



ΔΠΜΣ Μαθηματικής Προτυποποίησης (ΣΕΜΦΕ)

Μελέτη του κβαντικού πλεονεκτήματος αλγορίθμων Μηχανικής Μάθησης κατά τον Σχεδιασμό Φαρμάκων σε Κβαντικούς Υπολογιστές

Καρακάσης Ιωάννης  
09320015

*Επιβλέπων Καθηγητής:*  
Ανδρέας Σταφυλοπάτης  
(Καθηγητής Τμήματος ΗΜΜΥ του ΕΜΠ)

16/06/2022



# Περιεχόμενα

<b>1</b>	<b>Abstract</b>	<b>7</b>
<b>2</b>	<b>Εισαγωγή</b>	<b>11</b>
<b>3</b>	<b>Μέθοδοι Σχεδιασμού Φαρμάκων</b>	<b>13</b>
1.	Πακέτο RDKit . . . . .	16
<b>4</b>	<b>Σύνολο Δεδομένων Βαθμονόμησης και Εξερεύνηση των Δεδομένων</b>	<b>18</b>
<b>5</b>	<b>Μοντέλα Νευρωνικών Δικτύων για Κλασσικούς Υπολογιστές</b>	<b>28</b>
1.	Προβλήματα κατά τη χρήση των GNNs . . . . .	29
2.	Τύποι Graph Neural Networks . . . . .	29
3.	Αλγόριθμος GNN με εστίαση στην Εφαρμογή Συνελκτικών Δικτύων σε Γραφήματα . . . . .	31
4.	Generative Adversarial Networks . . . . .	33
5.	MolGAN model . . . . .	35
<b>6</b>	<b>Quantum Neural Networks</b>	<b>37</b>
1.	Τα θεμέλια για την Κβαντική Εποχή . . . . .	37

2.	Data Encoding στο QML . . . . .	42
3.	Μέθοδος Angle Encoding . . . . .	46
4.	Μεταβλητοί Auto-encoders . . . . .	47
5.	Quantum VAEs . . . . .	51
6.	Quantum Graph Neural Networks . . . . .	54
7.	Quantum GANs . . . . .	58
<b>7</b>	<b>ML Μοντέλα Κλασικών Υπολογιστών - Πειραματική Υλοποίηση</b>	<b>63</b>
1.	Convolutional Neural Network (CNN) . . . . .	69
2.	Graph Neural Network (GNN) . . . . .	75
3.	Generative Adversarial Network (GAN) . . . . .	78
<b>8</b>	<b>Μοντέλα Quantum ML - Πειραματική Υλοποίηση</b>	<b>83</b>
1.	Πώς να "τρέξετε" κβαντικούς αλγόριθμους . . . . .	83
2.	Σύγκριση μεθόδων κωδικοποίησης για ταξινομητές κβαντικών μεταβλητών	88
3.	Quantum Convolutional Neural Networks (QCNN) . . . . .	99
4.	Quantum Generative Neural Networks (QGAN) . . . . .	110
<b>9</b>	<b>Πειραματικά Αποτελέσματα - Συμπεράσματα</b>	<b>118</b>
<b>10</b>	<b>Βιβλιογραφία</b>	<b>121</b>
<b>11</b>	<b>Παράρτημα</b>	<b>125</b>
1.	Dataset Pairplot before Encoding . . . . .	126

2. QCNN - Quantum Circuit . . . . . 127

Η παρούσα διπλωματική εργασία υλοποιήθηκε στο πλαίσιο του μεταπτυχιακού προγράμματος της Σχολής Εφαρμοσμένων Μαθηματικών και Φυσικών Επιστημών, Μαθηματική Μοντελοποίηση στις Σύγχρονες Τεχνολογίες και Χρηματοοικονομικά, στο Εθνικό Μετσόβιο Πολυτεχνείο.

Θα ήθελα να πω ένα τεράστιο ευχαριστώ στους επιβλέποντες καθηγητές μου Γεώργιο Σιόλα και Ανδρέα Σταφιλοπάτη που τον τελευταίο χρόνο με βοηθούν και με καθοδηγούν σε αυτό το νέο ταξίδι της Μηχανικής Μάθησης και των Κβαντικών Υπολογιστών. Από την αρχή της συνεργασίας μας και σε όλη τη διάρκεια της συγγραφής αυτής της εργασίας, ήταν δίπλα μου και η παρουσία τους ήταν καθοριστική, για να ξεπεράσω αρκετές δυσκολίες κατά την εξέλιξή της.

Τέλος, θα ήθελα να ευχαριστήσω τους γονείς μου Ναταλία και Γιώργο και όλους τους κοντινούς μου ανθρώπους που με στήριξαν σε όλη αυτή την προσπάθεια. Αυτή η μελέτη είναι αφιερωμένη σε έναν στενό φίλο και στοχεύει στην ευαισθητοποίηση σχετικά με τις στρεσογόνες συνθήκες που επηρεάζουν τη σημερινή κοινωνία και ιδιαίτερα τους νέους.



# Κεφάλαιο 1

## Abstract

Η παρούσα διπλωματική εργασία εκπονήθηκε στο πλαίσιο του μεταπτυχιακού προγράμματος «Μαθηματική Προτυποποίηση σε Σύγχρονες Τεχνολογίες τη Χρηματοοικονομική», της Σχολής Εφαρμοσμένων Μαθηματικών και Φυσικών Επιστημών του Εθνικού Μετσόβιου Πολυτεχνείου. Η μελέτη που υποστηρίζει τη συγκεκριμένη διατριβή επικεντρώνεται στο να αποδείξει πώς τα Κλασικά Κβαντικά Μοντέλα Βαθιάς Μάθησης μπορούν να χρησιμοποιηθούν για την επίλυση ενός προβλήματος ταξινόμησης. Το σύνολο δεδομένων που επιλέχθηκε για αυτήν την εργασία ήταν το σύνολο δεδομένων QM9 που περιείχε πληροφορίες για μόρια και περιορίστηκε σε περίπου 20 χιλιάδες μόρια που μπορούν ενδεχομένως να χρησιμοποιηθούν για την ταξινόμηση φαρμάκων που μπορούν να χρησιμοποιηθούν ως τρικυκλικά αντικαταθλιπτικά φάρμακα ή για το σχεδιασμό νέων μορίων με βάση αυτό το σύνολο δεδομένων αναφοράς. Ωστόσο, ο κύριος σκοπός αυτής της διατριβής ήταν να δοκιμάσει μοντέλα Quantum Machine Learning σε αυτό το σύνολο δεδομένων και να αποδείξει ότι το κβαντικό ανάλογο των μοντέλων επιτυγχάνει την κβαντική υπεροχή. Με τον όρο κβαντική υπεροχή, εννοούμε το κβαντικό πλεονέκτημα που παρέχεται στην επίλυση υπολογιστικών προβλημάτων, όταν κάποιος χρησιμοποιεί έναν Κβαντικό Υπολογιστή. Προσπάθησα να χρησιμοποιήσω περίπου τις ίδιες παραμέτρους για να μπορώ να συγκρίνω τα μοντέλα. Τα αποτελέσματα που αναλύσαμε έδειξαν ότι όλοι οι κβαντικοί αλγόριθμοι που χρησιμοποιήθηκαν είχαν καλύτερη απόδοση από τους κλασσικούς ανάλογους ή τουλάχιστον στο ίδιο επίπεδο. Τα κβαντικά μοντέλα που δεν ξεπέρασαν τα κλασικά ανάλογα σε ακρίβεια, τα ξεπέρασαν σε χρόνο εκτέλεσης. Παρατηρήσαμε μοντέλα που απέδιδαν στο 1-2% του χρόνου που εκτελούσε το κλασικό ανάλογο τους. Αυτά τα ευρήματα είναι πολύ σημαντικά καθώς ο Quantum Computing και ειδικά οι αλγόριθμοι Quantum ML, είναι ένα πεδίο που αναπτύσσεται ταχέως αλλά εξακολουθεί να βρίσκεται στο στάδιο του προοιμίου του και επομένως δεν υπάρχει πληθώρα πειραματικών δεδομένων. Όπως



περιγράφεται, είναι πολύ δύσκολο να δοκιμαστούν όλα τα παραπάνω σε πραγματικούς κβαντικούς υπολογιστές, καθώς είναι περιορισμένοι, είναι υπανάπτυκτοι και είναι δύσκολο να προσπελαστούν για την εκτέλεση εκτεταμένων μελετών. Παρόλο που τρέξαμε όλους τους αλγόριθμους μας σε προσομοιώσεις κβαντικών υπολογιστών με τη βοήθεια του πακέτου Cirq από την Google, του πακέτου pennylane, το οποίο επικεντρώνεται στο Quantum ML και του πακέτου ανοιχτού κώδικα Qiskit που παρέχεται από την IBM, είναι σημαντικό να γνωρίζουμε τη συμπεριφορά του κβαντικού υπολογιστή για το χρόνο που μπορούμε να δημιουργήσουμε πιο σύνθετες αρχιτεκτονικές για να μπορούμε να δοκιμάσουμε την απόδοσή τους και να μπορούμε να τις εκμεταλλευτούμε.

This thesis was authored under the postgraduate program, "Mathematical Modeling in Cutting Edge Technologies Finance", of the School of Applied Mathematics and Physics Sciences of the National Technical University of Athens. The study supporting the specific thesis is focused on proving how Classical Quantum Deep Learning Models can be used to solve a classification problem. The dataset chosen for this task was the QM9 dataset containing information on molecules and was limited to approximately 20 thousand molecules that can possibly be used to classify drugs that can be used as tricyclic antidepressant drugs or to design new molecules based on this benchmark dataset. However, the main purpose of this thesis was to try Quantum Machine Learning models on this dataset and prove that the quantum analogous of the models achieve the quantum supremacy. By the term quantum supremacy, we mean the quantum advantage provided on solving computational problems, when someone utilises a Quantum Computer. I tried to use approximately the same parameters in order to be able to compare the models. The results that we analysed showed that all quantum algorithms used performed better than their classical analogous or at least on the same level. The quantum models that didn't outperform their classical analogous in accuracy, they outperformed them in run-time. We observed models that performed in 1-2% of the time that their classical analogous performed. These findings are very important as Quantum Computing and especially Quantum ML algorithms, is a field rapidly growing but still in its preamble stage and thus there is no abundance of experimental data. As described, it is very hard to test all the above in real quantum computers as they are limited, they are underdeveloped and are hard to access to run extended studies. Although, we ran all our algorithms on simulations of quantum computers with the help of Cirq package by Google, the pennylane package, which is focused in Quantum ML and the qiskit open source package provided by IBM, it is important to know the behavior of the quantum computers for when the time that we can build more complex architectures to be able to test their performance and be able to exploit them.



## Κεφάλαιο 2

### Εισαγωγή

Σκοπός αυτής της διπλωματικής εργασίας είναι να ανακαλύψω πώς μπορεί να επεκταθεί το πεδίο της ανακάλυψης φαρμάκων με τη χρήση αλγορίθμων Quantum Machine Learning. Πιο συγκεκριμένα, θα μελετήσω πώς ορισμένες γνωστές τεχνικές μπορούν να εφαρμοστούν σε σύνολα ιατρικών δεδομένων και θα προσπαθήσω να ορίσω κάποια συμπεράσματα για την απόδοση των προγνωστικών μοντέλων, ώστε να καταλάβω ποια είναι πιο κατάλληλη για συγκεκριμένες εργασίες, όπως η ανακάλυψη φαρμάκων.

Οι συνήθεις μεθοδολογίες που χρησιμοποιούνται για την ανακάλυψη φαρμάκων είναι μέσω της πειραματικής διαδικασίας και απαιτούν πολλές δοκιμές, οι οποίες συνίστανται σε αυτές τις μεθόδους χρονοβόρες και πολύ δαπανηρές. Αυτά μας οδήγησαν να χρησιμοποιήσουμε σύγχρονα εργαλεία, όπως προσομοιώσεις και προγνωστικά μοντέλα για να εκμεταλλευτούμε την αφθονία των δεδομένων και να χαρτογραφήσουμε πώς συμπεριφέρονται τα υλικά ναρκωτικών. Πρέπει να ξεκινήσει κανείς τη μελέτη του από τα χαρακτηριστικά των ατόμων και των μορίων έως ολόκληρες βάσεις δεδομένων υγειονομικής περίθαλψης για το πώς να τα συνδυάσει για να βρει θεραπεία για τις πιο κοινές ασθένειες. Ωστόσο, οι σημερινοί κλασικοί υπολογιστές, ακόμη και οι υπερυπολογιστές, μπορούν να προσομοιώσουν μόνο σχετικά απλά μόρια, περιορίζοντας τις δυνατότητές μας να μελετάμε πιο πολύπλοκα συστήματα. Ως εκ τούτου, το επίκεντρο της έρευνας είναι η επίτευξη της λεγόμενης κβαντικής υπεροχής. Αυτό μπορεί να επιτευχθεί με τη χρήση κβαντικών υπολογιστών, οι οποίοι εκμεταλλεύονται τους νόμους της κβαντικής μηχανικής για την επεξεργασία πληροφοριών. Οι κβαντικοί υπολογιστές χρησιμοποιούν κβαντικά bit, που ονομάζονται qubits, τα οποία μπορούν να κωδικοποιήσουν μηδέν και ένα ταυτόχρονα. Αυτό το φαινόμενο ονομάζεται υπέρθεση μιας κβαντικής κατάστασης.

Η Κβαντική Μηχανική Εκμάθηση χρησιμοποιεί κβαντικούς αλγόριθμους για την εκτέλεση σύνθετων εργασιών μηχανικής μάθησης. Το Quantum ML μας επιτρέπει να μεταφράσουμε αλγόριθμους ML σε ένα κβαντικό κύκλωμα που μπορεί να λάβει υπόψη την αρχιτεκτονική του κβαντικού υπολογιστή και να λειτουργεί πιο αποτελεσματικά. Οι περισσότεροι από αυτούς τους αλγόριθμους είναι φυσικά υβριδικά μοντέλα που χρησιμοποιούν συνδυασμό κλασικών και κβαντικών μοντέλων. Εταιρείες και ερευνητές έχουν πειραματιστεί περισσότερο σε αυτούς τους υβριδικούς αλγόριθμους, γιατί αυτή τη στιγμή είναι η πιο προσιτή επιλογή. Οι υβριδικοί αλγόριθμοι συνδυάζουν την ικανότητα αποθήκευσης ενός κλασικού υπολογιστή, η οποία είναι μεγαλύτερη από αυτή ενός κβαντικού υπολογιστή, με την υπολογιστική ισχύ μιας κβαντικής μηχανής.

## Κεφάλαιο 3

# Μέθοδοι Σχεδιασμού Φαρμάκων

Ο προσδιορισμός στόχου, ο μοριακός σχεδιασμός, οι προκλινικές έρευνες και οι κλινικές δοκιμές θεωρούνται όλα στάδια στον αγωγό ανάπτυξης φαρμάκων. Η αναζήτηση νέων φαρμακευτικών προϊόντων μπορεί να θεωρηθεί ως πλοήγηση στο χημικό διάστημα, που είναι η συλλογή όλων των οργανικών μορίων, και η πλοήγηση σε ανεξερεύνητο χημικό χώρο εμπίπτει στην κατηγορία του σχεδιασμού φαρμάκων "de novo". Για το εύρος αυτής της εργασίας, επιλέξαμε να διερευνήσουμε πώς οι πιο γνωστές τεχνικές μηχανικής μάθησης, όπως οι Variational Autoencoders (VAEs), τα Generative Adversarial Networks (GANs) και τα Recurrent Neural Networks (RNNs), συγκρίνονται μεταξύ τους και μπορούν να χρησιμοποιηθούν για την εκμάθηση λανθάνουσας αναπαράστασης μορίων και τη δημιουργία μεγάλου αριθμού υποψηφίων φαρμάκων για περαιτέρω έλεγχο υψηλής απόδοσης. Για να ξεπεράσουμε τη δυσκολία της διάστασης, θα χρειαστεί να χρησιμοποιήσουμε το κβαντικό ισοδύναμο όλων των προαναφερθέντων μοντέλων για να δειγματίσουμε μοριακές ενώσεις από τμήματα του χημικού χώρου που μπορεί να είναι της τάξης των  $10^{60}$ . Τα GAN, για παράδειγμα, βρίσκουν θεραπευτικούς υποψηφίους παράγοντας μοριακές δομές που υπακούουν σε χημικές και φυσικές ιδιότητες και έχουν υψηλή συγγένεια για σύνδεση με τον υποδοχέα μιας νόσου στόχου. Αντίθετα, ένα QGAN, ένα σύστημα απόδοσης qubit με υβριδικό generator, μπορεί να χρησιμοποιηθεί για να μάθει πλουσιότερες αναπαραστάσεις μορίων αναζητώντας εκθετικά τεράστιους χημικούς χώρους με λίγα qubit πιο αποτελεσματικά από ένα κλασικό GAN. Σε αυτούς τους τύπους μοντέλων χρησιμοποιούνται συνήθως μια υβριδική κβαντικό generator που υποστηρίζει ποικίλους αριθμούς qubits και στρώματα κβαντικών κυκλωμάτων και ένας συμβατικός discriminator.

Το Quantum GAN είναι μια από τις κύριες εφαρμογές των βραχυπρόθεσμων κβαντικών υπολογιστών λόγω της ισχυρής εκφραστικής του δύναμης στις κατανομές δεδομένων εκμάθησης ακόμη και με πολύ λιγότερες παραμέτρους σε σύγκριση με τα

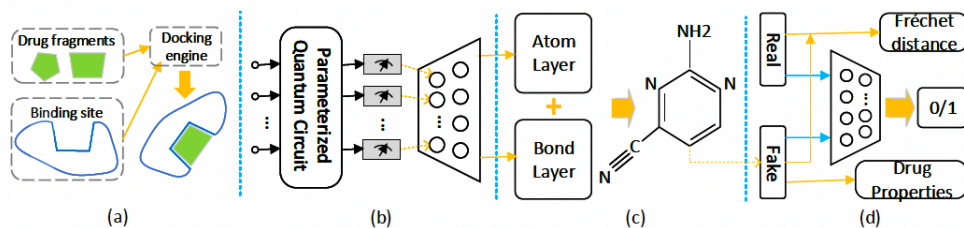
κλασικά GAN. Ωστόσο, το κβαντικό νευρωνικό δίκτυο βρίσκεται ακόμα στο στάδιο της γέννησής του λόγω των περιορισμών του qubit στους θορυβώδεις κβαντικούς υπολογιστές. Λαμβάνοντας υπόψη το συγκεκριμένο έργο της ανακάλυψης φαρμάκων, διερευνούμε πιθανά κβαντικά πλεονεκτήματα τόσο για παραγωγικά όσο και για προγνωστικά μοντέλα για τους ακόλουθους λόγους:

- ▶ Η εξερεύνηση παραμέτρων πύλης στον χώρο Hilbert είναι διαφορετική από την εξερεύνηση παραμέτρων νευρωνικού δικτύου
- ▶ Δεδομένης μιας χημικής περιοχής με αφθονία μορίων, η εγγενής πιθανολογική φύση των κβαντικών συστημάτων βοηθά στη δημιουργία πιο διαφορετικών και νέων μορίων που περιβάλλουν αυτήν την περιοχή.

Το Quantum ML περιλαμβάνει βελτιστοποίηση παραμέτρων παραμετροποιημένων κβαντικών κυκλωμάτων για να επιτευχθεί μια επιθυμητή σχέση εισόδου-εξόδου.

Μια άλλη δημοφιλής εφαρμογή της QML στην ανακάλυψη φαρμάκων είναι η χρήση νευρωνικών δικτύων γραφικών σε συνδυασμό με βιβλιοθήκες όπως το RDKit ή το DeepChem για την προεπεξεργασία τους. Μερικά δημοφιλή πειράματα είναι η πρόβλεψη της τοξικότητας και της διαλυτότητας των μορίων. Η πρώτη είναι μια εργασία ταξινόμησης, σε αντίθεση με τη δεύτερη, που είναι μια εργασία παλινδρόμησης. Έτσι, βλέπουμε ότι η πρόβλεψη κάθε μιας από τις ιδιότητες του μορίου μπορεί να απαιτεί διαφορετικούς τύπους μοντέλων που ταιριάζουν καλύτερα στις συνθήκες κάθε προβλήματος. Η κοινή θεωρία πίσω από τις περισσότερες ιδιότητες είναι η συναρτησιακή θεωρία της πυκνότητας, η οποία χρησιμοποιείται για τον υπολογισμό ορισμένων μοριακών ιδιοτήτων όπως οι ενέργειες και οι ενθαλπίες ως στόχοι.

Η παραδοσιακή μέθοδος ανακάλυψης φαρμάκων περιλαμβάνει τον προσδιορισμό των χαρακτηριστικών ενός υποδοχέα και των ενώσεων που μπορούν να διευκολύνουν τη σύνδεσή του. Και πάλι, αυτό γίνεται χρησιμοποιώντας τη συναρτησιακή θεωρία πυκνότητας. Μια μέθοδος ομοιότητας δακτυλικών αποτυπωμάτων μπορεί επίσης να χρησιμοποιηθεί για τον εντοπισμό πιθανών στόχων άλλων μορίων που είναι ενεργά έναντι του υποδοχέα. Το μοντέλο MolGAN χρησιμοποιεί ένα παραγωγικό ανταγωνιστικό δίκτυο για να μάθει και να δημιουργήσει μόρια παρόμοια με φάρμακα με βάση το σύνολο δεδομένων QM9. Για τη βελτίωση της ποιότητας των παραγόμενων ενώσεων και τη σταθεροποίηση της εκπαίδευσης GAN, χρησιμοποιείται ένα δίκτυο επιβράβευσης που βασίζεται στη χημική εγκυρότητα, το RDKit. Αν θέλουμε να χρησιμοποιήσουμε το Quantum ML, μπορούμε να χρησιμοποιήσουμε ένα υβριδικό κβαντικό-κλασικό μοντέλο, το οποίο είναι μια παραλλαγή του MolGAN. Αυτό το μοντέλο παρουσιάζουμε παρακάτω:



Σχήμα 3.1: QGAN

Το προτεινόμενο μοντέλο, από τη βιβλιογραφία, έχει αποδειχθεί ότι είναι αποδοτικό σε qubit και περιλαμβάνει ένα υβριδικό generator και έναν κλασικό discriminator. Το μοντέλο μαθαίνει αποτελεσματικά τις κατανομές μορίων με βάση το MolGAN. Όπως μπορούμε να δούμε στο 3.1:

- ▶ στοιχείο (α) η εγκυρότητα των μορίων που δημιουργούνται εξαρτάται από το εάν έχουν υψηλή συγγένεια προς τις θέσεις δέσμευσης του υποδοχέα.
- ▶ Στο δεύτερο μέρος (β), εμφανίζεται το κβαντικό στάδιο και
- ▶ Στο τρίτο τμήμα (γ) εφαρμόζεται ένα στρώμα ατόμου και ένα στρώμα δεσμού για τη δημιουργία συνθετικών μοριακών γραφημάτων.
- ▶ Τέλος, στην ενότητα (δ) μια παρτίδα πραγματικών μορίων από το σύνολο δεδομένων εκπαίδευσης (QM9) και μια παρτίδα συνθετικών μορίων που παράγονται από το στάδιο (γ) τροφοδοτούνται στον κλασικό discriminator για πραγματική πρόβλεψη και τον υπολογισμό της βαθμολογίας.

Στο συγκεκριμένο χαρτί χρησιμοποίησαν ως κριτήριο μοριοδότησης την απόσταση Fréchet. Μετά την παραπάνω διαδικασία, όλα τα αποτελέσματα σχετικά με τις ιδιότητες του φαρμάκου αξιολογούνται χρησιμοποιώντας τη συσκευασία RDKit.

Εάν προτιμάται μια πιθανολογική αναζήτηση, ο υποδοχέας μπορεί να αντιμετωπιστεί ως εικόνα  $N \times N \times N$  με 8 κανάλια που αντιστοιχούν στους τύπους ατόμων. Ο συνδετήρας θα θεωρηθεί ως ένας πίνακας γειτονίας  $N \times N$  με 6 κανάλια τύπου δεσμού και ένα διάνυσμα τύπου ατόμου  $N$  μήκους με 7 κανάλια τύπου ατόμου. Σε αυτό το σημείο, είναι σημαντικό να αναφέρουμε ότι η μέγιστη τιμή είναι  $N = 32$  άτομα, γεγονός που καθιστά αυτή την προσέγγιση εφικτή για τη δημιουργία μεγάλων μοριακών γραφημάτων. Για να συνεχίσουμε, χρησιμοποιήθηκαν δύο μεταβλητοί αυτοκωδικοποιητές, ένας για τον θύλακα του υποδοχέα και ένας για

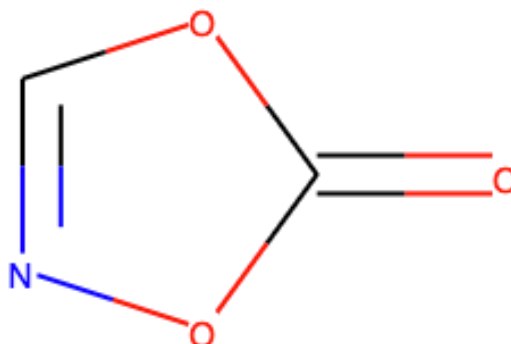


τον συνδέτη. Θα χρησιμοποιηθούν για να ταιριάζουν με ζεύγη τσέπης υποδοχέα και συνδέτη. Αποτελούνται από τρισδιάστατα συνελκτικά στρώματα που συμπιέζουν την αρχική εικόνα σε μια λανθάνουσα αναπαράσταση 32 στοιχείων πριν την αποσυμπιέσουν ξανά στην αρχική εικόνα τσέπης του υποδοχέα. Ένα αμφίδρομο δίκτυο αντιστοίχισης συνδέει τα δύο VAE, μετατρέποντας τη λανθάνουσα αναπαράσταση του θύλακα του υποδοχέα σε λανθάνουσα αναπαράσταση του συνδέτη και αντίστροφα. Η συνάρτηση ενεργοποίησης που χρησιμοποιήθηκε σε αυτή τη μελέτη ήταν ένα LeakyReLU σε συνδυασμό με φασματική κανονικοποίηση για την αποφυγή έκρηξης/εξαφάνισης διαβαθμίσεων. Το Hybrid VAE δημιουργείται με την εισαγωγή κβαντικών στρωμάτων στο νευρωνικό δίκτυο σε συγκεκριμένα σημεία. Το σημείο πόνου που ανακαλύφθηκε κατά τη διάρκεια αυτής της πειραματικής διαδικασίας ήταν ότι το στάδιο της κβαντικής ενσωμάτωσης περιορίζει την κβαντική υπεροχή του μοντέλου. Η κβαντική ενσωμάτωση που χρησιμοποιήθηκε ήταν η κωδικοποίηση γωνίας, ακολουθούμενη από μια μέθοδο επαναφόρτωσης δεδομένων για τη μέτρηση διαφορετικών τεχνικών για την προετοιμασία κβαντικής κατάστασης.

## 1. Πακέτο RDKit

Το RDKit είναι ένα δημοφιλές εργαλείο στη χημειοπληροφορική που είναι γραμμένο σε C++ και Python και επιτρέπει στους χρήστες να υπολογίζουν απευθείας τις ομοιότητες των μορίων. Οι ενσωματώσεις γραφημάτων θα μπορούσαν να χρησιμοποιηθούν για την εκτέλεση μιας πιο ενδεδειγμένης αναζήτησης ομοιότητας σε μόρια χρησιμοποιώντας κλασικούς αλγόριθμους. Για πολλές εργασίες, όπως η δημιουργία ενός μορίου, αυτό το πακέτο χρησιμοποιεί συμβολοσειρές SMILES (Simplified Molecular Input Line Entry System). Οι συμβολοσειρές SMILES περιέχουν ένα γράμμα που αντιπροσωπεύει ένα άτομο καθώς και άλλα σύμβολα που περιέχουν πληροφορίες σχετικά με τους δεσμούς μεταξύ των ατόμων. Ένα άλλο έργο στο οποίο υπερέχει το RDKit είναι η φόρτωση και η οπτικοποίηση μορίων. Όταν οραματιζόμαστε ένα μόριο, παίρνουμε μια εικόνα που μοιάζει με την παρακάτω:

smiles: O=c1ocno1 -357.16757



Σχήμα 3.2:

Όλες οι υποδομές του μορίου που περιέχονται στο συγκεκριμένο χαμόγελο φαίνονται στην παραπάνω εικόνα. Αυτή είναι εξαιρετικά χρήσιμη πληροφορία επειδή η συμπεριφορά ενός μορίου ως φαρμάκου θα μπορούσε να προσδιοριστεί από ορισμένες υποδομές που περιέχονται σε αυτό το SMILE. Μπορούμε να συμπεράνουμε από αυτό ότι τα μόρια με παρόμοιες βασικές υποδομές είναι πιθανό να συμπεριφέρονται παρόμοια. Όταν συγκρίνουμε διαφορετικά μόρια, μπορούμε να χρησιμοποιήσουμε το RDKit για να ταιριάξουμε αυτά τα μέρη. Η ομοιότητα Tanimoto είναι ένα κριτήριο που παρέχεται από το RDKit και μπορεί να χρησιμοποιηθεί για τη σύγκριση τους. Αυτή είναι μια δημοφιλής μέτρηση που χρησιμοποιεί αντιστοίχιση υποδομής.

Μετά τη φόρτωση και την οπτικοποίηση των μορίων, μπορεί κανείς να χρησιμοποιήσει το RDKit για να αναζητήσει ομοιότητες μεταξύ τους. Αυτό θα μπορούσε να επιτευχθεί ξεκινώντας με ένα μόριο που γνωρίζουμε τις ιδιότητες και τη συμπεριφορά του και αναζητώντας ομοιότητες με μόρια σε μια μεγάλη βάση δεδομένων, όπως το QM9 και που μπορεί να χρησιμοποιηθεί ως βανίλια. Εάν οι πληροφορίες και οι ιδιότητες που παρέχονται από το SMILES κατά την εργασία με το RDKit είναι ανεπαρκείς, μπορεί να χρησιμοποιηθεί το SMARTS. Το SMARTS είναι μια γενίκευση του SMILES που μπορεί να σας πει εάν ένα μόριο έχει ή όχι δακτυλίους. Ωστόσο, τα μοριακά δακτυλικά αποτυπώματα χρησιμοποιούνται για την κωδικοποίηση της δομής ενός μορίου σε κάθε περίπτωση, όπως συζητάμε αλλού σε αυτή τη διατριβή. Αυτά χρησιμοποιούνται κυρίως για τη σύγκριση μορίων υποδομών, χαρακτηριστικών και της συντριπτικής πλειοψηφίας των πληροφοριών που είναι αποθηκευμένες σε ένα μόριο. Σε ένα μόριο, κρατούν κάθε σχέδιο συγκεκριμένου μεγέθους.

## Κεφάλαιο 4

# Σύνολο Δεδομένων Βαθμονόμησης και Εξερεύνηση των Δεδομένων

Το κύριο ζήτημα με αυτή τη διατριβή ήταν η έλλειψη πειραματικών δεδομένων και η έλλειψη ποικιλίας σε διαφορετικούς τύπους υλοποιήσεων. Έτσι, ένα από τα πιο σημαντικά βήματα που ακολουθήθηκαν στη διαδικασία μας ήταν να επιλέξουμε ένα πολύ γνωστό σύνολο δεδομένων που θα μπορούσε να χρησιμοποιηθεί για τη βαθμονόμηση του μοντέλου μας. Η λανθασμένη επιλογή του συνόλου δεδομένων θα μπορούσε να οδηγήσει σε λανθασμένα αποτελέσματα και ψευδή συμπεράσματα στο πείραμά μας.

Για το σκοπό αυτής της διατριβής, χρησιμοποιήσαμε ένα σύνολο δεδομένων που δημιουργήθηκε από τη βάση δεδομένων Quantum Machines 9 (QM9). Αυτό το σύνολο δεδομένων περιέχει περισσότερες από 134 χιλιάδες σειρές πληροφοριών και μας παρέχει κβαντικές χημικές παραμέτρους για έναν σχετικό, συνεπή και εκτεταμένο χημικό χώρο οργανικών ενώσεων μικρού μεγέθους. Αυτή η βάση δεδομένων μπορεί να χρησιμοποιηθεί για τη συγκριτική αξιολόγηση υπαρχουσών μεθόδων, την ανάπτυξη νέων προσεγγίσεων όπως η υβριδική κβαντομηχανική/μηχανική εκμάθηση και τη συστηματική αναγνώριση των συνδέσεων δομής-ιδιότητας, όπως αναφέρεται στην τεκμηρίωσή της. Οι ιδιότητες της κβαντικής χημείας που αναφέρονται παραπάνω περιλαμβάνουν γεωμετρικές, ενεργητικές, ηλεκτρονικές και θερμοδυναμικές ιδιότητες. Πιο αναλυτικά, αυτή η βάση δεδομένων μπορεί να χρησιμοποιηθεί για να αναφέρει γεωμετρικές ελάχιστης ενέργειας, σχετικές αρμονικές συχνότητες, διπολικές ροπές, πολωσιμότητα, καθώς και ενέργειες εξατομίκευσης, ενθαλπίες και ελεύθερες ενέργειες. Εκτός από τις πληροφορίες δομής, το QM9 ενσωματώνει SMILES από το GDB9 και για χαλαρή γεωμετρία, καθώς και InChI για GDB9 και για χαλαρή γεωμετρία. Το σύνολο δεδομένων QM9 βασίζεται στο GDB9, την

πιο γενική βάση δεδομένων μοριακών κβαντικών υπολογισμών. Ωστόσο, το υψηλό υπολογιστικό κόστος εμποδίζει τη χρήση του σε τακτική βάση για εκτεταμένη χημική εξερεύνηση του διαστήματος.

Τώρα, για να εξηγήσουμε περαιτέρω τις παραπάνω πληροφορίες. Με τη λέξη SMILES αναφερόμαστε στο "Simplified Molecular Input Line Entry System", το οποίο είναι μια χημική αναπαράσταση που χρησιμοποιεί μια ακολουθία χαρακτήρων για να κωδικοποιήσει το μόριο. Το SMILE περιλαμβάνει κυρίως σύμβολα ατόμων και τους δεσμούς τους, επιπλέον των συντακτικών κανόνων που δηλώνουν την εγκυρότητα κάθε καταχώρησης μιας συμβολοσειράς. Τα χαρακτηριστικά που προέρχονται από την αναπαράσταση SMILES ονομάζονται τοπολογικοί περιγραφείς, καθώς η διαδικασία εξαγωγής χαρακτηριστικών βασίζεται στα αρχεία μορίων. Τέλος, τα πλεονεκτήματα της χρήσης SMILES είναι

- ▶ μια ερμηνεία των μορίων που είναι πιο φιλική προς τον χρήστη,
- ▶ είναι απλό να κωδικοποιήσουν το μοριακό γράφημα ή το αντίστροφο,
- ▶ χρησιμοποιούνται ευρέως και είναι ένας δημοφιλής τρόπος αναπαράστασης μορίων και
- ▶ αποτελούν μια αναπαράσταση που είναι ταχύτερη στον υπολογισμό.

Αν κάποιος θέλει να μελετήσει τα SMILES και τα σύνολα δεδομένων όπως το QM9 και να προσεγγίσει αυτά τα προβλήματα, πρώτα πρέπει να ακολουθήσει τη μεθοδολογία προεπεξεργασίας. Ευτυχώς, για τις περισσότερες περιπτώσεις αυτό το βήμα καλύπτεται από βιβλιοθήκες που χρησιμοποιούνται σε έργα βαθιάς μάθησης όπως το Deepchem ή το RDKit, όπως αναφέρθηκε στο προηγούμενο κεφάλαιο που είναι αφιερωμένο στην περιγραφή του πακέτου RDKit. Προσφέρουν χρήσιμες λειτουργίες για μοριακά δεδομένα, όπως φορτωτές δεδομένων, discriminator, χαρακτηριστικοί, μετρήσεις, ακόμη και μοντέλα GNN. Αυτές οι ρουτίνες σας επιτρέπουν να χρησιμοποιείτε τα SMILES ως αναπαραστάσεις συμβολοσειρών της δισδιάστατης ή τρισδιάστατης δομής του μορίου. Είναι σημαντικό να θυμάστε ότι αυτό το βήμα μεταφράζει οποιοδήποτε μόριο σε μια συγκεκριμένη συμβολοσειρά που είναι σχεδόν σίγουρα μοναδική και μπορεί να μεταφραστεί ξανά στη δισδιάστατη δομή. Επιπλέον, διάφορα μόρια μπορεί να αντιστοιχιστούν στην ίδια συμβολοσειρά SMILES, μειώνοντας την απόδοση του μοντέλου. Τα δυαδικά διανύσματα που λαμβάνονται από τα SMILES χρησιμοποιούνται στη συνέχεια για να αναπαραστήσουν εάν υπάρχει ή όχι μια συγκεκριμένη υποδομή του μορίου και αυτοί οι φορείς αναφέρονται ως δακτυλικά αποτυπώματα.

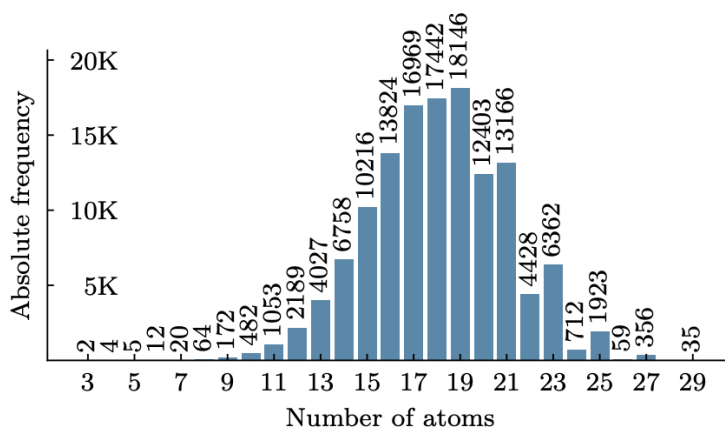
Τώρα, με τη χρήση του πακέτου RDKit και κάποια βασική ανάλυση δεδομένων, μπορούμε να λάβουμε μερικές πολύ σημαντικές πληροφορίες για το σύνολο

δεδομένων μας. Πρώτον, μπορεί κανείς να δει τη συχνότητα με την οποία εμφανίζεται ο αριθμός των ατόμων στο σύνολο δεδομένων μας. Αρχικά, μπορούμε να εκτυπώσουμε την περιγραφή κάθε ετικέτας του συνόλου δεδομένων που στην πραγματικότητα κάθε ετικέτα απεικονίζει σε μία ιδιότητα.

I.	Property	Unit	Description
1	tag	-	"gdb9"; string constant to ease extraction via grep
2	index	-	Consecutive, 1-based integer identifier of molecule
3	A	GHz	Rotational constant A
4	B	GHz	Rotational constant B
5	C	GHz	Rotational constant C
6	mu	Debye	Dipole moment
7	alpha	Bohr <sup>3</sup>	Isotropic polarizability
8	homo	Hartree	Energy of Highest occupied molecular orbital (HOMO)
9	lumo	Hartree	Energy of Lowest occupied molecular orbital (LUMO)
10	gap	Hartree	Gap, difference between LUMO and HOMO
11	r2	Bohr <sup>2</sup>	Electronic spatial extent
12	zpve	Hartree	Zero point vibrational energy
13	U0	Hartree	Internal energy at 0 K
14	U	Hartree	Internal energy at 298.15 K
15	H	Hartree	Enthalpy at 298.15 K
16	G	Hartree	Free energy at 298.15 K
17	Cv	cal/(mol K)	Heat capacity at 298.15 K

Σχήμα 4.1: Description of each property

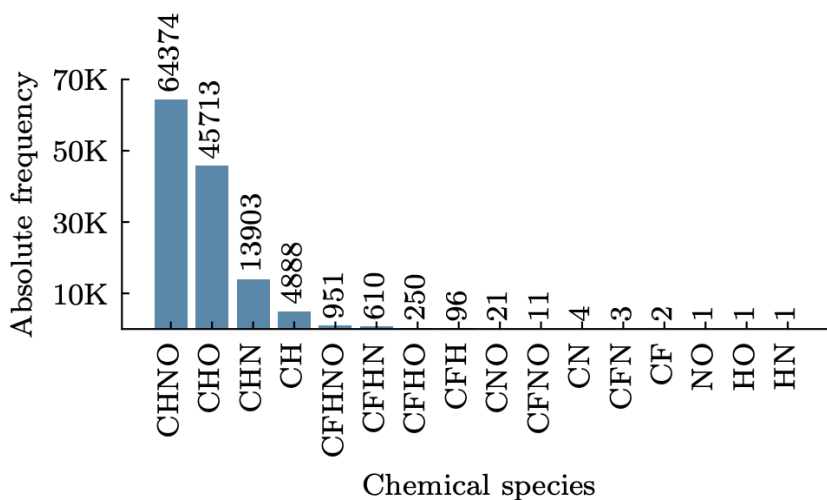
Στη συνέχεια μπορούμε να σχεδιάσουμε την κατανομή με βάση τον αριθμό των ατόμων σε κάθε μόριο και με βάση τον αριθμό των μορίων ανά χημικό είδος.



Σχήμα 4.2: Atoms Histogram

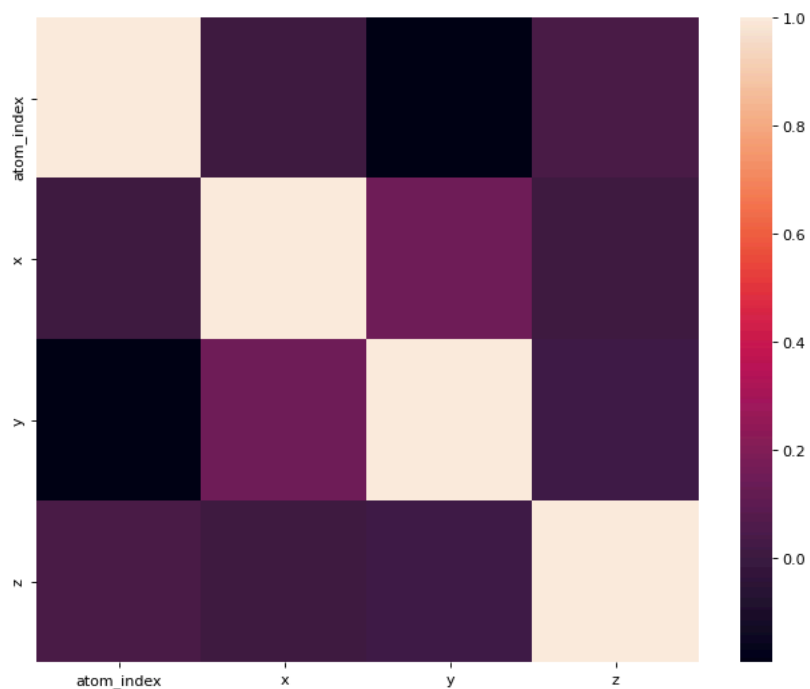
Κάποιος μπορεί να υποθέσει ότι η κατανομή που εμφανίζεται μπορεί να προσρμοστεί είναι μια Gaussian συνάρτηση με κεντρικό μέσο όρο 19 ατόμων και ότι

τα όρια είναι [3,29].



Σχήμα 4.3: Histogram based on the chemical species

Τώρα, απλώς για να απλοποιήσουμε τους υπολογισμούς μας, παίρνουμε ένα υποσύνολο και κάνουμε κάποια εξερεύνηση δεδομένων για να κατανοήσουμε περαιτέρω το σύνολο δεδομένων μας. Έτσι, αν θέλουμε να δούμε τη συσχέτιση των χωρικών μας δεδομένων σε συνδυασμό με τον δείκτη ατόμου, σχεδιάζουμε την παράσταση του θερμικού χάρτη παρακάτω.

