

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΕΦΑΡΜΟΣΜΕΝΩΝ ΜΑΘΗΜΑΤΙΚΩΝ ΚΑΙ ΦΥΣΙΚΩΝ ΕΠΙΣΤΗΜΩΝ



ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΣΩΤΗΡΗΣ ΚΑΛΛΙΣΙΗΣ

«ΠΑΡΑΓΩΓΙΚΑ ΜΟΝΤΕΛΑ ΤΕΧΝΗΤΗΣ ΝΟΗΜΟΣΥΝΗΣ ΜΕ ΑΠΟΘΟΡΥΒΟΠΟΙΗΣΗ ΠΙΘΑΝΟΘΕΩΡΗΤΙΚΩΝ ΜΟΝΤΕΛΩΝ ΔΙΑΧΥΣΗΣ : ΜΕΘΟΔΟΛΟΓΙΑ ΚΑΙ ΕΦΑΡΜΟΓΕΣ ΣΕ ΙΑΤΡΙΚΑ ΔΕΔΟΜΕΝΑ»

ΣΤΗΝ ΕΠΙΣΤΗΜΟΝΙΚΗ ΠΕΡΙΟΧΗ:
ΠΙΘΑΝΟΤΗΤΕΣ - ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

ΕΠΙΒΛΕΠΩΝ:
ΜΙΧΑΗΛ ΛΟΥΛΑΚΗΣ, ΚΑΘΗΓΗΤΗΣ, ΣΕΜΦΕ

ΤΡΙΜΕΛΗΣ ΕΠΙΤΡΟΠΗ
ΜΙΧΑΗΛ ΛΟΥΛΑΚΗΣ, ΚΑΘΗΓΗΤΗΣ, ΣΕΜΦΕ
ΑΝΤΩΝΙΟΣ ΠΑΠΑΠΑΝΤΟΛΕΩΝ, ΑΝΑΠΛΗΡΩΤΗΣ ΚΑΘΗΓΗΤΗΣ, ΣΕΜΦΕ
ΕΜΜΑΝΟΥΗΛ ΓΕΩΡΓΟΥΛΗΣ, ΚΑΘΗΓΗΤΗΣ, ΣΕΜΦΕ

ΑΘΗΝΑ, ΙΟΥΝΙΟΣ 2025

ΕΥΧΑΡΙΣΤΙΕΣ

Αρχικά, θα ήθελα να ευχαριστήσω τον Καθηγητή του Εθνικού Μετσόβιου Πολυτεχνείου, κύριο Μιχαήλ Λουλάκη για την ανάθεση της παρούσας διπλωματικής εργασίας αλλά και την άψογη συνεργασία που είχαμε καθ' όλη τη διάρκεια της εκπόνησής της. Επίσης, οφείλω ένα μεγάλο ευχαριστώ στους κύριους Αντώνιο Παπαπαντωλέων, Αλέξανδρο Σαπλαούρα και Γιώργο Στάμου, επίσης καθηγητές του ΕΜΠ, αλλά και στον κύριο Ανδρέα Αφαντίτη, Διευθύνων Σύμβουλο της NovaMechanics LTD για την παραχώρηση υπολογιστικών πόρων και την σημαντική τους βοήθεια στην ομαλή εκτέλεση των πειραμάτων που περιλαμβάνονται στην παρούσα εργασία.

Ασφαλώς, δεν θα μπορούσα να παραλείψω από αυτές τις ευχαριστίες, τόσο τους γόνεις μου που μου προσέφεραν τα πάντα για να καταφέρω να φτάσω μέχρι αυτό το σημείο, όσο και τους δικούς μου ανθρώπους, τους φίλους μου που με στήριξαν καθ' όλη τη διάρκεια των σπουδών και συνέβαλαν σε μεγάλο βαθμό στην επίτευξη των στόχων μου.

.....

Σωτήρης Καλλίσιης

© (2025) Εθνικό Μετσόβιο Πολυτεχνείο. 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 το συγγραφέα. Οι απόψεις και τα συμπεράσματα που περιέχονται σ' αυτό το έγγραφο εκφράζουν το συγγραφέα και δεν πρέπει να ερμηνευτεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

Περίληψη

Η ανάγκη για την παραγωγή συνθετικών δεδομένων υψηλής ποιότητας αυξάνεται ραγδαία σε πληθώρα εφαρμογών της τεχνητής νοημοσύνης, ιδιαίτερα όταν η συλλογή πραγματικών δεδομένων είναι περιορισμένη, δαπανηρή ή εγείρει ζητήματα απορρήτου, όπως συμβαίνει χαρακτηριστικά στον τομέα της ιατρικής απεικόνισης.

Η παρούσα διπλωματική εργασία μελετά την Αποθορυβοποίηση Πιθανοθεωρητικών Μοντέλων Διάχυσης (DDPMs), μια κατηγορία παραγωγικών μοντέλων τεχνητής νοημοσύνης που επιδεικνύουν ιδιαίτερη ικανότητα στην παραγωγή συνθετικών δεδομένων υψηλής ποιότητας. Με αφορμή τις αυξανόμενες ανάγκες σε τομείς όπως η ιατρική απεικόνιση, η εργασία εστιάζει τόσο στη θεωρητική θεμελίωση των DDPMs όσο και στην πρακτική τους αξιολόγηση.

Η μελέτη περιλαμβάνει την παρουσίαση του μαθηματικού τους υποβάθρου, των βασικών εννοιών της μηχανικής μάθησης και των παραγωγικών μοντέλων, την αρχιτεκτονική U-Net που τα υποστηρίζει, καθώς και τεχνικές εκπαίδευσης και αξιολόγησης. Επιπλέον, προτείνεται μια παραλλαγή των DDPMs, με την ονομασία MDDPM, η οποία εισάγει τροποποιήσεις στις δύο αλληλένδετες διαδικασίες που χαρακτηρίζουν τα μοντέλα διάχυσης.

Τα μοντέλα εφαρμόζονται πειραματικά στα σύνολα δεδομένων CIFAR-10, CelebA και ChestX-ray8, με την αξιολόγηση να βασίζεται τόσο σε ποιοτικά όσο και σε ποσοτικά κριτήρια, όπως οι δείκτες IS, FID και NLL.

Τα αποτελέσματα καταδεικνύουν τη δυναμική των DDPMs στην παραγωγή ρεαλιστικών συνθετικών εικόνων. Αν και το MDDPM δεν πέτυχε την παραγωγή ρεαλιστικών δειγμάτων, εμφανίζει ενδείξεις προοπτικής, δικαιολογώντας περαιτέρω μελέτη. Συνολικά, η εργασία αναδεικνύει τις δυνατότητες των πιθανοθεωρητικών παραγωγικών μοντέλων και την ανάγκη συνέχισης της έρευνας στον τομέα αυτό, ιδίως ως προς τις εφαρμογές τους σε ιατρικά δεδομένα.

Λέξεις Κλειδιά. Πιθανοθεωρητικά Μοντέλα Διάχυσης, Παραγωγικά Μοντέλα, Συνθετικά Δεδομένα, Τεχνητή Νοημοσύνη, DDPM, DDIM, Ιατρική Απεικόνιση, Επεξεργασία Ιατρικών Δεδομένων, U-Net, CIFAR-10, CelebA, ChestX-ray8

Abstract

The demand for generating high-quality synthetic data is rapidly increasing across a wide range of artificial intelligence applications, especially in cases where the collection of real-world data is limited, costly, or raises privacy concerns—such as in medical imaging.

This thesis explores Denoising Diffusion Probabilistic Models (DDPMs), a class of generative AI models that demonstrate exceptional capabilities in producing realistic synthetic data. Motivated by the growing needs in fields such as medical imaging, the study focuses both on the theoretical foundations of DDPMs and their practical evaluation.

The work includes a presentation of the mathematical background of the models, essential concepts from machine learning and generative modeling, the U-Net architecture on which they are based, as well as training and evaluation techniques. Furthermore, a modified version of DDPMs is proposed, referred to as MDDPM (Modified Denoising Diffusion Probabilistic Model), which introduces changes to the two interdependent processes that define diffusion models.

The models are experimentally applied to three datasets: CIFAR-10, CelebA, and ChestX-ray8. Evaluation is carried out using both qualitative visual inspection and quantitative metrics such as Inception Score (IS), Frechet Inception Distance (FID), and Negative Log-Likelihood (NLL).

The results highlight the strong potential of DDPMs in generating realistic synthetic images. Although the MDDPM variant did not succeed in producing highly realistic samples, it shows promising indications that justify further investigation. Overall, this thesis showcases the capabilities of probabilistic generative models and emphasizes the need for continued research in this area—particularly regarding their application to medical data.

Keywords. Diffusion Probabilistic Models, Generative Models, Synthetic Data Generation, Artificial Intelligence, DDPM, DDIM, Medical Imaging, Medical Data Processing, U-Net, CIFAR-10, CelebA, ChestX-ray8

Κατάλογος Σχημάτων-Εικόνων

2.1 Παράδειγμα ενός Δικτύου Bayes για 8 τυχαίες μεταβλητές. 15
3.1 Παραγωγικά και Διακριτικά Μοντέλα αναπαριστόμενα ως Bayesian Δίκτυα 18
3.2 Η Μαρκοβιανή αλυσίδα της προς τα εμπρός (αντίστροφης) διαδικασίας διάχυσης για την παραγωγή ενός δείγματος με σταδιακή πρόσθεση θορύβου (αποθορύβωση) 20
3.3 Αλγόριθμοι εκπαίδευσης και δειγμάτισης για μοντέλα DDPM . . . 25
3.4 Σύγκριση επίδοσης DDIM και DDPM 27
3.5 Αυτοκωδικοποιητής 33
3.6 Αυτοκωδικοποιητές Βασισμένοι σε Μεταβολική Συμπερασματολογία (Variational Autoencoders). 34
3.7 Αρχιτεκτονική GAN 35
3.8 Εκπαίδευση ενός GAN 36
3.9 Αντιστοίχηση δειγμάτων - τιμών ενέργειας στα Energy Based Models 37
3.10 Normalizing Flows : Μετασχηματισμοί από απλές σε πιο σύνθετες κατανομές 38
4.1 Ο νευρώνας του ανθρώπινου εγκεφάλου. Πηγή : Διαμαντάρας, Κ., & Μπότσης, Δ. (2019). Μηχανική μάθηση. Αθήνα: Κλειδάριθμος. . 41
4.2 Απλό Perceptron 42
4.3 Παράδειγμα δικτύου Perceptron πολλαπλών στρωμάτων (MLP) . . 43
4.4 Παράδειγμα ενός CNN με δύο συνελικτικά επίπεδα όπου η είσοδος έχει 3 κανάλια (red, green, blue) 45
4.5 Παράδειγμα Max Pooling με φίλτρο μεγέθους 3×3. 46
4.6 Παράδειγμα αντίστροφης συνέλιξης 46
4.7 Σχηματική αναπαράσταση του δικτύου U-Net 47
4.8 Residual Block 51
4.9 Scaled Dot-Product Attention 52
4.10 Attention Block 52
4.11 Παράδειγμα χρήσης του Dropout σε ένα Multi-Layer Perceptron . 53
5.1 Σύγκριση των Batch Gradient Descent , Stochastic Gradient Descent και Mini Batch Gradient Descent 56
7.1 Σύνολο Δεδομένων CIFAR-10 63
7.2 Σύνολο Δεδομένων ChestX-ray8 65

7.3 Σύνολο Δεδομένων CelebA 66
8.1 Διάγραμμα συνάρτησης απώλειας - βήματος εκπαίδευσης για το σύνολο δεδομένων CIFAR-10 68
8.2 Σταδιακή αποθορύβωση για παραγωγή δειγμάτων παραπλήσιων του συνόλου CIFAR-10 69
8.3 Δείγματα CIFAR-10 69
8.4 Διάγραμμα συνάρτησης απώλειας - βήματος εκπαίδευσης για το σύνολο δεδομένων ChestX-ray8 71
8.5 Σταδιακή αποθορύβωση για παραγωγή δειγμάτων ChestX-ray8 . . 71
8.6 Δείγματα ChestX-ray8 72
8.7 Διάγραμμα συνάρτησης απώλειας - βήματος εκπαίδευσης για το σύνολο δεδομένων CelebA 73
8.8 Δείγματα CelebA 73
8.9 Σταδιακή αποθορύβωση για παραγωγή δειγμάτων CelebA 74
8.10 Δείγματα MDDPM CIFAR-10 76
8.11 Δείγματα MDDPM CIFAR-10 77
10.1 Περισότερα Δείγματα ChestX-ray8 81
10.2 Περισότερα Δείγματα CelebA 82

Περιεχόμενα

1 Εισαγωγή **9**
1.1 Σκοπός και Αντικείμενο της Εργασίας 9
1.2 Ανασκόπηση Βιβλιογραφίας 10

2 Πιθανοθεωρητική Συμπερασματολογία **12**
2.1 Μεταβολική Συμπερασματολογία 13
2.2 Δίκτυα Bayes 14

3 Μηχανική Μάθηση **16**
3.1 Εισαγωγή 16
3.2 Είδη Μάθησης 17
3.2.1 Επιβλεπόμενη Μάθηση (Supervised Learning) 17
3.2.2 Μη - Επιβλεπόμενη Μάθηση (Unsupervised Learning) . . . 17
3.2.3 Ενισχυτική Μάθηση (Reinforcement Learning) 18
3.3 Διακριτικά και Παραγωγικά Μοντέλα (Discriminative and Generative Models) 18
3.3.1 Εφαρμογές παραγωγικών μοντέλων σε ιατρικά δεδομένα . . 19
3.4 Μοντέλα Διάχυσης 19
3.4.1 Αποθορυβωποίηση Πιθανοθεωρητικών Μοντέλων Διάχυσης (Denoising Diffusion Probabilistic Models - DDPMs) 20
3.4.2 Αποθορυβοποίηση Έμμεσων Πιθανοθεωρητικών Μοντέλων Διάχυσης (Denoising Diffusion Implicit Models - DDIMs) . 25
3.4.3 Τροποποιημένη Αποθορυβοποίηση Πιθανοθεωρητικών Μοντέλων Διάχυσης (Modified Denosing Diffusion Probabilistic Models - MDDPM) 28
3.4.4 Σύγκριση Αποθορυβοποίησης Έμμεσων Πιθανοθεωρητικών Μοντέλων Διάχυσης (DDIM) και Τροποποιημένης Αποθορυβοποίησης Πιθανοθεωρητικών Μοντέλων Διάχυσης (MDDPM) 32
3.5 Σύντομη ανασκόπηση άλλων γνωστών παραγωγικών μοντέλων . . 33
3.5.1 Αυτοκωδικοποιητές Βασισμένοι σε Μεταβολική Συμπερασματολογία (Variational Autoencoders) 33
3.5.2 Παραγωγικά Ανταγωνιστικά Δίκτυα (Generative Adversarial Networks) 34
3.5.3 Energy Based Models 36
3.5.4 Normalizing Flows 38
3.5.5 Σύγκριση Παραγωγικών Μοντέλων 39

4 Νευρωνικά Δίκτυα **41**
4.1 Συνελικτικά Νευρωνικά Δίκτυα 43
4.1.1 Συνελικτικό Επίπεδο 43
4.1.2 Επίπεδο Συγκέντρωσης 45

4.1.3 Επίπεδο Αντρίστροφης Συνέλιξης 46
4.1.4 Επίπεδο Κανονικοποίησης 46
4.2 Νευρωνικό Δίκτυο U-Net 47
4.3 Αρχιτεκτονικά Στοιχεία και Τεχνικές Βελτιστοποίησης Νευρωνικών Δικτύων 49
4.3.1 Time Embeddings 49
4.3.2 Residual Blocks 50
4.3.3 Μηχανισμοί Προσοχής (Attention Mechanisms) και Attention Blocks 51
4.3.4 Dropout 52
4.3.5 Εκθετικός Κινητός Μέσος Όρος 53

5 Εκπαίδευση Νευρωνικών Δικτύων 54
5.1 Gradient Descent (GD) 54
5.1.1 Stochastic Gradient Descent (SGD) 55
5.1.2 Mini-batch Gradient Descent (MB-GD) 55
5.1.3 Adam 56
5.2 Μέθοδος Οπισθοδιάδοσης (Backpropagation) 57

6 Μέτρα αξιολόγησης της επίδοσης του μοντέλου 59
6.1 Inseption Score (IS) 59
6.2 Frechet Inception Distance (FID) 60
6.2.1 Απόσταση Frechet 61
6.2.2 Ορισμός του Frechet Inception Score (FID) 61
6.3 Negative Log-Likelihood (NLL) 62

7 Σύνολα Δεδομένων και Προεπεξεργασία 63
7.1 CIFAR-10 63
7.1.1 Περιγραφή Συνόλου 63
7.1.2 Προεπεξεργασία 64
7.2 ChestX-ray8 64
7.2.1 Περιγραφή 64
7.2.2 Προεπεξεργασία 64
7.3 CelebA 65
7.3.1 Περιγραφή 65
7.3.2 Προεπεξεργασία 65

8 Πειραματική Διαδικασία και Αποτελέσματα 67
8.1 Αποτελέσματα CIFAR-10 68
8.2 Αποτελέσματα ChestX-ray8 70
8.3 Αποτελέσματα CelebA 72
8.4 Μετρικές 74
8.5 Αποτελέσματα Τροποποιημένης Αποθορυβοποίησης Πιθανοθεωρητικών Μοντέλων Διάχυσης (MDDPM) 76

9 Συμπεράσματα 78

10 Μελλοντικές Κατευθύνσεις **80**

Παράρτημα Α: Περισότερα δείγματα **81**

Βιβλιογραφία **83**

1 Εισαγωγή

1.1 Σκοπός και Αντικείμενο της Εργασίας

Η παρούσα διπλωματική εργασία εστιάζει στην θεωρητική θεμελίωση της Αποθορυβοποίησης Πιθανοθεωρητικών Μοντέλων Διάχυσης (Denoising Diffusion Probabilistic Models - DDPMs), μιας κατηγορίας παραγωγικών μοντέλων τεχνητής νοημοσύνης που έχουν, τα τελευταία χρόνια, αναδειχθεί ως ισχυρά εργαλεία για την παραγωγή συνθετικών δεδομένων, με εφαρμογές σε ποικίλους τομείς.

Αρχικά, ορίζονται βασικές έννοιες της πιθανοθεωρητικής συμπερασματολογίας, οι οποίες αποτελούν το θεωρητικό υπόβαθρο για την κατανόηση των μοντέλων διάχυσης.

Στην επόμενη παράγραφο, δίνονται ορισμοί και εννοιολογικά θεμέλια της μηχανικής μάθησης, με έμφαση στα παραγωγικά μοντέλα ως μια εισαγωγή στα μοντέλα διάχυσης. Ακολουθεί λεπτομερής μελέτη των DDPMs, περιλαμβάνοντας την αρχιτεκτονική τους, τη διαδικασία αποθορυβοποίησης μέσω αντίστροφης διάχυσης, καθώς και βασικές παραμέτρους που επηρεάζουν την απόδοσή τους, όπως ο αριθμός βημάτων διάχυσης και το πρόγραμμα διακύμανσης. Γίνεται επίσης αναφορά, στη μετέπειτα βελτίωση των DDPMs μέσω των DDIMs (Denoising Diffusion Implicit Models), τα οποία επιδιώκουν τη μείωση του χρόνου δειγματοληψίας. Στο πλαίσιο αυτό παρουσιάζεται και μία τροποποιημένη εκδοχή των DDPMs, με την ονομασία MDDPM (Modified Denoising Diffusion Probabilistic Model), η οποία εισάγει αλλαγές στον τρόπο μοντελοποίησης των δύο αλληλένδετων διαδικασιών που χαρακτηρίζουν τα μοντέλα διάχυσης. Η ενότητα ολοκληρώνεται με μια ανασκόπηση διαφόρων γνωστών παραγωγικών μοντέλων και την επισήμανση των ομοιοτήτων που έχουν με τα μοντέλα διάχυσης.

Ακολουθεί παρουσίαση των νευρωνικών δικτύων, αρχικά σε γενικότερο επίπεδο, πριν επικεντρωθούμε στην αρχιτεκτονική U-Net που χρησιμοποιείται στα DDPMs, και σε επιμέρους αρχιτεκτονικά στοιχεία και τεχνικές για τη βελτίωση της απόδοσης.

Στη συνέχεια, εξετάζονται οι βασικοί αλγόριθμοι εκπαίδευσης νευρωνικών δικτύων, καθώς και τα μέτρα που χρησιμοποιούνται για την αξιολόγηση της απόδοσης παραγωγικών μοντέλων.

Στο πειραματικό μέρος, τα προτεινόμενα μοντέλα εφαρμόζονται σε τρία διαφορετικά σύνολα δεδομένων: το CIFAR-10 ως γενικό σημείο αναφοράς της επίδοσης παραγωγικών μοντέλων, το CelebA που περιλαμβάνει πρόσωπα διασήμων, και το ChestX-ray8 ως χαρακτηριστικό παράδειγμα συνόλου ιατρικών εικόνων. Γίνεται εκτενής παρουσίαση της πειραματικής διαδικασίας και για κάθε περίπτωση αξιολογείται η δυνατότητα παραγωγής συνθετικών εικόνων μέσω οπτικοποίησης κάποιων δειγμάτων και με τον υπολογισμό καθιερωμένων με-

τρικών αξιολόγησης όπως Inception Score (IS), Frechet Inception Distance (FID) και Negative Log-Likelihood (NLL). Επίσης παρουσιάζονται τα πειραματικά αποτελέσματα της παραλλαγής MDDPM.

Τέλος, η εργασία ολοκληρώνεται με τη συγκέντρωση και συζήτηση των συμπερασμάτων που προκύπτουν από τα πειραματικά αποτελέσματα των DDPMs. Αν και η προτεινόμενη παραλλαγή του μοντέλου, MDDPM, δεν επέφερε άμεσα τα αναμενόμενα αποτελέσματα, παρουσιάζει ενδείξεις ότι, με περαιτέρω μελέτη, μπορεί να επιτύχει ικανοποιητική παραγωγή συνθετικών δειγμάτων. Η παρατήρηση αυτή ανοίγει το δρόμο για συγκριτική αξιολόγηση έναντι του αρχικού DDPM και τη διερεύνηση του κατά πόσο μπορεί να οδηγήσει σε βελτίωση της ποιότητας των παραγόμενων εικόνων. Συνολικά, τα αποτελέσματα αναδεικνύουν τις δυνατότητες των πιθανοθεωρητικών παραγωγικών μοντέλων και υπογραμμίζουν την επιτακτική ανάγκη για περαιτέρω ενασχόληση με τον τομέα αυτό, προκειμένου να αξιοποιηθούν πλήρως σε εφαρμογές τεχνητής νοημοσύνης και, ειδικότερα, στην επεξεργασία ιατρικών δεδομένων.

1.2 Ανασκόπηση Βιβλιογραφίας

Τα πιθανοθεωρητικά μοντέλα διάχυσης είναι μια ισχυρή και ευέλικτη κλάση παραγωγικών μοντέλων τεχνητής νοημοσύνης, η οποία έχει κεντρίσει το ενδιαφέρον των ερευνητών την τελευταία δεκαετία καθώς επιτυγχάνουν την παραγωγή υψηλής ποιότητας δειγμάτων, είτε μιλάμε για ήχο, είτε για εικόνες.

Τα μοντέλα διάχυσης προτάθηκαν αρχικά από τους Sohl-Dickstein, Weiss, Maheswaranathan και Ganguli σε μια δημοσίευση τους το 2015, οι οποίοι εμπνεύστηκαν από τις διαδικασίες διάχυσης σε συστήματα που δεν βρίσκονται σε θερμοδυναμική ισορροπία [1].

Τα μοντέλα διάχυσης, και συγκεκριμένα η αποθορυβοποίηση πιθανοθεωρητικών μοντέλων διάχυσης (Denoising Diffusion Probabilistic Models), έγιναν ευρέως γνωστά το 2020, μετά από δημοσίευση των Ho,Jain και Abbeel [2], οι οποίοι όρισαν μια νέα αρχιτεκτονική ενός μοντέλου που κάνει χρήση δύο μαρκοβιανών αλυσίδων, για την πρόσθεση και αφαίρεση θορύβου από τα δείγματα με σκοπό την παραγωγή νέων δειγμάτων από την αρχική κατανομή στην οποία ανήκουν τα δεδομένα . Λόγω των εξαιρετικής ποιότητας δειγμάτων που παρουσιάστηκαν στη συγκεκριμένη δημοσίευση καθώς και των ανταγωνιστικών, σε σχέση με τα μέχρι τότε κορυφαία παραγωγικά μοντέλα (GANs), επιδόσεων σε μετρικές (π.χ. FID) που είχαν τα μοντέλα που εκπαιδεύτηκαν, ακολούθησε εκτενής μελέτη όσο αφορά τα μοντέλα διάχυσης με διάφορες βελτιώσεις ή και παραλλαγές των DDPM να προτείνοται τα επόμενα χρόνια (όπως τα DDIM, Score-based Models κ.α.). Επίσης παρατηρείται χρήση των εν λόγω μοντέλων σε τομείς πέραν της μηχανικής όρασης (computer vision) . Αναφορικά, μια δημοσίευση με την οποία θα ασχολουθούμε στην συγκεκριμένη εργασία είναι αυτή των Nichol και Dhariwal [3], οι οποίοι πρότειναν τρόπους βελτίωσης της επίδοσης αλλά και της ταχύτητας των DDPM, κάποιους από τους οποίους θα χρησιμο-

ποιήσουμε και στα δικά μας μοντέλα.

Ιδιαίτερο ενδιαφέρον παρουσιάζει η εφαρμογή των DDPMs σε ιατρικής φύσεως δεδομένα, όπου η παραγωγή συνθετικών δειγμάτων μπορεί να βοηθήσει στην αντιμετώπιση προβλημάτων όπως η ανισορροπία δειγμάτων σε κλάσεις, η διατήρηση της ιδιωτικότητας των ασθενών και η έλλειψη μεγάλου πλήθους δειγμάτων για αποτελεσματική στατιστική ανάλυση ή εκπαίδευση διακριτικών μοντέλων μηχανικής μάθησης (π.χ. για ταξινόμηση). Σύγχρονες μελέτες έχουν δείξει ότι τα μοντέλα διάχυσης μπορούν να παράγουν υψηλής ποιότητας συνθετικές εικόνες από MRI εγκεφάλου [4], αξονικές τομογραφίες [5], καθώς και ιστολογικές εικόνες με ελεγχόμενη τοποθέτηση πυρήνων [6].

Η αυξημένη ικανότητα ελέγχου που προσφέρουν τα μοντέλα διάχυσης, σε συνδυασμό με τη σταθερότητα της εκπαίδευσης τους και την ποιοτική υπεροχή των παραγόμενων δειγμάτων, καθιστούν τα DDPMs έναν εξαιρετικά ισχυρό υποψήφιο για εφαρμογές σε ποικίλους τομείς.

2 Πιθανοθεωρητική Συμπερασματολογία

Η ενότητα αυτή παρουσιάζει τα βασικά θεωρητικά εργαλεία της πιθανοθεωρητικής συμπερασματολογίας, ξεκινώντας από τον Κανόνα του Bayes και συνεχίζοντας με θεμελιώδεις έννοιες όπως η εντροπία, η απόκλιση Kullback-Leibler και η μεταβολική συμπερασματολογία, μια προσεγγιστική μέθοδος για τον υπολογισμό κατανομών πιθανότητας. Επίσης παρουσιάζονται συνοπτικά τα Δίκτυα Bayes, μια κατηγορία πιθανοθεωρητικών γραφικών μοντέλων.

Ορισμός 2.1. *(Πιθανοθεωρητική Συμπερασματολογία - Probabilistic Inference) Έστω ότι έχουμε κάποια a-priori πληροφορία για την κατανομή μιας τυχαίας μεταβλητής W. Δεδομένης της παρατήρησης μια τυχαίας μεταβλητής R, η πιθανοθεωρητική συμπερασματολογία είναι η διαδικασία ενημέρωσης της a-priori πληροφορίας $\mathbb{P}(W)$ σε μια posteriori πληροφορία $\mathbb{P}(W|R)$ [7].*

Η βασική αρχή της πιθανοθεωρητικής συμπερασματολογίας είναι ο κανόνας του Bayes:

Θεώρημα 2.1. *(Κανόνας του Bayes)*
Δεδομένων δύο τυχαίων διανυσμάτων $Z \in \mathbb{R}^n$ και $X \in \mathbb{R}^m$, τότε για κάθε $z \in \mathbb{R}^n$ και $x \in \mathbb{R}^m$, ισχύει ότι:

$$p(Z = z|X = x) = \frac{p(X = x|Z = z) \cdot p(Z = z)}{p(X = x)} \tag{2.1}$$

Συχνά η τ.μ. Z είναι μια λανθάνουσα (latent) μεταβλητή, δηλαδή δεν μπορεί να μετρηθεί ενώ η X είναι παρατηρήσιμη. Σε αυτές τις περιπτώσεις γνωρίζοντας την a priori κατανομή της Z αλλά και την κατανομή της X δεδομένης της Z μπορούμε να γράψουμε :

$$p(X) = \int_{z \in Z} p(X|Z = z)p(Z = z)dz \tag{2.2}$$

και άρα :

$$p(Z|X) = \frac{p(X|Z)p(Z)}{\int_{z \in Z} p(X|Z = z)p(Z = z)dz} \tag{2.3}$$

Σε κάποιες περιπτώσεις ο υπολογισμός του ολοκληρώματος της εξίσωσης (2.2) μπορεί να γίνει αναλυτικά και να βρεθεί κλειστή μορφή αυτού, ενώ σε άλλες περιπτώσεις αυτό είναι αρκετά χρονοβόρο και υπολογιστικά "ακριβό". Για αυτό το λόγο έχουν προταθεί διάφορες μέθοδοι για προσέγγιση συναρτήσεων πυκνότητας πιθανότητας.
Μια εξ αυτών είναι η μεταβολική συμπερασματολογία (variational inference), η οποία στοχεύει στην εύρεση μιας καλής προσέγγισης μιας a-posteriori κατανομής, από την οποία μπορούμε να δειγματίζουμε εύκολα, μέσω βελτιστοποίησης.

2.1 Μεταβολική Συμπερασματολογία

Στην ενότητα αυτή θα μελετήσουμε τις βασικές ιδέες πίσω από τη μεταβολική συμπερασματολογία (variational inference). Δίνουμε αρχικά κάποιους βασικούς ορισμούς στα πλαίσια της πιθανοτικής συμπερασματολογίας:

Ορισμός 2.2. *(Έκπληξη - Surprise)*
Η έκπληξη ενός πιθανοτικού γεγονότος με πιθανότητα u ορίζεται ως:

$$S[u] = -logu \tag{2.4}$$

Ορισμός 2.3. *(Εντροπία)*
Η εντροπία μιας κατανομής p είναι η μέση έκπληξη των δειγμάτων από τη συγκεκριμένη κατανομή. Επομένως, η εντροπία είναι μια έννοια αβεβαιότητας όσο αφορά την κατανομή p. Αν η εντροπία παίρνει μικρές τιμές τότε μπορούμε να είμαστε πιο βέβαιοι για την τιμή $x \sim p$, σε σχέση με την περίπτωση που η εντροπία παίρνει μεγάλες τιμές.

$$H[p] = \mathbb{E}_{x\sim p}[S[p(x)]] = \mathbb{E}_{x\sim p}[-logp(x)] \tag{2.5}$$

Ορισμός 2.4. *(Διασταυρούμενη-Εντροπία)*
Η διασταυρούμενη εντροπία (cross-entropy) μιας κατανομής q σε σχέση με μια κατανομή p ορίζεται ως:

$$H[p||q] = \mathbb{E}_{X\sim p}[S[q(x)]] = \mathbb{E}_{X\sim p}[-logq(x)] \tag{2.6}$$

Ορισμός 2.5. *(Απόκλιση Kullback-Leibler)*
Η απόκλιση Kullback-Leibler είναι μια ποσότητα που μετρά την απόκλιση μεταξύ δύο κατανομών (δεν είναι μετρική αφού δεν ικανοποιεί την τριγωνική ανισότητα) και ορίζεται ως:

$$KL[p||q] = H[p||q] - H[p] = \mathbb{E}_{\theta\sim p}[S[q(\theta)] - S[p(\theta)]] = \mathbb{E}_{\theta\sim p}[log\frac{p(\theta)}{q(\theta)}] \tag{2.7}$$

Η απόκλιση Kullback-Leibler είναι μια μη αρνητική ποσότητα ($KL(p||q) \geq 0$) και παίρνει την τιμή 0 αν και μόνο αν $p = q$ ως μέτρα πιθανότητας.

Έχοντας τώρα δώσει τους βασικούς ορισμούς, μπορούμε να περιγράψουμε εν συντομία την διαδικασία που ακολουθείται στην μεταβολική συμπερασματολογία. Έστω ότι θέλουμε να προσεγγίσουμε την posterior κατανομή $p(Z|X)$ των λανθανουσών τυχαίων μεταβλητών Z δεδομένων των παρατηρήσιμων τ.μ. X. Αρχικά καθορίζεται μια οικογένεια κατανομών $\mathcal{D}$ πάνω στις λανθάνουσες τ.μ. Έπειτα προσπαθούμε να βρούμε το μέλος της οικογένειας αυτής που ελαχιστοποιεί την απόκλιση Kullback-Leibler με την πραγματική posterior κατανομή, δηλαδή:

$$q^*(Z) = argmin_{q(Z)\in\mathcal{D}} KL(q(Z)||p(Z|X)) \tag{2.8}$$

Βασικό τμήμα της διαδικασίας αποτελεί η επιλογή κατάλληλης οικογένειας κατανομών $\mathcal{D}$. Με τον όρο κατάλληλη, εννοούμε να είναι αρκετά "εύπλαστη" ώστε να μπορούμε να προσεγγίσουμε όσο το δυνατό καλύτερα την posterior

κατανομή αλλά και αρκετά "απλή" ώστε να είναι αποδοτικό το πρόβλημα βελτιστοποίησης [8].

Το κριτήριο που εμφανίζεται στην εξίσωση (2.8) δεν είναι όμως υπολογίσιμο καθώς θα προϋπόθετε τον υπολογισμό του $logp(X)$ το οποίο με τη σειρά του προϋποθέτει τον υπολογισμό του ολοκληρώματος της εξίσωσης (2.2). Για αυτό το λόγο χρησιμοποιείται ένα διαφορετικό κριτήριο για την βελτιστοποίηση του προβλήματος που είναι ισοδύναμο με την απόκλιση KL εκτός μιας σταθεράς. Το κριτήριο αυτό ονομάζεται ELBO (Evidence lower bound) και δίνεται από τον τύπο :

$$ELBO(q) = \mathbb{E}[logp(Z, X)] - \mathbb{E}[logq(Z)] \tag{2.9}$$

Εύκολα παρατηρούμε ότι $ELBO(q) = logp(X) - KL(q(Z)||p(Z|X))$ και αφού η απόκλιση KL είναι πάντα θετική το $ELBO(q)$ αποτελεί ένα κάτω φράγμα για το $logp(X)$ ενώ επίσης μπορούμε να ξαναγράψουμε το κριτήριο αυτό ως εξής:

$$\begin{aligned} ELBO(q) &= \mathbb{E}[logp(Z)] + \mathbb{E}[logp(X|Z)] - \mathbb{E}[logq(Z)] \\ &= \mathbb{E}[logp(X|Z)] - KL(q(Z)||p(Z)) \end{aligned}$$

2.2 Δίκτυα Bayes

Στην ενότητα αυτή θα μελετήσουμε τα Δίκτυα Bayes (Bayesian Networks), τα οποία ανήκουν στην οικογένεια των πιθανοθεωρητικών γραφικών μοντέλων και συνδυάζουν αρχές από τη θεωρία γραφημάτων, την θεωρία πιθανοτήτων, την επιστήμη των υπολογιστών και την στατιστική. Είναι μια δομή δεδομένων η οποία επιτρέπει την αναπαράσταση και τον υπολογισμό της από κοινού κατανομής ενός συνόλου τυχαίων μεταβλητών με βάση τις μεταξύ τους εξαρτήσεις.

Ορισμός 2.6. *(Δίκτυα Bayes)*
Ένα δίκτυο Bayes είναι ένας κατευθυνόμενος γράφος όπου κάθε κόμβος αντιστοιχεί σε μια τυχαία μεταβλητή (συνεχή ή διακριτή) και προσανατολισμένοι συνδέσμοι συνδέουν ζεύγη κόμβων. Αν υπάρχει ένας σύνδεσμος από τον κόμβο X στον κόμβο Y, τότε λέμε ότι ο X είναι γονέας του Y και επομένως ο Y εξαρτάται από τον X. Ένα δίκτυο Bayes δεν έχει προσανατολισμένους κύκλους και επομένως είναι ένας προσανατολισμένος ακυκλικός γράφος (directed acyclic graph, DAG). Κάθε κόμβος X_i διαθέτει πληροφορίες πιθανοτήτων, οι οποίες ποσοτικοποιούν την επίδραση των γονέων στον κόμβο χρησιμοποιώντας ένα πεπερασμένο αριθμό παραμέτρων (Οι παράμετροι αυτοί δηλαδή, αφορούν την δεσμευμένη κατανομή κάθε κόμβου δεδομένων του συνόλου των γονέων του). [9]

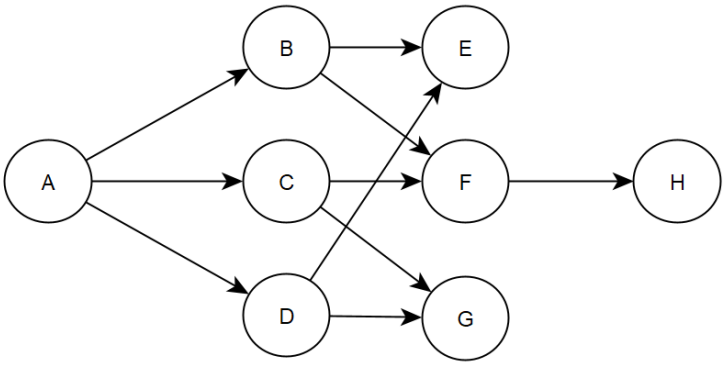
Η βασική χρήση των Δικτύων αυτών εναπόκειται στην ιδιότητα ότι κάθε τυχαία μεταβλητή είναι ανεξάρτητη από τις μεταβλητές που δεν είναι απογόνοι της στον γράφο δεδομένης της κατάστασης όλων των γονέων της. Με αυτό τον τρόπο μειώνεται σημαντικά ο αριθμός των παραμέτρων που χρειάζονται για να χαρακτηρίσουμε την από κοινού κατανομή του συνόλου των τ.μ. Επίσης, μας δίνεται ένας πιο αποδοτικός τρόπος για τον υπολογισμό των εκ

των υστέρων (posterior) πιθανοτήτων δεδομένων κάποιων παρατηρήσεων. [10] Μπορούμε λοιπόν να γράψουμε την από κοινού κατανομή ως εξής :

$$P(X_1, X_2, ..., X_n) = \prod_{i=1}^{n} P(X_i|\pi_i) \tag{2.10}$$

όπου π_i το σύνολο των γονέων της τ.μ. X_i στο δίκτυο Bayes.

