Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Χημικών Μηχανικών
Τομέας ΙΙ: Ανάλυσης, Σχεδιασμού και Ανάπτυξης Διεργασιών και Συστημάτων
Μονάδα Αυτόματης Ρύθμισης και Πληροφορικής

Διπλωματική Εργασία

Ανάπτυξη μοντέλων πρόβλεψης ιδιοτήτων μοριακών δομών με αρχιτεκτονικές νευρωνικών δικτύων γραφημάτων

Ιωάννης Σάββας

Επιβλέπων καθηγητής:
Χαράλαμπος Σαρίμβεης

Ιούνιος 2023

National Technical University of Athens
School of Chemical Engineering
Section II: Process Analysis and Plant Design
Unit of Process Control and Informatics

Diploma Thesis

Development of graph neural networks models for molecular property prediction

Ioannis Savvas

Supervisor Professor:
Haralambos Sarimveis

June 2023

Περίληψη

Στην παρούσα διπλωματική εργασία εξετάζεται η χρήση της βαθιάς μάθησης για την ανάπτυξη μοντέλων πρόβλεψης μοριακών ιδιοτήτων. Πιο συγκεκριμένα, διερευνάται η χρήση πέντε διαφορετικών αρχιτεκτονικών νευρωνικών δικτύων γραφημάτων (Graph Neural Network, GNN) για την πρόβλεψη της διαλυτότητας, πολικότητας και της συνθετικής προσβασιμότητας. Τα μοντέλα που αναπτύσσονται δέχονται ως είσοδο χημικά γραφήματα στα οποία οι κορυφές αποτελούν τα άτομα και οι ακμές τους χημικούς δεσμούς. Οι αρχιτεκτονικές αυτές είναι το συνελικτικό νευρωνικό δίκτυο γραφήματος (Graph Convolutional Network, GCN), το νευρωνικό δίκτυο γραφήματος συγκέντρωσης δείγματος (Graph Sample and Aggregate, GraphSAGE) και το νευρωνικό δίκτυο γραφήματος προσοχής (Graph Attention Network, GAT) με την παραλλαγή του (GATv2). Στο δίκτυο GraphSAGE γίνεται χρήση τελεστή μεγίστου και μέσου όρου ως συνάρτηση συγκέντρωσης. Το σύνολο των δεδομένων αποτελείται από 15.000 μικρά οργανικά μόρια τα οποία πάρθηκαν από την χημική βάση δεδομένων ZINC και περιέχουν τις επιθυμητές ιδιότητες υπολογισμένες. Τα μοντέλα εκπαιδεύονται, βελτιστοποιούνται και επικυρώνονται με τη χρήση της γλώσσας προγραμματισμού Python σε περιβάλλον Jupyter Notebook. Κατόπιν εκπαίδευσης αξιολογούνται στα δεδομένα εξέτασης (test data) ως προς το συντελεστή συσχέτισης (R^2) και το μέσο τετραγωνικό σφάλμα (MSE). Στο τέλος πραγματοποιείται μια σύγκριση ως προς τις καλύτερες και στιβαρότερες αρχιτεκτονικές και προκύπτουν ενδιαφέροντα συμπεράσματα. Τα αποτελέσματα δείχνουν πως οι πιο καινούριες αρχιτεκτονικές (GraphSAGE, GAT, GATv2) κάνουν καλύτερες προβλέψεις από την κλασσική αρχιτεκτονική GCN. Συγκεκριμένα, ως προς την πρόβλεψη της διαλυτότητας και της συνθετικής προσβασιμότητας, τα μοντέλα προσοχής επιτυγχάνουν καλύτερες προβλέψεις ενώ ως προς την πολικότητα τα μοντέλα GraphSAGE φαίνεται να έχουν υψηλότερη ακρίβεια.

Abstract

In the current diploma thesis, the focus is on the use of deep learning for creating models to predict molecular properties. Specifically, the models are based on Graph Neural Networks (GNN) and five different architectures are examined to predict solubility, polarity and synthetic accessibility. The developed models take as input chemical graphs, on which the nodes are the chemical atom symbols and the edges are the chemical bonds. The architectures which were used are the Graph Convolutional Network (GCN), the Graph Sample and Aggregate using maximum and mean as aggregate functions (GraphSAGE) and two Graph Attention Networks (GAT, GATv2). The data used is comprised of 15000 small organic molecules that were obtained from the chemical database ZINC and contain the desirable properties calculated. The models are trained and validated using Python programming language in Jupyter Notebook environment. After training, the models are evaluated on the test data using the Coefficient of determination (R^2) and Mean Squared Error (MSE). At the end of model development, a comparison is made in order to see which architectures do better on each of the three properties predicted. The results show that new architectures such as GraphSAGE, GAT and GATv2 are better than GCN in every property that was examined. More specifically, when it comes to predicting solubility and synthetic accessibility, Graph Attention models produce the best results. On the other hand, polarity is better predicted with the use of GraphSAGE.

Πρόλογος και Ευχαριστίες

Η παρούσα διπλωματική εργασία με τίτλο ‘Ανάπτυξη μοντέλων πρόβλεψης ιδιοτήτων μοριακών δομών με αρχιτεκτονικές νευρωνικών δικτύων γραφημάτων’ σηματοδοτεί και την ολοκλήρωση των προπτυχιακών σπουδών μου στη σχολή Χημικών Μηχανικών του Εθνικού Μετσόβιου Πολυτεχνείου. Καθώς η διπλωματική εργασία πλησιάζει προς το τέλος της, θα ήθελα να ευχαριστήσω τους σημαντικότερους ανθρώπους που συνέβαλαν στην εκπόνηση της.

Αρχικά, θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή της διπλωματικής μου εργασίας Χαράλαμπο Σαρίμβεη, ο οποίος μου έδωσε την ευκαιρία να ασχοληθώ με ένα τόσο ενδιαφέρον θέμα. Επίσης τον ευχαριστώ για την καθοδήγηση, υποστήριξη και επίβλεψη της εργασίας καθ’ όλη τη διάρκεια εκπόνησης της. Στη συνέχεια, θα ήθελα να ευχαριστήσω το μέλος του εργαστηρίου και υποψήφιο διδάκτορα Ιάσωνα Σωτηρόπουλο για τις συμβουλές και την υποστήριξη του, κυρίως στα πρώτα και σημαντικότερα βήματα της διπλωματικής αυτής εργασίας.

Επίσης, θέλω να ευχαριστήσω όλους τους κοντινούς φίλους, την αδερφή μου και την κοπέλα μου για την υποστήριξη στο τελευταίο διάστημα που αποτελούσε και τη συγγραφή της διπλωματικής εργασίας.

Τέλος, το μεγαλύτερο ευχαριστώ το οφείλω στους γονείς μου, Αναστάσιο και Ελένη για την ευκαιρία και την υποστήριξη τους τα τελευταία 5 χρόνια των σπουδών μου στην Αθήνα αλλά και στον παππού μου και την γιαγιά μου.

Περιεχόμενα

Κατάλογος Εικόνων

Κατάλογος Πινάκων

Εισαγωγή

Από την προηγούμενη δεκαετία μέχρι και σήμερα, η τεχνητή νοημοσύνη αποτελεί αναμφισβήτητα έναν από τους πιο πολυσυζητημένους και επιδραστικούς κλάδους της επιστήμης. Με τη χρήση του διαδικτύου, υπάρχει πλέον η διαθεσιμότητα τεράστιων ποσοτήτων δεδομένων τα οποία επιτρέπουν στα συστήματα τεχνητής νοημοσύνης να αντλούν πληροφορίες, να αναλύουν τα δεδομένα και να κάνουν προβλέψεις. Συνάμα, οι αλγόριθμοι μηχανικής μάθησης βελτιώνονται ολοένα και περισσότερο με την βαθιά μάθηση να βρίσκεται στο επίκεντρο των τελευταίων εξελίξεων. Η χρήση της βαθιάς μάθησης έχει οδηγήσει σε σημαντικές ανακαλύψεις στην αναγνώριση εικόνας (αυτό-οδηγούμενα αυτοκίνητα), στην επεξεργασία φυσικής γλώσσας (Generative Pre-trained Transformers, GPT) και σε αρκετούς ακόμα τομείς. Συνδυάζοντας τον τεράστιο όγκο δεδομένων και τους καινούργιους αλγορίθμους με την υπολογιστική δύναμη του σύγχρονου κεντρικού επεξεργαστή (CPU), της κάρτας γραφικών (GPU) και του επεξεργαστή τανυστών (TPU), η μηχανική μάθηση μπορεί να χρησιμοποιηθεί και στον κλάδο της ανακάλυψης φαρμάκων (drug discovery).

Μια επιστημονική περιοχή στην οποία η εφαρμογή μεθόδων μηχανικής μάθησης αυξάνεται ραγδαία είναι αυτή της ανακάλυψης νέων φαρμάκων. Η κλασσική προσέγγιση για την ανακάλυψη ενός νέου φαρμάκου είναι μια πολύ χρονοβόρα και δαπανηρή διαδικασία, αφού τα προκλινικά στάδια μπορούν να διαρκέσουν έως και 6 χρόνια με το κόστος να κυμαίνεται από εκατομμύρια μέχρι και δισεκατομμύριά δολάρια. Η τεχνητή νοημοσύνη είναι πλέον σε θέση να αυξήσει την ταχύτητα και να μειώσει το κόστος αυτής της διαδικασίας. Συγκεκριμένα, με τη χρήση συστημάτων μηχανικής μάθησης πραγματοποιούνται προβλέψεις μοριακών ιδιοτήτων σε υποψήφιες φαρμακευτικές ουσίες μειώνοντας έτσι τις απαραίτητες πειραματικές δοκιμές υποψήφιων φαρμάκων.

Στην παρούσα διπλωματική εργασία μελετάται η εφαρμογή αλγόριθμων βαθιάς μάθησης για την ανάπτυξη μοντέλων πρόβλεψης ιδιοτήτων μορίων που σχετίζονται με τη χρήση τους ως φαρμακευτικές ουσίες. Συγκεκριμένα μελετιούνται οι παρακάτω ιδιότητες: διαλυτότητα, πολικότητα και συνθετική προσβασιμότητα.

Οι ιδιότητες των χημικών μορίων εξαρτώνται από τα δομικά χαρακτηριστικά τους. Για αυτό και είναι απαραίτητο να οριστούν με κατάλληλο τρόπο μοριακοί δείκτες που θα συνδέουν τη δομή με τη μοριακή ιδιότητα. Στην παρούσα εργασία για τη μαθηματική αναπαράσταση της δομής των μορίων χρησιμοποιήθηκαν μοριακά γραφήματα. Οι κορυφές του γραφήματος αποτελούν τα χημικά άτομα ενώ οι ακμές τους χημικούς δεσμούς. Στην πραγματικότητα είναι μια μη ευκλείδεια αναπαράσταση, η οποία μπορεί να περιέχει σημαντικές πληροφορίες-δείκτες για κάθε άτομο και κάθε δεσμό στο μόριο ξεχωριστά.

Οι αλγόριθμοι βαθιάς μηχανικής μάθησης που είναι σε θέση να επεξεργάζονται χημικά γραφήματα είναι τα νευρωνικά δίκτυα γραφημάτων (Graph Neural Networks, GNN). Πιο συγκεκριμένα, οι αλγόριθμοι αυτοί δέχονται ως είσοδο μοριακά γραφήματα και μέσω των κρυφών στρωμάτων που χρησιμοποιούνται, ανανεώνουν συνεχώς με υπολογισμούς τις αναπαραστάσεις των κορυφών. Οι αρχιτεκτονικές των αλγορίθμων αυτών που χρησιμοποιήθηκαν στην παρούσα διπλωματική εργασία είναι το GCN, GraphSAGE, GAT και GATv2.

Στην αρχή της παρούσας διπλωματικής εργασίας δίνεται έμφαση στη γενική θεωρητική βάση των νευρωνικών δικτύων για την κατανόηση βασικών μαθηματικών εννοιών από τις οποίες απαρτίζονται. Στη συνέχεια παρουσιάζονται οι θεωρητικές έννοιες των γραφημάτων αλλά και οι αρχιτεκτονικές των νευρωνικών δικτύων γραφημάτων που χρησιμοποιούνται στην εργασία. Έπειτα γίνεται μια αναφορά στη συμβολοσειρά SMILES και στο σύνολο των δεδομένων που χρησιμοποιείται. Τέλος, παρουσιάζεται η ανάπτυξη των μοντέλων με βάση τις αρχιτεκτονικές και πραγματοποιείται η αξιολόγηση τους ως προς συγκεκριμένες μετρικές.

Κεφάλαιο 1: Μηχανική Μάθηση

Στη σύγχρονη εποχή της συνεχούς τεχνολογικής-επιστημονικής εξέλιξης αλλά και του τεράστιου όγκου διαθέσιμων πληροφοριών, η μηχανική μάθηση (machine learning, ML) αποτελεί έναν από τους πιο δυναμικούς κλάδους. Θα ήταν χρήσιμο αρχικά να γίνει μια αναφορά στις έννοιες της μηχανικής μάθησης, της τεχνητής νοημοσύνης (artificial intelligence, AI) και της βαθιάς μάθησης (deep learning, DL). Η μηχανική μάθηση μπορεί να οριστεί ως η μέθοδος με την οποία μια υπολογιστική μηχανή προσομοιώνει την ανθρώπινη συμπεριφορά και βασίζεται στην «εκπαίδευση» δεδομένων με τη χρήση αλγορίθμων «μάθησης». Τα τελευταία χρόνια, η πρόοδος της μηχανικής μάθησης οφείλεται στην ανάπτυξη βαθιών τεχνητών νευρωνικών δικτύων. Η επιστήμη που μελετά αυτά τα δίκτυα αποτελεί ένα υποσύνολο της μηχανικής μάθησης, την βαθιά μάθηση. Η νέα αυτή επιστήμη έχει καταφέρει σε μεγάλο βαθμό τόσο να προσομοιώσει τις ανθρώπινες δυνατότητες όσο και να τις ξεπεράσει. Τομείς όπως η κυβερνοασφάλεια, η επεξεργασία της φυσικής γλώσσας, η βιοπληροφορική, η υπολογιστική όραση και η ιατρική ανάλυση της πληροφορίας έχουν αναπτυχθεί ραγδαία λόγω της βαθιάς μάθησης. Η μηχανική μάθηση και κατά συνέπεια η βαθιά μάθηση εντάσσονται σε ένα ευρύτερο πλαίσιο, αυτό της τεχνητής νοημοσύνης. Εάν θεωρήσει κανείς πως η τεχνητή νοημοσύνη αποτελεί έναν εγκέφαλο, η μηχανική μάθηση είναι η διαδικασία με την οποία μπορεί αυτός να εκπαιδευτεί και η βαθιά μάθηση είναι η πιο αποδοτική μέθοδος που γνωρίζουμε μέχρι σήμερα [1],[2].

Εικόνα 1.1: Διαγραμματική απεικόνιση συσχέτισης Τεχνητής Νοημοσύνης, Μηχανικής Μάθησης και Βαθιάς Μάθησης

1.1 Κατηγορίες Αλγορίθμων Μηχανικής Μάθησης

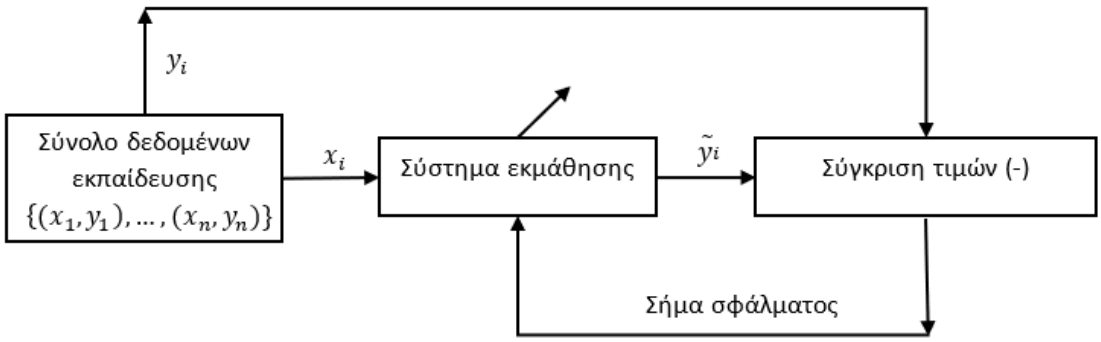
Ένας σύγχρονος ορισμός της μηχανικής μάθησης προήλθε από τον Tom Mitchell. Σύμφωνα με εκείνον, ένα υπολογιστικό πρόγραμμα μπορεί να «μάθει» από μια εμπειρία E σε συνάρτηση με μια κατηγορία εργασιών T και ένα κριτήριο αποδοτικότητας P, όταν η απόδοση του στις εργασίες T, μετρημένη με το κριτήριο P, βελτιώνεται μέσω της εμπειρίας E [3]. Οι αλγόριθμοι μηχανικής μάθησης προσαρμόζουν μοντέλα σε ένα σύνολο δεδομένων, αναζητώντας μοτίβα και σχέσεις που τα χαρακτηρίζουν με τελικό στόχο την εξαγωγή χρήσιμης πληροφορίας.

Υπάρχουν τέσσερα βασικά είδη αλγορίθμων μηχανικής μάθησης, κάθε ένα από τα οποία μπορεί να χρησιμοποιηθεί για την επίλυση διαφορετικού τύπου προβλήματος. Αυτές οι κατηγορίες είναι η επιβλεπόμενη μάθηση (supervised learning), η μη-επιβλεπόμενη μάθηση (unsupervised learning), η ημί-επιβλεπόμενη μάθηση (semi-supervised learning) και η ενισχυτική μάθηση (reinforcement learning).

1.1.1 Επιβλεπόμενη μάθηση

Η επιβλεπόμενη μάθηση περιλαμβάνει τους αλγόριθμους της μηχανικής μάθησης που «μαθαίνουν» τις σχέσεις που διέπουν ένα σύνολο ζευγαριών μεταβλητών εισόδου (X) με τις μεταβλητές εξόδου τους (y). Τα ζευγάρια αυτά ονομάζονται επισημασμένα δεδομένα. Ο στόχος αυτών των μοντέλων είναι να δημιουργήσουν ένα σύστημα το οποίο μπορεί να κάνει μια συσχέτιση των μεταβλητών X και y και να είναι σε θέση να προβλέπει αποκρίσεις σε καινούρια «ανεκπαίδευτα» δεδομένα εισόδου X. Η συσχέτιση αυτή καθορίζεται από ένα σύνολο παραμέτρων μάθησης του μοντέλου. Αυτές οι παράμετροι προκύπτουν από μια συνεχή προσεγγιστική διαδικασία που εκτελεί το μοντέλο επιβλεπόμενης μάθησης για να τους αποκτήσει. Τα μοντέλα αυτά χωρίζονται σε αυτά μπορούν να κάνουν πρόβλεψη μιας κατηγορίας-κλάσης (classification) και εκείνα που προβλέπουν μιας συνεχή τιμή (regression).

Κατά τη διάρκεια της εκπαίδευσης μοντέλων επιβλεπόμενης μάθησης, τροφοδοτούνται στον αλγόριθμο μάθησης δεδομένα εισόδου x_i (συνήθως σε μορφή διανύσματος ή πίνακα) και μέσω της εκπαίδευσης προκύπτει μια έξοδος (πρόβλεψη) $\tilde{y}_i$. Κατόπιν αυτή η τιμή συγκρίνεται με την πραγματική τιμή εξόδου y_i και υπολογίζεται η διαφορά τους (σφάλμα), η οποία εισέρχεται στο σύστημα μάθησης προσαρμόζοντας τις παραμέτρους του μοντέλου [4]. Οι αλγόριθμοι επιβλεπόμενης μάθησης για regression αφορούν και την παρούσα διπλωματική εργασία.



Εικόνα 1.2: Ενδεικτικό διάγραμμα ροής εκπαίδευσης μοντέλων επιβλεπόμενης μάθησης [4]

1.1.2 Μη επιβλεπόμενη μάθηση

Σε πολλές περιπτώσεις τα δεδομένα είναι μη επισημασμένα, δηλαδή περιέχουν μεταβλητές εισόδου χωρίς τις αντίστοιχες αποκρίσεις τους. Οι αλγόριθμοι μη επιβλεπόμενης μάθησης έρχονται να λύσουν αυτό ακριβώς το πρόβλημα. Τα μοντέλα αυτά μαθαίνουν αναπαραστάσεις και μοτίβα στα δεδομένα εισόδου. Διακρίνονται σε μοντέλα ομαδοποίησης (clustering), κανόνων συσχέτισης (association rules) και μείωσης διαστάσεων (Dimensionality reduction). Η πρώτη κατηγορία (clustering) αφορά την κατηγοριοποίηση των δεδομένων σε υποομάδες με κριτήριο διαχωρισμού τις ομοιότητες και διαφορές στα μεταξύ τους χαρακτηριστικά [5]. Η δεύτερη κατηγορία (association rules) υπάγεται στα μοντέλα μάθησης τα οποία έχουν σκοπό να ανακαλύψουν ενδιαφέροντες συσχετισμούς-κανόνες σε ένα αρκετά μεγάλο σύνολο δεδομένων [6]. Τέλος, οι αλγόριθμοι μείωσης διαστάσεων χρησιμοποιούνται για την συμπίεση των δεδομένων εισόδου υψηλού χώρου διαστάσεων σε ένα σχετικά χαμηλότερο χώρο αποσκοπώντας στην εύρεση απλούστερων τρόπων αναπαράστασης της πληροφορίας [7].

1.1.3 Ημι-Επιβλεπόμενη μάθηση

Η ημι-επιβλεπόμενη μάθηση είναι μια κατηγορία μεταξύ επιβλεπόμενης και μη επιβλεπόμενης μάθησης. Αφορά προβλήματα μάθησης τα οποία περιλαμβάνουν ένα μικρό σύνολο επισημασμένων δεδομένων, και ένα μεγαλύτερο σύνολο το οποίο περιέχει μονάχα μεταβλητές εισόδου. Η χρήση αλγορίθμων αυτού του είδους γίνεται όταν η διαδικασία απόκτησης αρκετών επισημασμένων δεδομένων είναι χρονοβόρα και έρχεται με μεγάλο χρηματικό κόστος [8] .

1.1.4 Ενισχυτική μάθηση

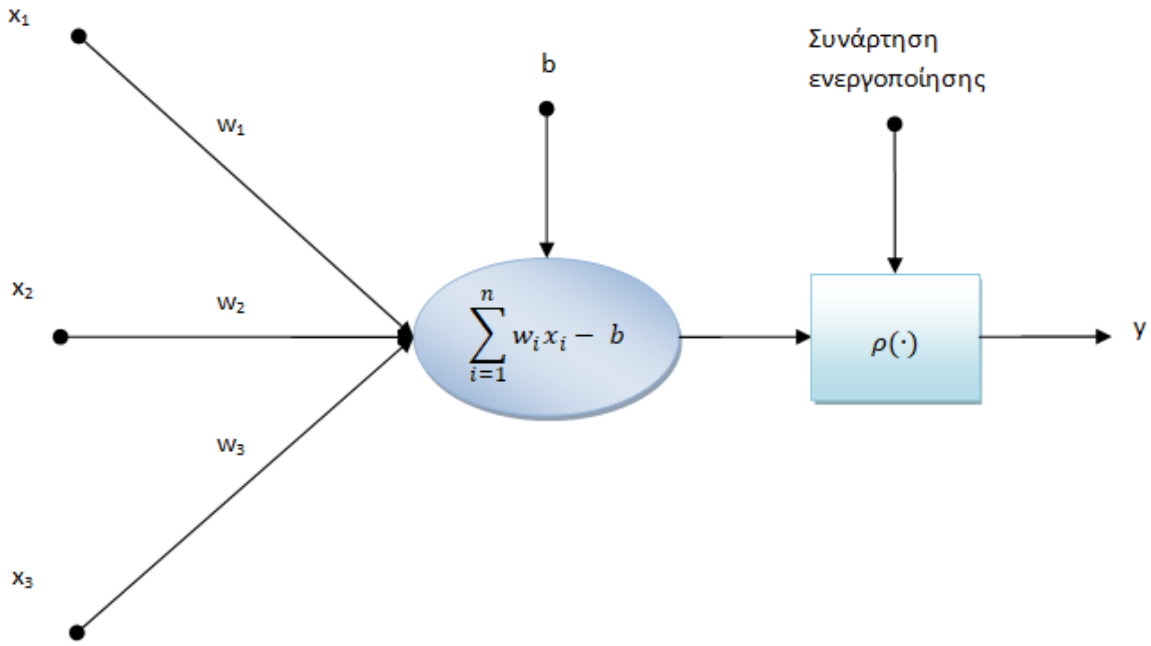
Οι άνθρωποι «μαθαίνουν» μέσω αλληλεπίδρασης με το περιβάλλον τους και σε αυτή τη βασική ιδέα στηρίζονται τα μοντέλα ενισχυτικής μάθησης. Συγκεκριμένα, τα προβλήματα ενισχυτικής μάθησης περιλαμβάνουν την εκμάθηση μέσω συσχέτισης καταστάσεων–ενεργειών ενός «πράκτορα» με σκοπό τη μεγιστοποίηση ενός αριθμητικού σήματος ανταμοιβής. Τα μοντέλα αυτά είναι κλειστού βρόγχου και αυτό διότι οι ενέργειες επηρεάζουν τις μετέπειτα εισόδους του συστήματος. Ο «πράκτορας» δεν γνωρίζει ποια ενέργεια να πράξει αλλά πρέπει ανακαλύψει την ενέργεια για την οποία η ανταμοιβή είναι η υψηλότερη δυνατή [9] .

1.2 Τεχνητά νευρωνικά δίκτυα

Η ιδέα πίσω από την δημιουργία των τεχνητών νευρωνικών δικτύων προέρχεται από τις βιολογικές διεργασίες που μπόρεσαν οι επιστήμονες να παρατηρήσουν σε πραγματικούς νευρώνες ανθρώπινου εγκεφάλου. Οι νευρώνες αποτελούν τα δομικά στοιχεία ενός νευρωνικού δικτύου. Ο πρώτος τεχνητός νευρώνας που αναπτύχθηκε, γνωστός και ως Perceptron εμπνεύστηκε από τα βασικά μέρη από τα οποία αποτελείται ένας ανθρώπινος νευρώνας και αυτά είναι οι δενδρίτες, τα κυτταρικά σώματα και οι νευράξωνες [10], [11].

Η βασική μονάδα των νευρώνων (βιολογικοί και τεχνητοί) είναι οι δενδρίτες, μέσω των οποίων μεταφέρονται σήματα στο κυτταρικό σώμα. Τα σήματα αφού συσσωρευτούν στο κυτταρικό σώμα τροποποιούνται και λαμβάνεται μια απόκριση η οποία συμπεραίνει αν θα πρέπει το σήμα να περάσει σε επόμενο νευρώνα ή όχι. Στη γενική του μορφή , ένας τεχνητός νευρώνας περιέχει παραμέτρους γνωστές ως βάρη (weights) $\{w_1, \dots, w_n\} \in R$, έναν σταθερό όρο (bias) b$\in R$ και μια συνάρτηση ενεργοποίησης ρ : $R \to R$ (θα αναλυθεί στη συνέχεια) και μαθηματικά ορίζεται ως [11] :

$$f(x_1, \dots, x_n) = \rho(\sum_{i=1}^{n} x_i w_i - b) : R^n \to R \qquad (1.1)$$



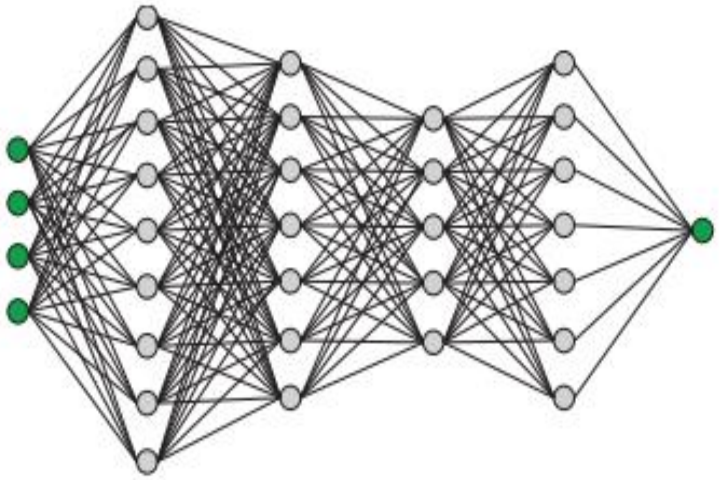
Εικόνα 1.3: Κλασσική αρχιτεκτονική ενός νευρώνα τριών εισόδων και μίας εξόδου με χρήση όρου bias

Όπως συμβαίνει και στον ανθρώπινο εγκέφαλο, οι νευρώνες ενώνονται και διατάσσονται σε διάφορα επίπεδα στρωμάτων (εισόδου – κρυφά – εξόδου) σχηματίζοντας έτσι ένα δίκτυο εμπρόσθιας ανατροφοδότησης (feed-forward neural network). Το δίκτυο αυτό μεταφέρει το σήμα από το στρώμα εισόδου και μέσω των κρυφών στρωμάτων στο στρώμα εξόδου. Έστω ότι το στρώμα εισόδου έχει διάσταση $N_0 = d \in R$, τα κρυφά στρώματα συμβολίζονται ως $N_l \, με \, l = \{1, \dots, L-1\}$ και N_L το στρώμα εξόδου. Τότε για $l = 1, \dots, L$ στρώματα, ορίζεται μια μη γραμμική συνάρτηση ενεργοποίησης $\rho : R \to R$ και T_l γραμμικές συναρτησεις με $W^{(l)} \in R^{N_l \, x \, N_{l-1}}$ τα μητρώα βαρών και $b^{(l)} \in R^{N_l}$ το διάνυσμα των σταθερών όρων του στρώματος l [11] .

$$T_l x = W^{(l)} x + b^{(l)} \; : \; R^{N_{l-1}} \to R^{N_l} \tag{1.2}$$

Ένα νευρωνικό δίκτυο βάθους l (neural network of depth L) αποτυπώνεται μαθηματικά ως:

$$\Phi(x) = T_L\rho\left(T_{L-1}\rho\left(\dots\rho\left(T_1(x)\right)\right)\right): \quad R^d \to R^{N_L} \quad , x \in R^d \tag{1.3}$$



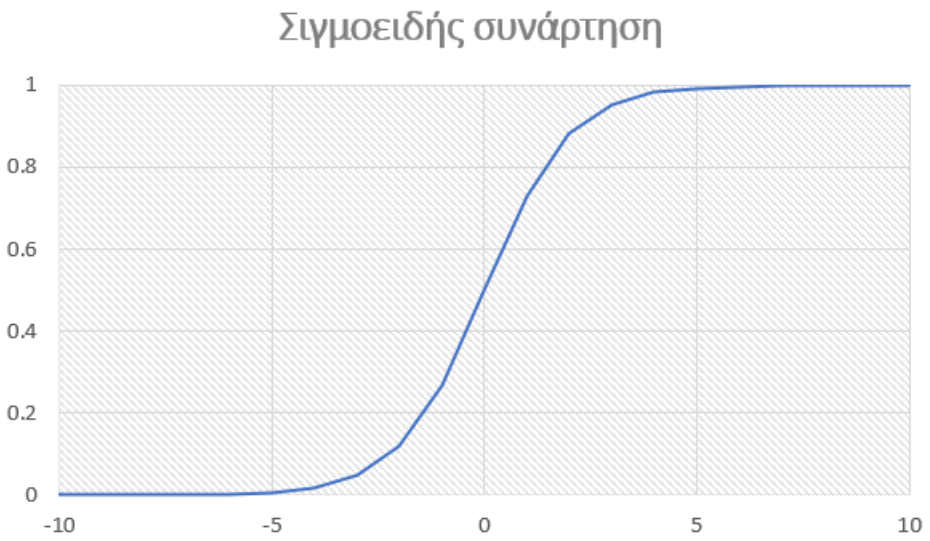
Εικόνα 1.4: Βαθύ νευρωνικό δίκτυο με 4 νευρώνες εισόδου, 4 κρυφά στρώματα και 1 νευρώνα εξόδου [11].

