



ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ

Σχολή Μηχανικών

Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

**Σχεδίαση και Ανάπτυξη
Αμφίδρομων Γεννητικών Ανταγωνιστικών
Δικτύων υπό όρους για την Ολοκλήρωση
Τρισδιάστατων Σχημάτων**

Διπλωματική Εργασία

της

Μαγκνταλένα Ιζαμπέλα Κζέσνιακ

Επιβλέπων Καθηγητής

Αθανάσιος Βουλόδημος, Επίκουρος Καθηγητής

Αθήνα-Αιγάλεω, Οκτώβριος 2021



ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ

Σχολή Μηχανικών

Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Σχεδίαση και Ανάπτυξη Αμφίδρομων Γεννητικών Ανταγωνιστικών Δικτύων υπό όρους για την Ολοκλήρωση Τρισδιάστατων Σχημάτων

Όνοματεπώνυμο Μαγκνταλένα Ιζαμπέλα Κζέσνιακ

Αριθμός Μητρώου 711161044

Επιβλέπων Καθηγητής Αθανάσιος Βουλόδημος
Επίκουρος Καθηγητής

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 14^η Οκτωβρίου του 2021.

Αθανάσιος Βουλόδημος
Επίκουρος Καθηγητής

Πάρις Μαστοροκόστας
Καθηγητής

Αναστάσιος Κεσίδης
Αναπληρωτής Καθηγητής

Αθήνα-Αιγάλεω , Οκτώβριος 2021



UNIVERSITY OF WEST ATTICA

School of Engineering

Department of Informatics and Computer Engineering

3D Shape Completion via Bidirectional Conditional Generative Adversarial Networks

Full Name Magdalena Izabela Krzesniak

Identification Number 711161044

Supervisor Athanasios Voulodimos
Assistant Professor

Athens-Egaleo, October 2021

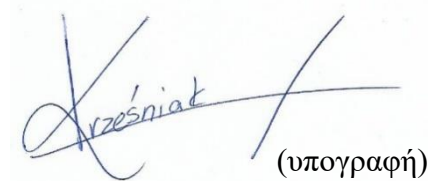
ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Η κάτωθι υπογεγραμμένη **Κζέσνιακ Μαγκνταλένα-Ιζαμπέλα** του Μαριούς, με αριθμό μητρώου **711161044** φοιτήτρια του Πανεπιστημίου Δυτικής Αττικής της Σχολής Μηχανικών του **Τμήματος Μηχανικών Πληροφορικής και Υπολογιστών**, δηλώνω υπεύθυνα ότι:

«Είμαι συγγραφέας αυτής της διπλωματικής εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, οι όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε ακριβώς είτε παραφρασμένες, αναφέρονται στο σύνολό τους, με πλήρη αναφορά στους συγγραφείς, τον εκδοτικό οίκο ή το περιοδικό, συμπεριλαμβανομένων και των πηγών που ενδεχομένως χρησιμοποιήθηκαν από το διαδίκτυο. Επίσης, βεβαιώνω ότι αυτή η εργασία έχει συγγραφεί από μένα αποκλειστικά και αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο δικής μου, όσο και του Ιδρύματος. Παράβαση της ανωτέρω ακαδημαϊκής μου ευθύνης αποτελεί ουσιώδη λόγο για την ανάκληση του πτυχίου μου»

Η Δηλούσα,

Κζέσνιακ Μαγκνταλένα-Ιζαμπέλα



(υπογραφή)

Περίληψη

Σε ένα μικρό χρονικό διάστημα, η ραγδαία αύξηση του ενδιαφέροντος για την **Όραση Υπολογιστών** και τη **Βαθιά Μάθηση** οδήγησε σε έναν τεράστιο αριθμό διαφόρων εφαρμογών πάνω σε κείμενο, εικόνες, και βίντεο. Οι εφαρμογές αυτές ποικίλλουν από απλά προβλήματα όπως η «ανίχνευση κίνησης σε στατικές κάμερες» σε πιο πολύπλοκα όπως η «Σύλληψη Αντικειμένων από ένα Ρομπότ μέσω όρασης». Με την ταχεία ανάπτυξη των Γραφικών Υπολογιστών και των τεχνολογιών απόκτησης τρισδιάστατων μοντέλων, οι εφαρμογές Βαθιάς Μάθησης σε τρισδιάστατα μοντέλα αντικειμένων έχουν προσελκύσει όλο και περισσότερη προσοχή. Οι αισθητήρες μπορούν πλέον να παρέχουν τρισδιάστατα δεδομένα με πλούσια γεωμετρία, σχήμα και κλίμακα, και συνοδευόμενα από δισδιάστατες εικόνες, μπορούν να μας δώσουν μια καλύτερη κατανόηση του περιβάλλοντος.

Τα Τρισδιάστατα Δεδομένα μπορούν να αναπαρασταθούν με πολλαπλούς τρόπους, εκ των οποίων ο πιο συχνά χρησιμοποιούμενος είναι τα **σημεία νέφους**, τα οποία διατηρούν τις πρωταρχικές γεωμετρικές πληροφορίες σε τρισδιάστατο χώρο, χωρίς διακριτοποίηση. Ωστόσο, η Βαθιά Μάθηση σε σημεία νέφους είναι ακόμα στα αρχικά στάδια λόγω των ιδιαίτερων προκλήσεων που αντιμετωπίζουμε κατά την επεξεργασία των σημείων νέφους με Βαθιά Νευρωνικά Δίκτυα. Μία από τις κύριες προκλήσεις είναι ότι μοντέλα καλής ποιότητας είναι δύσκολο να αποκτηθούν και συχνά παρουσιάζουν έλλειψη τμημάτων δεδομένων. Ένας τρόπος επίλυσης του προβλήματος αυτού είναι η δημιουργία ενός **μοντέλου ολοκλήρωσης τρισδιάστατων σχημάτων** ικανό να αποκαταστήσει τα τμήματα που λείπουν.

Σε αυτή την διατριβή, θα ερευνήσουμε μια οικογένεια μοντέλων που ονομάζονται **Βαθιά Παραγωγικά Μοντέλα**. Συγκεκριμένα θα επεκταθούμε σε βάθος στα **Γεννητικά Ανταγωνιστικά Δίκτυα (GAN)**, καθώς και στους **Μεταβλητούς Αυτοκωδικοποιητές (VAE)**. Τα GAN δημιουργούν νέα δεδομένα βάση ένα σύνολο δεδομένων εκπαίδευσης, με τα οποία έχουν ίδια στατιστικά, ενώ οι VAE διασφαλίζουν ότι η διανομή των κωδικοποιήσεων του κανονικοποιείται κατά την διάρκεια της εκπαίδευσης, έτσι ώστε ο λανθάνων χώρος να είναι επαρκής ποιότητας, για την δημιουργία νέων δεδομένων. Ο κύριος στόχος είναι η δημιουργία ενός δικτύου ολοκλήρωσης σχημάτων βάσει σημείου, βασισμένο σε Βαθιά Παραγωγικά Μοντέλα, το οποίο μπορεί να ολοκληρώσει ένα μερικό σχήμα με λογικά αποτελέσματα.

Λέξεις Κλειδιά

Βαθιά Μάθηση, Νευρωνικά Δίκτυα, Αμφίδρομα Γεννητικά Ανταγωνιστικά Δίκτυα υπό όρους, Μεταβλητοί Αυτοκωδικοποιητές, Σημεία Νέφους, Τρισδιάστατα σχήματα, Ολοκλήρωση σχήματος

Abstract

In a brief period, the rapid increase in interest in **Computer Vision** and **Deep Learning** has led to a plethora of different applications on text, images, and videos. Those applications range from simple problems such as “motion detection in static cameras” or “Spam Filtering” to more complex ones such as “Robot Object Grasping through Vision” or “Image-Caption Generation.” With the expeditious development of Computer Graphics, and 3D models acquisition technologies, Deep Learning applications on 3D Object Models have attracted increased attention. Sensors can provide us with 3D data with rich geometry, shape, and scale information, and accompanied by 2D images, can grant us with a better understanding of a certain environment.

Three-Dimensional Data can be represented in a variety of ways, one of the most commonly used being **point clouds** (PC), which preserve the primary geometric information in 3D space without any discretization. However, even Deep Learning on Point Clouds is still in its preliminary stages due to challenges encountered when processing PCs with Deep Neural Networks. One of the main challenges is that decent quality PC Models are hard to obtain and often miss parts of data. A way to solve this problem is to create a **3D Shape Completion Model** capable of restoring those missing parts.

In this dissertation, we will research a family of models, with a variety of applications in Computer Vision, called **Deep Generative Models**. Specifically, we will expand in-depth on **Generative Adversarial Networks** (GANs) as well as **Variational Autoencoders** (VAE). GANs create new data based on a set of given training data, with which they have the same statistics, whereas VAEs ensure that their encodings distribution is regularized during the training so that the latent space is of sufficient quality, to generate new data. The main objective is to create a point-based shape completion network based on Deep Generative Models that can complete a partial scheme with reasonable results.

Key words

Deep Learning, Neural Networks (NN), Bidirectional Conditional Generative Adversarial Network (BicGAN), Variational Autoencoders (VAE), Point Clouds, 3D Shapes, Shape completion

Ευχαριστίες

Η παρούσα διπλωματική εργασία εκπονήθηκε στο πλαίσιο του Προπτυχιακού Προγράμματος Σπουδών του Τμήματος Πληροφορικής και Υπολογιστών της Σχολής Μηχανικών του Πανεπιστημίου Δυτικής Αττικής και σηματοδοτεί την ολοκλήρωση των σπουδών μου. Ως την ελάχιστη δυνατή μνεία, με την παρούσα παράγραφο θα ήθελα να ευχαριστήσω θερμά τους ανθρώπους με τους οποίους συνεργάστηκα και συνέβαλαν στην εκπόνηση της εργασίας αυτής.

Ευχαριστώ θερμά τον επιβλέποντα καθηγητή μου, κύριο Αθανάσιο Βουλόδημο, για την καθοδήγηση του στην σωστή επιλογή θέματος για την εργασία μου, την εμπιστοσύνη που μου έδειξε εξ' αρχής καθώς και την επιστημονική του καθοδήγηση, τις υποδείξεις του, την επιμονή του, το αμείωτο ενδιαφέρον του, τη συμπαράστασή του, τη συνεχή του υποστήριξη και το αμείωτο ενδιαφέρον που έδειξε από την αρχή μέχρι το τέλος.

Επιπλέον, ιδιαίτερες ευχαριστίες θα ήθελα να απευθύνω στην συνεργάτιδα στην εταιρεία στην οποία εργάζομαι, Ελένη Βάθη, για τις εποικοδομητικές της υποδείξεις και την πολύτιμη συμβολή της στην ολοκλήρωση αυτής της εργασίας. Τόσο η επιστημονική όσο και η πνευματική της στήριξη, ιδιαίτερα στα αρχικά στάδια της εργασίας, ήταν ιδιαίτερα σημαντικές για μένα.

Τέλος, σε προσωπικό επίπεδο θα ήθελα να εκφράσω την ευγνωμοσύνη μου στην μητέρα μου, καθώς και άλλα κοντινά μου πρόσωπα όπως οι φίλοι μου για όλη τη στήριξη, τη συμπαράσταση και την κατανόησή τους, καθ' όλη τη διάρκεια των σπουδών μου.

Πίνακας Περιεχομένων

Περίληψη	vii
Abstract.....	ix
Ευχαριστίες.....	xi
Πίνακας Περιεχομένων.....	xii
Κατάλογος Πινάκων	xiv
Κατάλογος Σχημάτων.....	xiv
Κατάλογος Συντομογραφιών	xvi
1. Εισαγωγή.....	1
1.1 Περιγραφή Προβλήματος και Στόχοι Εργασίας	2
1.2 Διάρθρωση Κεφαλαίων	3
2. Θεωρητικό Υπόβαθρο	4
2.1 Τεχνητή Νοημοσύνη.....	4
2.1.1 Νοημοσύνη	4
2.1.2 Κατηγορίες TN	5
2.2 Μηχανική Μάθηση	6
2.2.1 Κατηγορίες Μηχανικής Μάθησης	7
2.2.2 Βαθιά Μάθηση.....	10
2.2.2.1 Τανυστές.....	11
2.3 Νευρωνικά Δίκτυα	12
2.3.1 Τεχνητός Νευρώνας.....	12
2.3.2 Κατηγορίες.....	14
2.3.3 Βαθιά Νευρωνικά Δίκτυα	16
2.3.4 Βασικοί Όροι.....	18
2.3.4.1 Μεταβλητότητα.....	18
2.3.4.2 Πόλωση.....	18
2.3.4.3 Δίλημμα πόλωσης-μεταβλητότητας.....	19
2.3.4.4 Ανεπαρκής προσαρμογή και υπερπροσαρμογή.....	19
2.4 Γεννητικά Ανταγωνιστικά Δίκτυα	20
2.4.1 Παραγωγικά και Διευκρινιστικά Μοντέλα	20
2.4.2 Λειτουργία των GANs	21
2.4.3 Προβλήματα στην Εκπαίδευση.....	25
2.4.4 Είδη GANs.....	26

2.4.4.1	Απλά GANs.....	27
2.4.4.2	Γεννητικά Ανταγωνιστικά Δίκτυα υπό όρους.....	28
2.4.4.3	Αμφίδρομα Γεννητικά Ανταγωνιστικά Δίκτυα.....	30
2.4.4.4	Ημι-Επιβλεπόμενα Γεννητικά Ανταγωνιστικά Δίκτυα.....	30
2.4.4.5	Γεννητικά Ανταγωνιστικά Δίκτυα Μεγιστοποίησης Πληροφοριών	31
2.4.4.6	Βοηθητικοί Ταξινομητές.....	32
2.5	Μεταβλητοί Αυτοκωδικοποιητές.....	33
2.6	Αναπαράσταση σχήματος.....	36
2.6.1	Ογκομετρικό πλέγμα εικονοστοιχείων	36
2.6.2	Πολυγωνικό Πλέγμα.....	38
2.6.3	Νέφος Σημείων	39
3.	Σύνολο Δεδομένων.....	40
3.1	ShapeNet.....	40
4.	Μεθοδολογία.....	47
4.1	Σχετικές Έρευνες.....	47
4.2	Μοντέλο Εκπαίδευσης	47
4.3	Διαδικασία εκπαίδευσης	50
4.4	Μετρικές Απόστασης.....	52
4.4.1	Απόσταση Ελάχιστου έργου	52
4.4.2	Απόσταση Hausdorff	54
5.	Πειράματα και Αξιολόγηση.....	57
5.1	Αποτελέσματα μοντέλου VAE	58
5.2	Αποτελέσματα μοντέλου GAN.....	63
6.	Σύνοψη.....	73
6.1	Συμπεράσματα	73
6.2	Περαιτέρω Έρευνα.....	75
	Βιβλιογραφία.....	77

Κατάλογος Πινάκων

Πίνακας 2.1. Τέσσερα επίπεδα Τεχνητής Νοημοσύνης με βάση τις λειτουργίες τους	5
Πίνακας 2.2. Διαφορές μεταξύ Μηχανικής και Βαθιάς Μάθησης	10
Πίνακας 2.3 Έγκριση και απόρριψη Δεδομένων στα cGANs.....	29
Πίνακας 3.1 Διαθέσιμα Σύνολα Δεδομένων	40
Πίνακας 3.2 Κατηγορίες αντικειμένων στο ShapeNet	43
Πίνακας 3.3 Τρισδιάστατο Μοντέλο σε μορφή PC και Ελλιπούς PC.	46
Πίνακας 4.1 Αρχιτεκτονική VAE – Στρώματα και Έξοδοι.....	48
Πίνακας 4.2 Αρχιτεκτονική GAN – Στρώματα και Έξοδοι	49
Πίνακας 5.1 Μετρικές Αξιολόγησης Διαδικασίας Εκπαίδευσης του μοντέλου GAN.....	65

Κατάλογος Σχημάτων

Εικόνα 2.1 Ταξινόμηση της Τεχνητής Νοημοσύνης.....	6
Εικόνα 2.2 Μοντέλα επιβλεπόμενη μάθησης	8
Εικόνα 2.3 Κατηγορίες Μηχανικής Μάθησης.....	9
Εικόνα 2.4 Διάγραμμα ροής Μηχανικής και Βαθιάς Μάθησης	11
Εικόνα 2.5 Σχέση απόδοσης και Δεδομένων.	11
Εικόνα 2.6 Τεχνητός Νευρώνας	13
Εικόνα 2.7 Συναρτήσεις ενεργοποίηση.	13
Εικόνα 2.8 Δομή Νευρωνικών Δικτύων	17
Εικόνα 2.9 Μεταβλητότητα και πόλωση Μοντέλου	19
Εικόνα 2.10 Προσαρμογή Μοντέλου	20
Εικόνα 2. 11 Εκπαίδευση Γεννητικών Ανταγωνιστικών Δικτύων 1.....	23
Εικόνα 2.12 Εκπαίδευση Γεννητικών Ανταγωνιστικών Δικτύων 2.....	24
Εικόνα 2.13 Παραλλαγές GANs	26
Εικόνα 2.14 Αρχιτεκτονική Απλών GANs	27
Εικόνα 2.15 Αρχιτεκτονική Deep Convolutional GANs.....	28
Εικόνα 2.16 Αρχιτεκτονική Conditional GANs	29
Εικόνα 2.17 Αρχιτεκτονική Bidirectional GANs	30

Εικόνα 2.18 Αρχιτεκτονική <i>Semi-Supervised GANs</i>	31
Εικόνα 2.19 Αρχιτεκτονική <i>Information Maximizing GANs</i>	32
Εικόνα 2.20 Αρχιτεκτονική <i>Auxiliary Classifier GANs</i>	32
Εικόνα 2.21 Βασική Αρχιτεκτονική VAE	33
Εικόνα 2.22 Γραφικό Μοντέλο που εμπλέκεται στο VAE.	34
Εικόνα 2.23 Αναπαράσταση 3D σχήματος	36
Εικόνα 2.24 Δομή voxel και πλέγματος voxel.....	37
Εικόνα 2.25 Voxel μέσα σε voxel Grid.....	37
Εικόνα 2.26 Αναπαράσταση σφαίρας με Πολυγωνικό Πλέγμα	38
Εικόνα 2.27 Βασικά τρισδιάστατα σχήματα ως νέφη σημείων	39
Εικόνα 3.1 Κατηγορίες του <i>ShapeNetCore v2</i>	43
Εικόνα 3.2 Απεικόνιση του προβλήματος της μεταφοράς	52
Εικόνα 4.1 Η αρχιτεκτονική του Υβριδικού Γεννητικού Μοντέλου.....	49
Εικόνα 4.2 Γραφική Παράσταση της Συνάρτησης Ενεργοποίησης <i>LeakyReLU</i>	50
Εικόνα 4.3 Παράδειγμα υπολογισμού <i>EMD</i>	54
Εικόνα 4.4 Παράδειγμα υπολογισμού <i>Hausdorff</i>	56
Εικόνα 5.1 Τρισδιάστατα Μοντέλα ως Νέφη n αριθμού Σημείων	57
Εικόνα 5.2 Έξοδος Μοντέλου VAE ανά επανάληψη εκπαίδευσης.....	59
Εικόνα 5.3 Είσοδος και έξοδος μοντέλου VAE	59
Εικόνα 5.4 Έξοδοι για διαφορετικές κλίσης της <i>LeakyReLU</i>	60
Εικόνα 5.5 Έξοδοι για διαφορετικούς ρυθμούς εκπαίδευσης	60
Εικόνα 5.6 Γραφική Παράσταση Σφάλματος <i>KL</i> σε σχέση με τις επαναλήψεις.....	61
Εικόνα 5.7 Γραφική Παράσταση Σφάλματος <i>EMD</i> σε σχέση με τις επαναλήψεις	62
Εικόνα 5.8 Αποτελέσματα VAE σε επιπλέον κλάσεις αντικειμένων	62
Εικόνα 5.9 Αποτελέσματα με Κατάρρευση Λειτουργίας του GAN.....	63
Εικόνα 5.10 Διαγράμματα μετρικών εκπαίδευσης μοντέλου GAN.....	68
Εικόνα 5.11 Διάγραμμα Απόστασης Ελάχιστου Έργου.....	68
Εικόνα 5.12 Αποτελέσματα μοντέλου GAN, για την κλάση «καρέκλα»	69
Εικόνα 5.13 Αποτελέσματα μοντέλου GAN, για την κλάση «τραπέζι»	70
Εικόνα 5.14 Ποικιλομορφία αποτελεσμάτων GAN	71

Κατάλογος Συντομογραφιών

TN - AI	<i>Τεχνητή Νοημοσύνη - Artificial Intelligence</i>
ANI (en)	<i>Artificial Narrow Intelligence – Στενή Τεχνητή Νοημοσύνη</i>
AGI (en)	<i>Artificial General Intelligence – Γενική Τεχνητή Νοημοσύνη</i>
ASI (en)	<i>Artificial Superintelligence – Τεχνητή Υπερνοημοσύνη</i>
ML (en)	<i>Machine Learning – Μηχανική Μάθηση</i>
CV (en)	<i>Computer Vision - Οραση Υπολογιστών</i>
DL (en)	<i>Deep Learning - Βαθιά Μάθηση</i>
NN (en) ΝΔ	<i>Neural Network(s) – Νευρωνικά Δίκτυα</i>
SVM (en)	<i>Support Vector Machines – Μηχανές Διανυσμάτων Υποστήριξης</i>
BNNs (en)	<i>Bayesian Neural Networks – Μπαγιεσιανά Νευρωνικά Δίκτυα</i>
MPs (en)	<i>Multilayer Perceptrons – Πολυστρωματικά Αντίληπτρα</i>
RNNs (en)	<i>Recurrent Neural Networks – Επαναλαμβανόμενα Νευρωνικά Δίκτυα</i>
CNNs (en)	<i>Convolutional Neural Networks – Συνελκτικά Νευρωνικά Δίκτυα</i>
GNNs (en)	<i>Graph Neural Networks – Γραφικά Νευρωνικά Δίκτυα</i>
GCNs (en)	<i>Graph Convolutional Networks – Γραφικά Συνελκτικά Δίκτυα</i>
GANs (en) ΓΑΔ	<i>Generative Adversarial Networks – Γεννητικά Ανταγωνιστικά Δίκτυα</i>
KLD (en)	<i>Kullback-Leibler Divergence – Απόκλιση KL</i>
JSD (en)	<i>Jensen-Shannon Divergence – Απόκλιση JS</i>
DCGANs (en)	<i>Deep Convolutional GANs – Βαθιά Συνελκτικά ΓΑΔ</i>
WGANs (en)	<i>Wasserstein GANs</i>
cGANs (en)	<i>Conditional GANs – ΓΑΔ υπό όρους</i>
BiGANs (en)	<i>Bidirectional GANs – αμφίδρομα ΓΑΔ</i>
SGANs (en)	<i>Semi-Supervised GANs – Ημι-Επιβλεπόμενα ΓΑΔ</i>
InfoGANs (en)	<i>Information Maximizing GANs – ΓΑΔ μεγιστοποίησης πληροφοριών</i>
AC-GANs (en)	<i>Auxiliary Classifier GANs – Βοηθητικοί Ταξινομητές</i>
VAE (en)	<i>Variational Autoencoder – Μεταβλητοί Αυτοκωδικοποιητές</i>
ReLU	<i>Rectified Linear Unit</i>
MLT (en)	<i>Wavefront Material Template Library</i>
EMD (en)	<i>Earth Mover’s Distance – Απόσταση Ελάχιστου Έργου</i>
Adam (en)	<i>Adaptive Moment Estimation</i>

1. Εισαγωγή

Η λέξη **αντίληψη** (*perception*) αναφέρεται στην κατανόηση μιας πληροφορίας ή ενός περιβάλλοντος μέσω της οργάνωσης, της ταυτοποίησης και της ερμηνείας αισθητηριακών πληροφοριών. Στον άνθρωπο οι πληροφορίες αυτές προέρχονται από τα μάτια, τα αυτιά, τη μύτη, τη γλώσσα ή/και το δέρμα. Η κυρίαρχη αίσθηση του ανθρώπου, καθώς και αυτή που οι ερευνητές προσπαθούν να αντιγράψουν στον τομέα της Τεχνητής Νοημοσύνης, και πιο συγκεκριμένα στον τομέα της Όρασης Υπολογιστών, είναι η **οπτική αντίληψη** (*visual perception*). Η Όραση Υπολογιστών επιδιώκει να επιτρέψει σε ένα σύστημα υπολογιστή να μπορεί με αυτόματο τρόπο να **βλέπει**, να **αναγνωρίζει**, καθώς και να **κατανοεί** τον οπτικό κόσμο, προσομοιώνοντας τον τρόπο με τον οποίο το κάνει η ανθρώπινη όραση. Ενώ η γενική ιδέα φαίνεται αρκετά απλή, η τεχνολογία πίσω από την Όραση Υπολογιστών είναι περίπλοκη, και έτσι το χάσμα με την ανθρώπινη όραση είναι ακόμα αρκετά μεγάλο. Παρ' όλα αυτά, χάριν στους ερευνητές που προσπαθούν εδώ και χρόνια να αναπτύξουν αλγόριθμους για εργασίες οπτικής αντίληψης, σήμερα ο τομέας αυτός είναι πολύ καλά ανεπτυγμένος. Πλέον είναι εφικτή η ανίχνευση συγκεκριμένων αντικειμένων και κατηγοριών αντικειμένων σε ποικίλες συνθήκες, η αυτόνομη οδήγηση ενός οχήματος σε μέτριες ταχύτητες σε ανοιχτούς δρόμους χωρίς την παρέμβαση κάποιου ανθρώπου, ή η αυτόματη παρατήρηση και κατανόηση ανθρώπινων δραστηριοτήτων.

Παρά την ραγδαία ανάπτυξη του τομέα, ένα από τα θεμελιώδες προβλήματα στην όραση των υπολογιστών, είναι η οπτική αντίληψη των **τριδιάστατων σχημάτων**. Το βασικό ζήτημα του προβλήματος αυτού είναι ότι τα πρότυπα/πτυχές οπτικής διέγερσης (*optical stimulation patterns*) είναι εγγενώς ασαφή. Η κύρια πτυχή της οπτικής διέγερσης στην ανθρώπινη όραση, είναι η αξιοσημείωτη ικανότητα προσδιορισμού τριδιάστατων δομών αντικειμένων, μέσω των προτύπων φωτός που προβάλλονται στον αμφιβληστροειδή (*retina*). Επιπλέον υπάρχουν και άλλες πτυχές της οπτικής διέγερσης της ανθρώπινης όρασης, οι οποίες βοηθούν τον άνθρωπο να καταλάβει επιπλέον πληροφορίες όπως η **υφή**, η **σκίαση**, και η **κίνηση**. Ο άνθρωπος δεν χρειάζεται ούτε να δει τα πόδια μιας καρέκλας για να ξέρει πως είναι εκεί, ούτε να δει μια κούπα καφέ από την μια πλευρά, και να ξέρει πως από μια άλλη πλευρά πιθανόν να υπάρχει μια λαβή. Από την άλλη ένας υπολογιστής δεν είναι αρκετά νοήμων, ούτε έχει σαφή πρότυπα οπτικής διέγερσης για να καταλάβει τόσο εύκολα τι αντικείμενο του έχει δοθεί σαν είσοδο. Παρόλο που οι ερευνητές προσπαθούν να μιμηθούν τον ανθρώπινο αμφιβληστροειδή, και έχουν αναπτυχθεί πολλά υπολογιστικά μοντέλα για την οπτική αντί-

ληψη της τρισδιάστατης δομής, τα τρισδιάστατα σχήματα δεν χρησιμοποιούνται αρκετά συχνά, λόγω της έλλειψης καλής αναπαράστασης.

Οι **τρειςδιάστατες αναπαραστάσεις αντικειμένων**, αποτελούν τον πυρήνα πολλαπλών εφαρμογών όρασης υπολογιστών, ρομποτικής επαυξημένης πραγματικότητας (*Augmented Reality*), εικονικής πραγματικότητας (*Virtual Reality*), και ιατρικής. Το 2010, η Microsoft κυκλοφόρησε μία από τις πρώτες συσκευές εισόδου ανίχνευσης κίνησης (*motion sensing input devices*), και συγκεκριμένα αισθητήρες εύρους βασικών προϊόντων, γνωστές σε όλους ως Microsoft Kinect. Με αυτό το τρόπο έγινε η εισαγωγή της σάρωσης **RGB-D** η οποία άρχισε να αποκτά σιγά σιγά τεράστια δυναμική. Πρόκειται για έναν συγκεκριμένο τύπο συσκευών ανίχνευσης βάθους που λειτουργούν σε συνδυασμό με μια κάμερα RGB, οι οποίες είναι σε θέση να προσθέσουν στην εικόνα πληροφορίες για το βάθος σε βάση εικονοστοιχείου. Στην προσθήκη αυτή βοηθάει ο υπολογισμός της παραμόρφωσης ενός γνωστού μοτίβου υπέρυθρου φωτός (*IR - infrared*) που προβάλλεται στη σκηνή.

Βέβαια με τα χρόνια φάνηκε ότι ενώ τα αποτελέσματα των αισθητήρων RGB-D είναι οπτικά ελκυστικά για τον άνθρωπο, η χρησιμότητα τους σε εφαρμογές γραφικών υπολογιστών είναι σχεδόν μηδενική, διότι απέχουν από την υψηλή ποιότητα των τρισδιάστατων γραφικών που δημιουργούνται από καλλιτέχνες. Οι ληφθείσες τρισδιάστατες σαρώσεις υποφέρουν από αποφράξεις (*occlusions*), που έχουν ως αποτέλεσμα να προκύπτουν ελλιπή τρισδιάστατα μοντέλα. Μια λύση στο πρόβλημα αυτό είναι η **ολοκλήρωση τρισδιάστατων σχημάτων** (*3D Shape Completion*). Πρόκειται για το πρόβλημα της αναδόμησης ενός μεμονωμένου αντικειμένου από ένα πλήθος παρατηρήσεων γνωστής κατηγορίας αντικειμένων με δεδομένα μια ελλιπής/μερική παρατήρηση του σχήματος του.

1.1 Περιγραφή Προβλήματος και Στόχοι Εργασίας

Όλο και περισσότερα προβλήματα που έχουν να κάνουν με την επεξεργασία, την κατανόηση ή την χρήση των τρισδιάστατων μοντέλων αντικειμένων, απασχολούν την επιστημονική κοινότητα. Πολλές μέθοδοι Βαθιάς Μάθησης που χρησιμοποιούνται για τις συμβατικές δισδιάστατες εικόνες, αποτελούν τον πυρήνα δημιουργίας μεθόδων για τα τρισδιάστατα μοντέλα αντικειμένων. Με αυτή τη λογική πλέον υπάρχουν πολλές μέθοδοι για την ταξινόμηση (*classification*) και αναγνώριση (*recognition*) τρισδιάστατων αντικειμένων, την κατάτμηση (*segmentation*) των αντικειμένων σε μικρότερα αντικείμενα, που βασίζονται στις αντίστοιχες μεθόδους ταξινόμησης, αναγνώρισης και κατάτμησης αντικειμένων σε μια εικόνα. Ένα πρόβλημα που απασχόλησε την κοινότητα σχετικά πρόσφατα είναι αυτό της ολοκλήρω-

σης εικόνας, δηλαδή του γεμίσματος μιας κατεστραμμένης περιοχής σε μια δισδιάστατη εικόνα.

Η παρούσα διπλωματική εργασία εστιάζει συγκεκριμένα στο αντίστοιχο πρόβλημα ολοκλήρωσης εικόνας, αλλά για τρισδιάστατα αντικείμενα, δηλαδή στην ολοκλήρωση ελλειπών τρισδιάστατων σχημάτων. Στόχος της είναι η μελέτη και η κατανόηση τόσο των **Γεννητικών Ανταγωνιστικών Δικτύων (GANs)**, και των **Μεταβλητών Αυτοκωδικοποιητών (VAE)**, όσο και της φύσης των δεδομένων τρισδιάστατων σχημάτων. Επιπρόσθετα, είναι και η δημιουργία ενός απλού μοντέλου **Ολοκλήρωσης Τρισδιάστατων Σχημάτων**, το οποίο ενδεχομένως να αποτελέσει την βάση για την περαιτέρω μελλοντική μου έρευνα στο αντικείμενο αυτό.

1.2 Διάρθρωση Κεφαλαίων

Η παρούσα διπλωματική διαρθρώνεται σε 6 κεφάλαια. Το **πρώτο** είναι η εισαγωγή της εργασίας, με κάποιες βασικές πληροφορίες και απαντήσεις σε ερωτήματα όπως, «*Τι ακριβώς εννοούμε με την ολοκλήρωση σχημάτων;*», «*Σε τι διαφέρει η ανθρώπινη όραση, με την όραση υπολογιστών;*», ή «*Γιατί επιλέχθηκε το συγκεκριμένο πρόβλημα;*». Το **δεύτερο** κεφάλαιο αποτελεί το θεωρητικό υπόβαθρο που είναι απαραίτητο για να κατανοήσει κανείς καλύτερα το συγκεκριμένο πρόβλημα. Το κεφάλαιο χωρίζεται σε πολλές ενότητες, ξεκινώντας από βασικές έννοιες όπως η *Τεχνητή Νοημοσύνη*, η *Μηχανική Μάθηση* και γενικά τα *Νευρωνικά Δίκτυα*, συνεχίζοντας συγκεκριμένα με τα Δίκτυα που θα χρησιμοποιηθούν για το *πρακτικό* κομμάτι, και τέλος κάποιες βασικές έννοιες για την *Αναπαράσταση των Τρισδιάστατων Σχημάτων*.

Στο **τρίτο** κεφάλαιο γίνεται η παρουσίαση του Συνόλου Δεδομένων που χρησιμοποιείται, καθώς και μια συνοπτική ανάλυση μεθόδων που χρησιμοποιήθηκαν για την προεπεξεργασία των δεδομένων. Στο **τέταρτο** κεφάλαιο γίνεται η ανάλυση της μεθοδολογίας που ακολουθήθηκε για την επίλυση του συγκεκριμένου προβλήματος, η αναφορά σε σημαντικές παραμέτρους οι τιμές των οποίων άλλαζαν στα πειράματα, καθώς και η ανάλυση των μετρικών που χρησιμοποιήθηκαν κατά την διάρκεια της εκπαίδευσης. Το **πέμπτο** κεφάλαιο περιέχει τα διάφορα πειράματα που έγιναν, τόσο στο μοντέλο VAE, όσο στο μοντέλο GAN. Στο **έκτο** κεφάλαιο γίνεται μια σύνοψη των αποτελεσμάτων, αναγράφονται τα συμπεράσματα για την μεθοδολογία που χρησιμοποιήθηκε, καθώς και προτείνονται πιθανές μελλοντικές κατευθύνσεις έρευνας στο κομμάτι της ολοκλήρωσης σχημάτων.

2. Θεωρητικό Υπόβαθρο

2.1 Τεχνητή Νοημοσύνη

Στην αυγή της τρίτης χιλιετίας, κανείς πλέον δεν μπορεί να αμφισβητήσει ότι η **Τεχνητή Νοημοσύνη** (*Artificial Intelligence AI*) υπάρχει και επηρεάζει την καθημερινή μας ζωή, καθώς οι περισσότεροι άνθρωποι την συναντούν από το πρωί έως το βράδυ, από βασικά προϊόντα υπολογιστών μέχρι και οικιακά είδη, όπως για παράδειγμα τα ψυγεία, τα πλυντήρια ρούχων και οι ρομποτικές ηλεκτρικές σκούπες. Η σύγχρονη κοινωνία είναι μάρτυρας της αξιοσημείωτης επιτυχίας που επιφέρει η τεχνητή νοημοσύνη με τη μορφή μηχανικής μετάφρασης (*machine translation*), αναγνώρισης ομιλίας (*speech recognition*), ταξινόμησης εικόνας (*image classification*), ανάκτησης πληροφορίας (*information retrieval*), και όχι μόνο [LMS20].

Αν και απέκτησε τεράστια προσοχή τα τελευταία χρόνια λόγω της ραγδαίας ανάπτυξης της τεχνολογίας η έννοια της τεχνητής νοημοσύνης δεν είναι καινοφανής. Ο όρος επινοήθηκε από τον John McCarthy το 1955-1956 στο συνέδριο του κολλεγίου Dartmouth [C20], όμως οι αρχές της σύγχρονης τεχνητής νοημοσύνης μπορούν να εντοπιστούν πριν από χιλιετίες, στις προσπάθειες των φιλοσόφων να περιγράψουν την ανθρώπινη σκέψη ως ένα συμβολικό σύστημα [K03].

Όμως, τι ακριβώς είναι η Τεχνητή Νοημοσύνη; Η έννοια της TN εκφράζεται με πολλούς διαφορετικούς τρόπους ανάλογα τους παρατηρητές. Ουσιαστικά πρόκειται για έναν κλάδο της Επιστήμης της Πληροφορικής, με ισχυρές επιρροές από τη επιστήμη της ψυχολογίας, τη φιλοσοφία, τη γλωσσολογία καθώς και τα μαθηματικά [K03], που ασχολείται με την κατασκευή και ανάπτυξη ευφυών πρακτόρων (*intelligent agents*) ως προγραμμάτων υπολογιστών και την κατανόηση της συμπεριφοράς τους [SM01]. Ο γενικός στόχος της είναι να δημιουργήσει συστήματα που ενσωματώνουν ή παρουσιάζουν κάποια **νοημοσύνη**.

2.1.1 Νοημοσύνη

Ο όρος **νοημοσύνη** είναι ένα αμφιλεγόμενο θέμα τόσο στον τομέα της Ψυχολογίας, όσο στον τομέα της Επιστήμης της Πληροφορικής. Στον τομέα της Ψυχολογίας οι αντιλήψεις για το τι ακριβώς είναι οι νοημοσύνη τείνουν να υποδηλώνουν πως είναι η πνευματική ικανότητα του ανθρώπου να αποκτά, διατηρεί και να χρησιμοποιεί τη γνώση, να μπορεί να αναγνωρίζει προβλήματα στο περιβάλλον έτσι ώστε να χρησιμοποιεί τη γνώση που έχει ήδη λάβει για την αντιμετώπισή τους [JL17]. Περιλαμβάνει δηλαδή διάφορες νοητικές ικανότητες

όπως η ικανότητα μάθησης, ικανότητα απόκτηση και αποθήκευση γνώσεων, ικανότητα γενίκευσης και ικανότητα υπολογισμού [Z20].

Αντίστοιχα, στον τομέα της Επιστήμης της Πληροφορικής η τεχνητή νοημοσύνη διαιρείται σε τέσσερις υποκατηγορίες, **μαθησιακή νοημοσύνη** (*learning intelligence*), **γνωστική νοημοσύνη** (*cognitive intelligence*), **γενικευμένη νοημοσύνη** (*generalization intelligence*) και **υπολογιστική νοημοσύνη** (*computational intelligence*). Τα συστήματα TN τροφοδοτούνται από αλγόριθμους, που κάνουν χρήση τεχνικών όπως η **Μηχανική Μάθηση** (*Machine Learning ML*), **Βαθιά Μάθηση** (*Deep Learning DL*) καθώς και **κανόνων**. Μέσω της Μηχανικής Μάθησης τα συστήματα αυτά τροφοδοτούνται με δεδομένα υπολογιστή χρησιμοποιώντας στατιστικές τεχνικές, που τα βοηθούν να αποκτήσουν γνώση, και σταδιακά να βελτιώνονται στα καθήκοντά τους, χωρίς την ανάγκη να προγραμματιστούν ειδικά για να το κάνουν αυτό [K03].

