρήσουμε ότι και οι 3 κινήσεις που ακολουθούν την κίνηση 2, ήτοι οι 3, 4 και 5 ταξινομούνται όλες λανθασμένα. Αυτό συμβαίνει λόγω της εναλλαγής θέσης πρόβλεψης της κίνησης 5, η οποία συμπαρασύρει σε λανθασμένη ταξινόμηση όλες τις κινήσεις που ακολουθούν.

Όσον αφορά την 2η τεχνική δεν απαιτείται κάποια αλλαγή στην έως τώρα προτεινόμενη μεθοδολογία, αναμένοντας βέβαια πιο χαμηλές αποδόσεις στα αποτελέσματα ταξινόμησης. Αντιθέτως, η 1η τεχνική θα μπορούσε να αντιμετωπιστεί μέσα από την ενσωμάτωση κάθε μίας λανθασμένης χρονικά πρόβλεψης σε μία από τις προαναφερθείσες κλάσεις ADD ή MISS ή εναλλακτικά να δημιουργηθεί μία επιπλέον κλάση NULL για αυτού του είδους τις περιπτώσεις. Για παράδειγμα, για το ΣΜΔ 34 που προαναφέρθηκε με πραγματικές κινήσεις τις {2, 3, 4, 5}, οι προβλέψεις θα μπορούσαν να θεωρηθούν οι {2, 3, 4, NULL}, έτσι ώστε να ταξινομηθεί λανθασμένα χρονικά μόνο η κίνηση 5 και να μη συμπαρασυρθούν μαζί οι κινήσεις 3 και 4. Βέβαια, η συλλογιστική αυτή αναμένεται να αυξήσει τα ποσοστά σωστής ταξινόμησης. Παρόλα αυτά, για καλύτερη εποπτεία της προτεινόμενης μεθοδολογίας ακολουθώντας παρουσιάζονται τα αποτελέσματα και με τις 2 τεχνικές που παρουσιάστηκαν.

Αξιολόγηση 1^{ης} Τεχνικής

Αρχικά λοιπόν επιστρατεύεται η 1η τεχνική εξαγωγής των αποτελεσμάτων, η οποία ισοδυναμεί με μία επιεική αξιολόγηση της προτεινόμενης μεθοδολογίας. Οι κλάσεις που χρησιμοποιούνται σε αυτή την περίπτωση είναι 8, δηλαδή 1, 2, 3, 4, 5, ADD, MISS και NULL. Να τονισθεί ξανά ότι για τις λανθασμένες χρονικά προβλέψεις επιλέχθηκε να χρησιμοποιηθεί η κλάση NULL, ώστε να γνωρίζουμε καλύτερα τη συχνότητα εμφάνισης του συγκεκριμένου φαινομένου. Τα αποτελέσματα λοιπόν που προκύπτουν με την τεχνική αυτή παρουσιάζονται στο Σχ. 7.11:



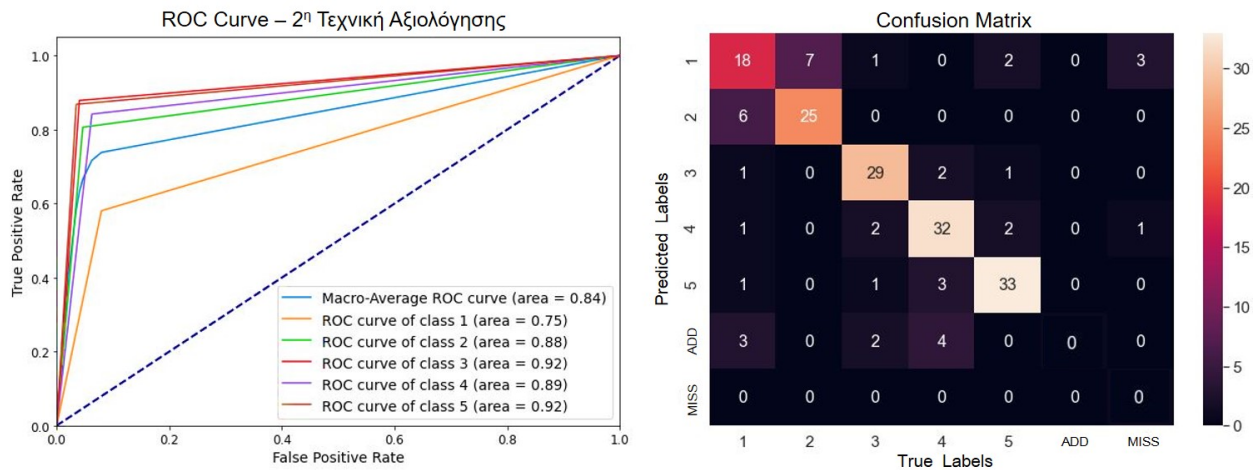
Σχήμα 7.11: Ταξινόμηση σημάτων μακράς διάρκειας (ΣΜΔ) χρησιμοποιώντας την 1^η τεχνική επιεικούς αξιολόγησης των αποτελεσμάτων.

Παρατηρώντας λοιπόν το Σχ. 7.11, ορισμένα συμπεράσματα που μπορούν να εξαχθούν συνοψίζονται ακολούθως:

1. Οι χαμηλότερες επιδόσεις επιτυγχάνονται για τις κινήσεις 1 και 2, για τις οποίες 13 από τις συνολικά 56 εμφανίσεις τους (ποσοστό ίσο με 23%) ταξινομούνται εναλλάξ. Το φαινόμενο αυτό είναι αναμενόμενο, καθώς οι συγκεκριμένες κινήσεις παρομοιάζονται αρκετά μεταξύ τους και η μόνη διαφορά τους έγκειται στον τρόπο με τον οποίο βιδώνεται ο κοχλίας. Βέβαια, να σημειωθεί ότι η εναλλάξ ταξινόμηση παρατηρείται ανεξάρτητα από το πλήθος των κινήσεων που περιλαμβάνονται στα σήματα μακράς διάρκειας του Πίν. 7.6. Όσον αφορά το συντελεστή AUC του Σχ. 7.11, οι τιμές του ανέρχονται σε 0.76 και 0.88 για τις κινήσεις 1 και 2, αντίστοιχα.
2. Οι κινήσεις 3, 4 και 5 παρουσιάζουν αρκετά υψηλότερα ποσοστά σωστής ταξινόμησης συγκριτικά με τις κινήσεις 1 και 2. Αυτό επαληθεύεται από το συντελεστή AUC, οι τιμές του οποίου ανέρχονται σε 0.99, 0.94 και 0.95 για τις κινήσεις 3, 4 και 5, αντίστοιχα. Αυτό οφείλεται στο γεγονός ότι οι 3 αυτές κινήσεις των χειρών των 2 χειριστών είναι πιο εύκολα διαχωρίσιμες μεταξύ τους και επιπλέον δεν παρομοιάζονται με τις κινήσεις 1 και 2 του βιδώματος του κοχλίου.
3. Η συνολική απόδοση της εξετασθείσας μεθοδολογίας κρίνεται αρκετά ικανοποιητική, με συντελεστή AUC ίσο με 0.88 και ποσοστό σωστής ταξινόμησης των 5 εξεταζόμενων κινήσεων ίσο με 81.42%. Το ποσοστό αυτό θεωρείται αρκετά υψηλό αν σκεφτεί κανείς ότι τα αποτελέσματα προκύπτουν με χρήση μόνο μίας συσκευής καταγραφής σημάτων από τον καρπό των 2 χειριστών και επιπλέον τα σήματα μακράς διάρκειας περιλαμβάνουν αρκετό θόρυβο από κινήσεις μη ενδιαφέροντος. Βέβαια, αν δε μας ενδιέφερε η χρονολογική σειρά των κινήσεων και αξιολογούσαμε απλά την ύπαρξη κάθε εκτελεσθείσας κίνησης στο σύνολο των προβλέψεων για κάθε ένα ΣΜΔ, το ποσοστό σωστής ταξινόμησης αυξάνεται στο 92.86%, το οποίο θεωρείται εξαιρετικά υψηλό για τη συγκεκριμένη εφαρμογή.
4. Στην κλάση ADD ταξινομούνται τρεις φορές η κίνηση 1, δύο φορές η κίνηση 3 και τέσσερις φορές η κίνηση 4. Αυτό συνεπάγεται ότι σε 9 διαφορετικά ΣΜΔ από τα συνολικά 50 προστέθηκε μία επιπλέον κίνηση, δηλαδή ποσοστό ίσο με 18%. Βέβαια, και στις 3 φορές όπου προστέθηκε η κίνηση 1 σαν επιπλέον κίνηση (δες ΣΜΔ 15, 41, 42 στον Πίν. 7.6), είναι αξιοσημείωτο ότι αυτό συνέβη ακριβώς πριν την κίνηση 2. Το φαινόμενο αυτό θεωρείται αρκετά λογικό, καθώς όπως προαναφέρθηκε οι 2 αυτές κινήσεις παρουσιάζουν μεγάλο ποσοστό ομοιότητας στα αρχικά βήματα εκτέλεσής τους.
5. Στην κλάση MISS ταξινομούνται τρεις φορές η κίνηση 1 και μία μόνο φορά η κίνηση 4. Αυτό συνεπάγεται ότι σε 4 διαφορετικά ΣΜΔ από τα συνολικά 50 παραλείφθηκε η πρόβλεψη μίας κίνησης, δηλαδή ποσοστό ίσο με 8%. Η συγκεκριμένη κλάση θεωρείται εξαιρετικά σημαντική, καθώς αντιπροσωπεύει την κατηγορία των κινήσεων οι οποίες δεν προβλέφθηκαν καθόλου από τα μοντέλα. Εφόσον λοιπόν το συγκεκριμένο ποσοστό είναι αρκετά χαμηλό, έπεται ότι οποιοσδήποτε ρομποτικός μηχανισμός θα αντιλαμβάνεται την πλειοψηφία των κινήσεων του χειριστή, ώστε στη συνέχεια να επιτευχθεί η αποτελεσματική συνεργασία τους.
6. Στην κλάση NULL ταξινομούνται δύο φορές η κίνηση 1, μία φορά η κίνηση 4 και μία φορά η κίνηση 5. Αυτό σημαίνει ότι σε μόλις 4 διαφορετικά ΣΜΔ από τα συνολικά 50 εφαρμόστηκε η 1^η τεχνική αξιολόγησης με χρήση της κλάσης NULL, δηλαδή ποσοστό ίσο ξανά με 8%. Κατά συνέπεια, τα αποτελέσματα σωστής ταξινόμησης των κινήσεων αναμένεται να μη μειωθούν αισθητά με την αυστηρότερη 2^η τεχνική αξιολόγησης των αποτελεσμάτων.
7. Τέλος, ο χρόνος πρόβλεψης του αλγορίθμου απόφασης είναι κατά μέσο όρο 18 φορές μικρότερος από τη συνολική διάρκεια της κίνησης μακράς διάρκειας. Το γεγονός αυτό είναι πολύ σημαντικό, καθώς η άμεση εξαγωγή των αποτελεσμάτων σε πραγματικό χρόνο θεωρείται καίρια για την αποτελεσματική συνεργασία χειριστή-ρομπότ στο επίπεδο παραγωγής.

Αξιολόγηση 2^{ης} Τεχνικής

Ακολούθως επιστρατεύεται η 2η τεχνική που παρουσιάστηκε προηγουμένως, η οποία αξιολογεί με αυστηρό τρόπο τη χρονολογική σειρά πρόβλεψης όλων των κινήσεων. Οι κλάσεις που χρησιμοποιούνται σε αυτή την περίπτωση είναι 7, δηλαδή 1, 2, 3, 4, 5, ADD, και MISS, ενώ παραλείπεται η κλάση NULL. Τα αποτελέσματα λοιπόν που προκύπτουν με την τεχνική αυτή παρουσιάζονται στο Σχ. 7.12:



Σχήμα 7.12: Ταξινόμηση σημάτων μακράς διάρκειας (ΣΜΔ) χρησιμοποιώντας την 2^η τεχνική αυστηρής αξιολόγησης των αποτελεσμάτων.

Παρατηρώντας το Σχ. 7.12, τα συμπεράσματα που μπορούν να εξαχθούν δε διαφέρουν σημαντικά από τα αντίστοιχα της 1ης τεχνικής. Συγκεκριμένα, οι κινήσεις 1 και 2 φαίνεται να ταξινομούνται εναλλάξ με την ίδια συχνότητα, ενώ στις κλάσεις ADD και MISS ταξινομούνται ακριβώς οι ίδιες κινήσεις με αυτές που προέκυψαν με την 1η τεχνική αξιολόγησης. Η μόνη διαφορά που εντοπίζεται είναι οι αποδόσεις ταξινόμησης για τις κινήσεις 3, 4 και 5, των οποίων οι συντελεστές AUC μειώνονται. Συγκεκριμένα, για την κίνηση 3 ο συντελεστής AUC μειώνεται σε 0.92 από 0.99, για την κίνηση 4 σε 0.89 από 0.94 και τέλος για την κίνηση 5 σε 0.92 από 0.94. Η μη αισθητή μείωση των αποδόσεων οφείλεται στο γεγονός ότι η 1η τεχνική που εισάγει την κλάση NULL εφαρμόστηκε σε μόλις 4 ΣΜΔ, όπως δηλαδή αναφέρθηκε παραπάνω στο συμπέρασμα 6.

