



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ  
UNIVERSITY OF PIRAEUS

**Καταπολέμηση της ανισορροπίας των  
κλάσεων με τη χρήση των Conditional GANs  
στην ιατρική απεικόνιση για τη διάγνωση  
της Πνευμονίας**

Από

Ηλίας Ζαμπετάκης

Υποβάλλεται

για την εκπλήρωση των προϋποθέσεων λήψης

Μεταπτυχιακού Διπλώματος

στα «Πληροφοριακά Συστήματα & Υπηρεσίες»

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

Φεβρουάριος 2024

## Περίληψη

Η αποτελεσματική διαχείριση των ιατρικών δεδομένων είναι σημαντική για τη διασφάλιση της αξιοπιστίας των συστημάτων ανίχνευσης με τη βοήθεια υπολογιστή (computer-aided detection-CAD) στη διάγνωση και στην επιλογή θεραπείας. Η παρούσα εργασία επικεντρώνεται στην πρόκληση της ανισορροπίας των κλάσεων στο πλαίσιο της διάγνωσης της πνευμονίας σε παιδιατρικό πληθυσμό με τη χρήση εικόνων ακτινογραφίας θώρακος. Η ανισορροπία των κλάσεων μπορεί να αλλοιώσει τις προβλέψεις, μειώνοντας τη διαγνωστική ακρίβεια. Στο πλαίσιο αυτό, χρησιμοποιούνται συνελκτικά νευρωνικά δίκτυα για τη δημιουργία ενός ταξινομητή και αξιοποιούνται καινοτόμες τεχνικές, συγκεκριμένα τα Conditional Generative Adversarial Networks (cGANs) ως τεχνική Oversampling, για την αντιμετώπιση αυτού του ζητήματος.

Η εργασία αποτελείται από τρία βασικά μεθοδολογικά στάδια. Στο πρώτο στάδιο, δημιουργείται, εκπαιδεύεται και αξιολογείται ένας ταξινομητής χρησιμοποιώντας τα προ-εκπαιδευμένα μοντέλα, ResNet-18, DenseNet-121, Mobile\_NetV2, ResNet-34 και VGG-16. Στο δεύτερο στάδιο, εφαρμόζονται τεχνικές βελτιστοποίησης του Classifier. Χρησιμοποιήθηκαν παραδοσιακές μέθοδοι όπως η χρήση του Weighted Random Sampler και πιο μοντέρνες όπως τα cGANs.

Όσον αφορά τα GANs, χρησιμοποιήθηκαν διάφορες αρχιτεκτονικές βελτιστοποίησης, με την αρχιτεκτονική WGAN-GP (Wasserstein GAN-Gradient Penalty) να θεωρείται η πιο κατάλληλη για την επίλυση του προβλήματος. Βασιζόμενοι σε αυτήν την αρχιτεκτονική, ξεκίνησε η ρύθμιση (fine-tuning) των υπερπαραμέτρων του παραγωγικού δικτύου, μέχρι να επιτευχθεί όσο το δυνατό καλύτερο οπτικό αποτέλεσμα στα συνθετικά δείγματα. Αφού εκτελέστηκαν μια σειρά από πειράματα αλλάζοντας τις υπερπαραμέτρους και πολλές δοκιμές, καταλήξαμε σε τρία μοντέλα cGAN: Το πρώτο cGAN ονομάστηκε cGAN (RMSProp), καθώς η κύρια αλλαγή ήταν η χρήση του RMSprop βελτιστοποιητή. Το δεύτερο ονομάστηκε cGAN (64 εποχές και Adam Optimizer) λόγω του αριθμού εποχών και της χρήσης του βελτιστοποιητή Adam και το τρίτο cGAN, cGAN (5 εποχές 128x128), λόγω του μικρού αριθμού εποχών, αλλά και του διαφορετικού μεγέθους εικόνων 128x128.

Στο τρίτο στάδιο, τα συνθετικά δεδομένα, τα οποία παράγονται από τα τρία παραπάνω cGAN, ενσωματώνονται στο σύνολο εκπαίδευσης για την ενίσχυση της κατηγορίας μειονότητας, Normal.

Από τη σύγκριση των μοντέλων που εκπαιδεύτηκαν με την προσθήκη συνθετικών δειγμάτων από τα τρία διαφορετικά cGAN και με βάση τα προεκπαιδευμένα μοντέλα που χρησιμοποιήθηκαν για την εκπαίδευση του ταξινομητή, διαπιστώνεται ότι το ResNet-18 είχε σταθερά καλές επιδόσεις σε πολλαπλές μετρικές, παρουσιάζοντας συχνά τις υψηλότερες τιμές. Αντίθετα, το MobileNet\_v2, παρά τις εξαιρετικές επιδόσεις του στο recall, παρουσίαζε χαμηλά αποτελέσματα σε όλες τις υπόλοιπες μετρικές (Precision, F1-Score, Accuracy). Το ResNet-34 πετυχαίνει και αυτό αξιοσημείωτα αποτελέσματα στο recall, αλλά υστερεί ελαφρώς σε precision σε σύγκριση με το ResNet-18 και το DenseNet-121. Τέλος, το DenseNet-121 επιδεικνύει συνέπεια αλλά συχνά υπολείπεται του ResNet-18 σε precision και accuracy ανάλογα με τα συνθετικά δείγματα που χρησιμοποιήθηκαν.

Ακόμη, λαμβάνοντας υπόψη τον αντίκτυπο των δειγμάτων cGAN στην απόδοση του ταξινομητή, τα συνθετικά δείγματα από το cGAN(128x128) τείνουν να έχουν καλύτερο αντίκτυπο στην ενίσχυση των μετρικών precision, F1-Score, accuracy, που είναι κρίσιμοι δείκτες απόδοσης για έναν ταξινομητή σε σύγκριση με τα δείγματα από τα cGAN(RMSprop) και cGAN(Adam). Επιπλέον, το ResNet-18 αναδεικνύεται ως ένα ευέλικτο προ-εκπαιδευμένο μοντέλο που παρουσιάζει αρκετά υψηλές τιμές σε διάφορες μετρικές απόδοσης με όποια σχεδόν συνθετικά δείγματα από cGAN και αν χρησιμοποιήθηκαν.

Στη συνέχεια, συγκρίθηκε η απόδοση του μοντέλου ως προς τις μετρικές precision, recall, f1-score και accuracy συγκρίνοντας τα αποτελέσματα του αρχικού μοντέλου (No GAN) στο οποίο δε χρησιμοποιήθηκαν συνθετικές εικόνες σε σχέση με τα μοντέλα που χρησιμοποιήθηκαν συνθετικές εικόνες παραγόμενες από τα παραπάνω cGAN. Έτσι, μελετήθηκε η συμβολή της χρήσης του cGAN στη βελτιστοποίηση του ταξινομητή ως προς τη μειωτική κλάση (Normal) αλλά και η επίδραση που είχαν τα προεκπαιδευμένα μοντέλα στην απόδοση του ταξινομητή. Από αυτή τη συγκριτική ανάλυση, προέκυψε ότι το μοντέλο με συνθετικά δείγματα από το cGAN (128x128) σε συνδυασμό με το ResNet-18 υπερτερεί σταθερά έναντι των άλλων μοντέλων όσον αφορά το precision, το F1-Score και το Accuracy.

Συμπερασματικά, μετά τη διεξαγωγή αρκετών πειραμάτων και διαφορετικών αρχιτεκτονικών επιτεύχθηκε ο στόχος και με τη χρήση cGAN βελτιστοποιήθηκε η απόδοση του ταξινομητή ως προς τη μειωτική κλάση (Normal) του Chest Xray Dataset που επιλέχθηκε για τις ανάγκες της συγκεκριμένης εργασίας.

Συνολικά, αναγνωρίζεται το μετασχηματιστικό δυναμικό των cGANs για τον μετριασμό της ανισορροπίας των κλάσεων στην ανάλυση ιατρικών εικόνων. Ωστόσο, απαιτείται περαιτέρω έρευνα για την ενίσχυση και επέκταση αυτών των μεθοδολογιών, διασφαλίζοντας παράλληλα τη δεοντολογική και ασθενοκεντρική εφαρμογή τους.

Λέξεις Κλειδιά: Generative Adversarial Networks, CNN, Class Imbalance, Pneumonia, Classification, Medical Imaging, Deep Learning, Diagnosis, cGANs, GANs, WGAN, Oversampling

Επιβλέπων: Ηλίας Μαγκλογιάννης  
Ακαδημαϊκή Θέση: Καθηγητής

## **Ευχαριστίες**

Θα ήθελα να εκφράσω την ειλικρινή μου ευγνωμοσύνη σε πολλά άτομα και ιδρύματα που διαδραμάτισαν καθοριστικό ρόλο στο ακαδημαϊκό μου ταξίδι κατά τη διάρκεια του μεταπτυχιακού μου προγράμματος.

Πρώτα απ' όλα, εκφράζω τις βαθύτατες ευχαριστίες μου στον επιβλέποντα καθηγητή μου, Καθηγητή Ηλία Μαγκλογιάννη, για την καθοδήγηση, τις γνώσεις και την αμέριστη υποστήριξη καθ' όλη τη διάρκεια της εργασίας μου. Είμαι ευγνώμων επίσης στον βοηθό του επιβλέποντα καθηγητή μου Νάσο Καλλιπολίτη για τη συνεχή βοήθεια, την υπομονή και επίλυση κάθε είδους απορίας που προέκυπτε κατά τη διάρκεια συγγραφής της διπλωματικής εργασίας.

Εκφράζω επίσης την ειλικρινή εκτίμησή μου στην εταιρεία που εργάζομαι, CMT ΠΡΟΟΠΤΙΚΗ ΕΠΕ, για την υποστήριξη και την ενθάρρυνση, επιτρέποντας μου να επιτύχω ισορροπία μεταξύ των επαγγελματικών και ακαδημαϊκών υποχρεώσεων και να εργαστώ σε ένα εργασιακό περιβάλλον που παρείχε πρακτικές γνώσεις εμπλουτίζοντας την ερευνητική και μαθησιακή μου εμπειρία.

Επιπλέον, θα ήθελα να ευχαριστήσω το διδακτικό προσωπικό και τους καθηγητές του Τμήματος Ψηφιακών Συστημάτων του Πανεπιστημίου Πειραιά

που μου μετέδωσαν τις γνώσεις τους κατά τη διάρκεια του μεταπτυχιακού μου προγράμματος. Η αφοσίωσή τους στη διδασκαλία και η προθυμία τους να κάνουν το κάτι παραπάνω για τους φοιτητές τους ήταν πραγματικά αξιέπαινες. Συνολικά, είμαι ευγνώμων σε όλους όσους αποτέλεσαν μέρος του ακαδημαϊκού μου ταξιδιού, και ανυπομονώ να εφαρμόσω τις γνώσεις και τις δεξιότητες που απέκτησα για να συνεισφέρω ουσιαστικά στον τομέα μου.

*Οι απόψεις που εκφράζονται εδώ, τα ευρήματα και τα συμπεράσματα είναι αυτά του συγγραφέως και δεν εκφράζουν τις απόψεις του «χρηματοδότη», του Πανεπιστημίου Πειραιώς.*

## Περιεχόμενα

<b>1 ΚΕΦΑΛΑΙΟ 1: ΕΙΣΑΓΩΓΗ</b> .....	<b>15</b>
1.1 ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ .....	15
1.2 Η ΑΝΙΣΟΡΡΟΠΙΑ ΤΩΝ ΚΛΑΣΕΩΝ .....	17
1.3 Η ΣΗΜΑΝΤΙΚΟΤΗΤΑ ΤΗΣ ΑΝΤΙΜΕΤΩΠΙΣΗΣ ΤΗΣ ΑΝΙΣΟΡΡΟΠΙΑΣ ΣΤΗΝ ΙΑΤΡΙΚΗ ΔΙΑΓΝΩΣΗ .....	18
1.4 ΔΟΜΗ ΤΗΣ ΕΡΓΑΣΙΑΣ .....	20
<b>2 ΚΕΦΑΛΑΙΟ 2: ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΕΠΙΣΚΟΠΗΣΗ</b> .....	<b>22</b>
2.1 ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΕΠΙΣΚΟΠΗΣΗ .....	22
2.2 ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ.....	38
2.2.1 Μηχανική Μάθηση .....	38
2.2.2 Μάθηση με επίβλεψη ( <i>Supervised Learning</i> ).....	38
2.2.3 Μάθηση χωρίς επίβλεψη ( <i>Unsupervised Learning</i> ).....	39
2.2.4 Ενισχυτική μάθηση ( <i>Reinforcement Learning</i> ).....	39
2.2.5 Βαθιά Μάθηση ( <i>Deep Learning</i> ).....	39
2.2.6 Ιστορική Αναδρομή .....	40
2.2.7 Τεχνητά Νευρωνικά Δίκτυα ( <i>Artificial Neural Networks</i> ).....	41
2.2.8 Συνελικτικά Νευρωνικά Δίκτυα .....	42
2.3 ΤΕΧΝΙΚΕΣ ΑΝΤΙΜΕΤΩΠΙΣΗΣ ΤΗΣ ΑΝΙΣΟΡΡΟΠΙΑΣ .....	46
2.3.1 <i>Undersampling</i> .....	47
2.3.2 <i>Oversampling</i> .....	49
2.3.3 <i>Cost-Sensitive Training</i> .....	52
2.3.4 <i>Ensemble Models</i> .....	54
2.3.5 <i>Generative Adversarial Networks</i> .....	55
2.4 GENERATIVE ADVERSARIAL NETWORKS (ΠΑΡΑΓΩΓΙΚΑ ΑΝΤΙΠΑΛΙΚΑ ΔΙΚΤΥΑ) 56	
2.4.1 Εισαγωγή στα GANs.....	56
2.4.2 Δομή και Λειτουργία.....	57
2.4.3 Εκπαίδευση- <i>Minimax Game</i> .....	60
2.4.4 Περιορισμοί των GANs .....	63
2.4.5 Αρχιτεκτονικές GANs.....	65
2.4.6 Μετρικές Αξιολόγησης GANs .....	76
<b>3 ΚΕΦΑΛΑΙΟ 3: ΜΕΘΟΔΟΛΟΓΙΑ</b> .....	<b>78</b>

3.1	CLASSIFIER .....	81
3.1.1	Σύντομη περιγραφή αρχιτεκτονικής.....	81
3.1.2	Εκπαίδευση CNN Classifier.....	81
3.2	GAN.....	85
3.2.1	Δημιουργία GAN .....	85
3.2.2	Discriminator .....	86
3.2.3	Generator.....	88
3.2.4	Εκπαίδευση DCGAN .....	91
3.2.5	WGAN& WGAN-GP .....	97
3.2.6	cGAN.....	105
<b>4</b>	<b>ΚΕΦΑΛΑΙΟ 4: ΠΕΙΡΑΜΑΤΑ-ΑΠΟΤΕΛΕΣΜΑΤΑ.....</b>	<b>118</b>
4.1	ΤΟ DATASET .....	118
4.1.1	Ανάλυση & Προεπεξεργασία .....	119
4.2	ΔΗΜΙΟΥΡΓΙΑ CLASSIFIER.....	122
4.2.1	Αποτελέσματα .....	125
4.3	CLASSIFIER ΜΕ ΧΡΗΣΗ WEIGHTED RANDOM SAMPLER .....	129
4.3.1	Αποτελέσματα .....	130
4.4	CLASSIFIER ΜΕ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΔΕΙΓΜΑΤΩΝ ΑΠΟ ΤΟ cGAN.....	131
4.4.1	Αποτελέσματα με χρήση εικόνων από cGAN(RMSPProp).....	133
4.4.2	Αποτελέσματα με χρήση εικόνων από cGAN(64 εποχές και Adam Optimizer).....	139
4.4.3	Αποτελέσματα με χρήση εικόνων από cGAN(5 εποχές 128x128 μέγεθος) .....	144
4.5	ΕΞΟΜΟΙΩΣΗ ΤΗΣ ΜΕΘΟΔΟΛΟΓΙΑΣ ΑΛΛΩΝ ΕΡΕΥΝΗΤΩΝ .....	150
4.5.1	Αποτελέσματα Classifier με χρήση εικόνων από cGAN (200 εποχές).....	155
4.5.2	Αποτελέσματα Classifier δημοσίευσης με χρήση συνθετικών δειγμάτων από ACGAN .....	155
<b>5</b>	<b>ΚΕΦΑΛΑΙΟ 5: ΣΥΖΗΤΗΣΗ-ΣΧΟΛΙΑΣΜΟΣ ΕΠΙ ΤΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ</b>	<b>156</b>
5.1	ΣΥΓΚΡΙΤΙΚΑ ΑΠΟΤΕΛΕΣΜΑΤΑ ΜΕΤΑΞΥ ΤΩΝ ΠΑΡΑΠΑΝΩ ΜΟΝΤΕΛΩΝ ΠΟΥ ΚΑΝΟΥΝ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΔΕΙΓΜΑΤΩΝ ΑΠΟ ΤΟ cGAN .....	156

5.2	ΣΥΓΚΡΙΤΙΚΑ ΑΠΟΤΕΛΕΣΜΑΤΑ ΜΕΤΑΞΥ ΤΩΝ ΜΟΝΤΕΛΩΝ ΠΟΥ ΚΑΝΟΥΝ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΔΕΙΓΜΑΤΩΝ ΑΠΟ ΤΟ CGAN ΜΕ ΤΟΝ ΑΡΧΙΚΟ CLASSIFIER .....	159
5.3	ΣΥΓΚΡΙΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ ΜΕ ΔΗΜΟΣΙΕΥΣΗ.....	163
<b>6</b>	<b>ΚΕΦΑΛΑΙΟ 6: ΣΥΜΠΕΡΑΣΜΑΤΑ &amp; ΕΠΙΛΟΓΟΣ .....</b>	<b>165</b>
6.1	ΣΥΜΠΕΡΑΣΜΑΤΑ.....	165
6.2	ΕΠΙΛΟΓΟΣ-ΜΕΛΛΟΝΤΙΚΕΣ ΠΡΟΣΕΓΓΙΣΕΙΣ .....	167
	<b>ΒΙΒΛΙΟΓΡΑΦΙΑ .....</b>	<b>171</b>



## Λίστα Εικόνων

ΕΙΚΟΝΑ 1 ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ- ΚΑΤΗΓΟΡΙΕΣ ΚΑΙ ΕΦΑΡΜΟΓΕΣ (ΠΗΓΗ ΑΠΟ ABDUL WAHID) .....	38
ΕΙΚΟΝΑ 2 Ο ΒΙΟΛΟΓΙΚΟΣ ΝΕΥΡΩΝΑΣ ΕΝΑΝΤΙ ΤΟΥ ΤΕΧΝΗΤΟΥ ΝΕΥΡΩΝΑ .....	41
ΕΙΚΟΝΑ 3 Η ΔΟΜΗ ΕΝΟΣ ΑΠΛΟΥ CNN.....	42
ΕΙΚΟΝΑ 4 Η ΣΥΝΔΕΣΗ ΜΕΤΑΞΥ ΤΟΥ ΤΕΛΕΥΤΑΙΟΥ ΕΠΙΠΕΔΟΥ ΕΝΟΣ CNN ΜΕ ΤΟ ΠΛΗΡΩΣ ΣΥΝΔΕΜΕΝΟ ΕΠΙΠΕΔΟ .....	43
ΕΙΚΟΝΑ 5 ΠΑΡΑΔΕΙΓΜΑ ΕΝΟΣ CNN ΜΕΧΡΙ ΤΟ ΣΤΑΔΙΟ ΤΗΣ ΤΑΞΙΝΟΜΗΣΗΣ .....	44
ΕΙΚΟΝΑ 6 ΔΙΑΔΙΚΑΣΙΑ ΕΚΠΑΙΔΕΥΣΗΣ ΕΝΟΣ CNN [71].....	44
ΕΙΚΟΝΑ 7 Η ΚΑΤΑΝΟΜΗ ΤΩΝ ΔΕΔΟΜΕΝΩΝ ΜΕ ΤΗ ΧΡΗΣΗ ΤΕΧΝΙΚΩΝ ΠΡΟΕΠΕΞΕΡΓΑΣΙΑΣ ΔΕΔΟΜΕΝΩΝ (OVESAMPLING, UNDERSAMPLING, HYBRID) [72]. .....	47
ΕΙΚΟΝΑ 8 ΣΤΡΑΤΗΓΙΚΕΣ ΑΝΤΙΜΕΤΩΠΙΣΗΣ ΤΗΣ ΑΝΙΣΟΡΡΟΠΙΑΣ ΤΩΝ ΚΛΑΣΕΩΝ .....	47
ΕΙΚΟΝΑ 9 ΕΞΙΣΟΡΡΟΠΗΣΗ ΤΩΝ ΔΕΔΟΜΕΝΩΝ ΜΕ ΤΗ ΜΕΘΟΔΟ ΤΗΣ ΥΠΟΔΕΙΓΜΑΤΟΛΗΨΙΑΣ .....	48
ΕΙΚΟΝΑ 10 ΕΞΙΣΟΡΡΟΠΗΣΗ ΤΩΝ ΔΕΔΟΜΕΝΩΝ ΜΕ ΤΗ ΜΕΘΟΔΟ ΤΗΣ ΥΠΕΡΔΕΙΓΜΑΤΟΛΗΨΙΑΣ .....	50
ΕΙΚΟΝΑ 11 ΕΦΑΡΜΟΓΗ ΤΗΣ SMOTE ΣΤΗ ΜΕΙΟΨΗΦΙΚΗ ΚΛΑΣΗ .....	50
ΕΙΚΟΝΑ 12 ΔΗΜΙΟΥΡΓΙΑ ΣΥΝΘΕΤΙΚΩΝ ΔΕΔΟΜΕΝΩΝ ΜΕ ΤΗΝ ΠΡΟΣΕΓΓΙΣΗ ADASYN.....	51
ΕΙΚΟΝΑ 13 BAGGING VS BOOSTING .....	54
ΕΙΚΟΝΑ 14 Η ΔΟΜΗ ΛΕΙΤΟΥΡΓΙΑΣ ΕΝΟΣ ΔΙΚΤΥΟΥ GAN.....	55
ΕΙΚΟΝΑ 15 Η ΡΑΓΔΑΙΑ ΕΞΕΛΙΞΗ ΤΩΝ GANS ΑΠΟ ΤΟ 2014 ΜΕΧΡΙ ΤΟ 2017 .....	57
ΕΙΚΟΝΑ 16 Η ΠΑΡΑΓΟΜΕΝΗ ΕΙΚΟΝΑ ΑΠΟ ΤΟΝ GENERATOR ΔΙΝΕΤΑΙ ΣΤΟ DISCRIMINATOR ΜΑΖΙ ΜΕ ΜΙΑ ΠΡΑΓΜΑΤΙΚΗ ΚΑΙ ΑΥΤΟΣ ΜΕ ΤΗ ΣΕΙΡΑ ΤΟΥ ΑΠΟΦΑΣΙΖΕΙ ΑΝ ΕΙΝΑΙ ΑΛΗΘΙΝΗ Η ΨΕΥΤΙΚΗ .....	58
ΕΙΚΟΝΑ 17 Η ΒΑΣΙΚΗ ΛΕΙΤΟΥΡΓΙΑ ΕΝΟΣ GAN .....	59
ΕΙΚΟΝΑ 18 ΤΟ ΜΙΝΜΑΧ GAME ΕΝΟΣ GAN ΜΕΧΡΙ ΝΑ ΦΤΑΣΕΙ ΣΕ ΙΣΟΡΡΟΠΙΑ NASH [48].....	61
ΕΙΚΟΝΑ 19 ΥΠΟΛΟΓΙΣΜΟΣ ΑΠΩΛΕΙΑΣ BCE.....	62
ΕΙΚΟΝΑ 20 ΔΙΑΔΙΚΑΣΙΑ ΕΚΠΑΙΔΕΥΣΗΣ GAN ΚΑΙ Η ΧΡΗΣΗ ΤΗΣ BCE .....	62
ΕΙΚΟΝΑ 21 MODE COLLAPSE ΠΑΡΑΔΕΙΓΜΑ.....	64
ΕΙΚΟΝΑ 22 ΠΑΡΑΔΕΙΓΜΑ ΔΙΚΤΥΟΥ GENERATOR ΜΕΣΑ ΣΕ ΕΝΑ DCGAN.....	66
ΕΙΚΟΝΑ 23 ΠΑΡΑΔΕΙΓΜΑ ΔΙΚΤΥΟΥ DISCRIMINATOR ΜΕΣΑ ΣΕ ΕΝΑ DCGAN .....	67
ΕΙΚΟΝΑ 24 Ο ΑΛΓΟΡΙΘΜΟΣ ΕΝΟΣ WGAN ΜΕ ΤΗ ΧΡΗΣΗ ΤΗΣ ΜΕΘΟΔΟΥ WEIGHT CLIPPING.[51] .....	70
ΕΙΚΟΝΑ 25 Ο DISCRIMINATOR ΚΑΙ CRITIC ΚΑΤΑ ΤΗΝ ΕΚΜΑΘΗΣΗ ΤΗΣ ΔΙΑΚΡΙΣΗΣ ΔΥΟ ΚΑΝΟΝΙΚΩΝ ΚΑΤΑΝΟΜΩΝ. ΌΠΩΣ ΦΑΙΝΕΤΑΙ, Ο DISCRIMINATOR ΕΝΟΣ ΜΙΝΙΜΑΧ GAN ΕΧΕΙ ΚΟΡΕΣΤΕΙ ΚΑΙ ΟΔΗΓΕΙ ΣΕ ΕΞΑΦΑΝΙΖΟΜΕΝΕΣ ΚΛΙΣΕΙΣ, ΕΝΩ Ο CRITIC ΤΟΥ WGAN ΠΑΡΕΧΕΙ ΠΟΛΥ ΠΙΟ ΚΑΘΑΡΕΣ ΚΛΙΣΕΙΣ ΣΕ ΟΛΑ ΤΑ ΜΕΡΗ ΤΟΥ ΧΩΡΟΥ [51].....	72
ΕΙΚΟΝΑ 26 Ο ΑΛΓΟΡΙΘΜΟΣ ΕΚΠΑΙΔΕΥΣΗΣ ΕΝΟΣ WGAN ΜΕ ΤΗ ΧΡΗΣΗ ΤΗΣ GRADIENT PENALTY [86] ..	73
ΕΙΚΟΝΑ 27 Η ΔΟΜΗ ΕΝΟΣ ΑΠΛΟΥ CGAN [49].....	74
ΕΙΚΟΝΑ 28 CGAN GENERATOR.....	75
ΕΙΚΟΝΑ 29 CGAN DISCRIMINATOR .....	75

ΕΙΚΟΝΑ 30 CGAN -Ο GENERATOR ΧΡΗΣΙΜΟΠΟΙΕΙ ΕΝΑ ΔΙΑΝΥΣΜΑ ΘΟΡΥΒΟΥ Z ΚΑΙ ΜΙΑ ΕΤΙΚΕΤΑ Y (ΜΙΑ ΑΠΟ ΤΙΣ ΠΙΘΑΝΕΣ N) ΣΑΝ ΕΙΣΟΔΟ ΓΙΑ ΝΑ ΠΑΡΑΞΕΙ ΕΝΑ ΨΕΥΤΙΚΟ ΠΑΡΑΔΕΙΓΜΑ $X^* Y$ ΤΟ ΟΠΟΙΟ ΠΡΟΣΠΑΘΕΙ ΝΑ ΔΕΙΧΝΕΙ ΡΕΑΛΙΣΤΙΚΟ ΚΑΙ ΤΑΙΡΙΑΣΤΟ ΜΕ ΤΗΝ ΕΤΙΚΕΤΑ Y.....	76
ΕΙΚΟΝΑ 31 Η ΡΟΗ ΕΡΓΑΣΙΩΝ ΚΑΤΑ ΤΗΝ ΕΚΠΑΙΔΕΥΣΗ ΤΟΥ ΤΑΞΙΝΟΜΗΤΗ.....	79
ΕΙΚΟΝΑ 32 Η ΡΟΗ ΕΡΓΑΣΙΩΝ ΠΟΥ ΑΚΟΛΟΥΘΗΘΗΚΕ ΓΙΑ ΤΗΝ ΕΠΙΛΟΓΗ ΤΩΝ ΚΑΛΥΤΕΡΩΝ ΜΟΝΤΕΛΩΝ CGAN.....	80
ΕΙΚΟΝΑ 33 Η ΡΟΗ ΕΡΓΑΣΙΩΝ ΓΙΑ ΤΗ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΤΗΣ ΜΕΙΩΤΙΚΗΣ ΚΛΑΣΗΣ NORMAL ΜΕ ΤΗ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΔΕΙΓΜΑΤΩΝ ΑΠΟ CGAN.....	80
ΕΙΚΟΝΑ 34 NVIDIA CUDA.....	83
ΕΙΚΟΝΑ 35 ΑΛΓΟΡΙΘΜΟΣ ΕΚΠΑΙΔΕΥΣΗΣ CNN.....	84
ΕΙΚΟΝΑ 36 ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ DATASET ΣΕ 64x64.....	86
ΕΙΚΟΝΑ 37 GAN-DISCRIMINATOR.....	87
ΕΙΚΟΝΑ 38 GAN-GENERATOR.....	89
ΕΙΚΟΝΑ 39 DCGAN -ΑΛΓΟΡΙΘΜΟΣ ΕΚΠΑΙΔΕΥΣΗΣ.....	93
ΕΙΚΟΝΑ 40 ΕΚΠΑΙΔΕΥΣΗ DCGAN ΣΕ CHEST XRAY DATASET- ΑΡΧΗ.....	95
ΕΙΚΟΝΑ 41 ΕΚΠΑΙΔΕΥΣΗ DCGAN ΣΕ CHEST XRAY DATASET- ΜΕΤΑ ΑΠΟ ΜΕΡΙΚΕΣ ΕΠΑΝΑΛΗΨΕΙΣ.....	96
ΕΙΚΟΝΑ 42 ΕΚΠΑΙΔΕΥΣΗ DCGAN ΣΕ CHEST XRAY DATASET ΤΕΛΟΣ ΕΚΠΑΙΔΕΥΣΗΣ.....	97
ΕΙΚΟΝΑ 43 WGAN-GP ΕΝΑΝΤΙΟΝ WGAN WEIGHT CLIPPING:ΑΡΧΗ ΕΚΠΑΙΔΕΥΣΗΣ.....	101
ΕΙΚΟΝΑ 44 WGAN-GP ΕΝΑΝΤΙΟΝ WGAN WEIGHT CLIPPING: STEP 20.....	102
ΕΙΚΟΝΑ 45 WGAN-GP ΕΝΑΝΤΙΟΝ WGAN WEIGHT CLIPPING: STEP 32.....	103
ΕΙΚΟΝΑ 46 WGAN-GP ΕΝΑΝΤΙΟΝ WGAN WEIGHT CLIPPING: ΤΕΛΟΣ ΕΚΠΑΙΔΕΥΣΗΣ.....	104
ΕΙΚΟΝΑ 47 CGAN ΒΡΟΓΧΟΣ ΕΚΠΑΙΔΕΥΣΗΣ.....	108
ΕΙΚΟΝΑ 48 CGAN ΣΤΟ MNIST ΑΡΧΗ ΕΚΠΑΙΔΕΥΣΗΣ.....	109
ΕΙΚΟΝΑ 49 CGAN ΣΤΟ MNIST ΣΤΗ ΜΕΣΗ ΕΚΠΑΙΔΕΥΣΗ.....	110
ΕΙΚΟΝΑ 50 CGAN ΣΤΟ MNIST ΣΤΟ ΤΕΛΟΣ ΤΗΣ ΕΚΠΑΙΔΕΥΣΗΣ.....	111
ΕΙΚΟΝΑ 51 ΕΞΑΓΟΜΕΝΑ ΣΥΝΘΕΤΙΚΑ ΔΕΙΓΜΑΤΑ ΑΠΟ ΤΟ ΕΚΠΑΙΔΕΥΜΕΝΟ CGAN: (ΑΡΙΣΤΕΡΟ ΜΕΡΟΣ CGAN ΔΕΙΓΜΑΤΑ, ΔΕΞΙΟ ΜΕΡΟΣ ΑΛΗΘΙΝΑ ΔΕΙΓΜΑΤΑ).....	112
ΕΙΚΟΝΑ 52 CGAN ΑΠΟΤΕΛΕΣΜΑΤΑ ΜΕΤΑ ΑΠΟ 5 ΕΠΟΧΕΣ.....	113
ΕΙΚΟΝΑ 53 CGAN ΑΠΟΤΕΛΕΣΜΑΤΑ ΜΕΤΑ ΑΠΟ 64 ΕΠΟΧΕΣ.....	114
ΕΙΚΟΝΑ 54 CGAN ΑΠΟΤΕΛΕΣΜΑΤΑ ΧΩΡΙΣ GRADIENT PENALTY ΚΑΙ ΧΩΡΙΣ WASSERSTEIN DISTANCE ΜΕΤΑ ΑΠΟ 80 ΕΠΟΧΕΣ (MODE COLLAPSE).....	115
ΕΙΚΟΝΑ 55 CGAN ΑΠΟΤΕΛΕΣΜΑΤΑ ΜΕΤΑ ΑΠΟ 100 ΕΠΟΧΕΣ ΚΑΙ ΧΡΗΣΗ RMSPROP ΒΕΛΤΙΣΤΟΠΟΙΗΤΗ.....	116
ΕΙΚΟΝΑ 56 CGAN ΑΠΟΤΕΛΕΣΜΑΤΑ ΜΕΤΑ ΑΠΟ 5 ΕΠΟΧΕΣ ΜΕ ΜΕΓΕΘΟΣ ΕΙΚΟΝΑ 128x128(GOOGLE COLAB).....	117
ΕΙΚΟΝΑ 57 ΠΑΡΑΔΕΙΓΜΑ ΕΙΚΟΝΩΝ ΜΕ ΕΝΤΑΣΕΙΣ ΓΚΡΙΖΩΝ ΕΙΚΟΝΟΣΤΟΙΧΕΙΩΝ.....	118
ΕΙΚΟΝΑ 58 ΠΑΡΑΔΕΙΓΜΑ ΑΚΤΙΝΟΓΡΑΦΙΑΣ ΧΩΡΙΣ ΠΝΕΥΜΟΝΙΑ (ΑΡΙΣΤΕΡΑ) ΚΑΙ ΑΚΤΙΝΟΓΡΑΦΙΑΣ ΜΕ ΠΝΕΥΜΟΝΙΑ (ΔΕΞΙΑ).....	119
ΕΙΚΟΝΑ 59 ΔΟΜΗ ΦΑΚΕΛΩΝ DATASET.....	119

ΕΙΚΟΝΑ 60 ΣΥΝΟΛΙΚΟ ΠΛΗΘΟΣ ΕΙΚΟΝΩΝ ΓΙΑ ΚΑΘΕ ΚΛΑΣΗ ΑΝΑ ΦΑΚΕΛΟ.....	120
ΕΙΚΟΝΑ 61 Το DATASET ΤΩΝ ΑΚΤΙΝΟΓΡΑΦΙΩΝ ΘΩΡΑΚΟΣ ΠΑΡΟΥΣΙΑΖΕΙ ΤΟ ΠΡΟΒΛΗΜΑ ΤΗΣ ΑΝΙΣΟΡΡΟΠΙΑΣ ΤΩΝ ΚΛΑΣΕΩΝ .....	121
ΕΙΚΟΝΑ 62 ΔΗΜΙΟΥΡΓΙΑ TRAIN SET ΚΑΙ TEST SET.....	122
ΕΙΚΟΝΑ 63 ΔΙΑΧΩΡΙΣΜΟΣ ΤΟΥ TRAINSET ΣΕ ΔΕΔΟΜΕΝΑ ΕΚΠΑΙΔΕΥΣΗΣ ΚΑΙ ΔΕΔΟΜΕΝΑ ΔΟΚΙΜΗΣ.....	123
ΕΙΚΟΝΑ 64 ΤΥΧΑΙΕΣ ΕΙΚΟΝΕΣ ΑΠΟ ΚΑΘΕ ΚΛΑΣΗ ΑΡΙΣΤΕΡΑ Η ΚΛΑΣΗ 0 (NORMAL) ΚΑΙ ΔΕΞΙΑ Η ΚΛΑΣΗ 1 (PNEUMONIA).....	123
ΕΙΚΟΝΑ 65 Το CUSTOMDATASETXRAY.PY Η ΔΟΜΗ ΤΗΣ ΚΛΑΣΗΣ ΠΟΥ ΔΙΑΧΕΙΡΙΖΕΤΑΙ ΤΟ DATASET ΜΕΣΩ ΤΟΥ PYTORCH .....	124
ΕΙΚΟΝΑ 66 CONFUSION MATRIX ΤΑΞΙΝΟΜΗΤΗ CNN (RESNET-18) .....	126
ΕΙΚΟΝΑ 67 CONFUSION MATRIX ΤΑΞΙΝΟΜΗΤΗ CNN (VGG16).....	126
ΕΙΚΟΝΑ 68 CONFUSION MATRIX ΤΑΞΙΝΟΜΗΤΗ CNN (DENSENET-121).....	127
ΕΙΚΟΝΑ 69 CONFUSION MATRIX ΤΑΞΙΝΟΜΗΤΗ CNN (MOBILENET_V2).....	127
ΕΙΚΟΝΑ 70 CONFUSION MATRIX ΤΑΞΙΝΟΜΗΤΗ CNN (RESNET-34) .....	128
ΕΙΚΟΝΑ 71 ΔΗΜΙΟΥΡΓΙΑ WEIGHTEDRANDOMSAMPLER ΚΑΙ ΑΝΑΘΕΣΗ ΒΑΡΩΝ.....	130
ΕΙΚΟΝΑ 72 CONFUSION MATRIX ΤΑΞΙΝΟΜΗΤΗ CNN ΜΕ ΧΡΗΣΗ WEIGHTEDRANDOMSAMPLER.....	130
ΕΙΚΟΝΑ 73 ΟΙ ΝΕΕΣ ΣΥΝΘΕΤΙΚΕΣ ΕΙΚΟΝΕΣ ΤΗΣ ΚΛΑΣΗΣ NORMAL ΜΕΤΑ ΤΗ ΧΡΗΣΗ ΤΟΥ ΜΟΝΤΕΛΟΥ.....	132
ΕΙΚΟΝΑ 74 CHESTXRAY DATASET ΜΕΤΑ ΤΗΝ ΠΡΟΣΘΗΚΗ ΤΩΝ ΝΕΩΝ ΣΥΝΘΕΤΙΚΩΝ ΕΙΚΟΝΩΝ ΤΗΣ ΚΛΑΣΗΣ NORMAL.....	133
ΕΙΚΟΝΑ 75 ΟΠΤΙΚΟΠΟΙΗΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ CLASSIFIER ΜΕ ΤΗ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΔΕΙΓΜΑΤΩΝ ΑΠΟ ΤΟ cGAN (RMSPROP).....	135
ΕΙΚΟΝΑ 76 CONFUSION MATRIX ΓΙΑ ΤΑΞΙΝΟΜΗΤΗ(RESNET-18) ΜΕ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΕΙΚΟΝΩΝ (CGAN RMSPROOP).....	136
ΕΙΚΟΝΑ 77 ΚΑΜΠΥΛΗ ROC .....	136
ΕΙΚΟΝΑ 78 CONFUSION MATRIX ΓΙΑ ΤΑΞΙΝΟΜΗΤΗ (MOBILENET_V2) ΜΕ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΕΙΚΟΝΩΝ (CGAN RMSPROOP).....	137
ΕΙΚΟΝΑ 79 ΚΑΜΠΥΛΗ ROC (MOBILENET_V2).....	137
ΕΙΚΟΝΑ 80 CONFUSION MATRIX ΓΙΑ ΤΑΞΙΝΟΜΗΤΗ (RESNET34) ΜΕ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΕΙΚΟΝΩΝ (CGAN RMSPROOP).....	138
ΕΙΚΟΝΑ 81ΚΑΜΠΥΛΗ ROC (RESNET34) .....	138
ΕΙΚΟΝΑ 82 CONSUSION MATRIX (DENSENET121) .....	139
ΕΙΚΟΝΑ 83 ΚΑΜΠΥΛΗ ROC (DENSENET-121) .....	139
ΕΙΚΟΝΑ 84 ΟΠΤΙΚΟΠΟΙΗΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ CLASSIFIER ΜΕ ΤΗ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΔΕΙΓΜΑΤΩΝ ΑΠΟ ΤΟ cGAN (ADAM).....	141
ΕΙΚΟΝΑ 85 CONFUSION MATRIX ΓΙΑ ΤΑΞΙΝΟΜΗΤΗ (RESNET-18) ΜΕ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΕΙΚΟΝΩΝ (CGAN 64 ΕΠΟΧΕΣ ADAM OPTIMIZER) .....	142
ΕΙΚΟΝΑ 86 ΚΑΜΠΥΛΗ ROC (RESNET-18).....	142
ΕΙΚΟΝΑ 87 CONFUSION MATRIX ΓΙΑ ΤΑΞΙΝΟΜΗΤΗ (MOBILENETV2) ΜΕ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΕΙΚΟΝΩΝ (CGAN 64 ΕΠΟΧΕΣ ADAM OPTIMIZER) .....	142

ΕΙΚΟΝΑ 88 ΚΑΜΠΥΛΗ ROC(MOBILNETV2) .....	143
ΕΙΚΟΝΑ 89 CONFUSION MATRIX ΓΙΑ ΤΑΞΙΝΟΜΗΤΗ (RESNET34) ΜΕ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΕΙΚΟΝΩΝ (CGAN 64 ΕΠΟΧΕΣ ADAM OPTIMIZER) .....	143
ΕΙΚΟΝΑ 90 ΚΑΜΠΥΛΗ ROC(RESNET34) .....	143
ΕΙΚΟΝΑ 91 CONFUSION MATRIX ΓΙΑ ΤΑΞΙΝΟΜΗΤΗ (DENSENET-121) ΜΕ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΕΙΚΟΝΩΝ (CGAN 64 ΕΠΟΧΕΣ ADAM OPTIMIZER) .....	144
ΕΙΚΟΝΑ 92 ΚΑΜΠΥΛΗ ROC(DENSENET-121).....	144
ΕΙΚΟΝΑ 93 ΟΠΤΙΚΟΠΟΙΗΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ CLASSIFIER ΜΕ ΤΗ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΔΕΙΓΜΑΤΩΝ ΑΠΟ ΤΟ CGAN (128x128) .....	146
ΕΙΚΟΝΑ 94 CONFUSION MATRIX ΓΙΑ ΤΑΞΙΝΟΜΗΤΗ(RESNET18) ΜΕ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΕΙΚΟΝΩΝ (CGAN 64 ΕΠΟΧΕΣ 128x128 ΜΕΓΕΘΟΣ).....	147
ΕΙΚΟΝΑ 95 ΚΑΜΠΥΛΗ ROC (RESNET18) .....	147
ΕΙΚΟΝΑ 96 CONFUSION MATRIX MOBILENET_V2 .....	148
ΕΙΚΟΝΑ 97 ΚΑΜΠΥΛΗ ROC (MOBILENET_V2).....	148
ΕΙΚΟΝΑ 98 CONFUSION MATRIX (RESNET-34) .....	149
ΕΙΚΟΝΑ 99 ΚΑΜΠΥΛΗ ROC (RESNET-34).....	149
ΕΙΚΟΝΑ 100 CONFUSION MATRIX (DENSENET-121).....	150
ΕΙΚΟΝΑ 101 ΚΑΜΠΥΛΗ ROC (DENSENET-121) .....	150
ΕΙΚΟΝΑ 102 Η ΑΠΩΛΕΙΑ ΤΟΥ DISCRIMINATOR ΚΑΙ ΤΗΝ ΕΚΠΑΙΔΕΥΣΗ 600 ΕΠΟΧΩΝ .....	152
ΕΙΚΟΝΑ 103 Η ΑΠΩΛΕΙΑ ΤΟΥ GENERATOR ΚΑΤΑ ΤΗΝ ΕΚΠΑΙΔΕΥΣΗ 600 ΕΠΟΧΩΝ .....	152
ΕΙΚΟΝΑ 104 Α) ΔΕΙΓΜΑΤΑ ΕΙΚΟΝΩΝ ΑΠΟ ΤΟ ΑΛΗΘΙΝΟ DATASET Β)ΤΗRESHOLDING ΤΩΝ ΑΛΗΘΙΝΩΝ ΔΕΙΓΜΑΤΩΝ ΑΥΤΩΝ Γ)ΣΥΝΘΕΤΙΚΑ ΔΕΙΓΜΑΤΑ ΑΠΟ CGAN ΑΝΑ 100 ΕΠΟΧΕΣ Δ) ΤΗRESHOLDING ΤΩΝ ΣΥΝΘΕΤΙΚΩΝ ΔΕΙΓΜΑΤΩΝ ΑΝΑ 100 ΕΠΟΧΕΣ.....	153
ΕΙΚΟΝΑ 105 CHEST XRAY DATASET ΜΕΤΑ ΤΗΝ ΠΡΟΣΘΗΚΗ 500 ΣΥΝΘΕΤΙΚΩΝ ΔΕΙΓΜΑΤΩΝ ΣΕ ΚΑΘΕ ΚΛΑΣΗ .....	154
ΕΙΚΟΝΑ 106 CONFUSION MATRIX ΓΙΑ ΤΑΞΙΝΟΜΗΤΗ ΜΕ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΕΙΚΟΝΩΝ (CGAN 200 ΕΠΟΧΕΣ ADAM OPTIMIZER).....	155
ΕΙΚΟΝΑ 107 ΣΥΓΚΡΙΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ ΜΕΤΑΞΥ ΟΛΩΝ ΤΩΝ ΠΑΡΑΠΑΝΩ.....	157
ΕΙΚΟΝΑ 108 ΑΠΟΤΕΛΕΣΜΑΤΑ CLASSIFIER ΜΕ ΧΡΗΣΗ CGAN ΕΝΑΝΤΙΟΝ CLASSIFIER ΧΩΡΙΣ ΧΡΗΣΗ CGAN .....	159
ΕΙΚΟΝΑ 109 ΑΠΟΤΕΛΕΣΜΑΤΑ ΤΟΥ CLASSIFIER ΜΕ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΔΕΙΓΜΑΤΩΝ ΑΠΟ CGAN ΕΝΑΝΤΙΟΝ ΤΟΥ CLASSIFIER ΤΗΣ ΔΗΜΟΣΙΕΥΣΗΣ[53] ΜΕ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΔΕΙΓΜΑΤΩΝ ΑΠΟ ACGAN .....	164

## Λίστα Πινάκων

ΠΙΝΑΚΑΣ 1 ΠΙΝΑΚΑΣ ΚΟΣΤΟΥΣ .....	53
ΠΙΝΑΚΑΣ 2 CNN ΥΠΕΡΠΑΡΑΜΕΤΡΟΙ ΕΚΠΑΙΔΕΥΣΗΣ.....	81
ΠΙΝΑΚΑΣ 3 Η ΠΟΡΕΙΑ ΤΟΥ ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΥ ΤΗΣ ΕΙΚΟΝΑΣ ΣΤΟΝ DISCRIMINATOR .....	88
ΠΙΝΑΚΑΣ 4 Η ΠΟΡΕΙΑ ΤΟΥ ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΥ ΤΟΥ ΔΙΑΝΥΣΜΑΤΟΣ ΘΟΥΡΥΒΟΥ ΣΤΟ GENERATOR ΚΑΙ Η ΜΕΤΑΤΡΟΠΗ ΤΟΥΣ ΣΕ ΜΙΑ ΕΙΚΟΝΑ 64X64.....	90
ΠΙΝΑΚΑΣ 5 ΑΡΧΙΚΟΠΟΙΗΣΗ ΥΠΕΡΠΑΡΑΜΕΤΡΩΝ DCGAN .....	91
ΠΙΝΑΚΑΣ 6 ΥΠΕΡΠΑΡΑΜΕΤΡΟΙ ΕΚΠΑΙΔΕΥΣΗΣ WGAN ΜΕ WEIGHT CLIPPING .....	99
ΠΙΝΑΚΑΣ 7 ΥΠΕΡΠΑΡΑΜΕΤΡΟΙ ΕΚΠΑΙΔΕΥΣΗΣ WGAN ΜΕ GRADIENT PENALTY .....	100
ΠΙΝΑΚΑΣ 8 ΥΠΕΡΠΑΡΑΜΕΤΡΟΙ CGAN.....	107
ΠΙΝΑΚΑΣ 9 ΥΠΕΡΠΑΡΑΜΕΤΡΟΙ ΤΟΥ CGAN ΣΤΟ MNIST .....	109
ΠΙΝΑΚΑΣ 10 ΥΠΕΡΠΑΡΑΜΕΤΡΟΙ ΤΟΥ CGAN ΣΤΟ CHESTXRAY DATASET .....	113
ΠΙΝΑΚΑΣ 11 ΜΕΤΡΙΚΕΣ ΑΞΙΟΛΟΓΗΣΗΣ CGAN .....	117
ΠΙΝΑΚΑΣ 12 ΚΑΤΑΝΟΜΗ ΔΕΔΟΜΕΝΩΝ .....	120
ΠΙΝΑΚΑΣ 13 ΑΠΟΔΟΣΗ CNN ΤΑΞΙΝΟΜΗΤΗ (5 ΕΠΟΧΕΣ) .....	125
ΠΙΝΑΚΑΣ 14 ΑΠΟΔΟΣΕΙΣ ΤΑΞΙΝΟΜΗΤΗ ΜΕ ΧΡΗΣΗ ΠΡΟ ΕΚΠΑΙΔΕΥΜΕΝΩΝ CNN ΜΟΝΤΕΛΩΝ (50 ΕΠΟΧΕΣ) ΧΩΡΙΣ ΣΥΝΘΕΤΙΚΑ ΔΕΙΓΜΑΤΑ .....	129
ΠΙΝΑΚΑΣ 15 ΑΠΟΔΟΣΗ CNN ΤΑΞΙΝΟΜΗΤΗ ΜΕ ΧΡΗΣΗ WEIGHTEDRANDOMSAMPLER (5 ΕΠΟΧΕΣ).....	130
ΠΙΝΑΚΑΣ 16 ΑΠΟΤΕΛΕΣΜΑΤΑ CLASSIFIER ΜΕ ΤΗ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΔΕΙΓΜΑΤΩΝ ΑΠΟ ΤΟ CGAN (RMSPROP).....	135
ΠΙΝΑΚΑΣ 17 ΑΠΟΤΕΛΕΣΜΑΤΑ CLASSIFIER ΜΕ ΤΗ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΔΕΙΓΜΑΤΩΝ ΑΠΟ ΤΟ CGAN (64 ΕΠΟΧΕΣ ADAM OPTIMIZER).....	141
ΠΙΝΑΚΑΣ 18 ΑΠΟΤΕΛΕΣΜΑΤΑ CLASSIFIER ΜΕ ΤΗ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΔΕΙΓΜΑΤΩΝ ΑΠΟ ΤΟ CGAN (5 ΕΠΟΧΕΣ 128X128 ΜΕΓΕΘΟΣ) .....	146
ΠΙΝΑΚΑΣ 19 ΥΠΕΡΠΑΡΑΜΕΤΡΟΙ ΕΚΠΑΙΔΕΥΣΗΣ CGAN.....	151
ΠΙΝΑΚΑΣ 20 ΥΠΕΡΠΑΡΑΜΕΤΡΟΙ ΕΚΠΑΙΔΕΥΣΗΣ CLASSIFIER .....	154
ΠΙΝΑΚΑΣ 21 ΑΠΟΤΕΛΕΣΜΑΤΑ CLASSIFIER ΜΕ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΔΕΙΓΜΑΤΩΝ ΑΠΟ CGAN (200 ΕΠΟΧΕΣ).....	155
ΠΙΝΑΚΑΣ 22 ΑΠΟΤΕΛΕΣΜΑΤΑ CLASSIFIER ΔΗΜΟΣΙΕΥΣΗΣ [53] ΜΕ ΧΡΗΣΗ ΣΥΝΘΕΤΙΚΩΝ ΔΕΙΓΜΑΤΩΝ ΑΠΟ ACGAN .....	155
ΠΙΝΑΚΑΣ 23 ΠΙΝΑΚΑΣ ΜΕΤΑΒΟΛΩΝ ΑΝΑ ΜΕΤΡΙΚΗ ΚΑΙ ΠΡΟΕΚΠΑΙΔΕΥΜΕΝΟ ΜΟΝΤΕΛΟ ΤΩΝ ΤΑΞΙΝΟΜΗΤΩΝ ΠΟΥ ΕΚΠΑΙΔΕΥΤΗΚΑΝ ΜΕ ΣΥΝΘΕΤΙΚΑ ΔΕΙΓΜΑΤΑ ΑΠΟ ΤΑ CGANS ΣΕ ΣΧΕΣΗ ΜΕ ΤΟ ΑΡΧΙΚΟ ΤΑΞΙΝΟΜΗΤΗ ΠΟΥ ΔΕΝ ΧΡΗΣΙΜΟΠΟΙΕΙ ΣΥΝΘΕΤΙΚΑ ΔΕΙΓΜΑΤΑ.....	160

**No table of figures entries found.**



# **1 ΚΕΦΑΛΑΙΟ 1: ΕΙΣΑΓΩΓΗ**

## **1.1 Περιγραφή του προβλήματος**

Η πνευμονία, η οποία χαρακτηρίζεται από τον Παγκόσμιο Οργανισμό Υγείας ως αναπνευστική λοίμωξη που προσβάλλει άτομα όλων των ηλικιών, ενέχει αυξημένο κίνδυνο για τα μικρά παιδιά και τους ενήλικες άνω των 65 ετών. Αυτή η μολυσματική πάθηση, που προκαλείται από ιούς ή βακτήρια, παραμένει σημαντική παγκόσμια αιτία παιδικής θνησιμότητας, με περισσότερους από 808.000 θανάτους παιδιών να καταγράφονται μόνο το 2017. Οι ασθενείς με πνευμονία αντιμετωπίζουν αναπνευστικές δυσκολίες λόγω της συσσώρευσης πύου και υγρού στους πνεύμονες, γεγονός που επηρεάζει τους μικροσκοπικούς αερόσακους, γνωστούς ως κυψελίδες. Η μετάδοση της πνευμονίας γίνεται μέσω της άμεσης επαφής με μολυσμένα άτομα, ενώ οι προϋπάρχουσες συνθήκες υγείας μπορούν να αυξήσουν τον κίνδυνο ανάπτυξης πνευμονίας [1].

Η διαγνωστική πορεία για την πνευμονία περιλαμβάνει μια σχολαστική αξιολόγηση, βασιζόμενη στα συμπτώματα και τα σημεία της οξείας λοίμωξης του κατώτερου αναπνευστικού συστήματος ως θεμελιώδη βάση. Οι ακτινογραφίες θώρακος, μια συμβατική και ευρέως χρησιμοποιούμενη απεικονιστική τεχνική, διαδραματίζουν καθοριστικό ρόλο στη διαγνωστική διαδικασία. [2] [3] Ωστόσο, η χρησιμότητα των ακτινογραφιών θώρακος δεν είναι χωρίς προκλήσεις. Παρά το γεγονός ότι αποτελούν ένα διαδεδομένο διαγνωστικό εργαλείο, η διαφοροποίηση της πνευμονίας από άλλες πνευμονικές παθήσεις παρουσιάζει δυσκολίες. Η ομοιότητα των θολεροτήτων στις ακτινογραφίες μεταξύ της πνευμονίας και άλλων ανωμαλιών, όπως ο καρκίνος του πνεύμονα και η υπερφόρτωση με υγρά, μπορεί να οδηγήσει σε παρερμηνείες.[4]

Πέρα από τις ακτινογραφίες θώρακος, οι προηγμένες απεικονιστικές μέθοδοι, όπως η αξονική τομογραφία (CT) και η μαγνητική τομογραφία (MRI), προσφέρουν πιο αποχρώσες πληροφορίες για τις περιπτώσεις πνευμονίας. Αυτές οι μέθοδοι παρέχουν λεπτομερείς εικόνες των πνευμόνων και μπορούν να βοηθήσουν στη διάκριση της πνευμονίας από άλλες πνευμονικές παθήσεις. Ωστόσο, αυτές οι εξελιγμένες τεχνικές συχνά συνοδεύονται από αυξημένο κόστος, πολυπλοκότητα και πιθανές παρενέργειες.

Οι περιορισμοί των παραδοσιακών διαγνωστικών μεθόδων, που βασίζονται σε ειδικούς ακτινολόγους, τονίζουν την ανάγκη για ακριβή και έγκαιρη ανίχνευση

της πνευμονίας. Στο πλαίσιο αυτό, τα τελευταία χρόνια έχει αυξηθεί το ενδιαφέρον για την αυτοματοποιημένη διάγνωση, με τη χρήση συστημάτων ανίχνευσης με τη βοήθεια υπολογιστή (computer-aided detection/diagnosis-CAD). [5] Οι προκλήσεις εξακολουθούν να υφίστανται στην ανάλυση των ακτινογραφιών θώρακος, όπου ακόμη και οι ειδικοί ακτινολόγοι μπορεί να δυσκολεύονται να διαφοροποιήσουν με ακρίβεια την πνευμονία από άλλες πνευμονικές παθήσεις.

Έτσι, η διασταύρωση της ιατρικής απεικόνισης και της διάγνωσης της πνευμονίας αντιμετωπίζει εγγενείς προκλήσεις, ιδίως στην ερμηνεία των ακτινογραφιών θώρακος. Οι περιορισμοί των παραδοσιακών μεθόδων, σε συνδυασμό με την πιθανότητα παρερμηνείας, αναδεικνύουν την επιτακτική ανάγκη για προηγμένες, ακριβείς και οικονομικά αποδοτικές διαγνωστικές προσεγγίσεις. Καθώς η τεχνολογία εξελίσσεται, η ενσωμάτωση αυτοματοποιημένων συστημάτων και προηγμένων τεχνικών απεικόνισης υπόσχεται την αντιμετώπιση αυτών των προκλήσεων και την αναδιαμόρφωση του τοπίου της διάγνωσης της πνευμονίας για τη βελτίωση των διαγνωστικών πλάνων των ασθενών. [3] [6]

Η μηχανική μάθηση εφαρμόζεται όλο και περισσότερο στον ιατρικό τομέα για να βοηθήσει στη διάγνωση, τη θεραπεία και την πρόβλεψη ασθενειών. Στα ιατρικά δεδομένα, οι αλγόριθμοι μηχανικής μάθησης χρησιμοποιούνται για την ανάλυση των δεδομένων του ασθενούς, συμπεριλαμβανομένων των δημογραφικών στοιχείων, του ιατρικού ιστορικού και των δεδομένων απεικόνισης, ώστε να γίνονται προβλέψεις και να παρέχονται προτάσεις διαχείρισης των περιστατικών και της νόσου. Αυτό επιτρέπει στους επαγγελματίες υγείας να λαμβάνουν τεκμηριωμένες αποφάσεις και να βελτιώνουν την έκβαση της νόσου.

Ορισμένες εφαρμογές της μηχανικής μάθησης στα ιατρικά δεδομένα περιλαμβάνουν την πρόβλεψη υποτροπών της νόσου, την ταξινόμηση διαγνωστικών εικόνων και την έγκαιρη αναγνώριση ασθενών υψηλού κινδύνου για άμεση παρέμβαση. Ενώ η βαθιά μάθηση έχει κάνει σημαντικά βήματα προόδου, δεν είναι αρκετά προηγμένη ώστε να αντικαταστήσει τους γιατρούς στην ιατρική διάγνωση. Αντίθετα, θα μπορούσε να χρησιμεύσει ως υποστηρικτικό εργαλείο για να τους βοηθήσει στις διαγνωστικές διαδικασίες. Μεταξύ άλλων, τα νευρωνικά δίκτυα είναι ιδιαίτερα πολύτιμα για τον χειρισμό



χρονοβόρων εργασιών, όπως η εξέταση ακτινογραφιών θώρακος για τον εντοπισμό ενδείξεων πνευμονίας.

Ωστόσο, η διαχείριση ιατρικών δεδομένων θέτει ορισμένες μοναδικές προκλήσεις, όπως η ανισορροπία των κλάσεων, τα ελλιπή ή μη ολοκληρωμένα δεδομένα, καθώς και ζητήματα ηθικής και προστασίας προσωπικών δεδομένων, τα οποία πρέπει να εξετάζονται προσεκτικά.

Η ανισορροπία κλάσεων είναι ένα κοινό πρόβλημα στα ιατρικά δεδομένα, όπου η κατανομή των κλάσεων σε ένα σύνολο δεδομένων είναι άνιση. Στις ιατρικές εφαρμογές, αυτό μπορεί να οδηγήσει σε σοβαρές συνέπειες, καθώς μπορεί να προκύψουν μεροληπτικές προβλέψεις, μειωμένη ακρίβεια και μη αποτελεσματικές διαγνώσεις. Για παράδειγμα, εάν ένα σύνολο δεδομένων έχει μεγάλο αριθμό ασθενών χωρίς τη νόσο και μόνο λίγα περιστατικά με κάποια σπάνια νόσο, ένας αλγόριθμος μηχανικής μάθησης που εκπαιδεύεται σε τέτοια δεδομένα είναι πιθανό να παραβλέψει τις περιπτώσεις σπάνιας νόσου. Στο πλαίσιο αυτό, η αντιμετώπιση της ανισορροπίας των κλάσεων είναι ζωτικής σημασίας για τη διασφάλιση της αξιοπιστίας και της αποτελεσματικότητας των ιατρικών διαγνώσεων καθώς και των θεραπευτικών αποφάσεων που βασίζονται σε αλγορίθμους μηχανικής μάθησης.

Στη συγκεκριμένη μελέτη, θα δοθεί έμφαση στην ανισορροπία των κλάσεων σε ιατρικά απεικονιστικά δεδομένα. Συγκεκριμένα, επιχειρείται η επίλυση του προβλήματος της ανισορροπίας των κλάσεων με σκοπό τη δημιουργία ενός ταξινομητή που θα προβλέπει την ύπαρξη ή μη της πνευμονίας δίνοντας σαν δεδομένο μια εικόνα ακτινογραφίας θώρακος από κάποιον ασθενή. Ο ταξινομητής αυτός λόγω του τύπου των δεδομένων κάνει χρήση Συνελικτικών Νευρωνικών Δικτύων (CNN). Για την επίλυση αυτού του ζητήματος, εφαρμόζονται διάφορες τεχνικές στην παρούσα εργασία. Ωστόσο, έμφαση δίνεται στη χρήση των Conditional GANs που αποτελούν μια καινοτόμα τεχνική επαύξησης δεδομένων τα τελευταία χρόνια, καθώς μπορούν να συνθέσουν νέα δείγματα που δεν έχουν υπάρξει ποτέ χρησιμοποιώντας τα υπάρχοντα δεδομένα.

## **1.2 Η Ανισορροπία των κλάσεων**

Πολλοί παραδοσιακοί αλγόριθμοι στη μηχανική μάθηση και προβλήματα εξόρυξης δεδομένων υποθέτουν ότι οι κλάσεις-στόχοι μοιράζονται εκ των προτέρων παρόμοιες πιθανότητες. Ωστόσο, σε πολλές εφαρμογές του

πραγματικού κόσμου, όπως οι ανίχνευση πετρελαιοκηλίδων, η ανίχνευση εισβολής σε δίκτυο και η ανίχνευση απάτης, η υπόθεση αυτή παύει να ισχύει. Σε τέτοια προβλήματα, σχεδόν όλα τα παραδείγματα χαρακτηρίζονται ως μία κλάση, ενώ πολύ λιγότερα παραδείγματα χαρακτηρίζονται ως η άλλη κλάση-συνήθως η πιο σημαντική. Αυτή η κατάσταση είναι γνωστή ως πρόβλημα ανισορροπίας κλάσεων. Σε αυτή την περίπτωση, οι συνήθεις ταξινομητές τείνουν να κατακλύζονται από την πλειοψηφία και αγνοούν τη μειοψηφική κλάση.

Η σημασία του συγκεκριμένου προβλήματος αναδεικνυόταν με αυξανόμενο ρυθμό μέσα στα χρόνια, καθώς όλο και περισσότεροι ερευνητές συνειδητοποιούσαν ότι οδηγεί σε μη βέλτιστες αποδόσεις ταξινόμησης και ότι οι περισσότεροι αλγόριθμοι συμπεριφέρονται λανθασμένα όταν τα σύνολα δεδομένων είναι ιδιαίτερα ανισόρροπα. Το πρόβλημα της ανισορροπίας κλάσεων αποτέλεσε ένα μείζον θέμα στη μηχανική μάθηση τα τελευταία χρόνια. Ακόμη, έχει αναγνωριστεί ότι μπορεί να εμφανιστεί σε πολλούς τομείς εφαρμογών, όπως ο εντοπισμός αναξιόπιστων πελατών τηλεπικοινωνιών, η ανίχνευση πετρελαιοκηλίδων σε δορυφορικές εικόνες από ραντάρ, η ταξινόμηση κειμένων, η διαχείριση κινδύνων, καθώς και η ιατρική διάγνωση (π.χ. σπάνιες ασθένειες και μεταλλάξεις σπάνιων γονιδίων). Από την οπτική των εφαρμογών, η φύση της ανισορροπίας χωρίζεται σε δύο περιπτώσεις: Τα δεδομένα είναι φυσικά ανισόρροπα (π.χ. απάτες με πιστωτικές κάρτες και σπάνιες ασθένειες) ή δεν είναι φυσικά ανισόρροπα, αλλά είναι πολύ δαπανηρή η απόκτηση δεδομένων της μειονοτικής κλάσης για μάθηση.[7]

### **1.3 Η σημαντικότητα της αντιμετώπισης της ανισορροπίας στην ιατρική διάγνωση**

Η ανισορροπία των κλάσεων, όπου μια κλάση υπερτερεί σημαντικά έναντι των άλλων, θέτει αρκετές κρίσιμες προκλήσεις που, αν δεν αντιμετωπιστούν, μπορούν να υποβαθμίσουν σημαντικά την αποτελεσματικότητα των μοντέλων μηχανικής μάθησης.

Στο πεδίο της ιατρικής διάγνωσης, όπου η έγκαιρη ανίχνευση και η αμερόληπτη λήψη αποφάσεων είναι υψίστης σημασίας, η αντιμετώπιση της ανισορροπίας των κλάσεων αναδεικνύεται σε κρίσιμο βήμα προς την αξιοποίηση του πλήρους δυναμικού της μηχανικής μάθησης.

Μία από τις πρωταρχικές ανησυχίες που σχετίζονται με την ανισορροπία των κλάσεων είναι η εισαγωγή μεροληψίας και η διάβρωση της δικαιοσύνης στις προβλέψεις των μοντέλων. Στην υγειονομική περίθαλψη, οι συνέπειες των μεροληπτικών μοντέλων είναι ολέθριες, καθώς μπορεί να ευνοούν την πλειοψηφούσα κλάση, παραγκωνίζοντας τη μειοψηφούσα κλάση - συχνά ασθενείς με συγκεκριμένες ιατρικές καταστάσεις. Διορθώνοντας αυτή την ανισορροπία, ανοίγουμε τον δρόμο για δίκαιες και αμερόληπτες εκτιμήσεις, διασφαλίζοντας ότι οι ανάγκες όλων των ασθενών, ανεξάρτητα από την κατάσταση της υγείας τους, λαμβάνονται υπόψη.

Η έγκαιρη ανίχνευση ασθενειών αποτελεί ακρογωνιαίο λίθο της αποτελεσματικής υγειονομικής περίθαλψης και η ανισορροπία των κλάσεων μπορεί να αποτελέσει σημαντικό εμπόδιο για την επίτευξη αυτού του στόχου. Σε περιπτώσεις όπου πρόκειται για σπάνιες παθήσεις, ένα ανισόρροπο σύνολο δεδομένων μπορεί να εμποδίσει την ικανότητα του μοντέλου μηχανικής μάθησης να εντοπίζει λεπτά μοτίβα που σχετίζονται με αυτές τις ασθένειες. Η διόρθωση της ανισορροπίας των κλάσεων γίνεται συνώνυμη με την ενίσχυση της ευαισθησίας του μοντέλου, επιτρέποντάς του να εντοπίζει και να επισημαίνει πιθανά προβλήματα υγείας σε πρώιμο στάδιο, διευκολύνοντας έτσι τις έγκαιρες παρεμβάσεις και βελτιώνοντας τα αποτελέσματα των ασθενών.

Επίσης η αποδοτική κατανομή των πόρων είναι μια άλλη κρίσιμη πτυχή του τομέα της υγείας που συνδέεται στενά με την αντιμετώπιση της ανισορροπίας. Με τον ακριβή εντοπισμό περιπτώσεων ιατρικών παθήσεων, τα μοντέλα μηχανικής μάθησης συμβάλλουν στην πιο συνετή κατανομή των περιορισμένων πόρων, διασφαλίζοντας ότι όσοι έχουν ανάγκη λαμβάνουν άμεση προσοχή. Ένα μοντέλο που στρέφεται προς την πλειοψηφούσα κλάση μπορεί ακούσια να καταναείμει λανθασμένα τους πόρους, παραμελώντας κρίσιμες περιπτώσεις και θέτοντας σε κίνδυνο τη συνολική αποτελεσματικότητα των συστημάτων υγειονομικής περίθαλψης.

Στην επιδίωξη μιας ασθενοκεντρικής προσέγγισης της υγειονομικής περίθαλψης, η αντιμετώπιση της ανισορροπίας αναδεικνύεται σε βασικό άξονα. Η προσέγγιση αυτή υπογραμμίζει τη σημασία των εξατομικευμένων και στοχευμένων ιατρικών παρεμβάσεων, αναγνωρίζοντας τις ποικίλες εκδηλώσεις των ασθενειών σε μεμονωμένους ασθενείς. Ένα ισορροπημένο μοντέλο, ικανό να αναγνωρίζει ένα ευρύ φάσμα προτύπων, ευθυγραμμίζεται άψογα με το ήθος

της ασθενοκεντρικής φροντίδας, προωθώντας ένα περιβάλλον όπου η υγειονομική περίθαλψη προσαρμόζεται στις μοναδικές ανάγκες του κάθε ατόμου.

Η μείωση των ψευδώς θετικών και αρνητικών αποτελεσμάτων είναι μια επίμονη πρόκληση στην ιατρική διάγνωση, και η ανισορροπία των κλάσεων επιδεινώνει αυτά τα ζητήματα. Ένα μοντέλο που επηρεάζεται από ανισόρροπα δεδομένα μπορεί να εμφανίζει μια τάση να είναι υπερβολικά συντηρητικό, οδηγώντας σε αυξημένο αριθμό ψευδώς αρνητικών, ή υπερβολικά αισιόδοξο, με αποτέλεσμα υψηλότερη συχνότητα ψευδώς θετικών αποτελεσμάτων. Η αντιμετώπιση της ανισορροπίας των κλάσεων συμβάλλει καθοριστικά στον μετριασμό αυτών των σφαλμάτων, ενισχύοντας τελικά την αξιοπιστία των διαγνωστικών προβλέψεων.

Επιπλέον, καθώς τα συστήματα υγείας υπόκεινται σε αυστηρά ρυθμιστικά πρότυπα, η αντιμετώπιση της ανισορροπίας κλάσεων καθίσταται απαραίτητη προϋπόθεση για τη συμμόρφωση. Αυτοί οι κανονισμοί απαιτούν δικαιοσύνη, ακρίβεια και διαφάνεια στα διαγνωστικά εργαλεία, διασφαλίζοντας ότι τα μοντέλα μηχανικής μάθησης πληρούν τα απαραίτητα πρότυπα για την ανάπτυξη σε κλινικές ρυθμίσεις. Με την υιοθέτηση μιας ισορροπημένης προσέγγισης, όχι μόνο τηρούμε τις κανονιστικές απαιτήσεις, αλλά και διασφαλίζουμε την ποιότητα και καταλληλότητα της παρεχόμενης περίθαλψης και εμπνέουμε εμπιστοσύνη στη χρήση αυτών των τεχνολογιών αιχμής στην επιστημονική και επαγγελματική κοινότητα του χώρου της υγείας.