2.1.2 Κατηγορίες TN

Υπάρχουν διάφοροι τρόποι με τους οποίους οι επιστήμονες κατηγοριοποιούν την Τεχνητή Νοημοσύνη. Ένας τρόπος βασίζεται στην ταξινόμηση της TN με βάση την ομοιότητα της με τον ανθρώπινο νου, την ικανότητα δηλαδή να σκέφτεται ή ακόμα και να αισθάνεται. Το πιο απλό επίπεδο απαιτεί μόνο τις πιο βασικές λειτουργίες ενός ανθρώπινου μυαλού, ενώ το πιο προχωρημένο επίπεδο απαιτεί λειτουργίες πάνω από αυτές που μπορεί να σκεφτεί και να κάνει ένας απλός άνθρωπος [G14A].

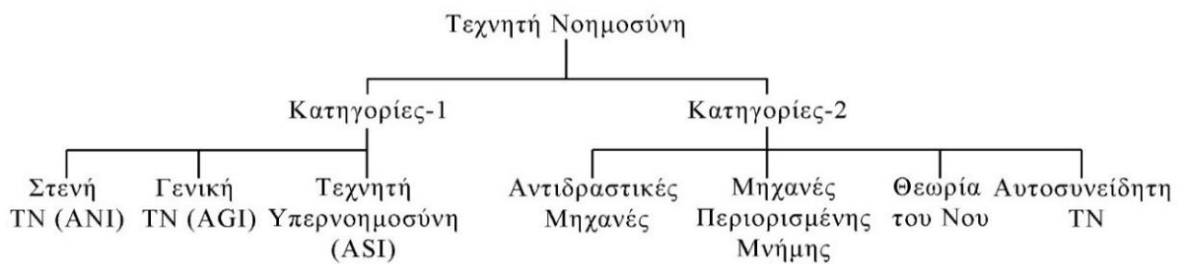
Επίπεδο Τεχνητής Νοημοσύνης	Λειτουργίες
Επίπεδο 1. Αντιδραστικές Μηχανές	Αντίδραση σε μια είσοδο με μια έξοδο, χωρίς το φαινόμενο της μάθησης.
Επίπεδο 2. Μηχανές Περιορισμένης Μνήμης	Αποθήκευση προηγούμενων δεδομένων ή/και προβλέψεων, για πραγματοποίηση καλύτερων προβλέψεων
Επίπεδο 3. Θεωρία του Νου	Αλληλεπίδραση σκέψεων και συναισθημάτων με τους ανθρώπους.
Επίπεδο 4. Αυτοσυνείδητη Τεχνητή Νοημοσύνη	Ανεξάρτητη νοημοσύνη με την οποία πιθανότατα ο άνθρωπος να πρέπει να διαπραγματεύεται.

Πίνακας 2.1. Τέσσερα επίπεδα Τεχνητής Νοημοσύνης με βάση τις λειτουργίες τους

Με βάση αυτό υπάρχουν τέσσερα διαφορετικά επίπεδα, το χαμηλότερο που είναι οι αντιδραστικές μηχανές (*reactive machines*), οι μηχανές περιορισμένης μνήμης (*limited memory machines*), η θεωρία του νου (*theory of mind*) και η αυτοσυνείδητη TN (*self-aware AI*) που είναι και το πιο προχωρημένο επίπεδο. Προς το παρόν, η TN έχει ξεπεράσει το πρώ-

το επίπεδο και βρίσκεται στο δεύτερο, με το τρίτο και το τέταρτο απλά να υπάρχουν ως θεωρίες [CMCS20] [SELF18] [G14A].

Δεδομένου ότι η έρευνα της TN αποσκοπεί την μίμηση της ανθρώπινης λειτουργίας από τις μηχανές, είναι απολύτως λογικό ότι ο βαθμός με τον οποίο το καταφέρνει αυτό χρησιμοποιείται ως ένα από τα κριτήρια προσδιορισμού των κατηγοριών της TN. Γι' αυτό το λόγο ένα εναλλακτικό σύστημα κατηγοριοποίησης, χωρίζει την TN σε τρία συστήματα – πραγματικά και υποθετικά, με βάση την ικανότητα του συστήματος να μιμηθεί ανθρώπινα χαρακτηριστικά. Τα συστήματα αυτά είναι η Στενή TN (*Narrow AI – ANI*) η οποία έχει ένα στενό φάσμα ικανοτήτων, η Γενική TN (*General AI – AGI*) της οποίας οι ικανότητες ισοδυναμούν με αυτές ενός μέσου ανθρώπου, καθώς και η Τεχνητή Υπερνοημοσύνη (*Superintelligence – ASI*) η οποία είναι πιο ικανή από έναν άνθρωπο, αλλά για την ώρα υποθετική [PBM18].



Εικόνα 2.1 Ταξινόμηση της Τεχνητής Νοημοσύνης. Υπάρχουν 2 διαφορετικές Κατηγορίες ταξινόμησης της TN, Κατηγορίες-1 δηλαδή με βάση την ικανότητα, και Κατηγορίες-2 δηλαδή με βάση την λειτουργικότητα.

2.2 Μηχανική Μάθηση

Η **Μηχανική Μάθηση** είναι ένα υποσύνολο της Τεχνητής Νοημοσύνης, που αναπτύσσει ένα μαθηματικό μοντέλο/δυναμικούς αλγόριθμους προκειμένου να αποκτά και να ενσωματώνει γνώση μέσω παρατηρήσεων, γνωστών ως “**δεδομένα εκπαίδευσης**”, έτσι ώστε να βελτιώνεται και να επεκτείνεται με την εκμάθηση των νέων γνώσεων [Z20]. Δημιουργεί δηλαδή ένα μοντέλο που λαμβάνει αποφάσεις αξιοποιώντας προηγούμενη γνώση και εμπειρία, και όχι ακολουθώντας οδηγίες στατικού προγραμματισμού. Ένας καλός ορισμός για το τι είναι η μάθηση δίνεται από τον T. M. Mitchell [M97], « Ένα πρόγραμμα υπολογιστή λέγεται ότι μαθαίνει από την εμπειρία *E* σε σχέση με κάποια κατηγορία εργασιών *T* και την απόδοση *P*, μόνο όταν η απόδοση *P* του σε εργασίες στο *T* βελτιώνεται με την εμπειρία *E*».

Μεγάλες εταιρείες όπως η Google, η Microsoft, η Amazon και πολλές άλλες, έχουν συμβάλει σε μεγάλο βαθμό στην ραγδαία αύξηση των σχετικών επιστημονικών δραστηριο-

τήτων την τελευταία δεκαετία [W09][GR18]. Οι επιστημονικές δραστηριότητες αυτές οργανώνονται σε τρεις κύριους ερευνητικούς τομείς [MCM13]:

- Μελέτες προσανατολισμένες στην εργασία (*Task-Oriented Studies*)

Έχουν ως στόχο την ανάπτυξη και ανάλυση συστημάτων (προγραμμάτων) για ένα προκαθορισμένο σύνολο μαθησιακών εργασιών, που θα βελτιώσουν την απόδοση και θα κάνουν τους υπολογιστές πιο ευφυείς.

- Γνωστική προσομοίωση (*Cognitive Simulation*)

Ο τομέας αυτός αφορά την διερεύνηση και ανάπτυξη υπολογιστικών μοντέλων της διαδικασίας ανθρώπινης μάθησης καθώς και την εκτέλεση προσομοιώσεων υπολογιστών.

- Θεωρητική Ανάλυση (*Theoretical Analysis*)

Στόχος του συγκεκριμένου τομέα είναι η διερεύνηση νέων μεθόδων μάθησης και η ανάπτυξη γενικών αλγορίθμων ανεξάρτητα από τις εφαρμογές.

Για να προσεγγίσουμε την βέλτιστη μάθηση χρειάζονται μεγάλα σύνολα δεδομένων. Συνήθως τα σύνολα αυτά χωρίζονται σε υποσύνολο εκπαίδευσης (*training set*) και υποσύνολο ελέγχου (*test set*). Το υποσύνολο εκπαίδευσης χωρίζεται σε δύο κομμάτια, το ένα με ετικέτες (*labeled set*) και το άλλο χωρίς ετικέτες (*unlabeled set*), ενώ το υποσύνολο ελέγχου αποτελείται από άγνωστα παραδείγματα [Z20]. Έστω ότι έχουμε ένα υποσύνολο εκπαίδευσης T . Το υποσύνολο με τις ετικέτες θα ισούται

$$T_l = X_l, Y_l = \{(x_1, y_1), \dots, (x_l, y_l)\} \quad (2.1)$$

, όπου στα προβλήματα παλινδρόμηση (*regression*) το y_i είναι η παλινδρόμηση ή προσαρμογή του x_i , ενώ στα προβλήματα ταξινόμησης το y_i ισούται με την αντίστοιχη ετικέτα κλάσης στην οποία ανήκει το x_i . Το υποσύνολο χωρίς ετικέτες θα ισούται με:

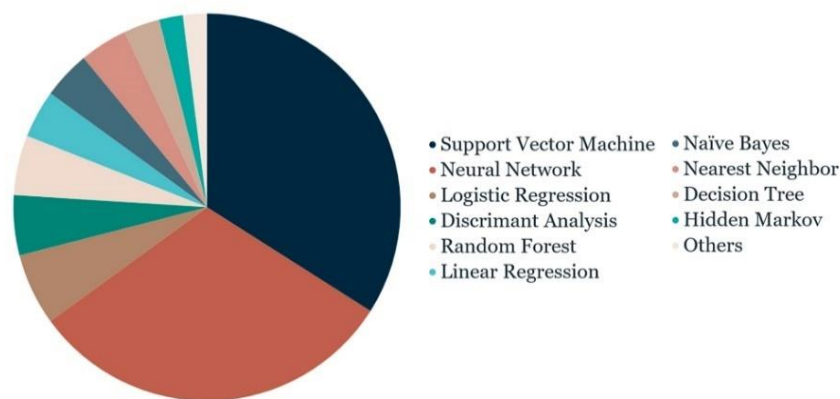
$$T_u = \{x_1, \dots, x_u\} \quad (2.2)$$

2.2.1 Κατηγορίες Μηχανικής Μάθησης

Με βάση τη φύση του υποσυνόλου εκπαίδευσης, έχουμε τις εξής κατηγορίες μηχανικής μάθησης, Ευκλείδεια Εκμάθηση Δεδομένων (*Regular or Euclidean structured data learning*) και Μη-Ευκλείδεια Εκμάθηση Δεδομένων ή Εκμάθηση Γράφων (*Graph machine learning*). Η Ευκλείδεια Εκμάθηση χωρίζεται σε πέντε επιμέρους κατηγορίες 1. Επιβλεπόμενη Μάθηση (*Supervised learning*), 2. Μη-επιβλεπόμενη Μάθηση (*Unsupervised learning*), 3. Ημι-επιβλεπόμενη Μάθηση (*Semi-supervised learning*), 4. Ενισχυτική Μάθηση (*Reinforcement learning*), και 5. Μεταφορική Μάθηση (*Transfer learning*) [C20] [Z20].

Ευκλείδεια Εκμάθηση Δεδομένων

- Επιβλεπόμενη Μάθηση : Η πιο θεμελιώδης μορφή μηχανικής μάθησης η οποία σχετίζεται με την ταξινόμηση και την παλινδρόμηση δεδομένων. Στην εκμάθηση υπό επίβλεψη εκπαιδεύουμε έναν αλγόριθμο μηχανικής μάθησης χρησιμοποιώντας δεδομένα, παράλληλα δίνοντας καθοδήγηση στο μοντέλο του αλγορίθμου με ετικέτες που σχετίζονται με τα δεδομένα [S15] [GBC16]. Έτσι μέσω ενός συνόλου δεδομένων με ετικέτες $X_{train}, Y_{train} = \{ x_1, y_1, \dots, x_l, y_l \}$ το μοντέλο θα μάθει έναν γενικό κανόνα που χαρτογραφεί τις εισόδους στις αντίστοιχες εξόδους [C20].



Εικόνα 2.2 Μοντέλα επιβλεπόμενη μάθησης. Στο γράφημα αυτό βλέπουμε διάφορα μοντέλα επιβλεπόμενης μάθησης. Παρατηρούμε πως τα δύο πιο δημοφιλή μοντέλα είναι οι Μηχανές Διαανυσμάτων Υποστήριξης (Support Vector Machines – SVM) και τα Νευρωνικά Δίκτυα (Neural Networks – NN)

- Μη-επιβλεπόμενη Μάθηση : Γνωστή και ως ανακάλυψη γνώσης, κάνει χρήση μη εκπαιδευμένων, μη ταξινομημένων και κατηγοριοποιημένων δεδομένων εκπαίδευσης. Προσπαθεί να κατανοήσει τα δεδομένα χωρίς ετικέτα $X_{train} = \{x_1, \dots, x_u\}$ [Z20] εξάγοντας χαρακτηριστικά και μοτίβα. Συνήθως χρησιμοποιείται η μέθοδος της ομαδοποίησης (*clustering*) για να ανακαλύψουμε τις διάφορες κλάσεις στα δεδομένα [C20].
- Ημι-επιβλεπόμενη Μάθηση : Μια κατηγορία τεχνικών που χρησιμοποιούν τόσο δεδομένα με ετικέτες όσο και μη επισημασμένα δεδομένα κατά την εκπαίδευση ενός μοντέλου. Για τα δεδομένα ισχύει ότι

$$X_{train}, Y_{train} = \{ x_1, y_1, \dots, x_l, y_l \} \cup \{ x_{l+1}, \dots, x_{l+u} \} \text{ με } l \ll u, \quad (2.3)$$

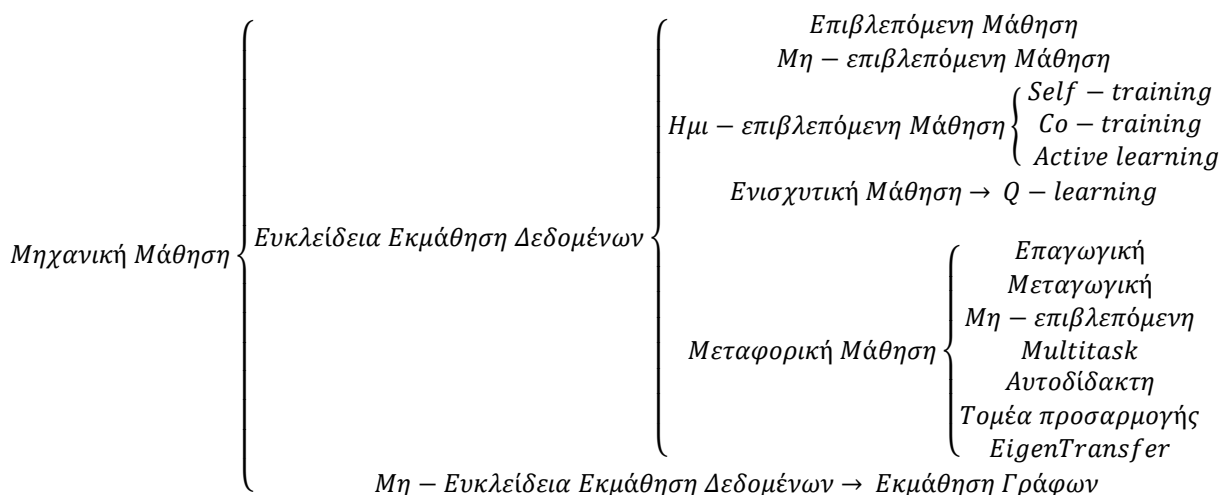
δηλαδή τα δεδομένα με ετικέτες είναι κατά πολύ λιγότερα από αυτά χωρίς ετικέτες [Z20]. Οι συγκεκριμένες τεχνικές χρησιμοποιούνται διότι ενώ τα δεδομένα είναι διαθέσιμα σε αφθονία, η απόκτηση ετικετών είναι δαπανηρή. Γι' αυτό το λόγο στην ημι-

επιβλεπόμενη μάθηση το μοντέλο μαθαίνει μοτίβα με τη χρήση των επισημασμένων δεδομένων και στην συνέχεια βελτιώνει τα όρια απόφασης μεταξύ των διαφόρων κλάσεων με τη χρήση των μη επισημασμένων δεδομένων [W14].

- **Ενισχυτική Μάθηση** : Είναι τύπος δυναμικού προγραμματισμού που εκπαιδεύει αλγόριθμους χρησιμοποιώντας ένα σύστημα ανταμοιβής και ποινής [Z20]. Αποτελείται από τρία βασικά συστατικά: α) τον πράκτορα εκμάθησης, β) το περιβάλλον με το οποίο αλληλεπιδρά ο πράκτορας, και γ) τις ενέργειες που εκτελούν οι πράκτορες. Μετά από λήψη πληροφορίας για την τρέχουσα κατάσταση του περιβάλλοντος ο πράκτορας εκτελεί κάποιες ενέργειες με σκοπό να αλλάξει αυτή την κατάσταση. Για κάθε ενέργεια ή σύνολο ενεργειών που εκτελεί ο πράκτορας λαμβάνει θετική ή αρνητική ανταμοιβή [KK07].
- **Μεταφορική Μάθηση** : Είναι η επαναχρησιμοποίηση ενός προ-εκπαιδευμένου μοντέλου για ένα νέο πρόβλημα το οποίο μοιάζει με το αρχικό. Αυτή η προσέγγιση είναι πολύ δημοφιλής στη βαθιά μάθηση, μιας και μπορεί να εκπαιδεύσει νευρωνικά δίκτυα με πολύ λιγότερα δεδομένα [Z20].

Εκμάθηση Γράφων

Πολλές νέες εφαρμογές μηχανικής μάθησης επιδιώκουν την χρήση δομημένων δεδομένων γράφου ως πληροφορίες χαρακτηριστικών για την πρόβλεψη και την ανακάλυψη νέων μοτίβων. Αυτό γίνεται διότι οι γράφοι μπορούν να αντιπροσωπεύουν πολύπλοκα σχεσιακά δεδομένα μιας και αποτελούνται από κόμβους και άκρα) [HYL17].



Εικόνα 2.3 Κατηγορίες Μηχανικής Μάθησης

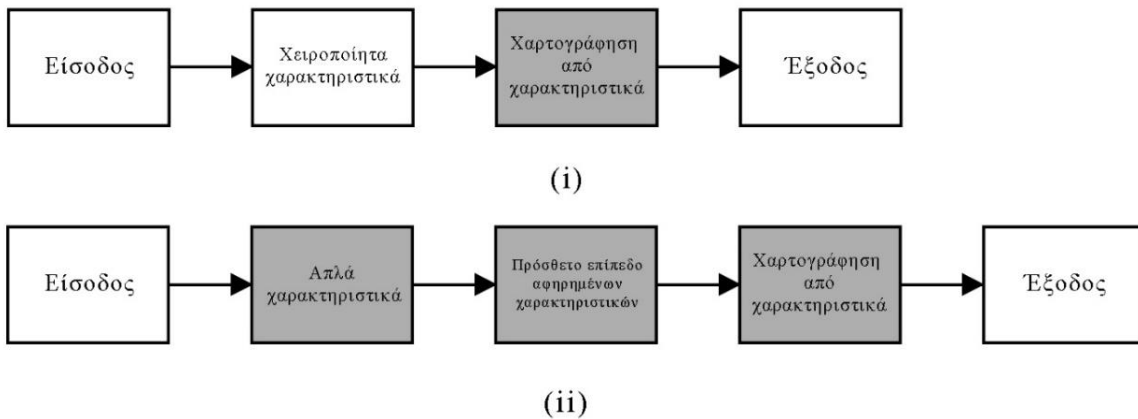
2.2.2 Βαθιά Μάθηση

Τα τελευταία χρόνια με την άνθηση του ενδιαφέροντος στην μηχανική μάθηση, επωφελήθηκαν κλάδοι όπως η **αναγνώριση ομιλίας** (*Speech Recognition*), η **όραση υπολογιστών** (*Computer Vision CV*) και η **επεξεργασία φυσική γλώσσα** (*Natural Language Processing NLP*), από τις λεγόμενες προσεγγίσεις **Βαθιάς Μάθησης** (*Deep Learning DL*) [LBH15]. Η Βαθιά Μάθηση, αναφέρεται συχνά και ως ιεραρχική μάθηση ή βαθιά δομημένη μάθηση, είναι μια υποκατηγορία της Μηχανικής Μάθησης. Είναι ο τομέας ο οποίος έχει ως στόχο να οδηγήσει την μηχανική μάθηση πιο κοντά σε έναν από τους αρχικούς της στόχους, δηλαδή την τεχνητή νοημοσύνη [LZLZ16]. Βασίζεται σε αλγορίθμους πολλών επιπέδων για την **εξαγωγή σύνθετων και εξεζητημένων χαρακτηριστικών** (*feature extraction*), έτσι ώστε να γίνει μοντελοποίηση σύνθετων σχέσεων μεταξύ δεδομένων, τα οποία μπορεί να είναι εικόνες, ήχος και κείμενο [A18][LZLZ16]. Για να ανακαλύψει τις περίπλοκες δομές σε μεγάλα σύνολα δεδομένων η βαθιά μάθηση χρησιμοποιεί τον αλγόριθμο **ανάστροφης διάδοσης** (*backpropagation algorithm*) έτσι ώστε να προσαρμόσει τις εσωτερικές παραμέτρους κάθε επιπέδου με βάση το προηγούμενο επίπεδο [LBH15].

Τόσο η μηχανική μάθηση όσο και η βαθιά μάθηση βοηθούν στην ανακάλυψη προτύπων στα δεδομένα, αλλά το κάνουν με πολύ διαφορετικούς τρόπους. Η κλασική μηχανική μάθηση εξαρτάται από την ανθρώπινη παρέμβαση, απαιτώντας επισημασμένα σύνολα δεδομένων για την καλή κατανόηση διαφορών μεταξύ των δεδομένων εισόδου. Η βασική διαφοροποίηση της βαθιάς μάθησης είναι ότι στέλνει την είσοδο σε ένα σύνολο από διαφορετικές στρώσεις του δικτύου, με κάθε δίκτυο να καθορίζει ιεραρχικά συγκεκριμένα χαρακτηριστικά των δεδομένων εισόδου.

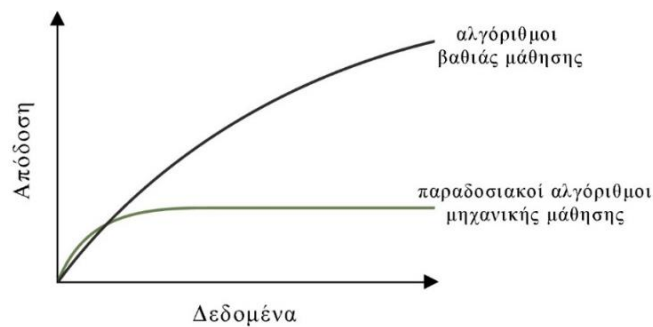
Μηχανική Μάθηση	Βαθιά Μάθηση
Δομημένα δεδομένα	Πολύπλοκες και μεγάλες ποσότητες δεδομένων
Χωρίς ειδικές δυνατότητες υλικού	Ειδικές δυνατότητες υλικού
Πιο γρήγορος χρόνος εκτέλεσης	Μεγαλύτερη διάρκεια
Εύκολη ερμηνεία	Δύσκολη ερμηνεία
Καλά αποτελέσματα για λιγότερα δεδομένα	Καλά αποτελέσματα για περισσότερα δεδομένα

Πίνακας 2.2. Διαφορές μεταξύ Μηχανικής και Βαθιάς Μάθησης



Εικόνα 2.4 Διάγραμμα ροής Μηχανικής και Βαθιάς Μάθησης. Στην παραπάνω εικόνα απεικονίζονται δύο διαγράμματα ροής (i) Μηχανική Μάθηση, και (ii) Βαθιά Μάθηση, τα οποία δείχνουν πως σχετίζονται αλλά και πως διαφέρουν μεταξύ τους. Τα σκιασμένα πλαίσια υποδεικνύουν στοιχεία που μπορούν να μάθουν από δεδομένα. Στην Μηχανική Μάθηση έχουμε παρέμβαση από ήδη υπάρχοντα χαρακτηριστικά, ενώ στην Βαθιά Μάθηση το μοντέλο μαθαίνει τα χαρακτηριστικά μόνο του από την αρχή.

Επιπρόσθετα, μια από τις σημαντικότερες διαφορές μεταξύ των δύο είναι η απόδοση των αλγορίθμων. Ένα κρίσιμο συστατικό στην απόδοση των μοντέλων Βαθιάς Μάθησης είναι οι μεγάλες ποσότητες δεδομένων. Με όσο περισσότερα δεδομένα ποτίσουμε το μοντέλο για να μπορεί να εκπαιδευτεί τόσο καλύτερα θα γίνονται τα αποτελέσματα. Από την άλλη στους αλγόριθμους Μηχανικής Μάθησης λόγω των χειροποίητων κανόνων που υπάρχουν, αν ξεπεραστεί ένα όριο δεδομένων εισόδου οι κανόνες αυτοί καταρρέουν μαζί με την απόδοση του μοντέλου.



Εικόνα 2.5 Σχέση απόδοσης και Δεδομένων.

2.2.2.1 Τανυστές

Όλο και περισσότερα προβλήματα Βαθιάς Μάθησης κάνουν χρήση πολύπλοκων πολυδιάστατων δεδομένων. Για την κωδικοποίηση αυτών των πολυδιάστατων δεδομένων, τα μοντέλα κάνουν χρήση των **τανυστών** (*tensors*). Οι τανυστές είναι μαθηματικά αντικείμενα που γενικεύουν μονόμετρα μεγέθη (*scalars*), διανύσματα (*vectors*) και πίνακες (*matrices*) σε υψηλότερες διαστάσεις. Πρόκειται για μια σειρά αριθμών διατεταγμένων σε κανονικό πλέγμα με μεταβλητό αριθμό αξόνων [GBC16].

Ορισμός 2.1. Ένας τανυστής που περιγράφεται από μια συστοιχία N -διαστάσεων ονομάζεται τανυστής N -τάξης και ορίζεται ως μια πολυγραμμική συνάρτηση σε έναν Καρτεσιανό διανυσματικό χώρο N -διαστάσεων, $\mathcal{T} \in \mathbb{K}^{I_1 \times I_2 \times \dots \times I_N}$, όπου \mathbb{K} δηλώνει είτε το πραγματικό πεδίο \mathbb{R} είτε το φανταστικό πεδίο \mathbb{C} , και I_N είναι ο αριθμός καταχωρήσεων ή η διάσταση της n -οστής κατεύθυνσης [Z20]. Έστω ότι έχουμε έναν τανυστή \mathcal{A} . Το στοιχείο του τανυστή \mathcal{A} στις συντεταγμένες k, l, m γράφεται ως $A_{k,l,m}$ [GBC16].

2.3 Νευρωνικά Δίκτυα

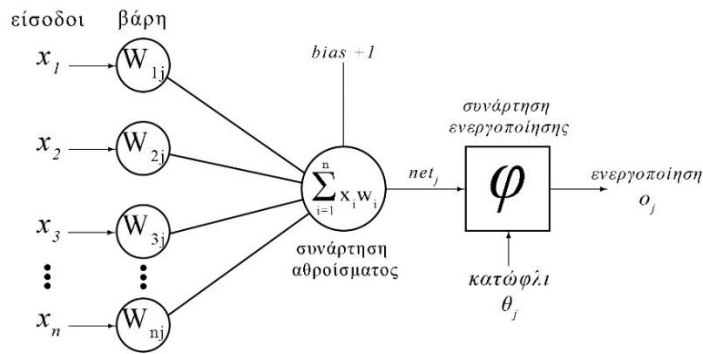
Τα **Νευρωνικά Δίκτυα** (*Neural Networks NN*), είναι μαθηματικοί αλγόριθμοι διαμορφωμένοι σύμφωνα με την αρχιτεκτονική των βιολογικών νευρωνικών δικτύων, σχεδιασμένοι να αναγνωρίζουν μοτίβα και σχέσεις μεταξύ δεδομένων, με βάση οργανωτικών αρχών που χρησιμοποιούν οι άνθρωποι. Αποτελούνται από τεράστιες ποσότητες διασυνδεδεμένων υπολογιστικών στοιχείων που ονομάζονται νευρώνες με αριθμητικά βάρη w_i που συντονίζονται και προσαρμόζονται στις διάφορες εισόδους [LZLZ16]. Ο μη παραμετρικός χαρακτήρας των νευρωνικών δικτύων επιτρέπει την ανάπτυξη μοντέλων χωρίς κάποια προηγούμενη γνώση για τα δεδομένα [C20]. Ενώ τα ΝΔ πλέον χρησιμοποιούνται για προβλήματα αναγνώρισης προτύπων, πρόβλεψης χρονοσειρών, επικύρωση δεδομένων, ανίχνευση ανωμαλιών, διαχείριση κινδύνων, υπάρχουν πολλοί περιορισμοί όταν τα προβλήματα είναι πολύ πιο περίπλοκα. Η διαδικασία δημιουργίας ενός αξιοπρεπούς ΝΔ είναι χρονοβόρα, απαιτεί τεράστιο όγκο δεδομένων εκπαίδευσης, καθώς και σωστό συντονισμό υπερπαραμέτρων. Επιπλέον τα ΝΔ είναι επιρρεπείς στην υπερπροσαρμογή και εμφανίζουν δυσκολίες στην αναγνώριση περίπλοκων σχέσεων μεταξύ δεδομένων [LZLZ16].

2.3.1 Τεχνητός Νευρώνας

Κάθε ΝΔ αποτελείται από διασυνδεδεμένες μονάδες που ονομάζονται νευρώνες. Κάθε νευρώνας αποτελείται από N αριθμό εισόδων x_i και μία έξοδος $h_{w,b}$.

$$h_{w,b} = f(W^T x + b) = f\left(\sum_{i=1}^n W_i x_i + b\right) \quad x, W \in \mathbb{R}^n, b \in \mathbb{R} \quad (2.4)$$

Οι εισοδοί, πολλαπλασιασμένες η κάθε μία με το αντίστοιχο βάρος της w_i , αθροίζονται μέσα στη συνάρτηση αθροίσματος (*summation function*). Πολλές φορές υπάρχει μια επιπλέον είσοδο b (*bias*) η οποία έχει σταθερή τιμή $+1$. Για να δώσει έξοδο ο νευρώνας μέσω της **συνάρτησης ενεργοποίησης** (*activation function*) το άθροισμα που προέκυψε από τις εισόδους

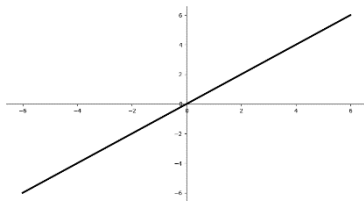


Εικόνα 2.6 Τεχνητός Νευρώνας. Κάθε είσοδος x_I έχει και ένα σχετικό βάρος x_i . Το άθροισμα των εισόδων με τα βάρη (συνάρτηση μεταφοράς) περνάει από μια συνάρτηση ενεργοποίησης (συνήθως μη-γραμμική).

πρέπει να είναι μεγαλύτερο από μια **τιμή κατωφλίου** (*threshold* θ) [A18].

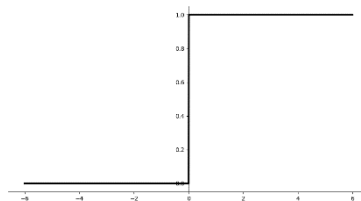
$$\sum_{i=1}^n W_i x_i + b - \theta > 0 \quad (2.5)$$

Η επιλογή σωστής συνάρτησης ενεργοποίησης είναι ένα πολύ κρίσιμο μέρος του σχεδιασμού νευρωνικών δικτύων. Οι συναρτήσεις αυτές καθορίζουν όρια για την έξοδο των νευρώνων. Στην Εικόνα 2.9 βλέπουμε 6 από τις πιο συχνά χρησιμοποιούμενες συναρτήσεις ενεργοποίησης.



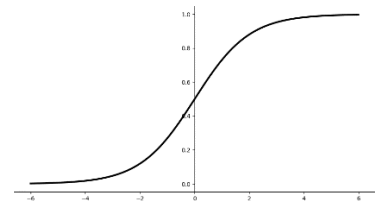
(i) Γραμμική Συνάρτηση

$$\phi(v) = v$$



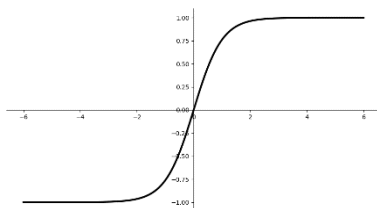
(ii) Βηματική Συνάρτηση

$$\phi(v) = \begin{cases} 1, & \text{αν } v \geq 0 \\ 0, & \text{αλλιώς} \end{cases}$$



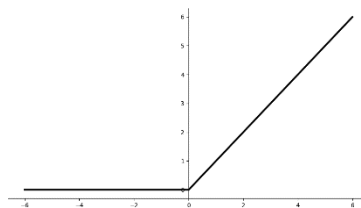
(iii) Σιγμοειδής Συνάρτηση

$$\phi(v) = \frac{1}{1 + e^{-v}}$$



(iv) Υπερβολική Συνάρτηση [tanh]

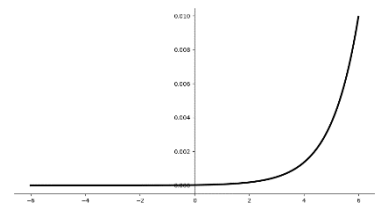
$$\phi(v) = \frac{e^{2v} - 1}{e^{2v} + 1}$$



(v) Μονάδα Γραμμικού

Ανορθωτή (RELU)

$$\phi(v) = \max\{0, v\}$$



(vi) Συνάρτηση Softmax

$$\phi(i) = \frac{e^{z_i}}{\sum_{j \in \text{ομάδα}} e^{z_j}}$$

$$\frac{\partial \phi_i}{\partial z_i} = \phi_i(1 - \phi_i)$$

Εικόνα 2.7 Συναρτήσεις ενεργοποίησης.

Όταν ξεκίνησαν να χρησιμοποιούνται τα ΝΔ η Σιγμοειδής συνάρτηση ήταν δημοφιλής, όμως με το καιρό προτιμήθηκε η Υπερβολική συνάρτηση λόγω της σταθερής κατάστασής της στο 0 (που είναι το μηδέν). Πρόσφατα όμως η Μονάδα Γραμμικού Ανορθωτή φάνηκε να αποδίδει πολύ καλύτερα από τις άλλες δύο σε διαφορετικές ρυθμίσεις. Η Συνάρτηση Softmax ενώ μοιάζει με την Σιγμοειδής συνάρτηση, χρησιμοποιείται στην πολύ-ταξινόμηση (*multi-classification*) σε αντίθεση με την Σιγμοειδή που χρησιμοποιείται στην δυαδική ταξινόμηση (*binary classification*) [Z20] [GBC16].

2.3.2 Κατηγορίες

Όπως και στην Μηχανική Μάθηση έτσι και στα Νευρωνικά Δίκτυα υπάρχουν δύο ευρείες κατηγορίες Νευρωνικών Δικτύων, (i) Ευκλείδεια Εκμάθηση Δεδομένων (*Euclidean Structured Model Learning*), και (ii) Μη-Ευκλείδεια Εκμάθηση Δεδομένων (*Non-Euclidean Structured Model Learning*).

Ευκλείδεια Εκμάθηση Δεδομένων

- Νευρωνικά Δίκτυα για Στοχαστική (πιθανοτική) εκμάθηση μοντέλων (*Neural networks for stochastic (probabilistic) model learning*)
 - Μπαγιεσιανά ΝΔ (*Bayesian neural networks BNNs*): Έχουν καθιερωθεί ως το πανταχού παρών εργαλείο μοντελοποίησης και συλλογισμού υπό αβεβαιότητα, η οποία απεικονίζεται ως κατανομή πιθανότητας μέσω των παραμέτρων της. Βασικός στόχος τους είναι η ελαχιστοποίηση της σχετικής εντροπίας [Z20].
 - Μηχανή Boltzmann (*Boltzmann machine*): Αποτελούνται από δύο στρώματα, το ορατό (*visible layer*) και το κρυφό (*hidden layer*). Το ορατό στρώμα δέχεται μια είσοδο n διαστάσεων $x = (x^{(1)}, \dots, x^{(n)})^T$, από την οποία προκύπτει μια κρυφή μεταβλητή m διαστάσεων $z = (z^{(1)}, \dots, z^{(m)})^T \in \{0,1\}^m$ (κρυφό στρώμα). Το ορατό στρώμα με το κρυφό στρώμα είναι πλήρως συνδεδεμένα μεταξύ τους [A18] [Z20].
 - Γεννητικά Ανταγωνιστικά Δίκτυα (*Generative Adversarial Networks GANs*): Τα GAN έχουν την ικανότητα να δημιουργήσουν πιστό ψεύτικο περιεχόμενο μαθαίνοντας αρκετά αποτελεσματικά την κατανομή των δεδομένων. Αποτελούνται από έναν Παραγωγό G (*Generator*) και έναν Διευκρινιστή D (*Discriminator*) [Z20]. Ο Παραγωγός λαμβάνει ως είσοδο έναν λανθάνοντα χώρο πιθανοτήτων και τον χαρτογραφεί σε ένα δείγμα που ανήκει στην κατανομή του συνόλου δεδομένων. Ο Διευκρινιστής λαμβάνει δείγματα και αποφασίζει ποια δεδομένα που έλαβε είναι

αληθινά και ποια συνθετικά, φτιαγμένα από τον Παραγωγό [ZYZZ20][GAN14]. Τα δύο δίκτυα αυτά ανταγωνίζονται μεταξύ τους και βελτιώνονται κατά την διάρκεια της αντίπαλης προπόνησης αυτής μέχρι να φτάσουν σε μια ισορροπία.