Η αναλυτική και ακριβής εύρεση των εκ των υστέρω πιθανοτήτων ακόμα και μέσω των δικτών Bayes είναι πολλές φορές υπολογιστικά μη εφικτή, για αυτό το λόγο έχουν αναπτυχθεί διάφορες προσεγγιστικές μέθοδοι όπως είναι οι αλγόριθμοι τυχαίας δειγματοληψίας (Monte Carlo) που παρέχουν προσεγγιστικές λύσεις σε τέτοιου είδους προβλήματα, η ακρίβεια των οποίων εξαρτάται από το πλήθος δειγμάτων που παράγονται. Μέσω των αλγορίθμων αυτών, που μπορούν να χρησιμοποιηθούν για πληθώρα προβλημάτων όπου στόχος είναι η εκτίμηση ποσοτήτων που είναι δύσκολο ή και αδύνατο να υπολογιστούν με ακρίβεια, μπορούμε να εκτιμήσουμε τις εκ των υστέρων πιθανότητες δειγματοληπτόντας με βάση τις πιθανότητες του δικτύου Bayes και καταμετρώντας τις διαφορετικές απαντήσεις από διαφορετικά συμβάντα.



Σχήμα 2.1: Παράδειγμα ενός Δικτύου Bayes για 8 τυχαίες μεταβλητές. Η τ.μ. Η είναι ανεξάρτητη όλων των υπολοίπων δεδομένης της F. Η από κοινού κατανομή μπορεί να γραφτεί ως εξής $P(A, B, C, D, E, F, G, H) = P(A)P(B|A)P(C|A)P(D|A)P(E|B, D)P(F|B, C)P(G|C, D)P(H|F)$

3 Μηχανική Μάθηση

3.1 Εισαγωγή

Η μηχανική μάθηση είναι ένας κλάδος της τεχνητής νοημοσύνης και περιλαμβάνει τεχνικές που χρονολογούνται από τη δεκαετία του 1950. Τα τελευταία χρόνια, η αυξημένη υπολογιστική ισχύς και η μείωση του κόστους των δεδομένων έχουν καταστήσει τη μηχανική μάθηση ένα πρακτικό και προσιτό εργαλείο.

"A computer program is said to learn from experience E with respect to some class of tasks T, and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E." (Mitchell, T. M. (1997). Machine learning.)

Ο ορισμός αυτός μας επιτρέπει να καταλάβουμε ότι υπάρχουν πολλά διαφορετικά είδη μοντέλων μηχανικής μάθησης, ανάλογα με τη φύση των ενεργειών T που θέλουμε το σύστημα να "μάθει" , τη φύση των μετρικών απόδοσης P που χρησιμοποιούνται για να αξιολογήσουν την επίδοση του συστήματος και τη φύση της εμπειρίας E που δίνουμε εμείς στο σύστημα.

Η ευελιξία που έχουν τα μοντέλα μηχανικής μάθησης, επιτρέπει τη χρήση τους σε ποικίλους τομείς και για διαφορετικούς σκοπούς. Κάποια παραδείγματα εφαρμογών των μοντέλων αυτών είναι:

- Πρόβλεψη επιδημιών : Ανάλυση δεδομένων για την εξάπλωση ασθενειών.
- Ανίχνευση απάτης μέσω αναγνώρισης ύποπτων συναλλαγών στις πιστωτικές κάρτες.
- Συστήματα προτάσεων σε ψυχαγωγικές εφαρμογές (Netflix, Spotify, κλπ) με βάση το προφίλ του κάθε χρήστη και τις μέχρι τώρα προτιμήσεις του.
- Διαφημιστική στόχευση : Προσωποποιημένες διαφημίσεις με βάση το ιστορικό περιήγησης.
- Μεταφράσεις κειμένου
- Πρόβλεψη καιρού
- Αυτόνομα αυτοκίνητα
- Δημιουργία ρεαλιστικών εικόνων, βελτίωση ποιότητας εικόνων, μετατροπή κειμένου σε εικόνα.

Στην ενότητα αυτή θα κάνουμε αρχικά μια σύντομη αναφορά στο διαχωρισμό των μοντέλων μηχανικής μάθησης με βάση το είδος μάθησης που χρησιμοποιείται. Έπειτα θα διακρίνουμε περεταίρω τα μοντέλα αυτά σε διακριτικά (discriminative) και παραγωγικά (generative) και στη συνέχεια θα παρουσιάσουμε αναλυτικά την Αποθορυβοποίηση Πιθανοθεωρητικών Μοντέλων Διάχυσης (Denoising Diffusion Probabilistic Models - DDPMs) που είναι και το βασικό

αντικείμενο αυτής της εργασίας. Ακολούθως, παρουσιάζουμε την Αποθορυβοποίηση Έμμεσων Πιθανοθεωρητικών Μοντέλων Διάχυσης (Denoising Diffusion Implicit Models -DDIMs) και δίνουμε μια δική μας παραλλαγή των DDPM στην ενότητα 3.4.3 της οποίας τα αποτελέσματα θα συγκρίνουμε τελικά με το ήδη υπάρχων μοντέλο. Τέλος στην ενότητα 3.5 κάνουμε μια σύντομη ανασκόπηση άλλων ευρέως διαδεδομένων παραγωγικών μοντέλων , όπως είναι οι Αυτοκωδικοποιητές βασισμένοι σε Μεταβολική Συμπερασματολογία (Variational Autoencoders), τα Παραγωγικά Ανταγωνιστικά Δίκτυα (Generative Adversarial Networks), τα Ενεργειακά Μοντέλα (Energy Based Models) και τα Normalizing Flows, και αναφέρουμε τυχόν ομοιότητες ή διαφορές που παρατηρούμε σε σχέση με τα Μοντέλα Διάχυσης.

3.2 Είδη Μάθησης

Η πιο συχνή κατηγοριοποίηση των μοντέλων μηχανικής μάθησης είναι με βάση το είδος μάθησης που χρησιμοποιείται στο μοντέλο. Υπάρχουν τρεις κύριες κατηγορίες, αν και πολλές φορές συναντάμε και υβριδικές μεθόδους (συνδυασμούς αυτών των κατηγοριών). [11]

3.2.1 Επιβλεπόμενη Μάθηση (Supervised Learning)

Το βασικό χαρακτηριστικό της επιβλεπόμενης μάθησης είναι ότι η εκπαίδευση του μοντέλου προϋποθέτει την ύπαρξη ενός συνόλου δεδομένων εκπαίδευσης με γνωστή έξοδο. Δηλαδή κάθε στοιχείο του συνόλου εκπαίδευσης αποτελείται από ένα σύνολο εισόδου (συνήθως αναφέρεται ως διάνυσμα χαρακτηριστικών) και μια επιθυμητή έξοδο. Στόχος των μοντέλων αυτών είναι να προβλέπουν την έξοδο πανομοιότηπων δεδομένων για τα οποία γνωρίζουμε μόνο το διάνυσμα χαρακτηριστικών. Παραδείγματα προβλημάτων που διαχειρίζεται η επιβλεπόμενη μάθηση είναι η ταξινόμηση (clustering) δεδομένων σε γνωστές κατηγορίες και η παλινδρόμηση (regression). Παραδείγματα αλγορίθμων που χρησιμοποιούνται στα πλαίσια της επιβλεπόμενης μάθησης είναι Δέντρα Απόφασης, Support Vector Machines , Νευρωνικά Δίκτυα, Γραμμική Παλινδρόμηση, Λογιστική Παλινδρόμηση, κ.α.

3.2.2 Μη - Επιβλεπόμενη Μάθηση (Unsupervised Learning)

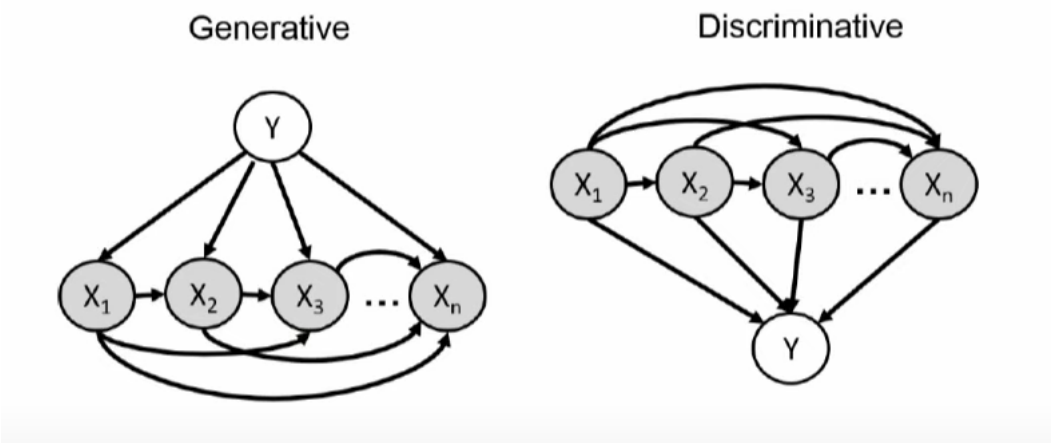
Η μη-επιβλεπόμενη μάθηση διαχειρίζεται δεδομένα των οποίων η "έξοδος" δεν είναι γνωστή. Σκοπός της είναι η αναγνώριση μοτίβων, συστάδων ή συσχετίσεων στα δεδομένα. Η πιο διαδεδομένη υπο-κατηγορία μοντέλων μη επιβλεπόμενης μάθησης είναι η συσταδοποίηση (clustering) η οποία αφορά το διαχωρισμό των δεδομένων σε συστάδες με βάση την ομοιότητα των χαρακτηριστικών τους και για την οποία χρησιμοποιούνται αλγόριθμοι όπως K-means, Ιεραρχική συσταδοποίηση, DBSCAN κ.α.

3.2.3 Ενισχυτική Μάθηση (Reinforcement Learning)

Στην ενισχυτική μάθηση ένα σύστημα (πράκτορας) αλληλεπιδρά με το περιβάλλον εκτελώντας ενέργειες (actions) και λαμβάνοντας ανταμοιβές (rewards). Οι ανταμοιβές αυτές μπορούν να δίνονται άμεσα μετά από κάθε ενέργεια ή να προκύπτουν στο τέλος κάποιας ακολουθίας ενεργειών. Στόχοι της ενισχυτικής μάθησης είναι να εκτιμηθεί η βέλτιστη πολιτική ενεργειών που μεγιστοποιεί την ανταμοιβή και να εκτιμηθεί η κατανομή πιθανότητας των ανταμοιβών.

3.3 Διακριτικά και Παραγωγικά Μοντέλα (Discriminative and Generative Models)

Τα μοντέλα μηχανικής μάθησης μπορούν επίσης να διαχωριστούν σε διακριτικά (discriminative) και παραγωγικά (generative). Η βασική διαφορά μεταξύ αυτών των δύο κατηγοριών είναι ότι τα διακριτικά μοντέλα μαθαίνουν τις διαχωριστικές επιφάνειες απόφασης μεταξύ διαφόρων κλάσεων ενώ τα παραγωγικά μοντέλα μαθαίνουν την κατανομή των δεδομένων σε κάθε κλάση. Συγκεκριμένα, έστω ότι έχουμε κάποια δεδομένα X και τις εξόδους Y. Ένα διακριτικό μοντέλο επικεντρώνεται στο να "μάθει" την κατανομή $p(Y|X)$ με βάση τα δεδομένα εκπαίδευσης, έτσι ώστε όταν μελλοντικά κληθεί να διαχειριστεί νέα δεδομένα, να μπορεί να τα κατατάξει στη σωστή κλάση. Τα διακριτικά μοντέλα δεν έχουν τη δυνατότητα να παράγουν καινούρια δεδομένα. Ένα παραγωγικό μοντέλο αντιθέτως, εκπαιδεύεται στο να μάθει της κατανομές $P(Y), P(X|Y)$ και μετέπειτα μπορεί να χρησιμοποιήσει τον κανόνα του Bayes για να υπολογίσει την κατανομή $P(Y|X)$ για να επιλύσει ένα αντίστοιχο πρόβλημα όπως αυτό του διακριτικού μοντέλου. Το πλεονέκτημα των παραγωγικών μοντέλων είναι ότι εφόσον μαθαίνουν την κατανομή $P(X|Y)$ μπορούν να παράξουν καινούρια δεδομένα που δεν έχουν ξανασυναντήσει, για κάθε μια από τις κλάσεις. [12]



Σχήμα 3.1: Παραγωγικά και Διακριτικά Μοντέλα αναπαριστόμενα ως Bayesian Δίκτυα

Τα πιο βασικά παραδείγματα διακριτικών μοντέλων είναι η λογιστική παλινδρόμηση, τα SVM (Support Vector Machines) και τα Δένδρα απόφασης ενώ παραδείγματα παραγωγικών μοντέλων είναι μεταξύ άλλων τα μοντέλα διά-

χυσης (diffusion models), τα GANs (Generative Adversarial Networks) και τα VAEs(Variational Autoencoders).

Τα διακριτικά μοντέλα χρησιμοποιούνται κυρίως για προβλήματα ταξινόμησης ενώ τα παραγωγικά μοντέλα μπορούν να χρησιμοποιηθούν για παραγωγή κειμένου και εικόνων, σύνθεση ήχου, εμπλουτισμό δεδομένων (data augmentation), ανίχνευση ανώμαλων δεδομένων και άλλα.

3.3.1 Εφαρμογές παραγωγικών μοντέλων σε ιατρικά δεδομένα

Ιδιαίτερα ενδιαφέρουσα είναι η εφαρμογή διαφόρων παραγωγικών μοντέλων σε ιατρικά δεδομένα για ποικίλους σκοπούς. Για παράδειγμα, στον ιατρικό τομέα πολλά σύνολα δεδομένων δεν περιέχουν ισόποσα δεδομένα από όλες τις δυνατές κλάσεις, συνήθως λόγω της σπάνιας φύσης κάποιων παθολογιών. Στην περίπτωση αυτή τα παραγωγικά μοντέλα μπορούν να παράξουν συνθετικά δεδομένα για τις υποεκπροσωπούμενες κλάσεις, ενισχύοντας έτσι την ποιότητα και τη γενικευσιμότητα των αλγορίθμων μηχανικής μάθησης που χρησιμοποιούνται για παράδειγμα είτε για διάγνωση, είτε για πρόβλεψη κάποιας ασθένειας. Παραδείγματα τέτοιων συνόλων αποτελούν τα NIH ChestX-ray14, CheXpert, και ACDC, τα οποία περιλαμβάνουν εικόνες X-ray, MRI και CT με μεγάλη κλινική σημασία αλλά περιορισμένη διαθεσιμότητα. [13] [14]

Επιπλέον, τα παραγωγικά μοντέλα μπορούν να χρησιμοποιηθούν για την προστασία προσωπικών δεδομένων. Σε περιπτώσεις όπου η χρήση πραγματικών ιατρικών δεδομένων είναι απαγορευτική για ερευνητικούς ή νομικούς λόγους, μπορούν να δημιουργήσουν συνθετικά δεδομένα που διατηρούν τα χαρακτηριστικά των αρχικών.

Παράλληλα, με τη χρήση παραγωγικών μοντέλων είναι δυνατή η προεκπαίδευση (pretraining) μοντέλων σε μεγάλους όγκους συνθετικών δεδομένων χαμηλής ανάλυσης (π.χ. 64×64 pixels), με σκοπό την επιτάχυνση της μάθησης και τη μείωση της εξάρτησης από ταξινομημένα δεδομένα. Η προσέγγιση αυτή έχει ιδιαίτερη αξία σε περιβάλλοντα όπου οι ανθρώπινοι πόροι για σχολιασμό είναι περιορισμένοι ή το κόστος της απόκτησης δεδομένων είναι υψηλό.

Τέλος, η παραγωγή συνθετικών ιατρικών δεδομένων έχει εκπαιδευτικές εφαρμογές. Μπορούμε να δημιουργήσουμε μεγάλο αριθμό στιγμιότυπων ιατρικών δεδομένων, τα οποία χρησιμοποιούνται για την εκπαίδευση και εξάσκηση ειδικών, χωρίς να απαιτείται πρόσβαση σε ευαίσθητες πληροφορίες πραγματικών ασθενών.

3.4 Μοντέλα Διάχυσης

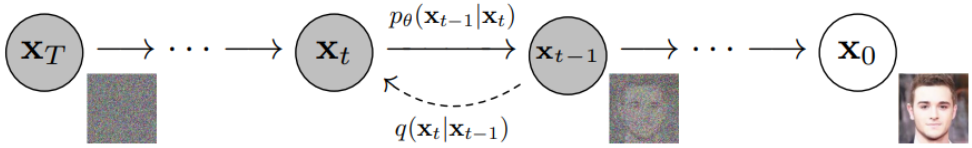
Τα μοντέλα διάχυσης είναι μια κατηγορία βαθιών παραγωγικών μοντέλων (deep generative models), τα οποία τα τελευταία χρόνια έχουν κινήσει το ενδιαφέρον της επιστημονικής κοινότητας και όχι μόνο.

Έχει αποδειχθεί ότι μπορούν να χρησιμοποιηθούν για σύνθεση εικόνων, ακόμα και με διάφορους περιορισμούς ή απαιτήσεις που μπορεί να δοθούν σε μορφή κειμένου, όπως επίσης και στον τομέα της "μηχανικής όρασης" (computer vision).[16] Περιέχουν δύο αλληλένδετες διαδικασίες: μια προκαθορισμένη διαδι-

κασία που ονομάζεται "forward process" (διαδικασία προς τα εμπρός), η οποία αντιστοιχεί τη κατανομή των δεδομένων σε μια απλούστερη κατανομή, συνήθως κάποια γκαουσιανή, και μια αντίστοιχη αντίστροφη διαδικασία που χρησιμοποιεί ένα εκπαιδευμένο νευρωνικό δίκτυο για να αντιστρέψει σταδιακά τις επιπτώσεις της "forward" διαδικασίας. [15]

3.4.1 Αποθορυβωποίηση Πιθανοθεωρητικών Μοντέλων Διάχυσης (Denoising Diffusion Probabilistic Models - DDPMs)

Ένα μοντέλο DDPM κάνει χρήση δύο μαρκοβιανών αλυσίδων για την μοντελοποίηση της προς τα εμπρός (forward) και της αντίστροφης (reverse) διαδικασίας, η μεν διαταράσσοντας τα δεδομένα, προσθέτοντας θόρυβο και η δε μετατρέποντας των θόρυβο πίσω σε δεδομένα. Η προς τα εμπρός διαδικασία (forward process) είναι συνήθως προκαθορισμένη (δεν επιδέχεται εκπαίδευσης) και έχει ως σκοπό τη μετατροπή της κατανομής των δεδομένων σε μια πιο απλή κατανομή. Αντίθετα, η αντίστροφη διαδικασία (reverse process) αντιστρέφει την προς τα εμπρός διαδικασία, μαθαίνοντας να αναγνωρίζει τον θόρυβο σε κάθε στάδιο μέσω ενός βαθιού νευρωνικού δικτύου. Με αυτό τον τρόπο μπορούν να παραχθούν νέα δεδομένα, δειγματίζοντας από την "απλή" κατανομή στην οποία καταλήγει η προς τα εμπρός διαδικασία και έπειτα χρησιμοποιώντας την αντίστροφη διαδικασία για να οδηγηθούμε σε ένα καινούριο δείγμα από την αρχική μας κατανομή. [15]



Σχήμα 3.2: Η Μαρκοβιανή αλυσίδα της προς τα εμπρός (αντίστροφης) διαδικασίας διάχυσης για την παραγωγή ενός δείγματος με σταδιακή πρόσθεση θορύβου (αποθορύβωση). Πηγή : Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. Advances in neural information processing systems, 33, 6840-6851.

Θα ορίσουμε τώρα το μοντέλο μας πιο αυστηρά, βασιζόμενοι στους ορισμούς των Jonathan Ho, Ajay Jain και Pieter Abbeel. [2]

Τα μοντέλα διάχυσης είναι μοντέλα λανθανουσών μεταβλητών της μορφής $p_\theta(\mathbf{x}_0) := \int p_\theta(\mathbf{x}_{0:T})d\mathbf{x}_{1:T}$ όπου οι μεταβλητές $\mathbf{x}_1, ..., \mathbf{x}_T$ είναι λανθάνουσες (latent) μεταβλητές που έχουν την ίδια διάσταση με τα δεδομένα $\mathbf{x}_0$ που ακολουθούν την άγνωστη κατανομή $q(\mathbf{x}_0)$. Η από κοινού κατανομή $p_\theta(\mathbf{x}_{0:T})$ είναι αυτό που ορίσαμε πριν ως αντίστροφη διαδικασία (reverse process) και ορίζεται ως μια μαρκοβιανή αλυσίδα με μεταβάσεις τις οποίες "μαθαίνει" στην περίοδο εκπαίδευσης, ξεκινώντας από την κατανομή $p(x_T) = \mathcal{N}(\mathbf{x}_T; \mathbf{0}, \mathbf{I})$:

$$p_\theta(\mathbf{x}_{0:T}) := p(\mathbf{x}_T) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t), \qquad p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) := \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t)) \quad (3.1)$$

Η προς τα εμπρός διαδικασία (forward process) ορίζεται επίσης ως μια Μαρκοβιανή αλυσίδα, η οποία προσθέτει σταδιακά γκαουσιανό θόρυβο στα δεδομένα με βάση ένα πρόγραμμα διακύμανσης (variance schedule) $\beta_1, ..., \beta_T$:

$$q(\mathbf{x}_{1:T}|\mathbf{x}_0) := \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}), \qquad q(\mathbf{x}_t|\mathbf{x}_{t-1}) := \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}) \tag{3.2}$$

Το πρόγραμμα διακύμανσης, δηλαδή οι διασπορές στην προς τα εμπρός διαδικασία συνήθως ορίζονται ως σταθερές με μικρές τιμές για τις αρχικές χρονικές στιγμές, οι οποίες αυξάνονται σταδιακά με την πάροδο του χρόνου. Η αύξηση μπορεί να γίνει με διαφορετικούς τρόπους, με τις πιο συνήθεις επιλογές να είναι γραμμική, συνημιτονοειδή ή σιγμοειδή αύξηση. Με τον τρόπο που ορίζεται η προς τα εμπρός διαδικασία, μας δίνεται η δυνατότητα να μπορούμε να δειγματίσουμε οποιαδήποτε χρονική στιγμή, έχοντας μόνο την αρχική και χωρίς να πρέπει να περάσουμε σειριακά από όλα τα προηγούμενα βήματα. Αυτό γίνεται με τη βοήθεια της εξής αναπαραμέτρησης στην οποία ορίζουμε $\alpha_t = 1 - \beta_t, \bar{\alpha_t} = \prod_{s=1}^{t} \alpha_s$:

$$\begin{aligned}
\mathbf{x}_t =& \sqrt{1-\beta_t}\mathbf{x}_{t-1} + \sqrt{\beta_t}\boldsymbol{\epsilon}_t^1, && \boldsymbol{\epsilon}_t^1 \sim \mathcal{N}(\mathbf{0}, \mathrm{I}) \\
=& \sqrt{(1-\beta_t)(1-\beta_{t-1})}\mathbf{x}_{t-2} + \sqrt{\beta_{t-1}(1-\beta_t)+\beta_t}\boldsymbol{\epsilon}_t^2, && \boldsymbol{\epsilon}_t^2 \sim \mathcal{N}(\mathbf{0}, \mathrm{I}) \\
=& ... \\
=& \sqrt{(1-\beta_t)(1-\beta_{t-1})...(1-\beta_1)}\mathbf{x}_0 \\
&+ \sqrt{\beta_1(1-\beta_2)...(1-\beta_{t-1})(1-\beta_t) + ... + \beta_{t-1}(1-\beta_t)+\beta_t}\boldsymbol{\epsilon}_t^t, && \boldsymbol{\epsilon}_t^t \sim \mathcal{N}(\mathbf{0}, \mathrm{I}) \\
=& \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_t^t, && \boldsymbol{\epsilon}_t^t \sim \mathcal{N}(\mathbf{0}, \mathrm{I})
\end{aligned}$$

Επομένως καταλήγουμε στην εξής δεσμευμένη κατανομή ως προς την αρχική κατάσταση (από την οποία μπορούμε να δειγματίσουμε εύκολα) :

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I}) \tag{3.3}$$

Αφού η από κοινού κατανομή των $\mathbf{x}_{1:T}$ δεσμευμένης της $\mathbf{x}_0$ είναι κανονική, οι εκ των υστέρων κατανομές της προς τα εμπρός διαδικασίας θα είναι επίσης κανονικές και μπορούμε να υπολογίσουμε την μέση τιμή και τον πίνακα συνδιακύμανσης τους ως εξής :

$$\begin{aligned}
q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) &= q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_0)\frac{q(\mathbf{x}_t|\mathbf{x}_0)}{q(\mathbf{x}_{t-1}|\mathbf{x}_0)} \\
&\propto exp\Big(-\frac{1}{2}\big(\frac{(\mathbf{x}_t - \sqrt{\alpha}_t\mathbf{x}_{t-1})^2}{\beta_t} + \frac{(\mathbf{x}_{t-1} - \sqrt{\bar{\alpha_{t-1}}}\mathbf{x}_0)^2}{1-\bar{\alpha_{t-1}}} - \frac{(\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{x}_0)^2}{1-\bar{\alpha}_t}\big)\Big) \\
&= exp\Big(-\frac{1}{2}\big((\frac{\alpha_t}{\beta_t} + \frac{1}{1-\bar{\alpha_{t-1}}})\mathbf{x}_{t-1}^2 - 2(\frac{\sqrt{\alpha_t}}{\beta_t}\mathbf{x}_t + \frac{\sqrt{\bar{\alpha_{t-1}}}}{1-\bar{\alpha_{t-1}}}\mathbf{x}_0)\mathbf{x}_{t-1} + C(\mathbf{x}_t, \mathbf{x}_0)\big)\Big)
\end{aligned}$$

Και άρα καταλήγουμε στο ότι :

$$\tilde{\beta}_t = 1/(\frac{\alpha_t}{\beta_t} + \frac{1}{1-\bar{\alpha_{t-1}}}) = \frac{1-\bar{\alpha_{t-1}}}{\alpha_t * (1-\bar{\alpha_{t-1}}) + \beta_t} * \beta_t = \frac{1-\bar{\alpha_{t-1}}}{1-\bar{\alpha}_t} * \beta_t$$

$$\begin{aligned}\tilde{\boldsymbol{\mu}}(\mathbf{x}_t, \mathbf{x}_0) &= (\frac{\sqrt{\alpha_t}}{\beta_t}\mathbf{x}_t + \frac{\sqrt{\bar{\alpha_{t-1}}}}{1-\bar{\alpha_{t-1}}}\mathbf{x}_0)/\frac{1-\bar{\alpha_{t-1}}}{1-\bar{\alpha}_t} * \beta_t \\ &= \frac{\sqrt{\alpha_t}(1-\bar{\alpha_{t-1}})}{1-\bar{\alpha}_t}\mathbf{x}_t + \frac{\sqrt{\bar{\alpha_{t-1}}}\beta_t}{1-\bar{\alpha}_t}\mathbf{x}_0\end{aligned}$$

Η εκ των υστέρων κατανομή της προς τα εμπρός διαδικασίας είναι επομένως η εξής:

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_{t-1}; \tilde{\boldsymbol{\mu}}(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t\mathbf{I}) \tag{3.4}$$

Ο στόχος πάνω στον οποίο εκπαιδεύεται το μοντέλο μας είναι η ελαχιστοποίηση της διασταυρούμενης εντροπίας της κατανομής $p_\theta(\mathbf{x}_0)$ ως προς την κατανομή $q(\mathbf{x}_0)$, δηλαδή η ελαχιστοποίηση της μέσης αρνητικής λογαριθμοποιημένης πιθανοφάνειας της κατανομής $p_\theta(\mathbf{x}_0)$ ως προς την κατανομή $q(\mathbf{x}_0)$:

$$\mathbb{E}_{q(\mathbf{x}_0)}[-logp_\theta(\mathbf{x}_0)] = \mathbb{E}_{q(\mathbf{x}_0)}[-log\big(\int p_\theta(\mathbf{x}_{0:T})d\mathbf{x}_{1:T}\big)] \tag{3.5}$$

$$= \mathbb{E}_{q(\mathbf{x}_0)}[-log\big(\int \frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}q(\mathbf{x}_{1:T}|\mathbf{x}_0)d\mathbf{x}_{1:T}\big)] \tag{3.6}$$

$$= \mathbb{E}_{q(\mathbf{x}_0)}[-log\big(\mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}\frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}\big)] \tag{3.7}$$

$$\text{(Ανισότητα Jensen)} \leq \mathbb{E}_{q(\mathbf{x}_{0:T})}\big[-log\frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}\big] \tag{3.8}$$

$$= \mathbb{E}_q\big[-log\frac{p(\mathbf{x}_T)\prod_{t=1}^T p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{\prod_{t=1}^T q(\mathbf{x}_t|\mathbf{x}_{t-1})}\big] \tag{3.9}$$

$$= \mathbb{E}_q\big[-logp(\mathbf{x}_T) - \sum_{t\geq 1} log\frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{x}_{t-1})}\big] \tag{3.10}$$

$$= \mathbb{E}_q\big[-logp(\mathbf{x}_T) - \sum_{t\geq 1} log\frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)q(\mathbf{x}_t|\mathbf{x}_0)}{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)q(\mathbf{x}_{t-1}|\mathbf{x}_0)}\big] \tag{3.11}$$

$$= \mathbb{E}_q\big[-log\frac{p(\mathbf{x}_T)}{q(\mathbf{x}_T|\mathbf{x}_0)} - \sum_{t>1} log\frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)} - logp_\theta(\mathbf{x}_0|\mathbf{x}_1)\big] \tag{3.12}$$

Με τη βοήθεια του ορισμού της απόκλισης Kullback-Leibler του ορισμού (2.5) και της εξίσωσης (2.7) μπορούμε να ξαναγράψουμε τον στόχο εκπαίδευσης της (3.12) ως εξής :

$$\mathbb{E}_q\big[\underbrace{D_{\mathrm{KL}}(q(x_T|x_0)\|p(x_T))}_{L_T} + \sum_{t>1}\underbrace{D_{\mathrm{KL}}(q(x_{t-1}|x_t, x_0)\|p_\theta(x_{t-1}|x_t))}_{L_{t-1}} - \underbrace{\log p_\theta(x_0|x_1)}_{L_0}\big] \tag{3.13}$$

Παράρτημα : Υπολογισμός Απόκλισης KL για Γκαουσιανές κατανομές

Υπενθυμίζουμε αρχικά ότι η πυκνότητα πιθανότητας μιας πολυδιάστατης κανονικής κατανομής με μέση τιμή $\boldsymbol{\mu}$ και πίνακα συνδιακύμανσης Σ δίνεται από τον τύπο (n η διάσταση του τυχαίου διανύσματος):

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{n/2} det(\Sigma)^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right) \tag{3.14}$$

Ανακαλούμε τον ορισμό (2.5) και την εξίσωση (2.7) για το πως ορίζεται η απόκλιση KL μεταξύ δύο κατανομών p και q. Αν τώρα η p και q είναι κανονικές κατανομές $\mathcal{N}(\boldsymbol{\mu}_1, \Sigma_1)$ και $\mathcal{N}(\boldsymbol{\mu}_2, \Sigma_2)$ αντίστοιχα, τότε μπορούμε να υπολογίσουμε την ποσότητα αυτή με κλειστό τύπο ως εξής :

$$\begin{aligned}
D_{KL}(p||q) &= \mathbb{E}_p[\log \frac{p}{q}] \\
&= \mathbb{E}_p[\log\, p - \log\, q] \\
(3.14) &= \frac{1}{2}\mathbb{E}_p\big[- \log\, \det(\Sigma_1) - (\mathbf{x}-\boldsymbol{\mu}_1)^T\Sigma_1^{-1}(\mathbf{x}-\boldsymbol{\mu}_1) + \log\, \det(\Sigma_2) \\
&+ (\mathbf{x}-\boldsymbol{\mu}_2)^T\Sigma_2^{-1}(\mathbf{x}-\boldsymbol{\mu}_2)\big] \\
&= \frac{1}{2}\log \frac{\det\Sigma_2}{\det\Sigma_1} \\
&+ \frac{1}{2}\mathbb{E}_p\left[-\mathrm{tr}(\Sigma_1^{-1}(\mathbf{x}-\boldsymbol{\mu}_1)(\mathbf{x}-\boldsymbol{\mu}_1)^T) + \mathrm{tr}(\Sigma_2^{-1}(\mathbf{x}-\boldsymbol{\mu}_2)(\mathbf{x}-\boldsymbol{\mu}_2)^T)\right] \\
&= \frac{1}{2}\log \frac{\det\Sigma_2}{\det\Sigma_1} + \frac{1}{2}\left(-\mathrm{tr}(\Sigma_1^{-1}\Sigma_1) + \mathbb{E}_p\left[\mathrm{tr}(\Sigma_2^{-1}(\mathbf{x}\mathbf{x}^T - 2\mathbf{x}\boldsymbol{\mu}_2^T + \boldsymbol{\mu}_2\boldsymbol{\mu}_2^T)\right]\right) \\
&= \frac{1}{2}\left(\log \frac{\det\Sigma_2}{\det\Sigma_1} - n + tr(\Sigma_2^{-1}(\Sigma_1 + \boldsymbol{\mu}_1\boldsymbol{\mu}_1^T - 2\boldsymbol{\mu}_2\boldsymbol{\mu}_1^T + \boldsymbol{\mu}_2\boldsymbol{\mu}_2^T))\right) \\
&= \frac{1}{2}\left(\log \frac{\det\Sigma_2}{\det\Sigma_1} - n + tr(\Sigma_2^{-1}\Sigma_1) + (\boldsymbol{\mu}_2-\boldsymbol{\mu}_1)^T\Sigma_2^{-1}(\boldsymbol{\mu}_2-\boldsymbol{\mu}_1)\right)
\end{aligned}$$

Επομένως έχουμε την εξής κλειστή μορφή για τον υπολογισμό της απόκλισης KL μεταξύ δύο κατανομών p και q :

$$D_{KL}(p||q) = \frac{1}{2}\left(\log \frac{\det\Sigma_2}{\det\Sigma_1} - n + tr(\Sigma_2^{-1}\Sigma_1) + (\boldsymbol{\mu}_2-\boldsymbol{\mu}_1)^T\Sigma_2^{-1}(\boldsymbol{\mu}_2-\boldsymbol{\mu}_1)\right) \tag{3.15}$$

Μπορούμε τώρα να δούμε πιο αναλυτικά τους όρους που εμφανίζονται στην εξίσωση (3.13). Ο όρος L_T δεν περιέχει παραμέτρους που μεταβάλλονται και επομένως μπορεί να θεωρηθεί ως σταθερά κατά την περίοδο εκπαίδευσης, δεδομένου ότι οι παράμετροι β_t είναι σταθερές κατά τη διάρκεια της εκπαίδευσης.

Θα ασχοληθούμε επίσης με τους όρους $L_{1:T-1}$, αλλά πρώτα πρέπει να καθορίσουμε τις επιλογές για την κατανομή $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) := \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t))$. Παρόλο που οι πίνακες συνδιασποράς μπορούν επίσης να αποτελούν παράμετρούς που μαθαίνουμε μέσω της εκπαίδευσης, στο [2] οι συγγραφείς , για λόγους

απλότητας του μοντέλου, τους ορίζουν ως μη μεταβαλλόμενες σταθερές που εξαρτούνται από το χρόνο και έχουν τη μορφή $\mathbf{\Sigma}_\theta(\mathbf{x}_t, t) = \sigma_t^2\mathbf{I}$. Αναφέρουν επίσης ότι πειραματικά οι επιλογές $\sigma_t^2 = \beta_t$ και $\sigma_t^2 = \tilde{\beta}_t = \frac{1-\bar{\alpha_{t-1}}}{a-\bar{\alpha_t}}\beta_t$ δίνουν παρόμοια αποτελέσματα. Η πρώτη επιλογή είναι βέλτιστη όταν τα αρχικά μας δεδομένα ακολουθούν τυποποιημένη πολυδιάστατη κανονική κατανομή ($\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$) και η δεύτερη επιλογή είναι βέλτιστη για όταν το $\mathbf{x}_0$ ορίζεται ντετερμινιστικά.