Εν κατακλείδι, η σημασία της αντιμετώπισης του προβλήματος της ανισορροπίας στην ιατρική διάγνωση εκτείνεται πέρα από το πεδίο των τεχνικών εκτιμήσεων. Με τη διόρθωση αυτής της ανισορροπίας, δίνουμε τη δυνατότητα στα μοντέλα μηχανικής μάθησης να συμβάλλουν ουσιαστικά στη λήψη ιατρικών αποφάσεων, προωθώντας μια πιο περιεκτική και ασθενοκεντρική προσέγγιση.

## **1.4 Δομή της εργασίας**

Στο ανωτέρω πλαίσιο, η συγκεκριμένη μελέτη εστιάζει στην αντιμετώπιση του προβλήματος της ανισορροπίας των κλάσεων στη διάγνωση της πνευμονίας, έχοντας ως κύρια τεχνική αντιμετώπισης τη χρήση των Παραγωγικών Αντιπαλικών Δικτύων (Generative Adversarial Networks ή GANs). Μελετά τις διάφορες αρχιτεκτονικές των GANs με σκοπό τη ρεαλιστικότερη παραγωγή νέων συνθετικών δειγμάτων τα οποία στη συνέχεια θα προστεθούν στο αρχικό dataset

που εμφανίζει το πρόβλημα και με τη βοήθεια προπαιδευμένων μοντέλων θα ερευνήσει την επιρροή που έχουν τα νέα συνθετικά δείγματα που παράχθηκαν μέσω GAN στον αρχικό ταξινομητή, αλλά και πως επηρεάζεται η απόδοση του ανάλογα με ποιο προεκπαιδευμένο μοντέλο έχει χρησιμοποιηθεί. Σε αυτήν τη μελέτη τα GANs χρησιμοποιούνται ως ένα μέσο υπερδειγματοληψίας όπως π.χ. ο αλγόριθμος SMOTE με σκοπό την επαύξηση των δεδομένων της μειωτικής κλάσης και παράλληλα τη βελτίωση της απόδοσης του δυαδικού ταξινομητή ως προς την κλάση που υστερεί.

Οι εργασίες οι οποίες πραγματοποιήθηκαν και τα αποτελέσματα αυτών παρουσιάζονται στις ακόλουθες ενότητες.

Το *Κεφάλαιο 1* αποτελεί μια εισαγωγή περιγράφοντας το πρόβλημα που μελετήθηκε, καθώς και αναλύοντας τους πιο διαδεδομένους τρόπους και τεχνικές αντιμετώπισης του προβλήματος.

Στο *Κεφάλαιο 2* παρουσιάζεται η βιβλιογραφική επισκόπηση της υφιστάμενης βιβλιογραφίας στο εξωτερικό, καθώς και το θεωρητικό υπόβαθρο της μηχανικής Μάθησης και της βαθιάς Μάθησης. Ακόμη, γίνεται εισαγωγή στα Generative Adversarial Networks (GANs), καθώς και στο θεωρητικό υπόβαθρο πίσω από αυτά περιγράφοντας αναλυτικά τη δομή τους, τον τρόπο λειτουργία τους, τους περιορισμούς, καθώς και τις αρχιτεκτονικές που χρησιμοποιήθηκαν στα πλαίσια της συγκεκριμένης διπλωματικής.

Στο *Κεφάλαιο 3* περιγράφεται αναλυτικά η μεθοδολογία που ακολουθήθηκε για την επίλυση του προβλήματος αλλά και τα βήματα βελτιστοποίησης των αποτελεσμάτων.

Στο *Κεφάλαιο 4* αναλύονται οι τρόποι διεξαγωγής όλων το πειραμάτων που διεξήχθησαν στην εργασία και όλα τα τεχνικά χαρακτηριστικά που αφορούν τους πόρους που χρησιμοποιήθηκαν, καθώς και τα αποτελέσματα των πειραμάτων.

Το *Κεφάλαιο 5* εστιάζει στην ευρύτερη συζήτηση των αποτελεσμάτων, ενώ στο *Κεφάλαιο 6* παρουσιάζονται τα συνολικά συμπεράσματα της εργασίας, καθώς και αναφορές και ιδέες για μελλοντικές ενέργειες που πιθανόν θα βελτιώσουν τα σημερινά αποτελέσματα ακόμα περισσότερο.

## **2 ΚΕΦΑΛΑΙΟ 2: ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΕΠΙΣΚΟΠΗΣΗ**

### **2.1 Βιβλιογραφική Επισκόπηση**

Η πνευμονία, μια από τις κύριες αιτίες νοσηρότητας και θνησιμότητας παγκοσμίως, απαιτεί έγκαιρη και ακριβή διάγνωση για αποτελεσματική αντιμετώπιση. Η ιατρική απεικόνιση, ιδίως οι ακτινογραφίες θώρακος, αποτελεί ακρογωνιαίο λίθο στη διάγνωση της πνευμονίας, παρέχοντας στους κλινικούς ιατρούς σημαντικές πληροφορίες για τις αλλοιώσεις που παρατηρούνται στους πνεύμονες και σχετίζονται άμεσα με την εμφάνιση της πνευμονίας. Στο πλαίσιο αυτό, η συμπερίληψη της αυτοματοποιημένης διάγνωσης με τη χρήση υπολογιστικών τεχνικών στη συνολική προσπάθεια της ιατρικής κοινότητας για έγκαιρη διάγνωση της πνευμονίας προσφέρει μια πολλά υποσχόμενη οδό ενισχύοντας τις ικανότητες των επαγγελματιών υγείας, συμβάλλοντας στην απλούστευση των διαδικασιών διάγνωσης και στον περιορισμό των διαγνωστικών σφαλμάτων.[8]

Αναλυτικότερα, η ιατρική απεικόνιση είναι ζωτικής σημασίας για τη διάγνωση της πνευμονίας, με διαφορετικές μεθόδους που επιλέγονται βάσει συγκεκριμένων και εξατομικευμένων απαιτήσεων ανά περιστατικό. Η ακτινογραφία θώρακος αποτελεί τη συνηθέστερη μέθοδο ιατρικής απεικόνισης στην περίπτωση της υπό εξέτασης νόσου, καθώς επιτρέπει την καταγραφή λεπτομερούς εικόνας των πνευμόνων και των ανωμαλιών των εσωτερικών οργάνων του θωρακικού τοιχώματος, αποκαλύπτοντας διηθήσεις στους πνεύμονες, που υποδηλώνουν φλεγμονή. Ωστόσο, έχει περιορισμούς, καθώς στερείται λεπτομέρειας και ικανότητας διαφοροποίησης των αιτιών. [9] Η αξονική τομογραφία θώρακος (CT) παρέχει λεπτομερείς εικόνες διατομής των πνευμόνων, χρήσιμες όταν τα αποτελέσματα της ακτινογραφίας δεν είναι σαφή ή υπάρχουν υποψίες επιπλοκών, παρά το γεγονός ότι συνεπάγεται υψηλότερη δόση ακτινοβολίας.[10] Ακόμη, το υπερηχογράφημα θώρακα έχει την ικανότητα φορητότητας και μη έκθεσης σε ακτινοβολία, με αποτέλεσμα να είναι πολύτιμο για την αξιολόγηση ενός κατακεκλιμένου ασθενούς και την καθοδήγηση διαδικασιών, όπως η αναρρόφηση πνευμόνων. Ωστόσο, έχει χαμηλότερη ευαισθησία σε σύγκριση με την ακτινογραφία ή την αξονική τομογραφία και ενδέχεται να μην διεισδύει βαθιά στους πνεύμονες. [11] Η μαγνητική τομογραφία θώρακος (MRI), αν και χρησιμοποιείται σπάνια λόγω κόστους και

διαθεσιμότητας, αποδεικνύεται πολύ χρήσιμη σε πολύπλοκες περιπτώσεις και στη διαφοροποίηση της πνευμονίας από άλλες πνευμονικές παθήσεις, προσφέροντας το πλεονέκτημα της μη έκθεσης σε ακτινοβολία. [12]

Η επιλογή της μεθόδου απεικόνισης εξαρτάται από διάφορους παράγοντες, όπως η κλινική εικόνα, η σοβαρότητα της κατάστασης, τα ιδιαίτερα χαρακτηριστικά του ασθενούς και η διαθεσιμότητα πόρων. Ο συνδυασμός της απεικόνισης με την καλλιέργεια πτυέλων και τις εξετάσεις αίματος βελτιώνει την ακρίβεια. Οι αναδυόμενες τεχνολογίες, όπως η ανάλυση εικόνας με τεχνητή νοημοσύνη, στοχεύουν στη βελτίωση της αποτελεσματικότητας και της ακρίβειας της διάγνωσης της πνευμονίας.

Με σκοπό την υποστήριξη και επιτάχυνση της διαγνωστικής διαδικασίας, η διάγνωση με τη βοήθεια υπολογιστή (CAD) αποκτά ολοένα και μεγαλύτερη σημασία στην ανάλυση ιατρικών εικόνων για τη διάγνωση της πνευμονίας σε διάφορες μεθόδους απεικόνισης. Στις ακτινογραφίες θώρακος, οι αλγόριθμοι βαθιάς μάθησης, εκπαιδευμένοι σε εκτεταμένα σύνολα δεδομένων, παρουσιάζουν υψηλή ακρίβεια στην ανίχνευση πνευμονίας. Ένα CAD σύστημα υποστηρίζει τους ακτινολόγους με την επισήμανση ύποπτων περιοχών, τη βελτίωση της ευαισθησίας και τη μείωση των λανθασμένων διαγνώσεων. Ωστόσο, εξακολουθούν να υπάρχουν ανησυχίες σχετικά με τον βαθμό εμπιστοσύνης, αλλά και την απροθυμία των επαγγελματιών υγείας να ενσωματώσουν τέτοιες τεχνικές στην κλινική πράξη. [13]

Ομοίως, ένα CAD σύστημα σε αξονικές τομογραφίες θώρακος βοηθά στην ανάλυση της πνευμονίας, στη διαφοροποίηση των τύπων της, στη μείωση του χρόνου ερμηνείας και στη βελτίωση της ροής εργασίας των ακτινολόγων. Τα προηγμένα συστήματα μπορούν να ποσοτικοποιήσουν τα οπτικά πρότυπα σε ιατρικές απεικονίσεις των πνευμόνων, βοηθώντας τις αποφάσεις θεραπείας. [14]

Στους υπερήχους θώρακος, αναπτύσσονται συστήματα CAD για την αυτόματη ανίχνευση των χαρακτηριστικών της πνευμονίας, που αποδεικνύονται πολύτιμα σε περιβάλλον περιορισμένων πόρων, αλλά και σε ειδικές περιπτώσεις ασθενών. [15] Χαρακτηριστικό παράδειγμα αποτέλεσε μία έρευνα στην οποία οι ερευνητές ανέπτυξαν ένα σύστημα CAD χρησιμοποιώντας τεχνικές βαθιάς μάθησης για να βοηθήσουν τους ακτινολόγους στη διάγνωση του συνδρόμου του ιού SARS-CoV-2 μέσω της υπερηχογραφίας των πνευμόνων (Lung Ultrasound Segmentation - LUS). Αναλυτικότερα, χρησιμοποίησαν μια προσέγγιση μάθησης μεταφοράς

(transfer learning) με ένα υπολειμματικό δίκτυο (ResNet) για την εξαγωγή χαρακτηριστικών από εικόνες LUS και τη διάκριση μεταξύ διαφορετικών καταστάσεων. Μέσω ολοκληρωμένης αξιολόγησης και πραγματοποίησης δοκιμών με 6 ακτινολόγους διαφορετικών επιπέδων εμπειρίας, διαπίστωσαν ότι το σύστημα CAD βελτίωσε σημαντικά τη διαγνωστική ακρίβεια για το σύνδρομο του ιού SARS-CoV-2, τόσο σε ισορροπημένα όσο και σε μη ισορροπημένα σύνολα δεδομένων. Οι υποσχόμενες επιδόσεις του συστήματος υποδηλώνουν ότι θα μπορούσε να βοηθήσει σε γρήγορες και προσιτές μεθόδους διαλογής για πνευμονικές ασθένειες, υποδεικνύοντας δυνατότητες πρακτικής εφαρμογής σε κλινικό περιβάλλον. [16] Τέλος, το CAD για τη διάγνωση της πνευμονίας στη μαγνητική τομογραφία είναι λιγότερο διαδεδομένο λόγω περιορισμένης χρήσης. Η ερευνητική κοινότητα αναζητά, ωστόσο, μεθόδους για την αυτοματοποιημένη διάγνωση της πνευμονίας με τη χρήση μαγνητικής τομογραφίας. [17]

Αναλυτικότερα, και όσον αφορά τις μεθόδους CAD για τη διάγνωση της πνευμονίας με τη χρήση εικόνων ακτινογραφίας θώρακος υπάρχουν οι παραδοσιακές τεχνικές επεξεργασίας εικόνας, οι τεχνικές μηχανικής μάθησης, καθώς και υβριδικές προσεγγίσεις.

Στις παραδοσιακές τεχνικές έχουμε την ανάλυση υφής, η οποία μετράει τη χωρική κατανομή των επιπέδων έντασης στην εικόνα προσπαθώντας να εντοπίσει ανωμαλίες πνευμονικών υφών που υποδηλώνουν τη νόσο. Για παράδειγμα, σε ένα άρθρο των Ortiz-Toro et al [18] ενσωματώθηκαν ορισμένες παραδοσιακές τεχνικές και διερευνήθηκε η αποτελεσματικότητα τριών μεθόδων χαρακτηρισμού της υφής της εικόνας -ακτινοδιαγνωστικής, fractal διάστασης και τη μέθοδο superpixel-based histogram- ως πιθανών βιοδεικτών για την εκπαίδευση μοντέλων Τεχνητής Νοημοσύνης για την ανίχνευση πνευμονίας σε εικόνες ακτινογραφίας θώρακος. Χρησιμοποιώντας τρεις αλγορίθμους τεχνητής νοημοσύνης K-Nearest Neighbors, Support Vector Machine και Random Forest - η μελέτη αξιολογεί την απόδοση του μοντέλου σε δύο σύνολα δεδομένων εικόνων ανοικτής πρόσβασης, συμπεριλαμβανομένου ενός συνόλου δεδομένων παιδιατρικών ακτινογραφιών θώρακος και ενός συνόλου δεδομένων που προέρχεται από ένα αποθετήριο που εστιάζει κυρίως στο COVID-19. Τα αποτελέσματα κατέδειξαν υψηλή ακρίβεια και ευαισθησία και για τις τρεις μεθόδους, υποδεικνύοντας την εγκυρότητά τους ως αξιόπιστα και εύκολα εφαρμόσιμα εργαλεία αυτόματης διάγνωσης της πνευμονίας. [18]



Στη συνέχεια, υπάρχει η τμηματοποίηση στην οποία οι αλγόριθμοι τμηματοποιούν αυτόματα την περιοχή του πνεύμονα και αναλύουν το σχήμα, το μέγεθος και τα χαρακτηριστικά έντασης για την ανίχνευση της πνευμονίας. Τέλος, η ανίχνευση ακμών ασχολείται με τον εντοπισμό ακμών που αντιστοιχούν σε ανατομικές δομές και ανάλυση των ιδιοτήτων τους για την ανίχνευση ορίων σύνδεσης (Connectivity Thresholds). Τα Connectivity Thresholds αποτελούν κατώφλια που καθορίζουν πόσο μακριά δύο pixels πρέπει να είναι ώστε να θεωρούνται συνδεδεμένα ή μέρος της ίδιας ακμής.[19], [20]

Αναφορικά με τις τεχνικές μηχανικής μάθησης, οι Μηχανές Διανυσμάτων Στήριξης (Support Vector Machines-SVM) ταξινομούν τις εικόνες ακτινογραφίας θώρακος ως φυσιολογικές ή ως παθογενείς βασιζόμενα στα χαρακτηριστικά που εξάγονται, όπως η υφή, οι ακμές και οι μετρήσεις έντασης. Σε μία έρευνα των Nagashree και Mahanand [21] χρησιμοποιήθηκε μία μέθοδος ενός SVM ταξινομητή για τη διάγνωση της πνευμονίας με τη χρήση εικόνων ακτινογραφίας. Ο ταξινομητής εκπαιδεύτηκε με χαρακτηριστικά Haar, το σχήμα και χαρακτηριστικά υφής. Όσον αφορά τα ευρήματα της μελέτης, η επίδοση του SVM συγκρίθηκε με άλλους αλγορίθμους ταξινόμησης, ήτοι δέντρα απόφασης, Bayes και K-Nearest Neighbors και προέκυψε ότι το SVM έχει καλύτερες συγκριτικές επιδόσεις στη διάγνωση της πνευμονίας. [21]

Επίσης αντί των SVM μπορούν να χρησιμοποιηθούν και μοντέλα Random Forest, καθώς μπορούν να εκπαιδευτούν χρησιμοποιώντας χαρακτηριστικά που εξάγονται από τις εικόνες, καταφέροντας λόγω της φύσης τους να βελτιώσουν την ευστάθεια και την ακρίβειά τους. Τα μοντέλα Random Forest αποτελούνται από πολλά δέντρα αποφάσεων, το καθένα από τα οποία έχει αναπτυχθεί με τη χρήση τυχαία επιλεγμένων δεδομένων. Σε μία μελέτη των Al Mamlook et al [22], αναπτύχθηκε ένα μοντέλο για την ανίχνευση της πνευμονίας με βάση εικόνες ακτινογραφίας θώρακος. Το μοντέλο χρησιμοποίησε επτά τεχνικές μηχανικής μάθησης και συνελκτικά νευρωνικά δίκτυα (CNN): Δέντρα απόφασης, Random Forest, K-nearest neighbor, AdaBoost, Gradient Boost, XGBoost. Η έρευνα στόχευε στην ταξινόμηση εικόνων ακτινογραφίας θώρακος ως φυσιολογικές ή μη φυσιολογικές. Η προτεινόμενη προσέγγιση βαθιάς μάθησης, η οποία εκπαιδεύτηκε με προεπεξεργασμένες εικόνες, πέτυχε ακρίβεια 98,46% στον εντοπισμό περιπτώσεων πνευμονίας. Τα ευρήματα αυτά υπογραμμίζουν την

αποτελεσματικότητα της βαθιάς μάθησης, ιδίως των CNN, στην απλοποίηση και τη βελτίωση της ανίχνευσης πνευμονίας από εικόνες ακτίνων X θώρακος [22].

Ακόμη, τα CNN έχουν επιδείξει αξιοσημείωτη επιτυχία στη διάγνωση της πνευμονίας χρησιμοποιώντας ακτινογραφίες θώρακα. Αυτά τα δίκτυα μπορούν να μάθουν αυτόματα τα σχετικά χαρακτηριστικά από τα ακατέργαστα δεδομένα εικονοστοιχείων χωρίς αυστηρή επεξεργασία των χαρακτηριστικών. Με την εκπαίδευση σε μεγάλα σύνολα δεδομένων από επισημειωμένες ακτινογραφίες, τα CNN μπορούν να συλλάβουν περίπλοκα μοτίβα και σχέσεις στις εικόνες, επιτρέποντας υψηλή ακρίβεια στην ανίχνευση και ταξινόμηση περιπτώσεων πνευμονίας για αυτό και αποτελούν μια από τις καλύτερες τεχνικές διάγνωσης της πνευμονίας και χρησιμοποιούνται ευρέως στη βιβλιογραφία. [19] [23], [24]

Οι παραδοσιακές τεχνικές επεξεργασίας εικόνας διακρίνονται από τις μεθόδους μηχανικής μάθησης καθώς στηρίζονται σε στατιστικούς κανόνες ή μηχανισμούς επεξεργασίας σήματος, ενώ οι αλγόριθμοι μηχανικής μάθησης βασίζονται στην εκπαίδευση από δεδομένα. Ωστόσο υπάρχουν και οι υβριδικές προσεγγίσεις που συνδυάζουν στοιχεία από και από τις δύο προσεγγίσεις, εκμεταλλευόμενες τα πλεονεκτήματα της ερμηνευσιμότητας των παραδοσιακών τεχνικών και τις δυνατότητες μάθησης των αλγορίθμων μηχανικής μάθησης με σκοπό τη βελτίωση της απόδοσης και της ερμηνευσιμότητας του μοντέλου. Ένα ακόμη παράδειγμα υβριδικού μοντέλου αποτέλεσε η μελέτη των Nandi και Mulimani [25], οι οποίοι παρουσίασαν ένα υβριδικό μοντέλο βαθιάς μάθησης για την ανίχνευση του COVID-19 και της πνευμονίας από ακτινογραφίες θώρακος, ειδικά σχεδιασμένο για χαμηλού κόστους προσωπικούς ψηφιακούς βοηθούς (PDA). Συνδυάζοντας το ResNet50 και το MobileNet, το μοντέλο προσφέρει αποτελεσματική ανίχνευση του COVID-19 και της πνευμονίας ακόμη και με περιορισμένους υπολογιστικούς πόρους. Οι αξιολογήσεις σε δύο σύνολα δεδομένων έδειξαν πολλά υποσχόμενη ακρίβεια. Ειδικότερα, η σημασία αυτού του μοντέλου επεκτείνεται στην ανίχνευση πνευμονίας παράλληλα με το COVID-19, καλύπτοντας κρίσιμες διαγνωστικές ανάγκες. [25]

Όπως αναλύθηκε, οι ανωτέρω μέθοδοι διάγνωσης με τη βοήθεια υπολογιστή (CAD) προσφέρουν πολλές δυνατότητες για τη διάγνωση της πνευμονίας σε ακτινογραφίες θώρακα. Ωστόσο, είναι κρίσιμο να αναγνωρίσουμε την ύπαρξη

περιορισμών και ενδεχόμενων ζητημάτων που ανακύπτουν από τη χρήση τους [19].

Μια σημαντική ανησυχία είναι η ακρίβεια και η δυνατότητα γενίκευσης των συστημάτων CAD. [19] Αυτά τα συστήματα βασίζονται σε μεγάλο βαθμό στην ποιότητα και την ποικιλομορφία των δεδομένων εκπαίδευσης. Χαρακτηριστικό παράδειγμα αποτελεί το γεγονός ότι τα περισσότερα μοντέλα έχουν εκπαιδευτεί σε δεδομένα ενηλίκων γεγονός που δυσκολεύει τη γενίκευσή τους σε παιδιατρικό πληθυσμό. Ακόμη, άλλα στοιχεία που επηρεάζουν τη μεροληψία των μοντέλων σχετίζονται με το είδος του κέντρου στο οποίο συλλέχθηκαν τα δεδομένα (π.χ. νοσοκομείο, καθώς και στον βαθμό εξειδίκευσης του κέντρου ενδεικτικά σε πνευμονικά περιστατικά επηρεάζοντας την ικανότητά τους να γενικεύουν σε πραγματικούς πληθυσμούς ασθενών. Επιπλέον, παράγοντες, όπως η κακή ανάλυση της εικόνας ή οι επικαλυπτόμενες δομές μπορούν να επηρεάσουν την ευαισθησία, επηρεάζοντας έτσι τη συνολική ακρίβεια της διάγνωσης CAD. Επιπλέον, η μεταβλητότητα των παραδειγμάτων πνευμονίας, ιδίως σε πολύπλοκες περιπτώσεις, μπορεί να αποτελέσει πρόκληση για τα συστήματα CAD, απαιτώντας προσεκτική ερμηνεία από τους ακτινολόγους. [19], [23], [26]

Επίσης κρίσιμα ζητήματα που αφορούν στα CAD είναι η ερμηνευσιμότητα και η επεξηγηματικότητα. Η ερμηνευσιμότητα αναφέρεται στο πώς μπορούμε να κατανοήσουμε και να ερμηνεύσουμε τα αποτελέσματα ή τις αποφάσεις που λαμβάνει ένα μοντέλο. Η επεξηγηματικότητα σχετίζεται με τον τρόπο με τον οποίο μπορούμε να παράγουμε εξηγήσεις για τις αποφάσεις που λαμβάνει ένα σύστημα. Ενώ η ερμηνευσιμότητα είναι περισσότερο συνδεδεμένη με την ικανότητα κατανόησης, η επεξηγηματικότητα αναφέρεται στη δυνατότητα παραγωγής σαφών εξηγήσεων για τον λόγο για τον οποίο ένα συγκεκριμένο αποτέλεσμα προκύπτει. Τα μοντέλα βαθιάς μάθησης, που συχνά αποτελούν τον πυρήνα των συστημάτων CAD, μπορούν να θεωρηθούν «μαύρα κουτιά», χωρίς διαφάνεια ως προς τον τρόπο με τον οποίο καταλήγουν σε διαγνώσεις. Αυτή η αδιαφάνεια μπορεί να εμποδίσει την εμπιστοσύνη και την αποδοχή μεταξύ των επαγγελματιών υγείας. [27]

Συνεχίζοντας, μια από τις μεγαλύτερες προκλήσεις που μπορεί να έρθουν αντιμέτωπα τα συστήματα CAD είναι το πρόβλημα της ανισορροπίας των κλάσεων. Με βάση το άρθρο των Ali et al (2013) [28], η ανισοκατανομή κλάσεων

σε σύνολα δεδομένων συμβαίνει όταν μια κλάση, συχνά αυτή που έχει το μεγαλύτερο ενδιαφέρον, δηλαδή η θετική ή μειοψηφική κλάση, δεν αντιπροσωπεύεται επαρκώς. Πρακτικά, αυτό σημαίνει ότι ο αριθμός των παραδειγμάτων της θετικής κλάσης (μειονότητα) είναι πολύ μικρότερος από τον αριθμό των παραδειγμάτων της αρνητικής κλάσης (πλειοψηφία). Επίσης, το άρθρο αναφέρει ότι αν τα σπάνια παραδείγματα δεν εμφανίζονται συχνά, το πιθανότερο είναι να προβλέπονται ως σπάνιες περιπτώσεις, να μην εντοπίζονται, ή να θεωρούνται ως θόρυβος ή ακραίες τιμές. Τα στοιχεία αυτά οδηγούν σε περισσότερες λανθασμένες ταξινομήσεις της θετικής κλάσης (μειοψηφία) σε σύγκριση με την αρνητική κλάση (πλειοψηφία). Συνήθως, η μειοψηφική κλάση έχει μεγαλύτερο ενδιαφέρον και σημασία, απαιτώντας έτσι άμεση αναγνώριση του προβλήματος.

Για παράδειγμα, κατά τη διάγνωση μιας σπάνιας ασθένειας, είναι καθοριστικός ο εντοπισμός μιας τέτοιας σπάνιας ιατρικής κατάστασης μεταξύ των φυσιολογικών πληθυσμών. Ένα άτομο που λαμβάνει λανθασμένη διάγνωση, δηλαδή λανθασμένη εντύπωση για την κατάσταση της υγείας του, μπορεί να υποστεί σοβαρή ψυχολογική πίεση και άγχος, καθώς η σκέψη ότι πάσχει από μία σοβαρή σπάνια ασθένεια, ενώ στην πραγματικότητα είναι υγιής, μπορεί να προκαλέσει σημαντικό άγχος και ανησυχία. Αντίθετα, εάν ο ασθενής που λαμβάνει λανθασμένη διάγνωση θεωρεί ότι είναι υγιής και δεν χρειάζεται θεραπεία, αυτό μπορεί να οδηγήσει σε ανεπάρκεια ή καθυστερημένη θεραπεία, με πιθανές επιπτώσεις στην υγεία του. και ακόμη και να αλλάξει την πορεία των διαθέσιμων θεραπειών και φαρμάκων. Συνεπώς, είναι καίριας σημασίας ένα μοντέλο ταξινόμησης να είναι σε θέση να ταξινομεί σωστά δεδομένα από τη μειοψηφική κλάση.

Καθώς το πρόβλημα της ανισορροπίας των κλάσεων απασχολούσε ολοένα και περισσότερο την ερευνητική κοινότητα, αρκετοί ερευνητές επιχείρησαν να αναπτύξουν τεχνικές αντιμετώπισης του προβλήματος. Μια πολύ συνηθισμένη τεχνική είναι η επαύξηση των δεδομένων (Data augmentation) κατά την οποία εκτελείται μία διαδικασία παραγωγής τεχνητών εικόνων της κλάσης που παρουσιάζει τη μειονότητα μέσω τεχνικών, όπως η αναστροφή, η περιστροφή και η προσθήκη θορύβου σε υπάρχουσες εικόνες που ανήκουν στη συγκεκριμένη κλάση με σκοπό να αυξήσει την ποικιλομορφία των δεδομένων.

Στο άρθρο [29] του Farukh Hashmi et al (2020) εφαρμόστηκε η συγκεκριμένη τεχνική, καθώς το dataset με τις ακτινογραφίες θώρακος παρουσιάζει ανισορροπία ως προς την υγιή κλάση (Normal) με αριθμό δειγμάτων 1283 εικόνες έναντι 3873 εικόνων με πνευμονία (Pneumonia). Αφού έγινε επαύξηση των εικόνων της κλάσης Normal δύο φορές, η τελική κατανομή του dataset ήταν 3399 εικόνες από την κλάση Normal και 3623 εικόνες από την κλάση πνευμονίας (Pneumonia). Στη συνέχεια, εφαρμοστήκαν τεχνικές transfer learning για τη διάγνωση της πνευμονίας αποδεικνύοντας ότι η επαύξηση δεδομένων οδήγησε σε Accuracy μεγαλύτερο του 96% και μέγιστη τιμή Απώλειας 0.101 για κάθε αρχιτεκτονική που χρησιμοποιήθηκε. [29]

Άλλες τεχνικές αντιμετώπισης είναι η υπερδειγματοληψία (Oversampling) και η υποδειγματοληψία (Undersampling). Ένας από τους πιο γνωστούς αλγόριθμους υπερδειγματοληψίας είναι ο SMOTE (Synthetic Minority Oversampling Technique), ο οποίος δημιουργεί προσομοιωμένα δεδομένα με βάση τις ομοιότητες μεταξύ ζευγών των υφιστάμενων δειγμάτων μειοψηφίας [30]. Μπορεί να δημιουργήσει νέα παραδείγματα για τη μειωτική κλάση λαμβάνοντας δείγματα από την κλάση στόχο και συνδυάζοντας τα χαρακτηριστικά τους ή τα χαρακτηριστικά των γειτόνων τους. Ο ταξινομητής με αυτή την τεχνική μπορεί να βελτιώσει σημαντικά την απόδοσή του ως προς την κλάση μειονότητας χρησιμοποιώντας τα νέα συνθετικά δείγματα για να φέρει το dataset σε ισορροπία με την πλειοψηφική κλάση. [31] Παρόλο που δεν έχει σχεδιαστεί για τον χειρισμό εικονιστικών διανυσματικών συνόλων δεδομένων, ο SMOTE έχει την δυνατότητα να εφαρμοστεί και σε εικόνες όπως παρουσιάστηκε και στην μελέτη [32] των Mullick et al (2020) στην οποία δοκιμάστηκε σε πολυκλασικά ανισόρροπα σύνολα δεδομένων εικόνων με μεγάλη επιτυχία ως τεχνική επαύξησης των δεδομένων της μειωτικής κλάσης. Επίσης στο άρθρο [33] των Basu et al (2023) οι ερευνητές βασιζόμενοι στο [32] αντιστοιχίσαν αρχικά τις εικόνες από ένα σύνολο με ακτινογραφίες θώρακα με σκοπό την διάγνωση του COVID-19 στον χώρο χαρακτηριστικών χρησιμοποιώντας ένα μοντέλο βαθιάς μάθησης και στην συνέχεια εφάρμοσαν τον SMOTE. Το τελικό σύνολο δεδομένων με τα χαρακτηριστικά που εξισορροπήθηκαν από τον SMOTE πέρασαν σε ένα πολυεπίπεδο Multi Layer Perceptron (MLP) για να αντιστοιχηθούν στην κλάση στόχο. Σε αυτήν την μελέτη μπορούμε να δούμε πως

ο SMOTE μπορεί να συνεργαστεί με ένα μοντέλο βαθιάς μάθησης με στόχο την ταξινόμηση εικόνων.

Ωστόσο, ο SMOTE έχει περιορισμένη γενικευσιμότητα και αποτυγχάνει όταν τα δεδομένα δεν μπορούν να διαχωριστούν γραμμικά, όπως και όταν έχουν υψηλή διάσταση σύμφωνα με το άρθρο [34] του Δούζα et al (2018).

Το πρόβλημα αυτό επιχείρησαν να επιλύσουν οι Wang et al [35] μετατρέποντας τον SMOTE σε έναν αλγόριθμο τοπικά γραμμικής ενσωμάτωσης (LLE) που μπορούσε να απεικονίσει δεδομένα υψηλής διάστασης σε έναν χώρο χαμηλής διάστασης, όπου τα δεδομένα θα ήταν περισσότερο διαχωρίσιμα. Στη συνέχεια, οι Wang et al συνέλεξαν τρία σύνολα δεδομένων από εικόνες ακτινογραφίας θώρακος και επαλήθευσαν τις επιδόσεις ανίχνευσης πνευμονικών νόσων της μεθόδου τους χρησιμοποιώντας τους ταξινομητές K-Nearest Neighbors, SVM και naïve Bayes. Η αναλογία κλάσεων μειοψηφίας προς την πλειοψηφία ήταν περίπου 1:25. Η ακρίβεια ταξινόμησης του αλγορίθμου LLE-SMOTE ήταν μεγαλύτερη από εκείνη του συμβατικού SMOTE κατά 2-4%. [35]

Με κίνητρο την επιτυχία των συνθετικών προσεγγίσεων, και κυρίως του SMOTE, οι He et al (2008) στο άρθρο [36] προτείνουν μια προσαρμοστική μέθοδο για την επίλυση του προβλήματος την ανισορροπίας, η οποία είναι γνωστή στη βιβλιογραφία ως αλγόριθμος ADASYN (Adaptive Synthetic sampling). Αυτό είχε ως αποτελέσματα τη μείωση της μεροληψίας που προκύπτει από την ανισορροπία των κλάσεων και την προσαρμοστική μετατόπιση του ορίου απόφασης στη διαδικασία ταξινόμησης ως προς τα δύσκολα παραδείγματα. Ο αλγόριθμος δοκιμάστηκε σε 5 διαφορετικά σύνολα δεδομένων με ανισορροπία στις κλάσεις του. Με βάση τα αποτελέσματα που σημειώθηκαν αποδείχτηκε η αποτελεσματικότητα του καθώς υπερίσχυσε σχεδόν σε όλες τις μετρικές αξιολόγησης (Accuracy, Precision, Recall, F1 Score, G mean), έναντι του αλγορίθμου SMOTE.

Εκτός από τις τεχνικές υπερδειγματοληψίας, η ανισορροπία των κλάσεων μπορεί να αντιμετωπιστεί και με τεχνικές υποδειγματοληψίας (Undersampling). Η υποδειγματοληψία λειτουργεί μειώνοντας τα δείγματα της πλειοψηφικής κλάσης. Η μείωση μπορεί να γίνει τυχαία, οπότε στην περίπτωση αυτή ονομάζεται τυχαία υποδειγματοληψία (Random Undersampling), είτε λαμβάνοντας υπόψη κάποιο στατικό κριτήριο, οπότε παίρνει το όνομα

πληροφορημένη υποδειγματοληψία (Informed Undersampling).[37] Στο άρθρο των Roshani et al (2016) [38], προτάθηκε μια μέθοδος υποδειγματοληψίας βασισμένη σε MLP (Multi Layer Perceptron), η οποία μπορούσε να διατηρήσει την πληροφορία της κατανομής κατά την υποδειγματοληψία χρησιμοποιώντας μια στοχαστική αξιολόγηση μέτρων για τον εντοπισμό της σημαντικής πληροφορίας από τα δείγματα της πλειοψηφίας, αλλά και της μειοψηφίας. Η μέθοδος εφαρμόστηκε σε 5 πραγματικά σύνολα ιατρικών δεδομένων με σκοπό την αξιολόγησή της. Συγκρίνοντας τη μέθοδο τους (MLPUS) με τεχνικές resampling, την τεχνική SMOTE, αλλά και με την περίπτωση καμίας τεχνικής, αποδείχθηκε ότι η MLPUS απέδωσε πολύ καλύτερα αποτέλεσμα έχοντας ως μετρική αξιολόγησης το Accuracy [38].

Επίσης, οι Devi et al (2017) [39] παρουσίασαν μια προσέγγιση για την αντιμετώπιση της ανισορροπίας των κλάσεων με τη χρήση της τεχνικής Tomek-link, ενισχύοντας την ακρίβεια της κατηγοριοποίησης και δημιουργώντας έναν βελτιωμένο αλγόριθμο υποδειγματοληψίας, συνδυάζοντας πτυχές της ανίχνευσης ακραίων τιμών και πλεονασμού στα βασικά συστήματα. Το προτεινόμενο σύστημα αποσκοπούσε στην ανίχνευση ακραίων, περιπτώσεων και θορυβωδών περιπτώσεων προκειμένου να εξαλειφθούν οι περισσότερες περιπτώσεις χωρίς να χαθούν πολύτιμες πληροφορίες. Αυτή η λύση υλοποιήθηκε και επικυρώθηκε με διάφορους ταξινομητές (Back Propagation Neural Network (BPNN), K-Nearest-Neighbor (K-NN), Support Vector Machine (SVM) and Naive Bayes) σε 10 διαφορετικά πραγματικά datasets, αναδεικνύοντας έτσι την υπεροχή της έναντι των βασικών συστημάτων.

Παρόλο που οι παραπάνω τεχνικές συνθετικής δειγματοληψίας παρουσιάζουν επιτυχία σε διάφορα σύνολα δεδομένων, όταν πρόκειται για πολυδιάστατα δεδομένα, όπως εικόνα και ήχος είναι λιγότερο αποδοτικά. Επομένως, όσο ο αριθμός των διαστάσεων αυξάνεται είναι απαραίτητο να διερευνηθούν νέες προσεγγίσεις για τη δημιουργία δειγμάτων για την αντιμετώπιση της ανισορροπίας των κλάσεων.[40] Η λύση στον προβληματισμό αυτόν δόθηκε με την εμφάνιση των Παραγωγικών μοντέλων (Generative Models). Η πρώτη προσέγγιση δόθηκε από τον Hinton et al (2006) [41], καθώς κατασκεύασαν ένα βαθύ παραγωγικό μοντέλο το DBNs (Deep Belief Networks) και έναν αλγόριθμο γρήγορης μη επιβλεπόμενης μάθησης για αυτό. Το μοντέλο DBNs και ο αλγόριθμος μάθησης οδήγησαν στην ανάπτυξη περισσότερων παραγωγικών

μοντέλων, όπως οι Μηχανές Περιορισμού Boltzmann (Restricted Boltzmann Machines ή RBMs) και οι Autoencoders.

Οι μηχανές περιορισμού Boltzmann (RBMs) είναι ένας τύπος πιθανοτικού παραγωγικού μοντέλου που χρησιμοποιεί νευρωνικά δίκτυα με διασυνδεδεμένες μονάδες σε μια μοναδική αρχιτεκτονική. Κάθε μονάδα σε ένα BM κατέχει μια δυαδική κατάσταση (0 ή 1), που αντιπροσωπεύει το "on" ή το "off", και αυτές οι μονάδες επηρεάζουν η μία την άλλη πιθανοτικά, δημιουργώντας έναν δυναμικό χορό ενεργοποίησης και αναστολής.

Στο άρθρο του Hwang et al (2020) [42], οι ερευνητές γνώριζαν ότι οι μηχανές Boltzmann υστερούν σε απόδοση σε αρκετές εργασίες βαθιάς μάθησης λόγω της στατιστικής τους βάσης, αλλά και επειδή εκπαιδεύονται με δειγματοληψία Monte Carlo και όχι με gradient descent et αλαρτήσεις απώλειας όπως συμβαίνει με άλλα μοντέλα βαθιάς μάθησης. Έτσι, παρήξαν μια συνάρτηση απωλειών με σκοπό την υιοθέτηση βελτιστοποίησης με βάση την κλίση. Στη συνέχεια, αξιολογήσαν το μοντέλο τους σε δυο εργασίες, ήτοι την ταξινόμηση και την παραγωγή συνθετικών εικόνων. Στην ταξινόμηση που χρησιμοποιήθηκαν 5 datasets (MNIST, Fashion-MNIST, CIFAR-10, Caltech-101) το μοντέλο κατάφερε να επικρατήσει μόνο σε μία περίπτωση (Fashion-MNIST) ανάμεσα σε άλλα RBM-based μοντέλα πετυχαίνοντας τιμή σφάλματος 9.26%. Στην παραγωγή συνθετικών δειγμάτων, το μοντέλο των ερευνητών εκπαιδεύτηκε στο CelebA dataset και κατάφερε να παράξει ποικίλα χρώματα δέρματος, εκφράσεων προσώπου και φύλο. Επίσης το πείραμα έδειξε μια ενδιαφέρουσα ικανότητα γενίκευσης του μοντέλου, όπως η αφαίρεση αξεσουάρ από το πρόσωπο. Το συγκεκριμένο μοντέλο ήταν έως τότε το μόνο μοντέλο που μπορούσε να παράξει έγχρωμες εικόνες προσώπου υψηλής ποιότητας χρησιμοποιώντας καθαρά RBM.

Οι Autoencoders λειτουργούν συμπιέζοντας τα δεδομένα εισόδου σε μια αναπαράσταση χαμηλότερης διάστασης (Latent Space) και στη συνέχεια ανακατασκευάζουν τα αρχικά δεδομένα όσο το δυνατόν πιο κοντά από αυτή τη συμπιεσμένη αναπαράσταση. Αυτή η διαδικασία αναγκάζει τον autoencoder να μάθει τα βασικά χαρακτηριστικά που κωδικοποιούν τα δεδομένα, απορρίπτοντας τους πλεονασμούς και το θόρυβο. [43]



Στο άρθρο [44] των Siddalingappa et al (2021) χρησιμοποιήθηκαν Autoencoders με νευρωνικά συνελκτικά δίκτυα θέλοντας να αντιμετωπίσουν την ανισορροπία μεταξύ φυσιολογικών και μη φυσιολογικών δειγμάτων σε εικόνες αξονικής τομογραφίας του καρκίνου του πνεύμονα. Σκοπός τους ήταν να βοηθήσουν στην ενίσχυση της πρόγνωσης στην ιατρική απεικόνιση ανιχνεύοντας ανωμαλίες στο σύνολο δεδομένων. Η διαδικασία που ακολουθήσαν περιελάμβανε τέσσερα στάδια. Το πρώτο αφορούσε την εκπαίδευση του Autoencoder με καθορισμένες υπερπαραμέτρους στις εικόνες αξονικής τομογραφίας καρκίνου του πνεύμονα. Το δεύτερο ήταν το στάδιο δοκιμής (test) κατά το οποίο ο Autoencoder ανακατασκευάζει την είσοδο από τον λανθάνοντα χώρο (Latent Space) με μικρές αλλαγές σε σχέση με τα αρχικά δεδομένα μετρούμενη με το μέσο τετραγωνικό σφάλμα (MSE). Το τρίτο στάδιο αφορούσε την αξιολόγηση και σύγκριση των τιμών MSE στα σύνολα εκπαίδευσης και δοκιμής. Η ανωμαλία πρόκυπτε εάν το MSE ξεπερνούσε ένα συγκεκριμένο κατώφλι που είχαν ορίσει οι ερευνητές, και τέλος το στάδιο της επικύρωσης οπότε αξιολογήθηκε η απόδοση με βάση τις μετρικές Accuracy και MSE και στην περίοδο εκπαίδευσης αλλά και στην περίοδο επικύρωσης. Το dataset στη συνέχεια ταξινομήθηκε ακόμα περισσότερο σε καλοήθης και κακοήθης. Τέλος η μετρική Accuracy σημείωσε για την ανίχνευση ανωμαλιών και τη διαδικασία της ταξινόμησης τιμές 98% και 97.2 αντίστοιχα, στοιχείο που υποδεικνύει ότι ο autoencoder που κατασκεύασαν μπορεί να ανιχνεύσει ανωμαλίες σε εικόνες αξονικής τομογραφίας του καρκίνου του πνεύμονα.

Ένας άλλο τύπος autoencoder είναι οι Variational Autoencoders (VAEs) οπότε αποτελούν και την πιο δημοφιλή κατηγορία autoencoder. Στους VAEs οι οποίοι αναφέρονται για πρώτη φορά στο άρθρο των Kingma et al (2014) [45], ο encoder αποκλίνει από την άμεση παραγωγή ενός λανθάνοντος διανύσματος (latent vector) δημιουργώντας διανύσματα μέσης τιμής και διακύμανσης, σχηματίζοντας λανθάνουσες κατανομές πιθανότητας. Κατά συνέπεια, ένα λανθάνον διάνυσμα λαμβάνεται δειγματοληπτικά από αυτές τις κατανομές. Αυτός ο σχεδιασμός εξασφαλίζει ότι για μια δεδομένη εικόνα εισόδου, τα λανθάνοντα διανύσματα δεν είναι πανομοιότυπα, αναγκάζοντας τον αποκωδικοποιητή να κατανοήσει την απεικόνιση από μια περιοχή λανθάνοντος χώρου σε μια ανακατασκευή και όχι μόνο από ένα σημείο. Αυτό οδηγεί σε μια πιο ομαλή διαδικασία ανακατασκευής εικόνων. [46]

Στο άρθρο των Chatterjee et al (2022) [47] γίνεται αντιμετώπιση του προβλήματος της ανισορροπίας των κλάσεων χρησιμοποιώντας του VAEs σε ένα σύνολο δεδομένων με εικόνες ακτινογραφίας θώρακος με σκοπό τη βελτίωση του ταξινομητή στην πρόβλεψη των κλάσεων “COVID-19”, “Pneumonia”, “Normal”. Αρχικά, με τους VAEs οι εικόνες του dataset μετατραπήκαν σε λανθάνουσα περιοχή μαθαίνοντας τα πιο σημαντικά χαρακτηριστικά. Στη συνέχεια, στη λανθάνουσα διανυσματική μορφή των δεδομένων εφαρμόστηκαν αρκετές τεχνικές επαναδειγματοληψίας για την εξισορρόπηση των ανισορροπων προυπαρχούσων κλάσεων. Έπειτα, το τροποποιημένο σύνολο δεδομένων στο νέο χώρο χαρακτηριστικών χρησιμοποιείται για την εκπαίδευση των ταξινομητών. Τα αποτελέσματα της συγκεκριμένη μελέτης απέδειξαν ότι η χρήση των VAEs συνέβαλαν σημαντικά στη βελτίωση του ταξινομητή στην ανίχνευσης της κλάσης “COVID-19”.

Τα Generative Adversarial Networks (GANs) τα οποία αναφέρθηκαν πρώτη φορά στο άρθρο [48] των Goodfellow et al (2014) παρουσίασαν ένα νέο πλαίσιο για την εκπαίδευση παραγωγικών μοντέλων μέσω μιας αντιπαλικής διαδικασίας. Αυτή η καινοτόμος προσέγγιση περιελάμβανε δύο μοντέλα: ένα παραγωγικό μοντέλο Generator (G) που αποτυπώνει την κατανομή των δεδομένων και ένα διαχωριστικό μοντέλο Discriminator (D) που εκτιμά την πιθανότητα ενός δείγματος που προέρχεται από τα δεδομένα εκπαίδευσης σε σχέση με την κατανομή του G. Τα GAN εισήγαγαν ένα παιχνίδι δύο παικτών minmax, όπου το G στόχευε στη μεγιστοποίηση του σφάλματος του D, καταλήγοντας με μια μοναδική λύση. Οι δυνατότητες της αρχιτεκτονικής επικυρώθηκαν μέσω ποιοτικών και ποσοτικών αξιολογήσεων, ανοίγοντας τον δρόμο για ευρεία υιοθέτησή της. Τα GANs έφεραν επανάσταση στον τομέα προσφέροντας ένα νέο πλαίσιο για τη δημιουργία δεδομένων.

Ωστόσο, έως εκείνη την στιγμή σε ένα GAN, δεν υπήρχε κανένας έλεγχος στους τρόπους που παράγονται τα δεδομένα. Οι Mirza et al με το άρθρο [49] (2014) κατάφεραν να κατευθύνουν τη διαδικασία παραγωγής δεδομένων τροφοδοτώντας το GAN με μια επιπρόσθετη πληροφορία, όπως για παράδειγμα μια ετικέτα κλάσης. Στο άρθρο αυτό αναφέρεται για πρώτη φορά η έννοια του GAN υπό συνθήκη (conditional GAN ή cGAN) κατά το οποίο και ο Generator και ο Discriminator εξαρτώνται από κάποια επιπρόσθετη πληροφορία. Παρόλο που

τα αποτελέσματα στη μελέτη αυτή είναι εισαγωγικά, προβάλλονται οι δυνατότητες των cGANs και ανοίγει ένα νέο πεδίο μελέτης.

Στη συνέχεια στο άρθρο [50] των Radford et al (2016) προτάθηκε μια παραλλαγή των GANs, τα Deep Convolutional GANs (DCGANs), η οποία ενσωματώνει βαθιά νευρωνικά δίκτυα συνελίξεων για τη βελτίωση της παραγωγής και διάκρισης εικόνων. Επίσης, χρησιμοποιήσαν εκπαιδευμένους Discriminators για εργασίες ταξινόμησης εικόνων, παρουσιάζοντας ανταγωνιστικές επιδόσεις σε σχέση με άλλους αλγορίθμους χωρίς επίβλεψη. Ακόμα, οπτικοποίησαν τα φίλτρα που μαθαίνουν τα GANs κατά την εκπαίδευσή τους και έδειξαν εμπειρικά ότι συγκεκριμένα φίλτρα έχουν μάθει να σχεδιάζουν συγκεκριμένα αντικείμενα.

Τα GANs παρά τα εντυπωσιακά αποτελέσματα που παρουσίαζαν υποφέρουν από ζητήματα, όπως η λειτουργία κατάρρευσης (Mode Collapse) και η μη σταθερότητα στην εκπαίδευση. Για αυτόν τον λόγο, η ερευνητική κοινότητα άρχισε να προσπαθεί να δώσει λύση σε τέτοια προβλήματα. Μια από αυτές αναφέρεται στο άρθρο των Arjovsky et al (2017) [51] στο οποίο παρουσίασαν μια διεξοδική διερεύνηση των θεωρητικών βάσεων, εισήγαγαν μια νέα παραλλαγή των GAN την WGAN που βασίζεται στην απόσταση Earth Mover και κατέδειξαν τα εμπειρικά πλεονεκτήματά της στην αντιμετώπιση κρίσιμων προκλήσεων εκπαίδευσης, καθιστώντας τα WGAN μια πολλά υποσχόμενη εξέλιξη στον τομέα της παραγωγικής μοντελοποίησης. Σε αυτήν τη μελέτη δείξαν ότι η μέθοδος αυτή μπορεί να βελτιώσει τη σταθερότητα της μάθησης, να απαλλαγεί από προβλήματα, όπως η κατάρρευση λειτουργίας και να παρέχει σημαντικές καμπύλες μάθησης χρήσιμες για την αποσφαλμάτωση και την αναζήτηση υπερπαραμέτρων.

Ενώ τα WGAN βελτιώνουν τη σταθερότητα σε σύγκριση με τις παραδοσιακές τεχνικές GAN, εξακολουθούν να παρουσιάζουν κάποια αστάθεια κατά τη διάρκεια της εκπαίδευσης και να αντιμετωπίζουν και αυτά προβλήματα Mode Collapse. Επίσης, όπως πολλές παραλλαγές GAN, απαιτούν προσεκτική ρύθμιση των υπερπαραμέτρων για βέλτιστη απόδοση. Τα προβλήματα αυτά ήρθε να περιορίσει η προσέγγιση του Gulrajani et al στο άρθρο [52] το 2017 χρησιμοποιώντας στο WGAN βαθμωτή ποινή (Gradient Penalty). Εισάγοντας τον όρο της ποινής στην απώλεια του Critic επιδείξαν ισχυρές επιδόσεις μοντελοποίησης και σταθερότητας δοκιμάζοντας την σε αρκετές αρχιτεκτονικές. Με τη μελέτη αυτή συνεισφέραν στην κοινότητα έναν αλγόριθμο εκπαίδευσης

για τα GANs που ήταν ο δρόμος για ισχυρότερες επιδόσεις μοντελοποίησης για σύνολα δεδομένων εικόνας.

Τα GANs όπως και άλλα παραγωγικά μοντέλα μπορούν να χρησιμοποιηθούν ως τεχνικές επαύξησης δεδομένων και να δώσουν λύση σε προβλήματα μη επαρκή δεδομένων ή προβλήματα όπως η ανισορροπία των κλάσεων. Στο άρθρο [53] των Calar-or et al (2021) χρησιμοποιήσαν ένα GAN τροφοδοτώντας το με επιπρόσθετη πληροφορία ως τεχνική επαύξησης δεδομένων για να αντιμετωπίσουν το πρόβλημα της ανισορροπίας σε ένα σύνολο δεδομένων εικόνας ακτινογραφίας θώρακος με σκοπό τη διάγνωση της πνευμονίας. Αφού εκπαιδύσαν το cGAN τροφοδοτήσαν το αρχικό σύνολο δεδομένων με νέες συνθετικές εικόνες παραγόμενες από το cGAN. Στη συνέχεια, χρησιμοποιώντας διαφορετικά προεκπαιδευμένα μοντέλα για την εκπαίδευση του ταξινομητή και κάνοντας σύγκριση της απόδοσης του ταξινομητή χωρίς τα συνθετικά δείγματα με την απόδοση του ταξινομητή χρησιμοποιώντας συνθετικά δείγματα απέδειξαν ότι η προσθήκη των συνθετικών δειγμάτων βελτίωσε σημαντικά τα μοντέλα CNN, ειδικά τα μοντέλα που κάνουν η χρήση της ResNet αρχιτεκτονικής και κυρίως το ResNet-18 το οποίο βελτίωσε την ακρίβεια και το F1-score κατά 13,36% και 16,13% αντίστοιχα.

Επίσης στο άρθρο [54] των Khan et al (2022) έρχονται αντιμέτωποι με το πρόβλημα της ανισορροπίας των κλάσεων σε μια προσπάθεια ανίχνευσης της διαβητικής αμφιβληστροειδοπάθειας χρησιμοποιώντας έναν ταξινομητή βαθιάς μάθησης. Λόγω της ανισορροπίας των κλάσεων αλλά και των λιγοστών επισημασμένων δεδομένων πολλές προσπάθειες έχουν οδηγήσει σε λανθασμένες ταξινομήσεις. Στη συγκεκριμένη μελέτη οι ερευνητές χρησιμοποιούν ένα GAN για να δημιουργήσουν συνθετικά δείγματα και να εκπαιδύσουν ξανά τον ταξινομητή. Τα τελικά αποτελέσματα έδειξαν ότι η χρήση συνθετικών δειγμάτων από το GAN στην εκπαίδευση του ταξινομητή πέτυχε υψηλότερο accuracy με τιμή 76% σε σχέση με τη χρήση των παραδοσιακών τεχνικών επαύξησης στον ταξινομητή που πέτυχαν τιμή 72%.

Συνεχίζοντας, τα μοντέλα διάχυσης (Diffusion Models) αποτελούν μία νέα μοντέρνα κατηγορία παραγωγικών μοντέλων, τα οποία έχουν καταφέρει να σπάσουν τη μακροχρόνια κυριαρχία των GANs στη σύνθεση εικόνας. Τα diffusion models αν και δε βρίσκονται πολύ καιρό στο προσκήνιο έχουν εντυπωσιάσει με τις δυνατότητες τους σε διάφορους τομείς, όπως η

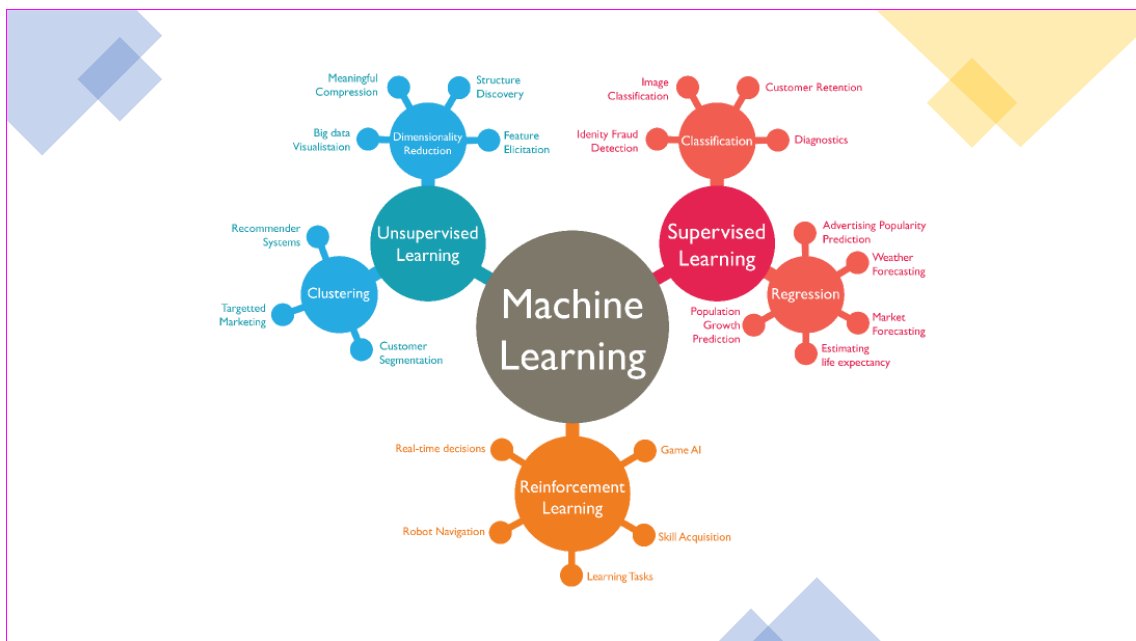
Υπολογιστική όραση, η επεξεργασία φυσικής γλώσσας και σε διεπιστημονικές εφαρμογές, όπως η υπολογιστική χημεία και η ανακατασκευή εικόνων [55], [56]. Λειτουργούν προσομοιώνοντας τη διαδικασία προσθήκης θορύβου στα δεδομένα, μετατρέποντάς τα προοδευτικά από έναν σαφή στόχο (όπως μια πραγματική εικόνα) σε καθαρό θόρυβο. Στη συνέχεια, μαθαίνουν την αντίστροφη διαδικασία, αφαιρώντας τον θόρυβο βήμα προς βήμα για να ανακτήσουν τα αρχικά δεδομένα. Αυτή η προσέγγιση "διάχυσης" και "αντίστροφης διάχυσης" τους επιτρέπει να συλλαμβάνουν τα πολύπλοκα χαρακτηριστικά και τις κατανομές των δεδομένων του πραγματικού κόσμου, επιτρέποντάς τους να παράγουν περιεχόμενο υψηλής ποιότητας, όπως εικόνες, βίντεο, ακόμη και κείμενο.[57].

Στο άρθρο [58] των Zhang et al (2023) γίνεται διερεύνηση της δυνατότητας εφαρμογής του μοντέλου SinDDM(Single Image Denoising Diffusion Model), ενός μοντέλου διάχυσης μεμονωμένης εικόνας με αποθορυβοποίηση σε ιατρικές εικόνες υπερηχογραφήματος πνεύμονα (LUS). Τα δείγματα που παράχθηκαν με την χρήση του SinDDM αξιολογήθηκαν με βάση την μετρική Single Image Frechet Inception Distance (SIFID) και την μετρική μέσης δομικής ομοιότητας (SSIM) και συγκρίθηκαν με τα δείγματα από ένα SinGAN(Single Image GAN). Τα αποτελέσματα έδειξαν ότι τα παραγόμενα δείγματα από το SinDDM υπερτερούν ως προς τις δύο μετρικές έναντι του SinGAN. Στην συνέχεια υλοποίησαν μια ταξινόμηση πολλαπλών κλάσεων με σκοπό να αξιολογήσουν την επίδραση των συνθετικών δειγμάτων από το SinDDM στην ανίχνευση ιατρικών παθήσεων. Τέλος υλοποίησαν μια παραλλαγή του SinDDM την FewDDM η οποία διαφέρει ως προς την εκπαίδευση καθώς χρησιμοποιεί περιορισμένο αριθμό δειγμάτων και όχι μια μόνο εικόνα όπως συμβαίνει με το SinDDM. Η FewDDM (Few Image Denoising Diffusion Model) όπως καταλήγουν οι ερευνητές, βελτιώνει κατά πολύ τα αποτελέσματα ως προς την ισορροπημένη ακρίβεια και τη συνολική ακρίβεια σε σύγκριση με τις μεθόδους SinDDM και SinGAN. Πιο συγκεκριμένα το F1-Score για την κλάση "Pneumonia" αυξάνεται σημαντικά όταν στα δεδομένα εκπαίδευσης προστίθεται και τακτική επαύξηση δεδομένων με συνθετικές εικόνες από το FewDDM.

## 2.2 Θεωρητικό Υπόβαθρο

### 2.2.1 Μηχανική Μάθηση

Η Μηχανική Μάθηση αποτελεί υποσύνολο της Τεχνητής Νοημοσύνης για τον σχεδιασμό και την ανάπτυξη αλγορίθμων και συστημάτων που επιτρέπουν σε έναν υπολογιστή να μάθει από δεδομένα και να κάνει προβλέψεις ή να πάρει αποφάσεις χωρίς να χρειάζεται ανθρώπινη παρέμβαση. Χρησιμοποιείται ευρέως σε εφαρμογές, όπως η αναγνώριση προτύπων, η κατηγοριοποίηση και η ομαδοποίηση. Οι τεχνικές μηχανικής μάθησης μπορούν να ταξινομηθούν σε τρεις γενικές κατηγορίες: (Εικόνα 1) τη *Μάθηση με επίβλεψη (Supervised Learning)*, τη *Μάθηση χωρίς επίβλεψη (Unsupervised Learning)* και την *Ενισχυτική μάθηση (Reinforcement Learning)*. [59]



Εικόνα 1 Μηχανική μάθηση- κατηγορίες και εφαρμογές (πηγή από Abdul Wahid)

### 2.2.2 Μάθηση με επίβλεψη (Supervised Learning)

Η μάθηση με επίβλεψη είναι ένας τύπος μηχανικής μάθησης όπου ο αλγόριθμος εκπαιδεύεται σε ένα σύνολο δεδομένων με ετικέτες. Στην επιβλεπόμενη μάθηση, το σύνολο δεδομένων αποτελείται από χαρακτηριστικά εισόδου (αλλιώς προβλέψεις ή ανεξάρτητες μεταβλητές) και ετικέτες εξόδου (αλλιώς αποκρίσεις ή εξαρτημένες μεταβλητές), οι οποίες είναι ήδη γνωστές. Στόχος της είναι η εκμάθηση μιας συνάρτησης αντιστοίχισης από τα χαρακτηριστικά εισόδου στις ετικέτες εξόδου ελαχιστοποιώντας το σφάλμα μεταξύ της προβλεπόμενης εξόδου και της πραγματικής εξόδου [60].

### **2.2.3 Μάθηση χωρίς επίβλεψη (Unsupervised Learning)**

Η μάθηση χωρίς επίβλεψη είναι ένας τύπος μηχανικής μάθησης όπου ο αλγόριθμος εκπαιδεύεται σε ένα σύνολο δεδομένων χωρίς ετικέτες. Στη μάθηση χωρίς επίβλεψη, δεν υπάρχουν προκαθορισμένες ετικέτες εξόδου και ο αλγόριθμος καλείται να βρει μοτίβα ή δομές στα δεδομένα από μόνος του.

Ο στόχος της είναι η εκμάθηση μιας συνάρτησης αντιστοίχισης από τα χαρακτηριστικά εισόδου σε μια χρήσιμη αναπαράσταση των δεδομένων. Αυτό μπορεί να περιλαμβάνει τον εντοπισμό συστάδων (Clustering), την ανακάλυψη κρυφών μεταβλητών ή τη μείωση διαστάσεων (Dimensionality Reduction) των δεδομένων [60]. Οι κυριότερες εργασίες αξιοποίησης της μάθησης χωρίς επίβλεψη είναι η συσταδοποίηση (Clustering) και η μείωση διαστάσεων (Dimensionality Reduction) [60].

### **2.2.4 Ενισχυτική μάθηση (Reinforcement Learning)**

Η ενισχυτική μάθηση (Reinforcement Learning) είναι ένας τύπος μηχανικής μάθησης όπου ένας πράκτορας (agent) μαθαίνει να λαμβάνει μια σειρά αποφάσεων αλληλοεπιδρώντας με κάποιο περιβάλλον. Ο στόχος της ενισχυτικής μάθησης είναι η εύρεση μιας βέλτιστης πολιτικής (optimal policy) που μεγιστοποιεί ένα μακροπρόθεσμο σήμα ανταμοιβής [61].

Στην ενισχυτική μάθηση, ο πράκτορας παρατηρεί την τρέχουσα κατάσταση του περιβάλλοντος, επιλέγει μια ενέργεια για εκτέλεση και λαμβάνει ένα σήμα ανταμοιβής με βάση την ενέργεια, καθώς και την προκύπτουσα κατάσταση μετάβασης. Στη συνέχεια, ενημερώνει την πολιτική του με βάση τις παρατηρούμενες ανταμοιβές και τις μεταβάσεις κατάστασης, με στόχο τη μεγιστοποίηση της αναμενόμενης αθροιστικής ανταμοιβής στην πάροδο του χρόνου.

### **2.2.5 Βαθιά Μάθηση (Deep Learning)**

Η βαθιά μάθηση (Deep Learning) είναι ένα υποπεδίο της μηχανικής μάθησης που αφορά τεχνικές και μεθόδους εμπνευσμένες από τη δομή και τη λειτουργία του εγκεφάλου. Συγκεκριμένα, στη βαθιά μάθηση χρησιμοποιούνται υπολογιστικά συστήματα και αλγόριθμοι, γνωστοί ως Τεχνητά Νευρωνικά Δίκτυα (ANN), ή εν συντομία Νευρωνικά Δίκτυα (NN), που εμπνεύστηκαν από

τα βιολογικά νευρωνικά δίκτυα που συνθέτουν τους εγκεφάλους των ανθρώπων και των ζώων [62].

## **2.2.6 Ιστορική Αναδρομή**

Η βαθιά μάθηση και τα νευρωνικά δίκτυα έχουν μια μακρά και συναρπαστική ιστορία, που χρονολογείται από τη δεκαετία του 1940. Όλα ξεκίνησαν με τους Warren McCulloch και Walter Pitts [63], οι οποίοι πρότειναν το πρώτο μαθηματικό μοντέλο ενός νευρωνικού δικτύου το 1943. Αυτό το μοντέλο έθεσε τα θεμέλια για τον τομέα της τεχνητής νοημοσύνης.

Στις δεκαετίες του 1950 και του 1960, οι ερευνητές συνέχισαν να βελτιώνουν και να αναπτύσσουν την έννοια των νευρωνικών δικτύων, αλλά η πρόοδος ήταν αργή λόγω της έλλειψης υπολογιστικής ισχύος. Μόνο στα τέλη της δεκαετίας του 1980 και στις αρχές της δεκαετίας του 1990 η βαθιά μάθηση και τα νευρωνικά δίκτυα άρχισαν να κερδίζουν μεγαλύτερη προσοχή, καθώς οι υπολογιστές έγιναν πιο ισχυροί και ο τομέας της μηχανικής μάθησης άρχισε να ανθίζει.

Μια από τις βασικές εξελίξεις στην ιστορία της βαθιάς μάθησης ήταν η εφεύρεση του αλγορίθμου backpropagation το 1986. Αυτός ο αλγόριθμος επέτρεψε στα νευρωνικά δίκτυα να μαθαίνουν από τα λάθη τους και να βελτιώνονται με την πάροδο του χρόνου, καθιστώντας δυνατή την εκπαίδευση πολύ μεγαλύτερων και πολύπλοκων δικτύων. [64] Ένα άλλο σημαντικό γεγονός ήταν η ανάπτυξη των συνελκτικών νευρωνικών δικτύων (ConvNets) τη δεκαετία του 1990. Τα ConvNets σχεδιάστηκαν ειδικά για εργασίες αναγνώρισης εικόνων και έκτοτε αποτελούν έναν από τους πιο ευρέως χρησιμοποιούμενους τύπους νευρωνικών δικτύων στον τομέα του Computer Vision [65].

Το 2012, ένας αλγόριθμος βαθιάς μάθησης που αναπτύχθηκε από την Google με το όνομα AlphaGo, νίκησε τον παγκόσμιο πρωταθλητή του παιχνιδιού Go, το οποίο είναι ένα πολύπλοκο επιτραπέζιο παιχνίδι με τεράστιο αριθμό πιθανών κινήσεων [66].

Η βαθιά μάθηση συνέχισε να εξελίσσεται και να γίνεται πιο διαδεδομένη, με εφαρμογή σε τομείς που κυμαίνονται από την επεξεργασία φυσικής γλώσσας έως την αυτοοδήγηση αυτοκινήτων. Επίσης, θεωρείται ένας από τους πιο υποσχόμενους τομείς έρευνας στην τεχνητή νοημοσύνη, με τη δυνατότητα να φέρει επανάσταση σε ένα ευρύ φάσμα βιομηχανιών και να μεταμορφώσει τον τρόπο με τον οποίο ζούμε και εργαζόμαστε.

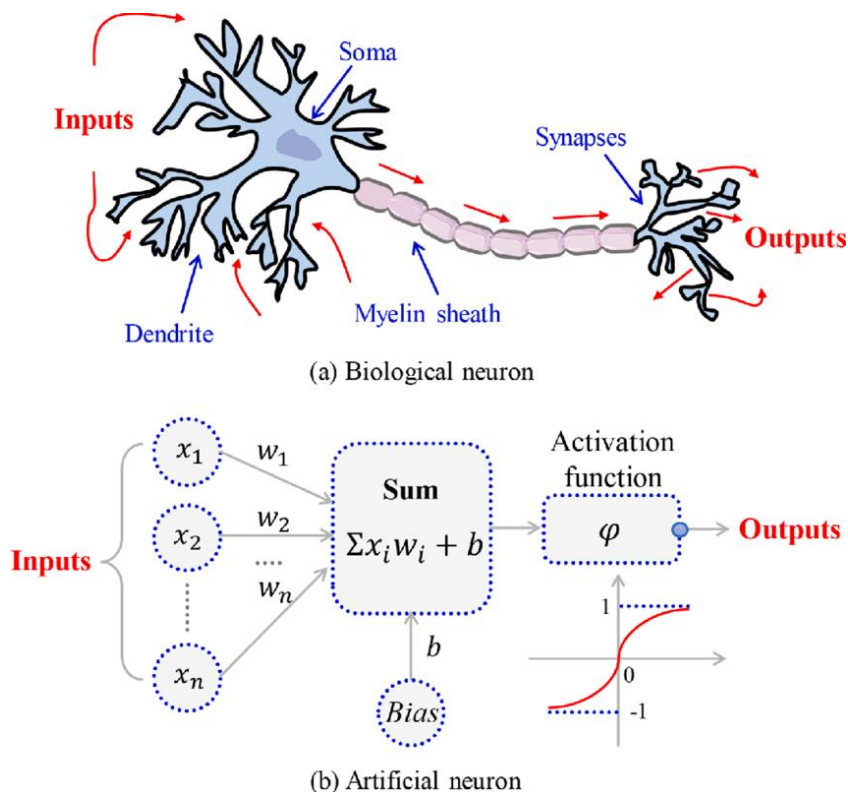


## 2.2.7 Τεχνητά Νευρωνικά Δίκτυα (Artificial Neural Networks)

Η ιδέα των Τεχνητών Νευρωνικών Δικτύων προέρχεται από τη βιολογική μελέτη των νευρωνικών δικτύων, τα οποία παίζουν πολύ σημαντικό ρόλο στη λειτουργία του ανθρώπινου σώματος, καθώς οποιαδήποτε λειτουργία γίνεται με τη βοήθειά τους. Με τη βοήθεια των εκατομμύρια διασυνδεδεμένων νευρώνων επιτυγχάνεται η παράλληλη επεξεργασία στο ανθρώπινο σώμα.