- Νευρωνικά Δίκτυα για Ντετερμινιστική Εκμάθηση Μοντέλου (*Neural networks for deterministic model learning*)
 - Πολυστρωματικά Αντίληπτρα (*Multilayer Perceptrons MLPs*): Είναι γνωστά και ως Νευρωνικά Δίκτυα πρόσθιας τροφοδότησης (*Feed Forward Neural Networks*). Το δίκτυο αυτό καθορίζει μια χαρτογράφηση $y = f(x; \theta)$ και μαθαίνει την τιμή των παραμέτρων θ το οποίο έχει ως αποτέλεσμα μια καλύτερη προσέγγιση της συνάρτησης. [GBC16]. Ονομάζεται έτσι διότι τα στρώματα τροφοδοτούνται το ένα από το άλλο σε εμπρόσθια κατεύθυνση από την είσοδο στην έξοδο [Z20].
 - Επαναλαμβανόμενα ΝΔ (*Recurrent Neural Networks RNNs*): Είναι η οικογένεια ΝΔ που χρησιμοποιείται κατά κύριο λόγο στην επεξεργασία διαδοχικών δεδομένων (*sequential data*) [A18]. Αυτό οφείλεται στο γεγονός πως κάθε στρώμα χρησιμοποιεί όχι μόνο την τρέχουσα τιμή εισόδου αλλά και τις προηγούμενες τιμές εξόδου [Z20]. Η διαδικασία μεταφοράς μνήμης προς τα μπρος αντιπροσωπεύεται μαθηματικά ως $h_t = f(W_h x_t + U_h h_{t-1} + b_h)$, όπου h_t είναι η τρέχουσα κατάσταση (κρυφό στρώμα), $W_h x_t$ είναι η είσοδος του στρώματος πολλαπλασιασμένη με το βάρος, και $U_h h_{t-1}$ είναι το προηγούμενο στρώμα και b_h είναι το bias [WFHP17].
 - Συνελκτικά ΝΔ (*Convolutional Neural Networks CNNs*): Είναι η οικογένεια ΝΔ που χρησιμοποιείται για την επεξεργασία δεδομένων με γνωστή τοπολογία πλέγματος, π.χ. εικόνες [GBC16]. Έστω ότι έχουμε μια εικόνα. Αυτή χωρίζεται σε μικρότερα κομμάτια και υποβάλλεται σε επαναλαμβανόμενες φάσεις συνελκτικού φιλτραρίσματος (*convolutional filters*), και μείωσης διαστάσεων-υποδειγματοληψίας (*pooling*), για να περάσει στην συνέχεια σε ένα πλήρως συνδεδεμένο πολυστρωματικό μη-συνελκτικό perceptron έτσι ώστε να βγει η τελική πρόβλεψη [WFHP17].
 - Αυτοκωδικοποιητής (*Autoencoder*): Είναι ένα συμμετρικό MLP το οποίο μαθαίνει μια αποτελεσματική κωδικοποίηση της εισόδου του [Z20]. Στόχος είναι η δημιουργία εξόδου όσο πιο κοντινής γίνεται με την είσοδο με μόνη διαφορά την μειωμένη διάσταση της [WFHP17]. Αποτελείται από δύο μέρη, τον κωδικοποιητή (*encoder*) και τον αποκωδικοποιητή (*decoder*). Η έξοδος του κωδικοποιητή είναι μια μειωμένη αναπαράσταση των δεδομένων εισόδου, και ορίζεται ως εξής :

$$f: x \in \mathbb{R}^l \mapsto h \in \mathbb{R}^m \text{ με } h_i := f_i(x) = \phi_e(\theta_i^T x + b_i), i = 1, 2, \dots, m \quad (2.6)$$

όπου ϕ_e είναι η συνάρτηση ενεργοποίησης του, συνήθως σιγμοειδής. Η έξοδος του αποκωδικοποιητή ορίζεται ως εξής :

$$g: h \in \mathbb{R}^m \mapsto x \in \mathbb{R}^l, \text{ με } h_j: = g_j(x) = \phi_d(\theta_j^T h + b_j), \quad j = 1, 2, \dots, l \quad (2.7)$$

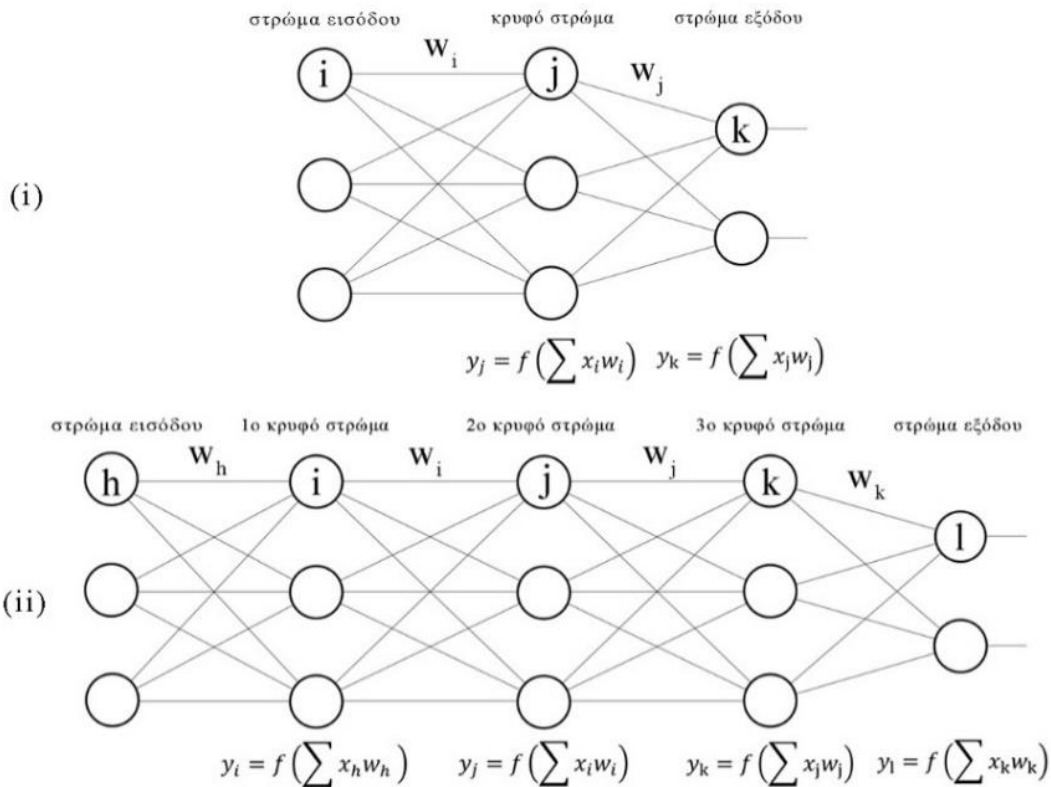
όπου η συνάρτηση ενεργοποίησης ϕ_d συνήθως είναι γραμμική ή σιγμοειδής [T20].

Μη-Ευκλείδεια Εκμάθηση Δεδομένων

- Ενσωμάτωση γράφων (*Graph embedding*): Στόχος του είναι να αντιπροσωπεύει έναν γράφο ως διάνυσμα ή σύνολο διανυσμάτων μικρών διαστάσεων, διατηρώντας όμως τις δομές του γράφου [Z20]. Υπάρχουν τέσσερις κατηγορίες γράφων που χρησιμοποιούνται ως είσοδος, 1) ομοιογενείς γράφοι (*homogeneous graphs*), 2) ετερογενείς γράφοι (*heterogeneous graphs*), 3) γράφοι με βοηθητικές πληροφορίες (*graphs with auxiliary information*), και 4) γράφοι σχεδιασμένοι από μη σχεσιακά δεδομένα (*graphs constructed from non-relational data*) [CZC18].
- Ενσωμάτωση δικτύου (*Network embedding*): Είναι η προσέγγιση της εκμάθησης αναπαράστασεων χαρακτηριστικών μικρών διαστάσεων των κόμβων ή των συνδέσμων σε ένα δίκτυο. Η βασική αρχή είναι η εκμάθηση κωδικοποιήσεων για τους κόμβους στο δίκτυο έτσι ώστε η ομοιότητα στο χώρο ενσωμάτωσης (*embedding space*) να αντικατοπτρίζει την ομοιότητα στο δίκτυο $f: v_i \rightarrow y_i \in \mathbb{R}^d$ [Z20].
- ΝΔ πάνω σε γράφους (*Neural networks on Graphs*): Πρόκειται για μοντέλα σύνδεσης που καταγράφουν τις εξαρτήσεις των γράφων μέσω μηνύματος που περνά μεταξύ των κόμβων του κάθε γράφου. Χωρίζονται σε ημι-Επιβλεπόμενα Δίκτυα (Γραφικά Νευρωνικά Δίκτυα (*Graph Neural Networks GNNs*), και Γραφικά Συνελικτικά Δίκτυα (*Graph Convolutional Networks GCNs*)), και μη-Επιβλεπόμενα Δίκτυα (Γραφικοί Αυτόματοι Κωδικοποιητές (*Graph Autoencoders*)) [GNN18].

2.3.3 Βαθιά Νευρωνικά Δίκτυα

Τα “Νευρωνικά Δίκτυα” και η “Βαθιά Μάθηση” είναι δύο όροι οι οποίοι συχνά χρησιμοποιούνται εναλλακτικά, κάτι που δεν είναι ορθό μιας και δεν είναι όλα τα νευρωνικά δίκτυα βαθιά (*πολυστρωματικά*) και δεν είναι όλες οι αρχιτεκτονικές βαθιάς μάθησης νευρωνικά δίκτυα. Ενώ τα κλασσικά ΝΔ αποτελούνται από πολλούς απλούς, συνδεδεμένους νευρώνες, τα μοντέρνα ΝΔ χωρίζονται σε δύο κατηγορίες [Z20]. Όπως βλέπουμε στην Εικόνα 2.10 έχουμε τα **Ρηχά ΝΔ** (*Shallow Neural Networks*) και τα **Πολυστρωματικά ή Βαθιά ΝΔ** (*Multilayer/Deep Neural Networks*).



Εικόνα 2.8 Δομή Νευρωνικών Δικτύων. (i) Παράδειγμα ενός Ρηχού Νευρωνικού Δικτύου (Shallow Neural Network). Αποτελούνται το πολύ από 2 κρυφά στρώματα. (ii) Παράδειγμα ενός Πολυστρωματικού Νευρωνικού Δικτύου πρόσω Τροφοδότησης (Feedforward Multilayer Neural Network) δύο κλάσεων, στο οποίο κάθε κόμβος σε ένα στρώμα συνδέεται με όλους τους νευρώνες στο επόμενο στρώμα (πλήρως συνδεδεμένο δίκτυο).

Στα βαθιά ΝΔ οι υπολογισμοί εκτελούνται από πολλά στρώματα. Τα πρόσθετα στρώματα ανάμεσα στο στρώμα εισόδου και εξόδου αναφέρονται ως κρυφά στρώματα επειδή οι υπολογισμοί που εκτελούνται δεν είναι ορατοί στον χρήστη. Έστω ότι $h^l \ x$ είναι η έξοδος από κάθε κρυφό στρώμα τότε για L κρυφά στρώματα η έξοδος του δικτύου $f(x)$ θα είναι ίση με :

$$f(x) = f[a^{L+1} (h^L (a^L (\dots (h^2 (a^2 (h^1 (a^1 (x)))))))] \quad [A18] \quad (2.8)$$

Για να θεωρηθεί ένα ΝΔ πολυστρωματικό πρέπει να αποτελείται τουλάχιστον από 5 στρώματα (ένα στρώμα εισόδου, ένα στρώμα εξόδου, και τουλάχιστον 3 κρυφά στρώματα). Σε διαφορετική περίπτωση το ΝΔ θεωρείται ρηχό (ένα στρώμα εισόδου, ένα στρώμα εξόδου, και το πολύ 2 κρυφά στρώματα). Τα ρηχά ΝΔ μπορούν να καταγράψουν μόνο πολύ βασικά χαρακτηριστικά των δεδομένων, ενώ τα βαθιά ΝΔ παρέχουν βαθιά κατανόηση των χαρακτηριστικών των δεδομένων εκπαίδευσης.

Στα πολυστρωματικά ΝΔ το διάνυσμα εισόδου N διαστάσεων x μετατρέπεται στις εξόδους με τις ακόλουθες εξισώσεις [A18]:

$$\begin{aligned} h_1 &= \Phi(W_1^T x) && [\text{Στρώμα Εισόδου προς Κρυφό Στρώμα}] \\ h_{p+1} &= \Phi(W_{p+1}^T h_p) \quad \forall p \{1 \dots k-1\} && [\text{Κρυφό Στρώμα προς Κρυφό Στρώμα}] \\ o &= \Phi(W_{k+1}^T h_k) && [\text{Κρυφό Στρώμα προς Στρώμα Εξόδου}] \end{aligned}$$

2.3.4 Βασικοί Όροι

2.3.4.1 Μεταβλητότητα

Ένα μοντέλο λέγεται πως εμφανίζει **υψηλή μεταβλητότητα** (variance) όταν δίνει μεγαλύτερη βαρύτητα στα δεδομένα εκπαίδευσης, αλλά αποδίδει άσχημα στα **δεδομένα αντεπικύρωσης** (*cross-validation data*). Το σφάλμα αυτό εμφανίζεται λόγω της ευαισθησίας του μοντέλου σε μικρές αυξομειώσεις κατά της διάρκεια της εκπαίδευσης. Αυτό συνεπάγεται της **υπερεκπαίδευσης** του μοντέλου (*overfitting*). Σύμφωνα με τους Gudivada & Rao [GR18] «Πιθανές λύσεις για την μείωση της υψηλής μεταβλητότητα είναι i) η αύξηση δεδομένων εκπαίδευσης, ii) η χρήση μικρότερων χαρακτηριστικών (μείωση στρώσεων (*layers*) στο νευρωνικό δίκτυο), iii) η πρόωγη διακοπή κατά τη διάρκεια της εκπαίδευσης, ή/και iv) η αύξηση της παραμέτρου κανονικοποίησης (*regularization parameter*)». Η μεταβλητότητα του μοντέλου μπορεί να υπολογιστεί από τον τύπο

$$\text{Var}(\theta) = E \left[(E[\theta] - \theta)^2 \right] \quad (2.9)$$

2.3.4.2 Πόλωση

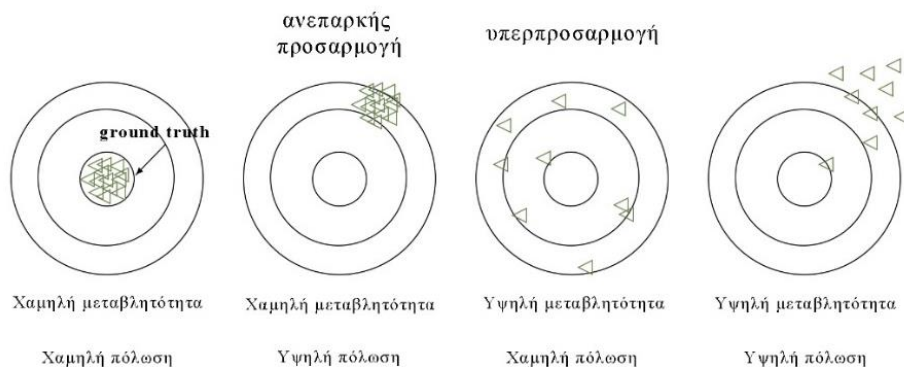
Η **πόλωση** (*bias*) είναι η διαφορά μεταξύ της μέσης πρόβλεψης του μοντέλου από την σωστή πρόβλεψη, που οφείλεται στις απλοποιητικές υποθέσεις που γίνονται από ένα μοντέλο για να γίνει πιο εύκολη η εκμάθηση του. Όσο υψηλότερη είναι η τιμή του τόσο λιγότερη προσοχή δίνει το μοντέλο στα δεδομένα εκπαίδευσης με αποτέλεσμα να έχουμε **ανεπαρκής προσαρμογή** (*underfitting*). Σύμφωνα με τους Gudivada & Rao [GR18] « Πιθανές λύσεις για την μείωση της υψηλής πόλωσης είναι i) η δημιουργία ενός πιο περίπλοκου μοντέλου (νευρωνικό δίκτυο με περισσότερες στρώσεις), ii) η εκπαίδευση του μοντέλου για περισσότερο χρόνο, iii) η μείωση του θορύβου στα δεδομένα, ή/και iv) η μείωση της παραμέτρου κανονικοποίησης (*regularization parameter*)». Η πόλωση υπολογίζεται ως εξής:

$$\text{Bias} = E[\theta] - \theta \quad (2.10)$$

2.3.4.3 Δίλημμα πόλωσης-μεταβλητότητας

Ενώ η μεταβλητότητα και η πόλωση είναι δύο διαφορετικά μεταξύ τους σφάλματα, μπορούν εύκολα να έρθουν σε αντιδιαστολή. Ένας αλγόριθμος δεν μπορεί να είναι πολύ περίπλοκος και λιγότερα περίπλοκος ταυτόχρονα, και ένα μοντέλο δεν μπορεί να εκπαιδευτεί ταυτόχρονα για πολύ χρόνο και για λίγο χρόνο. Στην προσπάθεια ταυτόχρονης ελαχιστοποίησης και των δύο πηγών σφάλματος προκαλείται μια σύγκρουση μεταξύ τους. Η αντιστάθμιση στην πολυπλοκότητα αυτή ονομάζεται **δίλημμα πόλωσης-μεταβλητότητας** (*bias-variance tradeoff*).

Όπως βλέπουμε στην Εικόνα 2.4 για να έχουμε ένα σωστά εκπαιδευμένο μοντέλο και τα δύο αυτά σφάλματα θα πρέπει να είναι αρκετά χαμηλά. Στην περίπτωση που τουλάχιστον ένα από αυτά έχει υψηλή τιμή το μοντέλο δεν εκπαιδεύεται σωστά. Για το λόγο αυτό σε ένα μοντέλο για να περιορίσουμε το δίλημμα αυτό υπολογίζουμε ένα συνολικό σφάλμα (*total error*), βρίσκοντας έτσι μια ισορροπία ανάμεσα στην μεταβλητότητα και την πόλωση. Το συνολικό σφάλμα προκύπτει από την πρόσθεση των δύο σφαλμάτων μαζί με το ανεπανόρθωτο



Εικόνα 2.9 Μεταβλητότητα και πόλωση Μοντέλου. Στην εικόνα βλέπουμε τα τέσσερα ζευγάρια τιμών των δύο σφαλμάτων, καθώς και την επίδρασή τους στην απόδοση του μοντέλου. Στο κέντρο του στόχου έχουμε το μοντέλο που προβλέπει τέλεια τις τιμές, ενώ όσο απομακρυνόμαστε από το κέντρο οι προβλέψεις γίνονται όλο και χειρότερες.

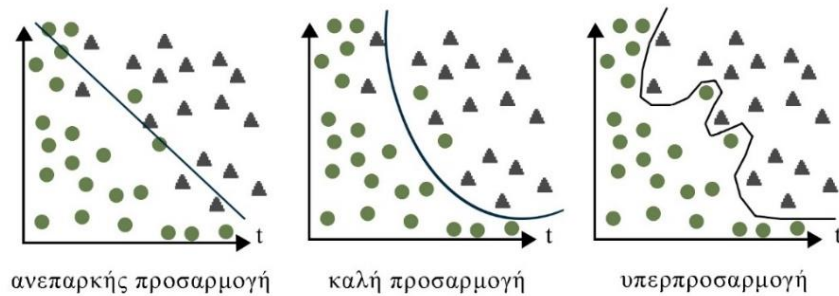
σφάλμα (*irreducible error*), το οποίο είναι ένα σφάλμα που προκύπτει από το θόρυβο στο σύνολο δεδομένων και δεν μπορεί να μειωθεί. Το συνολικό σφάλμα υπολογίζεται ως εξής:

$$Err = Bias^2 + Variance + Irreducible Error \quad (2.11)$$

2.3.4.4 Ανεπαρκής προσαρμογή και υπερπροσαρμογή

Η **υπερπροσαρμογή** (*overfitting*) και η **ανεπαρκής προσαρμογή** (*underfitting*) είναι οι δύο μεγαλύτερες αιτίες κακής απόδοσης στην μηχανική μάθηση. Ένα μοντέλο εμφανίζει υπερπροσαρμογή όταν έχει χαμηλό σφάλμα εκπαίδευσης και υψηλό σφάλμα ελέγχου. Αυτό συμβαίνει γιατί ο θόρυβος και οι τυχαίες αυξομειώσεις που εμφανίζονται στα δεδομένα εκ-

παίδευσης συλλέγονται και μαθαίνονται και αυτά και έτσι επηρεάζεται η ικανότητα γενίκευσης των μοντέλων [GBC16].



Εικόνα 2.10 Προσαρμογή Μοντέλου. Στην εικόνα βλέπουμε 3 γραφήματα με 3 διαφορετικές περιπτώσεις προσαρμογής μοντέλου, i) ανεπαρκής προσαρμογή (underfitting), ii) καλή προσαρμογή, και iii) υπερπροσαρμογή (overfitting).

Η κύρια αιτία της υπερπροσαρμογής είναι η χρήση μη-παραμετρικών και μη-γραμμικών μεθόδων πάνω σε γραμμικά δεδομένα, μιας και παρουσιάζουν μεγαλύτερη ευελιξία κατά την εκμάθηση. Από την άλλη ένα μοντέλο εμφανίζει ανεπαρκής προσαρμογή όταν έχει υψηλό σφάλμα τόσο στην εκπαίδευση όσο και στον έλεγχο του. Αυτό συμβαίνει όταν το μοντέλο είναι υπερβολικά κανονικοποιημένο και με λίγα χαρακτηριστικά, με αποτέλεσμα να μην μπορεί να μάθει και να καταγράψει την υποκείμενη τάση των δεδομένων. Η λύση είναι η χρήση διαφορετικού αλγόριθμου μηχανικής μάθησης ή η χρήση περισσότερων δεδομένων.

2.4 Γεννητικά Ανταγωνιστικά Δίκτυα

2.4.1 Παραγωγικά και Διευκρινιστικά Μοντέλα

Τα **Γεννητικά Ανταγωνιστικά Δίκτυα** (*Generative Adversarial Networks GANs*) έχουν αναδειχθεί ως ένα πολύ ισχυρό εργαλείο εκμάθησης μοντέλων για βασίζονται σε ένα σενάριο παιχνιδιού στο οποίο δύο Νευρωνικά Δίκτυα ανταγωνίζονται μεταξύ τους [GBC16]. Αποτελούνται από ένα **Παραγωγικό Μοντέλο** (*Generative Models*) και ένα **Διευκρινιστικό Μοντέλο** (*Discriminative Model*) [LYZF20].

Παραγωγικά Μοντέλα

Τα **Παραγωγικά Δίκτυα** εκτιμούν την **συνδυασμένη πιθανότητα** (*joint probability*) $P(\bar{X}, y)$, η οποία είναι η γεννητική πιθανότητα παρουσίας δεδομένων. Σε περίπτωση που χρειάζεται να εκτιμήσουμε την **υπό όρους πιθανότητα** (*conditional probability*) $P(y|\bar{X})$ του στόχου y , δεδομένου του X , μπορούμε να χρησιμοποιήσουμε το **Μπαγιεσιανό κανόνα** :

$$P(y|\bar{X}) = \frac{P(\bar{X}, y)}{P(\bar{X})} = \frac{P(\bar{X}, y)}{\sum_z P(\bar{X}, z)} \quad [A18] \quad (2.12)$$

Κάποια γνωστά Παραγωγικά Μοντέλα είναι i) ο Αφελής Ταξινομητής Bayes (*Naïve Bayes*), ii) τα Μπαγιεσιανά Δίκτυα (*Bayesian networks*), iii) τα Τυχαία Μαρκοβιανά Πεδία (*Markov random fields*), και iv) τα Κρυφά Μαρκοβιανά Μοντέλα (*Hidden Markov Models HMM*) [LYZF20].

Διευκρινιστικά Μοντέλα

Τα **Διευκρινιστικά Μοντέλα** εκτιμούν άμεσα την **υπό όρους πιθανότητα** $P(y|\bar{X})$ της ετικέτας y , δεδομένου των τιμών του X [A18]. Κάποια γνωστά Διευκρινιστικά Μοντέλα είναι i) η Λογιστική Παλινδρόμηση (*Logistic Regression*), ii) τα Νευρωνικά Δίκτυα (*Traditional Neural Networks*), iii) η μέθοδος Κοντινότερων Γειτόνων (*Nearest Neighbors*), και iv) τα Τυχαία Πεδία υπό όρους (*Conditional Random Fields CRFs*).

2.4.2 Λειτουργία των GANs

Τα GANs μπορούν να χαρακτηριστούν ως ένα παιχνίδι ανάμεσα σε δύο αντιπάλους. Ο Παραγωγός παράγει δείγματα $x = G(z; \theta_g)$, ενώ ο Διευκρινιστής προσπαθεί να μάθει να ξεχωρίζει με όσο μεγαλύτερη ακρίβεια μπορεί τα αληθινά δεδομένα από τα συνθετικά δεδομένα που παράγει ο Παραγωγός με τη χρήση της πιθανότητας $D(x; \theta_d)$ [GBC16]. Η βελτιστοποίηση του παιχνιδιού επιτυγχάνεται όταν ο Παραγωγός μαθαίνει την κατανομή των πραγματικών δεδομένων, ενώ ο Διευκρινιστής προσπαθεί να διακρίνει τα αληθινά από τα συνθετικά δεδομένα χωρίς εν τέλη να το πετυχαίνει. Με άλλα λόγια, στόχος του Παραγωγού είναι να μεγιστοποιήσει την πιθανότητα ο Διευκρινιστής να κάνει λάθος [ZYZ20]:

$$\min_G \max_D J(\theta_g, \theta_d) \quad (2.13)$$

όπου

$$J(\theta_g, \theta_d) = \mathbb{E}_{x \sim p_{data}(x)} [\log D(x; \theta_d)] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z; \theta_g); \theta_d))] \quad (2.14)$$

ή χωρίς τις παραμέτρους θ_g και θ_d ,

$$J(G, D) = \mathbb{E}_{x \sim p_{data}(x)} [\log D(x)] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))] \quad (2.15)$$

Τα Γεννητικά Ανταγωνιστικά Δίκτυα βασίζονται στο **μη συνεργατικό παιχνίδι μη-δενικού αθροίσματος** (*zero-sum non-cooperative game*), δηλαδή στην περίπτωση που το ένα δίκτυο κερδίσει αυτόματα το άλλο χάνει. Σε όλα τα ανταγωνιστικά παιχνίδια δύο ατόμων, ο

στόχος είναι η ισορροπία [Z20]. Έτσι και στα Γεννητικά Ανταγωνιστικά Δίκτυα η ιδανική περίπτωση εκπαίδευσης είναι αυτή στην οποία τα δύο αντίπαλα δίκτυα φτάνουν στην **ισορροπία Ναϋ** (*Nash equilibrium*). Σύμφωνα με τον Goodfellow [GAN14] το σημείο ισορροπίας ή αλλιώς η ελάχιστη τιμή του 2.13 επιτυγχάνεται εάν και μόνο εάν έχουμε $p_g(x) = p_{data}(x)$, με την τιμή εξόδου του Διευκρινιστή D να είναι το $\frac{1}{2}$ ανεξάρτητα της εισόδου. Ο βέλτιστος Διευκρινιστής για γνωστό Παραγωγό δίνεται από τον τύπο:

$$D_G^*(x) = \frac{p_{data}(x)}{p_{data}(x) + p_G(x)} \quad (2.16)$$

ο οποίο αποδεικνύεται ως εξής:

Όπως ξέρουμε ο στόχος του Διευκρινιστή, με γνωστό Παραγωγό είναι να μεγιστοποιήσει την συνάρτηση (2.15) :

$$\begin{aligned} J(G, D) &= \int_x p_{data}(x) \log D(x) dx + \int_x p_z(z) \log(1 - D(G(z))) dz \\ &= \int_x p_{data}(x) \log D(x) dx + p_g(x) \log(1 - D(x)) dx \end{aligned} \quad (2.17)$$

Η συνάρτηση $y \rightarrow a \log(y) + b \log(1 - y)$ επιτυγχάνει ολικό μέγιστο στο διάστημα $[0, 1]$ στο σημείο $\frac{a}{a+b}$, για κάθε $(a, b) \in \mathbb{R}^2 \setminus \{0, 0\}$. Αν ερμηνεύσουμε τον στόχο του Διευκρινιστή ως τη μεγιστοποίηση της λογαριθμικής πιθανότητας, δηλαδή της $P(y|x)$ με $y = 1$ για $x \in p_{data}$ και με $y = 0$ για $x \in p_g$, η **συνάρτηση του παιγνίου** μπορεί να εκφραστεί ως εξής:

$$\begin{aligned} C(G) &= \max_D J(G, D) \\ &= \mathbb{E}_{x \sim p_{data}} [\log D_G^*(x)] + \mathbb{E}_{z \sim p_z} [\log(1 - D_G^*(G(z)))] \\ &= \mathbb{E}_{x \sim p_{data}} [\log D_G^*(x)] + \mathbb{E}_{z \sim p_z} [\log(1 - D_G^*(x))] \\ &= \mathbb{E}_{x \sim p_{data}} \left[\log \frac{p_{data}(x)}{p_{data}(x) + p_G(x)} \right] + \mathbb{E}_{z \sim p_z} \left[\log \frac{p_G(x)}{p_{data}(x) + p_G(x)} \right] \end{aligned} \quad (2.18)$$

Επιπρόσθετα, σύμφωνα με τον Goodfellow [GAN14] το ολικό ελάχιστο του κριτηρίου $C(G)$ επιτυγχάνεται αν και μόνο αν έχουμε $p_g(x) = p_{data}(x)$, και η τιμή του είναι ίση με το $-\log 4$. Με βάση την συνάρτηση (2.16) για $p_g(x) = p_{data}(x)$ έχουμε:

$$\begin{aligned} D_G^*(x) &= \frac{p_{data}(x)}{p_{data}(x) + p_{data}(x)} \\ &= \frac{p_{data}(x)}{2 * p_{data}(x)} = \frac{1}{2} \end{aligned}$$

Οπότε για την συνάρτηση (2.18) θα έχουμε $C(G) = \log \frac{1}{2} + \log \frac{1}{2} = -\log 4$. Αφαιρούμε την έκφραση αυτή από την $C(G) = V(D_G^*, G)$, και προκύπτει ότι:

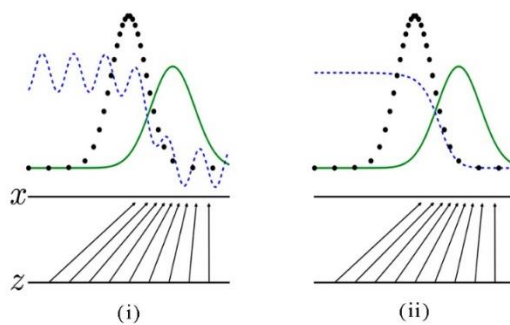
$$C(G) = -\log 4 + KL(p_{data} \parallel \frac{p_{data} + p_g}{2}) + KL(p_g \parallel \frac{p_{data} + p_g}{2}) \quad (2.19)$$

όπου KL είναι η **απόκλιση Kullback-Leibler** (*KL divergence*) ή **σχετική εντροπία** (*relative entropy*), η οποία είναι ένα μέτρο ανισότητας μεταξύ των κατανομών πιθανότητας [KL51]. Με βάση την απόκλιση KL μπορούμε να διακρίνουμε την **απόκλιση Jensen-Shannon** (*JS divergence*), η οποία είναι μέθοδος μέτρησης της ομοιότητας μεταξύ δύο κατανομών πιθανότητας,

$$KL(p_{data} \parallel \frac{p_{data} + p_g}{2}) + KL(p_g \parallel \frac{p_{data} + p_g}{2}) = 2 \bullet JSD(p_{data} \parallel p_g) \quad (2.20)$$

$$C(G) = -\log 4 + 2 \bullet JSD(p_{data} \parallel p_g) \quad (2.21)$$

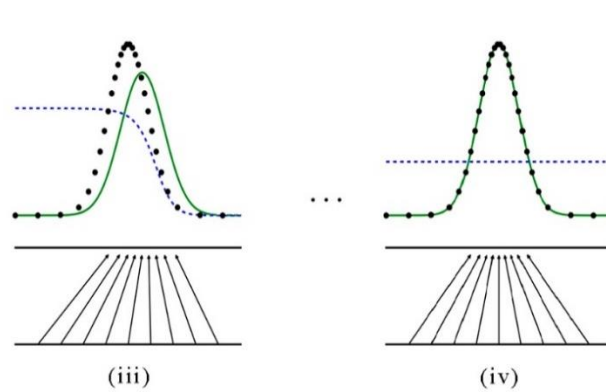
Για την απόκλιση JS ισχύει πάντα ότι $JSD(p_1 \parallel p_2) \geq 0$, με $JSD(p_1 \parallel p_2) = 0$ μόνο όταν οι κατανομές είναι ίσες. Και έτσι αποδεικνύεται ότι $C^* = -\log 4$, είναι το ολικό ελάχιστο της $C(G)$.



Εικόνα 2. 11 Εκπαίδευση Γεννητικών Ανταγωνιστικών Δικτύων 1.

Κατανομή Διευκρινιστή D : μπλε διακεκομμένη γραμμή. Κατανομή Παραγωγού G : πράσινη συμπαγής γραμμή. Κατανομή Παραγωγής Δεδομένων (αληθινά δεδομένα): μαύρες τελείες. Η πάνω οριζόντια γραμμή είναι μέρος των δεδομένων x , ενώ η κάτω οριζόντια γραμμή είναι μέρος των δεδομένων z .

(i) Τα αντίπαλα ζευγάρια είναι κοντά στην σύγκλιση, το p_g είναι παρόμοιο με το p_{data} , και ο Διευκρινιστής D μπορεί να διακρίνει τα αληθινά από τα συνθετικά δεδομένα με μερική ακρίβεια. (ii) Ο αλγόριθμος του D , εκπαίδευεται για να διακρίνει δείγματα από δεδομένα, και συγκρίνει στην τιμή $D_G^* x = \frac{p_{data} x}{p_{data} x + p_g x}$.



Εικόνα 2.12 Εκπαίδευση Γεννητικών Ανταγωνιστικών Δικτύων 2.

(iii) Μετά από ενημέρωση του Παραγωγού G , η κλίση του D οδηγεί το $G(z)$ να δημιουργεί δεδομένα που είναι πιο πιθανό να ταξινομηθούν ως αληθινά. (iv) Μετά από αρκετά στάδια εκπαίδευσης, τόσο ο G όσο και ο D φτάνουν στο σημείο που δεν μπορούν πλέον να βελτιωθούν διότι $p_{data} = p_g$. Ο D δεν μπορεί πλέον να διακρίνει μεταξύ των δύο κατανομών, $D_G^*(x) = \frac{1}{2}$.

Ο Αλγόριθμος 1 συνοψίζει την κύρια διαδικασία εκπαίδευσης των GANs, υιοθετώντας την λογική βελτιστοποίησης **μικρής παρτίδας** (*mini batch*). Οι ενημερώσεις των κλίσεων μπορούν να χρησιμοποιούν οποιονδήποτε τυπικό κανόνα εκμάθησης με **βάση την κλίση** (*gradient-based learning rule*). Η παρακάτω υλοποίηση του αλγορίθμου Εκπαίδευσης GANs βασίζεται στην υλοποίηση στο πρωτότυπο επιστημονικό άρθρο που αναφέρεται για πρώτη φορά στα Γεννητικά Ανταγωνιστικά Δίκτυα (*Generative adversarial nets 2014*, [GAN14]).

Αλγόριθμος 1 – Εκπαίδευση Γεννητικών Ανταγωνιστικών Δικτύων

Είσοδος: $k_{max} \geq 1$ ➤ Αριθμός επαναλήψεων εκπαίδευσης Διευκρινιστή D ανά εκπαίδευση G

Είσοδος: $batch_{mini} \geq 1$ ➤ Πλήθος δειγμάτων παρτίδας (*batch size*). Σε αυτή την περίπτωση μικρή παρτίδα (*minibatch*).

1: **ΑΛΓΟΡΙΘΜΟΣ ΔΙΑΔΙΚΑΣΙΑ ΕΚΠΑΙΔΕΥΣΗΣ**($k_{max}, batch_{mini}$)

2: **ΑΡΧΙΚΟΠΟΙΗΣΗ** $\theta_d^{(0)}, \theta_g^{(0)}$

3: **Όσο** θ_d και θ_g δεν συγκλίνουν, **επανάλαβε**

4: $k \leftarrow 0$

5: **Επανάλαβε**

6: $\{z^{(1)}, \dots, z^{(batch_{mini})}\} \leftarrow \text{ΔΕΙΓΜΑΤΟΛΗΨΙΑΘΟΡΥΒΟΥ}(batch_{mini}, p_z(z))$

7: $\{x^{(1)}, \dots, x^{(batch_{mini})}\} \leftarrow \text{ΔΕΙΓΜΑΤΟΛΗΨΙΑΠΡΑΓΜΑΤΙΚΗΣΚΑΤΑΝΟΜΗΣ}(batch_{mini}, p_{data}(x))$

- 8: $\theta_{temp} \leftarrow \text{ΥΠΟΛΟΓΙΣΕΚΛΙΣΗΔΙΕΥΚΡΙΝΙΣΤΗ}(x, z, \theta_d, \theta_g)$
- $$\nabla_{\theta_d} \left\{ \frac{1}{batch_{mini}} \sum_{i=1}^{batch_{mini}} (\log D(\mathbf{x}^{(i)}; \theta_d) + \log(1 - D(G(z^{(i)}; \theta_g); \theta_d))) \right\}$$
- 9: $\theta_d \leftarrow \theta_{temp}$
- 10: **μέχρις ότου** $k \geq k_{max}$
- 11: $\{z^{(1)}, \dots, z^{(batch_{mini})}\} \leftarrow \text{ΔΕΙΓΜΑΤΟΛΗΨΙΑΘΟΡΥΒΟΥ}(batch_{mini}, p_g(z))$
- 12: $\theta_{temp} \leftarrow \text{ΥΠΟΛΟΓΙΣΕΚΛΙΣΗΠΑΡΑΓΩΓΟΥ}(z, \theta_d, \theta_g)$
- $$\nabla_{\theta_g} \left\{ \frac{1}{batch_{mini}} \sum_{i=1}^{batch_{mini}} \log(1 - D(G(z^{(i)}; \theta_g); \theta_d)) \right\}$$
- 13: $\theta_g \leftarrow \theta_{temp}$
- 14: **Τέλος Επανάληψης**
- 15: **Τέλος Αλγόριθμου**

2.4.3 Προβλήματα στην Εκπαίδευση

Η δυναμική της εκπαίδευσης των GANs έχει αποδειχθεί ότι είναι εξαιρετικά ασταθής. Ενώ η βελτιστοποίηση τους είναι καθιερωμένη, κανένα από τα κοινά προβλήματα που εμφανίζουν δεν έχει επιλυθεί πλήρως, και έτσι αποτελούν ενεργούς τομείς έρευνας.

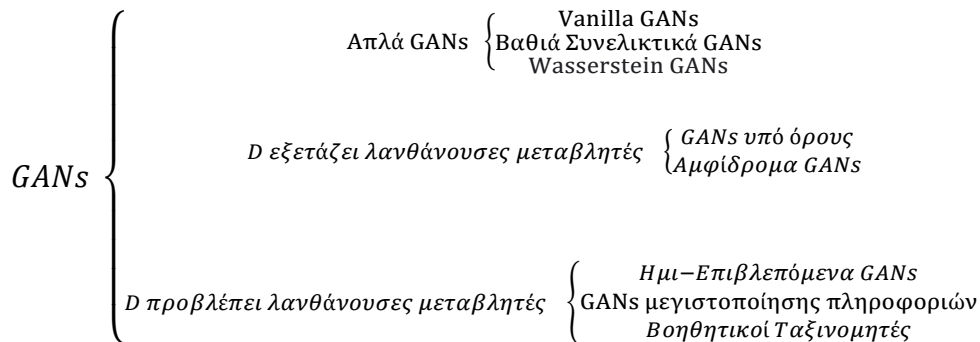
Ένα πρόβλημα που σχετίζεται με την εκπαίδευση των GANs είναι αυτό της **Κατάρρευσης Λειτουργίας** (*mode collapse*). Ο Παραγωγός ενώ καταφέρει να κάνει τον Διευκρινιστή να κατηγοριοποιήσει τα δεδομένα που παρήγαγε ως αληθινά, στην πραγματικότητα παράγει τις ίδιες εξόδους. Αυτό οφείλεται στο γεγονός ότι ο Παραγωγός δεν μαθαίνει σωστά την κατανομή των δεδομένων και εστιάζει σε συγκεκριμένα μέρη των δεδομένων, με αποτέλεσμα οι εξόδοι να παρουσιάζουν χαμηλή μεταβλητότητα (*low variability*) και χαμηλή ποικιλομορφία (*low diversity*) [T20] [WAN19].

