Μεταπτυχιακό Πρόγραμμα “ Πληροφοριακά Συστήματα & Υπηρεσίες ”

Ειδίκευση “ Μεγάλα Δεδομένα και Αναλυτική ”

Τμήμα Ψηφιακών Συστημάτων

“ Μοντέλο Πρόβλεψης Μετοχικών Τιμών βασισμένο σε Γενετικούς αλγορίθμους και Νευρωνικά Δίκτυα ”

Κεφαλά Μαργαρίτα

Επιβλέπουσα: Αναπλ. Καθηγήτρια Μαρία Χαλκίδη

Πανεπιστήμιο Πειραιά

Φεβρουάριος 2022

Στους Γονείς μου Κατερίνα Κ. και Κωνσταντίνο Κ.

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα πρώτα από όλους να ευχαριστήσω την καθηγήτριά μου κυρία Μαρία Χαλκίδη για την τόση υπομονή και επιμονή της σε εμένα.

Στη συνέχεια θα ήθελα να ευχαριστήσω ξεχωριστά τη μητέρα και τον πατέρα μου που ποτέ τους δε σταμάτησαν να πιστεύουν σε εμένα και να με στηρίζουν με πολλές φορές πάνω από τις δυνάμεις τους εφόδια, υλικά και μη.

ΠΕΡΙΛΗΨΗ

Στην παγκόσμια οικονομία όλα επηρεάζονται αλυσιδωτά, για αυτό ακριβώς το λόγο η πρόβλεψη των τιμών του χρηματιστηρίου είναι αναγκαία όχι μόνο για επενδυτές αλλά και για άτομα που δεν ενασχολούνται με αυτό. Για παράδειγμα έστω ότι η μετοχική τιμή μιας εγχώριας τράπεζας ξεκινά και πέφτει ραγδαία, αυτομάτως σηματοδοτεί τη δυσχερή θέση της εν λόγω με αποτέλεσμα οι καταθέτες να ξεκινούν τις μαζικές και ογκώδεις αναλήψεις. Οι δανειολήπτες αυτής θα έρθουν αντιμέτωποι με μεγαλύτερα επιτόκια δανεισμού, θα «αγοράζουν χρήμα ακριβότερα». Έτσι ένας εκτός χρηματοοικονομικού κύκλου επιχειρηματίας που έχει άμεση ανάγκη από ρευστό δεν θα μπορέσει να δανειστεί από την εν λόγω τράπεζα με αποτέλεσμα ακόμη και να έρθει στη δυσμενή θέση να κλείσει την επιχείρησή του.

Η αγορά των μετοχών θεωρείται ένα χαοτικό, περίπλοκο, εύθραυστο και δυναμικό περιβάλλον. Ωστόσο η πρόβλεψη των μετοχικών τιμών αποτελεί μια από τις πιο πολυσυζητημένες προκλήσεις. Η εξέλιξη της Μηχανικής Μάθησης έχει παρουσιάσει αξιοσημείωτα αποτελέσματα στην αναγνώριση ήχου, ταξινόμηση εικόνας. Εύλογα λοιπόν θεωρείται ότι οι μέθοδοι που εφαρμόζονται στην επεξεργασία ψηφιακών σημάτων μπορούν και να εφαρμοστούν σε χρηματιστηριακές τιμές καθώς και τα δυο επίπεδα αποτελούν δεδομένα χρονοσειρών.

Στόχος της παρούσας διπλωματικής εργασίας είναι η πρόβλεψη μετοχικών τιμών με μοντέλο το οποίο εκπαιδεύεται σύμφωνα με την Ενισχυτική Μηχανική Μάθηση κάνοντας χρήση των Γενετικών Αλγορίθμων και των Τεχνητών Νευρωνικών Δικτύων. Ως δεδομένα για έλεγχο και επαλήθευση του μοντέλου μας χρησιμοποιήθηκαν ελληνικών τραπεζών μετοχικές τιμές, όπως έχουν οριστεί στο Χρηματιστήριο Αθηνών.

Πίνακας περιεχομένων

Κεφάλαιο 1

ΕΙΣΑΓΩΓΗ

1. Περιγραφή προβλήματος δεδομένων και βιβλιογραφία

Η πρόβλεψη με ακρίβεια πορείας της μετοχικής αγοράς φάνταζε απίθανο παλαιότερα. Η Υπόθεση της Αποτελεσματικότητας της Αγοράς θεωρεί ότι οι τιμές των μετοχών αντανακλούν όλη την ήδη γνωστή πληροφορία, ενώ η νέα πληροφορία είναι που κάνει απρόβλεπτη την μετοχική τιμή. Τις τελευταίες δεκαετίες έχει γίνει άφθονη έρευνα σε αυτόν τον τομέα, η προσέλκυση καλών αποδόσεων έχει οδηγήσει σε μυριάδες μεθόδους για την πρόβλεψη των τιμών. Ωστόσο παραμένει ως μια απαιτητική εργασία κατά τους ερευνητές η πρόβλεψη των μετοχικών τιμών σε μη γραμμικές και μη στάσιμες χρηματοοικονομικές χρονοσειρές [5].

Στην εν λόγω διπλωματική εργασία θα περιγράψουμε την περίπτωση πρόβλεψης χρονοσειρών οικονομικών δεδομένων σε διάστημα Ν ημερών με χρήση της μεθόδου της ενισχυτικής μηχανικής μάθησης (Reinforcement Νευρωνικών Δικτύων.

Συνολικά θα ακολουθήσουν τέσσερα κεφάλαια. Το Κεφάλαιο 2 αναφέρεται στη μηχανική μάθηση, τους τρόπους εκπαίδευσης και τους βασικότερους αλγορίθμους ανά περίπτωση εκπαίδευσης. Το Κεφάλαιο 3 περιέχει τη μεθοδολογία ενισχυτικής μηχανικής μάθησης (Reinforcement learning) και τις μεθόδους που θα χρησιμοποιήσουμε και στην τελική μας ανάλυση. Το Κεφάλαιο 4 αναφέρεται στις χρονοσειρές οικονομικών δεδομένων και στα μετοχικά προϊόντα, στοιχεία τα οποία θα χρησιμοποιήσουμε στην ανάλυσή μας. Στο Κεφάλαιο 5 λαμβάνει χώρα η εφαρμογή μας - Reinforcement learning σε χρονοσειρές μετοχικών τιμών, την εξέλιξη αυτών κ.α. Τέλος στο Κεφάλαιο 6 δίνουμε τα αποτελέσματα και τα συμπεράσματα της μελέτης μας.

Κεφάλαιο 2

Οικονομικές Χρονοσειρές και Μετοχικό Προϊόν

Στο παρόν κεφάλαιο αναλύονται οι βασικές ορολογίες στις οποίες βασίζονται η ανάλυση και πρόβλεψης της διπλωματικής εργασίας.

2.1. Μετοχικό Προϊόν

Η μετοχή μια εταιρίας ή μιας τράπεζας αποτελεί ένα ασφάλιστρο το οποίο αντιπροσωπεύει ένα μερίδιο ιδιοκτησίας αυτής. Όταν αγοράζεται μια μετοχή είναι σαν να αποκτάται ένα μικρό κομμάτι της εταιρίας ή της τράπεζας, το αποκαλούμενο μερίδιο. Οι επενδυτές αγοράζουν μετοχές κατά κύριο λόγο «ποντάροντας» στην άνοδο της τιμής αυτής ώστε αργότερα να την πουλήσουν και να έχουν κέρδος από αυτή.

Για τις εταιρίες ή τις τράπεζες που εκδίδουν μετοχές στόχος τους είναι να συλλέξουν χρήματα για να αναπτυχθούν και να επενδύσουν με τη σειρά τους στην επιχείρησή τους. Για τους επενδυτές, είναι ένας τρόπος να κερδίσουν από τα χρήματά τους και να ξεπεράσουν τον πληθωρισμό με την πάροδο του χρόνου.

2.2. Χρονοσειρές και σειρές Χρηματιστηρίων

Από τα πρώτα χρόνια της λειτουργίας των χρηματιστηριακών αγορών παρουσιάστηκε έντονο ενδιαφέρουν από τους «παίχτες» της αγοράς για μια μοντελοποίηση, με σκοπό το κέρδος, των συναλλαγών. Υπάρχουν παργια αξιοποίηση μαθηματικών τύπων για προβλέψεις στην Wall Street. Η μοντελοποίηση και η πρόβλεψη οικονομικών χρονοσειρών παραμένει σήμερα ένα πολύ δύσκολο και απαιτητικό πρόβλημα.[1]

Για την επιστημονική κοινότητα η πολυπλοκότητα, η δυναμική και το θορυβώδους της φύσης των προβλέψεων αποτελεί αντικείμενο μελέτης. Χαρακτηριστικό παράδειγμα βραχυπρόθεσμης πρόβλεψης είναι οι δείκτες FTSE100 και DAX οι οποίοι διαπραγματεύονται στην Αγορά Παραγώγων,

Συμβόλαια Μελλοντικής Εκπλήρωσης (Futures Contract). Ένα Συμβόλαιο Μελλοντικής Εκπλήρωσης είναι μια σύμβαση μεταξύ δύο μερών και η τιμή του καθορίζεται από τον υπολογισμό της διαφοράς μεταξύ της τιμής που αγοράστηκε και την τιμή κλεισίματος του δείκτη την ημέρα λήξης της σύμβασης.

Την περίοδο της οικονομικής κρίσης οι αγορές έχουν απορυθμιστεί δημιουργώντας πολλά προβλήματα στις μεθόδους πρόβλεψης. Ο χειρισμός της πολυπλοκότητας και της μη γραμμικότητας που υπάρχουν στις οικονομικές χρονοσειρές, ιδιαίτερα κατά την διάρκεια τις τελευταίας οικονομικής κρίσης, έχουν οδηγήσει σε δυσκολίες στην ρύθμιση των παραμέτρων των αλγορίθμων και στο πρόβλημα της υπερεκπαίδευσης.

Χαρακτηριστική είναι η ελάχιστη μεταβλητότητα κατά την διάρκεια μιας μέρας των ελληνικών τραπεζικών μετοχών, αλλά οι μη γραμμικότητες που παρουσιάζουν με απότομες αυξητικές και κυρίως καθοδικές μεταβολές.

2.3. Μελέτες πρόβλεψης Χρονοσειρών με χρήση του διαδικτύου

Πολλοί ερευνητές έχουν ασχοληθεί μέχρι σήμερα με την πρόβλεψη ελληνικών και ξένων μετοχών των μετοχών και δεικτών του χρηματιστηρίου με χρήση πληροφοριών από το διαδίκτυο. Η προσπάθεια αυτή της πρόβλεψης εκτός από καθαρά ερευνητική, έχει περάσει και στον επιχειρηματικό κλάδο, με τα ποσά που δαπανούνται να είναι συνεχώς αυξανόμενα.[2]

Η πρόβλεψη μετοχών και δεικτών με τέτοιες μεθόδους απαιτούν ισχυρά μαθηματικά μοντέλα και συνδυασμό πολλών διαφορετικών μεθόδων, κυρίως υπολογιστικής νοημοσύνης. Οι μελέτες επικεντρώνονται κυρίως σε μετοχές με παγκόσμια εμβέλεια των μεγαλύτερων χρηματιστηρίων του κόσμου.

Έχουν γίνει σημαντικές μελέτες στην πρόβλεψη μετοχών του χρηματιστηρίου [2]. Η πρόβλεψη των μετοχών, μέχρι το πρόσφατο παρελθόν, συνηθιζόταν να γίνεται με την μελέτη των δεδομένων για την μετοχή στο παρελθόν και με διάφορους τεχνικούς οικονομικούς δείκτες [2].

Όπως όμως πολλές σύγχρονες έχουν δείξει, η πρόβλεψη είναι δυνατή και με την χρήση πληροφορίας από κείμενα αν βασιστούμε στην «υπόθεση της αποδοτικής αγοράς», όπου όλη η διαθέσιμη πληροφορία (είτε άρθρα είτε

πληροφορία όπως αυτή ανεβαίνει σε διάφορες Web 2.0 πλατφόρμες) μπορεί δυνητικά να επηρεάσει την αξία μιας μετοχής. [6]

Η «υπόθεση της αποδοτικής αγοράς» αντικατοπτρίζει την θεωρία των τυχαίων διαδρομών (random walks) στους γράφους. Παρά τις πρόσφατες εξελίξεις στην επεξεργασία της φυσικής γλώσσας και στην εξόρυξη δεδομένων, όταν τα δεδομένα τείνουν να αυξάνονται, τόσο σε αριθμό όσο και σε γνωρίσματα, σημαντικός αριθμός αλγορίθμων αντιμετωπίζουν σημαντικά προβλήματα που οδηγούν σε φτωχή προβλεπτική ικανότητα. [7]

Οι λύσεις που προτείνονται για την πρόβλεψη χρονοσειρών είναι αρκετές όπως με την χρήση Monte Carlo Αλυσίδων Markov με συμπερασματολογία κατά Bayes (Markov Chain Monte Carlo (MCMC) at Bayesian Inference), η οποία υπολογίζει-εκτιμά δεσμευμένες κατανομές πιθανοτήτων στις δομές που λαμβάνονται από ένα αυξητικό-δέντρο Naïve Bayes αλγόριθμο (Tree-Augmented Naive Bayes Algorithm (TAN)) [2]

Η μεγάλη διαστασημότητα (high-dimensionality) και σπανιότητα των δεδομένων του χρηματιστηρίου, έχει ως αποτέλεσμα η πλειοψηφία των πιο διαδεδομένων αλγορίθμων εξόρυξης δεδομένων να μην δίνουν το καλύτερο αποτέλεσμα είτε λόγω προβλημάτων σύγκλισης είτε προβλημάτων [1].

Η εξόρυξη δεδομένων τα τελευταία χρόνια χρησιμοποιείται για την αναζήτηση αξιόπιστων προτύπων ή/και λογικών σχέσεων μεταξύ των μεγεθών μιας σειράς. Χρησιμοποιείται κατάλληλη επικύρωση για την εύρεση των μοτίβων με τα ανιχνευμένα μοτίβα μεταξύ των σύνολο δεδομένων Η έρευνα που σχετίζεται με το χρηματιστήριο επικεντρώνεται σε μεγάλο βαθμό στην αγορά και στην πώληση. Αλλά αποτυγχάνει να αντιμετωπίσει τη διάσταση και την προσδοκία ενός αφελούς επενδυτή. Η γενική τάση προς το χρηματιστήριο μεταξύ της κοινωνίας είναι ότι είναι πολύ επικίνδυνο για επένδυση ή δεν είναι κατάλληλο για εμπορικές συναλλαγές. [5]

Η εποχιακή διακύμανση και η σταθερή ροή οποιουδήποτε δείκτη είναι σημαντικό να λαμβάνεται υπόψη με στόχο μια απόφαση επένδυσης [4].

Οι επενδυτές ενδιαφέρονται πολύ να μάθουν την προηγούμενη τάση, την εποχιακή ανάπτυξη ή τις παραλλαγές σε κάθε χρονική στιγμή. Μια γενική άποψη ή προσδοκία είναι ότι πρέπει να δώσει μια ολιστική άποψη της μετοχής αγορά.[3]

Δεδομένου ότι, είναι απαραίτητο να προσδιοριστεί ένα μοντέλο για να δείξει την τάση με επαρκείς πληροφορίες για το επενδυτής να πάρει μια απόφαση. Προτάσεις σε αυτά είναι η ανάλυση συχνοτήτων, γραμμικά μοντέλα ARIMA ,στατιστικές μέθοδοι, διαδικασίες τεχνητής νοημοσύνης [4].