1.3 Συνάρτηση ενεργοποίησης

Οι συναρτήσεις ενεργοποίησης χρησιμοποιούνται στα τεχνητά νευρωνικά δίκτυα και συγκεκριμένα «δρουν» σε κάθε νευρώνα μετατρέποντας το σήμα εισόδου σε σήμα εξόδου. Σε περίπτωση απουσίας τους, τα σήματα εξόδου θα ήταν γραμμικές συναρτήσεις και θα ήταν δύσκολο να αναγνωριστούν περίπλοκα μοτίβα στα δεδομένα. Η προβλεπτική ικανότητα ενός νευρωνικού δικτύου εξαρτάται σε μεγάλο βαθμό από την συνάρτηση ενεργοποίησης που εφαρμόζεται. Οι συναρτήσεις αυτές μπορούν να είναι γραμμικές είτε μη γραμμικές. Ωστόσο στον πραγματικό κόσμο τα σφάλματα είναι μη γραμμικής φύσεως και κατά συνέπεια οι μη γραμμικές συναρτήσεις προτιμώνται . Οι πιο διαδεδομένες μη-γραμμικές συναρτήσεις ενεργοποίησης δίνονται παρακάτω [12]:

- Σιγμοειδής συνάρτηση (sigmoid function): Η συνάρτηση βρίσκει μεγάλο πεδίο εφαρμογών σε περιπτώσεις κατηγοριοποίησης δυαδικών δεδομένων. Το πεδίο ορισμού της είναι το $\{-\infty\,, +\infty\}$ και μετατρέπει τα σήματα εισόδου σε τιμές μεταξύ του 0 και του 1. Είναι διαφοροποιήσιμη σε όλο το πεδίο ορισμού της , συμμετρική ως προς το 0 και ορίζεται ως εξής:

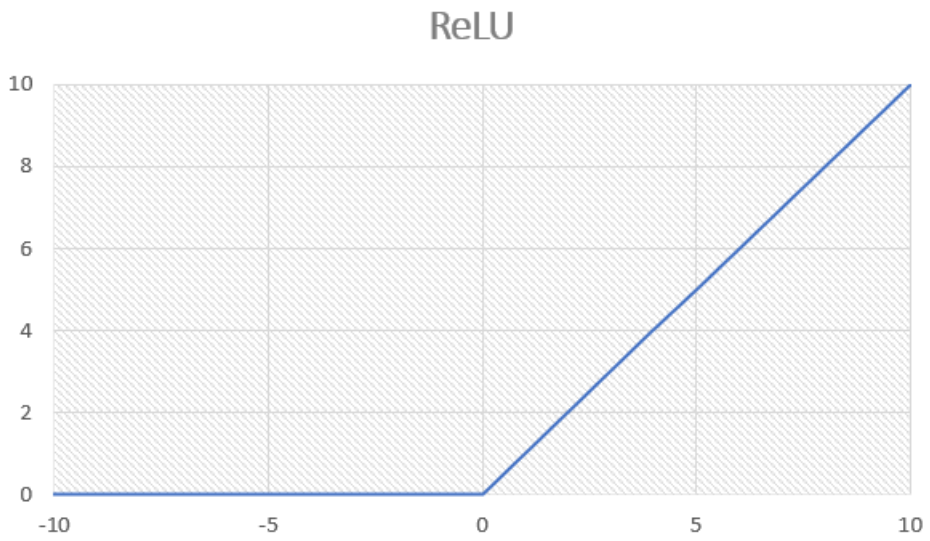
$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{1.4}$$



Εικόνα 1.5: Γραφική παράσταση σιγμοειδούς συνάρτησης

- Συνάρτηση διορθωμένης γραμμικής μονάδας (rectified linear unit – ReLU): Η συνάρτηση αυτή αποτελεί και την πιο ευρέως χρησιμοποιημένη στη βαθιά μάθηση. Επιστρέφει την τιμή της εισόδου αν αυτή είναι θετική, ενώ αν η τιμή εισόδου είναι αρνητική μηδενίζει το σήμα εξόδου και πρακτικά απενεργοποιεί τον νευρώνα. Αυτό την καθιστά αρκετά αποτελεσματική διότι δεν ενεργοποιεί όλους τους νευρώνες μαζί αλλά έναν συγκεκριμένο αριθμό κάθε φορά. Μαθηματικά ορίζεται ως :

$$f(x) = \max(0, x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{1.5}$$



Εικόνα 1.6: Γραφική παράσταση συνάρτησης ReLU

- Συνάρτηση διαρρέουσας διορθωμένης γραμμικής μονάδας (Leaky ReLU): Η συγκεκριμένη συνάρτηση αποτελεί μια τροποποίηση της ReLU. Η διαφορά έγκειται στις αρνητικές τιμές του x για τις οποίες ορίζεται ως μια πολύ μικρή γραμμική συνάρτηση ως προς x. Η μαθηματική της αποτύπωση είναι :

$$f(x) = \begin{cases} 0.01x, & x < 0 \\ x, & x \geq 0 \end{cases} \tag{1.6}$$

- Απαλοιφομένη συνάρτηση (Softmax): Η συνάρτηση softmax ουσιαστικά αποτελεί έναν συνδυασμό πολλαπλών σιγμοειδών συναρτήσεων. Σε αντίθεση με τη σιγμοειδή συνάρτηση χρησιμοποιείται σε προβλήματα ταξινόμησης πολλαπλών κλάσεων (άνω των 2). Πρακτικά δέχεται ένα σύνολο τιμών εισόδου και το μετατρέπει σε μια πιθανότητα από την οποία προκύπτει η κατηγοριοποίηση ως προς μια εκ των κλάσεων. Μαθηματικά εκφράζεται ως:

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \tag{1.7}$$

, όπου $j = 1, \dots, K$ οι νευρώνες του στρώματος που εφαρμόζεται η softmax.

1.4 Συνάρτηση κόστους

Προκειμένου να εκπαιδευτούν τα μοντέλα βαθιάς μάθησης, είναι αναγκαίο να οριστεί μια αντικειμενική συνάρτηση κόστους. Η συνάρτηση αυτή θα πρέπει να είναι μετρήσιμη, θα αξιολογεί και θα βελτιώνει την απόδοση του μοντέλου κατά την εκπαίδευση. Στην πλειοψηφία των προβλημάτων επιβλεπόμενης μηχανικής μάθησης, στόχος είναι να βρεθεί μια παραμετρική αντικειμενική συνάρτηση f_Θ η οποία να περιγράφει μια σχέση μεταξύ μεταβλητών εισόδου X και μεταβλητών εξόδου y [13]:

$$f: X \to y \tag{1.8}$$

Πιο συγκεκριμένα, ένα δοσμένο σύνολο εισόδων (inputs) $\{x_0, \dots, x_n\}$ χρησιμοποιείται για να εκπαιδεύσει ένα μοντέλο συσχετισμένο με ένα σύνολο τιμών εξόδου (targets) $\{y_0, \dots, y_n\}$. Μια συνάρτηση κόστους L ορίζεται ως η αντιστοίχιση των προβλέψεων $f(x_i)$ με τις πραγματικές τιμές εξόδου y_i. Κάθε μία από αυτές τις αντιστοιχίσεις συνιστά ένα κόστος $l \in R$ το οποίο συγκρίνει τις δύο αυτές τιμές. Συγκεντρώνοντας το σύνολο των δεδομένων που υπάρχουν μπορεί να υπολογιστεί το συνολικό κόστος, L ως εξής :

$$L(f|\{x_0, \dots, x_N\}, \{y_0, \dots., y_N\}) = \frac{1}{N}\sum_{i=1}^{N} L(f(x_i), y_i) \tag{1.9}$$

Οι συναρτήσεις κόστους χωρίζονται σε αυτές που είναι για ταξινόμηση (classification) και για παλινδρόμηση (regression). Το πρόβλημα της παρούσας εργασίας αφορά την παλινδρόμηση, δηλαδή την πρόβλεψη μιας συνεχούς μεταβλητής y (εξαρτημένη μεταβλητή) σε συνάρτηση με ένα σύνολο ανεξάρτητων μεταβλητών εισόδου x. Παρακάτω παρουσιάζονται ορισμένες βασικές συναρτήσεις κόστους προβλημάτων παλινδρόμησης:

- Συνάρτηση μέσου σφάλματος κλίσης (Mean Bias Error Loss, MBE): Συνιστά την πιο απλή συνάρτηση για παλινδρόμηση ωστόσο δεν χρησιμοποιείται συχνά. Ο λόγος έγκειται στο γεγονός ότι θετικά σφάλματα μπορούν να επισκιάσουν τα αρνητικά, γεγονός που οδηγεί σε λανθασμένη εκτίμηση των παραμέτρων. Ορίζεται ως:

$$L_{MBE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - f(x_i)) \tag{1.10}$$

- Συνάρτηση μέσου απόλυτου σφάλματος (Mean Absolute Error Loss, MAE, L1Loss): Χρησιμοποιώντας αυτή τη συνάρτηση, υπολογίζεται το μέσο απόλυτο σφάλμα μεταξύ πρόβλεψης και πραγματικής τιμής. Με αυτόν τον τρόπο λύνεται και το πρόβλημα της MBE, αφού τα θετικά σφάλματα δεν αναιρούν τα αρνητικά λόγω της απόλυτης τιμής. Σε αυτή τη συνάρτηση, οι μικρές τιμές σφαλμάτων λαμβάνονται ως εξίσου σημαντικές με τις μεγάλες τιμές. Μαθηματικά αποτυπώνεται ως:

$$L_{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - f(x_i)| \tag{1.11}$$

- Συνάρτηση μέσου τετραγωνικού σφάλματος (Mean Squared Error Loss, MSE, L2Loss): Η MSE υπολογίζει το μέσο τετραγωνικό σφάλμα της πρόβλεψης. Σε αντίθεση με το μέσο απόλυτο σφάλμα, στην MSE οι μεγάλες τιμές σφαλμάτων επηρεάζουν περισσότερο την συνάρτηση σχέση με τις μικρές.

$$L_{MSE} = \frac{1}{N}\sum_{i=1}^{N}\big(y_i - f(x_i)\big)^2 \tag{1.12}$$

1.5 Αλγόριθμοι βελτιστοποίησης

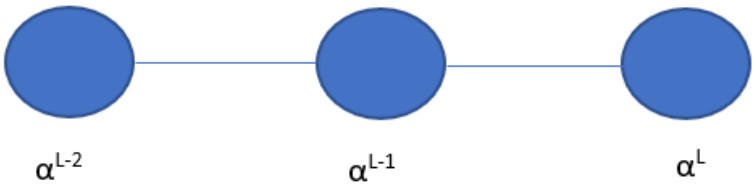
Οι συναρτήσεις κόστους από μόνες τους αποτελούν μια ένδειξη της καλής ή κακής εκπαίδευσης ενός νευρωνικού δικτύου. Οι αλγόριθμοι βελτιστοποίησης είναι εκείνοι οι οποίοι επιτρέπουν στον υπολογιστή να εκπαιδευτεί μέσω των δεδομένων. Επιδιώκουν δηλαδή να ελαχιστοποιήσουν τη συνάρτηση κόστους σε συνάρτηση με τις παραμέτρους (Θ) του νευρωνικού δικτύου [13]:

$$\min_{\Theta}\frac{1}{N}\sum_{i=1}^{N}L(f_{\Theta}(x_i), y_i) \tag{1.13}$$

$$\text{όπου } \theta \in \Theta \text{ οι παράμετροι του δικτύου}$$

- Αλγόριθμος οπισθοδιάδοσης

Η βαθιά μάθηση στηρίζεται στη βασική ιδέα της βελτιστοποίησης μιας επιλεγμένης αντικειμενικής συνάρτησης κόστους. Σε κάθε επανάληψη, επιδιώκεται η προσαρμογή-αναβάθμιση των παραμέτρων και σταθερών όρων του νευρωνικού δικτύου με σκοπό την ελαχιστοποίηση του συνολικού κόστους. Όπως αναφέρθηκε και σε προηγούμενο κεφάλαιο, ένα νευρωνικό δίκτυο πηγαίνει από το στρώμα εισόδου προς το στρώμα εξόδου μέσω της εμπρόσθιας τροφοδότησης (feed-forward). Είναι δυνατό ωστόσο, μετά από τον υπολογισμό του κόστους να εφαρμοστεί και ένας αλγόριθμος γνωστός ως οπισθοδιάδοση (backpropagation) [14]. Αυτός ο αλγόριθμος ξεκινώντας από το τελευταίο στρώμα του δικτύου κάνει υπολογισμούς προς τα πίσω κατά μήκος του δικτύου επιδιώκοντας την ελαχιστοποίηση της συνάρτησης κόστους. Ας υποθέσουμε ένα απλό παράδειγμα νευρωνικού δικτύου με έναν νευρώνα εισόδου, έναν κρυφό και έναν εξόδου όπως στην παρακάτω εικόνα.



Εικόνα 1.7: Νευρωνικό δίκτυο τριών συνολικών στρωμάτων με ένα νευρώνα το κάθε ένα [14].

Έστω ότι οι εξισώσεις εξόδου του κάθε νευρώνα (χωρίς και με την συνάρτηση ενεργοποίησης) και η συνάρτηση κόστους είναι οι εξής :

$$z^L = w^L a^{L-1} + b^L \tag{1.14}$$

$$\alpha^L = \sigma(z^L) \tag{1.15}$$

$$L = (\alpha^L - y)^2 \tag{1.16}$$

, με w^L να είναι το βάρος του νευρώνα στο στρώμα L, σ μια συνάρτηση ενεργοποίησης και y η τιμή της εξόδου. Η συνάρτηση κόστους έχει οριστεί σε μια απλή μορφή για την απλοποίηση της εφαρμογής οπισθοδιάδοσης. Ξεκινώντας από τον τελικό νευρώνα

εφαρμόζεται ο κανόνας της αλυσίδας (chain rule) σε κάθε ενεργοποιημένο νευρώνα. Με αυτόν τον τρόπο μπορεί να καθοριστεί η ευαισθησία της αντικειμενικής συνάρτησης κόστους ως προς το βάρος w^L ως εξής :

$$\frac{\delta C_k}{\delta w^L} = \frac{\delta z^L}{\delta w^L}\frac{\delta a^L}{\delta z^L}\frac{\delta C_k}{\delta \alpha^L} \tag{1.17}$$

Οι μερικές παράγωγοι της παραπάνω συνάρτησης μπορούν να υπολογιστούν εύκολα :

$$\frac{\delta C_k}{\delta \alpha^L} = 2(a^L - y) \tag{1.18}$$

$$\frac{\delta a^L}{\delta z^L} = \sigma'(z^L) \tag{1.19}$$

$$\frac{\delta z^L}{\delta w^L} = a^{L-1} \tag{1.20}$$

$$\frac{\delta C_k}{\delta w^L} = a^{L-1}\sigma'(z^L)2(a^L - y) \tag{1.21}$$

Ο κανόνας της αλυσίδας εφαρμόζεται για κάθε παράμετρο του μοντέλου (βάρη). Για ένα πιο περίπλοκο δίκτυο με πολλαπλά βάρη σε έναν νευρώνα στα ίδια ακριβώς στρώματα η μέθοδος αυτή γενικεύεται:

$$\frac{\delta C_k}{\delta w^L} = \frac{1}{n}\sum_{k=0}^{n-1}\frac{\delta C_k}{\delta w^L} \tag{1.22}$$

Και επιπλέον με τον ίδιο τρόπο υπολογίζονται οι μερικές παράγωγοι ως προς τους σταθερούς όρους (bias).

$$\frac{\delta C}{\delta b^L} = \frac{1}{n}\sum_{k=0}^{n-1}\frac{\delta C_k}{\delta b^L} \tag{1.23}$$

Οι παραπάνω εξισώσεις εφαρμόζονται για το στρώμα εξόδου και το αμέσως προηγούμενο κρυφό στρώμα. Για να προχωρήσει κανείς ακόμα πιο πίσω κατά μήκος του δικτύου, όπως είναι αναμενόμενο γίνεται ξανά εφαρμογή του κανόνα αλυσίδας. Υπολογίζονται έτσι οι παράγωγοι της συνάρτησης κόστους ως προς τους ενεργοποιημένους νευρώνες του αμέσως προηγούμενου στρώματος.

$$\frac{\delta C_k}{\delta a^{L-1}} = \frac{\delta z^L}{\delta a^{L-1}} \frac{\delta a^L}{\delta z^L} \frac{\delta C_k}{\delta a^L} \tag{1.24}$$

Τα νευρωνικά δίκτυα συνήθως είναι αρκετά πιο περίπλοκα, περιέχοντας αρκετούς νευρώνες και κρυφά στρώματα. Οι εξισώσεις που γενικεύουν την οπισθοδιάδοση για κάθε είδος βαθέως νευρωνικού δικτύου είναι οι εξής :

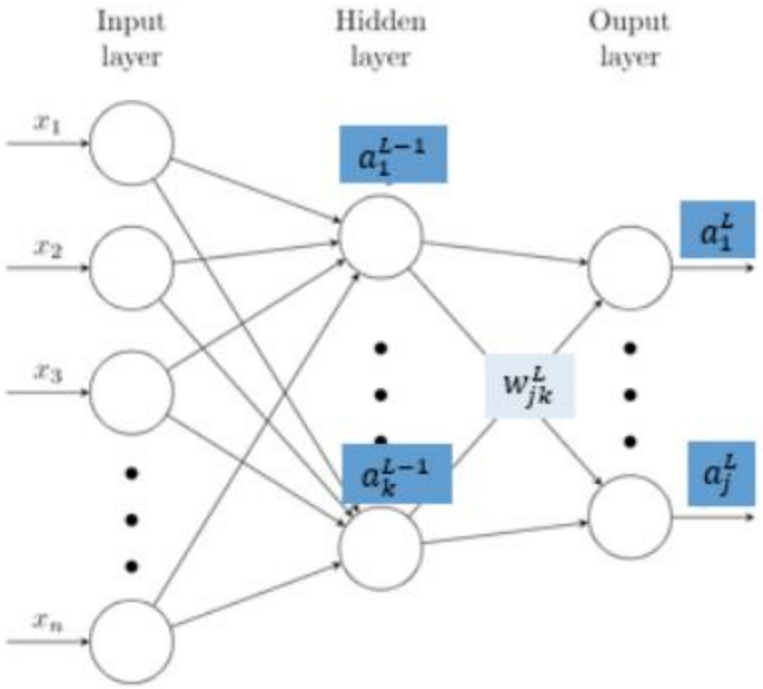
$$C_m = \sum_{j=0}^{n_{L-1}} \left(a_j{}^L - y_j\right)^2 \tag{1.25}$$

$$a_j = \sigma\left(z_j{}^L\right) \tag{1.26}$$

$$z_j{}^L = \cdots + \ w_{jk}{}^L a_k{}^{L-1} + \cdots \tag{1.27}$$

$$\frac{\delta C_k}{\delta w_{jk}{}^L} = \frac{\delta z^L}{\delta w_{jk}{}^L} \frac{\delta a_j{}^L}{\delta z_j{}^L} \frac{\delta C_m}{\delta a_j{}^L} \tag{1.28}$$

$$\frac{\delta C_m}{\delta a^{L-1}} = \sum_{j=0}^{n_{L-1}} \frac{\delta z^L}{\delta a_k{}^L} \frac{\delta a_j{}^L}{\delta z_j{}^L} \frac{\delta C_m}{\delta a_j{}^L} \tag{1.29}$$



Εικόνα 1.8: Νευρωνικό δίκτυο για κατανόηση των εξισώσεων της οπισθοδιάδοσης [14]

Όταν υπολογιστούν όλες οι μερικές παράγωγοι ως προς κάθε παράμετρο μπορούν να συνοψιστούν σε έναν διάνυσμα ή πίνακα. Η διαδικασία της οπισθοδιάδοσης μπορεί να επαναληφθεί όσες φορές επιθυμεί κανείς έως ότου κατάληξης σε μια ελάχιστη αποδεκτή τιμή της συνάρτησης κόστους. Κάθε φορά που εφαρμόζεται ο αλγόριθμος αυτός από το τελευταίο στρώμα έως το πρώτο αποτελεί μια επανάληψη ή εποχή.

$$\nabla C = \begin{bmatrix} \frac{\delta C}{\delta w^1} \\ \frac{\delta C}{\delta b^1} \\ \frac{\delta C}{\delta w^L} \\ \frac{\delta C}{\delta b^L} \end{bmatrix} \tag{1.30}$$

Οι αλγόριθμοι μάθησης που βασίζονται σε κατάβαση δυναμικού (gradient based learning algorithms) αποτελούν τους πιο ικανούς και διαδεδομένους για την επίτευξη μιας τέτοιας ελαχιστοποίησης. Στη συνέχεια παρατίθενται ορισμένοι από τους αλγόριθμους αυτούς [15]:

- Αλγόριθμος κατάβασης δυναμικού παρτίδας (Vanilla-Batch Gradient Descent, VBGD)

Για ένα δοσμένο σετ *T* επισημασμένων δεδομένων ο αλγόριθμος κατάβασης δυναμικού παρτίδας βελτιστοποιεί τις παραμέτρους σύμφωνα με την παρακάτω εξίσωση.

$$\theta^{(\tau)} = \theta^{(\tau-1)} - \eta \nabla_{\theta} L\left(\theta^{(\tau-1)}; T\right) \ , \tau \ \geq 1 \qquad (1.31)$$

Οι όροι της παραπάνω εξίσωσης μεταφράζονται ως εξής :

- Όρος $\boldsymbol{\tau}$: Ορίζεται ως η ενημέρωση της επανάληψης.

- Όρος $\boldsymbol{\theta^{(\tau)}}$: Εκφράζει την αναβαθμισμένη παράμετρο του μοντέλου για την εκάστοτε επανάληψη τ. Για $\tau = 0$, η τιμή $\theta^{(0)}$ υποδηλώνει την αρχική τιμή της παραμέτρου η οποία πολύ συχνά επιλέγεται τυχαία με βάση μια κατανομή (ομοιόμορφη ή κανονική).

- Όρος $\boldsymbol{\nabla_{\theta} L\left(\theta^{(\tau-1)}; T\right)}$: Υπολογίζει την παράγωγο της συνάρτησης κόστους σε συνάρτηση με την παράμετρο θ πάνω σε όλο το σετ T και την παράμετρο $\theta^{(\tau-1)}$ που επιτεύχθηκε στην αμέσως προηγούμενη επανάληψη $\tau - 1$.

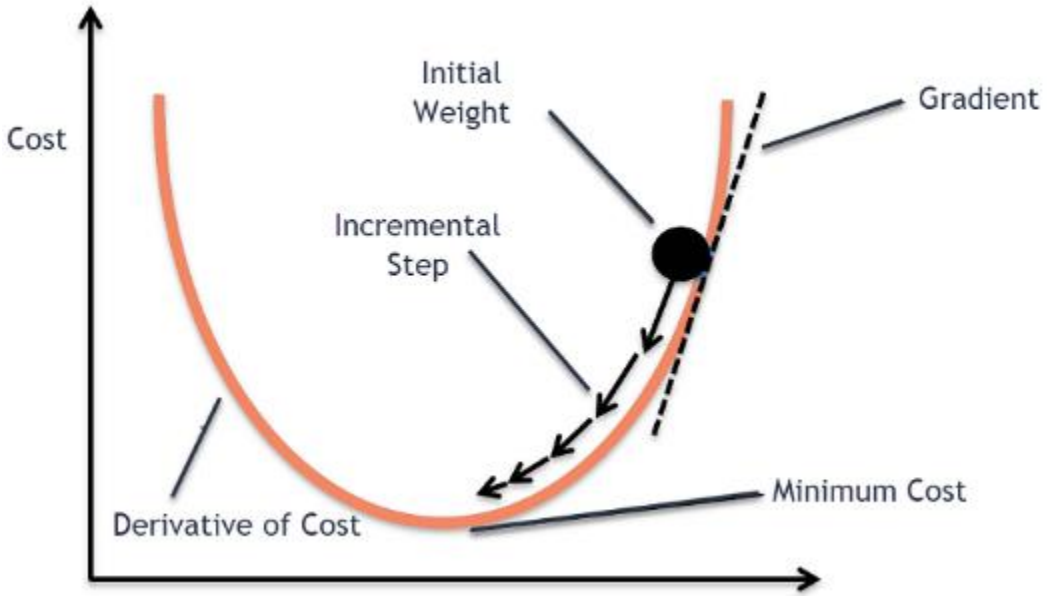
- Όρος $\boldsymbol{\eta}$: Ορίζεται ως ο ρυθμός μάθησης του αλγορίθμου κατάβασης δυναμικού. Αποτελεί μια υπερπαράμετρο του δικτύου η οποία καθορίζει το μέγεθος βήματος κατάβασης δυναμικού. Η τιμή του παίζει πολύ σημαντικό ρόλο στο πόσο αργά ή γρήγορα θα μπορέσει να ελαχιστοποιηθεί η συνάρτηση κόστους. Συνήθως οι τιμές που ορίζονται κυμαίνονται από 10^{-5} έως 10^{-1}.

Αυτή η επαναληπτική μέθοδος αναβάθμισης των παραμέτρων συνεχίζεται μέχρι να επιτευχθεί μια σύγκλιση. Η παράμετρος θ στη λήξη των επαναλήψεων αποτελεί μια ολική η τοπική βέλτιστη μεταβλητή του μοντέλου νευρωνικού δικτύου. Παρακάτω παρατίθεται ο ψευδοκώδικας του αλγορίθμου:

Αλγόριθμος κατάβασης δυναμικού παρτίδας

Απαίτησε : Σετ εκπαίδευσης : T ; Ρυθμός μάθησης η ; Κανονική κατανομή
Εξασφάλισε : Παράμετρος μοντέλου **θ**

1. Αρχικοποίηση παραμέτρου θ
2. Αρχικοποίηση σύγκλιση = Λάθος
3. **Όσο** σύγκλιση == Λάθος :
4. Υπολόγισε την παράγωγο $\nabla_\theta L(\theta; T)$ στο σετ δεδομένων T
5. Αναβάθμισε την μεταβλητή $\theta = \theta - \eta \nabla_\theta L(\theta; T)$
6. **Εάν** η συνθήκη σύγκλισης ισχύει **τότε**
7. σύγκλιση = Σωστή
8. **Τέλος εάν**
9. **Τέλος Όσο**
10. **Επέστρεψε** παράμετρο μοντέλου **θ**



Εικόνα 1.9: Γραφική παράσταση προσομοίωσης της μεταβολής του κόστους σε συνάρτηση με το ρυθμό μάθησης χρησιμοποιώντας gradient descent.
Πηγή : https://www.analyticsvidhya.com/blog/2020/10/how-does-the-gradient-descent-algorithm-work-in-machine-learning

Συμπερασματικά με τον προαναφερθέν αλγόριθμο, για να αναβαθμιστεί κάθε παράμετρος του μοντέλου πρέπει να υπολογίζεται η κλίση της συνάρτησης κόστους σε συνάρτηση με τις παραμέτρους $\theta \in R$ ολόκληρου του σετ δεδομένων για κάθε επανάληψη. Το γεγονός αυτό καθιστά τη μέθοδο αρκετά χρονοβόρα και επομένως δεν αποτελεί βέλτιστη επιλογή όταν ένα σετ δεδομένων είναι αρκετά μεγάλο. Για αυτήν ακριβώς τη χρονική βελτίωση έχουν δημιουργηθεί κάποιες παραλλαγές του παραπάνω αλγορίθμου.

- Στοχαστική κατάβαση δυναμικού (Stochastic Gradient Descent, SGD)

Σε αντίθεση με την κατάβαση δυναμικού παρτίδας, η SGD επιδιώκει την αναβάθμιση των παραμέτρων για κάθε κόστος $l \in R$ του κάθε επισημασμένου δεδομένου εκπαίδευσης (x_i, y_i). Με αυτόν τον τρόπο αυξάνεται αρκετά η ταχύτητα σύγκλισης, ωστόσο, υπάρχει η δυνατότητα παρουσίασης έντονων διακυμάνσεων στη συνάρτηση κόστους κατά την διαδικασία αναβάθμισης. Η εξίσωση στην οποία βασίζεται ο αλγόριθμος SGD είναι :

$$\theta^{(\tau)} = \theta^{(\tau-1)} - \eta \nabla_\theta L\left(\theta^{(\tau-1)}; (x_i, y_i)\right) , \tau \geq 1 \qquad (1.32)$$

Στην παραπάνω εξίσωση, ο όρος του κόστους $L\left(\theta^{(\tau-1)}; (x_i, y_i)\right)$ λαμβάνει υπόψιν κάθε ζευγάρι του σετ εκπαίδευσης και αποτελεί τη μοναδική διαφορά με τον κλασσικό αλγόριθμο Gradient Descent. Σύμφωνα με τον αλγόριθμο SGD, το σετ δεδομένων «ανακατεύεται» πριν την αναβάθμιση. Επιλέγοντας έναν αρκετά μικρό ρυθμό μάθησης ή σταδιακά μειώνοντας τον ανά τις επαναλήψεις, ο αλγόριθμος σχεδόν πάντα μπορεί να συγκλίνει σε μια βέλτιστη τιμή (τοπική η ολική) .

- Κατάβαση δυναμικού μικροπαρτίδας (Mini-batch Gradient Descent)

Ο αλγόριθμος κατάβασης δυναμικού μικροπαρτίδας αποτελεί μια μίξη των δυο προαναφερθέντων μεθόδων. Ειδικότερα, η αναβάθμιση των παραμέτρων γίνεται επιλέγοντας ένα υποσύνολο B των συνολικών δεδομένων T. Μαθηματικά αποτυπώνεται ως :

$$\theta^{(\tau)} = \theta^{(\tau-1)} - \eta \nabla_\theta L\left(\theta^{(\tau-1)}; B\right) \quad (1.33)$$

, όπου ο όρος κόστους $L\left(\theta^{(\tau-1)}; B\right)$ περιλαμβάνει μονάχα δεδομένα του μίνι – σετ B. Το μέγεθος μιας παρτίδας B εξαρτάται από το μέγεθος του συνολικού σετ εκπαίδευσης. Συνήθως οι τιμές κυμαίνονται από (64, 128 , 256). Γενικότερα ο αλγόριθμος είναι πιο αποδοτικός όταν το σύνολο T είναι αρκετά μεγάλο. Συγκρίνοντας με τον αλγόριθμο SGD, η νέα αυτή μέθοδος μπορεί να μειώσει σημαντικά την διακύμανση κατά την αναβάθμιση παραμέτρων και να οδηγήσει σε μια πιο σταθερή σύγκλιση [15].

- Αλγόριθμος Adam

Τα τελευταία χρόνια αποκτούν ολοένα και περισσότερο ενδιαφέρον και χρήση οι προσαρμοστικοί αλγόριθμοι βελτιστοποίησης (Adaptive optimization algorithms). Συγκεκριμένα έχουν αναπτυχθεί αλγόριθμοι όπως ο AdaGrad ο οποίος σχεδιάστηκε για να αντιμετωπίζει προβλήματα αραιών κλίσεων (sparse gradient) και ο RMSProp, ο οποίος λειτουργεί με καλή ακρίβεια σε προβλήματα μη σταθερών ρυθμίσεων (on-line and non-stationary settings). Ένας ακόμα πιο σύγχρονος αλγόριθμος , ο Adam (Adaptive moment estimation), σχεδιάστηκε με σκοπό να συνδυάζει τα χαρακτηριστικά των δύο προαναφερθέντων μεθόδων. Πλέον, ο Adam θεωρείται ένας από τους καταλληλότερους και πιο στιβαρούς αλγορίθμους βελτιστοποίησης όταν γίνεται χρήση μεγάλων σετ δεδομένων με παραμετρικούς χώρους υψηλών διαστάσεων [16].