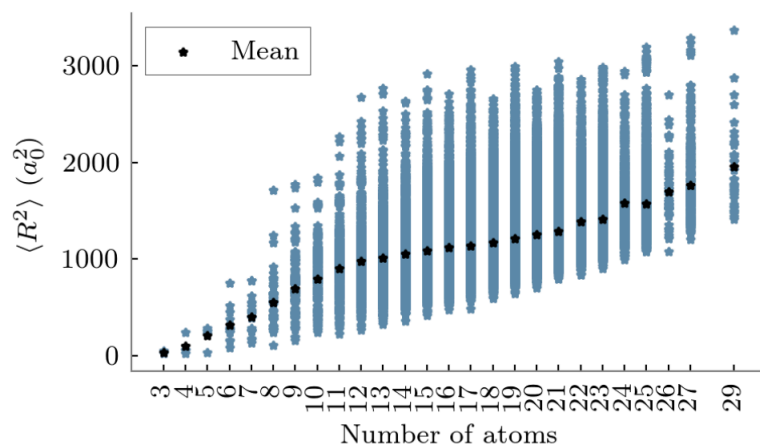


Σχήμα 4.4: Correlation of Spatial Data

και στη συνέχεια στον παρακάτω πίνακα παρουσιάζουμε κάποια βασική στατιστική ανάλυση σε αυτό το υποσύνολο υπολογίζοντας τη μέση τιμή, την τυπική απόκλιση κλπ.

	<b>Atom Index</b>	<b>x</b>	<b>y</b>	<b>z</b>
<b>count</b>	2.35887e+06	2.35887e+06	2.35887e+06	2.35887e+06
<b>mean</b>	9.757255e+00	9.495981e-02	-3.335625e-01	6.239050e-02
<b>std</b>	5.592444e+00	1.655403e+00	1.989328e+00	1.445876e+00
<b>min</b>	0.0000000e+00	-9.234889e+00	-9.933938e+00	-9.134765e+00
<b>25%</b>	4.0000000e+00	-8.746228e-01	-1.826097e+00	-8.424758e-01
<b>50%</b>	9.0000000e+00	5.183962e-02	-4.034906e-01	1.092888e-02
<b>75%</b>	1.3000000e+00	1.116163e+00	1.373848e+00	9.393901e-01
<b>max</b>	2.8000000e+00	9.382240e+00	1.018196e+01	7.8894733e+00

Εδώ παρουσιάζω ένα διάγραμμα διασποράς που δείχνει την ηλεκτρονική χωρική έκταση  $\langle R^2 \rangle$  με τον αριθμό των ατόμων.



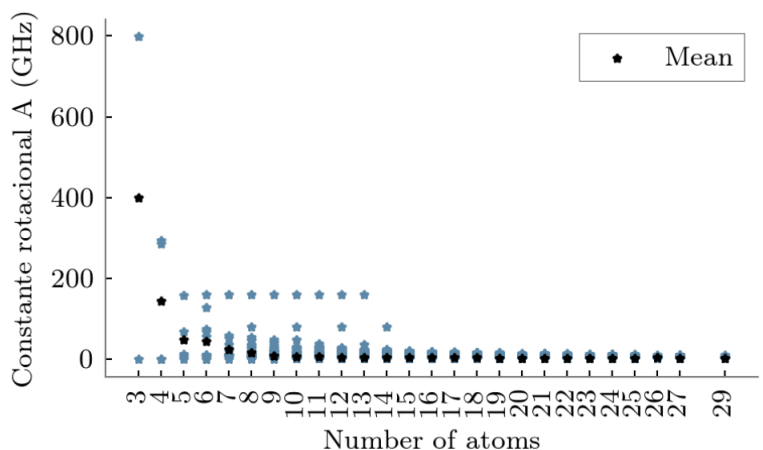
Σχήμα 4.5: Electronic spatial extent  $\langle R^2 \rangle$

Στη συνέχεια, μπορούμε εύκολα να υπολογίσουμε την αφθονία των ατόμων στο επιλεγμένο υποσύνολο:

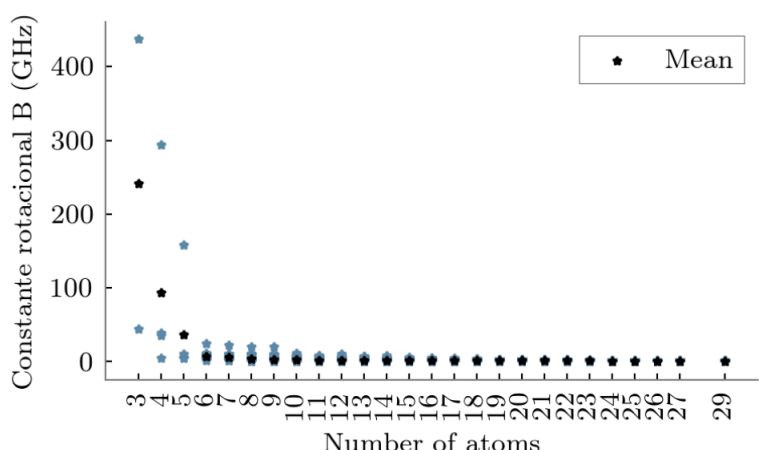
H	51.231456%
C	35.262954%
O	7.766499%
N	5.612082%
F	0.127010%

Για να συνεχίσουμε, δημιουργούμε μερικά διαγράμματα διασποράς για τις σταθερές περιστροφής:

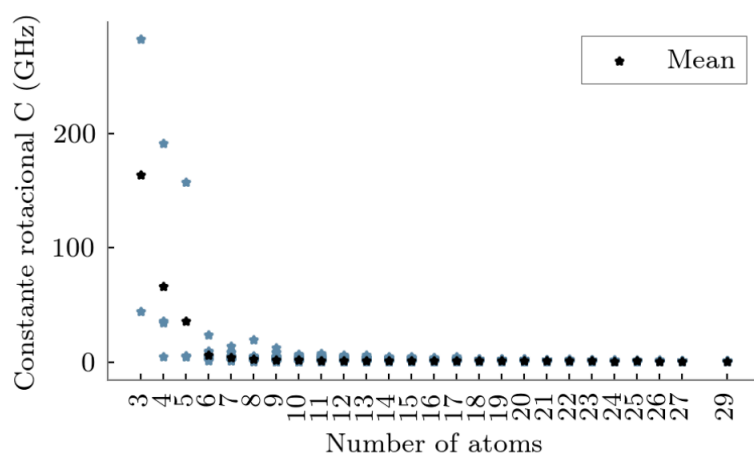




Σχήμα 4.6: Scatter plot for A rotational constant

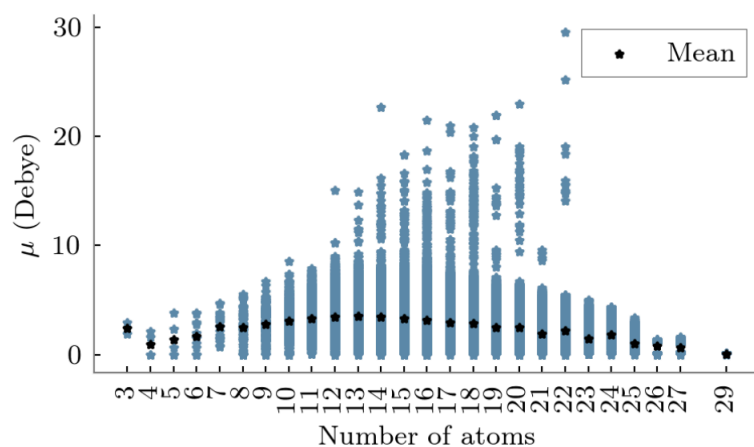


Σχήμα 4.7: Scatter plot for B rotational constant



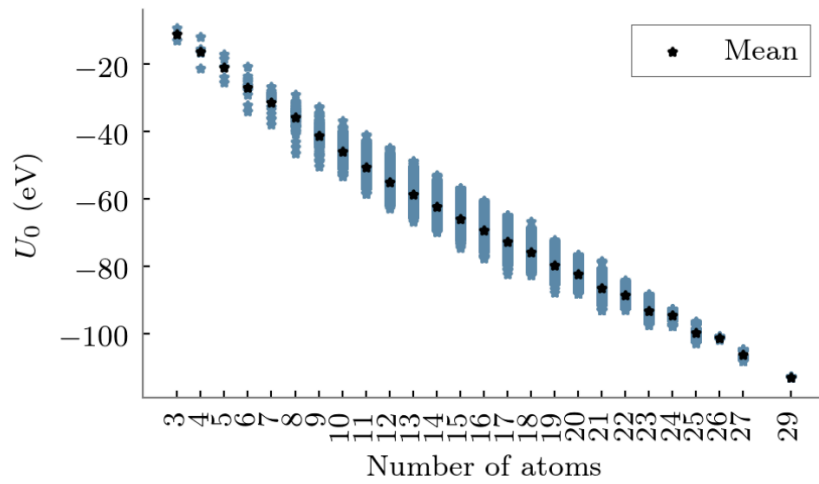
Σχήμα 4.8: Scatter plot for C rotational constant

Στη συνέχεια δημιουργώ ένα παρόμοιο διάγραμμα αντικαθιστώντας τη διπολική ροπή στον άξονα y.

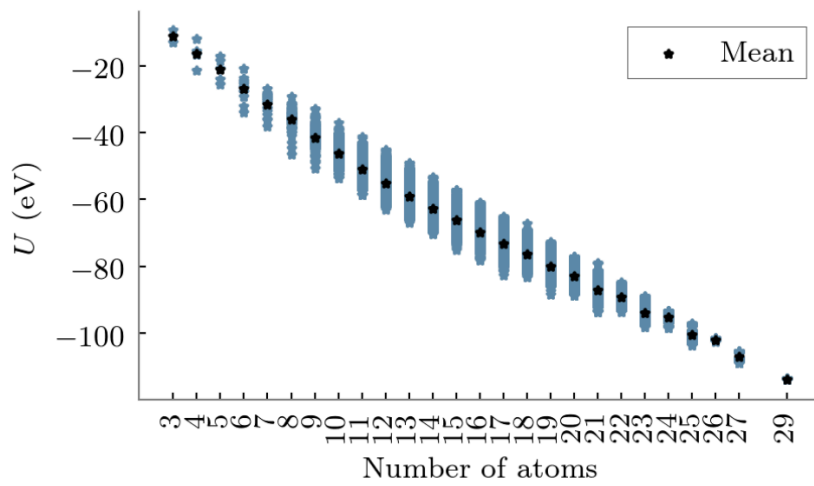


Σχήμα 4.9: Dipole moment vs Number of atoms

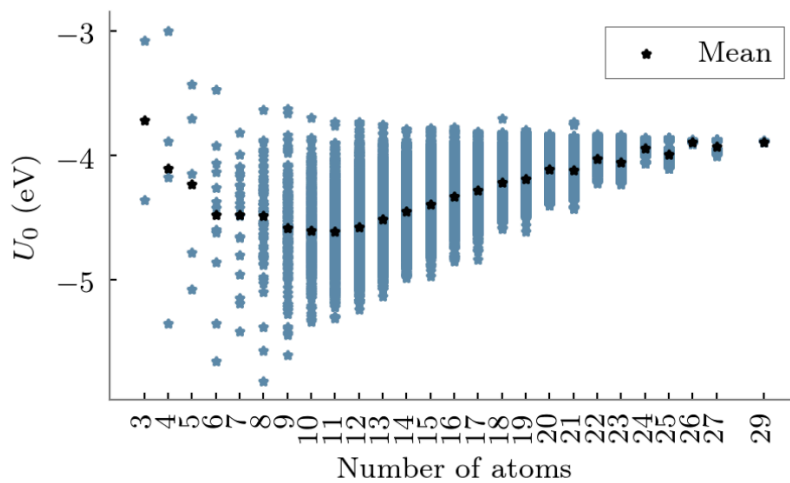
Τέλος, εμφανίζω μερικά διαγράμματα ενέργειας και ενθαλπίας καθώς μπορούμε να εξαγάγουμε πολλές υποθέσεις για το σύστημα κάθε μορίου από αυτές τις παραμέτρους.



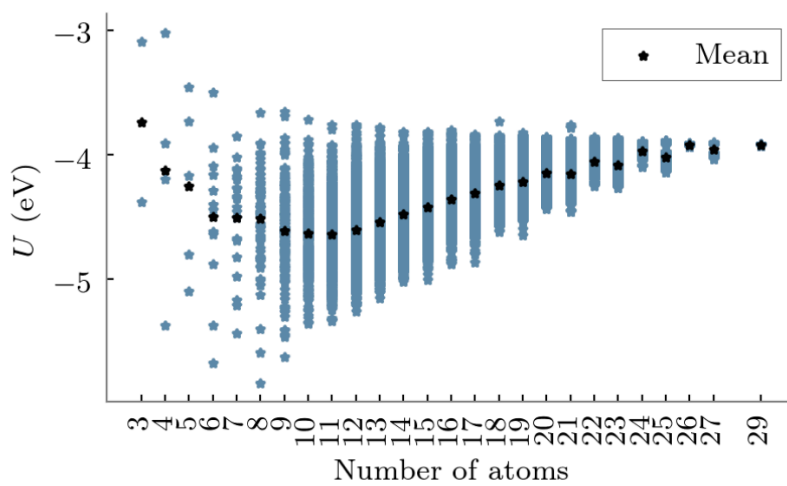
Σχήμα 4.10: Internal Energy at 0K by the number of atoms



Σχήμα 4.11: Internal Energy at 298.15K by the number of atoms



Σχήμα 4.12: Internal Energy at 0K per atom by the number of atoms



Σχήμα 4.13: Internal Energy at 298.15K per atom by the number of atoms

## Κεφάλαιο 5

# Μοντέλα Νευρωνικών Δικτύων για Κλασσικούς Υπολογιστές

Για αυτό το συγκεκριμένο πείραμα αποφασίσαμε να χρησιμοποιήσουμε διάφορες παραλλαγές των Convolutional Neural Networks, αλλά ο μόνος τύπος νευρωνικού δικτύου που είχε πιο νόημα στη χρήση και που ελπίζαμε ότι θα ταίριαζε καλύτερα στην περίπτωση χρήσης μας, ήταν η κατηγορία των Graph Neural Networks. Όπως αναφέρθηκε σε προηγούμενα κεφάλαια, το σύνολο δεδομένων μας περιέχει κυρίως τις γεωμετρικές ιδιότητες των μορίων και τις συντεταγμένες τους στον τρισδιάστατο χώρο. Έτσι, μπορεί κανείς να μοιάζει με τον συνδυασμό των ατόμων με απλά γραφήματα. Στην περίπτωσή μας, έχουμε πληροφορίες για τα διαφορετικά μόρια που αποτελούνται από τις 3D συντεταγμένες τους και τον τύπο του ατόμου. Φανταστείτε λοιπόν τα άτομα ως κόμβους και τους χημικούς δεσμούς ως ακμές. Αυτός είναι ο απλούστερος τύπος μοντελοποίησης που μπορεί να χρησιμοποιήσει κανείς. Τώρα, με τη χρήση των νευρωνικών δικτύων γραφημάτων έχουμε τη δυνατότητα να προβλέψουμε διάφορα είδη μοριακών ιδιοτήτων. Αυτό μπορεί να επιτευχθεί μέσω της λειτουργικής θεωρίας της Πυκνότητας για να υπολογίσουμε ιδιότητες όπως ενέργειες και ενθαλπίες ως στόχους μας.

Προκειμένου να γενικευθούν οι ικανότητες αυτών των μοντέλων για όλες τις περιπτώσεις, τα μοντέλα GNN κάνουν τέσσερις τύπους προβλέψεων. Ξεκινώντας από τοπικό επίπεδο, ένα GNN μπορεί να προβλέψει την ιδιότητα ενός κόμβου. Στη συνέχεια, λόγω των γειτονιών στα γραφήματα αυτού του είδους τα μοντέλα μπορούν να προβλέψουν τη σύνδεση μεταξύ δύο κόμβων και συνδυάζοντας αυτό με την προηγούμενη πρόβλεψη μπορεί να καταλήξουν σε υποθέσεις για ολόκληρη την ιδιότητα ενός γραφήματος. Μετά από αυτό, μπορούμε να κλιμακώσουμε αυτό το μοτίβο και να εφαρμόσουμε παρόμοιες τεχνικές για να προβλέψουμε την ομοιό-

τητα μεταξύ δύο κόμβων ή δύο γραφημάτων.

## 1. Προβλήματα κατά τη χρήση των GNNs

Ως γνωστόν, τα νευρωνικά δίκτυα γραφημάτων περιλαμβάνονται στο κεφάλαιο βαθιάς μάθησης. Η πλειονότητα των νευρωνικών δικτύων έχει σχεδιαστεί για να αντιμετωπίζει εισόδους σταθερού μεγέθους ή τακτικά διατεταγμένες. Δεν μπορούμε να εφαρμόσουμε την ίδια αρχή σε μοντέλα GNN σε αυτήν την περίπτωση. Η πολυπλοκότητα της τοπολογικής δομής των δικτύων ή των γραφημάτων είναι ένα από τα πιο διαδεδομένα ζητήματα που ανακύπτουν. Δεν μπορούμε να υποθέσουμε, κατά γενικό κανόνα, ότι η είσοδος θα έχει τη μορφή εικόνας με τετράγωνο πλέγμα εικονοστοιχείων και να ακολουθούμε συγκεκριμένους κανόνες για ακριβείς διαστάσεις και πυκνότητα. Σε ορισμένες περιπτώσεις, οι εισόδοι γραφημάτων στερούνται χωρικής εντοπιότητας και ομοιομορφίας. Με αυτό ως γεγονός, μπορούμε να δούμε γιατί οι συνδέσεις μεταξύ των κόμβων είναι αποτέλεσμα και όχι γεγονός. Επιπλέον, υπάρχει έλλειψη σταθερής διάταξης κόμβων ή οποιουδήποτε σημείου αναφοράς, προσθέτοντας δυσκολία στη μελέτη μας καθώς δεν μπορούμε να υποθέσουμε τον ισομορφισμό του προβλήματος. Και αν τα χωρικά ζητήματα δεν είναι αρκετά, μια άλλη σημαντική πτυχή που θα πρέπει να εξετάσει κανείς είναι ότι περιλαμβάνει διαφορετικές οντότητες ως κόμβους και διαφορετικούς τύπους αλληλεπιδράσεων μεταξύ των κόμβων, των ακμών. Τέλος, αυτό που το κάνει τόσο ιδιαίτερη περίπτωση είναι ότι έχει πλούσια και ετερογενή χαρακτηριστικά σχετικά με οντότητες και αλληλεπιδράσεις.

## 2. Τύποι Graph Neural Networks

Εάν επιλέξουμε αυτόν τον τύπο δικτύου, πρέπει επίσης να επιλέξουμε ποιον τύπο GNN θα χρησιμοποιήσουμε σε δεύτερο επίπεδο. Πρέπει πρώτα να επιλέξουμε μεταξύ ενός Νευρωνικού Δικτύου Επαναλαμβανόμενου Γραφήματος, ενός Χωρικού Συνελκτικού Δικτύου και ενός Φασματικού Συνελκτικού Δικτύου. Η λογική του Recurrent GNN είναι να ορίσει μια παραμετροποιημένη συνάρτηση  $fw$  με την οποία η τελική κατάσταση κόμβου θα χρησιμοποιηθεί για την παραγωγή μιας εξόδου μετά από  $k$  επαναλήψεις για να γίνει μια κρίση σχετικά με κάθε κόμβο. Το θεώρημα σταθερού σημείου Banach χρησιμοποιείται για την κατασκευή αυτού του μοντέλου. Η διαφορά μεταξύ αυτού και του Spatial Convolution Network είναι ότι το τελευταίο μοιάζει περισσότερο με το CNN. Αυτό συγκεντρώνει τα χαρακτηριστικά των γειτονικών κόμβων στον κεντρικό κόμβο. Οι δύο τελευταίες κατηγορίες,

γενικά, διευρύνουν τη διαδικασία συνέλιξης από δεδομένα πλέγματος σε δεδομένα γραφήματος. Η αναπαράσταση ενός κόμβου  $v$  δημιουργείται συνδυάζοντας τα δικά του χαρακτηριστικά  $x_v$  και αυτά των γειτόνων του  $x_u$ . Τα συνελκτικά δίκτυα γραφημάτων, σε αντίθεση με την πρώτη κατηγορία, στοιβάζουν πολλά συνελκτικά επίπεδα γραφημάτων για να εξαγάγουν αναπαραστάσεις κόμβων υψηλού επιπέδου και να βοηθήσουν στην ανάπτυξη πιο περίπλοκων μοντέλων GNN. Χρησιμοποιούν επίσης έναν καθορισμένο αριθμό στρωμάτων, το καθένα με ποικίλο βάρος. Με την προσθήκη φίλτρων από τη σκοπιά των συνελίξεων γραφημάτων εισάγοντας φίλτρα από την προοπτική της επεξεργασίας σήματος γραφήματος, όπου η συνελκτική λειτουργία γραφήματος θεωρείται ότι αφαιρεί τους θορύβους από τα σήματα γραφημάτων, οι τεχνικές που βασίζονται σε φασματικούς σχηματισμούς κατασκευάζουν συνελίξεις γραφημάτων. Τα RecGNN ενέπνευσαν χωρικούς τρόπους για τον ορισμό των συνελίξεων γραφημάτων μέσω της διάδοσης πληροφοριών. Αυτά ξεπερνούν τα υπόλοιπα σε αποτελεσματικότητα, ευελιξία και γενικότητα.

Αυτός ο τύπος GNN στοχεύει στην εκμάθηση αναπαραστάσεων κόμβων με επαναλαμβανόμενες νευρωνικές αρχιτεκτονικές, ξεκινώντας με μια πιο εκτεταμένη διερεύνηση των νευρωνικών δικτύων επαναλαμβανόμενων γραφημάτων. Η υποκείμενη υπόθεση είναι ότι κάθε κόμβος σε ένα γράφημα ανταλλάσσει πληροφορίες με τους γείτονές του συνεχώς μέχρι να βρεθεί μια σταθερή ισορροπία. Ως αποτέλεσμα, λειτουργεί σε μια από τις πιο θεμελιώδεις αρχές όλων των φυσικών συστημάτων: την αναζήτηση για το χαμηλότερο ενεργειακό σημείο. Τα RecGNN εξάγουν αναπαραστάσεις κόμβων υψηλού επιπέδου εφαρμόζοντας επανειλημμένα το ίδιο σύνολο παραμέτρων σε κόμβους σε ένα γράφημα.

Εάν επιλέξουμε αυτόν τον τύπο δικτύου, πρέπει επίσης να επιλέξουμε ποιον τύπο GNN θα χρησιμοποιήσουμε σε δεύτερο επίπεδο. Πρέπει πρώτα να επιλέξουμε μεταξύ ενός Νευρωνικού Δικτύου Επαναλαμβανόμενου Γραφήματος, ενός Χωρικού Συνελκτικού Δικτύου και ενός Φασματικού Συνελκτικού Δικτύου. Η λογική του Recurrent GNN είναι να ορίσει μια παραμετροποιημένη συνάρτηση  $fw$  με την οποία η τελική κατάσταση κόμβου θα χρησιμοποιηθεί για την παραγωγή μιας εξόδου μετά από  $k$  επαναλήψεις για να γίνει μια κρίση σχετικά με κάθε κόμβο. Το θεώρημα σταθερού σημείου Banach χρησιμοποιείται για την κατασκευή αυτού του μοντέλου. Η διαφορά μεταξύ αυτού και του Spatial Convolution Network είναι ότι το τελευταίο μοιάζει περισσότερο με το CNN. Αυτό συγκεντρώνει τα χαρακτηριστικά των γειτονικών κόμβων στον κεντρικό κόμβο. Οι δύο τελευταίες κατηγορίες, γενικά, διευρύνουν τη διαδικασία συνέλιξης από δεδομένα πλέγματος σε δεδομένα γραφήματος. Η αναπαράσταση ενός κόμβου  $v$  δημιουργείται συνδυάζοντας τα δικά του χαρακτηριστικά  $x_v$  και αυτά των γειτόνων του  $x_u$ . Τα συνελκτικά δίκτυα γραφημάτων, σε αντίθεση με την πρώτη κατηγορία, στοιβάζουν πολλά συνελκτικά επίπεδα γραφημάτων για να εξαγάγουν αναπαραστάσεις κόμβων υψηλού επιπέδου και να βοηθήσουν στην ανάπτυξη πιο περίπλοκων μοντέλων GNN. Χρησιμοποιούν

επίσης έναν καθορισμένο αριθμό στρωμάτων, το καθένα με ποικίλο βάρος. Με την προσθήκη φίλτρων από τη σκοπιά των συνελίξεων γραφημάτων εισάγοντας φίλτρα από την προοπτική της επεξεργασίας σήματος γραφήματος, όπου η συνελικτική λειτουργία γραφήματος θεωρείται ότι αφαιρεί τους θορύβους από τα σήματα γραφημάτων, οι τεχνικές που βασίζονται σε φασματικούς σχηματισμούς κατασκευάζουν συνελίξεις γραφημάτων. Τα RecGNN ενέπνευσαν χωρικούς τρόπους για τον ορισμό των συνελίξεων γραφημάτων μέσω της διάδοσης πληροφοριών. Αυτά ξεπερνούν τα υπόλοιπα σε αποτελεσματικότητα, ευελιξία και γενικότητα.

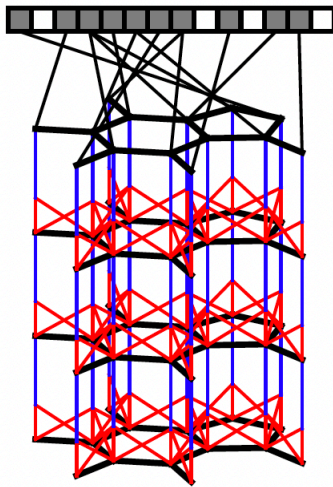
Αυτός ο τύπος GNN στοχεύει στην εκμάθηση αναπαραστάσεων κόμβων με επαναλαμβανόμενες νευρωνικές αρχιτεκτονικές, ξεκινώντας με μια πιο εκτεταμένη διερεύνηση των νευρωνικών δικτύων επαναλαμβανόμενων γραφημάτων. Η υποκείμενη υπόθεση είναι ότι κάθε κόμβος σε ένα γράφημα ανταλλάσσει πληροφορίες με τους γείτονές του συνεχώς μέχρι να βρεθεί μια σταθερή ισορροπία. Ως αποτέλεσμα, λειτουργεί σε μια από τις πιο θεμελιώδεις αρχές όλων των φυσικών συστημάτων: την αναζήτηση για το χαμηλότερο ενεργειακό σημείο. Τα RecGNN εξάγουν αναπαραστάσεις κόμβων υψηλού επιπέδου εφαρμόζοντας επανειλημμένα το ίδιο σύνολο παραμέτρων σε κόμβους σε ένα γράφημα.

### 3. Αλγόριθμος GNN με εστίαση στην Εφαρμογή Συνελικτικών Δικτύων σε Γραφήματα

Σε αυτήν την ενότητα αυτής της διατριβής, θα αναλύσουμε σύντομα έναν τύπο συνελικτικού νευρωνικού δικτύου που λειτουργεί απευθείας χρησιμοποιώντας ως δομές γραφημάτων εισόδου και στις περισσότερες περιοχές είναι καλύτερο να εφαρμοστεί στα Μοριακά Fingerprints. Η φύση του προβλήματος όπως φαίνεται παρακάτω: Για να εξεταστούν τα χημικά δακτυλικά αποτυπώματα με ένα συνελικτικό δίκτυο, πρώτα πρέπει να δημιουργηθεί ένα γράφημα που ταιριάζει με την αρχιτεκτονική του μορίου, με κόμβους που αντιπροσωπεύουν άτομα και άκρες που αντιπροσωπεύουν χημικούς δεσμούς. Ευτυχώς, τα SMILES στο σύνολο δεδομένων QM-9 ακολουθούν παρόμοια λογική. Οι πληροφορίες ταξιδεύουν μεταξύ γειτόνων στο γράφημα σε κάθε επίπεδο του CNN. Τέλος, κάθε κόμβος δικτύου ενεργοποιεί ένα bit στο διάνυμα δακτυλικών αποτυπωμάτων σταθερού μήκους.

Όπως συζητήθηκε προηγουμένως, μία από τις προκλήσεις με τη χρήση οποιουδήποτε τύπου νευρωνικού δικτύου είναι ότι σχεδόν σίγουρα θα απαιτούν εισόδους σταθερού μεγέθους, καθώς οι περισσότεροι αγωγοί μηχανικής εκμάθησης δεν μπορούν να διαχειριστούν εισόδους αυθαίρετου μεγέθους, κάτι που θα ήταν σημαντικό ζήτημα για την έρευνα μορίων. Αυτό το ζήτημα προκύπτει επειδή τα μόρια





Σχήμα 5.1: GCN from literature

μπορεί να έχουν οποιοδήποτε μέγεθος ή μορφή. Χρησιμοποιώντας ένα διαφορικό νευρωνικό δίκτυο με ένα γράφημα ως είσοδο, τα μοριακά διανύσματα δακτυλικών αποτυπωμάτων μπορούν να υπολογιστούν πιο γρήγορα. Από τη βιβλιογραφία, ο D. Duvenaud και οι συνεργάτες του ανέπτυξαν ένα μοντέλο που χρησιμοποιεί πολλά στρώματα, καθένα από τα οποία εφαρμόζει το ίδιο τοπικό φίλτρο σε κάθε άτομο και το περιβάλλον του, με τα τελικά στρώματα να συνδυάζουν χαρακτηριστικά από όλα τα άτομα του μορίου χρησιμοποιώντας ένα παγκόσμιο βήμα συγκέντρωσης. Κάθε χαρακτηριστικό ενός κυκλικού διανύσματος δακτυλικών αποτυπωμάτων μπορεί να ενεργοποιηθεί μόνο από ένα μεμονωμένο θραύσμα μιας μόνο ακτίνας και σε αντίθεση με αυτό, τα χαρακτηριστικά δακτυλικών αποτυπωμάτων νευρικού γραφήματος μπορούν να ενεργοποιηθούν από παραλλαγές της ίδιας δομής, καθιστώντας τα πιο ερμηνεύσιμα και επιτρέποντας μικρότερα διανύσματα χαρακτηριστικών .

Αυτή η προσέγγιση έχει ανώτερη προγνωστική ικανότητα σε σύγκριση με τα παραδοσιακά δακτυλικά αποτυπώματα για διαλυτότητα, φαρμακολογική αποτελεσματικότητα και δεδομένα απόδοσης οργανικών φωτοβολταϊκών. Η προσέγγιση μπορεί να βελτιστοποιηθεί ώστε να κωδικοποιεί απλώς σχετικά χαρακτηριστικά χρησιμοποιώντας διαφοροποιήσιμα δακτυλικά αποτυπώματα, μειώνοντας τις ανάγκες μεταγενέστερου υπολογισμού και τακτοποίησης και, όπως αναφέρθηκε προηγουμένως, η ενεργοποίηση χαρακτηριστικών γίνεται πολύ πιο απλή και ουσιαστική. Ωστόσο, ο υπολογισμός του νευρικού αποτυπώματος βάθους  $R$ , μήκους  $L$  ενός μορίου με άτομα  $N$  χρησιμοποιώντας ένα μοριακό συνελκτικό δίχτυ με χαρακτηριστικά  $F$  σε κάθε στρώμα κοστίζει  $O(RNFL + RNF^2)$ , η οποία είναι μια υπολογιστικά ακριβή μέθοδος. Επιπλέον, οι υλοποιήσεις τους εξακολουθούν να μην είναι ευαίσθητες στην ταξινόμηση στερεοϊσομερών.

Τα γραφήματα NN είναι καλύτερα κατάλληλα για εφαρμογές όπως ταξινόμηση κόμβων, ταξινόμηση γραφημάτων, ομαδοποίηση κόμβων, πρόβλεψη συνδέσμων και μεγιστοποίηση επιρροής. Ο κύριος σκοπός αυτής της διατριβής είναι να το χρησιμοποιήσει ως μοντέλο ταξινόμησης γραφημάτων για την κατηγοριοποίηση διαφορετικών μορίων, ή ακόμα και ως μοντέλο πρόβλεψης συνδέσμων για να διερευνήσει πώς διαφορετικοί τύποι ατόμων ενώνονται για να παράγουν νέα μόρια. Συζητήσαμε για τα Συνελικτικά Δίκτυα και τώρα θα θέλαμε να εφαρμόσουμε αυτές τις συνελικτώσεις σε γραφήματα. Ο στόχος αυτής της άσκησης είναι να επεκτείνει τις συνελίξεις σε οποιοδήποτε γράφημα.

Η βασική ιδέα πίσω από τις συνελίξεις γραφημάτων είναι ότι περιλαμβάνει απλούς υπολογισμούς, καθώς κάθε κόμβος συγκεντρώνει πληροφορίες από τους γύρω κόμβους για να ενημερώσει τις ενσωματώσεις του. Το διάνυμα χαρακτηριστικών από κάθε γειτονικό κόμβο του προηγούμενου επιπέδου συγκεντρώνεται με τον ίδιο τον κόμβο πάνω δεξιά. Στη συνέχεια, μπορούμε να εφαρμόσουμε έναν μη γραμμικό μετασχηματισμό στο διάνυμα συγκεντρωτικών κόμβων για να φτάσουμε στην ενημερωμένη ενσωμάτωση. Επιλέγουμε αυτό να είναι ένας εκπαιδευσιμος πίνακας βάρους ακολουθούμενος από μια συνάρτηση ενεργοποίησης (π.χ. ReLU). Εδώ είναι σημαντικό να σημειωθεί ότι η εφαρμογή της συνάρτησης ενημέρωσης και της μήτρας βάρους είναι ομοιόμορφη σε όλους τους κόμβους αυτού του επιπέδου, πράγμα που σημαίνει ότι ο μηχανισμός ενσωμάτωσης είναι ο ίδιος, κάτι που είναι λογικό αφού ακόμη και με CNN τα βάρη του φίλτρου εικόνας μοιράζονται μεταξύ τους, στρώμα. Για να το επεκτείνουμε πέρα από ένα νευρωνικό δίκτυο μονού επιπέδου, μπορούμε ένα βαθύτερο NN για να κατασκευάσουμε τον παραπάνω μετασχηματισμό. Τέλος, μετά τα παραπάνω βήματα, ο αντίστοιχος κόμβος του επόμενου επιπέδου ενημερώνεται στην προκύπτουσα ενσωμάτωση χαρακτηριστικών. Η ροή μπορεί να περιγραφεί ως αρχή από το γράφημα εισόδου, ακολουθούμενη από τα μπλοκ GNN. Μετά από αυτά υλοποιείται το μετασχηματισμένο γράφημα, στο οποίο προστίθεται ένα επίπεδο ταξινόμησης και τελικά φτάνουμε στο σημείο πρόβλεψης.

## 4. Generative Adversarial Networks

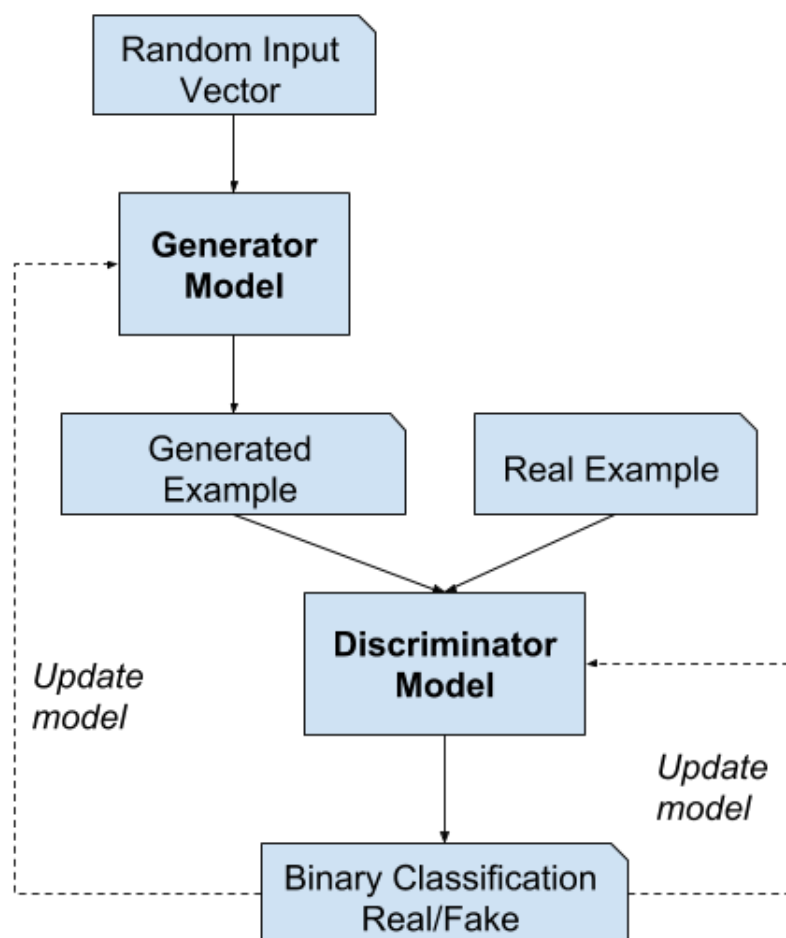
Στην κλασική μηχανική εκμάθηση, τα GAN χρησιμοποιούνται κυρίως για τη δημιουργία δεδομένων όπως εικόνες υψηλής ανάλυσης. Ωστόσο, έχει Παρατηρήθηκε ότι αυτού του είδους τα μοντέλα είναι υπολογιστικά ακριβά. Έτσι, τα κβαντικά ανάλογα είναι εδώ για να τα ξεπεράσουν. Ο σκοπός της χρήσης QGAN είναι επίσης ένας αγώνας για να ευδοκιμήσει κανείς στη διαδικασία βελτιστοποίησης ενώ

εφαρμόζει κβαντικά κυκλώματα για να παράγει μια καλή εκτίμηση σε οποιοδήποτε πρόβλημα υψηλής πολυπλοκότητας.

Αρχικά, στον κλασικό κόσμο τα γενετικά μοντέλα ανήκουν σε μη εποπτευόμενη μηχανική μάθηση και στην ακριβή περίπτωση των Generative Adversarial Networks χρησιμοποιούνται ως μέθοδος εκπαίδευσης για αυτά τα μοντέλα πλαισιώνοντας το πρόβλημα ως ένα εποπτευόμενο πρόβλημα μάθησης με δύο τύπους μοντέλων:

- ▶ The Generator Model, το οποίο είναι εκπαιδευμένο να δημιουργεί νέα δεδομένα και
- ▶ The Discriminator Model με στόχο να ταξινομήσει αυτά τα δεδομένα και να κατανοήσει ποια είναι πλαστά και αληθινά.

Αυτά τα δύο μοντέλα είναι ανταγωνιστικά μεταξύ τους και στο σύνολό τους, τα GAN, παρέχουν μια μέθοδο για την αύξηση δεδομένων για συγκεκριμένο τομέα και λύσεις σε προβλήματα που απαιτούν μια γενετική λύση. Το μοντέλο που περιγράφεται από τον Ian Goodfellow και ήταν η πρώτη προτεινόμενη μέθοδος που απελευθέρωσε τον χρήστη από τις αλυσίδες Markov και τα κατά προσέγγιση δίκτυα συμπερασμάτων κατά τη διάρκεια της εκπαίδευσης ή της δημιουργίας δειγμάτων.

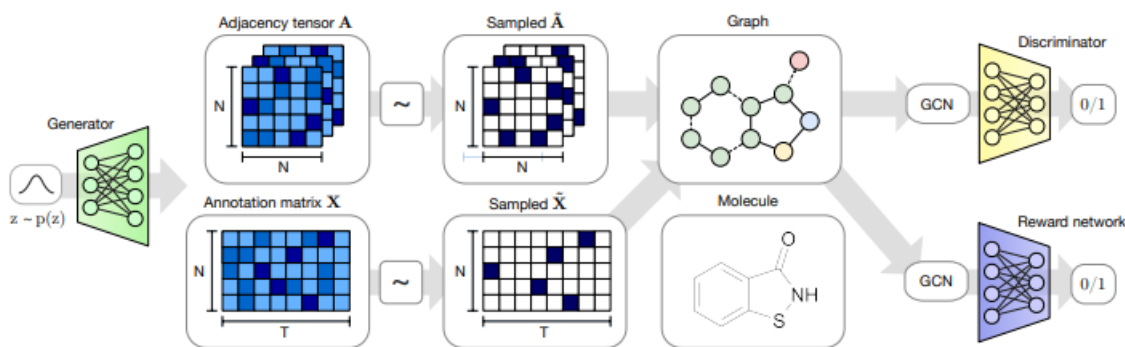


Σχήμα 5.2: GANs model

Παρέχω πιο συγκεκριμένες πληροφορίες για την περίπτωση χρήσης της ανακάλυψης φαρμάκων εξηγώντας το μοντέλο MolGAN στην επόμενη υποενότητα.

## 5. MolGAN model

Για το αντικείμενο της παρούσας διπλωματικής εργασίας, ωστόσο, εστιάζουμε σε έναν συγκεκριμένο τύπο GAN, το MolGAN. Έτσι, το μοντέλο αποτελείται από τρία κύρια στοιχεία: ένα generator  $G_\theta$ , μια συσκευή discriminator  $D_\phi$  και ένα δίκτυο ανταμοιβής  $\hat{R}_\psi$ .



Σχήμα 5.3: MolGAN flow

Όπως φαίνεται στο 5.3, ο generator παίρνει ένα δείγμα από μια προηγούμενη κατανομή και δημιουργεί έναν πυκνό τανυστή γειτνίασης  $A$  και έναν πίνακα σχολιασμού  $X$ . Στη συνέχεια, διαχωρίστε και διακριτά  $A$  και *overset*  $X$  λαμβάνονται από το  $A$  και το  $X$  αντίστοιχα μέσω κατηγορικής δειγματοληψίας. Ο συνδυασμός  $A$  και  $X$  αντιπροσωπεύει ένα σχολιασμένο μοριακό γράφημα που αντιστοιχεί σε μια συγκεκριμένη χημική ένωση. Τέλος, το γράφημα επεξεργάζεται τόσο από τα δίκτυα discriminator όσο και από τα δίκτυα ανταμοιβής που είναι αμετάβλητα στις μεταθέσεις σειράς κόμβων και βασίζονται σε επίπεδα Relational-GCN. Για την εκπαίδευση του discriminator χρησιμοποιείται η συνάρτηση απώλειας του μοντέλου WGAN, ενώ ο generator χρησιμοποιεί έναν γραμμικό συνδυασμό απώλειας WGAN και απώλειας RL:

$$L(\theta) = \lambda L_{WGAN} + (1 - \lambda) L_{RL} \quad (5.1)$$

όπου  $\lambda \in [0, 1]$  είναι η υπερπαράμετρος που ρυθμίζει την ανταλλαγή μεταξύ των δύο στοιχείων και το  $L_{WGAN}$  δίνεται από:

$$L(x^{(i)}, G_\theta(z^{(i)}; \phi) = -D_\phi(x^{(i)}) + D_\phi(G_\theta(z^{(i)})) + a(\|\nabla_{\hat{x}^{(i)}} D_\phi(\hat{x}^{(i)})\| - 1)^2 \quad (5.2)$$

όπου το πρώτο μέρος είναι η αρχική απώλεια WGAN και το δεύτερο μέρος είναι μια ποινή διαβάθμισης, με ένα: μια υπερπαράμετρο,  $\hat{x}^{(i)}$  έναν δειγματοληπτικό γραμμικό συνδυασμό  $x^{(i)}$  και  $G_\theta(z^{(i)})$ . Το  $G_\theta$  είναι το παραγωγικό μοντέλο, το  $D_\phi$  ένα διακριτικό μοντέλο που μαθαίνει να ταξινομεί εάν τα δείγματα προήλθαν από τη διανομή δεδομένων και όχι από το  $G_\theta$ .