Μελέτες επίσης έχουν δείξει ότι γενικά χαρακτηριστικά όπως οι συχνότητες που εμφανίζονται στις σειρές , ο θόρυβος κ.α. μπορούν να αποτελέσουν σημαντικές πληροφορίες για την βραχυχρόνια και μακροχρόνια πρόβλεψη των χρονοσειρών [8].

Σημαντικά μεγέθη φυσικά αποτελούν και τα στατιστικά χαρακτηριστικά των σειρών όπου οδηγούν σε σημαντική γνώση σε σχέση με την συμπεριφορά των σειρών. [9].

Κεφάλαιο 3

ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ (MACHINE LEARNING)

3.1. Ορισμός Μηχανικής Μάθησης

Ο όρος μάθηση καλύπτει ένα ευρύ φάσμα εννοιών που είναι δύσκολο να καθοριστούν. Ορισμοί του όρου μάθησης σε λεξικά περιλαμβάνουν φράσεις όπως «απόκτηση γνώσης», «κατανόηση», «δεξιότητα» και «τροποποίηση συμπεριφοράς βάσει εμπειρίας». Κατά καιρούς έχουν δοθεί διάφοροι ορισμοί της μάθησης όπως «μάθηση είναι να κάνουμε χρήσιμες αλλαγές στο μυαλό μας» ή «μάθηση είναι η δημιουργία ή η αλλαγή της αναπαράστασης των εμπειριών». Γενικώς θα μπορούσαμε να πούμε ότι η μάθηση είναι μια διαδικασία που μεταβάλει κάποιο σύστημα προς το “καλύτερο” όπως και αν αυτό ορίζεται. Η μηχανική μάθηση είναι η διαδικασία με την οποία τα υπολογιστικά συστήματα μπορούν να κατευθυνθούν προς τη βελτίωση της απόδοσης τους με το χρόνο και συνήθως αντιμετωπίζουν προβλήματα τεχνητής νοημοσύνης όπως αναγνώριση, διάγνωση, σχεδιασμός, πρόβλεψη κ.α. Οι αλγόριθμοι μηχανικής μάθησης κατηγοριοποιούνται βάσει της μεθόδου με την οποία μαθαίνουν σε δυο μεγάλες κατηγορίες: τη μάθηση με επίβλεψη (Supervised Learning), τη μάθηση χωρίς επίβλεψη (Unsupervised learning) και την ενισχυτική μάθηση (Reinforcement Learning) [17].

3.2. Μηχανική Μάθηση και Χρονοσειρές

Στην περιγραφική στατιστική, μια χρονοσειρά ορίζεται και αναλύεται ως ένα σύνολο τυχαίων μεταβλητών ταξινομημένων σε σχέση με το χρόνο. Οι χρονοσειρές μελετώνται τόσο για την ερμηνεία ενός φαινομένου, για τον προσδιορισμό των συνιστωσών μιας τάσης, της κυκλικότητας, της εποχικότητας, την πρόβλεψη των μελλοντικών αξιών του.

Στον χρηματοοικονομικό τομέα, ο κύριος στόχος είναι να αναγνωριστούν οι τάσεις, η εποχιακή συμπεριφορά και η συσχέτιση μέσω της χρήσης της τεχνικής ανάλυσης χρονοσειρών και της παραγωγής φίλτρων με βάση τις προβλέψεις.

Αφού ληφθούν στατιστικά δεδομένα εξόδου οικονομικών χρονοσειρών, μπορούν να χρησιμοποιηθούν για τη δημιουργία προσομοιώσεων μελλοντικών γεγονότων. Η προσομοίωση μας βοηθά να προσδιορίσουμε τον αριθμό των συναλλαγών, το αναμενόμενο κόστος και τις αποδόσεις συναλλαγών, τις απαιτούμενες οικονομικές και τεχνικές επενδύσεις, αρκετούς κινδύνους στις συναλλαγές κ.λπ.

Η αναγνώριση της σχέσης μεταξύ των χρονοσειρών και άλλων ποσοτήτων δίνει την δυνατότητα να αναλύσουμε για να βελτιώσουμε μια σειρά συναλλαγών. Για παράδειγμα, για να γνωρίζουμε τη διασπορά του ζεύγους συναλλάγματος και τη διακύμανσή του με μια πρόταση, οι εκτιμώμενες συναλλαγές μπορούν να συναχθούν για μια ορισμένη περίοδο για την πρόβλεψη ευρέως διαδεδομένου για μείωση του κόστους συναλλαγής.

Τα περισσότερα συστήματα μηχανικής μάθησης ορίζουν μοντέλα που μαθαίνουν και εφαρμόζονται σε ένα χρονικό σημείο λαμβάνοντας ελάχιστα ή καθόλου υπόψη τις δυναμικές αλληλεπιδράσεις που έχουν οδηγήσει στην τρέχουσα κατάσταση. Στη μηχανική μάθηση εφαρμόζονται νέοι αυτόνομοι αλγόριθμοι μηχανικής μάθησης οι οποίοι χρησιμοποιούν τα δεδομένα χρονοσειρών και εκπαιδεύονται από αυτά με στόχο την εξαγωγή γνώσης από αυτά. Η γνώση που παράγεται χρησιμοποιείται για να παραχθούν αποτελέσματα όπως π.χ. η πρόβλεψη της επόμενης μέρας.

Παρακάτω αναλύονται σειρά από μεθόδους που χρησιμοποιούνται για την ανάλυση και πρόβλεψη χρονοσειρών με χρήση της μηχανικής μάθησης.

3.3. Μηχανική Μάθηση με Επίβλεψη (Supervised Learning)

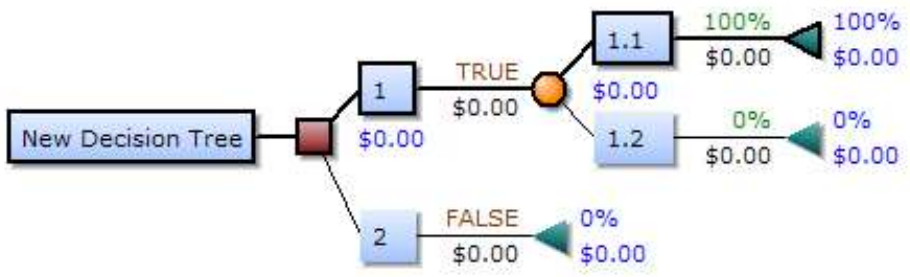
Στη μάθηση με επίβλεψη (Supervised Learning) στόχος του αλγορίθμου είναι μια συνάρτηση που εκφράζει όσο το δυνατόν καλύτερα το μοντέλο που μαθαίνει. Ο αλγόριθμος προσπαθεί να μάθει το μοντέλο από τα δεδομένα εκπαίδευσης που συνήθως είναι ζεύγη τιμών εισόδου και επιθυμητής εξόδου.

Βάσει επαγωγής, η συνάρτηση που κατασκευάζει ο αλγόριθμος προσεγγίζει καλά τα άγνωστα δεδομένα αφού πρώτα εκπαιδευτεί σε γνωστά δεδομένα. Η έξοδος της συνάρτησης μπορεί να είναι μια συνεχής τιμή ή κάποια διακριτή τιμή που προβλέπει μια τάξη. Στην πρώτη περίπτωση έχουμε παλινδρόμηση και στην δεύτερη ταξινόμηση. Παρακάτω θα δούμε τις κυριότερες τεχνικές μάθησης με επίβλεψη [17].

Κυριότερες Τεχνικές Μάθησης με Επίβλεψη

3.3.1. Δέντρα Ταξινόμησης/Απόφασης

Τα δέντρα ταξινόμησης/απόφασης είναι εργαλεία που βοηθάνε στην λήψη μιας απόφασης ή στην προσέγγιση μιας συνάρτησης που έχει ως έξοδο κάποια διακριτή τιμή. Στην μηχανική μάθηση ένα δένδρο ταξινόμησης είναι ένα μοντέλο πρόβλεψης που αντιστοιχεί τις παρατηρήσεις για κάποιο αντικείμενο με την έξοδο. Σε αυτά τα δένδρα τα φύλλα αναπαριστούν τις διακριτές ταξινομήσεις (εξόδους) του δένδρου ενώ κάθε κόμβος του δένδρου ορίζει μια συνθήκη ελέγχου της τιμής κάποιου χαρακτηριστικού. Κάθε κλαδί που ξεκινάει από ένα κόμβο αντιστοιχεί σε διαφορετική τιμή του χαρακτηριστικού που ελέγχει ο κόμβος και οδηγεί σε διαφορετικό φύλλο (έξοδο) [17].



Εικόνα 1. Δέντρο απόφασης

Ένας από τους λόγους που είναι δημοφιλής αυτή η τεχνική μάθησης είναι η ευκολία αναπαράστασης αλλά και κατανόησης της τεχνικής. Σημαντικό στοιχείο στα δένδρα ταξινόμησης είναι το πόσο εύκολα μπορεί να διαβαστεί και επεξηγηθεί το αποτέλεσμα που δίνει. Συνήθως τα αποτελέσματα των δένδρων ταξινόμησης μπορούν να αναπαρασταθούν με απλά μαθηματικά ή με ένα

σύνολο κανόνων if-then (εάν-τότε) και έτσι μπορούν εξηγήσουν γιατί δόθηκε το αποτέλεσμα που δόθηκε.

Ο πιο γνωστός αλγόριθμος της τεχνικής των δένδρων ταξινόμησης είναι ο ID3. Ο αλγόριθμος μπορεί να περιγραφεί σε τρία βήματα [17]:

1. Βρες την ανεξάρτητη μεταβλητή η οποία αν χρησιμοποιηθεί ως κριτήριο διαχωρισμού των δεδομένων εκπαίδευσης θα οδηγήσει σε κόμβους κατά το δυνατό διαφορετικούς σε σχέση με την εξαρτημένη μεταβλητή.
2. Κάνε τον διαχωρισμό.
3. Επανέλαβε τη διαδικασία για κάθε έναν από τους κόμβους που προέκυψαν μέχρι να μην είναι δυνατός περαιτέρω διαχωρισμός.

Με αυτά τα βήματα κατασκευάζεται ένα δένδρο άπληστα βάσει ενός μηχανισμού διαχωρισμού. Συνήθως ο μηχανισμός διαχωρισμού είναι αυτός της εντροπίας της πληροφορίας ο οποίος επιλέγει εκείνη την ανεξάρτητη μεταβλητή που οδηγεί στο πιο συμπαγές δένδρο. Ο αλγόριθμος είναι ευριστικός γιατί δεν εγγυάται ότι το δένδρο που παράγεται είναι το μικρότερο δυνατό.

3.3.2. Μάθηση Κανόνων

Η μάθηση κανόνων είναι η πιο εκφραστική και κατανοητή τεχνική μάθησης για τον άνθρωπο και συνεπώς συμβάλει στην καλύτερη κατανόηση των αποτελεσμάτων. Οι κανόνες μπορεί να προέρχονται και από άλλες αναπαραστάσεις όπως για παράδειγμα τα δένδρα ταξινόμησης. Δυο μεγάλες κατηγορίες κανόνων είναι οι προτασιακοί κανόνες που δεν περιέχουν μεταβλητές και οι κατηγορηματικοί κανόνες που περιέχουν.

Στους προτασιακούς κανόνες δεν απαιτείται η αναπαράσταση σχέσεων μεταξύ των χαρακτηριστικών. Ένας γενικός αλγόριθμος μάθησης τέτοιων κανόνων είναι ο αλγόριθμος σειριακής κάλυψης (Sequential Covering Algorithm). Αυτός ο αλγόριθμος μαθαίνει συριακά το σύνολο των κανόνων που όλοι μαζί καλύπτουν το σύνολο των θετικών παραδειγμάτων. Έχει το πλεονέκτημα να σπάει το πρόβλημα της μάθησης ενός ανεξάρτητου συνόλου κανόνων σε απλούστερα προβλήματα που το καθένα απαιτεί την μάθηση ενός μόνο κανόνα. Στη συνέχεια το τελικό σύνολο κανόνων ταξινομείται σε φθίνουσα

σειρά ακριβείας. Να σημειωθεί ότι ο αλγόριθμος δεν οπισθοδρομεί δεν εγγυάται ότι παράγει το ελάχιστο ή το καλύτερο σύνολο κανόνων [17].

Τα βήματα του αλγορίθμου είναι [17]:

1. Αρχικοποίησε το σύνολο κανόνων με το κενό σύνολο
2. Μάθε_έναν_Κανόνα (Εξαρτημένη_μεταβλητή,Μετβλητές,Παραδείγματα)
3. Αν ο κανόνας ικανοποιεί το κριτήριο απόδοσης:
4. Αφαίρεσε τα θετικά παραδείγματα που κάλυψε ο κανόνας αυτός
5. Πρόσθεσε τον κανόνα στο σύνολο κανόνων
6. Επανέλαβε από το 2, όσο ικανοποιείται το κριτήριο απόδοσης

Η συνάρτηση Μάθε_έναν_Κανόνα:

1. Έστω η βέλτιστη υπόθεση (αρχικά ο πιο γενικός κανόνας) που ταιριάζει με όλα τα παραδείγματα του συνόλου εκπαίδευσης
2. Επανέλαβε όσο υπάρχουν υποψήφιες υποθέσεις:
3. Εξειδίκευσε την βέλτιστη υπόθεση, προσθέτοντας το ζεύγος χαρακτηριστικού-τιμής που βελτιστοποιεί το κριτήριο απόδοσης

Το κριτήριο απόδοσης είναι ένα μέτρο της απόδοσης του κανόνα που το καθορίζει ο χρήστης όπως για παράδειγμα η εντροπία.

Οι κατηγορηματικοί κανόνες μοιάζουν με προτάσεις Horn με την διαφορά ότι μπορεί να περιέχουν άρνηση ατομικών εκφράσεων στο σώμα του κανόνα και δεν επιτρέπουν τη χρήση συναρτησιακών συμβόλων. Ο πιο γνωστός αλγόριθμος μάθησης κατηγορηματικών κανόνων είναι ο FOIL Quilan (1990) που είναι η συνέχεια της μάθησης των προτασιακών κανόνων. Ο αλγόριθμος χρησιμοποιεί ένα ειδικό μέτρο απόδοσης (FOIL-GAIN) που λαμβάνει υπ’ όψη του τυχόν συσχετίσεις μεταβλητών και αναζητά κανόνες που προβλέπουν πότε η δεσμευμένη μεταβλητή είναι αληθής (και όχι πότε είναι ψευδής).