Αλγόριθμος Adam

Απαίτησε : Βήμα εκπαίδευσης α
Απαίτησε : Εκθετικούς ρυθμούς αποσύνθεσης για τρέχουσες εκτιμήσεις: $\beta_1, \beta_2 \in [0, 1)$
Απαίτησε : Αντικειμενική παραμετρική συνάρτηση: $f(\theta)$
Απαίτησε : Αρχικοποίηση διανύσματος παραμέτρου θ_0

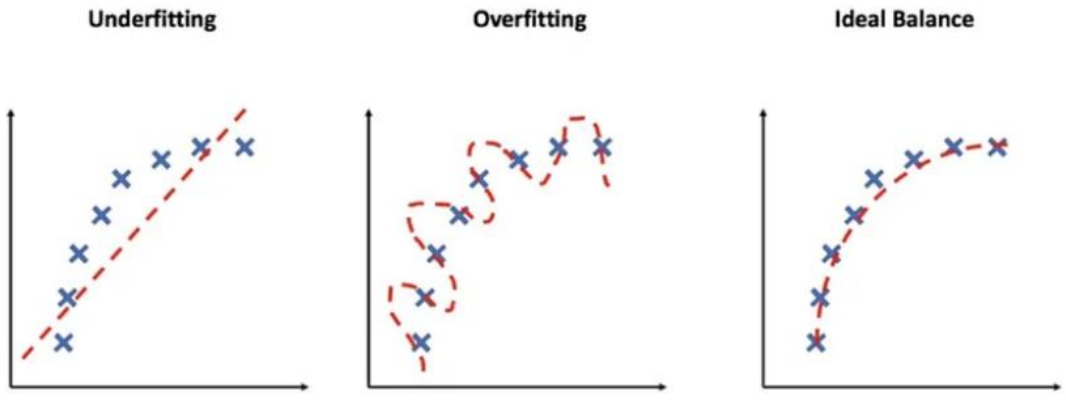
1. $m_0 \leftarrow 0$ (Αρχική τιμή διανύσματος 1ης ροπής)
2. $u_0 \leftarrow 0$ (Αρχική τιμή διανύσματος 2ης ροπής)
3. $t \leftarrow 0$ (Αρχική τιμή χρονικού βήματος)
4. **Όσο** θ_t δεν συγκλίνει κάνε:
5. $t \leftarrow t + 1$
6. $g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$
7. $m_t \leftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$
8. $u_t \leftarrow \beta_2 \cdot u_{t-1} + (1 - \beta_1) \cdot {g_t}^2$
9. $\hat{m}_t \leftarrow m_t / (1 - {\beta_1}^t)$
10. $\hat{u}_t \leftarrow u_t / (1 - {\beta_2}^t)$

11. $\theta_t \leftarrow \theta_{t-1} - \alpha \cdot \hat{m}_t / (\sqrt{\hat{u}_t} + \varepsilon)$
12. **Τέλος Όσο**
13. **Επέστρεψε** παράμετρο μοντέλου θ_t

Για τον αλγόριθμο Adam στην παρούσα εργασία ορίστηκαν $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon = 10^{-8}$, όπως ακριβώς και στη βιβλιογραφία. Όλες οι πράξεις διανυσμάτων του παραπάνω αλγορίθμου αποτελούν πράξεις στοιχείο προς στοιχείο.

1.6 Πρόβλημα υπερπροσαρμογής (Overfitting) και αντιμετώπιση

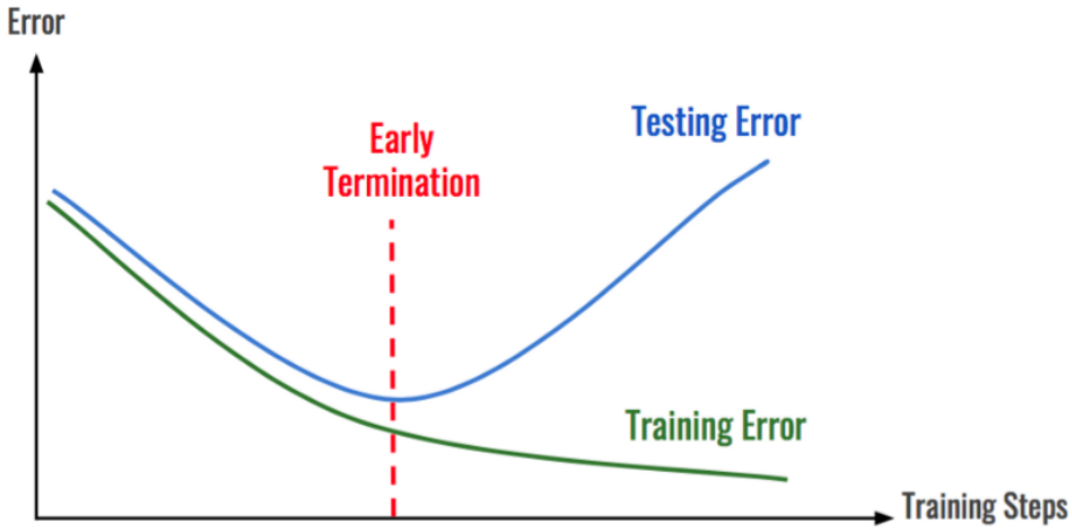
Στα προβλήματα μηχανικής μάθησης, τα μοντέλα μπορεί να εκπαιδεύονται πολύ καλά στα δεδομένα εκπαίδευσης, ωστόσο να μην είναι ικανά να κάνουν καλές προβλέψεις σε άγνωστα δεδομένα. Αυτό το φαινόμενο ονομάζεται υπερπροσαρμογή και συμβαίνει όταν το μοντέλο αποδίδει πολύ καλά στο σετ εκπαίδευσης (training set) και μόνο σε αυτό, μη μπορώντας να αποδώσει το ίδιο στο σύνολο των δεδομένων εξέτασης (test set). Το γεγονός αυτό οφείλεται στην ανικανότητα του μοντέλου να κατανοήσει την πληροφορία των αγνώστων δεδομένων που μπορεί να διαφέρει από αυτή των εκπαιδευόμενων. Ουσιαστικά τα μοντέλα που υπερπροσαρμόζονται μαθαίνουν «παπαγαλία» τα δεδομένα εκπαίδευσης και έτσι δεν είναι σε θέση να αναγνωρίσουν τα μοτίβα που διέπουν ένα σύνολο δεδομένων γενικότερα. Το πρόβλημα αυτό είναι αντιμετωπίσιμο και παρακάτω ακολουθούν μερικές μέθοδοι επίλυσης [17].



Εικόνα 1.10: Γραφική απεικόνισης φαινομένων υποεκπαίδευσης, υπερεκπαίδευσης και ομαλής εκπαίδευσης.
Πηγή :
https://subscription.packtpub.com/book/data/9781838556334/7/ch07lvl1sec82/underfitting-and-overfitting

- Γρήγορη παύση (early stopping)

Πολύ συχνά η ακρίβεια ενός μοντέλου σταματάει να βελτιώνεται ή και χειροτερεύει από ένα σημείο και μετέπειτα. Σε αυτή την περίπτωση, είναι θεμιτό να ελέγχεται η συνάρτηση κόστους του μοντέλου μετά από κάθε επανάληψη (εποχή). Η εκπαίδευση του μοντέλου λήγει όταν σταματάμε να βλέπουμε βελτίωση στην ακρίβεια του μοντέλου.



Εικόνα 1.11: Παράδειγμα γρήγορης παύσης σε διάγραμμα συνάρτησης κόστους-εποχών.
Πηγή : https://medium.com/analytics-vidhya/early-stopping-with-pytorch-to-restrain-your-model-from-overfitting-dce6de4081c5

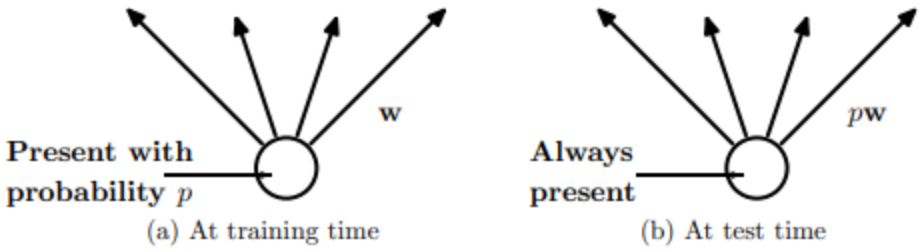
- Ομαλοποίηση (Regularization)

Η απόκριση ενός μοντέλου επηρεάζεται από τον αριθμό των χαρακτηριστικών (features), x στα δεδομένα εισόδου. Όσο περισσότερα είναι τα εισαγόμενα χαρακτηριστικά, τόσο αυξάνεται η πολυπλοκότητα του νευρωνικού δικτύου. Ένα υπερπροσπαρμοσμένο μοντέλο λαμβάνει υπόψιν όλα τα features ακόμα και αυτά που επηρεάζουν σε μικρό βαθμό την έξοδο (output) του δικτύου. Για να περιοριστεί αυτό το φαινόμενο υπάρχουν δύο γενικές κατευθύνσεις:

1. Επιλογή χρήσιμων features και αφαίρεση περιττών
2. Ελαχιστοποίηση παραμέτρων (βαρών) χαρακτηριστικών που επηρεάζουν ελάχιστα την πρόβλεψη

Οι μέθοδοι για να μπορέσει κανείς να αντιμετωπίσει αυτό το πρόβλημα είναι αρκετές. Η πιο διαδεδομένη μέθοδος όσον αφορά την εκπαίδευση βαθιών νευρωνικών δικτύων με πολλούς νευρώνες και παραμέτρους είναι η χρήση της συνάρτησης γνωστής ως Dropout. Η βασική ιδέα αυτής της συνάρτησης είναι να απενεργοποιείται ένα ποσοστό των νευρώνων και των συνδέσεων τους στα κρυφά στρώματα ενός δικτύου. Το αποτέλεσμα είναι, να αποτρέπει τους νευρώνες να προσαρμόζονται υπερβολικά καλά μονάχα στα δεδομένα εκπαίδευσης. Η γενική μεθοδολογία κατά την εκπαίδευση για να επιτευχθεί αυτό έχει ως εξής [18] :

- Απενεργοποίηση μια τυχαίας ποσότητα νευρώνων με βάση μια πιθανότητα p που επιλέγεται.
- Εκπαίδευση του δικτύου χρησιμοποιώντας μονάχα τους μη απενεργοποιημένους νευρώνες για μια εποχή.
- Επαναφορά των ενεργοποιημένων νευρώνων στο τέλος της επανάληψης.
- Επανάληψη της διαδικασίας μέχρις ότου να επιτευχθεί μια ικανοποιητική σύγκλιση.



Εικόνα 1.12: Παράδειγμα νευρώνων με και χωρίς τη χρήση συνάρτησης dropout [18]

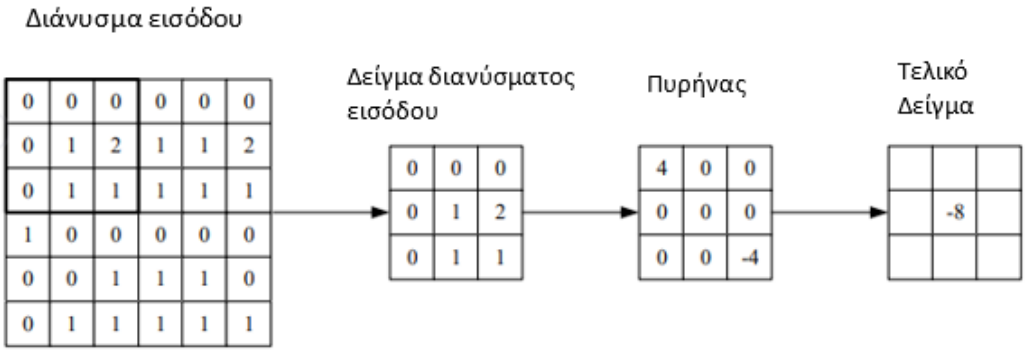
1.7 Συνελικτικά νευρωνικά δίκτυα (Convolutional neural networks, CNNs)

Τα συνελικτικά νευρωνικά δίκτυα αποτελούν αδιαμφισβήτητα μια από τις πλέον πιο διαδεδομένες αρχιτεκτονικές στη βαθιά μάθηση. Αρχικά, δημιουργήθηκαν με σκοπό να αναγνωρίζουν μοτίβα σε εικόνες και αποτελούν σημείο καμπής για την επιστημονική πρόοδο στην επιστήμη της όρασης των υπολογιστών. Είσοδοι του δικτύου αποτελούν οι εικόνες (συνήθως με τη μορφή pixel) και οι νευρώνες των κρυφών στρωμάτων οργανώνονται στον τρισδιάστατο χώρο με τις δύο διαστάσεις να αποτελούν τις χωρικές διαστάσεις εισόδου (ύψος και πλάτος). Η τρίτη διάσταση δεν αναφέρεται στο συνολικό αριθμό των στρωμάτων εντός του νευρωνικού δικτύου αλλά αφορά το βάθος του συνελικτικού δικτύου.

Γενικά τα CNNs περιέχουν τρείς διαφορετικούς τύπους στρωμάτων, το συνελικτικό στρώμα, το στρώμα υποδειγματοληψίας και το πλήρως συνδεδεμένο στρώμα. Αυτά τα 3 στρώματα στοιβάζονται κατά σειρά σχηματίζοντας το πιο απλό συνελικτικό νευρωνικό δίκτυο [19].

1.7.1 Συνελικτικό στρώμα

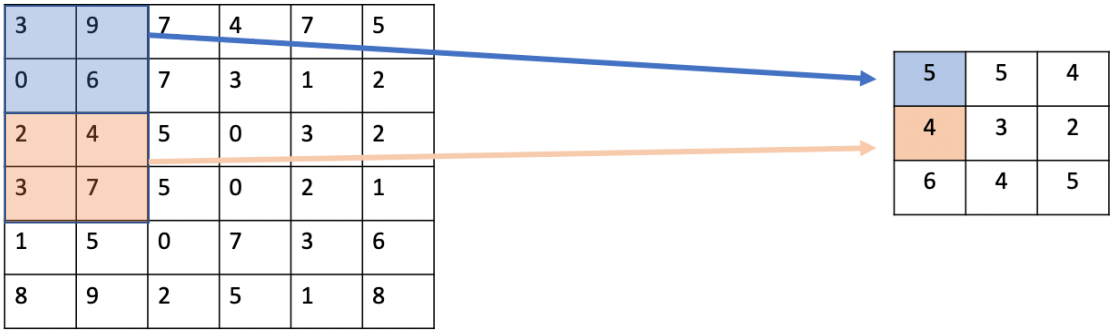
Το συνελικτικό στρώμα αποτελείται από ένα σύνολο εκπαιδεύσιμων φίλτρων. Αυτά τα φίλτρα εκπαιδεύονται με την χρήση πυρήνων (kernels). Οι πυρήνες είναι μικρών χωρικών διαστάσεων (2x2 , 3x3) και εκτείνονται σε κάθε πιθανή θέση κατά βάθος της διάστασης του εισαγόμενου δεδομένου παράγοντας χάρτες ενεργοποίησης (activation maps) 2D. Στη συνέχεια υπολογίζεται το βαθμωτό γινόμενο (scalar product) για όλες τις τιμές ενός πυρήνα. Από αυτό το γινόμενο, το νευρωνικό δίκτυο είναι σε θέση να μάθει τους πυρήνες (ενεργοποιήσεις) του και να αναγνωρίζει συγκεκριμένα χωρικά χαρακτηριστικά ανεξαρτήτου θέσεως σε δεδομένα εισόδου. Παρακάτω δίνεται μια απλή αναπαράσταση δράσης πυρήνα σε ένα συνελικτικό στρώμα.



Εικόνα 1.13: Παράδειγμα δράσης ενός πυρήνα 3x3 σε μια αντίστοιχη διάσταση του διανύσματος εισόδου [19].

1.7.2 Στρώμα υποδειγματοληψίας (Pooling Layer)

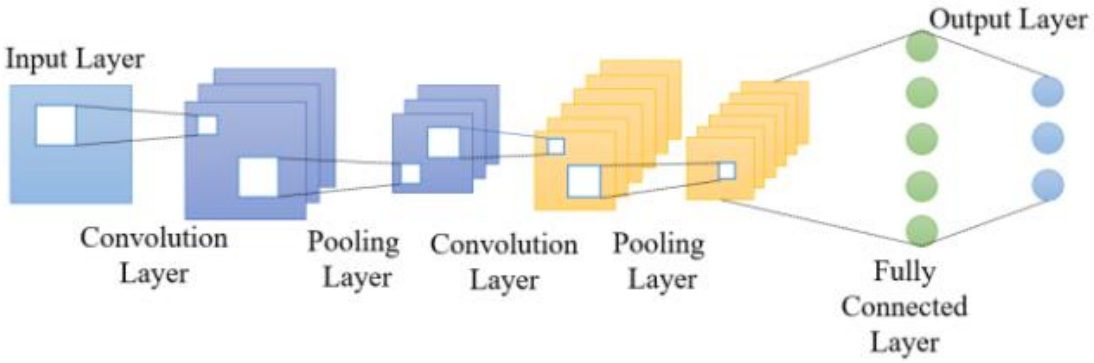
Τα στρώματα υποδειγματοληψίας στοχεύουν στην σταδιακή μείωση των διαστάσεων της αναπαράστασης μειώνοντας τις παραμέτρους και την πολυπλοκότητα του δικτύου. Αυτό το στρώμα εφαρμόζεται πάνω σε κάθε χάρτη ενεργοποίησης της εισόδου και προσαρμόζει την διάσταση χρησιμοποιώντας τη συνάρτηση μεγίστου, μέσου όρου ή αθροίσματος. Στις περισσότερες περιπτώσεις αυτά τα στρώματα αποτελούνται από πυρήνες διαστάσεων $2\ x\ 2$ και αυτό διότι δεν θέλουμε οι διαστάσεις να μειωθούν δραματικά οδηγώντας στην απώλεια μεγάλου μέρους των χαρακτηριστικών και κατά επέκταση της πληροφορίας.



Εικόνα 1.14: Χρήση στρώματος υποδειγματοληψίας μέσου όρου για την μείωση των διαστάσεων ενός διανύσματος εισόδου. Για κάθε πυρήνα 2x2 υπολογίζεται ο μέσος όρος και στρογγυλοποιείται στον πλησιέστερο ακέραιο
Πηγή : https://programmathically.com/what-is-pooling-in-a-convolutional-neural-network-cnn-pooling-layers-explained/

1.7.3 Πλήρως Συνδεδεμένο Στρώμα (Fully connected Layer)

Το τελευταίο κατά σειρά στρώμα εφαρμογής είναι το πλήρως συνδεδεμένο, το οποίο περιέχει νευρώνες που είναι άμεσα συνδεδεμένοι με τους νευρώνες των 2 προηγούμενων στρωμάτων (συνελικτικό και υποδειγματοληψίας). Τέτοιου είδους στρώματα στοχεύουν στην τελική πρόβλεψη που μπορεί να αφορά είτε κατηγοριοποίηση ή παλινδρόμηση.



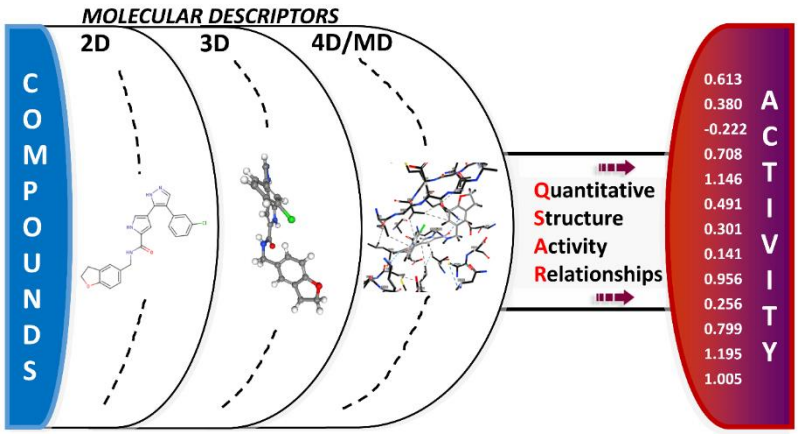
Εικόνα 1.15: Αρχιτεκτονική συνελικτικού νευρωνικού δικτύου

Κεφάλαιο 2: Ποσοτικές σχέσεις δομής – δράσης (QSAR)

Με την πρόοδο της πληροφορικής, η συλλογή, ανάλυση και αποθήκευση των δεδομένων έχει απλοποιηθεί σε μεγάλο βαθμό. Πλέον με τη δημιουργία μεγάλων βάσεων δεδομένων που περιέχουν εκατομμύρια χημικά μόρια η επιστήμη έχει στραφεί στην χρήση τους για την ανάπτυξη νέων φαρμάκων. Αυτό βεβαίως προϋποθέτει αποτελεσματικές διεργασίες που θα συνδυάζουν τις βάσεις δεδομένων με εικονικές βιβλιοθήκες με τη χρήση μορίων με γνωστές ιδιότητες (molecular properties). Οι ποσοτικές σχέσεις δομής – δράσης (QSAR modeling) είναι οι πλέον κυριαρχικές μέθοδοι που χρησιμοποιούνται για αυτά τα προβλήματα. Τα μοντέλα QSAR εξερευνούν και αξιοποιούν τις σχέσεις που συνδέουν τη χημική δομή ενός μορίου με την βιολογική του δράση-ιδιότητα, εξάγοντας χρήσιμες πληροφορίες για την δημιουργία μελλοντικών υποψήφιων φαρμάκων [20].

2.1 Δομή μοντέλων QSAR

Σε γενικές γραμμές μπορούμε να πούμε ότι τα μοντέλα QSAR είναι ένας συνδυασμός του κλάδου την ανάλυσης δεδομένων με τη στατιστική αποσκοπώντας στην εξαγωγή ιδιοτήτων βιολογικής δράσης μέσω της χημικής δομής. Για ένα σετ χημικών μορίων ορίζονται ως οι μοριακοί περιγραφείς τους (που έχουν υπολογιστεί ή μετρηθεί πειραματικά) $D_1, D_2, \dots, D_n$, με δράσεις $P_i = k'(D_1, D_2, \dots, D_n)$ όπου k' ένας μαθηματικός μετασχηματισμός που εφαρμόζεται στους περιγραφείς για την πρόβλεψη των δράσεων-ιδιοτήτων [20].



Εικόνα 2.1: Συσχέτιση μοριακών δομών 2D,3D,4D με βιολογική δράση
Πηγή : https://brc.ncsu.edu/blog/4d-quantitative-structure-activity-relationship-modeling-making-a-comeback

Προκειμένου να αναπτυχθεί ένα αξιόπιστο μοντέλο QSAR είναι θεμιτό να ακολουθούνται τα παρακάτω βήματα [21]:

- Επιλογή δομικά συγγενών ουσιών με καλά καταμερισμένες τιμές βιολογικής δράσης (επισημασμένα δεδομένα).
- Διαχωρισμός δεδομένων σε ομάδες εκπαίδευσης και εξέτασης (σε πολλές περιπτώσεις και επαλήθευσης).
- Προσδιορισμός περιγραφικών χαρακτηριστικών της δομής των χημικών ουσιών.
- Επιλογή κατάλληλης μεθόδου μηχανικής μάθησης για την εκπαίδευση του μοντέλου.
- Επιλογή κατάλληλων στατιστικών μεθόδων για την αξιολόγηση της επίδοσης του μοντέλου στα δεδομένα επαλήθευσης και εξέτασης.
- Καθορισμός πεδίου εφαρμογής μοντέλου.
- Επικαιροποίηση-βελτίωση μοντέλου με χρήση περισσότερων δεδομένων ή καλύτερων μεθόδων μηχανικής μάθησης.

2.2 Μοριακός περιγραφέας δομής 2D

Οι μοριακοί περιγραφείς μιας δομής αποτελούν τον πυρήνα των μοντελοποιήσεων QSAR. Χωρίζονται σε διαφορετικούς τύπους, με κάθε έναν να αντικατοπτρίζει διάφορα επίπεδα αναπαράστασης της χημικής δομής (2D, 3D, 4D).

Η δισδιάστατη ή τοπολογική αναπαράσταση μιας χημικής ένωσης ορίζει τη σύνδεση των ατόμων σε ένα μόριο ως προς την παρουσία και τη φύση των χημικών δεσμών. Μια τέτοια περιγραφή αντανακλά σημαντικά πλεονεκτήματα όπως:

- Παροχή απλής και χρήσιμης πληροφορίας για την μοριακή δομή.
- Η πληροφορία είναι αμετάβλητη σε σχέση με την περιστροφική μετάφραση μορίων (roto translation).
- Άμεσος υπολογισμός χωρίς την απαίτηση βελτιστοποίησης της δομής.

Μια από τις καταλληλότερες πλέον 2D αναπαραστάσεις μορίων αποτελεί η πληροφορία που εισάγεται σε ένα μοριακό γράφημα (θα συζητηθεί στο επόμενο κεφάλαιο).

2.3 QSAR μοντέλα βαθιάς μάθησης

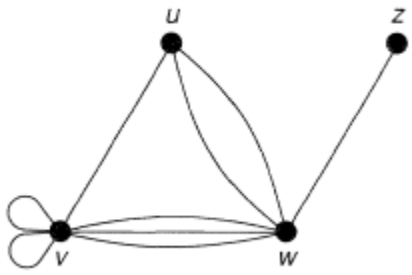
Η ανάπτυξη QSAR μοντέλων πρόβλεψης βασιζόταν για πολλά χρόνια σε απλούς αλγορίθμους μηχανικής μάθησης όπως είναι η Random Forest. Τα τελευταία χρόνια, η εξέλιξη των νευρωνικών δικτύων βάθους οδήγησε στη χρήση τους και στις μεθοδολογίες QSAR αντικαθιστώντας «απαρχαιωμένους» αλγορίθμους. Πιο συγκεκριμένα, η χρήση βαθιάς μάθησης αποδείχτηκε πως μπορεί να βελτιώσει τις αποδόσεις ενός μοντέλου Random Forest πολύ ικανοποιητικά. Αυτό οφείλεται σε μεγάλο βαθμό στην πληθώρα προσαρμόσιμων-εκπαιδεύσιμων παραμέτρων ενός βαθέως νευρωνικού δικτύου. Επίσης, τα βαθιά νευρωνικά δίκτυα μπορούν να αξιοποιήσουν την παράλληλη υπολογιστική ικανότητα μια σύγχρονης κάρτας γραφικών (GPU) , πράγμα που τους δίνει υπεροχή σε σχέση με κλασσικούς αλγορίθμους ως προς το υπολογιστικό κόστος [22].

Κεφάλαιο 3: Νευρωνικά δίκτυα γραφημάτων

Το γράφημα έχει εγκαθιδρυθεί ως ένα μαθηματικό εργαλείο το οποίο μπορεί να χρησιμοποιηθεί σε πολλούς κλάδους της επιστήμης. Μερικοί από αυτούς αφορούν την επιχειρησιακή έρευνα, χημεία, γενετική, γλωσσολογία, κοινωνία, γλωσσολογία καθώς και σε πολλούς τομείς της μηχανικής. Στη σύγχρονη εποχή που η τεχνητή νοημοσύνη αναπτύσσεται με ραγδαίους ρυθμούς, τα γραφήματα έχουν διεισδύσει στη μηχανική και βαθιά μάθηση. Χαρακτηριστική είναι η εφαρμογή τους στην ανακάλυψη φαρμάκων (drug discovery) μιας και όλο και περισσότερες φαρμακευτικές εταιρίες και ερευνητικά κέντρα στρέφουν το ενδιαφέρον τους σε αυτόν τον τομέα. Πιο συγκεκριμένα, τα γραφήματα μπορούν να χρησιμοποιηθούν για την πρόβλεψη ιδιοτήτων μορίων, μια από τις πιο σημαντικές εργασίες στον χώρο της υπολογιστικά – βοηθούμενης ανακάλυψης φαρμάκων (computer aided drug discovery) [23], [24].

3.1 Θεωρία γραφημάτων

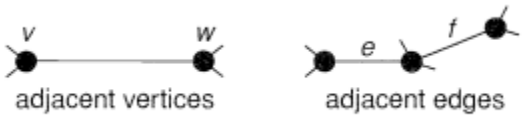
Ένα απλό γράφημα G αποτελείται από ένα μη κενό πεπερασμένο σύνολο στοιχείων $V(G)$ που ονομάζονται κορυφές και από ένα πεπερασμένο σύνολο $E(G)$ αποτελούμενο από μη διατεταγμένα ζεύγη διακριτών στοιχείων του $V(G)$ που ονομάζονται ακμές. Μια ακμή e_{vw} μπορεί να ενώσει το πολύ δύο κορυφές v, w ενός γραφήματος. Αυτές μπορεί να είναι διακριτές μεταξύ τους ή και όχι, δηλαδή μια ακμή μπορεί να ενώνει μια κορυφή με τον εαυτό της. Όταν οι ακμές ενός γραφήματος είναι διατεταγμένα ζεύγη, το γράφημα ονομάζεται κατευθυνόμενο (directed) ενώ στην περίπτωση που οι ακμές δεν έχουν καθορισμένη κατεύθυνση το γράφημα ονομάζεται μη κατευθυνόμενο (undirected) [23].



Εικόνα 3.1: Απλό μη κατευθυνόμενο γράφημα τεσσάρων κορυφών και συνδέσεων μεταξύ τους μέσω ακμών [23]

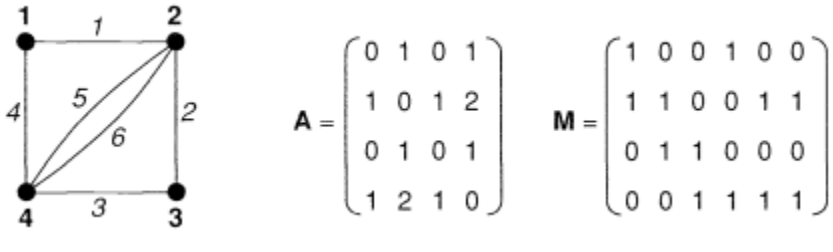
3.1.1 Γειτνίαση και πρόσπτωση γραφημάτων (Graph adjacency , incidence)

Δύο κορυφές v, w είναι γειτονικές (adjacent) στην περίπτωση που μια ακμή e_{vw} τις ενώνει και λέγεται ότι είναι προσπίπτουσες (incident) μεταξύ τους ως προς την ακμή e_{vw}. Ομοίως, δύο διακριτές ακμές e, f είναι γειτονικές μεταξύ τους όταν έχουν κοινή κορυφή. Ως βαθμός (degree) μιας κορυφής v ενός γραφήματος G ορίζεται ο αριθμός των προσπιπτουσών ακμών σε αυτή.



Εικόνα 3.2: Παράδειγμα δύο γειτονικών κορυφών και δύο γειτονικών ακμών [23]

Πίνακας γειτνίασης A (adjacency matrix) για ένα γράφημα G με κορυφές $\{1, \dots, n\}$ και ακμές $\{1, \dots, m\}$ ορίζεται ο πίνακας διαστάσεων $n\ x\ n$ του οποίου το κάθε στοιχείο ij ισούται με τον αριθμό των ακμών που ενώνουν δύο κορυφές i, j. Πίνακας πρόσπτωσης M (incidence matrix) ορίζεται ο πίνακας διαστάσεων $n\ x\ m$ του οποίου το κάθε στοιχείο ij ισούται με 1 στην περίπτωση που η κορυφή i είναι προσπίπτουσα της ακμής j, αλλιώς ισούται με 0 [23].