Όσον αφορά τη συνολική απόδοση ταξινόμησης, αυτή μειώνεται σε 76.37% από 81.42%, γεγονός που οδηγεί σε συντελεστή AUC ίσο με 0.84. Το φαινόμενο αυτό είναι αναμενόμενο, καθώς με μία μόνο λανθασμένη πρόβλεψη της χρονικής σειράς μίας κίνησης συμπαρασύρονται σε λανθασμένη ταξινόμηση όλες οι κινήσεις που ακολουθούν. Συγκεντρωτικά λοιπόν, οι αποδόσεις και των 2 τεχνικών αξιολόγησης αναγράφονται στον Πίν. 7.7. Τέλος, να σημειωθεί ότι οι αποδόσεις των 2 τεχνικών αξιολόγησης κρίνονται ως αρκετά ικανοποιητικές, καθώς αφενός μεν τα αποτελέσματα εξάγονται σε πολύ σύντομο χρονικό διάστημα χρησιμοποιώντας σήματα μόνο μίας συσκευής καταγραφής από τον καρπό 2 χειριστών και αφετέρου τα σήματα μακράς διάρκειας περιλαμβάνουν αρκετό θόρυβο από άλλες κινήσεις μη ενδιαφέροντος.

Πίνακας 7.7: Αποτελέσματα ταξινόμησης των 5 κινήσεων σε 50 ΣΜΔ με χρήση των 2 τεχνικών αξιολόγησης της προτεινόμενης μεθοδολογίας.

| Τεχνική Αξιολόγησης | Συνολική Ακρίβεια (%) | Macro AUC | AUC ανά Κίνηση | | | | |
|---------------------|-----------------------|-----------|----------------|------|------|------|------|
| | | | 1 | 2 | 3 | 4 | 5 |
| 1 | 81.42 | 0.88 | 0.76 | 0.88 | 0.99 | 0.94 | 0.95 |
| 2 | 76.37 | 0.84 | 0.75 | 0.88 | 0.92 | 0.89 | 0.92 |

Κεφάλαιο 8

Μελέτη Απόκρισης Υπολογιστικών Πόρων

Στο παρόν Κεφάλαιο χρησιμοποιούνται 4 διαφορετικοί υπολογιστικοί πόροι για να ελεγχθεί εάν τα έως τώρα αποτελέσματα μπορούν να εξαχθούν σε πραγματικό χρόνο. Συγκεκριμένα, στην Εν. 8.1 εξηγούνται οι έννοιες του edge computing και computation offloading και τονίζονται τα πλεονεκτήματα κάθε τεχνικής κατά τη διάρκεια της συμπερασματολογίας. Στην Εν. 8.2 περιγράφονται τα χαρακτηριστικά των 4 υπολογιστικών πόρων που θα εξετασθούν και τα αποτελέσματα της σύγκρισής τους παρουσιάζονται στην Εν. 8.3. Η Εν. 8.4 αποτελεί μία εισαγωγή στις έννοιες της συμπίεσης των μοντέλων και παρουσιάζονται ορισμένα αποτελέσματα ύστερα από τη μείωση των παραμέτρων των μοντέλων LSTM. Τέλος, στην Εν. 8.5 παρουσιάζονται πιθανές μελλοντικές επεκτάσεις για τη συμπερασματολογία σε πραγματικό χρόνο και επισημαίνονται ορισμένες προηγμένες τεχνικές που μπορούν να επιταχύνουν τις προβλέψεις των μοντέλων μειώνοντας ταυτόχρονα την υπολογιστική ισχύ που καταναλώνεται.

8.1 Σύγχρονοι Τρόποι Συμπερασματολογίας

Όπως αναφέρθηκε στην Εν. 7.1, όλα τα αποτελέσματα που παρουσιάστηκαν έως τώρα έχουν εξαχθεί με χρήση μίας GPU Tesla T4/NVIDIA-SMI 418.67 με 15 Gb RAM. Αυτό προϋποθέτει ότι κατά την εκτέλεση των πειραμάτων από τους χειριστές, τα δεδομένα που συλλέγονται θα πρέπει να αποστέλλονται στον τοπικό server που βρίσκεται η συγκεκριμένη GPU, ώστε στη συνέχεια να προβλεφθούν οι κινήσεις ενδιαφέροντος με χρήση των εκπαιδευμένων νευρωνικών δικτύων. Βέβαια, η χρήση μίας GPU σε έναν τοπικό server δε θα πρέπει να θεωρείται ως μία μοναδική λύση, καθώς υπάρχουν αρκετοί διαφορετικοί τρόποι για την εξαγωγή των αντίστοιχων αποτελεσμάτων. Συνοπτικά, όλες οι πιθανές μέθοδοι για την εφαρμογή της έως τώρα προτεινόμενης μεθοδολογίας συνοψίζονται ακολούθως:

- **Edge Computing:** Η συγκεκριμένη μέθοδος αφορά την εξαγωγή των αποτελεσμάτων από την πηγή παραγωγής των δεδομένων. Για να επιτευχθεί αυτό, καθίσταται αναγκαία συνθήκη η συσκευή καταγραφής των σημάτων να έχει ενσωματωμένο κύκλωμα που θα μπορεί να εκτελέσει τους απαραίτητους υπολογισμούς.
- **Edge Offloading:** Η συγκεκριμένη μεθοδολογία ουσιαστικά είναι αυτή που χρησιμοποιήθηκε για την εξαγωγή των έως τώρα αποτελεσμάτων. Συγκεκριμένα, σε αυτή την περίπτωση τα δεδομένα αποστέλλονται σε μία κοντινή συσκευή όπως είναι ο τοπικός server (edge device) και στη συνέχεια οι υπολογισμοί πραγματοποιούνται σε αυτή τη συσκευή.
- **Cloud Offloading:** Η συγκεκριμένη μεθοδολογία παρομοιάζεται με την ακριβώς προηγούμενη, με τη διαφορά όμως ότι τα δεδομένα αποστέλλονται στο cloud για να πραγματοποιηθούν οι υπολογισμοί. Αυτό σημαίνει ότι ο υπολογιστικός πόρος που εξάγει τα αποτελέσματα δε βρίσκεται κοντά στα δεδομένα και κατά συνέπεια ο χρόνος αποστολής των δεδομένων μπορεί να καθυστερήσει τις προβλέψεις των μοντέλων.

Οι όροι λοιπόν του edge computing και computation offloading (edge ή cloud) παρουσιάζονται αναλυτικότερα στις υποενότητες 8.1.1 και 8.1.2, αντίστοιχα.

8.1.1 Edge Computing

Ο όρος edge computing αναφέρεται στο μοντέλο εκείνο το οποίο επεκτείνει τις υπηρεσίες του cloud computing σε συσκευές που θεωρούνται κόμβοι (edges) του δικτύου. Με αυτόν τον τρόπο, οι υπηρεσίες του δικτύου μεταφέρονται όσο το δυνατόν πλησιέστερα στα δεδομένα και κατά συνέπεια αποτρέπεται η μεταφορά των δεδομένων στο cloud, η οποία λόγω του όγκου των δεδομένων θεωρείται μία αρκετά χρονοβόρα διαδικασία. Οι κόμβοι του δικτύου που χρησιμεύουν για την επεξεργασία των δεδομένων ονομάζονται edge devices και συνεισφέρουν στη βελτίωση της απόδοσης τέτοιου είδους συστημάτων.

Εκτός από τη μη αναγκαιότητα αποστολής των δεδομένων, η τεχνική edge computing αυξάνει επίσης τα επίπεδα ασφάλειας των υπολογισμών, καθώς αυτοί πραγματοποιούνται από τις συσκευές που συλλέχθηκαν τα δεδομένα κι όχι από μία απομακρυσμένη συσκευή στο cloud. Επιπλέον, στις περισσότερες περιπτώσεις η ισχύς που καταναλώνεται από τις συσκευές αυτές είναι μικρότερη από τις αντίστοιχες του cloud και συνεπώς προτιμούνται για θέματα ενεργειακής κατανάλωσης. Για να επιτευχθεί αυτό χρησιμοποιούνται ενσωματωμένες κάρτες γραφικών GPUs (π.χ. Jetson boards) ή ενσωματωμένες συστοιχίες πυλών προγραμματιζόμενου πεδίου FPGAs με μικρότερη απόδοση συγκριτικά με τις αντίστοιχες cloud συσκευές.

Η ιδέα της μεταφοράς των υπολογιστικών πόρων κοντά στα δεδομένα δεν είναι κάτι καινούριο σαν ιδέα [70]. Ο όρος edge computing χρησιμοποιήθηκε για πρώτη φορά το 2002 στον τομέα της επιχειρηματικότητας, όπου συγκεκριμένες εφαρμογές έπρεπε να μεταφερθούν από τον cloud server σε συγκεκριμένο κόμβο του δικτύου. Επιπλέον, ο ίδιος όρος επίσης χρησιμοποιήθηκε το

2004 για να περιγράψει ένα σύστημα, το οποίο μοίραζε τα δεδομένα και εκτελούσε καταναμημένα τις διεργασίες με σκοπό τη βελτίωση της απόδοσης των εφαρμογών [71].

8.1.2 Edge και Cloud Offloading

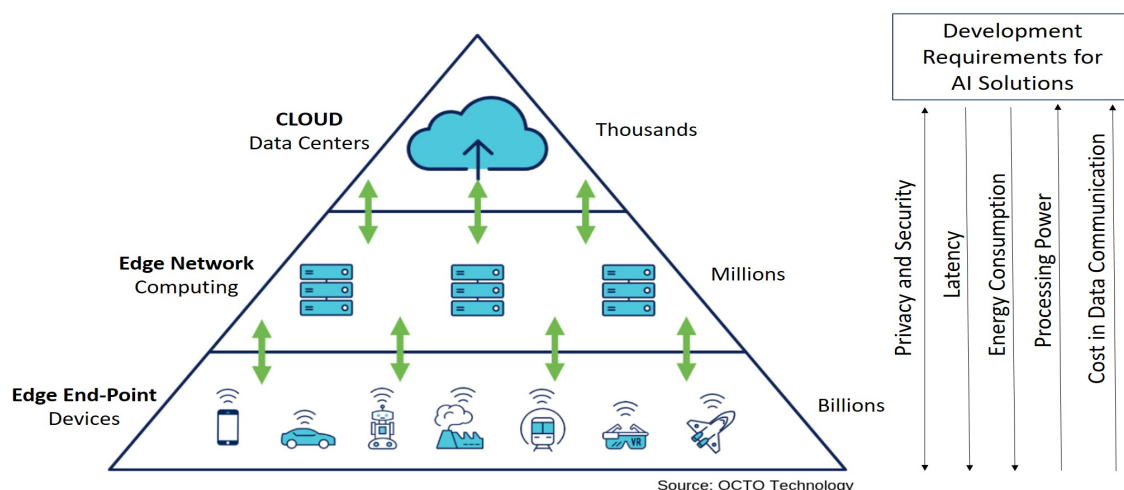
Η τεχνική computation offloading αναφέρεται στην εκτέλεση υπολογιστικών διεργασιών από έναν υπολογιστικό πόρο διαφορετικό του κύριου επεξεργαστή της συσκευής καταγραφής των δεδομένων. Τέτοιοι βοηθητικοί υπολογιστικοί πόροι μπορεί να είναι μία απομακρυσμένη συσκευή όπως ένας τοπικός (π.χ. με GPU Tesla V4) ή ένας cloud server (π.χ. με GPU V100). Η τεχνική αυτή χρησιμοποιείται στις περιπτώσεις όπου οι συσκευές καταγραφής δεν καθίστανται αρκετές για την ανάλυση των δεδομένων και κατά συνέπεια απαιτείται μία επιπρόσθετη συσκευή με μεγαλύτερη υπολογιστική ισχύ και αποθηκευτικό χώρο.

Στις περιπτώσεις λοιπόν που χρησιμοποιείται η τεχνική computation offloading, ο συνολικός χρόνος απόκρισης μπορεί να εκτιμηθεί αθροίζοντας τα εξής:

1. Το χρόνο αποστολής των δεδομένων από τη συσκευή καταγραφής στον βοηθητικό υπολογιστικό πόρο. Ο χρόνος αυτός θεωρείται μία αρκετά σημαντική υπερπαραμέτρος, καθώς μπορεί να καθυστερήσει αρκετά το συνολικό χρόνο απόκρισης.
2. Το χρόνο εκτέλεσης των διεργασιών στο βοηθητικό υπολογιστικό πόρο. Στην παρούσα εργασία ο χρόνος αυτός ισοδυναμεί με την πρόβλεψη των κινήσεων που περιλαμβάνονται σε ένα σήμα μακράς διάρκειας χρησιμοποιώντας τα εκπαιδευμένα δίκτυα του Κεφαλαίου 7.

Συνεπώς, όσον αφορά την τεχνική computation offloading αρχικά απαιτείται να εκτιμηθεί ο χρόνος μεταφοράς κάθε εξεταζόμενου σήματος στον τοπικό (edge offloading) ή στον cloud server (cloud offloading) και στη συνέχεια αυτοί να αθροιστούν στους χρόνους πρόβλεψης των μοντέλων. Θα πρέπει να σημειωθεί ότι συγκρίνοντας τον τοπικό με τον cloud server, η επεξεργαστική δύναμη του δεύτερου συνήθως είναι καλύτερη και η πρόβλεψη των μοντέλων ολοκληρώνεται σε μικρότερο χρονικό διάστημα. Αντιθέτως, ο χρόνος μεταφοράς των δεδομένων στο cloud καθυστερεί αρκετά, ενώ παράλληλα η κατανάλωση ισχύος είναι αρκετά μεγαλύτερη και συνεπώς όχι ενεργειακά αποδοτική. Τέλος, ένας cloud server απαιτεί μεγαλύτερο κόστος χρημάτων, εφόσον ανά έτος προϋποθέτει την επαναλαμβανόμενη πληρωμή για τη συνεχή χρήση του.