3.3.3. Μάθηση κατά Περίπτωση

Σε αυτή την τεχνική μάθησης τα δεδομένα μάθησης παραμένουν αυτούσια γιατί η τεχνική δεν απαιτεί κάποια μοντελοποίηση ή κωδικοποίηση των δεδομένων. Η βασική αρχή με την οποία δουλεύει η τεχνική είναι να κρατάει όλα τα δεδομένα μάθησης και όταν εμφανιστεί μια νέα περίπτωση προβαίνει σε επεξεργασία των δεδομένων. Αυτού του είδους η μάθηση ονομάζεται αναβλητική μάθηση (lazy learning). Ο χαρακτηριστικός αλγόριθμος αναβλητικής μάθησης είναι αυτός των k-πλησιέστερων γειτόνων (k-Nearest Neighbors). Για κάθε νέα περίπτωση που του δίνεται υπολογίζει την τιμή της περίπτωσης βάση των k πλησιέστερων γειτόνων. Οι πλησιέστεροι γείτονες υπολογίζονται με την ευκλείδεια απόσταση. Συνήθως ο αλγόριθμος των k-πλησιέστερων γειτόνων χρησιμοποιείται όταν τα δεδομένα μας μπορούν να αναπαρασταθούν σε ένα n-διάστατο ευκλείδειο χώρο και όταν τα δεδομένα μας είναι πάρα πολλά. Πλεονεκτήματα αυτής της τεχνικής είναι ότι η εκπαίδευση του αλγορίθμου είναι πολύ γρήγορη (ουσιαστικά δεν υπάρχει κάποια εκπαίδευση), μπορεί να μάθει και πολύπλοκες συναρτήσεις και δεν έχουμε απώλεια πληροφορίας γιατί πολύ απλά κρατάμε όλα τα δεδομένα μας στην μορφή που μας τα έδωσαν. Από την άλλη όμως η τεχνική μπορεί να καθυστερεί όταν γίνονται τα ερωτήματα, όταν δηλαδή έχουμε μια καινούργια περίπτωση, ειδικά αν τα χαρακτηριστικά των δεδομένων μας είναι πολλά και άρα ο αριθμός των διαστάσεων μεγάλος. Επίσης, δεν ξεχωρίζονται τα σημαντικά χαρακτηριστικά από τα λιγότερο σημαντικά μιας και δεν γίνεται κάποια προεπεξεργασία στα δεδομένα έτσι μεγαλώνει ο κίνδυνος να μην πάρουμε καλά αποτελέσματα λόγω του ότι όλα τα χαρακτηριστικά των δεδομένων λαμβάνονται το ίδιον υπόψη [14].

Μια βελτίωση των k-πλησιέστερων γειτόνων είναι ο αλγόριθμος k-πλησιέστερων γειτόνων με βάρη όπου ο κοντινότερος γείτονας έχει μεγαλύτερη επιρροή στο αποτέλεσμα παρά κάποιος πιο απομακρυσμένος. Ως βάρος χρησιμοποιείται το αντίστροφο της απόστασης. Για επιπλέον βελτίωση μπορεί να χρησιμοποιηθούν βάρη και στα χαρακτηριστικά των δεδομένων με τον παρακάτω τύπο

$$dist(x, y) = \sqrt{\sum_{i=0}^{n} w_i (x_i - y_i)^2}$$

όπου w_i το βάρος του χαρακτηριστικού i.

3.3.4. Μάθηση κατά Bayes

Η μάθηση κατά Bayes είναι μια από τις σημαντικότερες και τεχνικές μηχανικής μάθησης. Η διαφορά με τις άλλες τεχνικές είναι ότι βασίζεται σε μια διαφορετική εκδοχή του τι σημαίνει «μαθαίνω» από τα δεδομένα, χρησιμοποιεί πιθανότητες για να εκφράσει την αβεβαιότητα στην συνάρτηση που μαθαίνει. Η μάθηση δεν μπορεί να ξεκινήσει από το τίποτα, συνεπώς γίνεται κάποιου είδος παραδοχή για το μοντέλο που πρόκειται να «μάθουμε», με άλλα λόγια απαιτούνται κάποιες εκ των προτέρων πιθανότητες. Κατά την μάθηση αυτές οι πιθανότητες αναθεωρούνται και έτσι κάθε παράδειγμα μπορεί να μείωση ή να αύξηση την πιθανότητα μια υπόθεση να είναι σωστή. Αυτό δίνει την ευκαιρία στους αλγορίθμους που μαθαίνουν κατά Bayes να μην απορρίπτουν μια υπόθεση εάν δεν είναι σύμφωνη με τα παραδείγματα εκπαίδευσης. Αρνητικό στοιχείο της τεχνικής είναι η απαίτηση για την γνώση πολλών τιμών πιθανοτήτων και όταν αυτές δεν είναι δυνατό να υπολογιστούν, υπολογίζονται κατ' εκτίμηση από παλιότερες υποθέσεις ή από εμπειρική γνώση [14].

Μια απλουστευμένη εκδοχή της κατά Bayes μάθησης είναι ο απλός (αφελής) ταξινομητής Bayes όπου θεωρούνται ότι όλα τα χαρακτηριστικά είναι ανεξάρτητα μεταξύ τους. Για παράδειγμα, ένα φρούτο μπορεί να θεωρηθεί ότι είναι ένα μήλο αν είναι κόκκινο, στρογγυλό και έχει διάμετρο περίπου 4 εκατοστά. Ακόμη και αν τα χαρακτηριστικά αυτά εξαρτώνται από την ύπαρξη των άλλων χαρακτηριστικών, ένας απλός ταξινομητής Bayes θεωρεί ότι όλα αυτά τα χαρακτηριστικά συμβάλουν ανεξάρτητα στην πιθανότητα ότι το φρούτο είναι ένα μήλο. Η ποσότητα P που περιγράφει έναν απλό ταξινομητή Bayes για ένα σύνολο παραδειγμάτων εκφράζει την πιθανότητα να είναι c η τιμή της εξαρτημένης μεταβλητής C με βάση τις τιμές x=($x_1, x_2,$ ..., x_n) των χαρακτηριστικών X=(X_1, X_2, ..., X_n) και δίνεται από τη σχέση:

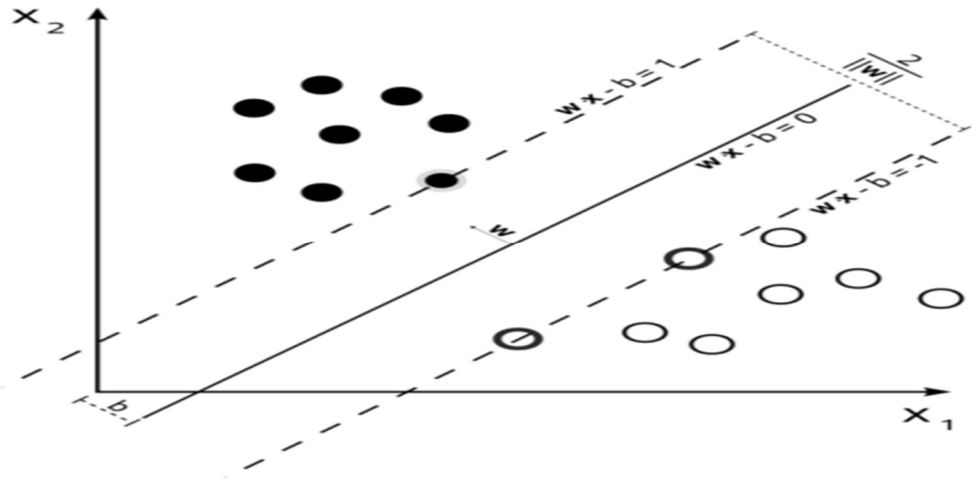
$$P(c|x) = P(c)\prod_i P(x_i|c)$$

όπου τα χαρακτηριστικά X_i θεωρούνται ανεξάρτητα μεταξύ τους. Ο υπολογισμός της παραπάνω ποσότητας για ένα σύνολο N παραδειγμάτων γίνεται με βάση τις σχέσεις:

- P(c) = N(c) / N ,
- P(x_i|c) = N(x_i,c) / N(c), για χαρακτηριστικό X_iμε διακριτές τιμές,
- P(x_i|c) = g(x_i, μ_c,σ_c^2), για χαρακτηριστικό X_i με αριθμητικές τιμές,
- όπου N(c) είναι ο αριθμός των παραδειγμάτων που έχουν στην εξαρτημένη μεταβλητή την τιμή c, N(x_i,c) είναι ο αριθμός των παραδειγμάτων που έχουν για το χαρακτηριστικό X_i και την εξαρτημένη μεταβλητή, τιμές x_i και c αντίστοιχα, και g(x_i,μ_c,σ_c^2), είναι η συνάρτηση πυκνότητας πιθανότητας Gauss με μέσο όρο μ_c και διασπορά σ_c για το χαρακτηριστικό X_i.

3.3.5. Μηχανές Διανυσμάτων Υποστήριξης

Οι μηχανές διανυσμάτων υποστήριξης (**Support Vector Machines** - **SVM**) προτάθηκαν από τον Vapkin και στηρίζονται στη θεωρία στατιστικής μάθησης. Έχουν πολύ καλή απόδοση ως ταξινομητές, χαμηλό υπολογιστικό κόστος και είναι ανθεκτικοί σε περιπτώσεις υπερπροσαρμογής. Σε ένα σύνολο δεδομένων με γραμμικά διαχωρίσιμες τάξεις οι μηχανές διανυσμάτων υποστήριξης προσπαθούν να βρουν το υπερεπίπεδο που επιτυγχάνει το μέγιστο διαχωρισμό μεταξύ των τάξεων.



Εικόνα 2. Υπερεπίπεδο μεγίστου περιθωρίου

Τα υποδείγματα με τη μικρότερη απόσταση από το υπερεπίπεδο μεγίστου περιθωρίου καλούνται διανύσματα υποστήριξης (support vectors) και το σύνολό τους μπορεί να καθορίσει μοναδικά το υπερεπίπεδο διαχωρισμού χωρίς να χρειάζονται τα υπόλοιπα υποδείγματα [14].

Οι μηχανές διανυσμάτων υποστήριξης μπορούν να διαχωρίσουν δυο τάξεις που δεν είναι γραμμικά διαχωρίσιμες με τη χρήση συναρτήσεων πυρήνα (kernel functions) (μη γραμμικός μετασχηματισμός) που μετασχηματίζουν τα υποδείγματα εστί ώστε να διαχωρίζονται γραμμικά με το ελάχιστο σφάλμα ταξινόμησης. Η τεχνική αυτή έχει σημαντική εφαρμογή στην κατηγοριοποίηση κειμένων και αναγνώριση προσώπων.

3.3.6. Μάθηση Εννοιών

Η μάθηση εννοιών ορίζεται ως η αναζήτηση και καταγραφή των χαρακτηριστικών που μπορούν να χρησιμοποιηθούν για τη διάκριση υποδειγμάτων διαφόρων κατηγοριών. Είναι μια τεχνική επαγωγικής μάθησης κατά την οποία το σύστημα που μαθαίνει δέχεται παραδείγματα που ανήκουν σε μια έννοια αλλά και παραδείγματα που δεν ανήκουν σε αυτή. Στόχος του συστήματος, όπως και σε άλλες τεχνικές μάθησης, είναι να δημιουργήσει ένα μοντέλο που να περιγράφει την έννοια που μαθαίνει, έτσι ώστε όταν του δοθεί μια άγνωστη περίπτωση να μπορεί να κρίνει αν ανήκει σε αυτή την έννοια ή όχι. Ένας αλγόριθμος αυτής της τεχνικής είναι ο αλγόριθμος απαλοιφής υποψηφίων (Candidate Elimination Algorithm) [14].

Ο αλγόριθμος απαλοιφής υποψηφίων περιορίζει το χώρο αναζήτησης κάνοντας γενικεύσεις και εξειδικεύσεις σε κάποιες αρχικές έννοιες με βάση τα δεδομένα εκπαίδευσης. Διατηρεί δύο σύνολα, G και S που μαζί αποτελούν ολόκληρο το σύνολο αναζήτησης. Το σύνολο G περιέχει τις πιο γενικές υποψήφιες υποθέσεις δηλαδή τις έννοιες, ενώ το S τις πιο εξειδικευμένες υποψήφιες υποθέσεις. Με βάση τα παραδείγματα ο αλγόριθμος περιορίζει το σύνολο G κάνοντας εξειδικεύσεις και διευρύνει το σύνολο S κάνοντας γενικεύσεις έως ότου εξαντληθούν τα στοιχεία των δύο συνόλων. Εξετάζοντας όλα τα παραδείγματα τα σύνολα G και S περιλαμβάνουν όλες τις υποθέσεις που ταξινομούν σωστά τα παραδείγματα.

3.4. Μηχανική Μάθηση χωρίς Επίβλεψη (Unsupervised Learning)

Στην μάθηση χωρίς επίβλεψη (Unsupervised Learning) στόχος είναι η αυτοματοποιημένη παραγωγή «γνώσης». Οι τεχνικές μάθησης αυτής της κατηγορίας προσπαθούν να ανακαλύψουν συσχετίσεις και ομάδες στα μη ταξινομημένα δεδομένα που τους δίνονται .

Κανόνες Συσχέτισης

Η εξόρυξη κανόνων συσχέτισης (association rules) έχει ως σκοπό την εύρεση συσχετίσεων μεταξύ αντικειμένων μιας βάσης δεδομένων. Ένας κανόνας συσχέτισης λέει ότι εάν επιλεγεί ένα σύνολο αντικειμένων X από ένα μεγαλύτερο σύνολο I τότε είναι πιθανό να επιλεγεί και το αντικείμενο Y. Μαθηματικά εκφράζεται ως $R : X \rightarrow Y$. Η ισχύς ενός κανόνα ελέγχεται μέσω δύο ποσοτήτων, την υποστήριξη και την εμπιστοσύνη. Η υποστήριξη supp(X) ορίζεται ως ο αριθμός των εγγράφων που περιέχουν το αντικείμενο X ως υποσύνολο. Ενώ η εμπιστοσύνη ορίζεται ως ο λόγος μεταξύ της υποστήριξης της κεφαλής του κανόνα προς την υποστήριξη του σώματος. Δηλαδή για τον κανόνα $\boldsymbol{R : X \rightarrow Y}$ έχουμε [14]:

υποστήριξη $\boldsymbol{supp(R) = supp(X \cup Y)}$

και εμπιστοσύνη $\boldsymbol{conf(R) = \frac{supp(X \cup Y)}{supp(X)}}$

Το πρόβλημα με τους κανόνες συσχετίσεις μιας βάσης δεδομένων είναι ο μεγάλος τους αριθμός που είναι της τάξης του $\boldsymbol{O}(\boldsymbol{3^n})$, όπου n είναι ο αριθμός των αντικειμένων της βάσης. Έτσι προκειμένου να επιλεγούν κανόνες που έχουν κάποια πρακτική αξία επιλέγονται τα σύνολα με υποστήριξη μεγαλύτερη ή ίση με την τιμή κατωφλίου s και μετά παράγονται όλοι οι κανόνες που προκύπτουν από τα σύνολα αυτά και έχουν εμπιστοσύνη μεγαλύτερη ή ίση με την τιμή κατωφλίου c.

Η αξία των κανόνων συσχέτισης είναι μεγάλη επειδή συνδυάζουν πολλά πλεονεκτήματα. Μπορούν να ανταπεξέλθουν σε πραγματικά προβλήματα μεγάλου μεγέθους με θόρυβο σε αντίθεση με άλλες τεχνικές μάθησης όπου η εκπαίδευση των δεδομένων πρέπει να γίνει σε καθαρά και προσεκτικά επιλεγμένα δεδομένα. Οι κανόνες από μόνοι τους είναι μια αναπαράσταση πιο κατανοητή στον άνθρωπο από κάθε άλλη αναπαράσταση γνώσης όπως δένδρα, ομάδες κτλ. Επίσης οι αλγόριθμοι μάθησης κανόνων συσχέτισης επιτυγχάνουν πολύ καλή απόδοση σε μεγάλο όγκο δεδομένων, κλιμακώνονται εύκολα και μπορούν να χειριστούν σύνθετα δεδομένα όπως χωρικά, πολυμεσικά, χρονικά κα. Το βασικότερο μειονέκτημα των κανόνων συσχέτισης είναι η επιλογή χρήσιμων και ουσιωδών κανόνων από το μεγάλο σύνολο που επιστρέφεται. Παρακάτω θα δούμε τις κυριότερες τεχνικές μάθησης χωρίς επίβλεψη.