Για ένα μοντέλο GAN, η σύγκλιση (*convergence*) μεταξύ του Διευκρινιστή και του Παραγωγού συχνά είναι φευγαλέα, και όχι σταθερή. Δεδομένου ότι η απώλεια του Παραγωγού βελτιώνεται όταν η απώλεια του Διευκρινιστή υποβαθμίζεται, και αντίστροφα, δεν μπορούμε να κρίνουμε τη σύγκλιση βάσει της τιμής της συνάρτησης απώλειας (*loss function*) [LB19]. Όταν ο Παραγωγός επιτύχει τέλεια, τότε ο Διευκρινιστής έχει ακρίβεια 0.5. Η εξέλιξη αυτή έχει ως αποτέλεσμα ο Διευκρινιστής να δίνει τυχαία ανατροφοδότηση (*feedback*) στον Παραγωγό. Στην περίπτωση που δεν σταματήσει έγκαιρα η εκπαίδευση στο σημείο που ξεκινάει η τυχαία ανατροφοδότηση αυτή, ο Παραγωγός θα αρχίσει να εκπαιδεύεται με λάθος ανατροφοδότηση, με αποτέλεσμα να μεγαλώνει η πιθανότητα του να καταρρεύσει [WAN19].

Τέλος, σύμφωνα με τη θεωρία ένας Διευκρινιστής ο οποίος πλησιάζει τον βέλτιστο παράγει ουσιαστική ανατροφοδότηση στον Παραγωγό, με αποτέλεσμα να δημιουργείται και ένας βελτιωμένος Παραγωγός. Στην πράξη όμως, η έρευνα έχει δείξει ότι εάν ο Διευκρινιστής είναι υπερβολικά καλός, τότε η εκπαίδευση του Παραγωγού μπορεί να αποτύχει. Για έναν τόσο ακριβή Διευκρινιστή, όπου ισχύει ότι $D(x) = 1, \forall x \in p_{data}$ και $D(G(z)) = 0, \forall G(z) \in p_z$, το κόστος δεν θα είναι ίσο με $2 \cdot JSD(p_{data} || p_g)$, αλλά τείνει προς το μηδέν. Έτσι θα δημιουργούνται κλίσεις κοντά στο μηδέν, με αποτέλεσμα ο Διευκρινιστής να μην παρέχει αρκετές πληροφορίες στον Παραγωγό με αποτέλεσμα να μην σημειώνει πρόοδο στην εκπαίδευση. Το πρόβλημα αυτό ονομάζεται πρόβλημα εξαφάνισης των διανυσμάτων κλίσης (*Vanishing gradient problem*) [WAN19] [LB19].

2.4.4 Είδη GANs

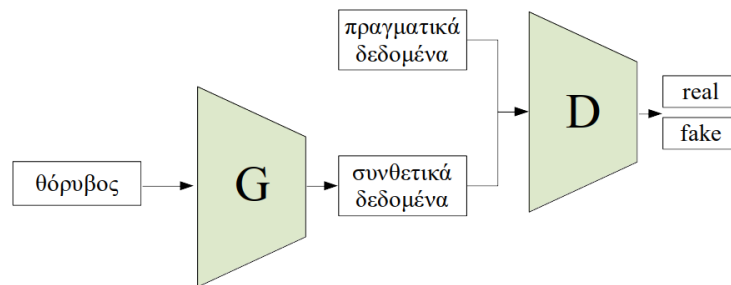
Τα Γεννητικά Ανταγωνιστικά Δίκτυα εισήχθησαν πρόσφατα ως ένας νέος τρόπος για την εκπαίδευση γεννητικών μοντέλων. Με την πάροδο του χρόνου, οι ερευνητές άρχισαν με βάση την αρχιτεκτονική των Απλών GANs να βρίσκουν βελτιωμένες επεκτάσεις της αρχιτεκτονικής αυτής, καθώς και νέες χρήσεις των GANs. Σε αυτή την υπό ενότητα θα μιλήσουμε για τις πιο κοινές ή ευρέως χρησιμοποιούμενες διακριτές επεκτάσεις της αρχιτεκτονικής των GANs και της εκπαίδευσής τους. Μπορούμε να διακρίνουμε τις παραλλαγές των GANs στην Εικόνα 2.11.



Εικόνα 2.13 Παραλλαγές GANs. Οι 6 συγκεκριμένες παραλλαγές χωρίζονται σε 3 υποκατηγορίες, 1) Απλά GANs, 2) GANs στα οποία ο Διευκρινιστής εξετάζει τις λανθάνουσες μεταβλητές, και 3) GANs στα οποία ο Διευκρινιστής προβλέπει λανθάνουσες μεταβλητές. Στην πρώτη κατηγορία ανήκουν τα Vanilla GANs, τα Βαθιά Συνελικτικά GANs (Deep Convolutional GANs DCGANs), και τα Wasserstein GANs (WGANs). Στην δεύτερη κατηγορία ανήκουν τα GANs υπό όρους (Conditional GANs cGANs), και τα αμφίδρομα GANs (Bidirectional GANs BiGANs). Στην τελευταία κατηγορία ανήκουν τα (iv) Ημι-Επιβλεπόμενα GANs (Semi-Supervised GANs SGANs), τα GANs μεγιστοποίησης πληροφοριών (Information Maximizing GANs InfoGANs), και οι Βοηθητικοί Ταξινομητές (Auxiliary Classifier GANs AC-GANs).

2.4.4.1 Απλά GANs

Σύμφωνα με την Εικόνα 2.11 τα **απλά GANs** διακρίνονται σε 3 δημοφιλείς κατηγορίες, i) **Vanilla GANs**, ii) **Βαθιά Συνελκτικά GANs**, και iii) **Wasserstein GANs**. Η πρώτη κατηγορία είναι τα γενικά απλά GANs τα οποία εξετάστηκαν σε προηγούμενες ενότητες. Η γενική αρχιτεκτονική των Απλών GANs είναι ίδια και στις τρεις περιπτώσεις όπως φαίνεται στην Εικόνα 2.13.



Εικόνα 2.14 Αρχιτεκτονική Απλών GANs

Γεννητικά Ανταγωνιστικά Δίκτυα Wasserstein

Υπάρχουν τρεις βασικές κατηγορίες συναρτήσεων απώλειας, i) η οικογένεια φ-αποκλίσεων (*f-divergence*), ii) οι μέθοδοι μετρικής ακέραιας πιθανότητας (*integral probability metric IPM*), καθώς και iii) οι βοηθητικές λειτουργίες απώλειας (*auxiliary loss functions*). Τόσο η απόκλιση KL όσο και η απόκλιση JSD, ανήκουν στην οικογένεια των φ-αποκλίσεων, και μετρούν την διαφορά μεταξύ δύο κατανομών πιθανότητας [WAN19]. Από την άλλη οι μέθοδοι IPM σχεδιάστηκαν για να αντιμετωπίζουν τους περιορισμούς των φ-αποκλίσεων, και να εμφανίζουν μια συνεπή απόσταση μεταξύ των κατανομών δεδομένων. Στην οικογένεια αυτών των συναρτήσεων απώλειας ανήκει η λεγόμενη απόσταση **Wasserstein** ή **απόσταση Earth Mover's (EMD)**, και ορίζεται ως εξής [WGAN17]:

$$W(p_{data}, p_g) = \inf_{\gamma \in \Pi(p_{data}, p_g)} \mathbb{E}_{(x,y) \sim \gamma} [\|x - y\|] \quad (2.22)$$

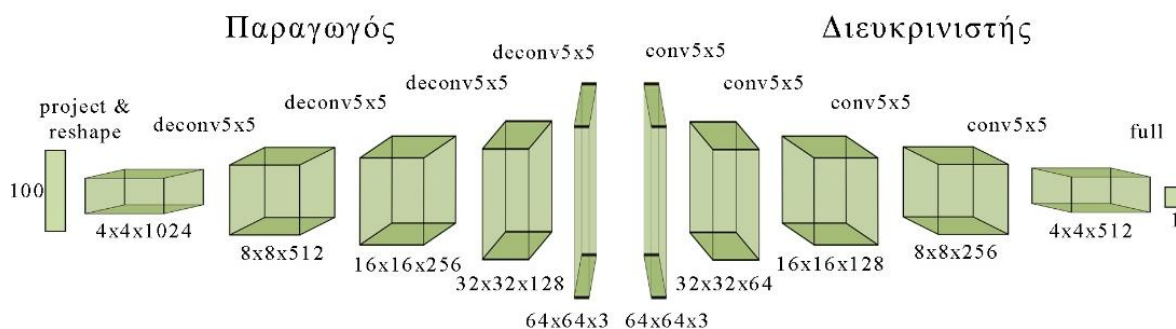
Με τη χρήση της απόστασης Wasserstein τα Γεννητικά Ανταγωνιστικά Δίκτυα Wasserstein (WGANs) έφεραν την θεωρία της **Βέλτιστης Μεταφοράς (Optimal Transport OP)** στα GANs, ως αντικειμενική λειτουργία τους. Η θεωρία αυτή διατυπώνεται ως η μετατόπιση της μάζας μιας πηγαίας κατανομής στην κατανομή στόχου με την ελάχιστη κατανάλωση έργου κατά την μεταφορά, η οποία επιτυγχάνεται για την απόσταση 1-Wasserstein η οποία είναι η ελάχιστη από τις πιθανές αποστάσεις μεταξύ των μοντέλων και της κατανομής των δεδομένων [T20]. Η επέκταση αυτή των GANs αναζητά εναλλακτικό τρόπο εκπαίδευσης του Παραγωγού για την καλύτερη προσέγγιση της κατανομής των δεδομένων σε ένα σύνολο δε-

δομένων εκπαίδευσης. Λόγω της σταθερότητας τους καθώς και του θεωρητικού πλαισίου έχουν αποκτήσει σημαντικό ενδιαφέρον στην επιστημονική κοινότητα.

Βαθιά Συνελικτικά Γεννητικά Ανταγωνιστικά Δίκτυα

Τα **Βαθιά Συνελικτικά Γεννητικά Ανταγωνιστικά Δίκτυα** (*Deep Convolutional Generative Adversarial Networks DCGANs*) είναι μία από τις πλέον πιο δημοφιλείς και επιτυχημένες παραλλαγές των GANs. Η αρχιτεκτονική αυτή χρησιμοποιεί Βαθιά Συνελικτικά Νευρωνικά Δίκτυα (Deep *CNNs*) τόσο για τον Παραγωγό όσο για τον Διευκρινιστή, καθώς και διαμορφώσεις των μοντέλων και της διαδικασίας εκπαίδευσης που οδηγούν στην πιο σταθερή εκπαίδευση του Παραγωγού [RMC15].

Ο πυρήνας της αρχιτεκτονικής του Διευκρινιστή χρησιμοποιεί την τυπική αρχιτεκτονική του Συνελικτικών Νευρωνικών Δικτύων. Ο πυρήνας της αρχιτεκτονικής του Παραγωγού κάνει χρήση **από-συνέλιξης** (*deconvolution*), η οποία είναι η μαθηματική λειτουργία που αντιστρέφει την επίδραση της συνέλιξης, μιας και από έναν λανθάνοντα φορέα χαμηλών διαστάσεων πρέπει χαρτογραφεί μια εικόνα υψηλών διαστάσεων [AKK21]. Μια απλή αρχιτεκτονική DCGANs απεικονίζεται στην **Εικόνα 2.14**. Επιπρόσθετα στην αρχιτεκτονική αυτή δεν υπάρχουν επίπεδα συμψηφισμού (*pooling layers*), αλλά τα αντικαθιστούν οι **Διασκελισμένες Συνελίξεις** (*Strided Convolutions*) στο Διευκρινιστή και οι **Κλασματικά-Διασκελισμένες Συνελίξεις** (*Fractional-Strided Convolutions*) στον Παραγωγό. Τόσο στο Διευκρινιστή, όσο και στον Παραγωγό, γίνεται ομαλοποίηση παρτίδας (*Batch Normalization*), και αφαιρούνται όλα τα πλήρως συνδεδεμένα κρυφά στρώματα (*Fully Connected Hidden Layers*) στην περίπτωση βαθύτερων αρχιτεκτονικών [RMC15].

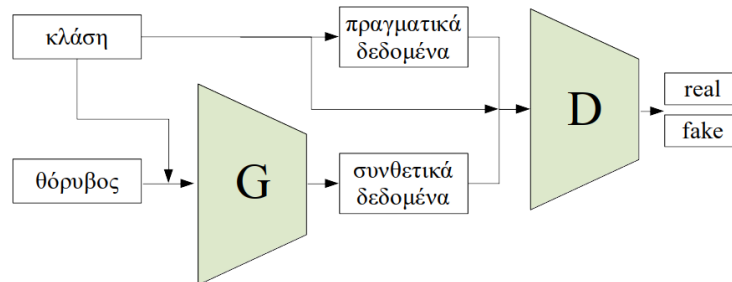


Εικόνα 2.15 Αρχιτεκτονική *Deep Convolutional GANs*

2.4.4.2 Γεννητικά Ανταγωνιστικά Δίκτυα υπό όρους

Τα **Γεννητικά Ανταγωνιστικά Δίκτυα υπό όρους** (*Conditional GANs cGANs*) περιεγραφήκαν για πρώτη φορά από τους Mehdi Mirza και Simon Osindero το επιστημονικό

άρθρο “*Conditional Generative Adversarial Nets*” [CGAN14]. Στο άρθρο, οι συγγραφείς παρακινούν την προσέγγιση με βάση την επιθυμία να κατευθύνουν τη διαδικασία δημιουργίας εικόνας του Παραγωγού.



Εικόνα 2.16 Αρχιτεκτονική *Conditional GANs*

Σύμφωνα με αυτούς ένα απλό Γεννητικό Ανταγωνιστικό Δίκτυο μπορεί επεκταθεί σε ένα μοντέλο υπό όρους εάν τόσο ο Παραγωγός όσο και ο Διευκρινιστής εξαρτώνται από κάποια πρόσθετη πληροφορία. Αυτή η βοηθητική πληροφορία μπορεί να είναι οτιδήποτε, όπως ετικέτα κλάσης, ένα σύνολο από ετικέτες, ή ακόμη και μια γραπτή περιγραφή [LB19]. Κατά την διάρκεια της εκπαίδευσης ο Παραγωγός μαθαίνει να παράγει δεδομένα για κάθε ετικέτα, στο σύνολο δεδομένων εκπαίδευσης, ξεχωριστά, ενώ ο Διευκρινιστής μαθαίνει να διακρίνει τα συνθετικά ζεύγη δεδομένων και ετικετών, από τα αληθινά ζεύγη. Ο Διευκρινιστής δεν θεωρεί συνθετικά μόνο τα ζεύγη τα οποία αποτελούνται από συνθετικά δεδομένα, αλλά απορρίπτει και τα ζεύγη που έχουν λάθος αντιστοίχιση δεδομένων και ετικέτας, ανεξάρτητα αν τα δεδομένα είναι ρεαλιστικά ή όχι. Με λίγα λόγια, για να «κερδίσει» το ανταγωνιστικό παιχνίδι ο Παραγωγός, πρέπει όχι μόνο να παράγει συνθετικά δεδομένα τα οποία είναι αληθοφανή, αλλά και να μπορεί να ταιριάζει τα δεδομένα που παράγει με τις σωστές ετικέτες [CGAN14].

Παραγωγός	Διευκρινιστής
Αληθοφανή συνθετικά δεδομένα + σωστή ετικέτα	Έγκριση
Αληθοφανή συνθετικά δεδομένα + λάθος ετικέτα	Απόρριψη
Μη αληθοφανή συνθετικά δεδομένα + σωστή ετικέτα	Απόρριψη
Μη αληθοφανή συνθετικά δεδομένα + λάθος ετικέτα	Απόρριψη

Πίνακας 2.3 Έγκριση και απόρριψη Δεδομένων στα *cGANs*

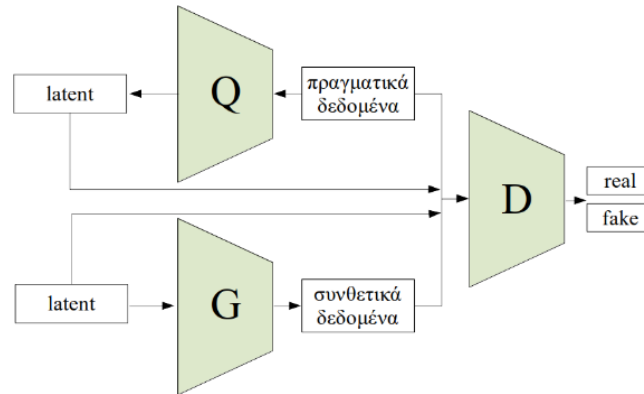
Η συνάρτηση του ανταγωνιστικού παιχνιδιού $\min_{G} \max_{D} J(G, D)$ στα *cGANs*, είναι η εξής:

$$\min_G \max_D J(G, D) = \mathbb{E}_{x \sim p_{data}(x)} [\log D(x|y)] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(x|y)))] \quad (2.23)$$

Η ικανότητα των *cGANs* να χειρίζονται τον λανθάνοντα χώρο με τέτοιο τρόπο ώστε να δημιουργούν συγκεκριμένες κατηγορίες πραγμάτων από διαφορετικά σύνολα δεδομένων με ετικέτες, είναι πολύ χρήσιμη και ανοίγει την πόρτα σε μια τεράστια γκάμα εφαρμογών.

2.4.4.3 Αμφίδρομα Γεννητικά Ανταγωνιστικά Δίκτυα

Τα Γεννητικά Ανταγωνιστικά Δίκτυα μπορούν να μάθουν **χαρτογράφηση γεννητικών μοντέλων** (*generative models mapping*) από απλές λανθάνουσες κατανομές δεδομένων έως αυθαίρετα περίπλοκες κατανομές δεδομένων [Z20]. Σε αρκετές εφαρμογές των GANs είναι σημαντικό να μπορεί το μοντέλο να μάθει την αντίστροφη χαρτογράφηση. Τα **Αμφίδρομα Γεννητικά Ανταγωνιστικά Δίκτυα** (*Bidirectional GANs BiGANs*) επεκτείνουν τα GANs με τη προσθήκη ενός τρίτου στοιχείου, του **κωδικοποιητή** (*encoder*) E [DKD16].



Εικόνα 2.17 Αρχιτεκτονική Bidirectional GANs

Ο κωδικοποιητής είναι υπεύθυνος για την χαρτογράφηση του χώρου δεδομένων x στον λανθάνον χώρο z . Με τη χρήση ενός BiGAN είναι πλέον δυνατή η προβολή των δεδομένων πίσω στον λανθάνοντα χώρο. Ενώ ο στόχος του Παραγωγού παραμένει ο ίδιος, ο στόχος του Διευκρινιστή πλέον δεν είναι μόνο η ταξινόμηση δεδομένων σε αληθινά και συνθετικά (x έναντι $G(z)$), αλλά και η ταξινόμηση ανάμεσα σε αληθινή (από τον κωδικοποιητή) και συνθετική (από το λανθάνον διάστημα z) κωδικοποίηση (πλειάδες $(x, E(x))$ έναντι $(G(z), z)$) [DKD16]. Η συνάρτηση του παιγνίου είναι η εξής:

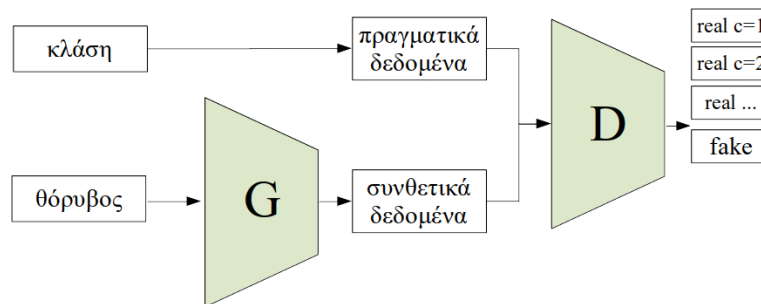
$$\begin{aligned} \min_{G,E} \max_D J_{G,E,D} &= \mathbb{E}_{x \sim p_x} \{ \mathbb{E}_{z \sim p_E(\bullet|x)} \log D(x, z) \} \\ &+ \mathbb{E}_{x \sim p_z} \{ \mathbb{E}_{x \sim p_G(\bullet|z)} \{ \log(1 - D(x, z)) \} \} \end{aligned} \quad (2.24)$$

2.4.4.4 Ημι-Επιβλεπόμενα Γεννητικά Ανταγωνιστικά Δίκτυα

Ο τομέας της **ημι-επιβλεπόμενης μάθησης** (*semi-supervised learning*) είναι ένας από τους πιο πολλά υποσχόμενους τομείς πρακτικής εφαρμογής των GANs. Στην ημι-επιβλεπόμενη μάθηση, ένα μικρό υποσύνολο του συνόλου εκπαίδευσης έχει μια ετικέτα κλάσης, σε αντίθεση με την επιβλεπόμενη μάθηση (*supervised learning*), στην οποία χρειαζόμαστε μια ετικέτα για κάθε στοιχείο στο σύνολο δεδομένων μας, ή τη μη-επιβλεπόμενη μάθηση (*unsupervised learning*), στην οποία δεν χρησιμοποιούνται ετικέτες. Για τη σωστή

λειτουργία της ημι-επιβλεπόμενης μάθησης, είναι σημαντικό το σύνολο των δεδομένων με ετικέτα και το σύνολο δεδομένων χωρίς ετικέτα να προέρχονται από την ίδια κατανομή δεδομένων [LB19].

Τα **Ημι-Επιβλεπόμενα Γεννητικά Ανταγωνιστικά Δίκτυα** (*Semi-Supervised GANs* *SGANs*) περιλαμβάνουν ταυτόχρονη εκπαίδευση ενός Επιβλεπόμενου Διευκρινιστή, ενός μη-Επιβλεπόμενου Διευκρινιστή, και του Παραγωγού.



Εικόνα 2.18 Αρχιτεκτονική *Semi-Supervised GANs*

Ο Παραγωγός, όπως και στα απλά GANs παράγει συνθετικά δεδομένα μέσω του θορύβου z . Ο Διευκρινιστής είναι ένας Ταξινομητής πολλαπλών κλάσεων μιας και λαμβάνει τρία είδη εισόδων δεδομένων: συνθετικά δεδομένα από τον Παραγωγό, αληθινά παραδείγματα χωρίς ετικέτα και πραγματικά παραδείγματα με ετικέτα (x, y) με $y \in N$. Εκπαιδεύεται με δύο διαφορετικούς τρόπους, 1) μη-επιβλεπόμενο, δηλαδή προβλέπει αν ένα στοιχείο είναι αληθινό ή συνθετικό, και 2) επιβλεπόμενο ταξινομεί τα αληθινά δεδομένα στις N κλάσεις του συνόλου δεδομένων εκπαίδευσης με ετικέτες. Με λίγα λόγια μαθαίνει να ταξινομεί $N+1$ κλάσεις [O16][LB19].

2.4.4.5 Γεννητικά Ανταγωνιστικά Δίκτυα Μεγιστοποίησης Πληροφοριών

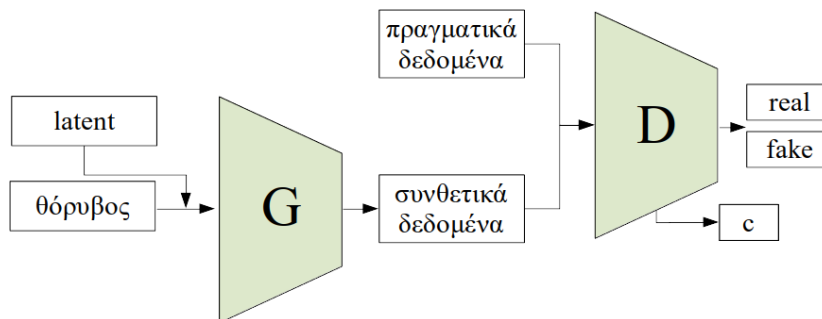
Τα **Γεννητικά Ανταγωνιστικά Δίκτυα Μεγιστοποίησης Πληροφοριών** (*Information Maximizing GANs* *InfoGANs*) εφαρμόζουν έννοιες της θεωρίας της πληροφορίας (*information theory*) στα GANs, για την εκμάθηση αποσυνδεδεμένων αναπαραστάσεων με μη-επιβλεπόμενο τρόπο. Πρόκειται για μοντέλα στα οποία εκτός από θόρυβο z ως είσοδο του Παραγωγού παρέχονται και λανθάνοντες κωδικοί c (*latent codes*) οι οποίοι έχουν ως στόχο τα εμφανή δομημένα σημασιολογικά χαρακτηριστικά (*salient structured semantic features*) της κατανομής των δεδομένων [INFO16]. Οι κωδικοί αυτοί καθίσταται ουσιαστικά αν και μόνο αν μεγιστοποιηθεί η αμοιβαία πληροφορία (*mutual information*) μεταξύ του λανθάνοντα κώδικα και της εξόδου του Παραγωγού.

Η αμοιβαία πληροφορία είναι η ποσότητα που μετρά τη σχέση μεταξύ δύο τυχαίων μεταβλητών, και την μείωση της αβεβαιότητας της μιας μεταβλητής λόγω γνώσης της άλλης. Συγκεκριμένα μετρά πόσες πληροφορίες μεταδίδονται, κατά μέσο όρο, σε μια τυχαία μεταβλητή για μια άλλη [WS49].

$$\begin{aligned}
 I(X; Y) &= D(P(x, y) || P(x)P(y)) = \sum_{i=1}^n \sum_{j=1}^m P(x_i, y_j) \log \frac{P(x_i, y_j)}{P(x_i)P(y_j)} \\
 &= H(X) - H(X|Y) = H(Y) - H(Y|X)
 \end{aligned}
 \tag{2.25}$$

Το minimax παίγνιο μεταξύ των δύο μοντέλων στο οποίο έχει γίνει ρύθμιση πληροφοριών (*information regularization*) δίνεται από τον εξής τύπο [INFO16]:

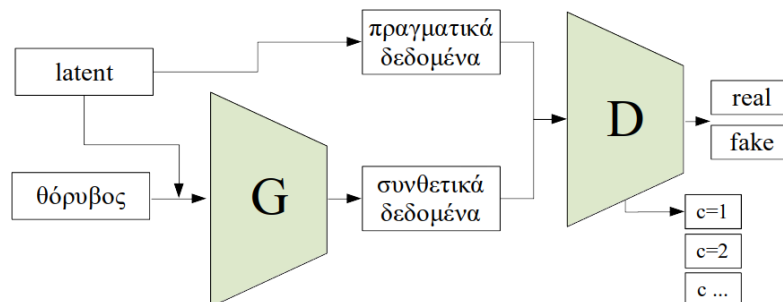
$$\min_G \max_D J_I(G, D) = J(G, D) - \lambda I(c; G(z, c))
 \tag{2.26}$$



Εικόνα 2.19 Αρχιτεκτονική Information Maximizing GANs

2.4.4.6 Βοηθητικοί Ταξινομητές

Οι **Βοηθητικοί Ταξινομητές** (*Auxiliary Classifier GANs AC-GANs*) είναι μια επέκταση των **cGANs**, στους οποίους ο Διευκρινιστής όχι μόνος πρέπει να διακρίνει τα αληθινά από τα συνθετικά δεδομένα εισόδου, αλλά να παρέχει και την σωστή ετικέτα κλάσης ανεξαρτήτως πηγής. Ως αποτέλεσμα, γίνεται σύνθεση μεγαλύτερων εικόνων υψηλής ποιότητας με μια πιο σταθερή διαδικασία εκπαίδευσης, στην οποία το μοντέλο μαθαίνει την αναπαράσταση στον λανθάνον χώρο, ο οποίος είναι ανεξάρτητος από τις ετικέτες [OOS16].



Εικόνα 2.20 Αρχιτεκτονική Auxiliary Classifier GANs

Η αντικειμενική συνάρτηση του παίγνιου αποτελείται από δύο μέρη:

- Φυσικός λογάριθμος πιθανότητας (*log-likelihood*) της σωστής πηγής LS:

$$L_S = E[\log P(S = real | X_{real})] + E[\log P(S = fake | X_{fake})] \quad (2.27)$$

- Φυσικός λογάριθμος πιθανότητας της σωστής κλάσης, LC:

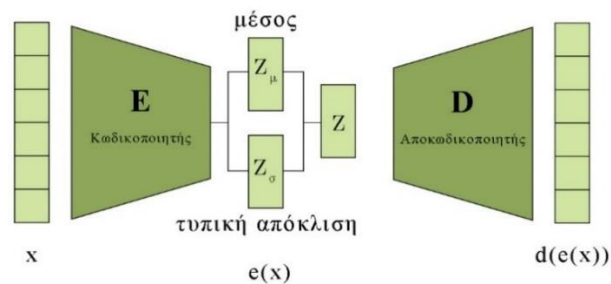
$$L_C = E[\log P(C = c | X_{real})] + E[\log P(C = c | X_{fake})] \quad (2.28)$$

2.5 Μεταβλητοί Αυτοκωδικοποιητές

Ενώ τα GAN είναι πλέον τα πιο δημοφιλή Παραγωγικά Μοντέλα, δεν είναι τα μόνα που χρησιμοποιούνται για πολλές εφαρμογές. Ένα άλλο είδος Παραγωγικών Μοντέλων είναι οι λεγόμενοι **Μεταβλητοί Αυτοκωδικοποιητές** (*Variational Autoencoders – VAE*) [VAE14]. Η γενική ιδέα ενός Απλού Αυτοκωδικοποιητή (*Autoencoder – AE*) είναι ο ορισμός δύο Νευρωνικών Δικτύων, ενός **Κωδικοποιητή** (*Encoder*), και ενός **Αποκωδικοποιητή** (*Decoder*), και η εκμάθηση του καλύτερου σχήματος κωδικοποίησης-αποκωδικοποίησης, με τη χρήση μιας επαναληπτικής διαδικασίας βελτιστοποίησης. Έτσι σε κάθε επανάληψη συγκρίνουμε τα δεδομένα που τροφοδοτούνται στον Κωδικοποιητή με αυτά που προκύπτουν από τον Αποκωδικοποιητή για να υπολογίσουμε το σφάλμα και έτσι να ενημερώσουμε τα βάρη και των δύο δικτύων.

Ένα VAE είναι ένας αυτόματος κωδικοποιητής του οποίου η διανομή κωδικοποιήσεων κανονικοποιείται κατά την εκπαίδευση, για την διασφάλιση των καλών ιδιοτήτων του λανθάνοντος χώρου με σκοπό την δημιουργία νέων δεδομένων. Σε αντίθεση με τους Απλούς Αυτοκωδικοποιητές οι οποίοι δεν μπορούν να χρησιμοποιηθούν για την παραγωγή νέων δεδομένων, τα VAE λόγω της κανονικοποίησης κατά την διάρκεια της εκπαίδευσης αποφεύγουν το πρόβλημα της υπερεκπαίδευσης, και διασφαλίζουν ότι ο λανθάνων χώρος είναι κατάλληλος για τη διαδικασία παραγωγής [GBC16].

Η εξίσωση Σφάλματος ενός Μεταβλητού Αυτοκωδικοποιητή στην αρχική της μορφή μπορεί να γραφτεί ως εξής:



Εικόνα 2.21 Βασική Αρχιτεκτονική VAE

$$\mathcal{L}(\theta, \varphi; x^{(i)}) = -D_{KL}(q_\varphi(z|x^{(i)})||p_\theta(z)) + \mathbb{E}_{q_\varphi(z|x^{(i)})}[\log p_\theta(x^{(i)}|z)] \quad (2.29)$$

Για να απλοποιηθεί η εξίσωση αυτή, έστω ότι τόσο το $q_\varphi(z|x^{(i)})$ όσο $p_\theta(z)$ είναι Γκαουσιανά. Τότε θα ισχύει ότι:

$$\begin{aligned} \int q_\theta(z) \log p(z) dz &= \int \mathcal{N}(z; \mu, \sigma^2) \log \mathcal{N}(z; 0, I) dz \\ &= -\frac{J}{2} \log(2\pi) - \frac{1}{2} \sum_{j=1}^J ((\mu_j^{(i)})^2 + (\sigma_j^{(i)})^2) \end{aligned} \quad (2.30)$$

Και:

$$\begin{aligned} \int q_\theta(z) \log q_\theta(z) dz &= \int \mathcal{N}(z; \mu, \sigma^2) \log \mathcal{N}(z; \mu, \sigma^2) dz \\ &= -\frac{J}{2} \log(2\pi) - \frac{1}{2} \sum_{j=1}^J (1 + \log((\sigma_j^{(i)})^2)) \end{aligned} \quad (2.31)$$

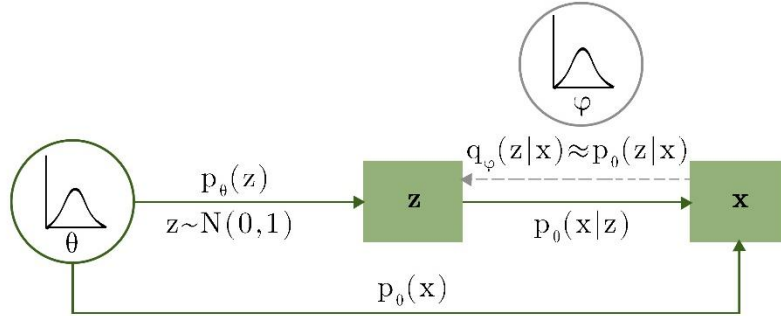
Οπότε σύμφωνα με το (2.30) και το (2.31), έχουμε ότι:

$$\begin{aligned} -D_{KL}(q_\varphi(z|x^{(i)})||p_\theta(z)) &= \int q_\theta(z) (\log p_\theta(z) - \log q_\theta(z)) dz \\ &= \frac{1}{2} \sum_{j=1}^J (1 + \log((\sigma_j^{(i)})^2) - (\mu_j^{(i)})^2 - (\sigma_j^{(i)})^2) \end{aligned} \quad (2.32)$$

Έτσι προκύπτει η πιο απλή μορφή της εξίσωσης Σφάλματος των VAE:

$$\mathcal{L}(\theta, \varphi; x^{(i)}) \cong \frac{1}{2} \sum_{j=1}^J (1 + \log((\sigma_j^{(i)})^2) - (\mu_j^{(i)})^2 - (\sigma_j^{(i)})^2) + \frac{1}{L} \sum_{l=1}^L \log p_\theta(x^{(i)}|z^{(i,l)}) \quad (2.33)$$

Όπου $z^{(i,l)} = \mu^{(i)} + \sigma^{(i)} \odot \varepsilon^{(l)}$ και $\varepsilon^{(l)} \sim \mathcal{N}(0, I)$



Εικόνα 2.22 Γραφικό Μοντέλο που εμπλέκεται στο VAE.

Ο Αλγόριθμος 2 συνοψίζει την κύρια διαδικασία εκπαίδευσης των VAE, υιοθετώντας την λογική βελτιστοποίησης **μικρής παρτίδας** (*mini batch*). Οι ενημερώσεις των κλίσεων μπορούν να χρησιμοποιούν οποιονδήποτε τυπικό κανόνα εκμάθησης με **βάση την κλίση** (*gradient-based learning rule*). Η παρακάτω υλοποίηση του αλγορίθμου Εκπαίδευσης VAEs βασίζεται στην υλοποίηση στο πρωτότυπο επιστημονικό άρθρο που αναφέρεται για πρώτη

φορά στους Μεταβλητούς Αυτοκωδικοποιητές (*Auto-Encoding Variational Bayes 2014*, [VAE14]).

Αλγόριθμος 2 – Εκπαίδευση Μεταβλητού Αυτοκωδικοποιητή

Είσοδος: $points \geq 100$ ➤ Αριθμός σημείων ανά δείγμα

Είσοδος: $batch_{mini} \geq 1$ ➤ Πλήθος δειγμάτων παρτίδας (*batch size*). Σε αυτή την περίπτωση μικρή παρτίδα (*minibatch*).

1: **ΑΛΓΟΡΙΘΜΟΣ ΔΙΑΔΙΚΑΣΙΑ ΕΚΠΑΙΔΕΥΣΗΣ**($batch_{mini}$)

2: **ΑΡΧΙΚΟΠΟΙΗΣΗ** θ, φ

3: **Όσο** θ και φ δεν συγκλίνουν, **επανάλαβε**

4: $\{x^{(1)}, \dots, x^{(batch_{mini})}\} \leftarrow \text{ΔΕΙΓΜΑΤΟΛΗΨΙΑ}(batch_{mini}, p_{data}(x), points)$

5: $\{\varepsilon^{(1)}, \dots, \varepsilon^{(batch_{mini})}\} \leftarrow \text{ΔΕΙΓΜΑΤΟΛΗΨΙΑΘΟΡΥΒΟΥ}(batch_{mini}, p_{\varepsilon}(\varepsilon))$

6: $g \leftarrow \text{ΥΠΟΛΟΓΙΣΕΚΛΙΣΗ}(\varepsilon, \theta, \varphi, x)$

$$\nabla_{\theta, \varphi} \left\{ \frac{1}{2} \sum_{j=1}^J \left(1 + \log((\sigma_j^i)^2) - (\mu_j^i)^2 - (\sigma_j^i)^2 \right) + \frac{1}{batch_{mini}} \sum_{l=1}^{batch_{mini}} \log p_{\theta}(x^{(i)} | \mu^{(i)} + \sigma^{(i)} \odot \varepsilon^{(l)}) \right\}$$

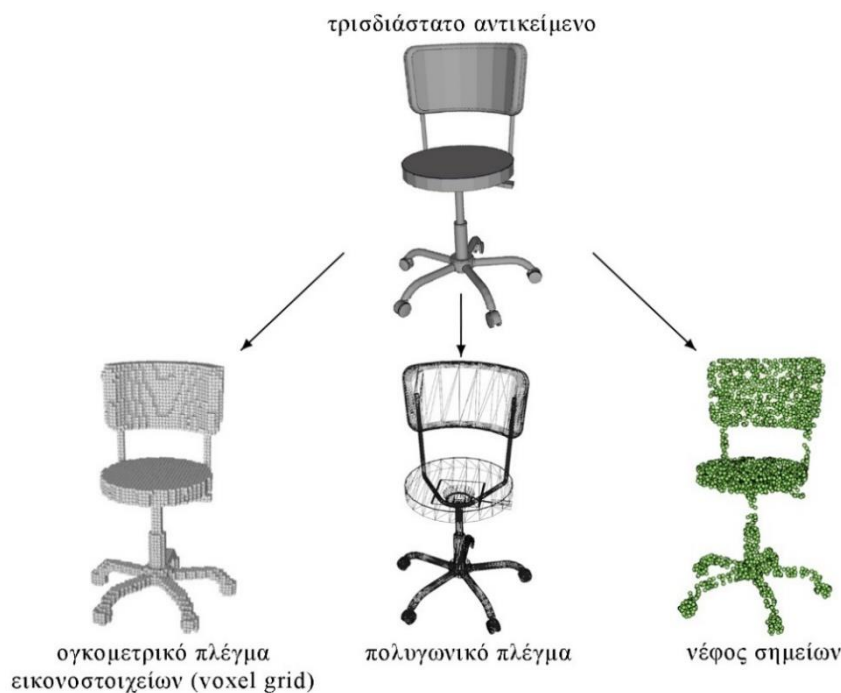
7: $\theta, \varphi \leftarrow \text{ΕΝΗΜΕΡΩΣΗ ΠΑΡΑΜΕΤΡΩΝ}(\theta, \varphi, g)$

8: **Τέλος Επανάληψης**

9: **Τέλος Αλγόριθμου**

2.6 Αναπαράσταση σχήματος

Ένα από τα θεμελιώδη θέματα στον τομέα της Όρασης Υπολογιστών και των Γραφικών Υπολογιστών (*Computer Graphics*) είναι η **αναπαράσταση τρισδιάστατων σχημάτων**. Σε αντίθεση με τις δισδιάστατες εικόνες οι οποίες παρουσιάζονται ως συστοιχίες εικονοστοιχείων, τα τρισδιάστατα δεδομένα μπορούν να αναπαρασταθούν με πολλαπλούς τρόπους, όπως το **ογκομετρικό πλέγμα εικονοστοιχείων** (*volumetric pixel grid* ή *voxel grid*), το **πολυγωνικό πλέγμα** (*polygonal mesh*), καθώς και το **νέφος σημείων** (*point cloud*).