Έχοντας λοιπόν την εξής κατανομή για την αντίστροφη διαδικασία $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) := \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \sigma_t^2\mathbf{I})$ και με βάση την εξίσωση (3.15) για τον υπολογισμό των αποκλίσεων KL μεταξύ γκαουσιανών κατανομών, μπορούμε να ξαναγράψουμε το L_{t-1} ως εξής :

$$L_{t-1} = \mathbb{E}_q\left[\frac{1}{2\sigma_t^2}||\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) - \boldsymbol{\mu}_\theta(\mathbf{x}_t, t)||^2\right] + C \tag{3.16}$$

όπου το C είναι σταθερά που δεν εξαρτάται από το θ. Είναι φανερό λοιπόν ότι θα πρέπει η μέση τιμή της αντίστροφης διαδικασίας $\boldsymbol{\mu}_\theta(\mathbf{x}_t, t)$ να εκτιμά την εκ των υστέρων μέση τιμή της προς τα εμπρός διαδικασίας $\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0)$. Γράφοντας με τη βοήθεια της εξίσωσης (3.3) την $\mathbf{x}_t$ ως $\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$, για $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$ μπορούμε να γράψουμε $\mathbf{x}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(\mathbf{x}_t(\mathbf{x}_0, \epsilon) - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon})$:

$$L_{t-1} - C = \mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\left[\frac{1}{2\sigma_t^2}||\tilde{\boldsymbol{\mu}}_t\left(\mathbf{x}_t(\mathbf{x}_0, \epsilon), \frac{1}{\sqrt{\bar{\alpha}_t}}(\mathbf{x}_t(\mathbf{x}_0, \epsilon) - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon})\right) - \boldsymbol{\mu}_\theta(\mathbf{x}_t(\mathbf{x}_0, \epsilon), t)||^2\right] \tag{3.17}$$

$$= \mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\left[\frac{1}{2\sigma_t^2}||\frac{1}{\sqrt{\bar{\alpha}_t}}\left(\mathbf{x}_t(\mathbf{x}_0, \epsilon) - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}\right) - \boldsymbol{\mu}_\theta(\mathbf{x}_t(\mathbf{x}_0, \epsilon), t)||^2\right] \tag{3.18}$$

Αφού η τιμή της $\mathbf{x}_t$ είναι διαθέσιμη κατά την εκτίμηση της $\mathbf{x}_{t-1}$ μπορούμε να παραμετροποιήσουμε την $\boldsymbol{\mu}_\theta$ ως εξής :

$$\boldsymbol{\mu}_\theta(\mathbf{x}_t, t) = \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \frac{1}{\sqrt{\bar{\alpha}_t}}(\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_\theta)) = \frac{1}{\sqrt{\bar{\alpha}_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta)\right) \tag{3.19}$$

όπου $\boldsymbol{\epsilon}_\theta$ είναι μια συνάρτηση που εκτιμά την $\boldsymbol{\epsilon}$ δεδομένης της $\mathbf{x}_t$. Με την παραμετροποίηση της εξίσωσης (3.19) η (3.18) απλοποιείται σε:

$$\mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\left[\frac{\beta_t^2}{2\sigma_t^2\alpha_t(1-\bar{\alpha}_t)}||\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t)||^2\right] \tag{3.20}$$

Οι συγγραφείς αναφέρουν επίσης ότι πειραματικά η χρήση μιας απλοποιημένης αντικειμενικής συνάρτησης, που αγνοεί τα βάρη που προσθέτουν οι παράμετροι $\sigma_t, \bar{\alpha}_t, \beta_t$ και δίνεται από τον τύπο:

$$L_{simple} = \mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\left[||\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t)||^2\right] \tag{3.21}$$

επέφερε δείγματα καλύτερης ποιότητας μετά το πέρας της εκπαίδευσης.

Οι αλγόριθμοι εκπαίδευσης και δειγμάτισης για το συγκεριμένο μοντέλο παρουσιάζονται παρακάτω. Κάποιες από τις έννοεις που περιέχονται στους αλγορίθμους (όπως ο αλγόριθμος gradient descent) θα τις μελετήσουμε παρακάτω.

Algorithm 1 Training

1: **repeat**
2: $\mathbf{x}_0 \sim q(\mathbf{x}_0)$
3: $t \sim \text{Uniform}(\{1, \dots, T\})$
4: $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
5: Take gradient descent step on
$$\nabla_\theta \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t) \right\|^2$$
6: **until** converged

Algorithm 2 Sampling

1: $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
2: **for** $t = T, \dots, 1$ **do**
3: $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
4: $\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) \right) + \sigma_t \mathbf{z}$
5: **end for**
6: **return** $\mathbf{x}_0$

Σχήμα 3.3: Αλγόριθμοι εκπαίδευσης και δειγμάτισης για μοντέλα DDPM. Πηγή : Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. Advances in neural information processing systems, 33, 6840-6851.

3.4.2 Αποθορυβοποίηση Έμμεσων Πιθανοθεωρητικών Μοντέλων Διάχυσης (Denoising Diffusion Implicit Models - DDIMs)

Τα DDIMs (Αποθορυβοποίηση Έμμεσων Πιθανοθεωρητικών Μοντέλων Διάχυσης) είναι Μοντέλα Διάχυσης που προτάθηκαν μετά από τα DDPMs με κυριότερο σκοπό να μειώσουν σε σημαντικό βαθμό το χρόνο που χρειάζεται για τη παραγωγή δειγμάτων στα DDPMs. [17] Συγκεκριμένα παρόλο που τα DDPMs επιδεικνύουν αποτελέσματα συγκρίσιμα με αυτά τον GANs (Παράγραφος 3.5.2), η παραγωγή 50 000 δειγμάτων μεγέθους 32*32 από ένα μοντέλο DDPM, θα χρειαζόταν 300 φορές πιο πολύ χρόνο από ότι από ένα μοντέλο GAN. Αυτό οφείλεται στο γεγονός ότι η προς τα εμπρός και η προς τα πίσω διαδικασία (forward and reverce processes) μοντελοποιούνται ως μαρκοβιανές αλυσίδες, και η παραγωγή κάθε δείγματος πρέπει να περάσει από συνολικά T χρονικές στιγμές, επομένως για την παραγωγή ενός και μόνο δείγματος χρειαζόμαστε T διαπεράσεις του νευρωνικού δικτύου. Καταλαμβαίνουμε λοιπόν ότι όταν το T επιλέγεται να είναι μεγάλο, το οποίο είναι επιθυμητό ώστε η προς τα εμπρός διαδικασία να συγκλίνει σε μια τυποποιημένη κανονική κατανομή, ο χρόνος παραγωγής δειγμάτων αυξάνεται ραγδαία.

Οι Song, Meng και Ermon λοιπόν, προτείνουν τα DDIMs, επεκτείνοντας αρχικά τα DDPMs. Η προς τα επρός διαδικασία επιλέγεται από μια οικογένεια μη μαρκοβιανών κατανομών $\mathcal{Q}$, που ορίζονται με βάση ένα πραγματικό διάνυσμα $\sigma \in \mathbb{R}^T_{\geq 0}$ και ικανοποιούν τα παρακάτω :

$$q_\sigma(\mathbf{x}_{1:T}) := q_\sigma(\mathbf{x}_T|\mathbf{x}_0) \prod_{t=2}^{T} q_\sigma(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \tag{3.22}$$

$$q_\sigma(\mathbf{x}_T|\mathbf{x}_0) = \mathcal{N}(\sqrt{\bar{\alpha}_T}\mathbf{x}_0, (1-\bar{\alpha}_T)\boldsymbol{I}) \tag{3.23}$$

$$q_\sigma(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathcal{N}\left(\sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_{t-1}-\sigma_t^2} \frac{\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{x}_0}{\sqrt{1-\bar{\alpha}_t}}, \sigma_t^2 \boldsymbol{I} \right), \forall t > 1 \tag{3.24}$$

Επιλέγοντας την οικογένεια των κατανομών $\mathcal{Q}$, να ορίζεται με αυτό τον τρόπο, οι συγγραφείς καταφέρνουν να έχουν την εξής δεσμευμένη κατανομή $q_\sigma(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\boldsymbol{I})$, $\forall t$ και επομένως έχουμε ακριβώς την ίδια μορφή όπως και στα DDPMs.

Παράρτημα: Απόδειξη του παραπάνω ισχυρισμού

Θα αποδείξουμε το συγκεκριμένο ισχυρισμό χρησιμοποιώντας επαγωγή. Γνωρίζουμε ότι για $t = T$ ο ισχυρισμός ισχύει καθώς είναι μια από τις απαιτήσεις που πρέπει να ικανοποιεί μια κατανομή για να ανήκει στην οικογένεια $\mathcal{Q}$. Έστω ότι ο ισχυρισμός ισχύει για t, θα αποδείξουμε ότι ισχύει για t-1.
Γνωρίζουμε λοιπόν ότι:

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, \boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{I})$$

$$\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_{t-1}-\sigma_t^2}\frac{\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{x}_0}{\sqrt{1-\bar{\alpha}_t}} + \sigma_t\boldsymbol{\epsilon}', \boldsymbol{\epsilon}' \sim \mathcal{N}(0, \boldsymbol{I})$$

Επομένως:

$$\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_{t-1}-\sigma_t^2}\frac{\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon} - \sqrt{\bar{\alpha}_t}\mathbf{x}_0}{\sqrt{1-\bar{\alpha}_t}} + \sigma_t\boldsymbol{\epsilon}'$$
$$\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_{t-1}}\boldsymbol{\epsilon}'', \boldsymbol{\epsilon}'' \sim \mathcal{N}(0, \boldsymbol{I})$$

Άρα:

$$q(\mathbf{x}_{t-1}|\mathbf{x}_0) = \mathcal{N}(\sqrt{\bar{\alpha}}_{t-1}\mathbf{x}_0, (1-\bar{\alpha}_{t-1})\boldsymbol{I})$$

Η προς τα πίσω διαδικασία (reverse/generative process) στα DDIMs μοντελοποιείται ώστε να εκτιμά την εκ των υστέρων κατανομή της προς τα εμπρός διαδικασίας $q_\sigma(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$, με το νευρωνικό δίκτυο να εκτιμά όπως και στα DDPMs το θόρυβο που προστίθεται σε κάθε χρονική στιγμή σε σχέση με την αρχική εικόνα. Ορίζονοτας λοιπόν την εξής συνάρτηση:

$$f_\theta^{(t)}(\mathbf{x}_t) := (\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_\theta^{(t)}(\mathbf{x}_t)/\sqrt{\bar{\alpha}_t}) \tag{3.25}$$

όπου $\boldsymbol{\epsilon}_\theta^{(t)}(\mathbf{x}_t)$ η εκτίμηση του νευρωνικού δικτύου, ορίζουμε την προς τα πίσω διαδικασία ως εξής:

$$p_\theta^{(t)}(\mathbf{x}_{t-1} \mid \mathbf{x}_t) = \begin{cases} \mathcal{N}(f_\theta^{(1)}(\mathbf{x}_1), \sigma_1^2\mathbf{I}) & \text{if } t = 1 \\ q_\sigma(\mathbf{x}_{t-1} \mid \mathbf{x}_t, f_\theta^{(t)}(\mathbf{x}_t)) & \text{otherwise} \end{cases} \tag{3.26}$$

Εύκολα αποδεικνείεται ότι για οποιοδήποτε διάνυσμα $\sigma \in \mathbb{R}_{\geq 0}^T$, η ανάλυση της συνάρτησης κόστους δίνει μια αντικειμενική συνάρτηση ισοδύναμη με αυτή των DDPMs εκτός μιας σταθεράς. Δηλαδή εκπαιδεύουμε με ακριβώς τον ίδιο τρόπο όπως και στα DDPM και με την ίδια συνάρτηση κόστους.

Αυτό είναι και το σημαντικό πλεονέκτημα των DDIM, ότι δηλαδή μπορούν να χρησιμοποιήσουν ήδη εκπαίδευμενα μοντέλα DDPM και να εφαρμόσουν απλά την δική τους διαδικασία παραγωγής δεδομένων για πιο γρήγορη δειγμάτιση. Ειδικότερα DDIM ονομάζονται τα μοντέλα για τα οποία $\sigma_t = 0, \forall t$, τα μοντέλα δηλαδή που γίνονται ντετερμινιστικά δεδομένων των $\mathbf{x}_t$ και $\mathbf{x}_0$. Αυτή

η επιλογή έχει ως αποτέλεσμα να χάνεται η στοχαστικότητα που υπήρχε στην προς τα πίσω διαδικασία των DDPMs και πλέον η διαδικασία παραγωγής να είναι ντετερμινιστική, δηλαδή ξεκινώντας από τον ίδιο γκαουσιανό θόρυβο θα καταλήξουμε στο ακριβώς ίδιο δείγμα, κάτι που δεν συνέβαινε στα DDPMs.

Η επιλογή αυτή έχει ως αποτέλεσμα ότι λόγω της ντετερμινιστικής φύσης της προς τα πίσω διαδικασίας μπορούμε να μειώσουμε τα χρονικά βήματα που χρησιμοποιούνται στην προς τα πίσω διαδικασία, θυσιάζοντας σε συγκριτικά μικρό βαθμό την ποιότητα των παραγώμενων δειγμάτων. Για την δειγμάτιση χρησιμοποιείται ο εξής κανόνας:

$$\mathbf{x}_{\tau_{i-1}} = \sqrt{\bar{\alpha}_{\tau_{i-1}}} \left(\frac{\mathbf{x}_\tau - \sqrt{1-\bar{\alpha}_{\tau_i}} \boldsymbol{\epsilon}_\theta^{(\tau_i)}(\mathbf{x}_{\tau_i})}{\sqrt{\bar{\alpha}_{\tau_i}}} \right) + \sqrt{1-\bar{\alpha}_{\tau_i}} \boldsymbol{\epsilon}_\theta^{(\tau_i)}(\mathbf{x}_{\tau_i}) \tag{3.27}$$

όπου $\{\tau\}_i$ μια υπακολουθία του $\{1, ..., T\}$.

Σημαντικό είναι επίσης να αναφέρουμε ότι για $\sigma_t = \sqrt{\frac{(1-\bar{\alpha}_{t-1})}{(1-\bar{\alpha}_t)}} \sqrt{1 - \frac{\bar{\alpha}_t}{\bar{\alpha}_{t-1}}}$ η μοντελοποίηση που ορίσαμε παραπάνω συμπίπτει με αυτή των DDPM και επομένως τα DDPM είναι μια ειδική περίπτωση κατανομής που ανήκει στην οικογένεια $\mathcal{Q}$.

Οι συγγραφείς έδειξαν ότι ακόμα και η χρήση 50 χρονικών στιγμών αντί για 1000 που χρησιμοποιούνται στα DDPM δίνει αρκετά καλά αποτελέσματα και μειώνει τον χρόνο δειγμάτισης κατά 20 φορές. Επίσης σύγκριναν τα αποτελέσματα τους με DDPM εκπαιδευόμενα σε αντίστοιχα πλήθη χρονικών στιγμών, και έδειξαν ότι τα αποτελέσματα των DDIM είναι σημαντικά καλύτερα. Παρόλα αυτά με τη χρήση όλων των βημάτων στη διαδικασία παραγωγής δειγμάτων τα DDPM αποδίδουν καλύτερα από ότι τα DDIM. Στο σχήμα 3.4 απεικονίζονται τα πειράματα που έκαναν οι συγγραφείς με την περίπτωση $\eta = 0$ να αντιστοιχεί στα DDIM και την περίπτωση $\eta = 1$ και $\hat{\sigma}$ να αντιστοιχούν στα DDPM. Η μετρική που χρησιμοποιείται είναι το FID, το οποίο μελετάμε στην παράγραφο 6.2 και στο οποίο καλύτερες τιμές είναι οι μικρότερες τιμές.

		CIFAR10 (32 × 32)					CelebA (64 × 64)				
	S	10	20	50	100	1000	10	20	50	100	1000
η	0.0	**13.36**	**6.84**	**4.67**	**4.16**	4.04	**17.33**	**13.73**	**9.17**	**6.53**	3.51
	0.2	14.04	7.11	4.77	4.25	4.09	17.66	14.11	9.51	6.79	3.64
	0.5	16.66	8.35	5.25	4.46	4.29	19.86	16.06	11.01	8.09	4.28
	1.0	41.07	18.36	8.01	5.78	4.73	33.12	26.03	18.48	13.93	5.98
	$\hat{\sigma}$	367.43	133.37	32.72	9.99	**3.17**	299.71	183.83	71.71	45.20	**3.26**

Σχήμα 3.4: Σύγκριση επίδοσης DDIM και DDPM. Μετρική : FID Πηγή : Song, J., Meng, C., & Ermon, S. (2020). Denoising diffusion implicit models. arXiv preprint arXiv:2010.02502.

Συμπερασματικά, Τα DDIMs αποτελούν μια ντετερμινιστική παραλλαγή των DDPMs, η οποία επιτρέπει σημαντικά ταχύτερη παραγωγή δειγμάτων χωρίς ουσιαστική απώλεια ποιότητας. Η βασική τους καινοτομία είναι η απαλοιφή της στοχαστικότητας από την προς τα πίσω διαδικασία διάχυσης, επιτυγχάνοντας ακριβή επανακατασκευή με λιγότερα χρονικά βήματα.

3.4.3 Τροποποιημένη Αποθορυβοποίηση Πιθανοθεωρητικών Μοντέλων Διάχυσης (Modified Denosing Diffusion Probabilistic Models - MDDPM)

Στην παρούσα ενότητα μελετάται μια δική μας τροποποίηση της μεθόδου που παρουσιάζεται στην Ενότητα 3.4.1 κατά την οποία οι δύο διαδικασίες (προς τα εμπρός (forward process) και αντίστροφη (reverse process) δεν εξαρτόνται πλέον μόνο από το προηγούμενο βήμα, αλλά από τα δύο προηγούμενα βήματα.

Διαισθητικά, διερευνούμε την πιθανότητα η δέσμευση σε περισότερα από ένα προηγούμενα βήματα, να επιφέρει καλύτερα και πιο γρήγορα αποτελέσματα, χωρίς όμως την αύξηση της υπολογιστικής πολυπλοκότητας της μεθόδου. Από την ανασκόπηση της σχετικής βιβλιογραφίας δεν προκύπτει κάποια προηγούμενη αναφορά σε τέτοια παραλλαγή του μοντέλου, γεγονός που μας οδηγεί να θεωρήσουμε την παρούσα ανάλυση ως πρωτότυπη. (Θα συζητήσουμε στην παράγραφο 3.4.4 γιατί αυτή η παραλλαγή, παρόλου που μοντελοποιείται ως μια μη μαρκοβιανή προς τα εμπρός διαδικασία, δεν εμπίπτει στην οικογένεια κατανομών $\mathcal{Q}$, που ορίζεται στα DDIMs).

Μοντελοποιούμε λοιπόν την προς τα εμπρός διαδικασία (forward process) η οποία προσθέτει σταδιακά γκαουσιανό θόρυβο στα δεδομένα με βάση ένα πρόγραμμα διακύμανσης (variance schedule) $\beta_1, ..., \beta_T$ ως εξής:

$$q(\mathbf{x}_{1:T}|\mathbf{x}_0) := q(\mathbf{x}_1|\mathbf{x}_0) \prod_{t=2}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_{t-2}) \tag{3.28}$$

$$q(\mathbf{x}_1|\mathbf{x}_0) := \mathcal{N}(\mathbf{x}_1; \sqrt{1-\beta_1}\mathbf{x}_0, \beta_1\mathbf{I}) \tag{3.29}$$

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_{t-2}) := \mathcal{N}(\mathbf{x}_t; d_t^I\mathbf{x}_{t-1} + d_t^{II}\mathbf{x}_{t-2}, \beta_t\mathbf{I}) \ t = 2, ..., T \tag{3.30}$$

Η οριακή κατανομή στην οποία θα συγκλίνει το ζευγάρι $(\mathbf{x}_t, \mathbf{x}_{t-1})$ καθώς $t \to \infty$ ορίζουμε ότι είναι η $\mathcal{N}\left(\vec{0}, \begin{bmatrix} \mathbf{I}_{n\times n} & \rho\mathbf{I}_{n\times n} \\ \rho\mathbf{I}_{n\times n} & \mathbf{I}_{n\times n} \end{bmatrix}\right)$ όπου η συνδιασπορά ρ μπορεί να θεωρηθεί ως μια υπερπαράμετρος του μοντέλου. Επιβάλουμε επίσης τους περιορισμούς ότι κάθε λανθάνουσα μεταβλητή θα πρέπει να έχει διασπορά 1 και συνδιασπορά με το προηγούμενο βήμα ίση με ρ, επομένως γράφοντας το $\mathbf{x}_t$ ως:

$$\mathbf{x}_t = d_t^I\mathbf{x}_{t-1} + d_t^{II}\mathbf{x}_{t-2} + \sqrt{\beta_t}\boldsymbol{\epsilon} \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\vec{0}, \mathbf{I}) \tag{3.31}$$

Έχουμε:

$$Var(\mathbf{x}_t) = 1, \forall t \in 1, ..., T \iff (d_t^I)^2 + (d_t^{II})^2 + 2\rho d_t^I d_t^{II} + \beta_t = 1$$

$$\begin{aligned} Cov(\mathbf{x}_t, \mathbf{x}_{t-1}) = \rho, \forall t \in 2, ..., T &\iff \rho = d_t^I + d_t^{II}\rho \\ &\iff d_t^I = \rho(1 - d_t^{II}) \end{aligned}$$

Και επομένως καταλήγουμε στο ότι:

$$d_t^I = \rho(1 - \sqrt{1 - \frac{\beta_t}{1-\rho^2}}) \tag{3.32}$$

$$d_t^{II} = \sqrt{1 - \frac{\beta_t}{1-\rho^2}} \tag{3.33}$$

Ορίζουμε τώρα τους πίνακες: $B_t = \begin{bmatrix} d_t^I & d_t^{II} \\ 1 & 0 \end{bmatrix}, \forall t \in 2, ..., T$ και γράφουμε την προς τα εμπρός διαδικασία ως εξής:

$$\begin{bmatrix} \mathbf{x}_t \\ \mathbf{x}_{t-1} \end{bmatrix} = B_t \begin{bmatrix} \mathbf{x}_{t-1} \\ \mathbf{x}_{t-2} \end{bmatrix} + \begin{bmatrix} \sqrt{\beta_t} \\ 0 \end{bmatrix} \boldsymbol{\epsilon} \tag{3.34}$$

Κινούμενοι αναδρομικά έυκολα μπορούμε να δούμε ότι:

$$\begin{bmatrix} \mathbf{x}_t \\ \mathbf{x}_{t-1} \end{bmatrix} = B_t B_{t-1} ... B_2 \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_0 \end{bmatrix} + \sqrt{(B_t B_{t-1} ... B_3 \begin{bmatrix} \sqrt{\beta_2} \\ 0 \end{bmatrix})^2 + ... + (B_t \begin{bmatrix} \sqrt{\beta_{t-1}} \\ 0 \end{bmatrix})^2 + (\begin{bmatrix} \sqrt{\beta_t} \\ 0 \end{bmatrix})^2 \boldsymbol{\epsilon}} \tag{3.35}$$

όπου η ύψωση στο τετράγωνο και η ρίζα εφαρμόζονται σε κάθε στοιχείο του κάθε διανύσματος. Ορίζοντας λοιπόν τους πίνακες :

$$\tilde{B}_t = B_t * ... * B_2$$

$$\tilde{B}_t^{-k} = B_t * ... * B_{2+k}, \qquad k = 1, ..., t-2$$

και χρησιμοποιώντας ότι $\mathbf{x}_1 = \sqrt{1-\beta_1}\mathbf{x}_0 + \sqrt{\beta_1}\boldsymbol{\epsilon}$ μπορούμε να γράψουμε

$$\begin{aligned} \mathbf{x}_t = &((\tilde{B}_t)_{11}\sqrt{1-\beta_1} + (\tilde{B}_t)_{12})\mathbf{x}_0 \\ &+ \sqrt{((\tilde{B}_t)_{11}\sqrt{\beta_1})^2 + ((\tilde{B}_t^{-1})_{11}\sqrt{\beta_2})^2 + ... + ((\tilde{B}_t^{-t+2})_{11}\sqrt{\beta_{t-1}})^2 + \beta_t}\boldsymbol{\epsilon} \end{aligned}$$

Ορίζουμε λοιπόν τις ποσότητες:

$$a_t = (\tilde{B}_t)_{11}\sqrt{1-\beta_t} + \tilde{B}_t)_{12} \tag{3.36}$$

$$c_t = ((\tilde{B}_t)_{11}\sqrt{\beta_1})^2 + ((\tilde{B}_t^{-1})_{11}\sqrt{\beta_2})^2 + ... + ((\tilde{B}_t^{-t+2})_{11}\sqrt{\beta_{t-1}})^2 + \beta_t \tag{3.37}$$

και καταλήγουμε στο ότι μπορούμε να δειγματίσουμε εύκολα οποιαδήποτε χρονική στιγμή με βάση την εξίσωση:

$$\mathbf{x}_t = a_t\mathbf{x}_0 + \sqrt{c_t}\boldsymbol{\epsilon}_t \tag{3.38}$$

Ορίζουμε στη συνέχεια την αντίστροφη διαδικασία, η οποία ξεκινά από την κατανομή $p(\mathbf{x}_{T-1}, \mathbf{x}_T) = \mathcal{N}\left(\vec{0}, \begin{bmatrix} \mathbf{I}_{n\times n} & \rho\mathbf{I}_{n\times n} \\ \rho\mathbf{I}_{n\times n} & \mathbf{I}_{n\times n} \end{bmatrix}\right)$ και οι μεταβάσεις εξαρτώνται από τα δύο προηγούμενα βήματα, με παραμέτρους που "μαθαίνει" κατά την περίοδο εκπαίδευσης.

$$p_\theta(\mathbf{x}_{0:T}) := p(\mathbf{x}_{T-1}, \mathbf{x}_T) \prod_{t=2}^{T} p_\theta(\mathbf{x}_{t-2}|\mathbf{x}_{t-1}, \mathbf{x}_t) \tag{3.39}$$

$$p_\theta(\mathbf{x}_{t-2}|\mathbf{x}_{t-1}, \mathbf{x}_t) := \mathcal{N}(\mathbf{x}_{t-2}; \boldsymbol{\mu}_\theta(\mathbf{x}_{t-1}, \mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_{t-1}, \mathbf{x}_t, t)) \tag{3.40}$$

Όπως και στο αρχικό μοντέλο αφού η από κοινού κατανομή των $\mathbf{x}_{1:T}$ δεσμευμένης της $\mathbf{x}_0$ είναι κανονική, οι εκ των υστέρων κατανομές της προς τα εμπρός διαδικασίας θα είναι επίσης κανονικές και μπορούμε να υπολογίσουμε της μέση τιμή και τον πίνακα συνδιακύμανσης. Προτού τα ορίσουμε μπορούμε να ξαναγράψουμε την $\mathbf{x}_t$ δεσμευμένη ως προς τις $\mathbf{x}_{t-1}, \mathbf{x}_0$ με τον εξής τρόπο:

$$\mathbf{x}_t = d_t^I \mathbf{x}_{t-1} + d_t^{II} a_{t-2} \mathbf{x}_0 + \sqrt{h_t}\boldsymbol{\epsilon}, \;\; h_t = (d_t^{II})^2 c_{t-2} + \beta_t \tag{3.41}$$

όπου χρησιμοποιήσαμε την εξίσωση (3.31) για τη χρονική στιγμή $t-2$. Τώρα μπορούμε να υπολογίσουμε την εκ των υστέρων κατανομή ως εξής:

$$\begin{aligned}
q(\mathbf{x}_{t-2}|\mathbf{x}_t, \mathbf{x}_{t-1}, \mathbf{x}_0) &= \frac{q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_{t-2}, \mathbf{x}_0) q(\mathbf{x}_{t-2}|\mathbf{x}_{t-1}, \mathbf{x}_0)}{q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_0)} \\
&= \frac{q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_{t-2}) q(\mathbf{x}_{t-1}|\mathbf{x}_{t-2}, \mathbf{x}_0) q(\mathbf{x}_{t-2}|\mathbf{x}_0)}{q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_0) q(\mathbf{x}_{t-1}|\mathbf{x}_0)} \\
&\underset{(3.34)}{\overset{(3.31)}{\propto}} exp\left(-\frac{1}{2} \frac{(\mathbf{x}_t - d_t^I \mathbf{x}_{t-1} - d_t^{II} \mathbf{x}_{t-2})^2}{\beta_t} \right. \\
&- \frac{1}{2} \frac{(\mathbf{x}_{t-1} - d_{t-1}^I \mathbf{x}_{t-2} - d_{t-1}^{II} a_{t-3} \mathbf{x}_0)^2}{h_{t-1}} - \frac{1}{2} \frac{(\mathbf{x}_{t-2} - a_{t-2}\mathbf{x}_0)^2}{c_{t-2}} \\
&\left. + \frac{1}{2} \frac{(\mathbf{x}_t - d_t^I \mathbf{x}_{t-1} - d_t^{II} a_{t-2} \mathbf{x}_0)^2}{h_t} + \frac{1}{2} \frac{(\mathbf{x}_{t-1} - a_{t-1}\mathbf{x}_0)^2}{c_{t-1}} \right) \\
&= exp\left[-\frac{1}{2} \left(\mathbf{x}_{t-2}^2 (\frac{(d_t^{II})^2}{\beta_t} + \frac{(d_{t-1}^I)^2}{h_{t-1}} + \frac{1}{c_{t-2}}) \right. \right. \\
&- 2\mathbf{x}_{t-2} (\frac{d_t^{II}}{\beta_t} \mathbf{x}_t + (\frac{d_{t-1}^I}{h_{t-1}} - \frac{d_t^I d_t^{II}}{\beta_t}) \mathbf{x}_{t-1} \\
&\left. \left. + (\frac{a_{t-2}}{c_{t-2}} - \frac{d_{t-1}^I d_{t-1}^{II} a_{t-3}}{h_{t-1}}) \mathbf{x}_0) + C(\mathbf{x}_t, \mathbf{x}_{t-1}, \mathbf{x}_0) \right) \right]
\end{aligned}$$

Επομένως:

$$\mathbf{x}_{t-2}|\mathbf{x}_t, \mathbf{x}_{t-1}, \mathbf{x}_0 \sim \mathcal{N}(\mathbf{x}_{t-2}; \tilde{\boldsymbol{\mu}}((\mathbf{x}_t, \mathbf{x}_{t-1}, \mathbf{x}_0), \tilde{\beta}_t) \tag{3.42}$$

όπου:

$$\tilde{\beta}_t = \frac{\beta_t h_{t-1} c_{t-2}}{(d_t^{II})^2 c_{t-2} h_{t-1} + (d_{t-1}^I)^2 \beta_t c_{t-2} + \beta_t h_{t-1}} \tag{3.43}$$

$$\tilde{\boldsymbol{\mu}}((\mathbf{x}_t, \mathbf{x}_{t-1}, \mathbf{x}_0) = \tilde{\beta}_t (\frac{d_t^{II}}{\beta_t} \mathbf{x}_t + (\frac{d_{t-1}^I}{h_{t-1}} - \frac{d_t^I d_t^{II}}{\beta_t}) \mathbf{x}_{t-1} + (\frac{a_{t-2}}{c_{t-2}} - \frac{d_{t-1}^I d_{t-1}^{II} a_{t-3}}{h_{t-1}}) \mathbf{x}_0) \tag{3.44}$$

Αντικαθιστώντας το $\mathbf{x}_0$ ως $\frac{1}{a_t}(\mathbf{x}_t - \sqrt{c_t}\boldsymbol{\epsilon}_t)$, παίρνουμε:

$$\begin{aligned}
\tilde{\boldsymbol{\mu}}((\mathbf{x}_t, \mathbf{x}_{t-1}, \boldsymbol{\epsilon}_t) = \tilde{\beta}_t \left((\frac{d_t^{II}}{\beta_t} + \frac{a_{t-2}}{a_t c_{t-2}} - \frac{d_{t-1}^I d_{t-1}^{II} a_{t-3}}{a_t h_{t-1}}) \mathbf{x}_t \right. \\
+ (\frac{d_{t-1}^I}{h_{t-1}} - \frac{d_t^I d_t^{II}}{\beta_t}) \mathbf{x}_{t-1} \\
\left. - \frac{\sqrt{c_t}}{a_t} (\frac{a_{t-2}}{c_{t-2}} - \frac{d_{t-1}^I d_{t-1}^{II} a_{t-3}}{h_{t-1}}) \boldsymbol{\epsilon}_t) \right)
\end{aligned}$$

Στόχος και σε αυτή την περίπτωση είναι η ελαχιστοποίηση της διασταυρούμενης εντροπίας της κατανομής $p_\theta(\mathbf{x}_0)$ ως προς την κατανομή $q(\mathbf{x}_0)$:

$$\begin{aligned}\mathbb{E}_{q(\mathbf{x}_0)}[-log p_\theta(\mathbf{x}_0)] &\overset{(3.8)}{\leq} \mathbb{E}_{q(\mathbf{x}_{0:T})}\big[-log\frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}\big] \\ &= \mathbb{E}_q\big[-log\frac{p(\mathbf{x}_T,\mathbf{x}_{T-1})\prod_{t=2}^T p_\theta(\mathbf{x}_{t-2}|\mathbf{x}_{t-1},\mathbf{x}_t)}{q(\mathbf{x}_1|\mathbf{x}_0)\prod_{t=2}^T q(\mathbf{x}_t|\mathbf{x}_{t-1},\mathbf{x}_{t-2})}\big] \\ &= \mathbb{E}_q\big[-log\frac{p(\mathbf{x}_T,\mathbf{x}_{T-1})}{q(\mathbf{x}_T,\mathbf{x}_{T-1}|\mathbf{x}_0)} - \sum_{t>2} log\frac{p_\theta(\mathbf{x}_{t-2}|\mathbf{x}_{t-1},\mathbf{x}_t)}{q(\mathbf{x}_{t-2}|\mathbf{x}_t,\mathbf{x}_{t-1},\mathbf{x}_0)} - log p_\theta(\mathbf{x}_0|\mathbf{x}_1,\mathbf{x}_2)\big]\end{aligned}$$

Με τη βοήθεια του ορισμού της απόκλισης Kullback-Leibler του ορισμού (2.5) και της εξίσωσης (2.7) μπορούμε να ξαναγράψουμε τον στόχο εκπαίδευσης της (3.12) ως εξής :

$$\mathbb{E}_q\big[\underbrace{D_{\mathrm{KL}}(q(\mathbf{x}_T,\mathbf{x}_{T-1}|\mathbf{x}_0)\|p(\mathbf{x}_T,\mathbf{x}_{T-1}))}_{L_T} + \sum_{t>2}\underbrace{D_{\mathrm{KL}}(q(\mathbf{x}_{t-2}|\mathbf{x}_t,\mathbf{x}_{t-1},\mathbf{x}_0)\|p_\theta(\mathbf{x}_{t-2}|\mathbf{x}_{t-1},\mathbf{x}_t))}_{L_{t-2}} - \underbrace{\log p_\theta(\mathbf{x}_0|\mathbf{x}_1,\mathbf{x}_2)}_{L_0}\big]$$

Όπως βλέπουμε η ανάλυση είναι αρκετά παρόμοια με την περίπτωση του αρχικού μοντέλου. Ακολουθόντας την ίδια συλλογιστική πορεία, πρέπει να καθορίσουμε τις επιλογές για την κατανομή $p_\theta(\mathbf{x}_{t-2}|\mathbf{x}_{t-1},\mathbf{x}_t) := \mathcal{N}(\mathbf{x}_{t-2};\boldsymbol{\mu}_\theta(\mathbf{x}_{t-1},\mathbf{x}_t,t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_{t-1},\mathbf{x}_t,t))$. Ορίζουμε, λοιπόν, τους πίνακες συνδιασποράς ως μη μεταβαλλόμενες σταθερές που εξαρτόνται από το χρόνο και έχουν τη μορφή $\boldsymbol{\Sigma}_\theta(\mathbf{x}_{t-1},\mathbf{x}_t,t) = \sigma_t^2\boldsymbol{I}$, με την επιλογή που θα χρησιμοποιήσουμε στο δικό μας μοντέλο να είναι $\sigma_t^2 = \tilde{\beta}_t$. Με βάση λοιπόν την εξίσωση (3.15) για τον υπολογισμό αποκλίσεων KL μεταξύ κανονικών κατανομών, μπορούμε να γράψουμε το L_{t-2} ως:

$$L_{t-2} = \mathbb{E}_q\left[\frac{1}{2\sigma_t^2}||\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t,\mathbf{x}_{t-1},\mathbf{x}_0) - \boldsymbol{\mu}_\theta(\mathbf{x}_t,\mathbf{x}_{t-1},t)||^2\right] + C \tag{3.45}$$

όπου το C είναι μια σταθερά που δεν εξαρτάται από το θ. Είναι φανερό ότι και σε αυτή την περίπτωση η μέση τιμή της αντίστροφης διαδικασίας θα πρέπει να εκτιμά την εκ των υστέρων μέση τιμή της προς τα εμπρός διαδικασίας και αφού οι $\mathbf{x}_t, \mathbf{x}_{t-1}$ είναι διαθέσιμες κατά την εκτίμηση της $\mathbf{x}_{t-2}$, μπορούμε να παραμετροποιήσουμε την $\boldsymbol{\mu}_\theta$ ως:

$$\begin{aligned}\boldsymbol{\mu}_\theta((\mathbf{x}_t,\mathbf{x}_{t-1},t) &= \tilde{\boldsymbol{\mu}}((\mathbf{x}_t,\mathbf{x}_{t-1},\boldsymbol{\epsilon}_\theta) \\ &= \tilde{\beta}_t\Bigg((\frac{d_t^{II}}{\beta_t} + \frac{a_{t-2}}{a_t c_{t-2}} - \frac{d_{t-1}^I d_{t-1}^{II} a_{t-3}}{a_t h_{t-1}})\mathbf{x}_t \\ &+ (\frac{d_{t-1}^I}{h_{t-1}} - \frac{d_t^I d_t^{II}}{\beta_t})\mathbf{x}_{t-1} \\ &- \frac{\sqrt{c_t}}{a_t}(\frac{a_{t-2}}{c_{t-2}} - \frac{d_{t-1}^I d_{t-1}^{II} a_{t-3}}{h_{t-1}})\boldsymbol{\epsilon}_\theta)\Bigg)\end{aligned}$$

όπου $\boldsymbol{\epsilon}_\theta$ είναι μια συνάρτηση που εκτιμά την $\boldsymbol{\epsilon}_t$ δεδομένης της $\mathbf{x}_t$. Με την παραμετροποίηση αυτή η (3.38) απλοποιείται σε:

$$\mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\left[\frac{\tilde{\beta}_t^2 c_t(\frac{a_{t-2}}{c_{t-2}} - \frac{d_{t-1}^I d_{t-1}^{II} a_{t-3}}{h_{t-1}})^2}{2\sigma_t^2\alpha_t}||\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t)||^2\right] \tag{3.46}$$

Ακολουθώντας το παράδειγμα των Jonathan Ho, Ajay Jain και Pieter Abbeel [2] επιλέγουμε ως κριτήριο εκπαίδευσης την απλοποιημένη αντικειμενική συνάρτηση:

$$\mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\left[||\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t)||^2\right] \tag{3.47}$$

3.4.4 Σύγκριση Αποθορυβοποίησης Έμμεσων Πιθανοθεωρητικών Μοντέλων Διάχυσης (DDIM) και Τροποποιημένης Αποθορυβοποίησης Πιθανοθεωρητικών Μοντέλων Διάχυσης (MDDPM)

Στην παράγραφο αυτή θα αποδείξουμε ότι η Τροποποιημένη Αποθορυβοποίηση Πιθανοθεωρητικών Μοντέλων Διάχυσης (MDDPM) δεν εμπίπτει στην οικογένεια των κατανομών που παρουσιάζεται στη δημοσίευση για την Αποθορυβοποίηση Έμμεσω Πιθανοθεωρητικών Μοντέλων Διάχυσης (DDIM). Θα το αποδείξουμε αυτό δείχνοντας ότι $q(\mathbf{x}_2|\mathbf{x}_0) \neq \mathcal{N}(\sqrt{\bar{\alpha}_2}\mathbf{x}_0, (1-\bar{\alpha}_2)\boldsymbol{I})$. Για το MDDPM έχουμε:

$$\mathbf{x}_2 = d_2^I\mathbf{x}_1 + d_2^{II}\mathbf{x}_0 + \sqrt{\beta_2}\boldsymbol{\epsilon}$$
$$\mathbf{x}_1 = \sqrt{1-\beta_1}\mathbf{x}_0 + \sqrt{\beta_1}\boldsymbol{\epsilon}$$

Επομένως:

$$\mathbf{x}_2 = (d_2^I\sqrt{1-\beta_1} + d_2^{II})\mathbf{x}_0 + \sqrt{\beta_2 + (d_2^I)^2\beta_1}\boldsymbol{\epsilon}$$

Για να είχαμε το ζητούμενο της οικογένειας κατανομών των DDIMs θα έπρεπε:

$$\begin{aligned}
1 &= (d_2^I\sqrt{1-\beta_1} + d_2^{II})^2 + \beta_2 + (d_2^I)^2\beta_1 \\
1 &= (d_2^I)^2(1-\beta_1) + 2d_2^I d_2^{II}\sqrt{1-\beta_1} + (d_2^{II})^2 + (d_2^I)^2\beta_1 + \beta_2 \\
1 &= (d_2^I)^2 + +2d_2^I d_2^{II}\rho + (d_2^{II})^2 + 2d_2^I d_2^{II}(\sqrt{1-\beta_1} - \rho) + \beta_2 \\
(3.32-3.33)1 &= 1 + 2d_2^I d_2^{II}(\sqrt{1-\beta_1} - \rho) + \beta_2 \\
\beta_2 &= -2d_2^I d_2^{II}(\sqrt{1-\beta_1} - \rho)
\end{aligned}$$

Αφού $\beta_2 > 0, d_2^I d_2^{II} > 0$, θα πρέπει $\rho^2 > 1-\beta_1 \implies \beta_1 > 1-\rho^2$, αφού όμως $\beta_2 > \beta_1$ εξ ορισμού, θα έχουμε, $\beta_2 > 1-\rho^2$ και άρα $1 - \frac{\beta_2}{1-\rho^2} < 0$, άρα τα $d_2^I d_2^{II}$ δεν είναι πραγματικοί αριθμοί, και έχουμε καταλήξει σε άτοπο.