Στην Εικόνα 2(a) φαίνεται η μορφή που έχει ένα βιολογικό νευρωνικό δίκτυο, καθώς και τα κύρια μέρη από τα οποία αποτελείται. Ακολουθώντας τη λογική των ανθρώπινων νευρωνικών δικτύων που αναφέρθηκε παραπάνω, κατασκευάστηκαν τα Τεχνητά Νευρωνικά Δίκτυα (ΤΝΔ) Εικόνα 2 (b). Ένα ΤΝΔ (ANN) αποτελείται από πολλά απλά στοιχεία επεξεργασίας διασυνδεδεμένα μεταξύ τους και χωρισμένα σε επίπεδα.

Παρόμοια με τον βιολογικό νευρώνα, το τεχνητό νευρωνικό δίκτυο έχει, επίσης, νευρώνες που λαμβάνουν εισροές από τα άλλα στοιχεία ή άλλους τεχνητούς νευρώνες και στη συνέχεια, αφού οι εισοδοί σταθμιστούν (weighted) και προστεθούν, το αποτέλεσμα μεταφέρεται μέσω μιας συνάρτησης (συνάρτηση μεταφοράς) στην έξοδο [67].



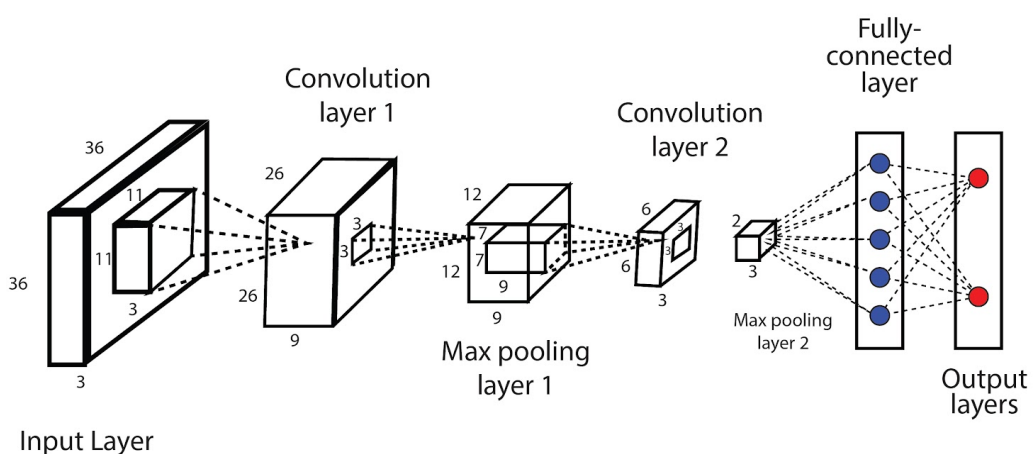
Εικόνα 2 Ο Βιολογικός νευρώνας έναντι του Τεχνητού νευρώνα

## 2.2.8 Συνελκτικά Νευρωνικά Δίκτυα

Τα Συνελκτικά Νευρωνικά Δίκτυα (Convolution Neural Networks -CNN ή convnets), είναι μια ειδική περίπτωση ΤΝΔ πρόσθιας τροφοδότησης (feed forward). Μοιάζουν πολύ με τα ΤΝΔ με την έννοια ότι αποτελούνται από νευρώνες με μαθησιακά βάρη και μεροληψία (bias). Στην ουσία, η διαφορά είναι ότι η αρχιτεκτονική των CNN κάνει τη σιωπηρή υπόθεση ότι η είσοδος που δέχεται είναι σαν μια εικόνα, γεγονός που επιτρέπει την κωδικοποίηση ορισμένων ιδιοτήτων στην αρχιτεκτονική [68].

Συγκεκριμένα, οι συνελιξεις (convolutions) αποτυπώνουν τη μετάφραση αναλλοίωτη (translation invariance) (π.χ. τα φίλτρα είναι ανεξάρτητα από τη θέση). Αυτό έχει ως αποτέλεσμα, η εμπρόσθια συνάρτηση να γίνεται πιο αποδοτική, ο αριθμός των παραμέτρων να μειώνεται σημαντικά, και το δίκτυο να γίνεται ευκολότερο στη διαδικασία βελτιστοποίησης και λιγότερο εξαρτημένο από το μέγεθος των δεδομένων [68].

Σε αντίθεση με τα απλά νευρωνικά δίκτυα, τα επίπεδα ενός CNN έχουν νευρώνες τοποθετημένους σε λίγες διαστάσεις [68]. Ένα CNN αποτελείται από μια ακολουθία επιπέδων, εκ των οποίων τα πιο συνηθισμένα είναι: *το επίπεδο συνέλιξης (convolution layer), το επίπεδο συγκέντρωσης (pooling layer), και τα πλήρως συνδεδεμένα επίπεδα (fully connected layers)*. Στην Εικόνα 3, παρουσιάζεται η βασική δομή ενός απλού CNN δύο επιπέδων και διαφορά επίπεδα που αναφέρθηκαν.



Εικόνα 3 Η δομή ενός απλού CNN

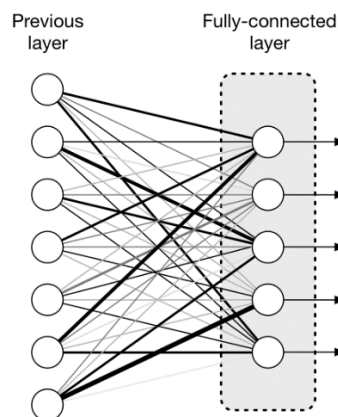
Το σημαντικότερο επίπεδο στην αρχιτεκτονική των Convolutional Neural Networks (CNN) είναι το συνελκτικό επίπεδο. Σε αυτό, εφαρμόζονται

μαθησιακά φίλτρα ή πυρήνες στην εικόνα εισόδου για εξαγωγή χαρακτηριστικών, όπως ακμές και σχήματα. Η συνέλιξη είναι η μαθηματική πράξη που υπολογίζει το εσωτερικό γινόμενο φίλτρου με μια περιοχή της εικόνας, γνωστή ως δεκτικό πεδίο.

Μετά το συνελκτικό επίπεδο, ακολουθεί το επίπεδο συγκέντρωσης (Pooling layer), που μειώνει τις διαστάσεις των χαρακτηριστικών, ενώ ενισχύει τα χαρακτηριστικά. Υπάρχουν διάφοροι τύποι, όπως το Max Pooling και το Average Pooling [69].

Οι συναρτήσεις ενεργοποίησης εφαρμόζονται μετά τα επίπεδα συνέλιξεων. Η Sigmoid χρησιμοποιείται για δυαδική ταξινόμηση, η tanh συχνά για εργασίες παλινδρόμησης, ενώ η Rectified Linear Unit (ReLU) είναι δημοφιλής λόγω της απλότητάς της και του χαμηλού υπολογιστικού κόστους [68].

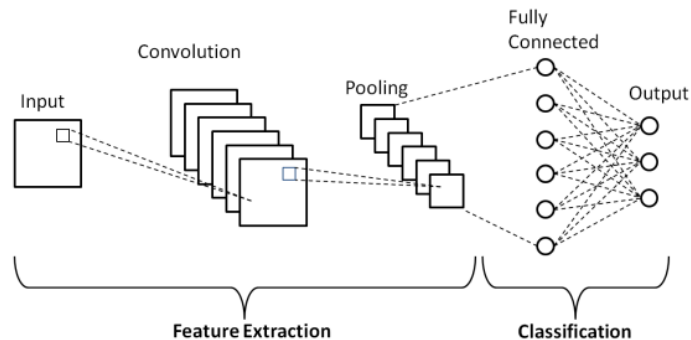
Τέλος, τα πλήρως συνδεδεμένα επίπεδα συνδέουν κάθε νευρώνα με όλους του προηγούμενου επιπέδου και εκτελούν γραμμικούς μετασχηματισμούς. Είναι κρίσιμα για την τελική απόφαση κλάσης εικόνας σε ένα CNN. Αυτή η διαδικασία φαίνεται συνοπτικά στο παράδειγμα της εικόνας εισόδου που ταξινομείται μέσω του Fully Connected layer. (Εικόνα 4) [65].



Εικόνα 4 Η σύνδεση μεταξύ του τελευταίου επιπέδου ενός CNN με το πλήρως συνδεδεμένο επίπεδο

Κάθε νευρώνας σε αυτά τα επίπεδα εκτελεί γραμμικό μετασχηματισμό, ακολουθούμενο από συνάρτηση ενεργοποίησης [70]. Τα πλήρως συνδεδεμένα επίπεδα είναι κρίσιμα για την τελική απόφαση σχετικά με την κλάση της εικόνας εισόδου σε ένα CNN [71].

Στην Εικόνα 5, φαίνεται ένα παράδειγμα όλης της διαδικασίας από τη στιγμή που η εικόνα εισάγεται στο δίκτυο μέχρι το τέλος που ταξινομείται μέσω του Fully Connected layer.

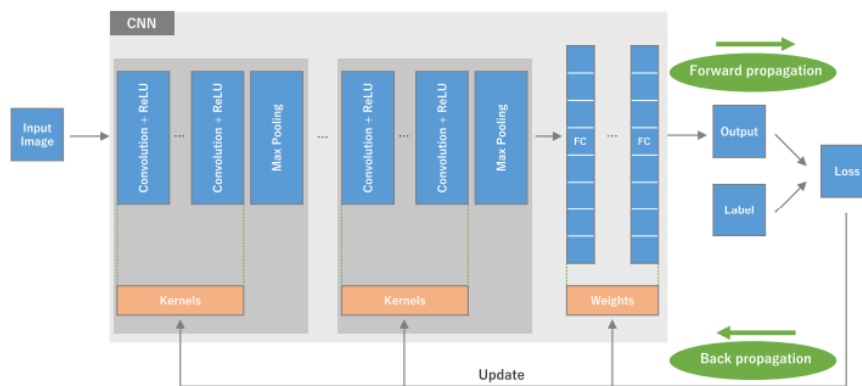


Εικόνα 5 Παράδειγμα ενός CNN μέχρι το στάδιο της ταξινόμησης

### Εκπαίδευση Δικτύου

Η εκπαίδευση ενός δικτύου είναι μια διαδικασία εύρεσης πυρήνων (kernels) στα επίπεδα συνέλιξης και βαρών στα πλήρως συνδεδεμένα επίπεδα που ελαχιστοποιούν τις διαφορές μεταξύ των προβλέψεων εξόδου και των δοθέντων πραγματικών ετικετών ενός συνόλου δεδομένων εκπαίδευσης [71].

Ο αλγόριθμος που συνήθως χρησιμοποιείται για την εκτέλεση της παραπάνω διαδικασίας και κατά τον οποίο εκπαιδεύονται συνήθως τα περισσότερα νευρωνικά δίκτυα είναι ο *Backpropagation* (οπισθοδιάδοση). Στον συγκεκριμένο αλγόριθμο, η συνάρτηση απωλειών (Loss function) και ο αλγόριθμος βελτιστοποίησης κλίσης (gradient descent optimization) παίζουν καθοριστικό ρόλο. Στην Εικόνα 6 παρουσιάζεται σχηματικά η διαδικασία εκπαίδευσης του δικτύου [71].



Εικόνα 6 Διαδικασία εκπαίδευσης ενός CNN [71].

### Συνάρτηση απώλειας (Loss function)

Μια συνάρτηση απωλειών, επίσης γνωστή ως συνάρτηση κόστους ή αντικειμενική συνάρτηση, είναι μια μαθηματική συνάρτηση που μετρά τη διαφορά μεταξύ των προβλεπόμενων τιμών και των πραγματικών τιμών του δικτύου μέσω της προς τα εμπρός διάδοσης. Συνήθως χρησιμοποιούμενη συνάρτηση απωλειών για ταξινόμηση πολλαπλών κατηγοριών είναι η διασταυρούμενη εντροπία (cross-entropy) [71].

#### 1. Απώλεια δυαδικής διασταυρούμενης εντροπίας/Απώλεια λογαρίθμου (Binary Cross Entropy/Log Loss)

Είναι μία από τις πιο συνηθισμένες συνάρτησης απώλειας που χρησιμοποιείται σε προβλήματα ταξινόμησης. Η απώλεια διασταυρούμενης εντροπίας μειώνεται καθώς η προβλεπόμενη πιθανότητα συγκλίνει στην πραγματική ετικέτα. Μετρά την απόδοση ενός μοντέλου ταξινόμησης του οποίου η προβλεπόμενη έξοδος είναι μια τιμή πιθανότητας μεταξύ 0 και 1.

Όταν ο αριθμός των κλάσεων είναι 2, πρόκειται για δυαδική ταξινόμηση:

$$L = -\frac{1}{m} \sum_{i=1}^m (y_i \cdot \log(y_i) + (1 - y_i) \cdot \log(1 - y_i)) \quad (1)$$

Όταν ο αριθμός των κλάσεων είναι παραπάνω από 2, πρόκειται για ταξινόμηση πολλαπλών κλάσεων.

$$L = -\frac{1}{m} \sum_{i=1}^m y_i \cdot \log(y_i) \quad (2)$$

### Gradient Descent

Η βελτιστοποίηση παραμέτρων σε ένα CNN αποτελεί μία πρόκληση. Ο αλγόριθμος βελτιστοποίησης Gradient Descent είναι κλειδί για εύρεση βαρών που ελαχιστοποιούν τη συνάρτηση απώλειας κατά την εκπαίδευση.

Ο αλγόριθμος Gradient Descent χρησιμοποιείται για την επαναληπτική ενημέρωση των βαρών και της μεροληψίας (bias) του δικτύου κατά τη φάση της εκπαίδευσης. Στόχος της εκπαίδευσης είναι η ελαχιστοποίηση μιας δεδομένης συνάρτησης απωλειών που μετρά τη διαφορά μεταξύ της προβλεπόμενης και της πραγματικής εξόδου του δικτύου.

Ο Gradient Descent λειτουργεί υπολογίζοντας την κλίση της συνάρτησης απωλειών σε σχέση με κάθε βάρος και προκατάληψη του δικτύου κάνοντας

χρήση του αλγορίθμου οπισθοδιάδοσης (backpropagation). Ο αλγόριθμος οπισθοδιάδοσης διαδίδει το σφάλμα από το επίπεδο εξόδου προς τα πίσω μέσω των επιπέδων του δικτύου, υπολογίζοντας την παράγωγο της συνάρτησης απώλειας, σε σχέση με κάθε βάρους και προκατάληψη.

Ο υπολογισμός της κλίσης της συνάρτησης απώλειας παρέχει την κατεύθυνση προς την οποία η συνάρτηση έχει τον πιο απότομο ρυθμό αύξησης και κάθε μαθησιακή παράμετρος ενημερώνεται προς την αρνητική κατεύθυνση της κλίσης με ένα αυθαίρετο μέγεθος βήματος που καθορίζεται με βάση μια υπερπαράμετρο που ονομάζεται ρυθμός μάθησης [71].

Έχουν προταθεί και χρησιμοποιηθεί ευρέως πολλές βελτιώσεις του αλγορίθμου gradient descent, όπως ο SGD με ορμή (SGD with momentum), ο RMSprop και ο Adam εκ των οποίων μερικές από αυτές χρησιμοποιήθηκαν και στη συγκεκριμένη μελέτη για την εκπαίδευση και βελτιστοποίηση του Generative Adversarial Network που υλοποιήθηκε και θα παρουσιαστεί αναλυτικά σε ακόλουθη ενότητα.

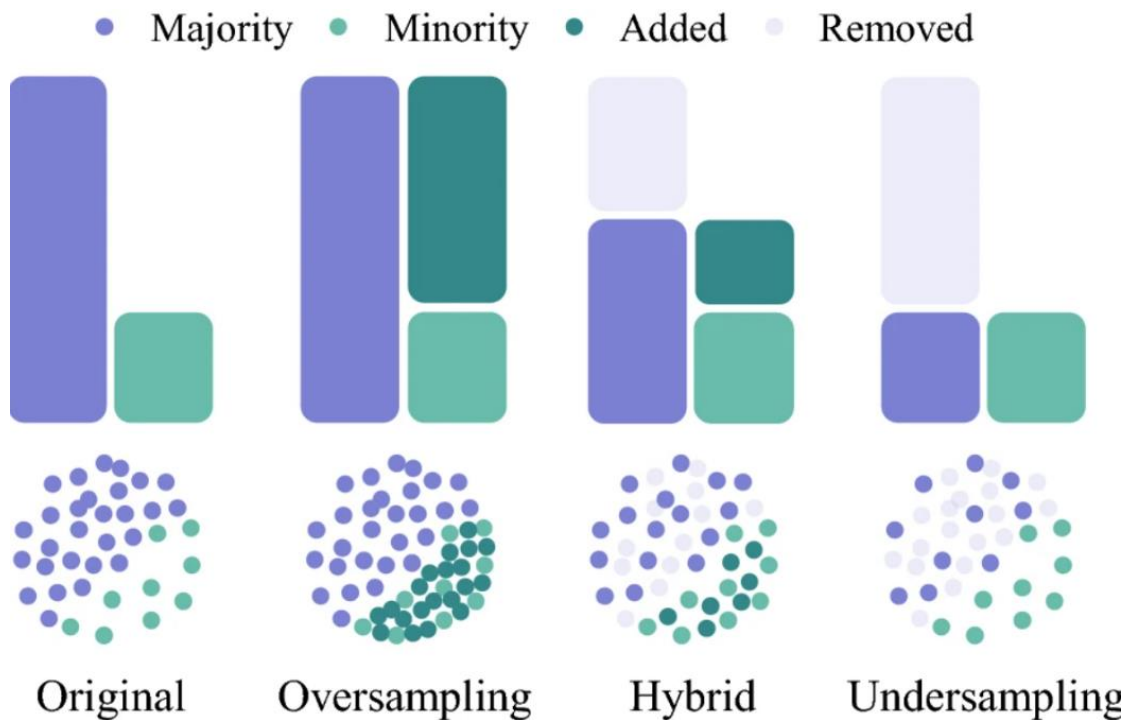
## **2.3 Τεχνικές Αντιμετώπισης της Ανισορροπίας**

Η λύση για τα ανισορροπημένα δεδομένα μπορεί να υλοποιηθεί σε δύο επίπεδα[72]:

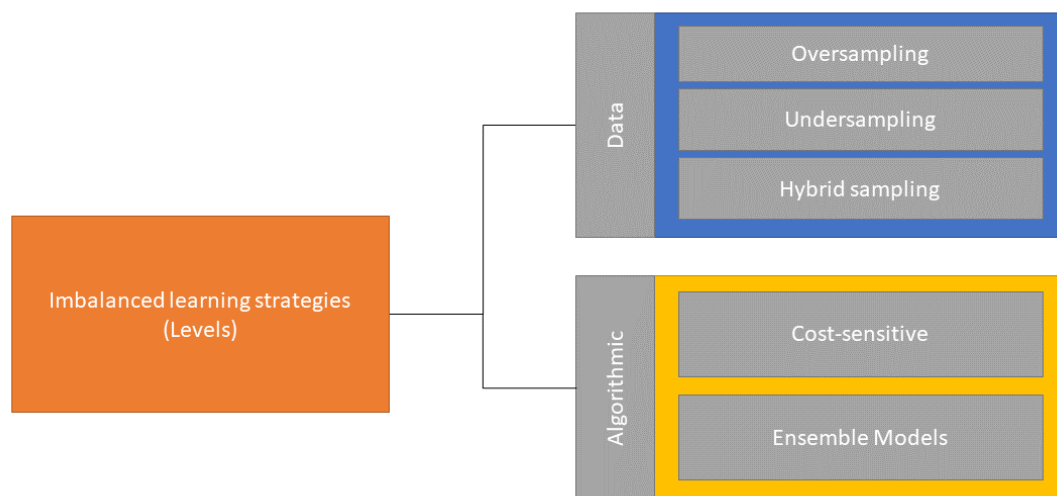
*Επίπεδο 1: Δεδομένων:* προεπεξεργασία των δεδομένων πριν την εκπαίδευση μέσω αλγορίθμων, για υποδειγματοσκόπηση (undersampling) της μεγαλύτερης κλάσης, υπερδειγματοσκόπηση (oversampling) της μικρότερης κλάσης ή και τα δύο (hybrid sampling) όπως απεικονίζεται στην Εικόνα 7.

*Επίπεδο 2: Αλγοριθμικών:* Επεξεργασία της εκπαίδευσης μέσω βελτιστοποιημένων αλγορίθμων για ανισορροπημένα δεδομένα, όπως οι αλγόριθμοι που επιμελούνται το κόστος (Cost-sensitive) και οι αλγόριθμοι μοντέλων συνόλου μηχανικής μάθησης (Ensemble Models).

Στην Εικόνα 8 φαίνονται σχηματικά τα επίπεδα επίλυσης του προβλήματος της ανισορροπίας των κλάσεων τα οποία και αναλύονται στη συνέχεια.



Εικόνα 7 Η κατανομή των δεδομένων με τη χρήση τεχνικών προεπεξεργασίας δεδομένων (Ovesampling, Undersampling, Hybrid) [72].



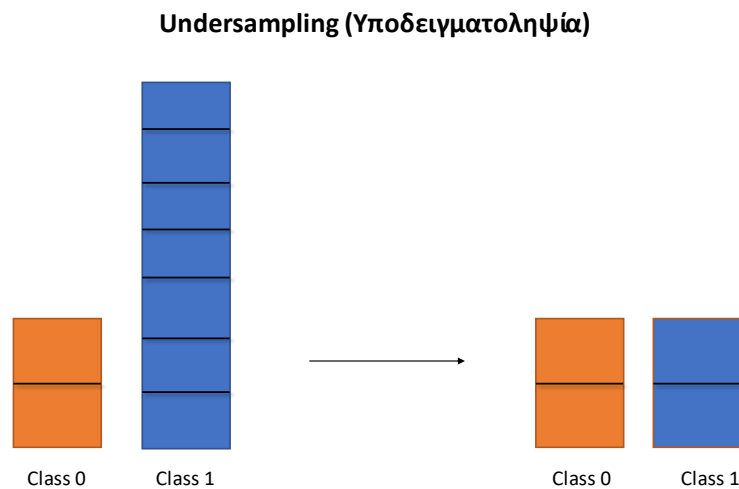
Εικόνα 8 Στρατηγικές αντιμετώπισης της ανισορροπίας των κλάσεων

### 2.3.1 Undersampling

Η υποδειγματοληψία αναφέρεται σε μια ομάδα τεχνικών που έχουν σχεδιαστεί για την εξισορρόπηση της κατανομής κλάσεων για ένα σύνολο δεδομένων ταξινόμησης που έχει λοξή κατανομή κλάσεων. Μια ανισόρροπη κατανομή κλάσεων θα έχει μία ή περισσότερες κλάσεις με λίγα παραδείγματα (οι κλάσεις μειοψηφίας) και μία ή περισσότερες κλάσεις με πολλά παραδείγματα (οι πλειοψηφικές κλάσεις). Για καλύτερη κατανόηση, μπορούμε να το σκεφτούμε

στο πλαίσιο ενός δυαδικού προβλήματος ταξινόμησης (δύο κλάσεων) όπου η κλάση 1 είναι η κλάση της πλειοψηφίας και η κλάση 0 είναι η κλάση της μειοψηφίας. Στην Εικόνα 9 παρουσιάζονται πως διαμορφώνονται τα δεδομένα μετά την εφαρμογή μια τεχνικής υποδειγματοληψίας.

Οι τεχνικές υποδειγματοληψίας αφαιρούν παραδείγματα από το σύνολο δεδομένων εκπαίδευσης που ανήκουν στην κλάση της πλειοψηφίας, ώστε να εξισορροπηθεί καλύτερα η κατανομή των κλάσεων, όπως η μείωση της στρέβλωσης από 1:100 σε 1:10, 1:2 ή ακόμη και σε 1:1 κατανομή κλάσεων.



Εικόνα 9 Εξισορρόπηση των δεδομένων με τη μέθοδο της υποδειγματοληψίας

Αυτό μπορεί να βοηθήσει στη βελτίωση της ακρίβειας του μοντέλου στην αναγνώριση της μειονοτικής κλάσης. Ωστόσο, απαιτείται προσοχή κατά την υποδειγματοληψία, καθώς μπορεί να χαθεί σημαντική πληροφορία από την πλειονοτική κλάση. Επίσης, αν το ποσοστό υποδειγματοληψίας είναι υπερβολικά υψηλό, μπορεί να προκύψει το φαινόμενο του *overfitting*.

Μια από τις πιο απλές τεχνικές υποδειγματοληψίας είναι η επιλογή τυχαίων παραδειγμάτων από την πλειονοτική κλάση και αφαίρεσή τους από το σύνολο δεδομένων εκπαίδευσης. Στη βιβλιογραφία αναφέρεται ως τυχαία υποδειγματοληψία (Random undersampling). Ωστόσο, παρά το γεγονός ότι είναι απλή στη χρήση, ένα σημαντικός περιορισμός είναι ότι τα παραδείγματα αφαιρούνται χωρίς να λαμβάνεται υπόψη το πόσο σημαντικά μπορεί να είναι για τον καθορισμό του ορίου απόφασης μεταξύ των κλάσεων, δηλαδή αρκετά χρήσιμες πληροφορίες μπορεί δυστυχώς να αφαιρεθούν και αυτό έχει ως αποτέλεσμα τη μείωση της απόδοσης του μοντέλου αντί της βελτίωσης [73].



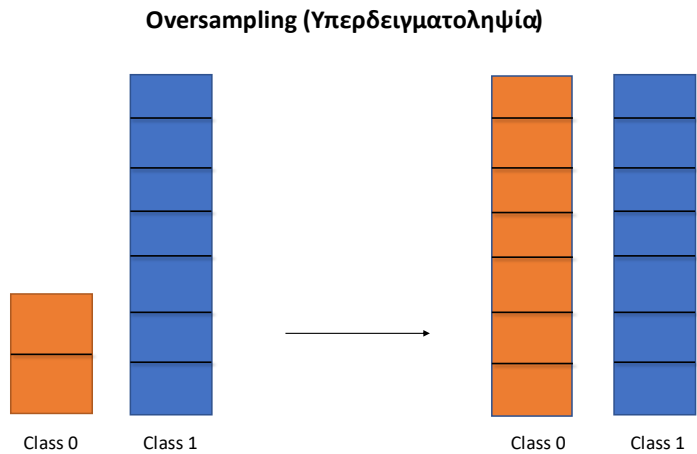
Μερικές τεχνικές που μπορεί να βοηθήσουν στον περιορισμό του παραπάνω προβλήματος είναι η υποδειγματοληψία με συνδέσμους Tomek, η οποία περιλαμβάνει την αφαίρεση των παραδειγμάτων της πλειοψηφικής κλάσης που είναι πιο κοντά στα παραδείγματα της μειονοτικής κλάσης. Έτσι, το όριο απόφασης μεταξύ των δύο κλάσεων γίνεται σαφέστερο και μπορεί να βοηθήσει στη σύλληψη των σημαντικών πληροφοριών στην κλάση της μειονότητας [39].

Άλλη τεχνική παρόμοιας λογικής με τους συνδέσμους Tomek είναι η υποδειγματοληψία κέντρου συστάδας (Cluster centroid undersampling) στην οποία γίνεται ομαδοποίηση των παραδειγμάτων της πλειοψηφικής κλάσης και στη συνέχεια αφαίρεση των παραδειγμάτων που βρίσκονται πλησιέστερα στα κέντρα των ομάδων. Με αυτή τη μέθοδο διατηρείται η σημαντική πληροφορία μειώνοντας παράλληλα τον αριθμό των παραδειγμάτων [74]. Τέλος, είναι η υποδειγματοληψία NearMiss που περιλαμβάνει την επιλογή των παραδειγμάτων της πλειοψηφικής κλάσης που βρίσκονται πλησιέστερα στα παραδείγματα της μειοψηφικής κλάσης. Στόχος της είναι η επιλογή παραδειγμάτων με μεγάλο μέρος σημαντικής πληροφορίας που μπορούν να βοηθήσουν στη διάκριση της κλάσης μειονότητας από την κλάση πλειοψηφίας [75].

### **2.3.2 Oversampling**

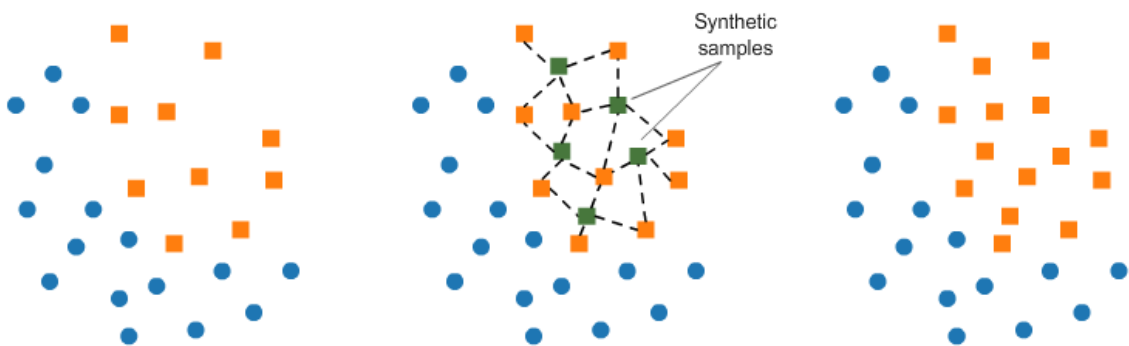
Μια άλλη τεχνική αντιμετώπισης τους προβλήματος της ανισορροπίας των κλάσεων είναι η υπερδειγματοληψία. Η υπερδειγματοληψία περιλαμβάνει την αύξηση του αριθμού των παραδειγμάτων που ανήκουν στη μειοψηφική κλάση με τη δημιουργία συνθετικών σημείων δεδομένων (data points) ή την αντιγραφή των υφιστάμενων. Αυτό γίνεται για να δημιουργηθεί ένα πιο ισορροπημένο σύνολο δεδομένων που μπορεί να χρησιμοποιηθεί για την εκπαίδευση ενός μοντέλου μηχανικής μάθησης. Με την επαύξηση των παραδειγμάτων της κλάσης μειονότητας, το μοντέλο μπορεί να μάθει να διακρίνει καλύτερα μεταξύ των κλάσεων μειονότητας και πλειοψηφίας και να παράγει ακριβέστερες προβλέψεις για την κλάση μειονότητας (Εικόνα 10).

Η υπερδειγματοληψία είναι μια αρκετά δημοφιλής τεχνική και χρησιμοποιείται ευρέως σε ένα μεγάλο φάσμα προβλημάτων ταξινόμησης, ωστόσο όπως και στη μέθοδο της υποδειγματοληψίας πρέπει να εφαρμόζεται με προσοχή καθώς η προσθήκη περισσότερων συνθετικών δεδομένων μπορεί να οδηγήσει σε overfitting και να μειώσει την ιδιότητα του μοντέλου για γενίκευση.



Εικόνα 10 Εξισορρόπηση των δεδομένων με τη μέθοδο της υπερδειγματοληψίας

Η πρώτη λύση στο πρόβλημα του overfitting δόθηκε από τον Chawla το 2002 με την τεχνική SMOTE (Synthetic Minority Oversampling Technique). Με τη συγκεκριμένη μέθοδο τα δείγματα παράγονται συνθετικά και δεν αντικαθίστανται από υφιστάμενα, γεγονός που δημιουργεί το συγκεκριμένο πρόβλημα. Συγκεκριμένα, η τεχνική αυτή χρησιμοποιείται για τη δημιουργία τεχνητών παραδειγμάτων μειοψηφίας κατά μήκος των τμημάτων γραμμής ενώνοντας τα παραδείγματα μειοψηφίας και τους K-Nearest Neighbors από τη μειοψηφική κλάση (Εικόνα 11). Βασιζόμενη στο απαιτούμενο ποσοστό που απαιτείται για την υπερδειγματοληψία, γίνεται τυχαία επιλογή από τους K-Nearest Neighbors. Από την άλλη, ένα μειονέκτημα που έχει η SMOTE είναι η υπερβολική γενίκευση της μειοψηφικής κλάσης χωρίς να λαμβάνει υπόψη την πλειοψηφική, γεγονός που μπορεί να αυξήσει την επικάλυψη μεταξύ των κλάσεων [76].

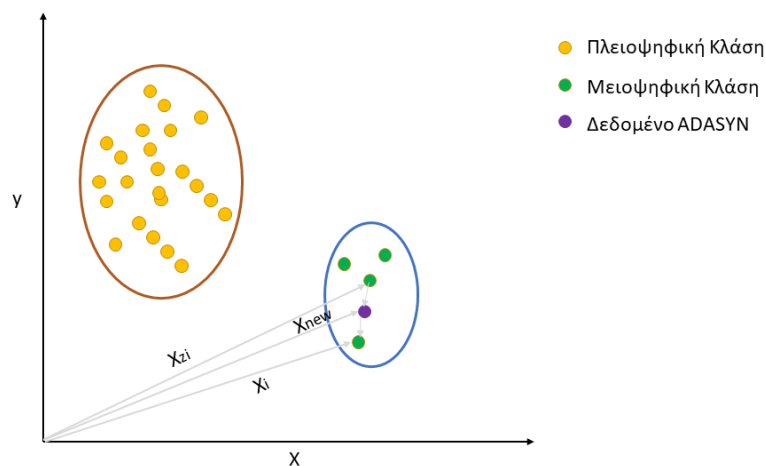


Εικόνα 11 Εφαρμογή της SMOTE στη μειοψηφική κλάση

Στην ιατρική απεικόνιση η εφαρμογή του SMOTE απευθείας σε ακατέργαστες ιατρικές εικόνες δεν θα ήταν αποτελεσματική λόγω της πλούσιας και

πολύπλοκης πληροφορίας που περιέχουν, μια επιτυχημένη προσέγγιση θα ήταν να γίνει πρώτα εξαγωγή χαρακτηριστικών. Αυτό μπορεί να υλοποιηθεί με τεχνικές όπως η ανάλυση της κατανομής της έντασης, η σύλληψη χωρικών μοτίβων για την υφή και ο εντοπισμός συνολικών σχημάτων. Όταν τα χαρακτηριστικά αυτά εξαχθούν τότε ο SMOTE μπορεί να εφαρμοστεί σε αυτόν τον χώρο χαρακτηριστικών, αντί για τα εικονοστοιχεία της αρχικής εικόνας, διασφαλίζοντάς έτσι ότι τα παραγόμενα συνθετικά δείγματα θα διατηρήσουν τις σχετικές πληροφορίες για την συγκεκριμένη εργασία. Μετά την δημιουργία των συνθετικών σημείων δεδομένων στον νέο χώρο χαρακτηριστικών ακολουθεί η ανακατασκευή τους πίσω σε εικόνες. Αυτό μπορεί να επιτευχθεί με την μέθοδο αντίστροφου μετασχηματισμού (inverse transformation) η οποία εκτελεί την τεχνική εξαγωγής χαρακτηριστικών αντίστροφα για την απόκτηση της αρχικής αναπαράστασης.[77], [78]

Μια ακόμα διαδεδομένη τεχνική υπερδειγματοληψίας είναι η προσαρμοστική προσέγγιση συνθετικής δειγματοληψίας (Adaptive Synthetic Sampling Approach) γνωστή στη βιβλιογραφία ως ADASYN (Εικόνα 12). Η ADASYN βασίζεται στην προσαρμοστική δημιουργία δεδομένων μειοψηφίας σύμφωνα με τις κατανομές τους. Τα πιο δύσκολα στην εκμάθηση δείγματα της μειονοτικής κλάσης χρησιμοποιούνται για τη δημιουργία περισσότερων συνθετικών δεδομένων, από τα δείγματα που μαθαίνονται ευκολότερα το οποίο με τη σειρά του οδηγεί στη μείωση της μεροληπτικής μάθησης που είχε εισαχθεί αρχικά λόγω της ανισοκατανομής των δεδομένων [76].



Εικόνα 12 Δημιουργία συνθετικών δεδομένων με την προσέγγιση ADASYN

Σε αντίθεση με τον αλγόριθμο SMOTE, στον οποίο οι αριθμοί των συνθετικών δειγμάτων που παράγονται για κάθε κλάση μειονότητας είναι ίδιοι, στον ADASYN χρησιμοποιείται κατανομή πυκνότητας για να ληφθεί αυτόματα η απόφαση για τον αριθμό συνθετικών δειγμάτων που απαιτούνται να παραχθούν για κάθε ένα δείγμα από την κλάση μειονότητας [76]. Ολοκληρώνοντας, η λογική του αλγορίθμου είναι ότι αποδίδει βάρη σε διαφορετικά δείγματα της μειονοτικής κλάσης προκειμένου να δημιουργηθούν διαφορετικές ποσότητες συνθετικών δεδομένων για κάθε δείγμα [76]. Στην ιατρική απεικόνιση η μέθοδος ADASYN προσδιορίζει την κλάση μειονότητας, η οποία θα μπορούσε να είναι εικόνες που αντιπροσωπεύουν μια σπάνια ασθένεια. Στην συνέχεια επικεντρώνεται στη δημιουργία νέων δεδομένων στην κατηγορία μειονότητας που έχουν παρόμοια χαρακτηριστικά με τα «δύσκολα» υπάρχοντα δεδομένα. Τα «δύσκολα» δεδομένα είναι εκείνα που βρίσκονται κοντά στο όριο απόφασης μεταξύ των κλάσεων, όπου η ταξινόμηση είναι πιο δύσκολη. Η ADASYN εντοπίζει αυτά τα σημεία κάνοντας ανάλυση των χαρακτηριστικών αυτών και των πλησιέστερων γειτόνων τους της κλάση πλειοψηφίας. Με βάση αυτή την ανάλυση, παράγει νέα συνθετικά δεδομένα που μοιράζονται παρόμοια χαρακτηριστικά με τα "δύσκολα" σημεία, αυξάνοντας ουσιαστικά την αντιπροσώπευσή τους στην κλάση μειονότητας.[79]

### **2.3.3 Cost-Sensitive Training**

Η μάθηση με ευαισθησία στο κόστος (Cost-Sensitive Training) είναι ένα υποπεδίο της μηχανικής μάθησης που λαμβάνει υπόψη το κόστος των σφαλμάτων πρόβλεψης (και ενδεχομένως άλλα κόστη) κατά την εκπαίδευση ενός μοντέλου μηχανικής μάθησης. Πρόκειται για ένα πεδίο μελέτης που σχετίζεται στενά με το πεδίο της μάθησης ανισοκατανομής (Imbalanced learning) που ασχολείται με την ταξινόμηση (Classification) σε σύνολα δεδομένων με λοξή κατανομή κλάσεων.

Στην παραδοσιακή μηχανική μάθηση, η έμφαση δίνεται στην ελαχιστοποίηση του σφάλματος ταξινόμησης, η οποία αντιμετωπίζει όλες τις λανθασμένες ταξινομήσεις εξίσου. Ωστόσο, σε πολλές εφαρμογές του πραγματικού κόσμου, η λανθασμένη ταξινόμηση ορισμένων περιπτώσεων μπορεί να είναι πιο δαπανηρή από άλλες. Για παράδειγμα, στην ιατρική διάγνωση, η λανθασμένη ταξινόμηση

ενός ασθενούς με σοβαρή πάθηση ως υγιούς μπορεί να είναι πιο δαπανηρή από τη λανθασμένη ταξινόμηση ενός υγιούς ατόμου ως ασθενή.

Στη μάθηση με ευαισθησία στο κόστος, ο στόχος είναι η ελαχιστοποίηση του συνολικού κόστους της λανθασμένης ταξινόμησης και όχι μόνο του ποσοστού σφάλματος ταξινόμησης. Αυτό επιτυγχάνεται με την απόδοση διαφορετικού κόστους λανθασμένης ταξινόμησης σε διαφορετικές κλάσεις ή περιπτώσεις. Ο πίνακας κόστους χρησιμοποιείται για τον καθορισμό του κόστους εσφαλμένης ταξινόμησης για κάθε κλάση. Για παράδειγμα, σε ένα πρόβλημα ταξινόμησης δύο κλάσεων, ο πίνακας κόστους μπορεί να οριστεί ως εξής:

Πίνακας 1 Πίνακας κόστους

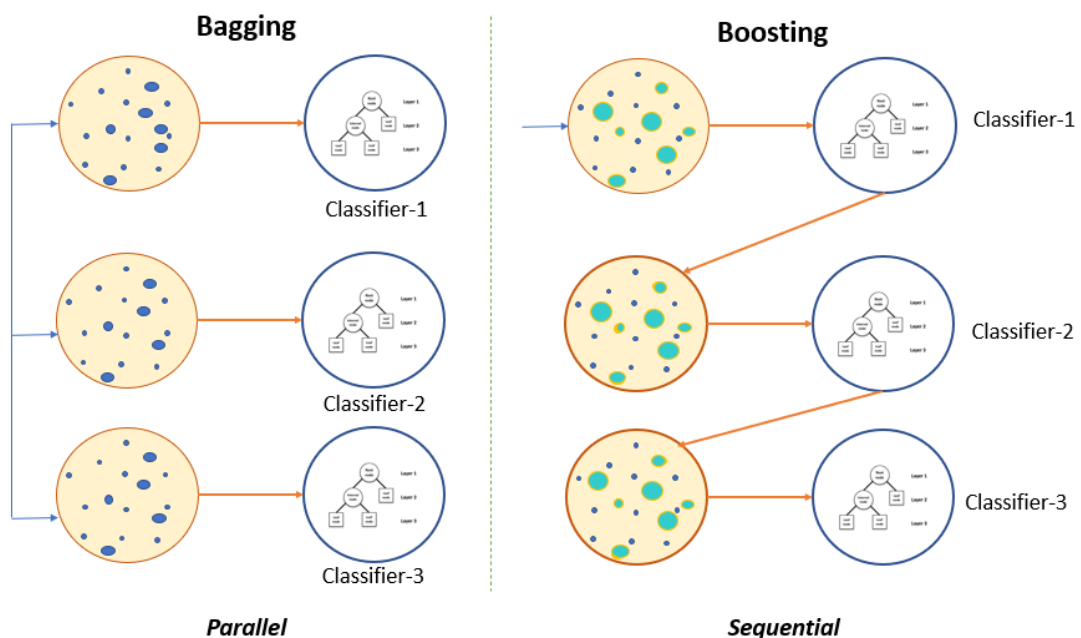
	<b>Πρόβλεψη ως Θετικό</b>	<b>Πρόβλεψη ως Αρνητικό</b>
<b>Πραγματικά Θετικό</b>	ΚΑΘ (Κόστος αληθών θετικών)	ΚΨΑ (Κόστος ψευδώς αρνητικών)
<b>Πραγματικά Αρνητικό</b>	ΚΨΘ(Κόστος ψευδώς θετικών)	ΚΑΑ (Κόστος αληθών αρνητικών)

Σε αυτόν τον πίνακα κόστους (Πίνακας 1), το κόστος των ψευδώς θετικών και των ψευδώς αρνητικών είναι διαφορετικό, αντικατοπτρίζοντας το γεγονός ότι η λανθασμένη ταξινόμηση μιας θετικής περίπτωσης ως αρνητικής (ψευδώς αρνητική) μπορεί να είναι πιο δαπανηρή από τη λανθασμένη ταξινόμηση μιας αρνητικής περίπτωσης ως θετικής (ψευδώς θετική).

Η μάθηση με ευαισθησία στο κόστος μπορεί να εφαρμοστεί σε διάφορους αλγορίθμους μηχανικής μάθησης, όπως τα δέντρα απόφασης (Decision Trees), οι μηχανές διανυσμάτων υποστήριξης (Support Vector Machines) και τα νευρωνικά δίκτυα (Neural Networks). Ο πίνακας κόστους μπορεί να ενσωματωθεί στη διαδικασία μάθησης τροποποιώντας την αντικειμενική συνάρτηση (objective function), προσαρμόζοντας το κατώφλι απόφασης ή χρησιμοποιώντας ειδικούς αλγορίθμους μάθησης που έχουν σχεδιαστεί για μάθηση με ευαισθησία στο κόστος.[80] Ως εκ τούτου, πολλές τεχνικές που αναπτύχθηκαν και χρησιμοποιούνται για τη μάθηση με ευαισθησία στο κόστος μπορούν να υιοθετηθούν για επίλυση προβλημάτων ταξινόμησης ανισοκατανομής [11].

### 2.3.4 Ensemble Models

Τα μοντέλα συνόλου (Ensemble Models) που ανήκουν στην κατηγορία αλγοριθμικών τεχνικών επίλυσης του προβλήματος της ανισορροπίας των κλάσεων αποτελούν ένα πολύ δυνατό εργαλείο αντιμετώπισης του προβλήματος. Τα μοντέλα συνόλου έχουν τη δυνατότητα να μπορούν να συνδυάσουν αποτελέσματα πολλαπλών μοντέλων, καθένα από τα οποία έχει εκπαιδευτεί σε διαφορετικά υποσύνολα δεδομένων ή διαφορετικούς αλγορίθμους με στόχο τη βελτίωση της ικανότητας γενίκευσης και την αύξηση της ακρίβειας (Accuracy) πρόβλεψης. Δυο από τις πιο δημοφιλείς συνδυαστικές τεχνικές είναι η ενίσχυση (boosting) και η σακκοποίηση (bagging).



Εικόνα 13 Bagging vs Boosting

Στην boosting, κάθε ταξινομητής εξαρτάται από τον προηγούμενο και εστιάζει στα σφάλματα αυτού. Παραδείγματα που έχουν ταξινομηθεί λάθος σε προηγούμενους ταξινομητές επιλέγονται πιο συχνά ή σταθμίζονται (weighted) περισσότερο, σε αντίθεση με την bagging, όπου κάθε μοντέλο του συνόλου ψηφίζει με ίση βαρύτητα. Δημιουργεί διαδοχικά μοντέλα, όπου κάθε επόμενο μοντέλο εστιάζει στα λάθη που έκανε το προηγούμενο, ειδικά σε περιοχές με υψηλό ποσοστό λανθασμένης ταξινόμησης της κλάσης μειονότητας. Με αυτόν τον τρόπο εξισορροπεί έμμεσα τις κλάσεις εστιάζοντας στη μάθηση από τις λανθασμένες ταξινομήσεις της μειωτικής κλάσης κατά την διαδικασία της εκπαίδευσης.[81]

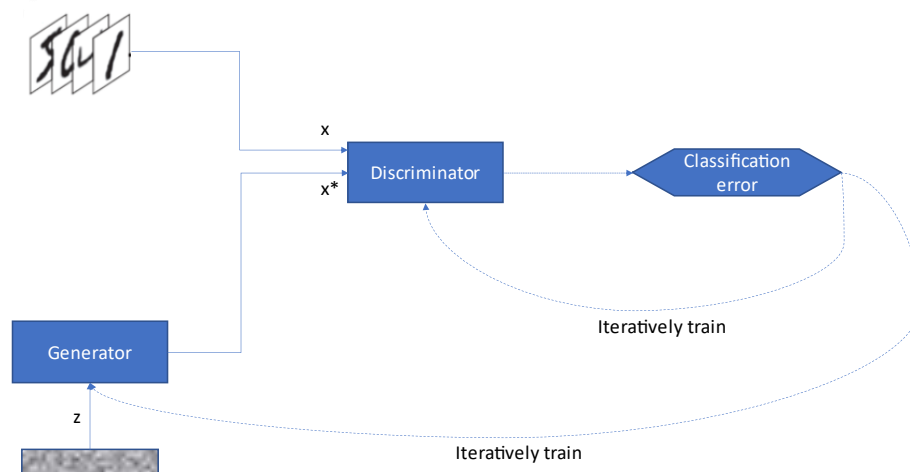
Με σκοπό να προωθηθεί η διακύμανση των μοντέλων, η bagging εκπαιδεύει κάθε μοντέλο στο σύνολο (ensemble), χρησιμοποιώντας ένα τυχαία επιλεγμένο υποσύνολο του συνόλου εκπαίδευσης (training set) με αντικατάσταση. Αυτό αυξάνει την ποικιλομορφία μεταξύ των επιμέρους μαθητών (learners) μειώνοντας την πιθανότητα όλα τα μοντέλα να μεροληπτούν προς την πλειοψηφούσα κλάση. Με την ποικιλομορφία που εισάγεται με την τεχνική bagging το αποτέλεσμα παρουσιάζει καλύτερες επιδόσεις στην κλάση της μειονότητας και μειωμένο κίνδυνο για overfitting στην κλάση πλειοψηφίας. [81] [82]

Τόσο το bagging όσο και το boosting μπορούν να συνδυαστούν με άλλες τεχνικές ειδικά σχεδιασμένες για την αντιμετώπιση της ανισορροπία των κλάσεων, όπως η υπερδειγματοληψία ή η υποδειγματοληψία, για μεγαλύτερη βελτίωση της απόδοσης.

### 2.3.5 Generative Adversarial Networks

Τελευταία τεχνική είναι η χρήση Generative Adversarial Networks (Παραγωγικά Αντιπαλικά Δίκτυα) ή αλλιώς GANs. Τα GANs είναι αλγοριθμικές τεχνικές που αποτελούνται από δύο νευρωνικά δίκτυα τα οποία τοποθετούνται το ένα ενάντια στο άλλο προκειμένου να δημιουργήσουν νέα συνθετικά παραδείγματα τα οποία δεν θα μπορούν να διακριθούν από τα πραγματικά.

Στην Εικόνα 14 φαίνεται ένα απλό GAN.



Εικόνα 14 Η δομή λειτουργίας ενός δικτύου GAN

## 2.4 Generative Adversarial Networks (Παραγωγικά Αντιπαλικά Δίκτυα)

### 2.4.1 Εισαγωγή στα GANs

Μια από τις πιο υποσχόμενες ανακαλύψεις τα τελευταία χρόνια στον τομέα της Βαθιάς μάθησης είναι τα Generative Adversarial Networks (Παραγωγικά Αντιπαλικά Δίκτυα) ή GANs. Η ιδέα πίσω από τα GANs προτάθηκε πρώτη φορά από τον Ian Goodfellow et al. σε μια έρευνα το 2014 [48]. Η συγκεκριμένη έρευνα πρότεινε μια νέα προσέγγιση στην παραγωγική μοντελοποίηση που μπορεί να μάθει να παράγει νέα δεδομένα παρόμοια με τα δεδομένα εκπαίδευσης χρησιμοποιώντας όχι ένα, αλλά δύο ξεχωριστά νευρωνικά δίκτυα. Επίσης, εισήγαγε διάφορες παραλλαγές των GANs, όπως τα Conditional GANs (CGANs) τα οποία επιτρέπουν τη δημιουργία δειγμάτων που εξαρτώνται από συγκεκριμένες πληροφορίες εισόδου. Ακόμη, υπάρχουν τα Deep Convolutional GANs (DCGANs), τα οποία χρησιμοποιούν νευρωνικά δίκτυα συνελιξων (CNN) για τη βελτίωση της σταθερότητας και της ποιότητας των παραγόμενων εικόνων. Αξίζει να σημειωθεί ότι τα GAN δεν ήταν το πρώτο πρόγραμμα υπολογιστή που χρησιμοποιήθηκε για τη δημιουργία δεδομένων, αλλά τα αποτελέσματά τους και η ευελιξία τους τα ξεχωρίζουν από όλα τα υπόλοιπα. Πριν από αυτά, υπήρχαν αρκετά παραγωγικά (generative) μοντέλα για αρκετές δεκαετίες. Τα πρώτα από αυτά βασιζόνταν σε ρητές συναρτήσεις πυκνότητας πιθανότητας, όπως τα μοντέλα μίξης Gauss και τα κρυφά μοντέλα Markov (Hidden Markov Models). Ωστόσο, αυτά τα μοντέλα περιορίζονταν από την ικανότητά τους να αποτυπώνουν τις πολύπλοκες, υψηλών διαστάσεων κατανομές δεδομένων σε πραγματικές εφαρμογές. Στη συνέχεια, αναπτύχθηκαν νέοι τύποι παραγωγικών μοντέλων, όπως οι μεταβλητοί αυτοκωδικοποιητές (Variational Autoencoders ή VAE) και οι μηχανές Boltzman. Αυτά τα μοντέλα παρόλο που ανήκουν στην κατηγορία των παραγωγικών μοντέλων διαφέρουν από τα GAN καθώς π.χ. οι VAEs μαθαίνουν να αναδημιουργούν την πληροφορία που λαμβάνουν ως είσοδο, ενώ τα GANs μαθαίνουν να δημιουργούν νέα δεδομένα τα οποία είναι τόσο ρεαλιστικά όσο τα δεδομένα εκπαίδευσης. Από αυτό και μόνο το γεγονός γίνεται αντιληπτή η μοναδικότητά τους.





Εικόνα 15 Η ραγδαία εξέλιξη των GANs από το 2014 μέχρι το 2017

Τα αποτελέσματα που έχουν πετύχει τα GANs είναι εντυπωσιακά, καθώς σε πράγματα τα οποία για πολύ μεγάλο διάστημα θεωρούνταν αδύνατον να επιτευχθούν από συστήματα τεχνητής νοημοσύνης κατάφεραν να υλοποιηθούν. Τέτοια παραδείγματα είναι η ικανότητα να δημιουργούν ψεύτικες εικόνες με ποιότητα που μοιάζει με του πραγματικού κόσμου, η μετατροπή μιας μουτζούρας σε μια εικόνα που μοιάζει με φωτογραφία ή η μετατροπή ενός βίντεο ενός αλόγου σε τρέξιμο μιας ζέβρας - και όλα αυτά χωρίς την ανάγκη τεράστιου όγκου δεδομένων εκπαίδευσης που απαιτούν κόπο για τη συλλογή τους [83]. Στην Εικόνα 15 παρουσιάζεται η ραγδαία εξέλιξη των GANs μέσα σε διάστημα δυο χρονών.

## 2.4.2 Δομή και Λειτουργία

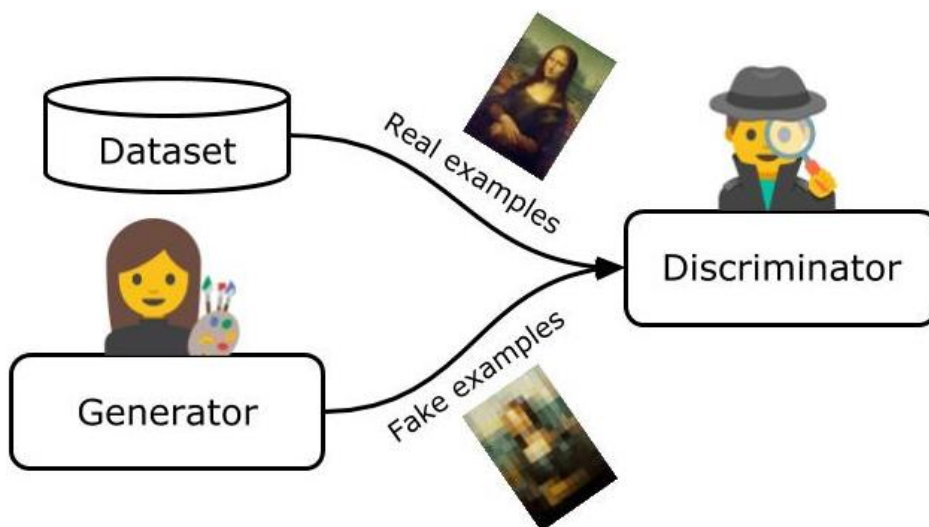
### Βασική Δομή

Τα Generative Adversarial Networks (GANs) αποτελούνται από δύο ταυτόχρονα εκπαιδευόμενα μοντέλα, το ένα ονομάζεται Generator (γεννήτρια) και εκπαιδεύεται με σκοπό να παράγει ψεύτικα δεδομένα, και το άλλο ονομάζεται Discriminator (διαχωριστής) το οποίο εκπαιδεύεται με σκοπό να διακρίνει τα ψεύτικα δεδομένα από τα πραγματικά [83].

Η λέξη *generative* υποδηλώνει τον γενικό σκοπό του μοντέλου ο οποίος δεν είναι άλλος από τη δημιουργία νέων δεδομένων, ενώ η λέξη αντίπαλος (*adversarial*) παραπέμπει στην παιγνιώδη, ανταγωνιστική δυναμική μεταξύ των δύο μοντέλων που αποτελούν το πλαίσιο GAN. Τα δεδομένα που θα μάθει να παράγει ένα GAN εξαρτώνται από την επιλογή του συνόλου εκπαίδευσης [83]. Έτσι αν για παράδειγμα, θέλουμε ένα GAN να συνθέτει εικόνες που μοιάζουν με πίνακες του

Leonardo da Vinci, θα χρησιμοποιηθεί ένα σύνολο δεδομένων εκπαίδευσης με έργα τέχνης του da Vinci [83].

Ο στόχος της γεννήτριας είναι να δημιουργήσει παραδείγματα που δεν διακρίνονται από τα πραγματικά δεδομένα του συνόλου εκπαίδευσης. Στο ανωτέρω παράδειγμα συνεπώς παράγονται πίνακες ζωγραφικής που μοιάζουν ακριβώς με αυτούς του da Vinci, ενώ ο στόχος του Discriminator είναι να διακρίνει τα ψεύτικα που παράγει η γεννήτρια από τα πραγματικά παραδείγματα που προέρχονται από το σύνολο δεδομένων εκπαίδευσης. Συνεπώς, ο Discriminator παίζει το ρόλο ενός εμπειρογνώμονα τέχνης που αξιολογεί τη γνησιότητα των πινάκων που πιστεύεται ότι είναι του da Vinci (Εικόνα 16). Τα δύο δίκτυα προσπαθούν συνεχώς να ξεγελάσουν το ένα το άλλο δηλαδή όσο καλύτερη γίνεται η γεννήτρια στη δημιουργία πειστικών δεδομένων, τόσο καλύτερος πρέπει να είναι και ο διαχωριστής στη διάκριση των πραγματικών παραδειγμάτων από τα ψεύτικα [83].



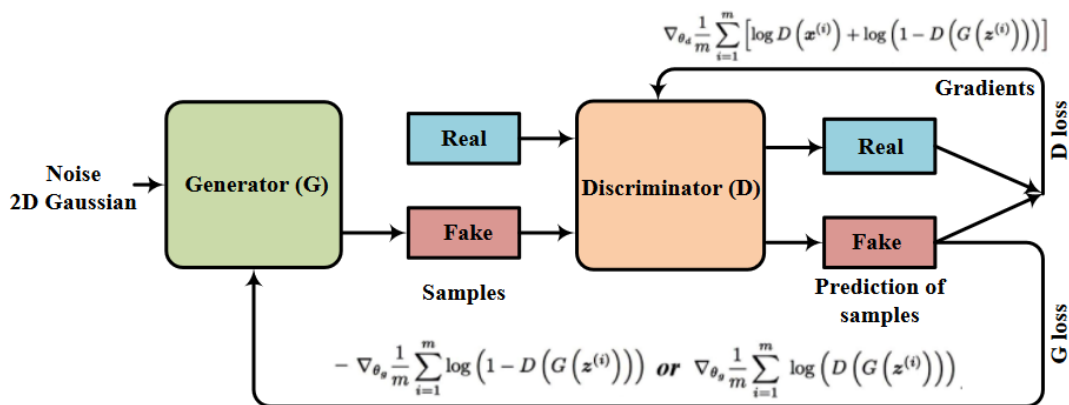
Εικόνα 16 Η παραγόμενη εικόνα από τον generator δίνεται στο discriminator μαζί με μία πραγματική και αυτός με τη σειρά του αποφασίζει αν είναι αληθινή η ψεύτικη

### *Τρόπος Λειτουργίας των GANs*

Ένα GAN αποτελείται από δύο μέρη τη γεννήτρια (Generator) και τον διαχωριστή (Discriminator). Ο στόχος του Generator είναι να παράγει δείγματα τα οποία αποτυπώνουν τα χαρακτηριστικά του συνόλου εκπαίδευσης, τόσο καλά έτσι ώστε τα δείγματα που παράγονται να μην ξεχωρίζουν από τα δεδομένα εκπαίδευσης. Μπορεί να θεωρηθεί ότι ο Generator λειτουργεί σαν μοντέλο αναγνώρισης αντικειμένων αλλά από την ανάποδη. Οι αλγόριθμοι αναγνώρισης

αντικειμένων μπορούν και μαθαίνουν μοτίβα σε εικόνες με στόχο τη διάκριση του περιεχομένου τους. Ο Generator, αντί να αναγνωρίζει μοτίβα σε μια εικόνα, μαθαίνει να τα δημιουργεί από το μηδέν και πράγματι τις περισσότερες φορές η είσοδος στον Generator δεν είναι πάρα ένα διάνυσμα αριθμών [83].

Ο Generator μαθαίνει από την ενημέρωση (feedback) που λαμβάνει από τις ταξινομήσεις του Discriminator. Ο στόχος του Discriminator δεν είναι άλλος από το να αποφασίσει αν ένα συγκεκριμένο παράδειγμα είναι πραγματικό (που προέρχεται από το σύνολο δεδομένων εκπαίδευσης) ή ψεύτικο (παραγόμενο από τον Generator). Συνεπώς, κάθε φορά που ο Discriminator ξεγελιέται και ταξινομεί μια ψεύτικη εικόνα ως πραγματική, ο Generator ξέρει ότι έκανε κάτι καλό. Από την άλλη, κάθε φορά που ο Discriminator ταξινομεί μια εικόνα που έχει παραχθεί από τον Generator ως ψεύτικη, τότε ο Generator λαμβάνει ενημέρωση ότι χρειάζεται περαιτέρω βελτίωση. Ο Discriminator συνεχίζει να βελτιώνεται όπως ακριβώς θα έκανε οποιοσδήποτε ταξινομητής, μαθαίνοντας δηλαδή ανάλογα από το πόσο μακριά είναι οι προβλέψεις του από τις πραγματικές ετικέτες (πραγματικές η ψεύτικες). Επομένως, καθώς ο Generator βελτιώνεται στην παραγωγή δεδομένων που μοιάζουν ρεαλιστικά, ταυτόχρονα βελτιώνεται και ο Discriminator στον διαχωρισμό των ψεύτικων [83].



Εικόνα 17 Η βασική λειτουργία ενός GAN

Από την οπτική της επιστήμης των μαθηματικών, τα GANs προσπαθούν να αναπαράγουν μια κατανομή πιθανοτήτων. Επομένως, για να πετύχουν τον σκοπό τους χρησιμοποιούν συναρτήσεις απώλειας (Loss functions) που αντικατοπτρίζουν την απόσταση μεταξύ της παραγόμενης κατανομής δεδομένων ( $P_g$ ) και την πραγματική κατανομή δεδομένων ( $P_{data}$ ). Οι συναρτήσεις απωλειών αποτελούν ένα τεράστιο πεδίο ερευνάς και μελέτης στον κόσμο των

GANs και οι προσεγγίσεις που ασχολούνται με τον υπολογισμό της διαφοράς αυτών των δυο κατανομών ποικίλουν. Στην Εικόνα 17 φαίνεται η βασική λειτουργία ενός GAN καθώς και οι δυο συναρτήσεις κόστους Discriminator Loss και Generator Loss, οι οποίες συνδυάζοντας την Binary Cross-Entropy Loss καθορίζουν όλη τη διαδικασία της εκπαίδευσης σε ένα minmax παιχνίδι.

## 2.4.3 Εκπαίδευση-Minimax Game

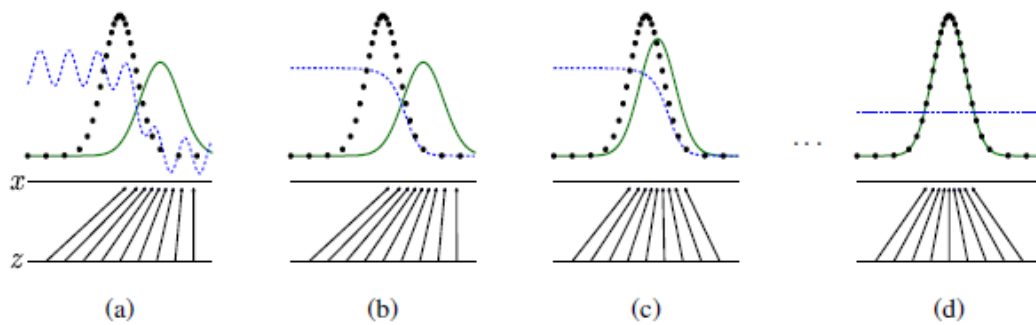
### *Minimax Loss*

Η απώλεια minmax, επίσης γνωστή ως αντιπαλική απώλεια (adversarial loss), αποτελεί βασικό συστατικό των GANs και αναφέρθηκε για πρώτη φορά στη μελέτη του Ian Goodfellow το 2014 [48] αποτελώντας την αντικειμενική συνάρτηση που ορίζει το παιχνίδι μεταξύ του Generator και του Discriminator. Η μαθηματική έκφραση της συνάρτησης είναι η παρακάτω:

$$\min_G \max_D V(D, G) = E_{x \sim p_{data}(x)} [\log(D(x))] + E_{z \sim p_z(z)} [\log(1 - D(G(z)))] \quad (3)$$

Ο Generator προσπαθεί να ελαχιστοποιήσει αυτή τη συνάρτηση, ενώ ο Discriminator προσπαθεί να τη μεγιστοποιήσει. Ο πρώτος όρος στη συνάρτηση απώλειας μετράει πόσο καλά ο discriminator είναι σε θέση να διακρίνει μεταξύ πραγματικών και ψεύτικων δεδομένων, ενώ ο δεύτερος όρος μετρά πόσο καλά ο generator μπορεί να ξεγελάσει τον διαχωριστή, ώστε να νομίζει ότι τα παραγόμενα δεδομένα είναι πραγματικά.

Κατά τη διάρκεια της εκπαίδευσης, τα δίκτυα Generator και Discriminator ενημερώνονται εναλλάξ για να βρεθεί μια ισορροπία Nash (Nash equilibrium), δηλαδή το σημείο όπου κανένας από τους δύο παίκτες δεν μπορεί να βελτιώσει το σκορ του αλλάζοντας τη στρατηγική του. Αυτή η ισορροπία επιτυγχάνεται όταν ο Generator παράγει δεδομένα που δεν διακρίνονται από τα πραγματικά δεδομένα και ο Discriminator δίνει πιθανότητα 0,5 τόσο για τα πραγματικά όσο και για τα ψεύτικα δεδομένα (Εικόνα 18 (d)).



Εικόνα 18 Το minmax game ενός GAN μέχρι να φτάσει σε ισορροπία Nash [48]

Στην Εικόνα 18 αποτυπώνεται σχηματικά τι συμβαίνει κατά το στάδιο της εκπαίδευσης στο minmax game του GAN. Στο στάδιο (a), οι δυο κατανομές  $P_g$  (πράσινη γραμμή) και  $P_{data}$  (μαύρη διακεκομμένη γραμμή) δεν έχουν έρθει ακόμα σε σημείο που να μη διαφέρουν, επομένως η εκπαίδευση συνεχίζεται. Στο στάδιο (b), οι δυο κατανομές έχουν έρθει πιο κοντά η μια με την άλλη και ο Discriminator είναι σε σημείο που μπορεί να διακρίνει τα αληθινά δεδομένα από τα ψεύτικα. Στο στάδιο (c) έχουν έρθει ακόμα πιο κοντά τόσο ώστε ο Discriminator αρχίζει να δυσκολεύεται αρκετά στο να αποφασίσει. Στο στάδιο (d) οι δυο κατανομές ταυτίζονται αρά έχουμε πέτυχει να ισχύει  $p_g = p_{data}$  δηλαδή μια ισορροπία Nash.

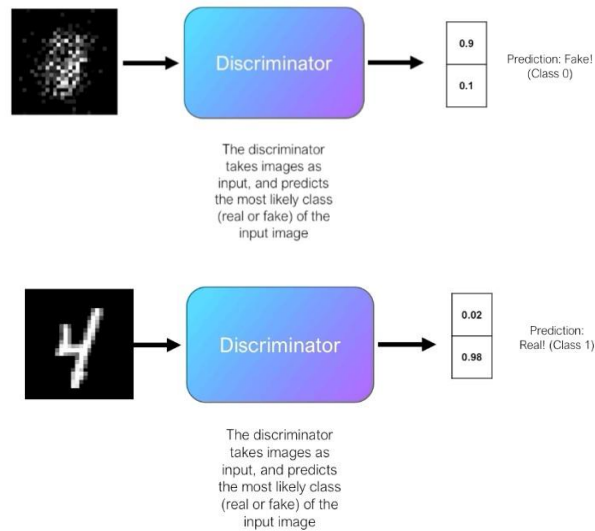
#### *Binary Cross Entropy Loss στα GANs*

Εκτός από την απώλεια minmax για να παράγει ένα GAN νέα συνθετικά δεδομένα χρειάζεται και η συνάρτηση απώλειας Binary Cross Entropy (BCE). Αυτή χρησιμοποιείται συνήθως στα GAN ως συνάρτηση απώλειας για την εκπαίδευση του δικτύου του Discriminator. Η BCE του Discriminator αποτελεί ένα πολύ σημαντικό σήμα σε ένα GAN κυρίως για τον Generator, καθώς τον βοηθάει να καταλάβει αν η εικόνα που δημιούργησε μοιάζει με τις πραγματικές. Θυμίζουμε ότι ο generator από μόνος του δεν έχει την ικανότητα να το κρίνει αυτό.

*Ο τρόπος λειτουργίας της μέσα στο GAN έχει ως εξής:*

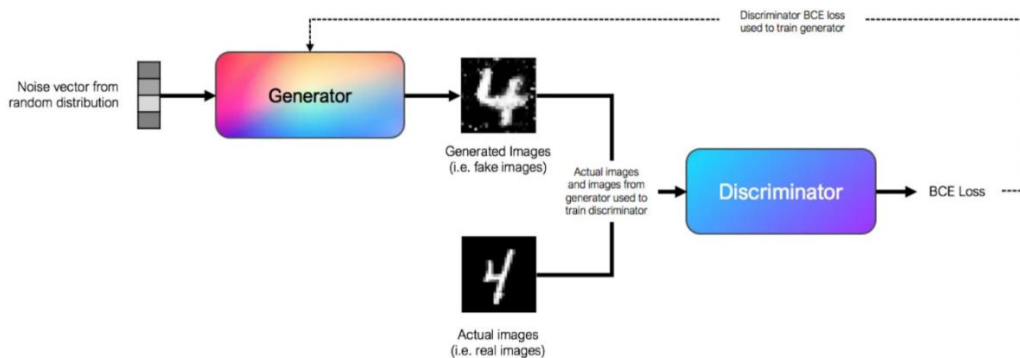
Αφού ο Generator δημιουργήσει μια νέα εικόνα, στέλνεται στον Discriminator και αυτός με τη σειρά του υπολογίζει την πιθανότητα η εικόνα να είναι αληθινή. Όταν η εικόνα που δημιούργησε ο generator δεν είναι καλή, ο discriminator ευκολά θα την ταξινομήσει ως ψεύτικη, οδηγώντας σε χαμηλή BCE. Ωστόσο, καθώς ο generator βελτιώνεται ο discriminator αρχίζει να κάνει περισσότερα

λάθη ταξινομώντας ψεύτικες εικόνες ως αληθινές, που οδηγεί σε υψηλή BCE. Επομένως, η BCE του Discriminator σηματοδοτεί την ποιότητα της εικόνας που παράγεται από τον Generator, ο οποίος ουσιαστικά προσπαθεί να μεγιστοποιήσει αυτή την απώλεια (Εικόνα 19).



Εικόνα 19 Υπολογισμός απώλειας BCE

Συνοψίζοντας όλα τα παραπάνω, δίνεται μια σύντομη περιγραφή της διαδικασίας εκπαίδευσης στην Εικόνα 20.



Εικόνα 20 Διαδικασία Εκπαίδευσης GAN και η χρήση της BCE

Αρχικά ένα GAN αποτελείται από δυο συνδεδεμένα δίκτυα, το δίκτυο του Generator και το δίκτυο του Discriminator. που στις περισσότερες περιπτώσεις επρόκειτο για feedforward νευρωνικά δίκτυα. Τα δίκτυα του Generator και του Discriminator έχουν το χαρακτηριστικό ότι εκπαιδεύονται εναλλάξ, παλεύοντας το ένα έναντι του άλλου.