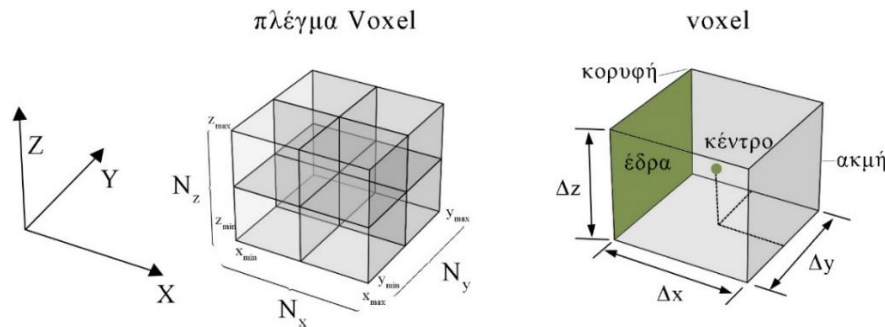


Εικόνα 2.23 Αναπαράσταση 3D σχήματος. Απεικόνιση τρισδιάστατης καρέκλας με τους τρεις διαφορετικούς τρόπους αναπαράστασης τρισδιάστατων σχημάτων.

2.6.1 Ογκομετρικό πλέγμα εικονοστοιχείων

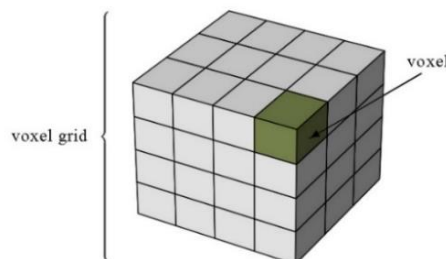
Η λέξη ογκομετρικό εικονοστοιχείων, **voxel** στα αγγλικά, είναι η συντομογραφία των λέξεων **όγκος** “**volume**” και **στοιχείο** “**element**”. Αποτελεί το τρισδιάστατο εννοιολογικό αντίστοιχο του δισδιάστατου εικονοστοιχείου pixel. Κάθε voxel αποτελεί μια κβαντική μονάδα όγκο και έχει μια αριθμητική τιμή, η οποία αντιπροσωπεύει ορισμένες μετρήσιμες ιδιότητες ή ανεξάρτητες μεταβλητές ενός πραγματικού αντικειμένου. Σε ένα ογκομετρικό πλέγμα εικονοστοιχείων καθορίζεται η σχετική τοποθεσία κάθε επιμέρους voxel για τη δημιουργία μιας μοναδικής δομής ογκομετρικού συνόλου δεδομένων δηλαδή ενός τρισδιάστατου αντικειμένου [K87] [KCY93].

Ορισμός 2.1. Έστω ότι έχουμε ένα ογκομετρικό εικονοστοιχείο v_i ενός συνόλου ογκομετρικών εικονοστοιχείων V . Για κάθε $v_i \in V$ ισχύει ότι $v_i = \{(x_i, y_i, z_i), w_i\}$ με $i \in \mathbb{N}$, όπου w_i αντιπροσωπεύει την τιμή κάποιων ιδιοτήτων των δεδομένων σε μια τοποθεσία $loc_i \subseteq \mathbb{R}^3$ με συντεταγμένες (x_i, y_i, z_i) . Κάθε ογκομετρικό εικονοστοιχείο v_{x_i, y_i, z_i} έχει διαστάσεις i, j, k όπου $i = j = k$.



Εικόνα 2.24 Δομή voxel και πλέγματος voxel. Ένα voxel απεικονίζεται ως ένας κύβος. Ένα πλέγμα από voxel αποτελείται από πολλαπλά voxel-κύβους.

Ορισμός 2.2. Ένα πλέγμα ογκομετρικών εικονοστοιχείων V είναι ένα σύνολο από ογκομετρικά εικονοστοιχεία $\{v_1, v_2, \dots, v_N\}$ και ισχύει ότι $V = \{x_i, y_i, z_i, w_i \mid i \in \mathbb{N}\}$ [KCY93].



Εικόνα 2.25 Voxel μέσα σε voxel Grid. Ένα ογκομετρικό πλέγμα από εικονοστοιχεία στο οποίο κάθε επιμέρους κύβος είναι ένα ογκομετρικό εικονοστοιχείο (voxel).

Το σύνολο τιμών w_i σε ένα ογκομετρικό πλέγμα εικονοστοιχείων διαφέρει από πλέγμα σε πλέγμα, δηλαδή η ιδιότητα που εκφράζουν αυτές οι τιμές μπορεί να είναι διαφορετική ανάμεσα σε δύο πλέγματα. Μπορεί να είναι απλά δυαδική, δηλαδή να λαμβάνει 0 για το φόντο και 1 για το αντικείμενο. Μπορεί όμως να είναι πιο σύνθετη και να αντιπροσωπεύει μια ιδιότητα όπως το χρώμα, η πυκνότητα, η θερμότητα ή και η πίεση. [K87]

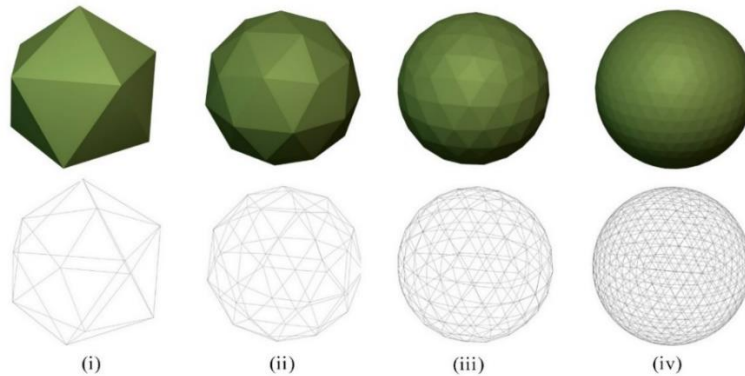
Ενώ το ογκομετρικό πλέγμα εικονοστοιχείων μπορεί να θεωρηθεί ως ένα **σύννεφο κβαντισμένων σημείων** (*quantized point cloud*) **σταθερού μεγέθους**, στην πράξη η χρήση τους στην τρισδιάστατη μοντελοποίηση είναι ανέφικτη λόγω της αραιότητας των αναπαρα-

στάσεων τους και της αύξησης των απαραίτητων υπολογιστικών πόρων με την αύξηση της λεπτομέρειας.

2.6.2 Πολυγωνικό Πλέγμα

Το **πολυγωνικό πλέγμα** είναι μια συλλογή από **κορυφές**, **ακμές** και **έδρες**, που μαζί προσεγγίζουν την αναπαράσταση ενός τρισδιάστατου αντικειμένου. Υπάρχουν ορισμένοι κανόνες για να κρίνουμε αν μια απεικόνιση ταιριάζει με τον ορισμό του πολυγωνικού πλέγματος. Δεν επιτρέπονται μεμονωμένες κορυφές και μεμονωμένες άκρες ή πολυγραμμές (*polylines*). Κάθε κορυφή πρέπει να μοιράζεται τουλάχιστον από μια άκρη, και κάθε επιμέρους άκρη πρέπει να μοιράζεται τουλάχιστον από μια έδρα. Τέλος, δεν είναι επιτρεπτή η αλληλοδιείσδυση των εδρών, δηλαδή η άκρη μιας έδρας δεν μπορεί να βρίσκεται στο εσωτερικό μιας άλλης έδρας [SE02] [FG08]. Οι έδρες συνήθως απαρτίζονται από τρίγωνα (*τριγωνικό πλέγμα*), τετράπλευρα ή άλλα απλά κυρτά πολύγωνα. Ο πιο συχνός τύπος πολυγωνικού πλέγματος είναι το **τριγωνικό πλέγμα**. [BKPAL10][SE02]

Ορισμός 2.3. Ένα τριγωνικό πλέγμα $M = (V, F)$ ορίζεται από ένα σύνολο κορυφών $V \subseteq \mathbb{R}^3$ και ένα σύνολο τριγωνικών όψεων $F \subseteq \{1, 2, \dots, |V|\}^3$ έτσι ώστε $f = f_1, f_2, f_3 \in F$ να ορίζει μια τριγωνική όψη που περικλείεται από τις αντίστοιχες κορυφές v_{f_1}, v_{f_2} και v_{f_3} . Μέσω των όψεων καθορίζονται έμμεσα και οι ακμές $E(F)$ μεταξύ των κορυφών. [BKPAL10]



Εικόνα 2.26 Αναπαράσταση σφαίρας με Πολυγωνικό Πλέγμα. Χρήση τριγωνικού πλέγματος για αναπαράσταση σφαίρας (i) υποδιαίρεση 1, (ii) υποδιαίρεση 2, (iii) υποδιαίρεση 3, (iv) υποδιαίρεση 4 [F01]. Παρατηρούμε πως όταν έχουμε μικρότερο αριθμό πολυγώνων είναι πολύ πιο δύσκολο να διακρίνουμε το γενικό σχήμα ενός αντικειμένου, στην προκειμένη περίπτωση μιας σφαίρας. Όσο αυξάνεται ο αριθμός των υποδιαίρεσεων – πολυγώνων του σχήματος τόσο περισσότερο μοιάζει το πολυγωνικό πλέγμα με το αρχικό σχήμα [SC18].

Κύριο κριτήριο για την επιλογή συγκεκριμένου πολυγώνου για τις έδρες ενός πλέγματος είναι η απλότητα αποθήκευσης και η ευκολία χρήσης στις διάφορες εφαρμογές. Συνεπώς, κατά κανόνα οι περισσότερες εφαρμογές υποστηρίζουν μόνο τριγωνικά πλέγματα. Σε μια πιο ρεαλιστική αναπαράσταση, ένα πλέγμα θα αποτελούνταν από πολύγωνα με οπές κα-

θώς και κοίλα πολύγωνα, το οποίο θα είχε ως αποτέλεσμα την περιπλοκή της εφαρμογής και του χειρισμού των αντικειμένων.

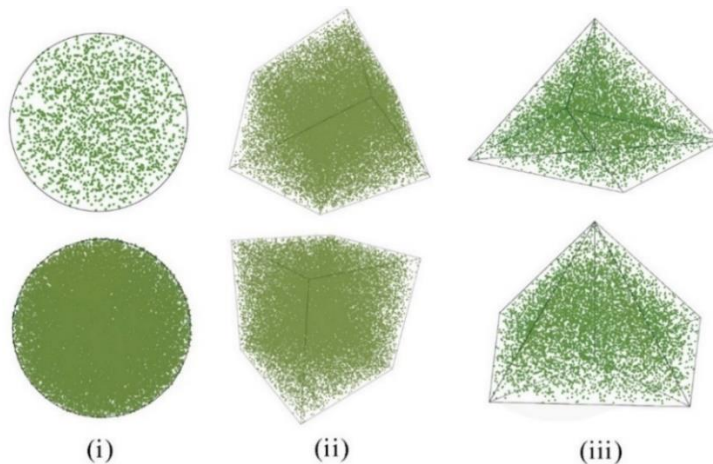
2.6.3 Νέφος Σημείων

Το **νέφος σημείων** είναι η πιο απλή αναπαράσταση τρισδιάστατων αντικειμένων μιας και αποτελεί μια **συλλογή τρισδιάστατων σημείων** που διανέμονται σε έναν τρισδιάστατο χώρο, χωρίς να υπάρχει μεταξύ τους συνδεσιμότητα. Ωστόσο, λόγω του μεγάλου βαθμού ελευθερίας που προκύπτει από την έλλειψη τοπικών συνδέσεων μεταξύ των σημείων η σύνθεσή τους με μεγάλη ακρίβεια είναι πολύ δύσκολη.

Ορισμός 2.4. Ένα νέφος σημείων P είναι ένα σύνολο από σημεία $\{p_1, p_2, \dots, p_N\}$ με διαστάσεις $[x, y, z]$, όπου x είναι το πλάτος, y είναι το ύψος και z είναι το βάθος. [CSKG17]

$$P = \{p_1, p_2, \dots, p_N\} \subseteq \mathbb{R}^3,$$

$$P = \{x_i, y_i, z_i \mid i \in \mathbb{N}\} \quad (2.34)$$



Εικόνα 2.27 Βασικά τρισδιάστατα σχήματα ως νέφη σημείων. Αναπαράσταση τριών βασικών τρισδιάστατων σχημάτων, (i) σφαίρας, (ii) κύβου, και (iii) πυραμίδας, σε μορφή νέφους σημείων. Στο σχήμα (i) βλέπουμε δύο διαφορετικές αναπαραστάσεις σφαίρας με διαφορετικό πλήθος σημείων, η κάτω σφαίρα κάτω έχει 2 έως 3 φορές περισσότερα σημεία από την πάνω σφαίρα. Στα σχήματα (ii) και (iii) έχουμε αναπαράσταση διαφορετικών προοπτικών από τις οποίες παρατηρούμε το τρισδιάστατο σχήμα.

Χρησιμοποιείται κατά κύριο λόγο σε εφαρμογές οι οποίες σχετίζονται με την κατανόηση σκηνής όπως η αυτόνομη οδήγηση ή η ρομποτική, σε εφαρμογές ανακατασκευής τρισδιάστατων μοντέλων (*3D Model Reconstruction*), καθώς και στην Επαυξημένη πραγματικότητα (*Augmented Reality*) και την Εικονική πραγματικότητα (*Virtual Reality*). Τα αντικείμενα σε μορφή νέφους σημείων παράγονται μέσω αισθητήρων βάθους, όπως LIDAR (*LIght Detection And Ranging*) και κάμερες RGB-D (όπου το D αντιπροσωπεύει το βάθος) [GWHL20].

3. Σύνολο Δεδομένων

3.1 ShapeNet

Το πρώτο βήμα για την σωστή εκπαίδευση ενός μοντέλου, είναι η σωστή επιλογή συνόλου δεδομένων. Όπως φαίνεται στο Πίνακα, υπάρχει μια πληθώρα διαθέσιμων συνόλων δεδομένων με τρισδιάστατα αντικείμενα.

Σύνολο Δεδομένων	Κλάσεις	Αντικείμενα	Τυπική Απόκλιση
IKEA	7	219	16.4
ShapeNet			
ShapeNetCore	55	51 300	1589.7
ShapeNetSem	270	12 000	343.3
PartNet	24	26 600	322.4
ObjectNet3D	100	44 147	42.0
Pascal3D+	12	79	1.9
ModelNet10	10	4 899	243.9
ModelNet40	40	12 311	215.7
MCB	68	58 700	1040.1
ABC	Μη Διαθέσιμο	1 000 000	Μη Διαθέσιμο
Pix3D	9	395	64.5

Πίνακας 3.1 Διαθέσιμα Σύνολα Δεδομένων. Σύνολα δεδομένων με πληροφορίες όπως αριθμό κλάσεων, καθώς και ο αριθμό τρισδιάστατων μοντέλων αντικειμένων. Επιπρόσθετα έγινε υπολογισμός της τυπικής απόκλισης κάθε συνόλου, για τον αριθμό τρισδιάστατων μοντέλων ανά κλάση.

Στην παρούσα εργασία, ως σύνολο δεδομένων επιλέχθηκε το **ShapeNet** [SHAPE15] το οποίο είναι ευρέως χρησιμοποιημένο από την ερευνητική κοινότητα, για την επίλυση προβλημάτων που έχουν σχέση με τα τρισδιάστατα δεδομένα. Πρόκειται για μια συνεχή συνεργατική προσπάθεια μεταξύ ερευνητών στο Princeton, στο Stanford και στο TTIC, για τη δημιουργία ενός πλούσια σχολιασμένου συνόλου τρισδιάστατων σχημάτων. Η ομάδα του ShapeNet παρέχει τέσσερα διαφορετικά υποσύνολα δεδομένων τα οποία μπορούν να χρησιμοποιηθούν για ερευνητικούς σκοπούς, 1. ShapeNetCore, 2. ShapeNetSem, 3. PartNet, και 4. Virtual Scans.

ShapeNetCore

Το συγκεκριμένο υποσύνολο του ShapeNet αποτελείται μόνο από καθαρά τρισδιάστατα μοντέλα, καθώς και κατηγοριοποίηση επαληθευμένη από ανθρώπους, και ευθυγραμμισμένους σχολιασμούς (*alignment annotations*). Καλύπτει 55 κατηγορίες αντικειμένων, π.χ. καρέκλα, αυτοκίνητο, με περίπου 51300 μοναδικά τρισδιάστατα μοντέλα. Υπάρχουν δύο εκδόσεις του υποσυνόλου αυτού, 1. ShapeNetCore v1 (Ιούλιος 2015), και 2. ShapeNetCore v2 (Φθινόπωρο 2016). Συγκεκριμένα θα χρησιμοποιήσουμε την δεύτερη έκδοση του υποσυνόλου αυτού.

Κάθε στοιχείο του υποσυνόλου αποτελείται από τα εξής αρχεία:

- **.json** – μετα-δεδομένα (*metadata*) για την ομαλοποίηση του τρισδιάστατου μοντέλου (πλαίσιο οριοθέτησης (*bounding box*), κεντροειδές (*centroid*)), καθώς και τα στατιστικά του μοντέλου (*αριθμός κορυφών*)
- **.obj** – τρισδιάστατο πλέγμα σε μορφή OBJ
- **.mlt** – MLT (*Wavefront Material Template Library*). Αρχείο υφών για το OBJ. Περιλαμβάνει πληροφορίες όπως ονόματα αρχείων με υφές καθώς και των ιδιοτήτων υλικών. Βοηθάει στην σωστή εφαρμογή υφών στο αντικείμενο που απεικονίζεται στο OBJ αρχείο.
- **.solid.binvox** – Συμπληρωμένο δυαδικό ογκομετρικό πλέγμα εικονοστοιχείων του τρισδιάστατου μοντέλου
- **.binvox** – Δυαδικό ογκομετρικό πλέγμα εικονοστοιχείων των επιφανειών του τρισδιάστατου μοντέλου
- **Images/.png** – υφές για το τρισδιάστατο μοντέλο
- **Screenshots/.png** – δισδιάστατες απεικονίσεις του τρισδιάστατου μοντέλου από διαφορετικές οπτικές γωνίες

Οι 55 κατηγορίες του ShapeNetCore v2 είναι οι εξής:

ID	Όνομα ελληνικά	Όνομα αγγλικά
2691156	αεροπλάνο	airplane, aeroplane, plane
2747177	σκουπιδοτενεκές	ashcan, trash can, garbage can, wastebin, ash bin, ash-bin, ashbin, dustbin, trash barrel, trash bin
2773838	τσάντα	bag, traveling bag, travelling bag, grip, suitcase
2801938	καλάθι	basket, handbasket
2808440	μπανιέρα	bathtub, bathing tub, bath, tub

2818832	κρεβάτι	bed
2828884	παγκάκι	bench
2843684	σπίτι πουλιών	birdhouse
2871439	ράφι	bookshelf
2876657	μπουκάλι	bottle
2880940	μπολ	bowl
2924116	λεωφορείο	bus, autobus, coach, charabanc, double-decker, jitney, motorbus, motorcoach, omnibus, passenger vehi
2933112	ντουλάπι	cabinet
2942699	κάμερα	camera, photographic camera
2946921	τενεκές	can, tin, tin can
2954340	καπέλο	cap
2958343	αυτοκίνητο	car, auto, automobile, machine, motorcar
3001627	καρέκλα	chair
3046257	ρολόι	clock
3085013	υπολογιστής	computer keyboard, keypad
3207941	πλυντήριο πιάτων	dishwasher, dish washer, dishwashing machine
3211117	οθόνη	display, video display
3261776	ακουστικά	earphone, earpiece, headphone, phone
3325088	βρύση	faucet, spigot
3337140	αρχείο	file, file cabinet, filing cabinet
3467517	κιθάρα	guitar
3513137	κράνος	helmet
3593526	δοχείο	jar
3624134	μαχαίρι	knife
3636649	λάμπα	lamp
3642806	φορητός υπολογιστής	laptop, laptop computer
3691459	ηχείο	loudspeaker, speaker, speaker unit, loudspeaker system, speaker system
3710193	γραμματοκιβώτιο	mailbox, letter box
3759954	μικρόφωνο	microphone, mike
3761084	φούρνος μικροκυμάτων	microwave, microwave oven
3790512	μοτοσικλέτα	motorcycle, bike
3797390	κούπα	mug
3928116	πιάνο	piano, pianoforte, forte-piano
3938244	μαξιλάρι	pillow
3948459	πιστόλι	pistol, handgun, side arm, shooting iron

διαστάσεις, εκτιμήσεις της σύνθεσής τους σε επίπεδο κατηγορίας και εκτιμήσεις του συνολικού όγκου και βάρους τους.

PartNet

Το υποσύνολο PartNet παρέχει λεπτομερή σχολιασμό μικρότερων τμημάτων τρισδιάστατων αντικειμένων που υπάρχουν στο ShapeNetCore. Συγκεκριμένα από τις 55 κατηγορίες του ShapeNetCore, το PartNet περιέχει τις 24 με περίπου 26700 τρισδιάστατα μοντέλα και 573600 τμήματα αντικειμένων.

Virtual Scans

Αυτό το υποσύνολο δεδομένων παρέχει εικονικά σαρωμένα (*virtually scanned*) τμηματικά μοντέλα (*partial models*) από το ShapeNetCore, καθώς και τους αντίστοιχους μετασχηματισμούς απόστασης των πλήρων μοντέλων (*complete models*). Αποτελείται από τις εικονικές σαρώσεις 8 κατηγοριών του ShapeNet (με 31400 τρισδιάστατα μοντέλα), σε ζεύγη περικομμένου υπογεγραμμένου πεδίου απόστασης και υπογεγραμμένου πεδίου απόστασης σε αναλύσεις 32^3 και 128^3 .

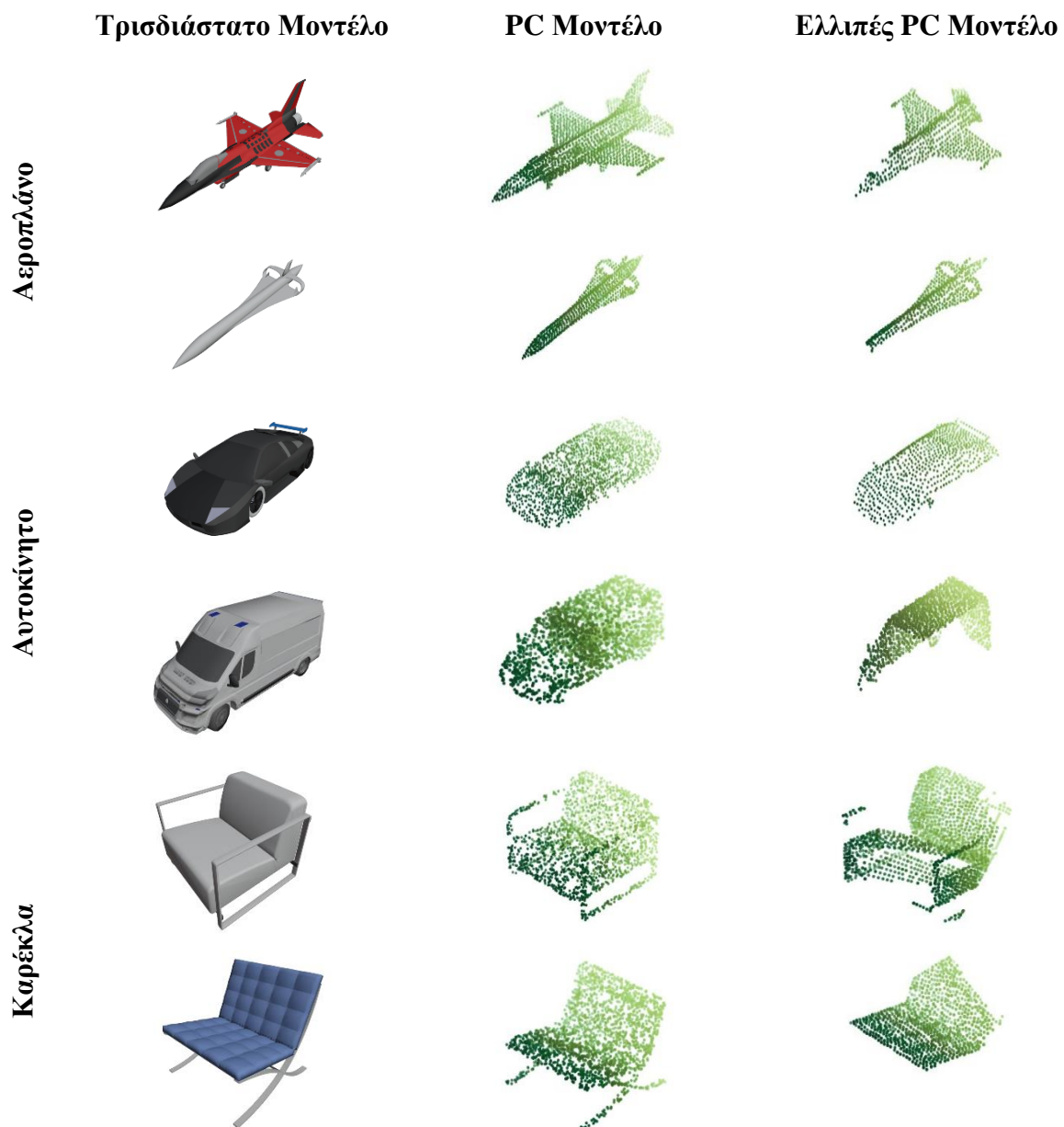
Από τα 4 αυτά υποσύνολα του ShapeNet επιλέχθηκε το ShapeNetCore v2. Συγκεκριμένα από τις 55 κατηγορίες αντικειμένων που περιείχε, έγινε επιλογή 5 κατηγοριών με τις οποίες εκπαιδεύτηκε το μοντέλο. Χρησιμοποιήθηκαν οι κατηγορίες: *Αεροπλάνο*, *Αυτοκίνητο*, *Καρέκλα*, *Τραπέζι* και *Λάμπα*.

Από τα τρισδιάστατα μοντέλα έγινε δημιουργία των αντίστοιχων **Μοντέλων Σημείων Νέφους** (*PC Models*), καθώς και των αντίστοιχων **ελλιπών Μοντέλων Σημείων Νέφους**, όπως βλέπουμε στην τελευταία στήλη στον [Πίνακα 3.3](#). Τα ελλιπή Μοντέλα Σημείων Νέφους δημιουργήθηκαν με την αφαίρεση τυχαίων κομματιών από τα αρχικά Μοντέλα Σημείων Νέφους, έτσι ώστε τα σημεία να είναι τα μισά από τα αρχικά. Ένας άλλος τρόπος που θα μπορούσε να γίνει αυτό θα ήταν με τη χρήση του PartNet υποσυνόλου του ShapeNet και την αφαίρεση τμημάτων, στα οποία έχει χωριστεί το κάθε μοντέλο.

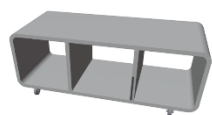
Στην συνέχεια έγινε διαχωρισμός σε υποσύνολο εκπαίδευσης, επαλήθευσης και ελέγχου. Επειδή οι κλάσεις αυτές ήταν ανισόρροπες, με τις κλάσεις «αυτοκίνητο», «καρέκλα», και «τραπέζι», να έχουν πάνω από 5500 συνολικά μοντέλα, την κλάση «αεροπλάνο» να έχει περίπου 3400 μοντέλα, και την κλάση «λάμπα» να έχει ούτε 2200 μοντέλα, ο διαχωρισμός τους ήταν ένα μεγάλο πρόβλημα από μόνο του. Χρησιμοποιώντας την πιο κλασική μέθοδο διαχωρισμού, δηλαδή 80% του συνολικού συνόλου δεδομένων στην εκπαίδευση και την επαλήθευση (με το 80% αυτού να χρησιμοποιείται για την εκπαίδευση και το 20% για την ε-

παλήθευση), και 20% του συνολικού συνόλου δεδομένων στον έλεγχο, η κλάση «λάμπα» καταλήγει με αρκετά λιγότερα δεδομένα εκπαίδευσης (περίπου 1387) από τις υπόλοιπες κλάσεις. Έτσι για να μην έχουμε τόσο μεγάλη απόκλιση αποφασίστηκε να γίνει ο εξής διαχωρισμός:

- 1) Σύνολο επαλήθευσης – 100 τρισδιάστατα μοντέλα ανά κλάση
- 2) Σύνολο ελέγχου – 400 τρισδιάστατα μοντέλα ανά κλάση
- 3) Σύνολο εκπαίδευσης – όλα τα υπόλοιπα μοντέλα που έμειναν ανά κλάση.



Τραπεζι



Λάμπα



Πίνακας 3.3 Τρισδιάστατο Μοντέλο σε μορφή PC και Ελλιπούς PC.

4. Μεθοδολογία

4.1 Σχετικές Έρευνες

Η ολοκλήρωση τρισδιάστατων σχημάτων (*3D Shape Completion*) συχνά χρησιμοποιείται για τον καθαρισμό σπασμένων τρισδιάστατων μοντέλων CAD. Πάρα πολλές μέθοδοι έχουν προταθεί για την αντιμετώπιση του προβλήματος αυτού.

Οι **συμβατικές μέθοδοι** επίλυσης του προβλήματος αυτού, κατηγοριοποιούνται σε προσεγγίσεις με **βάση τη γεωμετρία** (*geometry-based*), με **βάση την ευθυγράμμιση** (*alignment-based*), και με **βάση την μάθηση** (*learning-based*). Πολλές μέθοδοι ακολουθούν μια παρόμοια πορεία με προσεγγίσεις ολοκλήρωσης δισδιάστατων εικόνων, και έτσι Τρισδιάστατα Συνελκτικά Νευρωνικά Δίκτυα τα οποία εφαρμόζονται σε ογκομετρικά εικονοστοιχεία έχουν υιοθετηθεί ευρέως για την πρόκληση αυτή [SC18], [HLHKY17]. Επιπρόσθετα έχουν αναπτυχθεί και μέθοδοι που αντί να χρησιμοποιούν ογκομετρικά εικονοστοιχεία, δουλεύουν άμεσα σε νέφη σημείων [WCZC20], [CCM20], [MSN19], [PCN18], [ADMG18].

Οι πρόσφατες εξελίξεις στον τομέα αυτό υποδηλώνουν πως η **προσέγγιση** που βασίζεται σε **δεδομένα** (*data-driven*), δηλαδή με βάση τη μάθηση, είναι η πιο κατάλληλη για την επίλυση του προβλήματος. Συγκεκριμένα η **συμπλήρωση των ελλিপών σχημάτων** γίνεται με την χρήση **προηγούμενης γνώσης**, και **υπαρχόντων δεδομένων**. Μια επιπλέον προσέγγιση κάνει χρήση ανιχνευμένων δομών και κανονικοτήτων στα τρισδιάστατα σχήματα, όπως οι συμμετρίες, για να συμπληρώσει τα δεδομένα που λείπουν.

4.2 Μοντέλο Εκπαίδευσης

Στο Θεωρητικό Υπόβαθρο αναφέρθηκαν δύο από τα κύρια είδη Βαθιών Γεννητικών Μοντέλων, **Γεννητικά Ανταγωνιστικά Δίκτυα** (*GAN*), και **Μεταβλητοί Αυτοκωδικοποιητές** (*VAE*). Ανάμεσα στα δύο αυτά είδη, τα GAN έχουν επωφεληθεί από περισσότερη επιστημονική συνεισφορά με αποτέλεσμα να υπάρχουν πάρα πολλές εφαρμογές και παραλλαγές των μοντέλων αυτών. Το υψηλό αυτό ενδιαφέρον για τα GAN οφείλεται κυρίως στην απλότητα της αντίπαλης εκπαίδευσης, σε σχέση με τα VAE που έχουν πολύ πιο υψηλό βαθμό πολυπλοκότητας μιας και συνδυάζουν θεωρητικές έννοιες όπως το πιθανολογικό μοντέλο (*probabilistic model*), και το μεταβλητό συμπέρασμα (*variational inference*). Βέβαια, πολλές ιδέες των Αυτοκωδικοποιητών Μεταβλητών έχουν συνδυαστεί με ιδέες Γεννητικών Ανταγωνιστικών Δικτύων, δημιουργώντας έτσι **υβρίδια Γεννητικών Μοντέλων**. Έτσι και σε αυτή

την διπλωματική εργασία η κύρια ιδέα του Μοντέλου Εκπαίδευσης είναι ο συνδυασμός των δύο αυτών Γεννητικών Μοντέλων.

Η αρχιτεκτονική του κωδικοποιητή που χρησιμοποιήθηκε για τον Μεταβλητό Αυτοκωδικοποιητή είναι βασισμένη στην αρχιτεκτονική του μοντέλου **PointNet** [POINT16]. Ο Αποκωδικοποιητής είναι ένα απλό **Πολυστρωματικό Αντίληπτρο (MLP)**, τριών στρωμάτων.

Μεταβλητός Αυτοκωδικοποιητής VAE

Κωδικοποιητής E VAE	
Στρώματα	Έξοδος
Conv1D+BN+LeakyRELU	(3, N)
Conv1D+BN+LeakyRELU	(64, N)
Conv1D+BN+LeakyRELU	(128, N)
Conv1D+BN+LeakyRELU	(128, N)
Conv1D+BN+LeakyRELU	(256, N)
Conv1D+BN+LeakyRELU	(128, N)
Global Max Pooling	(128,)

Αποκωδικοποιητής D VAE	
Στρώματα	Έξοδος
Latent Code	(128,)
FC+LeakyReLU	(256,)
FC+LeakyReLU	(256,)
FC+LeakyReLU	(Nx3,)
Reshape	(3, N)

FC: Πλήρως Συνδεδεμένο (Fully Connected)-Linear Layer

Latent Codes: Λανθάνοντες Κωδικοί

N: Αριθμός Σημείων στο Νέφος

VAE	Στρώματα	Έξοδος
mean	FC	(64,)
logvar	FC	(64,)

Πίνακας 4.1 Αρχιτεκτονική VAE – Στρώματα και Έξοδοι

Δοκιμάστηκε και μια επιπλέον αρχιτεκτονική VAE, πριν γίνει η επιλογή χρήσης Συνόλου Δεδομένων σε αναπαράσταση Σημείων Νέφους, η οποία δουλεύει για ογκομετρικά πλέγματα εικονοστοιχείων (voxels). Το μοντέλο ήταν πολύ πιο περίπλοκο και η εκπαίδευσή του πιο χρονοβόρα, και επιπλέον τα voxel σαν αναπαράσταση σχήματος χρειάζονται περισσότερη μνήμη, με αποτέλεσμα να γίνει η επιλογή χρήσης αναπαράστασης Σημείων Νέφους.

Η αρχιτεκτονική τόσο του Παραγωγού G όσο και του Διευκρινιστή D είναι όπως και ο Αποκωδικοποιητής απλά Πολυστρωματικά Αντίληπτρα, τριών στρωμάτων.