Κυριότερες Τεχνικές Μάθησης Χωρίς Επίβλεψη

3.4.1. Ο αλγόριθμος Apriori

Ο πιο γνωστός αλγόριθμος εξόρυξης κανόνων συσχέτισης είναι ο Apriori. Ο αλγόριθμος στηρίζεται στην ιδιότητα a priori σύμφωνα με την οποία «αν ένα σύνολο αντικειμένων s είναι συχνό, τότε όλα τα υποσύνολα του s είναι επίσης συχνά». Ένα σύνολο λέγεται συχνό όταν έχει υποστήριξη μεγαλύτερη ή ίση της τιμής κατωφλίου. Ο Apriori δουλεύει σε δυο βήματα, στο πρώτο βήμα βρίσκει όλα τα συχνά σύνολα αντικειμένων και στο δεύτερο βήμα δημιουργεί τους κανόνες συσχέτισης που προκύπτουν από τα συχνά σύνολα που έχει βρει. Για την εύρεση συχνών συνόλων αρχικά ο αλγόριθμος βρίσκει συχνά αντικείμενα

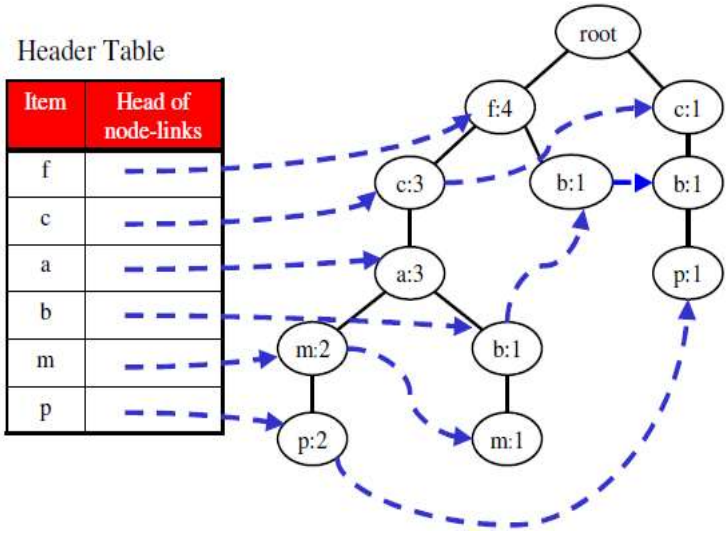
με μήκος 1, στη συνέχεια δημιουργεί επαναληπτικά υποψήφια σύνολα μήκους $k+1$ από τα συχνά σύνολα μήκους k. Σαρώνοντας μια φορά την συλλογή εγγράφων βρίσκει τα συχνά σύνολα από τα υποψήφια υποσύνολα μήκους $k+1$. Για την δημιουργία κανόνων συσχέτισης ελέγχει την εμπιστοσύνη όλων των πιθανών κανόνων που προκύπτουν από τα συχνά σύνολα του πρώτου βήματος και κρατάει μόνο τους κανόνες των οποίων η εμπιστοσύνη ξεπερνά το κατώφλι της εμπιστοσύνης. Ο αλγόριθμος τερματίζει όταν δεν μπορεί να βρει άλλα υποψήφια σύνολα .

3.4.2. Ο αλγόριθμος FP-Growth

Ο αλγόριθμος FP-Growth (Frequent Pattern Growth) είναι ένας εναλλακτικός αλγόριθμος εξόρυξης κανόνων συσχέτισης, συνήθως γρηγορότερος από τον Apriori. Ο αλγόριθμος χρησιμοποιεί ένα προθεματικό δένδρο αναπαράστασης της συλλογής που λέγεται FP-δένδρο και έτσι καταφέρνει να ανακτά γρηγορότερα τις εγγραφές των συνόλων. Για την δημιουργία του FP-δένδρου, αφαιρεί από το σύνολο των εγγράφων όλα τα αντικείμενα που δεν είναι συχνά και αναδιατάσσει τα υπόλοιπα σε φθίνουσα διάταξη ως προς την τιμή της υποστήριξής τους. Στη συνέχεια εισάγει τις αναδιαταγμένες εγγραφές στο προθεματικό δένδρο, όπου κοινά προθέματα αναπαρίστανται με το ίδιο μονοπάτι. Τέλος, για την διευκόλυνση της αναζήτησης δημιουργεί συνδεδεμένες λίστες για κάθε αντικείμενο του δένδρου. Παράδειγμα ενός FP-δένδρου φαίνεται στην παρακάτω εικόνα που έχει δημιουργηθεί βάση του πίνακα.

Items	Frequent Items
f, a, c, d, g, i, m, p	*f, c, a, m, p*
a, b, c, f, l, m, o	*f, c, a, b, m*
b, f, h, j, o	*f, b*
b, c, k, s, p	*c, b, p*
a, f, c, e, l , p, m, n	*f, c, a, m, p*

Ταξινομημένες εγγραφές για το FP-δένδρο



Εικόνα 3. Το FP-δένδρο αφού έχουν εισαχθεί όλες οι εγγραφές

Μόλις δημιουργηθεί το FP-δένδρο ο αλγόριθμος λειτουργεί αναδρομικά χρησιμοποιώντας το δένδρο για να βρει τα συχνά σύνολα. Αρχικά εξετάζει ξεχωριστά κάθε αντικείμενο κατά αύξουσα σειρά υποστήριξης και από την αντίστοιχη συνδεδεμένη λίστα βρίσκει τους κόμβους όπου εμφανίζεται το αντικείμενο αυτό, για να ακολουθήσει το μονοπάτι προς αυτούς τους κόμβους. Για κάθε μονοπάτι δίνει μια υποστήριξη ίση με τον αντίστοιχο αριθμό που έχει ο κόμβος στο FP-δένδρο. Στη συνέχεια συνδυάζει το υπό εξέταση αντικείμενο με τα μονοπάτια του για να βρει τα συχνά σύνολα.

3.4.3. Ομαδοποίηση (Clustering)

Η τεχνική της ομαδοποίησης προσπαθεί να εντοπίσει κάποια δομή στα δεδομένα που επεξεργάζεται και να δημιουργήσει ομάδες (clusters) αντικειμένων ή μια ιεραρχία ομάδων. Η τεχνική αυτή δημιουργεί ομάδες συνήθως με ένα μέτρο απόστασης των αντικειμένων με στόχο αντικείμενα της ίδιας ομάδας να είναι όσο το δυνατόν πιο κοντά μεταξύ τους και αντικείμενα διαφορετικής ομάδας να απέχουν όσο το δυνατόν περισσότερο. Το τι αποτελεί μια ομάδα, μπορεί να διαφέρει λόγω των δεδομένων. Για παράδειγμα, ομάδα

μπορεί να θεωρηθεί και κάποιο σημείο με μεγαλύτερη πυκνότητα εμφάνισης των αντικειμένων σε σχέση με κάποιο άλλο. Έτσι οι διαφορετικές θεωρήσεις της ομάδας οδηγούν σε ομάδες με διαφορετικές ιδιότητες, όπως σχήμα, πυκνότητα ή μέγεθος. Βέβαια για κάθε μια θεώρηση ομάδας υπάρχει και ο αντίστοιχος αλγόριθμος.

3.4.4. Ο αλγόριθμος k-Means

Ο αλγόριθμος k-Means είναι από τους πιο γνωστούς αλγόριθμους τμηματοποίησης που ομαδοποιούν σημειακά αντικείμενα δημιουργώντας k ομάδες. Ο αριθμός των ομάδων δίνεται ως παράμετρος στον αλγόριθμο. Αρχικά επιλέγονται k τυχαία κέντρα, χωρίς αυτό να σημαίνει πως τα κέντρα αντιστοιχούν σε κάποια αντικείμενα. Στη συνέχεια για κάθε αντικείμενο επιλέγεται μια ομάδα βάση του κέντρου στο οποίο είναι πιο κοντά. Αναθέτοντας όλα τα αντικείμενα σε κάποια ομάδα υπολογίζεται εκ νέου το κέντρο της ομάδας έτσι ώστε να είναι στη μέση όλων των αντικειμένων της ομάδας. Αυτή η διαδικασία επαναλαμβάνεται μέχρις ότου δεν ύπαρξη καμία μεταβολή των κέντρων. Σε τέτοια περίπτωση ο αλγόριθμος έχει συγκλίνει και τερματίζει. Στόχος του αλγορίθμου είναι να βελτιστοποιήσει όσο γίνεται το συνολικό τετραγωνικό λάθος. Το τετραγωνικό λάθος για μια ομάδα k ορίζεται ως:

$$\boldsymbol{e_k^2 = \sum_{i=1}^{n_k} \sum_{j=1}^{d} (p_{i_j}^k - m_j^k)^2}$$

Όπου n_k είναι το πλήθος των σημείων της ομάδας k, όπου p_{i_j}είναι η συντεταγμένη του σημείου i (για 1 ≤ i ≤ n_k) στη διάσταση j (για 1 ≤ j ≤ d), όπου m^k το κέντρο της ομάδας k. Και το συνολικό λάθος για όλες τις ομάδες ορίζεται ως: $E^2 = \sum_{i=1}^{K} e_k^2$.

1. Ο αλγόριθμος DBSCAN

Ο αλγόριθμος DBSCAN (Density Based Spatial Clustering of Applications with Noise) είναι ένας αλγόριθμος που δημιουργεί ομάδες βάσει της πυκνότητας των σημείων. Δέχεται σαν είσοδο δυο τιμές, μια τιμή Eps που είναι η ακτίνα με την

οποία θα προσδιορίζει τις ομάδες και μια τιμή MinPts που είναι ο ελάχιστη τιμή πυκνότητας για μια ομάδα. Ο αλγόριθμος είναι κατάλληλος για δεδομένα που δημιουργούν ομάδες με πυκνά σημεία οι οποίες είναι εύκολα διαχωρίσιμες από άλλες με χαμηλότερη πυκνότητα που ίσως αποτελούν θόρυβο. Πρέπει να σημειωθεί πως αλγόριθμος προϋποθέτει ότι οι ομάδες έχουν παρόμοια πυκνότητα μεταξύ τους. Στα πλεονεκτήματα που αλγορίθμου συμπεριλαμβάνεται το γεγονός ότι δεν επηρεάζεται από τον θόρυβο και μπορεί να επεξεργαστεί δεδομένα όπου τα σημεία τους δημιουργούν ομάδες με διαφορετικά σχήματα και μεγέθη. Επίσης ένα σημαντικό πλεονέκτημα είναι ότι δεν χρειάζεται να είναι γνωστός ο αριθμός των ομάδων όπως για παράδειγμα στο k-Means. Από την άλλη πλευρά τα μειονεκτήματά του περιλαμβάνουν την ευαισθησία που έχει στις παραμέτρους που του δίνονται δηλαδή στις τιμές Eps και MinPts.

Κεφάλαιο 4

ΕΝΙΣΧΥΤΙΚΗ ΜΑΘΗΣΗ, ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ ΚΑΙ ΓΕΝΕΤΙΚΟΙ ΑΛΓΟΡΙΘΜΟΙ

Ως στόχος της παρούσας διπλωματικής εργασίας είναι η κατασκευή ενός μοντέλου πρόβλεψης των μετοχικών τιμών. Μια διαδικασία η οποία πλαισιώθηκε με τον τρόπο εκπαίδευσης της Ενισχυτικής Μάθησης σε συνδυασμό με Γενετικούς Αλγορίθμους και Νευρωνικά Δίκτυα.

4.1. Ενισχυτική Μάθηση (Reinforcement Learning)

Η ενισχυτική μάθηση (Reinforcement Learning - RL) είναι ένας τομέας μηχανικής μάθησης που ασχολείται με το πώς οι ευφυείς πράκτορες πρέπει να αναλαμβάνουν ενέργειες σε ένα περιβάλλον προκειμένου να μεγιστοποιήσουν την έννοια της αθροιστικής ανταμοιβής. Η ενισχυτική μάθηση αποτελεί ένα κλάδο της μηχανικής μάθησης, παράλληλα με την εποπτευόμενη μάθηση και την μάθηση χωρίς επίβλεψη [13].

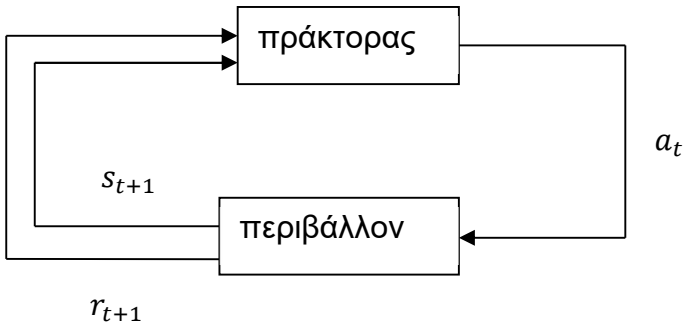
Η ενισχυτική μάθηση ουσιαστικά αποτελεί σειρά από κατάλληλα μέτρα για τη μεγιστοποίηση της ανταμοιβής σε μια συγκεκριμένη κατάσταση. Χρησιμοποιείται από διάφορα λογισμικά και μηχανήματα για να βρει την καλύτερη δυνατή συμπεριφορά ή διαδρομή που πρέπει να ακολουθήσει σε μια συγκεκριμένη κατάσταση. Η ενισχυτική μάθηση διαφέρει από την εποπτευόμενη μάθηση με τρόπο που στην εποπτευόμενη μάθηση τα δεδομένα εκπαίδευσης έχουν το κλειδί απάντησης, έτσι το μοντέλο εκπαιδεύεται με τη σωστή απάντηση, ενώ στην ενισχυτική μάθηση, δεν υπάρχει απάντηση, αλλά ο ενισχυτικός παράγοντας αποφασίζει τι να κάνει εκτελέσει τη δεδομένη εργασία. Ελλείψει συνόλου δεδομένων εκπαίδευσης, είναι βέβαιο ότι θα διδαχθεί από την εμπειρία του [13].

Η ενισχυτική μάθηση διαφέρει από την εποπτευόμενη μάθηση στο ότι δεν χρειάζεται να παρουσιαστούν επισημασμένα ζεύγη εισόδου/εξόδου και στο ότι δεν χρειάζεται να διορθωθούν ρητά οι μη βέλτιστες ενέργειες. Αντίθετα, η εστίαση είναι στην εξεύρεση ισορροπίας μεταξύ της εξερεύνησης (αχαρτογράφητης περιοχής) και της εκμετάλλευσης (της τρέχουσας γνώσης). Οι μερικώς εποπτευόμενοι αλγόριθμοι RL μπορούν να συνδυάσουν τα πλεονεκτήματα των εποπτευόμενων και των αλγορίθμων RL [14].

Η διαδικασία της μάθησης διαφέρει κατά πολύ από άλλες τεχνικές μάθησης που είχαμε συναντήσει. Στην ενισχυτική μάθηση δίνεται έμφαση στο πως θα μπορέσουμε να καταφέρουμε να εξισορροπήσουμε την εξερεύνηση (κάποιας άγνωστης περιοχής) του πράκτορα σε συνδυασμό με την εκμετάλλευση της ήδη υπάρχουσας γνώσης του.