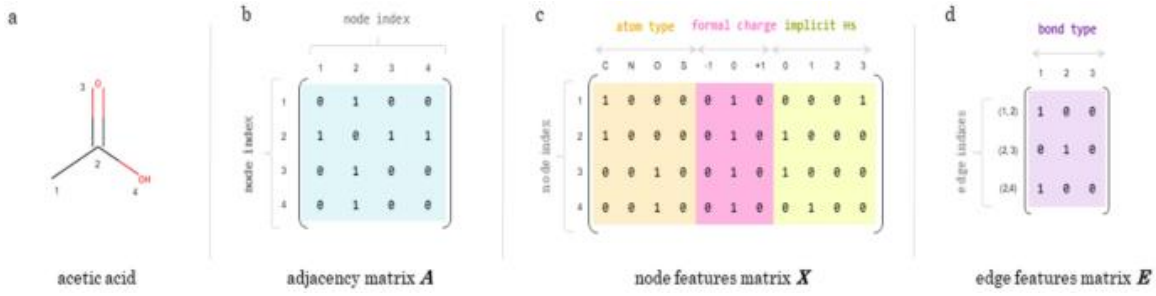


Εικόνα 3.3: Πίνακας γειτνίασης και πρόσπτωσης για ένα γράφημα τεσσάρων κορυφών [23]

3.2 Αναπαράσταση χημικού μορίου ως γράφημα

Η βασική ιδέα της μοριακής αναπαράστασης ως ένα γράφημα βασίζεται στην αντιστοίχιση των ατόμων που αποτελούν το μόριο ως κορυφές γραφήματος και τους χημικούς δεσμούς ως τις ακμές. Οι συνδέσεις των ατόμων μεταξύ τους με δεσμούς περιγράφονται από τον πίνακα γειτνίασης. Είναι σημαντικό να αναφερθεί ότι ο πίνακας αυτός δεν αναγνωρίζει το είδος των δεσμών και τους θεωρεί όλους ως μονούς.

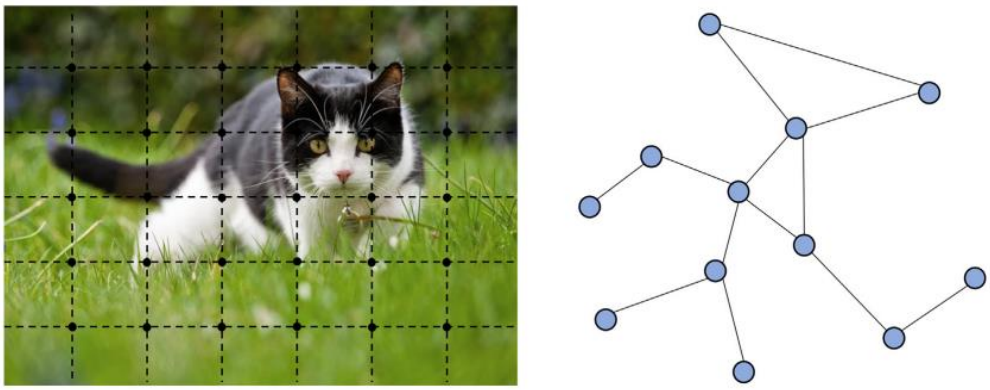
Παρά το γεγονός ότι ένα μοριακό γράφημα αποτελεί ένα δισδιάστατο αντικείμενο, μπορεί να περιέχει πληροφορίες έως και τρισδιάστατου περιεχομένου (δίνοντας χαρακτηριστικά σε κάθε κορυφή και ακμή όπως συντεταγμένες ατόμων, γωνίες δεσμών, χειρομορφία). Η ταυτότητα-πληροφορία των ατόμων αναπαρίσταται από ένα πίνακα γνωστό ως πίνακα χαρακτηριστικών κόμβων X (node features matrix). Αυτή η πληροφορία μπορεί να περιλαμβάνει το όνομα του στοιχείου, τα γειτονικά υδρογόνα και άλλα χαρακτηριστικά κωδικοποιημένα όπως θα δούμε στη συνέχεια. Ο πίνακας X έχει διαστάσεις $C\ x\ n$ όπου C είναι το πλήθος των ατομικών χαρακτηριστικών (χημική πληροφορία) που έχουν οριστεί και n το πλήθος των ατόμων. Ουσιαστικά κάθε γραμμή του πίνακα αποτελεί ένα μονοδιάστατο διάνυσμα που περιλαμβάνει την ταυτότητα ενός συγκεκριμένου ατόμου κάθε φορά. Η πληροφορία των χημικών δεσμών δίνεται από τον πίνακα χαρακτηριστικών ακμής E διαστάσεων $z\ x\ m$, όπου z είναι το πλήθος των εισαγόμενων δεσμικών χαρακτηριστικών και m ο αριθμός των ακμών. Κάθε γραμμή αυτού του πίνακα αντιστοιχεί σε μία ακμή e_{ij}. Μερικές πληροφορίες που μπορούν να κωδικοποιηθούν στον πίνακα αφορούν το είδος των δεσμών, την αρωματικότητα και άλλα. Είναι απαραίτητο να αναφερθεί πως σημαντικότερο ρόλο παίζει ο πίνακας χαρακτηριστικών κορυφής και δεν χρειάζεται κανείς να χρησιμοποιήσει και τους δύο μαζί [25].



Εικόνα 3.4: Παράδειγμα αναπαράστασης οξικού οξέος ως μοριακό γράφημα και εξαγωγή πίνακα γειτνίασης, χαρακτηριστικών κορυφής και χαρακτηριστικών ακμής [25]**.**

3.3 Νευρωνικά δίκτυα γραφημάτων (Graph Neural Networks)

Όπως αναφέρθηκε στο κεφάλαιο 1, οι αρχιτεκτονικές των συνελικτικών νευρωνικών δικτύων (CNN_s) δρουν σε δεδομένα τα οποία υπάγονται στο χώρο της ευκλείδειας γεωμετρίας (2D εικόνες). Ωστόσο, πολλές δομές δεδομένων μπορούν να θεωρηθούν και να αναπαρασταθούν ως γραφήματα. Για αυτό και ήταν αναμενόμενο τα CNN_s να γενικευτούν και να χρησιμοποιηθούν στα γραφήματα. Μια πρόκληση αυτής της γενίκευσης είναι η δυσκολία να οριστούν τοπικά συνελικτικά φίλτρα και τελεστές υποδειγματοληψίας σε γραφήματα. Επομένως, ήταν επακόλουθο η λογική των CNN_s να περάσει από τον ευκλείδειο χώρο στον μη ευκλείδειο. Αυτά τα βαθιά νευρωνικά δίκτυα γραφημάτων (Graph Neural Networks, GNN) υπάγονται στο χώρο της γεωμετρικής βαθιάς μάθησης. Προκειμένου να δημιουργήσουμε ένα μοντέλο νευρωνικού δικτύου γραφήματος πρέπει πρώτα να οριστούν κάποιες υπολογιστικές μονάδες (computational modules). Οι πιο ευρέως χρησιμοποιημένες είναι οι εξής [26]:



Εικόνα 3.5: Εικόνα σε ευκλείδειο χώρο και γράφημα σε μη ευκλείδειο [26]

- **Μονάδα διάδοσης (Propagation Module)**: Χρησιμοποιείται για να διαδώσει και να συγκεντρώσει την πληροφορία μεταξύ των κορυφών ενός γραφήματος. Η πληροφορία αυτή μπορεί να περιέχει χαρακτηριστική και τοπολογική πληροφορία μιας κορυφής. Μονάδες διάδοσης όπως ο τελεστής συνέλιξης, συλλέγουν πληροφορίες γειτονικών κορυφών ενώ ο τελεστής παράλειψης σύνδεσης (skip connection) μπορεί να χρησιμοποιηθεί για την επίλυση της υπερβολικής εξομάλυνσης. Πρακτικά με τη χρήση της μονάδας διάδοσης ανανεώνεται η αναπαράσταση της κάθε κορυφής καθώς διαδίδεται η πληροφορία ανά τα στρώματα.

- **Μονάδα δειγματοληψίας (Sampling Module)**: Σε περίπτωση μεγάλων γραφημάτων, η δειγματοληψία είναι απαραίτητη για να επιτευχθεί η διάδοση (propagation).

- **Μονάδα συγκέντρωσης (Pooling Module):** Είναι απαραίτητη για την εξαγωγή της πληροφορίας από τους κόμβους. Η λογική της μονάδας συγκέντρωσης είναι ίδια με αυτήν που πραγματοποιεί το Pooling στρώμα σε ένα απλό συνελικτικό νευρωνικό δίκτυο. Μειώνει δηλαδή τη διάσταση των κορυφών χρησιμοποιώντας τελεστή μέσου όρου, αθροίσματος ή μεγίστου.

3.4 Συνελικτικά δίκτυα γραφημάτων (Graph Convolutional Networks)

Τα μοντέλα νευρωνικών δικτύων γραφημάτων κάνουν χρήση της συνέλιξης ως μονάδα διάδοσης της πληροφορίας. Όπως αναφέρθηκε και νωρίτερα ο τελεστής αυτός βασίζεται στην ιδέα γενίκευσης συνελίξεων από έναν λειτουργικό χώρο (domain) στον χώρο των γραφημάτων (graph domain). Οι συντελεστές συνέλιξης στα γραφήματα μπορούν να κατηγοριοποιηθούν στις φασματικές μεθόδους (spectral approaches) και στις χωρικές μεθόδους (spatial approaches) [26].

3.4.1 Φασματικές προσεγγίσεις (spectral approaches)

Οι φασματικές μέθοδοι λειτουργούν με βάση μια φασματική αναπαράσταση του γραφήματος. Στηρίζονται στην επεξεργασία των σημάτων που προέρχονται από τα γραφήματα και ορίζουν τον συντελεστή συνέλιξης στον φασματικό χώρο. Γενικότερα, ένα σήμα x από το γράφημα, μετασχηματίζεται μέσω ενός μετασχηματισμού Fourier F στον φασματικό χώρο. Κατόπιν, γίνεται εφαρμογή του συντελεστή συνέλιξης και το τελικό σήμα μετατρέπεται πίσω στον χώρο των γραφημάτων με τη χρήση του αντίστροφου μετασχηματισμού Fourier F^{-1}.

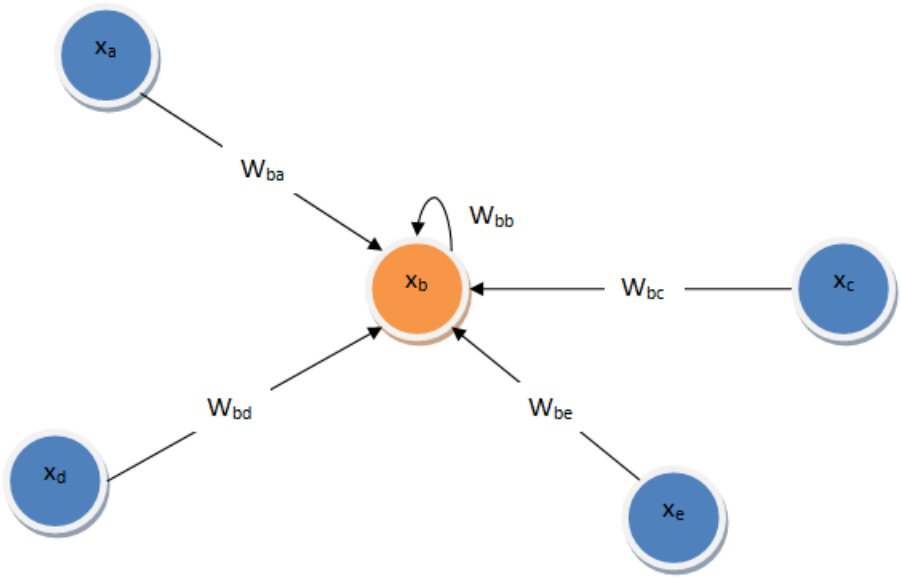
GCN (Graph Convolutional Network)

Τα μοντέλα των συνελικτικών νευρωνικών δικτύων χρησιμοποιούν έναν κανόνα διάδοσης κατά στρώματα, ο οποίος βασίζεται σε μια προσέγγιση πρώτης τάξης των φασματικών συνελίξεων σε γραφήματα. Ένα από τα πρωταρχικά τέτοια μοντέλα και το

πιο γνωστό δημιουργήθηκε από τους Kipf, Weiling [27] και είναι γνωστό ως GCN. Για ένα μοντέλο νευρωνικού δικτύου γραφήματος $f(X, A)$, η εξίσωση διάδοσης κατά στρώματα είναι η εξής:

$$H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right) \tag{3.1}$$

Όπου $\tilde{A} = A + I_N$, ο πίνακας γειτνίασης που περιλαμβάνει τις συνδέσεις κάθε κορυφής με τον εαυτό της, $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$ ένας πίνακας γνωστός και ως πίνακας βαθμού και $W^{(l)} \in R^{F\ x\ F'}$ ένας εκπαιδεύσιμος πίνακας βαρών, μοναδικός για κάθε στρώμα. Οι διαστάσεις F και F' είναι οι διαστάσεις εισόδου και εξόδου του στρώματος αντίστοιχα. Επίσης η συνάρτηση σ αποτελεί μια συνάρτηση ενεργοποίησης (ReLU) και $H^{(l)} \in R^{N\ x\ F}$ καλείται ο πίνακας των ενεργοποιήσεων στο στρώμα l, ο οποίος εισάγεται στο δίκτυο ως ο πίνακας χαρακτηριστικών $H^{(0)} = X$ [26], [27].



Εικόνα 3.6: Ανανέωση πληροφορίας κορυφής με βάση τις παραμέτρους των γειτονικών κορυφών σε ένα GCN

3.4.2 Χωρικές προσεγγίσεις (spatial approaches)

Στις χωρικές προσεγγίσεις των νευρωνικών δικτύων γραφημάτων η συνέλιξη ορίζεται απευθείας στο γράφημα με βάση την τοπολογία του. Πιο συγκεκριμένα, σε αυτές τις μεθόδους η συνέλιξη ορίζεται με βάση τις γειτονικές περιοχές διαφορετικού μεγέθους σε κάθε κόμβο.[26].

GraphSAGE (Graph Sample and Aggregate)

Το μοντέλο GraphSAGE μπορεί να θεωρηθεί σαν μια επέκταση του GCN. Η βασική ιδέα είναι να μπορέσουμε να συγκεντρώσουμε πληροφορίες χαρακτηριστικών (features) από μια τοπική γειτονιά ενός κόμβου. Πρακτικά κάθε κόμβος δεν λαμβάνει πληροφορίες ανανέωσης από κάθε γειτονικό του κόμβο αλλά από ένα υποσύνολο της συνολικής γειτονιάς του. Παρακάτω περιγράφεται η εξίσωση συγκέντρωσης γειτονικής πληροφορίας του μοντέλου μαζί με τον αλγόριθμο εμπρόσθιας τροφοδότησης του, για την διάδοση κατά στρώματα [26], [28].

$$h_{N(u)}^{k} = AGGREGATE_{k}(\{h_{u}^{k-1}, \forall\, u \in \mathrm{N}(\upsilon)\}) \tag{3.2}$$

$$h_{u}^{k} = \sigma\left(W^{k} \cdot CONCAT\left(h_{\upsilon}^{k-1}, h_{N(\upsilon)}^{k}\right)\right) \tag{3.3}$$

, όπου $AGGREGATE_{k,} \forall k \in \{1, \dots, K\}$ ένα σύνολο συναρτήσεων συγκέντρωσης πληροφορίας που αποτελείται από K ανεξάρτητες συναρτήσεις συγκέντρωσης και $W^{k}, \forall k \in \{1, \dots K\}$ ένα σύνολο βαρών για κάθε μια από αυτές, δηλαδή βάρη για κάθε κρυφό στρώμα. Τα βάρη χρησιμοποιούνται για να διαδώσουν την πληροφορία σε όλα τα στρώματα του δικτύου.

Αλγόριθμος εμπρόσθιας τροφοδότησης GraphSAGE

Είσοδος : Γράφημα $G(V, E)$; χαρακτηριστικά εισόδου $\{x_{\nu}, \forall \nu \in V\}$; βάθος K;πίνακες βαρών $W^{k}, \forall k \in \{1, \dots K\}$; μη γραμμική συνάρτηση ενεργοποίησης σ; διαφοροποιήσιμες συναρτήσεις συγκέντρωσης $AGGREGATE_{k,} \forall k \in \{1, \dots, K\}$; γειτονική συνάρτηση $N : \upsilon \rightarrow 2^{V}$

Έξοδος : Διανυσματική αναπαράσταση z_{υ} για όλα τα $\upsilon \in V$

1. $h_{v}^{0} \leftarrow x_{\upsilon}\ \forall \upsilon \in V$;
2. **Για** $k = 1 \dots K$ κάνε
3. **Για** $\upsilon \in V$ κάνε

4. $h_{N(u)}^{k} \leftarrow AGGREGATE_k(\{h_u^{k-1}, \forall\, u \in \mathrm{N}(\upsilon)\})$;
5. $h_u^k \leftarrow \sigma\,(W^k \cdot CONCAT(h_\upsilon^{k-1}, h_{N(\upsilon)}^{k}))$
6. Τέλος για
7. $h_u^k \leftarrow h_u^k / ||h_u^k||_2, \forall\, \upsilon \in V$
8. Τέλος για
9. $z_\upsilon \leftarrow h_u^K\ ,\ \forall\, \upsilon \in V$

Στον παραπάνω αλγόριθμο, όπως και στα περισσότερα νευρωνικά δίκτυα η είσοδος είναι τα χαρακτηριστικά κάθε κορυφής $x_u\,, \forall \in u\, V$ για όλες τις κορυφές V ενός γραφήματος. Ο αριθμός k υποδηλώνει την επανάληψη της εξωτερικής λούπας και h^kείναι η αναπαράσταση της κορυφής στην επανάληψη k. Αρχικά, κάθε κορυφή $\upsilon \in V$ συλλέγει τις πληροφορίες των γειτονικών κορυφών (όχι όλους τους γείτονες) $\{h_u^{k-1}, \forall\, u \in \mathrm{N}(\upsilon)\}$ σε ένα διάνυσμα $h_{N(\upsilon)}^{k-1}$, το οποίο εξαρτάται από τις αναπαραστάσεις της προηγούμενης επανάληψης (Για $k = 0$ οι αναπαραστάσεις είναι ο πίνακας χαρακτηριστικών, δηλαδή η είσοδος του δικτύου). Το επόμενο βήμα είναι η ένωση (concatenation) της τωρινής αναπαράστασης κορυφής h_u^{k-1} με το διάνυσμα συλλογής h_u^{k-1}. Αυτό το ενωμένο διάνυσμα περνάει μέσα από ένα πλήρες συνδεδεμένο στρώμα με τη χρήση μη γραμμικής συνάρτησης ενεργοποίησης σ στο επόμενο στρώμα. Τέλος για ένα δίκτυο βάθους K οι τελικές αναπαραστάσεις κόμβων του τελευταίου κρυφού στρώματος συμβολίζονται ως $z_\upsilon \leftarrow h_u^K\ ,\ \forall\, \upsilon \in V$. Οι συναρτήσεις ένωσης που χρησιμοποιήθηκαν στην παρούσα εργασία χρησιμοποιούν τους δυο παρακάτω τελεστές:

- Μέσο άθροισμα (Mean aggregator)

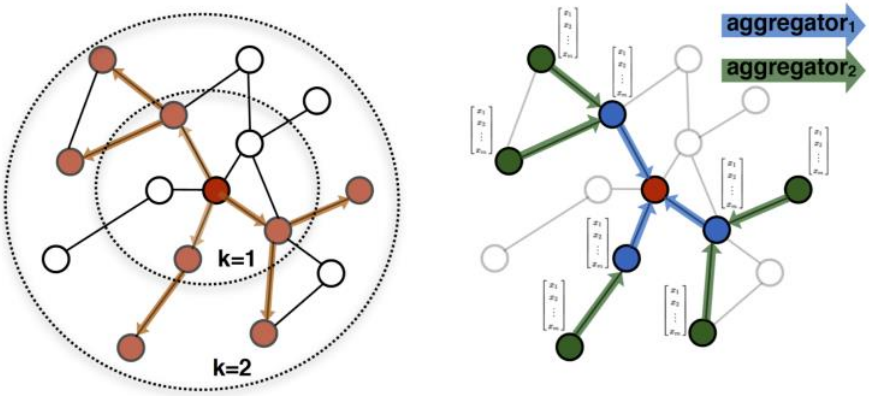
Η συνάρτηση αυτή υπολογίζει τον μέσο όρο των στοιχείων των διανυσμάτων $h_u^{k-1}, \forall\, u \in \mathrm{N}(\upsilon)$.

$$h_u^k \leftarrow \sigma(W \cdot MEAN(\{h_\upsilon^{k-1}\} \cup \{h_u^{k-1}, \forall\, u \in \mathrm{N}(\upsilon)\}) \qquad (3.4)$$

- Τελεστής συγκέντρωσης μεγίστου (Max Pooling aggregator)

Σε αυτήν την περίπτωση τα διανύσματα των γειτονικών κόμβων τροφοδοτούνται ανεξάρτητα το ένα με το άλλο στο νευρωνικό δίκτυο. Αυτό γίνεται με έναν στοιχειακό (element-wise) μετασχηματισμό συσσώρευσης γειτονικής πληροφορίας, τον τελεστή μεγίστου ως εξής:

$$AGGREGATE_k^{pool} = \max(\{\sigma(W_{pool} h_{u_i}^k + b), \forall\, u_i \in \mathrm{N}(v)\} \quad (3.5)$$



Εικόνα 3.7: Αναπαράσταση συλλογής γειτονικής πληροφορίας και εφαρμογής συνάρτησης συγκέντρωσης για την ανανέωση της κορυφής ενός γραφήματος με τη χρήση του GraphSAGE

3.5 Νευρωνικά δίκτυα γραφημάτων προσοχής (Attentional GNN)

Οι μηχανισμοί προσοχής έχουν εδραιωθεί αρκετά σε εργασίες μηχανικής μάθησης που βασίζονται σε ακολουθίες. Στα βαθιά νευρωνικά δίκτυα γραφημάτων, ο μηχανισμός αυτός χρησιμοποιείται για τη διάδοση πληροφορίας σε επόμενα στρώματα.

GAT Layer (Graph Attentional layer)

Η είσοδος ενός graph attentional layer είναι όπως και με τις προηγούμενες αρχιτεκτονικές, μια σειρά από τα χαρακτηριστικά κόμβων $h = \{\vec{h}_1, \vec{h}_2, \dots, \vec{h}_N\}, \vec{h}_i \in R^F$, όπου N ο αριθμός των κόμβων και F ο αριθμός των χαρακτηριστικών κάθε κόμβου. Κάθε στρώμα παράγει μια νέα αναπαράσταση $h' = \{\vec{h'}_1, \vec{h'}_2, \dots, \vec{h'}_N\}, \vec{h'}_i \in R^{F'}$ ως έξοδο. Ας υποθέσουμε έναν πίνακα βαρών $W \in R^{F' x F}$, τον οποίο θα χρησιμοποιήσουμε για να εφαρμόσουμε έναν κοινό γραμμικό συσχετισμό (shared linear transformation) σε κάθε κόμβο. Στη συνέχεια εφαρμόζεται μια μέθοδος γνωστή ως αυτό-προσοχή σε όλους τους

κόμβους, δηλαδή ένας κοινός μηχανισμός $\alpha : R^{F'} x R^{F'} \rightarrow R$. Με αυτόν τον τρόπο υπολογίζονται οι συντελεστές προσοχής:

$$e_{ij} = \alpha\left(W\vec{h}'_i, W\vec{h}'_j\right) \tag{3.6}$$

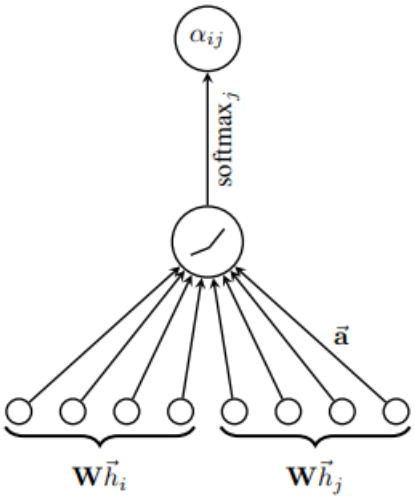
, οι οποίοι επισημαίνουν την επιδραστικότητα των χαρακτηριστικών της κορυφής j στην κορυφή i. Πρακτικά, αυτό σημαίνει πως κάθε κόμβος μπορεί να παρακολουθήσει τη συμπεριφορά κάθε άλλου κόμβου, «βλέποντας» τα χαρακτηριστικά του, απορρίπτοντας όλες τις δομικές πληροφορίες. Επίσης οι συντελεστές προσοχής e_{ij} υπολογίζονται για κόμβους $j \in N_i, όπου\ N_i$ είναι μια γειτονική περιοχή του κόμβου i. Δεδομένης μιας σύγκρισης μεταξύ των συντελεστών ανά τους κόμβους, τους ομαλοποιούμε χρησιμοποιώντας την συνάρτηση softmax :

$$\alpha_{ij} = softmax_j(e_{ij}) = \frac{\exp(e_{ij})}{\sum_{k \in N_i} \exp(e_{ik})} \tag{3.7}$$

Ο μηχανισμός προσοχής μπορεί να θεωρηθεί ένα μονοστρωματικό νευρωνικό δίκτυο εμπρόσθιας τροφοδότησης με παραμέτρους ένα διάνυσμα βαρών $\vec{a} \in R^{2F'}$ και εφαρμογή μιας μη γραμμικής συνάρτησης ενεργοποίησης LeakyReLU (για αρνητικές τιμές η LeakyReLU είναι μια γραμμική συνάρτηση με κλίση -0.2) . Επομένως μπορούμε να εξελίξουμε τον μηχανισμό προσοχής ως εξής:

$$\alpha_{ij} = \frac{\exp\left(LeakyReLU(\vec{a}^T[W\vec{h}'_i || W\vec{h}'_j])\right)}{\sum_{k \in N_i} \exp\left(LeakyReLU(\vec{a}^T[W\vec{h}'_i || W\vec{h}'_k])\right)} \tag{3.8}$$

Στην παραπάνω εξίσωση ο όρος $.^T$ υποδηλώνει την αναστροφή του πίνακα $\vec{a}$ και || συμβολίζει την ένωση των πινάκων βαρών.



Εικόνα 3.8: Δράση μηχανισμού προσοχής με τη χρήση παραμέτρων (βαρών) προσοχής [29]

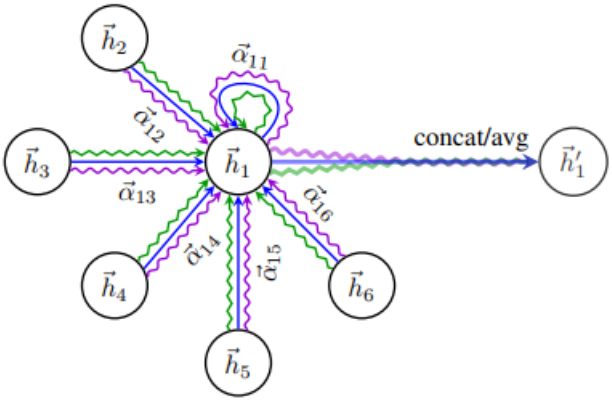
Έχοντας αποκτήσει τους συντελεστές α_{ij}, υπολογίζεται ένας γραμμικός συνδυασμός των χαρακτηριστικών που συνδέονται με αυτούς, για να ληφθεί το διάνυσμα εξόδου h':

$$\vec{h'}_i = \sigma(\sum_{j \in N_i} a_{ij} W \vec{h}_j) \tag{3.9}$$

Για να εξομαλυνθεί η διαδικασία μάθησης με τη μέθοδο της αυτό-προσοχής μπορούμε να εφαρμόσουμε ένα μηχανισμό γνωστό ως πολλαπλά κεφάλια προσοχής (multi-head attention). Ορίζουμε K ανεξάρτητους μηχανισμούς α_{ij} , όπου για τον καθένα εφαρμόζονται οι παραπάνω εξισώσεις και στο τέλος αθροίζονται για να δώσουν την έξοδο του στρώματος:

$$\vec{h'}_i = ||_{k=1}^{K} \sigma(\sum_{j \in N_i} {a^k}_{ij} W^k \vec{h}_j) \tag{3.10}$$

, όπου ${a^k}_{ij}$ είναι οι ομαλοποιημένοι συντελεστές προσοχής υπολογισμένοι από τον μηχανισμό προσοχής a^k και W^k είναι ο πίνακας βαρών γραμμικού μετασχηματισμού των εισόδων [29].



Εικόνα 3.9: Χρήση πολλαπλών κεφαλιών προσοχής σε μια κορυφή [29]

GATv2 layer

Η αρχιτεκτονική αυτή βασίζεται στην προηγούμενη (GAT), επιβάλλοντας μια μικρή αλλαγή στον τρόπο υπολογισμού των συντελεστών προσοχής. Ένα πρόβλημα όσον αφορά το GAT είναι ότι οι τα υπολογισμένα βάρη W και οι μηχανισμοί προσοχής εφαρμόζονται ταυτόχρονα και μπορούν με αυτό τον τρόπο να συμπτυχθούν σε ένα ενιαίο γραμμικό στρώμα. Επομένως θεωρείται ορθό να εφαρμοστεί ο μηχανισμός προσοχής a μετά το μη γραμμικό υπολογισμό και η εφαρμογή του πίνακα βαρών μετά τη συνένωση των κόμβων και επομένως να οριστεί ο συντελεστής προσοχής ως [30]:

$$e_{ij} = \left(\vec{a}^T LeakyReLU(W \cdot [\overrightarrow{h'}_i || \vec{h}'_j])\right) \tag{3.11}$$

3.6 Νευρωνικά δίκτυα προώθησης μηνυμάτων (Message Passing Neural Networks, MPNN)

Τα συνελικτικά όσο και τα δίκτυα προσοχής μπορούν να ενταχθούν σε μια πιο γενική θεώρηση, αυτή των νευρωνικών δικτύων προώθησης μηνυμάτων (MPNN). Τα μοντέλα

MPNN εξάγουν τα γενικά χαρακτηριστικά από πολλά κλασσικά μοντέλα [31]. Ας υποθέσουμε ένα μη κατευθυνόμενο γράφημα G με x_n χαρακτηριστικά κόμβων (node features) και e_{vw} χαρακτηριστικά ακμών (edge features). Η εμπρόσθια τροφοδότηση του νευρωνικού δικτύου περιλαμβάνει δύο φάσεις, την προώθηση μηνύματος και το στάδιο ανάγνωσης. Η χρονική διάρκεια της μετάβασης του μηνύματος διαρκεί για T χρονικά βήματα, κατά τη διάρκεια του οποίου εφαρμόζονται δύο συναρτήσεις. Η πρώτη αφορά τη συνάρτηση ανταλλαγής μηνύματος M_t ενώ η δεύτερη την αναβάθμιση-ανανέωση των κορυφών U_t. Κατά τη διάρκεια του σταδίου προώθησης μηνύματος οι κρυφές αναπαραστάσεις των κόμβων h_v^t ανανεώνονται σε συνάρτηση με το τρέχων μήνυμα m_v^{t+1} ως εξής :

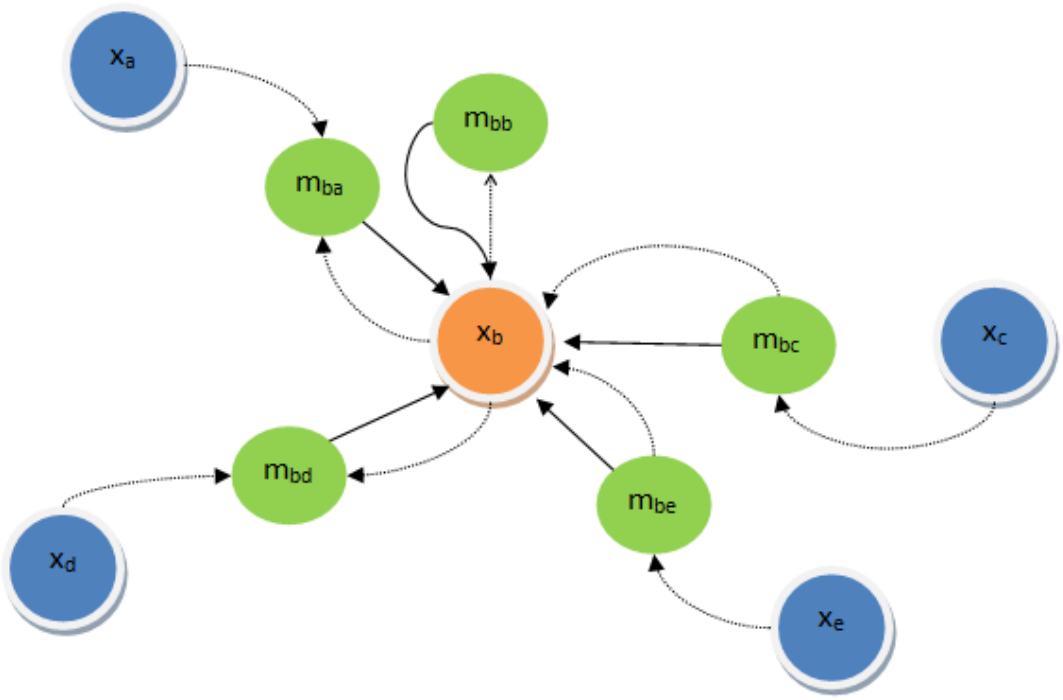
$$m_v{}^{t+1} = \sum_{w \in N(u)} M_t(h_v^t, h_w^t, e_{vw}) \tag{3.12}$$

$$h_v^{t+1} = U_t(h_v^t, m_v^{t+1}) \tag{3.13}$$

, με $N(v)$ να είναι το άθροισμα των κόμβων που είναι γειτονικοί στον κόμβο v. Μόλις τελειώσει αυτή η φάση, αρχίζει το στάδιο ανάγνωσης. Συγκεκριμένα, υπολογίζεται ένας πίνακας χαρακτηριστικών για το συνολικό γράφημα κάνοντας χρήση μιας συνάρτησης ανάγνωσης R ως εξής:

$$\hat{y} = R(\{h_v^T \mid u \in G\}) \tag{3.14}$$

Είναι σημαντικό να αναφερθεί πως οι συναρτήσεις M_t, U_t και R είναι διαφοροποιήσιμες και εκπαιδεύσιμες. Η συνάρτηση ανάγνωσης R δρα σε ένα σύνολο αναπαραστάσεων κόμβων και γενικά, αυτή η διαδικασία προώθησης μηνύματος μπορεί να εφαρμοσθεί σε πολλές ήδη υπάρχουσες αρχιτεκτονικές νευρωνικών δικτύων γραφημάτων αρκεί να οριστούν με σωστό τρόπο οι συναρτήσεις M_t , U_t και R.