Συγκεντρωτικά λοιπόν, η πυραμίδα του Σχ. 8.4 συνοψίζει τα οφέλη της χρήσης μίας edge device ή ενός cloud server σχετικά με την ασφάλεια των δεδομένων, την κατανάλωση ισχύος, το χρόνο συμπερασματολογίας, την επεξεργαστική δύναμη των συσκευών και τέλος τη χρονική καθυστέρηση λόγω της μεταφοράς των δεδομένων.



Σχήμα 8.1: Σύγκριση τεχνικών edge και cloud computing.

8.2 Πειραματικό Πλαίσιο

Όπως προαναφέρθηκε, τα έως τώρα αποτελέσματα έχουν εξαχθεί με χρήση μίας GPU Tesla T4/NVIDIA-SMI 418.67 με 15 Gb RAM από έναν τοπικό server. Στην παρούσα ενότητα παρουσιάζονται 3 επιπλέον υπολογιστικοί πόροι που μπορούν να χρησιμοποιηθούν κατά το στάδιο της συμπερασματολογίας, οι οποίοι είναι οι εξής:

1. Cloud Server: Αρχικά εξετάζεται ένας cloud server, ο οποίος περιλαμβάνει 2 Intel Xeon Gold 6138 CPU @ 2.00GHz, 128GB RAM και μία NVIDIA Tesla V100 GPU. Τα χαρακτηριστικά αυτά επιτρέπουν τόσο την εκπαίδευση των μοντέλων, όσο και τη διαδικασία της συμπερασματολογίας. Παρόλα αυτά, στη συγκεκριμένη εργασία εξετάζεται μόνο ο χρόνος της συμπερασματολογίας, έτσι ώστε να πραγματοποιηθεί μία σύγκριση του χρόνου απόκρισης των διαφορετικών υπολογιστικών πόρων.
2. NVIDIA Xavier AGX: Ο συγκεκριμένος επιταχυντής ουσιαστικά είναι μία συσκευή η οποία αποτελείται από έναν 8-πύρηνο ARM επεξεργαστή και μία Volta GPU 512 πυρήνων (CUDA cores) στο ίδιο ολοκληρωμένο κύκλωμα (System on Chip - SoC), το οποίο στη συνέχεια θα μπορούσε να ενσωματωθεί στη συσκευή καταγραφής σημάτων. Επιπλέον, η μνήμη RAM της συσκευής αυτής ανέρχεται στα 32GB.
3. NVIDIA Xavier NX: Ο συγκεκριμένος επιταχυντής παρομοιάζεται με τον AGX και αποτελείται από έναν 6-πύρηνο Carmel ARM επεξεργαστή, 8GB RAM και μία Volta GPU 384 πυρήνων (CUDA cores).

Η μελέτη της συμπερασματολογίας των εκπαιδευμένων δικτύων στις συνολικά 4 συσκευές που προαναφέρθηκαν αποσκοπεί σε 2 κυρίως λόγους. Αρχικά, αν η καταγραφή των σημάτων πραγματοποιηθεί από μία συσκευή στην οποία έχουν ενσωματωθεί τα ολοκληρωμένα κυκλώματα AGX και NX, τότε τα δεδομένα θα μπορούσαν να αναλύονται επιτόπου χωρίς να απαιτείται η αποστολή τους σε έναν απομακρυσμένο server. Κατά συνέπεια, με αυτόν τον τρόπο εξετάζεται η τεχνική του edge computing που παρουσιάστηκε στην υποενότητα 8.1.1. Εναλλακτικά, τα δεδομένα που συλλέγονται από τις συσκευές καταγραφής σημάτων θα μπορούσαν να αποστέλλονται στον τοπικό ή στο cloud server και στη συνέχεια να εξάγονται οι προβλέψεις των δικτύων. Συνεπώς, με αυτόν τον τρόπο θα εξετασθούν οι τεχνικές του edge και cloud offloading που παρουσιάστηκαν στην υποενότητα 8.1.2. Συγκεντρωτικά λοιπόν, εξετάζοντας τους 4 αυτούς υπολογιστικούς πόρους μπορούν να εξαχθούν χρήσιμα συμπεράσματα από τη χρήση τους και να ελεγχθεί αν μπορούν να χρησιμοποιηθούν στη γραμμή παραγωγής ενός εργοστασίου.

8.3 Πειραματικά Αποτελέσματα

Στην παρούσα ενότητα παρουσιάζονται τα αποτελέσματα της συμπερασματολογίας των νευρωνικών δικτύων, συγκρίνοντας τις αποδόσεις των 4 υπολογιστικών πόρων που παρουσιάστηκαν στην Εν. 8.2. Συγκεκριμένα, στην υποενότητα 8.3.1 συγκρίνονται οι χρόνοι συμπερασματολογίας των 4 συσκευών και εξετάζεται κατά πόσο αυτές μπορούν να εξάγουν αποτελέσματα σε πραγματικό χρόνο. Επίσης, η υποενότητα 8.3.2 αφορά την υπολογιστική ισχύ που καταναλώνεται από τη χρήση των 4 διαφορετικών υπολογιστικών πόρων και εξετάζεται ποιά από τις συσκευές αυτές είναι πιο ενεργειακά προσοδοφόρα. Κατά συνέπεια, ύστερα από την ολοκλήρωση των συγκρίσεων αυτών θα είμαστε σε θέση να γνωρίζουμε ποιός από τους 4 υπολογιστικούς πόρους που εξετάζονται είναι προτιμητέος όσον αφορά το χρόνο συμπερασματολογίας, τη χρονική καθυστέρηση λόγω της μεταφοράς των δεδομένων και τέλος την κατανάλωση ισχύος.

8.3.1 Συμπερασματολογία σε Πραγματικό Χρόνο

Αρχικά λοιπόν εξετάζεται κατά πόσο οι 4 υπολογιστικοί πόροι μπορούν να εξάγουν αποτελέσματα σε πραγματικό χρόνο, ούτως ώστε στη συνέχεια να ενσωματωθούν σε ένα πραγματικό περιβάλλον παραγωγής ενός εργοστασίου. Ας υποθέσουμε λοιπόν ότι ένας χειριστής ξεκινά να πραγματοποιεί ορισμένες διεργασίες και ταυτόχρονα η συσκευή καταγραφής σημάτων που βρίσκεται στον καρπό του αποθηκεύει τα δεδομένα από τις κινήσεις του χεριού του. Σύμφωνα με την Εν. 7.3, για να προσδιοριστεί αν σε ένα σήμα μακράς διάρκειας (ΣΜΔ) υπάρχει κάποια από τις 5 εξεταζόμενες κινήσεις, απαιτείται ο διαχωρισμός του σήματος αυτού σε μικρότερα τμήματα χρονοσειρών 2 παραθύρων με συγκεκριμένο ποσοστό επικάλυψης. Το πρώτο παράθυρο αφορά τα σήματα κινήσεων και διαχωρίζει οποιοδήποτε ΣΜΔ σε μικρότερα τμήματα χρονοσειρών 80 σημείων έκαστο, ενώ το δεύτερο παράθυρο αφορά τα σήματα συνιστωσών κινήσεων και οδηγεί σε τμήματα χρονοσειρών 20 σημείων έκαστο.

Για να ελεγχθεί λοιπόν αν οι 4 υπολογιστικοί πόροι μπορούν να εξάγουν αποτελέσματα σε πραγματικό χρόνο ακολουθούνται τα εξής βήματα:

1. Ο χειριστής ξεκινά να εκτελεί διάφορες διεργασίες. Μόλις συμπληρωθούν τα πρώτα 80 σημεία της χρονοσειράς, αυτά αποστέλλονται στον τοπικό ή στον cloud server για τη συμπερασματολογία, ή εναλλακτικά πραγματοποιείται πρόβλεψη από τα κυκλώματα AGX και NX που βρίσκονται ενσωματωμένα στη συσκευή καταγραφής των σημάτων.
2. Εφόσον τα διαδοχικά παράθυρα παρουσιάζουν συγκεκριμένο ποσοστό επικάλυψης, στη συνέχεια απαιτείται ένα μικρότερο πλήθος σημείων συγκριτικά με τα πρώτα 80, έτσι ώστε να δημιουργηθεί το 2ο χρονικά παράθυρο 80 σημείων. Για παράδειγμα, εφόσον στην παρούσα εργασία το ποσοστό επικάλυψης ανέρχεται στο 95% βάσει του Σχ. 7.10, αυτό συνεπάγεται ότι τα τελευταία 76 σημεία του πρώτου παραθύρου ταυτίζονται με τα πρώτα 76 του 2ου παραθύρου. Συνεπώς, απαιτείται η συμπλήρωση 4 επιπλέον σημείων για δημιουργηθεί το δεύτερο κατά σειρά παράθυρο.
3. Μόλις συμπληρωθεί το απαραίτητο πλήθος σημείων για το 2ο παράθυρο μήκους 80 σημείων, αυτό αποστέλλεται ξανά στον τοπικό ή στον cloud server, ή εναλλακτικά πραγματοποιείται επιτόπου πρόβλεψη από τους επιταχυντές AGX και NX. Η διαδικασία αυτή επαναλαμβάνεται έως ότου ο χειριστής ολοκληρώσει τις διεργασίες τις οποίες εκτελεί και κατά συνέπεια δεν υπάρχουν υπολοίπομα παράθυρα στο σήμα μακράς διάρκειας.

Σύμφωνα με τα παραπάνω βήματα, για να πραγματοποιηθεί πρόβλεψη των 5 εξεταζόμενων κινήσεων σε πραγματικό χρόνο, θα πρέπει για κάθε παράθυρο ο συνολικός χρόνος απόκρισης των 4 υπολογιστικών πόρων να είναι μικρότερος από το χρόνο που απαιτείται ώστε να συμπληρωθούν τα σημεία του επόμενου παραθύρου. Ο περιορισμός αυτός καθίσταται αναγκαία συνθήκη για μία σωστή συμπερασματολογία, καθώς θα αρχίσουν να συσσωρεύονται παράθυρα για τα οποία θα καθυστερεί αρκετά η πρόβλεψη των μοντέλων.

Όσον αφορά λοιπόν τον τοπικό και τον cloud server θα πρέπει το άθροισμα του χρόνου αποστολής των δεδομένων και του χρόνου πρόβλεψης των μοντέλων για κάθε παράθυρο να είναι μικρότερο από το χρόνο που απαιτείται ώστε να συμπληρωθούν τα σημεία του επόμενου παραθύρου. Αντιθέτως, για τους επιταχυντές AGX και NX όπου δεν πραγματοποιείται κάποια αποστολή των δεδομένων, αρκεί μόνο ο χρόνος πρόβλεψης των μοντέλων να είναι μικρότερος. Τα αποτελέσματα του συνολικού χρόνου απόκρισης των 4 υπολογιστικών πόρων παρουσιάζονται ακολουθώντας.

Χρόνοι Συμπερασματολογίας Τοπικού Server

Αρχικά λοιπόν εξετάζεται ο τοπικός server και παρουσιάζονται οι χρόνοι μεταφοράς και πρόβλεψης 5 τυχαίων παραθύρων, όπως επίσης οι μέσοι χρόνοι όλων των παραθύρων και για τα 50 σήματα μακράς διάρκειας. Τα αποτελέσματα αυτά παρουσιάζονται στον Πίν. 8.1:

Πίνακας 8.1: Χρόνοι συμπερασματολογίας παραθύρων με χρήση του τοπικού server.

| A/A Παράθυρο | Ταχ. Transfer (kB/sec) | Μέγεθος (kB) | Edge Offloading (msec) | Edge Computing (msec) |
|---------------------|---------------------------|-----------------|---------------------------|--------------------------|
| 1 | 364.7 | 0.294 | 0.81 | 15.0 |
| 2 | 276.4 | 0.330 | 1.19 | 15.1 |
| 3 | 331.3 | 0.320 | 0.97 | 15.1 |
| 4 | 255.1 | 0.329 | 1.29 | 15.3 |
| 5 | 236.8 | 0.367 | 1.55 | 16.7 |
| Μέσος Χρόνος | - | - | 1.27 | 15.5 |

Σύμφωνα με τα παραπάνω αποτελέσματα, ο μέσος χρόνος μεταφοράς ενός παραθύρου στον τοπικό server ανέρχεται στα 1.27 msecs, ενώ ο αντίστοιχος μέσος χρόνος πρόβλεψης των μοντέλων είναι περίπου 12 φορές μεγαλύτερος και ισούται με 15.5 msecs. Αυτό σημαίνει ότι για ένα τυχαίο παράθυρο, ο συνολικός χρόνος απόκρισης του τοπικού server ανέρχεται στα 16.76 msecs. Ο χρόνος αυτός είναι αρκετά μικρότερος από το χρόνο που απαιτείται ώστε να συμπληρωθούν τα 80 σημεία του επόμενου παραθύρου. Αυτό συμβαίνει διότι όπως προαναφέρθηκε απαιτούνται 4 επιπλέον σημεία για τη συμπλήρωση του επόμενου παραθύρου, ήτοι $4 \text{ σημεία}/15 \text{ Hz} = 267 \text{ msecs}$, εφόσον η συχνότητα δειγματοληψίας ισούται με 15 Hz. Συνεπώς, ο τοπικός server ενδείκνυται για την εκτέλεση της συγκεκριμένης εφαρμογής σε πραγματικό χρόνο.