# Κεφάλαιο 6

## Quantum Neural Networks

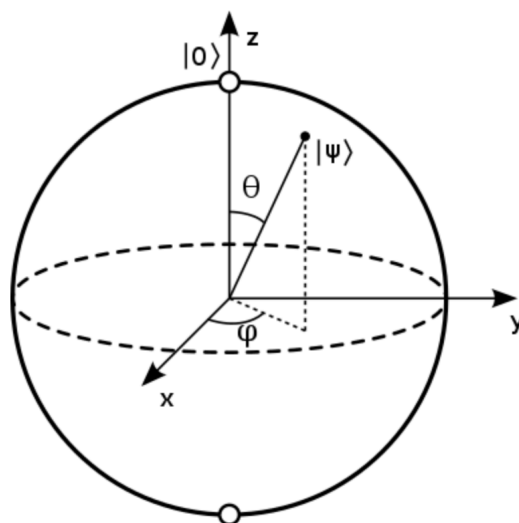
### 1. Τα θεμέλια για την Κβαντική Εποχή

Ενώ μπαίνουμε στην εποχή των Κβαντικών υπολογιστών, μπαίνουμε στην εποχή των πιθανοτήτων. Γυρίζουμε από την κλασική μονάδα πληροφοριών, το bit, στο κβαντικό ανάλογο, το qubit. Η βασική διαφορά που συναντάμε στον κβαντικό υπολογισμό είναι η ύπαρξη καταστάσεων υπέρθεσης. Τι εννοούμε με τον όρο υπέρθεση; Για να απαντήσουμε ότι πρέπει πρώτα να καταλάβουμε τι είναι το qubit. Στον κλασικό υπολογισμό, το bit μπορεί να λάβει δύο τιμές, 0 και 1. Σε ένα qubit, δεν χρησιμοποιούμε το 0 και το 1 ως αριθμούς, αλλά χρησιμοποιούμε  $|0\rangle$  και  $|1\rangle$  ως δύο κβαντικές καταστάσεις ενός συστήματος, που ονομάζεται κβαντικό bit ή qubit. Οι κβαντικοί υπολογιστές είναι κατασκευασμένοι για να εκμεταλλεύονται τη φύση των qubits υπακούοντας στους νόμους της κβαντικής μηχανικής. Σύμφωνα λοιπόν με την κβαντική θεωρία, ένα σωματίδιο εισέρχεται σε μια υπέρθεση καταστάσεων, στην οποία συμπεριφέρεται σαν να βρίσκεται και στις δύο καταστάσεις ταυτόχρονα. Έτσι, τώρα ο αριθμός των υπολογισμών που θα μπορούσε να κάνει ένας κβαντικός υπολογιστής είναι  $2^n$ , όπου  $n$  είναι ο αριθμός των qubits που χρησιμοποιούνται σε αυτόν τον υπολογιστή. Η κατάσταση ενός qubit μπορεί να γραφτεί ως εξής:

$$|q\rangle = \alpha|0\rangle + \beta|1\rangle$$

όπου  $|0\rangle$  είναι μια κατάσταση που περιγράφεται από το διάνυσμα  $(1, 0)$ ,  $|1\rangle$  ένα διάνυσμα κατάστασης όπως  $(0, 1)$  και οι σταθερές  $\alpha, \beta$  είναι δύο σταθερές ορίζοντας την % πιθανότητα ότι το qubit θα βρίσκεται σε κάθε κατάσταση και από τη θεωρία πιθανοτήτων  $|\alpha|^2 + |\beta|^2 = 1$ . Αυτά απεικονίζονται σε μια ενιαία σφαίρα, που ονομάζεται σφαίρα Bloch. Αυτή η σφαίρα είναι μια γεωμετρική αναπαράσταση του χώρου καθαρής κατάστασης ενός κβαντομηχανικού συστήματος δύο επιπέδων, ενός

qubit.



Σχήμα 6.1: Bloch sphere

Οι δύο καταστάσεις qubit  $|0\rangle$ ,  $|1\rangle$  αντιπροσωπεύονται από τους άξονες  $z$ ,  $-z$ . Οποιοδήποτε σημείο  $|\psi\rangle$  σε αυτή τη σφαίρα αντιπροσωπεύεται από την εξίσωση  $|q\rangle$  που έγραψα παραπάνω. Η σφαίρα Bloch είναι μια γενίκευση ενός μιγαδικού αριθμού  $z$  με  $|z|^2 = 1$  ως σημείο στον μοναδιαίο κύκλο στο μιγαδικό επίπεδο. Έτσι, καταλαβαίνουμε ότι η πιο γενική μορφή που μπορούμε να γράψουμε την εξίσωση ενός qubit είναι η παρακάτω:

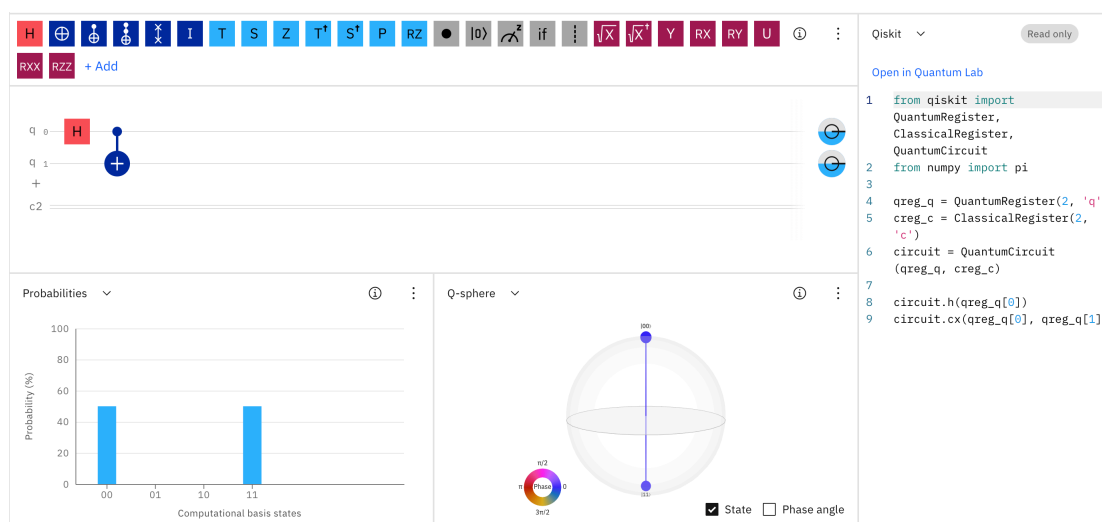
$$|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\phi}\sin\left(\frac{\theta}{2}\right)|1\rangle$$

όπου  $\theta, \phi \in [0, \pi]$ .

Για να χειριστούμε ένα qubit ή πολλαπλά qubits, χρησιμοποιούμε βασικές πύλες, όπως στον κλασικό υπολογισμό, που περιλαμβάνει το X-Gate, το Y-Gate, το CNOT Gate, το NAND Gate, το XOR Gate και το OR Gate, το Hadamard gate και μερικές πιο περίπλοκες λειτουργίες. Η Pauli X-Gate (NOT gate) αντιστρέφει την τιμή ενός qubit από 0 σε 1 ή 1 σε 0. Αυτή η πύλη αντιστροφής αναφέρεται συχνά ως bit-flip. Το Pauli Y-Gate ισοδυναμεί με τιμή bit και αναστροφή φάσης σε μία λειτουργία. Έτσι, μετατρέπει την τιμή +1 σε -1 ή αντίστροφα. Η πύλη CNOT λειτουργεί ως η κλασική πύλη AND και όπως μπορεί κανείς να φανταστεί αυτή η πύλη δεν λειτουργεί σε ένα μόνο qubit αλλά σε ένα σύστημα 2 ή περισσότερων. Ονομάζεται CNOT, επειδή είναι μια πύλη Ελεγχόμενης ΜΗ. Εάν κάποιο από τα qubit εισόδου είναι 0, το qubit που προκύπτει θα είναι 0. Υπάρχει επίσης η πύλη Toffoli (ή αλλιώς η πύλη CCNOT) για τα κυκλώματα των 3 qubit. Απλώς χρειάζονται δύο

qubit εισόδου και αναστρέφει την τιμή ενός qubit που προκύπτει εάν οι δύο εισοδοί έχουν τιμή 1. Εάν θέλουμε να φτάσουμε στην υπέρθεση, τότε πρέπει να χρησιμοποιήσουμε την πύλη Hadamard. Η πύλη Hadamard μετατρέπει μια κατάσταση 0 σε  $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$  και μια κατάσταση 1 σε  $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$ . Και στις δύο περιπτώσεις, η προκύπτουσα κατάσταση εμφανίζει ίση πιθανότητα για κάθε ενδεχόμενο. Η πύλη NAND λειτουργεί ως πύλη NOT AND και η αποκλειστική πύλη OR (XOR) μπορεί να δημιουργηθεί με μια πύλη CX. Αυτό παίρνει μια είσοδο και ένα qubit εξόδου. Το qubit εξόδου θα αντιστραφεί μόνο εάν το qubit εισόδου έχει τιμή 1.

Όλα τα παραπάνω μπορούν να χρησιμοποιηθούν για τη σύνθεση ενός κβαντικού κυκλώματος και ένας απλός τρόπος είναι να χρησιμοποιήσετε το IBM Quantum Composer για την οπτικοποίηση και τη δημιουργία αυτών των κυκλωμάτων. Είναι μια εφαρμογή web drag and drop που σας βοηθά να ενσωματώσετε στοιχεία ελέγχου κβαντικού κυκλώματος και να τα εκτελέσετε σε έναν προσομοιωτή ή σε έναν πραγματικό κβαντικό υπολογιστή στην IBM.



Σχήμα 6.2: IBM Quantum Composer

Όπως φαίνεται στο 6.2, πρώτα έχουμε πολλές πύλες και λειτουργίες στη διάθεσή μας και πολλές από αυτές περιλαμβάνονται στις πύλες που συζητήθηκαν προηγουμένως. Κάτω από το σύνολο των εργαλείων, μπορείτε να βρείτε την παλέτα μεταφοράς και απόθεσης, όπου μπορείτε να προσθέσετε τον αριθμό των qubits που θέλετε για το κύκλωμά σας, όσο είναι ο κλασικός καταχωρητής και μετά να εκτελέσετε λειτουργίες στα qubits. Από προεπιλογή, όλα τα qubit αρχικοποιούνται στην κατάσταση  $|0\rangle$ . Εδώ, χρησιμοποίησα ένα παράδειγμα για την κατασκευή μιας από τις τέσσερις καταστάσεις Bell, χρησιμοποιώντας ένα Hadamard και μια πύλη CX. Στο



κάτω μέρος της σελίδας, μπορούμε να βρούμε τις πιθανότητες πρόβλεψης, οι οποίες σε αυτήν την περίπτωση είναι 50% για να καταλήξουν στην κατάσταση  $|0\rangle$  και 50% για να καταλήξουν σε  $|11\rangle$ . Δίπλα σε αυτό το στοιχείο μπορούμε να βρούμε τη σφαίρα Q. Η σφαίρα Q παρέχει μια συνολική άποψη της κβαντικής κατάστασης πολλαπλών qubit στην υπολογιστική βάση. Το μέγεθος του κόμβου είναι ανάλογο για να εκφράζει την πιθανότητα της κατάστασης και το χρώμα αντικατοπτρίζει τη φάση κάθε βασικής κατάστασης. Τέλος, στη δεξιά πλευρά της οθόνης, βρίσκουμε την ενότητα Quantum Lab, όπου το IBM Quantum Composer μετατρέπει το υλοποιημένο κβαντικό μας κύκλωμα σε κώδικα rython χρησιμοποιώντας τη βιβλιοθήκη Qiskit. Αυτό είναι πολύ χρήσιμο καθώς σε πολλά έργα και ειδικά στην Κβαντική Μηχανική Εκμάθηση, χρησιμοποιούμε υβριδικούς κλασσικούς-κβαντικούς αλγόριθμους και είναι πολύ επιθυμητό να εναλλάσσουμε εργασίες χρησιμοποιώντας το ίδιο εργαλείο.

Τώρα που εξηγήσαμε μερικές βασικές πύλες και τις βασικές λειτουργίες του εργαλείου, είναι καλή στιγμή να επισημάνουμε τον τρόπο ανάγνωσης του ραβδόγραμματος πιθανοτήτων του IBM Quantum Composer για να κατανοήσουμε την έξοδο από ένα κβαντικό κύκλωμα. Σε αυτό το διάγραμμα μπορείτε να δείτε ποιο αποτέλεσμα θα είχε το qubit μετά την εφαρμογή της σειράς κβαντικών πυλών. Όπως υπαγορεύεται από τους νόμους της κβαντικής μηχανικής, για να γνωρίζουμε αυτό το αποτέλεσμα πρέπει να μετρήσουμε το κύκλωμά μας. Όταν όμως συνεχίσουμε με τη μέτρηση, η κυματική συνάρτηση του συστήματος καταρρέει. Τι εννοούμε με αυτό; Όλα τα αξιώματα της κβαντικής μηχανικής είναι εφαρμόσιμα σε κλειστά, απομονωμένα συστήματα. Τα δύο βασικά αξιώματα είναι τα ακόλουθα:

- ▶ Για ένα απομονωμένο κβαντικό σύστημα, μπορούμε να ορίσουμε έναν χώρο Hilbert, γνωστό και ως σύνθετο διανυσματικό χώρο με ένα εσωτερικό γινόμενο. Αυτό επιτρέπει πολλές λειτουργίες όπως μέτρηση μήκους και γωνιών.
- ▶ Η εξέλιξη ενός κλειστού συστήματος μπορεί να περιγραφεί με έναν ενιαίο μετασχηματισμό  $U$ , έτσι ώστε  $|\psi'\rangle = U|\psi\rangle$ .

Το κύριο συμπέρασμα των παραπάνω είναι ότι τα κλειστά συστήματα που περιγράφονται από την ενιαία χρονική εξέλιξη (φορμαλισμός του Χαμιλτονίου) μπορούν να μετρηθούν με προβολικές μετρήσεις.

Όταν ένα σύστημα περιέχει περισσότερα από ένα qubit και ειδικά όταν οι καταστάσεις δεν είναι καλά καθορισμένες καταστάσεις 0 και 1, τα πράγματα γίνονται πιο περίπλοκα. Πρώτα απ' όλα, η κυματική συνάρτηση του συστήματος γράφεται με τη μορφή:

$$|\psi\rangle = a_{00}|00\rangle + a_{01}|01\rangle + a_{10}|10\rangle + a_{11}|11\rangle$$

με  $\sum_{ij} |a_{ij}|^2 = 1$ .

Το πρόβλημα με αυτήν την κατάσταση είναι ότι παρόλο που έχουμε τους τέσσερις μιγαδικούς αριθμούς, οι περισσότερες πληροφορίες του συστήματος δεν είναι προσβάσιμες με μέτρηση. Εάν μετρήσουμε ολόκληρο το σύστημα ή εκτελέσουμε μερική μέτρηση ορισμένων qubits, λαμβάνουμε μια νέα κατάσταση. Η νέα υπέρθεση προκύπτει διαγράφοντας όλους εκείνους τους όρους του  $|\psi\rangle$  που δεν συνάδουν με το αποτέλεσμα της μέτρησης. Μετά την κανονικοποίηση για την υπακοή στους νόμους πιθανοτήτων και στους νόμους διατήρησης, παίρνουμε:

$$|\psi_{new}\rangle = \frac{a_{00}|00\rangle + a_{01}|01\rangle}{\sqrt{|a_{00}|^2 + |a_{01}|^2}}$$

Το μεγάλο ερώτημα τελικά είναι ότι μπορούμε να αποσυνθέσουμε ολόκληρη την κατάσταση μας σε μερικές μετρήσεις δύο qubits; Κλασικά δεν θα υπήρχε πρόβλημα με αυτό και θεωρήστε ότι κάθε δύο qubits θα πρέπει να είναι σε κατάσταση της μορφής  $\alpha|0\rangle + \beta|1\rangle$ . Ωστόσο, έχουμε δει στην πράξη ορισμένες πολιτείες, όπως οι πολιτείες Bell, οι οποίες είναι:

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \quad (6.1)$$

$$|\Phi^-\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle) \quad (6.2)$$

$$|\Psi^+\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle) \quad (6.3)$$

$$|\Psi^-\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle) \quad (6.4)$$

που δεν μπορούν να διαχωριστούν σε δύο ξεχωριστές καταστάσεις κάθε qubit. Αυτό το φαινόμενο ονομάζεται entanglement. Όταν δύο qubit είναι entangled, δεν μπορούμε να προσδιορίσουμε την κατάσταση κάθε qubit ξεχωριστά, επειδή η κατάσταση κάθε qubit έχει να κάνει τόσο με τη σχέση των δύο qubits όσο και με τις επιμέρους καταστάσεις τους. Οι παραπάνω καταστάσεις Bell θεωρούνται ως οι καταστάσεις στις οποίες δύο qubits είναι entangled στο μέγιστο.

Εάν μετρήσουμε την πρώτη κατάσταση, τότε το αποτέλεσμα είναι 50% 0 και 50% 1. Ωστόσο, εάν το αποτέλεσμα είναι 0, τότε μια μέτρηση του δεύτερου qubit θα έχει ως αποτέλεσμα 0 για 100% πιθανότητα ανεξάρτητα από τη χωρική απόσταση μεταξύ τα δύο qubits. Με τον ίδιο τρόπο όπως σε ένα GNN, λαμβάνουμε υπόψη τόσο την κατάσταση του κόμβου όσο και την αλληλεπίδραση που έχει με τους γείτονές του, εδώ τα qubits που βρίσκονται σε μπλεγμένη κατάσταση, συνδυάζονται. Αυτό είναι και το μυστικό της Κβαντικής Τηλεμεταφοράς.

## 2. Data Encoding στο QML

Οι αλγόριθμοι που χρησιμοποιούνται στην Κβαντική Μηχανική Μάθηση είναι υβριδικοί κλασσικοί-κβαντικοί αλγόριθμοι, όπως συζητήθηκε σε προηγούμενα κεφάλαια. Πρέπει να είμαστε σε θέση να προσαρμόσουμε τα δεδομένα εισόδου μας ώστε να μπορούν να χρησιμοποιηθούν από τα διάφορα είδη εργαλείων και βιβλιοθηκών που θα χρησιμοποιηθούν κατά την υλοποίηση λόγω της φύσης αυτών των μεθόδων. Για τον κβαντικό υπολογισμό, η κλασική κωδικοποίηση δεδομένων είναι ζωτικής σημασίας για τη συνολική σχεδίαση και την αποτελεσματικότητα του αλγορίθμου. Με τον όρο κωδικοποίηση, εννοούμε τη φόρτωση κλασσικών δεδομένων στην κατάσταση των qubits. Υπάρχουν διάφορες επιλογές για να το πετύχουμε αυτό. Γενικά, κατά τη διαδικασία κωδικοποίησης των qubit με δεδομένο feature vector  $\vec{x} = [x_1, \dots, x_N]^T \in R^N$ , τα γενικά qubit encoding maps  $\vec{x} \rightarrow E(\vec{x})$  δίνονται από τη σχέση

$$|x\rangle = \otimes_{i=1}^{\lfloor N/2 \rfloor} f_i(x_{2i-1}, x_{2i})|0\rangle + g_i(x_{2i-1}, x_{2i})|1\rangle \quad (6.5)$$

όπου  $f, g : R \times R \rightarrow C$  είναι τέτοια ώστε να ισχύει  $|f_i|^2 + |g_i|^2 = 1$  για κάθε  $i$ .

Η κωδικοποίηση μπορεί να κατηγοριοποιηθεί σε δύο μεγάλες κατηγορίες:

- ▶ Η ψηφιακή κωδικοποίηση είναι η αναπαράσταση δεδομένων ως συμβολοσειρές qubit. Αυτή η κατηγορία είναι προτιμότερη εάν τα δεδομένα πρέπει να υποβληθούν σε επεξεργασία με αριθμητικούς υπολογισμούς.
- ▶ Η αναλογική κωδικοποίηση αντιπροσωπεύει δεδομένα στα πλάτη μιας κατάστασης. Αυτή η κατηγορία είναι περισσότερο προτιμότερη για αλγόριθμους μηχανικής μάθησης, καθώς απαιτείται η χαρτογράφηση δεδομένων στον μεγάλο χώρο Hilbert της κβαντικής συσκευής.

Αρχικά, μία από τις πιο κοινές μεθόδους είναι η κωδικοποίηση βάσης. Όταν οι πραγματικοί αριθμοί πρέπει να χειρίζονται αριθμητικά σε έναν κβαντικό αλγόριθμο, αυτή η περίπτωση είναι πιο εφαρμόσιμη. Ο ιδανικός στόχος είναι η αναπαράσταση πραγματικών αριθμών στο δυαδικό σύστημα και στη συνέχεια η μετατροπή τους σε κβαντική κατάσταση σε κάποια υπολογιστική βάση. Αυτή η μέθοδος, ωστόσο, είναι απαγορευτικά δαπανηρή λόγω της μεγάλης ποσότητας των qubits που απαιτούνται. Η μέθοδος κωδικοποίησης Amplitude είναι μια άλλη επιλογή για την κωδικοποίηση των δεδομένων μας. Τα δεδομένα κωδικοποιούνται στα πλάτη μιας κβαντικής κατάστασης, όπως υποδηλώνει το όνομα της μεθόδου. Όταν πρέπει να εκμεταλλευτούμε τον τεράστιο χώρο Hilbert μιας κβαντικής συσκευής, αυτός ο τρόπος είναι

καλύτερος. Αυτή η κωδικοποίηση απαιτεί  $\log_2(n)$  qubits για την αναπαράσταση ενός σημείου δεδομένων  $n$ -διαστάσεων. Τέλος, θα πρέπει κανείς να συμπεριλάβει την κωδικοποίηση γωνίας, ή αλλιώς την κωδικοποίηση προϊόντων Tensor, ως μια από τις πιο γνωστές τεχνικές. Τα  $n$  κλασικά χαρακτηριστικά κωδικοποιούνται στη γωνία περιστροφής του  $n$  qubit. Απαιτεί μία περιστροφή σε κάθε qubit. Αυτή η μέθοδος είναι κυρίως χρήσιμη κατά την επεξεργασία δεδομένων σε κβαντικά νευρωνικά δίκτυα. Προτού τα δεδομένα μας είναι μια συμβατή είσοδος στο νευρωνικό μας δίκτυο, εφαρμόζουμε μια πύλη περιστροφής πάνω από τον άξονα  $x$  ή  $y$ ,  $R_X$  ή  $R_Y$ , αντίστοιχα. Μαθηματικά, αυτοί οι μετασχηματισμοί μπορούν να γραφτούν ως οι παρακάτω πίνακες:

$$R_x(\theta) = \begin{pmatrix} \cos(\frac{\theta}{2}) & -i\sin(\frac{\theta}{2}) \\ -i\sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2}) \end{pmatrix} \quad (6.6)$$

$$R_y(\theta) = \begin{pmatrix} \cos(\frac{\theta}{2}) & -\sin(\frac{\theta}{2}) \\ \sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2}) \end{pmatrix} \quad (6.7)$$

Όπως έχει παρατηρηθεί πειραματικά, αυτή η μέθοδος είναι πολύ αποτελεσματική στις λειτουργίες. Αυτό συμβαίνει επειδή, ανεξάρτητα από την ποσότητα των τιμών δεδομένων που πρέπει να κωδικοποιηθούν, απαιτείται μόνο ένας σταθερός αριθμός παράλληλων διεργασιών. Ωστόσο, επειδή κάθε στοιχείο διάνυσματος εισόδου παίρνει ένα qubit, αυτό είναι επίσης ακριβό από την άποψη των qubits.

Εκτός από τις παραπάνω βασικές και πιο γενικά χρησιμοποιούμενες τεχνικές, χρησιμοποιούνται και άλλες πιο εξειδικευμένες ή συνδυασμένες τεχνικές. Πρώτον, υπάρχει η μέθοδος κωδικοποίησης Qsample, η οποία συνδυάζει τις μεθόδους κωδικοποίησης βάσης και πλάτους. Συσχετίζει ένα διάνυσμα πραγματικού πλάτους με κλασικές διακριτές κατανομές πιθανοτήτων. Το πλεονέκτημα είναι ότι χρησιμοποιούμε κυρίως τη λογική πλάτους, αλλά όλα τα χαρακτηριστικά κωδικοποιούνται στο qubit. Έτσι, μπορούμε να υποθέσουμε ότι οι περισσότερες από τις παραπάνω επιλογές είναι κατάλληλες για χρήση σε συγκεκριμένες περιπτώσεις και η τελική επιλογή για την τεχνική κωδικοποίησης είναι πολύ σημαντική καθώς επηρεάζει το χρόνο εκτέλεσης της διαδικασίας φόρτωσης.

Σε ένα παράδειγμα περίπτωσης, οι ερευνητές δοκίμασαν έναν αλγόριθμο συμπίεσης descriptor για την Κβαντική Μηχανική Εκμάθηση Δεδομένων Sars-Cov-2. Χρησιμοποίησαν ένα MF για την παραγωγή δυαδικών αριθμών με προεπιλογή 2048 bit σε αυτόν τον αλγόριθμο. Ένας μεγάλος αριθμός περιγραφών, καθώς και η χρήση qubits στο δίκτυο Quantum Computing για την αναπαράστασή τους, μπορεί να είναι μια πρόκληση. Στις περισσότερες περιπτώσεις, ο θόρυβος αποσυνοχής που εισάγεται στο σύστημα qubit προκαλεί αποτυχία της ακρίβειας. Εάν υπάρχει έλλειψη σύνδεσης μεταξύ ορισμένων qubits, μπορεί να εγχυθεί κάποιος πρόσθετος θόρυβος στο σύστημα με βάση την αρχιτεκτονική του κβαντικού υπολογιστή. Τα χαρακτηριστικά του περιγραφέα 2048 κωδικοποιήθηκαν χρησιμοποιώντας τις ακόλουθες τέσσερις μεθόδους:

- ▶ PCA,
- ▶ Κοινή τεχνική μείωσης διαστάσεων γραμμικής διακριτικής ανάλυσης(LDA)
- ▶ Διαιρέστε 2048 bit δακτυλικών αποτυπωμάτων μορίων σε ομάδες "x", έτσι ώστε κάθε ομάδα να έχει bits "k". Ο αριθμός των bit θα πρέπει να διαιρεί πλήρως το 2048. Στη συνέχεια τα bit "k" μετατρέπονται σε τιμή βάσης 10 ή δεκαδική τιμή. Επαναλάβετε μέχρι να μετατραπούν όλες οι ομάδες σε δεκαδικό.
- ▶ Παρακολουθήστε τις θέσεις του 1 σε ολόκληρο τον πίνακα.

Μέσα στον κβαντικό υπολογιστή, χρησιμοποιήθηκε ένας κβαντικός αλγόριθμος για την επίλυση του άμεσου γινόμενου για πράξεις μήτρας και στη συνέχεια τον υπολογισμό του πίνακα M. Στη συνέχεια, ένας κβαντικός αλγόριθμος μπορεί να χρησιμοποιηθεί για τη μετατροπή σε κυματομορφές για την επίλυση της πλήρους Μηχανής Διανυσμάτων Υποστήριξης σε έναν κβαντικό υπολογιστή. Ενώ το SVM παρέχει γενικά πολλά υποσχόμενα αποτελέσματα, χρειάζεται πολύς χρόνος για την επίλυση γραμμικών εξισώσεων για την επίλυση του πίνακα πυρήνα χρησιμοποιώντας χάρτες χαρακτηριστικών.

Τέλος, στο παρακάτω διάγραμμα, μπορεί κανείς να δει πώς λειτούργησε η μείωση διαστάσεων και πώς αποθηκεύτηκαν τα δεδομένα.



Σχήμα 6.3: Data in the Bloch sphere

Γενικά, χρησιμοποιώντας μοντέλα SVM μειώνουμε τον τρισδιάστατο χώρο σε 2 διαστάσεις. Στη συγκεκριμένη περίπτωση, κατάφεραν να μειώσουν σε 3 έως 10 διαστάσεις για να εξετάσουν την αρχιτεκτονική qubit. Στο παραπάνω διάγραμμα, μπορείτε να βρείτε μια υψηλού επιπέδου αφαίρεση του τρόπου αποθήκευσης των δεδομένων στη σφαίρα Bloch. Τα δύο στρώματα, το A και το B με το  $U(x_A)$  να υποδηλώνει τον ενιαίο πίνακα που εφαρμόζεται στο διάνυσμα εισόδου και το  $U(\theta_A)$  να αντιπροσωπεύει τον ενιαίο πίνακα που εφαρμόζεται για την περιστροφή του διανύσματος στη σφαίρα Bloch.

Όπως αναφέρθηκε προηγουμένως, η μείωση διαστάσεων είναι μια κρίσιμη εργασία που μπορεί να επιτευχθεί χρησιμοποιώντας νευρωνικά δίκτυα. Για τη μετατροπή δεδομένων υψηλών διαστάσεων σε κώδικες χαμηλής διάστασης, ένα νευρωνικό δίκτυο πολλαπλών επιπέδων με ένα μικροσκοπικό κεντρικό στρώμα μπορεί να εκπαιδευτεί για την ανακατασκευή διανυσμάτων εισόδου υψηλών διαστάσεων. Η ανάλυση βασικών στοιχείων (PCA) είναι μια απλή και ευρέως χρησιμοποιούμενη μέθοδος στην κλασική προσέγγιση, η οποία ανακαλύπτει τις κατευθύνσεις της υψηλότερης διακύμανσης στο σύνολο δεδομένων και αναπαριστά κάθε σημείο δεδομένων με τις συντεταγμένες του κατά μήκος καθεμιάς από αυτές τις κατευθύνσεις. Ένα δίκτυο κωδικοποιητών πολλαπλών επιπέδων μετατρέπει δεδομένα υψηλών διαστάσεων σε κώδικα χαμηλής διάστασης και ένα παρόμοιο δίκτυο αποκωδικοποιητή ανακτά τα δεδομένα από τον κώδικα, είναι μια γενικευμένη προσέγγιση. Για αυτά τα δίκτυα, οι απαιτούμενες διαβαθμίσεις αποκτώνται εύκολα χρησιμοποιώντας τον κανόνα της αλυσίδας για την αντίστροφη διάδοση των παραγώγων σφαλμάτων πρώτα μέσω του δικτύου αποκωδικοποιητή και στη συνέχεια μέσω του δικτύου κωδικοποιητών. Μετά από αυτό, ένα σύνολο δυαδικών διανυσμάτων μπορεί να μοντελοποιηθεί χρησιμοποιώντας ένα δίκτυο δύο επιπέδων, την περιορισμένη μηχανή Boltzmann, στην οποία στοχαστικά, δυαδικά pixel συνδέονται με στοχαστικούς, δυαδικούς ανιχνευτές χαρακτηριστικών χρησιμοποιώντας συμμετρικά σταθμισμένες συνδέσεις. Για να συνεχίσουμε, το δίκτυο εκχωρεί μια πιθανότητα σε κάθε δυνατή εικόνα μέσω της συνάρτησης ενέργειας. Μια πιθανή κοινή διαμόρφωση  $(v, h)$  των ορατών και κρυφών μονάδων έχει ενέργεια ίση με:

$$E(v, h) = - \sum_{i \in \text{pixels}} b_i v_i - \sum_{j \in \text{features}} b_j h_j - \sum_{i, j} v_i h_j w_{ij} \quad (6.8)$$

όπου  $v_i, h_j$  είναι οι δυαδικές καταστάσεις του pixel  $i$  και το χαρακτηριστικό  $j$ ,  $b_i, b_j$  είναι οι προκαταλήψεις τους και  $w_{ij}$  είναι το βάρος μεταξύ τους.

Στη μοντελοποίηση που χρησιμοποιήθηκε για αυτή τη μελέτη, ήταν ένα ενιαίο επίπεδο δυαδικών χαρακτηριστικών, το οποίο χρησιμοποιήθηκε για εκμάθηση και τα αποτελέσματα χρησιμοποιήθηκαν ως δεδομένα για την εκμάθηση ενός δεύτερου επιπέδου χαρακτηριστικών. Αυτή η διαδικασία εκμάθησης επίπεδο προς επίπεδο μπορεί να εκτελεστεί όσες φορές χρειάζεται. Κάθε επίπεδο χαρακτηριστικών καταγράφει συσχετισμούς υψηλής τάξης μεταξύ των ενεργειών των μονάδων στο παρακάτω επίπεδο. Στη συνέχεια, το παγκόσμιο στάδιο λεπτομέρειας αντικαθιστά τη στοχαστική δραστηριότητα με ντετερμινιστική πιθανότητα πραγματικής αξίας και χρησιμοποιεί αντίστροφη διάδοση για να τελειοποιήσει τα βάρη για βέλτιστη ανακατασκευή σε ολόκληρο τον αυτόματο κωδικοποιητή. Η ταξινόμηση και η παλινδρόμηση μπορούν και οι δύο να ωφεληθούν από την προ-προπόνηση επίπεδο προς επίπεδο. Η προ-εκπαίδευση βρέθηκε ότι βοηθά στη γενίκευση διασφαλίζοντας ότι η πλειονότητα των πληροφοριών στα βάρη προέρχεται από τη μοντελοποίηση των εικόνων ή, στην περίπτωσή μας, από τις δομές του γραφήματος.

### 3. Μέθοδος Angle Encoding

Λόγω της φύσης του προβλήματός μας, πιθανότατα θα επιλέξουμε να χρησιμοποιήσουμε τη μέθοδο κωδικοποίησης γωνίας ως την καταλληλότερη για να κωδικοποιήσουμε τα δεδομένα μας στο νευρωνικό μας δίκτυο Quantum Graph. Επομένως, πρέπει να εμβαθύνουμε σε αυτήν την τεχνική για να κατανοήσουμε πώς να την εφαρμόσουμε και να διασφαλίσουμε ότι δεν θα έχουμε καμία διαρροή δεδομένων σε αυτό το βήμα της διαδικασίας. Η κωδικοποίηση γωνίας χρησιμοποιεί πύλες περιστροφής για την κωδικοποίηση κλασικών πληροφοριών  $x$ . Οι κλασικές πληροφορίες καθορίζουν τις γωνίες των πυλών περιστροφής, όπως φαίνεται παρακάτω:

$$|x\rangle = \otimes_i^n R(x_i)|0^n\rangle \quad (6.9)$$

όπου  $R$  μπορεί να είναι περιστροφή πάνω από οποιονδήποτε άξονα,  $R_x, R_y, R_z$ . Συνήθως, ο αριθμός των qubits που χρησιμοποιούνται για την κωδικοποίηση είναι ίσος με τη διάσταση του διανύσματος  $x$ . Ωστόσο, θα πρέπει πάντα να έχουμε κατά νου ότι κάθε κωδικοποίηση είναι ουσιαστικά μια ανταλλαγή μεταξύ τριών βασικών πτυχών.

- ▶ ο αριθμός των qubits πρέπει να είναι ελάχιστος,
- ▶ ο αριθμός των παράλληλων λειτουργιών πρέπει να είναι ελάχιστος για να ελαχιστοποιηθεί το πλάτος του κβαντικού κυκλώματος,
- ▶ τα δεδομένα πρέπει να παρουσιάζονται κατάλληλα για περαιτέρω υπολογισμούς.

Κατανοώντας αυτό το αντάλλαγμα, ίσως είναι πιο κατάλληλο σε ορισμένες περιπτώσεις να δοκιμάσετε την τεχνική κωδικοποίησης πυκνής γωνίας. Αυτή η υποκατηγορία εκμεταλλεύεται μια πρόσθετη ιδιότητα των qubits, τη σχετική φάση, για να χρησιμοποιήσει τα μόνα  $n/2$  qubits για να κωδικοποιήσει  $n$  σημεία δεδομένων. Αυτό θα μπορούσε να είναι πιο χρήσιμο σε πραγματικές περιπτώσεις, ενώ οι περισσότεροι κβαντικοί υπολογιστές εξακολουθούν να έχουν περιορισμένο αριθμό qubits. Στην κωδικοποίηση πυκνής γωνίας, δεδομένου ενός διανύσματος χαρακτηριστικών  $\vec{x} = [x_1, \dots, x_N]^T R^N$ , η διαδικασία που αντιστοιχίζει  $\vec{x}E(\vec{x})$  δίνεται από

$$|\vec{x}\rangle = \otimes_{i=1}^{\lfloor N/2 \rfloor} \cos(\pi x_{2i-1})|0\rangle + e^{2\pi i x_{2i}} \sin(\pi x_{2i-1})|1\rangle \quad (6.10)$$

Και όπως αναφέρθηκε παραπάνω, μπορούμε να αναφερθούμε στην κωδικοποίηση πυκνής γωνίας ως τη διαδικασία κωδικοποίησης δύο χαρακτηριστικών ανά qubit εκμεταλλευόμενοι τον σχετικό βαθμό ελευθερίας φάσης, μπορούμε επίσης να

δοκιμάσουμε κωδικοποίηση πυκνής γωνίας για δισδιάστατα δεδομένα  $\vec{x} \in R^2$  με ένα μόνο qubit που δίνεται από

$$|\vec{x}\rangle = \cos(\pi x_1)|0\rangle + e^{2\pi i x_2} \sin(\pi x_1)|1\rangle \quad (6.11)$$

με μήτρα πυκνότητας

$$\rho_x = \begin{pmatrix} \cos^2(\pi x_1) & e^{-2\pi i x_2} \cos(\pi x_1) \sin(\pi x_1) \\ e^{2\pi i x_2} \cos(\pi x_1) \sin(\pi x_1) & \sin^2(\pi x_1) \end{pmatrix} \quad (6.12)$$

Τέλος, η ακραία έκδοση αυτής της διαδικασίας ονομάζεται Superdense Angle Κωδικοποίηση(SDAE). Έστω  $\vec{x} = [x_1, \dots, x_N]^T \in R^N$  διάνυσμα χαρακτηριστικών και παράμετροι  $\theta, \phi \in R^N$ . Στη συνέχεια, η υπερπυκνή γωνία που κωδικοποιεί αντιστοιχίζει το  $\vec{x}E(\vec{x})$  που δίνεται από

$$|x\rangle = \otimes_{i=1}^{\lfloor N/2 \rfloor} \cos(\theta_i x_{2i-1} + \phi_i x_{2i})|0\rangle + \cos(\theta_i x_{2i-1} + \phi_i x_{2i})|1\rangle \quad (6.13)$$

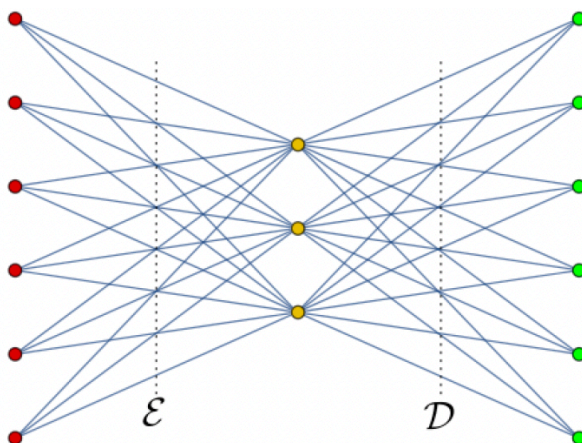
Βλέπουμε ότι αυτό το μοντέλο περιλαμβάνει δύο νέες υπερπαραμέτρους, το  $\theta$  και το  $\phi$ . Αυτό συμβαίνει για να βελτιστοποιηθεί η διαδικασία και να αυξηθεί η ευρωστία. Τώρα, όπως αναφέρεται στη βιβλιογραφία, η κωδικοποίηση πυκνής γωνίας φαίνεται να αποδίδει καλά σε όλα τα σύνολα δεδομένων και είναι ικανή να προσαρμόζεται καλά στο θόρυβο, ακόμη και να έχει καλύτερη απόδοση η ιδανική περίπτωση χωρίς θόρυβο και σταθερή κωδικοποίηση. Από μια άλλη άποψη, η κωδικοποίηση υπερπυκνής γωνίας δεν αποδίδει καλά σε κανένα εμφανιζόμενο σύνολο δεδομένων από τότε Το όριο απόφασης που δημιουργείται είναι εξαιρετικά μη γραμμικό και δεν μπορεί να ταξινομήσει σωστά περισσότερο από το μισό σύνολο δεδομένων.

## 4. Μεταβλητοί Auto-encoders

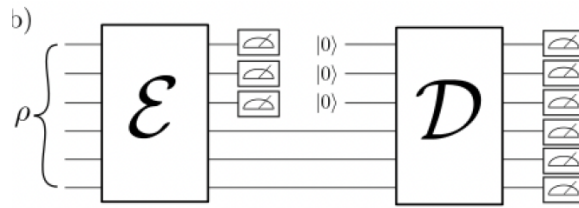
VAEs (Variational Autoencoders) είναι ευέλικτα μοντέλα παραγωγής που μπορούν να χρησιμοποιηθούν σε μια ποικιλία μέσων. Ένα συμβατικό δίκτυο αυτόματου κωδικοποιητή μπορεί να περιγραφεί ως ένα ζεύγος δύο συνδεδεμένων δικτύων, του κωδικοποιητή και του αποκωδικοποιητή, αν θέλουμε να το αναλύσουμε. Το δίκτυο κωδικοποιητή μετατρέπει μια είσοδο σε μια μικρότερη, πιο πυκνή αναπαράσταση που μπορεί να χρησιμοποιήσει το δίκτυο αποκωδικοποιητή για να τη μετατρέψει ξανά στην αρχική είσοδο. Εξετάσαμε τα Συνελικτικά Δίκτυα Γραφημάτων νωρίτερα σε αυτό το κεφάλαιο. Ως αποτέλεσμα, μπορούμε να συμπεράνουμε ότι η γενική λειτουργία των CNN είναι επίσης καλά κατανοητή. Εάν έχουμε ένα πείραμα



με εικόνες ως είσοδο, το CNN θα πάρει τη μεγάλη εικόνα και θα τη "συμπίεσει" σε μια πιο συμπαγή και πυκνή αναπαράσταση, την οποία στη συνέχεια θα χρησιμοποιήσει για να κατηγοριοποιήσει την εικόνα μέσω ενός πλήρως συνδεδεμένου δικτύου ταξινομητή. Οι κωδικοποιητές λειτουργούν με παρόμοιο τρόπο. Μετά από αυτό, το δίκτυο παράγει μια πολύ μικρότερη αναπαράσταση με αρκετές πληροφορίες ώστε το επόμενο στοιχείο του δικτύου να το μετατρέψει στην απαιτούμενη μορφή εξόδου. Οι κωδικοποιητές, από την άλλη πλευρά, εκπαιδεύονται σε συνδυασμό με άλλα τμήματα του δικτύου και στη συνέχεια βελτιστοποιούνται μέσω backpropagation για να παρέχουν κωδικοποιήσεις κατάλληλες για την εκάστοτε εργασία. Εάν χρησιμοποιήσουμε έναν αυτόματο κωδικοποιητή, το σύστημα θα δώσει εντολή στον κωδικοποιητή να δημιουργήσει κωδικοποιήσεις που είναι ιδιαίτερα ωφέλιμες για την ανακατασκευή της δικής του εισόδου. Όπως αναφέρθηκε προηγουμένως, το πλήρες δίκτυο συνήθως εκπαιδεύεται μαζί. Η απώλεια ανακατασκευής είναι μια ποινή που επιβάλλεται στο δίκτυο για την παραγωγή εξόδων που διαφέρουν από την είσοδο και είναι συνήθως το μέσο τετράγωνο σφάλμα ή διασταυρούμενη εντροπία μεταξύ της εξόδου και της εισόδου. Τώρα που έχουμε αποδείξει ότι η διαίσθηση όλων των κωδικοποιητών είναι να μειώσουν την είσοδο σε μια μικρότερη αναπαράσταση του εαυτού του, ο κωδικοποιητής πρέπει να επιλέξει να διαγράψει πληροφορίες σε αυτό το σενάριο. Ο κωδικοποιητής μαθαίνει να διατηρεί όσο το δυνατόν περισσότερες σχετικές πληροφορίες στην περιορισμένη κωδικοποίηση και να απορρίπτει έξυπνα άσχετα μέρη. Στη συνέχεια, ο αποκωδικοποιητής μαθαίνει να παίρνει την κωδικοποίηση και να την αναδομεί σωστά σε πλήρη εικόνα. Αν συνδυάσουμε τα παραπάνω μέρη, λαμβάνουμε το σύστημα ενός αυτόματου κωδικοποιητή.