Εικόνα 3.10: Παράδειγμα προώθησης μηνύματος από μια κορυφή στις γειτονικές κορυφές της και αντίστροφα.

Κεφάλαιο 4: Συμβολοσειρά SMILES

Το SMILES (Simplified Molecular Input Line Entry System) είναι ένα σύστημα χημικής σημειογραφίας που βασίζεται στη θεωρία γραφημάτων και είναι σχεδιασμένο για τη σύγχρονη επεξεργασία της χημικής πληροφορίας μέσω ενός υπολογιστή [32]. Δημιουργήθηκε με σκοπό να πετύχει δύο βασικά πράγματα. Πρώτον, να μπορεί να περιγράψει με τη μορφή γραφήματος τη δομή μιας χημικής ένωσης όσο τον δυνατόν καλύτερα. Δηλαδή πέρα από τα άτομα και τους χημικούς δεσμούς να είναι σε θέση να εξάγει παραπάνω πληροφορία από το μόριο. Δεύτερον, ήταν απαραίτητη μια εύκολα κατανοητή προδιαγραφή δομής με απλούς κανόνες. Η σημειογραφία SMILES ουσιαστικά είναι μια συμβολοσειρά που τελειώνει με την ύπαρξη κενού. Τα άτομα υδρογόνου μπορούν να παραλειφθούν αλλά και να συμπεριληφθούν ανάλογα με το πλαίσιο της χρήσης των SMILES. Οι αρωματικές δομές περιγράφονται απευθείας κατά προτίμηση με τη μορφή Kekule [33]. Παρακάτω περιγράφονται οι κανόνες σύμφωνα με τους οποίους μπορεί να εξαχθεί μια ψηφιακή μοριακή δομή με την είσοδο ενός SMILES.

4.1 Άτομα

Κάθε άτομο πλην του υδρογόνου αναγράφεται ως το ατομικό του σύμβολο ανάμεσα σε αγκύλες ([]). Τα χημικά οργανικά στοιχεία B, C, N, O, P, S, F, Cl, Br, I μπορούν να εξαιρεθούν από τις αγκύλες στην περίπτωση που ο αριθμός των συγγενικών υδρογόνων ισούται με το χαμηλότερο χημικό σθένος. Εάν το σύμβολο περιέχει δύο γράμματα, το δεύτερο γράφεται σε μικρά γράμματα. Επίσης σε μικρά γράμματα αναγράφονται και τα άτομα που βρίσκονται μέσα σε αρωματικό δακτύλιο. Μερικά παραδείγματα είναι τα εξής:

Χημικό άτομο-μόριο	Smiles
Μεθάνιο (CH_4)	C
Αμμωνία (NH_3)	N
Νερό (H_2O)	O
Υδροχλώριο (HCl)	Cl

Χρυσός (Au)	[Au]

Πίνακας 5.1 Παραδείγματα smiles απλών χημικών ατόμων-μορίων

Τα γειτονικά υδρογόνα και η φόρτιση πάντοτε βρίσκονται εντός αγκυλών. Ο αριθμός των υδρογόνων αναγράφεται με το σύμβολο H και δίπλα τον αριθμό. Με τον ίδιο ακριβώς τρόπο γράφεται και το φορτίο [+, -] ακολουθημένο από τον αριθμό των φορτισμένων σωματιδίων. Σε περίπτωση έλλειψης ο αριθμός αυτός θεωρείται μηδέν και για τις δύο περιπτώσεις. Μερικά παραδείγματα είναι :

Χημικά φορτισμένο άτομο-μόριο	Smiles
Πρωτόνιο	[H+]
Ανιόν υδροξυλίου	[OH-]
Κατιόν σιδήρου [II]	[Fe+2]

Πίνακας 5.2 Παραδείγματα smiles χημικά φορτισμένων ατόμων-μορίων

4.2 Χημικοί Δεσμοί

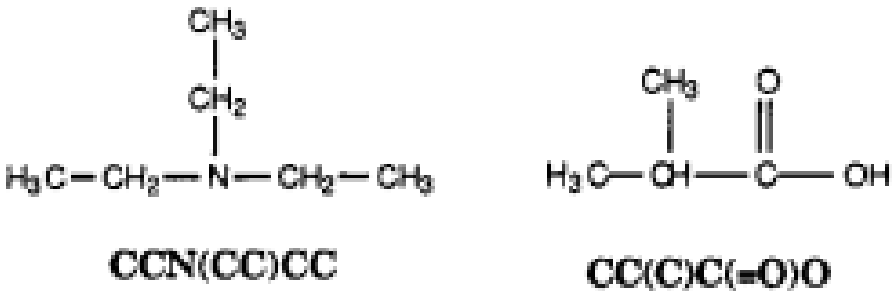
Το είδος του χημικού δεσμού (μονός, διπλός, τριπλός, αρωματικός) δηλώνεται με τη χρήση των συμβόλων - , == , # και :, αντίστοιχα. Συχνά ωστόσο οι μονοί και οι αρωματικοί δεσμοί παραλείπονται. Ομοίως παρατίθενται μερικά παραδείγματα:

Χημικό μόριο	Smiles
Αιθάνιο	CC
Αιθανόλη	CCO
Διοξείδιο του άνθρακα	O==C==O
Υδροκυάνιο	C#N

Πίνακας 5.3 4 Παραδείγματα απεικόνισης χημικών δεσμών smiles

4.3 Διακλαδώσεις

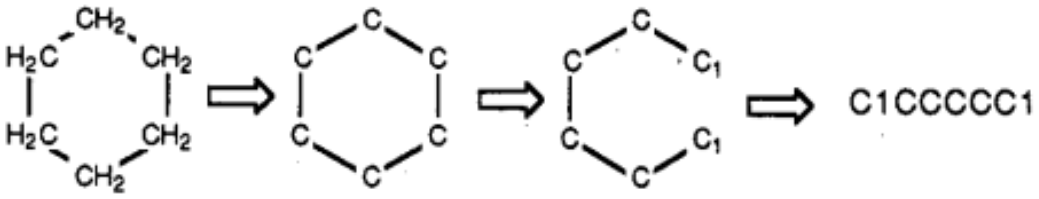
Τα άτομα που βρίσκονται σε διακλάδωση δηλώνονται μέσα σε παρενθέσεις. Για παράδειγμα :



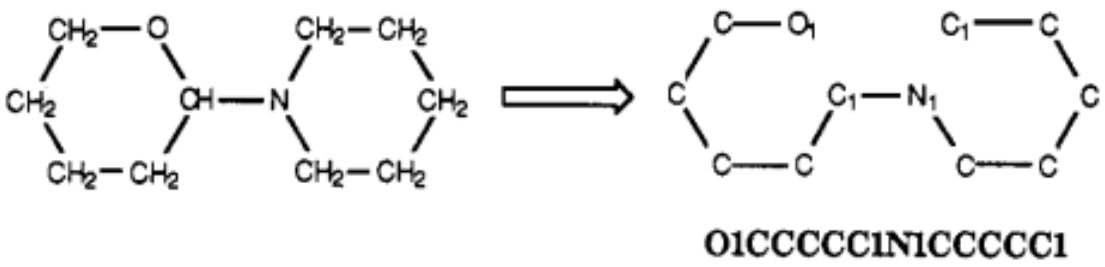
Εικόνα 4.1 Παράδειγμα γραφής SMILES τριεθυλαμίνης και ισοβουτυλικού οξέος [32]

4.4 Κυκλικές δομές

Οι κυκλικές δομές αναπαρίστανται δημιουργώντας σχάση ενός απλού ή αρωματικού δεσμού σε κάθε δακτύλιο. Με αυτόν τον τρόπο δημιουργείται ένα μη κυκλικό γράφημα και στην συνέχεια εφαρμόζονται όλοι οι παραπάνω κανόνες για την γραφή του μορίου. Είναι σημαντικό να αναφερθεί ότι το ίδιο μόριο μπορεί να έχει πολλές έγκυρες γραφές SMILES. Παραδείγματα σχάσης δακτυλίων αποτελούν τα παρακάτω :



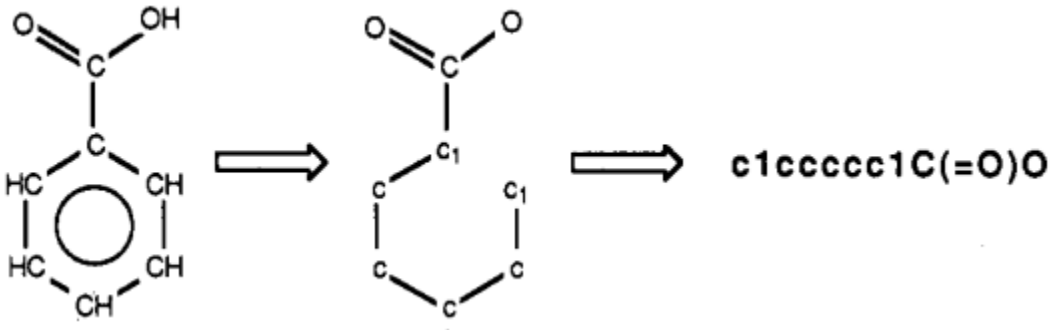
Εικόνα 4.2: SMILES κυκλοεξανίου [32]



Εικόνα 4.3: Παράδειγμα SMILES μορίου με δύο κυκλικές δομές [32]

4.5 Αρωματικότητα

Ο διαχωρισμός των αρωματικών δομών επιτυγχάνεται γράφοντας τα αρωματικά άτομα σε πεζά γράμματα. Επίσης, το σύστημα SMILES μπορεί να εντοπίζει αυτόματα αρωματικές δομές ακόμα και αν αναγράφονται στην είσοδο με κάποιον διαφορετικό τρόπο (μη πεζά γράμματα). Παράδειγμα αποτελεί το βενζοϊκό οξύ :



Εικόνα 4.4: SMILES βενζοϊκού οξέος [32]

Κεφάλαιο 5: Δεδομένα

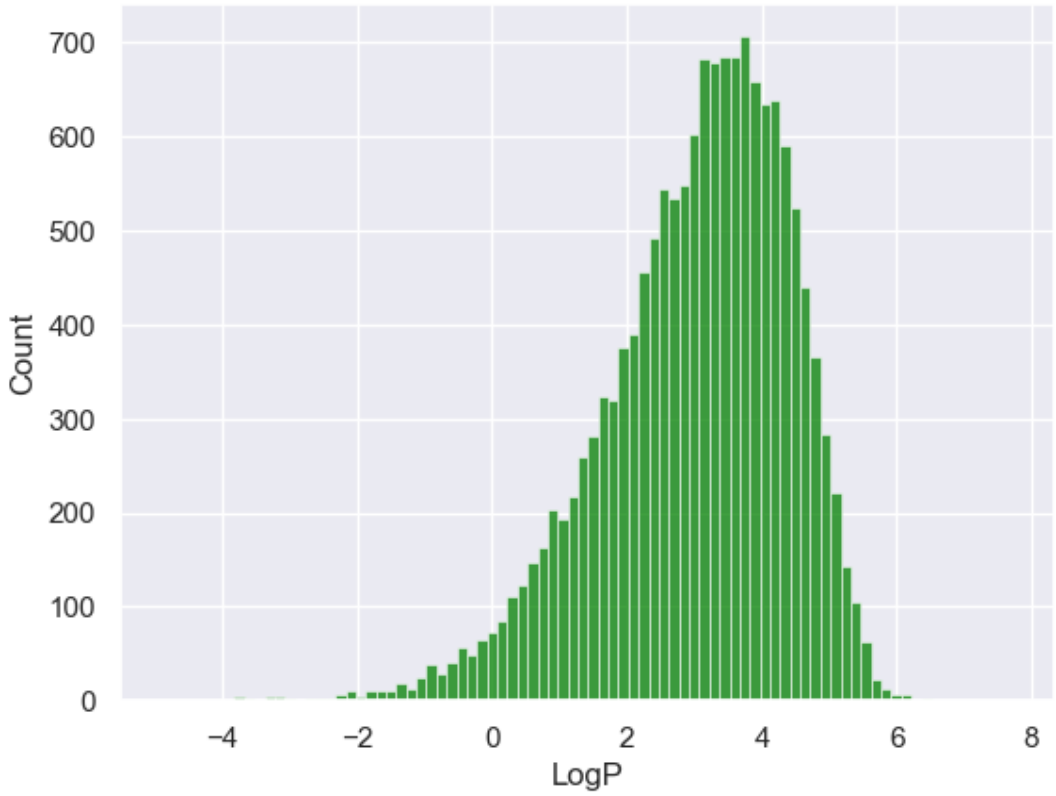
Στόχος της παρούσας διπλωματικής εργασίας είναι η δημιουργία μοντέλων που θα προβλέπουν μοριακές ιδιότητες σε μικρά οργανικά μόρια. Οι ιδιότητες αυτές αποτελούν την διαλυτότητα, την πολικότητα και την συνθετική προσβασιμότητα. Τα δεδομένα που χρησιμοποιήθηκαν πάρθηκαν από ένα επιστημονικό άρθρο [34], στο οποίο έγινε χρήση 500.000 δεδομένων που προήλθαν από τη βάση δεδομένων ZINC [35]. Συγκεκριμένα, στην ZINC τα μόρια μπορούν να παρθούν υπό τη μορφή SMILES και σε κάθε ένα από αυτά έχουν μετρηθεί πειραματικά ή με υπολογιστικές μεθόδους οι τρεις παραπάνω ιδιότητες. Στην εργασία αυτή χρησιμοποιήθηκε ένα υποσύνολο των 500.000 που αποτελείται από 15.000 δεδομένα.

5.1 Διαλυτότητα - λογαριθμικός συντελεστής συμμετοχής LogP

Η διαλυτότητα εκφράζεται από τον συντελεστή κατανομής P. Ο συντελεστής κατανομής P περιγράφει την τάση μια ουδέτερης (αφόρτιστης) ένωσης να διαλύεται σε ένα μη αναμίξιμο διφασικό σύστημα λιπιδίων και στο νερό. Οι ουσίες που διαλύονται περισσότερο στο νερό ονομάζονται υδρόφιλες ενώ αυτές που διαλύονται στα λιπίδια ονομάζονται λιπόφιλες. Ο συντελεστής P αποτελεί μια σημαντική μέτρηση για την φύση μιας ουσίας και κατά επέκταση της συμπεριφοράς της χημικής ένωσης όταν εκτίθεται σε διαφορετικά χημικά περιβάλλοντα. Πιο συγκεκριμένα, αποτελεί ένδειξη του κατά πόσο μια ουσία μπορεί να απορροφηθεί από φυτά, ζώα ανθρώπους ή να παρασυρθεί από υδατικά περιβάλλοντα. Λόγω των πολλών εφαρμογών του συντελεστή, αυτός χρησιμοποιείται για χάριν ευκολίας ως $log_{10}P$. Σε γενικές γραμμές ο $logP$ ερμηνεύεται ως εξής [36] :

- $logP < 0$ → Υψηλότερη συγγένεια υδατικής φάσης (αυξημένη υδροφιλικότητα)
- $logP = 0$ → Ισότιμη κατανομή μεταξύ λιπιδικής και υδατικής φάσης
- $logP > 0$ → Υψηλότερη συγγένεια στην λιπιδική φάση

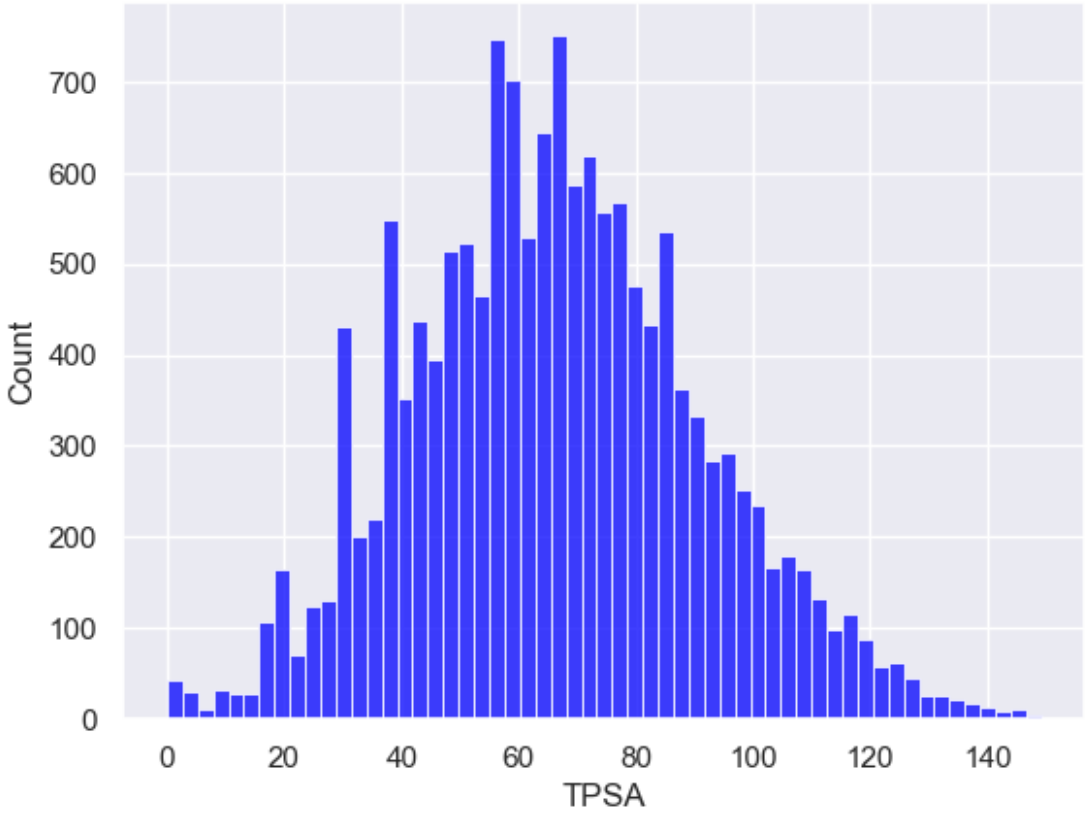
Ο συντελεστής βρίσκει ευρεία εφαρμογή στην ερεύνα φαρμακευτικών εταιριών με σκοπό την κατανόηση της συμπεριφοράς των φαρμακευτικών ενώσεων στο ανθρώπινο σώμα. Αυτό, γιατί η διαλυτότητα είναι ένας πολύ καθοριστικός παράγοντας για την απορρόφηση μιας ένωσης και την κατανομή της στο σώμα, όσο και για την διείσδυση σε ζωτικές μεμβράνες, βιολογικά εμπόδια (barriers), την μελέτη του μεταβολισμού και την απέκκριση.



Εικόνα 5.1 Κατανομή τιμής LogP στο σύνολο των δεδομένων

5.2 Πολικότητα – Τοπολογική πολική επιφάνεια (Topological polar surface area, TPSA)

Η Πολικότητα μιας χημικής ένωσης ερμηνεύεται από την Τοπολογική πολική επιφάνεια (TPSA). Αυτή ορίζεται ως το άθροισμα των συνεισφορών στη μοριακή επιφάνεια πολικών ατόμων όπως το οξυγόνο, το άζωτο και τα συνδεδεμένα υδρογόνα τους. Ο υπολογισμός της πολικής επιφάνειας είναι περίπλοκος. Εξαρτάται από την κατάλληλη αναπαράσταση της τρισδιάστατης μοριακής γεωμετρίας ή το σύνολο γεωμετριών για κάθε μόριο που μελετάται. Αυτή η πολυπλοκότητα ξεπεράστηκε με την ανάπτυξη μιας γρήγορης μεθόδου προσθετικού θραύσματος (additive fragment method). Αυτό συντέλεσε στον υπολογισμό αυτής της ιδιότητας για χρήση της σε εικονική προβολή (virtual screening) πολλών δεδομένων (μορίων). Ο συντελεστής TPSA έχει γίνει ιδιαίτερα ελκυστικός στην ιατρική χημεία αλλά και για την πρόβλεψη ιδιοτήτων ADME, όπως για παράδειγμα της τάσης διέλευσης φραγμού αίματος-εγκεφάλου [37].

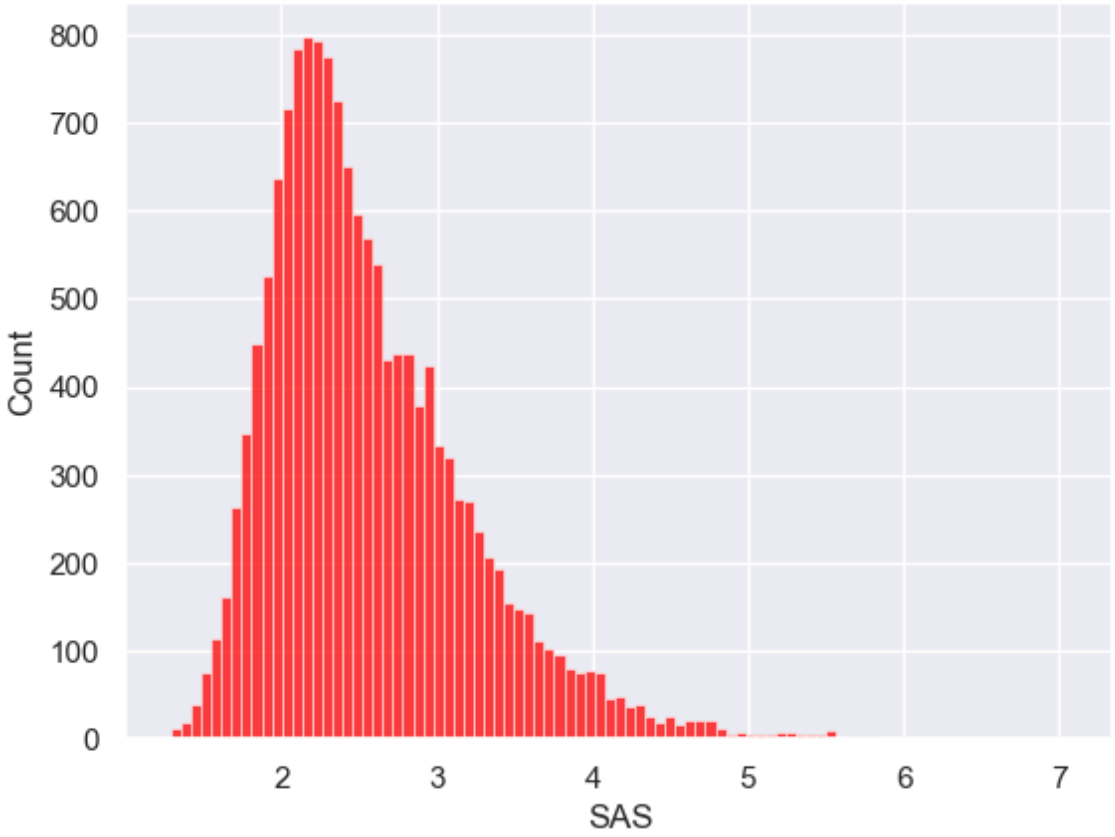


Εικόνα 5.2 Κατανομή τιμών TPSA στο σύνολο των δεδομένων

5.3 Συνθετική προσβασιμότητα - Δείκτης συνθετικής προσβασιμότητας (Synthetic accessibility score, SAS)

Η ευκολία σύνθεσης ενός χημικού μορίου περιγράφεται από τον δείκτη SAS. Αυτός έχει σχεδιαστεί με σκοπό να ορίζει την συνθετική προσβασιμότητα φαρμακευτικών ουσιών για εικονική προβολή. Υπολογίζεται ως το άθροισμα των σκορ των θραυσμάτων (fragment score) και των ποινών πολυπλοκότητας (complexity penalty). Το fragment score αποτελεί το άθροισμα των συνεισφορών όλων των θραυσμάτων στο μόριο διαιρούμενο με τον συνολικό αριθμό των θραυσμάτων σε αυτό. Το complexity score είναι ένας αριθμός που χαρακτηρίζει την ύπαρξη σύνθετων δομικών χαρακτηριστικών στο μόριο. Αυτές οι δομές περιλαμβάνουν την παρουσία δακτυλίων, στερεόκυκλων

(stereocenters), μακρόκυκλων (macrocycles) ή το μέγεθος του μορίου. Οι τιμές του δείκτη αυτού κυμαίνονται από 1 (εύκολη σύνθεση) έως 10 (δύσκολη σύνθεση) [38],[39].



Εικόνα 5.3 Κατανομή τιμών SAS στο σύνολο των δεδομένων

Κεφάλαιο 6: Μοντελοποίηση και παρουσίαση αποτελεσμάτων

6.1 Μετρικές αξιολόγησης

Για την αξιολόγηση της προβλεπτικής ικανότητας και της αξιοπιστίας των μοντέλων μηχανικής μάθησης χρησιμοποιούνται ποικίλες μετρικές αξιολόγησης. Όλα τα προβλήματα που παρουσιάζονται και επιλύονται στην εργασία αφορούν σε ανάλυση παλινδρόμησης (regression), δηλαδή την πρόβλεψη μιας συνεχούς τιμής και όχι ταξινόμηση σε κατηγορίες. Εδώ είναι θεμιτό να γίνει μια διαφοροποίηση μεταξύ της συνάρτησης κόστους και των μέτρων αξιολόγησης. Η βασική διαφορά έγκειται στο γεγονός ότι η αντικειμενική συνάρτηση κόστους χρησιμοποιείται για να μετρήσει την απόδοση του μοντέλου κατά τη διάρκεια της εκπαίδευσης, ή οποία χρησιμοποιείται και για την εφαρμογή του αλγορίθμου της οπισθοδιάδοσης. Οι μετρικές αξιολογούν την αποδοτικότητα του μοντέλου αφού τελειώσει η εκπαίδευση. Βεβαίως όμως, μια συνάρτηση κόστους μπορεί να χρησιμοποιηθεί και ως ένα μέτρο αξιολόγησης στο τελικό μοντέλο [40].

Συντελεστής συσχέτισης R^2

Για ένα μοντέλο γραμμικής παλινδρόμησης με τιμές πρόβλεψης $\hat{p}_i$, πραγματικές τιμές (targets) p_i και $\bar{p}$ ο μέσος όρος αυτών, ορίζουμε τρεις μεταβλητές ως εξής :

$$RSS = \sum_{i=1}^{N} (\hat{p}_i - p_i)^2 \tag{6.1}$$

$$ESS = \sum_{i=1}^{N} (\hat{p}_i - \bar{p})^2 \tag{6.2}$$

$$TSS = \sum_{i=1}^{N} (p_i - \bar{p})^2 \tag{6.3}$$

, όπου RSS είναι υπολειπόμενο άθροισμα των τετραγώνων, ESS το επεξηγημένο άθροισμα τετραγώνων και TSS το συνολικό άθροισμα τετραγώνων:

$$TSS = ESS + RSS \tag{6.4}$$

Για ένα γραμμικό μοντέλο ο συντελεστής συσχέτισης R^2 ορίζεται ως :

$$R^2 = 1 - \frac{RSS}{TSS} = \frac{ESS}{TSS} = 1 - \frac{\sum_{i=1}^{N}(\hat{p}_i - p_i)^2}{\sum_{i=1}^{N}(p_i - \bar{p})^2} \quad (6.5)$$

Από την παραπάνω εξίσωση, συμπεραίνουμε ότι ο συντελεστής συσχέτισης αποτελεί μια ένδειξη της διακύμανσης των προβλεπόμενων τιμών γύρω από τις πραγματικές τιμές. Όσο υψηλότερος είναι τόσο καλύτερη είναι και η προβλεπτική ικανότητα του μοντέλου με αυτόν να κυμαίνεται συνήθως μεταξύ 0 και 1.

Μέσο τετραγωνικό σφάλμα MSE

Το μέσο τετραγωνικό σφάλμα, όπως περιεγράφηκε στο κεφάλαιο 1, μπορεί να αποτελέσει μια συνάρτηση κόστους. Επιπλέον μπορεί να χρησιμοποιηθεί ως μετρική αξιολόγησης στα δεδομένα εξέτασης.

6.2 Δημιουργία δεδομένων εισόδου για νευρωνικά δίκτυα γραφημάτων

Η είσοδος για κάθε μόριο σε όλα τα μοντέλα νευρωνικών δικτύων γράφων που χρησιμοποιήθηκαν είναι ο πίνακας γειτνίασης A και ο πίνακας χαρακτηριστικών κορυφών X. Ο πίνακας χαρακτηριστικών ακμών δεν χρησιμοποιήθηκε σε καμία περίπτωση. Αυτό διότι θα αυξανόταν η υπολογιστική πολυπλοκότητα και ταυτόχρονα οι περισσότερες πληροφορίες των δεσμών εισάγονται αυτόματα από αυτές των κορυφών. Για παράδειγμα η πληροφορία του είδους των δεσμών, δίνεται εμμέσως με τη χρήση του πίνακα χαρακτηριστικών κορυφής X [34] .