Το βασικό μοντέλο ενισχυτικής μάθησης που εφαρμόζεται στις μαρκοβιανές διαδικασίες απόφασης αποτελείται από ένα σύνολο καταστάσεων του περιβάλλοντος S, από ένα σύνολο ενεργειών A που μπορούν να γίνουν στο περιβάλλον αυτό και από ένα σύνολο ανταμοιβών που R.

Σε κάθε διακριτή χρονική στιγμή t ο πράκτορας λαμβάνει ένα σήμα από το περιβάλλον του το οποίο το αντιλαμβάνεται σαν μια κατάσταση $s_t \in S$. Βάσει της κατάστασης s_t ο πράκτορας επιλέγει μια ενέργεια a_t από το σύνολο διαθέσιμων ενεργειών της κατάστασης, $a_t \in A(s_t)$. Στη συνέχεια λαμβάνει από το περιβάλλον μια νέα κατάσταση s_{t+1} και μια ανταμοιβή r_{t+1} , βασιζόμενα στην πράξη a_t . Σχηματικά η διαδικασία μπορεί να περιγραφεί ως εξής [13]:



Μετά από αυτές τις αλληλεπιδράσεις στόχος του πράκτορα είναι να αναπτύξει μια πολιτική $\pi: S \to A$ που μεγιστοποιεί την συνολική ανταμοιβή του $R = r_0 + r_1 + r_2 + \cdots + r_n$.

Η Ενισχυτική Μάθηση αντιμετωπίζει τα προβλήματα λήψης αποφάσεων, βελτιστοποιώντας τις μελλοντικές ανταμοιβές και αξιολογώντας τις μακροπρόθεσμες συνέπειες των ενεργειών τους, έχοντας τη δυνατότητα να θυσιάσει την άμεση ανταμοιβή για να κερδίσει περισσότερο σε βάθος χρόνου. Αυτή η ιδιότητα του RL, το καθιστά ελκυστικό για οικονομικές εφαρμογές, λόγω του γεγονότος ότι ο ορίζοντας μιας επένδυσης κυμαίνεται από ημέρες, χρόνια ή δεκαετίες [15].

Οι μακροπρόθεσμες ανταμοιβές είναι απαραίτητες για τις χρηματοοικονομικές επενδύσεις και οι «μυωπικοί» πράκτορες μπορούν να έχουν πολύ κακή απόδοση στη παραπάνω περίπτωση. Κατά συνέπεια η επιλογή και ο σχεδιασμός του σωστού σήματος ανταμοιβής είναι ζωτικής σημασίας για κάθε εφαρμογή. Ειδικά στα χρηματοοικονομικά και στις συναλλαγές, οι χειροκίνητη διαμόρφωση της ανταμοιβής είναι πολύ χρήσιμη[20].

4.2. Δυναμικά Συστήματα

Η ενισχυτική μάθηση είναι κατάλληλη για τον βέλτιστο έλεγχο δυναμικών συστημάτων. Ένα δυναμικό σύστημα προσαρμοζόμενου ελέγχου διαθέτει έναν προσαρμοζόμενο ελεγκτή (πράκτορας). Αυτός ο ελεγκτής αλληλοεπιδρά με το περιβάλλον λαμβάνοντας την ελεγχόμενη κατάσταση και εκτελώντας μια ενέργεια. Το περιβάλλον, το οποίο είναι μέρος ενός συστήματος, δίνει κάποιου είδους απόκριση για κάθε ενέργεια (ανταμοιβή). Τα κριτήρια απόδοσης εισάγονται για έναν προσαρμοζόμενο ελεγκτή. Αυτός ο τυπικός προσαρμοζόμενος ελεγκτής πρέπει να προσαρμόσει τον μηχανισμό ελέγχου με βάση την απόδοση και τα αποτελέσματα (πολιτική) και να επαναλάβει τον κύκλο. Η παρούσα μελέτη αφορά τα δυναμικά συστήματα διακριτού χρόνου, αν και οι περισσότερες από τις έννοιες που αναπτύσσονται επεκτείνονται σε συστήματα συνεχούς χρόνου. [16]:

Ο όρος **πράκτορας** αναφέρεται στον **ελεγκτή**, ενώ το **περιβάλλον** χρησιμοποιείται ως εναλλακτική του όρου **σύστημα**. Ο στόχος ενός αλγορίθμου ενισχυτικής μάθησης είναι η ανάπτυξη εκπαίδευση) ενός πράκτορα ικανού να αλληλοεπιδρά επιτυχώς με το περιβάλλον, έτσι ώστε να μεγιστοποιεί κάποιο βαθμωτό μέγεθος (στόχο) με την πάροδο του χρόνου.

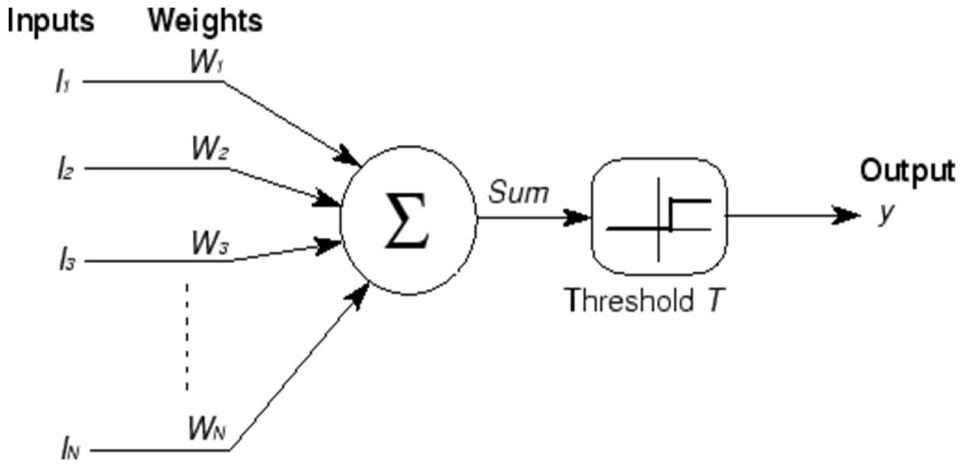
Η Ενέργεια (Action) είναι το σήμα ελέγχου που στέλνει ο πράκτορας στο σύστημα σε κάθε βήμα t. Αποτελεί τον μοναδικό τρόπο με τον οποίο ο πράκτορας μπορεί να επηρεάσει την κατάσταση του περιβάλλοντος και ως αποτέλεσμα, να οδηγήσει σε διαφορετικά σήματα ανταμοιβής.

Το σύνολο ενεργειών (action space), αναφέρεται στο σύνολο των ενεργειών που ένας πράκτορας επιτρέπεται να πραγματοποιήσει.

Η Ανταμοιβή (Reward) είναι ένα βαθμωτό σήμα απόκρισης που προσδιορίζει την απόδοση του πράκτορα σε κάθε διακριτό βήμα t. Στόχος του πράκτορα είναι να μεγιστοποιήσει την συνολική ανταμοιβή που μαζεύει σε μια ακολουθία βημάτων [19].

4.3. Νευρωνικά Δίκτυα (Neural Networks)

Η τεχνική μάθησης με νευρωνικά δίκτυα προσπαθεί να προσομοιώσει τον τρόπο μάθησης του ανθρώπινου εγκεφάλου. Ο ανθρώπινος εγκέφαλος έχει τον νευρώνα ως βασικό δομικό στοιχείο και αποτελείται από τον άξονα, τους δενδρίτες και τις συνάψεις. Ο άξονας είναι η πύλη εξόδου του νευρώνα και στέλνει σήματα προς άλλους νευρώνες, οι δενδρίτες είναι οι πύλες εισόδου του νευρώνα και οι συνάψεις είναι τα σημεία ένωσης μεταξύ του νευρώνα με δενδρίτες άλλων νευρώνων. Η μελέτη των νευρώνων και η προσπάθεια μαθηματικής μοντελοποίησης τους οδήγησαν στα τεχνικά νευρωνικά δίκτυα (Artificial Neural Network). Το μοντέλο McCulloch-Pitts είναι ένα απλό μοντέλο για την περιγραφή του νευρώνα [13].



Εικόνα 4. Μοντέλο McCulloch-Pitts

Τα $I_1, I_2, I_3 \ldots I_N$ είναι οι είσοδοι του νευρώνα και τα $w_1, w_2, w_3 \ldots w_N$ είναι τα συναπτικά βάρη. Αν το άθροισμα είναι μεγαλύτερο από το κατώφλι T τότε ο νευρώνας πυροβολεί διαφορετικά παραμένει αδρανής. Μαθηματικά μπορούμε να αναπαραστήσουμε το μοντέλο με $\boldsymbol{y = f(s - t)}$ όπου y είναι η έξοδος του νευρώνα, s το άθροισμα των εισόδων πολλαπλασιασμένα με τα βάρη τους, t το κατώφλι και f() η βηματική συνάρτησης ενεργοποίησης. Υπάρχουν και άλλες μοντελοποιήσεις νευρώνων με σημαντικότερη διαφορά την συνάρτηση ενεργοποίησης του νευρώνα. Η πιο διαδεδομένη συνάρτηση ενεργοποίησης είναι η σιγμοειδής

$\boldsymbol{f(u) = 1/(1 + e^{-u})}$.

Ένα νευρωνικό δίκτυο εκπαιδεύεται με ένα σύνολο από παραδείγματα που έχουν την είσοδο του νευρωνικού δικτύου αλλά και την έξοδο. Η έννοια της μάθησης και της εκπαίδευσης κρύβεται πίσω από τα βάρη του κάθε νευρώνα. Μετά από κάθε κύκλο εκπαίδευσης (εποχή) τα συναπτικά βάρη μεταβάλλονται αναλόγως του σφάλματος που προέκυψε. Σφάλμα ορίζεται ως η διαφορά μεταξύ του στόχου του νευρωνικού δικτύου και της εξόδου του.

Τα νευρωνικά δίκτυα χρησιμοποιούνται κυρίως για ταξινόμηση και παρεμβολή και βρίσκουν μεγάλη εφαρμογή σε πραγματικά δεδομένα όπου υπάρχει θόρυβος. Είναι πολύ γρήγορα στην εφαρμογή της γνώσης που έχουν αποκτήσει και μπορούν να χρησιμοποιηθούν σε συστήματα πραγματικού

χρόνου. Μειονέκτημα τους είναι ότι απαιτούν πολύ χρόνο στην εκπαίδευση και ότι δεν μπορεί να ερμηνευτεί η γνώση που αποκτούν από τον άνθρωπο.

4.4. Γενετικοί Αλγόριθμοι

Οι γενετικοί αλγόριθμοι είναι εμπνευσμένοι από την βιολογία και βασίζονται στην εξέλιξη μέσω της γενετικής μετάλλαξης, φυσικής επιλογής και διασταύρωσης. Ουσιαστικά αποτελούν μια μέθοδο αναζήτησης βέλτιστης λύσης και για αυτό χρησιμοποιούνται σε προβλήματα βελτιστοποίησης όπου ο πληθυσμός εξελίσσεται προς το καλύτερο [18] .

Η λειτουργία των γενετικών αλγορίθμων θα μπορούσαμε να πούμε ότι είναι σχετικά απλή. Ξεκινώντας ο αλγόριθμος δημιουργεί τυχαία αντίγραφα του γενετικού κώδικα που λέγονται χρωμοσώματα και αναπαριστούνται με ακολουθίες από bit. Στη συνέχεια αξιολογεί το κάθε χρωμόσωμα με μια συνάρτηση καταλληλόλητας και εφαρμόζει διαδικασίες αναπαραγωγής όπως η επιλεκτική αναπαραγωγή (selective reproduction), διασταύρωση (crossover) και τυχαία μετάλλαξη (random mutation). Ο νέος πληθυσμός που προκύπτει είναι η νέα γενιά και αποτελεί τον γενετικό κώδικα πάνω στον οποίο θα γίνει η νέα επανάληψη. Η διαδικασία της επανάληψης (αναπαραγωγή και εξέλιξη) μπορεί να τερματίσει με διάφορους τρόπους όπως μετά από ένα μέγιστο αριθμό επαναλήψεων ή όταν πάψει η αύξηση της καταλληλότητας ή ακόμα όταν βρεθεί κάποια ικανοποιητική καταλληλότητα [18].

Η διαδικασία της φυσικής επιλογής ξεκινά με την επιλογή των πιο ικανών ατόμων από έναν πληθυσμό. Παράγουν απογόνους που κληρονομούν τα χαρακτηριστικά των γονέων τα οποία θα προστεθούν στην επόμενη γενιά. Εάν οι γονείς έχουν καλύτερη φυσική κατάσταση, οι απόγονοί τους θα είναι καλύτεροι από τους γονείς και θα έχουν περισσότερες πιθανότητες να επιβιώσουν. Αυτή η διαδικασία συνεχίζει να επαναλαμβάνεται και στο τέλος, θα βρεθεί μια γενιά με τα καλύτερα άτομα.

Αυτή η έννοια μπορεί να εφαρμοστεί και στην τεχνητή επιλογή. Έτσι για ένα πρόβλημα σε ένα σύνολο λύσεων επιλέγουμε το σύνολο των καλύτερων από αυτές και παράγουμε τις επόμενες γενεές.

Σε έναν γενετικό αλγόριθμο λαμβάνονται υπόψη πέντε φάσεις.

- Αρχικός πληθυσμός
- Ταξινόμηση με βάση μια συνάρτηση αξιολόγησης
- Επιλογή
- Διασταύρωση
- Μετάλλαξη

Η διαδικασία ξεκινά με ένα σύνολο ατόμων που ονομάζεται Πληθυσμός [12]. Κάθε άτομο είναι μια λύση στο πρόβλημα .

Ένα άτομο χαρακτηρίζεται από ένα σύνολο παραμέτρων (μεταβλητών) τα οποία ονομάζονται γονίδια.

Σε έναν γενετικό αλγόριθμο, το σύνολο των τιμών των γονιδίων ενός ατόμου αναπαρίσταται με ένα πίνακα. Συνήθως, χρησιμοποιούνται δυαδικές τιμές (συμβολοσειρά 1 και 0). Με βάση την κωδικοποίηση των τιμών των γονιδίων κωδικοποιούμε και παράγουμε το λεγόμενο χρωμόσωμα.

Λειτουργία αξιολόγησης (Fitness)

Η συνάρτηση fitness καθορίζει πόσο κατάλληλο είναι ένα άτομο (την ικανότητα ενός ατόμου να ανταγωνίζεται άλλα άτομα). Ουσιαστικά με βάση μια συνάρτηση δίνεται βαθμολογία σε κάθε άτομο. Η πιθανότητα να επιλεγεί ένα άτομο για αναπαραγωγή βασίζεται στη βαθμολογία του.

Επιλογή

Η ιδέα της φάσης επιλογής είναι η διαδικασία επιλογής των πιο κατάλληλων άτομα με βάση των οποίων θα κληρονομήσουν τα γονίδιά τους στην επόμενη γενιά.

Δύο ζευγάρια ατόμων (γονείς) επιλέγονται με βάση την βαθμολογία της κατάστασης τους. Τα άτομα με υψηλή φυσική κατάσταση έχουν περισσότερες πιθανότητες να επιλεγούν για αναπαραγωγή.