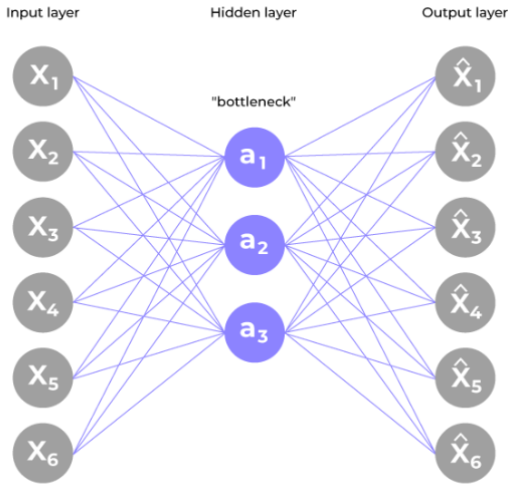
Επομένως έχουμε αποδείξει ότι για κάποιο t $q(\mathbf{x}_t|\mathbf{x}_0) \neq \mathcal{N}(\sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\boldsymbol{I})$ και άρα τα MDDPM δεν μπορούν να υπάγονται στην οικογένεια κατανομών $\mathcal{Q}$ που ορίζεται στις (3.22) - (3.24).

3.5 Σύντομη ανασκόπηση άλλων γνωστών παραγωγικών μοντέλων

Στην ενότητα αυτή παρουσιάζεται μια σύντομη ανασκόπηση ορισμένων ευρέως διαδεδομένων παραγωγικών μοντέλων μηχανικής μάθησης, τα οποία έχουν διαμορφώσει την εξέλιξη του πεδίου. Στο τελευταίο μέρος της ενότητας, αναδεικνύονται ομοιότητες και θεωρητικές συνδέσεις με το μοντέλο DDPM, το οποίο αποτελεί το βασικό αντικείμενο μελέτης της παρούσας εργασίας.

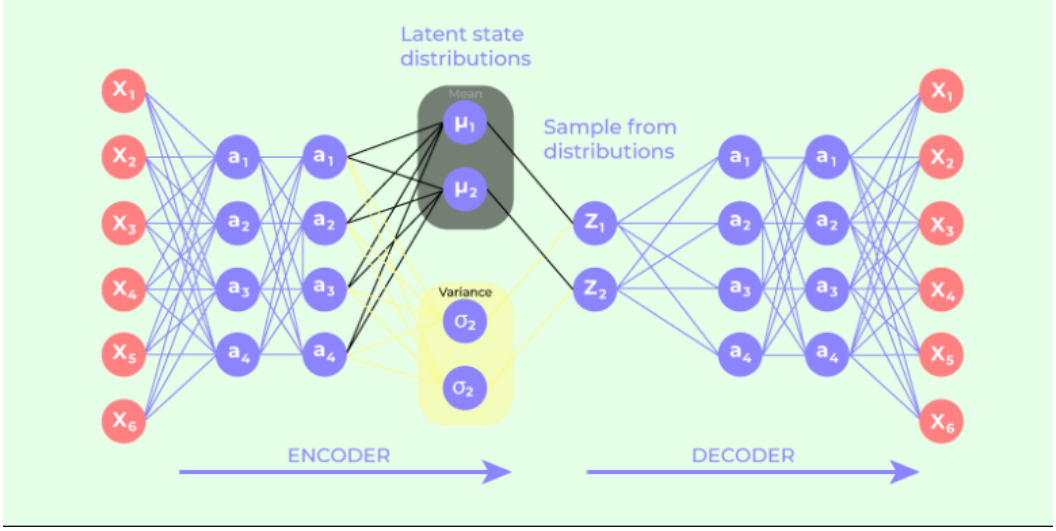
3.5.1 Αυτοκωδικοποιητές Βασισμένοι σε Μεταβολική Συμπερασματολογία (Variational Autoencoders)

Οι Varaiational Autoencoders (Αυτοκωδικοποιητές Βασισμένοι σε Μεταβολική Συμπερασματολογία) αποτελούν μια κατηγορία παραγωγικών μοντέλων που χρησιμοποιείται για παραγωγή δειγμάτων που έχουν κοινά χαρακτηριστικά (ή αλλιώς προέρχονται από την ίδια κατανομή) με τα δείγματα στο σύνολο δεδομένων εκπαίδευσης. [18] Πρωτού όμως παρουσιάσουμε τη λειτουργία των VAEs είναι απαραίτητο να ορίσουμε αρχικά του αυτοκωδικοποιητές (autoencoders).



Σχήμα 3.5: Αυτοκωδικοποιητής. Πηγή : https://www.geeksforgeeks.org/auto-encoders/

Οι αυτοκωδικοποιητές είναι ένα είδος νευρωνικού δικτύου (βλ. Ενότητα 4) το οποίο εκπαιδεύεται για να αναπαράγει την είσοδο του και απαρτίζεται από δύο βασικά συστατικά. Τον κωδικοποιητή, ο οποίος συμπιέζει την είσοδο σε μια αναπαράσταση χαμηλότερης διάστασης, την οποία συχνά αναφέρουμε ως "λανθάνων χώρο" και τον απωδικοποιητή, ο οποίος ανακατασκευάζει την αρχική είσοδο από τη συμπιεσμένη της μορφή. Στόχος εκπαίδευσης των αυτοκωδικοποιητών είναι η ελαχιστοποίηση του μέσοτ τετραγωνικού σφάλματος μεταξύ εισόδου και εξόδου. [19] Επιπλέον, οι αυτοκωδικοποιητές μπορούν να προσαρμοστούν για χρήση σε προβλήματα όπως η συμπλήρωση εικόνας, η επαναφορά κατεστραμμένων περιοχών, ή η αφαίρεση ανεπιθύμητων στοιχείων από εικόνες.



Σχήμα 3.6: Αυτοκωδικοποιητές Βασισμένοι σε Μεταβολική Συμπερασματολογία (Variational Autoencoders). Πηγή : https://www.geeksforgeeks.org/variational-autoencoders/

Σε αντίθεση με τους κλασικούς αυτοκωδικοποιητές, τα VAEs , επιδιώκουν να προσεγγίσουν την κατανομή των δεδομένων στον λανθάνοντα χώρο μέσω πιθανοκρατικής προσέγγισης. Έτσι δειγματίζοντας από αυτό το χώρο και χρησιμοποιώντας τον ήδη εκπαιδευμένο απωδικοποιητή μπορούν να παράγουν δείγματα παρόμοια με αυτά που υπάρχουν στο αρχικό σύνολο δεδομένων. Η κατανομή που χρησιμοποιείται κυρίως για να προσεγγίσει την κατανομή του "λανθάνων χώρου" είναι η κανονική κατανομή με άγνωστη μέση τιμή και διασπορά, τις οποίες το μοντέλο μαθαίνει μέσα από την εκπαίδευση. [20]Στα μοντέλα αυτά στόχος εκπαίδευσης είναι η ελαχιστοποίηση του μέσου τετραγωνικού σφάλματος μεταξύ της εισόδου και της εξόδου που προκύπτει από την επανακατασκευή αυτής και της απόκλισης KL μεταξύ της κατανομής που μαθαίνει το μοντέλο για τον "λανθάνων χώρο" και μιας εκ των προτέρων (prior) κανονικής κατανομής — η οποία λειτουργεί ως ρυθμιστικός παράγοντας για τη γενίκευση του μοντέλου.

$$L = \mathbb{E}_{q(z|x)}[ReconstructionLoss] + KL(q(z|x)||p(z)) \tag{3.48}$$

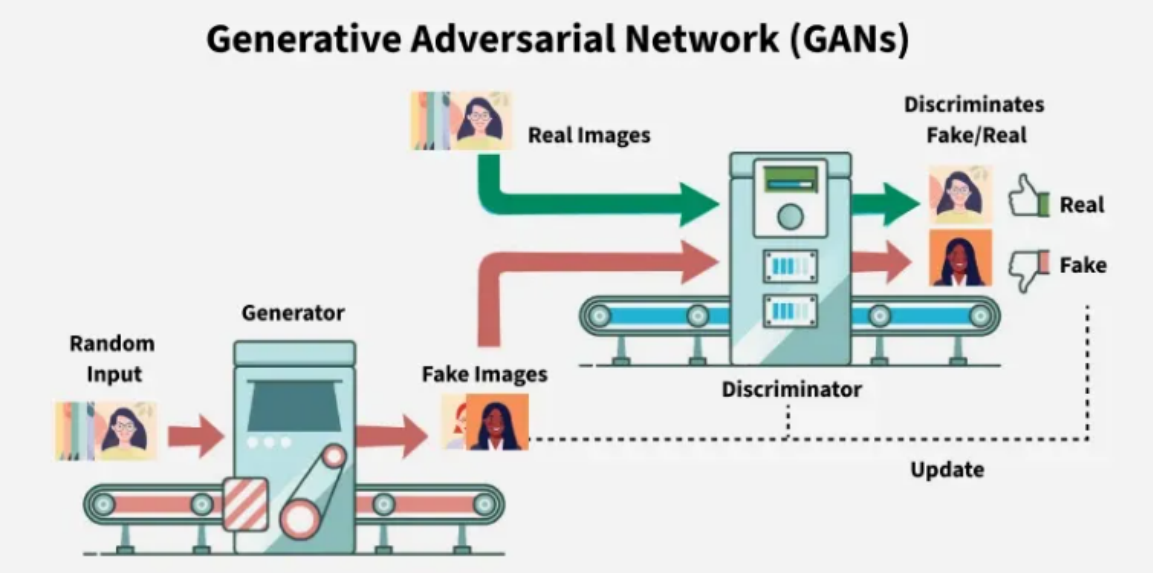
3.5.2 Παραγωγικά Ανταγωνιστικά Δίκτυα (Generative Adversarial Networks)

Τα Παραγωγικά Αναγωνιστικά Δίκτυα (Generative Adversarial Networks), GANs για συντομία είναι μια κλάση παραγωγικών μοντέλων που έχουν ως στόχο να μάθουν αυτόνομα μοτίβα στα δεδομένα του συνόλου εκπαίδευσης έτσι ώστε να παράγουν καινούρια δείγματα που έχουν κοινά χαρακτηριστικά με τα δεδομένα εκπαίδευσης.

Η αρχιτεκτονική των GANs βασίζεται στην αλληλεπίδραση δύο νευρωνικών δικτύων τα οποία τίθενται σε ένα ανταγωνιστικό παιχνίδι:

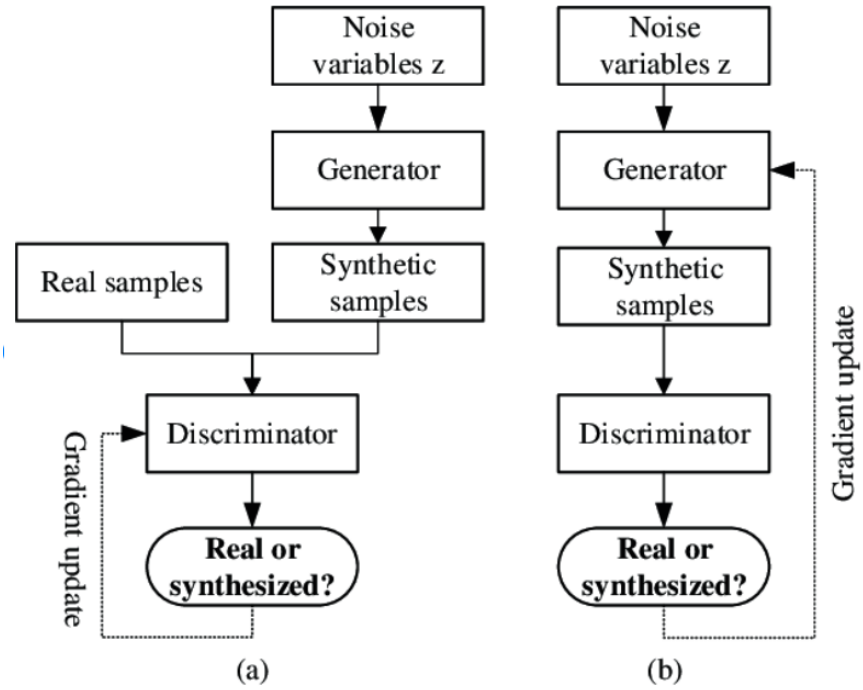
- Γεννήτορας (Generator): Δημιουργεί συνθετικά δεδομένα από τυχαίο θόρυβο στοχεύοντας να ξεγελάσει τον διευκρινιστή και να τον πείσει ότι αποτελούν πραγματικά δεδομένα,
- Διευκρινιστής (Discriminator): Λειτουργεί ως κριτής και καλείται να διακρίνει ανάμεσα σε πλαστά/συνθετικά και πραγματικά δεδομένα.

Τα δύο αυτά νευρωνικά δίκτυα εμπλέκονται σε ένα συνεχές ανταγωνιστικό παιχνίδι, όπου ο γεννήτορας εκπαιδεύεται ώστε να παράγει όσο το δύνατο πιο ρεαλιστικά δεδομένα ενώ ο διευκρινιστής εκπαιδεύεται στο να μπορεί να διακρίνει ακόμα και πολύ ρεαλιστικά συνθετικά δεδομένα από τα πραγματικά. Η ιδέα για τα GANs εμφανίστηκε, για πρώτη φορά σε μια δημοσίευση [21] το 2014.



Σχήμα 3.7: Αρχιτεκτονική GAN Πηγή : https://www.geeksforgeeks.org/generative-adversarial-network-gan/

Ο Γεννήτορας είναι ένα βαθύ νευρωνικό δίκτυο που λαμβάνει τυχαίο θόρυβο ως είσοδο και προσπαθεί να παράξει ρεαλιστικά συνθετικά δεδομένα. Εκπαιδεύεται λοιπόν, ώστε να μάθει την κατανομή στην οποία ανήκουν τα δείγματα του συνόλου δεδομένων εκπαίδευσης. Ο Διευκρινιστής είναι ένας δυαδικός ταξινομητής, που κατατάσσει κάθε είσοδο είτε στην κλάση των πραγατικών είτε στην κλάση των συνθετικών δεδομένων. Κατά την εκπαίδευση όταν ο Γεννήτορας καταφέρνει να παραπλανήσει τον διευκρινιστή με τα δείγματα που παράγει, τότε επιβραβέβεται και οι παραμέτροι του δεν επηρεάζονται, ενώ ο διευκρινιστής λαμβάνει ενημέρωση στις παραμέτρους του. Αντίθετα στην περίπτωση που ο διευκρινιστής ταξινομεί σωστά την είσοδο του, τότε ενημέρωση γίνεται μόνο στις παραμέτρους του γεννήτορα.



Σχήμα 3.8: Εκπαίδευση ενός GAN Πηγή : Wang, Q., Zhou, X., Wang, C., Liu, Z., Huang, J., Zhou, Y., ... & Cheng, J. Z. (2019). WGAN-based synthetic minority over-sampling technique: Improving semantic fine-grained classification for lung nodules in CT images. IEEE Access, 7, 18450-18463./

Το "παιχνίδι" αυτό, μπορούμε να πούμε ότι κατά κάποιο τρόπο αντιστοιχεί σε ένα παιγνίο δύο παικτών μηδενικού αθροίσματος από την θεωρία παιγνίων στο οποιό οι δύο παίκτες είναι ο Γεννήτορας και ο Διευκρινιστής.

3.5.3 Energy Based Models

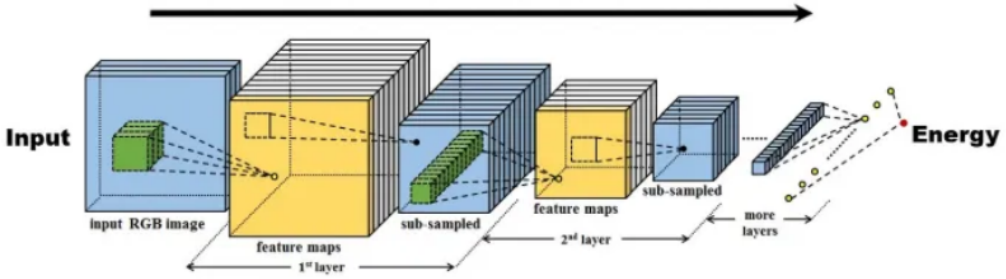
Τα Ενεργειακά Μοντέλα (Energy Based Models, EBMs για συντομία) είναι παραγωγικά μοντέλα μηχανικής μάθησης που έχουν ως κύριο σκοπό την προσέγγιση της κατανομής του συνόλου δεδομένων εκπαίδευσης μέσω μια συνάρτησης ενέργειας. Στα EBMs λοιπόν, προσπαθούμε να βρούμε τη συνάρτηση ενέργειας (energy function) $E(\mathbf{x})$, η οποία θα μας βοηθήσει να μοντελοποιήσουμε την κατανομή των δειγμάτων. Χαμηλότερες τιμές ενέργειας αντιστοιχούν σε υψηλότερες πιθανότητες και η περιθώρια κατανομή του κάθε δείγματος δίνεται ως:

$$\mathbb{P}(\mathbf{x}) = \frac{e^{-E(\mathbf{x})}}{Z} \tag{3.49}$$

όπου Z μια σταθερά κανονικοποίησης, δηλαδή:

- Για συνεχείς τιμές: $Z = \int_{\mathbf{x}} e^{-E(\mathbf{x})} d\mathbf{x}$
- Για διακριτές τιμές : $Z = \sum_{\mathbf{x}} e^{-E(\mathbf{x})}$

Η αντιστοίχηση μεταξύ δεδομένων εισόδου και τιμών ενέργειας γίνεται με τη χρήση ενός συνελικτικού νευρωνικού δικτύου, όπως φαίνεται και στο σχήμα 3.7.



Σχήμα 3.9: Συνελικτικό Νευρωνικό Δίκτυο με σκοπο την αντιστοίχηση δειγμάτων - τιμών ενέργειας στα Energy Based Models Πηγή : https://medium.com/@mathparracho/energy-based-models-and-jems-generative-intuition-and-practice-8a69f1779188

Η εκπαίδευση των EBMs βασίζεται στην ελαχιστοποίηση της ενέργειας που αντιστοιχεί σε πραγματικά δεδομένα και την αύξηση της ενέργειας δειγμάτων που δεν ανήκουν στην κατανομή στόχο. Ωστόσο, λόγω της συνάρτησης κανονικοποίησης αυτό δεν είναι εύκολο καθώς αυτή εξαρτάται από όλες τις δυνατές τιμές της συνάρτησης ενέργειας και δεν μπορεί να υπολογιστεί αναλυτικά. Επομένως δεν μπορούμε να εφαρμόσουμε την μέθοδο μέγιστης πιθανοφάνειας εξ αρχής, χωρίς τη χρήση κάποιας προσέγγισης. [22]

Για να ξεπεραστεί η δυσκολία υπολογισμού του Z, έχουν προταθεί διάφορες εναλλακτικές μέθοδοι εκπαίδευσης:

- Contrastive Divergence (CD):
  Προτάθηκε από τον Hinton για την εκπαίδυση των Boltzmann Machines.[23] Στην μέθοδο αυτή γίνεται προσέγγιση της απόρριψης δείγματος με λίγες επαναλήψεις Gibbs sampling, με σκοπό να μειωθεί η υπολογιστική πολυπλοκότητα. Η μέθοδος αυτή, ωστόσο, δεν συγκλίνει απαραίτητα σε μέγιστη πιθανοφάνεια.

- Score Matching:
  Χρησιμοποιείται για την παράκαμψη της ανάγκης κανονικοποίησης, ελαχιστοποιώντας την απόκλιση μεταξύ της βαθμίδας του λογαρίθμου της κατανομής και της βαθμίδας των δεδομένων. Εφαρμόζεται κυρίως σε συνεχείς κατανομές [24].

- Noise-Contrastive Estimation (NCE):
  Αντιμετωπίζει το πρόβλημα, ως ένα πρόβλημα δυαδικής ταξινόμησης μεταξύ πραγματικών δειγμάτων και θορύβου. Μαθαίνει λοιπόν την κατανομή μέσω του διαχωρισμού των δειγμάτων και του θορύβου, χωρίς την ανάγκη υπολογισμού του Z [25].

Τα EBMs χρησιμοποιούνται ευρέως για παραγωγή εικόνων, σε προβλήματα ενισχυτικής μάθησης, για αναγνώριση ανόμαλων σημείων (anomaly detection)

και επίσης ως διευκρινιστές σε Παραγωγικά Ανταγωνιστικά Δίκτυα (GANs) με ενεργειακή ερμηνεία.

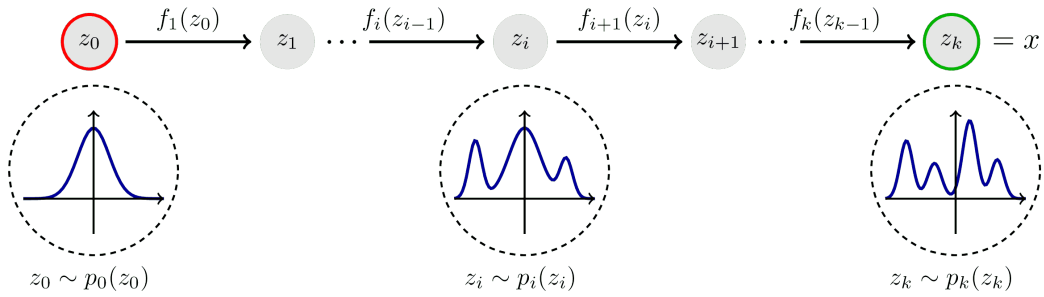
3.5.4 Normalizing Flows

Τα Normalizing Flows είναι ακόμα μια κλάση παραγωγικών μοντέλων μηχανικής μάθησης που βασίζεται στην πιθανοφάνεια όπως και τα VAEs. Στόχος των NF είναι να αντιστοιχίσουν απλές κατανομές (εύκολες στην δειγματοληψία και τον υπολογισμό των πυκνοτήτων) σε πιο σύνθετες κατανομές (τις οποίες μαθαίνουμε μέσα από τα δεδομένα) , μέσω μιας σειράς αντιστρέψιμων και διαφορίσιμων μετασχηματισμών [26].

Η βασική μαθηματική ιδέα στηρίζεται στον τύπο αλλαγής μεταβλητής για πυκνότητες πιθανότητας.

$$p_X(\mathbf{x}) = p_Z(f^{-1}(\mathbf{x})|det(\frac{\partial f^{-1}(\mathbf{x})}{\partial \mathbf{x}})| \tag{3.50}$$

όπου τα $\mathbf{x}$ και $\mathbf{z} = f^{-1}(\mathbf{x})$ έχουν την ίδια διάσταση,ο $(\frac{\partial f^{-1}(\mathbf{x})}{\partial \mathbf{x}})$ είναι ο ιακωβιανός πίνακας διάστασης $n \times n$.



Σχήμα 3.10: Normalizing Flows : Μετασχηματισμοί από απλές σε πιο σύνθετες κατανομές Πηγή: https://tikz.net/normalizing-flow/

Ορίζουμε τώρα τα μοντέλα NF . Έστω ένα μοντέλο λανθάνουσων μεταβλητών με παρατηρούμενες μεταβλητές X και λανθάνουσες μεταβλητές Z. Η αντιστοιχία μεταξύ Z και X δίνεται από τη ντετερμινιστική και αντιστρέψιμη συνάρτηση $f_0 : \mathbb{R}^n \to \mathbb{R}^n$, τ.ω. $X = f_0(Z)$ και $Z = f_0^{-1}(X)$. Σε αντίθεση με τα VAEs, τα NF απαιτούν τα X και Z να έχουν την ίδια διάσταση, όπως αναφέραμε και παραπάνω, και ο μετασχηματισμός να είναι αντιστρέψιμος. Για ένα δείγμα η λογαριθμοποιημένη πιθανοφάνεια δίνεται από τον τύπο.

$$logp_\theta(\mathbf{x}) = logp_\theta(\mathbf{z}) + log|det(\frac{\partial f^{-1}(\mathbf{x})}{\partial \mathbf{x}})| \tag{3.51}$$

Η εκπαίδευση των NFs γίνεται με μέγιστη πιθανοφάνεια (maximum likelihood estimation), με στόχο την εύρεση των παραμέτρων του μετασχηματισμού που μεγιστοποιούν την πιθανότητα των δεδομένων. Ένα πλεονέκτημα των NFs σε σχέση με τα GANs και τα VAEs είναι ότι η λογαριθμοποιημένη πιθανοφάνεια

μπορεί να υπολογιστεί αναλυτικά, χωρίς τη χρήση παραγώγων ή δειγματοληψίας, γεγονός που καθιστά την εκπαίδευση πιο σταθερή και θεωρητικά θεμελιωμένη.

Η κατανομή στον λανθάνοντα χώρο p_Z θεωρείται προκαθορισμένη (συνήθως $\mathcal{N}(0, \mathbf{I})$), και το ζητούμενο είναι να μάθουμε τις παραμέτρους του αντιστρέψιμου μετασχηματισμού f^{-1}, έτσι ώστε τα δεδομένα x να αντιστοιχούνται σε z με όσο το δυνατόν υψηλότερη πυκνότητα. Στην πράξη, οι μετασχηματισμοί f_i που απαρτίζουν τη ροή υλοποιούνται μέσω παραμετρικών συναρτήσεων – δηλαδή νευρωνικών δικτύων.

3.5.5 Σύγκριση Παραγωγικών Μοντέλων

Αν και τα μοντέλα DDPM διαφέρουν σημαντικά σε δομή και θεωρητική προσέγγιση από άλλες κατηγορίες παραγωγικών μοντέλων, παρουσιάζουν κάποιες ενδιαφέρουσες ομοιότητες με κάθε μια από τις κλάσεις μοντέλων που αναφέραμε παραπάνω.

Σχετικά με την σύγκριση με τα VAEs,τα DDPMs εκπαιδεύονται χρησιμοποιώντας το Evidence Lower Bound - ELBO (Εξίσωση 2.9) ενώ επίσης η αντίστροφη διαδικασία διάχυσης μοιάζει με αποκωδικοποίηση στον λανθάνοντα χώρο, όπου ο ντετερμινιστικός θόρυβος παίζει ρόλο παρόμοιο με εκείνον του λανθάνοντα χώρου στα VAEs.

Ενώ εκ πρώτης όψεως τα Μοντέλα Διάχυσης γενικότερα, δεν δείχνουν να έχουν σημαντικές ομοιότητες με τα GANs, ορισμένες πρόσφατες επεκτάσεις των DDPMs συνδυάζουν την αρχιτεκτονική των GANs με την αντίστροφη διάχυση για να βελτιώσουν την ταχύτητα και την ποιότητα των παραγώμενων δειγμάτων. Επίσης οι δύο κλάσεις μοντέλων, όπως και τα VAEs, εκπαιδεύονται χωρίς των αναλυτικό υπολογισμό των πιθανοφανειών, βασιζόμενες σε εναλλακτικές συναρτήσεις απώλειας.

Τα EBMs ίσως είναι η κλάση μοντέλων η οποία είναι πιο κοντά, από θεωρητική άποψη, στα Μοντέλα Διάχυσης και τα DDPMs καθώς η αντίστροφη διαδικασία διάχυσης μπορεί να ερμηνευτεί ως παράγωγος μιας ενεργειακής συνάρτησης που σταδιακά προσεγγίζει την κατανομή των δεδομένων. Ειδικότερα τα Score-based μοντέλα, στα οποία βασίζονται πολλές σύγχρονες εκδοχές των Μοντέλων Διάχυσης συνδέονται άμεσα με την εκπαίδευση ενεργειακών μοντέλων που συζητήσαμε στην παράγραφο 3.5.3.

Συγκεκριμένα, η απλοποιημένη αντικειμενική συνάρτηση που αναφέρεται στα DDPM ($L_{simple} = \mathbb{E}_{t,x_0,\epsilon}[||\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}, t)||^2]$), αντιστοιχεί σε μια αντικειμενική μορφή του τύπου "Score Matching" της μορφής [27]:

$$L := \mathbb{E}_{t\sim U(\{1,\dots,n\})}\mathbb{E}\left[\|\mathbf{s}_\theta(t, \mathbf{x}_t) - \nabla \log q_t(\mathbf{x}_t|\mathbf{x}_0)\|^2\right], \tag{3.52}$$

όπου $\mathbf{s}_\theta = \frac{-1}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta$.

Παράρτημα : Απόδειξη του παραπάνω ισχυρισμού

Μπορούμε να γράψουμε αρχικά την $\mathbf{x}_t$ ως $\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$. Επομένως:

$$q(\mathbf{x}_t|\mathbf{x}_0) = \frac{1}{\sqrt{2\pi(1-\bar{\alpha}_t)}} e^{-\frac{(\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{x}_0)^2}{2(1-\bar{\alpha}_t)}}$$

$$\begin{aligned}
\nabla \log q(\mathbf{x}_t|\mathbf{x}_0) &= \frac{\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{x}_0}{\sqrt{1-\bar{\alpha}_t}} \\
&\overset{(3.3)}{=} \frac{\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\frac{1}{\sqrt{\bar{\alpha}_t}}(\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_t)}{1-\bar{\alpha}_t} \\
&= \frac{1}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_t
\end{aligned}$$

Επομένως

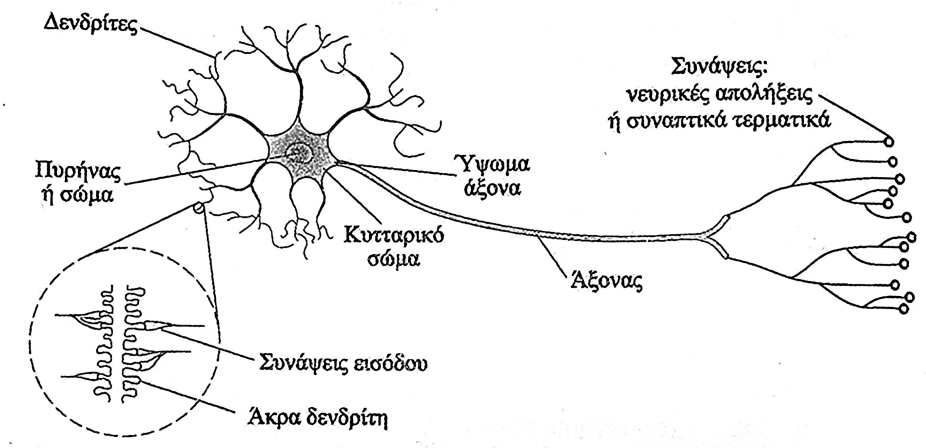
$$\begin{aligned}
L_{simple} &= \mathbb{E}_{t,x_0,\epsilon}[||\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}, t)||^2] \\
&= \mathbb{E}_{t\sim U(\{1,\dots,n\})}\mathbb{E}[||\sqrt{1-\bar{\alpha}_t}(\mathbf{s}_\theta(t,\mathbf{x}_t) - \nabla \log q_t(\mathbf{x}_t|\mathbf{x}_0))||^2]
\end{aligned}$$

Και άρα έχουμε αντιστοιχεία των δύο αντικειμενικών συναρτήσεων με μια διαφορά ενός όρου κανονικοποίησης με βάση τις παραμέτρους $\bar{\alpha}_t$.

Τέλος, η βασική ομοιότητα μεταξύ Normalizing Flows (NFs) και Μοντέλων Διάχυσης έγκειται στην προσπάθεια μετασχηματισμού μιας απλής, γνωστής κατανομής (όπως η κανονική) σε μια πολύπλοκη κατανομή που περιγράφει τα δεδομένα. Αν και τα NFs χρησιμοποιούν αντιστρέψιμους, παραμετρικούς μετασχηματισμούς, ενώ τα DDPMs βασίζονται σε στοχαστική εξελικτική διαδικασία, και τα δύο μοντέλα επιχειρούν να μάθουν τον ίδιο τύπο αντιστοίχισης.

4 Νευρωνικά Δίκτυα

Η έρευνα και η ανάπτυξη μοντέλων τεχνητών νευρωνικών δικτύων είναι εμπνευσμένη από τη δομή και τη λειτουργία του εγκεφάλου, του οποίου βασικό στοιχείο είναι οι νευρώνες. Επίσης η λειτουργία και ο τρόπος που εκτελεί υπολογισμούς ο ανθρώπινος εγκέφαλος είναι εντελώς διαφορετικά σε σχέση με τον συμβατικό ψηφιακό υπολογιστή. Οι νευρώνες είναι νευρικά κύτταρα τα οποία δημιουργούν ένα πυκνό δίκτυο επικοινωνίας μεταξύ τους.



Σχήμα 4.1: Ο νευρώνας του ανθρώπινου εγκεφάλου. Πηγή : Διαμαντάρας, Κ., & Μπότσης, Δ. (2019). Μηχανική μάθηση. Αθήνα: Κλειδάριθμος.

Τα τυπικά τεχνητά νευρωνικά δίκτυα όμως, χρησιμοποιούν πολύ πιο απλοποιημένα μοντέλα νευρώνων σε σχέση με αυτά της νευρολογίας ώστε να διατηρούνται μόνο τα πολύ αδρά χαρακτηριστικά αυτών [28]. Τα δύο βασικά χαρακτηριστικά που πρέπει να πληρούν τα μοντέλα νευρώνων ώστε να μπορούν να δημιουργήσουν ιδιαιτέρως ενδιαφέροντα δίκτυα είναι [29]:

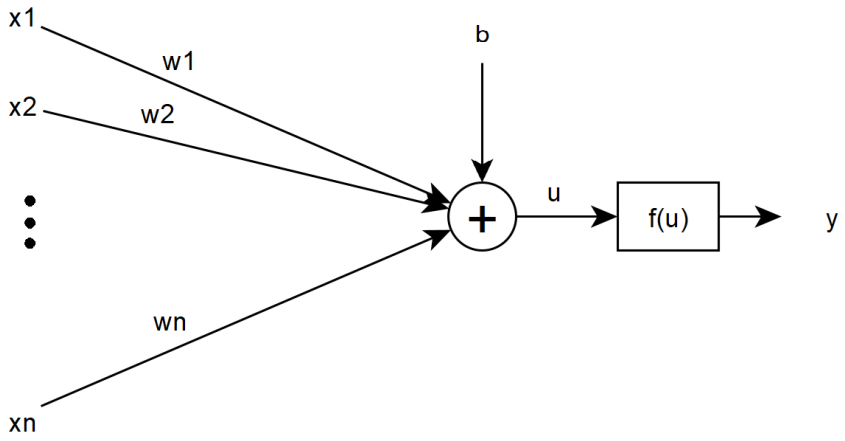
- Πλαστικότητα : Η έννοια αυτή είναι συνώνυμη του όρου "εξελισσόμενο" όταν συζητάμε για το ανθρώπινο νευρικό σύστημα. Οι νευρώνες πρέπει να έχουν ρυθμιζόμενες παραμέτρους ώστε να διευκολύνεται η διαδικασία μάθησης και να υπάρχει η δυνατότητα το σύστημα να προσαρμόζεται ανάλογα με το περιβάλλον του.

- Το δίκτυο πρέπει να αποτελείται από μεγάλο πλήθος νευρώνων ώστε να επιτυγχάνεται παραλληλισμός της επεξεργασίας και κατανομή των πληροφοριών .

Με την ευρεία ανάπτυξη του τομέα της μηχανικής μάθησης και της τεχνητής νοημοσύνης έχει πλέον αναπτυχθεί πληθώρα μοντέλων τεχνητών νευρωνικών δικτύων, καθένα από τα οποία χρησιμοποιείται για διαφορετικό σκοπό. Γενικότερα όμως μπορούμε να ορίσουμε τα τεχνητά νευρωνικά δίκτυα ως συστήματα πολλών υπολογιστικών κόμβων (νευρώνων) οι οποίοι συνδέονται μεταξύ τους. Μια τυπική δομή ενός τεχνητού νευρωνικού δικτύου αποτελείται από το στρώμα εισόδου, τα κρυμμένα στρώματα και το στρώμα εξόδου. Η ύπαρξη

μεγάλου πλήθους κρυμμένων στρωμάτων είναι αυτό που συχνά αναφέρεται ως βαθειά μάθηση (deep learning) [30].