Για την εκπαίδευση του generator, χρησιμοποιείται ένα διάνυσμα θορύβου από μια τυχαία κατανομή ως είσοδο. Συνήθως το διάνυσμα αυτό έχει μήκος ίσο με

100 και έχει επιλεγεί από την κανονική κατανομή ή κατανομή Gauss. Το διάνυσμα αυτό που αποτελεί την είσοδο περνάει από μια σειρά από πλήρως συνδεδεμένα επίπεδα στο feedforward TND. Η έξοδος που προκύπτει από τον generator είναι μια εικόνα. Για λόγους ευκολίας γίνεται υπόθεση ότι επρόκειτο για μια εικόνα από το MNIST dataset με διάσταση 28x28. Ο generator περνάει την εικόνα που δημιούργησε στο δίκτυο του discriminator και χρησιμοποιεί την BCE (Discriminator's BCE loss), για να βελτιστοποιήσει τα βάρη του, στοχεύοντας στη μεγιστοποίηση της τιμής της απώλειας αυτής. Όσον αφορά την εκπαίδευση του Discriminator, ως είσοδο χρησιμοποιούνται εικόνες με ετικέτες από τον Generator καθώς και πραγματικές εικόνες. Ο Discriminator μαθαίνει να ταξινομεί τις εικόνες ως πραγματικές ή ψεύτικες και εκπαιδεύεται κάνοντας χρήση της BCE. Πρακτικά η εκπαίδευση των δυο δικτύων γίνεται εναλλάξ. Το είδος αυτής της εκπαίδευσης μοιάζει με ένα αντιπαλικό παιχνίδι minimax δυο παικτών, καθώς ο Generator στοχεύει στη μεγιστοποίηση της απώλειας του Discriminator και ο Discriminator στοχεύει στην ελαχιστοποίηση της δικής του απώλειας.

#### **2.4.4 Περιορισμοί των GANs**

Τα Generative Adversarial Networks (GANs) έχουν φέρει επανάσταση στον τομέα της γενετικής μοντελοποίησης, επιτρέποντας τη δημιουργία ρεαλιστικών συνθετικών δεδομένων. Ωστόσο, παρά τα εκπληκτικά αποτελέσματα ενδέχεται να αντιμετωπίσουν αρκετές προκλήσεις κατά το στάδιο της εκπαίδευσης και της εφαρμογής. Μερικά από τα πιο γνωστά ζητήματα που μπορεί να αντιμετωπίσουν τα GANs είναι η κατάρρευση των τρόπων λειτουργίας (Mode Collapse) και η εξαφάνιση των κλίσεων (Vanishing Gradient). Για να αντιμετωπιστούν τα συγκεκριμένα ζητήματα απαιτείται προσεκτική εξέταση της αρχιτεκτονικής του μοντέλου, της διαδικασίας εκπαίδευσης και των υπερπαραμέτρων.

##### *Mode Collapse*

Η κατάρρευση του τρόπου λειτουργίας (Mode Collapse) είναι ένα πολύ γνωστό ζήτημα στα GAN, όπου ο Generator παράγει ένα περιορισμένο σύνολο εξόδων που δεν καταφέρνουν να συλλάβουν το πλήρες εύρος της κατανομής των πραγματικών δεδομένων. Αντί να παράγει ποικίλα δείγματα, επικεντρώνεται στην παραγωγή ενός μικρού υποσυνόλου των πιο κοινών μοτίβων στα δεδομένα, οδηγώντας σε απώλεια μεταβλητότητας στις παραγόμενες εξόδους. Στην Εικόνα

21 φαίνεται ένα παράδειγμα Mode Collapse καθώς το GAN παράγει το ίδιο πρόσωπο ξανά και ξανά, ενώ ιδανικά θα ήταν προτιμότερο να παράγει διαφορετικό πρόσωπο για κάθε τυχαία είσοδο [84].



Εικόνα 21 Mode Collapse παράδειγμα

Μερικοί τρόποι για την αντιμετώπιση του προβλήματος είναι τεχνικές κανονικοποίησης, όπως η αποσύνθεση βάρους (weight decay), η απομάκρυνση (dropout) ή η κανονικοποίηση δέσμης (batch normalization) που μπορούν να βοηθήσουν στην αποτροπή του overfitting του Generator σε ένα περιορισμένο σύνολο μοτίβων στα δεδομένα. Η κανονικοποίηση βοηθάει τον Generator να μάθει ποικίλα και αντιπροσωπευτικά χαρακτηριστικά των δεδομένων, μειώνοντας τον κίνδυνο κατάρρευσης του τρόπου λειτουργίας. Ακόμη, το πρόβλημα μπορεί να αντιμετωπιστεί με τη χρήση εναλλακτικών συναρτήσεων απώλειας, όπως η Wasserstein Loss η Least Square Loss, που παρέχουν πιο σταθερές και κατατοπιστικές κλίσεις τόσο στο Generator όσο και στον Discriminator. Με τη χρήση αυτών των, ο Discriminator μπορεί να καθοδηγήσει καλύτερα τον Generator προς ένα πιο ποικιλόμορφο σύνολο εξόδων.

#### Vanishing Gradient

Ένα δεύτερο ζήτημα που συναντάται στον κόσμο των GANs είναι το Vanishing Gradient. Όταν ο Discriminator είναι πολύ καλός, τότε η εκπαίδευση του Generator μπορεί να αποτύχει λόγω του Vanishing Gradient. Αυτό συμβαίνει γιατί αν ο Discriminator είναι τέλειος η πληροφορίες που παρέχει δεν είναι αρκετές έτσι ώστε ο Generator να σημειώσει πρόοδο [84].



Κατά τη διαδικασία του backpropagation χρησιμοποιείται ο κανόνας της αλυσίδας της διαφοροποίησης που έχει πολλαπλασιαστικό αποτέλεσμα. Στη συνέχεια η κλίση ρέει προς τα πίσω, από το τελευταίο επίπεδο έως το πρώτο. Όσο πηγαίνει προς τα πίσω τόσο μικρότερη γίνεται. Κάποιες φορές η κλίση είναι τόσο μικρή που τα αρχικά επίπεδα μαθαίνουν πολύ αργά ή σταματούν εντελώς να μαθαίνουν. Τότε, η κλίση δεν αλλάζει καθόλου τις τιμές των βαρών των αρχικών επιπέδων και η εκπαίδευση των αρχικών επιπέδων του δικτύου ουσιαστικά σταματά (πρόβλημα του Vanishing Gradient).

Οι λύσεις που έχουν προταθεί για την αντιμετώπιση του προβλήματος είναι η χρήση συναρτήσεων ενεργοποίησης που δεν υποφέρουν από κορεσμό (saturation), όπως η ReLU ή LeakyReLU αντί για τις sigmoid ή tanh περιορίζοντας τιμές εισόδου στο διάστημα 0 και 1 ή -1 και 1 αντίστοιχα έχοντας ως αποτέλεσμα την εκθετική μείωση της κλίσης. Επίσης, όπως και στο πρόβλημα του mode collapse, έτσι και εδώ μια λύση μπορεί να είναι οι τεχνικές κανονικοποίησης όπως η κανονικοποίηση δέσμης (batch normalization) και η κανονικοποίηση επιπέδου (layer normalization), οι οποίες μπορούν να βοηθήσουν στον περιορισμό του προβλήματος κανονικοποιώντας την είσοδο σε κάθε επίπεδο. Τέλος, η Wasserstein Loss η οποία μπορεί να μην λύνει ολοκληρωτικά το πρόβλημα, ωστόσο μπορεί να βοηθήσει στον περιορισμό του, καθώς ενθαρρύνει τον Discriminator να εξαγάγει συνεχείς ομαλές τιμές και να βοηθάει στην αποφυγή της εξαφάνισης των κλίσεων του Generator κατά τη διάρκεια της εκπαίδευσης.

## **2.4.5 Αρχιτεκτονικές GANs**

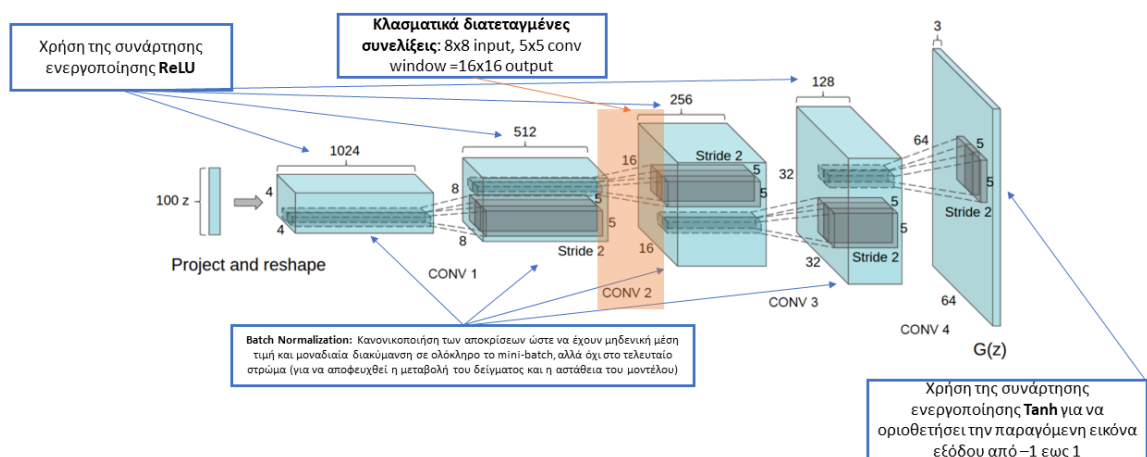
Τα δυο ζητήματα που αναφέρθηκαν παραπάνω συναντώνται αρκετά συχνά στα GANs. Στη συνέχεια, θα παρουσιαστούν κάποιες αρχιτεκτονικές που βοηθάνε στην επίλυση ή τον περιορισμό των παραπάνω ζητημάτων και βελτιστοποιούν τη διαδικασία της εκπαίδευσης έτσι ώστε να πετύχουν το καλύτερο αποτέλεσμα.

### **2.4.5.1 DCGAN**

Μια πολύ διαδεδομένη αρχιτεκτονική των GANs είναι τα Deep Convolutional Adversarial Networks ή DCGANs που έγιναν γνωστά με την έρευνα των Radford και Metz [50] και επρόκειτο για μια παραλλαγή των GANs που χρησιμοποιεί CNN (Convolutional Neural Network), όπως υποδηλώνει και το όνομά τους.

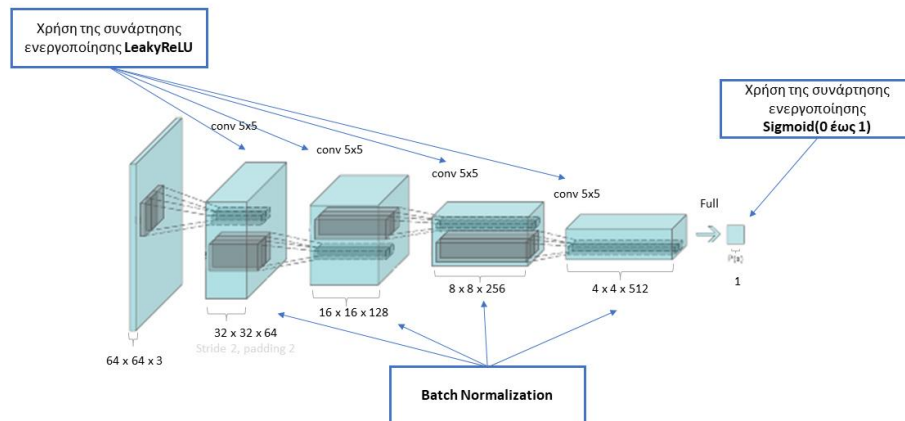
Τα DCGANs παράγουν καλύτερα και πιο σταθερά αποτελέσματα εκπαίδευσης όταν ένα πλήρως συνδεδεμένο επίπεδο αντικαθίσταται από ένα πλήρως συνελικτικό επίπεδο. Όσον αφορά τον σχεδιασμό τους αποτελούνται κυρίως από επίπεδα συνέλιξης χωρίς max pooling ή πλήρως συνδεδεμένα επίπεδα και χρησιμοποιούν συνελικτικό βήμα (convolutional stride) και μετατοπισμένη συνέλιξη (transposed convolutional) για το downsampling και το upsampling. Επίσης, χρησιμοποιείται κανονικοποίηση παρτίδας (Batch Normalization) [85] σε όλα τα επίπεδα εκτός από το επίπεδο εξόδου για τον Generator και το επίπεδο εισόδου για τον Discriminator, η οποία σταθεροποιεί τη μάθηση κανονικοποιώντας την είσοδο σε κάθε μονάδα, ώστε να έχει μηδενική μέση τιμή και μοναδιαία διακύμανση. Αυτό βοηθά στην αντιμετώπιση προβλημάτων εκπαίδευσης που προκύπτουν λόγω κακής αρχικοποίησης και βοηθά τη ροή κλίσης σε βαθύτερα μοντέλα. Τέλος, γίνεται χρήση της συνάρτησης ενεργοποίησης ReLU σε όλα τα επίπεδα του Generator εκτός του επιπέδου εξόδου στο οποία χρησιμοποιείται η Tanh [οριοθετημένη από 0-1], καθώς παρατηρήθηκε ότι η χρήση οριοθετημένης ενεργοποίησης επιτρέπει στο μοντέλο να κορεστεί πιο γρήγορα και να καλύψει τον χρωματικό χώρο της κατανομής εκπαίδευσης. Στο δίκτυο του discriminator σε όλα τα επίπεδα χρησιμοποιείται η Leaky ReLU, καθώς αποδείχθηκε μετά από δοκιμές ότι για μοντελοποίηση υψηλότερης ανάλυσης είχε αρκετά καλά αποτελέσματα σε σχέση με την ενεργοποίηση maxout που συναντάται στη μελέτη του Goodfellow [48], [50].

### a) Generator



Εικόνα 22 Παράδειγμα δικτύου Generator μέσα σε ένα DCGAN

## b) Discriminator



Εικόνα 23 Παράδειγμα δικτύου Discriminator μέσα σε ένα DCGAN

Στην Εικόνα 22 και Εικόνα 23 φαίνεται πως μοιάζει το δίκτυο του Generator και του Discriminator αντίστοιχα μέσα σε ένα DCGAN. Το δίκτυο του Generator (Εικόνα 22) δέχεται ένα διάνυσμα θορύβου  $100 \times 1$ , και αφού περάσει από τα επίπεδα συνελίξεων καταλήγει στην έξοδο που είναι  $64 \times 64 \times 3$ . Αυτό είναι αρκετά ενδιαφέρον για τον τρόπο που το πρώτο επίπεδο επεκτείνει τον τυχαίο θόρυβο από  $100 \times 1$  σε  $1024 \times 4 \times 4$ . Αυτό το επίπεδο είναι γνωστό ως *προβολή και αναδιαμόρφωση (project and reshape)*. Στη συνέχεια μετά από αυτό το επίπεδο, εφαρμόζονται κλασικά συνελκτικά επίπεδα, τα οποία αναδιαμορφώνουν το δίκτυο με βάση την παρακάτω εξίσωση:

$$(N+P-F)/S + 1 \quad (4)$$

Στο παραπάνω διάγραμμα (Εικόνα 22) φαίνεται ότι η παράμετρος  $N$ , (Ύψος/Πλάτος), πηγαίνει από 4 σε 8 από 8 σε 16 από 16 σε 32. Δε φαίνεται να υπάρχει κάποιο padding, η παράμετρος του φίλτρου πυρήνα  $F$  είναι  $5 \times 5$  και το stride είναι 2. Αυτή η εξίσωση είναι χρήσιμη για τη σχεδίαση εξατομικευμένων συνεκτικών επιπέδων για προκαθορισμένα μεγέθη εξόδου. Η πορεία του δικτύου του Generator ανά επίπεδο ακολουθεί την παρακάτω σειρά:

$$100 \times 1 \rightarrow 1024 \times 4 \times 4 \rightarrow 512 \times 8 \times 8 \rightarrow 256 \times 16 \times 16 \rightarrow 128 \times 32 \times 32 \rightarrow 64 \times 64 \times 3$$

Όσον αφορά το δίκτυο του Discriminator (Εικόνα 23), αυτό λειτουργεί σαν δυαδικός ταξινομητής που παίρνει μια εικόνα εισόδου και στην έξοδο δίνει μια κλιμακωτή πιθανότητα για το αν η εικόνα είναι ψεύτικη η αληθινή. Στο παράδειγμα αυτό, παίρνει μια εικόνα  $64 \times 64 \times 3$  και αφού την περάσει από όλα τα συνελκτικά επίπεδα, χρησιμοποιώντας Batch normalization και την Leaky Relu

στα σημεία που χρειάζεται, παράγει το τελικό αποτέλεσμα δηλαδή την πιθανότητα που αναφέρθηκε μέσω της συνάρτησης ενεργοποίησης Sigmoid.

#### **2.4.5.2 WGAN**

Τα WGAN (Wasserstein Generative Adversarial Network), αποτελούν έναν τύπο γεννητικού μοντέλου που χρησιμοποιεί την απόσταση Wasserstein για την εκπαίδευση του δικτύου. Παρουσιάστηκαν για πρώτη φορά το 2017 από τους Arjovsky et al. [51] σε μια βελτίωση της παραδοσιακής GAN αρχιτεκτονικής. Τα WGAN βοηθάνε στην αντιμετώπιση των παραπάνω ζητημάτων καθώς προσφέρουν μεγαλύτερη σταθερότητα κατά το στάδιο της εκπαίδευσης του μοντέλου σε σχέση με τις απλές αρχιτεκτονικές. Επίσης σε αντίθεση με τις παραδοσιακές αρχιτεκτονικές, η συνάρτηση απώλειας στα WGAN αποτελεί ένα κριτήριο τερματισμού για την αξιολόγηση του μοντέλου [86].

Αξίζει να σημειωθεί επίσης ότι στην αρχιτεκτονική WGAN, ο discriminator παίρνει το όνομα critic (κριτής). Ο λόγος που παίρνει αυτό το όνομα είναι επειδή δεν υπάρχει σιγμοειδής συνάρτηση ενεργοποίησης που να περιορίζει τις τιμές από 0 ή 1, δηλαδή ψεύτικο ή αληθινό, αλλά, το δίκτυο του discriminator-critic του WGAN επιστρέφει μια τιμή σε ένα εύρος που του επιτρέπει να ενεργεί λιγότερο αυστηρά ως κριτής (critic) [86].

Τα WGAN, πάρα το γεγονός ότι μπορεί να χρειαστούν περισσότερο χρόνο στην εκπαίδευση, αποτελούν μια από τις καλύτερες επιλογές για την επίτευξη αποδοτικότερων αποτελεσμάτων. Η βασική ιδέα λειτουργίας των GAN σχετίζεται με τη χρήση δυο κατανομών πιθανότητας. Η μια κατανομή αφορά την έξοδο που προκύπτει από τον Generator (Pg) και τα πραγματικά δεδομένα (Pr). Ο στόχος των Generative Adversarial Networks είναι να εξασφαλίσουν ότι και οι δύο αυτές κατανομές πιθανότητας είναι κοντά η μία στην άλλη, έτσι ώστε η παραγόμενη έξοδος να είναι ιδιαίτερα ρεαλιστική και υψηλής ποιότητας [86].

Για τον υπολογισμό της απόστασης μεταξύ των δύο αυτών κατανομών πιθανότητας έχουν προταθεί τρεις κύριες στατιστικές μέθοδοι. Αυτές είναι η απόκλιση Kullback–Leibler, η απόκλιση Jensen-Shannon (παραδοσιακή απώλεια GAN) και η απόσταση Wasserstein. Ο πιο συχνά χρησιμοποιήσιμος μηχανισμός είναι η απόκλιση Jensen-Shannon. Ωστόσο αυτή η μέθοδος παρουσιάζει προβλήματα, καθώς δουλεύει με κλίσεις που μπορούν να οδηγήσουν σε αστάθεια στο στάδιο της εκπαίδευσης [86]. Για να διορθωθούν

τέτοια επαναλαμβανόμενα ζητήματα γίνεται χρήση της Wasserstein απόστασης της οποίας η μαθηματική έκφραση ορίζεται παρακάτω. [51]:

$$\max_{w \in \mathcal{W}} \mathbb{E}_{x \sim \mathbb{P}_r} [f_w(x)] - \mathbb{E}_{z \sim p(z)} [f_w(g_\theta(z))] \quad (5)$$

Στη μαθηματική έκφραση το πρώτο μέρος της εξίσωσης αφορά τα πραγματικά δεδομένα ενώ το δεύτερο τα παραγόμενα από τον generator. Κατά την εκπαίδευση, ο discriminator-critic προσπαθεί να μεγιστοποιήσει την απόσταση μεταξύ των πραγματικών και των παραγόμενων δεδομένων. Αυτό συμβαίνει διότι θέλει να είναι σε θέση να μπορεί διακρίνει με επιτυχία τα δεδομένα. Από την άλλη, ο generator προσπαθεί να ελαχιστοποιήσει την απόσταση μεταξύ πραγματικών ( $\mathbb{P}_r$ ) και παραγόμενων δεδομένων ( $\mathbb{P}_g$ ), επειδή θέλει τα παραγόμενα δεδομένα να είναι όσο το δυνατόν πιο κοντά στα πραγματικά [86]. Επίσης, η μέγιστη τιμή (max) αντιπροσωπεύει τον περιορισμό Lipschitz του discriminator-critic.

Ο περιορισμός Lipschitz είναι μια τεχνική κανονικοποίησης που επιβάλλεται στον critic (discriminator) στα WGANs. Είναι απαραίτητος στα WGANs για τη σταθεροποίηση της εκπαίδευσης και τη διασφάλιση ότι η απόσταση Wasserstein, η οποία χρησιμοποιείται ως στόχος εκπαίδευσης, είναι καλά καθορισμένη.

Ο περιορισμός Lipschitz εφαρμόζεται στην έξοδο του critic-discriminator με σκοπό να εξασφαλιστεί ότι ικανοποιείται η συνθήκη συνέχειας Lipschitz. Μια συνάρτηση είναι συνεχής Lipschitz εάν υπάρχει μια σταθερά Lipschitz  $L$  τέτοια ώστε για δύο οποιαδήποτε σημεία εισόδου  $x$  και  $y$  η διαφορά στην έξοδο της συνάρτησης περιορίζεται από  $L$  φορές την ευκλείδεια απόσταση μεταξύ  $x$  και  $y$ :

$$|f(x) - f(y)| \leq L \cdot \|x - y\| \quad (6)$$

Στο πλαίσιο των WGANs, η επιβολή της συνέχειας Lipschitz είναι κρίσιμη για διάφορους λόγους όπως ο καλός καθορισμός της απόστασης Wasserstein. Ο περιορισμός Lipschitz διασφαλίζει ότι η συνάρτηση του critic δεν είναι πολύ "άγρια" ή μεταβάλλεται γρήγορα, καθιστώντας την απόσταση Wasserstein καλά ορισμένη οδηγώντας έτσι σε μια σταθερή και ουσιαστική εκπαίδευση. Επίσης η συνέχεια Lipschitz βοηθά στη σταθεροποίηση της διαδικασίας εκπαίδευσης,

εμποδίζοντας τον critic να αποδίδει ακραίες τιμές στα παραγόμενα και στα πραγματικά δείγματα, γεγονός που θα μπορούσε να οδηγήσει σε αστάθεια της εκπαίδευσης και κατάρρευση του τρόπου λειτουργίας. Τέλος βελτιώνει τις ιδιότητες σύγκλισης των WGAN, καθιστώντας τα λιγότερο ευαίσθητα στις επιλογές υπερπαραμέτρων και πιο ανθεκτικά κατά τη διάρκεια της εκπαίδευσης. Μια μέθοδος για τον περιορισμό του discriminator είναι η αποκοπή των βαρών (Weight Clipping). Η αποκοπή των βαρών περιλαμβάνει τον καθορισμό μιας τιμής κατώφλιου και την περικοπή των βαρών του discriminator που υπερβαίνουν αυτό το κατώφλι κατά τη διάρκεια κάθε επανάληψης στο στάδιο της εκπαίδευσης. Με τον περιορισμό των βαρών κατά αυτόν τον τρόπο, εξασφαλίζεται η ικανοποίηση του περιορισμού Lipschitz.

---

**Algorithm 1** WGAN, our proposed algorithm. All experiments in the paper used the default values  $\alpha = 0.00005$ ,  $c = 0.01$ ,  $m = 64$ ,  $n_{\text{critic}} = 5$ .

---

**Require:** :  $\alpha$ , the learning rate.  $c$ , the clipping parameter.  $m$ , the batch size.  $n_{\text{critic}}$ , the number of iterations of the critic per generator iteration.

**Require:** :  $w_0$ , initial critic parameters.  $\theta_0$ , initial generator's parameters.

```

1: while  $\theta$  has not converged do
2:   for  $t = 0, \dots, n_{\text{critic}}$  do
3:     Sample  $\{x^{(i)}\}_{i=1}^m \sim \mathbb{P}_r$  a batch from the real data.
4:     Sample  $\{z^{(i)}\}_{i=1}^m \sim p(z)$  a batch of prior samples.
5:      $g_w \leftarrow \nabla_w \left[ \frac{1}{m} \sum_{i=1}^m f_w(x^{(i)}) - \frac{1}{m} \sum_{i=1}^m f_w(g_\theta(z^{(i)})) \right]$ 
6:      $w \leftarrow w + \alpha \cdot \text{RMSPProp}(w, g_w)$ 
7:      $w \leftarrow \text{clip}(w, -c, c)$ 
8:   end for
9:   Sample  $\{z^{(i)}\}_{i=1}^m \sim p(z)$  a batch of prior samples.
10:   $g_\theta \leftarrow -\nabla_\theta \frac{1}{m} \sum_{i=1}^m f_w(g_\theta(z^{(i)}))$ 
11:   $\theta \leftarrow \theta - \alpha \cdot \text{RMSPProp}(\theta, g_\theta)$ 
12: end while

```

---

Εικόνα 24 Ο αλγόριθμος ενός WGAN με τη χρήση της μεθόδου Weight Clipping.[51]

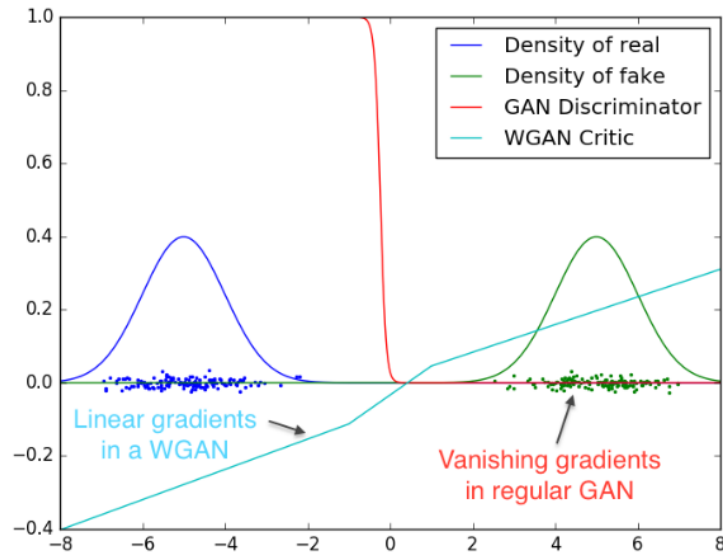
Στην Εικόνα 24 έχουμε τον αλγόριθμο που περιγράφει τη διαδικασία της εκπαίδευσης σε ένα WGAN χρησιμοποιώντας τη μέθοδο της αποκοπής βάρους. Ξεκινάει ορίζοντας τις παραμέτρους  $\alpha$  (ρυθμός μάθησης),  $m$  (μέγεθος της παρτίδας-batch size),  $c$  (σταθερά για την αποκοπή των βαρών) και  $n_{\text{critic}}$  (αριθμός επαναλήψεων του critic). Ο αλγόριθμος επαναλαμβάνει όσο η απώλεια  $\theta$  δεν συγκλίνει. Πλέον το κριτήριο για το αν συγκλίνει ή όχι καθορίζεται από το αν η τιμή του  $\theta$  πλησιάζει το 0. Αξίζει να σημειωθεί ότι σύμφωνα με τη μελέτη [51] όταν η τιμή του  $\theta$  συγκλίνει προς το 0, τότε έχει αποδειχθεί ότι σχετίζεται και με

εικόνες καλής ποιότητας. Στη συνέχεια υπάρχει μια επανάληψη, η οποία επαναλαμβάνεται πέντε φορές, και αυτό που κάνει είναι να εκπαιδεύει τον critic περισσότερο από τον generator. Συνεπώς, για κάθε πέντε βήματα ενημέρωσης για τον critic έχουμε ένα βήμα ενημέρωσης για τον Generator. Μέσα στο loop η διαδικασία δε διαφέρει σχεδόν καθόλου με ό,τι έχει αναφερθεί έως τώρα, δηλαδή υπολογίζεται η κλίση (Γραμμή 5) προσπαθώντας να μεγιστοποιηθεί. Γι' αυτό και στη συνέχεια γίνεται πρόσθεση αντί για αφαίρεση πολλαπλασιάζοντας τον ρυθμό μάθησης με την κλίση που βρέθηκε χρησιμοποιώντας τον βελτιστοποιητή RMSProp αντί για τον Adam.

Το πιο σημαντικό σημείο του παραπάνω αλγορίθμου είναι η αποκοπή που φαίνεται στη γραμμή 7. Συγκεκριμένα, χρησιμοποιώντας τη σταθερά αποκοπή  $c$ , 0,01, περιορίζει κάθε βάρος του δικτύου να πάρει την τιμή από -0.01 έως 0.01. Αυτό προκύπτει από το γεγονός ότι ο critic πρέπει να ικανοποιεί τον περιορισμό Lipschitz και ο τρόπος που βρήκαν οι ερευνητές να τον επιβάλουν είναι να περιορίσουν τα βάρη μέσα σε ένα μικρό διάστημα [-0,01, 0.01]. Τέλος ο αλγόριθμος τελειώνει με την εκπαίδευση του generator χρησιμοποιώντας ένα δείγμα θορύβου από τον λανθάνοντα χώρο και υπολογίζοντας το είδος της κλίσης της έκφρασης που υπάρχει στη γραμμή 5 σε σχέση με τις παραμέτρους  $\theta$  και υλοποιώντας το βήμα της ενημέρωσης αφαιρώντας (από τη στιγμή που θέλει να ελαχιστοποιήσει) από την παράμετρο  $\theta$ ,  $\alpha$  φορές τον βελτιστοποιητή RMSProp.

Στην Εικόνα 25 οι ερευνητές εκπαιδεύουν έναν discriminator GAN και έναν critic WGAN μέχρι τη βελτιστοποίηση. Αυτό που διαπιστώνεται είναι ότι ο discriminator μαθαίνει πολύ γρήγορα να διακρίνει μεταξύ ψεύτικου και πραγματικού, και συνεπώς δεν παρέχει αξιόπιστες πληροφορίες κλίσης. Απεναντίας, ο critic, δεν μπορεί να κορεστεί και συγκλίνει σε μια γραμμική συνάρτηση με καθαρές κλίσεις παντού. Ο περιορισμός των βαρών περιορίζει και την πιθανή ανάπτυξη της συνάρτησης να είναι το πολύ γραμμική σε διάφορα μέρη του χώρου, αναγκάζοντας τον βέλτιστο κριτή με αυτή τη συμπεριφορά [51]. Ωστόσο, η μέθοδος του weight clipping προκαλεί προβλήματα σε ορισμένες περιπτώσεις, όπως όταν το κατώφλι επιλογής είναι πολύ χαμηλό. Τότε, οδηγεί σε εξαφάνιση της κλίσης, σε περιπτώσεις μεγάλου αριθμού επιπέδων, μη χρήσης κανονικοποίησης παρτίδας (batch normalization) ή προβλημάτων που σχετίζονται με RNNs. Αν είναι πολύ υψηλό οδηγεί σε μεγάλους χρόνους

εκπαίδευσης, καθώς ο discriminator-critic χρειάζεται χρόνο για να προσαρμοστεί [86]. Αυτό δημιουργεί την ανάγκη για βελτίωση των WGAN.



Εικόνα 25 Ο discriminator και critic κατά την εκμάθηση της διάκρισης δύο κανονικών κατανομών. Όπως φαίνεται, ο discriminator ενός minimax GAN έχει κορεστεί και οδηγεί σε εξαφανιζόμενες κλίσεις, ενώ ο critic του WGAN παρέχει πολύ πιο καθαρές κλίσεις σε όλα τα μέρη του χώρου [51]

### 2.4.5.3 WGAN-GP (Wasserstein Generative Adversarial Network -Gradient Penalty)

Από τις καλύτερες λύσεις για την αντιμετώπιση του παραπάνω προβλήματος είναι η μέθοδος της ποινής κλίσης (Gradient Penalty). Η μέθοδος WGAN-GP προτείνει μια εναλλακτική λύση στην αποκοπή βάρους για να εξασφαλιστεί η ομαλή εκπαίδευση του δικτύου. Αντί για το ψαλίδισμα των βαρών, οι ερευνητές πρότειναν μια "ποινή κλίσης" προσθέτοντας έναν όρο απώλειας που διατηρεί την L2 νόρμα των κλίσεων του discriminator-critic κοντά στο 1 [87].

Στην Εικόνα 26 φαίνεται ο αλγόριθμος που αναπτύχθηκε από τους ερευνητές στη συγκεκριμένη δημοσίευση, [86] στον οποίο αρχικά ορίζονται κάποιοι βασικοί παράμετροι στην περίπτωση υλοποίησης της μεθόδους. Η παράμετρος  $\lambda$  ορίζει την ποινή κλίσης, ενώ το  $n_{critic}$  τον αριθμό των επαναλήψεων που κάνει ο critic στο διάστημα μιας επανάληψης του generator. Οι τιμές  $\alpha$  και  $\beta_1, \beta_2$  αναφέρονται στους περιορισμούς του βελτιστοποιητή Adam. Ο αλγόριθμος προτείνει τη χρήση μιας παρεμβατικής εικόνας παράλληλα με την παραγόμενη από τον generator εικόνα προτού προστεθεί η συνάρτηση απώλειας με ποινή κλίσης, ώστε να ικανοποιηθεί ο περιορισμός Lipschitz. Ο αλγόριθμος δε σταματά μέχρι να βρεθεί ικανοποιητική σύγκλιση στα απαιτούμενα δεδομένα [86].



---

**Algorithm 1** WGAN with gradient penalty. We use default values of  $\lambda = 10$ ,  $n_{\text{critic}} = 5$ ,  $\alpha = 0.0001$ ,  $\beta_1 = 0$ ,  $\beta_2 = 0.9$ .

---

**Require:** The gradient penalty coefficient  $\lambda$ , the number of critic iterations per generator iteration  $n_{\text{critic}}$ , the batch size  $m$ , Adam hyperparameters  $\alpha, \beta_1, \beta_2$ .

**Require:** initial critic parameters  $w_0$ , initial generator parameters  $\theta_0$ .

```

1: while  $\theta$  has not converged do
2:   for  $t = 1, \dots, n_{\text{critic}}$  do
3:     for  $i = 1, \dots, m$  do
4:       Sample real data  $\mathbf{x} \sim \mathbb{P}_r$ , latent variable  $\mathbf{z} \sim p(\mathbf{z})$ , a random number  $\epsilon \sim U[0, 1]$ .
5:        $\tilde{\mathbf{x}} \leftarrow G_\theta(\mathbf{z})$ 
6:        $\hat{\mathbf{x}} \leftarrow \epsilon \mathbf{x} + (1 - \epsilon) \tilde{\mathbf{x}}$ 
7:        $L^{(i)} \leftarrow D_w(\tilde{\mathbf{x}}) - D_w(\mathbf{x}) + \lambda(\|\nabla_{\hat{\mathbf{x}}} D_w(\hat{\mathbf{x}})\|_2 - 1)^2$ 
8:     end for
9:      $w \leftarrow \text{Adam}(\nabla_w \frac{1}{m} \sum_{i=1}^m L^{(i)}, w, \alpha, \beta_1, \beta_2)$ 
10:  end for
11:  Sample a batch of latent variables  $\{\mathbf{z}^{(i)}\}_{i=1}^m \sim p(\mathbf{z})$ .
12:   $\theta \leftarrow \text{Adam}(\nabla_\theta \frac{1}{m} \sum_{i=1}^m -D_w(G_\theta(\mathbf{z})), \theta, \alpha, \beta_1, \beta_2)$ 
13: end while

```

---

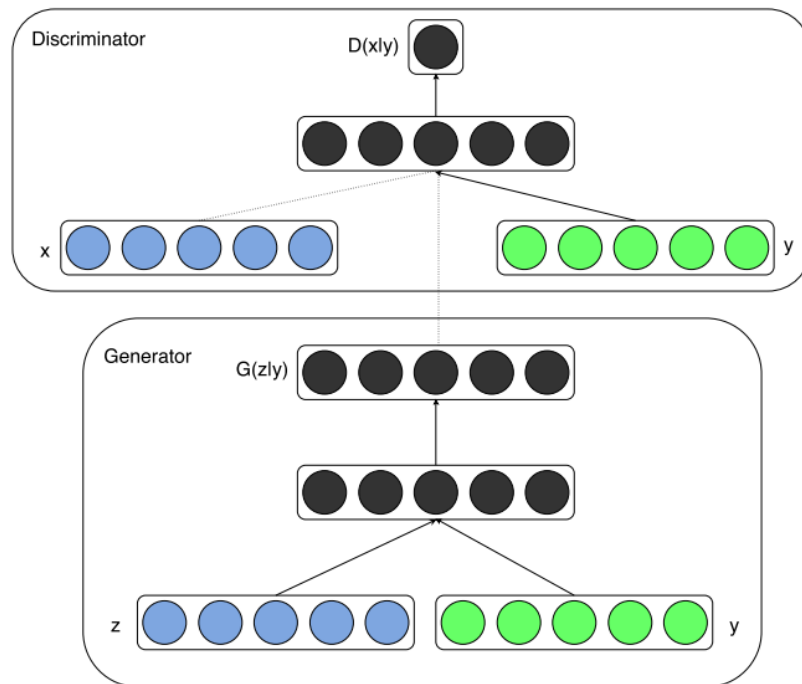
Εικόνα 26 Ο αλγόριθμος εκπαίδευσης ενός WGAN με τη χρήση της Gradient Penalty [86]

#### 2.4.5.4 cGAN (Conditional GANs)

Η αρχιτεκτονική που λειτουργεί συνδυαστικά με τις παραπάνω είναι τα GANs υπό συνθήκη (Conditional GANs ή cGANs). Η ανάγκη που δημιουργήθηκε για ανάπτυξη της συγκεκριμένης αρχιτεκτονικής πηγάζει από το γεγονός του ότι σε ένα μη εξαρτημένο (unconditional) παραγωγικό μοντέλο δεν υπάρχει κανένας έλεγχος στους τρόπους κατά τους οποίους παράγονται τα δεδομένα. Ωστόσο, με την υποσυνθήκη προσαρμογή του μοντέλου σε επιπρόσθετες πληροφορίες είναι δυνατόν να κατευθυνθεί η διαδικασία παραγωγής δεδομένων [49].

Τα GANs μπορούν να επεκταθούν σε ένα μοντέλο υπο-συνθήκη εάν τόσο ο generator όσο και ο discriminator εξαρτώνται από κάποια επιπρόσθετη πληροφορία  $y$ . Το  $y$  μπορεί να είναι οποιαδήποτε επιπλέον πληροφορία, όπως οι ετικέτες κλάσεων ή δεδομένα από άλλες λειτουργίες. Για να υλοποιηθεί η παραπάνω εξάρτηση, τροφοδοτείται η  $y$  τόσο στον discriminator όσο και στον generator ως επιπρόσθετο επίπεδο εισόδου [49]. Στον Generator ο αρχικός θόρυβος εισόδου  $P_z(\mathbf{z})$ , και η πληροφορία  $y$  συνδυάζονται σε κοινή κρυφή αναπαράσταση, και το πλαίσιο της αντιπαλικής εκπαίδευσης επιτρέπει σημαντική ευελιξία στον τρόπο σύνθεσης αυτής της κρυφής αναπαράστασης. Στον Discriminator, τα  $x$  και  $y$  αναπαρίστανται σαν είσοδοι σε μια διαχωριστική συνάρτηση [49]. Η αντικειμενική συνάρτηση ενός minimax παιχνιδιού δυο παιχτών παίρνει την ακόλουθη μορφή:

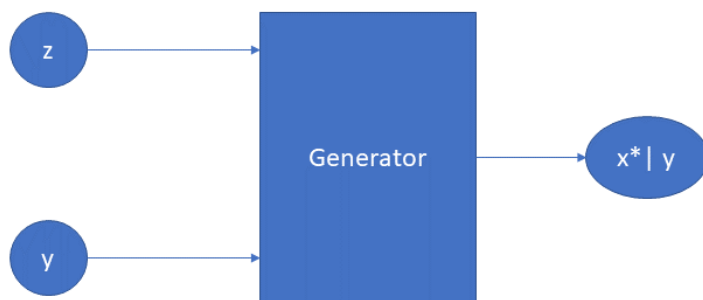
$$\min_G \max_D V(D, G) = E_{x \sim p_{data}(x)} [\log D(x | y)] + E_{z \sim p_z(z)} [\log(1 - D(G(z | y)))] \quad (7)$$



Εικόνα 27 Η δομή ενός απλού cGAN [49]

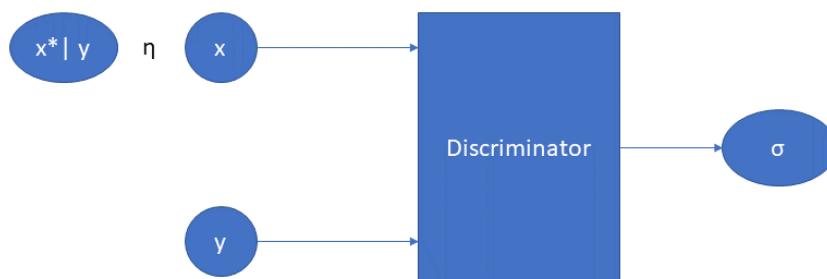
### Εκπαίδευση

Κατά τη διάρκεια της εκπαίδευσης ενός cGAN, ο Generator μαθαίνει να παράγει ρεαλιστικά παραδείγματα για κάθε ετικέτα στο σύνολο εκπαίδευσης και ο Discriminator μαθαίνει να διακρίνει τα ψεύτικα ζεύγη παράδειγμα-ετικέτας από τα πραγματικά ζεύγη παράδειγμα-ετικέτα. Αξίζει να σημειωθεί ότι σε ένα cGAN, ο Discriminator δε μαθαίνει να αναγνωρίζει ποια κλάση είναι ποια. Αυτό που μαθαίνει είναι αποδέχεται πραγματικά ταιριαστά ζεύγη ενώ απορρίπτει τα ζεύγη τα οποία δεν ταιριάζουν και τα ζεύγη στα οποία το παράδειγμα είναι ψεύτικο. Για παράδειγμα ο Discriminator σε ένα cGAN πρέπει να μάθει να απορρίπτει ένα ζευγάρι (2,5), ανεξάρτητα με το αν το παράδειγμα (2) είναι αληθινό η ψεύτικο, για τον λόγο ότι δεν ταιριάζει με την ετικέτα 5. Επίσης θα πρέπει να απορρίπτει όλα τα ζεύγη εικόνα-ετικέτας εάν η εικόνα είναι ψεύτικη παρόλο που μπορεί η εικόνα να ταιριάζει με την ετικέτα. Συνεπώς για να μπορέσει ο Generator να ξεγελάσει τον Discriminator δεν αρκεί μόνο να παράξει ρεαλιστικά παραδείγματα αλλά και να ταιριάζουν και με την ετικέτα τους. Όταν ο Generator εκπαιδευτεί ολοκληρωτικά δίνει τη δυνατότητα να επιλεγθεί το παράδειγμα και να ολοκληρωθεί η σύνθεση με την κατάλληλη ετικέτα σαν όρισμα [83].



Εικόνα 28 cGAN Generator

Στην Εικόνα 28 φαίνεται ο τρόπος που διαμορφώνεται το δίκτυο του Generator μέσα σε ένα cGAN. Ο Generator παίρνει σαν είσοδο διάνυσμα θορύβου  $z$  όπως και σε ένα κλασικό GAN, μαζί με μια ετικέτα  $y$  και τα χρησιμοποιεί για να συνθέσει ένα ψεύτικο παράδειγμα  $G(z,y)=x^*|y$ . Ο στόχος του συγκεκριμένου ψεύτικου παραδείγματος που παράγεται είναι να μοιάζει όσο το δυνατόν περισσότερο με ένα πραγματικό για τη συγκεκριμένη ετικέτα  $y$  [83].

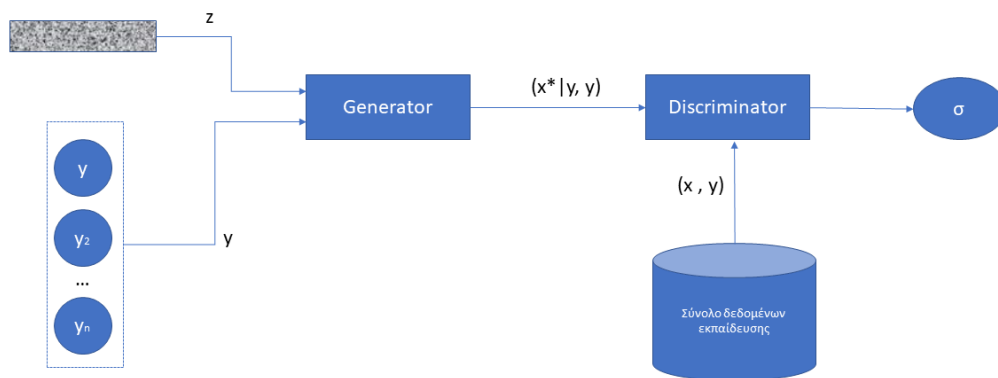


Εικόνα 29 cGAN Discriminator

Στην Εικόνα 29 παρουσιάζεται το δίκτυο του Discriminator μέσα σε ένα cGAN. Στην είσοδο δέχεται τα πραγματικά δεδομένα από το σύνολο εκπαίδευσης μαζί με τις αντίστοιχες ετικέτες τους  $(x,y)$  και τα παραγόμενα ψεύτικα από τον Generator παραδείγματα μαζί με την ετικέτα που χρησιμοποιήθηκε για τη σύνθεση τους  $(x^*|y,y)$ . Στα πραγματικά ζεύγη παράδειγμα-ετικέτα, ο Discriminator μαθαίνει πως να αναγνωρίζει τα πραγματικά δεδομένα και πως να αναγνωρίζει ταιριαστά ζεύγη. Στα παραγόμενα από τον Generator παραδείγματα μαθαίνει να αναγνωρίζει ψεύτικα ζεύγη εικόνας-ετικέτας, μαθαίνοντας έτσι να τα ξεχωρίζει από τα πραγματικά [83]. Ο Discriminator στην έξοδο δίνει μια μοναδική πιθανότητα που δείχνει την πελοίθησή του ότι η

είσοδος είναι ένα πραγματικό, ταιριαστό ζεύγος. Στόχος του Discriminator είναι να απορρίπτει όλα τα ψεύτικα παραδείγματα και όλα αυτά που δεν ταιριάζουν με την ετικέτα τους, ενώ αποδέχεται όλα τα πραγματικά ζεύγη [83].

Συνδυάζοντας τα δύο προαναφερόμενα υποδίκτυα του Generator και του Discriminator σε ένα ενιαίο δίκτυο (cGAN) (Εικόνα 30) παρατηρείται ότι για κάθε ψεύτικο παράδειγμα η ίδια ετικέτα  $y$  περνάει και στον Generator και στον Discriminator. Επίσης παρατηρείται ότι ο Discriminator δεν εκπαιδεύεται ποτέ απόλυτα να απορρίπτει τα ζευγάρια που δεν ταιριάζουν. Καθώς εκπαιδεύεται σε πραγματικά παραδείγματα με αταίριαστες ετικέτες, η ικανότητά του να εντοπίζει αταίριαστα ζεύγη είναι ένα αποτέλεσμα της εκπαίδευσής του να δέχεται μόνο πραγματικά ταιριαστά ζεύγη.



Εικόνα 30 cGAN -Ο Generator χρησιμοποιεί ένα διάνυσμα θορύβου  $z$  και μια ετικέτα  $y$  (μια από τις πιθανές  $n$ ) σαν είσοδο για να παράξει ένα ψεύτικο παράδειγμα  $x^* | y$  το οποίο προσπαθεί να δείχνει ρεαλιστικό και ταιριαστό με την ετικέτα  $y$

Τέλος, σαν γενικό συμπέρασμα σε ένα cGAN ο Discriminator λαμβάνει ψεύτικα παραδείγματα με ετικέτες  $(x^* | y, y)$  που παράγονται από τον Generator και πραγματικά παραδείγματα με ετικέτες  $(x, y)$  και μαθαίνει να διακρίνει αν ένα δεδομένο παράδειγμα-ετικέτα είναι πραγματικό η ψεύτικο [83].

## 2.4.6 Μετρικές Αξιολόγησης GANs

### 2.4.6.1 Fréchet Inception Distance (FID)

Η Fréchet Inception Distance (FID) είναι μια μετρική που χρησιμοποιείται για την αξιολόγηση της ποιότητας των Generative Adversarial Networks (GANs), συγκεκριμένα για την αξιολόγηση της ομοιότητας μεταξύ των κατανομών των πραγματικών και των παραγόμενων εικόνων. Μετρά την απόσταση μεταξύ δύο πολυμεταβλητών κατανομών Gaussian - η μία αντιπροσωπεύει τις

αναπαραστάσεις χαρακτηριστικών των πραγματικών εικόνων που λαμβάνονται από ένα προ-εκπαιδευμένο μοντέλο σύλληψης και η άλλη αντιπροσωπεύει τα χαρακτηριστικά των παραγόμενων εικόνων.[88]

Η βαθμολογία FID υπολογίζει την απόσταση Fréchet, η οποία λαμβάνει υπόψη τόσο τη μέση τιμή όσο και τη συνδιακύμανση των κατανομών. Χαμηλότερες βαθμολογίες FID υποδηλώνουν καλύτερη απόδοση, γεγονός που υποδηλώνει ότι οι παραγόμενες εικόνες ταιριάζουν στενά με την κατανομή της πραγματικής εικόνας όσον αφορά την οπτική ποιότητα και την ποικιλομορφία. [88]

Ο υπολογισμός περιλαμβάνει την εξαγωγή χαρακτηριστικών από ένα προ-εκπαιδευμένο δίκτυο Inception και στη συνέχεια τον υπολογισμό στατιστικών στοιχείων (μέση τιμή και συνδιακύμανση) τόσο για τις πραγματικές όσο και για τις παραγόμενες κατανομές χαρακτηριστικών εικόνων. Η βαθμολογία FID προκύπτει από αυτά τα στατιστικά στοιχεία χρησιμοποιώντας τον τύπο της απόστασης Fréchet. [88]

#### **2.4.6.2 Kernel Inception Distance (KID)**

Η Kernel Inception Distance (KID) είναι μια άλλη μετρική που χρησιμοποιείται για την αξιολόγηση της απόδοσης των Generative Adversarial Networks (GANs) μετρώντας τη διαφορά μεταξύ των κατανομών των πραγματικών και των παραγόμενων εικόνων. Η KID επικεντρώνεται συγκεκριμένα στην αξιολόγηση της στατιστικής ομοιότητας των αναπαραστάσεων χαρακτηριστικών που εξάγονται από ένα προεκπαιδευμένο βαθύ νευρωνικό δίκτυο[89].

Η KID χρησιμοποιεί τεχνικές ενσωμάτωσης του μέσου πυρήνα για να απεικονίσει τις κατανομές των πραγματικών και των παραγόμενων εικόνων σε έναν χώρο Hilbert με αναπαραγωγικό πυρήνα (RKHS). Υπολογίζει το τετράγωνο της Μέγιστης Μέσης Απόκλισης (MMD) μεταξύ αυτών των ενσωματώσεων, λαμβάνοντας υπόψη τις αναπαραστάσεις των εικόνων σε διάφορα επίπεδα ενός προ-εκπαιδευμένου νευρωνικού δικτύου (συχνά δίκτυο Inception).[90]

Η βαθμολογία ποσοτικοποιεί την ασυμφωνία μεταξύ των κατανομών των πραγματικών και των παραγόμενων εικόνων στο χώρο των χαρακτηριστικών. Χαμηλότερες βαθμολογίες KID υποδηλώνουν καλύτερη απόδοση, που σημαίνει ότι οι παραγόμενες εικόνες ταιριάζουν στενά με την κατανομή της πραγματικής εικόνας όσον αφορά τις αναπαραστάσεις χαρακτηριστικών.

### **3 ΚΕΦΑΛΑΙΟ 3: ΜΕΘΟΔΟΛΟΓΙΑ**

Στο Κεφάλαιο 3 της συγκεκριμένης μελέτης αναλύονται όλα τα στάδια της μεθοδολογίας που ακολουθήθηκαν προκειμένου να επιλυθεί το πρόβλημα της ανισορροπίας των κλάσεων στη διάγνωση της πνευμονίας. Τα βασικά βήματα ήταν 1) Η δημιουργία και εκπαίδευση του ταξινομητή, 2) Η αξιολόγηση των καλύτερων μοντέλων cGAN και 3) Η βελτιστοποίηση της μειωτικής κλάσης Normal με τη χρήση συνθετικών δειγμάτων από cGAN.

#### **Δημιουργία και Εκπαίδευση ταξινομητή**

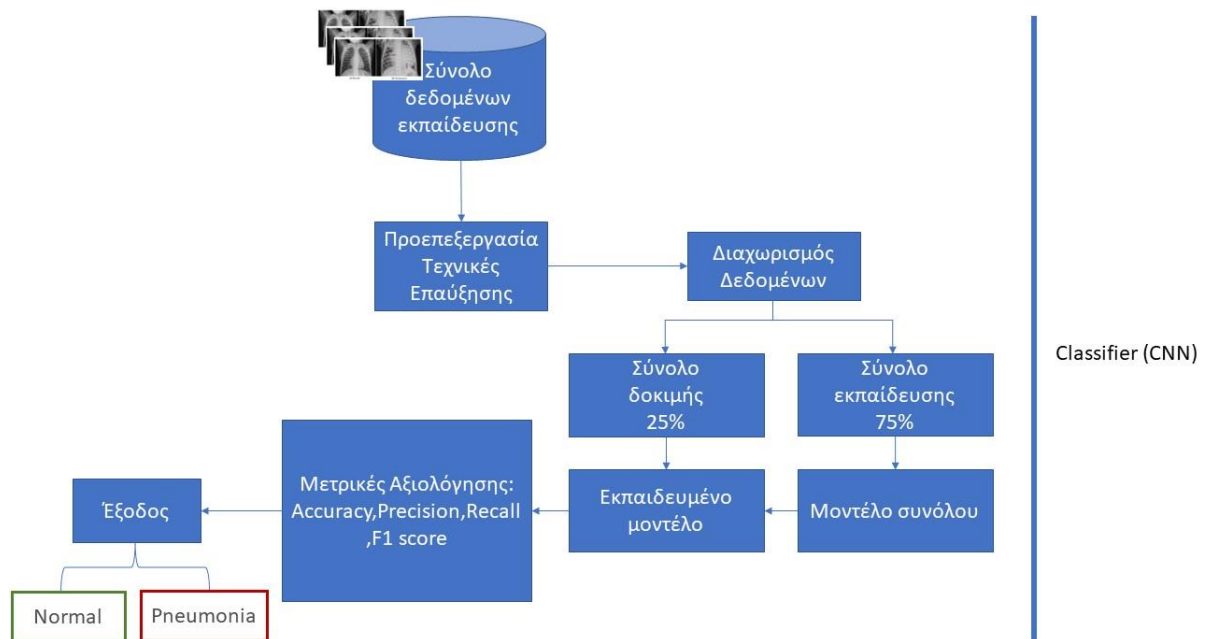
Στην αφετηρία των εργασιών της δημιουργίας και εκπαίδευσης του ταξινομητή, βρίσκεται η φάση της προεπεξεργασίας και των τεχνικών επαύξησης των δεδομένων (Εικόνα 31). Αυτό το αρχικό στάδιο κατέχει κρίσιμο ρόλο, περιλαμβάνοντας εργασίες, όπως ο καθαρισμός των δεδομένων, η μείωση του θορύβου και η χρήση τεχνικών επαύξησης του συνόλου δεδομένων. Αυτή η διαδικασία όχι μόνο διορθώνει τις εγγενείς ανωμαλίες στο σύνολο δεδομένων, αλλά και ενισχύει την ποικιλομορφία και τον όγκο τους. Η επαύξηση, ειδικότερα, συνεπάγεται τη σύνθεση πρόσθετων δεδομένων, που χρησιμεύουν στον εμπλουτισμό του συνόλου δεδομένων.

Μετά από την προετοιμασία των δεδομένων, ακολουθεί ο "Διαχωρισμός δεδομένων". Αυτό το βήμα χωρίζει το σύνολο δεδομένων σε δύο διακριτά υποσύνολα: 1) το σύνολο εκπαίδευσης και 2) το σύνολο δοκιμής. Το πρώτο αναλαμβάνει τον ρόλο του εκπαιδευτή, διευκολύνοντας τη διαδικασία μάθησης του ταξινομητή (CNN). Συνεπώς, αυτά τα επιμέρους σημεία συνθέτουν τη διαδικασία εκπαίδευσης του ταξινομητή (Εικόνα 31).

Εδώ πρέπει να σημειωθεί ότι για την εκπαίδευση του ταξινομητή στο σύνολο εκπαίδευσης χρησιμοποιήθηκαν τα προεκπαιδευμένα μοντέλα ResNet-18, ResNet34, DenseNet-121, MobileNet\_v2. Το σύνολο δοκιμής λειτουργεί ως πεδίο αξιολόγησης, μετρώντας την απόδοση του ταξινομητή σε δεδομένα που δεν είχαν προηγουμένως ληφθεί υπόψη. Ο διαχωρισμός αυτός είναι σημαντικός για να ενημερωθούμε για την ύπαρξη υπερβολικής προσαρμογής (overfitting), κατά την οποία ο ταξινομητής προσαρμόζεται υπερβολικά με τα δεδομένα εκπαίδευσης, εμποδίζοντας την ικανότητά του για αποτελεσματική γενίκευση.

Η επόμενη φάση περιλαμβάνει την αξιολόγηση της απόδοσης του ταξινομητή με τη χρήση των μετρικών αξιολόγησης. Οι μετρικές Accuracy, το Precision, το

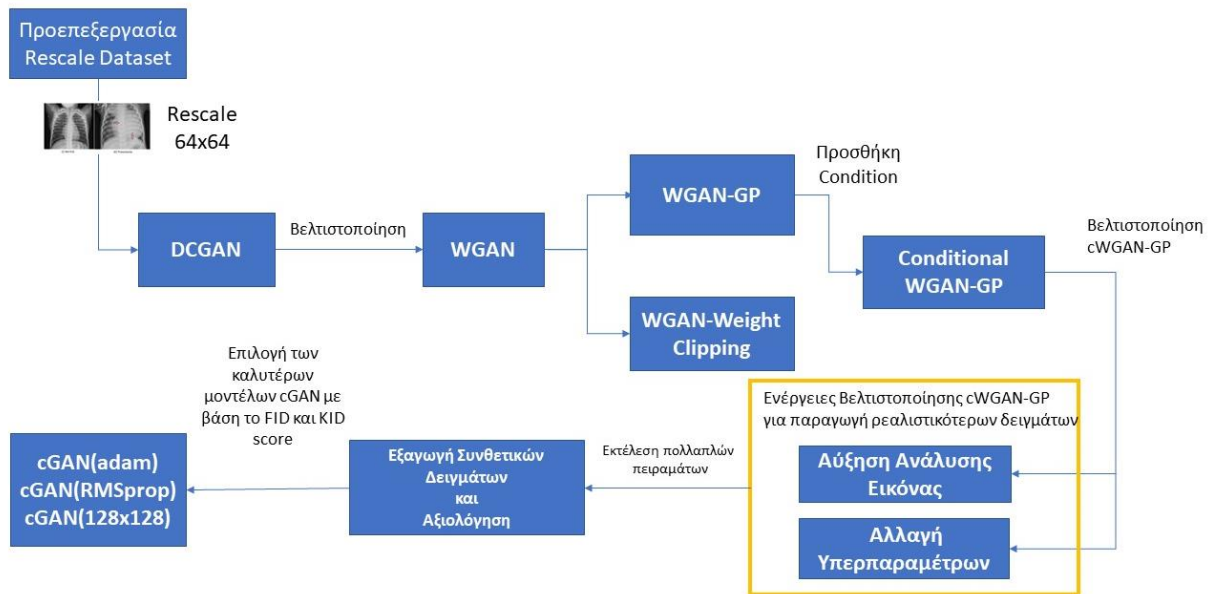
Recall και το F1-Score εφαρμόζονται στο σύνολο δοκιμών. Αυτές οι μετρικές παρέχουν ποσοτικές πληροφορίες σχετικά με την ικανότητα του ταξινομητή, περιγράφοντας την ακρίβεια, την ευαισθησία και τη συνολική προβλεπτική του απόδοση. Τέλος, ο ταξινομητής παρέχει ένα δυαδικό αποτέλεσμα, κατηγοριοποιώντας την εικόνα εισόδου ως φυσιολογική (Normal) ή πνευμονίας (Pneumonia) (Εικόνα 31).



Εικόνα 31 Η ροή εργασιών κατά την εκπαίδευση του ταξινομητή

### Δημιουργία και επιλογή των καλύτερων μοντέλων cGAN

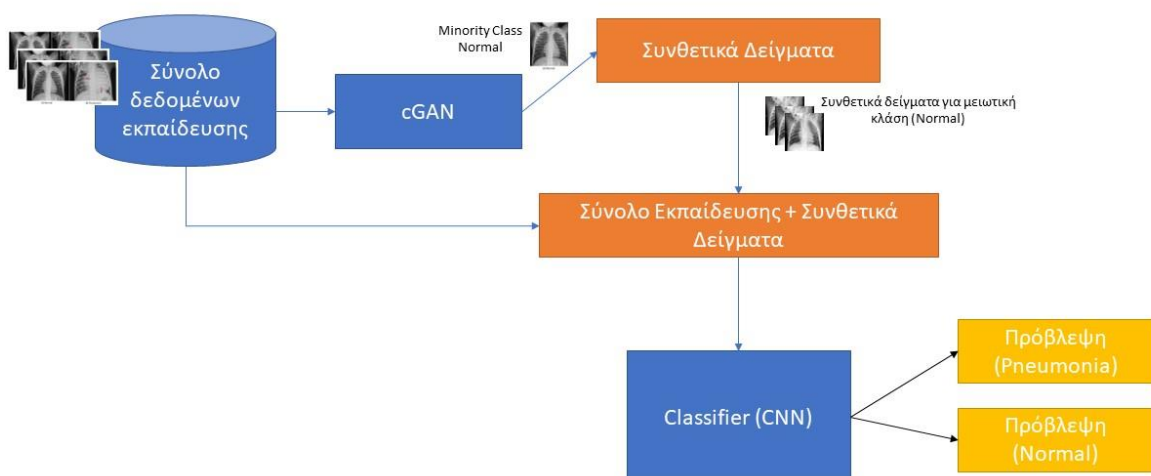
Μετά την προετοιμασία του ταξινομητή ακολουθεί η βελτιστοποίηση της απόδοσής του, αντιμετωπίζοντας το πρόβλημα της ανισορροπίας που υπήρχε στο dataset των εικόνων, αρχικά με μία παραδοσιακή τεχνική Oversampling, το Class weighting, αλλά και με την πιο μοντέρνα, όπως το cGAN για την παραγωγή νέων συνθετικών δειγμάτων για την ενίσχυση της κλάσης μειοψηφίας. Στο πλαίσιο αυτό, το μεγαλύτερο μέρος της μεθοδολογίας εστιάζει στον τρόπο που χτίστηκε το cGAN και στη διαδικασία που ακολουθήθηκε για τη βελτιστοποίηση του αποτελέσματος και την επιλογή των καλύτερων μοντέλων cGAN. Στο παρακάτω διάγραμμα (Εικόνα 32) μπορούμε να δούμε και σχηματικά αυτή τη διαδικασία η οποία αναλύεται στη συνέχεια του συγκεκριμένου Κεφαλαίου.



Εικόνα 32 Η ροή εργασιών που ακολουθήθηκε για την επιλογή των καλύτερων μοντέλων cGAN

### Βελτιστοποίηση της μειωτικής κλάσης Normal με τη χρήση συνθετικών δειγμάτων από cGAN

Το τελευταίο στάδιο περιλαμβάνει τη βελτιστοποίηση της μειωτικής κλάσης Normal με τη χρήση συνθετικών δειγμάτων από cGAN. Αφού έγινε η απαραίτητη επεξεργασία του dataset και ακολουθήθηκαν οι απαραίτητες ενέργειες διαχείρισής του και επιλέχθηκαν τα καλύτερα μοντέλα cGAN, τα συνθετικά δείγματα μαζί με το σύνολο εκπαίδευσης ενσωματώθηκαν στον δυαδικό ταξινομητή, ο οποίος επέτρεψε την πρόβλεψη της πνευμονίας και των περιπτώσεων Normal (Εικόνα 33).



Εικόνα 33 Η ροή εργασιών για τη βελτιστοποίηση της μειωτικής κλάσης Normal με τη χρήση συνθετικών δειγμάτων από cGAN.



## 3.1 Classifier

### 3.1.1 Σύντομη περιγραφή αρχιτεκτονικής

#### PneumoniaNet

Το δίκτυο δέχεται έναν τανιστή (Tensor) εισόδου  $x$  και τον περνά από διάφορα επίπεδα. Η ραχοκοκαλιά (backbone) του δικτύου είναι ένα μοντέλο ResNet-18, το οποίο αρχικοποιείται με προεκπαιδευμένα βάρη (αν pretrained=True). Ο Tensor  $x$  περνάει πρώτα από το αρχικό συνελκτικό επίπεδο conv1. Στη συνέχεια, εφαρμόζεται batch normalization, περνάει από τη συνάρτηση ενεργοποίησης ReLU και τέλος από max pooling. Αμέσως μετά, η έξοδος από τα αρχικά επίπεδα περνά από τέσσερα υπολειμματικά (residuals) μπλοκ (layer 1, layer 2, layer 3, layer 4) του μοντέλου ResNet-18 και μετά το τελευταίο υπολειμματικό μπλοκ, η έξοδος περνάει από τη μέση συγκέντρωση (Average Pooling). Ο Tensor που προκύπτει στη συνέχεια αναδιαμορφώνεται με τη χρήση της συνάρτησης view, ώστε να έχει σχήμα (batch\_size, 512), όπου 512 αντιστοιχεί στον αριθμό των χαρακτηριστικών εξόδου από το backbone.

Τέλος, ο αναδιαμορφωμένος tensor περνάει από ένα πλήρως συνδεδεμένο επίπεδο (fc) με 512 χαρακτηριστικά εισόδου και 1 χαρακτηριστικό εξόδου που χρησιμοποιείται για δυαδική ταξινόμηση (στην προκειμένη περίπτωση, για την ανίχνευση πνευμονίας). Η έξοδος του πλήρως συνδεδεμένου επιπέδου αντιπροσωπεύει την προβλεπόμενη πιθανότητα η είσοδος να ανήκει στη θετική κλάση (πνευμονία).

Συνολικά, η αρχιτεκτονική PneumoniaNet συνδυάζει τη ραχοκοκαλιά του ResNet-18 με ένα τροποποιημένο, αλλά πλήρως συνδεδεμένο επίπεδο για την ταξινόμηση της πνευμονίας. Ακόμη, αξιοποιεί το προ-εκπαιδευμένο μοντέλο ResNet-18 για την εξαγωγή χαρακτηριστικών και προσθέτει έναν προσαρμοσμένο ταξινομητή για την τελική εργασία ταξινόμησης.

### 3.1.2 Εκπαίδευση CNN Classifier

Χρησιμοποιώντας την παραπάνω αρχιτεκτονική εκπαιδεύτηκε ο ταξινομητής, με τη χρήση των ακόλουθων υπερπαραμέτρων (Πίνακας 2).

Πίνακας 2 CNN υπερπαραμέτροι εκπαίδευσης

Παράμετρος	Τιμή
Batch size	32
Learning rate	0.003

Epochs	5
Criterion	BCEWithLogitsLoss
Optimizer	Adam

Για το Batch size επιλέχθηκε η τιμή 32, καθώς επιτυγχάνει μια ισορροπία μεταξύ της υπολογιστικής απόδοσης και της σύγκλισης του μοντέλου. Μικρότερο batch size μπορούσε να οδηγήσει σε πιο θορυβώδεις ενημερώσεις κλίσης, ενώ μεγαλύτερο απαιτούσε πολύ μεγαλύτερη μνήμη.

Η τιμή του Learning rate 0.003 επιλέχθηκε εμπειρικά, καθώς μετά από δοκιμές παρατηρήθηκε ότι με τιμές μεγαλύτερες το μοντέλο υπερέβαινε τα βέλτιστα βάρη και οδηγούταν σε απόκλιση, ενώ δοκιμάζοντας πολύ μικρότερες προκαλούσε πολύ αργή σύγκλιση.

Ο αριθμός εποχών ορίστηκε 5, καθώς κάνουμε χρήση προεκπαιδευμένου μοντέλου και δεν προκύπτει η ανάγκη για περισσότερα περάσματα. Ως συνάρτηση απώλειας επιλέχθηκε η BCEWithLogitsLoss, καθώς είναι η συνάρτηση που χρησιμοποιείται συνήθως για προβλήματα δυαδικής ταξινόμησης, όπου η έξοδος του μοντέλου είναι μια ενιαία τιμή πιθανότητας ανά κλάση.

Τέλος, ως βελτιστοποιητής επιλέχθηκε ο Adam, καθώς είναι ένας προσαρμοστικός αλγόριθμος που συνδυάζει στοιχεία από τον RMSprop και το Momentum. Ο κύριος λόγος επιλογής του είναι οι καλές ιδιότητες σύγκλισης και προσαρμοστικότητας που διαθέτει σε διαφορετικούς τύπους δεδομένων.

Επίσης, για να μπορέσει να υλοποιηθεί με τον πιο αποτελεσματικό τρόπο η εκπαίδευση του μοντέλου, έγινε χρήση του CUDA framework της NVIDIA. Το CUDA, στην παρούσα εργασία, επιτρέπει να αξιοποιηθεί η ισχύς της NVIDIA κάρτας γραφικών (GPU) για την επιτάχυνση της διαδικασίας εκπαίδευσης του νευρωνικού δικτύου, η οποία αποτελεί μια περίπλοκη διαδικασία. Γενικά, η χρήση GPU στην περίπτωση των νευρωνικών δικτύων και ειδικά όταν το είδος των δεδομένων είναι εικόνες που απαιτούν πολύ περισσότερους υπολογισμούς σε σχέση με αλλά πιο απλά δεδομένα, όπως δεδομένα κειμένου, είναι απαραίτητη. Στη συγκεκριμένη μελέτη, τα περισσότερα πειράματα έγιναν με τη χρήση της NVIDIA GeForce MX250 2GB σε συνδυασμό με το CUDA (Version 11.7) (Εικόνα 34).

```

! nvidia-smi
Fri Dec 16 12:45:05 2022
+-----+
| NVIDIA-SMI 516.01          Driver Version: 516.01          CUDA Version: 11.7          |
+-----+-----+-----+
| GPU  Name                TCC/WDDM | Bus-Id          Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap |          Memory-Usage | GPU-Util  Compute M. |
|                               |                    |           MIG M.     |
+-----+-----+-----+
|   0   NVIDIA GeForce ... WDDM | 00000000:01:00.0 Off |           N/A       |
| N/A   51C   P0   N/A /  N/A   |  88MiB /  2048MiB |    0%      Default   |
|                               |                    |           N/A       |
+-----+-----+-----+

+-----+
| Processes: |
| GPU  GI   CI           PID  Type  Process name                      GPU Memory |
|      ID   ID                                     |            Usage |
+-----+-----+-----+
|   0   N/A  N/A       10176   C   ...\\envs\\deep_env\\python.exe    N/A |
+-----+

```

Εικόνα 34 NVIDIA Cuda

Βέβαια σε κάποια πειράματα που παρουσιάζονται στη συνέχεια, η συγκεκριμένη GPU δεν ήταν αρκετή. Στο πλαίσιο αυτό, έγινε χρήση των GPUs του Google Colab.