Crossover

Η διασταύρωση είναι η πιο σημαντική φάση σε έναν γενετικό αλγόριθμο. Για κάθε ζευγάρι γονέων που πρόκειται επιλέγεται για αναπαραγωγή, ένα σημείο διασταύρωσης επιλέγεται τυχαία μέσα από τα γονίδια και με βάση μια διαδικασία παράγονται οι απόγονοι. Πιο συγκεκριμένα σε ένα γονίδιο μπορεί να επιλεχθεί το μέγιστο, το ελάχιστο ο μέσος όρος των τιμών ή μια άλλη συνάρτηση που παίρνει σαν είσοδο τις δύο τιμές των γονιδίων των γονέων και παράγει ένα παιδί.

Μετάλλαξη

Σε ορισμένους νέους απογόνους, ορισμένα από τα γονίδιά τους μπορούν να υποβληθούν σε μετάλλαξη με μια πιθανότητα. Αυτό σημαίνει ότι μερικά από τα γονίδια θα αλλάξουν με συνήθως τυχαία διαδικασία παράγοντας ένα μεταλλαγμένο απόγονο.

Τερματισμός - Αποτέλεσμα

Ο αλγόριθμος τερματίζεται εάν ο πληθυσμός συγκλίνει (δεν παράγει απογόνους που διαφέρουν σημαντικά από την προηγούμενη γενιά). Τότε λέγεται ότι ο γενετικός αλγόριθμος έχει δώσει ένα σύνολο λύσεων στο πρόβλημά μας.

Διαφορετικά ο αλγόριθμος τερματίζει μετά από Ν γενεές.

Κεφάλαιο 5

Εφαρμογή σε χρονοσειρές τραπεζικών μετοχών

5.1. Σύστημα πρόβλεψης τραπεζικών μετοχών

Στόχος της εφαρμογής είναι να δημιουργηθεί ένα μοντέλο όπου μέσα από χρονοσειρά μιας μετοχής μοντέλο να μπορεί να αναγνωρίσει κάποια Petterns με βάση τα οποία τελικά να μπορεί να κάνει πρόβλεψη, με όσο το δυνατόν μεγαλύτερη ακρίβεια, της επόμενης τιμής της μετοχής. Οι μετοχές που επιλέχθησαν για να τις δοκιμές του αλγόριθμου ήταν μετοχές του τραπεζικού κλάδου και συγκεκριμένα της Εθνικής Τράπεζας, της Πειραιώς και της ALPHA.

Τα δεδομένα δίνονται από διάφορες πηγές αφού το χρηματιστήριο δίνει τις χρονοσειρές ελεύθερα για το άνοιγμα , το κλείσιμο, το ημερήσιο ελάχιστο, το ημερήσιο μέγιστο, τον όγκο συναλλαγών , αριθμό συναλλαγών . Στην περίπτωση μας επιλέξαμε την οικονομική δημοσιογραφική ιστοσελίδα naftemporiki.gr όπου προσφέρει με αρκετά οργανωμένο τρόπο δεδομένα χρονοσειρών.

Τα αρχεία που χρησιμοποιήσαμε ήταν σε μόρφή csv και περιλάμβανουν τις παρακάτω στήλες

- Trade Date
- High,
- Low,
- Open,
- Close,
- Volume,
- Prev. Close,
- Total Turnover,
- Num. Of Trans.

Σε όλα τα αρχεία το ιστορικό ήταν 1243 μέρες.

Στην περίπτωση μας χρησιμοποιήσαμε την στήλη close δηλαδή την τιμή κλεισίματος

Η ανάγκη της πρόβλεψης είναι σημαντική για κάθε κλάδο της οικονομίας και ακόμα περισσότερο για το χρηματιστήριο αφού η πιθανή ακριβή πρόβλεψη μπορεί να δώσει μεγάλη κερδοφορία.

Πιο συγκεκριμένα μπορεί να βοηθήσει εταιρίες επενδύσεων να λειτουργήσουν με ουσιαστικό και στοχοποιημένο τρόπο στις επενδύσεις τους, αφού μια ακριβή πρόβλεψη μπορεί να τους οδηγήσει σε πολύ αποτελεσματικές αγορές ή πωλήσεις και τελικά να επιτευχθεί το μέγιστο κέρδος.

Για να γίνει αυτό πρέπει να εφαρμοστούν μοντέλα όπως τα νευρωνικά δίκτυα με τον πλέον κατάλληλο τρόπο ορίζοντας τις βέλτιστες παραμέτρους ή αλγόριθμοι αναγνώρισης προτύπων που ανατροφοδώντας συνεχώς με τις νέες τιμές υπολογίζουν με μεγαλύτερη ακρίβεια την επόμενη τιμή

Π.χ. αν ορίσουμε ένα νευρωτικό δίκτυο τριών (3) επιπέδων τότε έχουμε να δούμε παραμέτρους όπως τους νευρώνες σε κάθε επίπεδο, τις παραμέτρους learning rate και ορμή (momentum) κ.α.

Για να βρούμε τις καλύτερες παραμέτρους πρέπει να κάνουμε πολλές δοκιμές που ουσιαστικά είναι άπειροι συνδυασμοί. Ένας τρόπος για να ξεπεράσουμε ένα τέτοιο πρόβλημα είναι η εφαρμογή γενετικών αλγορίθμων.

Θεωρώντας σαν χρωμοσώματα τις παραμέτρους που πειράζουμε δηλαδή τον αριθμό νευρώνων σε κάθε επίπεδο, το learning rate και το Momentum, τότε μπορούμε με χρήση ενός τυχαίου αριθμού και εφαρμογή ενός γενετικού αλγορίθμου Ν γενεών να έχουμε το νευρωνικό που με συνεχείς μεταλλάξεις και διασταυρώσεις του πληθυσμού θα πετύχει την καλύτερη επίδοση.

5.2. Αλγόριθμοι που χρησιμοποιούνται

Όπως αναφέραμε στις χρονοσειρές που επιθυμούμε ο στόχος είναι η πρόβλεψη της επόμενης μέρας. Για αν επιτύχουμε κάτι τέτοιο ορίζουμε τα παρακάτω:

Κάθε χρονοσειρά έχει μια ιστορία μέσα από την οποία θεωρούμε ότι δημιουργεί ένα δυναμικό σύστημα. Το σύστημα αυτό καθορίζεται από μια σειρά από παράγοντες που τελικά θα παράγουν τις επόμενες τιμές.

Η πρόβλεψη χρονοσειρών είναι ένα ακόμη γενικά μια διαδικασία που αντιμετωπίσουν οι Επιστήμονες Δεδομένων καθημερινά. Ανάλογα την περίπτωση έχουμε μεγαλύτερη επιτυχία ή όχι. Έτσι αν πραγματικά μια μετοχή έχει μια εξάρτηση με τα παρελθοντικά δεδομένα η πρόβλεψη είναι πιο επιτυχής. Αν όχι όπως στην περίπτωση πολύ υψηλού θορύβου τότε η χρονοσειρά συμπεριφέρεται σαν ένα τυχαίο σύστημα οπότε η πρόβλεψη μοιάζει με αδύνατη αφού η επόμενη τιμή θεωρείται σχεδόν τυχαία.

Υπάρχουν πολλοί τρόποι προσέγγισης του προβλήματος πρόβλεψης.

Συνήθης τακτικές είναι η περιγραφική στατιστική, η παλινδρομήση, τα νευρωνικά δίκτυα κ.α.

5.3. Αλγόριθμος με Νευρωνικά Δίκτυα και Γενετικό Αλγόρυθμο

Η περίπτωση πρόβλεψης με νευρωνικά δίκτυα είναι μια συνηθισμένη τακτική όπου χρησιμοποιούνται παρελθοντικά δεδομένα σαν είσοδος στο νευρωνικό και έξοδος του συστήματος είναι η επόμενη ή οι επόμενες Κ τιμές της σειράς. [16]

Στη περίπτωση που εξετάζουμε αναπτύσσουμε μια ακόμα πιο δυναμική περίπτωση αφού η χρήση μόνο ενός νευρωνικού δεν μας δίνει το πιο αξιόπιστο αποτέλεσματα. Πιο συγκεκριμένα σε κάθε έρευνα γίνεται μελέτη για ποιο νευρωνικό είναι πιο κατάλληλο όπου δοκιμάζονται σειρά από παραμέτρους. Η χρήση γενετικών για τον υπολογισμό του βέλτιστου μοντέλου αποτελεί μια τεχνική που μπορεί να βοηθήσει ώστε να καταλήξουμε σε ένα πιθανό βέλτιστο μοντέλο σε ένα πεπερασμένο χρόνο.

Οι παράμετροι σε κάθε νευρωνικό είναι αρκετές.

Πρώτη παράμετρος είναι πόσα σημεία θα χρησιμοποιηθούν σαν είσοδος στο νευρωνικό.

Δεύτερη παράμετρος είναι η τιμή της ορμής

Τρίτη παράμετρος είναι ο ρυθμός μάθησης

Τέταρτη παράμετρος είναι η δομή του νευρωνικού δηλαδή πόσα επίπεδα θα χρησιμοποιηθούν.

Πέμπτη παράμετρος που αφορά και πάλι την δομή , πόσοι κόμβοι ανά επίπεδο θα χρησιμοποιηθούν

Έτσι η εύρεση ενός νευρωνικού που δίνει την καλύτερη απόδοση φαίνεται πολύ δύσκολη αφού οι περιπτώσεις δοκιμών των παραμέτρων μοιάζουν με άπειρες.

Στην περίπτωση αυτή μια προσέγγιση είναι οι γενετικοί αλγόριθμοι με στόχο να βρεθεί το καλύτερο δυνατό νευρωνικό δίκτυο.

Με βάση τα παραπάνω θεωρούμε ένα νευρωνικό δίκτυο NN(tn,lr, p, levels,L[levels]) όπου:

tn: αφορά πόσα σημεία γυρίζει πίσω στον χρόνο

lr: ο ρυθμός μάθησης

p: η ορμή μάθησης

levels: ο αριθμός των επιπέδων

L: για κάθε επίπεδο ο αριθμός των κόμβων

Θεωρώντας κάθε ένα νευρωνικό σαν ένα ξεχωριστό άτομο για ένα γενετικό αλγόριθμο μπορούμε να παράγουμε ένα πληθυσμό χρησιμοποιώντας σαν γονίδια τις τιμές των παραπάνω παραμέτρων.

Έτσι δημιουργούμε μια γενιά με M άτομα NN(tn,lr, p, levels,L[levels])

Οι τιμές των μεταβλητών tn, lr, p , levels, L[levels] αποτελούν τα χαρακτηριστικά - γονίδια κάθε ατόμου και παίρνουν τυχαίες τιμές.

Για κάθε άτομο υπολογίζουμε Fn (συνάρτηση αξιολόγησης) μελλοντικές τιμές μετά από εκπαίδευσης που κάνουμε για X άτομα και υπολογίζουμε το score κάθε ατόμου.

Το score αποτελεί το ποσοστό επιτυχίας του νευρωνικού που ουσιαστικά είναι το μέσο τετραγωνικό σφάλμα.

Για κάθε παιδί παίρνουμε το μέσο τετραγωνικό σφάλμα και κατατάσσουμε με βάση το score σε αύξουσα σειρά τα παιδιά κάθε γενιάς.

Από τα πρώτα 1/3 παιδιά του πληθυσμού παράγουμε K1 νέα άτομα-παιδιά με τις διαδικασίες της διασταύρωσης και K2 παιδιά με μετάλλαξη. Η διασταύρωση γίνεται με χρήση του μέσου όρου δύο τυχαίων χαρακτηριστικών, ενώ η μετάλλαξη γίνεται με χρήση μια νέας τιμής σε ένα τυχαίο χαρακτηριστικό-γονίδιο ενός παιδιού.

Η μετάλλαξη και διασταύρωση γίνεται με μια πιθανότητα p1, p2 αντίστοιχα.

Τα νέα παιδιά τοποθετούνται στο τέλος της λίστας των M παιδιών.

Η διαδικασία επαναλαμβάνεται για N φορές (όπου N<100). Όπου ως N ορίζεται το πλήθος των γενεών που καλούμε το γενετικό αλγόριθμο να παράξει. Στο τέλος της διαδικασίας έχει δημιουργηθεί μια λίστα Νευρωνικών Δικτύων (απογόνων επικρατέστερων) με κατά φθίνουσα κατάταξη αποδοτικότητας. Έτσι στην αρχή της λίστας αυτής έχουμε το καλύτερο νευρωνικό που προέκυψε.

Για κάθε χρονικό βήμα που αφορά την επόμενη τιμή γίνεται και πάλι η διαδικασία με στόχο τον υπολογισμό του νέου καλύτερου νευρωνικού μετά την ανατροφοδότηση.

Σε βήματα ο αλγόριθμος είναι ο παρακάτω:

Βήμα1. Παραγωγή πληθυσμού M ατόμων NN(tn,lr, p, levels,L[levels]) με τυχαίες τιμές των tn,lr,p,levels,L[levels]

Βήμα2. N=1 (αριθμός γενεάς)

Βήμα3. Εκπαίδευση κάθε NN(tn,lr, p, levels,L[levels]) με χρήση των παρελθοντικών τιμών (παράθυρο K τιμών) για την πρόβλεψη επόμενων τιμών.

Βήμα4. Υπολογισμό του Score κάθε ατόμου

Βήμα5. Ταξινόμηση με φθίνουσα σειρά με βάση το Score

Βήμα6. Επιλογή δύο τυχαίων ατόμων για διασταύρωση από το M/3 του πληθυσμού και παραγωγή ενός νέου ατόμου με διασταύρωση.

Βήμα7. Επιλογή ενός τυχαίου ατόμου για μετάλλαξη από τα M/3 του πληθυσμού και παραγωγή ενός νέου ατόμου με μετάλλαξη

Βήμα3. Εκπαίδευση των νέων ατόμων με χρήση των παρελθοντικών τιμών (παράθυρο K τιμών) και υπολογισμός του Score

Βήμα8. N=N+1

Βήμα9. Αν N<100 Μετάβαση στο βήμα 4

Βήμα10. Ταξινόμηση με φθίνουσα σειρά με βάση το Score

Βήμα11. Καλύτερο μοντέλο το πρώτο μοντέλο του πληθυσμού

Σαν αποτέλεσμα τελικά θα πάρουμε ένα μοντέλο νευρωνικού δικτύου που θα χρησιμοποιεί tn σημεία από την χρονοσειρά για υπολογισμό του επόμενου σημείου, lr ρυθμό μάθησης, p ορμή και θα έχει levels επίπεδα με κόμβους ανά επίπεδο L[i] με το πίνακα L να έχει levels τιμές. Για κάθε υπολογισμό επόμενου σημείου εφαρμόζεται και πάλι ο αλγόριθμος δίνοντας και πάλι ένα νέο βέλτιστο νευρωνικό.

5.4. Αλγόριθμος με εύρεση του καλύτερου Pattern

Οι χρονοσειρές μπορούν να εμφανίζουν περιπτώσεις που μοιάζουν πολύ στην συμπεριφορά τους σε σχέση με ένα χρονικό παράθυρο. Στην περίπτωση αυτή ο εντοπισμός ενός ίδιου τέτοιου παράθυρου μπορεί να αποτελέσει πάλι ζητούμενο.

Η χρήση εύρεσης Patterns ή όμοιων καμπύλων σε σειρές έχει εφαρμοστεί σε μελέτες δίνοντας γενικά θετικά αποτελέσματα [10][11].