Χρόνοι Συμπερασματολογίας Cloud Server

Κατά αντιστοιχία με τον τοπικό server, ακολούθως παρουσιάζονται τα ίδια αποτελέσματα από τη χρήση του cloud server. Οι χρόνοι λοιπόν μεταφοράς και συμπερασματολογίας 5 τυχαίων παραθύρων, όπως επίσης οι μέσοι χρόνοι όλων των παραθύρων και για τα 50 σήματα μακράς διάρκειας παρουσιάζονται στον Πίν. 8.2:

Πίνακας 8.2: Χρόνοι συμπερασματολογίας παραθύρων με χρήση του cloud server.

| A/A Παράθυρο | Ταχ. Transfer (kB/sec) | Μέγεθος (kB) | Cloud Offloading (msec) | Cloud Computing (msec) |
|---------------------|---------------------------|-----------------|----------------------------|---------------------------|
| 1 | 27.1 | 0.294 | 10.8 | 14.4 |
| 2 | 17.1 | 0.330 | 19.3 | 14.9 |
| 3 | 16.4 | 0.320 | 19.5 | 14.9 |
| 4 | 17.8 | 0.329 | 18.0 | 14.7 |
| 5 | 10.6 | 0.367 | 34.7 | 15.7 |
| Μέσος Χρόνος | - | - | 21.84 | 15.0 |

Σύμφωνα με τον Πίν. 8.2, ο μέσος χρόνος μεταφοράς ενός παραθύρου στον cloud server ανέρχεται στα 21.84 msecs, δηλαδή είναι περίπου 17 φορές μεγαλύτερος από τον αντίστοιχο χρόνο μεταφοράς στον τοπικό server. Το φαινόμενο αυτό είναι αναμενόμενο, αφού ο τοπικός server είναι μία edge device η οποία βρίσκεται πλησιέστερα στα δεδομένα συγκριτικά με τον cloud server. Αντίστοιχα, ο μέσος χρόνος συμπερασματολογίας ισούται με 15 msecs, δηλαδή είναι ελάχιστα μικρότερος από το χρόνο συμπερασματολογίας του τοπικού server. Κατά συνέπεια, ο συνολικός χρόνος απόκρισης για ένα τυχαίο παράθυρο στον cloud server ανέρχεται στα 36.81 msecs και είναι διπλάσιος από τα 16.76 msecs του τοπικού server. Παρόλα αυτά όμως θεωρείται ξανά αρκετά μικρότερος από το χρόνο που απαιτείται ώστε να συμπληρωθούν τα 80 σημεία του επόμενου παραθύρου (εφόσον $36.81 \text{ msecs} < 267 \text{ msecs}$) και κατά συνέπεια ο cloud server μπορεί να χρησιμοποιηθεί για συμπερασματολογία σε πραγματικό χρόνο.

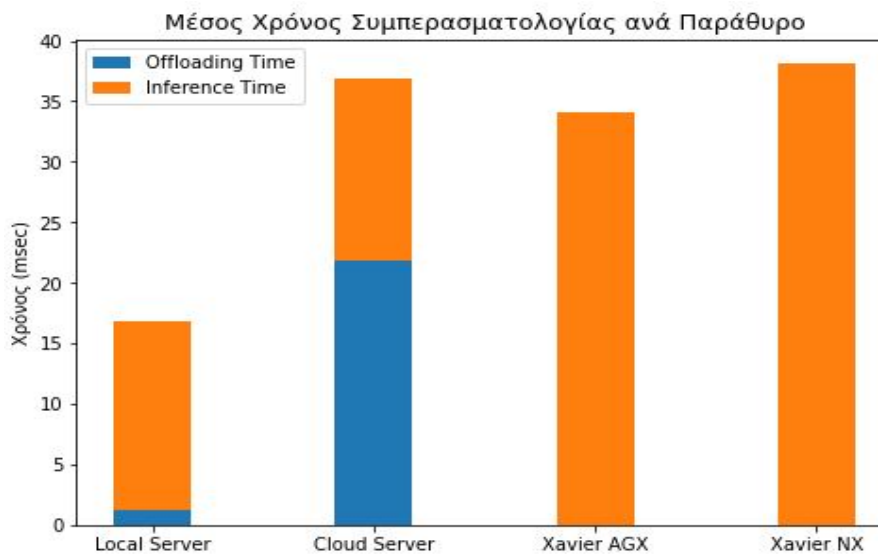
Χρόνοι Συμπερασματολογίας AGX και NX

Τέλος, παρουσιάζονται τα ίδια αποτελέσματα από τη χρήση των ενσωματωμένων συστημάτων AGX και NX. Θα πρέπει να σημειωθεί ότι τα ολοκληρωμένα αυτά κυκλώματα δεν απαιτούν τη μεταφορά των δεδομένων, καθώς είναι ενσωματωμένα στη συσκευή καταγραφής των σημάτων. Τα αποτελέσματα αυτά παρουσιάζονται στον Πίν. 8.3, βάσει του οποίου προκύπτει ότι κατά μέσο όρο ο χρόνος συμπερασματολογίας ανά παράθυρο ανέρχεται στα 34.1 msec για τον AGX και 38.2 msec για τον NX, αντίστοιχα.

Πίνακας 8.3: Χρόνοι συμπερασματολογίας παραθύρων με χρήση των AGX και NX.

| A/A Παραθύρου | AGX Edge Computing (msec) | NX Edge Computing (msec) |
|---------------------|------------------------------|-----------------------------|
| 1 | 33.4 | 36.2 |
| 2 | 33.2 | 39.2 |
| 3 | 32.2 | 37.1 |
| 4 | 32.4 | 38.1 |
| 5 | 37.5 | 43.2 |
| Μέσος Χρόνος | 34.1 | 38.2 |

Συγκεντρωτικά λοιπόν, τα παραπάνω αποτελέσματα των 4 υπολογιστικών πόρων συνοψίζονται στο Σχ. 8.4, όπου για κάθε περίπτωση απεικονίζεται τόσο ο χρόνος μεταφοράς των δεδομένων (offloading time) όσο και ο χρόνος πρόβλεψης των μοντέλων (inference time). Συγκεκριμένα, ο μικρότερος συνολικός χρόνος απόκρισης παρατηρείται στον local server, για τον οποίο οι χρόνοι μεταφοράς δεδομένων και πρόβλεψης είναι αρκετά μικροί. Όσον αφορά τον cloud server ο συνολικός χρόνος απόκρισης είναι μεγαλύτερος, καθώς παρόλο που ο χρόνος πρόβλεψης των μοντέλων είναι ελάχιστα μικρότερος από αυτόν του τοπικού server, η μεταφορά των δεδομένων στο cloud καθυστερεί αρκετά. Όσον αφορά τους επιταχυντές AGX και NX παρατηρείται ότι δε διαφέρουν αισθητά μεταξύ τους και από το συνολικό χρόνο απόκρισης του cloud server, χωρίς όμως να απαιτείται για τις συσκευές αυτές μεταφορά δεδομένων. Τέλος, να σημειωθεί ότι όλοι οι χρόνοι των 4 υπολογιστικών πόρων θεωρούνται ικανοποιητικοί, καθώς σε όλες τις περιπτώσεις η συνολική διάρκεια απόκρισης είναι μικρότερη των 267 msec που απαιτούνται για να συμπληρωθούν τα σημεία του επόμενου χρονικά παραθύρου.

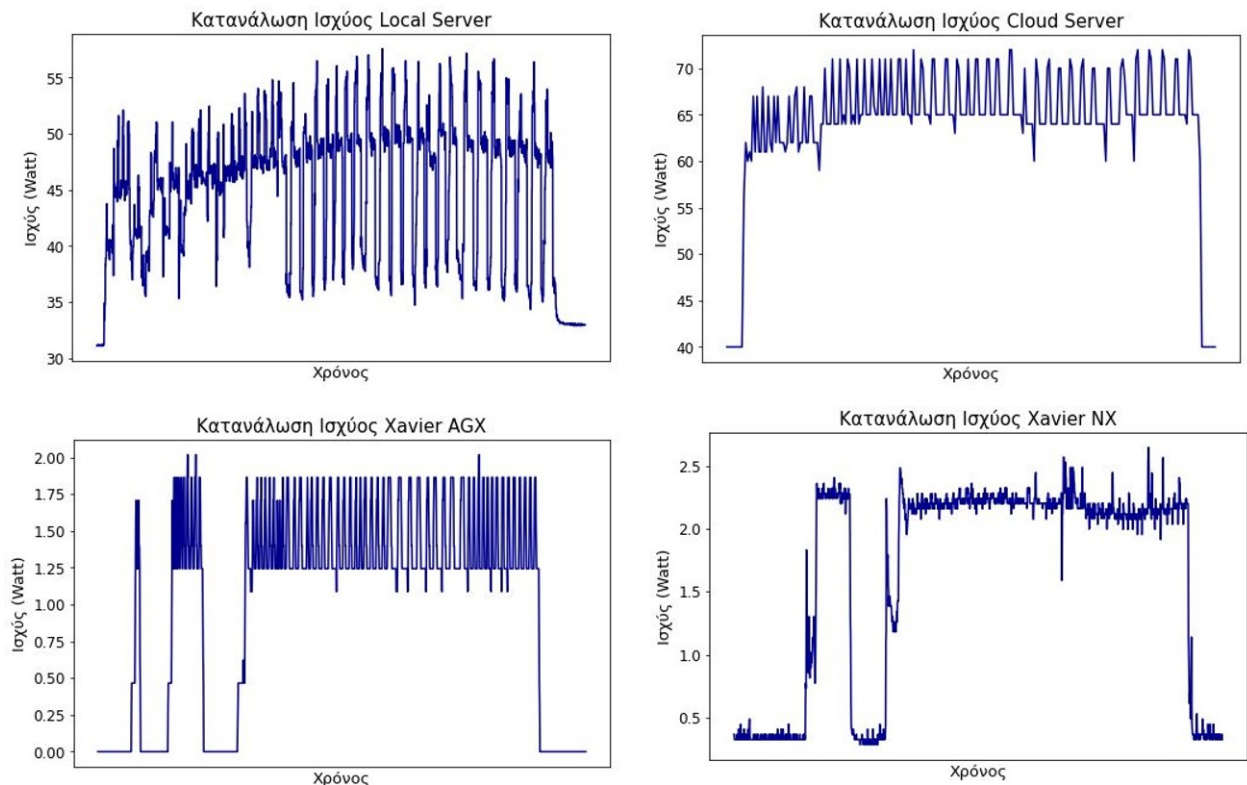


Σχήμα 8.2: Σύγκριση χρόνων συμπερασματολογίας ανά παράθυρο για τους 4 υπολογιστικούς πόρους.

8.3.2 Κατανάλωση Ισχύος Αρχιτεκτονικών

Εκτός του συνολικού χρόνου απόκρισης των 4 υπολογιστικών πόρων, ακολούθως εξετάζεται η ισχύς (σε Watt) που καταναλώνουν όταν σε αυτούς τροφοδοτηθούν και τα 50 σήματα μακράς διάρκειας. Όπως έχει ήδη προαναφερθεί, τα ενσωματωμένα συστήματα αναμένεται να οδηγήσουν σε χαμηλότερα επίπεδα ισχύος κατά τη συμπεραματολογία, καθώς είναι πιο ενεργειακά αποδοτικά. Η κατανάλωση ισχύος λοιπόν παρουσιάζεται στο Σχ. 8.3, με ορισμένα συμπεράσματα που μπορούν να εξαχθούν να είναι τα εξής:

- Όσον αφορά τον τοπικό server, η ισχύς ταλαντώνεται με μέση τιμή περίπου τα 47 Watt. Σε λίγο πιο αυξημένες τιμές οδηγεί ο cloud server, για τον οποίο η μέση κατανάλωση ισχύος ανέρχεται περίπου στα 65 Watt. Το φαινόμενο αυτό είναι αναμενόμενο, αφού η κάρτα γραφικών Tesla V100 του cloud server είναι πιο προηγμένη από την Tesla T4 του τοπικού server, με αποτέλεσμα να οδηγεί σε ταχύτερη πρόβλεψη των μοντέλων (δες Σχ. 8.4) και κατά συνέπεια να καταναλώνει περισσότερη ισχύ.
- Αντιθέτως, η ισχύς που καταναλώνουν τα ολοκληρωμένα κυκλώματα Xavier AGX και NX κυμαίνεται σε αρκετά χαμηλότερα επίπεδα, με κατά μέσο όρο τις τιμές τους να ανέρχονται σε 1.5 Watt για τον AGX και 2.2 Watt για τον NX, αντίστοιχα. Αυτό σημαίνει ότι η χρήση των ενσωματωμένων συστημάτων οδηγεί σε τουλάχιστον 95% μείωση της κατανάλωσης ισχύος συγκριτικά με τον τοπικό και τον cloud server. Το πλεονέκτημα αυτό, σε συνδυασμό με τη μη αναγκαιότητα αποστολής των δεδομένων, αποτελούν χαρακτηριστικά γνωρίσματα των ενσωματωμένων συστημάτων, τα οποία σχεδιάζονται με τέτοιο τρόπο ώστε η πρόβλεψη των μοντέλων να πραγματοποιείται σε σύντομο χρονικό διάστημα και χωρίς να καταναλώνονται μεγάλα ποσά ενέργειας.



Σχήμα 8.3: Σύγκριση κατανάλωσης ισχύος (σε Watt) μεταξύ των 4 υπολογιστικών πόρων και για τα 50 σήματα μακράς διάρκειας.