Στην Εικόνα 35 φαίνεται ο αλγόριθμος εκπαίδευσης του μοντέλου.

```
def train_model(model, criterion, optimizer, num_epochs, device="cuda"):
    print("Train started")
    since = time.time()
    best_model_wts = copy.deepcopy(model.state_dict())
    best_acc = 0
    for epoch in tqdm(range(num_epochs), leave=False):
        for phase in ["train", "val"]:
            if phase == "train":
                model.train()
            else:
                model.eval()

            running_loss = 0.0
            running_corrects = 0

            for i, (inputs, labels) in tqdm(enumerate(dataloaders[phase]), leave=False, total=len(dataloaders[phase])):
                inputs = inputs.to(device)
                labels = labels.to(device)
                #Start backpropagation
                optimizer.zero_grad()
                with torch.set_grad_enabled(phase=="train"):
                    outputs = model(inputs)
                    preds = outputs.sigmoid() > 0.5
                    loss = criterion(outputs, labels.float())
                    if phase == "train":
                        loss.backward()
                        optimizer.step()
                running_loss += loss.item() * inputs.size(0)
                running_corrects += torch.sum(preds==labels.data)

            if (i % logging_steps[phase]==0) & (i > 0):
                avg_loss = running_loss / ((i+1) * batch_sizes[phase])
                avg_acc = running_corrects / ((i+1) * batch_sizes[phase])

                print(f"[{phase}]:{epoch+1} / {num_epochs} | loss:{avg_loss} | acc:{avg_acc}")

            epoch_loss = running_loss / dataset_sizes[phase]
            epoch_acc = running_corrects / dataset_sizes[phase]
            print("{} Loss: {:.4f} Acc: {:.4f}".format(phase, epoch_loss, epoch_acc))
            if phase == "val" and epoch_acc > best_acc:
                best_acc = epoch_acc
                best_model_wts = copy.deepcopy(model.state_dict())

        print()
        time_elapsed = time.time() - since
        print(f"Training took {time_elapsed} seconds")
        model.load_state_dict(best_model_wts)
    return model
```

Εικόνα 35 Αλγόριθμος εκπαίδευσης CNN

Δημιουργήθηκε μια συνάρτηση εκπαίδευσης η οποία λαμβάνει σαν όρισμα το CNN μοντέλο, τη συνάρτηση απώλειας, τον βελτιστοποιητή, τον αριθμό των εποχών, και το device που θα χρησιμοποιηθεί, δηλαδή την GPU (cuda). Για κάθε εποχή, το μοντέλο εκπαιδεύεται στο σύνολο εκπαίδευσης και αξιολογείται στο σύνολο επικύρωσης. Κατά τη διάρκεια της φάσης εκπαίδευσης (*phase=train*), το μοντέλο τίθεται σε κατάσταση εκπαίδευσης (*model.train()*), επιτρέποντας τον υπολογισμό κλίσης και τις ενημερώσεις παραμέτρων. Στη φάση επικύρωσης (*phase=val*), το μοντέλο μεταβαίνει σε λειτουργία αξιολόγησης (*model.eval()*) για να απενεργοποιήσει τον υπολογισμό κλίσης. Ακολούθως, για κάθε batch δεδομένων, οι εικόνες εισόδου και οι επικέτες του μεταφέρονται στην GPU (cuda). Οι κλίσεις του βελτιστοποιητή μηδενίζονται (*optimizer.zero\_grad()*) και το μοντέλο εκτελεί ένα πέρασμα προς τα εμπρός για να κάνει προβλέψεις.

Μετά από αυτό, υπολογίζεται η απώλεια με βάση τις προβλέψεις που έκανε το μοντέλο και τις επικείμενες αλήθειες. Στην περίπτωση που βρίσκεται στη φάση εκπαίδευσης εκτελείται οπισθοδιάδοση (backpropagation) και ο βελτιστοποιητής ενημερώνει τις παραμέτρους του μοντέλου. Για την παρακολούθηση της προόδου της εκπαίδευσης, κρατήθηκαν μετρικές απώλειας (running\_loss) και ακρίβειας (running\_corrects).

Ανά τακτά χρονικά διαστήματα, υπολογίζονται και εκτυπώνονται η μέση απώλεια και η μέση ακρίβεια σε οποία φάση του μοντέλου (train η val) παρέχοντας πληροφορίες για την απόδοση του μοντέλου και την πρόοδο της εκπαίδευσης. Κάθε φορά που ολοκληρώνεται μια εποχή, η ακρίβεια επικύρωσης συγκρίνεται με την καλύτερη ακρίβεια που έχει επιτευχθεί μέχρι εκείνη τη στιγμή. Αν η τρέχουσα είναι υψηλότερη, τα βάρη του μοντέλου ενημερώνονται διασφαλίζοντας ότι διατηρείται το μοντέλο με την καλύτερη επίδοση.

Οι κύριες μετρικές αξιολόγησης που χρησιμοποιήθηκαν για τον συγκεκριμένο ταξινομητή είναι η ακρίβεια (Accuracy), η οποία μετρά το ποσοστό των σωστά ταξινομημένων δειγμάτων στο σύνολο επικύρωσης και η απώλεια (εκπαίδευσης και επικύρωσης), η οποία βοηθά στην αξιολόγηση της σύγκλισης και της γενίκευσης του μοντέλου. Επίσης, βοηθητικές ήταν και οι μετρικές Precision, Recall και F1 score για την εξαγωγή συμπερασμάτων.

## **3.2 GAN**

Στην προσπάθεια περαιτέρω βελτίωσης του ταξινομητή ως προς την κλάση Normal, η οποία είναι το σημείο που το μοντέλο υστερεί, δημιουργήθηκε ένα GAN δίκτυο με σκοπό την παραγωγή νέων συνθετικών δειγμάτων. Στη συνέχεια, αυτά προστέθηκαν στο αρχικό dataset, με επανάληψη της διαδικασίας εκπαίδευσης του ταξινομητή.

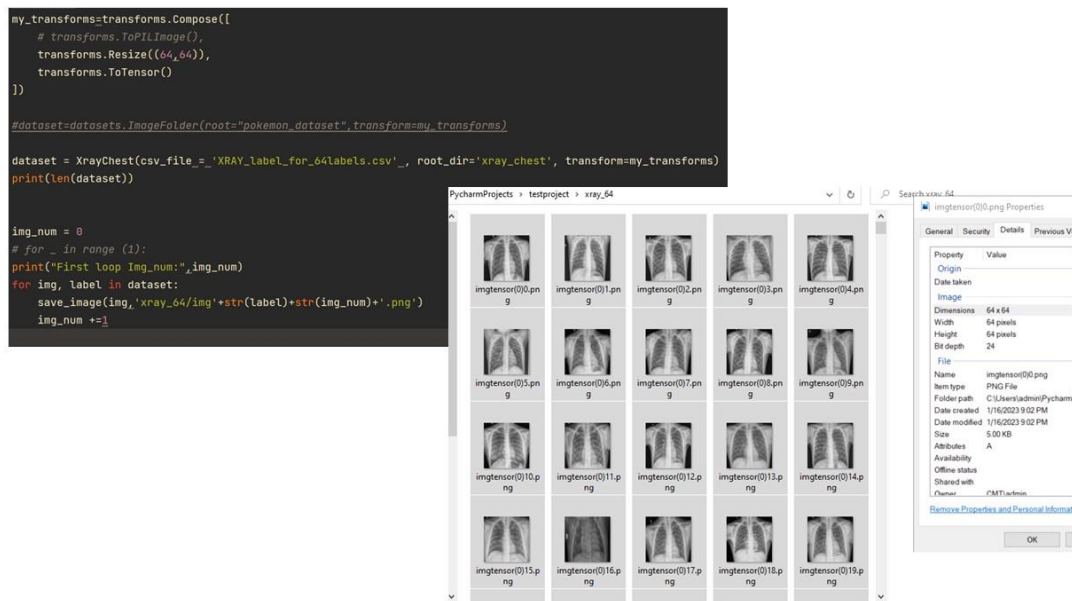
### **3.2.1 Δημιουργία GAN**

#### *Προεπεξεργασία*

Αρχικά, με σκοπό την έναρξη της διαδικασίας δημιουργίας του GAN έγινε επεξεργασία του dataset. Για λόγους ευκολίας, κυρίως όσον αφορά τη χρήση του PyTorch, συγκεντρώθηκαν όλες οι εικόνες από τους φακέλους *train* και *test* (Εικόνα 59) σε έναν φάκελο με όνομα `xray_chest`. Στον φάκελο αυτόν έγινε αντιστοίχιση ενός αρχείου `csv` που δημιουργήθηκε με τη βοήθεια του Jupyter

Notebook, το οποίο περιελάμβανε το όνομα της εικόνας και το label της κλάσης στην οποία ανήκει.

Εφόσον υπήρχε ο φάκελος με τις εικόνες συγκεντρωμένες, καθώς και το αρχείο με τις αντίστοιχες ετικέτες και χρησιμοποιώντας την κλάση XrayChest εφαρμόστηκε μετασχηματισμός αλλαγής διαστάσεων, γιατί όλες οι εικόνες θα πρέπει να έχουν το ίδιο μέγεθος πριν περάσουν στο GAN. Οι διαστάσεις των εικόνων που επιλέχθηκαν ήταν 64x64, καθώς οποιοδήποτε άλλο μέγεθος ήταν μη διαχειρίσιμο με βάση τους πόρους που ήταν διαθέσιμοι (Εικόνα 36).



Εικόνα 36 Μετασχηματισμός dataset σε 64x64

Στη συνέχεια, ακολούθησε η κατασκευή του δικτύου GAN χρησιμοποιώντας το PyTorch framework και τις βιβλιοθήκες του. Η αρχιτεκτονική που επιλέχθηκε είναι των Deep Convolutional GANs (DCGAN), δεδομένης της καλύτερης απόδοσής τους σε δεδομένα εικόνων.

### 3.2.2 Discriminator

Πρώτα, ξεκίνησε η κατασκευή του δικτύου του Discriminator (Εικόνα 37) δημιουργώντας μια κλάση.

```

import torch.nn as nn
class Discriminator(nn.Module):
    def __init__(self, channels_img, features_d):
        super(Discriminator, self).__init__()
        self.disc = nn.Sequential(
            #input: N x channels_img x 64 x 64
            nn.Conv2d(channels_img, features_d, kernel_size=4, stride=2, padding=1), #32x32
            nn.LeakyReLU(0.2),
            self._block(features_d, features_d*2, 4, 2, 1), #16x16
            self._block(features_d*2, features_d * 4, 4, 2, 1), #8x8
            self._block(features_d*4, features_d * 8, 4, 2, 1), #4x4
            nn.Conv2d(features_d*8, 1, kernel_size=4, stride=2, padding=0), #1x1
            nn.Sigmoid()
        )
    def _block(self, in_channels, out_channels, kernel_size, stride, padding):
        return nn.Sequential(
            nn.Conv2d(
                in_channels,
                out_channels,
                kernel_size,
                stride,
                padding,
                bias=False,
            ),
            nn.BatchNorm2d(out_channels),
            nn.LeakyReLU(0.2),
        )
    def forward(self, x):
        return self.disc(x)

```

Εικόνα 37 GAN-Discriminator

Η κλάση του Discriminator αποτελεί υποκλάση της `nn.Module`<sup>1</sup> και παίρνει σαν όρισμα δυο παραμέτρους. Η μια είναι ο αριθμός των καναλιών (`channel_img`) της εισαγόμενης εικόνας και η άλλη ο αριθμός των καναλιών που επρόκειτο να αλλάζουν από το ένα επίπεδο στο επόμενο στο δίκτυο. Στη συνέχεια, δημιουργήθηκαν τα διάφορα επίπεδα του δικτύου με τη βοήθεια της `nn.Sequential`, τα οποία ήταν κυρίως συνελικτικά επίπεδα ακολουθούμενα από συναρτήσεις ενεργοποίησης.

### *Τρόπος Λειτουργίας*

Στην αρχή, υπάρχει ένα συνελικτικό επίπεδο το οποίο δέχεται σαν είσοδο την εικόνα με τον αριθμό των καναλιών της και εφαρμόζει συνέλιξη 2-διαστάσεων με μέγεθος φίλτρου (kernel size) 4x4, stride 2 και padding 1. Αυτό το επίπεδο μειώνει το μέγεθος των διαστάσεων της εικόνας εισόδου σε 32x32, ενώ

<sup>1</sup> <https://pytorch.org/docs/stable/generated/torch.nn.Module.html>

ταυτόχρονα αυξάνει τον αριθμό των καναλιών (features\_disc). Στη συνέχεια, εφαρμόζεται στην εικόνα μια Leaky ReLU με αρνητική κλίση 0.2. Αυτή εισάγει μια μικρή αρνητική κλίση για αρνητικές τιμές εισόδου, επιτρέποντας στο δίκτυο να μαθαίνει από αρνητικές κλίσεις.

Για την υλοποίηση των υπολοίπων επίπεδων, τα οποία επαναλαμβάνουν ένα μεγάλο μέρος της διαδικασίας, χρησιμοποιήθηκε η βοηθητική συνάρτηση `_block`. Με την `_block` ουσιαστικά δημιουργούνται τα ενδιάμεσα επίπεδα του δικτύου του Discriminator με ίδιο ακριβώς τρόπο με πριν, με μόνη διαφορά το ότι μετά από κάθε συνελκτικό επίπεδο και πριν την εφαρμογή της Leaky ReLU υπάρχει ένα επίπεδο BatchNormalization. Η `_block` καλείται τρεις φορές στο συγκεκριμένο δίκτυο μειώνοντας τις διαστάσεις της εικόνας και διπλασιάζοντας τον αριθμό των feature\_disc σε κάθε κλήση. Μετά τις κλήσεις των `_block` υπάρχει ένα τελευταίο συνελκτικό επίπεδο όπου η διάσταση της εικόνας μειώνεται σε 1x1 χωρίς padding. Μετά, αυτή περνάει από μια Sigmoid συνάρτηση ενεργοποίησης περιορίζοντας την έξοδο σε ένα διάστημα [0-1]. Το τελικό αποτέλεσμα αντικατοπτρίζει την πιθανότητα της εικόνας να είναι αληθινή. Η εικόνα στο δίκτυο του Discriminator ακολουθεί την εξής μετατροπή (Πίνακας 3).

Πίνακας 3 Η πορεία του μετασχηματισμού της εικόνας στον Discriminator

Διαστάσεις	Κανάλια
64x64	3
32x32	64
16x16	128
8x8	256
4x4	512
1x1	1

### 3.2.3 Generator

Στη συνέχεια, κατασκευάστηκε το δίκτυο του Generator δημιουργώντας μια δεύτερη κλάση (Εικόνα 38), η οποία αποτελεί επίσης υποκλάση της nn.Module.



```

class Generator(nn.Module):
    def __init__(self, z_dim, channels_img, features_g):
        super(Generator, self).__init__()
        self.gen = nn.Sequential(
            #input :N x z_dim x1 x1
            self._block(z_dim, features_g*16, 4, 1, 0), #N x f_g*16 x 4 x4
            self._block(features_g*16, features_g*8, 4, 2, 1), #8x8
            self._block(features_g*8, features_g*4, 4, 2, 1), #16x16
            self._block(features_g * 4, features_g * 2, 4, 2, 1), # 32x32
            nn.ConvTranspose2d(features_g*2, channels_img, kernel_size=4, stride=2, padding=1,
            ), #64x64 original image size
            nn.Tanh(), #[-1,1]
        )

    def _block(self, in_channels, out_channels, kernel_size, stride, padding):
        return nn.Sequential(
            #upsampinig for this reason convtranspose2s
            nn.ConvTranspose2d(
                in_channels,
                out_channels,
                kernel_size,
                stride,
                padding,
                bias=False,
            ),
            nn.BatchNorm2d(out_channels),
            nn.ReLU(),
        )

    def forward(self, x):
        return self.gen(x)

```

Εικόνα 38 GAN-Generator

Το δίκτυο του Generator αρχικοποιείται από τις εξής παραμέτρους:

Η πρώτη παράμετρος είναι μια μεταβλητή  $z\_dim$ , η οποία αντιπροσωπεύει τη διάσταση του διανύσματος θορύβου ( $z$ ). Είναι ουσιαστικά το μέγεθος της τυχαίας εισόδου που χρησιμοποιείται για τη δημιουργία των εικόνων. Η δεύτερη όπως και πριν είναι ο αριθμός των καναλιών που είναι επιθυμητό να έχει στο τέλος η παραγόμενη εικόνα. Στην περίπτωση της παρούσας εργασίας, ορίστηκε η τιμή 3 (RGB που αφορά έγχρωμη εικόνα). Η τρίτη παράμετρος είναι μια μεταβλητή που ονομάζεται  $feature\_g$  και ορίζει τον αριθμό των καναλιών σε κάθε επίπεδο του δικτύου του Generator. Η τιμή που δόθηκε σε αυτήν τη μεταβλητή είναι 64. Χρησιμοποιήθηκε και πάλι η `nn.Sequential` για την κατασκευή των επιπέδων του δικτύου καθώς και η βοηθητική συνάρτηση `_block`. Με τη βοήθεια αυτών των δύο συναρτήσεων, δημιουργήθηκε ένα σύνολο διαδοχικών επιπέδων και συναρτήσεων που εκτελούνται στη σειρά. Το `block` που δημιουργήθηκε περιλαμβάνει ένα transposed Convolution (`ConvTranspose2d`)

επίπεδο, ακολουθούμενο από μια συνάρτηση Batch Normalization (BatchNorm2d) και τέλος μια συνάρτηση ενεργοποίησης ReLU.

#### *Τρόπος Λειτουργίας*

Στο πρώτο επίπεδο του δικτύου που χρησιμοποιείται η `_block`, δίνεται ως είσοδος το `z_dim` με τιμή 100, το οποίο μέσω του πρώτου επιπέδου της `_block` που είναι ένα transposed συνελικτικό επίπεδο (`nn.ConvTranspose2d`), το οποίο επεκτείνει σε 1024 κανάλια ( $features\_g * 16 = 64 * 16 = 1024$ ) με μέγεθος πυρήνα (kernel size) 4x4, stride 1 και padding 0. Στη συνέχεια, εκτελείται κανονικά Batch Normalization και η συνάρτηση ReLU όπως αναφέρεται στο `_block`.

Ακολουθώντας, καλούνται διαδοχικά πολλαπλές συναρτήσεις `_block` λειτουργώντας με την παραπάνω λογική και μειώνοντας στο μισό τον αριθμό των καναλιών αυξάνοντας ταυτόχρονα τις διαστάσεις της εικόνας. Το kernel size παραμένει 4x4, το stride από 1 γίνεται 2, ενώ το padding από 0 αυξάνεται σε 1. Μόλις η εικόνα μέσω των διαδοχικών κλίσεων της `_block` φτάσει το μέγεθος 32x32, χρησιμοποιείται ένα `ConvTranspose2d` επίπεδο για τη μετατροπή των διαστάσεων της εικόνας από 32x32 και 128 κανάλια σε 64x64 και 3 κανάλια.

Πρέπει να σημειωθεί, ότι ο λόγος που δεν χρησιμοποιείται η `_block` εδώ είναι επειδή δεν είναι επιθυμητή η εκτέλεση των συναρτήσεων Batch Norm2d και ReLU στο τελευταίο σημείο. Τέλος, η έξοδος που προκύπτει από τον Generator συμπιέζεται με τη χρήση της συνάρτησης ενεργοποίησης Tanh, η οποία απεικονίζει τις τιμές των εικονοστοιχείων στο εύρος [-1, 1].

Πίνακας 4 Η πορεία του μετασχηματισμού του διανύσματος θορύβου στο Generator και η μετατροπή τους σε μια εικόνα 64x64

Διαστάσεις	Κανάλια
1x1	100
4x4	1024
8x8	512
16x16	256
32x32	128
64x64	3

Μετά τις δύο κλάσεις του Discriminator και του Generator στο ίδιο αρχείο `model.py` υπάρχουν δυο ακόμα συναρτήσεις η `forward` και η `initialize_weights`. Η `forward` υλοποιεί το εμπρόσθιο πέρασμα του δικτύου του Discriminator ή του Generator ανάλογα σε ποιο βρισκόμαστε και λειτουργεί παίρνοντας ως όρισμα

ένα τανιστή εισόδου  $x$  τον οποίο τον περνάει από το αντίστοιχο δίκτυο (Discriminator/Generator), επιστρέφοντας την παραγόμενη έξοδο.

Η *initialize\_weights* είναι υπεύθυνη για την αρχικοποίηση των βαρών των μονάδων μέσα στο συγκεκριμένο νευρωνικό δίκτυο και παίρνει ως όρισμα εισόδου το μοντέλο και προσπελάζει όλα τα επίπεδα του μοντέλου. Αν βρεθεί σε ένα επίπεδο που αποτελεί `nn.Conv2d`, `nn.ConvTranspose2d` ή `nn.BatchNorm2d`, τότε αρχικοποιεί τα βάρη. Τα βάρη στην περίπτωση αυτή αρχικοποιούνται με τιμές που προέρχονται από κανονική κατανομή με μέση τιμή (mean) 0,0 και τυπική απόκλιση (standard deviation) 0,02. Ο καθορισμός του μέσου όρου της κανονικής κατανομής σε 0,0 εξασφαλίζει ότι τα αρχικά βάρη είναι κεντραρισμένα γύρω από το μηδέν, στοιχείο που βοηθά στη διατήρηση της συμμετρίας και αποτρέπει την αρχική προτίμηση κάποιας συγκεκριμένης κατεύθυνσης. Από την άλλη, η επιλογή μια μικρής τυπικής απόκλισης, όπως η τιμή 0,02, βοηθάει στο να διατηρούνται τα αρχικά βάρη σχετικά κοντά στο μηδέν και εντός ενός λογικού εύρους. Επομένως, αποτρέπει τα αρχικά βάρη από το να είναι πολύ μεγάλα ή πολύ μικρά, γεγονός που μπορεί να οδηγήσει σε προβλήματα που αναφέρθηκαν σε προηγούμενο Κεφάλαιο, όπως η εξαφάνιση των κλίσεων (Vanishing Gradient) κατά τη διάρκεια της εκπαίδευσης.

### 3.2.4 Εκπαίδευση DCGAN

Μόλις ολοκληρώθηκε το DCGAN, δημιουργήθηκε ο αλγόριθμος εκπαίδευσης. Το πρώτο βήμα ήταν να οριστούν ορισμένες υπερπαραμέτροι, απαραίτητες για την αρχικοποίηση του DCGAN (Generator μοντέλο-Discriminator μοντέλο), αλλά και ταυτόχρονα και για τη διαδικασία της εκπαίδευσής του.

Πίνακας 5 Αρχικοποίηση υπερπαραμέτρων DCGAN

<b>Υπερπαραμέτρος</b>	<b>Τιμή</b>
Device	cuda
Learning Rate	0.0002
Batch Size	16
Image Size	64
Number of Channels	3
Z Dimension	100
Number of Epochs	5
Discriminator Features	64
Generator Features	64

Αναλυτικότερα, επιλέχθηκε η GPU (cuda) έναντι της cpu, λόγω του ότι οι GPUs προσφέρουν ταχύτερους υπολογισμούς σε σύγκριση με τις CPU, επιταχύνοντας

έτσι τη διαδικασία εκπαίδευσης. Για ρυθμό μάθησης (learning rate) επιλέχθηκε η τιμή 0.0002, καθώς υψηλότεροι ρυθμοί μάθησης ενδέχεται να οδηγήσουν σε ασταθείς ενημερώσεις ή αποκλίσεις κατά τη διάρκεια της εκπαίδευσης. Αυτή η τιμή χρησιμοποιείται συνήθως στα GAN για την εξισορρόπηση της ταχύτητας σύγκλισης και της σταθερότητας. Ως μέγεθος παρτίδας (Batch Size) επιλέχθηκε η τιμή 16, καθώς μεγαλύτερες τιμές στο batch size οδηγούσαν σε προβλήματα μνήμης (χωρητικότητα) ή αυξάναν κατά πολύ τον χρόνο εκπαίδευσης. Για μέγεθος εικόνας (Image Size) επιλέχθηκε η τιμή 64, καθώς οι εικόνες που θα τροφοδοτηθεί το δίκτυο θα είναι 64x64. Το συγκεκριμένο μέγεθος, όπως αναφέρθηκε και παραπάνω, έχει επιλεγεί για τον λόγο ότι μεγαλύτερα μεγέθη εικόνων δεν ήταν εύκολα διαχειρίσιμα λόγω έλλειψης πόρων.

Για τη διάσταση του διανύσματος θορύβου εισόδου Z (Z dimension), επιλέχθηκε η τιμή 100, γιατί χρησιμοποιείται ευρέως στη βιβλιογραφία ως μια από τις πιο αποτελεσματικές τιμές, καθώς παρέχει αρκετή ποικιλία έτσι ώστε να βοηθήσει τον generator να αποκτήσει ποικιλία στις εξόδους του. Για τον αριθμό εποχών επιλέχθηκε η τιμή 5, καθώς ήταν προτιμότερο να παραμείνει σε χαμηλά επίπεδα ο χρόνος εκπαίδευσης.

Τέλος, για τις παραμέτρους Discriminator Features και Generator Features επιλέχθηκε η τιμή 64 και αναφέρονται στον αριθμό των χαρακτηριστικών στα συνελκτικά επίπεδα του Discriminator και του Generator. Αυτή η τιμή μπορεί να επηρεάσει την αναπαραστατική ικανότητα του δικτύου. Μια τιμή σαν το 64 μπορεί να επιτύχει μια ισορροπία μεταξύ της πολυπλοκότητας του μοντέλου και της αποτελεσματικότητας της εκπαίδευσης για το συγκεκριμένο πρόβλημα. Επίσης, αξίζει να σημειωθεί ότι οι συγκεκριμένες παράμετροι εξαρτώνται άμεσα από το μέγεθος της εικόνας.

Στη συνέχεια, με βάση τις παραπάνω υπερπαραμέτρους (Πίνακας 5) αρχικοποιήθηκε το δίκτυο του Generator (gen), και του Discriminator (disc) και των βελτιστοποιητών τους (opt\_gen και opt\_disc). Επίσης, χρησιμοποιώντας τη συνάρτηση ImageFolder από την torchvision έγινε φόρτωση του dataset εκπαίδευσης και δημιουργήθηκε ένας φορτωτής δεδομένων (data loader) για την προσπέλαση του συνόλου δεδομένων σε παρτίδες (batches).

```

# dataset= datasets.MNIST(root="dataset/",train=True,
transform=transforms.download=True)
dataset=datasets.ImageFolder(root="xray_64_v2",transform=transforms)
# dataset=datasets.ImageFolder(root="celeb_dataset",transform=transforms)
loader = DataLoader(dataset,batch_size=BATCH_SIZE,shuffle=True)
gen = Generator(Z_DIM,CHANNELS_IMG,FEATURES_GEN).to(device)
disc = Discriminator(CHANNELS_IMG,FEATURES_DISC).to(device)
initialize_weights(gen)
initialize_weights(disc)
opt_gen = optim.Adam(gen.parameters()),lr=LEARNING_RATE,betas=(0.5, 0.999))
opt_disc = optim.Adam(disc.parameters()),lr=LEARNING_RATE,betas=(0.5, 0.999))
criterion=nn.BCELoss()
#fixed noise for see progression of training
fixed_noise = torch.randn(32,Z_DIM,1,1).to(device)
#For tensorboard
writer_real = SummaryWriter("logs/real")
writer_fake = SummaryWriter("logs/fake")
step=0
#Set on training mode both
gen.train()
disc.train()

```

```

for epoch in range(NUM_EPOCHS):
for batch_idx, (real, _) in enumerate(loader):
real=real.to(device)
noise=torch.randn(BATCH_SIZE, NOISE_DIM, 1, 1).to(device)
fake=gen(noise)
#Train Discriminator: max log(D(x)) + log(1-D(z))
disc_real=disc(real)
disc_real=disc_real.reshape(-1) #N X 1 X 1 X 1
loss_disc_real=criterion(disc_real,torch.ones_like(disc_real))
disc_fake = disc(fake.detach()).reshape(-1)
loss_disc_fake=criterion(disc_fake,torch.zeros_like(disc_fake))
loss_disc=(loss_disc_real+loss_disc_fake)/2
disc.zero_grad()
loss_disc.backward(retain_graph=True)
opt_disc.step()
===Train: Generator min log(1-D(Gz)) <-> max log(D(Gz))
output= disc(fake).reshape(-1)
loss_gen = criterion(output,torch.ones_like(output))
gen.zero_grad()
loss_gen.backward()
opt_gen.step()
# Print losses occasionally and print to tensorboard
if batch_idx % 100 == 0:
print(
f'Epoch [{epoch}/{NUM_EPOCHS}] Batch {batch_idx}/{len(loader)} \
Loss D: {loss_disc:.4f}, loss G: {loss_gen:.4f}'
)
with torch.no_grad():
fake = gen(fixed_noise)
# take out (up to) 32 samples
img_grid_real = torchvision.utils.make_grid(
real[:32],normalize=True
)
img_grid_fake = torchvision.utils.make_grid(
fake[:32],normalize=True
)
writer_real.add_image("Real",img_grid_real,global_step=step)
writer_fake.add_image("Fake",img_grid_fake,global_step=step)
save_image(img_grid_fake, f'gridpict/normalimg + str(epoch) + ".jpg")
save_image(fake, f'gridpict/fakeimage + str(epoch) + ".jpg")
step += 1

```

Εικόνα 39 DCGAN -Αλγόριθμος Εκπαίδευσης

### Βρόγχος Εκπαίδευσης

Η διαδικασία εκπαίδευσης ακολουθεί έναν βρόχο που εκτελείται για έναν καθορισμένο αριθμό εποχών. Μέσα σε κάθε εποχή, ο αλγόριθμος επαναλαμβάνει κάθε παρτίδα (batch) πραγματικών εικόνων (real) και εκτελεί τα ακόλουθα βήματα:

**Αρχικά η γεννήτρια παράγει ψεύτικες εικόνες (fake)** περνώντας έναν τυχαίο θόρυβο (noise) μέσω του μοντέλου του Generator (gen).

**Εκπαίδευση δικτύου Discriminator:** Ο Discriminator εκπαιδεύεται για να διακρίνει μεταξύ πραγματικών και ψεύτικων εικόνων. Υπολογίζει τις εξόδους του διαχωριστή τόσο για τις πραγματικές εικόνες (disc\_real), όσο και για τις ψεύτικες (disc\_fake). Ο στόχος του Discriminator είναι να μεγιστοποιήσει τις λογαριθμικές τιμές για τις πραγματικές εικόνες και να ελαχιστοποιήσει τις λογαριθμικές τιμές για τις ψεύτικες εικόνες.

Η απώλεια (loss\_disc) του Discriminator υπολογίζεται συγκρίνοντας τις εξόδους του με τις αντίστοιχες ετικέτες. Για τις πραγματικές εικόνες, οι ετικέτες τίθενται σε 1 (torch.ones\_like(disc\_real)), υποδεικνύοντας ότι ο Discriminator πρέπει να τις ταξινομήσει ως πραγματικές. Για τις ψεύτικες εικόνες, οι ετικέτες τίθενται σε 0 (torch.zeros\_like(disc\_fake)), υποδεικνύοντας ότι ο Discriminator πρέπει να τις ταξινομήσει ως ψεύτικες. Οι παράμετροι του Discriminator ενημερώνονται

με οπισθοδιάδοση (backpropagation) της απώλειας του Discriminator και κλήση της `opt_disc.step()` για την εκτέλεση του βήματος βελτιστοποίησης.

### ***Εκπαίδευση δικτύου Generator***

Στη συνέχεια ο Generator εκπαιδεύεται ώστε να παράγει πιο ρεαλιστικές εικόνες που μπορούν να ξεγελάσουν τον Discriminator και υπολογίζει την έξοδο του για τις ψεύτικες εικόνες που παράγονται από τον Generator (output). Η απώλεια (`loss_gen`) του Generator υπολογίζεται συγκρίνοντας την έξοδο του Discriminator με την ετικέτα 1 (`torch.ones_like(output)`). Ο στόχος του Generator είναι η μεγιστοποίηση του λογαρίθμου της εξόδου του Discriminator όταν δίνονται ψεύτικα/παραγόμενα δείγματα, υποδεικνύοντας έτσι ότι οι παραγόμενες εικόνες πρέπει να ταξινομηθούν ως πραγματικές.

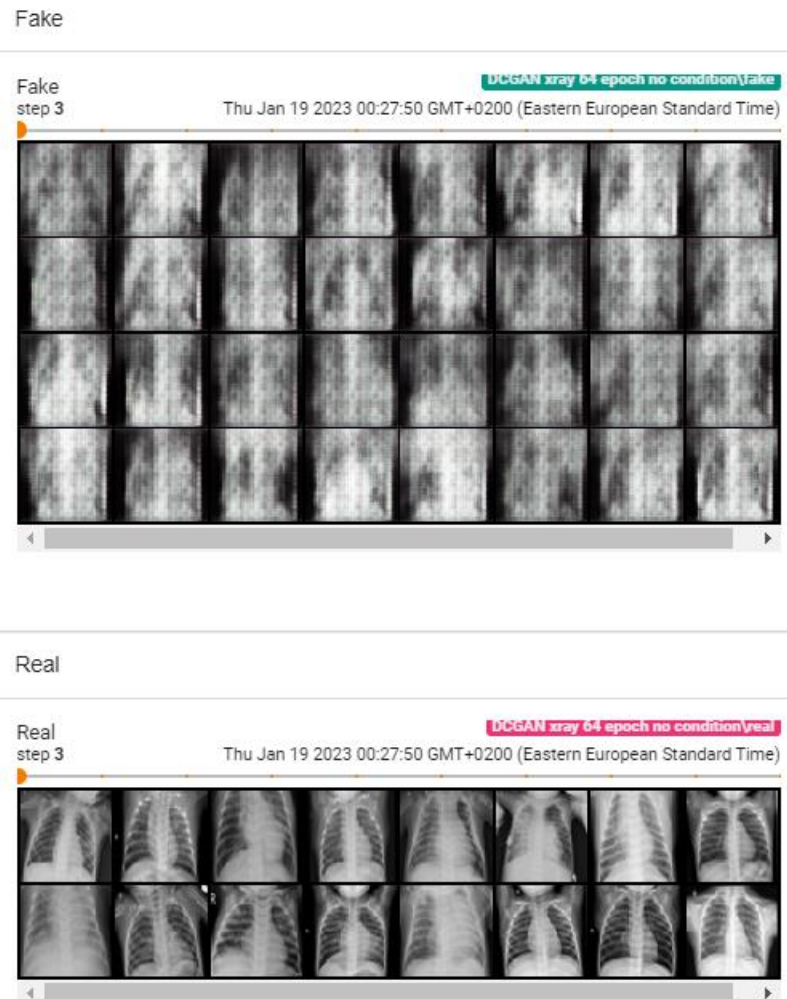
Όσον αφορά τις παραμέτρους του Generator, αυτές ενημερώνονται, όπως και στον Discriminator δηλαδή με την οπισθοδιάδοση της απώλειας (`loss_gen`) και την κλήση της συνάρτησης `opt_gen.step()`, η οποία χρησιμοποιεί τον optimizer για βελτιστοποίηση των παραμέτρων.

Τέλος, με σκοπό την καλύτερη παρακολούθηση της εξέλιξης της διαδικασίας εκπαίδευσης του DCGAN μας, πραγματοποιείται εκτύπωση ανά τακτά χρονικά διαστήματα της προόδου της εκπαίδευσης, εμφανίζοντας την εποχή στην οποία βρίσκεται η εκπαίδευση, τον δείκτη παρτίδας (`batch_idx`) και τις τρέχουσες απώλειες του Discriminator και Generator. Επίσης, με τη χρήση του TensorBoard, οπτικοποιούνται οι πραγματικές και ψεύτικες εικόνες στην οθόνη έχοντας έτσι τη δυνατότητα ζωντανής παρακολούθησης των παραγόμενων εικόνων και την εξέλιξη που έχουν με την πάροδο των επαναλήψεων.

Για την πρώτη δοκιμή του GAN χρησιμοποιήθηκαν δυο διαδομένα dataset εικόνων, όπως το MNIST dataset που αποτελείται από ασπρόμαυρες εικόνες μικρού μεγέθους που απεικονίζουν χειρόγραφους αριθμούς. Το δεύτερο ήταν το Celeb Dataset που περιλαμβάνει έγχρωμες εικόνες διάσημων προσώπων. Αφού εκπαιδεύτηκε το δίκτυο χρησιμοποιώντας τις παραπάνω υπερπαραμέτρους (Πίνακας 5) (αλλάζοντας μόνο το Number of Channels από 3 σε 1 στην περίπτωση του Mnist) και επαληθευτική η λειτουργία του αλγορίθμου στα δυο αυτά δοκιμαστικά dataset, στη συνέχεια, επιχειρήθηκε εφαρμογή της εκπαίδευσης στο dataset με τις ακτινογραφίες θώρακα. Η μόνη αλλαγή που έγινε στις υπερπαραμέτρους για το συγκεκριμένο dataset ήταν ο αριθμός των εποχών

που ορίστηκε στις 64 από 5, καθώς υπήρχαν διαθέσιμες λιγότερες εικόνες σε σχέση με τα προηγούμενα dataset.

Δοκιμή σε Chest Xray dataset:



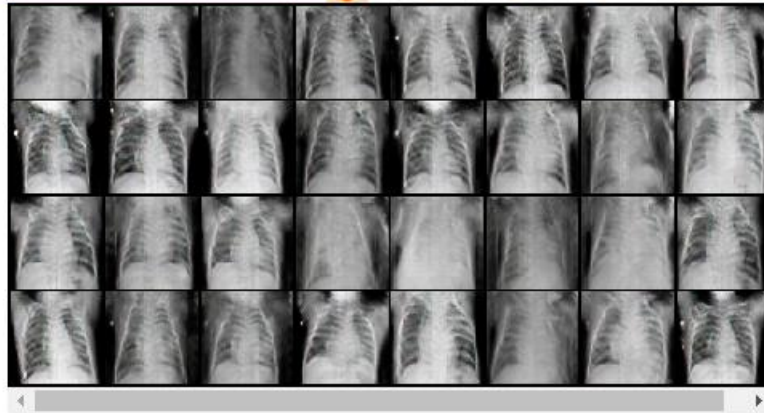
Εικόνα 40 Εκπαίδευση DCGAN σε Chest Xray Dataset- Αρχή

Fake

Fake  
step 53

DCGAN xray 64 epoch no condition/fake

Thu Jan 19 2023 00:49:19 GMT+0200 (Eastern European Standard Time)

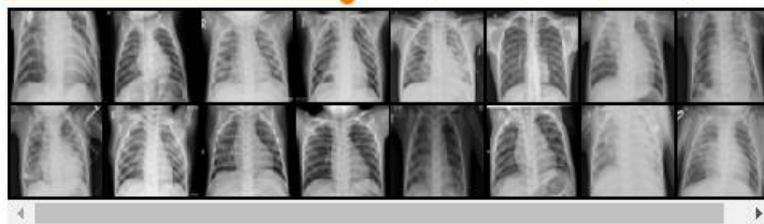


Real

Real  
step 53

DCGAN xray 64 epoch no condition/real

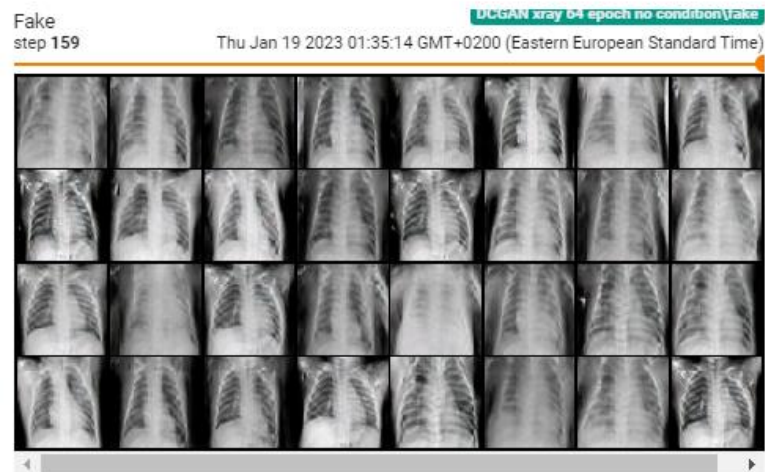
Thu Jan 19 2023 00:49:19 GMT+0200 (Eastern European Standard Time)



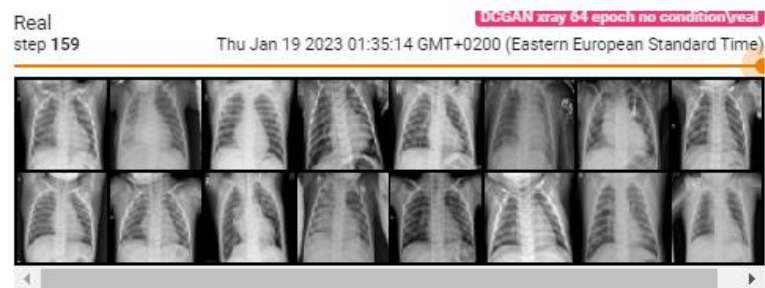
Εικόνα 41 Εκπαίδευση DCGAN σε Chest Xray Dataset- Μετά από μερικές επαναλήψεις



Fake



Real



Εικόνα 42 Εκπαίδευση DCGAN σε Chest Xray Dataset Τέλος εκπαίδευσης

Όπως φαίνεται και από τις εικόνες της παραπάνω δοκιμής το DCGAN έχει μετατρέψει το διάνυσμα θορύβου σε εικόνες ακτινογραφίας θώρακα. Επομένως, το γεγονός αυτό επιβεβαιώνει ότι ο αλγόριθμος είναι σε θέση να δεχτεί περαιτέρω βελτίωση. Στη συνέχεια, παρουσιάζεται ο τρόπος με τον οποίο βελτιώθηκε ο αλγόριθμος εκπαίδευσης χρησιμοποιώντας τις αρχιτεκτονικές WGAN και WGAN-GP, καθώς και ο τρόπος που ενσωματώθηκε η κλάση στο GAN για να δημιουργηθεί το τελικό Conditional GAN, το οποίο θα χρησιμοποιηθεί για τη δημιουργία συνθετικών δειγμάτων για την κλάση Normal του ταξινομητή που παρουσιάζεται στα προηγούμενα πειράματα ότι υστερεί.

### 3.2.5 WGAN & WGAN-GP

Με σκοπό τη μελέτη της τελικής αρχιτεκτονικής που θα χρησιμοποιηθεί επιχειρείται η εφαρμογή της αρχιτεκτονικής WGAN. Με βάση τη μελέτη 36, ένα

WGAN μπορεί να εξασφαλίσει μεγαλύτερη σταθερότητα κατά την εκπαίδευση του δικτύου, να γλιτώσει από πολύωρες δοκιμές παίζοντας με τις υπερπαραμέτρους και να αποφύγει προβλήματα που μπορεί να εμφανιστούν κατά τη διάρκεια της εκπαίδευσης, όπως το mode collapse. Όπως έχει αναφερθεί και στο θεωρητικό υπόβαθρο, ένα WGAN προσφέρει μεγαλύτερη σταθερότητα, αλλά θέλει περισσότερο χρόνο να εκπαιδευτεί. Επίσης, στο WGAN, η απώλεια στο τέλος της εκπαίδευσης υποδεικνύει ευρήματα για το αποτέλεσμα.

Έχοντας υπόψη όλα τα παραπάνω και από τη στιγμή που το dataset έχει να κάνει με δεδομένα υγείας και εικόνες ακτινογραφίας, θεωρήθηκε καταλληλότερη μια αρχιτεκτονική, η οποία παρόλο που χρειάζεται αρκετό χρόνο εκπαίδευσης, τα αποτελέσματα είναι όσο το δυνατόν πιο σίγουρα.

Αρχικά, εκπαιδεύτηκε το WGAN με τη μέθοδο Weight Clipping. Οι υπερπαραμέτροι που χρησιμοποιήθηκαν για την εκπαίδευση είναι οι ίδιες που χρησιμοποιήθηκαν και για την εκπαίδευση του DCGAN (Πίνακας 5) διαφέροντας μόνο ως προς την τιμή του Learning Rate, όπου εδώ επιλέχθηκε η τιμή 0.00005 έναντι της τιμής 0.0002. Επίσης για την εφαρμογή της μεθόδου Weight Clipping ήταν αναγκαία η προσθήκη της υπερπαραμέτρου Weight Clip στην οποία δώσαμε την τιμή 0.01, καθώς αναφέρεται στη βιβλιογραφία ότι είναι ένα κοινό κατώφλι που εξασφαλίζει σταθερότητα στη διαδικασία εκπαίδευσης του Wasserstein GAN.

Τέλος, χρειάστηκε να προστεθεί και άλλη μία υπερπαραμέτρος η Critic Iterations, η οποία αφορά στον αριθμό επαναλήψεων που εκπαιδεύεται ο critic (discriminator) πριν από την ενημέρωση των βαρών του Generator. Αυτή η τιμή ορίζεται για να διασφαλιστεί μια πιο σταθερή και ανθεκτική διαδικασία εκπαίδευσης. Ο αριθμός 5 ως αριθμός εποχών που επιλέχθηκε αποτελεί ένα κοινό σημείο εκκίνησης και έχει διαπιστωθεί ότι λειτουργεί καλά στην πράξη για αρκετές εφαρμογές. Επίσης, η επιλογή σχετικά μικρής τιμής ήταν αποτέλεσμα του σκεπτικού ότι ένας μεγαλύτερος αριθμός επαναλήψεων του critic θα επιβράδυνε τη διαδικασία εκπαίδευσης κατά πολύ.

Οι παραπάνω υπερπαραμέτροι WGAN με Weight Clipping παρουσιάζονται στον ακόλουθο πίνακα (Πίνακας 6).

Πίνακας 6 Υπερπαράμετροι εκπαίδευσης WGAN με Weight Clipping

Υπερπαράμετρος	Τιμή
Device	"cuda" if available, otherwise "cp"
Learning Rate	0.00005
Batch Size	16
Image Size	64
Number of Channels	3
Z Dimension	100
Number of Epochs	5
Discriminator Features	64
Generator Features	64
Critic Iterations	5
Weight Clip	0.01

Ο αλγόριθμος εκπαίδευσης που υλοποιήθηκε αναφέρεται αναλυτικά στο θεωρητικό υπόβαθρο (Εικόνα 24). Αυτό που πρέπει να αναφερθεί εδώ, ωστόσο, είναι ότι για να τρέξει το WGAN τροποποιήθηκε το μοντέλο, αφαιρώντας την Sigmoid function από το τελευταίο επίπεδο του δικτύου του Discriminator (Critic), καθώς ήταν προτιμότερο η έξοδος να παίρνει τιμές από το 0 έως το 1 όπως έκανε και πριν.

Η μέθοδος του Weight Clipping σύμφωνα με την έρευνα [51] αποτελεί μια πολύ κακή πρακτική για την επιβολή του περιορισμού Lipschitz, καθώς η παράμετρος clipping είτε είναι πολύ μεγάλη, είτε πολύ μικρή και συνεπώς μπορεί να δημιουργήσει προβλήματα. Γι' αυτόν τον λόγο, υλοποιήθηκε το WGAN και με τη μέθοδο του Gradient Penalty, η οποία σύμφωνα με τη σχετική δημοσίευση [86] επιβάλλει με πολύ καλύτερο τρόπο τον περιορισμό Lipschitz.

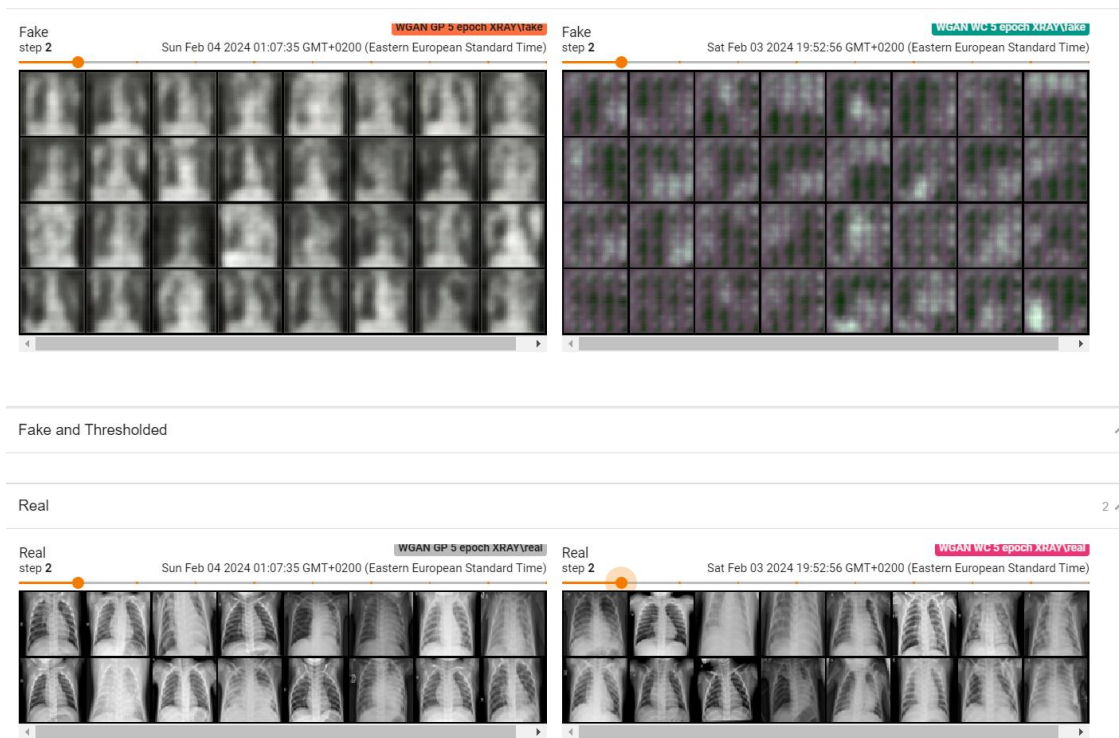
Οι υπερπαράμετροι που χρησιμοποιήθηκαν για την εκπαίδευση του WGAN-GP είναι όμοιοι με τις υπερπαράμετρους που χρησιμοποιήσαμε στο WGAN με Weight-Clipping (Πίνακας 6) διαφέροντας μόνο ως προς την τιμή του ρυθμού μάθησης (learning rate) όπου επιλέχθηκε η τιμή 0.0001. Η τιμή αυτή προήλθε εμπειρικά κατά τη διάρκεια του συντονισμού των υπερπαραμέτρων και οδηγούσε σε καλύτερη απόδοση, και διαφέροντας επίσης ως προς το γεγονός ότι τη θέση της υπερπαραμέτρου Weight Clip εδώ την παίρνει η υπερπαράμετρος  $\lambda_{GP}$  (LAMBDA-GP), η οποία είναι απαραίτητη εξ ορισμού για τον έλεγχο της ισχύος της Gradient penalty (ποινή κλίσης). Η τιμή 10 η οποία επιλέχθηκε καθορίζει το πόσο επηρεάζει ο όρος Gradient Penalty τη συνολική απώλεια. Η επιλογή της συγκεκριμένης τιμής βασίστηκε στο γεγονός ότι οι ερευνητές χρησιμοποίησαν την ίδια στη μελέτη [86]. Μια υψηλότερη τιμή υποδηλώνει

μεγαλύτερη έμφαση στην επιβολή του περιορισμού Lipschitz μέσω της Gradient Penalty. Στον παρακάτω πίνακα (Πίνακας 7) αναφέρονται οι υπερπαραμέτροι εκπαίδευσης του WGAN με τη μέθοδο Gradient Penalty.

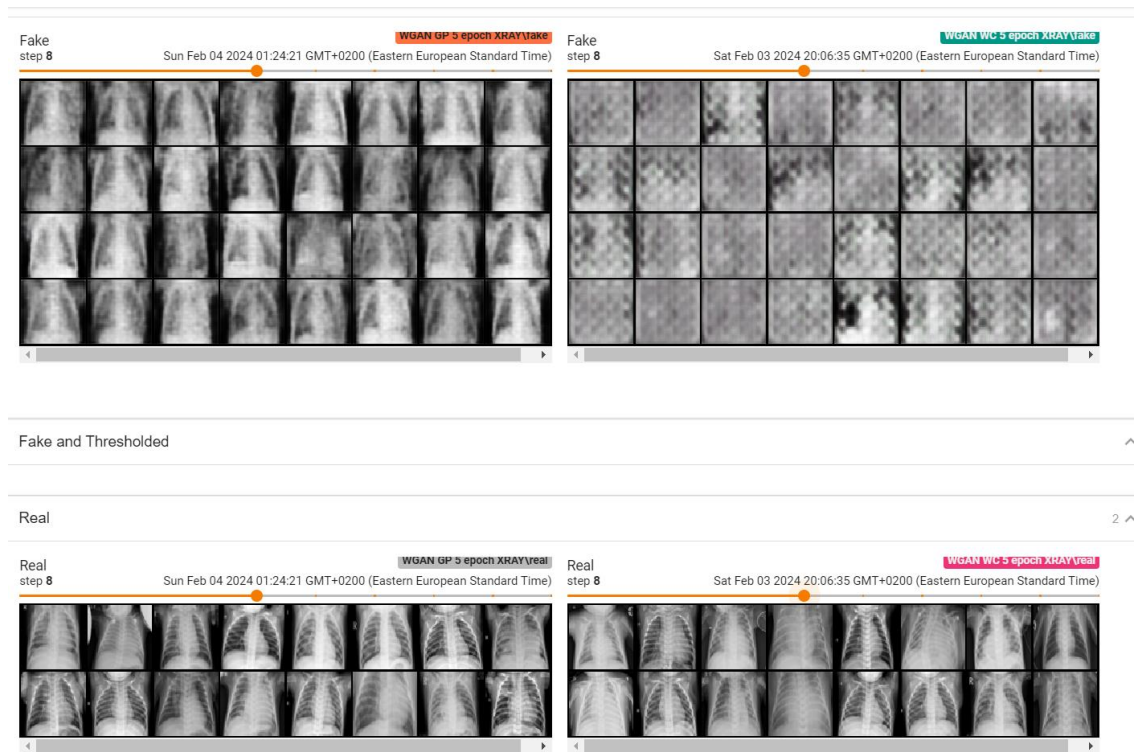
Πίνακας 7 Υπερπαραμέτροι εκπαίδευσης WGAN με Gradient Penalty

<b>Υπερπαραμέτρος</b>	<b>Τιμή</b>
Device	cuda
Learning Rate	0.0001
Batch Size	16
Image Size	64
Number of Channels	3
Z Dimension	100
Number of Epochs	5
Discriminator Features	64
Generator Features	64
Critic Iterations	5
LAMBDA GP	10

Ο αλγόριθμος εκπαίδευσης που υλοποιήθηκε αναλύεται στο θεωρητικό υπόβαθρο (Εικόνα 26). Στο μοντέλο του WGAN, αυτό που αλλάζει σε σχέση με την υλοποίηση με Weight Clipping είναι ότι στο δίκτυο του Discriminator (Critic) αντί για την BatchNorm2d() που υπάρχει στην \_block πριν από την LeakyReLU, χρησιμοποιήθηκε η InstanceNorm2d() που εφαρμόζει κανονικοποίηση σε κάθε δείγμα ξεχωριστά και όχι σε batches όπως η BatchNorm2d.

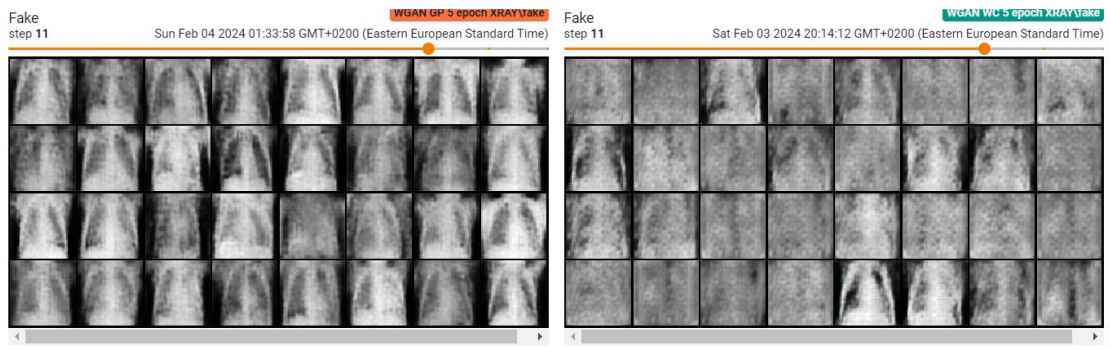


Εικόνα 43 WGAN-GP εναντίον WGAN Weight Clipping: Αρχή Εκπαίδευσης  
 Δοκιμές στο Xray Dataset: Οι δύο υλοποιήσεις του WGAN πραγματοποιήθηκαν στο Xray dataset. Στην Εικόνα 43, στο WGAN-GP (αριστερό μέρος εικόνας) στο Step 2 ο θόρυβος αρχίζει να παίρνει κάποια μορφή.



Εικόνα 44 WGAN-GP εναντίον WGAN Weight Clipping: Step 20

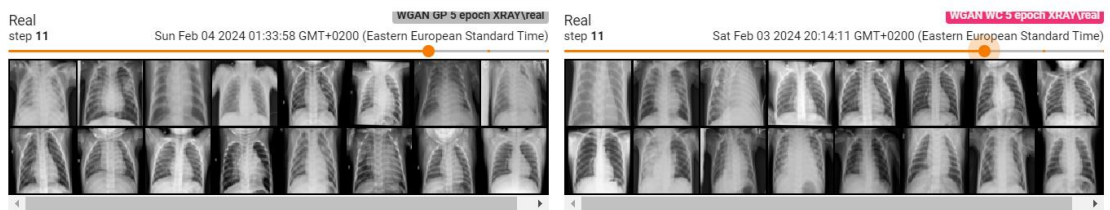
Στη συνέχεια στην Εικόνα 44 παρατηρείται ότι το WGAN-GP στο Step 8 έχει ήδη αρχίσει να σχηματίζει ακτινογραφίες, ενώ το WGAN-Weight Clipping εξακολουθεί να εμφανίζει θόρυβο.



Fake and Thresholded

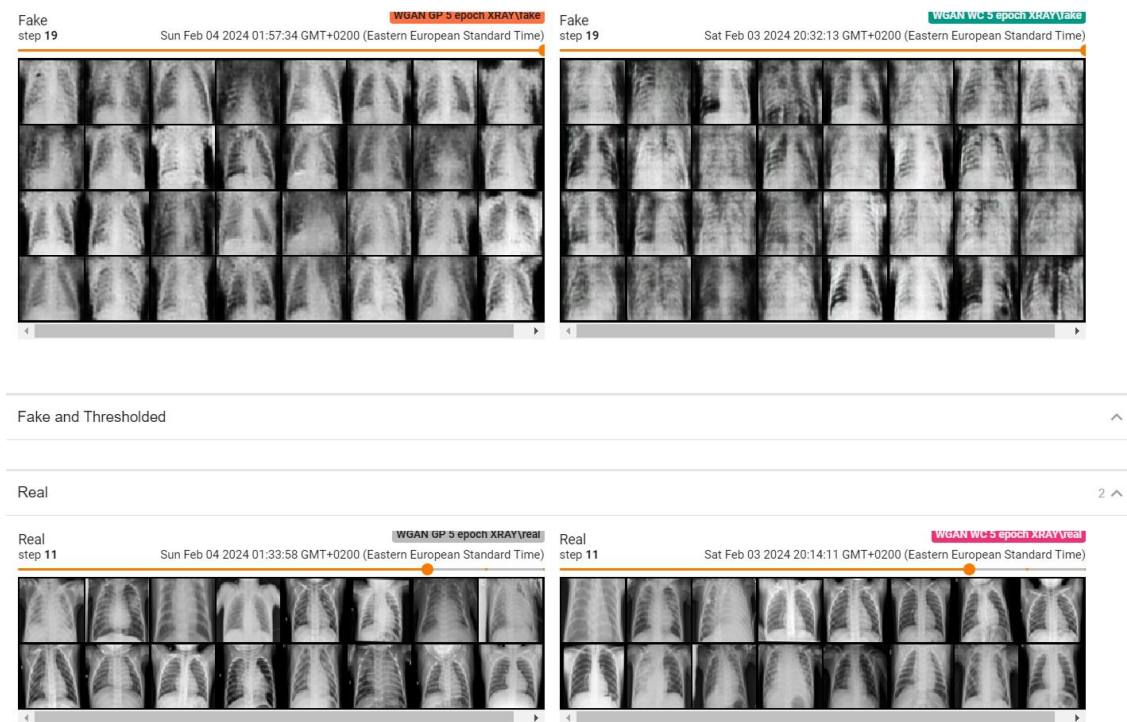
Real

2



### Εικόνα 45 WGAN-GP εναντίον WGAN Weight Clipping: Step 32

Στην Εικόνα 45 που η εκπαίδευση έχει φτάσει στο Step 11 παρατηρείται ότι και η WGAN-GP βελτιώνεται όλο και περισσότερο, ενώ είναι εμφανείς οι ακτινογραφίες και στη WGAN-Weight Clipping.



Εικόνα 46 WGAN-GP εναντίον WGAN Weight Clipping: Τέλος Εκπαίδευσης

Τέλος, στην Εικόνα 46 που η εκπαίδευση έχει σχεδόν ολοκληρωθεί, το αποτέλεσμα είναι σχεδόν παρόμοιο. Ο χρόνος που έκανε να ολοκληρωθεί η εκπαίδευση για το WGAN-GP ήταν 54 λεπτά και 2 δευτερόλεπτα, ενώ ο χρόνος ολοκλήρωσης της WGAN-Weight Clipping ήταν 41 λεπτά και 50 δευτερόλεπτα. Να σημειωθεί ότι τα δυο πειράματα έτρεξαν σε υπολογιστή με επεξεργαστή Intel Core i7 16 GB RAM και GPU Nvidia GeForce MX 2 GB.

Επίσης, σε περαιτέρω δοκιμές που έγιναν κατά το στάδιο διερεύνησης των δύο μεθόδων σε πείραμα που έγινε χρησιμοποιώντας το celeb dataset το οποίο περιλαμβάνει μεγαλύτερο όγκο εικόνων οι χρόνοι εκπαίδευσης ήταν 6 ώρες 22 λεπτά και 14 δευτερόλεπτα για το WGAN-GP. Για το WGAN-Weight Clipping ήταν 3 ώρες 50 λεπτά και 11 δευτερόλεπτα χρησιμοποιώντας το ίδιο hardware. Συνεπώς, οδηγούμαστε στο συμπέρασμα ότι η WGAN-GP μπορεί να είναι πιο σταθερή κατά το στάδιο της εκπαίδευσης, αλλά σε μεγαλύτερο όγκο εικόνων και μεγαλύτερο αριθμό επαναλήψεων μπορεί να είναι αρκετά πιο αργή σε σχέση με την WGAN-Weight Clipping.



### **3.2.6cGAN**

Στη συνέχεια, επιλέχθηκε η μέθοδος WGAN-GP ως η τελική αρχιτεκτονική, καθώς είχε τις περισσότερες αναφορές στη βιβλιογραφία, οι οποίες εστίασαν στο γεγονός ότι η WGAN-GP μειώνει το ρίσκο για mode collapse. Μόλις επιλέχθηκε η τελική αρχιτεκτονική, ξεκίνησε υλοποίηση του Conditional GAN. Για τη δημιουργία της συνθήκης χρειάστηκε να τροποποιηθεί η κλάση του μοντέλου με στόχο να ενταχθεί σε αυτό η ετικέτα της κλάσης. Η μέθοδος που επιλέχθηκε για να εισαχθεί η ετικέτα μέσα στα δίκτυα του Discriminator και του Generator είναι η προσθήκη ενός επιπέδου ενσωμάτωσης.

#### *Αρχιτεκτονική Discriminator*

Η αρχιτεκτονική του Discriminator στο cGAN δε διαφέρει και πολύ από το WGAN-GP που υλοποιήθηκε στην προηγούμενη ενότητα. Αναλυτικότερα, αποτελείται από 5 συνελκτικά επίπεδα με συνάρτηση ενεργοποίησης Leaky ReLU. Η εικόνα εισόδου περνάει μέσα από αυτά τα επίπεδα, μειώνοντας τη δειγματοληψία της μέχρι να φτάσει σε διάσταση 1x1. Η παραγόμενη έξοδος χρησιμοποιείται στη συνέχεια για τον προσδιορισμό της πιθανότητας η εικόνα εισόδου να είναι πραγματική. Αυτό που διαφοροποιεί τη συγκεκριμένη αρχιτεκτονική από την προηγούμενη είναι το επίπεδο ενσωμάτωσης.

Το στοιχείο self.embed είναι ένα αντικείμενο της κλάσης nn.Embedding της PyTorch. Χρησιμοποιείται για την ενσωμάτωση των ετικετών (num\_classes) σε μια συνεχή αναπαράσταση. Το μέγεθος της ενσωμάτωσης ορίζεται ως  $img\_size * img\_size$ , όπου το  $img\_size$  αναφέρεται στις διαστάσεις της εικόνας. Αυτή η ενσωμάτωση εν συνεχεία συνδέεται με την εικόνα εισόδου (x) κατά μήκος της διάστασης καναλιών, δημιουργώντας έναν τανιστή με σχήμα  $N \times (channels\_img + 1) \times img\_size \times img\_size$ , όπου το N αντιπροσωπεύει το μέγεθος της παρτίδας. Ο σκοπός αυτής της ενσωμάτωσης είναι να παρέχει πληροφορίες υπό συνθήκη στον Discriminator, επιτρέποντάς του να μάθει να διακρίνει με βάση τις παρεχόμενες πληροφορίες ετικέτας.

#### *Αρχιτεκτονική Generator*

Η αρχιτεκτονική του Generator είναι η αντίστροφη του Discriminator. Ξεκινά με ένα πλήρως συνδεδεμένο επίπεδο για την αναβάθμιση (upsampling) του διανύσματος θορύβου, ακολουθούμενο από διάφορα ανεστραμμένα επίπεδα συνέλιξης (transposed Conv2d) για τη σταδιακή αύξηση των διαστάσεων της

παραγόμενης εικόνας. Η έξοδος της γεννήτριας είναι μια παραγόμενη εικόνα με τις ίδιες διαστάσεις με τις πραγματικές εικόνες.

Παρόμοια με τον Discriminator, το στοιχείο `self.embed` είναι ένα επίπεδο `nn.Embedding`. Ενσωματώνει τις ετικέτες (`num_classes`) σε μια συνεχή αναπαράσταση μεγέθους `embed_size`. Η ενσωμάτωση στη συνέχεια αποσυμπιέζεται και αποσυμπιέζεται ξανά για να ταιριάζει με τις διαστάσεις της παραγόμενης εικόνας. Τέλος, το διάνυσμα θορύβου ( $x$ ) και η ενσωμάτωση των ετικετών συνενώνονται κατά μήκος της διάστασης του καναλιού και περνούν από το δίκτυο Generator.

Συνοψίζοντας όλα τα παραπάνω, παρατηρείται ότι με την προσθήκη αυτή ο Generator έχει την πληροφορία για το ποια ετικέτα πρέπει να δημιουργήσει, ενώ ο Discriminator έχει την πληροφορία για το ποια εικόνα πρέπει να είναι. Επομένως, ο Generator για να ξεγελάσει τον discriminator δεν αρκεί μόνο να παράξει ένα ρεαλιστικό δείγμα, αλλά πρέπει το δείγμα να αντιστοιχεί και στη συγκεκριμένη κλάση.

Στο δίκτυο του Discriminator προστέθηκε άλλη μια παράμετρος σαν είσοδος, η `num_classes`, η οποία συμβολίζει τον αριθμό των κλάσεων βάσει των οποίων θα κληθεί να δημιουργήσει δείγματα. Στη συνέχεια, τροποποιήθηκε η συνάρτηση `forward`, η οποία εκτός από την εικόνα εισόδου ( $x$ ) τώρα θα παίρνει και την ετικέτα (`labels`). Τέλος, προστέθηκε το επίπεδο ενσωμάτωσης (`self.embed`) χρησιμοποιώντας την `nn.Embedding`, το οποίο αντιστοιχίζει τις ετικέτες εισόδου σε μια συνεχή διανυσματική αναπαράσταση. Ακολουθώντας οι ενσωματώσεις συνενώνονται με τις εικόνες εισόδου ( $x$ ).

Στον παρακάτω πίνακα (Πίνακας 8) καταγράφονται οι υπερπαραμέτροι που χρησιμοποιήθηκαν για την εκπαίδευση, αλλά και την τροποποίηση του δικτύου (cGAN) στην πρώτη δοκιμή, ώστε να παραχθούν συνθετικές εικόνες χρησιμοποιώντας μια συνθήκη. Οι περισσότερες παραμέτροι παραμένουν ίδιες με το παράδειγμα WGAN-GP που είδαμε στην προηγούμενη ενότητα. Αυτό που αλλάζει είναι ότι έχει προστεθεί μια υπερπάρμετρος, η `Number of Classes`, η οποία υποδηλώνει τον αριθμό των κλάσεων και στην περίπτωση μας θα είναι ο αριθμός 2 καθώς έχουμε την κλάση `Normal` και την κλάση `Pneumonia`.