Η κατασκευή των πινάκων A, X γίνεται με τη χρήση της χημικής βιβλιοθήκης rdkit της Python [41]. Συγκεκριμένα, για την κατασκευή του πίνακα χαρακτηριστικών κορυφής, είναι απαραίτητο οι κατηγορικές μεταβλητές εισόδου του προβλήματος (χημικά άτομα, αριθμός γειτονικών υδρογόνων) να κωδικοποιηθούν ως δυαδικές μεταβλητές (0/1) για την εκπαίδευση του μοντέλου. Αυτό επιτυγχάνεται μέσω της one hot κωδικοποίησης (one-hot encoding). Ας υποθέσουμε πως ένα μόριο περιέχει τα εξής χημικά άτομα {C,O,other} (εξαιρούνται τα υδρογόνα λόγω των SMILES). Αυτή η πληροφορία μπορεί να

κωδικοποιηθεί ως ένα μοναδικό διάνυσμα που προκύπτει από τρία πιθανά για κάθε κόμβο (άτομο) ενός γραφήματος δηλαδή ως [0,0,1] , [0,1,0] ,[1,0,0]. Γενικά, όλα τα στοιχεία του κάθε διανύσματος είναι 0 εκτός από ένα το οποίο είναι μονάδα και υποδηλώνει την εκάστοτε χημική πληροφορία (Στην περίπτωση μας, το χημικό σύμβολο). Το μήκος του διανύσματος ισούται με το πλήθος των κατηγορικών μεταβλητών του εκάστοτε χημικού χαρακτηριστικού [42], [43]. Τα χαρακτηριστικά των κόμβων (ατόμων) του συνόλου δεδομένων του προβλήματος παρουσιάζονται παρακάτω:

Χαρακτηριστικό ατόμου (atom feature)	Πιθανές τιμές στο σύνολο δεδομένων	Μήκος διανύσματος
Χημικό Σύμβολο	$['C',' O',' N',' F',' Br',' Cl','S'](one\ hot)$	7
Βαθμός ατόμου (Γειτονικά άτομα)	$[1,2,3,4](one\ hot)$	4
Γειτονικά υδρογόνα	$[0,1,2,3](one\ hot)$	4
Τυπικό φορτίο	$[-1,0,1](one\ hot)$	3
Σθένος (Implicit)	$[0,1,2,3](one\ hot)$	4
Σθένος (Explicit)	$[1,2,3,4,6](one\ hot)$	5
Υβριδισμός	$['SP',' SP2',' SP3'](one\ hot)$	3
Αρωματικότητα	[0/1](ακέραιος)	1

Πίνακας 6.1 Εισαγόμενα χαρακτηριστικά κορυφών (node features) στο νευρωνικό δίκτυο και κωδικοποίηση τους

6.3 Ανάπτυξη μοντέλων

Τα μοντέλα εκπαιδεύτηκαν με τη χρήση των βιβλιοθηκών pytorch [44] και pytorch geometric [45] που είναι οι πλέον πιο διαδεδομένες βιβλιοθήκες για την ανάπτυξη σύγχρονων νευρωνικών δικτύων. Τα 15.000 δεδομένα που χρησιμοποιήθηκαν διαχωρίστηκαν σε δεδομένα εκπαίδευσης-επαλήθευσης και εξέτασης με τυχαίο τρόπο και αναλογία 0.75:0.15:0.10 κάνοντας χρήση της βιβλιοθήκης sklearn [46]. Αυτό σημαίνει πως 11,250 δεδομένα χρησιμοποιούνται για την εκπαίδευση σε συνδυασμό με 2,250 δεδομένα τα οποία επικυρώνουν το μοντέλο μετά από κάθε εποχή. Μόλις τα μοντέλα εκπαιδευτούν, εξετάζονται ως προς μετρικές αξιολόγησης σε ένα σύνολο 1500

άγνωστων δεδομένων. Είναι σημαντικό να αναφερθεί ότι η εκπαίδευση για την πρόβλεψη των τριών ιδιοτήτων (LogP , TPSA, SAS) πραγματοποιήθηκε ταυτόχρονα για κάθε μοντέλο. Τα μοντέλα που εκπαιδεύτηκαν βασίστηκαν στις εξής αρχιτεκτονικές που προαναφέρθηκαν στο Κεφάλαιο 3 (GCN, GraphSAGE, GAT, GATv2). Παρόλο που οι αρχιτεκτονικές διαφέρουν μεταξύ τους, οι περισσότερες υπερπαράμετροι του μοντέλου είναι κοινές για όλες. Αυτοί είναι:

- Συνάρτηση κόστους: Ως συνάρτηση κόστους για τα μοντέλα που αναπτύχθηκαν επιλέχθηκε το μέσο απόλυτο σφάλμα (L1). Αυτή η επιλογή έγινε κυρίως διότι λόγω του μεγάλου συνόλου δεδομένων, δεν θέλαμε ένα μεμονωμένο μεγάλο σφάλμα να επηρεάζει σε υψηλό βαθμό της συνάρτηση κόστους όπως για παράδειγμα θα έκαναν οι συναρτήσεις κόστους MSE , RMSE. Επίσης, η βιβλιογραφία έχει δείξει πως είναι μια πολύ καλή συνάρτηση για το συγκεκριμένο πρόβλημα [34].

- Ρυθμός μάθησης: Ένας καλός ρυθμός μάθησης για το συγκεκριμένο πρόβλημα πρόβλεψης συνεχούς τιμής είναι 0.001 [34].

- Αλγόριθμος βελτιστοποίησης: Ο πλέον διαδεδομένος αλγόριθμος βελτιστοποίησης, ο οποίος φαίνεται να δίνει καλά αποτελέσματα για το σύνολο δεδομένων μας είναι ο Adam [34].

- Συνάρτηση ενεργοποίησης: Οι καλύτερες επιλογές όσον αφορά την επιλογή συνάρτησης ενεργοποίησης είναι η ReLU, η leaky ReLU και άλλες παραλλαγές της. Στο παρών πρόβλημα έγινε χρήση της ReLU.

- Συνάρτηση dropout: Εξετάστηκε η μη χρήση συνάρτησης dropout αλλά και η χρήση με πιθανότητα ίση με 0.1.

- Αριθμός κρυφών στρωμάτων: Τα κρυφά στρώματα που εξετάζονται στην παρούσα εργασία ποικίλουν από 2 έως 6. Ως ανώτατο όριο ορίστηκαν τα 6 διότι για τα αρκετά υπολογιστικά απαιτητικά μοντέλα προσοχής (graph attention), ο χρόνος εκπαίδευσης αυξάνεται δραματικά με την περαιτέρω αύξηση στρωμάτων.

- Αριθμός νευρώνων κρυφών στρωμάτων : Σε κάθε κρυφό στρώμα, χρησιμοποιείται ο ίδιος αριθμός κρυφών νευρώνων. Αυτοί ορίστηκαν στους 30 και αυτό για να μπορεί στο τέλος να υπάρξει μια ισότιμη σύγκριση των μοντέλων.

- Μέγεθος παρτίδας δεδομένων εκπαίδευσης και επαλήθευσης: Για την εισαγωγή των δεδομένων στο πρόβλημα, δημιουργήθηκαν φορτωτές δεδομένων εκπαίδευσης

(train loaders) και επικύρωσης (validation loaders) με μέγεθος παρτίδας 100 στην κάθε περίπτωση.

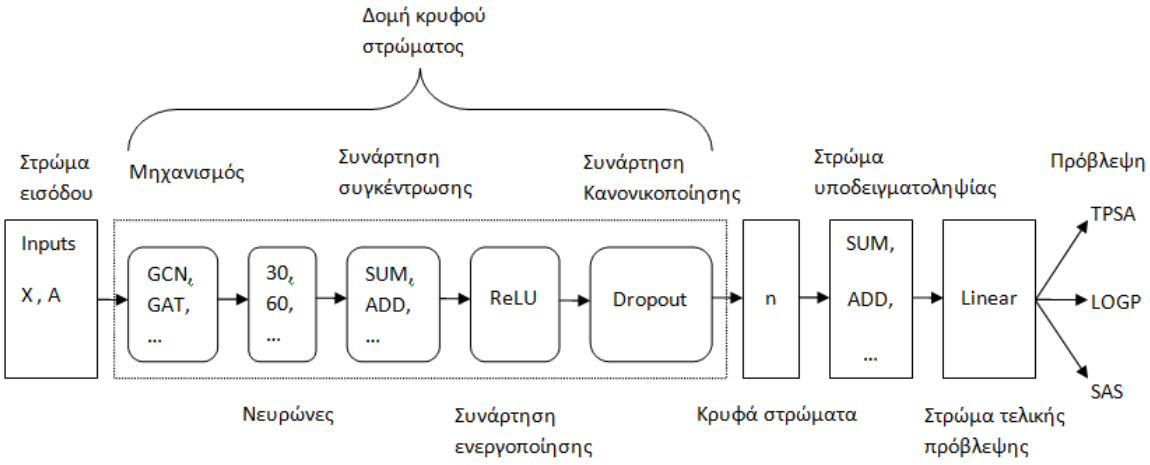
- Στρώμα υποδειγματοληψίας : Μετά από το τελευταίο κρυφό στρώμα εφαρμόζεται ένα στρώμα υποδειγματοληψίας για την ελάττωση της πληροφορίας των κορυφών πριν την τελική πρόβλεψη. Δοκιμάστηκαν τα στρώματα μέσου όρου, αθροίσματος και μεγίστου.

- Στρώμα τελικής πρόβλεψης : Χρήση μια απλής γραμμικής συνάρτησης με σκοπό την εξαγωγή της τελικής πρόβλεψης βάση του στρώματος υποδειγματοληψίας.

Υπερπαράμετρος	Τιμές
Μέγεθος παρτίδων	100
Συνάρτηση ενεργοποίησης	Relu
Αντικειμενική συνάρτηση κόστους	Μέσο απόλυτο σφάλμα (L1 Loss)
Ρυθμός μάθησης	10^{-3}
Αριθμός κρυφών στρωμάτων	{2,3,4,5,6}
Αριθμός κρυφών νευρώνων	30
Συνάρτηση dropout	{False, True (p=0.1)}
Στρώμα υποδειγματοληψίας	{mean, sum, max}
Στρώμα τελικής πρόβλεψης	Γραμμικό στρώμα (Linear layer)
Κεφάλαια προσοχής (GAT)	8
Συνάρτηση συγκέντρωσης (SAGE)	{mean, max}

Πίνακας 6.2 Εύρος τιμών υπερπαραμέτρων νευρωνικών δικτύων

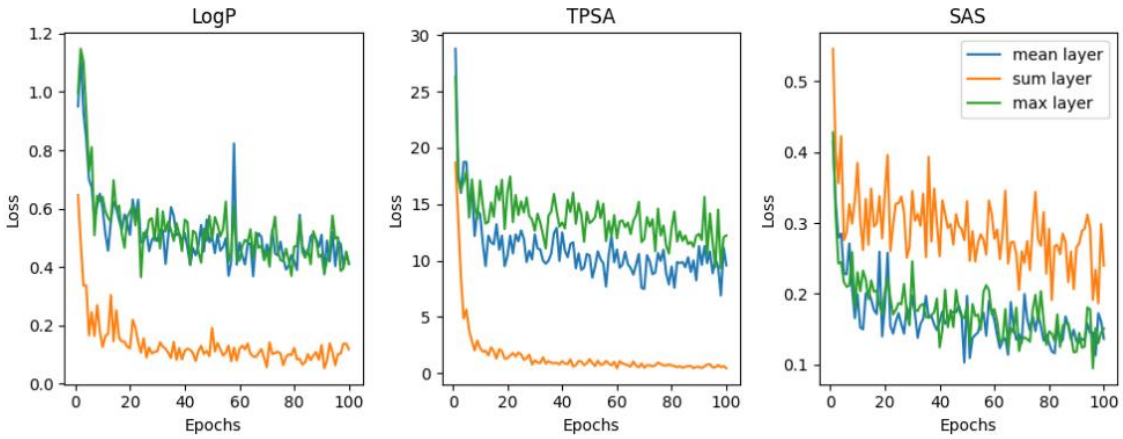
Κάθε μοντέλο βελτιστοποιήθηκε ως προς τα κρυφά στρώματα, κρατώντας σταθερούς τους κρυφούς νευρώνες. Στο παρακάτω σχήμα δίνεται μια ενδεικτική αρχιτεκτονική για όλα τα μοντέλα που αναπτύχθηκαν



Εικόνα 6.1 Δομή νευρωνικών δικτύων γραφημάτων

6.3.1 Αποτελέσματα GCN

Ξεκινώντας την βελτιστοποίηση των μοντέλων, αρχικά έγινε μια δοκιμή ως προς τα τελικά στρώματα υποδειγματοληψίας. Δοκιμάστηκε στρώμα μέσου όρου, αθροίσματος και μεγίστου χωρίς την χρήση της συνάρτησης dropout για τα κρυφά στρώματα. Οι εποχές-επαναλήψεις για όλα τα μοντέλα ορίστηκαν στις 100. Η συνάρτηση κόστους περιλαμβάνει τα μέσα απόλυτα σφάλματα ανά επανάληψη για κάθε προβλεπόμενη ιδιότητα στα δεδομένα εκπαίδευσης ξεχωριστά.



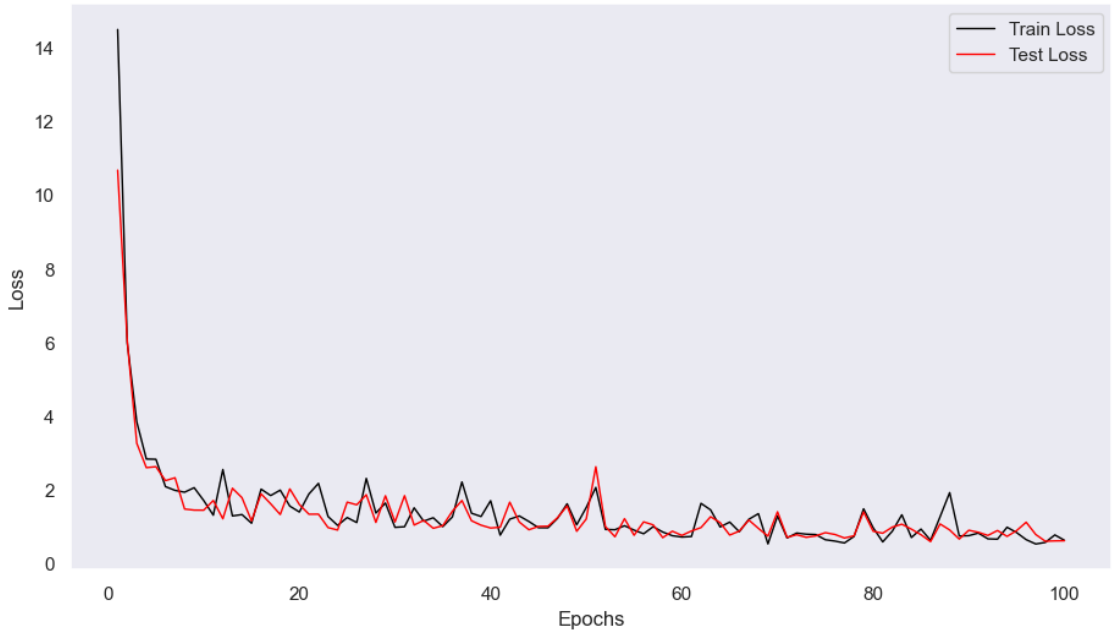
Εικόνα 6.2 Διαγράμματα κόστους εκπαίδευσης – εποχών για κάθε μια ιδιότητα χρησιμοποιώντας στρώματα υποδειγματοληψίας μέσου όρου, αθροίσματος και μεγίστου

Σύμφωνα με τα παραπάνω διαγράμματα, παρατηρείται πως το στρώμα αθροίσματος λειτουργεί πολύ καλύτερα για την εκπαίδευση των ιδιοτήτων LogP και TPSA. Αντιθέτως η χρήση του για την εκπαίδευση της ιδιότητας SAS είναι χειρότερη και χρησιμοποιώντας στρώμα μεγίστου ή μέσου όρου επιτυγχάνεται καλύτερη βελτιστοποίηση. Αυτά τα δύο στρώματα φαίνεται πως ελαχιστοποιούν το κόστος με παρόμοιο ρυθμό, ωστόσο το στρώμα μέσου όρου έχει μια ελάχιστη υπεροχή έναντι του μεγίστου. Συμπερασματικά, για την εκπαίδευση όλων των μοντέλων θέτουμε στρώμα αθροίσματος για την πρόβλεψη LogP και SAS και στρώμα μέσου όρου για το SAS. Είναι σημαντικό να αναφερθεί πως αυτή η επιλογή χρησιμοποιείται και για όλες τις αρχιτεκτονικές που ακολουθούν μιας και η διαφορά στην απόδοση φαίνεται κατά πολύ καλύτερη. Το επόμενο βήμα είναι η επιλογή κρυφών στρωμάτων. Συγκεκριμένα επιλέχθηκαν στρώματα από το εύρος $\{2, \dots, 6\}$ κρατώντας σταθερούς τους κρυφούς νευρώνες στους 30. Κατόπιν εκπαίδευσης κάθε μοντέλου, αυτά αξιολογήθηκαν ως προς το μέσο τετραγωνικό σφάλμα στο σύνολο των άγνωστων «ανεκπαίδευτων» δεδομένων προκειμένου να διαπιστωθεί το καλύτερο μοντέλο για κάθε μια ιδιότητα. Επίσης, δεν χρησιμοποιήθηκε συνάρτηση dropout σε κανένα εκ των 5 μοντέλων.

Μοντέλο GCN			
Κρυφά στρώματα	Μέσο τετραγωνικό σφάλμα δεδομένων εξέτασης		
	LogP	TPSA	SAS
2	0.02954	0.7049	0.03920
3	0.02752	0.6918	0.03260
4	0.02669	**0.3838**	0.03125
5	0.02239	0.4142	**0.02922**
6	**0.02086**	0.8609	0.03007

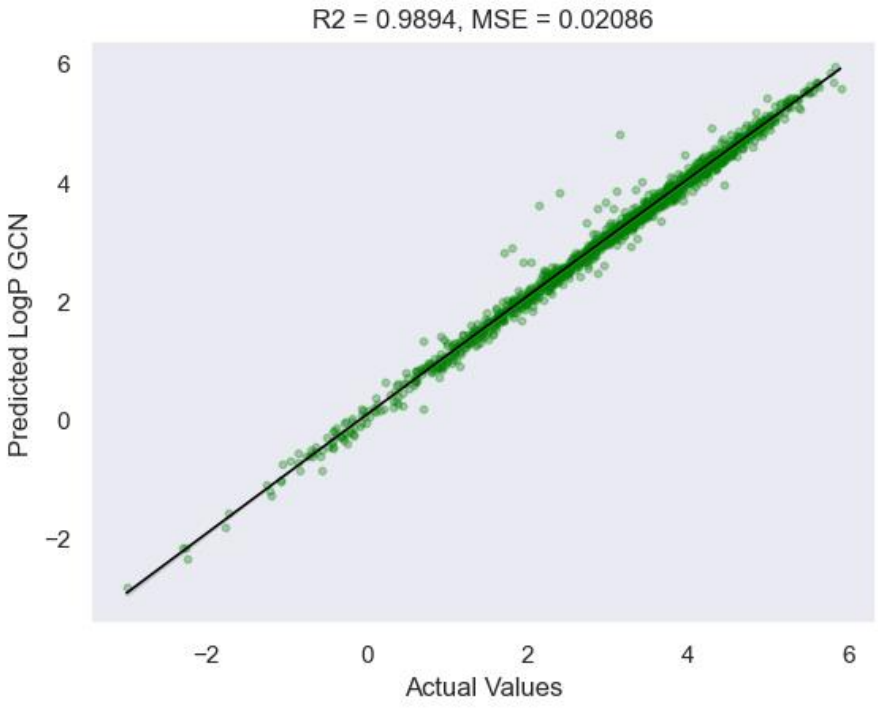
Πίνακας 6.3 Μέσα τετραγωνικά σφάλματα ιδιοτήτων LogP, TPSA, SAS δεδομένων εξέτασης (άγνωστων) στο δίκτυο GCN με εύρος στρωμάτων 2 έως 6

Σύμφωνα με τα αποτελέσματα του παραπάνω πίνακα, προκύπτει ότι τα δεδομένα εξέτασης έχουν καλύτερο μέσο τετραγωνικό σφάλμα LogP καθώς αυξάνονται τα κρυφά στρώματα με το καλύτερα να εμφανίζεται για 6 στρώματα. Η ιδιότητα TPSA φαίνεται να εμφανίζει καλύτερο σφάλμα στα 4 στρώματα. Όμοια με το LogP, γίνεται καλύτερη πρόβλεψη SAS με αύξηση κρυφών στρωμάτων και η καλύτερη επιλογή είναι τα 5.

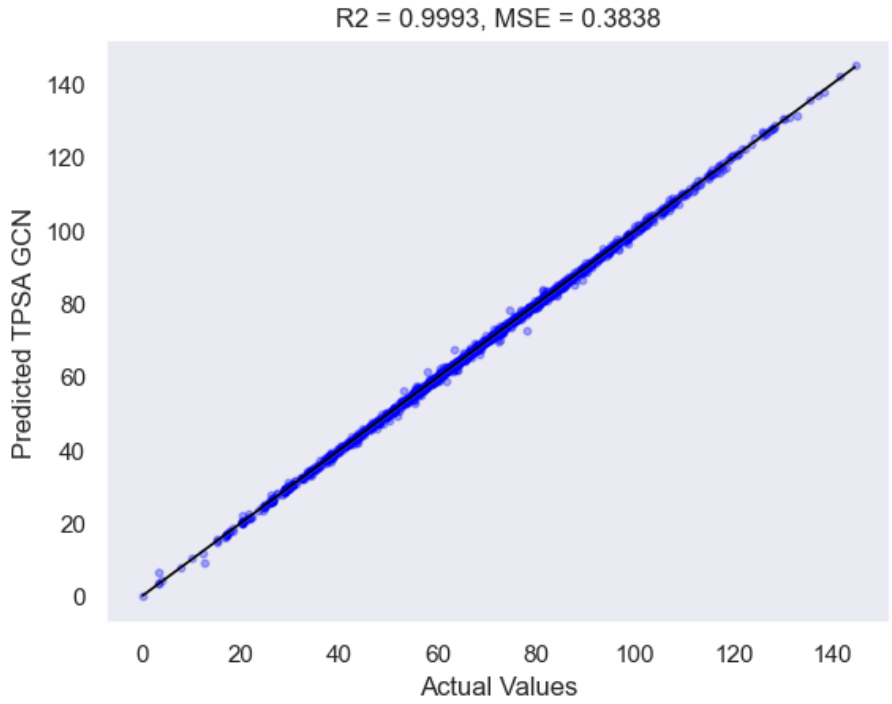


Εικόνα 6.3 Διάγραμμα κόστους (εκπαίδευσης, επαλήθευσης) συναρτήσει εποχών για τα βέλτιστα στρώματα πρόβλεψης κάθε ιδιότητας (6 κρυφά στρώματα για LogP, 4 κρυφά στρώματα για TPSA και 5 για SAS) του μοντέλου GCN

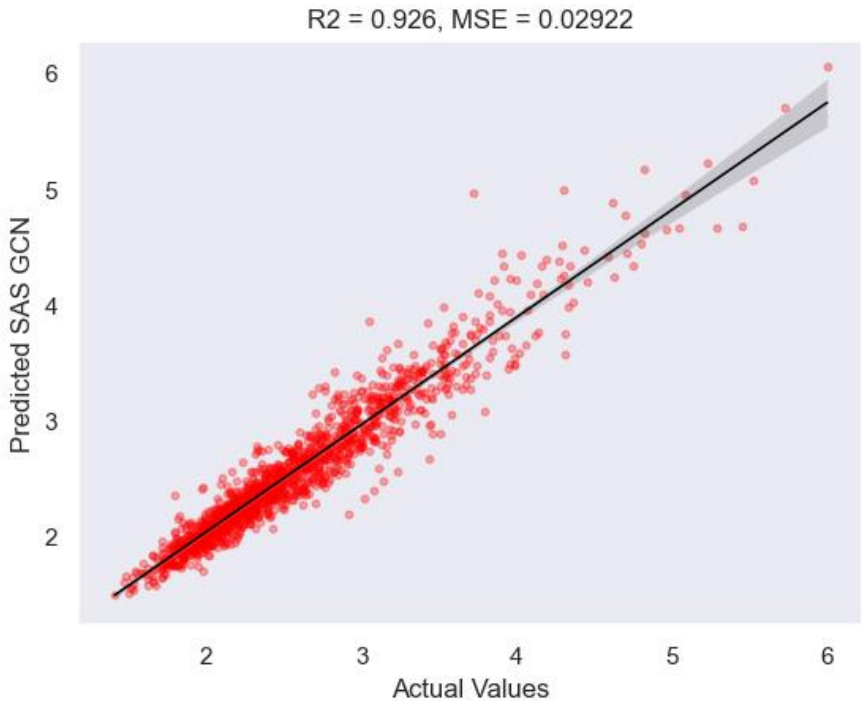
Από το σχήμα παρατηρείται η μεταβολή της συνάρτησης κόστους κατά την εκπαίδευση και την επαλήθευση χρησιμοποιώντας τα βέλτιστα στρώματα για κάθε μια προβλεπόμενη ιδιότητα. Μετά από περίπου 10 εποχές εκπαίδευσης παρατηρείται μια σταθεροποίηση του κόστους, με αυτό να μεταβάλλεται με μικρότερο ρυθμό από το σημείο αυτό και μετά. Επιπλέον το μοντέλο βελτιστοποιεί τις παραμέτρους του χωρίς να παρατηρείται υπερπροσαρμογή, μιας και η συνάρτηση κόστους στα δεδομένα επαλήθευσης δεν παρουσιάζει διακυμάνσεις και μειώνεται σταδιακά.



Εικόνα 6.4 Απεικόνιση προβλεπόμενων-πραγματικών τιμών LogP για το βέλτιστο μοντέλο GCN



Εικόνα 6.5 Απεικόνιση προβλεπόμενων-πραγματικών τιμών TPSA για το βέλτιστο μοντέλο GCN



Εικόνα 6.6 Απεικόνιση προβλεπόμενων-πραγματικών τιμών SAS για το βέλτιστο μοντέλο GCN

6.3.2 Αποτελέσματα SAGE

SAGE mean

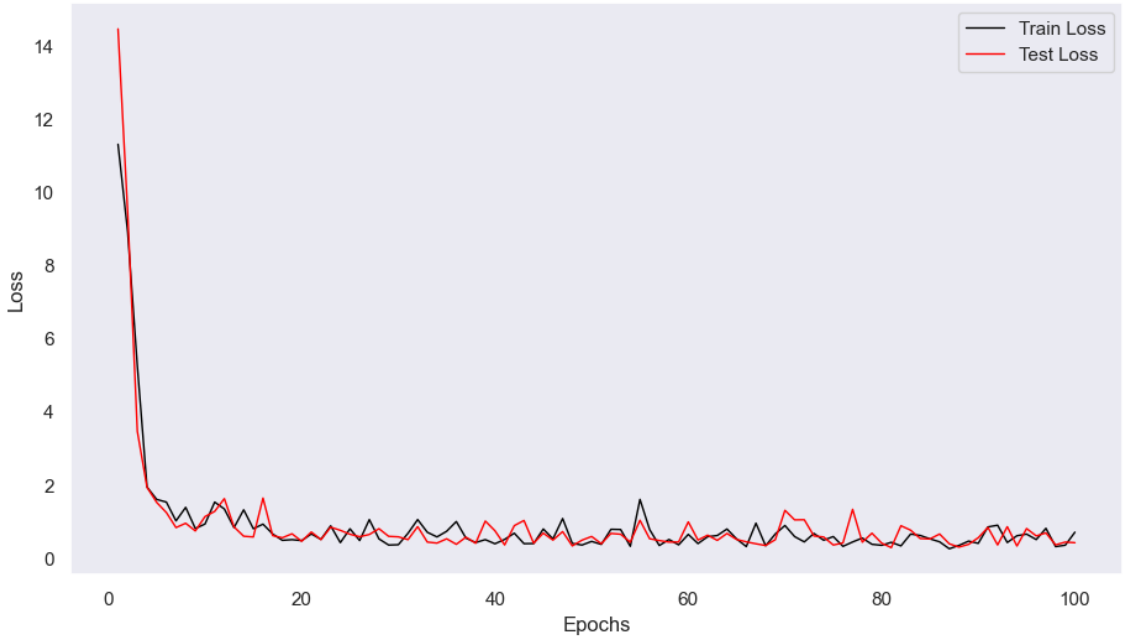
Όπως και με τα μοντέλα GCN, τα μοντέλα SAGE αρχικά εξετάστηκαν ως προς τα κρυφά στρώματα με 30 κρυφούς νευρώνες για 100 εποχές.

Μοντέλο SAGE (mean)			
Κρυφά στρώματα	Μέσο τετραγωνικό σφάλμα δεδομένων εξέτασης		
	LogP	TPSA	SAS
2	0.01467	0.08053	0.03175

3	0.01140	**0.06689**	**0.02852**
4	0.01062	0.06768	0.03220
5	0.01526	2.6	0.03312
6	**0.009411**	0.4711	0.02907

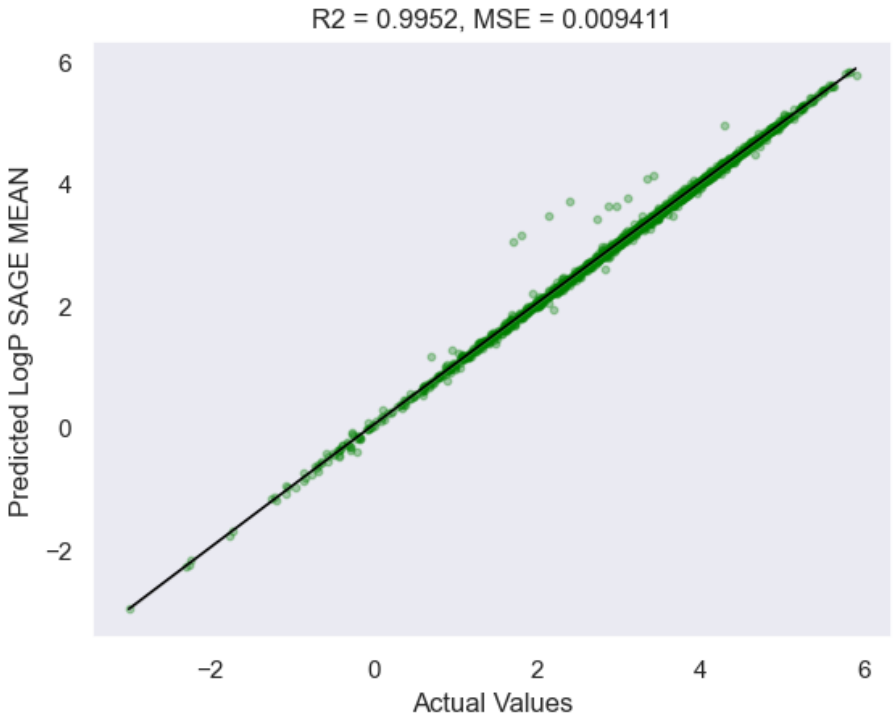
Πίνακας 6.4 Μέσα τετραγωνικά σφάλματα ιδιοτήτων LogP, TPSA, SAS δεδομένων εξέτασης (άγνωστων) στο δίκτυο GraphSAGE (mean) με εύρος στρωμάτων από 2 έως 6.

Η εκπαίδευση του GraphSAGE (mean) φαίνεται να αποφέρει πολύ καλά αποτελέσματα κατά την πρόβλεψη των δεδομένων εξέτασης. Συγκεκριμένα, όσο αυξάνονται τα κρυφά στρώματα, αυξάνεται και η ακρίβεια του LogP με το καλύτερο μοντέλο να είναι στα 6 στρώματα. Αντιθέτως, η αύξηση κρυφών στρωμάτων χειροτερεύει την πρόβλεψη του TPSA και του SAS και για αυτό επιλέγονται τα 3 κρυφά στρώματα και στις δύο περιπτώσεις. Για αυτά τα στρώματα που προαναφέρθηκαν παρουσιάζεται η μεταβολή της συνάρτησης κόστους με τις εποχές.