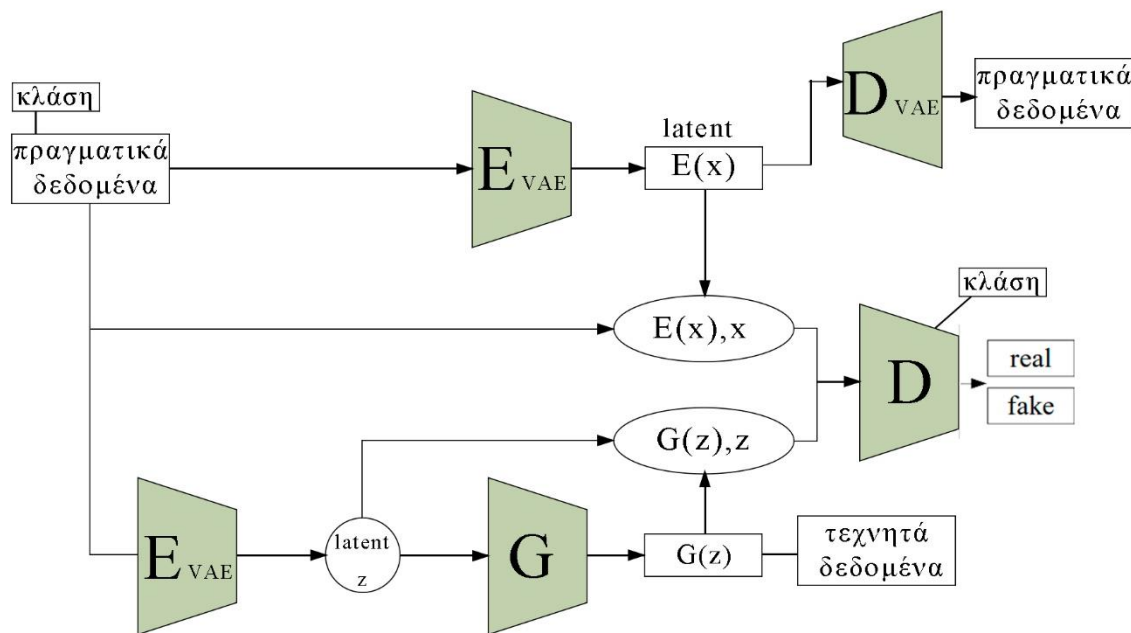
Γεννητικό Ανταγωνιστικό Δίκτυο GAN

Παραγωγός G		Διευκρινιστής D	
Στρώματα	Έξοδος	Στρώματα	Έξοδος
Concat(Latent, z noise)	(192,)	Latent codes	(128,)
FC+LeakyReLU	(256,)	FC+LeakyReLU	(256,)
FC+LeakyReLU	(512,)	FC+LeakyReLU	(512,)
FC	(128,)	FC	(1,)

FC: Πλήρως Συνδεδεμένο (*Fully Connected*)-*Linear Layer*

Latent Codes: Λανθάνοντες Κωδικοί

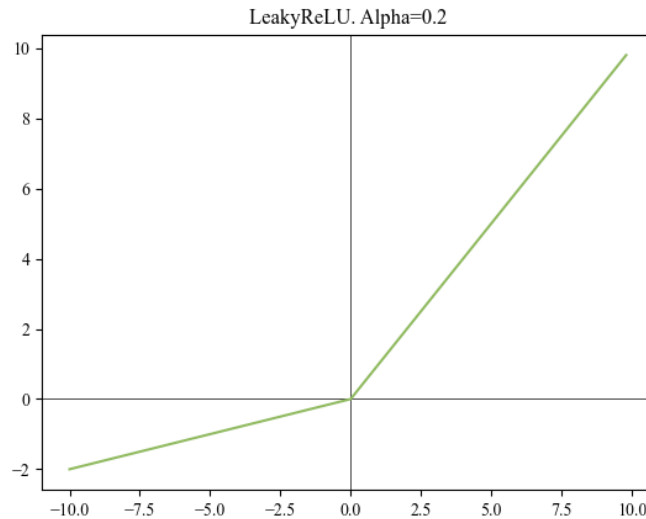
Πίνακας 4.2 Αρχιτεκτονική GAN – Στρώματα και Έξοδοι



Εικόνα 4.1 Η αρχιτεκτονική του Υβριδικού Γεννητικού Μοντέλου

Η συνάρτηση ενεργοποίησης που επιλέχθηκε τόσο για το VAE, όσο για το GAN είναι η **LeakyReLU**. Πρόκειται για μια παραλλαγή της συνάρτησης **Μονάδας Γραμμικού Ανορθωτή** (ReLU), η οποία αναφέρθηκε και στο [Θεωρητικό Υπόβαθρο](#). Η LeakyReLU υπολογίζεται ως εξής:

$$LeakyReLU\ x = \begin{cases} x, & x \geq 0 \\ \alpha x, & x < 0 \end{cases} \quad (4.1)$$



Εικόνα 4.2 Γραφική Παράσταση της Συνάρτησης Ενεργοποίησης LeakyReLU

Ο λόγος που επιλέχθηκε η LeakyReLU έναντι της απλής ReLU είναι, ότι σε δίκτυα που υπάρχει σημείο συμφόρησης (*bottleneck*), όπως το δίκτυο VAE, η ReLU “κλειδώνει” εμποδίζοντας τη διάδοση της κλίσης με αποτέλεσμα να μην μαθαίνει το Νευρωνικό Δίκτυο, ενώ η LeakyReLU συνεχίζει να διαδίδει την κλίση με αποτέλεσμα το Νευρωνικό Δίκτυο να συνεχίζει να εκπαιδεύεται [XWCL15].

Οι συγγραφείς του επιστημονικού άρθρου στο οποίο παρουσιάστηκε για πρώτη φορά το LeakyReLU προτείνουν την χρήση του συντελεστή κλίσης 0.1 [LRELU13]. Πολλές πηγές κάνουν χρήση του συντελεστή κλίσης 0.01.

4.3 Διαδικασία εκπαίδευσης

Πριν ξεκινήσει η διαδικασία εκπαίδευσης είναι απαραίτητο να κατανοήσουμε και να επιλέξουμε κάποιες παραμέτρους και βασικές έννοιες οι οποίες εμπλέκονται και αυτές στην εκπαίδευση του αλγορίθμου.

Βελτιστοποιητής

Ο ρόλος του **βελτιστοποιητή** (*optimizer*), είναι η ελαχιστοποίηση της εξίσωσης σφάλματος, συνδέοντας την εξίσωση σφάλματος με τις παραμέτρους του μοντέλου, ενημερώνοντας το μοντέλο ανάλογα την έξοδο της εξίσωσης σφάλματος. Με λίγα λόγια με βάση την εξίσωση σφάλματος ένας βελτιστοποιητής καταλαβαίνει πότε η εκπαίδευση οδεύει προς την σωστή κατεύθυνση και πότε όχι. Για την βελτιστοποίηση της εκπαίδευσης και των δύο μοντέλων έγινε χρήση του αλγορίθμου του **Προσαρμοστικού Εκτιμητή Ορμής** (Adaptive Moment Estimation – Adam). Είναι ο συνδυασμός δύο άλλων μηχανισμών βελτιστοποίησης,

του **RMSProp** και του **Momentum**, και χρησιμοποιεί κομμάτια των προηγούμενων κλίσεων στην τρέχουσα κλίση. Είναι πλέον ο πιο διαδομένος βελτιστοποιητής για την εκπαίδευση νευρωνικών δικτύων.

Εποχές

Οι **εποχές** (*epochs*) είναι ο αριθμός των φορών που ο αλγόριθμος εκπαίδευσης θα λειτουργήσει σε ολόκληρο το σύνολο δεδομένων εκπαίδευσης. Για μια εποχή κάθε δείγμα του συνόλου δεδομένων μπορεί να ενημερώσει τις εσωτερικές παραμέτρους του μοντέλου. Και στα δύο μοντέλα ο αριθμός εποχών εκπαίδευσης ήταν μεταβλητός ανάλογα με τα αποτελέσματα που έδινε για έναν x αριθμό εποχών. Ανά τακτά χρονικά διαστήματα με τη χρήση αυτόματης αποθήκευσης **σημείων ελέγχου** (*checkpoints*), μπορούσε να γίνει χρήση του τελευταίου για την συνέχεια εκπαίδευσης του μοντέλου, ή να γίνει επιλογή προγενέστερου σημείου με το βέλτιστο μοντέλο.

Παρτίδα

Το **μέγεθος παρτίδας** (*batch size*) είναι ο αριθμός παραδειγμάτων του συνόλου δεδομένων που χρησιμοποιείται για μια επανάληψη, δηλαδή για ένα προδρομικό και οπισθοδρομικό πέρασμα. Με λίγα λόγια για κάθε εποχή ο αριθμός των επαναλήψεων εξαρτάται από το μέγεθος της παρτίδας. Υπάρχουν 3 επιλογές:

- **Λειτουργία παρτίδας** (*batch mode*), όπου το μέγεθος παρτίδας είναι ίσο με το μέγεθος του συνόλου δεδομένων, και έτσι σε μια εποχή έχουμε μόνο μια επανάληψη,
- **Λειτουργία μικρής παρτίδας** (*mini-batch mode*), όπου το μέγεθος παρτίδας είναι μεγαλύτερο από το 1 αλλά μικρότερο από το μέγεθος του συνόλου δεδομένων.
- **Στοχαστική Λειτουργία** (*stochastic mode*), όπου το μέγεθος παρτίδας είναι ίσο με ένα.

Τόσο για την εκπαίδευση του VAE, όσο για την εκπαίδευση του GAN, χρησιμοποιήθηκε η λειτουργία μικρής παρτίδας.

Ρυθμός εκπαίδευσης

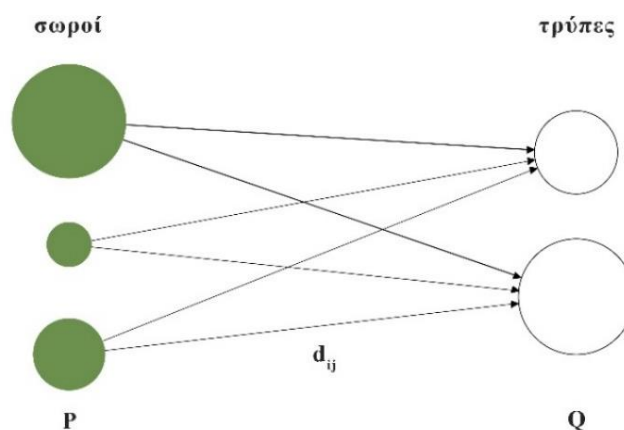
Ο **ρυθμός εκπαίδευσης** (*learning rate*) είναι μια παράμετρο εκπαίδευσης νευρωνικών δικτύων, της οποίας η τιμή κυμαίνεται ανάμεσα στο 0.0 και το 1.0, και ελέγχει το πόσο γρήγορα προσαρμόζεται το μοντέλο στο πρόβλημα. Στις περισσότερες περιπτώσεις όσο μικρότερη είναι η τιμή του, τόσες περισσότερες εποχές χρειάζεται το μοντέλο.

4.4 Μετρικές Απόστασης

Για την αξιολόγηση των αποτελεσμάτων των μοντέλων VAE και GAN, δηλαδή την σύγκριση ομοιότητας ανάμεσα στο μοντέλο που δημιουργήσαμε με το αληθινό μοντέλο σε μορφή PC, χρειάζεται να γίνει χρήση κάποιου μέτρου ομοιότητας. Στην εργασία χρησιμοποιήθηκαν 3 διαφορετικές μετρικές απόστασης σημείων για την αξιολόγηση τόσο κατά τη διάρκεια της εκπαίδευσης, όσο για την αξιολόγηση των αποτελεσμάτων πάνω στο σύνολο ελέγχου. Συγκεκριμένα για την εκπαίδευση του μοντέλου VAE καθώς και την αξιολόγηση των μοντέλων πάνω στα δεδομένα ελέγχου, χρησιμοποιήθηκε ως μέτρο αξιολόγησης η Απόσταση Ελάχιστου Έργου, για την εκπαίδευση του μοντέλου GAN χρησιμοποιήθηκε ως μέτρο αξιολόγησης η Απόσταση Hausdorff.

4.4.1 Απόσταση Ελάχιστου έργου

Το μέτρο το οποίο είναι το πιο συχνά χρησιμοποιούμενο σε εφαρμογές Ολοκλήρωσης Σχημάτων είναι το μέτρο Απόστασης Ελάχιστου έργου, ή αλλιώς **Earth Mover's Distance** - EMD. Το μέτρο αυτό υπολογίζει το ελάχιστο έργο που απαιτείται για την μετατροπή μιας κατανομής σε μία άλλη [EMD00]. Πιο συγκεκριμένα, το μέτρο αυτό επιλύει το **πρόβλημα της μεταφοράς**. Το πρόβλημα της μεταφοράς αποτελεί μια ειδική κατηγορία προβλημάτων του γραμμικού προγραμματισμού, η οποία ασχολείται με την αποστολή προϊόντων προερχόμενων από διάφορες πηγές σε διάφορους προορισμούς.



Εικόνα 3.2 Απεικόνιση του προβλήματος της μεταφοράς

Έστω ότι έχουμε δύο κατανομές:

$$P = \{p_i, w_{p_i}\}, i = 1..m\}$$

$$Q = \{q_j, w_{q_j}\}, j = 1..n\}$$

όπου p_i , και q_j τα διανύσματα θέσης κάθε κατανομής και w_{p_i} , w_{q_j} τα αντίστοιχα βάρη. Επιπρόσθετα, έστω $F(P, Q) = [f_{ij}]$ η ροή ποσού βάρους f_{ij} που μεταφέρεται από την θέση p_i στην θέση q_j . Το έργο της ροής υπολογίζεται ως εξής:

$$WORK(P, Q, F) = \sum_{i=1}^m \sum_{j=1}^n d_{ij} f_{ij}$$

Όπου $d_{ij} = |p_i - q_j|$ η απόσταση του σημείου p_i από το σημείο q_j . Για την εύρεση του ελάχιστου έργου πρέπει να γίνει πρώτα εύρεση της ροής $F(P, Q)$ η οποία το ελαχιστοποιεί.

Υπάρχουν 4 Περιορισμοί για τις ροές:

1. **Περιορισμός 1.** Όλες οι ροές πρέπει να είναι μη αρνητικές. Η μεταφορά από το P στο Q δεν είναι αντιστρέψιμη.

$$f_{ij} \geq 0, 1 \leq i \leq m, 1 \leq j \leq n$$

2. **Περιορισμός 2.** Το συνολικό βάρος του υποσυνόλου P' δεν μπορεί να υπερβαίνει το συνολικό βάρος όλου του συνόλου $P - W_P$.

$$\sum_{j=1}^n f_{ij} \leq w_{p_i}, 1 \leq i \leq m$$

3. **Περιορισμός 3.** Το συνολικό βάρος που γίνεται αποδεκτό από το Q δεν μπορεί να υπερβαίνει την συνολική χωρητικότητα του $Q - W_Q$.

$$\sum_{i=1}^m f_{ij} \leq w_{q_j}, 1 \leq j \leq n$$

4. **Περιορισμός 4.** Η συνολική ροή πρέπει να είναι όσο το δυνατόν μεγαλύτερη.

$$\sum_{i=1}^m \sum_{j=1}^n f_{ij} = \min \left(\sum_{i=1}^m w_{p_i}, \sum_{j=1}^n w_{q_j} \right)$$

Εφόσον βρεθούν οι ροές οι οποίες ελαχιστοποιούν το έργο, τότε η εξίσωση που υπολογίζει το Earth Mover's Distance είναι η εξής:

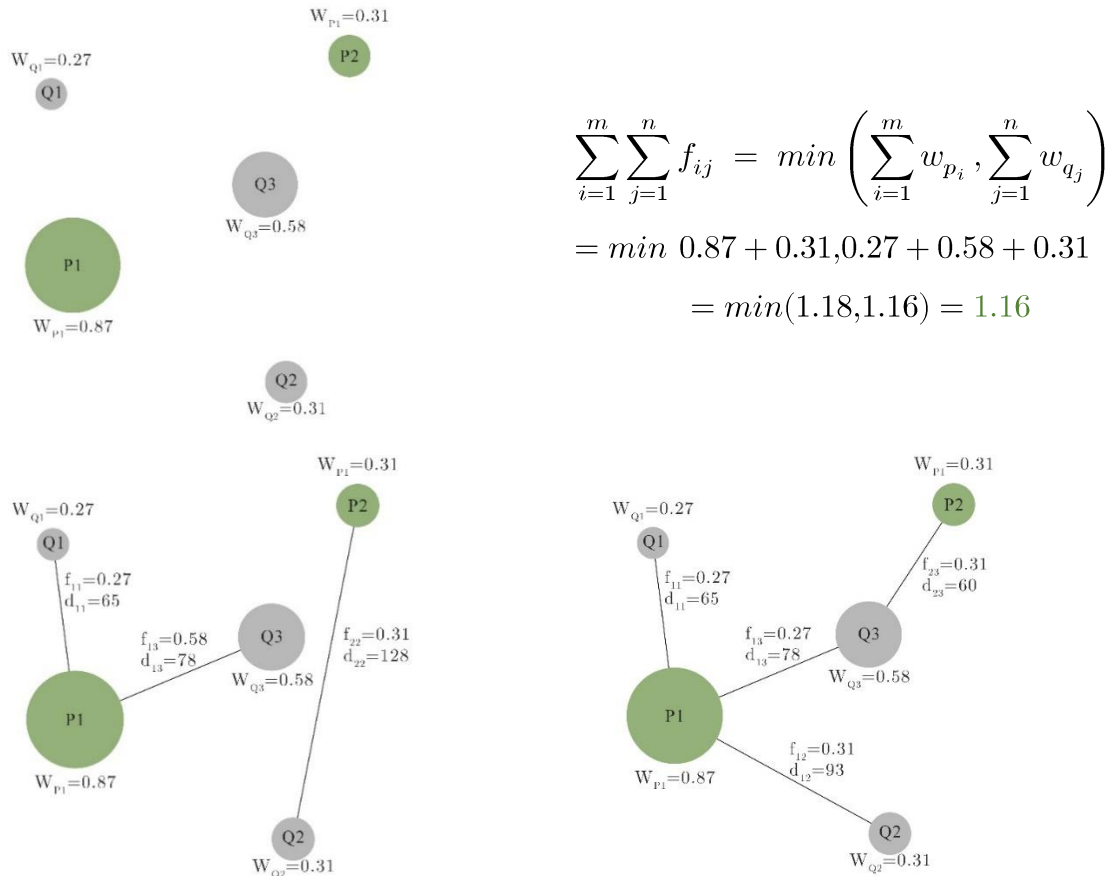
$$EMD(P, Q) = \frac{WORK(P, Q, F)}{\sum_{i=1}^m \sum_{j=1}^n f_{ij}}$$

Εκτός από τους Περιορισμούς υπάρχουν και 2 ιδιότητες που έχει το συγκεκριμένο μέτρο ομοιότητας. Συγκεκριμένα:

1. **Ιδιότητα 1.** Δεν χρειάζεται οι συνολικές μάζες των κατανομών P και Q , να είναι ίσες. Μπορούμε να έχουμε είτε Κατανομές Ίσου Βάρους (*Equal-Weight Distributions*), είτε Ανισομερείς Κατανομές Βάρους (*Unequal-Weight Distributions*).

2. **Ιδιότητα 2.** Δεν είναι απαραίτητη η αντιστοιχία μεταξύ του πλήθους σημείων των κατανομών. Δηλαδή δεν χρειάζεται να ισχύει $m = n$.

Για την καλύτερη κατανόηση του μέτρου αυτού ακολουθεί ένα απλό παράδειγμα με Ανισομερείς Κατανομές Βάρους.



$$\begin{aligned} WORK_1 &= \\ &= 0.27 * 65 + 0.58 * 78 + 0.31 * 128 \\ &= 102.47 \end{aligned}$$

Μη βέλτιστη ροή

$$EMD_{(1)} = \frac{102.47}{1.16} = 88.34$$

$$\begin{aligned} WORK_2 &= \\ &= 0.27 * 65 + 0.27 * 78 + 0.31 * 93 + 0.31 \\ &\quad * 60 = 86.04 \end{aligned}$$

Βέλτιστη ροή

$$EMD_{(2)} = \frac{86.04}{1.16} = 74.17$$

Εικόνα 4.3 Παράδειγμα υπολογισμού EMD

4.4.2 Απόσταση Hausdorff

Η **απόσταση Hausdorff** είναι ένα μέτρο ανομοιότητας για δύο σύνολα σημείων σε έναν μετρικό χώρο. Ορίζεται ως η μέγιστη απόσταση από οποιοδήποτε σημείο σε ένα από τα

δύο σύνολα, στο κοντινότερο σημείο στο άλλο σύνολο. Σύμφωνα με τον ορισμό η αμφίδρομη απόσταση Hausdorff [HKR93] ανάμεσα σε δύο σύνολα σημείων $A = \{\alpha_1, \alpha_2, \dots, \alpha_n\}$, και $B = \{b_1, b_2, \dots, b_n\} \in \mathbb{E}^2$, ισούται με:

$$\delta_H(A, B) = \max(\delta_H(A, B), \delta_H(B, A)),$$

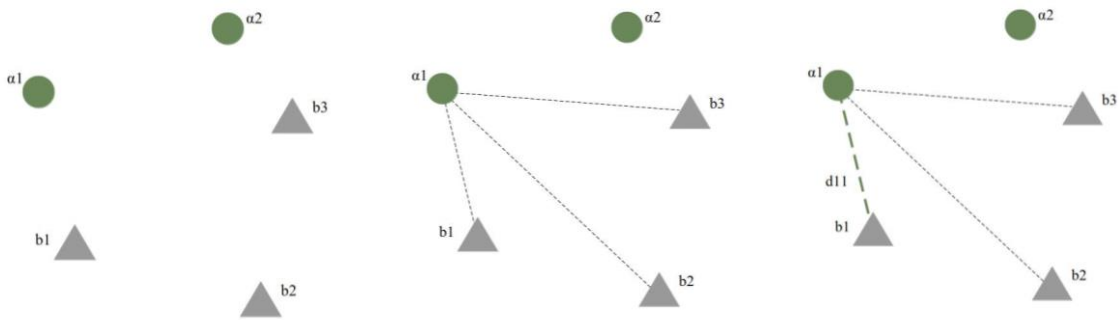
όπου η μονόδρομη απόσταση του A από το B είναι:

$$\delta_H(A, B) = \max_{\alpha \in A} \min_{b \in B} \|\alpha - b\|$$

και η μονόδρομη απόσταση του B από το A είναι:

$$\delta_H(B, A) = \max_{b \in B} \min_{\alpha \in A} \|b - \alpha\|$$

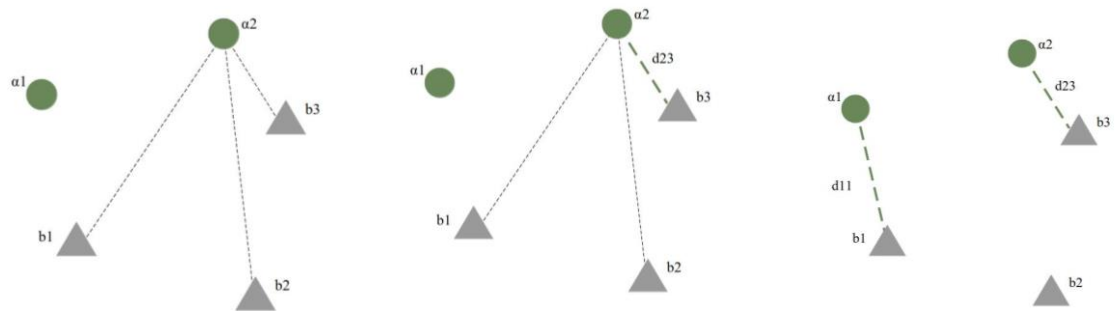
Παρακάτω βλέπουμε ένα απλό παράδειγμα υπολογισμού της απόστασης Hausdorff ανάμεσα σε δύο σύνολα σημείων:



Έστω ότι A και B, και θέλουμε να βρούμε $\delta_H(A, B)$

Βρίσκουμε τις αποστάσεις του α_1 από όλα τα σημεία του B

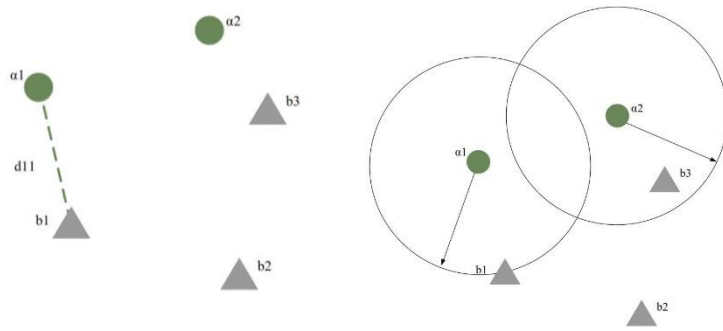
Κρατάμε την ελάχιστη



Βρίσκουμε τις αποστάσεις του α_2 από όλα τα σημεία του B

Κρατάμε την ελάχιστη

Βρίσκουμε την μέγιστη από τις δύο αποστάσεις



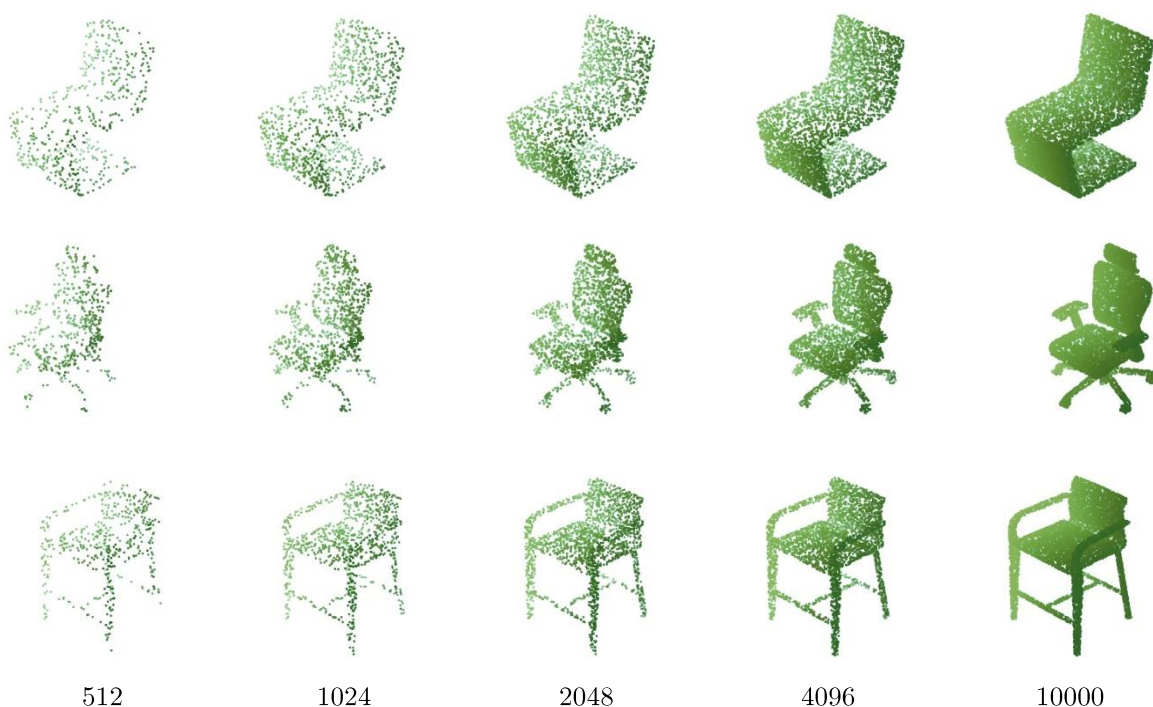
Και αυτή είναι η $\delta_H(A, B)$

Εικόνα 4.4 Παράδειγμα υπολογισμού Hausdorff

5. Πειράματα και Αξιολόγηση

Στο κεφάλαιο αυτό θα δούμε τα διάφορα πειράματα για να βρεθεί το βέλτιστο πιθανό μοντέλο για τις συνθήκες που υπήρχαν. Επιλέχθηκε μία από τις 5 κλάσεις εκπαίδευσης, συγκεκριμένα η κλάση “καρέκλα”, για την παρουσίαση των αποτελεσμάτων της εκπαίδευσης. Η κλάση αυτή αποτελείται από 5350 τρισδιάστατα μοντέλα εκπαίδευσης, 100 μοντέλα επαλήθευσης και 400 μοντέλα ελέγχου.

Μία από τις βασικές παραμέτρους εισόδου τόσο του μοντέλου VAE όσο και του μοντέλου GAN είναι ο αριθμός σημείων νέφους για κάθε τρισδιάστατο σχήμα. Όπως βλέπουμε και στην [Εικόνα 5.1](#) όσο περισσότερα σημεία έχει ένα νέφος σημείων τόσο πιο εύκολα διακρίνουμε την γεωμετρία του τρισδιάστατου αντικειμένου. Αυτό αυτόματα σημαίνει πως για περισσότερα σημεία, πιθανόν θα έχουμε καλύτερα αποτελέσματα. Όπως μπορούμε να διακρίνουμε από την εικόνα, τα 512 σημεία δεν είναι αρκετά για να καταλάβουμε πλήρως την γεωμετρία ενός τρισδιάστατου αντικειμένου, ενώ τα 1024 φαίνεται να είναι αρκετά μόνο σε πιο εύκολα σχήματα. Έτσι πάρθηκε η απόφαση χρήσης περισσότερων από 2048 σημείων.



Εικόνα 5.1 Τρισδιάστατα Μοντέλα ως Νέφη n αριθμού Σημείων

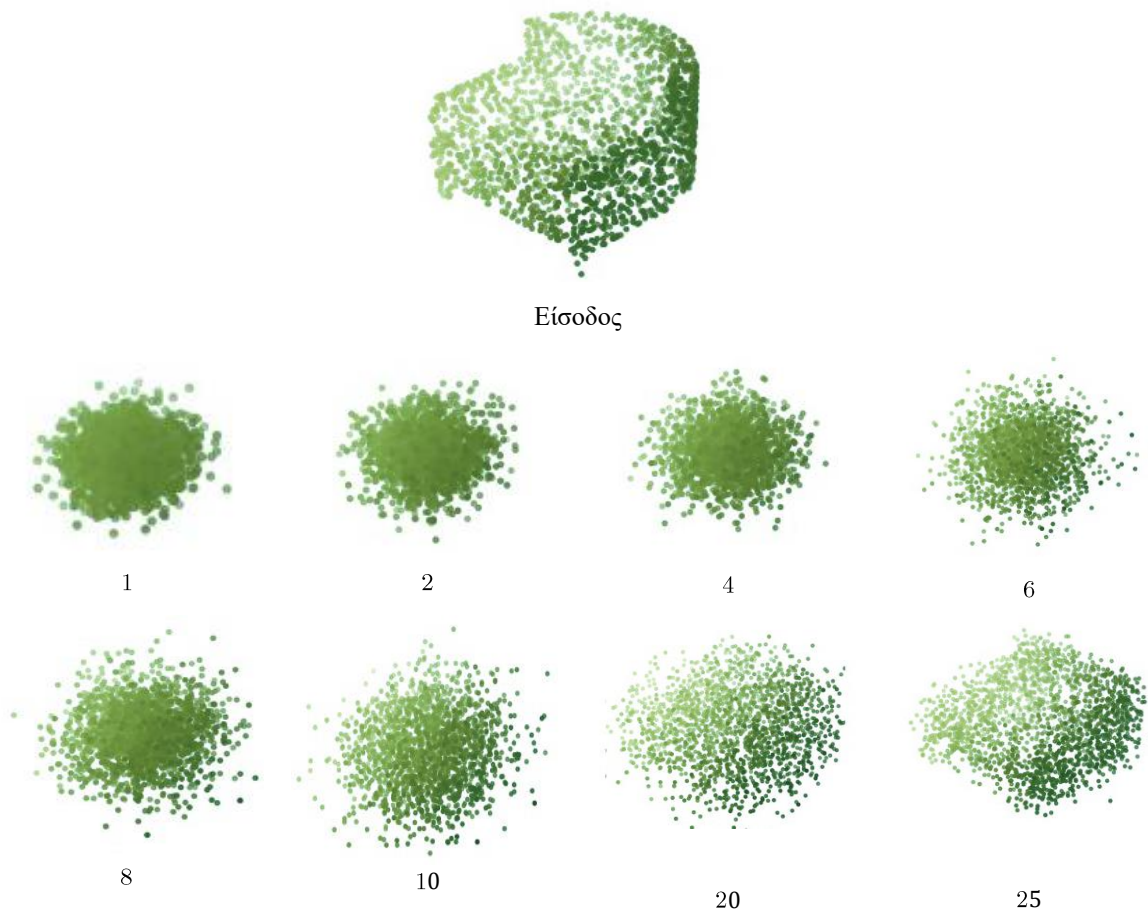
Για την εκπαίδευση των δύο μοντέλων έγινε χρήση της κάρτας γραφικών Nvidia GTX 1660 Super, με ενσωματωμένη μνήμη 6GB, καθώς και πυρήνες CUDA. Η κάρτα αυτή επέτρεψε την εκπαίδευση των μοντέλων για 2048 σημεία, μιας και για περισσότερα τα 6GB μνήμης που προσφέρει δεν ήταν αρκετά. Επιπλέον ο μέγεθος παρτίδας (*batch size*) για κάθε

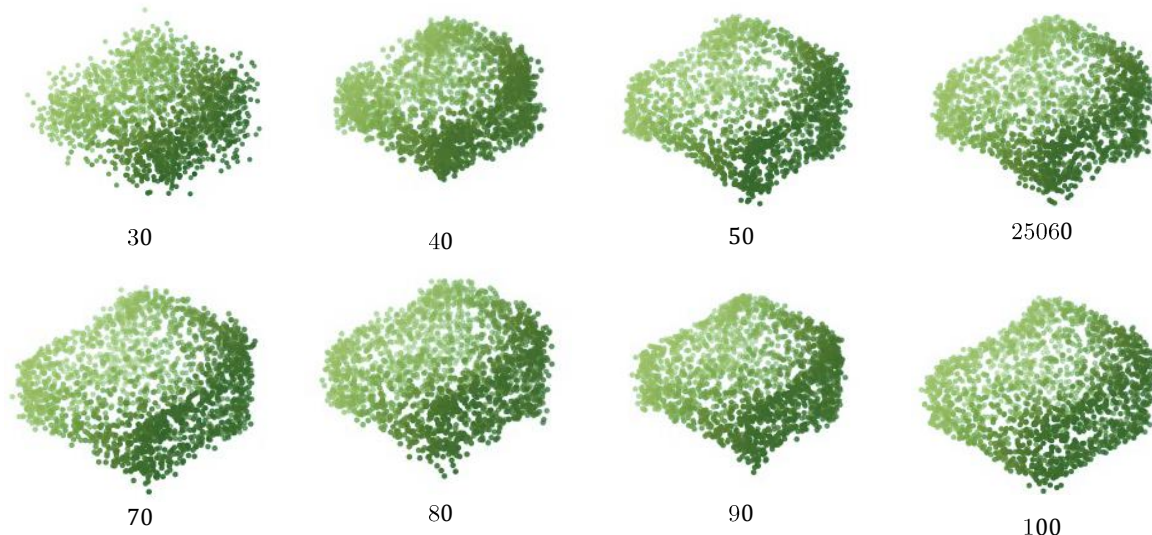
επανάληψη της εκπαίδευσης δεν μπορούσε να ξεπεράσει τα 100 τρισδιάστατα μοντέλα για τον ίδιο λόγο. Οι δύο αυτοί περιορισμοί λόγω του υλικού, σαφώς και είχε αρνητικές επιπτώσεις στα τελικά αποτελέσματα, όπως θα δούμε στη συνέχεια.

5.1 Αποτελέσματα μοντέλου VAE

Μιας και ο αριθμός σημείων έπρεπε να μείνει σταθερός στα 2048 και η παρτίδα δεν μπορούσε να ξεπεράσει τα 100 τρισδιάστατα μοντέλα, δεν υπήρχαν πολλά διαφορετικά πειράματα που μπορούσαν να γίνουν για τον Μεταβλητό Αυτοκωδικοποιητή. Συγκεκριμένα εκτός από την εκπαίδευση σε διαφορετικό αριθμό εποχών, έγιναν πειράματα σε διαφορετικούς ρυθμούς εκμάθησης (*learning rate*), καθώς και διαφορετική κλίση στην συνάρτηση ενεργοποίησης.

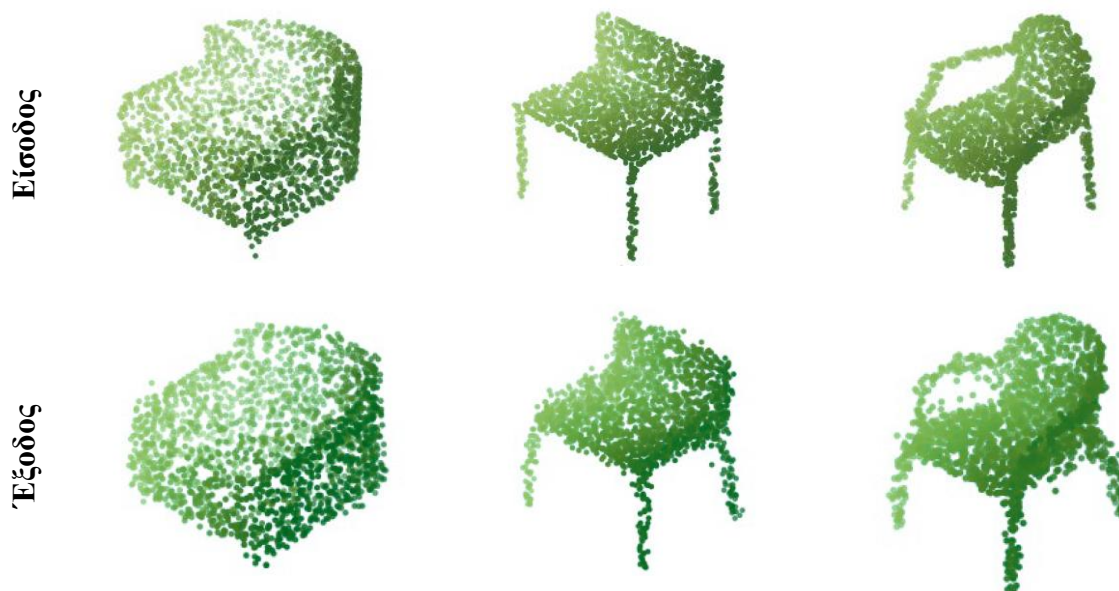
Αρχικά το μοντέλο εκπαιδεύτηκε για 100 εποχές, με ρυθμό εκμάθησης 0.005, και κλίση 0.1. Η κάθε εποχή αποτελείται από περίπου 50 επαναλήψεις παρτίδας. Συνολικά δηλαδή για 100 εποχές είχαμε 4850 επαναλήψεις. Στην [Εικόνα 5.2](#) βλέπουμε πως αλλάζει η έξοδος του VAE όσο περνάνε οι επαναλήψεις.





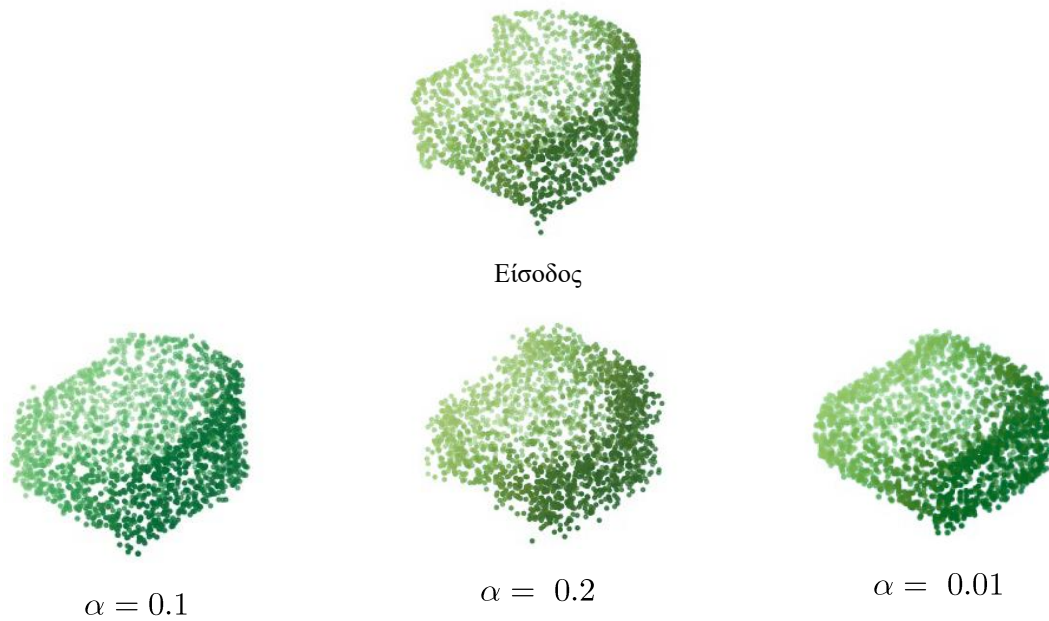
Εικόνα 5.2 Έξοδος Μοντέλου VAE ανά εποχή εκπαίδευσης

Όπως παρατηρούμε η έξοδος σταματάει να αλλάζει μετά από κάποια επανάληψη αλλά δεν φτάνει να έχει ποτέ την ίδια γεωμετρία με την είσοδο, το οποίο οφείλεται είτε στο γεγονός πως τα 2048 σημεία είναι λίγα, είτε ότι πρέπει να εκπαιδύσουμε το μοντέλο με μικρότερο ρυθμό εκπαίδευσης και περισσότερες εποχές. Επίσης μπορεί να πρέπει να γίνει καλύτερη βελτιστοποίηση παραμέτρων, τόσο των στρωμάτων, όσο και της εκπαίδευσης.