Το πιο απλό μοντέλο είναι ένα δίκτυο που αποτελείται από ένα μόνο νευρώνα και οι μόνες συνδέσεις που υπάρχουν είναι μεταξύ των εισόδων $x_1, ..., x_n$ και του νευρώνα.



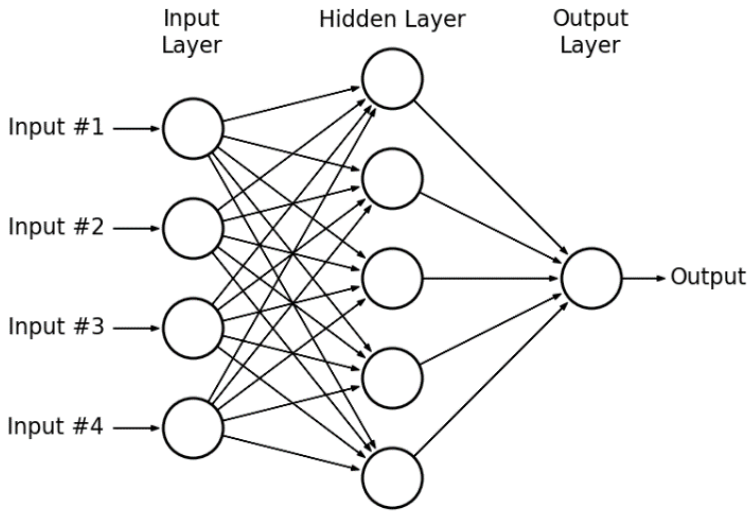
Σχήμα 4.2: Απλό Perceptron : Βασίζεται στο μοντέλο των McCulloch και Pits για τον νευρώνα

Ένα παράδειγμα ενός τέτοιου μοντέλου είναι το μοντέλο Perceptron που φαίνεται και στο σχήμα (4.2), στο οποίο η έξοδος υπολογίζεται ως εξής :

$$y = f\left(\sum_{i=1}^{n} w_i x_i + b\right) \tag{4.1}$$

όπου w_i τα συναπτικά βάρη, b η πόλωση (bias term) και f η λεγόμενη συνάρτηση ενεργοποίησης. Για την συνάρτηση ενεργοποίησης στο απλό Perceptron επιλέγεται είτε η βηματική 0/1 είτε η βηματική -1/1 ενώ σε πιο περίπλοκα μοντέλα χρησιμοποιούνται και άλλες συναρτήσεις όπως η σιγμοειδής, η γραμμική και η συνάρτηση ράμπας ή ανορθώμενη γραμμική μονάδα (Rectified Linear Unit - ReLU) όπως είναι επίσης γνωστή.

Ένα άλλο είδος μοντέλου είναι τα δίκτυα Perceptron πολλών στρωμάτων (MLP) (σχήμα 4.3) τα οποία χρησιμοποιούν πολλαπλά στρώματα νευρώνων για να υλοποιήσουν συναρτήσεις που δεν είναι εφικτό να υλοποιηθούν με ένα απλό δίκτυο Perceptron. Τα MLP χρησιμοποιούνται κυρίως σε προβλήματα ταξινόμησης ή παλινδρόμησης.



Σχήμα 4.3: Παράδειγμα δικτύου Perceptron πολλαπλών στρωμάτων (MLP). Πηγή : Hassan, H., Negm, A., Zahran, M., & Saavedra, O. (2015). Assessment of artificial neural network for bathymetry estimation using high-resolution satellite imagery in shallow lakes: Case study El Burullus Lake. International Water Technology Journal

Στην περίπτωση δεδομένων εικόνας ή γενικότερα πολυδιάστατων οντοτήτων, η χρήση δικτύων MLP δεν είναι ευνοϊκή καθώς αυξάνονται ραγδαία οι υπολογιστικές απαιτήσεις για την εκπαίδευση τους, λόγω του μεγάλου αριθμού βαρών που θα προϋπόθετε ένας τέτοιος σχεδιασμός. Επιπλέον, ένα σημαντικό μειονέκτημα αυτών των δικτύων είναι ότι δεν μπορούν να αναγνωρίσουν μοτίβα που εμφανίζονται σε διαφορετικές θέσεις μιας εικόνας (δεν παρουσιάζουν δηλαδή μεταφορική αμεταβλητότητα), κάτι που είναι απαραίτητο σε προβλήματα αναγνώρισης προτύπων, ταξινόμησης εικόνων κ.α. Για τον λόγο αυτό αναπτύχθηκαν τα Συνελικτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks) που αντιμετωπίζουν αυτά τα προβλήματα και τα οποία θα μελετήσουμε εκτενώς στην παράγραφο 4.1 .

4.1 Συνελικτικά Νευρωνικά Δίκτυα

Τα Συνελικτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks - CNNs) είναι μια κλάση μοντέλων τεχνητών νευρωνικών δικτύων σχεδιασμένη για να χειρίζεται πολυδιάστατα δεδομένα και κυρίως δεδομένα εικόνας. Προσομειώνουν τον οπτικό φλοιό του εγκεφάλου και βασίζονται στο μοντέλο neocognitron [31], το οποίο αποτέλεσε τον πρόδρομο των σύγχρονων CNN και σχεδιάστηκε με σκοπό την ανίχνευση μοτίβων σε εικόνες. Πρωτού, όμως, προχωρήσουμε στην ανάλυση των CNN, πρέπει να δώσουμε κάποιους ορισμούς.

4.1.1 Συνελικτικό Επίπεδο

Ορισμός 4.1. *(Συνέλιξη)*
Η Συνέλιξη (convolution) στα πλαίσια της μηχανικής μάθησης και της βαθειάς μάθησης ορίζεται ως εξής:
Έστω ένα διάνυσμα εισόδου $\boldsymbol{x}$ μήκους N και το διάνυσμα βαρών (ονομάζεται

επίσης φίλτρο ή πυρήνας) $\boldsymbol{w}$ *μήκους* L*, τότε η συνέλιξη* $[\boldsymbol{w} \circledast \boldsymbol{x}]$ *είναι ένα διάνυσμα μήκους* $N - L + 1$*, του οποίου η* i*-οστή συντεταγμένη δίνεται από τον τύπο:*

$$[\boldsymbol{w} \circledast \boldsymbol{x}](i) = \sum_{u=0}^{L-1} w_u x_{i+u} \tag{4.2}$$

Η συνέλιξη στις δύο διαστάσεις για είσοδο $\boldsymbol{X}$ *με διαστάσεις* (x_h, x_w) *και φίλτρο* $\boldsymbol{W}$ *με διαστάσεις* (f_h, f_w) *είναι ένας πίνακας με διαστάσεις* $(x_h - f_h + 1, x_w - f_w + 1)$*, του οποίου το* (i, j) *στοιχείο δίνεται ως εξής:*

$$[\boldsymbol{W} \circledast \boldsymbol{X}](i, j) = \sum_{u=0}^{f_h-1} \sum_{v=0}^{f_w-1} w_{u,v} x_{i+u,j+v} \tag{4.3}$$

Η προκύπτουσα έξοδος της συνέλιξης $\boldsymbol{Y} = \boldsymbol{W} \circledast \boldsymbol{X}$ *ονομάζεται χάρτης χαρακτηριστικών (feature map).*

Επειδή η χρήση της συνέλιξης μειώνει σημαντικά τη διαστατικότητα του χάρτη χαρακτηριστικών, χρησιμοποιείται συχνά η μέθοδος του συμπληρώματος με μηδενικά (zero padding) ώστε να επιβραδύνει ή και να αποτρέψει την μείωση της διαστατικότητας μετά από κάθε συνελικτικό επίπεδο. Σε αυτή την μέθοδο προσθέτουμε μηδενικά περιμετρικά της εισόδου. Αν το φίλτρο έχει μέγεθος (f_h, f_w), η είσοδος (x_h, x_w) και προσθέτουμε μηδενικά μεγέθους p_h, p_w τότε η έξοδος θα έχει μέγεθος: $(x_h - f_h + 2p_h + 1, x_w - f_w + 2p_w + 1)$

Ορισμός 4.2. *(Ιδια Συνέλιξη)*
Αφορά την περίπτωση που επιθυμούμε η έξοδος να έχει τις ίδιες διαστάσεις με την είσοδο. Χρησιμοποιούμε λοιπόν την τεχνική του zero padding, επιλέγοντας $2p_h = f_h - 1, 2p_w = f_w - 1$*.*

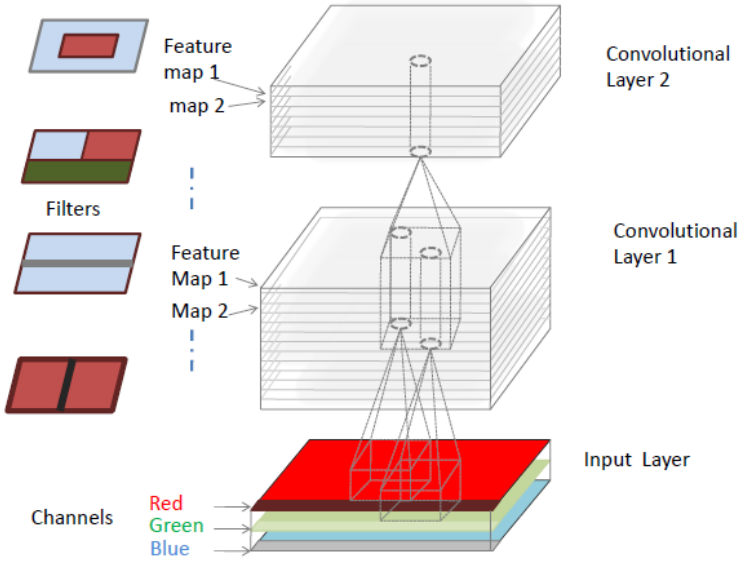
Ορισμός 4.3. *(Συνέλιξη με βήμα - Strided convolution)*
Αποτελεί μια παραλλαγή της συνέλιξης όπου το φίλτρο μετατοπίζεται με βήμα (stride) $s > 1$*. Η έξοδος έχει μικρότερο μέγεθος, επιτυγχάνεται δηλαδή downsampling και συγκεκριμένα (με βάση τους συμβολισμούς που δώσαμε στον προηγούμενο ορισμό και strides* s_h, s_w*) θα έχει μέγεθος* $\lfloor \frac{x_h - f_h + 2p_h + s_h}{s_h} \rfloor \times \lfloor \frac{x_w - f_w + 2p_w + s_w}{s_w} \rfloor$*.*

Πολλαπλά κανάλια εισόδου και εξόδου
Τα δεδομένα εικόνας συχνά δεν αποτελούνται μόνο από ένα πίνακα, αλλά πολλαπλούς δυδιάστατους πίνακες, καθένας από τους οποίους ονομάζεται κανάλι (channel) (Για παράδειγμα εικόνες RGB ή υπερφασματικές εικόνες που προέρχονται από δορυφόρους). Σε μια τέτοια περίπτωση, θα πρέπει να δημιουργηθεί ένα φίλτρο για κάθε ένα από τα κανάλια εισόδου. Επίσης, λόγω του ότι κάθε φίλτρο μπορεί να ανιχνεύσει μόνο ένα είδος χαρακτηριστικού, αν επιθυμούμε να ανιχνεύουμε πολλαπλά χαρακτηριστικά θα πρέπει να χρησιμοποιήσουμε ξεχωριστά φίλτρα για κάθε χαρακτηριστικό. Για να το επιτύχουμε αυτό μετατρέπουμε το φίλτρο $\boldsymbol{W}$ σε ένα τετραδιάστατο πίνακα βαρών (Έχουμε ένα δυδιάστατο πίνακα βαρών για κάθε κανάλι εισόδου και για κάθε χαρακτηριστικό που θέλουμε να ανιχνεύσουμε) [32]. Επίσης μεταβάλλουμε τον ορισμό

της συνέλιξης που δώσαμε για να αντιστοιχεί στα καινούρια χαρακτηριστικά της εισόδου και του πίνακα βαρών:

$$z_{i,j,d} = b_d \sum_{u=0}^{f_h-1} \sum_{v=0}^{f_w-1} \sum_{c=0}^{C-1} w_{u,v,c,d} x_{si+u,sj+v,c} \tag{4.4}$$

όπου C ο αριθμός των καναλιών και s το βήμα (stride).



Σχήμα 4.4: Παράδειγμα ενός CNN με δύο συνελικτικά επίπεδα όπου η είσοδος έχει 3 κανάλια (red, green, blue). Πηγή : Géron, A. (2019). Hands-on machine learning with Scikit-Learn and TensorFlow: Concepts, tools, and techniques for building intelligent systems (2nd ed.). O'Reilly Media.

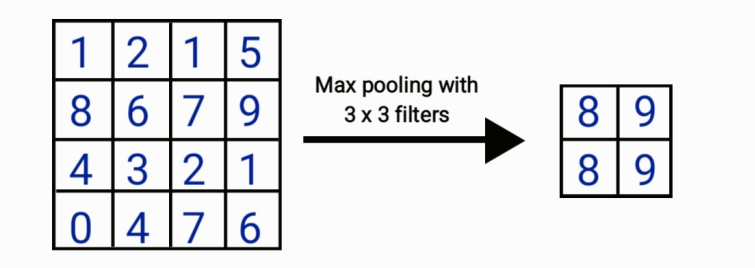
Το διάνυσμα των τιμών των χαρτών χαρακτηριστικών σε ένα συγκεκριμένο σημείο $z_{i,j1:D}$ ονομάζεται συχνά υπερστήλη (hypercolumn).

4.1.2 Επίπεδο Συγκέντρωσης

Τα επίπεδα συγκέντρωσης (pooling layers) παρεμβάλονται συχνά μεταξύ δύο επιπέδων συνέλιξης με σκοπό την μείωση της διάστασης των χαρτών χαρακτηριστικών, διατηρώντας παράλληλα τα πιο σημαντικά χαρακτηριστικά. Τα επίπεδα συγκέντρωσης δεν περιέχουν βάρη αλλά εφαρμόζουν μια συνάρτηση σε μικρές υποπεριοχές της εισόδου, μειώνοντας με αυτό τον τρόπο τη διάσταση της (down-sampling /υπο-δειγματοληψία) [33]. Τα τρια βασικά είδη συγκέντρωσης (pooling) είναι:

- Max Pooling : Από κάθε υποπεριοχή της εικόνας επιλέγεται το μεγαλύτερο στοιχείο.
- Average Pooling : Από κάθε υποπεριοχή της εικόνας υπολογίζεται και κρατείται ο μέσος όρος των στοιχείων.

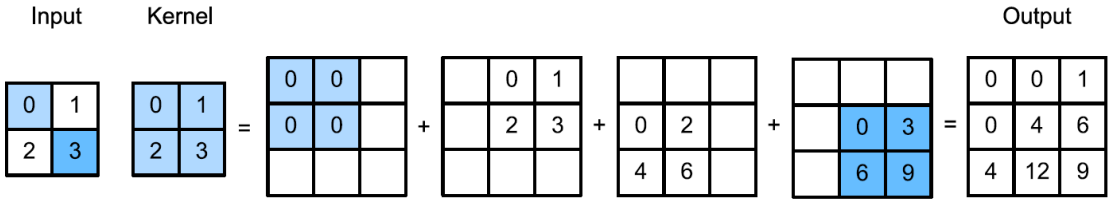
- Sum Pooling : Από κάθε υποπεριοχή της εικόνας υπολογίζεται και κρατείται το άθροισμα των στοιχείων.



Σχήμα 4.5: Παράδειγμα Max Pooling με φίλτρο μεγέθους 3×3.

4.1.3 Επίπεδο Αντρίστροφης Συνέλιξης

Τα συνελικτικά νευρωνικά δίκτυα κωδικοποιητή-αποκωδικοποιητή (encoder-decoder CNNs), όπως είναι και το U-Net το οποίο θα μελετήσουμε εκτενώς στην παράγραφο 4.2 αρχικά συμπιέζουν την είσοδο, χρησιμοποιώντας επίπεδα συνέλιξης και συγκέντρωσης και στη συνέχεια χρησιμοποιούν αυτή τη συμπιεσμένη πληροφορία για να ανακατασκευάσουν κάποια άλλη πληροφορία που συχνά έχει την ίδια διάσταση με την είσοδο. Για να επιτευχθεί αυτό χρειαζόμαστε τεχνικές υπερδειγματοληψίας (upsampling) οι οποίες αυξάνουν την διάσταση της εισόδου τους. Μια από αυτές τις τεχνικές είναι η αντίστροφη συνέλιξη (transposed convolution) κατά την οποία χρησιμοποιείται ένας πυρήνας, όπως και στα επίπεδα συνέλιξης. Στην αντίστροφη συνέλιξη, κάθε ένα από τα στοιχεία της εισόδου πολλαπλασιάζεται με καθένα από τα στοιχεία του πυρήνα και στη συνέχεια αθροίζουμε τα αποτελέσματα. Πιο αναλυτικά η διαδικασία παρουσιάζεται στο σχήμα 4.6. [34]



Σχήμα 4.6: Παράδειγμα αντίστροφης συνέλιξης για 2x2 είσοδο και 2x2 πυρήνα. Πηγή: Murphy, K. P. (2022). Probabilistic machine learning: an introduction. MIT press.

4.1.4 Επίπεδο Κανονικοποίησης

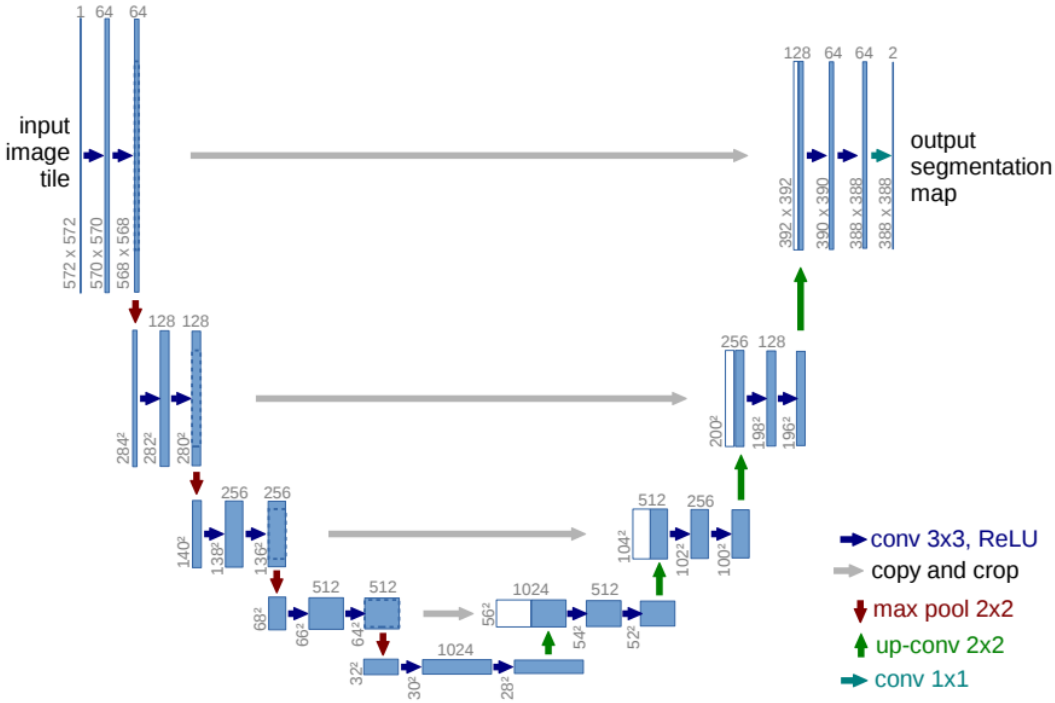
Τα επίπεδα κανονικοποίησης (normalization layers) χρησιμοποιούνται για να κανονικοποιήσουν τις τιμές που προκύπτουν μετά από την εφαρμογή των συναρτήσεων ενεργοποίησης (activation function) σε κάθε επίπεδο. Έχουν προταθεί διάφορες μεθόδοι κανονικοποιήσης ανάμεσα στις οποιές είναι το Batch Normalization [35] και το Group Normalization [36], το οποίο θα εφαρμόσουμε στο δικό μας μοντέλο στη συνέχεια.

Στο Batch Normalization οι ενεργοποιήσεις κανονικοποιούνται έτσι ώστε σε κάθε mini-batch να έχουν μέση τιμή 0 και διασπορά 1.

Στο Group Normalization, αντί αυτή η κανονικοποίηση να γίνεται ανά mini-batch, δημιουργούμε ομάδες διαιρώντας τα κανάλια της εικόνας και κανονικοποιούμε με βάση αυτές τις ομάδες. Με αυτό τον τρόπο αποφεύγεται η εξάρτηση της κανονικοποίησης από το batch-size που επιλέγεται κάθε φορά και έτσι η φάση εκπαίδευσης του μοντέλου είναι πιο σταθερή ακόμη και για μικρά batch-sizes.

4.2 Νευρωνικό Δίκτυο U-Net

Το U-Net είναι ένα είδος συνελικτικού νευρωνικού δικτύου το οποίο προτάθηκε από τους Ronneberger, Fischer και Brox [37] με κύριο σκοπό τη χρήση του σε προβλήματα τμηματοποίησης/κατάτμησης εικόνων (image segmentation) και πήρε το όνομα του λόγω του ότι η σχηματική αναπαράσταση του δικτύου μοιάζει με το αγγλικό γράμμα U, όπως φαίνεται και στο σχήμα 4.7.



Σχήμα 4.7: Σχηματική αναπαράσταση του δικτύου U-Net. Πηγή: Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18 (pp. 234-241). Springer international publishing.

Η αρχιτεκτονική του U-Net βασίζεται στο πρότυπο μοντέλων κωδικοποιητή - αποκωδικοποιητή (encoder - decoder) . Η αρχιτεκτονική κωδικοποιητή - απωκιδοκοποιητή είναι μια αρχιτεκτονική βαθειάς μάθησης η οποία χρησιμοποιείται για διάφορα προβλήματα επεξεργασίας φυσικής γλώσσας (Natural Language Proccessing tasks) και μηχανικής όρασης (Computer Vision).

Το U-Net σχεδιάστηκε ώστε να μπορεί να εκμεταλλεύεται αποδοτικά ένα μικρό αριθμό από δεδομένα, διατηρώντας παράλληλα ακρίβεια και ταχύτητα. Αποτελείται από ένα συμπτησσόμενο μονοπάτι, τον κωδικοποιητή (encoder) και ένα αναπτυσσώμενο μονοπάτι, τον αποκωδικοποιητή (decoder).

Ο κωδικοποιητής είναι υπεύθυνος για να ανιχνεύει μοτίβα στην είσοδο. Περιέχει διαδοχικά επίπεδα συνέλιξης , συγκέντρωσης και ενδεχομένως κανονικοποίησης με σκοπό την μείωση της χωρικής διάστασης των χαρτών χαρακτηριστικών και την αύξηση, παράλληλα, του βάθους τους, ώστε να μπορούν να ανιχνεύουν διάφορα και πιο περίπλοκα μοτίβα στην είσοδο.

Αντίστοιχα, ο αποκωδικοποιητής περιέχει επίπεδα συνέλιξης αλλά και αντίστροφης συνέλιξης με σκοπό την αποκωδικοποίηση των κωδικοποιημένων δεδομένων που προκύπτουν από το πρώτο μέρος του νευρωνικού δικτύου και την κατασκευή ενός χάρτη τμηματοποίησης/ κατάτμησης της εικόνας.

Το κεντρικό σημείο του δικτύου ονομάζεται "bottleneck" και είναι αυτό που συνδέει το συμπτησσόμενο με το αναπτυσσόμενο μονοπάτι και περιέχει την πληροφορία στην πιο συμπιεσμένη της μορφή.

Στο σχήμα 4.7 μπορούμε να παρατηρήσουμε πφως μια μονοχρωματική εικόνας μεγέθους $572 \times 572 \times 1$ εισέρχεται στο συγκεκριμένο δίκτυο U-Net και μετατρέπεται σε ένα δυαδικό χάρτη τμηματοποίησης μεγέθους $388 \times 388 \times 2$. Αν και στη συγκεκριμένη περίπτωση η έξοδος δεν έχει την ίδια χωρική διάσταση με αυτή της εισόδου, αυτό θα μπορούσε εύκολα να γίνει χρησιμοποιόντας την τεχνική του συμπληρώματος με μηδενικά (zero padding) που συζητήσαμε παραπάνω , κάτι που θα μας φανεί ιδιαίτερα χρήσιμο στη δική μας περίπτωση. Παρατηρούμε επίσης ότι καθώς η εικόνα διέρχεται από το συμπτησσόμενο μονοπάτι, το μήκος και το πλάτος της μειώνονται ενώ ο αριθμός των καναλιών αυξάνεται. Το γεγονός αυτό επιτρέπει να ανιχνευθούν περίπλοκα μοτίβα υψηλού επιπέδου καθώς η εικόνα διέρχεται μέσα από το μονοπάτι.

Στο σχήμα 4.7 παρατηρούμε επίσης κάποια βέλη τα οποία συνδέουν τα επίπεδα του κωδικοποιητή με επίπεδα του αποκωδικοποιητή. Τα βέλη αυτά ονομάζονται συνδέσεις παράκαμψης (skip connections) και χρησιμοποιούνται κυρίως για να μεταφέρουν πληροφορία στον αποκωδικοποιητή που πιθανόν να χάνεται κατά την υποδειγματοληψία (downsampling) που εκτελείται στον κωδικοποιητή. Η είσοδος σε κάθε στάδιο του αποκωδικοποιητή λοιπόν, προκύπτει από την ένωση (concatenation) της αντίστοιχης πληροφορίας που προέρχεται από τον κωδικοποιητή, και παρακάμπτει τα ενδιάμεσα επίπεδα, και της εξόδου του προηγούμενου επιπέδου επιπέδου του αποκωδικοποιητή, η οποία επιδέχεται υπερδειγματοληψία (upsampling), μέσω της εφαρμογής ενός επιπέδου αντίστροφης συνέλιξης. [38]

Στα μοντέλα με τα οποία ασχολούμαστε στην εν λόγω εργασία το U-Net δεν χρησιμοποιείται για προβλήματα κατάτμησης εικόνας (image segmentation tasks). Αντίθετα, στην περίπτωση των Denoising Diffusion Probabilistic Models και άλλων παρόμοιων πιθανοθεωρητικών μοντέλων διάχυσης το U-Net χρησιμοποιείται είτε για την πρόβλεψη της αρχικής εικόνας από την θορυβοποιημένη είτε, πιο συχνά, για την πρόβλεψη του θορύβου που έχει προστεθεί στην εικόνα κατά την διαδικασία της διάχυσης σε μια δεδομένη στιγμή. Εκτιμώντας λοιπόν τον θόρυβο που έχει προστεθεί σε κάθε χρονική στιγμή μας επιτρέπει

να ανακτήσουμε σταδιακά την εικόνα μέσω της αντίστροφης διαδικασία (reverse process).

Αφού σε κάθε χρονική στιγμή ο τυχαίος θόρυβος που προστίθεται σε μια εικόνα είναι διαφορετικού βαθμού και εξαρτάται από το πρόγραμμα διακύμανσης (variance schedule) που έχει επιλεχθεί θα πρέπει μάζι με την θορυβοποιημένη εικόνα, το νευρωνικό δίκτυο να παίρνει και ως είσοδο την χρονική στιγμή σε κάποια μορφή. Για την επίλυση του συγκεκριμένου προβλήματος έχει προταθεί η χρήση των time embeddings, τα οποία συζητούμε στην παράγραφο 4.3.1 .

4.3 Αρχιτεκτονικά Στοιχεία και Τεχνικές Βελτιστοποίησης Νευρωνικών Δικτύων

Στην ενότητα αυτή θα παρουσιάσουμε κάποια βασικά δομικά στοιχεία καθώς και τεχνικές βελτιστοποίησης νευρωνικών δικτύων, που χρησιμοποιούνται ευρέως σε σύγχρονες αρχιτεκτονικές, ιδιαίτερα σε εφαρμογές όπως η επεξεργασία εικόνας, η παραγωγή δεδομένων και ειδικότερα σε αρχιτεκτονικές που χρησιμοποιούνται σε μοντέλα διάχυσης. Συγκεκριμένα μελετάμε τα Time Embeddings, τα οποία προσθέτουν χρονική πληροφορία στο μοντέλο, τα Residual Blocks, που επιτρέπουν την εκπαίδευση βαθύτερων μοντέλων, χωρίς την υποβάθμιση της απόδοσης τους και τους μηχανισμούς προσοχής (Attention Mechanisms) και τα Attention Blocks που ενισχύουν την ικανότητα του μοντέλου να επικεντρώνεται στις πιο "σημαντικές" περιοχές της εισόδου. Επίσης, αναλύουμε δύο σημαντικές τεχνικές, το Dropout και το EMA (Exponential Moving Average - Εκθετικός κινητός μέσος όρος) που χρησιμοποιούνται για αποτροπή της υπερεκπαίδευσης (overfitting) και σταθεροποίηση των παραμέτρων κατά την εκπαίδευση αντίστοιχα.

4.3.1 Time Embeddings

Σκοπός των Time Embeddings είναι να εισάγουν στο νευρωνικό δίκτυο πληροφορία για το στάδιο της προς τα εμπρός διαδικασίας (forward process) στο οποίο πάρθηκε η θορυβοποιημένη εικόνα. Η πιο συχνά χρησιμοποιούμενη τεχνική για ενσωμάτωση της πληροφορίας για το χρόνο είναι εμπνευσμένη από τα positional encodings που χρησιμοποιούνται στα μοντέλα μετασχηματιστών (transformers) [39]. Χρησιμοποιείται λοιπόν το λεγόμενο sinusoidal time embedding , το οποίο ορίζεται ως εξής :

$$PE(t)_{2i} = sin(\frac{t}{10000^{2i/d}}) \tag{4.5}$$

$$PE(t)_{2i+1} = cos(\frac{t}{10000^{2i/d}}) \tag{4.6}$$

όπου d είναι η επιθυμητή διάσταση του embedding (π.χ. 128,256,512). Παρόλο που η τεχνική αυτή επιτυγχάνει μια ισχυρή αναπαράσταση της πληροφορίας για το χρόνο, συχνά το time embedding περνά από περεταίρω επεξεργασία πρωτού εισαχθεί στο U-Net μέσω ενός μικρού Multi-Layer Perceptron (MLP) με μια μη γραμμική συνάρτηση ενεργοποίησης, επιτρέπωντας με αυτό τον τρόπο

στο μοντέλο μας να εκπαιδευτεί στο πως να χειρίζεται την χρονική πληροφορία όσο το δυνατό πιο αποδοτικά. Για παράδειγμα χρησιμοποιώντας ένα MLP με δύο γραμμικά επίπεδα και ένα ενδιάμεσο μη γραμμικό, έστω που χρησιμοποιεί την SiLU(Sigmoid Linear Unit), το τελικό time embedding που θα εισαχθεί στο U-Net θα ήταν το εξής :

$$t_{emb} = Linear_2(SiLU(Linear_1(e))) \tag{4.7}$$

όπου $e = PE(t)$ το sinusoidal embedding.

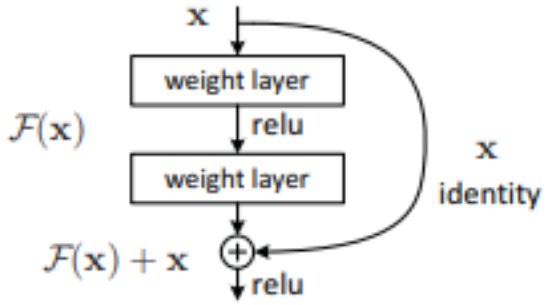
Έχοντας πλέον το time embedding, πρέπει να καθορίσουμε πως αυτό θα εισαχθεί στο νευρωνικό μας δίκτυο. Υπάρχουν 2 κύριες επιλογές. Στην πρώτη (Additive Intergration) το time embedding προστίθεται σε κάποιο χάρτη χαρακτηριστικών ενώ στη δεύτερη (Adaptive Normalization) τα time embeddings χρησιμοποιούνται για να ρυθμίσουν τις παραμέτρους στα επίπεδα κανονικοποίησης. Στην δική μας περίπτωση, χρησιμοποιείται η πρώτη επιλογή.

4.3.2 Residual Blocks

Την τελευταία δεκαετία, η χρήση βαθειάς μάθησης (deep learning) και βαθειών νευρωνικών δικτύων (deep neural networks) έχει επιφέρει σημαντικά αποτελέσματα και αξιοσημείωτες βελτιώσεις στον τομέα της μηχανικής μάθησης, επιτυγχάνοντας προβλέψεις με αρκετά υψηλή ακρίβεια. Ένα ερώτημα λοιπόν που επακολουθεί είναι αν μπορούμε συνεχώς να βελτιώνουμε την ακρίβεια των προβλέψεων προσθέτοντας ακόμη περισσότερα επίπεδα στο νευρωνικό μας δίκτυο, έτσι ώστε να μπορεί να αντιλαμβάνεται πιο περίπλοκα μοτίβα και συσχετίσεις μεταξύ της εισόδου και της εξόδου. Ενώ διαισθητικά περιμένουμε ότι η απάντηση σε αυτή την ερώτηση θα είναι καταφατική, σε πολλές περιπτώσεις η προσθήκη περισσότερων επιπέδων, όχι μόνο δεν βελτιώνει την επίδοση του μοντέλου αλλά την επιδεινώνει. Το πρόβλημα αυτό είναι γνωστό ως "degradation" (υποβάθμιση) και το παράδοξο είναι ότι η επιδείνωση των αποτελεσμάτων δεν οφείλεται σε προβλήματα υπερπροσαρμογής, όπως θα περιμέναμε, καθώς χειροτερεύει και η επίδοση των μοντέλων στο σύνολο δεδομένων εκπαίδευσης.

Την λύση σε αυτά τα προβλήματα έρχονται να δώσουν τα "Residual Blocks" τα οποία προτάθηκαν για πρώτη φορά από τους He, Zhang, Ren και Sun στην δημοσίευση τους "Deep Residual Learning for Image Recognition" , το 2015 [40]. Αν αρχικά, στόχος του νευρωνικού δικτύου ήταν να προσεγγίσει μια συνάρτηση $\mathcal{H}(\mathbf{x})$ όπου $\mathbf{x}$ το διάνυσμα εισόδου, στα Residual Blocks χρησιμοποιούνται συνδέσεις παράκαμψης (skip connections), τις οποίες συζητήσαμε και παραπάνω, με στόχο να προσεγγίζεται η συνάρτηση $\mathcal{H}(\mathbf{x})-\mathbf{x}$. Ένα παράδειγμα ενός Residual Block παρουσιάζεται στο σχήμα 4.8.

Η χρήση των Residual Block έχει αποδεχθεί πειραματικά ότι αντιμετωπίζει τα προβλήματα επιδείνωσης της επίδοσης ενός μοντέλου καθώς αυξάνουμε το βάθος του. Με αυτόν τον τρόπο, επιτρέπεται η εκπαίδευση βαθύτερων νευρωνικών δικτύων με υψηλότερη ακρίβεια στις προβλέψεις.



Σχήμα 4.8: Residual Block Πηγή: He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

4.3.3 Μηχανισμοί Προσοχής (Attention Mechanisms) και Attention Blocks

Οι μηχανισμοί προσοχής (Attention Mechanisms) επέφεραν επαναστατικές αλλαγές και βελτιώσεις στον τομέα της βαθειάς μάθησης, επιτρέποντας στα μοντέλα να επικεντρώνονται στα πιο "σημαντικά" κομμάτια της εισόδου. Με τον όρο "σημαντικά" εννούμε τα κομάτια της εισόδου που δείχνουν να έχουν τη μεγαλύτερη συσχέτιση με την έξοδο.

Οι συγκεκριμένες τεχνικές αναπτύχθηκαν αρχικά για προβλήματα επεξεργασίας φυσικής γλώσσας (Natural Language Processing). Στη συνέχεια όμως οι μηχανισμοί αυτοί, ή διάφορες παραλλαγές τους χρησιμοποιήθηκαν και για προβλήματα διαφορετικής φύσεως, όπως είναι αυτά της μηχανικής όρασης (Computer Vision).

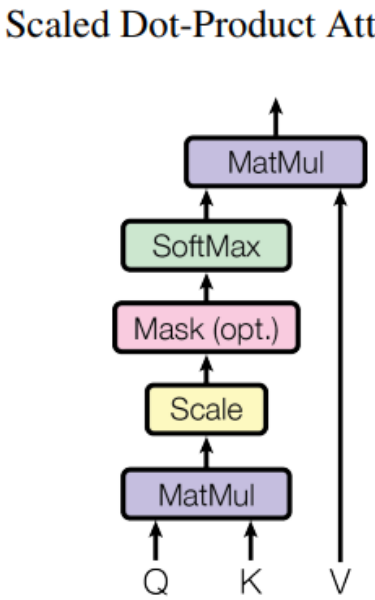
Ειδικότερα, θα παρουσιάζουμε την τεχνική του Self-Attention, η οποία χρησιμοποιείται έτσι ώστε το μοντέλο να αντιλφθεί τη συσχέτιση μεταξύ διαφορετικών σημείων της εισόδου και υλοποιείται υπολογίζοντας ένα σκορ για κάθε συνδυασμό δύο "κομματιών" της εικόνας.

Στους μηχανισμούς προσοχής, η συνάρτηση που εφαρμόζεται μπορεί να χαρακτηριστεί ως μια αντιστοιχεία ενός ερωτήματος (query) και ενός συνόλου από ζεύγη κλειδιού-τιμής (key-value) σε μια έξοδο, όπου το ερώτημα, το κλειδί, η τιμή και η έξοδος είναι όλα διανύσματα. Η έξοδος υπολογίζεται ως ένα άθροισμα με βάρη των τιμών, όπου το βάρος κάθε τιμής υπολογίζεται χρησιμοποιώντας μια συνάρτηση "συμφωνίας" μεταξύ ερωτήματος και κλειδιού. Οι συγγραφείς στο [39] προτείνουν τη λεγόμενη "Scaled Dot-Product Attention" (Σχήμα 4.9) , κατά την οποία η είσοδος απαρτίζεται από ερωτήματα(queries) και κλειδιά(keys) διάστασης d_k και τιμές (values) διάστασης d_v. Αρχικά υπολογίζεται το εσωτερικό γινόμενο του κάθε ερωτήματος με κάθε κλειδί (λειτουργεί ως συνάρτηση "συμφωνίας" μεταξύ ερωτήματος και κλειδιού) το οποίο κανονικοποιούμε διαιρόντας με $\sqrt{d_k}$ και έπειτα εφαρμόζεται η συνάρτηση softmax για να υπολογιστούν τα βάρη. Τέλος η έξοδος (value) υπολογίζεται ως το άθροισμα με βάρη των τιμών.