Σχήμα 6.4: Variational Auto-Encoder



Σχήμα 6.5: VAE Circuit

Στην πρώτη εικόνα παραπάνω, παρουσιάζουμε μια γραφική αναπαράσταση του αυτόματου κωδικοποιητή έξι bit με ένα λανθάνοντα χώρο τριών bit. Ο χάρτης  $T$  κωδικοποιεί μια είσοδο έξι bit, τα οποία εμφανίζονται με κόκκινα σημεία, σε μια ενδιάμεση κατάσταση τριών bit, η οποία συμβολίζεται με κίτρινα σημεία στο κέντρο, μετά την οποία ο αποκωδικοποιητής  $D$  προσπαθεί να ανακατασκευάσει τα bit εισόδου στην έξοδο, που εμφανίζεται ως πράσινα σημεία. Στη δεύτερη εικόνα, μπορεί κανείς να βρει την υλοποίηση του κυκλώματος που αντιστοιχεί στον παραπάνω κβαντικό αυτόματο κωδικοποιητή 6-3-6. Το πρόβλημα με τους συνηθισμένους αυτόματους κωδικοποιητές, από την άλλη πλευρά, μπορεί να βρεθεί στο στάδιο της παραγωγής. Το θέμα είναι ότι ο λανθάνοντας χώρος στον οποίο ο αυτόματος κωδικοποιητής μεταφράζει τις εισόδους του και όπου αποθηκεύονται τα κωδικοποιημένα διανύσματά του μπορεί να μην είναι συνεχής ή να επιτρέπει εύκολη παρεμβολή. Αυτό είναι ένα σημαντικό πρόβλημα, καθώς θέλουμε να κάνουμε τυχαία δειγματοληψία από τον λανθάνοντα χώρο ή να παράγουμε παραλλαγές σε μια εικόνα εισόδου από έναν συνεχή λανθάνοντα χώρο κατά τη δημιουργία μοντέλων παραγωγής. Εάν υπάρχουν ασυνέχειες στο χώρο και δειγματιστούμε ή δημιουργήσουμε μια παραλλαγή από εκεί, ο αποκωδικοποιητής θα παράγει απλώς μια απίθανη έξοδο επειδή δεν έχει ιδέα πώς να αντιμετωπίσει αυτό το τμήμα του λανθάνοντος χώρου.

Σε αυτό το σημείο είναι καιρός να εισαγάγουμε Variational Autoencoders. Αυτός ο τύπος αυτόματων κωδικοποιητών έχει μια μοναδική ιδιότητα που τους διαχωρίζει από τους τυπικούς αυτοκωδικοποιητές και που τους καθιστά καταλληλότερους για γενετική μοντελοποίηση. Οι λανθάνοντες χώροι τους είναι σχεδιασμένοι να είναι συνεχείς, επιτρέποντας εύκολη τυχαία δειγματοληψία και παρεμβολή. Αυτό επιτυγχάνεται επιβάλλοντας στην έξοδο του κωδικοποιητή ένα διάνυσμα μέσου όρου  $\mu$  και ένα διάνυσμα τυπικών αποκλίσεων  $\sigma$ . Παρατηρούμε ότι η διαφορά από τους τυπικούς αυτοκωδικοποιητές είναι ότι τα διανύσματα που παράγονται διπλασιάζονται αλλά κάθε διάνυσμα έχει το ίδιο μέγεθος με αυτό που θα είχε το διάνυσμα εξόδου από τον τυπικό κωδικοποιητή. Η γενιά είναι στοχαστική και με αυτόν τον σχεδιασμό μπορεί κανείς να επιτύχει ότι η κωδικοποίηση θα ποικίλλει ένα κάθε πέρασμα λόγω της δειγματοληψίας πάνω από την είσοδο. Αυτή η κωδικοποίηση δείγματος που στη συνέχεια μεταβιβάζεται στον αποκωδικοποιητή επιτυγχάνεται με την εκμετάλ-

λευση των παραμέτρων ενός διανύσματος που περιλαμβάνει τυχαίες μεταβλητές μήκους ίδια με τα αρχικά διανύσματα, χρησιμοποιώντας τα  $\mu, \sigma$  της  $i$ -ης τυχαίας μεταβλητής, από που γίνεται η δειγματοληψία. Ο μέσος έλεγχος του διανύσματος ελέγχει όπου πρέπει να επικεντρώνεται η κωδικοποίηση μιας εισόδου, ενώ η τυπική απόκλιση ελέγχει την επιτρεπόμενη απόκλιση από τη μέση τιμή. Καθώς οι κωδικοποιήσεις είναι τυχαία από οπουδήποτε μέσα στη διανομή, ο αποκωδικοποιητής μαθαίνει ότι όχι μόνο ένα μόνο σημείο στον λανθάνοντα χώρο αναφέρεται σε ένα δείγμα αυτής της κατηγορίας, αλλά όλα τα κοντινά σημεία αναφέρονται στο ίδιο όπως και στο άλλο. Αυτό επιτρέπει στον αποκωδικοποιητή όχι μόνο να αποκωδικοποιεί μεμονωμένες, συγκεκριμένες κωδικοποιήσεις στον λανθάνοντα χώρο, αλλά και μικρές αποκλίσεις του εαυτού του, καθώς ο αποκωδικοποιητής εκτίθεται σε μια σειρά παραλλαγών της κωδικοποίησης της ίδιας εισόδου κατά τη διάρκεια της εκπαίδευσης.

Η ικανότητα χρήσης παρεμβολής μεταξύ κλάσεων είναι βασικό χαρακτηριστικό των VAE. Επειδή οι τιμές των  $\mu, \sigma$  δεν έχουν περιορισμούς, ο κωδικοποιητής μπορεί να μάθει να δημιουργεί εξαιρετικά διαφορετικά  $\mu$  για ξεχωριστές κλάσεις, ομαδοποιώντας τις και ελαχιστοποιώντας τις  $\sigma$ , διασφαλίζοντας ότι οι ίδιες οι κωδικοποιήσεις δεν διαφέρουν σημαντικά από τα δείγματα. Αυτό επιτρέπει στον αποκωδικοποιητή να επανασυναρμολογήσει τα δεδομένα εκπαίδευσης γρήγορα. Η απόκλιση Kullback-Leibler (KL) πρέπει να εισαχθεί στη συνάρτηση απώλειας για να εξαναγκαστούν ομαλές παρεμβολές και να επιτραπεί η δημιουργία νέων δειγμάτων. Αυτή η μέτρηση μετριέται μεταξύ δύο κατανομών πιθανότητας και απλώς μετρά πόσο αποκλίνουν μεταξύ τους. Εάν θέλουμε να βελτιστοποιήσουμε τις παραμέτρους κατανομής πιθανότητας,  $\mu, \sigma$  ώστε να μοιάζουν πολύ με αυτές της κατανομής στόχου, πρέπει να προσπαθήσουμε να ελαχιστοποιήσουμε την απόκλιση KL. Για VAE, η απώλεια KL ισοδυναμεί με το άθροισμα όλων των αποκλίσεων KL μεταξύ του στοιχείου  $X_i N(\mu_i, \sigma_i^2)$  στο  $X$  και του τυπικού κανονικού. Φτάνει στο ελάχιστο για  $\mu_i = 0$  και  $\sigma_i = 1$ . Αυτό το μέτρο ταιριάζει άριστα στους αυτόματους κωδικοποιητές, καθώς ενθαρρύνει τον κωδικοποιητή να κατανέμει ομοιόμορφα όλες τις κωδικοποιήσεις γύρω από το κέντρο του λανθάνοντος χώρου. Η καθαρή απώλεια KL προκαλεί την πυκνή τοποθέτηση κωδικοποιήσεων τυχαία γύρω από το κέντρο του λανθάνοντος χώρου, με ελάχιστο σεβασμό για την ομοιότητα μεταξύ των κοντινών κωδικοποιήσεων. Είναι δύσκολο για τον αποκωδικοποιητή να αποκρυπτογραφήσει οτιδήποτε σημαντικό από αυτήν την περιοχή. Ως αποτέλεσμα, αυτή η διαδικασία χρειάζεται βελτίωση. Αυτό παράγει έναν λανθάνοντα χώρο που, σε τοπική κλίμακα, διατηρεί την ομοιότητα των γύρω κωδικοποιήσεων με ομαδοποίηση που συμπίεζεται σφιχτά γύρω από την προέλευση του λανθάνοντος χώρου. Ακολουθώντας αυτή τη στρατηγική, μπορούμε τελικά να δημιουργήσουμε ξεχωριστά συμπλέγματα που μπορεί να αποκωδικοποιήσει ο αποκωδικοποιητής. Το επιπλέον πλεονέκτημα είναι ότι κατά την παρεμβολή, δεν υπάρχουν ξαφνικά κενά μεταξύ των συστάδων, καθώς έχουμε έναν ομαλό συνδυασμό χαρακτηριστικών που μπορεί να

καταλάβει ένας αποκωδικοποιητής.

## 5. Quantum VAEs

Κάποιος μπορεί να χρησιμοποιήσει μία από τις προσεγγίσεις που περιγράφονται παραπάνω, ή οποιονδήποτε βασικό κβαντικό αυτόματο κωδικοποιητή που μπορεί να ενσωματώσει κάποιες διαδικασίες ενσωμάτωσης, για την κωδικοποίηση κλασικών δεδομένων σε qubits. Οι συγκεκριμένες επιλογές κβαντικής ενσωμάτωσης και μέτρησης καθορίζονται κυρίως από τις διαστάσεις και τις κλίμακες του συνόλου δεδομένων και η βέλτιστη επιλογή είναι συχνά ένα υβριδικό μοντέλο, καθώς οι καθαροί κβαντικοί κωδικοποιητές ισχύουν μόνο για ανακατασκευή δεδομένων κανονικοποιημένης κλίμακας. Ο λόγος για αυτό είναι ότι το ανώτερο όριο τόσο της προσδοκίας όσο και της πιθανότητας είναι 1.

Κάποιος μπορεί να ξεκινήσει με το BQ-BVAE/AE που υιοθετεί την ενσωμάτωση πλάτους και την έξοδο προσδοκίας για τον κωδικοποιητή και την ενσωμάτωση γωνίας και την έξοδο πιθανότητας για τον αποκωδικοποιητή. Αυτός ο αυτόματος κωδικοποιητής έχει ήδη χρησιμοποιηθεί στο σύνολο δεδομένων QM9. Χρησιμοποιήθηκαν δύο παραλλαγές του ίδιου μοντέλου, η πρώτη ήταν πλήρως βασικά κβαντικά VAEs (F-BQ-VAEs) και η δεύτερη ήταν μια υβριδική κβαντική παραλλαγή (H-BQ-VAEs). Η κύρια σύγκριση της απόδοσής τους αποδείχθηκε πειραματικά όπως φαίνεται:

TABLE I  
COMPARISON OF NUMBER OF TRAINABLE PARAMETERS.

Parameter Type	VAE(AE)	F-BQ-VAE(AE)	H-BQ-VAE(AE)
Quantum	0 (0)	108 (108)	108 (108)
Classical	5694 (5610)	84 (0)	4286 (4202)
Total	5694 (5610)	192 (108)	4394 (4310)

### Σχήμα 6.6: Autoencoders

Φάνηκε ότι το BQ-VAE/AE έμαθε γρηγορότερα για κανονικοποιημένα μόρια QM9 ως συνάρτηση των εποχών εκπαίδευσης. Η άλλη περίπτωση που μελετήθηκε είναι η κατηγορία των κλιμακωτών κβαντικών αυτοκωδικοποιητών.

Σε αυτό το υποκεφάλαιο, στοχεύουμε να εστιάσουμε στο κβαντικό ανάλογο

του προηγούμενου κεφαλαίου. Η λογική πίσω από έναν κβαντικό αυτόματο κωδικοποιητή είναι ότι τέτοια συστήματα μας επιτρέπουν να εκτελούμε ανάλογες εργασίες μηχανικής μάθησης για κβαντικά συστήματα χωρίς εκθετικά δαπανηρή κλασική μνήμη, όπως η μείωση της διαστάσεων των δεδομένων. Το μοντέλο δεν απέχει πολύ από την κλασική πραγματικότητα, καθώς έχει αποδειχθεί ότι για ένα συγκεκριμένο μοντέλο, αν επιλέξουμε μια συγκεκριμένη ρύθμιση παραμέτρων στο κβαντικό δίκτυο, ανάγεται ακριβώς σε κλασικό νευρωνικό δίκτυο. Κάθε κόμβος σε ένα απλό σύστημα κβαντικού αυτόματο κωδικοποιητή αντιπροσωπεύει ένα qubit, με τον καταχωρητή εξόδου να αντιπροσωπεύεται από το πρώτο επίπεδο του δικτύου. Ένας ενιαίος μετασχηματισμός από το ένα στρώμα στο επόμενο αντιπροσωπεύεται από τις ακμές που συνδέουν γειτονικά στρώματα. Η εύρεση μονάδων που διατηρούν τις κβαντικές πληροφορίες της εισόδου μέσω του μικρότερου ενδιάμεσου λανθάνοντος χώρου είναι η κύρια εργασία μάθησης για έναν κβαντικό κωδικοποιητή. Μια επιτυχημένη κωδικοποίηση μπορεί να περιγραφεί ως αυτή στην οποία  $F(|\psi_i\rangle, \rho_i^{out}) \approx 1$  για όλες τις καταστάσεις εισόδου, όπου  $F$  είναι η πιστότητα των καταστάσεων και μπορεί να οριστεί μαθηματικά ως

$$F(|\psi_i\rangle, \rho_i^{out}) = \langle \psi_i | \rho_i^{out} | \psi_i \rangle \quad (6.14)$$

Η πιστότητα (fidelity) είναι ένα μέτρο του πόσο μοιάζουν δύο καταστάσεις και εκφράζεται ως η πιθανότητα η μία κατάσταση να περάσει ένα τεστ για να αναγνωριστεί ως η άλλη. Για να επιστρέψουμε στο θέμα αυτού του κεφαλαίου, όπως ορίζεται σε πρόσφατες μελέτες, εάν έχουμε ένα σύνολο  $\{p_i \cdot |\psi_i\rangle_{AB}\}$  καθαρών καταστάσεων σε  $n + k$  qubits, όπου τα υποσυστήματα  $A$  και  $B$  αποτελούνται από  $n$  και  $k$  qubits, αντίστοιχα, τότε μπορούμε να θεωρήσουμε μια οικογένεια ενιαίων τελεστών  $\{U^{\vec{p}}\}$ . Αυτοί οι τελεστές ενεργούν σε  $n + k$  qubits, όπου  $\vec{p} = \{p_1, p_2, \dots\}$  είναι κάποιο σύνολο παραμέτρων που ορίζουν ένα ενιαίο κβαντικό κύκλωμα. Έστω επίσης  $|a\rangle_{B'}$  κάποια σταθερή καθαρή κατάσταση αναφοράς  $k$  qubits. Χρησιμοποιώντας κλασικές μεθόδους μηχανικής μάθησης, θέλουμε να βρούμε τη μονάδα  $U^{\vec{p}}$ , η οποία μεγιστοποιεί τη μέση πιστότητα, την οποία ορίζουμε ως συνάρτηση κόστους

$$C_1(\vec{p}) = \sum_i p_i \cdot F(|\psi_i\rangle, \rho_{i,\vec{p}}^{out}) \quad (6.15)$$

with,

$$\rho_{i,\vec{p}}^{out} = (U^{\vec{p}})_{AB}^+ Tr_B [U_{AB}^{\vec{p}} [\psi_{i_{AB}} \otimes a_{B'}] (U^{\vec{p}_{AB}})^+] (U^{\vec{p}})_{AB'} \quad (6.16)$$

όπου χρησιμοποιήσαμε  $|\psi_i\rangle \langle \psi_i|_{AB} = \psi_{i_{AB}}$  και  $|a\rangle \langle a|_{B'} = a_{B'}$ . Ο κύριος στόχος είναι να βρεθεί η καλύτερη μονάδα  $U^{\vec{p}}$ , η οποία κατά μέσο όρο διατηρεί καλύτερα την κατάσταση εισόδου του κυκλώματος. Μετά από ορισμένες υποθέσεις και υπολογισμούς, υπάρχει επίσης ένας εναλλακτικός ορισμός της συνάρτησης κόστους ως προς την πιστότητα της κατάστασης σκουπιδιών,

$$C_2(\vec{p}) = \sum_i p_i \cdot F(Tr_A [U^{\vec{p}} |\psi_i\rangle \langle \psi_i|_{AB} (U^{\vec{p}})^+], |a\rangle_B) \quad (6.17)$$

Αυτή η εξίσωση διαφέρει από την προηγούμενη, καθώς θεωρείται ότι  $C_1 \leq C_2$ . Ως κατάσταση σκουπιδιών θεωρούμε την κατάσταση με τον πίνακα πυκνότητας  $\rho'_{B'} = Tr_A[|\psi'_i\rangle\langle\psi'_i|_{AB'}]$  και είναι προτιμότερο να δούμε αυτήν την κατάσταση αντί για ανίχνευση πάνω από το σύστημα AB.

Για να είναι αποτελεσματική η υλοποίηση του κβαντικού αυτοκωδικοποιητή, ο αριθμός των παραμέτρων και ο αριθμός των πυλών στο κύκλωμα θα πρέπει να κλιμακώνονται πολυωνυμικά με τον αριθμό των qubit εισόδου. Η άλλη επιλογή για τη δημιουργία του  $U^{\vec{p}}$  είναι η χρήση ενός προγραμματιζόμενου κβαντικού κυκλώματος που αποτελείται από ένα σταθερό δίκτυο πυλών, όπου ένας πολυωνυμικός αριθμός παραμέτρων που σχετίζονται με τις πύλες αποτελεί το  $\vec{p}$ . Το επόμενο βήμα είναι να εκπαιδεύσετε το δίκτυο μεγιστοποιώντας τη συνάρτηση κόστους του αυτόματου κωδικοποιητή,  $C_2$ . Η σχηματική αναπαράσταση της εκπαιδευτικής διαδικασίας μπορεί να απεικονιστεί στο 6.7. Αποτελείται από τα ακόλουθα βήματα:

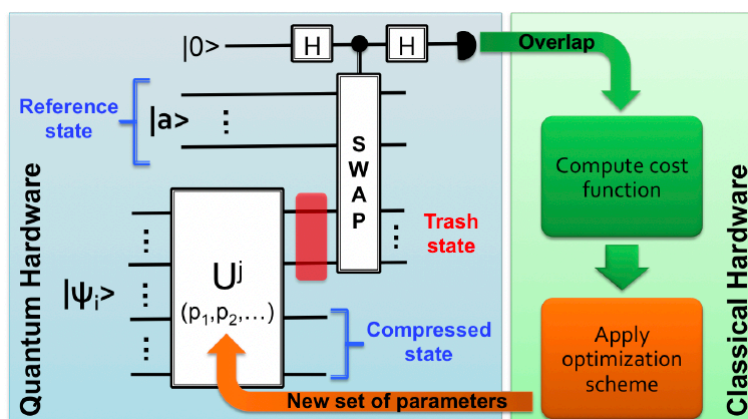
Προετοιμάστε αποτελεσματικά την κατάσταση εισόδου,  $|\psi_i\rangle$  και την κατάσταση αναφοράς.

Εξέλξη κάτω από τη μονάδα κωδικοποίησης,  $U^{\vec{p}}$ , όπου  $\vec{p}$  είναι το σύνολο των παραμέτρων σε ένα δεδομένο βήμα βελτιστοποίησης.

Μετρήστε την πιστότητα μεταξύ της κατάστασης κόδου και της κατάστασης αναφοράς μέσω μιας δοκιμής SWAP. Η εκπαιδευτική προσέγγιση χρησιμοποιεί ένα κβαντικό-κλασικό υβριδικό σχήμα, στο οποίο ο κβαντικός υπολογιστής χειρίζεται την προετοιμασία και τη μέτρηση κατάστασης, ενώ ένας κλασικός υπολογιστής χειρίζεται τη βελτιστοποίηση. Ο σταθμισμένος μέσος όρος πιστότητας μεταξύ της κατάστασης σκουπιδιών που παράγεται από τη συμπίεση και της κατάστασης αναφοράς ορίζεται ως η συνάρτηση κόστους του κβαντικού αυτοκωδικοποιητή. Η συνάρτηση κόστους περιορίζεται από 1 και η βελτιστοποίησή της σημαίνει την ελαχιστοποίηση της τιμής των  $\log_{10}(1 - C_2)$ . Μετά τον υπολογισμό του  $C_2$ , μπορούμε να το τροφοδοτήσουμε σε μια κλασική ρουτίνα βελτιστοποίησης που επιστρέφει ένα νέο σύνολο παραμέτρων για το κύκλωμα συμπίεσης μας. Όλα αυτά τα βήματα επαναλαμβάνονται μέχρι να συγκλίνει ο αλγόριθμος βελτιστοποίησης. Τέλος, όπως περιγράφεται σε αυτή την υποενότητα, η εκπαίδευση θα μπορούσε να υλοποιηθεί με καταστάσεις συγκεκριμένου μεγέθους, που λαμβάνονται από ένα δεδομένο ansatz και στη συνέχεια μπορεί να χρησιμοποιηθεί ως εργαλείο προετοιμασίας κατάστασης. Μόλις το σύστημα έχει εκπαιδευτεί να συμπιέζει ένα συγκεκριμένο σύνολο καταστάσεων, η μονάδα αποσυμπίεσης ( $U^+$ ) μπορεί να χρησιμοποιηθεί για τη δημιουργία καταστάσεων παρόμοιες με αυτές που χρησιμοποιήθηκαν αρχικά για εκπαίδευση. Αυτό μπορεί να επιτευχθεί με την προετοιμασία μιας κατάστασης της μορφής  $|\Psi_I\rangle \otimes |a\rangle$  και την εξέλιξη της κάτω από  $U^+$ , όπου το  $|\Psi_I\rangle$  έχει το

μέγεθος του λανθάνοντος χώρου και το  $|a\rangle$  είναι η κατάσταση αναφοράς που χρησιμοποιείται για την εκπαίδευση. Ωστόσο, ένα σημαντικό στοιχείο κατά τη χρήση κβαντικών αυτοκωδικοποιητών αυτού του τύπου είναι ότι η εντροπία von-Neumann του τελεστή πυκνότητας που αντιπροσωπεύει το σύνολο  $\{p_i, |\psi_i\rangle_{AB}\}$  περιορίζει τον αριθμό των qubits στα οποία μπορεί να συμπιεστεί αθόρυβα.

Όπως φαίνεται στην εικόνα 6.6, οι αυτόματοι κωδικοποιητές συρρικνώνουν το διάστημα μεταξύ του πρώτου και του δεύτερου επιπέδου. Ένα σημαντικό πράγμα που πρέπει να έχετε κατά νου όταν εργάζεστε με ένα δίκτυο αυτόματου κωδικοποιητή, οι κόμβοι εισόδου πρέπει να απορριφθούν μετά την αρχική κωδικοποίηση και στη συνέχεια αρχικοποιούμε ένα νέο σύνολο qubits με βάση μια κατάσταση αναφοράς, το οποίο θα χρησιμοποιηθεί για την υλοποίηση του τελική αποκωδικοποίηση  $D$  που θα συγκριθεί με την αρχική κατάσταση.



Σχήμα 6.7: Training QVAEs

## 6. Quantum Graph Neural Networks

Τα νευρωνικά δίκτυα κβαντικού γραφήματος είναι μια αναδύομενη κατηγορία αλγορίθμων QML που είναι ιδιαίτερα κατάλληλος να εκτελεστεί σε κατανεμημένα κβαντικά συστήματα σε ένα κβαντικό δίκτυο. Τώρα, αν θέλουμε να καταλάβουμε πώς λειτουργεί ένα υβριδικό κβαντικό-κλασικό μοντέλο, μπορούμε να το χρησιμοποιήσουμε ως παράδειγμα για να μεταβούμε από τη λογική ενός κλασικού δικτύου γραφημάτων στον κβαντικό υπολογισμό. Ο στόχος αυτού του GNN είναι να προβλέψει την πιθανότητα κάθε ακμή σε ένα γράφημα συμβάντων να είναι ένα τμήμα διαδρομής. Αυτό γίνεται με την ελπίδα να τροφοδοτηθούν νέα γραφήματα συμβάντων στο εκπαιδευμένο QGNN, το οποίο στη συνέχεια μπορεί να συναγάγει προβλέψεις ακμών που μπορούν να χρησιμοποιηθούν για την πρόβλεψη της συναρμολό-

γησης του φαρμάκου σε μικροσκοπικά μόρια. Ένα δίκτυο ακμών και κόμβων, που μοιράζονται βάρη μεταξύ των επιπέδων και εντός του ίδιου επιπέδου, συνθέτουν το συγκεκριμένο μοντέλο. Επειδή εφαρμόζονται επανειλημμένα, τα δίκτυα ακμών και κόμβων ανταλλάσσουν βάρη μεταξύ των επιπέδων. Αυτό επιστράτη την προσοχή στο γεγονός ότι χρησιμοποιούνται μόνο ένα δίκτυο κόμβων και ένα δίκτυο ακμών και εφαρμόζονται επανειλημμένα, επομένως όλα τα βάρη αυτών των δικτύων μοιράζονται σε όλους τους κόμβους και τα επίπεδα σε κάθε εφαρμογή.

Ξεκινώντας με το δίκτυο εισόδου, περιγράφουμε την είσοδο μας ως μια μήτρα όλων των σημείων δεδομένων ανά κόμβο, η οποία στη συνέχεια μεταφέρεται σε έναν χώρο ενσωμάτωσης υψηλότερης διάστασης χρησιμοποιώντας ένα νευρωνικό δίκτυο μονού επιπέδου. Σε αυτό το σημείο, χρησιμοποιούμε  $N_D$  τον αριθμό των κρυφών διαστάσεων ως υπερπαραμέτρο που υπαγορεύει το μέγεθος όλων των ενσωματώσεων κόμβων και ορίζουμε

$$InputNet(X) = \sigma(W^{(i)}X) \quad (6.18)$$

όπου  $X$  είναι ο πίνακας χαρακτηριστικών εισόδου,  $W^{(i)}$  ο πίνακας βάρους για το πλήρως συνδεδεμένο επίπεδο και  $\sigma$  η συνάρτηση ενεργοποίησης. Τώρα, το σχήμα της εισόδου ορίζεται από τον αριθμό των κόμβων  $N_v$  και τον αριθμό των συντεταγμένων, 3 και επομένως  $dim(input) = (N_v, 3)$ . Όταν παρέχονται οι χωρικές συντεταγμένες κάθε σημείου δεδομένων, το InputNetwork τις μετατρέπει σε χώρο ενσωμάτωσης με διαστάσεις  $N_D$ . Στη συνέχεια, οι ενσωματώσεις κόμβων τροφοδοτούνται μέσω του δικτύου ακμών και στη συνέχεια του δικτύου κόμβων για επαναλήψεις  $N_I$ , που συνδέονται με τις προοπτικές χωρικές συντεταγμένες τους. Με κάθε μετάδοση των συστατικών δικτύων, οι λανθάνουσες μεταβλητές του κόμβου (ενσωματώσεις σε  $N_D$ ) και τα βάρη των ακμών ενημερώνονται εναλλάξ.

Αντίθετα, το σχήμα εξόδου δίνεται από  $dim(output) = (N_v, N_D)$  και η έξοδος αντιπροσωπεύει τις ενσωματώσεις κόμβων για κάθε σημείο δεδομένων.

Στην επόμενη ενότητα που πρέπει να αναλύσουμε είναι το Edge Network. Αυτό το μέρος του μοντέλου στοχεύει να προβλέψει την πιθανότητα ύπαρξης μιας συγκεκριμένης ακμής. Η είσοδος αυτού του δικτύου είναι ένα ζεύγος διάνυσμάτων χαρακτηριστικών κόμβων. Εδώ πρέπει να σημειώσουμε ότι κάθε διάνυσμα χαρακτηριστικών κόμβου είναι η συνένωση των 3 σταθερών χωρικών συντεταγμένων. Η έξοδος είναι απλώς ένας float που αντιπροσωπεύει την πιθανότητα ότι το ζεύγος των κόμβων είναι συνδεδεμένο και λαμβάνουμε  $N_D$  εκπαιδευσιμες τιμές ενσωμάτωσης από αυτό το στάδιο. Εάν το  $h$  είναι το διάνυσμα χαρακτηριστικών στην είσοδο ή η έξοδος στο  $k$ -ο επίπεδο και το  $e_k$  είναι η πιθανότητα στο  $k$ -ο επίπεδο, μπορούμε να ορίσουμε την παρακάτω εξίσωση για να μοντελοποιήσουμε μαθηματικά τον σκοπό του δικτύου ακμών.

$$e_k = EdgeNetwork(h_o^{(k)} + h_i^{(k)}) \quad (6.19)$$



Το Edge Network είναι ένα υβριδικό κβαντικό-κλασικό Νευρωνικό Δίκτυο και αφού ληφθεί ένα ενωμένο διάνυσμα που περιέχει ένα ζεύγος κόμβων, τα δεδομένα τροφοδοτούνται μέσω ενός κβαντικού νευρωνικού δικτύου που βρίσκεται ανάμεσα σε δύο εκπαιδευσιμα κλασικά στρώματα. Για το επίπεδο ταξινόμησης εισόδου γνωρίζουμε ότι ο στόχος είναι να συμπυκνωθεί καλύτερα η διάσταση από  $2(3 + N_D)$  σε  $N_Q$ , που είναι ο αριθμός των qubits. Στη συνέχεια, οι έξοδοι  $N_Q$  επανακλιμακώνονται σε  $[0, \pi]$ , το οποίο στη συνέχεια παραμετροποιεί το κύκλωμα κωδικοποίησης πληροφοριών (IEC). Δυστυχώς, το σχήμα κωδικοποίησης, ανεξάρτητα από το τι είναι, κωδικοποίηση γωνίας ή πλάτους ή κάποιος συνδυασμός μεθόδων κωδικοποίησης, δεν είναι εκπαιδευσιμο. Συνεχίζοντας μετά το βήμα κωδικοποίησης, η κβαντική κατάσταση εξελίσσεται κάτω από το εκπαιδευσιμο παραμετροποιημένο κύκλωμα (PQC). Στη συνέχεια, όλα τα qubits μετρώνται από το PQC για να τροφοδοτηθούν στη συνέχεια σε ένα κλασικό πλήρως συνδεδεμένο στρώμα, το οποίο εξάγει μια τιμή πιθανότητας μίας άκρης χρησιμοποιώντας μια συνάρτηση ενεργοποίησης.

Κάθε διάνυσμα χαρακτηριστικών κόμβου έχει μια ενθυλακωμένη ενσωμάτωση που έχει εκπαιδευτεί για να συμπυκνώνει τις πιο χρήσιμες πληροφορίες για τη λήψη αυτής της απόφασης. Όπως γνωρίζουμε από τη βασική περίπτωση των συνελκτικών NN γραφημάτων, κάθε επίπεδο GNN, σε αυτήν την περίπτωση το δίκτυο κόμβων, εκπαιδευεται ώστε να παρέχει τη βέλτιστη ενσωμάτωση, έτσι ώστε οι δύο ενσωματώσεις κόμβων να μπορούν να χρησιμοποιηθούν μαζί για να προσδιοριστεί εάν υπάρχει ή όχι μια άκρη μεταξύ τους. Αξίζει επίσης να σημειωθεί ότι η τοποθέτηση του κβαντικού νευρωνικού δικτύου μεταξύ δύο κλασικών στρωμάτων μας δίνει μεγαλύτερη ελευθερία με τα σχήματα εισόδου και εξόδου του συγκεντρωτικού δικτύου. Έτσι, η ύπαρξη ενός κλασικού επιπέδου εισόδου μας επιτρέπει να παίζουμε ελεύθερα με το κρυφό μέγεθος της διάστασης, διατηρώντας επίσης ανεξάρτητη ελευθερία ως προς τον αριθμό των qubits.

Όσον αφορά το δίκτυο κόμβων, όλες οι προβλέψεις του δικτύου ακμών είναι ζωτικής σημασίας για την επαναληπτική ενημέρωση των ενσωματώσεων κόμβων, οι οποίες συμπυκνώνουν τις ιδιότητες κρυφών γραφημάτων. Το δίκτυο ακμών του κβαντικού GNN είναι υπεύθυνο για την πρόβλεψη τμημάτων τροχιάς ως υποσυστατικό. Ως αποτέλεσμα, είναι κρίσιμο για το δίκτυο κόμβων. Επιπλέον, έχει αποδειχθεί πειραματικά ότι οι πληροφορίες διαδίδονται σε περισσότερους κόμβους με περισσότερες επαναλήψεις. Καταγράφουμε ευρύτερα χαρακτηριστικά γραφήματος με αυτόν τον τρόπο, επιτρέποντάς του να ενημερώνει τα τοπικά χαρακτηριστικά ενός κόμβου με περισσότερη κατανόηση.

Η είσοδος του δικτύου κόμβων είναι μια συνδυασμένη τριάδα 3 διανυσμάτων χαρακτηριστικών κόμβων:

- Ένα σταθμισμένο άθροισμα σε όλα τα γειτονικά χαρακτηριστικά κόμβου που

βρίσκονται στο επίπεδο ανιχνευτή πριν από το επίπεδο  $h_j$ ,

- ▶ Ένα σταθμισμένο άθροισμα  $h'_{j,output}$  σε όλα τα γειτονικά χαρακτηριστικά κόμβου που βρίσκονται σε ένα επίπεδο ανιχνευτή ακριβώς μπροστά από το επίπεδο  $h_j$ ,
- ▶ το διάνυσμα χαρακτηριστικών του κόμβου στόχου.

Η αρχιτεκτονική του δικτύου κόμβων αποτελείται από ένα επίπεδο εισόδου με διάσταση  $3(3 + N_D)$  και ένα επίπεδο εξόδου διάστασης  $N_D$  και είναι υπεύθυνο για τον περιορισμό της ενσωμάτωσης του κόμβου. Το τελευταίο μέρος του αλγορίθμου που πρέπει να ελεγχθεί, είναι το ansatz που θα χρησιμοποιήσουμε κατά την εκπαίδευση του QGNN.

Έτσι, αν θέλουμε να συνοψίσουμε, το δίκτυο ακμών χρησιμεύει ως λειτουργία προσοχής που τροφοδοτεί πληροφορίες στο δίκτυο κόμβων στο οποίο είναι σημαντικό να προσέχουμε τους γύρω κόμβους. Αυτό βοηθά το δίκτυο ακμών να αναγνωρίσει την ύπαρξη ενός τμήματος διαδρομής, καθώς υπάρχει υψηλότερος συσχετισμός μεταξύ δύο κόμβων που είναι πιο πιθανό να σχηματίσουν μια άκρη.

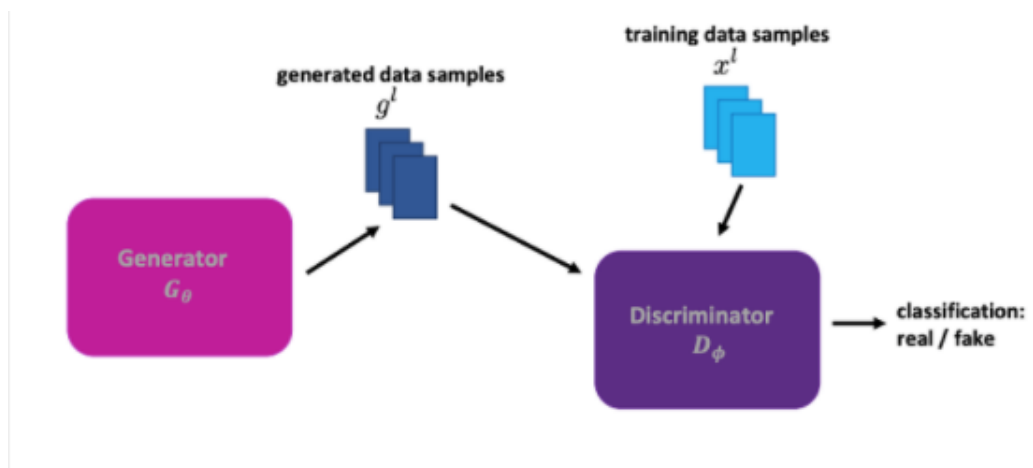
Όταν εκπαιδεύουμε το NN, πρέπει να υπολογίσουμε τις κβαντικές κλίσεις ανεξάρτητα χρησιμοποιώντας αναποτελεσματικούς κανόνες αναλυτικής μετατόπισης παραμέτρων. Στην εκπαιδευτική διαδικασία εκτελούνται τα παρακάτω βήματα:

1. Αρχικά, τροφοδοτούμε μια παρτίδα μεμονωμένων γραφημάτων και πραγματοποιούμε ένα πέρασμα προς τα εμπρός για να φτάσουμε στην προβλεπόμενη δομή ακμών για κάθε μεμονωμένο γράφημα.
2. Στη συνέχεια υπολογίζουμε τη δυαδική απώλεια διασταυρούμενης εντροπίας μεταξύ των προβλεπόμενων τιμών ακμών και των ετικετών.
3. Μετά από αυτό διαδίδουμε πίσω την απώλεια για να βρούμε τις διαβαθμίσεις κάθε κλασικού και κβαντικού βάρους, ακολουθώντας τον κανόνα μετατόπισης παραμέτρων. Οι κλασικές διαβαθμίσεις υπολογίζονται ως συνήθως αξιοποιώντας την αυτόματη διαφοροποίηση.
4. Ως τελευταίο βήμα, ενημερώνουμε κάθε βάρος χρησιμοποιώντας υπολογισμένες κλίσεις με το βελτιστοποιητή της επιλογής μας.
5. Στη συνέχεια επαναλαμβάνουμε τα παραπάνω βήματα για όλες τις παρτίδες σε μια epoch, πάνω από 10-100 epochs.

Στα παραπάνω βήματα, αναφέραμε τη διασταυρούμενη εντροπία ως συνάρτηση απώλειας. Όταν εκπαιδεύουμε το μοντέλο μπορούμε να τροφοδοτήσουμε τις ενσωματώσεις σε οποιαδήποτε συνάρτηση απώλειας και να εκτελέσουμε στοχαστική κλίση για να εκπαιδεύσουμε τις παραμέτρους βάρους όπως υπαγορεύονται από τα υπόλοιπα βήματα. Η συνάρτηση απώλειας θα πρέπει να επιλεγεί με βάση την εγγύτητα του κόμβου στο γράφημα και θα πρέπει να εκπαιδεύσουμε το μοντέλο για μια εποπτευόμενη εργασία, όπως η ταξινόμηση κόμβων.

## 7. Quantum GANs

Το επόμενο επίπεδο είναι να δοκιμάσει κάποιος το κβαντικό ανάλογο αυτού του μοντέλου. Τα QGAN υπακούουν στους ίδιους κανόνες με το κλασικό ανάλογό του, αλλά έχουν πολλές διαφορές, ξεκινώντας από την αρχιτεκτονική:



Σχήμα 6.8: QGANs

Τα QGAN είναι ένα υβριδικό κβαντικό-κλασικό μοντέλο, όπου τουλάχιστον ένα μέρος του βασίζεται σε κβαντικούς αλγόριθμους, είτε είναι ο discriminator είτε ο generator. Σε αυτό το μοντέλο, ο αλγόριθμος χρησιμοποιεί την αλληλεπίδραση του μοντέλου Generator και του μοντέλου Discriminator για να μάθει την κατανομή πιθανότητας που βασίζεται στα δεδομένα εκπαίδευσης. Ο στόχος αυτού του είδους δικτύου είναι ο κβαντικός generator να μάθει την υποκείμενη κατανομή πιθανότητας των δεδομένων εκπαίδευσης και φορτώνει μια κβαντική κατάσταση που είναι ένα μοντέλο της κατανομής στόχου.

Όπως μπορεί κανείς να συμπεράνει από την παραπάνω δήλωση, τα QGAN εί-

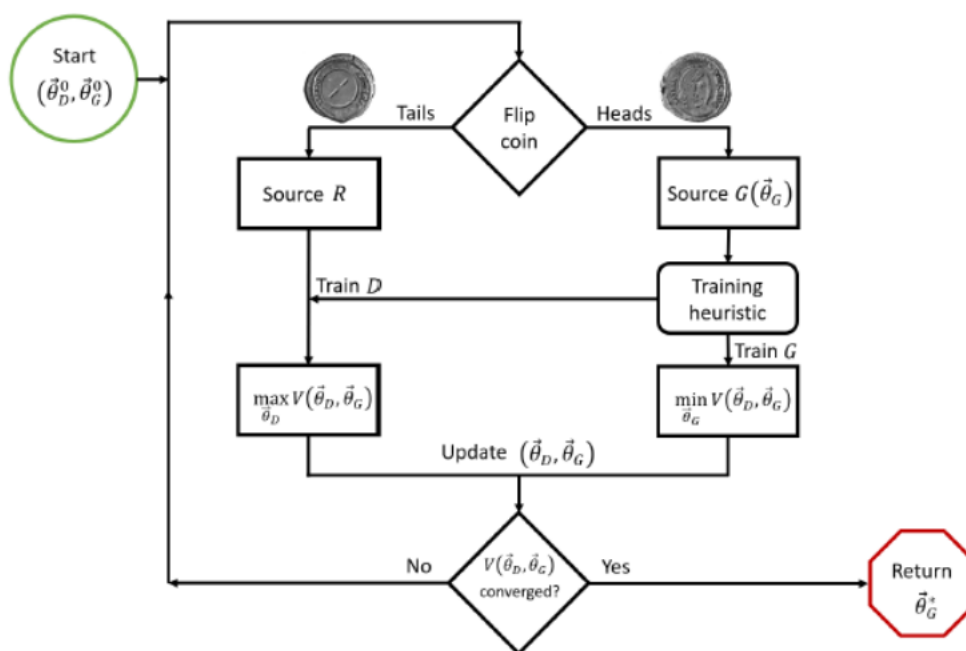
ναι πολύ χρήσιμα επειδή μπορούν να φορτώσουν σε πολυωνυμικό χρόνο τυχαίες κατανομές πιθανοτήτων σε καταστάσεις κβαντικών δεδομένων. Ο αλγόριθμος εκπαιδεύει τον κβαντικό generator για να δημιουργήσει μια κβαντική κατάσταση, η οποία αντιπροσωπεύει την υποκείμενη κατανομή πιθανότητας των δεδομένων, και επιτυγχάνει το λεγόμενο κβαντικό πλεονέκτημα σε συνδυασμό με άλλους αλγόριθμους όπως η Κβαντική Εκτίμηση Πλάτους (QAE). Από μια άλλη οπτική, αυτός ο αλγόριθμος μπορεί να χρησιμοποιηθεί για κλασικά δεδομένα για αποτελεσματική QIP (Quantum Information Processing) και έχει εφαρμογές στα χρηματοοικονομικά και τραπεζικά. Φυσικά, όλα τα παραπάνω έχουν σημασία γιατί οι Κβαντικοί αλγόριθμοι έχουν τη δυνατότητα να ξεπεράσουν τα κλασικά τους ανάλογα και είναι μια πιο αποτελεσματική μέθοδος λόγω του υπολογιστικού κόστους και του χρόνου-κόστους φόρτωσης δεδομένων σε κβαντικές καταστάσεις ( $O(2^n)$ ).

Για να υλοποιηθούν όλες οι προαναφερθείσες διεργασίες, στα QGAN και στο Quantum ML θα πρέπει κανείς να χρησιμοποιήσει Variational Quantum Algorithms. Είναι το κλειδί για την επίτευξη του επιθυμητού κβαντικού πλεονεκτήματος. Γενικά, τα VQA έχουν αναπτυχθεί για μια μεγάλη ποικιλία εφαρμογών. Τα VQA χρησιμοποιούν εργασίες κωδικοποιημένες σε μια συνάρτηση κόστους, η οποία αξιολογείται από έναν κβαντικό υπολογιστή, και χρησιμοποιούν έναν κλασικό βελτιστοποιητή για να εκπαιδεύσουν ένα παραμετροποιημένο κβαντικό κύκλωμα και με τη δομή τους έχουν καταφέρει να αναδειχθούν ως ηγετική στρατηγική για την αντιμετώπιση των περιορισμών των σημερινών κβαντικών συσκευών. Μερικοί από αυτούς τους πολύ δεσμευτικούς περιορισμούς είναι ο περιορισμένος αριθμός Qubits που χρησιμοποιούνται και οι διαδικασίες θορύβου που περιορίζουν το βάθος του κυκλώματος. Έτσι, η προσαρμοστική φύση τους βοηθά να ξεπεραστούν αυτά τα προβλήματα. Για την υλοποίηση αυτών των κβαντικών αλγορίθμων επιλέγουμε από μια ποικιλία εργαλείων που δημιουργήθηκαν από την IBM για την υλοποίηση και οπτικοποίηση κβαντικών κυκλωμάτων μέσω του IBM-Quantum Experience και, τέλος, για τη μέτρηση και την έξοδο των αποτελεσμάτων μας.