8.4 Βελτιστοποιήσεις για Συμπερασματολογία σε Πραγματικό Χρόνο

Τα τεχνητά νευρωνικά δίκτυα που εκπαιδεύονται στα πλαίσια διαφόρων εφαρμογών είναι συνήθως μοντέλα με μεγαλύτερο αριθμό παραμέτρων (over-parametrized models) από αυτόν που συνήθως απαιτείται, τα οποία προφανώς είναι δύσκολα διαχειρίσιμα. Για το λόγο αυτό, η συμπίεση τέτοιων μοντέλων μειώνοντας τον αριθμό των παραμέτρων τους αποτελεί αντικείμενο μελέτης πολλών ερευνητών το τελευταίο χρονικό διάστημα. Συγκεκριμένα, μειώνοντας τον αριθμό των παραμέτρων του μοντέλου, αυτό στη συνέχεια ισοδυναμεί με ένα νευρωνικό δίκτυο με αραιή συνδεσιμότητα (sparse connectivity) των κόμβων που περιλαμβάνει. Το γεγονός αυτό οδηγεί σε αρκετά πλεονεκτήματα, ορισμένα από τα οποία απαριθμούνται ακολούθως:

1. Ο απαιτούμενος αποθηκευτικός χώρος του μοντέλου είναι μικρότερος, εφόσον αποθηκεύεται μικρότερο πλήθος παραμέτρων.
2. Αν η απομάκρυνση των παραμέτρων πραγματοποιηθεί με εξειδικευμένο τρόπο, τότε μπορεί να μειωθεί το πλήθος των πράξεων που απαιτούνται και κατά συνέπεια να μειωθεί παράλληλα η κατανάλωση ισχύος.
3. Η απόδοση των νευρωνικών δικτύων μπορεί να παραμείνει στα ίδια επίπεδα με τα αρχικά μοντέλα ή έστω να μειωθεί ελάχιστα, παρόλο που χρησιμοποιείται μικρότερος αριθμός παραμέτρων.

Στην υποενότητα που ακολουθεί λοιπόν παρουσιάζονται ορισμένες μέθοδοι μείωσης των παραμέτρων των νευρωνικών δικτύων LSTM και στη συνέχεια προσαρμόζεται η καταλληλότερη από αυτές στα εκπαιδευμένα δίκτυα που παρουσιάστηκαν στο Κεφάλαιο 7.

8.4.1 Μέθοδοι Pruning Μοντέλων LSTM

Στην υποενότητα αυτή παρουσιάζονται ορισμένες μέθοδοι μείωσης (pruning) των παραμέτρων των νευρωνικών δικτύων LSTM, τα οποία αποτελούν τη βάση των μοντέλων που χρησιμοποιούνται στην παρούσα εργασία. Σύμφωνα με την υποενότητα 2.2.1, σε κάθε δίκτυο LSTM χρησιμοποιούνται 8 διαφορετικοί πίνακες παραμέτρων \mathbf{W} και 8 διανύσματα μεροληψίας \mathbf{b} , όπως παρουσιάζονται ακολούθως:

- Πύλη forget: Πίνακες \mathbf{W}_{if} , \mathbf{W}_{hf} και διανύσματα \mathbf{b}_{if} , \mathbf{b}_{hf} (δες Εξ. (2.17))
- Πύλη input: Πίνακες \mathbf{W}_{ii} , \mathbf{W}_{hi} και διανύσματα \mathbf{b}_{ii} , \mathbf{b}_{hi} (δες Εξ. (2.18))
- Κεντρικό κελί: Πίνακες $\mathbf{W}_{i\bar{c}}$, $\mathbf{W}_{h\bar{c}}$ και διανύσματα $\mathbf{b}_{i\bar{c}}$, $\mathbf{b}_{h\bar{c}}$ (δες Εξ. (2.19))
- Πύλη output: Πίνακες \mathbf{W}_{io} , \mathbf{W}_{ho} και διανύσματα \mathbf{b}_{io} , \mathbf{b}_{ho} (δες Εξ. (2.20))

Για να μειωθεί το πλήθος των παραμέτρων των δικτύων LSTM, είναι προφανές ότι θα πρέπει να μηδενιστούν ορισμένα στοιχεία των 8 πινάκων \mathbf{W} και των 8 διανυσμάτων μεροληψίας \mathbf{b} . Εφόσον στην παρούσα εργασία η βιβλιοθήκη που χρησιμοποιείται για την εκπαίδευση και τη συμπερασματολογία των νευρωνικών δικτύων είναι το pytorch, ακολούθως παρουσιάζονται ορισμένες τεχνικές pruning που προτείνονται από τη βιβλιοθήκη αυτή και οδηγούν στο μηδενισμό των παραμέτρων των νευρωνικών δικτύων:

1. Συνάρτηση `prune.random_unstructured`: με την τεχνική αυτή ουσιαστικά μηδενίζονται ορισμένα στοιχεία ενός πίνακα ή ενός διανύσματος μεροληψίας με τυχαίο τρόπο, αφού πρώτα όμως ο χρήστης επιλέξει το ποσοστό μηδενισμού των παραμέτρων.

2. Συνάρτηση `prune.random_structured`: η τεχνική αυτή παρομοιάζεται με την 1η συνάρτηση, με τη διαφορά όμως ότι ο μηδενισμός των παραμέτρων δε γίνεται με τυχαίο τρόπο. Η συγκεκριμένη συνάρτηση χρησιμοποιείται στα συνελικτικά δίκτυα με σκοπό το μηδενισμό ολόκληρων καναλιών (channels), με αποτέλεσμα να μειώνονται παράλληλα και οι πράξεις εντός του δικτύου και συνεπώς η υπολογιστική ισχύς που απαιτείται.
3. Συνάρτηση `prune.l1_unstructured`: με την τεχνική αυτή ουσιαστικά μηδενίζονται το ποσοστό των στοιχείων ενός πίνακα ή ενός διανύσματος που επιλέγει ο χρήστης βάσει της χαμηλότερης L1-νόρμας τους.
4. Συνάρτηση `prune.ln_structured`: η τεχνική αυτή παρομοιάζεται με την 3η συνάρτηση, με τη διαφορά όμως ότι ο μηδενισμός πραγματοποιείται σε ολόκληρα κανάλια συνελικτικών δικτύων ανάλογα με τη μικρότερη L_n -νόρμα τους.

Σύμφωνα με τις παραπάνω συναρτήσεις της βιβλιοθήκης `pytorch`, οι τεχνικές 2 και 4 αναφέρονται σε δομημένη (structured) μείωση παραμέτρων και συνεπώς δεν μπορούν να χρησιμοποιηθούν με τα νευρωνικά δίκτυα LSTM, καθώς αφορούν κανάλια συνελικτικών δικτύων. Αντιθέτως, οι συναρτήσεις 1 και 3 της μη δομημένης (unstructured) μείωσης παραμέτρων μπορούν να εφαρμοσθούν στους πίνακες \mathbf{W} και στα διανύσματα μεροληψίας \mathbf{b} , ώστε να διερευνηθεί κατά πόσο αυτές οι μέθοδοι pruning μπορούν να επηρεάσουν τα τελικά αποτελέσματα των δικτύων LSTM που παρουσιάστηκαν στο Κεφάλαιο 7.

8.4.2 Αποτελέσματα Μεθόδων Pruning

Στην παρούσα υποενότητα παρουσιάζονται τα αποτελέσματα της μείωσης των παραμέτρων των δικτύων LSTM, χρησιμοποιώντας μία από τις τεχνικές που παρουσιάστηκαν στην υποενότητα 8.4.1. Όπως προαναφέρθηκε, με χρήση της βιβλιοθήκης `pytorch` οι μοναδικές τεχνικές pruning που μπορούν να προσαρμοστούν στους πίνακες \mathbf{W} και στα διανύσματα μεροληψίας \mathbf{b} των δικτύων LSTM είναι οι συναρτήσεις της μη δομημένης (unstructured) μείωσης παραμέτρων.

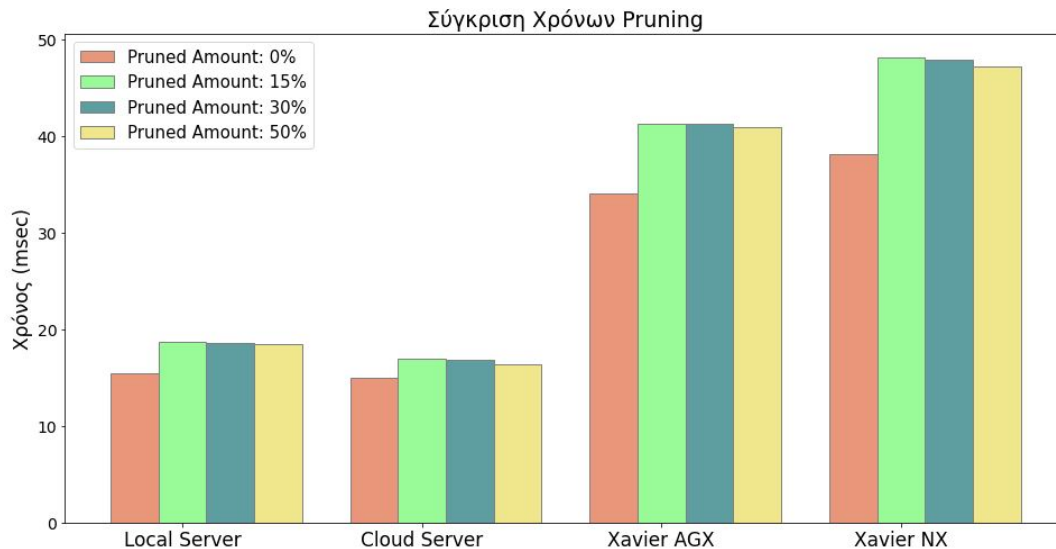
Στην παρούσα εργασία επιλέγεται να μελετηθεί μόνο η συνάρτηση `prune.l1_unstructured`, με χρήση της οποίας μηδενίζονται το ποσοστό των στοιχείων ενός πίνακα ή ενός διανύσματος που επιλέγει ο χρήστης βάσει της χαμηλότερης L1-νόρμας τους. Η μελέτη της συνάρτησης `prune.l1_unstructured` κι όχι η `prune.random_unstructured` οφείλεται στο γεγονός ότι σε μοντέλα χρονοσειρών, η οποιαδήποτε μεταβολή σε τυχαίες παραμέτρους μπορεί να τροποποιήσει αισθητά τα αποτελέσματα, καθώς χάνεται η μοντελοποίηση των χρονικών εξαρτήσεων. Αντιθέτως, αν ο μηδενισμός πραγματοποιηθεί σε παραμέτρους με πολύ χαμηλές τιμές, οι μεταβολές που θα προκύψουν μπορεί να μην είναι τόσο αισθητές.

Ακολούθως λοιπόν η συνάρτηση `prune.l1_unstructured` προσαρμόζεται σε όλα τα δίκτυα LSTM των σημάτων κινήσεων και συνιστωσών κινήσεων του Κεφαλαίου 7 και τα αποτελέσματα που προκύπτουν αναγράφονται στον Πίν. 8.4. Συγκεκριμένα, εξετάζονται διάφορα ποσοστά pruning των παραμέτρων και ελέγχονται: (α) ο χρόνος που απαιτείται για την πρόβλεψη των μοντέλων και (β) η μείωση της ακρίβειας των μοντέλων.

Πίνακας 8.4: Αποτελέσματα μείωσης παραμέτρων των δικτύων LSTM του Κεφαλαίου 7.

| Ποσοστό Pruned Παραμέτρων (%) | Ακρίβεια (%) | Μέσοι Χρόνοι ανά Παράθυρο (msecs) | | | |
|----------------------------------|-----------------|-----------------------------------|-------|-------|-------|
| | | Local | Cloud | AGX | NX |
| 0 | 81.42 | 15.50 | 15.00 | 34.11 | 38.20 |
| 15 | 81.41 | 18.75 | 16.93 | 41.26 | 48.21 |
| 30 | 81.08 | 18.60 | 16.87 | 41.28 | 47.96 |
| 50 | 72.68 | 18.45 | 16.40 | 40.93 | 47.24 |

Σύμφωνα με τα αποτελέσματα του Πίν. 8.4, αρχικά παρατηρείται ότι αύξηση του ποσοστού μηδενισμού των παραμέτρων οδηγεί σε μείωση της ακρίβειας των μοντέλων. Αυτό γίνεται πιο αισθητό στην περίπτωση που μηδενιστούν οι μισές παράμετροι των δικτύων LSTM (ποσοστό pruned ίσο με 50%), όπου η ακρίβεια μειώνεται σε 72.68% από την αρχική τιμή 81.42% του μη τροποποιημένου δικτύου. Επιπλέον, όσον αφορά το χρόνο πρόβλεψης των μοντέλων, παρατηρείται ότι παρόλη τη μείωση του αριθμού των παραμέτρων ο χρόνος αυτός δε μειώνεται καθόλου. Αυτό οφείλεται στο γεγονός ότι με τη συγκεκριμένη μέθοδο pruning ουσιαστικά δεν απομακρύνονται τελείως οι παράμετροι του μοντέλου, αλλά αντιθέτως μηδενίζονται. Κατά συνέπεια, οι αλγεβρικές πράξεις των πινάκων πραγματοποιούνται κανονικά και δεν παρατηρείται μείωση στο χρόνο συμπερασματολογίας. Τα αποτελέσματα αυτά απεικονίζονται επίσης στο Σχ. 8.4.



Σχήμα 8.4: Σύγκριση μέσων χρόνων συμπερασματολογίας ανά παράθυρο για διάφορα ποσοστά μείωσης των παραμέτρων.