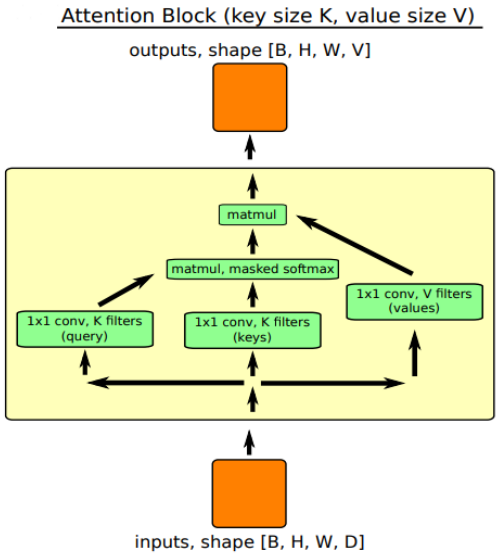
Οι υπολογισμοί αυτοί μπορούν να γίνουν ταυτόχρονα, μαζεύοντας τα ερωτήματα, τα κλειδιά και τις τιμές στους πίνακες Q, K, V αντίστοιχα και υπολο-

γίζοντας τον πίνακα των εξόδων ως:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{4.8}$$



Σχήμα 4.9: Scaled Dot-Product Attention. Πηγή: Vaswani, A., et al. (2017). *Attention is all you need*. NeurIPS, 30.



Σχήμα 4.10: Attention Block. Πηγή: Chen, X., et al. (2018). *Pixelsnail*. ICML.

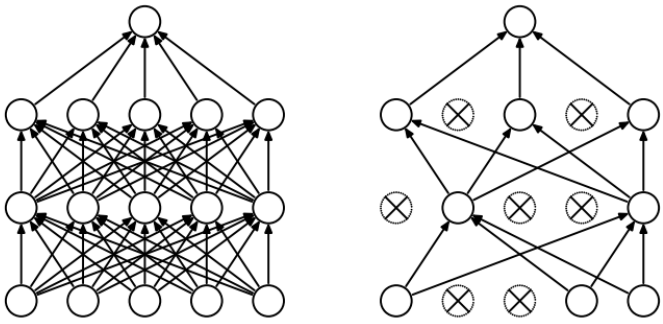
Στην δική μας περίπτωση όπου η είσοδος είναι εικόνα, εφαρμόζουμε μεταξύ διαδοχικών μπλοκ συνελίξεων, ένα Attention Block κατά το οποίο, όπως φαίνεται και στο σχήμα 4.10, χρησιμοποιούμε την είσοδο για να δημιουργήσουμε τα διανύσματα ερωτημάτων, κλειδιών και τιμών μέσα από συνελίξεις με πυρήνα μεγέθους 1×1 και το επιθυμητό μέγεθος καναλιών και έπειτα εφαρμόζουμε την πιο πάνω διαδικασία. [41] Με αυτό τον τρόπο επιτυγχάνουμε ότι κάθε pixel της εικόνας μαζεύει πληροφορία από όλα τα υπόλοιπα pixel , βασιζόμενη στο πόσο σχετική είναι αυτή η πληροφορία.

4.3.4 Dropout

Η υπερπροσαρμογή (overfitting) είναι ένα σύνηθες πρόβλημα στη βαθειά μάθηση κατά το οποίο ένα νευρωνικό δίκτυο αποδίδει ικανοποιητικά σε ότι αφορά το σύνολο δεδομένων εκπαίδευσης, αλλά δεν επιτυγχάνει την ίδια απόδοση σε καινούρια δεδομένα.

Μια τεχνική για την αντιμετώπιση του συγκεκριμένου προβλήματος είναι το Dropout [42]. Το Dropout είναι μια μέθοδος κανονικοποίησης κατά την οποία ανάλογα με μια παράμετρο η οποία ονομάζεται "dropout rate", σε κάθε επανάληψη της εκπαίδευσης "απενεργοποιείται" τυχαία ένα ποσοστό των νευρώνων

με σκοπό να αποφευχθεί η υπερπροσαρμογή. Μετά από κάθε επανάληψη επαναφέρουμε τους νευρώνες που "απενεργοποιήθηκαν" και επιλέγονται τυχαία κάποιοι άλλοι.



(a) Standard Neural Net (b) After applying dropout.

Σχήμα 4.11: Παράδειγμα χρήσης του Dropout σε ένα Multi-Layer Perceptron. Πηγή: SrivastavaN, H., & Krizhevsky, A. (2014). Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 15(1), 1929.

4.3.5 Εκθετικός Κινητός Μέσος Όρος

Ο Εκθετικός Κινητός Μέσος Όρος (Exponential Moving Average, EMA για συντομία) είναι μια τεχνική που χρησιμοποιείται συχνά σε νευρωνικά δίκτυα που βελτιστοποιούνται με στοχαστικές μεθόδους (Stochastic gradient descent, Mini-batch gradient descent) για την πιο ομαλή ενημέρωση των βαρών μετά από κάθε βήμα του αλγορίθμου. Συγκεκριμένα χρησιμοποιείται ο εξής τύπος για την ενημέρωση των βαρών:

$$\mathbf{x}_{t+1}^{EMA} = a\mathbf{x}_{t}^{EMA} + (1-a)\mathbf{x}_{t+1} \tag{4.9}$$

όπου η παράμετρος a ονομάζεται συντελεστής απομείωσης (decay factor) και καθορίζει τον ρυθμό με τον οποίο ενσωματώνεται η νέα πληροφορία στον κινητό μέσο.

Η τεχνική αυτή είναι πολύ σημαντική καθώς επιτρέπει μια πιο σταθερή εκπαίδευση του μοντέλου, λιγότερο επιρρεπή στο θόρυβο.

5 Εκπαίδευση Νευρωνικών Δικτύων

Στην ενότητα αυτή θα μελετήσουμε τις βασικές ιδέες που αφορούν την εκπαίδευση νευρωνικών δικτύων και την εκτίμηση των βέλτιστων παραμέτρων, όσον αφορά κάποια συνάρτηση απώλειας (loss function). Η εκπαίδευση ενός νευρωνικού δικτύου προϋποθέτει την ύπαρξη ενός συνόλου δεδομένων εκπαίδευσης των οποίων η έξοδος είναι γνωστή. Το νευρωνικό δίκτυο κάνει προβλέψεις πάνω στα δεδομένα εκπαίδευσης και έπειτα συγκρίνει τις προβλέψεις με τις πραγματικές εξόδους ώστε να υπολογίζει το σφάλμα. Έπειτα χρειάζεται να ακολουθηθεί μια διαδικασία ενημέρωσης των βαρών με βάση τα υπολογιζόμενα σφάλματα. Η διαδικασία αυτή επαναλαμβάνεται αρκετές φορές εώς ότου φτάσουμε σε ένα προκαθορισμένο αριθμό επαναλήψεων ή μέχρι η συνάρτηση απώλειας να φτάσει κάτω από κάποιο επιθυμητό όριο [44].

5.1 Gradient Descent (GD)

Η κατάβαση δυναμικού (gradient descent) είναι η τεχνική βελτιστοποιήσης που κυριαρχεί στα σύγχρονα νευρωνικά δίκτυα. Η τεχνική αυτή εκμεταλλεύεται την βαθμίδα (gradient) της συνάρτησης απώλειας, η οποία δίνει πληροφορία για το πως μεταβάλλεται η συνάρτηση απώλειας καθώς μεταβάλλουμε τους συντελεστές του μοντέλου σε κάθε κατεύθυνση. Χρησιμοποιώντας λοιπόν την βαθμίδα μπορούμε σε κάθε επανάληψη να ενημερώνουμε τα βάρη κατάλληλα με σκοπό την ελαχιστοποίηση της συνάρτησης απώλειας. Τα βήματα της μεθόδου είναι τα ακόλουθα [45]:

1. Ξεκινάμε δίνοντας κάποιες αρχικές τιμές στα βάρη.
2. Υπολογίζουμε τις εξόδους του νευρωνικού δικτύου με τις τρέχουσες τιμές των βαρών για τα δεδομένα εκπαίδευσης σε μια προς τα εμπρός διαπέραση του νευρωνικού δικτύου (forward pass).
3. Υπολογίζουμε την τιμή της συνάρτησης απώλειας.
4. Υπολογίζουμε τη βαθμίδα.
5. Ενημερώνουμε τα βάρη και επαναλαμβάνουμε.

Για την ενημέρωση των βαρών είναι σημαντικό επίσης να οριστεί ένας συντελεστής, ο οποίος ονομάζεται ρυθμός μάθησης (learning rate) και καθορίζει το πόσο έντονα αλλάζουν οι τιμές των βαρών σε κάθε επανάληψη. Η διαδικασία αυτή επαναλαμβάνεται είτε εώς ώτου φτάσουμε ένα συγκεκριμένο αριθμό επαναλήψεων είτε μέχρι να φτάσει η συνάρτηση απώλειας κάτω από ένα επιτρεπτό όριο.

Η βαθμίδα της συνάρτησης (gradient) προσδιορίζει την κατεύθυνση προς την οποία πρέπει να κινηθούμε σε κάθε επανάληψη ώστε να ελαχιστοποιήσουμε όσο το δυνατόν περισσότερο την συνάρτηση απώλειας (direction of steepest descend). Η βαθμίδα όμως καθορίζει όπως είπαμε μόνο την κατεύθυνση προς την οποία πρέπει να κινήθουμε και όχι το μέγεθος του κάθε βήματος. Εδώ

χρησιμοποιείται ο ρυθμός μάθησης (learning rate) που αναφέραμε παραπάνω.Η παράμετρος αυτή είναι μια από τς σημαντικότερες υπερπαραμέτρους αφού πολύ μεγάλος ρυθμός μάθησης μπορεί να οδηγήσει σε ταλαντώσεις μεταξύ σημείων που δίνουν την ίδια τιμή στη συνάρτηση απώλειας και δεν συγκλίνουν ποτέ, ενώ πολύ μικρός ρυθμός μάθησης, οδηγεί σε μια πολύ αργή διαδικασία μάθησης. Επομένως η παράμετρος αυτή πρέπει να επιλέγεται κατάλληλα κάθε φορά ώστε να αποφεύγονται τα προαναφερθέντα προβλήματα, ενώ συχνά μπορεί να επαναληφθεί η διαδικασία μάθησης με διαφορετικές τιμές για το ρυθμό μάθησης, ώστε να επιλεχθεί η κατάλληλη, που δίνει τελικά την μικρότερη τιμή στην συνάρτηση απώλειας.

Παρόλο που η κατάβαση δυναμικού (gradient descent) είναι ένας αποτελεσματικός αλγόριθμος για την εύρεση του ελάχιστου της συνάρτησης απώλειας, έχει δύο σημαντικά μειονεκτήματα [46]:

1. Η αρχικοποίηση των βαρών καθορίζει σημαντικά τη λύση που θα προκύψει από τον αλγόριθμο καθώς περίπλοκες συναρτήσεις μπορούν να έχουν πολλά τοπικά ελάχιστα, στα οποία η μέθοδος μπορεί να παγιδευτεί και να μην μπορέσει ποτέ να φτάσει στο ολικό ελάχιστο.

2. Η απλή εκδοχή του αλγορίθμου χρησιμοποιεί ολόκληρο το σύνολο εκπαίδευσης για να υπολογίσει τη βαθμίδα σε κάθε επανάληψη και επομένως είναι υπολογιστικά " ακριβή " και αργή. Η εκδοχή αυτή ονομάζεται επίσης και batch gradient descend λόγω του γεγονότος ότι χρησιμοποιεί όλα τα δεδομένα μάθησης σε ένα "batch".

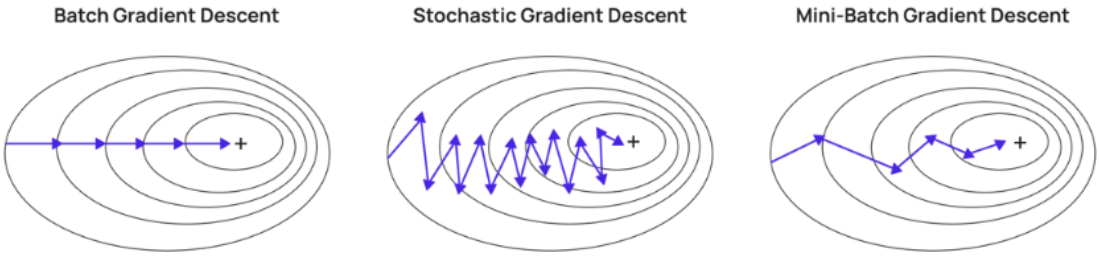
Για την αντιμετώπιση των μειονεκτημάτων αυτών χρησιμοποιείται η μέθοδος της στοχαστικής κατάβασης δυναμικού (Stochastic Gradient Descent) την οποία θα μελετήσουμε στη συνέχεια.

5.1.1 Stochastic Gradient Descent (SGD)

Η στοχαστική κατάβαση δυναμικού (Stochastic Gradient Descent) είναι μια παραλλαγή της μεθόδου κατάβασης δυναμικού (Gradient Descent) κατά την οποία σε κάθε επανάληψη δεν υπολογίζουμε τη βαθμίδα χρησιμοποιώντας ολόκληρο το σύνολο δεδομένων εκπαίδευσης. Ανταυτού επιλέγεται τυχαία ένα δείγμα από τα δεδομένα εκπαίδευσης και υπολογίζεται η βαθμίδα βάση αυτού. Με αυτό τον τρόπο μειώνεται σε σημαντικό βαθμό η υπολογιστική πολυπλοκότητα του αλγορίθμου, ενώ επίσης αντιμετωπίζεται και το πρόβλημα των τοπικών ελαχίστων, αφού η στοχαστική φύση του SGD επιτρέπει, τις περισότερες φορές, την λύση να φτάσει "πολύ κοντά" στο ολικό ελάχιστο της συνάρτησης απώλειας.

5.1.2 Mini-batch Gradient Descent (MB-GD)

Μια ακόμη παραλλαγή του αρχικού αλγορίθμου που αποτελεί μια "μέση λύση" μεταξύ των δύο προηγούμενων τεχνικών είναι το Mini-batch Gradient Descend, κατά την οποία αντί να υπολογίζεται η βαθμίδα από ένα δείγμα, όπως στο SGD, ή από όλο το σύνολο δεδομένων εκπαίδευσης, όπως στο GD,



Σχήμα 5.1: Σύγκριση των Batch Gradient Descent , Stochastic Gradient Descent και Mini Batch Gradient Descent. Πηγή : Illustration of gradient descent. Adapted from What is Gradient Descent?, by AlwaysAI (2024), https://alwaysai.co/blog/what-is-gradient-descent. Copyright 2024 by AlwaysAI.

τα δεδομένα εκπαίδευσης διαιρούνται σε μικρότερα σύνολα, που ονομάζονται mini-batches και έχουν σταθερό μέγεθος, και σε κάθε επανάληψη χρησιμοποιείται ένα mini-batch για τον υπολογισμό της βαθμίδας. Η παραλλαγή αυτή συγκλίνει επίσης πιο γρήγορα από ότι η κατάβαση δυναμικού (Gradient Descent / Batch Gradient Descent) καθώς σε μια επανάληψη, δηλαδή σε μια διαπέραση του συνόλου δεδομένων εκπαίδευσης, τα βάρη ενημερώνονται περισσότερες φορές (στον GD ενημερώνονται μόνο μια φορά ενώ στον MB-GD ενημερώνονται μετά το τέλος κάθε mini-batch, άρα ανάλογα με το πλήθος των mini-batches στα οποία διαιρέθηκε το αρχικό σύνολο δεδομένων).

Μια σχηματική σύγκριση των τριών τεχνικών παρουσιάζεται στο σχήμα 5.1.

5.1.3 Adam

Το ακρωνύμιο Adam προκύπτει από τις λέξεις ADAptive Moment Estimation και είναι ένας αλγόριθμος βελτιστοποίησης που χρησιμοποιείται συχνά σε προβλήματα μηχανικής μάθησης και ειδικότερα σε προβλήματα βαθειάς μάθησης. Είναι μια μέθοδος εύκολη στην υλοποίηση, υπολογιστικά αποδοτική και με μικρές απαιτήσεις όσο αφορά τη μνήμη, επομένως είναι κατάλληλη για προβλήματα όπου είτε έχουμε μεγάλο αριθμό δεδομένων στο σύνολο εκπαίδευσης, είτε έχουμε μεγάλο αριθμό παραμέτρων προς εκτίμηση. [47]. Επίσης, συνδυάζει τα πλεονεκτήματα δύο προκατόχων της στο πεδίο των αλγορίθμων βελτιστοποίησης στοχαστκής φύσεως, του AdaGrad [48] και του RMSProp [49].

Η μέθοδος βασίζεται σε προσαρμοζόμενες εκτιμήσεις των ροπών χαμηλής τάξης της βαθμίδας της συνάρτησης κόστους. Ειδικότερα χρησιμοποιούνται εκθετικοί κινητοί μέσοι όροι (exponential moving averages) για την εκτίμηση της πρώτης ροπής (μέσης τιμής) και της δεύτερης ροπής της βαθμίδας. [50] Δύο μη αρνητικές υπερπαραμέτροι, οι οποίες συμβολίζονται συνήθως με β_1, β_2 καθορίζουν το ρυθμό με τον οποίο κινούνται οι δύο κινητοί μέσοι όροι. Απαραίτητο είναι επίσης να καθοριστεί το λεγόμενο "learning rate", ή αλλιώς ρυθμός μάθησης, που καθορίζει το πόσο έντονα αλλάζουν οι τιμές των παραμέτρων σε κάθε επανάληψη, αλλά και να αρχικοποιηθούν οι παράμετροι του προβλήματος. Σε κάθε επανάληψη υπολογίζεται αρχικά η βαθμίδα της αντικειμενικής συνάρτησης για την τρέχουσα τιμή των παραμέτρων , έπειτα ενημερώνονται οι κινητοί μέσοι όροι και διορθώνονται ως προς τη μεροληψία και τέλος ενημερώνεται η

τιμή των παραμέτρων. Παρακάτω δίνεται ο ψευδωκώδικας του αλγορίθμου, σε μια μεταφρεσμένη εκδοχή του κώδικα που παρατίθεται στην δημοσίευση των Kingma και Lei Ba.

Algorithm 1: Adam: Adaptive Moment Estimation (Όπως παρουσιάζεται στην δημοσίευση των Kingma και Lei Ba [47]

Require: α: Βήμα εκπαίδευσης (stepsize)
Require: $\beta_1, \beta_2 \in [0, 1)$: Ρυθμοί εκθετικής αποσύνθεσης για τις εκτιμήσεις ροπών
Require: $f(\theta)$: Στοχαστική αντικειμενική συνάρτηση (Συνάρτηση απώλειας ή συνάρτηση κέρδους) με παραμέτρους θ
Require: θ_0: Αρχική συνθήκη για τις παραμέτρους
$m_0 \leftarrow 0$ (Αρχικοποίηση 1ης ροπής)
$v_0 \leftarrow 0$ (Αρχικοποίηση 2ης ροπής)
$t \leftarrow 0$ (Αρχικοποίηση χρονικού βήματος)
while θ_t not converged **do**
  $t \leftarrow t + 1$
  $g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$ (Υπολογισμός βαθμίδας)
  $m_t \leftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$ (Ενημέρωση της μεροληπτικής εκτίμησης της 1ης ροπής)
  $v_t \leftarrow \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$ (Ενημέρωση της μεροληπτικής εκτίμησης της 2ης ροπής)
  $\hat{m}_t \leftarrow m_t / (1 - \beta_1^t)$ (Υπολογισμός της διορθωμένης ως προς την μεροληψία εκτίμησης της 1ης ροπής)
  $\hat{v}_t \leftarrow v_t / (1 - \beta_2^t)$ (Υπολογισμός της διορθωμένης ως προς την μεροληψία εκτίμησης της 2ης ροπής)
  $\theta_t \leftarrow \theta_{t-1} - \alpha \cdot \hat{m}_t / (\sqrt{\hat{v}_t} + \epsilon)$ (Ενημέρωση παραμέτρων)
end while
return θ_t

5.2 Μέθοδος Οπισθοδιάδοσης (Backpropagation)

Έχουμε συζητήσει παραπάνω για το πως εκπαιδεύουμε τα νευρωνικά δίκτυα χρησιμοποιώντας διάφορες παραλλαγές της κατάβασης δυναμικού (Gradient Descent). Στις παραπάνω αναφορές μας όμως, δεν έχουμε διευκρινήσει τον τρόπο με τον οποίο υπολογίζεται η βαθμίδα της συνάρτησης απώλειας ως προς καθένα από τα βάρη του νευρωνικού δικτύου. [51] Χρειαζόμαστε λοιπόν ένα αλγόριθμο, ο οποίος θα μας επιτρέπει να υπολογίζουμε την μερική παράγωγο της συνάρτησης απώλειας ως προς οποιοδήποτε από τα βάρη του νευρωνικού δικτύου, ώστε μετέπειτα να χρησιμοποιήσουμε την παράγωγο αυτή για την ενημέρωση της τιμής του εν λόγω βάρους. Για την εκτέλεση της διαδικασίας υπολογισμού της βαθμίδας ο πιο ευρέως γνωστός αλγόριθμος είναι η οπισθοδιάδοση (backpropagation) ή προς τα πίσω διαπέραση (backwards pass), όπως είναι αλλιώς γνωστή. Για να χρησιμοποιήσουμε τον αλγόριθμο αυτό θα πρέπει η συνάρτηση απώλειας (loss function) να πληρεί δύο προϋποθέσεις:

1. Η συνάρτηση απώλειας θα πρέπει να μπορεί να γραφεί ως ο μέσος όρος των συναρτήσεων απώλειας για κάθε ένα από τα δείγματα που ανήκουν στο σύνολο δεδομένων εκπαίδευσης (π.χ. η τετραγωνική συνάρτηση απώλειας $C = \frac{1}{2n} \sum_x ||y(x) - a(x)||^2$ ικανονοποιεί αυτή την προϋπόθεση αφού μπορεί να γραφεί ως $C = \frac{1}{n} \sum_x C_x, C_x = \frac{1}{2}||y(x) - a(x)||^2$ όπου $y(x)$ η πραγματική έξοδος για το δείγμα x και $a(x)$ το διάνυσμα ενεργοποιήσεων του στρώματος εξόδου του νευρωνικού δικτύου).

2. Η συνάρτηση απώλειας θα πρέπει να μπορεί να γραφεί συναρτήσει της εξόδου του νευρωνικού δικτύου.

Για την εύρεση των μερικών παραγώγων ο αλγόριθμος της οπισθοδιάδοσης ξεκινά υπολογίζοντας τις μερικές παραγώγους της συνάρτησης απώλειας ως προς την έξοδο του νευρωνικού δικτύου και στην συνέχεια χρησιμοποιώντας τον κανόνα της αλυσίδας συνεχίζει προς τα κρυμμένα επίπεδα για να υπολογίσει τις μερικές παραγώγους ως προς κάθε ένα από τα βάρη του νευρωνικού δικτύου μέχρι να φτάσει στο στρώμα εισόδου.

Ανακεφαιλεώνοντας λοιπόν τα στάδια για την εκπαίδευση ενός νευρωνικού δικτύου είναι τα εξής :

1. Αρχικοποίηση των βαρών του νευρωνικού δικτύου.

2. Επιλογή του ρυθμού μάθησης a, ή δοκιμή διαφόρων τιμών για να βρούμε τη βέλτιστη όσο αφορά την ελαχιστοποίηση της συνάρτησης απώλειας.

3. Επιλογή μιας εκ των Batch Gradient Descent, Stochastic Gradient Descent, Mini-Batch Gradient Descent, Adam η οποιουδήποτε άλλου αλγορίθμου βελτιστοποίησης και επιλογή του batch size για να καθορίσουμε πόσα δείγματα θα χρησιμοποιούνται για τον υπολογισμό της βαθμίδας.

4. Εκτελούμε μια προς τα εμπρός διαπέραση του νευρωνικού δικτύου για τον υπολογισμό των εξόδων για το τρέχον batch/δείγμα/minibatch.

5. Υπολογίζουμε τη συνάρτηση απώλειας.

6. Υπολογίζουμε τις μερικές παραγώγους της συνάρτησης απώλειας ως προς τα βάρη του νευρωνικού δικτύο χρησιμοποιώντας τον αλγόριθμο της οπισθοδιάδοσης (backpropagation).

7. Ενημερώνουμε τα βάρη με βάση τον κανόνα:

$$w_{new} = w_{old} - a * \frac{\partial C}{\partial w_{old}}. \tag{5.1}$$

8. Όταν διαπεράσουμε όλα τα δείγματα του συνόλου δεδομένων τότε έχει ολοκληρωθεί μια εποχή (epoch) της διαδικασίας μάθησης. Αν έχουμε φτάσει σε ένα επιθημητό αριθμό εποχών ή επιθυμητή τιμή για τη συνάρτηση απώλειας τότε τερματίζουμε, αλλιώς επαναλαμβάνουμε τα βήματα 4-7.

6 Μέτρα αξιολόγησης της επίδοσης του μοντέλου

Τα τελευταία χρόνια, έχει κριθεί ιδιαίτερα σημαντική η ανάπτυξη μετρικών αξιολόγησης παραγωγικών μοντέλων "μαύρου κουτιού" (black-box generative models), μετρικών δηλαδή που δεν υποθέτουν τίποτα για τη φύση του παραγωγικού μοντέλου που αξιολογούν. Χρησιμοποιώντας τέτοιου είδους μετρικές, μας δίνεται η δυνατότητα να συγκρίνουμε την επίδοση παραγωγικών μοντέλων διαφορετικής φύσεως και να εξάγουμε συμπεράσματα σχετικά με την καταλληλότητα και την αποτελεσματικότητα των μοντέλων. [52]

6.1 Inseption Score (IS)

Το Inception Score είναι μια "μετρική" που χρησιμοποιείται ευρέως τα τελευταία χρόνια για την αξιολόγηση της απόδοσης διαφόρων παραγωγικών μοντέλων που αφορούν εικόνες. Η συγκεκριμένη "μετρική" προτάθηκε αρχικά από τους Salimans, Goodfellow, Zaremba, Cheung, Radford και Chen στην προσπάθεια τους να αξιολογήσουν διαφορετικά μοντέλα τύπου GANs.

Αρχικά, η αξιολόγηση βασιζόταν σε κριτικές από ανθρώπινους σχολιαστές, οι οποίοι καλούνταν να διακρίνουν αν μια εικόνα είναι πραγματική ή παραγόμενη από μοντέλα τεχνητής νοημοσύνης. [53] Για να καταλάβουμε όμως το πως ορίζεται το Inception Score, πρέπει αρχικά να παρουσιάσουμε ένα ευρέως γνωστό συνελικτικό νευρωνικό δίκτυο. Σημαντικό είναι επίσης να αναφέρουμε ότι παρακάτω παρουσιάζουμε το Inception Score όπως αρχικά αναπτύχθηκε. Σε περιπτώσεις όπου η χρήση του ImageNet που αναφέρουμε παρακάτω δεν είναι κατάλληλη μπορεί εναλλακτικά να χρησιμοποιειθεί κάποιο νευρωνικό δίκτυο του ίδιου τύπου αλλά προ εκπαιδευμένο σε κατάλληλα δεδομένα που να αντιστοιχούν στην μελέτη που γίνεται σε κάθε περίπτωση. Παρόλα αυτά για λόγους απλότητας, παρακάτω υποθέτουμε ότι η χρήση του συγκεκριμένου συνόλου δεδομένων και νευρωνικού δικτύου είναι κατάλληλη.

Το Inception v3 Network [54] είναι ένα βαθύ συνελικτικό νευρωνικό δίκτυο, σχεδιασμένο για προβλήματα ταξινόμησης στο ευρέως διαδεδομένο σύνολο δεδομένων ImageNet. Το ImageNet αποτελείται από 1.2 εκατομμύρια πολυχρωματικές εικόνες από 1000 κλάσεις. Το Inception v3 Network δεδομένης μιας εικόνας x, παράγει ένα διάνυσμα πιθανοτήτων $p(y|x) \in [0,1]^{1000}$ το οποίο αντιπροσωπεύει την πιθανότητα η εικόνα να ανήκει σε κάθε μια από τις 1000 κλάσεις.

Χρησιμοποιώντας λοιπόν ένα προ-εκπαιδευμένο νευρωνικό δίκτυο τύπου Inception v3 Network το Inception Score υπολογίζει ένα στατιστικό των παραγώμενων εικόνων του μοντέλου μας ως εξής :

$$IS(G) = exp(\mathbb{E}_{\mathbf{x}\sim p_g} D_{KL}(p(y|\mathbf{x})||p(y))) \tag{6.1}$$

όπου p_g η κατανομή που προκύπτει από την εκπαίδευση του παραγωγικού μας μοντέλου G, $p(y|\mathbf{x})$ η υπό συνθήκη κατανομή της κάθε κλάσης και $p(y) = \int_{\mathbf{x}} p(y|\mathbf{x}) p_g(\mathbf{x})$ η περιθώρια κατανομή των κλάσεων.

Σκοπός της ανάπτυξης του Inception Score ήταν η ανάπτυξη μιας "μετρικής" που θα αξιολογούσε τις εξής απαιτήσεις που υπάρχουν από ένα παραγωγικό

μοντέλο:

1. Οι εικόνες που παράγονται πρέπει να περιέχουν αντικείμενα που διακρίνονται ξεκάθαρα (Αποφυγή θολών εικόνων) ή αλλιώς θέλουμε η $p(y|\mathbf{x})$ να έχει χαμηλή εντροπία.

2. Το παραγωγικό μοντέλο θα πρέπει να παράγει εικόνες με μεγάλη ποικιλομοφία και από όλες τις διαφορετικές κλάσεις του ImageNet, ή αλλιώς η $p(y)$ να έχει υψηλή εντροπία.

Αν και οι δύο αυτές απαιτήσεις ικανοποιούνται, τότε αναμένουμε μια μεγάλη τιμή για την απόκλιση KL μεταξύ των κατανομών $p(y)$ και $p(y|\mathbf{x})$ και άρα μεγάλη τιμή για το Inception Score. Συχνά, λόγω του ότι η περιθώρια κατανομή των κλάσεων είναι άγνωστη, χρησιμοπιείται η εξής εκτίμηση για το Inception Score:

$$IS(G) \approx exp(\frac{1}{N}\sum_{i=1}^{N} D_{KL}(p(y|x^{(i)})||\hat{p}(y)) \tag{6.2}$$

όπου $\hat{p}(y) = \frac{1}{N}\sum_{i=1}^{N}(p(y|x^{(i)})$.

Όπως αναφέραμε παραπάνω, η επιλογή του νευρωνικού δικτύου που χρησιμοποιείται για την αξιολόγηση του παραγωγικού μας μοντέλου σχετίζεται με το σύνολο δεδομένων στο οποίο εργαζόμαστε, και πόσο αυτό το σύνολο έχει κοινά χαρακτηριστικά με το ImageNet. Σε σύνολα δεδομένων, όπως για παράδειγμα, ακτινογραφίες θώρακα που θα δούμε παρακάτω συχνά επιλέγονται νευρωνικά δίκτυα εκπαιδευμένα για την ταξινόμηση αντίστοιχων δεδομένων από ακτινογραφίες. Παρόλα αυτά, η επιλογή αυτή έχει μελετηθεί εκτενώς, με πολλές έρευνες να αναφέρουν ότι η χρήση του νευρωνικού δικτύου, εκπαιδευμένου στο ImageNet για τον υπολογισμό των μετρικών επέφερε αποτελέσματα που συμφωνούσαν σε μεγαλύτερο βαθμό με την ανθρώπινη αξιολόγηση των μοντέλων [55], για αυτό το λόγο στη δική μας περίπτωση θα χρησιμοποιήσουμε το IS όπως αρχικά προτάθηκε, ανεξαρτήτως του dataset στο οποίο εργαζόμαστε.

6.2 Frechet Inception Distance (FID)

Η "μετρική" Frechet Inception Distance (FID) αναπτύχθηκε (Heusel, Ramsauer, Unterthiner & Nessler) [56] αφορμώμενη από το Inception Score που αναφέραμε παραπάνω σε μια προσπάθεια η αξιολόγηση ενός παραγωγικού μοντέλου να λαμβάνει υπόψιν και την κατανομή του συνόλου των δεδομένων εκπαίδευσης και όχι μόνο την κατανομή των παραγώμενων δειγμάτων. Πρωτού όμως αναλύσουμε το πως υπολογίζεται το FID, θα παρουσιάσουμε μια μετρική που ονομάζεται απόσταση Frechet (Frechet Distance) [57], η οποία είναι επίσης γνωστή και ως απόσταση Wasserstein-2 (Wasserstein-2 distance) [58] όταν μιλάμε συγκεκριμένα για σύγκριση κατανομών πιθανότητας .

6.2.1 Απόσταση Frechet

Ορισμός 6.1. *(Απόσταση Frechet)*
Έστω (M,d) ένας χώρος μέτρου και μ, ν δύο μέτρα πιθανότητας στον M με πεπερασμένες 2 ροπές. Τότε η απόσταση Frechet μεταξύ των δύο μέτρων πιθανότητας ορίζεται ως εξής:

$$d^2(\mu,\nu) = inf_{\gamma\in\Gamma(\mu,\nu)}\mathbb{E}_{(X,Y)} \sim \gamma[d(X,Y)^2] \tag{6.3}$$

Όπου $\Gamma(\mu,\nu)$ το σύνολο όλων των δυνατών συζεύξεων των μέτρων πιθανότητας μ και ν.

Ορισμός 6.2. *(Σύζευξη μέτρων πιθανότητας)*
Η σύζευξη (coupling) δύο μέτρων πιθανότητας μ και ν στον ίδιο μετρήσιμο χώρο $(M,\mathcal{M})$ όπου $\mathcal{M}$ μια σ-άλγεβρα υποσυνόλων του M, είναι οποιοδήποτε μέτρο πιθανότητας $\mathbb{P}$ στον μετρήσιμο χώρο γινόμενο $(M\times M, \mathcal{M}\otimes\mathcal{M})$ (όπου $\mathcal{M}\otimes\mathcal{M}$ η μικρότερη σ-αλγεβρα που περιέχει το $M\times M$) με περιθώριες κατανομές τις μ και ν [59], δηλαδή :

$$\mu = \mathbb{P}\circ\pi \quad \nu = \mathbb{P}\circ\pi^{'} \tag{6.4}$$

όπου

$$\pi(x,x^{'}) = x, \quad \pi^{'}(x,x^{'}) = x^{'} \quad (x,x^{'})\in M\times M \tag{6.5}$$

Στο [60] οι Dowson και Landau, παραθέτουν τον τύπο υπολογισμού της απόστασης Frechet για πολυδιάστατες κανονικές κατανομές, τον οποίο και θα χρειαστούμε στη συνέχεια, στον ορισμό του FID. Η απόσταση Frechet, λοιπόν για δύο πολυδιάστατες κανονικές κατανομές μπορεί να αποδειχθεί ότι ορίζεται ως εξής:

$$d^2 = |\mu_X - \mu_Y|^2 + tr(\Sigma_X + \Sigma_Y - 2(\Sigma_X\Sigma_Y)^{1/2}) \tag{6.6}$$

όπου μ_X, μ_Y τα διανύσματα μέσης τιμής και Σ_X, Σ_Y οι πίνακες συνδιακύμανσης των τ.μ. X και Y αντίστοιχα.

6.2.2 Ορισμός του Frechet Inception Score (FID)

Για τον υπολογισμό του Frechet Inception Distance, όπως και στο Inception Score, γίνεται χρήση ενός Inception v3 Network που έχει εκπαιδευτεί για ταξινόμηση εικόνων από το σύνολο δεδομένων ImageNet. Στην περίπτωση αυτή όμως το μοντέλο αυτό εφαρμόζεται τόσο σε συνθετικές εικόνες, παραγώμενες από το εκάστοτε μοντέλο, όσο και σε πραγματικές εικόνες από το σύνολο δεδομένων που χρησιμοποιήθηκε για την εκπαίδευση του μοντέλου.

Συγκεκριμένα, χρησιμοποιείται το "coding layer" του δικτύου Inception v3, το οποίο είναι το τελευταίο επίπεδο συγκέντρωσης πριν το στρώμα εξόδου και θεωρείτε ότι σε αυτό το επίπεδο το δίκτυο ποσοτικοποιεί διάφορα χαρακτηριστικά της εικόνας εισόδου. Χρησιμοποιώντας λοιπόν τις ενεργοποιήσεις (activations) στο συγκεκριμένο επίπεδο για μεγάλο αριθμό τόσο πραγματικών όσο και συνθετικών εικόνων, εκτιμάται η μέση τιμή και ο πίνακας συνδιασποράς των ενεργοποιήσεων για τα πραγματικά δεδομένα (m, C) και για τα συνθετικά δεδομένα (m_g, C_g) και υπολογίζεται η απόσταση μεταξύ των δύο κατανομών

χρησιμοποιώντας την απόσταση Frechet για πολυδιάστατες κανονικές κατανομές που δώσαμε στην εξίσωση (7.6). Υποθέσαμε λοιπόν, ότι οι ενεργοποιήσεις του coding layer ακολουθούν πολυδιάστατη κανονική κατανομή με αντίστοιχη μέση τιμή και πίνακα συνδιασποράς για κάθε μια από τις δύο περιπτώσεις.

Λόγω του ότι το FID, μετρά την απόσταση μεταξύ των δύο κατανομών, μικρότερες τιμές υποδεικνύουν καλύτερης ποιότητας παραγώμενες εικόνες.

6.3 Negative Log-Likelihood (NLL)

Η αρνητική λογαριθμοποιημένη πιθανοφάνεια (Negative Log Likelihood - NLL) είναι μια μετρική απώλειας που χρησιμοποιείται για να ποσοτικοποιήσει πόσο καλά ένα πιθανοθεωρητικό μοντέλο εξηγεί τα παρατηρούμενα δεδομένα. Σε πολλά μοντέλα μηχανικής μάθησης το κριτήριο εκπαίδευσης είναι η ελαχιστοποίηση της αρνητικής λογαριθμοποιημένης πιθανοφάνειας, η οποία είναι ισοδύναμη με την μεγιστοποίηση της λογαριθμοποιημένης πιθανοφάνειας του μοντέλου πάνω στα δεδομένα. Ο λόγος που χρησιμοποιείται η αρνητική λογαριθμοποιημένη πιθανοφάνεια είναι επειδή οι περισσότεροι αλγόριθμοι βελτιστοίησης είναι σχεδιασμένοι να ελαχιστοποιούν την απώλεια.