Κατά την εφαρμογή QGAN, μπορεί κανείς να επιλέξει από μια ποικιλία ζευγών generator-discriminator ανάλογα με το περιβάλλον εκτέλεσης είτε σε κβαντικούς υπολογιστές είτε σε προσομοιώσεις (μια πιο λεπτομερής ανάλυση σε αυτό εμφανίζεται στο κεφάλαιο "Πειραματική Μέθοδος"). Σε αυτά τα μοντέλα, τα πραγματικά δεδομένα πρέπει να εμπλέκονται στο στάδιο προετοιμασίας κατάστασης, συνήθως μέσω κωδικοποίησης πλάτους, για την κωδικοποίηση κλασικών δεδομένων σε κβαντική κατάσταση, και αυτό το στάδιο παίρνει  $N \log(M)$  qubits όπου  $N$  είναι το μέγεθος του συνόλου εκπαίδευσης και το  $M$  είναι διάσταση χαρακτηριστικού. Στην περίπτωση μας, εάν μελετούσαμε το σύνολο δεδομένων QM9, χρειάζονται περισσότερα από 90 qubits για να ανακαλύψουμε πραγματικά μόρια. Αυτός ο συνολικός αριθμός των απαιτούμενων qubits για την ανακατασκευή συνθετικών μορίων είναι  ${}^9_2 \log 5 + 9 \log 5 > 90$ , όπου 5 είναι ο αριθμός των τύπων δεσμών και των τύπων ατόμων που περιέχονται στο QM9. Έτσι, δεδομένης της πολυπλοκότητας των εργασιών

της κατανομής μορίων εκμάθησης, το πλήρες κβαντικό GAN δύσκολα μπορεί να κωδικοποιήσει όλα τα δεδομένα εκπαίδευσης με κβαντικό τρόπο. Αυτό το γεγονός καθιστά ένα πλήρες κβαντικό μοντέλο μη εφικτό στο εγγύς μέλλον και οι ερευνητές έχουν προτείνει υβριδικά μοντέλα για να ξεπεραστεί αυτή η δυσκολία. Αυτά τα μοντέλα περιλαμβάνουν ένα υβριδικό generator και όλες οι παραλλαγές εκμεταλλεύονται την ισχυρή εκφραστική ισχύ των μεταβλητών κβαντικών κυκλωμάτων με εκθετική επιτάχυνση έως και  $O(\text{poly}(\log M))$  χρόνο. Τέλος, στα περισσότερα μοντέλα που χρησιμοποιούνται, ο discriminator μιμείται το μοντέλο MoGAN και τα δίκτυα ανταμοιβής απορρίπτονται καθώς παρατήρησαν ότι η αξία ανταμοιβής είναι πολύ μικρή για να συμβάλει αισθητά στην εκπαίδευση του μοντέλου. Ένας μοναδικός βελτιστοποιητής χρησιμοποιείται για την ενημέρωση όλων των παραμέτρων και βαρών της κβαντικής πύλης στο νευρωνικό δίκτυο ταυτόχρονα και ο discriminator ενημερώνεται εναλλακτικά.

Η γενικευμένη διαδικασία που ακολουθείται από ένα QGAN, η οποία περιγράφεται παρακάτω, μπορεί να συνοψιστεί στο ακόλουθο διάγραμμα.



Σχήμα 6.9: QGAN flow

Γενικά, τα δύο κβαντικά κυκλώματα (generator και discriminator) μεγιστοποιούν και ελαχιστοποιούν το ίδιο πρόβλημα βελτιστοποίησης, όπως στην κλασική αναλογία. Ο generator μαθαίνει τις καλύτερες παραμέτρους για να στείλει στις κβαντικές πύλες της μετά από πολλά βήματα εκπαίδευσης και διαχωρίζει μια κβαν-

τική κατάσταση που είναι σχετικά παρόμοια με την κβαντική κατάσταση που αντιπροσωπεύει πραγματικά δεδομένα. Αρχικά, λαμβάνει κανείς τιμές προσδοκίας από ένα παράδειγμα διάκρισης. Οι τιμές προσδοκίας καθορίζονται από τα δεδομένα που παρέχονται στον discriminator και εάν χρησιμοποιήθηκαν τόσο τα πραγματικά όσο και τα δημιουργημένα δεδομένα, ελήφθησαν δύο τιμές προσδοκίας, που υποδεικνύουν την απόδοση του ατόμου που διακρίνει και στους δύο τύπους δεδομένων. Μετά από αυτή τη φάση, το μοντέλο μπορεί να χρησιμοποιήσει την παραδοσιακή μέθοδο μιας συνάρτησης κόστους για να βελτιστοποιήσει τις παραμέτρους του. Η ελαχιστοποίηση της συνάρτησης κόστους συνεπάγεται τη μεγιστοποίηση της πιθανότητας σωστής ταξινόμησης πλαστών δεδομένων. Στα κλασικά GAN, χρησιμοποιούμε δύο συναρτήσεις κόστους, μία για τον discriminator και μία για το generator, αλλά σε ένα QGAN μπορούμε να τις επισημοποιήσουμε σε ένα πρόβλημα βελτιστοποίησης αντιπάλου, που δίνεται από την έκφραση:

$$\min_{\vec{\theta}_D} \max_{\vec{\theta}_G} (Pr(D(\vec{\theta}_D, R) = |real\rangle) - Pr(D(\vec{\theta}_D, G(\vec{\theta}_G, z)) = |fake\rangle)) \quad (6.20)$$

όπου  $\theta_D$  είναι το διάνυσμα των παραμέτρων που συνδέουμε στο διακριτικό μας ansatz, το  $\theta_G$  είναι το διάνυσμα των παραμέτρων που συνδέουμε στο generator μας ansatz. Είναι σημαντικό να σημειωθεί ότι και τα δύο μέρη εξαρτώνται από το  $\theta_D$ , αλλά μόνο το δεύτερο μέρος εξαρτάται από το  $\theta_G$ . Μαθηματικά, χρειάζεται να ρυθμίσουμε το QGAN έτσι ώστε τα  $|real\rangle$  και  $|fake\rangle$  να είναι ορθογώνια για να διατηρηθεί η διακριτικότητα των δύο για τα δύο για εκπαίδευση Discriminator, με αποτέλεσμα μια καλύτερη Generator. Χρησιμοποιούμε αυθαίρετα ως γεγονός ότι  $| \rangle = |0\rangle$  και  $| \rangle = |1\rangle$  με ιδιοτιμές  $+1, -1$ . Για το διακριτικό χρησιμοποιούμε μια ενιαία λειτουργία της μορφής:

$$U_D = D(\vec{\theta}_D) \otimes I^{\otimes m} \quad (6.21)$$

με  $m$ : τον αριθμό των qubits σε έναν καταχωρητή. Οι αντίστοιχοι τελεστές για το generator και την πηγή  $R$  είναι:

$$U_G = I^{\otimes(1+d)} \otimes G(\vec{\theta}_G, z) \quad (6.22)$$

$$U_R = I^{\otimes(1+d)} \otimes R \quad (6.23)$$

Τώρα μπορούμε να ορίσουμε την κατάσταση όταν το  $U_D$  εφαρμόζεται μετά το  $U_G$ :

$$\rho^{DG}(\vec{\theta}_D, \vec{\theta}_G, z) = U_D(\vec{\theta}_D) \rho^G(\vec{\theta}_G, z) U_D^+(\vec{\theta}_D) \quad (6.24)$$

και η κβαντική κατάσταση όταν το  $U_D$  εφαρμόζεται μετά το  $U_R$ :

$$\rho^{DR}(\vec{\theta}_D) = U_D(\vec{\theta}_D) \rho^R U_D^+(\vec{\theta}_D) \quad (6.25)$$

όπου  $\rho^G(\vec{\theta}_G, z) = U_G(\vec{\theta}_G) \rho^0(z) U_G^+(\vec{\theta}_G)$ . Επομένως, η συνάρτηση κόστους γίνεται:

$$V(\vec{\theta}_G, \vec{\theta}_D) = \frac{1}{2} (\cos^2(\phi) Tr(Z \rho^{DR}(\vec{\theta}_D)) - \sin^2(\phi) Tr(Z \rho^{DG}(\vec{\theta}_D, \vec{\theta}_G, z))) \quad (6.26)$$

όπου  $\phi$  είναι η μεροληψία της πηγής και  $Z$  είναι παρατηρήσιμο. Σημειώστε ότι η προσδοκώμενη τιμή ενός πίνακα πυκνότητας σε σχέση με ένα παρατηρήσιμο  $Z$  είναι το ίχνος αυτού του τελεστή  $Z$  που εφαρμόζεται στον πίνακα πυκνότητας. Τέλος, ο κανόνας ενημέρωσης που χρησιμοποιείται για τις παραμέτρους  $D(\theta_D^k)$  ή  $G(\theta_G^k)$ , με ποσοστά εκμάθησης  $X_D^k$  και  $X_G^k$ :

$$\bar{\theta}_D^{k+1} = \bar{\theta}_D^k + x_D^k \nabla_{\bar{\theta}_D} V(\bar{\theta}_D^k, \bar{\theta}_G^k) \quad (6.27)$$

$$\bar{\theta}_G^{k+1} = \bar{\theta}_G^k - x_G^k \nabla_{\bar{\theta}_G} V(\bar{\theta}_D^k, \bar{\theta}_G^k) \quad (6.28)$$

όπου  $k$  είναι το βήμα.

Αν θεωρήσουμε ότι χρησιμοποιούμε διασταυρούμενη εντροπία ως συνάρτηση απώλειας μεταξύ των πραγματικών και των παραγόμενων δεδομένων  $S(\rho^R || \rho^G) = Tr(\rho^R(\log_2 \rho^R - \log_2 \rho^G))$  και συγκλίνει στο 0.

## Κεφάλαιο 7

# ML Μοντέλα Κλασικών Υπολογιστών - Πειραματική Υλοποίηση

Σε αυτό το κεφάλαιο, θα παρουσιάσουμε και θα σχολιάσουμε την πειραματική διαδικασία που επικεντρώνεται στα κλασικά μοντέλα Machine Learning. Προσπάθησα να προσεγγίσω ένα πρόβλημα ταξινόμησης. Η αρχική ιδέα ήταν να αναζητηθούν ομοιότητες μορίων που χρησιμοποιούνται σε ραδιοφάρμακα σε σύγκριση με τα μόρια που παρέχονται από το σύνολο δεδομένων QM9. Ωστόσο, ο έλεγχος ομοιότητας που εφάρμοσα μέσω της βιβλιοθήκης rdkit, έδειξε αρκετά χαμηλές βαθμολογίες και αποφάσισα να προχωρήσω από αυτό το πρόβλημα. Για αυτόν τον έλεγχο ομοιότητας χρησιμοποίησα ραδιενεργά ισότοπα συμπεριλαμβανομένων των  $^{18}F$ ,  $^{11}C$  και  $^3H$ . Η μέτρηση που χρησιμοποιήθηκε ήταν η ομοιότητα Tanimoto και τα αποτελέσματα που λήφθηκαν ήταν παραγγελίας  $10^{-2}$  με μέγιστο όριο στο 0,3-0,4. Αυτή η μέτρηση θεωρείται ως κατάλληλη επιλογή για έλεγχο ομοιότητας βάσει δακτυλικών αποτυπωμάτων. Σε αυτόν τον αλγόριθμο, εξετάζουμε δύο σετ A και B δυαδικών ψηφίων δακτυλικών αποτυπωμάτων. Το AB είναι το σύνολο των κοινών δυαδικών ψηφίων των δακτυλικών αποτυπωμάτων τόσο του μορίου A όσο και του μορίου B. Ο συντελεστής Tanimoto δίνεται από την παρακάτω εξίσωση:

$$T(A, B) = \frac{A^B}{A + B - A^B} \quad (7.1)$$

και κυμαίνεται από 0 όταν τα δακτυλικά αποτυπώματα δεν έχουν κοινά κομμάτια, έως 1 όταν τα δακτυλικά αποτυπώματα είναι πανομοιότυπα. Μια καλή τιμή που μπορεί να χρησιμοποιηθεί για να θεωρηθούν δύο μόρια ως "παρόμοια" είναι πάνω από 0,85.



Ένα πρόβλημα που τράβηξε την προσοχή μου είναι ο αυξανόμενος αριθμός νέων με ψυχικά προβλήματα. Έτσι, αποφάσισα να επικεντρώσω τη μελέτη μου για την ανακάλυψη φαρμάκων σε μόρια που θα μπορούσαν να χρησιμοποιηθούν ως τρικυκλικά αντικαταθλιπτικά. Αυτά τα φάρμακα χρησιμοποιούνται για να ανακουφίσουν την κατάθλιψη επηρεάζοντας τους νευροδιαβιβαστές που χρησιμοποιούνται για την επικοινωνία μεταξύ των εγκεφαλικών κυττάρων. Όπως τα περισσότερα αντικαταθλιπτικά, τα κυκλικά αντικαταθλιπτικά δρουν επηρεάζοντας τελικά αλλαγές στη χημεία του εγκεφάλου και στην επικοινωνία στα κυκλώματα των νευρικών κυττάρων του εγκεφάλου που είναι γνωστό ότι ρυθμίζουν τη διάθεση, βοηθώντας στην ανακούφιση της κατάθλιψης. Τα κυκλικά αντικαταθλιπτικά εμποδίζουν την επαναρρόφιση των νευροδιαβιβαστών σεροτονίνης και νορεπινεφρίνης, αυξάνοντας τα επίπεδα αυτών των δύο νευροδιαβιβαστών στον εγκέφαλο. Ο Οργανισμός Τροφίμων και Φαρμάκων (FDA) έχει εγκρίνει ορισμένα αντικαταθλιπτικά που ονομάζονται δοξεπίνη, προτριπτυλίνη, αμοξαπίνη και αμιτριπτυλίνη. Επίσης, μερικές φορές κυκλικά αντικαταθλιπτικά χρησιμοποιούνται για τη θεραπεία καταστάσεων εκτός της κατάθλιψης, όπως η ιδεοψυχαναγκαστική διαταραχή, η αγχώδης διαταραχή ή ο πόνος που σχετίζεται με τα νεύρα.

Έτσι, ο σκοπός αυτής της μελέτης είναι να μελετήσει αυτό το πρόβλημα ταξινόμησης των μορίων που χρησιμοποιούνται για το σχεδιασμό αντικαταθλιπτικών φαρμάκων. Για να το επιτύχουμε αυτό, χρησιμοποιήσαμε στο οπλοστάσιό μας διαφορετικούς τύπους Νευρωνικών Δικτύων, συμπεριλαμβανομένων των συνελκτικών, γραφικών και παραγωγικών αντιπάλων δικτύων. Αυτή η στρατηγική έχει ως στόχο να δώσει ως σφαιρικά συμπεράσματα σχετικά με το σύνολο δεδομένων QM9 και πώς μπορεί να χρησιμοποιηθεί αξιόπιστα για το σχεδιασμό αντικαταθλιπτικών φαρμάκων. Ωστόσο, πριν εφαρμόσουμε όλα τα αναφερόμενα μοντέλα μηχανικής εκμάθησης, πρέπει να προετοιμάσουμε το σύνολο δεδομένων μας για να είμαστε έτοιμοι να το χρησιμοποιήσουμε ως είσοδο στα μοντέλα μας.

Αρχικά, φόρτωσα αυτό το σύνολο δεδομένων και όλα τα προαπαιτούμενα πακέτα στο περιβάλλον μου. Το περιβάλλον που χρησιμοποιήθηκε ήταν τα Jupyter Notebook της Google Colab που χρειάζονταν μια λύση cloud. Το σύνολο δεδομένων QM9 φορτώθηκε αυτόματα χρησιμοποιώντας τη βιβλιοθήκη chainer chemistry και η αρχική του μορφή είναι συμπιεσμένο αρχείο όπου οι πληροφορίες κάθε μορίου αποθηκεύονται σε κάθε αρχείο "xyz". Το Chainer Chemistry συγχωνεύει αυτόματα αυτές τις πληροφορίες σε ένα αρχείο csv εσωτερικά. Στη συνέχεια, για την εξαγωγή δεδομένων QM9 χρησιμοποιήθηκε ο GGNNPreprocessor. Εγκαθιστούμε τον προεπεξεργαστή και ακολουθούμε μεθόδους "get-qm9" για να εξαγάγουμε όλες τις ετικέτες. Μετά από αυτήν την ενέργεια, εξήχθησαν 15 τύποι φυσικών ιδιοτήτων και ο βασικός τύπος κωδικοποίησης των πληροφοριών που απαιτούνται είναι μέσω συμβολοσειρών SMILES μαζί με το ίδιο το σύνολο δεδομένων. Η κλάση GGNNPreprocessor λαμβάνει ως ορίσματα:

1. Ο μέγιστος αριθμός ατόμων για κάθε μόριο και εάν ο αριθμός των ατόμων είναι μεγαλύτερος από αυτήν την τιμή, αυτά τα δεδομένα απλώς αγνοούνται.
2. Το μέγεθος εξόδου, το οποίο καθορίζει το μέγεθος του πίνακα που επιστρέφεται με τη μέθοδο "get-input-features". Εάν ο αριθμός των ατόμων στο μόριο είναι μικρότερος από αυτήν την τιμή, οι επιστρεφόμενες συστοιχίες συμπληρώνονται για να έχουν σταθερό μέγεθος.

Μετά την ολοκλήρωση του παραπάνω βήματος παρατηρούμε ότι το σύνολο δεδομένων QM9 είναι μια κλάση του NumpyTupleDataset, όπου τα χαρακτηριστικά του i-ου συνόλου δεδομένων είναι προσβάσιμα από το "dataset[i]". Επιπλέον, λόγω της χρήσης της κλάσης GGNPreprocessor, κάθε σύνολο δεδομένων αποτελείται από τα ακόλουθα χαρακτηριστικά:

1. χαρακτηριστικό ατόμου: ατομικός αριθμός ενός συγκεκριμένου μορίου,
2. Χαρακτηριστικό μήτρας γεινίασης: Ο προεπεξεργαστής GGNN εξάγει τη μήτρα γεινίασης κάθε τύπου σύνδεσης, Χαρακτηριστικό ετικέτας
3. : χημικές ιδιότητες (ετικέτες) συγκεκριμένου μορίου.

Οι πίνακες γεινίασης κάθε τύπου δεσμού μεταξύ ατόμων σε ένα συγκεκριμένο μόριο:

Μονός δεσμός:

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Διπλός δεσμός:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

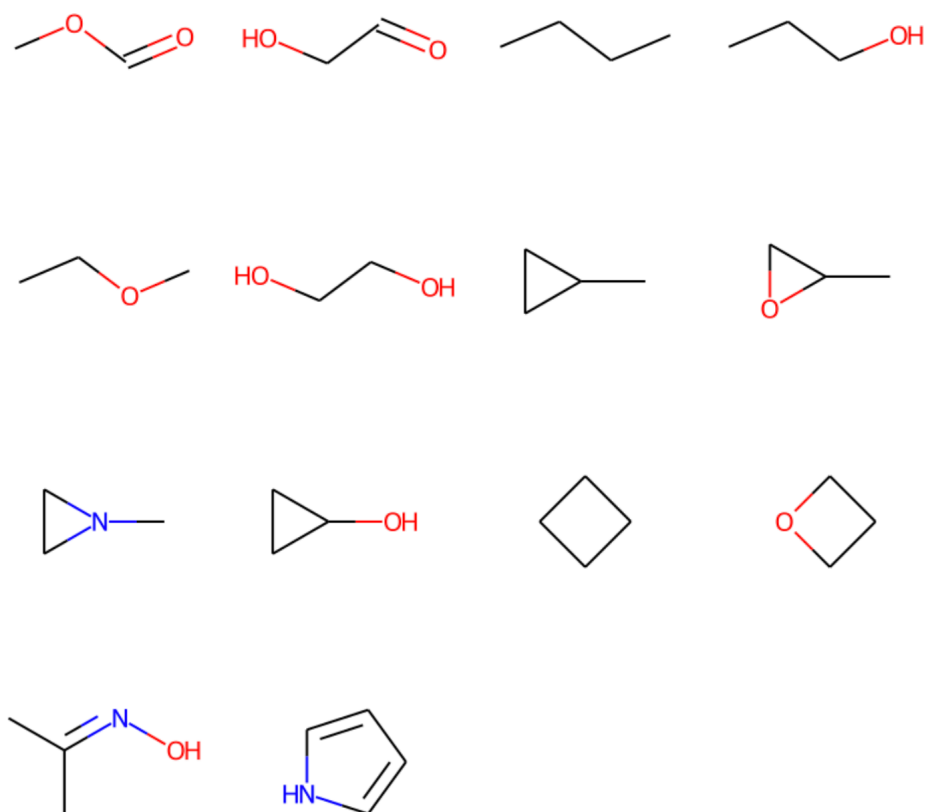
Τριπλός Δεσμός:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Αρωματικός δεσμός:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Στη συνέχεια, χρησιμοποιώ ορισμένες μονάδες οπτικοποίησης του πακέτου RDKit για να εμφανίσω ορισμένες σειρές του συνόλου δεδομένων μας σε εικόνες, προκειμένου να το κατανοήσω καλύτερα [7.1].



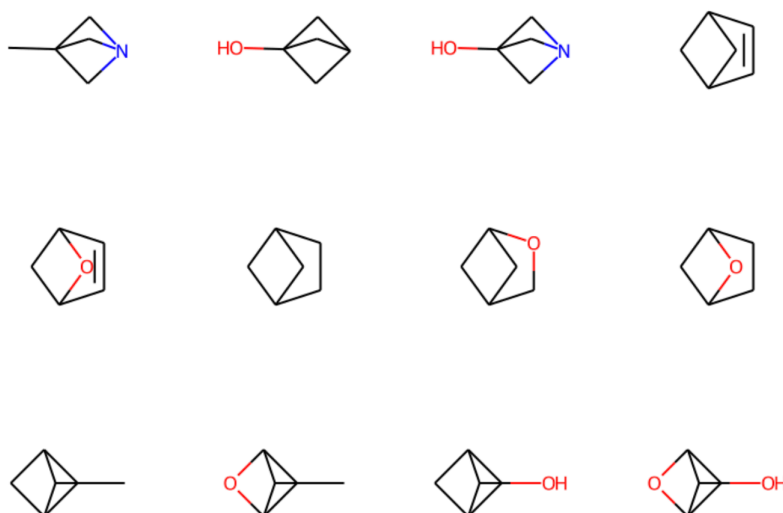
Σχήμα 7.1: Visualization of molecules in a grid

Για να συνεχίσω, δημιούργησα μια διαδραστική απεικόνιση ολόκληρου του συνόλου δεδομένων για τα Jupyter notebooks, προκειμένου να έχω πρόσβαση σε κάθε σημείο στο σύνολο δεδομένων των 13 χιλιάδων σειρών. Τέλος, είναι καιρός να περιορίσουμε το σύνολο δεδομένων στις ανάγκες μας. Αρχικά, επικοινωνήσα με έναν έλεγχο ομοιότητας στα μόρια της Σεροτονίνης, της Ντοπαμίνης και της Δοξεπίνης για να βρω κοινά χαρακτηριστικά που θα χρησιμοποιήσω ως κριτήρια για τον περιορισμό του συνόλου δεδομένων. Όπως είναι προφανές από το όνομα των φαρμάκων, φιλτράρω τη βάση δεδομένων για να διατηρήσω ως μόρια μόνο μόρια που έχουν τρεις υποδομές δακτυλίου σε ολόκληρη τη δομή τους. Με αυτό το κριτήριο περιόρισα τα δεδομένα μου σε 19672 σειρές. Αποθήκευσα όλους τους δείκτες μορίων για να μπορέσω να τα απομονώσω από ολόκληρο το σύνολο δεδομένων. Τώρα που έχω τις σειρές που θέλω να διατηρήσω, το δεύτερο κριτήριο που χρησιμοποίησα για να ταξινομήσω το σύνολο δεδομένων μου ήταν ο συντελεστής  $\log P$ . Αυτός είναι ο συντελεστής κατανομής οκτανόλης-νερού  $\log P$  και χρησιμοποιείται σε μελέτες QSAR και ορθολογικό σχεδιασμό φαρμάκων ως μέτρο της μοριακής υδροφοβικότητας. Η υδροφοβικότητα επηρεάζει την απορρόφηση του φαρμάκου, τη βιοδιαθεσιμότητα, τις υδρόφοβες αλληλεπιδράσεις φαρμάκου-υποδοχέα, το με-

	SMILES	logP
Doxepin	<chem>CN(C)CCC=C1C2=CC=CC=C2COC3=CC=CC=C31</chem>	4.29
Serotonin	<chem>C1=CC2=C(C=C1O)C(=CN2)CCN</chem>	0.21
Dopamine	<chem>C1=CC(=C(C=C1CCN)O)O</chem>	-0.98
Amoxapine	<chem>C1CN(CCN1)C2=NC3=CC=CC=C3OC4=C2C=C(C=C4)Cl</chem>	3.4

ταβολισμό των μορίων, καθώς και την τοξικότητά τους. Άλλη χρήση αυτού του συντελεστή είναι η βασική χρήση του σε μελέτες για την περιβαλλοντική τύχη των χημικών ουσιών. Η τιμή των γνωστών μορίων που χρησιμοποιούνται ως ansatz στην παρούσα μελέτη φαίνεται στον παρακάτω πίνακα [7]. Ωστόσο, καθώς η μελέτη της χημείας και των βιολογικών ιδιοτήτων όλων αυτών των φαρμάκων δεν είναι ο τομέας εμπειρογνωμοσύνης μου, για να είμαστε πιο ακριβείς θα χρειαζόμασταν τη βοήθεια ενός έμπειρου ερευνητή. Όλα τα χαρακτηριστικά και τα χαρακτηριστικά που χρησιμοποιούνται για την ταξινόμηση των δεδομένων βασίζονται σε προσωπικές παρατηρήσεις, καθώς θέλουμε απλώς να δοκιμάσουμε τις επιδόσεις των μοντέλων μας σε αυτό το σύνολο δεδομένων και επομένως δεν μπορούν να εφαρμοστούν σε πραγματικές περιπτώσεις έως ότου παρέχονται επίσημα χημικά κριτήρια για αυτά τα χαρακτηριστικά από έναν επαγγελματία συγκεκριμένη περιοχή.

Το τελικό μας σύνολο δεδομένων περιλαμβάνει μόρια της παρακάτω μορφής:



Σχήμα 7.2: Molecule Visualization for final dataset

Ανάλογα με τη σκοπιμότητα των μοντέλων μας χρησιμοποιούμε είτε μια δυαδική ταξινόμηση, είτε μια ταξινόμηση πολλαπλών κλάσεων χωρίζοντας τα δεδομένα μας σε 4 κατηγορίες με βάση τις τιμές του συντελεστή logP. Στις περισσότερες περιπτώσεις στα επόμενα κεφάλαια, χρησιμοποιούμε δεδομένα με τη μορφή εικόνων/πίνακες σε συνδυασμό με τη δυνατότητα logP για το πρόβλημα ταξινόμησης και σε λίγες περιπτώσεις εκμεταλλευόμαστε άλλα δομικά χαρακτηριστικά του μοντέλου για να δείξουμε πώς μπορούμε να βελτιστοποιήσουμε την κωδικοποίησή τους σε ταξινομητές. Τέλος, στοχεύουμε να καταλήξουμε σε ποιο επίπεδο μπορούμε να εκμεταλλευτούμε την κβαντική υπεροχή αυτή τη στιγμή, δεδομένου του γεγονότος ότι οι κβαντικοί υπολογιστές βρίσκονται σε προκαταρκτικό στάδιο και βελτιώνονται συνεχώς.

## 1. Convolutional Neural Network (CNN)

Σε αυτή τη συγκεκριμένη μελέτη περίπτωσης, αποφάσισα να επικεντρωθώ σε μια ανάλυση ταξινόμησης πολλαπλών τάξεων του συνόλου δεδομένων. Αποφάσισα να χρησιμοποιήσω ως είσοδο τις μήτρες που έχουμε αποκτήσει από τις εικόνες των μορίων. Ο σκοπός της πρακτικής χρήσης εικόνων σε αυτό το είδος δικτύου είναι επειδή τα CNN είναι απρόσβλητα στη χωρική διακύμανση και ως εκ τούτου είναι σε θέση να ανιχνεύουν χαρακτηριστικά οπουδήποτε στις εικόνες εισόδου. Επίσης τα CNN είναι φτιαγμένα με τρόπο που μπορούν να παράγουν εξαιρετικές προβλέψεις με ελάχιστη προεπεξεργασία εικόνας. Για αυτήν την περίπτωση χώρισα τα δεδομένα μου σε 4 τάξεις με βάση τον συντελεστή logP και χώρισα τα δεδομένα μου σε 80% σύνολο δεδομένων εκπαίδευσης και 20% σύνολο δεδομένων δοκιμής.

Ξεκίνησα επιλέγοντας ένα απλό μοντέλο MLP με μερικά επίπεδα, το οποίο εμφανίζεται παρακάτω [7.3].

Model: "model"

Layer (type)	Output Shape	Param #
input_2 (InputLayer)	[(None, 64, 64, 4)]	0
dense_9 (Dense)	(None, 64, 64, 5)	25

=====  
Total params: 25  
Trainable params: 25  
Non-trainable params: 0  
=====

Σχήμα 7.3: One Layer Initial Model

Τα αποτελέσματα ήταν αρκετά απογοητευτικά όπως μπορεί κανείς να παρατηρήσει [1].

loss	accuracy	precision	val_loss	val_accuracy	val_precision
2.6527	0.5113	0.0306	25.0061	0.0490	0.0032
1.5340	0.6363	0.0704	29.3843	0.0490	0.0031
1.2716	0.6908	0.0924	27.2253	0.0490	0.0032
1.1985	0.6908	0.0988	28.5666	0.0490	0.0033

Από την αρχή παρατηρήθηκε overfitting σε αντίθεση με τη φυσιολογική συμπεριφορά της απώλειας και της ακρίβειας της προπόνησης, η απώλεια στα δεδομένα επικύρωσης συνέχιζε να αυξάνεται και η εκπαίδευση σταμάτησε μόνο μετά από 4 epochs, επειδή παρέμεινε σταθερή και διακόπηκε από την Πρόωρη Διακοπή. Επίσης, τόσο η ακρίβεια εκπαίδευσης όσο και η ακρίβεια επικύρωσης διατηρήθηκαν κάτω από 1%. Έτσι, έκανα μερικές επιλογές για να αλλάξω την αρχιτεκτονική του μοντέλου και μπορεί κανείς να βρει την εξήγηση στο τελικό μοντέλο.

Η τελική επιλογή για την αρχιτεκτονική που χρησιμοποιείται για αυτό το μοντέλο φαίνεται στο [7.4]. Μπορεί κανείς να δει ότι έχουμε χρησιμοποιήσει ένα Διαδοχικό μοντέλο με τα ορίσματά του που περιλαμβάνουν έναν βελτιστοποιητή Adam, 4 νευρώνες και μεταβλητή εγκατάλειψη. Τα κρυφά επίπεδα του μοντέλου περιλαμβάνουν 3 δισδιάστατα συνελκτικό φίλτρο μεγέθους πυρήνα 3 3 για την τυποποίηση των εισόδων σε κάθε επίπεδο για κάθε μίνι παρτίδα, σε συνδυασμό με μια λειτουργία ενεργοποίησης ReLU. Καθώς ο χρόνος είναι πάντα περιορισμός, απο-

φασίσαμε να εφαρμόσουμε πολλά βήματα ομαλοποίησης παρτίδων προκειμένου να μειώσουμε τον αριθμό των εποχών που απαιτούνται για την εκπαίδευση του δικτύου και να ξεπεράσουμε το φαινόμενο της υπερβολικής προσαρμογής που εμφανίστηκε σε προηγούμενα μοντέλα που δοκίμασα.

```

Model: "sequential_2"
-----
Layer (type)                Output Shape                Param #
-----
conv2d_6 (Conv2D)           (None, None, None, 100)    2800

batch_normalization_6 (Batc (None, None, None, 100)    400
hNormalization)

max_pooling2d_6 (MaxPooling (None, None, None, 100)    0
2D)

dropout_8 (Dropout)         (None, None, None, 100)    0

conv2d_7 (Conv2D)           (None, None, None, 80)     72080

batch_normalization_7 (Batc (None, None, None, 80)     320
hNormalization)

max_pooling2d_7 (MaxPooling (None, None, None, 80)     0
2D)

dropout_9 (Dropout)         (None, None, None, 80)     0

conv2d_8 (Conv2D)           (None, None, None, 32)     23072

batch_normalization_8 (Batc (None, None, None, 32)     128
hNormalization)

max_pooling2d_8 (MaxPooling (None, None, None, 32)     0
2D)

dropout_10 (Dropout)        (None, None, None, 32)     0

flatten_2 (Flatten)         (None, None)                0

dense_4 (Dense)             (None, 8)                   73992

dropout_11 (Dropout)        (None, 8)                   0

dense_5 (Dense)             (None, 4)                   36

=====
Total params: 172,828
Trainable params: 172,404
Non-trainable params: 424
-----

```

Σχήμα 7.4: CNN architecture for classification problem

Επίσης, όπως αναφέρθηκε για το επίπεδο ομαλοποίησης παρτίδας, λόγω των προβλημάτων υπερπροσαρμογής που εμφανίστηκαν στις πρώτες προσπάθειες του μοντέλου που αναλύθηκε σε αυτήν την ενότητα, αποφάσισα να προσθέσω ένα δι-διάστατο στρώμα Max Pooling μετά από κάθε φίλτρο για να το παρέχω στο επόμενο στρώμα μόνο μια αφηρημένη μορφή της αναπαράστασης. Χρησιμοποιώντας αυτό το επίπεδο, κατάφερα να μειώσω την υπερβολική προσαρμογή καθώς μείωσα τη δειγματοληψία των επόμενων επιπέδων και παρείχα μια βασική αναλλοίωτη μετάφραση στην εσωτερική αναπαράσταση. Αφού εφάρμοσα αυτή την τριάδα στρω-



μάτων πολλές φορές, χρησιμοποίησα μια εγκατάλειψη μετά από κάθε τριάδα για να αγνοήσω τυχαία ορισμένους νευρώνες προσωρινά. Τέλος, για να διασφαλίσω ότι η εισαγωγή μου θα εισαχθεί ως μονοδιάστατη στα τελικά πυκνά στρώματα σε συνδυασμό με επιπλέον στρώματα εγκατάλειψης. Πριν βγω από το μοντέλο, χρησιμοποίησα ένα τελικό πυκνό στρώμα με συνάρτηση ενεργοποίησης softmax, καθώς χρησιμοποιείται σε πολυωνμικές περιπτώσεις για την κανονικοποίηση της εξόδου του μοντέλου.

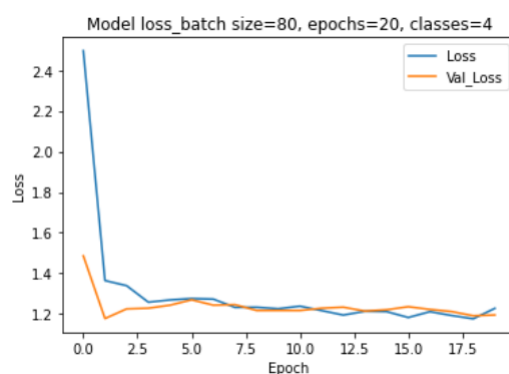
Οι μετρήσεις που χρησιμοποιήθηκαν στο παρόν μοντέλο ήταν η κατηγορική συνάρτηση διασταυρούμενης εντροπίας ως συνάρτηση απώλειας, ο βελτιστοποιητής Adam και η ακρίβεια και η ακρίβεια. Σε κάθε εκτέλεση, παρακολουθούσα την ακρίβεια επικύρωσης και με τη χρήση της επιστροφής κλήσης Early Stop σταματούσα την προπόνηση αφού πέτυχα τη μέγιστη τιμή της κατά τη διάρκεια 5 εποχών, καθώς θεώρησα ότι το μοντέλο δεν θα βελτιωνόταν περαιτέρω. Τα καλύτερα αποτελέσματα που ανακτήθηκαν από αυτό το μοντέλο με ρυθμό εκμάθησης  $lr = 0,0001$  φαίνονται στον παρακάτω πίνακα [7.5]:

	loss	accuracy	precision_3	val_loss	val_accuracy	val_precision_3
0	2.496617	0.456500	0.439192	1.486472	0.200000	0.211982
1	1.364678	0.665000	0.646102	1.177607	0.710417	0.699797
2	1.339337	0.683000	0.681728	1.225186	0.718750	0.697395
3	1.258200	0.714932	0.701617	1.228504	0.691667	0.688391
4	1.269119	0.705500	0.703335	1.243383	0.679167	0.677618
5	1.276160	0.702500	0.701292	1.269683	0.685417	0.683128
6	1.273295	0.703500	0.703039	1.243559	0.731250	0.726337
7	1.232521	0.718500	0.712099	1.245967	0.691667	0.695122
8	1.233191	0.728500	0.725646	1.217297	0.704167	0.704167
9	1.225855	0.740500	0.733760	1.217422	0.729167	0.721992
10	1.238567	0.742500	0.740907	1.217267	0.712500	0.701461
11	1.216776	0.725500	0.723383	1.228390	0.704167	0.707113
12	1.194726	0.733000	0.732535	1.233482	0.735417	0.730612
13	1.213123	0.727000	0.721920	1.214403	0.708333	0.713389
14	1.211676	0.738000	0.738048	1.220848	0.654167	0.652807
15	1.182723	0.727500	0.725012	1.235777	0.687500	0.687243
16	1.211328	0.719500	0.718703	1.222572	0.733333	0.734864
17	1.192584	0.732500	0.732500	1.211921	0.722917	0.721649
18	1.176320	0.747500	0.693756	1.191294	0.718750	0.715164
19	1.227817	0.716500	0.620489	1.195041	0.712500	0.723493

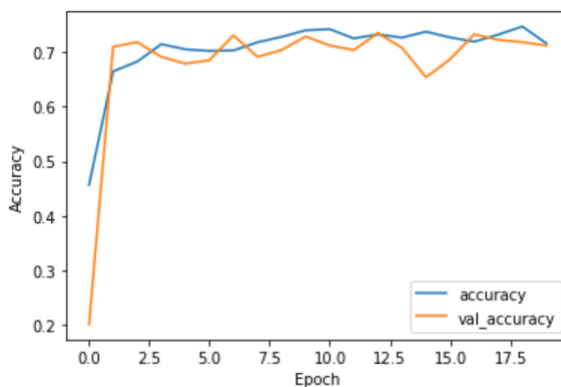
Σχήμα 7.5: CNN statistics with lr=0.0001

Τα αποτελέσματα λήφθηκαν με παραμέτρους που ορίστηκαν ως μέγεθος παρτίδας = 80, epochs = 20, νευρώνες = 4. Το μοντέλο εκτελέστηκε σε GPU για 113,33 λεπτά με μέσο χρόνο του epoch ίσο με 7,25 λεπτά.

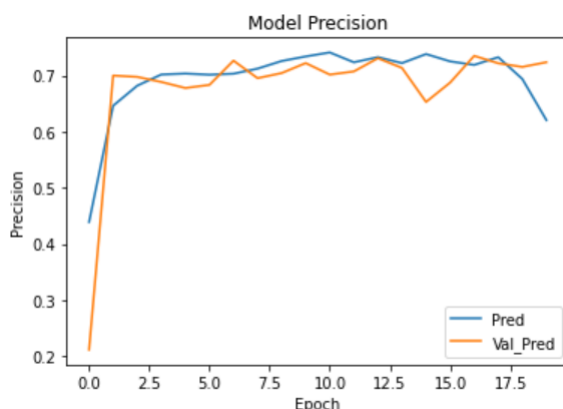
Όπως μπορεί κανείς να παρατηρήσει, με αυτό το μοντέλο καταφέραμε να φτάσουμε στην υψηλότερη ακρίβεια εκπαίδευσης 74,25%, υψηλότερη ακρίβεια 74,091% στη σειρά 10, αλλά την υψηλότερη ακρίβεια επικύρωσης στη σειρά 12, η οποία είναι ίση με 73,54% και η ακρίβεια επικύρωσης ίση σε 73,06%. Τέλος, τόσο η συνάρτηση απώλειας εκπαίδευσης όσο και η συνάρτηση απώλειας επικύρωσης ακολουθούν την φθίνουσα συμπεριφορά  $x^2$  που υποτίθεται ότι πρέπει να κάνουν. Μπορείτε να δείτε την απεικόνιση αυτών των αριθμών παρακάτω:



Σχήμα 7.6: Model Loss (batch size=80, lr=0.0001, epochs=20)



Σχήμα 7.7: Model Accuracy (batch size=80, lr=0.0001, epochs=20)



Σχήμα 7.8: Model Precision (batch size=80, lr=0.0001, epochs=20)

Για να βεβαιωθώ ότι αυτά τα αποτελέσματα ήταν τα καλύτερα που μπορούσαμε να επιτύχουμε με τη συγκεκριμένη αρχιτεκτονική μοντέλου, έκανα μια ανάλυση παραμέτρων για να βρω τις βελτιστοποιημένες παραμέτρους.

	Avg Acc.	Avg Val.Acc.	Avg Prec.	Avg Val.Prec.
batch s=40, lr=0.01	0.720247	0.72542	0.703839	0.635477
batch s=80, lr=0.01	0.705395	0.68157	0.694795	0.679685
batch s=100, lr=0.01	0.726905	0.7267	0.71905	0.6789
batch s=100, lr=0.001	0.710743	0.69432	0.700125	0.693432
batch s=100, lr=0.0001	0.706395	0.68157	0.694745	0.679685

Έτσι, παρατηρούμε από τα παραπάνω αποτελέσματα ότι οι βέλτιστες παράμετροι είναι μέγεθος παρτίδας=100 και ρυθμός εκμάθησης=0,01 και παρατηρούμε ότι η απόδοση του μοντέλου είναι χειρότερη όσο αυξάνουμε το μέγεθος της παρτίδας. Αυτό το συμπέρασμα προκύπτει από την ακρίβεια επικύρωσης και την ακρίβεια επικύρωσης, που είναι πιο κοντά στην τιμή εκπαίδευσης, όταν αυξάνουμε το μέγεθος της παρτίδας. Ειδικά, με τόσο μεγάλη διαφορά μεταξύ των μετρήσεων εκπαίδευσης και των μετρήσεων επικύρωσης, θα μπορούσαμε να υποθέσουμε ότι υπάρχει κάποιο είδος υποπροσαρμογής που συμβαίνει. Όσον αφορά την εξάρτηση της απόδοσης του μοντέλου από την παράμετρο ρυθμού εκμάθησης, παρατηρούμε ότι η βέλτιστη τιμή είναι lr=0,01 καθώς επιτυγχάνει την υψηλότερη απόδοση και δεν προκαλεί υπερπροσαρμογή. Αν αυξούσαμε ακόμη περισσότερο το ποσοστό μάθησης, θα μπορούσε κανείς να παρατηρήσει έντονο overfitting.

## 2. Graph Neural Network (GNN)

Για τους σκοπούς αυτού του κεφαλαίου στοχεύουμε να χρησιμοποιήσουμε τα νευρωνικά δίκτυα γραφικών για να κάνουμε προβλέψεις στο σύνολο δεδομένων μας, που περιέχουν πληροφορίες για μόρια από τη βάση δεδομένων QM9 και τέλος να διευκρινίσουμε εάν ορισμένα από αυτά μπορούν να χρησιμοποιηθούν ως συστατικά αντικαταθλιπτικών φαρμάκων. Για την κατασκευή αυτού του μοντέλου, χρησιμοποίησα το TensorFlow και το PyTorch. Στη συγκεκριμένη περίπτωση, αποφάσισα να χρησιμοποιήσω ένα Graph Convolutional Network (GCN). Ωστόσο, η πρώτη εργασία που πρέπει να γίνει είναι να προετοιμάσουμε τα δεδομένα εισόδου μας. Τα δεδομένα γραφήματος θα δημιουργηθούν από τη στήλη του συνόλου δεδομένων που περιέχει τις συμβολοσειρές SMILES. Για να μετατρέψω όλα τα SMILES σε είσοδο γραφήματος, δημιουργώ μια συνάρτηση "smiles2graph", η οποία λαμβάνει μια συμβολοσειρά SMILES ως το μόνο όρισμα. Για να διαχειριστώ αυτά τα διανύσματα εισόδου, χρησιμοποιώ επίσης τα γραφήματα που περιέχονται ήδη στο σύνολο δεδομένων QM9, τη μήτρα γειτνίασης για κάθε αντίστοιχο δεσμό που ζευγαρώνει άτομα σε κάθε μόριο.