Τέλος, θα πρέπει να σημειωθεί ότι ένα πλεονέκτημα των μεθόδων pruning αποτελεί ο μικρότερος αποθηκευτικός χώρος των μοντέλων. Αυτό είναι ιδιαίτερα σημαντικό κυρίως για τα ενσωματωμένα συστήματα, η μνήμη των οποίων είναι στις περισσότερες περιπτώσεις μικρότερη από έναν κεντρικό ή έναν cloud server και κατά συνέπεια ενδέχεται να μην μπορούν διαχειριστούν εξαιρετικά μεγάλα μοντέλα κατά τη διαδικασία της συμπερασματολογίας.

8.5 Μελλοντικοί στόχοι

8.5.1 Υλοποίηση Δικτύων σε FPGA

Ένας εναλλακτικός υπολογιστικός πόρος για τη μελέτη της απόκρισης των δικτύων αποτελούν οι συστοιχίες πυλών προγραμματιζόμενου πεδίου (FPGAs). Οι συσκευές αυτές ουσιαστικά είναι ένας τύπος προγραμματιζόμενου ολοκληρωμένου κυκλώματος γενικής χρήσης το οποίο διαθέτει πολύ μεγάλο αριθμό τυποποιημένων πυλών και άλλων ψηφιακών λειτουργιών όπως απαριθμητές, καταχωρητές μνήμης, γεννήτριες PLL κ.α. Χαρακτηριστικό γνώρισμα των FPGAs είναι ότι μπορεί να επιτύχουν πολύ χαμηλούς χρόνους συμπερασματολογίας και κατά συνέπεια να χρησιμοποιηθούν σε εφαρμογές που απαιτούν απόκριση σε πραγματικό χρόνο. Ωστόσο, η υλοποίηση των δικτύων που έχουν χρησιμοποιηθεί έως τώρα δεν είναι εφικτή σε FPGA, καθώς στην επίσημη βιβλιογραφία τα δίκτυα LSTM δεν έχουν προγραμματιστεί ακόμα στις γλώσσες περιγραφής υλικού VHDL ή Verilog, με χρήση των οποίων προγραμματίζονται οι συσκευές αυτές. Για το λόγο αυτό, η συμπερασματολογία των δικτύων σε FPGAs τύπου π.χ. Alveo U200 ή U250 αποτελεί αντικείμενο μίας μελλοντικής εργασίας.

8.5.2 Quantization και Approximate Computing

Τέλος, η προτεινόμενη μεθοδολογία μπορεί να πραγματοποιηθεί χρησιμοποιώντας 2 καινοτόμους τρόπους συμπερασματολογίας, δηλαδή την τεχνική quantization και την approximate computing. Ο όρος quantization αναφέρεται σε τεχνικές για υπολογισμούς και αποθήκευση τανυστών με χρήση λιγότερων bits από την ακρίβεια κινητής υποδιαστολής (floating point precision) [72]. Αυτό επιτρέπει μια πιο συμπαγή αναπαράσταση μοντέλου, επιτυγχάνοντας ταυτόχρονα υψηλές αποδόσεις σε πολλές πλατφόρμες υλικού. Το PyTorch υποστηρίζει την τεχνική quantization με χρήση ακεραίων INT8 σε σύγκριση με τα τυπικά μοντέλα που είναι αποθηκευμένα με ακρίβεια FP32, γεγονός που επιτρέπει τη μείωση επί 4 στο μέγεθος του μοντέλου και κατά συνέπεια μειώνεται επί 4 και ο απαιτούμενος αποθηκευτικός χώρος του μοντέλου. Επιπλέον, οι υπολογισμοί με INT8 είναι συνήθως 2 έως 4 φορές ταχύτερη σε σύγκριση με τον υπολογισμό FP32 και κατά συνέπεια η συμπερασματολογία επιτυγχάνεται ταχύτερα.

Τέλος, η τεχνική approximate computing χρησιμοποιείται στις περιπτώσεις εκείνες που δε μας ενδιαφέρει η πρόβλεψη των μοντέλων να είναι εξαιρετικά ακριβής, αλλά ταυτόχρονα τα συνολικά αποτελέσματα να μην τροποποιούνται αισθητά [73]. Χαρακτηριστικό παράδειγμα αποτελούν τα προβλήματα ταξινόμησης των νευρωνικών δικτύων, στα οποία μπορούμε να θυσιάσουμε ένα μικρό ποσοστό ακρίβειας των προβλέψεων πραγματοποιώντας κατά προσέγγιση υπολογισμούς. Με τον τρόπο αυτό μειώνεται αισθητά ο χρόνος που απαιτείται για τις προβλέψεις των μοντέλων και κατά συνέπεια η συμπερασματολογία ολοκληρώνεται ταχύτερα. Ορισμένες τεχνικές κατά προσέγγισης υπολογισμών αφορούν τη μείωση του αριθμού των επαναλήψεων, διατήρηση λιγότερων δεκαδικών ψηφίων κατά τους υπολογισμούς, απομάκρυνση περιττών παραμέτρων, χρησιμοποίηση πιο γρήγορων αλγορίθμων με ελάχιστη μείωση στην απόδοση (π.χ. αλγόριθμοι clustering αντί για πολύπλοκα μοντέλα) κ.α. Τέλος, με τους κατά προσέγγιση υπολογισμούς μειώνεται επίσης η ενέργεια που καταναλώνεται και για το λόγο αυτό χρησιμοποιείται ταυτόχρονα με την τεχνική quantization για μείωση της κατανάλωσης ισχύος των υπολογιστικών πόρων.

Κεφάλαιο 9

Επίλογος

Το Κεφάλαιο αυτό είναι το τελευταίο της παρούσας εργασίας και έχει ως στόχο να συνοψίσει τα αποτελέσματα που έχουν προκύψει έως τώρα. Συγκεκριμένα στην Εν. 9.1 παρουσιάζονται ορισμένα συμπεράσματα από την προτεινόμενη μεθοδολογία και από τα αποτελέσματα που προέκυψαν στα προηγούμενα Κεφάλαια, υπενθυμίζοντας στον αναγνώστη τα οφέλη και τις δυσκολίες κατά την επεξεργασία των σημάτων από τα πειράματα των 2 χειριστών. Τέλος, στην Εν. 9.2 προτείνονται ορισμένες ιδέες και πιθανές επεκτάσεις βάσει των έως τώρα αποτελεσμάτων. Οι επεκτάσεις αυτές αφορούν τόσο τις τεχνικές βαθιάς μηχανικής μάθησης, όσο και τον τρόπο που μπορεί να μελετήσει κάποιος σήματα μακράς διάρκειας που περιλαμβάνουν κινήσεις ενδιαφέροντος.

9.1 Γενικά Συμπεράσματα

Η παρούσα εργασία ασχολήθηκε με την αναγνώριση των ανθρώπινων δραστηριοτήτων (Human Activity Recognition - HAR), μελετώντας κινήσεις που πραγματοποιούνται από εργαζόμενους στο επίπεδο παραγωγής. Για το σκοπό αυτό, 2 χειριστές με τη βοήθεια μίας συσκευής καταγραφής σημάτων εκτέλεσαν 625 πειράματα από 5 διαφορετικές κινήσεις, οι οποίες δεν είναι εύκολα διαχωρίσιμες μεταξύ τους μέσω της μη παραμετρικής ανάλυσης των σημάτων. Για το λόγο αυτό επιστρατεύθηκαν τεχνικές βαθιάς μηχανικής μάθησης με σκοπό όχι μόνο το διαχωρισμό των κινήσεων αλλά και την αποτελεσματική ταξινόμησή τους.

Αρχικά λοιπόν εξετάστηκαν τα νευρωνικά δίκτυα αυτοκωδικοποιητών, τα οποία χρησιμοποιούνται για την συμπίεση των σημάτων και την εξαγωγή κατάλληλων χαρακτηριστικών. Στη συνέχεια, με τη βοήθεια των χαρακτηριστικών αυτών εκπαιδεύτηκαν νευρωνικά δίκτυα χρονοσειρών LSTM, με σκοπό την επιτυχή ταξινόμηση των 5 εξεταζόμενων κινήσεων. Τέλος, αντικείμενο της εργασίας αυτής αποτέλεσαν τα σήματα μακράς διάρκειας (ΣΜΔ). Τα σήματα αυτά είναι χρονοσειρές πολύ μεγαλύτερης διάρκειας οι οποίες περιλαμβάνουν τουλάχιστον μία από τις 5 επιθυμητές δραστηριότητες, αφετέρου όμως περιλαμβάνουν και θόρυβο είτε λόγω της εναλλαγής της κίνησης είτε λόγω της ύπαρξης χρονικών περιόδων στις οποίες ο χειριστής δεν πραγματοποιεί κάποια από τις 5 δραστηριότητες.

Τα συμπεράσματα λοιπόν που μπορούν να εξαχθούν από όλη την ανάλυση που προηγήθηκε συνοψίζονται ακολούθως:

1. Όσον αφορά την εξαγωγή χαρακτηριστικών από σήματα χρονοσειρών, οι υψηλότερες επιδόσεις επιτυγχάνονται από τα δίκτυα HF-V-LSTM-AE. Αυτό οφείλεται στους εξής 2 λόγους: (1) εκτιμάται μία κατανομή με πλήρη πίνακα συνδιασπορών στο μεσαίο επίπεδο με χρήση των κανονικοποιημένων ροών και (2) χρησιμοποιούνται δίκτυα LSTM τόσο στον κωδικοποιητή όσο και στον αποκωδικοποιητή. Συνεπώς, με τα συγκεκριμένα δίκτυα αφενός μεν ερμηνεύονται αποτελεσματικότερα οι συχτίσεις των συμπιεσμένων δεδομένων και αφετέρου τονίζεται η χρήση δικτύων LSTM στις περιπτώσεις όπου μελετώνται χρονοσειρές.
2. Όστερα από τα δίκτυα HF-V-LSTM-AE, αρκετά υψηλές αποδόσεις επιτυγχάνονται από τα δίκτυα V-LSTM-AE και VL-LSTM/MLP-AE. Στην πρώτη περίπτωση χρησιμοποιούνται δίκτυα LSTM στο τμήμα του αποκωδικοποιητή, αλλά εκτιμάται ένας διαγώνιος πίνακας συνδιασπορών στο μεσαίο επίπεδο ο οποίος δεν ερμηνεύει τις συσχετίσεις των συμπιεσμένων δεδομένων. Αντιθέτως, τα δίκτυα VL-LSTM/MLP-AE εκτιμούν έναν πλήρη πίνακα συνδιασπορών, αλλά ταυτόχρονα χρησιμοποιούν δίκτυα MLP στο τμήμα του αποκωδικοποιητή έτσι ώστε να εφαρμοστεί η Λαπλασιανή προσέγγιση. Αυτοί λοιπόν είναι οι λόγοι για τους οποίους τα δίκτυα αυτά παρουσιάζουν ελάχιστα μικρότερη απόδοση από τα δίκτυα HF-V-LSTM-AE.
3. Τα δίκτυα SAV-LSTM/MLP-AE παρουσιάζουν χαμηλότερες αποδόσεις από τα δίκτυα των συμπερασμάτων 1 και 2. Αυτό οφείλεται στο γεγονός ότι στα μοντέλα αυτά δε χρησιμοποιούν δίκτυα LSTM στο τμήμα του αποκωδικοποιητή και ταυτόχρονα εκτιμάται ένας διαγώνιος πίνακας συνδιασπορών στο μεσαίο επίπεδο. Βέβαια, οι αποδόσεις των μοντέλων αυτών είναι υψηλότερες από τα απλά δίκτυα V-LSTM/MLP-AE, καθώς με τη βοήθεια της semi-amortized variational συμπερασματολογίας μειώνεται το amortization σφάλμα που προκύπτει από το τμήμα του κωδικοποιητή. Παρόλα αυτά, τα συγκεκριμένα μοντέλα δεν ενδείκνυνται για την εξαγωγή χαρακτηριστικών από σήματα χρονοσειρών.
4. Όσον αφορά το στάδιο ταξινόμησης των κινήσεων και συνιστωσών κινήσεων, τα ποσοστά που επιτυγχάνονται είναι εξαιρετικά υψηλά, παρόλο που οι 5 εξεταζόμενες δραστηριότητες δεν είναι εύκολα διαχωρίσιμες. Συγκεκριμένα, για τα σήματα των 5 κινήσεων το ποσοστό σωστής κατηγοριοποίησης ανέρχεται στο 100%, ενώ για τα σήματα των 15 συνιστωσών κινήσεων στο 97.7%. Βέβαια, τα αποτελέσματα αυτά προκύπτουν αν εξετασθούν απομονωμένα τα σήματα

αυτά, χωρίς δηλαδή να υπάρχουν χρονοσειρές με θόρυβο που δεν αντιστοιχίζονται σε κάποια από τις εξεταζόμενες κινήσεις και συνιστώσες κινήσεις.