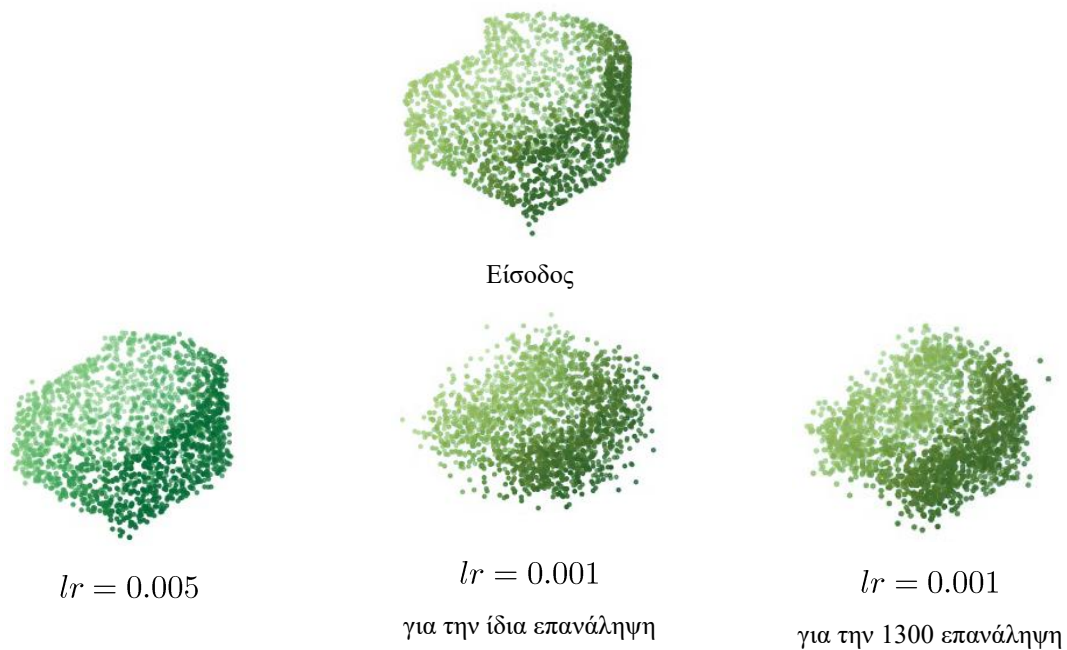
Εικόνα 5.3 Είσοδος και έξοδος μοντέλου VAE

Ένα από τα πειράματα που έγινε ήταν η χρήση διαφορετικής κλίσης στην Συνάρτηση Ενεργοποίησης LeakyReLU, για όλες τις υπόλοιπες παραμέτρους ίδιες σε κάθε μοντέλο. Τα αποτελέσματα για τις τιμές: 0.1, 0.2 και 0.01, φαίνονται στην [Εικόνα 5.4](#).



Εικόνα 5.4 Έξοδοι για διαφορετικές κλίσης της LeakyReLU

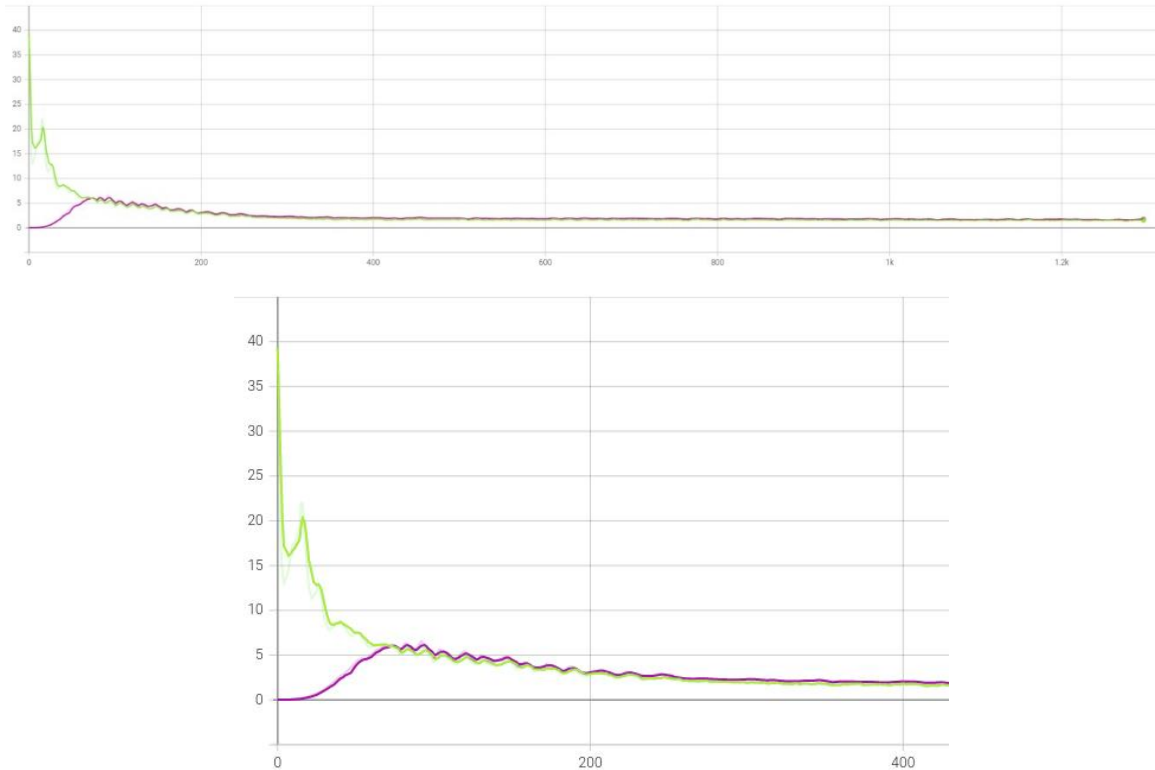
Και στις δύο περιπτώσεις δεν έχουμε μεγάλη διαφορά στο αποτέλεσμα, βέβαια για 0.2 παρατηρούμε πως έχουμε περισσότερο θόρυβο από ότι για την τιμή 0.1.



Εικόνα 5.5 Έξοδοι για διαφορετικούς ρυθμούς εκπαίδευσης

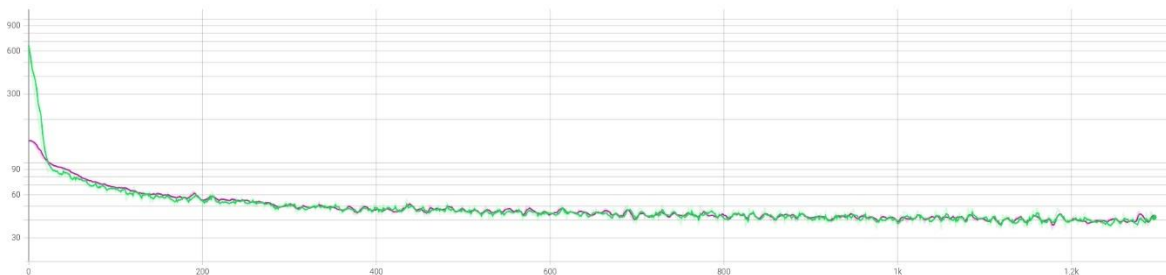
Όπως φαίνεται μειώνοντας τον ρυθμό με τον οποίο εκπαιδεύεται το μοντέλο σε πιο αργό, για τις ίδιες επαναλήψεις έχουμε πολύ διαφορετικά αποτελέσματα. Για 480 επαναλήψεις με ρυθμό 0.005 έχουμε αρκετά καλά αποτελέσματα, ενώ για ρυθμό 0.001 δεν μπορούμε να διακρίνουμε πολλές ομοιότητες με το αρχικό τρισδιάστατο μοντέλο.

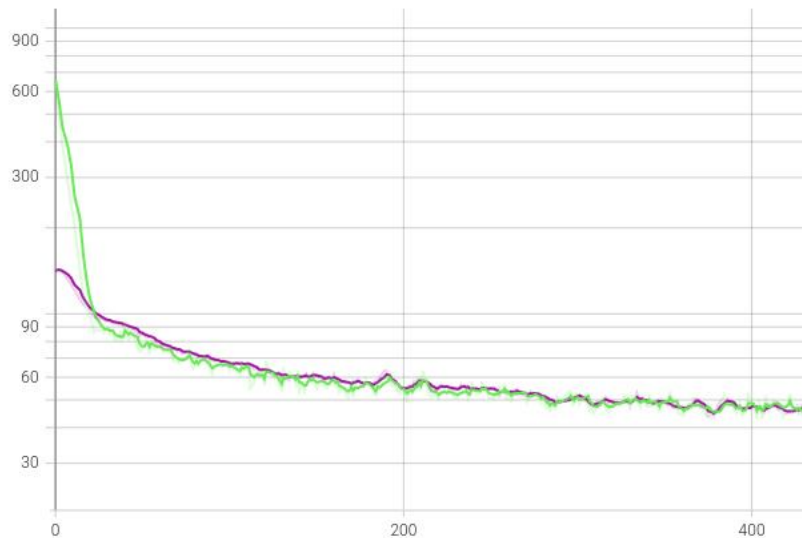
Στην [Εικόνα 5.6](#) και την [Εικόνα 5.7](#) βλέπουμε την συμπεριφορά δύο Σφαλμάτων 1. Σφάλμα Kullback–Leibler (*KL Divergence Loss*), και 2. Σφάλμα Απόστασης Ελάχιστου έργου (*EMD Loss*).



Εικόνα 5.6 Γραφική Παράσταση Σφάλματος KL σε σχέση με τις επαναλήψεις

Όσο μικρότερη είναι η τιμή της απόκλισης KL, τόσο καλύτερα έχουμε ταιριάξει την Είσοδο με την Έξοδο. Με άλλα λόγια τόσο καλύτερα αποτελέσματα θα πάρουμε από τον Decoder. Για δύο σύνολα σημείων έχουμε απόκλιση 0 όταν τα δύο σύνολα είναι ίδια, και τιμή που μπορεί να φτάνει μέχρι και το άπειρο, στις υπόλοιπες περιπτώσεις. Από την γραφική παράσταση βλέπουμε πως η τιμή αυτή μειώνεται σταδιακά κατά την διάρκεια της εκπαίδευσης, και η ελάχιστη τιμή την οποία πήραμε για περισσότερες από 1200 επαναλήψεις ήταν **~1.27**.

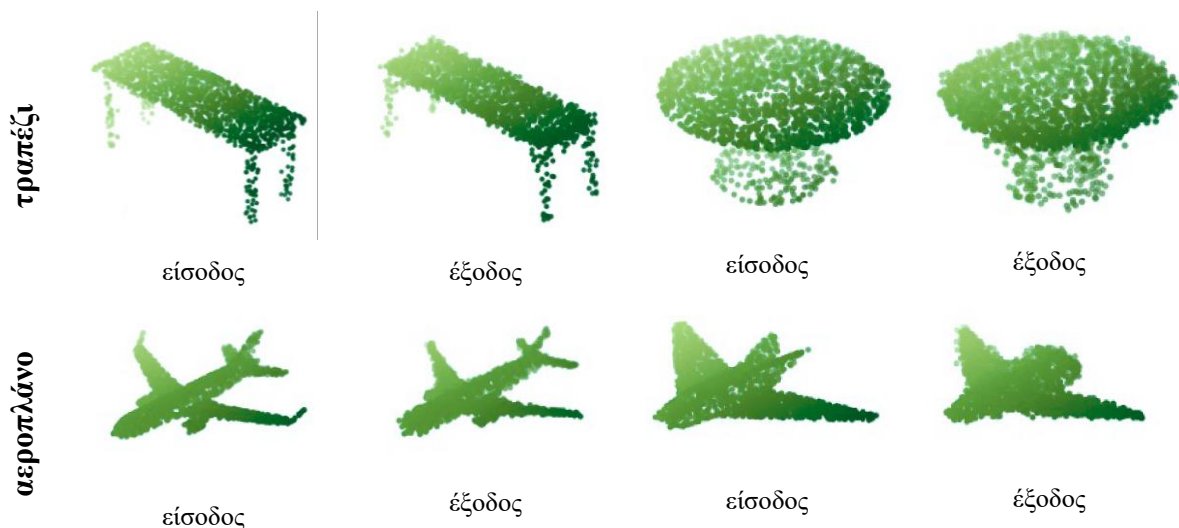




Εικόνα 5.7 Γραφική Παράσταση Σφάλματος EMD σε σχέση με τις επαναλήψεις

Όπως και με την τιμή της απόκλισης KL, όσο μικρότερο είναι το Σφάλμα EMD τόσο πιο κοντινά είναι τα δύο σύνολα σημείων. Από την γραφική παράσταση βλέπουμε πως και αυτή μειώνεται σταδιακά κατά την διάρκεια της εκπαίδευσης, και η ελάχιστη τιμή την οποία πήραμε για περισσότερες από 1200 επαναλήψεις ήταν **~26.3**.

Παρακάτω βλέπουμε κάποια αποτελέσματα του μοντέλου VAE για δύο επιπλέον κλάσεις, 1. τραπέζι, και 2. αεροπλάνο, για τις υπερπαραμέτρους που πήραμε τα καλύτερα αποτελέσματα για την κλάση “καρέκλα”.

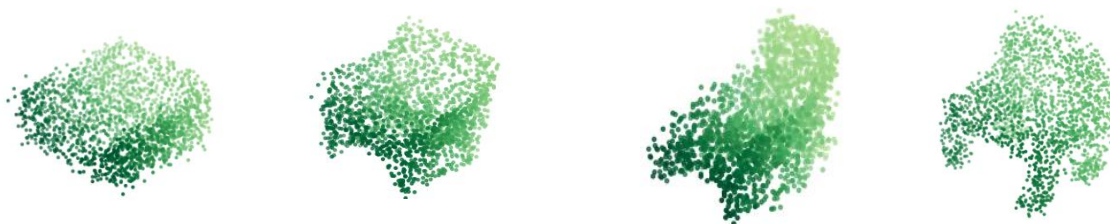


Εικόνα 5.8 Αποτελέσματα VAE σε επιπλέον κλάσεις αντικειμένων

5.2 Αποτελέσματα μοντέλου GAN

Σε σχέση με το μοντέλο VAE, η αρχιτεκτονική των GAN είναι πιο πολύπλοκη με αποτέλεσμα να υπάρχουν ακόμα περισσότεροι περιορισμοί. Συγκεκριμένα για να γίνει εκπαίδευση πάνω σε τρισδιάστατα μοντέλα που αποτελούνται από 2048 σημεία, το μέγεθος παρτίδας έπρεπε να αλλάξει από 100 σε 10 δεδομένα ανά παρτίδα. Αυτό σημαίνει πως από εκεί που είχαμε 54 επαναλήψεις ανά εποχή φτάσαμε περίπου στις 540 επαναλήψεις. Αυτό είχε ως αποτέλεσμα η εκπαίδευση του GAN ανά εποχή, ενώ εξ' αρχής θα ήταν πιο χρονοβόρα από την εκπαίδευση του VAE, να διαρκέσει 3.75 φορές περισσότερο χρόνο. Επιπρόσθετα χρειάστηκαν πολλές παραπάνω εποχές. Συγκεκριμένα το τελικό μοντέλο GAN εκπαιδεύτηκε για 1000 εποχές. Λόγω του ότι η εκπαίδευση ενός τέτοιου δικτύου χωρίς την απαραίτητη υπολογιστική ισχύ ήδη είναι μια μεγάλη πρόκληση, σε σχέση με τα VAE δεν έγιναν τόσα πολλά πειράματα στις διάφορες παραμέτρους, αλλά χρησιμοποιήθηκαν κατά κανόνα οι προκαθορισμένες ή προτεινόμενες τιμές των παραμέτρων.

Μιας και για την εκπαίδευση των GAN έγινε χρήση και της αρχιτεκτονικής των VAE, για την κωδικοποίηση και αποκωδικοποίηση του λανθάνοντα χώρου, τα αποτελέσματα εξαρτήθηκαν κατά πολύ από την απόδοση του μοντέλου VAE. Μετά την εκπαίδευση του Μεταβλητού Αυτοκωδικοποιητή, έγινε χρήση του τελικού μοντέλου για την εκπαίδευση του Γεννητικού Ανταγωνιστικού Δικτύου. Σύντομα διαπιστώθηκε πως το μοντέλο έφτασε σε **Κατάρρευση Λειτουργίας** (mode collapse). Αυτό σημαίνει πως ο Παραγωγός βρήκε κάποιες εξόδους με τις οποίες μπορεί να ξεγελάσει τον Διευκρινιστή, χωρίς αυτές να μοιάζουν απαραίτητα με το Τρισδιάστατο Μοντέλο Εισόδου. Παρακάτω βλέπουμε κάποιες από τις εξόδους του Παραγωγού όταν υπήρχε αυτό το πρόβλημα.



Εικόνα 5.9 Αποτελέσματα με Κατάρρευση Λειτουργίας του GAN

Όπως βλέπουμε, τα αποτελέσματα μοιάζουν με καρέκλες αλλά στην πραγματικότητα ο Παραγωγός έχει σταματήσει να εκπαιδεύεται. Το mode collapse αυτό οφειλόταν σε ένα μικρό λάθος κατά την διάρκεια της φόρτωσης των δεδομένων για την εκπαίδευση των GAN.

Συγκεκριμένα τα δεδομένα εκπαίδευσης που φορτώνονταν είχαν μια απόκλιση 90 μοιρών από τα δεδομένα με τα οποία εκπαιδεύτηκε το δίκτυο VAE (GAN: -45 μοίρες, VAE: 45 μοίρες). Αποτέλεσμα ήταν ο Παραγωγός να δέχεται λάθος κωδικοποιημένα δεδομένα με αποτέλεσμα να μην μπορεί να παράγει σωστές εξόδους. Αυτό προκάλεσε τον Παραγωγό να βρει σιγά σιγά κάποια δεδομένα που έμοιαζαν με καρέκλες αλλά δεν έμοιαζαν με τα αρχικά αντικείμενα, μόνο και μόνο για να ξεγελάσει το Διευκρινιστή. Το πρόβλημα αυτό λύθηκε με το που χρησιμοποιήθηκε η ίδια κλίση των δεδομένων και για τα δύο μοντέλα.

Όταν πλέον λύθηκαν κάποια βασικά προβλήματα που εμφανίστηκαν έγινε επιλογή κάποιων μετρικών αξιολόγησης της εκπαίδευσης. Συγκεκριμένα χρησιμοποιήθηκαν 8 διαφορετικές μετρικές για την ανάλυση και αξιολόγηση του μοντέλου GAN κατά την διάρκεια της εκπαίδευσης, όπως φαίνεται στο [Πίνακα 5.1](#).

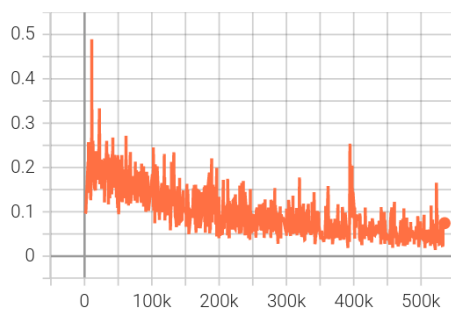
Όνομα	Μετρική	Περιγραφή
D-real-loss	Σφάλμα Διευκρινιστή - Πραγματικά Δεδομένα	<p>Πρόκειται για το σφάλμα του Διευκρινιστή να διακρίνει τα αληθινά δεδομένα.</p> $MSE_{real} = \frac{1}{N} \sum_{i=1}^N (f_i - y_i)^2$ <p>Όπου f_i είναι η ετικέτα που επιστρέφει ο Διευκρινιστής, και y_i η αληθινή ετικέτα, σε αυτή την περίπτωση 1 (real).</p>
D-fake-loss	Σφάλμα Διευκρινιστή - Συνθετικά Δεδομένα	<p>Πρόκειται για το σφάλμα του Διευκρινιστή να διακρίνει τα τεχνητά δεδομένα.</p> $MSE_{fake} = \frac{1}{M} \sum_{i=1}^M (f_i - y_i)^2$ <p>Όπου f_i είναι η ετικέτα που επιστρέφει ο Διευκρινιστής, και y_i η αληθινή ετικέτα, σε αυτή την περίπτωση 0 (fake).</p>
D-GAN	Γενικό Σφάλμα Διευκρινιστή	<p>Πρόκειται για το γενικό σφάλμα του Διευκρινιστή.</p> $MSE = \frac{MSE_{real} + MSE_{fake}}{2}$ <p>Στην προκειμένη περίπτωση η μέση τιμή α-</p>

		νάμεσα στα δύο προηγούμενα σφάλματα. Θα μπορούσε να είναι και το άθροισμα των δύο σφαλμάτων.
G-GAN	Σφάλμα Παραγωγού	<p>Πρόκειται για το σφάλμα του Παραγωγού να ξεγελάσει τον Διευκρινιστή.</p> $MSE_G = \frac{1}{M} \sum_{i=1}^M (f_i - y_i)^2$ <p>Όπου f_i είναι η ετικέτα που επιστρέφει ο Διευκρινιστής, και y_i η ετικέτα που χρειάζεται για να ξεγελάσει ο Παραγωγός τον Διευκρινιστή, δηλαδή 1 (real).</p>
emd_loss	Σφάλμα Απόστασης Ελάχιστου Έργου	Το μέσο Σφάλμα Απόστασης Ελάχιστου Έργου ανάμεσα στο Πραγματικό Πλήρης Τρισδιάστατο Μοντέλο, και την Αποκωδικοποιημένη έξοδο του Παραγωγού.
partial-rec	Σφάλμα Ολοκλήρωσης Σχήματος	Είναι η απόσταση Hausdorff ανάμεσα στα Ελλειπείς Μοντέλα και τα Συνθετικά Μοντέλα.
z_L1	Σφάλμα Λανθάνοντα χώρου z	<p>Όπου y_1 είναι ο λανθάνον χώρος που προκύπτει όταν δώσουμε σαν είσοδο τα συνθετικά δεδομένα στο Κωδικοποιητή του VAE, ενώ το y_2 είναι τυχαίος θόρυβος.</p> $L1 = \sum_{i=1}^N y_1 - y_2 $
EG-loss	Γενικό Σφάλμα Παραγωγού/Κωδικοποιητή	Πρόκειται για το άθροισμα τριών σφαλμάτων: G-GAN, partial-rec και z_L1

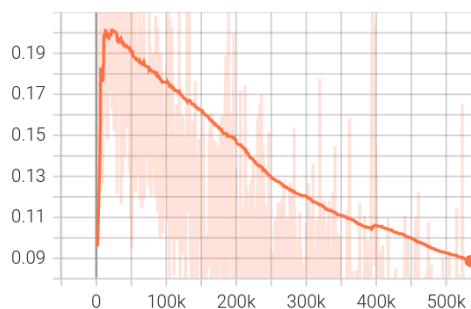
Πίνακας 5.1 Μετρικές Αξιολόγησης Διαδικασίας Εκπαίδευσης του μοντέλου GAN

Στην [Εικόνα 5.10](#) φαίνονται τα διαγράμματα των μετρικών σε σχέση με τις συνολικές επαναλήψεις. Το ένα διάγραμμα αντιπροσωπεύει τις πραγματικές τιμές ανά επανάληψη και το άλλο αντιπροσωπεύει τις τιμές ανά επανάληψη μετά από εξομάλυνση.

D-GAN



Πραγματικό διάγραμμα

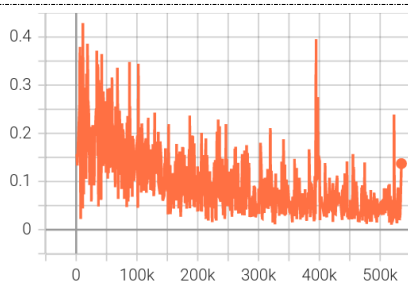


Διάγραμμα μετά από εξομάλυνση

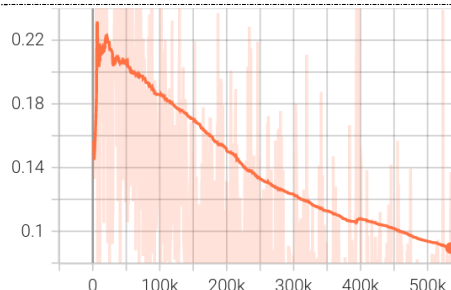
Αρχική τιμή: 0.2 - Βήμα 535χιλ: πραγματικό: 0.07445, smoothed: 0.08071

Το σφάλμα του Διευκρινιστή να διαχωρίζει τα πραγματικά από τα συνθετικά δεδομένα μειώνεται σταδιακά κατά την διάρκεια της εκπαίδευσης, που σημαίνει πως αυξάνεται και η ακρίβεια με την οποία διακρίνει την σωστή κλάση (real ή fake).

D-real-loss



Πραγματικό διάγραμμα

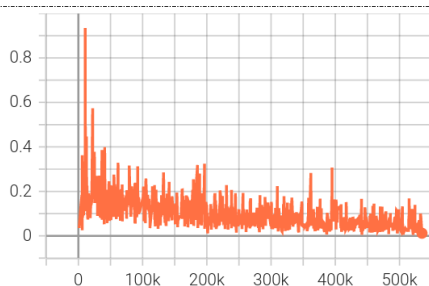


Διάγραμμα μετά από εξομάλυνση

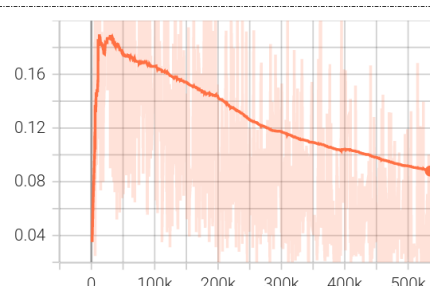
Αρχική τιμή: 0.187 - Βήμα 535χιλ: πραγματικό: 0.01176, smoothed: 0.08796

Το σφάλμα του Διευκρινιστή να διακρίνει σωστά τα πραγματικά δεδομένα μειώνεται σταδιακά, που είναι και το επιθυμητό αποτέλεσμα μιας και θέλουμε ο Διευκρινιστής να κατηγοριοποιεί σωστά τα πραγματικά δεδομένα.

D-fake-loss



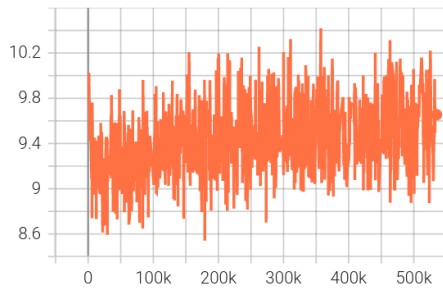
Πραγματικό διάγραμμα



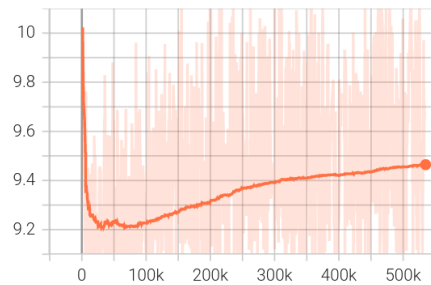
Διάγραμμα μετά από εξομάλυνση

Αρχική τιμή: 0.2249 - Βήμα 535χιλ: πραγματικό: 0.1371, smoothed: 0.08946

Το σφάλμα του Διευκρινιστή να διακρίνει σωστά τα συνθετικά δεδομένα μειώνεται και αυτό σταδιακά. Αυτό σημαίνει πως ο Διευκρινιστής εκπαιδεύεται σωστά, αλλά ο Παραγωγός δεν παράγει δεδομένα τα οποία μπορούν να ξεγελάσουν εύκολα το Διευκρινιστή. Πιθανόν το μοντέλο να μην φτάσει ποτέ σε ισορροπία Nash.

EG-loss

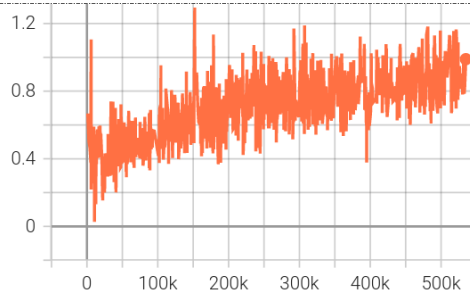
Πραγματικό διάγραμμα



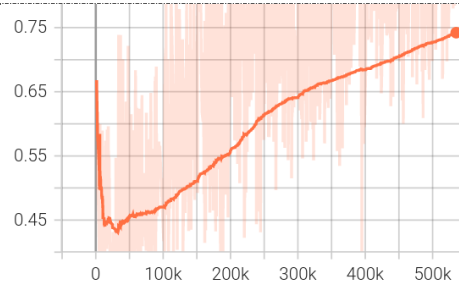
Διάγραμμα μετά από εξομάλυνση

Αρχική τιμή: 10.02 - Βήμα 535χιλ: πραγματικό: 9.654, smoothed: 9.464

Το Σφάλμα αυτό αυξάνεται σταδιακά από το 9.2 στο 9.4. Αυτό συμβαίνει διότι οι 2 από τις 3 τιμές με τις οποίες υπολογίζεται αυξάνονται και αυτές.

G-GAN

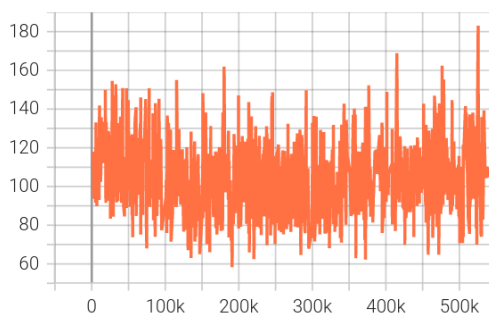
Πραγματικό διάγραμμα



Διάγραμμα μετά από εξομάλυνση

Αρχική τιμή: 0.44 - Βήμα 535χιλ: πραγματικό: 0.9893, smoothed: 0.7421

Το Σφάλμα του Παραγωγού να μπορέσει να παράγει δεδομένα που ξεγελάνε το Διευκρινιστή αυξάνεται με δύο διαφορετικές κλίσεις. Παρατηρούμε πως μετά την επανάληψη 250χιλ. η κλίση της ευθείας μικραίνει.

emd_loss

Πραγματικό διάγραμμα

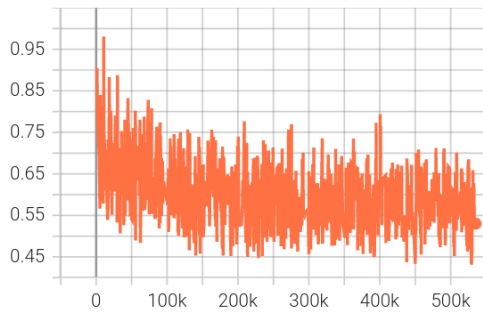


Διάγραμμα μετά από εξομάλυνση

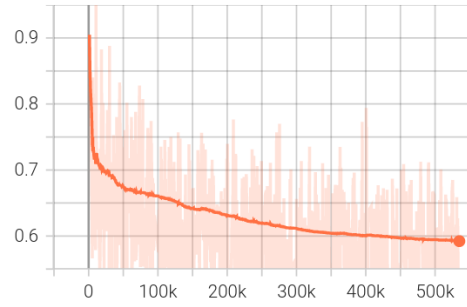
Αρχική τιμή: 114.8 - Βήμα 535χιλ: πραγματικό: 101.2, smoothed: 103.8

Η μέση τιμή της Απόστασης Ελάχιστου έργου ανά επανάληψη μειώνεται μέχρι περίπου την επανάληψη 350χιλ. Στην συνέχεια φαίνεται να αυξάνεται σταδιακά. Αυτό μπορεί να σημαίνει πως ο Παραγωγός προσπαθεί να κάνει αναπροσαρμογή των εξόδων του, πράγμα που μπορεί να προκύψει και από το διάγραμμα G-GAN. Στην [Εικόνα 5.11](#) φαίνεται καλύτερα η καμπύλη.

partial-rec



Πραγματικό διάγραμμα

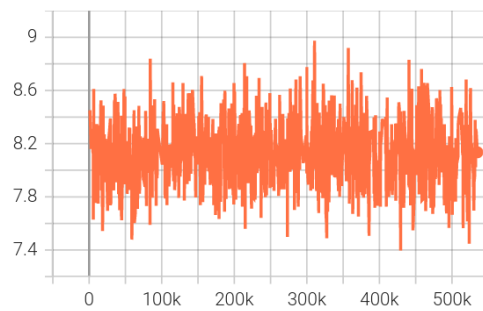


Διάγραμμα μετά από εξομάλυνση

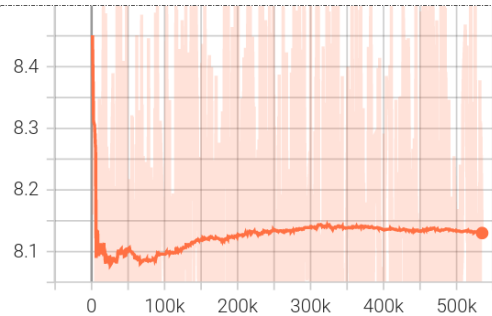
Αρχική τιμή: 0.73 - Βήμα 535χιλ: πραγματικό: 0.5301, smoothed: 0.5922

Το σφάλμα της ολοκλήρωσης σχήματος μειώνεται σταδιακά.

z_L1



Πραγματικό διάγραμμα

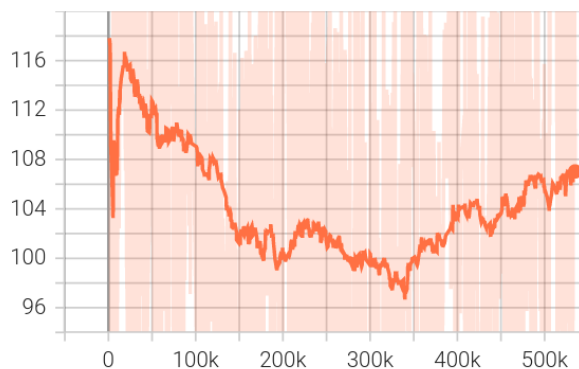


Διάγραμμα μετά από εξομάλυνση

Αρχικά: 8.269 - Βήμα 535χιλ: πραγματικό: 8.13, smoothed: 8.13

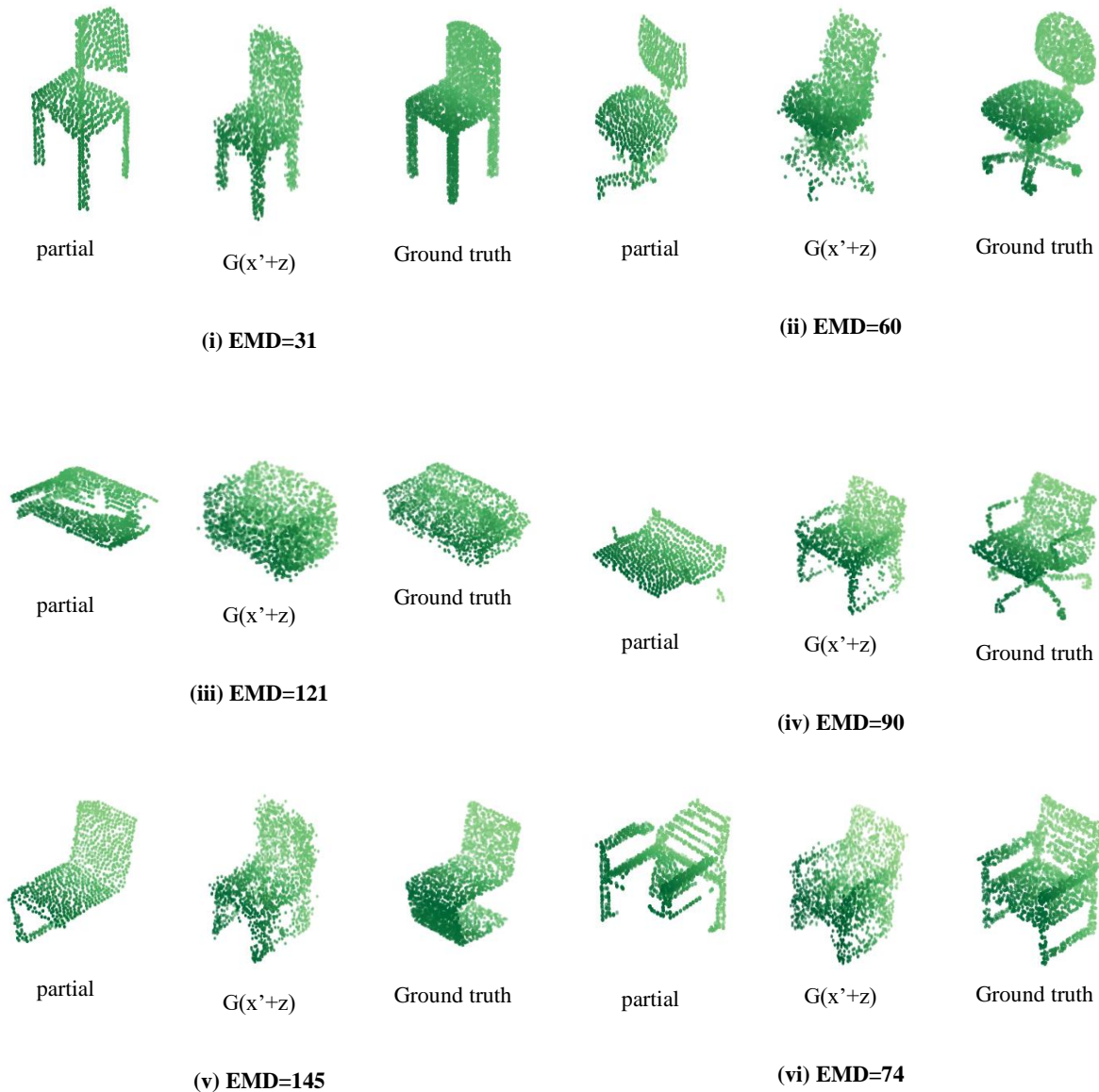
Το Σφάλμα λανθάνοντα χώρου αρχικά μειώνεται, στην συνέχεια για αρκετές επαναλήψεις αυξάνεται με αργούς ρυθμούς, μέχρις ότου αρχίσει να μειώνεται πάλι με αργούς ρυθμούς.

Εικόνα 5.10 Διαγράμματα μετρικών εκπαίδευσης μοντέλου GAN



Εικόνα 5.11 Διάγραμμα Απόστασης Ελάχιστου Έργου

Στην [Εικόνα 5.12](#) βλέπουμε κάποια από τα αποτελέσματα του GAN πάνω στο σύνολο δεδομένων ελέγχου, για την κλάση «καρέκλα».