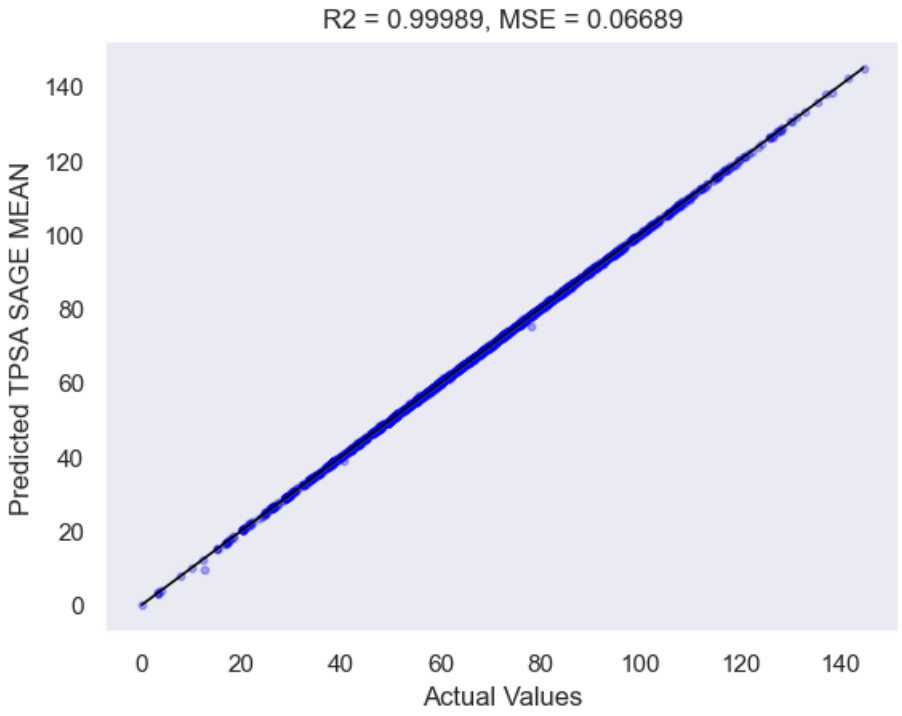


Εικόνα 6.7 Διάγραμμα κόστους (εκπαίδευσης, επαλήθευσης) συναρτήσει εποχών για τα βέλτιστα στρώματα πρόβλεψης κάθε ιδιότητας (6 κρυφά στρώματα LogP και 3 κρυφά στρώματα γιαTPSA, SAS) του μοντέλου GraphSAGE (mean)

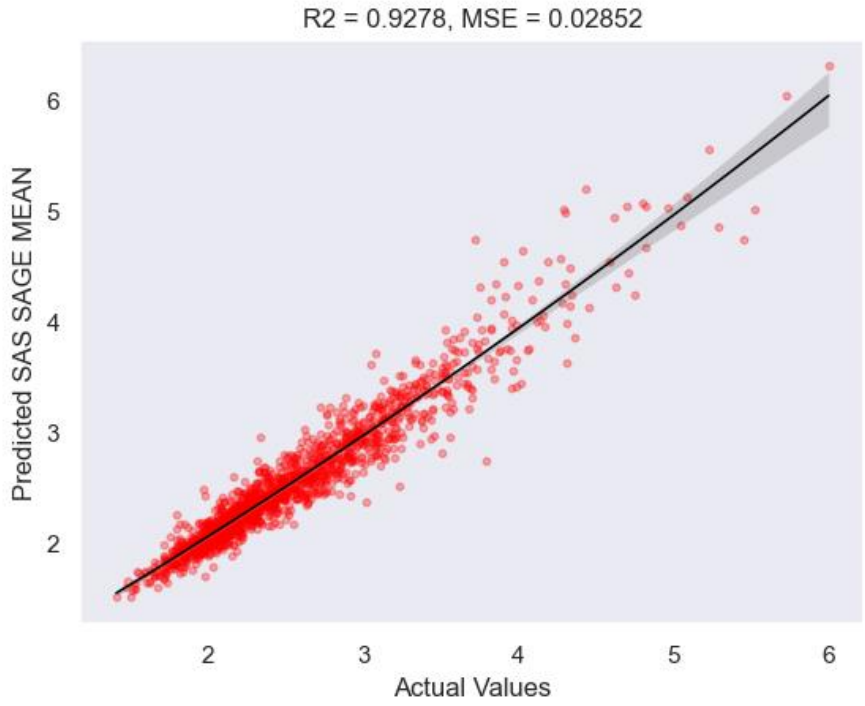
Ομοίως και με το μοντέλο GCN, η συνάρτηση κόστους στο GraphSAGE (mean) φαίνεται να σταθεροποιείται μετά από τις 10 εποχές. Προσεγγίζει χαμηλότερες τιμές από το GCN τόσο ως προς τα δεδομένα εκπαίδευσης όσο και ως προς τα επαλήθευσης. Επίσης, παρουσιάζει μικρότερες αιχμές (spikes) σε σχέση με το GCN και η διαδικασία της εύρεσης βέλτιστων παραμέτρων είναι πιο ομαλή.



Εικόνα 6.8 Απεικόνιση προβλεπόμενων-πραγματικών τιμών LogP για το βέλτιστο μοντέλο SAGE MEAN



Εικόνα 6.9 Απεικόνιση προβλεπόμενων-πραγματικών τιμών TPSA για το βέλτιστο μοντέλο SAGE MEAN



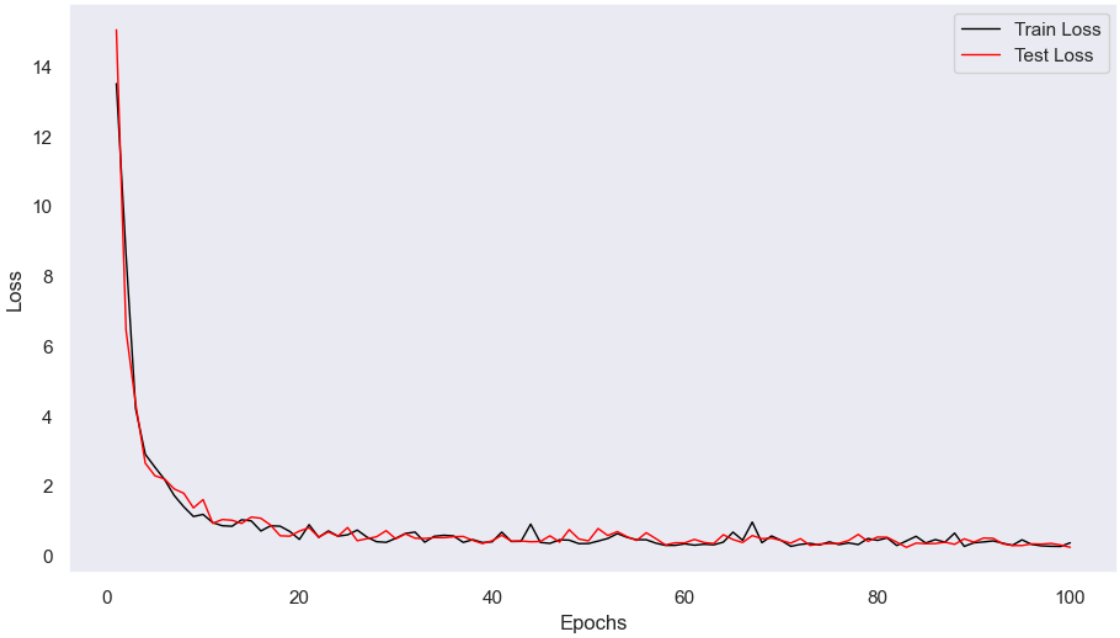
Εικόνα 6.10 Απεικόνιση προβλεπόμενων-πραγματικών τιμών SAS για το βέλτιστο μοντέλο SAGE MEAN

SAGE max

Μοντέλο SAGE (max)			
Κρυφά στρώματα	Μέσο τετραγωνικό σφάλμα δεδομένων εξέτασης		
	LogP	TPSA	SAS
2	0.01449	**0.03950**	0.02895
3	0.01491	0.1252	0.02704
4	0.01088	0.1128	0.02741
5	0.01135	2.006	**0.02303**
6	**0.007236**	0.1745	0.03088

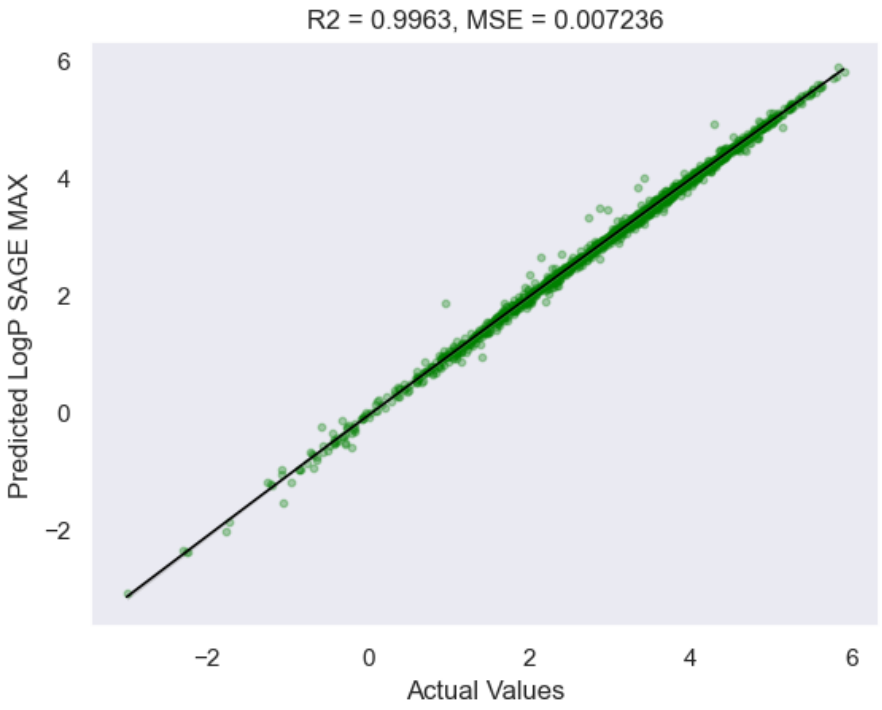
Πίνακας 6.5 Μέσα τετραγωνικά σφάλματα ιδιοτήτων LogP, TPSA, SAS δεδομένων εξέτασης (άγνωστων) στο δίκτυο GraphSAGE Max με εύρος στρωμάτων 2 έως 6

Με τη χρήση τη συνάρτησης μεγίστου στο GraphSAGE, λαμβάνονται εξίσου καλά αποτελέσματα πρόβλεψης και των τριών ιδιοτήτων. Όπως και με το GraphSAGE (mean), η πρόβλεψη του LogP βελτιώνεται με την αύξηση των κρυφών στρωμάτων με το καλύτερο σφάλμα να παρατηρείται στα 6. Η ιδιότητα TPSA δίνει μικρότερο μέσο τετραγωνικό σφάλμα με τη χρήση 2 στρωμάτων ενώ η SAS για 5 κρυφά στρώματα. Επιπλέον παρατηρείται ότι για 5 κρυφά στρώματα, η πρόβλεψη TPSA είναι κατά πολύ χειρότερη.

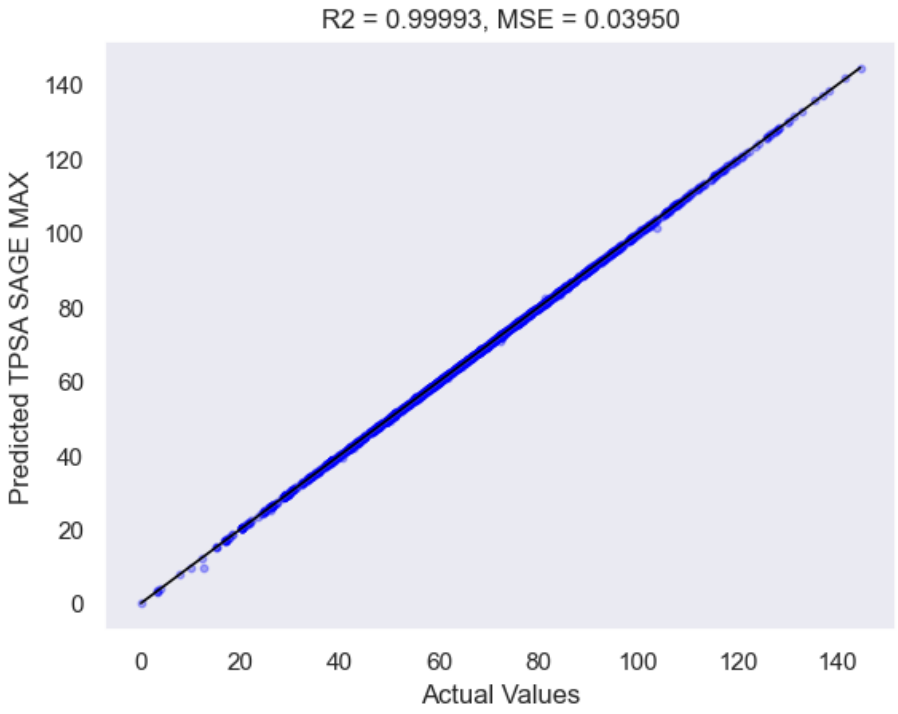


Εικόνα 6.11 Διάγραμμα κόστους (εκπαίδευσης, επαλήθευσης) συναρτήσει εποχών για τα βέλτιστα στρώματα κάθε ιδιότητας (6 κρυφά στρώματα LogP,2 κρυφά στρώματα TPSA και 5 κρυφά στρώματα SAS) του μοντέλου GraphSAGE (max)

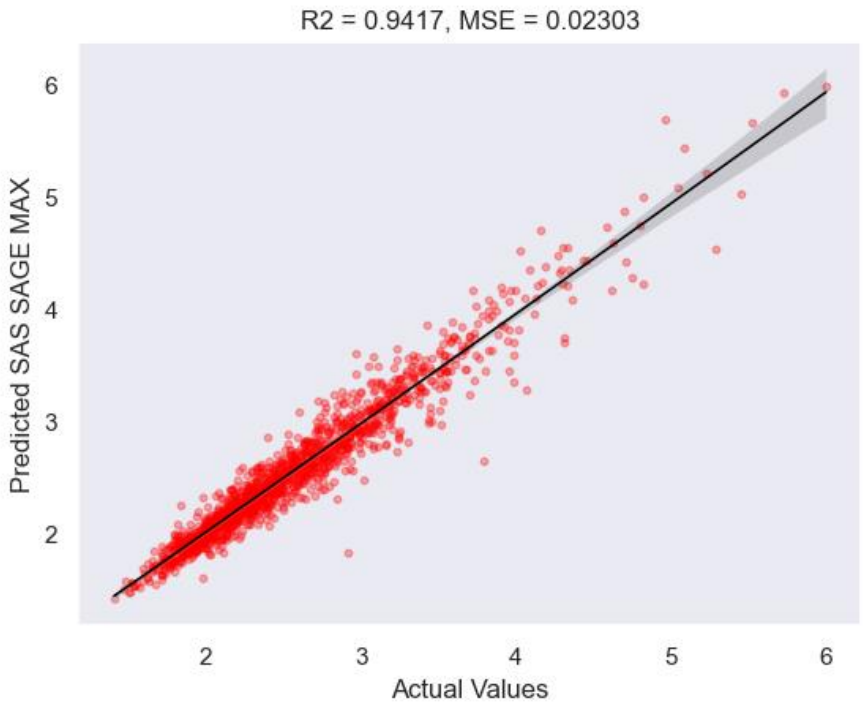
Παρατηρώντας την διαδικασία της μεταβολής του κόστους με τις εποχές, διαπιστώνεται ότι είναι περισσότερο ομαλή σε σχέση με το GraphSAGE (mean) και παρουσιάζονται μικρότερες διακυμάνσεις στη συνάρτηση κόστους τόσο των δεδομένων εκπαίδευσης όσο και στα επαλήθευσης.



Εικόνα 6.12 Απεικόνιση προβλεπόμενων-πραγματικών τιμών LogP για το βέλτιστο μοντέλο SAGE MAX



Εικόνα 6.13 Απεικόνιση προβλεπόμενων-πραγματικών τιμών TPSA για το βέλτιστο μοντέλο SAGE MAX



Εικόνα 6.14 Απεικόνιση προβλεπόμενων-πραγματικών τιμών TPSA για το βέλτιστο μοντέλο SAGE MAX

6.3.3 Αποτελέσματα GAT

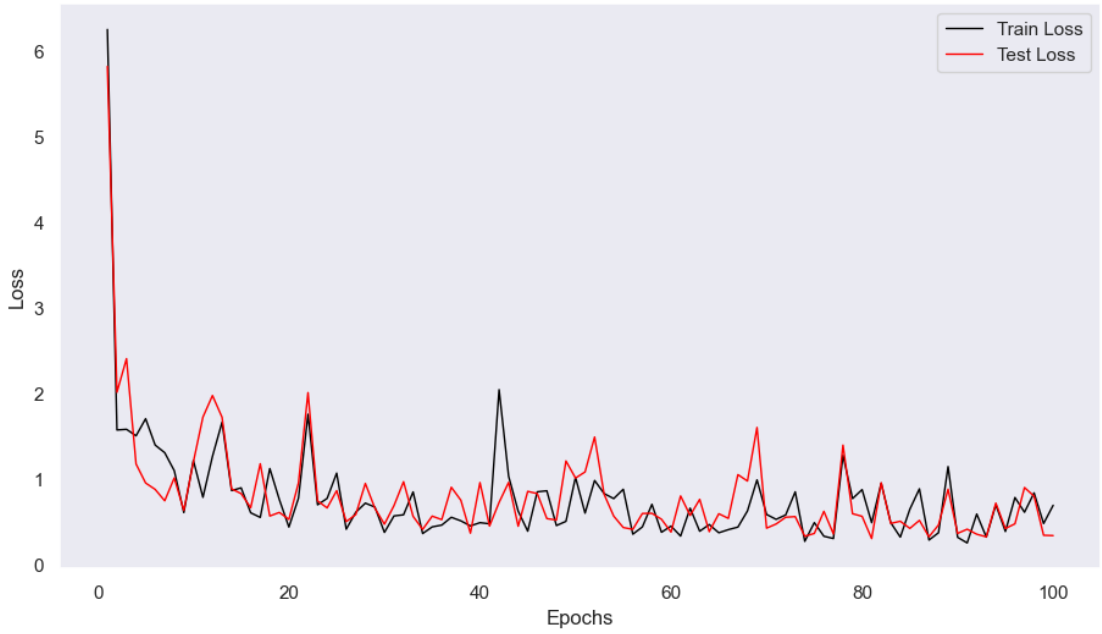
Εξετάζοντας με βάση τα ίδια δομικά χαρακτηριστικά των ανωτέρω δικτύων και εφαρμόζοντας τον μηχανισμό της προσοχής με 8 ανεξάρτητα κεφάλια γίνεται η βελτιστοποίηση των μοντέλων GAT.

Μοντέλο GAT			
Κρυφά στρώματα	Μέσο τετραγωνικό σφάλμα δεδομένων εξέτασης		
	LogP	TPSA	SAS

2	0.009750	**0.1223**	0.02444
3	**0.004967**	0.1561	0.02434
4	0.006736	0.6121	0.02125
5	0.006314	1.576	**0.02072**
6	0.005753	0.7476	0.02253

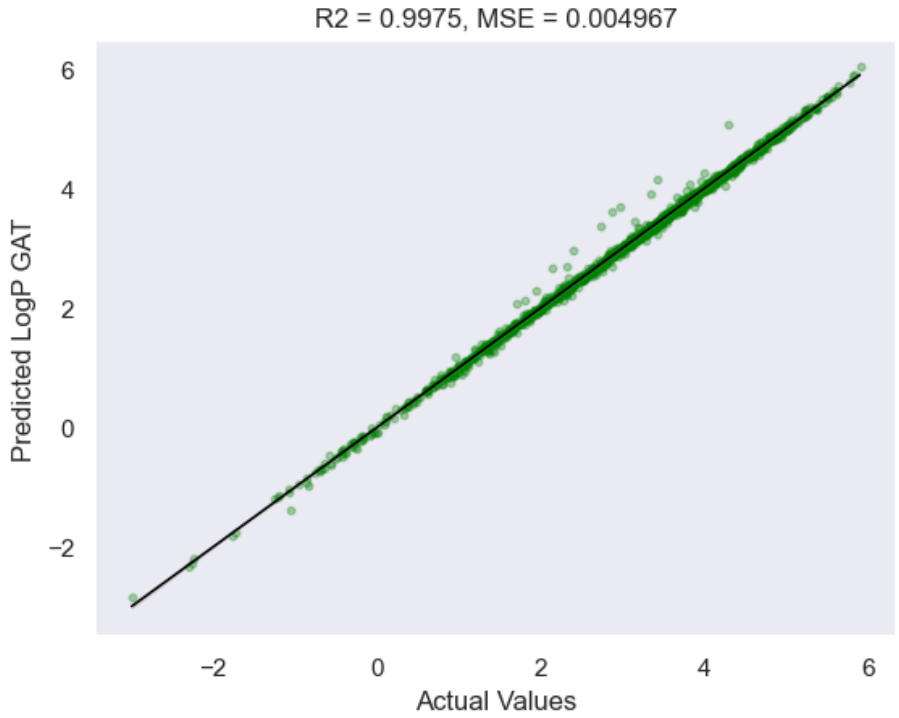
Πίνακας 6.6 Μέσα απόλυτα σφάλματα ιδιοτήτων LogP, TPSA, SAS δεδομένων εξέτασης (άγνωστων) στο δίκτυο GAT με εύρος στρωμάτων 2 έως 6

Τα καλύτερα μοντέλα GAT για την πρόβλεψη των τριών ιδιοτήτων LogP, TPSA, SAS είναι αρκετά ικανοποιητικά και για τις τρεις ιδιότητες. Για την καλή πρόβλεψη LogP και TPSA δεν απαιτούνται πολλά κρυφά στρώματα παρά μονάχα 3 και 2 αντίστοιχα. Αντιθέτως, για την ιδιότητα SAS η αύξηση κρυφών στρωμάτων αποφέρει καλύτερα αποτελέσματα με την βέλτιστη να είναι στα 5 στρώματα.

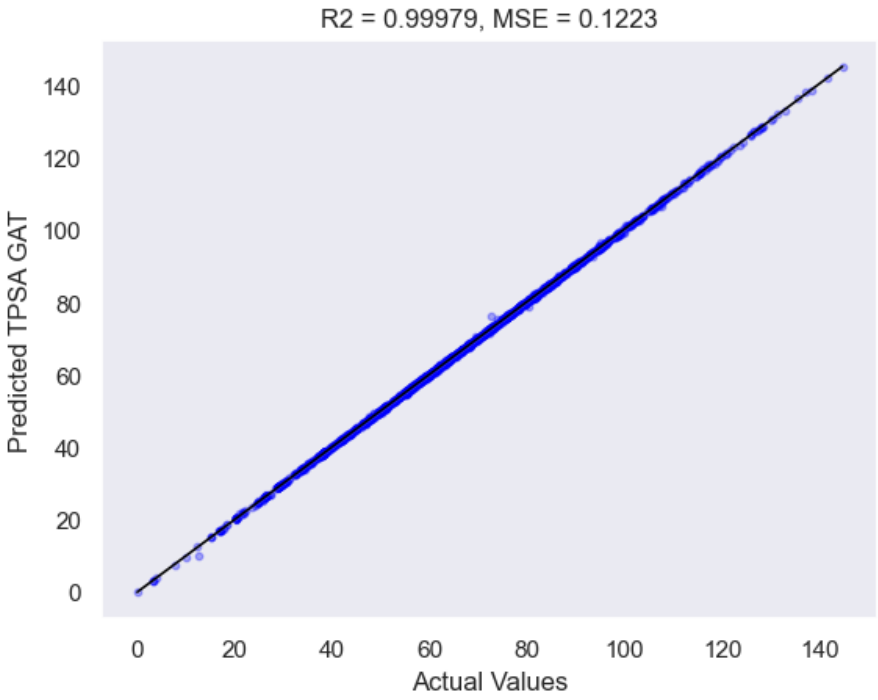


Εικόνα 6.15 Διάγραμμα κόστους (εκπαίδευσης, επαλήθευσης) συναρτήσει εποχών για τα βέλτιστα στρώματα κάθε ιδιότητας (3 κρυφά στρώματα LogP, 2 για TPSA και 5 κρυφά στρώματα SAS)

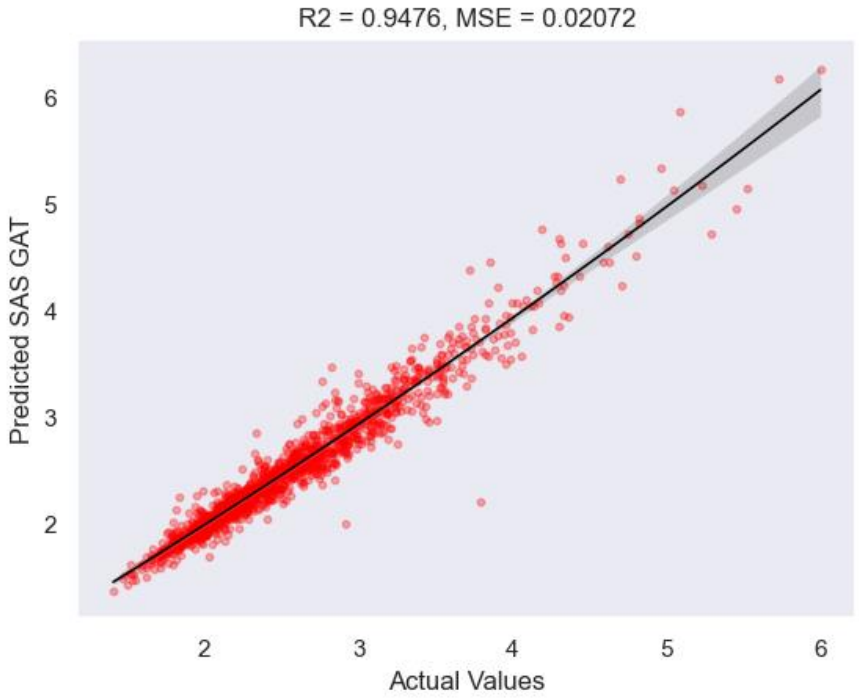
Μια διαφορά που παρατηρείται βλέποντας τις καμπύλες κόστους εποχών για το GAT σε σχέση με τις προηγούμενες αρχιτεκτονικές, είναι οι αιχμές (spikes) του διαγράμματος. Παρόλα αυτά, τα τελικά αποτελέσματα είναι ικανοποιητικά παρά τα spikes και η αντικειμενική συνάρτηση κόστους μειώνεται.



Εικόνα 6.16 Απεικόνιση προβλεπόμενων-πραγματικών τιμών LogP για το βέλτιστο μοντέλο GAT



Εικόνα 6.17 Απεικόνιση προβλεπόμενων-πραγματικών τιμών TPSA για το βέλτιστο μοντέλο GAT



Εικόνα 6.18 Απεικόνιση προβλεπόμενων-πραγματικών τιμών SAS για το βέλτιστο μοντέλο GAT

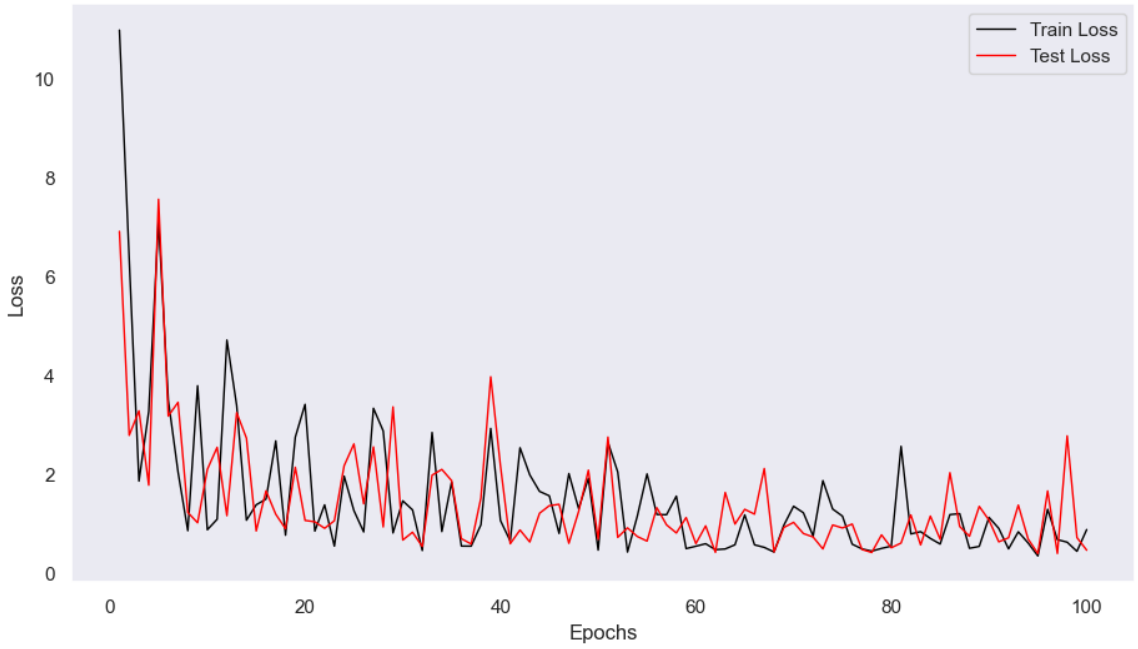
6.3.4 Αποτελέσματα GATv2

Όπως αναφέρθηκε και στη θεωρία, τα GATv2 αποτελούν μια τροποποίηση του GAT όσον αφορά τη μονάδα διάδοσης. Εξετάστηκαν με τις ίδιες παραμέτρους που επιλέχθηκαν και στο GAT και ελέγχθηκε στη συνέχεια το κατά πόσο αυτή η αλλαγή μπορεί να βελτιώσει την πρόβλεψη.

Μοντέλο GATv2			
Κρυφά στρώματα	Μέσο τετραγωνικό σφάλμα δεδομένων εξέτασης		
	LogP	TPSA	SAS
2	0.008493	0.3264	0.02143
3	0.007875	2.174	0.02116
4	**0.004945**	0.3999	**0.01877**
5	0.005496	**0.2160**	0.01962
6	0.005333	1.078	0.0223

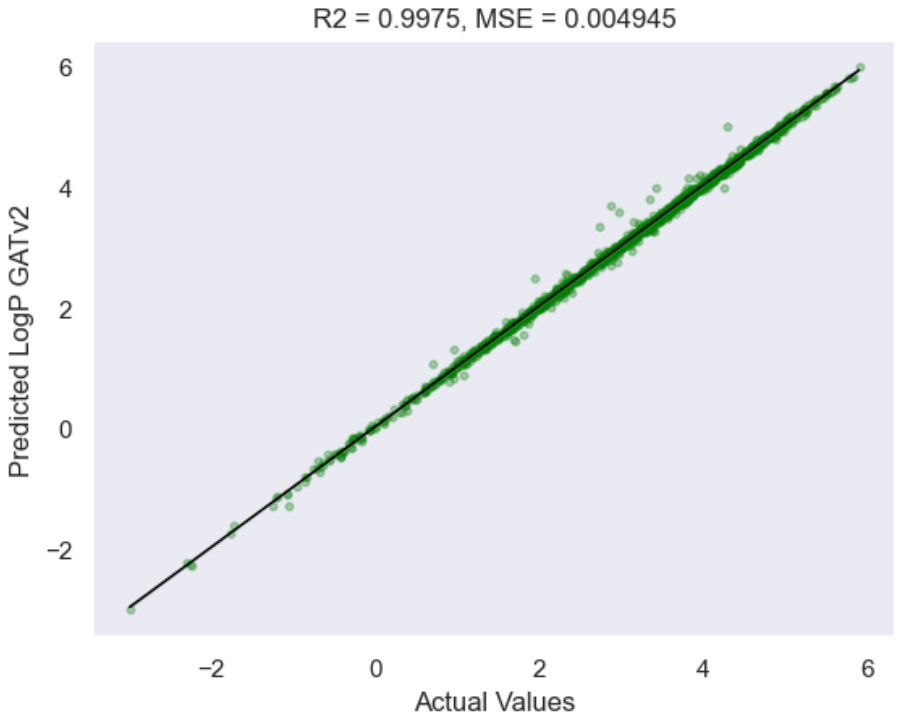
ΠΙΝΑΚΑΣ 6.7 Μέσα απόλυτα σφάλματα ιδιοτήτων LogP, TPSA, SAS δεδομένων εξέτασης (άγνωστων) στο δίκτυο GATv2 με εύρος στρωμάτων 2 έως 6

Στα μοντέλα GATv2, τα μοντέλα LogP φαίνεται να αποδίδουν καλύτερα με αύξηση των κρυφών στρωμάτων και το καλύτερο μέσο τετραγωνικό σφάλμα ήταν στα 4 κρυφά στρώματα. Για την πρόβλεψη TPSA και SAS, επιλέχθηκαν τα 5 και 4 στρώματα αντιστοίχως.