Πίνακας 8 Υπερπαραμέτροι cGAN

<b>Υπερπαραμέτρος</b>	<b>Τιμή</b>
Device	cuda
Learning Rate	0.0001
Batch Size	16
Image Size	64
Number of Classes	2
GEN_EMBEDDING	100
Number of Channels	3
Z Dimension	100
Number of Epochs	100
Discriminator Features	64
Generator Features	64
Critic Iterations	5
LAMBDA GP	10

Αφού τροποποιήθηκε το μοντέλο έπρεπε να αλλάξει και ο αλγόριθμος εκπαίδευσης προσθέτοντας και την ετικέτα στον βρόγχο εκπαίδευσης για να μπορούν τα δύο δίκτυα Discriminator-Generator να υλοποιήσουν το έργο τους. Στην Εικόνα 47 φαίνεται ο τρόπος που τροποποιήθηκε ο βρόγχος εκπαίδευσης του WGAN-GP. Αναλυτικότερα, προστέθηκε και το label της εικόνας (Πράσινη υπογράμμιση) για να μετατραπεί το WGAN-GP σε Conditional. Στο σημείο αυτό αξίζει να γίνει υπενθύμιση ότι εφόσον χρησιμοποιείται η WGAN-GP αρχιτεκτονική, ο Discriminator ονομάζεται Critic.

```

for epoch in range(NUM_EPOCHS):

    for batch_idx, (real, labels) in enumerate(loader):

        real=real.to(device)
        cur_batch_size=real.shape[0]
        labels=labels.to(device)

        for _ in range(CRITIC_ITERATIONS):
            noise = torch.randn((BATCH_SIZE, NOISE_DIM, 1, 1)).to(device)
            fake = gen(noise,labels)
            critic_real = critic(real, labels)
            critic_real = critic_real.reshape(-1)

            critic_fake = critic(fake, labels).reshape(-1)
            gp = gradient_penalty(critic,labels, real, fake, device=device)
            loss_critic = (
                -(torch.mean(critic_real) - torch.mean(critic_fake))+ LAMBDA_GP*gp
            )
            critic.zero_grad()
            loss_critic.backward(retain_graph=True)
            opt_critic.step()

        ###Train generatpr :min = -E[critic(gen_fake)]
        output= critic(fake,labels).reshape(-1)
        loss_gen= -torch.mean(output)
        #update steps
        gen.zero_grad()
        loss_gen.backward()
        opt_gen.step()

```

Εικόνα 47 cGAN βρόγχος εκπαίδευσης

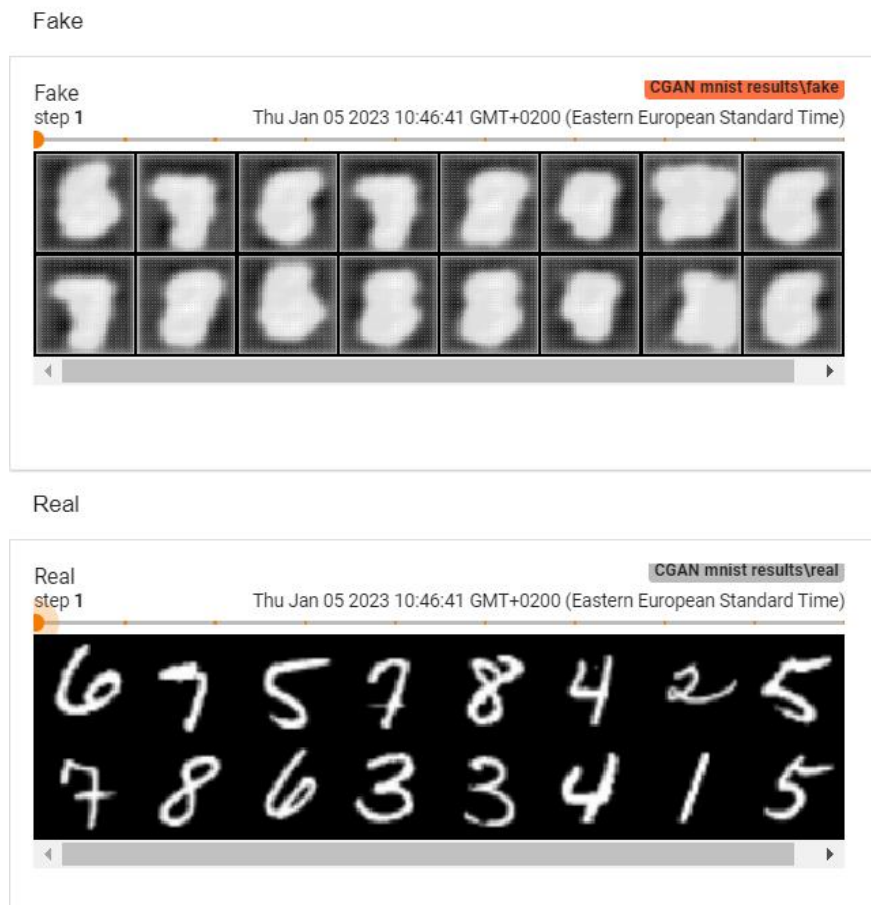
### **Δοκιμή cGAN σε MNIST dataset:**

Για να ελεγχθεί η λειτουργικότητα του cGAN και να επαληθευτεί το γεγονός ότι πλέον λειτουργεί και παράγει δείγματα με βάση την κλάση της εικόνας που του έχει δοθεί, χρησιμοποιήθηκε αρχικά το MNIST dataset. Στον παρακάτω πίνακα (Πίνακας 9), καταγράφονται οι υπερπαραμέτροι που χρησιμοποιήθηκαν για την πρώτη δοκιμή. Αυτό που αξίζει να αναφερθεί είναι ότι ορίστηκαν 10 διαφορετικές κλάσεις που αντιστοιχούν στα χειρόγραφα νούμερα που αναπαρίστανται στις εικόνες (από 0-9) και το Number of channels το οποίο είναι 1 επειδή πρόκειται για ασπρόμαυρη εικόνα. Τέλος, ως αριθμό εποχών ορίστηκε το 3. Οι υπόλοιπες υπερπαραμέτροι δε διαφέρουν από τα προηγούμενα παραδείγματα που παρουσιάστηκαν στην προηγούμενη ενότητα για τη συγκεκριμένη δοκιμή.

Πίνακας 9 Υπερπαραμέτροι του cGAN στο MNist

Υπερπαραμέτρος	Τιμή
Device	cuda
Learning Rate	0.0001
Batch Size	16
Image Size	64
Number of Classes	10
GEN_EMBEDDING	100
Number of Channels	1
Z Dimension	100
Number of Epochs	3
Discriminator Features	64
Generator Features	64
Critic Iterations	5
LAMBDA GP	10

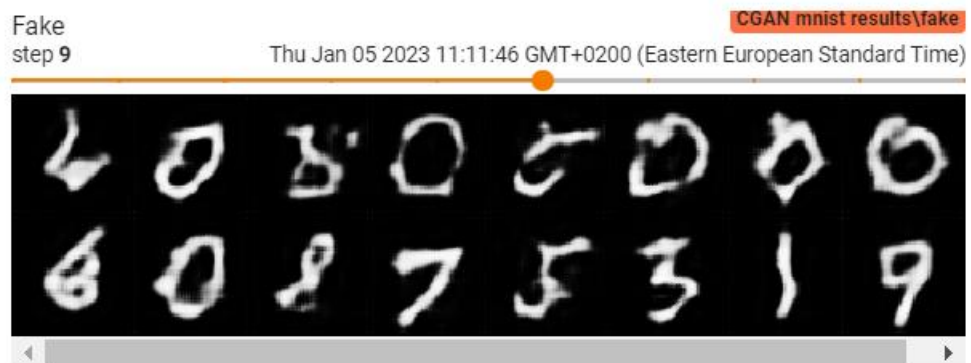
Στην Εικόνα 48 φαίνεται πως ξεκινάει η εκπαίδευση υλοσυνθήκη. Σε αυτό το σημείο, ακόμα τα παραγόμενα δείγματα (Fake) δεν έχουν πάρει κάποια μορφή, ωστόσο φαίνεται ότι ορισμένα χαρακτηριστικά σχηματίζονται σταδιακά και δεν φαίνεται αποκλειστικά και μόνο ο θόρυβος.



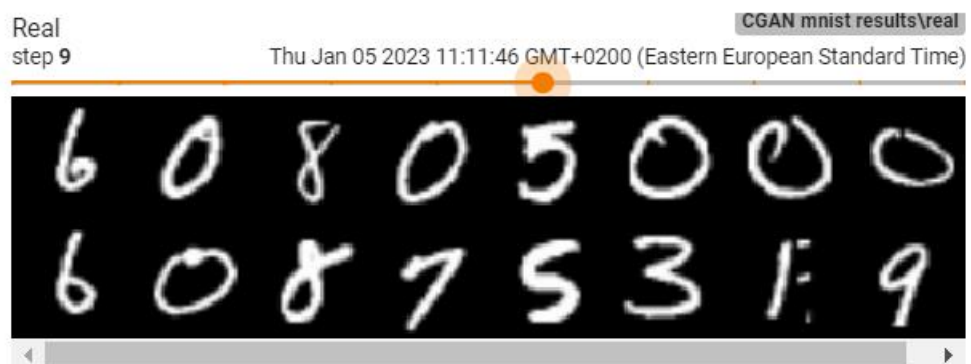
Εικόνα 48 cGAN στο Mnist αρχή εκπαίδευσης

Στην Εικόνα 49 η εκπαίδευση του cGAN έχει φτάσει περίπου στη μέση και ήδη έχει βελτιώσει αισθητά την ικανότητα παραγωγής δειγμάτων με βάση την κλάση που έχει δοθεί. Έχει ήδη καταφέρει να σχηματίσει κάποια νούμερα δεδομένης της κλάσης. Παράδειγμα στο πεδίο (Real) φαίνεται ότι όταν του δίνεται το χειρόγραφο ψηφίο της κλάσης 0 έχει μάθει να το συνθέτει αρκετά καλά. Το ίδιο συμβαίνει και με το ψηφίο 9 και 3. Από την άλλη, φαίνεται ότι με το ψηφίο 6 θέλει περισσότερο χρόνο και δεν είναι τόσο αποτελεσματική.

Fake



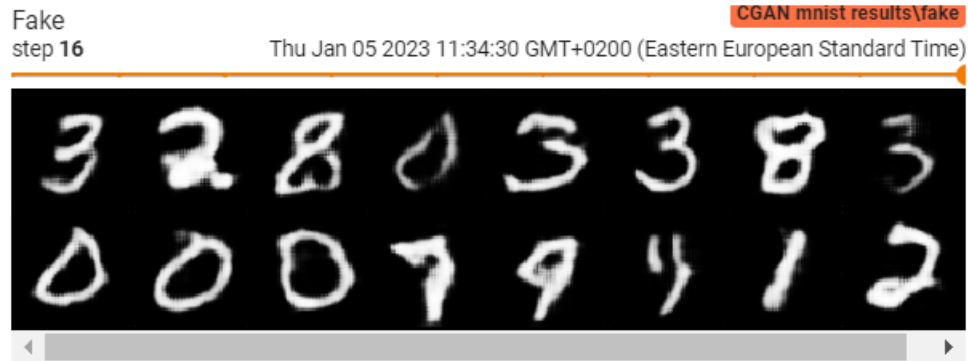
Real



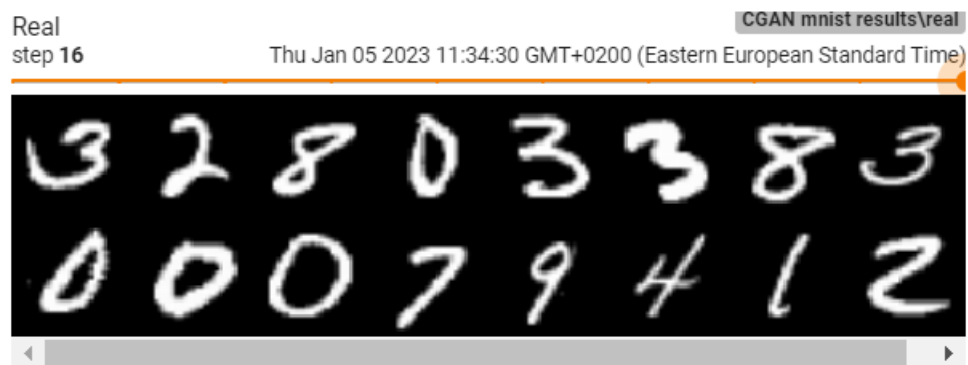
Εικόνα 49 cGAN στο mnist στη μέση εκπαίδευση

Στην Εικόνα 50, το cGAN έχει ολοκληρώσει την εκπαίδευσή του (3 εποχές) και έχει καταφέρει να αποτυπώσει σωστά σχεδόν όλα τα νούμερα που του δοθήκαν (Real), στοιχείο που σημαίνει ότι το cGAN έχει πέτυχει τον σκοπό του.

Fake



Real

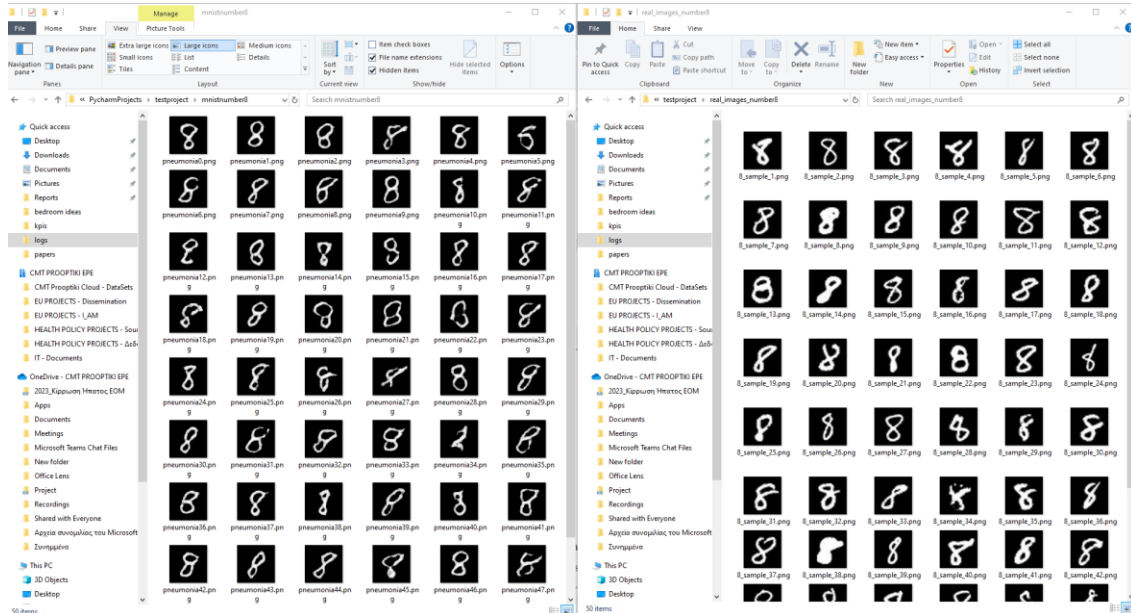


Εικόνα 50 cGAN στο MNist στο τέλος της εκπαίδευσης

Ο χρόνος που χρειάστηκε για να ολοκληρωθεί η εκπαίδευση ήταν 47 λεπτά και 49 δευτερόλεπτα. Μετά το τέλος της εκπαίδευσης, αποθηκεύτηκε το μοντέλο του generator και δημιουργώντας ένα νέο Python script ολοκληρώθηκε η φόρτωσή του με τη βοήθεια του PyTorch, ζητώντας του να δημιουργήσει 50 νέα συνθετικά δείγματα από την κλάση του ψηφίου 8.

Στην Εικόνα 51 φαίνονται οι νέες συνθετικές εικόνες που παρήγαγε το δίκτυο δίνοντάς του σαν όρισμα το αντίστοιχο label. Οι εικόνες που παρήγαγε το δίκτυο αξιολογήθηκαν με βάση τα ποσοτικά μέτρα Average SSIM και Average MSE. Αναλυτικότερα, οι τιμές που σημειώθηκαν ήταν οι ακόλουθες: Average SSIM τιμή 0.5339, τιμή που αντιστοιχεί σε μέτριο επίπεδο ομοιότητας μεταξύ των παραγόμενων δειγμάτων και των πραγματικών. Ως προς το Average MSE καταγράφηκε η τιμή 0.1037, η οποία υποδηλώνει χαμηλή διαφορά μεταξύ των

τιμών των εικονοστοιχείων. Ωστόσο, λαμβάνοντας υπόψη και την οπτική επιθεώρηση καταλήγουμε στο γεγονός ότι το cGAN έχει παράξει έναν ικανοποιητικό δείγμα εικόνων που δεν διαφέρουν σχεδόν καθόλου από τις πραγματικές.



Εικόνα 51 Εξαγόμενα συνθετικά δείγματα από το εκπαιδευμένο cGAN: (αριστερό μέρος cGAN δείγματα, δεξιό μέρος αληθινά δείγματα)

### Δοκιμή cGAN σε Chest Xray dataset:

Η επομένη δοκιμή έγινε στο dataset με τις ακτινογραφίες, το οποίο είναι και το κυρίως Dataset της συγκεκριμένης μελέτης. Όσον αφορά τις υπερπαραμέτρους το μονό πράγμα που άλλαξε σε σχέση με την προηγούμενη δοκιμή ήταν το Number of Classes, το οποίο σε αυτή την περίπτωση είναι 2 και το Number of Channels από 1 σε 3, καθώς πρόκειται για εικόνα RGB. Έχοντας σα βάση τον παρακάτω πίνακα υπερπαραμέτρων, έτρεξε το cGAN αλλάζοντας σταδιακά τον αριθμό των εποχών.

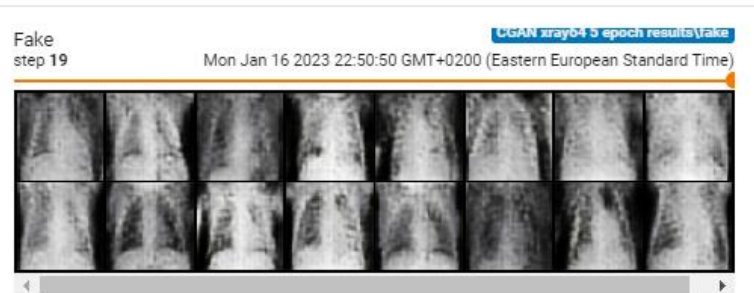


Πίνακας 10 Υπερπαράμετροι του cGAN στο ChestXray Dataset

Υπερπαράμετρος	Τιμή
Device	cuda
Learning Rate	0.0001
Batch Size	16
Image Size	64
Number of Classes	2
GEN_EMBEDDING	100
Number of Channels	3
Z Dimension	100
Number of Epochs	5
Discriminator Features	64
Generator Features	64
Critic Iterations	5
LAMBDA GP	10

Η πρώτη δοκιμή έγινε για 5 εποχές. Στην Εικόνα 52 παρατηρείται ότι οι εικόνες που έχει συνθέσει το cGAN έχουν αρχίσει και μοιάζουν με ακτινογραφίες, αλλά η ποιότητά τους είναι αρκετά χαμηλή στοιχείο που αναδεικνύει την αναγκαιότητα περαιτέρω εκπαίδευσης του δικτύου για επιπρόσθετες εποχές.

Fake



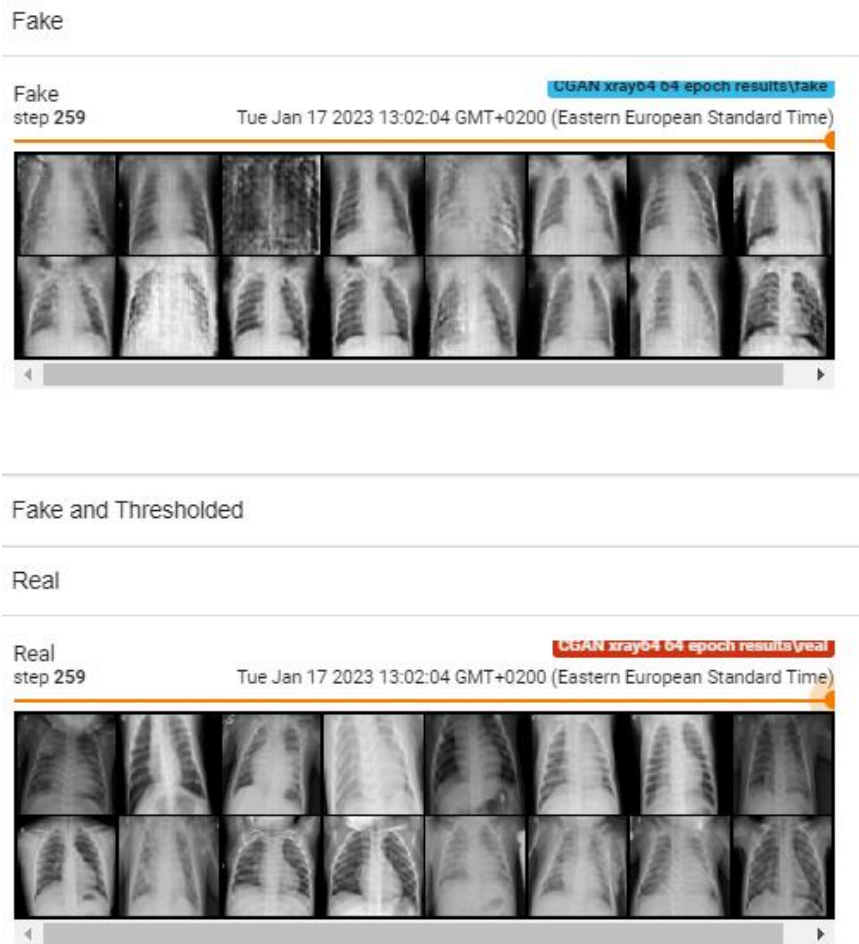
Fake and Thresholded

Real



Εικόνα 52 cGAN αποτελέσματα μετά από 5 εποχές

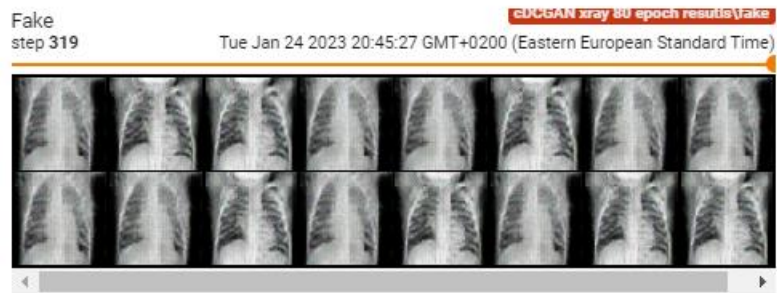
Η δεύτερη δοκιμή έγινε για 64 εποχές. Στην Εικόνα 53 παρατηρείται ότι το δίκτυο δίνοντας τις περισσότερες εποχές έχει πέτυχει βελτιωμένα αποτελέσματα σε σχέση με την πρώτη δοκιμή. Επίσης, φαίνεται ότι όταν του δίνεται μια εικόνα με πνευμονία (δηλαδή με πιο έντονη τη λευκή σκίαση) έχει φτιάξει μία εικόνα που η λευκή σκίαση είναι εξίσου έντονη. Από αυτό προκύπτει ότι το condition που έχει δοθεί στο cGAN λειτουργεί.



Εικόνα 53 cGAN αποτελέσματα μετά από 64 εποχές

Στη συνέχεια, επιχειρήθηκε η εκπαίδευση του δικτύου για 80 εποχές, αλλά χωρίς της απόσταση Wasserstain και χωρίς Gradient Penalty. Όπως φαίνεται και στην παρακάτω εικόνα (Εικόνα 54), η δοκιμή αυτή απέτυχε, καθώς το μοντέλο οδηγήθηκε σε Mode Collapse.

Fake



Fake and Thresholded

Real



Εικόνα 54 cGAN αποτελέσματα χωρίς gradient penalty και χωρίς Wasserstein distance μετά από 80 εποχές (Mode Collapse)

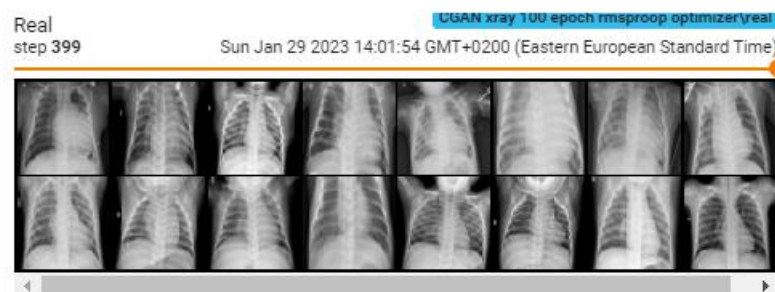
Επομένη δοκιμή ήταν η αντικατάσταση του βελτιστοποιητή Adam με τον RMSprop. Επίσης, για τη χρήση του RMSprop άλλαξε και το LEARNING\_RATE σε 0.00005. Μετά την παραμετροποίηση του δικτύου, ακολούθησε εκπαίδευση για 100 εποχές. Στην Εικόνα 55, φαίνεται το αποτέλεσμα του cGAN. Αυτό που παρατηρείται σε σχέση με τις προηγούμενες δοκιμές είναι ότι έχει καταφέρει να συνθέσει ακόμα πιο ρεαλιστικά δείγματα με περισσότερη λεπτομέρεια (π.χ. τα κόκαλα στον θώρακα είναι πιο ευδιάκριτα).

Fake



Fake and Thresholded

Real



Εικόνα 55 cGAN αποτελέσματα μετά από 100 εποχές και χρήση RMSprop βελτιστοποιητή.

Στην προσπάθεια παραγωγής όσο το δυνατόν καλύτερων εικόνων μέσω του cGAN, τροποποιήθηκε το δίκτυο (cGAN) έτσι ώστε να λαμβάνει μια εικόνα μεγαλύτερων διαστάσεων 128x128 και να παράγει σαν έξοδο μια εικόνα 128x128. Εδώ πρέπει να σημειωθεί ότι οι πόροι του υπολογιστή ήταν περιορισμένοι και δεν ήταν εφικτή η υλοποίηση της δοκιμής για μεγάλο αριθμό εποχών. Αυτό που έγινε ήταν να μεταφερθεί ο κώδικας με όλα τα απαραίτητα αρχεία στο Google Colab χρησιμοποιώντας την Free έκδοσή του, καταφέρνοντας έτσι την ολοκλήρωση της δοκιμής για 5 εποχές. Στην Εικόνα 56, φαίνονται οι εικόνες που έχει καταφέρει να συνθέσει το cGAN. Το αποτέλεσμα, παρά τον μικρό αριθμό εποχών, είναι αρκετά ρεαλιστικό και οι εικόνες είναι κοντά στα δείγματα που έχουν ζητηθεί (Real) με βάση την κλάση.



Εικόνα 56 cGAN αποτελέσματα μετά από 5 εποχές με μέγεθος εικόνα 128x128(Google Colab)

Μετά από αρκετά πειράματα και δοκιμές αλλά και αλλαγές στις υπερπαραμέτρους, εξαγάγαμε δείγματα από κάθε cGAN που δημιουργήσαμε, χρησιμοποιώντας τις μετρικές αξιολόγησης GANs FID και KID, καθώς και τις μετρικές SSIM και MSE, και συγκρίναμε τις παραγόμενες συνθετικές εικόνες για την ενδιαφερόμενη κλάση (Normal) από κάθε cGAN με τις εικόνες του αρχικού dataset για τη συγκεκριμένη κλάση. Τα δίκτυα cGAN που μας έφεραν τις καλύτερες τιμές σε FID, KID, SSIM και MSE είναι αυτά που φαίνονται στον παρακάτω πίνακα (Πίνακας 11) και είναι και αυτά τα οποία χρησιμοποιήσαμε για τα περάματά μας με σκοπό τη βελτιστοποίηση της απόδοσης του ταξινομητή ως προς τη μειωτική κλάση (Normal).

Πίνακας 11 Μετρικές Αξιολόγησης cGAN

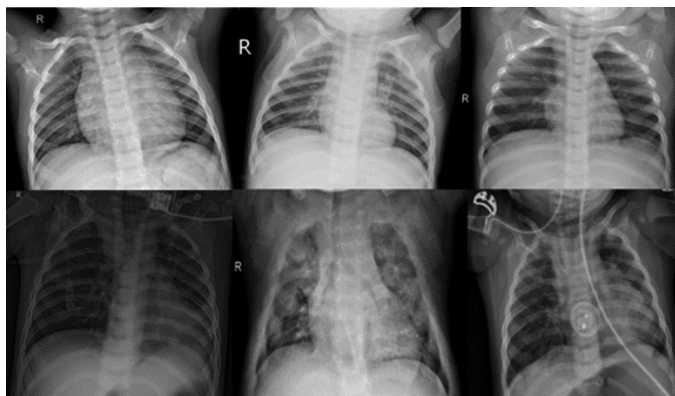
	<b>FID</b>	<b>KID</b>	<b>SSIM</b>	<b>MSE</b>
<b>cGAN(adam)</b>	194.5449	0.2826	0.5111	0.0386
<b>cGAN(RMSprop)</b>	132.1694	0.2037	0.5014	0.0371
<b>cGAN(128x128)</b>	258.6015	0.3619	0.5661	0.0346

## 4 ΚΕΦΑΛΑΙΟ 4: ΠΕΙΡΑΜΑΤΑ-ΑΠΟΤΕΛΕΣΜΑΤΑ

### 4.1 Το Dataset

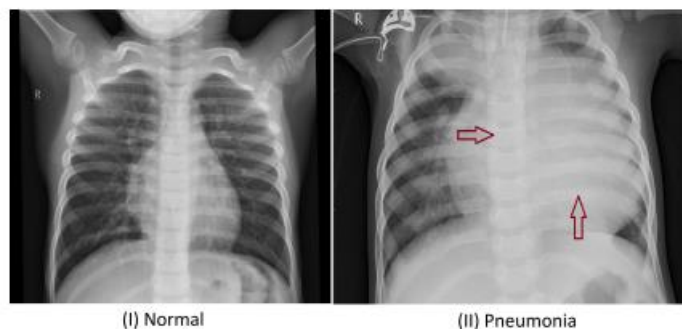
Τα δεδομένα που χρησιμοποιήθηκαν για την παρούσα μελέτη ανακτήθηκαν από την ιστοσελίδα Kaggle και αποτελούνται από 5.856 ακτινογραφίες θώρακος (πρόσθια-οπίσθια) παιδιατρικών ασθενών που συλλέχθηκαν και ενσωματώθηκαν στο κείμενο των Kermany et al (2018) [91], εκ των οποίων 4.273 επισημάνθηκαν ως θετικές περιπτώσεις (με πνευμονία) και 1.583 ως αρνητικές περιπτώσεις (φυσιολογικές).

Το σύνολο δεδομένων προήλθε από συνολικά 5.856 παιδιατρικούς ασθενείς του Ιατρικού Κέντρου Γυναικών και Παιδιών στην πόλη Guangzhou, στους οποίους η απεικόνιση με ακτίνες X θώρακος πραγματοποιήθηκε στο πλαίσιο της συνήθους κλινικής φροντίδας του ασθενούς. Μετά από προσεκτική αξιολόγηση από εμπειρογνώμονες του ιατρικού κέντρου στον τομέα της ακτινογραφίας θώρακος, κάθε εικόνα υποβλήθηκε σε ποιοτικό έλεγχο. Έτσι, εικόνες ακτινογραφίας με χαμηλή ποιότητα ή μη αναγνώσιμες σαρώσεις αφαιρέθηκαν από το αρχικό σύνολο δεδομένων. Η πολυπλοκότητα της ταξινόμησης μπορεί να αποδοθεί στα διαφορετικά μεγέθη, τον προσανατολισμό και τις εντάσεις των γκριζων εικονοστοιχείων των ακτινογραφιών X, όπως φαίνεται στην Εικόνα 57.



Εικόνα 57 Παράδειγμα εικόνων με εντάσεις γκριζων εικονοστοιχείων

Μπορεί να παρατηρηθεί ότι στις εικόνες των περιπτώσεων πνευμονίας, οι κυψελίδες γεμίζουν με έκκριση (φλεγμονώδες υγρό) που εμφανίζεται ως λευκή κηλίδα στην ακτινογραφία θώρακος (Εικόνα 58).



Εικόνα 58 Παράδειγμα ακτινογραφίας χωρίς πνευμονία (αριστερά) και ακτινογραφίας με πνευμονία (δεξιά)

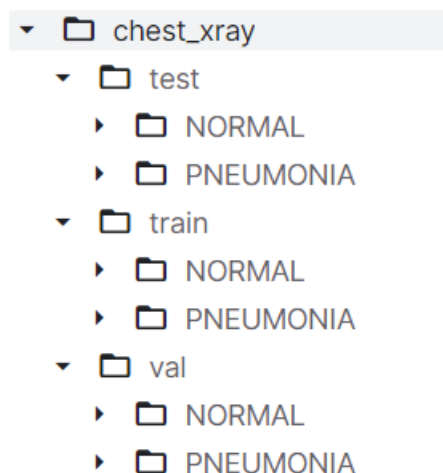
Η υπόλευκη περιοχή αντιστοιχεί στην πνευμονική θολερότητα, η οποία χαρακτηρίζει μια πνευμονική ενοποίηση [92].

#### 4.1.1 Ανάλυση & Προεπεξεργασία

Το dataset που χρησιμοποιήθηκε είχε μέγεθος 1.24 GB και ήταν χωρισμένο σε τρεις κεντρικούς φακέλους test, train και val (Εικόνα 59). Ο κάθε φάκελος από αυτούς είχε μέσα δύο υποφακέλους που αφορούσαν την κλάση στην οποία ανήκαν οι εικόνες. Οι φάκελοι με όνομα NORMAL περιείχαν εικόνες ακτινογραφιών ανθρώπων που δεν έχουν πνευμονία, ενώ οι φάκελοι με όνομα PNEUMONIA περιείχαν εικόνες ακτινογραφιών ασθενών με πνευμονία.

#### Data Explorer

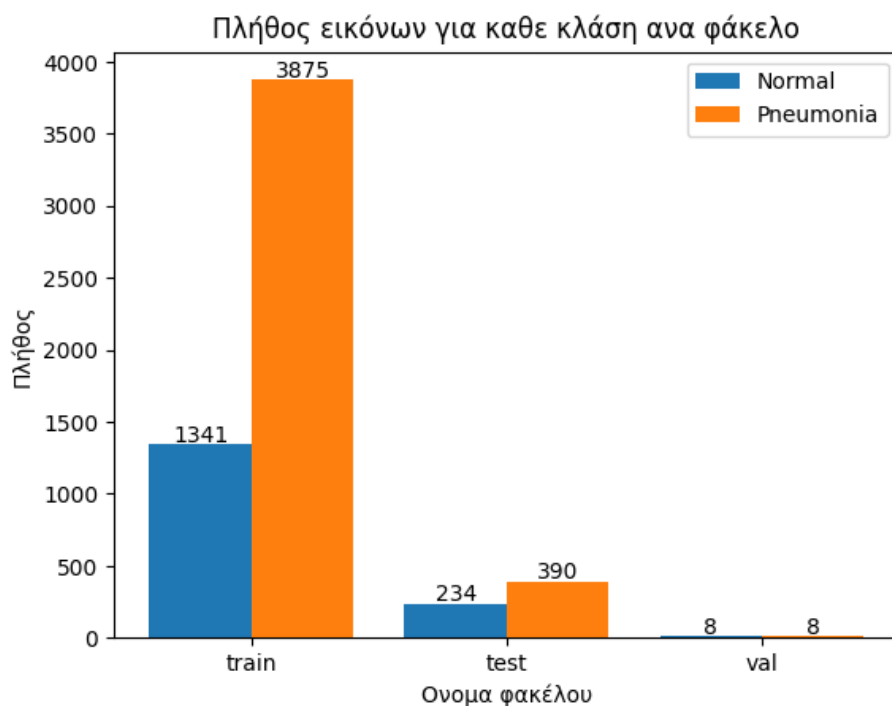
Version 2 (1.24 GB)



Εικόνα 59 Δομή φακέλων dataset

Αξίζει να σημειωθεί ότι οι εικόνες ανθρώπων χωρίς πνευμονία αναγνωρίζονται στο συγκεκριμένο dataset από τις λέξεις κλειδιά που βρίσκονται στο όνομα της

κάθε εικόνας, ήτοι “IM” και “NORMAL”. Οι εικόνες που αφορούν τη συγκεκριμένη νόσο, στο όνομα φέρουν τις λέξεις κλειδιά “virus” η «bacteria». Στην Εικόνα 60, φαίνεται αναλυτικά το πλήθος των εικόνων για κάθε κλάση ανά φάκελο. Σε αυτό το σημείο, επιλέχθηκε ο φάκελος val που περιέχει πολύ λίγες εικόνες (8) για κάθε κλάση σύνολο (14) να μην συμπεριληφθεί στη διαδικασία της εκπαίδευσης και να χρησιμοποιηθεί στο τέλος αποκλειστικά για testing.



Εικόνα 60 Συνολικό πλήθος εικόνων για κάθε κλάση ανά φάκελο

Στο παρακάτω πίνακα (Πίνακας 12) φαίνεται αθροιστικά για κάθε κλάση ο αριθμός των εικόνων που υπάρχει κάθε φάκελο και ο συνολικός αριθμός όλων των εικόνων του dataset.

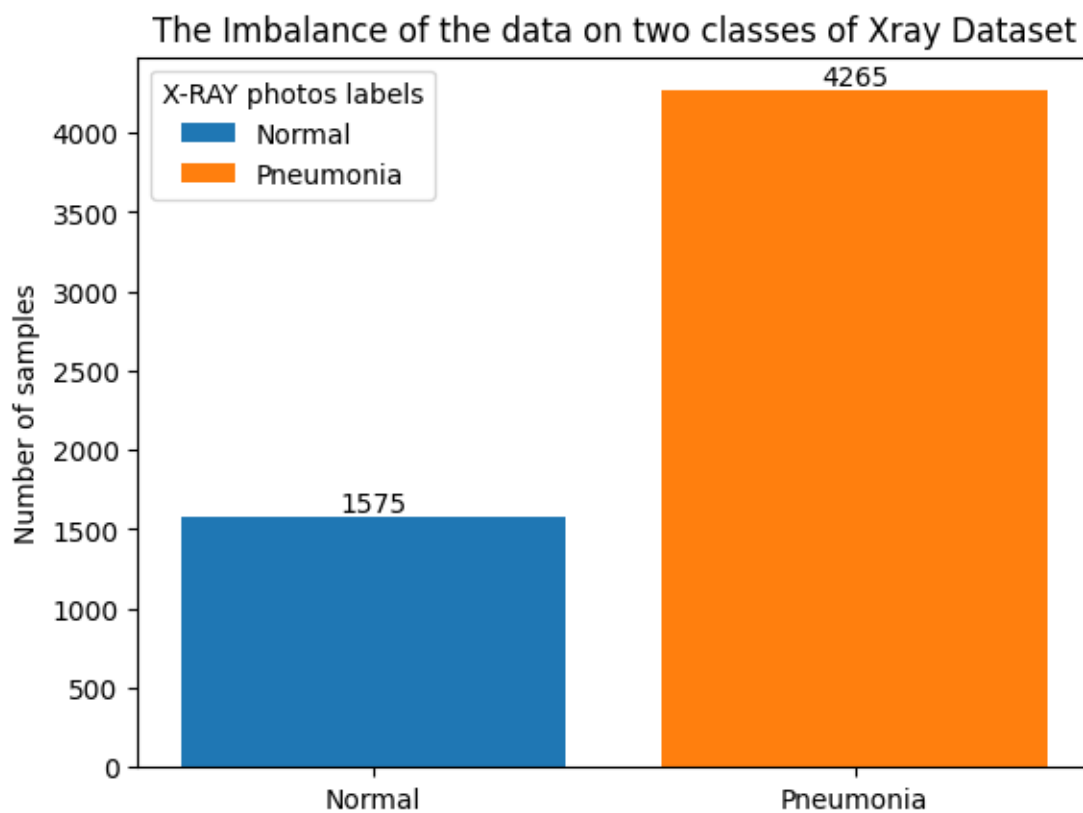
Πίνακας 12 Κατανομή δεδομένων

	<b>Normal</b>	<b>Pneumonia</b>
Train	1341	3875
Test	234	390
Val	8	8
Σύνολο	1583	4273
<b>Σύνολο Δεδομένων</b>	<b>5856</b>	

Μετά την απομόνωση του φακέλου val έγινε ομαδοποίηση των δεδομένων για να μελετηθεί η αναλογία των δεδομένων μεταξύ των δύο κλάσεων. Αυτό που παρατηρείται από την Εικόνα 61 είναι ότι το dataset παρουσιάζει ανισορροπία



ως προς τις κλάσεις που αποτελείται. Συγκεκριμένα, οι εικόνες με πνευμονία είναι αρκετά περισσότερες (4265) από τις εικόνες που είναι υγιείς (1575). Αυτό είναι ένα ζήτημα που μπορεί να επηρεάσει σημαντικά την απόδοση του ταξινομητή, όπως αναφέρθηκε και σε προηγούμενο Κεφάλαιο. Στη συγκεκριμένη περίπτωση, το μοντέλο αν εκπαιδευτεί με αυτά τα δεδομένα θα έχει μια τάση στο να προβλέπει πιο σωστά την κλάση που φέρει την ασθένεια της πνευμονίας και να αστοχεί στην περίπτωση της κλάσης που είναι υγιείς. Συνεπώς, εικόνες από υγιή πληθυσμό μπορεί να προβλέπεται εσφαλμένα ότι έχουν πνευμονία.



Εικόνα 61 Το Dataset των ακτινογραφιών θώρακος παρουσιάζει το πρόβλημα της ανισορροπίας των κλάσεων

Με τη διαπίστωση του προβλήματος της ανισορροπίας, συνεχίστηκε η επεξεργασία του dataset, κατασκευάζοντας τα απαραίτητα train και test sets. Με τη βοήθεια της γλώσσας προγραμματισμού Python και χρησιμοποιώντας πλέον μόνο τους φακέλους train και test δημιουργήθηκε το σύνολο εκπαίδευσης (train set) και το σύνολο δοκιμής (test set).

```
train_paths = train_normal+train_pneumonia
test_paths = test_normal+test_pneumonia
```

```
train_labels=[0] * len(train_normal)+ [1] * len(train_pneumonia)
test_labels=[0] * len(test_normal) + [1] * len(test_pneumonia)
```

Εικόνα 62 Δημιουργία train set και test set

Στην Εικόνα 62 στο πρώτο κελί φαίνονται τα paths των εικόνων που θα χρησιμοποιηθούν για την κατασκευή του trainset το οποίο περιλαμβάνει όλες τις εικόνες από τον φάκελο Train->Normal (1341) και όλες τις εικόνες από τον φάκελο Train->Pneumonia (3875) ενωμένα σε μια μεταβλητή με όνομα train\_paths. Η ίδια διαδικασία ακολουθείται και με τις εικόνες που βρίσκονται στον φάκελο Test->Normal (234) και στον φάκελο Test->Pneumonia (390). Στο δεύτερο κελί δημιουργήθηκαν τα labels για κάθε εικόνα στο trainset και στο test set βάζοντας 0 στις εικόνες που δεν έχουν πνευμονία και 1 στις εικόνες που έχουν πνευμονία. Τέλος, έγινε επαλήθευση με print του μεγέθους των μεταβλητών, ώστε να διασφαλιστεί ότι για το κάθε path εικόνας (είτε στο trainset, είτε στο test set) υπάρχει και το αντίστοιχο label. Τα μεγέθη είναι τα ακόλουθα. Εφόσον ο αριθμός των path και ο αριθμός των labels είναι ίσος για κάθε ζευγάρι σημαίνει ότι η διαδικασία έχει ολοκληρωθεί επιτυχώς.

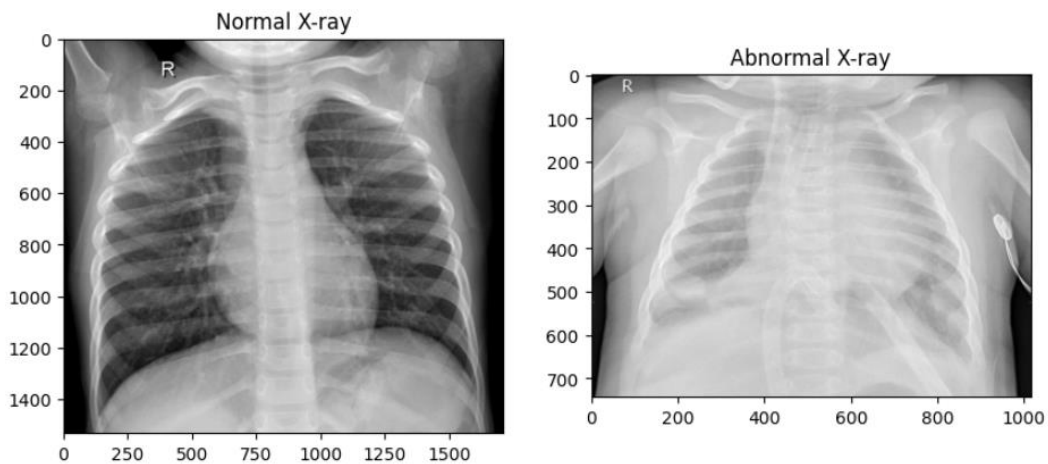
```
train_paths:5216, train_labels:5216 και test_paths:624, test_labels:624
```

## 4.2 Δημιουργία Classifier

Αφού έγινε η προετοιμασία των δυο παραπάνω set δεδομένων και ήρθαν στην επιθυμητή μορφή, ξεκίνησε η διαδικασία δημιουργίας του ταξινομητή. Χρησιμοποιώντας την παρακάτω εντολή (Εικόνα 63) train\_test\_split της βιβλιοθήκης scikit-learn χωρίστηκε το trainset σε 75% δεδομένα εκπαίδευσης και 25% δεδομένα δοκιμής. Επίσης με την επιλογή stratify μέσα στη συνάρτηση αυτή είναι εφικτό να διατηρηθεί η αναλογία μεταξύ των δυο κλάσεων και να μην μοιραστούν για παράδειγμα στα δυο σετ μόνο δεδομένα από τη μια κλάση. Στη συγκεκριμένη περίπτωση, με τον παραπάνω διαχωρισμό, το 75% των train δεδομένων (5216) αντιστοιχεί σε 3912 δεδομένα εκ των οποίων στην κλάση 0 υπάρχουν 1006, ενώ στην κλάση 1 2906.

```
train_paths,valid_paths,train_labels,valid_labels=train_test_split(train_paths,
                                                                    train_labels,
                                                                    stratify=train_labels)
```

Εικόνα 63 Διαχωρισμός του trainset σε δεδομένα εκπαίδευσης και δεδομένα δοκιμής. Εδώ πρέπει να σημειωθεί ότι επειδή για τον ταξινομητή θα χρησιμοποιηθεί CNN, στην εκπαίδευση του μοντέλου λαμβάνουν μέρος και τα δεδομένα δοκιμής. Στη διαδικασία, ακολούθησε η εκτύπωση σε οθόνη ενός τυχαίου δείγματος εικόνων από κάθε κλάση.



Εικόνα 64 Τυχαίες εικόνες από κάθε κλάση αριστερά η κλάση 0 (*Normal*) και δεξιά η κλάση 1 (*Pneumonia*).

Βλέποντας τις παραπάνω εικόνες (Εικόνα 64), διαπιστώνεται ότι οι δυο εικόνες έχουν διαφορετικό μέγεθος. Η μια είναι 1400x1500 και η άλλη 700x1000 οπότε κοιτάζοντας και τις υπόλοιπες εικόνες του φακέλου τα μεγέθη τους ήταν διαφορετικά. Το γεγονός αυτό δυσκόλευε την εκπαίδευση του μοντέλου σε συνδυασμό και με το γεγονός ότι επειδή πρόκειται για ιατρικές εικόνες που έχουν υψηλή ανάλυση, υπάρχει και ανάγκη για υψηλούς υπολογιστικούς πόρους.

Για να αντιμετωπιστούν τέτοιου είδους προβλήματα χρειάστηκε να μετασχηματιστούν τα set δεδομένων με τη βοήθεια του PyTorch<sup>2</sup> Framework. Ακολουθώντας τη μεθοδολογία που προτείνει το documentation του PyTorch για μετασχηματισμό των συγκεκριμένων δεδομένων, έπρεπε να δημιουργηθεί μια

---

<sup>2</sup> Το PyTorch είναι ένα framework μηχανικής μάθησης βασισμένο στη βιβλιοθήκη Torch, η οποία χρησιμοποιείται για εφαρμογές υπολογιστικής όρασης και επεξεργασία φυσικής γλώσσας. Το PyTorch παρέχει έναν τρόπο εύκολης κατασκευής νευρωνικών δικτύων και την αποτελεσματική εκπαίδευσή τους, γεγονός που έχει οδηγήσει στο να γίνει ένα από τα πιο δημοφιλή frameworks που χρησιμοποιούνται στην έρευνα.

κλάση, η οποία είναι απαραίτητη για να είναι εφικτή η εφαρμογή οποιουδήποτε μετασχηματισμού μέσω του PyTorch (Εικόνα 65).

```
import os
import pandas as pd
import torch
from torch.utils.data import Dataset
from skimage import io

class XrayChest(Dataset):
    def __init__(self, csv_file, root_dir, transform=None):
        self.annotations = pd.read_csv(csv_file)
        self.root_dir=root_dir
        self.transform=transform

    def __len__(self):
        return len(self.annotations) #5856

    def __getitem__(self, index):
        img_path = os.path.join(self.root_dir,self.annotations.iloc[index, 0])
        image = io.imread(img_path)
        y_label = torch.tensor(int(self.annotations.iloc[index, 1]))

        if self.transform:
            image = self.transform(image)

        return (image, y_label.item())
```

Εικόνα 65 Το customDatasetXray.py η δομή της κλάσης που διαχειρίζεται το dataset μέσω του PyTorch

Οι μετασχηματισμοί που εφαρμόστηκαν στα δύο σύνολα δεδομένων trainset και testset ήταν οι εξής:

Αρχικά, ορίστηκε το μέγεθος των εικόνων σε  $224 \times 224$  και κάθε κανάλι του τανιστή (Tensor) κανονικοποιήθηκε με μέσο όρο (0,4823, 0,4823, 0,4823) και τυπική απόκλιση (0,2216, 0,2216, 0,2216), ώστε τα εικονοστοιχεία να κυμαίνονται από 0 έως 1 προκειμένου να βοηθήσει το μοντέλο να συγκλίνει στο στάδιο της εκπαίδευσης. Επίσης στο train dataset χρησιμοποιήθηκε επαύξηση δεδομένων εφαρμόζοντας τυχαία περιστροφή 15 μοιρών στα δεδομένα για να αυξηθεί τεχνητά το μέγεθος και η ποιότητα του πεδίου εκπαίδευσης. Μετά τον μετασχηματισμό των δεδομένων, ακολούθησε η κατασκευή του ταξινομητή.

### Μετρικές αξιολόγησης

Για την αξιολόγηση της ταξινόμησης απόδοσης των μοντέλων CNN, υπολογίζονται οι μετρικές *precision*, *recall*, *accuracy*, και *F1-score*, οι εξισώσεις των οποίων δίνονται παρακάτω:

$$Precision = \frac{TP}{TP + FP} \quad (8)$$

$$Recall = \frac{TP}{TP + FN} \quad (9)$$

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (10)$$

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (11)$$

*TP (True Positive)* - ο αριθμός των δειγμάτων πνευμονίας που είναι σωστά ταξινομημένα

*TN (True Negative)* - ο αριθμός των φυσιολογικών δειγμάτων που είναι σωστά ταξινομημένα

*FP (False Positive)* - ο αριθμός των φυσιολογικών δειγμάτων που είναι λανθασμένα ταξινομημένα

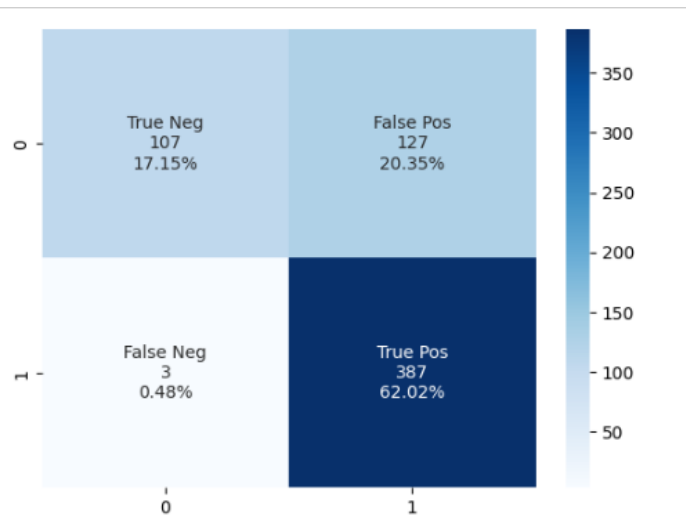
*FN (False Negative)* - ο αριθμός των δειγμάτων πνευμονίας που είναι λανθασμένα ταξινομημένα.

#### 4.2.1 Αποτελέσματα

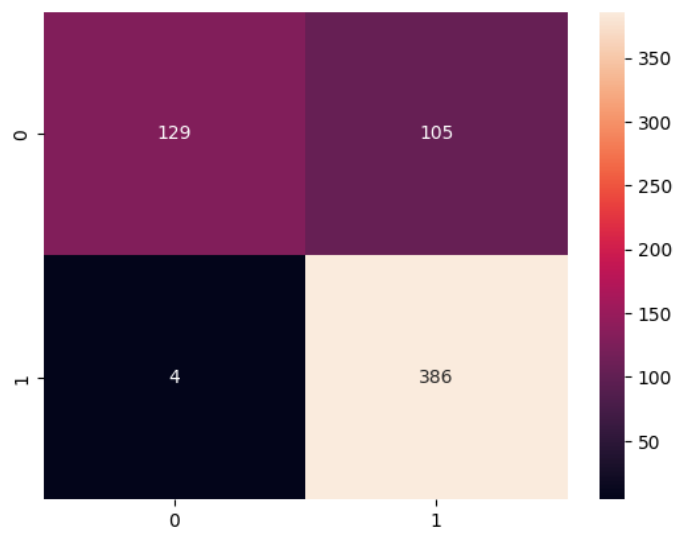
Αφού ολοκληρώθηκε η εκπαίδευση του μοντέλου χρησιμοποιήθηκε το test set (*test\_paths:624, test\_labels:624*) για την τελική αξιολόγηση του ταξινομητή μας. Αφού έγιναν οι απαραίτητοι μετασχηματισμοί στο set χρησιμοποιήθηκε το εκπαιδευμένο πλέον μοντέλο για να γίνουν οι προβλέψεις. Τα αποτελέσματα παρουσιάζονται στον παρακάτω πίνακα (Πίνακας 13 και Εικόνα 66).:

Πίνακας 13 Απόδοση CNN ταξινομητή (5 εποχές)

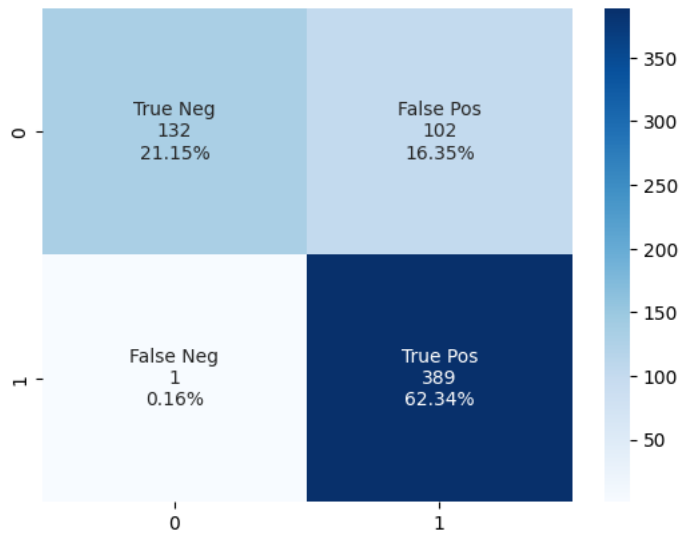
	<b>Precision</b>	<b>Recall</b>	<b>F1-Score</b>	<b>Accuracy</b>	<b>Training Time(minutes)</b>	<b>Environment</b>
<b>ResNet-18</b>	0.7529	0.9923	0.8562	0.7917	23 min 29 sec	Localhost
<b>VGG16</b>	0.7862	0.9897	0.8763	0.8253	68 min 23 sec	Colab
<b>Densenet121</b>	0.7923	0.9974	0.8831	0.8349	46 min 17 sec	Colab
<b>MobileNet_V2</b>	0.7217	0.9974	0.8375	0.7580	24 min 59 sec	Localhost
<b>ResNet-34</b>	0.8272	0.9821	0.8980	0.8606	25 min 45 sec	Localhost



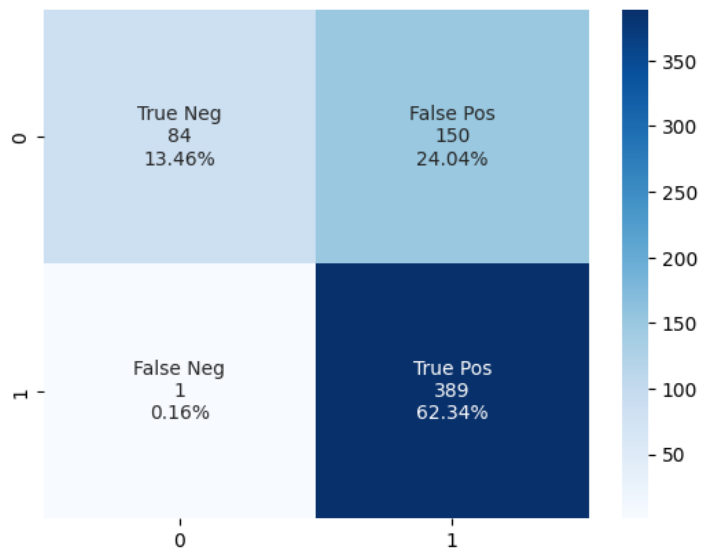
Εικόνα 66 Confusion Matrix ταξινομητή CNN (ResNet-18)



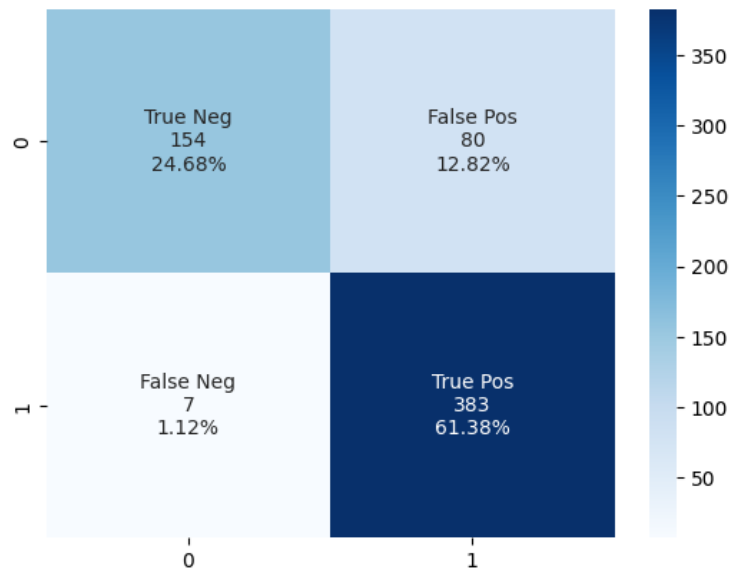
Εικόνα 67 Confusion Matrix ταξινομητή CNN (VGG16)



Εικόνα 68 Confusion Matrix ταξινομητή CNN (DenseNet-121)



Εικόνα 69 Confusion Matrix ταξινομητή CNN (MobileNet\_v2)



Εικόνα 70 Confusion Matrix ταξινομητή CNN (ResNet-34)

Τα αποτελέσματα φαίνεται ότι το ResNet-34 πέτυχε το υψηλότερο F1-score (0,8980) και accuracy (0,8606), αναδεικνύοντας την αποτελεσματικότητά του στη συνολική απόδοση της διαδικασίας της ταξινόμησης. Το μοντέλο επιδεικνύει μια καλά ισορροπημένη αντιστάθμιση μεταξύ precision (0,8272) και recall (0,9821), υποδεικνύοντας την ικανότητά του να ταξινομεί σωστά τις θετικές περιπτώσεις, ελαχιστοποιώντας παράλληλα τα ψευδώς θετικά αποτελέσματα. Ο χρόνος εκπαίδευσης σε περιβάλλον localhost ήταν 25 λεπτά και 45 δευτερόλεπτα. Το ResNet-18, εκπαιδεύτηκε στο localhost σε 23 λεπτά και 29 δευτερόλεπτα, και πετυχαίνει ένα υψηλό F1-score 0,8592 και accuracy 0,7917. Με precision 0,7529 και εντυπωσιακή recall 0,9923, το ResNet-18 παρέχει ορθό εντοπισμό θετικών περιπτώσεων, διατηρώντας παράλληλα ένα σχετικά υψηλό precision. Αυτό το καθιστά ιδιαίτερα κατάλληλο για εφαρμογές όπου η ελαχιστοποίηση των ψευδώς θετικών αποτελεσμάτων είναι μεγαλύτερης σημασίας.

Το VGG16, αν και απαιτεί μεγαλύτερο χρόνο εκπαίδευσης στο 68 λεπτά και 23 δευτερόλεπτα σε περιβάλλον Google Colab, παρουσιάζει ανταγωνιστικές επιδόσεις με F1-score 0,8763 και accuracy 0,8253. Αυτό υποδηλώνει ότι το VGG16 μπορεί να είναι πιο απαιτητική από υπολογιστική άποψη, αλλά αντισταθμίζει με μια ισχυρή ισορροπία μεταξύ precision και recall. Συνεχίζοντας με το DenseNet121, που εκπαιδεύτηκε στο Colab για 46 λεπτά και 17 δευτερόλεπτα, επιδεικνύει υψηλό F1-score 0,8831, και υψηλό recall 0,9974, ενώ το precision είναι χαμηλότερο 0,7923. Τέλος το MobileNet\_V2, που



εκπαιδεύτηκε στο localhost σε 24 λεπτά και 59 δευτερόλεπτα, παρουσιάζει υψηλή recall 0,9974, καθιστώντας το ικανό στον σωστό εντοπισμό θετικών περιπτώσεων. Ωστόσο, το χαμηλότερο precision (0,7217) και F1-score (0,8375) υποδηλώνουν μεγαλύτερη πιθανότητα ψευδώς θετικών περιστατικών.

Στον παρακάτω πίνακα (Πίνακας 14) αναφέρονται τα αποτελέσματα περαιτέρω πειραμάτων στην προσπάθεια επίτευξης καλύτερων αποτελεσμάτων εκπαιδεύοντας τον ταξινομητή για 50 εποχές. Τα προεκπαιδευμένα μοντέλα που χρησιμοποιήθηκαν στην προσπάθεια αυτή είναι το VGG16, ResNet-18 και το ResNet50.

Πίνακας 14 Αποδόσεις ταξινομητή με χρήση προ εκπαιδευμένων CNN μοντέλων (50 εποχές) χωρίς συνθετικά δείγματα

	<b>Precis ion</b>	<b>Rec all</b>	<b>F1- Score</b>	<b>Accur acy</b>	<b>Training Time(minutes)</b>	<b>Environ ment</b>
<b>VGG- 16</b>	0.7702	0.99 74	0.8692	0.8125	400.74 min/ 6 hrs 40 min 44 sec	Collab
<b>ResNet -18</b>	0.7971	0.99 74	0.8861	0.8183	276.11 min / 4 hrs 36 min 6 se	Localhost
<b>ResNet -50</b>	0.7718	0.99 74	0.8702	0.8141	94.81 min / 1 hr 34 min 48 sec	Collab

### 4.3 Classifier με χρήση Weighted Random Sampler

Στη συνέχεια, επιχειρήθηκε η βελτιστοποίηση του παραπάνω ταξινομητή ως προς την πρόβλεψη της κλάσης 0 (Normal) χρησιμοποιώντας μια τεχνική Data sampling. Η συνάρτηση που χρησιμοποιήθηκε είναι η WeightedRandomSampler του PyTorch για τη δημιουργία ενός προσαρμοσμένου δειγματολήπτη (sampler) δεδομένων για την εκπαίδευση ενός νευρωνικού δικτύου.

Η WeightedRandomSampler επιτρέπει να ανατεθούν διαφορετικά βάρη σε κάθε δείγμα στο σύνολο δεδομένων. Αυτά τα βάρη επηρεάζουν την πιθανότητα επιλογής ενός δείγματος κατά τη διαδικασία δειγματοληψίας. Όσο μεγαλύτερο είναι το βάρος που ανατίθεται σε ένα δείγμα, τόσο πιο πιθανό είναι να επιλεγεί κατά τη διάρκεια της δειγματοληψίας. Επίσης, μπορεί να χρησιμοποιηθεί για υπερδειγματοληψία (Oversampling), αναθέτοντας υψηλότερα βάρη στα δείγματα της μειονοτικής κλάσης, καθιστώντας έτσι πιο πιθανό να επιλεγούν κατά τη διάρκεια της εκπαίδευσης. Ακριβώς αυτό έγινε και στην παρούσα εργασία, δίνοντας στη μειονοτική κλάση (Normal) μεγαλύτερο βάρος (0.0007451564828614009), ενώ στην πλειοψηφική κλάση (Pneumonia)

μικρότερος βάρος (0.00025799793601651185) (Εικόνα 71) και ακολούθησε επανεκπαίδευση του μοντέλου με τις ίδιες παραμέτρους (Πίνακας 2).

```
class_weights=[]
for root, subdir,files in os.walk(root_dir):
    if len(files)>0:
        class_weights.append(1/len(files))
        print(class_weights)

sample_weights=[0]*len(train_dataset)

for idx, (data,label) in enumerate(train_dataset):
    class_weight=class_weights[label]
    sample_weights[idx] = class_weight

sampler=WeightedRandomSampler(sample_weights, num_samples=len(sample_weights), replacement=True)
|

[0.0007451564828614009]
[0.0007451564828614009, 0.00025799793601651185]
```

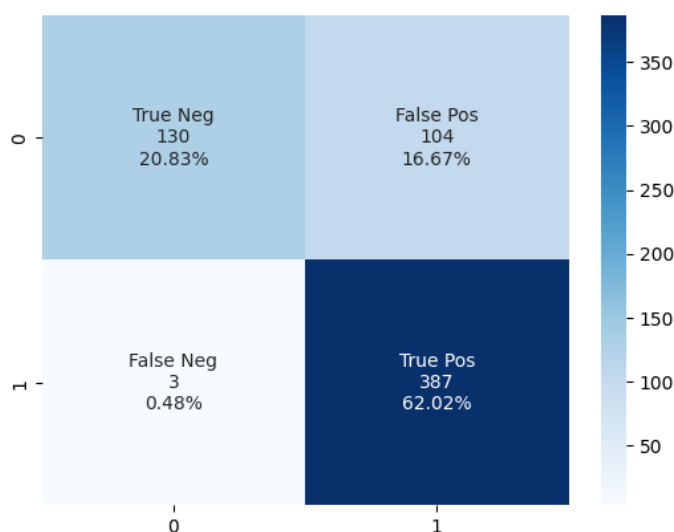
Εικόνα 71 Δημιουργία WeightedRandomSampler και ανάθεση βαρών

### 4.3.1 Αποτελέσματα

Μετά την εκπαίδευση του ταξινομητή αυτήν φορά χρησιμοποιώντας την μέθοδο για Weighted Random Sampler για χρησιμοποιώντας το προεκπαιδευμένο μοντέλο ResNet-18 έχουμε τα παρακάτω αποτελέσματα.

Πίνακας 15 Απόδοση CNN ταξινομητή με χρήση WeightedRandomSampler (5 εποχές)

	<b>Precision</b>	<b>Recall</b>	<b>F1-Score</b>	<b>Accur acy</b>	<b>Training Time(minutes)</b>	<b>Environ ment</b>
<b>ResNet -18</b>	0.7882	0.9923	0.8785	0.8285	30 min 3 sec	Localhost



Εικόνα 72 Confusion Matrix Ταξινομητή CNN με χρήση WeightedRandomSampler

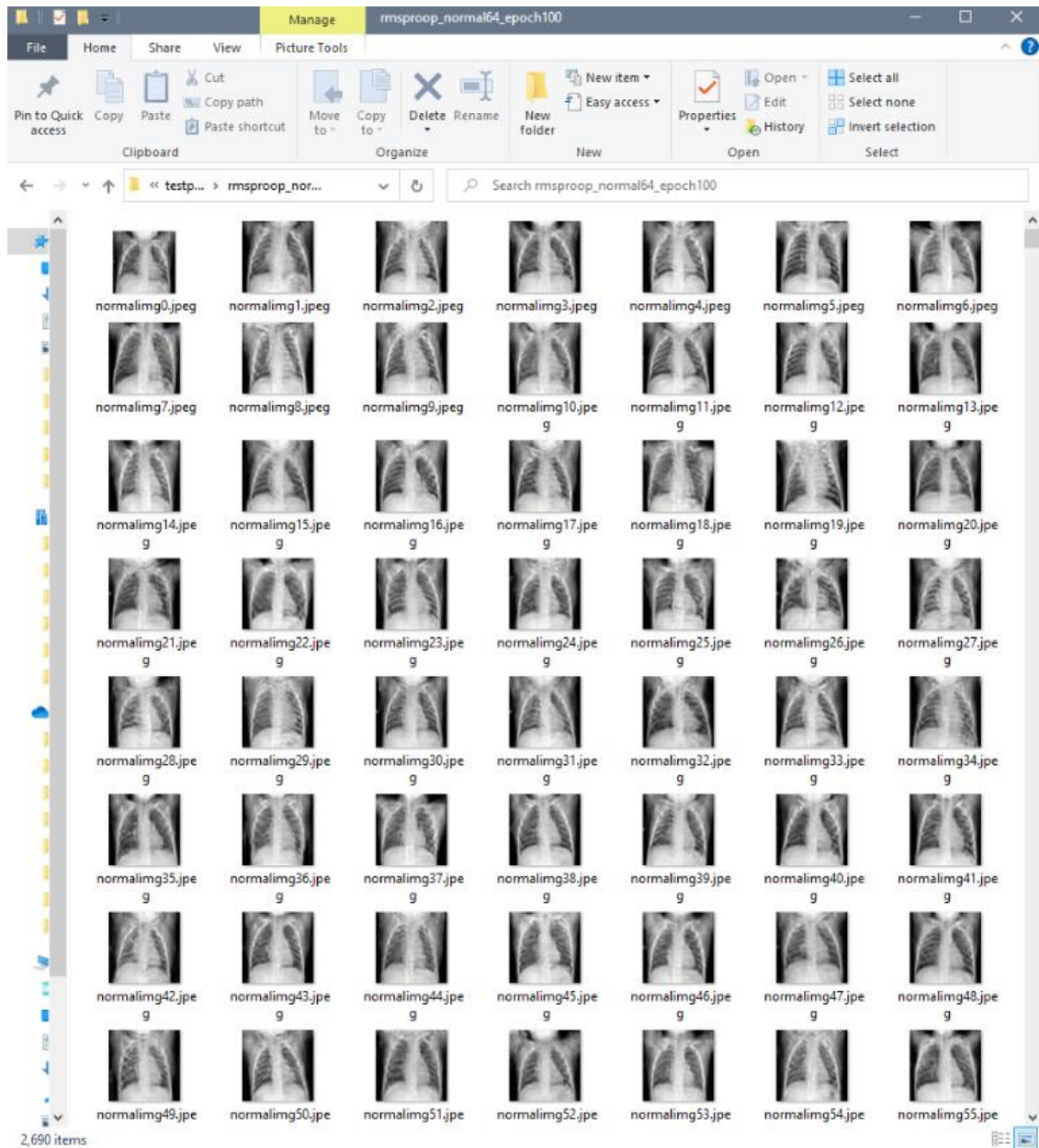
Παρατηρώντας τα αποτελέσματα για το (Πίνακας 15 και Εικόνα 72) και συγκρίνοντάς τα με τα προηγούμενα (Πίνακας 13 και Εικόνα 66) για το ResNet-18 προκύπτουν τα ακόλουθα. Φαίνεται ότι η ακρίβεια ήταν 0.7917 και μετά τη χρήση του WeightedRandomSampler βελτιώθηκε ελαφρά και έφτασε το 0.8285. Όσον αφορά το Precision στην πρώτη περίπτωση είναι 0.7529, που σημαίνει ότι από όλες τις περιπτώσεις που προβλέφθηκαν ως θετικές, περίπου το 75.29% ήταν πράγματι θετικές, ενώ στη δεύτερη περίπτωση είναι 0.7882, (που σημαίνει το 78.82% των θετικών προβλέψεων ήταν σωστές) παρουσιάζοντας μια μικρή βελτίωση σε σύγκριση με την πρώτη.

Το recall (Sensitivity) είναι 0.9923, παραμένοντας ίδιο και στις δύο περιπτώσεις. Αυτό σημαίνει ότι το μοντέλο αναγνώρισε με συνέπεια περίπου το 99.23% των θετικών περιπτώσεων σωστά. Η ειδικότητα (Ποσοστό αληθώς αρνητικών) στην πρώτη περίπτωση είναι 0.4572 υποδηλώνοντας ότι μόνο το 45.72% περίπου των πραγματικών αρνητικών περιπτώσεων αναγνωρίστηκε σωστά, ενώ στη δεύτερη περίπτωση είναι 0.5555 γεγονός που δείχνει βελτίωση σε σύγκριση με την πρώτη δηλαδή το 55.55% των αρνητικών περιπτώσεων αναγνωρίστηκαν σωστά.