Για να συνεχίσω, ορίζω επίσης μια προσαρμοσμένη συνάρτηση που λαμβάνει συμβολοσειρές SMILES ως το μόνο όρισμα εισόδου και εξάγει τα χαρακτηριστικά του ατόμου για ένα μεμονωμένο μόριο με τη μορφή διανύσματος. Το σύνολο δεδομένων που χρησιμοποιήθηκε είναι το ίδιο που χρησιμοποιήθηκε στο προηγούμενο κεφάλαιο για το μοντέλο του CNN. Τώρα που έχω τα γραφήματα με τα χαρακτηριστικά, ήρθε η ώρα να ορίσω τις ετικέτες. Ορίζω τις 4 διαφορετικές κλάσεις όπως έγινε πριν, χρησιμοποιώντας τον συντελεστή logP ως κριτήριο:

- ▶ Class 1:  $\log P < -2.84025$ ,
- ▶ Class 2:  $\log P \in [-2.84025, -0.7715)$ ,
- ▶ Class 3:  $\log P \in [-0.7715, 1.29725)$  and Class 4:  $\log P \geq 1.29725$

Στη συνέχεια χρησιμοποιώ τη μέθοδο One-Hot Encoding για να ενσωματώσω τα δεδομένα μου. Στη συνέχεια κατασκευάζω ένα σύνολο δεδομένων γραφήματος DGL. Εκτυπώνοντας το σύνολο δεδομένων μου, παρατηρώ τη μορφή στην οποία έχουν σχήμα τα δεδομένα. Κάθε σειρά δεδομένων περιέχει τον αριθμό των κόμβων, τον αριθμό των ακμών και το αντίστοιχο διάνυσμα χαρακτηριστικών με το σχήμα του. Στη συνέχεια, ορίζω μια κλάση που λαμβάνει ως όρισμα το σύνολο δεδομένων DGL και δημιουργεί το τελικό συνθετικό σύνολο δεδομένων που θα χρησιμοποιήσω ως είσοδο για το Graph Convolutional Network μου. Σε αυτό το σημείο, χρησιμοποιώ έναν φορτωτή δεδομένων Graph και έναν τυχαίο δειγματολήπτη και ορίζω επίσης

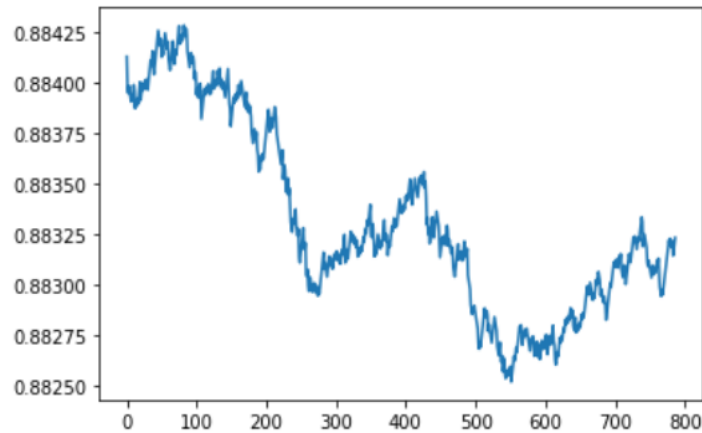
	nodes	edges
<b>count</b>	19672.0	19672.0
<b>mean</b>	8.9	21.2
<b>std</b>	0.4	1.2
<b>min</b>	5.0	12.0
<b>25%</b>	9.0	20.0
<b>50%</b>	9.0	22.0
<b>75%</b>	9.0	22.0
<b>max</b>	9.0	22.0

Σχήμα 7.9: Graph dataset statistics

μια κλάση για το μοντέλο GCN που περιέχει τα περάσματα προς τα εμπρός και προς τα πίσω.

Ως τελευταίο βήμα του σταδίου προεπεξεργασίας δεδομένων, εξήγα τα παρακάτω στατιστικά στοιχεία σχετικά με τους κόμβους και τις άκρες των γραφημάτων που δημιουργούνται στο σύνολο δεδομένων. Τώρα που έχω συγκεντρώσει όλα τα προαπαιτούμενα, ορίζω το μοντέλο μου με τη βοήθεια της κλάσης GCN και χρησιμοποιώ το μήκος του διανύσματος χαρακτηριστικών και τον αριθμό των κλάσεων. Χρησιμοποιώ επίσης έναν βελτιστοποιητή Adam και τη συνάρτηση διασταυρούμενης εντροπίας του PyTorch ως συνάρτηση απώλειας. Για την προετοιμασία του μοντέλου, χρησιμοποίησα έναν AtomEncoder σε συνδυασμό με ένα dataloader, 3 συνελκτικά φίλτρα και ένα γραμμικό επίπεδο στα κρυφά κανάλια με βάση τον αριθμό των κλάσεων στο σύνολο δεδομένων. Για το πέρασμα προς τα εμπρός, λαμβάνω πρώτα τις ενσωματώσεις κόμβων, μετά πρόσθεσα μια συνολική μέση δεξαμενή ως επίπεδο ανάγνωσης και, τέλος, χρησιμοποιώ έναν τελικό ταξινομητή.

Μετά την εκτέλεση του μοντέλου, λαμβάνουμε μια μέση τιμή συνάρτησης απώλειας ίση με 0,114 και μια μέση επικύρωση ίση με 88,352%, που είναι ένα εξαιρετικό αποτέλεσμα. Η γενική συμπεριφορά της ακρίβειας σε συνάρτηση με το μέγεθος της παρτίδας φαίνεται παρακάτω [7.10]:



Σχήμα 7.10: GCN Validation Accuracy vs Batch size

Για να επιτύχω αυτά τα αποτελέσματα, χρησιμοποίησα συνολικά 64 κρυφά επίπεδα, ποσοστό εκμάθησης ίσο με 0,01 και εγκατάλειψη ίσο με 0,5 στον τελικό ταξινομητή του μοντέλου.

Έτσι, μπορούμε να συμπεράνουμε ότι είναι πιο αποτελεσματικό να χρησιμοποιείται ένα Graph Convolutional Network αντί για ένα CNN για την ταξινόμηση των μορίων αυτού του συνόλου δεδομένων. Το μεγάλο πλεονέκτημα που έχει ένα GCN και το καθιστά πολύ ισχυρό είναι ότι μπορεί να λαμβάνει γραφήματα με μεταβλητό μέγεθος ως είσοδο, που έρχεται σε αντίθεση με ένα Συνελικτικό NN, το οποίο μπορεί να κατανοήσει είσοδο μόνο με συγκεκριμένο μέγεθος. Θα θέλαμε να κατασκευάσουμε το κβαντικό ανάλογο του για να μπορέσουμε να συγκρίνουμε τα αποτελέσματα και να καταλήξουμε σε συμπεράσματα για το κβαντικό πλεονέκτημα, αλλά όπως αναφέρεται στη βιβλιογραφία, τα κβαντικά γραφήματα NN είναι πολύ περίπλοκα μοντέλα και δεν είναι ακόμα εντελώς εφικτά, αλλά τα έχουμε μελετήσει σε μια θεωρητική επίπεδο. Όπως αναφέρθηκε σε ένα έγγραφο από το CERN, όπου εφάρμοσαν ένα QGNN για την πρόβλεψη τροχιών σωματιδίων, έχει πολύ υψηλό υπολογιστικό κόστος καθώς 1 epoch είχε διάρκεια 1 εβδομάδας.

### 3. Generative Adversarial Network (GAN)

Για αυτό το κεφάλαιο, υλοποιώ ένα Generative Adversarial Network και χρησιμοποιώ τις εικόνες των μορίων ως είσοδο για το μοντέλο GAN μου. Το αρχικό μέγεθος που εισάγω τις εικόνες είναι 600 600 pixel. Στη συνέχεια, αφού κόψω τα δεδομένα μου για εκπαίδευση, δοκιμή, σύνολα δεδομένων επικύρωσης, ορίζω τις νέες διαστάσεις για τις εικόνες μας, οι οποίες θα είναι 28 28 pixel. Χρησιμοποιώ επίσης ένα μέγεθος παρτίδας ίσο με 100, με ένα δείγμα προπόνησης μήκους ίσο με 800 και 20 epochs εκπαίδευσης. Το σύνολο δεδομένων μου έχει χωριστεί σε 11829 εικόνες για το σύνολο δεδομένων τρένου, 3944 εικόνες για το σύνολο δεδομένων επικύρωσης και 1945 εικόνες για το σύνολο δεδομένων δοκιμής, με τη χρήση των Εργαλείων δημιουργίας δεδομένων εικόνας. Μέσω της συσκευής των γεννητριών, κανονικοποιώ επίσης τις εικόνες ώστε να ταιριάζουν  $[0, 1]$ .

Στη συνέχεια, ορίζω ένα μέγεθος buffer ίσο με το μέγεθος του δείγματος εκπαίδευσης και ανακατεύω τα δεδομένα. Στη συνέχεια ορίζω το μοντέλο generator, το οποίο είναι ένα διαδοχικό μοντέλο. Ως πρώτο βήμα χρησιμοποιώ ένα Dense layer με Batch Normalization και μια λειτουργία ενεργοποίησης Leaky ReLU. Στη συνέχεια χρησιμοποιώ τρία στρώματα με ένα διδιάστατο συνελκτικό φίλτρο μετατόπισης σε συνδυασμό ξανά με Normalization παρτίδας και μια λειτουργία ενεργοποίησης Leaky ReLU. Ωστόσο, στο τρίτο επίπεδο χρησιμοποιώ μια υπερβολική εφαπτομένη, tanh, συνάρτηση ενεργοποίησης αντί για Leaky ReLU. Όλα τα παραπάνω μπορούν να συνοψιστούν στο διάγραμμα μοντέλου [7.11].

```

Model: "sequential"

```

Layer (type)	Output Shape	Param #
dense (Dense)	(None, 12544)	1254400
batch_normalization (Batch Normalization)	(None, 12544)	50176
leaky_re_lu (LeakyReLU)	(None, 12544)	0
reshape (Reshape)	(None, 7, 7, 256)	0
conv2d_transpose (Conv2DTranspose)	(None, 7, 7, 128)	819200
batch_normalization_1 (Batch Normalization)	(None, 7, 7, 128)	512
leaky_re_lu_1 (LeakyReLU)	(None, 7, 7, 128)	0
conv2d_transpose_1 (Conv2DTranspose)	(None, 14, 14, 64)	204800
batch_normalization_2 (Batch Normalization)	(None, 14, 14, 64)	256
leaky_re_lu_2 (LeakyReLU)	(None, 14, 14, 64)	0
conv2d_transpose_2 (Conv2DTranspose)	(None, 28, 28, 1)	1600

```

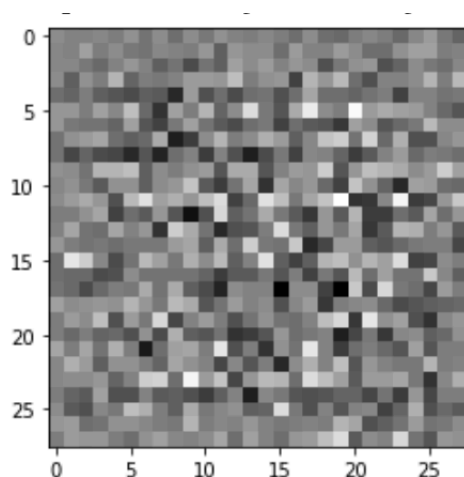
=====
Total params: 2,330,944
Trainable params: 2,305,472
Non-trainable params: 25,472

```

Σχήμα 7.11: Generator Model Summary

Στη συνέχεια προσθέτω τυχαίο θόρυβο σε μια εικόνα και την τροφοδοτώ στο generator για να δω τα αποτελέσματα [7.12].





Σχήμα 7.12: Noisy Image

Στο επόμενο βήμα ορίζω το μοντέλο discriminator, το οποίο είναι επίσης ένα διαδοχικό μοντέλο. Ως πρώτο βήμα, προσθέτω δύο διδιάστατα συνελκτικά φίλτρα σε συνδυασμό με μια λειτουργία ενεργοποίησης Leaky ReLU με dropout=0,3. Τέλος, χρησιμοποιώ ένα επίπεδο Flatten και ένα Dense layer για να ολοκληρώσω αυτό το μοντέλο.

```
Model: "sequential_3"
```

Layer (type)	Output Shape	Param #
conv2d_2 (Conv2D)	(None, 14, 14, 64)	1664
leaky_re_lu_8 (LeakyReLU)	(None, 14, 14, 64)	0
dropout_2 (Dropout)	(None, 14, 14, 64)	0
conv2d_3 (Conv2D)	(None, 7, 7, 128)	204928
leaky_re_lu_9 (LeakyReLU)	(None, 7, 7, 128)	0
dropout_3 (Dropout)	(None, 7, 7, 128)	0
flatten_1 (Flatten)	(None, 6272)	0
dense_3 (Dense)	(None, 1)	6273

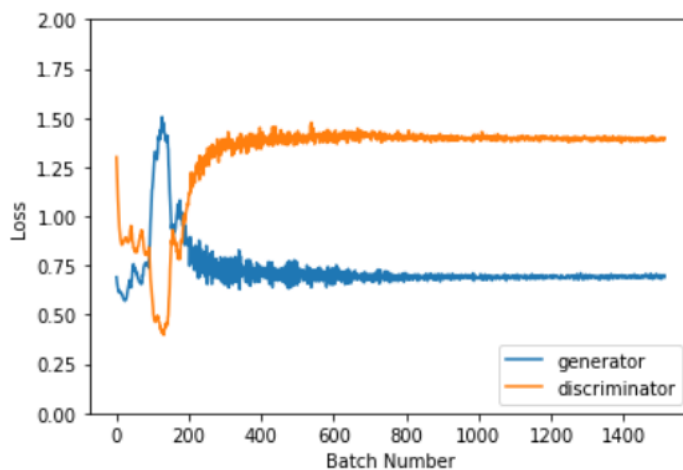
```

=====
Total params: 212,865
Trainable params: 212,865
Non-trainable params: 0
=====

```

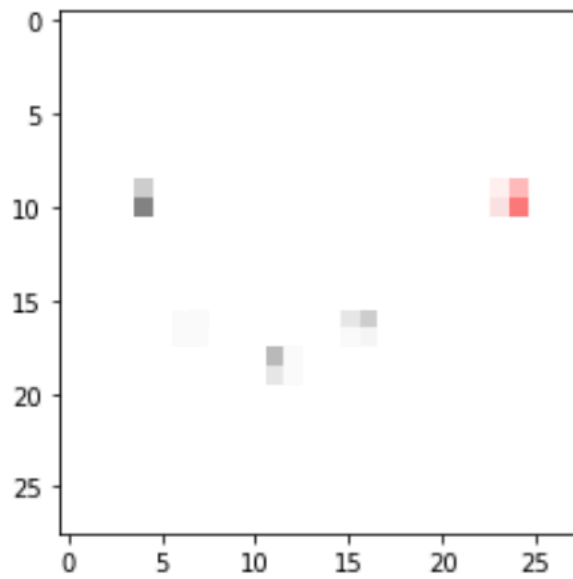
Σχήμα 7.13: Discriminator Model Summary

Στη συνέχεια χρησιμοποιώ ένα Adam optimizer και προσαρμοσμένη συνάρτηση απώλειας για κάθε μοντέλο. Ορίζω τη διάσταση του θορύβου στο 100, τις epochs στο 15 και εκπαιδεύω όλο το μοντέλο. Παρακάτω, εμφανίζω τη συνάρτηση απώλειας συγκρίνοντας το generator και τον discriminator [??].

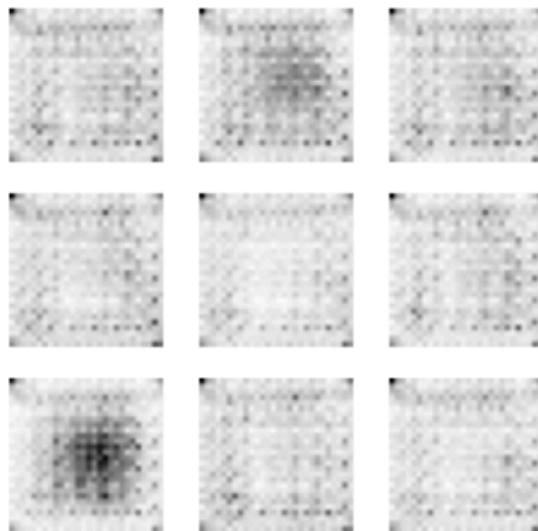


Σχήμα 7.14: Loss function of Generator vs the Discriminator as a function of batch size

Οι εικόνες που δημιουργούνται από διαφορετικές περιόδους λειτουργίας είναι [7.15] για μια εικόνα που δημιουργήθηκε σε RGB και το [7.16], όπου προσπάθησα να εργαστώ με εικόνες σε ασπρόμαυρη κλίμακα.



Σχήμα 7.15: RGB Generated Image



**Time for epoch 9 is 233.37536644935608 sec**

Σχήμα 7.16: Grayscale Generated Image

Και οι δύο εικόνες είναι πολύ θολές λόγω του θορύβου και καταλήξαμε στο γεγονός ότι εκτελέσαμε τον αλγόριθμο για λίγο αριθμό εποχών. Αν συνεχίζαμε να εκτελούμε τον αλγόριθμο για μεγάλο αριθμό εποχών θα είχαμε καλύτερη εικόνα.

## Κεφάλαιο 8

# Μοντέλα Quantum ML - Πειραματική Υλοποίηση

### 1. Πώς να "τρέξετε" κβαντικούς αλγόριθμους

Το πιο σημαντικό ερώτημα είναι σε ποια συσκευή εκτελούμε τους κβαντικούς αλγόριθμους μας και δοκιμάζουμε τα κβαντικά μας κυκλώματα; Αν θέλουμε να περιγράψουμε την περίπτωση της βιβλιοθήκης Qiskit, μπορούμε να χρησιμοποιήσουμε τα backends της IBM. Τις περισσότερες φορές τα κβαντικά κυκλώματα εκτελούνται σε προσομοιωτές Qiskit. Πώς υπολογίζουμε ένα κβαντικό κύκλωμα σε έναν κλασικό υπολογιστή; Πρέπει να επανεξετάσουμε τι είναι πραγματικά μια κβαντική κατάσταση και τελεστής. Όπως χρησιμοποιήθηκε στο προηγούμενο κεφάλαιο, στην κβαντική μηχανική, χρησιμοποιούμε τον συμβολισμό Dirac που περιέχει τα *bra - ket*,  $| \rangle$ . Σε μια απλή μαθηματική γλώσσα, μπορούμε να μεταφράσουμε αυτή την κατάσταση σε διάνυσμα:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

Οι κβαντικοί τελεστές μπορούν και πάλι να μεταφραστούν σε πίνακες. Αν για παράδειγμα χρησιμοποιήσουμε τον τελεστή Hadamard,  $H$ , χρησιμοποιούμε έναν πίνακα της μορφής:

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

Έτσι, μπορεί κανείς εύκολα να καταλάβει ότι ο υπολογισμός ενός κβαντικού κυκλώματος είναι απλώς ένας πολλαπλασιασμός πίνακα. Το διάνυσμα που προκύπτει υποδηλώνει τα πλάτη μέτρησης, των οποίων τα τετράγωνα είναι οι πιθανότητες

μέτρησης. Εδώ είναι ένα σημείο που μπορεί κανείς να καταλάβει πιο εύκολα ένα μέρος της κβαντικής υπεροχής. Καθώς όλες αυτές οι πράξεις μπορούν να θεωρηθούν ως πολλαπλασιασμοί μήτρας, μπορεί κανείς εύκολα να καταλάβει ότι εάν οι διαστάσεις του πίνακα είναι πολύ μεγάλες, αυτή η λειτουργία θα ήταν πολύ χρονοβόρα και δαπανηρή για έναν κλασικό υπολογιστή. Αν το θέσουμε με απλό τρόπο, οι κβαντικοί υπολογιστές μπορούν να πολλαπλασιάσουν πολλούς πίνακες σε ένα μόνο βήμα.

Ένας από τους πιο δημοφιλείς παρόχους για προσομοιωτές είναι αυτός που παρέχει backends Basic Aer. Αυτή είναι μια ενότητα κβαντικών προσομοιωτών που βασίζονται σε Python. Υπάρχουν αρκετές υλοποιήσεις. Κάποιος μπορεί να χρησιμοποιήσει τον προσομοιωτή qasm για να ανακτήσει εμπειρικά τις μετρήσεις μέτρησης. Αυτός ο προσομοιωτής λαμβάνει υπόψη μόνο τα κλασικά bit που χρησιμοποιούμε όταν μετράμε τα qubits μας. Χρησιμοποιεί μια παράμετρο που ονομάζεται "shots" για να υποδείξει στον υπολογιστή πόσες φορές να τρέξει το κύκλωμα και να λάβει το μετρημένο αποτέλεσμα. Καθώς αυξάνουμε τον αριθμό των βολών, αυξάνουμε την ακρίβεια των αποτελεσμάτων μας. Αυτός ο προσομοιωτής μπορεί να μην είναι η πιο ακριβής επιλογή, αλλά παρέχει στον χρήστη τις πιο ρεαλιστικές συνθήκες, όπως ο θόρυβος. Εάν κάποιος θέλει να χρησιμοποιήσει έναν προσομοιωτή για να υπολογίσει την ακριβή κατάσταση στην οποία βρίσκεται ένα qubit, πιθανότατα θα πρέπει να επιλέξει τον προσομοιωτή διανυσμάτων κατάστασης. Ωστόσο, μια σημαντική σημείωση πριν χρησιμοποιήσετε αυτόν τον προσομοιωτή είναι να αφαιρέσετε όλες τις μετρήσεις στο κβαντικό μας κύκλωμα. Όπως αναφέρθηκε σε προηγούμενο κεφάλαιο αυτής της διατριβής, εάν μετρήσουμε την κατάσταση οποιουδήποτε qubit στο κύκλωμά μας, αυτό θα καταρρεύσει την κβαντική υπέρθεση και αναπόφευκτα θα οδηγήσει σε μια καθορισμένη κατάσταση στην οποία θα μπορούσε ενδεχομένως να βρίσκεται το σύστημα. Το backend του προσομοιωτή διανυσμάτων κατάστασης υπολογίζει την κατάσταση του το δεδομένο κβαντικό σύστημα. Τέλος, ένας άλλος δημοφιλής προσομοιωτής που παρέχεται από το πακέτο Basic Aer είναι ο ενιαίος προσομοιωτής. Αυτός ο προσομοιωτής εκτελεί το κύκλωμα μία φορά και επιστρέφει τον τελικό πίνακα μετασχηματισμού του ίδιου του κυκλώματος. Η εκτέλεση του κυκλώματος σε μια κατάσταση εισόδου απλώς πολλαπλασιάζει τον πίνακα μετασχηματισμού με το διάνυσμα κατάστασης. Όπως και πριν, όταν χρησιμοποιούμε αυτόν τον προσομοιωτή το κύκλωμά μας δεν πρέπει να περιέχει μετρήσεις.

Εκτός από όλους τους προσομοιωτές, μερικοί κβαντικοί υπολογιστές έχουν κατασκευαστεί και είναι διαθέσιμοι για δημόσια χρήση, φιλοξενούμενοι από την IBM-Quantum. Όλες οι κλήσεις εκτελούνται μέσω του IBMQ API μέσω Διαδικτύου και κάποιος πρέπει να εγγραφεί για να αποκτήσει ένα κλειδί API για πρόσβαση στον λογαριασμό του και εκτέλεση του κώδικά του στους απομακρυσμένους διακομιστές. Το πρώτο βήμα είναι να φορτώσετε τον λογαριασμό σας IBMQ χρησιμοποιώντας το κλειδί API. Αφού φορτωθεί ο λογαριασμός, μπορείτε να ανακτήσετε έναν πάροχο

υποστήριξης για να εκτελέσει τον κώδικά σας σε έναν κβαντικό υπολογιστή. Σε αυτά τα backends, μπορείτε να βρείτε είτε προσομοιωτές είτε πραγματικούς κβαντικούς υπολογιστές. Ως είσοδος λαμβάνουν ένα qobject, το οποίο είναι η μορφή σειριοποίησης του Qiskit API και επιστρέφει ένα αντικείμενο BaseJob. Αυτό το αντικείμενο επιτρέπει την ασύγχρονη εκτέλεση εργασιών για την ανάκτηση αποτελεσμάτων από ένα backend όταν ολοκληρωθεί η εργασία. Κατά την εκτέλεση του κβαντικού κυκλώματος, μπορείτε να παρακολουθείτε την κατάσταση εκτέλεσης μιας εργασίας σε έναν κβαντικό υπολογιστή χρησιμοποιώντας πολλές εντολές όπως φαίνεται παρακάτω.

```
1 from qiskit.tools import job_monitor
2
3 backend2 = qiskit.BasicAer.get_backend('qasm_simulator')
4 backend2.name() # qasm_simulator
5 backend2.version # 1
6 backend2.provider().backends() # [<QasmSimulatorPy('qasm_simulator')>,
7
8 job = qiskit.execute(qc, backend2)
9
10 job_monitor(job) # Problem status of the running job
11 # Job Status: job has successfully run
12
13 job.status()
14 # <JobStatus.DONE: 'job has successfully run'>
15
16 job.result().get_counts()
```

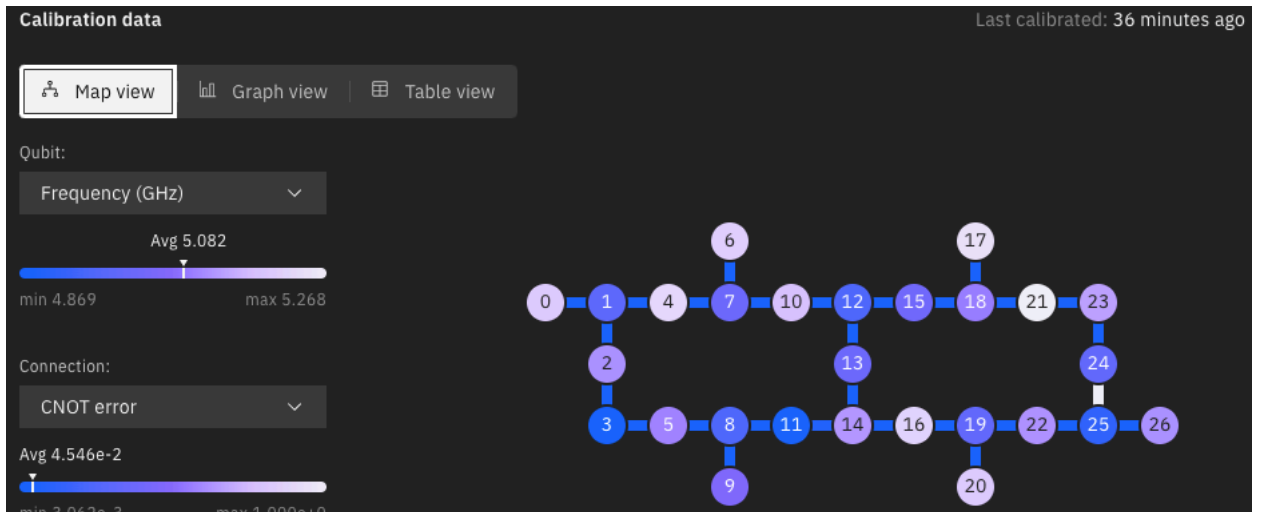
Σχήμα 8.1: Job Monitoring

Όταν επιλέγετε έναν πραγματικό κβαντικό υπολογιστή, πρέπει να είστε προσεκτικοί. Είναι πολύ σημαντικό να επιλέξει έναν κβαντικό υπολογιστή που μπορεί να του προσφέρει την πιο κατάλληλη αρχιτεκτονική για τη συγκεκριμένη χρήση. Μπορεί κανείς να δει όλα τα διαθέσιμα συστήματα επισκεπτόμενος τους προσωπικούς του λογαριασμούς.

Name	Qubits	QV	CLOPS	Status	Total pending jobs	Processor type
<a href="#">ibm_washington</a> <span>Exploratory</span>	127	64	850	● Online - Queue paused	1784	Eagle r1
<a href="#">ibmq_brooklyn</a> <span>Exploratory</span>	65	32	1.5K	● Online	26	Hummingbird r2
<a href="#">ibmq_kolkata</a>	27	128	2K	● Online	91	Falcon r5.11
<a href="#">ibmq_montreal</a>	27	128	2K	● Online	280	Falcon r4
<a href="#">ibmq_mumbai</a> <span>Exploratory</span>	27	128	1.8K	● Online	658	Falcon r5.1
<a href="#">ibm_cairo</a>	27	64	2.4K	● Online - Queue paused	308	Falcon r5.11
<a href="#">ibm_auckland</a> <span>Exploratory</span>	27	64	2.4K	● Online	572	Falcon r5.11
<a href="#">ibm_hanoi</a>	27	64	2.3K	● Online	303	Falcon r5.11
<a href="#">ibmq_toronto</a>	27	32	1.8K	● Online	128	Falcon r4
<a href="#">ibm_peekskill</a> <span>Exploratory</span>	27	-	-	● Online	0	Falcon r8

Σχήμα 8.2: Real Quantum Computers

Όπως φαίνεται στην παραπάνω εικόνα, παρέχονται όλες οι προδιαγραφές, συμπεριλαμβανομένου του εάν ο κβαντικός υπολογιστής είναι διαθέσιμος. Στη συνέχεια παρέχονται ορισμένες αρχιτεκτονικές πληροφορίες, όπως ο αριθμός των qubits και ο Κβαντικός Όγκος (QV) που είναι μια μέτρηση που μετρά τις δυνατότητες και τα ποσοστά σφαλμάτων ενός κβαντικού υπολογιστή. Η μέθοδος QV ποσοτικοποιεί το μεγαλύτερο τυχαίο κύκλωμα ίσου πλάτους και βάθους που μπορεί να εφαρμόσει με επιτυχία ο υπολογιστής. Συνεχίζοντας, για κάθε σύστημα έχουμε τη μέτρηση CLOPS, η οποία σημαίνει Λειτουργίες βρόχου κυκλώματος ανά δευτερόλεπτο. Αυτή είναι μια μέτρηση που συσχετίζεται με το πόσο γρήγορα ένας κβαντικός επεξεργαστής μπορεί να εκτελέσει κυκλώματα. Πιο συγκεκριμένα, μετρά την ταχύτητα που ο επεξεργαστής μπορεί να εκτελέσει επίπεδα ενός παραμετροποιημένου κυκλώματος μοντέλου του ίδιου είδους που χρησιμοποιείται για τη μέτρηση του QV. Τέλος, μπορεί κανείς να δει τις εργασίες που εκκρεμούν να γίνουν στο συγκεκριμένο σύστημα και τον τύπο του επεξεργαστή στον οποίο τρέχουν. Επιπλέον, σε αυτή τη σελίδα μπορείτε να βρείτε περισσότερες πληροφορίες για κάθε σύστημα, όπως οι συχνότητες των qubits σε GHz και τα ποσοστά σφαλμάτων.



Σχήμα 8.3: ibmq\_kolkata error map

Στην παραπάνω εικόνα, οι πληροφορίες που αναφέρονται παραπάνω εμφανίζονται για το σύστημα IBM kolkata μαζί με τον χάρτη σφαλμάτων του. Ως χάρτης σφαλμάτων ορίζουμε την απεικόνιση των συνδέσεων κόμβων ενός χάρτη πύλης, επιπλέον του αναμενόμενου ποσοστού σφάλματος στο backend. Χρησιμοποιώντας τον όρο χάρτης πύλης αναφερόμαστε στην οπτικοποίηση των συνδέσεων μεταξύ των κόμβων σε μια φυσική συσκευή κβαντικού υπολογισμού.

Τώρα, εάν κάποιος χρησιμοποιεί το πακέτο PennyLane σε συνδυασμό με το Qiskit, πρέπει να γνωρίζει ότι αυτή η βιβλιοθήκη κβαντικής μηχανικής εκμάθησης έχει σχεδιαστεί από την αρχή για να είναι αγνωστικιστής υλικού και συσκευών, επιτρέποντας την εύκολη αποστολή κβαντικών λειτουργιών σε διαφορετικές συσκευές. Ένας μεμονωμένος υπολογισμός μπορεί ακόμη και να περιλαμβάνει πολλές κβαντικές συσκευές από διαφορετικούς προμηθευτές. Το PennyLane προσφέρει πολλές ενσωματωμένες κβαντικές συσκευές, όπως απλός προσομοιωτής qubit διανυσμάτων κατάστασης γραμμένο σε Python ή χρησιμοποιώντας TensorFlow και υποστηρίζει την κλασική οπίσθια διάδοση. Άλλες επιλογές περιλαμβάνουν έναν γρήγορο προσομοιωτή qubit διανυσμάτων γραμμένο με backend C++ ή έναν προσομοιωτή qubit μικτής κατάστασης γραμμένο σε Python. Πολλά πρόσθετα μπορούν να εγκατασταθούν ξεχωριστά, συμπεριλαμβανομένων των ενσωματώσεων με Qiskit, Amazon Braket, Cirq, Strawberry Fields και άλλα.



## 2. Σύγκριση μεθόδων κωδικοποίησης για ταξινομητές κβαντικών μεταβλητών

Πριν προχωρήσουμε στην κατασκευή διαφορετικών τύπων κβαντικών κυκλωμάτων για να μετατρέψουμε τα κλασικά μοντέλα στο κβαντικό ανάλογό τους. Σε αυτήν την ενότητα, στοχεύω να μελετήσω διαφορετικές τεχνικές κωδικοποίησης των πληροφοριών που παρέχονται από το σύνολο δεδομένων μου, το οποίο είναι υποσύνολο της βάσης δεδομένων QM9. Σχεδιάζω ένα μεταβλητό κβαντικό κύκλωμα για κάθε κωδικοποίηση και χρησιμοποιώ τις 4 από τις 5 στήλες του συνόλου δεδομένων μου ως στόχο για την υλοποίηση μιας βασικής δυαδικής ταξινόμησης 0 και 1. Οι ταξινομητές που χρησιμοποιούνται βασίζονται στη στήλη που περιέχει τα δεδομένα συντελεστή  $\log P$  και τη στήλη που περιέχει τον ατομικό αριθμό κάθε μορίου. Σκέφτηκα να χρησιμοποιήσω ένα στρώμα για να σχεδιάσω το ansatz που θα χρησιμοποιήσω για αυτά τα μοντέλα και θα μελετήσω με ποιον τρόπο ο αριθμός του στρώματος επηρεάζει την απόδοση του κβαντικού μας κυκλώματος. Έτσι, όλες οι διαδικασίες που φαίνονται παρακάτω μπορούν να περιγραφούν ως τα ακόλουθα βήματα:

1. Προεπεξεργασία δεδομένων,
2. Κβαντική ενσωμάτωση των κλασικών δεδομένων και
3. Εκπαίδευση ενός πολυεπίπεδου μεταβλητού κβαντικού ταξινομητή

Ξεκινάω τη μελέτη αυτού του κεφαλαίου διαμορφώνοντας το σύνολο δεδομένων μου. Χρησιμοποιώ το αρχικοποιημένο σύνολο δεδομένων

Unnamed: 0	SMILES	logP
0	<chem>C1C2CC1C2</chem>	0.63610
1	<chem>C1C2CC1O2</chem>	0.16200
2	<chem>CC12CC(C1)C2</chem>	-0.82470
3	<chem>CC12CC(C1)O2</chem>	0.24940
4	<chem>CC12CN(C1)C2</chem>	0.13978
...	...	...
19667	<chem>FC(F)(F)C12CC(C1)O2</chem>	1.13150
19668	<chem>FC(F)(F)C12CN(C1)C2</chem>	0.07450
19669	<chem>FC(F)(F)C1C2CC1C2</chem>	-0.06800
19670	<chem>FC(F)(F)C1C2CC1O2</chem>	0.26280
19671	<chem>FC(F)(F)C1C2CN1C2</chem>	0.07450

Σχήμα 8.4: Initial Dataset

και χρησιμοποιώντας αυτό αποκτώ αριθμητικά χαρακτηριστικά για κάθε μόριο που είναι ζωτικής σημασίας για την κατανόηση των ιδιοτήτων του και που καθορίζουν τη δομή κάθε μορίου. Τα χαρακτηριστικά που υπολογίστηκαν είναι

- ▶ Number of atoms,
- ▶ Number of bonds,
- ▶ Atomic Number (Z),
- ▶ logP coefficient,
- ▶ Number of single bonds and
- ▶ Number of double bonds

Τέλος, το σύνολο δεδομένων που θα χρησιμοποιηθεί για όλες τις παρακάτω τεχνικές είναι το παρακάτω:

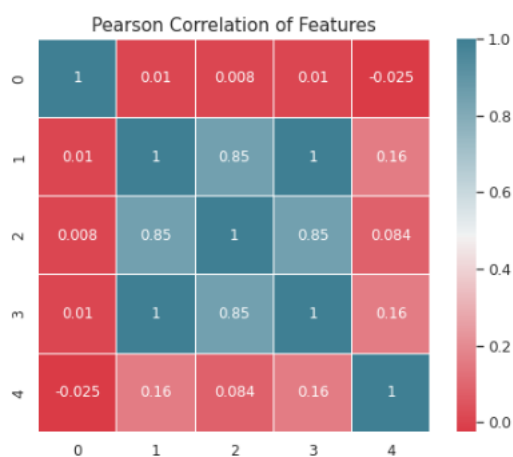
	logP	atomic_numbers	num_atoms	Z	double_bonds
0	0.63610	30	5	30	0
1	0.16200	32	5	32	0
2	-0.82470	36	6	36	0
3	0.24940	38	6	38	0
4	0.13978	37	6	37	0

Σχήμα 8.5: Dataset

και χρησιμοποιώντας τη βιβλιοθήκη pandas λαμβάνω τα παρακάτω στατιστικά στοιχεία, τα οποία χαρακτηρίζουν το συγκεκριμένο σύνολο δεδομένων:

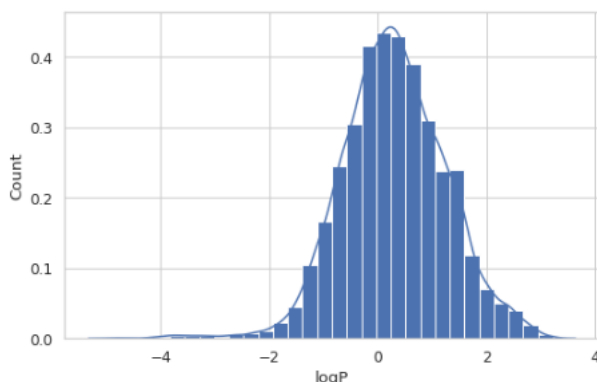
	logP	atomic_numbers	num_atoms	Z	double_bonds
count	19672.000000	19672.000000	19672.000000	19672.000000	19672.000000
mean	0.301025	56.366816	8.870730	56.366816	0.494866
std	0.963120	2.898807	0.380119	2.898807	0.604471
min	-4.909700	30.000000	5.000000	30.000000	0.000000
25%	-0.304680	56.000000	9.000000	56.000000	0.000000
50%	0.283300	57.000000	9.000000	57.000000	0.000000
75%	0.936770	58.000000	9.000000	58.000000	1.000000
max	3.366800	65.000000	9.000000	65.000000	3.000000

Για να κατανοήσουμε καλύτερα το σύνολο δεδομένων μας, παρέχω ορισμένα πρόσθετα στατιστικά στοιχεία, όπως ένα θερμικό χάρτη της συσχέτισης χαρακτηριστικών Pearson



Σχήμα 8.6: Pearson Correlation

και το εύρος τιμών της πρώτης στήλης



Σχήμα 8.7: Range of logP coefficient

Τέλος, εάν κάποιος θέλει περισσότερες πληροφορίες για την περιγραφή του συνόλου δεδομένων, μπορεί να κοιτάξει στο Παράρτημα αυτής της διατριβής, το οποίο περιέχει ένα ζεύγος γραφήματος του συνόλου δεδομένων.

Μετά από όλα τα στατιστικά στοιχεία που εμφανίζονται παραπάνω, συνεχίζω με την προεπεξεργασία των χαρακτηριστικών του συνόλου δεδομένων. Ξεκινάω κανονικοποιώντας τις στήλες στα σύνολα δεδομένων αμαξοστοιχίας και δοκιμής ώστε να χωρούν στο εύρος  $[0,1]$ . Μετά την επανακλιμάκωση, χωρίζω το πλαίσιο δεδομένων δοκιμής στις αντίστοιχες ετικέτες συνόλου χαρακτηριστικών και γειώνω την αλήθεια και παγώνω τα δεδομένα εκπαίδευσης και δοκιμής εισαγωγής. Τέλος, μετατοπίζω τις ετικέτες αλήθειας από 0, 1 σε -1, 1 για να ταιριάζουν με τις προσδοκώμενες τιμές του πίνακα Pauli Z. Επιπλέον, χωρίζω τα δεδομένα μου για να εκπαιδεύσω και να δοκιμάσω υποσύνολα και να τα ανακατέψω και να τα χωρίσω σε σετ εκπαίδευσης και επικύρωσης.

#### Τεχνική Amplitude Encoding:

Τώρα, ως πρώτη τεχνική δοκιμάζω τη μέθοδο Amplitude Encoding. Επειδή έχω 4 χαρακτηριστικά στο σύνολο δεδομένων μου, αποφασίζω να χρησιμοποιήσω 2 qubit και χρησιμοποιώ τη συσκευή "default.qubit" που παρέχεται από το πακέτο pennylane και χρησιμοποιώ τις προεπιλεγμένες λήψεις=100, για το πόσες φορές θα επαναλάβω τις μετρήσεις των κυκλωμάτων, για να παρέχει τα τελικά αποτελέσματα. Αρχικά, ορίζω ένα πρότυπο στρώματος ως μια ακολουθία εκπαιδευσιμων πυλών. Αυτό είναι παρόμοιο με τα επίπεδα σε ένα νευρωνικό δίκτυο. Αυτό μπορεί να επιτευχθεί χρησιμοποιώντας ένα μόνο qubit για περιστροφές και μερικές ελεγχόμενες πύλες NOT (CNOT) ως εμπλοκές μεταξύ των ζευγών qubits. Ως δεύτερο

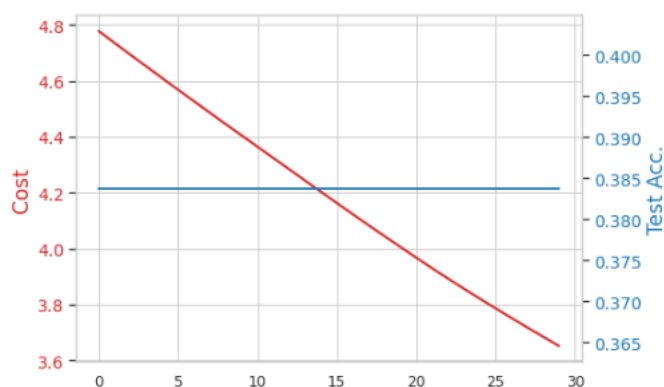
βήμα, αρχικοποιώ την κβαντική συσκευή και προετοιμάζω το κβαντικό σύνολο δεδομένων χρησιμοποιώντας πύλες Hadamard. Μετά την αρχικοποίηση, μπορώ τελικά να ενσωματώσω τα κλασικά μου δεδομένα ως κβαντική κατάσταση στον χώρο Hilbert και να δημιουργήσω το κβαντικό μου μοντέλο.

Για να μπορέσω να μετρήσω την απόδοση των τεχνικών, ορίζω μια συνάρτηση απώλειας και συγκεκριμένα μια τετραγωνική συνάρτηση απώλειας και την ακρίβεια για το μοντέλο δυαδικής ταξινόμησης. Από την άλλη, ορίζω το μοντέλο του κβαντικού ταξινομητή μεταβλητής και την αντίστοιχη συνάρτηση απώλειας και εκπαιδεύω το μοντέλο. Για την εκπαίδευση, αρχικοποιώ τις παραμέτρους, τον αριθμό των επιπέδων ως 1, τα αρχικά βάρη και την αρχική προκατάληψη του μοντέλου.

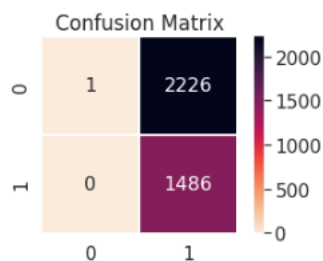
Επίσης χρησιμοποίησα batch-size=120, epochs=30, learning rate=0.01 για έναν Adam optimizer. Τα βάρη εξόδου που λαμβάνω είναι ίσα με

$$\begin{pmatrix} 0.02082524 & 0.00774552 & -0.01176826 \\ -0.00760863 & -0.00517855 & 0.00466557 \end{pmatrix} \quad (8.1)$$

Τα αποτελέσματα όμως, εμφανίζονται παρακάτω.