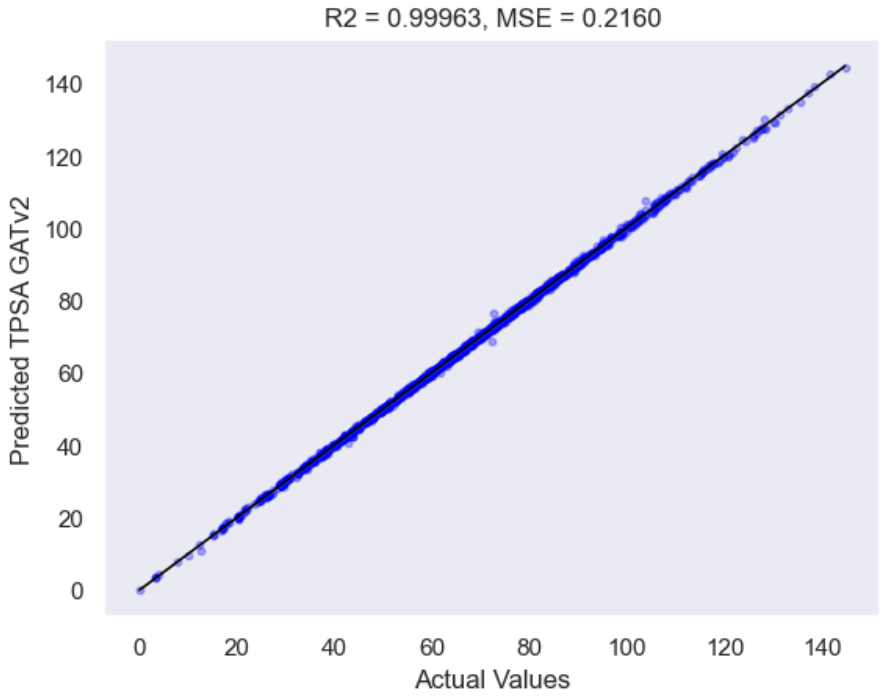


Εικόνα 6.19 Διάγραμμα κόστους (εκπαίδευσης, επαλήθευσης) συναρτήσει εποχών για τα βέλτιστα στρώματα κάθε ιδιότητας (4 κρυφά στρώματα LogP, 5 κρυφά στρώματα TPSA και 4 κρυφά στρώματα SAS)

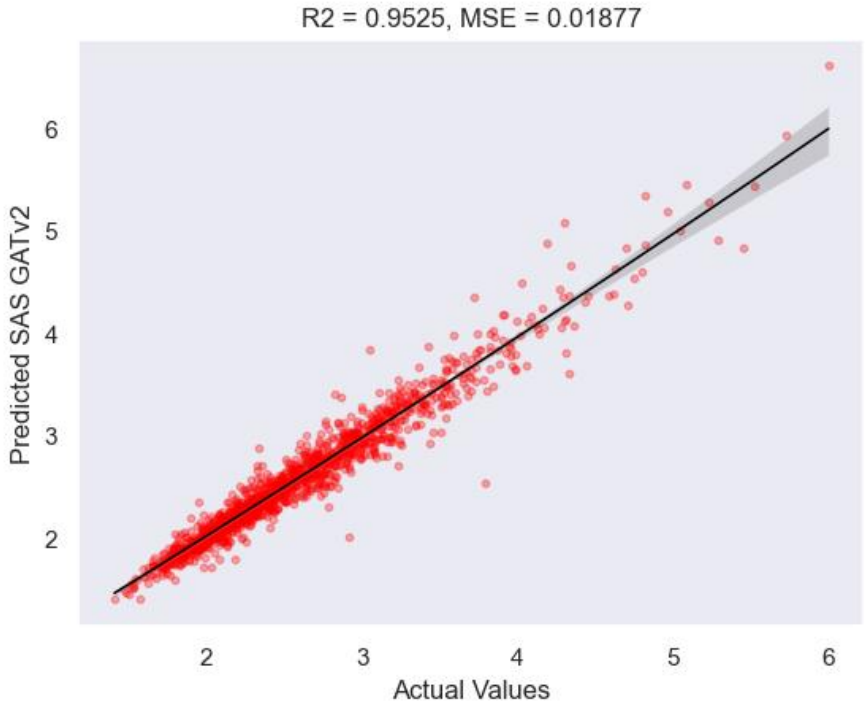
Όπως και με τα GAT, στα GATv2 εμφανίζονται spikes της συνάρτησης κόστους και μάλιστα με μεγαλύτερη διακύμανση, ωστόσο η συνάρτηση κόστους εκπαίδευσης και επαλήθευσης πάλι μειώνεται.



Εικόνα 6.20 Απεικόνιση προβλεπόμενων-πραγματικών τιμών LogP για το βέλτιστο μοντέλο GATV2



Εικόνα 6.21 Απεικόνιση προβλεπόμενων-πραγματικών τιμών TPSA για το βέλτιστο μοντέλο GATV2

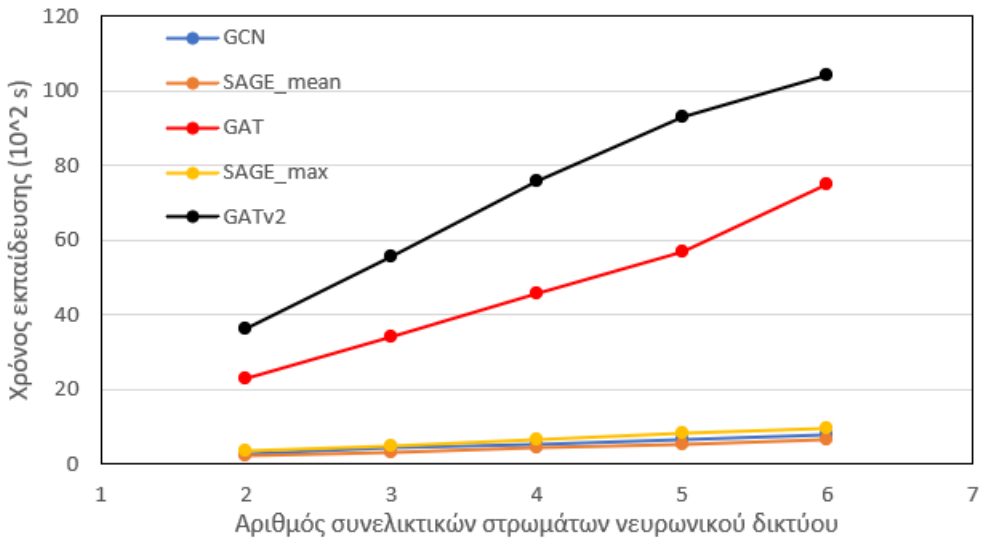


Εικόνα 6.22 Απεικόνιση προβλεπόμενων-πραγματικών τιμών SAS για το βέλτιστο μοντέλο GATV2

6.4 Σύγκριση μοντέλων

6.4.1 Υπολογιστικός χρόνος εκπαίδευσης

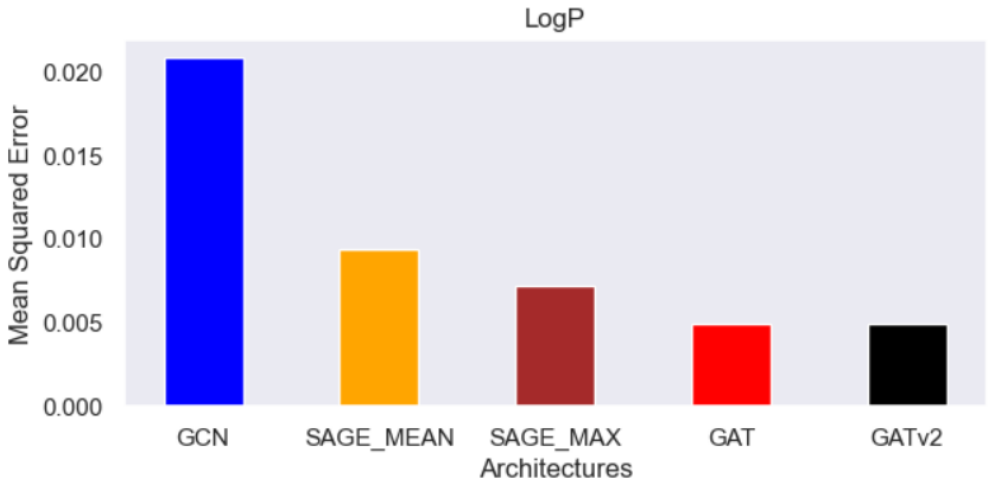
Λόγω της διαφορετικής πολυπλοκότητας των διάφορων αρχιτεκτονικών, οι χρόνοι εκπαίδευσης διαφέρουν. Παρακάτω παρατίθεται μια ένδειξη του υπολογιστικού χρόνου εκπαίδευσης των μοντέλων ως συνάρτηση του αριθμού των κρυφών στρωμάτων.



Εικόνα 6.23 Διάγραμμα χρόνου εκπαίδευσης (10^{-2}s) για κάθε μοντέλο συναρτήσει κρυφών στρωμάτων.

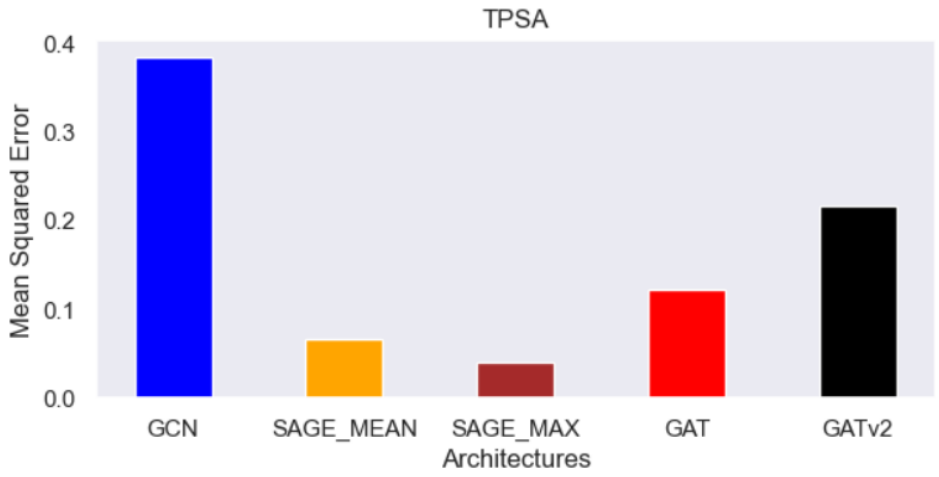
Εξετάζοντας το διάγραμμα είναι φανερό, ότι οι αρχιτεκτονικές GCN, SAGE είναι υπολογιστικά φθηνότερες. Αντιθέτως τα μοντέλα GAT , GATv2 είναι πιο χρονοβόρα ως προς την εκπαίδευση. Αυτό οφείλεται κυρίως στον υπολογισμό των συντελεστών προσοχής που απαιτεί υπολογισμούς αρκετών εκθετικών όρων και στη χρήση πολλαπλών κεφαλιών προσοχής. Επίσης όσον αφορά τα δύο μοντέλα προσοχής, το GAT απαιτεί λιγότερο χρόνο για την εκπαίδευση των δεδομένων από ότι το GATv2.

6.4.2 Μέσα τετραγωνικά σφάλματα



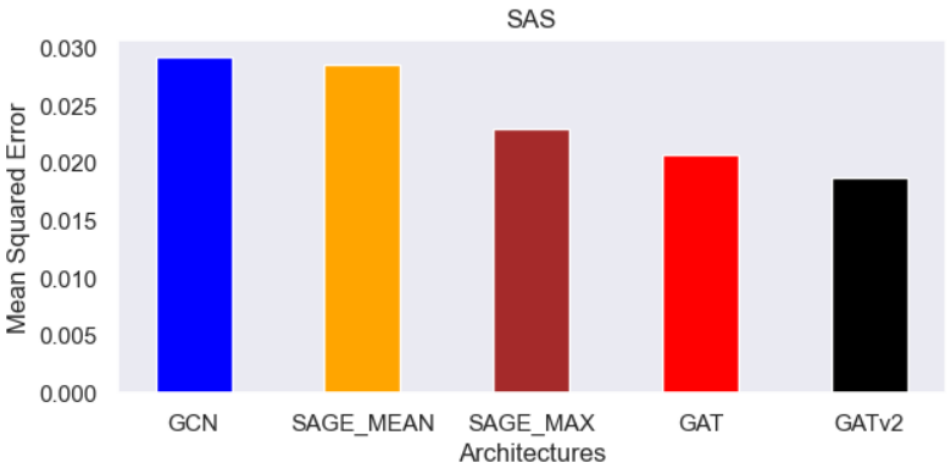
Εικόνα 6.24 Ραβδόγραμμα μέσων τετραγωνικών σφαλμάτων LogP στα βέλτιστα μοντέλα κάθε αρχιτεκτονικής

Λαμβάνοντας τις καλύτερες τιμές του μέσου τετραγωνικού σφάλματος LogP για κάθε μοντέλο, δημιουργήθηκε το παραπάνω σχήμα. Αρχικά, είναι φανερό ότι οι αρχιτεκτονικές GraphSAGE αλλά και οι attentional δίνουν μικρότερο σφάλμα και είναι καλύτερες από το GCN. Οι καλύτερες προβλέψεις έγιναν με τα μοντέλα προσοχής GAT, GATv2 χωρίς να υπάρχει κάποια μεγάλη διαφορά μεταξύ τους. Τα μοντέλα SAGE ακολουθούν, με το SAGE (max) να είναι ελάχιστα καλύτερο από το SAGE (mean). Αν λάβει κανείς υπόψιν και τον υπολογιστικό χρόνο εκπαίδευσης, το μοντέλο GAT είναι το καλύτερο από όλα για την πρόβλεψη της ιδιότητας. Επίσης, η πρόβλεψη αυτή γίνεται μόλις με 3 κρυφά στρώματα πράγμα που σημαίνει ότι δεν χρειάζονται πολλοί παράμετροι για να εξαχθεί μια καλή πρόβλεψη της διαλυτότητας.



Εικόνα 6.25 Ραβδόγραμμα μέσων τετραγωνικών σφαλμάτων TPSA στα βέλτιστα μοντέλα κάθε αρχιτεκτονικής

Όπως και στην πρόβλεψη LogP έτσι και στην TPSA, το κλασσικό νευρωνικό δίκτυο συνέλιξης (GCN) είναι χειρότερο από τα υπόλοιπα μοντέλα. Γενικότερα, για την πρόβλεψη TPSA, φαίνεται πως την καλύτερη δουλειά την κάνουν οι αρχιτεκτονικές GraphSAGE . Στη συνέχεια ακολουθούν τα μοντέλα προσοχής με το GAT να υπερέχει ξανά σε σχέση με το GATv2, επιτυγχάνοντας μια επίσης καλή πρόβλεψη. Το SAGE (max) λαμβάνεται ως το καλύτερο μοντέλο και με τη χρήση μονάχα 2 κρυφών στρωμάτων προσεγγίζει πολύ χαμηλή τιμή στο μέσο τετραγωνικό σφάλμα.



Εικόνα 6.26 Ραβδόγραμμα μέσων τετραγωνικών σφαλμάτων SAS στα βέλτιστα μοντέλα κάθε αρχιτεκτονικής

Τέλος, η πρόβλεψη SAS φαίνεται να είναι καλύτερη χρησιμοποιώντας τους μηχανισμούς προσοχής. Η αρχιτεκτονική SAGE (mean) είναι ελάχιστα καλύτερη από το κλασσικό GCN, ενώ η SAGE (max) είναι καλύτερη και από τις δύο. Την καλύτερη πρόβλεψη την κάνει το μοντέλο GATv2 με τη χρήση τεσσάρων κρυφών στρωμάτων.

Κεφάλαιο 7: Συμπεράσματα και Προτάσεις για Μελλοντική Έρευνα

7.1 Συμπεράσματα

Στην παρούσα διπλωματική εργασία, αναπτύχθηκαν μοντέλα πρόβλεψης μοριακών ιδιοτήτων με τη χρήση βαθιάς μηχανικής μάθησης. Σκοπός ήταν να υπολογιστούν με υψηλή ακρίβεια ο συντελεστής κατανομής (LogP), η τοπολογική πολική επιφάνεια (TPSA) και ο δείκτης συνθετικής προσβασιμότητας (SAS). Τα μοντέλα βασίστηκαν στις αρχιτεκτονικές νευρωνικών γραφημάτων (GNN). Συγκεκριμένα αναπτύχθηκαν μοντέλα GCN, GraphSAGE, GAT και GATv2 και στο τέλος πραγματοποιήθηκε μια μεταξύ τους σύγκριση για να προκύψει το καλύτερο μοντέλο για κάθε μία ιδιότητα.

Το σετ των δεδομένων αποτελείτο από 15.000 μικρά οργανικά μόρια με γνωστές τιμές των τριών ιδιοτήτων. Το κάθε μόριο εισάγεται στο μοντέλο ως μοριακό γράφημα με τον πίνακα χαρακτηριστικών κορυφής X και τον πίνακα γειτνίασης A. Ο πίνακας X περιέχει για κάθε κορυφή, δηλαδή χημικό άτομο, πληροφορίες που το χαρακτηρίζουν.

Όσον αφορά την αποτελεσματικότητα των μοντέλων για την πρόβλεψη κάθε ιδιότητας προέκυψαν κάποια ενδιαφέρων συμπεράσματα. Αρχικά, συμπεραίνεται πως οι πιο νέες αρχιτεκτονικές GraphSAGE, GAT, GATv2 είναι καλύτερες από την κλασσική αρχιτεκτονική GCN και στις τρεις ιδιότητες. Συγκεκριμένα, στα δεδομένα εξέτασης (test), τα μέσα τετραγωνικά σφάλματα των ιδιοτήτων είναι πάντοτε καλύτερα για τις προαναφερθείσες αρχιτεκτονικές από ότι στο GCN. Εξετάζοντας κάθε ιδιότητα ξεχωριστά, προκύπτει ότι για την πρόβλεψη LogP και SAS, τα μοντέλα προσοχής (attentional) παρουσιάζουν καλύτερα προβλεπτικά αποτελέσματα ενώ η ιδιότητα TPSA προβλέπεται καλύτερα με τη χρήση GraphSAGE νευρωνικών δικτύων.

7.2 Μελλοντικές Προτάσεις

Σκοπός της παρούσας εργασίας ήταν η ανάπτυξη μοντέλων γραφημάτων για την πρόβλεψη της διαλυτότητας, πολικότητας και συνθετικής προσβασιμότητας. Τα αποτελέσματα που προέκυψαν από την πρόβλεψη των τριών ιδιοτήτων είναι αρκετά ικανοποιητικά. Μια πιθανή μελλοντική κατεύθυνση που θα μπορούσε να πάρει η συγκεκριμένη εργασία είναι να οριστεί ένας τομέας εφαρμογής (domain of applicability) για το μοντέλο. Δηλαδή, κάθε πρόβλεψη που επιτυγχάνεται να συνοδεύεται με μια πιθανότητα που θα περιλαμβάνει την αξιοπιστία του μοντέλου σε αυτήν την πρόβλεψη. Επίσης, αν και ο αριθμός δεδομένων φάνηκε ικανοποιητικός, είναι δυνατό να χρησιμοποιηθούν περισσότερα δεδομένα με μεγαλύτερο εύρος τιμών και στις τρεις ιδιότητες. Για παράδειγμα, να εισαχθούν περισσότερα μόρια με αρνητικές τιμές LogP ή μεγάλες τιμές SAS και με αυτόν τον τρόπο να καλύπτουν όλο το φάσμα των πιθανών τιμών.

Τέλος, για μια πιο γενική κατεύθυνση στην υπολογιστική ανακάλυψη φαρμάκων μπορεί να συμβάλλει η ενισχυτική μάθηση. Με την χρήση δύο νευρωνικών δικτύων, από τα οποία το ένα θα είναι το γεννητικό (generative) και το άλλο το προβλεπτικό (predictive) είναι δυνατή μια ταυτόχρονη βελτιστοποίηση με βάση τις αρχές της ενισχυτικής μάθησης. Στόχος είναι η δημιουργία νέα χημικών μορίων που θα έχουν τις επιθυμητές ιδιότητες [47].

Ακρωνύμια

ReLU	Rectified Linear Unit
MAE	Mean Absolute Error
MSE	Mean Squared Error
SGD	Stochastic Gradient Descent
CNN	Convolutional Neural Network
QSAR	Quantitative Structure - Activity Relationship
MPNN	Message Passing Neural Network
SMILES	Simplified Molecular Input Line Entry System
LogP	Logarithmic Partition Coefficient
TPSA	Topological Polar Surface Area
SAS	Synthetic Accessibility Score
GNN	Graph Neural Network
GCN	Graph Convolution Network
GraphSAGE	Graph Sample And Aggregate
GAT	Graph Attention Network

Βιβλιογραφία

[1] M. Ali, A. Dewan, A. K. Sahu, and M. M. Taye, "Understanding of Machine Learning with Deep Learning: Architectures, Workflow, Applications and Future Directions," *Computers 2023, Vol. 12, Page 91*, vol. 12, no. 5, p. 91, Apr. 2023, doi: 10.3390/COMPUTERS12050091.

[2] C. Janiesch, P. Zschech, and K. Heinrich, "Machine learning and deep learning," *Electronic Markets*, vol. 31, no. 3, pp. 685–695, Sep. 2021, doi: 10.1007/S12525-021-00475-2/TABLES/2.

[3] Tom Mitchell, "Machine Learning": Available at: http://www.cs.cmu.edu/~tom/mlbook.html

[4] Q. Liu and Y. Wu, "Supervised Learning," *Encyclopedia of the Sciences of Learning*, pp. 3243–3245, 2012, doi: 10.1007/978-1-4419-1428-6_451.

[5] J. Wang and F. Biljecki, "Unsupervised machine learning in urban studies: A systematic review of applications," *Cities*, vol. 129, Oct. 2022, doi: 10.1016/J.CITIES.2022.103925.

[6] K. J. Cios, R. W. Swiniarski, W. Pedrycz, and L. A. Kurgan, "Unsupervised Learning: Association Rules," *Data Mining*, pp. 289–306, Oct. 2007, doi: 10.1007/978-0-387-36795-8_10.

[7] W. Jia, M. Sun, J. Lian, and S. Hou, "Feature dimensionality reduction: a review," *Complex & Intelligent Systems 2022 8:3*, vol. 8, no. 3, pp. 2663–2693, Jan. 2022, doi: 10.1007/S40747-021-00637-X.

[8] Y. Ouali, C. Hudelot, and M. Tami, "An Overview of Deep Semi-Supervised Learning," Jun. 2020, Accessed: May 10, 2023. [Online]. Available: https://arxiv.org/abs/2006.05278v2

[9] Barto Sutton, "Reinforcement learning: An introduction": Available at: http://incompleteideas.net/book/the-book-2nd.html

[10] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain.," *Psychol Rev*, vol. 65, no. 6, pp. 386–408, 1958, doi: 10.1037/h0042519.

[11] G. Kutyniok, "The Mathematics of Artificial Intelligence," Mar. 2022, Accessed: May 10, 2023. [Online]. Available: https://arxiv.org/abs/2203.08890v1

[12] S. Sharma, S. Sharma, and A. Athaiya, “ACTIVATION FUNCTIONS IN NEURAL NETWORKS,” *International Journal of Engineering Applied Sciences and Technology*, vol. 04, no. 12, pp. 310–316, May 2020, doi: 10.33564/IJEAST.2020.v04i12.054.

[13] L. Ciampiconi, A. Elwood, M. Leonardi, A. Mohamed, and A. Rozza, “A survey and taxonomy of loss functions in machine learning,” Jan. 2023.

[14] A. Lopez, D. Math, M. Professor, and C.-K. Li, “Neural Networks: The Backpropagation Algorithm”.

[15] J. Zhang, “Gradient Descent based Optimization Algorithms for Deep Learning Models Training,” Mar. 2019, Accessed: May 11, 2023. [Online]. Available: https://arxiv.org/abs/1903.03614v1

[16] D. P. Kingma and J. Lei Ba, “ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION”.

[17] M. Decuyper, M. Stockhoff, S. Vandenberghe, al -, and X. Ying, “An Overview of Overfitting and its Solutions,” *J Phys Conf Ser*, vol. 1168, no. 2, p. 022022, Feb. 2019, doi: 10.1088/1742-6596/1168/2/022022.

[18] N. Srivastava, G. Hinton, A. Krizhevsky, and R. Salakhutdinov, “Dropout: A Simple Way to Prevent Neural Networks from Overfitting,” *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, 2014.

[19] K. O’Shea and R. Nash, “An Introduction to Convolutional Neural Networks,” *Int J Res Appl Sci Eng Technol*, vol. 10, no. 12, pp. 943–947, Nov. 2015, doi: 10.22214/ijraset.2022.47789.

[20] A. Tropsha, “Best practices for QSAR model development, validation, and exploitation,” *Mol Inform*, vol. 29, no. 6–7, pp. 476–488, Jul. 2010, doi: 10.1002/MINF.201000061.

[21] A. Cherkasov *et al.*, “QSAR Modeling: Where Have You Been? Where Are You Going To?,” 2013, doi: 10.1021/jm4004285.

[22] J. Ma, R. P. Sheridan, A. Liaw, G. E. Dahl, and V. Svetnik, “Deep neural nets as a method for quantitative structure-activity relationships,” *J Chem Inf Model*, vol. 55, no. 2, pp. 263–274, Feb. 2015, doi: 10.1021/CI500747N/SUPPL_FILE/CI500747N_SI_003.PDF.

[23] Robin J Wilson, “Introduction to Graph Theory Fourth edition”: Available at: https://dl.acm.org/doi/10.5555/22577

[24] O. Wieder *et al.*, “A compact review of molecular property prediction with graph neural networks,” *Drug Discov Today Technol*, vol. 37, pp. 1–12, Dec. 2020, doi: 10.1016/J.DDTEC.2020.11.009.

[25] L. David, A. Thakkar, R. Mercado, and O. Engkvist, “Molecular representations in AI-driven drug discovery: a review and practical guide,” *J Cheminform*, vol. 12, no. 1, p. 56, Dec. 2020, doi: 10.1186/s13321-020-00460-5.

[26] J. Zhou *et al.*, “Graph Neural Networks: A Review of Methods and Applications,” *AI Open*, vol. 1, pp. 57–81, Dec. 2018, doi: 10.1016/j.aiopen.2021.01.001.

[27] T. N. Kipf and M. Welling, “Semi-Supervised Classification with Graph Convolutional Networks,” *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*, Sep. 2016, Accessed: May 15, 2023. [Online]. Available: https://arxiv.org/abs/1609.02907v4

[28] W. L. Hamilton, R. Ying, and J. Leskovec, “Inductive Representation Learning on Large Graphs,” *Adv Neural Inf Process Syst*, vol. 2017-December, pp. 1025–1035, Jun. 2017, Accessed: May 16, 2023. [Online]. Available: https://arxiv.org/abs/1706.02216v4

[29] P. Veličković, A. Casanova, P. Liò, G. Cucurull, A. Romero, and Y. Bengio, “Graph Attention Networks,” *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, Oct. 2017, doi: 10.1007/978-3-031-01587-8_7.

[30] S. Brody, U. Alon, and E. Yahav, “How Attentive are Graph Attention Networks?,” May 2021, Accessed: May 16, 2023. [Online]. Available: https://arxiv.org/abs/2105.14491v3

[31] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, “Neural Message Passing for Quantum Chemistry,” 2017.

[32] D. Weininger, “SMILES, a Chemical Language and Information System: 1: Introduction to Methodology and Encoding Rules,” *J Chem Inf Comput Sci*, vol. 28, no. 1, pp. 31–36, Feb. 1988, doi: 10.1021/CI00057A005/ASSET/CI00057A005.FP.PNG_V03.

[33] G. E. Hein, “Kekulé and the Architecture of Molecules”, Accessed: May 12, 2023. [Online]. Available: https://pubs.acs.org/sharingguidelines

[34] S. Ryu, J. Lim, S. H. Hong, and W. Y. Kim, “Deeply learning molecular structure-property relationships using attention- and gate-augmented graph convolutional network,” May 2018, doi: 10.1039/b000000x/find.

[35] J. J. Irwin and B. K. Shoichet, “ZINC--a free database of commercially available compounds for virtual screening.,” *J Chem Inf Model*, vol. 45, no. 1, pp. 177–82, 2005, doi: 10.1021/ci049714+.

[36] S. K. Bhal, “Application Note LogP-Making Sense of the Value”, Accessed: May 19, 2023. [Online]. Available: www.acdlabs.com

[37] S. Prasanna and R. Doerksen, “Topological Polar Surface Area: A Useful Descriptor in 2D-QSAR,” *Curr Med Chem*, vol. 16, no. 1, pp. 21–41, Dec. 2008, doi: 10.2174/092986709787002817.

[38] P. Ertl and A. Schuffenhauer, “Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions,” *J Cheminform*, vol. 1, no. 1, p. 8, Dec. 2009, doi: 10.1186/1758-2946-1-8.

[39] S. Skoraczy´nski, G. S. Skoraczy´nski, M. Kitlas, and A. Gambin, “Critical assessment of synthetic accessibility scores in computer-assisted synthesis planning”, Accessed: May 23, 2023. [Online]. Available: https://github.com/grzsko/ASAP.

[40] V. Plevris, G. Solorzano, N. P. Bakas, and M. E. A. Ben Seghier, “INVESTIGATION OF PERFORMANCE METRICS IN REGRESSION ANALYSIS AND MACHINE LEARNING-BASED PREDICTION MODELS,” *World Congress in Computational Mechanics and ECCOMAS Congress*, 2022, doi: 10.23967/ECCOMAS.2022.155.

[41] Rdkit: Available at: https://www.rdkit.org/

[42] P. Cerda, G. Varoquaux, and B. Kégl, “Similarity encoding for learning with dirty categorical variables,” *Mach Learn*, vol. 107, pp. 1477–1494, 2018, doi: 10.1007/s10994-018-5724-2.

[43] Z. Xiong *et al.*, “Pushing the boundaries of molecular representation for drug discovery with the graph attention mechanism,” *J Med Chem*, vol. 63, no. 16, pp. 8749–8760, Aug. 2020, doi: 10.1021/ACS.JMEDCHEM.9B00959/SUPPL_FILE/JM9B00959_SI_001.PDF.

[44] Pytorch: Available at: https://pytorch.org/docs/stable/index.html

[45] Pytorch Geometric: Available at: https://pytorch-geometric.readthedocs.io/en/latest/

[46] Scikit-Learn: Available at: https://scikit-learn.org/stable/

[47] M. Popova, O. Isayev, and A. Tropsha, “Deep Reinforcement Learning for de-novo Drug Design”.