5. Τελευταίο κομμάτι μελέτης αποτελούν τα σήματα μακράς διάρκειας (ΣΜΔ), η διάρκεια των οποίων είναι αρκετά μεγαλύτερη από αυτές των κινήσεων και ταυτόχρονα περιέχουν αρκετό θόρυβο ανά ταχτά χρονικά διαστήματα. Για το λόγο αυτό χρησιμοποιήθηκε ένας αλγόριθμος απόφασης με σκοπό το φιλτράρισμα των σημάτων αυτών και την απομάκρυνση των τμημάτων μη ενδιαφέροντος. Ανάλογα την τεχνική αξιολόγησης των αποτελεσμάτων, το ποσοστό σωστής χρονολογικής ταξινόμησης των κινήσεων εντός των εξεταζόμενων ΣΜΔ ανέρχεται σε τιμές από 76% έως και 81%. Οι αποδόσεις αυτές κρίνονται ως αρκετά ικανοποιητικές, αν σκεφτεί κανείς ότι προέκυψαν από μία μόνο συσκευή καταγραφής σημάτων εξετάζοντας δραστηριότητες που δε διαφέρουν αισθητά από άλλες συνήθεις κινήσεις που ενδεχομένως εκτελούν οι εργαζόμενοι.
6. Όσον αφορά τη συμπερασματολογία σε πραγματικό χρόνο, σύμφωνα με τα αποτελέσματα του Κεφαλαίου 8 εξάγεται το συμπέρασμα ότι και οι 4 εξεταζόμενοι υπολογιστικοί πόροι μπορούν να επιτύχουν το στόχο αυτό στο περιβάλλον παραγωγής ενός εργοστασίου. Αυτό αποτελεί ένα ισχυρό πλεονέκτημα της μεθοδολογίας που παρουσιάστηκε, καθώς δίνει τη δυνατότητα της χρήσης εναλλακτικών υπολογιστικών πόρων με παρόμοια αποτελέσματα σε πραγματικό χρόνο.
7. Αν ζητούμενο αποτελεί η χαμηλή κατανάλωση ισχύος χωρίς σημαντική διαφορά στο συνολικό χρόνο απόκρισης, τότε τα ενσωματωμένα συστήματα προτείνονται για χρήση έναντι ενός τοπικού ή ενός cloud server. Να σημειωθεί επίσης ότι τα ενσωματωμένα συστήματα δεν απαιτούν τη μεταφορά των δεδομένων και κατά συνέπεια η ύπαρξη ενός δικτύου δεν αποτελεί προϋπόθεση για την εκτέλεση της μεθοδολογίας στο περιβάλλον παραγωγής.
8. Όσον αφορά τις μεθόδους μείωσης των παραμέτρων των μοντέλων παρατηρείται ότι δεν επιτυγχάνεται ταυτόχρονη μείωση του συνολικού χρόνου απόκρισης. Αυτό οφείλεται στο γεγονός ότι η μέθοδος που επιστρατεύτηκε δεν απομακρύνει τελείως ορισμένες παραμέτρους των μοντέλων LSTM αλλά αντιθέτως τις μηδενίζει και συνεπώς, οι αλγεβρικές πράξεις εντός του δικτύου εκτελούνται κανονικά. Παρόλα αυτά, ένα βασικό πλεονέκτημα των μεθόδων pruning είναι η μείωση της χωρητικότητας που απαιτεί η αποθήκευση των μοντέλων, το οποίο είναι εξαιρετικά σημαντικό κυρίως για τα ενσωματωμένα συστήματα με μικρό αποθηκευτικό χώρο μνήμης.

9.2 Μελλοντική Εργασία

Τέλος, στην παρούσα ενότητα προτείνονται ορισμένες επεκτάσεις της έως τώρα προτεινόμενης μεθοδολογίας. Οι επεκτάσεις αυτές αφορούν τόσο τις τεχνικές βαθιάς μηχανικής μάθησης, όσο και τον τρόπο που μπορεί να μελετήσει κάποιος σήματα μακράς διάρκειας που περιλαμβάνουν κινήσεις ενδιαφέροντος. Οι ιδέες αυτές συνοψίζονται ακολούθως:

1. Η πρώτη ιδέα αφορά τα σήματα καταγραφής, τα οποία θα μπορούσαν να ληφθούν από περισσότερα σημεία του σώματος των εργαζόμενων και με χρήση διαφορετικών συσκευών από αυτή του τηλεφώνου. Για παράδειγμα, η χρήση Myo Armband αποτελεί μία ενδεικτική επιλογή που μπορεί να διευκολύνει τη διεξαγωγή των πειραμάτων, καθώς μπορεί να προσαρτηθεί με μεγαλύτερη ευκολία σε διάφορα σημεία του σώματος. Με αυτόν τον τρόπο ίσως βελτιώνονταν τα αποτελέσματα, καθώς ο συνδυασμός σημάτων προσδίδει επιπλέον πληροφορία κατά τη μοντελοποίηση χωρίς να χρειαστεί να τροποποιηθεί η μεθοδολογία που παρουσιάστηκε.
2. Εφόσον η βέλτιστη εξαγωγή χαρακτηριστικών από τα σήματα χρονοσειρών επιτυγχάνεται με τη βοήθεια κανονικοποιημένων ροών, θα μπορούσαν να μελετηθούν ροές οι οποίες ανήκουν στην κατηγορία των General Normalizing flows και για τις οποίες η ορίζουσα του Ιακωβιανού πίνακα

δεν ισούται με 1. Χαρακτηριστικό παράδειγμα θεωρείται η κανονικοποιημένη Inverse Autoregressive Flow (IAF) [68], η οποία βασίζεται στη χρήση νευρωνικών δικτύων αυτοπαλινδρόμησης (autoregressive neural nets) και όταν συνδυαστεί με τα μοντέλα VAE βελτιώνει εξαιρετικά την εκ των υστέρων κατανομή των συμπιεσμένων δεδομένων.

3. Όσον αφορά τα δίκτυα χρονοσειρών, εναλλακτική προσέγγιση στο συγκεκριμένο πρόβλημα μπορεί ενδέχεται να αποτελέσει η χρήση των Transformers [69]. Τα δίκτυα αυτά χρησιμοποιούνται για τη μοντελοποίηση χρονοσειρών, με τη διαφορά όμως ότι δίνουν έμφαση (attention) σε συγκεκριμένα τμήματα των σημάτων, αγνοώντας ταυτόχρονα τα τμήματα εκείνα τα οποία δεν προσδίδουν καμία πληροφορία στην τελική πρόβλεψη. Κατά συνέπεια, τα νευρωνικά δίκτυα αυτά αποτελούν ενδεχομένως μία κατάλληλη τεχνική μελέτης των σημάτων μακράς διάρκειας, από τα οποία επιθυμούμε να απομονώσουμε τις κινήσεις ενδιαφέροντος μέσα από το φιλτράρισμα και την απομάκρυνση των τμημάτων που περιλαμβάνουν θόρυβο.
4. Όπως προαναφέρθηκε στην Εν. 8.5, ένας εναλλακτικός υπολογιστικός πόρος για τη μελέτη της απόκρισης των δικτύων αποτελούν οι συστοιχίες πυλών προγραμματιζόμενου πεδίου (FPGAs). Εφόσον όμως τα δίκτυα LSTM δεν έχουν προγραμματιστεί ακόμα στις γλώσσες περιγραφής υλικού VHDL ή Verilog, μία πιθανή μελλοντική επέκταση θεωρείται η προσπάθεια υλοποίησης των δικτύων αυτών σε FPGAs (π.χ. Alveo U200 ή U250), ώστε να πραγματοποιηθεί η σύγκριση της απόδοσής τους με τους 4 υπολογιστικούς πόρους που εξετάστηκαν στην παρούσα εργασία. Παράλληλα, μπορούν να εφαρμοσθούν οι τεχνικές quantization και approximate computing, ώστε να μειωθεί περαιτέρω τόσο ο χρόνος συμπερασματολογίας όσο και η υπολογιστική ισχύς.
5. Τέλος, τα εκπαιδευμένα μοντέλα θα μπορούσαν να αξιολογηθούν σε σήματα από μία πραγματική γραμμή παραγωγής. Με αυτόν τον τρόπο θα εξάγονταν ορισμένα χρήσιμα συμπεράσματα για την αποτελεσματικότητα της έως τώρα προτεινόμενης μεθοδολογίας με σκοπό την άμεση συνεργασία χειριστή-ρομπότ.

Βιβλιογραφικές Αναφορές

- [1] M. Al-Amin, W. Tao, D. Doell, R. Lingard, Z. Yin, M. C. Leu and R. Qin, Action recognition in manufacturing assembly using multimodal sensor fusion, *Procedia Manufacturing*, volume 39, pp. 158-167, Elsevier Ltd., 2019. doi:[10.1016/j.promfg.2020.01.288](https://doi.org/10.1016/j.promfg.2020.01.288).
- [2] H. Junker, O. Amft, P. Lukowicz and G. Troster, Gesture spotting with body-worn inertial sensors to detect user activities, *Pattern Recognition*, volume 41, issue 6, pp. 2010–2024, Elsevier Ltd., 2008. doi:[10.1016/j.patcog.2007.11.016](https://doi.org/10.1016/j.patcog.2007.11.016).
- [3] T. T. Ngo, Y. Makihara, H. Nagahara, Y. Mukaigawa and Y. Yagi, Similar gait action recognition using an inertial sensor, *Pattern Recognition*, volume 48, issue 4, pp. 1289–1301, Elsevier Ltd., 2015. doi:[10.1016/j.patcog.2014.10.012](https://doi.org/10.1016/j.patcog.2014.10.012).
- [4] Q. Li, G. Zhou and J. A. Stankovic, Accurate, fast fall detection method using posture and context information, *Proceedings of the 6th ACM International Conference on Embedded Networked Sensor Systems*, pp. 443–444, ACM, 2008. doi:[10.1145/1460412.1460494](https://doi.org/10.1145/1460412.1460494).
- [5] Y. Tang, S. Wang, Y. Chen and Z. Chen, PPCare: a personal and pervasive health care system for the elderly, *9th International Conference on Ubiquitous Intelligence and Computing and 9th International Conference on Autonomic and Trusted Computing*, pp. 935-939, IEEE, 2012. doi:[10.1109/UIC-ATC.2012.60](https://doi.org/10.1109/UIC-ATC.2012.60).
- [6] K. Altun, B. Barshan, Human activity recognition using inertial/magnetic sensor units, *International Workshop on Human Behavior Understanding (HBU)*, pp. 38–51, Springer, 2010. doi:[10.1007/978-3-642-14715-9_5](https://doi.org/10.1007/978-3-642-14715-9_5).
- [7] R. Raghavendra , A. Del Bue, M. Cristani and V. Murino, Abnormal crowd behavior detection by social force optimization, *Human Behavior Understanding (HBU)*, pp. 134-145, Springer, 2011. doi:[10.1007/978-3-642-25446-8_15](https://doi.org/10.1007/978-3-642-25446-8_15).
- [8] E. Miluzzo, N. D. Lane, S. B. Eisenman and A. T. Campbell, CenceMe – injecting sensing presence into social networking applications, *Smart Sensing and Context (EuroSSC)*, pp. 1-28, Springer, 2007. doi:[10.1007/978-3-540-75696-5_1](https://doi.org/10.1007/978-3-540-75696-5_1).
- [9] L. C. Gunther, S. Karcher and T. Bauernhansl, Activity recognition in manual manufacturing: detecting screwing processes from sensor data, *Procedia CIRP*, volume 81, pp. 1177-1182, Elsevier Ltd., 2019. doi:[10.1016/j.procir.2019.03.288](https://doi.org/10.1016/j.procir.2019.03.288).
- [10] G. Dogan, I. Cay, S. S. Ertas, S. R. Keskin, N. Alotaibi and E. Sahin, Where are you? human activity recognition with smartphone sensor data, *Adjunct Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2020 ACM International Symposium on Wearable Computers*, pp. 301-304, ACM, 2020. doi:[10.1145/3410530.3414354](https://doi.org/10.1145/3410530.3414354).
- [11] A. Wang, G. Chen, J. Yang, S. Zhao and C. Y. Chang, A comparative study on human activity recognition using inertial sensors in a smartphone, *IEEE Sensors Journal*, volume 16, issue 11, pp. 4566–4578, IEEE, 2016. doi:[10.1109/JSEN.2016.2545708](https://doi.org/10.1109/JSEN.2016.2545708).

- [12] A. Cherubini, R. Passama, A. Crosnier, A. Lasnier and P. Fraisse, Collaborative manufacturing with physical human–robot interaction, *Robotics and Computer-Integrated Manufacturing*, volume 40, pp. 1–13, Elsevier Ltd., 2016. doi:10.1016/j.rcim.2015.12.007.
- [13] L. D. Riek, Healthcare robotics, *Communications of the ACM*, volume 60, issue 11, pp. 68–78, ACM, 2017. doi:10.1145/3127874.
- [14] T. Stiefmeier, G. Ogris, H. Junker, P. Lukowicz and G. Troster, Combining motion sensors and ultrasonic hands tracking for continuous activity recognition in a maintenance scenario, *10th IEEE International Symposium on Wearable Computers*, pp. 97–104, IEEE, 2006. doi:10.1109/ISWC.2006.286350.
- [15] T. Stiefmeier, D. Roggen and G. Troster, Fusion of string-matched templates for continuous activity recognition, *11th IEEE International Symposium on Wearable Computers*, pp. 41–44, IEEE, 2007. doi:10.1109/ISWC.2007.4373775.
- [16] T. Stiefmeier, D. Roggen, G. Ogris, P. Lukowicz and G. Troster, Wearable activity tracking in car manufacturing, *IEEE Pervasive Computing*, volume 7, issue 2, pp. 42–50, IEEE, 2008. doi:10.1109/MPRV.2008.40.
- [17] H. Koskimaki, V. Huikari, P. Siirtola, P. Laurinen and J. Roning, Activity recognition using a wrist-worn inertial measurement unit: a case study for industrial assembly lines, *17th Mediterranean Conference on Control and Automation*, pp. 401–405, IEEE, 2009. doi:10.1109/MED.2009.5164574.
- [18] T. Peterek, M. Penhaker, P. Gajdos and P. Dohnalek, Comparison of classification algorithms for physical activity recognition, *Innovations in Bio-inspired Computing and Applications*, pp. 123–131, Springer, 2014. doi:10.1007/978-3-319-01781-5_12.
- [19] C. A. Ronao and S. B. Cho, Human activity recognition using smartphone sensors with two-stage continuous hidden markov models, *10th International Conference on Natural Computation (ICNC)*, pp. 681–686, IEEE, 2014. doi:10.1109/ICNC.2014.6975918.
- [20] F. J. Ordóñez and D. Roggen, Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition, *Sensors*, volume 16, issue 1, p. 116, MDPI, 2016. doi:10.3390/s16010115.
- [21] A. Murad and J. Y. Pyun, Deep recurrent neural networks for human activity recognition, *Sensors*, volume 17, issue 11, p. 2556, MDPI, 2017. doi:10.3390/s17112556.
- [22] W. Jiang and Z. Yin, Human activity recognition using wearable sensors by deep convolutional neural networks, *Proceedings of the 23rd ACM international conference on Multimedia*, pp. 1307–1310, ACM, 2015. doi:10.1145/2733373.2806333.
- [23] S. R. Shakya, C. Zhang and Z. Zhou, Comparative study of machine learning and deep learning architecture for human activity recognition using accelerometer data, *International Journal of Machine Learning and Computing*, volume 8, issue 6, pp. 577–582, IJMLRC, 2018. doi:10.18178/ijmlc.2018.8.6.748.
- [24] W. Tao, Z. H. Lai, M. C. Leu and Z. Yin, Worker activity recognition in smart manufacturing using IMU and sEMG signals with convolutional neural networks, *Procedia Manufacturing*, volume 26, pp. 1159–1166, Elsevier Ltd., 2018. doi:10.1016/j.promfg.2018.07.152.