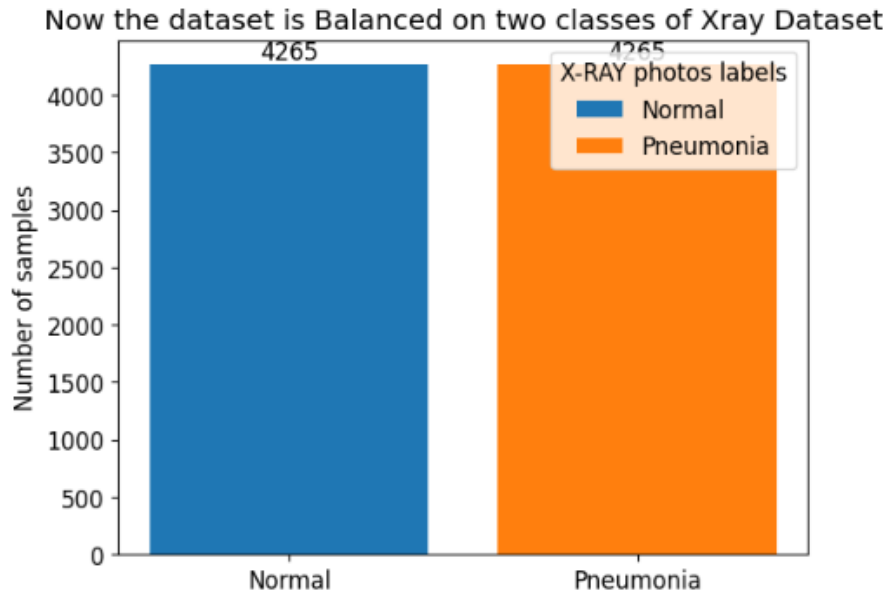
Ολοκληρώνοντας, προκύπτει ότι με τη χρήση του WeightedRandomSampler στην εκπαίδευση του μοντέλου επιτεύχθηκε μια μικρή βελτίωση στο Accuracy, στο Precision και στο Specificity, ενώ το Recall παραμένει το ίδιο υποδεικνύοντας σταθερή απόδοση στον εντοπισμό θετικών περιπτώσεων. Το μοντέλο με την WeightedRandomSampler παρουσιάζει μεγαλύτερη ικανότητα να προβλέπει σωστά αρνητικές περιπτώσεις σε σχέση με πριν, όπως αποδεικνύεται από τη βελτίωση που φαίνεται στο Specificity.

#### **4.4 Classifier με χρήση συνθετικών δειγμάτων από το cGAN**

Στη συνέχεια, τα μοντέλα cGAN, που πέτυχαν τα υψηλότερα score FID και KID (Πίνακας 11), χρησιμοποιήθηκαν για να δημιουργηθούν συνθετικές εικόνες για την κλάση Normal, η οποία είναι η μειωτική. Οι εικόνες αυτές προστέθηκαν στο αρχικό dataset. Δημιουργήθηκαν συνολικά 2690 νέες συνθετικές εικόνες (Εικόνα 73) της κλάσης Normal και το dataset πλέον ήταν ισορροπημένο (Εικόνα 74).



Εικόνα 73 Οι νέες συνθετικές εικόνες της κλάσης Normal μετά τη χρήση του μοντέλου



Εικόνα 74 Chestxray dataset μετά την προσθήκη των νέων συνθετικών εικόνων της κλάσης Normal

Στη συνέχεια, εκπαιδεύτηκε ξανά ο ταξινομητής με τις ίδιες παραμέτρους (Πίνακας 2) κάνοντας χρήση των προεκπαιδευμένων μοντέλων ResNet-18, MobileNet\_v2, ResNet-34 και DenseNet-121. Για κάθε πείραμα που εκτελέστηκε τα προεκπαιδευμένα μοντέλα παρέμεναν τα ίδια όπως και οι υπερπαραμέτροι ενώ το μόνο πράγμα που άλλαζε σε κάθε κύκλο πειραμάτων ήταν τα συνθετικά δείγματα που προσθέταμε στο dataset, καθώς δοκιμάστηκαν δείγματα και από τα τρία cGAN.

Επιαναλαμβάνοντας την ίδια διαδικασία και χρησιμοποιώντας εικόνες από τα παραγόμενα μοντέλα (cGAN) προέκυψαν τα παρακάτω αποτελέσματα.

#### 4.4.1 Αποτελέσματα με χρήση εικόνων από cGAN(RMSProp)

Το πρώτο πείραμα πραγματοποιήθηκε με τη βοήθεια 4 προεκπαιδευμένων μοντέλων και αφορά την εκπαίδευση του ταξινομητή με την προσθήκη συνθετικών δειγμάτων που παρήχθησαν από το cGAN(RMSProp). Ο Πίνακας 16 και η Εικόνα 75 παρουσιάζουν τα αποτελέσματα του πειράματος ανά προεκπαιδευμένο μοντέλο και ανά μετρική που εξετάστηκε.

Η μετρική Precision μετρά την ακρίβεια των θετικών προβλέψεων. Ένα υψηλό precision υποδηλώνει λιγότερα ψευδώς θετικά αποτελέσματα. Το ResNet-34 παρουσιάζει την υψηλότερη τιμή (0,8246), ακολουθούμενο από το DenseNet-121 (0,8152). Αυτό σημαίνει ότι τα συγκεκριμένα μοντέλα παρουσιάζουν

καλύτερη ικανότητα να κάνουν ακριβείς θετικές προβλέψεις, ενώ τα ResNet-18 (0,7638) και MobileNet\_v2 (0,7683) παρουσιάζουν ελαφρώς χαμηλότερες τιμές. Όσον αφορά τη μετρική Recall (ευαισθησία - sensitivity), αυτή αξιολογεί την ικανότητα του μοντέλου να εντοπίζει το ποσοστό των θετικών δεδομένων που ταξινομήθηκαν σωστά. Στο συγκεκριμένο πείραμα, το προεκπαιδευμένο μοντέλο MobileNet\_v2 ξεχωρίζει πετυχαίνοντας την υψηλότερη τιμή (0,9948), υπερέχοντας έτσι στον εντοπισμό θετικών περιπτώσεων σε σχέση με τα υπόλοιπα. Το ResNet-18 ακολουθεί με μικρή διαφορά πετυχαίνοντας την τιμή 0,9871. Εξίσου υψηλή τιμή πετυχαίνουν τα μοντέλα ResNet-34 (0,9769) και DenseNet-121(0,9846).

Συνεχίζοντας, καταγράφονται τα αποτελέσματα για τη μετρική F1-Score, που αποτελεί τον μέσο όρο της precision και recall, και αντικατοπτρίζει την ισορροπία μεταξύ των δύο μετρικών. Το ResNet-34 επιτυγχάνει το υψηλότερο F1-Score (0,8943), υποδηλώνοντας την ισορροπημένη απόδοσή του σε ακρίβεια και ανάκληση. Το DenseNet-121 ακολουθεί από κοντά με F1-Score 0,8919, παρουσιάζοντας παρόμοια ισορροπία, ενώ το MobileNet\_v2 και το ResNet-18 παρουσιάζουν ελαφρώς χαμηλότερες τιμές, 0,8670 και 0,8612 αντίστοιχα.

Σχετικά με τη μετρική Accuracy, που μας δείχνει τη συνολική ορθότητα του μοντέλου, σε όλες τις κλάσεις, το ResNet-34 εμφανίζει την υψηλότερη τιμή 0,8557 ακολουθούμενο από το DenseNet-121 με τιμή 0,8509. Το εύρημα αυτό υποδεικνύει την ικανότητά τους να πετυχαίνουν ικανοποιητικές προβλέψεις για όλες τις κλάσεις. Τα μοντέλα MobileNet\_v2 και ResNet-18 σημείωσαν ελαφρώς χαμηλότερες τιμές, ήτοι 0,8092 και 0,8012 αντίστοιχα.

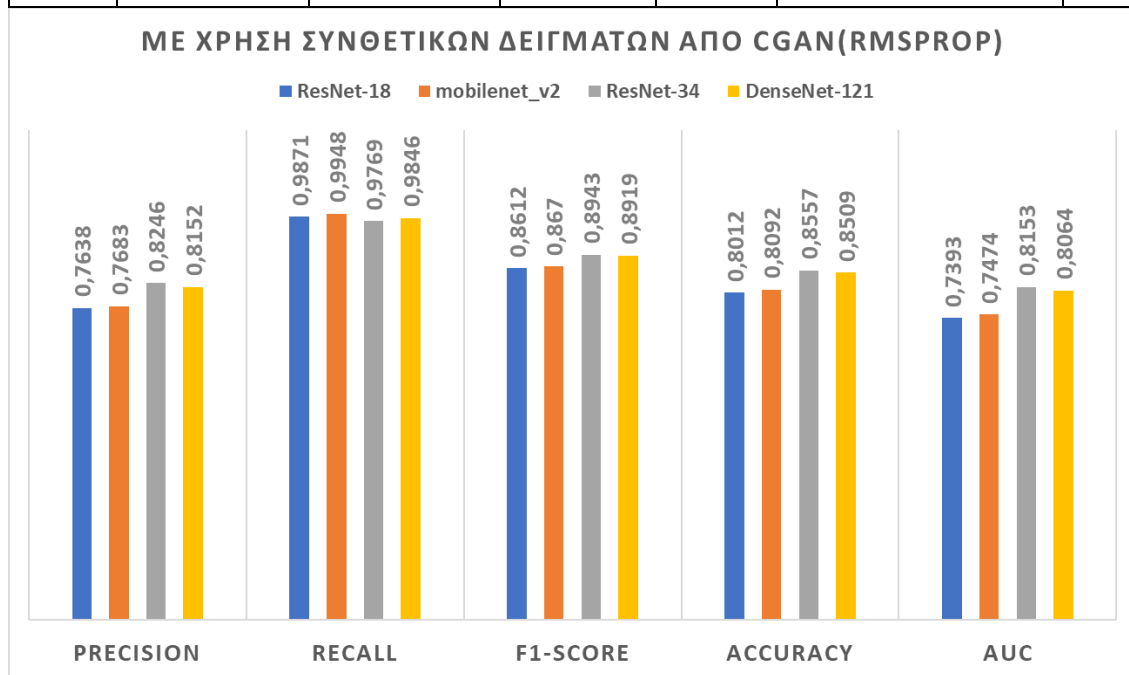
Η AUC μετρά την ικανότητα του ταξινομητή να διακρίνει μεταξύ των κλάσεων. Το ResNet-34 διαθέτει την υψηλότερη AUC (0,8153), υποδεικνύοντας την ισχυρή διακριτική του ικανότητα μεταξύ θετικών και αρνητικών περιπτώσεων. Το DenseNet-121 ακολουθεί με AUC 0,8064, ενώ το MobileNet\_v2 (0,7474) και το ResNet-18 (0,7393) επιτυγχάνουν ελαφρώς χαμηλότερες τιμές AUC.

Τέλος, σχετικά με τον χρόνο εκπαίδευσης του κάθε μοντέλου, το ResNet-34 απαιτούσε τον μεγαλύτερο χρόνο εκπαίδευσης (29 λεπτά και 13 δευτερόλεπτα) μεταξύ των μοντέλων. Ακολουθούν τα μοντέλα MobileNet\_v2 (25 λεπτά 35 δευτερόλεπτα) και DenseNet-121 (24 λεπτά 6 δευτερόλεπτα). Το ResNet-18

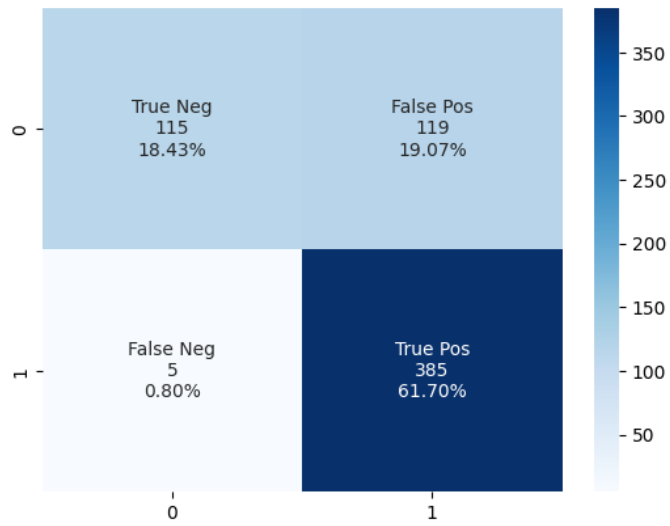
χρειάστηκε τον λιγότερο χρόνο εκπαίδευσης (22 λεπτά 15 δευτερόλεπτα) χωρίς όμως να υπάρξει σημαντική διαφορά.

Πίνακας 16 Αποτελέσματα Classifier με τη χρήση συνθετικών δειγμάτων από το cGAN (RMSprop)

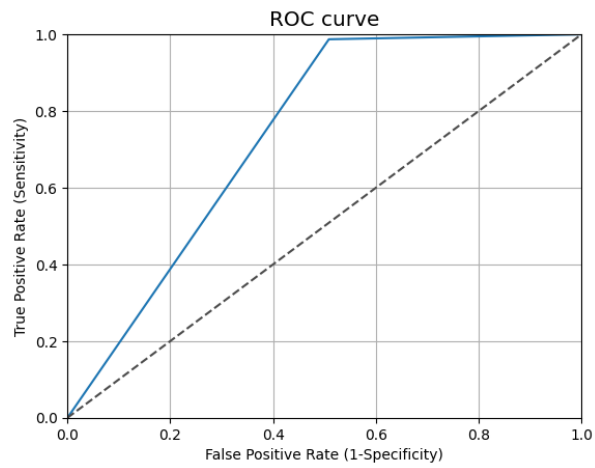
Pretrain model	Precision	Recall	F1-Score	Accuracy	AUC	Training Time(minute s)	Environment
<b>ResNet-18</b>	0.7638	0.9871	0.8612	0.8012	0.7393	22 min 15 sec	Localhost
<b>MobileNet_v2</b>	0.7683	0.9948	0.8670	0.8092	0.7474	25 min 35 sec	Localhost
<b>ResNet-34</b>	0.8246	0.9769	0.8943	0.8557	0.8153	29 min 13 sec	Localhost
<b>DenseNet-121</b>	0.8152	0.9846	0.8919	0.8509	0.8064	24 min 6 sec	Colab



Εικόνα 75 Οπτικοποίηση αποτελεσμάτων Classifier με τη χρήση συνθετικών δειγμάτων από το cGAN (RMSprop)

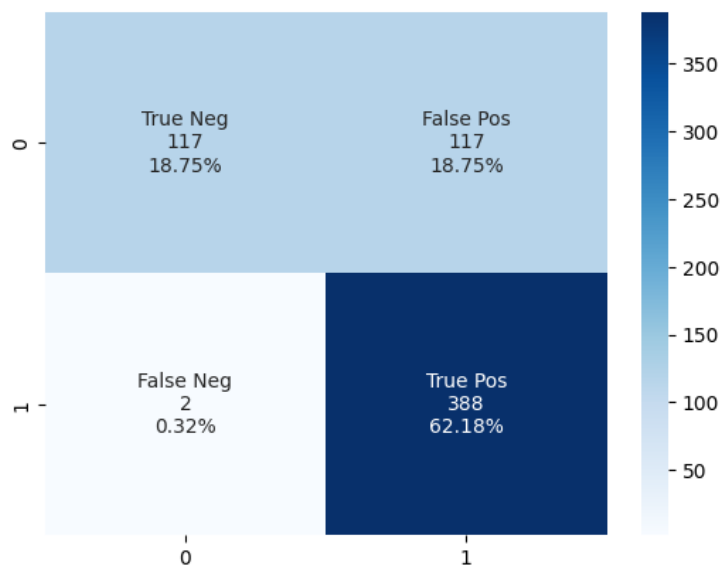


Εικόνα 76 Confusion Matrix για ταξινομητή (ResNet-18) με χρήση συνθετικών εικόνων (cGAN RMSprop)

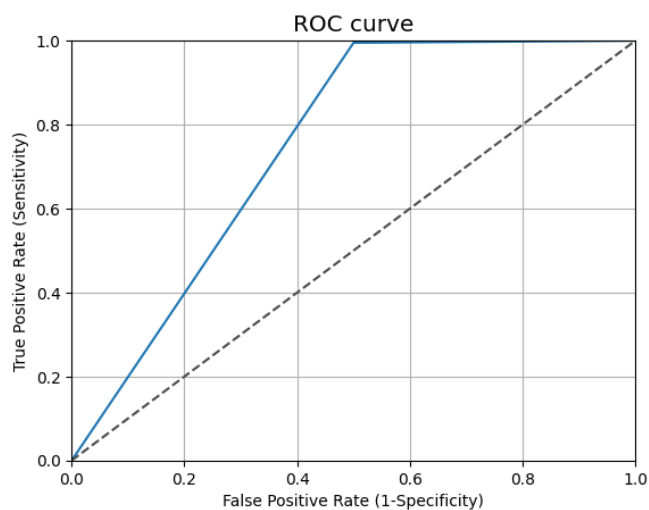


Εικόνα 77 Καμπύλη ROC

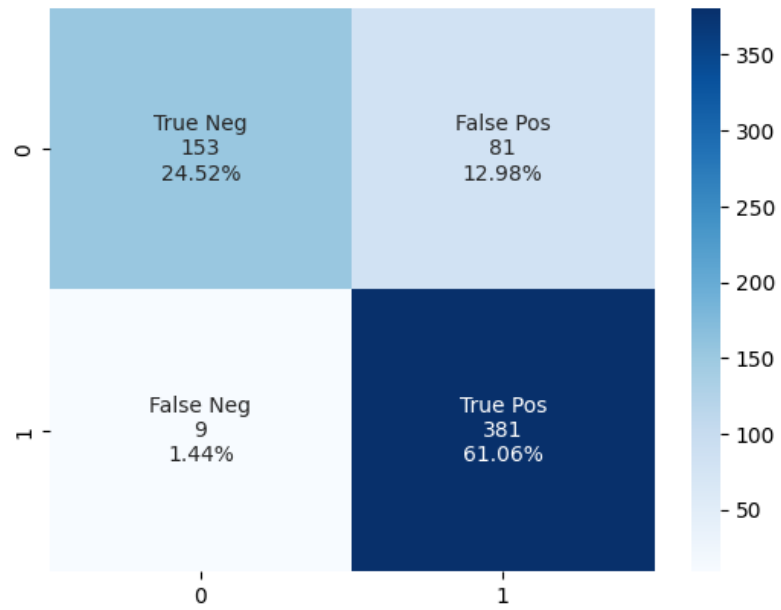




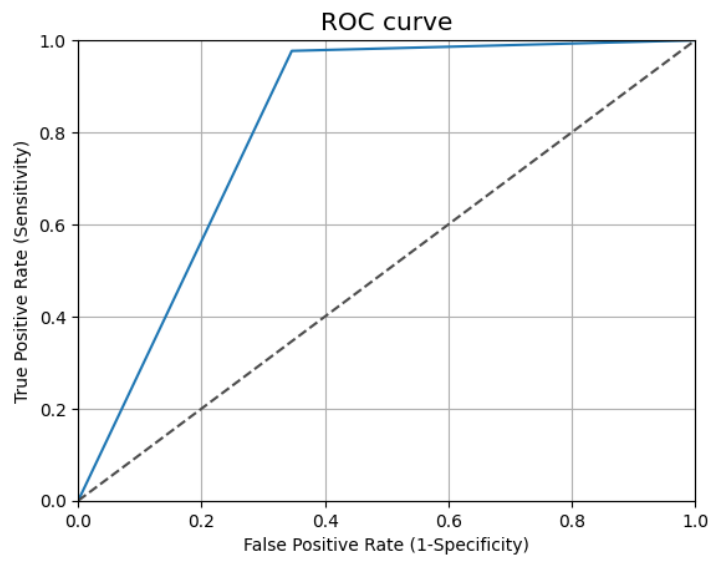
Εικόνα 78 Confusion Matrix για ταξινομητή (MobileNet\_V2) με χρήση συνθετικών εικόνων (cGAN RMSprop)



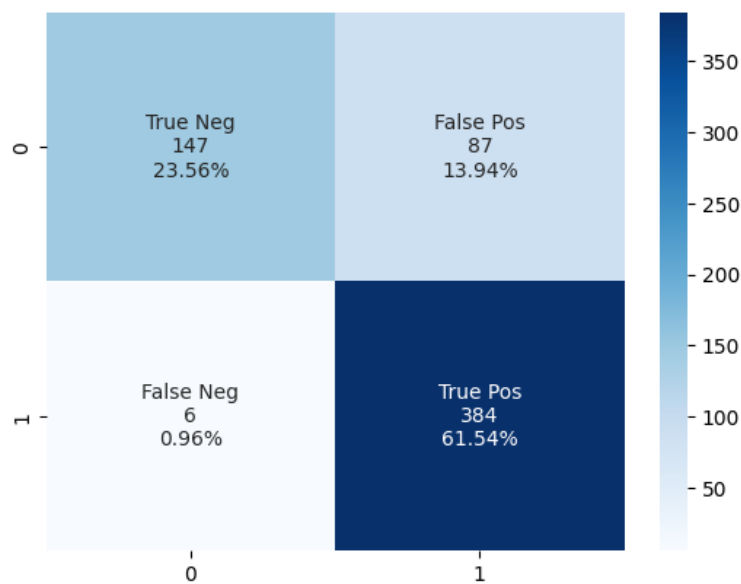
Εικόνα 79 Καμπύλη ROC (MobileNet\_V2)



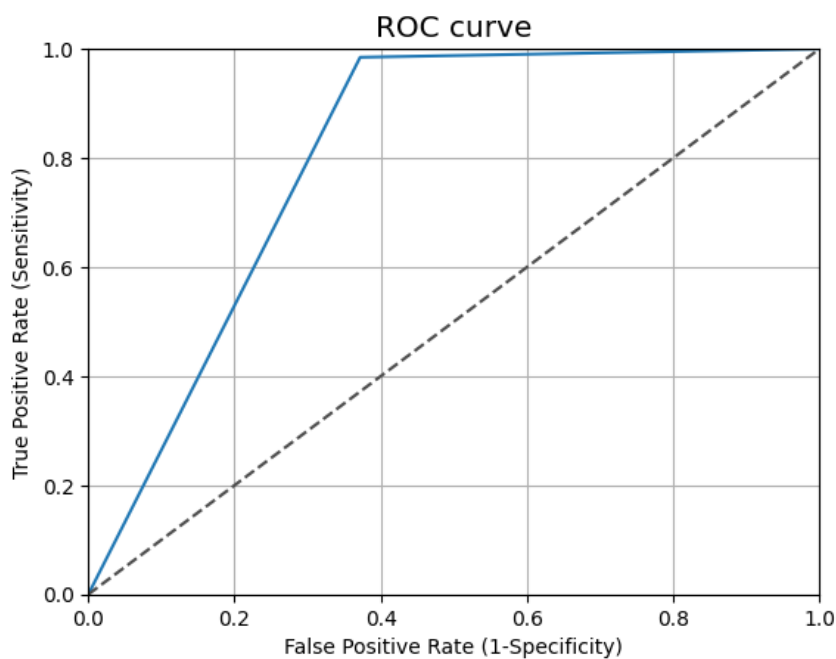
Εικόνα 80 Confusion Matrix για ταξινομητή (ResNet34) με χρήση συνθετικών εικόνων (cGAN RMSprop)



Εικόνα 81 Καμπύλη ROC (ResNet34)



Εικόνα 82 Confusion Matrix (DenseNet121)



Εικόνα 83 Καμπύλη ROC (DenseNet-121)

#### 4.4.2 Αποτελέσματα με χρήση εικόνων από cGAN(64 εποχές και Adam Optimizer)

Το δεύτερο πείραμα πραγματοποιήθηκε με τη βοήθεια 4 προεκπαιδευμένων μοντέλων και αφορά την εκπαίδευση του ταξινομητή με την προσθήκη συνθετικών δειγμάτων που παρήχθησαν από το cGAN(Adam). Ο Πίνακας 17 και η Εικόνα 84 παρουσιάζουν τα αποτελέσματα του πειράματος ανά προεκπαιδευμένο μοντέλο και ανά μετρική, η οποία εξετάστηκε.

Το ResNet-18 παρουσιάζει την υψηλότερη τιμή στο precision (0,8377), υποδεικνύοντας την ικανότητά του στην ακριβή πρόβλεψη θετικών περιπτώσεων ακολουθούμενο από το DenseNet-121 το οποίο πετυχαίνει τιμή στο precision 0,8105.

Το MobileNet\_v2 και το ResNet-34 παρουσιάζουν χαμηλότερες τιμές precision (0,7369 και 0,7755, αντίστοιχα), υποδηλώνοντας ελαφρώς λιγότερο ακριβείς θετικές προβλέψεις σε σύγκριση με τα προηγούμενα δύο.

Όσον αφορά τη μετρική recall, το ResNet-34 επιδεικνύει εξαιρετική ανάκληση (0,9923), υποδεικνύοντας την ικανότητά του να συλλαμβάνει σχεδόν όλες τις πραγματικές θετικές περιπτώσεις. Ακολουθεί το DenseNet-121 (0,9871), το οποίο παρουσιάζει επίσης υψηλή ανάκληση. Το MobileNet\_v2 και το ResNet-18 παρουσιάζουν σχετικά χαμηλότερες τιμές από τα άλλα δύο, αλλά και πάλι είναι σημαντικά υψηλές (0,9769 και 0,9794, αντίστοιχα).

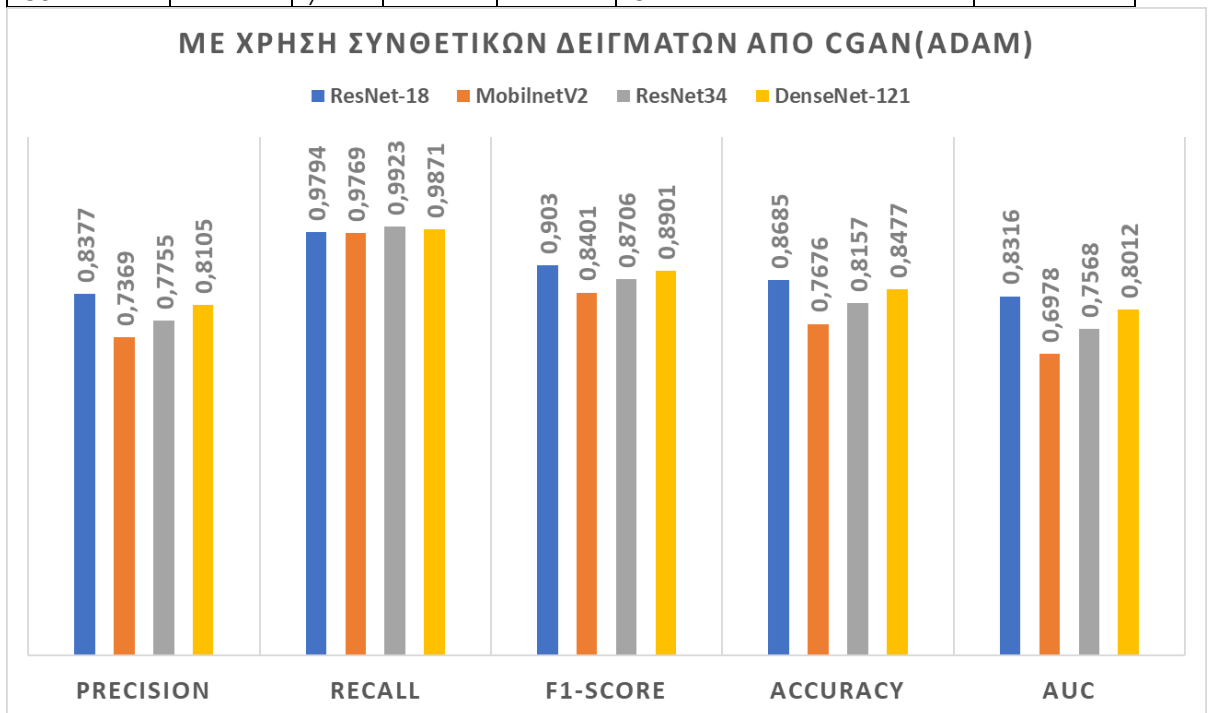
Συνεχίζοντας καταγράφονται τα αποτελέσματα για τη μετρική F1-Score με το ResNet-18 να επιτυγχάνει την υψηλότερη τιμή (0,903), υποδεικνύοντας μια ισορροπημένη απόδοση τόσο στο precision όσο και στο recall. Αυτό υποδηλώνει ότι το ResNet-18 επιτυγχάνει μια καλή ισορροπία μεταξύ ακριβών θετικών προβλέψεων και καταγραφής σχετικών περιπτώσεων. Το DenseNet-121 ακολουθεί από κοντά με F1-Score 0,8901, παρουσιάζοντας παρόμοια ισορροπία αλλά ελαφρώς χαμηλότερη από το ResNet-18. Το MobileNet\_v2 και το ResNet-34 παρουσιάζουν συγκριτικά χαμηλότερες τιμές F1-Score (0,8401 και 0,8706, αντίστοιχα). Επίσης, το ResNet-18 προηγείται και στη μετρική Accuracy με τιμή 0,8685, γεγονός που υποδηλώνει τη συνολική ορθότητά του σε όλες τις κλάσεις. Το DenseNet-121 ακολουθεί από κοντά με ακρίβεια 0,8477. Τα ResNet-34 και MobileNet\_v2 παρουσιάζουν ελαφρώς χαμηλότερες τιμές στο Accuracy (0,8157 και 0,7676, αντίστοιχα). Το ResNet-18 συνεχίζει να υπερτερεί και στη μετρική AUC πετυχαίνοντας την υψηλότερη τιμή (0,8316), υποδεικνύοντας έτσι ισχυρή διακριτική ικανότητα μεταξύ θετικών και αρνητικών περιπτώσεων. Το DenseNet-121 ακολουθεί από κοντά με AUC 0,8012. Ωστόσο, σε χαμηλότερα επίπεδα κυμαίνονται οι τιμές των ResNet34 και MobileNet\_v2 με τιμές 0,7568 και 0,6978 αντίστοιχα.

Τέλος, σχετικά με τους χρόνους εκπαίδευσης, το ResNet-18 χρειάστηκε τον λιγότερο χρόνο (22 λεπτά και 38 δευτερόλεπτα) ακολουθούμενο από το

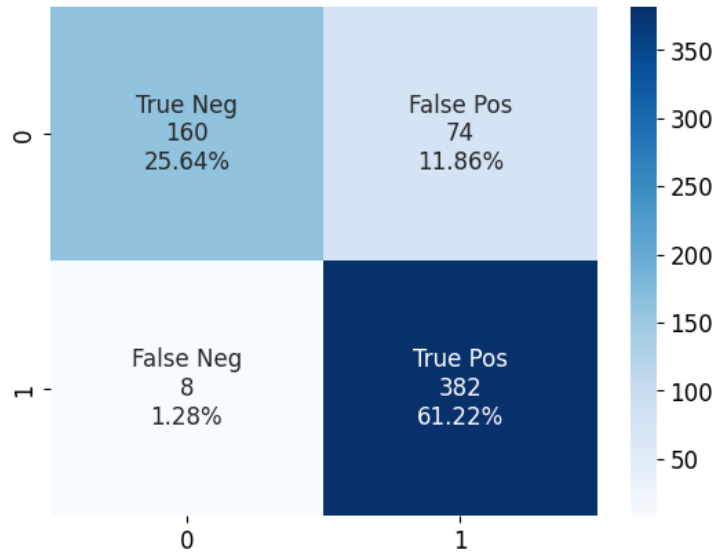
MobileNet\_v2 με χρόνο 24 λεπτά και 37 δευτερόλεπτα. Τα ResNet-34 και DenseNet-121 χρειάστηκαν περισσότερο χρόνο με χρόνους (30 λεπτά και 27 δευτερόλεπτα) και (47 λεπτά και 33 δευτερόλεπτα) αντίστοιχα.

Πίνακας 17 Αποτελέσματα Classifier με τη χρήση συνθετικών δειγμάτων από το cGAN (64 εποχές Adam Optimizer)

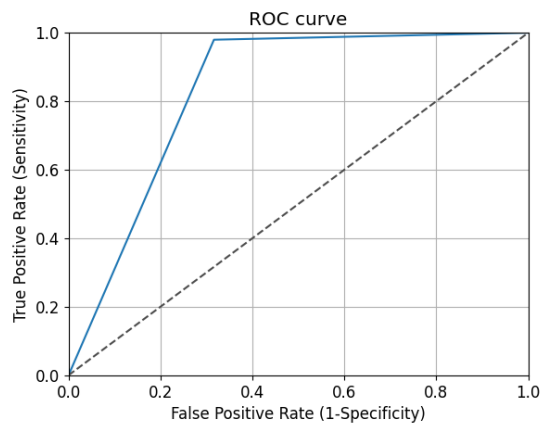
	<b>Precision</b>	<b>Recall</b>	<b>F1-Score</b>	<b>Accuracy</b>	<b>AUC</b>	<b>Training Time(minutes)</b>	<b>Environment</b>
<b>ResNet-18</b>	0.8377	0.9794	0.9030	0.8685	0.8316	22 min 38 sec	Localhost
<b>MobilnetV2</b>	0.7369	0.9769	0.8401	0.7676	0.6978	24 min 37 sec	Localhost
<b>ResNet34</b>	0.7755	0.9923	0.8706	0.8157	0.7568	30 min 27 sec	Localhost
<b>DenseNet-121</b>	0.8105	0.9871	0.8901	0.8477	0.8012	47 min 33 sec	Colab



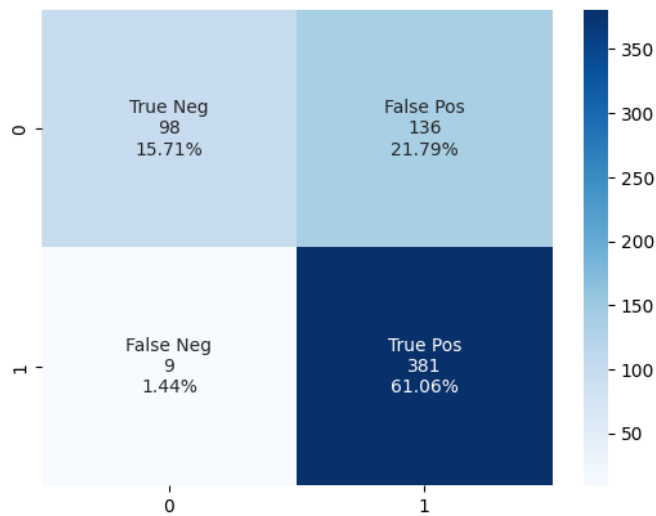
Εικόνα 84 Οπτικοποίηση αποτελεσμάτων Classifier με τη χρήση συνθετικών δειγμάτων από το cGAN (Adam)



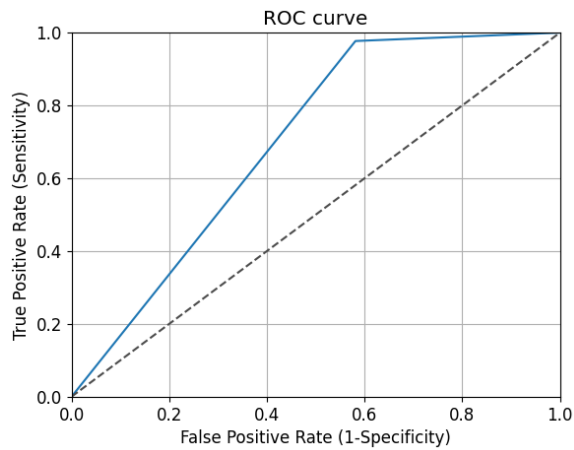
Εικόνα 85 Confusion Matrix για ταξινομητή (ResNet-18) με χρήση συνθετικών εικόνων (cGAN 64 εποχές Adam Optimizer)



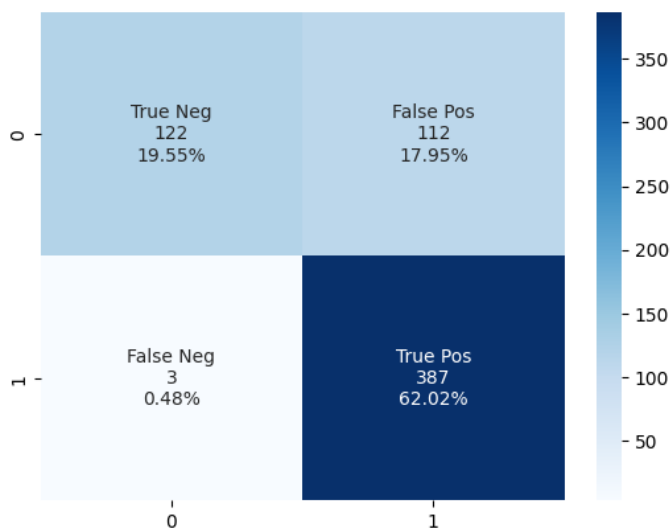
Εικόνα 86 Καμπύλη ROC (ResNet-18)



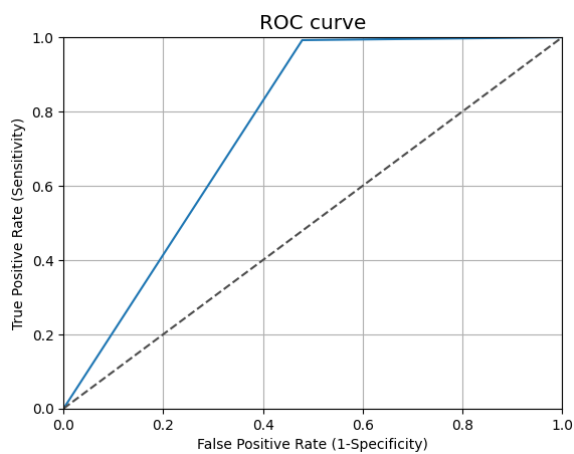
Εικόνα 87 Confusion Matrix για ταξινομητή (MobileNetV2) με χρήση συνθετικών εικόνων (cGAN 64 εποχές Adam Optimizer)



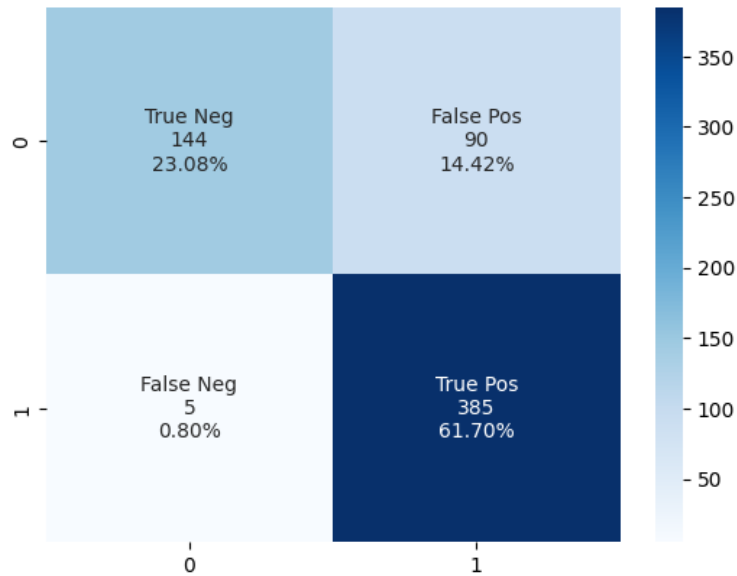
Εικόνα 88 Καμπύλη ROC(MobilNetV2)



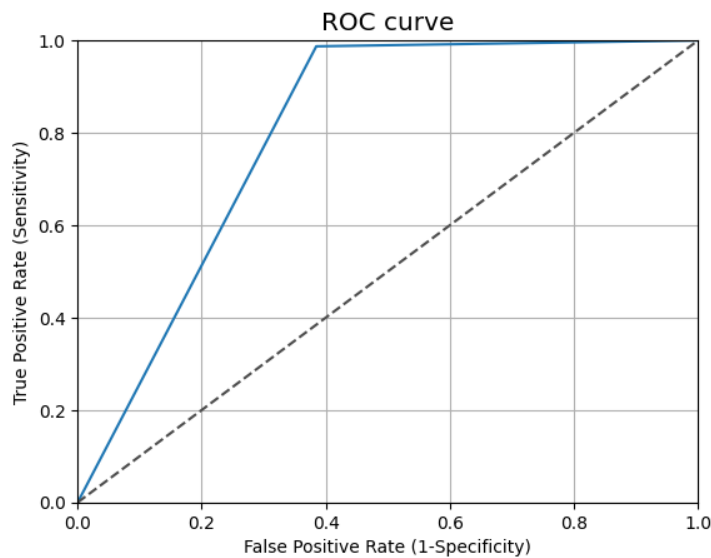
Εικόνα 89 Confusion Matrix για ταξινομητή (ResNet34) με χρήση συνθετικών εικόνων (cGAN 64 εποχές Adam Optimizer)



Εικόνα 90 Καμπύλη ROC(ResNet34)



Εικόνα 91 Confusion Matrix για ταξινομητή (DenseNet-121) με χρήση συνθετικών εικόνων (cGAN 64 εποχές Adam Optimizer)



Εικόνα 92 Καμπύλη ROC(DenseNet-121)

#### 4.4.3 Αποτελέσματα με χρήση εικόνων από cGAN(5 εποχές 128x128 μέγεθος)

Το τρίτο πείραμα πραγματοποιήθηκε με τη βοήθεια 4 προεκπαιδευμένων μοντέλων και αφορά την εκπαίδευση του ταξινομητή με την προσθήκη συνθετικών δειγμάτων που παρήχθησαν από το cGAN(128x128). Ο Πίνακας 18 και η Εικόνα 93 παρουσιάζουν τα αποτελέσματα του πειράματος ανά προεκπαιδευμένο μοντέλο και ανά μετρική η οποία εξετάστηκε.

Ξεκινώντας από τη μετρική precision το ResNet-18 προηγείται με τιμή (0,8705) έχοντας το DenseNet-121 να ακολουθεί από κοντά με ακρίβεια 0,8475. Τα



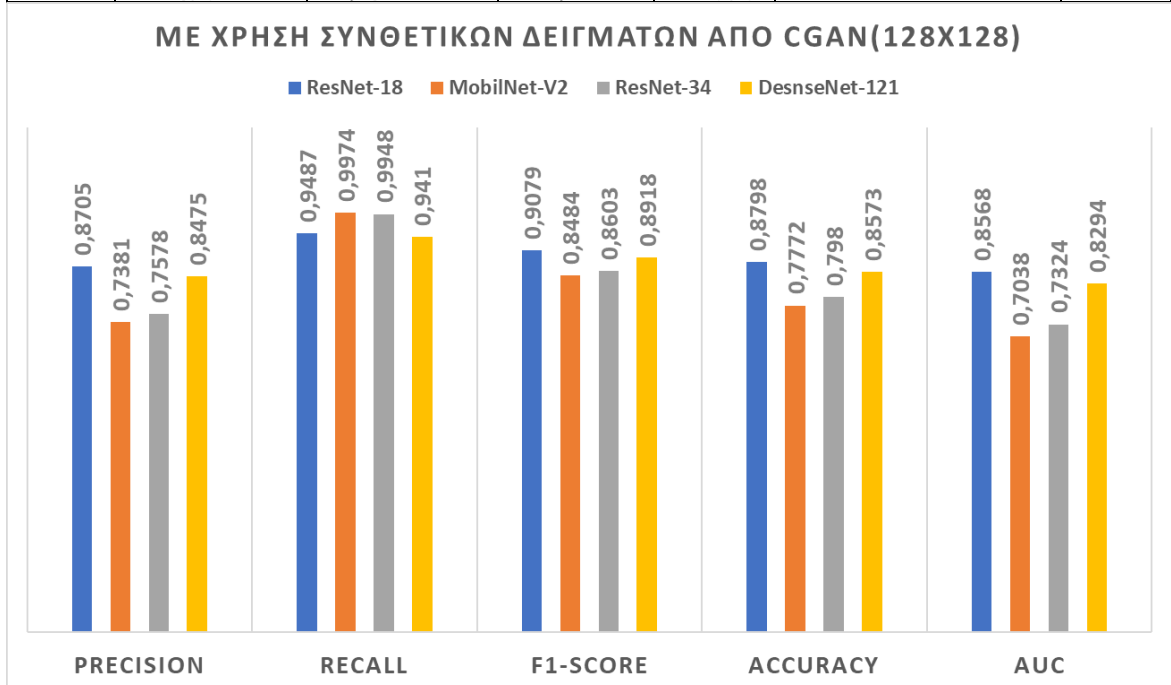
ResNet-34 και το MobileNet\_v2 παρουσιάζουν συγκριτικά χαμηλότερες τιμές ακρίβειας (0,7578 και 0,7381, αντίστοιχα). Στη μετρική Recall το MobileNet\_V2 εμφανίζει την υψηλότερη τιμή (0,9974), καταφέροντας να αναγνωρίσει σχεδόν όλες τις πραγματικές θετικές περιπτώσεις. Εξίσου πολύ υψηλή τιμή πετυχαίνει και το ResNet-34 (0,9948) ακολουθούμενο από τα μοντέλα ResNet-18 και DenseNet-121 με τιμές (0,9487) και (0,9410) αντίστοιχα που πετυχαίνουν σχετικά μικρότερες τιμές σε σχέση με τα προηγούμενα δύο. Όσον αφορά τη μετρική F1-Score, το ResNet-18 επιτυγχάνει την υψηλότερη τιμή (0,9079), υποδεικνύοντας μια ισορροπημένη απόδοση σε precision και recall. Το DenseNet-121 ακολουθεί με F1-Score 0,8918, παρουσιάζοντας παρόμοια ισορροπία, αλλά ελαφρώς χαμηλότερη από το ResNet-18, ενώ τα MobileNet\_v2 και ResNet-34 παρουσιάζουν συγκριτικά χαμηλότερες τιμές F1-Score (0,8484 και 0,8603, αντίστοιχα).

Σχετικά με το Accuracy το ResNet-18 προηγείται με τιμή (0,8798), ακολουθούμενο από το DenseNet-121 (0,8573), ενώ ελαφρώς χαμηλότερες τιμές παρουσιάζουν τα μοντέλα ResNet-34 και MobileNet\_v2 (0,7980 και 0,7772, αντίστοιχα). Επίσης άλλη μια πρωτιά κατακτά το ResNet-18 όσον αφορά τη μετρική AUC με τιμή (0,8568) παρουσιάζοντας έτσι μια ισχυρή διακριτική ικανότητα μεταξύ των θετικών και αρνητικών περιπτώσεων. Κοντά του είναι και το DenseNet-121 με τιμή 0,8294, ενώ πιο χαμηλές τιμές πετυχαίνουν τα μοντέλα ResNet-34 και MobileNet\_v2 (0,7324 και 0,7038, αντίστοιχα).

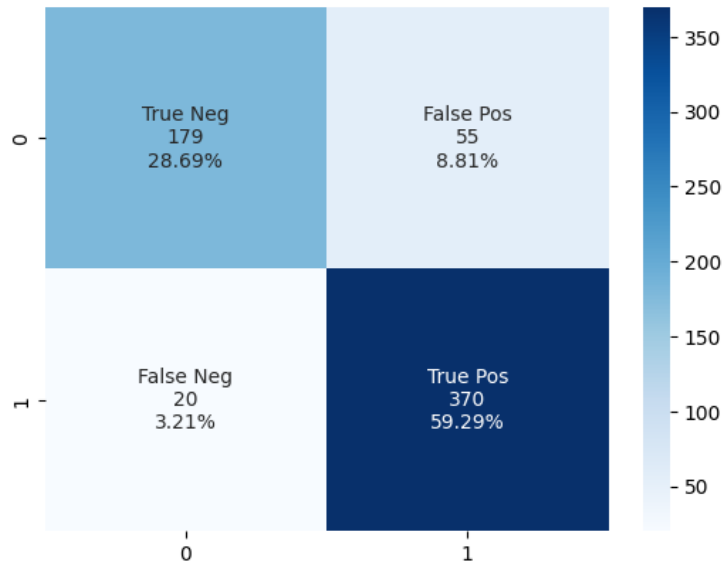
Τέλος, σχετικά με τους χρόνους εκπαίδευσης το ResNet-18 χρειάστηκε τον λιγότερο χρόνο (23 λεπτά και 55 δευτερόλεπτα) έχοντας μικρή διαφορά από το MobileNet\_v2 που χρειάστηκε μόλις 25 λεπτά και 41 δευτερόλεπτα. Το ResNet-34 χρειάστηκε 31 λεπτά και 56 δευτερόλεπτα, ενώ τον περισσότερο χρόνο χρειάστηκε το DenseNet-121 (46 λεπτά και 1 δευτερόλεπτο).

Πίνακας 18 Αποτελέσματα Classifier με τη χρήση συνθετικών δειγμάτων από το cGAN (5 εποχές 128x128 μέγεθος)

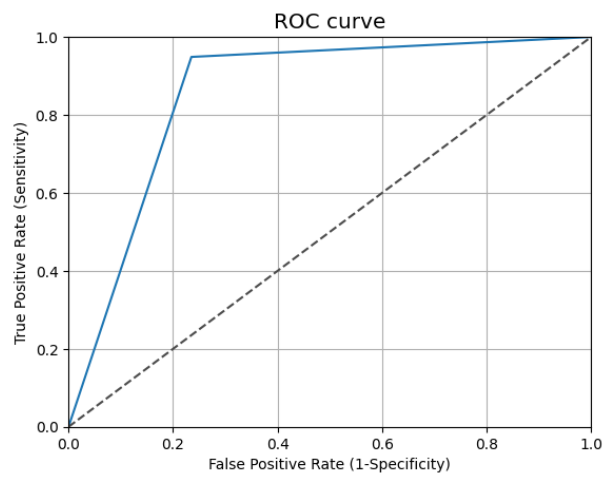
	Precision	Recall	F1-Score	Accuracy	AUC	Training Time(minutes)	Environment
<b>ResNet-18</b>	0.8705	0.9487	0.9079	0.8798	0.8568	23 min 55 sec	Local host
<b>MobilNet-V2</b>	0.7381	0.9974	0.8484	0.7772	0.7038	25 min 41 sec	Local host
<b>ResNet-34</b>	0.7578	0.9948	0.8603	0.7980	0.7324	31 min 56 sec	Local host
<b>DesnseNet-121</b>	0.8475	0.9410	0.8918	0.8573	0.8294	46 min 1 sec	Colab



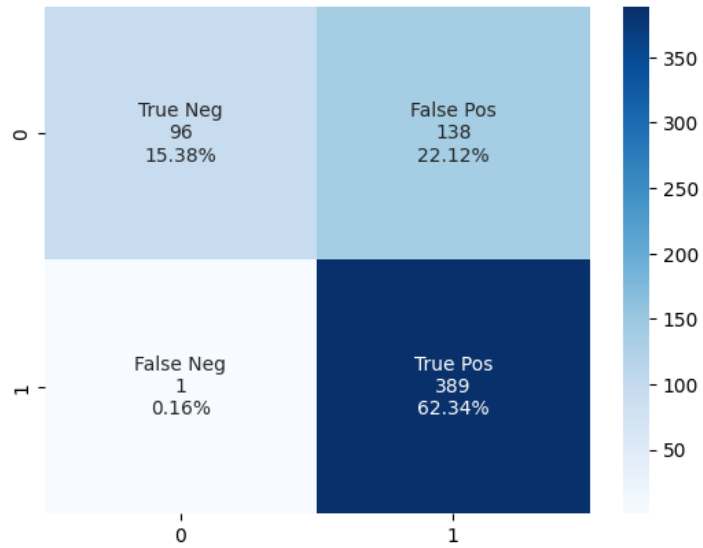
Εικόνα 93 Οπτικοποίηση αποτελεσμάτων Classifier με τη χρήση συνθετικών δειγμάτων από το cGAN (128x128)



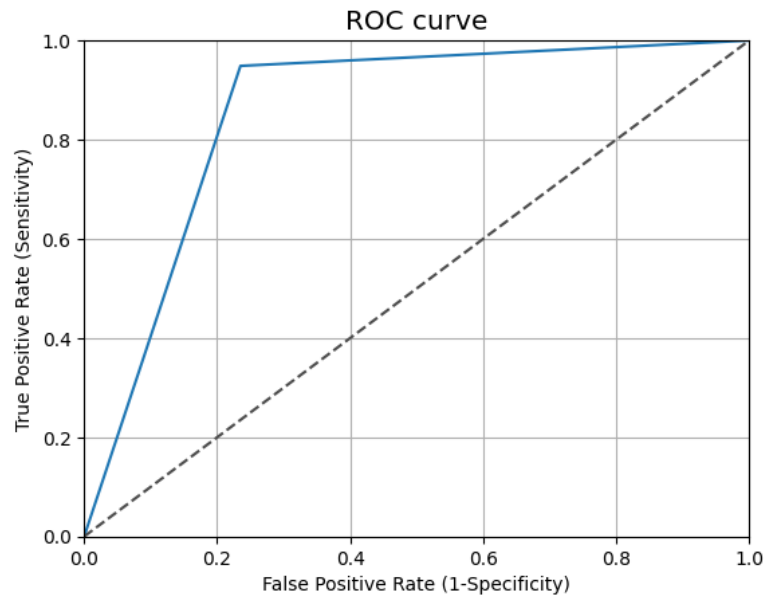
Εικόνα 94 Confusion Matrix για ταξινομητή(ResNet18) με χρήση συνθετικών εικόνων (cGAN 64 εποχές 128x128 μέγεθος)



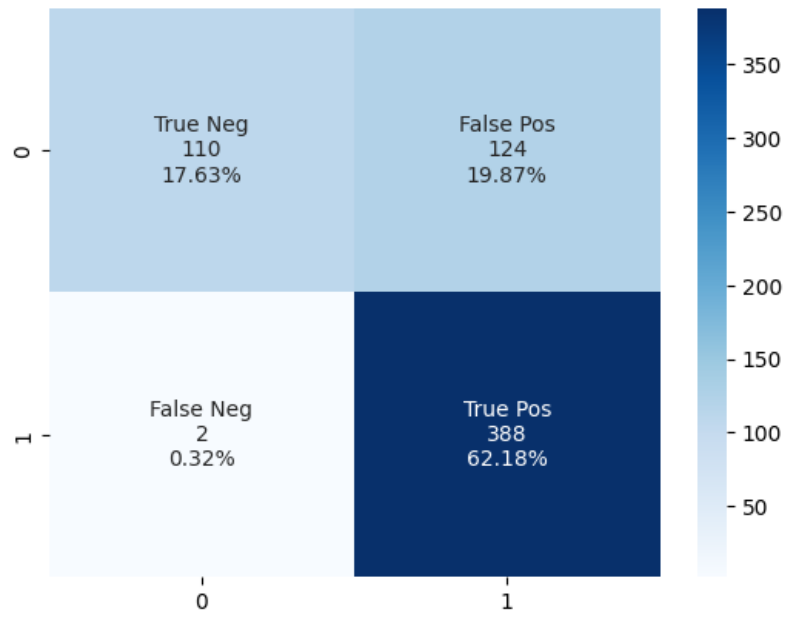
Εικόνα 95 Καμπύλη ROC (ResNet18)



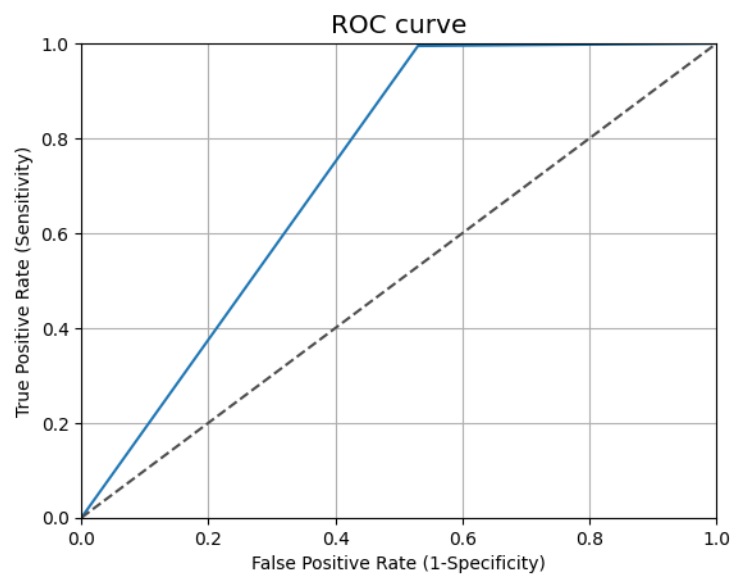
Εικόνα 96 Confusion matrix MobileNet\_V2



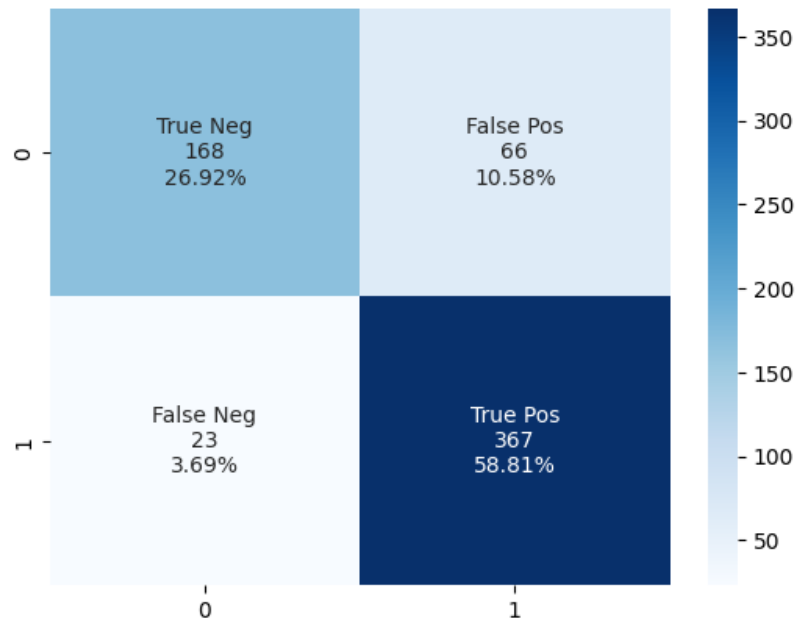
Εικόνα 97 Καμπύλη ROC (MobileNet\_V2)



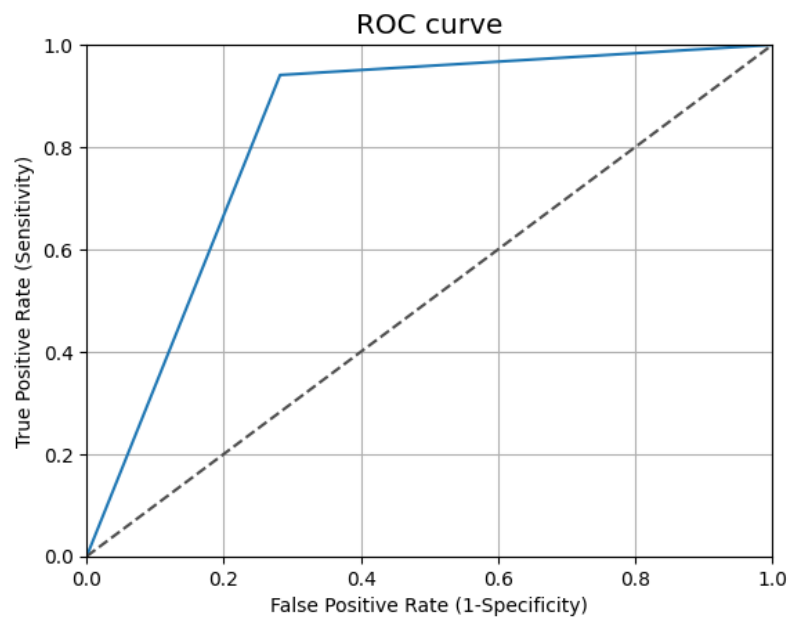
Εικόνα 98 Confusion matrix (ResNet-34)



Εικόνα 99 Καμπύλη ROC (ResNet-34)



Εικόνα 100 Confusion Matrix (DenseNet-121)



Εικόνα 101 Καμπύλη ROC (DenseNet-121)

#### 4.5 Εξομοίωση της μεθοδολογίας άλλων ερευνητών

Τέλος, στη συγκεκριμένη ενότητα επιχειρήθηκε μια προσπάθεια μεθοδολογικής αναπαραγωγής και συγκριτικής ανάλυσης. Πρωταρχικός στόχος ήταν η αναπαραγωγή της μεθοδολογίας που χρησιμοποίησαν οι συγγραφείς στη σχετική δημοσίευση [53], τηρώντας όσο το δυνατόν περισσότερο, την πειραματική τους διάταξη. Επόμενος στόχος ήταν η σύγκριση του αποτελέσματος του δικού μας πειράματος με το δικό τους.

Για να διατηρηθεί μια ακριβοδίκαιη και συνεπής σύγκριση, το μοντέλο cGAN μας υποβλήθηκε σε εκπαίδευση που εκτείνεται στις 600 εποχές. Αυτή η διάρκεια ευθυγραμμίζεται με τη δημοσίευση [53], παρέχοντας μια ισχυρή βάση για την αξιολόγηση της απόδοσης του ταξινομητή μας σε σχέση με το μοντέλο των συγγραφέων.

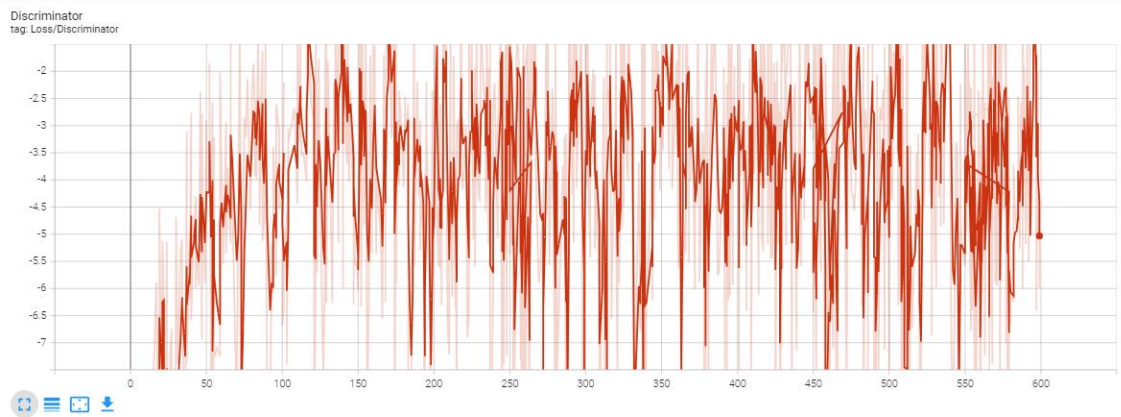
Η εκπαίδευση έγινε χρησιμοποιώντας τις υπερπαραμέτρους που χρησιμοποιήσαμε και προηγουμένως (Πίνακας 7) έχοντας ως μοναδική διαφορά τον αριθμό των εποχών (600). Για λόγους ευκολίας αναφέρονται συνοπτικά και παρακάτω (Πίνακας 19). Για την υλοποίηση του συγκεκριμένου πειράματος ήταν αναγκαία η χρήση πόρων GPU στο Google Colab. Η διαδικασία ήταν αρκετά απαιτητική και ολοκληρώθηκε σε διάστημα 6 ημερών.

Πίνακας 19 Υπερπαραμέτροι εκπαίδευσης cGAN

<b>Υπερπαραμέτρος</b>	<b>Τμή</b>
Device	Cuda(GPU)
Learning Rate	0.0001
Batch Size	16
Image Size	64
Number of Channels	3
Number of Epochs	600
LAMDA_GP	10

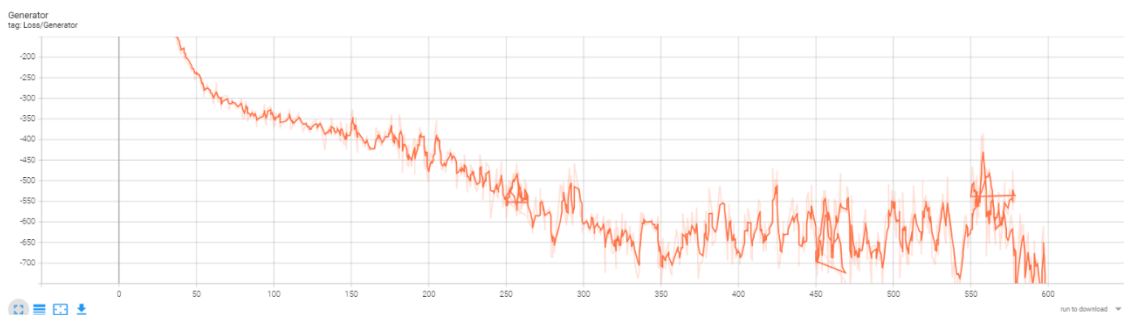
#### Διαδικασία επιλογής μοντέλου

Στη συνέχεια, με τη βοήθεια του Tensorboard, οπτικοποιήθηκε σε ένα γράφημα η απώλεια (loss) του Discriminator-Critic και του Generator. Στο γράφημα που αποτυπώνεται στην Εικόνα 102, ο Discriminator προσπαθεί να μεγιστοποιήσει την ικανότητά του να διακρίνει μεταξύ των παραγόμενων δεδομένων και των πραγματικών δεδομένων. Αυτό συμβαίνει από την αρχή (εποχή 0) μέχρι περίπου την εποχή 100-150. Στη συνέχεια, παρατηρείται ότι η κυματομορφή παρουσιάζει μεγάλα σκαμπανεβάσματα, πράγμα που σημαίνει ότι δεν συνεχίζει να βελτιώνεται και παραμένει στάσιμος.



Εικόνα 102 Η απώλεια του Discriminator και την εκπαίδευση 600 εποχών

Την ίδια στιγμή, ο Generator στην Εικόνα 103 προσπαθεί να ελαχιστοποιήσει την απόσταση μεταξύ των παραγόμενων δεδομένων της και των πραγματικών. Το διάστημα που δείχνει ότι ο Generator βελτιώνει την απόδοσή του ομαλά είναι από την εποχή 0 μέχρι την εποχή 150-250. Από εκεί και πέρα, η γραμμή παρουσιάζει έντονες διακυμάνσεις πράγμα που φαίνεται ότι δε βοηθάει την εκπαίδευση περισσότερο.



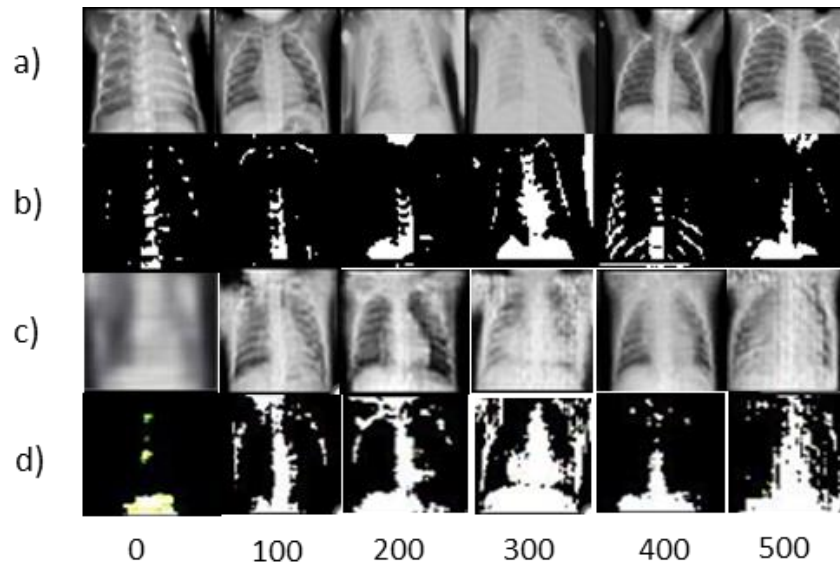
Εικόνα 103 Η απώλεια του Generator κατά την εκπαίδευση 600 εποχών

Στη συνέχεια τυπώθηκαν κάποια δείγματα παραγόμενων εικόνων κάθε 100 εποχές, ώστε να διαπιστωθεί και στην πράξη ποιο είναι το καλύτερο μοντέλο που πρέπει να παραμείνει για την παραγωγή συνθετικών δειγμάτων με σκοπό τη βελτιστοποίηση του Classifier.

Στην Εικόνα 104 παρατηρείται πως εξελίσσεται η παραγωγή δειγμάτων κατά την εκπαίδευση του cGAN. Αυτό που προκύπτει είναι ότι οι εικόνες από την εποχή 100 και 200 είναι πιο ευδιάκριτες και καθαρές σε σχέση με τις υπόλοιπες που ακολουθούν. Για να μελετηθούν οι αλλαγές με σαφήνεια από τη μία εποχή στην άλλη, έγινε μετατροπή στις εικόνες σε δυαδική εικόνα με κατώφλι (thresholding), ώστε να διαχωριστεί η δομή του πνεύμονα από τις λευκές θολερότητες. Έτσι είναι εφικτό να δει κανείς ότι έχουν σχηματιστεί τα κόκκαλα



στον θώρακα, αλλά και το μεσαίο κόκαλο της σπονδυλικής στήλης. Στις υπόλοιπες, ο θόρυβος είναι πολύ πιο αισθητός. Γι' αυτό το μοντέλο που επιλέχθηκε να χρησιμοποιηθεί είναι αυτό που έχει εκπαιδευτεί για 200 εποχές.

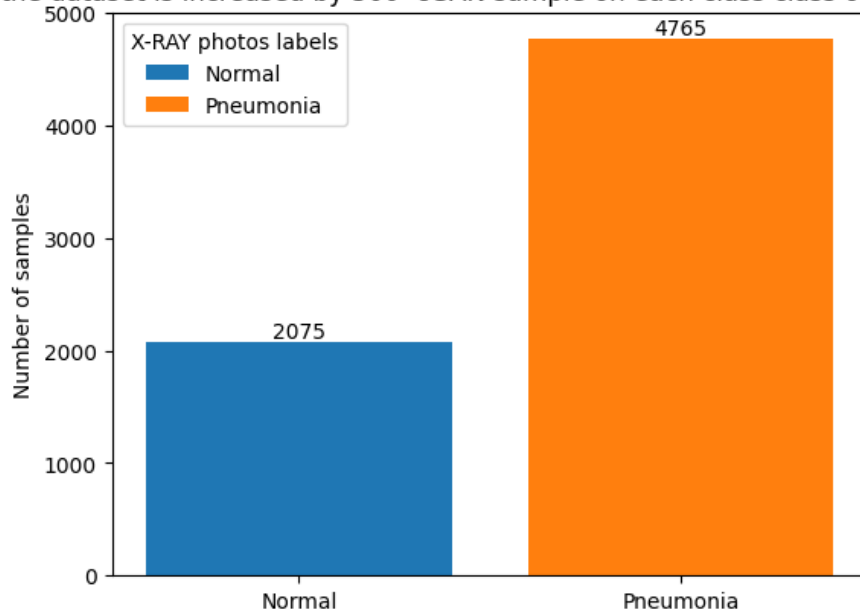


Εικόνα 104 a) Δείγματα εικόνων από το αληθινό dataset b) Thresholding των αληθινών δειγμάτων αυτών c) Συνθετικά δείγματα από cGAN ανά 100 εποχές d) Thresholding των συνθετικών δειγμάτων ανά 100 εποχές.

#### Προεπεξεργασία Dataset και επαύξηση δεδομένων με δείγματα από cGAN(200 εποχών)

Αφού επιλέχθηκε το μοντέλο, δημιουργήθηκαν με τη βοήθεια του cGAN (200 εποχών) 500 νέα δείγματα για κάθε κλάση ( $k=500$ ), όπως ακριβώς έκαναν και οι ερευνητές στη δημοσίευση [53]. Τα νέα συνθετικά δείγματα προστέθηκαν στο αρχικό dataset έχοντας πλέον την παρακάτω κατανομή ανά κλάση.

Now the dataset is increased by 500 cGAN sample on each class class of Xray Dataset



Εικόνα 105 Chest xray dataset μετά την προσθήκη 500 συνθετικών δειγμάτων σε κάθε κλάση

Στη συνέχεια, εκπαιδεύτηκε ο Classifier για άλλη μια φορά χρησιμοποιώντας τις παρακάτω υπερπαραμέτρους (Πίνακας 20). Οι συγκεκριμένες υπερπαραμέτροι χρησιμοποιήθηκαν και από τους ερευνητές στην εκπαίδευση του δικού τους Classifier χαρακτηρίζοντας τους ως τους καλύτερους δυνατούς.