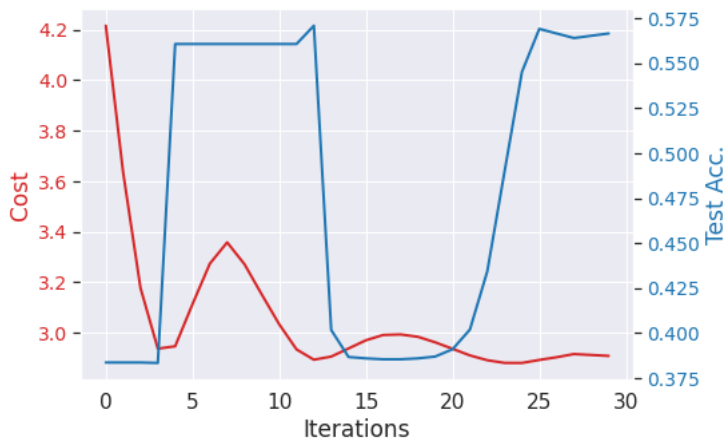
Σχήμα 8.8: Model Accuracy Cost for 1 layer



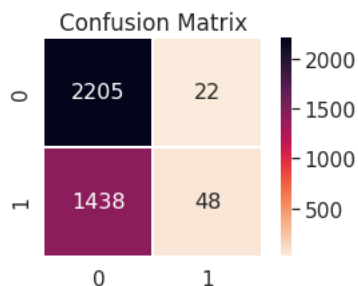
Σχήμα 8.9: Confusion Matrix of the single layer model

Όπως μπορεί να παρατηρηθεί από την πρώτη γραφική παράσταση, η ακρίβεια παραμένει σταθερή και είναι ίση με 38.5%, που είναι πολύ κακό και επίσης η απώλεια έχει γραμμική φθίνουσα σειρά αντί για  $x^2$  φθίνουσα συμπεριφορά. Έτσι, συμπεραίνουμε ότι το μοντέλο αποτυγχάνει να ταξινομήσει τα δεδομένα.

Για να συνεχίσω την ανάλυσή μας, προσπαθώ να χρησιμοποιήσω batch-size=170, epochs=30, Learning rate=0,01 και 2 layers για το ίδιο μοντέλο. Αυτή τη φορά έλαβα τα παρακάτω αποτελέσματα:



Σχήμα 8.10: Model Accuracy Cost for 2 layers

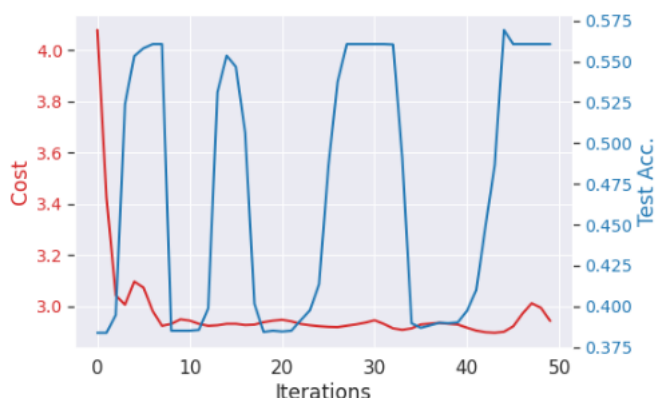


Σχήμα 8.11: Confusion Matrix of the two-layer model

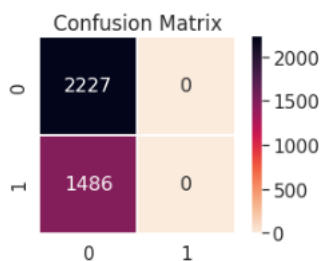
Εδώ παρατηρούμε μια μικρή βελτίωση καθώς η ακρίβεια του μοντέλου αυξήθηκε περίπου 25% και έφτασε σε μια μέση τιμή 57.26%. Επίσης η συνάρτηση κόστους έχει καλύτερη συμπεριφορά καθώς είναι φθίνουσα και μη γραμμική και φαίνεται ότι τις τελευταίες epochs συγκλίνει αργά.

Τέλος, προσπάθησα για τις ίδιες παραμέτρους να αυξήσω τον αριθμό των στρώσεων σε 3. Τα αποτελέσματα ήταν παρόμοια με την προηγούμενη περίπτωση, όσον

αφορά την ακρίβεια, καθώς έφτασε ξανά στο 56.595%. Ωστόσο, η συνάρτηση κόστους γίνεται πιο ομαλή και η σύγκλιση του μοντέλου είναι ταχύτερη και πιο ομαλή.



Σχήμα 8.12: Model Accuracy Cost for 3 layers



Σχήμα 8.13: Confusion Matrix of the three-layer model

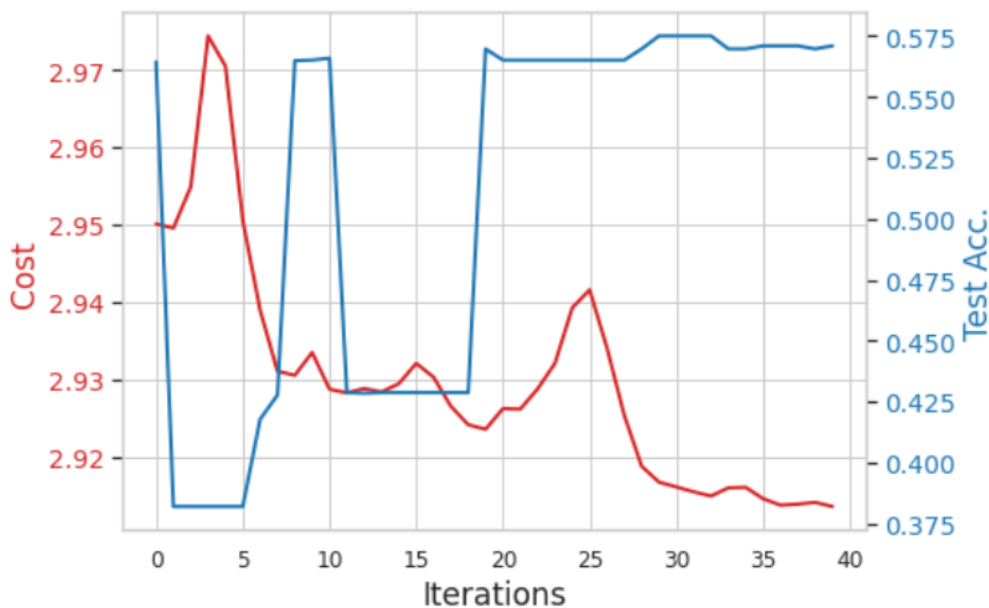
Έτσι, το γενικό συμπέρασμα για αυτήν την τεχνική είναι ότι δεν μπορούμε να φτάσουμε σε πολύ υψηλή ακρίβεια δοκιμής. Ωστόσο, εάν χρησιμοποιούσαμε έναν πιο σύνθετο ταξινομητή, μπορεί να δώσει καλύτερα αποτελέσματα. Ωστόσο, παρατηρήσαμε ότι αυξάνοντας τον αριθμό των επιπέδων στον ταξινομητή μεταβλητών, καταφέρνουμε να επιταχύνουμε τη σύγκλιση του μοντέλου και επίσης γίνεται πιο ομαλή. Η απόδοση του μοντέλου είναι αυξημένη. Για τον συγκεκριμένο βελτιστοποιητή, τον Adam, με ρυθμό εκμάθησης 0,1 εξακολουθούμε να επιτυγχάνουμε ταχύτερη αρχική σύγκλιση, ωστόσο όπως μπορεί να δει κανείς στα διαγράμματα, η απόδοση είναι ασταθής και κορεστεί γρήγορα. Εάν μειώσουμε τον ρυθμό εκμάθησης, ο ταξινομητής έχει πιο αργή αλλά ομαλή σύγκλιση.

#### Τεχνική Angle Encoding:

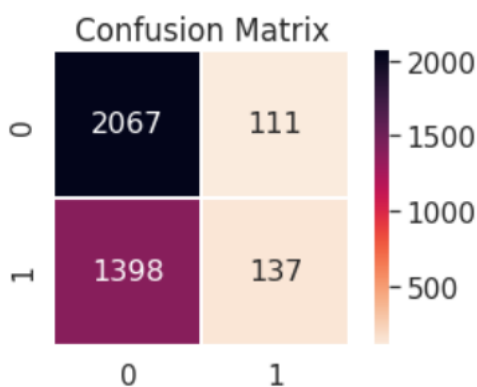
Σε αυτή τη μέθοδο ακολουθώ την ίδια ακριβώς ροή προεπεξεργασίας για τα προεκπαιδευμένα δεδομένα. Η μεγάλη διαφορά μεταξύ αυτής της τεχνικής και της

προηγούμενης είναι ο αριθμός των qubits που χρειάζονται για την υλοποίηση. Δεδομένου ότι υπάρχουν 4 χαρακτηριστικά στο σύνολο δεδομένων μας, χρειαζόμαστε 4 qubits για να εκτελέσουμε την κωδικοποίηση. Στην κωδικοποίηση γωνίας,  $N$  χαρακτηριστικά κωδικοποιούνται στις γωνίες περιστροφής των  $n$  qubits, όπου  $N \leq n$ .

Ξεκινάω πάλι με το μοντέλο ενός στρώματος. Οι παράμετροι που αρχικοποιήθηκαν είναι batch-size=150, total-iterations=40, learning-rate=0,1. Τα αποτελέσματα που λαμβάνουμε δεν είναι πολλά υποσχόμενα



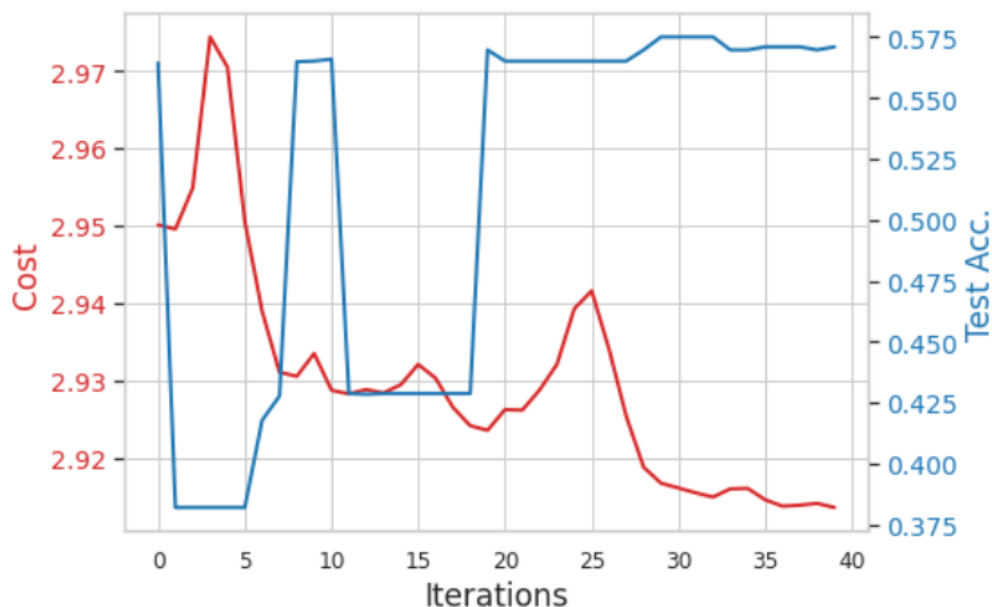
Σχήμα 8.14: Model Loss Accuracy stats - One layer



Σχήμα 8.15: Confusion Matrix of single-layer problem

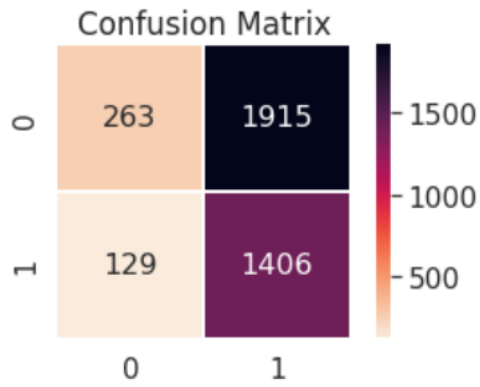


Επιπροσθέτως, επαναλαμβάνω την ίδια διαδικασία με τις ίδιες παραμέτρους όπως πριν, εκτός από τον ρυθμό εκμάθησης, τον οποίο μειώσαμε σε  $\text{lr}=0,01$ . Τα αποτελέσματα μπορείτε να τα δείτε παρακάτω:



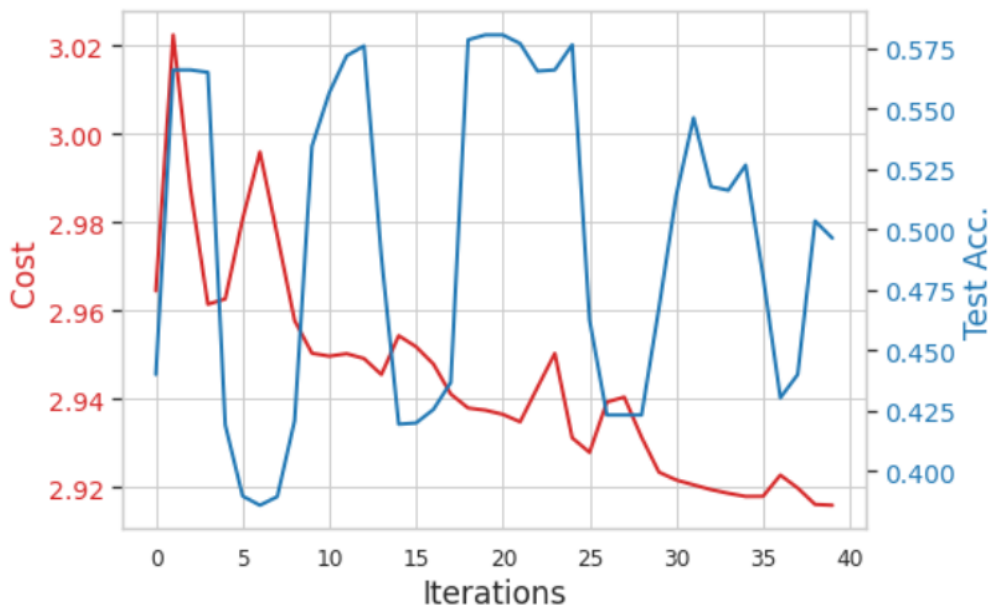
Σχήμα 8.16: Model Loss Accuracy - single-layer model

Και πάλι σε αυτή την περίπτωση βλέπουμε ότι από την ακρίβεια και το μοντέλο σύγχυσης ότι το μοντέλο μας αποτυγχάνει να ταξινομήσει τα δεδομένα μας, με μέση ακρίβεια 37%. Αντίθετα, βλέπουμε ότι η συνάρτηση απώλειας γίνεται πιο ομαλή και συγκλίνει, επομένως μπορούμε να συμπεράνουμε ότι κάτι δεν πάει καλά εδώ. Η κακή απόδοση μπορεί να εξηγηθεί ως προσθήκη επιπέδων στο κύκλωμά μας του χρόνου εκπαίδευσης για κάθε epoch και το μοντέλο πρέπει να εκπαιδευτεί για περισσότερες epochs. Έτσι, πιστεύουμε ότι εάν εκπαιδεύσουμε το μοντέλο μας για περισσότερες epochs, το κύκλωμα θα έχει την καλύτερη ακρίβεια και θα έχει καλύτερη απόδοση από τα προηγούμενα κυκλώματα που έχουν λιγότερα στρώματα. Ωστόσο, αυτό έχει μεγάλο υπολογιστικό κόστος και δεν μπορούμε να ικανοποιήσουμε αυτήν την ανάγκη σε αυτή τη διαδικασία.

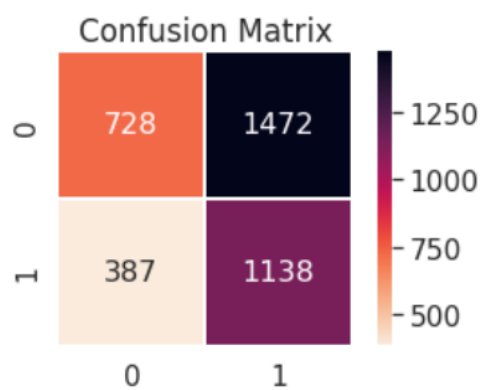


Σχήμα 8.17: Confusion Matrix of single-layer problem with lower lr

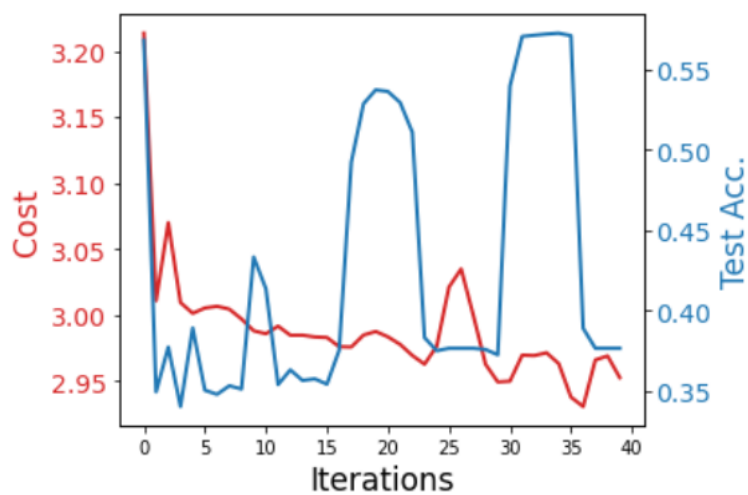
Μετά από αυτό επανέλαβα το πρώτο ολόκληρο βήμα, για να αποκτήσω καλύτερες γνώσεις και χρησιμοποίησα 2 κβαντικά στρώματα για κωδικοποίηση. Τα αντίστοιχα αποτελέσματα μπορούν να εμφανίζονται στο []:



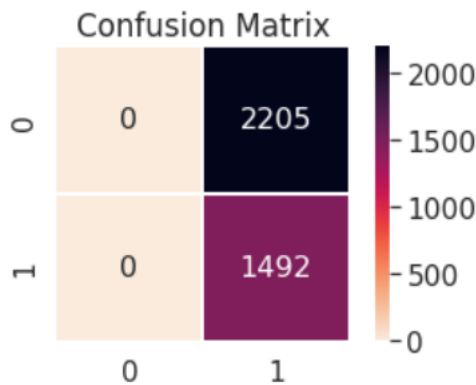
Σχήμα 8.18: Model Loss Accuracy - two-layer model



Σχήμα 8.19: Confusion Matrix of single-layer problem with  $lr=0.1$



Σχήμα 8.20: Model Loss Accuracy - three-layer model



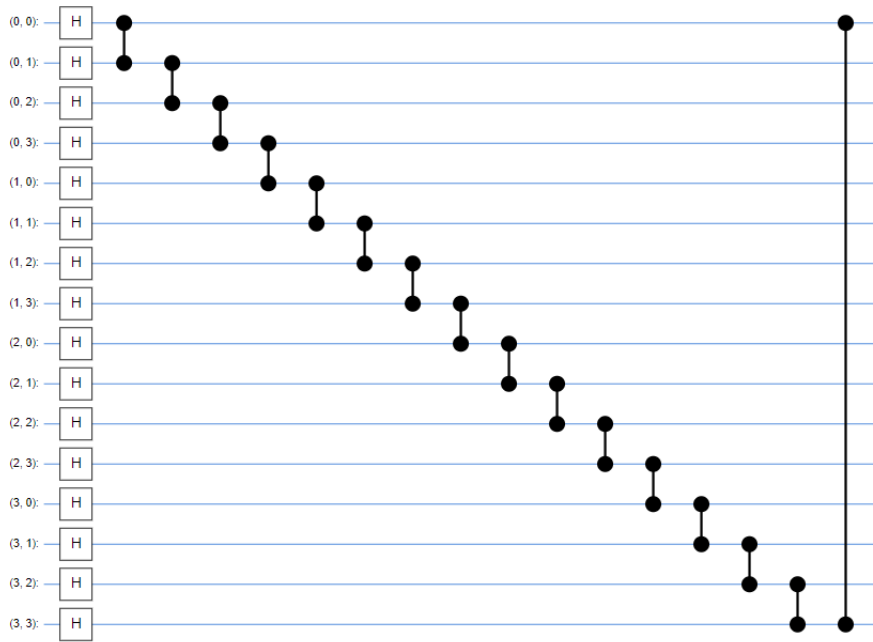
Σχήμα 8.21: Confusion Matrix of three-layer problem with  $lr=0.1$

### 3. Quantum Convolutional Neural Networks (QCNN)

Σε αυτό το κεφάλαιο, προσπάθησα να εφαρμόσω το κβαντικό ανάλογο του Συνελκτικού Νευρωνικού Δικτύου που δημιούργησα σε προηγούμενο κεφάλαιο. Όπως είδαμε, στο κλασικό μοντέλο οι βέλτιστες παράμετροι ήταν ο ρυθμός εκμάθησης ίσος με  $lr=0,01$  και το μέγεθος παρτίδας=100. Έτσι, κράτησα αυτές τις παραμέτρους για να μπορέσω να συγκρίνω τα δύο προβλήματα και να βγάλω ένα συμπέρασμα σχετικά με το ερώτημα: μπορούμε να επιτύχουμε κβαντική υπεροχή στον τομέα των μοντέλων του CNN;

Πρώτον, το πρόβλημα που μελετάμε σε αυτή την περίπτωση είναι το ίδιο με το κλασικό ανάλογο του. Εξετάζουμε την ταξινόμηση του ίδιου συνόλου δεδομένων μορίου, το οποίο έχει χωριστεί σε τέσσερις κατηγορίες, με βάση τον συντελεστή  $\log P$ . Σε αυτήν την περίπτωση, χρησιμοποιήσαμε μια εικόνα από το generator αλλά δεν πραγματοποιήσαμε augmentation των δεδομένων. Όπως παρατηρήθηκε στην κλασική περίπτωση, πιθανότατα θα προκαλέσει υπερπροσαρμογή και επειδή δεν το χρησιμοποιήσαμε στο κλασικό μοντέλο, θέλουμε να διατηρήσουμε τις ίδιες συνθήκες και για το κβαντικό πρόβλημα. Όλες οι εικόνες κανονικοποιήθηκαν σε  $[0, 1]$  και έχουμε χωρίσει το σύνολο δεδομένων μας σε δώδεκα μέρη. Το έχουμε χωρίσει σε σύνολο δεδομένων εκπαίδευσης, δοκιμής και επικύρωσης και το καθένα περιέχει τις τέσσερις κλάσεις. Μετά τον διαχωρισμό, έχουμε 11829 εικόνες στο σύνολο δεδομένων εκπαίδευσης, 3944 εικόνες για χρήση στο σύνολο δεδομένων επικύρωσης και 3945 εικόνες στο σύνολο δεδομένων δοκιμής.

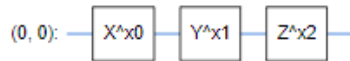
Το κβαντικό κύκλωμα που δημιουργήθηκε για αυτήν την περίπτωση εμφανίζεται παρακάτω



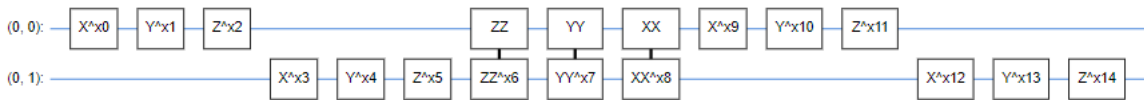
Σχήμα 8.22: Quantum circuit for Quantum CNN

Όπως μπορεί κανείς να δει στο [8.22], εφαρμόσαμε μια πύλη Hadamard σε κάθε qubit ως πρώτο στρώμα πυλών και στη συνέχεια εφαρμόσαμε μια ελεγχόμενη πύλη Z (CZ) μεταξύ κάθε qubit και του προηγούμενου. Τέλος, εφαρμόζουμε μια ελεγχόμενη πύλη Z μεταξύ του πρώτου και του τελευταίου qubit. Το κβαντικό κύκλωμα που εμφανίζεται έχει σχεδιαστεί με τη χρήση του πακέτου Cirq, το οποίο ανήκει στην Google.

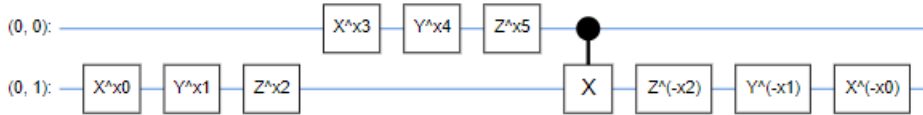
Στη συνέχεια δημιουργώ ένα κύκλωμα που εκτελεί μια περιστροφή της σφαίρας Bloch γύρω από τον άξονα X, Y και Z και αυτό εξαρτάται από τις τιμές στα "σύμβολα". Αυτό το κύκλωμα Cirq θα λειτουργεί ως ενιαίος τελεστής σε ένα μόνο qubit [8.23]. Για να το επεκτείνω, δημιουργώ επίσης ένα κύκλωμα που θα ενεργεί αυθαίρετα σε δύο qubits [8.24] και, τέλος, κατασκευάζω ένα τρίτο κύκλωμα για να κάνω μια παραμετροποιημένη λειτουργία 'ομαδοποίησης', η οποία προσπαθεί να μειώσει την εμπλοκή από δύο qubits σε ένα μόνο [8.25].



Σχήμα 8.23: Single Qubit Unitary Operator



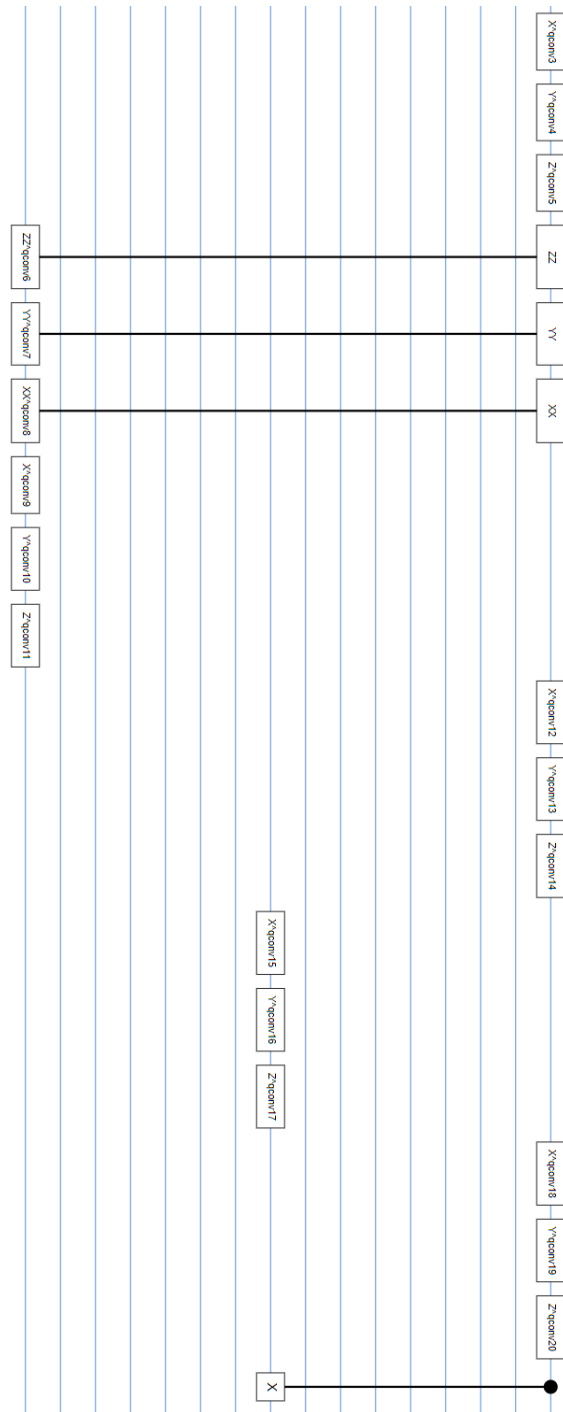
Σχήμα 8.24: Two Qubits Unitary Operator



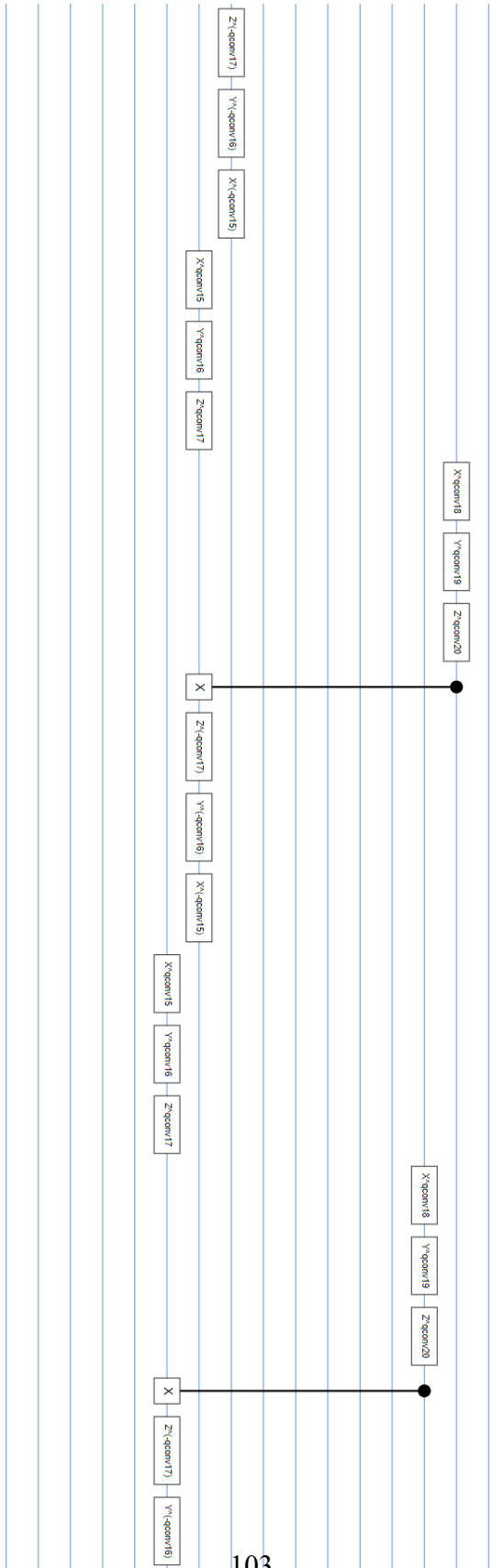
Σχήμα 8.25: Pooling on Two Qubits: Unitary Operator

Όλα τα κυκλώματα έχουν υλοποιηθεί με βάση το ότι χρησιμοποιούμε 1 qubit με βάθος κυκλώματος παραλλαγής. Το τελικό κύκλωμα που εμφανίζεται παρακάτω χρησιμοποιεί 4 qubits για να μπορέσει να εκμεταλλευτεί τις δυνατότητες των χειριστών μας, που ορίζονται παραπάνω στο συγκεκριμένο σύνολο δεδομένων.

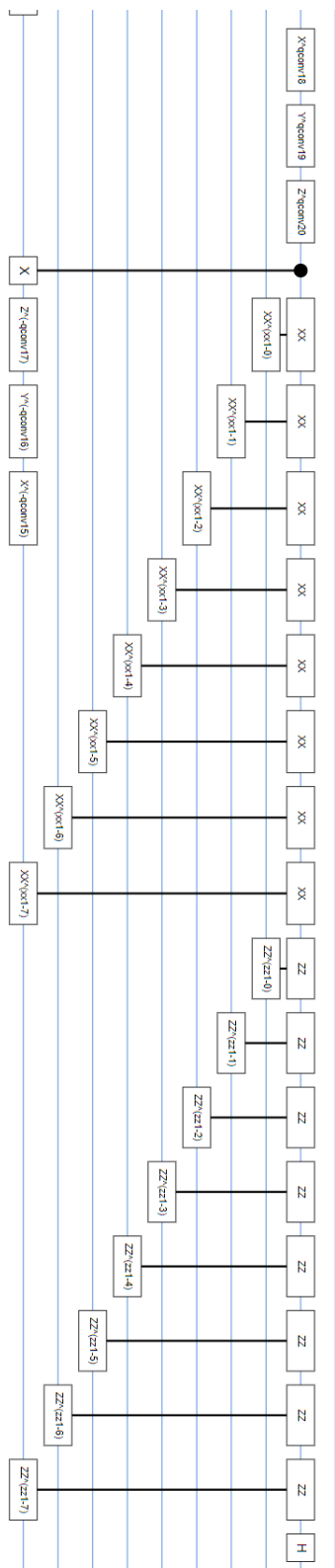
Για να μπορέσω να κατανοήσω τη θεμελιώδη περίπτωση της ταξινόμησης σε αυτό το κεφάλαιο που είναι ακόμα εντελώς άγνωστη σε εμάς, αποφάσισα να παρουσιάσω αποτελέσματα για δυαδική ταξινόμηση. Αυτό συμβαίνει επειδή όλες οι σειρές δεδομένων αν και ανήκουν σε διαφορετικές κλάσεις, μετατρέπονται σε αναπαράσταση bit, προκειμένου να είναι δυνατή η μετάβαση στην αναπαράσταση qubit. Φτάνοντας σε αυτήν την είσοδο, κάνουμε μια δυαδική ταξινόμηση στο επίπεδο των μηδενικών και των μονάδων. Το τελικό κύκλωμα που σχεδιάστηκε για αυτήν την υλοποίηση περιλαμβάνει δύο σχεδόν πανομοιότυπα μπλοκ λειτουργιών πύλης, συμπεριλαμβανομένων συνελκτικών στρωμάτων. Στη συνέχεια συνεχίζουμε με τα ακόλουθα στρώματα εμπλοκής, χρησιμοποιώντας ελεγχόμενες πύλες Z και πύλες Hadamard για να προκαλέσουμε υπέρθεση. Μετά την εφαρμογή των δύο πρώτων μπλοκ κβαντικών πυλών, τα επόμενα στρώματα φαίνονται παρακάτω:



Το παρακάτω μπλοκ εφαρμόζεται δύο φορές σε διαφορετικό επίπεδο του κβαντικού καταχωρητή.







Σχήμα 8.26: Final block of Gates

Το μοντέλο που χρησιμοποιείται έχει την παρακάτω μορφή:

```

Model: "model"
-----
Layer (type)                Output Shape                Param #
-----
input_1 (InputLayer)        [(None,)]                   0
-----
add_circuit (AddCircuit)    (None,)                     0
-----
pqc (PQC)                   (None, 1)                   37
-----
Total params: 37
Trainable params: 37
Non-trainable params: 0

```

Σχήμα 8.27: QCNN Model Summary

Τα αποτελέσματα που λαμβάνουμε από αυτό το υβριδικό κβαντικό-κλασικό μοντέλο εμφανίζονται παρακάτω:

	accuracy	loss	val_accuracy	val_loss
0	0.738281	0.720991	0.769309	0.471293
1	0.765625	0.462365	0.834350	0.429513
2	0.814453	0.455481	0.769309	0.461561
3	0.796875	0.477502	0.822154	0.446415
4	0.775391	0.483203	0.822154	0.438809

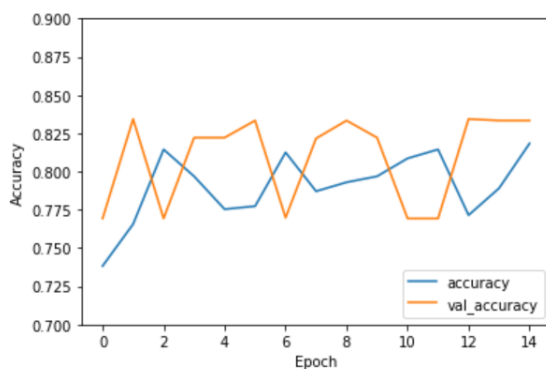
Σχήμα 8.28: Model Results: QCNN

Όπως μπορεί να δει κανείς από τα στατιστικά στοιχεία του προβλήματος, λάβαμε πολύ υψηλή ακρίβεια από την αφετηρία της διαδικασίας. Καταφέραμε να επιτύχουμε μέγιστη ακρίβεια εκπαίδευσης 81.445% και μέγιστη ακρίβεια επικύρωσης 83.435%. Αυτό το υβριδικό κβαντικό-κλασικό μοντέλο έχει τρέξει σε GPU με χρόνο εκτέλεσης που διαρκεί 117,75 λεπτά με μέσο χρόνο epoch 7,85 λεπτά. Για αυτήν την περίπτωση, χρησιμοποιήσαμε το μοντέλο για 15 epochs, αλλά με πολύ μικρότερο μέγεθος παρτίδας από την κλασική περίπτωση. Χρησιμοποιήσαμε μέγεθος παρτίδας ίσο με 16 με ρυθμό εκμάθησης, lr=0,01 σε βελτιστοποιητή Adam. Επιλέξαμε να εκπαιδεύσουμε το σύνολο δεδομένων διαχωρίζοντάς το σε 80% σύνολο δεδομένων εκπαίδευσης και 20% σύνολο δεδομένων δοκιμής και, στη συνέχεια, δειγματοληπτικά ένα σύνολο 500 δειγμάτων.

Το πρώτο συμπέρασμα είναι ότι αυτά τα αποτελέσματα είναι πολύ καλύτερα από την κλασική περίπτωση, οπότε φτάνουμε σε άμεσο κβαντικό πλεονέκτημα. Αυτό οφείλεται στο γεγονός ότι χρησιμοποιήσαμε ένα κβαντικό κύκλωμα με με-

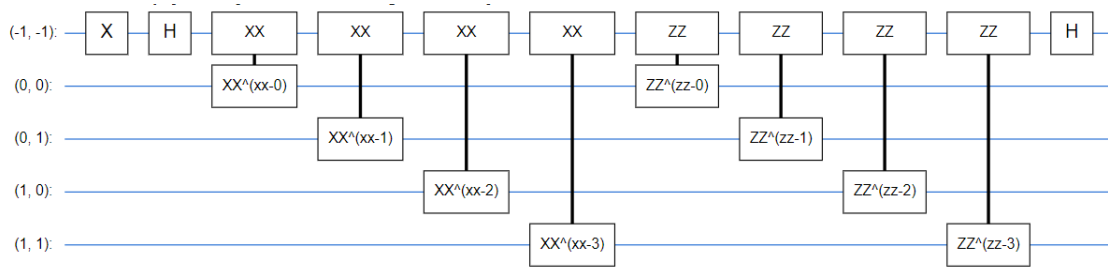
γάλο βάθος. Ως βάθος, εννοούμε τον αριθμό των στρωμάτων που χρησιμοποιούνται στο κύκλωμα με κάθε στρώμα στο κβαντικό κύκλωμα να αντικαθιστά τα κλασικά πολλαπλά στρώματα με απλές κβαντικές πύλες, οι οποίες εκτελούν λειτουργίες όπως διακύμανση πλάτους και περιστροφή των διανυσμάτων εισόδου στη σφαίρα Bloch. Ο λόγος που επέλεξα ένα τόσο "σύνθετο" κύκλωμα με μεγάλο βάθος και διάφορες πύλες, είναι επειδή, όπως παρατηρήθηκε στην απλή περίπτωση των μεθόδων κωδικοποίησης δεδομένων που παρέχουν είσοδο σε γενικούς ταξινομητές κβαντικών μεταβλητών, η απόδοση των κβαντικών μοντέλων μπορεί να βελτιστοποιηθεί αυξάνοντας τον αριθμό των στρώματα.

Όπως μπορεί κανείς να παρατηρήσει στο [8.29] η ακρίβεια μπορεί να μην αυξάνεται, αλλά είναι καλό σημάδι ότι οι ακρίβειες εκπαίδευσης και επικύρωσης ακολουθούν την ίδια συμπεριφορά.

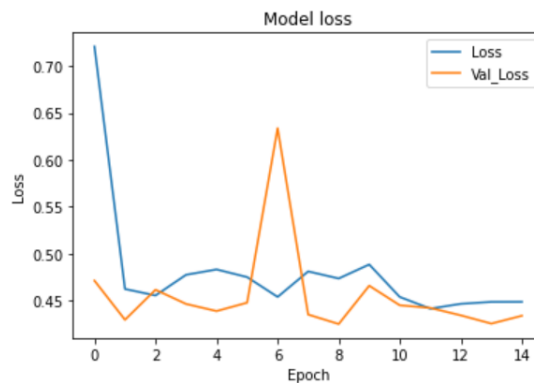


Σχήμα 8.29: Model Accuracy: QCNN

Δεύτερον, όσον αφορά την απώλεια του υβριδικού κβαντικού-κλασικού μοντέλου, παρατηρούμε την αναμενόμενη συμπεριφορά τόσο στη συνάρτηση απώλειας εκπαίδευσης όσο και στη συνάρτηση απώλειας δοκιμής με μόνο μια ακραία τιμή γύρω στις 6 epochs.