Για να δούμε κατά πόσο ένα pattern ταιριάζει με το χρονικό σημείο στην σειρά μας ορίζουμε το παρακάτω:

Έστω A(t-n,t) ένα τμήμα της χρονοσειρά μας από το t-n έως την χρονική στιγμή t.

Ο αλγόριθμος εντοπίζει ένα αντίστοιχο ίδιο χρονικό τμήμα $B(t_2-n,t_2)$ χρησιμοποιώντας τον συντελεστή συσχέτισης. Αν η τιμή του συντελεστή συσχέτισης είναι 1 τότε έχουμε απόλυτη ταύτιση. Διαφορετικά αν η τιμή είναι κοντά στο 0 ή αρνητική τότε έχουμε μη ταύτιση.

Στην περίπτωση αυτή το ερώτημα είναι η τιμή του n. Δηλαδή πόσα σημεία πριν μπορούμε να εντοπίσουμε την καλύτερη ομοιότητα να υπάρχει.

Ο αλγόριθμος που εφαρμόζεται είναι ο παρακάτω:

Είσοδος αλγορίθμου : Χρονοσειρά A(t)

Μεταβλητές:

C[ ] πίνακας συντελεστών συσχέτισης

N[ ] πίνακας παραθύρου με σημείο μέγιστης συσχέτισης

Αρχικές τιμές: n=5, k=0

Βήμα 1. Παίρνουμε το διάνυσμα x=A(t-n,t) όπου t η τελευταία χρονική τιμή της χρονοσειράς.

Βήμα 2. Για κάθε i=0 μέχρι i=t-n κάνουμε συσχέτιση υπολογίζουμε το c(i) όπου c(i) ο συντελεστής συσχέτισης των A(t-n,t) με B(i,i+n).

Βήμα 3. Αποθηκεύω στο N[k,0]=n , N[k,1]=max(c) ,N[k,2]=i

Βήμα 4. k=k+1, n=n+1

Βήμα 5. Αν n<100 επανέλαβε το βήμα 1

Βήμα 6. Παίρνουμε το k με μέγιστο συντελεστή συσχέτιση στην δεύτερη στήλη του πίνακα N

Βήμα 7: Αν το N[k,1]>0.7 τότε υπολογίζουμε σαν επόμενη τιμή την τιμή A(N[k,2]+n+1)

Δηλαδή δημιουργούμε ένα αλγόριθμο όπου έχουμε μετατόπιση της σειρά από την μία χρονική σειρά μέχρι την αρχή της σειρά μας με αρχική τιμή στο n =5

Εντοπίζουμε όλους τους συντελεστές κατά pearson.

Στην συνέχεια αυξάνουμε το n κατά 1 και επαναλαμβάνουμε την διαδικασία.

Η διαδικασία επαναλαμβάνεται μέχρι το n=100.

Επιλέγουμε το pattern (t2,n) που έχει τον μεγαλύτερο συντελεστή συσχέτισης και δίνουμε σαν τιμή πρόβλεψης την επόμενη τιμή του pattern από την χρονική στιγμή t2.

Σε περίπτωση που έχουμε πολύ χαμηλό συντελεστή συσχέτισης τότε εφαρμόζουμε την διαδικασία που περιγράψαμε με τα νευρωνικά δίκτυα.

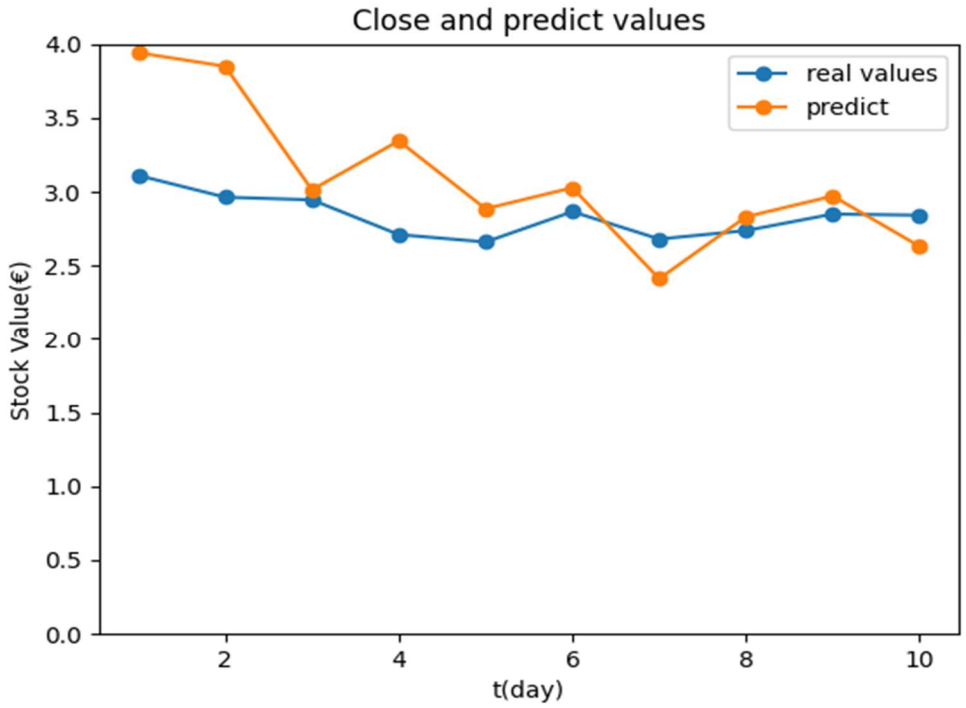
Σε κάθε βήμα εφαρμόζουμε την νέα τιμή της χρονοσειράς και παίρνουμε τα αποτελέσματα.

Ο κώδικας των αλγορίθμων φαίνεται στο παράρτημα Ι

5.5. Αποτελέσματα

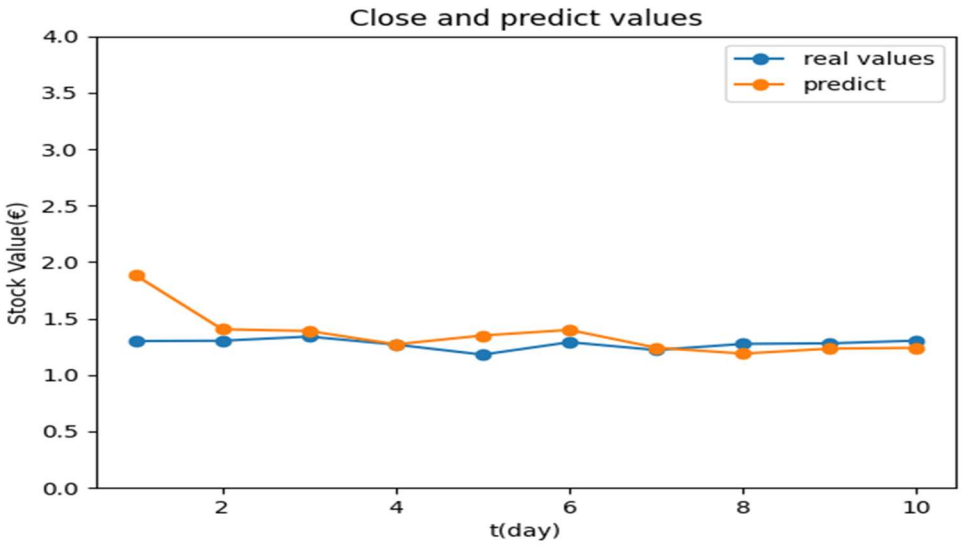
Εφαρμόζοντας τον αλγόριθμο που περιγράφεται στην ενότητα 5.2 , με χρήση νευρωνικών και γενετικών αλγορίθμων έχουμε διαγραμματικά τα παρακάτω αποτέλεσμα πρόβλεψης. Στα επακόλουθα διαγράμματα απεικονίζονται ανά τράπεζα η προβλεπόμενη τιμή για το τελευταίο δεκαήμερο των δεδομένων, σε σχέση με την πραγματική τιμή όπως αυτή έχει οριστεί.

Στην περίπτωση της Εθνικής Τράπεζας Ελλάδος έχουμε :



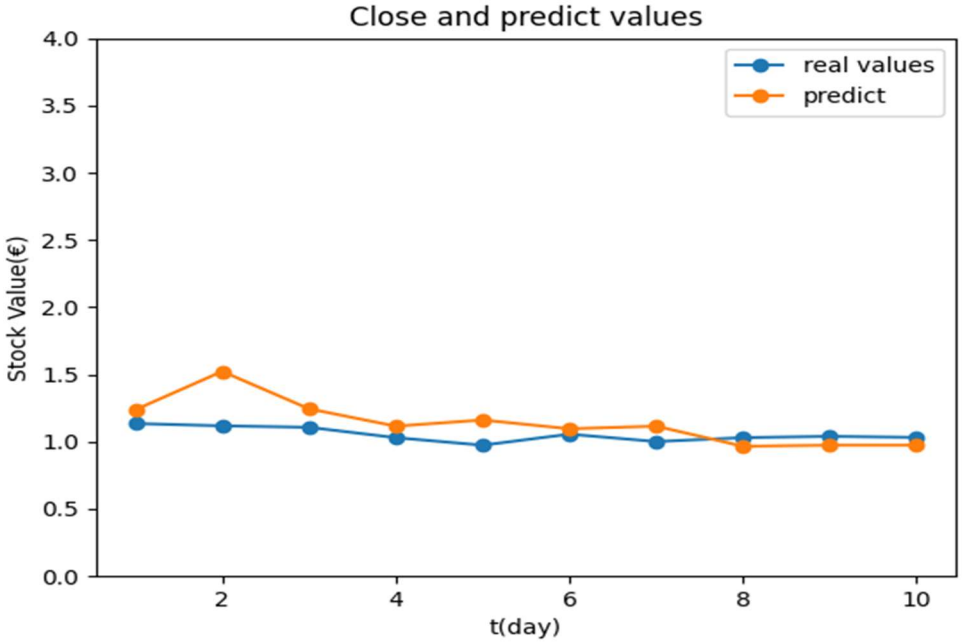
Εικόνα 5. Πρόβλεψη 10 επόμενων σημείων στην μετοχή της Εθνικής Τράπεζας

Αντίστοιχα στην περίπτωση της Πειραιώς έχουμε :



Εικόνα 6. Πρόβλεψη 10 επόμενων σημείων στην μετοχή της Τράπεζας ΠΕΙΡΑΙΩΣ

Αντίστοιχα στην περίπτωση της ALPHA έχουμε :



Εικόνα 7. . Πρόβλεψη 10 επόμενων σημείων στην μετοχή της Τράπεζας ALPHA

Από τις παραπάνω γραφικές παραστάσεις παρατηρούμε μια σχετική ακρίβεια των προβλέψεων αφού οι τιμές φαίνεται να μην αποκλίνουν από την πραγματική τιμή.

Για την ακρίβεια των αποτελεσμάτων παίρνουμε σαν μέτρα σύγκρισης των συντελεστή συσχέτισης και το μέσο σχετικό τετραγωνικό σφάλμα.

Ο συντελεστής συσχέτισης της πρόβλεψης από το μοντέλο μας για 50 επόμενα σημεία και το μέσο σχετικό τετραγωνικό σφάλμα φαίνεται στον παρακάτω πίνακα:

Μετοχή	Συντελεστής συσχέτισης	Μέσο τετραγωνικό Σφάλμα
Εθνικής	0.72	0.03
ALPHA	0.73	0.07
Πειραιώς	0.68	0.05

Το συμπέρασμα είναι ότι η πρόβλεψη είναι αρκετά καλή αφού παρουσιάζει μια ομοιότητα με συντελεστή συσχέτισης > 0.7

Με χρήση του δεύτερου αλγόριθμου που αναφέρεται στη ενότητα 5.3 και χρησιμοποιεί σύγκριση patterns έχουμε τα παρακάτω αποτελέσματα:

Ο συντελεστής συσχέτισης της πρόβλεψης από το μοντέλο μας για 50 επόμενα σημεία και το μέσο σχετικό τετραγωνικό σφάλμα είναι:

Μετοχή	Συντελεστής συσχέτισης	Μέσο τετραγωνικό Σφάλμα
Εθνικής	0.74	0.06
ALPHA	0.73	0.05
Πειραιώς	0.71	0.05

Είναι σημαντικό να αναφέρουμε ότι με την χρήση του γενετικού έχουμε ένα στοχαστικό σύστημα που μπορεί να δίνει διαφορετικά αποτελέσματα αφού η επιλογή του βέλτιστου νευρωνικού περιέχει λόγω του γενετικού τυχαιότητα.

Όπως διαπιστώνουμε από τα αποτελέσματα η χρήση των patterns φαίνεται να έχει μια σχετικά καλύτερη απόδοση αν και τα αποτελέσματα δεν διαφέρουν αισθητά. Βλέπουμε δηλαδή μια σχετική ομοιότητα μεταξύ των δύο μοντέλων αλλά με μικρή διαφορά μεγαλύτερου συντελεστή συσχέτισης και μικρότερου σφάλματος και στις τρεις χρονοσειρές στο αλγόριθμο που συγκρίνει patterns.

Επιπλέον ο χρόνος εκτέλεσης κάθε μεθόδου είναι ένας σημαντικός και αναμφισβήτητα συγκριτικός παράγοντας, ο οποίος εξετάσθηκε επίσης. Οι χρόνοι εκτέλεσης των αλγορίθμων είναι:

- Για τον αλγόριθμο του νευρωνικών με γενετικούς αλγορίθμους ο μέσος χρόνος εκτέλεσης για 10 σημεία πρόβλεψης ήταν στα **92sec**.

- Αντίστοιχα στο αλγόριθμο σύγκρισης-αναγνώρισης patterns ο μέσος χρόνος εκτέλεσης για 10 σημεία πρόβλεψης ήταν στα **62sec**.

Διαπιστώνουμε ότι ο αλγόριθμος με τα νευρωνικά δίκτυα και τους γενετικούς καθυστερεί περισσότερο σε σχέση με τον αλγόριθμο που κάνει χρήση αναζήτηση όμοιων patterns. Η μικρή διαφορά στην απόδοση ενθαρρύνει την χρήση του αλγορίθμου με νευρωνικά και γενετικό όταν θέλουμε πάρα πολύ άμεσα αποτελέσματα με χρήση μεγάλων χρονοσειρών που διαθέτουν πολλές ιστορικές τιμές.

Ο περιορισμός βέβαια του μέγιστου αριθμού γενεών μικρότερο του 50 δίνει εκτελείται γρηγορότερα σε σχέση με την εφαρμογή που αναλύθηκε στο προπορευόμενο κεφάλαιο 5.3 όπου παράχθηκαν 100 γενεές. Η απόδοση σε αυτή την περίπτωση δεν φάνηκε να διαφέρει ιδιαίτερα (έδωσε ίδια αποτελέσματα). Το γεγονός αυτό δείχνει ότι ο γενετικός συγκλίνει σχετικά πιο γρήγορα σε ένα καλό νευρωνικό μοντέλο. Άρα μια σχετική μείωση του πλήθους

γενεών της εφαρμογής νευρωνικού με γενετικούς αλγορίθμους φαίνεται να συγκλίνει στο χρόνο της εφαρμογής σύγκρισης-εύρεσης patterns του κεφαλαίου 5.4 .