Διαισθητικά η NLL μας δείχνει πόσο "έκπληκτο" είναι το μοντέλο μας από τα παρατηρούμενα δεδομένα. Δηλαδή, μικρές τιμές της NLL, δείχνουν ότι το μοντέλο δίνει μεγάλη πιθανότητα στα παρατηρούμενα δεδομένα και άρα έχει επιτύχει καλή προσαρμογή. Συνοψίζοντας, η NLL είναι μια μετρική της απόδοσης ενός μοντέλου, βασιζόμενη στην πιθανότητα τα παρατηρούμενα δεδομένα να προβλεφθούν από το μοντέλο.

Ειδικότερα, στα παραγωγικά μοντέλα, όπου στόχος είναι η προσέγγιση της πραγματικής κατανομής τω δεδομένων, η αρνητική λογαριθμοποιημένη πιθανοφάνεια είναι μια ποσότητα που αναδεικνύει σε ποιο βαθμό η κατανομή που παράγει ο μοντέλο προσεγγίζει την πραγματική κατανομή. Για αυτό συχνά η NLL χρησιμοποιείται και για την εκπαίδευση των μοντέλων αυτών.

Στην περίπτωση των DDPMs, όπου δεν μπορούμε να υπολογίσουμε την NLL αναλυτικά, χρησιμοποιούμε ως προσέγγιση την ELBO (Evidence Lower Bound), η οποία –όπως αναφέρθηκε– χρησιμοποιείται τόσο για εκπαίδευση όσο και για αξιολόγηση του μοντέλου. Συχνά, για σκοπούς σύγκρισης, η NLL εκφράζεται σε bits ανά διάσταση (bits/dim), μετατρέποντας την τιμή της σε πιο ερμηνεύσιμη μονάδα μέτρησης. Η μετατροπή αυτή δίνεται από την εξίσωση:

$$NLL(bits/dim) = \frac{1}{D * log2} NLL \tag{6.7}$$

όπου $D = pixels * channels$, το D δηλαδή αντιπροσωπεύει το συνολικό αριθμό των pixel που υπάρχουν σε κάθε εικόνα.

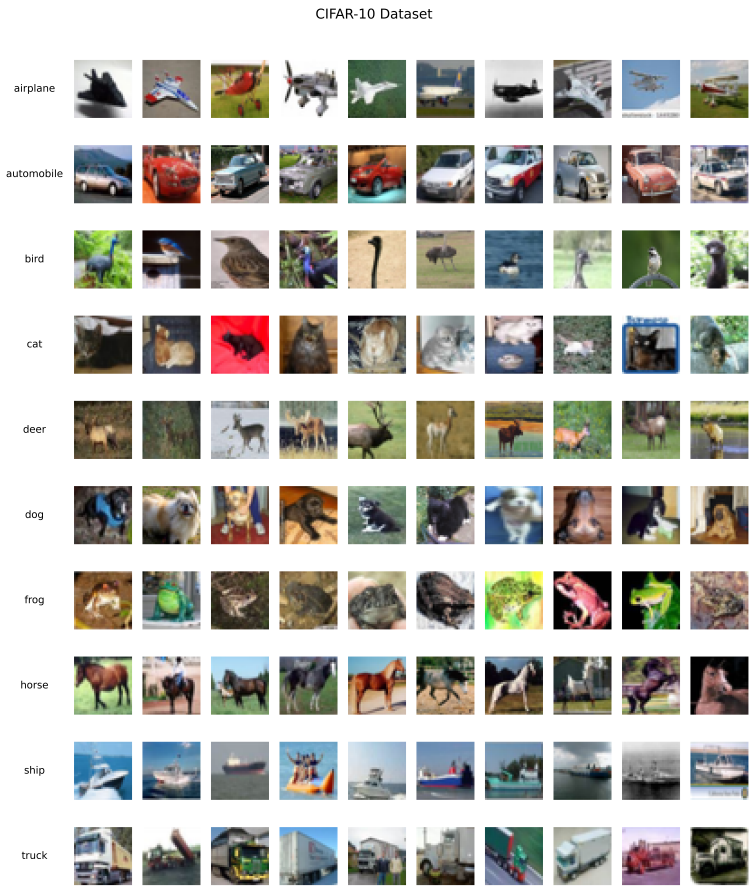
7 Σύνολα Δεδομένων και Προεπεξεργασία

Στην ενότητα αυτή θα μελετήσουμε τα διαφορετικά σύνολα δεδομένων (datasets) στα οποία εφαρμόζουμε τα παραγωγικά μας μοντέλα καθώς και την επεξεργασία που εφαρμόσαμε στα δείγματα του κάθε συνόλου πρωτού χρησιμοποιηθούν για την εκπαίδευση των μοντέλων.

7.1 CIFAR-10

7.1.1 Περιγραφή Συνόλου

Το σύνολο δεδομένων CIFAR-10 είναι ένα από τα πιο ευρέως διαδεδομένα σύνολα δεδομένων για το κομμάτι της ακαδημαϊκής κοινότητας που ασχολείται με την τεχνητή νοημοσύνη, καθώς χρησιμοποιείται πολύ συχνά σε προβλήματα μηχανικής μάθησης, είτε για την αξιολόγηση ενός ταξινομητή (classifier) είτε για την εκπαίδευση παραγωγικών μοντέλων, όπως και στη δική μας περίπτωση. Αποτελείται από 60 000 έγχρωμες εικόνες μεγέθους 32*32 pixel, από τις οποίες οι 50 000 χρησιμοποιούνται για εκπαίδευση και οι 10 000 για επικύρωση/αξιολόγηση των αλγορίθμων. Το CIFAR-10 είναι διαθέσιμο μέσω της βιβλιοθήκης torchvision.datasets της PyTorch.



Σχήμα 7.1: Σύνολο Δεδομένων CIFAR-10

Οι εικόνες που ανήκουν στο CIFAR-10 προέρχονται από 10 διαφορετικές κατηγορίες (όπως φαίνεται και στο σχήμα 7.1) : αεροπλάνα, αυτοκίνητα, πουλιά, γάτες, ελάφια, σκύλοι, βάτραχοι, άλογα, πλοία και φορτηγά.

7.1.2 Προεπεξεργασία

Οι εικόνες του CIFAR-10 προτού χρησιμοποιηθούν για την εκπαίδευση των μοντέλων, επιδέχονται την ακόλουθη προεπεξεργασία:

- Μετατροπή των εικόνων σε διανύσματα (tensors) της μορφής (B,C,H,W) (Batch, Κανάλια, Ύψος, Πλάτος)
- Κανονικοποίηση των τιμών των pixel από τιμές στο [0,1] σε τιμές στο [-1,1] μέσω ενός γραμμικού μετασχηματισμού : $pixel_{new} = 2 * pixel_{old} - 1$
- Αναστρέφουμε τυχαία κάποιες από τις εικόνες, ώστε να υπάρχει πιο μεγάλη ποικιλομορφία στα δεδομένα.

7.2 ChestX-ray8

7.2.1 Περιγραφή

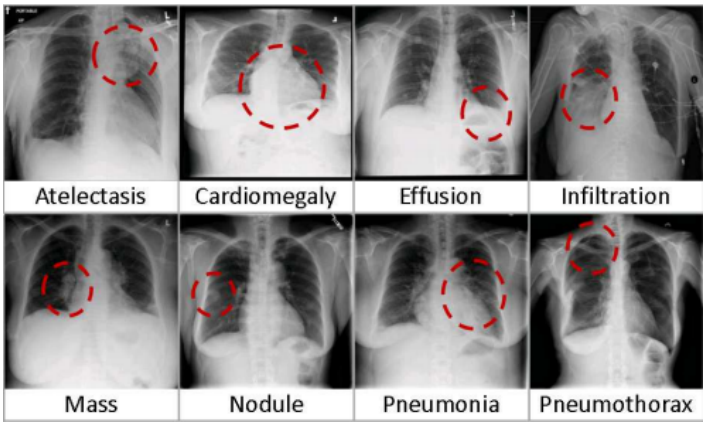
Το ChestX-ray8 είναι ένα δημόσιο σύνολο δεδομένων που περιέχει ακτινογραφίες θώρακα. Δημοσιεύτηκε από το Nataional Institutes of Health (NIH) το 2017 [61] . Περιλαμβάνει 112 120 εικόνες X-ray από 30 805 ασθενείς, οι οποίες έχουν μέγεθος 1024*1024 και είναι μονοχρωματικές. Κάθε μια από τις εικόνες συνοδεύεται από μια ετικέτα που αφορά την παθολογία που απεικονίζεται στην συγκεκριμένη ακτινογραφία. Στο σύνολο δεδομένων απεικονίζονται συνολικά 14 παθολογίες (Atelectasis, Consolidation, Infiltration, Pneumothorax, Edema, Emphysema, Fibrosis, Effusion, Pneumonia, Pleural thickening, Cardiomegaly, Nodule Mass, Hernia)

Το ChestX-ray8 έχει ευρεία χρήση σε προβλήματα ταξινόμησης, έχει όμως χρησιμοποιηθεί και για εκπαίδευση παραγωγικών μοντέλων [62]. Γενικότερα, χάρη στο μεγάλο του μέγεθος και τη δημόσια διαθεσιμότητά του, έχει χρησιμοποιηθεί εκτενώς για αλγορίθμους βαθειάς μάθησης σε ιατρικές εικόνες θώρακα, ειδικά σε περιπτώσεις όπου απαιτείται αντιμετώπιση της ανισορροπίας μεταξύ κλάσεων.

7.2.2 Προεπεξεργασία

Οι εικόνες του ChestX-ray8 προτού χρησιμοποιηθούν για την εκπαίδευση των μοντέλων, επιδέχονται την ακόλουθη προεπεξεργασία:

- Μετατροπή των εικόνων σε διανύσματα (tensors) της μορφής (B,C,H,W) (Batch, Κανάλια, Ύψος, Πλάτος).
- Σμίκρυνση εικόνων ώστε να έχουν μέγεθος 64*64 αντί για 1024*1024 (Για λόγους υπολογιστικών πόρων).



Σχήμα 7.2: Σύνολο Δεδομένων ChestX-ray8. Πηγή: Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., & Summers, R. M. (2017). Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2097-2106).

- Κανονικοποίηση των τιμών των pixel από τιμές στο [0,1] σε τιμές στο [-1,1] μέσω ενός γραμμικού μετασχηματισμού : $pixel_{new} = 2 * pixel_{old} - 1$
- Αναστρέφουμε τυχαία κάποιες από τις εικόνες, ώστε να υπάρχει πιο μεγάλη ποικιλομορφία στα δεδομένα.

7.3 CelebA

7.3.1 Περιγραφή

Το σύνολο δεδομένων CelebA (CelebFaces Attributes) είναι ένα δημόσιο σύνολο δεδομένων που δημοσιεύτηκε από το Chinese University of Hong Kong το 2015 [63]. Περιέχει συνολικά 202.599 έγχρωμες φωτογραφίες προσώπων από 10.177 διαφορετικά άτομα, οι οποίες παρουσιάζουν μεγάλη ποικιλομορφία ως προς τη στάση, την έκφραση, το υπόβαθρο και το φωτισμό. Οι εικόνες έχουν ανάλυση 178×218 pixels και συνοδεύονται από 40 δυαδικά χαρακτηριστικά (binary attributes) για κάθε πρόσωπο, όπως «φοράει γυαλιά», «χαμογελάει», «έχει γένια», κ.ά. Το CelebA χρησιμοποιείται ευρέως σε εφαρμογές αναγνώρισης προσώπου, επεξεργασίας εικόνας και εκμάθησης χαρακτηριστικών προσώπου.

7.3.2 Προεπεξεργασία

Οι εικόνες του CelebA προτού χρησιμοποιηθούν για την εκπαίδευση των μοντέλων, επιδέχονται την ακόλουθη προεπεξεργασία:

- Μετατροπή των εικόνων σε διανύσματα (tensors) της μορφής (B,C,H,W) (Batch, Κανάλια, Ύψος, Πλάτος).
- Μετατροπή των εικόνων σε τετράγωνες 178*178 αποκόπτοντας το κάτω μέρος της εικόνας που δεν αντιστοιχεί στο πρόσωπο.



Σχήμα 7.3: Σύνολο Δεδομένων CelebA. Πηγή: https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html

- Σμίκρυνση εικόνων ώστε να έχουν μέγεθος 64*64 αντί για 178*178 (Για λόγους υπολογιστικών πόρων).
- Κανονικοποίηση των τιμών των pixel από τιμές στο [0,1] σε τιμές στο [-1,1] μέσω ενός γραμμικού μετασχηματισμού : $pixel_{new} = 2 * pixel_{old} - 1$
- Αναστρέφουμε τυχαία κάποιες από τις εικόνες, ώστε να υπάρχει πιο μεγάλη ποικιλομορφία στα δεδομένα.

8 Πειραματική Διαδικασία και Αποτελέσματα

Στην ενότητα αυτή θα περιγράψουμε τις λεπτομέριες σχετικά με τα πειράματα που εκτελέσαμε. Ο κώδικας για την εκτέλεση των πειραμάτων βασίστηκε σε μεγάλο βαθμό στον δημόσια διαθέσιμο κώδικα της OpenAI (https://github.com/openai/improved-diffusion). Παρακάτω περιγράφουμε συνοπτικά τις παραμέτρους που χρησιμοποιήσαμε για κάθε ένα από τα πειράματα μας.

Όπως φαίνεται και στον πίνακα, οι παράμετροι αλλάζουν, ανά σύνολο δεδομένων. Στις εικόνες μικρότερου μεγέθους του συνόλου CIFAR-10 εκπαιδεύουμε για περισσότερα βήματα καθώς το συγκεκριμένο σύνολο περιέχει εικόνες που διαφέρουν σε μεγάλο βαθμό μεταξύ τους, ενώ για τα σύνολα ChestX-ray8 και CelebA , όλα τα δείγματα έχουν κοινά χαρακτηριστικά (π.χ. χαρακτηριστικά προσώπου ή θώρακα) και επομένως δεν χρειάζεται τόσο μεγάλος αριθμός βημάτων. Και στα τρια σύνολα δεδομένων χρησιμοποιήσαμε γραμμικό πρόγραμμα διακύμανσης, 2 μπλοκ υπολοίπου ανά επίπεδο και 4 U-Net επίπεδα. Στις μικρότερου μεγέθους εικόνες χρησιμοποιήσαμε batch size 128 και συντελεστή μάθησης 2e-4, ενώ για τις εικόνες μεγαλύτερης ανάλυσης, οι συγγραφείς στο [2] αναφέρουν ότι η εκπαίδευση με συντελεστή μάθησης 2e-4 είναι ασταθής, οπότε χρησιμοποιούμε 1e-4 και batch size 64. Σε όλα τα μοντέλα χρησιμοποιήσαμε τον βελτιστοποιητή Adam που περιγράφεται στην ενότητα 5.1.3, εκθετικό κινητό μέσο όρο στα βάρη με συντελεστή απομείωσης (decay factor) ίσο με 0,9999 και dropout rate ίσο με 0.1.

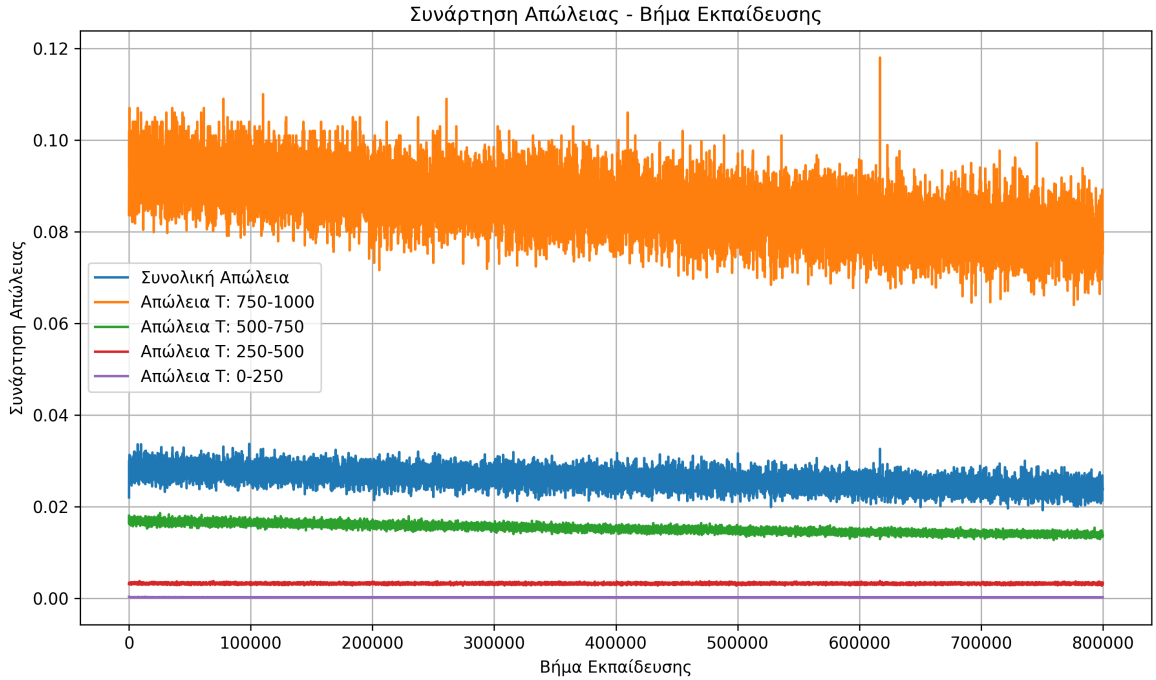
Παράμετρος	**CIFAR-10**	**ChestX-ray8**	**CelebA**
Μέγεθος εικόνας	32×32	64×64	64×64
Batch size	128	64	64
Αριθμός Βημάτων	800 000	500 000	500 000
Βελτιστοποιητής	Adam	Adam	Adam
Συντελεστής Μάθησης	2e-4	1e-4	1e-4
Μπλοκ υπολοίπου ανά επίπεδο	2	2	2
Επίπεδα U-Net (down/up sampling)	4	4	4
Συντελεστής απομείωσης	0,999	0,999	0,999
Κανάλια ανά επίπεδο	(128,256,256,256)	(128,256,384,512)	(128,256,384,512)
Πρόγραμμα διακύμανσης	Γραμμικό $0.0001 \to 0.02$	Γραμμικό $0.0001 \to 0.02$	Γραμμικό $0.0001 \to 0.02$
Dropout rate	0.1	0.1	0.1

Πίνακας 1: Παράμετροι για κάθε σύνολο δεδομένων

8.1 Αποτελέσματα CIFAR-10

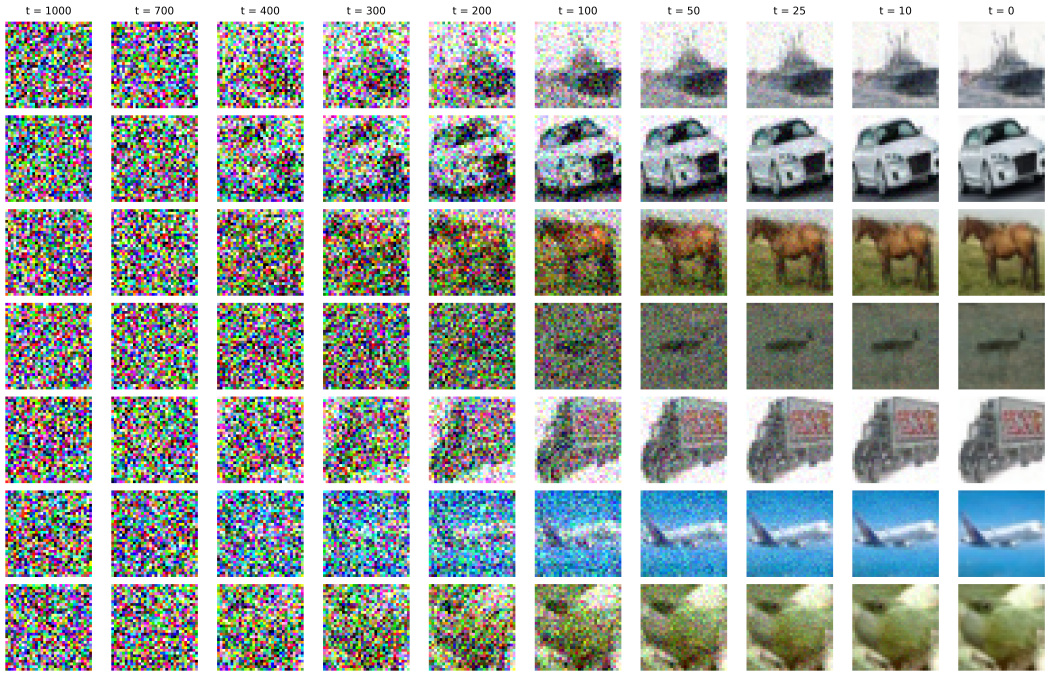
Όπως αναφέραμε και παραπάνω η εκπαίδευση του μοντέλου στο σύνολο δεδομένων CIFAR-10 έγινε για 800 000 βήματα. Για την εκπαίδευση χρησιμοποιήθηκε η NVIDIA Tesla V100 με ρυθμό εκπαίδευσης 1.85 βήματα/δευτερόλεπτο και συνολικό χρόνο εκπαίδευσης 120 ώρες.

Στο σχήμα 8.1 παρατηρούμε το διάγραμμα της τιμής της συνάρτησης απώλειας σε κάθε βήμα της εκπαίδευσης καθώς και την τιμή της συνάρτησης απώλειας που αφορά δείγματα που αντιστοιχούν σε χρονικές στιγμές στα διαστήματα 0-250, 250-500, 500-750, 750-1000. Η συνολική απώλεια, μπορεί να θεωρηθεί ως ο σταθμισμένος μέσος όρος των τεσσάρων τιμών που αντιστοιχούν σε κάθε διάστημα χρονικών στιγμών. Όπως είναι αναμενόμενο, το μεγαλύτερο ποσοστό της απώλειας προέρχεται από χρονικές στιγμές στο διάστημα 750-1000, όπου τα στιγμιότυπα τείνουν προς τον γκαουσιανό θόρυβο και έχουν χαθεί σε μεγάλο βαθμό τα χαρακτηριστικά της αρχικής εικόνας. Σε αυτά τα στιγμιότυπα είναι πιο δύσκολο να εκτιμηθεί ο ακριβής θόρυβος που έχει προστεθεί στην αρχική εικόνα. Αντίθετα σε νωρίτερα στάδια τις προς τα εμπρός διαδικασίας, όπου διατηρούνται ακόμα κάποια από τα χαρακτηριστικά των αρχικών εικόνων, παρατηρούμε μικρότερη απόκλιση μεταξύ του πραγματικού και του εκτιμώμενου θορύβου.

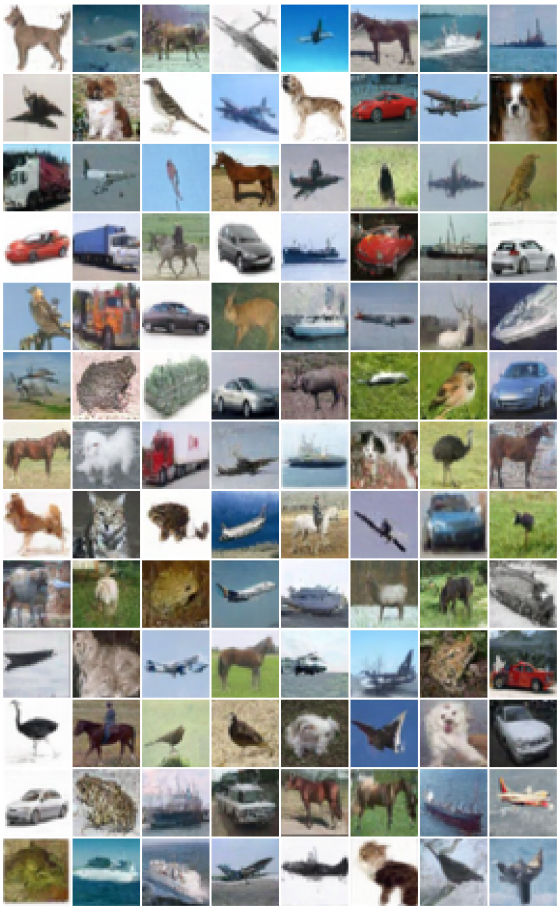


Σχήμα 8.1: Διάγραμμα συνάρτησης απώλειας - βήματος εκπαίδευσης για το σύνολο δεδομένων CIFAR-10

Στα σχήματα 8.2 και 8.3 φαίνονται κάποια από τα δείγματα που παρήγαγε το μοντέλο καθώς και κάποια παραδείγματα της παραγωγής δειγμάτων με σταδιακή αποθορύβωση. (παρουσιάζονται οι χρονικές στιγμές T =100, 700, 400, 300, 200, 100, 50, 25, 10, 0)



Σχήμα 8.2: Σταδιακή αποθορύβωση για παραγωγή δειγμάτων CIFAR-10



Σχήμα 8.3: Δείγματα CIFAR-10

8.2 Αποτελέσματα ChestX-ray8

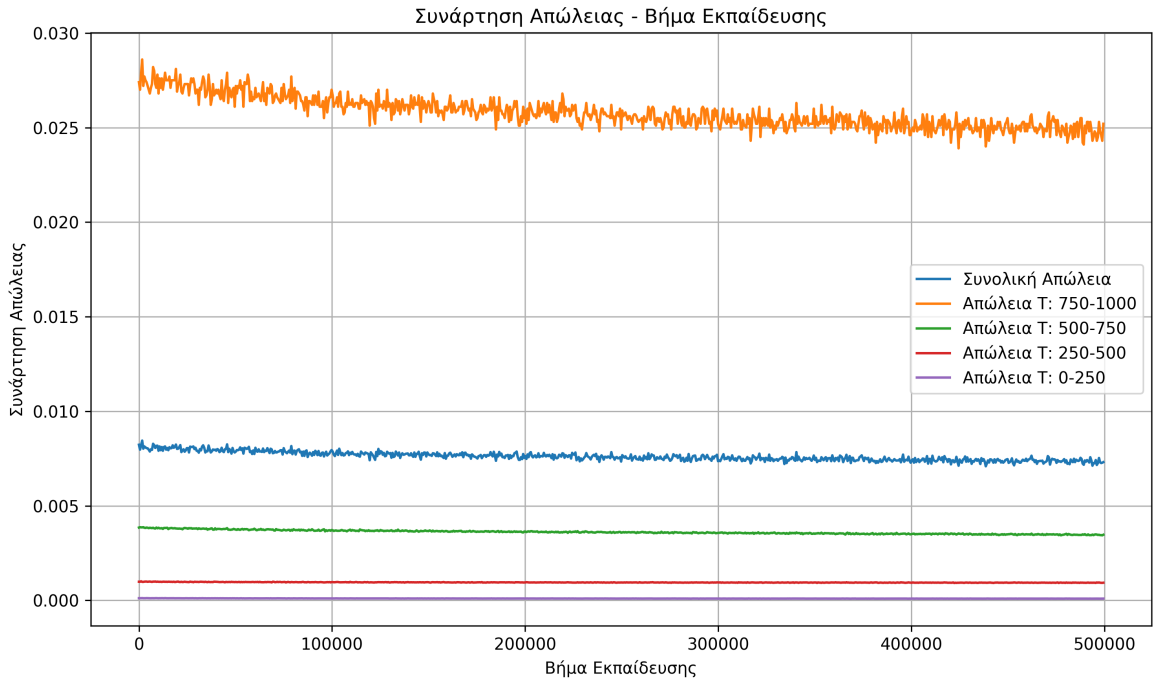
Η εκπαίδευση του μοντέλου στο σύνολο δεδομένων ChestX-ray8 έγινε για 500 000 βήματα λόγω της μικρότερης ποικιλομορφίας που παρατηρείται στο συγκεκριμένο σύνολο δεδομένων σε σχέση με το CIFAR-10. Η εκπαίδευση έγινε χρησιμοποιώντας GPU NVIDIA Tesla V100, με ρυθμό 1.38 βημάτων/δευτερόλεπτο και συνολική διάρκεια περίπου 100 ώρες.

Στο Σχήμα 8.4 απεικονίζεται η τιμή της συνάρτησης απώλειας ανά βήμα εκπαίδευσης για το μοντέλο που εκπαιδεύτηκε στο ChestX-ray8. Η παρατήρηση που έγινε για το αντίστοιχο διάγραμμα του μοντέλου στο σύνολο δεδομένων CIFAR-10 ισχύει και σε αυτή την περίπτωση, δηλαδή ότι το μεγαλύτερο ποσοστό της απώλειας οφείλεται σε χρονικές στιγμές που ανήκουν στο διάστημα 750-1000.

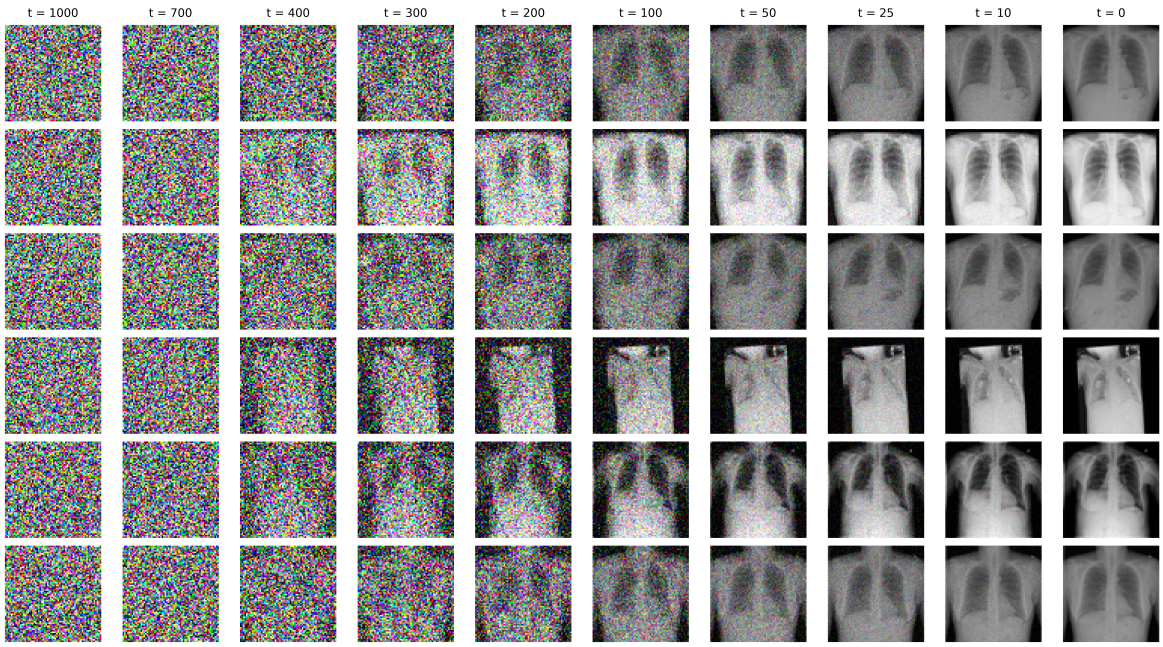
Ωστόσο, η απώλεια παρουσιάζει πιο ομαλή συμπεριφορά, με λιγότερες διακυμάνσεις σε σχέση με το αντίστοιχο διάγραμμα του Σχήματος 8.1 (CIFAR-10). Το γεγονός αυτό οφείλεται στα δεδομένα που απαρτίζουν το σύνολο δεδομένων ChestX-ray8, τα οποία, σε αντίθεση με αυτά του συνόλου CIFAR-10, είναι πιο ομοιογενή, επιδεικνυούν δηλαδή αρκετά κοινά στοιχεία μεταξύ τους. Ως αποτέλεσμα, η εκπαίδευση στο ChestX-ray8 είναι πιο σταθερή και η τιμή της συνάρτησης απώλειας για οποιοδήποτε υποσύνολο του συνόλου δεδομένων είναι αρκετά παραπλήσια. Αντίθετα, στο CIFAR-10 παρατηρούνται μεγαλύτερες διακυμάνσεις στη συνάρτηση απώλειας, πιθανώς επειδή το μοντέλο μαθαίνει με διαφορετικούς ρυθμούς για κάθε κατηγορία, ανάλογα με το εκάστοτε batch.

Τέλος, αξίζει να σημειωθεί ότι η συνάρτηση απώλειας για το μοντέλο του ChestX-ray8 κυμαίνεται σε αρκετά χαμηλότερα επίπεδα (κάτω από 0.009), σε σύγκριση με την απώλεια περίπου 0.02 του μοντέλου στο CIFAR-10, παρά τη μεγαλύτερη ανάλυση των εικόνων. Αυτό πιθανόν να οφείλεται στην αυξημένη χωρητικότητα του δικτύου, καθώς στο μοντέλο για το ChestX-ray8 χρησιμοποιήθηκαν περισσότερα κανάλια, παρέχοντας έτσι μεγαλύτερη εκφραστική ισχύ στο δίκτυο για να χειριστεί τα πιο υψηλής ανάλυσης δεδομένα του ChestX-ray8.

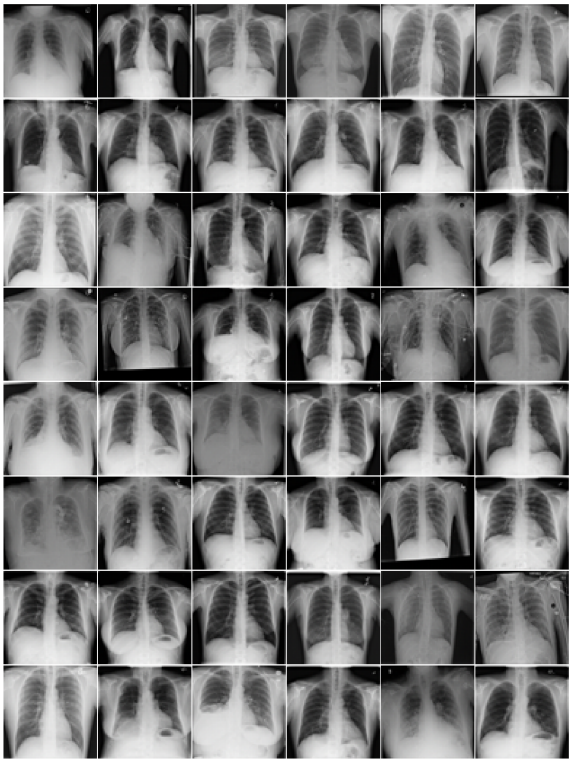
Στα σχήματα 8.5 και 8.6 φαίνονται κάποια από τα δείγματα που παρήγαγε το μοντέλο καθώς και κάποια παραδείγματα της παραγωγής δειγμάτων με σταδιακή αποθορύβωση. (παρουσιάζονται οι χρονικές στιγμές T = 100, 700, 400, 300, 200, 100, 50, 25, 10, 0)



Σχήμα 8.4: Διάγραμμα συνάρτησης απώλειας - βήματος εκπαίδευσης για το σύνολο δεδομένων ChestX-ray8



Σχήμα 8.5: Σταδιακή αποθορύβωση για παραγωγή δειγμάτων ChestX-ray8



Σχήμα 8.6: Δείγματα ChestX-ray8

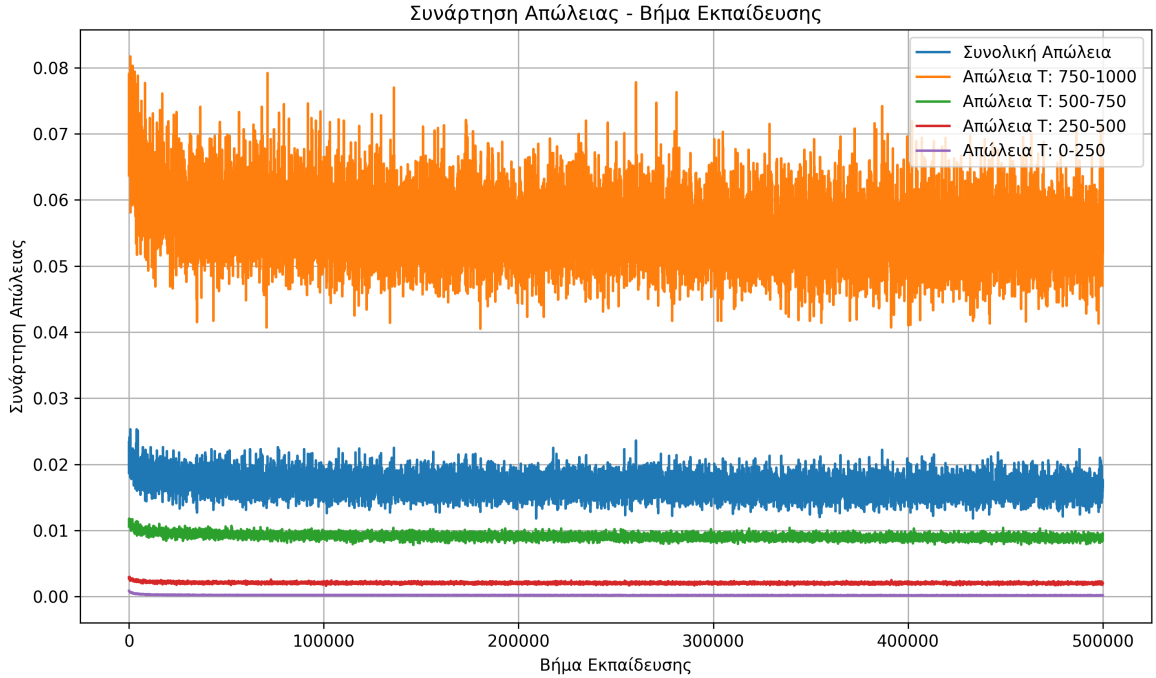
8.3 Αποτελέσματα CelebA

Η εκπαίδευση του μοντέλου στο σύνολο δεδομένων CelebA έγινε για 500 000 βήματα, όπως ακριβώς και στην περίπτωση του συνόλου ChestX-ray8. Η εκπαίδευση έγινε χρησιμοποιώντας GPU NVIDIA Tesla V100, με ρυθμό 1.38 βημάτων/δευτερόλεπτο και συνολική διάρκεια περίπου 100 ώρες.