Σχήμα 8.31: Quantum Circuit with smaller depth



Σχήμα 8.30: Model Loss: QCNN

Λόγω της απόδοσής του, θέλαμε να δοκιμάσουμε μια πρόσθετη υλοποίηση για την περίπτωση των Κβαντικών Συνελκτικών Νευρωνικών Δικτύων, καθώς είναι μια περιοχή με ποικίλες χρήσεις και αποτελεί τη βάση για να συνεχίσουμε σε πιο σύνθετους τύπους Νευρωνικών Δικτύων. Σε αυτή την περίπτωση, επιλέγω την ίδια στρατηγική όπως στο προηγούμενο μοντέλο του κβαντικού CNN. Χρησιμοποιώ το ίδιο σύνολο δεδομένων με 4 κλάσεις, αλλά κωδικοποιώ δυαδικά τα δεδομένα μου, προκειμένου να τα εμφανίσω στην αναπαράσταση bit. Αφού επιτύχουμε αυτόν τον μετασχηματισμό, μπορούμε να μελετήσουμε το πρόβλημα δυαδικής ταξινόμησης που εμφανίζεται. Για αυτήν την περίπτωση, δοκιμάζουμε ένα διαφορετικό κβαντικό κύκλωμα, σχεδιασμένο με διαφορετικές κβαντικές πύλες και μικρότερο βάθος. Το τρέχον κβαντικό κύκλωμα φαίνεται παρακάτω: και το αντίστοιχο Νευρωνικό Δίκτυο, το οποίο συνδυάζεται με το κβαντικό κύκλωμα για να εκπαιδευσουμε το μοντέλο μας μπορεί να περιγραφεί ως ένα απλό μοντέλο PQC που περιλαμβάνει μόνο 8 παραμέτρους [8.32].

```
Model: "sequential"
```

Layer (type)	Output Shape	Param #
ρqc (PQC)	(None, 1)	8

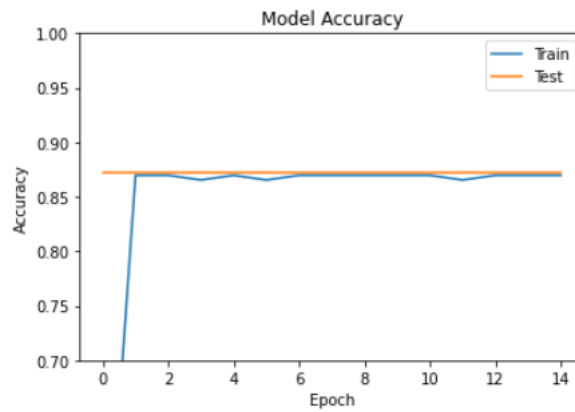
```
Total params: 8
Trainable params: 8
Non-trainable params: 0
```

Σχήμα 8.32: PQC model for QCNN

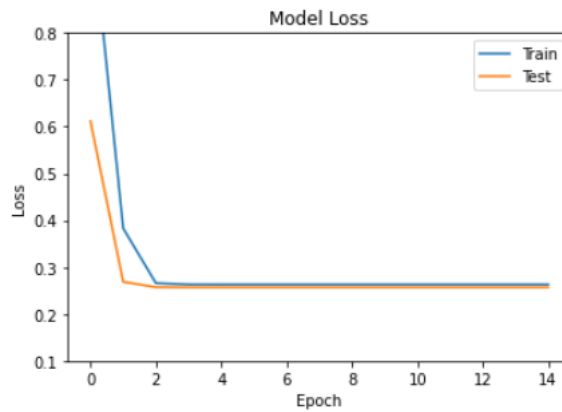
Καταφέρνουμε να φτάσουμε σε υψηλότερη ακρίβεια 1-2% αλλά το μοντέλο μας είναι ακίνητο και δεν συνεχίζει να μαθαίνει, όπως μπορεί να δει κανείς.

```
Epoch 1/15
114/114 [=====] - 8s 72ms/step - loss: 1.0676 - hinge_accuracy: 0.4423 - val_loss: 0.6116 - val_hinge_accuracy: 0.8716
Epoch 2/15
114/114 [=====] - 9s 76ms/step - loss: 0.3829 - hinge_accuracy: 0.8697 - val_loss: 0.2685 - val_hinge_accuracy: 0.8716
Epoch 3/15
114/114 [=====] - 8s 71ms/step - loss: 0.2658 - hinge_accuracy: 0.8697 - val_loss: 0.2572 - val_hinge_accuracy: 0.8716
Epoch 4/15
114/114 [=====] - 9s 77ms/step - loss: 0.2630 - hinge_accuracy: 0.8654 - val_loss: 0.2569 - val_hinge_accuracy: 0.8716
Epoch 5/15
114/114 [=====] - 9s 77ms/step - loss: 0.2629 - hinge_accuracy: 0.8697 - val_loss: 0.2569 - val_hinge_accuracy: 0.8716
Epoch 6/15
114/114 [=====] - 9s 75ms/step - loss: 0.2629 - hinge_accuracy: 0.8654 - val_loss: 0.2569 - val_hinge_accuracy: 0.8716
Epoch 7/15
114/114 [=====] - 8s 74ms/step - loss: 0.2629 - hinge_accuracy: 0.8697 - val_loss: 0.2569 - val_hinge_accuracy: 0.8716
Epoch 8/15
114/114 [=====] - 9s 76ms/step - loss: 0.2629 - hinge_accuracy: 0.8697 - val_loss: 0.2569 - val_hinge_accuracy: 0.8716
Epoch 9/15
114/114 [=====] - 9s 75ms/step - loss: 0.2629 - hinge_accuracy: 0.8697 - val_loss: 0.2569 - val_hinge_accuracy: 0.8716
Epoch 10/15
114/114 [=====] - 8s 73ms/step - loss: 0.2629 - hinge_accuracy: 0.8697 - val_loss: 0.2569 - val_hinge_accuracy: 0.8716
Epoch 11/15
114/114 [=====] - 8s 72ms/step - loss: 0.2629 - hinge_accuracy: 0.8697 - val_loss: 0.2569 - val_hinge_accuracy: 0.8716
Epoch 12/15
114/114 [=====] - 8s 70ms/step - loss: 0.2629 - hinge_accuracy: 0.8654 - val_loss: 0.2569 - val_hinge_accuracy: 0.8716
Epoch 13/15
114/114 [=====] - 9s 82ms/step - loss: 0.2629 - hinge_accuracy: 0.8697 - val_loss: 0.2569 - val_hinge_accuracy: 0.8716
Epoch 14/15
114/114 [=====] - 9s 78ms/step - loss: 0.2629 - hinge_accuracy: 0.8697 - val_loss: 0.2569 - val_hinge_accuracy: 0.8716
Epoch 15/15
114/114 [=====] - 9s 82ms/step - loss: 0.2629 - hinge_accuracy: 0.8697 - val_loss: 0.2569 - val_hinge_accuracy: 0.8716
```

Σχήμα 8.33: Smaller QCNN statistics



Σχήμα 8.34: Smaller QCNN - Model Accuracy



Σχήμα 8.35: Smaller QCNN - Model Loss

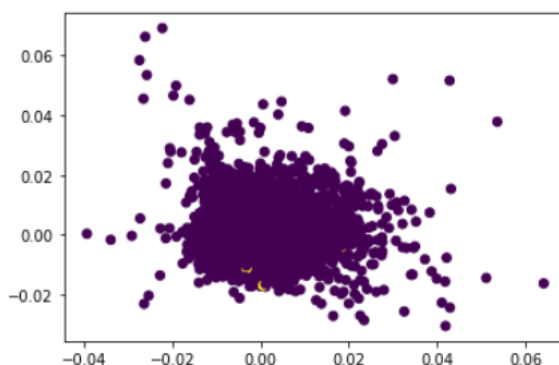
Μετά την αξιολόγηση του μοντέλου μας, λαμβάνουμε μια μέση απώλεια ίση με 0.26355 και μια μέση ακρίβεια ίση με 86.905%, που είναι μια αρκετά καλή βαθμολογία. Ωστόσο, η υψηλή ακρίβεια δεν είναι η πιο μοναδική πτυχή αυτού του μοντέλου. Όπως μπορεί κανείς να παρατηρήσει στο 8.33, το μοντέλο μας έτρεχε κατά μέσο όρο 8 δευτερόλεπτα ανά epoch και με 15 epochs, πήραμε τα αποτελέσματά μας σε μόλις 2.125 λεπτά. Έτσι, σε αυτήν την περίπτωση μπορεί να μην έχουμε αυξήσει την ακρίβεια του μοντέλου σε σύγκριση με την προηγούμενη υλοποίηση, ωστόσο πετύχαμε την κβαντική υπεροχή καθώς το μοντέλο μας έτρεχε στο 0.0185% του χρόνου εκτέλεσης του προηγούμενου μοντέλου.

## 4. Quantum Generative Neural Networks (QGAN)

Για αυτό το μοντέλο φόρτωσα το σύνολο δεδομένων των αντικαταθλιπτικών φαρμάκων, το οποίο περιλαμβάνει 19,5 χιλιάδες σειρές χορδών SMILES. Στη συνέχεια εκμεταλλεύομαι το πακέτο RDKit, μετατρέπω τα SMILES σε μοριακές εικόνες που θα χρησιμοποιηθούν ως είσοδος για το κβαντικό μας GAN. Οι εικόνες χωρίστηκαν σε 4 κατηγορίες, χρησιμοποιώντας τις τιμές των συντελεστών logP ως κριτήριο. Χρησιμοποιώ επίσης ένα Image Data Generator για να βελτιώσω τα δεδομένα εισόδου μου και να αλλάξω την κλίμακα όλων των εικόνων, προκειμένου να κανονικοποιηθούν στο  $[0,1]$  διάστημα. Το μέγεθος στόχου των εικόνων που χρησιμοποιούνται έχει διαστάσεις  $(28, 28)$  και χρησιμοποιώ μέγεθος παρτίδας ίσο με 64 στο generator. Έτσι, παίρνω ένα σύνολο από:

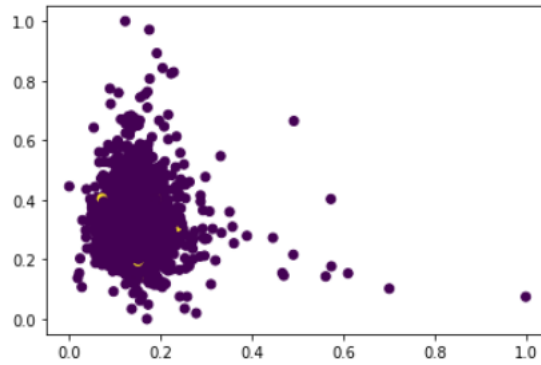
- ▶ train images of dimension:  $(10013, 28, 28, 4)$ ,
- ▶ train labels of dimension:  $(10013,4)$ ,
- ▶ test images of dimension:  $(3339, 28, 28, 4)$  and
- ▶ test labels of dimension:  $(3339,4)$ .

Στη συνέχεια, μειώνω τη διάσταση των πινάκων σε 1 και αλλάζω το μέγεθος των εικόνων σε  $(16,16)$  και αναδιαμορφώνω το σύνολο δεδομένων μου για να εκπαιδεύσω τις εικόνες  $(10000, 256)$ . Για να συνεχίσω, για να μειώσω τις διαστάσεις των προβλημάτων, χρησιμοποιώ τον αλγόριθμο PCA με  $k=4$ . Μετά από αυτό το βήμα, λαμβάνω ως αποτέλεσμα ένα διάνυσμα που περιγράφει τον λόγο διακύμανσης και είναι  $V.R. = [0,02082756, 0,01983622, 0,01914438, 0,01784568]$  και ο αντίστοιχος Σωρευτικός Λόγος Διακύμανσης  $= 0,07765384577214718$ . Στη συνέχεια λαμβάνουμε το διάγραμμα διασποράς στον διανυσματικό χώρο [8.36]:



Σχήμα 8.36: Resulting Vector Space, after PCA algorithm

και το κανονικοποιώ σε  $[0,1]$  εύρος τιμών:



Σχήμα 8.37: Normalized Vector Space, to  $[0,1]$

Αυτή τη στιγμή μετρώ τον αριθμό των πραγματικών δεδομένων και είναι ίσος με 59 και μετά εξάγω τα σχετικά δεδομένα από το σύνολο δεδομένων.

Τώρα, για να συνεχίσω, κατασκευάζω το κβαντικό κύκλωμα χρησιμοποιώντας τη συσκευή "default.qubit" που παρέχεται από το πακέτο pennylane. Κατασκευάζω 3 συναρτήσεις με τις κβαντικές πύλες, μία για δειγματοληψία των παραγόμενων δεδομένων, το generator και το discriminator του μοντέλου GAN. Έτσι, σε αυτή την περίπτωση επιλέγουμε να χρησιμοποιήσουμε ένα πλήρως κβαντικό μοντέλο αντί για ένα υβριδικό. Στη συνέχεια δημιουργώ μια συνάρτηση κβαντικών πυλών για την εφαρμογή του discriminator στα πραγματικά δεδομένα και μια συνάρτηση για την εφαρμογή του discriminator στα δεδομένα που δημιουργούνται. Αυτά ακολουθούνται από μια κβαντική συνάρτηση που υπολογίζει την πιθανότητα να έχουμε πραγματικά αληθινά αποτελέσματα και την πιθανότητα να ληφθούν ψεύτικα αληθινά αποτελέσματα. Τέλος, για να ολοκληρώσουμε την εφαρμογή μας, κατασκευάζω μια συνάρτηση απώλειας για το generator και μια άλλη για τον discriminator. Στο πρώτο παίρνω ως επιχείρημα την πιθανότητα να βγει ένα ψεύτικο αληθινό αποτέλεσμα και στο δεύτερο αφαιρώ τις δύο πιθανότητες:

$$disccost = Pr(fake - true) - Pr(real - true)$$

Οι εξισώσεις αυτών των πιθανοτήτων μπορούν να εκφραστούν ως εξής:

$$P_{fake} = abs\left(\frac{E(SWAPTEST(D, G)) - 0.5}{0.5}\right) \quad (8.2)$$

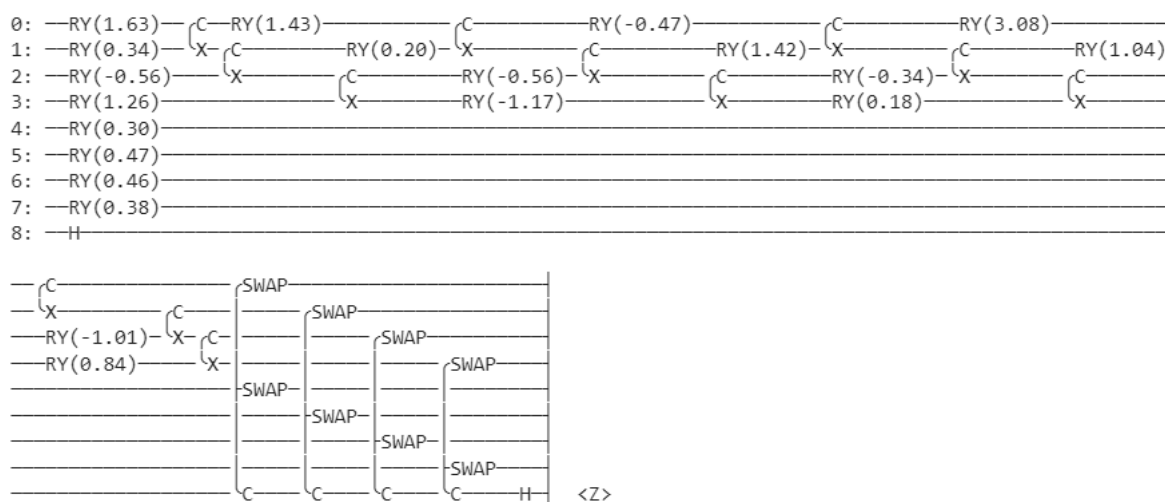
and

$$P_{real} = abs\left(\frac{E(SWAPTEST(D, Real)) - 0.5}{0.5}\right) \quad (8.3)$$



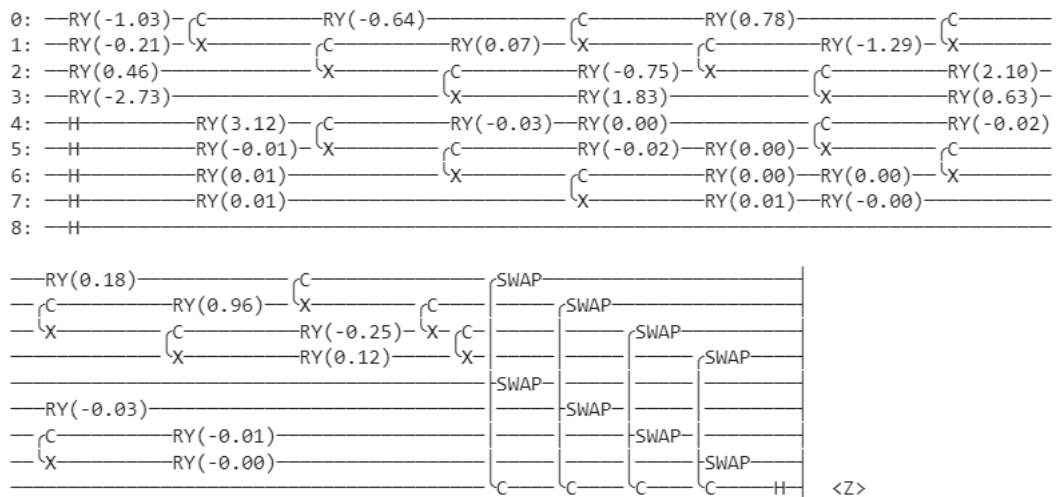
Αυτά μπορούν να χρησιμοποιηθούν για τη μέτρηση της απώλειας του discriminator και το generator και μπορούν να χρησιμοποιηθούν για την ενημέρωση των καθοριστικών παραμέτρων των δικτύων παρόμοια με ένα κλασικό GAN.

Η συνάρτηση πραγματικού discriminator μπορεί να απεικονιστεί με το παρακάτω κύκλωμα:



Σχήμα 8.38: Visualization of the real discriminator function

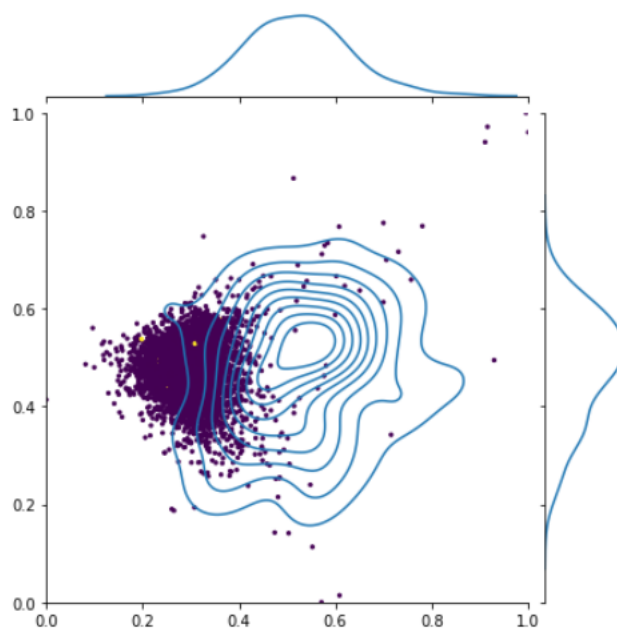
Ο discriminator για τα δεδομένα που παράγονται μπορεί να απεικονιστεί στο παρακάτω κβαντικό κύκλωμα:



Σχήμα 8.39: Visualization of the discriminator function on generated images

Αυτό είναι ένα από τα λίγα μοντέλα, στα οποία δεν χρησιμοποιήσα Adam optimizer και αντ' αυτού χρησιμοποιήσα βελτιστοποιητή gradient descent με ρυθμό εκμάθησης,  $\text{lr}=0,05$ . Τα αποτελέσματα που δίνονται σε κάθε epoch έχουν την παρακάτω μορφή:

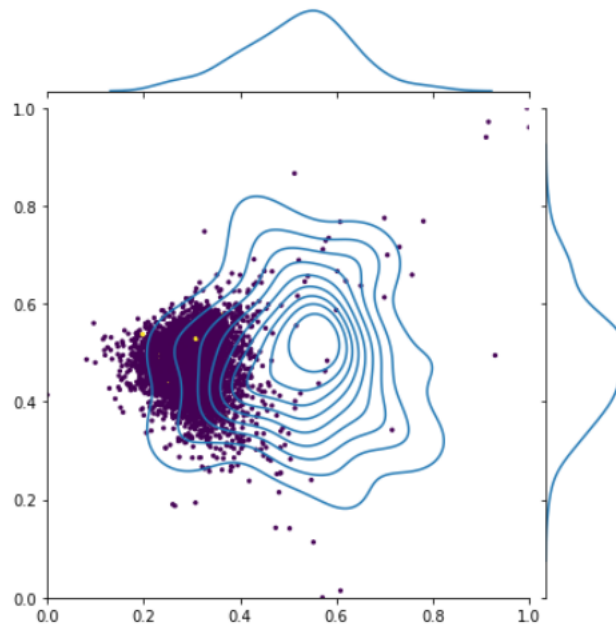
	Step #	Cost
<b>Epoch 0</b>	Step 1	-0.0500985666
	Step 16	-0.0501516775
	Step 31	-0.051600991791
	Step 46	-0.051554220037
Pr(Real = True) = 0.648361598057		
Pr(Gen = True) = 0.6032677431298		
<b>Generator</b>		
<b>Epoch 0</b>	Step 1	-0.6032677
	Step 16	-0.6032677
	Step 31	-0.6032677
	Step 46	-0.6032677
Discriminator cost = -0.05160991791		



Σχήμα 8.40: Vector space after classification

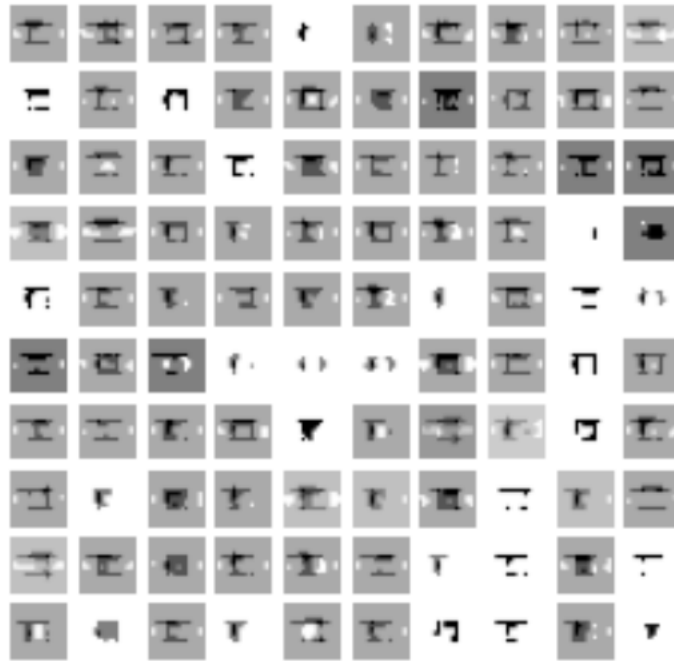
Έτσι, για αυτήν την περίπτωση δείχνουμε μια συνάρτηση κόστους για τον discriminator και αντί για τη μέτρηση του κοινού μοντέλου, που είναι η ακρίβεια, συγκρίνουμε για κάθε epoch τις πιθανότητες του διακριτή για να προβλέψουμε ένα πραγματικό αληθινό αποτέλεσμα στις πραγματικές εικόνες και την πιθανότητα του διακριτικού για να προβλέψει ένα ψεύτικο αληθινό σε μια εικόνα που δημιουργήθηκε. Στο διάγραμμα [8.40], που εμφανίζεται παραπάνω, μπορούμε να δούμε τον κανονικοποιημένο διανυσματικό χώρο και την κατανομή των δεδομένων μετά την προσπάθεια ταξινόμησης τους σε κάθε epoch. Οι γραμμές ορίζουν τα επίπεδα εμπιστευτικότητας για τον υπολογισμό της πιθανότητας ένα σημείο να ανήκει στην επιθυμητή κλάση. Το μοντέλο έτρεξε για 32 epochs σε μια GPU με χρόνο εκτέλεσης 4.582 λεπτά με μέσο όρο 8.591 δευτερόλεπτα για κάθε epoch. Με αυτό ως γεγονός, καταλαβαίνουμε γρήγορα γιατί αυτό το μοντέλο ξεπερνά το κλασικό ανάλογο μοντέλο του. Στο πρώτο μέρος του πίνακα, εμφανίζονται οι τιμές της συνάρτησης κόστους discriminator και στο δεύτερο τμήμα του πίνακα, υπολογίζεται η συνάρτηση κόστους συνάρτησης του generator.

Το τελικό αθροιστικό διάγραμμα του διανυσματικού χώρου που προκύπτει από τον αλγόριθμο είναι το παρακάτω:



Σχήμα 8.41: Cumulative Vector space after training

Και η μέση πιστότητα σκηνής που λαμβάνουμε είναι  $[0, 51914769, 0, 48085231]$ . Παρακάτω, εμφανίζω σε πλέγμα μερικές από τις δημιουργημένες εικόνες από το κβαντικό μας μοντέλο. Είναι θολά και έχουν πολύ θόρυβο. Ωστόσο, αν προσθέσουμε πολλά βήματα, θα γίνουν πιο ξεκάθαρα. Το πρόβλημα της έλλειψης χρόνου δεν μπορούσε να μας αφήσει να φτάσουμε στη σκηνή, στην οποία οι εικόνες θα είχαν πεντακάθαρη εμφάνιση.



Σχήμα 8.42: Grid of Generated Images from our Quantum GAN

Συμπερασματικά, αποφάσισα να χρησιμοποιήσω αυτόν τον τύπο αρχιτεκτονικής για το Quantum GAN που εκτελεί τον discriminator και το generator αποκλειστικά σε κβαντικό υλικό και χρησιμοποιώντας δοκιμές ανταλλαγής σε qubits για τον υπολογισμό της τιμής των συναρτήσεων απώλειας. Όταν χρησιμοποιείται ο ίδιος αριθμός παραμέτρων, με βάση τη βιβλιογραφία, αυτό το είδος μοντέλου ξεπερνά όχι μόνο τα κλασικά GAN, αλλά επίσης ξεπερνά τα άλλα κβαντικά GAN στη βιβλιογραφία έως και 125% όσον αφορά την ομοιότητα μεταξύ των παραγόμενων κατανομών και των αρχικών δεδομένων -σκηνικά. Αυτή η αρχιτεκτονική βασίστηκε στο γενικό σχέδιο QuGAN Η αρχιτεκτονική τροφοδοτεί τα κλασικά μας δεδομένα, τα οποία μεταφράζονται σε κβαντικά δεδομένα. Μετά τον μετασχηματισμό, οι παράμετροι και τα κβαντικά δεδομένα χρησιμοποιούνται για να μάθουν ο κβαντικός discriminator και τον κβαντικό generator, καθώς και τα κβαντικά δεδομένα.

Οι δοκιμές SWAP χρησιμοποιούνται γενικά για τη μέτρηση ομοιοτήτων στον κβαντικό χώρο. Ένα τεστ SWAP είναι ένας κβαντικός αλγόριθμος που μετρά τη διαφορά μεταξύ δύο κβαντικών καταστάσεων. Αυτός ο αλγόριθμος απαιτεί τα δύο qubit, των οποίων την ομοιότητα θα μετρήσουμε και ένα τρίτο qubit  $|\Phi\rangle$  για να βοηθήσει στη μέτρηση. Στο μοντέλο μας, η δοκιμή SWAP χρησιμοποιείται για να μπορέσει να ολοκληρώσει την επικοινωνία μεταξύ των δύο υποκυκλωμάτων, του κβαντικού generator και του κβαντικού discriminator. Ο αριθμός των πλατών που

πρέπει να υπολογιστούν είναι  $2^n$ , (εδώ  $n=4$  qubits) και οι πίνακες που περιγράφουν το σύστημα ή τους μετασχηματισμούς φτάνουν σε μεγέθη  $2^n \times 2^n$ . Λόγω του τελευταίου γεγονότος, η προσομοίωση κβαντικών κυκλωμάτων μπορεί γρήγορα να καταστεί ανέφικτη και μπορεί να γίνει πολύ δαπανηρή για την προσομοίωση. Αυτός είναι και ο λόγος, δεν μπορούμε να παίζουμε με πολύ μεγάλο αριθμό εποχών για να παράγουμε καλύτερες εικόνες.

## Κεφάλαιο 9

# Πειραματικά Αποτελέσματα - Συμπεράσματα

Σε αυτή τη διατριβή, προσπαθήσαμε να δούμε σε ποιο επίπεδο μπορούμε να εφαρμόσουμε κβαντικούς αλγόριθμους για την ανάλυση και την πρόβλεψη μορίων από μια μεγάλη βάση δεδομένων, που περιέχει 193 χιλιάδες σειρές. Ωστόσο, αποφασίσαμε να εστιάσουμε τη μελέτη μας σε ένα υποσύνολο αυτού του συνόλου δεδομένων και να δούμε πώς μπορούμε να ταξινομήσουμε τα μόρια ως συμβατά για χρήση στο σχεδιασμό αντικαταθλιπτικών φαρμάκων. Μια άλλη περίπτωση είναι η ταξινόμηση των αντικαταθλιπτικών φαρμάκων σε 4 κατηγορίες με βάση τον συντελεστή κατανομής οκτανόλης-νερού, συντελεστή  $\log P$ , κάθε μορίου. Χρησιμοποιήσαμε μια στρατηγική ροή προεπεξεργασίας για να χρησιμοποιήσουμε διαφορετικές εισόδους που ταιριάζουν καλύτερα στις ανάγκες κάθε μοντέλου, προκειμένου να επιτύχουμε την καλύτερη απόδοση για κάθε μοντέλο. Χρησιμοποιήσαμε πίνακες, εικόνες, γραφήματα και διανύσματα χαρακτηριστικών ως χώρους εισόδου για τα μοντέλα μας. Για τα κλασικά υπολογιστικά μοντέλα χρησιμοποιήσαμε τις προαναφερθείσες εισόδους για να εφαρμόσουμε τη μελέτη τόσο δυαδικής όσο και πολυκλασικής ταξινόμησης σε ορισμένες περιπτώσεις. Ωστόσο, στα περισσότερα μοντέλα Κβαντικής Μηχανικής Εκμάθησης «αναγκαστήκαμε» να μελετήσουμε την περίπτωση της δυαδικής ταξινόμησης. Αυτό συνέβη λόγω του τρόπου που διαμορφώσαμε τον χώρο εισόδου μας. Κατά τη διάρκεια της ροής προεπεξεργασίας μας, πριν εκπαιδεύσουμε τα μοντέλα μας, μετατρέψαμε τα δεδομένα μας σε αναπαράσταση bit για να μπορέσουμε κατά συνέπεια να μεταφερθούμε στην αναπαράσταση qubit από εκεί. Έτσι, αφού κωδικοποιήσαμε τα δεδομένα μας σε αυτές τις συγκεκριμένες κβαντικές καταστάσεις, προσπαθήσαμε να ταξινομήσουμε τα δεδομένα μας χρησιμοποιώντας αυτήν την αναπαράσταση.

Πριν αναλύσουμε τα υπόλοιπα μοντέλα, είναι σημαντικό να αναφέρουμε ως αποποίηση ευθυνών ότι η διαφορετική περίπτωση και οι διαφορετικοί τύποι μοντέλου δεν προσεγγίστηκαν με τον ίδιο τρόπο, ούτε παρουσιάσαμε τα αποτελέσματα με πανομοιότυπη αναπαράσταση. Αυτό συμβαίνει γιατί θέλαμε να μελετήσουμε πρώτα κάθε περίπτωση ξεχωριστά και μετά να συγκρίνουμε τα αποτελέσματα. Η περίπτωση είναι διαφορετικό είδος προβλήματος και επομένως χρειάζεται διαφορετική στρατηγική για να το προσεγγίσουμε. Καθώς προσπαθούσαμε να καταλάβουμε πώς να κωδικοποιήσουμε τα δεδομένα εισόδου στα κβαντικά μας κυκλώματα, συγκρίναμε δύο τεχνικές κβαντικής ενσωμάτωσης. Και τα δύο είχαν χαμηλή απόδοση για το σύνολο δεδομένων μας και απέτυχαν καθώς έφτασαν τη μέγιστη ακρίβεια 57%. Πιστεύουμε ότι αυτό το πρόβλημα θα μπορούσε να ξεπεραστεί με τις πιο ισχυρές συσκευές από τις συσκευές και μια μέθοδος που θα μπορούσε να μελετηθεί περισσότερο και που θα μπορούσε ενδεχομένως να προσφέρει καλύτερα αποτελέσματα θα μπορούσε να είναι η μέθοδος κωδικοποίησης πυκνής γωνίας για αυτήν την περίπτωση. Αυτό οφείλεται στο γεγονός ότι πιστεύαμε ότι η μέθοδος κωδικοποίησης γωνίας είχε καλύτερη απόδοση από τη μέθοδο κωδικοποίησης πλάτους και επειδή ταιριάζει καλύτερα στο συγκεκριμένο σύνολο δεδομένων. Αυτό οφείλεται στο γεγονός ότι τα δεδομένα μας περιλαμβάνουν γεωμετρικές παραμέτρους και έχουν συμμετρίες που θα μπορούσαν να ταιριάζουν με τη μέθοδο κωδικοποίησης γωνίας.

Χρησιμοποιώντας ως γεγονός την παραπάνω αναπαράσταση bit/qubit βλέπουμε τα αποτελέσματα κάθε μοντέλου και καταλήξαμε σε μια σύγκριση πρώτου επιπέδου των μοντέλων και ορίζοντας το κβαντικό πλεονέκτημα που επιτυγχάνεται σε κάθε περίπτωση. Πρώτον, βλέπω τα αποτελέσματα του Συνελκτικού Νευρωνικού Δικτύου με δύο Κβαντικά Συνελκτικά μοντέλα [9].

	Accuracy	Runtime
CNN	74.25%	113.33 minutes
QCNN 1	83.435%	117.75 minutes
QCNN 2	86.905%	2.125 minutes

Έτσι, μπορούμε να παρατηρήσουμε ότι και για τις δύο περιπτώσεις υπάρχει ένα κβαντικό πλεονέκτημα, είτε όσον αφορά την ακρίβεια του μοντέλου είτε τον χρόνο εκτέλεσης. Βλέπουμε ότι η διαφορά δεν είναι πολύ μεγάλη μεταξύ του CNN και του πρώτου μοντέλου QCNN, περίπου 10

Τώρα όσον αφορά το μοντέλο Graph Convolutional, καταφέραμε να μελετήσουμε μόνο την κλασική υπολογιστική περίπτωση, καθώς το κβαντικό ανάλογό του



βρίσκεται ακόμα σε προκαταρκτική θεωρητική κατάσταση και είναι υπολογιστικά και έγκαιρα πολύ δαπανηρό σε πιο εφαρμοσμένες περιπτώσεις. Ωστόσο, παρατηρήσαμε ότι το Graph Convolutional Network ξεπέρασε τις επιδόσεις του μοντέλου CNN, επιτυγχάνοντας ακρίβεια ίση με 88,352% ταυτόχρονα.

Τέλος, όσον αφορά τα Generative Adversarial Networks, υπάρχουν διάφοροι τρόποι προσέγγισης αυτής της μελέτης. Ωστόσο, επέλεξα να προσεγγίσω το Quantum GAN με δοκιμαστικές πύλες SWAP και ένα πλήρως κβαντικό μοντέλο. Θα μπορούσαμε να χρησιμοποιήσουμε και ένα υβριδικό μοντέλο, αλλά είδαμε ότι εάν κωδικοποιήσουμε τα δεδομένα εισόδου με τον σωστό τρόπο, ένα πλήρως κβαντικό κύκλωμα αποδίδει καλύτερα και εάν έχουμε την υπολογιστική ισχύ, υπερτερεί τόσο της υβριδικής όσο και της κλασικής υπολογιστικής μεθόδου. Βλέπουμε ότι το κβαντικό GAN τρέχει για 32 epochs σε καλό χρόνο εκτέλεσης 4.582 λεπτών. Όταν χρησιμοποιείται ο ίδιος αριθμός παραμέτρων, με βάση τη βιβλιογραφία, αυτό το είδος μοντέλου ξεπερνά όχι μόνο τα κλασικά GAN, αλλά επίσης ξεπερνά τα άλλα κβαντικά GAN στη βιβλιογραφία έως και 125

Έτσι, κατανοούμε ότι οι κβαντικοί υπολογιστές είναι ένα αναδυόμενο πεδίο που έχει πολλά να προσφέρει. Ωστόσο, το υλικό είναι ακόμα ανεπαρκώς ανεπτυγμένο και μπορούμε να εκμεταλλευτούμε εν μέρει το κβαντικό πλεονέκτημα προς το παρόν. Όλες οι θεωρητικές και υπολογιστικές μελέτες, μαζί με τη δική τους, δείχνουν ότι αν καταφέρουμε να κατασκευάσουμε τις κατάλληλες συσκευές, θα ξεκλειδώσουμε ένα μεγάλο δυναμικό με τον Quantum Computing. Επομένως, αυτή είναι μια ενθαρρυντική σκέψη να συνεχίσουμε την έρευνα σε αυτό το πεδίο για ένα ταξίδι στη μείωση της πολυπλοκότητας των προβλημάτων μας.

# Κεφάλαιο 10

## Βιβλιογραφία

1. "Convolutional Networks on Graphs for Learning Molecular Fingerprints", D. Duvenaud, D. Maclaurin, J.A.I. Bombarelli, T. Hirzel, A.A Guzik, R.P. Adams, 2015,
2. H.L. Molgan. "The generation of a unique machine description for chemical structure. *Journal of Chemistry Documentation*, 5(2):107-113, 1965,
3. RDKit: Open-Source cheminformatics. [www.rdkit.org](http://www.rdkit.org) [11 April 2013],
4. D.Weininger, SMILES, a chemical language and information system. *Journal of chemical information and computer sciences*, 28(1):31-36,1988,
5. Thomas Unterthiner, Andreas Mayr, G "unter Klambauer, Marvin Steijaert, J"org Wenger, Hugo Ceulemans, and Sepp Hochreiter. "Deep learning as an opportunity in virtual screening". *Advances in Neural Information Processing Systems*, 2014,
6. "Semi-Supervised Classification with Graph Convolutional Networks.", T.N. Kipf, M.Welling, published as a conference paper at ICLR 2017,
7. "Quantum autoencoders for efficient compression of quantum data", J.Romero, J.P.Olson, A.A. Guzik., (2017),
8. "Quantum Variational Autoencoder", A.Khoshaman, W. Vinci, B.Denis, E. Andriyash, H. Sadeghi, M.H. Amin,
9. "Quantum Convolutional Neural Networks: Powering Image Recognition with Quantum Circuits, M. Henderson, S. Shakya, S.Pradhan, T.Cook, (2019),
10. Quantum Graph Neural Networks, G.Verdon, T. McCourt, (2019),

11. "Generative chemistry: drug discovery with deep learning generative models", Y.Bian, X. Xie,
12. "Quantum Generative Models for Small Molecule Drug Discovery", J. Li, R. Topaloglou, S. Ghosh, (2021),
13. "Invited: Drug Discovery Approaches using Quantum Machine Learning", J. Li, M. Alam, C. M. Sha, J. W. N. V. Dokholyan, S. Ghosh, (2021),
14. "Robust data encodings for quantum classifiers", R. LaRose, B. Coyle, (2020),
15. "Quantum Machine Learning Algorithms for Drug Discovery Applications", K. Batra, K.Z.Zorn, D.H. Foil, E. Minerali, V.O. Gawriljuk, T.R. Lane, S. Ekins, *J. Chem.Inf.Model.* 2021, 61, 2641-2647,
16. "Convolution filter embedded quantum gate autoencoder", K.Shiba, K. Sakatomo, K. Yamaguchi, D.B. Mall, T. Sogabe,
17. G.E. Hinton and R.R. Salakhutdinov,
18. "Reducing the Dimensionality of Data with Neural Networks, G.E.Hinton, R.R. Salskhutdinov, *Science* 313 (5786), pp.504-507, 2006,
19. "A Comprehensive Survey on Graph Neural Networks, Z. Wu, S.Pan, F. Chen, G. Long, C. Zhang, P.S. Yu,
20. "Machine Learning Prediction of Nine Molecular Properties Based on the SMILES Representation of the QM9 Quantum-Chemistry Dataset", G. A. Pinheiro, J. Mucelini, M.D. Soares, R.C. Prati, J. L. F. Da Silva, M. G. Quiles, *J.Phys.Chem. A* 2020, 124,47,9854-9866,
21. Yann LeCun et al., 1998, Gradient-Based Learning Applied to Document Recognition
22. Adit Deshpande, 2016, The 9 Deep Learning Papers You Need To Know About (Understanding CNNs Part 3)
23. C.-C. Jay Kuo, 2016, Understanding Convolutional Neural Networks with A Mathematical Model,
24. Qiskit documentation by IBM, <https://qiskit.org/documentation/>,
25. "Quantum Graph Convolutional Neural Networks", J. Zheng, Q. Gao, Q. Lv,
26. "Encoding classical data into a quantum computer", K.J.B. Ghosh, (2021),

27. "Hybrid quantum classical graph neural networks for particle track reconstruction", C.Tuysuz, C. Rieger, K. Novotny, B. Demirkoz, D. Dobos, K. Potamianos, S. Vallecorsa, J.-R. Vlimant, R. Forster, *Quantum Machine Intelligence* 3, 29 (2021),
28. "A Gentle Introduction to Graph Neural Networks", Sanchez-Lengeling, B. Reif, E. Pearce, A. Wilteschlko,
29. "Neural Message Passing for Quantum Chemistry", Gilmer J., Schoenholz S.S., 2017, *Proceedings of the 34th International Conference on Machine Learning*, vol.48, pp.1263-1272,
30. "Learning Convolutional Neural Networks for Graphs", Niepert M., Ahmed M., S.S. Riley, P.F. Vinalys, 2017, *Proceedings of the 33rd International Conference on International Conference on Machine Learning*, Vol.48, pp.2014-2023,
31. "How Powerful are Graph Neural Networks?", Xu K., Hu W., Leskovec J., 2017, *Advances in Neural Information Processing Systems*, Vol 30, pp.1024-1034, Curran Associates,
32. <https://quantumalgorithms.org/chap-classical-data-quantum-computers.html>
33. <https://medium.com/mllearning-ai/quantum-data-and-its-embeddings-1-3b022b2f1245>
34. "Dataset's chemical diversity limits the generalizability of machine learning predictions", M.Glavatskikh, J. Leguy, G.Hunault, T. Cauchy, B. Da Mota, *Journal of Cheminformatics* 11, 69, 2019
35. QM9 documentation,
36. Supporting Information for "Physically inspired deep learning of molecular excitations and photoemission spectra", J. Westermayr, R.J. Maurer, *The Royal Society of Chemistry*, 2021,
37. "Early phase drug discovery: Cheminformatics and computational techniques in identifying lead series", C. Bryan, Z. Lei, H. Decornez, D.B. Kitchen, *Bioorganic Medicinal Chemistry*, vol.20, 18,2012, 5324-5342,
38. "Machine Learning in cheminformatics and drug discovery", Y.C. Lo, S.E. Rensi, R.B. Altman
39. <https://pubchem.ncbi.nlm.nih.gov/>,
40. "Autoencoding Undirected Molecular Graphs with Neural Networks", J.J. W. Olsen, P.E. Christensen, M.H. Hansen, A.R. Johansen, (2020),

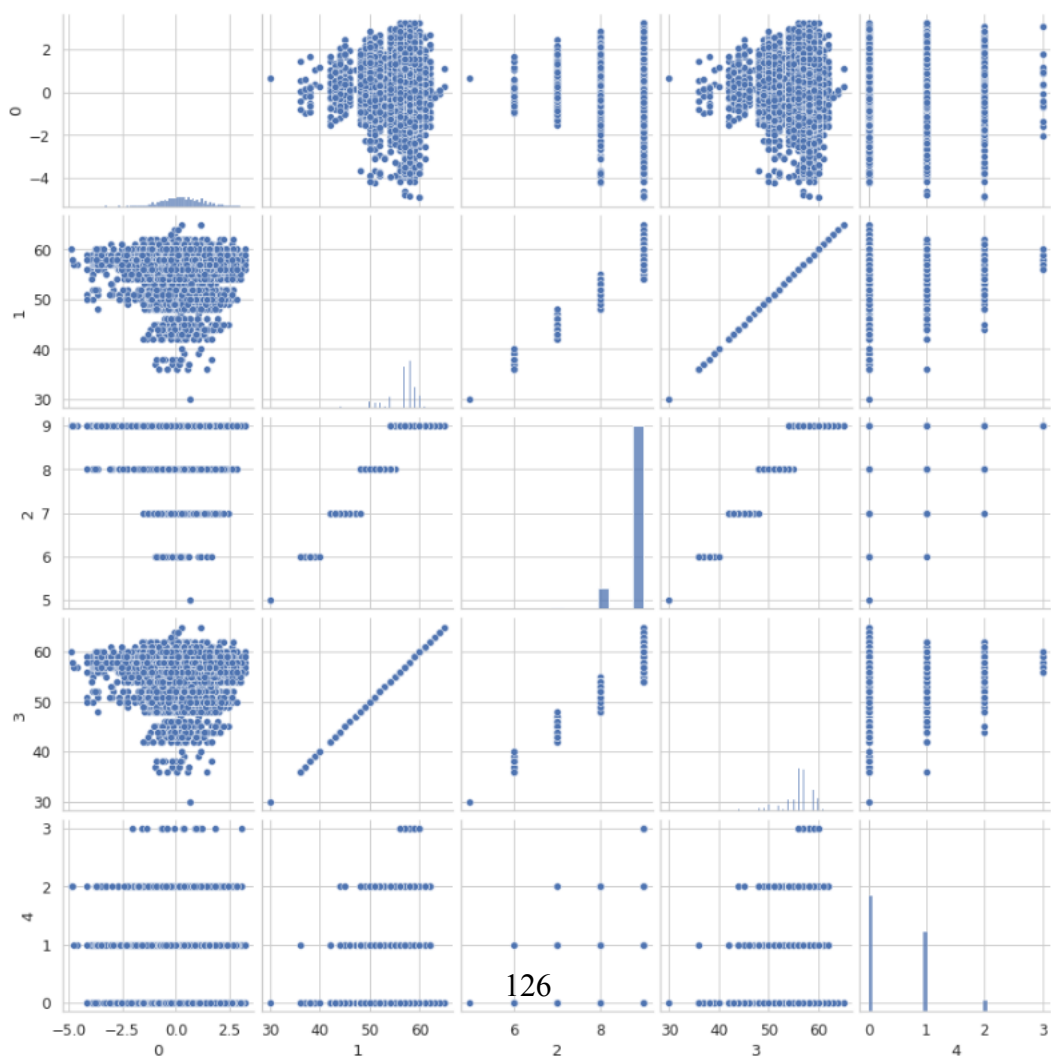
41. Circuit-centric quantum classifiers, M. Schuld, A. Bocharov, K. Svore, N. Wiebe,
42. "Classification with Quantum Neural Networks on Near Term Processors", E. Fahri, H. Neven.



# Κεφάλαιο 11

## Παράρτημα

### 1. Dataset Pairplot before Encoding



Σχήμα 11.1: Dataset from the chapter with the Encoding techniques

## 2. QCNN - Quantum Circuit

The block used twice in the quantum circuit that is described in the section of the Quantum Convolutional NN, can be find in the below 6 images.