Εικόνα 5.12 Αποτελέσματα μοντέλου GAN, για την κλάση «καρέκλα»

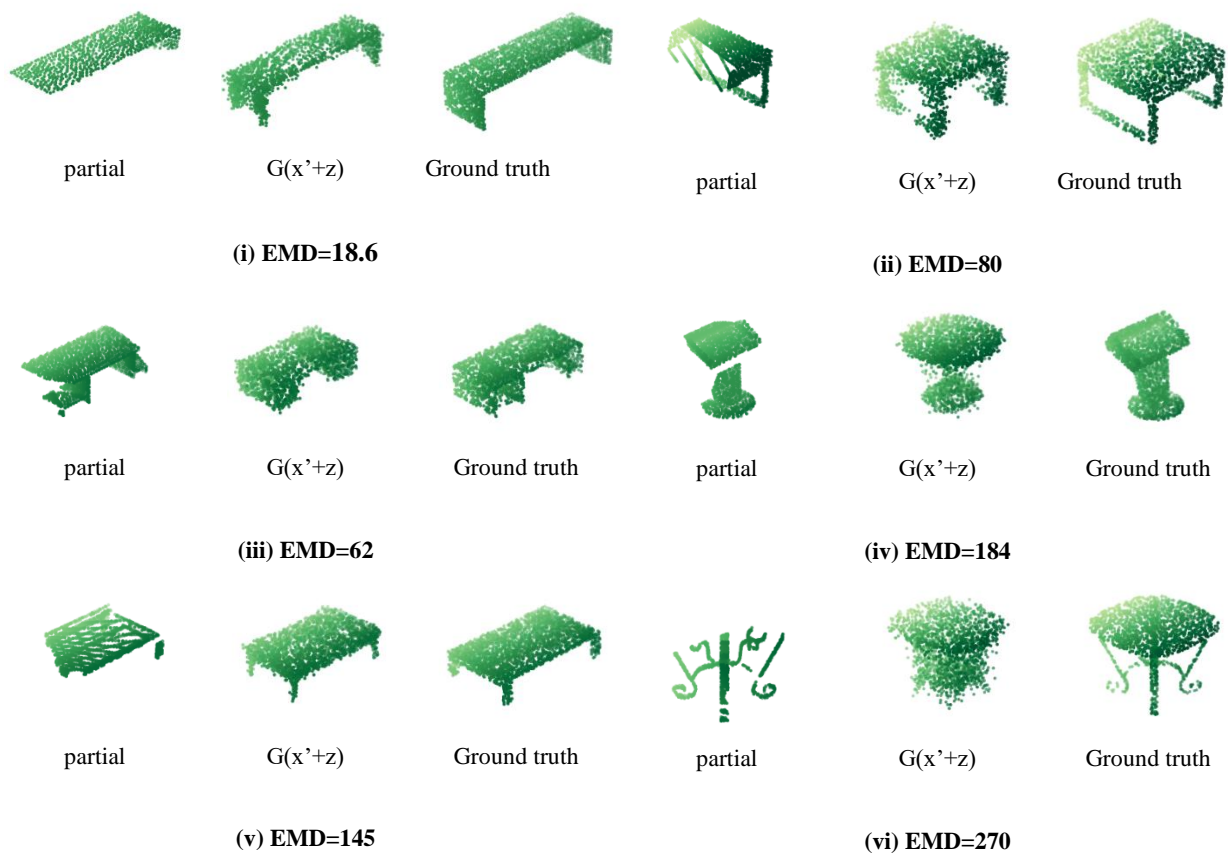
Από την παραπάνω εικόνα αποτελεσμάτων παρατηρούμε τα εξής:

- Όταν δεν λείπουν ολόκληρα βασικά τμήματα καρέκλας, όπως πχ όλα τα πόδια ή ολόκληρη η πλάτη, τα αποτελέσματα μοιάζουν περισσότερο στα πραγματικά μοντέλα (i), (ii), και (vi).
- Όταν λείπουν βασικά τμήματα των αντικειμένων ο Παραγωγός έχει μεγαλύτερη ελευθερία στο να παράγει νέα δεδομένα τα οποία μπορεί να μοιάζουν ή όχι στο πραγματικό αντικείμενο. Το ελλιπές μοντέλο του αντικειμένου (iv) αποτελείται μόνο από το κάθισμα της καρέκλας με αποτέλεσμα ο Παραγωγός να μπορεί να ολοκληρώσει τα

πόδια και την πλάτη της καρέκλας οπωσδήποτε θέλει. Στην συγκεκριμένη περίπτωση ενώ η πλάτη και τα υποβραχιόνια στην έξοδο του GAN και το πραγματικό μοντέλο μοιάζουν, η έξοδος έχει πολύ διαφορετικά πόδια από το πραγματικό μοντέλο.

- Στην περίπτωση που τα αντικείμενα έχουν αρκετές λεπτομέρειες αυτές τις περισσότερες φορές χάνονται (ii) και (vi). Αυτό έχει να κάνει με τις δυνατότητες του Αποκωδικοποιητή. Αν δεν έχει εκπαιδευτεί επαρκώς για κάποιες συγκεκριμένες λεπτομέρειες δεν θα μπορεί να κάνει σωστή κωδικοποίηση και αποκωδικοποίηση.

Στην [Εικόνα 5.13](#) βλέπουμε κάποια από τα αποτελέσματα του GAN πάνω στο σύνολο δεδομένων ελέγχου, για την κλάση «τραπέζι».

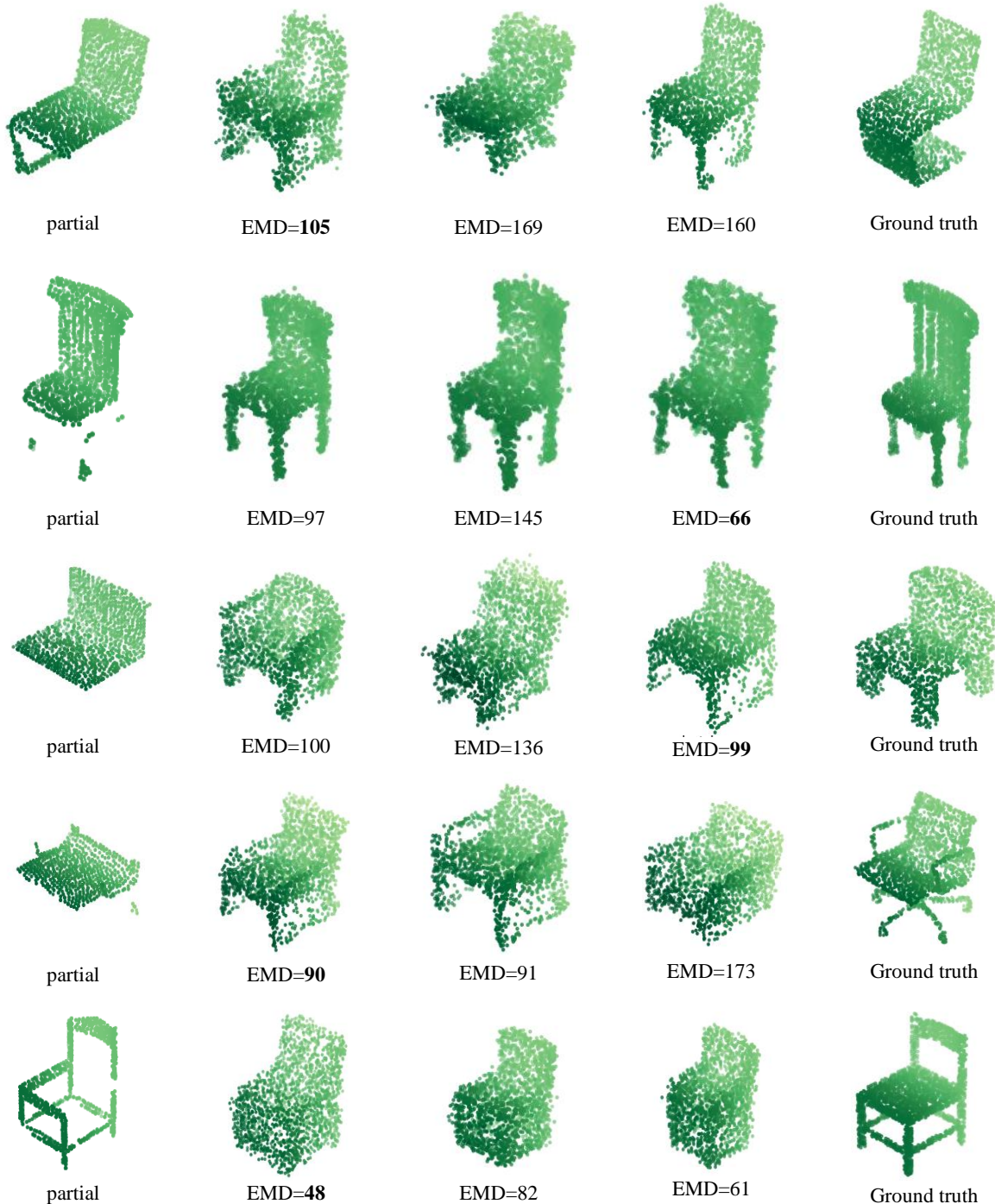


Εικόνα 5.13 Αποτελέσματα μοντέλου GAN, για την κλάση «τραπέζι»

Επιπρόσθετα από τις παρατηρήσεις των αποτελεσμάτων των GAN στα «τραπέζια», παρατηρούμε ότι:

- Στην περίπτωση που έχουμε συμμετρικό αντικείμενο, αν το ελλειπές μοντέλο έχει τμήματα τόσο από την μία όσο από την άλλη πλευρά τότε η έξοδος του GAN έχει μεγαλύτερη πιθανότητα να είναι κοντινή στο πραγματικό αντικείμενο (i), (ii), (iii), και (v).

Η χρήση VAE για την Κωδικοποίηση και Αποκωδικοποίηση του λανθάνοντα χώρου επιτρέπει στον Παραγωγό να παράγει μεγαλύτερη ποικιλία αποτελεσμάτων. Στην [Εικόνα 5.14](#) βλέπουμε 3 αποτελέσματα ανά ένα partial μοντέλο καθώς και το πραγματικό μοντέλο.



Εικόνα 5.14 Ποικιλομορφία αποτελεσμάτων GAN

Ανάλογα τα κομμάτια που λείπουν έχουν και ανάλογη ποικιλία στα αποτελέσματα. Στις περιπτώσεις που λείπουν μεγάλα τμήματα από τα πόδια της καρέκλας έχουμε μεγάλες

διαφορές ανάμεσα στις 3 διαφορετικές εξόδους. Επιπλέον παρατηρούμε πως το μοντέλο έχει την τάση να προσθέτει συγκεκριμένο τύπο ποδιών στις περισσότερες καρέκλες.

Συνήθως στα ελλιπή μοντέλα CAD τα κομμάτια του αντικείμενου που λείπουν είναι τυχαία. Γι' αυτό το λόγο όταν δημιουργήθηκε το σύνολο δεδομένων με τα ελλιπή τρισδιάστατα αντικείμενα αφαιρέθηκαν τυχαία σημεία πάνω στο αντικείμενο, και όχι τμήματα π.χ. πόδι καρέκλας, φτερό αεροπλάνου κτλ. Ενώ η προσέγγιση αυτή είναι πιο κοντινή στις αληθινές συνθήκες, αυτό έχει μια επίπτωση στην ποικιλομορφία των αποτελεσμάτων. Μιας και ο Αυτοκωδικοποιητής που χρησιμοποιήθηκε μπορεί από μόνος του να παράγει νέα δεδομένα, οι έξοδοι του Αποκωδικοποιητή θα έπρεπε να έχουν αρκετές διαφορές μεταξύ τους.

Μια άλλη προσέγγιση δημιουργίας συνόλου δεδομένων με ελλιπή τρισδιάστατα αντικείμενα θα ήταν η χρήση του υποσυνόλου PartNet του ShapeNet. Συγκεκριμένα το PartNet χωρίζει τα αντικείμενα σε τμήματα. Για παράδειγμα μια καρέκλα μπορεί να αποτελείται από πόδια, το κάθισμα, υποβραχιόνια, και την πλάτη. Το σύνολο θα μπορούσε να αποτελείται από μοντέλα στα οποία λείπει ένα από αυτά τα τμήματα ή και περισσότερα. Έτσι κάθε φορά ο Παραγωγός θα πρέπει να προσθέσει συγκεκριμένα τμήματα που λείπουν με αποτέλεσμα να υπάρχει μεγαλύτερη ποικιλομορφία αποτελεσμάτων.

6. Σύνοψη

6.1 Συμπεράσματα

Η χρήση σημείων νέφους των τρισδιάστατων αντικειμένων εκπαίδευσης με μόλις 2048 σημεία είχε κάποιες αρνητικές επιπτώσεις στην ποιότητα των αποτελεσμάτων. Αρχικά, στα πειράματα που έγιναν στο μοντέλο VAE με μικρές αλλαγές στις διάφορες παραμέτρους, παρατηρήθηκε πως μετά από έναν αριθμό εποχών, ανεξάρτητα από τις υπόλοιπες παραμέτρους, η εκπαίδευση έμενε στάσιμη. Αυτό είχε ως αποτέλεσμα να μην μπορούμε να πάρουμε τη βέλτιστη έξοδο από τον Αποκωδικοποιητή του μοντέλου, χωρίς αυτό να σημαίνει απαραίτητα πως τα αποτελέσματα που πήραμε δεν είναι πολύ καλά. Επιπλέον για κάποια πιο περίπλοκα αντικείμενα με περισσότερες λεπτομέρειες, κάποιες χάνονται στην αποκωδικοποίηση.

Οι Μεταβλητοί Αυτοκωδικοποιητές, όπως λέει και το όνομα τους, προσθέτουν μεταβλητότητα στα αποτελέσματά τους, δηλαδή μπορούν να δημιουργήσουν καινούργια δεδομένα που μοιάζουν με το αρχικό σύνολο δεδομένων. Αυτό σίγουρα δίνει μια μεγαλύτερη ποικιλομορφία στα αποτελέσματα, αλλά ταυτόχρονα έχει και κάποια μειονεκτήματα. Γι' αυτό το λόγο η χρήση κάποιου άλλου Αυτοκωδικοποιητή μπορεί να μας έδινε καλύτερες εξόδους. Μια λύση, εκτός από την χρήση περισσότερων σημείων, θα ήταν η χρήση άλλου Αυτοκωδικοποιητή. Ένα από τα μοντέλα που θα μπορούσαν να χρησιμοποιηθούν στη θέση του VAE είναι ο Απλός Αυτοκωδικοποιητής, ο οποίος μαθαίνει τη λανθάνουσα δομή των διάφορων χαρακτηριστικών του συνόλου δεδομένων και έτσι δίνει πιο σταθερά αποτελέσματα. Ένα δεύτερο μοντέλο που θα μπορούσε να χρησιμοποιηθεί είναι μια παραλλαγή του VAE, η λεγόμενη β -VAE. Πρόκειται για έναν Μεταβλητό Αυτοκωδικοποιητή ο οποίος παίρνει σαν είσοδο και μια ρυθμιζόμενη μεταβλητή β , που εξισορροπεί την λανθάνουσα χωρητικότητα του καναλιού και τους περιορισμούς ανεξαρτησίας με την ακρίβεια ανακατασκευής.

Όσον αφορά το Γεννητικό Ανταγωνιστικό Δίκτυο, η ποιότητα των αποτελεσμάτων εξαρτάται από πολλούς παράγοντες. Ο πιο βασικός από αυτούς είναι τα αποτελέσματα του VAE, μιας και τα κωδικοποιημένα δεδομένα χρησιμοποιούνται τόσο στην εκπαίδευση του Παραγωγού όσο στην εκπαίδευση του Διευκρινιστή. Όσο καλύτερα αποτελέσματα θα είχαμε στο VAE, τόσο καλύτερα αποτελέσματα θα είχαμε στο GAN. Στην προκειμένη περίπτωση επειδή τα αποτελέσματα του VAE δεν είναι βέλτιστα, αμέσως μειώνεται και η απόδοση του GAN. Ο Διευκρινιστής και ο Παραγωγός δεν φτάνουν ποτέ στην ισορροπία Nash, τουλάχιστον για τις εποχές που εκπαιδεύτηκε το μοντέλο, και έτσι ο Παραγωγός δεν παράγει δεδομένα που μπορούν να μπερδέψουν το Διευκρινιστή.

Από την άλλη όμως, τα αποτελέσματα που παράγει είναι αρκετά καλά για το πρόβλημα της ολοκλήρωσης σχήματος. Για τις διάφορες εξόδους του μοντέλου για το σύνολο ελέγχου, παρατηρήθηκε ότι για πιο απλά σχήματα χωρίς πολλές λεπτομέρειες ο Παραγωγός κάνει εξαιρετική δουλειά στο να ολοκληρώσει σωστά τα αντικείμενα. Στα πιο περίπλοκα αντικείμενα με περισσότερες λεπτομέρειες δεν κάνει πάντα τόσο καλή δουλειά. Αυτό συμβαίνει γιατί ο Αποκωδικοποιητής του VAE έχει εκπαιδευτεί για συγκεκριμένα δεδομένα, οπότε είναι πιο εύκολο να αποκωδικοποιήσει σωστά ένα πιο εύκολο αντικείμενο που πιθανόν να έχει εκπαιδευτεί σε κάτι παρόμοιο από το να αποκωδικοποιήσει ένα αντικείμενο γεμάτο λεπτομέρειες. Λύσεις σε αυτό το πρόβλημα εκτός από το να γίνει προσπάθεια βελτιστοποίησης του Αυτοκωδικοποιητή, θα ήταν η εύρεση καλύτερου υλικού έτσι ώστε να μπορεί να γίνει αύξηση σημείων νέφους και μεγέθους παρτίδας σε μεγαλύτερους αριθμούς, καθώς και η χρήση πιο πολύπλοκων αρχιτεκτονικών του Παραγωγού και του Διευκρινιστή.

Σε γενικές γραμμές όμως ενώ μπορούν να παράγουν αξιοσημείωτα αποτελέσματα, η εκπαίδευση ενός σταθερού μοντέλου GAN είναι πολύ δύσκολη, μιας και από την φύση τους τα μοντέλα αυτά είναι πολύ ασταθής. Η βελτίωση του ενός δικτύου από τα δύο, έχει αρνητικές επιπτώσεις στην απόδοση του δεύτερου δικτύου, μιας και εκπαιδεύονται ταυτόχρονα. Οι αποδόσεις και στα δύο δίκτυα μπορούν ξαφνικά να αλλάξουν ραγδαία, μπορούν ξαφνικά να ισορροπήσουν, ή να καταλήξουμε σε Κατάρρευση Λειτουργίας. Σίγουρη λύση δεν είναι η εκπαίδευση του Δικτύου για περισσότερο χρόνο διότι ενώ υπάρχει περίπτωση να φτάσει ο Παραγωγός και ο Διευκρινιστής σε ισορροπία Nash, υπάρχει και η περίπτωση της υπερεκπαίδευσης του Δικτύου. Σίγουρα για πολύ καλύτερα αποτελέσματα πρέπει να γίνουν πολλαπλά πειράματα με διαφορετικές αρχιτεκτονικές, παραμέτρους, και ίσως και σύνολα δεδομένων.

6.2 Περαιτέρω Έρευνα

Η χρήση των Γεννητικών Μοντέλων, όπως τα GANs, και τα VAEs, δεν είναι πρωτόγνωρη σε εφαρμογές αύξησης δεδομένων (*data augmentation*), ολοκλήρωσης εικόνων και τρισδιάστατων μοντέλων, και γενικά παραγωγής νέων δεδομένων. Έχουν ήδη προταθεί πάρα πολλά επιστημονικά άρθρα με λύσεις του προβλήματος ολοκλήρωσης τρισδιάστατων σχημάτων που κάνουν χρήση Γεννητικών Μοντέλων. Τι θα γίνει όμως αν χρησιμοποιήσουμε μια τεχνολογία η οποία κανονικά χρησιμοποιείται για ακολουθιακά δεδομένα όπως το κείμενο, σε τρισδιάστατα μοντέλα αντικειμένων;

Τα τελευταία χρόνια μια καινούργια γενιά Νευρωνικών Δικτύων, οι λεγόμενοι **Μετασχηματιστές** (*Transformers*), ξεκίνησε να αποκτά φήμη σε εφαρμογές Επεξεργασίας Φυσικής Γλώσσας (*NLP*) [ATTE17]. Πρόκειται για μοντέλα Βαθιάς Μάθησης τα οποία υιοθετούν τον μηχανισμό προσοχής (*attention mechanism*), ζυγίζοντας έτσι διαφορετικά την σημασία των επιμέρους σημείων των δεδομένων εισόδου. Αναπτύχθηκαν για να επιλύσουν προβλήματα όπως η μηχανική μετάφραση με χρήση νευρωνικών δικτύων (*neural machine translation*), ή το πρόβλημα της μεταγωγής ακολουθίας (*sequence transduction*). Με λίγα λόγια χρησιμοποιούνται σε προβλήματα που μετατρέπουν μια ακολουθία σε μία άλλη ακολουθία, όπως είναι η αναγνώριση ομιλίας (*speech recognition*), μετατροπή κειμένου σε λόγο (*text-to-speech transformation*), ή και αυτόματη περίληψη κειμένου (*automatic document summarization*).

Η ερευνητική κοινότητα όμως δεν αρκέστηκε μόνο στην χρήση των Μετασχηματιστών για προβλήματα Επεξεργασίας Φυσικής Γλώσσας. Σιγά σιγά ξεκίνησαν να χρησιμοποιούνται σε προβλήματα που συνδυάζουν και την Όραση Υπολογιστών, όπως η Δημιουργία Λεζάντας Εικόνας (*image captioning*), και πλέον χρησιμοποιούνται και για προβλήματα καθαρά Όρασης Υπολογιστών, όπως η ταξινόμηση εικόνας (*image classification*), τμηματοποίηση (*segmentation*), σημασιολογική τμηματοποίηση (*semantic segmentation*), ανίχνευση αντικειμένων (*object detection*) κτλ. Οι έρευνες έχουν δείξει κιόλας ότι οι Μετασχηματιστές είναι ικανοί να παράγουν κορυφαία αποτελέσματα σε πολλά προβλήματα Όρασης Υπολογιστών. Κάποια πρόσφατα άρθρα χρήσης Μετασχηματιστών σε εικόνες: [CMSUKZ20], [IMTR18], και [GTP20].

Όμως, όπως οι περισσότερες αρχιτεκτονικές μπορούν με κάποιες αλλαγές να πάρουν ως είσοδο άλλο τύπο δεδομένων από αυτόν για τον οποίο προορίζονταν, έτσι διαπιστώνεται ότι μπορεί να γίνει και στην περίπτωση των Μετασχηματιστών. Ένα μοντέλο Μετασχηματιστή που κανονικά εκπαιδεύεται σε κείμενο, μπορεί να εκπαιδευτεί εξίσου σε εικόνα. Το ίδιο

μοντέλο που παράγει τεχνητό κείμενο από υπάρχον κείμενο, μπορεί να παράγει τεχνητές εικόνες από υπάρχουσες εικόνες. Και εφόσον η αλλαγή αυτή μπορεί να γίνει για δισδιάστατες εικόνες, οι Μετασχηματιστές μπορούν να χρησιμοποιηθούν και για τρισδιάστατα μοντέλα αντικειμένων. Αυτό θα έχει ως αποτέλεσμα με κάποιες μικρές αλλαγές στην αρχιτεκτονική να μπορούμε να δώσουμε ως είσοδο ελλιπή μοντέλα αντικειμένων, και ως έξοδο να πάρουμε ολοκληρωμένα μοντέλα αντικειμένων.

Έτσι ως αντικείμενο περαιτέρω έρευνας στο κομμάτι της Ολοκλήρωσης Τρισδιάστατων Μοντέλων Αντικειμένων, θα μπορούσε να είναι η χρήση των Μετασχηματιστών για την επίλυση του προβλήματος αυτού. Μιας και είναι μια αρκετά καινούργια τεχνολογία ειδικά για εφαρμογές Όρασης Υπολογιστών, οι δυνατότητες είναι πάρα πολλές. Έχουν δημιουργηθεί πάρα πολλά μοντέλα με κορυφαία αποτελέσματα σε προβλήματα NLP τα οποία θα μπορούσαν να χρησιμοποιηθούν και για άλλους τύπους δεδομένων, και πιθανότατα να δημιουργήσουν νέες υπερσύγχρονες τεχνολογίες (*state-of-the-art*) με κορυφαία αποτελέσματα.

Βιβλιογραφία

- [WCZC20] Wu, R., Chen, X., Zhuang, Y., & Chen, B. (2020). Multimodal Shape Completion via Conditional Generative Adversarial Networks. *Computer Vision – ECCV2020*, 281–296. https://doi.org/10.1007/978-3-030-58548-8_17
- [BKPAL10] Botsch M., Kobbelt L., Pauly M., Alliez P., Levy B. *Polygon Mesh Processing*. AK Peters / CRC Press, pp.250, 2010, 9781568814261. [inria-00538098](https://doi.org/10.1007/978-1-4419-9661-1_17)
- [CSKG17] Charles, R. Q., Su, H., KAIchun, M., & Guibas, L. J. (2017). PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. 2017 IEEE Conference on Computer Vision and Pattern Recognition (C.VPR), 77–85. <https://doi.org/10.1109/C.VPR.2017.16>
- [KCY93] Kaufman A., Cohen D., & Yagel R., *Volume graphics*, in *Computer*, vol. 26, no. 7, pp. 51–64, July 1993, <https://doi.org/10.1109/MC.1993.274942>.
- [SC18] Sherman, W. R., & CrAIg, A. B. (2018). *Understanding Virtual Reality: Interface, Application, and Design (The Morgan Kaufmann Series in Computer Graphics)* (2nd ed.). Morgan Kaufmann.<https://doi.org/10.1016/C2013-0-18583-2>
- [SE02] Schneider, P., & Eberly, D. H. (2002). *Geometric Tools for Computer Graphics (The Morgan Kaufmann Series in Computer Graphics)* (1st ed.). Morgan Kaufmann. <https://doi.org/10.1016/B978-1-55860-594-7.X5000-0>
- [F01] Farin, G. (2001). *Curves and Surfaces for CAGD: A Practical Guide (The Morgan Kaufmann Series in Computer Graphics)* (5th ed.). Morgan Kaufmann. <https://doi.org/10.1016/B978-1-55860-737-8.X5000-5>
- [FG08] Frey, P., & George, P. L. (2008). *Mesh Generation: Application to Finite Elements* (2nd ed.). Wiley-ISTE. <https://doi.org/10.1002/9780470611166>
- [K87] Kaufman, A. (1987). *Efficient algorithms for 3D scan-conversion of parametric curves, surfaces, and volumes. Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques - SIGGRAPH '87*. <https://doi.org/10.1145/37401.37423>
- [GWHL20] Guo, Y., Wang, H., Hu, Q., Liu, H., Liu, L., & Bennamoun, M. (2020). *Deep Learning for 3D Point Clouds: A Survey*. IEEE transactions on pattern analysis and machine intelligence, PP.
- [JL17] Jaarsveld, S., & Lachmann, T. (2017). Intelligence and Creativity in Problem Solving: The Importance of Test Features in Cognition Research. *Frontiers in Psychology*, 8, 8–134. <https://doi.org/10.3389/fpsyg.2017.00134>
- [Z20] Zhang, X. (2020). *A Matrix Algebra Approach to Artificial Intelligence* (1st ed. 2020 ed.). Springer. <https://doi.org/10.1007/978-981-15-2770-8>
- [C20] Cohen, S. (2020). *Artificial Intelligence and Deep Learning in Pathology*. Elsevier Gezondheidszorg. <https://doi.org/10.1016/C2018-0-02465-2>
- [K03] Kurfess, F. J. (2003). *Artificial Intelligence. Encyclopedia of Physical Science and Technology*, 609–629. <https://doi.org/10.1016/b0-12-227410-5/00027-2>
- [WC03] Walczak, S., & Cerpa, N. (2003). Artificial Neural Networks. *Encyclopedia of Physical Science and Technology*, 631–645. <https://doi.org/10.1016/b0-12-227410-5/00837-1>

- [LMS20] Lawless, W., Mittu, R., & Sofge, D. (2020). *Human-Machine Shared Contexts* (1st ed.). Academic Press. <https://doi.org/10.1016/C2018-0-05386-4>
- [SM01] Smelser, N. J., & Baltes, P. B. (2001). *International Encyclopedia of Social & Behavioral Sciences* (1st ed.). Pergamon.
- [PBM18] Page, J., BAIn, M., & Mukhlis, F. (2018). *The Risks of Low Level Narrow Artificial Intelligence*. 2018 IEEE International Conference on Intelligence and Safety for Robotics (ISR), <https://doi.org/10.1109/iisr.2018.8535903>
- [CMCS20] Cuzzolin, F., Morelli, A., Cîrstea, B., & Sahakian, B. J. (2020). *Knowing me, knowing you: theory of mind in AI*. *Psychological Medicine*, <https://doi.org/10.1017/s0033291720000835>
- [SELF18] Chatila, R., Renaudo, E., Andries, M., Chavez-Garcia, R.-O., Luce-Vayrac, P., Gottstein, R., Alami, R., Clodic, A., Devin, S., Girard, B., & Khamassi, M. (2018). *Toward Self-Aware Robots*. *Frontiers in Robotics and AI*, 5, 1–20. <https://doi.org/10.3389/frobt.2018.00088>
- [G14A] Goertzel, B. (2014a). Artificial General Intelligence: Concept, State of the Art, and Future Prospects. *Journal of Artificial General Intelligence*, 5(1), 1–48. <https://doi.org/10.2478/jagi-2014-0001>
- [W09] Woolf, B. P. (2010). *Building Intelligent Interactive Tutors: Student-centered Strategies for Revolutionizing E-learning* (1st ed.). Morgan Kaufmann.
- [GR18] Gudivada, V. N., & Rao, C. R. (2018). *Computational Analysis and Understanding of Natural Languages: Principles, Methods and Applications (Volume 38) (Handbook of Statistics, Volume 38)* (1st ed.). North Holland.
- [M97] Mitchell, T. M. (1997). *Machine Learning* (1st ed.). McGraw-Hill Education.
- [MCM13] Michalski, R. S., Carbonell, J. G., & Mitchell, T. M. (2013). *Machine Learning: An Artificial Intelligence Approach (Symbolic Computation)* (1983rd ed.). Springer. <https://doi.org/10.1007/978-3-662-12405-5>
- [S15] Sugiyama, M. (2015). *Introduction to Statistical Machine Learning* (1st ed.). Morgan Kaufmann. <https://doi.org/10.1016/C2014-0-01992-2>
- [W14] Wittek, P. (2014). *Quantum Machine Learning: What Quantum Computing Means to Data Mining (Elsevier Insights)* (1st ed.). Academic Press. <https://doi.org/10.1016/C2013-0-19170-2>
- [KK07] Kononenko, I., & Kukar, M. (2007). *Machine Learning and Data Mining*. Elsevier Gezondheidszorg.
- [HYL17] Hamilton, W.L., Ying, R., & Leskovec, J. (2017). *Representation Learning on Graphs: Methods and Applications*.
- [LZLZ16] Li, X., Zhang, G., Li, K., & Zheng, W. (2016). Deep Learning and Its Parallelization. *Big Data*, 95–118. <https://doi.org/10.1016/b978-0-12-805394-2.00004-0>
- [LBH15] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444. <https://doi.org/10.1038/nature14539>
- [A18] Aggarwal, C. C. (2018). *Neural Networks and Deep Learning* [E-book]. Springer Publishing. <https://doi.org/10.1007/978-3-319-94463-0>

- [GBC16] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. Amsterdam University Press. <http://www.deeplearningbook.org>
- [ZYZ20] Zhang, Z., Yang, L., & Zheng, Y. (2020). *Multimodal medical volumes translation and segmentation with generative adversarial network*. Handbook of Medical Image Computing and Computer Assisted Intervention, 183–204. <https://doi.org/10.1016/b978-0-12-816176-0.00013-2>
- [WFHP17] Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2017). Deep learning. *Data Mining*, 417–466. <https://doi.org/10.1016/b978-0-12-804291-5.00010-6>
- [GAN14] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). *Generative adversarial nets*. In Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2 (NIPS'14). MIT Press, Cambridge, MA, USA, 2672–2680.
- [T20] Theodoridis, S. (2020). *Neural Networks and Deep Learning*. *Machine Learning*, 901–1038. <https://doi.org/10.1016/b978-0-12-818803-3.00030-1>
- [CZC18] Cai, H., Zheng, V. W., & Chang, K. C.-C. (2018). A Comprehensive Survey of Graph Embedding: Problems, Techniques, and Applications. *IEEE Transactions on Knowledge and Data Engineering*, 30(9), 1616–1637. <https://doi.org/10.1109/tkde.2018.2807452>
- [GNN18] Zhou, J., Cui, G., Zhang, Z., Y, Yang, C., Liu, Z., Wang, L., Li, C., & Sun, M. (2018). *Graph Neural Networks: A Review of Methods and Applications* <https://arxiv.org/abs/1812.08434>
- [LYZF20] Lan, L., You, L., Zhang, Z., Fan, Z., Zhao, W., Zeng, N., Chen, Y., & Zhou, X. (2020). *Generative Adversarial Networks and Its Applications in Biomedical Informatics*. *Frontiers in Public Health*, 8, <https://doi.org/10.3389/fpubh.2020.00164>
- [KL51] Kullback, S., & Leibler, R. A. (1951). On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1), 79–86. <https://doi.org/10.1214/aoms/1177729694>
- [CGAN14] Mirza, M., & Osindero, S. (2014). Conditional Generative Adversarial Nets, <https://arxiv.org/abs/1411.1784>
- [LB19] Langr, J., & Bok, V. (2019). *GANs in Action: Deep learning with Generative Adversarial Networks* (1st ed.). Manning Publications.
- [WGAN17] Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein Generative Adversarial Networks. *Proceedings of the 34th International Conference on Machine Learning*, in *Proceedings of Machine Learning Research* 70:214–223, <https://arxiv.org/abs/1701.07875>
- [WAN19] Wiatrak, M., Albrecht, S.V., & Nystrom, A. (2019). *Stabilizing Generative Adversarial Networks: A Survey*. <https://arxiv.org/abs/1910.00927>
- [RMC15] Radford, A., Metz, L., & Chintala, S. (2015). *Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks*. <https://arxiv.org/abs/1511.06434>
- [AKK21] Alqahtani, H., Kavakli-Thorne, M. & Kumar, G. (2021). *Applications of Generative Adversarial Networks (GANs): An Updated Review*. *Arch Computat Methods Eng* **28**, 525–552. <https://doi.org/10.1007/s11831-019-09388-y>
- [DKD16] Donahue, J., Krähenbühl, P., & Darrell, T. (2016). *Adversarial Feature Learning*.

- <https://arxiv.org/abs/1605.09782>
- [O16] Odena, A. (2016). *Semi-Supervised Learning with Generative Adversarial Networks*. <https://arxiv.org/abs/1606.01583>
- [INFO16] Chen, X., Duan, Y., Houthoofd, R., Schulman, J., Sutskever, I., & Abbeel, P. (2016). *InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets*. <https://arxiv.org/abs/1606.03657>
- [WS49] Weaver, W., & Shannon, C. E. (1949). *The mathematical theory of communication*. Urbana: University of Illinois Press.
- [OOS16] Odena, A., Olah, C., & Shlens, J. (2016). *Conditional Image Synthesis With Auxiliary Classifier GANs*. <https://arxiv.org/abs/1610.09585>
- [CCM20] Chen, X., Chen, B., & Mitra, J. N. (2020). *Unpaired Point Cloud Completion on Real Scans using Adversarial Training*. <https://arxiv.org/abs/1904.00069>
- [MSN19] Liu, M., Sheng, L., Yang, S., Shao, J., & Hu, S. (2019). *Morphing and Sampling Network for Dense Point Cloud Completion*. <https://arxiv.org/abs/1912.00280>
- [PCN18] Yuan, W., Khot, T., Held, D., Mertz, C., & Hebert, M. (2018). *PCN: Point Completion Network*. <https://arxiv.org/abs/1808.00671>
- [ADMG18] Achlioptas, P., Diamanti, O., Mitliagkas, I., & Guibas, L. (2018). *Learning Representations and Generative Models for 3D Point Clouds*. <https://arxiv.org/abs/1707.02392>
- [SC18] Dai, A., Ritchie, D., Bokeloh, M., Reed, S., Sturm, J., & Nießner, M. (2018). *ScanComplete: Large-Scale Scene Completion and Semantic Segmentation for 3D Scans*. <https://arxiv.org/abs/1712.10215>
- [HLHKY17] Han, X., Li, Z., Huang, H., Kalogerakis, E., & Yu, Y. (2017). *High-Resolution Shape Completion Using Deep Neural Networks for Global Structure and Local Geometry Inference*. <https://arxiv.org/abs/1709.07599>
- [SHAPE15] Chang, A. X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., Xiao, J., Yi, L., & Yu, F. (2015). *ShapeNet: An Information-Rich 3D Model Repository*, <https://arxiv.org/pdf/1512.03012.pdf>
- [EMD00] Rubner, Y., Tomasi, C., & Guibas, L., (2000). *The earth mover's distance as a metric for image retrieval*. IJCV 2000. <https://link.springer.com/content/pdf/10.1023/A:1026543900054.pdf>
- [ATTE17] Polosukhin, I., Kaiser, L., Gomez, A. N., Jones, L., Uszkoreit, J., Parmar, N., Shazeer, N., & Vaswani, A. (2017). *Attention Is All You Need*. <https://arxiv.org/abs/1706.03762>
- [CMSUKZ20] Carion N., Massa F., Synnaeve G., Usunier N., Kirillov A., Zagoruyko S. (2020) *End-to-End Object Detection with Transformers*. In: Vedaldi A., Bischof H., Brox T., Frahm JM. (eds) Computer Vision – ECCV 2020. ECCV 2020. Lecture Notes in Computer Science, vol 12346. Springer, Cham. https://doi.org/10.1007/978-3-030-58452-8_13
- [IMTR18] Parmar, N., Vaswani, A., Uszkoreit, J., Kaiser, L., Shazeer, N., Ku, A., & Tran, D. (2018). *Image Transformer*. <https://arxiv.org/abs/1802.05751>
- [GTP20] Chen, M., Radford, A., & Sutskever, I. (2020). *Generative Pretraining from Pixels*.

- <https://openai.com/blog/image-gpt/>
https://cdn.openai.com/papers/Generative_Pretraining_from_Pixels_V1_ICML.pdf
- [VAE14] Kingma, D. P., & Welling, M. (2014). *Auto-Encoding Variational Bayes*.
<https://arxiv.org/abs/1312.6114>
- [POINT16] Qi, R. C., Su, H., Mo, K., & Guibas, L. J. (2016). *PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation*. <https://arxiv.org/abs/1612.00593>
- [XWCL15] Xu, B., Wang, N., Chen, T., & Li, M. (2015). *Empirical Evaluation of Rectified Activations in Convolutional Network*. <https://arxiv.org/abs/1505.00853>
- [LRELU13] Maas, A. L., Hannun, A. Y., & Ng, A. Y. (2013). *Rectifier Nonlinearities Improve Neural Network Acoustic Models*,
https://ai.stanford.edu/~amaas/papers/relu_hybrid_icml2013_final.pdf
- [HKR93] Huttenlocher, D. P., Klanderman, G. A., & Rucklidge, W. J., (1993). *Comparing images using the Hausdorff distance*, in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 15, no. 9, pp. 850-863, Sept. 1993, doi: [10.1109/34.232073](https://doi.org/10.1109/34.232073).