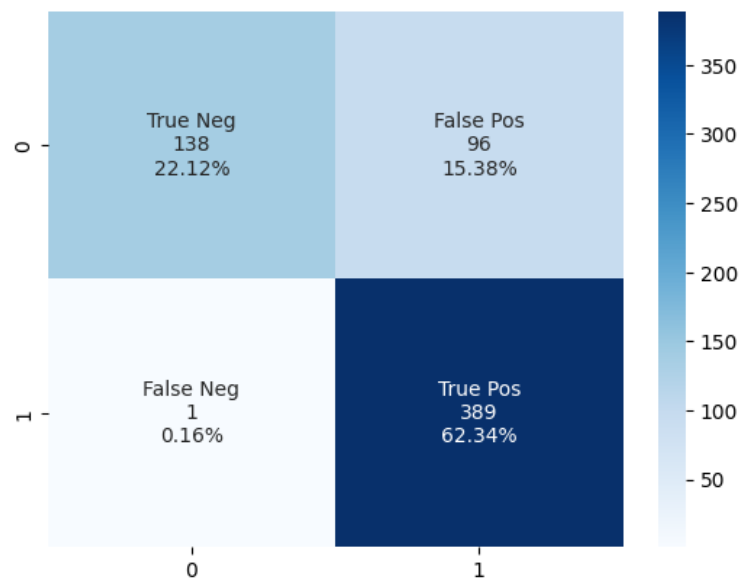
Πίνακας 20 Υπερπαραμέτροι εκπαίδευσης Classifier

Παράμετρος	Τιμή
Batch size	16
Learning rate	0.0001
Epochs	50
Criterion	Cross-entropy
Optimizer	Adam

#### 4.5.1 Αποτελέσματα Classifier με χρήση εικόνων από cGAN (200 εποχές)

Πίνακας 21 Αποτελέσματα Classifier με χρήση συνθετικών δειγμάτων από cGAN (200 εποχές)

	<b>Precision</b>	<b>Recall</b>	<b>F1-Score</b>	<b>Accuracy</b>	<b>Training Time(minutes)</b>	<b>Environment</b>
<b>ResNet-18</b>	0.8020	0.9974	0.8891	0.8445	107 min 20 sec	Colab



Εικόνα 106 Confusion Matrix για ταξινομητή με χρήση συνθετικών εικόνων (cGAN 200 εποχές Adam Optimizer)

#### 4.5.2 Αποτελέσματα Classifier δημοσίευσης με χρήση συνθετικών δειγμάτων από ACGAN

Πίνακας 22 Αποτελέσματα Classifier δημοσίευσης [53] με χρήση συνθετικών δειγμάτων από ACGAN

	<b>Precision</b>	<b>Recall</b>	<b>F1-Score</b>	<b>Accuracy</b>	<b>Training Time(minutes)</b>	<b>Environment</b>
<b>ResNet-18</b>	0.9128	0.9124	0.9120	0.9124	25 min 14 sec	-

## **5 ΚΕΦΑΛΑΙΟ 5: ΣΥΖΗΤΗΣΗ-ΣΧΟΛΙΑΣΜΟΣ ΕΠΙ ΤΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ**

### **5.1 Συγκριτικά αποτελέσματα μεταξύ των παραπάνω μοντέλων που κάνουν χρήση συνθετικών δειγμάτων από το cGAN**

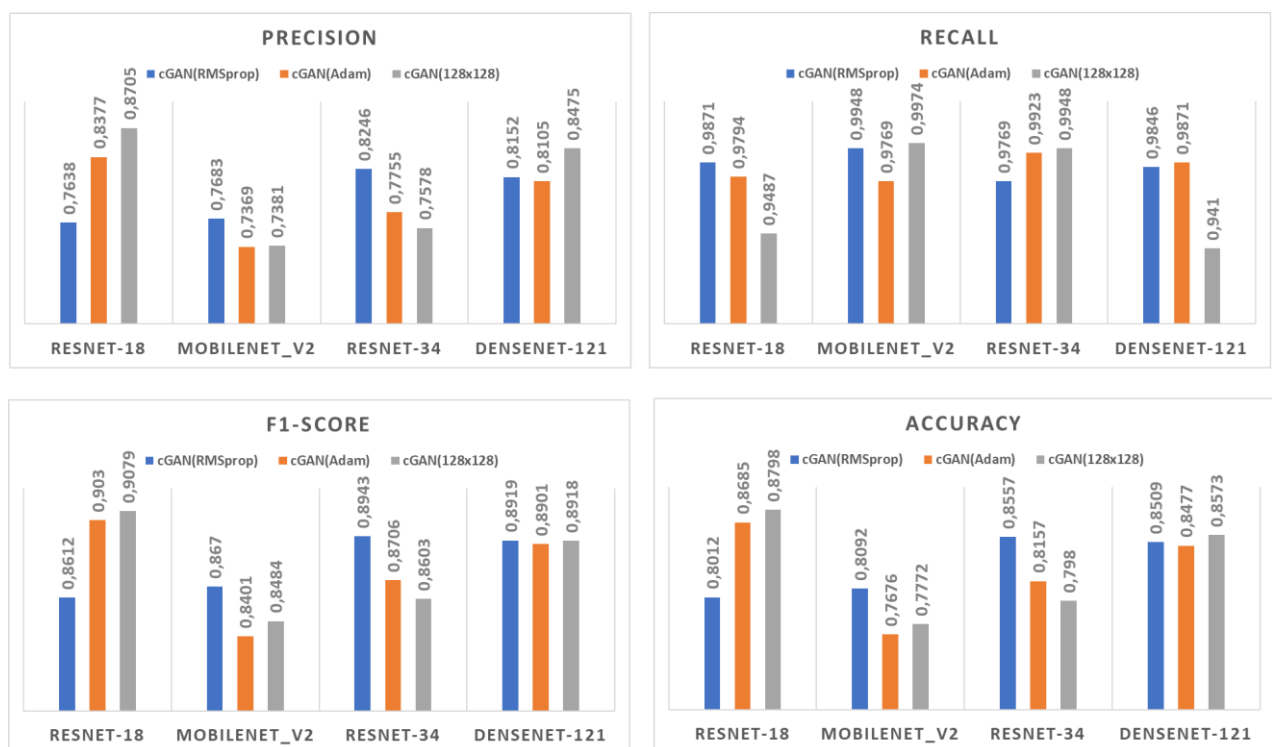
Συγκρίνοντας τα αποτελέσματα των παραπάνω μοντέλων που εκπαιδεύτηκαν με την προσθήκη συνθετικών δειγμάτων από τα τρία διαφορετικά cGAN και με βάση το προεκπαιδευμένο μοντέλο που χρησιμοποιήθηκε για την εκπαίδευση του ταξινομητή, διαπιστώνεται ότι όσο αφορά το Precision στα περισσότερα μοντέλα που εκπαιδεύτηκαν τα δείγματα από το cGAN (128x128) φαίνεται ότι παρουσιάζουν την υψηλότερη τιμή σε σύγκριση με τα δείγματα από τα cGAN (RMSProp) και cGAN(Adam). Ειδικότερα, ο Classifier που εκπαιδεύτηκε με το προεκπαιδευμένο μοντέλο ResNet-18 και δείγματα από το cGAN (128x128) επιτυγχάνει την υψηλότερη τιμή (0,8705) σε σύγκριση με τα αντίστοιχα δείγματα που χρησιμοποιήθηκαν από το cGAN (RMSprop) και cGAN (Adam) που πέτυχαν τιμές 0,7638 και 0,8377 αντίστοιχα υποδεικνύοντας έτσι ότι έχει τα λιγότερα ψευδώς θετικά (FP) αποτελέσματα. Επίσης, τα δείγματα από το cGAN (128x128) σε συνδυασμό με το προεκπαιδευμένο δίκτυο DenseNet-121 πετυχαίνουν εξίσου υψηλή τιμή (0,8475), με τα δείγματα από τα cGAN (RMSprop) και cGAN(Adam) να ακολουθούν με παρόμοιες τιμές 0,8152 και 0,8105 αντίστοιχα.

Συνεχίζοντας τη σύγκριση με βάση τη μετρική Recall, τα δείγματα από το cGAN (128x128) σε συνδυασμό με το προεκπαιδευμένο μοντέλο MobileNet\_v2 σημειώνουν την υψηλότερη τιμή (0,9974) υποδεικνύοντας έτσι ότι το μοντέλο έχει τα λιγότερα ψευδώς αρνητικά (FN) αποτελέσματα σε σχέση με τα υπόλοιπα. Ωστόσο, όπως φαίνεται και στο παρακάτω γράφημα (Εικόνα 107) σε όλα τα πειράματα το recall με οποιαδήποτε συνθετικά δείγματα και οποιοδήποτε προεκπαιδευμένο μοντέλο και αν χρησιμοποιήθηκε πετυχαίνει τιμή πάνω από 0,94.

Σχετικά με το F1-Score τα δείγματα που χρησιμοποιήθηκαν από το cGAN (128x128) σε συνδυασμό με το προεκπαιδευμένο μοντέλο ResNet-18 επιδεικνύουν την υψηλότερη τιμή 0,9079 σε σύγκριση με τα αντίστοιχα μοντέλα που χρησιμοποίησαν δείγματα από τα cGAN (RMSProp) και cGAN (Adam) που

επιτυγχάνουν 0,8612 και 0,903 αντίστοιχα. Το εύρημα αυτό υποδηλώνει ότι επιτυγχάνεται καλή ισορροπία μεταξύ των μετρικών Precision και Recall. Παρόμοιες τάσεις παρατηρούνται και στα μοντέλα ResNet-34 και DenseNet-121, ενώ οι χαμηλότερες τιμές στο F1-Score παρατηρούνται όταν ο Classifier εκπαιδεύτηκε με τη βοήθεια του MobileNet\_v2.

Στη συνέχεια, με βάση τη μετρική Accuracy παρατηρούμε ότι το μοντέλο ResNet-18 σε συνδυασμό με τα συνθετικά δείγματα που παράχθηκαν από το cGAN(128x128) παρουσιάζει την υψηλότερη τιμή (0,8798) και συνεπώς το υψηλότερο ποσοστό σωστών προβλέψεων. Αμέσως μετά ακολουθεί πάλι το ResNet-18 με συνθετικά δείγματα από το cGAN(Adam), ενώ τις πιο χαμηλές τιμές στο Accuracy τις σημειώνει το μοντέλο MobileNet\_V2 με τη χρήση συνθετικών δειγμάτων από το cGAN(Adam) με τιμή (0,7676).



Εικόνα 107 Σύγκριση αποτελεσμάτων μεταξύ όλων των παραπάνω

Από όλα τα παραπάνω, διαπιστώνουμε ότι τα συνθετικά δείγματα που χρησιμοποιήσαμε για κάθε ένα από τα πειράματα μας, οδηγούν σε διαφορετικά αποτελέσματα όσο αφορά την απόδοση του ταξινομητή μας. Εξίσου όμως μεγάλη επίδραση έχει και το προεκπαιδευμένο μοντέλο που χρησιμοποιήσαμε σε κάθε πείραμα μας για την εκπαίδευση. Μελετώντας τα αποτελέσματα που

απεικονίζονται στα παραπάνω γραφήματα (Εικόνα 107), διαπιστώνεται ότι το ResNet-18 έχει σταθερά καλές επιδόσεις σε πολλαπλές μετρικές, παρουσιάζοντας συχνά τις υψηλότερες τιμές. Αντίθετα, το MobileNet\_v2, παρά τις εξαιρετικές επιδόσεις του στο recall, παρουσιάζει χαμηλά αποτελέσματα σε όλες τις υπόλοιπες μετρικές (Precision, F1-Score, Accuracy). Το ResNet-34 πετυχαίνει και αυτό αξιοσημείωτα αποτελέσματα στο recall, αλλά υστερεί ελαφρώς σε precision σε σύγκριση με το ResNet-18 και το DenseNet-121. Τέλος, το DenseNet-121 επιδεικνύει συνέπεια αλλά συχνά υπολείπεται του ResNet-18 σε precision και accuracy ανάλογα με τα συνθετικά δείγματα που χρησιμοποιήθηκαν.

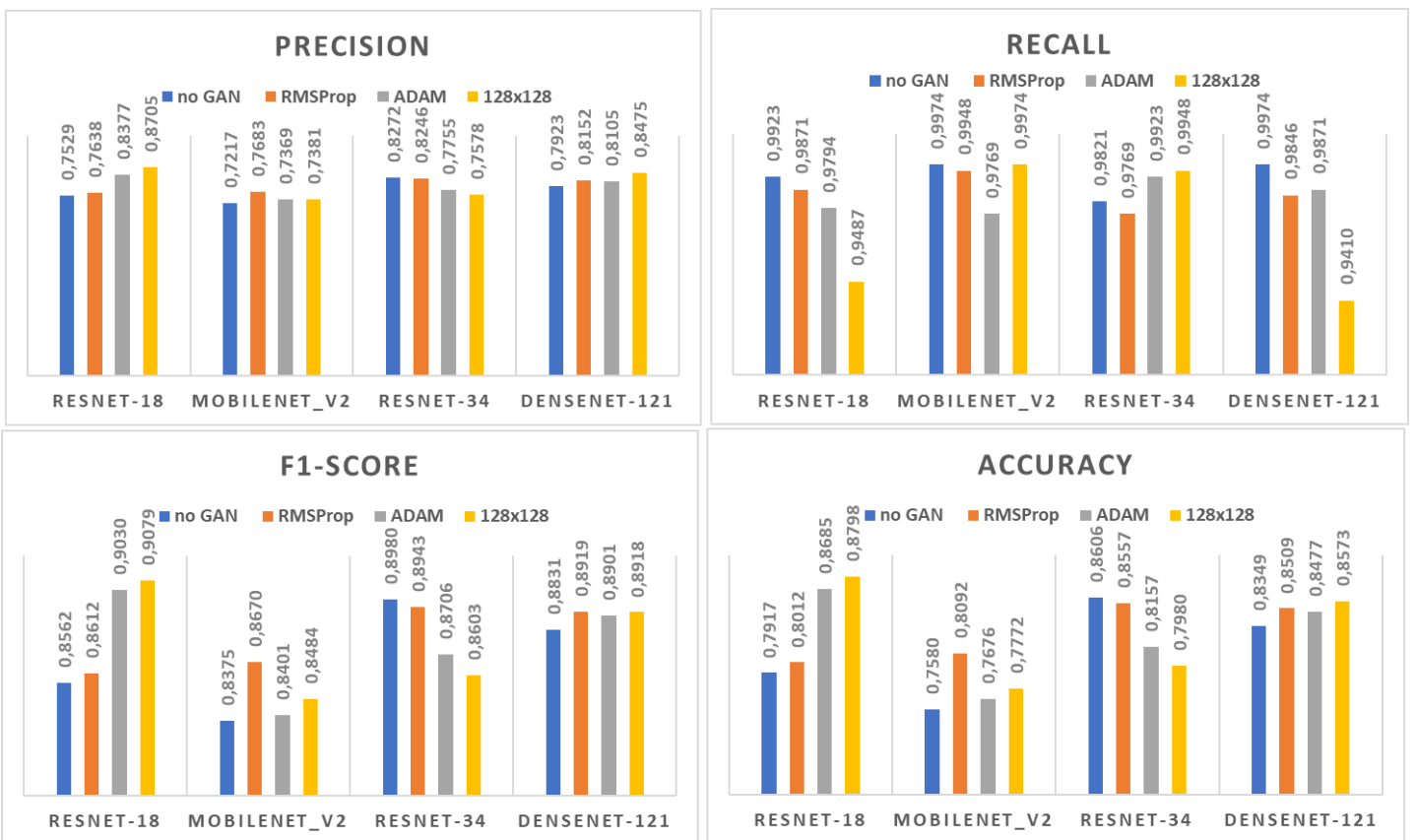
Συνοψίζοντας, τα δείγματα που παράχθηκαν από το cGAN(128x128) οδηγούν γενικά σε βελτιωμένες επιδόσεις σε πολλαπλές μετρικές, ενισχύοντας ιδιαίτερα το precision και το accuracy για διάφορα μοντέλα. Όσον αφορά τα προ-εκπαιδευμένα μοντέλα, το ResNet-18 επιδεικνύει ισχυρή απόδοση με όποια συνθετικά δείγματα και αν χρησιμοποιήθηκαν αναδεικνύοντας την προσαρμοστικότητα και την αποτελεσματικότητά του σε διάφορα σενάρια.

Κλείνοντας και λαμβάνοντας υπόψη τον αντίκτυπο των δειγμάτων cGAN στην απόδοση του ταξινομητή, τα συνθετικά δείγματα από το cGAN(128x128) τείνουν να έχουν καλύτερο αντίκτυπο στην ενίσχυση των μετρικών precision, F1-Score, accuracy, που είναι κρίσιμοι δείκτες απόδοσης για έναν ταξινομητή σε σύγκριση με τα RMSprop και Adam. Επιπλέον, το ResNet-18 αναδεικνύεται ως ένα ευέλικτο προ-εκπαιδευμένο μοντέλο που έχει σταθερά καλή απόδοση σε διαφορετικά δείγματα cGAN και διάφορες μετρικές απόδοσης.

Είναι σημαντικό να σημειωθεί ότι η αξιολόγηση των επιδόσεων ενός μοντέλου δεν βασίζεται αποκλειστικά σε ένα μόνο σύνολο μετρήσεων. Άλλοι παράγοντες, όπως η συγκεκριμένη περίπτωση χρήσης, οι υπολογιστικοί πόροι, η ερμηνευσιμότητα και το κόστος των ψευδώς θετικών και ψευδώς αρνητικών αποτελεσμάτων, θα πρέπει επίσης να λαμβάνονται υπόψη κατά την επιλογή του καλύτερου μοντέλου.

## 5.2 Συγκριτικά αποτελέσματα μεταξύ των μοντέλων που κάνουν χρήση συνθετικών δειγμάτων από το cGAN με τον αρχικό Classifier

Στη συνέχεια συγκρίθηκε η απόδοση του μοντέλου ως προς τις μετρικές που αναφέρθηκαν παραπάνω συγκρίνοντας τα αποτελέσματα του αρχικού μοντέλου στο οποίο δε χρησιμοποιήθηκαν συνθετικές εικόνες (Πίνακας 13) σε σχέση με τα μοντέλα που χρησιμοποιήθηκαν συνθετικές εικόνες παραγόμενες από τα παραπάνω cGAN. Σκοπό αποτέλεσε η μελέτη του κατά πόσο η χρήση του cGAN συνέβαλε στη βελτιστοποίηση του ταξινομητή ως προς τη μειωτική κλάση (Normal).



Εικόνα 108 Αποτελέσματα Classifier με χρήση cGAN εναντίον Classifier χωρίς χρήση cGAN

Πίνακας 23 Πίνακας μεταβολών ανά μετρική και προεκπαιδευμένο μοντέλο των ταξινομητών που εκπαιδεύτηκαν με συνθετικά δείγματα από τα cGANs σε σχέση με το αρχικό ταξινομητή που δεν χρησιμοποιεί συνθετικά δείγματα

Metric	Pretrained Model	No GAN	$\Delta$ (RMSProp)	$\Delta$ (ADAM)	$\Delta$ (128x128)
Precision	ResNet-18	0,7529	+0,0109	+0,0848	+0,1176
	MobileNetV2	0,7217	+0,0466	+0,0152	+0,0164
	ResNet-34	0,8272	-0,0026	-0,0517	-0,0694
	DenseNet-121	0,7923	+0,0229	+0,0182	+0,0552
Recall	ResNet-18	0,9923	-0,0052	-0,0129	-0,0436
	MobileNetV2	0,9974	-0,0026	-0,0205	0,0000
	ResNet-34	0,9821	-0,0052	+0,0102	+0,0127
	DenseNet-121	0,9974	-0,0128	-0,0103	-0,0564
F1-Score	ResNet-18	0,8562	+0,0050	+0,0468	+0,0517
	MobileNetV2	0,8375	+0,0295	+0,0026	+0,0109
	ResNet-34	0,8980	-0,0037	-0,0274	-0,0377
	DenseNet-121	0,8831	+0,0088	+0,0070	+0,0087
Accuracy	ResNet-18	0,7917	+0,0095	+0,0768	+0,0881
	MobileNetV2	0,7580	+0,0512	+0,0096	+0,0192
	ResNet-34	0,8606	-0,0049	-0,0449	-0,0626
	DenseNet-121	0,8349	+0,0160	+0,0128	+0,0224

Σε αυτή τη συνολική ανάλυση της απόδοσης ταξινόμησης εικόνων, διερευνήσαμε τον αντίκτυπο της ενσωμάτωσης των Conditional Generative Adversarial Networks (cGAN) με τρεις διαφορετικές διαμορφώσεις – cGAN(RMSProp), cGAN(ADAM) και cGAN(128x128) σε τέσσερα προ-εκπαιδευμένα μοντέλα: ResNet-18, MobileNetV2, ResNet-34 και DenseNet-121. Οι μετρικές αξιολόγησης που χρησιμοποιήθηκαν ήταν οι Precision, Recall, F1-Score και Accuracy. Η κεντρική εστίαση ήταν η σύγκριση των αποτελεσμάτων που επιτεύχθηκαν με τη χρήση συνθετικών δειγμάτων cGANs με τα αποτελέσματα χωρίς τη χρήση συνθετικών δειγμάτων (No GAN).

Ένας από τους πρωταρχικούς στόχους ήταν η κατανόηση των αυξητικών ή μειωτικών επιδράσεων των cGANs στις μετρικές αξιολόγησης όταν εφαρμόζονται σε διάφορα προ-εκπαιδευμένα μοντέλα. Για να διευκολυνθεί αυτή η σύγκριση, υπολογίστηκαν οι διαφορές ( $\Delta$ ) στις μετρικές επιδόσεων μεταξύ κάθε διαμόρφωσης cGAN και του αρχικού ταξινομητή (Πίνακας 23).

Αρχικά όσον αφορά τη μετρική Precision, η χρήση cGAN είχε διαφοροποιημένες επιδράσεις. Το μοντέλο που χρησιμοποίησε δείγματα παραγόμενα από το



cGAN(128x128) βελτίωσε σταθερά το precision κυρίως σε συνδυασμό με το ResNet-18, καθώς και τα μοντέλα με τη χρήση συνθετικών δειγμάτων από το cGAN(RMSprop) και το cGAN(Adam). Ωστόσο για το προεκπαιδευμένο μοντέλο ResNet-34 υπήρξε μείωση της απόδοσης ως προς το precision για όλα τα συνθετικά δείγματα που συνδικάστηκαν με αυτά μοντέλα. Αυτό το αποτέλεσμα υποδεικνύει ότι η χρήση δειγμάτων παραγόμενα από cGAN επηρεάζει το precision ανάλογα με το προεκπαιδευμένο μοντέλο, τονίζοντας την ανάγκη για εξατομικευμένη επιλογή.

Συνεχίζοντας με το Recall τα περισσότερα μοντέλα παρουσίασαν πτώση της απόδοσης τους με οποιαδήποτε συνθετικά δείγματα και αν προστέθηκαν με μόνη εξαίρεση το μοντέλο ResNet-34 όπου η χρήση συνθετικών δειγμάτων από CGAN(Adam) και CGAN(128x128) βελτίωσε την απόδοση του. Ωστόσο, ορισμένα προ-εκπαιδευμένα μοντέλα, όπως το MobileNetV2, παρουσίασαν ισχυρό recall ακόμη και χωρίς τη χρήση συνθετικών δειγμάτων (no GAN). Αυτό υποδηλώνει ότι η χρήση παραγόμενων συνθετικών δειγμάτων από cGANs μπορεί να έχει σημαντικότερο αντίκτυπο σε μοντέλα με αρχικά χαμηλότερο recall.

Ακολουθώντας σχετικά με το F1-score τα μοντέλα που χρησιμοποιούν δείγματα από το cGAN(128x128) παρουσίασαν βελτιώσεις στο F1-Score σε συνδυασμό με πολλαπλά προ-εκπαιδευμένα μοντέλα, υπογραμμίζοντας την αποτελεσματικότητά του στην επίτευξη ισορροπίας μεταξύ precision και recall. Ωστόσο, παρατηρήθηκαν συμβιβασμοί για μοντέλα όπως το ResNet-34, όπου το F1-Score μειώθηκε με τη χρήση οποιονδήποτε συνθετικών δειγμάτων από τα cGANs. Επίσης το μοντέλο που κάνει χρήση συνθετικών δειγμάτων από το cGAN(128x128) έδειξε σταθερά τις πιο ουσιαστικές βελτιώσεις στη μετρική Accuracy, ιδίως για το ResNet-18. Αξίζει να σημειωθεί ότι η χρήση δειγμάτων από διαφορετικό cGAN επηρέασε διαφορετικά το accuracy σε όλα τα μοντέλα, υποδεικνύοντας έτσι ότι ορισμένες διαμορφώσεις μπορεί να είναι καταλληλότερες για συγκεκριμένα προ-εκπαιδευμένα μοντέλα.

Οι τιμές δέλτα παρείχαν λεπτομερή κατανόηση των αλλαγών σε κάθε μετρική όταν εισήχθησαν δείγματα από τα cGAN. Τα θετικά δέλτα υποδείκνυαν βελτιώσεις, ενώ τα αρνητικά δέλτα υπογράμμιζαν μειώσεις στην απόδοση. Συνολικά, τα αποτελέσματα υποδηλώνουν ότι η επιλογή της μεθόδου cGAN και η αλληλεπίδρασή της με τα προ-εκπαιδευμένα μοντέλα είναι πολύ σημαντική

κατά την πραγματοποίηση εργασιών ταξινόμησης εικόνων. Είναι προφανές ότι η χρήση συνθετικών δειγμάτων από το cGAN (128x128) οδηγεί σε σταθερά καλές επιδόσεις σε διάφορες μετρικές, γεγονός που το καθιστά μια υποσχόμενη επιλογή για τη βελτίωση της απόδοσης ταξινόμησης.

Λαμβάνοντας υπόψη τις μετρικές που αναφέρθηκαν παραπάνω, το μοντέλο με συνθετικά δείγματα από το cGAN (128x128) σε συνδυασμό με το ResNet-18 υπερτερεί σταθερά έναντι των άλλων μοντέλων όσον αφορά το precision, το F1-Score και το Accuracy. Ωστόσο, έχει χαμηλότερο recall από το αρχικό μοντέλο που δεν κάνει χρήση συνθετικών δειγμάτων. Η επιλογή του καλύτερου μοντέλου εξαρτάται από τις ειδικές απαιτήσεις της εφαρμογής. Στην περίπτωση αυτής της εργασίας, ο κύριος ο στόχος ήταν η βελτιστοποίηση της απόδοσης του μοντέλου με τη χρήση ενός cGAN ως προς τη μειωτική κλάση, πράγμα το οποίο και επιτυγχάνεται, όπως φαίνεται από την τιμή του precision μετά τη χρήση του cGAN (128x128).

Επειδή στη συγκεκριμένη εργασία, ο στόχος ήταν να γίνει διαχείριση της ανισορροπίας των δυο κλάσεων και να βελτιωθεί η απόδοση του μοντέλου ως προς τη μειωτική κλάση με τη χρήση GAN, δε δόθηκε έμφαση τόσο στο πραγματικό νόημα και τη βαρύτητα που έχει η κάθε κλάση.

Για παράδειγμα, στο συγκεκριμένο πρόβλημα τυγχάνει η μειωτική κλάση να είναι η κλάση Normal, η οποία περιλαμβάνει εικόνες που δεν φέρουν την ασθένεια (υγιής πληθυσμός). Αυτό σημαίνει ότι το μοντέλο θα έχει μια τάση να προβλέπει πιο σωστά τα άτομα τα οποία φέρουν την ασθένεια της πνευμονίας και να αστοχεί στα υγιή. Δηλαδή μπορεί να ταξινομούσε κάποιον ότι έχει την ασθένεια, ενώ δεν την έχει στην πραγματικότητα. Αυτό προφανώς δεν είναι σωστό αλλά δεν έχει την ίδια βαρύτητα με το να συμβαίνει το ανάποδο δηλαδή, ένα άτομο το οποίο έχει την ασθένεια να τον ταξινομήσει ως υγιή γιατί τότε οι συνέπειες μπορεί να προβούν μοιραίες. Επομένως για τις ανάγκες της παρούσας εργασίας, το καλύτερο μοντέλο είναι αυτό που χρησιμοποιεί τα συνθετικά δείγματα από το cGAN (128x128). Εάν ο στόχος ήταν η μείωση των ψευδώς αρνητικών αποτελεσμάτων, τότε το μοντέλο χωρίς τη χρήση συνθετικών εικόνων cGAN σε συνδυασμό με το MobileNet\_V2 θα μπορούσε να προτιμηθεί λόγω του υψηλού recall του.

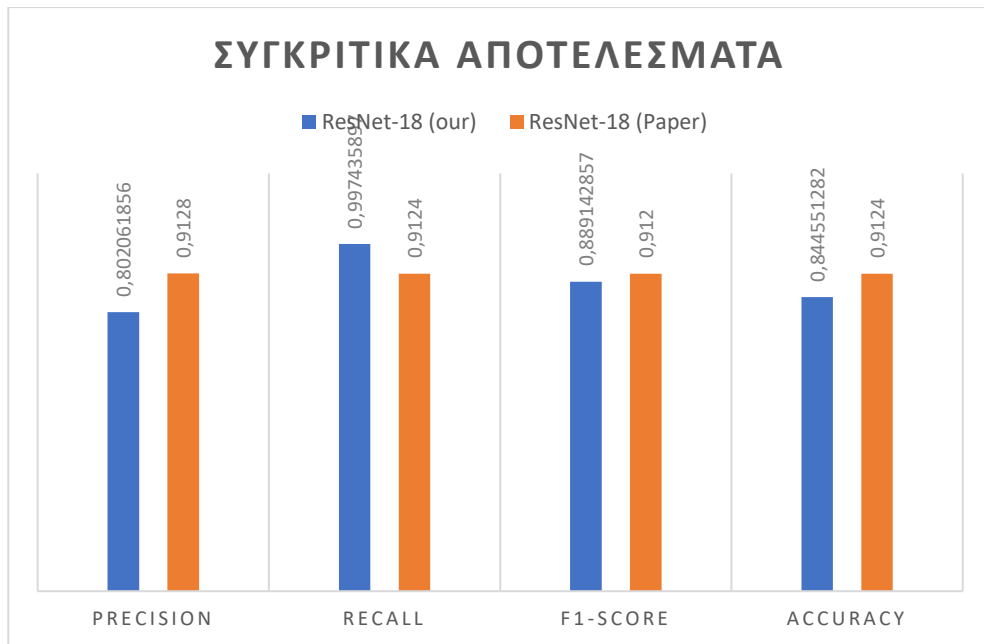
### 5.3 Σύγκριση αποτελεσμάτων με δημοσίευση

Στην ενότητα αυτή συγκρίνουμε τα αποτελέσματα του ταξινομητή που δημιουργήσαμε στην ενότητα 4.5.1 με τον ταξινομητή των ερευνητών της (ενότητα 4.5.2) δημοσίευσης [53].

Συγκρίνοντας τα αποτελέσματα (Εικόνα 109) των δύο παραπάνω μοντέλων, διαπιστώνεται ότι όσον αφορά τη μετρική Precision το μοντέλο ResNet-18 των ερευνητών χρησιμοποιώντας δείγματα από το ACGAN πετυχαίνει μεγαλύτερη τιμή (0,9128) σε σχέση με το μοντέλο αυτής της εργασίας (0,8021) υποδηλώνοντας έτσι ότι έχει λιγότερα ψευδώς θετικά αποτελέσματα (FP).

Από την άλλη, σχετικά με τη μετρική Recall το παρόν μοντέλο υπερισχύει έναντι του μοντέλου των ερευνητών με τιμή (0.9974) έναντι (0.9124), πράγμα που σημαίνει ότι αναγνωρίζει μεγαλύτερο ποσοστό πραγματικών θετικών (TP) περιπτώσεων από ότι το μοντέλο των ερευνητών. Σχετικά με τη μετρική F1-Score το μοντέλο των ερευνητών πετυχαίνει μεγαλύτερη τιμή (0.9120) σε σχέση με το δικό μας (0.8891), το οποίο φανερώνει καλύτερη εξισορρόπηση μεταξύ του precision και recall σε σχέση με το δικό μας. Τέλος, όσον αφορά τη μετρική Accuracy το μοντέλο των ερευνητών υπερτερεί και πάλι έναντι του δικού μας με τιμή 0.9124 έναντι 0.8446 πράγμα που σημαίνει ότι έχει μεγαλύτερο ποσοστό σωστών προβλέψεων.

Εάν θέλαμε να επιλέξουμε ένα από τα δυο μοντέλα η επιλογή μεταξύ αυτών των μοντέλων θα εξαρτιόταν από τις συγκεκριμένες απαιτήσεις της εφαρμογής. Η μετρική precision στην συγκεκριμένη εργασία είναι πιο κρίσιμη, επομένως το μοντέλο των ερευνητών αποτελεί σίγουρα καλύτερη επιλογή. Ωστόσο, εάν η σύλληψη όσο το δυνατόν περισσότερων θετικών περιπτώσεων (high recall) είναι απαραίτητη, τότε το μοντέλο που κατασκευάστηκε στην παρούσα εργασία θα ήταν καταλληλότερο, παρά τις ελαφρώς χαμηλότερες τιμές σε Precision και F1-Score.



Εικόνα 109 Αποτελέσματα του Classifier με χρήση συνθετικών δειγμάτων από cGAN εναντίον του Classifier της δημοσίευσης[53] με χρήση συνθετικών δειγμάτων από ACGAN

*Σημείωση: Όσον αφορά τους χρόνους εκτέλεσης δεν είναι εφικτό να συμπεριληφθεί στη σύγκριση αν και σίγουρα είναι ένα κριτήριο που παίζει σημαντικό ρολό διότι δεν είναι ακριβώς γνωστοί οι χρόνοι που υλοποιήθηκε το πείραμα των ερευνητών αλλά και τι υλικό χρησιμοποιήθηκε κατά την έρευνα. Στη δημοσίευση αναφέρουν επίσης ότι οι χρόνοι είναι ενδεικτικοί.*

## **6 ΚΕΦΑΛΑΙΟ 6: ΣΥΜΠΕΡΑΣΜΑΤΑ & ΕΠΙΛΟΓΟΣ**

### **6.1 Συμπεράσματα**

Στόχος της παρούσας διπλωματικής εργασίας ήταν η διαχείριση του προβλήματος της ανισορροπίας των κλάσεων σε ένα dataset που αφορά ιατρικά δεδομένα και η χρήση cGAN ως τρόπο αντιμετώπισης του προβλήματος. Μετά τη διεξαγωγή αρκετών πειραμάτων και διαφορετικών αρχιτεκτονικών επιτεύχθηκε ο στόχος και με τη χρήση cGAN βελτιστοποιήθηκε η απόδοση του ταξινομητή ως προς τη μειωτική κλάση (Normal) του Chest Xray Dataset που επιλέχθηκε για τις ανάγκες της συγκεκριμένης εργασίας.

Το μοντέλο που ανταποκρίθηκε καλύτερα από όλα ήταν ο Classifier που εκπαιδεύτηκε ξανά με την προσθήκη συνθετικών δειγμάτων από το cGAN (128x128), καθώς συγκέντρωσε τα λιγότερα ψευδώς θετικά (FP) στην κλάση 0 (Normal). Επίσης, ο αριθμός των δειγμάτων που προστέθηκαν ήταν 2690 για τη μειωτική κλάση. Αυτό που διαπιστώθηκε από το συγκεκριμένο αποτέλεσμα ήταν ότι η ανάλυση της εικόνας επηρεάζει σημαντικά την απόδοση του μοντέλου, καθώς όσο αυξάνεται, τόσο βελτιώνεται και η απόδοση του μοντέλου.

Επίσης εντύπωση προκαλεί το γεγονός ότι ενώ το cGAN(128x128) είχε τις υψηλότερες βαθμολογίες FID και KID μεταξύ των τριών μοντέλων cGAN, εξακολουθεί να φαίνεται ότι είχε το καλύτερο αντίκτυπο στην εκπαίδευση και την απόδοση του ταξινομητή όταν τα παραγόμενα δείγματά του χρησιμοποιήθηκαν για να αυξήσουν τα δεδομένα εκπαίδευσης του ταξινομητή. Αυτό μπορεί να οφείλεται στο γεγονός ότι οι εικόνες που παράγονται από το cGAN(128x128) μπορεί να μην έχουν μεγάλη ποικιλομορφία. Αντιθέτως, φαίνεται να είναι πιο σχετικές ή χρήσιμες για τη συγκεκριμένη εργασία ταξινόμησης στην οποία εκπαιδεύεται ο ταξινομητής. Συνεπώς, παρόλο που η ποικιλομορφία (μετρούμενη από το FID και το KID) παρουσίαζε υψηλότερες τιμές σε σχέση με τα άλλα μοντέλα cGAN, οι εικόνες που παρήχθησαν ενδέχεται να περιέχουν χαρακτηριστικά ή μοτίβα που είναι πιο χρήσιμα για την εκπαίδευση του ταξινομητή. Επίσης, ενδέχεται να επαυξάνουν το σύνολο δεδομένων με τρόπο τέτοιο που να προσομοιώνει αποτελεσματικά τις παραλλαγές του πραγματικού κόσμου, βοηθώντας τον ταξινομητή να γενικεύει καλύτερα σε δεδομένα που δεν έχει ξαναδεί.

Ένα άλλο βασικό στοιχείο φαίνεται να είναι το μέγεθος της εικόνας. Η αλλαγή στο μέγεθος της εικόνας βελτίωσε τα αποτελέσματα εκπαιδύοντας το cGAN για μόλις 5 εποχές. Το μειονέκτημα βέβαια είναι ότι η αλλαγή στο μέγεθος της εικόνας πρακτικά σημαίνει αύξηση των επιπέδων μέσα στα δίκτυα του Generator και του Discriminator, στοιχείο που αυξάνει την πολυπλοκότητα και τον χρόνο εκπαίδευσης κατά πολύ. Παρόλα αυτά, το συγκεκριμένο πείραμα έδειξε ότι για να πετύχουμε καλύτερα αποτελέσματα δεν χρειάζεται απαραίτητα να έχουμε μεγάλο αριθμό επαναλήψεων, καθώς και ότι ο μικρός αριθμός επαναλήψεων δεν σημαίνει και πιο γρήγορη εκπαίδευση του μοντέλου.

Στη συνέχεια, εκπαιδύοντας ένα cGAN για 600 εποχές και επιλέγοντας το καλύτερο μοντέλο (κρατώντας checkpoint ανά 50 εποχές), το οποίο ήταν αυτό με τις 200 εποχές) και χρησιμοποιώντας παρόμοιες παραμέτρους με τους ερευνητές, εκπαιδεύσαμε ξανά τον Classifier με αριθμό συνθετικών δειγμάτων  $k=500$  για κάθε κλάση και συγκρίναμε το μοντέλο μας (χρήση ResNet-18) με το αντίστοιχο μοντέλο των ερευνητών. Διαπιστώσαμε ότι με βάση τις μετρικές το μοντέλο μας αν και υστερεί στις περισσότερες μετρικές υπερτερώντας μόνο στο Recall, σε μια περίπτωση που θα μας ένοιαζε ο όσο το δυνατόν μεγαλύτερος αριθμός θετικών περιπτώσεων (στην περίπτωση μας κλάση 1: Πνευμονία) θα μπορούσε να προτιμηθεί έναντι του μοντέλου των ερευνητών.

Επίσης, από όλα τα παραπάνω πειράματα που υλοποιήθηκαν στη συγκεκριμένη εργασία διαπιστώσαμε ότι οι υπερπαραμέτροι όπως το learning rate, το batch size, ο optimizer στην εκπαίδευση των GANs μπορεί να επηρεάσουν όλη την απόδοση του μοντέλου είτε θετικά, είτε αρνητικά. Επίσης μεγάλο ρόλο έχει και η αρχιτεκτονική. Στη δική μας μελέτη φάνηκε ότι η χρήση GAN με Wasserstein Distance με Gradient Penalty απέδωσε καλύτερα από κάθε άλλη αρχιτεκτονική που δοκιμάστηκε στο πλαίσιο υλοποίησης της συγκεκριμένης εργασίας.

### *Προβλήματα*

Το κύριο πρόβλημα που είχαμε κατά τη διάρκεια υλοποίησης της εργασίας είναι η έλλειψη πόρων, στοιχείο που περιόρισε αρκετά την υλοποίηση πειραμάτων. Η εκπαίδευση ενός cGAN αποτελεί μια δύσκολη διαδικασία για έναν συμβατικό υπολογιστή. Για παράδειγμα, ο περιορισμός του hardware κυρίως σε μέγεθος GPU οδηγούσε σε προβλήματα έλλειψης μνήμης όταν προσπαθούσαμε να δοκιμάσουμε διαφορετικές παραμέτρους, όπως η αύξηση του batch size και η

αλλαγή της εικόνας σε μεγαλύτερο μέγεθος, για να δούμε την επίπτωση στην απόδοση του μοντέλου. Το ίδιο ίσχυε και για την αδυναμία χρήσης περισσότερων επιπέδων είτε στα δίκτυα του Generator και του Discriminator, είτε στη χρήση διαφορετικών pretrained μοντέλων για την εκπαίδευση του Classifier με περισσότερα επίπεδα. Για αυτόν τον λόγο, βλέπουμε στην εργασία ότι το ResNet 18 επικρατεί περισσότερο στα περισσότερα πειράματα, καθώς ήταν ένα δίκτυο το οποίο μπορούσε να υποστηριχτεί σε μεγάλο βαθμό από το hardware μας. Για τα πιο απαιτητικά πειράματα που βλέπουμε στην εργασία χρειάστηκε να χρησιμοποιήσουμε αρκετές φορές επί πληρωμή συνδρομητικά πακέτα του Google Colab, τα οποία μας έδιναν πόρους ισχυρών GPU. Επίσης, το γεγονός ότι μετασχηματίσαμε το κυρίως dataset σε 64x64 για το μεγαλύτερο μέρος των δοκιμών μας οφείλεται στο συγκεκριμένο πρόβλημα.

## **6.2 Επίλογος-Μελλοντικές προσεγγίσεις**

Στην παρούσα διπλωματική εργασία, επιχειρήθηκε η αντιμετώπιση μιας κρίσιμης πρόκλησης στην ανάλυση ιατρικών εικόνων, δηλαδή του προβλήματος της ανισορροπίας των κλάσεων στο σύνολο δεδομένων ακτινών X του θώρακα (Chest Xray Dataset). Συγκεκριμένα, χρησιμοποιήσαμε τα Conditional Generative Adversarial Networks (cGANs) για τη δημιουργία συνθετικών δειγμάτων της μειονοτικής κλάσης, Normal, με απώτερο στόχο την ενίσχυση της απόδοσης του ταξινομητή μας.

Η διαδικασία που ακολουθήσαμε ξεκίνησε με μια ολοκληρωμένη διερεύνηση του ζητήματος της ανισορροπίας των κλάσεων, το οποίο είναι σύνηθες στα σύνολα δεδομένων ιατρικής απεικόνισης. Αναγνωρίσαμε ότι η μειονότητα των περιπτώσεων Normal στο σύνολο δεδομένων θα μπορούσε να οδηγήσει σε μεροληπτικούς ταξινομητές, ενέχοντας σημαντικούς κλινικούς κινδύνους.

Τα cGANs ως λύση αποδείχθηκαν πολύ σημαντικά για την πραγματοποίηση της μελέτης. Εξαρτώντας το δίκτυο του Generator από την ετικέτα της κλάσης, αξιοποιήσαμε τα GANs για τη σύνθεση υψηλής πιστότητας, σχετικών με το πλαίσιο εικόνων ακτινογραφίας θώρακα. Τα πειράματά μας έδειξαν ότι αυτή η στρατηγική συνθετικής επαύξησης βελτίωσε την απόδοση του ταξινομητή, με αποτέλεσμα πιο ακριβείς προβλέψεις σε σχέση με την κλάση Normal.

Όσον αφορά μελλοντικές προσθήκες που θα μπορούσαν να γίνουν, αυτές μπορεί να αφορούν την εφαρμογή τεχνικών, όπως τα Progressive GAN (ProGAN). Τα

Progressive GANs (Generative Adversarial Networks) είναι μια προηγμένη παραλλαγή της παραδοσιακής αρχιτεκτονικής GAN που εισήχθη για τη δημιουργία εικόνων υψηλής ανάλυσης, ενώ παράλληλα αντιμετωπίζει ορισμένες κοινές προκλήσεις στην εκπαίδευση γενετικών μοντέλων.

Η ικανότητά τους να παράγουν εικόνες υψηλής ποιότητας και ποικιλίας τα έχει καταστήσει δημοφιλή επιλογή σε τομείς όπου η ρεαλιστική σύνθεση εικόνων είναι κρίσιμη. Σίγουρα η χρήση μιας τέτοιας τεχνικής θα βελτιώνει ακόμα περισσότερο το τελικό αποτέλεσμα και τα συνθετικά δείγματα που θα παράγαμε θα ήταν πολύ πιο ποιοτικά και ρεαλιστικά.

Επίσης, μεγάλο ενδιαφέρον θα είχε και η χρήση diffusion Models (Μοντέλα Διάχυσης) τα οποία είναι μια κατηγορία παραγωγικών μοντέλων που χρησιμοποιούνται στη μηχανική μάθηση για την πιθανολογική μοντελοποίηση δεδομένων. Σε αντίθεση με τα παραδοσιακά παραγωγικά μοντέλα, όπως τα Generative Adversarial Networks (GANs) ή οι Variational Autoencoders (VAEs), τα μοντέλα διάχυσης επικεντρώνονται στη μοντελοποίηση του σταδιακού μετασχηματισμού μιας απλής βασικής κατανομής στην κατανομή των δεδομένων-στόχου.

Ένα από τα κύρια χαρακτηριστικά των μοντέλων διάχυσης είναι η εκπαίδευσή τους με βάση την πιθανοφάνεια, όπου το μοντέλο εκπαιδεύεται για να μεγιστοποιήσει άμεσα την πιθανότητα δημιουργίας παρατηρούμενων δεδομένων. Αυτό βέβαια έρχεται σε αντίθεση με τα GAN, όπου η εκπαίδευση περιλαμβάνει μια αντιπαλική διαδικασία μεταξύ generator και discriminator.

Τα diffusion models υποστηρίζουν τη δημιουργία δειγμάτων υψηλής ποιότητας, αντιμετωπίζοντας προκλήσεις, όπως η κατάρρευση του τρόπου λειτουργίας (πρόβλημα που αντιμετωπίσαμε και εμείς σε αυτήν την εργασία) που μπορεί να αντιμετωπιστούν σε άλλα παραγωγικά μοντέλα. Η σταδιακή φύση τους επιτρέπει, επίσης, λεπτομερή έλεγχο της παραγωγικής διαδικασίας.

Θα μπορούσε επίσης να χρησιμοποιηθεί και κάποιος συνδυασμός των δύο αυτών τεχνικών. Θα μπορούσαν ενδεικτικά να προκύψουν υβριδικά μοντέλα που ενσωματώνουν τα πλεονεκτήματα και των δύο προσεγγίσεων, αξιοποιώντας την εκπαίδευση των μοντέλων διάχυσης με βάση την πιθανότητα και το πρότυπο της αντιπαλικής εκπαίδευσης των Progressive GAN.



Η επιτυχής εφαρμογή των cGANs στην ανάλυση ιατρικών εικόνων αναδεικνύει τις δυνατότητές τους να φέρουν επανάσταση στις στρατηγικές επαύξησης δεδομένων σε διάφορους τομείς. Στο πλαίσιο αυτό, είναι σημαντικό να αναγνωρίσουμε τις ηθικές και ρυθμιστικές διαστάσεις κατά την ανάπτυξη τέτοιων συστημάτων υποβοηθούμενων από τεχνητή νοημοσύνη στην κλινική πρακτική. Η υπεύθυνη ενσωμάτωση λύσεων που βασίζονται σε cGAN στις ροές εργασίας στον τομέα της διάγνωσης και θεραπείας απαιτεί αυστηρή επικύρωση, ερμηνευσιμότητα και τήρηση των καθιερωμένων ιατρικών προτύπων και πρωτοκόλλων.

Κλείνοντας, η παρούσα εργασία τόνισε τις μετασχηματιστικές δυνατότητες των cGANs στον περιορισμό των προβλημάτων ανισορροπίας κλάσεων στην ανάλυση ιατρικών εικόνων. Βελτιώνοντας την απόδοση του ταξινομητή στην ανίχνευση περιπτώσεων πνευμονίας και μη. Η ερευνητική κοινότητα, ωστόσο, οφείλει να εστιάσει στη βελτίωση και επέκταση αυτών των μεθοδολογιών, λαμβάνοντας πάντα υπόψη ζητήματα ηθικής για τη χρήση της τεχνητής νοημοσύνης, καθώς και τη διάσταση της ασθενοκεντρικότητας, διασφαλίζοντας έτσι τη θετική επίπτωση τέτοιων καινοτόμων εργαλείων στην κοινωνία.



## Βιβλιογραφία

- [1] “Pneumonia.” Accessed: Apr. 20, 2023. [Online]. Available: <https://www.who.int/health-topics/pneumonia>
- [2] “Pneumonia - Diagnosis | NHLBI, NIH.” Accessed: Dec. 28, 2023. [Online]. Available: <https://www.nlm.nih.gov/health/pneumonia/diagnosis>
- [3] A. Manickam, J. Jiang, Y. Zhou, A. Sagar, R. Soundrapandiyan, and R. Dinesh Jackson Samuel, “Automated pneumonia detection on chest X-ray images: A deep learning approach with different optimizers and transfer learning architectures,” *Measurement*, vol. 184, p. 109953, Nov. 2021, doi: 10.1016/j.measurement.2021.109953.
- [4] G. Liang and L. Zheng, “A transfer learning method with deep residual network for pediatric pneumonia diagnosis,” *Comput. Methods Programs Biomed.*, vol. 187, p. 104964, Apr. 2020, doi: 10.1016/j.cmpb.2019.06.023.
- [5] M. Salehi, R. Mohammadi, H. Ghaffari, N. Sadighi, and R. Reiazi, “Automated detection of pneumonia cases using deep transfer learning with paediatric chest X-ray images,” *Br. J. Radiol.*, vol. 94, no. 1121, p. 20201263, May 2021, doi: 10.1259/bjr.20201263.
- [6] D. Santos, “Detection and Segmentation of Pneumonia in Medical Images Using Convolutional Neural Networks,” *Qeios*, Sep. 2023, doi: 10.32388/4SAIUY.
- [7] X. Guo, Y. Yin, C. Dong, G. Yang, and G. Zhou, “On the Class Imbalance Problem,” in *2008 Fourth International Conference on Natural Computation*, Jul. 2008, pp. 192–201. doi: 10.1109/ICNC.2008.871.
- [8] Y. Li, Z. Zhang, C. Dai, Q. Dong, and S. Shirzadikia, “Accuracy of deep learning for automated detection of pneumonia using chest X-Ray images: A systematic review and meta-analysis,” *Comput. Biol. Med.*, vol. 123, p. 103898, Jul. 2020, doi: 10.1016/j.combiomed.2020.103898.
- [9] V. Jain, R. Vashisht, G. Yilmaz, and A. Bhardwaj, “Pneumonia Pathology,” in *StatPearls*, Treasure Island (FL): StatPearls Publishing, 2024. Accessed: Feb. 10, 2024. [Online]. Available: <http://www.ncbi.nlm.nih.gov/books/NBK526116/>
- [10] C. Hacking, “Pneumonia | Radiology Reference Article | Radiopaedia.org,” Radiopaedia. Accessed: Feb. 10, 2024. [Online]. Available: <https://radiopaedia.org/articles/pneumonia?lang=us>
- [11] Y. Xia, Y. Ying, S. Wang, W. Li, and H. Shen, “Effectiveness of lung ultrasonography for diagnosis of pneumonia in adults: a systematic review and meta-analysis,” *J. Thorac. Dis.*, vol. 8, no. 10, pp. 2822–2831, Oct. 2016, doi: 10.21037/jtd.2016.09.38.
- [12] A. M. Y. Cao, J. P. Choy, L. N. Mohanakrishnan, R. F. Bain, and M. L. van Driel, “Chest radiographs for acute lower respiratory tract infections,” *Cochrane Database Syst. Rev.*, vol. 2013, no. 12, p. CD009119, Dec. 2013, doi: 10.1002/14651858.CD009119.pub2.
- [13] E. J. Hwang, J. H. Lee, J. H. Kim, W. H. Lim, J. M. Goo, and C. M. Park, “Deep learning computer-aided detection system for pneumonia in febrile neutropenia patients: a diagnostic cohort study,” *BMC Pulm. Med.*, vol. 21, no. 1, p. 406, Dec. 2021, doi: 10.1186/s12890-021-01768-0.
- [14] W. H. Kim and Y. I. Kim, “Neoplastic Paneth cells in the experimental murine carcinoma of the small intestine,” *J. Korean Med. Sci.*, vol. 5, no. 4, pp. 197–203, Dec. 1990.

- [15] S. A. Alzahrani, M. A. Al-Salamah, W. H. Al-Madani, and M. A. Elbarbary, "Systematic review and meta-analysis for the use of ultrasound versus radiology in diagnosing of pneumonia," *Crit. Ultrasound J.*, vol. 9, p. 6, Feb. 2017, doi: 10.1186/s13089-017-0059-y.
- [16] S. Shang *et al.*, "Performance of a computer aided diagnosis system for SARS-CoV-2 pneumonia based on ultrasound images," *Eur. J. Radiol.*, vol. 146, p. 110066, Jan. 2022, doi: 10.1016/j.ejrad.2021.110066.
- [17] E. Alexopoulou *et al.*, "Imaging of Acute Complications of Community-Acquired Pneumonia in the Paediatric Population—From Chest Radiography to MRI," *Children*, vol. 11, no. 1, Art. no. 1, Jan. 2024, doi: 10.3390/children11010122.
- [18] C. Ortiz-Toro, A. García-Pedrero, M. Lillo-Saavedra, and C. Gonzalo-Martín, "Automatic detection of pneumonia in chest X-ray images using textural features," *Comput. Biol. Med.*, vol. 145, p. 105466, Jun. 2022, doi: 10.1016/j.compbiomed.2022.105466.
- [19] C. Qin, D. Yao, Y. Shi, and Z. Song, "Computer-aided detection in chest radiography based on artificial intelligence: a survey," *Biomed. Eng. OnLine*, vol. 17, p. 113, Aug. 2018, doi: 10.1186/s12938-018-0544-y.
- [20] M. Mittal *et al.*, "An Efficient Edge Detection Approach to Provide Better Edge Connectivity for Image Analysis," *IEEE Access*, vol. 7, pp. 33240–33255, 2019, doi: 10.1109/ACCESS.2019.2902579.
- [21] S. Nagashree and B. S. Mahanand, "Pneumonia Chest X-ray Classification Using Support Vector Machine," in *Proceedings of International Conference on Data Science and Applications*, M. Saraswat, C. Chowdhury, C. Kumar Mandal, and A. H. Gandomi, Eds., in *Lecture Notes in Networks and Systems*. Singapore: Springer Nature, 2023, pp. 417–425. doi: 10.1007/978-981-19-6634-7\_29.
- [22] R. Al Mamlook, S. Chen, and H. Bzizi, *Investigation of the performance of Machine Learning Classifiers for Pneumonia Detection in Chest X-ray Images*. 2020. doi: 10.1109/EIT48999.2020.9208232.
- [23] D. J. Alapat, M. V. Menon, and S. Ashok, "A Review on Detection of Pneumonia in Chest X-ray Images Using Neural Networks," *J. Biomed. Phys. Eng.*, vol. 12, no. 6, pp. 551–558, Dec. 2022, doi: 10.31661/jbpe.voio.2202-1461.
- [24] R. Kundu, R. Das, Z. W. Geem, G.-T. Han, and R. Sarkar, "Pneumonia detection in chest X-ray images using an ensemble of deep learning models," *PLoS ONE*, vol. 16, no. 9, p. e0256630, Sep. 2021, doi: 10.1371/journal.pone.0256630.
- [25] R. Nandi and M. Mulimani, "Detection of COVID-19 from X-rays using hybrid deep learning models," *Res. Biomed. Eng.*, vol. 37, no. 4, pp. 687–695, Dec. 2021, doi: 10.1007/s42600-021-00181-0.
- [26] S. Candemir and S. Antani, "A review on lung boundary detection in chest X-rays," *Int. J. Comput. Assist. Radiol. Surg.*, vol. 14, no. 4, pp. 563–576, 2019, doi: 10.1007/s11548-019-01917-1.
- [27] V. Pitroda, M. M. Fouda, and Z. M. Fadlullah, "An Explainable AI Model for Interpretable Lung Disease Classification," in *2021 IEEE International Conference on Internet of Things and Intelligence Systems (IoT&IS)*, Aug. 2021, pp. 98–103. doi: 10.1109/IoT&IS53735.2021.9628573.
- [28] A. Ali, S. M. Shamsuddin, and A. Ralescu, "Classification with class imbalance problem: A review," vol. 7, pp. 176–204, Jan. 2015.

- [29] M. F. Hashmi, S. Katiyar, A. G. Keskar, N. D. Bokde, and Z. W. Geem, "Efficient Pneumonia Detection in Chest Xray Images Using Deep Transfer Learning," *Diagnostics*, vol. 10, no. 6, Art. no. 6, Jun. 2020, doi: 10.3390/diagnostics10060417.
- [30] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.
- [31] J. Sun, J. Lang, H. Fujita, and H. Li, "Imbalanced enterprise credit evaluation with DTE-SBD: Decision tree ensemble based on SMOTE and bagging with differentiated sampling rates," *Inf. Sci.*, vol. 425, pp. 76–91, Jan. 2018, doi: 10.1016/j.ins.2017.10.017.
- [32] S. S. Mullick, S. Datta, and S. Das, "Generative Adversarial Minority Oversampling," arXiv, Aug. 26, 2020. doi: 10.48550/arXiv.1903.09730.
- [33] A. Basu, S. Das, S. S. Mullick, and S. Das, "Do Preprocessing and Class Imbalance Matter to the Deep Image Classifiers for COVID-19 Detection? An Explainable Analysis," *IEEE Trans. Artif. Intell.*, vol. 4, no. 2, pp. 229–241, Apr. 2023, doi: 10.1109/TAI.2022.3149971.
- [34] G. Douzas, F. Bacao, and F. Last, "Improving imbalanced learning through a heuristic oversampling method based on k-means and SMOTE," *Inf. Sci.*, vol. 465, pp. 1–20, Oct. 2018, doi: 10.1016/j.ins.2018.06.056.
- [35] J. Wang, M. Xu, H. Wang, and J. Zhang, "Classification of Imbalanced Data by Using the SMOTE Algorithm and Locally Linear Embedding," in *2006 8th international Conference on Signal Processing*, Aug. 2006. doi: 10.1109/ICOSP.2006.345752.
- [36] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, Jun. 2008, pp. 1322–1328. doi: 10.1109/IJCNN.2008.4633969.
- [37] "A Review on Imbalanced Data Handling Using Undersampling and Oversampling Technique," *Int. J. Recent Trends Eng. Res.*, vol. 3, no. 4, pp. 444–449, May 2017, doi: 10.23883/IJRTER.2017.3168.0UWXM.
- [38] V. Babar and R. Ade, "A Novel Approach for Handling Imbalanced Data in Medical Diagnosis using Undersampling Technique," *Commun. Appl. Electron.*, vol. 5, no. 7, pp. 36–42, Jul. 2016, doi: 10.5120/cae2016652323.
- [39] "Redundancy-driven modified Tomek-link based undersampling: A solution to class imbalance | Elsevier Enhanced Reader." Accessed: Mar. 05, 2023. [Online]. Available: <https://reader.elsevier.com/reader/sd/pii/S0167865516302719?token=2D7F2076C1E3B781D9B3F8BDC84B7F47B51F57226E1485074C6B5CE1DAA20BoFC8AB7A5D7AAFDDoDAF4E67Fo1ECA6202&originRegion=eu-west-1&originCreation=20230305145211>
- [40] R. Blagus and L. Lusa, "Evaluation of SMOTE for High-Dimensional Class-Imbalanced Microarray Data," in *2012 11th International Conference on Machine Learning and Applications*, Sep. 2012, pp. 89–94. doi: 10.1109/ICMLA.2012.183.
- [41] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A Fast Learning Algorithm for Deep Belief Nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006, doi: 10.1162/neco.2006.18.7.1527.
- [42] J. Hwang, W. Hwang, and J. Jo, "Tractable loss function and color image generation of multinary restricted Boltzmann machine." arXiv, Nov. 26,

2020. Accessed: Feb. 15, 2024. [Online]. Available: <http://arxiv.org/abs/2011.13509>
- [43] D. Bank, N. Koenigstein, and R. Giryes, "Autoencoders," in *Machine Learning for Data Science Handbook: Data Mining and Knowledge Discovery Handbook*, L. Rokach, O. Maimon, and E. Shmueli, Eds., Cham: Springer International Publishing, 2023, pp. 353–374. doi: 10.1007/978-3-031-24628-9\_16.
- [44] "Paper11J -Anomaly\_Detection\_on\_Medical\_Images.pdf," Google Docs. Accessed: Feb. 16, 2024. [Online]. Available: [https://drive.google.com/file/d/1YqcJLPphju3I8IkkM1xBegnK2zfKFF\\_v/view?usp=embed\\_facebook](https://drive.google.com/file/d/1YqcJLPphju3I8IkkM1xBegnK2zfKFF_v/view?usp=embed_facebook)
- [45] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes." arXiv, Dec. 10, 2022. doi: 10.48550/arXiv.1312.6114.
- [46] L. Girin, S. Leglaive, X. Bie, J. Diard, T. Hueber, and X. Alameda-Pineda, "Dynamical Variational Autoencoders: A Comprehensive Review," *Found. Trends® Mach. Learn.*, vol. 15, no. 1–2, pp. 1–175, 2021, doi: 10.1561/22000000089.
- [47] S. Chatterjee, S. Maity, M. Bhattacharjee, S. Banerjee, A. K. Das, and W. Ding, "Variational Autoencoder Based Imbalanced COVID-19 Detection Using Chest X-Ray Images," *New Gener. Comput.*, vol. 41, no. 1, pp. 25–60, Mar. 2023, doi: 10.1007/s00354-022-00194-y.
- [48] I. J. Goodfellow *et al.*, "Generative Adversarial Networks." arXiv, Jun. 10, 2014. Accessed: Dec. 21, 2022. [Online]. Available: <http://arxiv.org/abs/1406.2661>
- [49] M. Mirza and S. Osindero, "Conditional Generative Adversarial Nets." arXiv, Nov. 06, 2014. Accessed: May 08, 2023. [Online]. Available: <http://arxiv.org/abs/1411.1784>
- [50] A. Radford, L. Metz, and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks." arXiv, Jan. 07, 2016. Accessed: Dec. 21, 2022. [Online]. Available: <http://arxiv.org/abs/1511.06434>
- [51] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN." arXiv, Dec. 06, 2017. Accessed: Apr. 26, 2023. [Online]. Available: <http://arxiv.org/abs/1701.07875>
- [52] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, "Improved Training of Wasserstein GANs." arXiv, Dec. 25, 2017. doi: 10.48550/arXiv.1704.00028.
- [53] C. F. Cala-or, A. A. Ambita, A. Carajay, and J. F. Sanz, "Detection of Pneumonia in Chest X-Ray Images Using Deep Transfer Learning and Data Augmentation With Auxiliary Classifier Generative Adversarial Network," *Manila J. Sci.*, vol. 14, pp. 35–54, 2021.
- [54] M. H.-M. Khan, Z. Mungloo-Dilmohamud, K. Jhumka, N. Z. Mungloo, and C. Pena-Reyes, "Investigating on Data Augmentation and Generative Adversarial Networks (GAN s) for Diabetic Retinopathy," in *2022 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)*, Maldives, Maldives: IEEE, Nov. 2022, pp. 1–5. doi: 10.1109/ICECCME55909.2022.9988321.
- [55] A. GÜngör *et al.*, "Adaptive Diffusion Priors for Accelerated MRI Reconstruction," *Med. Image Anal.*, vol. 88, p. 102872, Aug. 2023, doi: 10.1016/j.media.2023.102872.

- [56] L. Yang *et al.*, “Diffusion Models: A Comprehensive Survey of Methods and Applications,” *ACM Comput. Surv.*, vol. 56, no. 4, p. 105:1-105:39, Aug. 2023, doi: 10.1145/3626235.
- [57] J. Ho, A. Jain, and P. Abbeel, “Denoising Diffusion Probabilistic Models.” arXiv, Dec. 16, 2020. doi: 10.48550/arXiv.2006.11239.
- [58] X. Zhang, A. Gangopadhyay, H.-M. Chang, and R. Soni, *Diffusion Model-Based Data Augmentation for Lung Ultrasound Classification with Limited Data*. 2023.
- [59] Y. Baştanlar and M. Özuysal, “Introduction to Machine Learning,” in *miRNomics: MicroRNA Biology and Computational Analysis*, vol. 1107, M. Yousef and J. Allmer, Eds., in *Methods in Molecular Biology*, vol. 1107. , Totowa, NJ: Humana Press, 2014, pp. 105–128. doi: 10.1007/978-1-62703-748-8\_7.
- [60] T. Jo, *Machine Learning Foundations: Supervised, Unsupervised, and Advanced Learning*. Cham: Springer International Publishing, 2021. doi: 10.1007/978-3-030-65900-4.
- [61] P. Y. Glorionec, “Reinforcement Learning: an Overview,” 2000.
- [62] K. G. Kim, “Book Review: Deep Learning,” *Healthc. Inform. Res.*, vol. 22, no. 4, p. 351, 2016, doi: 10.4258/hir.2016.22.4.351.
- [63] S. McCulloch, W. L. H. Pitts, “A LOGICAL CALCULUS OF THE IDEAS IMMANENT IN NERVOUS ACTIVITY”.
- [64] D. E. Rumelhart, R. Durbin, R. Golden, and Y. Chauvin, “Backpropagation: The Basic Theory”.
- [65] N. Aloysius and G. Madathilkulangara, *A review on deep convolutional neural networks*. 2017, p. 0592. doi: 10.1109/ICCSP.2017.8286426.
- [66] J. X. Chen, “The Evolution of Computing: AlphaGo,” *Comput. Sci. Eng.*, vol. 18, no. 4, pp. 4–7, Jul. 2016, doi: 10.1109/MCSE.2016.74.
- [67] V. Sharma, S. Rai, and A. Dev, “A Comprehensive Study of Artificial Neural Networks,” *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*.
- [68] J. Teuwen and N. Moriakov, “Convolutional neural networks,” in *Handbook of Medical Image Computing and Computer Assisted Intervention*, Elsevier, 2020, pp. 481–501. doi: 10.1016/B978-0-12-816176-0.00025-9.
- [69] V. Sze, Y.-H. Chen, T.-J. Yang, and J. Emer, “Efficient Processing of Deep Neural Networks: A Tutorial and Survey.” arXiv, Aug. 13, 2017. Accessed: Mar. 24, 2023. [Online]. Available: <http://arxiv.org/abs/1703.09039>
- [70] K. O’Shea and R. Nash, “An Introduction to Convolutional Neural Networks.” arXiv, Dec. 02, 2015. Accessed: Mar. 30, 2023. [Online]. Available: <http://arxiv.org/abs/1511.08458>
- [71] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, “Convolutional neural networks: an overview and application in radiology,” *Insights Imaging*, vol. 9, no. 4, pp. 611–629, Aug. 2018, doi: 10.1007/s13244-018-0639-9.
- [72] V. Werner de Vargas, J. A. Schneider Aranda, R. dos Santos Costa, P. R. da Silva Pereira, and J. L. Victória Barbosa, “Imbalanced data preprocessing techniques for machine learning: a systematic mapping study,” *Knowl. Inf. Syst.*, vol. 65, no. 1, pp. 31–57, Jan. 2023, doi: 10.1007/s10115-022-01772-8.
- [73] S. Mishra, “Handling Imbalanced Data: SMOTE vs. Random Undersampling,” vol. 04, no. 08.
- [74] “Clustering-based undersampling in class-imbalanced data | Elsevier Enhanced Reader.” Accessed: Mar. 05, 2023. [Online]. Available:

- <https://reader.elsevier.com/reader/sd/pii/S0020025517307235?token=4E46279CAD31ABC8BEE9C758F7240C1F5D97944AA07997FED328D5E4710C4D89FE3F67ECA915D53B23A22AD981836786&originRegion=eu-west-1&originCreation=20230305144813>
- [75] N. M. Mqadi, N. Naicker, and T. Adeliyi, “Solving Misclassification of the Credit Card Imbalance Problem Using Near Miss,” *Math. Probl. Eng.*, vol. 2021, pp. 1–16, Jul. 2021, doi: 10.1155/2021/7194728.
- [76] A. Gosain and S. Sardana, “Handling class imbalance problem using oversampling techniques: A review,” in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Sep. 2017, pp. 79–85. doi: 10.1109/ICACCI.2017.8125820.
- [77] P. Desiana *et al.*, “Combining CNN Feature Extractors and Oversampling Safe Level SMOTE to Enhance Amniotic Fluid Ultrasound Image Classification,” Jan. 2024, doi: 10.22266/ijies2024.0229.24.
- [78] M. Gupta, P. Mehndiratta, and A. Bhardwaj, “Object Recognition in Hand Drawn Images Using Machine Ensembling Techniques and Smote Sampling,” in *Information, Communication and Computing Technology*, A. B. Gani, P. K. Das, L. Kharb, and D. Chahal, Eds., in *Communications in Computer and Information Science*. Singapore: Springer, 2019, pp. 228–239. doi: 10.1007/978-981-15-1384-8\_19.
- [79] G. Ahmed *et al.*, “DAD-Net: Classification of Alzheimer’s Disease Using ADASYN Oversampling Technique and Optimized Neural Network,” *Molecules*, vol. 27, no. 20, Oct. 2022, doi: 10.3390/molecules27207085.
- [80] C. X. Ling and V. S. Sheng, “Cost-Sensitive Learning and the Class Imbalance Problem”.
- [81] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera, “A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches,” *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.*, vol. 42, no. 4, pp. 463–484, Jul. 2012, doi: 10.1109/TSMCC.2011.2161285.
- [82] “Classification and Regression Trees, Bagging, and Boosting | Elsevier Enhanced Reader.” Accessed: Apr. 20, 2023. [Online]. Available: <https://reader.elsevier.com/reader/sd/pii/S0169716104240111?token=EAD9F73C21513FD0FoEoAB72B26A722260C07B7DC715A1EB44E1ACBE82F4C4F418EC5BE0EF52780B8F6E84A74D3C9366&originRegion=eu-west-1&originCreation=20230420145233>
- [83] J. Langr and V. Bok, *GANs in action: deep learning with generative adversarial networks*. Shelter Island, New York: Manning Publications, 2019.
- [84] Y. Wang, “A Mathematical Introduction to Generative Adversarial Nets (GAN).” arXiv, Aug. 31, 2020. Accessed: Apr. 19, 2023. [Online]. Available: <http://arxiv.org/abs/2009.00169>
- [85] S. Ioffe and C. Szegedy, “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift.” arXiv, Mar. 02, 2015. Accessed: May 05, 2023. [Online]. Available: <http://arxiv.org/abs/1502.03167>
- [86] Y. Ti, *Gradient Penalty Approach for Wasserstein Generative Adversarial Networks*. 2022.
- [87] K. Team, “Keras documentation: WGAN-GP overriding `Model.train\_step`.” Accessed: Apr. 28, 2023. [Online]. Available: [https://keras.io/examples/generative/wgan\\_gp/](https://keras.io/examples/generative/wgan_gp/)



- [88] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium." arXiv, Jan. 12, 2018. Accessed: Dec. 16, 2023. [Online]. Available: <http://arxiv.org/abs/1706.08500>
- [89] E. Betzalel, C. Penso, A. Navon, and E. Fetaya, "A Study on the Evaluation of Generative Models." arXiv, Jun. 22, 2022. Accessed: Dec. 16, 2023. [Online]. Available: <http://arxiv.org/abs/2206.10935>
- [90] M. Bińkowski, D. J. Sutherland, M. Arbel, and A. Gretton, "Demystifying MMD GANs." arXiv, Jan. 14, 2021. Accessed: Dec. 16, 2023. [Online]. Available: <http://arxiv.org/abs/1801.01401>
- [91] D. S. Kermany *et al.*, "Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning," *Cell*, vol. 172, no. 5, pp. 1122-1131.e9, Feb. 2018, doi: 10.1016/j.cell.2018.02.010.
- [92] A. Saraiva *et al.*, "Classification of Images of Childhood Pneumonia using Convolutional Neural Networks:," in *Proceedings of the 12th International Joint Conference on Biomedical Engineering Systems and Technologies*, Prague, Czech Republic: SCITEPRESS - Science and Technology Publications, 2019, pp. 112–119. doi: 10.5220/0007404301120119.