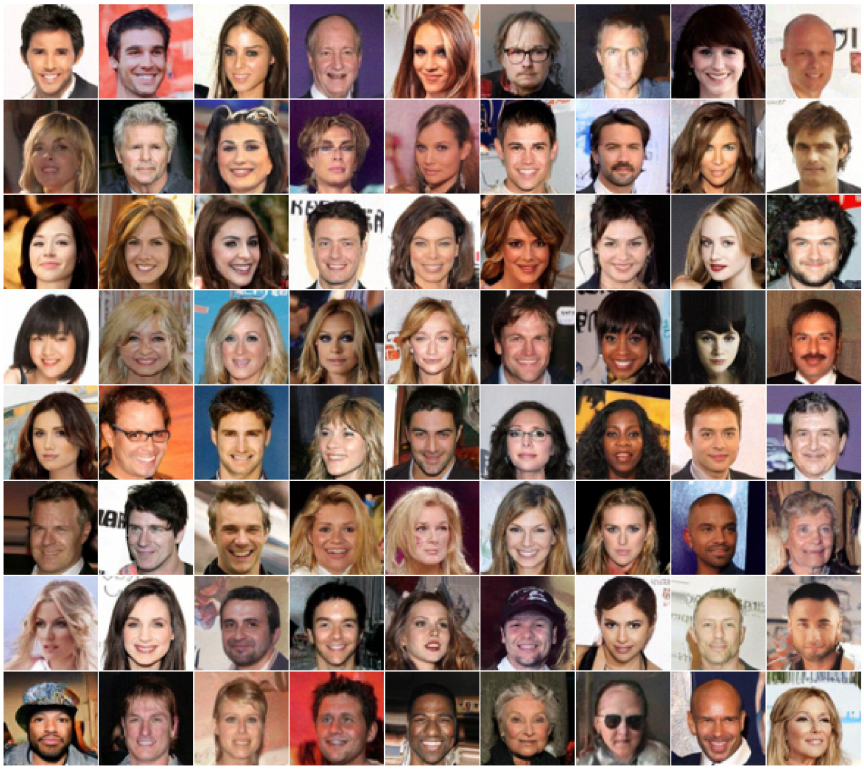
Στο Σχήμα 8.7 απεικονίζεται η τιμή της συνάρτησης απώλειας ανά βήμα εκπαίδευσης για το μοντέλο που εκπαιδεύτηκε στο CelebA. Το διάγραμμα αυτό επιβεβαιώνει τις παρατηρήσεις που κάναμε για τη σύγκριση των διαγραμμάτων των σχημάτων 8.1 και 8.4, καθώς το CelebA, όντας ένα σύνολο με ασφαλώς πιο ποικιλόμορφα δείγματα σε σχέση με το ChestX-ray8 για αυτό παρατηρούμε και πιο μεγάλες αυξομειώσεις στη συνάρτηση κόστους, ενώ επίσης έχουμε πιο μεγάλες αυξομειώσεις και σε σχέση με το διάγραμμα του CIFAR-10, πιθανών λόγο της μεγαλύτερης διάστασης των δεδομένων, αλλά και του μικρότερου batch size.

Επίσης, η συνάρτηση απώλειας και για το μοντέλο του CelebA κυμαίνεται σε χαμηλότερα επίπεδα, σε σύγκριση με την απώλεια του μοντέλου στο CIFAR-10 αλλά σε λίγο πιο ψηλά επίπεδα σε σχέση με το μοντέλο του ChestX-ray8, λόγω της μεγαλύτερης ποικιλομορφίας στα δεδομένα.

Στα σχήματα 8.5 και 8.6 φαίνονται κάποια από τα δείγματα που παρήγαγε το μοντέλο καθώς και κάποια παραδείγματα της παραγωγής δειγμάτων με σταδιακή αποθορύβωση. (παρουσιάζονται οι χρονικές στιγμές T = 100, 700, 400, 300, 200, 100, 50, 25, 10, 0)



Σχήμα 8.7: Διάγραμμα συνάρτησης απώλειας - βήματος εκπαίδευσης για το σύνολο δεδομένων CelebA



Σχήμα 8.8: Δείγματα CelebA



Σχήμα 8.9: Σταδιακή αποθορύβωση για παραγωγή δειγμάτων CelebA

8.4 Μετρικές

Αφού παρατηρήσαμε τόσο την εξέλιξη της τιμής της συνάρτησης απώλειας και οπτικοποιήσαμε κάποια δείγματα από κάθε σύνολο δεδομένων, παρατηρήσαμε αρκετά ικανοποιητικά αποτελέσματα σε όλες τις περιπτώσεις. Στο σημείο αυτό, υπολογίζουμε τις τιμές των μετρικών IS, FID και NLL που περιγράφηκαν στην παράγραφο 6, για τα τρία μοντέλα.

Για τον υπολογισμό των Inception Score (IS) και Fréchet Inception Distance (FID), χρησιμοποιήσαμε σε όλα τα σύνολα δεδομένων ένα κοινό προεκπαιδευμένο νευρωνικό δίκτυο στο ImageNet, ακολουθώντας μια τυπική προσέγγιση αξιολόγησης σε παραγωγικά μοντέλα.

Στον παρακάτω πίνακα παρουσιάζονται συγκεντρωτικά οι τιμές των μετρικών για κάθε μοντέλο, καθώς και η τιμή του FID από τη σχετική βιβλιογραφία για κάθε σύνολο (μαζί με την αντίστοιχη παραπομπή). Αναφέρεται μόνο η μετρική FID από τη βιβλιογραφία, καθώς είναι η μοναδική για την οποία υπάρχουν αναφορές και για τα τρία δεδομένα.

Μετρική	**ChestX-ray8**	**CIFAR-10**	**CelebA**
FID	18.2	3.19	3.26
IS	2.04 ± 0.05	9.48 ± 0.11	4.05 ± 0.02
NLL	2.53	3.75	3.11
FID (reference)	26.71 [62]	3.17 [2]	3.26 [17]

Πίνακας 2: Αποτελέσματα μετρικών για κάθε σύνολο δεδομένων

Η τιμή του IS είναι υψηλότερη στο σύνολο CIFAR-10, κάτι που είναι αναμενόμενο καθώς εξαρτάται από την ποικιλομορφία των παραγώμενων δειγμάτων. Αντίθετα στα σύνολα δεδομένων CelebA και ChestX-ray8 όπου τα δεδομένα είναι πιο ομοιογενή έχουμε μικρότερη τιμή στο δείκτη IS. Το αποτέλεσμα αυτό επιβεβαιώνει τη γνωστή αδυναμία της συγκεκριμένης μετρικής, καθώς δεν λαμβάνει υπόψη την ομοιότητα μεταξύ παραγόμενων και πραγματικών δειγμάτων, με αποτέλεσμα να επηρεάζεται έντονα από την εσωτερική ποικιλία του συνόλου.

Στην περίπτωση του FID, όπου έχουμε και τιμές από βιβλιογραφικές τιμές, παρατηρούμε ότι οι δικές μας τιμές είναι είτε αντίστοιχες μες αυτές τις βιβλιογραφίας, ή ακόμα και βελτιωμένες στην περίπτωση των ιατρικών δεδομένων. Οι πολύ κοντίνες τιμές στα σύνολα CIFAR-10 και CelebA με τη βιβλιογραφία οφείλονται στο γεγονός ότι οι συγκεκριμένες δημοσιεύσεις ανέφεραν λεπτομερώς την πειραματική διαδικασία που ακολούθησαν και παρατηρήσαμε αρκετές ομοιότητες στις παραμέτρους που χρησιμοποιήθηκαν σε αυτές σε σχέση με τη δική μας περίπτωση. Η δημοσίευση σχετικά με το ChestX-ray8, όμως δεν αναφέρει τις πειραματικές λεπτομέρειες και η απόκλιση μεταξύ της δικής μας τιμής με αυτή της δημοσίευσης, ενδεχομένως να οφείλεται σε καλύτερη επιλογή παραμέτρων ή εκπαίδευση για περισσότερα βήματα. Αξίζει να σημειωθεί ότι τα σύνολα CIFAR-10 και CelebA περιέχουν πολύχρωμες εικόνες με θεματολογία παρόμοια με το ImageNet, στο οποίο βασίστηκε ο ταξινομητής μας, γεγονός που ευνοεί τον υπολογισμό του FID. Αντιθέτως, το σύνολο ChestX-ray8 περιλαμβάνει μονοχρωματικές ιατρικές εικόνες, οι οποίες διαφέρουν σημαντικά από το ImageNet, οδηγώντας πιθανώς σε μικρή παραμόρφωση της μετρικής. Πράγματι, δοκιμάσαμε εναλλακτικά την εξαγωγή χαρακτηριστικών από ταξινομητή προεκπαιδευμένο σε ιατρικά δεδομένα, ωστόσο τα αποτελέσματα FID ήταν χειρότερα, και επιλέξαμε να διατηρήσουμε την αρχική επιλογή.

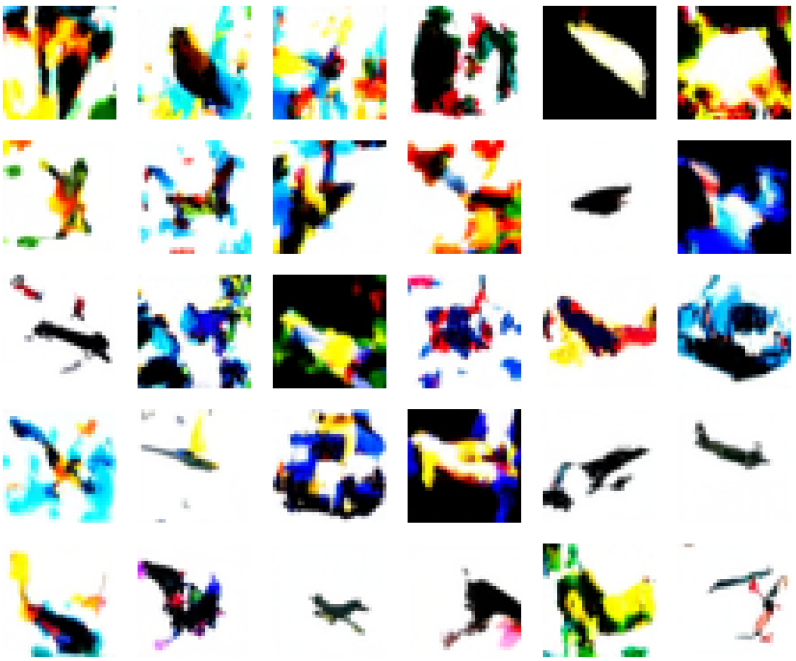
Τέλος, οι τιμές της αρνητικής λογαριθμοποιημένης πιθανοφάνειας (NLL) δείχνουν υψηλό βαθμό αποδοτικότητας και χαμηλή απώλεια σε όλα τα μοντέλα, με την καλύτερη τιμή να εμφανίζεται στο σύνολο ChestX-ray8, γεγονός που υποδεικνύει την υψηλή πιθανότητα τα παραγόμενα δείγματα να έχουν παραχθεί από την κατανομή του μοντέλου.

8.5 Αποτελέσματα Τροποποιημένης Αποθορυβοποίησης Πιθανοθεωρητικών Μοντέλων Διάχυσης (MDDPM)

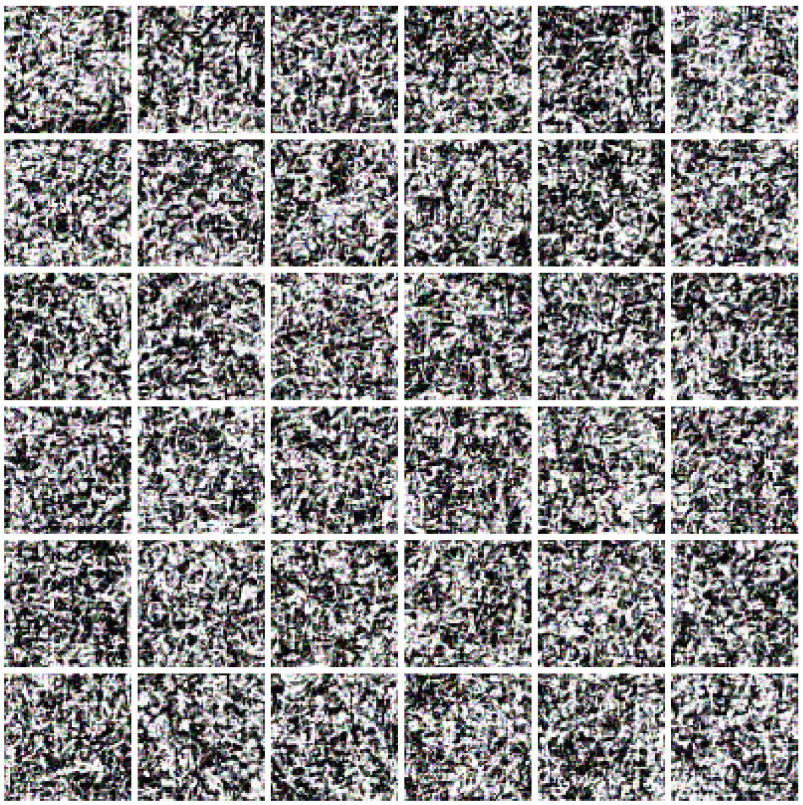
Με σκοπό την σύγκριση των αποτελεσμάτων μεταξύ DDPM και MDDPM επιχειρήσαμε να εκπαιδεύσουμε μοντέλα MDDPM για τα σύνολα δεδομένων CIFAR-10 και ChestX-ray8 με ακριβώς τις ίδιες παραμέτρους που ορίσαμε παραπάνω και μόνη διαφορά το άνω όριο του γραμμικού προγράμματος διακύμανσης, το οποίο μεταβάλαμε από 0.02 σε 0.04 αφού μελετήσαμε πειραματικά ότι αυτή ήταν η κατάλληλη τιμή ώστε για 1000 βήματα διάχυσης να πετύχουμε την επιθυμητή οριακή κατανομή.

Δυστυχώς τα μοντέλα αυτά δεν κατάφεραν την παραγωγή ρεαλιστικών δειγμάτων όπως παρατηρείται στα σχήματα 8.10 και 8.11, για αυτό δεν προχωρήσαμε ούτε στον υπολογισμό μετρικών για αυτά τα μοντέλα αλλά ούτε και σε σύγκριση των δειγμάτων αυτών που με τα δείγματα των DDPMs.

Παρόλα αυτά, τα δείγματα υποδεικνύουν την προοπτική που ενδεχομένως να έχει η συγκεκριμένη παραλλαγή, καθώς στα δείγματα που αφορούν το σύνολο δεδομένων CIFAR-10 μπορούμε να διακρίνουμε ξεκάθαρα κάποιες φιγούρες που ανήκουν στις κατηγορίες του συνόλου, έχουμε όμως πλήρη αποτυχία στον χρωματισμό των εικόνων. Αντίθετα στα δείγματα των ακτινογραφιών πετυχαίνουμε τους χρωματισμούς αλλά τα δείγματα παρουσιάζονται σαν ασπρόμαυρος θόρυβος.



Σχήμα 8.10: Δείγματα MDDPM CIFAR-10



Σχήμα 8.11: Δείγματα MDDPM CIFAR-10

Οι πιθανοί λόγοι της αποτυχίας των πειραμάτων μας, που θα μπορούσαν να λειτουργήσουν και ως έναυσμα για περαιτέρω μελέτη της μεθόδου είναι οι παρακάτω:

- Ανεπαρκής επιλογή υπερπαραμέτρων:
  Όπως αναφέραμε στην αντίστοιχη ενότητα, η εκπαίδευση των μοντέλων MDDPM έγινε με τιμές υπερπαραμέτρων ακριβώς τις ίδιες με τα μοντέλα DDPM και χωρίς να δοκιμαστούν διαφορετικές τιμές. Η επιλογή αυτή ενδεχομένως να είναι λανθασμένη καθώς η συγκεκριμένη τροποποίηση μπορεί να χρειάζεται διαφορετικές υπερπαραμέτρους για να λειτουργήσει επιτυχώς.

- Ανάγκη για εκτενέστερη εκπαίδευση των συγκεκριμένων μοντέλων.

- Πιθανή λανθασμένη επιλογή της παραμέτου συνδιασποράς ρ:
  Η παράμετρος συνδιασποράς ρ επιλέχθηκε, στα δύο μοντέλα που εκπαιδεύσαμε, να έχει τιμή ίση με 0.7 και για λόγους υπολογιστικών πόρων, δεν μπορέσαμε να εκπαιδεύσουμε μοντέλα για διαφορετικές τιμές του ρ για κάθε σύνολο δεδομένων, ώστε να δούμε την επίδραση που θα είχε αυτή η αλλαγή.

- Περαιτέρω μελέτη της αντίστροφης διαδικασίας ως προς την ευστάθεια.

9 Συμπεράσματα

Στην παρούσα διπλωματική εργασία διερευνήσαμε την Αποθορυβοποίηση Πιθανοθεωρητιρικών Μοντέλων Διάχυσης (DDPM) με σκοπό την θεωρητική θεμελίωση της και την πειραματική δοκιμή αυτών των μοντέλων σε ετερόκλητα σύνολα δεδομένων και ειδικότερα σε ιατρικά δεδομένα, όπου οι ανάγκες για συνθετικά δείγματα είναι υψηλές λόγω προβλημάτων ανισορροπίας, περιορισμένων διαθέσιμων δειγμάτων και ζητημάτων προσωπικών δεδομένων. Παράλληλα, υπήρξε προσπάθεια βελτίωσης των αποτελεσμάτων με την εισήγηση της Τροποποιημένης Αποθορύβωσης Πιθανοθεωρητικών Μοντέλων Διάχυσης (MDDPM). Αρχικός στόχος ήταν η σύγκριση των αποτελεσμάτων των δύο μεθόδων στα ίδια σύνολα δεδομένων και με τις ίδιες υπερπαραμέτρους, ώστε να εξάγουμε συμπεράσματα για το κατά πόσο αυτή η τροποποιημένη μέθοδος βελτιώνει τα ήδη υπάρχωντα μοντέλα.

Όπως είδαμε και στην αντίστοιχη ενότητα, η τροποποιημένη εκδοχή επιφέρει αποτελέσματα κατώτερα των προσδοκιών, αφού δεν επιτυγχάνει την παραγωγή ρεαλιστικών δειγμάτων. Παρόλα αυτά παρουσιάζει υποσχόμενα αποτελέσματα που εντείνουν το ενδιαφέρον για περαιτέρω μελέτη και πειραματισμό με τη συγκεκριμένη τροποποίηση ή παραλλαγές αυτής. Οι πιθανοί λόγοι αποτυχίας σχετίζονται με: ανεπαρκή επιλογή υπερπαραμέτρων, την ανάγκη για εκτενέστερη εκπαίδευση, πιθανή λανθασμένη επιλογή της παραμέτρου ρ. Επομένως η σύγκριση μεταξύ των δύο μεθόδων δεν είναι εφικτή, για αυτό το λόγο τα συμπεράσματα μας επικεντρώνονται στην ανάλυση των αποτελεσμάτων των μοντέλων DDPM στα τρία σύνολα δεδομένων.

Η εκπαίδευση ενός μοντέλου διάχυσης στο σύνολο CIFAR-10, το οποίο αποτελείται από πολύχρωμες εικόνες χαμηλής ανάλυσης με σημαντική ποικιλομορφία, απέδωσε εξαιρετικά αποτελέσματα με εξαιρετικά ρεαλιστικά παραγόμενα δείγματα και μετρικές συγκρίσιμες με αυτές της βιβλιογραφίας. Αντίστοιχα αποτελέσματα παρατηρήσαμε επίσης και στο σύνολο CelebA, όπου το μοντέλο μας κατάφερε να παράγει ρεαλιστικά πρόσωπα, με έντονες διαφορές μεταξύ τους, αποδεικνύοντας για άλλη μια φορά τις δυνατότητες των συγκεκριμένων μοντέλων.

Τα αποτελέσματα μας στο σύνολο δεδομένων ChestX-ray8, είναι άκρως υποσχόμενα, καθώς επιτυγχάνουν τιμές μετρικών ακόμα καλύτερες από ότι αυτές τις βιβλιογραφίας και παράγουν συνθετικά δεδομένα με κοινά χαρακτηριστικά με αυτά του συνόλου δεδομένων εκπαίδευσης, που μπορούν να χρησιμοποιηθούν για τον εμπλουτισμό των διαθέσιμων ακτινογραφιών θώρακα και την αντιμετώπιση προβλημάτων ανισορροπίας κλάσεων, μικρού αριθμού δειγμάτων και προσωπικών δεδομένων που παρατηρούνται συχνά σε σύνολα ιατρικής φύσεως.

Συνολικά, η μελέτη αυτή καταδεικνύει ότι το μοντέλο DDPM αποτελεί μια ισχυρή και ευέλικτη προσέγγιση για την παραγωγή συνθετικών δειγμάτων, ικανή να αποδώσει ρεαλιστικά δείγματα ακόμα και σε συνθήκες περιορισμένων ή εξειδικευμένων δεδομένων. Η υψηλή επίδοση σε όλες τις μετρικές (FID, IS, NLL) και σε όλα τα σύνολα υποδηλώνει τη γενικευσιμότητα του μοντέλου και τη σταθερότητά του στην εκπαίδευση .Η εφαρμογή των μοντέλων αυτών σε ια-

τρικά δεδομένα αποδείχθηκε όχι μόνο εφικτή αλλά και ιδιαίτερα αποδοτική, με προοπτική αξιοποίησης σε πραγματικά σενάρια, όπως η δημιουργία συνθετικών ακτινογραφιών για εκπαίδευση διακριτικών μοντέλων, αντιμετώπιση ανισορροπίας κατηγοριών ή ακόμα και ανωνυμοποίηση δεδομένων.

10 Μελλοντικές Κατευθύνσεις

Η παρούσα μελέτη και τα αποτελέσματα της, ανέδειξαν τις δυνατότητες των πιθανοθεωρητικών μοντέλων διάχυσης στην παραγωγή ρεαλιστικών συνθετικών δειγμάτων σε ετερόκλητα σύνολα δεδομένων. Παρόλο που τα αποτελέσματα ήταν ικανοποιητικά, όσο αφορά τα πειράματα DDPM, προκύπτουν αρκετές κατευθύνσεις για περαιτέρω μελέτη και εμβάθυνση στη συγκεκριμένη μεθοδολογία. Κάποιες από αυτές τις κατευθύνσεις είναι:

- Περαιτέρω μελέτη της τροποποιημένης μεθόδου MDDPM:
  Όπως αναφέραμε, παρόλο που η τροποποιημένη μέθοδος δεν επέφερε τα αναμενόμενα αποτελέσματα, παρουσίασε ενδείξεις ότι μπορεί να βελτιωθεί ώστε να επετευχθεί ο στόχος αυτός. Επομένως προτείνεται εξερεύνηση περισοτέρων τιμών της παραμέτρου συνδιασποράς ρ, για κάθε ένα από τα υπό εξέταση σύνολα δεδομένων, εκπαίδευση των μοντέλων για περισότερα βήματα ή σε μεγαλύτερα σύνολα δεδομένων και πειραματισμός με διαφορετικές τιμές των παραμέτρων για κάθε μοντέλο.

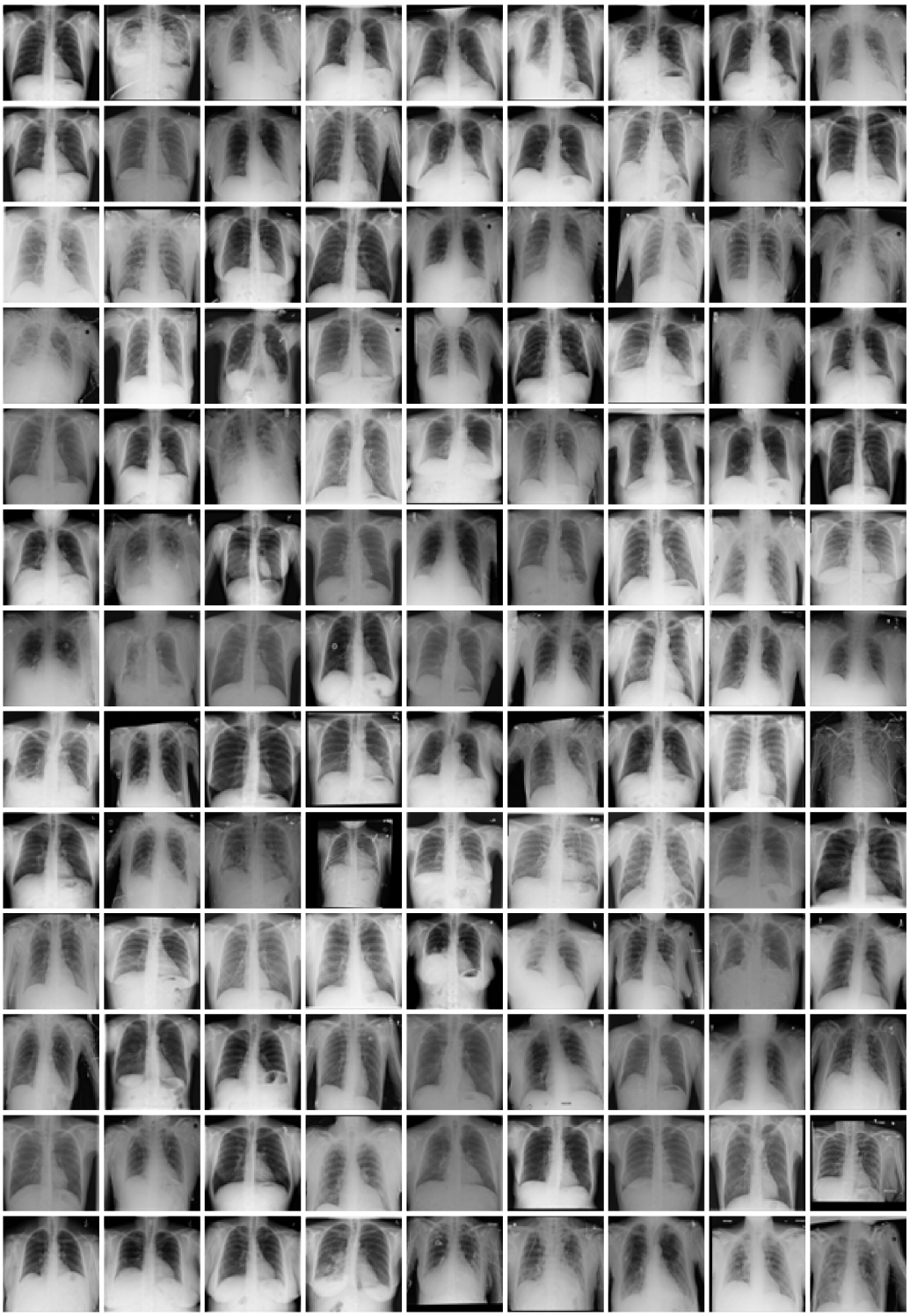
- Επέκταση των μοντέλων σε ιατρικά δεδομένα μεγαλύτερης διάστασης:
  Όπως αναφέραμε και στην αντίστοιχη παράγραφο για την εφαρμογή παραγωγικών μοντέλων σε ιατρικά δεδομένα, τα τελευταία χρόνια υπήρξαν αρκετές έρευνες που μελέτησαν την εφαρμογή αυτή, κάποιες από τις οποιές ασχολούνται με τριδιάστατα ιατρικά δεδομένα, όπως MRI. Μια μελλοντική επέκταση της συγκεκριμένης εργασίας, θα ήταν η δοκιμή τόσο των DDPM, όσο και της τροποποιημένης μεθόδου MDDPM, εφόσον επιφέρει τα απαιτούμενα αποτελέσματα, σε τέτοιου είδους δεδομένα.

- Μελέτη και θεωρητική θεμελίωση βελτιώσεων και διαφόρων παραλλαγών της Αποθορυβοποίησης Πιθανοθεωρητικών Μοντέλων Διάχυσης (DDPM) που υπάρχουν στην βιβλιογραφία.

- Μελέτη εναλλακτικών μετρικών για την αξιολόγηση της απόδοσης παραγωγικών μοντέλων, εφαρμοσμένων σε ιατρικά δεδομένα.
  Όπως είδαμε τα FID και IS δεν ήταν ιδανικές μετρικές για παραγωγικά μοντέλα εκπαιδευμένα σε ιατρικά δεδομένα, καθώς βασίζονται σε νευρωνικά δίκτυα εκπαιδευμένα στο σύνολο ImageNet, του οποίου τα δείγματα έχουν μηδαμινές ομοιότητες με τα ιατρικά δεδομένα με τα οποία ασχολούμαστε. Επομένως η αναζήτηση μετρικών αποκλειστικών σε τέτοια μοντέλα είναι απαραίτητη.

- Μελέτη της εφαρμογής των Μοντέλων Διάχυσης σε διαφορετικούς τύπους δεδομένων (π.χ. ήχος, βίντεο κ.α.)

Παράρτημα Α: Περισότερα δείγματα



Σχήμα 10.1: Περισότερα Δείγματα ChestX-ray8



Σχήμα 10.2: Περισότερα Δείγματα CelebA

Βιβλιογραφία

[1] Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., & Ganguli, S. (2015, June). Deep unsupervised learning using nonequilibrium thermodynamics. In International conference on machine learning (pp. 2256-2265). pmlr.

[2] Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. Advances in neural information processing systems, 33, 6840-6851.

[3] Nichol, A. Q., & Dhariwal, P. (2021, July). Improved denoising diffusion probabilistic models. In International conference on machine learning (pp. 8162-8171). PMLR.

[4] Pinaya, W. H., Tudosiu, P. D., Dafflon, J., Da Costa, P. F., Fernandez, V., Nachev, P., ... & Cardoso, M. J. (2022, September). Brain imaging generation with latent diffusion models. In MICCAI Workshop on Deep Generative Models (pp. 117-126). Cham: Springer Nature Switzerland.

[5] Txurio, M. S., Román, K. L. L., Marcos-Carrión, A., Castellote-Huguet, P., Santabárbara-Gómez, J. M., Oliver, I. M., & Ballester, M. A. G. (2023, May). Diffusion models for realistic CT image generation. In International KES Conference on Innovation in Medicine and Healthcare (pp. 335-344). Singapore: Springer Nature Singapore.

[6] Shrivastava, A., & Fletcher, P. T. (2023, October). Nasdm: Nuclei-aware semantic histopathology image generation using diffusion models. In international conference on medical image computing and computer-assisted intervention (pp. 786-796). Cham: Springer Nature Switzerland.

[7] Krause A., & Hübotter J. (2023). Probabilistic Artificial Intelligence. MIT Press.

[8] Blei, D. M., Kucukelbir, A., & McAuliffe, J. D. (2018). Variational inference: A review for statisticians. Journal of the American Statistical Association, 112(518), 859-877.

[9] Russell, S. J., & Norvig, P. (2016). Artificial intelligence: a modern approach. pearson.

[10] Ben-Gal, I. (2008). Bayesian networks. Encyclopedia of statistics in quality and reliability.

[11] Morales, E. F., & Escalante, H. J. (2022). Chapter 6 - A brief introduction to supervised, unsupervised, and reinforcement learning. In A. A. Torres-García, C. A. Reyes-García, L. Villaseñor-Pineda, & O. Mendoza-Montoya (Eds.), Biosignal processing and classification using computational learning and intelligence (pp. 111–129).

[12] Harshvardhan, G. M., Gourisaria, M. K., Pandey, M., & Rautaray, S. S. (2020). A comprehensive survey and analysis of generative models in machine learning. Computer Science Review, 38, Article 100285.

[13] Khader, F., Müller-Franzes, G., Tayebi Arasteh, S., Han, T., Haarburger, C., Schulze-Hagen, M., ... & Truhn, D. (2023). Denoising diffusion probabilistic models for 3D medical image generation. Scientific Reports, 13(1), 7303.

[14] Kazerouni, A., Aghdam, E. K., Heidari, M., Azad, R., Fayyaz, M., Hacihaliloglu, I., & Merhof, D. (2023). Diffusion models in medical imaging: A comprehensive survey. Medical image analysis, 88, 102846.

[15] Yang, L., Zhang, Z., Song, Y., Hong, S., Xu, R., Zhao, Y., ... & Yang, M. H. (2023). Diffusion models: A comprehensive survey of methods and applications. ACM Computing Surveys, 56(4), 1-39.

[16] Cao, H., Tan, C., Gao, Z., Xu, Y., Chen, G., Heng, P. A., & Li, S. Z. (2024). A survey on generative diffusion models. IEEE Transactions on Knowledge and Data Engineering.

[17] Song, J., Meng, C., & Ermon, S. (2020). Denoising diffusion implicit models. arXiv preprint arXiv:2010.02502.

[18] Kingma, D. P., & Welling, M. (2013, December). Auto-encoding variational bayes.

[19] Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. science, 313(5786), 504-507.

[20] Doersch, C. (2016). Tutorial on variational autoencoders. arXiv preprint arXiv:1606.05908.

[21] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. Advances in neural information processing systems, 27.

[22] Du, Y., & Mordatch, I. (2019). Implicit generation and modeling with energy based models. Advances in neural information processing systems, 32.

[23] Welling, M., & Hinton, G. E. (2002, August). A new learning algorithm for mean field Boltzmann machines. In International conference on artificial neural networks (pp. 351-357). Berlin, Heidelberg: Springer Berlin Heidelberg.

[24] Hyvärinen, A., & Dayan, P. (2005). Estimation of non-normalized statistical models by score matching. Journal of Machine Learning Research, 6(4).

[25] Gutmann, M., & Hyvärinen, A. (2010, March). Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In Proceedings of the thirteenth international conference on artificial intelligence and statistics (pp. 297-304). JMLR Workshop and Conference Proceedings.

[26] Kobyzev, I., Prince, S. J., & Brubaker, M. A. (2020). Normalizing flows: An introduction and review of current methods. IEEE transactions on pattern analysis and machine intelligence, 43(11), 3964-3979.

[27] Nakano, Y. (2024). Convergence of the denoising diffusion probabilistic models. arXiv preprint arXiv:2406.01320.

[28] Haykin, S. (2009). Neural networks and learning machines (3rd ed.). Pearson.

[29] Διαμαντάρας, Κ., & Μπότσης, Δ. (2019). Μηχανική μάθηση. Αθήνα: Κλειδάριθμος.

[30] O'shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. arXiv preprint arXiv:1511.08458.

[31] Fukushima, K. (1988). Neocognitron: A hierarchical neural network capable of visual pattern recognition. Neural Networks, 1(2), 119-130

[32] Murphy, K. P. (2022). Probabilistic machine learning: an introduction. MIT press.

[33] Scherer, D., Müller, A., & Behnke, S. (2010, September). Evaluation of pooling operations in convolutional architectures for object recognition. In International conference on artificial neural networks (pp. 92-101). Berlin, Heidelberg: Springer Berlin Heidelberg.

[34] Kolarik, M., Burget, R., & Riha, K. (2019, October). Upsampling algorithms for autoencoder segmentation neural networks: a comparison study. In 2019 11th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT) (pp. 1-5). IEEE.

[35] Ioffe, S., & Szegedy, C. (2015, June). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International conference on machine learning (pp. 448-456). pmlr.

[36] Wu, Y., & He, K. (2018). Group normalization. In Proceedings of the European conference on computer vision (ECCV) (pp. 3-19).

[37] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18 (pp. 234-241). Springer international publishing.

[38] Drozdzal, M., Vorontsov, E., Chartrand, G., Kadoury, S., & Pal, C. (2016, September). The importance of skip connections in biomedical image segmentation. In International Workshop on Deep Learning in Medical Image Analysis (pp. 179-187). Cham: Springer International Publishing.

[39] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.

[40] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

[41] Chen, X., Mishra, N., Rohaninejad, M., & Abbeel, P. (2018, July). Pixelsnail: An improved autoregressive generative model. In International conference on machine learning (pp. 864-872). PMLR.

[42] SrivastavaN, H., & Krizhevsky, A. (2014). Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 15(1), 1929.

[43] Theodoridis, S. (2015). Machine learning: a Bayesian and optimization perspective. Academic press.

[44] Chollet, F., & Chollet, F. (2021). Deep learning with Python. Simon and Schuster.

[45] Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). Deep learning (Vol. 1, No. 2). Cambridge: MIT press.

[46] Elgendy, M. (2020). Deep learning for vision systems. Manning.

[47] Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

[48] Duchi, J., Hazan, E., & Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. Journal of machine learning research, 12(7).

[49] Tieleman, T. (2012). Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural networks for machine learning, 4(2), 26.

[50] Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2023). Dive into deep learning. Cambridge University Press.

[51] Rojas, R., & Rojas, R. (1996). The backpropagation algorithm. Neural networks: a systematic introduction, 149-182.

[52] Barratt, S., & Sharma, R. (2018). A note on the inception score. arXiv preprint arXiv:1801.01973.

[53] Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., & Chen, X. (2016). Improved techniques for training gans. Advances in neural information processing systems, 29.

[54] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2818-2826).

[55] Woodland, M., Castelo, A., Al Taie, M., Albuquerque Marques Silva, J., Eltaher, M., Mohn, F., ... & Brock, K. K. (2024, October). Feature extraction for generative medical imaging evaluation: New evidence against an evolving trend. In International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 87-97). Cham: Springer Nature Switzerland.

[56] Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., & Hochreiter, S. (2017). Gans trained by a two time-scale update rule converge to a local nash equilibrium. Advances in neural information processing systems, 30.

[57] Fréchet, M. (1957). Sur la distance de deux lois de probabilité. In Annales de l'ISUP (Vol. 6, No. 3, pp. 183-198).

[58] Vaserstein, L. N. (1969). Markov processes over denumerable products of spaces, describing large systems of automata. Problemy Peredachi Informatsii, 5(3), 64-72.

[59] Hollander, F. D. (2012). Probability Theory: The Coupling Method. Mathematical Institute, Leiden University.

[60] Dowson, D. C., & Landau, B. (1982). The Fréchet distance between multivariate normal distributions. Journal of multivariate analysis, 12(3), 450-455.

[61] Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., & Summers, R. M. (2017). Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2097-2106).

[62] Pan, S., Wang, T., Qiu, R. L., Axente, M., Chang, C. W., Peng, J., ... & Yang, X. (2023). 2D medical image synthesis using transformer-based denoising diffusion probabilistic model. Physics in Medicine & Biology, 68(10), 105004.

[63] Liu, Z., Luo, P., Wang, X., & Tang, X. (2015). Deep learning face attributes in the wild. Proceedings of the IEEE International Conference on Computer Vision (ICCV), 3730–3738. https://doi.org/10.1109/ICCV.2015.425