- [25] D. Rumelhart, G. Hinton and R. Williams, Learning representations by back-propagating errors, *Nature*, volume 323, pp. 533–536, 1986. doi:10.1038/323533a0.
- [26] I. Goodfellow, Y. Bengio and A. Courville, Deep Learning, MIT Press, 2016.
- [27] H. Jaeger, Adaptive nonlinear system identification with echo state networks, *Proceedings of the 15th International Conference on Neural Information Processing Systems (NIPS)*, pp. 609-616, ACM, 2002. doi:10.5555/2968618.2968694.
- [28] H. Jaeger, Echo state network, *Scholarpedia*, volume 2, issue 9, p. 2330, 2007. doi:10.4249/scholarpedia.2330.
- [29] W. Maass, T. Natschläger and H. Markram, Real-time computing without stable states: a new framework for neural computation based on perturbations, *Neural Computation*, volume 14, issue 11, pp. 2531-2560, 2002. doi:10.1162/089976602760407955.
- [30] T. Lin, B. G. Horne, P. Tino and C. L. Giles, Learning long-term dependencies is not as difficult with NARX recurrent neural networks, *Proceedings of the 8th International Conference on Neural Information Processing Systems (NIPS)*, pp. 577-583, ACM, 1995. doi:10.5555/2998828.2998910.
- [31] J. Chung, C. Gulcehre, K. Cho and Y. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, *Proceedings of the 27th International Conference on Neural Information Processing Systems (NIPS)*, arXiv, 2014. arXiv:1412.3555v1.
- [32] S. Hochreiter and J. Schmidhuber, Long short-term memory, *Neural Computation*, volume 9, issue 8, pp. 1735-1780, ACM, 1997. doi:10.1162/neco.1997.9.8.1735.
- [33] X. Yuan, L. Li and Y. Wang, Nonlinear dynamic soft sensor modeling with supervised long short-term memory network, *IEEE Transactions on Industrial Informatics*, volume 16, issue 5, pp. 3168-3176, IEEE, 2019. doi:10.1109/TII.2019.2902129.
- [34] S. Ruder. An overview of gradient descent optimization algorithms, arXiv, 2016. arXiv:1609.04747v2.
- [35] M. B. Perry, The exponentially weighted moving average, *Wiley Encyclopedia of Operations Research and Management Science*, 2011. doi:10.1002/9780470400531.eorms0314.
- [36] D. P. Kingma and J. Ba, Adam: a method for stochastic optimization, *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, arXiv, 2015. arXiv:1412.6980v9.
- [37] J. Duchi, E. Hazan and Y. Singer, Adaptive subgradient methods for online learning and stochastic optimization, *Journal of Machine Learning Research (JMLR)*, volume 12, pp. 2121–2159, 2011.
- [38] T. Tieleman and G. Hinton, Lecture 6.5 - RMSprop, *COURSERA: Neural Networks for Machine Learning*, Technical report, 2012.
- [39] D. H. Ballard, Modular learning in neural networks, *AAAI-87: Proceedings of the 6th National Conference on Artificial Intelligence*, volume 1, pp. 279-284, AAAI Press, 1987.
- [40] J. Schmidhuber, Deep learning in neural networks: an overview, *Neural Networks*, volume 61, pp. 85-117, Elsevier Ltd., 2015. doi:10.1016/j.neunet.2014.09.003.

- [41] I. Gulrajani, K. Kumar, F. Ahmed, A. Taiga, F. Visin, D. Vazquez and A. Courville, PixelVAE: a latent variable model for natural images, *Proceedings of the 5th International Conference on Learning Representations (ICLR)*, arXiv, 2017. [arXiv:1611.05013v1](https://arxiv.org/abs/1611.05013v1).
- [42] K. Gregor, I. Danihelka, A. Graves, D. J. Rezende and D. Wierstra, DRAW: a recurrent neural network for image generation, *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, volume 37, pp. 1462-1471, PMLR, 2015. <http://proceedings.mlr.press/v37/gregor15.html>.
- [43] A. Roberts, J. Engel, C. Raffel, C. Hawthorne and D. Eck, A hierarchical latent vector model for learning long-term structure in music, *Proceedings of the 35th International Conference on Machine Learning (ICML)*, volume 80, pp. 4364-4373, PMLR, 2018. <http://proceedings.mlr.press/v80/roberts18a.html>.
- [44] A. Sagheer and M. Kotb, Unsupervised pre-training of a deep lstm-based stacked autoencoder for multivariate time series forecasting problems, *Scientific Reports*, volume 9, p. 19038, 2019. [doi:10.1038/s41598-019-55320-6](https://doi.org/10.1038/s41598-019-55320-6).
- [45] M. E. Tipping and C. M. Bishop, Probabilistic principal component analysis, *Journal of the Royal Statistical Society: Series B*, volume 61, issue 3, pp. 611-622, 1999. [doi:10.1111/1467-9868.00196](https://doi.org/10.1111/1467-9868.00196).
- [46] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.
- [47] S. Kullback and R. A. Leibler, On information and sufficiency, *The annals of mathematical statistics*, volume 22, issue 1, pp. 79-86, 1951. [doi:10.1214/aoms/1177729694](https://doi.org/10.1214/aoms/1177729694).
- [48] M. Hoffman, D. M. Blei, C. Wang and J. Paisley, Stochastic variational inference, *Journal of Machine Learning Research (JMLR)*, volume 14, issue 4, pp. 1303-1347, 2013. <http://jmlr.org/papers/v14/hoffman13a.html>.
- [49] D. P. Kingma and M. Welling, Auto-encoding variational bayes, *Proceedings of the 2nd International Conference on Learning Representations (ICLR)*, arXiv, 2014. [arXiv:1312.6114v10](https://arxiv.org/abs/1312.6114v10).
- [50] D. P. Kingma and M. Welling, An introduction to variational autoencoders, *Foundations and Trends in Machine Learning*, volume 12, issue 4, pp. 307-392, 2019. [doi:10.1561/22000000056](https://doi.org/10.1561/22000000056).
- [51] Y. Kim, S. Wiseman, A. C. Millter, D. Sontag and A. M. Rush, Semi-amortized variational autoencoders, *Proceedings of the 35th International Conference on Machine Learning (ICML)*, volume 80, pp. 2678-2687, PMLR, 2018. <http://proceedings.mlr.press/v80/kim18e.html>.
- [52] J. Domke, Generic methods for optimization-based modeling, *Proceedings of the 15th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 22, pp. 318-326, PMLR, 2012. <http://proceedings.mlr.press/v22/domke12.html>.
- [53] Y. Park, C. D. Kim and G. Kim, Automatic learning rate maximization by on-line estimation of the hessian's eigenvectors, *Proceedings of the 6th International Conference on Neural Information Processing Systems (NIPS)*, 1993.
- [54] B. A. Pearlmutter, Fast exact multiplication by the hessian, *Neural Computation*, volume 6, issue 1, pp. 147-160, 1994.

- [55] Y. Park, C. D. Kim and G. Kim, Variational laplace autoencoders, *Proceedings of the 36th International Conference on Machine Learning (ICML)*, volume 97, pp. 5032-5041, PMLR, 2019. <http://proceedings.mlr.press/v97/park19a.html>.
- [56] R. Pascanu, G. Montufar and Y. Bengio, On the number of response regions of deep feedforward networks with piecewise linear activations, arXiv, 2014. [arXiv:1312.6098v5](https://arxiv.org/abs/1312.6098v5).
- [57] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killen, Z. Lin, N. Gimeshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai and S. Chintala, PyTorch: an imperative style, high-performance deep learning library, *Advances in Neural Information Processing Systems (NeurNIPS)*, volume 32, pp. 8024-8035, arXiv, 2019. [arXiv:1912.01703v1](https://arxiv.org/abs/1912.01703v1).
- [58] E. G. Tabak and E. Vanden-Eijnden, Density estimation by dual ascent of the log-likelihood, *Communications in Mathematical Sciences*, volume 8, issue 1, pp. 217–233, International Press, 2010. [doi:10.4310/CMS.2010.v8.n1.a11](https://doi.org/10.4310/CMS.2010.v8.n1.a11).
- [59] E. G. Tabak and C. V. Turner, A family of nonparametric density estimation algorithms, *Communications on Pure and Applied Mathematics*, volume 66, issue 2, pp. 145–164, Wiley, 2013. [doi:10.1002/cpa.21423](https://doi.org/10.1002/cpa.21423).
- [60] D. Rezende and S. Mohamed, Variational inference with normalizing flows, *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, volume 37, pp. 1530-1538, PMLR, 2015. <http://proceedings.mlr.press/v37/rezende15.html>.
- [61] J. M. Tomczak and M. Welling, Improving variational auto-encoders using householder flow, arXiv, 2017. [arXiv:1611.09630v4](https://arxiv.org/abs/1611.09630v4).
- [62] C. Bischof and X. Sun, On orthogonal block elimination, *Argonne National Laboratory, Argonne, IL*, Technical Report MCS-P450-0794, 1994.
- [63] X. Sun and C. Bischof, A basis-kernel representation of orthogonal matrices, *SIAM Journal on Matrix Analysis and Applications*, volume 16, issue 4, pp. 1184–1196, 1995. [doi:10.1137/S0895479894276369](https://doi.org/10.1137/S0895479894276369).
- [64] A. S. Householder, Unitary triangularization of a nonsymmetric matrix, *Journal of the ACM (JACM)*, volume 5, issue 4, pp. 339–342, ACM, 1958. [doi:10.1145/320941.320947](https://doi.org/10.1145/320941.320947).
- [65] L. Maaten and G. Hinton, Visualizing data using t-SNE, *Journal of Machine Learning Research (JMLR)*, volume 9, issue 86, pp. 2579-2605, 2008. <https://www.jmlr.org/papers/v9/vandermaaten08a.html>.
- [66] R. Duda, P. Hart and D. Stork, *Pattern Classification*, 2nd Edition, Wiley, 2000.
- [67] T. Fawcett, An introduction to ROC analysis, *Pattern Recognition Letters*, volume 27, issue 8, pp. 861-874, Elsevier Ltd., 2006. [doi:10.1016/j.patrec.2005.10.010](https://doi.org/10.1016/j.patrec.2005.10.010).
- [68] D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever and M. Welling, Improved variational inference with inverse autoregressive flow, *Proceedings of the 30th International Conference on Neural Information Processing Systems (NIPS)*, pp. 4743-4751, ACM, 2016. [doi:10.5555/3157382.3157627](https://doi.org/10.5555/3157382.3157627).
- [69] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser and I. Polosukhin, Attention is all you need, *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS)*, pp. 6000-6010, ACM, 2017. [doi:10.5555/3295222.3295349](https://doi.org/10.5555/3295222.3295349).

- [70] R. Buyya and S. N. Srirama, Internet of Things (IoT) and new computing paradigms, *Fog and Edge Computing: Principles and Paradigms*, pp. 1-23, Wiley, 2019. doi:[10.1002/9781119525080.ch1](https://doi.org/10.1002/9781119525080.ch1).
- [71] H. H. Pang and K. Tan, Authenticating query results in edge computing, *Proceedings of 20th International Conference on Data Engineering*, pp. 560-571, IEEE, 2004. doi:[10.1109/ICDE.2004.1320027](https://doi.org/10.1109/ICDE.2004.1320027).
- [72] V. Sze, Y. Chen, T. Yang and J. S. Emer, Efficient processing of deep neural networks: A tutorial and survey, *Proceedings of the IEEE*, volume 105, issue 12, pp. 2295-2329, IEEE, 2017. doi:[10.1109/JPROC.2017.2761740](https://doi.org/10.1109/JPROC.2017.2761740).
- [73] S. Mittal, A survey of techniques for approximate computing, *ACM Computing Surveys*, volume 48, issue 4, pp. 1-33, ACM, 2016. doi:[10.1145/2893356](https://doi.org/10.1145/2893356).