Κεφάλαιο 6

Συμπεράσματα

Στην μελέτη αυτή εφαρμόστηκαν δύο μοντέλα για την πρόβλεψη χρηματιστηριακών χρονοσειρών. Στα μαθηματικά, μια χρονοσειρά είναι μια σειρά σημείων δεδομένων με χρονική σειρά. Ουσιαστικά μια χρονοσειρά είναι μια ακολουθία που λαμβάνεται σε διαδοχικά ίσα χρονικά σημεία. Έτσι είναι μια ακολουθία δεδομένων διακριτού χρόνου. Παραδείγματα χρονοσειρών είναι τα ύψη της παλίρροιας των ωκεανών, ο αριθμός των ηλιακών κηλίδων και η ημερήσια τιμή κλεισίματος του βιομηχανικού μέσου όρου Dow Jones.

Η ανάλυση χρονοσειρών περιλαμβάνει μεθόδους για την ανάλυση δεδομένων χρονοσειρών με σκοπό την εξαγωγή σημαντικών στατιστικών και άλλων χαρακτηριστικών των δεδομένων. Η πρόβλεψη χρονοσειρών είναι η χρήση ενός μοντέλου για την πρόβλεψη μελλοντικών τιμών με βάση τις προηγουμένως παρατηρηθείς τιμές. Συχνά χρησιμοποιούμε στατιστικές μεθόδους όπως παλινδρόμηση χρησιμοποιείται συχνά με τέτοιο τρόπο ώστε να ελέγχονται οι σχέσεις μεταξύ μιας ή περισσότερων διαφορετικών χρονοσειρών, αυτός ο τύπος ανάλυσης δεν ονομάζεται συνήθως "ανάλυση χρονοσειρών", η οποία αναφέρεται ειδικότερα σε σχέσεις μεταξύ διαφορετικών χρονικών σημείων μέσα σε μια ενιαία σειρά. Η ανάλυση διακοπτόμενων χρονοσειρών χρησιμοποιείται για τον εντοπισμό αλλαγών στην εξέλιξη μιας χρονοσειράς από πριν έως μετά από κάποια παρέμβαση που μπορεί να επηρεάσει την υποκείμενη μεταβλητή.

Τα δεδομένα χρονοσειρών έχουν μια φυσική χρονική σειρά. Αυτό κάνει την ανάλυση χρονοσειρών να ξεχωρίζει από τις συγχρονικές μελέτες, στις οποίες δεν υπάρχει φυσική σειρά των παρατηρήσεων.

Στην περίπτωση μας μελετήσαμε τρεις χρονοσειρές οι οποίες αφορούν μεγάλες τράπεζες της Ελλάδος. Είναι οι χρονοσειρές των τραπεζών Εθνική, ALPHA , Πειραιώς.

Εφαρμόστηκαν δύο αλγόριθμοι. Ο πρώτος χρησιμοποιεί ανατροφοδοτούμενα νευρωνικά με ένα γενετικό αλγόριθμο για τον εντοπισμό σε κάθε χρονική στιγμή του βέλτιστου νευρωνικού μοντέλου με στόχο την καλύτερη δυνατή πρόβλεψη.

Ο δεύτερος αλγόριθμος είναι ένα αλγόριθμος αναζήτησης όμοιων προτύπων όπου με βάση τον συντελεστή συσχέτισης υπολογίζει πόσο μοιάζει τμήμα της χρονοσειρά με το σημείο που θέλουμε να κάνουμε πρόβλεψη.

Και στις δύο περιπτώσεις γίνεται επαναληπτική εκτέλεση για κάθε σημείο και ανατροφοδοτείται με την κάθε επόμενη τιμή δίνοντας κάθε φορά την καλύτερη πρόβλεψη για την αμέσως επόμενη τιμή.

Τα αποτελέσματα ήταν αρκετά θετικά όπου είχαμε ένα μέσο τετραγωνικό σφάλμα αρκετά μικρό και στις 3 περιπτώσεις περίπου στο 6% και στις 3 χρονοσειρές.

Βιβλιογραφία

1. Lin, Jessica; Keogh, Eamonn; Lonardi, Stefano; Chiu, Bill (2003). "A symbolic representation of time series, with implications for streaming algorithms". 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery. New York: ACM Press. pp. 2–11. CiteSeerX 10.1.1.14.5597. doi:10.1145/882082.882086. S2CID 6084733.
2. Liao, T. Warren (2005). "Clustering of time series data - a survey". Pattern Recognition. Elsevier. 38 (11): 1857–1874. Bibcode:2005PatRe..38.1857W. doi:10.1016/j.patcog.2005.01.025. – via ScienceDirect (subscription required)
3. Aghabozorgi, Saeed; Shirkhorshidi, Ali S.; Wah, Teh Y. (2015). "Time-series clustering – A decade review". Information Systems. Elsevier. 53: 16–38. doi:10.1016/j.is.2015.04.007. – via ScienceDirect (subscription required)
4. Keogh, Eamonn J. (2003). "On the need for time series data mining benchmarks". Data Mining and Knowledge Discovery. Kluwer. 7: 349–371. doi:10.1145/775047.775062. ISBN 158113567X. S2CID 41617550. – via ACM Digital Library (subscription required)
5. Agrawal, Rakesh; Faloutsos, Christos; Swami, Arun (October 1993). "Efficient Similarity Search In Sequence Databases". 4th International Conference on Foundations of Data Organization and Algorithms. International Conference on Foundations of Data Organization and Algorithms. Vol. 730. pp. 69–84. doi:10.1007/3-540-57301-1_5. – via SpringerLink (subscription required)
6. Chen, Cathy W. S.; Chiu, L. M. (September 2021). "Ordinal Time Series Forecasting of the Air Quality Index". Entropy. 23 (9): 1167. Bibcode:2021Entrp..23.1167C. doi:10.3390/e23091167. PMC 8469594. PMID 34573792.
7. Sarkar, Advait; Spott, Martin; Blackwell, Alan F.; Jamnik, Mateja (2016). "Visual discovery and model-driven explanation of time series patterns".

2016 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC). IEEE. doi:10.1109/vlhcc.2016.7739668.

8. Bloomfield, P. (1976). Fourier analysis of time series: An introduction. New York: Wiley. ISBN 978-0471082569.
9. Shumway, R. H. (1988). Applied statistical time series analysis. Englewood Cliffs, NJ: Prentice Hall. ISBN 978-0130415004.
10. Sandra Lach Arlinghaus, PHB Practical Handbook of Curve Fitting. CRC Press, 1994.
11. William M. Kolb. Curve Fitting for Programmable Calculators. Syntec, Incorporated, 1984.
12. S.S. Halli, K.V. Rao. 1992. Advanced Techniques of Population Analysis. ISBN 0306439972 Page 165 (cf. ... functions are fulfilled if we have a good to moderate fit for the observed data.)
13. Barto, A. G. (1992). Reinforcement learning and adaptive critic methods. In D. A. White and D.
14. A. Sofge (eds.), Handbook of Intelligent Control: Neural, Fuzzy, and Adaptive Approaches, pp. 469–491. Van Nostrand Reinhold, New York.
15. Barto, A. G. (1995b). Reinforcement learning. In M. A. Arbib (ed.), Handbook of Brain Theory and Neural Networks, pp. 804–809. MIT Press, Cambridge, MA.
16. Gordon, G. J. (2001). Reinforcement learning with function approximation converges to a region. Advances in neural information processing systems.
17. Russell, S., Norvig, P. (2009). Artificial Intelligence: A Modern Approach. Prentice-Hall, Englewood Cliffs, NJ.
18. Poli, R., Langdon, W. B., McPhee, N. F. (2008), A Field Guide to Genetic Programming, Lulu.com, freely available from the internet ISBN 978-1-4092-0073-4
19. Sutton, Richard S. and Barto, Andrew G., Reinforcement Learning: An Introduction, MIT Press, 1998
20. Wiering, M., Van Otterlo, M. (2012). Reinforcement Learning. Springer Berlin Heidelberg

Παράτημα Ι

Οι αλγόριθμοι των εφαρμογών μας φαίνονται παρακάτω:

Αλγόριθμος ενότητας 5.2.1

```
import pandas as pd
import matplotlib.pyplot as pl
from scipy.stats import pearsonr

import random
from sklearn import tree
#import pydotplus
from sklearn.neural_network import MLPRegressor
from sklearn.metrics import mean_squared_error

import matplotlib.pyplot as plt
import matplotlib.image as pltimg
import time

def NN(x,k, lr,mom,L):
    X=[]
    Y=[]
    n=len(x)
    for i in range(300):
        X.append(x[n-i-k:n-i-2])
        Y.append(x[n-i-1])



    Xtrain=X[51:299]
    Ytrain=Y[51:299]
```

```
    clf = MLPRegressor(hidden_layer_sizes=L,momentum=mom,
learning_rate_init=lr,random_state=1, max_iter=300)
    clf = clf.fit(Xtrain, Ytrain)

    Xtest=X[0:50]
    Ytest=Y[0:50]

    Y2=clf.predict(Xtest)
    sc=mean_squared_error(Y2,Ytest)

    return sc



f=input("Give filename:")
sp=int(input("Give number of points to predict and compare:"))
df=pd.read_csv(f)




y=list(df["Close"])
start_time=time.time()
endt=len(y)
xpred=[]
for t in range(sp):
    x=y[0:endt-sp-2+t]
    crom=[]
    ev=[]
    MC=30
    N=50

    pcross=0.9
    pmul=0.8
```

```
print("start")

for i in range(0,MC):
    lr=random.random()
    nn=random.randint(3,10)
    L=(random.randint(5,100),)
    for n1 in range(nn-1):
        lev=random.randint(5,100)
        L=L+(lev,)

    k=random.randint(10,100)
    mom=random.random()
    c=[k, lr,  mom ,L ]
    crom.append(c)
    score=NN(x,k, lr,mom,L)
    ev.append(score)
    print (i)

for i in range (0,N):
    print("Gen: ",i)
    ev,crom=zip(*sorted(zip(ev,crom),reverse=False))
    ev=list(ev)
    crom=list(crom)
    if(random.random()<pcross):
        i1=random.randrange(0,int(MC/3))
        i2=random.randrange(0,int(MC/3))
        k=int((crom[i1][0]+crom[i2][0])/2)
        lr=(crom[i1][1]+crom[i2][1])/2
        mom=(crom[i1][2]+crom[i2][2])/2
        L1=crom[i1][3]
        L2=crom[i2][3]
        LL=[]
```

```
        if(len(L1)>=len(L2)):
            for ii in range(len(L2)):
                LL.append(int((L1[ii]+L2[ii])/2))
            for ii in range(len(L2)+1,len(L1)):
                LL.append(L1[ii])

        else:
            for ii in range(len(L1)):
                LL.append(int((L1[ii]+L2[ii])/2))
            for ii in range(len(L1)+1,len(L2)):
                LL.append(L2[ii])

        L=tuple(LL)
        c=[k,lr,  mom ,L]

        for j in range (0,4):
            crom[MC-2][j]=c[j]
        score=NN(x,k, lr,mom,L)
        ev[MC-2]=score

    if(random.random()< pmul):
        i1=random.randrange(0,int(MC/3))
        i2=random.randrange(0,3)
        k=[]
        k.append(random.randint(10,100))
        k.append(random.random())
        k.append(random.random())
        nn=random.randint(3,10)
        L=(random.randint(5,100),)
        for n1 in range(nn-1):
            lev=random.randint(5,100)
            L=L+(lev,)
        k.append(tuple(L))
```

```
        for j in range (0,4):
            crom[MC-2][j]=crom[i1][j]
        crom[MC-2][i2]=k[i2]
        i2=random.randrange(0,3)
        crom[MC-2][i2]=k[i2]

        score=NN(x,crom[MC-2][0],crom[MC-2][1],crom[MC-2][2],crom[MC-
2][3])
        ev[MC-2]=score


  ev,crom=zip(*sorted(zip(ev,crom),reverse=False))
  print("Best Solution:", crom[0])
  print("Evaluate Solution:", ev[0])

  k=crom[0][0]
  lr=crom[0][1]
  mom=crom[0][2]
  L=crom[0][3]

  X=[]
  Y=[]
  n=len(x)
  for i in range(300):
     X.append(x[n-i-k:n-i-2])
     Y.append(x[n-i-1])

  Xtrain=X[51:299]
  Ytrain=Y[51:299]
```

```
    clf = MLPRegressor(hidden_layer_sizes=L,momentum=mom,
learning_rate_init=lr,random_state=1, max_iter=300)
    clf = clf.fit(Xtrain, Ytrain)
    px=clf.predict([x[n-k+1:n-1]])
    xpred.append(px[0])

end_time=time.time()

print("Execution time:"+str(end_time-start_time))

t1=range(sp)
pl.plot(t1,y[len(y)-sp-1:len(y)-1],marker='o', label='real values')
pl.plot(t1,xpred,marker='o',label='predict')
pl.ylim(0,max(y)+2)
pl.title('Close and predict values')
pl.xlabel('t')
pl.ylabel('value')
plt.legend()
pl.show()
print(pearsonr(y[len(y)-sp-1:len(y)-1],xpred))
print(mean_squared_error(y[len(y)-sp-1:len(y)-1],xpred))
```

Αλγόριθμος Ενότητας 5.3

```
import pandas as pd
import matplotlib.pyplot as pl
from scipy.stats import pearsonr
```

```
import random
from sklearn import tree
#import pydotplus
from sklearn.neural_network import MLPRegressor
from sklearn.metrics import mean_squared_error

import matplotlib.pyplot as plt
import matplotlib.image as pltimg
import time



def cc(x,ti,n):
    n1=len(x)
    x1=x[ti:ti+n-1]
    x2=x[len(x)-n:len(x)-1]

    p=pearsonr(x1,x2)
    return p[0]



f=input("Give filename:")
sp=int(input("Give number of points to predict and compare:"))
df=pd.read_csv(f)




y=list(df["Close"])
endt=len(y)

start_time=time.time()
xpred=[]
for t in range(sp):
```

```
    print(t)
    x=y[0:endt-sp-2+t]

    N=[]

    mx=-10
    mxi=-1
    k=0
    for n in range(5,50):
        for i in range(len(x)-2*n):
            c=cc(x,i,n)
            N.append([i,c,n])
            if(c>mx):
                mxi=k
                mx=c
            k=k+1
    if(N[mxi][0]>0.7):
        mxx1=max(x[ N[mxi][0]:N[mxi][0]+N[mxi][2] ])
        mnx1=min(x[ N[mxi][0]:N[mxi][0]+N[mxi][2] ])
        lv1=mxx1-mnx1

        mxx2=max(x[ len(x)-N[mxi][2]:len(x)-1  ])
        mnx2=min(x[ len(x)-N[mxi][2]:len(x)-1  ])
        lv2=mxx2-mnx2

        nk=lv2/lv1

        xpred.append(nk*(x[N[mxi][0]+N[mxi][2]]-mnx1)+mnx2)
    else:
        xpred.append(x[len(x)-1])

end_time=time.time()
print("Execution time:"+str(end_time-start_time))
```

```
t1=range(sp)
pl.plot(t1,y[len(y)-sp-1:len(y)-1],marker='o', label='real values')

pl.plot(t1,xpred,marker='o', label='predict')
pl.ylim(0,max(y)+2)
pl.title('Close and predict values')
pl.xlabel('t')
pl.ylabel('value')
plt.legend()
pl.show()



print(pearsonr(y[len(y)-sp-1:len(y)-1],xpred))
print(mean_squared_error(y[len(y)-sp-1:len(y)-1],xpred))
```